Analyzing Molecular Interactions

Nothing can happen in biology unless something binds to something else. Although much of the effort in bioinformatics to date has focused on the detection of homology or the deduction of structure and/or function from sequence, in the long run, for bioinformatics to make a real contribution to drug design and cell biology, it will be necessary to be able to predict what other molecules a given gene product will bind to and how tight that binding will be. At present, no one knows how to do this routinely. This chapter presents some of the tools currently available for solving—or attempting to solve—various aspects of the problem.

Fundamental to any treatment of molecular interactions is recognition of the fact that, when anything tries to bind to the surface of a protein, it does so in the presence of a 55 M concentration of a competing ligand: water. The surfaces of protein molecules are coated with a layer of bound solvent about 1 to 2 molecules deep (Fig. 8.1.1). A typical protein will have at least 2 to 3 bound waters per amino acid, numerically, and although most will be on the surface, a few will be buried in cavities or at the interfaces between subunits. Displacement of bound solvent from a potential binding site can be easy or difficult; it seems reasonable to assume that the degree of difficulty must relate in some fashion to how tightly any ligand can bind to that site or whether the site is accessible to ligands at all. Yet, most computational approaches to analyzing or predicting ligand binding sites and affinities have either ignored the role of bound water or treated it in a very general way.

There is experimental evidence that a general treatment may be as bad as neglecting solvent altogether. Crystallographic analyses of protein structures in different solvents, or in the same solvent but in different crystal lattices, have suggested the existence of at least three different classes of water molecules on a protein surface. Tightly bound solvent molecules are observed under all conditions; disordered solvent molecules are either never observed or are found in only one or two structures of the same protein, indicating very weak binding. A third, intermediate class of waters appears in many but not all structures and has positions that vary somewhat from structure to structure, suggesting binding sites of intermediate strength. In a future unit for this chapter, methods to analyze these classes of water molecules will be presented, and will offer the intriguing suggestion that ligand binding sites primarily involve displacement of the intermediate waters rather than the other two classes (Mattos, 2002). It may be possible to rationalize this striking fact by considering solvent entropy and the contributions it makes to binding. Tightly bound waters are simply too strongly associated with the protein surface to be displaced; they basically occlude the sites they are on. Disordered solvent molecules can be displaced easily, but no entropy gain occurs when they are—they are already conformationally unrestricted. Intermediate waters, on the other hand, are not held so tightly that they cannot be displaced by a ligand, yet are held tightly enough that freeing them up to go into the solvent will produce an entropy gain that can help drive ligand binding thermodynamically. If this analysis is correct, it suggests that computational approaches to finding water sites of intermediate affinity could provide a means for identifying ligand binding sites on the surface of a protein, even when nothing is known about what binds there. Since it has also been shown that the locations of bound waters trace the conformation of bound ligands in the binding site, reliable methods for predicting solvent positions could also provide a first-pass outline for the design of drugs.

Once ligands can be modeled into protein binding sites—a task that sounds straightforward but is actually extremely difficult—the next step is determining affinity. Computational approaches to this problem have focused on analysis of the free energy, and, in this

Contributed by Gregory A. Petsko

Current Protocols in Bioinformatics (2003) 8.1.1-8.1.3

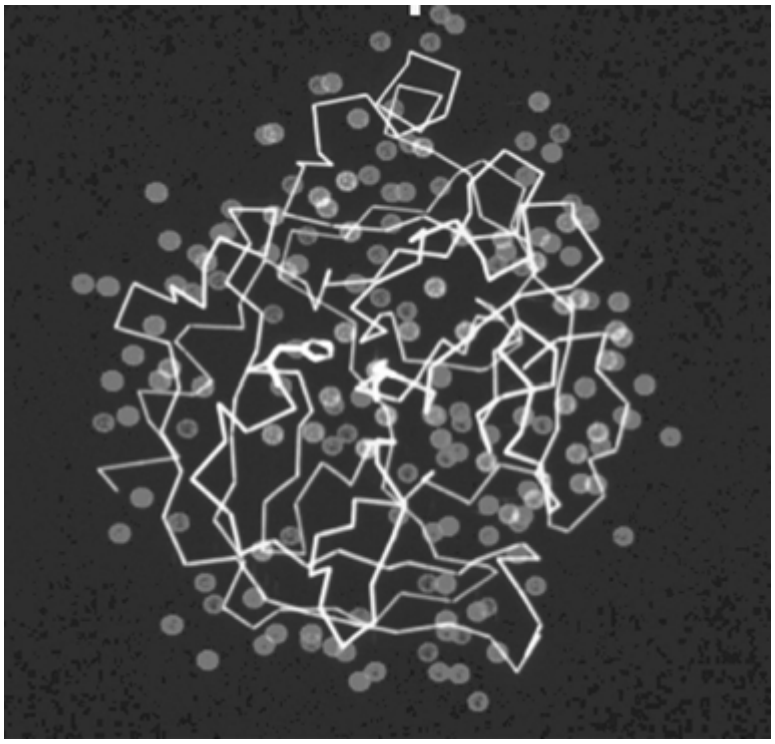


Figure 8.1.1 The crystal structure of the bacterial serine protease subtilisin with the bound water molecules observed crystallographically indicated as blue spheres. Figure courtesy of Dagmar Ringe, Brandeis University.

chapter, a future unit will discuss the available tools for such calculations (Mattos and Ringe, 2001). Once it was thought that computing affinities to within an order or magnitude or two would be satisfactory, but current methods aim to do much better than that—it is not unreasonable to expect an accuracy to within a few kilojoules or less in favorable situations. Coping with the effects of protein conformational changes and nonstandard binding modes is still the challenge for such calculations. Better methods for predicting these situations in advance are sorely needed.

Reliable tools for calculating electrostatic interaction energies are equally necessary. Of all the forces that exist between molecules, the electrostatic term is the hardest to compute accurately. One reason is that the charges, and partial charges, on the ionizable groups and polar groups involved are simply not known with certainty. Another reason is that the dielectric constant term in the familiar Coulomb potential term is almost impossible to estimate. The dielectric constant is really a property of bulk solvent, and whatever else one may say about the environment of a ligand binding site, it seems certain that in most respects it does not resemble bulk solvent. There is no universally agreed upon method for calculating the screening effect of solvent and protein atoms in the microenvironment of a protein binding pocket. Future units in this chapter will describe a number of approaches to the problem of correctly determining the electrostatic energies between two molecules (Sheinerman et al., 2000). These approaches offer hope of a forthcoming solution to this most thorny problem of estimating interaction energies.

Although the fields of drug design and metabolic biochemistry are mostly concerned with the interactions between proteins and small molecules, cell biology is more often interested in macromolecular interactions, usually those of proteins with one another. Bioinformatics is only beginning to tackle this most challenging problem: given the

structure (or ultimately, the sequence) of a gene product, how does one predict, from first principles or from comparative analysis, what other gene's product it will associate with, and at what sites and with what consequences (Al-Lazikani et al., 2001)? Cesarini and Gomez describe databases and tools that provide the first tentative steps in answering these questions (*UNIT 8.2*).

In the end, bioinformatics will need to address even more difficult questions than these in regard to interactions. Many protein complexes in the cell are dynamic; how are we to predict the lifetime of a complex? Can computationally determined affinities be related to on- and off-rates in the absence of experimental data? What about protein turnover rates: can they be predicted from sequence and/or structural information? Many proteins interact with one another only in the vicinity of the membrane: what are the effects of membranes on protein conformation and binding properties? Can we ever predict how a protein's structure and dynamics will change in response to ligand binding? These and other challenges for the future of bioinformatics will most likely require completely new methods for analyzing intermolecular interactions.

LITERATURE CITED

Al-Lazikani, B., Jung, J., Xiang, Z., and Honig, B. 2001. Protein structure prediction. *Curr. Opin. Chem. Biol.* 5:51-56.

Mattos, C. 2002. Protein-water interactions in a dynamic world. *Trends Biochem. Sci.* 27:203-208.

Mattos C. and Ringe, D. 2001. Proteins in organic solvents. *Curr. Opin. Struct. Biol.* 11:761-764.

Sheinerman, F.B, Norel, R., and Honig, B. 2000. Electrostatic aspects of protein-protein interactions. *Curr. Opin. Struct. Biol.* 10:153-1599.

Contributed by Gregory A. Petsko
Brandeis University
Waltham, Massachusetts

Prediction of Protein-Protein Interaction Networks

Shawn M. Gomez,[1] Kwangbom Choi,[2] and Yang Wu[1]

[1]Joint Department of Biomedical Engineering, University of North Carolina at Chapel Hill, Chapel Hill, North Carolina
[2]Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, North Carolina

ABSTRACT

This unit offers a general overview of several techniques that have been developed for inferring functional and/or protein-protein interaction networks. The majority of these use whole-genome sequences as their primary input source of data. In addition, a few methods that utilize both protein features and experimental protein-protein interaction data directly in the prediction of new interactions have recently been developed. While an exhaustive list of approaches is not presented, it is hoped that the reader will gain a sense of how these approaches are implemented and an idea of their relative strengths and weaknesses, and a broader perspective on the type of work being conducted in this highly active area of research. *Curr. Protoc. Bioinform.* 22:8.2.1-8.2.14. © 2008 by John Wiley & Sons, Inc.

Keywords: protein interactions • bioinformatics • interaction networks

INTRODUCTION

A significant challenge facing researchers today is determining how to extract and synthesize new knowledge from ever increasing amounts of data. While the quantity of these data is vast—for example, over 720 fully-sequenced genomes having been published—they promise to shed significant light on our understanding of biological systems. The information generated from more recent developments in high-throughput technologies, such as mRNA expression microarrays and yeast two-hybrid techniques (Eisen et al., 1998; Botstein, 1999; Ito et al., 2000; Uetz et al., 2000), only add to the opportunities and challenges of useful knowledge extraction and synthesis.

How can these data sources be mined for relevant information? A variety of tools already exist, many of which are described in this volume, to help researchers find useful relationships between biological entities. An obvious example is the software tool BLAST (*UNITS 3.3 & 3.4*), a standard method capable of linking one molecule to another strictly on the basis of sequence similarity (Altschul et al., 1997). If a new molecule of interest can be linked through homology to a protein or gene of known function, either with BLAST or other related tools, it is generally assumed that the function of the new molecule is the same or related. However, what happens if no high-similarity matches can be found? What if the most similar genes or proteins have no known function? As a recent example, in the *Fugu rubripes* genome, ~25% of genes have no relative in the genome of *H. sapiens* (Aparicio et al., 2002). In general, anywhere from 40% to 70% of gene sequences can be assigned a putative function through homology-based means, with prokaryotes being the best characterized (Eisenberg et al., 2000). As more genomes are sequenced, a large number of cross-organism gene similarities will surely be found; however, issues in determining their biological role will remain. Driven at least in part by these difficulties, recent work has focused on looking at the protein components of a cell from the viewpoint of both functional and physical interaction networks.

As structural building blocks, regulators of gene expression, and components of signaling pathways, proteins provide the core of what may be considered cellular function. Unlike genes, proteins will appear, disappear, and be modified within their appropriate cellular context, depending upon a variety of intra- and extracellular conditions, thus forming the cellular proteome. Determining protein function in this context is a challenging endeavor, with the

Current Protocols in Bioinformatics 8.2.1-8.2.14, June 2008
Published online June 2008 in Wiley Interscience (www.interscience.wiley.com).
DOI: 10.1002/0471250953.bi0802s22

ability to list putative interactions or functional relationships between proteins representing a significant accomplishment. In this context, it is clear that the computational identification of protein networks is particularly valuable for several reasons. For instance, it can help assign function to novel proteins, either through the discovery of a direct physical interaction with proteins of known function, or by "guilt through association," where putative function is assigned through network linkages. For proteins with known function, the determination of potentially new interactions can also lead to the discovery of novel functions. Predictions such as these can thus be particularly helpful in determining "high-likelihood" targets for future experimental effort. While of benefit when used alone, computational approaches for the prediction of protein interactions are particularly valuable as a companion to experimental techniques, as different experimental approaches are known to give an incomplete and often contradictory picture of protein relationships.

What follows is a general overview of several techniques that have been developed for inferring functional and/or protein-protein interaction networks. The majority of these approaches use whole-genome sequences as their primary input source of data. In addition, a number of methods that utilize both protein features and experimental protein-protein-interaction data directly in the prediction of new interactions have recently been developed. As a result, a slightly more detailed description of one such method is also provided. While an exhaustive list of approaches is not presented, it is hoped that the reader will gain a sense of how these approaches are implemented and an idea of their relative strengths and weaknesses, as well as some perspective on the type of work being conducted in this highly active area of research.

APPROACHES

Conservation of Gene Position

The organization of prokaryotic genomes into operons provides the foundation for some of the earliest attempts at predicting protein interactions (Dandekar et al., 1998; Overbeek et al., 1999). This method assumes that a physical interaction or even functional relationship between a pair of proteins provides selective pressure, helping to maintain gene order or the relative position of genes within the genome (Demerec and Hartman, 1959). The basic idea is that if a pair of genes are repeatedly observed together within a small region of DNA across multiple genomes, it is likely that the proteins expressed by these genes either interact or are functionally linked. Note that such an approach has only recently become possible, as it requires the comparison of completely sequenced genomes between which evolutionary distances are sufficiently large that genome rearrangements have had time to occur, yet not so large that significant numbers of orthologous genes have been lost.

For their study, Dandekar and colleagues (1998) looked at the ordering of genes in three separate sets of prokaryotic genomes: proteobacteria, Gram-positive bacteria, and *Archaea*. Within each genome set, clusters of genes were extracted and analyzed. Here, a cluster is defined as a group of genes which have the same order across the set of three genomes. Extracted genes were orthologous, but shared <50% sequence identity. By using this approach, ~100 genes appeared as conserved pairs or clusters for each of the three sets of genome triplets considered. In general, 75% of these gene pairs/clusters were known to interact. An additional 20% had at least some supporting evidence of a physical interaction. The remaining 5% either had no known function or there was no available evidence in support of an interaction.

A similar approach was presented in the work of Overbeek et al. (1999), which focused on comparing sets of gene "runs" from over 30 prokaryotic genomes. Within these runs, all genes were separated by gaps of no more than 300 bp and were required to lie on the same strand of DNA (Fig. 8.2.1); however, conservation of identical gene order across runs was not required. Genes within these runs were then compared for similarity. What was specifically sought were pairs of genes in one run that each have as their best hit a corresponding pair of genes within a run from another genome. In this approach, it is best if gene pairs are seen across many genomes and that those genomes are evolutionarily distant from each other. Using a phylogenetic distance-based scoring scheme, over 50,000 gene pairs were extracted and were then clustered into groups of co-occurring genes, presumably representing functionally related groups. While the overall effectiveness of this approach has not been fully quantified, a small example is instructive. In their analysis of the glycolytic pathway, Overbeek et al. (1999) found two distinct clusters of bacterial origin containing

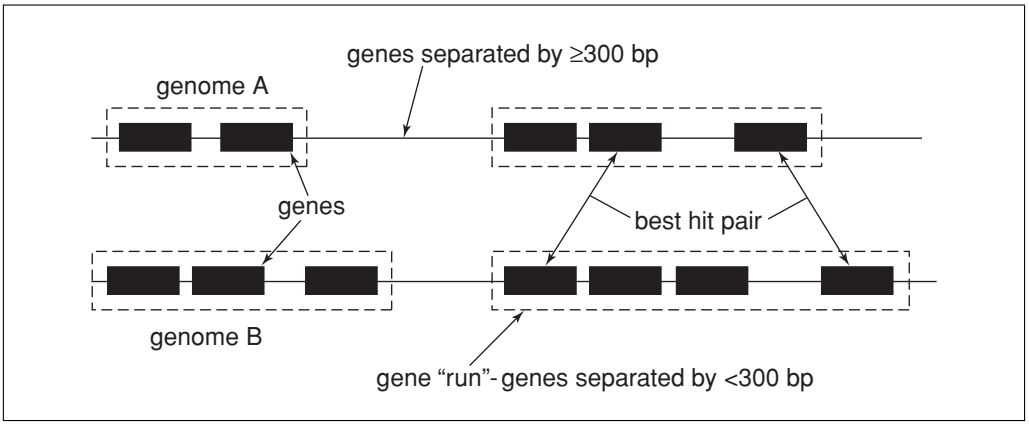


Figure 8.2.1 Diagram of conserved gene cluster approach used by Overbeek et al. (1999).

		Predicted interaction: yes	Predicted interaction: no
Real interaction	yes	true positive	false negative
	no	false positive	true negative

Figure 8.2.2 Commonly used descriptors of prediction accuracy. In this example, a true positive (TP) is one which the interaction is both known to exist and predicted to exist. A false positive (FP) is one in which the interaction is known not to exist, but predicted as existing. True (TN) and false negatives (FN) are the negatives of these conditions, respectively. Based on this table, the success rate or total accuracy is equal to (TP + TN)/(TP + FP + TN + FN); the sensitivity, TP rate, or recall is equal to TP/(TP + FN); the specificity or precision is equal to TP/(TP + FP); and the FP rate is equal to FP/(FP + TN).

a total of nine genes encoding glycolytic enzymes. The first cluster contained six genes, of which five were known, and had supporting evidence for being part of this pathway. The sixth protein was hypothetical, but was believed to be a transcriptional regulator, as it had homology to another hypothetical transcriptional regulator and weak similarity to the *deoR* family of transcriptional regulators. The second cluster contained three genes of which two, phosphofructokinase and pyruvate kinase, were supported as being part of an operon. The third gene in this cluster was the α chain of DNA polymerase III (*dnaE*). Overbeek et al. (1999) assumed that, while such a relationship was possible, the functional relationship of *dnaE* in this cluster was a false-positive result (Fig. 8.2.2).

The quality of these predictions is generally believed to be quite good for both approaches, and, in special cases, the accuracy of predictions may be $\geq$90%. While these approaches are obviously well suited to the study of prokaryotic genomes, recent work is beginning to suggest that they may also be useful in the discovery of functional linkages between genes within eukaryotes (Hallas et al., 1999; Wu and Maniatis, 1999; Lawrence, 2002).

Gene Fusion

It is often possible to find instances where proteins/domains observed as two separate and distinct molecules in one species appear as a single fused protein in another. Such fusion events typically bring together proteins involved in the same function or process, presumably for reasons of improved efficiency and regulation (Yanai et al., 2001). A common example is the fusion of two interacting subunits of the *Escherichia coli* DNA gyrase, Gyr A and Gyr B, into a single protein in yeast, topoisomerase II (Berger et al., 1996). Finding a fusion protein within a reference genome, and assuming that selective pressure is required for such a fusion event to occur, leads to the prediction that the two component

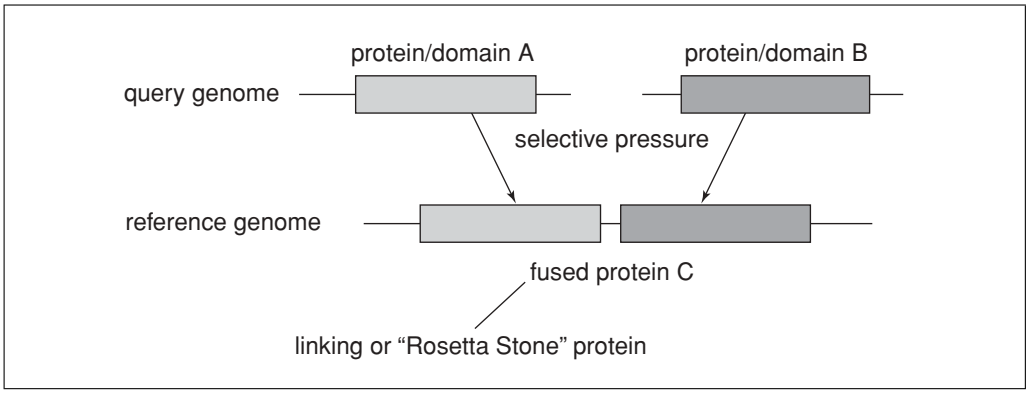


Figure 8.2.3 One method of gene fusion. Individual proteins, A and B, from one genome can often be found as a single fused protein, C, in another genome. The finding of such a fused protein suggests that protein A and B interact either physically or functionally.

proteins are likely to be physically or functionally associated (Fig. 8.2.3). This is the basis for the gene fusion or "Rosetta Stone" method (Enright et al., 1999; Marcotte et al., 1999).

By using BLAST (*UNITS 3.3 & 3.4*) to find fused proteins within a reference genome to which a pair of query proteins has significant similarity (but no similarity to each other), Enright et al. (1999) were able to find 215 proteins from *E. coli*, *H. influenzae*, and *M. jannaschii* involved in 64 unique fusion events. For this analysis, the precision was estimated at 75% (64 fusion events with 21 false positives: 64/85 × 100). Of the fused pairs with known function, most were metabolic enzymes.

In the work of Marcotte et al. (1999), two different approaches were used to search for fusion events in both *E. coli* and yeast. In the first approach, proteins were characterized in terms of their ProDom and Pfam (*UNIT 2.5*) domain composition and then compared to a similarly characterized set of reference proteins taken from SWISS-PROT (Corpet et al., 1998; Bateman et al., 1999). In this manner, they identified 3531 protein pairs in *E. coli* that could be linked through a fusion or Rosetta Stone protein found in SWISS-PROT. In the second approach, they used nonoverlapping regions of high sequence similarity rather than domains, and for *E. coli*, they found 4487 potentially interacting protein pairs. It is interesting to note that most pairs could be identified by only one of these approaches, with only 1209 pairs identified by both methods. Prediction accuracy was assessed by comparing annotations (finding annotation keywords shared by both proteins), database searches (looking for experimental evidence of the interaction within appropriate databases), and phylogenetic profiles (described below). The total accuracy was estimated to be on the order of 65%. By filtering out "promiscuous" domains, for instance the SH2 domains which are known to be present in many unrelated proteins, the total number of predicted interactions in *E. coli* dropped from 3531 to 749, with a corresponding estimated 47% improvement in accuracy.

Protein Phylogenetic Profiles

If proteins are functionally linked and thus involved as a group in a particular process, pathway, or structure, it may be expected that their evolution would also be linked; specifically, their pattern of inheritance would be identical—i.e., two such proteins would always be either inherited together as a pair or not at all. For instance, one would expect the protein components of a flagellum to be inherited together, with loss of one or more components resulting in a nonfunctional structure. This pattern of inheritance is the basis of the phylogenetic profile method, and was first used for generating profiles for all *E. coli* proteins against 16 other fully sequenced genomes (Pellegrini et al., 1999).

With this method, a profile is created for each protein in a target genome. As depicted in Figure 8.2.4, the profile itself consists of an *n*-character string, where *n* is the number of genomes used in the comparison and the *i*th position of the string corresponds to the *i*th genome. The absence or presence of a homolog to the protein in each of the surveyed genomes is marked in the string with a zero or one, respectively. After profiles have been generated for each protein, the proteins are clustered together according to the similarity of their profiles. Proteins having identical or nearly identical profiles (Pellegrini et al., 1999, also looked at profiles differing at only a single position or "bit") are then predicted to be functionally linked.

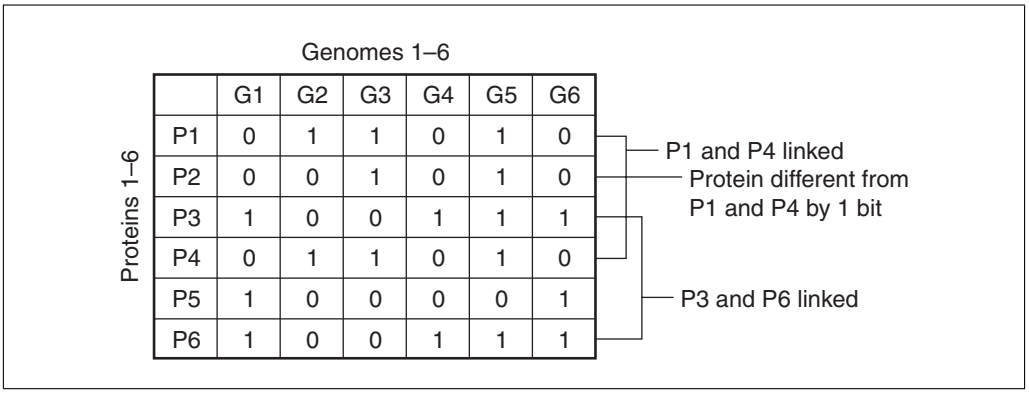


Figure 8.2.4 The phylogenetic profile method. Genomes (G1 to G6) are searched for the absence (0) or presence (1) of proteins (P1 to P6). Genes with identical profiles, or perhaps differing at a single position, can be linked into functionally related groups.

As one example of the accuracy of this method, the profile for the ribosomal protein RL7 was studied. Four other proteins across 16 genomes were found to have identical profiles, with three of the four being known to have ribosome-associated function. There were 27 profiles that differed by a single bit. Of these, 15 were also known to have function related to RL7. Thus, in this particular example, at least 60% of the predictions were assumed to be accurate. This approach was also evaluated as part of a study attempting to predict protein function on a genomic scale (Marcotte et al., 1999). Using *S. cerevisiae* as the model system, they estimated a false-positive rate of 30% and the ability to successfully predict known functional interactions at 33%.

A promising aspect of this method is that as the number of completely sequenced genomes increases, the number of unique profiles grows exponentially. For n complete genomes, there are 2^n possible profiles, rapidly increasing the discriminative power of this approach. In addition, with the expected significant growth in the number of eukaryotic organisms sequenced, the applicability of this method will grow significantly. A disadvantage of this method is the large number of false positives that are often generated. However, recent work by Barker and Pagel (2005) has improved on this basic approach. By using a maximum likelihood approach that incorporates phylogenetic tree information, a 35% improvement in the prediction of functional protein linkages was achieved.

Coevolution and Correlation of Phylogenetic Distances

The previous phylogenetic profile method is based on the idea of a coevolutionary process where the pattern of inheritance of certain sets of proteins is shared across species. Similarly, at the sequence level, coevolutionary processes have also been proposed as occurring between interacting protein pairs. Here, the premise is that interacting proteins must coevolve with one another so as to maintain their functionality and/or ability to interact with one another. Such coevolution can be detected through the similarity of their phylogenetic trees and has been proposed/shown to occur for a number of protein families (e.g., Moyle et al., 1994; Fryxell, 1996; Goh et al., 2000).

In these approaches, a sequence alignment/phylogenetic tree is generated for each potentially interacting protein family, after which the problem becomes one of tree comparison. While a number of methods have been developed for the comparison of phylogenetic trees, it turns out that only the simplest approach has generally been adopted, which involves the comparison not of the trees, but rather their underlying distance matrices. Specifically, the Pearson correlation coefficient between distance matrices is calculated, with high correlations indicating high degrees of similarity and hence coevolution. This approach is commonly referred to as the "mirrortree" method and has been estimated to generate predictions with 66% true positives for protein family pairs showing correlations >0.8 (Pazos and Valencia, 2001).

This approach has since undergone additional development, with a major improvement being the subtraction of the inherent similarity between trees that arises from the fact that the members of the two protein families being compared are each drawn from the same set of branches from the "tree of life" ("tol"; Pazos et al., 2005). This inherent similarity is corrected by subtracting background correlations between 16S rRNA orthologs (from

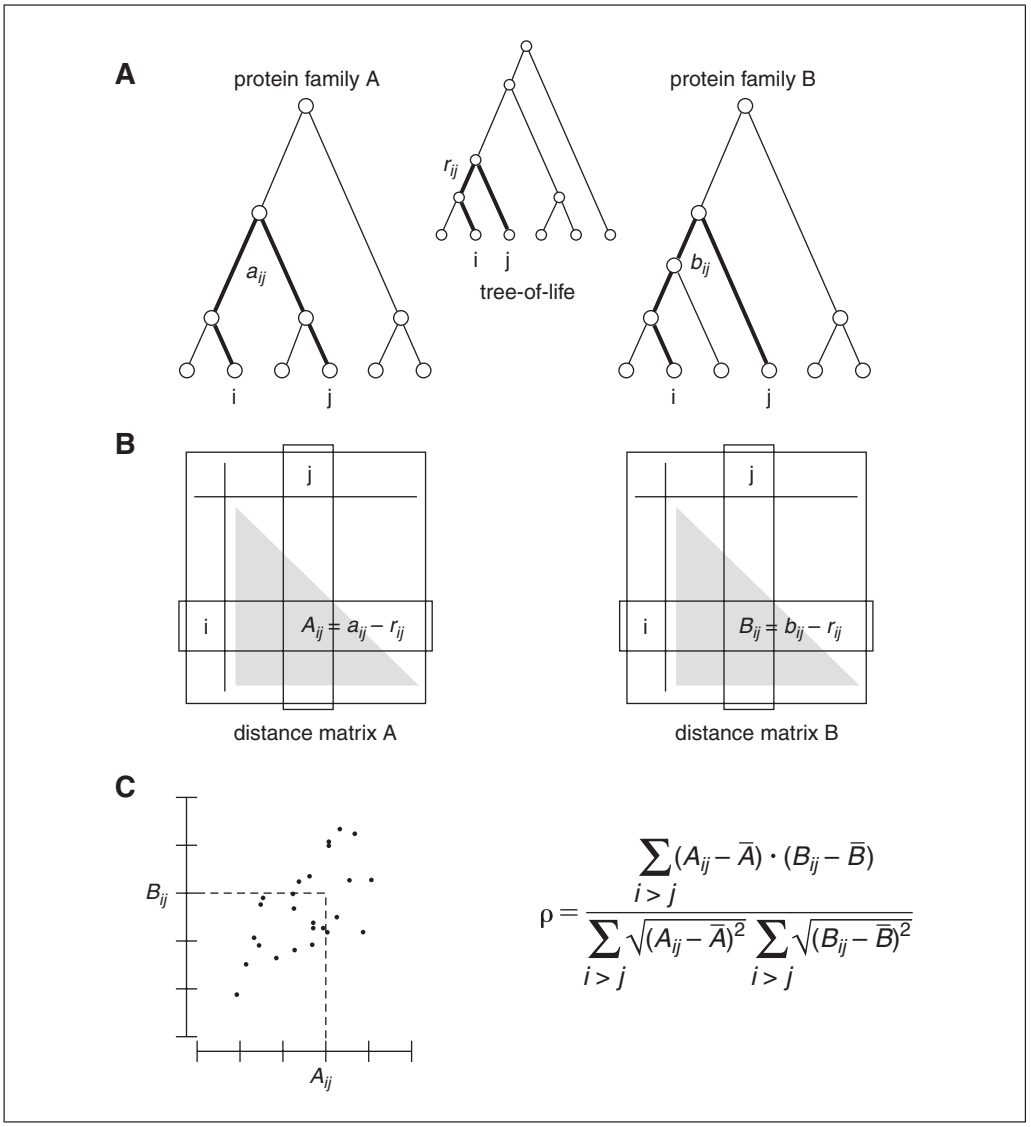


Figure 8.2.5 Coevolution and correlation of phylogenetic distances. (**A**) Trees or sequence alignments of two possibly interacting protein families are first generated along with the 16S ribosomal RNA sequence alignments for the same taxa. (**B**) Distance matrices are generated from the alignments (with tree-of-life distances subtracted from the distance matrices in the case of the tol-mirrortree approach) and the correlation (**C**) between matrices determined, typically, using the Pearson correlation coefficient.

the same taxa as the protein families) from the protein family correlations. Similarly, use of partial correlation has been suggested for such corrections (Sato et al., 2005). A general schematic of the "tol-mirrortree" approach is shown in Figure 8.2.5.

This approach has also been applied to the coevolution of protein domains (Jothi et al., 2006), has been extended to handle larger-sized trees (Jothi et al., 2005), has been used to try and infer binding specificity (Ramani and Marcotte, 2003), and has incorporated basic tree topology information and been implemented using supervised learning approaches (Craig and Liao, 2007). Recently, Yeang and Haussler (2007) developed a full continuous-time Markov process model describing sequence coevolution and used it to detect coevolution within and between protein domains.

Probabilistic Prediction of Interaction Networks

Due to the increased availability and use of high-throughput methods, there has been a rapid rise in the amount of experimental protein-protein interaction data available for the study of molecular systems. As the quantity and quality of these data grow, methods capable of extracting useful information are becoming increasingly valuable. As a result, a

number of projects have begun to investigate the use of this interaction data, in combination with various types of protein features, for inferring the existence of protein-protein interactions (e.g., Bock and Gough, 2001; Gomez et al., 2001, 2003; Sprinzak and Margalit, 2001; Wojcik and Schachter, 2001; Deng et al., 2002; Gomez and Rzhetsky, 2002; Riley et al., 2005).

The use of protein features is based on the assumption that in order for a protein interaction to occur, at least one pair of features—i.e., one in the upstream and one in the downstream protein—is necessary to establish the interaction. Features can be anything from stretches of identical charge to structural domains or motifs (e.g., a protein kinase domain). Implicit in this assumption is that features are basic units of protein function—i.e., they are independent, evolutionarily conserved "modules" that can be assembled into a variety of forms, providing the diversity observed within protein utility today. As a result, knowledge of feature pairs that are known to interact in one species is potentially transferable to similar pairs found in another.

A large number of such features have been studied and catalogued, the Pfam (*UNIT 2.5*) and InterPro (*UNIT 2.7*) databases being typical examples, with most accessible through the Web (Bateman et al., 1999; Apweiler et al., 2001; also see Chapter 2 for details). Likewise, online databases such as the Database of Interacting Proteins (DIP) and the Biomolecular Interaction Network Database (BIND), combined with published datasets extracted from multiple sources, have greatly facilitated the development of this approach (Xenarios et al., 2000; Bader et al., 2001; also see Internet Resources). As noted, several groups have made investigations using such data types. To give a better idea of how at least one of these methods work, the approach described by Gomez et al. (2001, 2003) will be described in greater detail. The results described here are generally relevant to other methods (e.g., see Sprinzak and Margalit, 2001 or Deng et al., 2002), and the interested reader is encouraged to access the literature for more details on the differences between implementations (see General Observations and Strategies below for more on why this is a good idea).

A probabilistic model

The method described here is a probabilistic one, and is based on the representation of a protein network as a graph, with proteins forming the vertices and interactions represented as the edges between them. It consists of two components, one for assigning a probability to each edge between proteins (a local property), and one for generating a probability for each particular "shape" (a global property, namely the particular arrangement of edges connecting all proteins) that the network can take. These two components can be combined, basically through multiplication of their respective probabilities, to give the final probability of any particular network. In practice, approaches such as this one use a large set of interaction data, often called a "training" set, in the generation of model parameters. After training, predictions of interactions for a new set of proteins can then be made.

As the first step of this approach, component domains are found for each protein in the network (Fig. 8.2.6). Next, for each protein-connecting edge, counts are taken of every unique domain-domain interaction. In the end, what is produced is a matrix of counts detailing how many times a domain of type X was found in an interaction with a domain of type Y. This matrix of counts can now be converted into a matrix of domain-domain probabilities through a variety of methods.

An important assumption in this model is that, in the absence of any data, an edge between any two proteins is possible. Specifically, it is supposed that the initial or "prior" probability of interaction between any two domains is equal to 0.5—i.e., the toss of an unbiased coin. This choice is based on a number of considerations, one of which is that it allows for both "attractive" (probabilities >0.5) and "repulsive" (probabilities <0.5) domain-domain interactions. As counts of particular domain-domain interactions increase, the probability of a particular interaction moves away from 0.5, the equivalent of adding bias to the coin. In the conversion of each element of the count matrix into a probability, if a particular domain-domain interaction has never been observed, this assumption requires that a 0.5 value be used for its probability of interaction. Similarly, if a set of likely negative interactions can be provided (interactions between proteins that are thought not to occur), it becomes possible to generate probabilities <0.5. This is done under the assumption that if one observes a pair of domains occurring in many proteins that are thought not to interact, then the domain pair is actually predictive with regard to the absence of an interaction, thus lowering the probability below 0.5. Given

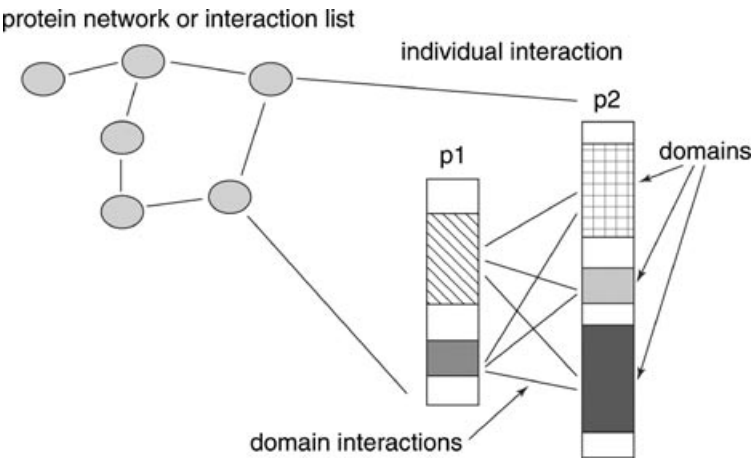


Figure 8.2.6 Extraction of domain data for the prediction of protein interactions. Given a set of protein interactions, all individual domain-domain interactions are extracted and counted. After training, counts are converted into probabilities of domain-domain interaction as well as protein-protein interaction. In the second stage, network topology is incorporated to improve predictions. See text for details.

this matrix of domain-domain probabilities, a probability of interaction between a pair of proteins can easily be produced—e.g., by taking the average. Probabilities between 0.5 and 1 support an interaction, with probabilities closer to 1 representing increased confidence in its existence. Probabilities less than 0.5 represent the absence of an interaction.

While not a detailed description of the first part of the model, it should be clear that it is now possible to use protein features and protein interaction data together to generate predictions. Given a set of proteins with domains, but without knowledge of any associations, a probability can now be assigned to all possible interactions between them based on knowledge extracted from the training data. A domain-pair that is enriched within the training data will provide greater support for an interaction between a new pair of proteins sharing the domain-pair, increasing its probability above 0.5. As will be discussed later, increasing the amount and quality of both interaction and domain data is an important factor in the accuracy and coverage of predictions.

Network topology

By itself, the process just described can be used to predict protein-protein interactions, assigning probabilities to all possible pair-wise interactions. However, a unique aspect of this approach is the addition of information concerning the structure or topology of the network into the generation of predictions. Here, topology refers to the shape of the network being studied, and, in this case, the generation of the connectivity distribution. This distribution gives the probability $P(k)$ of a protein having k edges or interactions. When this is plotted in log-log coordinates with the number of edges on the x axis and probability on the y axis, it becomes apparent that the plot is essentially linear with a negative slope (Fig. 8.2.7). This distribution suggests that the majority of proteins will have very few connections, while a very small percentage will be very highly connected. What is interesting is that this particular type of distribution—a power-law distribution—has been found for a number of biological (e.g., metabolic) as well as man-made (e.g., World Wide Web power grid) networks (Barabasi and Albert, 1999; Jeong et al., 2000). It also implies that networks of this type can be characterized as being "scale-free"—i.e., a network with this property will look the same across multiple scales. If a subnetwork is extracted out of a much larger network, the connectivity distribution will look identical for each. Protein interaction networks have also been shown to be scale-free, and thus share these as well as other properties (Gomez et al., 2001; Jeong et al., 2001).

Alone, this topology information can be used as a guide in predictions by giving higher probabilities to those networks that look "more biologically realistic," thus helping to filter erroneous predictions, especially with regard to false positives. This is particularly important since, as is probably becoming quite evident, all methods are capable of generating

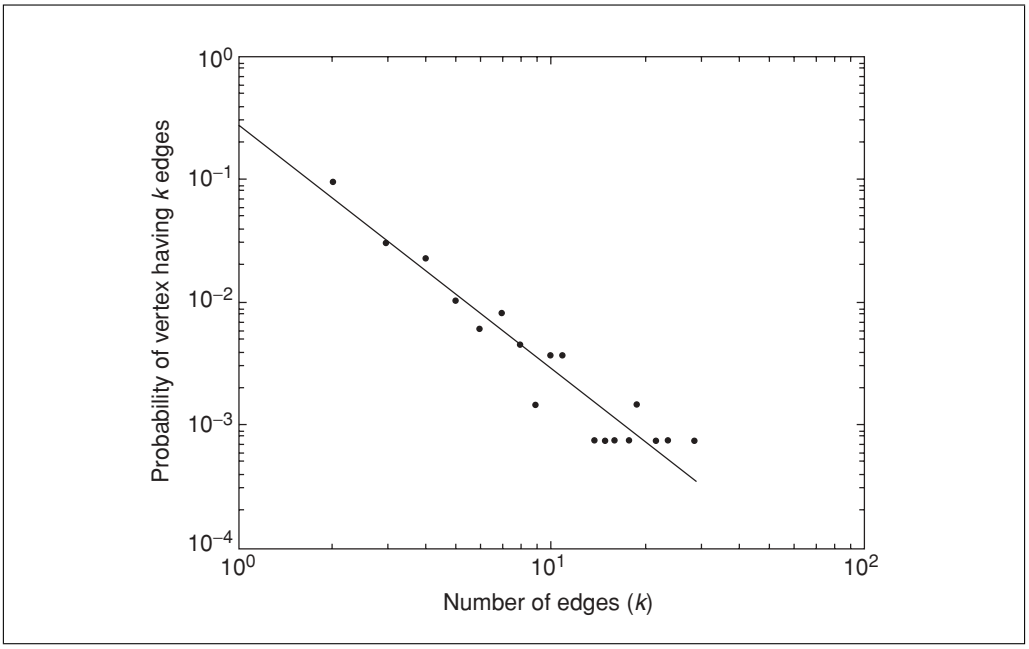


Figure 8.2.7 A sample connectivity distribution for a yeast protein network extracted from the DIP database (see Internet Resources). The majority of proteins will have few interactions (left end of the x axis); however, a few will be highly connected (right end).

large numbers of predictions with a significant portion being potentially wrong. The inclusion of topology into this model is one way to reduce the noise generated by these errors and focus predictions onto those networks that are more biologically relevant.

Finally, it is possible to combine the probabilities of a group of interactions with that of a network topology so that a probability for the complete network can be generated. As a result, different hypothesized networks, consisting of both known and predicted edges, can be directly compared with more likely ones chosen for further investigation.

Prediction

This approach was tested by attempting to predict *S. cerevisiae* protein-protein interactions (Gomez et al., 2001). In this case, protein features were based on Pfam (*UNIT 2.5*), and a total of 642 interactions were used for training and testing (Bateman et al., 1999). While more interaction data were available, the amount used here was limited due to a number of factors, including the requirement that both proteins in an interaction used either for training or predictions must have at least one domain. In addition, the number of domains that can be found in these data is dependent on the cutoff threshold used.

The effectiveness of this technique was assessed with the use of cross-validation, a common and extremely useful technique used for evaluating the effectiveness of a method when data are limited (Witten and Frank, 2002). In cross-validation, the data set is generally broken into equally sized subsets, or folds, with all but one of these folds being used for training. Predictions are then performed on the single remaining fold. In an iterative manner, predictions are made for each fold. The accuracy of predictions described here was assessed using leave-one-out cross-validation, where all but one edge from the data are used for training, and then a prediction is made as to whether the remaining edge exists or not. All edges are predicted in turn, and afterwards the total accuracy is assessed.

For all 642 edges in the network, it was found that 93% of these edges could be predicted, representing true positives. The remaining 7% represent false negatives. False-positive predictions were similarly assessed and found to lie at ~10%, and, thus, correctly predicted true negatives were at 90% (ROC score of ~0.65). The use of negative information was found to greatly improve predictions, with ROC scores improving to 0.7 (Gomez et al., 2003). In addition, if one of the proteins had been observed as having other interactions before (i.e., the model had been trained on other known interactions for a given protein), accuracy of predictions of that protein with a new/novel protein were greatly increased (ROC scores > 0.8). Note that it cannot be said for sure if all false positives are actually

incorrect predictions, as it is possible that some of these predictions are real, though currently unknown, interactions.

Summary

Limitations of this approach arise primarily from the fact that at this time, not all proteins have identifiable domains that can be used as features (Gomez and Rzhetsky, 2002). As a result, only a portion of available interaction data can be used for training. Also, predictions can only be made for those proteins that have at least one domain. To bypass this issue, different types of features capable of providing better coverage are currently being investigated. In addition, while growing rapidly, interaction data are only now becoming of sufficient quantity that high-confidence predictions can be made. As it does, however, the quality of predictions should improve rapidly. For instance, true positive rates in excess of 90% are possible if the interaction data set is large enough that there is a high degree of redundancy for multiple domain-domain pairs (also see Sprinzak and Margalit, 2001).

This approach is readily applicable to eukaryotic genomes and can integrate data derived from different sources into a single prediction. A major benefit of this approach is that it provides a probability for any given interaction. A researcher can instantly identify the relative strength of predictions and then decide which are worth investigating further. In addition, since this approach is probabilistic in nature, it is quite easy to integrate additional information into the prediction. For instance, knowledge of the localization of a protein to particular regions of a cell can help improve predictions; if it is known that two proteins are found in the same subcellular compartment, the probability of an interaction should increase, or at least stay the same. Also, given the variable accuracy of current interaction data, a probabilistic framework provides a natural mechanism for dealing with these uncertainties.

Predicting Protein Interactions Through Data Integration

As large quantities of both experimental and computational data are continuing to accumulate, a major challenge is how to combine these data into accurate predictions of protein interaction/function. With the exception of the just discussed probabilistic methods, the gene position, gene fusion, phylogenetic profiles, and coevolution methods are essentially "stand-alone," and do not combine multiple and/or different types of data.

Thus, in an effort to make more accurate and comprehensive predictions, recent efforts have focused on the problem of how to combine multiple types of genomic data into a single consensus prediction. Note that within these various data types are both direct and indirect information regarding protein interactions. As an example of indirect information, it has been shown that interacting proteins often show a high degree of coexpression. Thus, correlations in gene-expression data can also be predictive of protein interactions. As another example, yeast two-hybrid data, while providing direct information regarding protein interactions, are known to be a "noisy" data type prone to large numbers of false positives (Sprinzak et al., 2003). In the rest of this section, we give a brief overview of three data integration approaches that have been used in the prediction of protein interactions: the Bayes classifier (Jansen et al., 2003), support vector machine approaches (Ben-Hur and Noble, 2005), and decision tree methods (Zhang et al., 2004).

One of the earliest efforts to predict protein interactions through the integration of different data types was performed by Jansen et al. (2003). In their work, five genomic data sets (mRNA coexpression, MIPS and GO biological function, and information as to whether proteins are essential to survival) were combined through a Naïve Bayes network (BN). Separately, four high-throughput interaction datasets consisting of yeast two-hybrid and in vivo pull-down experiments were integrated with a fully connected Bayesian network (which does not assume independence between datasets). Finally, both sets were again integrated through another Naïve Bayes network. Results from this work indicate that evidence for an interaction arising from any single data source did not have sufficient "weight" or sufficiently high likelihood to be predicted—i.e., no interactions were predicted. On the other hand, 9897 interactions were predicted for the combined genomic features data set, with another 163 interactions predicted from the integrated high-throughput experiments.

Since then, similar studies have been carried out on the prediction of protein interactions by integrating different genomic features using Bayesian approaches. For example, Lu et al. (2005) explored the limits of genomic evidence integration by combining 16 different genomic features using a boosted Naïve Bayes classifier to predict protein interactions in yeast. Rhodes et al. (2005) looked at the prediction of human protein interactions

by integrating different genomic data using a naïve Bayes classifier. Scott and Barton (2007) also conducted a similar human study by integrating different evidences including orthology, functional associations, and local network topology. By using each evidence itself, the ROC100 ranges from 0 to 0.032 and the number of predicted interactions ranges from 0 to 4830 at a posterior odds ratio greater than 1. However, by integrating different evidences, the most accurate predictor has a much higher ROC100 of 0.094 with 34780 interactions identified at a posterior odds ratio greater than 1.

Non-Bayesian data integration models are also of interest and have the often desirable property of not needing any prior information or discretization of the raw genomic data. Bock and Gough (2001) combined known protein interactions collected from different experiments using a support vector machine (SVM) classifier to predict protein interactions based on primary structure and associated physicochemical properties. Gomez et al. (2003) also looked at the use of SVMs for interaction prediction, and compared them to probabilistic methods. Later, Ben-Hur and Noble (2005) proposed to predict protein-protein interactions in yeast by combing data sources including protein sequences, Gene Ontology annotations, local properties of the network, and homologous interactions using different SVM kernels. In this work, the classifier was able to predicted 80% of the known true positive interactions at a false positive rate of 1%.

Decision tree approaches are another widely used non-Bayesian data integration method, and the importance of different data types can be easily assessed through these methods. In the work of Zhang et al. (2004), a probabilistic decision tree was used to predict proteins in the same complex in yeast by integrating different gene or protein characteristics. Lin et al. (2005) utilized a random forest method for an integrated prediction of protein-protein interactions in yeast. They showed that, although computationally more expensive, the random forest method had better performance over the logistic regression and BN approaches. More specifically, random forests based on the MIPS and GO information gave highly accurate classifications (classification error = 2.76%), and adding other genomic data types did little for improving prediction (classification error = 3.95%).

OBSERVATIONS AND CONCLUSIONS

The approaches described here provide the ability to extract potentially useful information on interactions, both functional and physical, from increasingly common large-scale high-throughput data sets. With their aid, researchers may be better able to cut away extraneous or otherwise confusing information, focusing in on the most relevant aspects of a given process. Highly promising predictions can be followed up with direct experimentation.

A number of challenges currently exist, however, not the least of which is properly assessing the accuracy of these approaches. Which method is best? It must be emphasized that determining the effectiveness of any single method is often an extremely difficult task. Generally, large amounts of trusted, experimentally verified interaction data are not available at this time. Also, deciding whether a predicted interaction is in fact real is often impossible without further experimental work. Some of these challenges are highlighted by von Mering et al. (2002), who analyzed several experimental and computational approaches (gene order, phylogenetic profiles, and, gene fusion) and tried to assess their efficacy in predicting protein-protein interactions. Using all methods, and evaluating predictions with a trusted set of yeast protein complexes as a reference, they found over 80,000 potential interactions. However, only 2,400 were supported by more than one method. Computational methods were extremely competitive with experimental approaches. Even so, predictions from any single computational method could only be confirmed for 10% of the trusted interaction set. If three methods were combined, however, over 70% of predictions could be confirmed as being accurate.

Thus, an important observation that should be made is that none of these methods are exclusive. In fact, it should be assumed that it is necessary to use multiple, complementary methods. Different approaches have different biases, and these can be used to maximize the coverage of predictions. Similarly, understanding these biases will aid in the accurate assessment of the reliability of predictions. Thus the data integration approaches discussed earlier will take on an increasingly important role in future methodologies.

These results also highlight the importance of rigorous validation on appropriate test data.

If the performance of a method can be well characterized on a test data set, it is much easier to assess the confidence of predictions on novel data, as well as to compare predictions from different methods. Managing tradeoffs in performance will also be assisted. For example, increasing the accuracy of predictions will have the effect of decreasing the coverage, with fewer total predictions being generated.

Great emphasis is currently being placed on attempting to understand how biological systems regulate and control their behavior. Understanding this regulation requires a deeper appreciation for the relationships between genes, proteins, and other cellular components. While still in their infancy, the techniques presented here should help provide useful insight into the structure, dynamics, and function of biological systems.

LITERATURE CITED

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D.J. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* 25:3389-3402.

Aparicio, S., Chapman, J., Stupka, E., Putnam, N., Chia, J.M., Dehal, P., Christoffels, A., Rash, S., Hoon, S., Smit, A., Gelpke, M.D., Roach, J., Oh, T., Ho, I.Y., Wong, M., Detter, C., Verhoef, F., Predki, P., Tay, A., Lucas, S., Richardson, P., Smith, S.F., Clark, M.S., Edwards, Y.J., Doggett, N., Zharkikh, A., Tavtigian, S.V., Pruss, D., Barnstead, M., Evans, C., Baden, H., Powell, J., Glusman, G., Rowen, L., Hood, L., Tan, Y.H., Elgar, G., Hawkins, T., Venkatesh, B., Rokhsar, D., and Brenner, S. 2002. Whole-genome shotgun assembly and analysis of the genome of *Fugu rubripes*. *Science* 297:1301-1310.

Apweiler, R., Attwood, T.K., Bairoch, A., Bateman, A., Birney, E., Biswas, M., Bucher, P., Cerutti, I., Corpet, L.F., Croning, M.D., Durbin, R., Falquet, L., Fleischmann, W., Gouzy, J., Hermjakob, H., Hulo, N., Jonassen, I., Kahn, D., Kanapin, A., Karavidopoulou, Y., Lopez, R., Marx, B., Mulder, N.J., Oinn, T.M., Pagni, M., Servant, F., Sigrist, C.J., and Zdobnov, E.M. 2001. The InterPro database, an integrated documentation resource for protein families domains and functional sites. *Nucleic Acids Res.* 29:37-40.

Bader, G.D., Donaldson, I., Wolting, C., Ouellette, B.F., Pawson, T., and Hogue, C.W. 2001. BIND—The Biomolecular Interaction Network Database. *Nucleic Acids Res.* 29:242-245.

Barabasi, A.L. and Albert, R. 1999. Emergence of scaling in random networks. *Science* 286:509-512.

Barker, D. and Pagel M. 2005. Predicting functional gene links from phylogenetic-statistical analyses of whole genomes. *PLoS Comput. Biol.* 1:e3.

Bateman, A., Birney, E., Durbin, R., Eddy, S.R., Finn, R.D., and Sonnhammer, E.L. 1999. Pfam 3.1: 1313 multiple alignments and profile HMMs match the majority of proteins. *Nucleic Acids Res.* 27:260-262.

Ben-Hur, A. and Noble, WS. 2005. Kernel methods for predicting protein-protein interactions. *Bioinformatics* 21:i38-i46.

Berger, J.M., Gamblin, S.J., Harrison, S.C., and Wang, J.C. 1996. Structure and mechanism of DNA topoisomerase II. *Nature* 379:225-232.

Bock, J.R. and Gough, D.A. 2001. Predicting protein—protein interactions from primary structure. *Bioinformatics* 17:455-460.

Botstein, D. 1999. Of genes and genomes. *Ann. N.Y. Acad. Sci.* 882:32-41.

Corpet, F., Gouzy, J., and Kahn, D. 1998. The ProDom database of protein domain families. *Nucleic Acids Res*. 26:323-326.

Craig, R.A. and Liao, L. 2007. Phylogenetic tree information aids supervised learning for predicting protein-protein interaction based on distance matrices. *BMC Bioinformatics*. 8:6.

Dandekar, T., Snel, B., Huynen, M., and Bork, P. 1998. Conservation of gene order: A fingerprint of proteins that physically interact. *Trends Biochem. Sci.* 23:324-328.

Demerec, M.E. and Hartman, P. 1959. Complex loci in microorganisms. *Annu. Rev. Microbiol.* 13:377-406.

Deng, M., Mehta, S., Sun, F., and Chen, T. 2002. Inferring domain-domain interactions from protein-protein interactions. *Genome Res*. 12:1540-1548.

Eisen, M.B., Spellman, P.T., Brown, P.O., and Botstein, D. 1998. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. U.S.A.* 95:14863-14868.

Eisenberg, D., Marcotte, E.M., Xenarios, I., and Yeates, T.O. 2000. Protein function in the post-genomic era. *Nature* 405:823-826.

Enright, A.J., Iliopoulos, I., Kyrpides, N.C., and Ouzounis, C.A. 1999. Protein interaction maps for complete genomes based on gene fusion events. *Nature* 402:86-90.

Fryxell, K.J. 1996. The coevolution of gene family trees. *Trends Genet* 12:364-369.

Goh, C.-S., Bogan, A.A., Joachimiak, M., Walther, D., and Cohen, F.E. 2000. Co-evolution of proteins with their interaction partners. *J. Mol. Biol.* 299:283-293.

Gomez, S.M. and Rzhetsky, A. 2002. Towards the prediction of complete protein—protein interaction networks. *Pac. Symp. Biocomput.* 2002:413-424.

Gomez, S.M., Lo, S.H., and Rzhetsky, A. 2001. Probabilistic prediction of unknown metabolic and signal-transduction networks. *Genetics* 159:1291-1298.

Gomez, S.M., Noble, W.S., and Rzhetsky, A. 2003. Learning to predict protein-protein interactions from protein sequences. *Bioinformatics* 19:1875-1881.

Hallas, C., Pekarsky, Y., Itoyama, T., Varnum, J., Bichi, R., Rothstein, J.L., and Croce, C.M. 1999. Genomic analysis of human and mouse TCL1 loci reveals a complex of tightly clustered genes. *Proc. Natl. Acad. Sci. U.S.A.* 96:14418-14423.

Ito, T., Tashiro, K., Muta, S., Ozawa, R., Chiba, T., Nishizawa, M., Yamamoto, K., Kuhara, S., and Sakaki, Y. 2000. Toward a protein-protein interaction map of the budding yeast: A comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proc. Natl. Acad. Sci. U.S.A.* 97:1143-1147.

Jansen, R., Yu, H., Greenbaum, D., Kluger, Y., Krogan, N.J., Chung, S., Emili, A., Snyder, M., Greenblatt, J.F., and Gerstein, M. 2003. A Bayesian networks approach for predicting protein-protein interactions from genomic data. *Science* 302:449-453.

Jeong, H., Tombor, B., Albert, R., Oltvai, Z.N., and Barabasi, A.L. 2000. The large-scale organization of metabolic networks. *Nature* 407:651-654.

Jeong, H., Mason, S.P., Barabasi, A.L., and Oltvai, Z.N. 2001. Lethality and centrality in protein networks. *Nature* 411:41-42.

Jothi, R., Kann, M.G., and Przytycka, T.M. 2005. Predicting protein-protein interaction by searching evolutionary tree automorphism space. *Bioinformatics* 21:i241-i250.

Jothi, R., Cherukuri, P.F., Tasneem, A., and Przytycka, T.M. 2006. Co-evolutionary analysis of domains in interacting proteins reveals insights into domain-domain interactions mediating protein-protein interactions. *J. Mol. Biol.* 362:861-875.

Lawrence, J.G. 2002. Shared strategies in gene organization among prokaryotes and eukaryotes. *Cell* 110:407-413.

Lin, N., Wu, B., Jansen, R., Gerstein, M., and Zhao, H. 2005. Information assessment on predicting protein-protein interactions. *BMC Bioinformatics* 5:154.

Lu, L.G., Xia, Y., Paccanaro, A., Yu, H., and Gerstein, M. 2005. Assessing the limits of genomic data integration for predicting protein networks. *Genome Res*. 15:945-953.

Marcotte, E.M., Pellegrini, M., Ng, H.L., Rice, D.W., Yeates, T.O., and Eisenberg, D. 1999. Detecting protein function and protein-protein interactions from genome sequences. *Science* 285:751-753.

Moyle, W.R., Campbell, R.K., Myers, R.V., Bernard, M.P., Han, Y., and Wang, X. 1994. Co-evolution of ligand-receptor pairs. *Nature* 368:251-255.

Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G.D., and Maltsev, N. 1999. The use of gene clusters to infer functional coupling. *Proc. Natl. Acad. Sci. U.S.A.* 96:2896-2901.

Pazos, F. and Valencia, A. 2001. Similarity of phylogenetic trees as an indicator of protein-protein interaction. *Protein Eng*. 14:609-614.

Pazos, F., Ranea, J.A., Juan, D., and Sternberg, M.J. 2005. Assessing protein co-evolution in the context of the tree of life assists in the prediction of the interactome. *J. Mol. Biol.* 352:1002-1015.

Pellegrini, M., Marcotte, E.M., Thompson, M.J., Eisenberg, D., and Yeates, T.O. 1999. Assigning protein functions by comparative genome analysis: Protein phylogenetic profiles. *Proc. Natl. Acad. Sci. U.S.A.* 96:4285-4288.

Ramani, A.K. and Marcotte, E.M. 2003. Exploiting the co-evolution of interacting proteins to discover interaction specificity. *J. Mol. Biol.* 327:273-284.

Rhodes, D.R., Tomlins, S.A., Varambally, S., Mahavisno, V., Barrette, T., Kalyana-Sundaram, S., Ghosh, D., Pandey, A., and Chinnaiyan, A.M. 2005. Probabilistic model of the human protein-protein interaction network. *Nat. Biotechnol.* 23:951-959.

Riley, R., Lee, C., Sabatti, C., and Eisenberg, D. 2005. Inferring protein domain interactions from databases of interacting proteins. *Genome Biol*. 6:R89.

Sato, T., Yamanishi, Y., Kanehisa, M., and Toh, H. 2005. The inference of protein-protein interactions by co-evolutionary analysis is improved by excluding the information about the phylogenetic relationships. *Bioinformatics* 21:3482-3489.

Scott, M.S. and Barton, G.J. 2007. Probabilistic prediction and ranking of human protein-protein interactions. *Bioinformatics* 8:239.

Sprinzak, E. and Margalit, H. 2001. Correlated sequence-signatures as markers of protein-protein interaction. *J. Mol. Biol.* 311:681-692.

Sprinzak, E., Sattath, S., and Margalit, H. 2003. How reliable are experimental protein-protein interaction data? *J. Mol. Biol.* 327:919-923.

Uetz, P., Giot, L., Cagney, G., Mansfield, T.A., Judson, R.S., Knight, J.R., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., Qureshi-Emili, A., Li, Y., Godwin, B., Conover, D., Kalbfleisch, T., Vijayadamodar, G., Yang, M., Johnston, M., Fields, S., and Rothberg, J.M. 2000. A comprehensive analysis of protein-protein interactions in *Saccharomyces cerevisiae. Nature* 403:623-627.

von Mering, C., Krause, R., Snel, B., Cornell, M., Oliver, S.G., Fields, S., and Bork, P. 2002. Comparative assessment of large-scale data sets of protein-protein interactions. *Nature* 417:399-403.

Witten, I.H. and Frank, E. 2000. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann, San Francisco, Calif.

Wojcik, J. and Schachter, V. 2001. Protein-protein interaction map inference using interacting domain profile pairs. *Bioinformatics* 17:S296-S305.

Wu, Q. and Maniatis, T. 1999. A striking organization of a large family of human neural cadherin-like cell adhesion genes. *Cell* 97:779-790.

Xenarios, I., Rice, D.W., Salwinski, L., Baron, M.K., Marcotte, E.M., and Eisenberg, D. 2000. DIP: The database of interacting proteins. *Nucleic Acids Res.* 28:289-291.

Yanai, I., Derti, A., and DeLisi, C. 2001. Genes linked by fusion events are generally of the same functional category: A systematic analysis of 30 microbial genomes. *Proc. Natl. Acad. Sci. U.S.A.* 98:7940-7945.

Yeang, C.-H. and Haussler, D. 2007. Detecting coevolution in and among protein domains. *PLoS Comput. Biol.* 3:e211.

Zhang, L.V., Wong, S.L., King, O.D., and Roth, F.P. 2004. Predicting co-complexed protein pairs using genomic and proteomic data integration. *BMC Bioinformatics.* 5:38.

INTERNET RESOURCES

http://dip.doe-mbi.ucla.edu

The Database of Interacting Proteins (DIP). A database of both manually and automatically curated experimental protein-protein interactions.

http://string.embl.de

STRING is a database of known and predicted protein-protein interactions. The interactions include direct (physical) and indirect (functional) associations taken from high-throughput experiments, genomic context, coexpression, and literature.

http://www.bind.ca

The Biomolecular Interaction Network Database (BIND). Database of interactions, molecular complexes, and pathways. Includes interactions other than protein-protein (e.g., protein-DNA).

http://cbm.bio.uniroma2.it/mint

The Molecular Interactions Database (MINT). A manually curated database designed to store functional interactions between biological molecules (i.e., proteins RNA and DNA).

http://portal.curagen.com/extpc/com.curagen.portal.servlet.Yeast

PathCalling Yeast Interaction Database. Database of results from Uetz et al. (2000).

http://wit.mcs.anl.gov/WIT2

The WIT homepage. A Web site of reconstructed metabolic pathways for a number of genomes.

http://mips.gsf.de

The Munich Information Center for Protein Sequences (MIPS) homepage. Maintains curated database designed to store functional interactions between biological molecules (e.g., proteins, RNA, DNA).

http://www.genome.ad.jp/kegg

KEGG: Kyoto Encyclopedia of Genes and Genomes. In addition to other material, this site provides a database of molecular interactions as well as metabolic and signal transduction pathways.

http://www.ecocyc.org

The Encyclopedia of Escherichia coli Genes and Metabolism (EcoCyc) Web site.

http://pim.hybrigenics.com

Web site for Hybrigenics' Protein Interaction Map (PIM) functional proteomics software platform.

Evaluation of Electrostatic Interactions

Of the many interactions made between associating molecules, electrostatic interactions are particularly interesting for several reasons. It is generally accepted that the driving force for most macromolecular association events is the hydrophobic effect, the entropic benefit of releasing solvent from the binding surfaces of each molecule. This effect is nonspecific, however, with any burial of the same surface area contributing equally (Chothia, 1974; Chothia and Janin, 1975; Sharp et al., 1991). Van der Waals interactions are also relatively nonspecific, with only substantial steric clashes resulting in large unfavorable energies, and individual favorable interactions being relatively small in magnitude. On the other hand, electrostatic interactions are highly specific; electrostatic interaction energies can range from highly favorable to highly unfavorable depending on the identity and geometry of the interacting groups. Furthermore, electrostatic interactions act over a significantly longer range (the energy of interaction between two charged groups falls off linearly with distance and the interaction of two dipoles decreases with the cube of the distance) than do van der Waals interactions, which decrease with the sixth power of the distance between the interacting groups. In addition, solvation effects can make the energetics of electrostatic interactions nonintuitive; groups making favorable interactions in the bound state of a complex may make even more favorable interactions with solvent in the unbound state, causing the net contribution to binding to become unfavorable (Hendsch and Tidor, 1994, 1999). Thus, while it is relatively clear that the most favorable van der Waals interactions are made by making the maximal contact between groups without steric interference, and that the hydrophobic effect favors the burial (and conversely disfavors the solvent exposure) of nonpolar groups, in order to understand electrostatic interactions it is necessary to consider in detail both the bound and unbound states.

Described here are several computational procedures for the analysis of electrostatic interactions in molecular complexes, all based on a continuum model of solvation. In particular, three methods will be described, each of increasing sophistication and requiring correspondingly larger computational resources. The first section describes how to compute the residual potential, a measure of how electrostatically complementary a ligand is for its receptor (see Basic Protocol 1). Residual potential is particularly useful as a visual measure, but the degree of complementarity can also be quantified. The second procedure describes electrostatic component analysis, a method by which the electrostatic contribution to the binding free energy (ΔG_{Bind}) can be broken up into terms directly attributable to individual chemical groups (see Basic Protocol 2). In this way, contributions to binding can be computed for individual residues or of any group of residues, giving a highly detailed description of interaction electrostatics. Finally, electrostatic affinity optimization is described (see Basic Protocol 3). This methodology allows for the computation of the set of partial atomic charges on a ligand that leads to the best electrostatic binding free energy. This procedure is particularly useful in determining what portions of a ligand are the most suboptimal, and thus provide the greatest opportunity for the design of improvements. Furthermore, analysis of the optimal charge distribution itself can suggest types of chemical modifications most likely to lead to enhanced affinity. All these procedures are currently being integrated into a suite of software.

NOTE: All the methods described here are based on a continuum model of solvation, described by the linearized Poisson-Boltzmann equation (Warwicker and Watson, 1982; Gilson and Honig, 1987; Gilson et al., 1988; Mohan et al., 1992). Thus, all the assumptions implicit in this choice of model are contained within these protocols. In most cases, a rigid-body docking model is also assumed, although this approximation is strictly required only for the analysis of residual potential (see Basic Protocol 1).

Contributed by David F. Green and Bruce Tidor

Current Protocols in Bioinformatics (2003) 8.3.1-8.3.16

ANALYSIS OF ELECTROSTATIC COMPLEMENTARITY

The electrostatic contribution to binding involves the counterplay of interactions between the binding partners in the bound state with interactions between each molecule and the solvent in the unbound state. As a result, the degree to which a ligand is electrostatically complementary to its target receptor can be described by how well balanced these contributions are. This balance can be plotted onto the surface of a ligand in a manner analogous to the plots of surface potentials frequently used. Rather than plotting the overall electrostatic potential, however, a new potential is defined as the sum of the ligand desolvation potential (the difference of the electrostatic potential of the ligand in the bound and unbound states) and the receptor interaction potential (the electrostatic potential of the receptor in the bound state, defined within and on the ligand surface). If a ligand is the perfect electrostatic complement of a target receptor, the residual potential will be zero everywhere within the ligand. A nonzero residual potential indicates a region of suboptimal electrostatic interaction.

Necessary Resources

Hardware

Silicon Graphics (SGI) computer running the IRIX operating system.

Software

Scripts and GRASP macros for computing and processing electrostatic potentials (*http://web.mit.edu/tidor/www/residual*)

GRASP (Nicholls et al., 1991; *http://trantor.bioc.columbia.edu/grasp*) or equivalent software capable of displaying electrostatic potentials on a molecular surface

Files

PDB-format coordinate file with all chains of the ligand labeled `A` and all chains of the receptor labeled `B`

DelPhi-format charge and atomic radius files (Fig. 8.3.1A and 8.3.1B) for the complex of interest

Several such files are included with GRASP.

1. Since the residual potential scripts are restricted to complexes consisting of exactly two individual chains named A and B, if the ligand or receptor consists of multiple segments or different chain identifiers, rename the chain identifiers in the PDB file to obtain the desired results.

   While tedious, this allows most of the rest of the process to be automated.

2. Rename the PDB, charge, and radius files `complex.pdb`, `complex.crg`, and `complex.siz`, respectively.

Set up GRASP

3. Run GRASP. Right click on the main window and select Read from the menu. Select Grasp Macro File and type in the name of the macro file from the residual potential distribution (i.e., `residual.macros`).

   The macros are now loaded.

4. Prepare GRASP by right clicking the main window, selecting Macros from the menu, and then selecting Residual Setup.

```
A
123456789012345678901 2
aaaaaarrrnnnncqqqqqqqq
N     GLU 2  A-0.40000
H     GLU 2  A 0.40000

B
12345678901234567
aaaaaannnrrrrrrrr
N     ALA 1.5000
H     ALA 1.0000
```

Figure 8.3.1 (**A**) Example of a charge file for the complex of interest in DelPhi format. Each line of this file contains one charge entry. The first column is the atom name, the second the residue name, the third the residue ID, the fourth column the chain ID, and the fifth column the partial atomic charge. Two example lines are shown for a backbone NH group, with the nitrogen having a partial charge of `-0.4` and the hydrogen `+0.4`. (**B**) Example of a radius file for the complex of interest. Each line of this file contains one radius entry. The first column is the atom name, the second is the residue name, and the third is the atomic radius. Again, two example lines are shown, the nitrogen having a radius of 1.5 Å, the hydrogen 1.0 Å.

This will eliminate the cross hairs, set the inner dielectric constant to 4.0, the bulk salt concentration to 0.145 M, and load the PDB, charge, and radius files.

The command sequence following this step will compute the residual potential treating chain A as the ligand and chain B as the receptor. To reverse the roles of A and B (defining B as the ligand and A as the receptor), simply swap A and B throughout the instructions.

Computing the residual potential

5. Create and display the molecular surface of molecule A by right clicking on the main window and selecting Macros followed by Generate Surface of A.
6. Reload all partial atomic charges by right clicking the main window and selecting Macros followed by Read Charges.

   This is necessary to prepare for the next step.
7. Compute the unbound potential of the ligand and display it on the surface by right clicking on the main window and selecting Macros followed by Generate/Map/Store 1 Unbound A Potential.

   This also saves the unbound potential of the ligand as General Property 1.
8. Compute the bound potential of the ligand and display it on the surface by right clicking on the main window and selecting Macros, then Generate/Map/Store 2 Bound A Potential.

   This also saves the bound potential as General Property 2.
9. Subtract the unbound from the bound potential, which will yield the desolvation potential, and save it in General Property 1, by right clicking the main window and selecting Macros followed by Difference/Store1 Ligand Desolvation Potential.

   This potential is not displayed, only stored in General Property 1.
10. Reload all partial atomic charges by right clicking the main window and selecting Macros followed by Read Charges.

    This is necessary to prepare for the next step.

11. Compute the bound potential of the receptor on the ligand's surface, save it in General Property 2, and display it by right clicking the main window and selecting Macros then Generate/Map/Store2 B.

12. Determine the sum of the interaction and desolvation potentials (i.e., residual potential), by right clicking the main window and selecting Macros followed by Compute Residual Potential.

Analyzing the residual potential

13. Use the following procedures to view the residual potential and its two components, the ligand desolvation potential and the receptor interaction potential, all of which have been computed and stored in the variables potential, property 1, and property 2, respectively.

    a. *To view the ligand desolvation potential (property 1):* Right click the main window and select Macros followed by Display Desolvation Potential.

    b. *To view the receptor interaction potential (property 2):* Right click the main window and select Macros followed by Display Interaction Potential.

    c. *To view the residual potential (potential):* Right click the main window and select Macros followed by Display Residual Potential.

14. For each potential, right click on the white window left of the horizontal color bar, select Input Relative Values, and enter the low, middle, and high potential values for the scale (e.g., `-60,0,30`).

    Note that each potential is mapped to a different scale as shown on the horizontal color bar on the screen. It is usually best to plot them on the same scale, or at least ones which are comparable, which is the purpose behind this step.

    Note, that zero should be used for the middle value to separate colors effectively. Also, because the interaction and desolvation potentials should be equal in magnitude and opposite in sign, if one scale is (`-60,0,30`*), then the other should be (*`-30,0,60`*).*

15. If desired, use the mouse to rotate the surface so that the active site is clearly visible.

    The SGI snapshot utility is the simplest way to capture the image to a file. Invoke snapshot with the `snapshot` *command. Resize the capture window with the left mouse button, and capture the image with the right mouse button.*

16. If desired, set the background to white, gray, or black, and hide the stick drawing of the receptor using the following procedures.

    a. *To set the molecule background white or gray:* Right click the window displaying the molecule and select Macros followed by Background White.

    b. *To set the molecule background black:* Right click the window displaying the molecule and select Macros followed by Background Black.

    c. *To hide the stick drawing of the complex:* Right click on the window displaying the molecule and select Display followed by Hide and then Bonds.

    Hiding the stick drawing will leave only the molecular surface visible.

    An example of the residual potential computed for two related ligands is shown in Figure 8.3.2. The protein ligand on the right-hand side differs from that on the left by the mutation of three residues to lysine. These mutations lead to increased electrostatic complementarity as revealed by the reduction in the (red) residual potential in moving from the wild-type (left panel) to the mutant (right panel).

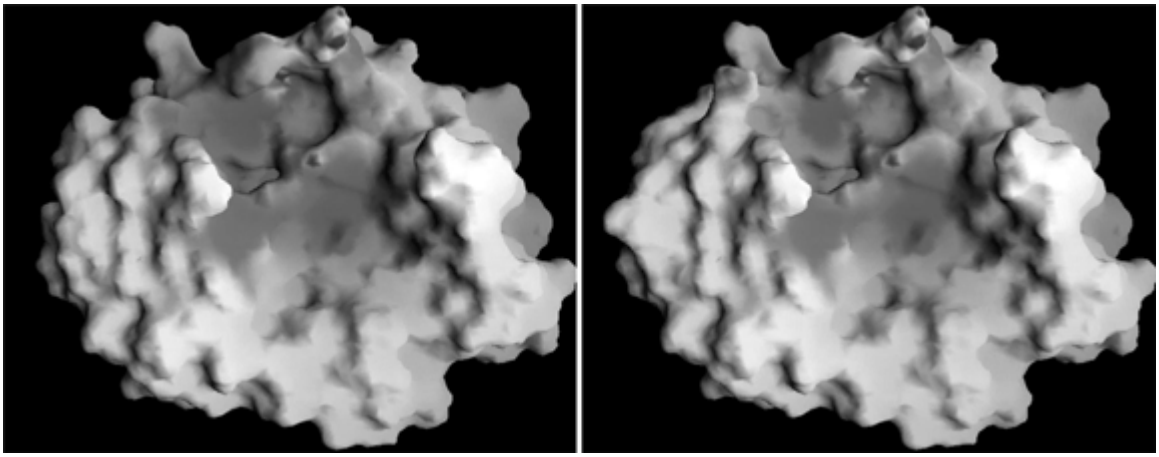


Figure 8.3.2 Increased electrostatic complementarity is indicated by a smaller magnitude residual potential. The large negative residual potential on the left-hand side is reduced in the right-hand figure. The ligand on the right has a several additional positively charged residues that interact with negative groups on the receptor. *This black and white facsimile of the figure is intended only as a placeholder; for full-color version of figure go to http://currentprotocols.com/colorfigures.*

ELECTROSTATIC COMPONENT ANALYSIS

BASIC PROTOCOL 2

One of the greatest strengths of the linearized Poisson-Boltzmann model for analyzing electrostatic interactions lies in the separability of the energetic contributions of various groups of atoms. Described here is a technique for carrying out calculations to break down the electrostatic binding free energy into contributions from each user-defined group in a system. Most typically, proteins are split into three groups for each residue (i.e., side chain, backbone amino, and backbone carbonyl). Nucleic acids are similarly split into three groups for each nucleotide (i.e., base, ribose, and phosphate) and small molecules are partitioned in a user-defined manner. For each of these groups, numerous energetic terms are calculated. These include the desolvation energy of the individual group, the solvent screened interactions between the group and each group on the binding partner in the bound state (intermolecular interactions), and the difference in solvent screening of the interactions between the group and other groups on the same molecule in the bound and unbound states (intramolecular interactions). These are termed the desolvation, direct interactions, and indirect interactions, respectively, and their sum gives the total electrostatic binding free energy.

Necessary resources

To perform this procedure, the user will need DelPhi (*UNIT 8.4*; Gilson and Honig, 1987; Gilson et al., 1988; Sharp and Honig, 1990; *http://trantor.bioc.columbia.edu/delphi*) or another Poisson-Boltzmann solver, scripts for setting up and processing the required Poisson-Boltzmann calculations, scripts for analyzing data, and a workstation with a Unix-like operating system to run them (scripts for distribution are currently in preparation by the authors). In addition, a PDB-format coordinate file and DelPhi-formatted charge and radius files (Fig. 8.3.1) are required.

Define the system

Before any calculations can be performed, the system must be defined. Typically, the bound complex is rigidly separated into the relevant isolated (unbound) states; multiple molecules may be considered to be associated in a single unbound state if this is the biologically functional state. Structurally important water molecules are generally considered as being associated with one of the molecules in the unbound state, but may be

considered as separate binding components, isolated in the unbound state, if this seems more appropriate.

Electrostatic contributions to stability can be considered as well, defining a native state as above, and defining some choice of a model for the unfolded state. The simplest model of the unfolded state for amino acid side chains is the side chain free in solution; however, other models, including the side chain in the context of a region of polypeptide backbone, or some model of the full sequence in a nonnative state, may also be considered. Additional parameters, such as variations in the parameters used in the continuum electrostatic calculations, can be specified according to specifications of the software used.

Set up and run continuum electrostatic calculations

The next step involves setting up and executing all necessary continuum electrostatic calculations, and can take a substantial length of time and significant computational resources. For each group in the system, two continuum electrostatic calculations must be performed, namely the potential produced by the charges of the group alone in the context of the bound and the unbound shapes must both be computed. From these potentials, a number of energetic contributions are derived. The desolvation energy of the group is computed from the difference of the self-energies of the bound and unbound states. All pairwise intramolecular interactions between groups (indirect interactions) are computed from the difference between the bound and unbound potentials at the atoms of each group. Finally, all pairwise intermolecular interactions between groups (i.e., direct interaction energies involving the group are computed from the bound state potentials at the atoms of each group) are computed. Due to the reciprocity implicit in the continuum electrostatic model, the interaction energy computed using the potentials generated by one half of an interacting pair is equal to that computed using the potentials generated from the other half. With a relatively coarse grid, optimized for speed, each of these calculations takes roughly 15 min/group on a typical workstation; however, as a typical complex may have hundreds of groups, the calculation may take several days if not split over multiple processors.

Analyzing results

A component analysis provides a huge volume of data, a desolvation energy for each group, and all pairwise intergroup interaction energies, both intra- and intermolecular. Thus, it is useful to define several terms to simplify the analysis. The mutation energy of a group is defined as the sum of the desolvation penalty paid by the group and all of its intra- and intermolecular interactions. This corresponds to the relative energies of the natural system and a hypothetical mutant in which the group of interest, and that group alone, is replaced with a hydrophobic isostere. In other words, the mutation energy is the energetic contribution from "turning on" the charges of the group in the context of the natural system. Since the mutation energy fully counts all the interactions of a group ("turning off" the charges on a group eliminates all interactions) mutation energies cannot be added together without double counting some interactions. To provide an energetic term for each group, which will be summed to give the total electrostatic energy, the contribution energy of a group is defined as the sum of the group desolvation energy and one half of all intra- and intermolecular interactions. The contribution energy does not correspond to any thermodynamic cycle, but is a useful measure for partitioning the electrostatic interaction among various groups. In Table 8.3.1 the top ten groups for binding in a simple bimolecular system are shown, sorted by mutation energy. In this case, several highly favorable groups are pinpointed (dominated by large favorable direct interactions), but Asp163 on chain B and Glu213 on chain A are computed to contribute unfavorably to binding by over 2.5 kcal/mol. Considering only the total interactions made by a group may be useful in some cases, but also neglects the detailed nature of the

Table 8.3.1 Components Ranked by Mutation Energy for a Simple Bimolecular Binding System[a,b]

Component[c]	Desolv.	Inter. A	Inter. B	Contrib.	Mut.
LYS 74 B	6.8	−13.2	−7.9	−3.7	−14.3
ASP 49 B	12.8	−25.4	1.4	0.8	−11.1
LYS 208 A	1.9	1.5	−11.1	−2.9	−7.7
LYS 48 A	2.2	−3.4	−4.3	−1.7	−5.5
ARG 217 A	2.0	1.2	−8.7	−1.7	−5.4
ASP 163 B	1.7	3.2	−0.4	3.2	4.6
ARG 160 B	2.4	−5.9	0.4	−0.4	−3.1
LYS 86 A	2.0	−2.2	−2.9	−0.6	−3.1
SER 209 A	0.9	1.5	−5.3	−1.0	−2.9
GLU 213 A	3.2	−0.0	−0.4	3.0	2.7

[a]Abbreviations: Contrib., contribution energy; desolv., desolvation energy; inter. A, interaction energy with chain A; inter. B, interaction energy with chain B; mut., mutational energy.

[b]The system described is TEM1 β-lactamase (A) binding to its inhibitor BLIP (B).

[c]A component is the group of side chain atoms of the specified residue.

Table 8.3.2 Component Analysis of Lys 74B[a]

Parameter	Value (kcal/mol)
Contribution	−3.7
Desolvation	6.8
Direct (A)	−13.2
Mutation	−14.3
Indirect (B)	−7.9

[a]See Table 8.3.1 for other important residues and Table 8.3.3 for residues which interact with Lys 74B, and are thus are components of the contribution.

Table 8.3.3 Individual Interactions of Lys 74B with Other Residues[a]

Interacting residue	Contribution (kcal/mol)
TYR 143 B	−1.0
GLU 73 B	−3.8
LYS 48 A	1.7
GLU 79 A	−14.4
ASP 106 A	−1.3
GLU 141 A	−1.7
Carbonyl 141 B	−2.3

[a]See Table 8.3.1 for other important residues and Table 8.3.2 for a component analysis of Lys 74B.

energetic description provided by a component analysis. Tables 8.3.2 and 8.3.3 show all the individual interactions >1.0 kcal/mol made by Lys 74B, which makes a highly favorable contribution to binding. Note that the large favorable direct interaction is dominated by a single interaction with Glu79.

BASIC PROTOCOL 3

ELECTROSTATIC AFFINITY OPTIMIZATION

Breaking down the electrostatic binding free energy further and considering every atom in the system as its own group leads to a particularly interesting result. Due to the linear response of the linearized Poisson-Boltzmann model, the potential produced by any single charge is directly related to the potential produced by a unit charge at the same position, with this potential simply scaled by the value of the charge. This leads to an expression for the binding free energy where the effects of the charges on the ligand and receptor are separated from the effects of the geometry of binding. This allows the charge distributions of the molecules to be varied without additional computation and provides a framework in which to compute an optimal charge distribution for a ligand (i.e., the set of charges on a ligand which provides the best electrostatic binding free energy to a target receptor relative to any other ligand charge distribution).

Necessary resources

To perform this procedure, the user will need DelPhi (*UNIT 8.4*; Gilson and Honig, 1987; Gilson et al., 1988; Sharp and Honig, 1990; *http://trantor.bioc.columbia.edu/delphi*) or another Poisson-Boltzmann solver, scripts for setting up and processing the required Poisson-Boltzmann calculations, scripts for analyzing data, and a workstation with a Unix-like operating system to run them (scripts for distribution are currently in preparation by the authors). In addition, a PDB-format coordinate file and DelPhi-formatted charge and radius files (Fig. 8.3.1) are required.

Define the system

Before calculations can be performed, the system must be defined. As with component analysis (see Basic Protocol 2), the bound complex is rigidly separated into the relevant isolated (unbound) states, with structurally important water molecules partitioned appropriately. All the considerations for setting up a system for component analysis are equally applicable to charge optimization. In addition, the user must specify the subset of ligand atoms to be optimized. Unless substantial computational resources are available, only a select set of atoms should be considered in the initial analysis, since a pair of continuum electrostatic calculations must be performed for every optimized ligand atom. Once again, specification of variations in the parameters used in the continuum electrostatic calculations can be implemented according to specifications of the software used.

Set up and run continuum electrostatic calculations

The next step involves preparing and executing all necessary continuum electrostatic calculations, and can take a substantial length of time and significant computational resources. For each atom included for optimization, two continuum electrostatic calculations must be performed, namely computing the potential produced by a single unit charge on the atom of interest in the context of the bound and the unbound shapes. This provides all of the elements for the ligand desolvation matrix: the diagonal elements corresponding to the desolvation potential of each atom, and the off-diagonal elements corresponding to intramolecular interaction potentials between ligand atoms, both taken as the difference of the bound and unbound state potentials. This matrix can be used to obtain all the interaction elements between optimized and fixed atoms as well. The interaction vector consists of an element for each ligand atom, corresponding to the sum of the interaction potentials from all nonoptimized atoms. In addition, calculations on both the receptor and

the nonoptimized atoms of the ligand in the bound and unbound states must be performed. These calculations are required in order to reconstitute the full electrostatic binding free energy and yield the desolvation energy of all fixed atoms. In addition, they can be used to give interaction potentials between the fixed and optimized atoms. Using a coarse grid for the continuum electrostatic calculations, optimized for speed, each of these calculations takes roughly 15 min on a typical workstation. Thus the calculations for a small molecule ligand or a single amino acid may be completed between several hours and a day, while for larger systems multiple processors are required if the calculations are to be completed in any reasonable length of time.

Analyzing results

The optimization can be done several ways, depending of the nature of the optimization problem. In the simplest cases, a direct solution of the optimal charge distribution ($\vec{Q}^{\text{opt}}_{\text{l}}$) can be obtained from the ligand desolvation matrix **L** and the interaction vector ($\vec{C}_{\vec{Q}\text{r}}$). The optimal charge distribution is given by:

$$\vec{Q}^{\text{opt}}_{\text{l}} = -\frac{1}{2}\mathbf{L}^{-1}\vec{C}_{\vec{Q}_{\text{r}}}$$

and the optimal binding free energy (ΔG^{opt}) by:

$$\Delta G^{\text{opt}} = -\frac{1}{4}\vec{C}^{\dagger}_{\vec{Q}_{\text{r}}}\mathbf{L}^{-1}\vec{C}_{\vec{Q}_{\text{r}}}$$

where the dagger denotes the conjugate transpose of the vector.

The inversion of the **L** matrix can be done by a variety of methods, but typically singular value decomposition (SVD) is used due to its ability to appropriately deal with problems resulting from imprecision in the numerical methods. While this approach is useful for some cases, most notably for analysis of tightly bound small molecules, in other cases the optimal charge distributions obtained by the direct calculation are nonphysical. This is a result of certain charges that pay a very small desolvation penalty upon binding, and thus can take on extremely large charges to maximize the (often equally small) interactions made by the atom. In many cases, these charges can be constrained to more chemically realistic values with little energetic cost. In these situations, it is best to optimize using software designed for constrained optimization, minimizing the objective function:

$$\Delta G^{\text{var}} = \vec{Q}^{\dagger}_{l}\mathbf{L}\vec{Q}_{l} + \vec{C}^{\dagger}_{\vec{Q}_{\text{r}}}\vec{Q}_{l}$$

which is the portion of the electrostatic binding free energy dependent on the ligand charges ($\vec{Q}_{\text{l}}$).

Shown in Table 8.3.4 is an example of the energetic results of the optimization of the side-chain charges of the lysine previously identified in the component analysis (Tables 8.3.1 to 8.3.3). The optimal charge distribution has a binding free energy ~2.5 kcal/mol better than the wild-type lysine (i.e., $\Delta G_{\text{Bind}} - \Delta G_{\text{WT}} = -2.5$), but both the optimal and wild-type charges contribute 15.0 kcal/mol or more of favorable binding free energy relative to a hydrophobic isostere (e.g., $\Delta G_{\text{Bind}} - \Delta G_{\text{Ref}} = -17.2$). The properties of the optimal charge distribution are shown in Tables 8.3.5 and 8.3.6. While the optimal charges are somewhat different than those of a lysine, the net charge is $+1e$, the dipole moment of the optimal charges resemble that of lysine, and the optimal charges on the terminal NH_3 looks similar to an ammonium group. Note this favorably contributing wild-type residue is close to optimal.

Table 8.3.4 Energetics of Charge Optimization of an Amino Acid Side Chain

	Optimum	Wild type	Reference
ΔG_{total}	31.2 ± 1.1[a]	33.6 ± 1.1[b]	48.4 ± 1.1[c]
Ligand desolvation	54.4 ± 0.2	49.1 ± 0.1	50.3 ± 0.1
Interaction	–71.2 ± 1.1	–63.4 ± 1.1	49.8 ± 1.1
Receptor desolvation	48.0 ± 0.1	48.0 ± 0.1	48.0 ± 0.1

[a]Referred to as ΔG_{Bind}.

[b]Referred to as ΔG_{WT}.

[c]Referred to as ΔG_{Ref}.

Table 8.3.5 Optimal Charge Distribution of Lys 74B of BLIP Binding to TEM1 β-Lactamase[a,b]

Atom	Q	Atom	Q	Atom	Q	Atom	Q
C_β	–0.85	C_γ	0.20	C_δ	0.75	C_ε	–0.05
NZ	–0.75	HZ1	0.85	HZ2	–0.01	HZ3	0.85

[a]Also see Tables 8.3.1 to 8.3.4.

[b]For this residue, the total charge on the optimum, wild type, and hydrophobic reference are +1.00, +1.00, and 0.00, respectively.

Table 8.3.6 Dipole Moment for Lys 74B of BLIP Binding to TEM1 β-Lactamase[a]

Optimal Charges			Wild-Type			Reference		
P_x	P_y	P_z	P_x	P_y	P_z	P_x	P_y	P_z
0.37	–1.11	3.13	0.15	–0.21	1.17	0.00	0.00	0.00

[a]Also see Tables 8.3.1 to 8.3.5.

GUIDELINES FOR UNDERSTANDING RESULTS

Analyzing the Residual Potential

The simplest way to analyze the residual potential (see Basic Protocol 1) is to simply look at it. Regions of high complementarity are indicated by a small residual potential (white), while regions of noncomplementarity are indicated by a high residual potential (red if negative, blue if positive). Noncomplementarity can arise from three possible situations: a ligand (or a region of a ligand) may be anticomplementary, undercharged, or overcharged. A region of anticomplementarity is one in which the desolvation potential of the ligand (in some sense a measure of local net charge) is the same sign as the receptor interaction potential—essentially a region where like-charges are interacting—and thus clearly a region of noncomplementarity. The other two situations are somewhat less obvious. If one considers a receptor with a positively charged residue poised for interaction, there are several ways a ligand may interact with the residue. Interactions can be made with polar neutral residues or with negatively charged residues on the ligand, and a number of either type of interaction is possible. If not enough negative (or partially negative) groups are appropriately oriented, the ligand may be undercharged (i.e., the receptor interaction potential is greater in magnitude than the ligand desolvation potential) and increasing the negative character of that region of the ligand should promote tighter binding. On the other hand, too many negative groups interacting with a fixed positive charge leads to an overcharged ligand (i.e., the receptor interaction potential is smaller in

magnitude than the ligand desolvation potential) and binding should be enhanced by reducing the overall negative charge of the region of interest. Which case is true in any given situation can be determined by careful consideration of the potentials, as well as analysis of the interacting groups on both the ligand and receptor. An example of an undercharged ligand is shown in Figure 8.3.2A. Adding additional positive charge onto the ligand increases the complementarity (Fig. 8.3.2B). One key point about the residual potential is that it is fundamentally asymmetric, describing the binding of one component deemed the ligand to another component considered the receptor. In a system where one half of a complex (the ligand) is perfectly complementary for binding the other (the receptor), the reverse is generally not true, namely the receptor is not perfectly complementary for binding to the ligand.

Component Analysis

The results of a component analysis (see Basic Protocol 2) are relatively straightforward to interpret: a favorable (negative) mutation term indicates a group whose conversion to a hydrophobic isostere would lead to reduced binding affinity, while a positive mutation term indicates a group whose mutation to a hydrophobic replacement would improve binding. These results, however, apply only to the *binding* free energy; a group which has an unfavorable contribution to binding may be important for the stability of the bound conformation (or of the native state of a protein) and thus it is important to consider what intramolecular interactions a group is making in choosing a target for mutation. Another important consideration is the nonadditivity of the mutation free energy. Just as the results of an experimental alanine scan are nonadditive, so are the mutation terms from a component analysis. In both cases, all the interactions of a group are eliminated upon the mutation, and thus interactions between two groups would be counted twice if the energies were simply added. To obtain the effect of mutating a pair of residues to hydrophobic replacements, the interaction between the groups must be subtracted from the sum of the individual mutation free energies. Finally, the component analysis only considers electrostatic effects, and thus should not be considered a direct measure of how an experimentally realizable mutation would perform. Rather, the detailed description of the electrostatic interactions obtained through component analysis provides a means of identifying which regions of a binding interface are particularly important for binding (highly favorable), which regions seem to oppose binding (highly unfavorable), and which regions play little electrostatic role in binding. This understanding can clearly be incredibly useful in designing tighter binding complexes, and in some circumstances, simple modifications may be directly suggested by the results. It is important to keep in mind, however, that the computed energies are not directly comparable to an experimental result.

Electrostatic Optimization

The results of electrostatic optimization (see Basic Protocol 3) can be divided into two linked, but distinct pieces. First, there is the energetics of optimization, the improvement in binding free energy upon optimization, both in relation to the natural system and in relation to the hydrophobic isostere, which is the reference state of a component analysis. Secondly, there are the characteristics of the optimal charge distribution (net charge as well as individual partial atomic charges), which can also be compared to those of the natural system. These two properties are clearly linked, but it is important to consider both when analyzing the results of a charge optimization. Some regions of a molecule are electrostatically unimportant for binding, not significantly desolvated on binding, and not poised for interactions in the bound state. Thus, deviations in charge away from the optimum have little effect in these areas. Other areas are highly desolvated upon binding, and even small variations in charge in these regions can have large energetic effects. For design purposes, particularly for large systems, it is best to look first at the energetics of

optimization, and then focus attention on those regions which provide opportunities for significant improvement upon optimization. These are *optimal* electrostatic improvements, and the improvements of any chemical modifications are likely to be smaller in magnitude. Once regions of energetic suboptimality are determined, the analysis of the optimal charge distributions can give a great deal of insight into the binding system. The optimal net charge in a region, and how the energetics of binding change with the net charge fixed at different values, can give general suggestions as to what type of functional groups are preferred at different positions. The details of the optimal partial charges may provide further insight, but when considering the results at such a detailed level, it is important to consider the energetic effects of small deviations, as a close match of optimal and natural charges is only necessary in regions where small variations have large energetic effects. When multiple residues are chosen for optimization, it is best to consider the optimization of each residue individually, only subsequently optimizing the charges on multiple residues simultaneously. This allows for a separation of the effects of each residue individually from the optimal interactions between the two groups, which in many cases can lead to results which are far from chemically reasonable. In all cases, it is best to use the results of charge optimization as a guide, observing regions which are close to optimal, designing modifications to areas of suboptimality, and then modeling the effects of an actual chemical substitution, rather than directly drawing conclusions from the optimization. Close matches between optimal and natural charges (and energies) are relatively straightforward to interpret. Mutations are likely to reduce affinity, but making changes to region of suboptimality may not improve binding, as the mutations must move the ligand closer to optimal to be effective.

COMMENTARY

Background Information

Continuum model of solvation and its applications

Over the past two decades, the continuum model of solvation has been shown to be a powerful tool for the analysis of electrostatic interactions in biological systems. Continuum methods allow the solvation energetics of biological macromolecules in an aqueous, moderate ionic strength environment to be calculated relatively quickly and accurately (Warwicker and Watson, 1982; Gilson and Honig, 1987; Davis and McCammon, 1990; Bashford and Karplus, 1990). In the continuum model, molecules are generally described as a set of partial point charges located at atomic centers embedded in a low dielectric region described by the molecular surface (Gilson et al., 1988; Mohan et al., 1992).

Applications have included analysis of the electrostatic field around biological molecules (Warwicker and Watson, 1982; Gilson et al., 1997), studies on prediction of the pK_a of titratable groups in proteins (Yang et al., 1993; Potter et al., 1994; van Vlijmen et al., 1998), and numerous investigations into the electrostatic contributions to protein stability (Gilson and Honig, 1989; Hendsch and Tidor, 1994, 1996; Xiao and Honig, 1999) and the affinity of macromolecular complexes (Zacharias et al., 1992; Misra et al., 1994; Froloff et al., 1997; Misra et al., 1998; Hendsch and Tidor, 1999; Archontis et al., 2001). Recently, theoretical and methodological advances have made it possible to use continuum electrostatics as a tool in designing more tightly and specifically associating molecular complexes (Lee and Tidor, 1997; Kangas and Tidor, 2001; Sarkar et al., 2002).

The ability to separate the electrostatic free energy into contributions from various groups has been applied to numerous systems, leading to a deeper understanding of the detailed nature of electrostatic interactions. Initial applications studying electrostatic contributions to protein stability revealed the great importance of considering solvation effects. An analysis of several systems showed that buried salt-bridges in protein cores generally contribute little to protein stability, and in many cases contribute unfavorably relative to hydrophobic isosteres due to the large cost of desolvating charged groups upon burial (Hendsch and Tidor, 1994). Studies on binding in protein-protein and protein-DNA complexes further revealed the intricacies of electrostatic interactions. Substantial energetic contributions may arise from *indirect*

interactions, the enhancement of intramolecular electrostatic interactions upon binding as a result of the reduced screening by solvent in the bound state (Hendsch and Tidor, 1999). Analyses of electrostatic contributions to both stability and affinity have been shown to be useful in making experimentally validated predictions. Improving electrostatic interactions can lead to enhanced stability (Hendsch et al., 1996; Spector et al., 2000) and to variation in binding affinity and specificity (Nohaile et al., 2001).

Electrostatic optimization

The success of detailed analysis of electrostatic interactions in biological systems led to the development of the theory behind the electrostatic optimization protocol several years ago (Lee and Tidor, 1997; Kangas and Tidor, 1998, 1999; Kangas and Tidor, 2000), and initial applications on simplified model systems suggested that its application to biological complexes might provide similarly useful insights (Chong et al., 1998). The procedure is founded on the idea of balancing the desolvation penalty paid by the ligand on binding with the interactions the ligand can make with the receptor in the bound state. The desolvation penalty increases proportionally to the charge of the ligand, since solvation energies are due to the interaction of a charge with a reaction field proportional to that charge, while the interaction energy increases linearly with ligand charge, as the receptor charge distribution is fixed. As a result, the overall electrostatic binding free energy of a ligand can be described by a paraboloid in ligand charge space, with the minimum of the paraboloid, the electrostatic optimum, being the point at which the most interactions are made for the smallest desolvation penalty (Fig. 8.3.3). Initial applications of the optimization protocol have borne out its utility. In the barnase-barstar complex, one of the highest affinity protein complexes known, comparison of optimal and natural charges on barstar showed remarkable agreement (i.e., barstar is electrostatically optimized to bind to barnase; Lee and Tidor, 2001a). Application of the charge optimization approach to cation binding sites, both to a potassium selective crown ether and to a calcium binding protein, revealed close agreement between the optimal charges and the charges of the known preferred cations (Sulea and Purisima, 2001). In two enzyme systems, chorismate mutase from *B. subtilis* and glutaminyl-tRNA synthetase from *E. coli*, the optimal and natural charges of small molecule inhibitors similarly show close agreement in many regions. Differences in both of these cases suggest chemical modifications are likely to improve binding (Kangas and Tidor, 2001; Green and Tidor, unpub. observ.). In addition, the optimization procedure has recently been applied to the design of several proposed modifications to a protein inhibitor of HIV-1 cell entry, which are computed to significantly enhance binding (Green and Tidor, unpub. observ.), and to the design of cytokines with enhanced pH-dependent binding, which results in both increased lifetime and potency (Sarkar et al., 2002).

The concept of the residual potential as a measure of the balance of the desolvation paid by a ligand and the interactions it can regain with the receptor in the bound state is a direct outcome of the theory of electrostatic optimization. Just as the optimization procedure computes the charges for which the ligand desolvation penalty and favorable interactions are optimally balanced, the residual potential visually displays this balance (i.e., the residual potential is zero everywhere for the optimal ligand). In the barnase-barstar system, the optimality of barstar can be seen in the residual potentials, and comparison with the residual potential of barnase for binding barstar clearly shows that barstar, the evolved inhibitor, is more complementary to barnase than barnase, the enzyme with additional function, is to barstar (Lee and Tidor, 2001b).

Critical Parameters and Troubleshooting

The primary variable parameters of these calculations are those related to the implementation of the continuum electrostatic calculations. In particular, an internal dielectric constant of 4.0 is suggested for most calculations, although applications in the literature have suggested the use of values ranging from 2.0 to 20.0. The external dielectric constant is typically set to 80.0, with an ionic strength of 0.145 M (typical cellular ionic strength). Parameters determining the size of the grid used in the finite-difference solution of the Poisson-Boltzmann equation can also be varied. Often, a relatively coarse grid of $65 \times 65 \times 65$ Å is used due to the large number of calculations required for most analyses. The optimization procedure involves a matrix inversion, generally carried out using SVD (singular value decomposition). A standard value for the SVD cutoff is 1×10^{-5} of the largest singular value, and typically the null space is excluded from the optimization. In some cases, however, par-

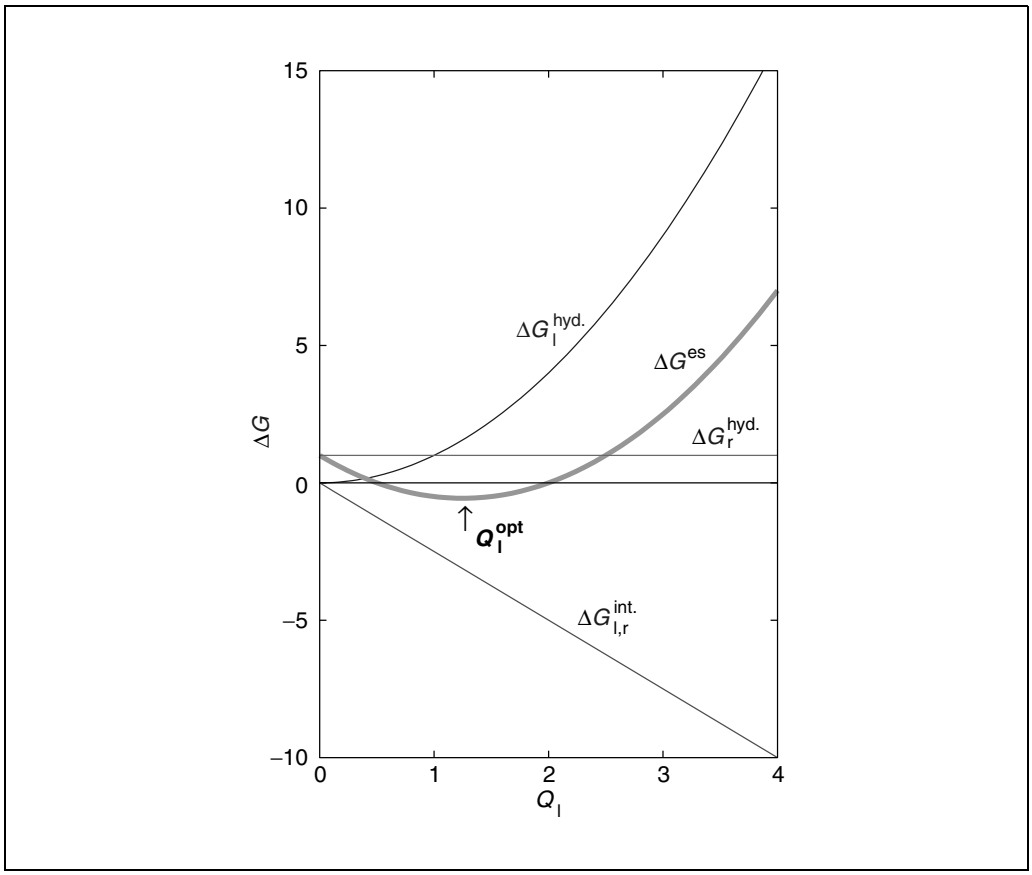


Figure 8.3.3 The electrostatic binding free energy (ΔG^{es}) varies quadratically with ligand charge (Q_l). The desolvation free energy of the ligand (ΔG^{hyd}_l) varies with the square of the charges on the ligand, while the free energy of interaction with the receptor ($\Delta G^{int}_{l,r}$) varies linearly with the ligand charges. As the receptor desolvation free energy (ΔG^{hyd}_r) is independent of the ligand charges, the net electrostatic binding free energy is a quadratic function of the ligand charge distribution. As a result, there is a single minimum on the free energy surface, corresponding to the optimal ligand charge distribution.

ticularly when constraints can not be satisfied in the optimization, it may be useful to allow the null space to be populated in the optimization, penalizing this population with a harmonic penalty. In addition, the constraints applied during optimization can be varied. Most often, the net charge of each residue is constrained to be an integer between $-1e$ and $+1e$, and no individual partial atomic charge is allowed to exceed $0.85e$ in magnitude. These constraints limit the optimization to the space of charges observed in amino acids, but may be removed or varied if desired.

Suggestions for Further Analysis

The calculations described here consider only the electrostatic contributions to the binding free energy. Other contributions, including steric interactions (favorable and unfavorable), covalent strain, the hydrophobic effect, and entropic terms, also play an important role in determining affinity and specificity of binding. For design applications in particular, but for analysis of existing complexes as well, consideration of at least some of these additional contributions will lead to a more complete and more accurate understanding of the system.

Literature Cited

Archontis, G., Simonson, T., and Karplus, M. 2001. Binding free energies and free energy components from molecular dynamics and Poisson-Boltzmann calculations. Application to amino acid recognition by aspartyl-tRNA synthetase. *J. Mol. Biol.* 306:307-327.

Bashford, D. and Karplus, M. 1990. pK_a's of ionizable groups in proteins: Atomic detail from a continuum electrostatic model. *Biochemistry* 29:10219-10225.

Chong, L.T., Dempster, S.E., Hendsch, Z.S., Lee, L.-P., and Tidor, B. 1998. Computation of electrostatic complements to proteins: A case of charge stabilized binding. *Protein Sci.* 7:206-210.

Chothia, C. 1974. Hydrophobic bonding and accessible surface area in proteins. *Nature (London)* 248:338-339.

Chothia, C. and Janin, J. 1975. Principles of protein–protein recognition. *Nature (London)* 256:705-708.

Davis, M.E. and McCammon, J.A. 1990. Electrostatics in biomolecular structure and dynamics. *Chem. Rev.* 90:509-521.

Froloff, N., Windemuth, A., and Honig, B. 1997. On the calculation of binding free energies using continuum methods: Application to MHC class I protein–peptide interactions. *Protein Sci.* 6:1293-1301.

Gilson, M.K. and Honig, B.H. 1987. Calculation of electrostatic potentials in an enzyme active site. *Nature (London)* 330: 84-86.

Gilson, M.K. and Honig, B. 1989. Destabilization of an alpha-helix-bundle protein by helix dipoles. *Proc. Natl. Acad. Sci. U.S.A.* 86:1524-1528.

Gilson, M.K., Sharp, K.A., and Honig, B.H. 1988. Calculating the electrostatic potential of molecules in solution: Method and error assessment. *J. Comput. Chem.* 9:327-335.

Gilson, M.A., Given, J.A., Bush, B.L., and McCammon, J.A. 1997. The statistical-thermodynamic basis for computation of binding affinities: A critical review. *Biophys. J.* 72:1047-1069.

Hendsch, Z.S. and Tidor, B. 1994. Do salt bridges stabilize proteins? A continuum electrostatics analysis. *Protein Sci.* 3:211-226.

Hendsch, Z.S. and Tidor, B. 1999. Electrostatic interactions in the GCN4 leucine zipper: Substantial contributions arise from intramolecular interactions enhanced on binding. *Protein Sci.* 8:1381-1392.

Hendsch, Z.S., Jonsson, T., Sauer, R.T., and Tidor, B. 1996. Protein stabilization by removal of unsatisfied polar groups: Computational approaches and experimental tests. *Biochemistry* 35:7621-7625.

Kangas, E. and Tidor, B. 1998. Optimizing electrostatic affinity in ligand–receptor binding: Theory, computation, and ligand properties. *J. Chem. Phys.* 109:7522-7545.

Kangas, E. and Tidor, B. 1999. Charge optimization leads to favorable electrostatic binding free energy. *Phys. Rev. E* 59:5958-5961.

Kangas, E. and Tidor, B. 2000. Electrostatic specificity in molecular ligand design. *J. Chem. Phys.* 112:9120-9131.

Kangas, E. and Tidor, B. 2001. Electrostatic complementarity at ligand binding sites: Application to chorismate mutase. *J. Phys. Chem. B.* 105:880-888.

Lee, L.-P. and Tidor, B. 1997. Optimization of electrostatic binding free energy. *J. Chem. Phys.* 106:8681-8690.

Lee, L.-P. and Tidor, B. 2001a. Barstar is electrostatically optimized for tight binding to barnase. *Nat. Struct. Biol.* 8:73-76.

Lee, L.-P. and Tidor, B. 2001b. Optimization of binding electrostatics: Charge complementarity in the barnase-barstar protein complex. *Protein Sci.* 10:362-377.

Misra, V.K., Sharp, K.A., Friedman, R.A., and Honig, B. 1994. Salt effects on ligand–DNA binding: Minor groove binding antibiotics. *J. Mol. Biol.* 238:245-263.

Misra, V.K., Hecht, J.L., Yang, A.-S., and Honig, B. 1998. Electrostatic contributions to the binding free energy of the λ cI repressor to DNA. *Biophys. J.* 75:2262-2273.

Mohan, V., Davis, M.E., McCammon, J.A., and Pettitt, B.M. 1992. Continuum model calculations of solvation free energies: Accurate evaluation of electrostatic contributions. *J. Phys. Chem.* 96:6428-6431.

Nicholls, A., Sharp, K.A., and Honig, B. 1991. Protein folding and association: Insights from the interfacial and thermodynamic properties of hydrocarbons. *Proteins: Struct. Funct. Genet.* 11:281-296.

Nohaile, M.J., Hendsch, Z.S., Tidor, B., and Sauer, R.T. 2001. Altering dimerization specificity by changes in surface electrostatics. *Proc. Natl. Acad. Sci. U.S.A.* 98:3109-3114.

Potter, M.J., Gilson, M.K., and McCammon, J.A. 1994. Molecule pK(a) prediction with continuum electrostatics. *J. Am. Chem. Soc.* 116:10298-10299.

Sarkar, C.A., Lowenhaupt, K., Horan, T., Boone, T.C., Tidor, B., and Lauffenbuger, D.A. 2002. Rational cytokine design for increased lifetime and enhanced potency using pH-activated "histidine switching." *Nat. Biotech.* 20:908-913.

Sharp, K.A. and Honig, B. 1990. Electrostatic interactions in macromolecules: Theory and applications. *Annu. Rev. Biophys. Biophys. Chem.* 19: 301-332.

Sharp, K.A., Nicholls, A., Fine, R.F., and Honig, B. 1991. Reconciling the magnitude of the microscopic and macroscopic hydrophobic effects. *Science* 252:106-109.

Spector, S., Wang, M.H., Carp, S.A., Robblee, J., Hendsch, Z.S., Fairman, R., Tidor, B., and Raleigh, D.P. 2000. Rational modification of protein stability by the mutation of charged surface residues. *Biochemistry* 39:872-879.

Sulea, T. and Purisima, E.O. 2001. Optimizing ligand charges for maximum binding affinity. A solvated interaction energy approach. *J. Phys. Chem. B* 105: 889-899.

van Vlijmen, H.W.T., Schaefer, M., and Karplus, M. 1998. Improving the accuracy of protein pK(a) calculations: Conformational averaging versus the average structure. *Proteins* 33:145-158.

Vanderbei, R.J. 1999. LOQO: An interior-point code for quadratic programing. *Optimization Methods and Software* 12:451-454.

Warwicker, J. and Watson, H.C. 1982. Calculation of the electric potential in the active site cleft due to α-helix dipoles. *J. Mol. Biol.* 157:671-679.

Xiao, L. and Honig, B. 1999. Electrostatic contributions to the stability of hyperthermophilic proteins. *J. Mol. Biol.* 289:1435-1444.

Yang, A.S., Gunner, M.R., Sampogna, R., Sharp, K., and Honig, B. 1993. On the calculation of pK(a)s in proteins. *Proteins* 15:252-265.

Zacharias, M., Luty, B.A., Davis, M.E., and McCammon, J.A. 1992. Poisson–Boltzmann analysis of the lambda-repressor-operator interaction. *Biophys. J.* 63:1280-1285.

Key References

Hendsch and Tidor, 1999. See above.

Contains a detailed description of the implementation of component analysis and its application to the GCN4 leucine zipper.

Lee and Tidor, 1997. See above.

Outlines the theory behind the optimization of electrostatic binding free energy.

Kangas and Tidor, 1998. See above.

A detailed description of the electrostatic optimization procedure, including a definition of electrostatic complementarity.

Internet Resources

http://web.mit.edu/tidor/www/residual

Obtaining the Residual Potential Web site.

http://trantor.bioc.columbia.edu/grasp

The Grasp Web site.

http://trantor.bioc.columbia.edu/delphi

The DelPhi Web site.

Contributed by David F. Green and Bruce Tidor
Massachusetts Institute of Technology
Cambridge, Massachusetts

Using DelPhi to Compute Electrostatic Potentials and Assess Their Contribution to Interactions

BASIC PROTOCOL

An important feature of biological function is the ability of the molecules to bind one another in a highly specific manner. Electrostatic interactions play a significant role in this regard and are important in protein-protein interactions, protein stability, and binding ligands and substrates. Several theoretical studies have attempted to calculate the electrostatic energy accurately (Sheinerman et al., 2000) and molecular dynamics simulations have produced a number of striking successes (Bash et al., 1987; McCammon, 1987; Beveridge and DiCapua, 1989; Straatsma and McCammon, 1991; Miyamoto and Kollman, 1993). However, the need to sample a large ensemble of conformational states limits this approach. Further, empirical methods have also been used to estimate binding free energies (Andrews et al., 1984; Williams et al., 1991; Pearlman and Rao, 1998). Yet, while these methods provide extremely useful qualitative measurements, they are generally not able to yield accurate quantitative results (Ajay and Murcko, 1995).

One common method of treating electrostatic interactions involves retaining an atomic-level description of the protein while using continuum methods to describe the solvent molecules. This approach is based on solving the Poisson-Boltzmann (PB) equation, which takes care of the effect of dielectric and ionic strength (Gilson and Honig, 1988; Honig et al., 1993). However, the PB equation can be solved analytically only for objects with a regular geometry. Since protein structures are highly irregular, solving the PB equation requires a numerical approach. This unit focuses on one of the common programs in use today for this type of calculation, DelPhi. The DelPhi program is based on the finite difference approximation method (see Background Information; Klapper et al., 1986; Nicholls and Honig, 1991; Rocchia et al., 2001). Although this method is widely used, careful consideration should be taken when addressing the accuracy of the electrostatic free energy calculated, as will be further elaborated.

The protocol described below focuses on the commercial version of DelPhi, an Insight II module, which can be purchased from Accelrys. This version has the advantages of being menu driven and accepts multiple input-file formats. There is also a standalone version available from the authors of DelPhi. See Internet Resources for links to both versions.

Note that the molecular dynamics package CHARMM (Brooks et al., 1983) has a routine to solve the PB equation numerically. Although this unit does not discuss how to use this routine, the numerical algorithm used in the CHARMM package is the same as that used in DelPhi.

