Knowledge-based approaches for modelling the 3D structural interactome

Anisah W. Ghoorah

Département de formation doctorale en informatique **École doctorale IAEM Lorraine**
Collégium Sciences et Technologies

Extraction de Connaissances pour la Modélisation tri-dimensionnelle de l'Interactome Structural

THÈSE

présentée et soutenue publiquement le 22 novembre 2012

pour l'obtention du

Doctorat de l'Université de Lorraine

(spécialité informatique)

par

Anisah W. Ghoorah

Composition du jury

Rapporteurs :	David Sherman	DR INRIA Bordeaux
	Pierre Tufféry	DR INSERM, Université Paris Diderot
Examinateurs :	Bernard Girau	PR Université de Lorraine
	Pascale Kuntz-Cosperec	PR Université de Nantes
	Anne Poupon	DR CNRS, INRA Tours
	Malika Smaïl-Tabbone	MC Université de Lorraine
Directeurs de thèse :	Marie-Dominique Devignes	CR CNRS, INRIA Nancy
	David Ritchie	DR INRIA Nancy

Équipe ORPAILLEUR – INRIA Nancy Grand Est
Laboratoire Lorrain de Recherche en Informatique et ses Applications (LORIA)
UMR 7503 - Campus Scientifique - BP 239 - 54506 Vandœuvre-lès-Nancy Cedex

Mis en page avec la classe thloria.

Table of Contents

Chapter 1

Introduction

1.1 The Protein Interactome

Proteins are one of the main macromolecular components of life, and protein-protein interactions (PPIs) are central to many biological processes. Recently, the term interactome has been coined to describe the set of all PPIs in the cell. The human interactome has been estimated to involve about 130,000 PPIs (Venkatesan *et al.*, 2009). Understanding in detail how the interactome works could improve our understanding of these biological processes in the cell. Furthermore, because each PPI involves a physical three-dimensional (3D) interaction, knowing how proteins interact at the structural level is crucial for understanding the molecular basis of these biological processes. Studying PPIs on a large scale has gained much attention from both academic and corporate scientific organizations because understanding which proteins interact could be very useful for drug development since almost all current drugs are directed against proteins (Pandey and Mann, 2000). Therefore, adding 3D structural information to all the PPIs in the interactome (i.e. defining the *structural interactome*; Figure 1.1) should enrich our understanding of the biological machinery in the cell for therapeutic purposes.

1.2 Modelling 3D Structures of Protein-Protein Complexes

Recent advances in molecular biology techniques and high-throughput technologies have allowed important progress to be made towards a comprehensive coverage of protein-protein interactions and the 3D structures of protein-protein complexes. Currently, X-ray crystallography is the main 'gold standard' experimental technique used to obtain 3D structures of protein-protein complexes. This technique accounts for over 88% of the structures deposited in the Protein Data Bank (PDB; Berman *et al.*, 2002), the main worldwide repository of protein 3D structures.

However, recent analyses showed that currently only a small fraction of PPIs have a 3D structure in the PDB. For example, it has been estimated that the PDB contains structures of protein complexes for only 8% of the human PPIs (Stein *et al.*, 2011). This shortage of structural PPI data is mainly due to limitations of current experimental techniques. For example, the transient nature

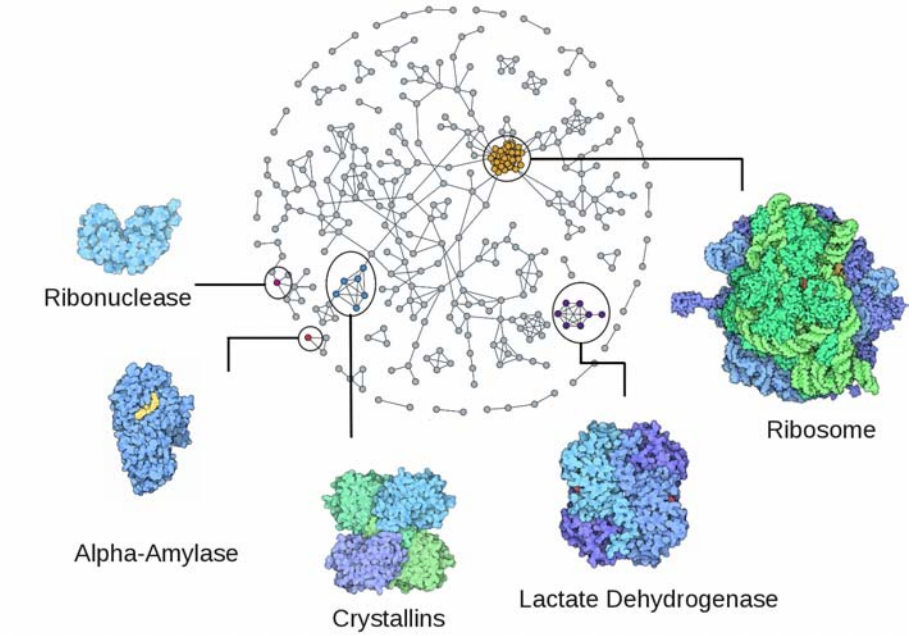


Figure 1.1: An illustration of the structural interactome. Every PPI or protein assembly in the interactome has a 3D structure. Figure adapted from Aloy and Russell (2004). Protein images were obtained from the RCSB-PDB "molecule of the month" archive.

of some PPIs make crystallization difficult. Hence, computational methods have been developed to bridge this gap.

Computational modelling approaches aim to calculate the 3D structure of a protein-protein complex starting from either the individual sequences (homology-based modelling) or the individual structures (*ab-initio* docking) of the constituent proteins. Although good progress has been made, it has been shown that the reliability of the predictions is improved if experimental information from related PPIs is incorporated (Lensink and Wodak, 2010).

Previous studies have shown that pairs of proteins with more than 25% sequence similarity often interact in a similar way (Aloy *et al.*, 2003). This suggests that fairly accurate 3D models of protein-protein complexes can be obtained if there exist structural templates. Moreover, several studies have shown that proteins often interact via just one or a small number of binding sites (Shoemaker *et al.*, 2006, Kim *et al.*, 2006), which suggests that proteins often re-use their binding sites. Recently, Kundrotas *et al.* (2012) estimated that 47% of known human PPIs can be modelled using existing templates. Furthermore, Skolnick conjectured that there now exists a representative set of protein-protein interfaces from which new interfaces can be modelled (Gao and Skolnick, 2010). This suggests that modelling the 3D structures of most/all putative PPIs could be an achievable task, although it will be an enormous endeavour. Hence, there is much interest in developing efficient computational methods to organize and describe current knowledge of structural PPIs to facilitate the re-use of 3D information in modelling of PPIs.

1.3 Structural PPI Resources

Currently, a large part of bioinformatics focuses on analyzing relationships between protein sequences. Relatively little work has been done on describing 3D structures of protein complexes. Furthermore, although the volume of structural PPI data is much less than protein sequence data set, reasoning about 3D structures is much harder. This is mainly due to the three-dimensional aspect of protein structures and the intricate nature of their interactions. Hence, there is a need to organize and describe in a systematic way all structural PPI data in order to facilitate the reasoning about 3D structures and re-using 3D information to model unknown PPIs. Several groups have collected and classified the structures of protein-protein complexes in the PDB with the aim of analysing PPIs. Some recent structural PPI databases and classifications will be described in the following chapters (Chapter 2 and 4).

Current structural PPI and DDI databases clearly constitute useful bioinformatics resources. However, it is not yet straight-forward to use them to model unknown protein-protein complexes. For example, human expertise is needed to gather information from different resources and to interpret the information found in order to guide 3D interaction modelling. Moreover, in many cases, there are multiple database hits and the expert has to go through each of them by hand and by eye. This tedious task often means that modelling approaches do not predict as accurately as possible 3D structures of protein-protein complexes because all available PPI information was not used. Additionally, knowledge of existing protein interaction modes is often not available in an easily accessible way, and so cannot easily be incorporated into docking algorithms. Indeed, it can require much effort to use such databases to help model even a single unknown protein-protein complex. One could even argue that with the growing number of structural data in public databases, manual data retrieval and analysis has become impractical. Hence, there is a need to develop a more systematic classification of PPIs in order to facilitate automatic PPI analysis and to improve the prediction of protein-protein complexes.

1.4 Knowledge Discovery in Databases and Data Mining

Knowledge discovery in databases (KDD) is a powerful technique for detecting patterns in large data sets. Therefore, it is reasonable to suppose that KDD techniques could be applied to structural PPI data. This should help to provide a better 3D picture of the known protein interactome, and to guide predictions of the 3D structures of unsolved protein complexes.

In general, KDD is a process in which voluminous low-level data is transformed into a more compact, abstract, and useful form (Fayyad *et al.*, 1996). The KDD process is often iterative and interactive, and consists of processing large volumes of data in order to extract information or knowledge that is non-trivial, potentially useful, significant, and reusable (Napoli, 2005). The KDD process involves several steps, some of which are understanding the domain application, preparing the data, finding useful features that describe the data, performing data mining, and interpreting the patterns and models found. [1] In practice, the data preparation step consists of several sub-steps such as

[1] An introduction to KDD and data mining can be found in Hand *et al.* (2001) and Han and Kamber (2005).

data selection, data enrichment, data transformation, and data integration. Data selection is the stage at which we identify and collect data sets or databases to work with. Data enrichment is the process of adding additional information to the data sets or accessing additional resources to obtain extra information. Data transformation is the task of converting raw data into a more convenient form for data analysis. Data integration/linkage is the non-trivial task of combining several data sources, eliminating redundancy and conflicts to create a single dataset. The main KDD steps are illustrated in Figure 1.2. The iterative nature of the KDD process emphasises the need to organise the data in a structured way (Hand *et al.*, 2001). Consequently, the data preparation step is often the one which requires the most amount of human expertise and effort.

While KDD refers to the overall process of discovering useful knowledge from data, data mining refers to a particular step in this process. Data mining may be defined as the automatic or semi-automatic data analysis step within the KDD process. Data mining usually involves applying algorithms which are often grouped as classification, clustering and association, for example. Fayyad *et al.* (1996) define two types of goal in data mining, namely (i) the verification of a hypothesis, and (ii) the discovery of patterns/models. The discovery goal is either a predictive or a descriptive type of goal. Prediction involves using attribute values of instances in a database to predict unknown attribute values of a new target instance. Description focuses on finding non-trivial and meaningful patterns or models describing the data. For example, given a data set in which instances are defined by a set of attributes and where one of the attributes is a class label, the task of a classification algorithm is to construct a model which describes the data set. The model is often expressed as a set of rules which should be able to predict to which class a new instance belongs. Decision trees are examples of rule-based classifications. On the other hand, a clustering algorithm is applied to a dataset when one wants to identify classes. This is done by grouping similar instances into clusters.

In KDD, discovering new knowledge can often involve defining new concepts (the knowledge to be learned or described). The output of the KDD process is termed concept description, which ideally brings out and describes in a concrete way previously unknown or non-obvious relationships in a data set. Depending on the KDD concepts and the data mining algorithms used, concept descriptions are not necessarily expressed as rules. For example, they may be described in terms of closely related instances. The key aspect is that they are described and represented in an easy way that can be readily used to solve new problems.

1.5 Thesis Aims and Objectives

This thesis aims to develop a systematic knowledge-based approach for representing, describing and manipulating all available 3D protein-protein interactions. This approach should provide an easy way to access and exploit knowledge of existing PPIs in order to study 3D interactions on a large scale and to help develop new knowledge-based ways of analysing interfaces and predicting protein-protein complexes. For example, from the homology principle, we expect that using knowledge of the spatial organization of binding sites in protein families could provide a way to guide protein docking calculations by focusing the calculation around only those binding sites that are employed by homologous proteins. It therefore seems reasonable to suppose that using such techniques should help to reduce the number of false positives and improve the accuracy of docking predictions.

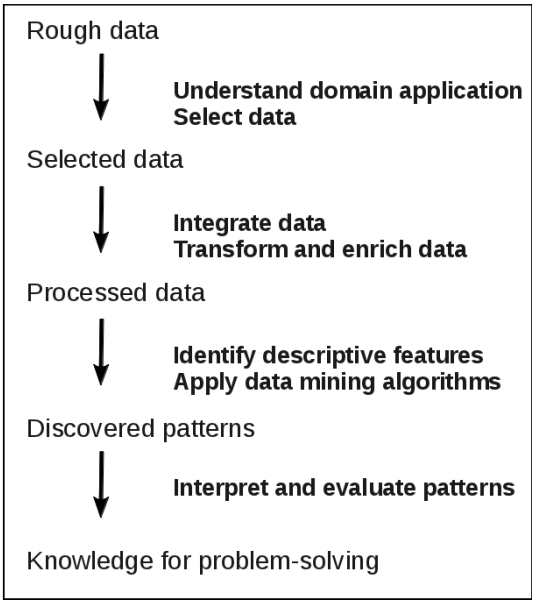


Figure 1.2: The KDD cycle. Figure adapted from Napoli (2005).

The specific objectives here are: (i) to develop a classification of interaction modes in protein domain families in order to describe and summarise PPI data; (ii) to develop a systematic approach to re-use structural knowledge of existing PPIs to facilitate 3D PPI modelling (in particular, docking by homology); (iii) to create a framework to support large scale analyses of 3D protein-protein interface features; and (iv) to provide a structural PPI search engine to facilitate docking by homology.

Since conventional data mining techniques cannot be directly applied to protein structural data, we need to adapt the usual KDD approach to achieve these goals. Hence, describing PPIs in a computationally useful and automatic way is extremely challenging. However, it is necessary at this point to emphasize that this thesis does not present a new data mining algorithm *per se*. Rather, this thesis tackles a difficult problem of describing 3D PPIs to facilitate the reuse of knowledge of existing PPIs to solve unknown PPIs.

1.6 Overview of Thesis

The rest of this thesis is organised as follows. Since developing an understanding of the application domain is essential in any KDD process, a general introduction to proteins, protein-protein interactions, and their 3D structures is given in *Chapter 2*. Computational methods to predict the 3D structures of protein-protein complexes and some existing bioinformatics resources that collect and classify 3D structural PPI data are also introduced.

Chapter 3 presents a new integrated database resource called KBDOCK, which was developed during this thesis. KBDOCK integrates 3D coordinate data from the PDB, protein domain family information from Pfam, and DDIs from 3DID. The novelty of KBDOCK is that it provides a spatial view of DDIs for every Pfam protein domain family. *Chapter 4* begins by presenting a more detailed description of existing classifications of protein-protein interfaces. The new concept of “domain

family binding site" (DFBS), introduced in this thesis is described. Next, an analysis of the distribution of DFBSs across protein domain families is presented. This is followed by a discussion of the advantages that the DFBS concept provides.

In *Chapter 5*, a summary of results from previous protein-protein interface analyses is given. Next, a structural classification of DFBSs similar to the CATH classification is described. Some statistical analyses of DFBSs carried out to identify frequent secondary structure features at interfaces are described. *Chapter 6* presents a case-based reasoning (CBR) approach to retrieve distinct structural templates and to model unknown protein-protein complexes. The utility of this CBR approach is demonstrated using the Protein Docking Benchmark dataset. Finally, *Chapter 7* summarises the contributions of this thesis, and it presents some future possible developments and some scientific prospects. Appendix A describes the KBDOCK web server, which provides public access to the KBDOCK resource and its case-based docking functionality. Appendix C and Appendix B contains copies of one published article and three manuscripts in preparation, respectively.

Chapter 2

Biological Context – Modelling 3D Protein-Protein Interactions

Contents

2.1 Protein Molecules and their 3D Structures

2.1.1 Why Study Proteins?

Proteins are one of the major groups of macromolecules essential to all living organisms. Proteins perform many biological functions and they participate in virtually all processes within biological cells. For example, proteins participate in cell signaling, molecular transportation, and cellular regulation, and they also act as structural elements and components of the immune system.

2.1.2 Building Blocks and Architecture of Proteins

Amino Acids are the Building Block of Proteins. Proteins are made up of polypeptides. A polypeptide is formed when amino acids covalently join to each other in a sequential manner releasing water molecules (Figure 2.1). A polypeptide is thus composed of a chain of amino acid residues (or simply "residues"). This sequence defines the "primary" structure of a protein. All of the 20 common amino acids have a central carbon atom (C_α) to which are attached a hydrogen atom, an amino group, a carboxy group and a side chain. What distinguishes one amino acid from another is the side chain (often known as the R group) attached to the C_α (Figure 2.1). This side-chain varies from a single hydrogen atom in glycine to a large aromatic group of atoms in tryptophan. A typical protein contains 200–300 amino acids, but some are much smaller and some are much bigger.[2]

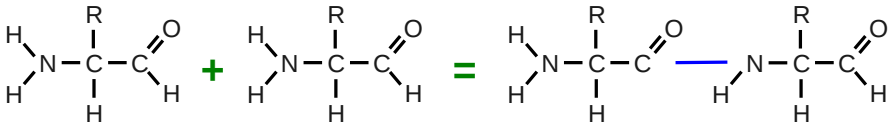


Figure 2.1: An amino acid contains an amino group, a carboxy group, a side-chain group known as the R group, and a hydrogen atom. Amino acids are joined together by peptide bonds (shown in blue).

Levels of Protein Architecture. When a protein is being made by the ribosome, its polypeptide chain is linear and non-functional. To become functional, the polypeptide has to fold and coil (Figure 2.2) into some unique stable 3D structure (often called its "tertiary" structure).[3] This occurs through intermediate forms, when regular segments of the polypeptide fold locally into stable 3D structures (secondary structures) called α-helices and β-strands. Regions with no specific secondary structure are called loop or irregular regions. Some examples of secondary structures are illustrated in Figure 2.3. Many proteins are formed by the association of more than one folded polypeptide chain. The resulting structure is often called the "quaternary" structure of a protein. According to Anfinsen's dogma (Anfinsen *et al.*, 1961, Kresge *et al.*, 2006), the primary sequence of a protein determines its tertiary structure. More generally, the central dogma of molecular biology states that the sequence of amino acids is determined by the sequence of nucleotides in the gene encoding it. Thus, a protein's amino acid sequence determines its 3D structure which in turn determines its biological function.

[2] A good introduction to proteins and their structures can be found in Branden and Tooze (1999).
[3] An introduction to protein folding can be found at http://www.nature.com/horizon/proteinfolding.

Because protein molecules can contain many thousands of atoms which are too many to visualise simultaneously, the 3D structures of proteins are often drawn using e.g. simplified "cartoon" or "ribbon" representations to illustrate stable secondary structures. There also exist other graphical representations, e.g. atoms only, atoms with connecting bonds, and the molecular surface of the protein (Figure 2.4).

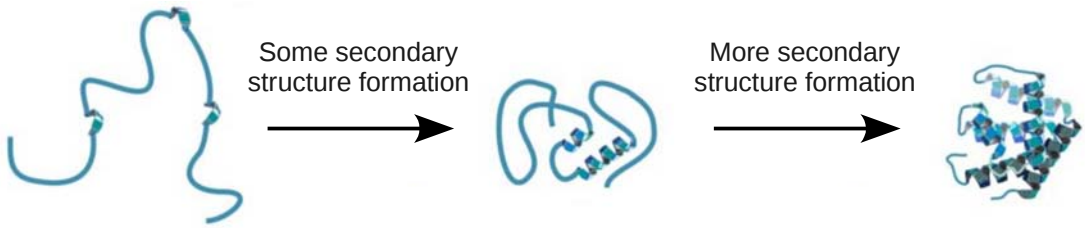


Figure 2.2: An illustration of the formation of secondary structures and protein folding. Figure adapted from Fedyukina and Cavagnero (2011).

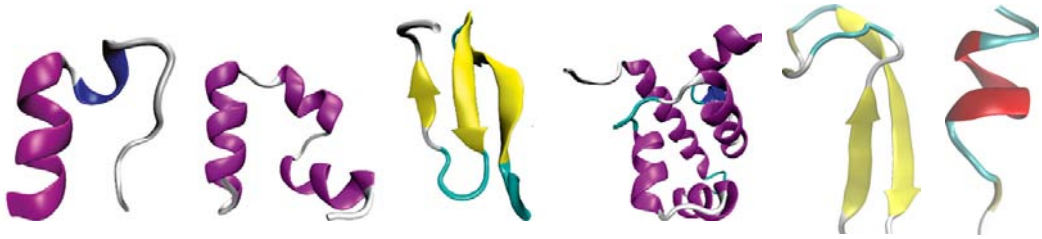


Figure 2.3: Examples of secondary structures: α-helix (purple), β- sheet (yellow), turn (cyan), irregular/coil (white), 3-10 helix (blue), pi-helix (red), and beta-bridge (olive green). The first four fragments were reproduced from Freddolino *et al.* (2010). The last two fragments are extracted from the PDB structures 1abb and 1ba7.

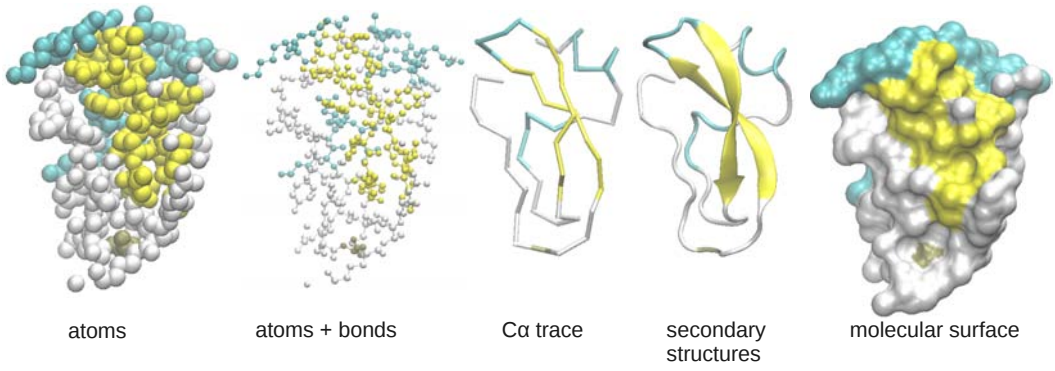


Figure 2.4: Different graphical representations of a protein molecule. Here, a trypsin inhibitor (PDB 1brb chain I) was used for illustration. Figure produced using the VMD visualisation program (Humphrey *et al.*, 1996).

Databases of Protein Sequences. Databases of protein sequences are valuable resources for the study of protein biological function. The Universal Protein Resource (UniProt; Apweiler *et al.*, 2004b) is the main publicly available protein sequence resource. UniProt is a multi-database resource. For example, UniProt's Swiss-Prot contains non-redundant high-quality annotated protein sequences, whereas UniProt's TrEMBL contains redundant automatically annotated sequences. The annotation includes the function of the protein, the processes the protein is involved in, the type of cell the protein is located in, and the post-translational modifications (review by Apweiler *et al.*, 2004a). These annotations are often described using a controlled vocabulary of terms called the Gene Ontology (Ashburner *et al.*, 2000) to allow consistent descriptions of proteins and thus to facilitate database queries.

2.1.3 Protein Domains and their Classifications

What is a Protein Domain? Proteins are often composed of one or more structural subunits called domains. A domain is a compact region of protein structure that is generally made up of a continuous segment of amino-acids, and is often capable of folding sufficiently stably to exist on its own. For example, Figure 2.5 shows the 3D structure of a osteonectin protein which consists of three domains, namely *FOLN* (PF09289; blue), *Kazal_1* (PF00050; red) and *SPARC_Ca_bdg* (PF10591; green). Domains vary in size, but most are around 200 amino acids or less. On average, a protein is folded into approximately two domains (Sali *et al.*, 2003). In the evolution of proteins, different combinations of domains give rise to the diverse range of proteins found in nature. In structural classifications of proteins and domains, a "domain family" is a group of domains sharing similar structural folds, whereas a "protein family" is a group of single-domain proteins or multi-domain proteins. (Copley *et al.*, 2002). In this thesis, the term "protein domain family" is used to refer to either a protein family or a domain family. A domain is often associated with a given function. Therefore, identifying domains within proteins may provide insights into their function (Copley *et al.*, 2002 and references therein). For these reasons, protein sequence databases often classify and organize their sequences into protein domain families and superfamilies.

Sequence-Based Domain Definitions. There exist computational methods to identify domains in proteins. Because sequence data is more abundant than structural data and since protein sequence determines protein structure, most domain definitions are based on the identification of conserved sequences (Copley *et al.*, 2002). Examples of sequence-based domain definitions include Pfam (Finn *et al.*, 2010) and SMART (Letunic *et al.*, 2009). These approaches usually involve collecting and aligning similar sequences automatically, manually editing the alignment to improve quality, and performing an iterative search to identify other related sequences using a hidden Markov model (HMM) sequence profile. For example, the current version of Pfam contains 13,672 protein domain families. Figure 2.5 illustrates the Pfam domain assignments for a given sequence.

Structure-Based Domain Definitions. The two widely used structure-based domain classifications are SCOP (Murzin *et al.*, 1995) and CATH (Orengo *et al.*, 1997). Both SCOP and CATH have four-level hierarchical classifications. The four levels are: class (secondary structure con-

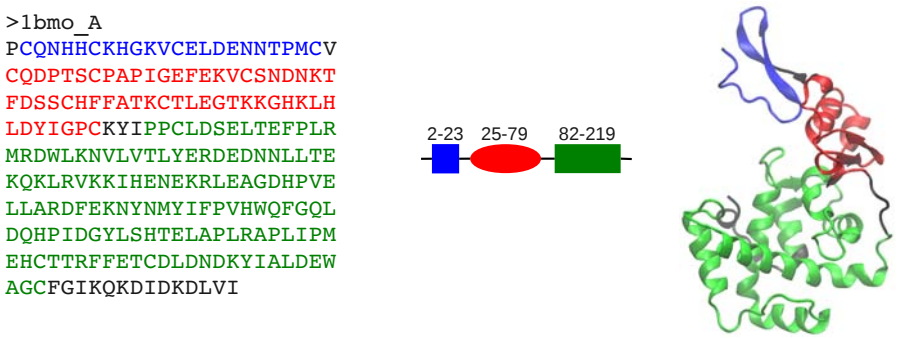


Figure 2.5: Pfam domain assignments for a given query sequence (PDB 1bmo chain A). Pfam identifies three domains, namely, *FOLN* (PF09289; blue), *Kazal_1* (PF00050; red) and *SPARC_Ca_bdg* (PF10591; green). The positions of domains in a sequence are often represented using a "sequence bar" schematic illustration where different shapes represent different domains. On the right, the 3D structure of PDB 1bmo (chain A) is shown. The three domains have been drawn with different colors.

tent), fold/architecture (arrangement of secondary structures), superfamilies/topology (connectivity between secondary structures), and families/homology (sequence, structure and function similarity). Figure 2.6 illustrates the CATH top level classification. Since it is well known that protein folds are often more evolutionary conserved than their sequences (Chothia and Lesk, 1986), structure-based classifications are able to identify evolutionary relationships not detected by sequence analysis and hence may provide better insights into function. For these reasons, several groups have calculated structure-based sequence alignments of SCOP or CATH domain families. PALI (Gowri *et al.*, 2003) and DALI (Holm and Rosenstrom, 2010) are two examples of databases of structure-based sequence alignments.

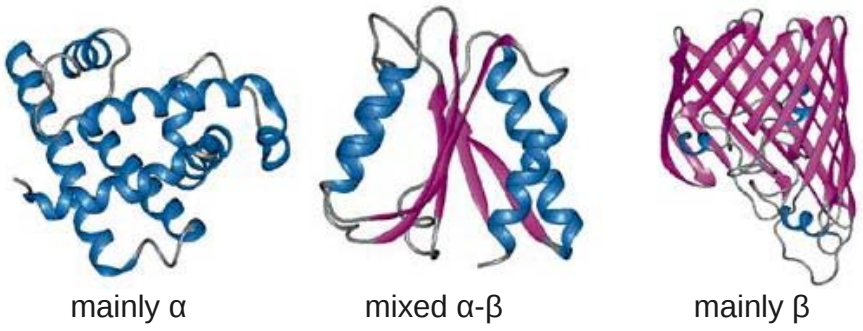


Figure 2.6: Illustration of the top level of CATH classification. There are four classes in total: (1) mainly α, (2) mainly β, (3) mixed α-β, and (4) few secondary structures (not shown here). Figure reproduced from Orengo *et al.* (1997).

Integrated Resource of Domain Classifications. Given the growing number of computational methods to identify domains in protein sequences and structures, some integrated resources have been developed to provide a unified framework for domain analysis. These include CDD (Marchler-Bauer *et al.*, 2009) and InterPro (Hunter *et al.*, 2012). CDD combines domain definitions from

several sources such as Pfam (Finn *et al.*, 2010), SMART (Letunic *et al.*, 2009), and TIGRFAM (Selengut *et al.*, 2007) with 3D structure information from the PDB to define domain boundaries and to guide multiple sequence alignments. InterPro is currently the largest integrated database of domain definitions and functional annotations. It includes Pfam, SMART, TIGRFAM, ProDom (Bru *et al.*, 2005), PIRSF (Nikolskaya *et al.*, 2006), HAMAP (Lima *et al.*, 2009), SUPERFAMILY (de Lima Morais *et al.*, 2011), CATH-Gene3D (Lees *et al.*, 2010), PANTHER (Mi *et al.*, 2010), and PROSITE (Sigrist *et al.*, 2010).

2.1.4 Coverage of Protein 3D Structures

Why are Protein 3D Structures Important? Since the biological function of proteins are determined by their 3D structures, it is essential to know their 3D structures to understand how they function at the molecular level. A 3D structure provides details about atomic contacts which may be useful in designing drugs to disrupt an interaction. Currently, the principal experimental techniques used to obtain 3D structures are crystallography (X-ray), nuclear magnetic resonance (NMR) and cryo-electron microscopy (cryo-EM).

3D Structure Repository and Coverage. The Protein Data Bank (PDB; Berman *et al.* (2002); http://www.rcsb.org/pdb/) is the main worldwide repository of 3D protein structures. Currently, the PDB contains some 80,000 protein structures. X-ray and NMR techniques account for over 99% (88% X-ray and 11% NMR) of the 3D structures deposited in the PDB. Compared to UniRef100, which has some 18,000,000 distinct sequences, the PDB has only 47,000 distinct sequences (as of September 2012).[4] This means that there are many proteins for which there are no known 3D structures. Furthermore, due to limitations in current experimental techniques, such as the difficulty in obtaining protein crystals, it is unlikely that all proteins will have their 3D structures solved in a foreseeable future. For this reason, important efforts has been made to develop computational approaches to predict the 3D structures of proteins from their amino acid sequence (Section 2.1.5; reviews by Wallner and Elofsson, 2005 and Zhang, 2008).