Necessary Resources

Hardware

Silicon Graphics IRIS workstations

Software

Insight II modeling program and DelPhi module (Accelrys; see Internet Resources) *or*
DelPhi stand-alone program (Columbia University; see Internet Resources)

Contributed by Assaf Oron, Kannan Gunasekaran, Haim Wolfson, and Ruth Nussinov
Current Protocols in Bioinformatics (2003) 8.4.1-8.4.12

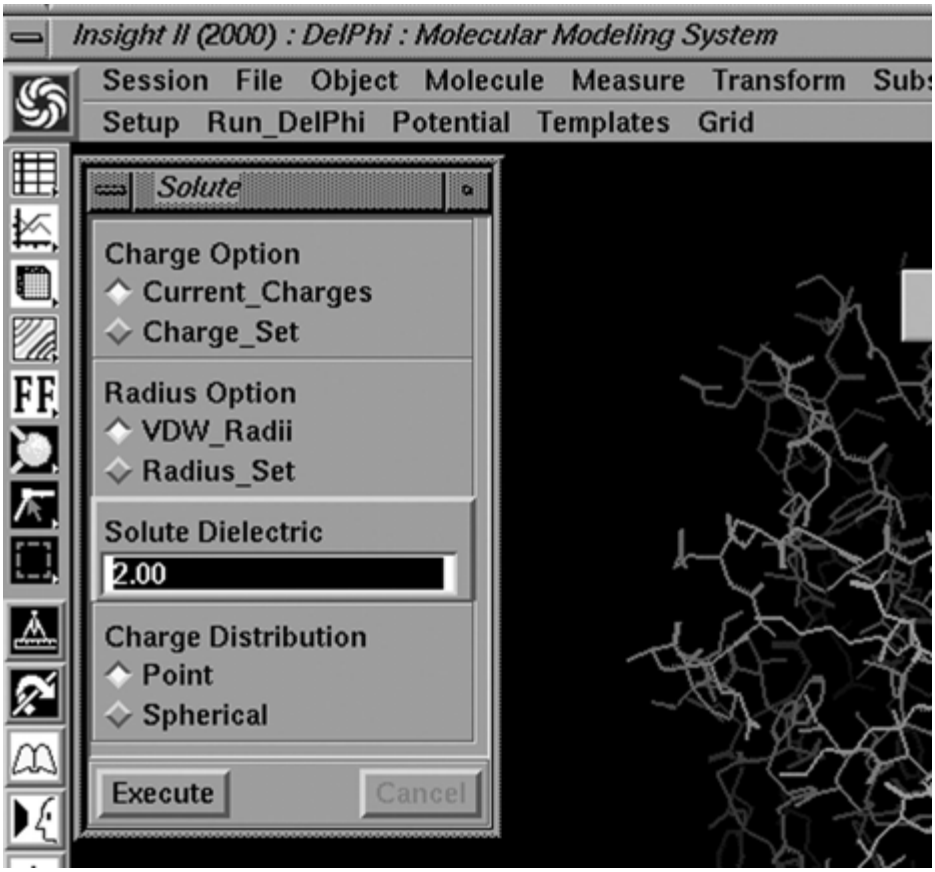


Figure 8.4.1 The Solute window. Assign the atomic charge, atomic radius, and solute dielectric here. For proteins, the dielectric constant ranges from 2.0 to 5.0.

Files

Three-dimensional structure of the unbound and bound proteins in PDB or other Insight II–readable format

Start the program

1. Start the Insight II program from any directory and access the DelPhi module by clicking the Modules icon under Insight II. Load the input file from any accessible directory by selecting Get under the Molecule menu.

Set up calculation parameters

2. Set solute parameters by selecting Solute from the Setup pull-down menu (Fig. 8.4.1). Select the desired options, choosing either the default (Current_Charges, VDW_Radii) or other sets (e.g., Charge_Set, Radius_Set) for the Charge and Radius Options. Also enter the Solute Dielectric and select either Point or Spherical for Charge Distribution option. Select Execute.

Atomic charges define the charge distribution of the system, while the radii help define the dielectric boundary. The default charge option (i.e., Current_Charges) uses the charges given in the source file. When choosing this option, careful consideration to assignment of all charged atoms should be taken. It is possible to select from a number of existing general charge sets designed for use with proteins or nucleic acids such as Protein Formal, CHARMM, AMBER, and others by selecting Charge_Set option. A user-defined charge set

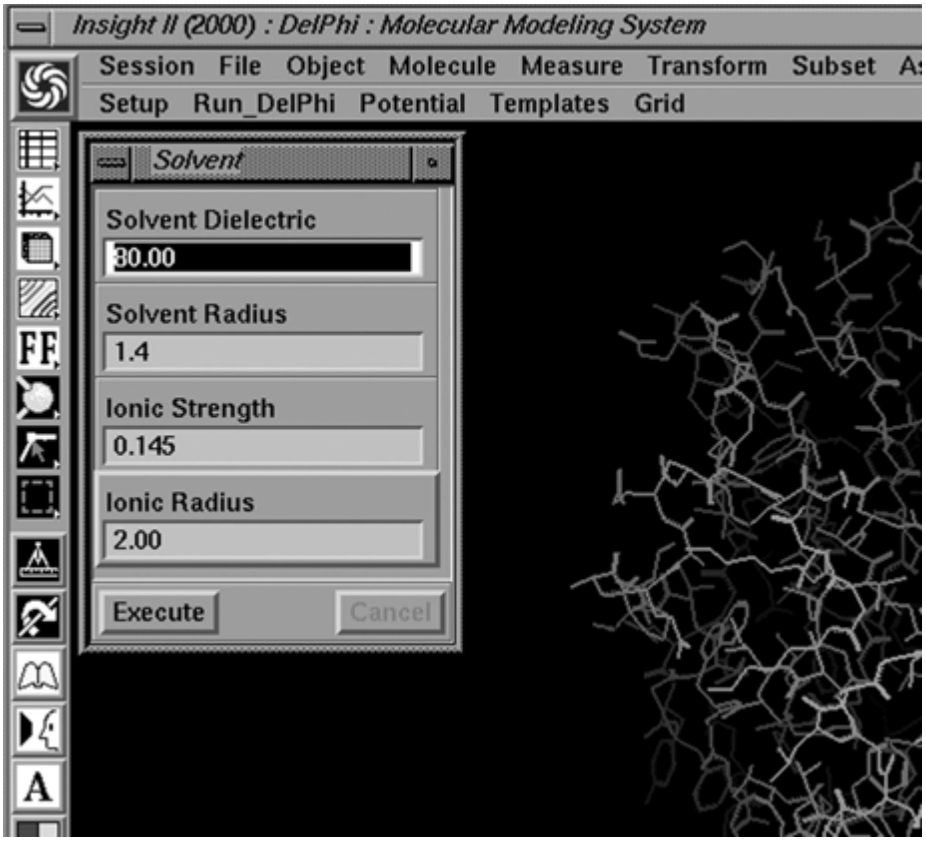


Figure 8.4.2 The Solvent window. Set the solvent characteristics (i.e., Solvent Dielectric, Solvent Radius, Ionic Strength, and Ionic Radius) here. A dielectric constant of 80 and solvent radius of 1.4 Å is used for water.

which corresponds to a certain molecule being examined can also be assigned by selecting Charge_Set.

The default option for atomic radius (i.e., VDW_Radii) uses the "Insight II" van der Waals radii. This set is a united atom scheme where hydrogens are assigned a radius of 0.0 and the radii of carbon, nitrogen, and oxygen are slightly larger than normal. Alternatively, similar to assigning charges, a user-defined set may be chosen or one of the general purpose sets given to specify the radius set by selecting Radius_Set. The choice of charge and radius sets depends on the molecule being examined and the question addressed.

The charge as a distribution option determines whether the atomic charge shape is to be treated as spherical or a point. This is useful when the user is dealing with a charge very close to an interface between different dielectric media. The default option is set to point.

The user should also assign the Solute Dielectric. Appropriate values usually range from 2.0 to 5.0 (see Commentary).

3. Set solvent parameters by selecting Solvent from the Setup pull-down menu and entering the Solvent Dielectric, Solvent Radius, Ionic Strength, and Ionic Radius (Fig. 8.4.2). Select Execute.

The first parameter is the Solvent Dielectric, which is usually set to 80 for water. The next, Solvent Radius, specifies the radius of the sphere of the solvent molecule. This parameter is significant in defining the accessible surface, which is performed by rolling this sphere on the surface of the protein. The Ionic Strength and Ionic Radius parameters simulate the

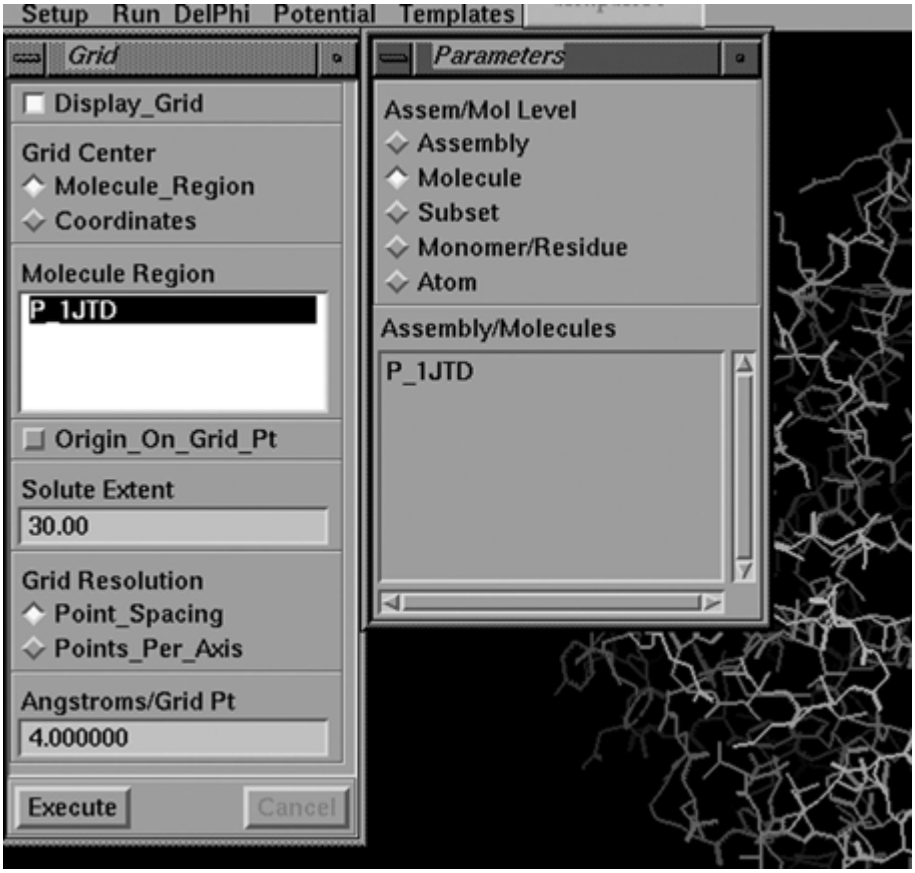


Figure 8.4.3 The Grid window. DelPhi calculates electrostatic energy by mapping the molecule onto a three-dimensional grid. The accuracy of the calculated electrostatic potentials depends heavily on the resolution of the grid. It is generally accepted that a grid resolution of 4 grid points/Å gives accurate enough results.

ions in the solvent. They specify ion concentration and average radius. The default values (i.e., 0.145 and 2.0 Å, respectively) are typical for physiological conditions.

4. Select Grid from the Setup pull-down menu (Fig. 8.4.3). Decide whether to show the grid (i.e., Display_Grid) and set the grid calculation characteristics Grid Center, Solute Extent, and Grid Resolution. Select Execute.

   Usually it is desirable to locate the molecule in the center of the grid. This is accomplished by selecting Molecule_Region under Grid Center. However, for specific electrostatic simulation the molecule can be placed at different areas of the grid by selecting Coordinates instead. The Molecule_Region box specifies the region of the molecule to be placed in the center of the grid. This is linked to the Assem/Mol Level aid window that lists the available objects under different molecular levels (e.g., Assembly, Molecule). The Origin_On_Grid_pt instructs whether the origin of the Cartesian space is to lie exactly on a grid point. This option is useful for displaying the grids with other graphics program.

   The Solute Extent is the percentage of the cubic box edge occupied by the solute's greatest Cartesian dimension. The Focussing command is usually used later when altering the solute extent (Fig. 8.4.4). A small Solute Extent gives relatively inaccurate results at a certain resolution (i.e., four grid points per angstrom), while a large value would also alter results by an inaccurate representation of the molecule in a limited area surrounded by water. Usually, a small Solute Extent is initially chosen (~30%). After going through an electrostatic potential calculation, one should go through a Focussing procedure (see next

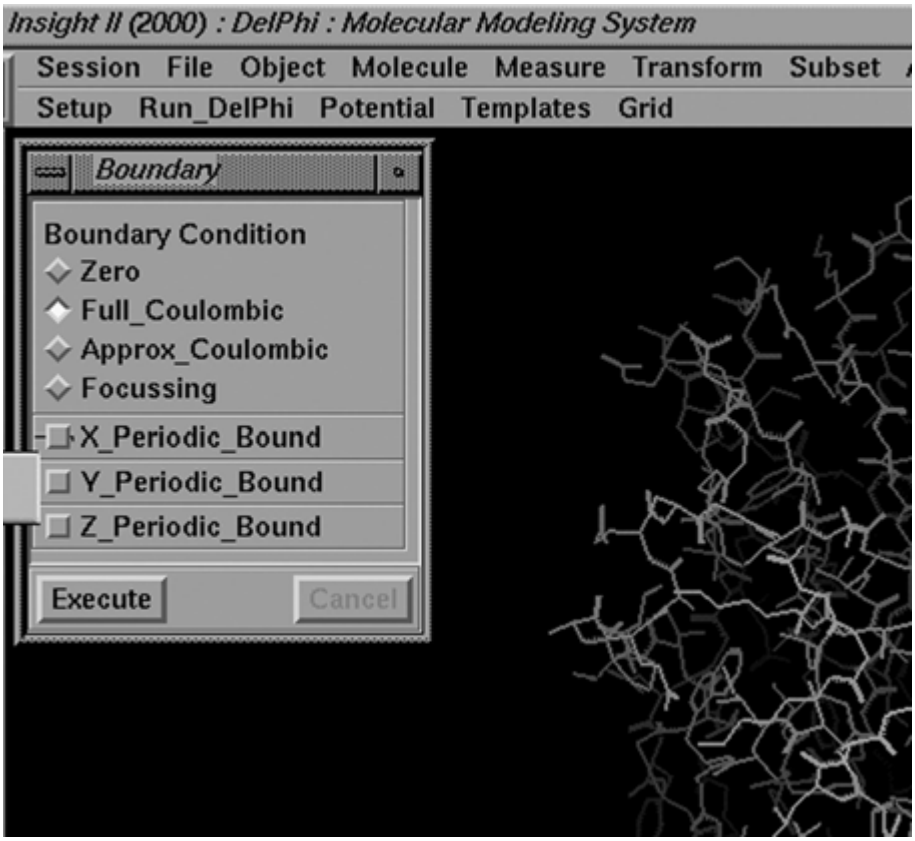


Figure 8.4.4 The Boundary window. Set grid boundaries here. The most common and recommended option is to use the Full_Coulombic choice.

step). During this procedure the user should choose a higher Solute Extent (~80%), but the boundary points of the focused grid may then be obtained from the potential map of the initial calculation. Using Focusing, a more accurate modeling of the solvent and solute is achieved.

Grid Resolution specifies the number of grid points that will be calculated in the particular run. Higher resolution will require a longer period of calculation. The user can specify the resolution directly with the Angstroms/Grid Pt parameter. Likewise, the user can indirectly specify the number of grid points along the box edge by selecting Points_Per_Axis followed by Number_of_Points. It is generally accepted that a grid resolution of 4 grid points/Å (or 0.25 Å/grid point) gives sufficiently accurate results

5. Set boundary conditions by selecting Boundary from the Setup pull-down menu (Fig. 8.4.4). Choose one of the three Boundary Condition choices available: Zero, Full_Coulombic, or Approx_Coulombic. If desired, also select the periodic boundaries. Select Execute.

   The grid boundary points cannot be calculated like the interior grid points (step 3), since there are no reference grid points surrounding them. Naturally, values that are assigned to these points will affect the electrostatic potential map. The Full_Coulombic option is the most recommended. This method calculates the potential due to every charge using the Debye-Huckel approximation. Other possible choices are Approx_Coulombic, which considers distances from charge centers, and the simplifying Zero option, which assigns a value of zero. Use of the Focussing command requires a potential map generated from an initial calculation (Fig. 8.4.5).

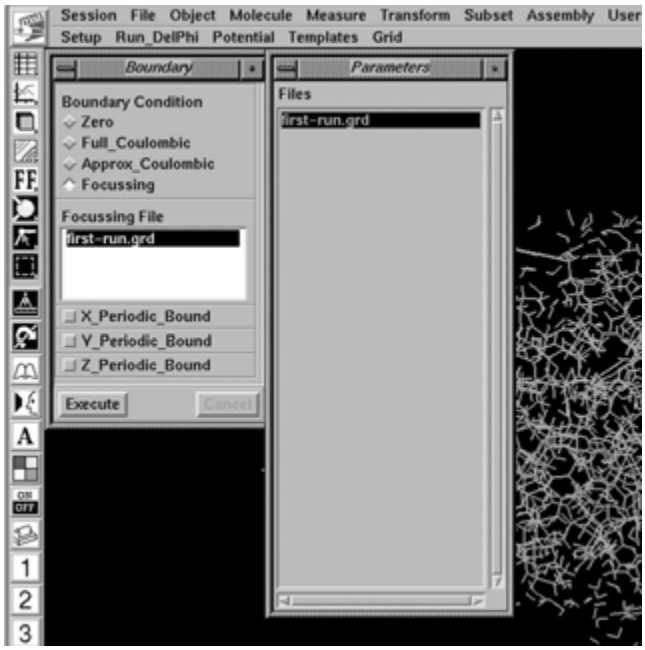


Figure 8.4.5 The boundary window. The use of Focussing procedure requires two runs of Delphi program. In the first run, a potential map with a small Solute Extent (~30%; Fig. 8.4.3) is generated. This potential map file is used for the second run in which the Focussing is chosen as the boundary condition as shown in the figure. Higher Solute Extent (~80%) could be chosen in the grid window during the second run. Also, the Auto_Get_Grid option must be turned on in the run window.

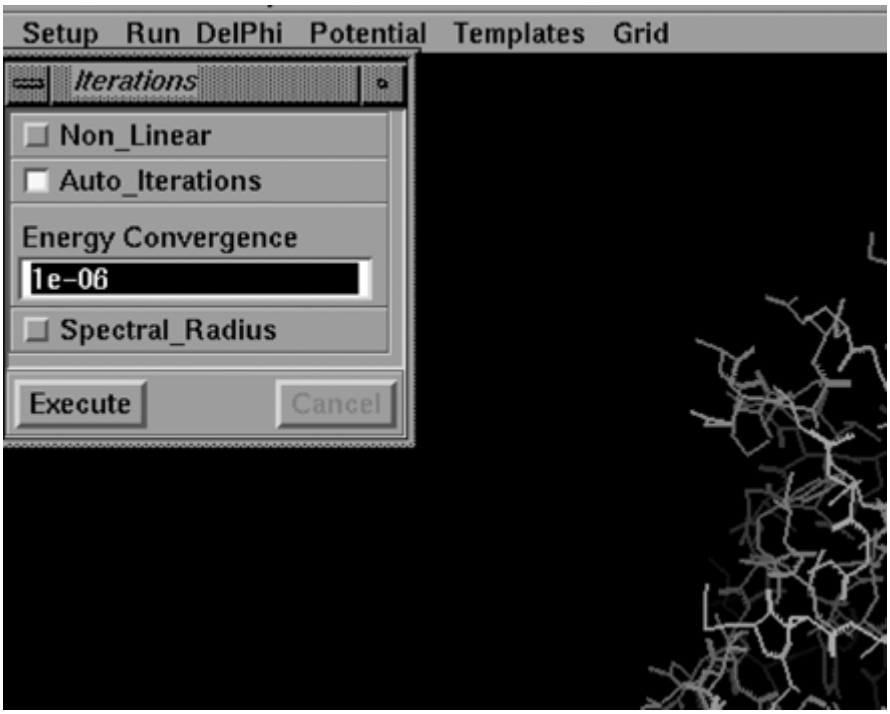


Figure 8.4.6 The Iterations window. Assign iteration characteristics here. Selecting Auto_ Iterations allows the continuation of calculations until an energy convergence is reached. The user may also choose between a nonlinear and linear calculation by turning on or off the Non_Linear option, respectively.

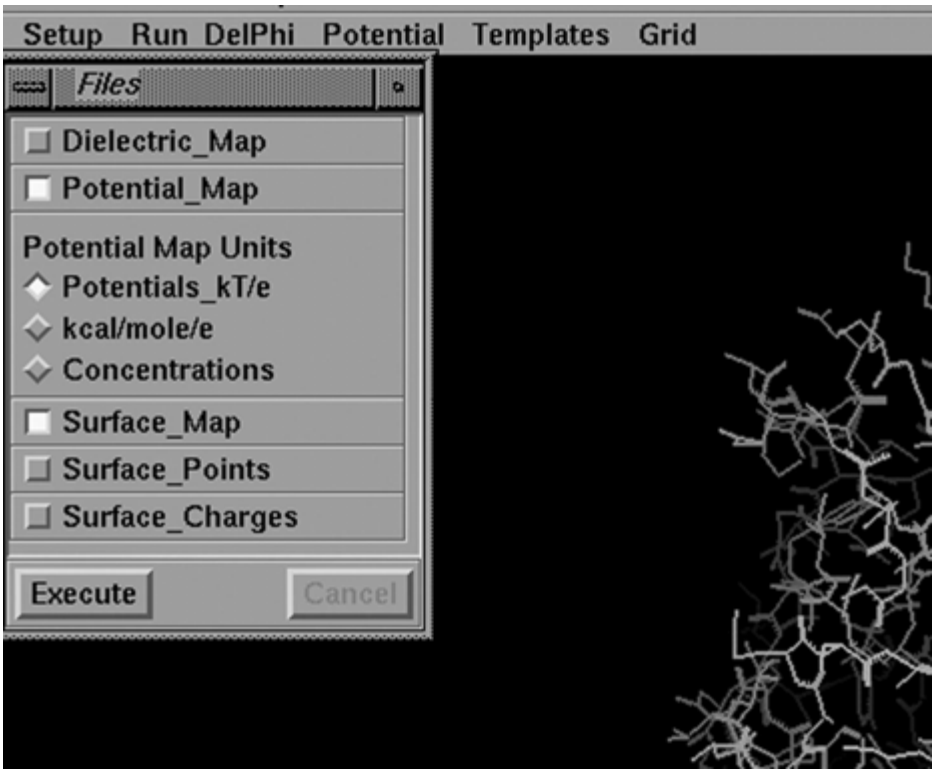


Figure 8.4.7 The Files window. The necessary data (file outputs) needed should be chosen from the various options.

The periodic boundary commands (i.e., X_Periodic_Bound, Y_Periodic_Bound, and Z_Periodic_Bound) are useful when there are repeated portions in the analyzed (symmetric) molecule. These commands may save calculation time. In these cases, the calculation is simplified due to similar potential values at opposite edges. In general, these commands are not applicable for proteins so they need not be turned on.

6. Select Iterations from the Setup pull-down menu to set iteration characteristics (Fig. 8.4.6). Choose a nonlinear equation by clicking the Non_Linear button, or leave it unselected to choose a linear equation. If desired, alter the number of iterations by selecting Auto_Iterations and setting the Energy Convergence value. Select Spectral_Radius if desired. Select Execute.

   The Non_Linear option uses a default value of 500 iterations, while the linear uses the nonlinear equation only for refinement. The user may also select the number of iterations to suit their needs by using the Auto_Iterations command. This option will continue the iteration until an energy convergence criterion is reached. The spectral radius, which is a critical parameter in the algorithm that solves for the potential, can also be used as the convergence criterion. The Spectral Radius is calculated at the optimal spectral value, at which the rate of convergence is peaked. It is also possible to enter a custom spectral radius value by selecting the Spectral_Radius option.

7. Select Files from the Setup pull-down menu to set the output file characteristics (Fig. 8.4.7). Select a Dielectric_Map and/or a Potential_Map. If selecting a Potential_Map, also set the Potential Map Units. If desired also select Surface_Map, Surface_Points, and Surface_Charge. Click execute at the bottom of the pull-down menu.

   A log file, which will be produced automatically and can be viewed through any text editor, contains overall information concerning the calculation. This file (`*.log`) includes data

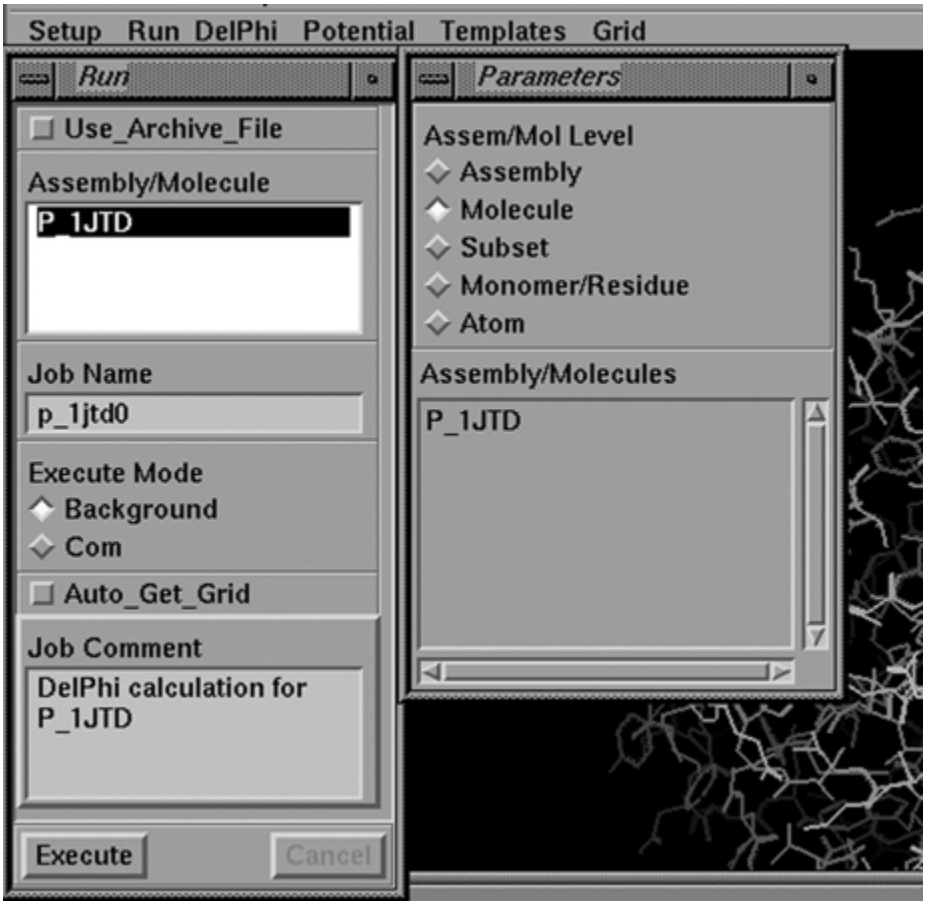


Figure 8.4.8 The Run DelPhi window. This command allows to specify the job details, such as to run as background or create an input file for command line submission at a later stage.

regarding the calculation process, files, and other general criteria used including any errors encountered. It should be noted that the energy values requested would also be specified in this file. All the output files are stored in the directory from which the program is executed.

A potential map file (`*.grd`*) contains the electrostatic potential values calculated during the procedure. In the files pull-down menu, Potential Map should be chosen and a requested set of units specified.*

A dielectric map file (`*.eps`*) specifies dielectric values at each point. This file might be used also to define surface boundary. A dielectric map output will be produced when the Dielectric Map option is chosen.*

A modified coordinate file (`*.atm`*) contains the new coordinates of the molecule used, radius and charge information that were applied earlier. The file is given in a PDB format with the occupancy and temperature factor fields replaced by the radius and charge information. The surface files (Surface_Map, Surface_Points, and Surface_Charge) contain the surface characteristics of the molecule being studied, and these output files (*`*.srf`* and *`*.sch`*) may be used with other programs such as GRASP (http://trantor.bioc.columbia.edu/grasp).*

Perform the calculation

8. Run the program by selecting Run DelPhi from the main toolbar (Fig. 8.4.8). Give a Job Name and specify which atoms or molecules will be included in the calculation

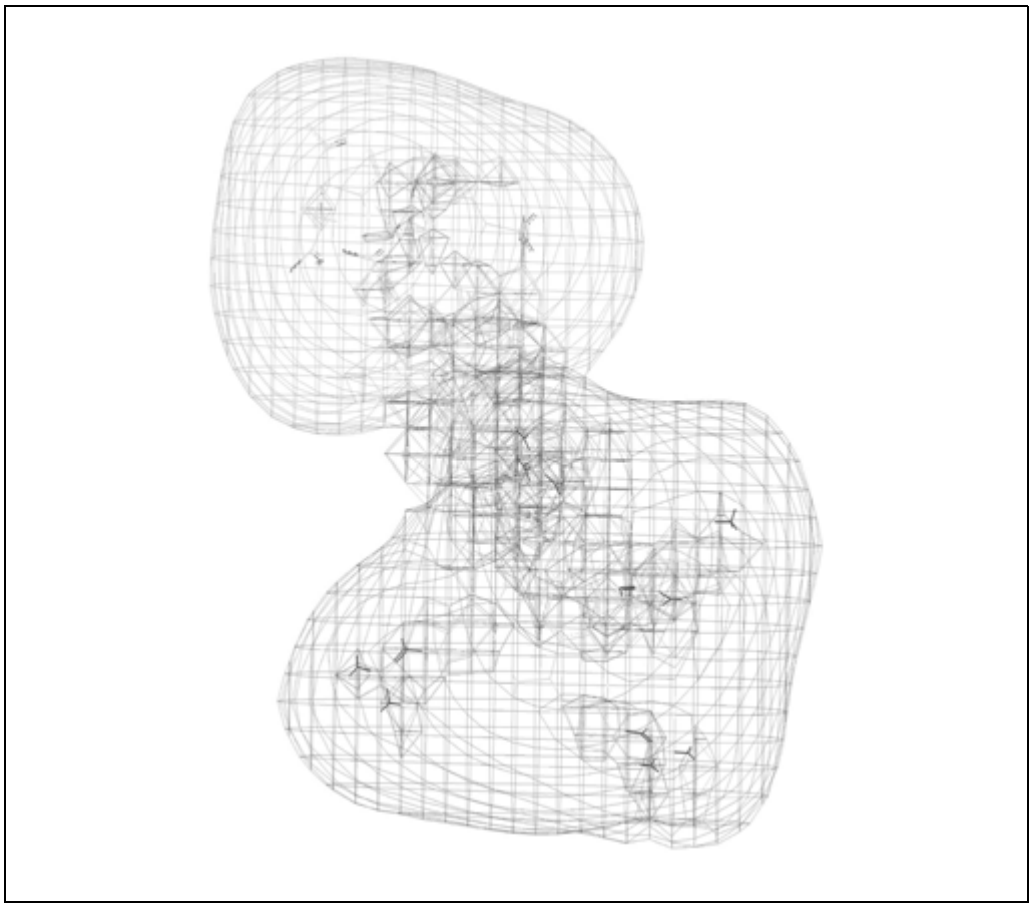


Figure 8.4.9 The potential map calculated for an α-helix which has positively charged residues at the N terminus and negatively charged residues at the C-terminus (sequence: RKHRRAAAAAADEDE). The potential map is plotted using the contour program available with insight II: −1 kcal/mole/e contour is displayed in red, and +1 kcal/mole/e contour is displayed in green. *This black and white facsimile of the figure is intended only as a placeholder; for full-color version of figure go to http://currentprotocols.com/colorfigures.*

by selecting the appropriate file from the list that will appear in the Assembly/Molecule text box. Choose to automatically submit this job and execute it in the background by selecting Background for Execute Mode, or store the input file for manual command-line submission by selecting Com. The Auto_Get_Grid option must be turned on when using the Focussing precedure. For manual submission, return to an X-term window to access the command line, and enter `job-name.csh &` to execute the program.

The information given in the Job Comment will be stored in the output log file, which also lists the names of the input and output files, atomic charges and radii, calculation parameters, and the energy results. The potential map output file (`*.grd`) can be loaded into the contour program that is available with insight II (refer to the Insight II manual to learn how to use the program), and the electrostatic energy surface can be viewed (Fig. 8.4.9).

GUIDELINES FOR UNDERSTANDING RESULTS

Although the use of a continuum electrostatic model successfully evaluates the binding free energy in some cases, it does have limitations. The first limitation comes from the use of a numerical solution, instead of analytical, in order to solve the PB equation (see unit introduction). Froloff et al. (1997) used the DelPhi program together with calculating other free energy contributions to validate calculated versus experimental values. This

was done on eight cases of MHC class I protein–peptide complexes. The results produced were in low agreement with the experimental values and moreover, the calculation failed to reproduce the correct order of peptide binding energies. However, it should be pointed out that these calculations do not directly impugn the accuracy of the electrostatic calculation made by DelPhi, since the other energetic values play a significant role in the binding process (specifically, the nonpolar free energies).

These results could be explained partially based on the important factors mentioned above (i.e., atomic location, charge assignment, and choice of dielectric constants). Moreover, in calculating binding energy for protein-protein interactions, the theoretical modeling of the bound state may not accurately reflect the actual binding state. In some bound states there is a layer of water between the two proteins (or molecules) that does not necessarily agree with the continuum model. Assigning a dielectric constant of 80 to these solvent molecules or, on the other hand, regarding them as part of the protein will not reflect their true contribution to the electrostatic potential calculated. Therefore, careful consideration of these issues should lead to better estimation of electrostatic potentials.

COMMENTARY

Background Information

The DelPhi program can be used as a stand-alone product or as a module accessible via graphical interface programs such as Insight II, Accelrys's molecular modeling program. The DelPhi application in the Insight II program runs on Silicon Graphics IRIS workstation. The program also comes with a detailed manual and step-by-step instructions on how to use it. In order to run the program a three-dimensional structure of the unbound and bound proteins is needed, drawn from experimental data (crystallography, NMR) or modeling techniques.

The DelPhi program method

The DelPhi program calculates the electrostatic free energy based on continuum electrostatics and by solving the Poisson-Boltzmann equation numerically. It should be noted that when calculating binding free energies, other energetic considerations should be taken into account. The most significant being the nonpolar free energy that relates to the hydrophobic effect (Pearlman and Rao, 1998).

The calculation of the electrostatic free energy involves two steps. The first step is the estimation of coulombic energy. The second is the solvation free energy. The coulombic energy is the free energy of assembling the atomic charges from infinity in a medium of dielectric constant equal to that of the protein's interior. The solvation free energy is the result of transferring the protein from a medium of its own dielectric constant to water. Assessing these quantities for each protein and for the bound conformation enables one to calculate the electrostatic free energy.

The nonlinear Poisson-Boltzmann equation takes into account the effects of field reduction due to the medium and its boundaries as well as all charges. The PB equation is defined as follows:

$$\nabla[\varepsilon(r)\nabla\Phi(r)] - \overline{\kappa^2}\Phi(r)\left[1+\frac{\Phi(r)^2}{6}+\frac{\Phi(r)^4}{120}+\ldots\right] + 4\prod\rho_{int}(r) = 0$$

where, ε is the dielectric constant of the solvent, Φ the electrostatic potential (in kT/e), $\tilde{n}$ the charge density, and κ the Debye-Huckel inverse length (dependent on temperature and ionic strength). The dielectric constant reflects reorientation in the medium due to an external electrostatic field. The PB equation provides a physically complete treatment of electrostatic interactions in solution, but as used it is nevertheless approximate.

The DelPhi program uses the finite-difference solutions to the PB equation, first by mapping the molecule into a grid and then calculating the electrostatic potential at every grid point. It must be noted that for every grid point, the PB equation is satisfied. Since the potential at each grid point depends on its neighbors' potentials, the calculation is repeated iteratively until a self-consistent solution for the electrostatic energy of the entire system is found. This type of approach is termed the Finite-Difference method. The accuracy of the calculated electrostatic potentials depends heavily on the resolution of the grid.

The next step involves calculating the electrostatic energy using the electrostatic potentials and charges in the molecule. The electrostatic energy is calculated by completing an energetic cycle. For each protein and bound ligand configuration, the electrostatic energy will be calculated in vacuum (dielectric constant of 1) and in the solvent (dielectric constant of 80 for water). The electrostatic component of the solvation energy of each configuration is defined as the difference in the total electrostatic energies of these two calculations. The program offers two different methods to compute the electrostatic energy; (1) total plus grid and (2) reaction field, which are actually two representations of the PB equation. The energetic values which the program outputs (chosen by selecting Energies from the Setup pull-down menu) are described below.

Total plus grid. This is the total electrostatic energy of the charged molecule including the grid energy. It is calculated as half the sum of the charge on each atom times the total potential at the atom position. Computation of the potential is achieved by linear interpolation of the values at the surrounding grid points.

Reaction field. This is the energy resulting from the polarization in the surrounding of the molecule at its position. Computing is done by first calculating the induced surface charge at each surface point and then these charges are used to calculate the potential at every charge.

Coulombic energy. The energy required to bring the charges of the molecule from infinity to their final resting place using the interior dielectric.

As mentioned previously, there are several different models for calculating the electrostatic energy. One model, which the authors have found important and useful, is the generalized Born/Surface area (GB/SA) model (Tobias, 2001). This model is based on the Born equation and is an approximation to the PB equation (Dominy and Brooks, 1999). The model calculates surface contributions (based on solvent accessible surface) and electrostatic contributions based on pair-wise interaction between charged atoms (Coulomb's law).

Critical Parameters and Troubleshooting

Atomic placement and charge assignment

It seems that one of the main inaccuracies of the method might be a result of inaccuracies in the crystal coordinates during the model building and refinement procedures. Before using the program, it is important to go through possible inaccuracies in the three-dimensional structure. Supporting this assumption is work done by Froloff et al. (1997), who inspected the difference in free energy calculation when performing a restrained energy minimization in comparison to the original crystal structure. In this case a difference of up to 8 kcal/mol was observed in comparison to the crystal structure for eight MHC class I proteins.

Another important issue is charge assignment, which is done in the early steps of running the program. The electrostatic potential is highly dependent on the placement of charged atoms in the molecule. Attention should be paid on assuring that accurate values of charge be placed on charged amino acid residues.

Dielectric constant

The DelPhi program is based on a continuum electrostatic model, which applies a homogeneous dielectric constant to the solvent and solute rather then explicitly accounting for the polarization of each atom. This model simplifies the calculation, yet the choice of an adequate internal dielectric constant of a macromolecule is crucial in the calculation (Gilson and Honig, 1986; Schutz and Warshel, 2001). The dielectric constant reflects reorientation in the lattice due to an external electrostatic field. The common dielectric constant used for water is 80 due to significant reorientations. An accurate calculation of the protein dielectric constant would take into account the effects on the different groups in the molecule. Consideration of dielectric constants regarding the specific protein should be taken into account when choosing its assignment during the procedure. A dielectric constant of 2 to 5 is usually used for proteins, though higher values have been assigned. Explicit modeling of structural responses imply more appropriate use of smaller dielectric constants (Alexov and Gunner, 1999; Nielsen et al., 1999).

Literature Cited

Ajay, A. and Murcko, M.A. 1995. Computational methods to predict binding free energy in ligand-receptor complexes. *J. Med. Chem.* 38:4953-4967.

Alexov, E.G. and Gunner, M.R. 1999. Calculated protein and proton motions coupled to electron transfer: Electron transfer from QA to QB in bacterial photosynthetic reaction centers. *Biochemistry* 38:8253-8270.

Andrews, P.R., Craik, D.J., and Martin, J.L. 1984. Functional group contributions to drug-receptor interactions. *J. Med. Chem.* 27:1648-1657.

Bash, P.A., Singh, U.C., Brown, F.K., Langridge, R., and Kollman, P.A. 1987. Free energy calculations by computer simulation. *Science* 235:574-576.

Beveridge, D.L. and DiCapua, F.M. 1989. Free energy via molecular simulation: Applications to chemical and biomolecular systems. *Biophys. Chem.* 18:431-492.

Brooks, B.R., Bruccoleri, R.E., Olafson, B.D., States, D.J., Swaminathan, S., and Karplus, M. 1983. CHARMM: A program for macromolecular energy minimization and dynamic calculations. *J. Comput. Chem.* 4:187-217.

Dominy, B. and Brooks, III, C.L. 1999. Development of a generalized Born model parameterization for proteins and nucleic acids. *J. Phys. Chem.* 103:3765-3773.

Froloff, N., Windemuth, A., and Honig, B.H. 1997. On the calculation of binding free energies using continuum methods: Application to MHC class I protein-peptide interactions. *Protein Science* 6:1293-1301.

Gilson, M.K. and Honig, B.H. 1986. The dielectric constant of a folded protein. *Biopolymers* 25:2097-2119.

Gilson, M.K. and Honig, B.H. 1988. Energetics of charge-charge interactions in proteins. *Proteins* 3:32-52.

Honig, B.H., Sharp, K., and Yang, A.S. 1993. Macroscopic models of aqueous solution: Biological and chemical applications. *J. Phys. Chem.* 97:1101-1109.

Klapper, I., Hagstrom, R., Fine, R., Sharp, K., and Honig, B.H. 1986. Focusing of electric fields in the active site of Cu-Zn superoxide dismutase: Effects of ionic strength and amino-acid modification. *Proteins* 1:47-59.

McCammon, J.A. 1987. Computer-aided molecular design. *Science* 238: 486-491.

Miyamoto, S. and Kollman, P.A. 1993. Absolute and relative binding free energy calculations of the interaction of biotin and its analogs with streptavidin using molecular dynamics/free energy perturbation approaches. *Proteins* 16:226-245.

Nicholls, A. and Honig, B.H. 1991. A rapid finite difference algorithm, utilizing successive over-relaxation to solve Poisson-Boltzmann equations. *J. Comput. Chem.* 12:435-445.

Nielsen, J.E., Andersen, K.V., Honig, B., Hooft, R.W.W., Klebe, G., Vriend, G., and Wade, R.C. 1999. Improving macromolecular electrostatics calculations. *Protein Eng.* 12:657-662.

Pearlman, D.A., and Rao, G.B. 1998. Free energy calculation: Methods and applications. *In* The Encyclopedia of Computational Chemistry, vol. 2. (Schleyer, P.v.R., Jorgensen, W.L., Schaefer III, H.F., Schreiner, P.R., and Thiel, W., eds.), pp.1053-1058. John Wiley & Sons, Chichester, U.K.

Rocchia, W., Alexov, E., and Honig, B. 2001. Extending the applicability of nonlinear Poisson-Boltzmann equation: Multiple dielectric constants and multivalent ions. *J. Phys. Chem. B.* 105:6507-6514.

Schutz, C.N. and Warshel, A. 2001. What are the dielectric "constants" of proteins and how to validate electrostatic models? *Proteins* 44:400-417.

Sheinerman, F.B., Norel, R., and Honig, B. 2000. Electrostatic aspects of protein-protein interactions. *Curr. Opin. Struct. Biol.* 10:153-159.

Straatsma, T.P. and McCammon, J.A. 1991. Theoretical calculations of relative affinities of binding. *Method Enzymol.* 202:497-511.

Williams, D.H., Cox, J.P.L., Doig, A.J., Gardner, M., Gerhard. U., Kaye, P.T., Lal, A.R., Nicholls, I.A., Salter, C.J., and Mitchell, R.C. 1991. Toward the semiquantitative estimation of binding constants. Guides for peptide-peptide binding in aqueous solution. *J. Am. Chem. Soc.* 113:7020-7030.

Tobias, D.J. 2001. Electrostatic calculations: Recent methodological advances and applications to membranes. *Curr. Opin. Struct. Biol.* 11:253-261.

Key Reference

Honig et al., 1993. See above.

Covers the fundamental theoretical and practical aspects of DelPhi.

Internet Resources

http://www.accelrys.com

Accelrys Web site.

http://trantor.bioc.columbia.edu/delphi

Web site to obtain the source code of DelPhi program, available at the Department of Biochemistry, Columbia University.

Contributed by Assaf Oron and Haim Wolfson
Tel Aviv University
Tel Aviv, Israel

Kannan Gunasekaran
Laboratory of Experimental and Computational Biology
National Cancer Institute
Frederick, Maryland

Ruth Nussinov
Laboratory of Experimental and Computational Biology
SAIC-Frederick
Frederick, Maryland and
Tel Aviv University
Tel Aviv, Israel

Searching the MINT Database for Protein Interaction Information

Gianni Cesareni,[1] Andrew Chatr-aryamontri,[1] Luana Licata,[1] and Arnaud Ceol[1]

[1]University of Rome Tor Vergata, Rome, Italy

ABSTRACT

The Molecular Interactions Database (MINT) is a relational database designed to store information about protein interactions. Expert curators extract the relevant information from the scientific literature and deposit it in a computer readable form. Currently (April 2008), MINT contains information on almost 29,000 proteins and more than 100,000 interactions from more than 30 model organisms. This unit provides protocols for searching MINT over the Internet, using the MINT Viewer. *Curr. Protoc. Bioinform.* 22:8.5.1-8.5.13. © 2008 by John Wiley & Sons, Inc.

Keywords: MINT • protein-protein interaction • database

INTRODUCTION

MINT is a relational database designed to store information about protein interactions (see Fig. 8.5.1; Chatr-aryamontri et al., 2007). Expert curators extract the relevant information from the scientific literature and deposit it in a computer readable form. MINT, IntAct (Kerrien et al., 2007), and the Database of Interacting Proteins (DIP; Salwinski et al., 2004) are founders and active members of the IMEx consortium, which shares curation efforts and exchanges completed records on molecular interaction data, similar to the successful global collaborations achieved by the International Nucleotide Sequence Databases (INSD) that have been developed and maintained collaboratively among DDBJ, EMBL, and GenBank for over 18 years. Other well established protein interaction databases currently accessible on the Web include: MIPS (Guldener et al., 2006), BioGRID (Stark et al., 2006) and HPRD (Mishra et al., 2006).

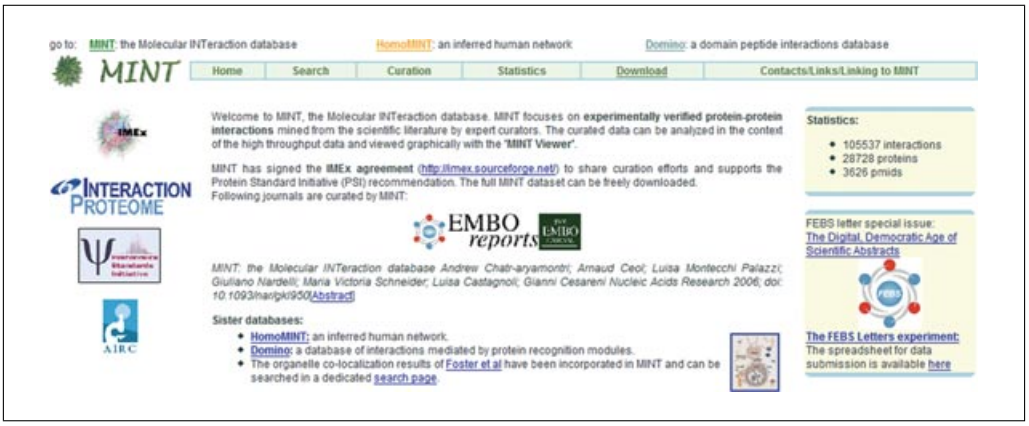


Figure 8.5.1 MINT homepage.

Current Protocols in Bioinformatics 8.5.1-8.5.13, May 2008
Published online May 2008 in Wiley Interscience (www.interscience.wiley.com).
DOI: 10.1002/0471250953.bi0805s22

SEARCHING MINT OVER THE INTERNET

MINT can be accessed freely at *http://mint.bio.uniroma2.it/mint*. This protocol describes how to retrieve interaction data and visualize the resulting interaction network with the MINT viewer. The present data structure of MINT and its links to external data sources is also described.

Necessary Resources

Hardware

Computer with Internet connection

Software

Browser suitable for simple searches of the MINT database (e.g., Firefox, *http://www.mozilla.org/firefox*; Safari, *www.apple.com/safari*; *or* Internet Explorer, *http://www.microsoft.com*)

Files

No local files required

1. Open your favorite browser and connect directly to MINT at *http://mint.bio.uniroma2.it/mint*.
2. Press the SEARCH button to go to the search page (Fig. 8.5.2), which proposes a form to interrogate the database.
3. Search proteins using identifiers of external databases: UniProt, Ensembl, Flybase, SGD, Wormbase, OMIM, HUGO, or Reactome. Alternatively, look for a gene or protein name or any keywords described in the UniProt knowledge base. Boolean operators are not supported.

   For instance, if one were interested in the interactions mediated by the human protein Lck, without knowing its accession number, one should enter Lck in the "gene/protein name" search field.

   It is also possible to display all interactions described in any chosen publication by filling the PMID field with the article's PubMed identifier (PMID).

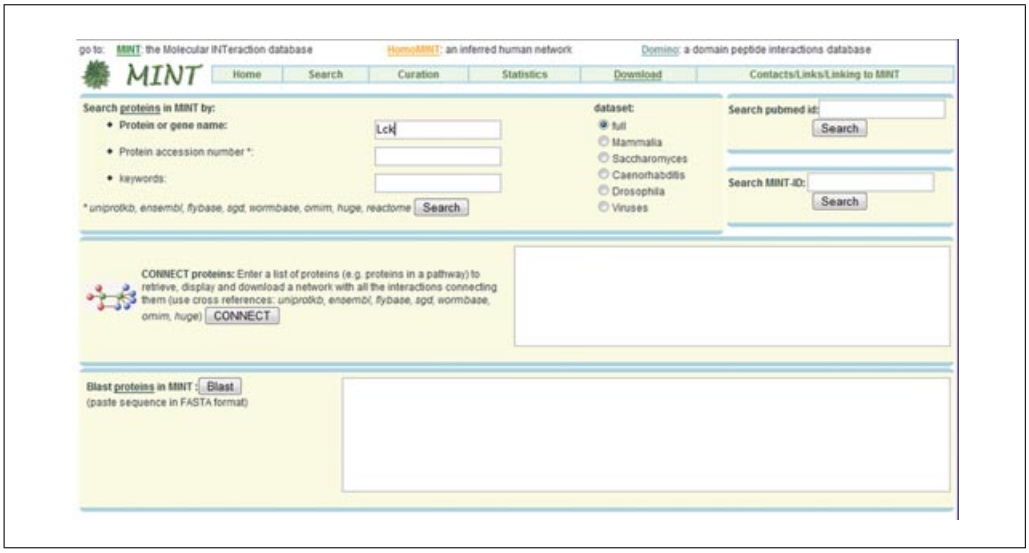


Figure 8.5.2 Search page. SwissProt/Trembl accession numbers, gene and protein names, or keywords can be used to search the database.

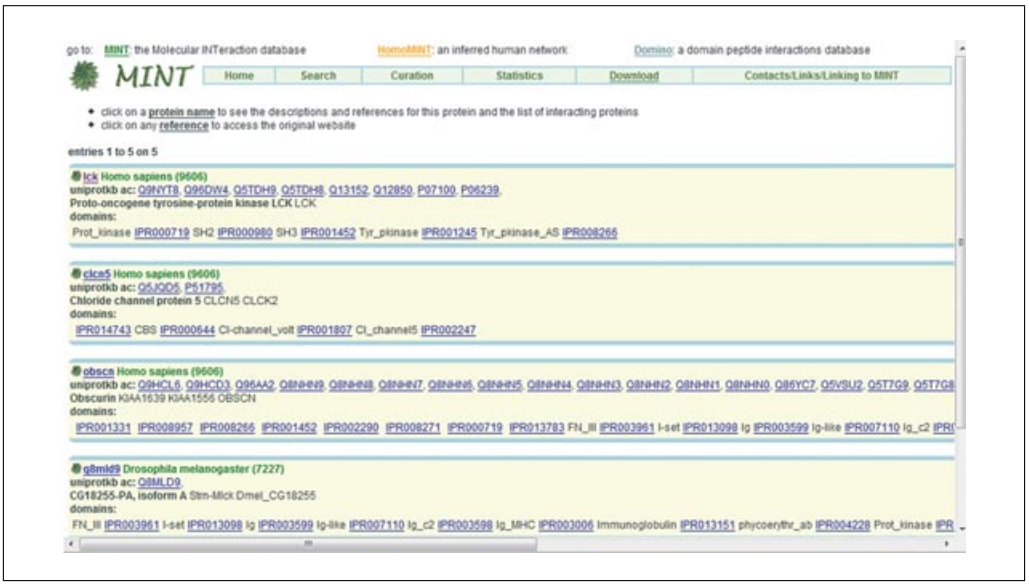


Figure 8.5.3 Result of a search with the protein name Lck. Each protein ID can be clicked to obtain more detailed information about the selected protein.

4. This initial query can return more than one protein. For each retrieved protein, the following information is reported: the gene name, organism, UniProt accession number, protein name, and a list of the annotated domains, linked to Interpro, if available.

   For instance, the search for Lck, performed in the current version of MINT, yields four proteins (Fig. 8.5.3) containing the word Lck either in the gene name, protein name, or protein description.

5. Click on the gene name to gain access to more details about the selected protein in the results page. The left frame contains a summary of the information provided by UniProt about this protein. It includes gene names and synonyms, description of the protein, keywords, diseases, the domain structure of the protein linked to Interpro, and a list of GO terms associated with the protein. Cross-references to several databases, e.g., PDB (*UNIT 1.2*), Ensembl (*UNIT 1.15*), or OMIM (*UNIT 1.9*) are also included (Fig. 8.5.4, left frame). One last link redirects to the European Molecular Biology Laboratory (EMBL) ADAN database where predicted interactions are automatically searched.

   For example, click on "Proto-oncogene tyrosine-protein kinase LCK". The left frame contains the type of information described above, while the right frame of the page lists the twenty interactions involving Lck stored in the MINT database (Fig. 8.5.4, right frame).

6. Click on the number reported in the interactions column, corresponding to the number of distinct pieces of experimental evidence supporting the interaction that are stored in MINT, and a new frame appears in which the MINT "core information" for the chosen interaction is displayed in the left frame.

   For example, in Figure 8.5.5, all the information stored in MINT supporting the interaction between Cbl and Lck is shown. The gene names of the two interacting proteins and the domains that are responsible for the interaction are reported side by side in a table. The interacting domains are represented as amino acid ranges, and, whenever this range contains either a structurally or functionally defined domain, the corresponding name and reference in Interpro (UNIT 2.7) is also indicated (i.e., SH3 or OD, oligomerization domain). The domain identification method field specifies how the region that is sufficient for interaction was experimentally identified.

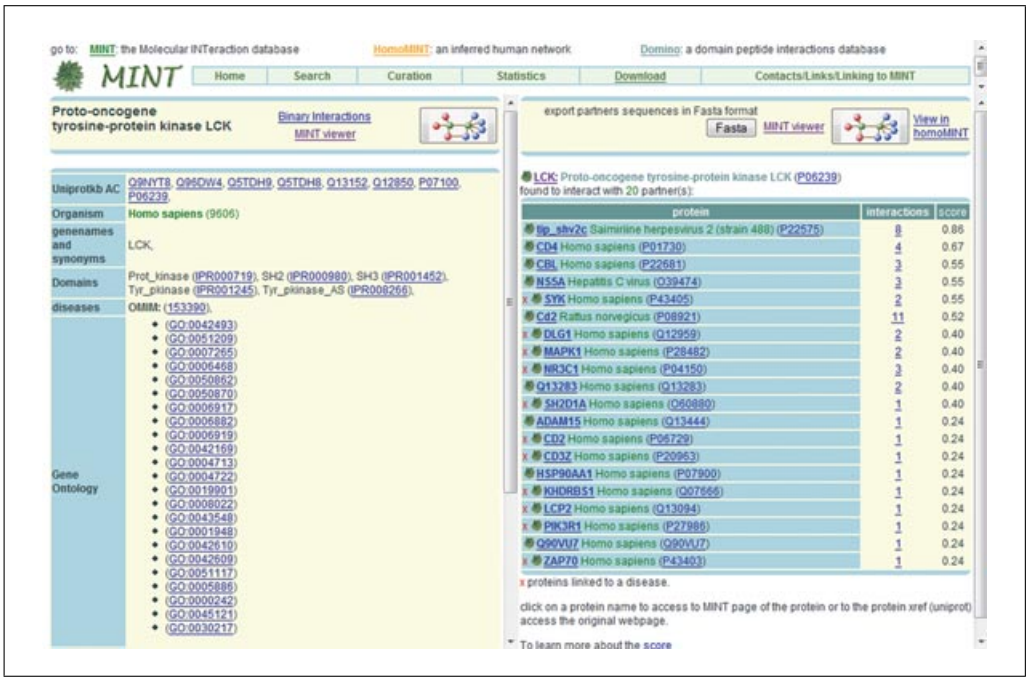


Figure 8.5.4 Result of clicking on the gene name on the protein name search results page (Fig. 8.5.3).The panel on the left displays information about the selected protein. Most of this information is automatically extracted from the SwissProt/Trembl database. The panel on the right lists all the interactions, among the ones stored in MINT, in which the selected protein participates. The MINT VIEWER icon permits graphical display of the interactions listed in the interaction table.

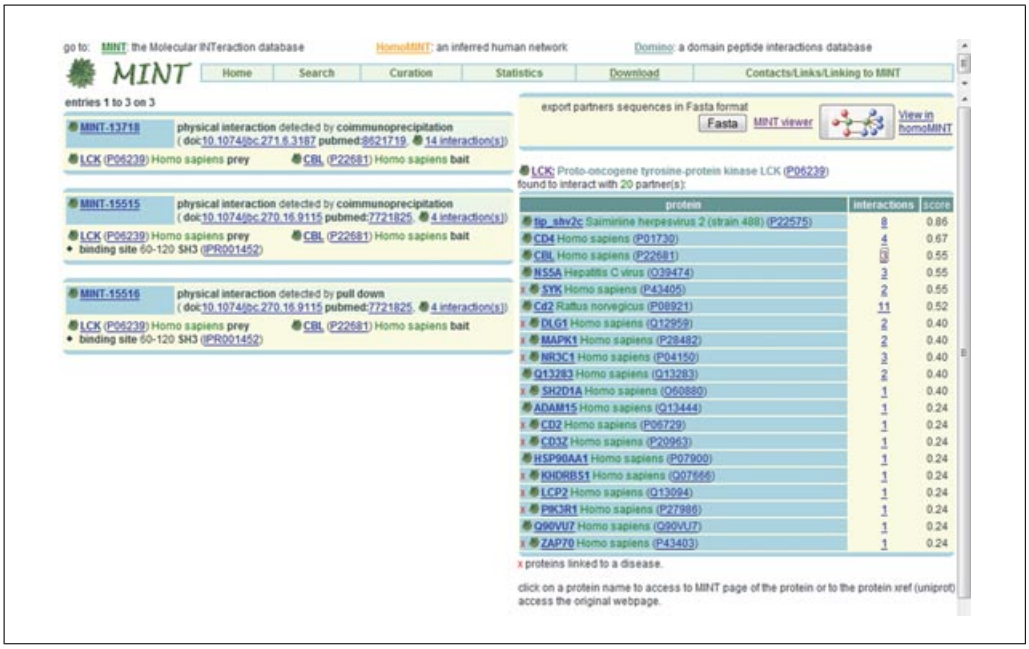


Figure 8.5.5 Result of clicking on the interaction number in the right panel of the previous page (Fig. 8.5.4). A new table is presented that contains detailed information about the experimental evidences supporting the interaction.

In each displayed entry it is indicated whether protein A simply binds to B or in addition chemically modifies it. In this example, the Lck binds to Cbl. The experimental method used to demonstrate the functional relationship and the PubMed ID (PMID) of the article reporting the interaction are also given. There are convenient hyperlinks to the source databases for information about each interacting protein and to the abstract(s) of the article(s) that describes the interaction.

MINT VIEWER

ALTERNATE PROTOCOL

Alternatively the interaction network can be explored through a graphic interface: the MINT viewer (Fig. 8.5.6).

Necessary Resources *(also see Basic Protocol 1)*

Software

Java 1.4 or greater: download Sun Java Runtime Environment (JRE) from *http://java.sun.com/javase/downloads/index.jsp*

Additional programs for viewing networks (optional), e.g., MITAB (flat file), XML PSI1.0, XML PSI 2.5, Osprey

1. Carry out Basic Protocol 1, steps 1 to 5. Then click the MINT viewer icon, below the frame in which all the interaction partners of the selected protein (e.g., Lck) are listed, to load the MINT viewer applet and to obtain a graphic display of the interaction network centered on Lck.

   This may take a few seconds. Each protein is represented as an oval whose area is proportional to its molecular weight.

   Protein interactions are represented by lines (edges) connecting the proteins (nodes). Proteins with OMIM entries are now highlighted in red.

2. Both nodes and edges are interactive; click on them to obtain the display of additional information about the partner proteins and their interactions or to obtain the expansion of the displayed network.

   For instance, clicking on the number reported on the edge connecting any two proteins displays, in the left frame (see Fig. 8.5.7), the description of the experimental evidence for this interaction also shown in Figure 8.5.5.

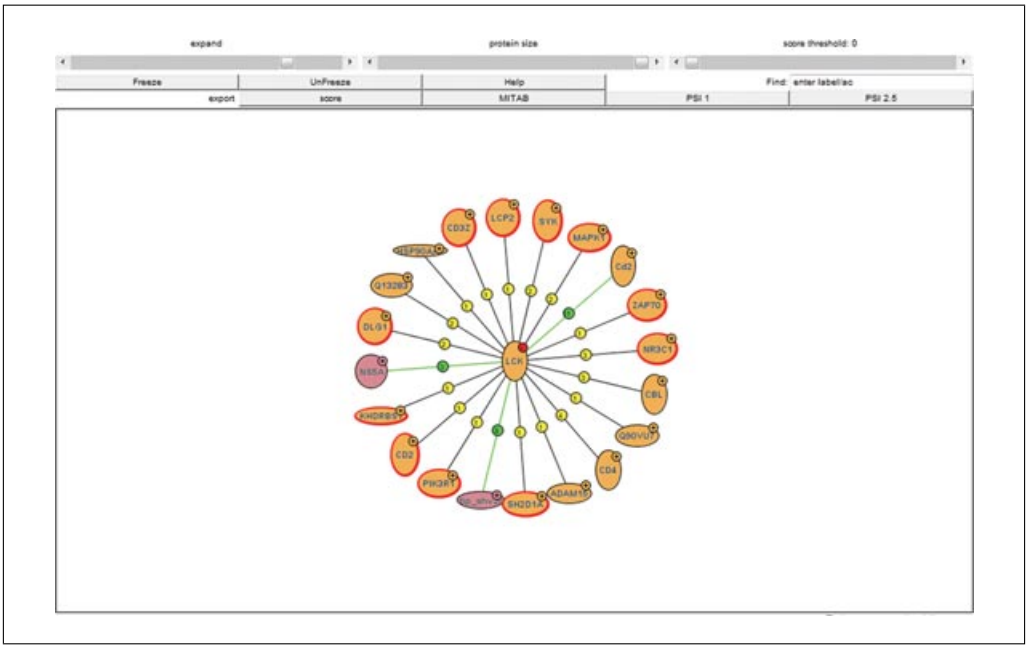


Figure 8.5.6 Graphic representation of the Lck interaction network obtained by clicking the MINT Viewer icon.

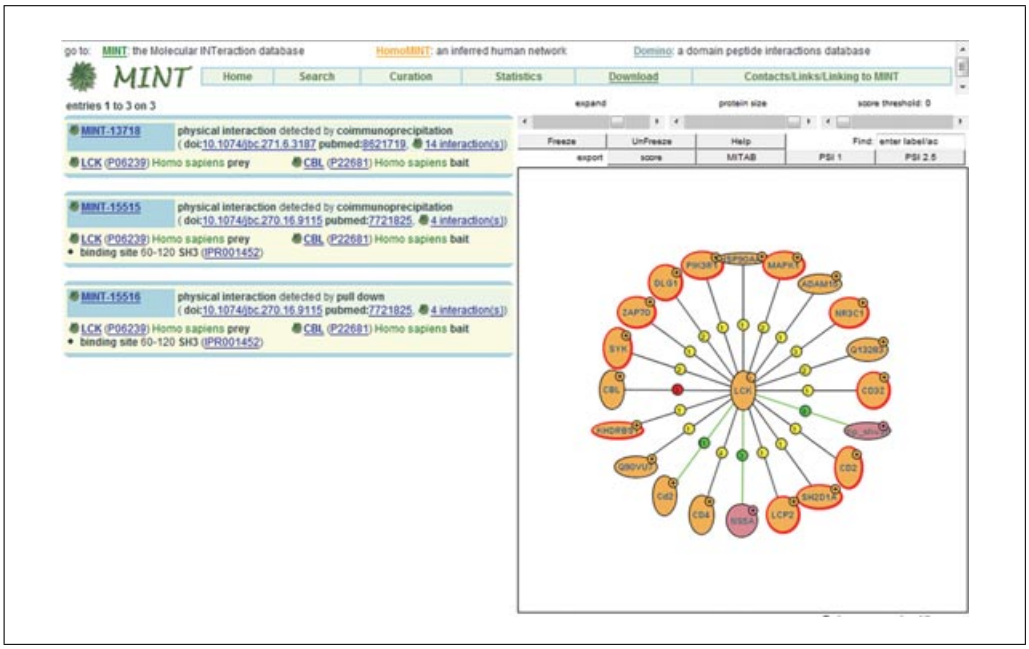


Figure 8.5.7 Detailed information supporting the interaction between Lck and MAPK1. This was obtained by clicking on the circle sitting on the connecting edge in the previous figure (see Fig. 8.5.6 and text).

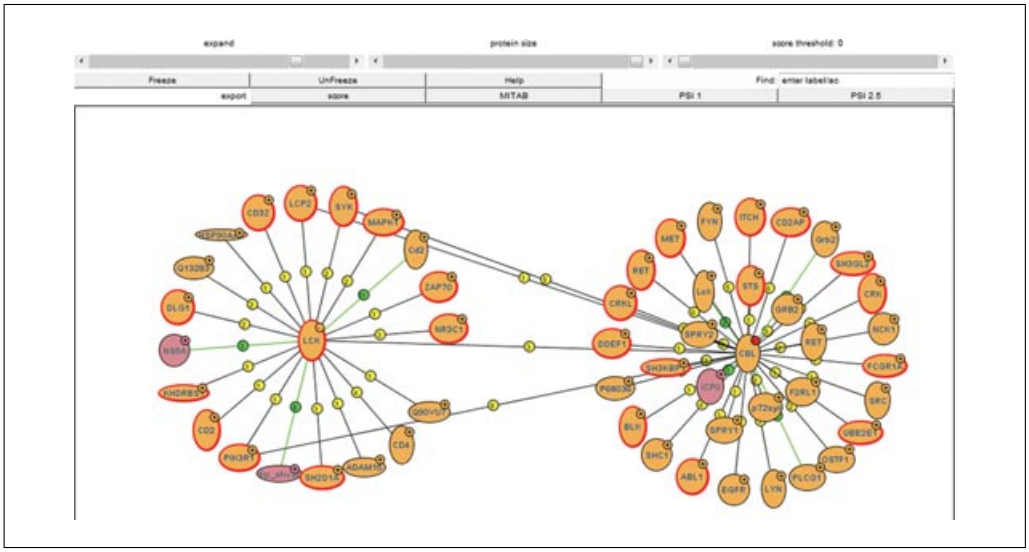


Figure 8.5.8 Extension of the Lck interaction network, obtained by clicking on the (+) symbol representing the Cbl protein (see Fig. 8.5.7 and text).

3. Click on the (+) symbol on each protein to expand the interaction network to include all the interactions for the selected protein (Fig. 8.5.8). Each protein displayed in the viewer frame moves as if it were held by springs connected to the partner proteins in the network.

4. Modulate the tension of the springs by moving a scroll bar (just above the viewer frame) named "expand". This will allow varying the distance between the interaction pairs. Freeze or unfreeze the movement of any protein by clicking on it.

   The movement of the proteins is only for convenience of display and is not correlated to any specific feature of the interaction.

5. Alternatively, freeze the whole network by pressing the Freeze All button. Once the network is still, it is possible to adjust the position of any protein by clicking and dragging in order to obtain a clearer display.

6. Change the protein size by using the scroll bar (just above the viewer frame) named "protein size", and scroll the whole network view by clicking the background and dragging the mouse.

7. Use the scroll bar (above the viewer frame) named "score threshold" to set a confidence threshold. Each interaction in the MINT database is scored for confidence based on the following MINT scoring system (see Chatr-aryamontri et al., 2008).

   Cumulative evidence. We empirically define this as the sum of all the supporting evidence weighted by coefficients that reflects the user confidence on the specific approach, and thus cumulative evidence is user defined in the following equation:

   $$x = \sum_i h_i d_i e_i + n/10$$

   d reflects the size of the experiment. Experiments are defined as large scale if the article reporting them describes more than 50 interactions; otherwise they are defined as small scale. This coefficient is set to 1 for small scale and to 0.5 for large scale experiments.

   e depends on the type of experiment supporting the interaction and emphasizes evidence of direct interaction, where $e = 1$. With respect to experimental support that does not provide unequivocal evidence of direct interaction (e.g., co-immunoprecipitation, pull down), $e = 0.5$.

   h HomoMINT (the human network) is completed with interactions inferred from experiments with ortholog proteins in model organisms. Such evidence is weighted by the Inparanoid confidence value, which is related to sequence homology.

   n is the number of different publications supporting the interaction.

   > *No single experimental approach has maximum sensitivity (no false negative) and specificity (no false positive); thus, confidence can only be built on the integration of orthogonal experimental evidence. In the scoring system currently implemented in MINT we aimed at scoring high the evidence supporting direct interaction.*

   Score. The score is calculated as a function of the cumulative evidence (x) according to the empirical formula:

   $$S = 1 - a^{-a}$$

   a determines the initial slope of the curve and is chosen ($a = 1.4$) so that the function has a suitable dynamic range and only well supported interactions obtain a value close to 1.

   Interactions that have a confidence score lower than a set threshold will be deleted from the viewer (Fig. 8.5.9).

8. Alternatively, download and view the network displayed in the viewer in various file formats: MITAB (flat file), XML PSI1.0, XML PSI 2.5, and Osprey.

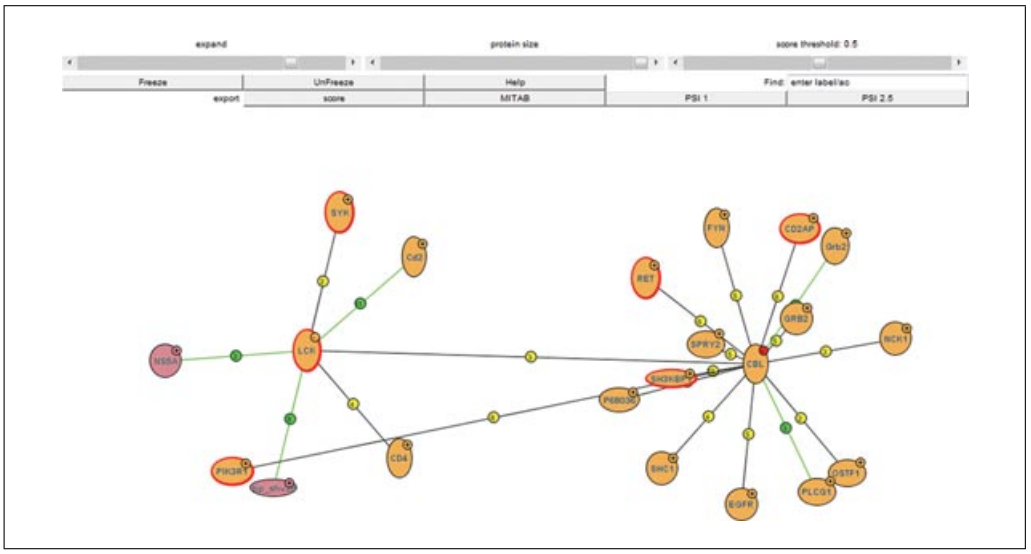


Figure 8.5.9 The Lck interaction network with interactions having a confidence score lower than a set threshold deleted from the viewer (see Fig. 8.5.8 and text).

BASIC PROTOCOL 2

SUBMITTING INTERACTION DATA

MINT entries are currently curated by a small team of expert and specifically trained curators. However, all scientists are encouraged to submit protein interaction information in their own fields of interest. Any scientist considering submitting interaction data is advised to contact the MINT curation team (*curation@mint.bio.uniroma2.it*). The submitter will be given a preformatted Microsoft Excel spreadsheet file (*http://imex.sourceforge.net/doc/imex-curationManual.doc*), accompanied with explicit instructions on how to complete it. This file was developed by the IMEx consortium and facilitates standardized representation of the minimal information required to describe a protein-protein interaction.

Necessary Resources

Hardware

Computer with Internet connection

Software

Internet browser (e.g., Firefox, *http://www.mozilla.org/firefox*; Safari, *www.apple.com/safari*; *or* Internet Explorer, *http://www.microsoft.com*)
Microsoft Excel (e.g., see *http://office.microsoft.com/en-us/excel/default.aspx*)

Files

Microsoft Excel spreadsheet file, preformatted: obtain from *http://imex.sourceforge.net/doc/imex-curationManual.doc*

1. Download the Excel spreadsheet file.
2. Open the file on your computer.

   You need to have Microsoft Excel installed.
3. Access the Manuscript Information page and fill in the fields. This page contains fields for the submitter's contact information, which will be included in the XML file (Fig. 8.5.10).
4. To report the interaction open the Interaction submission page and fill in the fields (Fig. 8.5.11). The filling of the `Database`, `Experimental_role`,

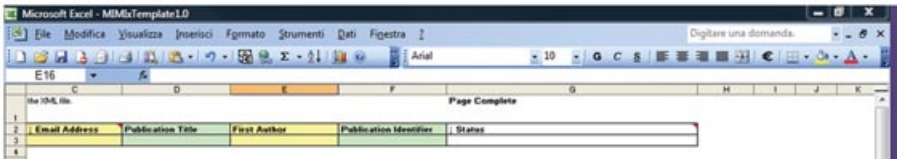


Figure 8.5.10 Manuscript information page. Excel spreadsheet page downloaded from *http://imex.sourceforge.net/doc/imex-curationManual.doc* and used to provide contact and bibliographical information when submitting protein interaction data to the curators of MINT.

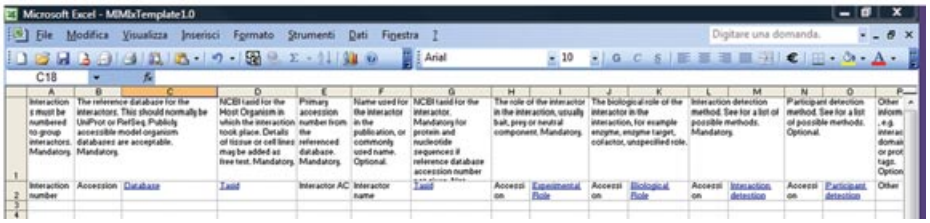


Figure 8.5.11 Interaction submission form. Excel spreadsheet page downloaded from *http://imex.sourceforge.net/doc/imex-curationManual.doc* and used to submit protein interaction information to the curators of MINT.

Table 8.5.1 Definitions of Fields in the Interaction Submission Form

Field	Description
Interaction number	Two interactors with the same interaction number are engaged in the same interaction
Database	Database to which the protein identifier is linked
Taxid	NCBI taxonomy ID for host organism in which the interaction takes place; enter −1 if the interaction occurs in vitro
Interactor AC	Primary accession number from the protein database
Interactor name	Name used in the publication for the interactor (e.g., Lck p56-LCK)
Taxid:	NCBI taxonomy ID for the interactor (e.g., 9606 for a human interactor)
Experimental role	The role of the interactor in the experiment; usually bait, prey, or neutral component
Biological role	Role of the interactor in the interaction, usually unspecified, but other terms (e.g., enzyme, enzyme target, or electron donor) can be used
Interaction detection	Interaction detection method (e.g., two-hybrid, CoIP)
Participant detection	Participant detection method; see Table 8.5.3 for a list of possible methods
Other	Other information (e.g., interacting domains, protein tags)

`Biological_role`, `Interaction_detection`, and `Participant_detection` fields are facilitated by drop-down menus, which contain suggestions for the most appropriate terms (see Table 8.5.1).

5. Send the file to the MINT curation team (*curation@mint.bio.uniroma2.it*) for quality control review. Once accepted, the MINT team will release the interaction(s) to the public database.

GUIDELINES FOR UNDERSTANDING RESULTS

Each MINT entry consists of a series of fields that describe the interacting partners and the type of interaction between them. In MINT, the experimental and biological roles of each protein are annotated in different fields. For example, most experimental techniques are asymmetric, and baits and preys can be easily distinguished. For instance, in a co-immunoprecipitation experiment the protein that is affinity purified with the corresponding specific antibody is the bait, while the proteins that co-purify are the preys. If the interaction involves an enzymatic modification, then biological roles can be specified, one partner being identified as the enzyme and the other as the substrate. Each of the two partners is described by the fields explained in Table 8.5.2. All of the controlled vocabularies (see Table 8.5.3) mentioned in this section are continuously updated and revised to make them more interoperable with the equivalent controlled vocabularies used by other interaction databases.

Table 8.5.2 Fields Describing the Interaction of Two Proteins in MINT Flat Files[a,b]

Tab-separated fields	Explanation
Protein A information	
ID interactor A (bait)	UniProt and/or RefSeq accession number of protein A
Alias interactor A (bait)	Alternative identifier for interactor A (e.g., the official gene symbol as defined by HUGO); only one alias given
Taxid interactor A (bait)	Organism taxid
Experimental role A (bait)	Experimental role of protein A (should be bait or neutral)
MINT protein group A (taxon)	Taxonomy group assigned by MINT for bait
Protein B information	
ID interactors B (all preys)	UniProt and/or RefSeq accession number of each protein B
Alias(es) interactors B (preys)	Alternative identifier for each interactor B
Taxid(s) interactors B (preys)	Organism taxid of each prey
Experimental role B (preys)	Experimental role of preys (should be prey or neutral)
MINT protein group B (taxon)	Taxonomy group assigned by MINT for preys
Interaction attributes	
Interaction detection method(s)	Method used to demonstrate the interaction
Publication identifier(s)	PubMed identifier (PMID), which permits hyperlinking of the entry to the abstract of the manuscript that reports the experiments supporting the interaction
Interaction type(s)	Type of interactions between partner proteins (e.g., binds, phosphorylates) reported by selecting the appropriate item from a controlled vocabulary (see Table 8.5.3)
Interaction identifier(s)	The MINT identifier assigned to the interaction
BioSource_taxid	The taxid of the organism where the interaction takes place
Negative	If set to true, the interaction has been shown not to occur
MINT confidence score	MINT confidence score as previously defined

[a]By convention, partner A is the bait in the binding experiment and partner B is the prey. If the interaction implicates an enzymatic modification, partner A is the enzyme and partner B is the substrate.
[b]Use the controlled vocabulary listed in Table 8.5.3.

Table 8.5.3 Experimental Methods Controlled Vocabulary

biochemical
affinity technologies
saturation binding
filter binding
far western blotting
enzyme linked immunosorbent assay
competition binding
display technologies
bacterial display
phage display
filamentous phage display
lambda phage display
t7 phage display ribosome display
yeast display
electrophoretic mobility shift assay
chromatography technologies
affinity chromatography technologies
coimmunoprecipitation
anti bait coimmunoprecipitation
anti tag coimmunoprecipitation
pull down
gst pull down
his pull down
tandem affinity purification
ion exchange chromatography
molecular sieving
reverse phase chromatography
array technologies
peptide array
protein in situ array
protein array
proteinchip on a surface-enhanced laser desorption/ionization
comigration in non denaturing gel electrophoresis
blue native page
cosedimentation
cosedimentation through density gradients
cosedimentation in solution
cross-linking studies
enzymatic studies
deacetylase assay
gtpase assay
methyltransferase assay
phosphatase assay
protease assay
protein kinase assay
footprinting
biophysical
circular dichroism
electron resonance
electron nuclear double resonance
electron paramagnetic resonance
fluorescence technologies
bioluminescence resonance energy transfer

continued

Table 8.5.3 Experimental Methods Controlled Vocabulary, *continued*

fluorescence correlation spectroscopy
classical fluorescence spectroscopy
fluorescence polarization spectroscopy
fluorescent resonance energy transfer
homogeneous time resolved fluorescence
fluorescence-activated cell sorting
bacterial display
yeast display
isothermal titration calorimetry
light scattering
molecular sieving
mass spectrometry studies of complexes
nuclear magnetic resonance
scintillation proximity assay
surface plasmon resonance
x-ray crystallography
protein complementation assay
cytoplasmic complementation assay
membrane bound complementation assay
transcriptional complementation assay
lex-a dimerization assay
two hybrid
two hybrid pooling approach
two hybrid array
protein tri hybrid
imaging techniques
electron microscopy
light microscopy
fluorescence microscopy
colocalization by fluorescent probes cloning
colocalization by immunostaining
bimolecular fluorescence complementation

COMMENTARY

Background Information

Updating MINT. Each new entry is stored in a provisional table and undergoes further automatic and manual quality control checks before release to the stable searchable version of MINT.

Downloading MINT. MINT files are available and can be obtained by clicking the relevant link in the MINT homepage. Academic and commercial users can freely use the data for their research. MINT releases its dataset in different formats:

PSI-MI XML 1 and 2.5: XML files supporting the Protein Standard Initiative – Molecular Interaction format;

MITAB: PSI-MI tab-delimited format, where complexes are exploded into binary interactions;

MINT flat file: simple tab-delimited format, where complexes are represented on a single line.

We estimate that more than 30,000 articles in the scientific literature describe protein interactions. At the time of writing, only the interactions reported in more than 3600 of these articles have been processed by MINT curators. As a consequence, MINT is largely incomplete—a problem common to all interaction databases. In the near future the IMEx consortium will begin to exchange database records so that a single search at a single database will retrieve all data curated by IMEx members. In the meantime, users are advised to perform separate searches of the databases. Users with expertise in any specific field are encouraged to report missing interactions or

errors in any of the entries. As described in Basic Protocol 2, anyone may submit a new entry under the supervision of the curation team.

Critical Parameters and Troubleshooting

The curators aim to report, as accurately as possible, the information reported in a research article. It is clear, however, that aside from curation errors, a MINT entry might not contain all the experimental details important in understanding the biological relevance of any given interaction (see Table 8.5.2). Thus, the user is advised to consult the original literature before drawing any conclusions.

Literature Cited

Chatr-aryamontri, A., Ceol, A., Licata, L., and Cesareni, G. 2008. Protein interactions: Integration leads to belief. TiBS. In press.

Chatr-aryamontri, A., Ceol, A., Palazzi, L.M., Nardelli, G., Schneider, M.V., Castagnoli, L., and Cesareni, G. 2007. MINT: The Molecular INTeraction database. *Nucleic Acids Res.* 35:D572-D574.

Guldener, U., Munsterkotter, M., Oesterheld, M., Pagel, P., Ruepp, A., Mewes, H.W., and Stumpflen, V. 2006. MPact: The MIPS protein interaction resource on yeast. *Nucleic Acids Res.* 34:D436-D441.

Kerrien, S., Alam-Faruque, Y., Aranda, B., Bancarz, I., Bridge, A., Derow, C., Dimmer, E., Feuermann, M., Friedrichsen, A., Huntley, R., Kohler, C., Khadake, J., Leroy, C., Liban, A., Lieftink, C., Montecchi-Palazzi, L., Orchard, S., Risse, J., Robbe, K., Roechert, B., Thorneycroft, D., Zhang, Y., Apweiler, R., and Hermjakob, H. 2007. IntAct—Open source resource for molecular interaction data. *Nucleic Acids Res.* 35:D561-D565.

Mishra, G.R., Suresh, M., Kumaran, K., Kannabiran, N., Suresh, S., Bala, P., Shivakumar, K., Anuradha, N., Reddy, R., Raghavan, T.M., Menon, S., Hanumanthu, G., Gupta, M., Upendran, S., Gupta, S., Mahesh, M., Jacob, B., Mathew, P., Chatterjee, P., Arun, K.S., Sharma, S., Chandrika, K.N., Deshpande, N., Palvankar, K., Raghavnath, R., Krishnakanth, R., Karathia, H., Rekha, B., Nayak, R., Vishnupriya, G., Kumar, H.G., Nagini, M., Kumar, G.S., Jose, R., Deepthi, P., Mohan, S.S., Gandhi, T.K., Harsha, H.C., Deshpande, K.S., Sarker, M., Prasad, T.S., and Pandey, A. 2006. Human protein reference database—2006 update. *Nucleic Acids Res.* 34:D411-D414.

Salwinski, L., Miller, C.S., Smith, A.J., Pettit, F.K., Bowie, J.U., and Eisenberg, D. 2004. The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res.* 32:D449-D451.

Stark, C., Breitkreutz, B.J., Reguly, T., Boucher, L., Breitkreutz, A., and Tyers, M. 2006. BioGRID: A general repository for interaction datasets. *Nucleic Acids Res.* 34:D535-D539.

Internet Resources

http://mint.bio.uniroma2.it/mint

http://dip.doe-mbi.ucla.edu

http://www.ebi.ac.uk/intact

The Molecular Interactions Database (MINT), Database of Interacting Proteins (DIP), and IntAct Web sites. These are founders and active members of the IMEx consortium, which shares curation efforts and exchanges completed records on molecular interaction data.

http://mips.gsf.de

http://www.thebiogrid.org

http://www.hprd.org

The MIPS, BioGRID, and HPRD Web sites for other well established protein interaction databases currently accessible on the Web.

Identifying Functional Sites Based on Prediction of Charged Group Behavior

The sequences of the human genome and the genomes of about one thousand species are now known. Structural genomics projects are now engaged in finding the three-dimensional structures of hundreds, and eventually thousands, of gene products. The next task on the horizon is to discover the function of these protein structures. This unit describes the implementation and interpretation of THEMATICS (Theoretical Microscopic Titration Curves), a simple computational procedure for the identification of active sites in proteins from the three-dimensional structure alone. THEMATICS identifies reactive sites, including residues involved in catalysis and recognition.

THEMATICS ANALYSIS USING THE UHBD PACKAGE

BASIC PROTOCOL

To perform a THEMATICS analysis, the most computationally intensive step is the calculation of the electric field function. There are multiple programs available that can be used to perform this step. In the present unit, the detailed steps for executing a THEMATICS analysis will be illustrated using the UHBD (University of Houston Brownian Dynamics; Madura et al., 1995) package. However, any number of programs can be used to solve the Poisson-Boltzmann equations for the electric field function of a protein structure and then to compute the predicted titration curves. The UHBD program has been chosen for the present purposes because it is freely available, the source codes are downloadable, and a detailed manual is available on the Web site (*http://adrik.bchs.uh.edu/uhbd/*). Any of the other programs for the calculation of protein electric fields will involve substantially similar steps.

The author of this unit and her research group wish to make available a unified THEMATICS program that accepts the coordinates of a protein as input and predicts information about active-site location as output. However as of this writing, such an integrated, fully automated, user-friendly code for the prediction of functional information from protein structure is not yet available. At the present time, a THEMATICS analysis consists of a series of calculations, described below.

Because a user-friendly, unified THEMATICS code is not yet available, the author's research group is currently running the calculations on structures submitted by outside investigators. The group may be reached at *mjo@neu.edu*. A number of THEMATICS calculations have been performed upon request by the author and associates, including structures submitted on a confidential basis prior to publication, and this practice will continue. The steps outlined below are for investigators who wish to run the calculations themselves.

While there may be some differences among the available programs in system requirements and in the necessary input files, the general requirements are likely to be similar to those described under Necessary Resources, below.

Necessary Resources

Hardware

The calculations do not require any specialized hardware and have been run on a variety of systems, including small ones, e.g., PC or Macintosh (Linux or Unix-based; both desktops and laptops) or SGI workstation, as well as large parallel cluster systems. Most of the author's calculations currently are

Contributed by Mary Jo Ondrechen
Current Protocols in Bioinformatics (2004) 8.6.1-8.6.10

performed on a four-node Debian Linux cluster built from four Dell desktops. Graphics display capability is useful for the interpretation of the results.

Software

There are a number of programs available that solve the Poisson-Boltzmann equations numerically for protein structures. As indicated above, a Unix or Linux operating system is generally required. The programs in the UHBD package are written in FORTRAN 77, so an f77 compiler is necessary. Detailed steps for the installation and compilation of the UHBD program are given in the Web-accessible manual.

For visualization of protein structures, PyMol (*http://pymol.sourceforge.net/*) is easy to use. This program requires that Python (*http://www.python.org/*) also be installed.

Files

A file with the atomic coordinates of the atoms in the protein, typically in PDB format, is required as input. The Poisson-Boltzmann programs require a file of the force field parameters for the 20 amino acids.

Obtain three-dimensional protein structure

Start with the three-dimensional structure of the query protein. Generally this is a PDB file. This file may be obtained from the Protein Data Bank (Berman et al., 2000), determined experimentally by X-ray diffraction or NMR, or generated as a theoretical model structure.

Most of the structures analyzed by the author's group to date have been determined by X-ray diffraction, and nearly all of them have had 3.0 Å resolution or better; NMR structures have also been successfully used. In preliminary work by the author's research group on structures built from comparative modeling, THEMATICS has correctly identified the active sites (Shehadi et al., 2004). It is not yet known precisely how good a structure must be in order to get correct THEMATICS predictions about the important residues in catalysis and recognition. Therefore, it is not clear at this time what the chance is that a THEMATICS prediction is correct for a low-resolution structure, a model structure based on weak homology, or a structure built from threading. It is apparent that a structure that may not be of sufficient quality to predict accurate pK_a's may still be good enough for THEMATICS to find the correct active site. THEMATICS has proven to be highly reliable for experimentally determined structures with reasonably good resolution.

THEMATICS analyses are generally performed on the protein structure alone, with the coordinates of any substrate molecules, cofactors, and solvent molecules removed. To determine accurate pK_a's for active-site residues under conditions that mimic the catalytically active state of the enzyme, one needs to include the substrate molecule, any cofactors, and any tightly bound solvent molecules. However, for purposes of active-site location, the calculation is performed on the protein only. To screen protein structures of unknown function, one generally will not have information about the nature or location of the substrate molecule and other species inside the binding pocket. Therefore, for comparative purposes, THEMATICS analyses are performed on the protein only. Since THEMATICS depends on specific types of *interactions* between ionizable residues and not on the precise values of the pK_a's, one can still find the correct active site even without all of the reactive species present in the structure. If one desires more accurate pK_a's and average values for the charges, one can always repeat the calculations with all of the important species included.

Protein structures that contain metal ions constitute a special case. Metal ions in a protein crystal structure are generally recognizable by their coordination, even if the function of the protein is not known. For metal-containing proteins, the author usually performs

the calculation twice, with and without the metal ions. The predicted titration curves for the residues near the metal-containing sites tend to have similar shape whether the metal ions are present or not. In particular, the presence or absence of the metal ion generally does not affect the classification of a residue as either THEMATICS-positive or -negative. However, the metal ions do shift the titration curves significantly along the pH axis. The predicted pK_a's are several pH units higher if the metal ions are not included. If the metal ions are included, one has a more reasonable model for the system, but the titration curves may be shifted below the standard pH range over which analysis is typically performed.

The structures determined by X-ray crystallography usually do not contain the coordinates of the hydrogen atoms. Some of the programs require that some or all of the missing hydrogen atoms be added into the structure. To accomplish this, go to the instructions on the PDB Web site for adding hydrogen atoms to a PDB structure file (*http://beta.rcsb.org/pdb/pe/explorer/help_hyd.htm*). This PDB page contains links that lead to the WHAT IF Web site. The protein modeling program WHAT IF (*http://www.cmbi.kun.nl/gv/servers/WIWWWI/*; Vriend, 1990) has numerous capabilities, including a feature to add the missing hydrogen atoms to a structure file. These programs are generally based on a specified force field and perform a free-energy minimization on the hydrogen atoms. Some such software is freely available and downloadable, including the programs Complete (*http://mmtsb.scripps.edu/software/mmtsbtoolset.html*) and Reduce (*http://kinemage.biochem.duke.edu/software/softdownphp/downredpro.php*; Word et al., 1999). There are other protein modeling programs that can be used to add the hydrogen atoms, including the commercial programs InsightII (*http://www.accelrys.com/insight/index.html*), BioMedCACheche (*http://www.cachesoftware.com/biomedcache/index.shtml*), GETATOMS (*http://www.softberry.com/berry.phtml?topic=getatoms&group=programs&subgroup=propt*), and SYBYL (*http://www.tripos.com/*).

Calculate the electrical potential function for the protein

Perform a Poisson-Boltzmann calculation to obtain the electrical potential function for the protein structure (e.g., *UNIT 8.4*). Table 8.6.1 lists some of the different programs that perform this calculation.

If using the UHBD program, one must provide some information about one's assumptions regarding the ionization states of each histidine and cysteine residue.

In particular, the imidazole ring of the histidine side chain has two sites that can be protonated, the nitrogen atoms Nδ and Nε. This means that the neutral histidine residue

Table 8.6.1 Programs to Calculate Electrical Potentials and pK_a's for Proteins

Program	URL	Reference
APBS	*http://agave.wustl.edu/apbs/*	Baker et al. (2001)
DelPhi	*http://honiglab.cpmc.columbia.edu* *http://www.accelrys.com/insight/DelPhi_page.html*	Yang et al. (1993); *UNIT 8.4*
MEAD	*http://www.scripps.edu/bashford/*	You and Bashford (1995)
MM_SCP	*http://fulcrum.physbio.mssm.edu/~mehler/text/pka.html*	Mehler and Guarnieri (1999)
pep	*http://www.scripps.edu/case/beroza*	Beroza and Case (1996)
UHBD	*http://adrik.bchs.uh.edu/uhbd/* *http://mccammon.ucsd.edu/uhbd.html*	Madura et al. (1995)
WHAT IF	*http://www.cmbi.kun.nl/whatif/*	Vriend (1990)
Zap	*http://www.eslc.vabiotech.com/zap/*	

has two tautomeric forms. It is necessary to specify which nitrogen atom is assumed to be protonated first. (It is assumed for computational simplicity that one of the two forms of the neutral His dominates.) In the absence of any specific information, one may assume that Nε is protonated first. To save time, one can assume that all of the His residues are in the same tautomeric form. The most accurate pK_a's are obtained by examination of the local environment of both nitrogen atoms of each histidine residue, and then determining which nitrogen is likely to be protonated first. However, one can also assume that all neutral histidine residues are in the same tautomeric form. This faster, simpler assumption may reduce the reliability of the calculated pK_a's but does not appear to reduce the effectiveness of active-site location.

One must also list in the input file the sequence numbers of the cysteine residues that are ionizable. A cysteine residue is considered ionizable if it is not involved in a disulfide bridge. In addition, one must identify the residues that are at the N-terminus and the C-terminus of each subunit.

Calculate and analyze titration curves for ionizable groups

Perform the calculation of the titration curves for each ionizable group in the protein structure. This includes the side chains of each Arg, Asp, Cys, Glu, His, Lys and Tyr, plus the N- and C-termini. In the UHBD package, the program HYBRID (Gilson, 1993; Antosiewicz et al., 1996a) uses a hybrid Monte Carlo procedure to generate the C(pH) curves (mean net charge C as a function of the pH). Analyze the predicted titration curves and identify those that deviate from the typical Henderson-Hasselbalch shape. The C(pH) curves for a typical Henderson-Hasselbalch residue are sigmoidal with a steep negative slope in the region where the pH equals the pK_a. Identify the curves that are nonsigmoidal or that have a significantly less steep slope in the region near the pK_a. The corresponding residues are labeled as "THEMATICS-positive" residues. Some examples are shown in Figures 8.6.1, Figure 8.6.2, and Figure 8.6.3.

Figure 8.6.1 shows the predicted titration curves (mean net charge C as a function of pH) for five tyrosine residues in alanine racemase. Alanine racemase is a pyridoxal phosphate-dependent bacterial enzyme that catalyzes the interconversion of D- and L-alanine. It is

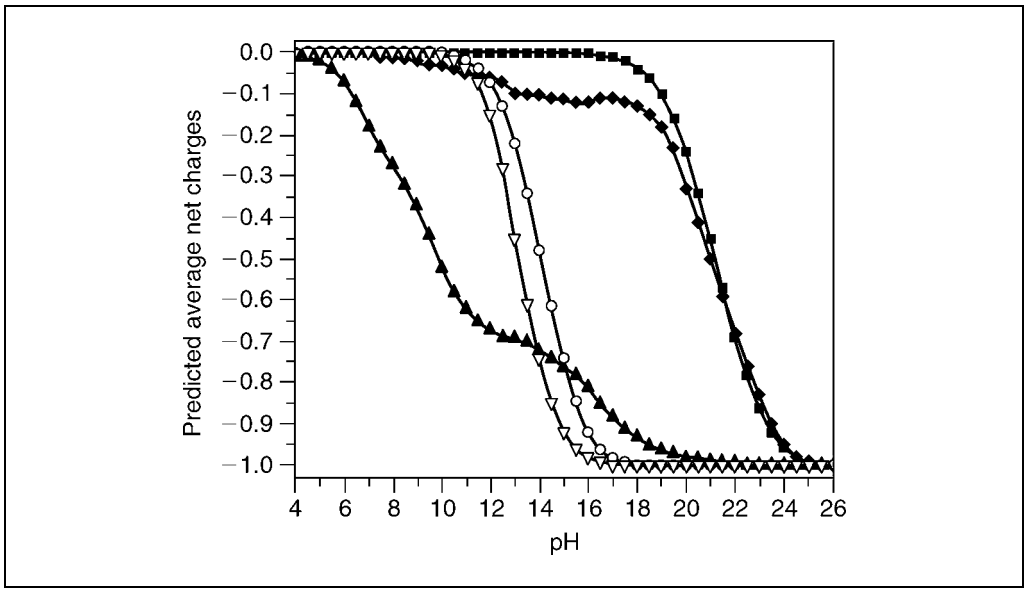


Figure 8.6.1 Alanine racemase tyrosines. Predicted titration curves (ensemble average charge C as a function of the pH) for tyrosine residues Y225 (solid squares), Y239 (hollow circles), Y265 (solid triangles), Y269 (hollow triangles), and Y284 (solid diamonds) in one of the two subunits of the alanine racemase dimer.

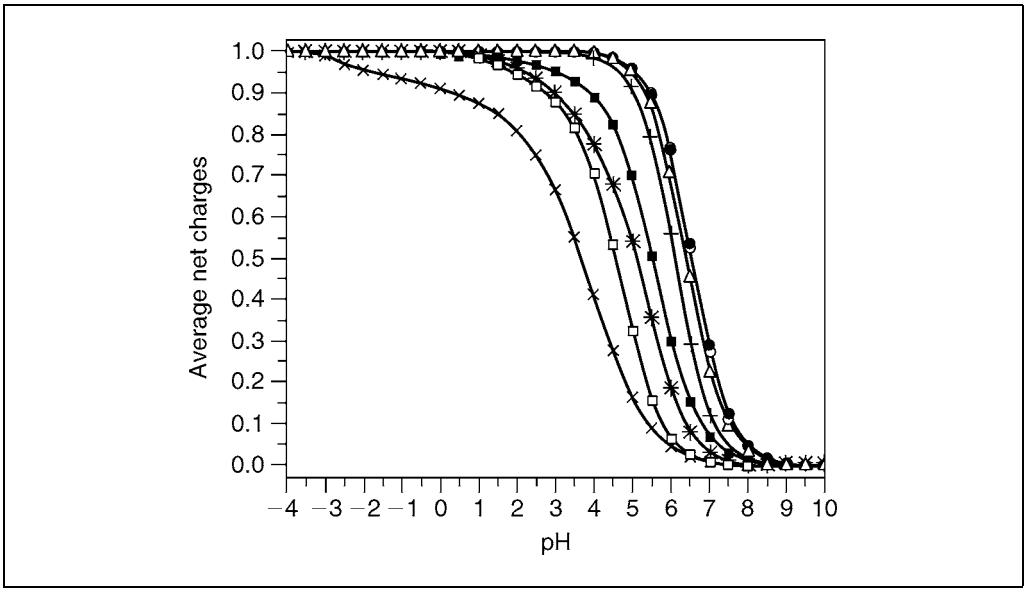


Figure 8.6.2 TIM histidines. Predicted titration curves (ensemble average charge C as a function of the pH) for histidine residues H26 (plus signs), H95 (times signs), H100 (asterisks), H115 (hollow squares), H185 (solid squares), H195 (hollow circles), H224 (solid circles), and H248 (hollow triangles) in one of the two subunits of the triosephosphate isomerase (TIM) dimer.

used in bacterial cell wall construction and is a target for antibiotics. Calculations were performed on the biologically active dimer structure from *Bacillus stearothermophilus* (PDB code 1BD0; Stamper et al., 1998). Figure 8.6.1 depicts the predicted titration curves for the tyrosine residues between 225 and 284 inclusive of one of the two subunits of the dimer. Notice the atypical, nonsigmoidal shapes of residues Y265 and Y284. It has been established by site-directed mutagenesis (Watanabe et al., 1999) and by a competitive inhibitor structure (Stamper et al., 1998) that Y265 (represented by the solid triangles) is one of two catalytic bases that abstracts the α-hydrogen atom from alanine to catalyze racemization. Y284 (represented by the solid diamonds) is also located in the active-site pocket and is a neighbor of the reacting alanine moiety (Stamper et al., 1998).

Figure 8.6.2 shows the predicted titration curves (mean net charge C as a function of pH) for all eight histidine residues in one of the two subunits of the triosephosphate isomerase (TIM) dimer. TIM catalyzes the interconversion of D-glyceraldehyde-3-phosphate (GAP) to dihydroxyacetone phosphate (DHAP). Calculations were performed on the biologically active dimer structure from chicken (PDB code 1TPH; Zhang et al., 1994). C(pH) curves are shown for H26 (plus signs), H95 (times signs), H100 (asterisks), H115 (hollow squares), H185 (solid squares), H195 (hollow circles), H224 (solid circles), and H248 (hollow triangles). Note the elongated, nonsigmoidal shape of the curve for active-site residue H95 (represented by the times signs). Experimental evidence has established that H95 is directly involved in catalysis (Lodi, 1991; Zhang et al., 1994).

Figure 8.6.3 shows the predicted titration curves (mean net charge C as a function of pH) for five of the aspartate residues in 6-hydroxymethyl-7,8-dihydropterin pyrophosphokinase (HPPK), a bacterial phosphate transferase. Calculations were performed on the *E. coli* structure (PDB code 1HKA; Xiao et al., 1999). Curves are shown for D49 (hollow squares), D95 (hollow circles), D97 (solid triangles), D117 (hollow triangles), and D153 (hollow diamonds). Note that the catalytic residue D97 (represented by the solid triangles) has a nonsigmoidal shape. Originally, when very conservative criteria were applied, the active-site residue D95 (represented by the hollow circles) was not classified by the author's research group as a positive residue because of its sigmoidal shape. Since then, more examples of active-site residues that exhibit a shallow negative slope, as opposed to the

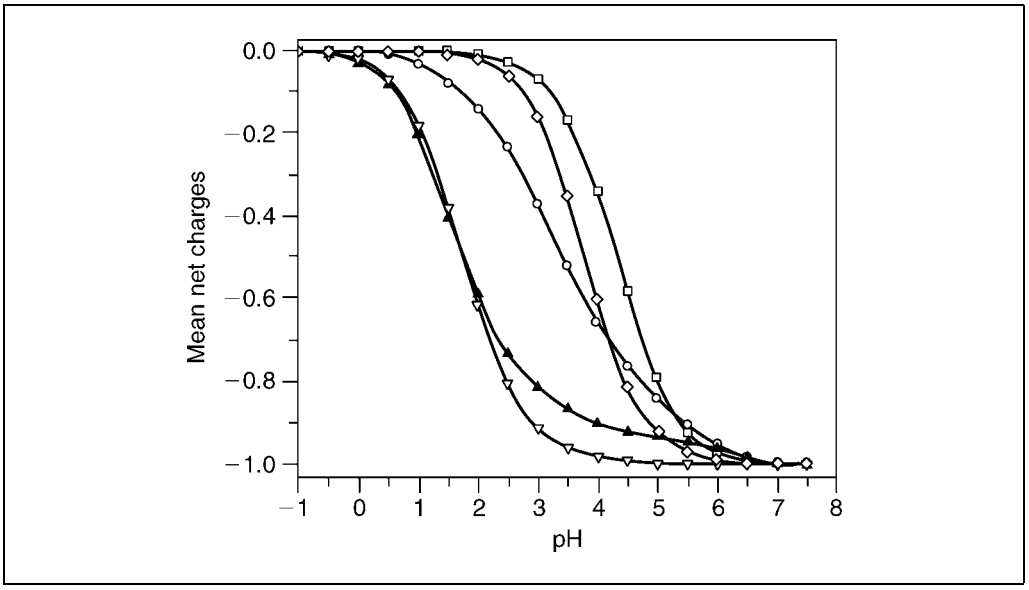


Figure 8.6.3 HPPK aspartates. Predicted titration curves (ensemble average charge C as a function of the pH) for aspartate residues D49 (hollow squares), D95 (hollow circles), D97 (solid triangles), D117 (hollow triangles), and D153 (hollow diamonds) in HPPK.

Table 8.6.2 THEMATICS Results for Some Selected Enzymes[a]

Enzyme	Species	THEMATICS positives
Adenosine kinase	Human	[**D18**, **D300**, **E226**]
Alanine racemase	*Bacillus stearothermophilus*	[**R219**, **C311′**, **K39**, **Y43**, **Y265′**, *Y284′*, **Y354**, *C358*, *Y164*], [R366], [D68]
Apurinic/apyrimidinic endonuclease	Human	[**D210**, **D283**, **D308**, **E96**, **H309**, **Y171**]
Colicin E3	*E. coli*	[**R495**, **R545**, **E517**, **H526**, **Y519**]
Germin	Barley	[**D60**, **E58**, **H88**, **H90**, **H137**], [Nterm, D2, R133]
HIV-1 protease	HIV-1	[**D25**, **D25′**]
HPPK	*E. coli*	[**D95**, **D97**, **H115**]
Δ^5-3-Ketosteroid isomerase	*Pseudomonas putida*	[**D103**, **Y16**, **Y32**, **Y57**]
Papain	Papaya	[**C25**, **H159**], [**K17**, **K174**, **Y186**], [E52], [R96]
Triosephosphate isomerase	Chicken	[**H95**, **E165**, *C126*, *Y164*]

[a]Residues that form a cluster in coordinate space are shown together in square brackets. Known active-site residues are shown in **boldface**. Residues that are nearest neighbors of active-site residues are shown in *italics*. For proteins with two or more subunits, residues that are members of a different subunit are marked with a prime sign (′).

sharp negative slope of the more typical residues, have been observed. Therefore the elongated titration curve of D95 is now classified as positive. However, regardless of which set of criteria is used, one still is able to select enough positive residues to locate the active site.

It is possible to select the perturbed titration curves using mathematical criteria, rather than by visual inspection, and thus automate the analysis. The author's group intends to make this selection program available in the future as a part of an integrated THEMATICS software package.

Table 8.6.2 shows the complete set of THEMATICS-positive residues for ten selected enzymes.

Review protein structure

Identify which of the positive residues are in physical proximity. A set of positive residues in proximity is termed a THEMATICS-positive cluster. If the reactive atom of a residue is within a specified cutoff distance of the reactive atom of any residue in a cluster, then that residue is considered a cluster member. A reasonable value for the cutoff distance is 7 Å. The author's group has tried different values for the cutoff distance, and values in the range 6 Å to 10 Å seem to work fairly well. The cutoff distance simply represents the typical upper bound for the spacing between adjacent reactive residues in an active-site pocket.

Note that the residues that exhibit perturbed titration behavior in Figure 8.6.1, Y265 and Y284 for alanine racemace, belong to a cluster of nine positive residues as shown in Table 8.6.2. This cluster contains the known catalytic residues K39 and Y265′ (Stamper et al., 1998; Watanabe et al., 1999). Similarly the perturbed residue in Figure 8.6.2, H95 of TIM, belongs to a cluster of four residues, as shown in Table 8.6.2. This cluster consists of the two catalytic residues H95 and E165 (Lodi, 1991; Zhang et al., 1994) and also contains two adjacent residues, C126 and Y164. The two perturbed residues of Figure 8.6.3, D95 and D97 of HPPK, belong to a three-member cluster at the known active site (Xiao et al., 1999).

GUIDELINES FOR UNDERSTANDING RESULTS

To verify the efficacy of THEMATICS, the author's research group has tried the method on proteins for which the important active-site residues have been established experimentally by site-directed mutagenesis and/or by structures containing a substrate mimic inhibitor. Information about catalytic residues was obtained from the Catalytic Residue Dataset (CATRES), *http://www.ebi.ac.uk/thornton-srv/databases/CATRES/index.html* (Bartlett et al., 2002), and from PDBsum, *http://www.biochem.ucl.ac.uk/bsm/pdbsum/index.html*, in addition to protein-specific literature articles.

To date, THEMATICS has been tried on about 100 proteins with experimentally characterized active sites, and has correctly found a positive cluster at the known active site for about 90% of known enzymes. The method does give false positives, in that there are residues with perturbed titration curves that are not near any known active site. However, these false positives tend to be isolated in space. Some examples can be seen in Table 8.6.2: R366 and D68 of alanine racemase and E52 and R96 of papain. If the clustering in physical space of two or more positive residues is used as the criterion for active-site prediction, these isolated false positive residues do not diminish the precision of the method.

The examples in Figure 8.6.1, Figure 8.6.2, and Figure 8.6.3, indicate catalytically important residues that are members of THEMATICS-positive clusters. These clusters also contain residues that are important in recognition. For instance, R219 of alanine racemase (see Table 8.6.2) interacts with the pyridine nitrogen atom of the Schiff base intermediate. Recently it has been shown that THEMATICS identifies not only the catalytically important ionizable residues of the serine protease kex2 (Holyoak et al., 2003), but also the specificity determinants of this highly specific protease (Ringe et al., 2004).

In certain cases, THEMATICS finds more than one positive cluster—e.g., for papain, the known cysteine protease active site corresponds to the cluster [**C25, H159**]. However, THEMATICS also finds a second cluster, [**K17, K174, Y186**]. The function of this second cluster is not known, but it is highly conserved and appears to be perfectly conserved across a spectrum of plant proteases. The high conservation of THEMATICS-positive clusters lends support to the assertion that they are functionally important. In cases where

more than one cluster is found, THEMATICS has narrowed down the location of the possible active site(s).

COMMENTARY

Background Information

Introduction to THEMATICS

THEMATICS stands for Theoretical Microscopic Titration Curves. This method exploits the predicted titration properties of the ionizable residues in a protein structure in order to identify reactive sites within that structure. THEMATICS is based on well established Poisson-Boltzmann methods (Warwicker and Watson, 1982; Bashford and Karplus, 1991; Gilson, 1993; Yang et al., 1993; Karshikoff, 1995; Madura et al., 1995; Antosiewicz et al., 1996a; Antosiewicz et al., 1996b; Alexov and Gunner, 1997; Mehler and Guarnieri, 1999; Nielsen et al., 1999) for determining the electrostatic potential function of a protein structure. The potential function is used to calculate the mean net charge C as a function of the pH (a titration curve) for each titratable residue in the structure. Anomalies in the predicted titration properties of a small number of the residues in a protein have been reported previously, and arise from interactions between ionizable groups (Bashford and Gerwert, 1992; Sampogna and Honig, 1994; Beroza et al., 1995; Carlson et al., 1999). The author and colleagues have now established that a cluster of such anomalous residues in physical proximity is a reliable predictor of active-site location (Ondrechen et al., 2001; Ondrechen, 2002; Shehadi et al., 2002). Thus, there is a way to identify the active site of a protein, even if the sequence and the structure bear no resemblance to those of any previously characterized protein. Active-site identification constitutes an important first step in the determination of the function of a protein. The only input required is the three-dimensional structure.

The first working hypothesis is that perturbed titration behavior helps to promote catalysis and reversible recognition. A catalytic acid or base in the active site must regenerate itself in each turnover cycle, and therefore protonation must be reversible. If a charged residue is involved in recognition of a substrate group of opposite charge, the conversion of that residue to its neutral form will enable release of the reactive molecule. Thus, charged residues involved in reversible recognition likewise have an advantage if they can protonate and deprotonate reversibly. A residue that obeys the Henderson-Hasselbalch equation has a narrow pH range where it exists in both protonated and deprotonated forms. A residue with an elongated titration curve has an increased range of conditions over which the ionizable group protonates reversibly.

The second working hypothesis is that the perturbed titration curves arise simply from the polyprotic nature of proteins. The Henderson-Hasselbalch equation applies to monoprotic acids in the absence of any pH-dependent electric field. The number of ionizable groups in a protein structure (roughly one-third of the total residues have ionizable side chains) leads to large numbers of interactions between ionizable groups, and hence some of the residues have perturbed titration curves.

It is not certain at this time what is the meaning of the residues in positive clusters that are nearest neighbors of reactive residues, for instance Y164 of TIM. These "second shell" residues are shown in italics in Table 8.6.2, and, like the catalytically active residues, they tend to be highly conserved. It has not yet been determined whether they actually play some role in the catalytic process, or whether they simply happen to be subjected to the same pH-dependent electric field as the catalytically important residues, and thus exhibit anomalous titration behavior. Some site-directed mutagenesis experiments may help to clarify this.

Also, it is not known at this time whether there is any significance to the isolated false-positive residues. They could arise fortuitously from the polyprotic protein structure or they could be markers of some kind of reactivity.

THEMATICS-positive clusters tend to be subsets of residues identified as highly conserved by sequence-comparison, evolutionary trace, and maximum-likelihood methods (Lichtarge et al., 1996; Sjolander, 1998; Lichtarge and Sowa, 2002; Pupko et al., 2002; Glaser et al., 2003; Yao et al., 2003). THEMATICS is unique among the protein-function predictive methods in that it identifies specific locations in space where chemical reactivity and recognition are likely to occur.

Acknowledgements

The author thanks Dr. Leonel Murga and Ms. Ying Wei for their assistance. This work was supported by the National Science Foundation under Grant MCB-0135303, and by the Institute for Complex Scientific Software.

Literature Cited

Alexov, E.G. and Gunner, M.R. 1997. Incorporating protein conformational flexibility into the calculation of pH-dependent protein properties. *Biophys. J.* 72:2075-2093.

Antosiewicz, J., Briggs, J.M., Elcock, A.H., Gilson, M.K., and McCammon, J.A. 1996a. Computing the ionization states of proteins with a detailed charge model. *J. Comp. Chem.* 17:1633-1644.

Antosiewicz, J., McCammon, J.A., and Gilson, M.K. 1996b. The determinants of pKa's in proteins. *Biochemistry* 35:7819-7833.

Baker, N.A., Sept, D., Joseph, S., Holst, M.J., and McCammon, J.A. 2001. Electrostatics of nanosystems: Application to microtubules and the ribosome. *Proc. Natl. Acad. Sci. U.S.A.* 98:10037-10041.

Bartlett, G.J., Porter, C.T., Borkakoti, N., and Thornton, J.M. 2002. Analysis of catalytic residues in enzyme active sites. *J. Mol. Biol.* 324:105-121.

Bashford, D. and Gerwert, K. 1992. Electrostatic calculations of the pK_a values of ionizable groups in bacteriorhodopsin. *J. Mol. Biol.* 224:473-486.

Bashford, D. and Karplus, M. 1991. Multiple-site titration curves of proteins: An analysis of exact and approximate methods for their calculation. *J. Phys. Chem.* 95:9556-9561.

Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., and Bourne, P.E. 2000. The protein data bank. *Nucleic Acids Res.* 28:235-242.

Beroza, P. and Case, D.A. 1996. Including side chain flexibility in continuum electrostatic calculations of protein titration. *J. Phys. Chem.* 100:20156-20163.

Beroza, P., Fredkin, D.R., Okamura, M.Y., and Feher, G. 1995. Electrostatic calculations of amino acid titration and electron transfer, Q-AQB-->QAQ-B, in the reaction center. *Biophys. J.* 68:2233-2250.

Carlson, H.A., Briggs, J.M., and McCammon, J.A. 1999. Calculation of the pKa values for the ligands and side chains of *Escherichia coli* D-alanine:D-alanine ligase. *J. Med. Chem.* 42:109-117.

Gilson, M.K. 1993. Multiple-site titration and molecular modeling: Two rapid methods for computing energies and forces for ionizable groups in proteins. *Proteins* 15:266-282.

Glaser, F., Pupko, T., Paz, I., Bell, R.E., Bechor-Shental, D., Martz, E., and Ben-Tal, N. 2003. ConSurf: Identification of functional regions in proteins by surface-mapping of phylogenetic information. *Bioinformatics* 19:163-164.

Holyoak, T., Wilson, M.A., Fenn, T.D., Kettner, C.A., Petsko, G.A., Fuller, R.S., and Ringe, D. 2003. 2.4 Å resolution crystal structure of the prototypical hormone-processing protease kex2 in complex with an ala-lys-arg boronic acid inhibitor. *Biochemistry* 42:6709-6718.

Karshikoff, A. 1995. A simple algorithm for the calculation of multiple site titration curves. *Protein Eng.* 8:243-248.

Lichtarge, O. and Sowa, M.E. 2002. Evolutionary predictions of binding surfaces and interactions. *Curr. Opin. Struct. Biol.* 12:21-27.

Lichtarge, O., Bourne, H.R., and Cohen, F.E. 1996. An evolutionary trace method defines binding surfaces common to protein families. *J. Mol. Biol.* 257:342-358.

Lodi, P.J. and Knowles, J.R. 1991. Neutral imidazole is the electrophile in the reaction catalyzed by triosephosphate isomerase: Structural origins and catalytic implications. *Biochemistry* 30:6948-6956.

Madura, J.D., Briggs, J.M., Wade, R.C., Davis, M.E., Luty, B.A., Ilin, A., Antosiewicz, J., Gilson, M.K., Bagheri, B., Scott, L.R., and McCammon, J.A. 1995. Electrostatics and diffusion of molecules in solution: Simulations with the University of Houston Brownian Dynamics program. *Comput. Phys. Commun.* 91:57-95.

Mehler, E.L. and Guarnieri, F. 1999. A self-consistent, microenvironment modulated screened Coulomb potential approximation to calculate pH-dependent electrostatic effects in proteins. *Biophys. J.* 77:3-22.

Nielsen, J.E., Andersen, K.V., Honig, B., Hooft, R.W.W., Klebe, G., Vriend, G., and Wade, R.C. 1999. Improving macromolecular electrostatics calculations. *Protein Eng.* 12:657-662.

Ondrechen, M.J. 2002. THEMATICS as a tool for functional genomics. *Genome Inform.* 13:563-564.

Ondrechen, M.J., Clifton, J.G., and Ringe, D. 2001. THEMATICS: A simple computational predictor of enzyme function from structure. *Proc. Natl. Acad. Sci. U.S.A.* 98:12473-12478.

Pupko, T., Bell, R.E., Mayrose, I., Glaser, F., and Ben-Tal, N. 2002. Rate4Site: An algorithmic tool for the identification of functional regions in proteins by surface mapping of evolutionary determinants within their homologues. *Bioinformatics* 18:S71-S77.

Ringe, D., Wei, Y., Boino, K.R., and Ondrechen, M.J. 2004. Protein structure to function: Insights from computation. *Cell. Mol. Life Sci.* 61:387-392.

Sampogna, R.V. and Honig, B. 1994. Environmental effects on the protonation states of active site residues in bacteriorhodopsin. *Biophys. J.* 66:1341-1352.

Shehadi, I.A., Yang, H., and Ondrechen, M.J. 2002. Future directions in protein function prediction. *Mol. Biol. Rep.* 29:329-335.

Shehadi, I.A., Uzun, A., Murga, L.F., Ilyin, V., and Ondrechen, M.J. 2004. THEMATICS is effective for active site prediction in comparative model structures. *In* Proceeding of the Second Asia-Pacific Bioinformatics Conference (APBC2004), Dunedin, New Zealand, vol. 29 (Y.P.P. Chen ed.), pp. 209-215.

Sjolander, K. 1998. Phylogenetic inference in protein superfamilies: Analysis of SH2 domains. *In* Proceedings of the Conference Intelligent Systems for Molecular Biology 1998, vol. 6, pp. 165-74.

Stamper, G.F., Morollo, A.A., Ringe, D., and Stamper, C.G. 1998. Reaction of alanine racemase with 1-aminoethylphosphonic acid forms a stable external aldimine. *Biochemistry* 37:10438-10445.

Vriend, G. 1990. WHAT IF: A molecular modeling and drug design program. *J. Mol. Graph.* 8:52-56.

Warwicker, J. and Watson, H.C. 1982. Calculation of the electric potential in the active site cleft due to alpha-helix dipoles. *J. Mol. Biol.* 157:671-679.

Watanabe, A., Yoshimura, T., Mikami, B., and Esaki, N. 1999. Tyrosine 265 of alanine racemase serves as a base abstracting α-hydrogen from L-alanine: The counterpart residue to lysine 39 specific to D-alanine. *J. Biochem.* 126:781-786.

Word, J.M., Lovell, S.C., Richardson, J.S., and Richardson, D.C. 1999. Asparagine and glutamine: Using hydrogen atom contacts in the choice of sidechain amide orientation. *J. Mol. Biol.* 285:1733-1745.

Xiao, B., Shi, G., Chen, X., Yan, H., and Ji, X. 1999. Crystal structure of 6-hydroxymethyl-7,8-dihydropterin pyrophosphokinase, a potential target for the development of novel antimicrobial agents. *Structure Fold. Des.* 7:489-496.

Yang, A.S., Gunner, M.R., Sampogna, R., Sharp, K., and Honig, B. 1993. On the calculation of pKas in proteins. *Proteins* 15:252-265.

Yao, H., Kristensen, D.M., Mihalek, I., Sowa, M.E., Shaw, C., Kimmel, M., Kavraki, L., and Lichtarge, O. 2003. An accurate, sensitive, and scalable method to identify functional sites in proteins. *J. Mol. Biol.* 326:255-261.

You, T. and Bashford, D. 1995. Conformation and hydrogen ion titration of proteins: A continuum electrostatic model with conformational flexibility. *Biophys. J.* 69:1721-1733.

Zhang, Z., Sugio, S., Komives, E.A., Liu, K.D., Knowles, J.R., Petsko, G.A., and Ringe, D. 1994. Crystal structure of recombinant chicken triosephosphate isomerase-phosphoglycolohydroxamate complex at 1.8-Å resolution. *Biochemistry* 33:2830-2837.

Contributed by Mary Jo Ondrechen
Northeastern University
Boston, Massachusetts

Using the Reactome Database

The completion of multiple genomes in recent years has led to an explosion of information about known and predicted gene products. This information explosion has been accelerated by the invention of high-throughput experimental techniques, such as microarrays (see Chapter 7), yeast two-hybrid screens, and ChIP on Chip techniques, which allow experimentalists to ask questions about tens of thousands of genes simultaneously. As a result, biological researchers now face an embarrassment of riches: there is simply too much information to easily digest and interpret.

One way to reduce the complexity of this information is to adopt a high-level view of biological pathways. A microarray experiment that changes the expression pattern of thousands of genes may only affect the expression patterns of a small handful of biochemical pathways. Hence there is a high degree of interest in the bioinformatics community in creating pathway databases. The Reactome project, covered in this unit, is one such database. It is a curated collection of well documented molecular reactions that span the gamut from simple intermediate metabolism (e.g., sugar catabolism) to complex cellular events such as the mitotic cell cycle. These reactions are gathered by experts in the field, peer reviewed, and edited by professional staff members prior to being published in the database. A semiautomated procedure supplements this information by identifying likely orthologous molecular reactions in mouse, rat, zebrafish, and other model organisms.

The protocols in this unit illustrate how to use Reactome to learn the steps of a biological pathway and see how one pathway interacts with another. Basic Protocol 1 describes how to navigate and browse through the Reactome database. Basic Protocol 2 and Alternate Protocol 1 explain how to identify the pathways in which a molecule of interest is involved using either the common name or accession number, respectively. Basic Protocol 3 details how to use the Pathfinder tool to search the database for possible connections within and between pathways. Alternate Protocol 2 describes when and how to use the Advanced Search feature.

NOTE: This information is based on Reactome in July 2004. Some of the Web pages may have changed somewhat since the unit written.

BROWSING A REACTOME PATHWAY

BASIC PROTOCOL 1

This protocol will introduce the basic navigational techniques needed to browse the Reactome database.

Necessary Resources

Hardware

Computer capable of supporting a Web browser, and an Internet connection

Software

Any modern Web browser will work. The formatting of the Reactome pages may look best using Internet Explorer 4.0 or higher, or Netscape 7.0 or higher.

Contributed by Lincoln D. Stein
Current Protocols in Bioinformatics (2004) 8.7.1-8.7.16

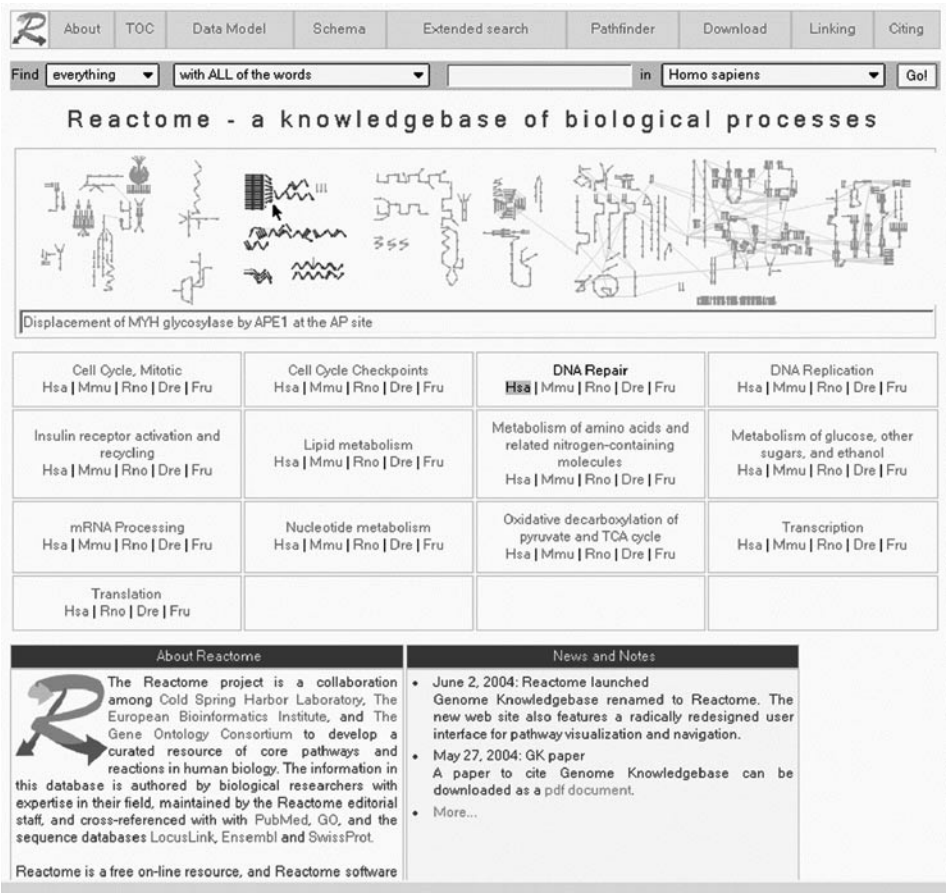


Figure 8.7.1 The Reactome home page features an interactive reaction map. Each "constellation" is a pathway. As the mouse is moved over the reaction map, the corresponding pathway in the table of contents is highlighted.

1. Point the browser to the Reactome home page at *http://www.reactome.org*.

   The home page (Fig. 8.7.1) has several elements.

   *The **menu bar** at the very top of the page provides access to the top-level sections of the Reactome site. "About" is a description of the project as a whole; "TOC" is an extensive table of contents for the resource; "Data Model" documents the underlying structure of the database and introduces the technical terms used by the database; "Schema" is a more technical view of the data model; "Extended search" is an interface to structured searches of the database; "Pathfinder" allows one to search for reactions that connect one pathway to another (see Basic Protocol 3); "Download" provides access to the whole database as a single bulk download; and the "Linking" and "Citing" sections provide information on how to link to Reactome and how to cite its contents in journal articles, respectively.*

   *The **search bar**, located just below the menu bar, provides for flexible keyword searches on the Reactome database.*

   *Below the search bar is the **reaction map**, also known as the "starry sky" in some of the Reactome documentation. This is a birds-eye view of all the reactions known to the database. Each arrow in the reaction map corresponds to a single reaction in Reactome, where a reaction is defined as a molecular interaction that transforms one or more input molecules into one or more output molecules. The reactions are grouped into a set of distinctive "constellations" that correspond to pathways of closely related reactions. As one moves the mouse over the reaction map, the pathway underneath the pointer becomes highlighted, and the corresponding pathway header in the table of contents lights up. If the*

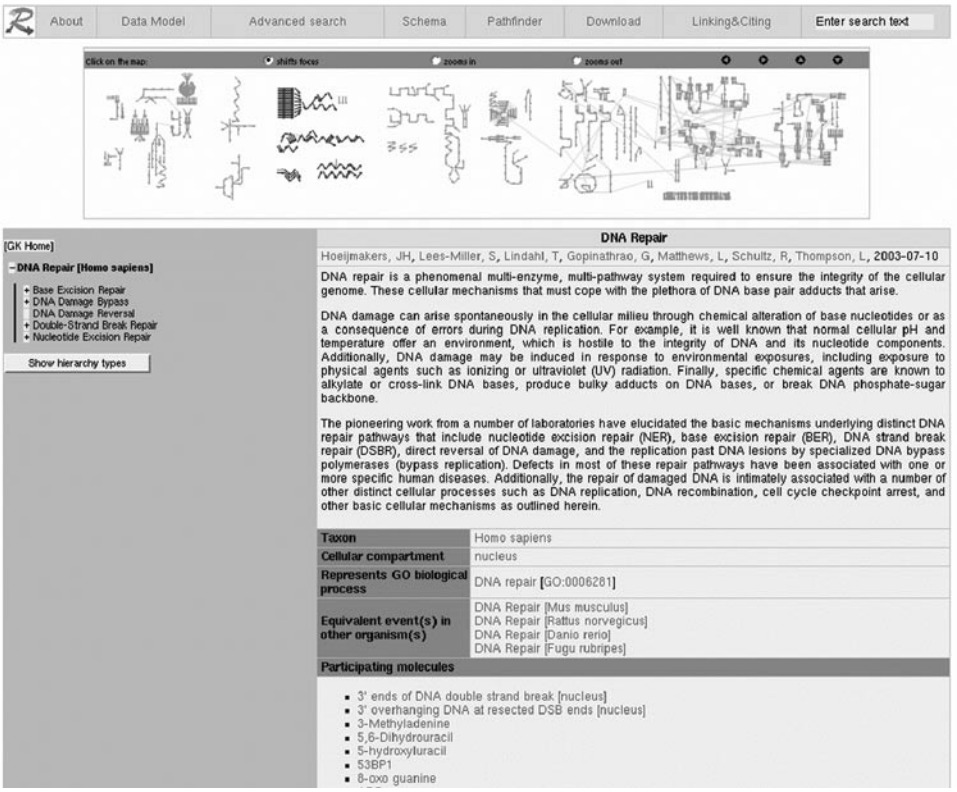


Figure 8.7.2 The top-level page describing DNA Repair. The navigation panel to the left contains an expanding hierarchical representation of the current pathway, showing the subpathways that participate in it.

mouse is paused over an individual event arrow, a message box with the name of the event pops open.

*Below the reaction map is the **table of contents**, which provides top-down access to the pathways known to Reactome. Underneath the name of each pathway are links that lead to reactions that occur in Homo sapiens (Hsa) and each of four model organisms: Mmu, Mus musculus; Rno, Rattus norvegicus; Dre, Danio rerio; Fru, Fugu rubripes. Clicking on the name of the pathway is equivalent to clicking on Hsa, and will lead to the beginning of the corresponding human pathway.*

***Links and notes** provides news and information about the project .*

2. To begin browsing the reactions contained within the DNA Repair pathway, click on the DNA Repair title in the table of contents. This will load a page corresponding to the top level of the DNA Repair pathway for *Homo sapiens* (Fig. 8.7.2).

The elements of this page are:

*The **reaction map**. This version of the reaction map has an additional control bar containing radio buttons that allow one to change how the reaction map behaves when one clicks in it. "Shifts focus" (the default) will link to the description of an individual reaction when its arrow is clicked. "Zooms in" and "Zooms out" will magnify or reduce the resolution of the reaction map, respectively. In addition, there is a set of arrow icons on the right side of this control bar that will scroll the reaction map in the corresponding direction when the map is in a zoomed-in state (i.e., when the "Zooms in" radio button is selected).*

*The **navigation panel**, occupying the vertical rectangle on the far left of the screen. This panel shows a collapsing hierarchical view of the DNA Repair pathway. The five headings underneath DNA Repair are the major divisions of the pathway, such as Base Excision*

Repair and Double-Strand Break Repair. The + marks mean that there are subheadings underneath the headings. Clicking on a + will expand the topic to show its subparts.

*The **main screen**, to the right of the navigation panel, containing the description of the pathway. This is the meat of the information contained within Reactome. The main screen begins with the authors, peer reviewers, and editors for this pathway, along with the date that the pathway was first released. This is followed by a text "summation" that describes the pathway. Below the summation are more details about the pathway, including the taxon in which the reaction occurs, the Gene Ontology classification(s) of the pathway, and the cellular compartment in which the pathway is known to occur. Further down are two important fields. The field that reads "Equivalent event(s) in other organism(s)" allows one to jump to the corresponding processes in the other model organism systems. The "Participating molecules" field lists all proteins, nucleic acids, complexes and small molecules, and complexes of these entities that are involved in any of the myriad aspects of DNA repair.*

3. Drill down into the Global Genomic Nucleotide Excision Repair subpathway as follows. The last entry in the navigation panel is Nucleotide Excision Repair. Click on it to open this level of the hierarchy, revealing the subentries "Global Genomic NER (GG-NER)" and "Transcription-coupled NER (TC-NER)." Click on Global Genomic NER (GG-NER), to reveal the page shown in Figure 8.7.3.

Notice that the navigation panel has now expanded by a level to reveal the relationship between global genomic nucleotide excision repair and the more general pathways that it belongs to on the one hand, and to the more specific pathways ("DNA Damage Recognition. . .," "Formation of incision complex. . .," etc.) on the other hand. Further, the

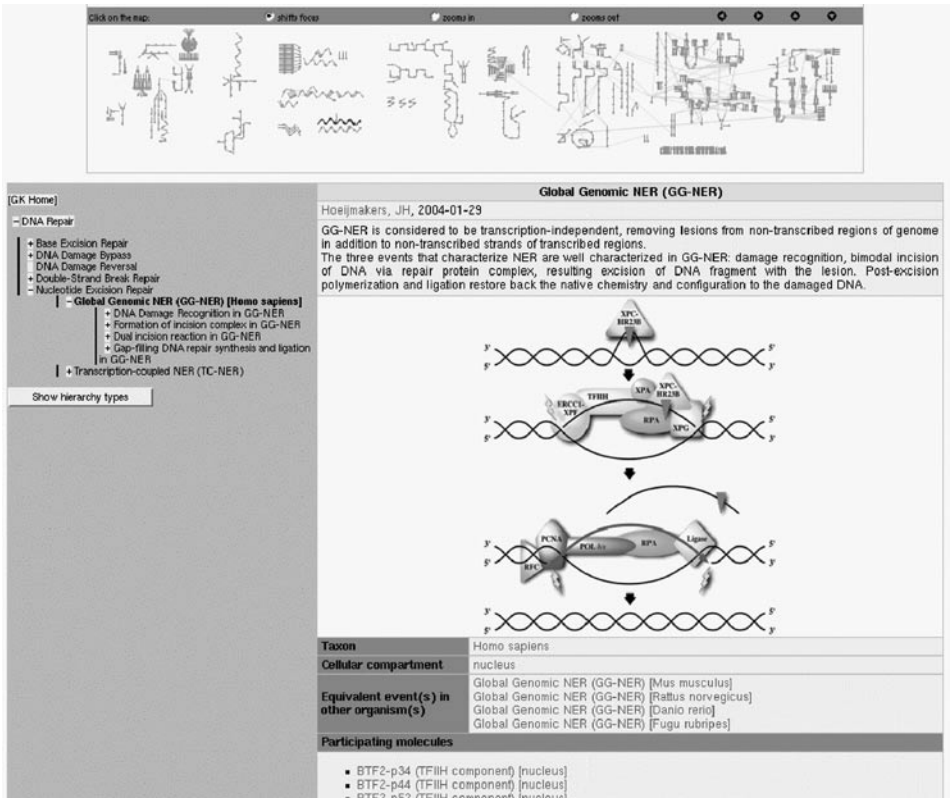


Figure 8.7.3 The main screen after drilling down to the Global Genomic NER (GG-NER) subpathway. The navigation panel on the left has opened up to indicate the subpathways of GG-NER, and the highlighting in the reaction map now indicates the reactions involved in this subpathway only.

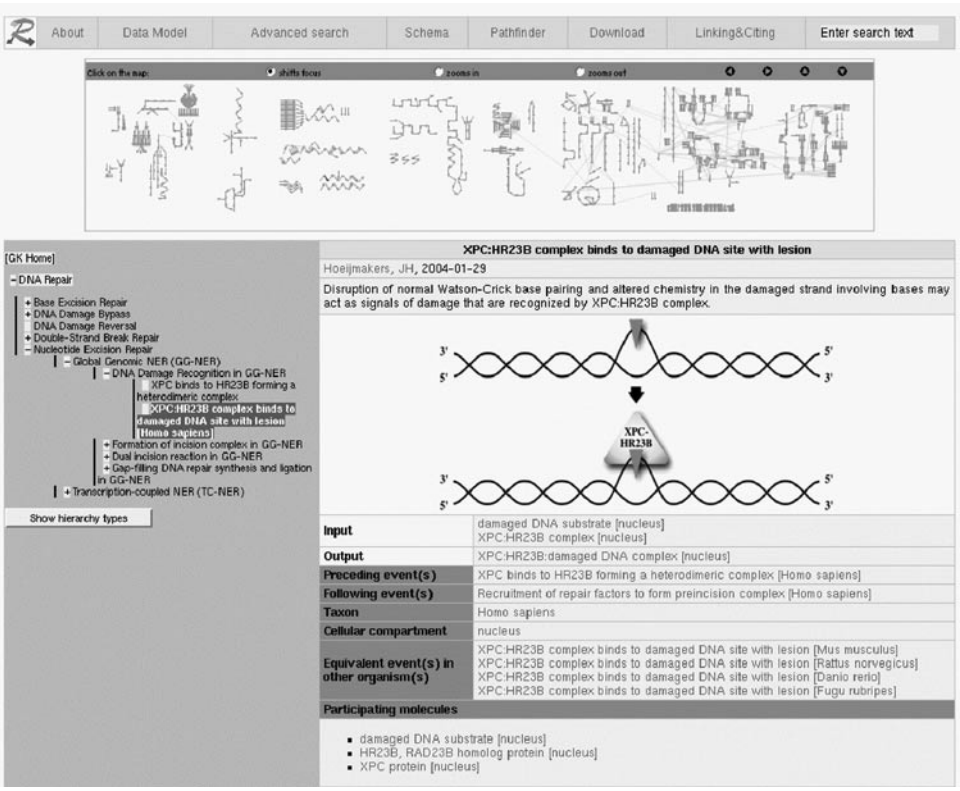


Figure 8.7.4 An individual reaction. Notice that a single reaction arrow is highlighted in the reaction map, and that the information in the main screen now shows the constituent input and output molecular compounds that participate in this reaction.

highlighting in the reaction map is now restricted to the reactions that are involved in global genomic nucleotide excision repair. The main screen describes the process in text form, and is accompanied by a cartoon overview. A much smaller set of participating molecules (partially scrolled out of view in the figure) lists the proteins, complexes, and other molecules that participate in this process.

4. In order to drill down to the reaction level, continue to click on subpathways. Eventually the reaction level will be reached, where processes are described as the interactions of individual molecules. To see this, return to the navigation panel and click first on "DNA Damage Recognition in GG-NER" and then on "XPC:HR23B complex binds to damaged DNA site with lesion [Homo sapiens]" to go to the page shown in Figure 8.7.4.

   A reaction-level page is similar to the upper-level pages, with a few important differences. First of all, the reaction map on the reaction-level page highlights a single reaction arrow only, indicating that one is at the lowest level of a pathway. Second, several additional fields appear below the text description of the reaction. These new fields include Input, which lists the molecules that enter the reaction, and Output, which lists the molecules that result from the reaction. In the case of the current reaction, the inputs are the damaged DNA substrate and the XPC:HR23B nucleotide excision complex, while the output is the complex of XPC:HR23B with the damaged DNA. In other words, this reaction describes the binding of XPC:HR23B to damaged DNA prior to the subsequent enzymatic reactions that cleave the DNA and excise the damaged base pair.

   Two other new fields are also shown. "Preceding event(s)" describes the reaction that immediately precedes this one temporally, in this case "XPC binds to HR23B forming a heterodimeric complex. . . .". "Following event(s)" describes the reaction that immediately follows this one: "Recruitment of repair factors to form preincision complex. . .". One can

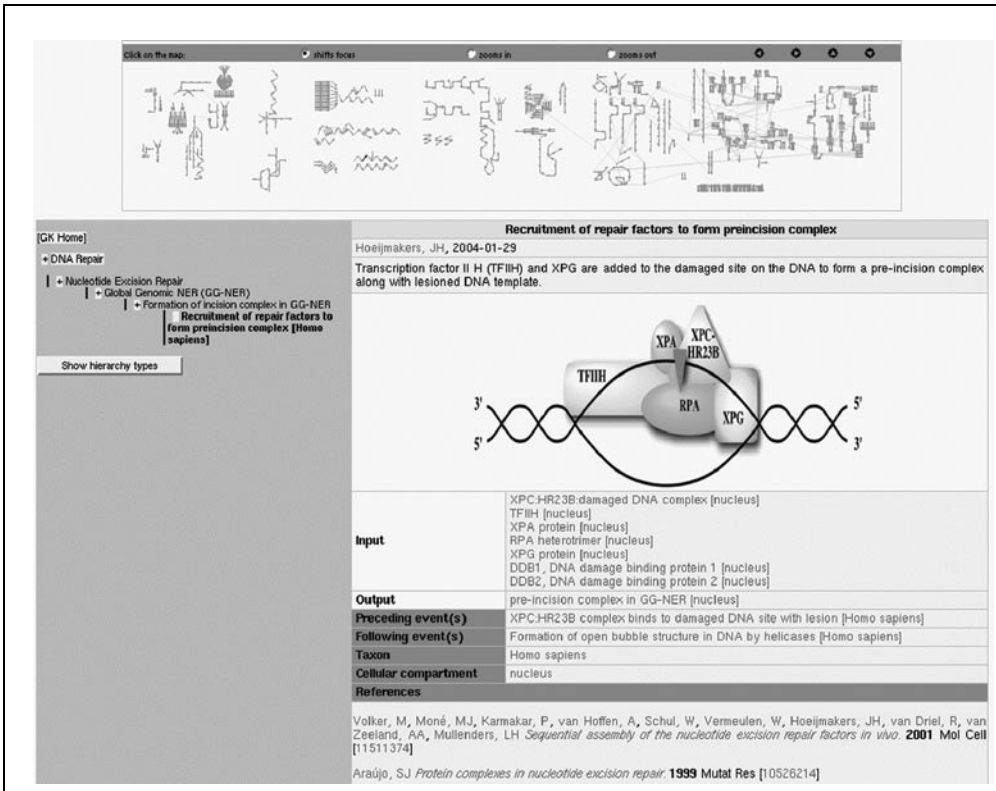


Figure 8.7.5 After clicking the "Following event(s)" link in the previous figure, the next step in the GG-NER pathway is displayed.

click on the preceding and following events to follow the reactions backward and forward in time.

5. Move to the next reaction by clicking on the "Following event(s)" link, "Recruitment of repair factors to form preincision complex." This will lead to the page shown in Figure 8.7.5, which describes the recruitment of six new proteins and complexes to create a single complex bound at the site of the damaged DNA. This page shows a preceding event of "XPC:HR23B complex binds to damaged DNA site with lesion [Homo sapiens]," which is the page shown in Figure 8.7.4, and a following event of "Formation of open bubble structure in DNA by helicases [Homo sapiens]." By clicking on the "Following event(s)" link, it would be possible to continue to follow the process forward in time.

The relationship between the "levels" of the navigation bar on the one hand and the "Preceding event(s)" and "Following event(s)" links, on the other hand, may not be immediately clear. These represent two distinct ways of viewing pathways. The nested levels of the navigation bar reflect levels of abstraction in the conceptual organization of pathways. As one moves deeper into the hierarchy, the contents of the main screen become more and more specific and move closer to the biochemical reaction level. The "Preceding event(s)" and "Following event(s)" links, on the other hand, usually only appear when one is at the reaction level, and move backward and forward in time, remaining always at individual reactions. It might seem to be redundant to have this dual mode of navigation, but it is there for a good reason. Because biological knowledge is incomplete, there are many instances where it is known that "something happens next," but the specific molecules that are involved in this next step are not yet characterized. In this case, the "Following event(s)" link will be missing, and one must step up in the hierarchy to a more general description of the pathway in order to connect to the next known, well characterized reaction in the process.

Figure 8.7.5 also illustrates an important aspect of Reactome, the References section at the bottom of the screen. Every reaction described in the database is supported by some type of provenance. The three main types of provenance are direct literature citations, an indirect assertion made by arguing from protein-based similarity in a model organism, and an assertion made by the author of the module. In the case of direct literature citations, the citation describes experiments performed using a system derived from the taxon under consideration. For example, the first reference in the current reaction describes in vivo experiments performed on human tissue culture cells that provided direct evidence via molecular cross-linking of an association between the XPC:HR23B/DNA complex and the repair factors recruited during this step.

Often, knowledge of human biology is derived from work on model organisms. If understanding of a reaction is derived from work on a model organism system, the references will describe those experiments. Internally, direct evidence and indirect evidence from model organisms are kept distinct, but the user interface does not currently reflect that fact.

Finally, the high-level, more general pathways will usually be based on an assertion by the author of the module and supported by one or more review articles. Click on the author's name at the top of the main screen to see the list of review articles that describe the pathway.

6. Reactome provides information about the subunits of a complex, as well as the larger ensembles of proteins that a complex participates in. In this example, from the "Recruitment of repair factors to form preincision complex" page, click on the "TFIIH" link in the Input section. This will load a page that contains information about the TFIIH (transcription factor IIH) complex (Fig. 8.7.6).

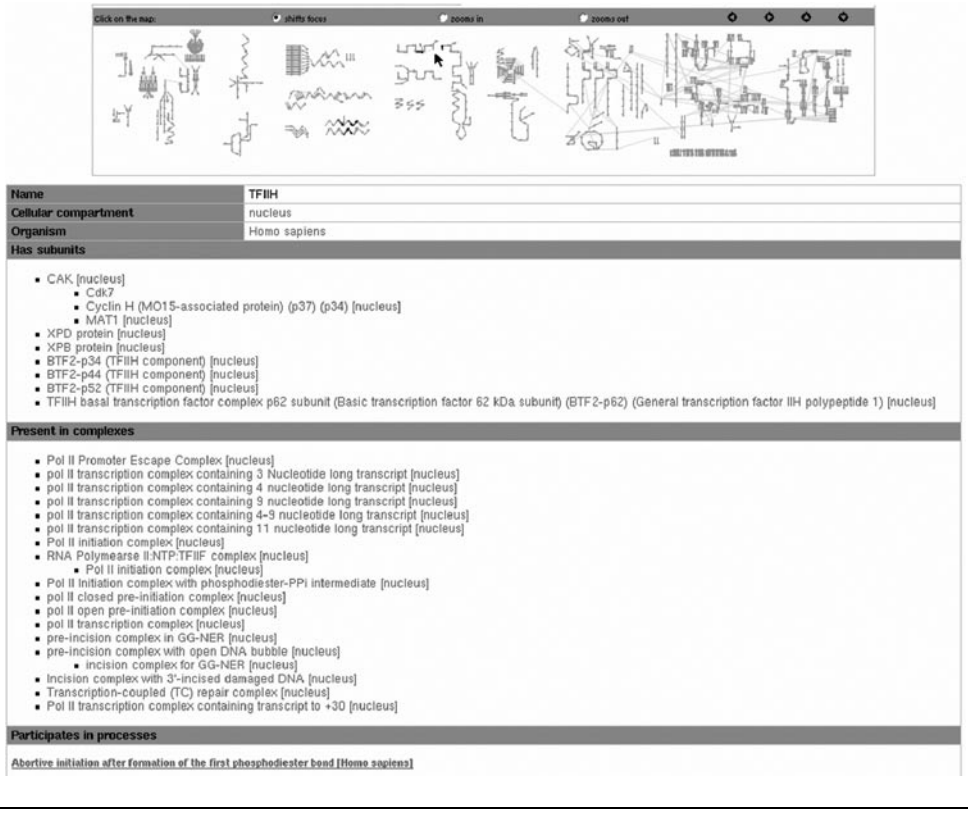


Figure 8.7.6 This page describes the TFIIH protein. In addition to describing its subunit structure, the page notes all the macromolecular complexes and pathways in which TFIIH participates. The reaction map highlighting indicates that, in addition to DNA repair processes, TFIIH is involved in mRNA transcription (arrow).

Because this page describes a molecule and not a reaction or pathway, there is no navigation panel on the left. However, the reaction map at the top of the page is still present, and it lights up to highlight the reactions in which TFIIH participates. Mousing over the highlighted pathways reveals that, in addition to the DNA excision repair pathway that has been browsed in the steps above, TFIIH also participates in PolII-mediated RNA transcription. This connection between RNA transcription and DNA repair might surprise biologists who are not well acquainted with DNA excision repair, and illustrates how Reactome bridges the disciplines.

The section near the bottom of Figure 8.7.6 labeled "Participates in processes" lists all the pathways and reactions in which TFIIH participates. Although not shown in Figure 8.7.6, at the top of this section there is an extensive list of all the reactions in which the current molecule participates. This is organized in a hierarchical manner that mirrors the pathway hierarchy of the navigation panel. At the bottom, these events are organized into three groups: all events that produce TFIIH, all that consume it, and all that are catalyzed by it.

7. To learn more about a protein subunit, click on the subunit of interest. In this case, one of the subunits of TFIIH is Cdk7 (shown in Fig. 8.7.6; it complexes with Cyclin H and MAT1 to form the CAK subcomplex, which in turn is one of the major components of TFIIH). Click on the Cdk7 link to load a page that describes it (Fig. 8.7.7). In addition to highlighting the DNA repair and RNA transcription constellations, the reaction map now shows highlighting in the Mitotic Cell Cycle constellation as well (upper left quadrant of the image), reflecting Cdk7's role as a cell-cycle checkpoint molecule.

   This page is called the "reference entity" page because it contains links to UniProt, Ensembl, and other reference databases that describe the molecule.

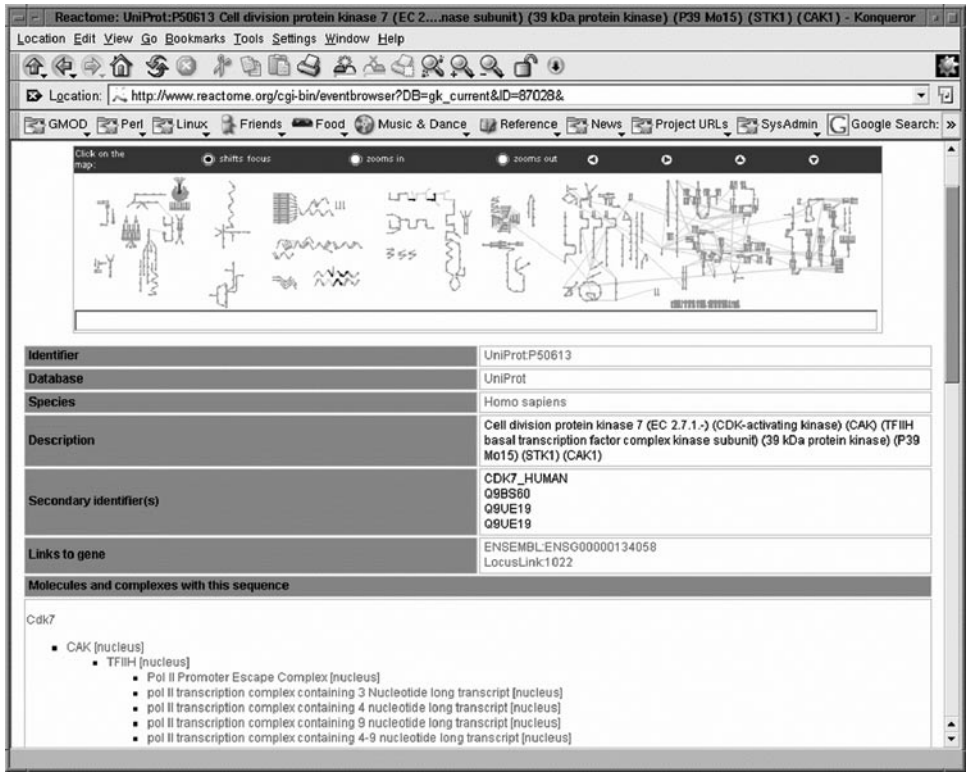


Figure 8.7.7 The reference entity page describes the relationship between a molecule as it is represented in Reactome and one or more entries in a third-party database such as SwissProt.

FINDING THE PATHWAYS INVOLVING A GENE OR PROTEIN

This protocol will describe how to identify pathways and reactions that involve a gene or protein of interest. For the purposes of illustration, the cyclin-dependent kinase 7 gene will be used, which has the following identifiers:

Protein product:	Common name: Cdk7
	UniProt (SwissProt): CDK7_HUMAN
Gene:	LocusLink: 1022
	GenBank: NM_001799
	Ensembl: ENSG00000134058.

See Alternate Protocol 1 to search by a database accession number rather than by a common name.

Necessary Resources

Hardware

Computer capable of supporting a Web browser, and an Internet connection

Software

Any modern Web browser will work. The formatting of the Reactome pages may look best using Internet Explorer 4.0 or higher, or Netscape 7.0 or higher.

1. Point the browser to the Reactome home page at *http://www.reactome.org*.
2. On the home page (Fig. 8.7.1), in the search bar near the top of the page (see annotation to step 1 of Basic Protocol 1), click the text box (second box from the right-hand side of the search bar), type `Cdk7`, then press the Enter key (or click the Go! button). This brings up the search results page shown in Figure 8.7.8.

   For now, ignore the text fields and buttons that occupy most of the real estate at the top of the page, and focus on the section at the bottom under the heading "Found 8 instances in the following categories." This section tells the user that Reactome knows of 1 Literature reference, 1 summation, 4 ReferenceEntities, and 2 PhysicalEntities that have something to do with Cdk7. Summations are the text paragraphs that appear at the top of pages that describe pathways and reactions. ReferenceEntities are lists of protein and gene entries that appear in online genome databases. What are needed, although not apparent from the name, are the PhysicalEntities, which is the term that Reactome uses for anything that has

Figure 8.7.8 Results from the quick search on the Reactome home page are displayed at the bottom of the full-featured Advanced Search page.

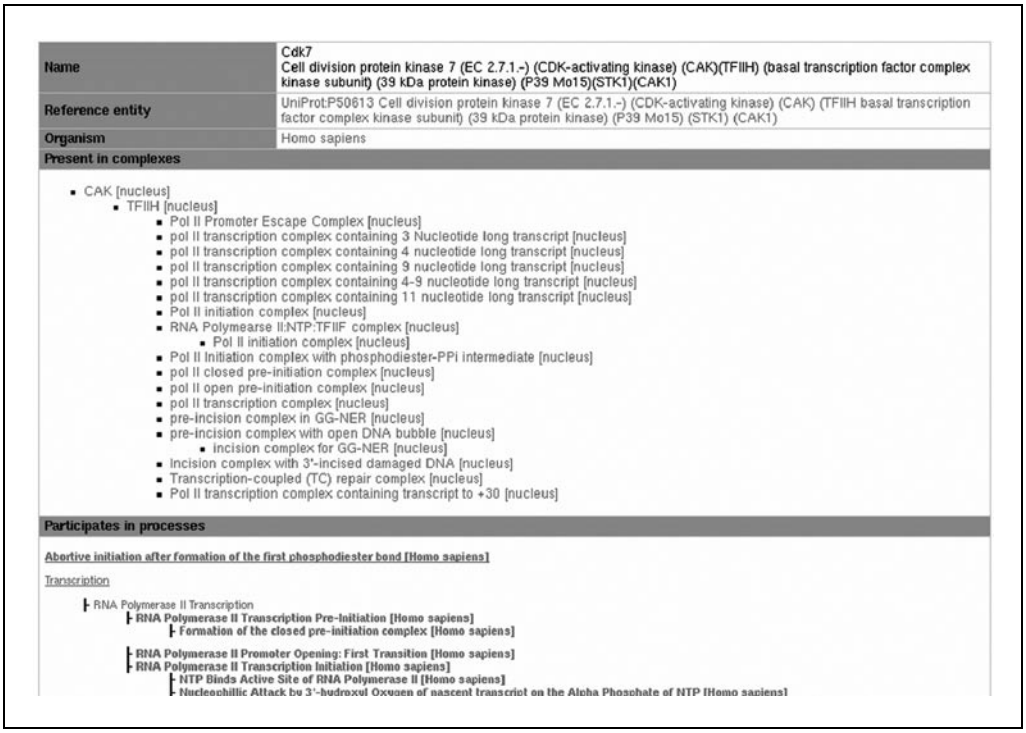


Figure 8.7.9 Following the search results to the Cdk7 page displays the structure of Cdk7 and a hierarchical list of the pathways in which it is known to participate.

mass, such as a macromolecule. The search interface will be modified in the near future to make it easier to interpret.

3. Navigate to the Cdk7 entry by clicking on the "2" link that appears after the PhysicalEntity label. This will lead to a list of two entries in Reactome, MAT1 (also known as "Cdk7 assembly factor") and Cdk7 itself. Click on the "Cdk7" link. This will lead to the page shown in Figure 8.7.9.

   This page, which is similar to the TFIIH page shown in Figure 8.7.7, describes everything that Reactome knows about Cdk7, including its names in other online databases, the protein complexes that it belongs to, and the pathways and reactions that it participates in. Any of these links can be clicked to begin browsing the pathways involving Cdk7 as described in Basic Protocol 1.

ALTERNATE PROTOCOL 1

FINDING THE PATHWAYS INVOLVING A GENE OR PROTEIN USING SwissProt, Ensembl, OR LocusLink NAME

Instead of searching for a gene or protein using its common name, as described in Basic Protocol 2, one may wish to use the accession number by which it is known in SwissProt, Ensembl, or LocusLink. The steps for doing so, using a SwissProt accession number, are presented here. The same procedure works for Ensembl or LocusLink identifiers. However, Reactome does not currently recognize GenBank accession numbers, e.g., NM_001799, because of the redundancy of GenBank entries. If one wish to find a protein based on its GenBank accession number, one should first use NCBI LocusLink to find the correct LocusLink number, and then use this number to access the appropriate entry in Reactome.

Necessary Resources

Hardware

Computer capable of supporting a Web browser, and an Internet connection

Software

Any modern Web browser will work. The formatting of the Reactome pages may look best using Internet Explorer 4.0 or higher, or Netscape 7.0 or higher.

1. Point the browser to the Reactome home page at *http://www.reactome.org*.
2. On the home page (Fig. 8.7.1), in the search bar near the top of the page (see annotation to step 1 of Basic Protocol 1), click the text box (second box from the right-hand side of the search bar), type `CDK7_HUMAN`, then press the Enter key (or Click the Go! button).

   This brings up a reference entity page (see Basic Protocol 1, step 7) similar to the one shown in Figure 8.7.7.

3. Navigate to the molecule or pathway of interest. The reference entity page is similar in most respects to the PhysicalEntity page shown in Figure 8.7.9. From here it is possible to navigate to the pathways and reactions in which Cdk7 takes part, view the complexes that contain Cdk7, or link to the PhysicalEntity page shown in Figure 8.7.9.

USING ADVANCED SEARCH

ALTERNATE PROTOCOL 2

The simple searches shown in Basic Protocol 2 and Alternate Protocol 1 will suffice for many situations. However, the default search casts a very wide net and may return more hits than one wants. If this is the case, one may wish to use the Advanced Search, which gives much finer control over the search. To illustrate, this protocol describe how to search for "pyruvate dehydrogenase," whose default search returns multiple hits on compounds, events, literature references, and other database entries.

Necessary Resources

Hardware

Computer capable of supporting a Web browser, and an Internet connection

Software

Any modern Web browser will work. The formatting of the Reactome pages may look best using Internet Explorer 4.0 or higher, or Netscape 7.0 or higher.

1. Point the Web browser to the Reactome home page at *http://www.reactome.org*.
2. On the home page (Fig. 8.7.1), in the search bar near the top of the page (see annotation to step 1 of Basic Protocol 1), click the text box (second box from the right-hand side of the search bar), and type `pyruvate dehydrogenase`.

   Pyruvate dehydrogenase is a protein complex, and one might like to limit the search to database entries for complexes, as in step 3.

3. Go to the pull-down menu on the far left of the search bar and change the scope of the search from "everything" to "complexes," then press the Go! button to the right of the search bar.

   This will return a list of 13 complexes that contain the words "pyruvate dehydrogenase," including FADH2-linked pyruvate dehydrogenase complex, pyruvate dehydrogenase E2 holoenzyme, S-acetyldihydrolipoamide linked, and pyruvate dehydrogenase E2 trimer.

   By default, the search will find matches in Homo sapiens. If one wishes to see matches in another species, one can change the search parameters as in step 4.

4. To see matches in another species, press the browser's Back button to return to the Reactome home page. Locate the pull-down menu on the far right of the search bar (which reads "Homo sapiens" by default), and change it to "Rattus norvegicus." Press the Go! button again.

   A page will appear displaying 13 matches on the orthologous set of pyruvate dehydrogenase complexes in the Norway rat.

5. If the search is retrieving unwanted matches, it is possible to further limit the set of hits by specifying an exact match for "pyruvate dehydrogenase complex," which will find database objects that match the search phrase exactly from end to end. Press the browser's Back button to return to the main page. Locate the search type pull-down menu, second to the left in the search bar, and change its setting from the default "with ALL of the words" to "with the EXACT PHRASE ONLY." It will also be necessary to modify the search phrase from "pyruvate dehydrogenase" to "pyruvate dehydrogenase complex" by clicking in the text field in the search bar and modifying the search phrase appropriately. Also change the species menu back to "Homo sapiens." Press the Go! button.

   Because there is only one hit, this search will lead directly to the page that describes pyruvate dehydrogenase complex.

   The pull-down menu also has an option (EXACT PHRASE) that allows the user to search for a match on the phrase embedded in a longer string of text or by itself, which is a type of wildcard match.

BASIC PROTOCOL 3

USING THE PATHFINDER

The Pathfinder tool allows one to search for reactions that connect one molecule to another. It is a powerful exploratory and visualization tool when used prudently. To illustrate how it works, this protocol will describe a search in Reactome for events that connect the DNA origin of replication, an essential ingredient in DNA replication, to the polymerase II transcription complex, a key entity in pol II-mediated RNA transcription.

Necessary Resources

Hardware

Computer capable of supporting a Web browser, and an Internet connection

Software

Any modern Web browser will work. The formatting of the Reactome pages may look best using Internet Explorer 4.0 or higher, or Netscape 7.0 or higher.

1. Point the browser to the Reactome home page at *http://www.reactome.org*.
2. In the menu bar at the top of the home page, click on the link labeled Pathfinder.
3. Type "origin of replication" into the text field labeled "Start compound or event name," and "pol II transcription complex" into the text field labeled "End compound or event name," then press Enter or click the Go! button.

   The Pathfinder works by accepting a starting compound, pathway or reaction, and an ending compound, pathway or reaction. It then attempts to find the shortest set of reactions in the database that connects the two.

4. After step 3 has been carried out, Reactome will find all likely matches to the two compounds. If it finds more than one, which is usually the case, it will place the candidates in a pair of pull-down menus (Fig. 8.7.10). The user should pull down

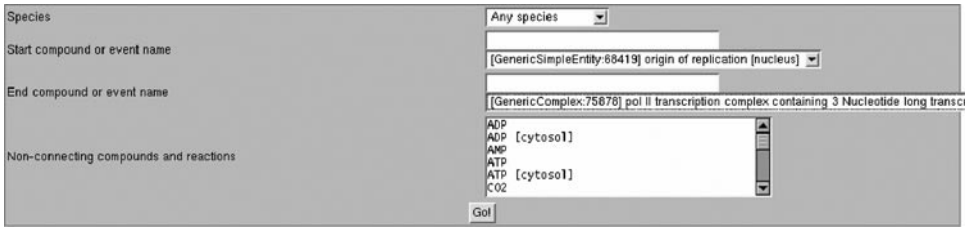


Figure 8.7.10 After entering the names of the start and end compounds, the Pathfinder will display pull-down menus of candidate compounds known to Reactome. Pull down the menus in order to fine tune Reactome's choice of compounds.

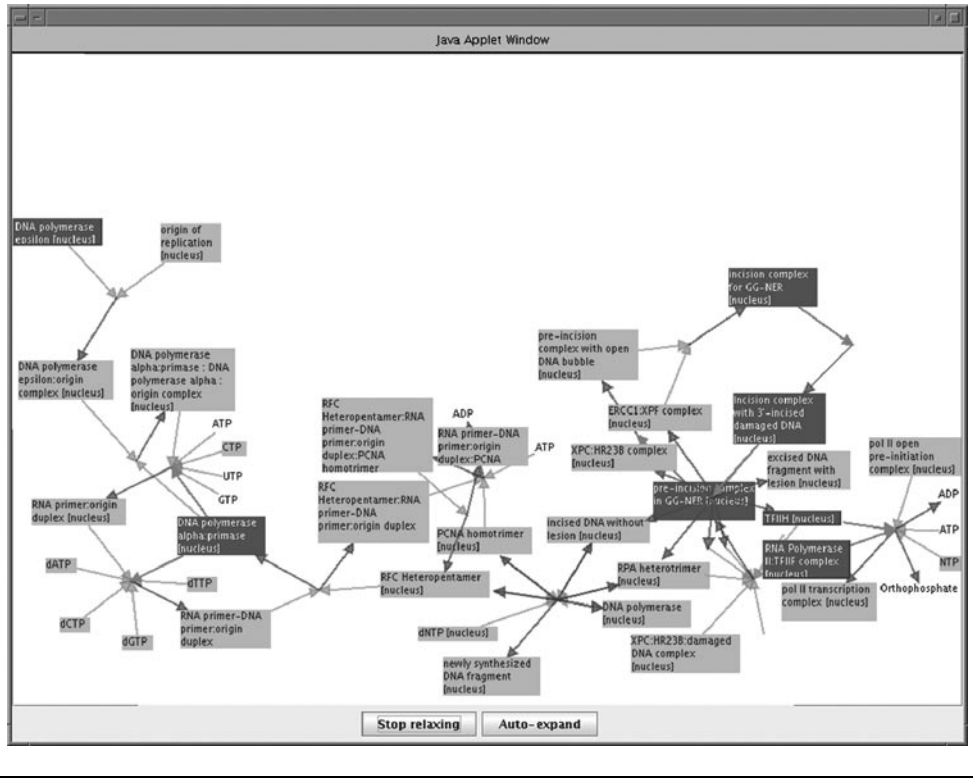


Figure 8.7.11 The PathFinder graphic display shows all the steps necessary to traverse from the starting compound to the ending compound.

the menus and select the best match to what was intended. In the current example, Reactome got the start compound right, but guessed incorrectly for the end compound, returning an intermediate complex that happens to involve pol II transcription. Click on the end compound menu and select "pol II transcription complex." If the sought-after item is not found at first, try rephrasing it and typing it into the appropriate text field, then pressing Enter. This will update the list of candidates in the pull-down menu.

5. When the correct start and end compounds have been selected, press the Go! button at the bottom of the panel. In a few seconds the page will refresh and display a list of reactions that together connect the origin of replication to the pol II transcription complex. The found path traverses the DNA repair pathways, which involve both DNA replication and transcription factors. One can click on any of these steps to begin browsing Reactome at that point.

6. Press the button labeled View in Pathway underneath the Pathfinder list. Provided that Java is installed and running on the system, a new image window will pop up that shows this pathway in a graphical form (Fig. 8.7.11).

The user can interact with the pathway visualization in a limited manner in order to make it more visually appealing. To do this, press the button labeled "Stop relaxing" to stop the automatic layout process and fix the reaction boxes in place. Next, grab the boxes with the mouse and move them into the preferred positions.

The Pathfinder visualization does not currently support exporting the display as a static image. However, it is possible to use the screenshot feature of one's local computer (Alt-Print Scr on the PC) in order to capture the pathway. Also note that the View in Pathway button will not appear unless Java is installed.

COMMENTARY

Background Information

The Reactome project is a collaboration between Cold Spring Harbor Laboratory and The European Bioinformatics Institute, and aims to collect structured information on all the biological pathways in the human (Joshi-Tope et al., 2003; see Internet Resources for online version of this paper). The project is building its database by inviting faculty-level laboratory researchers to contribute a pathway or sub-pathway to the database. To achieve this, contributors are instructed on the use of a specialized piece of authoring software and are assisted in their work by a staff of curators based at the two institutions. After authoring, each pathway is checked for consistency both manually and automatically and then sent to one or more external peer reviewers. The pathway is published to the Web when all internal and external peer review is satisfactory. In many ways, the project resembles a review journal, except that its output is a database rather than a series of papers.

In order to assist authors in organizing their domain of knowledge into a set of defined pathways, *Reactome* relies on frequent "mini-jamborees" of roughly a half-dozen authors. During these jamborees, which are held in conjunction with international meetings, authors working on a set of related pathways get together in the same room and work out the logical structure of their topic. This is also an opportunity for Reactome curators to train the authors in the use of the authoring software.

Reactome uses a simple scheme for describing biological pathways in which all molecular interactions are defined as reactions. A reaction takes a series of inputs and transforms them into a series of outputs, where inputs and outputs are any type of molecular compound. For example, the reaction in which proinsulin is cleaved to form the α and β chains takes as its input proinsulin, and produces the insulin α and β polypeptides.

Representing biology as a set of molecular reactions turns out to have broad expressive power, but sometimes the results are disorienting. For example, the reaction in which insulin binds to the insulin receptor takes as its inputs extracellular insulin and the extracellular portion of the insulin receptor, and produces as its output the complex of insulin and its receptor, which, in Reactome, is represented as a distinct molecular entity. The reaction by which extracellular glucose is transported into the cytosol transforms extracellular D-glucose into intracellular D-glucose. Hence, a search of Reactome for "D-glucose" will find both D-glucose (intracellular cytosolic) and D-glucose (extracellular).

In addition to inputs and outputs, Reactome reactions have a discrete set of additional attributes. For those reactions that are mediated by catalysts, the catalyst enzyme and its activity are noted. Reactions are also annotated using the cellular compartment in which they occur. While Reactome does not pretend to be a definitive source of information on the cellular location of macromolecules, its data model is set up to work smoothly with future databases of subcellular localization; information on the subcellular location of macromolecules will help automated path-prediction software distinguish plausible pathways from impossible ones. Finally, each reaction is supported by literature citations, either those reporting experiments performed directly in the human system, or those performed on model systems when there is high-quality protein similarity data to suggest that the same reaction is likely to occur in humans.

In order to assist with the comprehensibility of the resource, the reactions are annotated with text narratives and illustrations, and are

organized into a series of discrete goal-driven pathways.

Reactome is related to several other pathway databases, but has distinct methodologies and aims. The Human Protein Reference Database (HPRD; Peri et al., 2003) is also a hand-curated database of biological pathways. The HPRD focus, however, is to annotate individual proteins and their physical and genetic interactions. HPRD contains information derived from large-scale screening studies as well as individual papers that report pairwise interactions. A result of this methodology is that many of the interactions found in HPRD are speculative and subject to change. Reactome takes a much more conservative approach; it represents far fewer molecular interactions than HPRD does, but they are more likely to be correct and less subject to revision.

HumanCyc (Krieger et al., 2004) is a database of biological pathways that uses a data model generally similar to Reactome, although the user interface and underlying database technology are quite different in detail. The focus of HumanCyc is intermediate metabolism, however. It tends to have more information on the creation and utilization of small molecules than does Reactome, but less information on such higher-level processes as transcription, translation, and the cell cycle.

The Kyoto Encyclopedia of Genes and Genomes, or KEGG (Kanehisa et al., 2004) features an extensive set of biological pathway charts. Like HumanCyc, KEGG focuses on intermediate metabolism rather than higher-level pathways. Its data model differs fundamentally from Reactome's by representing the motivating force of all reactions in the form of catalyst activities via Enzyme Commission EC numbers. Because there is not a one-to-one mapping between EC activity and polypeptide, it can be problematic to relate a protein represented in SwissProt to a reaction represented in KEGG.

Finally, the BioCarta project (*http://www.biocarta.com*) represents human biology as a series of colorful high-resolution diagrams. Unlike Reactome or the other projects mentioned earlier, these diagrams are the end product of the project; there is no underlying database. The focus of BioCarta is to be an education and visualization tool, rather than to support data mining and pattern discovery.

The Reactome database is far from complete. At the time this module was written, Reactome covered just 8% of the human genome, a number conservatively estimated by dividing the number of human SwissProt entries that take part in Reactome reactions by the total number of human entries in the entire SwissProt database. Because all of the other pathway databases mentioned here are also incomplete, the biologist faces the daunting task of visiting each of these sites in an attempt to fill in the holes in one database's coverage with information from the others. The BioPAX project (*http://www.biopax.org*) promises to improve this situation by creating a standardized file format for representing biological pathways and reactions. Reactome and many of the other pathway databases have committed to exporting their data in BioPAX format. In the future, this will enable the databases to exchange pathways and to co-curate data, thereby accelerating the rate in which the gaps are closed.

Reactome is a fully open-source project. All the software developed for use in Reactome is available for download and redistribution, and the data itself is available in a variety of formats. The Download link on the *Reactome* Web site provides instructions for obtaining data and software.

The Reactome dataset is available as relational database tables in a format compatible with MySQL (*http//www.mysql.com*; *UNIT 9.2*) and as files compatible with the Protégé-2000 knowledgebase editor (*http://protege.stanford.edu*) and will soon be available as tab-delimited text files.

Literature Cited

Joshi-Tope, G., Vastrik, I., Gopinath, G.R., Matthews, L., Schmidt, E., Gillespie, M., D'Eustachio, P., Jassal, B., Lewis, S., Wu, G., Birney, E., and Stein, L. 2003. The Genome Knowledgebase: A Resource for Biologists and Bioinformaticists. Cold Spring Harbor Symposia on Quantitative Biology LXVIII:237-244. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, N.Y.

Kanehisa, M., Goto, S., Kawashima, S., Okuno, Y., and Hattori, M. 2004. The KEGG resource for deciphering the genome. *Nucleic Acids Res.* 32:D277-D280.

Krieger, C.J., Zhang, P., Mueller, L.A., Wang, A., Paley, S., Arnaud, M., Pick, J., Rhee, S.Y., and Karp, P.D. 2004. MetaCyc: A multiorganism database of metabolic pathways and enzymes. *Nucleic Acids Res.* 32:D438-D442.

Peri, S., Navarro, J.D., Amanchy, R., Kristiansen, T.Z., Jonnalagadda, C.K., Surendranath, V., Niranjan, V., Muthusamy, B., Gandhi, T.K., Gronborg, M., Ibarrola, N., Deshpande, N., Shanker, K., Shivashankar, H.N., Rashmi, B.P., Ramya, M.A., Zhao, Z., Chandrika, K.N., Padma, N., Harsha, H.C., Yatish, A.J., Kavitha, M.P., Menezes, M., Choudhury, D.R., Suresh, S., Ghosh, N., Saravana, R., Chandran, S.,

Krishna, S., Joy, M., Anand, S.K., Madavan, V., Joseph, A., Wong, G.W., Schiemann, W.P., Constantinescu, S.N., Huang, L., Khosravi-Far, R., Steen, H., Tewari, M., Ghaffari, S., Blobe, G.C., Dang, C.V., Garcia, J.G., Pevsner, J., Jensen, O.N., Roepstorff, P., Deshpande, K.S., Chinnaiyan, A.M., Hamosh, A., Chakravarti, A., and Pandey, A. 2003. Development of human protein reference database as an initial platform for approaching systems biology in humans. *Genome Res.* 10:2363-2371.

Internet Resources

http://www.biocarta.com

The Biocarta human pathways project.

http://www.biopax.org

BioPAX: Biological Pathways Exchange. Standardizing the file format for representing biological pathways.

http://www.reactome.org

The Reactome home page.

http://www.reactome.org/gk_symposium.pdf

Online version of Joshi-Tope et al. (2003).

Contributed by Lincoln D. Stein
Cold Spring Harbor Laboratory
Cold Spring Harbor, New York

Analyzing Networks with VisANT

VisANT is a software platform for visually building and analyzing networks of relations among and between biological entities. Network nodes can represent various levels of biological organization, including molecules, complexes, pathways, and other functional modules. VisANT is supported by the Predictome database, which includes several hundred thousand relations based on some 33 experimental and computational methods. Networks uncovered by VisANT can be easily saved online and thereby shared with the wider community.

VisANT is predicated on the desirability of accessing and integrating multiple methods for inferring and extending relations across multiple species, and accessing and using data in a way that is not limited by the existence of diverse nomenclatures. One of its intermediate goals is to simulate and test hypotheses about the behavior of a cell under changes in environmental conditions; a long range goal is to do the same for groups of cells and organs.

Because VisANT displays relations based on a number of different kinds of evidence, links between nodes are displayed in different ways, depending on whether they represent direct physical interactions (e.g., yeast two-hybrid experiments, chromosome immunoprecipitation, mass spectrometry), functional correlations (e.g., microarray perturbation data, phylogenetic profiles), causal relations, and so forth. Some of these are discussed below, others are in the online VisANT user's manual *http://visant.bu.edu/vmanual*. VisANT also allows simultaneous searching of multiple genes and proteins for 72 species. Searchable terms include protein name, gene name, open reading frame (ORF) ID, GI number, and KEGG pathway ID. It also supports special retrieval terms for specific species such as locus link ID and Online Mendelian Inheritance in Man (OMIM; *UNIT 1.2*) for *Homo sapiens*. Additional details are in the VisANT user's manual.

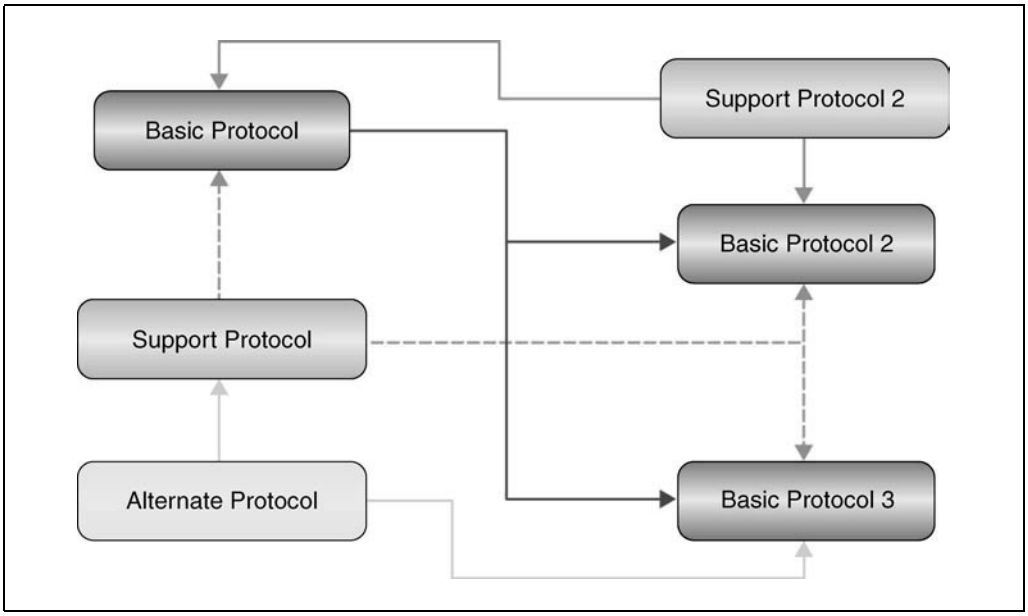


Figure 8.8.1 Relationships between protocols. Protocols are colored by type, with the direction of the line indicating relationship—for example, the Alternate Protocol is used by Basic Protocol 3 and Support Protocol 1. To distinguish the relationships of the Support Protocols, dashed lines are used for Support Protocol 1.

Contributed by Zhenjun Hu, Joseph Mellor, and Charles DeLisi
Current Protocols in Bioinformatics (2004) 8.8.1-8.8.24

This unit is organized around a set of visual data-mining protocols, i.e., procedures for constructing, displaying, manipulating, and analyzing large numbers of relations (Fig. 8.8.1). The first method (see Basic Protocol 1) covers basic network construction, while its alternative (see Alternate Protocol) shows how to quickly build and combine large scale networks. Additionally, an introduction to integrative analysis and annotation of constructed networks is presented (see Basic Protocol 2). Meta-network application (i.e., visualization and analysis of higher order networks, and embedding of multiple scales of organization) is discussed as well (see Basic Protocol 3). Finally, Support Protocol 1 describes analytical functions, such as those that enable characterization of network topology, while Support Protocol 2 introduces online network saving and sharing.

BASIC PROTOCOL 1

BASIC NETWORK CONSTRUCTION

As an example of how relations are used to visualize and analyze complex networks, this discussion will focus on the network of interactions in which the *Saccharomyces cerevisiae* proteins STE3 and FUS1 are embedded.

Necessary Resources

Hardware

Any computer with Internet access

Software

Java compatible browser
Java Run-time Environment (JRE) 1.4 or above (see Internet Resources)

Files

None

1. Start a Java-compatible browser and open the VisANT start page (*http://visant.bu.edu*). If the Start button in the WEB page (Fig. 8.8.2) is not visible, follow the instructions in the VisANT user's manual to install the required software (JRE).
2. Click the Start button, which will cause a VisANT window, having three main components, Menu Bar, Control Panel, and Network Panel (Fig. 8.8.3), to appear. Keep the start page open during all procedures.
3. Clear the network panel by clicking the Clear button in the control panel.
4. Select the genome to be analyzed, *S. cerevisiae* in this case, by scrolling through the In Species pulldown menu in the control panel.
5. Type `FUS1` and `STE3` in the Search Compound, Pathway & Protein/Gene Name box of the control panel.
6. Open the View menu on the menu bar and click Methods Tables (Fig. 8.8.4). Close the methods table in the usual way (e.g., click "X" in upper right corner).

   The Methods Table can also be accessed by right clicking on the network panel to invoke a pop-up menu.

   Notice that all methods are checked. This means that all associations stored in the Predictome database will be displayed. Throughout this discussion, when no method is specified, they are all invoked.

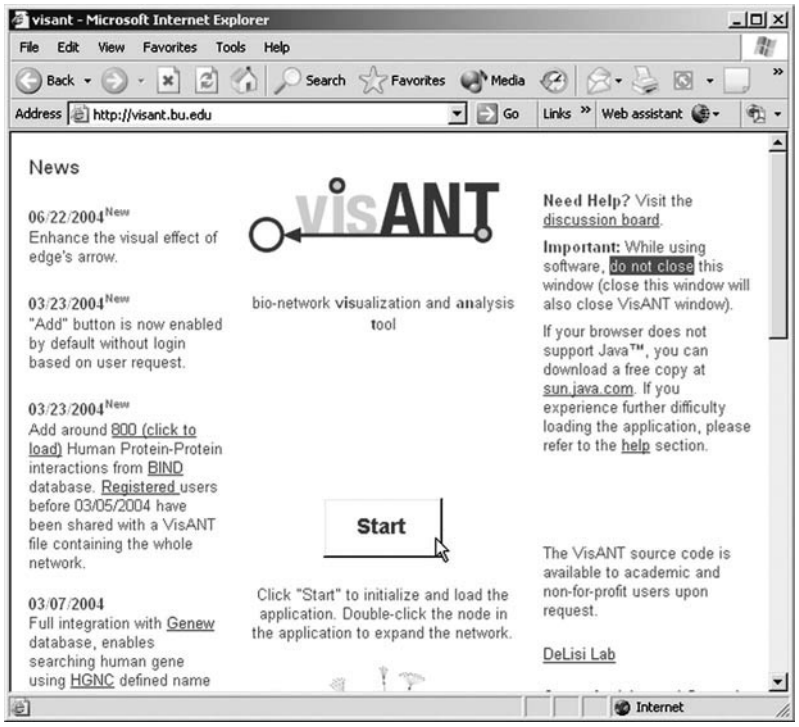


Figure 8.8.2 The VisANT start page.

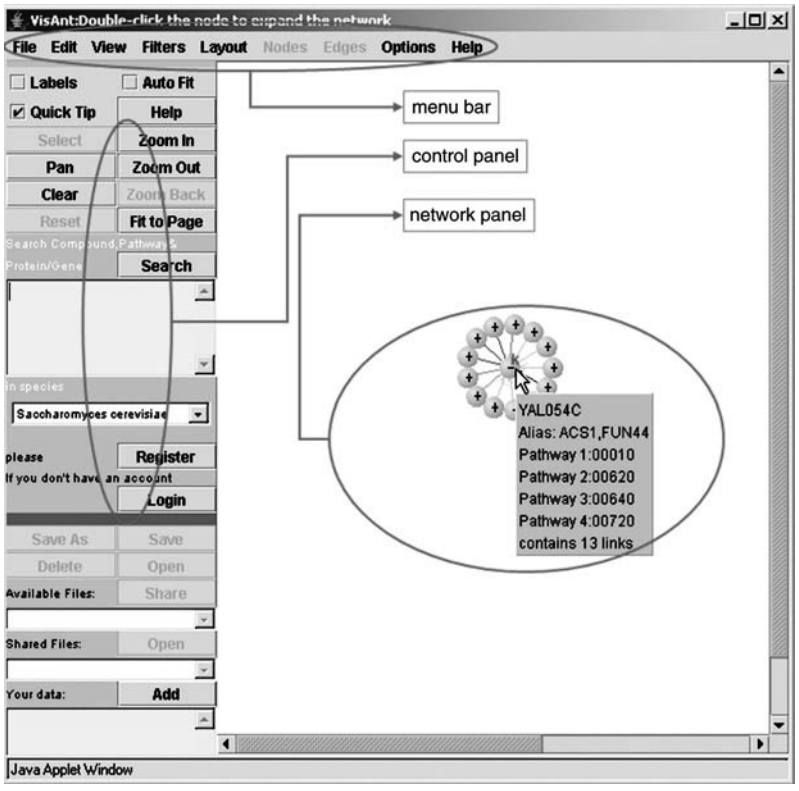


Figure 8.8.3 VisANT main window.

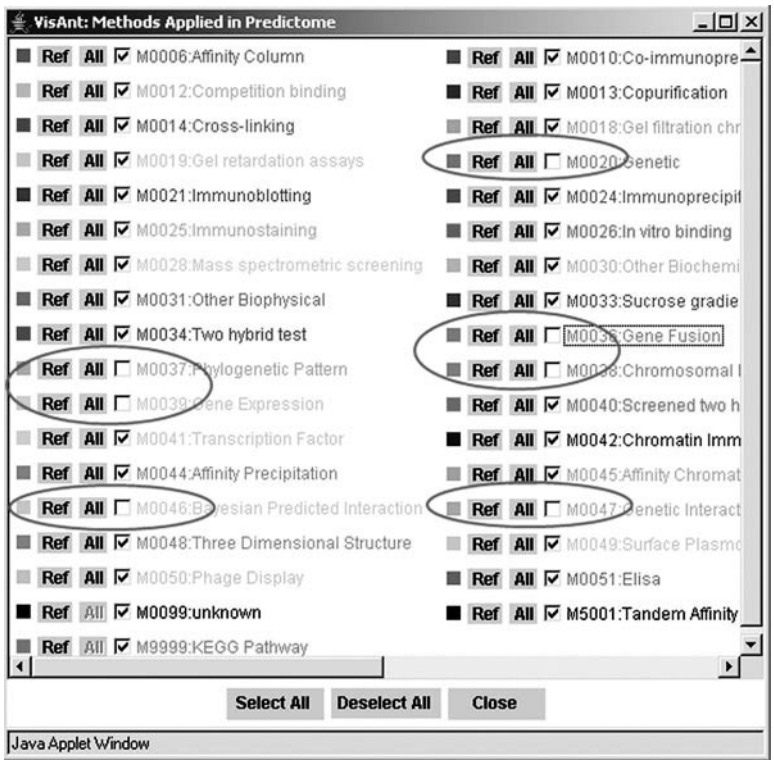


Figure 8.8.4 Methods table.

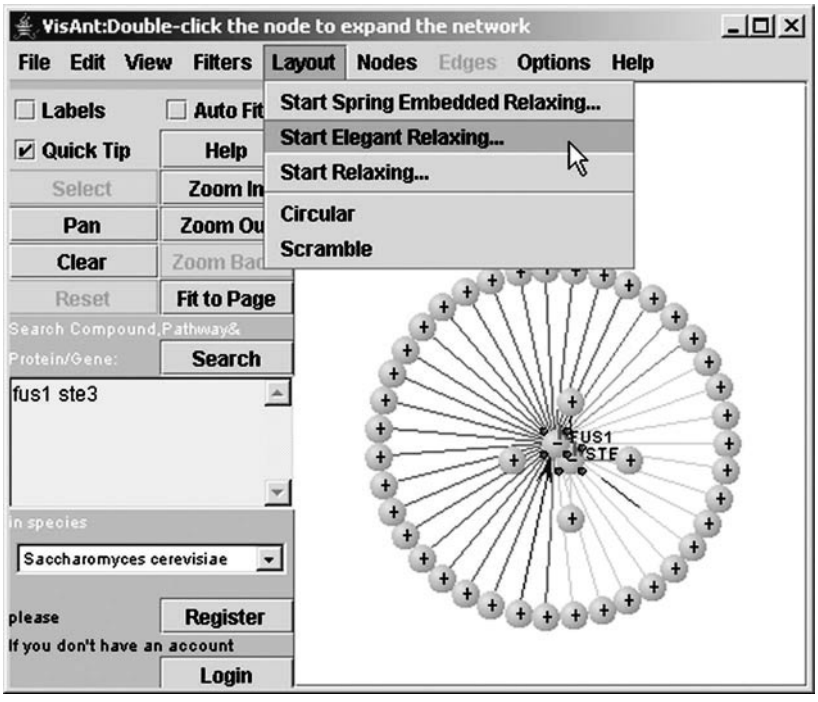


Figure 8.8.5 Searching interactions of FUS1 and STE3 proteins. The circles represent genes or proteins, depending on the assay by which the relations were obtained; the connecting lines (links) represent relations established by the selected methods. The methods table can be viewed by clicking on the View menu in the menu bar. A minus sign (−) in the node indicates that the interaction has been expanded (i.e., all links are shown) while a plus symbol (+) indicates that links remain hidden.

7. Click the Search button to start the search, which will result in VisANT displaying all proteins to which the two seeds (`FUS1` and `STE3`) are related (functionally or physically), where lines between the nodes (circles) represent associations (Fig. 8.8.5).

   When a search term is found in the Predictome database, all related information as well as its binary interactions will be returned to VisANT and displayed, with the seed node (i.e., the search term) labeled. (Fig. 8.8.5).

   The initial display, especially when many genes are requested simultaneously, is likely to be cramped. Adjust the view as described in step 8.

8. If necessary, present results more clearly by clicking Layout on the menu bar and selecting one of the network relaxation options (Fig. 8.8.5). Stop the animated layout process at any time by clicking the Stop Relaxing button.

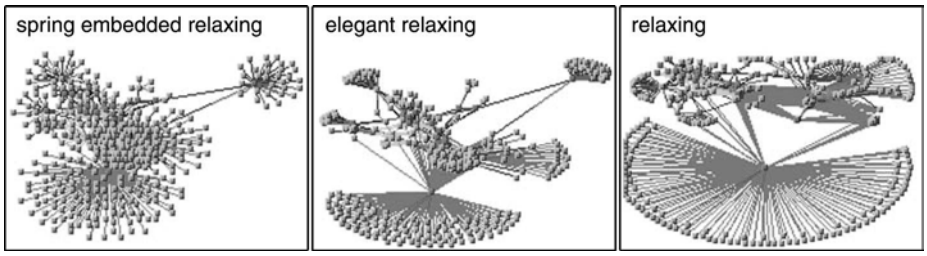


Figure 8.8.6 The difference between three spring-forces-based layout algorithms.

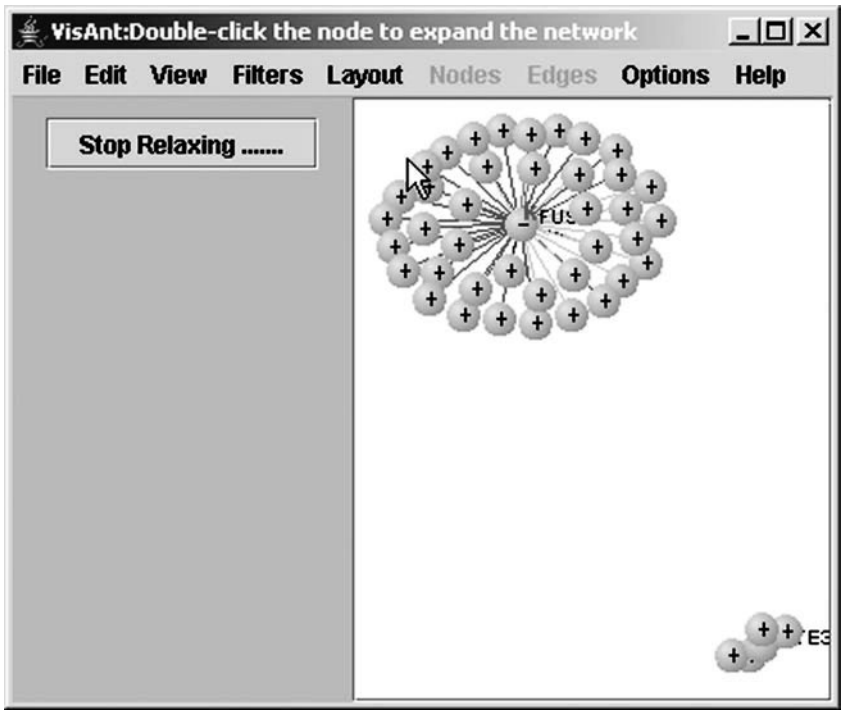


Figure 8.8.7 The result after invoking a relaxation algorithm. In this case the Elegant Relaxation algorithm was used (see descriptions below).

The layout options are designed to separate the nodes based on their connectivity. VisANT has implemented three spring-force based layout algorithms. Figure 8.8.6 shows characteristic layouts using each. The layout processes are animated and can be stopped at any time by clicking the stop relaxing button (Fig. 8.8.7).

Note that Figure 8.8.7 clearly indicates that FUS1 and STE3 proteins do not interact directly. They may, however, interact indirectly. See steps 9 and 10 for information on how to find indirect links.

9. To find indirect links, expand each node (i.e., find all nodes to which each node is linked) by dragging the mouse to draw a rectangle as shown in Figure 8.8.8. Select Query Selected from the Nodes menu in the menu bar to search the Predictome database for interacted nodes.

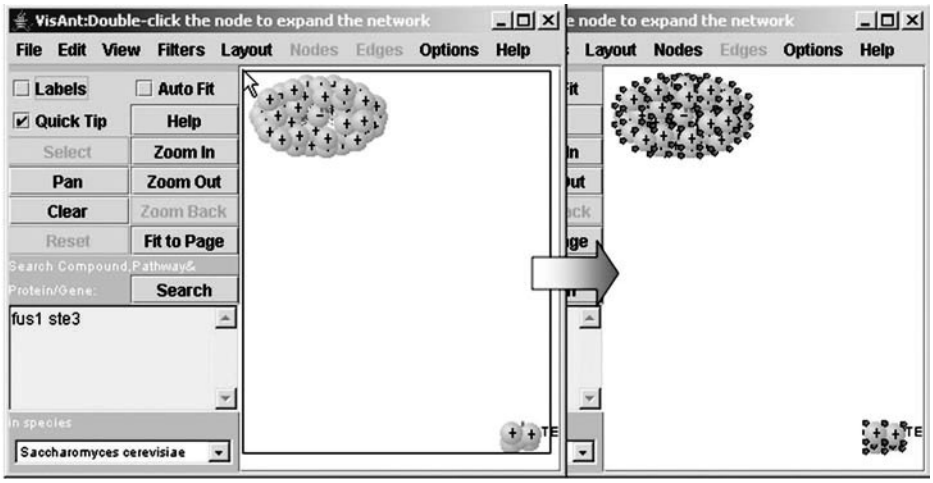


Figure 8.8.8 How to select all the nodes in the network panel. Note that selected nodes are clearly marked on the screen.

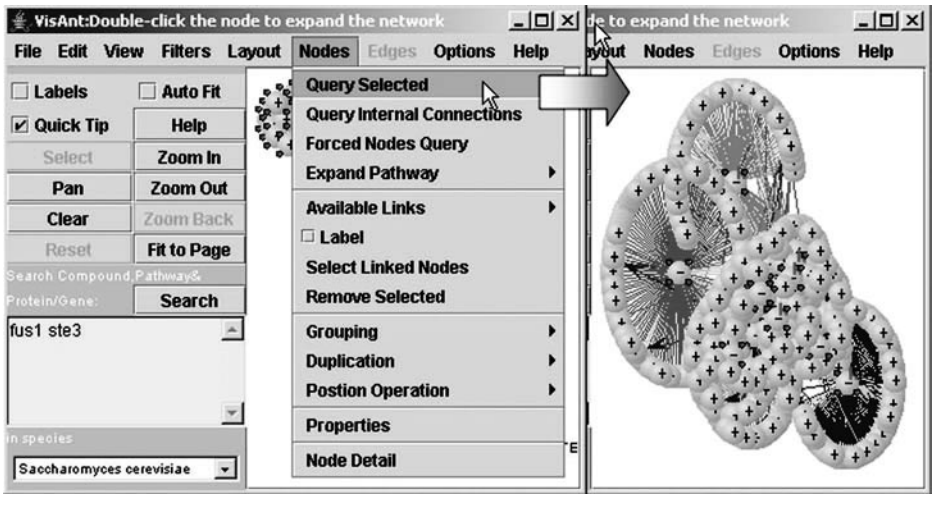


Figure 8.8.9 Querying all the nodes in the network panel.

10. Click the Fit to Page button in the control panel once the queries have been completed. If desired, double click an individual node to expand it.

    The expanded network is shown in Figure 8.8.9. Query represents the request to the Predictome database for interacted nodes and related information.

11. Relax the network (step 8) and click Zoom Out (control panel) several times. Click the Fit To Page button to stretch the network and shrink the nodes.

12. Type `STE3` and `FUS1` in the Search Compound, Pathway & Protein/Gene box of the control panel. Click on Search to highlight these two proteins in the network panel, with the usual four dot signature on the periphery indicating they are selected.

    Note that STE3 and FUS1 might still be difficult to identify at a very low level of magnification (Fig. 8.8.10).

13. Determine if there is a path between STE3 and FUS1 by selecting Find Shortest Paths Between Selected Nodes under the Filters menu in the Menu bar (see Support Protocol 1 for details).

    In this example, the results indicate that there are four shortest paths between STE3 and FUS1 (Fig. 8.8.11); connections between the two proteins are therefore verified.

    Note that the VisANT "tool-tip" can help identify nodes of interest by displaying information. For example, Fig 8.8.10 shows the mouse cursor pointing to the `STE3` *node. VisANT also allows users to add information to the tool-tip, as described (see Basic Protocol 2). However, the easiest method for determining paths is that described in the step above.*

14. Save the network as `CPBI_1` online by clicking on the Save As button in the control panel.

    See Support Protocol 2 for additional information about online network saving.

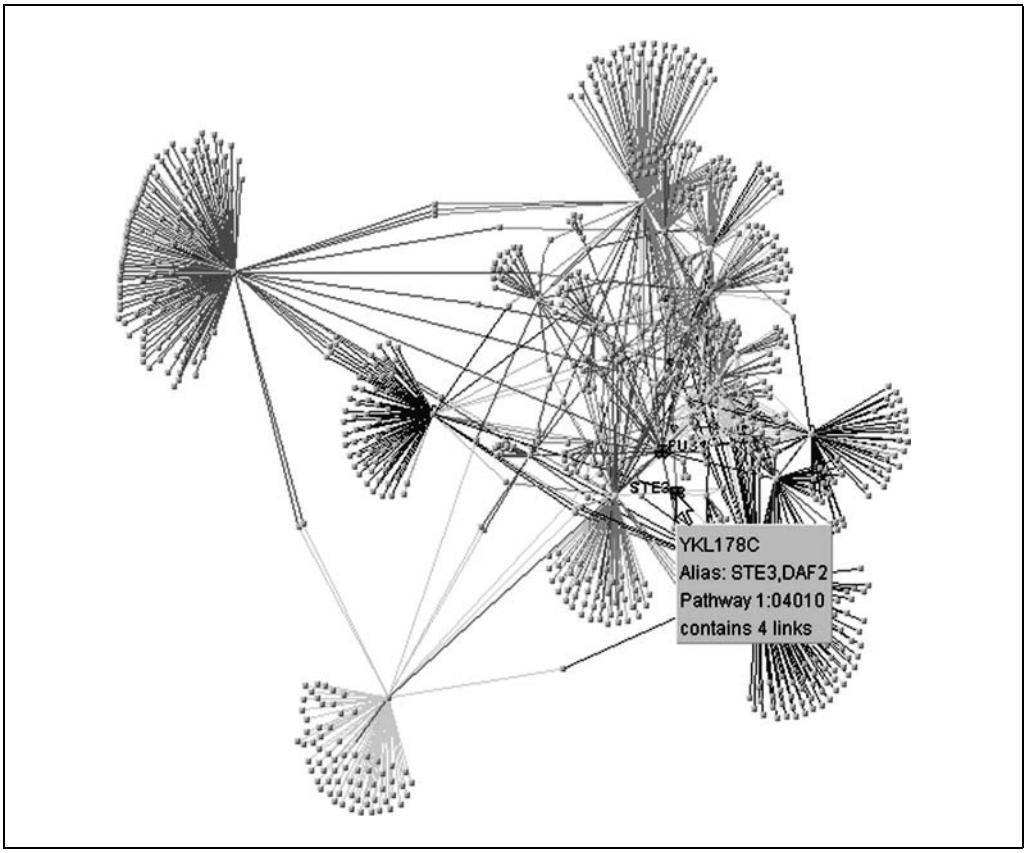


Figure 8.8.10 A low resolution view of the network that contains STE3 and FUS1.

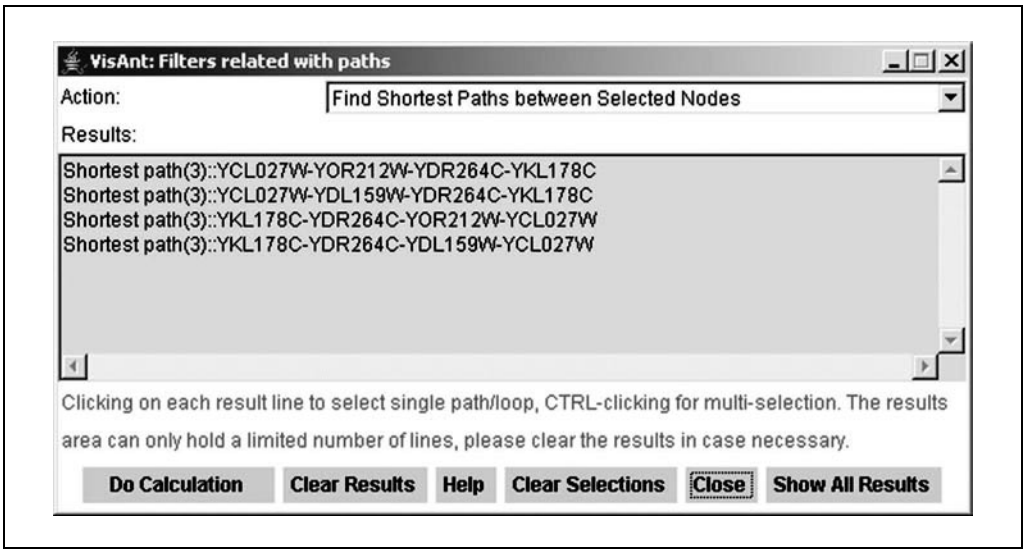


Figure 8.8.11 Shortest paths between STE3 and FUS1.

ALTERNATE PROTOCOL

CONSTRUCTING AND COMPARING LARGE-SCALE NETWORKS

To facilitate large scale analysis of interaction networks, VisANT enables method-based quick load of large interaction data sets. The following example illustrates the simultaneous use of physical protein-protein interaction (PPI) data based on yeast-two hybrid experiments (Uetz et al., 2000), and synthetic genetic array data (Tong et al., 2001) for *S. cerevisae*.

Necessary Resources

Hardware

Any computer with Internet access

Software

Java compatible browser
Java Run-time Environment (JRE) 1.4 or above (see Internet Resources)

Files

None

Select method

1. Start the browser, open the VisANT start page (*http://visant.bu.edu*), and click the start button as described (see Basic Protocol 1, steps 1 and 2).

   Remember that the start page must be kept open during all procedures.

2. Clear the network panel by clicking the Clear button in the control panel.

3. Invoke the Methods Table (see Basic Protocol 1, step 6), selecting method 34 (`M0034`; yeast two hybrid; Fig. 8.8.4). Click All to load all interactions obtained by this method.

   Figure 8.8.12 shows the interactions laid out with the circular layout algorithm. The dense field of blue results from the large number of connections between nodes. The green around the periphery are nodes which in this view are too small to resolve, and the jagged blue edge results from self correlated nodes. The Ref button of method 34 can be used to access references for individual interactions.

4. Click on all of method M0047 to load synthetic genetic array data.

   The combined network is shown in Figure 8.8.13.

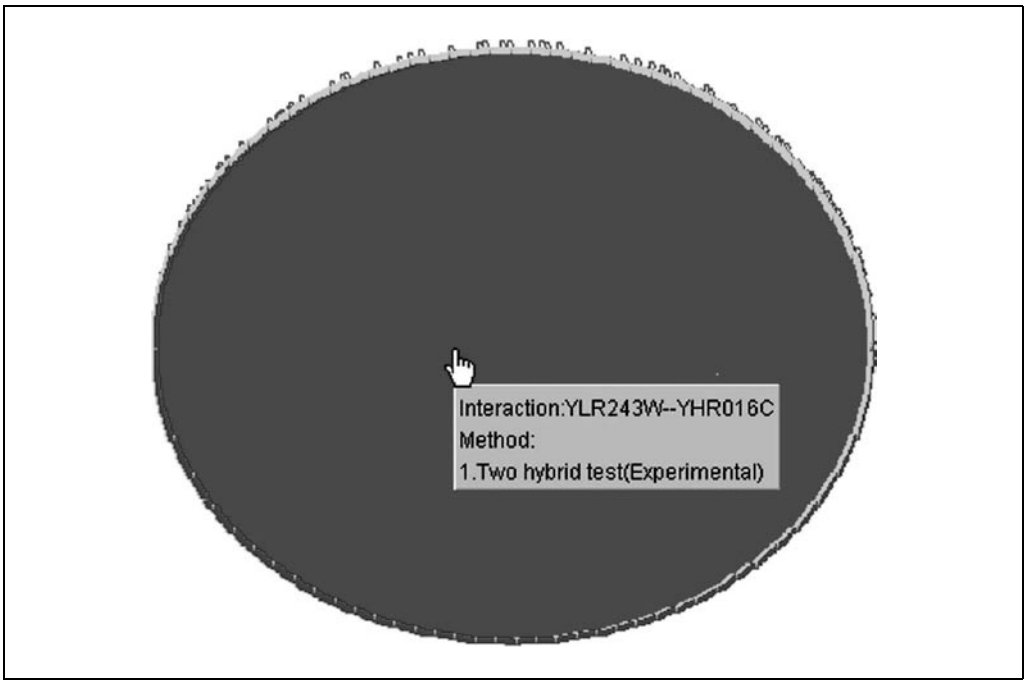


Figure 8.8.12 PPI network (yeast two hybrid) of *S. cerevisiae*. This black and white facsimile of the figure is intended only as a placeholder; for full-color version of figure go to *http://www.interscience.wiley.com/c_p/colorfigures.htm*.

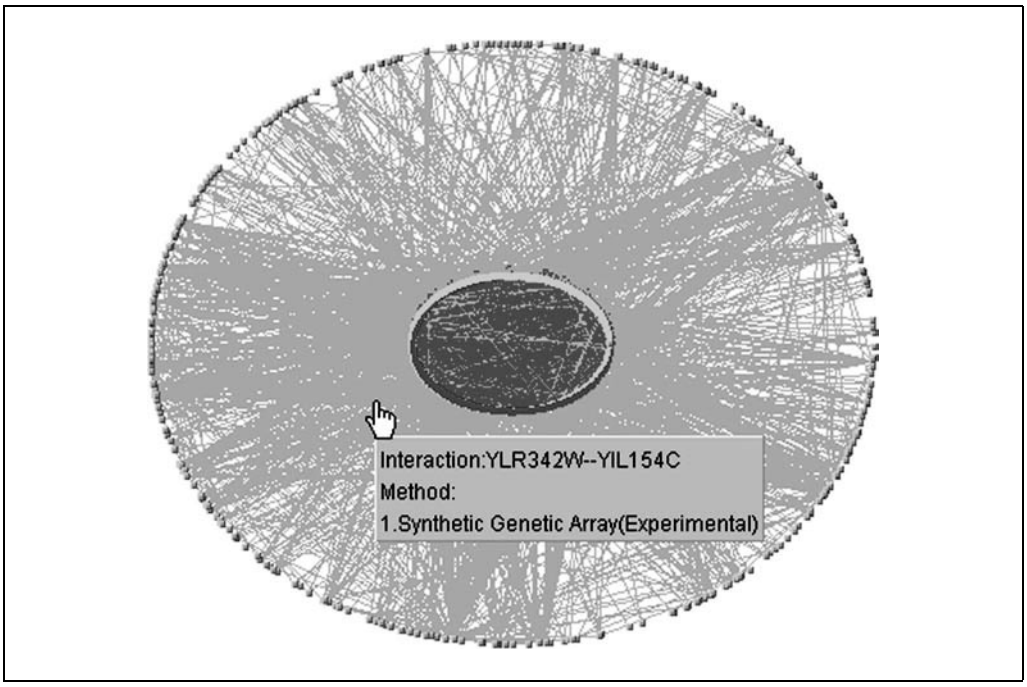


Figure 8.8.13 Combined network of PPI (blue region) and genetic network (green) for *S. cerevisiae*. This black and white facsimile of the figure is intended only as a placeholder; for full-color version of figure go to *http://www.interscience.wiley.com/c_p/colorfigures.htm*.

5. Ensure that pop-up blocking functions, such as those invoked by Google, are turned off. Invoke Statistics Report under the View menu in the menu bar (Fig. 8.8.3), which will cause a new browser window to appear as shown in Figure 8.8.14.

   Figure 8.8.14 shows that there is very little intersection between the two networks. In particular there are only six overlaps (the edges associated with both methods) between 3627 genetic and 6445 physical interactions.

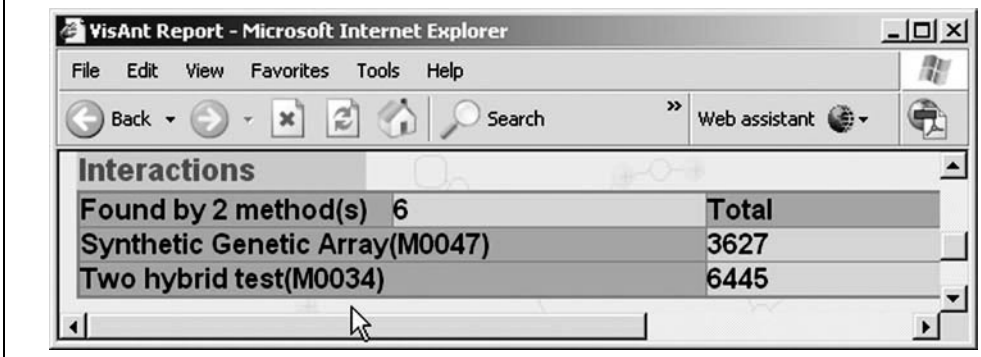


Figure 8.8.14 Status report of the combined network

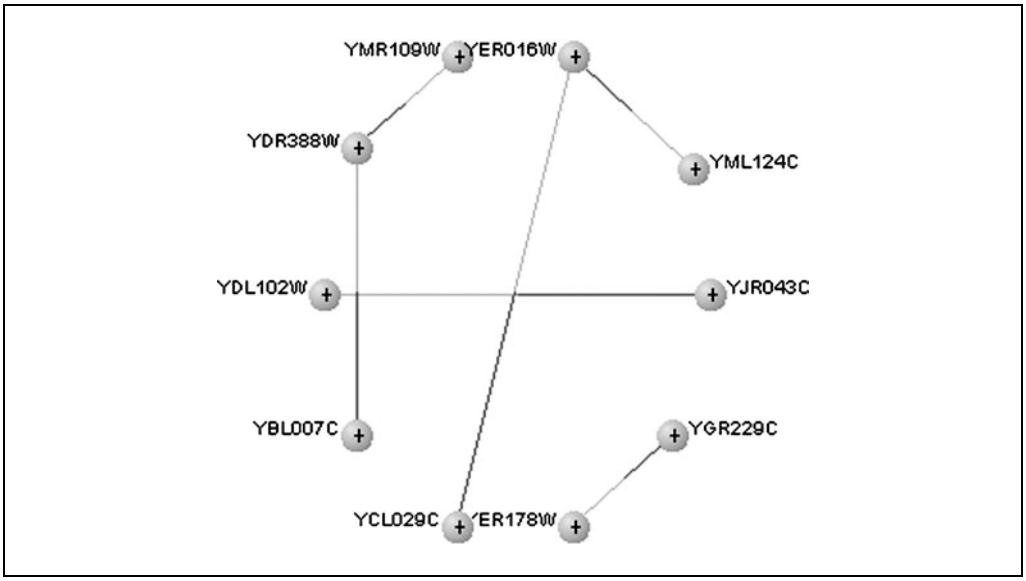


Figure 8.8.15 The intersection of the combined network. Each edge is labeled with two colors, indicating that the association is obtained by two methods. This black and white facsimile of the figure is intended only as a placeholder; for full-color version of figure go to *http://www.interscience.wiley.com/c_p/colorfigures.htm*.

Determine common nodes

6. Invoke Select N(odes)&E(dges) With Edge Discovered By Multiple Methods under menu Filters in the menu bar, which will cause the common edges to be selected.
7. Invoke Reverse Selection under the Edit menu in the menu bar, which will cause the selection to become reversed so that nodes and edges that are not identified by both methods will be selected.
8. Invoke Remove Selected under the Nodes menu to remove all nodes that are not common to the two networks.
9. Click the Zoom Out and then Reset buttons in the control panel to restore the remaining nodes to their original size.
10. Remove or hide edges that are not determined by both methods by invoking Hide Link under the Edges menu.

The final network is shown in Figure 8.8.15.

QUANTITATIVE CHARACTERISTICS OF NETWORK TOPOLOGIES

Biological networks typically consist of one or more significantly overrepresented motifs. For example, feed-forward loops are common in yeast and *E. coli*. At present, VisANT identifies feed-forward motifs and cycles (feedback). In addition, options are available for compiling statistics on various network characteristics including shortest paths between nodes, number of links per node (degree distribution), and the average path length between nodes. The following protocols demonstrate VisANT functions related to transcription factor/target networks.

Necessary Resources

Hardware

Any computer with Internet access

Software

Java compatible browser
Java Run-time Environment (JRE) 1.4 or above (see Internet Resources)

Files

None

1. Start the browser, open the VisANT start page (*http://visant.bu.edu*), and click the start button as described (see Basic Protocol 1, steps 1 and 2).

   Remember that the start page must be kept open during all procedures.

Display distribution of edges per node

2. Load transcription factor-target pairs determined by chromatin immunoprecipitation (ChIP; Lee et al., 2002) using the same procedure as described for yeast two hybrid (see Alternate Protocol, steps 2 and 3), except substituting method 42 (M0042) for method 34 (M0024; Fig. 8.8.4).

3. View the degree distribution of the network by invoking Degree Distribution under the View menu in the menu bar, which will cause a window showing the distribution to appear (Fig. 8.8.16). Make sure Log Plot is checked.

   The number of edges per node is measured along the horizontal axis, while the corresponding number of nodes is measured along the vertical axis. The equation at the upper right is the power law that best fits the observations.

Cycle detection

4. Close the Degree Distribution window.

5. Ensure there is no selected node in Network Pane l, as otherwise VisANT will only show the detected cycles that include at least one selected node. To deselect nodes, click on any empty position in the network panel.

6. Invoke Find Cycles (i.e., feedback loops) under Filters in the menu bar (Fig. 8.8.3), which will cause a cycle with three nodes to be selected.

   A cycle is defined as a closed unidirectional path of the network. Here, the unidirectional path only requires that each edge of the path has the corresponding direction of the cycle, which means that if an edge is bidirectional, then this edge can always be in the cycle. VisANT supports hybrid networks, which can either be directional or directionless, and a directionless edge is treated as bidirectional when performing topological analysis.

7. To isolate the cycle, first reverse the selection by invoking Reverse Selection under Edit in the menu bar and then invoking Remove Selected under Nodes in the menu bar to remove the selected nodes.

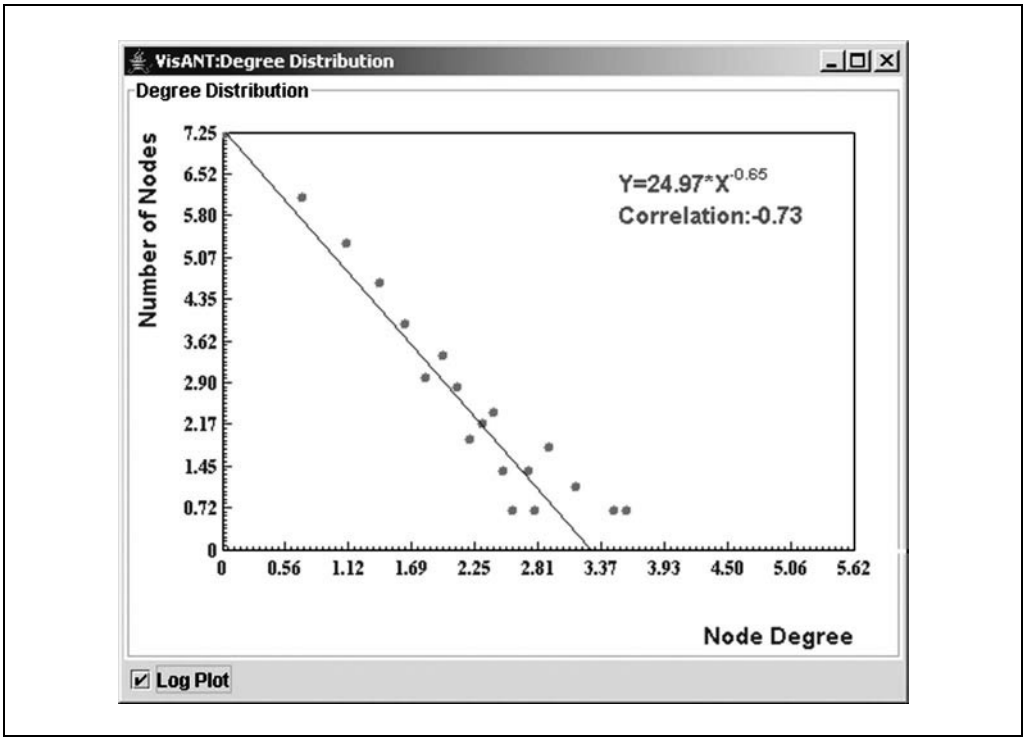


Figure 8.8.16 Degree distribution of regulatory network (ChIP).

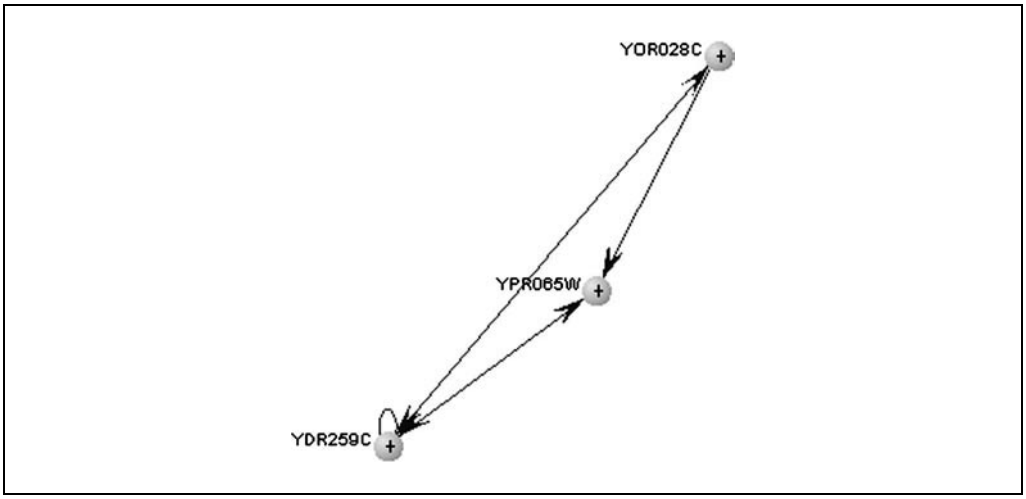


Figure 8.8.17 Feedback loop retrieved from a complex transcription-factor/target network.

8. Click Zoom Out in the control panel and then click the Reset button to restore nodes to normal size. Turn the label on and click the Fit To Page button in the control panel to obtain the cycle as shown in Fig. 8.8.17.

   This is the feedback loop described in Lee et al. (2002).

Shortest paths

9. Clear the network panel.

10. Add the following lines into the Your Data text field in the control panel (Fig. 8.8.3), using tabs to separate each component and a hard return to complete each line (i.e., tab-delimited format):

```
YNL325C  YLR452C 1  M0041
YOR212W  YDL230W 0  M0039
YNL325C  YHR005C 0  M0039
YNL325C  YNL128W 0  M0039
YOR212W  YLR452C 0  M0039
YLR452C  YHR005C 0  M0039
YPR165W  YDL230W 0  M0039
YPR165W  YLR452C 0  M0039
YOR212W  YHR005C 0  M0039
YOR212W  YHR005C 0  M0040
YHR005C  YHR005C 0  M0039
```

Each line represents a binary interaction between node 1 (first column) and node 2 (second column). The integer in the third column represents the direction of the interaction, i.e., 0 signifies an undirected link, 1 indicates that the link has a direction from node 1 to node 2, and −1 indicates that the link has a reverse direction from node 1 to node 2. (In the near future, there will be many different types of directions indicating different types of biological relationship; refer to the VisANT user manual for more detail.) The last column represents the associated method (method ID) of this interaction. For example, in the first line given above, node 1 is `YNL325C`*, node 2 is* `YLR452C`*, the direction is* `1`*, and the method ID is* `M0041`*. Therefore,* `YNL325C` *binds* `YLR452C`*, because* `M0041` *represents gene regulation.*

The method ID represents the method used to uncover the interaction/association, and allows the biological interpretation of the edge (interaction/association). For example, a directed edge from node 1 to node 2 with method ID of M0039 can be interpreted as node 1 activates node 2, because M0039 represents gene expression. On the other hand,

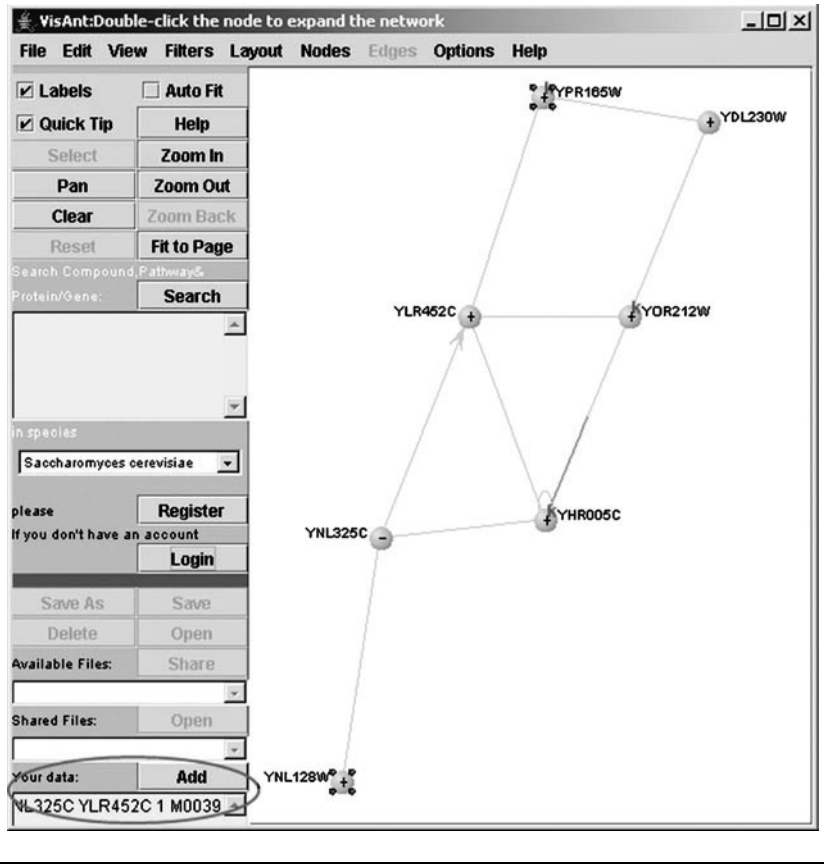


Figure 8.8.18 An example of shortest path detection.

the same edge can be interpreted as indicating that the protein of node 1 binds the gene of node 2 (i.e., node 1 is a transcription factor of node 2) if the method is M0041, because M0041 represents the transcription factors of gene regulation from TRANSFAC database (http://www.gene-regulation.com).

Note that columns 3 and 4 are optional. The default value of the direction is `0` *and the default method ID is* `M9999`*, which indicates that the method is unknown.*

11. Click the Add button to display the network data. Invoke Start Spring Embedded Relaxing. . . under the Layout menu to make the network similar to the one shown in Figure 8.8.18. Check Labels and deselect all nodes by clicking on any empty location in the network panel.
12. Select the nodes YNL128W and YPR165W by holding down the control key and clicking on them.
13. Invoke Find Shortest Paths between Selected Nodes under Filters to obtain the following paths.

```
Shortest path(3)::YNL128W-YNL325C>YLR452C-YPR165W

Shortest path(4)::YPR165W-YLR452C-YHR005C-YNL325C-
YNL128W
```

Because the edge from `YNL325C` *to* `YLR452C` *is directional, the shortest path from* `YNL128W` *to* `YPR165W` *is not same as the one from* `YPR165W` *to* `YNL128W`*. If more than two nodes are selected, VisANT will exhaustively search for the shortest paths between all pairs.*

SUPPORT PROTOCOL 2

ONLINE SAVING AND READING OF THE NETWORK

VisANT provides online saving, reading, and sharing functions. Data security necessitates registration by researchers wishing to use these capabilities; however, the only required information for registration is an email address. The registration can be started by clicking the Register button in the control panel or by visiting following *http://visant.bu.edu:8080/test/register.jsp*.

Additional information can be found in the VisANT user manual *http://visant.bu.edu/vmanual*. In the near future VisANT will be enabled to save the network file to a local disk by using either the sign VisANT applet or allowing VisANT to start through Java Web Start.

Necessary Resources

Hardware

Any computer with Internet access

Software

Java compatible browser
Java Run-time Environment (JRE) 1.4 or above (see Internet Resources)

Files

None

1. Clear the network panel, change species to *Homo sapiens*, and search for `P53` without deselecting any methods (see Basic Protocol 1, steps 1 to 4). Select and query all nodes in the network panel (see Basic Protocol 1, steps 9 to 11).
2. Login to VisANT by clicking the Login button in the control panel.

3. Click the Save As button in the control panel to save the network using the file name `CPBI_2`, which will cause the file to be saved to the VisANT application server through the network with `CPBI_2` listed in the Available Files drop-down list in the control panel.

   The file is stored in the VisANT application server. Storage is limited to ten files per user. The global font size and the status of the three checkboxes in the control panel are not stored in the network file. All other information, including customized annotation, is stored in the file. If files have been saved previously, they will be shown in the drop-down list named Available Files in the control panel.

4. *Share a network file.* Select `CPBI_2` from the Available Files drop-down list and click the Share button. In the first text box in the window, enter the email addresses of the users with whom the file is to be shared: in this example, the user's own. Click the OK button.

   A network must be saved before it can be shared with other users. The network can be shared with any users, irrespective of whether they are registered with VisANT. Please refer to the VisANT user's manual for more information.

5. Open the file that has been shared. Log out of VisANT by clicking the Logout button in the control panel and then log in again. Note the emailed file is shown in the drop-down list named Shared Files in the control panel (Fig. 8.8.3).

6. Select the file from the drop-down list and click the Open button just above it to open the shared file.

   Once the shared file is opened, it will no long appear in the list of Shared Files and will be deleted upon logging out unless it is saved.

ANALYZING THE BIOLOGICAL NETWORK

BASIC PROTOCOL 2

Here the network utilized above (see Basic Protocol 1) is again processed, but this time it is pruned by using physical links only.

Necessary Resources

Hardware

Any computer with Internet access

Software

Java compatible browser
Java Run-time Environment (JRE) 1.4 or above (see Internet Resources)

Files

`CPBI_1` saved on-line (see Basic Protocol 1)

1. Start VisANT, login, and load saved file `CPBI_1`.

   Refer to Support Protocol 2 for detailed instructions on loading the saved network.

2. Filter out computational and genetic interactions by clicking on (and therefore removing) the checks in the Methods Table (Fig. 8.8.4) for methods M0020, M0036, M0037, M0038, M0039, M0046, and M0047.

   Methods M0020 and M0047 represent edges of genetic association based on knockout experiments. Methods M0036, M0037, M0038, M0039, and M0046 represent those edges of functional association predicted computationally.

3. Close the Methods Table. Layout the network and save it as `CPBI_4`.

   The resulting network is shown in Figure 8.8.19.

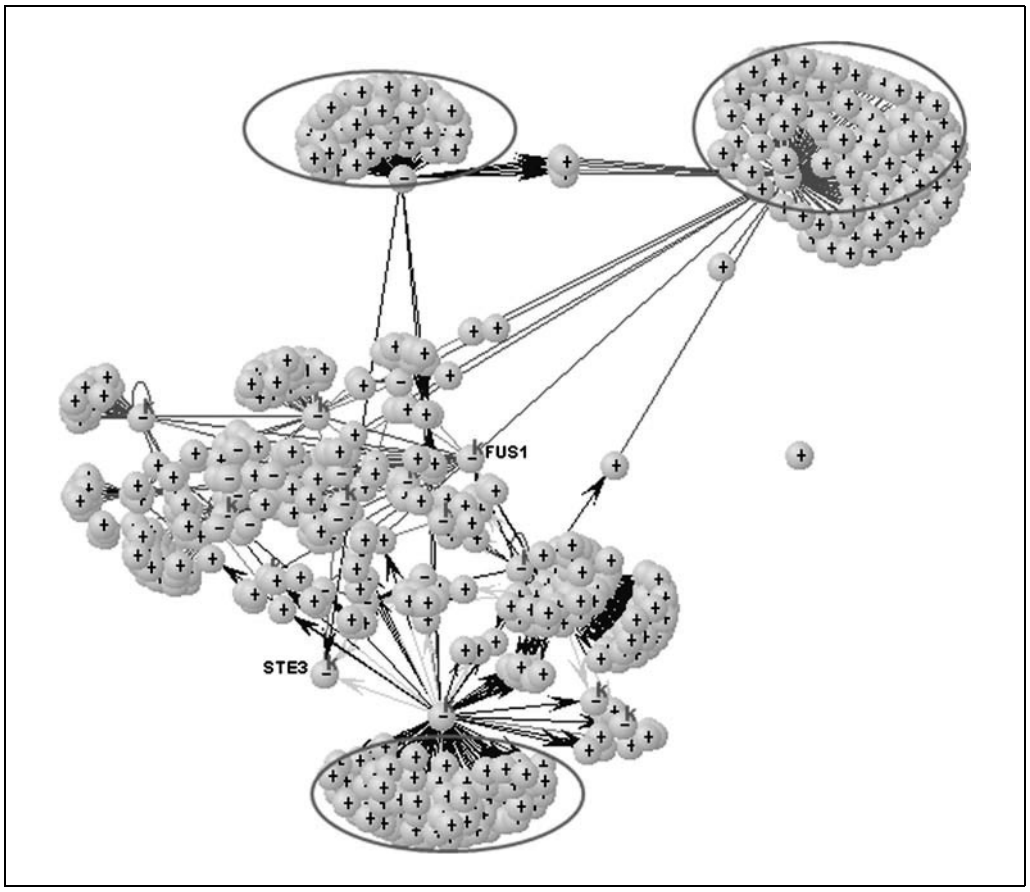


Figure 8.8.19 The network of physical interactions within which STE3 and FUS1 are embedded.