Conserved Protein Folds. From the principle of homology, evolutionarily related (homologous) protein sequences are generally assumed to share a similar 3D structure. One of the earliest studies of protein structures estimated that the large majority of proteins belong to about one thousand fold families (Chothia, 1992), suggesting that protein folds are often more evolutionarily conserved than their sequences (Chothia and Lesk, 1986). The current versions of the main protein structural classifications SCOP and CATH report 1,195 and 1,282 protein folds, respectively. Although Chothia's estimate has stood 20 years, it is difficult to say if nature is indeed restricted to these one thousand or so fold families. For example, other estimates range up to a few thousands (Govindarajan *et al.*, 1999). However, statistics from the PDB (Figure 2.7) show that there has been no significant growth in the number of distinct folds for both SCOP and CATH during the last five years.[5]

[4] Statistics obtained from the RCSB PDB website. The number of distinct sequences is calculated using blastclust (http://blast.ncbi.nlm.nih.gov/) with a sequence identity level of 100%. The number of sequences in UniRef100 was obtained from http://mrs.cmbi.ru.nl/mrs-5/status

[5] http://www.rcsb.org/pdb/static.do?p=general_information/pdb_statistics/index.html

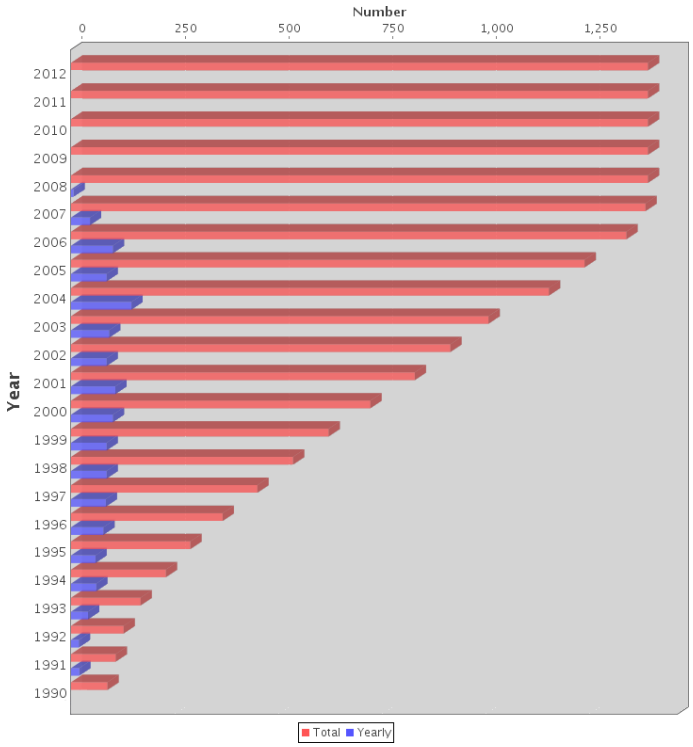


Figure 2.7: The growth of distinct SCOP folds per year since 1976. From this figure, it is clear that the number of folds has not increase for the last four years. Figure reproduced from the RCSB website (September 25th, 2012).

2.1.5 Computational Methods to Predict 3D Structures of Proteins

Most protein structure prediction algorithms use the principles of sequence and structural homology, and thus are referred to as homology modelling or comparative modelling algorithms. Given a target protein sequence, comparative modelling identifies homologous structures or templates and constructs a target-template multiple sequence alignment. Next, the protein backbone is built from the aligned region and side-chain coordinates are added using conformational sampling. MODELLER (Sali and Blundell, 1993) is a widely used protein structure prediction program. Previous studies have shown that fairly good 3D models of single proteins can be obtained if there already exists a protein structure with sequence similarity of 25% or more (Aloy *et al.*, 2005).

On the other hand, when there is no direct homology but only templates with low sequence identity, fragment-based techniques may be used as an alternative way to model protein structures. Similar to comparative modelling, fragment-based modelling also consists of identifying homologous or analogous[6] templates and aligning them with the target protein sequence. Typically, a library of fragments of protein structures is constructed from the selected templates or from a non-redundant set of protein structures. Finally, a protein model is obtained by reassembling the fragments with some form of physics-based minimisation such as Monte Carlo simulation. Rosetta (Simons *et al.*, 1997) and TASSER (Zhang and Skolnick, 2004, 2005*a*) are examples of fragment-based algorithms.

[6]Two proteins are analogous when they share some sequence or fold similarities even though they are not evolutionary related.

Zhang and Skolnick suggested that the PDB will often contain suitable structural templates to model the structures of new protein sequences. However, it still remains a difficulty to retrieve such templates using current sequence or structure alignment methods. The dataset of Zhang and Skolnick consisted of 1,489 protein sequences sharing less than 35% sequence identity. For nearly all of these sequences, they found a structural template with a similar fold to the native structure with an average RMSD of 2.5 Å and a sequence alignment coverage of 82%. Overall, their TASSER algorithm built models with an RMSD less than 6 Å from the native structure, and in 97% of the cases with an RMSD less than 4 Å.

The Critical Assessment of protein Structure Prediction (CASP; http://predictioncenter.org) experiment establishes the current state of the art in single protein structure prediction from protein sequence. CASP results show that the best predictions are often made when there exists some sequence homology (Moult *et al.*, 2011). Some recent databases which collect manually-built and computer-generated 3D models of proteins include PMDB (Castrignano *et al.*, 2006), MMDB (Wang *et al.*, 2007*b*), and MODBASE (Pieper *et al.*, 2009).

2.2 Protein-Protein Interactions and their 3D Structures

2.2.1 Why Study Protein-Protein Interactions?

To perform their biological functions, proteins interact with other proteins to form protein assemblies. According to Alberts (1998), almost all biological processes in a cell are carried out by assemblies of 10 or more proteins. Furthermore, Alberts suggested that the entire cell may be viewed as a large, elaborate, and dynamical molecular network. Understanding the full network of protein-protein interactions, the so-called “interactome”, could provide useful insights into the mechanisms of disease. As discussed in Chapter 1, knowing *which* proteins interact and *how* they interact at the structural level is crucial for understanding the molecular basis of biological processes. The 3D structures of known protein-protein complexes provide crucial atomic details about binding which can be useful to understand the effect of genetic variations, and to design therapeutic drugs.

2.2.2 Databases of Experimentally-Detected and Predicted PPIs

Recent advances in molecular biology techniques have allowed biologists to detect protein-protein interactions (PPIs) on a large scale. For example, the development of high throughput technologies (HTT) such as yeast two-hybrid (Y2H), tandem affinity purification coupled to mass spectrometry (TAP-MS), and protein-fragment complementation assays (PCA) has increased substantially the number of recorded protein-protein interactions. HTT PPI data sets are now available for more than 30 organisms in public databases such as BioGRID (Stark *et al.*, 2006), DIP (Salwinski *et al.*, 2004), IntAct (Hermjakob *et al.*, 2004), and MINT (Chatr-Aryamontri *et al.*, 2007). Other organism-specific PPI databases also exist such as, MPact (Guldener *et al.*, 2006) for yeast and HPRD (Peri *et al.*, 2003) for human. The PPIs detected by HTT experiments are often represented as an interaction network. Several tools for exploring and analysing PPI networks have been developed. One well-known example is Cytoscape (http://www.cytoscape.org/).

Due its small size, the yeast interactome is the most complete one to date among eukaryotes. On the other hand, the human interactome which has been estimated to involve about 130,000 PPIs (Venkatesan *et al.*, 2009) currently has 66,000 recorded PPIs. This relative shortage of PPI data has led to the development of computational methods to predict protein interaction partners. These methods rely on genomic and sequence information such as phylogenetic profiles, conservation of gene neighborhood, gene fusion, and correlated mutations (review by Valencia and Pazos, 2002). Recent databases of predicted PPIs include OPHID (Brown and Jurisica, 2005), DOMINE (Raghavachari *et al.*, 2008), PIPs (McDowall *et al.*, 2009), STRING (Jensen *et al.*, 2009), and HAPPI (Chen *et al.*, 2009). However, Yu *et al.* (2010) observed that due to the nature of the PPI data used to train machine learning algorithms, the accuracy reported by such algorithms is often significantly over-estimated. As well as the shortage of PPI data, it is also a fact that HTT techniques often produce many false-positive interactions and miss many interactions. For example, it has been estimated that there are 30-60% false positives and 40-80% false negatives in yeast two-hybrid and affinity-purification PPI data (von Mering *et al.*., 2002, Aloy and Russell, 2002, 2006). Hence, developing computational methods which can distinguish true and false PPIs coming from HTT experiments will be useful. A review covering experimental techniques to detect PPIs, PPI databases, and computational methods to predict PPIs can be found in Shoemaker and Panchenko (2007*a*,*b*).

2.2.3 Different Types of Protein-Protein Interactions

In the analysis of PPIs, it is important to distinguish between different types of interaction. Early work classified PPIs according to the structural similarity of their constituent proteins and the thermodynamics and kinetics (i.e. duration) of their associations (Nooren and Thornton, 2003). For example, interactions between identical proteins may be termed as “homo” PPIs. On the other hand, when dissimilar proteins interact, their interactions are called “hetero”. Interactions between proteins which exist only in complexed form are known as “obligate” PPIs. In contrast, interactions between proteins that can exist independently are called “non-obligate” PPIs. Short-lived PPIs are known as “transient” PPIs, while complexes that do not dissociate during their lifetime are known as “permanent” PPIs. Obligate interactions are usually permanent, whereas non-obligate interactions may be transient or permanent (Nooren and Thornton, 2003). Homo complexes are usually permanent (Jones and Thornton, 1996). PPIs are also classified depending on the location of the constituent proteins within one (“intra”) or on two (“inter”) polypeptide chains. In this thesis, we aim to describe hetero PPIs because these are often the most difficult structures to solve experimentally (Ezkurdia *et al.*, 2009, Jones and Thornton, 1996).

2.2.4 Coverage of 3D Protein-Protein Interactions

Due to the complex physical nature of protein-protein interactions, structural genomics initiatives have been striving to solve 3D structures for every PPI in the interactome. Aloy and Russell (2002) estimated that there is a total of 10,000 representative protein-protein interactions, and they proposed that most interactions in nature will conform to one of these interactions. According to them, about 2,000 of the 10,000 representative interactions are known presently (year 2002). This is due to current limitations in experimental techniques. For example, for short-lived transitory associa-

tions, it is very difficult to obtain protein crystals. Moreover, although NMR can be used quite easily to obtain 3D structures of proteins and protein-protein complexes consisting of 300 amino acids, it is painstaking and costly to solve larger complexes. For these reasons, only a very small proportion of the 3D structures deposited in the PDB correspond to protein-protein complexes. For example, it has been estimated that the PDB contains structures of protein complexes for only 8% of the human PPIs (Figure 2.8; Stein *et al.*, 2011, Kundrotas *et al.*,2012). It seems unlikely that it will become possible to solve the structures of protein complexes using current structural genomics techniques in the forseeable future (Aloy and Russell, 2002, 2006, Ritchie, 2008). Hence, there is a need to develop computational methods to bridge the gap (Aloy and Russell, 2006, Stein *et al.*, 2011).

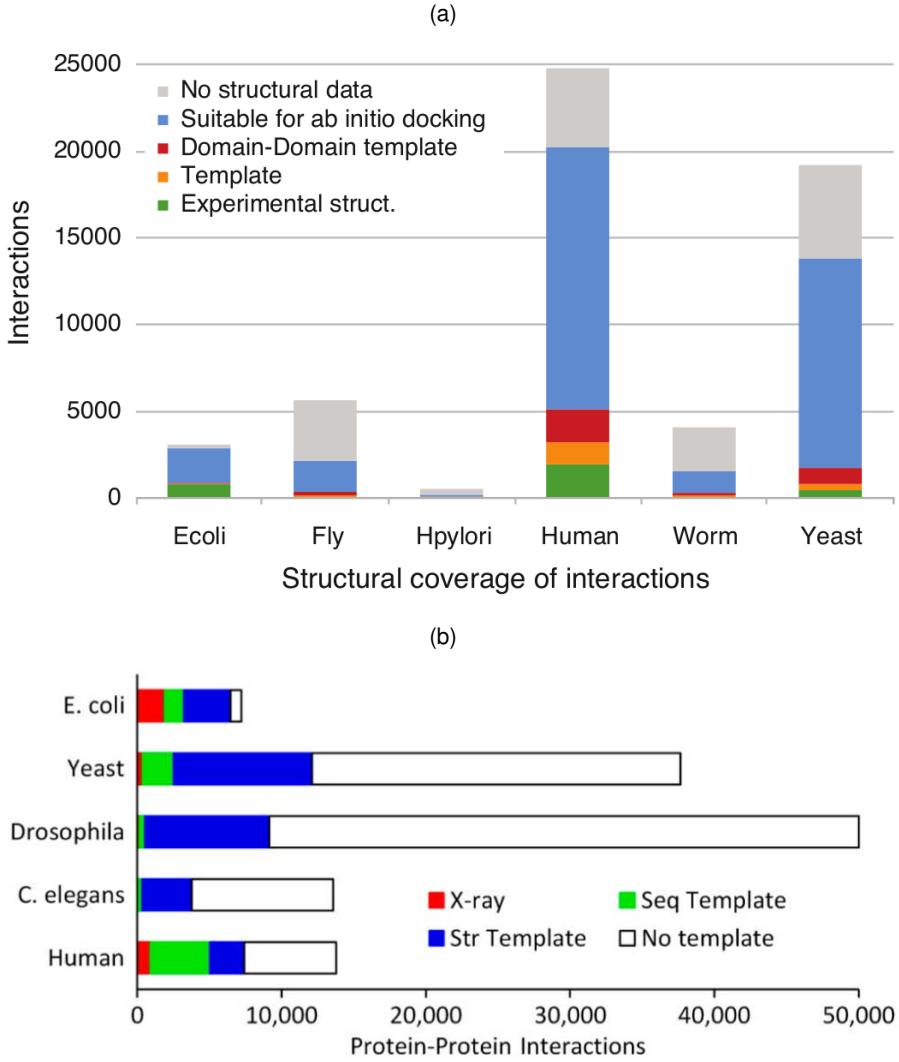


Figure 2.8: The coverage of 3D structures of all experimentally-solved PPIs for different organisms. Both charts show that the proportion of PPIs with a experimentally-solved 3D structure to those without one is highly disproportionate except for *E. coli*. Figures (a) and (b) were reproduced from Stein *et al.* (2011) and Kundrotas *et al.* (2012), respectively.

2.2.5 Previous Analyses of Protein-Protein Complexes

The collection of experimental 3D structures of protein-protein complexes has allowed several groups to study the biophysical properties of protein binding sites and interfaces. A protein-protein interface is the region where two proteins make direct physical contact. In this thesis, we refer to a protein-protein interface as a protein interface or simply interface. Similarly for a domain-domain interface. A binding site is one half of an interface and is often referred to as a face by many groups. See Figure 2.9 for an illustration of a binding site and an interface.

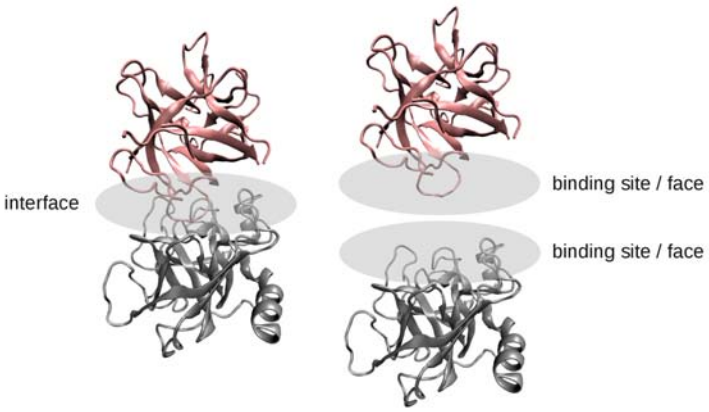


Figure 2.9: An interface is defined as the contacting region between two proteins. An interface is formed by a pair of binding sites or faces. Here, the PDB 1avw is used for illustration.

Since the size of a protein interface provides a measure of the strength of the binding (Jones and Thornton, 1996), this simple property is often calculated and analysed. Interface size is measured as the accessible surface on both proteins that becomes inaccessible to solvent in the complex. This area is calculated as the sum of the solvent-accessible surface areas (ASA) of the isolated proteins less that of the protein-protein complex (Lo Conte *et al.*, 1999).

In one of the earliest structural studies of 19 protein-protein complexes, Janin and Chothia (1990) observed that the interfaces of protease-inhibitor complexes and antigen-antibody proteins often have similar properties, e.g. interface sizes and number of hydrogen bonds. On the other hand, Jones and Thornton (1996) found that many interfaces have roughly equal proportions of helix, sheet, and loop residues, with some interfaces containing only one type of secondary structure, but most being mixed. Other studies based on alanine scanning mutagenesis have shown that the stability of a complex is determined by so-called hot spot residues at the interface. These hot spot interface residues can contribute a large proportion to the total binding energy (Bogan and Thorn, 1998). Bogan and Thorn found that large amino acids such as tryptophan, arginine and tyrosine are frequent hot spot residues. Thorn and Bogan (2001) collected all experimentally determined hot-spot residues by alanine scanning and made them publicly available in the ASEdb database. Ma *et al.* (2003) found that hot spot residues are more structurally conserved in PPI interfaces than in other surface regions, and they proposed that this tendency might be used to help predict the locations of unknown binding sites. Caffrey *et al.* (2004) found that the residues at protein interfaces are usually more conserved than other surface residues, particularly in enzyme/inhibitor complexes, but they also found that such differences are not sufficient to predict interface patches by conservation alone.

2.2.6 Current Protein-Protein Interface Prediction Algorithms

Identifying protein binding sites and hot-spot residues is important for drug design. Hence, many computational methods have been developed for predicting protein binding sites. The findings discussed in the previous paragraph have been useful in determining which binding site features can be used as predictive attributes in machine learning and other techniques. However, the general opinion is that no single parameter can distinguish between binding surface and other surface patches (Zhou and Qin, 2007). For example, prediction algorithms use a number of sequence and structural properties, e.g. sequence conservation, proportions of amino acids, secondary structure, solvent accessibility to train their machine learning algorithms. Such algorithms include support vector machines (Bradford and Westhead, 2005), neural networks (Ofran and Rost, 2007*a*), conditional random fields (Li *et al.*, 2006), random forests (Sikic *et al.*, 2009) and naïves Bayes (Murakami and Mizuguchi, 2010). Other groups have developed statistical methods to predict protein binding sites, e.g. the evolutionary trace method of Lichtarge *et al.* (1996), the surface patch analysis of Jones and Thornton (1997), the ProMate algorithm of Neuvirth *et al.* (2004), and the WHISCY algorithm of de Vries *et al.* (2006). In addition, other groups have developed so-called meta-predictors, e.g. Qin and Zhou (2007*a*), which combine multiple binding site prediction algorithms. de Vries and Bonvin (2008) reviewed the performance of several predictors and they concluded that algorithms which incorporate 3D features as well as sequence-based features performs better. Several reviews are available on binding sites prediction algorithms (de Vries and Bonvin, 2008, Ezkurdia *et al.*, 2009, Fernandez-Recio, 2011).

2.3 Modelling 3D Structures of Protein-Protein Complexes

Modelling 3D interactions is the computational task of calculating the 3D structure of a protein-protein complex starting from either the individual sequences (homology-based modelling) or the individual structures of the constituent proteins (*ab-initio* docking). This is different to other computational methods focusing on predicting only the binding regions of the constituent proteins (see Section 2.2.6) and not the 3D interaction mode.

2.3.1 Template-Based Modelling of Protein Complexes

It has been shown that pairs of proteins with greater than 25% sequence similarity often interact in similar ways (Aloy *et al.*, 2005), and that the active sites of distantly related proteins are often very similar in geometry (see Chothia (1992) and references therein). Template-based complex modelling techniques are based on these general observations. Most algorithms start with the individual sequences of the constituent proteins. Similar to template-based modelling of single proteins, the crucial step is to identify the best available structural protein-protein complex template. If there exists direct homology, then the process is simple and straightforward. The constituent proteins are modelled separately (Section 2.1.5) and a structural alignment is performed to superpose the modelled proteins onto the structural template to obtain a model of the complex (Kundrotas and Alexov, 2006, Kundrotas *et al.*, 2008, 2010). On the other hand, if no direct homology exists,

fragment-based techniques are used to model the protein-protein complex from a library of single protein structures and complex structures (e.g. see Mukherjee and Zhang, 2011). Some algorithms perform an information-driven sequence alignment to obtain the optimal target-template alignment. For example, sequence alignments are guided by other information such as interface residues in protein-protein complex structural templates (Lu *et al.*, 2002, Kundrotas *et al.*, 2008) and residue solvent accessibility (Launay and Simonson, 2008). On the other hand, some approaches start from the 3D structures of the individual proteins. For example, the approach of Günther *et al.* (2007) is a local structure alignment based method which exploits in some way Chothia's observation that distantly-related proteins often have similar binding site geometries.

Kundrotas's Homology-Based Approaches. Kundrotas and Alexov (2006) performed a "cross-docking" experiment to identify interacting pairs of proteins and to build by homology the structure of the corresponding hetero complexes. For this study, they filtered their ProtCom database (Kundrotas and Alexov, 2007) using a sequence similarity of 40% and excluding complexes with small and very large interfaces. In this way, their database contains 92 hetero protein-protein complexes and a further 326 hetero domain-domain interactions which were derived from intra-chain DDIs giving a total of 418 pairs of true interacting proteins/domains. By enumerating all the possible pairs of proteins which can be formed by these 418 pairs of proteins and domains, they obtained 350,284 putative pairs of proteins. They experimented with two different approaches - sequence and structural similarity to retrieve homologous templates from their ProtCom database in order to identify true interacting pairs of proteins. They used the sequence search tool PSI-BLAST and the structural alignment program SKA (Yang and Honig, 2000) program to search for templates and they used the NEST (Petrey *et al.*, 2003) program to model a 3D structure of a protein/domain. They considered a pair of query sequences to interact if there is at least one database template involving both query sequences. In the same way, they considered a pair of query structures to interact if there is at least one database template involving a pair of structures similar to the query pair. Using this approach, Kundrotas *et al.* found that their sequence and structure similarity approach correctly identified 19% and 86 % of 418 true interacting pairs, respectively (out of a total of 350,284 pairs). However, the ratio of false to true positive was 5:1 and 7:1 for their sequence and structure-based approaches, respectively. Hence, although the structure-based approach has a better recovery rate, their sequence-based approach has a better performance. To model the structure of the complex, (i) for the sequence-based approach, the two structures are modelled individually using the template complex and are put together to give a model of the complex, and (ii) for the structure-based approach, the two structures are superposed onto the templates to build the model of the complex. Kundrotas *et al.* evaluated the model of the complex using the number of correct interface residues and the RMSD of heavy atoms between the model and the solution. They found that intra DDI templates can be useful in predicting structures of protein-protein complexes.

Kundrotas *et al.* (2008) extended their PSI-BLAST-based template retrieval approach to incorporate interface information in the PSI-BLAST sequence profiles. They found that 74 out of 463 pairs of query protein sequences produce statistically significant pairs of global alignments. In other words, 16% of pairs of query sequences have a template. Significant alignments are those with $>$20% sequence identity and coverage of $\geq$40%. This represents an increase in the number of

templates retrieved compared to their previous study, although this improvement may be due to the new 3D structures which had since been added to their database. However, it seems there is no improvement if any in the quality of the models produced from using an augmented PSI-BLAST sequence profile. In a more recent study, Kundrotas *et al.* (2012) uses the structural alignment program TM-align (Zhang and Skolnick, 2005*b*) to retrieve structural templates. They found that a structure-based approach is able to identify templates for more PPIs than a sequence-based approach (see Figure 2.8 in Section 2.2.4).

The ISEARCH Approach. Günther *et al.* (2007) exploit the fact that non-homologous pairs of proteins may have similar interfaces to predict the 3D structures of protein-protein complexes using a structural superposition method. Their method called ISEARCH uses a domain-domain interface library which consists of the structures of DDIs defined by the SCOP domain classification. Pairs of interacting SCOP domains are grouped into distinct SCOP superfamily-superfamily clusters. For each such cluster, a single representative pair of interacting domain-domain is considered in order to reduce computational cost. Given a pair of query structures, Günther *et al.* use their local structural alignment program called NeedleHaystack (Hoppe and Frömmel, 2003) to search for pairs of backbone regions which are similar to the domain-domain interface backbone patches in their library. For each hit obtained, the query structures are transformed according to the superposition onto the corresponding interface patches to build the model of the complex. Overall, the ISEARCH approach found at least one acceptable model for 45 of the 59 benchmark cases. As expected, if the query and template domain pairs belong to the same superfamily pair then an acceptable model is obtained (20 cases). Interestingly, in 35 cases, at least one acceptable model was obtained using templates from a different pair of superfamily to the query. These results confirmed previous suggestions that protein complexes may be modelled from remote homologous pairs of interacting proteins. However, this local structural similarity-based approach often gives too many templates. For example, for a given pair of structures, ISEARCH finds about 1500 templates on average.

2.3.2 Ab-Initio Docking

Ab-initio docking aims to calculate the 3D structure of a protein complex starting from the 3D structures of the unbound components. This problem was first described some thirty years ago (Wodak and Janin, 1978), and since then many computational docking algorithms have been developed. A typical protein docking algorithm involves two main stages (Figure 2.10). The first stage is a global search to generate a list of configurations having good shape complementarity using a simple scoring function. This list could contain up to a few thousands configurations and it often contains a near-native configuration. To search and generate rapidly candidate configurations, most current docking algorithms use the fast Fourier transform correlation (FFT) technique. For example, ZDOCK (Chen and Weng, 2002), GRAMM (Vakser and Aflalo, 1994), and FTDock (Gabb *et al.*, 1997). Other global search techniques include spherical polar Fourier correlations (Hex; Ritchie and Kemp, 2000), geometric hashing (PatchDock; Schneidman-Duhovny *et al.*,2005), and Monte Carlo sampling (RosettaDock; Wang *et al.*, 2007). The second stage in docking two proteins is to rank the list of configurations using a sharper scoring function comprising physics-based or statistical poten-

tials. Physics-based potentials are derived from electrostatics, hydrogen bonding, and van de Waals atomic interactions. Statistical potentials (also called knowledge-based potentials) are derived from atom-atom contacts from existing protein-protein complexes (see e.g. Moont *et al.*, 1999, Chuang *et al.*, 2008). The ranking step often includes an initial clustering of similar orientations and thus ranking only the representative orientation from each cluster.

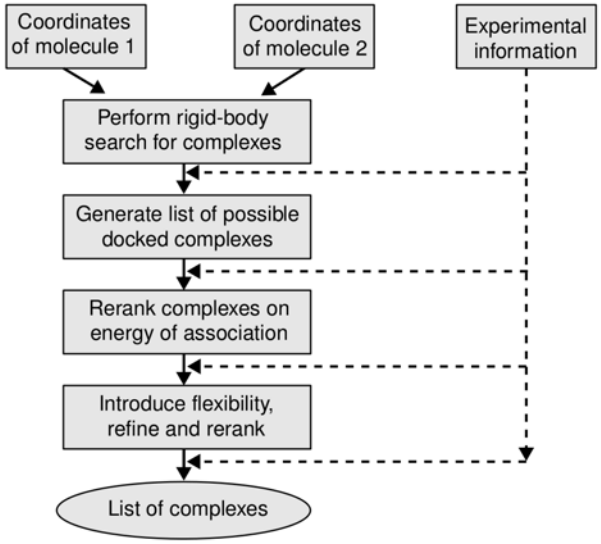


Figure 2.10: The stages of protein-protein docking. Figure reproduced from Smith and Sternberg (2002).

Modelling protein-protein complexes is a challenging task because most proteins are not rigid. They change their shape to interact with other molecules to perform their biological function. However, many protein-protein complexes were found to involve little or no conformation change in their backbone (Król *et al.*, 2009). The main conformational changes were observed in the side-chains of the proteins (Król *et al.*, 2009). Some docking algorithms include a third stage to simulate the backbone and side-chain rearrangements and dynamics. The HADDOCK program (Dominguez *et al.*, 2003) is an example of a docking program which simulates both side-chain and backbone flexibility using molecular dynamics. Movshovitz-Attias *et al.* (2010) recently carried out a detailed study on the utility of using structural templates of monomeric proteins when modelling conformational flexibility in a number of docking problems. They found that in 19 out of 26 target complexes, homologous templates improve the docking predictions. Furthermore, they found that templates with different levels of sequence identity are all useful sources of information for modelling backbone flexibility. Several reviews on protein docking algorithms are available (Halperin *et al.*, 2002, Smith and Sternberg, 2002, Bonvin, 2006, Ritchie, 2008, Vajda and Kozakov, 2009, Król *et al.*, 2009).

Korkin's Restrained Docking Experiment. Previous studies showed that the locations of protein interaction sites are often conserved within domain families (Korkin *et al.*, 2005, Shoemaker *et al.*, 2006). Korkin *et al.* (2006) performed a study to find out whether binding site information can improve the accuracy of docking predictions. They evaluated their method on a set of 20 single and multi-domain protein-protein complexes (which is a relatively small test set compared to the docking benchmark of Hwang *et al.*, 2010). For each SCOP domain in a target complex, they used their

PIBASE database (Davis and Sali, 2005) to extract binding sites from all domains of the same SCOP family and they mapped these binding sites onto the target domains using structural alignments. For a given target domain-domain complex, Korkin *et al.* performed restrained structure-based docking using PatchDock (Schneidman-Duhovny *et al.*, 2004) for all possible pairs of binding sites. They used a shape complementarity scoring function. They collected all the docking predictions from the multiple PatchDock runs and they filtered out predictions that do not involve the proposed pair-wise binding sites. They scored the remaining predictions using the DOPE statistical potential of Shen and Sali (2006) and they retained the top prediction. Korkin *et al.* compared the predictions obtained by restrained docking on (i) both sides, (ii) one side, and (iii) blind docking. They found that by restraining the docking on both sides increases significantly the docking prediction (15 out of 20 complexes have an RMSD less than 10Å). On the other hand, restraining the docking on only one side does not give much improvement from blind docking (i.e. 8 out of 20 with blind docking going to 9 out of 20 with restrained docking).