Simplify the network by removing unwanted edges

4. Turn on the node label by checking the Labels checkbox in the control panel. Double click on following nodes to hide their connections:

   YDR310C, YMR047C, YMR043W, YER118C, YPL049C, YHR084W, YER032W, YFL026W, YKL209C, YDR461W, YFL039C, YNL058C, YBL105C, YLR117C, YNL271C, YHR158C, YAL017W, YCR040W, YCL055W, YNR044W, YER149C, YKR101W, YIL159W, YOR212W, YHR005C, YBL016W, YDR264C, YNL054W, YPR008W, YOR017W, YGR040W, YLR452C, YMR232W, YAL031C, YDL159W, YKL092C.

   If there is any difficulty finding the nodes, copy them to the Search Compound, Pathway & Protein/Gene Name box and click the Search button to locate them. Layout the network (see Basic Protocol 1, step 8), to produce a network similar to the one shown in Figure 8.8.20.

   Clusters such as those circled in Figure 8.8.19, obviously have no contribution and can either be removed or hidden. In this example the edges are hidden rather than eliminated.

5. Select all nodes to the left of FUS1 and remove them by invoking the Remove Selected under the Nodes in the menu bar (Fig. 8.8.3).

6. Move the node around to make the network similar to the one shown in Figure 8.8.21, which will allow the connectivity between FUS1 and STE3 to be visually examined.

   Network node properties and appearance can be readily changed.

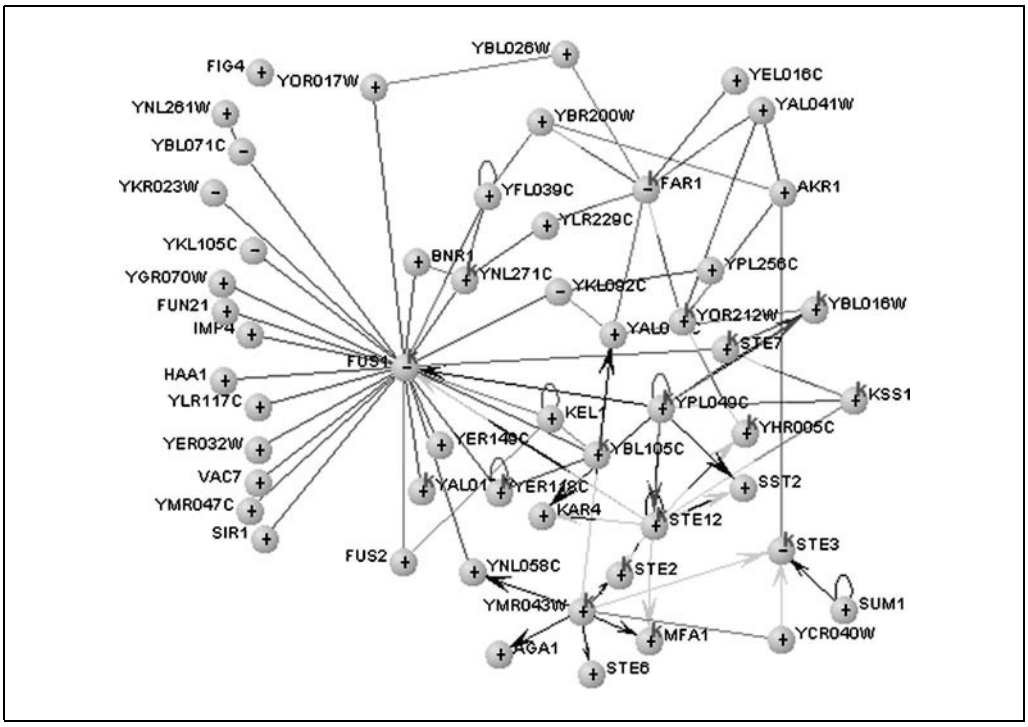


Figure 8.8.20 Network after collapse of a set of nodes.

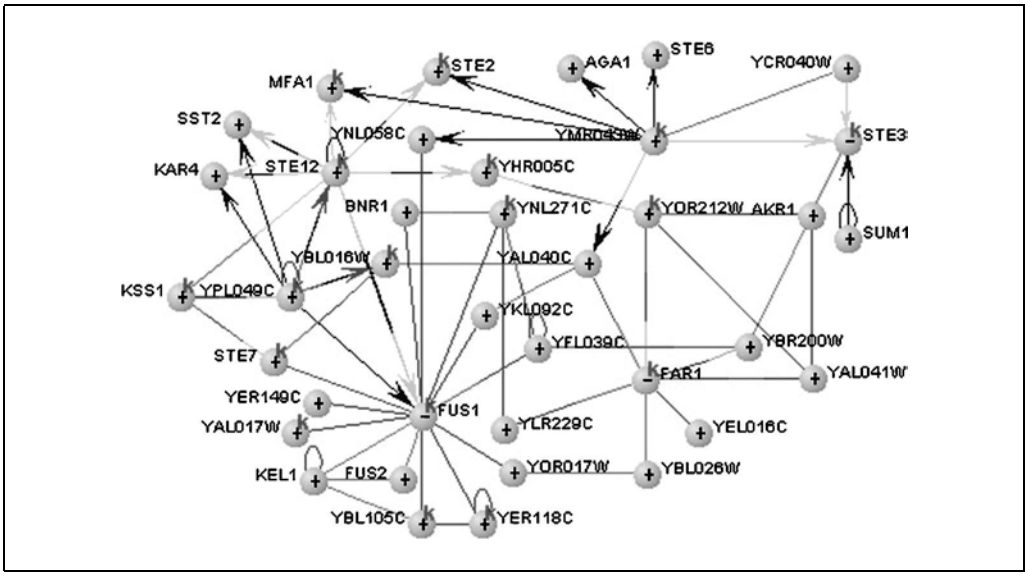


Figure 8.8.21 The pruned physical interaction network containing FUS1 and STE3.

7. Select both FUS1 and STE3 and invoke the node property window by selecting Properties under the Nodes menu. Change node Size to 27 and change the label Position to Center (Fig. 8.8.22).

   Once a node's properties are specified, they will not change even if there is the global change of the network, such as zoom in/out.

8. Detect the shortest path between FUS1 and STE3 (see Support Protocol 1), which will cause all nodes on the shortest path to be selected.

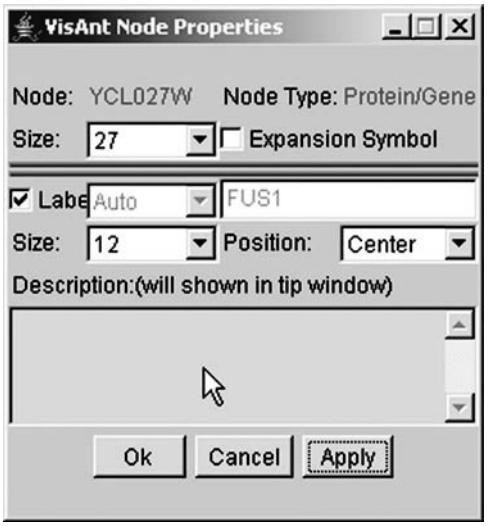


Figure 8.8.22 Node Properties window.

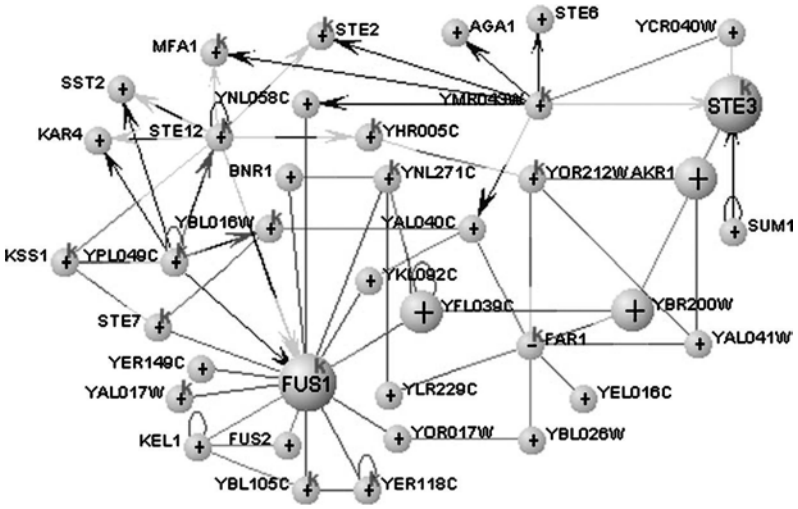


Figure 8.8.23 Network with annotated shortest path between FUS1 and STE3.

9. Press the `CTRL` key and use the mouse to click on STE3 and FUS1 to deselect the two nodes while leaving other nodes unmodified. Invoke the node property window again and set node size to `20`.

   The resulting network is shown in Figure 8.8.23, with the shortest path clearly distinguished.

10. Activate the SGD URL link for FUS1 using the Available Links option in the Node menu. Copy the SGD description of FUS1.

    VisANT has been integrated with many different databases. For the yeast genome, the most frequently used data source for functional annotation is the Sacharomyces Genome Database (SGD).

11. Open the property window of FUS1 (see step 7). Paste the SGD description into the Description field of its property window (Fig. 8.8.22), which will cause the functional annotation to become part of FUS1's tool-tip (Fig. 8.8.24).

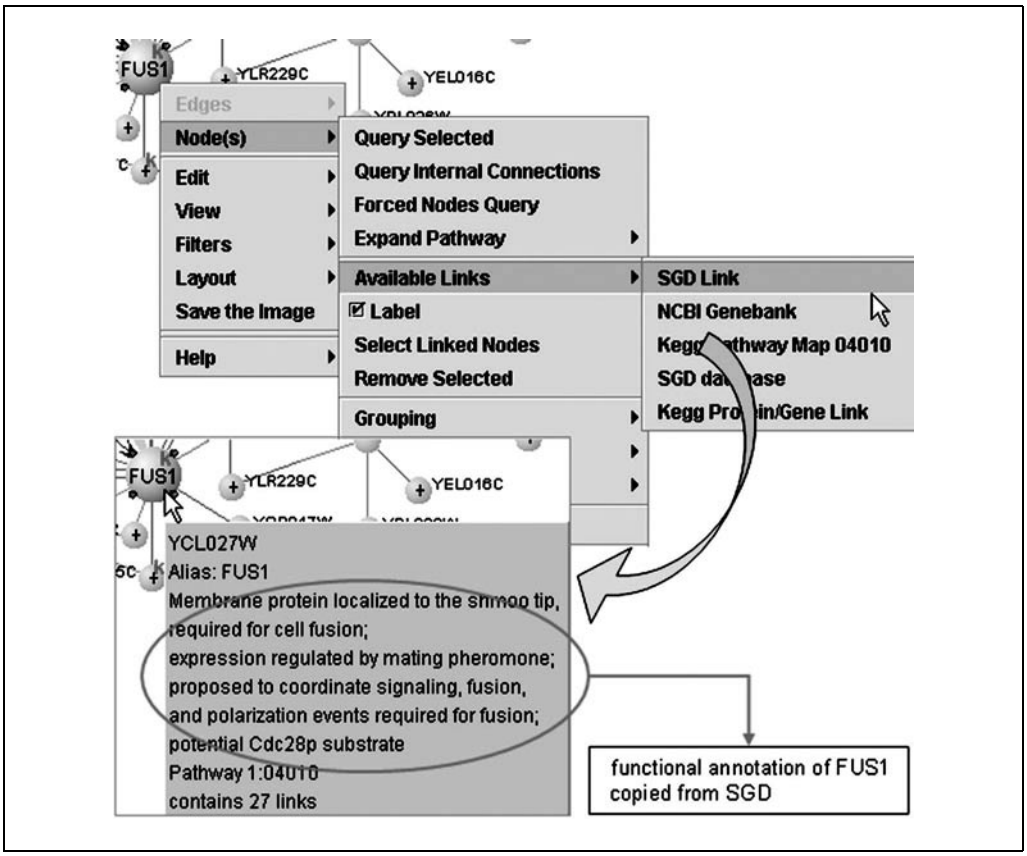


Figure 8.8.24 Adding node annotation from a linked data source.

META-NETWORKS: AN APPLICATION TO PROTEIN COMPLEXES

BASIC PROTOCOL 3

Networks can generally be decomposed into collections of dense subnets (Fig. 8.8.19). When referring to a network of nodes, each of which represents a subnetwork, the term "meta-network" is often used. This representation corresponds to functional organization of the cell as a network of molecular motifs and leads naturally to a hierarchical organization—i.e., there are multiple levels of meta-networks. Here, only one of a particular type, viz molecular complexes, is discussed (Gavin et al., 2002). Information about meta-network implementation in VisANT can be found in the user's manual.

Necessary Resources

Hardware

Any computer with Internet access

Software

Java compatible browser
Java Run-time Environment (JRE) 1.4 or above (see Internet Resources)

Files

None

1. Start a browser and open the VisANT start page (*http://visant.bu.edu*). If the Start button in the WEB page (Fig. 8.8.2) is not visible, follow the instructions in the VisANT user's manual *http://visant.bu.edu/vmanual* to install the required software (JRE).

2. Click the Start button, which will cause a VisANT window, having three main components (menu bar, control panel, and network panel; Fig. 8.8.3), to appear. Keep the start page open during all procedures.

3. Clear the network panel by clicking the Clear button in the control panel.

4. Do not deselect any methods (Fig. 8.8.4).

5. Click All next to method M5001 to load all proteins studied by tandem affinity mass spectrometry. Relax the network by clicking Layout on the menu bar and selecting one of the relaxation options. Press the Stop Relaxing button to terminate the process (also see Basic Protocol 1, step 8).

   The result should be similar to that shown in Figure 8.8.25A. The gray edge between two complexes indicates that there is at least one protein shared by both complexes. If the number of shared proteins is greater than one, the number will be shown along the edge.

6. Click on Degree Distribution in the View menu.

   The result (Fig. 8.8.26) should indicate that in this case the network does not obey a power law.

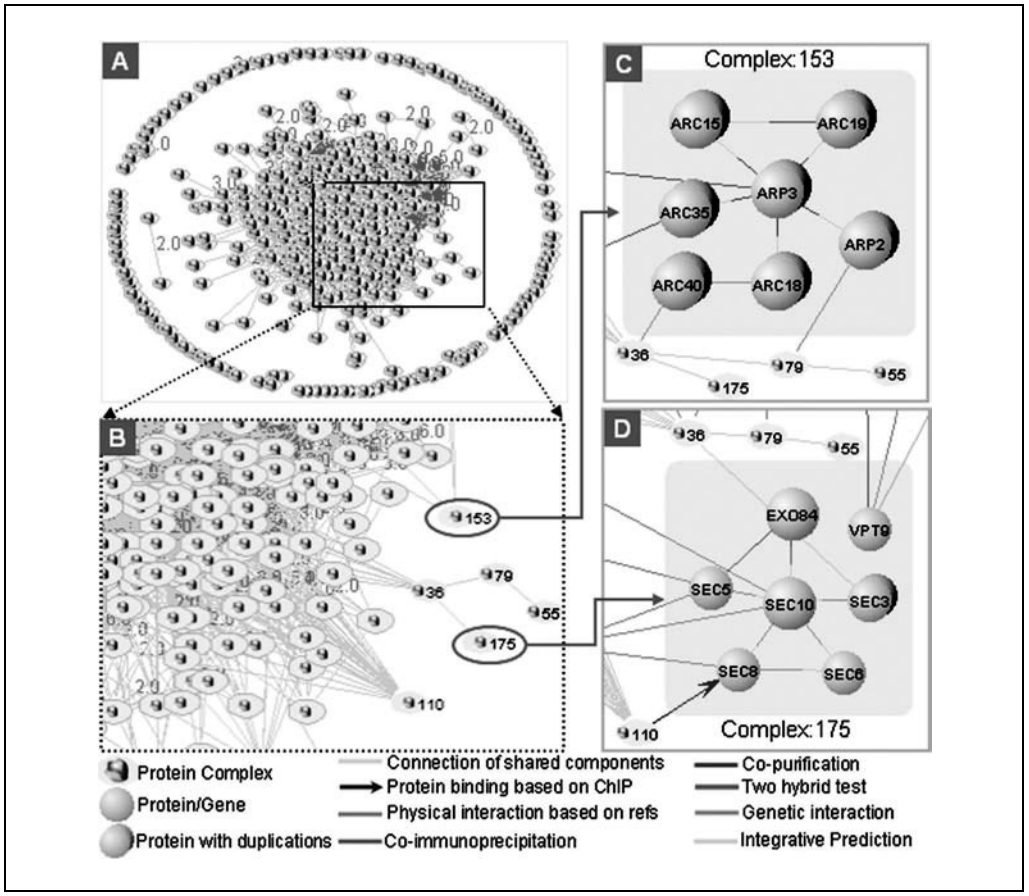


Figure 8.8.25 Integration of different data sources. Complexes (meta-nodes) were determined by tandem affinity mass spectrometry (Gavin et al., 2002); the internal connections were determined by a variety of methods as indicated. (**A**) Network of protein complex after it has been laid out. The rectangle represents the region of interest for zoom-in. (**B**) The region of interest of the network after zoom-in, with several complexes labeled according to its original reference. (**C**) Internal network structure of Complex 153 after integration with the interaction data from the Predictome database. All nodes are connected. (**D**) Internal network structure of Complex 175 after integration with the data from Predictome database. This black and white facsimile of the figure is intended only as a placeholder; for full-color version of figure go to *http://www.interscience.wiley.com/c_p/colorfigures.htm*.

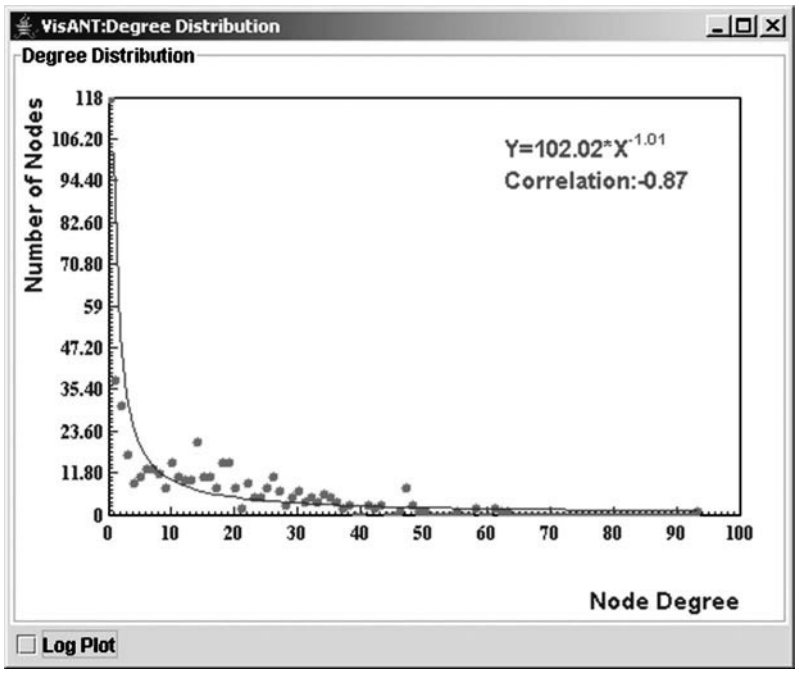


Figure 8.8.26 Degree distribution of complex network: the power-law does not hold.

7. Zoom in on the region shown in Figure 8.8.25A by holding down the left mouse button to drag a rectangle over the area of interest.

   The zoomed network is shown in Figure 8.8.25B.

8. Search complex M5001:175 and M5001:153. Label the complexes as shown in Figure 8.8.25B.

   In the original data source (Gavin et al., 2002), the protein complexes are only numbered. To make sure that the names of the complex are unique, the authors prefix the method name with the numbers.

9. Filter out computational and genetic interactions by clicking on (and therefore removing) the checks in the Methods Table (Fig. 8.8.4) for methods M0020, M0036, M0037, M0038, M0039, M0046, and M0047.

   Methods M0020 and M0047 represent edges of genetic association based on knockout experiments. Methods M0036, M0037, M0038, M0039, and M0046 represent those edges of functional association predicted computationally.

10. Close the Methods Table. Layout the network and save it as `CPBI_complex`.

11. Double click complex 153, which will cause the complex to expand, revealing that it consists of seven proteins. Select all seven proteins and invoke Query Selected under the Nodes menu.

12. Select properties under the Nodes menu. Alter the parameters to make it look similar to the network shown in Figure 8.8.25C.

    See Basic Protocol 2, step 7 for more information on node properties.

    SGD indicates that this seven-member complex is a stable cytoskeleton-associated assembly, Arp2/3, required for the nucleation of actin filaments in all eukaryotic cells.

13. Repeat steps 7 to 10 for complex 175 and its internal connections (Fig. 8.8.25D).

SGD indicates that complex 175 is a set of proteins required for polarized exocytosis and cell separation in eukaryotic cells (Wang et al., 2002). The Exocyst complex has been well studied and is known to contain single copies of eight subunits: Sec3, Sec5, Sec6, Sec8, Sec10, Sec15, Exo70, and Exo84. Exo70 and Sec15 are apparently missed by mass spectrometry (Gavin et al., 2002), while Vpt9 had not been found connected to above subunits prior to the experiments of Gavin et al. (2002). On the other hand, it is interesting to note that all sec proteins are genetically connected, which is not surprising because the components of the Exocyst were initially identified as products of sec genes in Yeast. The topology of the internal network suggests that Sec10 may play a key role.

COMMENTARY

Background Information

The VisANT tool for network visualization and analysis is a flexible web-enabled program for quick and simple manipulation of biological interaction data. Biological interaction and network data can be derived from any method that detects associations between genes, proteins, or other biomolecules. As broad categories, some methods are experimental (e.g., yeast two hybrid, ChIP), while others are more computational and predictive of functional information (e.g., sequence similarity). As a network tool, VisANT enables users to manipulate and annotate bionetworks and pathways in a cohesive graphical interface with the goal of facilitating annotation and layering of user-defined information.

VisANT is accessible from any recent Java-enabled web browser on any platform. It supports a growing number of standard exchange formats and database referencing standards, such as KEGG/KGML (Kanehisa et al., 2002), Proteomics Standards Initiative (PSI; Hermjakob et al., 2004), BioPAX (in progress), GenBank (Benson et al., 2003), and the Gene Ontology (Ashburner et al., 2000). Multiple species are supported, to the extent that computed or experimental evidence of interactions or associations are available in public datasets or the Predictome database (Mellor et al., 2002).

Predictome includes relations collected from other interaction databases (such as BIND; Bader et al., 2003) and MIPS (Mewes et al., 2000), from large scale studies based on public literature, or from inferences drawn using evolution-based computational methods. It is fully integrated with standard nomenclatures of different species (HUGO; Povey et al., 2001), Flybase (Gelbart et al., 1997), SGD (McMullan et al., 2004), and others. The VisANT tool and Predictome database are under constant development. Please visit VisANT home page for latest updates.

Critical Parameters and Troubleshooting

The quality of a VisANT network is heavily dependent on the reliability of biological interactions/associations used to construct it, which in turn relies on the quality of the experimental and inferential methods used to detect the interactions. At the time of this writing, proteins are displayed as linked if they are correlated by one or another method, but no weight is given to the reliability of the method by which a correlation (link) is established. In general, the reliability of a link will increase when it is established by more than a single method (Yanai and DeLisi, 2002) but here as well, the degree of improvement has not been quantitated. These restrictions are in the process of being removed, so that in the near future, links will be assigned probabilities and different sources of evidence will be combined using a Bayesian formalism (Imoto et al., 2003; Kim et al., 2004; Yu et al., 2004).

The current meta-network implementation does not allow duplication of individual interactions. For example, if proteins A and B coexist in complexes I and II, their interaction in both complexes cannot be displayed simultaneously. The capacity to display hierarchies of networks is a new capability which is evolving to enable dense functional modules to be represented as nodes, to find and evaluate the statistical significance of any motifs in these higher order networks, and to in turn represent the motifs as nodes in yet higher order networks. VisANT has a discussion board (*http://visant.bu.edu/discussion*) for comments and suggestions.

Literature Cited

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M., and Sherlock, G. 2000. Gene

ontology: Tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.* 25:25-29.

Bader, G.D., Betel, D., and Hogue, C.W. 2003. BIND: The Biomolecular Interaction Network Database. *Nucleic Acids Res.* 31:248-250.

Benson, D.A., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Wheeler, D.L. 2003. GenBank. *Nucleic Acids Res.* 31:23-27.

McMullan, G., Christie, J.M., Rahman, T.J., Banat, I.M., Ternan, N.G., and Marchant R. 2004. Saccharomyces Genome Database (SGD) provides tools to identify and analyze sequences from *Saccharomyces cerevisiae* and related sequences from other organisms. *Nucleic Acids Res.* 32 Database issue:D311-D314.

Gavin, A.C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J.M., Michon, A.M., Cruciat, C.M., Remor, M., Hofert, C., Schelder, M., Brajenovic, M., Ruffner, H., Merino, A., Klein, K., Hudak, M., Dickson, D., Rudi, T., Gnau, V., Bauch, A., Bastuck, S., Huhse, B., Leutwein, C., Heurtier, M.A., Copley, R.R., Edelmann, A., Querfurth, E., Rybin, V., Drewes, G., Raida, M., Bouwmeester, T., Bork, P., Seraphin, B., Kuster, B., Neubauer, G., and Superti-Furga, G. 2002. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature* 415:141-147.

Gelbart, W.M., Crosby, M., Matthews, B., Rindone, W.P., Chillemi, J., Russo Twombly, S., Emmert, D., Ashburner, M., Drysdale, R.A., Whitfield, E., Millburn, G.H., de Grey, A., Kaufman, T., Matthews, K., Gilbert, D., Strelets, V., and Tolstoshev, C. 1997. FlyBase: A Drosophila database. The FlyBase consortium. *Nucleic Acids Res.* 25:63-66.

Hermjakob, H., Montecchi-Palazzi, L., Bader, G., Wojcik, J., Salwinski, L., Ceol, A., Moore, S., Orchard, S., Sarkans, U., von Mering, C., Roechert, B., Poux, S., Jung, E., Mersch, H., Kersey, P., Lappe, M., Li, Y., Zeng, R., Rana, D., Nikolski, M., Husi, H., Brun, C., Shanker, K., Grant, S.G., Sander, C., Bork, P., Zhu, W., Pandey, A., Brazma, A., Jacq, B., Vidal, M., Sherman, D., Legrain, P., Cesareni, G., Xenarios, I., Eisenberg, D., Steipe, B., Hogue, C., and Apweiler, R. 2004. The HUPO PSI's molecular interaction format–a community standard for the representation of protein interaction data. *Nat. Biotechnol.* 22:177-183.

Imoto, S., Kim, S., Goto, T., Miyano, S., Aburatani, S., Tashiro, K., and Kuhara, S. 2003. Bayesian network and nonparametric heteroscedastic regression for nonlinear modeling of genetic network. *J. Bioinform. Comput. Biol.* 1:231-252.

Kanehisa, M., Goto, S., Kawashima, S., and Nakaya, A. 2002. The KEGG databases at GenomeNet. *Nucleic Acids Res.* 30:42-46.

Kim, S., Imoto, S., and Miyano, S. 2004. Dynamic Bayesian network and nonparametric regression for nonlinear modeling of gene networks from time series gene expression data. *Biosystems* 75:57-65.

Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, I., Zeitlinger, J., Jennings, E.G., Murray, H.L., Gordon, D.B., Ren, B., Wyrick, J.J., Tagne, J.B., Volkert, T.L., Fraenkel, E., Gifford, D.K., Young, R.A. 2002. Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science* 298:799-804.

Mellor, J.C., Yanai, I., Clodfelter, K.H., Mintseris, J., DeLisi, C. 2002. Predictome: A database of putative functional links between proteins. *Nucleic Acids Res.* 30:306-309.

Mewes, H.W., Frishmanm D., Gruber, C., Geier, B., Haase, D., Kaps, A., Lemcke, K., Mannhaupt, G., Pfeiffer, F., Schuller, C., Stocker, S., Weil, B. 2000. MIPS: A database for genomes and protein sequences. *Nucleic Acids Res.* 28:37-40.

Povey, S., Lovering, R., Bruford, E., Wright, M., Lush, M., Wain, H. 2001. The HUGO Gene Nomenclature Committee (HGNC). *Hum. Genet.* 109:678-680.

Tong, A.H., Evangelista, M., Parsons, A.B., Xu, H., Bader, G.D., Page, N., Robinson, M., Raghibizadeh, S., Hogue, C.W., Bussey, H., Andrews, B., Tyers, M., and Boone, C. 2001. Systematic genetic analysis with ordered arrays of yeast deletion mutants. *Science* 294:2364-2368.

Uetz, P., Giot, L., Cagney, G., Mansfield, T.A., Judson, R.S., Knight, J.R., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., Qureshi-Emili, A., Li, Y., Godwin, B., Conover, D., Kalbfleisch, T., Vijayadamodar, G., Yang, M., Johnston, M., Fields, S., and Rothberg, J.M. 2000. A comprehensive analysis of protein-protein interactions in *Saccharomyces cerevisiae*. *Nature* 403:623-627.

Wang, H., Tang, X., Liu, J., Trautmann, S., Balasundaram, D., McCollum, D., and Balasubramanian, M.K. 2002. The multiprotein exocyst complex is essential for cell separation in *Schizosaccharomyces pombe*. *Mol. Biol. Cell* 13:515-529.

Yu, J., Smith, V.A., Wang, P.P., Hartemink, A.J., and Jarvis, E.D. 2004. Advances to Bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*. [available online ahead of print at *http://bioinformatics.oupjournals.org/cgi/reprint/bth448v1*]

Yanai, I. and DeLisi, C. 2002. The society of genes: Networks of functional links between genes from comparative genomics. *Genome Biol.* 25:research0064. [Epub at *http://www.pubmedcentral.nih.gov/articlerender.fcgi?tool=pubmed&pubmedid=12429063*]

Key References

Hu, Z., Mellor, J., Wu, J., DeLisi, C. 2004. VisANT: An online visualization and analysis tool for biological interaction data. *BMC Bioinformatics* 5:17.

Explains the design principals and future development of VisANT.

Mellor et al., 2002. See above.

Introduces the development of Predictome database.

Internet Resources

http://visant.bu.edu

VisANT homepage.

http://visant.bu.edu/vmanual

The VisANT user's manual.

http://predictome.bu.edu

Homepage for the Predictome database

http://java.sun.com

Free source of Java run-time environment 1.4 or above. Refer to VisANT user manual for detailed instruction.

Contributed by Zhenjun Hu,
Joseph Mellor, and Charles DeLisi
Boston University
Boston, Massachusetts

Searching, Viewing, and Visualizing Data in the Biomolecular Interaction Network Database (BIND)

In fields ranging from medicine to biotechnology to agriculture, the use of traditional published biological information is slowly giving way to a systems-wide approach for solving biological questions. Invaluable initiatives like the Human Genome Project provided researchers with a "parts list" of life, but did not provide much information about how these parts assemble to create cells, tissues, and organisms. Structural genomics and proteome projects have gone a long way toward closing the gaps, offering information about protein content, numbers, and modifications. In many respects, however, genomic and proteomic data repositories address biological functions in only one or two dimensions, definitively answering questions of identity and expression levels, but providing little information about function. These efforts provide a snapshot of what is going on within a cell, rather than how these component parts interact to form complexes and pathways—the other dimensions that are critical to biological functions.

More recently, however, several groups have been addressing the abovementioned problem, finding ways to pull together biomolecular interaction and pathway data from various sources into central repositories against which researchers can test their hypotheses and probe for new insights. The Biomolecular Interaction Network Database (BIND; Alfarano et al., 2005) is one example of this effort.

BIND comprises data from peer-reviewed literature and direct submissions, and was conceived using the world's most comprehensive integrated bioinformatics standards, including those used by the NCBI for storing biomolecular sequence, taxonomy, structure, and literature information. BIND's data model was the first of its kind to be peer reviewed prior to database development, and is now a mature standard data format spanning molecular interactions, small molecule chemical reactions, interfaces from three-dimensional structures, and genetic-interaction networks.

BIND allows researchers to identify macromolecular complexes, metabolic pathways, and potential clues to drug targets and leads. BIND has close to 200,000 records of interaction data directly deposited by researchers or extracted from the peer-reviewed literature and a variety of genomic, proteomic, pathway, and disease-specific databases. These data are curated and validated using rigorous bioinformatics standards.

Using any of more than 20 different search functions available through BIND's Web interface, researchers can identify interacting molecules on the basis of parameters such as sequences, gene names, publication record, and species origin, and examine how these interactions fit into greater molecular networks using BIND's Interaction Viewer. Alternatively, new features recently added to BIND allow researchers to search relatively broad terms, such as cancer, and pinpoint molecules of particular interest based on characteristics such as subcellular colocalization, biological function, and binding partners. Furthermore, because each record has been hand curated and annotated using a variety of informatics tools, molecule descriptions are heavily cross-referenced to supplemental genetic or structural data that might prove important for further analysis. Finally, scientists interested in the interplay between interactions and small molecules (which can give clues to biological function and also help pinpoint druggable targets) can also find information about potential small molecules that bind to each protein in BIND through the SMID (Small Molecule Interaction Database) links provided in the BIND interface.

Contributed by Randall C Willis and Christopher W.V. Hogue
Current Protocols in Bioinformatics (2005) 8.9.1-8.9.30

BIND supports additional file formats to achieve compatibility with other database efforts including the HUPO PSI Level 2. BIND is a founding partner in the emerging consortium of interaction databases called the International Molecular-Interaction Exchange (IMEx) consortium, alongside the Molecular Interactions database (MINT; Zanzoni et al., 2002), the IntAct Project (Hermjakob et al., 2004) and the Database of Interacting Proteins (DIP; Xenarios et al., 2002).

In this unit, an overview of the BIND interface is given in Basic Protocol 1, and protocols are provided for searching BIND via the Internet. Basic Protocol 2 describes how to search BIND using simple text, database identifiers, or short labels to retrieve records. Basic Protocols 3 to 5 describe searching BIND using field-specific information, BINDBlast, and the statistics page, respectively. Protocols for viewing BIND search results (Basic Protocol 6) and individual records (Basic Protocol 7), and for exporting search results for use with other software (Basic Protocol 8) are also included. Furthermore, protocols are provided for finding associated small-molecule binding sites and for visualizing biomolecular interactions within BIND (Basic Protocol 9) or transferring this information to the visualization software Cytoscape and Cn3D (Basic Protocol 10).

BASIC PROTOCOL 1

THE BIND INTERFACE: GETTING STARTED

This protocol describes the BIND interface, its various options, and how to get started using BIND from the home page.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

No local files are required

1. Point the browser to *http://bind.ca*. The BIND home page, similar to the view in Figure 8.9.1, will appear.

   The BIND home page is organized with the key search and submission tools listed in the center of the page underneath a query box (labeled Search). Searching, submitting, and downloading functions are accessed by scrolling the mouse cursor over the appropriately labeled button in the row of icons (Fig. 8.9.1.). The home page also contains several pieces of critical information about the version history and use of BIND, and provides access to information about BIND administration ("About"), curation, and development via the left navigation column, and basic database statistics via the boxes on the top right. Further down the right-hand side, an identifier search box is provided for queries using accession numbers and codes from a variety of databases, and, at the bottom right (under Imports), a box is provided containing quick links to datasets provided on the BIND Web site that are incorporated from other databases.

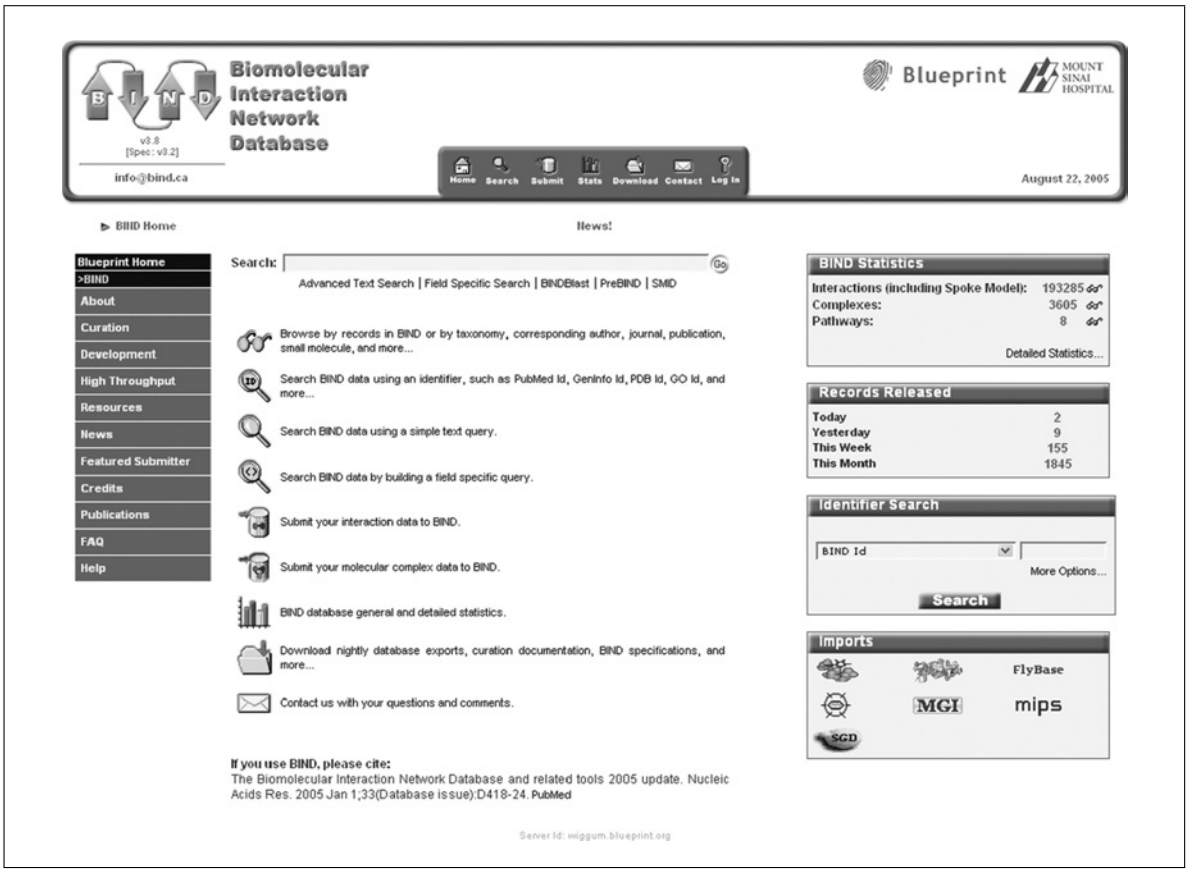


Figure 8.9.1 The BIND home page accessed via a WWW browser at *http://bind.ca.*

2. Clicking on the version statement under the logo in the top-left corner (e.g., "v3.8" in Fig. 8.9.1) will provide the user with information about features added in the last few system updates. Click the browser Back button to return to the home page.

3. Clicking on the bars in the box on the left-hand side of the screen links users to background information about BIND.

   BIND's latest Web services software spans over 2000 metadata fields and is constructed using the J2EE software platform. The About bar links to a description of BIND's construction and function. The Curation bar links to a document that describes how BIND data are treated during the submission and curation process. The Development bar offers database developers and advanced users information about BIND software systems, associated tools, and download instructions. The High Throughput bar provides information about large-scale data sets submitted to BIND. The Resources bar links to user manuals, tutorials, and external resources related to BIND use. The News bar links to updates of data sets included in BIND and related announcements on BIND development. The Featured Submitter bar links to highlights of noteworthy scientists whose labs have provided BIND records. The Credits bar identifies current and past scientists involved in BIND development. The Publications bar links to papers describing BIND. The FAQ bar provides access to answers to routine questions about BIND. Finally, the Help bar links to tutorials for searching and submitting, and links to the E-mail address info@bind.ca where user questions can be submitted. The Search Help Page offers advanced users a tutorial in formulating text queries using the Lucene syntax from the query box.

4. Users can search BIND through a variety of mechanisms available on the home page (Fig. 8.9.1) that are displayed schematically in Figure 8.9.2. The BIND Text Search window (illustrated in Fig. 8.9.2A; accessed by placing the mouse cursor over the Search icon at the top of the home page and selecting Text Search from the pop-up menu) allows users to search BIND using simple text queries (described in

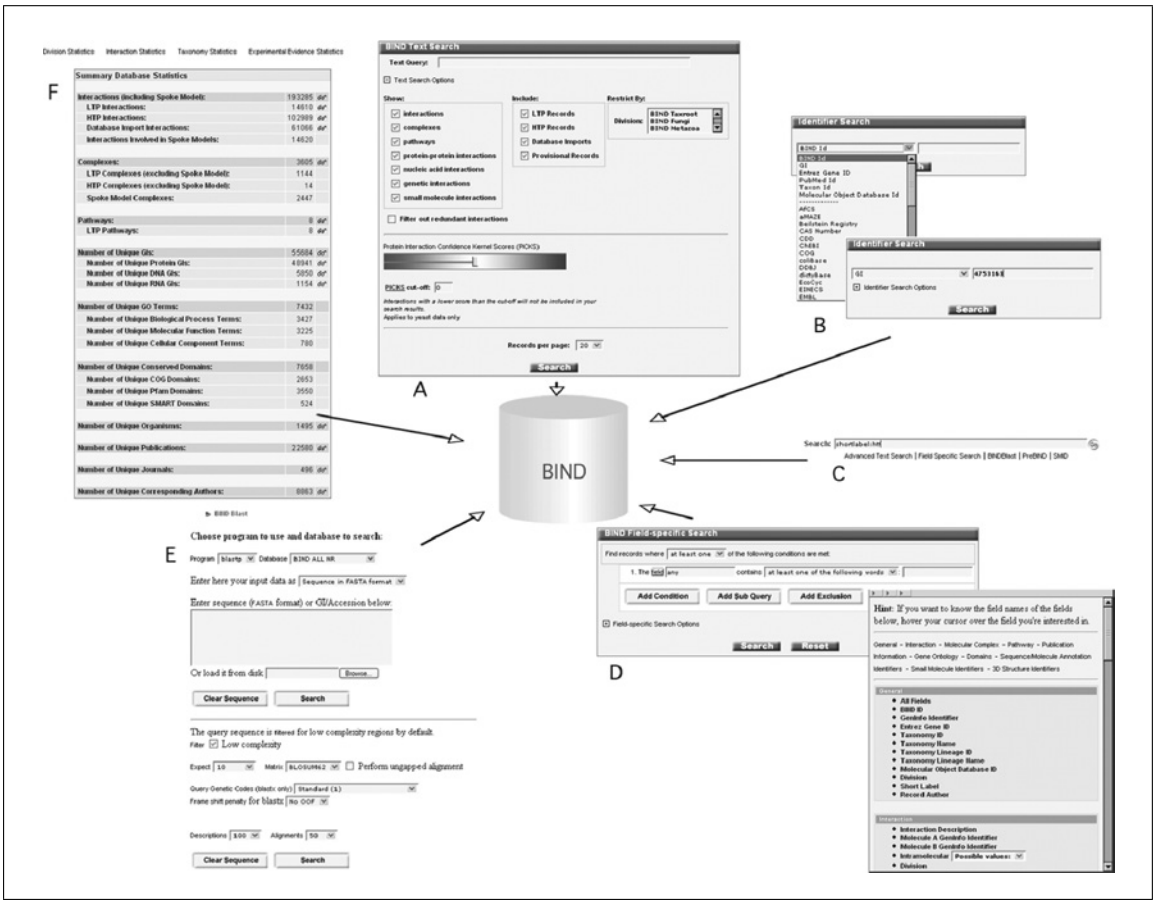


Figure 8.9.2 Multiple methods for querying BIND. (**A**) Text Query (see Basic Protocol 2) with options expanded by clicking the [+] symbol (note that this has changed to [–] here). Options are provided for limiting the query to specific types of interactions or to specific BIND divisions organized as branches of the taxonomy tree. One can also choose to filter out redundant records and change the number of records per page. LTP refers to hand-curated records, HTP are high-throughput records. (**B**) Identifier Search (see Basic Protocol 2) with pull-down list of over 40 different databases. Example here uses an NCBI GenInfo (GI) number. (**C**) Text search box (see Basic Protocol 2) from the BIND home page showing the syntax for a field-specific query for a molecule short label. (**D**) Field-Specific query dialog box (see Basic Protocol 3) showing the expanded field-name list box from which database fields may be chosen. (**E**) BINDBlast (see Basic Protocol 4) provides a sequence-similarity based query interface for BIND. (**F**) BIND Statistics (see Basic Protocol 5) are convenient for quickly finding relevant subsets of BIND records which can be browsed by clicking on the spectacles icon.

Basic Protocol 2). By clicking on the other options in that same pop-up menu, users can search BIND using other parameters such as BINDBlast (illustrated in Fig. 8.9.2E; described in Basic Protocol 4) or field-specific searching (illustrated in Fig. 8.9.2D; described in Basic Protocol 3). Clicking on the Stats icon above the Search window on the home page allows searching via the statistics pages (Basic Protocol 5).

5. Clicking on the icons in the middle of the page or in the blue bar at the top of the BIND home page illustrated in Figure 8.9.1 links users to mechanisms to search BIND, review basic BIND statistics, submit data to BIND, download BIND-related materials, get BIND help, or log in to the personal version of BIND used for submitting data to BIND.

   Briefly, the functions of the icons in the middle of the page (Fig. 8.9.1) are as follow. The spectacles icon allows users to browse BIND data using a variety of criteria such as small molecules, taxonomy, or journal names (described in Basic Protocol 5). The three magnifying glass icons indicate different methods of searching, including simple text, database identifiers, and field-specific queries (described in Basic Protocols 2 to 4).

The two cylinder icons link users to mechanisms to submit interaction or complex data to BIND. The bar graph icon provides information about basic BIND statistics classified by parameters such as taxonomy and experimental method. The folder icon links to BIND's FTP site, where files can be downloaded. The envelope icon allows users to submit questions about BIND directly to User Services via info@bind.ca.

The key icon in the blue bar at the top of the page links to a log-in page where BIND users can submit and examine their own data in a private session called My-BIND.

6. The top-most right-hand box (BIND Statistics) contains a link labeled Detailed Statistics that leads to the current number of the interaction, complex, and pathway records in BIND, including all data sources. Clicking on the spectacle icon will bring up the actual records described. Browsing BIND presents database records in reverse order with respect to when they were added, so that the latest records appear at the top of the list.
7. The second-right hand box from the top (Records Released) shows the number of records recently released. Clicking on This Month displays a list of the records added over the past calendar month.
8. The second right-hand box from the bottom (Identifier Search) offers the ability to rapidly search BIND using any of over 50 different database identifiers (described in Basic Protocol 2).
9. The bottom right-hand box (Imports) allows users to browse records arising from specific third-party databases imported into BIND. Place the mouse cursor over each of the icons to reveal the names of the databases imported.

SEARCHING BIND USING TEXT QUERIES, IDENTIFIER FUNCTION, OR MOLECULE SHORT LABEL

BASIC PROTOCOL 2

This protocol describes the three most direct methods for searching BIND. The first series of steps involves the use of simple text descriptions of the gene, protein, small molecule, or condition of interest, which generally yields the widest spectrum of results. The second series of steps involves the use of specific identifiers from BIND itself or other databases that provide information about the target molecule, such as genomic, proteomic, publication, or organism repositories. The final series of steps involve the use of molecule short labels to search BIND.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

No local files are required

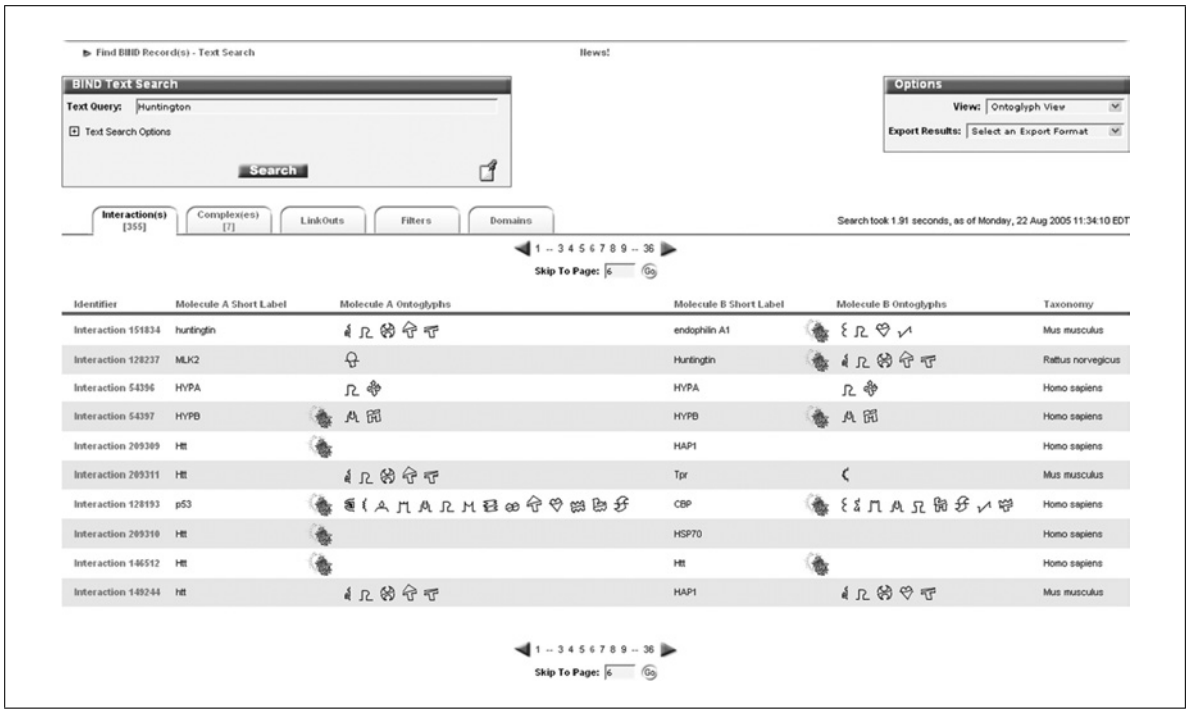


Figure 8.9.3 Summary of BIND results from simple text query for `huntington`.

Searching using a text query

1. Point the browser to *http://bind.ca*. Scroll over the Search icon (magnifying glass) in the row of icons at the top of the page that appears (Fig. 8.9.1) and click on Text Search in the pop-up menu. A text search window will appear.

2. If interested in BIND records associated with a disease name such as "Huntington Disease," simply type the search term `huntington` into the Text Query box. Click the Search button at the bottom of the window to begin the search.

3. In this case, the example illustrated is a search for all interactions related to Huntington disease, in which a simple text query using the string `huntington` is performed. By using this relatively generic term, the user accesses the full realm of biomolecular interactions with the term `huntington` in the BIND record, and does not limit the results to interactions involving the "huntingtin" protein, which is the central target in Huntington disease.

4. BIND performs an exhaustive search of each field in every record for the text term `huntington` and returns a list of interactions (Fig. 8.9.3), each summarized on a single line.

   Figure 8.9.3 shows the list of interactions for the search term `huntington`*, submitted in step 3.*

5. Each interaction summary includes the BIND ID, the short label names of the interacting molecules, their corresponding OntoGlyphs, and associated Taxonomy information indicating the species of origin. Records are grouped by types (interactions, complexes) under tabs. A Tab is also provided for LinkOuts to provide links from the entire data set to other databases (e.g., PubMed abstracts and NCBI Sequences). The Options box at the top right has two drop-down menus: View, which changes the way that the browser shows the list of records, and Export Results, which offers methods for saving the results in various standard formats (see Basic Protocols 6 to 10).

Searching using the identifier function

The following steps describe how to retrieve records using database identifiers (Figure 8.9.2B). In some cases, the user will have precise information about the identifier that retrieves the molecule of interest from other biomolecular databases, such as model organism repositories, genomic sequence databases, publication collections, or small-molecule libraries. Database identifiers are usually lists of unique codes, which may be numeric or alphanumeric. Examples are GenBank Accession numbers (e.g., `AAA90987.1`) and NCBI GenInfo (GI) numbers (e.g., `841190`) for sequences and PDB codes (e.g., `1OMD`) for 3-D structure records. Table 8.9.1 offers URLs and examples of such databases, the identifiers of which can be used to search BIND. By using identifiers from third-party databases, users more precisely focus their BIND search results than through a simple text query. Searching with a sequence or other identifier may return no results if that molecule is not found in a BIND record. Conversely it may return more than one records if the molecule appears in multiple interaction records. Searching with a BIND ID returns a single interaction corresponding to that exact BIND record.

6. At the BIND main page (Fig. 8.9.1), scroll over the Search icon on the top menu and click on Identifier Search. An Identifier Search window will appear.

7. BIND can be searched with many molecule identifiers from numerous molecular databases. A list of identifiers in the form of a drop-down menu may be viewed by clicking the triangle in the Identifier Search box (Fig. 8.9.2B) within the window that was invoked in step 6. Enter the corresponding identifier in the text box to the right of the drop-down menu, then press the Search button.

   As an example, a search will be performed using the GenInfo (GI) Identifier corresponding to a version of the huntingtin molecule (`gi:4753163`*) in BIND at the time of writing.*

8. If the identifier is unknown, use the information in Table 8.9.1 to search other databases for the identifier of the specific molecule of interest.

9. The window that will appear after submitting the identifier in step 7 contains a list of interactions similar to that in Figure 8.9.3. Clicking on any of the molecule short labels (“huntingtin” for this example) in the list returns a “molecule-centric” view (Fig. 8.9.4) listing interactions involving huntingtin (`gi:4753163`). Each record is summarized by a BIND ID, a description of the interaction, the species, and the publication supporting the interaction.

Searching using the molecule short name (“shortlabel”)

The following steps describe searching BIND using molecule short names, stored as a field called “shortlabel” (Figure 8.9.2C). BIND’s field-specific search allows searching of the “shortlabel” fields of all records; the field-specific search function is explained in Basic Protocol 3. A quicker method using short labels is shown here.

10. Return to the BIND home page as described in step 1. Locate the search box near the center of the page (see Fig. 8.9.1).

11. If the field to search for is already known, as is the case here, the query can be prefixed with the fieldname. For example, “htt” is the short label for the huntingtin molecule and is therefore to be searched in the “shortlabel” field. The query to be typed is therefore `shortlabel:htt`. Click on the GO button to execute this search.

    To find two short labels in an interaction, try the query `shortlabel:htt AND shortlabel:p53`.

12. Results are returned in the OntoGlyphs view (similar to Fig. 8.9.3).

Table 8.9.1 Finding Identifier Information for BIND Searches

Information known	Identifier needed	Example: Find interactions involving . . .	URL: Where can I find the identifiers?
3-D Structure	MMDB	Tyrosine phosphatase structure with MMDB i.d. 307	*http://ncbi.nlm.nih.gov/Structure/MMDB/mmdb.shtml*
	PDB	Citrate synthase structure with PDB i.d. 1AL6	*http://www.rcsb.org/pdb/*
	mmCIF Het	Oxaloacetate bound as heteroatom to PDB i.d. 1AL6	*http://ndbserver.rutgers.edu/mmcif*
	EMD	GroEL-ATP with EMD i.d. 1047	*http://www.ebi.ac.uk/msd-srv/emsearch/index.html*
Disease	OMIM	IL-8 with OMIM i.d. 146930	*http://ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM*
Domain	CDD	Domain in SHIP1 with CDD i.d. SH2	*http://ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml*
	COG	Domain in MreB with COG i.d. DnaK	*http://ncbi.nlm.nih.gov/COG*
	SMART	Domain in Syn1A with SMART i.d. SynN	*http://smart.embl-heidelberg.de*
	Pfam	Domain in Nedd5 protein with Pfam i.d. GTP_CDC	*http://www.sanger.ac.uk/Software/Pfam*
Gene	DDBJ	Histone H3 protein with DDBJ i.d. BAA93621	*http://www.ddbj.nig.ac.jp*
	RefSeq	Metazoan protein Neil 1 with RefSeq i.d. NP_078884	*http://ncbi.nlm.nih.gov/RefSeq/*
	LocusLink	Metazoan protein BLC-2 with LocusLink i.d. 12043	*http://ncbi.nlm.nih.gov/LocusLink*
	GenBank	TRAP25 with GenBank i.d. AAH08226	*http://ncbi.nlm.nih.gov/Genbank*
	Entrez Gene	Histone H3 protein with Entrez Gene i.d. 852295	*http://ncbi.nih.gov/Entrez*
	EMBL	KEM1 gene with EMBL i.d. CAA38520	*http://www.ebi.ac.uk/embl/index.html*
Ontology term	GO	The GO term DNA Binding	*http://www.geneontology.org*
Organism	TAIR	EGL3 protein with TAIR i.d. At1g63650	*http://www.arabidopsis.org*
	Taxon	*Homo sapiens* with Taxon i.d. 9606	*http://ncbi.nlm.nih.gov/Taxonomy*

continued

Table 8.9.1 Finding Identifier Information for BIND Searches, *continued*

Information known	Identifier needed	Example: Find interactions involving . . .	URL: Where can I find the identifiers?
Organism (*continued*)	Flybase	Virilizer protein with Flybase i.d. FBgn0003977	*http://flybase.bio.indiana.edu*
	MGI	Mouse SRC with MGI i.d. 98397	*http://www.informatics.jax.org*
	Wormbase	Worm GST protein with Wormbase i.d. gst-5	*http://www.wormbase.org*
	RGD	SRF protein with RGD i.d. 621489	*http://rgd.mcw.edu*
	SGD	Yeast Ste11 protein with SGD i.d. S0004354	*http://www.yeastgenome.org*
	dictyBase	Gene named mlcE	*http://dictybase.org*
Pathway	AfCS	GRB2 with AfCS i.d. A001088	*http://www.signaling-gateway.org*
Protein	UniProt	Dcp1B with UniProt i.d. Q96BP8	*http://www.pir.uniprot.org*
	Swiss-Prot	PKA with Swiss-Prot i.d. Q8K1M3	*http://us.expasy.org/sprot*
	GI	Barx2 protein with GI 7304917	*http://ncbi.nlm.nih.gov*
	TrEMBL	Translated nucleotide sequence	*http://www.ebi.ac.uk/trembl*
	IPI	AFG3-like Protein 1 with IPI00015171	*http://www.ebi.ac.uk/IPI/IPIhelp.html*
	PIR	Clp endopeptidase with PIR i.d. I40508	*http://pir.georgetown.edu*
Publication	PubMed	Cell publication about HIV with PMI.D. 14505570	*http://ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed*
Small molecule	MDL	Histidine with MDL i.d. MFCD00064315	*http://www.mdli.com*
	Merck Index	IP6 with Merck Index i.d. 7542 (12)	*http://www.merck.com*
	Beilstein Registry	Glucose with Beilstein i.d. 1724615	*http://www.mdl.com/products/knowledge/crossfire_beilstein/*
	CAS Number	GTP with CAS i.d. 86-01-1	*http://www.cas.org/*
	Klotho	Palmitate with Klotho i.d. KLM0000296	*http://www.biocheminfo.org/klotho/*
	EINECS	ATP with EINECS i.d. 200-283-2	*http://ecb.jrc.it/esis/*

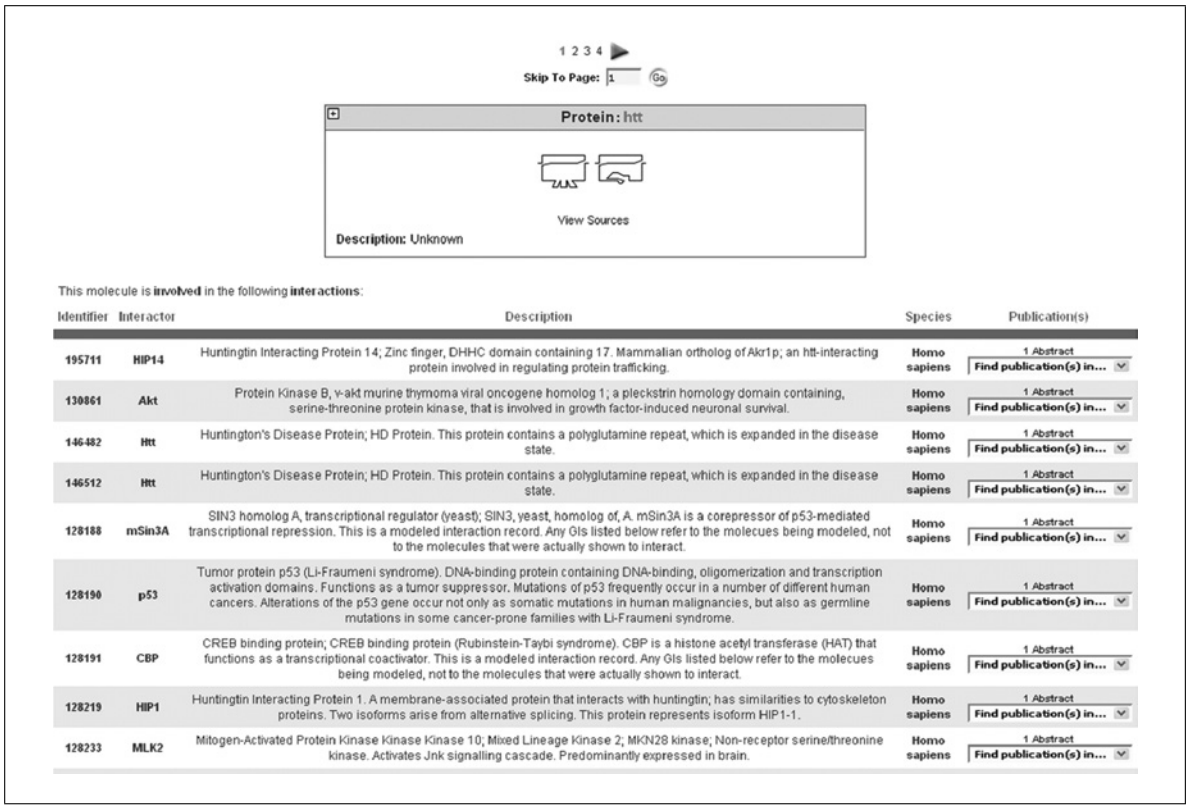


Figure 8.9.4 Molecule-centric summary of BIND results from GI query for `huntingtin`.

BASIC PROTOCOL 3

SEARCHING BIND USING THE FIELD-SPECIFIC FUNCTION

The following protocol describes how to retrieve records in BIND using the field-specific search tool provided (Fig. 8.9.2D). In contrast to text searches, field-specific queries are refined to search only specified fields for a given criteria. By focusing the terms of the search more specifically using the curated fields of information in BIND, users are more likely to find records of interest, and will be less likely to have to sort manually through large amounts of information.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

No local files are required

1. Point the browser to *http://bind.ca* to access the BIND home page. Place the mouse cursor over the Search icon on the top of the page and click on Field-Specific Search in the pop-up menu that appears. A Field-Specific Search box will appear (see Fig. 8.9.2D and Fig. 8.9.5).

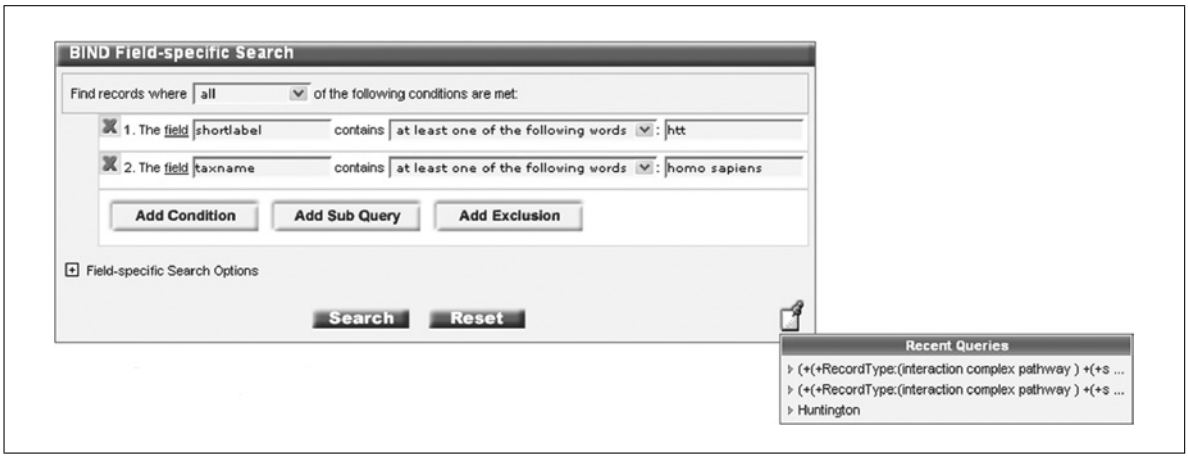


Figure 8.9.5 Creating a field-specific query in this case used to search for a protein with a specific name from a specific organism. Recent queries are expanded in a list at the bottom right-hand corner of the dialog box.

2. The field-specific search box allows one to specify any number of conditions for the search using the Add Condition button. Too many conditions will often leave the user with no search result, so start with a small list and add conditions one at a time. Each new condition added appears on a separate numbered line. Clicking the "field" link on each numbered line opens a pop-up menu with a sorted list of searchable fields as shown at the bottom right-hand corner of Figure 8.9.2D. Click on any item in this list to include it in the query. Next, type the word to search for in that field in the text box at the right. Press the Search button at the bottom of the box to execute the query.

3. The topmost line ("Find records where . . . of the following conditions are met") controls how the listed query conditions are treated. The user interface is structured to be logical and easily understood by novices. By default it is set to "at least one." For experienced database users, this corresponds to a Boolean OR search of all of the search conditions together. The alternate menu choice "all" corresponds to a Boolean AND search of all of the search conditions together.

4. Add query terms to include in the search results or add terms to be excluded from the search results. To expand the query, add additional field query conditions with the Add Condition button. Add a Sub Query using the button of that name, in order to refine a condition (i.e., to add more AND or OR terms to it). Terms in a specific field can be excluded from being listed in the search using the Add Exclusion button. Terms can be removed by clicking the red `x` adjacent to them (Fig. 8.9.5).

   The search can be further refined by expanding the Field-Specific Search Options as described in the following steps.

5. As an example, query BIND for all huntingtin protein interactions in humans by clicking the "field" link on the line for the first query field, selecting Short Label from the menu that appears, and typing `htt` as the first field query term. This is equivalent to the `shortlabel:htt` search done at the end of Basic Protocol 2. Next, select Add Condition. Click on the "field" link on the line for the new query field that appears, and select the listed field Taxonomy Name from the menu that appears. Type `homo sapiens` as the query term in that query filed. To return results that satisfy both conditions, change "at least one" condition on the top line to "all" conditions. Click the Search button to begin the query.

   The Reset button will clear all conditions. To revisit recent queries, click on the pin-and-paper icon at the bottom right corner of the search box.

6. Results are returned in the OntoGlyphs view (similar to Fig. 8.9.3).

BASIC PROTOCOL 4

SEARCHING BIND BY SEQUENCE: BINDBlast

The following protocol describes how to retrieve records in the BIND using a protein sequence search (Fig. 8.9.2E). BINDBlast uses NCBI's BLAST program to search against the sequence of the proteins found in BIND. BINDBlast returns a list of proteins found in BIND records that are similar in sequence to the query protein. This feature is recommended for users with genes or proteins of interest, as it will typically find related interactions in BIND from a number of different organisms. It also affords the opportunity to identify interactions of related proteins and thereby possibly identify secondary pathways in which the molecule of interest may participate.

Necessary Resources

Hardware

Workstation with connection to the Internet

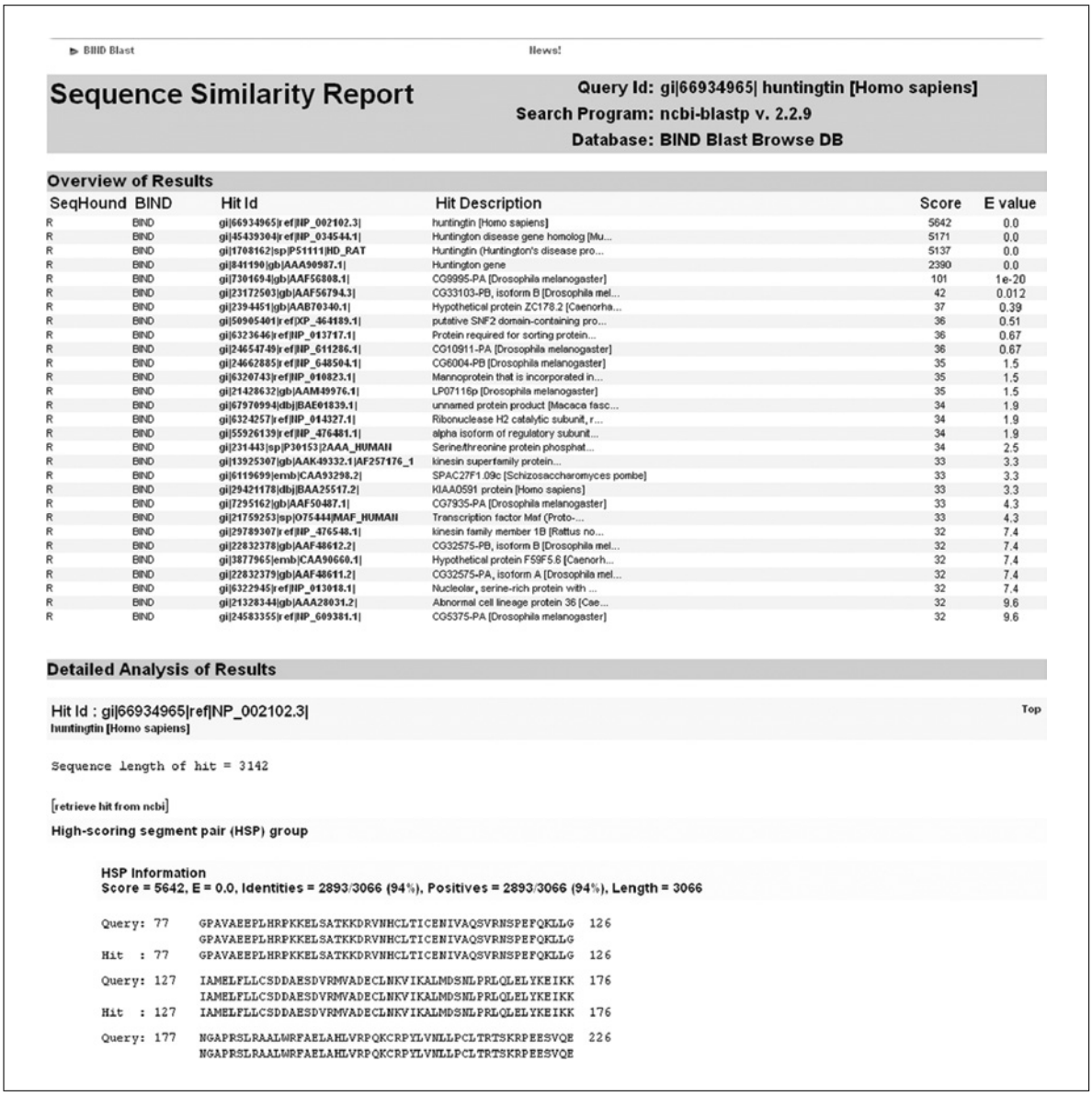
Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

No local files are required

1. Point the browser to *http://bind.ca* to access the BIND home page. Place the mouse cursor over the Search icon on the top of the page and click on BINDBlast in the pop-up menu that appears. The BINDBlast interface will appear (Fig. 8.9.2E).
2. The three pull-down menus at the top of the BINDBlast interface window (Fig. 8.9.2E) specify the type of BLAST to run (only protein-protein BLAST is available), the BIND division to be searched, and the format of the input data. The lower section of the BINDBlast interface window allows one to modify some of the search parameters. It is possible turn the filter on or off for low-complexity regions, change the expect value or search matrix, turn the ungapped alignment on or off, select a genetic code, change the frame-shift penalty, and change the number of descriptions and alignments returned. As an example, to BLAST the sequence of the human huntingtin protein, select Accession or GI from the "Enter here your input data as" pull-down menu, and enter the accession number `NP_002102.1` on the first line of the sequence text box immediately below. Click the Search button.
3. The search results page for the query submitted in step 2, shown in Figure 8.9.6, is similar to the results page for any BLAST search. After the page title, the Overview of Results displays a sorted table of the best hits. Table columns from left to right link to R, which is the redundant group of sequences from Blueprint's SeqHound database; to BIND, which is the list of BIND records in the Molecule Centric view containing its interaction partners; to Hit ID, which leads to the sequence hits in the NCBI databases; and to Hit Description, which leads to the sequence alignment further down the page. The corresponding score and e-value are also provided. This is followed by a color-coded, pairwise alignment of each hit with the query protein sequence. For more specific BLAST help, visit the NCBI Web site (*http://www.ncbi.nlm.nih.gov/blast/*; also see Chapter 3).



▶ BIND Blast News!

Sequence Similarity Report

Query Id: gi|66934965| huntingtin [Homo sapiens]
Search Program: ncbi-blastp v. 2.2.9
Database: BIND Blast Browse DB

Overview of Results

SeqHound	BIND	Hit Id	Hit Description	Score	E value
R	BIND	gi|66934965|ref|NP_002102.3|	huntingtin [Homo sapiens]	5642	0.0
R	BIND	gi|45439304|ref|NP_034544.1|	Huntington disease gene homolog [Mu...	5171	0.0
R	BIND	gi|1708162|sp|P51111|HD_RAT	Huntingtin (Huntington's disease pro...	5137	0.0
R	BIND	gi|841190|gb|AAA90987.1|	Huntington gene	2390	0.0
R	BIND	gi|7301694|gb|AAF56808.1|	CG9995-PA [Drosophila melanogaster]	101	1e-20
R	BIND	gi|23172503|gb|AAF56794.3|	CG33103-PB, isoform B [Drosophila mel...	42	0.012
R	BIND	gi|239-4451|gb|AAB70340.1|	Hypothetical protein ZC178.2 [Caenorha...	37	0.39
R	BIND	gi|50905401|ref|XP_464189.1|	putative SNF2 domain-containing pro...	36	0.51
R	BIND	gi|6323646|ref|NP_013747.1|	Protein required for sorting protein...	36	0.67
R	BIND	gi|24654749|ref|NP_611286.1|	CG10911-PA [Drosophila melanogaster]	36	0.67
R	BIND	gi|24662885|ref|NP_648504.1|	CG6004-PB [Drosophila melanogaster]	35	1.5
R	BIND	gi|6320743|ref|NP_010823.1|	Mannoprotein that is incorporated in...	35	1.5
R	BIND	gi|21428632|gb|AAM49976.1|	LP07116p [Drosophila melanogaster]	35	1.5
R	BIND	gi|67970994|dbj|BAE01839.1|	unnamed protein product [Macaca fasc...	34	1.9
R	BIND	gi|6324257|ref|NP_014327.1|	Ribonuclease H2 catalytic subunit, r...	34	1.9
R	BIND	gi|55926139|ref|NP_476481.1|	alpha isoform of regulatory subunit...	34	1.9
R	BIND	gi|231443|sp|P30153|2AAA_HUMAN	Serine/threonine protein phosphat...	34	2.5
R	BIND	gi|13925307|gb|AAK49332.1|AF257476_1	kinesin superfamily protein...	33	3.3
R	BIND	gi|6119699|emb|CAA93298.2|	SPAC27F1.09c [Schizosaccharomyces pombe]	33	3.3
R	BIND	gi|29421178|dbj|BAA25517.2|	KIAA0591 protein [Homo sapiens]	33	3.3
R	BIND	gi|7295162|gb|AAF50487.1|	CG7935-PA [Drosophila melanogaster]	33	4.3
R	BIND	gi|21759253|sp|O75444|MAF_HUMAN	Transcription factor Maf (Proto-...	33	4.3
R	BIND	gi|29789307|ref|NP_476548.1|	kinesin family member 1B [Rattus no...	32	7.4
R	BIND	gi|22832378|gb|AAF48612.2|	CG32575-PB, isoform B [Drosophila mel...	32	7.4
R	BIND	gi|3877965|emb|CAA90660.1|	Hypothetical protein F59F5.6 [Caenorh...	32	7.4
R	BIND	gi|22832379|gb|AAF48611.2|	CG32575-PA, isoform A [Drosophila mel...	32	7.4
R	BIND	gi|6322945|ref|NP_013018.1|	Nucleolar, serine-rich protein with ...	32	7.4
R	BIND	gi|21328344|gb|AAA28031.2|	Abnormal cell lineage protein 36 [Cae...	32	9.6
R	BIND	gi|24583355|ref|NP_609381.1|	CG5375-PA [Drosophila melanogaster]	32	9.6

Detailed Analysis of Results

Hit Id : gi|66934965|ref|NP_002102.3| Top
huntingtin [Homo sapiens]

Sequence length of hit = 3142

[retrieve hit from ncbi]

High-scoring segment pair (HSP) group

HSP Information
Score = 5642, E = 0.0, Identities = 2893/3066 (94%), Positives = 2893/3066 (94%), Length = 3066

```
Query: 77    GPAVAEEPLHRPKKELSATKKDRVNHCLTICENIVAQSVRNSPEFQKLLG  126
              GPAVAEEPLHRPKKELSATKKDRVNHCLTICENIVAQSVRNSPEFQKLLG
Hit  : 77    GPAVAEEPLHRPKKELSATKKDRVNHCLTICENIVAQSVRNSPEFQKLLG  126

Query: 127   IAMELFLLCSDDAESDVRMVADECLNKVIKALMDSNLPRLQLELYKEIKK  176
              IAMELFLLCSDDAESDVRMVADECLNKVIKALMDSNLPRLQLELYKEIKK
Hit  : 127   IAMELFLLCSDDAESDVRMVADECLNKVIKALMDSNLPRLQLELYKEIKK  176

Query: 177   NGAPRSLRAALWRFAELAHLVRPQKCRPYLVNLLPCLTRTSKRPEESVQE  226
              NGAPRSLRAALWRFAELAHLVRPQKCRPYLVNLLPCLTRTSKRPEESVQE
```

Figure 8.9.6 Overview of BINDBlast results. The sequence alignments are truncated in the figure for clarity, but can be seen by scrolling down.

SEARCHING BIND BY BROWSING THE STATISTICS PAGE

BASIC PROTOCOL 5

The following describes how to retrieve sets of interaction data by browsing the statistics section of BIND (Fig. 8.9.2F). Precomputed queries are run on a nightly basis to generate tables of statistics. The statistics pages provide links to these precomputed queries, which many users find to be of interest. This search method offers a wide spectrum of results ranging from the type of molecules in an interaction, the type of curation provided, the experimental conditions used to determine the interaction, interaction subsets based on organism, or interaction subsets submitted by a specific author.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

No local files are required

1. Point the browser to *http://bind.ca* to access the BIND home page. Place the mouse cursor over the Stats icon at the top of the page, and click Record Statistics in the pop-up menu that appears. This will bring up the Summary Database Statistics page, as shown in Figure 8.9.2F.

   There are four other statistics pages that can be browsed from this page, as represented by the links near the top of the page labeled Division Statistics, Interaction Statistics, Taxonomy Statistics, and Experimental Evidence Statistics. Take a moment to explore each page.

2. Within the Summary Database Statistics page, navigate through the various statistics and select the set of data of interest for viewing. For example, to view the Number of Unique Organisms in BIND, click on the spectacles icon to the right of the number on that line in the table of statistics.

3. On the page which then appears, the query results will appear in alphabetical order and will include the Taxonomy ID field as well as the number of occurrences (interactions) containing the specific organism of choice.

4. Click the entry in the Occurrences field to browse for the interactions associated with a particular species. The results will appear in the OntoGlyph view (similar to Fig. 8.9.3).

5. Go back to the Summary Database Statistics page (as in step 1, or use the browser Back button) and select the Division Statistics link from the row of links at the top of the page.

   The page which then appears contains a set of statistics on the various divisions in BIND, including the external database imports that have been included in BIND, such as MIPS Mammalian/Yeast (MIPS), FlyBase), and Mouse Genome Informatics (MGI). The records from each division can be selected more precisely by molecular interaction type, for example, "Protein interacting with Protein" or "Protein interacting with DNA."

6. To view the interactions associated with a division or subset, click on the magnifying glass or icon to the right of the entry.

7. Interaction Statistics, Taxonomy Statistics, and Experimental Evidence Statistics pages can also be obtained through links at the top of the Summary Database Statistics page.

BASIC PROTOCOL 6

VIEWING BIND SEARCH RESULTS

Once a search has been performed, the list of results can be viewed using various formats available in the Options box drop down-menu that appears at the upper right-hand side of the screen above the search result folders (see Fig. 8.9.3 for location of the Options box; see Figure 8.9.7A for a close-up view of the drop-down menu). Different views of the search results are available for formatting within the Web browser: OntoGlyph View (default; illustrated in Fig. 8.9.3 and Fig. 8.9.7B), ProteoGlyph View (protein domain symbols; Fig. 8.9.7C), Single Line View (Fig. 8.9.7D), GO Summary View (not shown in figure), and Domain Summary View (Fig. 8.9.7E; similar to GO Summary View).

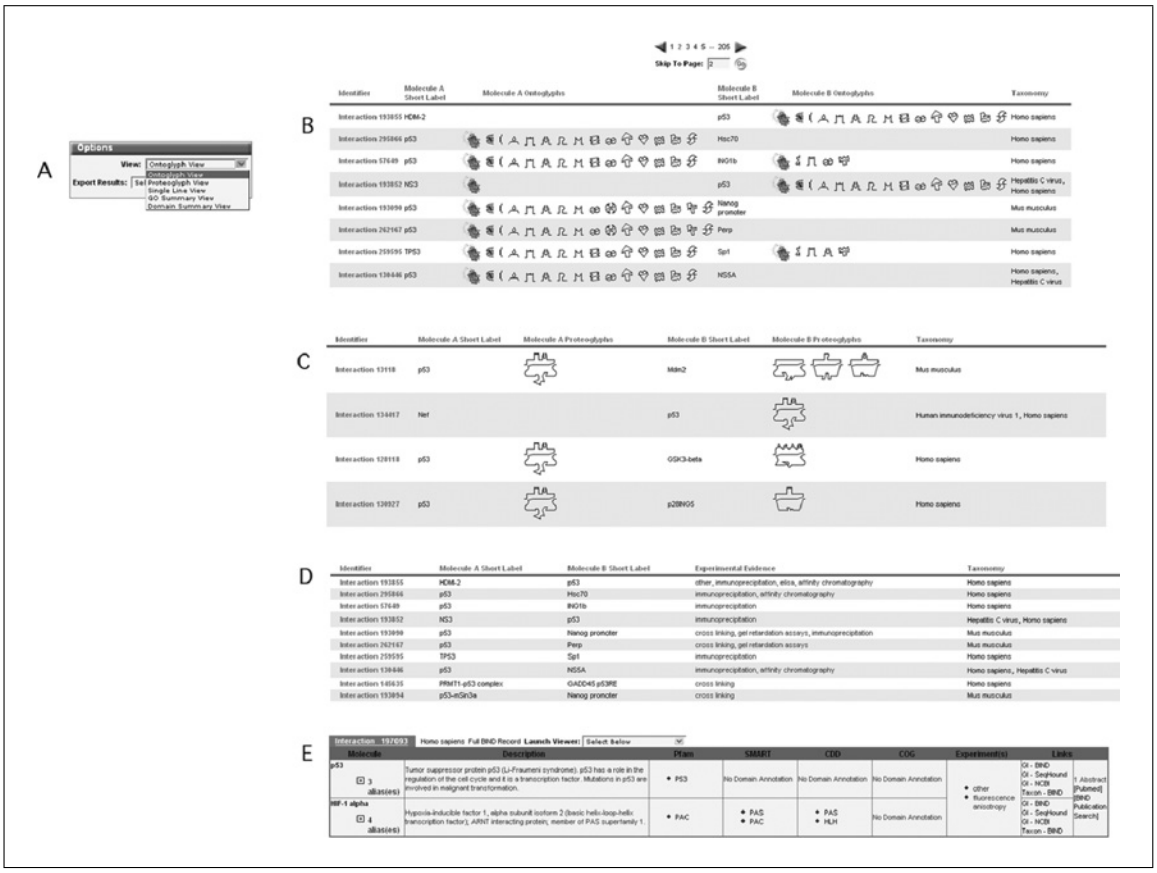


Figure 8.9.7 Multiple methods for viewing BIND search results. (**A**) The Options dialogue that appears at the upper right-hand side of each search list returned by a BIND query. (**B**) OntoGlyph View. (**C**) ProteoGlyph View. (**D**) Single Line View. (**E**) Domain Summary View (similar to GO summary view, not shown).

The choice of which view format to pursue will depend on the type of information the user wishes to see or in what context the information already obtained is understood. For example, the user may be interested in seeing the listing of specific GO terms as text rather than as the symbolic summary provided by OntoGlyphs. Alternatively, one may be looking for interacting molecules within a search result and be interested in those that may have specific structural domains as defined by the SMART, PFAM, COG, or CDD domain databases.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

No local files are required

1. The default setting for viewing search results is the OntoGlyph view (Fig. 8.9.3 and Fig. 8.9.7B). This view provides a one-line summary for each BIND record, which includes the BIND ID, Taxon, Molecule A and Molecule B of the interaction, and the OntoGlyphs for each molecule. Holding the mouse cursor over each OntoGlyph will show the title of the OntoGlyph in English, and if the browser is properly configured, Chinese. An icon is also provided if the participating molecule is a protein with predicted small-molecule binding sites. If the BIND record is a complex, only the Taxon and the number of subunits in the complex are indicated. The OntoGlyphs represent three types of protein attributes: function, binding, and cellular localization. OntoGlyphs are based on a combination of the NCBI's Cluster of Orthologous Groups (COGs) functional categories and Gene Ontology (GO) terms. Clicking on an OntoGlyph brings up a new search window listing all the interactions in BIND with that symbol.

2. The Filters tab (shown in Fig. 8.9.3; not visible in Fig. 8.9.7) can be selected to list the summary of OntoGlyphs from the entire set of search results. An array of OntoGlyphs is presented with the number of times the OntoGlyph annotation is found in the search results in order of annotation frequency. Select the subset of records containing one or more annotations using the controls on this Filters tab. OntoGlyphs (and the features that they represent) may be included or excluded from the search set by coloring the specific OntoGlyph green (one click) or red (three clicks). Coloring the OntoGlyph yellow (two clicks) causes the filter to find records in which the annotation co-occurs in both A and B interacting partners. This is useful for selecting subsets of molecules that are found in the same cellular compartment. To execute the selected Filter, press the Filter button at the bottom of the screen.

3. Select the ProteoGlyph view (example in Fig. 8.9.7C) from the View drop-down list (Fig. 8.9.7A) in the Options box at the top right of the screen illustrated in Figure 8.9.3. This page lists the domains found in each interacting molecule as symbols that contain OntoGlyph icons on the top for binding information, and machine-derived unique symbols for each domain on the bottom. Place the mouse cursor over a ProteoGlyph to see the domain name. Click on a ProteoGlyph to see the set of BIND interactions containing the same domain.

   The Domains tab (visible in Fig. 8.9.3) can be selected to list the summary of ProteoGlyphs in the entire set of search results. An array of ProteoGlyphs is presented with the number of times the domain is found in the search results, in order of the most frequent to least frequent domain. One can select the subset of records containing one or more domains using the controls on this Domains tab. Domains may be included or excluded from the search set by coloring the specific domain green (one click) or red (three clicks).

4. Select the Single Line View (example in Figure 8.9.7D) from the View drop-down list (Fig. 8.9.7A) in the Options box at the top right of the screen illustrated in Figure 8.9.3. This view provides a one-line summary of each BIND record, which includes the BIND ID, Molecule A and Molecule B short labels, the experimental evidence, and the taxonomy of the interacting molecules. If the BIND record is a complex, the Taxonomy and number of subunits in the complex is indicated. The BIND ID is hyperlinked to the detailed Interaction/Complex record. Clicking Taxonomy name leads to an OntoGlyph view of all the records in BIND from that specific Taxonomy. Molecule A and B Short Labels are hyperlinked to a Molecule-Centric View of all the BIND records with the same redundant sequence.

5. Select the GO Summary View (not shown in Fig. 8.9.7D) from the View drop-down menu (Fig. 8.9.7A) in the Options box at the top right of the screen illustrated in Figure 8.9.3. This view provides a detailed summary of each BIND record including a link to the detailed Interaction/Complex record, Molecule A and B Short Labels,

Molecule A and B aliases, Molecule A and B descriptions, and hyperlinked GO annotation (Molecular Function, Cellular Component, and Biological Process) for Molecule A and B. The type of experiment used to demonstrate the interaction is indicated in the Experiment(s) column. Links to other BIND records containing the same sequence as Molecule A and B, as well as links to the SeqHound and NCBI records detailing the individual molecules involved in the Interaction/Complex, are also provided.

6. Select the Domain Summary View (example in Fig. 8.9.7E) from the View drop-down menu (Fig. 8.9.7A) in the Options box at the top right of the screen illustrated in Figure 8.9.3. This view provides a summary of each BIND record, which includes Molecule A and B aliases, Molecule A and B descriptions, and hyperlinked COG, Pfam, SMART, and CDD Domain annotations for Molecule A and B. The type of experiment used to demonstrate the interaction is indicated in the Experiment(s) column. Links to other BIND records containing the same GI as Molecule A and Molecule B are also provided, as well as links to the SeqHound and NCBI records detailing the individual molecules involved in the Interaction/Complex.
7. Click on the domain links under the Pfam, SMART, CDD, or COG fields to open a new window containing the domain record for that molecule from the appropriate database.

VIEWING INTERACTION RECORDS

BASIC PROTOCOL 7

This protocol describes the types of information available in a BIND interaction record and how users can link to other, related sources of information. Once users have identified the molecular interaction(s) of interest, they will likely be interested in learning more about the details of the interaction, e.g., in what paper(s) the interaction was characterized, by what method it was identified, or what information about functionality the interaction provides.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

No local files are required

1. An example single record (Fig. 8.9.8) can be found by typing the BIND identifier `128118` into the query box on the BIND home page (*http://bind.ca*). Otherwise, any BIND identifier link in a list of search results will lead to the single record view. The two fields labeled Molecule A and Molecule B provide information about the molecules involved in the interaction (Fig. 8.9.8). Each molecule contains information on:

   a. Molecule type and short label.
   b. ProteoGlyphs and OntoGlyphs.
   c. Description.

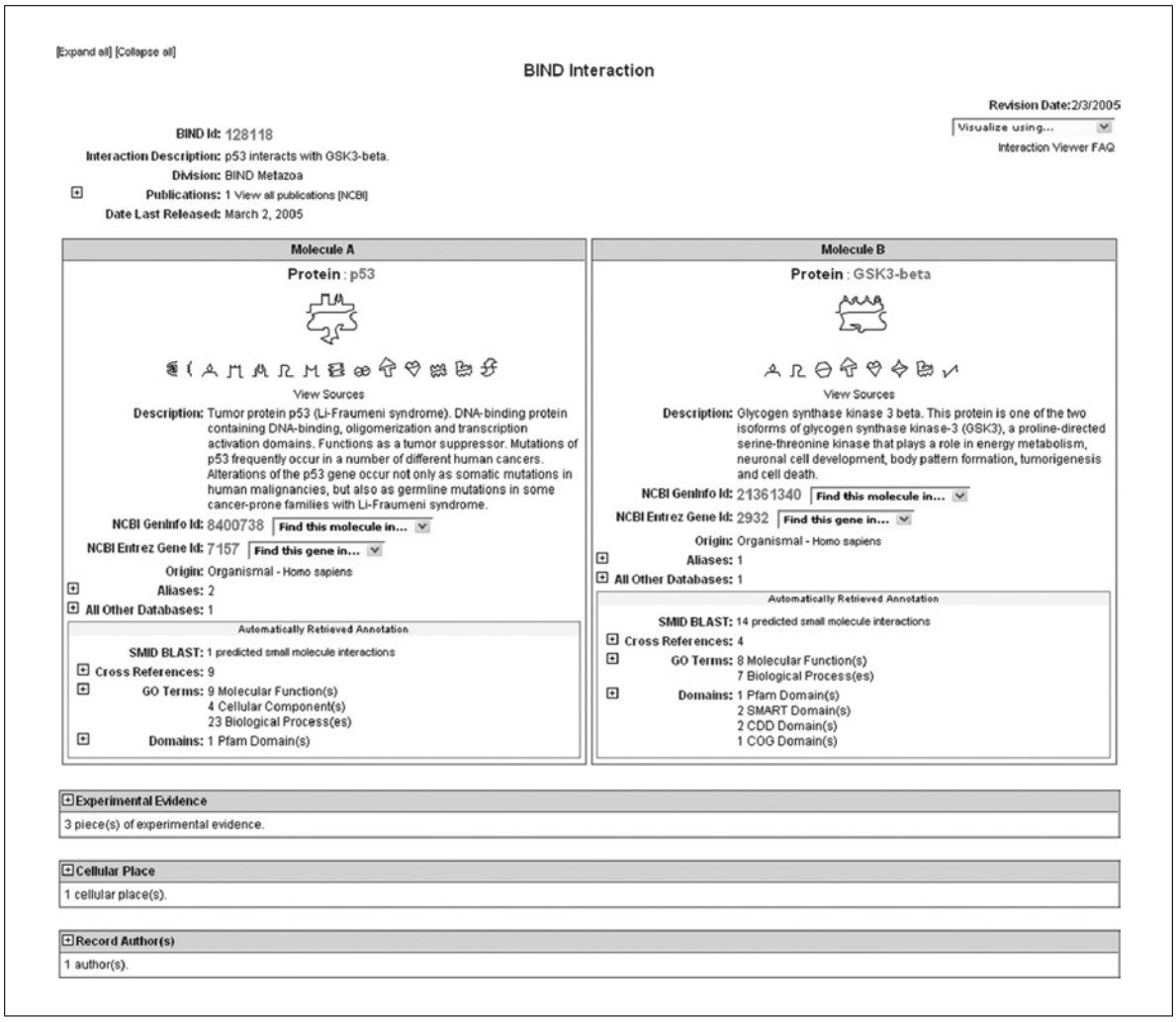


Figure 8.9.8 Single BIND interaction record view, collapsed. The plus signs [+] expand each section to reveal more details. The "Expand all" link at top will expand the entire record.

d. Molecule identifier with drop down list of links.

e. Origin.

f. Aliases.

g. All other databases.

h. Automatically retrieved Annotation including Small Molecule Hits to SMID BLAST, Database Cross References, GO terms, and Domains.

2. Click on the View Sources link to open a new window (not shown in any of the figures) containing the ProteoGlyph and OntoGlyph summary records for that protein. This page contains the list of original GO Annotations, including the GO identifiers, term name, and annotation source, as well as the GO evidence codes. It also contains the list of conserved domains found with the RPS-BLAST tool that have an e-score of <0.01. These tables are used in the derivation of ProteoGlyph and OntoGlyph symbolic summaries. Close this window to return to the window containing the interaction record.

3. If the molecule identifier is a geninfo (GI) number, select SeqHound or NCBI from the "find this molecule in..." pull-down menu to open the molecule record in SeqHound or NCBI, respectively. Select BIND from that same pull-down menu to launch a search for other records in BIND containing that GI. The results will be

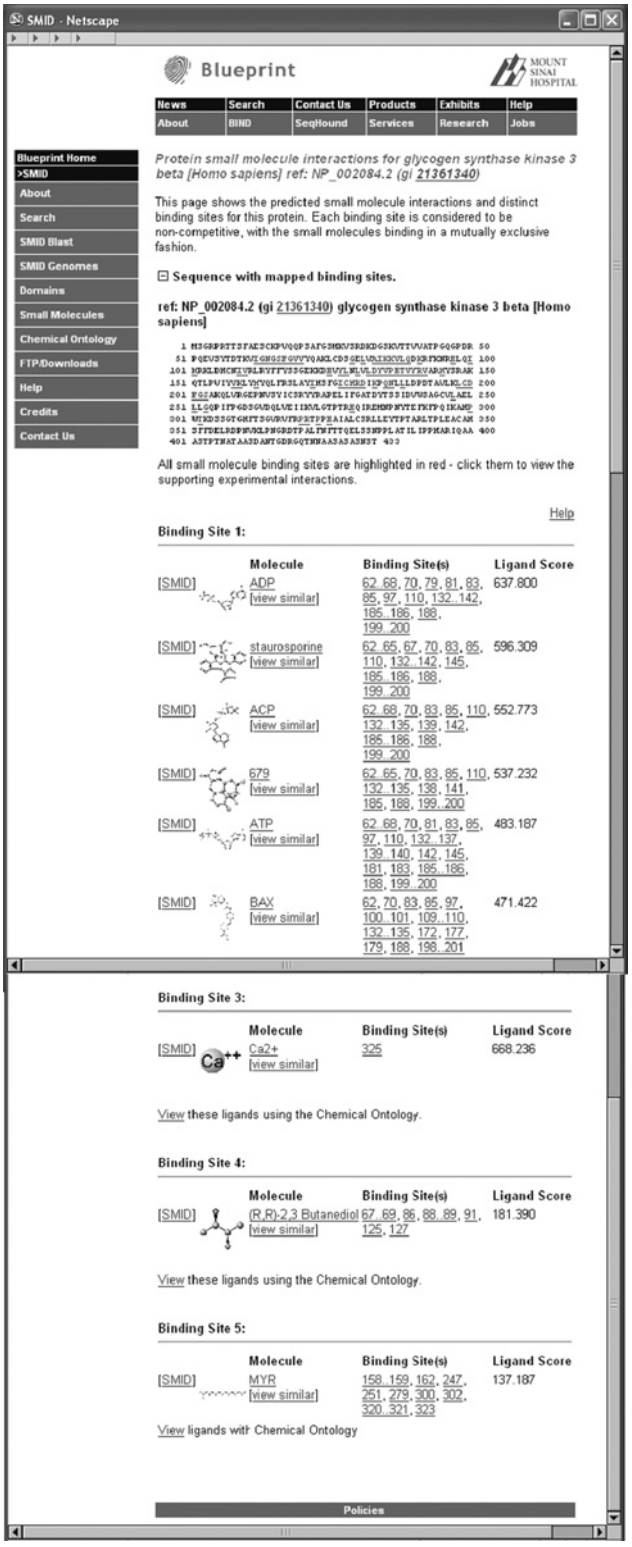


Figure 8.9.9 The link to predicted small-molecule interactions from Figure 8.9.8 leads to the Blueprint Small Molecule Interaction Database (SMID) listing. The protein sequence has been expanded in this figure using the [+] symbol to show binding ligands in color. Binding sites and small molecules are derived by similarity from 3-D structures in the MMDBBIND 3DSM division.

displayed in a new window. Close this window to return to the window containing the interaction record.

4. If the molecule type is a protein, selecting the domain of interest from the SMART/Pfam/CDD/COG Domain(s) (click on the plus sign to the left of the "domains" section of the entry to view the options here) will link to the records in the respective domain databases.

5. If the origin is organismal, click on the organism name to open a new window containing the NCBI Taxonomy Browser record for that organism. Close this window to return to the window containing the interaction record.

6. Click on the plus sign (+) beside Aliases to view a list of alternate names for this molecule.

7. If the molecule has predicted small-molecule interactions based on similarity to a known crystal structure with a bound small molecule, click on the text next to the SMID BLAST line to open a new window showing the list of small molecule sites predicted (shown in Fig. 8.9.9). Click on the plus sign (+) beside the phrase "Sequence with mapped binding sites" to expand the sequence as shown in the figure. Mouse over each colored amino acid residues and a list of the small molecule ligands they are predicted to bind will appear.

8. If the molecule has GO terms, click on the plus sign (+) beside GO Terms to open lists corresponding to Molecular Function, Cellular Component, and Biological Process GO assignments for that molecule.

9. To view the annotation such as experimental evidence, binding sites, chemical action, chemical state, or cellular place, click on the plus sign (+) to view the corresponding annotation for those fields.

10. If the record is a Molecular Complex (example BIND ID 201907), click on the plus sign (+) beside "Sub units" to view the molecules comprising the molecular complex.

    For each subunit, there is a window containing the same information as found for molecules A and B in the interaction record.

11. The window labeled Interaction List contains a list of BIND interaction records that make up the complex. Click on an interaction accession to view an interaction record for that complex. Use the browser Back button to return to the molecular complex record.

12. The list of interaction records will be labeled as either ordered or unordered. If the label is ordered, the interactions occur in the order shown to create the final molecular complex.

BASIC PROTOCOL 8

EXPORTING BIND SEARCH RESULTS FOR USE WITH OTHER SOFTWARE

Once a search has been completed and the search results page is displayed, a few options are available for viewing results. Results can be saved in a variety of formats from a list of search results, or from a single record. The BIND Web interface offers a variety of formats for further processing and analysis of datasets by the BIND user. Lists of search results have been tested and are compatible with large query sets (i.e., 20,000 results or larger); however, patience may be required for the save operation to be completed; download speeds will vary with the user's Internet connection.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Most browsers are suitable for basic BIND searches, but the most recent versions of Microsoft Internet Explorer, Mozilla Firefox, and Netscape Navigator are recommended.

Files

Local files are not required; some are created in this exercise

Exporting all search results

On the right-hand side of a search results page (see, e.g., Fig. 8.9.3), there is an Options box that has two drop-down menus: View and Export Results. The View drop-down menu is described in Basic Protocol 2. Under Export Results, the options include: Comma Separated Values (CSV) files (compatible with Microsoft Excel and other spreadsheets); Cytoscape SIF; PSI level 2; GI Pair (CSV); FASTA sequences; Go Annotator (CSV); Domain Assignment (CSV); DB Cross Reference (CSV); BIND ID (CSV); BIND Flat File; BIND Submit XML; and BIND Submit ASN.1. The use of the Cytoscape SIF export format is discussed as an example in the following steps.

1. To export a CSV file of BIND IDs, GI Pairs, Go Annotator, Domain Assignment, or DB cross reference lists, select the corresponding term from the Export drop-down menu in the Options box.

   This will give the option to save the comma-separated list or to open the list in Microsoft Excel.

2. To export the FASTA format version of the search results, select FASTA Format from the Export drop-down menu in the Options box.

   This will prompt the user to save the document or to open the document in an application of choice.

3. Other formats, such as XML, PSI level 2, ASN.1, and Flat File, give the user many options for data manipulation.

4. Depending on the number of search results, it may take some time for export operations to complete.

Exporting individual records

Once the search has been completed the record to be viewed has been selected, the BIND Web interface offers a variety of formats for further processing and analysis of the individual interaction record.

5. From the single record view (see Basic Protocol 7 and Fig. 8.9.8), the Options box at the upper right hand of the record shows two drop-down lists, Format and Export Results. The options under the Format drop-down menu change the format in which the browser displays the individual BIND record. From the Export Results drop-down menu, users can select from several different formats to export the data, such as HTML, XML, ASN.1, PSI level 2, and Flat File.

   To find a particular field in which a given text term in a raw BIND record, use the ASN.1 Format, then search for the text term using the browser's "Find in this page" (or equivalent text-searching) feature.

6. The Export Results pull-down menu works the same way as it does for multiple records. In addition to the standard formats, one can retrieve a PDF version of the individual interaction record.

BASIC PROTOCOL 9

VISUALIZING INTERACTIONS USING THE BIND INTERACTION VIEWER (BIV)

The BIND Interaction Viewer (BIV) is a tool to visualize and analyze molecular interactions, complexes, and pathways (Fig. 8.9.10). The BIV uses OntoGlyphs to display information about a protein via attributes such as molecular function, biological process, and subcellular localization. OntoGlyphs allows the user to graphically and interactively explore interaction networks by visualizing interactions in the context of 34 functional, 25 binding specificity, and 24 subcellular localization OntoGlyph categories.

Necessary Resources

Hardware

Workstation with connection to the Internet

Software

Internet browser. Up-to-date versions of common browsers are recommended, e.g., Microsoft Internet Explorer, Netscape Navigator, Mozilla Firefox

The BIND Interaction Viewer requires Java 2 Runtime Environment, Standard Edition v.1.4.2 or higher (*http://www.java.com*)

Files

No local files required

1. Access BIND and search the database as described in Basic Protocol 1 for the interaction(s) of interest (e.g., p53 interacting with htt, BIND ID 128190, by typing `128190` in the query box).

2. To obtain the graphical display of the interaction represented in BIND ID 128190, select Interaction Network 3.5 from the "Visualize using . . ." drop-down menu on the far right of the screen.

   The user can also obtain the graphical display of an interaction of interest by clicking on File and selecting one of the following options: "retrieve BIND ID," "Open interaction file," or "Import BIND ID list." The BIND interaction graph view displays the proteins as rectangles with the associated OntoGlyphs and DNA or RNA as straight-edge rectangles, the interaction as a line (edge), and the OntoGlyph legend on the right. Thick lines indicate interactions determined with multiple experiments.

3. To select a specific molecule from the graph view, click on Molecules from the tool bar and chose either "short label" or "molecule type" (proteins, DNA, RNA, or small molecules). Clicking on a specific molecule will also select it. Alternatively, the user can select a molecule based on its OntoGlyph category. Click on the OntoGlyph of interest and only the protein(s) that has the chosen OntoGlyph will be selected in the graph view. The molecule has been selected when it appears with a blue outline.

4. Double click on the selected molecule to extend the interaction network to include any interaction (in BIND) in which the selected protein participates. This can also be done by selecting the molecule of interest (e.g., p53) and then clicking on the Show Interactions tab below the tool bar or by right-clicking on the molecule and selecting "show interactions."

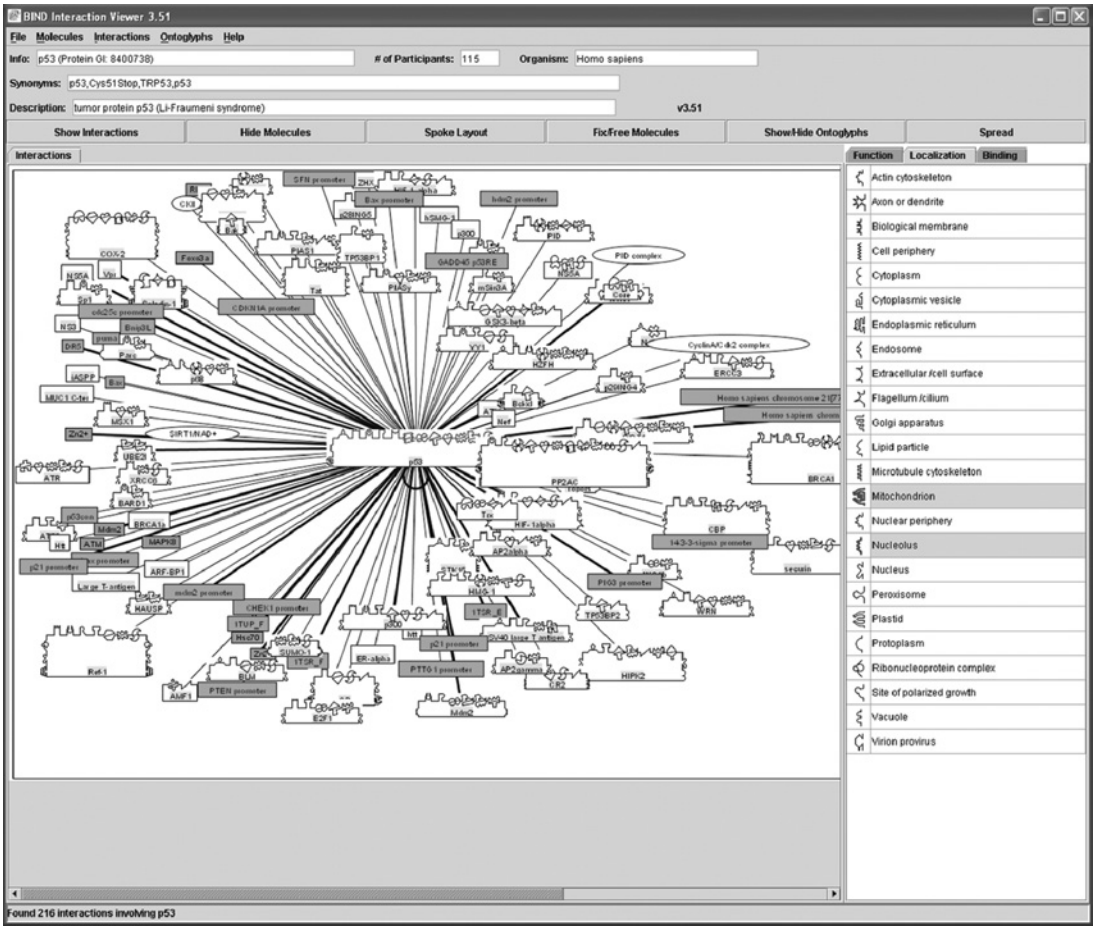


Figure 8.9.10 BIV graphical display of all p53 interactions found in BIND.

Double clicking on p53 will cause the graphical display to adjust to include all the registered interactions of p53. To stop the molecules from moving, click the Freeze tab above the graphical display. Depending on the number of interactions, it may take some time for BIV to draw an image. The results should look like Figure 8.9.10.

5. Several features can change the appearance of the graphical view. Adjust the position of the molecules in the network relative to one other by clicking on the molecule and dragging it to another spot. If the molecule is dragged past the window, scroll bars will enlarge the canvas for the network. Click on the Spoke Layout tab, above the graphical display to change the layout from spread to spoke. The spoke layout repositions neighboring molecules in a circle around the selected molecule(s). Clicking on the Fix/Free Molecules tab fixes or frees the selected molecule(s). A "fixed" molecule is anchored in place and is not moved by the automated layout features such as Spoke Layout or Spread. Clicking on the "Show/Hide OntoGlyphs" tab will display or hide the OntoGlyph view on the selected molecule(s).

6. To hide a molecule or molecules, select a molecule from the interaction network and click on the Hide Molecules tab above the graphical display. To unhide the molecules that have been hidden, click on Molecules from the tool bar and select Show all Hidden.

More than one molecule can be selected by dragging a box over all the molecules of interest or by holding the control key down while clicking on the molecules of interest.

7. The short label of the selected molecule can be changed to known aliases by right clicking on the molecule and selecting a "molecule alias" from the list.

8. Placing the mouse cursor over a glyph on a molecule reveals the meaning of that glyph. Placing the mouse cursor over a line (edge) reveals the two interacting molecules and the BIND ID that corresponds to the interaction record. By clicking on the line, the interaction record will be retrieved for the selected interaction. The thickness of a line (edge) indicates how often the interaction appears in BIND.

9. Search BIND from the graphical display by right clicking on the molecule of interest and select "search BIND with molecule." This uses the molecule identifier to search and retrieves all BIND interaction records that contain this molecule.

10. The OntoGlyph summary can be viewed by right-clicking on the molecule of interest and selecting "view OntoGlyph summary."

    Right clicking on p53 and selecting "view OntoGlyph summary" generates an OntoGlyph summary table specific for p53, which contains a description of the OntoGlyph as well as GO annotations and their source.

11. Right clicking on a specific glyph on the OntoGlyph legend will link to NCBI bookshelves or specialized Web pages, which provide more information about the OntoGlyph.

    Right clicking on the signal transduction icon will take the user to chapter 15, "Signal-Transduction Pathways: An Introduction to Information Metabolism," in Biochemistry, Fifth edition, Berg, J., et al.

12. The legend of OntoGlyphs on the right hand side is active and can be used to select sets of molecules that have specific annotation. The list of selected molecules can be inverted using Invert Selection under the Molecules menu. It is then possible to use the Hide Molecules button to remove molecules with unwanted annotation from the network.

13. It is possible to save the image to a publication-quality vector-graphics file such as a PostScript or PDF or SVG file by using the Export Graphics option under the File menu. A dialog box will allow a filename to be specified, and a pull-down menu will list the options for exporting, including `.jpg`, `.png`, `.raw`, `.ps`, `.svg`, and `.pdf` file formats. Large interaction networks can be reproduced from this viewer.

BASIC PROTOCOL 10

EXPORTING BIND INTERACTION DATA FOR VIEWING WITH CYTOSCAPE OR Cn3D

In many cases, the user will want to integrate BIND data with other visualization tools to see a 2-D or 3-D graphical image of the protein interaction. This is particularly useful when trying to model small-molecule modulators of the interaction for efforts such as drug discovery. These protocols offer steps in visualizing protein interactions from BIND using Cytoscape or Cn3D.

Necessary Resources

Hardware

Workstation with connection to the Internet

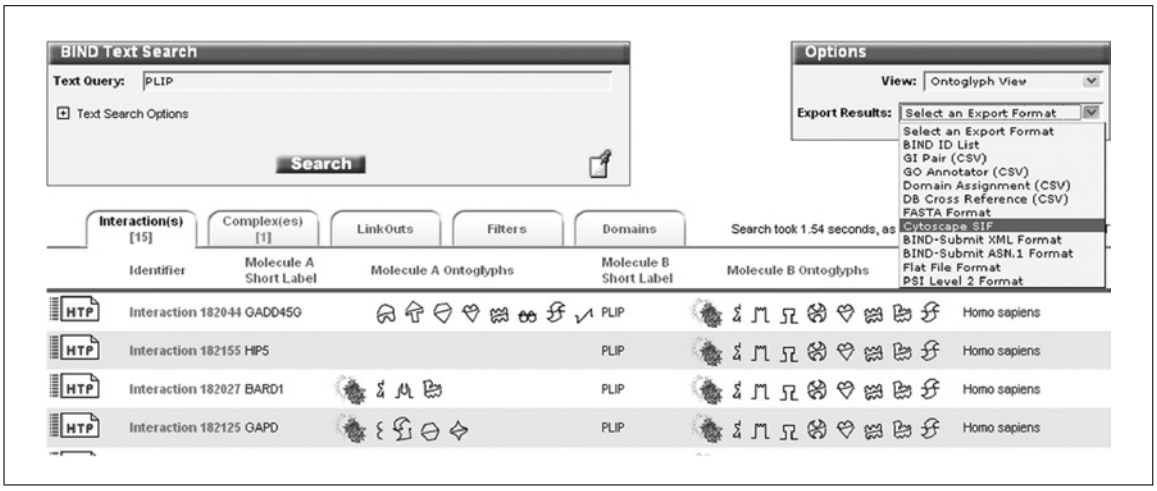


Figure 8.9.11 Downloading search results into a local Cytoscape SIF file.

Software

Internet browser. Up-to-date versions of common browsers are recommended, e.g., Microsoft Internet Explorer, Netscape Navigator.

For Cytoscape: the Cytoscape interaction viewer requires Cytoscape Version 2.0 or higher (*http://www.cytoscape.org*) and Java 1.4.2

For Cn3D: the structure viewer requires Cn3D (*http://www.ncbi.nlm.nih.gov/Structure/CN3D/cn3d.shtml*)

Files

For Cytoscape: Local Cytoscape SIF files required

Exporting BIND interaction data for use with Cytoscape

BIND interaction data can also be transferred to Cytoscape, an open-source bioinformatics software platform for visualizing molecular interaction networks and integrating these interactions with gene expression profiles and other data, e.g., microarray expression data. Cytoscape has a number of plug-in modules that perform advanced algorithms on interaction networks such as shortest-path calculations and complex finding algorithms. Information about using Cytoscape, including an online tutorial, can be found at *http://cytoscape.org*.

1. Access BIND and search the database as described in Basic Protocol 1 for the interaction(s) of interest. For this example, search for all interactions with the text `PLIP`.
2. From the Select an Export Format pull-down menu under Options on the right-hand side of the search results page, select Cytoscape SIF (Fig. 8.9.11).
3. Save search results as a Cytoscape SIF file to the local hard drive. Load Cytoscape 2.0 and open the SIF file that was saved by choosing File/Load/Graph from the tool bar in Cytoscape.

Exporting BIND interactions from 3-D structures for use with Cn3D

BIND has two divisions of interactions arising from 3-D structures obtained from the Protein Data Bank (PDB). These are referred to as MMDBBIND divisions (Salama et al., 2001-2002), as they are derived from NCBI's MMDB database along the way. These records are a complete set of algorithmically generated, fully annotated BIND interaction records with atomic-resolution interaction information. Initially, pairwise

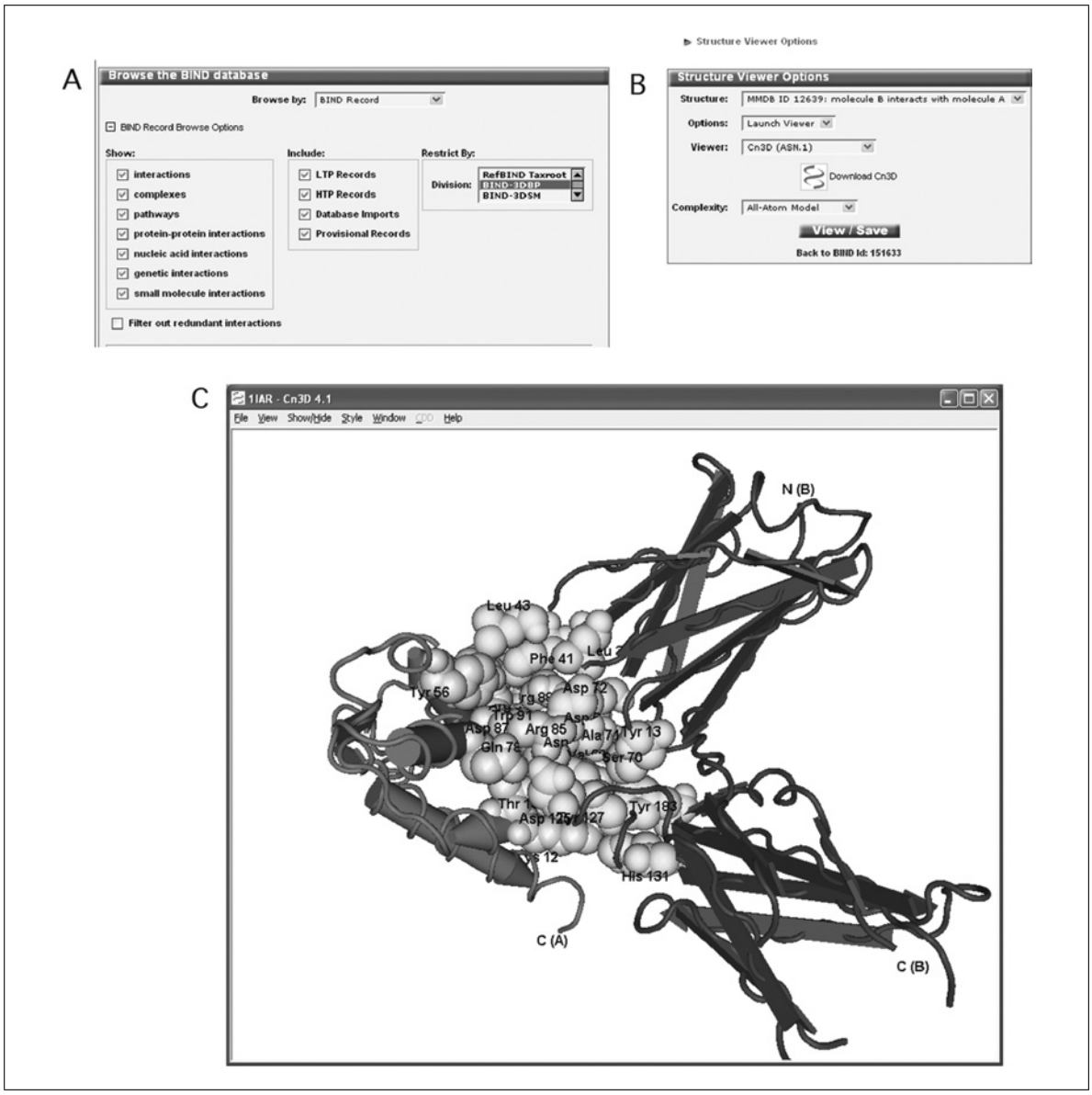


Figure 8.9.12 Searching and viewing 3-D structural interactions. (**A**) Browse the BIND-3DBP division to see the set of BIND biopolymer interactions from 3-D structures. (**B**) The dialog box for launching Cn3D. (**C**) Example of the protein-interaction interface highlighting that BIND provided in the default view of the interaction loaded into Cn3D. For the color version of this figure go to *http://www.currentprotocols.com*.

protein-protein, protein-DNA, protein-RNA, DNA-DNA, DNA-RNA, and RNA-RNA interactions contained within each PDB file were recorded in 3D Biopolymer Division of MMDBBIND (3DBP), with both residue-level and atomic-level detail at the interaction site, and filters were applied to remove most crystal-packing artifacts. Small-molecule interactions are further processed to remove nonbiological small molecules, and ions are filtered with special binding-site classifiers that reduce insignificant hits. The 3D Small Molecule Division (3DSM) is also used for the creation of the Small Molecule Interaction Database (SMID), as well as the resulting small-molecule annotation shown in Figure 8.9.9. Records are additionally annotated with experimental descriptions, and, where available, annotation and short labels, have also been added using Entrez Gene, SGD, and other sources. The BIND interface allows a user to view the 3-D structure of each MMDBBIND record in Cn3D with the interacting residues on each chain highlighted.

4. To find all MMDBBIND interactions in the data set, select Browse by: BIND Record. Expand the BIND Record Browse Options by clicking on [+], and select Restrict by Division: BIND-3DBP (Fig. 8.9.12A), then click the Browse button to obtain the results.
5. Below the heading "BIND interaction" on the right-hand side at the top of the record, click on a structure record of interest (e.g., BIND ID 151633; IL-4Rα interacting with IL-4).
6. To view the structure record, use the "Visualize using" tool and select Structure Viewer from the drop-down menu. A new window will open with the Structure Viewer Options (Fig. 8.9.12B).
7. Select Cn3D from the Viewer drop-down menu (a link is provided). From the Complexity drop-down menu, select the All-Atom Model.
8. Click on View to obtain the 3-D structure. The interaction interface will be highlighted in the 3-D view (Figure 8.9.12C) and in the sequence view (not shown).

COMMENTARY

Background Information

Only by paying diligent attention to user requests have the authors of this unit been able to present a coherent and advanced user interface for BIND. The authors are pleased to note that they received a very positive review (Gilbert, 2005) recently in *Briefings in Bioinformatics* that reinforces their work by stating "One can expect to get useful results that may be well integrated with one's research needs." Readers who are following the protocols in this unit should continue to E-mail the authors at *info@bind.ca* regarding any observations, needs, or desires; they will do their best to help, given the available resources.

Other resources not fully described here include the PreBIND and Textomy systems used for text-mining protein interactions. New work has also been done to provide a comprehensive resource of small-molecule interactions from 3-D structures mapped to genes on the authors' SMID-Genomes system. These and other tools are worth further exploration on the authors' Web site at *http://www.blueprint.org*.

Populating BIND

Published biomolecular interaction data generated by wet lab research efforts is valuable to ongoing research. Simply publishing the data without archiving it in a computationally accessible format, however, results in a lost opportunity to maximize its impact for the research community. BIND allows researchers fast access to interaction data and details about binding site information and kinetic parameters that would otherwise only be available in print and through laborious time-consuming literature searches.

Curation of the detailed information of BIND is one of the key features for its success, and is in sharp contrast to the sparse interaction records produced by other databases either by hand curation or using text-mining tools. Curation activities at Blueprint funded between 2002 and 2005 have provided a substantial body of knowledge in the BIND database and on the use of the database. The growing numbers of citations of the publications for BIND have demonstrated that this information is in demand. The authors' approach to working with journals on BIND curation has been to focus on cooperation with journal representatives and to commit to curating each issue as it is published within the journal's editorial deadlines. The editors and corresponding author are then provided with BIND accession numbers and links to the BIND records. Both journal representatives and authors are invited to provide feedback. By providing this value-added service in an integrated and efficient manner for authors and editors, Blueprint makes the decision to include BIND accession numbers in published text an easy one.

Using this model, Blueprint has established relationships with several high-profile journal publishers, including AAAS/*Science*, Nature Publishing Group, Cell Press, Blackwell Publishing, and SAGE Publications, that amount to ~350 biomolecular interactions per month. Some of these publishers submit prepublication manuscripts to BIND for curation so that the BIND Accession numbers appear in the manuscript when it is published. In such cases, reciprocal links are provided between the publication and the BIND records, allowing for simple on-line user access to the BIND

records directly from journal Web sites. Unlike databases like GenBank, there is no long-term funding currently in place for BIND or other interaction database curation. The authors hope that, working together with other databases in the International Molecular Exchange (IMEX) consortium, they may work towards achieving the long-term funding of interaction curation.

BIND records can also be submitted directly by researchers working in the field by simply requesting a log-in and setting up a My-BIND page. The record is then curated and cross-referenced by a BIND curator to ensure that all pertinent information is included in the record; upon publication of the paper reporting the data, the record is released into the public domain.

Directed curation

BIND also participates and coordinates directed-curation projects (DCPs), which involve concentrating curation activity on a particular area of interest and capturing the global assembly of biomolecular interaction information in the published literature related to an area of interest, for example, a particular disease or a particular family of proteins. The ultimate goal of a DCP is a complete set of highly annotated and computationally relevant interaction and pathway data, deposited into BIND, that the research community will be able to immediately utilize to guide experimentation.

Although many databases and publication sources carry information related to human disease, directed curation offers many benefits that these other sources are incapable of providing. Perhaps the greatest challenge that faces the scientific community is that much of the information is widely distributed across many formats and that little of this information is computationally accessible. With directed curation, however, researchers will have the opportunity to rapidly reach a critical mass of data, and can be confident that their experimental explorations are based on all of the available data and not just on what they could access. By building this critical mass of data, the confidence in both experimentally derived and predicted interactions maps is expected to increase. Also, the accomplishment of the DCP goals will allow clearer visualization of complex pathways represented as a network of interactions. Analysis of pathway data will lead to new insights regarding the causes and treatment of disease, and also regarding protein function, potential therapeutics, and the identity or nature of unidentified modulators.

As an example, in August 2004, Blueprint Asia announced a collaboration with the Novartis Institute for Tropical Diseases (NITD) to facilitate the company's research into dengue fever, a debilitating infectious disease. Dengue fever is a mosquito-borne viral infection that causes fever and severe joint pain and, in more severe cases, can lead to hemorrhage, shock, and ultimately death. Prevalent in tropical and subtropical regions, the disease affects 50 million people across five continents, and infection rates are increasing dramatically. A large proportion of the estimated 500,000 cases that require hospitalization each year are children. Scientists who wish to explore this set can type in the code `BPA001` in the BIND text search box and reveal the 304 Interactions and 44 Complex records that comprise Dengue Fever and flaviviridiae interactions. Interested parties who would like work on collaborations to fund and undertake a directed curation project are encouraged to write to the authors of this unit at *info@bind.ca*.

Model organisms

BIND has also engaged several top genome databases to coordinate their submission, curation, and presentation efforts into BIND. The genome databases include information from a variety of model organisms including yeast, fly, mouse, and rat. The objective of these collaborations is to better integrate genomic and interaction data and thereby provide a unified structure and search mechanism to facilitate researcher access to this information.

It is only by marrying genomic and interaction data that a complete understanding of mechanisms within the cell can be reached. By engaging with such key model organism database organizations, Blueprint-NA has sought to present and deploy BIND as the bridge between these research communities.

Examples of current relationships include interactions with Saccharomyces Genome Database, Mouse Genome Informatics, Rat Genome Database, and FlyBase.

Pathways

Every biological event is the result of a signal being passed from one portion of a cell to another or from one cell to another. Thus, it is critical that scientists be able to identify and understand the myriad steps in these signaling pathways. One of the challenges of making this goal a reality, however, is the development of tools and methods for scientists to use in elucidating these complex biomolecular pathways. Thus, as with the model

organism collaborations, BIND also works with pathway database groups to cross-reference biomolecular data. Examples include *Science* STKE, EcoCyc, aMAZE, PID, and AfCS, with the aim of working towards pathway curation standards, data exchange, and reciprocal hyperlinks. Developing BIND so that it can faithfully archive the contents of pathway databases and allow for the integrated query and visualization of pathways, complexes, and interactions remains the long-term vision of the authors, as this information and its integration is an absolute requirement for achieving detailed computational models of cellular activity.

Downloading BIND Data

Files containing BIND data and supporting documentation (e.g., specifications, curation manuals, etc.) can also be downloaded freely from the BIND FTP site, which can be accessed from the main BIND Web page by clicking on the folder icon or directly from the FTP site at *ftp://ftp.blueprint.org/pub/BIND/*.

Critical Parameters and Troubleshooting

While use of BIND may seem daunting to the uninitiated, the authors have made every effort to ensure that users are provided with thorough documentation and tutorials to supplement their own exploratory efforts. More than any other database of its kind, the BIND repository and the curation standards and practices used to fill it have been documented in a series of user manuals that are freely available on the BIND Web site (*http://www.blueprint.org/bind/bind_publications.html*).

Furthermore, to assist users in finding the records of interest or to explain how best to use BIND, the authors have set up a series of tutorials that offer users a step-wise guide to BIND and its related tools. These tools can be found on the BIND Help page (*http://www.blueprint.org/bind/bind_help.html*).

And finally, when all else fails, or when the user is uncertain as to how best to phrase the query, BIND offers a User Services function that can be accessed by sending email to *info@bind.ca*. Upon receipt of the E-mail, a User Services Coordinator will either directly answer the query or will redirect it to a specialist at BIND, and oftentimes to the Principal Investigator who can best address the concern, question, or problem. It is by interacting directly with the user community that BIND developers and curators best understand the needs of the scientific community and respond with new data standards or BIND interface tools.

Acknowledgements

Cheryl Wolting, Edwin Haldorsen, Farah Juma, Rosa Pirone, and Martha Bajec contributed to preparing the instructional material in this article in their capacity of User Services and Curation staff while employed at the Blueprint Initiative. Funding for BIND between 2002 and 2005 was provided by Genome Canada through the Ontario Genomics Institute, and by the Ontario R&D Challenge Fund. Funding for Blueprint Asia curation 2004-2005 was provided by the Economic Development Board of Singapore. Funding for BIND from 1999 to 2006 has been provided by the Canadian Institutes of Health Research in a grant to CWVH.

Literature Cited

Alfarano, C., Andrade, C.E., Anthony, K., Bahroos, N., Bajec, M., Bantoft, K., Betel, D., Bobechko, B., Boutilier, K., Burgess, E., Buzadzija, K., Cavero, R., D'Abreo, C., Donaldson, I., Dorairajoo, D., Dumontier, M.J., Dumontier, M.R., Earles, V., Farrall, R., Feldman, H., Garderman, E., Gong, Y., Gonzaga, R., Grytsan, V., Gryz, E., Gu, V., Haldorsen, E., Halupa, A., Haw, R., Hrvojic, A., Hurrell, L., Isserlin, R., Jack, F., Juma, F., Khan, A., Kon, T., Konopinsky, S., Le, V., Lee, E., Ling, S., Magidin, M., Moniakis, J., Montojo, J., Moore, S., Muskat, B., Ng, I., Paraiso, J.P., Parker, B., Pintilie, G., Pirone, R., Salama, J.J., Sgro, S., Shan, T., Shu, Y., Siew, J., Skinner, D., Snyder, K., Stasiuk, R., Strumpf, D., Tuekam, B., Tao, S., Wang, Z., White, M., Willis, R., Wolting, C., Wong, S., Wrong, A., Xin, C., Yao, R., Yates, B., Zhang, S., Zheng, K., Pawson, T., Ouellette, B.F., and Hogue, C.W. 2005. The Biomolecular Interaction Network Database and related tools 2005 update. *Nucl. Acids Res.* 33:D418-424.

Gilbert, D. 2005. Biomolecular interaction network database. *Brief. Bioinform.* 6:194-198.

Hermjakob, H., Montecchi-Palazzi, L., Lewington, C., Mudali, S., Kerrien, S., Orchard, S., Vingron, M., Roechert, B., Roepstorff, P., Valencia, A., Margalit, H., Armstrong, J., Bairoch, A., Cesareni, G., Sherman, D., and Apweiler, R. 2004. IntAct: An open source molecular interaction database. *Nucl. Acids Res.* 32:D452-D455.

Salama, J.J., Donaldson, I., and Hogue, C.W. 2001-2002. Automatic annotation of BIND molecular interactions from three-dimensional structures. *Biopolymers* 61:111-120.

Xenarios, I., Salwinski, L., Duan, X.J., Higney, P., Kim, S., and Eisenberg, D. 2002. DIP: The Database of Interacting Proteins: A research tool for studying cellular networks of protein interactions. *Nucl. Acids Res.* 30:303-305.

Zanzoni, A., Montecchi-Palazzi, L., Quondam, M., Ausiello, G., Helmer-Citterich, M., and

Cesareni, G. 2002. MINT: A Molecular INTeraction database. *FEBS Lett*. 513:135-140.

Internet Resources

http://bind.ca

Web site for the Biomolecular Interaction Network Database (BIND).

http://mint.bio.uniroma2.it/mint/

Web site for the Molecular Interactions (MINT) Database.

http://dip.doe-mbi.ucla.edu

Web site for the Database of Interacting Proteins (DIP).

http://www.adobe.com

Web site for Adobe Acrobat Reader.

http://www.cytoscape.org

Web site for Cytoscape.

http://www.ebi.ac.uk/intact/index.jsp#ack

Web site for the IntAct Project.

http://www.java.com

Web site for Java.

http://www.microsoft.com

Web site for Microsoft Office (including Excel).

http://www.ncbi.nlm.nih.gov/Structure/CN3D/cn3d.shtml

Web site for the NCBI Structure group's Cn3D.

Contributed by Randall C Willis and Christopher W.V. Hogue
The Blueprint Initiative
Samuel Lunenfeld Research Institute
Mount Sinai Hospital
Toronto, Ontario, Canada

Active Site Profiling to Identify Protein Functional Sites in Sequences and Structures Using the Deacon Active Site Profiler (DASP)

With the exponentially increasing sizes of protein sequence and structure databases, the annotation of the functions of these sequences and structures is an ever-increasing problem. The commonly used method of annotation transfer for function identification is error prone (Hegyi and Gerstein, 2001; Rost, 2002; Baxter, et al., 2004), and even proteins with >50% sequence identity can exhibit different functions. Furthermore, for many applications, including substrate analysis or inhibitor identification in the pharmaceutical industry, simple identification of a general function is not enough. The most useful function annotation methods will characterize the functional site features and allow the user to analyze those features, so that details, e.g., substrate or inhibitor specificity, can be identified. A near-term goal is the automated identification of residues that affect substrate or inhibitor binding, the specificity determinants.

Active site profiling was developed to allow the analysis of the functional sites' features and the conservation or variation of those features across the protein family (Cammer et al., 2003). The first step in active site profiling is to create a functional site signature, i.e., an extraction of the sequence features in the structural vicinity of a functional site. An active site profile is the sequence alignment of these signatures for a given set of proteins, usually a protein family. The method thus reveals similarities and differences among functional sites across protein families and allows the user to identify potential specificity determinants of the functional site. The active site profile can be used as the basis for clustering to identify relationships between proteins with related functional sites. This is analogous to what is done with multiple sequence alignments to identify homologous proteins.

Basic Protocol 1 describes the creation of the active site profile (ASP), and Basic Protocol 2 describes the use of that ASP to search sequence databases for similar motifs (see Fig. 8.10.1). In the first protocol ASPs are created from proteins of known structure; thus, to perform Basic Protocol 1, a small number of examples of the functional site must be known, and proteins containing the functional site must be present in the structure database. The second protocol uses the ASP to create a position-specific scoring matrix (PSSM), a matrix that describes the frequency of occurrence of any amino acid at each position (Gribskov et al., 1987), which is then used to search protein sequences for similar motifs. Proteins in the sequence databases with similar functional sites are easily identified with this protocol. Analysis of these more complete ASPs can aid in understanding functional mechanisms and specificity determinants across the entire family, not just those represented in the structure database. In addition, if only two structures are known initially, this second protocol can be used to search the sequences contained in the structure database for similar functional sites, in a boot-strapping approach, to create a more robust profile.

Contributed by Jacquelyn S. Fetrow
Current Protocols in Bioinformatics (2006) 8.10.1-8.10.16

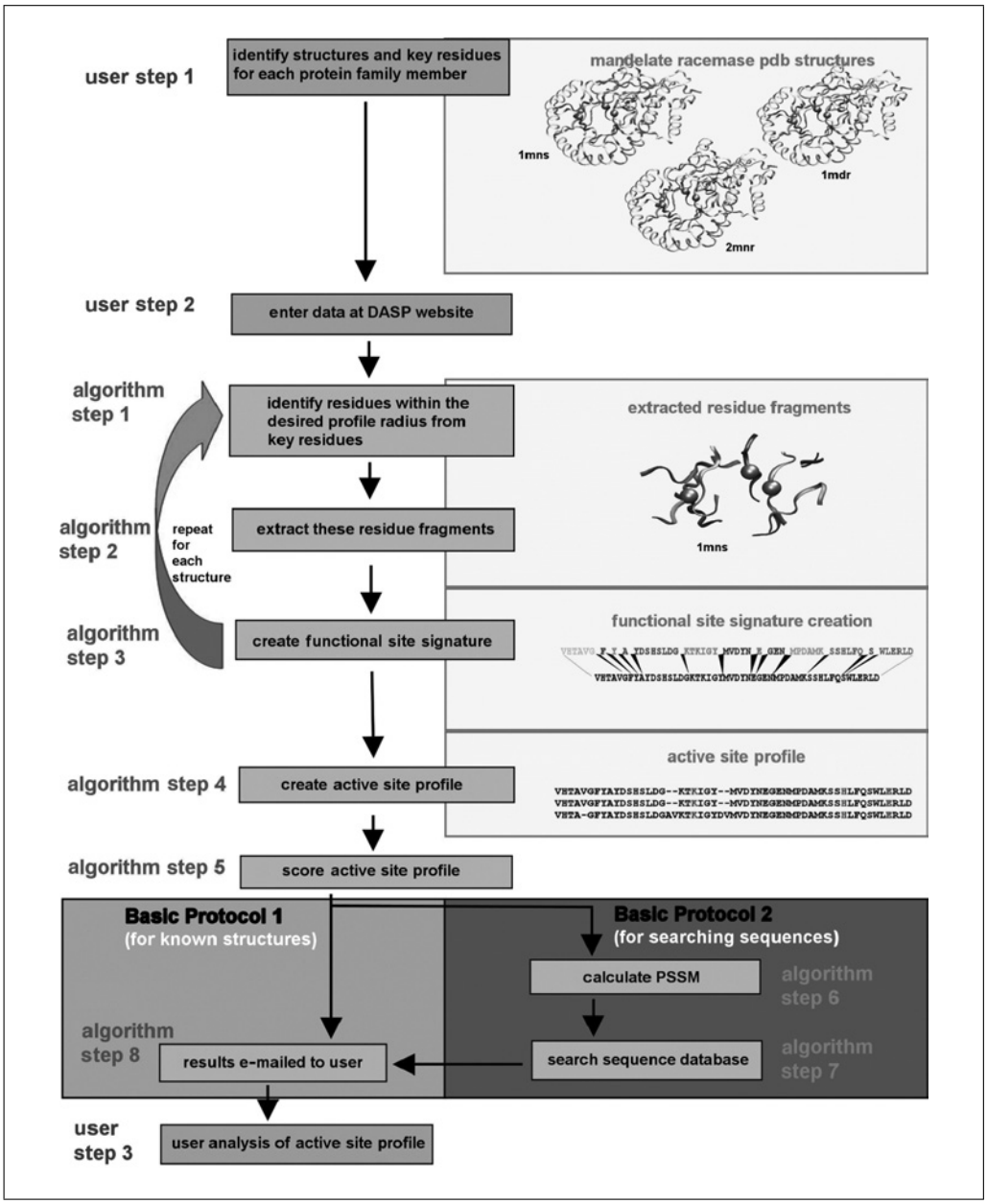


Figure 8.10.1 Schematic representation of the user and algorithm steps in Basic Protocols 1 and 2. Pink boxes and arrows indicate steps performed by the program algorithm, blue boxes and arrows indicate steps performed by the user. Gray boxes indicate two ways to use DASP: (1) searching for signatures in known structures as in Basic Protocol 1 and (2) using an ASP to search sequences for similar signatures as in Basic Protocol 2. The green and yellow boxes on the right illustrate some steps applied to the mandelate racemase protein family. (see Fig. 8.10.5 for the ASPs identified). Protein structures for three mandelate racemases are shown at the upper right, labeled with their pdb filenames 1mns, 2mnr, and 1mdr. A closer view of the active site for 1mns is shown underneath. For the color version of this figure go to *http://www.currentprotocols.com*.

CONSTRUCTION OF THE ACTIVE SITE PROFILE FOR A FUNCTIONAL SITE

A user will follow Basic Protocol 1 to compare the features around a common functional site in proteins of known structure. This protocol describes the creation of a functional site signature for each protein of known structure followed by an alignment of those signatures to create the active site profile (ASP) for the functional site of interest.

Necessary Resources

Hardware

Computer with Internet access. The type of machine is not limiting, although, because this is a client-based approach, if the user's machine is slow, performance will be slow, and search times will be long (particularly searches of GenBank). Use of DASP with the Mac OS has not been tested.

Software

Internet browser (e.g., Internet Explorer, *http://www.microsoft.com*, or Mozilla, *http://www.mozilla.org*) that supports Java 1.5 (*http://www.java.com*)

DASP: available through the Deacon Active Site Profile (DASP) Web site (*http://dasp.deac.wfu.edu*)

E-mail system capable of handling returned files (generally small)

Files

Protein Databank (PDB; see *UNIT 1.9*) file names and key residues for at least two protein structures that are known to exhibit the function of interest. PDB files are currently extracted from the PDB database stored on the DASP server and may not always be as up-to-date as the actual PDB Web site. Newer PDB files may occasionally not be available.

NOTE: Figure 8.10.1 illustrates the various user steps and algorithm steps involved in this protocol.

User step 1: Identify structures and key residues for each protein family member

1. For each functional site of interest, identify at least two protein structures that contain the functional site and identify one or more key residues in each structure that are essential, or important to the chemistry, binding, or other aspect of the function.

   Key residues are usually conserved within the protein family. For enzyme active sites, key residues usually selected are structurally conserved and crucial to the protein's chemistry and its catalytic function (Cammer et al., 2003; Baxter et al., 2004; Huff et al., 2005). The residues do not necessarily have the same identity in each protein, but their locations and spatial relationship in the protein should be conserved, e.g., as described for the fuzzy functional form motifs (FFFs; Fetrow and Skolnick, 1998; Fetrow et al., 1998).

   Key residue identification is based on the user's expert knowledge, literature analysis, analysis of mutant data, sequence and structure comparisons, or analysis of functional motifs. As an example, the author analyzed the mandelate racemase enzyme active site using the procedure outlined in Figure 8.10.1. Three PDB files containing this active site (shown in Fig. 8.10.1) were initially identified and residues essential to the catalytic activity were identified by literature analysis and structural comparison (Table 8.10.1). More details about key residue selection can be found in the Critical Parameters: Key residues.

Table 8.10.1 PDB Files and Key Residues for the Enzyme-Active Site of the Mandelate Racemases

	1mdr	1mns	2 mnr	Selection criteria
Key residue 1	Lys 166	Lys 166	Lys 166	Catalytic base; works with His 297 to abstract proton from substrate
Key residue 2	His 297	His 297	His 297	Catalytic base; works with Lys 166 to abstract proton from substrate
Key residue 3	Glu 317	Glu 317	Glu 317	General acid

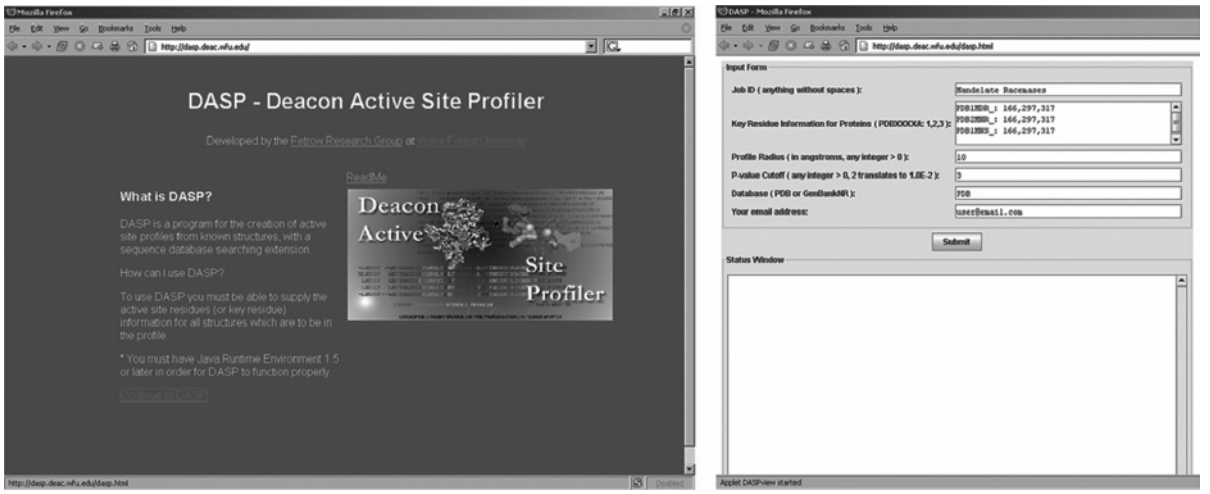


Figure 8.10.2 Screen shot of the DASP Web site and data input page. On the left is the Web page that the user should see upon going to the Web site *http://dasp.deac.wfu.edu.* The data input page that the user sees upon clicking the "Continue to DASP" button is shown on the right. The data necessary for applying Basic Protocol 2 to the mandelate racemases is shown in the input fields. These input data were used to obtain the ASP shown in Figure 8.10.5A (top).

User step 2: Enter data at DASP Web site

2. Go to the DASP Web site (*http://dasp.deac.wfu.edu*). The homepage (shown in Fig. 8.10.2) provides general information about the Web site. Click on "Continue to DASP" to enter the data entry input screen (also shown in Fig. 8.10.2).

3. Enter the following information:

   a. Job identification number (any combination of numbers or letters, without spaces).

      This number is designed to aid the user's organization of multiple job submissions and should be something meaningful to the user.

   b. PDB file names and key residues for each PDB file name and the residue numbers that identify the key residues (identified in Basic Protocol 1, step 1) in the format specified on the Web site.

      The format on the Web site must be followed exactly, as shown in Figure 8.10.2. Also, be careful to enter the correct residue number for each residue from the PDB file.

   c. Profile radius in angstroms (most often the default value of 10 Å).

      There may be instances where an inhibitor or substrate extends farther from the key residues than 10 Å, and in this case the user may wish to use a different search radius.

d. E-mail address (the address to which the results are e-mailed).

Note that for Basic Protocol 1, there is no need to enter a p-value or database. These are required only for Basic Protocol 2.

Algorithm step 1: Identify residues within the desired profile radius from key residues

4. The center of mass is calculated for each key residue. Any residue that contains at least one atom within the profile radius of each center of mass is identified.

Algorithm step 2: Extract residue fragments

5. Each residue, containing an atom within the profile radius is extracted. When this procedure identifies consecutive residues (which is often the case), the complete fragments are extracted.

Algorithm step 3: Create functional site signature

6. The fragments (or occasionally single residues) are concatenated from the N-terminal fragment to the C-terminal fragment to form a *functional site signature* for this protein, as illustrated in Figure 8.10.1. Algorithm steps 1 to 3 are repeated for each protein structure that was entered by the user.

Mandelate racemase active site signatures are shown in Figure 8.10.1.

A functional site signature consists of the protein fragments extracted from the vicinity of a given active site, and concatenated to form a linear sequence (as illustrated in Fig. 8.10.1).

The author's own research has largely focused on enzyme active sites, but the protocol is not limited to enzyme active sites. It can be applied to any site associated with a molecular function, e.g., ligand or cofactor binding sites. Recently, the author has been exploring the ATP-binding site of kinases using this method (Ahlers and Fetrow, unpub. observ.).

Algorithm step 4: Create active site profile

7. After a signature is created for each protein, the DASP program aligns the sequences of the signatures (using ClustalW, v 1.81; Higgins et al., 1996) to create the active site profile (ASP).

The mandelate racemase active site ASP is shown in Figure 8.10.1.

Algorithm step 5: Score active site profile

8. The ASP score is calculated, as described by Cammer et al. (2003), by evaluating the variation in residue types for each functional site residue for four conditions (residue variation assigned by ClustalW) as follows: identity, $(S_I) = +1.0$; strongly conserved, $(S_S) = +0.2$; weakly conserved, $(S_W) = +0.1$; and gap, $(S_G) = -0.5$. The values at each residue position are summed to generate a score that is then normalized by the number of positions in the active site as in the following equation where:

$$\text{Score} = \frac{\sum_{1}^{n} S_I + \sum_{1}^{m} S_S + \sum_{1}^{k} S_W + \sum_{1}^{l} S_G}{N}$$

S_I is the score for positions that are fully conserved and n is the number of such positions along the profile;
S_S is the score for the positions that are strongly conserved and m is the number of these positions along the profile;
S_W is the score for the positions that are weakly conserved and k is the number of those positions along the profile;
S_g is the score for each gap and l is the number of gaps along the profile;
N is the number of positions in the ASP.

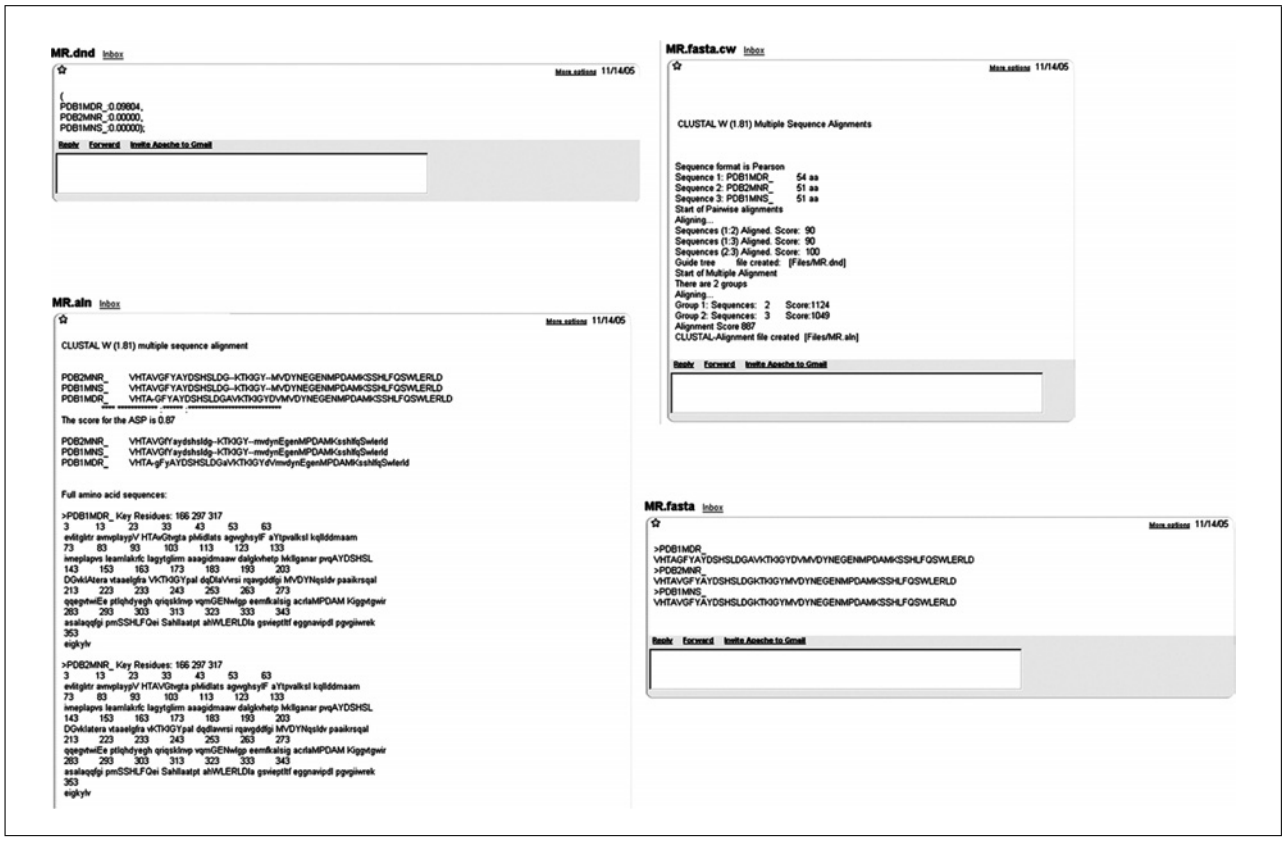


Figure 8.10.3 Examples of files that are e-mailed to the user as a result of Basic Protocols 1 and 2. The files containing the results for applying Basic Protocol 1 to the mandelate racemases protein family are: `MR.dnd` (upper left), `MR.fasta.cw` (upper right), `MR.aln` (lower left), and `MR.fasta` (lower right). Contents of the files are described in the text.

Value assignments for these parameters have been derived empirically such that fully conserved identities dominate the score (Cammer et al., 2003).

The score for the initial mandelate racemase enzyme active site ASP using the three initial proteins (Fig. 8.10.5) is 0.86.

Algorithm step 8: E-mail results to user

9. The results are then e-mailed to the user. For Basic Protocol 1, the user will receive four separate e-mails with subject lines of `[filename].aln`, `[filename].fasta`, `[filename].dnd`, and `[filename].fasta.cw`.

Receipt of the e-mail for this process is usually rapid (on the order of minutes), but, because this is not a job submission service, timing is dependent on the user's machine. DASP is implemented as a Java applet, so the code is downloaded to and executed on the client's computer; therefore, the performance of DASP is solely based on the power of the user's computer.

The `[filename].aln` *file contains the main results: the signatures for each input protein (aligned into a profile), the profile score, and the complete sequence for the protein, with the fragments that are part of the functional site signature identified by upper case letters, so the user can identify the location of the active site fragments within the entire sequence. The* `[filename].fasta` *file contains the sequences of the signatures in FASTA format to facilitate input into other programs. The* `[filename].dnd` *file is output from the ClustalW alignment of the functional site signatures and contains information for creating a dendrogram of the functional site signatures. The* `[filename].fasta.cw` *file captures the screen output of ClustalW, and it displays information about input sequence length, pairwise percent identity between each input sequence, and the multiple sequence alignment score.*

Screen shots of these files for the mandelate racemase family are shown in Figure 8.10.3. The e-mail delivery system is known to work with both the Linux and Windows operating systems. E-mail to the Mac OS has not been tested.

User step 3: Analyze active site profile

10. After e-mail receipt, the user analyzes the returned results. An ASP score of 0.25 or higher is usually indicative of a good profile that exhibits a relatively conserved functional site. Lower scores might indicate the presence of a false positive in the identified proteins or the identification of a diverse protein superfamily or suprafamily (see Guidelines for Understanding Results for details).

    Family versus superfamily ASPs: the cutoff score of 0.25 works for most protein families that the author has identified. However, larger, more diverse families, or super- or supra-families will have a lower score and the ASP alignment will not be robust. If the score is low, or the alignment is weird, analysis of a superfamily may be occurring. In this case, identify the common subgroups (hierarchical clustering is a good tool for this), and resubmit each subgroup to the DASP site individually to calculate ASPs for each subfamily.

USE OF THE FUNCTIONAL SITE PROFILE TO SEARCH THE SEQUENCE DATABASE

BASIC PROTOCOL 2

Basic Protocol 2 extends Basic Protocol 1 by creating a position-specific scoring matrix (PSSM) from the ASP and using the PSSM to search a sequence database (either GenBankNR or the protein sequences from the PDB structures; data sets located on the DASP Web server) for proteins with related functional site signatures. In this way, the information about a functional site from the known three-dimensional structures can be used to search sequence databases where the structure of the protein may be unknown. This allows the user to identify functional sites of potential interest in sequences that may or may not have any functional information associated with them.

If a user has created an ASP using Basic Protocol 1, the user simply enters the same information that was entered for Basic Protocol 1. The ASP is recalculated, followed by the calculation of the PSSM and the sequence search. In addition to the output from Basic Protocol 1, the output of Basic Protocol 2 also includes a list of sequences (above a user-identified cutoff score) that contain the fragments found in the original ASP.

Necessary Resources

Hardware

Computer with Internet access. The type of machine is critical for computationally intensive and potentially time-consuming sequence database searches. E-mail to the Mac OS has not been tested.

Software

Internet browser (e.g., Internet Explorer, *http://www.microsoft.com*, or Mozilla, *http://www.mozilla.org*) that supports Java 1.5 (*http://www.java.com*)

DASP: available through the Deacon Active Site Profile (DASP) Web site (*http://dasp.deac.wfu.edu*)

E-mail system capable of handling returned files (megabyte size)

Files

PDB sequences obtained as a FASTA-formatted file from the National Center for Biotechnology Information (NCBI; *http://www.ncbi.nlm.nih.gov*). The user must provide the same information that was provided for Basic Protocol 1 (i.e., PDB filenames and key residues) and identify the sequences to be searched (currently either GenBankNR or PDB sequences can be searched).

NOTE: Figure 8.10.1 illustrates the various user steps and algorithm steps involved in this protocol.

User step 1: Identify structures and key residues for each protein family member

1. Identify the PDB and key residues for the function of interest, as described in Basic Protocol 1, step 1.

User step 2: Enter data at DASP Web site

2. Access the DASP Web site (*http://dasp.deac.wfu.edu*). Enter the information described in Basic Protocol 1, step 2 and in addition include the following:

   a. *p*-value cutoff.

      The p-value is a score that indicates the significance of the potential match between a sequence and the ASP. Enter the integer value of the exponent for the cutoff score. For example, if the user desires the cutoff score to be 10^{-4}, the integer 4 should be entered in the Web page. The purpose of the p-value cutoff is to limit the number of sequences that are returned to the user. Generally, the author inputs a value of 3 or 4 (corresponding to cutoffs of 10^{-3} or 10^{-4}, respectively). Smaller values (such as 1 or 2) will return many sequences.

      The p-value is an important parameter. A larger (less significant) p-value does not significantly increase the search time (because the search is exhaustive), but it does affect both the creation and e-mailing of the output file. If the output is large, the I/O becomes limiting and the creation of the output file can take a significant amount of time. In addition, if the user's e-mail service limits the size of file attachments, then a large output file might never reach the user's e-mail box.

   b. database.

      Currently, the user can search either the GenBank sequences or the sequences in the PDB data files. These databases are stored locally on the DASP Web server and are not updated as often as the actual databases. The PDB sequences that are searched were obtained as a FASTA-formatted file from NCBI, not from the PDB Web site itself.

Algorithm steps 1, 2, 3, 4, and 5

3. Perform these steps as described in Basic Protocol 1 and outlined in Figure 8.10.1.

Algorithm step 6: Calculate PSSM

4. Each motif is identified from the ASP. A motif represents the alignment of each fragment from the protein structures. A position specific scoring matrix (PSSM) is calculated for each motif or fragment, basically as previously described (Huff, 2005).

   PSSMs provide a method for finding the position in a query sequence which best matches a motif (Gribskov et al., 1987). A PSSM is typically a $20 \times n$ matrix, where 20 is the number of standard amino acids and n is the number of columns in the multiple sequence alignment (Bailey and Gribskov, 1998b). Each cell contains the score given to the corresponding amino acid when found in the corresponding position.

Algorithm step 7: Search sequence database

5. Each PSSM (one for each motif or fragment in the ASP) created is then matched against the protein sequences in the database. Each of the *p*-values for all the

sequences is then normalized, multiplied, and combined using the QFAST algorithm (Bailey and Gribskov, 1998a) in order to arrive at a final statistically significance score.

The p-value represents the probability of finding a match as good as the observed match in a random spot of a random sequence (Bailey and Gribskov, 1998a). QFAST is an algorithm implemented as a part of DASP to combine the p-values of individual fragments to generate a p-value for the alignment of the complete signatures.

Algorithm step 8: Results e-mailed to user

6. The results of the search are then e-mailed to the user. The user receives the same four e-mails as described for Basic Protocol 1 (and shown in Fig. 8.10.3). Two additional e-mails are also received: `[filename]_{genbank,pdb}search.out` and `[filename]_newsigs.fasta`.

   Screen shots of these e-mails are shown in Figure 8.10.4.

   The `.aln`, `.fasta`, `.cw`, *and* `.dnd` *files are the same as described in Basic Protocol 1 (examples shown in Fig. 8.10.3). The* `[filename]_newsigs.fasta` *file contains the putative functional site signature for all sequences found in the search, in FASTA format. These are the actual fragments identified by the PSSM search. The* `[filename]_{P-value cutoff}_{PDB, GB}_search.out` *file lists all sequences found in the search, including GenBank or PDB identification number with p-value above the user-specified cutoff. All sequences which share 100% sequence identity are listed together in the output, so their score is listed only once. Examples of the e-mails specific to Basic Protocol 2 are shown in Figure 8.10.4, with specific application to the mandelate racemase protein family. The e-mail delivery system is known to work with both the Linux and MS Windows operating systems. E-mail to the Mac OS has not been tested.*

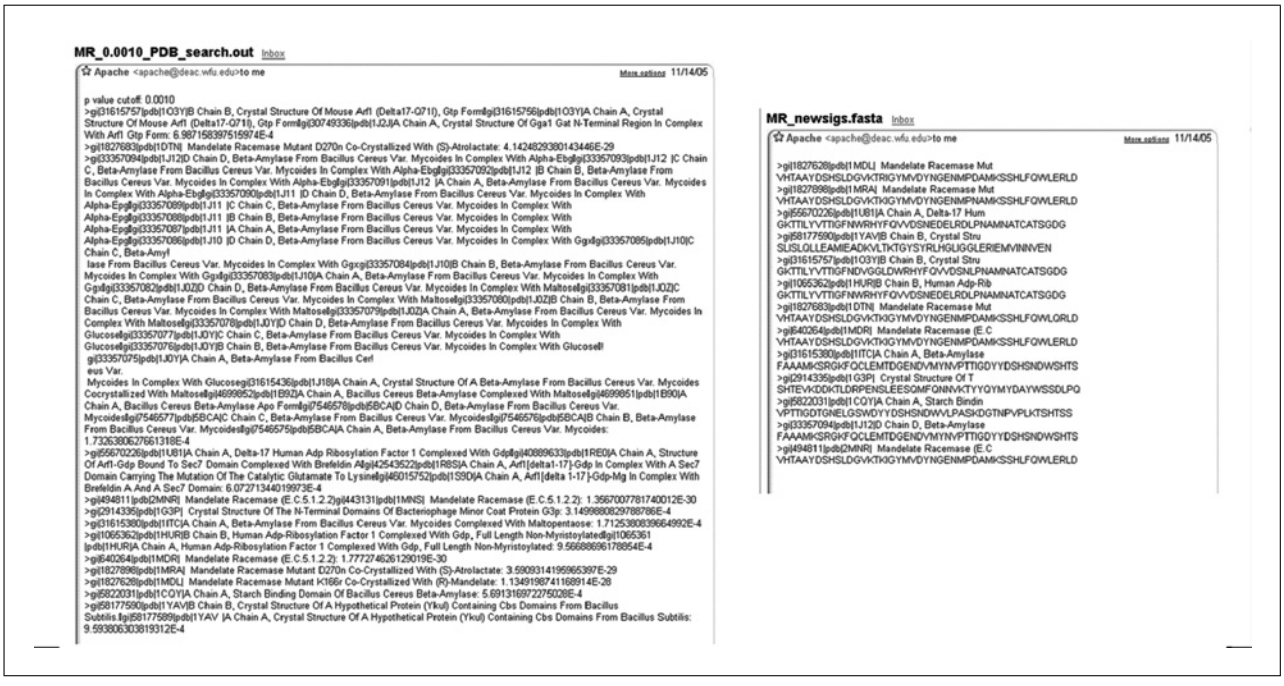


Figure 8.10.4 Examples of files that are e-mailed to the user as a result of Basic Protocol 2 only. The extra files containing the results for applying Basic Protocol 2 to the mandelate racemases protein family are: `MR_0.0010_PDB_search.out` (left) and `MR_newsigs.fasta` (right). Contents of the files are described in the text. The identification of each sequence is indicated with "`>`" at beginning of lines and the *p*-value at the end of the line in the `MR_0.0010_PDB_search_out` file. Note: in GenBank sequence files, all sequences which share 100% sequence identity are listed together in the output, so their score is listed only once, and their names are concatenated.

Completion of searches and mail receipt for this search process can be slow. This is not a job submission service, but instead it runs locally on the user's local machine. Running on the Wake Forest laptops (IBM Thinkpad, Pentium M, 2.2 GHz), the process takes 5 to 10 min for searching the PDB and 4 to 6 hr for searching GenBank. These timings are vastly dependent on the local machine.

User step 3: User analysis of active site profile

7. The user should analyze the results when they are received. Were the correct residues identified as the key residues? Do they align in the active site profile?

   Guidelines for understanding the results are presented in the next section, and ideas for subsequent analysis are presented in the Suggestions for Further Analysis section.

GUIDELINES FOR UNDERSTANDING RESULTS

Basic Protocol 1: Active site profile construction

Basic Protocol 1 describes the process for construction of functional site signatures for each input protein of known structure, alignment of those signatures to create the ASP for those functions, and calculation of a simple score for the alignment (Fig. 8.10.1). This protocol is used when proteins of known structure contain similar active sites and one wants to compare the features of those active sites or cluster the protein family based only on the features around the active site. For example, three protein structures were initially identified as mandelate racemases, and their key residues were identified from the literature (Table 8.10.1; St. Maurice and Bearne, 2004; Siddiqi et al., 2005). The ASP created using the process described in Basic Protocol 1 for these three proteins is shown in Figure 8.10.5A (top profile of three sequences); the ASP score for this profile is 0.86.

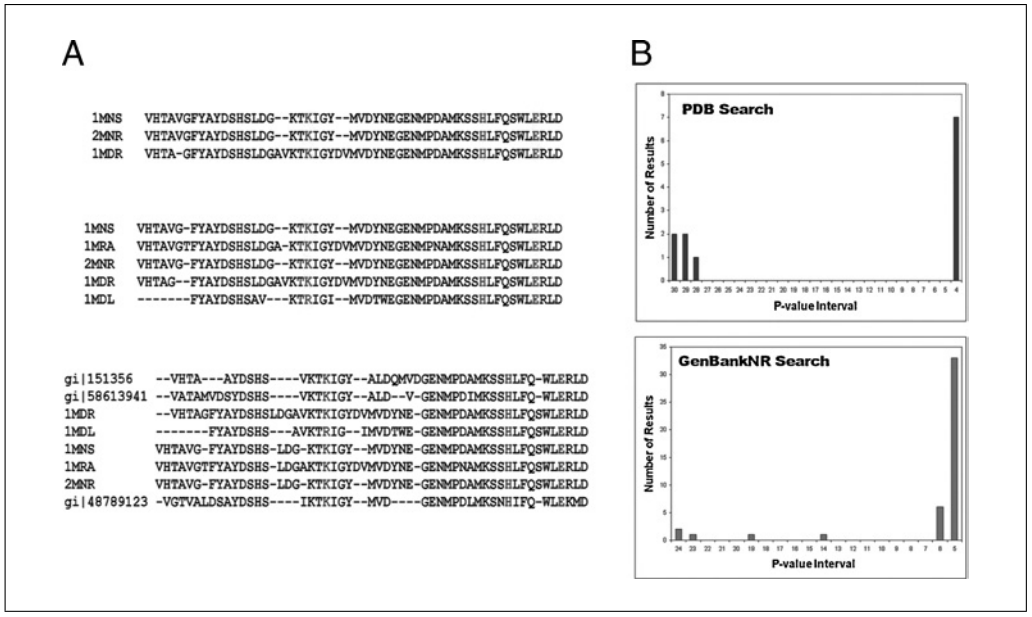


Figure 8.10.5 Example of applying Basic Protocols 1 and 2 to the mandelate racemase protein family. (**A**) Mandelate racemase active site profiles: original set with three PDBs, with ASP score of 0.86 (top); complete profile identified after the bootstrap procedure described in the text, with ASP score of 0.56 (middle); and profile resulting from sequence search of GenBankNR, with ASP score of 0.27 (bottom). The known key residues are identified from structural information and are shown as red letters, while the hypothesized key residues identified from the sequence searches (no known structure) are shown as blue letters. (**B**) Distributions of the *p*-values for the mandelate racemase searches of the PDB sequences (top) and GenBankNR sequences (bottom). The *x*-axis represents the negative of the exponent of the *p*-value (e.g., 30 represents a *p*-value of 10^{-30}). For the color version of this figure go to *http://www.currentprotocols.com*.

The user should critically evaluate these results in several ways. Previous results suggest that a score of 0.25 or better indicates that the proteins are clearly related, at least in the region around this functional site (Cammer et al., 2003). If DASP returns such a score, the user can be fairly certain the functional sites are similar. However, the inverse is not true. If the score is less that 0.25, it does not necessarily mean that the proteins are unrelated at this functional site. Scores of less than 0.25 have been obtained in several cases where (1) the protein family is rather diverse at this functional site, (2) the proteins represent a superfamily or suprafamily (as defined by Babbitt and coworkers; Gerlt and Babbitt, 2001), (3) an error occurred in the input of the proteins and key residue numbers, or (4) the proteins are really unrelated at this site.

The first analysis one should perform is to look at the key functional residues (identified by the user in step 1 and indicated by red letters for the mandelate racemases in Fig. 8.10.5) and determine whether the residues listed in the output are what the user expected. Do they match in all the sequences? Residues that are incorrect could be indicative of errors in entering the residue numbers or of mutation at the functional residue (either natural diversity or specific mutations introduced by experimenters).

Residues that are present but misaligned are indicative of diversity within the families or the possible existence of subfamilies within a larger family. An example is observed within the large glutaredoxin/thioredoxin superfamily (Fig. 8.10.6). The three key residues for this superfamily are two cysteines in a CXXC motif along with a proline that is close in three-dimensional space, but not close in linear sequence to the CXXC motif. These three residues exhibit a common spatial relationship (Fetrow and Skolnick, 1998). In the ASP generated for several members of this superfamily, the subfamilies are easy to identify by eye (as in Fig. 8.10.6), but hierarchical clustering can quantitatively identify them. Members of each cluster can be entered into DASP as a group to obtain the ASP and score for each subfamily. One would expect the ASP scores for the individual subfamilies to be higher than the ASP score for the overall superfamily because the members of a subfamily should be more closely related. Clustering might also identify one member that is an outlier; in this case, the outlying functional site signature should be removed from the profile. Such a result would indicate that this functional site is not as closely related as the other signatures in the profile.

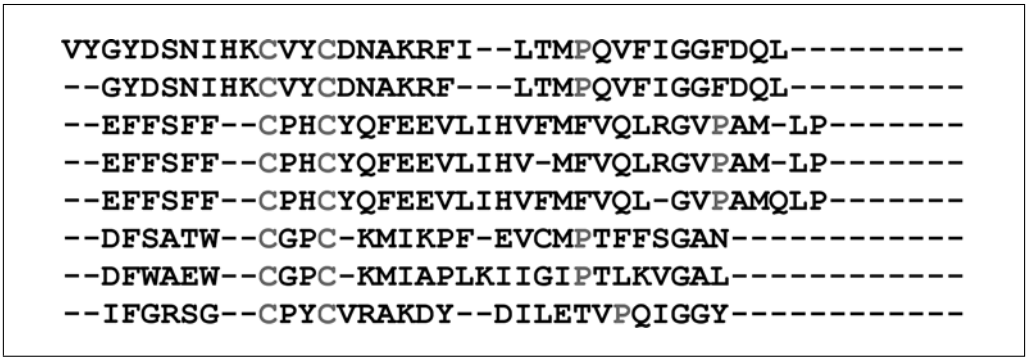


Figure 8.10.6 Part of the glutaredoxin/thioredoxin superfamily active site profile showing different subfamilies within this large superfamily. From the top, the functional site signatures are as follows (listed as pdb filename, protein name): 1aaz, T4 glutaredoxin; 1aba, T4 glutaredoxin; 1ac1, DsbA (disulphide bond forming protein); 1acv, DsbA; 1dsb, DsbA; 1auc, thioredoxin; 2trx, thioredoxin; 1ego, glutaredoxin. The four subfamilies that are visible by eye (from the overall sequence similarity and the alignment of the key residue proline, shown in red) correlate with the biologically relevant subfamilies in this superfamily. For the color version of this figure go to *http://www.currentprotocols.com.*

There is one case where the functional site signatures will not line up in the ASP, even though the proteins are members of a functional family. In protein families where there is significant structural diversity at the functional site, such as hinge-loop motion or domain motion, the functional site signatures will not align well, and it will appear that there are two subfamilies in the family. This is caused by using protein structures to identify the signatures and specifying a selection radius of a particular size. A large motion, such as a hinge-loop motion, results in structures with the loop open or closed, thus incorporating or removing it from the functional site signature. In such cases, one keeps both distinct profiles to represent the structural diversity within the family.

Basic Protocol 2: Searching the sequence database

Basic Protocol 2 utilizes an ASP produced from the implementation of Basic Protocol 1 to search protein sequences for proteins with related functional sites; the search can include protein sequences for which structures have not necessarily been solved. In contrast to Basic Protocol 1, where the user wishes to compare active site features in proteins of known structure, Basic Protocol 2 allows the user to identify potentially related proteins from amino acid sequences alone. This protocol can be applied in at least two ways: (1) to bootstrap the creation of a complete ASP for a family of proteins and (2) to search the sequence databases for proteins that might be related.

In the bootstrap application, one only needs to identify two family members with known structures, build the profile for these two members, and use that profile to search sequences in the PDB database. If new, true-positive sequences are identified from the PDB files, they can be added to the profile by the user through the DASP Web site, and the new profile can be applied to rescreening the sequences of the PDB proteins. The new profile is complete in the sense that it contains all members of the family that are known and represented in the structure database and can be constructed by bootstrapping from only a few known members. If the complete profile has a good score (generally greater than 0.25), the functional sites are similar.

This process is outlined for the mandelate racemases in Figure 8.10.5. Initially, only three mandelate racemase structures were identified, and the initial ASP was built for those three structures (Fig. 8.10.5A, top). A PSSM for this profile was created and that PSSM was used to screen the PDB sequences. Three additional members of the mandelate racemase protein family were identified; however, one of these proteins contained a mutation at one of the key residues. The mutant protein was eliminated from the final, complete ASP (Fig. 8.10.5A, middle; ASP score of 0.56) to avoid contamination of the mandelate racemase profile. If this signature were included in the final ASP, the mutated key residue would affect the PSSM at this position, thus perhaps skewing the database search results. Rescreening the PDB with this complete profile did not result in the identification of any additional members of the family (data not shown). It is useful to analyze the results by plotting a distribution of the *p*-values that are reported with these searches. A typical result is a bimodal distribution for the scores (Fig. 8.10.5B, top). The small group to the left with highly significant *p*-values represents the true positives, and a larger group with less significant scores (typically greater than 10^{-5}) represents nonmatches.

In the second application of Basic Protocol 2, one might like to identify proteins from the sequence database with related functional sites. Such proteins often (but not always) correlate with proteins identified by BLAST searches (see Suggestions for Further Analysis). The mandelate racemase complete ASP (Fig. 8.10.5A, middle) was used to search GenBank sequences, and the resulting ASP is shown in Figure 8.10.5A (bottom). Minimally, this search should identify the sequences from the PDB structures that were used to create the profile. Most often, it will also identify related

proteins (with higher *p*-values) and unrelated proteins (with lower *p*-values). Using the mandelate racemase ASP to search GenBank identified three additional sequences: mandelate racemase found in several Pseudomonads (gi|151356|gb|AAC15504.1|), MdlA from *Pseudomonas fluorescens* (gi|58613941|gb|AAW79574.1|), and L-alanine-DL-glutamate epimerase and related enzymes of enolase superfamily [*Burkholderia fungorum* LB400] (gi|48789123|ref|ZP_00285102.1|). These hits exhibited significant *p*-values of 10^{-23}, 10^{-19}, and 10^{-14}, respectively, indicating that all have functional sites related to the proteins in the original ASP. A plot of the distribution of the *p*-values for this sparsely populated family exhibits bimodal distribution (Fig. 8.10.5B, bottom). The limits of this searching tool have not been completely explored, but it has been observed that for more diverse families the separation between *p*-values of related and unrelated proteins is not as distinct (J.S. Fetrow, unpub. observ.).

COMMENTARY

Background Information

Most often, large-scale function assignment relies on automated annotation transfer from the most similar or related protein sequence to uncharacterized proteins. It has been observed that annotation transfer based on protein sequence comparison often fails at sequence similarity levels below 25% to 30% identity, which has lead to significant misannotations in the sequence databases (Hegyi and Gerstein, 2001; Rost, 2002; Baxter et al., 2004).

To address limitations of function annotation transfer, sequence motifs such as PRINTS (Attwood, 1998), BLOCKS (Henikoff et al., 1999), and Prosite (Hofmann et al., 1999) have become useful tools for annotating complete genomes and large-scale proteomic sequence sets. To facilitate structure-based annotation of computationally derived protein models, the author had previously developed the Fuzzy Functional Form (FFF) technology (Fetrow and Skolnick, 1998). An FFF is created by first identifying key residues, as described in the Critical Parameters section. Each FFF is then validated using families of protein structures and experimental information found in the literature. Creation of an FFF does not depend on multiple sequence alignments or sequence pattern identification. Thus a precise determination of protein function, based on key active site residues and their geometric arrangement in space, can be made. In addition, this functional site-centered approach can make unambiguous assignment of multiple functional sites in a single polypeptide structure.

The use of a small number of key residues is both an advantage and a limitation of the FFF method. The small number of residues allows these structural descriptors to identify large families of proteins (e.g., serine hydrolases; Baxter et al., 2004), but limits their value for identifying function specificity determinants such as those residues important for defining substrate specificity in enzymes. Determination of the features that are essential to the specificity of protein function is becoming increasingly important to the process of protein function annotation.

To overcome the limitations of the FFF method, the author developed the functional site profiling method (Cammer et al., 2003). This method was implemented in the DASP Web site described in this unit. The ASP encodes key similarities and distinguishing features of the active sites within a protein family. This structural profile-based function assignment method provides subfamily classification and physicochemical information relevant to identifying specificity determinants at functional sites. An extension of the method allows searching the sequence databases based on the information in the ASPs (Huff, 2005; Huff et al., 2005), and a Web site implementing this method is described in Basic Protocol 2. The approach is used to identify sequences which contain protein fragments related to those in the ASP. Analysis of the signatures in these larger profiles can be useful in identifying the specificity determinants for inhibitors in structure-based drug discovery methods (Huff et al., 2005).

Critical Parameters

Key residues (Basic Protocol 1)

One important parameter is the initial choice of key residues. In the author's focus on enzyme active sites, amino acids that are crucial for the chemistry of the mechanism and have structures and positions that are conserved in the proteins of interest are usually identified. For example, in a typical serine protease, the author would choose the conserved

serine, histamine, and aspartic acid of the catalytic triad. In the mandelate racemases, histidine, lysine, and glutamic acid (Table 8.10.1) were chosen because experimental evidence indicates that these residues play key acid and base catalyst roles in the racemase enzymatic reaction (St. Maurice and Bearne, 2004; Siddiqi et al., 2005). Choice of these residues is nonempirical and is based on the user's interpretation of available experimental data, literature, and structure and sequence comparison of family members. Changing the residues can change the signature somewhat, but it does not usually have a big impact on the results, unless a chosen residue is at the edge or outside of the functional site that is being studied. It is easy to mistype the key residues into the DASP Web site. Use the PDB-based numbering for the residues and always check to be sure that the expected residue names are returned.

Search radius (Basic Protocol 1)

A second important parameter is the radius of the spheres that are used to identify the functional site signature. The functional site signature is identified as all residues with at least one atom within the search radius of the center of mass of one of the key residues. In initial studies, 10 Å was shown to be enough to encompass all or most of the functional site, without including too much nonspecific structure (Cammer et al., 2003); thus, the default is set to 10 Å. However, this is a user-settable parameter because functional sites can be different sizes. For instance, if an enzyme active site binds a long substrate, identification of the substrate-binding site signatures might require use of a larger search radius. Similarly, small functional sites might call for using a smaller search radius. The user can vary the search radius to see what impact this parameter has on the results.

Cutoff score (Basic Protocol 2)

A third important parameter is the cutoff score that is used for searching the sequence databases. All *p*-values smaller than (more significant than) this cutoff are reported to the user, while larger *p*-values are not reported. The purpose of this cutoff score is to limit the amount of information returned to the user. As mentioned above, large output files can impact the amount of time required for output file creation and/or can be stripped by e-mail servers that limit the attached file size. For the families that have been studied so far (mandelate racemases, cyclooxygenases, and a few others), a *p*-value of 10^{-4} or 10^{-3} has been a good balance between being certain to identify all true positives and not getting too many false hits.

Troubleshooting

Potential problems have been described throughout this article. These problems are summarized below:

1. DASP does not run. This happens most often because the user does not have the Java 1.5 plug-in for the Web browser. The author has experienced some problems with the Internet browser Netscape, as well.

2. ASP does not look right, or the key residues are not identified properly. Check PDB numbers of the key residues. The PDB numbers are specific to each PDB file.

3. No e-mail is returned for the PDB search within 1 to 2 hr or for the GenBankNR search in 24 hr. It is likely that the output file is too big. Try a more stringent *p*-value, to get a smaller result file size. In addition, the local computer might be too slow to run the GenBank search in a reasonable time.

4. Cannot find the PDB file in the GenBankNR output. The input PDB files should appear in the GenBankNR search results, output sequence list, FASTA file list, and/or final ASP. The GenBank results group identical sequences together, so that only the first one in the list gets extracted for further analysis. Check the GenBank sequence numbers to see if it is identical to another sequence that is listed in the output.

5. DASP cannot find the PDB file that the user wishes to enter. This occurs when the PDB file is new and is not in the files that DASP searches. PDB files are updated annually on the DASP server, so the file listings might be up to a year behind. In a future release, the ability to include user-input PDBs for searching will be added.

Currently, the output files do not contain a profile for the GenBankNR search, only the FASTA-formatted sequences for the functional site signatures. The profiles shown in Figure 8.10.5 were created using ClustalW to align the signatures from the alignment files and visually identify the alignment of the key residues. Future development of the DASP Web site may implement automated alignment.

A limitation of the current implementation is that it does not handle functional sites with key residues that are located on different protein chains. If a functional site is not identified or a sequence which is known to have this functional site is pulled up at a very low score, this might be the issue.

Suggestions for Further Analysis

Basic Protocols 1 and 2

For both protocols, the ultimate result is a multiple sequence alignment of the functional site signatures. This alignment can be analyzed using any tools that are typically utilized to analyze multiple sequence alignments, e.g., ClustalW (*UNIT 2.3*), PileUp (*UNIT 3.6*), and T-Coffee (*UNIT 3.8*). In the author's research on enzyme active sites, hierarchical clustering of the signatures in the ASP is an important first step. Hierarchical clustering groups the residues based on either sequence identity or the ASP score. Because the signatures were built from only the protein fragments around the active site, the clusters begin to identify differences between the members of the family around those active sites. For example, hierarchical clustering would identify the subfamilies that are visible in the sample of the glutaredoxin/thioredoxin superfamily (Fig. 8.10.6). Those differences can lead to a subfamily clustering based on what might be called specificity determinants of the functional sites, possibly indicating differences in substrate specificity or details of enzyme mechanism that might not be obvious from either analysis of the complete profile or clustering of the full sequence alignment. Investigation of such specificity determinants in the cyclooxygenases (Huff et al., 2005) and other protein families have been initiated.

Basic Protocol 2

It is useful in analyzing database search results (particularly GenBank) to compare the results of the DASP search to results using BLAST (see *UNIT 3.4*) as the search tool (Altschul et al., 1997). In the author's experience, high-scoring matches to the ASP from DASP should also be found by a BLAST search using one of the template sequences as the query. High-scoring sequences found by a BLAST search should also score high in the DASP search. Differences between BLAST and ASP scores are found in the twilight zone of sequence similarity. Sequences that are highly similar in the fragments around the functional site might score very well using the DASP search, but score very poorly in a BLAST search.

Literature Cited

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Shang, Z., Miller, W., and Lipman, D.J. 1997. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucl. Acids Res.* 25:3389-3402.

Attwood, T.K., Beck, M.E., Flower, D.R., Scordis, P., and Selly, J. 1998. The PRINTS protein fingerprints database in its fifth year. *Nucl. Acids Res.* 26:304-308.

Bailey, T.L. and Gribskov, M. 1998a. Combining evidence using p-values: Application to sequence homology searches. *Bioinformatics* 14:48-54.

Bailey, T.L. and Gribskov, M. 1998b. Methods and statistics for combining motif match scores. *J. Comput. Biol.* 5:211-221.

Baxter, S.M., Rosenblum, J.S., Knutson, S., Nelson, M.R., Montimurro, J.S., Di Gennaro, J.A., Speir, J.A., Burbaum, J.J., and Fetrow, J.S. 2004. Synergistic computational and experimental proteomics approaches for more accurate detection of active serine hydrolases in yeast. *Mol. Cell. Proteomics* 3:209-225.

Cammer, S.A., Hoffman, B.T., Speir, J.A., Canady, M.A., Nelson, M.R., Knutson, S., Gallina, M., Baxter, S.M., and Fetrow, J.S. 2003. Structure-based active site profiles for genome analysis and functional family subclassification. *J. Mol. Biol.* 334:387-401.

Fetrow, J.S. and Skolnick, J. 1998. Method for prediction of protein function from sequence using the sequence-to-structure-to-function paradigm with application to glutaredoxins/thioredoxins and T1 ribonucleases. *J. Mol. Biol.* 281:949-968.

Fetrow, J.S., Godzik, A., and Skolnick, J. 1998. Functional analysis of the Escherichia coli genome using the sequence-to-structure-to-function paradigm: Identification of proteins exhibiting the glutaredoxin/thioredoxin disulfide oxidoreductase activity. *J. Mol. Biol.* 282:703-711.

Gerlt, J.A. and Babbitt, P.C. 2001. Divergent evolution of enzymatic function: Mechanistically diverse superfamilies and functionally distinct suprafamilies. *Annu. Rev. Biochem.* 70:209-246.

Gribskov, M., McLachlan, A.D., and Eisenberg, D. 1987. Profile analysis: Detection of distantly related proteins. *Proc. Natl. Acad. Sci. U.S.A.* 84:4355-4358.

Hegyi, H. and Gerstein, M. 2001. Annotation transfer for genomics: Measuring functional divergence in multi-domain proteins. *Genome Res.* 11:1632-1640.

Henikoff, S., Henikoff, J.G., and Pietrokovski, S. 1999. Blocks+: A non-redundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics* 15:471-479.

Higgins, D.G., Thompson, J.D., and Gibson, T.J. 1996. Using CLUSTAL for multiple sequence alignments. *Methods Enzymol.* 266:383-402.

Hofmann, K., Bucher, P., Falquet, L., and Bairoch, A. 1999. The Prosite database, its status in 1999. *Nucl. Acids Res.* 27:215-219.

Huff, R.G. 2005. DASP. Active Site Profiling for Identification of Functional Sites in Protein Sequences and Structures. Thesis, Wake Forest University, Winston-Salem, N.C.

Huff, R.G., Bayram, E., Tan, H., Knutson, S.T., Knaggs, M.H., Richon, A.B., Santago, P., II, and Fetrow, J.S. 2005. Chemical and structural diversity in cyclooxygenase protein active sites. *Chem. and Biodiversity* 2:1533-1552.

Rost, B. 2002. Enzyme function less conserved than anticipated. *J. Mol. Biol.* 318:595-608.

Siddiqi, F., Bourque, J.R., Jiang, H., Gardner, M., St. Maurice, M., Blouin, C., and Bearne, S.L. 2005. Perturbing the hydrophobic pocket of mandelate racemase to probe phenyl motion during catalysis. *Biochemistry* 44:9013-9021.

St. Maurice, M. and Bearne, S.L. 2004. Hydrophobic nature of the active site of mandelate racemase. *Biochemistry* 43:2524-2532.

Key References

Cammer et al., 2003. See above.

Describes original research leading to the development of the active site profiling method. Details are given about scoring and validation.

Baxter et al., 2004. See above.

Describes a computational method for profiling sequences with an experimental proteomics method. Detailed analysis of serine hydrolases in yeast is presented.

Internet Resources

http://dasp.deac.wfu.edu

This DASP Web site allows access to the active site profiling software.

Contributed by Jacquelyn S. Fetrow
Wake Forest University
Winston-Salem, North Carolina

Structure-Based pK_a Calculations Using Continuum Electrostatics Methods

The biological function of many proteins is governed by electrostatics; therefore, electrostatic free energy and pK_a values can be useful to correlate structure with function. Structure-based calculations are necessary to bridge the gap between structure and function when pK_a values and electrostatic energies cannot be measured experimentally, or when it is of interest to elucidate the physical and structural determinants of these energies. This unit describes protocols for calculation of pK_a values in proteins with finite difference Poisson-Boltzmann (FDPB) methods that have been calibrated extensively against experimental data, and which can contribute significant insight into the properties of surface charges in proteins.

The literature on the application of FDPB methods for pK_a calculations is extensive. Many different implementations of FDPB methods are available that differ mainly in the manner in which the polarizability of the protein is treated. The protocols described in this unit are for pK_a calculations using the UHBD (University of Houston Brownian Dynamics) software developed by McCammon and colleagues (Davis et al., 1991; Madura et al., 1995). These are among the easiest FDPB methods to use for pK_a calculations.

CALCULATING pK_a VALUES USING THE FDPB METHOD AND THE SINGLE-SITE CHARGE MODEL (FDPB/SS)

BASIC PROTOCOL

pK_a values can be calculated with the FDPB method with several different protocols. The simplest one is the single-site method, FDPB/SS (Fig. 8.11.1). This method employs a static protein structure. The ionization processes are modeled by the addition of a single unit charge at a specified titratable atom (Antosiewicz et al., 1994). The electrostatic potential due to the unit charge is calculated with the FDPB algorithm. The FDPB/SS calculation requires two separate finite difference calculations for each ionizable residue, one for the residue in the protein environment, and a second one for the residue when it is in the aqueous environment. The detailed steps that need to be followed to perform these calculations are software-dependent. The following procedure delineates the sequence of steps and the files and parameters (referenced in parentheses) that must be specified either as defaults or by the user in FDPB/SS calculations with the UHBD software package (see Table 8.11.1; Madura et al., 1995). All the examples described in this section are for calculations with the UHBD code. For detailed information about the UHBD package, consult the web site listed in Table 8.11.1, or contact the developers directly. Note that the procedures outlined below are based on the authors' experience as users of UHBD.

Necessary Resources

Hardware

Computer capable of running Windows, Unix, or a Macintosh operating system

Software

Software for FDPB calculations is available for SGI, Linux, AIX, Windows and Mac configurations. Not all configurations are supported by all available software (see Table 8.11.1).

Contributed by Carolyn A. Fitch and Bertrand García-Moreno E.

Current Protocols in Bioinformatics (2006) 8.11.1-8.11.22

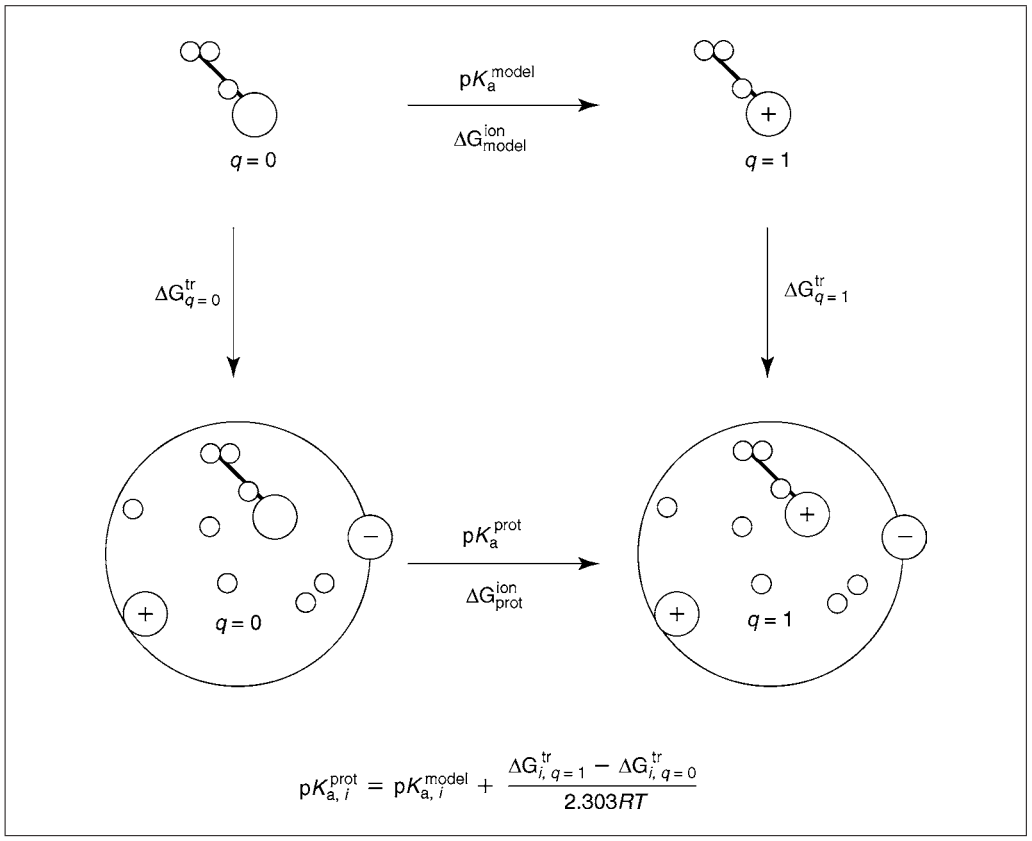


Figure 8.11.1 Thermodynamic cycle for pK_a calculations. Thermodynamic cycle used in the FDPB/SS method for pK_a calculations. pK_a^{model} represents the pK_a of an ionizable group in a model compound. pK_a^{prot} is the pK_a of the group in the protein. The transfer free energies, ΔG_i^{tr}, are the calculated electrostatic free energy changes for transferring the ionizable group from water to the protein environment in the neutral ($q = 0$) and ionized ($q = 1$) states. Larger circles denote ionizable groups in the protein; smaller circles denote the polar atoms of the protein, which are treated in these methods in terms of partial charges.

Software (Table 8.11.1) is needed both for calculation of electrostatic potentials by solving the linearized PB equation with a FDPB solver, and also for the calculation of pK_a values starting from the calculated electrostatic potentials. Executables for some of the software listed in Table 8.11.1 are available for downloading. A compiler (C, C++, or Fortran) may be needed to execute other packages. Additionally, a plotting package and molecular visualization software is useful. Web servers have become available recently that will perform pK_a calculations for user-specified structures (see Table 8.11.1).

Files

A three-dimensional molecular structure of the protein. Typically this is a PDB-formatted file obtained from X-ray crystallography, NMR spectroscopy, or a structure produced with a modeling program.

A parameter file containing atomic partial charges and radii. The format of this file will be specific to the software being used, and is usually supplied with the software package.

The software packages usually supply all other necessary files and scripts. Users can and should explore how different charges, radii, and input parameters affect the calculated pK_a values and energies. It is important to emphasize that most packages for FDPB calculations allow the user to modify existing protocols by altering input parameters. This will require the ability to modify or write Unix scripts to control the flow of the calculations.

Table 8.11.1 Downloadable Software for pK_a Calculations with FDPB Methods

Software	Platform	URL
DelPhi	SGI, Linux, PC, AIX, Mac	*http://trantor.bioc.columbia.edu/delphi/*
MEAD (Macroscopic Electrostatics with Atomic Detail)	Unix, Windows	*http://www.scripps.edu/mb/bashford/*
PEP (Paul's electrostatics programs)	Unix	*http://www.scripps.edu/mb/case/*
UHBD (University of Houston Brownian Dynamics)	Unix	*http://mccammon.ucsd.edu/uhbd.html*
ZAP	Unix, Irix, Windows, Mac	*http://www.eyesopen.com/www.eslc.vabiotech.com/*
APBS (Adaptive Poisson Boltzman Solver)	Unix, Windows web-server	*http://apbs.sourceforge.net/*
H^{++} webserver[a,b]	Web based	*http://biophysics.cs.vt.edu/H++/*
HYBRID[c]	Unix	*http://gilsonlab.umbi.umd.edu/index.html*
KARLSBERG[a]	Linux	*http://agknapp.chemie.fu-berlin.de/agknapp/*
MCCE[d] (Multi-Conformation Continuum Electrostatics)	Unix, Mac	*http://www.sci.ccny.cuny.edu/~mcce/*
PCE webserver[a] (Protein Continuum Electrostatics)	Web based	*http://bioserv.rpbs.jussieu.fr/Help/PCE.html*
WHATIF-pK script[d,e]	SGI, Lunix, Windows	*http://swift.cmbi.kun.nl/whatif/*
Rebecca Wade lab scripts[e]	Unix	*http://projects.villa-bosch.de/mcm/software/pka*

[a]Uses MEAD solver.
[b]Uses HYBRID pK_a calculation.
[c]Also packaged with UHBD.
[d]Uses DelPhi solver.
[e]Uses UHBD solver.

Preparation of the input molecular structure

The steps outlined below are specific for the UHBD software. The same general steps would have to be performed for pK_a calculations with other FDPB solvers.

1. Add hydrogen atoms to the structure in the neutral state (`pkaS-addH` is a UHBD specific script used to run CHARMm; `pkaS-hbuild.inp` is the CHARMm input file).

   There are a variety of ways to do this. Each program will have its own syntax and idiosyncrasies. In UHBD, this step utilizes the CHARMm (Brooks et al., 1983) `HBUILD` *command to add polar hydrogen atoms, followed by a minimization step applied to the added H atoms. In this simulation the structure must be in the neutral state. Note that, depending on the protocol being used, the addition of hydrogen atoms to a structure can be a critical step that needs to be explored in detail. This is particularly important in cases of hydrogen bonded groups, where the position of hydrogen atoms might affect the results.*

Calculation of the molecular electrostatic potential using an FDPB solver

2. Define the set of atomic parameters to be used (`pkaS.dat`).

   All FDPB solvers require the specification of atomic radii and atomic charges. In the FDPB/SS method the atomic radii are from the OPLS parameter set (Jorgensen and Tirado-Rives, 1988) and the atomic partial charges are from the CHARMm polar-only hydrogen set (Brooks et al., 1983). In calculations with the FDPB/SS method partial charges are only needed for the residues in their neutral state.

3. Define input parameters (`pkaS-doinp.inp`).

   The input parameters defined by the user are: (a) the protein dielectric constant, ε_{in}, (b) the solvent dielectric constant, ε_{H_2O}, (c) grid size and spacing, (d) temperature, and (e) ionic strength (see Table 8.11.2). The most important of these parameters (i.e., the ones that influence the value of the calculated electrostatic potentials most significantly) are the protein dielectric constant and the grid specifications, discussed below (see Fig. 8.11.2) The maximum number of iterations refers to the maximum iterations used for the FDPB solver. The dielectric boundary is defined by a probe-accessible surface using a probe radius of 1.4 Å and 500 points per atom sphere (Richards, 1977; Gilson et al., 1988). Use of a probe radius of 0.0 Å shifts the dielectric boundary to the van der Waal's surface, as discussed ahead (Zhou and Vijayakumar 1997).

4. Define tautomeric state of residues (within `pkaS-doinp.inp` file).

   The desired tautomeric state of the ionizable groups are usually specified in input files. In some packages this is handled automatically, for example, through the use of multiple conformations. The definition of tautomeric states should be explored in depth in cases of ionizable groups in networks of polar and ionizable groups.

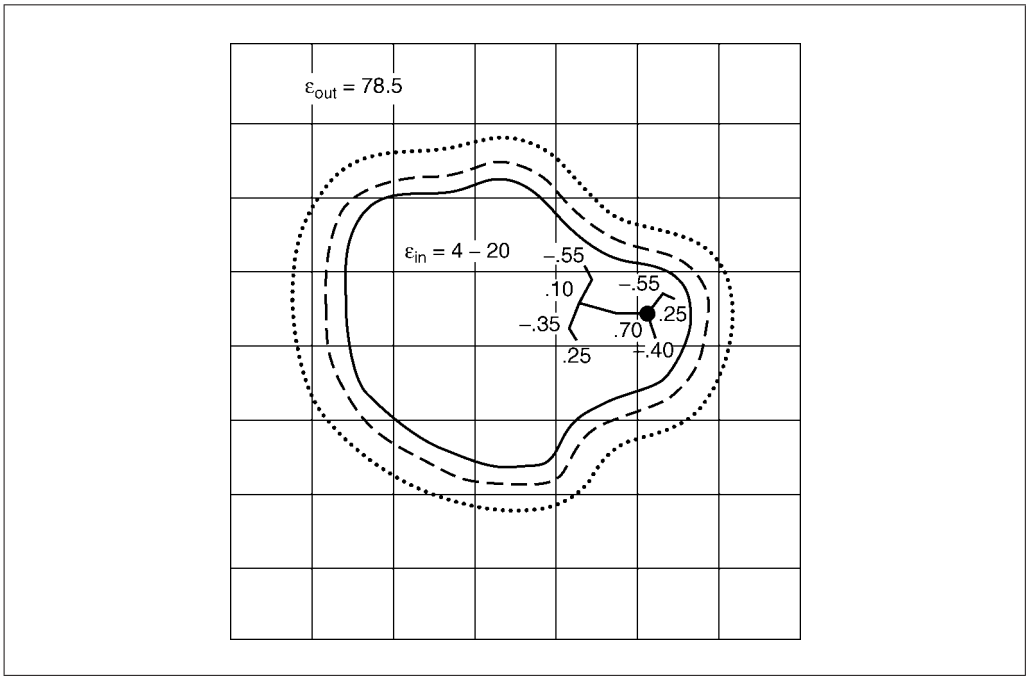


Figure 8.11.2 Model of the protein-water system used for calculation of electrostatic potentials with FDPB methods. The solid line represents the van der Waal's envelope of the protein. The dashed line describes the water-accessible surface that constitutes the boundary between the water phase with high dielectric constant and the protein phase with low dielectric constant ε_{in}. The dotted line represents the ion exclusion surface. A single Asp side chain is represented, with partial charges given for the atoms of the group. The grid is necessary for the solution of the Poisson-Boltzmann equation by the method of finite differences.

Table 8.11.2 Input Parameters to FDPB/SS Method Using the UHBD Package

FDPB Input Parameters		
Number of grid sets to use	4	
Grid specifications	***Spacing (Å)***	***Grid dimensions***
Coarse grid	1.5	65^3
Focused grid 1	1.2	15^3
Focused grid 2	0.75	15^3
Focused grid 3	0.25	20^3
Maximum number of iterations	300	
Temperature (K)	298	
Dielectric constant of protein (ε_{in})	20	
Dielectric constant of solvent (ε_{out})	78.5	
Ionic strength (mM)	100	
Radius of ion probe (Å)	2.0	
Probe radius for protein/solvent dielectric boundary (Å)	1.4	
Number of points for atom surface	500	
Model pK_a values		
c-term	3.8	
Asp	4.0	
Glu	4.4	
His	6.3	
n-term	7.5	
Tyr	9.6	
Lys	10.4	
Arg	12.0	

5. Run the FDPB solver (`pkaS-dosbs` script).

   In pK_a calculations with the FDPB/SS method the solver is called twice for each residue (once for the residue in the protein, and again for the residue in water for each grid specified). To model a residue in water the coordinates of the residue are extracted from the protein structure. Boundary conditions for the calculation of potentials are set by default to the sum of potentials of the individual atoms treated as Debye-Hückel spheres.

   The output from this step is the "`potentials`" file. This file contains the self-energy and the Coulomb potential for each ionizable group—i.e., the energy arising from interactions with all other sites assuming that these sites are fully charged; this and other energy terms used to calculate pK_a values with FDPB methods are described in more detail in Background Information.

The steps that are performed automatically in the UHBD script `pkaS-dosbs` are:

a. Files are prepared for UHBD input
b. Grids are calculated
c. The PB equation is solved by looping over all ionizable groups

The following calculations are performed:

i. Calculation of Born energies. This involves calculation of the difference in the solvation energy of a group in water and in the protein.

ii. Calculation of background energies. This involves calculation of the Coulomb interaction energy of a group with all background charges in the protein. Note that these interaction energies are between an ionizable group and permanent dipoles. It is not to be confused with the Coulomb energy between ionizable groups.

iii. Calculation of Coulomb interaction potential of each titratable site, owing to their interactions with all other ionizable groups.

Calculation of the pH-dependent ionization state of a protein

The energy of the interactions between ionizable groups is dependent on the state of ionization of each group; therefore, this energy is pH dependent. Because each ionizable group can exist in charged or neutral forms, a protein with N ionizable groups can have 2^N possible charge states. Ideally, to titrate a protein in silico, the energy of each state should be considered explicitly. In practice, this cannot be done for proteins with a large number of ionizable residues. This problem can be handled with statistical mechanical approximations or with a Monte-Carlo treatment. A number of different algorithms are available to titrate proteins in silico (see Table 8.11.1). In UHBD, a Monte-Carlo routine (dopss) and a cluster routine (hybrid) are called by default within the `pkaS-dosbs` script. The input to these programs is the `potentials` file created in step 5.

6. Run an in silico pH titration (hybrid).

   In the application described in this section, all electrostatic energies are referenced to the fully neutral state of the protein, as described by Gilson (1993). Each titratable site is allowed to be either fully charged or fully neutral. The computation begins with the fully ionized state and iterates until the fractional occupancies converge. The user must specify the pH range and the size of the pH steps to be evaluated. Default parameters of cluster size 10 and fixed ionization cutoff of 0.05 are used in the hybrid code.

   The output file (`hybrid.out`*) contains three sets of data:*

   a. The average charge and free energy of the protein at each pH value of interest.

   b. The calculated pK_a values for all ionizable groups.

   c. The average charge of each group at each pH.

Table 8.11.3 Examples of pK_a Values Calculated with FDPB Method with the `1stn.pdb` Structure[a]

Residue	Type	Atom	pK_a(model)	pK_a(app)	Δ pK_a(app)
6	Lys	NZ	10.4	10.44	0.04
8	His	ND1	6.3	6.36	0.06
9	Lys	NZ	10.4	11.95	1.55
10	Glu	CD	4.4	2.58	−1.82
16	Lys	NZ	10.4	9.96	−0.44
19	Asp	CG	4.0	2.52	−1.48
21	Asp	CG	4.0	0.52	−3.48
24	Lys	NZ	10.4	10.25	−0.15
27	Tyr	OH	9.6	12.45	2.85
28	Lys	NZ	10.4	11.01	0.61

[a]Parameters used: ε_{in}=20, 100 mM ionic strength.

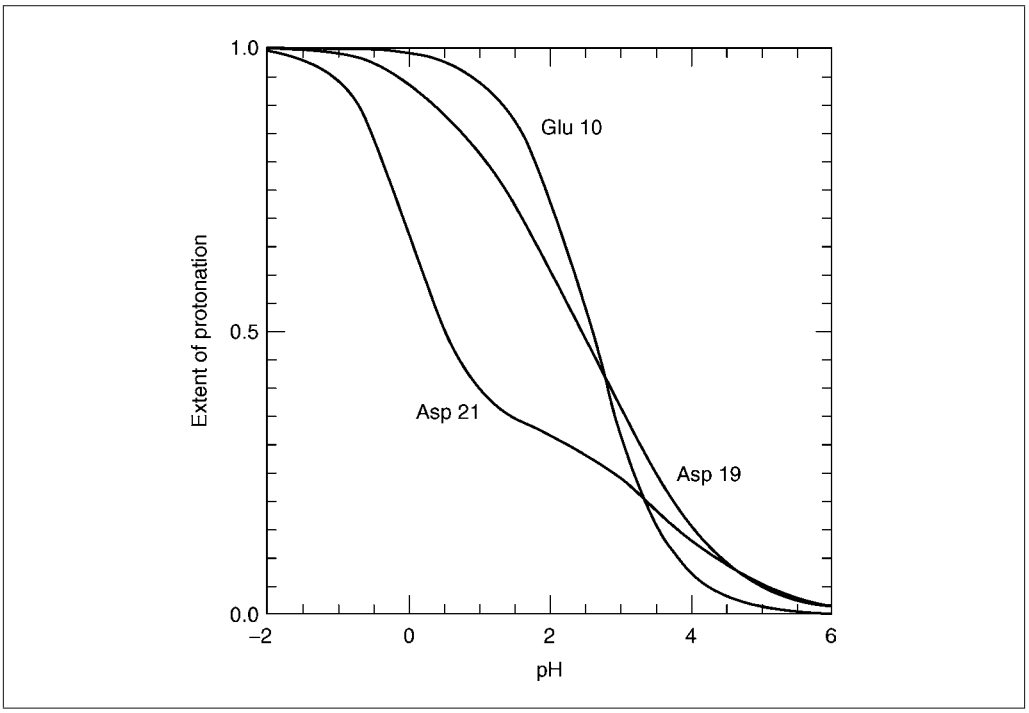


Figure 8.11.3 H^+ titrations of three acidic groups calculated with the FDPB/SS method. The curves were calculated with the `1stn.pdb` structure with FDPB/SS method using $\varepsilon_{in} = 20$ and ionic strength = 100 mM. The pK_a value [listed under pK_a(app) in Table 8.11.3] represents the pH where the extent of protonation is 0.5.

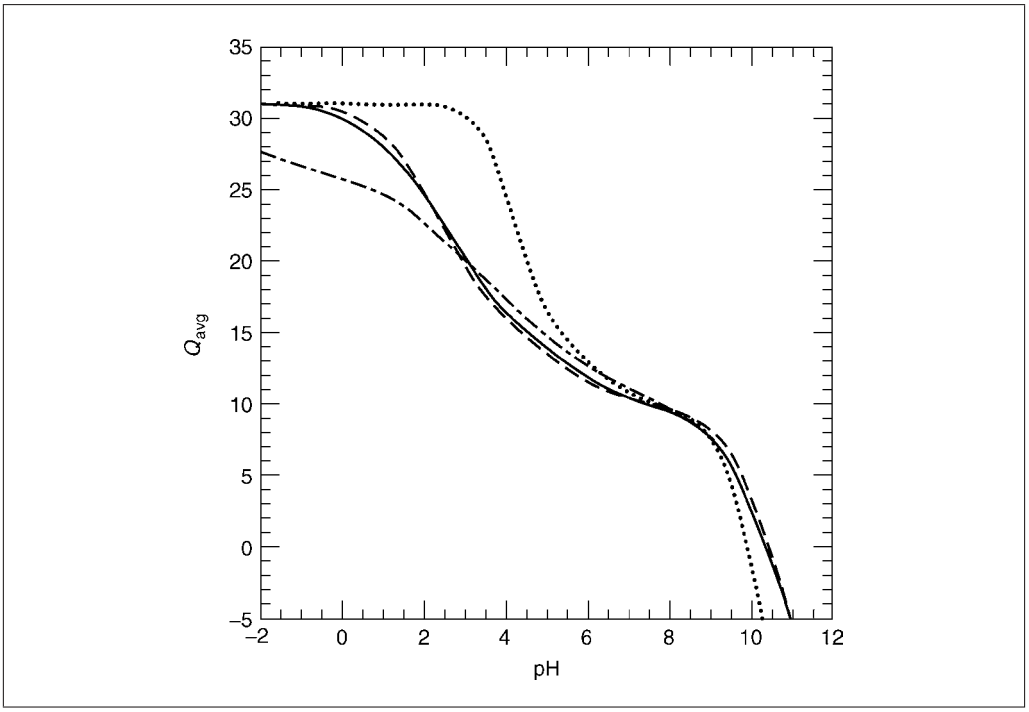


Figure 8.11.4 Overall H^+ titration calculated by FDPB methods. Plot of the average charge (Q) of `1stn.pdb` calculated with FDPB in 100 mM ionic strength: (solid line) FDPB/SS with $\varepsilon_{in} = 20$; (dashed-dot) FDPB/F with $\varepsilon_{in} = 4$; (dashed) FDPB/SS-HH with $\varepsilon_{in} = 20$; (dotted) calculated with the Henderson-Hasselbalch equation using the pK_a values of model compounds.

7. Plot and analyze results.

Table 8.11.3 lists the output pK_a values computed for the first ten residues in staphylococcal nuclease using the structure with accession code `1stn.pdb`. *These values were computed using an internal protein dielectric constant $\varepsilon_{in} = 20$ and ionic strength 100 mM. pK_a(model) describe the input pK_a values obtained from experimental measurements in model compounds. pK_a(app) describes the pH at which the ionizable group is half titrated (i.e., its charge is |0.5|). pK_a(app) are the pK_a values that can be compared with values measured experimentally, for example, by NMR spectroscopy.*

Sample H^+ titration curves of four groups in nuclease are plotted in Figure 8.11.3. Note that in a protein, the shape of the titration curves of individual groups can be quite different from the standard titration curve of an isolated group in water.

The average charge of the protein, Q, is calculated as the sum of the titration curves of all individual groups. An example is shown in Figure 8.11.4. Also plotted in this figure are Q computed by summing the isotherms calculated with the Henderson-Hasselbalch equation using the pK_a(app) values (FDPB/SS-HH), and Q computed using the model compound pK_a values (HH). The titration curve calculated with the pK_a values of model compounds is assumed to represent the titration curve of the denatured state of the protein.

ALTERNATE PROTOCOL

CALCULATING pK_a VALUES USING THE FDPB METHOD AND THE FULL CHARGE MODEL (FDPB/F)

The full charge FDPB method, FDPB/F, offers a more realistic way to model the charge state of an ionizable site (Bashford and Gerwert, 1992; Yang et al., 1993). This method differs from the FBDP/SS single-site method described above in two ways. First, in the FDPB/F method the ionized form of the titratable group is not represented with a single unit charge. Instead, the unit charge is distributed over several atoms of the residue. The exact manner in which charge is distributed in the neutral and ionized state depends on the atomic parameter set used. The sum total of partial charges in each ionizable side chain in the charged state totals ± 1. The second way in which the FDPB/F protocol differs from the FDPB/SS protocol is that four separate FD calculations are needed for each titratable group in the FDPB/F method. The neutral and charged forms of each group need to be considered separately, for the group both in water and in the protein.

The FDPB/F method is particularly useful in cases of ionizable groups in active sites or in networks of polar or charged groups (Trylska et al., 1999). In these cases the outcome of the calculations is highly dependent on the microscopic distribution of charge in the side chain of the ionizable groups. The FDBP/F uses a more realistic charge distribution than the simpler FDPB/SS protocol. When calculations are performed with $\varepsilon_{in} \approx 4$, the FDPB/F method is more accurate than the FDPB/SS method (Antosiewicz et al., 1996b). In general, however, values of $\varepsilon_{in} \approx 20$ give the best agreement between calculated and measured pK_a values. When $\varepsilon_{in} \approx 20$ is used, the FDPB/SS and FDPB/F methods usually give comparable results. For this reason, the calculations with FDPB/SS are preferable for many applications; they are twice as fast as the calculations with the FDPB/F method because they require half the number of FD calculations.

The steps for the FDPB/F protocol are similar to the steps described for the basic FDPB/SS calculation (Antosiewicz et al., 1996a). The differences between the two methods are in the preparation of the input molecular structure and in the actual calling of the FDPB solver. These differences are noted below.

There are two key differences between the FDPB/SS and the FDPB/F calculations: (1) The first difference is in how the input molecular structure is prepared by addition of hydrogen atoms to the protein. In the FDPB/SS calculations, H atoms are added to polar atoms to model the fully neutral form of each ionizable group in the molecule (i.e., the COOH form of carboxylic groups, and the NH2 form of basic groups), whereas in the

FDPB/F calculations, H atoms are added to polar groups to model the fully protonated state of each group (i.e., The COOH form of carboxylic groups, and the NH3+ form of basic groups). (2) The second difference is that in the FDPB/F calculation the user must employ scripts that call four FDPB calculations that use the charged states corresponding to the residue in either the charged or the neutral state.

Necessary Resources

Hardware

Computer capable of running Windows, Unix, or a Macintosh operating system

Software

Software for FDPB calculations is available for SGI, Linux, AIX, Windows and Mac configurations. Not all configurations are supported by all available software (see Table 8.11.1).

Software (Table 8.11.1) is needed both for calculation of electrostatic potentials by solving the linearized PB equation with a FDPB solver, and also for the calculation of pK_a values starting from the calculated electrostatic potentials. Executables for some of the software listed in Table 8.11.1 are available for downloading. A compiler (C, C++, or Fortran) may be needed to execute other packages. Additionally, a plotting package and molecular visualization software is useful. Web servers have become available recently that will perform pK_a calculations for user-specified structures (see Table 8.11.1).

Files

A three-dimensional molecular structure of the protein. Typically this is a PDB-formatted file obtained from X-ray crystallography, NMR spectroscopy, or a structure produced with a modeling program.

A parameter file containing atomic partial charges and radii. The format of this file will be specific to the software being used, and is usually supplied with the software package.

The software packages usually supply all other necessary files and scripts. Users can and should explore how different charges, radii, and input parameters affect the calculated pK_a values and energies. It is important to emphasize that most packages for FDPB calculations allow the user to modify existing protocols by altering input parameters. This will require the ability to modify or write Unix scripts to control the flow of the calculations.

Preparation of the input molecular structure

1. Add hydrogen atoms to the molecular structure. In this step the UHBD script `addH` adds hydrogen atoms to the fully protonated form of the protein (in the FDPB/SS method the hydrogen atoms are added to the protein in the neutral state, i.e., acidic groups are protonated and basic groups are deprotonated). As stated in the description of the Basic Protocol, this step can be critical for hydrogen-bonded groups. Some groups may need more detailed analysis than provided by these protocols.

Calculation of the molecular electrostatic potential using an FDPB solver

2. Define the set of atomic parameters to be used (`pkaS.dat`).

   Several choices are available for the set of atomic radii and atomic charges (Bashford et al., 1993; Antosiewicz et al., 1996b). The format of the `pkaS.dat` *file for the FDPB/F calculation must include the partial charges corresponding to both the neutral and the charged forms of the ionizable groups. Both the CHARMm (MacKerell et al., 1998) and PARSE (Sitkoff et al., 1994) atomic charge sets have been used with UHBD.*

3. Define input parameters (`doinp.inp`).

   Proceed the same way as in the Basic Protocol.

4. Define tautomeric state of residues.

   Proceed the same way as in the Basic Protocol.

5. Run the FDPB solver (`dosbs` script). Note that scripts used to run the FDPB/SS and FDPB/F calculations have the same format, but are not identical. The main difference between these scripts is that the FDPB solver is called four times in the FDPB/F calculations.

   In pK_a calculations with the FDPB/F protocol the FDPB solver is called twice for each residue in the charged state and twice for each group in the neutral state (once for each state for the residue in the protein, and once for each site for the residue in water for each grid specified). These steps are implemented automatically in the script `dosbs`.

   The output from this step is the `potentials` *file. This file is equivalent in format to the output from the FDPB/SS calculation, and can be used as the input to step 6 (i.e. run an in silico pH titration) in the Basic Protocol.*

GUIDELINES FOR UNDERSTANDING RESULTS

What Can Continuum Electrostatics Calculations Be Used For?

FDPB methods are useful to calculate pK_a values of individual ionizable groups. They also provide estimates of the pH and ionic strength dependence of the charge state of individual groups and of the entire protein, and of the electrostatic contributions to stability (these are all thermodynamically coupled quantities). Note that the calculation of redox properties of proteins with FDPB methods is entirely analogous to the calculation described above (Ullmann and Knapp, 1999).

Structure-based pK_a calculations such as the ones described above with the FDPB method can be useful in many situations. For example, the experimental determination of pK_a values or electrostatic energies is not possible with all proteins. This is the case with many membrane proteins in which electrostatic effects can be highly relevant for function, but where the experimental methods developed for measurement of pH dependent energetics in water-soluble proteins cannot be applied. Structure-based calculations with a method that has been tested and calibrated against experimental data are a useful alternative. Continuum calculations are also useful for dissecting molecular determinants of electrostatic effects measured experimentally. For example, calculations can be used to attempt dissection of measured pK_a values into contributions from the Born energy, from background energy, and from Coulomb interaction energy (discussed in the next section). Judicious application of computational methods for structure-based calculation of pK_a values can contribute significant insight into the structural basis of observed functional processes.

Examples of Data that Can Be Calculated

Examples of the output data from FDPB/SS calculations are shown in Figures 8.11.3 and 8.11.4 and in Table 8.11.3. The pK_a values obtained from FDPB calculations are macroscopic or apparent pK_a values, defined as the pH where a group is half-titrated. These pK_a values are directly comparable to pK_a values measured by NMR spectroscopy (Lee et al., 2002). The average charge of the protein calculated with FDPB methods, shown in Figure 8.11.4, can be compared with H^+ binding curves measured potentiometrically (Fitch et al., 2005). Three H^+ binding curves, calculated in different ways, are compared in Figure 8.11.4. One curve was calculated with the FDPB/SS method using a high value for the protein dielectric constant, $\varepsilon_{in} = 20$. This curve is similar to the curve obtained by summing the titration curves of the individual groups using the calculated pK_a values and the Henderson-Hasselbalch equation (when strongly interacting sites are present these two curves need not be the same). The third curve (dashed-dot) was

obtained using FDPB/F with $\varepsilon_{in} = 4$. This curve suggests that the pK_a values in this calculation are shifted significantly, relative to the values in model compounds in water; it illustrates how electrostatic effects can be exaggerated in calculations that use low values of ε_{in}. The fourth curve in Figure 8.11.4 (thick solid line) represents the titration of the denatured state, calculated with the pK_a values of model compounds. This curve represents the case of noninteracting and fully solvated ionizable groups. In many cases this curve is considered to be a valid representation of the H^+ titration properties of the denatured state. This is tantamount to assuming that electrostatic interactions in the denatured state are negligible. This might or might not be a valid assumption, depending on the protein. Note that the calculations yield the isoelectric point (i.e., the pH where $Q = 0$), and that the pH-dependent component (i.e., electrostatic) of the unfolding free energies can be obtained by integration of the difference between H^+ binding curves of protein in denatured and native states (Whitten and García-Moreno, 2000).

The pK_a values for a representative acidic residue in nuclease, calculated with different FDPB methods discussed ahead, are shown in Figure 8.11.5. These data illustrate the variability in the pK_a values calculated with different methods. Note that the calculated pK_a values can be very different depending on the type of calculation that is performed (FDPB/SS versus FDPB/F) and on the parameters used in the calculation. The data in Figure 8.11.6 illustrate how the calculations can be used to dissect pK_a values into different energetic contributions (e.g., Born, background, and Coulomb). Note that the calculated pK_a values usually represent averaged properties composed of contributions of different sign. For example, the self-energy term calculated with FDPB/SS for this group is destabilizing because the group in the protein is not as well hydrated as when it is in water (Fig. 8.11.6). This shifts the calculated pK_a value towards higher values.

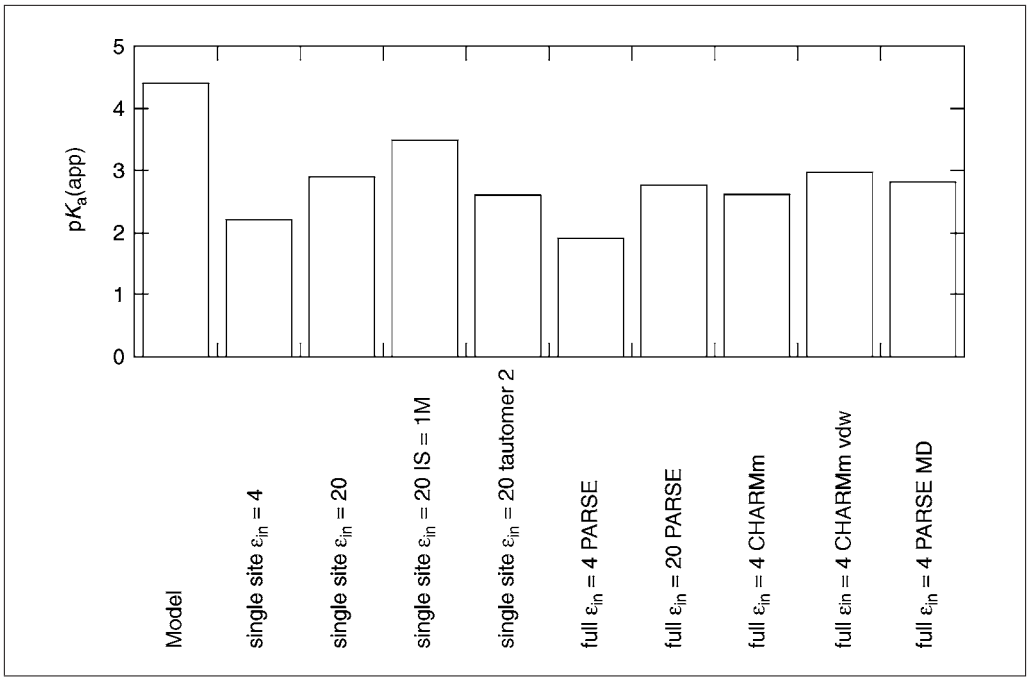


Figure 8.11.5 Comparison of pK_a(app) values of an acidic residue calculated with different FDPB methods. The set of pK_a values for a representative group in staphylococcal nuclease were calculated with nine different implementations of FDPB methods to illustrate the range of values and their sensitivity to different parameters. The effects of different values of ε_{in} (4 versus 20), different ionic strengths (100 mM versus 1 M), different charge distribution methods (FDPB/SS versus FDPB/F), different atomic charge sets (PARSE versus CHARMm), different tautomeric states, different structures (static versus MD relaxed), and different definition of the dielectric boundary (water accessible versus van der Waal's), are compared.

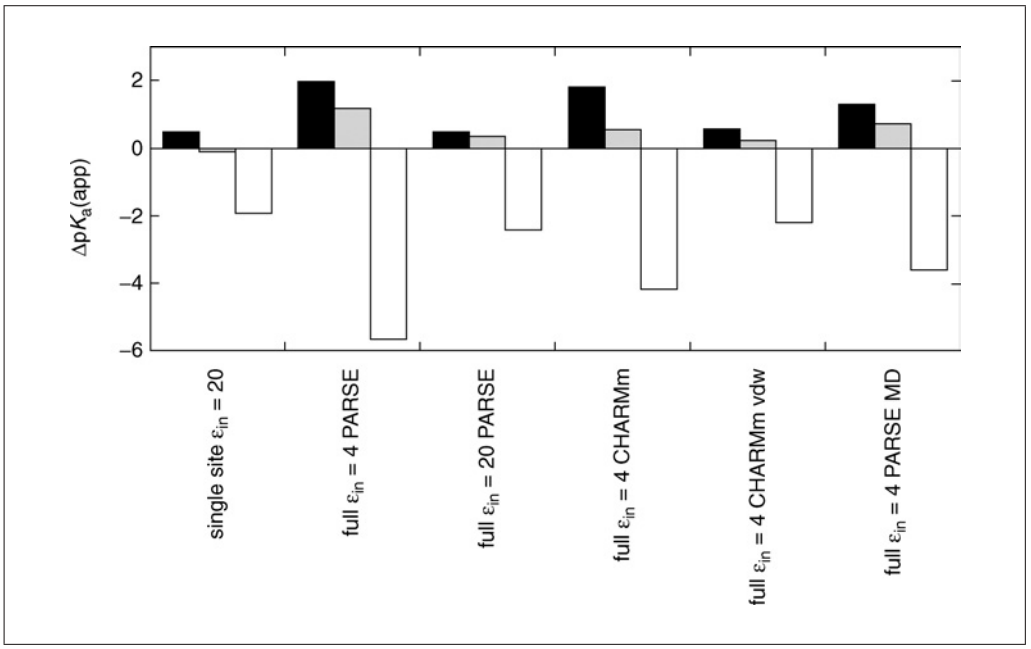


Figure 8.11.6 Energetic contributions to pK_a(app) values calculated with different FDPB methods. These data illustrate how the calculated pK_a values are parsed into Born (solid), background (gray), and Coulomb (white) energies by different implementations of FDPB methods.

However, the net shift in the pK_a value of this group, relative to the pK_a value of 4.4 of a Glu in a model compound in water, is downward because a stabilizing Coulomb interaction with a nearby Lys residue compensates for the unfavorable loss of hydration experienced by this group in the protein.

Comparison of Results Obtained with Different Protocols

The calculations outlined in the two protocols described above represent the simplest, most standard FDPB calculations. Different protocols have been developed to improve the agreement between measured and calculated data. The effects of variations in the Basic Protocol on the calculated data are illustrated in Figures 8.11.5 and 8.11.6. Use of a higher dielectric constant ($\varepsilon_{in} = 20$) always attenuates the calculated energies regardless of the exact protocol used. This attenuation is not as dramatic as what can be achieved, for example, by raising the ionic strength in a calculation from 100 mM to 1 M. The choice of tautomer (i.e., the manner in which the proton-bearing oxygen of Glu, for example, is assigned), can also lead to significant shifts in the calculated pK_a values. The consequences of a change in tautomeric state can be more extreme when ε_{in} is low and when groups are elements of hydrogen bond networks.

The FDPB/F calculation for the representative acidic residue in Figure 8.11.5 yields results similar to the FDPB/SS. Ideally, the FDPB/F method should be used when comparing calculations performed with low ε_{in} value (Antosiewicz et al., 1996b). For the group shown in Figure 8.11.5, the pK_a values calculated using the CHARMm (MacKerell et al., 1998) and PARSE (Sitkoff et al., 1994) atomic charge sets differ by 1 pK_a unit when $\varepsilon_{in} = 4$. The difference in these two calculations is not due to the Born energy. It originates with the background and Coulomb terms (Fig. 8.11.6). Two additional methods to treat electronic polarization are illustrated in Figure 8.11.5. One method defines the interface between the two different dielectric regions using the van der Waal's surface instead of the solvent-exclusion surface that is normally used (Zhou and Vijayakumar, 1997). The other method uses a structure that represents the average structure sampled in a molecular dynamics trajectory (van Vlijmen et al., 1998). Both of these treatments tend

to increase the apparent dielectric constant used to calculated energies. Consequently, the calculated pK_a values tend to become more normal (i.e., the calculated shifts in pK_a values are smaller).

Caveat Emptor

The utility of continuum methods for calculation of pK_a values is still controversial. The calculations presented in Figures 8.11.3 through 8.11.6 are for surface ionizable groups. In general, it is widely acknowledged that continuum methods are useful to calculate properties of surface groups when measures are taken to avoid exaggeration of calculated electrostatic effects. For example, the standard FDBP/F calculation with a static structure and $\varepsilon_{in} = 4$ exaggerates the magnitude of the calculated electrostatic free energies. Use of high values of ε_{in} markedly improves the agreement between calculated and measured values, albeit at the expense of clarity on the physics implicit in the calculations (i.e., the physical and structural meaning of arbitrarily high values of ε_{in} is not clear). Situations in which properties of some groups (i.e., ion pairs) are reproduced better with $\varepsilon_{in} = 4$, and those of other groups with $\varepsilon_{in} \geq 20$, are not uncommon.

The continuum methods described in this unit are not yet reliable for calculations of pK_a values of internal ionizable groups (i.e., groups that are totally or almost totally buried). For examples refer to the case of a fully internal Lys residue in staphylococcal nuclease (Fitch et al., 2002). For a critical discussion of fundamental shortcomings of continuum methods see the review by Schutz and Warshel (2001). In general, continuum methods applied to static structures are not appropriate for calculations of pK_a values and electrostatic energies for internal groups in proteins. Structural reorganization contributes significantly in these cases, and in continuum calculations this is not reproduced systematically with dielectric constants.

COMMENTARY

Background Information

Calculation of pK_a values of ionizable groups in proteins

pK_a values describe the ionization equilibrium of titratable groups. The pK_a values of ionizable groups in proteins can differ from the pK_a values of model compounds in water for two main reasons: (1) proteins contain many ionizable groups that can affect each other's pK_a values through direct Coulomb interactions, and (2) the polarity and the polarizability of proteins and water are very different. The goal of pK_a calculations is to predict pK_a values and electrostatic free energies in a physically meaningful way. The most useful computational methods for this purpose are based on the thermodynamic cycle shown in Figure 8.11.1 (Tanford and Roxby, 1972; Warshel, 1981; Matthew et al., 1985). This cycle illustrates that the calculation of pK_a values involves a calculation of the shifts in pK_a values for ionizable groups in proteins relative to the pK_a values measured experimentally in model compounds. These calculations require knowledge of the electrostatic potential in the protein-water system (p$K_a \propto$ Gibbs free energy = electrostatic potential $\times$ charge). Electrostatic potentials can be calculated with relatively simple concepts from classical electrostatics applied to the analysis of the crystallographic or NMR structure of a protein. In electrostatics calculations with all-atom methods, all the atoms in the system, their polarizabilities, and their dynamics, are treated explicitly. In continuum methods, the response of protein and water to the ionization of titratable sites is treated approximately, in terms of macroscopic dielectric constants. Despite the approximations inherent to continuum methods, they are still the method of choice for reliable pK_a calculations.

Poisson-Boltzmann electrostatics

The continuum methods for structure-based calculation of electrostatic free energies and pK_a values described in this unit are based on the linearized Poisson-Boltzmann (PB) equation. The PB equation describes the electrostatic potential in the protein-water system. Its derivation can be found in textbooks in physical chemistry or statistical thermodynamics. The application of PB electrostatics to the calculation of pK_a values in proteins was first attempted by Linderstrøm-Lang shortly after the publication of the model-dependent

solution of the linearized PB equation by Debye and Hückel (Linderstrøm-Lang, 1924). A more sophisticated model-dependent solution of the PB equation for spherical bodies was published by Tanford and Kirkwood (Tanford, 1957), and modified by Gurd and coworkers (Matthew et al., 1985; Havranek and Harbury, 1999; Garcia-Moreno and Fitch, 2004) for calculation of ionization properties of surface groups. This unit describes protocols for the calculation of pK_a values with continuum methods based on the numerical solution of the linearized PB equation by the method of finite differences (Warwicker and Watson, 1982; Klapper et al., 1986). The finite difference Poisson-Bolztmann (FDPB) method is currently the most popular continuum model for calculation of electrostatic energies and pK_a values.