2.3.3 The CAPRI Blind Docking Experiment

Although the main goal of protein docking is to predict the 3D structure of an unknown protein-protein complex, recent cross-docking studies have used docking to distinguish true interactions from multiple candidate complexes (Sacquin-Mora *et al.*, 2008, Wass *et al.*, 2011, Melquiond *et al.*, 2012). Such "cross-docking" studies show that docking can be used to identify PPIs with some success (Sacquin-Mora *et al.*, 2008). This suggests that docking could be used to confirm PPIs from HTT experiments. However, current docking algorithms still face some difficulties. Similar to CASP, the Critical Assessment of Protein Interactions (CAPRI) establishes the current state of the art in modelling 3D structures of protein-protein complexes (http://capri.ebi.ac.uk). Although good progress has been made, CAPRI results show that it is still very challenging to produce a satisfactory 3D model (2.5Å RMSD) of a protein complex using *ab initio* docking algorithms (Lensink and Wodak, 2010). The main difficulty is to identify the near-native solutions from the list of candidate solutions. On the other hand, several studies have shown that using the principles of homology or experimental information such as mutagenesis data to guide and constrain docking calculations (Figure 2.10) can improve the reliability of the predictions significantly (van Dijk *et al.*, 2005, Lensink and Wodak, 2010). Moreover, Sacquin-Mora *et al.* (2008) showed that prediction of interacting partners can be improved if the correct binding interface on each protein is known *a priori*. Furthermore, Qin and Zhou (2007*b*) showed that constraining the docking search space using their interface predictor cons-PPISP improved the docking rankings in some cases (8 out of 20 CAPRI targets). These results suggest it would be useful to define and characterise systematically protein binding sites in order to guide protein docking calculations and 3D interaction modelling in general.

2.4 Existing Structural PPI Resources

Given the growing amount of structural protein interaction data, several groups have developed bioinformatics resources to integrate, organise, and classify heterogeneous protein interaction data with the aim of facilitating analysis and more importantly encouraging data re-use. Table 2.1 sum-

marises some of the recent PPI databases. Several of these are particularly relevant to this thesis, and are described in further detail in the following chapters.

Database	Literature	Domain	Classification
Pibase http://pibase.janelia.org/queries.html	Davis and Sali (2005)	SCOP, CATH	Hierarchical clustering of secondary structure topologies, contacting residue and secondary structure types.
3D-Complex http://supfam.mrc-lmb.cam.ac.uk/elevy/3dcomplex/Home.cgi	Levy *et al.* (2006)	–	Graph-based classification of topologies of subunits in protein-protein complexes.
Dockground http://dockground.bioinformatics.ku.edu/	Douguet *et al.* (2006)	–	Collection of X-ray structures and docking models of protein complexes with their unbound constituent proteins.
Scoppi http://141.30.193.6/scoppi/	Winter *et al.* (2006)	SCOP	Classification of protein-protein interfaces using 1D sequence and 3D structure features.
Scowlp http://www.scowlp.org/scowlp/	Teyra *et al.* (2008)	SCOP	SCOP-based hierarchical classification of protein-protein complexes.
PiSite http://pisite.hgc.jp/	Higurashi *et al.* (2009)	–	Collection of 3D structures of transient hub proteins and their interaction sites.
3did http://3did.irbbarcelona.org/	Stein *et al.* (2010)	Pfam	HMM sequence profile-based hierarchical classification of protein-protein interfaces.
Gwidd http://gwidd.bioinformatics.ku.edu/	Kundrotas *et al.* (2010)	–	Collection of all experimentally-determined solved PPIs and docking models of PPIs without a known 3D structure. PPI sources include BIND (Bader and Hogue, 2000) and DIP (Salwinski *et al.*, 2004).
Ibis http://www.ncbi.nlm.nih.gov/Structure/ibis/ibis.cgi	Shoemaker *et al.* (2010)	CDD	Database of protein-protein interactions grouped by their CDD domain followed by their partner domain and the location of their binding sites.
InterEvol http://biodev.cea.fr/interevol/Default.aspx	Faure *et al.* (2012)	–	Close and distant structural homologues of protein-protein complexes with multiple sequence and structural alignments.

Table 2.1: A list of recent structural PPI databases.

2.4.1 Classifications of 3D Structures of Protein-Protein Complexes

3D-Complex (Levy *et al.*, 2006) is a structural classification of all X-ray protein complexes in the PDB. The classification is based on sequence identity, structural homology, interface contacts and the symmetry between the biological subunits in a protein complex. 3D-Complex uses the SCOP classification to determine structural homology. It defines an interface when there are at least ten residue contacts between two subunits. Using the calculated sequence identity, structural homology and interact contact information, 3D-Complex constructs a graph to represent the "topology" of a complex (Figure 2.11). To construct a hierarchical classification, 3D-Complex first groups complexes with similar graph topologies followed by structural homology, the number of genes, and finally, sequence identities. Using this approach, 3D-Complex grouped 14,112 non-identical structures into 3,473 families (structural homology) and 191 topologies.

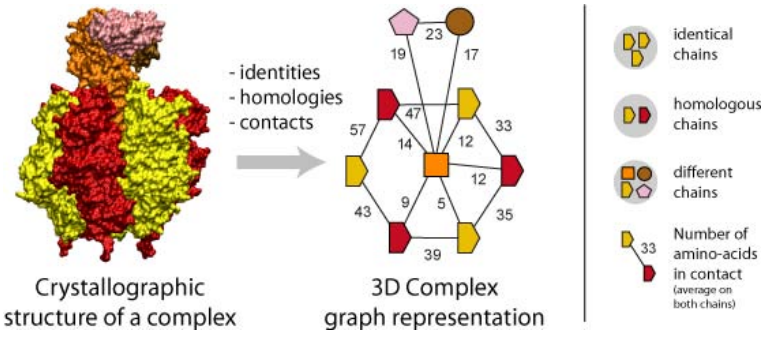


Figure 2.11: Illustration of the graph-based representation of a protein complex in 3D-Complex. Figure reproduced from Levy *et al.* (2006).

2.4.2 Characterisations of Protein Functional Sites

PROSITE (Sigrist *et al.*, 2010) is primarily a database of protein family and domain sequence profiles which are derived from structure-based sequence alignments. The PROSITE developers developed the ProRule database (Sigrist *et al.*, 2005) which contains manually created rules describing in a controlled vocabulary the PROSITE profiles. In addition, PROSITE uses sequence profiles[7] to derive regular expressions for biologically important residues such as enzyme catalytic sites, prosthetic group attachment sites, metal ion binding amino acids, cysteines involved in disulfide bonds, and other regions involved in ligand binding. These regular expressions, known as patterns, are around 10 to 20 amino acids in length. For example, the PROSITE entry PS00286 corresponds to the *Squash* Pfam domain family of serine protease inhibitors. The PROSITE consensus sequence pattern for this family is "C-P-x(5)-C-x(2)-[DN]-x-D-C-x(3)-C-x-C", where an "x" indicates any amino acid. The active site is indicated as a hash sign in the pattern "xxCx#xxxxCxxxxxCxxxCxCxxxxxCx". Compared to sequence profiles, such PROSITE patterns are not tolerant to mismatches. For example, a single mismatch will cause a pattern to fail. Hence, sequence profiles are still a popular way to describe protein families and domains and their functionally important amino acid residues.

[7] A sequence profile is usually derived from a large multiple sequence alignment and it contains for each position in the sequence alignment a probability value for each of the 20 amino acids.

2.4.3 Classifications of Protein-Protein Interfaces

Several groups have collected and classified all available protein-protein interfaces in the PDB. Some recent PPI databases include PIBASE (Davis and Sali, 2005), SCOPPI (Winter *et al.*, 2006), SCOWLP (Teyra *et al.*, 2006), IBIS (Shoemaker *et al.*, 2010) and 3DID (Stein *et al.*, 2010). Because protein domains may often be identified as structural and functional units, the 3D structures of protein complexes are often analysed in terms of their component domain-domain interactions. Hence, structural PPI databases often describe PPIs in terms of DDIs. However, this reductionist way of describing PPIs gets complicated when dealing with multi-domain PPIs. Existing methods may be grouped according to the kind of information used in the classification, e.g. interface residues mapped onto sequence alignments, and/or structural alignments. For example, 3DID identifies and describes DDIs in PDB structures using Pfam and it clusters domain-domain interfaces using a sequence-based hierarchical clustering. 3DID gives an average of about 10 interfaces per Pfam family. On the other hand, SCOPPI defines DDIs using the SCOP classification, and classifies domain-domain interfaces using a sequence and structure-based approach. SCOPPI gives on average about 5.4 distinct interface types per SCOP domain family. In general, sequence-based methods tend to give larger numbers of interface types than structure-based methods.

2.4.4 Structural Databases of Protein-Protein Complexes

GWIDD (Kundrotas *et al.*, 2010) is a database of experimentally-determined 3D structures of protein-protein complexes as well as 3D models obtained by homology modelling using the NEST program (Petrey *et al.*, 2003). For every PPI in the BIND and DIP databases, if there is no corresponding 3D structure in the PDB, GWIDD searches for a homologous PDB complex from which it builds by homology a 3D model of the complex. The GWIDD database organises the experimentally-solved and modelled 3D structures of PPIs according to the organisms they belong to. GWIDD contains 126,897 PPIs involving 43,976 proteins from a total of 771 different organisms. Currently, GWIDD contains 10,924 experimentally-solved structures and 14,635 modelled structures. These numbers show that only 20% of known PPIs have an experimentally-solved or model-built 3D structure (see Figure 2.8).

2.4.5 Integrated Databases, APIs and Libraries

In order to facilitate the reuse of existing bioinformatics resources, several groups have developed integrated databases and programming environments. For example, Biskit (Grunberg *et al.*, 2007) is a modular Python library developed to facilitate the manipulation and analysis of macromolecular structures, protein complexes, and molecular dynamics trajectories. With the aim of automating and parallelizing complex workflows, Biskit gathers several popular programs such as TMAlign for structural alignment, DSSP for secondary structure prediction, MODELLER for structure prediction, Hex for docking, and Amber for molecular dynamics simulations. On the other hand, SNAPPI-DB (Jefferson *et al.*, 2007) is a relational database designed mainly for dealing with structural domain-domain interactions. It integrates structural, sequence, and functional information from PDB, SCOP, CATH, Pfam, InterPro, GO, for example. SNAPPI-DB is a useful integrated database since DDIs are described in terms of three different domain definitions and domain-domain inter-

faces are classified and aligned by their domain family or superfamily pair. In addition, SNAPPI-DB provides a Java API to facilitate database access for other applications. SNAPPI-DB provides a good starting point for many computational methods such as data mining and machine learning.

2.4.6 Docking Benchmark Datasets

Several groups have collected pairs of unbound protein structures and their corresponding complex. These data sets are useful to assess the performance of docking algorithms. The Protein Docking Benchmark (Hwang *et al.*, 2010) and DOCKGROUND (Douguet *et al.*, 2006) are two examples of publicly available data sets. The Protein Docking Benchmark is a non-redundant expert-curated set of 176 protein complexes for which the bound complex structures and most of the unbound component structures have been solved by X-ray crystallography to a resolution of 3.25 Åor better. Hwang *et al.* divided the benchmark into 52 enzyme-inhibitor complexes, 25 antigen-antibody complexes, and 99 "Other" complexes, and they classified each target as "Rigid", "Medium", and "Difficult" according to the degree of conformational changes between the bound and unbound structures. Targets in the Rigid class should be amenable to rigid body docking algorithms, whereas Difficult targets normally require a flexible docking algorithm to be used in conjunction with prior knowledge about the binding mode.

2.5 Conclusion

Although several structural PPI databases have been described recently (Tuncbag *et al.*, 2009), in our opinion, none of them has been specifically designed to facilitate template-based protein docking. For example, for a given SCOP family, the SCOPPI database (Winter *et al.*, 2006) outputs all DDIs involving the query. The DDIs are grouped according to their partner domain. For each group, multiple sequence alignments with interacting residues marked are available for both the query and partner family. Other information available includes their in-house interface type, interface area and volume, screenshot of the interface, and links to related publications. Similarly, for a given Pfam family, 3DID outputs a list of DDIs grouped by their interface profile. IBIS outputs a list of PPIs involving a given query protein. The interactions are listed as DDIs, which are grouped by their partner domain and their binding site.

Although these databases are useful, they cannot be used to provide docking templates directly for many reasons. For example, (i) many of them cannot be queried with two domains simultaneously, which means that one has to collect and interpret output from two or more database searches; (ii) it is often the case that the user has to work his way through a long list of complexes, which means that one often does not model as accurately as possible the 3D structure of a protein-protein complex because all available PPI information was not used; (iii) binding sites and domain interactions of a given query domain cannot be visualised interactively in a common coordinate frame; (iv) most databases are based on an "interface" classification instead of "binding site", which means the fact that binding sites with domain families are often conserved is not exploited; (v) knowledge of existing protein interaction modes is often not available in an easily accessible way, and so cannot easily be incorporated into docking algorithms, e.g. current classifications have too many 'types' of interfaces to be useful; (vi) none of them allows to connect to a docking server using selected PPI information.

Chapter 3

Introducing KBDOCK – An Integrated Database of 3D Protein Domain Interactions

Contents

3.1 Introduction

This chapter describes the data selection, data enrichment, data transformation, and data integration steps which I performed to build an integrated relational database which I call KBDOCK. In Chapter 2, we have seen that there exist several resources collecting structural information on protein domains and their pair-wise interactions (review by Tuncbag *et al.*, 2009). Our wish is to reuse as much as possible such resources to achieve our aim. Since PPI structural data are not available in a "structured" form for direct input to data mining algorithms such as clustering or classification, it is essential to identify and collect all relevant data in one place. From our discussion of proteins and their interactions in Chapter 2, we have seen that proteins are often grouped into protein domain families and proteins which belong to a specific domain family have similar sequences and share similar folds, and hence similar biological functions. Moreover, protein-protein interactions are often described in terms of domain-domain interactions. For example, Figure 3.1 shows sequences and structures of the enzyme inhibitor *Kunitz BPTI* domain family. This figure shows some protein-protein complexes involving the *Kunitz BPTI* domain family. In terms of overall topology, from this figure, one can observe that the *Kunitz BPTI* family has roughly only two "binding sites" (one at the "north" and one at the "south"). It is much easier to see this feature from the 3D structures than from the primary structures because distant amino acids in a protein sequence may be close in its 3D structure. Hence, as our first line of attack, we wish to describe the spatial nature of PPIs at the protein domain family level for all PPIs with known 3D structures. To do this in a way which is suitable for subsequent data mining applications, it is important to identify and collect all relevant data from the diverse sources into one integrated database.

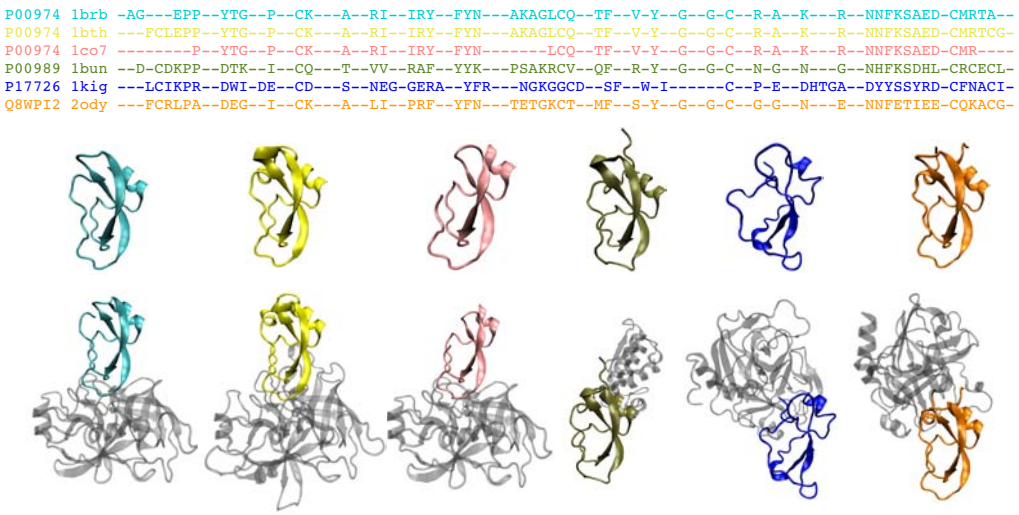


Figure 3.1: This figure shows six sequences and the corresponding 3D structures of the enzyme inhibitor *Kunitz BPTI* domain family. It also shows six enzyme-inhibitor complexes (with the enzyme in gray, *Kunitz BPTI* inhibitor in color). We can observe that the *Kunitz BPTI* domain family involves roughly two binding sites. In the first three complexes, the binding region is to the "south" while in the last three complexes, the binding region is to the "north". One of our KDD aims is to define useful ways of classifying domain-domain interactions according to the spatial position of their binding site for every protein domain family.

3.2 The Three Selected Data Sources

The KBDOCK database is built from three primary data sources, namely the Pfam database (Finn *et al.*, 2010), the 3DID database (Stein *et al.*, 2010), and the PDB (Berman *et al.*, 2002). Domain family information such as multiple sequence alignments is obtained from Pfam. Domain-domain interactions for all available 3D structures are obtained from 3DID, and protein 3D coordinates are obtained from the PDB. These three databases and the reasons for which we chose them are discussed in more detail in the following sub-sections.

3.2.1 The Pfam Protein Domain Family Database

Since its first release in 1996, Pfam (Finn *et al.*, 2010) has become the most widely used database of protein families and domain families. The current Pfam database (version 26.0) contains 13,672 manually curated protein domain families (known as Pfam-A or simply Pfam). Pfam classifies 80% of UniProtKB sequences. Automatically generated families (Pfam-B) are available separately. Of the 13,672 Pfam entries, 26.5% of them have no functional annotation. Pfam entries are classified into four categories namely “family”, “domain”, “repeat” and “motif” (Bateman *et al.*, 2002). A family contains sequence-related members. Some families can be defined more specifically as domains, repeat or motif. A domain is an independent structural unit, or a reusable sequence unit that may be found in multiple protein contexts. On the other hand, repeats occur in tandem to form a globular domain. Motifs describe short sequence units found outside globular domains. In this thesis, we use the term domain family to refer to any of the above Pfam entries.

A Pfam family is built in a four-step process: (i) building a manually-curated high-quality multiple sequence alignment (called the seed alignment); (ii) constructing a sequence profile using a hidden Markov model (HMM) from the seed alignment using HMMER3;[8] (iii) searching the HMM profile against the UniProtKB sequence database, and (iv) choosing manually-curated family-specific statistical thresholds (Finn *et al.*, 2010). All sequence regions that score above the threshold are included in the full alignment for the family.

Recently, Pfam introduced the notion of “Pfam clans” which consist of very closely-related protein families Finn *et al.* (2006). To help assess whether families are closely-related, Pfam uses high sequence similarity in addition with structure and function similarity. A clan contains two or more Pfam families that may have a common evolutionary origin. The total number of clans in the current version of the Pfam database is 499 for a total of 13,672 Pfam entries. Since Pfam uses structure similarity to group families into clans, many of the Pfam clans have a similar family membership to SCOP superfamilies. However, there is not a one-to-one relationship between a Pfam clan and a SCOP superfamily. The main difference between Pfam and SCOP is that the Pfam classification is not confined to families that have a known 3D structure. Indeed, some Pfam clans contain groups of related families in which none of the members have a known 3D structure.

Pfam is publicly accessible at http://pfam.sanger.ac.uk/. All Pfam data can be downloaded from ftp://ftp.sanger.ac.uk/pub/databases/Pfam/. Data are available in various flat file formats (e.g. Fasta, Stockholm, Selex and MSF) and also as a set of MySQL relational database files. For example, a

[8] HMMER3 is available at http://hmmer.janelia.org/

Stockholm flat file for a given Pfam entry comprises a multiple sequence alignment of all UniProt sequences involving that Pfam. For those UniProt sequences which have a 3D structure in the PDB, Pfam lists their PDB codes and provides secondary structure information. Furthermore, using a 32-class amino acid classification, Pfam calculates for each Pfam entry a "consensus" amino acid sequence, which is derived from the family multiple sequence alignment. For example, a position in the multiple sequence alignment is considered to be "conserved" if 60% of the amino acids at that position belong to the same amino acid class. The first 20 of the 32 amino acid classes correspond to the 20 standard amino acids and the remaining 12 classes are as follows: alcohol, aliphatic, aromatic, charged, hydrophobic, negative, polar, positive, small, tiny, turnlike, and 'not conserved'.[9] In a similar way, Pfam also provides a consensus secondary structure sequence for every Pfam entry, which has at least one 3D structure in the PDB. Pfam is updated almost annually. Here, Pfam version 24.0 was used. This contains 11,912 Pfam-A entries and 3,132 of them are grouped into 423 Pfam clans.

3.2.2 The 3DID Domain-Domain Interaction Database

The 3DID database (Stein *et al.*, 2010) contains domain-domain and domain-peptide interactions for which 3D structures are available in the PDB. To collect domain-domain interactions, 3DID scans all the structures in the PDB and assigns Pfam domains to each individual structure using the Pfam domain assignment program HMMER3. To identify a domain-domain interface, 3DID calculates atomic contacts between two domains in the same structure, and a DDI is defined when there are at least five residue-residue contacts between the two domains. A residue-residue contact occurs if there is at least one hydrogen bond (N-O distances $\leq$ 3.5 Å), salt bridge (N-O distances $\leq$ 5.5 Å), or van de Waals interaction (C-C distances $\leq$ 5 Å). These interactions are classified as main-chain to main-chain, main-chain to side-chain, or side-chain to side-chain contacts. To remove crystallographic contacts, interfaces with small buried areas are disregarded. This means that 3DID may contain DDIs arising from crystal contacts. The 3DID database is publicly available at http://3did.irbbarcelona.org, and MySQL dump and flat files containing the full dataset are available for download. The 3DID database is updated weekly to include newly released PDB structures. We chose the 3DID database as our source of DDIs because it uses the Pfam classification to describe domains, and because it is one of the most complete and up-to-date structural DDI databases currently available. The version of 3DID used here (November 2009) contains a total of 140,612 DDIs drawn from 29,922 PDB structures. A total of 3,755 different Pfam families are involved in at least one DDI.

3.2.3 The Protein Data Bank

The Protein Data Bank (PDB; http://www.rcsb.org/pdb/; Berman *et al.*, 2012) is the main world-wide archive of structural data of biological macromolecules. The PDB started in 1974 with 7 structures and it has since grown to contain 83,266 protein related structures. Nearly 97% of them are

[9] see ftp://ftp.sanger.ac.uk/pub/databases/Pfam/current_release/userman.txt for full details on the 32 -class amino acid classification.

protein-related structures, and 3% are nucleic acids structures. The PDB contains several static cross-references to other major data resources. These external resources include SCOP, CATH and Pfam for domain annotation, GO Terms for functional annotation, DSSP for secondary structures, and MSDsite for ligand binding site annotation. In addition, PDB uses external programs to search its content. For example, BLAST, FASTA, PSI-BLAST, blast2seq, Smith-Waterman and Needleman-Wunsch are used for sequence search, and jFATCAT, jCE, TM-Align and TOPMATCH are used for structure search. PDB structures are available in a plain text file called the PDB format. Recently, the PDB introduced two new file formats namely the macromolecular crystallographic information file format (mmCIF) and a XML format (PDBML). Here, the original PDB format is used for compatibility with external programs such as DSSP, ProFit, Hex, and many others.

3.3 Representing and Querying Pfam and 3DID Data Using Prolog

Given the heterogeneous and complex nature of structural protein interaction data sets, familiarising oneself with such data cannot be achieved using standard data mining toolkit such as Weka[10] because these software require as input relational data. Therefore, as a quick way to get started with the raw Pfam and 3DID protein interaction data, I chose to use the Prolog language over other popular bio-languages such as Perl and Python. Here, I use the SWI-PROLOG system[11].

Prolog is a simple but powerful language for solving logic problems. It is also commonly used in database system applications. Hence, in order to facilitate data processing for KDD, all Pfam and 3DID data were converted to Prolog terms. It should be noted that Pfam and 3DID data are not congruent. In several cases, there are conflicts between the Pfam and 3DID data, such as the first and last residue numbers of a given domain. Moreover, Pfam data contain several artefacts, e.g. the position of a domain is not unspecified. Transforming all Pfam and 3DID data into Prolog terms allows an easy way to resolve most such conflicts.

Prolog's term unification feature allows quick data retrieval and easy processing. In Prolog, all data structures are called terms. A term is either a constant, a variable or a compound term. Syntactically, a constant is either a number or an "atom". A Prolog atom is a sequence of characters preceded by a lower case character (e.g. `i_am_an_atom`) or enclosed within single quotes (e.g. `'I am an atom'`). A variable is a sequence of characters preceded by an upper case character (e.g. `I_am_a_variable`) or preceded by an underscore (e.g. `_i_am_a_variable`). Compound terms allow the representation of data with substructure. A compound term consists of a functor followed by a sequence of one or more arguments. `maximal_asa(AminoAcid, MaxASA)` is an example of a compound term. Here, `maximal_asa` is the name of the functor and it has 2 arguments.

Prolog describe relations in terms of clauses. There are two types of clause: facts and rules. A rule has a head and a body. Clauses with empty bodies are called facts. Prolog facts are predicate expressions that declare the properties of objects, or relationships between objects in a database. For example, the Prolog fact `maximal_asa('GLY', 84).` declares that the amino acid 'GLY' has a maximal surface accessiblity of 84. On the other hand, the Prolog clause

[10] http://www.cs.waikato.ac.nz/ml/weka/
[11] http://www.swi-prolog.org/

```
is_surface(AminoAcid, ASA) :-
 maximal_asa(AminoAcid, MaxASA),
 ASA >= MaxASA.
```

is an example of Prolog rule. Here, the functor `is_surface` returns `true` or `false`. Once we have a database of facts and rules (known as a fact base), we can ask questions about the stored information. For example, if we suppose the Prolog fact base contains the fact `maximal_asa('GLY', 84)` then a simple query might be `maximal_asa('GLY', MaxASA).` To answer this query, the Prolog system unifies the two clauses `maximal_asa('GLY', MaxASA)` and `maximal_asa('GLY', 84)` in order to give the answer `MaxASA = 84`. Using this unification technique, Prolog can answer database queries as well as deduce new facts.

Using the above principle, I converted all of 3DID and Pfam data to Prolog facts. For example, I defined the Prolog term `pfam_pdb(PfamAC, PDB, Chain, Start, End, UniProtAC, UStart, UEnd)` to represent a PDB domain instance extracted from the Pfam Stockholm file. The term contains eight arguments namely the Pfam accession number, the PDB four-letter code, the chain identifier, the domain start and end region on the PDB chain, the UniProt identifier and the domain start and end on the UniProt sequence. `pfam_entry_type(PfamAC, Type)` represents the Pfam entry type. The Prolog term `ddi_3did(PDB, Chain1, Start1, End1, PfamAC1, PfamAC2, Chain2, Start2, End2, PfamAC2, PfamAC2)` represents a DDI instance extracted from the 3DID flat file.

The built-in Prolog function `findall(Object, Goal, List)` can be used to perform searches in a quick way. For example, by entering as query the following clause

```
findall( PfamAc-Start-End,
         pfam_pdb(PfamAc, '1avw', 'B', Start, End, _, _, _),
         ListPfamAc).
```

the Prolog engine will retrieve a list of domain regions involved in the chain labelled 'B' in the PDB structure '1avw'.

Furthermore, Prolog supports call-outs to other major programming languages such as C[12] and MySQL[13]. Thus it is easy to add new functionality to a Prolog application. For example, SWI-PROLOG does not support regular expressions (regex). Hence, I wrote a Prolog / C function called `pl_regmatch(term_t myregex, term_t mystring, term_t result)` to allow regex queries to be performed within the Prolog system. Here, `result` equals `true` if `mystring` matches `myregex`. I use call-outs to C mainly to speed file parsing and other slow algorithms such as the Needleman-Wunsch's dynamic programming algorithm for sequence alignment. In addition, Prolog allows to execute Bourne shell commands to the operating system. Hence, calling external programs such as DSSP, HMMER, NRDB, and ProFit is easily done.

[12] SWI-PROLOG / C interface
http://www.swi-prolog.org/pldoc/doc_for?object=section(1,'9',swi('/doc/Manual/foreign.html'))
[13] SWI-PROLOG / ODBC interface
http://www.swi-prolog.org/pldoc/package/odbc.html

3.4 Collecting Representative Biological Hetero Structural PPIs

3.4.1 Classifying DDIs as Intra, Homo and Hetero

As discussed in Chapter 2, studies of structural PPIs often distinguish between homo and hetero protein-protein complexes (Ofran and Rost, 2003*a*, Guharoy and Chakrabarti, 2005, 2007) while others distinguish between enzyme-inhibitor and antibody-antigen (Lo Conte *et al.*, 1999). Although the 3DID database stores all observed DDIs, our main goal is to predict the 3D structures of heteromeric PPIs, because these are often the most difficult structures to solve experimentally (Ezkurdia *et al.*, 2009, Jones and Thornton, 1996). Therefore, for each protein domain family in Pfam, all DDIs involving that Pfam are extracted from 3DID and are classified as either “intra”, “homo”, or “hetero”. Figure 3.2 illustrates these types of domain interactions schematically. I consider a DDI to be intra if the interacting domains belong to a single protein chain, and homo if the interacting domains belong to different instances of the same protein chain in a given PDB structure. Otherwise, the interaction is considered to be hetero. For example, Figure 3.3 shows a homo interaction between two identical protein chains (PDB 1bmo). Here, only hetero DDIs are considered further.

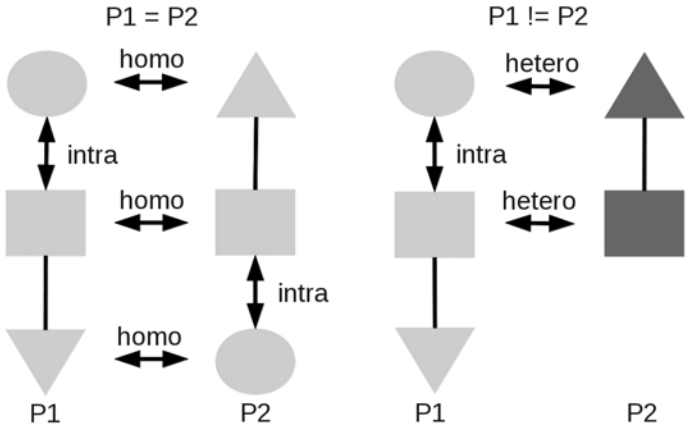


Figure 3.2: Schematic illustration of the different types of DDI that may occur between two protein chains, P1 and P2. Protein chains can contain one or more domains and domains may be in direct contact hence forming domain-domain interfaces (represented by lines with arrow heads). However, some domains are connected by linker regions (represented by straight lines; see e.g. Figure 3.3 PDB 2c4b) thus there is no domain-domain interface. Each shape (circle, rectangle, triangle) represents a different Pfam domain.