Two steps in the calculation of pK_a values

The standard calculation of pK_a values involves two separate and independent steps. The first step entails calculation of the Coulomb energies of interaction between charged groups, and of the self-energy of each ionizable group. The self-energy of an ionizable moiety includes a term to account for the Born energy proper (the difference of the self-energy of a charge in media with two different dielectric constants), and another term to account for the background energy arising from interactions between the ionizable group and the permanent dipoles of the protein (in this model, the permanent dipoles are described in terms of partial charges). The Coulomb energy and the self-energy are used to calculate the shift in the pK_a value of each ionizable group in the protein relative to the pK_a value in a model compound in water. pK_a values can also be calculated using only the self-energy term. The pK_a values thus calculated are referred to as intrinsic pK_a values (pK_{int}).

The second step in the calculation of pK_a values involves the calculation of the charge state of each ionizable group. Owing to Coulomb interactions between the ionizable groups, the state of protonation of the different groups is coupled. In contrast to the calculation of the energies in the first step, which is complex and still fraught with approximations, treatment of the coupling between ionizable sites is robust, rigorous, and very efficient.

Role of the FDPB solver in pK_a calculations

The calculation of the self-energies and of the energies of Coulomb interactions requires knowledge of electrostatic potential in the protein-water system. The FDPB solver (Warwicker and Watson, 1982; Klapper et al., 1986) is used to calculate the electrostatic potential by solution of the Poisson-Botlzmann equation, which includes the effects of the counterions (i.e., electrolytes in solution) on the potential. The calculation that is performed by the FDPB solver starts with the superposition of a Cartesian lattice on the protein-solvent system defined by the atomic coordinates of the protein (Fig. 8.11.2). The molecular surface of the protein is defined by the van der Waal's surface calculated with a standard set of atomic radii. A Richard's probe (Richards, 1977) is used to define the surface that represents the boundary between the low dielectric protein interior (ε_{in}) and the high dielectric bulk water region (ε_{out}). The solvent phase includes counterions, structured about the protein according to the Boltzmann distribution function as described by the Poisson-Boltzmann equation. Atomic charges from the protein are assigned to grid points and dielectric constants are assigned to the faces surrounding the points. Electrostatic potentials in the discretized protein-water-ion system are calculated with the linearized PB equation using the method of finite differences.

The thermodynamic cycle

The thermodynamic cycle most commonly used for calculation of pK_a values of ionizable groups in proteins is illustrated in Figure 8.11.1 (Linderstrøm-Lang, 1924; Tanford, 1950; Warshel, 1981; Warshel et al., 1989; Bashford and Karplus, 1990). This cycle renders the calculation of pK_a values from first principles in a vacuum or in the protein unnecessary. Instead, the problem is transformed into a problem of calculation of the difference in electrostatic energies of the charged and neutral forms of an ionizable group, in protein and in water. The electrostatic free energy of a single group, $\Delta\Delta G_{elec}$, is given by:

$$\Delta\Delta G_{\mathrm{elec},i} = \Delta G^{\mathrm{tr}}_{i,q=z} - \Delta G^{\mathrm{tr}}_{i,q=0} = \Delta G^{\mathrm{ion}}_{i,\mathrm{prot}} - \Delta G^{\mathrm{ion}}_{i,\mathrm{model}}$$

Equation 8.11.1

where ΔG_i^{tr} is the free energy for transfer from aqueous solution to protein for site i in the charged state ($q = z$ where $z = +1$ for basic groups and -1 for acidic groups) and in the neutral state ($q = 0$). $\Delta G_i^{\mathrm{ion}}$ is the ionization energy of the group in the protein or in the model compound. $\Delta\Delta G_{elec}$ represents the shift induced by the macromolecular environment

on the pK_a value of a model compound at site *i*. The pK_a value of the ionizable group *i* in the protein can thus be defined by:

$$pK_{a,i} = pK_{a,i}^{model} - \left(\frac{z_i}{2.303RT}\right)\Delta\Delta G_{elec}$$

Equation 8.11.2

where pK_a^{model} is the experimental pK_a value of a model compound in water and $z_i = +1$ for acidic residues and -1 for basic residues. In essence, a calculation of pK_a values entails calculating the corrections needed to account for the effects of the protein environment on the equilibrium between the charged and neutral forms of an ionizable residue, relative to the ionization reaction of a model compound in water.

The statistical thermodynamic problem

The calculation of pK_a values involves an electrostatic problem (i.e., the calculation of the electrostatic potential with the FDPB solver in the protein-water system, where different phases have different dielectric properties) and a statistical thermodynamic problem (i.e., the calculation of the state of ionization of each site). The statistical thermodynamic problem arises from the fact that a protein with *N* ionizable residues can access 2^N protonation states. Because Coulomb interactions are long-range, the charge state of a given site is influenced by the charged state of all other titratable sites on the protein—a case of multiple and interacting ligand binding sites. Even with the very fast computers available today, enumeration of all the states of ionization and calculation of their electrostatic energy is only possible when the number of titratable groups is 30 or less. For larger systems approximations must be invoked. Several different approximations have been discussed in the literature (Tanford and Roxby, 1972; Bashford and Karplus, 1990; Beroza et al., 1991; Gilson, 1993; Yang et al., 1993), all of which limit the number of protonation states that must be treated explicitly.

The iterative approach of Tanford and Roxby (Tanford and Roxby, 1972) is a mean field approximation where a titratable site is assumed to have an average charge that depends on the average charge of all other groups. This approximation is exact when electrostatic interactions are weak, but it fails to converge when interactions between charged groups are strong (Bashford and Karplus, 1991). The reduced sites method by Bashford and Karplus fixes the charge state of groups that are either 95% protonated or deprotonated at a particular pH, and uses the exact calculation for all other sites (Bashford and Karplus, 1990, 1991). The cluster (HYBRID) method of Gilson (1993) identifies clusters of ionizable groups that interact strongly, based on a defined interaction energy cutoff. In this method, the ionization state of each cluster is treated exhaustively with a full ionization polynomial (i.e., all charged states of the cluster are considered explicitly), and all cluster-cluster interactions are treated with a mean field approximation. A similar approach was employed by Yang et al. (1993) with clusters determined based on a distance of interaction. Monte-Carlo methods have also been implemented for this purpose (Beroza et al., 1991; Karshikoff, 1995). These methods are successful in treating cooperative interactions of H^+ binding in proteins, and they can be used to treat this aspect of pK_a calculations with high reliability. In sum, although the exact treatment of multiple and cooperative H^+ binding reactions is impossible for large proteins, the problem has been solved by extremely accurate approximations.

Calculation of the electrostatic energy

Calculation of the electrostatic energy (Equation 8.11.1) in proteins is a difficult problem. This free energy is proportional to the electrostatic potential (ϕ), which can be described with the linearized Poisson-Boltzmann equation,

$$\nabla\cdot\left[\varepsilon(\mathbf{r})\nabla\phi(\mathbf{r})\right] - \kappa^2\varepsilon(\mathbf{r})\phi(\mathbf{r}) = -4\pi\rho(\mathbf{r})$$

Equation 8.11.3

In this equation ε and ρ are the position dependent dielectric constant and charge density respectively and κ is the inverse Debye length. The FDPB method solves this equation numerically relying on the model in Figure 8.11.2. In this model the ion-exclusion surface is defined by the dotted line. The solvent accessible surface is defined by the dashed line. As an example in Figure 8.11.2, partial charges are given for a single neutral Asp residue. In the FDPB/SS method the potential due to a unit charge at the site of ionization is calculated everywhere in the system for both the protein and the model compound. To represent the model compound in silico, the side chain of the group of interest is removed from the

protein and embedded in water. Free energies of ionization are obtained from the potential and the charge assigned to each atom.

The free energies in Equation 8.11.1 reflect the changes in solvation incurred upon transferring an amino acid from water (i.e., the model compound in water) to the protein environment. In calculations with the FDPB method this transfer free energy is described through the three terms: (1) ΔG_{Born}-the Born or hydration energy (2) ΔG_{bg}-the energy due to Coulomb interactions with the background partial charges of the protein, and (3) ΔG_{ij}-the energy due to the pairwise Coulomb interactions with all other titratable sites on the protein. The first two terms together constitute the self-energy. The Born energy is always destabilizing for groups in the protein because ionizable groups are never as well solvated in the protein as they are in water. The background energy depends on the nature of the polar atoms surrounding the residue of interest.

Parameters that can affect the calculated pK_a values

The data calculated with FDPB methods with water-soluble proteins depend primarily on four parameters: (1) the atomic charges, (2) the atomic radii, (3) the atomic structure, and (4) the internal (i.e., protein) dielectric constant. For membrane proteins, the dielectric constant used to represent the dielectric response of the bilayer is also a critical parameter. It is worth noting that the calculation of protein electrostatics is really about calculation of the polarization or the response of the environment to charge. In continuum electrostatics methods, the response of both protein and water to the presence of charge is described with macroscopic dielectric constants. All the contributions to the dielectric response that are not modeled well by the dielectric constants chosen for the calculations, must be modeled explicitly. Because electrostatic energy is inversely proportional to the dielectric constant, the dielectric constants used in the calculation have a significant impact on the calculated energies and pK_a shifts. All energies will become larger as the dielectric constant decreases.

How should the protein dielectric constant, ε_{in}, be treated?

The value of ε_{in} used in FDPB calculations is controversial and the focus of research in several laboratories. ε_{in} is the single most important parameter in continuum electrostatic calculations. It is well established that in calculations of pK_a values of surface groups with FDPB methods, agreement between calculated and measured values is maximized when a high ε_{in} value ($\varepsilon_{in} = 20$) is used (Antosiewicz et al., 1994; Antosiewicz et al., 1996b; Warwicker, 2004; Teixeira et al., 2005). Lower values exaggerate the calculated energies regardless of the implementation of FDPB used. The value of $\varepsilon_{in} = 20$ is much higher than the dielectric constants of 2 to 4 measured experimentally in dried proteins (Harvey and Hoekstra, 1972). Values of $\varepsilon_{in} = 2$ are thought to reflect mainly electronic polarizability, and values of $\varepsilon_{in} = 4$ are thought to include additional contributions by relaxation of permanent dipoles (Schutz and Warshel, 2001). Simulations suggest that the dielectric constant experienced by ionizable groups in the interior is much lower than for surface groups (Simonson, 2003). Some ionizable groups that are buried or partially buried are better modeled with values of ε_{in} lower than 10 (Trylska et al., 1999; Fitch et al., 2002). However, for surface groups, use of high values of $\varepsilon_{in} = 20$ remains the best choice.

Why do high values of ε_{in} improve results?

High values of ε_{in} are meant to account implicitly for reorganization processes not treated explicitly by the FDPB algorithms, especially in calculations with static structures (Harvey and Hoekstra, 1972; Gilson, 1995; Krishtalik et al., 1997; Sham et al., 1997, 1998; Warshel and Papazyan, 1998; Simonson et al., 1999, 2004; Schutz and Warshel, 2001; Fitch et al., 2002; Simonson, 2003; Archontis and Simonson, 2005). By definition, the dielectric constant should reproduce the equilibrium dipole fluctuations and polarizations induced by a charge. However, proteins are structurally and dynamically heterogeneous, therefore, a dielectric tensor would likely be more appropriate to describe the difference in polarizability in different regions in a protein (Baker et al., 2000; Holst et al., 2000). All-atom calculations avoid the use of dielectric constants all together (i.e., $\varepsilon_{in} = 1$) because all the contributions to the dielectric response of the protein by charges, dipoles, relaxation and polarization are treated explicitly. This type of calculation can contribute significant insight into the origins of observed effects, but they are still not sufficiently accurate to be used for prediction of pK_a values.

The protocols described in this chapter use static structures. In calculations with this protocol, all equilibrium fluctuations that

affect electrostatic energies must be reproduced implicitly through the dielectric constant. The value of $\varepsilon_{in} = 20$ needed in FDPB/SS calculations to reproduce experimental pK_a values of surface residues is thought to represent the effects of dielectric reorganization that are not treated explicitly in the simulations (Archontis and Simonson, 2005). Note that even when $\varepsilon_{in} = 20$ is used, FDPB calculations can and do fail—even with this high dielectric constant they can exaggerate the magnitude of shifts in pK_a values (Fitch et al., 2005). Some FDPB methods use more than one value of ε_{in}; they allow for multiple values depending on the location of the ionizable group of interest (Antosiewicz et al., 1994; Karshikoff, 1995; Simonson and Perahia, 1995; Antosiewicz et al., 1996b; Demchuk and Wade, 1996; Voges and Karshikoff, 1998; Nielsen and Vriend, 2001; Schaefer et al., 2001). MD simulations have also been used to improve the agreement between calculated and measured pK_a values (van Vlijmen et al., 1998). Multiple structures along the MD trajectory are selected for pK_a calculations with FDPB/F with $\varepsilon_{in} = 4$, and the results from different calculations are averaged. In this type of calculation the MD simulations are being used to relax the protein. The results obtained with MD-relaxed structures using FDPB/F and $\varepsilon_{in} = 4$ tend to be comparable to the ones obtained with a static structure with $\varepsilon_{in} = 20$. The MD-based method has two drawbacks. First, many relevant modes of relaxation can be missed in the limited time scale sampled with standard MD simulations. Second, the correction with MD is usually applied in a physically incorrect manner. The correct physical application of this method would require separate MD simulations for groups in the neutral and in the charged state (Schutz and Warshel, 2001).

Other empirical approximations that improve agreement between calculated and experimental data

In another empirical scheme developed to maximize the agreement between calculated and measured pK_a values, the dielectric boundary between the low $\varepsilon_{in} = 4$ protein and the $\varepsilon_{in} = 80$ water phase is defined using the van der Waal's surface instead of the water-accessible surface (Antosiewicz et al., 1994; Vijayakumar and Zhou, 2001). This effectively increases the local value of ε_{in} because contributions from the $\varepsilon_{in} = 80$ phase to the net dielectric effect are increased. Yet another approach to maximizing agreement between calculated and measured effects depends on the use of multiple structures rather than the use of a single static structure. Multiple X ray structures (Bashford and Karplus, 1990; Bashford et al., 1993; Yang et al., 1993; Antosiewicz et al., 1994), structures from NMR spectroscopy (Antosiewicz et al., 1996b; Khare et al., 1997; Dillet et al., 1998; Gorfe et al., 2002), or structures generated from MD simulations (Bashford and Gerwert, 1992; Yang et al., 1993; Sham et al., 1997; Wlodek et al., 1997; Zhou and Vijayakumar, 1997; van Vlijmen et al., 1998; Koumanov et al., 2001; Gorfe et al., 2002; Soares et al., 2002) have been used for this purpose. This approach minimizes the very strong dependence of the calculated effects on the details of the structure, especially when $\varepsilon_{in} \approx 4$ is used.

Problems with tautomeric states

Among recent and promising approaches to optimize calculations, the one that deserves special attention involves the explicit and rigorous treatment of tautomeric states and multiple protonation sites (Bashford et al., 1993; Oberoi and Allewell, 1993; Nielsen et al., 1999; Trylska et al., 1999; Demchuk et al., 2000; Baptista and Soares, 2001; Koumanov et al., 2002). The choice of tautomeric state (i.e., placement of protons) in a calculation can be critical. Similarly, multiple rotamers or multiple side chain conformations can be considered in attempts to improve the calculations (Bashford and Karplus, 1990; You and Bashford, 1995; Beroza and Case, 1996, 1998). These developments have culminated in a method that optimizes proton placement and side chain rotamers with a Monte Carlo procedure. This sophisticated multiple conformational continuum electrostatic method (MCCE) developed by Gunner and colleagues has improved the accuracy of pK_a calculations (Alexov and Gunner, 1997, 1999; Georgescu et al., 2002; Alexov, 2003).

Choice of atomic parameters

One goal of many FDPB methods is to have predictive power while retaining use of a low value of ε_{in}. When a low value of ε_{in} is used in electrostatic calculations, the choice of atomic parameters can become important (Bashford et al., 1993; Antosiewicz et al., 1996b; Teixeira et al., 2005). The Born energy, for example, depends on the accessible surface, thus its value will depend on the choice of van der Waal's radii. Background energies are partly determined by the atomic partial charge set used. Atomic parameter

sets are usually adjusted to agree with quantum mechanical calculations of small molecules (Neria et al., 1996). The PARSE atomic parameter set was parameterized to reproduce solvation energies of small organic molecules using the PB model (Sitkoff et al., 1994). This parameter set has been shown to improve results in calculations with $\varepsilon_{in} = 4$ (Antosiewicz et al., 1996a,b). In general, the sensitivity of the calculations to the atomic parameters increases when $\varepsilon_{in} = 4$ is used, as does the sensitivity to structural details.

Site-bound water molecules

In cases where waters are known to play structural or catalytic roles, it is often necessary to treat some water molecules explicitly. This is especially important for internal groups, which are often buried in complex with water molecules (Dwyer et al., 2000). A number of treatments have been proposed to handle these situations within the framework of FDPB calculations (Yang et al., 1993; Warwicker, 1994, 1997; Gibas and Subramaniam, 1996; Scharnagl et al., 1999; Trylska et al., 1999; Spassov et al., 2001; Fitch et al., 2002).

Electrostatic interactions in unfolded proteins

The electrostatic contributions to the stability of a protein that can be calculated with FDPB continuum methods assumes that in the unfolded state, all ionizable groups titrate with the pK_a values of model compounds. Experimental evidence suggests that this is not a valid assumption for all proteins (Kuhlman et al., 1999). Several approaches have been developed to estimate the magnitude of electrostatic effects in the denatured state of proteins (Dimitrov and Crichton, 1997; Schaefer et al., 1998; Elcock, 1999; Warwicker, 1999; Kundrotas and Karshikoff, 2002; Zhou, 2002, 2003). When the H^+ binding properties of the unfolded protein are not reproduced correctly, the predicted pH dependence of the electrostatic free energy can be in significant error (Fitch et al., 2005).

Size of the grid used in FDPB calculations

The choice of grid size should be evaluated for each particular application. The grid specifications listed in Table 8.11.2 should be sufficient for small proteins (Gilson et al., 1988; Yang et al., 1993). Molecules with a single dimension greater than 40 Å should use a larger coarse grid (centered on the protein) allowing for an adequate solvent region and well defined boundary potentials. Several calculations should be performed to test the validity of the chosen value. Focused grids, centered on the titrating residue, allow for finer grids to be used at the charged site for short-range interactions, without increasing computational costs. The smallest grid used should encompass the charge moiety of the titrating residue. Focusing is likely to be critical for areas of steep gradient of electrostatic potential.

How to decide which protocol to use?

As discussed previously, FDPB calculations with static structures and $\varepsilon_{in} = 20$ give the best overall agreement between calculated and experimental pK_a values for surface residues. This is true for both the FDPB/SS and FDPB/F methods. It is noted here again that if one chooses to work with a low value of ε_{in}, then the full charge distribution method, FDPB/F, should be used. Use of low values of ε_{in} might be warranted, for example, if the groups of interest are partially buried or involved in networks of hydrogen bonds and ions pairs. Because FDPB/F calculations require four FDPB calculations, they are slower than the FDPB/SS calculations, which require only two. For smaller proteins this should not be an issue as computer speeds are adequate to handle these calculations in a reasonable time. In general, for larger systems or for systematic calculations with many proteins, FDPB/SS calculations with $\varepsilon_{in} = 20$ are likely to be more useful.

Suggestions for Further Analysis

Continuum methods are still the most widely used methods for calculation of pK_a values and electrostatic energies. Other methods are available that are not based on FDPB. Interested readers should become familiar with: (1) screened Coulomb potential method (Mehler and Guarnieri, 1999) (2) PROPKA (Li et al., 2005) and (3) Tanford-Kirkwood method (Matthew et al., 1985; Havranek and Harbury, 1999). One method that deserves special mention is the family of algorithms based on the Protein Dipole Langevin Dipole (PDLD) method developed by Warshel and colleagues (Warshel and Levitt, 1976; Warshel, 1981; Warshel and Russell, 1984; Warshel and Aqvist, 1991; Sham et al., 1997; Schutz and Warshel, 2001). The PDLD methods relax the atom-centered partial charge assumption, which is likely a poor choice for closely interacting atoms. Instead, polarizable protein dipoles are modeled. The PDLD methods also improve on some of the most limiting features of standard continuum methods, by

allowing for explicit treatment of some aspects of protein reorganization concomitant with ionization. In general, the physical principles embodied in PDLD calculations are more rigorous than in standard continuum calculations; therefore, they can contribute more physical and structural insight than standard continuum methods, especially in problems of structure-function relationship related to catalysis and bioenergetics, where structural reorganization matters greatly. Interested readers should also be aware of interesting applications of standard FDPB methods to the analysis of various aspects of protein structure and function. For example, the THEMATICS algorithm uses structure-based calculations of microscopic titration curves with FDPB methods to identify functional groups in enzymes (Ondrechen et al., 2001).

Literature Cited

Alexov, E. 2003. Role of the protein side-chain fluctuations on the strength of pair-wise electrostatic interactions: Comparing experimental with computed pK(a)s. *Proteins* 50:94-103.

Alexov, E.G., and Gunner, M.R. 1997. Incorporating protein conformational flexibility into the calculation of pH-dependent protein properties. *Biophys. J.* 74:2075-2093.

Alexov, E.G. and Gunner, M.R. 1999. Calculated protein and proton motions coupled to electron transfer: Electron transfer from Q_A- to Q_B in bacterial photosynthetic reaction centers. *Biochemistry* 38:8253-8270.

Antosiewicz, J., McCammon, J.A., and Gilson, M.K. 1994. Prediction of pH-dependent properties of proteins. *J. Mol. Biol.* 238:415-436.

Antosiewicz, J., Briggs, J.M., Elcock, A.H., Gilson, M.K., and McCammon, A. 1996a. Computing ionization states of proteins with a detailed charge model. *J. Comput. Chem.* 17:1633-1644.

Antosiewicz, J., McCammon, A.J., and Gilson, M.K. 1996b. The determinants of pK_as in proteins. *Biochemistry* 35:7819-7833.

Archontis, G. and Simonson, T. 2005. Proton binding to proteins: A free-energy component analysis using a dielectric continuum model. *Biophys. J.* 88:3888-3904.

Baker, N., Holst, M., and Wang, F. 2000. Adaptive multilevel finite element solution of the Poisson-Boltzmann equation II. Refinement at solvent-accessible surfaces in biomolecular systems. *J. Comput. Chem.* 21:1343-1352.

Baptista, A.M. and Soares, C.M. 2001. Some theoretical and computational aspects of the inclusion of proton isomerism in the protonation equilibrium of proteins. *J. Phys. Chem. B* 105:293-309.

Bashford, D. and Gerwert, K. 1992. Electrostatic calculations of the pK_a values of ionizable groups in bacteriorhodopsin. *J. Mol. Biol.* 224:473-486.

Bashford, D. and Karplus, M. 1990. pK_a's of ionizable groups in proteins: Atomic detail from a continuum electrostatic model. *Biochemistry* 29:10219-10225.

Bashford, D. and Karplus, M. 1991. Multiple-site titration curves of proteins: An analysis of exact and approximate methods for their calculation. *J. Phys. Chem.* 95:9556-9561.

Bashford, D., Case, D., Dalvit, C., Tennant, L., and Wright, P. 1993. Electrostatic calculations of side-chain pK_a values in myoglobin and comparison with NMR data for histidines. *Biochemistry* 32:8045-8056.

Beroza, P. and Case, D.A. 1996. Including side chain flexibility in continuum electrostatic calculations of protein titration. *J. Phys. Chem.* 100:20156-20163.

Beroza, P. and Case, D.A. 1998. Methods to address the change in conformation resulting from ionization process OR fluctuations inherent at a particular pH or in a particular structure. *Methods Enzymol.* 295:170-189.

Beroza, D., Fredkin, D.R., Okamura, M.Y., and Feher, G. 1991. Protonation of interacting residues in a protein by a Monte Carlo method: Application to lysozyme and the photosynthetic reaction center of rhodobacter sphaeroides. *Proc. Natl. Acad. Sci. U.S.A.* 88:5804-5808.

Brooks, B.R., Bruccoleri, R.E., Olafson, B.D., States, D.J., Swaminathan, S., and Karplus, M. 1983. CHARMM: A Program for macromolecular energy, minimization, and dynamics calculations. *J. Comput. Chem.* 4:187-217.

Davis, M.E., Madura, J.D., Luty, B.A., and McCammon, J.A. 1991. Electrostatics and diffusion of molecules in solution-Simulations with the University of Houston Brownian Dynamics program. *Comp. Phys. Commun.* 62:187-197.

Demchuk, E. and Wade, R.C. 1996. Improving the continuum dielectric approach to calculating pK_a s of ionizable groups in proteins. *J. Phys. Chem.* 100:17373-17387.

Demchuk, E., Genick, U.K., Woo, T.T., Getzoff, E.D., and Bashford, D. 2000. Protonation states and pH titration in the photocycle of photoactive yellow protein. *Biochemistry* 39:1100-1113.

Dillet, V., Dyson, H.J., and Bashford, D. 1998. Calculations of electrostatic interactions and pK(a)s in the active site of Escherichia coli thioredoxin. *Biochemistry* 37:10298-10306.

Dimitrov, R.A. and Crichton, R.R. 1997. Self-consistent field approach to protein structure and stability .1. pH dependence of electrostatic contribution. *Proteins* 27:576-596.

Dwyer, J.J., Gittis, A.G., Karp, D.A., Lattman, E.E., Spencer, D.S., Stites, W.E., and García-Moreno E., B. 2000. High apparent dielectric constants in the interior of a protein reflect water penetration. *Biophys. J.* 79:1610-1620.

Elcock, A.H. 1999. Realistic modeling of the denatured states of proteins allows accurate calculations of the pH dependence of protein stability. *J. Mol. Biol.* 294:1051-1062.

Fitch, C.A., Karp, D.A., Lee, K.K., Stites, W.E., Lattman, E.E., and García-Moreno E., B. 2002. Experimental pK_a values of buried residues: analysis with continuum methods and role of water penetration. *Biophys. J.* 82:3289-3304.

Fitch, C.A., Whitten, S.T., Hilser, V.J., and García-Moreno E., B. 2005. Molecular mechanism of pH-driven conformational transitions of proteins: Insights from continuum electrostatics calculations of acid unfolding. *Proteins* 63:113-126.

Garcia-Moreno E., B. and Fitch, C.A. 2004. Structural interpretation of pH and salt-dependent processes in proteins with computational methods. *In* Energetics Of Biological Macromolecules, Pt E. (M. J.M. Holt, M.L. Johnson, and G.K. Ackers, eds.) pp. 20-51. Academic Press Inc., San Diego.

Georgescu, R.E., Alexov, E.G., and Gunner, M.R. 2002. Combining conformational flexibility and continuum electrostatics for calculating pK(a)s in proteins. *Biophys. J.* 83:1731-1748.

Gibas, C.J. and Subramaniam, S. 1996. Explicit solvent models in protein pK_a calculations. *Biophys. J.* 71:138-147.

Gilson, M.K. 1993. Multiple-site titration and molecular modeling: Two rapid methods for computing energies and forces for ionizable groups in proteins. *Proteins* 15:266-282.

Gilson, M.K. 1995. Theory of electrostatic interactions in macromolecules. *Curr. Biol.* 5:216-223.

Gilson, M.K., Sharp, K.A., and Honig, B.H. 1988. Calculating the electrostatic potential of molecules in solution-Method and error assessment. *J. Comput. Chem.* 9:327-335.

Gorfe, A.A., Ferrara, P., Caflisch, A., Marti, D.N., Bosshard, H.R., and Jelesarov, I. 2002. Calculation of protein ionization equilibria with conformational sampling: pK(a) of a model leucine zipper, GCN4 and barnase. *Proteins* 46:41-60.

Harvey, S.C. and Hoekstra, P. 1972. Dielectric relaxation spectra of water adsorbed on lysozyme. *J. Phys. Chem.* 76:2987-2994.

Havranek, J.J. and Harbury, P.B. 1999. Tanford-Kirkwood electrostatics for protein modeling. *Proc. Natl. Acad. Sci. U.S.A.* 96:11145-11150.

Holst, M., Baker, N., and Wang, F. 2000. Adaptive multilevel finite element solution of the Poisson-Boltzmann equation I. Algorithms and examples. *J. Comput. Chem.* 21:1319-1342.

Jorgensen, W.L. and Tirado-Rives, J. 1988. The OPLS potential functions for proteins. Energy minimizations for crystals of cyclic peptides and crambin. *J. Am. Chem. Soc.* 110:1657-1666.

Karshikoff, A. 1995. A simple algorithm for the calculation of multiple-site titration curves. *Protein Eng.* 8:243-248.

Khare, D., Alexander P., Antosiewicz J., Bryan P., Gilson M., and Orban J. 1997. pK_a measurements from nuclear magnetic resonance for the B1 and B2 immunoglobin G-binding domains of protein G: Comparison with calculated values for nuclear magnetic resonance and x-ray structures. *Biochemistry* 36:3580-3589.

Klapper, I., Hagstrom, R., Fine, R., Sharp, K., and Honig, B. 1986. Focussing of electric fields in the active site of Cu-Zn superoxide dismutase: Effects of ionic strength and amino-acid modification. *Proteins* 1:47-59.

Koumanov, A., Spitzner, N., Rüterjans, H., and Karshikoff, A.D. 2001. Ionization properties of titratable groups in ribonuclease T1 II. Electrostatic analysis. *Eur. Biophy. J.* 30:198-206.

Koumanov, A., Ruterjans, H., and Karshikoff, A. 2002. Continuum electrostatic analysis of irregular ionization and proton allocation in proteins. *Proteins* 46:85-96.

Krishtalik, L.I., Kuznetsov, A.M., and Mertz, E.L. 1997. Electrostatics of proteins: Description in terms of two dielectric constants simultaneously. *Proteins* 28:174-182.

Kuhlman, B., Luisi, D., Young, P., and Raleigh, D. 1999. pK_a values and the pH dependent stability of the N-terminal domain of L9 as probes of electrostatic interactions in the denatured state: Differentiation between local and nonlocal interactions. *Biochemistry* 38:4896-4903.

Kundrotas, P.J. and Karshikoff, A. 2002. Modeling of denatured state for calculation of the electrostatic contribution to protein stability. *Prot. Sci.* 11:1681-1686.

Lee, K.K., Fitch, C.A., Lecomte, J.T.J., and Garcia-Moreno E., B. 2002. Electrostatic effects in highly charged proteins: Salt sensitivity of pK_a values of histidines in staphylococcal nuclease. *Biochemistry* 41:5656-5667.

Li, H., Robertson, A.D., and Jensen, J.H. 2005. Very fast empirical prediction and rationalization of protein pK_a values. *Proteins* 61:704-721.

Linderstrøm-Lang, K. 1924. On the ionization of proteins. *C R Trav. Lab. Carlsberg* 15:1-29.

MacKerell, A.D., Bashford, D., Bellott, M., Dunbrack, R.L., Evanseck, J.D., Field, M.J., Fischer, S., Gao, J., Guo, H., Ha, S., Joseph-McCarthy, D., Kuchnir, L., Kuczera, K., Lau, F.T.K., Mattos, C., Michnick, S., Ngo, T., Nguyen, D.T., Prodhom, B., Reiher, W.E., Roux, B., Schlenkrich, M., Smith, J.C., Stote, R., Straub, J., Watanabe, M., Wiorkiewicz-Kuczera, J., Yin, D., and Karplus, M. 1998. All-atom empirical potential for molecular modeling and dynamics studies of proteins. *J. Phys. Chem. B* 102:3586-3616.

Madura, J.D., Briggs, J.M., Wade, R.C., Davis, M.E., Luty, B.A., Ilin, A., Antosiewicz, J., Gilson, M.K., Bagheri, B., Scott, L.R., and McCammon, J.A. 1995. Electrostatics and diffusion of molecules in solution-Simulations with the University of Houston Brownian Dynamics program. *Comp. Phys. Commun.* 91:57-95.

Matthew, J.B., Gurd, F.R.N., García-Moreno E., B., Flanagan, M.A., March, K.L., and Shire, S.J. 1985. pH-dependent properties in proteins. *CRC Crit. Rev. Biochem.* 18:91-197.

Mehler, E.L. and Guarnieri, F. 1999. A self-consistent, microenvironment modulated screened coulomb potential approximation to calculate pH-dependent electrostatic effects in proteins. *Biophys. J.* 75:3-22.

Neria, E., Fischer, S., and Karplus, M. 1996. Simulation of activation free energies in molecular systems. *J. Chem. Phys.* 105:1902-1921.

Nielsen, J.E. and Vriend, G. 2001. Optimizing the hydrogen-bond network in Poisson-Boltzmann equation-based pK(a) calculations. *Proteins* 43:403-412.

Nielsen, J.E., Andersen, K.V., Honig, B., Hooft, R.W.W., Klebe, G., Vriend, G., and Wade, R.C. 1999. Improving macromolecular electrostatics calculations. *Prot. Eng.* 12:657-662.

Oberoi, H. and Allewell, N.M. 1993. Multigrid solution of the nonlinear Poisson-Boltzmann Equation and calculation of titration curves. *Biophys. J.* 65:48-55.

Ondrechen, M.J., Clifton, J.G., and Ringe, D. 2001. THEMATICS: A simple computational predictor of enzyme structure from function. *Proc. Natl. Acad. Sci. U.S.A.* 98:12473-12478.

Richards, F.M. 1977. Areas, volumes, packing, and protein structure. *Annu. Rev. Biophys. Bioeng.* 6:151-176.

Schaefer, M., Van Vlijmen, H.W.T., and Karplus, M. 1998. Electrostatic contributions to molecular free energies in solution. *In* Advances In Protein Chemistry, Vol 51. (E. Di Cera, D.E. Eisenberg, and F.M. Richards, eds.) pp. 1-57. Academic Press Inc., San Diego.

Schaefer, M., Bartels, C., Leclerc, F., and Karplus, M. 2001. Effective atom volumes for implicit solvent models: Comparison between Voronoi volumes and minimum fluctuation volumes. *J. Comput. Chem.* 22:1857-1879.

Scharnagl, C., Raupp-Kossmann, R., and Fischer, S.F. 1999. Molecular basis for pH sensitivity and proton transfer in green fluorescent protein: Protonation and conformational substates from electrostatic calculations. *Biophys. J.* 77:1839-1857.

Schutz, C.N. and Warshel, A. 2001. What are the dielectric "constants" of proteins and how to validate electrostatic models? *Protein* 44:400-417.

Sham, Y.Y., Chu, Z.T., and Warshel, A. 1997. Consistent calculations of pK_a's of ionizable residues in proteins: Semi-microscopic and microscopic approaches. *J. Phys. Chem. B* 101:4458-4472.

Sham, Y.Y., Muegge, I., and Warshel, A. 1998. The effect of protein relaxation on charge-charge interactions and dielectric constants of proteins. *Biophys. J.* 74:1744-1753.

Simonson, T. 2003. Electrostatics and dynamics of proteins. *Reports On Progress In Physics* 66:737-787.

Simonson, T. and Perahia, D. 1995. Internal and interfacial dielectric-properties of Cytochrome-C from molecular-dynamics in aqueous-solution. *Proc. Natl. Acad. Sci. U.S.A.* 92:1082-1086.

Simonson, T., Archontis, G., and Karplus, M. 1999. A Poisson-Boltzmann study of charge insertion in an enzyme active site: The effect of dielectric relaxation. *J. Phys. Chem. B* 103:6142-6156.

Simonson, T., Carlsson, J., and Case, D.A. 2004. Proton binding to proteins: pK(a) calculations with explicit and implicit solvent models. *J. Am. Chem. Soc.* 126:4167-4180.

Sitkoff, D., Sharp, K.A., and Honig, B. 1994. Accurate calculation of hydration free energies using macroscopic solvent models. *J. Phys. Chem.* 98:1978-1988.

Soares, T.A., Lins, R.D., Straatsma, T.P., and Briggs, J.M. 2002. Internal dynamics and ionization states of the macrophage migration inhibitory factor: Comparison between wild-type and mutant forms. *Biopolymers* 65:313-323.

Spassov, V.Z., Luecke, H., Gerwert, K., and Bashford, D. 2001. pK(a) calculations suggest storage of an excess proton in a hydrogen-bonded water network in bacteriorhodopsin. *J. Mol. Biol.* 312:203-219.

Tanford, C. 1950. Preparation and properties of serum and plasma proteins. XXIII. Hydrogen ion equilibria in native and modified human serum albumins. *J. Am. Chem. Soc.* 72:441-451.

Tanford, C. 1957. Theory of protein titration curves II. Calculations for simple models at low ionic strength. *J. Am. Chem. Soc.* 79:5340-5347.

Tanford, C. and Roxby, R. 1972. Interpretation of protein titration curves: Application to lysozyme. *Biochemistry* 11:2192-2198.

Teixeira, V.H., Cunha, C.A., Machuqueiro, M., Oliveira, A.S.F., Victor, B.L., Soares, C.M., and Baptista, A.A. 2005. On the use of different dielectric constants for computing individual and pairwise terms in Poisson-Boltzmann studies of protein ionization equilibrium. *J. Phys. Chem. B* 109:14691-14706.

Trylska, H., Antosiewicz, J., Geller, M., Hodge, C.N., Klabe, R.M., Head, M.S., and Gilson, M.K. 1999. Thermodynamic linkage between the binding of protons and inhibitors to HIV-1 protease. *Prot. Sci.* 8:180-195.

Ullmann, G.M. and Knapp, E.W. 1999. Electrostatic models for computing protonation and redox equilibria in proteins. *Eur. Biophys. J.* 28:533-551.

van Vlijmen, H.W.T., Schaefer, M., and Karplus, M. 1998. Improving the accuracy of protein pK_a calculations: Conformational averaging versus the average structure. *Proteins* 33:145-158.

Vijayakumar, M. and Zhou, H.-X. 2001. Salt bridges stabilize the folded structure of barnase. *J. Phys. Chem. B* 105:7334-7340.

Voges, D. and Karshikoff, A. 1998. A model of a local dielectric constant in proteins. *J. Chem. Phys.* 108:2219-2227.

Warshel, A. 1981. Calculations of enzymatic-reactions - calculations of pK_a, proton-transfer reactions, and general acid catalysis reactions in enzymes. *Biochemistry* 20:3167-3177.

Warshel, A. and Aqvist, J. 1991. Electrostatic Energy and Macromolecular Function. *Annu. Rev. Biophys. Biophys. Chem.* 20:267-298.

Warshel, A. and Levitt, M. 1976. Theoretical studies of enzymic reactions-Dielectric, electrostatic and steric stabilization of carbonium-ion in reaction of lysozyme. *J. Mol. Biol.* 103:227-249.

Warshel, A. and Papazyan, A. 1998. Electrostatic effects in macromolecules: Fundamental concepts and practical modeling. *Curr. Opin. Struct. Biol.* 8:211-217.

Warshel, A. and Russell, S.T. 1984. Calculations of electrostatic interactions in biological-systems and in solutions. *Q. Rev. Biophys.* 17:283-422.

Warshel, A., Naray-Szabo, G., Sussman, F., and Hwang, J.K. 1989. How do serine proteases really work. *Biochemistry* 28:3629-3637.

Warwicker, J. 1994. Improved continuum electrostatic modeling in proteins, with comparison to experiment. *J. Mol. Biol.* 236:887-903.

Warwicker, J. 1997. Improving pK_a calculations with consideration of hydration entropy. *Prot. Eng.* 10:809-814.

Warwicker, J. 1999. Simplified methods for pK(a) and acid pH-dependent stability estimation in proteins: Removing dielectric and counterion boundaries. *Prot. Sci.* 8:418-425.

Warwicker, J. 2004. Improved pK(a) calculations through flexibility based sampling of a water-dominated interaction scheme. *Prot. Sci.* 13:2793-2805.

Warwicker, J. and Watson, H.C. 1982. Calculation of the electric potential in the active site cleft due to a-helix dipoles. *J. Mol. Biol.* 157:671-679.

Whitten, S.T. and García-Moreno E., B. 2000. pH dependence of stability of staphylococcal nuclease: Evidence of substantial electrostatic interactions in the denatured state. *Biochemistry* 39:14292-14304.

Wlodek, S.T., Antosiewicz, J., and McCammon, J.A. 1997. Prediction of titration properties of structures of a protein derived from molecular dynamics trajectories. *Prot. Sci.* 6:373-382.

Yang, A.-S., Gunner, M.R., Sampogna, R., Sharp, K., and Honig, B. 1993. On the calculation of pK_as in proteins. *Proteins* 15:252-265.

You, T. and Bashford, D. 1995. Conformation and hydrogen ion titration of proteins: A continuum electrostatic model with conformational flexibility. *Biophys. J.* 69:1721-1733.

Zhou, H.-X. 2002. A Gaussian-chain model for treating residual charge-charge interactions in the unfolded state of proteins. *Proc. Natl. Acad. Sci. U.S.A.* 99:3569-3574.

Zhou, H.-X. 2003. Direct test of the Gaussian-chain model for treating residual charge-charge interactions in the unfolded state of proteins. *J. Am. Chem. Soc.* 125:2060-2061.

Zhou, H.-X. and Vijayakumar, M. 1997. Modeling of protein conformational fluctuations in pK_a predictions. *J. Mol. Biol.* 267:1002-1011.

Key References

Davis et al., 1991. See above.

Madura et al., 1995. See above.

Antosiewicz et al., 1996a. See above.

The above references contain a description of methodology for FDPB calculations with UHBD.

Simonson, 2003. See above.

Ullman and Knapp, 1999. See above.

Garcia-Moreno and Fitch, 2004. See above.

Archontis and Simonson, 2005. See above.

The above references contain reviews of PB and FDPB methods.

Sham et al., 1997. See above.

Simonson et al., 1999. See above.

Schutz and Warshel, 2001. See above.

Simonson et al., 2004. See above.

The above references contain in depth discussions of problems of protein reorganization.

Internet Resources

See Table 8.11.1.

http://enzyme.ucd.ie/Science/pKa/pKa_introduction)

Prof. Jens Nielsen's website discusses many aspects of pK_a calculations. Tools are available at this website for calculations and analysis of H^+ titration curves.

http://honiglab.cpmc.columbia.edu/mcce/mcce.html

Explanation of MCCE method

Contributed by Carolyn A. Fitch and Bertrand García-Moreno E.
Johns Hopkins University
Baltimore, Maryland

Flexible Ligand Docking with Glide

High-throughput crystallography and genomic efforts have lead to increased availability of high-resolution crystal structures of protein receptors. Such target protein structures are very valuable in drug-discovery projects since detailed knowledge of protein-ligand interactions can facilitate the discovery of leads and optimization of leads to drugs. A common computational strategy for structure-based lead discovery is analogous to high-throughput screening (HTS), namely to screen compounds from a virtual database for their predicted affinity to a particular protein target. Analysis of the docked protein-ligand geometries provides insight into driving forces of the binding process. Glide performs such high-throughput virtual screening (HTVS) experiments in silico, predicting protein-ligand binding modes and ranking ligands according to empirical scoring functions.

Glide docks flexible ligands into a rigid receptor structure by rapid sampling of the conformational, orientational, and positional degrees of freedom of the ligand. There are three modes of running Glide which differ in how ligand degrees of freedom are sampled and in the scoring function employed. All three modes generate an exhaustive set of conformers for a ligand and employ a series of hierarchical filters to enable rapid evaluation of ligand degrees of freedom. The SP GlideScore scoring function is used to rank compounds docked by SP or HTVS Glide. XP Glide begins with SP Glide docking and then refines the predicted docking modes using an anchor-and-grow algorithm to more thoroughly sample ligand degrees of freedom. The XP GlideScore scoring function includes special recognition terms to identify and reward structural motifs important to binding.

To introduce desired bias into a docking experiment it is possible to force certain interactions to be formed. This can be used as an effective filter in cases such as kinases where ligands typically form hydrogen bonds to the backbone in the hinge region. Hydrogen bond constraints may be used to require a hydrogen bond to be found between a particular protein group and a successfully docked ligand. Similarly, constraints may be applied to require metal ligation, atoms of a given chemistry to be found about a point in space relative to the receptor, or a hydrophobic group to be found in a volume of space relative to the receptor.

Ligand based similarity is a powerful tool that takes advantage of the adage, “similar structure, similar affinity.” Using a ligand similarity metric computed for each ligand against a set of probe molecules, the scoring function is altered to preferentially score ligands with high or low similarity. This can be of use in avoiding undesirable regions of chemical space.

In Basic Protocol 1 the grid generation process which creates grided potentials for the protein is outlined. In Basic Protocol 2 flexible ligand docking with Glide is presented. In Alternate Protocol 1 the grid generation process when constraints are to be used is outlined. In Alternate Protocol 2 the process of applying constraints in a flexible ligand docking experiment is presented. In Alternate Protocol 3 the process of applying molecular similarity in a flexible ligand docking experiment is presented. Support Protocol 1 outlines how to prepare ligand structures for flexible ligand docking with Glide. Support Protocol 2 indicates how to optimally prepare protein structures for use in Glide. Finally, Support Protocol 3 describes how to obtain and install the necessary software for all protocols.

Contributed by Matthew P. Repasky, Mee Shelley, and Richard A. Friesner
Current Protocols in Bioinformatics (2007) 8.12.1-8.12.36

STRATEGIC PLANNING

There are three basic steps in docking ligands flexibly with Glide: preparation of the protein and ligand structures to be used, a grid generation step, and a flexible ligand docking step. The protein preparation and ligand preparation steps are necessary to ensure that structures provided to Glide meet its minimum requirements. These two preparatory steps utilize Schrodinger applications other than Glide. If your input protein and ligand structures meet the requirements for use in Glide (see Support Protocol 1 and Support Protocol 2 for more details) than these two optional protocols (Support Protocols 1 and 2) may be skipped.

Prior to flexible ligand docking, Basic Protocol 1 must be completed to create a set of grids embodying the binding site into which Glide docks ligands. When using constraints to place restrictions on ligand docking, an alternative grid generation protocol, Alternate Protocol 1 (Grid Generation with Constraints protocol), is required to be completed prior to starting Alternate Protocol 2 (Flexible Ligand Docking with Constraints protocol). When using similarity to modulate the ranking of ligands in docking, Alternate Protocol 1 (Grid Generation with Constraints protocol) or Basic Protocol 1 (Grid Generation protocol) is required to have been completed prior to the start of Alternate Protocol 3 (Flexible Ligand Docking with Similarity protocol). See Figure 8.12.1 for a visual representation of protocol dependencies.

Glide experiments are most conveniently prepared using the Maestro graphical user interface (GUI). This article details the creation, execution, and analysis of Glide experiments within the Maestro GUI. The versions of Glide and Maestro referred to in this unit are 4.0 and 7.5, respectively. Similar versions of the software will behave in an analogous fashion, though details of experiment setup and execution may differ.

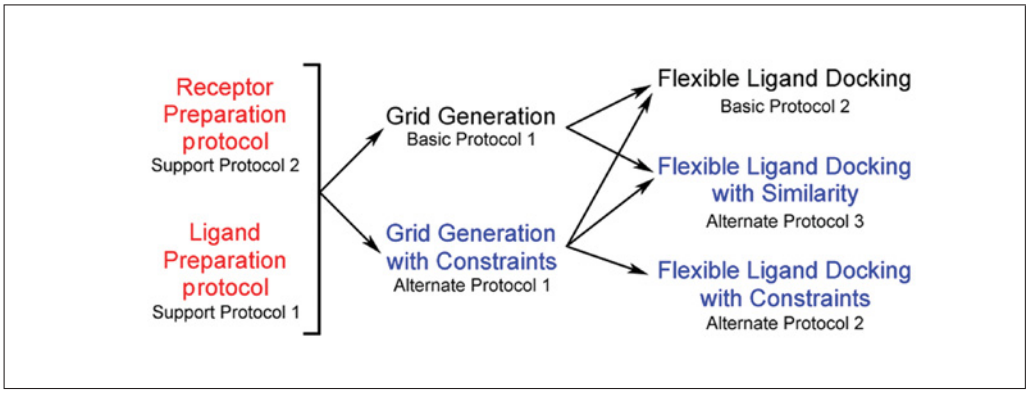


Figure 8.12.1 Figure of dependencies among protocols.

BASIC PROTOCOL 1

GRID GENERATION

In this protocol the protein into which ligands will be docked is analyzed and a set of grid files are generated that enable Glide to search for favorable interactions between the ligand and binding-site region. The shape and properties of the receptor are represented on a set of grids for positioning and scoring ligand poses. A pose is a complete specification of the ligand structure: conformation, position, and orientation relative to the receptor. Receptor grid generation requires a prepared protein which must be an all atom structure with appropriate bond orders and formal charges. The procedure for preparing a receptor structure is described in Support Protocol 2.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

Software

Glide and Maestro (see Support Protocol 3)

Files

A receptor structure in Maestro format prepared using Support Protocol 2.

1. Download and install Maestro and Glide on an accessible computer (see Support Protocol 3).
2. Start a Maestro session. At a prompt type `$SCHRODINGER/maestro &`.

   The Maestro window will appear. This window has a series of menu items across the top, a set of toolbar icons on the left-hand side and the Workspace where molecules are visualized, on the right-hand side.

Load prepared protein into Maestro

3. Import the prepared protein structure into Maestro. To import the protein structure file, open the Import panel by clicking the "Import structures" icon from the Maestro toolbar. In the Input panel the desired structure file is imported by entering its name directly in the entry box as an absolute or relative path, or by selecting from a list of files. Specify Maestro as the format of the protein structure file. Click the Import button to import the protein into Maestro where it will be viewed in the Workspace.

Set Glide options for grid generation

4. Open the Glide Receptor Grid Generation Panel by selecting the Receptor Grid Generation submenu under the Glide option of the Applications menu in the Maestro Panel. The Receptor Grid Generation panel has three tabs: Receptor, Site, and Constraints. The settings in the Receptor and Site tabs are described in this section, and the settings in the Constraints tab will be discussed in Alternate Protocol 1.
5. *Defining receptor in the Workspace:* (see Fig. 8.12.2 for the Receptor tab). If only the receptor is included in the Workspace and no ligand is present, skip this step. If the structure in the Workspace is a receptor with a ligand, identify the ligand molecule so that it can be excluded from receptor grid generation. Everything not identified as the ligand will be treated as part of the receptor. To select the ligand, ensure "Pick to identify ligand molecule" is selected, and pick an atom in the ligand molecule. If "Show markers" is selected, the identified ligand molecule is displayed with markers (displayed in dark green online).
6. *Setting the scaling factor for van der Waal's radii of nonpolar receptor atoms:* Glide does not allow for receptor flexibility in docking, but reducing van der Waal's radii of nonpolar atoms can mimic the effects of receptor flexibility to a certain degree. The "Scale by text box" entry box specifies the scaling factor. Van der Waal's (vdW) radii of nonpolar receptor atoms are multiplied by this value. The default value is 1.0, where no scaling is done. Scaling of vdW radii is performed only on nonpolar atoms, defined as those for which the absolute value of the partial atomic charge is less than or equal to the number in the text box. The default value is 0.25.

   Many experiments have demonstrated that using a scaling factor of 1.0 for the protein and 0.8 for ligand radii generally leads to optimal results with a wide variety of proteins. There are however, a few exceptions where a different combination of scaling factors, e.g., 0.8 for the protein and 1.0 for the ligand, leads to more favorable results.

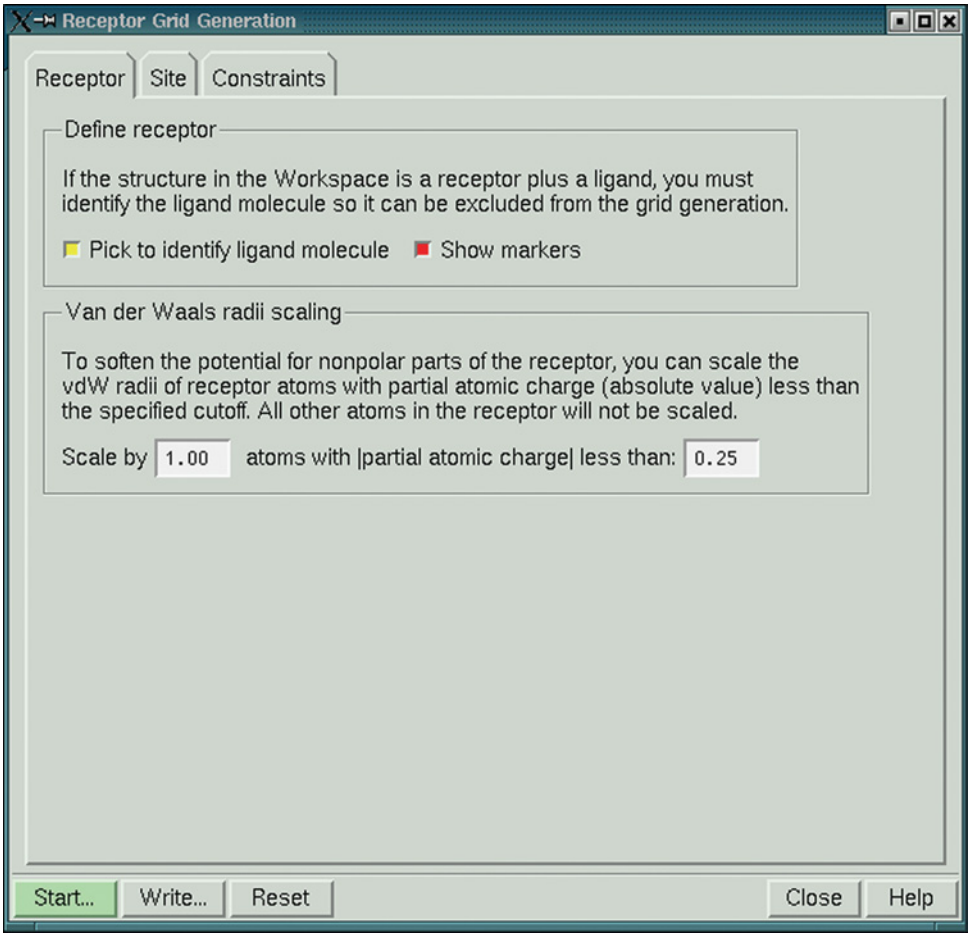


Figure 8.12.2 The Receptor tab of the Receptor Grid Generation panel.

7. *Specifying grid center and size:* The settings in the Site tab (Fig. 8.12.3) determine where the scoring grids are centered and how large they are. Glide uses two boxes to organize the calculation. The *enclosing box* defines the space in which grids are calculated. This is also the box within which all the ligand atoms must be contained. The *bounding box* defines the space within which the ligand center must be contained. The ligand center is defined as the midpoint of the line drawn between the two most widely separated atoms. The enclosing and bounding boxes share a common center.

   If the structure in the Workspace consists of a receptor and the ligand molecule which has been identified in the Receptor tab, Glide uses the position and size of the ligand to calculate a default center and a default size for the enclosing box. Upon opening the Site tab, the Workspace displays the center of the enclosing box as a set of coordinate axes (colored bright green onscreen), and the boundaries of the region as a cube (shown in purple onscreen).

   There are two other options for specifying the center of grids. The "Centroid of selected residues" option centers grids at the centroid of a set of user-selected residues. The "Specify Residue" button becomes available upon choosing this option. To select the residues, click "Specify Residue." The Active Site Residues dialog box opens. Use the picking controls to select residues that best define the binding site. Selected residues are marked in pink onscreen when the Active Site Residues dialog box is open. The center and the default boundaries of the enclosing box are updated

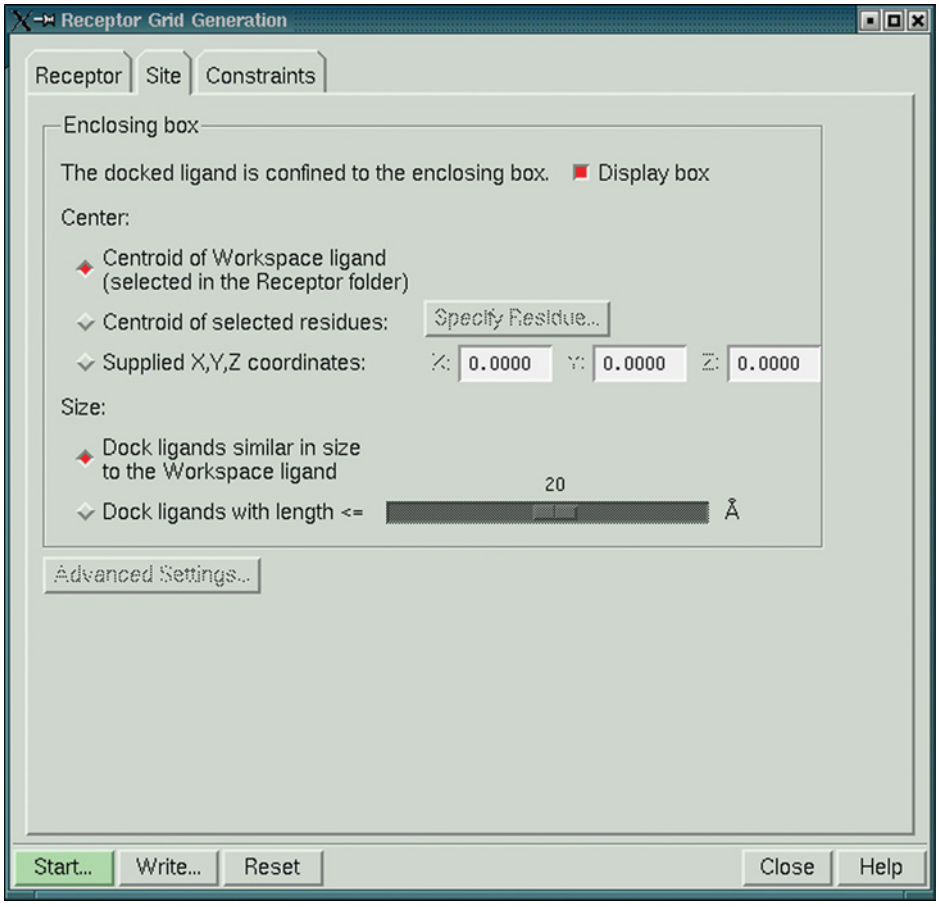


Figure 8.12.3 The Site tab of the Receptor Grid Generation panel.

and displayed after each residue pick. The list of selected residues is displayed in the dialog box. The "Supplied X, Y, Z coordinates" option centers grids based upon Cartesian coordinates. The X, Y, and Z text boxes become available when this option is chosen.

8. *Setting the box sizes:* The Size section provides options for specifying the size of the enclosing box. The default option is "Dock ligands similar in size to the Workspace ligand," which is suitable when the ligands to be docked are of similar size as the Workspace ligand. The second option, "Dock ligands with length <=," is useful when the Workspace structure does not contain a ligand. Use the slider to choose an appropriate maximum ligand length. The slider is set to 20 Å by default. To change the size of the bounding box or to use noncubic boxes, click the "Advanced Settings" button. The Site – Advanced Settings dialog box (Fig. 8.12.4) opens, and the bounding box is displayed as a cube outlined in bright green onscreen. The ligand center of each docked pose must remain within this box. The Size sliders can be used to increase or decrease the dimensions of each side of the box. The default is 10 Å on each side; the allowed range is 6 Å to 14 Å. Noncubic boxes can be useful when the binding site is spatially extended in one or two directions.

Generate Glide grids

9. *Submitting a job and monitoring progress:* Click the Start button to open the Receptor Grid Generation - Start dialog box. By default, grid files are written into the current working directory. If desired, an alternate directory can be specified by typing the

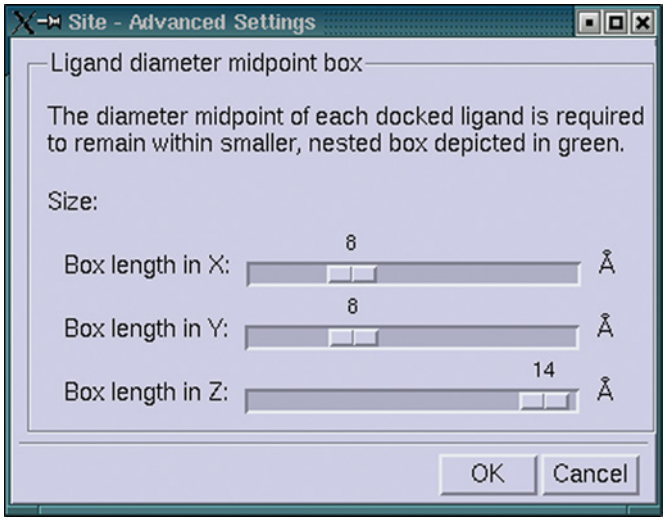


Figure 8.12.4 The Site – Advanced Settings dialog box.

path into the Directory for grid files text box, or browsing for the directory. Type a job name into the job Name text box. This becomes the base name of grid files. One can also specify the Host machine using the Host pull-down menu. To start the job, click Start.

BASIC PROTOCOL 2

FLEXIBLE LIGAND DOCKING

Glide performs flexible ligand docking into a rigid protein structure. There are two primary goals of flexible ligand docking: to accurately predict ligand poses and to rank ligands by predicted affinities to the protein. Here, a "pose" is the relative position and orientation of a ligand to a receptor including specification of the ligand conformation. A pose is defined as a complete specification of the ligand structure: conformation, position, and orientation relative to the receptor. Prior to running this protocol, a set of Glide grids must have been generated; see Basic Protocol 1 for further detail. Additionally, to obtain optimal results with Glide, it is very important to prepare ligand structures appropriately as outlined in Support Protocol 1.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

Software

Glide and Maestro (see Support Protocol 3))

Files

A file of ligand structures to be docked in Maestro or SD format, and a set of Glide grid files generated by completing Basic Protocol 1

1. Download and install Maestro and Glide on an accessible computer (see Support Protocol 1).

Set up flexible ligand docking

2. *Starting a Maestro session:* At a prompt type `$ SCHRODINGER/maestro &`.

   The Maestro window will appear. This window has a series of menu items across the top, a set of toolbar icons on the left-hand side and the Workspace where molecules are visualized, on the right-hand side.

3. *Opening the Glide Ligand Docking Panel:* From the Maestro Applications Menu, select the Ligand Docking submenu under the Glide option.

   The Ligand Docking panel will appear as shown in Figure 8.12.5. The window has a series of five tabs in which user adjustable options are grouped, Settings, Ligands, Constraints, Similarity, and Output. The Settings tab which is active in Figure 8.12.5 will be viewable the first time the Ligand Docking panel is accessed during a Maestro session. To view alternative tabs left-click on the tab title.

4. *Specifying the precalculated protein grids:* The grid files will have been precalculated in Basic Protocol 1. On the Settings tab of the Glide Ligand Docking Panel, the Receptor grid base name for precalculated protein grids may be entered directly in the entry box as an absolute or relative path, or may be selected from a list of files by clicking the Browse button.

5. *Selecting the docking precision, e.g., HTVS/SP/XP:* On the Settings tab of the Glide Ligand Docking Panel, click the appropriate radio button to select between HTVS (high-throughput virtual screening), SP (standard precision), or XP (extra precision) docking modes.

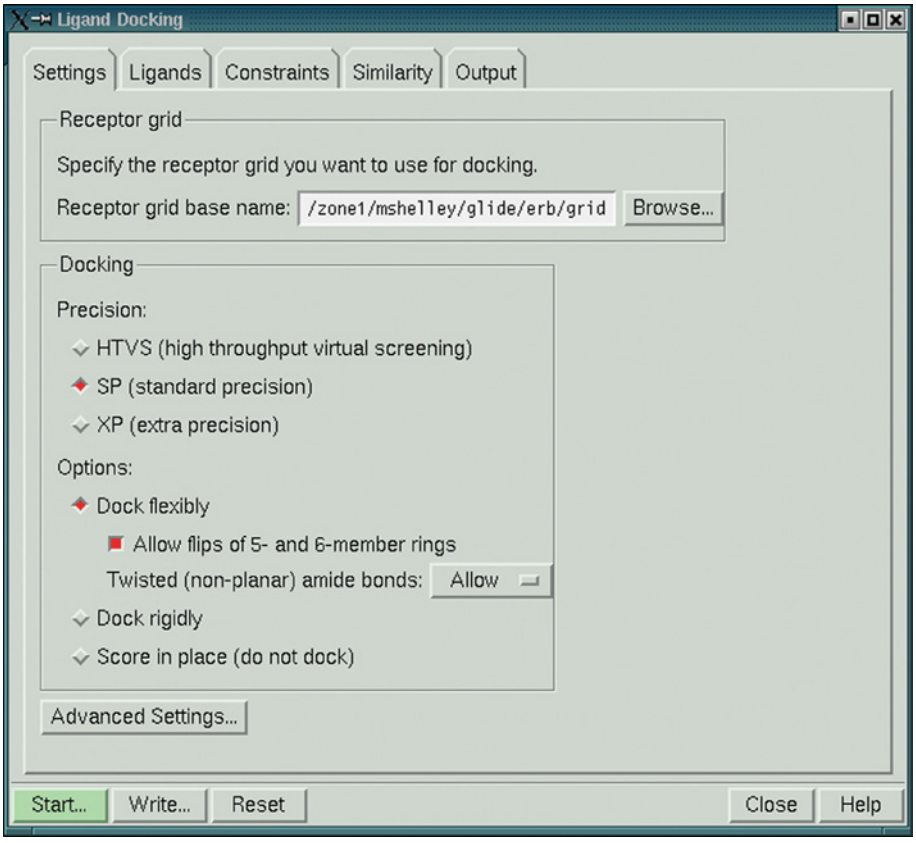


Figure 8.12.5 The Settings tab of the Glide ligand-docking panel.

This setting specifies the docking algorithm, scoring function, and extent of sampling that will be performed. High-throughput virtual screening docking is intended for the rapid screening of large ligand databases (~1 sec/ligand). While employing the same scoring function and sampling methodology as SP, HTVS has significantly more restricted sampling of poses than SP docking and cannot be used with constraints. Standard-precision docking is appropriate for accurate docking and database screening (~15 sec/ligand). Standard precision is the default. Extra-precision docking and scoring employs a harder scoring function that was optimized to minimize the number of false positives in screening. As such, extended sampling is required to effectively identify well-scoring poses (~10 min/ligand).

6. *Specifying flexible docking to be performed:* On the Settings tab of the Glide Ligand Docking Panel ensure the Dock flexibly radio button is selected.

   Flexible ligand docking is the default in Maestro and is the most common approach when performing ligand docking.

   Alternatives to flexible ligand docking include rigid docking and score in place which are selectable in Maestro. In rigid docking the input ligand conformer is docked without varying the ligand conformation. Score in place calculates the GlideScore for the input ligand geometry and does not make any attempt to alter the input pose.

7. *Selecting treatment of five- and six-membered rings:* On the Settings tab of the Glide Ligand Docking Panel click the selection box for "Allow flips of 5- and 6-member rings." This will enable Glide to sample 5- and 6-membered ring conformations during flexible docking.

   At present, conformation generation is limited to variation around acyclic torsion bonds, generation of conformations of nonaromatic 5- and 6-membered rings and generation of pyramidalizations at certain trigonal nitrogen centers, such as in sulfonamides. The user can control whether ring conformations are generated or not with this option which is selected by default.

8. *Selecting treatment of amides:* This has different options and meanings when applied to HTVS/SP and XP experiments. For HTVS/SP, on the Settings tab of the Glide Ligand Docking Panel select "Allow" or "Forbid" after "Twisted (non-planar) amide bonds." For XP select "Penalize" or "Do not penalize."

   For an XP docking experiment, the amide C-N torsion is always treated as rotatable. Selecting to Penalize twisted (non-planar) amides applies a penalty to the Emodel pose-selection function and to the XP GlideScore for any amide with an O=C-N-X dihedral angle (where X is any atom) further than 10° from planarity. The penalty makes it less likely for such nonplanar amide torsions to be found in the top-ranked poses for a ligand or to be ranked highly relative to other ligands. Selecting "Do not penalize" allows the torsion about the C-N amide bond to be freely rotated without penalty. In HTVS/SP modes, selecting "Allow" allows such amide torsions to be freely sampled while "Forbid" treats the torsion as nonrotatable, kept at its input conformation. To "Allow" such sampling is the SP default, while to "Penalize" is the XP default. The defaults have been chosen to generate optimal enrichments for a wide range of systems. They allow twisted amides a conformation occasionally necessary for ligands to adopt strained conformations that arise as a consequence of the rigid receptor approximation. When docking a ligand taken from a co-crystallized complex back into its native ligand geometry it is best to forbid twisted amides.

9. *Specifying ligands to be docked:* On the Ligands tab (see Fig. 8.12.6) of the Glide Ligand Docking Panel, specify the prepared ligands to be docked. The ligands may come from a file which can be specified by entering an absolute or relative path or selected from a list of files by clicking on the Browse button. Alternatively, a single ligand may be docked from the Maestro Workspace or a set of ligands selected in the Maestro Project Table may be docked.

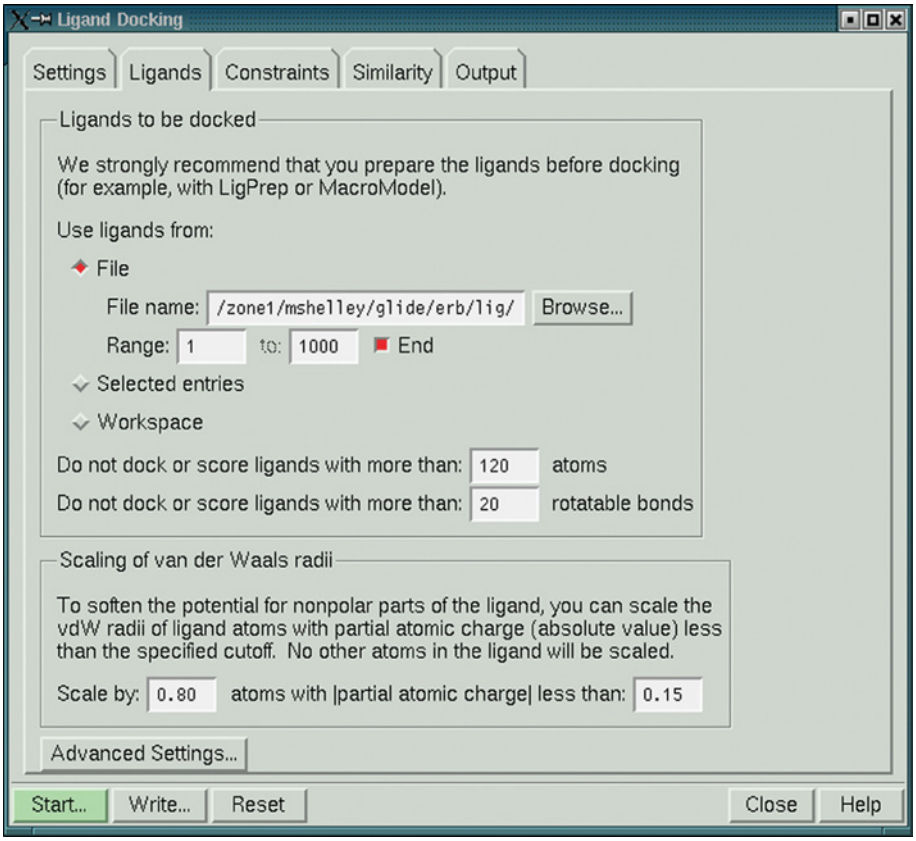


Figure 8.12.6 The Ligands tab of the Glide ligand panel.

10. *Specifying ligand vdW scaling of nonpolar ligand atoms:* In the Ligands tab of the Glide Ligand Docking Panel, specify atoms that will have scaled van der Waal's radii. Atoms are selected to be scaled by their absolute partial atomic charges being less than a given value (0.15 by default) with a default scaling factor of 0.80.

    Scaling of nonpolar ligand vdW radii is essential due to the rigid receptor approximation utilized by Glide. A value of 0.8 has been used almost exclusively, though for a small number of cases it has been found beneficial to scale the protein vdW of nonpolar atoms by 0.8 and to not scale nonpolar ligand vdW radii.

Submit and monitor a Glide flexible ligand docking experiment

In Maestro click the Start button on the Glide Ligand Docking Panel to display the Ligand Docking – Start panel as shown in Figure 8.12.7. In this panel the job name that uniquely identifies the job to be run, the host the job is to be run from, and job distribution options must be specified. The job name should be a single word without special characters ([!@#$%^&*]).

11. *Submitting a flexible ligand docking experiment for execution:* The host is selected from a list of hosts specified in the `schrodinger.hosts` file (see Support Protocol 3). Docking jobs may be split into a number of subjobs that are to be distributed over a number of processors.

12. *Monitoring flexible ligand docking experiment:* Progress of the Glide ligand docking experiment is monitored in the Monitor Panel of Maestro. This panel, shown in Figure 8.12.8, is displayed automatically when a Glide ligand docking experiment is

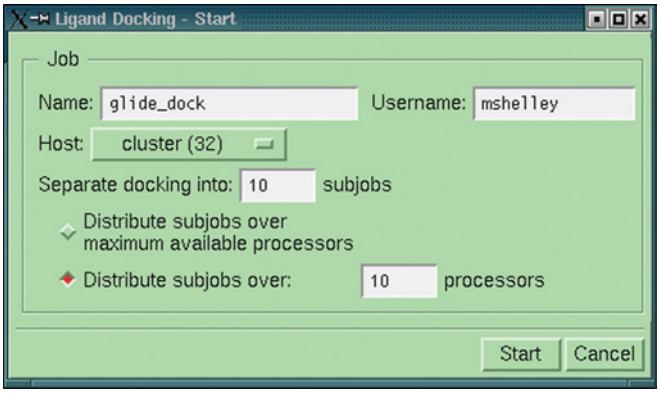


Figure 8.12.7 The Ligand Docking - Start menu.

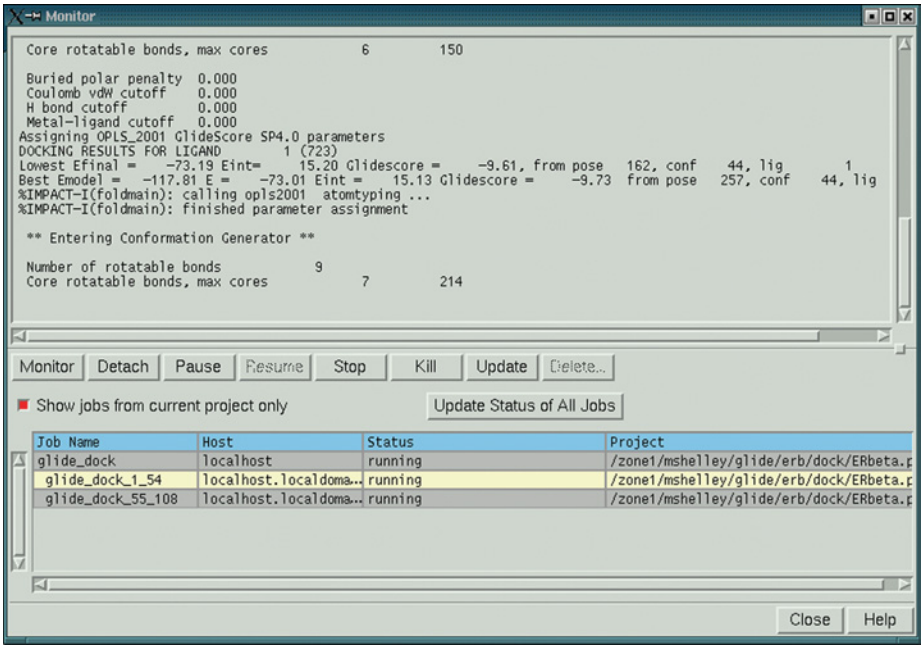


Figure 8.12.8 The Maestro Monitor panel monitoring a Glide docking job.

started. Alternatively, the Glide Monitor panel can be opened by selecting Monitor in the Maestro Applications menu.

In the Monitor panel Glide processes may be monitored by following the log file information that is displayed. When necessary, processes may be killed and/or paused from this panel.

Analysis of Glide results in Maestro

13. *Importing structural results of the Glide flexible ligand docking experiment into Maestro:* A structure file of poses ranked by GlideScore is output by Glide to either a library file (named as `jobname_lib.mae`) or a pose viewer file (named as `jobname_pv.mae`). The pose viewer file contains a copy of the protein structure into which ligands were docked while the library file contains only the docked ligand poses. To import a pose viewer or library file, open the Import panel by clicking the "Import structures" icon (image of an arrow pointing from a file folder to a spreadsheet) from the Maestro toolbar. Pick Maestro as the format, and specify

the desired structure file by selecting from a list of files or by entering its name directly in the entry box as an absolute or relative path. Click the Import button to import the structures into Maestro Project Table. The first structure will be viewed in the Workspace and all ligand properties including the GlideScore will be displayed in the Project Table. If a library file has been imported, import also the receptor structure so that it can be used for examining binding modes. Basic Protocol 1 produces a copy of the receptor structure in `jobname.mae` which can be used here.

14. *Analyzing poses by GlideScore:* Docked ligands are displayed in the Project Table sorted in ascending order by GlideScore, e.g., ligands with the most negative GlideScore are found at the top of the Project Table.

    The top-ranked compounds from Glide are those with the most negative GlideScores. Both XP and SP GlideScores approximately cover the range of experimental binding affinities in kcal/mol. The typical range of SP GlideScores for known active ligands is from −6 to −14 kcal/mol. XP GlideScores for known actives are generally more negative and typically range from −7 to −18 kcal/mol. If multiple poses per-ligand were retained, the poses of a given ligand should be evaluated using the Emodel pose selection function. Emodel values are displayed in the Project Table for all docked ligands.

15. *Analyzing poses by protein-ligand interactions:* The receptor structure can be locked in the Workspace, while stepping through ligands to examine binding modes. Select the receptor entry in the Project Table with a left-mouse click on its row, and lock it in the Workspace by choosing "Fix" (Project Table → Entry → Fix). Select a set of ligand pose entries: left-mouse click on the first ligand row and then Shift + left-mouse click on the last ligand row. The selected entries appear highlighted in the Project Table. Step through selected ligand poses using the ePlayer and examine protein-ligand interactions. The ePlayer can be played forward, backward, or stopped at any point, and the speed of automatic display can be adjusted. It can also be used in Step Mode to manually go to the next or the previous entry or to go to the first (start) or the last (end) entry in the selection. See Maestro online Help or the user manual for details. While stepping through ligand poses, H-bonds to the receptor, or close contacts with the receptor, can be visualized. Display one of the selected ligand poses with a left-mouse click on its "In" column. The Workspace will now contain two entries: the receptor and one ligand pose. Click on the "Display H-bonds" icon (image of a dashed line extending from the letter "H") in the Maestro toolbar, ensure that "Inter H-bonds" has been selected, and then click on any atom of the ligand in the Workspace. All H-bonds between the ligand pose and the receptor will appear in yellow dashed lines. Stepping through ligand poses using the ePlayer will show the H-bonds between the currently displayed ligand pose and the receptor. Similarly, close contacts can be visualized between each ligand pose and the receptor. Open the Measurement panel (Tools → Measurements) and click on the "Contacts" tab. Define the receptor entry as "Atom set 1," by picking any atom of the receptor after selecting "Entries" as the pick state for Atom Set 1. Define the ligand molecule as "Atom set 2," by picking any atom of the currently displayed ligand pose after choosing "Molecules" as the pick state for Atom set 2. Visualize contacts between the currently displayed ligand pose and the receptor, by stepping through ligand poses using the ePlayer.

    The pose viewer Maestro formatted file lists the protein structure first followed by all successfully docked ligand structures in the same frame of reference. A pose viewer formatted file may be created by appending the Glide output docked ligand structures in Maestro format to the end of the protein structure file in Maestro format.

GRID GENERATION WITH CONSTRAINTS

Glide constraints are protein-ligand interactions that the user believes to be important to the binding mode. Glide is designed to work well without any docking constraints, but using docking constraints can be useful for screening out ligands or poses that do not meet the user-specified criteria. Glide provides a powerful and flexible mechanism to define and apply several types of constraints. A positional constraint is a requirement that ligand atoms occupy a certain region of space relative to the receptor. A hydrogen bond constraint is a requirement that a particular receptor-ligand hydrogen bond be formed. A metal constraint is a requirement that a particular metal-ligand interaction be present. A hydrophobic constraint is a requirement that hydrophobic heavy atoms of the ligand occupy a specified hydrophobic region in the binding site.

Any Glide constraints that may be used in docking must be defined in advance when the receptor grids are generated. In the docking stage, the user can select all or a subset of constraints to apply.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

Software

Glide and Maestro (see Support Protocol 3)

Files

A receptor structure in Maestro format prepared using Support Protocol 2

1. Download and install Maestro and Glide on an accessible computer (see Support Protocol 3).
2. *Setting up a grid generation experiment without constraints:* Follow steps 2 to 7 of Basic Protocol 1 to prepare a grid generation experiment without constraints.

Set constraints

3. *Specifying constraints in the Constraints tab of the Glide Receptor Grid Generation Panel*: The Constraints tab of the Receptor Grid Generation panel is used to define Glide constraints. It has three subtabs, Positional, H-bond/Metal, and Hydrophobic, for specifying different types of constraints (Fig. 8.12.9). Up to ten constraints can be defined in an experiment. In the subsequent docking stage (Basic Protocol 2), up to four constraints may be required to be satisfied.

Setting positional constraints

Positional constraints define a region of space relative to the protein that must contain a particular type of ligand atom(s). Positional constraints can be used to require interactions between any kind of protein and ligand atom. In docking set up, SMARTS patterns will define what type of ligand atoms can satisfy each positional constraint.

To add a positional constraint, click New on the Constraints tab of the Receptor Grid Generation panel. This button opens the New Position dialog box (Fig. 8.12.10). If desired, the name of the positional constraint and its radius can be specified. The standard picking controls can be used to select atoms to define a position. The position is the centroid of the selected atoms, and must lie inside the enclosing box. While picking is in progress, the constraint is marked with a gray sphere. After selecting a desired set of atoms, click OK. Then, the constraint is added to the Positions table, and the sphere turns

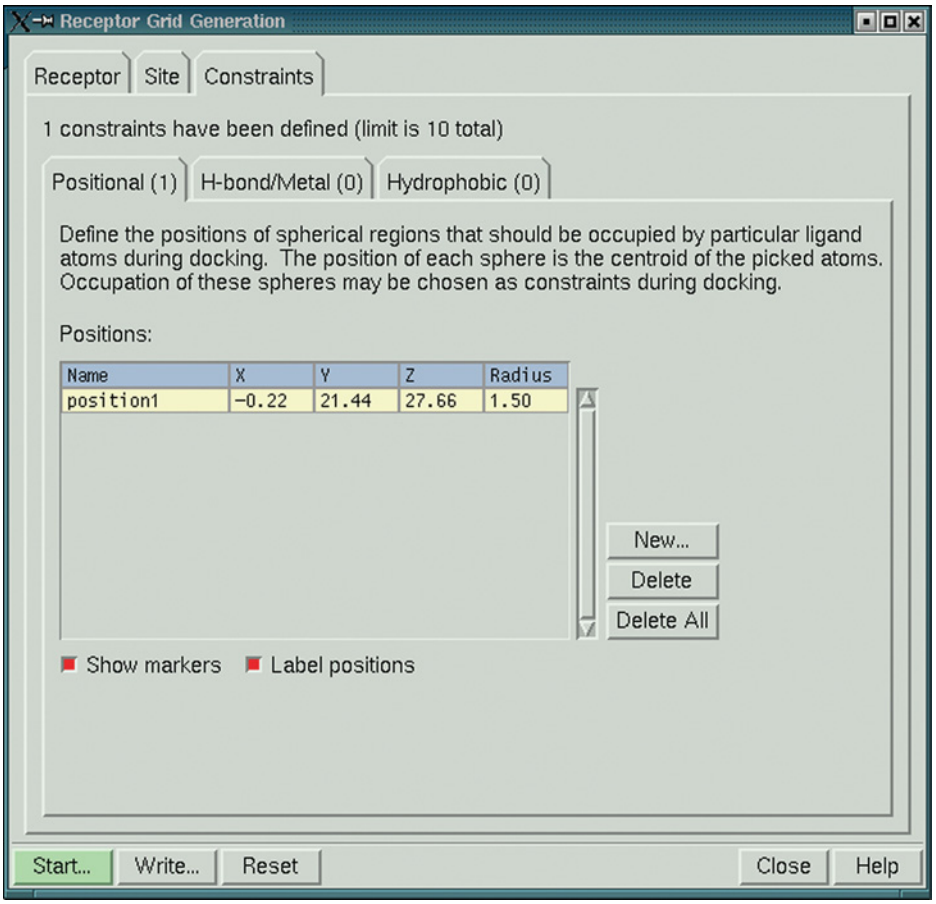


Figure 8.12.9 The Constraints tab of the Receptor Grid Generation panel showing the Positional subtab.

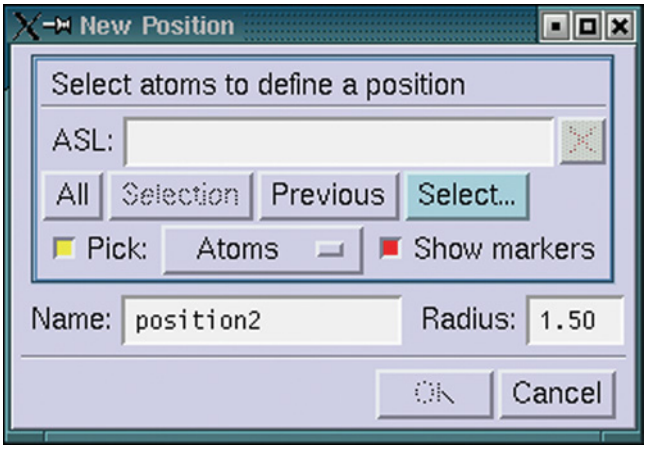


Figure 8.12.10 The New Position dialog box.

yellow. To delete a constraint, select it in the Positions table and click Delete; to delete all constraints, click Delete all.

The Positions table displays the name, coordinates, and radius of the constraint sphere for each constraint. The name, coordinates, and radius of the sphere can be edited, by clicking in the table cell and changing the value.

The selected constraint is marked by a yellow sphere. The other positional constraints are marked by red spheres. If Show markers is selected, selecting the Label positions option displays the name of the constraint in the Workspace. Labels and constraints are colored identically.

Setting hydrogen bond and metal constraints

For hydrogen-bonding interactions, the receptor atom must be a polar hydrogen (including the thiol hydrogen in cysteine), nitrogen, or oxygen. If an atom with one or more symmetry-equivalent atoms in its functional group is chosen, the symmetry-equivalent atoms will be selected as well, and collectively count as one constraint. For example, if a constraint is created by picking one oxygen atom of a carboxylate group, Glide includes the other oxygen atom in the same constraint. A ligand interaction with either oxygen atom will satisfy that single constraint. Figure 8.12.11 shows the H-bond/Metal subtab of the Constraints tab.

For metal-ligand interaction constraints, the receptor atom must be a metal ion. Metal-ligand constraints can also include restrictions on the formal charges of the interacting ligand atoms. Such requirements are added during the set up of flexible ligand docking experiments.

The criteria to define a hydrogen bond or metal-ligand interaction is set by default to H-acceptor distances of 1.2 Å to 2.5 Å, donor angles >90°, and acceptor angles >60°.

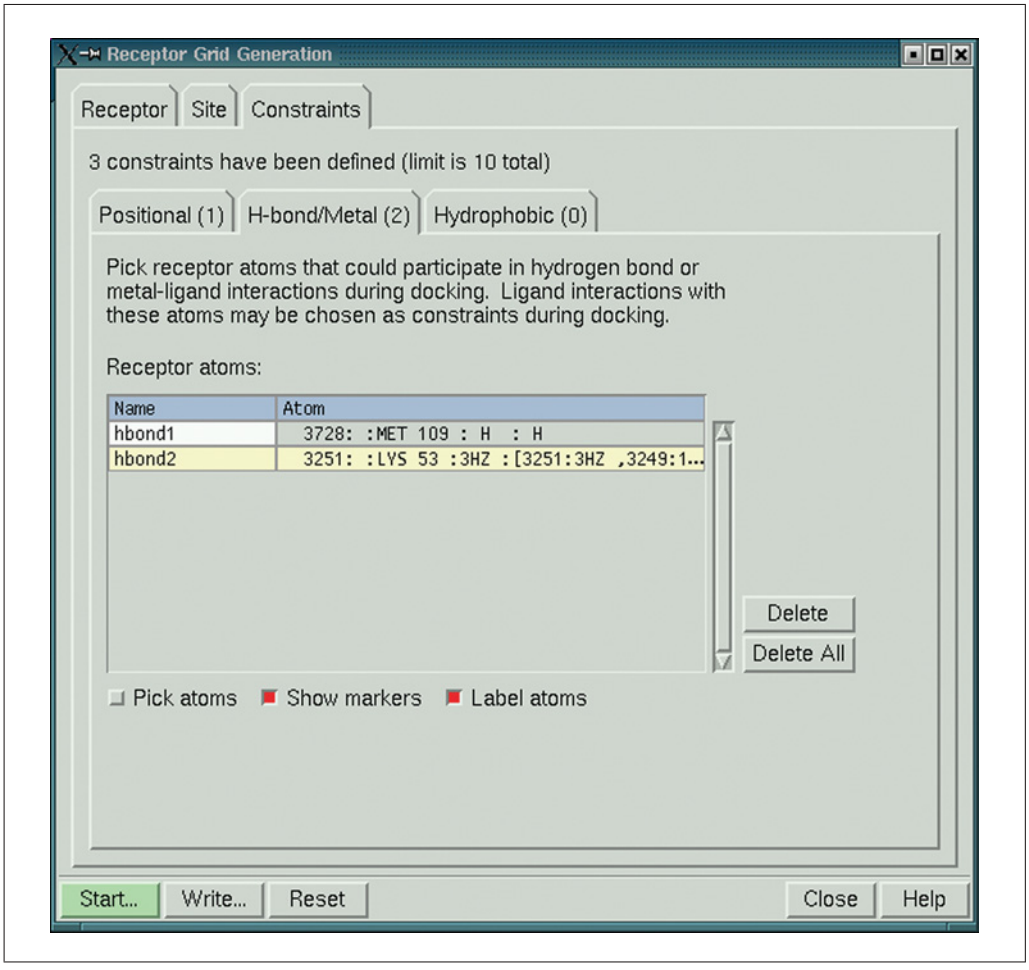


Figure 8.12.11 The Constraints tab of the Receptor Grid Generation panel showing the H-bond/Metal subtab.

Receptor atoms selected for constraints must lie inside the enclosing box (displayed in purple).

To display hydrogen bonds in the Workspace, choose Inter from the Display H-bonds button menu and click on a ligand atom. The hydrogen bonds between the ligand and the receptor are displayed. This should make it easier to locate relevant receptor atoms.

To set hydrogen bond or metal constraints, ensure that "Pick atoms" is selected in the H-bond/Metal constraints tab, and pick the desired atoms in the Workspace. If "Show markers" is selected, a red cross and red padlock appear next to each atom picked, and the constraint name is displayed. If the picked atom is one of a set of symmetry-equivalent atoms, all the atoms in the set are marked.

The selected atoms appear in the Receptor atoms list in the format:

Atom number : Chain name : Residue name Residue number : PDB atom name : symmetry-set

If the picked atom is part of a symmetry equivalent set, its identification is followed by square brackets enclosing the number and name of each atom in the set, separated by commas.

For example,

2325:H:ASP 189 : OD2: [2325: OD2,2324: OD1]

where the oxygen atom (atom number 2325, PDB atom name "OD2") in the carboxylate group of ASP 189 in Chain H has been selected. This atom is equivalent to the other oxygen in the carboxylate group (atom number 2324, PDB atom name "OD1").

2600:H:GLY 216 : H : H

where the H atom (atom number 2600, PDB atom name "H") in the backbone NH of GLY 216 in Chain H has been selected. There is no symmetry-equivalent atom to this H atom.

To delete a single hydrogen bond or metal constraint, select it in the list and click Delete. Delete All can be used to delete all the listed constraints.

Setting hydrophobic constraints

A hydrophobic constraint requires that a hydrophobic region of the receptor be occupied by one or more hydrophobic heavy atoms in the ligand. The possible hydrophobic regions are identified from a hydrophobic map of the receptor site. One or more of the hydrophobic regions can be selected as a constraint, and the size of the region that must be occupied can be determined by adding or deleting cubic volumes ("cells") to the region. When setting up a docking job, one or more of these regions can be selected and how many atoms must occupy each region can be specified.

The Hydrophobic subtab (Fig. 8.12.12) has the Setup section for generating the hydrophobic map and the Define regions section for selecting the hydrophobic constraint regions.

Generating the hydrophobic map

To generate the hydrophobic map of a binding site, click Locate Hydrophobic Cells in the Setup section to start a job to generate the hydrophobic map of the receptor site. The gray octagon at the upper right of the panel turns green and spins; when the job finishes it stops spinning and turns gray again.

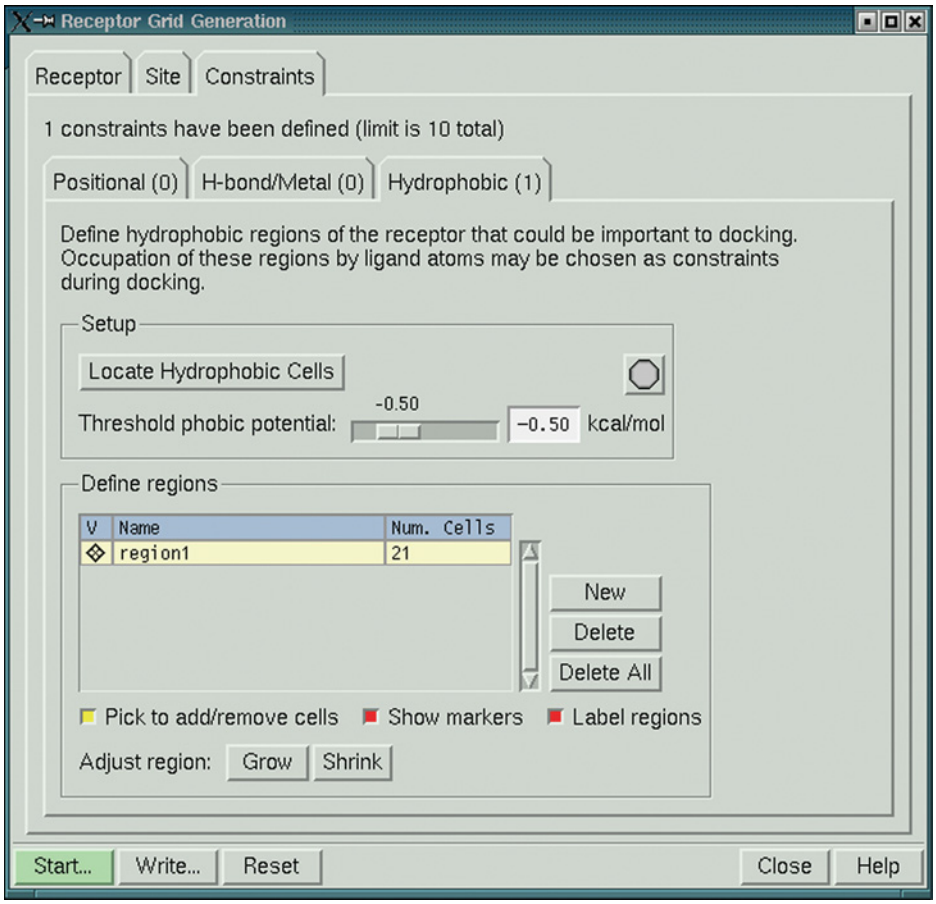


Figure 8.12.12 The Constraints tab of the Receptor Grid Generation panel showing the Hydrophobic subtab.

Translucent gray cubes that represent the hydrophobic regions around the binding site should be displayed when the job finishes. The threshold corresponds to the isovalue contour at which the hydrophobic map is displayed, and has a default value of −0.5.

Defining hydrophobic constraint regions

After the hydrophobic map has been generated, the table in the Define regions section contains a single default constraint region, region1, with 0 in the Num. Cells column (equivalent to no constraint). A region is defined by a set of hydrophobic cells. The cells in the region do not have to be contiguous. To add individual cells to a region, select Pick to add/remove cells and click on cells in the Workspace. The cursor has the label C, to indicate that cell picking is active. The cell color changes to red when it is added (Fig. 8.12.13). The last picked cell in each region is outlined in yellow. To remove a cell that has already been added, click on the cell in the Workspace. Its color changes to gray.

The Grow and Shrink buttons can be used for adding or removing cells layer by layer. Clicking Grow adds the cells that are nearest neighbors to the most recently selected cell (outlined in yellow). Clicking Grow a second time selects a layer of cells adjacent to those most recently selected. To remove a layer of cells, click Shrink. Each click removes one layer.

To add a new hydrophobic constraint region, click New. Then, a new row is added to the table. To delete a constraint region, click Delete. The name of a region can be edited, and the visualization markers can be turned off by deselecting the box in the V column.

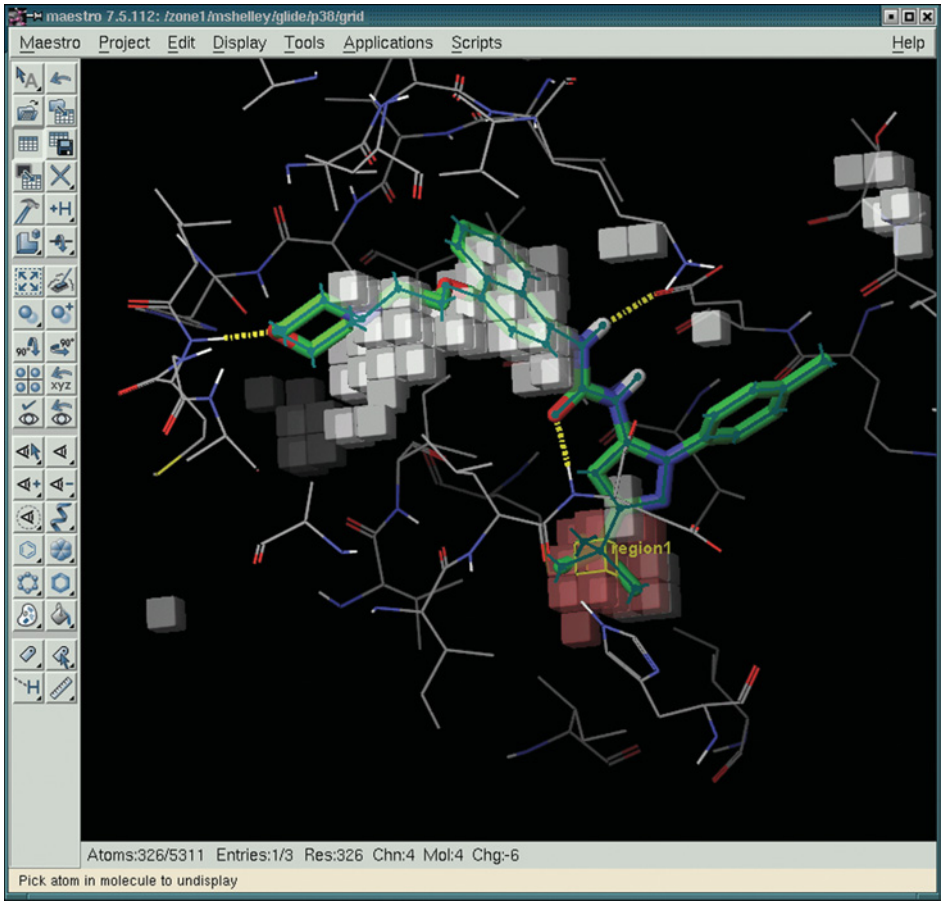


Figure 8.12.13 Hydrophobic constraint regions displayed in the Maestro Workspace. For color version of this figure see *http://www.currentprotocols.com*.

When the Label regions option is selected, regions are labeled with their name in the Workspace.

Generate Glide grids

4. *Submitting a grid generation with constraints experiment for execution:* In Maestro click the Start button on the Glide Receptor Generation Panel to display the Grid Generation – Start panel. In this panel the job name that uniquely identifies the job to be run, the directory in which the job will be run, and the host the job is to be run from must be specified. The job name should be a single word without special characters ([!@#$%^&*]). The host is selected from a list of hosts specified in the `schrodinger.hosts` file (see Support Protocol 3).

5. *Monitoring grid generations with constraints experiment:* Progress of the Glide ligand docking experiment is monitored in the Monitor Panel of Maestro. This panel, shown in Figure 8.12.8, is displayed automatically when a Glide ligand docking experiment is started. Alternatively, the Glide Monitor panel can be opened by selecting Monitor in the Maestro Applications menu.

   In the Monitor panel Glide processes may be monitored by following the log file information that is displayed. Processes may be killed and/or paused from this panel.

ALTERNATE PROTOCOL 2

FLEXIBLE LIGAND DOCKING WITH CONSTRAINTS

It is sometimes desirable to enforce specific ligand-protein interactions or to limit specific chemical functionality to a defined region of space relative to the receptor. A hydrogen bond or metal ligation constraint consists of the protein atom involved in the specific protein-ligand interaction. For a ligand to satisfy the defined constraint it must exhibit the corresponding functionality to form the protein-ligand interaction which may be located in a suitable pose such that the interaction is formed. For instance, if a hydrogen bond constraint was defined including a protein backbone carbonyl (acceptor) Glide will require a ligand hydrogen bond donor to interact with the carbonyl for that pose to have satisfied the defined constraint. It is not necessary for users to define what constitutes a hydrogen bond acceptor/donor or metal ligating group as this has been defined within Glide.

A positional constraint consists of a point in space relative to the receptor and a radius from this point within which some user-specified chemical functionality must be found for a pose to satisfy the defined constraint. The chemical functionality which must be found with the sphere is defined using SMARTS or SMILES patterns. Glide provides a series of predefined patterns for common functionality such as hydrophobic atoms and donor/acceptor atoms. In addition, any valid pattern or set of patterns may be used.

A hydrophobic constraint consists of a set of boxes placed relative to the receptor within which a user-specified number of hydrophobic atoms must be found for a pose to satisfy the defined constraint.

Constraints must have been previously defined during the grid generation protocol (see Alternate Protocol 1) to be applied during flexible ligand docking. Constraints cannot be used in HTVS mode. In addition to defining and applying a single constraint, optional constraints provide a mechanism in Glide to apply combinations of constraints using Boolean logic. Up to four groups of constraints may be defined in which all or a specified number of the constraints must be simultaneously satisfied. These groups may then be simultaneously used where each group is required to be satisfied in order for a pose to be said to have satisfied constraints, providing a mechanism to create and apply very sophisticated combinations of constraints.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

Software

Glide and Maestro (see Support Protocol 3)

Files

A file of ligand structures to be docked in Maestro or SD format, and a set of Glide grid files generated by completing Alternate Protocol 1

1. Download and install Maestro and Glide on an accessible computer (see Support Protocol 3).

Set up flexible ligand docking with constraints in Maestro

2. *Setting up a flexible ligand docking experiment without constraints:* Following steps 2 to 10 in Basic Protocol 1, set up and prepare a ligand or series of ligands for flexible ligand docking.

3. *Opening the Constraints tab of the Glide Ligand Docking Panel:* From the Maestro Applications Menu select the Ligand Docking submenu under the Glide option. Select the constraints tab by right clicking the tab.

   The Constraints tab of the Ligand Docking constraints panel will appear as shown in Figure 8.12.14.

4. *Optional: Display Receptor to view available constraints:* In the Constraints tab of the Glide Ligand Docking panel, click on the Display Receptor button to replace the Maestro Workspace with a view of the protein in which constraints have been annotated. Possible positional constraint sites are viewed as spheres, possible hydrophobic constraints are viewed as a set of cubes, and possible hydrogen bonding sites are stars with padlocks on the relevant protein atoms.

5. *Specifying constraints that will constitute a constraint group (group 1) and their relationship:* Up to ten prepared constraints can be defined as being part of a constraint group and up to four constraint groups can be simultaneously applied in a flexible ligand docking with constraints experiment. To apply constraints in flexible ligand docking, at least one constraint group is required to be defined with at least one constraint. In most cases only a single constraint group will need to be defined. Each defined constraint group must be satisfied for a ligand to satisfy constraints. A constraint group is satisfied by having either all or a specified number of defined constraints in that group satisfied. This enables sets of constraints to be combined with Boolean logic. To add a constraint to a group, click the button corresponding to that constraint in the Use column of the Available constraints table of the Group 1 tab in the Constraints tab of the Ligand Docking Panel.

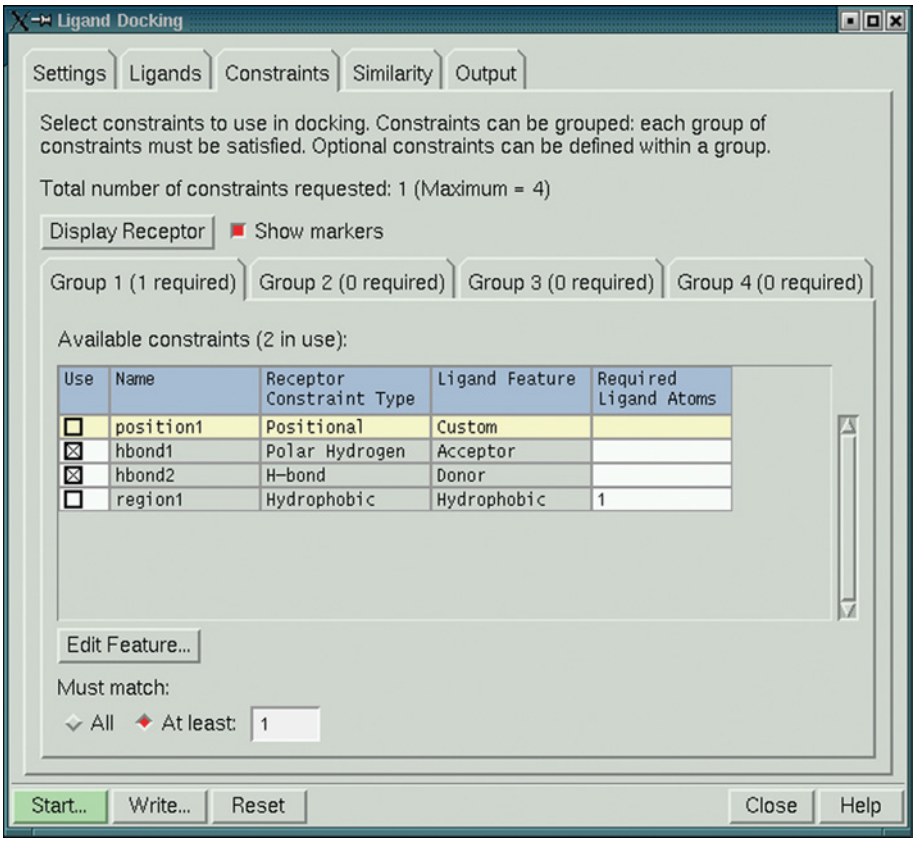


Figure 8.12.14 The Constraints tab of the Glide Ligand Docking panel.

6. *Setting up ligand criteria for specified hydrophobic constraints in group 1:* This is only required if a hydrophobic constraint has been used in the current group. For hydrophobic constraints the number of ligand atoms required to be found within the boxes that comprise the constraint should be set by changing the value in the Required Ligand Atoms column of the Available constraints table for the used hydrophobic constraint.

7. *Setting up desired chemical functionality to form protein-ligand interactions for constraints in constraint group 1:* This is only required if a positional constraint has been used in the current group. For positional constraints this specifies the chemical functionality that must be found within the sphere defined by the constraint for it to be satisfied. For hydrogen bond or metal constraints this specifies the chemical functionality that must form a hydrogen bond or ligate with the protein sites. For hydrophobic constraints this specifies the chemical functionality that must be found within the boxes that comprise the constraint. This is specified by selecting an active constraint in the Available constraints table and clicking on the Edit Feature button to display the Edit Features Panel.

   Ligand features are identified by a collection of SMARTS patterns that define a feature type. There are six feature types: Acceptor, Charged Acceptor, Neutral Acceptor, Donor, Hydrophobic, and Custom. The feature definitions for these six types form a feature set, which can be imported and exported. Each constraint can have its own feature definition, so one can have a different definition of a given feature type for each constraint. However, the same feature definition from the same set is used for a given constraint in all groups. For each feature definition one can add patterns, edit and delete custom patterns, and define patterns for exclusion for functional groups. Feature sets can be imported and exported. The patterns that define a feature set are displayed in the Pattern list table of the Edit Feature Panel. If the patterns in a given feature do not cover all the functional groups desired in the definition, additional patterns can be added. To add a new SMARTS pattern, click the table row above which the pattern is to be inserted then click the New button. In the New Pattern dialog box one can provide a SMARTS pattern and define the atoms that must satisfy the constraint. There are two ways to provide a SMARTS string. The first is to type the string into the SMARTS pattern text box. The second is to select atoms in the Maestro Workspace, then click the Get from selection button. Maestro generates a SMARTS string for the selected atoms and places it into the SMARTS pattern text box where it may be further edited. Atoms in the ligand that must satisfy the constraint are specified in terms of the SMARTS pattern. For positional, metal, and hydrogen bond constraints only one atom should be specified. For hydrophobic constraints, the nonhydrogen atoms of the hydrophobic group should be specified. To specify the atoms, enter the atom numbers as a comma-separated list in the Numbers text box: atom 1 is the first atom in the SMARTS pattern and so on. To add the new pattern to the feature, click the OK button. To edit a pattern, select the table row for the pattern, then click the Edit button. In the Edit Pattern dialog box modify the SMARTS pattern or obtain a new pattern from the Workspace selection and change atoms in that pattern that must satisfy the constraint. To delete a pattern select the table row for the pattern in the Edit Feature Panel and click the Delete button.

   For positional constraints a SMARTS pattern must be defined. This is typically done by adding the desired SMARTS pattern as a Custom constraint. For hydrogen bond, metal, and hydrophobic constraints it is rarely necessary to alter the default feature definitions.

8. *Optional: Defining additional constraint groups:* In step 7 individual constraints were combined to form a single constraint group. To enable the application of very sophisticated combinations of constraints, multiple constraint groups may be

necessary. When multiple constraint groups are defined, each constraint group must be satisfied in order for a ligand pose to satisfy the user-defined constraints.

For the most common uses of constraints it is not necessary to define multiple constraint groups. For instance, to require a single hydrogen bond to be formed between ligands and the protein the constraint should be set up as the hydrogen bond constraint in group 1 with All constraints required to match. If there are two hydrogen-bonds possible, of which at least one must be formed, the two hydrogen bonds should be used in constraint group 1 with at least 1 constraint required to match. If there are two sets of two hydrogen bonds, where in each set at least one of the hydrogen bonds must be met, then two constraint groups should be defined with two hydrogen bond constraints in each with at least one required constraint defined in both constraint groups.

Submit and monitor a Glide flexible ligand docking experiment

9. *Submitting a flexible ligand docking experiment with constraints for execution:* In Maestro click the Start button on the Glide Ligand Docking Panel to display the Ligand Docking – Start panel as shown in Figure 8.12.7. In this panel the job name that uniquely identifies the job to be run, the host the job is to be run from, and job distribution options must be specified. The job name should be a single word without special characters ([!@#$%^&*]). The host is selected from a list of hosts specified in the `schrodinger.hosts` file as described in Support Protocol 3. Docking jobs may be split into a number of subjobs that may be distributed over a number of processors.

10. *Monitoring the flexible ligand docking with constraints experiment:* Progress of the Glide ligand docking experiment can be monitored in the Monitor Panel of Maestro. This panel, shown in Figure 8.12.8, is displayed automatically when a Glide ligand docking experiment is started. Alternatively, the Glide Monitor panel can be opened by selecting Monitor in the Maestro Applications menu.

Analysis of Glide results

11. *Analyzing poses by GlideScore and protein-ligand interactions:* Following steps 14 to 15 in Basic Protocol 2, analyze the results of the flexible ligand docking with constraints experiment.

FLEXIBLE LIGAND DOCKING WITH SIMILARITY

ALTERNATE PROTOCOL 3

Similarity algorithms provide a mechanism for quantifying how alike or unlike two molecules are. Various methods of calculating similarities have been used as additional criterion for selecting likely candidates—molecules structurally similar to known actives—from large molecular databases. Glide provides users the ability to calculate molecular similarities between a set of probe molecules and each molecule to be docked. The maximum similarity found to any of the probe molecules can be used to modulate the SP or XP GlideScore to reward or penalize ligands that show high similarity to the probe molecules.

Glide uses an atom-pair similarity scoring algorithm. In atom-pair similarity, the two molecules being compared are first processed to generate sets of atom pairs. Each non-hydrogen atom is represented by a similarity type based on the connectivity, bond orders, and formal charges of the molecule. For each pair of similarity types, the shortest bond path (the bond path with the smallest number of connections) is determined. The unique combination of "type(atom A) + connectivity distance + type(atom B)" defines one atom pair. All atom pairs for a given molecule constitute the atom pair list for that molecule. The similarity between two molecules is a function of the number of atom pairs that appear in both lists. The similarity function is normalized so that the result is a number between 0.0 (no atom pairs in common) and 1.0 (identical atom pair lists).

The maximum similarity to any probe molecule is used to modulate the GlideScore for a given pose of a ligand resulting in the ligand being energetically rewarded or penalized for having high/low similarity to the probe molecules. By default, no similarity scoring is performed.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

Software

Glide and Maestro (see Support Protocol 3)

Files

A file of ligand structures to be docked in Maestro or SD format, and a set of Glide grid files generated by completing Alternate Protocol 1. A structure file of probe molecules must be provided in Maestro or SD format. These probe molecules should be prepared analogously to the ligands to be docked though only a single tautomerization and ionization state for each ligand should be present.

1. Download and install Maestro and Glide on an accessible computer (see Support Protocol 3).

Set up of flexible ligand docking with similarity in Maestro

2. *Setting up a flexible ligand docking experiment in Maestro without similarity:* Following steps 2 to 10 of Basic Protocol 2, set up and prepare a ligand or series of ligands for flexible ligand docking.

3. *Defining whether to reward or penalize ligand for having high molecular similarity to any of the probe molecules:* In the Similarity tab of the Glide Ligand Docking Panel of Maestro (see Fig. 8.12.15) the mode of using similarity must be defined. By default similarity is not included in a docking experiment. To include similarity select the Find similar ligands option if ligands are to be rewarded for high similarity to the probe molecules and select Find dissimilar ligands if ligands are to be penalized for high similarity.

4. *Specifying the probe molecules:* Probe molecules are those against which similarities for docked ligands will be calculated. Specify the probe molecules in the Similarity tab of the Glide Ligand Docking Panel by clicking the Browse button to import a Maestro or SD formatted structure file. Alternatively, a relative or absolute pathname to the file with probe molecules may be specified in the File of known actives entry box.

   Up to 100 molecules with <200 atoms each may be included as probes. It is recommended to use a smaller representative set of ligands rather than a larger set with high ligand similarity.

5. *Specifying the functional form used to modulate GlideScore by similarity:* The functional form used to modulate the GlideScore by the maximum similarity to any probe molecule is defined in the Similarity tab of the Glide Ligand Docking Panel by adjusting four parameters, the Maximum GlideScore penalty, the penalty range maximum and minimum, and a rejection parameter. The Maximum GlideScore penalty is the maximum value that will be added to the GlideScore assuming full contribution from similarity. The penalty range minimum and maximum define the range of similarities between which the penalty will scale linearly from the Maximum GlideScore penalty

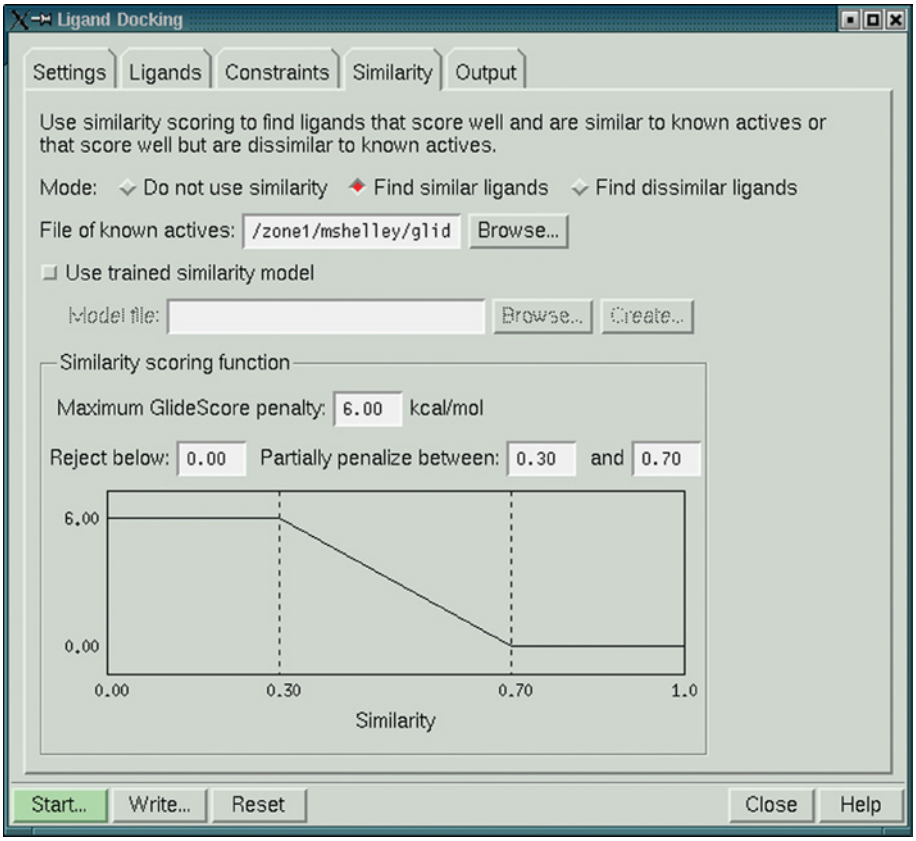


Figure 8.12.15 The Similarity tab of the Glide Ligand Docking panel.

to zero. Any ligands with a maximum similarity less than the 'Reject below' setting will not be docked.