3.4.2 Distinguishing Between Crystallographic and Biological Contacts

As discussed in Chapter 2, some proteins are made of multiple polypeptide chains. Moreover, some proteins often form additional interfaces during the crystallization process, which do not exist in solution. Since the polypeptide chains are often identical, it is difficult to distinguish the biological interface from crystallographic artefacts. Hence, several groups have developed ways to assign the quaternary structures and to identify the biological contacts in crystal structures. PISA (Krissinel and Henrick, 2007), previously known as PQS, is one of the most widely used software for protein

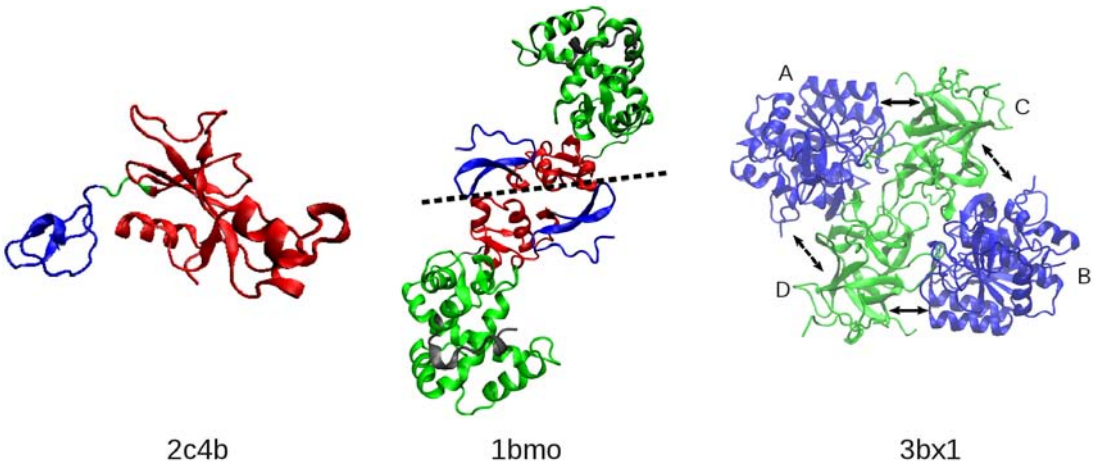


Figure 3.3: PDB 2c4b shows an example of a domain linker (in green) between the *Squash* (in blue) and *Ribonuclease* (in red) domains. Thus, this example does not involve an intra DDI because the two domains are not in physical contact. PDB 1bmo shows an example of a homo PPI interface formed by two symmetrical hetero DDIs (between the red and blue domains). PDB 3bx1 is a complex between an *alpha-amylase* (blue) and *Kunitz inhibitor* (green). The biological hetero DDI (solid arrow) is between chain A and C; and chain B and D. The other interfaces formed are crystal artefacts (dotted arrows).

quaternary structure analysis. For example, a PDB file usually contains the quaternary structure information provided by the authors or by PISA. However such information is often incomplete or inconsistent (Ponstingl *et al.*, 2003).

It has been shown that biological interactions usually have larger interfacial areas than non-biological interactions (Janin and Rodier, 1995, Carugo and Argos, 1997). Several groups have developed machine learning methods to distinguish between biological contacts and crystallographic contacts using features such as interface area, and chemical complementarity of interface residues. These include NOXclass (Zhu *et al.*, 2006) and DiMoVo (Bernauer *et al.*, 2008), for example.

In KBDOCK, biological and crystallographic contacts are distinguished using interface areas. We use the DSSP program (Kabsch and Sander, 1983) to calculate the solvent accessible surfaces (ASA) buried within each domain interface. If a given domain has multiple interactions with other identical domains, e.g. due to crystal packing, we assume that the interaction with the largest buried SAS corresponds to the biological interaction, and only this DDI is retained. For example, Figure 3.3 shows a crystallographic structure (PDB 3bx1) of a complex between the enzyme alpha-amylase and the Kunitz inhibitor. In this example, the biological interaction is between chain A and C, or chain B and D. The other interfaces are crystal artefacts. However, in a few cases, e.g. PDB 1s6v, the biological interface has a smaller area than the crystallographic one.

3.4.3 Obtaining a Non-Redundant Set of DDIs

Sequence clustering methods have been used by sequence databases to provide non-redundant sets of sequences for various purposes such as to provide functional annotation or to reduce com-

putation in sequence database search. For example, UniProtKB provides three databases namely UniRef100, UniRef90 and UniRef50 for sequence identity cut-off level of 100%, 90% and 50%, respectively. For example, in UniRef100 identical sequences (i.e. 100%) are presented as a single entry. There exist several sequence clustering programs.[14] Examples include NRDB90 (Holm and Sander, 1998) and CD-HIT (Li *et al.*, 2001).

In order to provide a non-biased set of 3D structures of DDIs, and since 3DID collects all observed DDIs, it is important to detect and eliminate duplicate or near-duplicate DDIs which may arise in several ways. For example, the same protein complex might have been solved under different crystallographic conditions, or a single crystal structure can sometimes contain different copies of the same complex. In order to deal with such cases in KBDOCK, the sequences of the DDI partners are concatenated and written to a FASTA file, and the NRDB90 program (Holm and Sander, 1998) is used with a similarity threshold of 99% to collect automatically a list of distinct non-redundant (NR) DDIs. It is worth noting that because we consider every structure to be useful, (e.g. similar pairs of proteins can interact in different ways) a high similarity threshold is used in order to retain as many non-duplicate structures as possible.

3.5 Annotating DDIs with Sequence and Structural Information

3.5.1 Identifying Conserved PDB Residues Using Pfam Consensus Sequences

As discussed in Chapter 2, amino acid residues in protein-protein interfaces are often more conserved than other surface residues. Indeed, due to evolutionary pressure, active site residues are often less likely to undergo mutation than other residue positions (Zvelebil *et al.*, 1987). This phenomenon has been exploited previously by several groups to predict molecular interaction sites (see reviews by de Vries and Bonvin, 2008; Ezkurdia *et al.*, 2009; Fernandez-Recio, 2011). For example, the evolutionary trace method of Lichtarge *et al.* (1996) relies on consensus sequence alignments to identify functional sites in new protein 3D structures.

On the other hand, some biologists believe that proteins may mutate a few of their interface amino acids in the course of evolution (see Pazos *et al.*, 1997 and references therein). The residues around the interface in a protein-protein complex can mutate in a coordinated way. For example, a mutation on one side of the interface might need to be stabilized by a complementary change in another residue of the other side of the interface. These mutations are thus termed correlated mutations. The existence of correlated mutated surface residues in multiple sequence alignments of pairs of protein families can be used to predict the physical location of protein-protein interfaces. Pazos *et al.* (1997) found that correlated mutations may be sufficient in many cases for discriminating near-native from incorrect docking predictions. Hence, domain family multiple sequence alignments which have been augmented with 3D interface information may represent an important asset for such studies and others.

In KBDOCK, the non-redundant sets of hetero DDIs are annotated with conserved amino acid information from Pfam. For each Pfam domain family, the Pfam database provides a multiple se-

[14] A list of available sequence clustering programs can be found at http://wikipedia.org/wiki/Sequence_clustering

quence alignment and a consensus sequence of all UniProt sequences belonging to that family (see Section 3.2.1). We follow the Pfam convention of considering a residue to be conserved if at least 60% of the amino acids at a given position in the multiple sequence alignment are of the same amino acid type. However, because Pfam uses UniProt sequences rather than PDB structures, and because PDB structures may contain gaps or unresolved regions, we align each PDB sequence with its Pfam-aligned UniProt sequence using the Needleman-Wunsch global alignment algorithm in order to map every PDB residue to its corresponding Pfam consensus position.[15] This mapping allows the Pfam consensus information to be transferred to each PDB residue position in order to allow conserved residues to be identified. Figure 3.4 shows the Pfam consensus-based sequence alignments of two example Pfam domain families, namely *Kazal 1* and *Kunitz BPTI*.

```
Consensus_PF00050          ....................Cs....tphpP.VC.G.o.D.s.p.T.Y.sNcCth.....C........pp.s.t.s.l.p.l.t.+..G.C.
PF00050_1cgj_I_7_56        ---------AKCYNELNG--CT---LE-YRP-VC-G-T-D-G-D-T-Y-PNECVL-----C-F-E-NR-KR-Q-T-S-I-L-I-Q-K-SGPC-
PF00050_1ppf_I_8_56        -----------CSEYPKP-ACT----LEYRP-LC-G-S-D-N-K-T-Y-GNKCNF-----C-N-A-VV-ES-N-G-T-L-T-L-S-H-FGKC-
PF00050_1tgs_I_3_56        ------QREATCTSEVSG--CP----KIYNP-VC-G-T-D-G-I-T-Y-SNECVL-----C-S-E-NK-KR-Q-T-P-V-L-I-Q-K-SGPC-
PF00050_1yu6_D_8_56        -----------CSEYPKP-ACT----LEYRP-LC-G-S-D-N-K-T-Y-GNKCNF-----C-N-A-VV-ES-N-G-T-L-T-L-S-H-FGKC-
PF00050_2p6a_D_152_210     VDQTNNAYCVTCNRI-----CPE-PASSEQY-LC-G-N-D-G-V-T-Y-SSACHLRK-ATC--------LL-G-R-S-I-G-L-A-Y-EGKC-
PF00050_3sgb_I_8_56        -----------CSEYPKP-ACT----LEYRP-LC-G-S-D-N-K-T-Y-GNKCNF-----C-N-A-VV-ES-N-G-T-L-T-L-S-H-FGKC-

Consensus_PF00014          ....C..sh.ptG..s.C.p...s..hh.....+a..aYs...spsppCp..pF..h.Y..u.G.C.t.G..N...p..NpFtopcc.CpptC..
PF00014_1brb_I_4_56        -AG---EPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTA--
PF00014_1brc_I_4_56        ---VCSEQA-ETG--P-C-R---A--MI----SRW--YFD---VTEGKCA--PF--F-Y--G-G-C-G-G--N---R--NNFDTEEY-CMAVCG-
PF00014_1bth_P_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_1bun_B_6_58        --D-CDKPP-DTK--I-C-Q---T--VV----RAF--YYK---PSAKRCV--QF--R-Y--G-G-C-N-G--N---G--NHFKSDHL-CRCECL-
PF00014_1bzx_I_504_556     ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_1co7_I_4_56        --------P-YTG--P-C-K---A--RI----IRY--FYN-------LCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMR----
PF00014_1eaw_B_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_1f7z_I_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_1fak_I_0_55        APDFCLEPP-YDG--P-C-R---A---L---HLRY--FYN---AKAGLCQ--TF----Y-YG-G-C--LA--K---R--NNFESAED-CMRTC--
PF00014_1fy8_I_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_1kig_I_504_560     ---LCIKPR-DWI-DE-C-D---S--NEG--GERA--YFR---NGKGGCD--SF--W-I------C-P-E--DHTGA--DYYSSYRD-CFNACI-
PF00014_1t8l_D_4_56        ---FCLEPP-YTG--P-C----MA--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CLRTCG-
PF00014_1taw_B_4_56        ---VCSEQA-ETG--P-C-R---A--MI----SRW--YFD---VTEGKCA--PF--F-Y--G-G-C-G-G--N---R--NNFDTEEY-CMAVCG-
PF00014_1tfx_C_4_56        ---FCFLEE-DPG--I-C-R---G--YI----TRY--FYN---NQTKQCE--RF--K-Y--G-G-C-L-G--N---M--NNFETLEE-CKNIC-E
PF00014_1yc0_I_249_301     --Y-CLASN-KVG--R-C-R---G--SF----PRW--YYD---PTEQICK--SF--V-Y--G-G-C-L-G--N---K--NNYLREEE-CILACR-
PF00014_1zjd_B_4_56        ---VCSEQA-ETG--P-C-R---A--MI----SRW--YFD---VTEGKCA--PF--F-Y--G-G-C-G-G--N---R--NNFDTEEY-CMAVCG-
PF00014_1zr0_B_4_56        ---ICLLPL-DYG--P-C-R---A--LL----LRY--YYD---RYTQSCR--QF--L-Y--G-G-C-E-G--N---A--NNFYTWEA-CDDAC-W
PF00014_2fi4_I_4_56        ---FCLEPP-YTG--P-S-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_2ijo_I_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_2kai_I_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_2ody_F_20_72       ---FCRLPA-DEG--I-C-K---A--LI----PRF--YFN---TETGKCT--MF--S-Y--G-G-C-G-G--N---E--NNFETIEE-CQKACG-
PF00014_2r9p_F_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_2ra3_C_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_2tpi_I_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTCG-
PF00014_3btt_I_504_556     ---FCLEPP-YTG--P-C-----TA--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CLRTCG-
PF00014_3d65_I_6_58        ---FCELPA-DTG--P-C-R---V--RF----PSF--YYN---PDEKKCL--EF--I-Y--G-G-C-E-G--N---A--NNFITKEE-CESTC-A
PF00014_3gym_J_4_56        ---FCLEPP-YTG--P-C-K---A--RI----IRY--FYN---AKAGLCQ--TF--V-Y--G-G-C-R-A--K---R--NNFKSAED-CMRTC--
```

Figure 3.4: Two examples of Pfam consensus-based sequence alignments of PDB domain sequences. The *Kazal 1* (PF00050) domain family has 6 NR domain sequences which are involved in hetero DDIs. The *Kunitz BPTI* (PF00014) domain family has 27 NR domain sequences. Core interface residues are shown in green and rim residues are shown in blue. The consensus sequence is shown in orange.

3.5.2 Classifying Interface Residues as Core or Rim

Recently, protein interfaces have been described in terms of "core" and "rim" regions (Lo Conte *et al.*, 1999, Chakrabarti and Janin, 2002). The core region is defined by atoms that are fully buried on complex formation while the surrounding rim region is defined by atoms that remains partially accessible (Figure 3 in Chakrabarti and Janin, 2002). In their work, Chakrabarti and Janin identified interface residues as those which lose solvent accessibility on complex formation. They defined an interface residue as core if it contain at least one buried atom, otherwise it is defined as rim.

[15] Example of a PDB/UniProt SwissProt residue mapping program is PDBSWS (Martin, 2005).

In KBDOCK, interface residue-residue pairings are provided by the 3DID database (Section 3.2.2). We used DSSP to calculate the change in solvent accessibility for each interaction residue between the separate and complexed structures of each domain. We defined an interaction residue to be a core interface residue if it loses at least 75% of its accessible surface area on going from the isolated to the complexed structure. Otherwise, it is considered to be a rim interface residue. Figure 3.5 illustrates the notion of a core and rim residue using the trypsin/Kunitz inhibitor complex (PDB 1brb). The domain family sequence alignments shown in Figure 3.4 have been annotated with core and rim residues. Core and rim residues may provide useful additional information when predicting binding sites in homologous proteins.

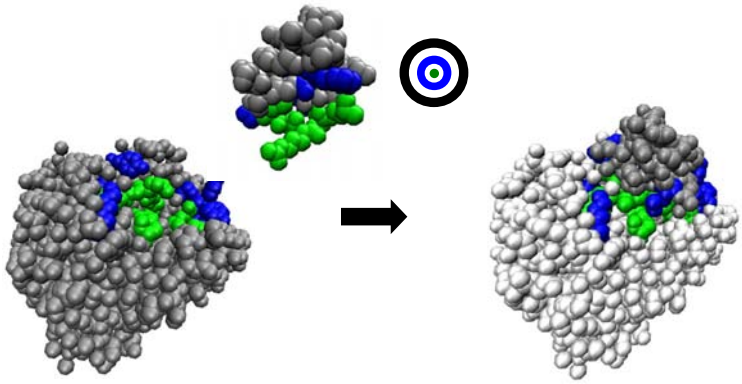


Figure 3.5: This figure illustrates the notion of core and rim residue. The core of the interface (in green) is surrounded by partially accessible rim residues (in blue). The core and rim residues can be thought of as forming a kind of "target" though such targets are certainly not perfectly circular. Here, the complex trypsin/Kunitz inhibitor (PDB 1brb) is used for illustration.

3.5.3 Adding Secondary Structure Information Using DSSP

We use DSSP to annotate all PDB residues in KBDOCK with secondary structure information. DSSP defines seven secondary structure types namely: α-helix, 3-10 helix, π-helix, extended strand, isolated beta-bridge, hydrogen bonded turn, bend, and loop/irregular. Some examples of these secondary structure types are illustrated in Figure 2.3.

In addition, we annotate every PDB residue in KBDOCK as surface or buried. We consider a residue to be at the surface if its relative accessibility is greater or equal to 5% (Jones and Thornton, 1997). Relative ASA = (ASA * 100) / maximum ASA. The maximum accessibility of an amino acid is equal to its accessible surface in a pentapeptide in an extended conformation. We use the maximum ASA theoretical values given in Zhou and Shan (2001).[16] Lastly, since 3DID does not specify the interface non-covalent interactions, for every interface residue-residue pairing, we calculate and store these non-covalent interaction types namely salt bridge, hydrogen bond and van de Waals. However, for some 3DID residue-residue pairs, no specific interaction type is identified.

[16] Maximum ASA values for the twenty amino acids: Ala 106, Arg 248, Asn 157, Asp 163, Cys 135, Gln 198, Glu 194, Gly 84, His 184, Ile 169, Leu 164, Lys 205, Met 188, Phe 197, Pro 136, Ser 130, Thr 142, Trp 227, Tyr 222, Val 142 Å^2.

3.6 Superposing DDIs in 3D Space Using ProFit

In order to describe DDIs for every Pfam domain family, it is important to place all of the DDIs involving a given Pfam in a consistent orientation in 3D space, as illustrated in Figure 3.6. There exist many structural alignment programs.[17] However, only a few of them such as SSM (Krissinel and Henrick, 2004) and ProFit[18] allow specific regions to be aligned. ProFit is a least-squares fitting program based on McLachlan's algorithm (McLachlan, 1982). ProFit has several useful features, e.g. ability to specify structural regions (or zones) and atom subsets to fit, iterative updating and optimization of fitting zones, RMSD calculated over fitted region, ability to identify and fit zones from sequence alignment, and multiple structure fitting. We chose ProFit because it is available as a command-line program and thus can be launched easily within a Prolog program.

Our mapping between the Pfam consensus sequence and PDB residue numbers provides a convenient way to identify the conserved residue positions of all domains. Hence, it is straightforward to derive ProFit zones from these conserved positions and to place the PDB domains (along with their interacting partner) in a common coordinate frame (Figure 3.6). The fitting was performed on C_α atoms with no iteration.

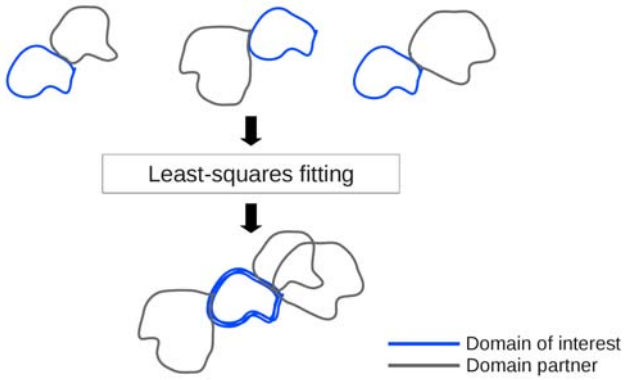


Figure 3.6: Superposing a set of Pfam domains (blue) in order to put their interaction partners (grey) in a common coordinate frame.

3.7 Summary of the KBDOCK Data Processing Steps

In summary, the main data processing steps to integrate data from the 3DID, Pfam, and PDB databases are as follows: (1) for each Pfam query, retrieve DDIs involving the query Pfam from 3DID; (2) classify the DDIs as intra, homo and hetero interactions and retain only hetero DDIs; (3) filter out crystal contacts using interface area criteria; (4) filter out duplicate or near-duplicate DDIs using sequence identity to obtain a list of non-redundant hetero DDIs; (5) align the corresponding PDB domain sequences (involving the query Pfam) against the Pfam consensus sequence to obtain a multiple sequence alignment; (6) identify conserved residue positions from the multiple sequence alignment; (7) superpose the corresponding PDB structures of the DDIs in a common coordinate

[17] A list of structural alignment programs is available at http://wikipedia.org/wiki/Structural_alignment_software
[18] http://bioinf.org.uk

frame; (8) annotate all PDB residues with secondary structure and solvent accessibility information; (9) classify residues as buried or surface; and (10) classify interacting residues as core and rim.

As mentioned in Section 3.3, all data processing and integration steps were made using a small set of Prolog programs. However, for some specific tasks, e.g. parsing PDB files, C was used for speed. All calculated data are written to flat files which are then uploaded to a relational database as described in the next section. Figure 3.7 summarises the main processing steps used to integrate data from the 3DID, Pfam and PDB databases.

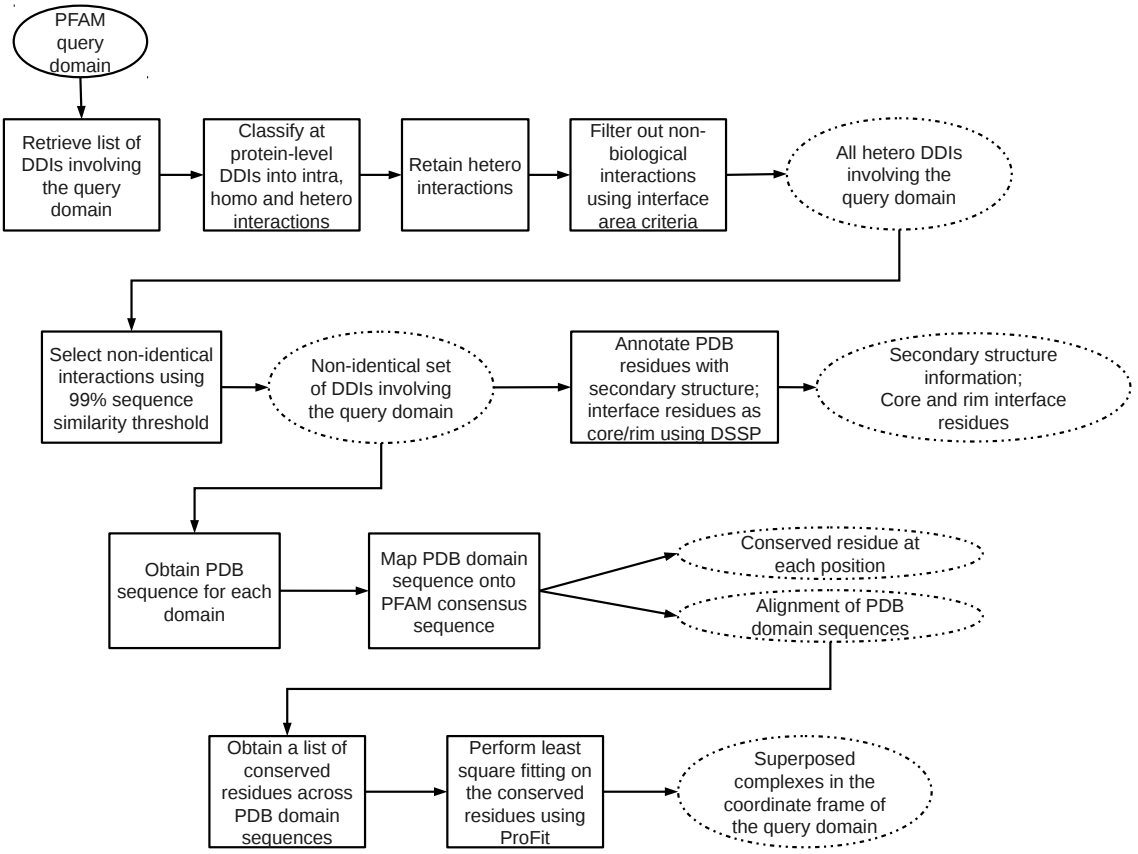


Figure 3.7: A flowchart showing the main processing steps used to integrate and enrich 3D structural DDI data from three data sources: 3DID, Pfam and PDB.

3.8 The KBDOCK Data Model

Although Prolog is a powerful programming language, a Prolog fact base is not very efficient for storing and manipulating large data sets. For this reason, we chose to store all calculated data in a relational database. Relational databases provide a unified mechanism for fast access to selected parts of the data (Hand *et al.*, 2001). In Hand *et al.* (2001), a data model is defined as a set of constructs that can be used to describe the structure of the data, plus a set of operations for manipulating the data. In a relational data model, data are presented in tables (relations). Table column names are attributes and rows are instances. The structured query language (SQL) is

a standard programming language based on relation algebra for defining database structures, for managining data, for managining access, and for managing changes.

We built the KBDOCK database using the open source MySQL relational database engine.[19] Figure 3.8 shows a simplified UML class diagram of the structure of the KBDOCK database (see Figure 3.9 for a relational data model). The three main classes are *PDB*, *Pfam_entry*, and *DDI*. For example, an instance of *Pfam_entry* has one or many *UniProt domain* instances, and a UniProt domain instance may have one (has zero) or several PDB domain instances. A *PDB domain* instance may participate in one or many *DDI* instance. A *DDI* instance has one or many *interface residue* instances.

The current version of KBDOCK stores a total of 2,721 non-redundant hetero DDIs for a total of 1,035 different Pfam domain families. To provide public access to the KBDOCK database, I implemented a web server which is available at http://kbdock.loria.fr.

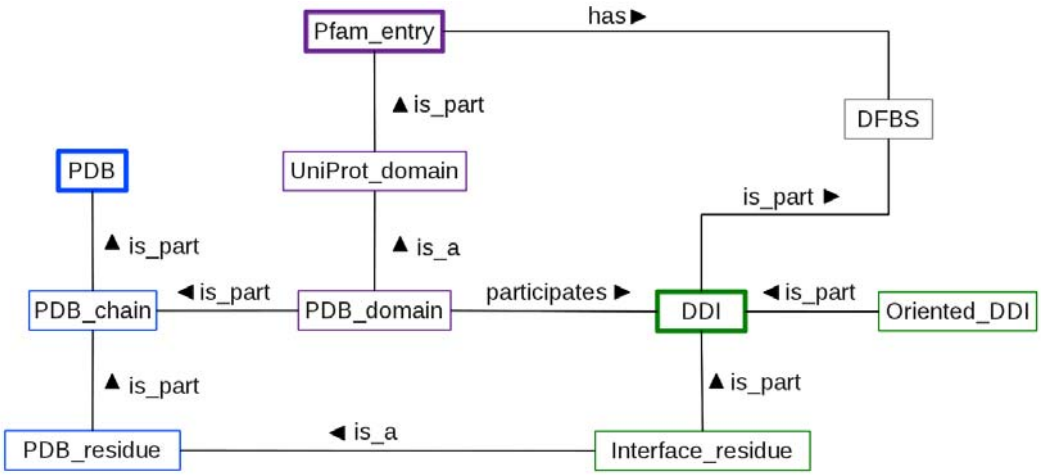


Figure 3.8: A simplified view of the UML class diagram of the KBDOCK database. Rectangles represent classes and edges represent relationships. There are 3 main classes: *PDB*, *Pfam_entry*, and *DDI*. The class *DFBS* is described in the next chapter. See Figure 3.9 for a detailed relational data model.

[19] http://www.mysql.com

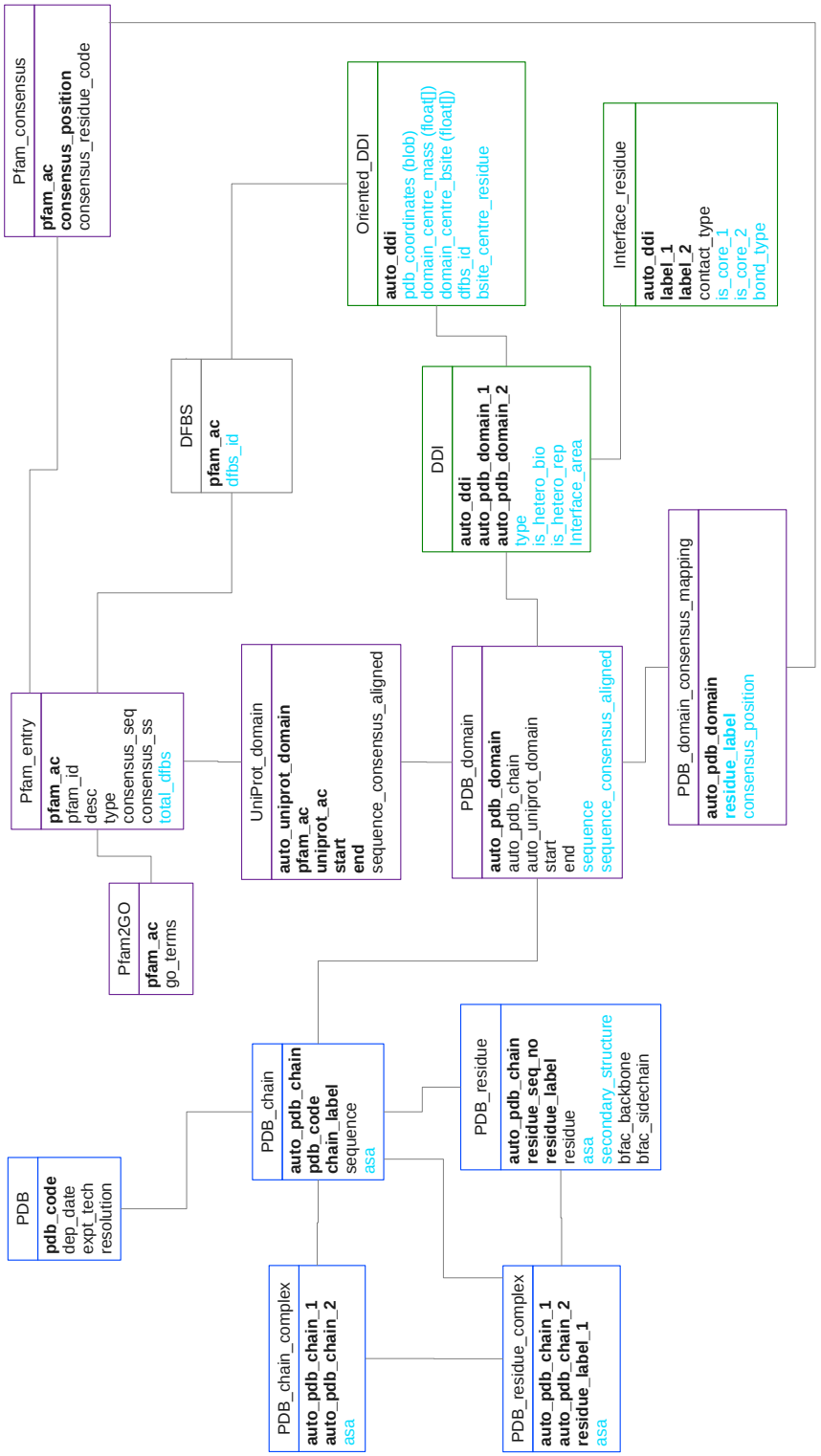


Figure 3.9: A relational data model of the KBDOCK database. Attributes which we calculated are in cyan. The *DFBS* entity is described in Chapter 4.

3.9 Exploring DDIs in Protein Domain Families with KBDOCK

3.9.1 Querying KBDOCK

Figure 3.10 shows an example of an SQL query and output to retrieve a non-redundant list of DDIs involving the query domain family *Kunitz_legume* (Pfam ID PF00197) from KBDOCK. The inhibitor *Kunitz_legume* interacts with 4 different enzymes namely *Trypsin* (PF00089), *Peptidase_S8* (PF00082), *Alpha-amylase* (PF00128), and *Thioredoxin* (PF00085). Figure 3.11 shows the Pfam consensus based sequence alignment and superposition of the *Kunitz_legume* PDB domains. The sequence alignment is augmented with core and rim interface residues as discussed previously (Section 3.5.1 and 3.5.2). The color rendering in the sequence alignment display is done using HTML. In the future, a more sophisticated tool, e.g. Jalview (Waterhouse *et al.*, 2009) may be used. Visualisation of the DDI superposition is done in VMD (Humphrey *et al.*, 1996). A small TCL script was written to allow automatic loading of PDB files in VMD with a specific molecule representation. This helps the visualisation of several domain family superpositions (see next subsection). For the web server (see Appendix A), the Jmol plugin (http://jmol.sourceforge.net) is used.

Figure 3.11 shows the multiple sequence alignment of the *Kunitz_legume* domain family along with a 3D view of the superposition of the members of this family and their interaction partners. This figure demonstrates that a multiple sequence alignment is often not sufficient to understand the 3D nature of interaction modes. From the 3D superposition, it is clear that the *Kunitz_legume* domain has roughly four binding sites. However, this is not at all evident from the multiple sequence alignment even when it is color-coded to highlight the known binding site residues. This example emphasises the importance of analysing structural interactions in terms of the 3D spatial relationships.

3.9.2 Exploring Pfam Domain Family Superpositions

In KDD, data exploration is a useful way to get a feel for the data. In most KDD processes, data exploration often involves calculating means and standard deviations, or plotting statistical charts in order to observe various distributions of the data (Hand *et al.*, 2001).

However, in some KDD cases, the manual nature of data exploration makes it possible to explore only a few cases (Hand *et al.*, 2001). For example, in our study of 3D protein-protein interactions in Pfam domain families, data exploration tasks may include e.g. observing the distribution of interaction modes and the diversity of interacting partners in Pfam domain families. The KBDOCK's DDI superpositions provides a natural way to explore DDIs in Pfam domain families.

As a more extensive example of data exploration, Table 3.1 lists 20 different Pfam domain families in KBDOCK. These families were chosen because they contain multiple instances of DDIs and because some of them interact with more than one Pfam partner. For example, the *Kunitz legume* domain family has five non-redundant hetero DDIs which correspond to four distinct Pfam partners (Figure 3.11). However, these numbers do not reveal the complexity and richness of the 3D nature of the interactions.

```
mysql> select pc.pdb,
    -> pdbd1.pfam_ac, pc1.chain, pdbd1.start, pdbd1.end,
    -> pdbd2.pfam_ac, pc2.chain, pdbd2.start, pdbd2.end
    ->   from ddi as d,
    ->   pdb_domain as pdbd1, pdb_chain as pc1,
    ->   pdb_domain as pdbd2, pdb_chain as pc2
    ->     where d.auto_pdb_domain_1  = pdbd1.auto_pdb_domain
    ->     and   d.auto_pdb_domain_2  = pdbd2.auto_pdb_domain
    ->     and   pdbd1.auto_pdb_chain = pc1.auto_pdb_chain
    ->     and   pdbd2.auto_pdb_chain = pc2.auto_pdb_chain
    ->     and   d.type               = 'hetero'
    ->     and   d.is_hetero_rep      = '1'
    ->     and   pdbd1.pfam_ac        = 'PF00197';

+------+---------+-------+-------+------+---------+-------+-------+------+
| pdb  | pfam_ac | chain | start | end  | pfam_ac | chain | start | end  |
+------+---------+-------+-------+------+---------+-------+-------+------+
| 1ava | PF00197 | D     | 5     | 177  | PF00128 | B     | 17    | 324  |
| 1avw | PF00197 | B     | 502   | 675  | PF00089 | A     | 16    | 238  |
| 2iwt | PF00197 | B     | 5     | 177  | PF00085 | A     | 14    | 118  |
| 2qyi | PF00197 | B     | 606   | 777  | PF00089 | A     | 16    | 238  |
| 3bx1 | PF00197 | D     | 5     | 177  | PF00082 | B     | 6     | 266  |
+------+---------+-------+-------+------+---------+-------+-------+------+
5 rows in set (0.76 sec)
```

Figure 3.10: Example of a KBDOCK database query and raw text output from MySQL to retrieve a non-redundant list of biological hetero DDIs involving the query Pfam domain PF00197.

In order to fully appreciate the 3D complexity and variety of domains and their interactions, Figure 3.12 shows the DDI superposition of the 20 domain families given in Table 3.1. These families consist of multiple DDI instances and it would be difficult to comprehend if one of the DDI was not superposed well. On the other hand, when the domain of interest is superposed, the spatial arrangement of the interacting partners can be appreciated much more easily. For example, the *Kunitz BPTI* domain has two inhibitory loops, one to the north and one to the south. *Kunitz BPTI* interacts with *Trypsin* and *Peptidase S7* at the 'north' loop and with *Trypsin* and *Phospholip A2 1* at the 'south' loop. Here, the number of interaction modes is visually apparent in contrast to the augmented sequence alignment in Figure 3.4.

On the other hand, in contrast to surface loop-rich *Kunitz legume*, the figure shows that inhibitor *Potato inhibit* interacts with two different Pfam domain namely *Trypsin* and *Peptidase S8* via a single binding region. Domain families having a single Pfam partner include *Ecotin* and *Squash* and *Ribonuclease*. Both *Ecotin* and *Squash* interact with *Trypsin* via a single binding region. *Ribonuclease* interact with *Barstar* via a single binding region. On the other hand, *Thioredoxin* interacts with seven different Pfam domains. The DDI superposition for this family show that there are roughly only one or two overlapping binding regions. Overall, visual inspection of domain family superpositions strongly suggest that Pfam domains have a few number of binding modes despite the high number of DDIs and distinct partners.

The DDI superpositions suggest that identifying distinct binding sites in domain families will provide a useful way to avoid the need of examining tens of similar DDIs which correspond to the same interaction mode. For example, all of the domain instances of *Ecotin* and *Squash* interact with

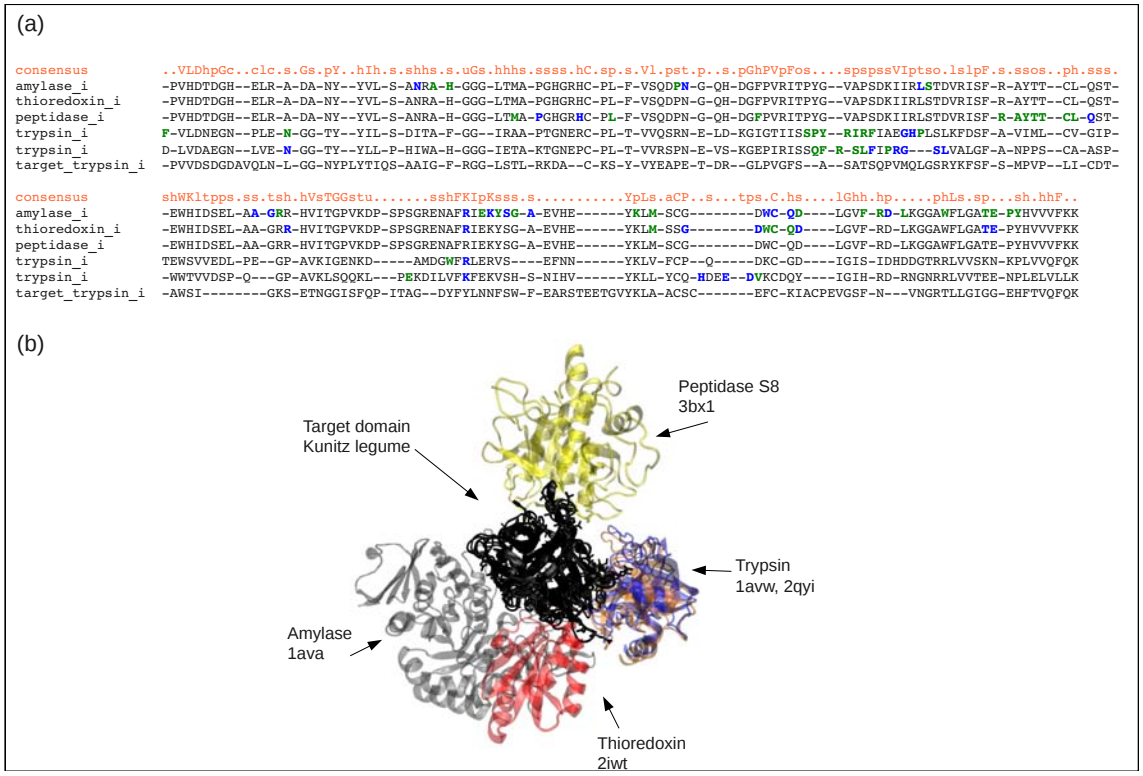


Figure 3.11: This figure illustrates the KBDOCK database output for the query domain *Kunitz_legume*. The query domain interacts with 4 different Pfam domains. **(a)** shows the "augmented" Pfam consensus based PDB sequence alignment. The consensus sequence is shown in orange. Core and rim interface residues are shown in green and blue, respectively. **(b)** The DDIs are superposed in the coordinate frame of the query. The *Kunitz_legume* domains are shown in black.

Trypsin via a single binding region. Their DDI superposition also shows that the DDIs are clearly very similar because they overlay very well. Therefore, one could examine only one representative DDI to find information to model a homologous DDI. Hence, there is the need to identify distinct binding regions and interaction modes in protein domain families.

Furthermore, for Pfam domains that have many different partners and several distinct binding sites, it is essential to obtain a representative complex for each distinct interaction mode. For example, *Actin*, *Lectin C*, *Pkinase* and *Trypsin* are all involved in interactions with several different Pfam domains and their DDI superpositions clearly show that these domain families interact in a number of different ways. As another example, the *Lys* domain family is a glycoside hydrolase enzyme family and has ten non-redundant hetero DDIs. The domain family superposition shows that there are approximately five binding regions on the *Lys* domain. Hence, it would be useful to group automatically these ten DDIs according to the similarity of their binding location. This would indicate the distinct binding regions in a given domain family.

PfamID	Pfam Name	Function	NR Hetero DDIs	Distinct Partners
PF00197	Kunitz legume	protease inhibitor	5	4
PF00014	Kunitz BPTI	protease inhibitor	27	3
PF00280	Potato inhibit	protease inhibitor	8	2
PF00089	Trypsin	protease	98	32
PF00062	Lys	hydrolase	10	4
PF00545	Ribonuclease	hydrolase	9	1
PF00022	Actin	protein binding	24	12
PF00059	Lectin C	glycoprotein binding	14	5
PF00111	Fer2	ferredoxin	14	5
PF00085	Thioredoxin	redox protein	8	7
PF03974	Ecotin	protease inhibitor	8	1
PF00299	Squash	protease inhibitor	7	1
PF00050	Kazal 1	protease inhibitor	6	3
PF00079	Serpin	protease inhibitor	6	2
PF00228	Bowman-Birk leg	protease inhibitor	5	1
PF00031	Cystatin	protease inhibitor	3	1
PF00082	Peptidase S8	protease	14	6
PF00128	Alpha amylase	hydrolase	5	4
PF00017	SH2	signalling protein	4	2
PF00069	Pkinase	kinase	24	15

Table 3.1: Total number of hetero DDIs in KBDOCK for 20 example Pfam entries. The last two columns give the number of NR hetero DDIs and the number of distinct partners (by Pfam). These Pfam domain families were selected because they have multiple instances of DDIs and some of them interact with more than one Pfam domain family.

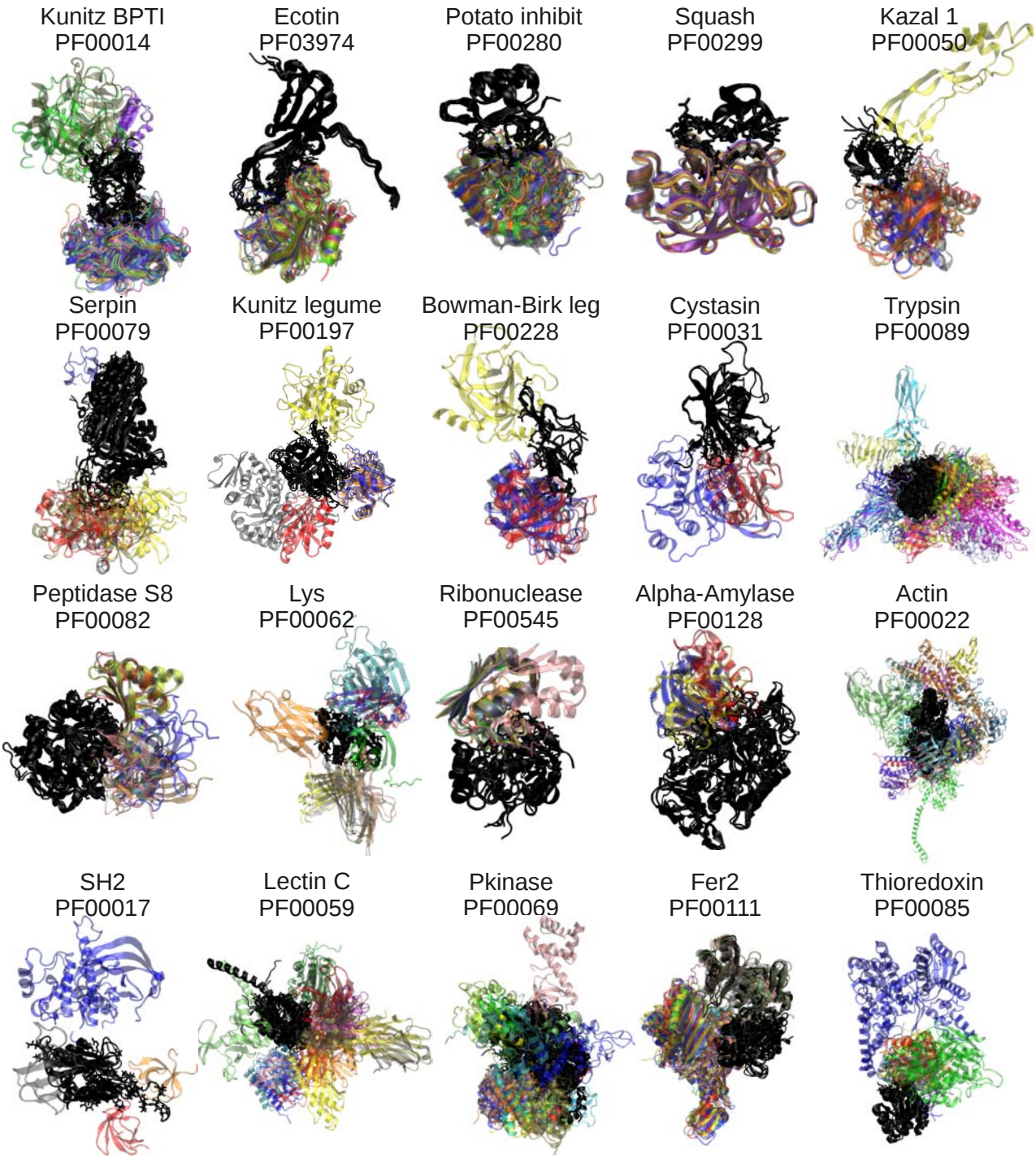


Figure 3.12: DDI superpositions for twenty example Pfam domains (Table 3.1) in the coordinate frame of the query. In each case, the query domain is shown in black. This figure was produced using VMD (Humphrey *et al.*, 1996). A TCL script was written to allow automatic loading of PDB files in VMD.

3.10 Conclusion

This chapter has introduced the integrated database, KBDOCK, which contains all available hetero DDIs. KBDOCK constitutes one of the largest collections of 3D hetero DDIs to date. One particular feature of KBDOCK that distinguishes it from other databases is that DDIs are placed in a common coordinate frame thanks to Pfam consensus-based sequence alignments. Furthermore, all DDIs in KBDOCK are annotated with core/rim, secondary structure, and amino acid conservation information. The data in KBDOCK are organised in a systematic way according to Pfam domain families and all data are stored in a relational database. This facilitates manipulating data accessing and querying.

Although other PPI databases such as SNAPPI-DB already exist (Jefferson *et al.*2007; see Section 2.4), in this thesis we chose to build a new integrated database because we wish to describe protein domain binding sites rather than domain-domain interfaces. We chose to use a subset of the 3DID database which corresponds to 'low-level' DDI data as our primary source of DDIs because it uses the Pfam domain classification to describe interactions in 3D structures and because it is one of the most up-to-date database currently available.

This chapter has shown that superposing DDIs of a Pfam domain family in a common coordinate frame provides a straightforward way to explore structural relationships between the members of a given Pfam. This constitutes a first step towards describing binding sites in Pfam domain families. From the examples of DDI superpositions which we have discussed in this chapter, we saw that most of them have a few binding regions. This raises the question of whether domain families can really have such large numbers of binding sites that other studies have suggested. For example, PIBASE (Davis and Sali, 2005) gives 30,975 binding site types for a total of 1,946 SCOP domain families. In the next chapter, we try to answer this question.

In terms of KDD, this chapter has described the data selection, data enrichment, data transformation, and data integration steps. This chapter also showed that DDI exploration was made easier thanks to the 3D superpositions that KBDOCK calculates for each domain family of interest. This has allowed us to identify and formulate our first KDD goal – to classify DDIs according to the spatial position of their binding sites for every Pfam domain family which has hetero DDIs. Classifying DDIs in protein domain families according to the spatial position of their binding site will provide many useful prospects for example, (i) to identify the distinct binding regions for any given domain family, (ii) to provide a systematic way to characterise domain family binding sites, and (iii) to facilitate DDI information retrieval for a given docking target. These topics are explored in detail in the following chapters.

Chapter 4

Spatial Clustering of Protein Domain Family Binding Sites

Contents

4.1 Previous Protein-Protein Interface Classifications

Several groups have developed ways of classifying protein-protein interfaces to aid understanding of protein-protein interactions. Since PPIs are often described in terms of DDIs, most of these classifications cluster interfaces between two inter-chain domains or between entire protein chains. For example, domain interface classifications have been developed by Davis and Sali (2005), Kim *et al.* (2006), and Stein *et al.* (2009). On the other hand, Shulman-Peleg *et al.* (2004), Keskin *et al.* (2004), and Aung *et al.* (2008) have developed protein interface classifications.

4.1.1 The PIBASE Domain-Domain Interface Classification

Davis and Sali (2005) define domain-domain interactions using both the SCOP and CATH domain definitions. They compute several properties, e.g. buried ASA, contacting residue types, contacting secondary structure types, interface chemical bonds (for a complete list see Table 1 in supplementary material of Davis *et al.*, 2005). They calculate "secondary structure topology fingerprints" for all domain interfaces and binding sites. They perform a first round of hierarchical clustering to remove redundancy and a second round of clustering to group the topology fingerprints to define domain-domain interface classes and binding site classes. Their approach gives 18,755 interface classes and 30,975 binding site classes for 1,946 SCOP domain families involving a total of 20,912 non-redundant domain-domain interactions having an interface area $\geq$300Å. Their classification is available as part of the PIBASE database (Davis and Sali, 2005).

4.1.2 The SCOPPI Domain-Domain Interface Classification

The SCOPPI database describes domain-domain interactions using the SCOP domain definition (Kim *et al.*, 2006 and Winter *et al.*, 2006). Kim *et al.* (2006) define a domain-domain interface as a pair of two interacting domain faces. For each SCOP domain family, they defined "face vectors" which contain a list of ones and zeros to represent the contacting and non-contacting residues of each domain, respectively. The face vectors are then grouped according to a cosine similarity function. To reduce computational cost, only representative faces from each cluster are superposed. These representative faces are then clustered again according to their "face overlap" (spatial overlap of face atoms) and "face angle" (between a pair of face centroids and the common centroid of the domains) (Kim *et al.*, 2006). They found that 34% of the SCOP domain families have only one face, 25% have two faces, 15% have three faces, and 9% have four faces (supplementary material of Kim *et al.*, 2006). Kim *et al.* combine pairs of faces to define what they call "interface types". Their dataset contains 92,979 domain-domain interfaces which are grouped into 8,381 distinct interface types (Winter *et al.*, 2006). Their classification of domain interfaces is available online as part of the SCOPPI database (Winter *et al.*, 2006).

4.1.3 The 3DID Database of Domain-Domain Interfaces

As described earlier (Section 3.2.2), the 3DID database describes DDIs using the Pfam domain definition (Stein *et al.*, 2005). Stein *et al.* start by defining domain face types by applying complete linkage hierarchical clustering to identify groups of common face contacting residues within a Pfam domain family using the HMM domain family sequence profiles (Stein *et al.*, 2009). Similar to the work of Kim *et al.* (2006), Stein *et al.* combine two domain face types to define what they called an "interaction topology". Using this approach, they found that 50% of 159,557 domain-domain interfaces have only one interaction topology and only a small fraction show ten or more interaction topologies. Stein *et al.* used a further round of hierarchical clustering to group together face types that overlap by 25% or more (see Figure 3 of Stein *et al.*, 2010). They called these clusters "global interface clusters" and they found that 50% of 4,186 Pfam domain families have one or two global interface clusters, and few have 10 or more (see Figure 3 of Stein *et al.*, 2010). Their classification is available as part of the 3DID database (Stein *et al.*, 2005).

4.1.4 The I2I-SiteEngine Protein-Protein Interface Classification

The I2I-SiteEngine approach developed by Shulman-Peleg *et al.* (2004, 2005) also clusters protein-protein interfaces. In this method, an interface is represented by a pair of interacting surface regions (binding sites) and a set of pseudo-centers representing exposed functional atom groups. Thus, Shulman-Peleg *et al.* define an interface as two sets of interacting triangles that consist of triplets of functional groups forming three inter-chain interactions. They use a hashing-based algorithm to detect whether two triplets have complementary properties. Complementary triplets are then superposed before scoring the shape and physicochemical properties of the interfaces. Their approach was performed on a small dataset of 64 protein-protein interfaces. These were grouped into 22 clusters. Similar to the work of Aung *et al.* (2008), they found that pairs of proteins with dissimilar folds often have similar interfaces. The I2I-SiteEngine is also available as a web server at http://bioinfo3d.cs.tau.ac.il/I2I-SiteEngine/.

4.1.5 Keskin's Classification of Protein-Protein Interfaces

Keskin *et al.* (2004) clustered protein-protein interfaces using the backbone of Cα atoms from interacting residues and their neighbouring residues. Their approach consists of three steps: (i) interface fragments are superposed and scored using a geometric hashing-based algorithm, (ii) similar interface fragments are then clustered, and finally (iii) pairs of whole proteins with similar interfaces are clustered using a heuristic iterative hierarchical clustering procedure. Their procedure groups 21,686 protein-protein interfaces into 3,799 clusters. When a 50% sequence similarity filtering is applied together with a merging process to avoid clusters with too few members, 103 clusters are obtained. Keskin *et al.* divide the 3,799 clusters into three main categories, which they call Type I, II and III. Type I and II clusters contain pairs of interacting proteins that are homologous. Type I members have similar interfaces but Type II members have dissimilar interfaces. Type III clusters contain pairs of interacting proteins that are non-homologous but have similar faces (one side of the interface). Type I, II and III clusters are illustrated in Figure 4.1. Out of 103 non-redundant clusters,

54 belong to Type I and II. They subsequently extended this work to characterise interface features between the different cluster types (Keskin and Nussinov, 2007). This is discussed in more details in Chapter 5.

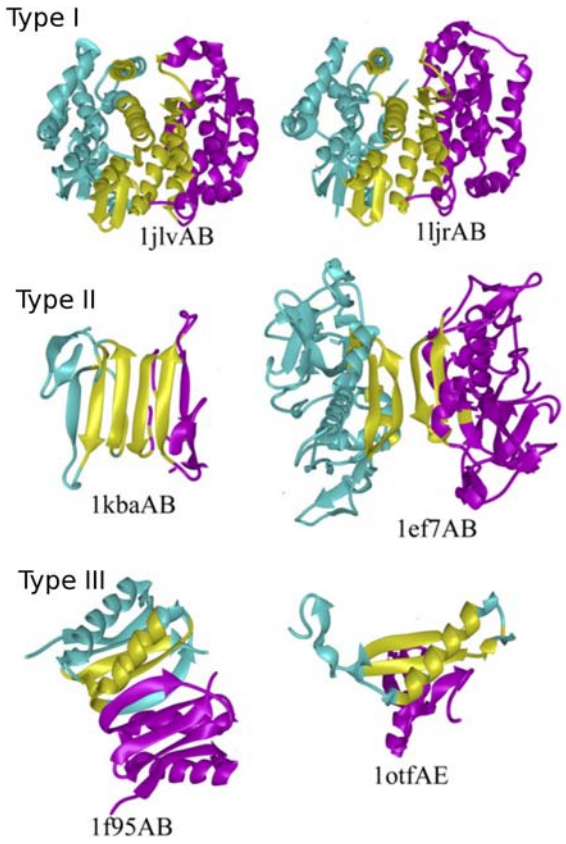


Figure 4.1: This figure illustrates the three types of interface clusters of Keskin *et al.* (2004). Type I and Type II clusters contain pairs of proteins that are homologous. Type I members have similar interfaces but Type II members have dissimilar interfaces. Type III clusters contain pairs of proteins that are non-homologous but have similar faces (one side of the interface). Figure reproduced from Keskin and Nussinov (2007).

4.1.6 The PPiClust Approach for Clustering Protein-Protein Interfaces

The PPiClust method of Aung *et al.* (2008) does not describe PPIs in terms of DDIs. Instead, protein-protein interfaces are clustered to identify distinct interfaces. Their approach consists of constructing a matrix of Euclidean distances between the Cα atoms of each interface ($\leq$5Å). The 2D matrices are then cut into submatrices and representative submatrices are reduced to 1D feature vectors using a frequency-based approach. The 1D feature vectors are then clustered using nearest neighbour with the cosine distance function. The clusters obtained are then validated using the silhouette approach[20] which provides a visual analysis of how statistically significant the clusters are.

[20] Rousseeuw (1987) described a method for assessing the optimal number of clusters and the memberships of clusters. Their approach is known as the 'silhouette' approach.

Their approach groups 2,634 interfaces into 1,716 clusters, 1,301 of which are singleton clusters. 260 clusters contain two interfaces each and 79 clusters contains three interfaces each. They also found that protein complexes with different structural folds often have similar interfaces.

4.2 Previous Studies of Protein-Protein Interaction Modes

Several groups have studied the diversity of protein interaction modes. For example, Aloy and Russell (2003) showed that homologous pairs of proteins sharing 30-40% sequence similarity tend to interact in the same way. Other studies have found that the locations of protein interaction/binding sites are often conserved, especially within domain families, regardless of the structures of their binding partners (Korkin *et al.*, 2005). Additionally, it has also been observed that many protein families employ only one or a small number of binding sites (Shoemaker *et al.*, 2006), suggesting that the same surface patch is often re-used. Indeed, it has been demonstrated previously that the structure of an unknown protein complex may often be successfully modelled using the known binding sites of homologous domains. In the context of protein docking, previous work include, e.g. Korkin *et al.* (2006), Günther *et al.* (2007), Launay and Simonson (2008), Kundrotas *et al.* (2008). These approaches are discussed further in Chapter 6. Here, we describe briefly the methods and results of Aloy and Russell (2003), Korkin *et al.* (2005), and Shoemaker *et al.* (2006).

4.2.1 Aloy's Analysis of Interaction Modes Between Domain Families

Aloy and Russell (2003) investigated whether homologous pairs of domains (members of pairs of SCOP/Pfam domain families) interact in the same way. To compare the binding modes of two DDIs, A-B and A'-B', where A and A' belong to the same SCOP/Pfam domain family, they superpose the complexes firstly, with respect to A, and secondly with respect to B. For each superposed complex, they calculate for each constituent protein, its centre of mass and 6 nearest coordinates (giving a total of 14 coordinates for a complex). They then calculate what they call an "interaction RMSD" (iRMSD) between two sets of 14 coordinates to measure the similarity between two interaction modes. They define a pair of interations to be similar if they have an iRMSD of $\leq$5Å. Their dataset contains 29,915 DDIs which are derived from 62 interacting pairs of SCOP folds. Aloy and Russell found that pairs of proteins sharing 30-40% sequence similarity tend to interact in the same way.

4.2.2 Korkin's Analysis of Binding Sites Within SCOP Families

Korkin *et al.* (2005) analysed the SCOP domain binding sites in their PIBASE database (Davis *et al.*, 2005). They define a "localization index" which measures the degree of overlap between binding sites observed for a given domain family. This index is calculated from a structure-based alignment procedure. Korkin *et al.* found that 72% of the 1,847 SCOP domain families in PIBASE have binding sites with localization values greater than expected by chance – 30% of which are statistically significant. They also found that only 8% of SCOP families have significantly low localization index, i.e. the binding sites are not conserved within a family. Hence, the results of Korkin *et al.* suggest that most domain families often have conserved binding regions.

4.2.3 Shoemaker's Analysis of Interaction Modes Between Domain Family Pairs

Shoemaker *et al.* (2006) used the CDD domain definition (Marchler-Bauer *et al.*, 2009) to study the interaction modes between domain families. They defined a conserved binding mode (CBM) in which different members of interacting domain families interact in a similar way. Shoemaker *et al.* calculate the similarity between two interaction modes using a structure-based alignment approach. Using their approach, 34,095 DDIs representing 1,798 pairs of interacting domain families, are grouped according to their interface similarity using a single-linkage hierarchical clustering procedure. They found that the 1,798 pairs of interacting domain families (derived from 34,095 DDIs) have a total 6,250 binding modes, of which 1,416 are CBMs. Moreover, they found that 833 out of 1,798 pairs ($\sim$46%) of domain families have one CBM, suggesting that the same surface and interaction mode between similar pairs of domains are reused. Shoemaker *et al.* found that few domain family pairs exhibit more than 12 CBMs (see Figure 1 of Shoemaker*et al.*, 2006).