Submit and monitor a Glide flexible ligand docking with similarity experiment

6. *Submitting a flexible ligand docking experiment with similarity for execution:* In Maestro click the Start button on the Glide Ligand Docking Panel to display the Ligand Docking – Start panel as shown in Figure 8.12.7. In this panel the job name that uniquely identifies the job to be run, the host the job is to be run from, and job distribution options must be specified. The job name should be a single word without special characters ([!@#$%^&*]). The host is selected from a list of hosts specified in the `schrodinger.hosts` file as described in Support Protocol 3. Docking jobs may be split into a number of subjobs that may be distributed over a number of processors.

7. *Monitoring the flexible ligand docking with similarity experiment:* Progress of the Glide ligand docking experiment can be monitored in the Monitor Panel of Maestro. This panel, shown in Figure 8.12.8, is displayed automatically when a Glide ligand docking experiment is started. Alternatively, the Glide Monitor panel can be opened by selecting Monitor in the Maestro Applications menu.

Analysis of Glide results

8. *Analyzing poses by GlideScore and protein-ligand interactions:* Following steps 14 to 15 from Basic Protocol 2, analyze the results of the flexible ligand docking with similarity experiment.

LIGAND PREPARATION

Protonation and tautomeric states of ligands are important in docking as they directly affect the ability of a ligand to form hydrogen bond interactions with the receptor. Glide scoring takes into account the significant free energy penalty associated with desolvation. This makes generating the correct protonation states for ligands crucial. Chirality is important because it affects molecular shape and can affect binding affinity by orders of magnitude. Corporate databases or purchasable compounds databases are often stored in SMILES or 2-D representations, and may contain counter-ions from salts. To generate optimal results with Glide, each ligand should have a high quality 3-D conformation. The initial geometries are important because conformation generation within Glide only samples torsions, keeping the bond lengths and bond angles in the input structure. Database ligands may also contain problematic ligand structures that are either chemically incorrect or have species that are not covered by force field parameters.

The LigPrep program provides a versatile and robust procedure for preparing ligands for small scale docking as well as large scale database screening, addressing the issues outlined above. In step 5, Epik can be used as an option.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

Software

LigPrep, Epik and Maestro (see Support Protocol 3)

Files

A file containing ligand structures. The supported formats are Maestro, SD, and SMILES strings.

1. Download and install Maestro and Glide on an accessible computer (see Support Protocol 3).

Set up ligand preparation through LigPrep in Maestro

2. Open the LigPrep Panel by selecting the LigPrep option under the Maestro Applications menu. The LigPrep panel will appear as shown in Figure 8.12.16.
3. *Selecting the source of input structure*: Input structures can come from (a) a file containing multiple structures, (b) selected entries in the Maestro Project Table, and (c) entries included in the workspace. Taking input structures from a file is the best option for a large number of structures. Supported file formats are Maestro, SD, and SMILES.
4. *Selecting the force field to be used*: OPLS_2005 (default) and MMFFs are supported.
5. *Selecting an option for treating stereoisomers*: With "Retain specified chiralities," R/S chiralities specified in an SD or Maestro structure file will be respected. For any chiral centers that are not specified in the input structure file, both R and S states will be generated. The second option "Determine chiralities from 3D structure" can be used when the input ligands are 3-D structures and the chiralities will be determined based on Cartesian coordinates. With the last option "Generate all combinations," both R and S states will be generated for all chiral centers disregarding any chiralities that are specified in structure files. One can set the upper limit on the

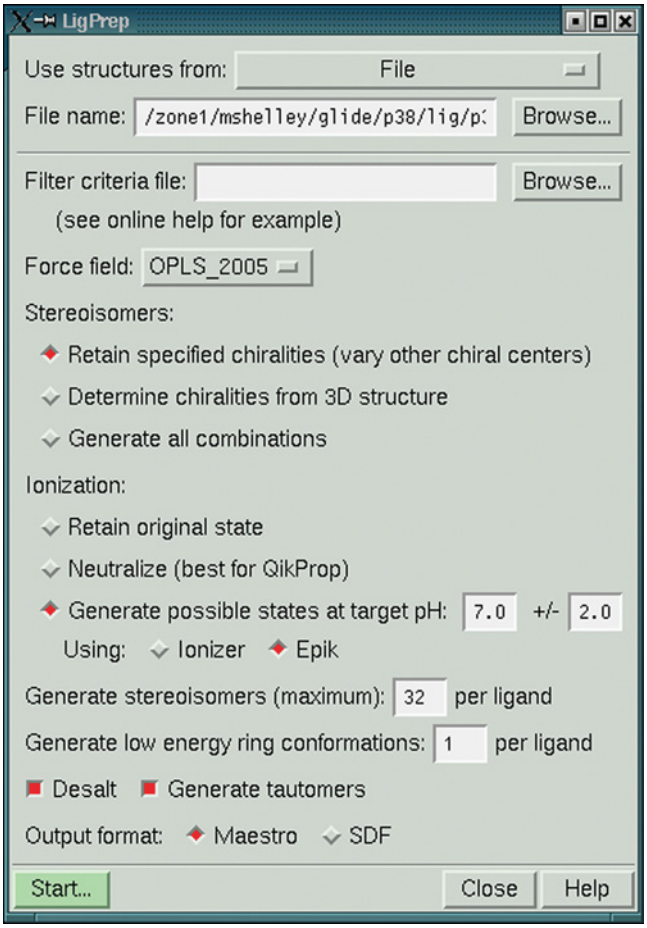


Figure 8.12.16 The LigPrep Panel.

number of stereoisomers per ligand by entering a number in the box after "Generate stereoisomers (maximum)."

6. *Selecting an option for generating protonation states*: The default option "Generate possible states at target pH" is for producing probable protonation states at a specified pH range. The Ionizer mode uses a simple SMARTS pattern matching for estimating pK_a vales. If the Epik mode is selected, the Epik program is used for more accurately predicting pK_a and generating protonation. For the best results, use the Epik mode. With the "Neutralize" option, all functional groups are neutralized, and with the "Retain original state" option, protonation states in the input structures are kept unchanged.

7. *Setting the number of ring conformations per ligand*: By default, LigPrep generates the lowest ring conformation only. If desired, multiple ring conformations can be generated. LigPrep will generate a specified number of lowest energy ring conformations.

8. *Setting other options*: If "Desalt" is selected, counter-ions from salts will be removed. If "Generate tautomers" is selected, LigPrep will generate reasonable tautomeric states.

9. *Selecting the format of output structures*: Maestro and SD format are supported.

Submit and monitor a LigPrep process

10. *Submitting a LigPrep process execution:* In Maestro click the Start button on the LigPrep Panel to display the LigPrep – Start panel. In this panel whether the output structures should be incorporated into the Maestro Project table, the job name that uniquely identifies the job to be run, the host the job is to be run from, and job distribution options must be specified. The job name should be a single word without special characters ([!@#$%^&*]). The host is selected from a list of hosts specified in the `schrodinger.hosts` file as described in Support Protocol 3. Docking jobs may be split into a number of subjobs that may be distributed over a number of processors.

11. *Monitoring the flexible ligand docking with similarity experiment:* Progress of the Glide ligand docking experiment can be monitored in the Monitor Panel of Maestro. This panel shown in Figure 8.12.8 is displayed automatically when a Glide ligand docking experiment is started. Alternatively, the Glide Monitor panel can be opened by selecting Monitor in the Maestro Applications menu.

SUPPORT PROTOCOL 2

RECEPTOR PREPARATION

The PDB format that is used for storing a protein structure does not store bond order information for ligands or other nonstandard residues. X-ray structures have critical weaknesses: hydrogen locations are not accurately resolved, making it difficult to determine the locations of hydroxyl and thiol hydrogen atoms in the protein, ligands, or cofactors, as well as the protonation/tautomeric state of histidine. Without the knowledge of the hydrogen locations, it is generally not possible to distinguish the oxygen and nitrogen in the amides of ASN and GLN. A 180° flip of the relevant chi dihedral angle, transposing the oxygen and nitrogen atoms, will often produce an alternate structure that is equally consistent with the electron density. A similar ambiguity exists with histidine, with respect to the carbon and nitrogen atoms of the imidazole ring. In order to make the best use of X-ray structures in modeling studies, it is important to resolve these structural ambiguities. The Protein Prep Wizard provides a streamlined procedure for converting a raw PDB structure into a well-prepared structure for docking.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

Software

Python scripts, Protein Preparation Wizard (`prepwizard.py`) and Protein Assignment (`protassign.py`), Maestro, Epik, and Glide. See Support Protocol 3.

Files

A file containing a receptor structure

1. Install the python scripts using the Manage Scripts panel on Maestro.
2. *Opening the Prep Wizard panel:* Maestro → Scripts → Protein Preparation Wizard. The Prep Wizard panel appears as shown in Figure 8.12.17.
3. *Importing a receptor structure into the Workspace of Maestro*: As an option, if Prime and its associated third-party databases have been installed, the Import PDB button

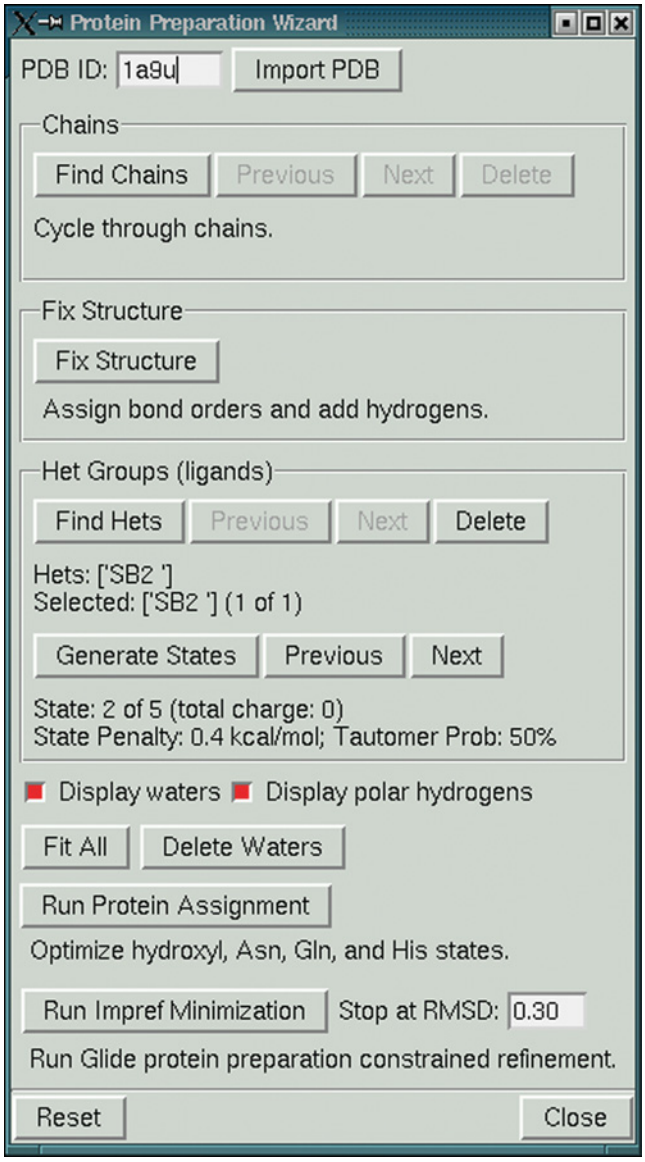


Figure 8.12.17 The Protein Preparation Wizard panel.

can be used to extract a receptor structure from a database on disk and import it into the Workspace.

Prime is a highly accurate protein structure prediction suite of programs that integrates Comparative Modeling and Threading. The Comparative Modeling path incorporates the complete protein structure prediction process from template identification, to alignment, to model building, and finally to refinement. Refinement involves side-chain prediction, loop prediction, and minimization. The threading path takes a sequence through a Fold Recognition module to alignment, model building, and refinement. In the context of this Support Protocol it provides a mechanism for extracting the desired receptor structure from the Prime PDB database.

Selecting a receptor structure: The receptor structure typically is a co-crystallized structure from Protein Data Bank, corporate databases, or other sources. If there are several choices (for example, co-crystallized with different ligands) for a given receptor, several criteria can be used for selecting the best structure. Usually it is best to choose a structure that contains a drug-like compound resembling the ligands that will be used in subsequent docking studies. The X-ray resolution and B factors, missing residues and atoms

are among other factors to consider. If it is expected that ligand-induced conformational changes lead to a few distinct conformations of the receptor, it makes sense to prepare each conformation and separately use for docking.

4. *Selecting chains to keep*: If the receptor structure consists of multiple chains, click Find Chains to color the Workspace structure by chain. The selected chain is colored cyan and the other chains are colored dark blue. Clicking Previous or Next moves the selection in alphabetical order of the chain names. Click Delete to remove chains that will not be used for modeling.

5. *Assigning bond orders to ligands and other nonstandard residues*: In the Prep Wizard panel click the "Fix Structure" button. In this step, hydrogen atoms are also added to the entire structure in the Workspace.

6. *Verifying bond orders and formal charges have been correctly assigned*: Click the "Find Hets" button on the Prep Wizard panel to examine ligands and other nonstandard residues. Visually inspect to confirm that all bond orders and formal charges have been correctly assigned.

   "Hets" refers to HET groups in the pdb file: molecules, part-molecules, or nonstandard residues (excluding waters) defined by HETATM records.

7. *Evaluating ligand protonation and tautomeric states:* Determine all reasonable protonation and tautomeric states of ligands by clicking the "Generate States" button. This runs an Epik calculation to determine the pK_a of ionizable groups for a more

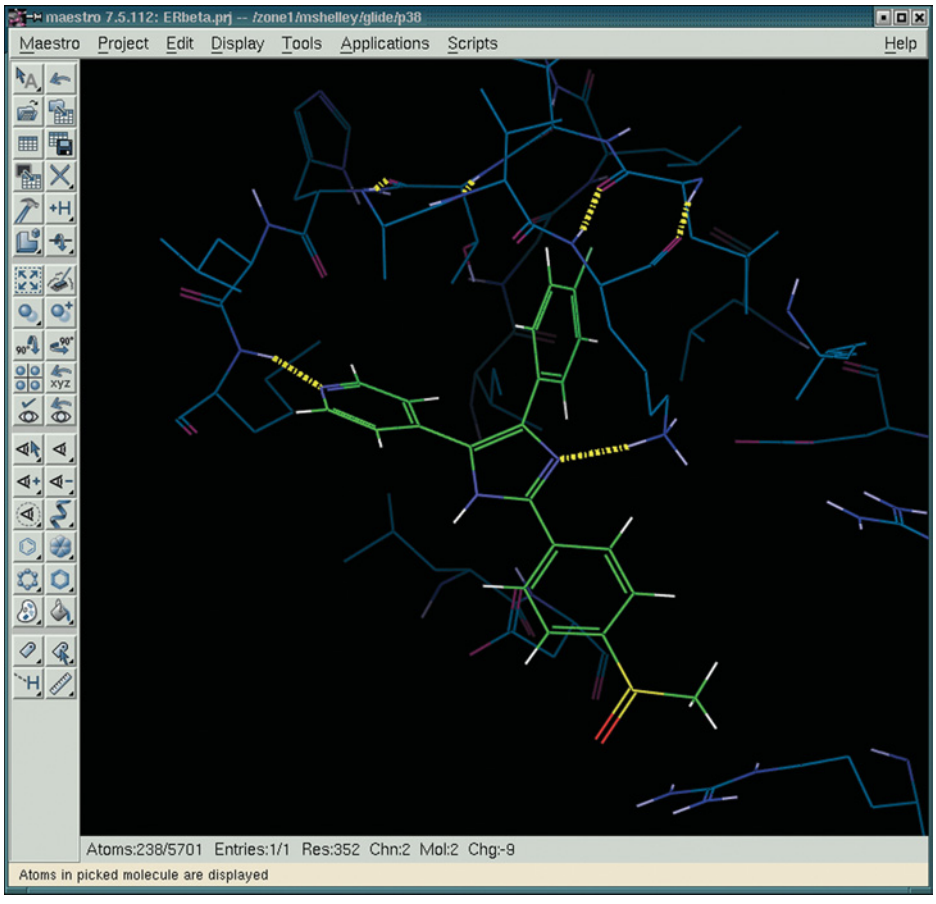


Figure 8.12.18 The Selected tautomeric/protonation state of a ligand displayed in the Workspace. For color version of this figure see *http://www.currentprotocols.com.*

accurate assessment of the ionization state. Examine each state by pressing the Previous and Next buttons and select the most appropriate state. See Figure 8.12.18 for an example.

8. Delete waters and add hydrogens.

9. *Evaluating protein protonation and tautomeric states:* Click the "Run Protein Assignment" button to determine tautomeric/protonation states of each histidine residue, make chi flips when appropriate, and adjust hydroxyl and thiol orientations.

10. *Reviewing the structure and making custom modifications if necessary*: For certain receptors (for example, metalloproteases), special treatment of protonation states are necessary.

11. *Gently relax the structure by performing restrained energy minimization*: Click the "Run Impref Minimization" button to run an impref minimization of the protein and ligand. By default the minimization is run until heavy atoms deviate from the crystal structure with an RMSD of 0.3.

    Values of 0.18 to 0.30 are recommended with 0.18 providing the smallest geometric difference between the minimized structure and the original crystal structure.

SOFTWARE INSTALLATION

SUPPORT PROTOCOL 3

The Glide and Maestro programs are commercial software licensed through Schrodinger, LLC. They are provided only as precompiled binaries for supported platforms. For Glide or Maestro to function, a valid license must first be obtained from Schrodinger, LLC.

Necessary Resources

Hardware

Unix/Linux workstation (e.g., Linux PC, Windows PC, IBM Power Series, Silicon Graphics)

1. Request a Logon account for the Schrodinger Web site at *http://www.schrodinger.com*. A Logon account is required to download the Glide and Maestro applications.

2. Log on using your account and obtain the Glide and Maestro applications by visiting the Schrodinger Support Center (*http://www.schrodinger.com* → Resources & Downloads → Script Center). In the Software Downloads section follow the "Current Release Download" link and select Glide and Maestro by picking the appropriate operating system for your system. Follow remaining instructions to download the software.

3. Complete instructions for installation are available in the Installation Guide documentation provided with the download. Manuals for Glide and Maestro are also provided in the download.

    Information regarding setup of the `schrodinger.hosts` *file is also found in the Installation Guide.*

4. Request required licenses by contacting *help@schrodinger.com* or your local Schrodinger representative.

5. *Optional:* Obtain the `prepwizard.py` and `protassign.py` scripts necessary for Support Protocol 2 by visiting the Schrodinger Script Center (see *http://www.schrodinger.com* → Resources & Downloads → Script Center) to download the Protein Preparation Wizard and Protein Assignment script.

6. *Optional:* Obtain LigPrep and Epik software for Support Protocol 1 by including them in the software selected for download in step 2 of this protocol and following the remaining steps (3 to 4) of the protocol.

GUIDELINES FOR UNDERSTANDING RESULTS

Flexible ligand docking experiments with Glide are typically performed for two purposes, the identification of likely protein-ligand binding modes (Friesner et al., 2004) and in screening databases to identify compounds that are likely to have high binding affinity to a protein (Halgren et al., 2004). In database screening, ligands are ranked by SP or XP GlideScore. Both SP and XP GlideScores are empirical scoring functions developed to rank ligands by binding affinities. Recently, docking experiments with XP Glide have also been used to generate a collection of descriptors that describe the free energy of binding between the protein and ligand. These descriptors can then be used to gain insight into protein-ligand interactions that are important in developing ligands with high binding affinity for a target protein and in QSAR approaches with other descriptors to predict protein-ligand binding affinities. In all of these approaches there are two basic analysis steps, (1) analysis of the poses generated by Glide docking and (2) analysis of ligand ranks by SP and XP GlideScore. The analysis instructions provided below are also applicable in flexible ligand docking with constraints and/or similarity.

Processed output from grid generation with Glide is provided in a set of files. The output `jobname.log` and `jobname.out` files capture run-time messages indicating Glide's status. Warning and error messages from the Glide process will be output to these two files so they should be checked to ensure that the Glide process was completed successfully.

Processed output from flexible ligand docking with Glide is provided in four files. The output `jobname.out` and `jobname.log` files capture run-time messages indicating Glide's status as well as all per-ligand results. As with grid generation runs, these two files should be checked to ensure the Glide process was completed successfully. Ligand rankings, GlideScores, GlideScore components, and other per-ligand information are captured in the plain text `jobname.rept` file. The `jobname.rept`, `jobname.out`, and `jobname.log` files can be viewed with any text editor. For the `jobname.rept` file columns are aligned, thus, a fixed-space font such as Courier should be used. The 3-D structure of each ligand successfully docked in the reference frame of the protein, optionally along with the protein structure, is found in the `jobname_pv.mae` Glide pose viewer file (if the protein is included) or in the `jobname_lib.mae` file (if only ligands were requested as output). These structure files are always output in Maestro format.

Accurately predicted binding modes can provide valuable insights into understanding protein-ligand interactions and performing structure-based lead optimization. In order to test the performance of a docking algorithm in predicting binding modes, one can carry out an experiment in which the ligand is extracted from a co-crystallized protein-ligand complex and docked back into the protein. Take the top-ranked docked pose, scored by the Emodel function as discussed in the Commentary, and compare it with the co-crystallized ligand structure. The metric often used for accessing docking accuracy is the RMSD (root-mean squared deviation) of heavy-atom coordinates between the predicted mode and the correct mode. The lower the RMSD between the native co-crystallized ligand geometry and the docked pose, the better Glide positioned the ligand relative to the receptor. While there isn't universal consensus, most published work has used an RMSD of 2.5 Å to separate well-docked (<2.5 Å) from poorly-docked (>2.5 Å) poses. A pose under 1.0 Å is generally considered to be within the accuracy of ligand coordinate refinement in the co-crystallized complex. See Figure 8.12.19 for examples of well-docked and poorly-docked ligand poses.

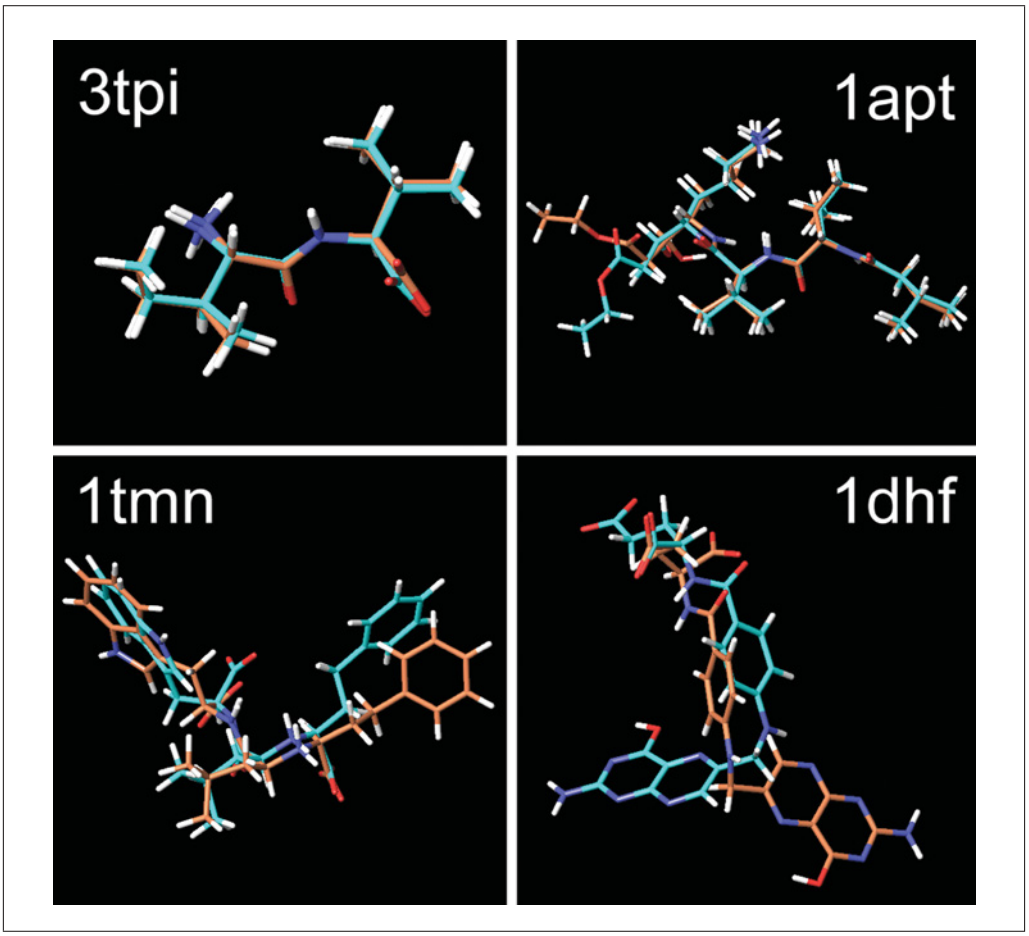


Figure 8.12.19 Examples of well-docked and poorly-docked ligands from docking co-crystallized ligands back into their prepared proteins. Ligands in blue are the native structures and those in green are the top-ranked docked structures. RMSDs for docked 3tpi, 1apt, 1tmn, and 1dhf ligands are 0.44, 1.26, 1.97, and 5.44 Å, respectively. For color version of this figure see *http://www.currentprotocols.com*.

Accurately predicted binding modes are crucially important for Glide to accurately rank ligands by GlideScore. The GlideScore scoring functions were designed to maximize enrichment by ranking ligands with known binding affinities. A poorly-docked pose (i.e., with RMSD >2.5 Å to the correct pose) is not likely to receive a GlideScore commensurate with its experimental affinity.

High-throughput virtual screening experiments performed with Glide are similar in spirit to high-throughput screening experiments, only run in silico. A large database of ligands is docked and the top fraction of the GlideScore ranked output ligands is then used in further processing with the expectation that the top fraction of ligands will be enriched in ligands with at least 10 micromolar or better experimental binding affinities. A common workflow for using Glide in high-throughput virtual screening is to dock a very large library of compounds using the fast HTVS mode. Using the more extensive sampling of SP mode, the top fraction of the output ligands from HTVS mode are docked. For more accurate ranking of ligands, the top-fraction of output ligands from the SP mode experiment is then docked with XP Glide. At each step of this workflow the top fraction of output ligands is further enriched in compounds likely to demonstrate affinity to the target protein.

Database enrichment experiments are commonly used to test the ability of a docking program to enrich the top fraction of output ligands in compounds with affinity to the

target protein. In this experiment, a set of ligands which are known to bind to a protein target, generally with at least 10 micromolar or better experimental binding affinity, are docked into that protein along with a set of decoy ligands. The decoy ligands are typically taken from various databases of available compounds with the assumption that only a small fraction of the decoy ligands would demonstrate measurable biological affinity to the protein target. The ability of the docking algorithm to rank known active ligands above decoy ligands is then tested. A perfect enrichment experiment in Glide would show all known active compounds to have better GlideScores (more negative) than all decoy ligands.

Several metrics are used to analyze the enrichment of high-ranked known active ligands:

1. Traditional enrichment metric (Pearlman and Charifson, 2001)

   $EF = (N_{total}/N_{sampled})*(HITS_{sampled}/HITS_{total})$

   Here, N_{total} is the number of ligands in the docked database, $N_{sampled}$ is the number of ligands in the docked database to be examined, $HITS_{total}$ is the total number of known active ligands, and $HITS_{sampled}$ is the number of known active ligands found in the top $N_{sampled}$ ligands of the docked database. Thus if only 10% of the scored and ranked database need to be assayed to recover all the $HITS_{total}$ active ligands, the enrichment factor would be 10. In other words the number of active ligands in the top 10% of the database is enriched 10-fold over a random distribution of active ligands. If only half the total number of known active ligands are found in the top 10% (i.e., if $HITS_{sampled}/HITS_{total} = 0.5$) the enrichment factor would be 5. This enrichment metric has three main weaknesses, it is dependant on the number of known active ligands and penalizes active ligands that are outranked by other known active ligands, it is dependent on the number of decoy ligands employed, and it does not measure the distribution of known active ligands, rather uses only the lowest-ranked known active ligand found in the $N_{sampled}$ to set the enrichment.

2. Weighted enrichment metric (Halgren et al., 2004)

   $EF' = (50\%/APR_{sampled})*(HITS_{sampled}/HITS_{total})$

   For this metric, larger values indicate more known active ligands are found to be ranked higher than the decoy ligands. Here, $APR_{sampled}$ is the average percentile rank of the $HITS_{sampled}$ known active ligands with $HITS_{sampled}$ and $HITS_{total}$ as defined in the traditional enrichment metric. Thus, if the active ligands are uniformly distributed over the entire ranked database, the average percentile rank for an active ligand would be 50% and the enrichment factor would be 1. This metric considers the ranks of all $HITS_{sampled}$ known active ligands in the examined portion of the ranked database. Generally, the enrichment from this metric is larger than the traditional enrichment metric if known active ligands are concentrated toward the beginning of the $N_{sampled}$ ranked positions. Enrichments by this metric will be smaller than the traditional enrichment metric if known active ligands are concentrated toward the end of the list. This metric is dependent on the number of known active ligands and the number of decoy ligands.

3. Average number of outranking decoy ligands (Friesner et al., 2006)

   The average number of outranking decoy ligands metric is an easily interpreted measure of the average number of decoy ligands that are found to outrank known active ligands. Specifically, the number of database ligands with a GlideScore that is superior to each active ligand is tabulated, these values are summed, and the result is then divided by the total number of active compounds in the data set. Smaller

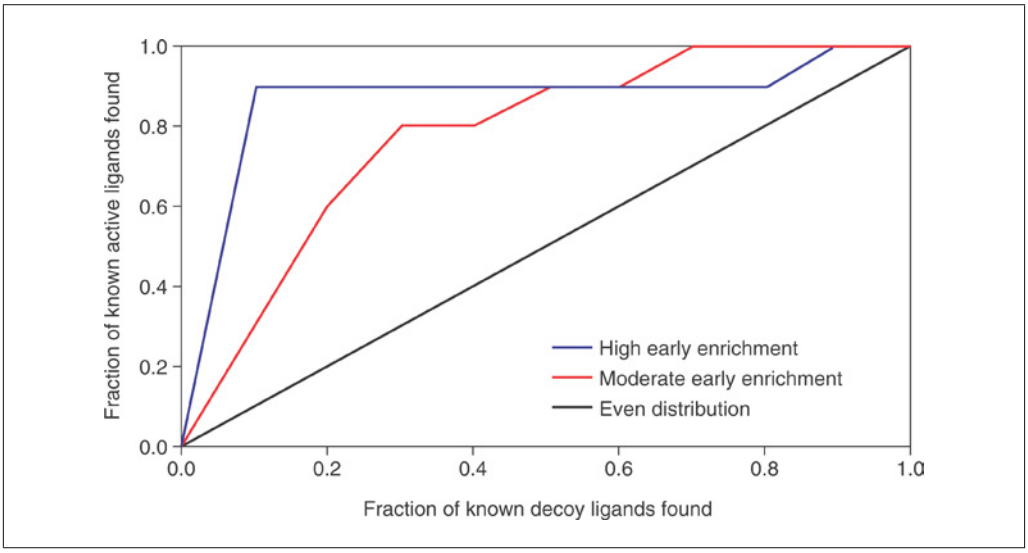


Figure 8.12.20 Three hypothetical enrichment curves. In the high early enrichment example 90% of known active ligands are found before 10% of the known decoy ligands were recovered. In the moderate early enrichment example 30% of known active ligands were found before 10% of the known decoy ligands were recovered. These results stand in contrast to what would be expected by chance, shown in the even distribution curve. For color version of this figure see *http://www.currentprotocols.com*.

values indicate better enrichment; a value of zero indicates no decoy ligands were found to be ranked above any known active ligands. This method is independent of the number of known active ligands, though it is dependent on the number of decoy ligands.

4. Enrichment curves, also known as receiver-operating-characteristic curves

   The previously defined metrics attempt to quantify with a single number the number and ranks of known actives in an enrichment experiment. To display more information including the ranks of the known active ligands it is convenient to plot enrichment curves. An enrichment curve is a plot of the percent of known actives recovered versus the percent of database screened (Fig. 8.12.20). The area under an enrichment curve may be calculated to provide a single quantitative performance metric. This metric spans the range of 0.0 to 1.0 with 0.0 indicating no known actives were recovered and 1.0 indicating all known actives were recovered ranked ahead of all decoy ligands.

COMMENTARY

Background Information

Glide was designed to aid and guide lead discovery and lead optimization in pharmaceutical research. Glide's main roles include predicting the binding modes for small molecules in protein/ligand complexes, finding new leads via virtual screening, and aiding in understanding structure-activity relationships.

Glide uses a flexible ligand–rigid receptor approach, i.e., ligand conformations are sampled, but the receptor coordinates are held constant during docking. The rigid receptor approximation allows rapid evaluation of ligand docking and is remarkably useful in a wide range of applications. This, despite the fact that the act of ligand biding is known to effect protein conformation (Teague, 1996). Reducing the van der Waal's radii of nonpolar atoms in the ligands or the receptor allows some extra room to accommodate different ligand structures into a rigid receptor. The default scaling factors for the van der Waal's radii in Glide (1.0 for receptor and 0.8 for ligands) have been selected to strike a balance between accurate docking/scoring and compensation for the lack of receptor flexibility. If conformational changes of the receptor are substantial, receptor flexibility should be taken into account more explicitly. If it is expected that ligand-induced conformational changes lead

to a few distinct conformations of the receptor, it makes sense to prepare each of these conformations, separately dock ligands into them, and then combine the docking results. This method is often called "ensemble docking." The Induced Fit Docking procedure (Sherman et al., 2006) explicitly models ligand-induced conformational changes for each ligand. It can be used for studying binding modes of specific ligands or for generating receptor conformations that can be used for ensemble docking.

All docking methodologies must tackle two difficult problems, ligand free-energy scoring and sampling of ligand conformations and locations relative to the rigid receptor. Two very different approaches to solving the sampling and scoring problem are available within Glide. The SP/HTVS Glide methods use a series of hierarchical filters to search for possible locations and conformations of a ligand in the active site region of the receptor (Halgren et al., 2004). For each core conformation of a ligand, an exhaustive search of possible positions and orientations is performed over a 1 $\mathring{A}^3$ grid overlaid on the active site of the protein. The search starts with the selection of site points on the grid which a ligand center could feasibly occupy and progressively a more accurate evaluation is performed on the poses that pass each stage of the hierarchical filters. In the final stage, a small number of poses of the ligand (400 by default) are optimized on van der Waal's and electrostatic grids representing the receptor, and poses are re-scored using GlideScore. The selection of best-docked structure for a ligand among the up to 400 poses minimized is made using a model energy score called "Emodel" that combines the energy score, the binding affinity predicted by GlideScore, and the internal strain energy for the model potential used to direct the conformation generation algorithm. It should be noted that the last term is not designed to be an accurate evaluation of the ligand strain energy. An estimate of the ligand free energy used for comparing the ligand pose to other ligands is then calculated via the SP GlideScore scoring function shown below.

$$SP\ GlideScore = C_{lipo\text{-}lipo} \sum f(r_{lr}) + C_{hbond\text{-}neut\text{-}neut} \sum g(\Delta r)h(\Delta\alpha) + C_{hbond\text{-}neut\text{-}charged} \sum g(\Delta r)h(\Delta\alpha) + C_{hbond\text{-}charged\text{-}charged} \sum g(\Delta r)h(\Delta\alpha) + C_{max\text{-}metal\text{-}ion} \sum f(r_{lm}) + C_{rotb}H_{rotb} + C_{polar\text{-}phob}V_{polar\text{-}phob} + C_{coul}E_{coul} + C_{vdW}E_{vdW} + solvation\ terms$$

The SP GlideScore is empirical as coefficients (C) were fit to maximize enrichment and correlation with experimental binding affinities. The ligand free-energy is estimated with a lipophilic term that rewards placing hydrophobic ligand and receptor moieties in close contact, three hydrogen bond terms to reward the formation of protein-ligand hydrogen bonds, a metal ligation term, a rotatable bond term to roughly estimate the entropy loss upon binding, a term to penalize close contact between polar and hydrophobic moieties, and terms to account for ligand solvation.

XP Glide involves significantly more extensive sampling of poses and a scoring function based on a more rigorous evaluation of protein-ligand interactions (Friesner et al., 2006). The sampling method in XP Glide is based on an anchor and refined growth strategy. Anchor fragments of the docked ligand, typically rings or other rigid fragments, are chosen from the set of poses output from an initial docking with SP Glide designed to obtain a large diversity of docked structures. Ligand poses are regrown one side chain at a time at very high resolutions from these anchor positions. A set of candidate molecules is selected by combining high-scoring individual conformations at each side chain. Energy minimization of the candidate molecules is carried out and ligand poses are ranked according to the Emodel pose-selection function. Grid-based water scoring technology is then applied to the top-scoring structures and the full XP GlideScore scoring function is computed. At this point side chains which suffer energetic penalties are regrown to eliminate such penalties if possible. This focused sampling is essential for allowing the use of the rigorously discriminating XP GlideScore scoring function as well as for finding the best scoring basins of attraction. It is important to note that the coupling between the extra sampling and the XP scoring means that it is not recommended to just score the SP poses with XP scoring.

$$XP\ GlideScore = E_{coul} + E_{vdW} + E_{bind} + E_{penalty}$$
$$E_{bind} = E_{hyd\_enclosure} + E_{hb\_nn\_motif} + E_{hb\_cc\_motif} + E_{PI} + E_{hb\_pair} + E_{phobic\_pair}$$
$$E_{penalty} = E_{desolvation} + E_{ligand\_strain}$$

In the XP GlideScore scoring function, specific complex structural motifs are identified as leading to enhanced binding affinities. Such motifs include (1) a hydrophobic enclosure which identifies a group of lipophilic ligand atoms enclosed on two opposite faces by lipophilic protein atoms, (2) special neutral-neutral hydrogen bonds which are single or correlated hydrogen bonds in a

hydrophobically enclosed environment, and (3) five categories of special charged-charged hydrogen bonds. The XP scoring function includes terms that represent such motifs along with the pairwise hydrogen bond and hydrophobic terms from SP GlideScore, a novel water scoring desolvation energy term, an estimate of ligand strain, a term for pi-pi and pi-cation interactions, and weighted coulomb and van der Waal's terms.

Several programs from commercial and academic sources are available to perform flexible ligand docking. These programs all use different methods to address sampling, scoring, or both. Several comparison studies have been run by independent researchers including Perola et al. (2004), Kontoyianni et al. (2004), and Krovat et al. (2005) which show the performance of Glide in pose prediction and enrichment compares favorably across a wide range of protein targets.

Critical Parameters and Troubleshooting

The quality of Glide docking results depends on several factors including the quality of input protein and ligand structures, the grid dimensions used in grid generation, and van der Waal's scaling factors for protein and ligand. In certain cases, the treatment option can significantly alter results.

Glide pose prediction and ligand ranking by free energies are very dependent on the quality of protein and ligand preparation. For both ligand and proteins, the input must posses a valid lewis structure and appropriate protonation and tautomeric states. If docking results are poor, it is strongly recommended to examine the receptor and ligand structures, since structural preparation has been found to be the most common cause of problems. Check for noncomplimentary protein-ligand interactions that may be due to nonphysical protein residue or ligand protonation/tautomerization. Furthermore, it is important to use a protein structure that is most appropriate for a given run. For instance, if screening a large database of ligands for lead discovery, an open form of the binding site may be desired to maximize the number of ligands that will fit. However, in optimizing protein-ligand interactions it may be most desirable to use a form of the binding site most similar to that with a lead compound.

The grid dimensions for the bounding and enclosing boxes used in the grid generation step are set automatically if a ligand is provided in the binding site. The formula to calculate these defaults has been determined to generate the best results in pose prediction and enrichment for a wide range of systems. These default sizes are to be used if ligands to be docked are approximately the same size as the ligand used in grid generation. If they are considerably larger, ligands will likely be lost during docking as the poses may have ligand atoms outside the enclosing box or the ligand center outside the bounding box. If the grid boxes are too large, inefficient sampling will result. If no ligand is available to specify the grid dimensions it is recommended to have the enclosing box cover regions of the binding site where it is desired for ligands to interact. The bounding box should generally be maintained at either 10 or 12 Å except for very floppy ligands in which the ligand center is likely to fall outside a 12 × 12 × 12–Å box.

The default van der Waal's scaling factors for Glide (1.0 for the protein, 0.8 for the ligand) have been found to generate the best results for the widest range of systems. However, there are exceptions where using smaller scaling and/or scaling the protein instead of the ligand have generated better results and the user should explore different scaling factors if docking results are unsatisfactory.

The amide potential governing rotation of the C-N bond is heavily dependent on substituent effects and the chemical environment of the amide group. Thus, while it is often desired to treat the C-N bond as nonrotatable, fixed at either *cis* or *trans*, there are often situations where it is necessary to have the C-N bond treated as rotatable. To ensure that ligands with amide functional groups are docked in the manner expected by the user, this option should be considered.

Flexible ligand docking is the most widely used mode of Glide. Glide supports flexible docking for ligands with up to 35 rotatable bonds. For very large and flexible ligands that have >35 rotatable bonds, rigid docking in combination with external generation of conformers can be used. The MCMM method or mixed LMOD/MCMM method in MacroModel are often used for generating conformers for such applications (see the MacroModel user manual for further details).

Literature Cited

Friesner, R.A., Banks, J.L., Murphy, R.B., Halgren, T.A., Klicic, J.J., Mainz, D.T., Repasky, M.P., Knoll, E.H., Shelley, M., Perry, J.K., Shaw, D.E., Francis, P., and Shenkin, P.S. 2004. Glide: A new approach for rapid, accurate docking and scoring. 1. Method and assessment of docking accuracy. *J. Med. Chem.* 47:1739-1749.

Friesner, R.A., Murphy, R.B., Repasky, M.P., Frye, L.L., Greenwood, J.R., Halgren, T.A., Sanschagrin, P.C., and Mainz, D.T. 2006. Extra precision Glide: Docking and scoring incorporating a model of hydrophobic enclosure for protein-ligand complexes. *J. Med. Chem.* 49:6177-6196.

Halgren, T.A., Murphy, R.B., Friesner, R.A., Beard, H.S., Frye, L.L., Pollard, W.T., and Banks, J.L. 2004. Glide: A new approach for rapid, accurate docking and scoring. 2. Enrichment factors in database screening. *J. Med. Chem.* 47:1750-1759.

Kontoyianni, M., McClellan, L.M., and Sokol, G.S. 2004. Evaluation of docking performance: Comparative data on docking algorithms. *J. Med. Chem.* 47:558-565.

Krovat, E.M., Steindl, T., and Langer, T. 2005. Recent advances in docking and scoring. *Curr. Comp. Aid Drug Des.* 1:93-102.

Pearlman, D.A. and Charifson, P.S. 2001. Improved scoring of ligand-protein interactions using OWFEG free energy grids. *J. Med. Chem.* 44:502-511.

Perola, E., Walters, W.P., and Charifson, P.S. 2004. A detailed comparison of current docking and scoring methods on systems of pharmaceutical relevance. *Proteins* 56:235-249.

Sherman, W., Day, T., Jacobson, M.P., Friesner, R.A., and Farid, R. 2006. Novel procedure for modeling ligand/receptor induced fit effects. *J. Med. Chem.* 49:534-553.

Teague, S.J. 1996. Implications of protein flexibility for drug discovery. *Nature Rev. Drug Discovery* 9:175-186.

Internet Resources

http://www.schrodinger.com

Get information on or to download Glide and auxiliary applications

---

Contributed by Matthew P. Repasky
Schrodinger, L.L.C.
New York, New York

Mee Shelley
Schrodinger, L.L.C.
Portland, Oregon

Richard A. Friesner
Columbia University
New York, New York

Exploring Biological Networks with Cytoscape Software

Natalie Yeung,[1] Melissa S. Cline,[2] Allan Kuchinsky,[3] Michael E. Smoot,[4] and Gary D. Bader[1]

[1]University of Toronto, Donnelly Centre for Cellular and Biomolecular Research, Toronto, Ontario, Canada
[2]Department of Molecular, Cell, and Developmental Biology, University of California, Santa Cruz, California
[3]Agilent Technologies, Santa Clara, California
[4]Department of Bioengineering, University of California, San Diego, La Jolla, California

ABSTRACT

Cytoscape is a free software package for visualizing, modeling, and analyzing molecular and genetic interaction networks. As a key feature, Cytoscape enables biologists to determine and analyze the interconnectivity of a list of genes or proteins. This unit explains how to use Cytoscape to load and navigate biological network information and view mRNA expression profiles and other functional genomics and proteomics data in the context of the network obtained for genes of interest. Additional analyses that can be performed with Cytoscape are also discussed. *Curr. Protoc. Bioinform.* 23:8.13.1-8.13.20. © 2008 by John Wiley & Sons, Inc.

Keywords: network visualization • network analysis • systems biology • protein interactions • biological network

INTRODUCTION

Cytoscape is a free, open-source software platform used to graphically visualize biological networks (Shannon et al., 2003). Networks contain nodes, representing objects (such as proteins), and connecting edges representing relationships between them (such as physical interactions). Importantly, Cytoscape allows the integration of experimental and other relevant data—such as Gene Ontology annotation (*UNIT 7.2*) and gene expression profiles—stored as node and edge attributes, in a network context. These can be mapped to visual attributes (such as node shape or edge color) allowing data to be visualized in a network context in many useful ways.

The most basic Cytoscape task is to visualize a network created from interaction data (Basic Protocol). Expression data can then be loaded as node attributes and visualized on the network by mapping node attributes to node colors (Alternate Protocol). These tasks are graphically summarized in the protocol flowchart (Fig. 8.13.1). Many analyses can be performed using Cytoscape plug-ins, which are downloadable extensions of the main Cytoscape software. Plug-ins add functionality, e.g., fetching network data from public sources and analyzing network topology to find biologically interesting patterns. Cytoscape can be downloaded for desktop use on Windows, Mac OS X, and Linux machines (Support Protocol 1). Other UNIX platforms that support recent versions of Java are also supported.

All protocols and figures shown use Cytoscape 2.5.2 (the most recent version, as of July 2007).

Current Protocols in Bioinformatics 8.13.1-8.13.20, September 2008
Published online September 2008 in Wiley Interscience (www.interscience.wiley.com).
DOI: 10.1002/0471250953.bi0813s23

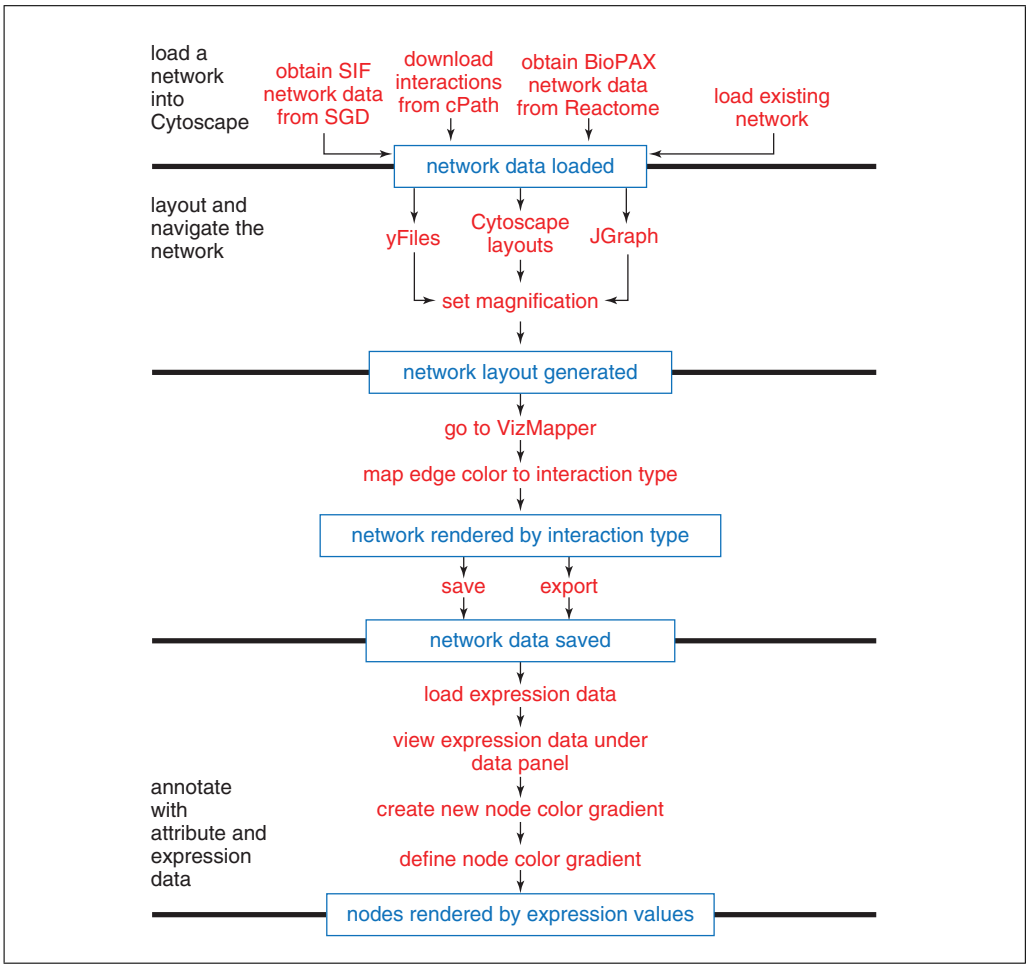


Figure 8.13.1 Flowchart summarizing the protocols defined in this unit.

BASIC PROTOCOL

VISUALIZE A NETWORK

This protocol outlines the steps necessary to create, lay out, and view networks in Cytoscape, along with tips for navigating the network and setting custom visual properties. Four network data loading methods are described; the first three involve downloading network data from online databases, while the fourth describes loading an existing local file.

Necessary Resources

Hardware

Computer with 1 GHz CPU or higher, a high-end graphics card, 60 MB of available hard disk space, at least 512 MB of free physical RAM (for networks up to 5000 edges; at least 1 GB of RAM (for larger networks) and a minimum screen resolution of 1024 × 768 (recommended; requirements depend on the size of the networks to be imported and analyzed)

Internet connection to obtain network data from online databases (not necessary to visualize interaction data from a local file)

Software

Operating System: Windows, Mac OS X, Linux, or another platform that supports Java

Java 2 Platform, Standard Edition, version 5.0 or higher *(http://java.sun.com/javase/downloads/index.jsp)*.

Internet browser: e.g., Microsoft Internet Explorer (*http://www.microsoft.com*), Mozilla Firefox (*http://www.mozilla.org/firefox)*, or Apple Safari (*www.apple.com/safari*), if downloading network files

Table 8.13.1 File Formats Supported by Cytoscape

File format	Description	Related URL
SIF (Simple Interaction Format)	Text format invented for Cytoscape (see Fig. 8.13.2)	*http://www.cytospace.org*
CYS (Cytoscape session file)	Default Cytoscape file format, containing both interaction data and visual properties	*http://www.cytospace.org*
GML (Graph Markup Language)	Standard network file format supported by multiple generic network software packages	*http://www.infosun.fml.uni-passau.de/Graphlet/GML*
XGMML (eXtensible Graph Markup and Modeling Language)	Standard XML format similar to but preferred over GML, since it can contain more information	*http://www.cs.rpi.edu/~puninj/XGMML*
SBML (Systems Biology Markup Language)	Standard XML format for representing mathematical pathway models	*http://sbml.org/documents*
PSI-MI (Proteomics Standards Initiative-Molecular Interaction format)	XML standard format for molecular interactions supported by molecular interaction databases	*http://www.psidev.info/index.php?q = node/60*
BioPAX (Biological PAthway eXchange)	Standard format for pathway information supported by multiple pathway databases	*http://www.biopax.org*

Cytoscape 2.5.2, downloaded from *http://cytoscape.org* (see Support Protocol 1 to install a local copy)

Files

No external files required for downloading network data from online databases

Local files (if used): e.g., Microsoft Excel (`.xls`) or text files containing interaction data arranged in columns; example files available in the Cytoscape/sampleData folder created during Cytoscape installation (Support Protocol 1), some of which can be opened, viewed, and edited in a plain text editor such as Notepad or TextEdit; see Table 8.13.1 for standard supported file formats

Load a network into Cytoscape

1. Open Cytoscape by clicking the Cytoscape icon created during installation.

   This step is for users who used the automatic install program. See Support Protocol 1 for the installation procedure and a description of the user interface.

   For alternate methods of opening Cytoscape, see Support Protocol 1, step 5.

2. The Cytoscape desktop will appear (Fig. 8.13.3).

   The toolbar at the top of the desktop contains command buttons with tooltips (the name of the function will appear when the mouse hovers over the button for more than a few seconds). The center of the screen, which is blank when Cytoscape is started, will display networks as they are loaded.

   At the left of the screen is the Control Panel, which has four major tabs: the Network tree viewer, the VizMapper, a network Editor, and a basic Filters function.

   The Network tree viewer displays a list of all loaded networks and the number of nodes and edges that they contain. It also contains the Network overview panel at the bottom of the tab, which shows the current network with a blue box highlighting the portion currently being viewed.

   The VizMapper controls the node, edge, and global network visual properties of Cytoscape networks, and facilitates user-defined mapping of attribute data to visual properties.

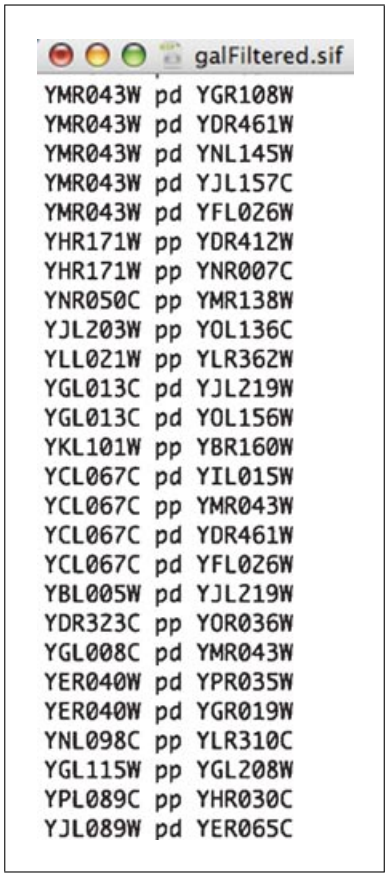


Figure 8.13.2 A few lines from the `galFiltered.sif` protein interaction network file included in the `Cytoscape/sampleData` directory. The first and last columns contain node IDs, while the middle column defines an edge type.

3. Obtain new or load existing network data, using one of the following methods:

   a. Obtain yeast network data from the Saccharomyces Genome Database (SGD, Christie et al., 2004; Support Protocol 2).

   b. Obtain network data using the cPath database (Support Protocol 3).

   c. Obtain a biological pathway from the Reactome database (Support Protocol 4).

   d. Load an existing network data file (Support Protocol 5).

4. Click Import to load the network. Cytoscape will display a progress screen as it loads the data.

5. Check that loading status is successful, and then click Close.

   Steps 1 through 4 can be repeated multiple times, i.e., many networks can be loaded in separate Cytoscape windows.

6. To switch between networks, click on the filenames in the Network tab of the Control Panel (Fig. 8.13.3). Note that only one session can be loaded at a time.

Layout and navigate the network

7. If a network is not displayed after the data is successfully loaded, create a view of the network by selecting the Edit→Create View menu option.

   Small networks will have a view automatically created when they are loaded, while large networks (i.e., thousands of nodes and edges) will be loaded without a view. Larger networks are usually slower and harder to work with, due to their need for greater computational resources. However, they can be reduced to a selected subset of nodes and edges using the Filters function and then viewed as a smaller network. Filters are described in more detail in step 11e of this protocol.

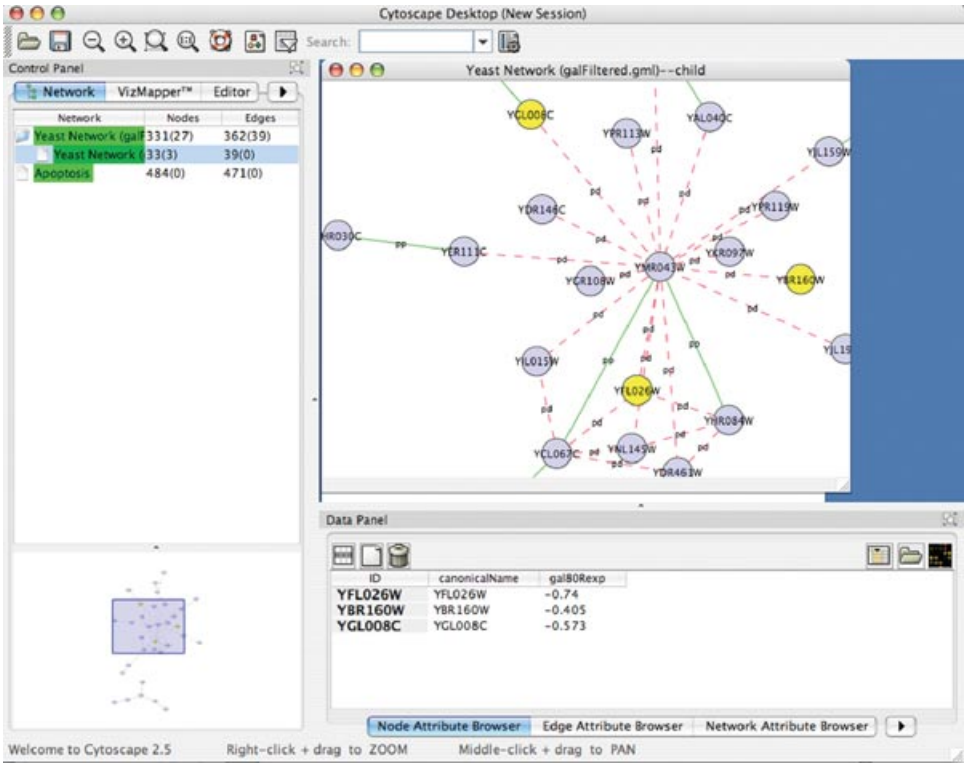


Figure 8.13.3 The basic Cytoscape user interface.

8. Apply a layout using the Layout menu. Applying a layout to a network moves the positions of nodes and edges to reduce overlap, provide a clearer visual representation of the data, and make the structure of the network more interpretable.

   Cytoscape offers a set of tools for automated layout, using a variety of algorithms, e.g., hierarchical, circular, and attribute-based layouts. In addition to automatic layout for the entire network, some of the tools optionally operate on selected parts of a network.

   Different layouts are tailored for different types of networks. Hierarchical layouts work better on tree-like networks, circular layouts work better if the network is circular, and force-directed type layouts—including the Cytoscape "Force-Directed" layout—are better for well connected networks. Force-directed layout algorithms model edges as springs and nodes as like-charged particles, so nodes repel each other and edges spring, but connect nodes at a preferred length. After a short simulation of this physical system, the layout produces a network layout where nodes do not overlap but are not too far away from each other.

   While most layouts do not consider information about the network other than the connectivity, some attribute-based layouts are available that place nodes and edges based on their attributes. Examples include using edge weights to calculate edge length or clustering nodes with common annotations together (Garcia et al., 2007).

   Networks can also be manually laid out by selecting single or multiple nodes and dragging them across the screen.

   The Rotate and Scale functions can be applied to the whole network or a subset of it, while Align, Distribute, and Stack (which allow aligning, evenly distributing, or stacking selected nodes in space on the canvas) require some or all nodes to be selected. To select nodes, click on each one while holding down the Shift key or click and drag to select an area containing the node set.

9. Adjust the viewing area and magnification of the network. There are six methods for navigating across a network:

   a. *Zoom out:* View a larger region of the network by clicking on the button depicting a magnifying glass with a minus (–) sign.

   b. *Zoom in:* View a smaller region of the network in greater detail by clicking on the button depicting a magnifying glass with a plus (+) sign.

   c. *Zoom to a selected region:* View a selected subset of the network by clicking on the button depicting a magnifying glass with a dotted rectangle.

   d. *View the entire network:* See the entire network at once by clicking on the button depicting a magnifying glass labeled 1:1.

   e. *Pan across the network:* View different portions of the network by clicking and dragging the blue box shown in the Network Overview in the lower left-hand corner of Cytoscape.

   f. *Continuous zoom:* Zoom in and out of a network by right clicking the mouse and dragging the mouse up and down over the network view.

10. Create a child network (new network containing a subset of the original parent) by selecting nodes and/or edges and then going to File→New→Network→From selected nodes, all edges. A second window will appear containing the new network.

11. Select nodes and edges of interest using one of the following methods:

    a. Hold down the Shift key while clicking on nodes and edges.

    b. Click and drag to select a region of the network.

    c. Use the options provided under Nodes and Edges in the Select menu.

    d. Use the Quick Find search box provided in the Cytoscape toolbar (see Fig. 8.13.4).

       Quick Find provides a fast way to select nodes or edges that share an attribute value or range of values. The default search is by node name, so typing the first few letters of a node name in the search box will bring up a list of all matching node names. Click on the configuration icon directly to the right of the search box to change the search to another node or edge attribute. For numerical attributes, the search box will change into a slider that allows the selection of a numerical range (see Fig. 8.13.5).

    e. Create and apply a filter using the Filters tab in the Control Panel.

       The Filters tab features a user-friendly interface that is also accessible from the funnel icon on the Cytoscape toolbar (see Fig. 8.13.4).

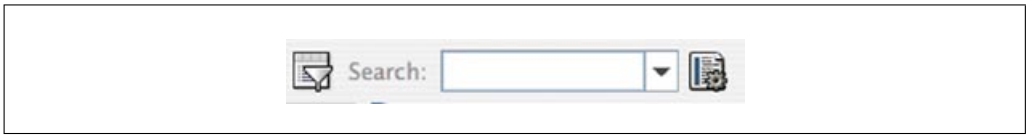


Figure 8.13.4 The funnel icon at the left of the figure opens the Filters tab in the Control Panel. Next to it is the Quick Find search box, and the Quick Find configuration icon is at the far right.

Figure 8.13.5 The Quick Find search box can filter numerical attributes by dragging the two triangles to define minimum and maximum values. All nodes or edges falling within this range will be selected.

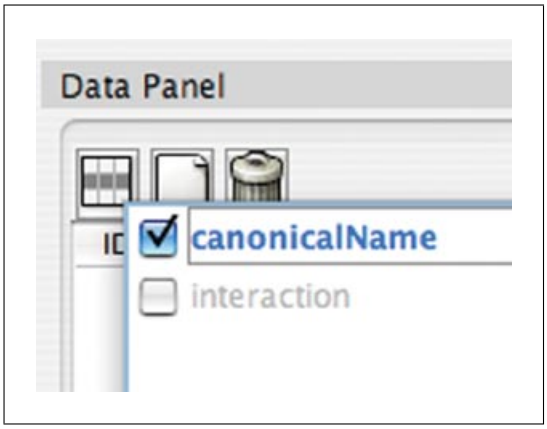


Figure 8.13.6 The Select Attributes icon is found at the far left of the Data Panel (the rectangle with a grey horizontal stripe). When clicked, a list of attributes appears. These will be displayed in the Data Panel if they are checked.

Filters serve as a more complex and flexible form of searching than Quick Find. Boolean and regular expression searches are supported, as well as all of the functionality available within Quick Find. More information on how to use filters can be found in the Filters chapter of the Cytoscape user manual (accessed from the Help menu online at http://www.cytoscape.org/cgi-bin/moin.cgi/Cytoscape_User_Manual; included in the Cytoscape installation directory).

In Cytoscape 2.5.2, these filters are restricted to AND *and* OR *Boolean expressions. Future versions of Cytoscape will extend this functionality.*

Set visual properties for network nodes and edges

12. Click on the Edge Attribute Browser tab in the Data Panel at the bottom of the screen to view the attributes associated with the edges in the network. By default, the edge ID (identifier) attribute is displayed.

    Selecting edges in the network will display their respective attribute values in the Edge Attributes tab. To view other attributes, click on the Select Attributes button at the left of the Data Panel to display a list of available attributes and highlight the desired ones by clicking on them (Fig. 8.13.6). Close the list by right-clicking or by clicking anywhere outside the box. To see the desired attributes for an edge, the edge must be selected in the view (see step 10, above, for selection details).

13. Open the VizMapper tab in the Control Panel and access VizMapper in one of three ways:

    Select the View→Open VizMapper menu option.
    Select the VizMapper icon in the main button bar.
    Click on the VizMapper tab in the Control Panel at the left of the screen.

    The VizMapper controls how visual properties, such as node or edge color, are assigned from attribute data.

14. Create a new visual style by clicking on the Options button at the top right of the VizMapper tab and selecting the Create new Visual Style . . . option. Enter a name for the new style.

    Once created, visual styles can be modified, saved, and applied to other networks. Alternatively, an existing similar visual style can be copied (using the Copy existing Visual Style . . . option) and then modified, which may take less time than defining a new one.

15. Set the colors of edges in the network to correspond to the type of interactions they represent:

    a. Select the visual attribute by double-clicking the Edge Color entry listed in the Unused Properties section of the Visual Mapping Browser. Edge Color will now appear at the top of the list, under the Edge Visual Mapping Category.

b. Select the network attribute by clicking on the cell to the right of Edge Color and choosing "interaction" from the drop-down list that appears.

c. Select an appropriate Mapping Type according to the data values of the network attribute; in this case, choose Discrete Mapper. All existing attribute values for "interaction" will then be displayed.

Two other attribute mapper types exist in addition to Discrete mappers. A Passthrough mapper directly passes through the attribute value to the visual attribute. This makes most sense for visual attributes such as labels. The other mapper type is the Continuous mapper, which maps a continuous data attribute to a continuous visual attribute, such as mRNA expression values mapped to a node color gradient.

d. Set the mapping relationship. Click the empty cell next to one of the interaction values. Buttons marked "..." (more detail) and "X" (delete) will appear on the right side of the cell. Click on the "..." button and select a color from the color palette. The change will immediately appear on the network.

A different color can be assigned for each value of the network attribute that exists.

This procedure can also be used to map any node data attribute to any node visual property.

Save and export the network

16. Save the network using the Save or Save As ... options in the File menu. This saves the entire Cytoscape session, including the network and all its Node and Edge attributes, as a Cytoscape-specific `.cys` file, which can then be opened for further viewing or editing at a later time.

Cytoscape session files can also be shared with collaborators or as supplementary material for a paper for viewing or editing.

17. Export the network as an image file using File→Export→Network View As Graphics.

A number of standard image types are supported. PDF format is recommended for publication-quality figures. Other options available include exporting the network to a standard interaction data file type for use in other software packages.

18. Exit Cytoscape by selecting File→Quit.

SUPPORT PROTOCOL 1

INSTALLING CYTOSCAPE LOCALLY

This support protocol provides instructions for downloading and installing Cytoscape, along with an introduction to the various components of its user interface.

Necessary Resources

Hardware

Computer with 1 GHz CPU or higher, a high-end graphics card, 60 MB of available hard disk space, at least 512 MB of free physical RAM (for networks up to 5000 edges; at least 1 GB of RAM (for larger networks) and a minimum screen resolution of 1024 × 768 (recommended; requirements depend on the size of the networks to be imported and analyzed)

Internet connection (required to download Java and Cytoscape)

Software

Operating System: Windows, Mac OS X, Linux, or another platform that supports Java

Internet browser: e.g., Microsoft Internet Explorer (*http://www.microsoft.com*), Mozilla Firefox (*http://www.mozilla.org/firefox)*, or Apple Safari (*www.apple.com/safari*)

Files

None required

1. If not already installed, download and install the Java 2 Platform, Standard Edition, version 5.0 or higher (*http://java.sun.com/javase/downloads/index.jsp*).
2. Go to *http://cytoscape.org* and click on the link marked All Releases, then Download Cytoscape 2.5.2 at the top right of the screen.
3. Accept the terms of the Lesser GNU Public License (LGPL), fill in the user registration form, and click the Proceed to Download button.
4. Click on the appropriate installation package to download it, and then double-click on the downloaded icon to start the installation process. Note that the directory in which Cytoscape is installed will be the directory in which Cytoscape initially starts.

   The installation package is roughly 40 MB in size and may take some time to download on slower Internet connections.

5a. *To launch Cytoscape:* Click on the icon created in the Cytoscape installation directory.

5b. *To launch Cytoscape by an alternative means:* Open in Windows by double-clicking `cytoscape.bat` or `cytoscape.jar`, or open in Linux and Mac OS X by running `cytoscape.sh` directly from the command line with various parameters described in the Cytoscape user manual (accessed from the Help menu online at *http://www.cytoscape.org/cgi-bin/moin.cgi/Cytoscape_User_Manual*; included in the Cytoscape installation directory).

6. The Cytoscape desktop will appear (Fig. 8.13.3).
7. To exit Cytoscape, use the File→Quit menu option.

OBTAIN YEAST NETWORK DATA FROM SACCHAROMYCES GENOME DATABASE (SGD)

SUPPORT PROTOCOL 2

The SGD provides physical and genetic interactions for yeast, which may be downloaded as a Cytoscape SIF file (see Table 8.13.1).

Necessary Resources

See Basic Protocol

1. Launch Cytoscape as in Basic Protocol, step 1, and go to *http://db.yeastgenome.org/cgi-bin/batchDownload* and scroll down to the section labeled Step 1: Your Input.
2. Under Enter Feature/Standard Gene names, enter a gene symbol such as `PPA2`.
3. Under Step 2, under the section labeled Other data, check the boxes for physical and genetic interactions and click Submit. The Web browser will be redirected to a page labeled Download Data.
4. Note the SIF filename near the right side of this page at a link labeled Name of Downloadable File, and click on the link to download the `.sif` file.
5. If the file is not automatically uncompressed during download, uncompress it.
6. Continue to Support Protocol 4, step 3 (Load an existing network data file) to import contents of the `.sif` file into Cytoscape.

SUPPORT PROTOCOL 3

OBTAIN NETWORK DATA USING THE cPath DATABASE

Another useful resource for Cytoscape data is the cPath database and Cytoscape plug-in (Cerami et al., 2006). Currently, the Cytoscape cPath plug-in draws data from the MINT (Zanzoni et al., 2002; *UNIT 8.5*) and IntAct (Hermjakob et al., 2004) databases.

Necessary Resources

See Basic Protocol

1. Launch Cytoscape as in Basic Protocol, step 1, and go to File→New→Network→Construct network using cPath . . . A window should appear, as shown in Figure 8.13.7.
2. Select the desired species in the species pull-down menu, which is set to All Organisms by default.
3. In the box labeled Search cPath, enter a gene name (e.g., `p53`) and click on the Search button. Cytoscape will produce a network similar to the one shown in Figure 8.13.8 (shown with the JGraph radial layout).

Figure 8.13.7 The cPath Cytoscape plug-in searches the MINT and IntAct databases to automatically import network data into Cytoscape.

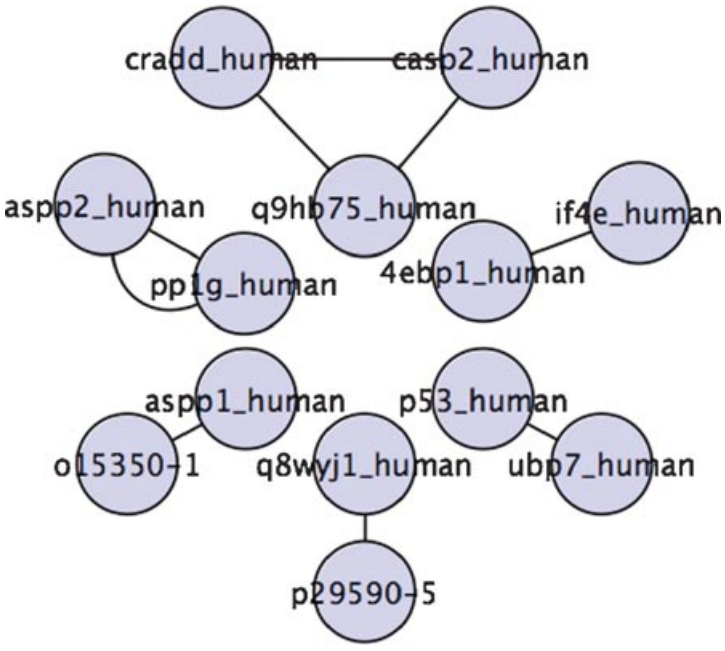


Figure 8.13.8 A sample Cytoscape network created using the cPath plug-in to search for p53 in Homo sapiens. The JGraph radial layout was applied.

The maximum number of records is set to Limit to 10 by default. While the default setting is useful for exploratory queries with a single gene of interest, a larger number of records must typically be retrieved to achieve connectivity between a set of genes of interest. Note that the number of interactions retrieved may be greater than the limit set, because many database records contain more than one protein interaction. In these cases, all proteins in the interaction are connected to each other, up to an internal threshold set in the cPath plug-in.

4. To obtain all interactions for this gene set, select No Limit, remembering that a higher limit will result in a longer download time. The Cytoscape canvas will show a protein interaction network with proteins (nodes) arranged in a grid, connected by retrieved interactions (edges).

   cPath searches can include other attributes, such as diseases (e.g., lymphoma) and biological processes (e.g., apoptosis). Search terms can also be combined using the `AND` *and* `OR` *operations (e.g.,* `p53 AND apoptosis`*).*

5. By default, Cytoscape displays networks with 10,000 or fewer nodes because large networks take a long time to draw. For larger networks, request a view by right-clicking on the network label in the Network Tree Viewer and select Create View from the pop-up menu.

OBTAIN A BIOLOGICAL PATHWAY FROM THE REACTOME DATABASE

SUPPORT PROTOCOL 4

The Reactome database (*UNIT 8.7*; Joshi-Tope et al., 2005) is a biological pathway database containing curated human information, along with inferred orthologous pathways in a number of other species. It provides pathways in a number of formats (see Table 8.13.1), including BioPAX (*http://www.biopax.org*).

Necessary Resources

See Basic Protocol

1. Launch Cytoscape as in the Basic Protocol, step 1, and go to the Reactome home page at *http://reactome.org*. The default species displayed in the reaction map is *Homo sapiens*.
2. Click on the drop-down list immediately above the map to change species if necessary.
3. Select a pathway by clicking on its image in the reaction map or the labels underneath. A summary page will appear.
4. Scroll down to the bottom and click the link marked [BioPAX] to download a Reactome file (extension `.owl`) containing the pathway data.
5. Continue to Support Protocol 5 (Load an existing network data file) to import the `.owl` file.

LOAD AN EXISTING NETWORK DATA FILE

SUPPORT PROTOCOL 5

This protocol provides different procedures for a number of file formats.

Necessary Resources

See Basic Protocol

1. Launch Cytoscape as in Basic Protocol, step 1.

2a. *To open a Cytoscape session file (*`.cys`*):* Go to File→Open. Select the session file and click Open.

2b. *To open a text or Excel file:* Go to File→Import. . .→Network from Table (Text/MS Excel). Select the appropriate file using the Select File button, and define the importing options and data columns, with the help of the preview at the bottom of the dialog box.

Since free-format tables contain user-defined columns instead of a standard format, a preview window is provided to indicate how Cytoscape will interpret the input data. Once a file is selected, the first few lines of the file contents will be shown.

If Cytoscape is not parsing files correctly, it may be necessary to change the advanced settings. Check the box marked Show Text File Import Options to display these settings.

Drop-down menu lists are available for specifying the columns containing Source Nodes (purple), Interaction/Edge Type, (red), and Target Nodes (orange). The preview will color-code each column accordingly; blue is used to indicate columns that will be interpreted as edge attributes.

Note that node attributes must be imported separately. Any data columns that are not to be loaded into Cytoscape can be disabled by clicking on the header (Column X, where X is the column number) in the preview. A Reload button is provided to refresh the preview after making any changes (Fig. 8.13.9).

Edge attributes can be subdelimited within a column by right-clicking on the column header and selecting the List option as the Attribute Data Type. For example, this might be used to indicate PubMed records relevant to the interactions. Select or enter the appropriate List Delimiter and click OK. Note that this sub-delimiter must be different from the delimiter used to separate columns.

For Excel users: Only single-sheet workbooks are currently supported.

2c. *To open a different supported network file type (SIF, GML, XGMML, SBML, PSI-MI, BioPAX; see Table 8.13.1)*: Go to File→Import→Network (Multiple File Types). Select the appropriate file and click Open.

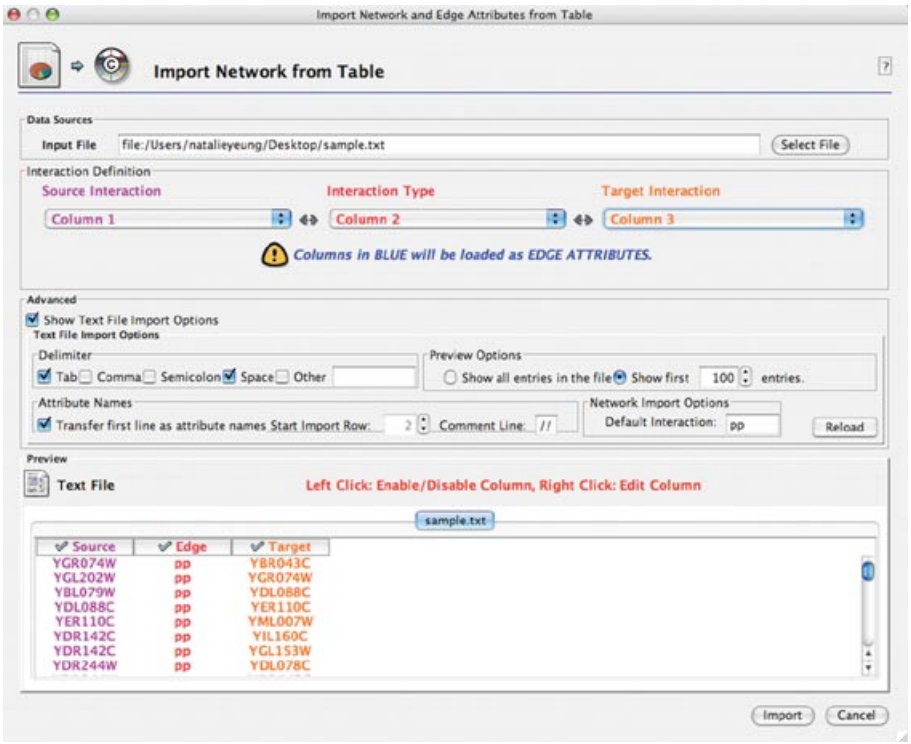


Figure 8.13.9 The window that appears when importing an Excel or delimited text network file.

2d. *To open files from the local hard drive:* Select Local Data Source Type (this is the default) and choose the file using the Select button. Selecting Remote Data Source Type allows files to be loaded from the Internet by typing in the URL or using Cytoscape bookmarks.

The directory in which Cytoscape is installed contains a folder called sampleData. This folder holds a number of example files containing published experimental data. References for these data are available in the Cytoscape user manual (accessed from the Help menu online at http://www.cytoscape.org/cgi-bin/moin.cgi/Cytoscape_User_Manual; included in the Cytoscape installation directory).

INTEGRATE EXPRESSION DATA

ALTERNATE PROTOCOL

Cytoscape offers the ability to combine network data with expression data, which can provide information about network dynamics over time or across different experimental conditions. This alternate protocol outlines the process of loading expression data and then visualizing it on an existing network.

In order to import attribute files or expression data into Cytoscape, the gene or protein identifier in the file must exactly match the corresponding Cytoscape node ID (or other Cytoscape attribute that has been previously loaded). If no matching identifiers are present, additional identifiers can be created using external online ID mapping services such as Synergizer (*http://llama.med.harvard.edu/cgi/synergizer/translate*), provided by the Roth laboratory at Harvard University.

Necessary Resources *(also see Basic Protocol)*

Files

Network files, downloaded (see Basic Protocol, step 3)
Expression data files, created locally: currently supported expression data file formats include Excel spreadsheets and delimited text (tab, comma, or space delimiters), along with standard file extensions such as `.mrna` and `.pvals` (see Figs. 8.13.10 and 8.13.11; also see the Expression Data chapter of the Cytoscape user manual)

NOTE: To use this protocol as a tutorial, go to the Cytoscape/sampleData folder to select `galFiltered.sif` as the network file and `galExpData.pvals` as the expression data file.

1. Launch Cytoscape, then load and layout a network (see the Basic Protocol, steps 1 to 8).

For standard file formats

2a. Load an expression data file by going to the drop-down list and selecting the attribute that is the same in both the network and expression data files. Click Import.

3a. A status window will appear showing the number of experimental conditions found and information on significance values (if found in the file). Click the Close button.

For nonstandard file formats (e.g., text and Excel)

2b. Load an expression data file by using the File→Import→Attribute from Table (text/MS Excel . . . option).

3b. This will pull up a window similar in operation to the one used to import text and Excel network files (see Basic Protocol, step 3d). Be sure that the values in the column labeled Key (blue) exactly match those of a column in the network file. Click the Close button.

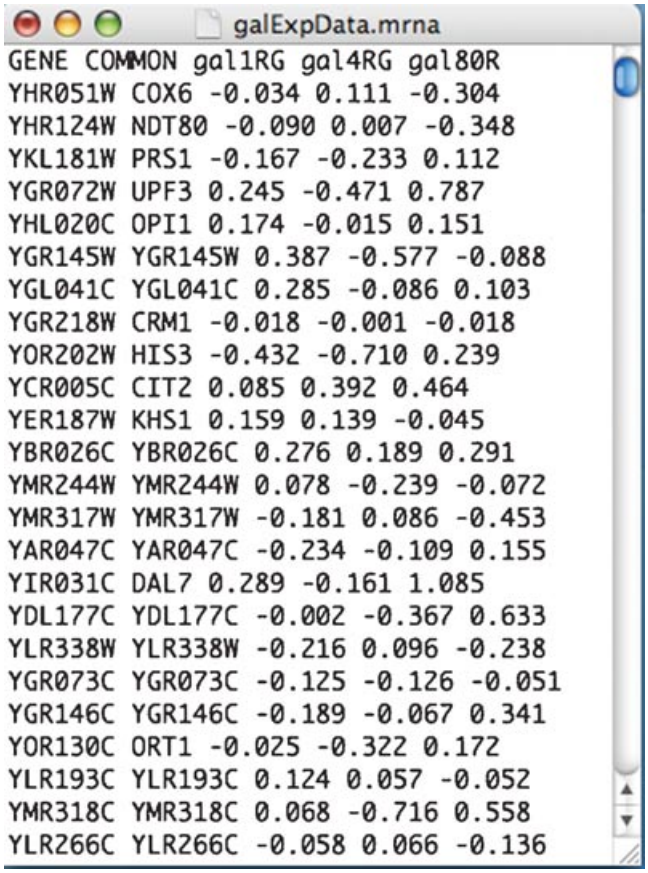


Figure 8.13.10 The first few lines of `galExpData.mrna`, a sample expression data file. The first row is a header row. The first column contains gene names, and the second has the common names for each gene, followed by expression level data from three experimental conditions. The first column is mapped to the node IDs in the network unless otherwise specified.

4. View the expression data by going to the Node Attribute Browser tab in the Data Panel and displaying the experimental conditions of interest (see Basic Protocol, step 9).
5. Open the VizMapper and copy the default visual style (see Basic Protocol, steps 12 to 15).
6. Define a node color gradient that corresponds to experimental expression data to create multiple mappings for visualizing multiple data attributes (see Basic Protocol, steps 12 to 15, for more detail):
   a. Select Node Color.
   b. Define the Map Attribute value as one of the experimental conditions (e.g., `Gal80RGexp` in the `galExpData.pvals` sample file).
   c. Select Continuous Mapping as the Mapping Type.
   d. Double-click on the white rectangle next to Graphical View to open the Color Gradient Mapper. This dialog is used to define the points where colors will change.
   e. Click the Add button twice to create the first two boundary points. Additional clicks will add boundary points that will show up as overlapping triangles at the right of the scale.

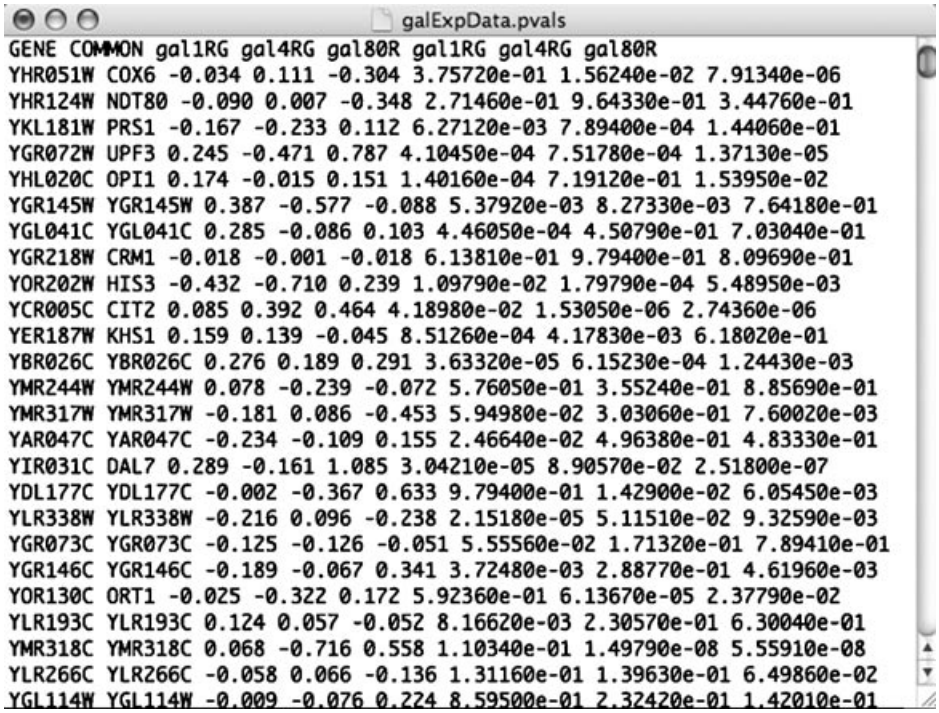


Figure 8.13.11 The first few lines of `galExpData.pvals`, an expression data file included in the `Cytoscape/sampleData directory`. The first row is a header row. The first column contains gene names, and the second has the common names for each gene. The next three columns contain expression level data from three experimental conditions. The last three columns contain the significance or *p*-values associated with each piece of experimental data. Note that the *p*-value columns must contain exactly the same headers in the same order as the data columns in order for Cytoscape to associate the *p*-values with the data.

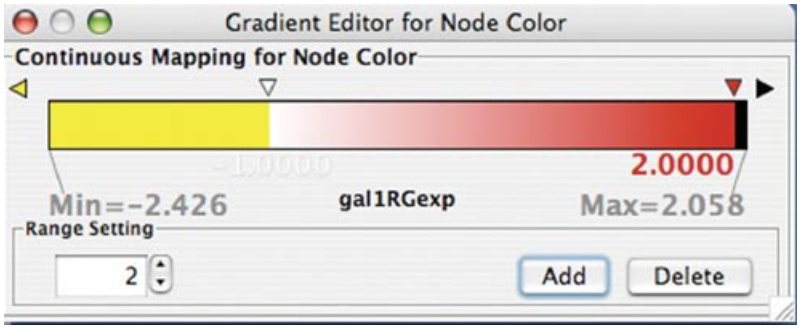


Figure 8.13.12 An example color gradient created to map node color to the `gal80RGexp` condition in the `galExpData.pvals` sample file.

f. Click and drag each triangle to define the boundaries between colors on the scale, or type the desired value in the Range Setting box. The values shown on the scale correspond to the existing values for the experimental condition that has been chosen. To delete boundary points, use the Delete button, but note that at least two boundary points must exist (not including the two default extremes) in order to create a gradient.

g. Define the color gradients between boundary points by double-clicking on the triangles at each of the endpoints, in turn, to open a color palette. Select a color

and click OK. Nodes with expression levels of this value will be colored with the selected color.

Nodes that have expression levels in between the two boundary points will be rendered with a color in between the two boundary colors. For example, if the lower boundary point is white and the upper boundary point is red, nodes with expression values in between the two points will be colored pink, with darker shades of pink indicating higher expression values. When the boundary colors are set, the colors of the nodes will be updated immediately (see Fig. 8.13.12).

GUIDELINES FOR UNDERSTANDING RESULTS

The protocols provided here can stand alone as methods for analyzing biological networks and also serve as a starting point for more in-depth analysis using various Cytoscape analysis plug-ins. Plug-ins can be downloaded for use directly from Cytoscape (via Plug-in → Manage Plug-ins) or online at *http://cytoscape.org/plugins2.php*.

The Basic Protocol, which produces a two-dimensional network, can be used to infer certain biological properties based on topology. For instance, critical genes and proteins tend to be hubs (nodes connected to many other nodes) or part of the shortest path through the network between two other nodes (Yu et al., 2007). Plug-ins such as PeSca and ShortestPath implement shortest path algorithms for use in Cytoscape. Additional plug-ins are available for creating networks, e.g., the Agilent Literature Search plug-in, which extracts relationships about given genes or proteins automatically from multiple online sources, including PubMed (Vailaya et al., 2005).

Certain network data formats include explicit nodes denoting modules or complexes, e.g., the BioPAX Reactome networks. For networks without this information represented, it is possible to infer complexes by searching for groups of nodes with a high degree of internal connectivity (interactions amongst themselves) compared to external connectivity (interactions with nodes outside the group). Putative complexes can be identified visually, or automatically using Cytoscape's MCODE plug-in (Bader and Hogue, 2003).

The Alternate Protocol superimposes expression data on a network, which can result in some interesting biological insights. Combining expression and interaction data is a procedure sometimes performed to find causative disease agents when comparing control and case samples for clinical studies. While the causative agents might not exhibit dramatic expression changes themselves, one can often see significant and coordinated variation of expression (co-expression) in genes regulated by the causative agents. Using the network as a visual aid to find common neighbors of co-expressed genes is therefore an effective method of finding possible causative agents. This process can be automated by the Active Modules plug-in, which finds active regions of a network across multiple experimental molecular profile measurements (Ideker et al., 2002).

More generally, a plausible biological explanation for co-expression of genes or proteins is functional relatedness. This is especially true in prokaryotes, where functionally-related genes may be organized into the same operons in the genome. Genes involved in a complex can exhibit just-in-time assembly, where one highly regulated critical gene controls the overall activity of the entire complex (de Lichtenberg et al., 2005). Comparing different expression patterns across experimental conditions can also reveal different mechanisms that cause the same end result. The BiNGO plug-in finds significantly over-represented Gene Ontology terms annotated to the genes of interest. This helps identify functions enriched in a set of genes, including sets of genes that are co-expressed (Maere et al., 2005). Additional Cytoscape tutorials explaining these uses in more detail are available from the Cytoscape Web site (*http://www.cytoscape.org*).

COMMENTARY

Background Information

Biological network visualization is an important tool in systems biology. While traditional reductionist biology focuses on a single gene or protein, systems biology focuses on the interplay of multiple genes or proteins: how they form regulated subsystems and how changes in experimental conditions affect subsystem behavior. While systems biology can include mathematical modeling of network dynamics, network visualization is arguably the most common method of modeling systems; it does not require detailed measurement of subsystem dynamics and can suggest information about gene function and impact of gene loss or transcriptional repression. A typical biological pathway presents enough complexity that it is difficult for the human mind to process new observations in the context of the whole pathway. Visualization offers a straightforward mechanism to assess the new observations and existing data together.

Biological network data comes in two major forms: curated pathways and interaction networks. Curated pathways describe sequences of intermolecular interactions that yield some measurable result. Examples include converting organic compounds into energy (*metabolic pathways*); transmitting an extracellular signal into the nucleus, resulting in transcription (*signal transduction pathways*); or transcribing a set of genes after production of the necessary transcription factors (*regulatory networks*). Curated pathway repositories contain descriptions of pathways, derived from a combination of the literature and experimental verification. Major pathway repositories include KEGG (Wixon and Kell, 2000), Reactome (Joshi-Tope et al., 2005), and BioCyc (Krummenacker et al., 2005); additional repositories are listed in Pathguide (Bader et al., 2006). These are rich sources of information, describing the context and consequences of each interaction, but they are limited in coverage. In general, they describe basic cellular processes that are highly conserved between organisms and certain processes involved in well studied diseases. In other instances, where the data is sparser, protein interaction networks can be a useful alternative.

Protein interaction networks contain nodes representing proteins and edges representing experimentally measured interactions between the proteins. Interactions are potential associations; they may occur in a cell if the proteins are both present and in the correct modification states. Major interaction data repositories include IntAct (Hermjakob et al., 2004), MINT (*UNIT 8.5*; Zanzoni et al., 2002), BIND (*UNIT 8.9*; Bader et al., 2001), DIP (Xenarios et al., 2002), and HPRD (Peri et al., 2004); additional repositories are listed in Pathguide (Bader et al., 2006).

The most common type of interaction data is measured using the yeast two-hybrid method, a genetic technique for detecting pairs of proteins that can interact. This technique has been adapted for high-throughput use and now represents the majority of interaction data (Hermjakob et al., 2004).

Other interaction data comes from biochemical purification experiments (e.g., co-immunopreciptation, pull-down, and tap-tagging assays), which have also been used in high-throughput studies. While these assays report interactions that may occur in the cell, they do not report which of the proteins were in direct physical contact. Rather, they find a set of proteins that likely represent a population of complexes. Thus, for such data, interaction is interpreted as membership in the same complex. The contrast between these two types of interaction data illustrate why different types of interactions demand slightly different interpretation. Thus, when analyzing interaction networks, it is useful to distinguish the varying interaction types.

Another element of this protocol is coloring nodes according to expression data. First of all, this provides a visual indication of what portions of the network might be produced, indicating where an interaction might occur in a protein interaction network or where there might be a missing element in a pathway. Expression data can provide further information on network dynamics. For example, when several genes are part of the same complex, the complex might not be active until all genes are expressed (de Lichtenberg et al., 2005). Finally, there are cases where functionally-related proteins are produced from co-expressed genes. Prokaryotic genomes contain operons, sections of DNA that contain genes and are transcribed together as a unit, and genes in the same operon tend to be functionally-related. Yet even in eukaryotes, genes that are co-expressed in multiple species and experimental contexts tend to be functionally-related (Stuart et al., 2003).

Altogether, biological network visualization is highly useful for integrating multiple data types in the context of known biological

processes. While biological network visualization has been discussed in this unit, Cytoscape is capable of handling any type of network. As long as the data can be represented as sets of nodes and edges, Cytoscape can display the data as a network. For example, the StructureViz Cytoscape plug-in allows the user to compare related protein structures under the Chimera protein structure viewer, while a Cytoscape network relates the protein structure(s) to others in the same structural family (Morris et al., 2007).

Critical Parameters and Troubleshooting

Out of memory errors

Symptoms: Cytoscape behaves strangely. Java null pointer exception error messages may appear, or there will be no reported error but the expected action does not occur.

Possible causes: This type of problem will occur when Cytoscape tries to analyze very large networks or when a number of other applications are also running on the computer.

Remedies: Make more memory available to Cytoscape by closing unnecessary networks and applications, rebooting the computer, or increasing Cytoscape's memory allocation on the computer (see *http://cytoscape.org/cgi-bin/moin.cgi/How_to_increase_memory_for_Cytoscape* for details).

Data integration errors

Symptom: Expression or attribute data files are not properly integrated with the loaded network.

Possible causes: The gene identifier columns that synchronize the two files do not match exactly, or the files may not be in the correct format.

Remedies: Use the Node or Edge Attribute tabs (see Basic Protocol, step 12) to check that the network identifiers exactly match the identifiers in the expression or attribute data file. To determine the correct format of an attribute or expression file, see the Web sites provided in Table 8.13.1.

Large networks

Symptoms: The network loads without an automatically generated view, or the dataset is so large that effective analysis is difficult.

Cause: The loaded network is very large.

Remedies: Cytoscape can create views for large networks (see Basic Protocol, step 7), and child networks can also be created (see Basic Protocol, step 9) to create a smaller and more manageable network.

Acknowledgments

Cytoscape is developed through an ongoing collaboration between the University of California at San Diego, the University of Toronto, the Institute for Systems Biology, Memorial Sloan-Kettering Cancer Center, Institut Pasteur, Agilent Technologies, and the University of California at San Francisco. We gratefully acknowledge the contributions of many Cytoscape developers: Nada Amin, Mark Anderson, Iliana Avila-Campilo, Richard Bonneau, Ethan Cerami, Rowan Christmas, Michael Creech, Benjamin Gross, Kristina Hanspers, Larissa Kamenkovich, Ryan Kelley, Sarah Killcoyne, Nerius Landys, Samad Lotia, Andrew Markiel, John Morris, Keiichiro Ono, Owen Ozier, Alexander R. Pico, Paul Shannon, Robert Sheridan, Aditya Vailaya, Jonathan Wang, Peng-Liang Wang, Chris Workman, and principal investigators Annette Adler, Bruce R. Conklin, Leroy Hood, Trey Ideker, Chris Sander, Ilya Schmulevich, Benno Schwikowski, and Guy J. Warner.

Many research groups have developed plug-ins to Cytoscape and provided them for download free of charge from *http://www.cytoscape.org*. These plug-ins represent key contributions to the overall utility of Cytoscape, and we gratefully thank the authors for their contributions. Thanks to Vuk Pavlovic for editing help.

Funding for Cytoscape is provided by the U.S. National Institute of General Medical Sciences of the National Institutes of Health under award number GM070743-01. Corporate funding is provided through a contract from Unilever PLC. Cytoscape contributions by G.D.B. were funded in part by Genome Canada through the Ontario Genomics Institute.

Literature Cited

Bader, G.D. and Hogue, C.W. 2003. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics* 4:2 (*http://www.biomedcentral.com/1471-2105/4/2*).

Bader, G., Donaldson, I., Wolting, C., Ouellette, B., Pawson, T., and Hogue, C. 2001. BIND—The Biomolecular Interaction Network Database. *Nucleic Acids Res.* 29:242-245.

Bader, G.D., Cary, M.P., and Sander, C. 2006. Pathguide: A pathway resource list. *Nucleic Acids Res.* 34:D504- D506.

Cerami, E.G., Bader, G.D., Gross, B.E., and Sander, C. 2006. cPath: Open source software for collecting, storing, and querying biological pathways. *BMC Bioinformatics* 7:497 (*http://www.biomedcentral.com/1471-2105/7/497*).

Christie, K.R., Weng, S., Balakrishnan, R., Costanzo, M.C., Dolinski, K., Dwight, S.S., Engel, S.R., Feierbach, B., Fisk, D.G., Hirschman, J.E., Hong, E.L., Issel-Tarver, L., Nash, R., Sethuraman, A., Starr, B., Theesfeld, C.L., Andrada, R., Binkley, G., Dong, Q., Lane, C., Schroeder, M., Botstein, D., and Cherry, J.M. 2004. Saccharomyces Genome Database (SGD) provides tools to identify and analyze sequences from *Saccharomyces cerevisiae* and related sequences from other organisms. *Nucleic Acids Res.* 32:D311-D314.

de Lichtenberg, U., Jensen, L.J., Brunak, S., and Bork, P. 2005. Dynamic complex formation during the yeast cell cycle. *Science* 307:724-727.

Garcia, O., Saveanu, C., Cline, M., Fromont-Racine, M., Jacquier, A., Schwikowski, B., and Aittokallio, T. 2007. GOlorize: A Cytoscape plug-in for network visualization with Gene Ontology-based layout and coloring. *Bioinformatics* 23:394-396.

Hermjakob, H., Montecchi-Palazzi, L., Lewington, C., Mudali, S., Kerrien, S., Orchard, S., Vingron, M., Roechert, B., Roepstorff, P., Valencia, A., Margalit, H., Armstrong, J., Bairoch, A., Cesareni, G., Sherman, D., and Apweiler, R. 2004. IntAct: An open source molecular interaction database. *Nucleic Acids Res.* 32:D452-D455.

Ideker, T., Ozier, O., Schwikowski, B., and Siegel, A.F. 2002. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics* 18:S233-S240.

Joshi-Tope, G., Gillespie, M., Vastrik, I., D'Eustachio, P., Schmidt, E., de Bono, B., Jassal, B., Gopinath, G.R., Wu, G.R., Matthews, L., Lewis, S., Birney, E., and Stein, L. 2005. Reactome: A knowledgebase of biological pathways. *Nucleic Acids Res.* 33:D428-D432.

Krummenacker, M., Paley, S., Mueller, L., Yan, T., and Karp, P.D. 2005. Querying and computing with BioCyc databases. *Bioinformatics* 21:3454-3455.

Maere, S., Heymans, K., and Kuiper, M. 2005. BiNGO: A Cytoscape plug-in to assess overrepresentation of gene ontology categories in biological networks. *Bioinformatics* 21:3448-3449.

Morris, J.H., Huang, C.C., Babbitt, P.C., and Ferrin, T.E. 2007. structureViz: Linking Cytoscape and UCSF Chimera. *Bioinformatics* 23:2345-2347.

Peri, S., Navarro, J.D., Kristiansen, T.Z., Amanchy, R., Surendranath, V., Muthusamy, B., Gandhi, T.K., Chandrika, K.N., Deshpande, N., Suresh, S., et al. 2004. Human protein reference database as a discovery resource for proteomics. *Nucleic Acids Res.* 32:D497-D501.

Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., and Ideker, T. 2003. Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Res.* 13:2498-2504.

Stuart, J.M., Segal, E., Koller, D., and Kim, S.K. 2003. A gene-coexpression network for global discovery of conserved genetic modules. *Science* 302:249-255.

Vailaya, A., Bluvas, P., Kincaid, R., Kuchinsky, A., Creech, M., and Adler, A. 2005. An architecture for biological information extraction and representation. *Bioinformatics* 21:430-438.

Wixon, J. and Kell, D. 2000. The Kyoto encyclopedia of genes and genomes—KEGG. *Yeast* 17:48-55.

Xenarios, I., Salwínski, L., Duan, X., Higney, P., Kim, S.M., and Eisenberg, D. 2002. DIP, the Database of Interacting Proteins: A research tool for studying cellular networks of protein interactions. *Nucleic Acids Res.* 30:303-305.

Yu, H., Kim, P.M., Sprecher, E., Trifonov, V., and Gerstein, M. 2007. The importance of bottlenecks in protein networks: Correlation with gene essentiality and expression dynamics. *PLoS Comput. Biol.* 3:e59 (*http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.0030059*).

Zanzoni, A., Montecchi-Palazzi, L., Quondam, M., Ausiello, G., Helmer-Citterich, M., and Cesareni, G. 2002. MINT: A Molecular INTeraction database. *FEBS Lett.* 513:135-140.

Key References

Bader et al., 2006. See above.

Pathguide provides an extensive list of electronic pathway resources, both public and private, along with references and URLs for each.

Shannon et al., 2003. See above.

This article provides further background on Cytoscape, and the questions that it was first developed to address.

Internet Resources

http://www.cytoscape.org

The home page of the Cytoscape project contains download links, the latest manual, plug-ins, online tutorials, and links to the Cytoscape discussion forums and project development wiki.

http://java.sun.com

This is the central Internet resource for Sun Java, with download links, documentation, and software development packages. Java must be installed for Cytoscape to run. Most computers already have Java installed.

http://www.yeastgenome.org

The Saccharomyces Genome Database, available at this site, contains a wealth of yeast genomic and experimental data, tools, and resources for the study of yeast; in particular, the SGD maintains a large database of yeast interaction data, and provides this data in formats including Cytoscape SIF format.

http://www.reactome.org

Reactome provides curated pathway data for many of the key pathways in humans, and over twenty other organisms.

http://llama.med.harvard.edu/cgi/synergizer/translate

The Synergizer offers an effective, usable solution to one of the most frequent and frustrating problems in computational molecular biology: identifier mapping.

Using AutoDock for Ligand-Receptor Docking

Garrett M. Morris,[1] Ruth Huey,[1] and Arthur J. Olson[1]

[1]The Scripps Research Institute, La Jolla, California

ABSTRACT

This unit describes how to set up and analyze ligand-protein docking calculations using AutoDock and the graphical user interface, AutoDockTools (ADT). The AutoDock scoring function is a subset of the AMBER force field that treats molecules using the United Atom model. The unit uses an X-ray crystal structure of Indinavir bound to HIV-1 protease taken from the Protein Data Bank (*UNIT 1.9*) and shows how to prepare the ligand and receptor for AutoGrid, which computes grid maps needed by AutoDock. Indinavir is prepared for AutoDock, adding the polar hydrogens, and partial charges, and defining the rotatable bonds that will be explored during the docking. The input files for AutoGrid and AutoDock are created, and then the grid map calculation run, followed by the docking calculation in AutoDock. Finally, this unit describes some of the ways the results can be analyzed using AutoDockTools. *Curr. Protoc. Bioinform.* 24:8.14.1-8.14.40. © 2008 by John Wiley & Sons, Inc.

Keywords: AutoDock • protein-ligand docking • virtual screening • computer-aided drug design

INTRODUCTION

This unit introduces ligand-protein docking simulations, using the AutoDock suite of programs (Goodsell and Olson, 1990; Morris et al., 1996, 1998; Huey et al., 2007). It will explain how to use the graphical user interface, AutoDockTools (ADT), which helps a user to set up the two molecules for docking, launches the calculations in AutoGrid and AutoDock, and, when the dockings are completed, also lets the user visualize the docked conformations of the ligand-protein complexes interactively in three dimensions.

The example in this unit uses X-ray crystal structure data for Indinavir (protease inhibitor) bound to HIV-1 protease (Chen et al., 1994), taken from the Protein Data Bank (PDB; *UNIT 1.9*; Berman et al. 2000), to compute atomic affinity grid maps using AutoGrid, and it explains how to set up and carry out the virtual docking experiments. Basic Protocol 1 addresses downloading and installing the programs necessary for performing the virtual experiments and analyzing the results. The next set of protocols (Basic Protocol 2, 3, 4, and 5) describes how to prepare molecular data for the macromolecules and ligands to be studied. Additional sets of protocols address setting grid parameters (Basic Protocol 6), setting up and running the docking simulations (Basic Protocols 7, 8, and 9), and visualizing and analyzing the results from AutoDock using ADT (Basic Protocols 10, 11, 12, and 13). The Guidelines for Understanding Results section discusses some ways to assess the quality of the docking results.

NOTE: Some user-created video screencasts showing how to use AutoDock with ADT are also available (see *http://youtube.com/results?search_query=autodock*).

Current Protocols in Bioinformatics 8.14.1-8.14.40, December 2008
Published online December 2008 in Wiley Interscience (www.interscience.wiley.com).
DOI: 10.1002/0471250953.bi0814s24

DOWNLOAD AND INSTALL AutoDock, AutoGrid, AND AutoDockTools

Before it is possible to simulate molecular docking using AutoDock, it is necessary to obtain both the software to prepare and analyze AutoDock dockings and the docking software itself. The AutoDock and AutoGrid programs are distributed together. AutoDock performs the docking, but in order to speed up the interaction energy calculation, it requires grid maps that describe the field of interaction energies around the macromolecular target of interest. These maps are precomputed using AutoGrid and can be reused for any number of dockings. There is also a graphical user interface (GUI) called AutoDockTools, which is distributed separately. This protocol describes the necessary hardware and how to obtain the software to run AutoDock dockings.

Necessary Resources

Hardware

Computer with Internet access

Platforms (operating systems running on a specific chip architecture) including Darwin and Mac OS X running on PowerPC G3, G4, G5 and Intel Core processors; Linux running on AMD, Intel x86 and Itanium processors; IRIX running on Silicon Graphics MIPS, and Solaris running on Sun Sparc; support for Microsoft Windows possible, by running Cygwin (Linux-like environment; freely available from *http://www.cygwin.com*); full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock and AutoGrid: available free of charge to academic and government institutions for noncommercial use under the GNU General Public License from The Scripps Research Institute's Molecular Graphics Laboratory (MGL; *http://autodock.scripps.edu/downloads*), source code included

AutoDockTools (*http://autodock.scripps.edu/resources/adt*)

Up-to-date Internet browser, e.g., Internet Explorer (*http://www.microsoft.com/ie*), Netscape (*http://browser.netscape.com*), Firefox (*http://www.mozilla.org/firefox*), or Safari (*http://www.apple.com/safari*)

Obtain the software

1. Point the browser to *http://autodock.scripps.edu/downloads*.
2. Click on "registration form" next to AutoDock 4 to go to the registration page.
3. Fill in the registration form and click on "submit". This navigates to the download page. Select the platform and or/source code.
4. Download the distribution and uncompress the files. Read the `README` files in each directory.

   The `README` *files contain important installation instructions and information.*

5. Point the browser to *http://autodock.scripps.edu/resources/adt* for instructions on how to download and install AutoDockTools (ADT).

Download input files

The files necessary to complete this protocol are available for any kind of hardware and can be obtained as follows.

6. Point the browser to *http://autodock.scripps.edu/faqs-helphelp-center/tutorial/using-autodock-with-autodocktools*.
7. Click on the link labeled "Input files and results files for the AutoDock 4 tutorial [tutorial4.tar.gz, 5.5 MB]".

8. If the Web browser does not automatically uncompress the downloaded file, type the following UNIX commands in a terminal window:

```
gunzip tutorial4.tar.gz
tar xvf tutorial4.tar
```

9. Copy the input files into the current directory by typing:

```
cp tutorial4/*.pdb
```

PREPARING THE STRUCTURES (BASIC PROTOCOLS 2, 3, 4, AND 5)

As in all other realms of computation, the quality of the results of a simulation will depend on the quality of the inputs. In molecular docking, the structure of the target macromolecule is required, as is the structure of the small molecule (or ligand); molecular docking predicts how the small molecule will be most likely to bind to the macromolecule. Usually, these macromolecular structures come from X-ray crystallography, although nuclear magnetic resonance (NMR) is sometimes used. The small molecule structures can be obtained from X-ray crystallography, but can also be computed using a variety of computational methods.

The scoring function in AutoDock is based on the United Atom version of the AMBER force field, in which nonpolar hydrogen atoms are removed to reduce the number of atoms to be simulated, and the van der Waals' radius of the heavy atom to which they are connected is increased accordingly, along with the appropriate modification of its partial charge to preserve the original total charge. This means that both the ligand and the target macromolecule, usually a protein, must be modeled with explicit polar hydrogen atoms; the nonpolar hydrogen atoms are treated implicitly (i.e., described by the larger van der Waals radius of the heavy atoms to which they had been attached). Since AutoDock computes the electrostatic interaction energy, it also means that both molecules will require partial atomic charges to be assigned to all their atoms. In order to estimate the free energy change of solvation upon binding, AutoDock uses a method based on atomic solvation parameters. These, too, need to be assigned to all the interacting atoms, and they are looked up based on the AutoDock atom types that must be supplied in the input structures' files. This is accomplished using PDBQT-formatted files (unique to AutoDock 4 and AutoGrid 4 and very similar to PDB format). The PDBQT-formatted files store partial charges (hence the Q—conventionally used to represent partial atomic charge—in their name) and atom types (the T in PDBQT). Basic Protocols 2, 3, and 4 explain how to prepare the PDBQT files of macromolecule and the ligand necessary for docking. One of the new features in AutoDock 4 is the ability to allow user-defined side chains in the receptor to change conformation during the docking. Basic Protocol 5 describes how to select these flexible side chains and set up the required input files for AutoGrid and AutoDock calculations.

Preparing the Macromolecule

BASIC PROTOCOL 2

The first place to look for macromolecular structures is the Protein Data Bank (PDB; see *UNIT 1.9*), but sometimes these structures may have a variety of potential problems that need to be corrected before they can be used in AutoGrid and AutoDock. These potential problems include, e.g., missing side-chain atoms, added waters, more than one molecule, chain breaks, or disordered atoms with alternate locations.

AutoDockTools (ADT) is part of MGLTools, from the Molecular Graphics Laboratory at The Scripps Research Institute, is built on the Python Molecule Viewer (PMV), and has an evolving set of tools designed to solve these kinds of problems. In particular, two modules, editCommands and repairCommands, permit the addition or deletion of hydrogens, repair of incomplete residue side chains by adding missing atoms, modification of histidine

protonation, and modification of the protonation of intra-chain breaks, among many other useful tools.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

`hsg1.pdb`: Protein Data Bank (PDB; see *UNIT 1.9*) file for X-ray crystal structure data for HIV-1 protease

Load the molecule

1a. *To start AutoDockTools on PC operating systems:* Double-click on the AutoDockTools icon. This will start the AutoDockTools GUI.

1b. *To start AutoDockTools on Mac OS X:* Open the Applications folder and double-click on the AutoDockTools icon. This will start the AutoDockTools GUI.

2. Click on File > Read Molecule. This will open a file browser, showing all the files in the current directory (Fig. 8.14.1).

3. Select `hsg1.pdb` and click on Open. Alternatively, press the Enter key on the keyboard while the cursor is still in the entry. This loads structural data for a molecule named hsg1 into ADT.

The alternate way of opening the file can be used for many parts of the GUI in ADT.

The bonds between bonded atoms are represented as lines, while nonbonded atoms, e.g., metal ions and oxygen atoms of water molecules, are shown as small squares. The nonbonded atoms in hsg1 are the oxygen atoms of water molecules that were present in the crystal structure. These waters will be removed later.

Remove water molecules and add hydrogens

4. Use the mouse buttons on a three-button mouse alone or with a modifier key to modify the view of the molecules in the three-dimensional (3D) viewer.

   a. To zoom in or out, press and hold down the Shift key and then click and drag with the middle mouse button.

   b. To rotate the molecule, just click and drag with the middle mouse button.

Under Mac OS X, the Option key can be used instead of the middle mouse button, while the right mouse button can be emulated by using the Command key. For all of the possible combinations, see Table 8.14.1.

It is also possible to press certain keys in the 3D viewer window to change the view of the molecule. See Table 8.14.2 for more details.

5. Click on the Color > by Atom Type menu item. Click on All Geometries, and then click OK (Fig. 8.14.2). All of the molecules will be colored according to the chemical element, as follows:

Carbons that are aliphatic (C) – white
Carbons that are aromatic (A) – green
Nitrogens (N) – blue
Oxygens (O) – red
Sulfurs (S) – yellow
Hydrogens (H) - cyan.

Table 8.14.1 Mouse Button and Modifier Keys for Manipulating Objects in ADT When the Cursor is Over the 3D Viewer[a]

Modifier key	Left	Middle (Option/Alt)	Right (Command/⌘)
None	Pick	Rotate	Translate left/right (x) and up/down (y)
Shift	None	Scale or zoom	Translate in/out (z)

[a]The keyboard equivalents to emulate the middle and right mouse buttons in Mac OS X are shown in parentheses.