4.3 How Large is the Space of Interface Types?

The studies described above have shown that binding sites and interfaces are often reused, especially within domain families (Section 4.2). However, beyond listing the residues observed at the interface between a given pair of proteins or protein domains, there is no generally accepted way to define what actually constitutes a protein binding site or to quantify whether or not two binding sites are structurally similar. Although the previous work of e.g. Keskin *et al.* (2004), Shulman-Peleg *et al.* (2005) and Aung *et al.* (2008) bring out an interesting observation that protein complexes with different structural folds often have similar interfaces, their approaches nevertheless give remarkably large numbers of different interface types. On the other hand, PIBASE (Davis and Sali, 2005), SCOPPI (Winter *et al.*, 2006), and 3DID (Stein *et al.*, 2009) also classify domain-domain interfaces, and their classifications also give high number of interface types (Section 4.1). This raises very important questions. Does the PDB contain really so many different interface types? Or do previous approaches seriously overestimate the number of biologically distinct interfaces. Recently, Gao and Skolnick (2010) estimate that the number of distinct interface types is roughly 1,000. This number is much smaller than what previous studies found. According to them, 89% of known interfaces have a close structural neighbour with similar backbone Cα geometry and interface pattern. Clearly, the number of distinct interface types depends on the data set, the descriptors, and the method used. It seems rather strange that different investigators can arrive at such dramatically different estimates for the total number of possible interfaces.

4.4 Reusing Protein Interface or Binding Site Information

From the studies described above, there are three important observations: (i) homologous pairs of proteins often interact in the same way, (ii) binding sites within domain families are often conserved, and (iii) non-homologous pairs of proteins can have similar interfaces. However, in template-based modelling of protein-protein complexes, the third observation has not been exploited because it is difficult to find a template interface when there is no sequence or structural homology because most PPI/DDI databases are only searchable by sequences/domains.

Clearly, when there exists a homologous PPI (a "full" template PPI) in the database, from the principle of homology, one can model directly a target PPI as previous results have shown that similar pairs of proteins often interact in the same way. Case 1 in Figure 4.2 illustrates the idea of a full template PPI. A full template PPI involves the same domains as the target PPI. However, it is often the case that there does not exist a full template PPI. Given that binding sites are often re-used irrespective of their binding partners (Section 4.1), if no full template PPI exists, it seems reasonable to reuse the binding sites of the individual target proteins (Case 2; Figure 4.2). For example, if there exists a template PPI involving one of the target protein, we can reasonably assume that the binding site in the template protein may be re-used in the target. Hence, docking calculations may be restricted to only these binding regions.

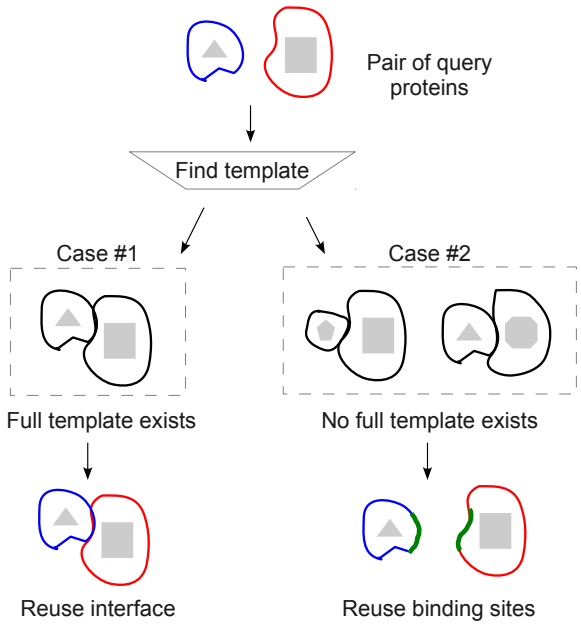


Figure 4.2: This figure illustrates the two main cases in template-based modelling of protein-protein complexes. When a PPI template involving both query proteins exists (full PPI template), the target complex can be modelled directly from the template PPI. When there is no full PPI template, known binding sites on the individual proteins may be used to guide protein docking, for example. Here, different domains are represented in different shapes in gray to illustrate homology.

However, from the point of view of protein docking, it is not straightforward to use current PPI databases (e.g. SCOPPI, 3DID, PIBASE) to retrieve key interface information and more so binding site information to guide docking calculations. As discussed in Chapter 1, human expertise is needed to gather information from several different resources and to process the database query output to extract potentially useful information to guide docking. For example, PIBASE gives very large number of face types (30,975 face types for a total of 1,946 SCOP domain families). Here again, this raises a question. Do SCOP families really have so many binding sites? According to SCOPPI, 83% of SCOP families have up to four faces, which seems more reasonable. However, SCOPPI does not exploit these face types directly but instead chooses to focus on interface types. Hence, current databases does not provide an easy way to facilitate the reuse of binding sites in domain families.

The objective is to define and describe systematically protein binding sites in domain families in order to facilitate the reuse of both binding site and interface information in protein docking. Since protein docking is inherently a spatial problem (with six degrees of freedom in the simplest rigid body assumption), we wish to consider the relative spatial arrangements of binding sites in protein domain families. Previous approaches have classified protein-protein interfaces only. It seems clear from the above discussion that we should be able to work with binding sites as well as interfaces. This means we need to develop a simple but robust 3D method to compare and cluster individual binding sites without requiring detailed information about the binding partners. Our KBDOCK database provides the framework for this experimental study.

4.5 Classifying Domain Binding Sites in KBDOCK

4.5.1 Defining a Domain Binding Site Vector

Since we chose to focus on binding sites in protein domain families, it follows that we need to find a way of analysing domain-domain interactions with their associated binding sites in protein domain families. Domain instances of a Pfam domain family which are involved in DDIs have different groups of face/binding site residues. Here, we wish to devise a way of grouping these residues into one or more binding sites by exploiting the core and rim residue information in KBDOCK (Section 3.5.2; Figure 3.5). From the augmented sequence alignments in Figure 3.4, it is clear that we cannot distinguish easily the number of "spatial" binding sites because sequence residues are not necessary neighbours in 3D space. Therefore, we need an approximate but robust way to define the spatial location of binding sites in a Pfam domain family.

In order to group automatically DDI instances which share a common binding site in 3D space, it is essential to design a descriptor which encodes the spatial position of each binding site. Our DDI superpositions described in Section 3.9.2 provides a straight-forward and natural way to define such a variable and allow us to exploit the core and rim information in KBDOCK. In this section, we introduce the notion of a "domain binding site direction vector" as illustrated in Figure 4.3. We define a binding site vector, $\underline{V}$, pointing from the centre of mass, $\underline{D}$, of the domain to the centre of its binding site, $\underline{C}$ (Figure 4.3). An approximate centre of a particular instance of a binding site, $\underline{C}$, can be obtained by calculating a weighted average of the corresponding core (weight 75%) and rim (weight 25%) C_α coordinates. The centre of mass, $\underline{D}$, is calculated from all the atoms in the domain. Domain binding site vector, $\underline{V} = (\underline{C} - \underline{D})/|\underline{C} - \underline{D}|$. We calculate and store $\underline{D}$, $\underline{C}$, and $\underline{V}$ for every superposed DDI in KBDOCK (*Oriented_DDI* in Figure 3.9).

4.5.2 Spatial Clustering of Domain Binding Site Vectors

Similar binding site vectors correspond to similar spatial positions of binding sites. To illustrate this idea, Figure 4.4 shows the superposition of three complexes with respect to a given Pfam domain family (shown in blue). For each complex, we retrieve from KBDOCK its binding site vector (v1, v2 and v3 in Figure 4.4). Since the domains belong to the same family, their centre of mass is likely to be the similar. Hence, the centre of mass can be used as a reference point. In this figure,

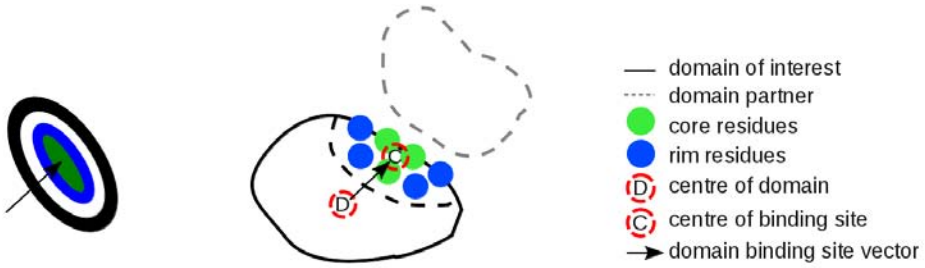


Figure 4.3: A binding site direction vector starts from the centre of mass of the domain to the centre of binding site. The centre of binding site is calculated using a weighted average of 75% core and 25% rim residues. The binding site vector can be likened to a target arrow (shown on the left).

vector v1 and v2 are similar, thus corresponding to a similar binding site position. On the other hand, v3 points in a very different direction, thus corresponding to a different binding site.

In order to group automatically DDIs with spatially similar binding sites for a given Pfam domain family, we cluster the dimensionless binding site vectors using Ward's hierarchical agglomerative clustering algorithm (Ward, 1963). The Euclidean distance measure was used (Figure 4.5). This is illustrated in Figure 4.4. We manually inspect the clusters obtained for several Pfam domain families using a TCL script to load structures automatically using a specific molecule representation in VMD. We find that a clustering distance threshold of 0.4 delineates quite well the distinct spatial positions of the binding sites. In some cases, this cut-off overestimates the number of binding sites. Overlapping binding sites are sometimes counted as distinct. Because KBDOCK uses a high sequence similarity threshold (99%; Section 3.4.3), it retains many similar structures. However, since our approach is not based on frequency, this does not give a biased grouping.

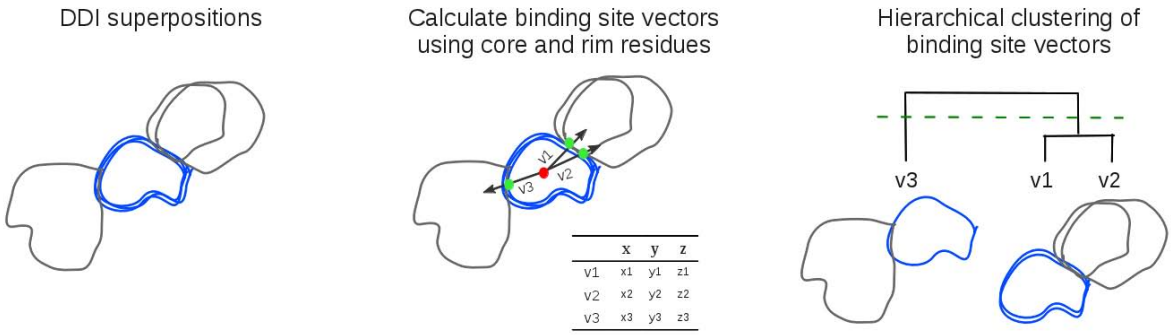


Figure 4.4: For every Pfam domain family in KBDOCK, the superposed DDIs are retrieved and for each superposed DDI, a binding site direction vector is calculated and stored in KBDOCK (see Figure 4.3). For each family, the vectors are retrieved and clustered using a hierarchical clustering algorithm.

Here, we chose a hierarchical clustering algorithm because the resulting dendrogram provides a convenient way to decide what cut-off value to use to define an optimal number of clusters. There exist several different hierarchical clustering algorithms namely, average linkage, single linkage, complete linkage, and Ward's. These algorithms give more or less a similar clustering with differences only in the cluster boundaries. We decided to use Ward's clustering over other hierarchical cluster-

ing because Ward's method is known to produce compact clusters.[21] We use the R packages "dist" and "hclust" to cluster the binding site vectors.[22] Since the clustering algorithm uses the distance between the end point of two vectors and since we normalised the binding site vectors, this is equivalent to clustering the angle between those vectors (Figure 4.5). Hence, in our case the Euclidean distance and the cosine similarity are equivalent.

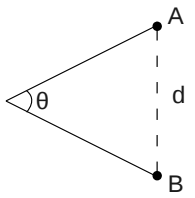


Figure 4.5: The Euclidean distance and the cosine similarity are equivalent when comparing two unit-length vectors. Here, we consider two binding site direction vector similar if they are separated by a distance of no more than 0.4.

4.6 Defining Domain Family Binding Sites

Superposing families of related DDIs in a common coordinate frame and clustering their binding site direction vectors provides a straight-forward way to analyse structural relationships between the members of a given Pfam domain. For example, Figure 4.6 shows the superpositions and binding site vectors calculated for the five DDIs involving the *Kunitz legume* Pfam family. Here, PDB 1avw (porcine trypsin / soybean trypsin inhibitor) and PDB 2qyi (bovine trypsin / trypsin inhibitor) complexes share a common binding site, and clearly have very similar domain binding site vectors. This figure clearly shows that this domain has four distinct interaction sites, one of which is common to two different trypsin/inhibitor complexes.

Figure 4.7 shows the clustering dendrogram obtained for the *Kunitz legume* family. Ward's clustering gives four clusters for this family. Here, a spatial cluster defines what we called a "domain family binding site" (DFBS). The first cluster corresponds to two different *Kunitz legume*/*Trypsin* DDIs namely PDB codes 1avw and 2qyi. As the pair-wise superposition shows, these two DDIs clearly have the same binding site positions. As the figure shows, each of the three remaining clusters represents a distinct binding site. For example, the *Kunitz legume*/*Alpha-amylase* (PDB 1ava), *Kunitz legume*/*Peptidase S8* (PDB 3bx1) and *Kunitz legume*/*Thioredoxin* (PDB 2iwt) DDIs interact via distinct binding sites. Hence, the notion of binding site direction vector provides a simple and effective way to identify distinct binding sites in a given Pfam domain family.

In KBDOCK, domain interactions are annotated with a DFBS using the following name convention: *f*/*b* where *f* is the Pfam AC of the domain and *b* is the b'th cluster or DFBS for that domain family. For example, the *Kunitz legume* domain family (PF00197) has four DFBSs PF00197/1,

[21] Ward's clustering minimizes the increase in the variance in distances when merging two clusters. Internal variance is computed as the sum of distances between each sample in the group and the group's centroid.

[22] This description of R dist and hclust packages can be found at http://stat.ethz.ch/R-manual/

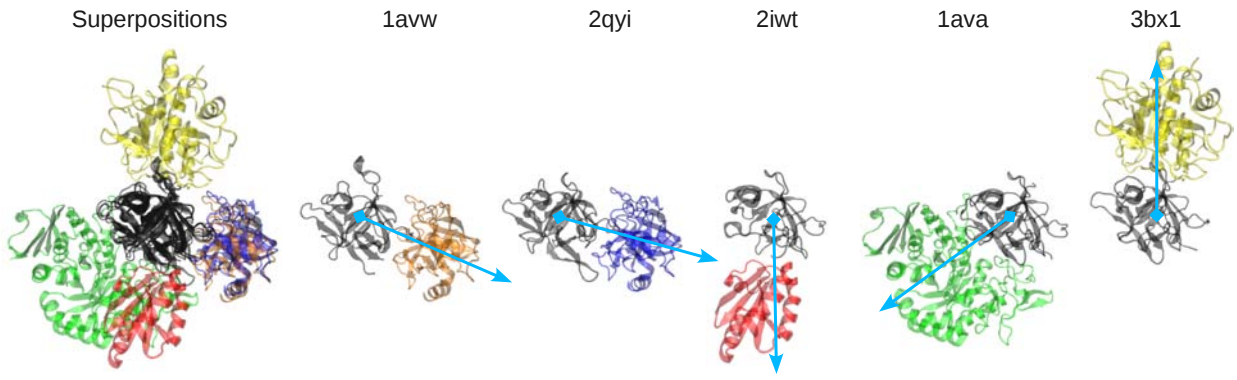


Figure 4.6: This figure shows the superpositions and domain binding site direction vectors of the five DDIs of the *Kunitz legume* Pfam family.

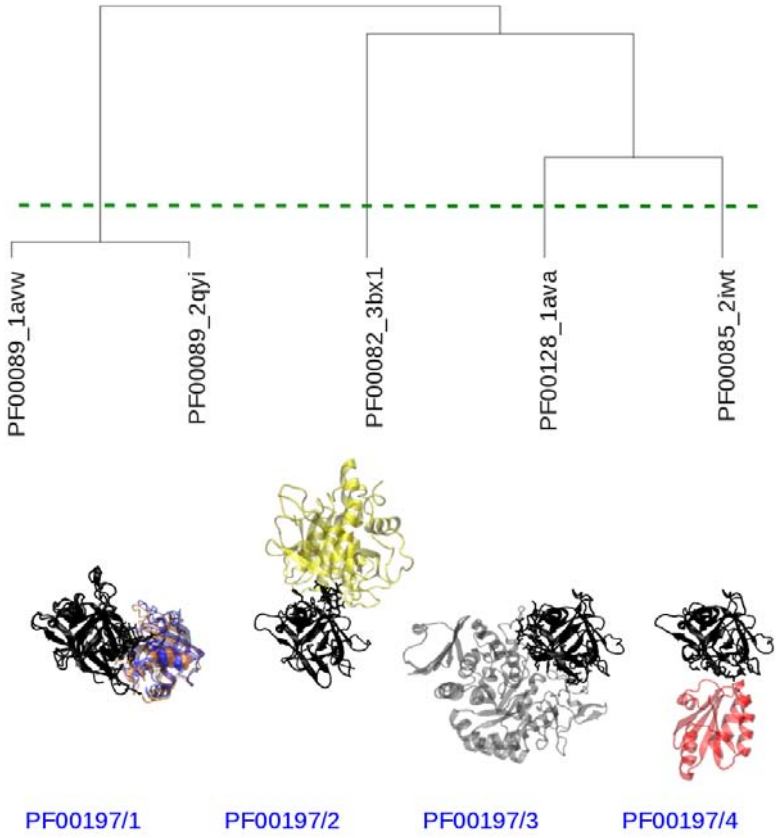


Figure 4.7: This figure shows the dendrogram obtained when clustering domain binding site vectors of the *Kunitz legume* domain family which involves 5 non-redundant hetero DDIs. Here, a cluster defines a domain family binding site. The domain of interest is shown in black. The DFBS identifiers are shown in blue.

PF00197/2, PF00197/3 and PF00197/4. DFBS PF00197/1 involves an interaction with the *Alpha amylase* domain family (PDB 1ava). Similarly, PF00197/2 interacts with *Trypsin* (PDB 1avw and 2qyi), PF00197/3 interacts with *Thioredoxin* (PDB 2iwt), and PF00197/4 interacts with *Peptidase S8* (PDB 3bx1).

Figure 4.8 shows the clustering dendrogram obtained for a further eight example Pfam families which are listed in Table 4.1. This table summarises the number of hetero DDIs and calculated number of binding sites for the ten example Pfam domain families. This table shows that these Pfam domains typically have from one to four binding sites according to our spatial clustering algorithm. In most cases, visual inspection of the superposition (Chapter 3, Figure 3.12) readily confirms the calculated number of binding sites given in Table 4.1. For example, the *Potato inhibit* domain interacts with eight other domains belonging to the *Trypsin* and *Peptidase S8* families using a single binding site. *Ribonuclease* domains interact with nine other domains belonging to the *Barstar* domain family via a single binding site. On the other hand, the *Kunitz BPTI* domain has two inhibitory binding sites, one to the "north" and one to the "south" as shown in Figure 4.8. *Kunitz BPTI* interacts with *Trypsin* and *Peptidase S7* at the "north" loop and with *Trypsin* and *Phospholip A2 1* at the "south" loop. Conversely, *Thioredoxin* domains interact with seven different Pfam families, but it does so using just two overlapping binding sites. However, for domain families which have multiple binding sites and which interact with several different domain partners (e.g. *Fer2*, *Lectin C*, *Lys*, *Actin*, and *Trypsin*), there is not a clear-cut separation of binding sites, e.g. the number of binding sites is ± 1. Moreover, in these cases, it can be difficult to distinguish all of the interactions visually. Therefore, KBDOCK allows the user to select and display only those DDIs involving a given binding site (more details in Appendix A).

Pfam ID	Pfam name	Function	No. DDIs	No. Partners	No.binding sites
PF00197	Kunitz legume	protease inhibitor	5	4	4
PF00014	Kunitz BPTI	protease inhibitor	27	3	2
PF00280	Potato inhibit	protease inhibitor	8	2	1
PF00089	Trypsin	protease	98	32	6
PF00062	Lys	hydrolase	10	4	5
PF00545	Ribonuclease	hydrolase	9	1	1
PF00022	Actin	protein binding	24	12	4
PF00059	Lectin C	glycoprotein binding	14	5	4
PF00111	Fer2	ferredoxin	14	5	3
PF00085	Thioredoxin	redox protein	8	7	2

Table 4.1: Summary of the number of DDIs, number of distinct partners (by Pfam) and calculated binding sites for ten example Pfam domains stored in KBDOCK.

It is interesting to note that even domain families which are involved in many DDIs and which have several different Pfam partners such as *Trypsin* and *Actin* still have only a small number of distinct binding sites. Compared to the previous approaches discussed in Section 4.1, this suggests that the spatial clustering scheme in KBDOCK is giving a "sharper" view of the binding regions in protein families.

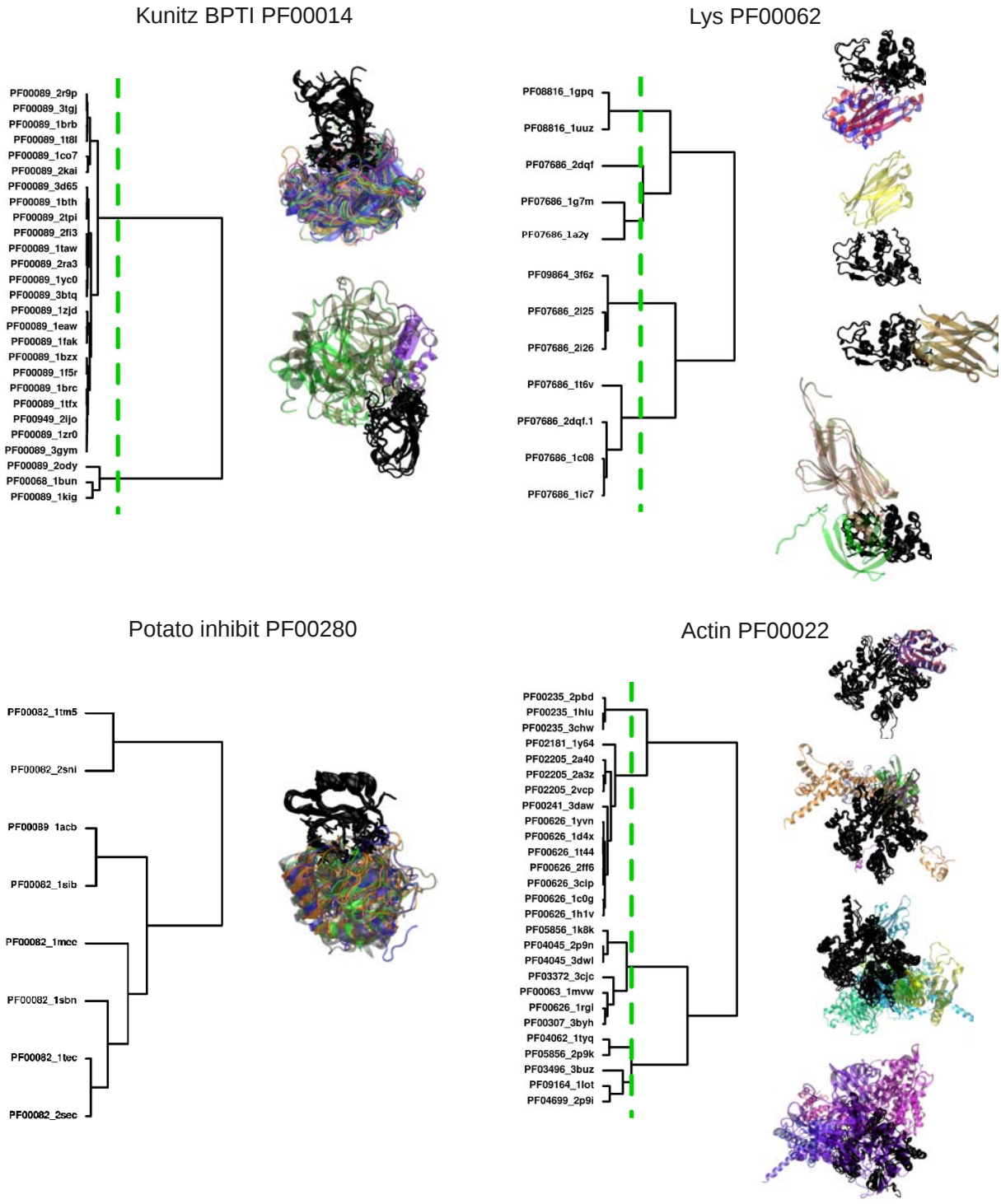


Figure 4.8: (Continued on next page).

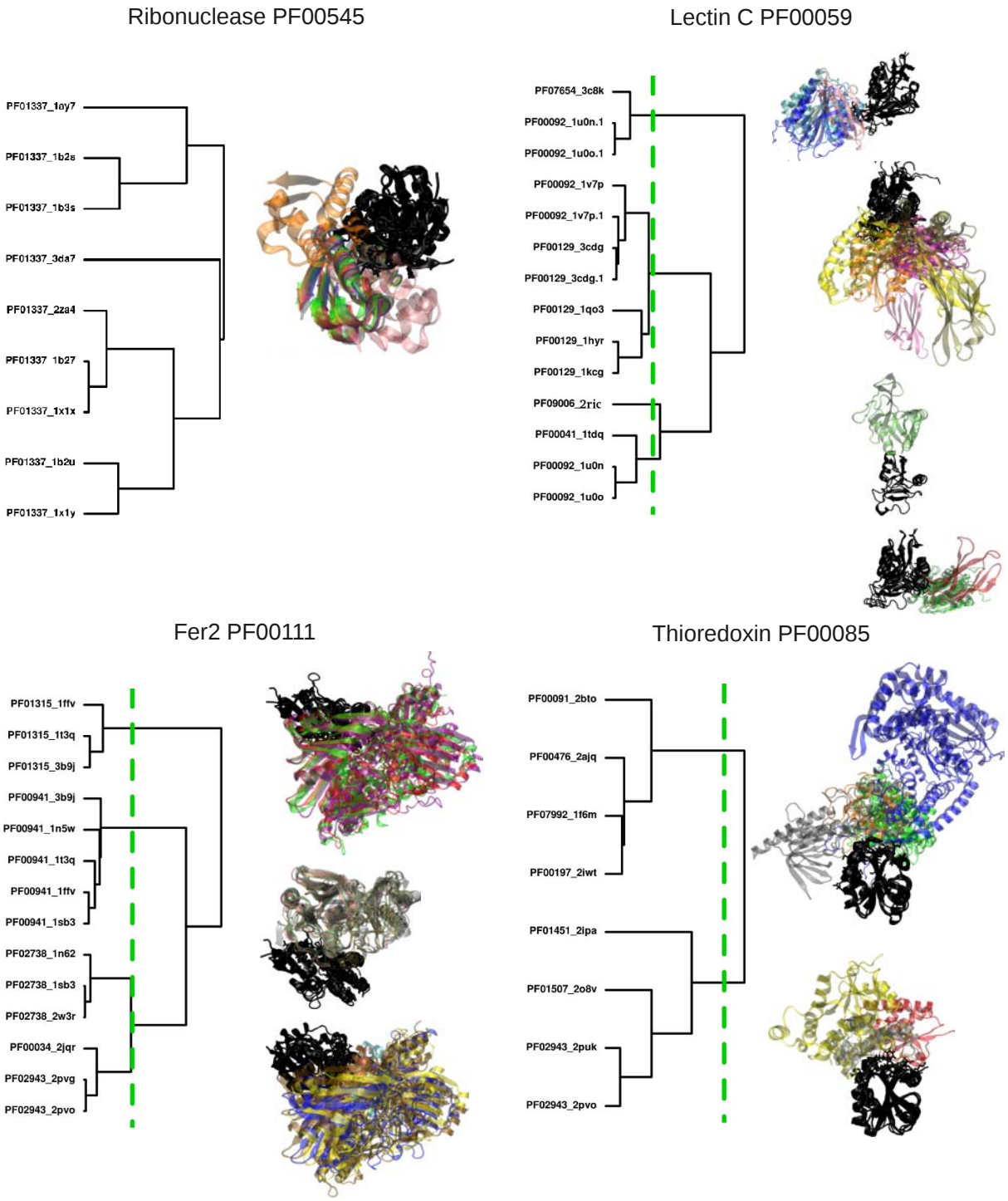


Figure 4.9: This figure shows the binding site clusters obtained by clustering the DDI binding site vectors for eight example Pfam families listed in Table 4.1. In each case, the domain of interest is shown in black.

4.7 Distribution of DFBS in Pfam Domain Families

Because KBDOCK stores DDI superpositions for all Pfam domain families which have hetero DDIs, it is easy to apply the above spatial clustering to all Pfam domain families which have more than one hetero DDI. Therefore, we calculated and stored spatial clusters in KBDOCK for all of the 1,029 Pfam domain families which are involved in hetero interactions. The entity *DFBS* in the KBDOCK relational model (Figure 3.9) represents this information. Overall, KBDOCK calculates 1,637 DFBSs for 1,035 domain families which have hetero DDIs in KBDOCK. To study the evolution of the number of distinct binding sites in time, we filter out DDIs by the PDB deposition date (1999, 2000, 2009). Figure 4.10 shows the distribution and the change with time of the number of binding sites per domain family of all NR hetero DDIs in KBDOCK. However, we exclude the very large *C1-set* immunoglobulin domain family because this family is involved in an exceptionally large number of DDIs and binding sites.

This figure confirms that most domains typically have from one to four hetero binding sites, and only a very small number of domains such as *Trypsin* (6 binding sites) have more than this. Indeed, over 65% of all hetero domains in KBDOCK have just one binding site, which supports the notion that domain binding sites are often re-used in different DDIs. It is interesting to note that despite the growing number of Pfam domains for which KBDOCK contains hetero complexes, the relative proportion of domains having 1, 2, 3, or 4 binding sites seems to be remarkably stable.

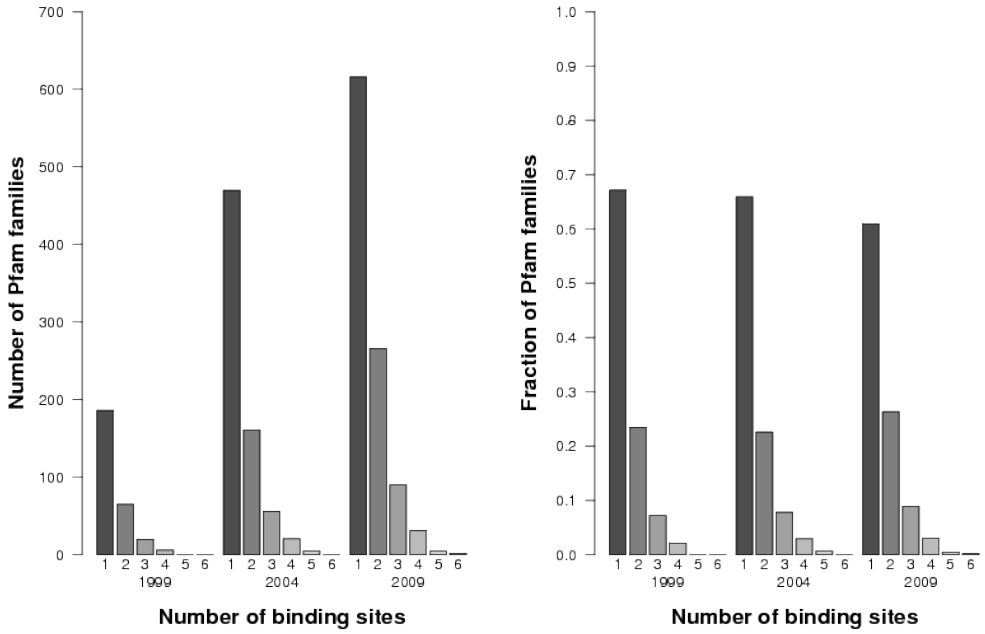


Figure 4.10: The calculated number of hetero binding sites per domain family by PDB deposition date for all Pfam families except the *C1-set* immunoglobulin domains. The total number of Pfam domains (excluding *C1-set*) for which KBDOCK has hetero DDIs are 277, 709, and 1,035 for the years 1999, 2004, and 2009, respectively.

4.8 Discussion

By superposing the structures of all hetero DDIs in Pfam domain families and by using the simple notion of a binding site direction vector to define the central region a protein binding site, KBDOCK identifies a small number of spatially distinct hetero binding sites for each Pfam domain family. This led us to conceive the notion of domain family binding sites. Here, we studied the DFBSs of hetero DDIs because hetero complexes are usually the most difficult to solve experimentally (Jones and Thornton, 1996 and Ezkurdia *et al.*, 2009). However, our approach could equally be used to study homo DDIs as well.

Compared to previous approaches which use several attributes to cluster binding sites or interfaces, our DFBS-based approach uses only knowledge of the Pfam domain and the calculated binding site direction vector associated with each domain interaction. This allows us to calculate sharper clusters of closely related binding sites compared to other approaches. For example, as described in Section 4.1, Kim *et al.* found that 34% of SCOP domain families have one binding site type whereas using KBDOCK, we found that nearly 70% of Pfam domain families have just one hetero binding site. Moreover, the PIBASE approach of Davis and Sali (2005) gives a very high number of binding site types (a total of 30,975) compared to a total of 1,439 DFBSs calculated by KBDOCK. Clearly, from the point of view of re-using binding site information, a sharper definition of binding site should be useful.

It is also interesting to note that Kim *et al.* (2006) found that although the number of interface types continues to grow, the rate of growth is currently much less than the growth in the total number of multi-domain structures that are being solved (Figure 5 of Kim et al.,2006). Our analysis of the rate of growth in the number of hetero binding sites since 1999 (Figure 5) also shows only a modest increase in the number of Pfam families having multiple hetero binding sites, despite over a three-fold increase in the number of Pfam families for which hetero complexes are now available. This strongly supports the notion that protein binding sites are very often re-used (Korkin *et al.*, 2005, Shoemaker *et al.*, 2006). Of course, the hetero complexes available in the PDB are not necessarily representative of the whole structural interactome. Nonetheless, if the very small numbers of hetero protein binding sites found here do indeed turn out to be typical, this will have considerable implications for future data-driven and template-based docking approaches, and for populating 3D PPI networks on a genomic scale.

Chapter 5

Classifying and Analysing Domain Family Binding Sites

Contents

5.1 Related Work on Protein-Protein Interface Analysis

Chapter 4 has shown how previous studies have demonstrated that homologous pairs of proteins often interact in the same way, and that proteins often interact via just one or a small number of binding sites, suggesting that binding sites are often re-used. However, these kinds of studies raise many interesting and challenging questions. For example, what features distinguish a protein binding surface from the rest of the surface? Furthermore, what features distinguish a promiscuous binding site from a selective binding site?

To try to answer such questions, several groups have studied the physicochemical properties of protein binding sites and interfaces with the aim of structurally characterizing protein-protein recognition sites. Such protein-protein recognition studies highlight or reveal useful properties which machine learning algorithms could exploit to predict protein binding sites. For recent reviews, see e.g., Moreira *et al.* (2007), de Vries and Bonvin (2008), Keskin *et al.* (2008), Ezkurdia *et al.* (2009), Patil *et al.* (2010), and Fernandez-Recio (2011). For example, Keskin *et al.* (2008) summarise the factors which have been examined by several groups in order to characterise protein binding sites and interfaces. These include amino acid residue conservation, the proportion of polar, non-polar and charged residues, the types of secondary structures present, the shape and surface area of the binding site, the number of water molecules buried on binding, the number of available hydrogen bonds and salt bridges, the strength of the interaction, and the presence of so-called "hot spot" residues for example. Here, we describe some results of several groups working on protein-protein interface characterization.

5.1.1 Various Ways of Dissecting Protein Binding Sites

Lo Conte *et al.* (1999) and Chakrabarti and Janin (2002) described the binding site in terms of core and rim residues. They defined core binding site residues as those with at least one buried binding site atom. Rim residues are binding site residues for which all atoms are accessible to solvent. On the other hand, Levy (2010) adapted the core/rim model to add another region which he called "support". He defined support residues as those which are largely buried in the unbound protein and become more buried in the complex. According to Levy's model, core binding site residues are those which are exposed in the unbound protein and become buried in the complex. Rim residues are largely exposed in the unbound form and becomes less exposed in the complex. As described in Chapter 3, KBDOCK adopts Lo Conte's core/rim definition of binding site residues.

5.1.2 Hot Spot Residues

Experimental studies based on alanine scanning mutagenesis have shown that the stability of a complex is determined by so-called hot spot residues. A hot spot residue is defined as a residue that when mutated to alanine leads to a significant drop in the binding energy. Hot spot interface residues are believed to contribute a large proportion to the total binding energy. Thorn and Bogan (2001) have collected all hot-spots residues experimentally determined by alanine scanning and made them publicly available in the ASEdb database (http://www.asedb.org). The ASEdb database

contains 2,919 single alanine mutations in protein-protein, protein-nucleic acid, and protein-small molecule complexes. Bogan and Thorn (1998) carried out a study of hot spot residues in 12 hetero protein-protein complexes. They observed that the interfaces formed an O-ring structure in which the hot spot residues are surrounded by residues which occlude solvent from the hot spots. They found that amino acids such as Trp, Arg and Tyr are the frequent hot spot residues and that Ser, Thr, Leu and Val are the rare hot spot residues. Keskin *et al.* (2005) observed that as the interface sizes increases, the number of hot spot residues increase linearly. As proposed by Bogan and Thorn (1998), Keskin *et al.* (2005) observed that hot spot residues are usually found at the core of the interface rather than the rim.

5.1.3 Hydrogen Bonds and Salt Bridges Across Interfaces

Since hydrogen bonds and salt bridges are known to contribute to the protein stability and binding energy, these have been analysed in several studies, e.g. Janin and Chothia (1990), Xu *et al.* (1997), Lo Conte *et al.* (1999), and Ofran and Rost (2003*a*). [23] The datasets of Janin and Chothia (1990) and Xu *et al.* (1997) consisted of 19 and 319 protein-protein complexes, respectively. Despite the increase in the size of the dataset, they both found that on average there are 10 hydrogen bonds in protein-protein interfaces and that the number of hydrogen bonds is strongly correlated with interface size. They also found that that there are on average 20 water molecules at interfaces, with most interfaces having less than 10 water molecules. Xu *et al.* (1997) found that there are on average 2 salt bridges at the interface. Moreover, Ofran and Rost (2003*a*) found that homo interfaces are usually depleted in salt bridges. Janin and Chothia (1990) suggested that small conformational changes occur to facilitate the close packing and hydrogen bond formation at the interface. Lo Conte *et al.* (1999) who studied 75 protein-protein complexes found that interface size is often related to the conformational change.

5.1.4 Interface Residue Composition

Janin and Chothia (1990) found that the binding sites consists on average of 34$\pm$7 Å closely-packed residues, and that interface sizes are 1600$\pm$350 Å^2. Jones and Thornton (1996) analysed 32 homo and 27 hetero complexes and found that homo interfaces which are usually permanent are more likely to be hydrophobic, and that hetero interfaces are often more planar than homo interfaces. Their study also showed that enzyme-inhibitor and permanent hetero complexes often have good shape complementarity. [24] Jones *et al.* (2000) analysed 151 intra-chain domain-domain interfaces and found that these are smaller in size due to the flexible linkers and that the interface amino acid composition is not much different from those on other surface region. Glaser *et al.* (2001) who analysed 621 representative interfaces derived from 440 3D structures found that large interfaces are often composed of hydrophobic residues while small interfaces prefer non-polar residues. Bartlett

[23] An introduction to chemical interactions in folded proteins can be found in Chapter 1 of Petsko and Dagmar (2004).

[24] Shape complementarity along with electrostatics are the two main biophysical properties used in nearly all docking methods.

et al. (2002) analysed 151 enzyme catalytic sites and observed that 65% of catalytic residues are charged (H, R, K, E, D) amino acid residues. They also found that 85% of 151 enzymes have $\geq$50% of their catalytic residues located in the three largest clefts.

Chakrabarti and Janin (2002) analysed 70 protein-protein complexes and found that while the rim interface region has no particular distinctive feature, the core interface possesses hydrophobic residues and has a preference for aromatic residues such as Trp, Tyr and Phe. Guharoy and Chakrabarti (2005) analysed 122 homo and 70 hetero protein complexes. They uses sequence entropies derived from multiple sequence alignments to analyze the conservation of core and rim residues at interfaces in homo and hetero complexes and in crystal contacts. Moreover, Guharoy and Chakrabarti analysed the interfaces formed by true interactions and crystallization. They found that core residues are more conserved than rim residues in true interfaces and that there is no significant difference in sequence entropies of core and rim residues in crystal contact interfaces. Hence, they suggest that core/rim sequence entropy ratio may be used to distinguish biological from crystallographic interfaces. Levy (2010) also confirmed that hydrophobic amino acids are preferred at the protein interior, and that charged amino acids are preferred at the surface. He found that on average interface core, rim, and support regions have similar numbers of residues, with core residues contributing over two thirds of the contact surface. Levy also found that the amino acid composition of the rim is nearly identical with that of the surface, that of the support is identical to that of the interior, and that interface core composition is intermediate between the surface and the interior.

5.1.5 Interface Residue-Residue Contacts

Several groups have studied the preferred residue-residue contacts at protein-protein interfaces, e.g. Glaser *et al.* (2001), Ofran and Rost (2003*a*), and De *et al.* (2005). For example, Glaser *et al.* found that residue-residue contacts between pairs of large hydrophobic residues such as Trp and Leu are the most favoured, and that contacts between pairs of small residues, such as Gly and Ala are the least favoured. Ofran and Rost assembled a dataset of 1,812 3D structures in which no pair of proteins had more than 25% sequence identity. The 1,812 structures involved 6 types of interfaces among which are permanent/transient homo/hetero protein-protein interfaces and intra-chain domain-domain interfaces. They found that the six types of interfaces have significantly different amino acid residue composition and residue-residue contacts. For example, they found that tryptophan is underrepresented in homo interfaces and that transient homo interfaces are very likely to involve contacts between identical amino acid residues. Given 1000 residues or 1000 residue-residue contacts, Ofran *et al.* could predict with $\geq$63% accuracy to which of the six types an interface belong. The same group developed a sequence-only interaction site predictor called ISI (Ofran and Rost, 2003*b*, 2007*b*). De *et al.* (2005) analysed 82 obligate and 30 non-obligate complexes and found that obligate and non-obligate interfaces have different residue-residue contacts e.g. obligate interfaces are mainly non-polar and have on average 20$\pm$14 contacts while non-obligate interfaces have 13$\pm$6 contacts. Caffrey *et al.* (2004) found that transient interfaces usually are smaller in size than permanent interfaces.

5.1.6 Conservation of Amino Acid Residues at Interfaces

Ma *et al.* (2003) analyzed surface and interface residues using multiple structure alignments and they found that structurally conserved residues in protein families may indicate a binding site, in particular if these residues are Trp, Phe and Met. Keskin *et al.* (2005) found that the core interface is highly conserved and is surrounded by a moderately conserved rim region. In addition, Hu *et al.* (2000) suggested that polar hot spot residues are highly conserved. Caffrey *et al.* (2004) found that in general residues at protein interfaces are usually more conserved than other surface residues, particularly in enzyme/inhibitor complexes. Active sites residues are known to be more conserved than any other exposed residues to retain their biological function (Bartlett *et al.*, 2002). On the other hand, antibody interfaces are not conserved because they always need to interact with new foreign antigens. Choi *et al.* (2009) found that most proteins have a more conserved interface than the rest of the surface and that permanent interfaces are more conserved than transient interfaces. Residues on large protein interfaces are inclined to be more conserved than residues on small interfaces (Choi *et al.*, 2009 and references therein). Choi *et al.* suggested that interface conservation may be used to discriminate between near-native and incorrect docking predictions. Although several groups agree that interface residues are likely to be conserved, using a surface patch analysis, Caffrey *et al.* found that amino acid conservation alone is not sufficient to predict interface patches.

5.1.7 Non-Homologous Interactions With Structurally-Similar Faces

Keskin and Nussinov (2007) analysed their three interface types (Keskin *et al.*, 2004) which we discussed in Section 4.1. Briefly, Type I clusters contain homologous protein-protein complexes with similar interfaces and functions. Type II clusters contain non-homologous complexes with similar interfaces but dissimilar functions. Type III clusters contain non-homologous complexes with similar faces (one side of the interface) but dissimilar functions. Keskin *et al.* regard Type I interfaces are specific-partner interface and Type III interfaces as “promiscuous” because structurally-similar faces/binding sites are interacting with proteins with different folds and functions (multi-partner interface). For example, Keskin *et al.* found that (i) multi-partner interfaces are smaller in size (1235 $Å^2$) than specific-partner interfaces (1967 $Å^2$), (ii) multi-partner interfaces are not as closely packed as specific-partner interfaces, (iii) multi-partner interfaces have 77% α-helix content while specific-partner interfaces have 38%, (iv) residues of single-partner interfaces are more conserved than residues of multi-partner interfaces, (v) structurally similar proteins interacting with different proteins are smaller and their interfaces are more planar.

5.1.8 Secondary Structure Preferences at Interfaces

An early study by Jones and Thornton (1996) who analysed 32 homo and 27 hetero complexes found that many interfaces have roughly equal proportions of helix, sheet, and loop residues, with some interfaces containing only one type of secondary structure, but most being mixed. Bartlett *et al.* (2002) analysed 151 enzyme catalytic sites and observed that 50% of catalytic residues are involved in loops. De *et al.* (2005) found that (i) non-obligatory interfaces are more likely to be involved in irregular secondary structures, (ii) obligate interfaces have a preference for β-β contacts and (iii) non-obligatory interfaces are depleted of beta-sheets. More recently, Guharoy and Chakrabarti (2007)

carried out an analysis of the content of secondary structures in obligate homo and non-obligate hetero protein-protein complexes. Their dataset consists of 122 homo and 204 hetero complexes. They defined a secondary structure segment (SSS) as a continuous segment of residues bounded by interface residues. They used DSSP to calculate secondary structures and they grouped the eight secondary structure types into 3 groups: helix (H), strand (S) and non-regular (NR) region. They defined 4 secondary structure interface types namely α, β, mixed ($\alpha\beta$) and NR. α interfaces contain $\geq$40% interface residues in helix and $<$10% in strand. β interfaces contain $\geq$40% interface residues in strand and $<$10% in helix. Mixed interfaces contain $\geq$40% interface residues in helix and strands, with $\geq$10% in each group. NR interfaces contain $>$60% interface residues in turn, loop or other unstructured regions. According to this classification, 32% of their hetero dataset belongs to the NR type; the β content is almost the same in the homo and hetero datasets; and α and $\alpha\beta$ are more abundant in the homo dataset. According to their classification, Guharoy and Chakrabarti found that (i) homo interfaces prefer helix and strand structures; (ii) the secondary structure content does not vary with interface size in homo complexes; (iii) α and β become more abundant in larger interfaces in hetero complexes; (iv) homo interfaces are mainly composed of α–α, α–NR and NR–NR pairings of interface secondary structures; (v) α–β and β–β pairings are under-represented in both the homo and hetero interfaces.

5.1.9 Structural Analyses of Hub Proteins

As discussed in Section 2.2.2, high-throughput techniques such as Y2H and TAP-MS detect PPIs on a large scale and the resulting PPIs can be represented as a protein interaction network. Proteins which interact with several proteins with different functions are known as “hub” proteins. Hub proteins are those with a large number of interactions in a protein-protein interaction network. It is believed that hub proteins must have some specific structural features to allow them to recognize and interact with several other proteins. Hence, several groups (e.g. Higurashi *et al.*, 2008) have analysed the 3D structures of hub and non-hub proteins in order to identify features that distinguish hub from non-hub proteins. These features include the structural flexibility of the proteins, the protein surface charge, and the number of distinct binding regions on the proteins (Patil *et al.*, 2010). For example, due to structural flexibility, a protein can adopt multiple distinct conformations and thus may interact with proteins with different shapes. Loops which have usually higher flexibility than helices and strands were shown not to be a distinguishing feature of hub proteins (Higurashi *et al.*, 2008). Hub proteins can also interact with different proteins through distinct binding regions. As we saw in Figure 4.8, *Thioredoxin* interacts with seven different Pfam partners through two overlapping binding sites. While structural flexibility is likely to be involved in large proteins, small hub proteins have been shown to have highly charged surface residues (see Patil *et al.* and references therein).

Tsai *et al.* (2009) suggest that the presence of hubs in PPI network is not due to a single protein structure having many different interactions but that hub proteins are simply different forms of just one protein obtained from a single gene. They suggest that one should consider “gene products” rather than “proteins” in PPI networks. For example, they propose that cellular processes such as alternative splicing, post-translational modification and allostery effects result in different conformations or different binding specificity for the hub protein. They believe that alternative splicing in exons generates large number of isomers and each of them interacts with a protein.

5.2 Large-Scale Analysis of Protein Domain Family Binding Sites

As the previous section demonstrates that there is considerable interest in understanding how experimentally observed PPI networks can be explained in terms of 3D structural interactions (Tsai *et al.*, 2009, Patil *et al.*, 2010). However, it is not a trivial task to analyse protein-protein interfaces to identify distinctive features for each interface type namely intra-chain *vs* inter-chain domain interfaces; inter-chain homo *vs* hetero protein interfaces; permanent *vs* transient protein complexes; obligate *vs* non-obligate protein complexes; and hub *vs* non-hub proteins. For example, there is the need for a sufficiently diverse data set of 3D structures of protein-protein complexes (Ezkurdia *et al.*, 2009). Furthermore, there is the need to distinguish accurately between permanent and transient complexes, and obligate and non-obligate complexes. In addition, accurate distinction between crystal artifacts and true interactions is necessary. Clearly, there is a need for a non-redundant data set that contains 3D structures for all the different types of protein-protein complexes mentioned above. Results from previous interface analyses highlights the need for a well-defined terminology and methodology with which to describe, classify, and analyse the structural nature of PPIs and protein binding sites.

Overall, the results of the studies discussed above are largely congruent with one another. The general opinion is that there is no single parameter which can distinguish between binding surface and other surface patches (Zhou and Qin, 2007). For example, de Vries and Bonvin (2008) reviewed the performance of several interface predictors and they concluded that predictors who incorporate both 3D features as well as sequence-based features tend to perform better. We have seen from previous analyses of protein-protein interfaces discussed above that many of the datasets are relatively small. Thanks to our large spatial classification of annotated hetero DDIs (Chapter 3 and 4), we are now in a position to carry out a large study of domain family binding sites (DFBS). In practice, KBDOCK allows us to analyse most of the features used by previous studies (Section 5.1). For example, amino acid composition, residue-residue contacts, secondary structures, non-covalent interactions, and hub/non-hub interactions. Here, we focus on analysing the secondary structure features of domain family binding sites.

5.3 KBDOCK Provides a Large Dataset for Statistical Analyses

In Chapter 3, we described how we built the KBDOCK database from three primary data sources namely, the 3DID database for DDI information, the Pfam domain family classification, and the PDB. KBDOCK superposes and spatially clusters a set of non-redundant hetero DDIs in order to identify a small number of DFBSs for each Pfam domain (Chapter 4). The non-redundant set of DDIs was obtained using a high sequence similarity cut-off of 99% in order to retain as many DDIs as possible since highly similar DDIs may still interact in different ways. For example, the double-headed arrowhead protease inhibitor API-A interacts with two trypsins via distinct binding sites (PDB 3e8l).

To achieve a robust classification and reliable statistics, KBDOCK filters its DDI instances involving each DFBS using a 60% sequence similarity threshold in order to retain only distinct pairs of domains associated with any given DFBS. For example, 3DID has 23 DDIs for the *Kunitz legume*

domain family which KBDOCK reduces to 5 non-redundant hetero DDIs (sequence threshold of 90%), and which it then clusters spatially to identify 4 DFBSs on the Kunitz legume domain. From Figure 4.7 one can see that one DFBS *Kunitz legume* is common to two very similar DDIs. Hence, in this case, the 60% filter reduces these two DDI instances to one DDI instance.

In contrast to previous approaches which analysed instances of protein interfaces individually, here we describe interactions at the level of Pfam domain family. Since Pfam domain families may involve one or more DFBSs, interactions are described in terms of DFBS-DFBS pairs. Here, we consider a total of 1,439 Pfam DFBSs located on 947 different Pfam domain families, and which are involved in a total of 1,009 DFBS-DFBS interactions or simply domain family interactions (DFIs). In KBDOCK, a DFBS is denoted as *f*/*b* which means DFBS *b* on domain family *f* where *f* is the Pfam AC of the domain and *b* is the b'th DFBS for that domain family. For example, as shown in Figure 4.7, the *Kunitz legume* domain family (PF00197) has four DFBSs namely PF00197/1, PF00197/2, PF00197/3 and PF00197/4. Each DFBS may interact with one or more DFBSs located on a partner domain family. For example, PF00197/1 interacts with an *alpha-amylase* DFBS (PF00128/1) Similarly, PF00197/2 interacts with a *Trypsin* DFBS, PF00197/3 interacts with a *Thioredoxin* DFBS, and PF00197/4 interacts with a *Peptidase S8* DFBS.

5.4 Annotating DFBSs with Secondary Structure Information

As described in Section 3.5.3, KBDOCK annotates domain and DFBS residues with secondary structural information using the DSSP program (Kabsch and Sander, 1983). DSSP defines eight types of SSE: α-helix (H), 3/10-helix (G), π-helix (I), residue in isolated β-bridge (B), extended strand (E), hydrogen bonded turn (T), bend (S), and loop/irregular (L). However, because several of these types are broadly quite similar, and because only a few instances of turns and bends are found in the KBDOCK database, we group the eight DSSP types into three main SSE classes which we denote here as α (H, G and I), β (B and E), and γ (T, S, and L). We then calculate the SSE propensity, $P_{f,b}(s)$, of each DFBS defined by the domain family f and its DFBS b for each SSE class, s, as the average of the DSSP frequencies in the corresponding member domain binding sites:

$$P_{f,b}(s) = \frac{1}{M} \sum_{m=1}^{M} \frac{N^s_{m,b}}{N^\alpha_{m,b} + N^\beta_{m,b} + N^\gamma_{m,b}}, \tag{5.1}$$

where M is the total number of non-redundant DDIs involving Pfam family f and DFBS b. $N^s_{m,b}$ is the count of the number of residues of type s at DFBS b of the m^{th} DDI member of family f. Each SSE propensity value calculated in this way is automatically normalised to fall within the range $[0, 1]$. That is, $P_{f,b}(\alpha) + P_{f,b}(\beta) + P_{f,b}(\gamma) = 1$.

5.5 Classifying and Analysing DFBSs

In order to examine whether DFBSs might exhibit any preferred combinations of secondary structures, we first used Ward's hierarchical clustering algorithm (Ward, 1963), as implemented in the R

software[25], to cluster the 1,439 DFBSs on the selected three classes of SSE. Visual inspection of the resulting dendrogram indicated that using 7 clusters would be the most parsimonious. However, because we believe hierarchical clustering is not necessarily the most robust clustering technique to use with smooth continuous functions, we next applied the expectation-maximization (EM) algorithm as implemented in the Weka data mining toolkit[26], to re-cluster the DFBSs using this number of target clusters. We then applied Weka's "JRip" propositional rule learning algorithm to generate a set of rules able to map DFBS instances to the selected clusters. In the following sections we refer to the clusters described by these rules as "DFBS SSE types." We then analysed the DFBSs and their interactions with respect to these DFBS SSE types.

5.6 Secondary Structure-Based Classification of DFBSs

Table 5.1 shows the mean and standard deviations (SDs) of the DFBS clusters obtained from EM clustering. Because these clusters are seen to describe biologically interesting combinations of SSE classes (e.g. "mainly α", etc.), and because each cluster has a broadly similar number of members, we adopted these clusters as a useful classification of the secondary structural composition of DFBSs. Visual inspection of Table 5.1 suggests that these clusters may be labeled as "α" (mainly α), "α + γ" (approximately equal α and γ with almost no β), "β + γ" (mainly β plus some γ), "γ + α" (mainly γ plus some α), "γ + β" (mainly γ plus some β), "γ" (nearly all γ), and "α + β + γ" (approximately equal α, β, and γ). It is interesting to note that there is no specific "α + β" DFBS SSE type in this classification. Although binding sites containing both α and β SSEs are observed quite frequently (cluster 7, 161 instances), they always contain a considerable fraction of γ SSEs (average 34.6%). Indeed, Table 5.1 shows that each of the DFBS SSE types contains a significant γ component.

Cluster	1	2	3	4
No. DFBSs	261	207	258	118
$P(\alpha)$	80.1 ± 11.5	53.7 ± 5.7	29.6 ± 8.2	4.5 ± 6.5
$P(\beta)$	0.0 ± 0.0	0.4 ± 1.2	5.9 ± 6.8	61.7 ± 13.6
$P(\gamma)$	19.3 ± 11.5	46.0 ± 5.8	64.5 ± 8.7	33.8 ± 13.9
DFBS SS Type	α	$\alpha + \gamma$	$\gamma + \alpha$	$\beta + \gamma$

Cluster	5	6	7
No. DFBSs	209	225	161
$P(\alpha)$	3.8 ± 5.4	4.3 ± 6.2	41.1 ± 16.6
$P(\beta)$	30.3 ± 9.4	2.6 ± 4.4	24.2 ± 13.2
$P(\gamma)$	65.9 ± 9.7	93.0 ± 7.4	34.6 ± 12.9
DFBS SS Type	$\gamma + \beta$	γ	$\alpha + \beta + \gamma$

Table 5.1: Mean and SDs (per cent units) of the SSE propensities for the seven DFBS clusters obtained using EM clustering.

Figure 5.1 shows how our dataset of 1,439 DFBSs is distributed over the seven DFBS SSE types.

[25] http://www.r-project.org
[26] http://www.cs.waikato.ac.nz/ml/weka/

This figure confirms that helix and irregular SSEs are the most common types of SSE in domain binding sites. It is worth noting that despite the fact that a "mainly α" DFBS requires a considerably higher proportion of α SSEs than the proportion of β SSEs in a "β + γ" DFBS, Figure 5.1 still shows that the most common type of DFBS are those that involve α SSEs.

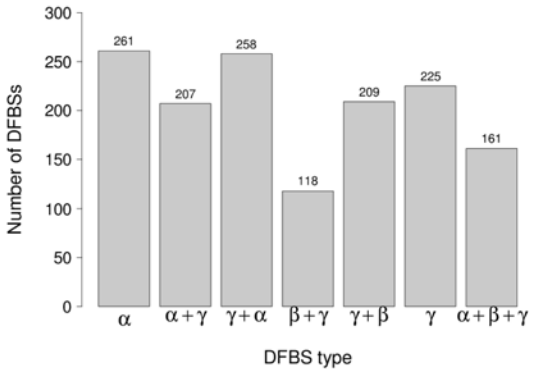


Figure 5.1: The distribution of DFBSs over the seven DFBS SSE types.

Although Table 5.1 shows the observed SSE propensities, it does not provide a convenient way to classify a new instance. We therefore used the JRip algorithm using a 10-fold cross validation to generate a set of rules able to map DFBS instances to the selected clusters. These rules are able to classify correctly 96% of the DFBS instances. However, because some of these rules are rather complex, they were manually simplified by rounding each threshold to the nearest 5%. Table 5.2 shows the simplified rules obtained in this way.

Binding Site SSE Propensity Rule	DFBS SS Type
$P(\alpha) >= 70$	α
$P(\beta) >= 45$	β + γ
$P(\beta) >= 20$ & $P(\alpha) <= 15$	γ + β
$P(\gamma) >= 80$	γ
$P(\alpha) >= 45$ & $P(\gamma) >= 35$	α + γ
$P(\alpha) >= 20$ & $P(\gamma) >= 55$	γ + α
Otherwise	α + β + γ

Table 5.2: Simplified relationships between binding site SSE propensities (per cent units) and assigned DFBS SS types.

5.7 Do DFIs Have SSE Pairing Preferences?

Figure 5.2 shows some examples of DDIs involving various associations of DFBS types. With seven DFBS SSE types, there are $7 * 6/2 + 7 = 28$ possible pairs of DFBS SSE types. Table 5.3 lists the numbers of occurrences of DFIs for each pair of DFBS SSE types (total 1,009 DFIs). To determine the significance of these numbers, we applied a standard confidence interval statistical test (95% significance level) to compare the observed frequencies of occurences with what would be

expected from a random distribution (Devore, 2008). The values in Table 5.3 which are significantly different from random (i.e. 21 out of 28) according to this test are shown in bold. Table 5.3 shows that the most frequent types of DFIs consist of interactions between identical DFBS SSE types except for $\gamma \leftrightarrow \gamma$. For example, $\alpha \leftrightarrow \alpha$ (62/1009), $\gamma + \alpha \leftrightarrow \gamma + \alpha$ (59/1009), $\gamma + \beta \leftrightarrow \gamma + \beta$ (57/1009) are frequent DFIs. Also frequent are DFIs $\gamma + \alpha \leftrightarrow \gamma$ (64), $\gamma + \alpha \leftrightarrow \alpha$ (47), and $\gamma + \alpha \leftrightarrow \alpha + \gamma$ (55). As might be expected from Table 5.1, Table 5.3 shows that the γ SSE is present in all frequent associations of DFBS SSE types.