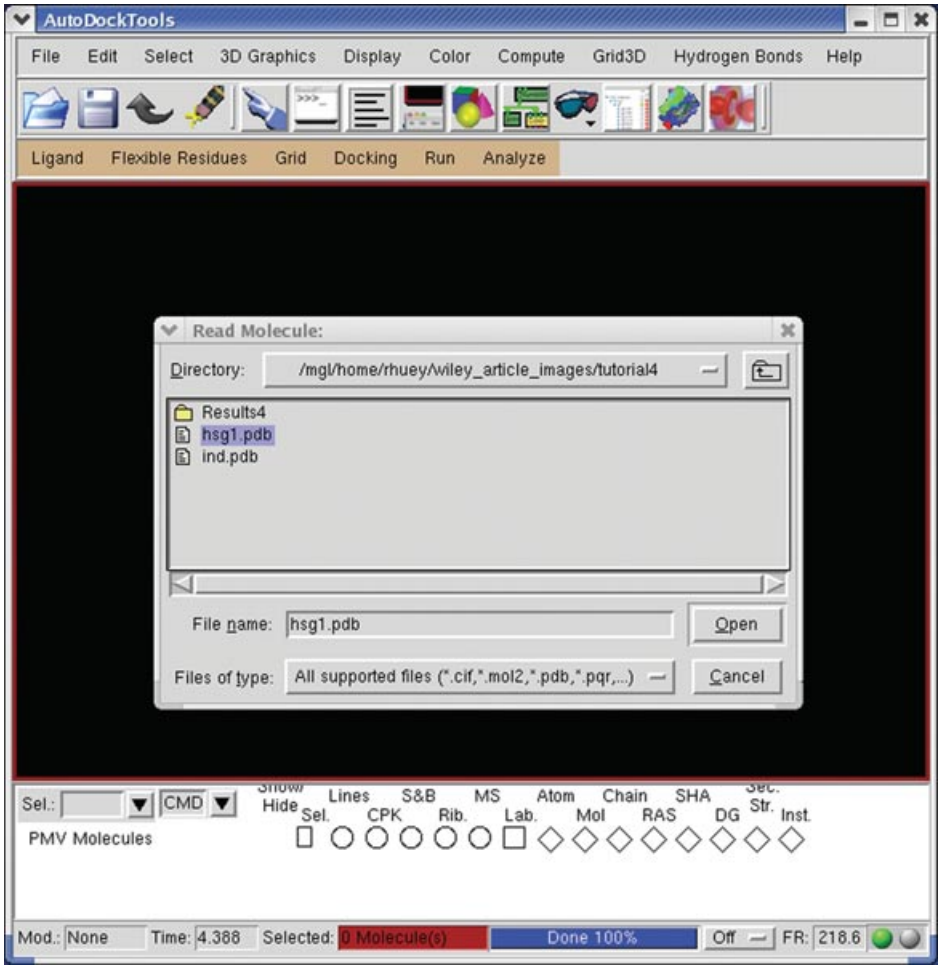


Figure 8.14.1 The AutoDockTools graphical user interface (GUI) has two rows of menus; the upper row is for more generic operations, while the lower row are specific to AutoDockTools. This figure shows the Read Molecule file browser about to load `hsg1.pdb`; note that the file browser is set to show all file types.

6. Click on the Select > Select From String menu item.

   This is used to build up a selection based on text strings typed for the Molecule, Chain, Residue, and/or Atom level. These strings can be names, numbers, ranges of numbers, or Python (http://www.python.org) lambda expressions that are evaluated to build a set. The strings can contain regular expressions including wild cards, such as the asterisk symbol (*), which matches anything. To select all atoms in the water molecules, type `HOH*` *in the Residue text field, press the Tab key to move to the next text field, i.e., the Atom entry, and type* `*`.

Table 8.14.2 Keys Used to Reset the View of the Molecule When the Cursor is Over the 3D Viewer

Key	Action
R	Reset view
N	Normalize: scale molecule(s) so all visible molecules fit in the viewer
C	Center on the center of rotation of all the molecules
D	Toggle on/off depth cueing (blends molecule into background farther away)

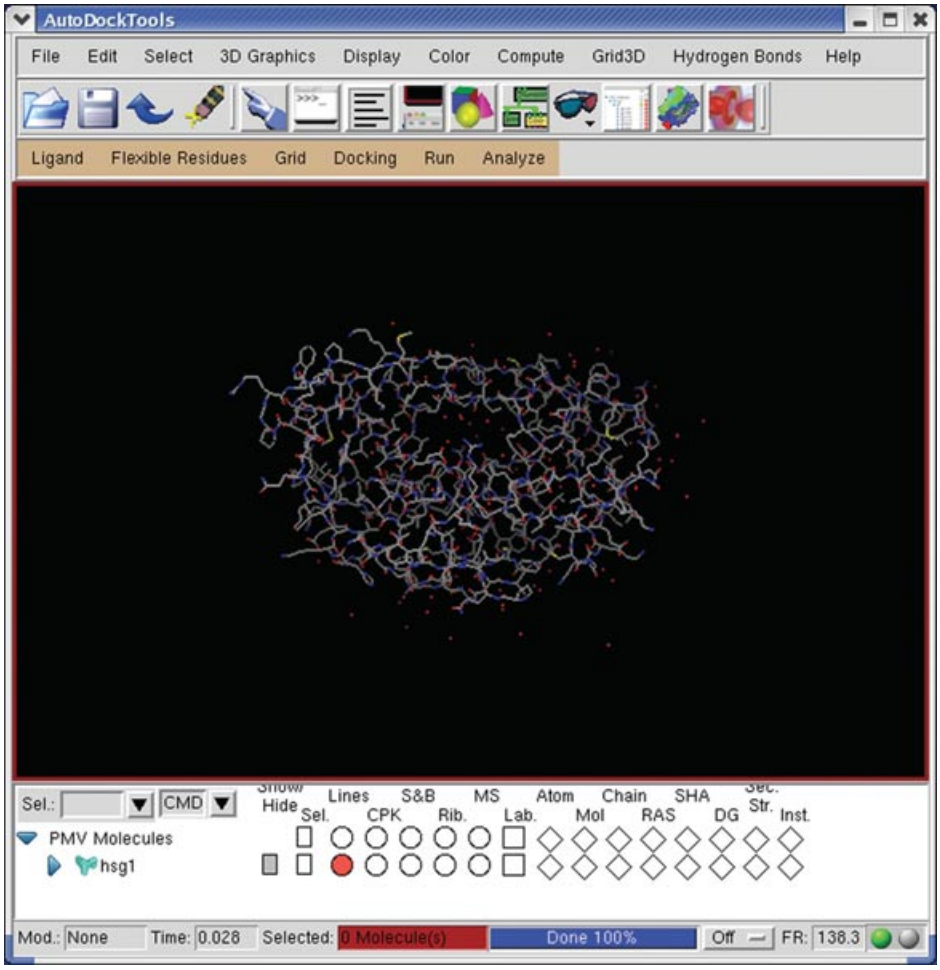


Figure 8.14.2 The receptor molecule HIV-1 protease from the PDB structure (1HSG) colored by atom type. For the color version of this figure go to *http://www.currentprotocols.com*.

7. Click Add. If a dialogue box appears asking to "change selection level to Atom", click Yes.

   Note ADT shows Selected: 127 Atom(s) with a yellow background in the center of the message-bar at the bottom of the ADT window.

8. Click Dismiss to close the Select From String widget.

9. Choose Edit > Delete > Delete AtomSet. If there is a current selection, it is deleted by this command. A confirmation dialogue box appears because deleting an AtomSet (or a molecule) cannot be undone.

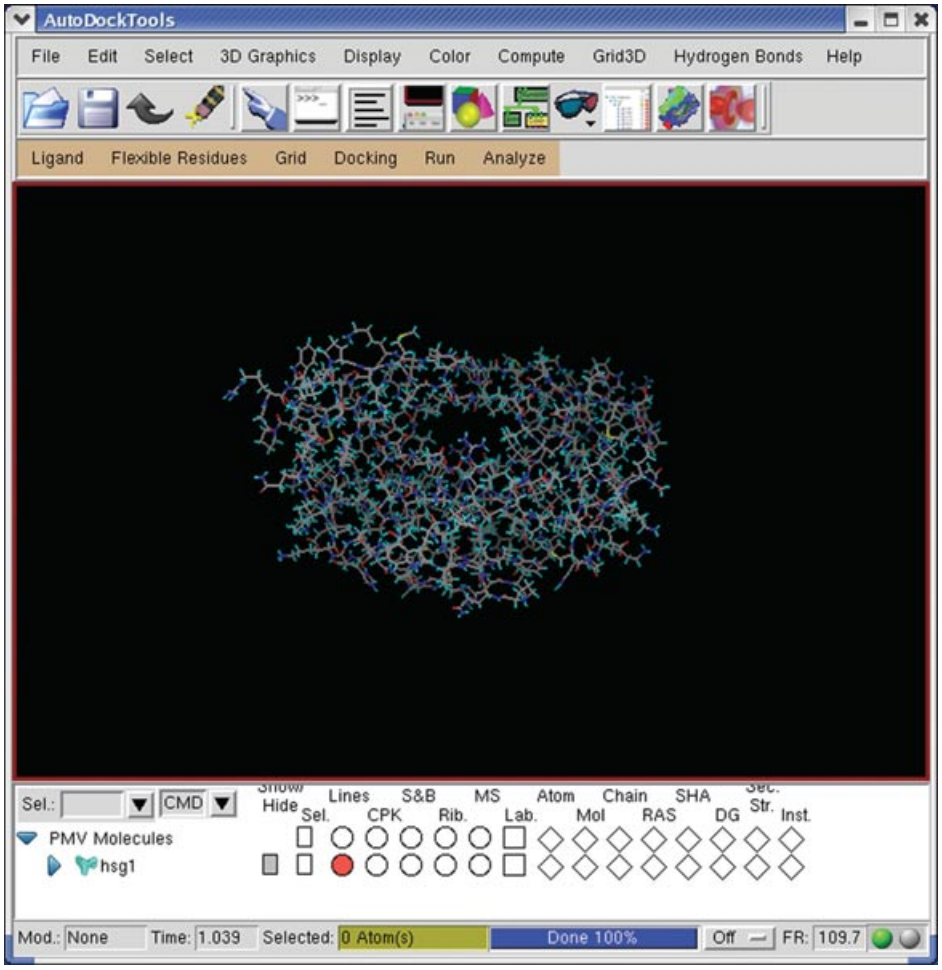


Figure 8.14.3 All hydrogen atoms have been added to HIV-1 Protease. For the color version of this figure go to *http://www.currentprotocols.com*.

10. Click on CONTINUE. The selected water oxygen atoms will disappear from the viewer.

11. Click on Edit > Hydrogens > Add Choose to add All Hydrogens using Method noBondOrder with yes to renumbering. Click OK to add all hydrogens. This causes 1612 hydrogen atoms to be added to hsg1 (Fig. 8.14.3).

    At this point, the macromolecule is cleaned up by removing water molecules, and hydrogen atoms have been added. Saving the macromolecule (steps 12 and 13) is highly recommended, but optional, because Basic Protocol 4 shows how to save the cleaned-up molecule as a PDBQT file, adding the necessary charges and atom types, assuming the macromolecule is still loaded in the AutoDockTools GUI.

    Note that saving the cleaned-up macromolecule as a PDB file is not necessary for AutoGrid and AutoDock calculations. The rigid macromolecule PDBQT file is used by AutoGrid. If there is one, the PDBQT file containing the flexible residues portion of the macromolecule is used by AutoDock.

Save the modified molecule (optional)

12. Save the molecule as a PDB file by choosing File > Save > Write PDB, and typing in `hsg1.pdb` as the filename.

13. To choose which types of PDB records to write (the default is to write ATOM and HETATM records only), whether to Sort Nodes, and whether to Save Transformed

Coords, choose Sort Nodes but leave all the other check-buttons off so that no CONECT records are written. Click OK to write the file.

The ATOM and HETATM records in the PDB file format store the names of the atoms and various structural data about each atom, in standard amino acids and nonstandard residues respectively. The CONECT records are another descriptor in PDB files that describe nonstandard bonds in the structure.

BASIC PROTOCOL 3

Preparing the Ligand

AutoDock ligands require partial atomic charges for each atom. AutoDock distinguishes between aliphatic and aromatic carbons; it also distinguishes between nitrogen atoms that can accept hydrogen bonds and those that are already bonded to hydrogen atoms and unable to hydrogen bond. The AutoDock types of these two types of nitrogen are NA and N, respectively. AutoDock ligands are written in files with special keywords recognized by AutoDock. The keywords ROOT, ENDROOT, BRANCH, and ENDBRANCH establish a "torsion tree" object that has a root and branches. (Fig. 8.14.4 shows the "root" of the

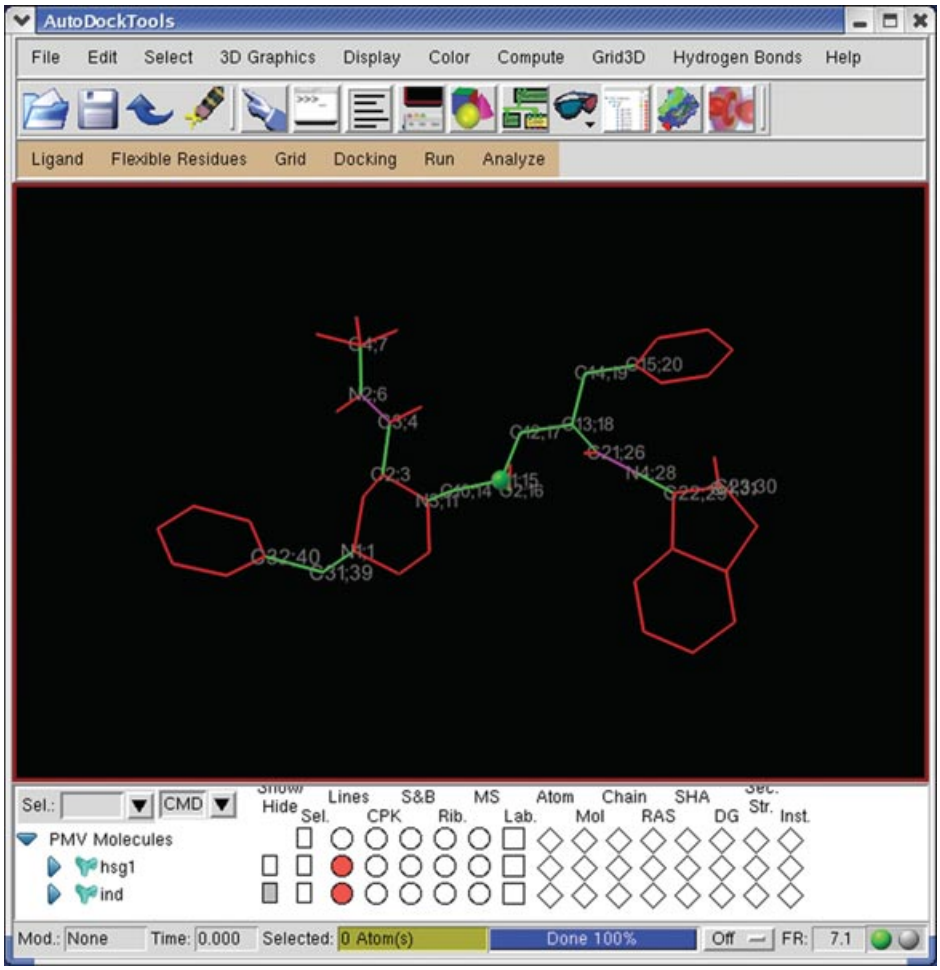


Figure 8.14.4 The "root" of the torsion tree is shown as a (green) sphere with the rotatable bonds as green lines. Bonds that could rotate but are not set to be rotatable are shown as magenta lines, and bonds that cannot rotate are shown as red lines. In this case, only one atom would appear between the ROOT and ENDROOT records, while all the atoms moved by each rotatable bond would appear between appropriately-labeled BRANCH and ENDBRANCH records. (These labels refer to the serial numbers of the two atoms involved in the rotatable bond.) For the color version of this figure go to *http://www.currentprotocols.com*.

torsion tree for Indinavir as a green sphere; this is the ligand used in this protocol.) The root is a rigid set of atoms, while the branches are rotatable groups of atoms connected to the rigid root. The keyword TORSDOF describes the number of torsional degrees of freedom in the ligand. In the AutoDock 4 force field, the TORSDOF value for a ligand is the total number of possible torsions in the ligand, but excluding rotatable bonds in rings, bonds to leaf atoms, amide bonds, guanidinium bonds, and so on. TORSDOF is used in estimating the change in free energy caused by the loss of torsional degrees of freedom upon binding.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

`ind.pdb` (a PDB file containing the ligand Indinavir with added hydrogen atoms, supplied in the GNU-zipped tar file `tutorial4.tar.gz`, downloaded in Basic Protocol 1)

It is possible to follow what happens with the ligand more easily by undisplaying the macromolecule first.

1. To undisplay the macromolecule, click on Display > Show/Hide Molecule. Click on the check-button labeled "hsg1:ON/OFF" to undisplay the macromolecule, hsg1.
2. Close the widget by clicking on the X in the top right corner of the widget (or the left-most red circle at the top of the window on Mac OS X).

The ligand can be set up using the Ligand menu in ADT in one of two ways: (1) by opening an existing PDB or SYBYL mol2 file (Ligand > Input > Open . . .), or (2) by choosing a ligand that is already loaded in the viewer (Ligand > Input > Choose . . .). In either case, the ligand must already have hydrogen atoms added; this can be done in ADT using the Edit > Hydrogens > Add menu option and then accepting the default settings and clicking OK, or using another program. In this protocol, we will use a ligand PDB file that already has hydrogen atoms added.

3. To load the ligand, use the following steps:
   a. Click on Ligand > Input > Open. . . to a file browser.
   b. Click on the PDBQT files: (`*.pdbqt`) menu button to display file type choices, and click on the PDB files: (`*.pdb`) button.
   c. Choose `ind.pdb` and click on Open.
   d. With the cursor over the 3D Viewer window, press "r", then "n", and finally "c" on the keyboard to improve the display of the ligand by resetting, normalizing, and centering the displayed molecules (see Table 8.14.2).
4. After the ligand is loaded, ADT automatically prepares it for AutoDock. This process involves a number of steps:
   a. ADT checks for and merges nonpolar hydrogens with the heavier atoms to which they are attached, unless the user preference "adt_automergeNPHS" is set not to do so.

b. ADT detects whether the ligand already has charges or not. If not, ADT computes Gasteiger charges. Remember that for the Gasteiger calculation to work correctly, the ligand *must* have all hydrogen atoms added, including both polar and nonpolar ones. If the charges are all zero, ADT will try to add charges. It checks whether the total charge per residue is an integer.

c. ADT assigns an "AutoDock type" to each atom. For peptide ligands, ADT uses a look-up dictionary for planar cyclic carbons (unless another user preference, "autotors_use ProteinAromaticList", is set not to do so). For other ligands, ADT determines which are planar cyclic carbons by calculating the angle between adjacent normals to all the atoms in the ring. If the angle is less than the default cutoff of 7.5° for all the atoms in the ring, the ring carbons' atom names will be assigned AutoDock type A. Nitrogen atoms that can accept hydrogen-bonds are assigned AutoDock type NA, while those that cannot are assigned N. In Indinavir, the AutoDock type of atom N5 in the heterocycle is NA, while the other nitrogens are assigned N. All polar hydrogens are assumed to be able to donate a hydrogen bond and are assigned the AutoDock type HD. Oxygen atoms can accept hydrogen bonds and are assigned AutoDock type OA. Likewise, sulfur atoms are assigned the AutoDock type SA.

   NOTE: *Since AutoDock uses a United Atom representation, by default ADT merges nonpolar hydrogens. This involves adding the charge of the nonpolar hydrogen atom to that of the carbon to which it is bonded, and then removing the nonpolar hydrogen from the molecule and adjusting the van der Waals radius of the carbon accordingly. It is possible to set a user-preference in ADT so as not to merge nonpolar hydrogens automatically. It is possible to model a nonpolar hydrogen explicitly, although this procedure is recommended only for expert users.*

d. ADT displays a message for the formatted ligand describing what type of charges were added, how many non-polar hydrogens, aromatic carbons and rotatable bonds were found, the number of torsional degrees of freedom detected (TORSDOF) and the amount the total charge differed from an integral value (total charge error). Click on OK to close the message window.

   NOTE: *Always add all hydrogen atoms to the ligand before selecting it to be the ligand.*

5. Click on Ligand > Torsion Tree > Detect Root. . .. ADT determines which atom is the best root and marks it with a green sphere (Fig. 8.14.4).

   This is the atom in the ligand nearest to the center of the network of bonds in the molecule. In the case of a tie, the atom that is in a cycle is picked to be the root. If neither atom is in a cycle, the first atom found is picked. (If both are in a cycle, the first one is picked). This can be a slow process for large ligands.

   The rigid portion of the molecule includes this root atom and all atoms connected to it by nonrotatable bonds (which will be examined in the next section). It is possible to visualize the current root portion with Ligand > Torsion Tree > Show Root Expansion and to hide this with Ligand > Torsion Tree > Show/Hide Root Marker. For this ligand, the root includes only the best root atom, atom C11, because all of its bonds to other atoms are rotatable.

6. Click on Ligand > Torsion Tree > Choose Torsions. . . to open a widget that displays the number of currently active bonds.

   Bonds that cannot be rotated are colored red. Bonds that could be rotated but are currently marked as inactive are colored purple. Bonds that are active, i.e., rotatable, are colored green (Fig. 8.14.4). Bonds in rings and cycles cannot be rotated. Bonds to leaf atoms in the tree do not move any atoms and are thus nonrotatable. Only single bonds can be rotated (not double or aromatic).

7. If desired, toggle the activity of a bond or group of bonds by clicking directly on it in the viewer. Alternatively, use the buttons on this widget to toggle the activity of a variety of bond types, e.g., peptide bonds, amide bonds, bonds between selected atoms, or all rotatable bonds.

   By default, amide bonds are treated as nonrotatable. Note that two bonds have been inactivated, the bond between atoms N2;6 and C3;4 and the bond between atoms C21;26 and N4;28. Note that the current total number of rotatable bonds is 14. Note, however, that amide bonds can be made rotatable by clicking on Make all amide bonds rotatable.

8. Before closing this widget by clicking Done, make sure all the bonds except the two amide bonds are active; "14/32" on the widget indicates that 14 are currently active out of the maximum number of torsions allowed by AutoDock, i.e., 32.

9. For the purposes of this protocol, click on Ligand > Torsion Tree > Set Number of Torsions. . ..

   This feature allows the setting of the total number of active torsions, and it selects them depending on whether they move the fewest atoms or the most. To see this distinction, set the radio button for "fewest atoms", type `6` *in the entry, and then press Enter on the keyboard. ADT will turn off all but 6 torsions, leaving active the torsions that move the fewest atoms.*

   Normally, the torsions selected will depend on the particular ligand being docked.

10. Set the radio button to "most atoms**"** and type `<Enter>` in the entry window.

    This shows a very different set of 6 rotatable bonds.

11. For this protocol, choose the 6 torsions that move the fewest atoms. Click Dismiss to close the widget.

12. Click on Ligand > Output > Save as PDBQT. . . to open a file browser. Type `ind.pdbqt` and click Save.

    It is important to write a PDBQT file. This is an AutoDock 4 specific file format that resembles a PDB file but is augmented by partial atomic charges and AutoDock atom types.

    Types are one or two characters long. Aromatic cyclic carbons are distinguished from aliphatic carbon atoms by the use of A replacing C. Nitrogens that can accept hydrogen bonds are distinguished from nitrogens that are unable to hydrogen-bond by the use of NA versus N, respectively. Support for distinguishing hydrogen-bonding sulfur and oxygen atoms from nonhydrogen-bonding sulfur and oxygen atoms also exists, although these are less common types. It is possible to model nonpolar hydrogen atoms that cannot donate a hydrogen bond, using the type H. For elements other than C, N, O, S, and H, the chemical symbol for the element is its AutoDock type. Atoms with two character atom names are no longer renamed in AutoDock 4 although they were in AutoDock 3. Cl is used for chlorine, Br for bromine, and Fe for iron, for example.

Saving the Macromolecule in PDBQT Format

BASIC PROTOCOL 4

The receptor macromolecule file used by AutoDock must be in PDBQT format, which is essentially a PDB-like format with partial atomic charges and AutoDock atom types (Fig. 8.14.5). To accomplish this, if the macromolecule is not in the viewer, read it in, as described in Basic Protocol 2, steps 2 and 3. Note that saving the cleaned-up macromolecule as a PDB file is not necessary for AutoGrid and AutoDock calculations. The rigid macromolecule PDBQT file is only used by AutoGrid. If there is one, the PDBQT file containing the flexible residues portion of the macromolecule is only used by AutoDock.

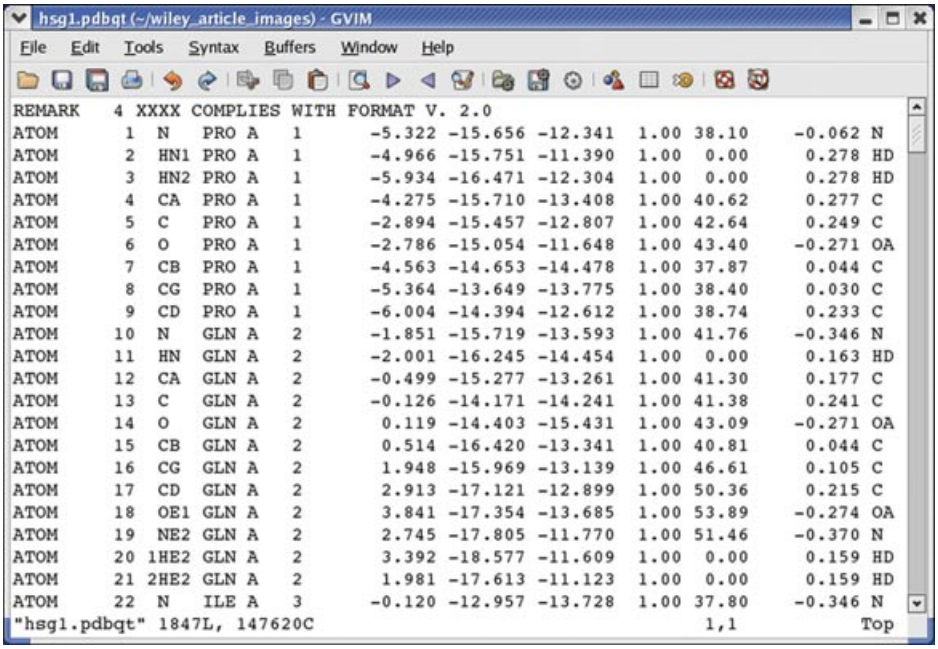


Figure 8.14.5 Part of a PDBQT file for the macromolecule used in this protocol, HIV protease. Note the last two columns, showing the partial atomic charge and the AutoDock 4 atom type for each atom. Abbreviations in last column: C, aliphatic carbon; H, hydrogen not capable of hydrogen-bonding; HD, polar hydrogen able to donate hydrogen bond; N, nitrogen not capable of accepting a hydrogen bond; OA, oxygen capable of accepting a hydrogen bond.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

Modified `hsg1.pdb` file (Basic Protocol 2)

1. Undisplay the ligand using Display > Show/Hide Molecule. Click on Grid > Macromolecule > Choose. . . and choose hsg1. Selecting the macromolecule in this way causes the following sequence of initialization steps to be carried out automatically.

   a. ADT checks that the molecule has charges. If not, it adds Gasteiger partial charges (Gasteiger and Marsili, 1978) to each atom. Remember that before computing charges, all hydrogen atoms must have been added to the macromolecule, not just "polar-only".

   b. ADT then merges nonpolar hydrogen atoms, unless the user preference "adt_automergeNPHS" is set not to do so.

   c. ADT also determines the AutoDock atom types of atoms in the macromolecule. AutoDock 4 can accommodate any number of atom types in the macromolecule.

   d. ADT reports the steps carried out in the initialization process.

2. Click on OK to continue. Since the molecule just chosen has been modified by ADT, a file browser opens to specify a file name.

3. Type `hsg1.pdbqt` and click Save.

The macromolecule must be saved in a PDBQT-formatted file for use by AutoGrid 4.

Preparing the Flexible Residues (Optional)

BASIC PROTOCOL 5

AutoDock 3 docks a flexible ligand to a rigid receptor. AutoDock 4 adds support for the option of including a conformational search of several designated residues in the receptor. If the receptor is treated as having flexible side chains, the molecule must be saved as two separate PDBQT files; if the receptor is to be treated as rigid, then it must be saved in one PDBQT file. As is the case for the ligand, the rotatable bonds in the moving side chains of the receptor are described in a "flexible residues" input PDBQT file with AutoDock 4 specific keywords `BEGIN_RES` and `END_RES`, as well as `ROOT`, `ENDROOT`, `BRANCH`, and `ENDBRANCH`. Note that the flexible residues are written in a separate PDBQT file to the ligand PDBQT file. The keyword "flexres" followed by the name of the flexible residues PDBQT file must be specified in the docking parameter file (DPF). In general, the more rotatable bonds in a docking experiment, the more calculations must be performed during the search to find a good solution.

Whether the receptor is treated as flexible or not, a PDBQT file for the rigid part of the macromolecule must be prepared; it is this file that is used for the AutoGrid 4 calculation. (Note that if the receptor has any flexible parts, but these atoms are not properly removed from the rigid PDBQT file before the grid maps are calculated, then during the AutoDock 4 calculation the moving atoms in the flexible residues will collide with their stationary representations, generating extreme energies.) The recommended file-naming convention is to use `receptor_flex.pdbqt` for the moving atoms in the receptor, and `receptor_rigid.pdbqt` for the rigid part.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

Modified `hsg1.pdbqt` file (Basic Protocol 4)

1. Undisplay the ligand using Display > Show/Hide Molecule.

2a. *If hsg1 is in the viewer:* Choose it as the macromolecule: click on Flexible Residues > Input > Choose Macromolecule. . ..

2b. *If hsg1 is not in the viewer:* Read in `hsg1.pdbqt` by clicking on Flexible Residues > Input > Open Macromolecule. . .. A dialog box appears asking to merge nonpolar hydrogen atoms; click Yes. A second dialogue box appears, stating that Gasteiger charges and AutoDock 4 atom types were added and that the nonpolar hydrogen atoms were merged. Click OK to continue.

3. Select the residues to be flexible, using Select > Select From String, and type `ARG8` in the Residue entry; click the Add button.

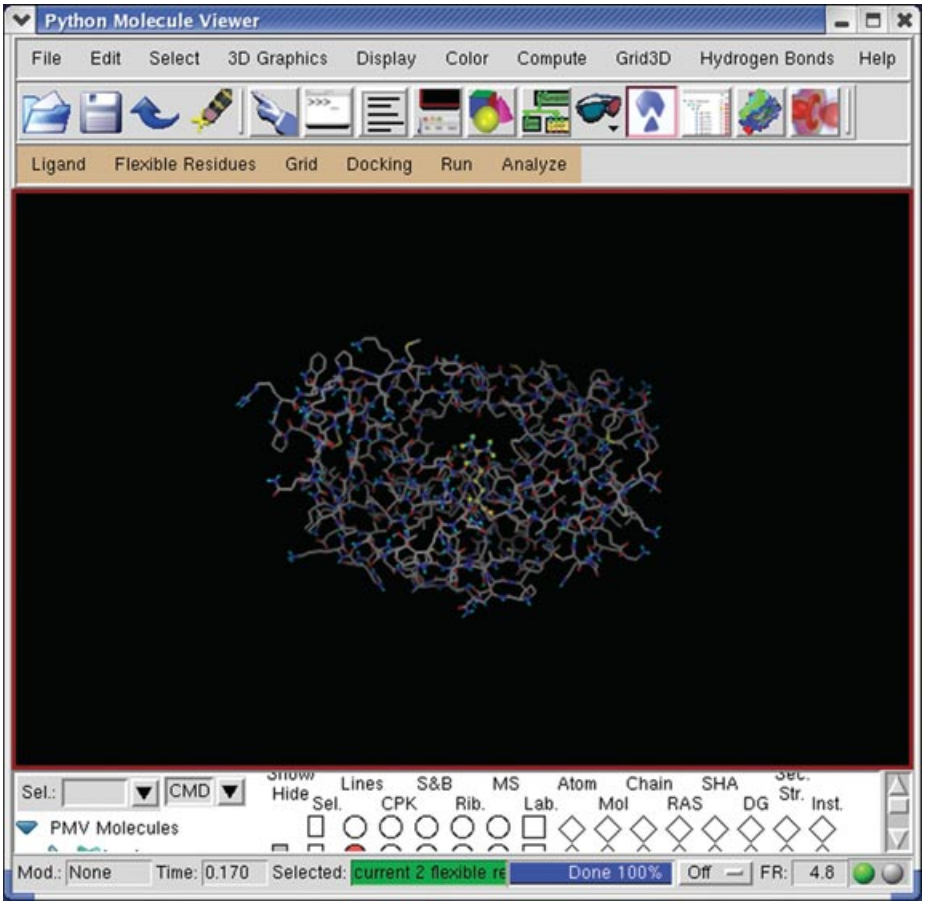


Figure 8.14.6 Two Arg-8 side chains have been selected to be flexible residues. Each selected atom is indicated with yellow crosses. Note that in the middle of the bottom row, the number of selected residues is shown in the box after the word Selected. For the color version of this figure go to *http://www.currentprotocols.com*.

4. Click Dismiss to close the Select From String widget. Check that "current 2 flexible residue(s)" appears in the Selected entry below the 3D viewer (Fig. 8.14.6).

5. Set up the flexibility pattern in the selected residues, by choosing Flexible Residues > Choose Torsions in Residues. . ..

   This hides all the nonselected residues in the macromolecule. The side chains of the selected residues are shown with currently rotatable bonds colored green, unrotatable bonds colored red and nonrotatable bonds colored magenta. The total number of rotatable bonds is listed in the Torsion Count widget. Clicking on a rotatable bond (green) makes it nonrotatable (red), and vice versa.

6. Click on the rotatable bond between the CA and CB atoms in each arginine residue to inactivate it. This leaves a total of 6 rotatable bonds in the 2 flexible ARG8 residues (Fig. 8.14.7).

7. Click on Close.

Save the macromolecule

As discussed earlier, the macromolecule must be saved in two files: one containing the formatted, flexible ARG8 residues, and the other containing the rest of the residues in the macromolecule.

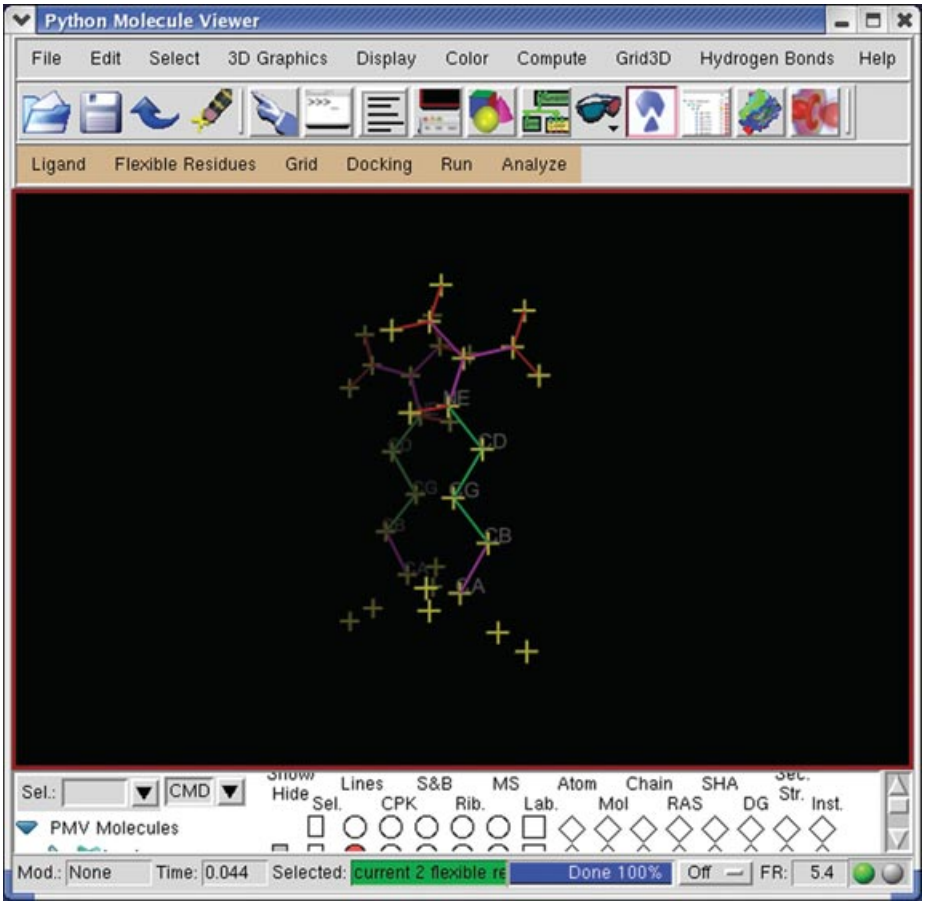


Figure 8.14.7 A close-up of the rotatable bonds selected in the Arg-8 sidechains. Three green rotatable bonds are set to be rotatable in each Arg-8 side chain: CB-CG, CG-CD and CD-CE. For the color version of this figure go to *http://www.currentprotocols.com*.

8. Save the flexible residues by clicking on Flexible Residues > Output > Save Flexible PDBQT. . ., and type `hsg1_flex.pdbqt` in the AutoFlex File browser and click Save.

9. Save the non-moving, rigid residues by clicking on Flexible Residues > Output > Save Rigid PDBQT. . ., and type `hsg1_rigid.pdbqt` in the AutoFlex Non-Flexible Residue Output File browser. Click the Save button.

   It is possible to redisplay the rest of the macromolecule using Flexible Residues > Redisplay Macromolecule.

PREPARING THE GRID PARAMETER FILE

BASIC PROTOCOL 6

The grid parameter file (GPF) tells AutoGrid 4 which receptor to compute the potentials around, the types of maps to compute, and the location and extent of those maps. It may also specify a custom library of pairwise potential energy parameters. In general, one map is calculated for each atom type in the ligand, plus an electrostatic potential map and a desolvation energy map. These grid maps are necessary for AutoDock, and they describe how various "probe" atom types (e.g., aliphatic carbons, hydrogen-bonding oxygens and hydrogens) interact at regularly spaced intervals throughout the grid box. AutoDock version 4 also requires electrostatic potential maps and desolvation energy maps.

Table 8.14.3 Menu Buttons of the "Grid Options" Widget

Button	Action
File	This menu is used to close the Grid Options widget, which also causes the grid box to disappear. It is possible to Close saving current values to keep any changes made while using this widget, or Close without saving, to forget the changes.
Center	This menu sets the center of the grid box in four ways: >, Pick an atom, > Center on ligand, > Center on macromolecule, or > On a named atom.
View	This menu changes the visibility of the grid box using Show box, and whether it is displayed as lines or faces, using the toggle Show box as lines. It is also possible to show or hide the center marker using Show center marker, and to adjust its size using Adjust marker size.

Using the procedure described below, the Grid Options widget can be used to customize the display. This widget has menu buttons at the top, as described in Table 8.14.3. In addition, the Grid Options widget displays the Current Total Grid Points per map, showing how many grid points each grid map will have: $(n_x + 1) \times (n_y + 1) \times (n_z + 1)$, where n_x, n_y, and n_z are the numbers of grid points in the x, y, and z dimensions, respectively. There are also three thumbwheels that change the number of points in the x, y, and z dimensions at even intervals between 2 and 126. The default values are 40, 40, 40. AutoDock requires one grid point to be at the centre of the grid maps, so AutoGrid always adds one grid point to the user-specified even-value number of grid points in each dimension.

There is one thumbwheel that interactively adjusts the spacing between the grid points. The default value is 0.375 Å between grid points, which is about a quarter of the length of a carbon-carbon single bond. Grid spacing values of up to 1.0 Å can be set using this widget, when a large volume is to be investigated. If larger grid spacing values are desired, edit the GPF in a text editor before running AutoGrid. Lastly, there are also entries and thumbwheels that change the location of the center of the grid box.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

Rigid receptor file: `hsg1.pdbqt` (from Basic Protocol 4) *or* `hsg1_rigid.pdbqt` (Basic Protocol 5)
Modified ligand file (Basic Protocol 3)

1. Click on Grid > Set Map Types.

   The types of maps depend on the types of atoms in the ligand. Thus, one way to specify the types of maps is by choosing a ligand.

2a. *If the ligand formatted earlier is still in the viewer:* Choose Grid > Set Map Types > Choose Ligand. . ..

2b. *If the ligand formatted earlier is not still in the viewer:* Use Grid > Set Map Types > Open Ligand. . ..

3. *Optional*: If modeling flexibility in some of the residues in the receptor, and if the flexible residues formatted earlier are still in the viewer, choose Grid > Set Map Types > Choose FlexRes. . .. If not, use Grid > Set Map Types > Open FlexRes. . ..

   To use the same macromolecule with a variety of different ligands, choose to calculate all of the required maps via Set Map Types > Directly.

4. Click on Grid > Grid Box, which opens the Grid Options panel (Fig. 8.14.8).

5. Adjust the number of points in each dimension to 60. Notice that each map will have 226,981 grid points, each with its own unique value of interaction energy.

6. Type in `2.5`, `6.5`, and `−7.5` in the *x* center, *y* center and *z* center entries, respectively.

   This will center the grid box on the active site of the HIV-1 protease, hsg1.

7. Close this widget by clicking File > Close saving current.

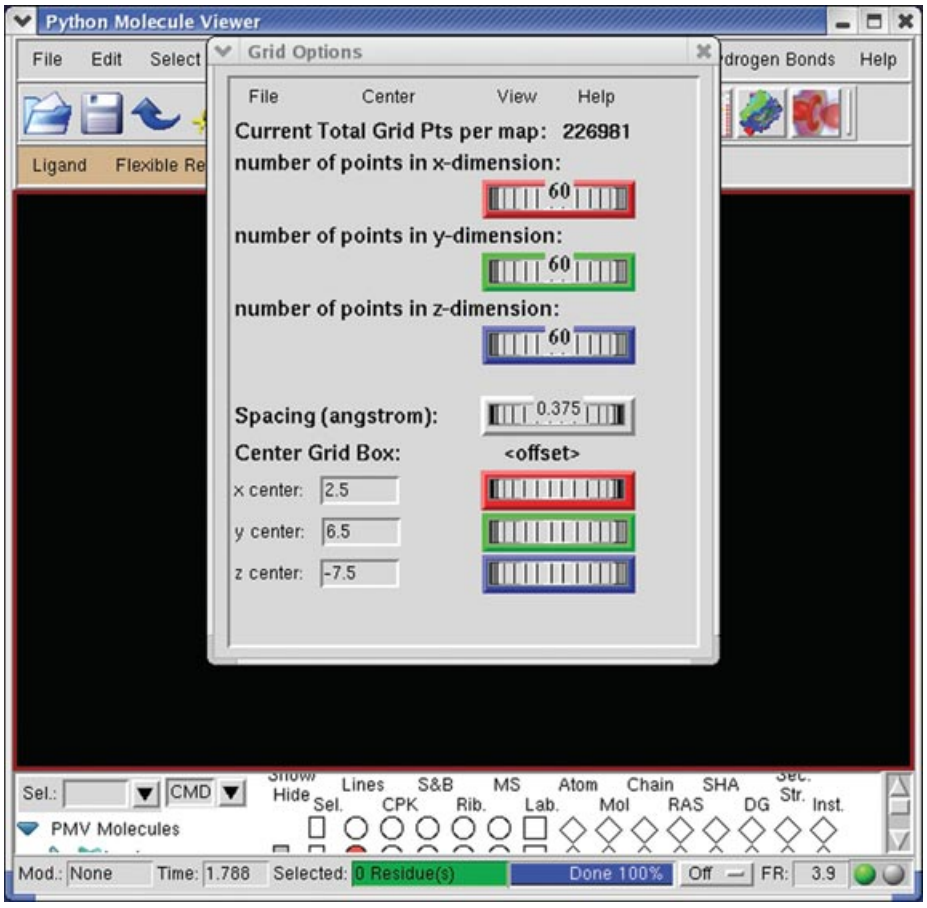


Figure 8.14.8 It is possible to control the size of the AutoGrid box used to compute the grid maps using the Grid Options panel. The number of grid points can be changed by dragging on the thumbwheel, or by typing in the value and pressing Enter while the cursor is over the thumbwheel. Similarly, the grid point spacing in Å and the *x,y,z* coordinates of the center of the grid box can be specified using this panel.

8. Click on Grid > Output > Save GPF. . . to open a file browser and specify the name of the GPF.

   The convention is to use `.gpf` *as the extension. Save the GPF as* `hsg1.gpf`.

9. Click on Grid > Edit GPF to open the GPF in an editing window. This shows the contents of the file that was saved in step 8 (`hsg1.gpf`).

   IMPORTANT NOTE: *To set up a docking using flexible residues in the receptor, make sure the specified receptor file is named* `hsg1_rigid.pdbqt`*, because grid maps must be calculated using a PDBQT file for the receptor molecule that lacks the moving residues.*

   If there are any flexible residues in the receptor, and the optional Basic Protocol 5 was followed, make sure that any atom types that are in the flexible residues but not in the ligand (e.g. S in Cys or Met) are included in the list following the ligand_types keyword.

10. Save changes made to the content of the GPF (if any were necessary) using the Write button.

11. Click either OK or Cancel to close this widget.

RUNNING AutoDock (BASIC PROTOCOLS 7, 8, AND 9)

Before a docking can be performed using AutoDock, a three-dimensional "array" of interaction energies must be calculated for all the atom types in the ligand, and any moving parts of the receptor. This set of atom type interaction energies, or *grid maps*, must be calculated using AutoGrid, and once calculated for a given receptor, these same maps can be used for any ligands that possess these atom types. To control how AutoDock will perform the docking, the parameters for the docking must be saved in a docking parameter file (DPF). With the ligand and receptor PDBQT files, grid maps, and DPF in hand, the docking can be carried out; Basic Protocols, 7, 8 and 9 explain these steps in detail.

BASIC PROTOCOL 7

Starting AutoGrid 4

In general, AutoGrid 4 (and AutoDock 4) must be run in the directories where the rigid macromolecule, ligand, and parameter files are to be found. The named files in the parameter file must not include pathnames.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

`hsg1.gpf` (Basic Protocol 6)

1. Click on Run > Run AutoGrid, which will open the Run AutoGrid widget.

2. Specify which machine to use, using the first two entries in the widget.

   By default the local machine is named in the "Macro Name:" entry and in the "Host name:" entry. It is possible to define macros to specify other machines with Run- > Host Preferences.

3. As needed, set the other options described in Table 8.14.4.

Table 8.14.4 AutoGrid 4 Options

Option	Description
Program Pathname	Specifies the location of the autogrid4 executable. If it is not in the path, use the Browse button to locate it.
Parameter Filename	Specifies the grid parameter file (GPF). If a GPF was written during this session, this widget will automatically fill in the GPF filename in the Parameter Filename: entry. If not, use the Browse button to the right of the entry to locate the desired GPF.
Log Filename	Specifies the grid log file (GLG). Selecting a GPF fills in the name for the GLG based on the stem part of the GPF filename.
Nice Level	Specifies a nice level for remote jobs.
Cmd	Displays the UNIX command that will be executed when the Launch button is clicked.

4a. *To start the AutoGrid 4 job from the menu:* Click on Launch. On most platforms, this opens an AutodockProcess Manager widget that displays specifics about current AutoGrid and AutoDock jobs, and can be used to terminate a process by selecting its entry.

4b. *To start the AutoGrid 4 job from the command line:* Type the following command.

```
% autogrid4 -p hsg1.gpf -l hsg1.glg &
```

The % symbol is not to be typed; it represents the UNIX prompt.

Setting Up the Docking

BASIC PROTOCOL 8

The docking parameter file (DPF) tells AutoDock which grid map files to use, which ligand molecule to dock, what its center and number of torsions are, where to start the ligand, which flexible residues to move if side chain motion in the receptor is to be modeled, which docking algorithm to use, and how many runs to do. It usually has the file extension `.dpf`. Four different docking algorithms are currently available in AutoDock: SA, the original Monte Carlo simulated annealing; GA, a traditional Darwinian genetic algorithm; LS, local search; and GA-LS, which is a hybrid global-local search that combines the genetic algorithm with local search. The GA-LS is also known as a Larmarckian genetic algorithm (LGA) because "offspring" are allowed to inherit the local search adaptations of their "parents". The LGA was compared with a traditional genetic algorithm (GA) and Monte Carlo simulated annealing (SA) by Morris et al. (1998), where it was shown that the LGA was the most robust and efficient of these three search algorithms.

Each search method has its own set of parameters, and these must be set before running the docking experiment itself. These parameters include, e.g., what kind of random number generator to use, step sizes. The most important parameters affect how long each docking will run. In simulated annealing, the number of temperature cycles, number of accepted moves, and number of rejected moves determine how long a docking will take. In the GA and GA-LS, the number of energy evaluations and number of generations affect how long a docking will run.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

`hsg1.pdbqt` (Basic Protocol 4) *or* `hsg1_rigid.pdbqt` (Basic Protocol 5)
`hsg1_flex.pdbqt` (optional; Basic Protocol 5)

Choose the molecule files

1. Click on Docking > Macromolecule > Set Rigid Filename. Select `hsg1.pdbqt` in the PDBQT Macromolecule File: panel.
2. *Optional*: If you are including flexible residues in your experiment, select the appropriate `receptor_rigid.pdbqt` file.

   Selecting the appropriate file is mandatory if including flexible residues. In this tutorial this would be `hsg1_rigid.pdbqt`.
3. Click on Docking > Ligand > Choose... and choose "ind". Click Select Ligand. This opens a panel displaying the name of the current ligand, its atom types, its center, its number of active torsions, and its number of torsional degrees of freedom.

   It is possible to set a specific initial position of the ligand and initial relative dihedral offsets and values for its active torsions.
4. For this protocol, use the defaults and click Close to close this widget.
5. *Optional*: If modeling side chain flexibility in the receptor, it is also necessary to specify the name of the PDBQT file containing the flexible residues in the docking parameter file, typically `receptor_flex.pdbqt`. Click on Docking > Macromolecule> Set Flexible Residues Filename..., and choose `hsg1_flex.pdbqt`, then click Open.

Set docking parameters

6. Click on Docking > Search Parameters... > Genetic Algorithm... to set the genetic algorithm-specific parameters (Fig. 8.14.9).

   It is advisable when setting up a new docking to do a trial run with fewer energy evaluations (~25,000 evals).
7. For this protocol, use the defaults and click Close to continue.
8. Click on Docking > Docking Parameters.

   Here it is possible to choose which random number generator to use, the random number generator seeds, the energy outside the grid, the maximum allowable initial energy, the maximum number of retries, the step size parameters, output format specification, and whether or not to do a cluster analysis of the results. There is usually no need to change any of these parameters and settings.
9. For this protocol, use the defaults and just click Close.
10. Click on Docking > Output > Lamarckian GA... and specify the name of the DPF.

    This file will contain docking parameters and instructions for a Lamarckian genetic algorithm (LGA) docking, also known as a hybrid genetic algorithm-local search (GA-LS).

    ADT allows changes to be made to the parameters for any of the four possible search methods at any time. The choice of the specific search algorithm is made only when choosing which kind of docking parameter file to write.
11. Type in `ind.dpf` and click on Save.

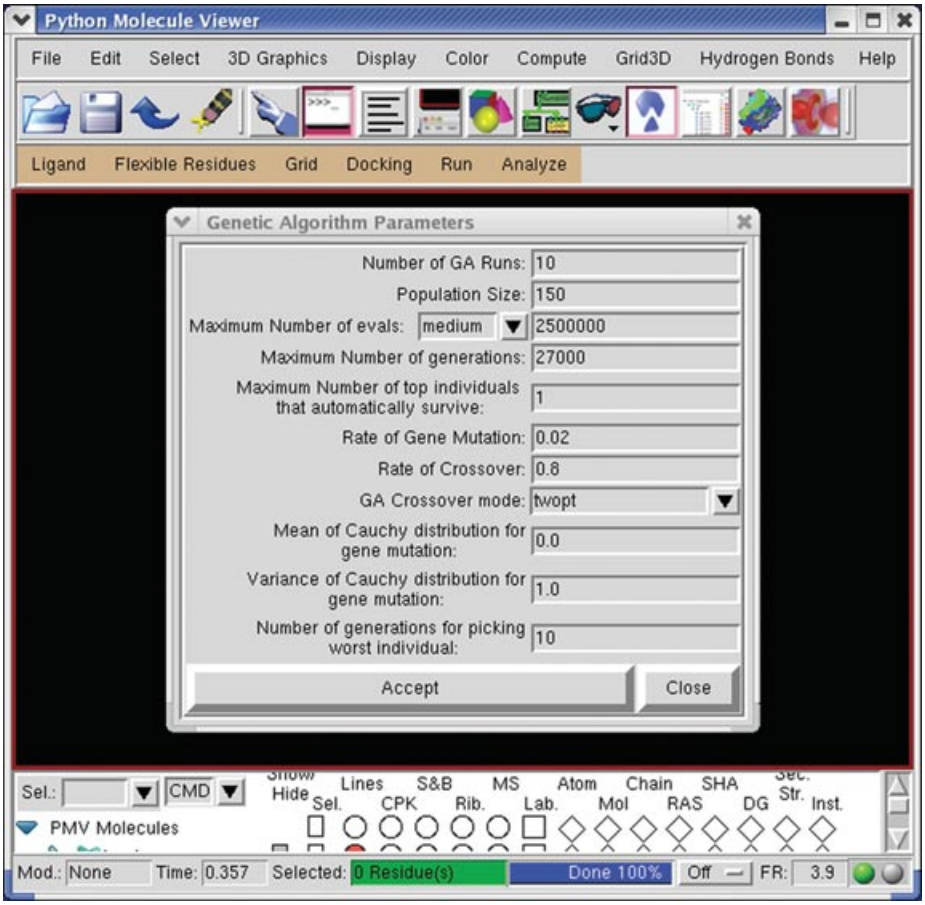


Figure 8.14.9 The most important parameters for the Genetic Algorithm (GA) and Lamarckian Genetic Algorithm (LGA) are set in this panel. The number of independent docking runs, the size of the population, and how long each docking will run can be set here. The GA and LGA will terminate when either the maximum number of energy evaluations (evals) or the maximum number of generations is reached, whichever comes first.

12. Click on Docking > Edit DPF. . . to look at the contents of the file from step 11.
13. Check that the output filename, `ind.pdbqt`, appears after the keyword "move" and that "torsdof" is set to `14`.
14. If modeling flexible residues, make sure that the keyword `flexres` is included and that the stem of the map names is `hsg1_rigid`.
15. Click either OK or Cancel to continue.

Starting AutoDock 4

BASIC PROTOCOL 9

In general, AutoDock 4 must be run in the directory where the macromolecule, ligand, GPF, and DPF are to be found. If the docking involves flexible residues in the receptor, the "flexres" file must be found in the directory also. Also, the named files in the parameter file must not include pathnames.

The procedure described below (step 1) will open the Run AutoDock widget. The first two entries in the widget are used to specify which machine to use. By default the local machine is named in the Macro Name entry and in the Host Name entry. It is possible to define macros to specify other machines. Other parameters than can be customized are described in Table 8.14.5.

Table 8.14.5 "Run AutoDock" Widget Options

Option	Description
Program Pathname	Specifies the location of the `autodock4` executable. If it is not in the path of the account being used, click the Browse button to locate it.
Parameter Filename	Specifies the DPF file. If there is no DPF specified, use the Browse button to the right of the entry to locate the desired DPF.
Log Filename	Specifies the log file. Selecting a DPF in the "Parameter Filename" entry automatically creates a corresponding name for the DLG using the same stem as the DPF.
Nice Level	Specifies a UNIX nice level or priority for remote jobs.
Cmd	Shows the command that will be invoked when clicking on Launch.

Following execution of this protocol, AutoDock 4 stores the progress of the dockings in the log file specified by the `-l` flag (which in this case is `ind.dlg`). The time required for the docking calculations depends on the maximum number of evaluations in each run (`ga_num_evals`), the maximum number of generations (`ga_num_generations`), and the total number of runs (`ga_run`) specified in the docking parameter file (`ind.dpf` in this example). In addition, the complexity of the search will depend on the number of torsions; rigid ligands can be docked more quickly than flexible ligands. When the calculation finishes, the last lines printed in the docking log file include the phrase Successful Completion and the amount of time taken for the calculation.

NOTE: Reading and interpreting docking logs are described in the Basic Protocols 10, 11, 12, and 13.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

AutoGrid map files: `hsg1_rigid.maps.fld`, `hsg1_rigid.A.map`, `map hsg1_rigid.C.map`, `hsg1_rigid.HD.map`, `hsg1_rigid.N.map`, `hsg1_rigid.NA.map`, `hsg1_rigid.OA.map`, `hsg1_rigid.e.map`, `hsg1_rigid.d.map` (Basic Protocol 7)
PDBQT file containing the ligand: `ind.pdbqt` (Basic Protocol 3)
PDBQT file containing flexible residues: `hsg1_flex.pdbqt` (optional; Basic Protocol 5)
Docking parameter file (DPF): `ind.dpf` (Basic Protocol 8)

1. Click on Run > Run AutoDock. . . to open the Run AutoDock widget.

2a. *To start the AutoDock job from the menu:* Click on Launch. This opens an AutoDock Process Manager widget that shows details about currently running AutoDock jobs.

This can be used to terminate an AutoDock process by selecting its entry.

2b. To start the AutoDock job from the command line: Type the following.

```
% autodock4 -p ind.dpf -l ind.dlg &
```

The % symbol is not to be typed; it represents the UNIX prompt.

ANALYZING AutoDock RESULTS (BASIC PROTOCOLS 10, 11, 12, AND 13)

Having performed a number of dockings, it is necessary to analyze the results. This typically involves organizing the results into clusters of conformationally similar binding modes. AutoDock performs conformational cluster analysis if the DPF keyword "analysis" is given, but it is also possible to re-cluster docking results using ADT. ADT can also be used to display both the docked conformations and interactive histograms of the clusterings.

Basic Protocols 10, 11, 12, and 13 explain how to use ADT to read in a docking log file from AutoDock, determine if the each of the dockings has searched sufficiently (checking that there are enough energy evaluations and generations) and to evaluate the chemical reasonableness of the interactions between the docked conformations of the ligand and the receptor. Successful docking calculations display convergence on a small number of clusters; this reflects the thoroughness of the search. If a large enough number of evaluations and generations is used, the docking results will tend to form conformationally similar clusters. The interpretation of AutoDock results is somewhat open-ended; in large part, it depends on the user's chemical insight.

Reading Docking Logs

BASIC PROTOCOL 10

Reading a docking log or a set of docking logs is the first step in analyzing the results of docking experiments. While docking, AutoDock outputs a detailed record to the file specified after the `-l` flag. These log files can be very long; in this example, `ind.dlg` contains over 11,000 lines (see Fig. 8.14.10 for an excerpt). The log file includes details about the docking that are output as AutoDock parses the input docking parameter file. For example, for each AutoGrid map it reads in, AutoDock reports opening the map file and how many data points it read in. When AutoDock parses the input ligand PDBQT file, it reports building various internal data structures. After the input phase, AutoDock begins the specified number of docking runs. It reports which run number it is starting; it may report specifics about each generation depending on the requested output level. When each docking is completed, AutoDock outputs the docked conformation of the ligand in PDBQT format. After completing all of the requested docking runs, and if the "analysis" command was included in the DPF, AutoDock begins a conformational analysis of all the docked conformations it found. At the very end, it reports a summary of the amount of time taken and the words Successful Completion. The level of detail in the log file is controlled by the DPF keyword "outlev". For dockings using the Lamarckian GA search method, an outlev of 0 is recommended.

The most important parts in a docking log file are the docked structures found at the end of each run, energies of these docked structures, and conformational clustering analysis. It is a page showing this information that needs to be the sample log file page. The clustering analysis proceeds as follows: The similarity of docked structures is measured by computing the root-mean-square-deviation (RMSD) between the coordinates of corresponding atoms. The docked conformations are sorted by energy, from lowest to highest, and the lowest energy structure that has not yet been clustered forms the seed for a new cluster. The remaining conformations are compared in turn to the seed by computing their RMSD, and if this is less than a user-defined RMSD threshold (usually 2 Å), that conformation is added to the cluster. The process is repeated until all the docked conformations have been compared.

```
8074 DPF> ga_run 10                          # do this many hybrid GA-LS runs
8075
8076 Number of requested LGA dockings = 10 runs
8077
8078
8079     BEGINNING LAMARCKIAN GENETIC ALGORITHM DOCKING
8080
8081 Run:    1 / 10
8082 Date:   Tue Jul 17 13:46:03 2007
8083 Output level is set to 1.
8084
8085
8086 Creating an initial population of 150 individuals.
8087
8088 Assigning a random translation, a random orientation and 12 random torsions to each of the 150 individuals.
8089
8090 Beginning Lamarckian Genetic Algorithm (LGA), with a maximum of 2500000
8091 energy evaluations.
8092
8093 Generation: 100   Oldest's energy: -14.725    Lowest energy: -14.725 Num.evals.: 97440   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8094 Generation: 200   Oldest's energy: -16.291    Lowest energy: -16.291    Num.evals.: 192832   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8095 Generation: 300   Oldest's energy: -16.398    Lowest energy: -16.398    Num.evals.: 292065   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8096 Generation: 400   Oldest's energy: -17.030    Lowest energy: -17.030    Num.evals.: 387743   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8097 Generation: 500   Oldest's energy: -17.178    Lowest energy: -17.178    Num.evals.: 476624   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8098 Generation: 600   Oldest's energy: -17.454    Lowest energy: -17.454    Num.evals.: 569106   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8099 Generation: 700   Oldest's energy: -17.607    Lowest energy: -17.607    Num.evals.: 664115   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8100 Generation: 800   Oldest's energy: -17.685    Lowest energy: -17.685    Num.evals.: 760812   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8101 Generation: 900   Oldest's energy: -17.731    Lowest energy: -17.731    Num.evals.: 858947   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8102 Generation: 1000   Oldest's energy: -17.770    Lowest energy: -17.770    Num.evals.: 954067   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8103 Generation: 1100   Oldest's energy: -17.777    Lowest energy: -17.777    Num.evals.: 1048388   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8104 Generation: 1200   Oldest's energy: -17.817    Lowest energy: -17.817    Num.evals.: 1146010   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8105 Generation: 1300   Oldest's energy: -17.823    Lowest energy: -17.823    Num.evals.: 1245926   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
  :
  :
8115 Generation: 2300   Oldest's energy: -17.939    Lowest energy: -17.939    Num.evals.: 2217282   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8116 Generation: 2400   Oldest's energy: -17.939    Lowest energy: -17.939    Num.evals.: 2310728   Timing: Real= 0.02s,  CPU= 0.02s,  System= 0.00s
8117 Generation: 2500   Oldest's energy: -17.939    Lowest energy: -17.939    Num.evals.: 2408625   Timing: Real= 0.02s,  CPU= 0.03s,  System= 0.00s
8118 Final-Value: -17.939
8119
8120 Run completed;  time taken for this run:
8121 Real= 4m 38.13s,  CPU= 4m 32.46s,  System= 0.04s
8122
8123 1:50 41" p.m., 07/17/2007
8124 Total number of Energy Evaluations: 2500279
8125 Total number of Generations:        2593
8126
8127
8128     FINAL LAMARCKIAN GENETIC ALGORITHM DOCKED STATE
8129     ________________________________________________
8130
8131
8132 State:  2.609 6.130 -7.730  -0.648 0.631 0.426 -106.277   -178.46 -32.92 -125.89 14.67 140.53 -11.85 100.67 15.53 90.79 87.08 42.34 90.57
8133
8134 DOCKED: MODEL        1
8135 DOCKED: USER    Run = 1
```

Figure 8.14.10 A small excerpt from an AutoDock log file (DLG), with line numbers added for clarity. It shows the output at the beginning and end of a docking, and the beginning of the output of the docked ligand PDBQT file, each line of which is preceded by the string DOCKED:. The USER records give information about the number of the docking run, the docking parameter file (DPF) used, the estimated free energy of binding, an energy breakdown, and a description of the position, orientation, and conformation of the ligand. Much more information is present in the DLG, but cannot be shown in this figure. DLG files can be read in by AutoDockTools, which greatly facilitates the analysis of the dockings.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

AutoDock log file (`ind.dlg`; Basic Protocol 9)

1. If there are any molecules visible in the viewer, undisplay them using Display > Show/Hide Molecule.
2. Click on Analyze > Dockings > Open. . . to choose the AutoDock log file to analyze. This command opens a file browser that looks for files with the extension `.dlg`. Choose `ind.dlg`.

Reading a docking log creates a Docking instance in ADT. A Conformation instance is created for each docked result found in the docking log. A Conformation represents a specific state of the ligand and has either a particular set of state variables from which all the ligand atoms' coordinates can be computed or the coordinates themselves. Conformations also have energies: docked energy, estimated binding energy, and possibly, per atom electrostatic and van der Waals/H-bond energies. AutoDock 4 computes the free energy of binding and reports a detailed energy breakdown.

ADT reports how many docked conformations were read in from the AutoDock docking log (DLG) and gives instructions on to how to visualize the docked conformations or states.

3. If there are any warning messages from the AutoDock, they are recorded in the docking log. To view these in ADT, open the Python shell, type `mv.docked.warnings`, and press Enter. Note that if there is a previous Docking instance in the viewer, ADT asks whether to add this DLG to the previous Docking instance.

   This only makes sense when the same AutoGrid map files, ligand, and DPF files were used for both docking experiments.

   It is worth explaining the other options in the Analyze > Dockings submenu here. Open All... reads all DLG files in the specified directory. Again, this only makes sense when the same AutoGrid map files, ligand, and DPF files were used for all the docking experiments. Clear... removes dockings from ADT, and Select... changes the current docking being analyzed.

4. Click on Analyze > Conformations > Load... to open "ind" in the Conformation Chooser. This displays a concise list of the docked conformations, their cluster ranks, energies, and cluster RMSD values (Fig. 8.14.11).

   The lower panel lists the docked conformations for the ligand, grouped according to the clustering performed at the end of the AutoDock calculation. Clicking once on an entry displays information about it in the upper panel. Double-clicking on an entry updates the conformation of the ligand. The input ligand PDBQT conformation is always the first entry in this list.

   The information displayed in the upper panel includes the rank of the conformation. For example, the best result is always `1_1`. The number before the underscore is the rank of the cluster this result belongs to, while the number after it is the rank in the cluster. Docked Energy is the sum of the intermolecular and internal energy components. Cluster RMS is the root mean square deviation (RMSD) between this docking and the seed (or lowest energy member) of this cluster. `1_1` is the seed for the first cluster, so its Cluster RMS is 0.0. Ref RMS is the RMSD between this docking and the reference structure (specified in the DPF by the `rmsref` command). If no reference structure is specified in the DPF, the input ligand structure is used as the reference. freeEnergy is the sum of the intermolecular energy plus the torsion entropy penalty (which is a constant times the number of rotatable bonds in the ligand), while Ki is calculated from the Docked Energy.

5. Double click on the `ind_1_1` entry to put the ligand in the lowest energy docked conformation. Look at the information displayed in the top panel. Scroll down through the list to see how many clusters were formed with these docking results. Notice the range in energy between the "best" docking and the seed of the highest-energy cluster.

   If the protocol is repeated and the results from a previous AutoDock run are compared with the current ones, it will become clear that the hybrid genetic-algorithm-local search method is stochastic.

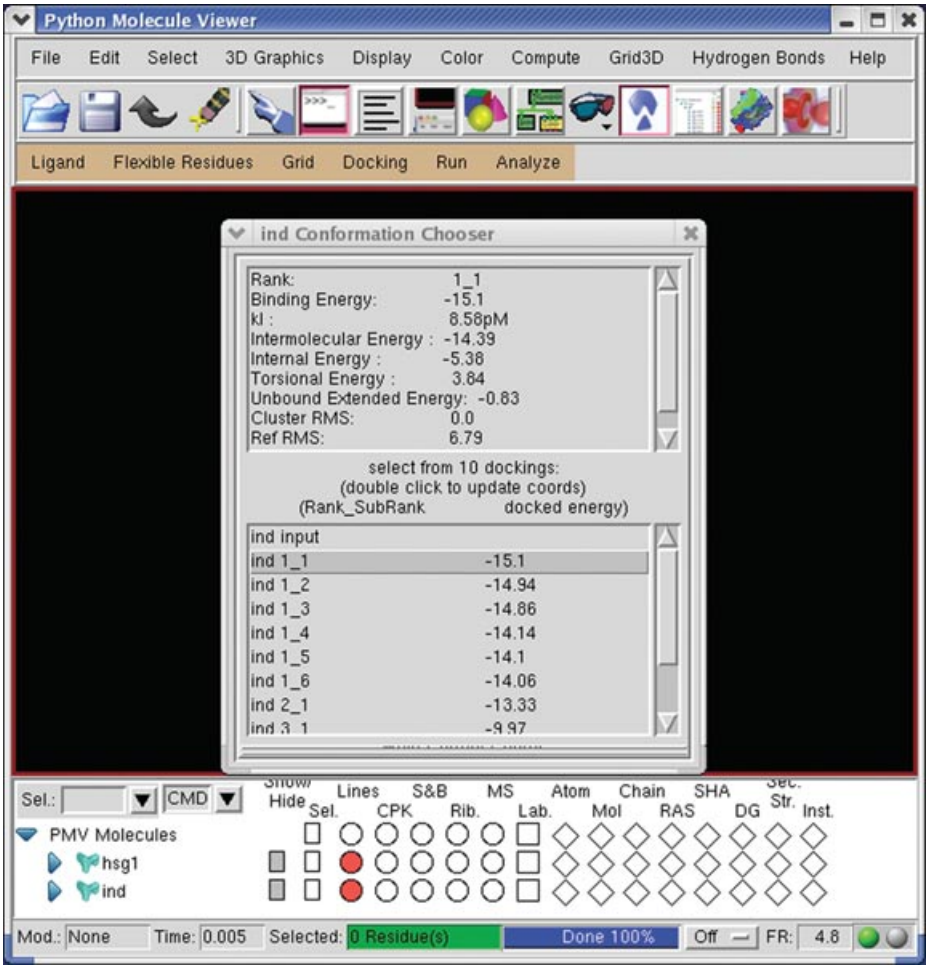


Figure 8.14.11 The docked conformations are listed in the lower part of the Conformation Chooser panel. They are named according to the rank of the cluster to which they belong and their rank in that cluster. Clicking on one of the entries in this list displays information about it in the upper panel; in this case, `ind_1_1` has been selected, which is the lowest energy conformation in the lowest energy cluster. Double-clicking on one of these entries updates the conformation of the ligand in the 3D viewer to the corresponding coordinates.

BASIC PROTOCOL 11

Visualizing Docked Conformations

Visualize the docked conformations of the current Docking instance, which was created earlier by reading `ind.dlg`. The best docking result can be considered to be the conformation with the lowest (docked) energy, or it can be selected based on its RMS deviation from a reference structure, usually the crystallographic binding mode.

At the end of each docking run, AutoDock outputs the conformation with the lowest energy of the ligand it found during that run. This docked conformation consists of a position, orientation, and set of torsion angles, if any, and is characterized by an estimated free energy of binding, which is the sum of the intermolecular energy, the internal energy, and the torsional energy minus the unbound system's internal energy. AutoDock also reports van der Waals energy and an electrostatic energy for each atom.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

AutoDock log file (`ind.dlg`; Basic Protocol 9)
Receptor PDBQT files (Basic Protocol 4): `hsg1.pdbqt` *or* `hsg1_rigid.pdbqt` (if some residues in the receptor were treated as flexible; Basic Protocol 5)

1. Hide the macromolecule and input ligand using the Display > Show/Hide Molecule command and zoom in on the docked ligand using Shift-Button2.
2. Click on Analyze > Conformations > Play.... This opens a Conformation Player (CP) panel that can be used to examine the docked conformations. The CP is shown in the upper part of Fig. 8.14.13. It has buttons with forwards and backwards arrows for controlling the direction of playback.

   The conformation player has a current list of conformations that consists of all of the docked conformations, ordered by run. In this case there are 10 dockings, so the list is [0,1,2,3... 10]: "0" is reserved for the original input conformation.

   The Conformation Player consists of the parts shown in Table 8.14.6, working out from the type-in entry field.
3. Step through the sequence of conformations one by one using the black arrows.
4. Open the Set Play Options widget by clicking on the button with the & symbol. Set the conformation to 4. Change the coloring scheme to "vdw" or "elect_stat" in the dropdown menu labeled "Color by".

Table 8.14.6 Parts of the Conformation Player

Part	Function
Type-in entry (center)	Provides random access to any conformation by its ID. Valid IDs depend on which menu button was last used to start the player. (Fig. 8.14.13 shows conformation `1_1` has been chosen.)
Black arrow (solid triangle) buttons (either side of the type-in entry field)	Moves the user to the next or previous conformation in current list.
White arrow (hollow triangle) buttons	Starts playing through the list of conformations according to current play mode parameters (see below). Clicking again on a white arrow button stops playback. While a play button is active, its icon is changed to double vertical bars.
Double black arrow buttons	Starts playing as fast as possible in the specified direction.
Double black arrow plus line buttons	Advance to the beginning or end of the conformation list.
Ampersand button (&)	Opens the Set Play Options window.
Quatrefoil button (⌘)	Closes the Conformation Player.

5. Set the Play Mode to "continuously in 1 direction" from the Play Mode menu. Click on the forward white arrow. Click this button again to stop play back.
6. Open the Play Parameters widget and set the "start frame" to 1. This excludes the input conformation, which is always conformation 0. Adjust the "frame rate" to 3.
7. Display information about each conformation by opening the Conformation Info widget by clicking on Show Info.

BASIC PROTOCOL 12

Clustering Conformations

An AutoDock docking experiment sometimes produces several low energy solutions. To some extent, the reliability of a docking result depends on the similarity of its final docked conformations (although there are known instances where a ligand actually can bind to the same receptor in more than one quite distinct conformation). One way to measure the reliability of a result is to compare the RMSD of the lowest energy conformations and their RMSD to one another, to group them into families of similar conformations or "clusters."

The DPF keyword "analysis" determines whether clustering is done by AutoDock. It is also possible to cluster conformations with ADT. By default, AutoDock clusters docked results at 0.5 Å RMSD. This process involves ordering all of the conformations by docked energy, from lowest to highest. The lowest energy conformation is used as the seed for the first cluster. Next, the second conformation is compared to the first. If its RMSD is less than the RMSD tolerance, it is added to the first cluster. If not, it becomes a member of a new cluster. This process is repeated with the rest of the docked results, grouping them into families of similar conformations.

First, examine the AutoDock clustering read in from `ind.dlg`, then make new clusterings at different RMS values from the "rmstol" value specified in the DPF.

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

AutoDock log file (`ind.dlg`; Basic Protocol 9)
Receptor PDBQT files (Basic Protocol 4): `hsg1.pdbqt` *or* `hsg1_rigid.pdbqt` (if some residues in the receptor were treated as flexible; Basic Protocol 5)
Map files from the AutoGrid calculation (Basic Protocol 7): `hsg1.*.map` *or* `hsg1_rigid.*.map` (if some residues in the receptor were treated as flexible)

1. Click on Analyze > Clusterings > Show. . . to open an interactive histogram chart. It is labeled `ind:rms = 2.0 clustering` (Fig. 8.14.12).

   The heights of histogram bars indicate the number of docked conformations in each cluster, computed at the specified RMSD. The x-position of each histogram bar is plotted at the energy of the conformation with lowest energy in the cluster. The clusters reported in the AutoDock docking log (DLG) are sorted by the energy of the lowest-energy conformation in that cluster and are initially colored blue.

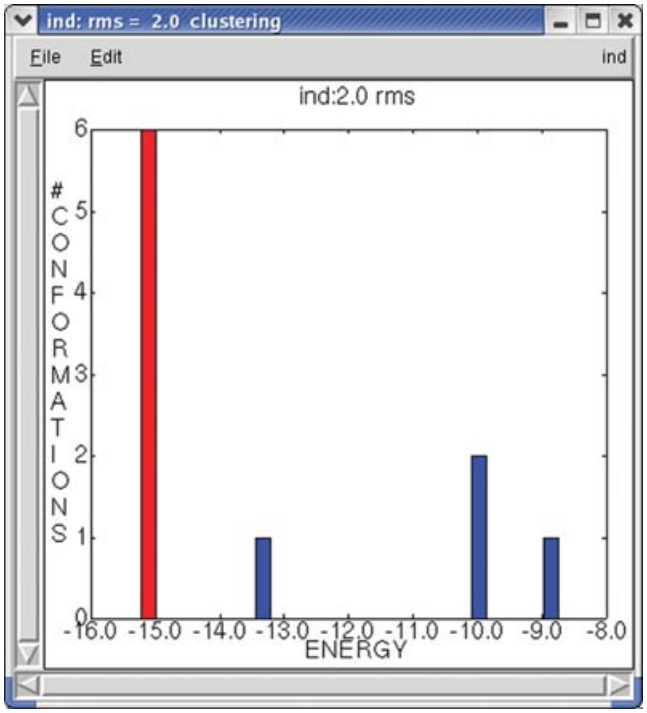


Figure 8.14.12 An interactive histogram. Clicking on a bar in the histogram links the conformations in the corresponding cluster to the player and updates the ligand to the coordinates of the lowest energy conformation in that cluster.

For example, the lowest energy conformation in the second cluster is `2_1` *(and is about −13.5 Kcal/mol in energy in the example in Fig. 8.14.12). Clicking on a bar makes that cluster the current sequence for the ligand's Conformation Player, and the bar's color changes from blue to red.*

The Conformation Rank_Number Info window (lower left of Fig. 8.14.13) shows both refRMS (the RMSD between the current reference and the displayed conformation) and clRMS (the RMSD between the displayed conformation and the lowest energy conformation in this cluster). As described above in the tour of the conformation player, it is possible to set the reference structure to that of any of the docked conformations when it is the current conformation. When viewing clustering results, this is especially useful because it allows the examination of the RMSD between cluster members.

2. To set the reference structure to that of any of the docked conformations when it is the current conformation, choose a cluster and use the arrow key to step forward to its lowest energy conformation, e.g., 1_1. Set the RMS reference to this conformation. Now, stepping through the cluster will show the RMSD between the lowest energy member of this cluster, i.e., 1_1, and the rest of the conformations in this cluster (Fig. 8.14.13).
3. If desired, inspect other clusters by picking a different bar in the interactive histogram.
4. Alternatively, save the histogram as a PostScript file for printing later on by selecting Edit > Write from the interactive histogram's menu to open a file browser and enter a filename. Use `.ps` for the filename's extension.
5. Select File > Exit to close.

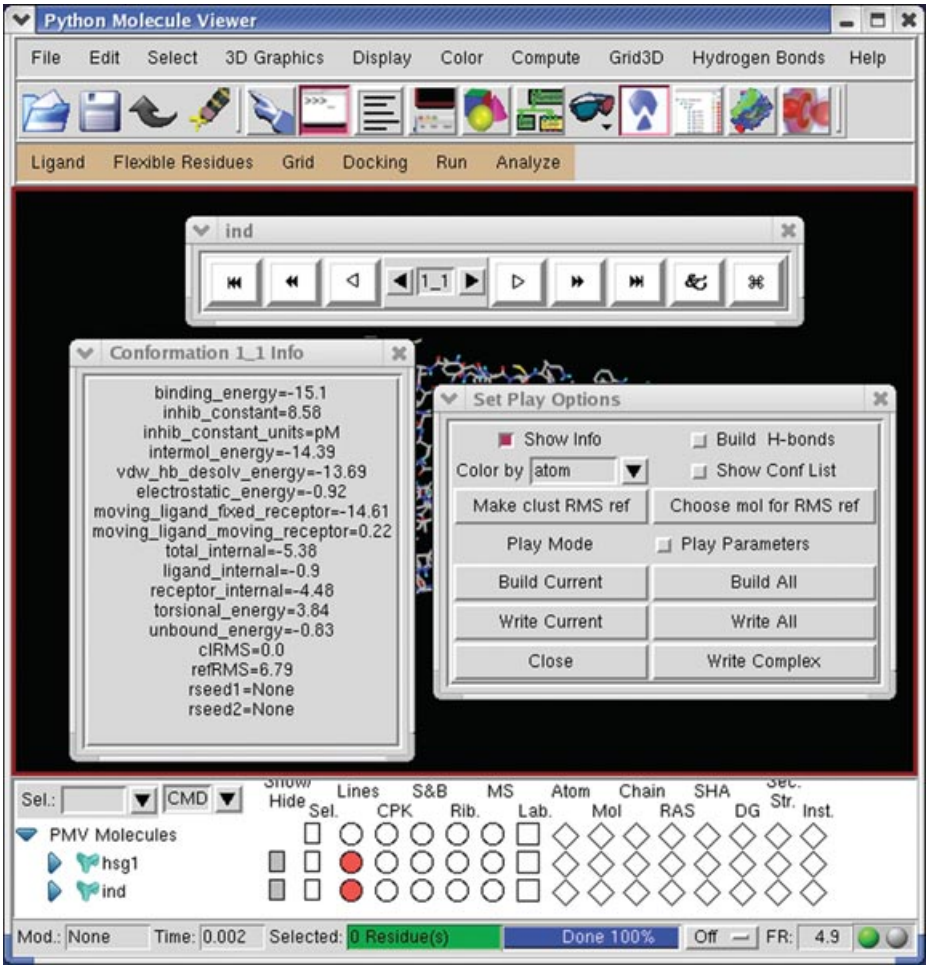


Figure 8.14.13 Open Set Play Options panel to change play options via the ampersand (&) button, and click on "Show Info" to open the "Conformation 1_1 info" widget.

Examine clusters of docked Indinavir molecules

HIV protease is a homodimer and it has C2 symmetry. This means that rotating the molecule by 180° around its axis of symmetry results in a view identical to the original.

6. Build a copy of the lowest energy conformation: cluster 1, conformation 1. First, display it using the conformation player, then click the Build button.

7. Click on the second bar in the histogram and display the lowest energy member of this second cluster by using the arrow keys next to the entry. If this result does not show C2 symmetry, try another cluster bar.

   It should be possible to see the symmetry-related docked conformations (Fig. 8.14.14). Note that since the search method used in the docking is stochastic, the outcome of the docking is random, so the C2-related binding mode may not always be observed.

8. To facilitate comparing the docked conformations, choose File > Preferences > Set Commands to be Applied on Objects and select colorByMolecules. When this is on, every time a new molecule is built or added to the viewer (up to a current limit of 20), it is colored differently.

   Note that when reading several docking logs for the same ligand-receptor pair into ADT, it is necessary to use the Analyze > Clusterings > Recluster. . . option to create a clustering.

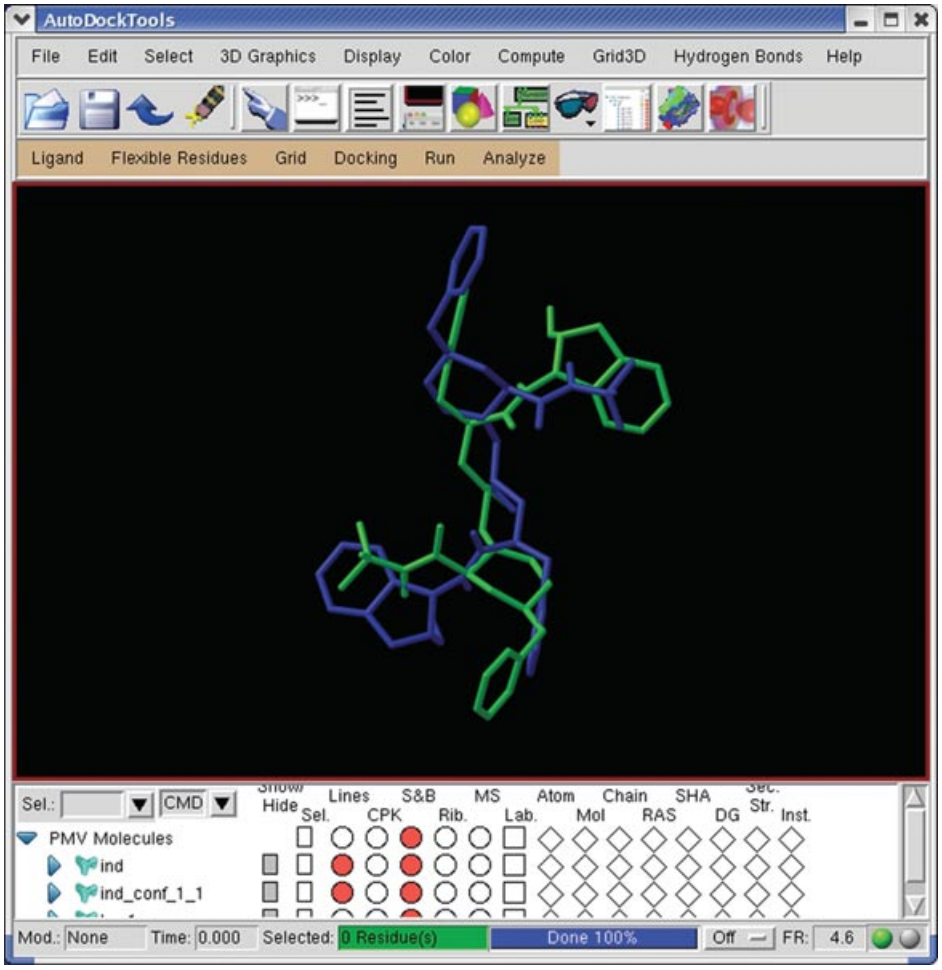


Figure 8.14.14 The C2 symmetry of the binding site of HIV-1 is reflected in these two symmetrically related docked conformations of Indinavir. For the color version of this figure go to *http://www.currentprotocols.com*.

9. Click on Analyze > Clusterings > Recluster. . . and enter a series of new RMS tolerances as floating point numbers, separated by spaces.

   These will be used to perform new clustering operations on the docked results. The time consuming step in clustering is computing a difference matrix between the conformations being compared. Larger RMS values require fewer comparisons; conformations that are more similar require fewer comparisons. A clustering output file will be written by typing a name in the OutputfileName entry; the convention is to use the extension `.clust` *for these files.*

10. If a warning appears that says "Ligand not in input conformation. Do you want to cluster anyway?" click Cancel. Then in the entry field in the middle of the Conformation Player (shown in the upper part of Fig. 8.14.13, labeled "ind") type 0, and press Return or Enter. Then repeat step 9.

    It is important to set the ligand to the original input conformation (numbered 0) before clustering.

11. Type in a list of RMSD tolerances separated by spaces, e.g., `1.0 2.0 3.0`, and click on OK.

    For this example, the reclustering should be very fast. It is possible to visualize the new clusterings by repeating step 1.

Visualizing Conformations in the Complex

Ultimately, the goal of a docking experiment is to illustrate the docked result in the context of the macromolecule, explaining the docking in terms of the overall energy landscape. The interactions between the ligand and the macromolecule are driven by energy composed of van der Waals (vdW), electrostatic, hydrogen bonding, and desolvation component energies.

This protocol has three parts: (1) First, to evaluate the chemical reasonableness of the docking results, represent the macromolecule as a solvent-excluded molecular surface using the MSMS algorithm (a molecular surface calculating method due to Sanner et al., 1996), check whether the ligand has docked in a pocket on the receptor, and check whether the pairwise-interactions between atoms in the ligand and those in the receptor are reasonable. (2) Next, explore the energy landscape of the binding site, representing atomic affinity values and the electrostatic potential using 3D isocontours. This view of the docking can illuminate the observed and predicted binding modes, and in the application of drug design, it can suggest chemical modifications of the ligand that may improve binding affinity. (3) Finally, visualize all the docked structures at once, to inspect the overall binding pattern.

Necessary Resources

Hardware

Platforms (operating systems running on a specific chip architecture): full list of supported platforms available at *http://autodock.scripps.edu/obtaining*

Software

AutoDock, AutoGrid, and AutoDockTools (Basic Protocol 1)

Files

AutoDock log file (`ind.dlg`; Basic Protocol 9)
Receptor PDBQT files: `hsg1.pdbqt` (Basic Protocol 4) *or* `hsg1_rigid.pdbqt` (if some residues in the receptor were treated as flexible; Basic Protocol 5)
Map files from the AutoGrid calculation (Basic Protocol 7): `hsg1.*.map` *or* `hsg1_rigid.*.map` (if some residues in the receptor were treated as flexible)

Show the macromolecule using a molecular surface

1a. *If hsg1 is still in the viewer:* Use Display > Show/Hide Molecule to display it. Undisplay any docked conformations that may have already been built.

1b. *If hsg1 is not currently displayed:* Use Display > Show/Hide Molecule to redisplay it.

1c. *If hsg1 is not present in the viewer:* Use Analyze > Macromolecule > Open... instead.

If `hsg1.pdbqt` *cannot found in the current directory, a file browser opens to ask where it can be found.*

2. Click on Analyze > Macromolecule > Choose... to link `hsg1` to the current docking.

3. Click on Select > Direct Select to open a Direct Select widget, where it is possible to pick a molecule, chain or named saved set.

4. Click on Molecule List . . . to display check-buttons for hsg1 and ind. Click on hsg1. Click on Dismiss to close the widget.

5. Click on Compute > Molecular Surface > Compute Molecular Surface. This opens an MSMS Parameters Panel: widget where it is possible to set the probe radius and density parameters for a molecular surface (MSMS) computation.

   The density parameter controls the quality of the calculated mesh.

6. Increase the Density to 10. Click on OK to start the computation.

7. Click on Color > by DG colors, then choose the MSMS-MOL geometry and finally click OK. The molecular surface will be colored according to the David Goodsell coloring scheme based on the element of the nearest atom (Fig. 8.14.15).

   This view of the docking shows how the docked ligand fits into the macromolecule.

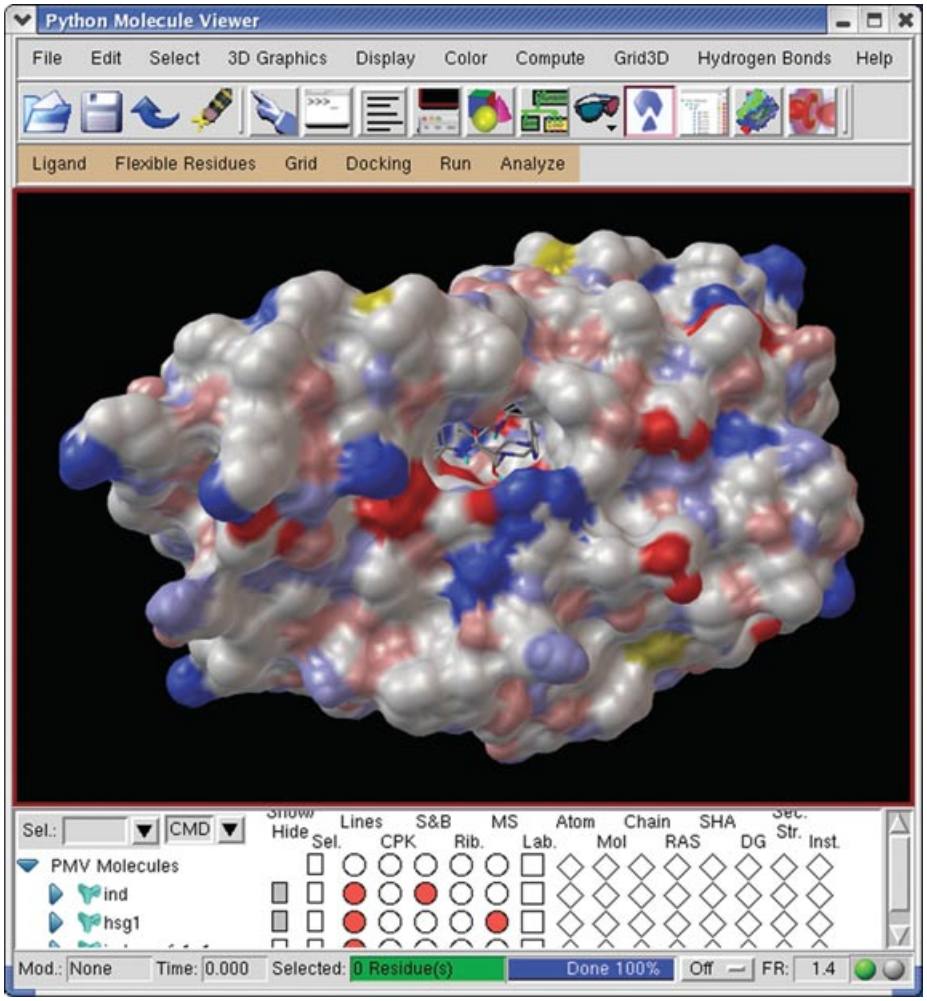


Figure 8.14.15 Indinavir docked into a pocket in HIV-1. Here the molecular surface has been colored using DG colors, the ligand is displayed as ball-and-sticks, and the complex has been rotated using the middle mouse button to show the active site tunnel. In the DG color scheme (created by David Goodsell and available as a setting option in ADT and the related molecular viewer PMV), neutral oxygen and nitrogen atoms are pink and light blue, respectively, while charged oxygen and nitrogen atoms are red and dark blue, respectively. This has the effect of highlighting the charged parts of charged amino acids: the acidic side chains Asp and Glu appear red, while basic side chains Arg and Lys appear blue. For the color version of this figure go to *http://www.currentprotocols.com*.

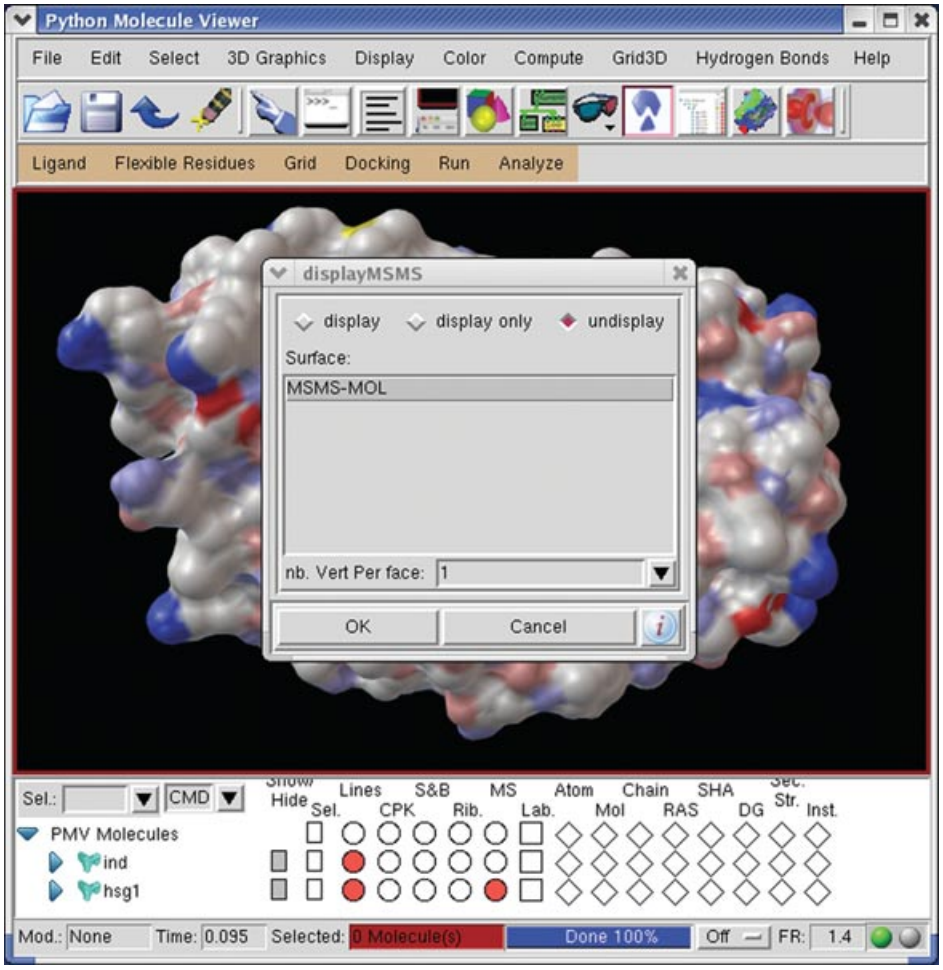


Figure 8.14.16 The "displayMSMS" widget is used to select specific molecular surfaces and set their visibility. Here the undisplay radio-button is checked, so clicking on the OK button will result in undisplaying the molecular surface, which is named MSMS-MOL.

8. Use the Un/Display > Molecular Surface menu option to hide the molecular surface (Fig. 8.14.16).

Visualize the docked conformations using atomic affinity grid maps

It can be very instructive to visualize the docked conformations in the context of the atomic affinity grid maps. This may be particularly useful for computer-aided drug design. Note that in ADT, the grid isocontours are colored by atom type. In the steps that follow, ADT will be used to plot the oxygen affinity map calculated by AutoGrid as a 3D isocontour, to show how a key oxygen atom of Indinavir binds in a pocket of oxygen affinity between the two catalytic ASP25 residues of the HIV-1 Protease molecule.

9. Click on Analyze > Grids > Open. . .. This opens a list chooser of the grids used in this docking.

10. Select the oxygen affinity map `hsg1.OA.map` and click OK. The AutoGrid map file is read into the viewer and visualized as an isocontour in 3D.

Adjust the isocontour value for the oxygen affinity map

Every point in the grid box where the energy value computed by AutoGrid is equal to the isocontour level will be connected together by lines or polygons. It is possible to change the value of the isocontour level, which is an energy in Kcal/mol; the step between grid

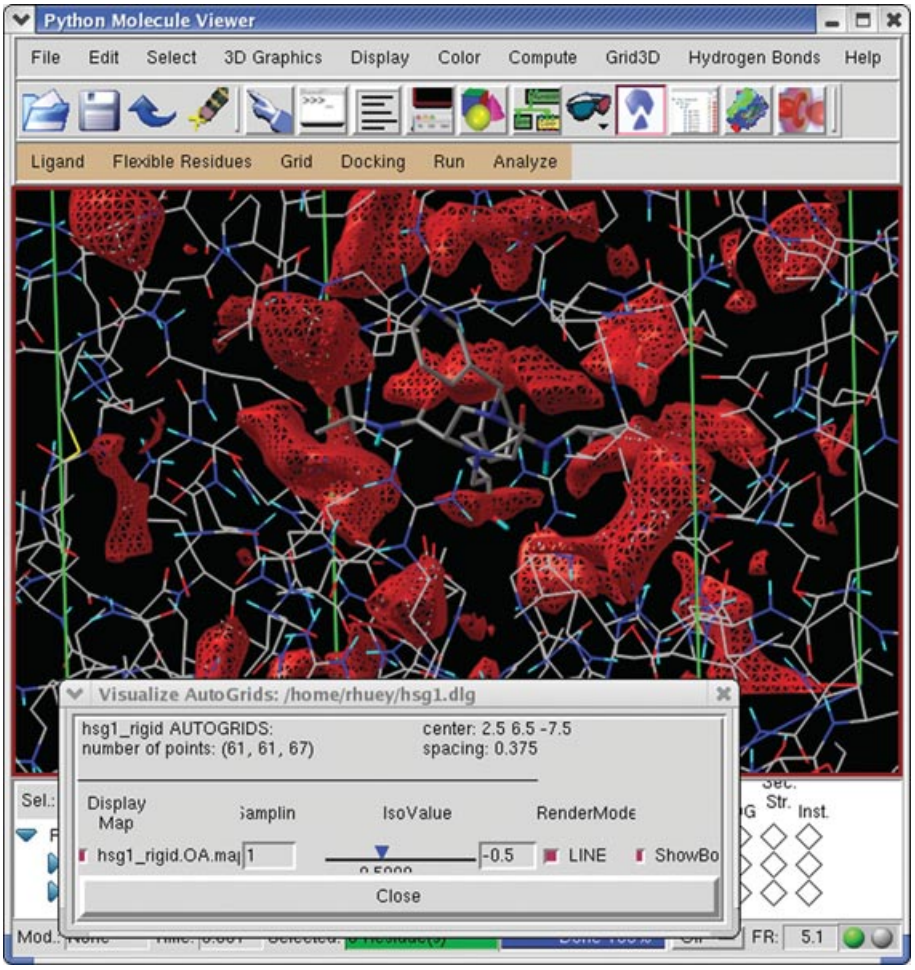


Figure 8.14.17 The isocontour value to display can be set to −0.5 Kcal/mol by dragging the small blue triangle in the Visualize AutoGrid widget to the left, or by positioning the cursor over the entry to the left of the LINE check button and typing `-0.5` followed by the Return or Entry key. For the color version of this figure go to *http://www.currentprotocols.com*.

points for sampling the grid values; and whether to show the isocontoured regions as lines or filled (solid) polygons. It is possible to also toggle the visibility of the Grid and its bounding box. The following illustrates the kind of information obtainable from the atomic affinity grid maps (Fig. 8.14.17).

11. Set the IsoValue to `−0.5`; if this value is typed into the slider entry, remember to press Return or Enter.
12. Set the Sampling to `1` and press Return or Enter.
13. Display `hsg1.pdbqt`; if it is not present in the viewer, use Analyze > Macromolecule > Open. . ..
14. Choose Select > Select From String and type in `ASP25` into the Residue field and then click Select.
15. Click Yes to change selection level if necessary and Dismiss to close the Select From String widget.
16. Choose Display > Sticks And Balls to open the Display Sticks and Balls widget. Increase the quality to `15` and click OK.

17. Choose Color > By Atom Type and select “balls” and “sticks” in the widget that opens, and click OK.

18. Choose a low-energy docked conformation using the conformation player.

19. Rotate the molecules in the viewer.

    Note that an oxygen atom in the inhibitor IND201:O2 is buried in a bow-tie shaped pocket of Oxygen-affinity. After using Build (see below) to construct other low-energy docked conformations, the same O2 atom should be observed sitting in this region.

20. Click Display Map and Show Box to undisplay the isocontour and its bounding box before clicking Dismiss.

Visualize all the docked confirmations at once

It can be useful to visualize all the docked conformations at once by placing spheres, one for each docking, at the center of each docked conformation.

21. Click on Analyze > Dockings > Show as Spheres. . .. This command represents each docked conformation by a sphere. A sphere is placed at the average position of the coordinates of all the atoms in each conformation.

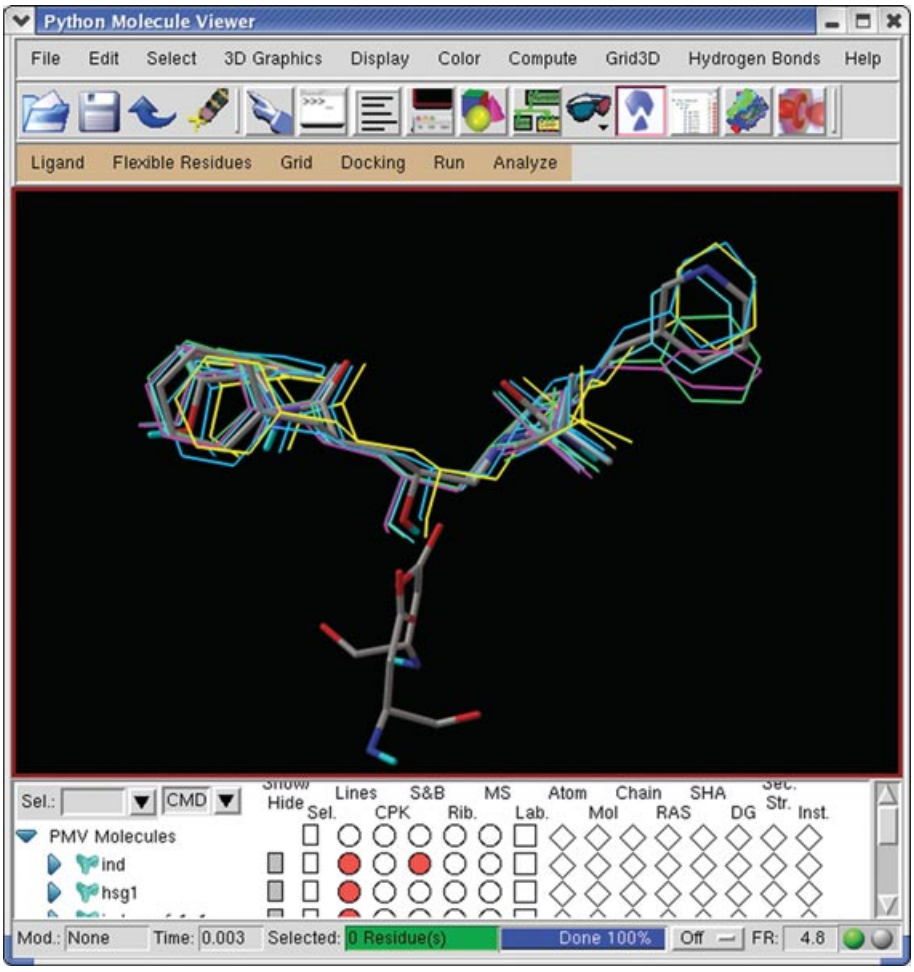


Figure 8.14.18 Building new molecules can show all of the docked conformations in a cluster simultaneously for each one. In the Set Play Options widget, click on Build All. Use the Color > By Molecules menu option to differentiate among the docked structures. For the color version of this figure go to *http://www.currentprotocols.com*.

22. Click on `ind.dlg` in the list. Clicking on the name of a docking log in the list makes the spheres representing its results visible only if the associated ligand is visible.

23. Reduce the radii to 0.1 Å to see the different docking positions more distinctly.

    It is possible to change the radii of the spheres, their color, and their smoothness (or quality).

24. Click on the ampersand symbol (&) in the conformation player and then click Build All. This builds a new molecule for each docked conformation in the current set bound to the conformation player.

25. Use Color- > By Molecules to help distinguish the various docked poses (Fig. 8.14.18).

GUIDELINES FOR UNDERSTANDING RESULTS

Docking has been shown to be very valuable for lead discovery (Shoichet et al., 2002), but it should be stressed that docking methods are not perfect, and even though a docked conformation may be found and is predicted to have a good binding energy, it does not mean that this compound will bind when assayed. Scoring functions are far from perfect, owing to the many approximations that must be made when trying to perform a docking in a reasonable amount of time. Indeed, current methods have trouble rank-ordering the list of hits, and successes in virtual screening may be due more to filtering out compounds that are "wrong for the target, rather than selecting those that are right" (Leach et al., 2006). There have been several excellent publications that compare docking methods and scoring functions (Kontoyianni et al. 2004; Warren et al. 2006) and provide some cautions on how such comparisons should be carried out (Cole et al., 2005). More recently, molecular docking has been combined with pharmacophore-based methods, while ligand-based screening has been shown to be increasingly useful in virtual screening when the structure of the target receptor is unknown (Kontoyianni et al. 2008).

COMMENTARY

Background Information

Ligand-protein docking

AutoDock has become one of the most widely cited ligand-protein docking software packages (Sousa et al., 2006), and even though it still supports the Monte Carlo simulated annealing search method that was introduced in version 1, the vast majority of AutoDock's applications use the more efficient Lamarckian genetic algorithm described in this unit. The protocols show how to use ADT to set up AutoGrid and AutoDock calculations and how to analyze these results. However, it is not the only GUI designed for AutoGrid and AutoDock. AutoDock has been shown to be useful for so-called blind docking (Hetenyi and van der Spoel, 2002, 2006), where the binding pocket is not known, and this is where an alternative, third-party GUI called Blind Docking Tool (BDT; Vaque et al., 2006), can be used to help set up the grid maps required to span the entire protein for such calculations.

This unit cannot cover the many methods for preparing the input structures for docking. Indeed, many of these methods are covered elsewhere in other protocols. However, the GIGO maxim (garbage in, garbage out) applies here, too. It is necessary to ensure that all the amino acid side chains are properly reconstructed if, for example, they are missing in the original x-ray crystal structure; this is especially important near the active site. Any ionizable side chains should be properly protonated at the desired pH. The hydrogen bonding network in the protein should be optimized to ensure that amido-containing and imidazole-containing side chains use the properly assigned rotamers. Water molecules in the receptor are sometimes necessary for ligand binding, and assigning the correct positions for the polar hydrogens in the water molecules can be vital for proper recognition of the ligand. When present, cofactors should also be properly prepared, with particular attention being paid to the partial atomic charges on any metal

atoms that may be present; these should reflect its ionization state.

Protein flexibility

Although AutoDock 4 introduced the ability to model side chain flexibility in the receptor, and while this can be a very useful approximation in certain classes of docking problems, there are clearly many more degrees of freedom in the receptor that are not explored by AutoDock during such dockings. Macromolecular flexibility is increasingly gaining the proper recognition it deserves in molecular simulation as a key aspect of molecular recognition (Cozzini et al., 2008). In the context of flexible ligand-protein docking, McCammon and co-workers have introduced the relaxed complex method (Lin et al., 2002, 2003; McCammon, 2005), in which the flexibility of the target is effectively pre-calculated using molecular dynamics calculations carried out on the apo form. Then selected snapshots from the trajectory are used to compute AutoGrid maps and perform dockings using AutoDock, although this approach lends itself to any flexible ligand-protein docking software. Indeed, this approach helped to identify a novel binding pocket in HIV integrase (Schames et al., 2004).

Critical Parameters and Troubleshooting

Convergence

Evaluate the convergence of the dockings to determine the thoroughness of the search, by clicking on Analyze > Clusterings > Show. . .. If the independent dockings produce a small number of conformationally similar clusters (and preferably just one), then the docking searches have used a large enough number of evaluations. However, if the results do not show reasonable clustering, it is advisable to repeat the docking calculation with an increased number of evaluations, set using the `ga_num_evals` command in the DPF. In general, the more torsions a ligand has, the more evaluations will be needed for each docking. Docking ligands with more than 8 to 10 active torsions usually requires increasing the number of evaluations by a factor of ten or more, to the million to ten million range. If the number of torsions exceeds ∼15, then it will be difficult for the Lamarckian GA to find a good binding mode, and it is advisable to reduce the number of rotatable bonds in the ligand by fixing the torsion angles at some reasonable value.

GA and Lamarckian GA

In general, when using the GA and Lamarckian GA, and keeping all the other parameters constant, better results can be obtained by using a population size larger than 50, typically 200 to 300 individuals.

Repeating docking

It is a good idea to repeat a given docking 50 to 100 times, to obtain a good sample of binding modes for conformational cluster analysis.

Ligands too large for the binding pocket

If only highly-positive-energy conformations are found at the end of a docking, this may be because the ligand is too large for the binding pocket, or even too large for the grid box. Visual inspection of the docking results using ADT's Analyze menu should help to reveal if this is the case. If this happens, try increasing the size of the grid box by increasing either the grid spacing and/or the number of grid points. Alternatively, try docking a smaller ligand.

Ligands that bind in free space away from the receptor

If the ligand appears to bind in free space far away from the receptor of interest, make sure that the grid box is centered on the receptor. The location of the grid box with respect to the protein can be viewed using ADT's Analyze > Grids > Open. . . and the Analyze > Macromolecule > Open. . . menu items.

Chemical reasonableness

Evaluate how chemically reasonable the best results are by examining the interactions between the receptor and the best docked conformation(s). Click on the lowest energy cluster in the clustering histogram. Put the ligand in the lowest energy conformation using the Conformation Player. Click on Analyze > Macromolecule > Choose. . . to look at the interactions between the ligand and nearby atoms in the receptor and consider the following:

Is the ligand bound inside a pocket in the receptor?

Are the chemical interactions complementary? Are nonpolar atoms in the ligand docked near nonpolar atoms in the receptor? Are polar atoms in the ligand docked near polar atoms in the receptor? Are negatively charged atoms in one molecule found near positively charged atoms in the other?

If it is already known that a particular residue or residues in the protein interact with the ligand, and is that interaction observed in the docked result?

Do the interactions seem reasonable in the context of what is known about the ligand-receptor complex from experimental results, e.g., mutation studies?

Failure to redock

Sometimes, a failure to re-dock a ligand into a protein of known X-ray crystallographic structure can indicate that the ligand should adopt a different tautomeric form than the one that was docked, or that a key side chain in the protein should be neutral instead of charged. In these cases, it is advisable to try docking alternative tautomeric forms of the ligand, or to build a new set of grid maps based on the alternative protonation of the protein's side chain(s). Also, remember that some bound water molecules are never displaced by the binding of a ligand, and instead, the tightly bound solvent molecule remains as part of the protein structure, and the ligand interacts with it as though it were a polar protein atom. If such a solvent molecule is not included in the receptor site, the docking may fail.

Suggestions for Further Analysis

The interpretation of AutoDock results is open-ended. The field of drug design requires chemical insight and creativity, and docked conformations of the ligand may suggest chemical modifications, e.g., side-group substitutions. It is worth noting that in the pharmaceutical industry, medicinal chemists may visually inspect hundreds of docked structures for chemical reasonableness during the drug discovery process.

Acknowledgments

This is manuscript number 18481 from The Scripps Research Institute. The authors are grateful for funding provided by R01-GM069832.

Literature Cited

Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., and Bourne, P.E. 2000. The Protein Data Bank. *Nucleic Acids Res.* 28:235-242.

Chen, Z., Li, Y., Chen, E., Hall, D.L., Darke, P.L., Culberson, C., Shafer, J.A., and Kuo, L.C. 1994. Crystal structure at 1.9-Å resolution of human immunodeficiency virus (HIV) II protease complexed with L-735,524, an orally bioavailable inhibitor of the HIV proteases. *J. Biol. Chem.* 269:26344-26348.

Cole, J.C., Murray, C.W., Nissink, J.W., Taylor, R.D. and Taylor, R. 2005. Comparing protein-ligand docking programs is difficult. *Proteins* 60:325-332.

Cozzini, P., Kellogg, G.E., Spyrakis, F., Abraham, D.J., Costantino, G., Emerson, A., Fanelli, F., Gohlke, H., Kuhn, L.A., Morris, G.M., Orozco, M., Pertinhez, T.A., Rizzi, M., and Sotriffer, C.A. 2008. Target flexibility: An emerging consideration in drug discovery and design. *J. Med. Chem.* 51:6237-6255.

Gasteiger, J. and Marsili, M. 1978. A new model for calculating atomic charges in molecules. *Tetrahedron Lett.* 34:3181-3184.

Goodsell, D.S. and Olson, A.J. 1990. Automated docking of substrates to proteins by simulated annealing. *Proteins* 8:195-202.

Hetenyi, C. and van der Spoel, D. 2002. Efficient docking of peptides to proteins without prior knowledge of the binding site. *Protein Science* 11:1729-1737.

Hetenyi, C. and van der Spoel, D. 2006. Blind docking of drug-sized compounds to proteins with up to a thousand residues. *FEBS Lett.* 580:1447-1450.

Huey, R., Morris, G.M., Olson, A.J., and Goodsell, D.S. 2007. A semi-empirical free energy force field with charge-based desolvation. *J. Comput. Chem.* 28:1145-1152.

Kontoyianni, M., McClellan, L.M. and Sokol, G.S. 2004. Evaluation of docking performance: Comparative data on docking algorithms. *J. Med. Chem.* 47:558-565.

Kontoyianni, M., Madhav, P., Suchanek, E. and Seibel, W. 2008. Theoretical and practical considerations in virtual screening: A beaten field? *Curr. Med. Chem.* 15:107-116.

Leach, A.R., Shoichet, B.K. and Peishoff, C.E. 2006. Prediction of protein-ligand interactions. Docking and scoring: Successes and gaps. *J. Med. Chem.* 49:5851-5855.

Lin, J.H., Perryman, A.L., Schames, J.R., and McCammon, J.A. 2002. Computational drug design accommodating receptor flexibility: The relaxed complex scheme. *J. Am. Chem. Soc.* 124:5632-5633.

Lin, J.H., Perryman, A.L., Schames, J.R., and McCammon, J.A. 2003. The relaxed complex method: Accommodating receptor flexibility for drug design with an improved scoring scheme. *Biopolymers* 68:47-62.

McCammon, J.A. 2005. Target flexibility in molecular recognition. *Biochim. Biophys. Acta* 1754:21-24.

Morris, G.M., Goodsell, D.S., Huey, R., and Olson, A.J. 1996. Distributed automated docking of flexible ligands to proteins: Parallel applications of AutoDock 2.4. *J. Comput. Aided Mol. Des.* 10:293-304.

Morris, G.M., Goodsell, D.S., Halliday, R.S., Huey, R., Hart, W.E., Belew, R.K., and Olson, A.J. 1998. Automated docking using a Lamarckian genetic algorithm and an empirical binding free

energy function. *J. Comput. Chem.* 19:1639-1662.

Sanner, M.F., Olson, A.J., and Spehner, J.C. 1996. Reduced surface: An efficient way to compute molecular surfaces. *Biopolymers* 38:305-320.

Schames, J.R., Henchman, R.H., Siegel, J.S., Sotriffer, C.A., Ni, H., and McCammon, J.A. 2004. Discovery of a novel binding trench in HIV integrase. *J. Med. Chem.* 47:1879-1881.

Shoichet, B.K., McGovern, S.L., Wei, B., and Irwin, J.J. 2002. Lead discovery using molecular docking. *Curr. Opin. Chem. Biol.* 6:439-446.

Sousa, S.F., Fernandes, P.A., and Ramos, M.J. 2006. Protein-ligand docking: Current status and future challenges. *Proteins* 65:15-26.

Vaque, M., Arola, A., Aliagas, C., and Pujadas, G. 2006. BDT: An easy-to-use front-end application for automation of massive docking tasks and complex docking strategies with AutoDock. *Bioinformatics* 22:1803-1804.

Warren, G.L., Andrews, C.W., Capelli, A.M., Clarke, B., LaLonde, J., Lambert, M.H., Lindvall, M., Nevins, N., Semus, S.F., Senger, S., Tedesco, G., Wall, I.D., Woolven, J.M., Peishoff, C.E., and Head, M.S. 2006. A critical assessment of docking programs and scoring functions. *J. Med. Chem.* 49:5912-5931.