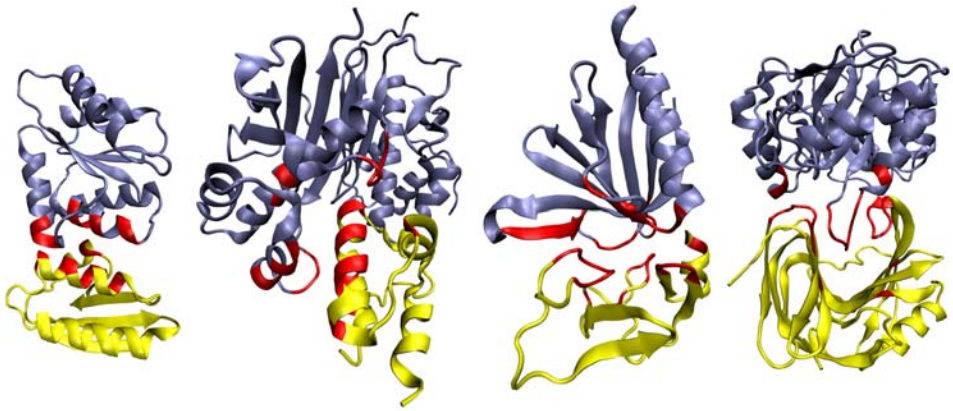


Figure 5.2: Examples of different types of DFI: (a) α with α (PDB code 1VRC, chains A, C); (b) $\alpha + \gamma$ with α (PDB code 2CG5, chains A, B); (c) $\beta + \gamma$ with γ (PDB code 2YIB, chains F, B); (d) $\gamma + \alpha$ with $\gamma + \beta$ (PDB code 1TE1, chains A, B). Binding site SSEs are shown in red.

	α	$\alpha + \gamma$	$\beta + \gamma$	$\gamma + \alpha$	$\gamma + \beta$	γ	$\alpha + \beta + \gamma$
α	**62**	53	**21**	**47**	**33**	**32**	**22**
$\alpha + \gamma$		**30**	**24**	**55**	**22**	33	26
$\beta + \gamma$			**15**	**18**	**32**	**27**	**20**
$\gamma + \alpha$				**59**	**51**	**64**	25
$\gamma + \beta$					**57**	58	35
γ						34	**21**
$\alpha + \beta + \gamma$							**33**

Table 5.3: The numbers of DFIs *observed* for each pair of DFBS SS types. The total number of DFIs is 1,009. Numbers of occurences which differ significantly from what would be expected in a random distribution are shown in bold.

In order to compare these frequencies more readily, Table 5.4 shows the marginal probabilities derived from Table 5.3. Here again, statistically significant probabilities are shown in bold. For example, if a given DFBS has been classified as α type, Table 5.4 shows that the probability that any partner of that domain will also have a mainly α binding site is 23% (statistically significant). On the other hand, the probability that any partner of that domain will have at least some α SSEs is $23 + 19.7 + 17.4 + 8.2 = 68.3\%$. Similarly, if a given DFBS has been classified as $\beta + \gamma$, then the probability of observing $\beta + \gamma$ is only 9.6%. More generally, this table shows that interactions between pairs of α-rich DFBSs and also those between pairs of γ-rich DFBSs are quite probable,

whereas α-β and β-β interactions are somewhat rare.

	Partner						
Query	α	$\alpha+\gamma$	$\beta+\gamma$	$\gamma+\alpha$	$\gamma+\beta$	γ	$\alpha+\beta+\gamma$
α	**23.0**	19.7	**7.8**	**17.4**	12.3	11.6	8.2
$\alpha+\gamma$	21.8	12.3	**9.9**	**22.6**	9.1	13.6	10.7
$\beta+\gamma$	13.4	15.3	9.6	11.5	20.4	17.2	12.6
$\gamma+\alpha$	14.7	17.2	**5.6**	**18.5**	16.0	20.1	7.9
$\gamma+\beta$	**11.5**	**7.6**	**11.1**	**17.7**	**19.8**	20.1	12.2
γ	11.9	12.3	**10.0**	**23.8**	21.6	12.6	7.8
$\alpha+\beta+\gamma$	12.1	14.3	**11.0**	13.7	19.2	11.5	**18.2**

Table 5.4: The marginal probabilities (per cent units) of observing each type of "partner" binding site for a given "query" binding site. Numbers of occurences which differ significantly from what would be expected in a random distribution are shown in bold.

5.8 Are Binding Site Surfaces Special?

In a similar manner, we investigated whether the mean SSE propensity of a domain binding site is different from the SSE propensity of the rest of the domain's accessible surface. Table 5.5 shows the marginal probabilities regarding the prediction of a domain's binding site given knowledge of the SSE character of the rest of the domain's surface (statistically significant probabilities in bold). Comparing the main diagonal and off-diagonal elements of this table suggests that the type of SSE in the binding site is well correlated with the SSE type of the domain's surface as a whole. In other words, there is often little or no difference between the SSE character of a domain's binding site and that of the rest of the domain's surface. For example, a surface that is β + γ is also likely to have a β + γ binding site (36.0%). On the other hand, it is interesting to note that a surface of γ + β is likely to have a β + γ binding site (34.1%). In other words, this suggests that the β part of a γ + β domain is rather likely to appear in the binding site. Conversely, β + γ surfaces are likely to have a γ binding site (22.7%), i.e. a binding site that *does not* have a β component.

	Binding Site						
Surface	α	$\alpha+\gamma$	$\beta+\gamma$	$\gamma+\alpha$	$\gamma+\beta$	γ	$\alpha+\beta+\gamma$
α	**52.0**	**23.6**	**2.7**	**11.5**	0.7	**4.7**	**4.7**
$\alpha+\gamma$	**41.3**	**26.9**	**1.9**	**18.1**	**0.0**	**8.8**	**3.1**
$\beta+\gamma$	**0.4**	**0.4**	**36.0**	**11.3**	**20.2**	**22.7**	**8.9**
$\gamma+\alpha$	11.7	**19.2**	**9.2**	25.2	**5.2**	**17.7**	**11.7**
$\gamma+\beta$	12.2	7.3	**34.1**	**4.9**	**19.5**	**17.1**	**4.9**
γ	3.2	0.0	3.2	22.6	3.2	**67.7**	**0.0**
$\alpha+\beta+\gamma$	**15.6**	11.7	14.8	**18.0**	**9.0**	**11.9**	**19.0**

Table 5.5: The marginal probabilities (per cent units) of observing a particular type of binding site with respect to the SSE type of the rest of the domain's surface. Statistically significant probabilities are shown in bold.

5.9 Are Multi-Partner Binding Sites Special?

We also assessed whether there are any significant differences between single partner binding sites and binding sites that interact with more than one domain. Figure 5.3 shows the distribution of the number of distinct Pfam partners for both the 947 Pfam domain families and the 1,439 DFBSs stored in the KBDOCK database. This figure shows that some 62% (584 out of 947) of these Pfam families interact with just one Pfam family, 21% interact with two different Pfam families, and only 17% interact with three or more different Pfam families. When considering DFBSs, the trend is even stronger, with over 80% (1,186 out of 1,439) of DFBSs involving interactions with just one Pfam family. Only 17.5% (252 out of 1,439) of DFBSs involve interactions with more than one Pfam family, and very few (in fact just 42) involve interactions with more than three different Pfam families.

In a similar manner, Table 5.6 shows the DFBS frequency and marginal probabilities of the various DFBS SSE types according to the number of their domain family partners. This table shows that for single-partner DFBSs, the observed frequencies of SSE types is almost no different from what would be expected from random (final row of Table 5.6). However, multi-partner DFBSs tend to be slightly depleted in α-containing SSEs and richer in γ-containing SSEs, although these tendencies are not especially strong.

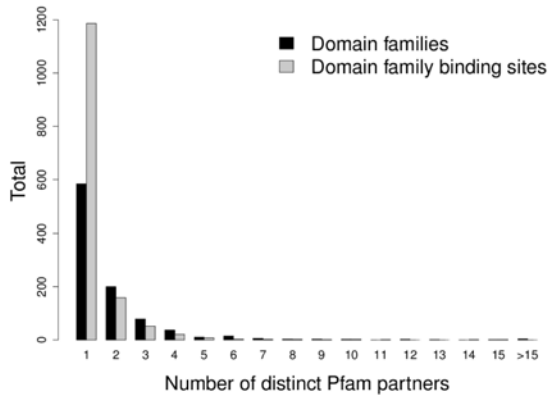


Figure 5.3: Histogram of the number of different partners by Pfam domain family and by DFBS.

DFBSs	DFIs	α	$\alpha+\gamma$	$\beta+\gamma$	$\gamma+\alpha$	$\gamma+\beta$	γ	$\alpha+\beta+\gamma$
1186	1	18.8	14.3	8.8	16.8	13.6	16.1	11.6
159	2	15.1	13.2	4.4	25.8	18.9	12.6	10.1
52	3	19.2	25.0	3.9	11.5	17.3	17.3	5.8
42	>3	9.5	7.1	11.9	28.6	21.4	9.5	9.5
Expected		18.1	14.4	8.2	17.9	14.5	15.6	11.1

Table 5.6: Marginal probability comparison (per cent units) of binding site type with respect to the number of Pfam partners. The final row shows the probabilities that would be expected from a random distribution.

Table 5.7 lists the number of distinct Pfam domain partners for the 10 Pfam domains which have the greatest numbers of DFBSs and domain partners. It is interesting to note that the Trypsin domain currently has the most interactions, presumably due to its rich variety of substrates and

because interactions involving proteases have been heavily studied as therapeutic targets. As might be expected, most of the other domains such as Ras, PKinase, ubiquitin, V-set, and C1-Set are central to cell regulation, signaling, and the immune system, for example. Thus, the identity and function of the domains listed in Table 5.7 are rather consistent with their known function and with evidence from high throughput experiments. For example, Patil *et al.* (2010) report that kinase domains are frequently observed in PPI network hubs, with some 405 hubs having some kind of kinase activity.

Pfam domain	Partners per DFBS					Total Partners
Peptidase S8	1	5				6
Cytochrome C1	1	3	5			9
Ubiquitin conjugating enzyme	1	2	2	5		10
Photosynthetic reaction centre	2	3	3	5		13
Protein kinase domain	2	4	4	5		15
Immunoglobulin C1-set	1	1	5	6	7	20
Ubiquitin	2	2	3	6	9	22
Immunoglobulin V-set	5	6	10	10		31
Ras 4	2	7	8	8	11	36
Trypsin	3	4	5	14	15	41

Table 5.7: The numbers of distinct Pfam partners for the 10 Pfam domains having the greatest numbers of DFBSs

Finally, in order to estimate whether there are any gross physical differences between DFBSs with just one binding site partner and those with more than one partner, we used DSSP to calculate the solvent accessible surface (SAS) of each DFBS (these calculations assume that each binding site contributes equally to the buried SAS at a DDI interface). Table 5.8 shows the average surface areas and number of binding site residues of DFBSs according to the number of Pfam partner domains. This table suggests that smaller DFBSs tend to have *more* interaction partners. Applying a Wilcoxon signed rank test to the difference in size between single-partner and two-partner DFBSs shows that this difference is statistically significant at the 5% level ($p = 0.014$). However, there are too few instances of three-partner DFBSs to confirm this trend statistically.

DFBSs	DFIs	Binding Site SAS (^{2})	Binding Site Residues
1186	1	709 ± 560	15 ± 11
159	2	576 ± 377	12 ± 7
52	3	559 ± 340	12 ± 7
42	> 3	670 ± 253	14 ± 6

Table 5.8: Average DFBS sizes calculated with respect to the number of their Pfam domain partners.

5.10 Discussion and Conclusion

We have shown that hetero domain binding sites can be clustered into seven main secondary structure types, each consisting of different proportions of the main secondary structure elements (SSEs), namely helices, sheets, and irregular structures. In addition, we have proposed simple rules with which to classify new instances of binding sites. Thus our structural classification of domain binding sites naturally extends the top level of the CATH domain family classification (i.e., α, β, α + β, and irregular), as originally defined by Levitt and Chothia (1976). We used our binding site classification to determine whether there are any general relationships between the SSEs of pairs of binding sites, and to investigate whether there are any differences in the structural features of domain binding sites which have just one domain partner (the majority) and those with more than one domain partner.

This classification and study was possible thanks to the well-characterized and non-redundant set of binding sites stored in KBDOCK using our spatial clustering algorithm described in Chapter 4. Because KBDOCK is built from 3DID, which we consider to be the largest and most up-to-date DDI database, our SSE-based classification of binding sites may be considered as the most comprehensive one to date. Hence, the present work represents one of the largest systematic studies of structural domain interactions to have been described to date and, to our knowledge, the first study to have considered quantitatively the nature of such interactions at the domain family level.

Our three SSE propensity "coordinates" (α, β, γ) are compatible with those of the top level of the CATH domain family classification. However, the continuous nature of these coordinates mean that it is difficult to find simple clustering rules to describe binding sites, and any derived rules may need to evolve as new DDIs are added to the structural interactome. It should also be emphasised that a CATH class does not necessarily entail structural homology. Similarly, our classification of binding sites does not imply any kind of structural homology. It aims only to provide a practical framework in which to describe different combinations of domain binding site SSEs in a way that reflects well the observed SSE propensities. Nevertheless, for present purposes, we believe that the very large coverage and the lack of redundancy of our dataset make our classification quite reliable.

Several previous studies of structural PPIs have identified potentially interesting relationships between the shapes and physical properties of protein-protein interfaces, and most such studies have suggested that it might be possible to use such properties predictively. However, these earlier studies of PPIs have been somewhat limited by the relatively small numbers of hetero protein-protein complexes available, and by the problem of how to select a suitable sub-set of protein binding sites to work with. The study described here is based on a set of 1,009 DFIs between a total of 1,439 DFBS involving 947 different Pfam domain families, representing the first systematic study of DDIs at the Pfam domain family level.

Our results confirm previous findings that α-helices are found more often at interfaces than β-sheets. More specifically, we find that α-α interactions and irregular-irregular interactions are quite probable, whereas α-β and β-β interactions are rather strongly disfavoured. On the other hand, we find there is very little difference, if any, between the SSE character of single partner binding sites and multi-partner binding sites. However, two-partner binding sites are found to have significantly smaller surface areas than single partner binding sites. Knowledge of these secondary structure

pairing propensities could be useful for the prediction of unknown DDIs, especially if combined with other near-orthogonal physical properties (de Vries and Bonvin, 2008).

Perhaps more significantly, our results show that some 60% of domain families and 80% of the DFBSs for which 3D structural DDI information is available interact with just one type of Pfam domain, and that very few DFBSs interact with more than three different Pfam domains. Interestingly, these DFBSs are always found in domains containing more than one DFBS (see Table 5.7). Although our analysis can successfully identify some known hub proteins such as the Ras and protein kinase domains, we do not see any obvious evidence at a structural level to explain the very large numbers of interactions reported by HTT PPI experiment. This suggests either that the PDB still contains example of only a very small number of the PPIs observed in HTT experiments, or that the explanation proposed by Tsai *et al.* (2009) may indeed turn out to be a more satisfactory way to explain the "hub phenomenon."

In conclusion, our structural classification of DFBSs provides a useful way to classify and analyse the secondary structure propensities of DDIs, and it highlights some SSE pairing preferences which might be useful for the prediction of unknown DDIs. We expect KBDOCK can be used in a similar way to analyse other protein interface features on a large scale. Furthermore, the next version of KBDOCK will include homo and intra DDIs and therefore comparative studies between intra, homo and hetero domain-domain interfaces can be performed using the DFBS concept.

Chapter 6

Protein-Protein Docking Using Case-Based Reasoning

Contents

6.1 Introduction

Chapter 2 reviewed some existing homology-based approaches for modelling protein-protein complexes. This chapter describes a new case-based reasoning (CBR) approach to model the structures of protein-protein complexes from the known domain-domain interactions stored in KBDOCK. Our approach is based primarily on two assumptions, namely, (i) similar pairs of domains interact in a same way, and (ii) binding sites within domain families are often conserved. Here, we apply these assumptions concretely using the notion of domain family binding sites (DFBSs) which was introduced in Chapter 4. More specifically, we assume that the interface between a new pair of proteins can be modelled using suitable instances of DFBSs in the KBDOCK database. This chapter introduces some basic principles of CBR and it then shows how these can be applied to the protein docking problem. Finally it presents the results obtained using this approach on the Protein Docking Benchmark.

6.2 Overview of Case-Based Reasoning

CBR algorithms aim to solve new problems by adapting the solutions found for similar previous cases (Kolodner, 1992). CBR is a very broadly defined method of problem-solving, and many types of CBR systems have been implemented in many different ways. For reviews on CBR and CBR systems, see Kolodner (1992), Aamodt and Plaza (1994), Watson and Marir (1994), Bergmann *et al.* (2005), and de Mántaras *et al.* (2005). Most CBR systems typically maintain a "case base" (CB) of previous cases, and they solve problems (new cases) by applying four main steps, namely, (i) retrieving the most similar case or cases from the CB, (ii) re-using or adapting those cases in order to better match the problem and to propose a solution, (iii) revising the proposed solution if necessary, and (iv) retaining the solved case in the CB for future use. The revision and retention steps are rarely performed without human intervention. These steps are illustrated in Figure 6.1. Here, we briefly describe each step.

Case Content and Representation. The success of a CBR system depends to a large extent on the structure and content of its CB (Aamodt and Plaza, 1994). Since a new case is solved by retrieving previous similar cases, the case retrieval step needs to be effective and time efficient if the CB is large. A case typically comprises a description of the problem and a description of the solution. Often, a description detailing how a solution was derived is also stored in a case (de Mántaras *et al.*, 2005). One common way of representing a case is the feature-vector approach in which each case is represented by a vector of attribute-value pairs.

Case Indexing and Retrieval. Cases are often indexed to facilitate efficient retrieval. In general, indices are ideally abstract to allow their retrieval in a wider context but at the same time indices need to be concrete in order to allow cases to be recognised for case retrieval (Watson and Marir, 1994). Indexing is usually performed on existing attributes or on derived attributes. A variety of similarity measures exist for case retrieval and their applicability depends on the case representation used.

For a feature-vector case representation, the k-nearest neighbour algorithm is commonly used for case classification and retrieval. In this representation, attributes can be assigned weights to express their importance in the retrieval step (de Mántaras *et al.*, 2005). For example, while a local similarity measure might use only a selected set of attributes, a global similarity measure could be computed as a weighted average of all the attributes.

Case Adaptation. Once a similar case is retrieved, a CBR system adapts that case to propose a solution to the new problem. The case adaptation step often depends on various aspects such as the differences between the past case and the new case, and which parts of the past case can be adapted to solve the new case (Aamodt and Plaza, 1994). There are many adaptation techniques in CBR such as reusing the past case solution without any adaptation, adjusting the parameters of the past case solution according to the parameters of the new case, and reusing the past method that constructed the solution (Aamodt and Plaza, 1994, Watson and Marir, 1994). Some CBR systems generate solutions to new problems by combining multiple past case solutions.

Case Revision and Retention. Before a new case solution can be added to the case base, the case solution needs to be evaluated. This is often done by a human expert. If the case solution proposed is not satisfactory, the solution is revised using domain-specific knowledge or by the human expert. Once a satisfactory solution is obtained, the CBR system updates its case base with the new case solution. This can be done in two ways. Either the new case is added if there has been case revision, or, the past case is generalized if needed (Aamodt and Plaza, 1994).

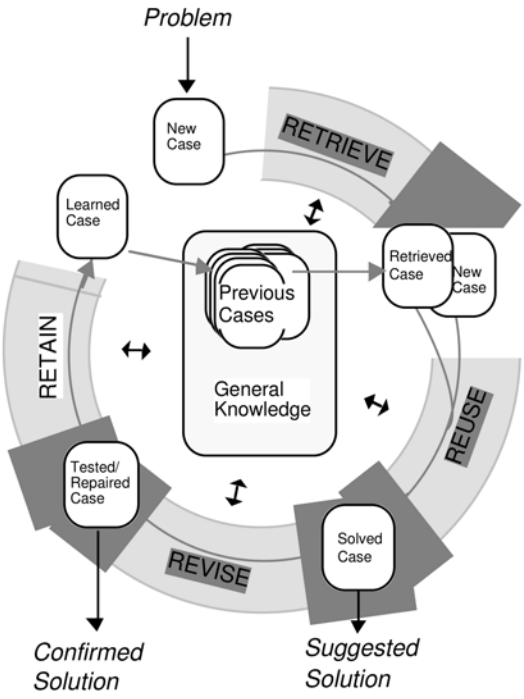


Figure 6.1: A typical CBR cycle. Figure reproduced from Aamodt and Plaza (1994).

6.3 A Formal CBR Approach to Docking By Homology

From a computational point of view, docking by homology can be considered as a kind of case-based reasoning process. Here, it might at first seem natural to build a collection of solved protein complexes to serve as the CB, and to predict the structure of a new complex by matching its component proteins to the structures in the CB. However, this would allow only a limited number of full-length single-chain complexes to be modelled. In order to be able to deal with a much wider and more diverse range of protein-protein modelling problems, we therefore treat individual domains and their associated domain-level interactions as the structural units of knowledge. From this starting point, a basic docking problem would then be expressed by querying the case base with a pair of query domains, and the CBR system would return a closely matching instance of a DDI from its case base. More generally, it seems reasonable to suppose that we can use some basic principles from CBR to develop a systematic way to exploit partial or incomplete structural DDI information in order to model both pair-wise and multi-domain protein complexes. In other words, even if a complete multi-domain protein-protein template does not exist, by applying the principles of homology all the way from sequence families to DDIs and PPIs, it should still be possible to propose 3D models of multi-domain protein complexes by reasoning about the available structural knowledge of the component domains. However, given that a protein may consist of one or more domains, and knowing that domain families may have several binding sites, it follows that multiple possible combinations of DDIs should be considered when modelling PPIs. Hence, predicting PPIs from DDIs is a non-trivial problem.

6.4 The KBDOCK Case Representation

Figure 6.2 illustrates the main steps of our CBR-based approach to protein docking. Unlike other CBR systems, we do not apply the final CBR step of storing the generated solutions in the case base in order to ensure that all of our predictions are derived only from experimentally solved and validated 3D structures. As mentioned in Section 6.2, a case is a collection of attributes or features which describe a solved problem (here, the experimentally determined structures of a pair of interacting domains). In general, each case may be described by a number of *indexed* and *non-indexed* attributes. Indexed attributes are used for case retrieval, whereas non-indexed attributes provide useful contextual information. Here, the Pfam domain identifiers of the query structures serve as the main indexed attributes, whereas the non-indexed attributes include PDB codes, PDB chain identifiers, amino acid sequences and atomic coordinates. If necessary, indexed attributes may be derived from the non-indexed attributes. For example, KBDOCK uses PfamScan (Finn *et al.*, 2010) to determine the Pfam identifiers of the problem domains from their sequences.

As illustrated in Figure 6.3, the information associated with each case includes instance-specific information such as the lists of residues of each domain which participate in a specific DDI, along with other derived instance-specific information such as the calculated geometric centre of the binding site, and the residue of each domain which KBDOCK assigns as the central, or “key”, residue of that particular binding site. KBDOCK also stores a domain family binding site (DFBS) identifier for each DDI instance (Chapter 4). Thus, instances of DDIs in the CB may be grouped and retrieved ac-

cording to both the Pfam families and the DFBSs involved. For example, the *Kunitz_legume* domain family has five non-redundant hetero DDI cases involving four distinct DFBSs. KBDOCK identifies each DFBS using a compound identifier, *PfamAC/BindingSite*. Thus, for example, PF00197/1 refers to the first DFBS of the *Kunitz_legume* family.

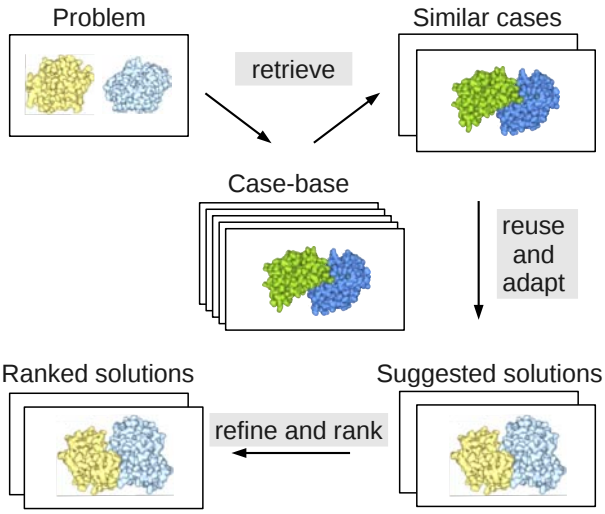


Figure 6.2: Overview of the KBDOCK CBR-inspired approach for modelling DDIs. Unlike other CBR systems, KBDOCK may propose one or more solutions and it does not retain new solutions in its case base. The figure is adapted from (Watson, 1997).

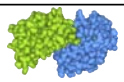
PDB	*1avw*	Structure	
Deposition date	*27-Sep-97*		
Expt. Technique	*X-ray diffraction*	Resolution	*1.75 Å*
Chain_1	*A*	Chain_2	*B*
Sequence_1	*IVGGYTCAANSI...*	Sequence_2	*DFVLDNEGNPL...*
PfamID_1	*Trypsin*	PfamID_2	*Kunitz_legume*
PfamAC_1	*PF00089*	**PfamAC_2**	*PF00197*
Region_1	*16-238*	Region_2	*502-675*
BindingSite_1	*2*	**BindingSite_2**	*1*
BS_res_1	*{Phe-502, ...}*	BS_res_2	*{His-57, ...}*
BS_centre_res_1	*Ser-195*	BS_centre_res_2	*Ser-560*
BS_centre_xyz_1	*(x, y, z)*	BS_centre_xyz_2	*(x, y, z)*

Figure 6.3: An example of a DDI case in KBDOCK. Each case consists of a collection of attributes or features. Indexed attributes which may be used for case retrieval (the Pfam accession codes and the binding site identifier) are shown in bold. For cases that match a given problem, the non-indexed attributes such as the domain sequences or the pre-computed binding site centre residues are used to guide the case adaptation and refinement steps, and to rank the proposed solutions.

6.5 The KBDOCK Case Retrieval

6.5.1 Pfam-based Case Retrieval

By denoting a pair of Pfam DFBSs as $d1/b1$ and $d2/b2$ (meaning DFBS $b1$ on domain family $d1$), we use the notation $c(d1/b1, d2/b2)$ to represent a DDI case in the CB. Similarly, by following the Prolog convention of using upper case identifiers to represent unknown or uninstantiated instances, we denote a new problem (query) as $q(d1/B1, d2/B2)$, or often just simply $q(d1, d2)$. This notation allows known binding sites to be given as part of the query if such knowledge is available, and it allows partial or incomplete matches with the CB to be represented in a consistent way. However, because the Pfam domain identifiers are the main indexed attributes, $d1$ and $d2$ are always instantiated. Naturally, the aim is to find cases which match (or, more generally, which can be *unified with*) the given query specification. If both of the query domains can be unified with cases in the CB, we call this a full-homology (FH) problem, and we denote the set of matching cases as $FH(d1, d2)$.

It is worth noting that even for the most favourable problems in which FH cases exist in the CB, the stored cases may involve more than one pair of DFBSs. For example, a recent CAPRI target concerned the complex between trypsin and arrowhead protease inhibitor A (Lensink and Wodak, 2010) which has two solutions involving two different inhibitor binding sites (PDB code 3E8L). On the other hand, it is also possible for one or both of the given query domains to match individually one half of a known DDI in the CB. We call such problems semi-homology (SH) problems, and we let $SH(d1, D2)$ and $SH(D1, d2)$, where $D1 \neq d1$ and $D2 \neq d2$, denote the two possible sets of SH cases for a given pair of query domains. Furthermore, we call a problem for which both $|SH(d1, D2)| > 0$ and $|SH(D1, d2)| > 0$ a SH-two problem, and we call a problem in which only one query domain has matching cases a SH-one problem. This distinction becomes significant at the docking refinement stage (Section 6.7). Of course, if $|FH(d1, d2)| = |SH(d1, D2)| = |SH(D1, d2)| = 0$, then no homologous cases exist, and it is necessary to adapt other more distantly related cases or to use *ab initio* docking. In the present study, we do not consider the possibility of applying adaptation beyond the Pfam level. In other words, if no FH, SH-two or SH-one are found, KBDOCK does not attempt a generalization step (such as going up a level in the Pfam hierarchy) to find further possible cases.

6.5.2 The Single-Domain Docking Test Set

In order to explore the utility of using KBDOCK to find homology templates for protein docking, we apply our approach to a subset of the protein docking targets in the Protein Docking Benchmark (Hwang *et al.*, 2010) which was described in Section 2.4.6. We selected all single domain complexes belonging to the "Enzyme-Inhibitor" (here called "Enzyme") and "Other" categories of the Docking Benchmark for this preliminary experiment. In other words, for simplicity we exclude the Benchmark "Antibody" complexes (because apart from involving the antibody hypervariable loops, antibody-antigen interactions generally do not entail homology) and we exclude all other complexes involving multiple domains. This gives a first test set of 36 Enzyme and 37 Other target complexes. For all targets, the structures of the unbound domains given by Hwang *et al.* (2010) were used as query domains, and the corresponding crystallographic complex (i.e. the expected solution) was excluded from the modelling procedure.

It should be noted that the Docking Benchmark complexes do not necessarily provide an unbiased set of homology modelling targets. For instance, because several of the Benchmark proteins have been relatively well studied, it is possible that the PDB could contain more homologues of those complexes than randomly selected complexes. In order to take into account this possible source of bias, a stringent test would be to exclude as templates all structures with more recent PDB deposition dates than the target structure. However, filtering complexes by target date (hereafter called "date filtering") often excludes a large proportion of the database. Therefore, in order to provide upper and lower bounds on the utility of template-based modelling, and to try to quantify the growing usefulness of knowledge-based approaches, we report results both with and without date filtering.

6.5.3 Coverage of FH, SH-two and SH-one Cases

Table 6.1 shows that KBDOCK retrieves FH templates for 45 out of single-domain 73 targets when no date filtering is applied. The number of SH-two and SH-one templates retrieved are 8 and 16, respectively. No templates could be found for 4 out of 73 targets. As might be expected, fewer FH templates are available when PDB date filtering is applied, and this causes an increase in the number of SH-two or SH-one templates. For example, the number of FH templates drops from 45 to 26 (out of 73). Of these 19 affected targets, 8 targets become SH-two or SH-one problems and 11 targets have no templates. Overall, applying date filtering increases the number of targets for which there are no suitable templates from 4 to 15. On the other hand, the overall number of SH-one and SH-two templates increases from 24 to 32 templates. As might be expected, the number of targets for which KBDOCK finds no template increases from 4 to 15 (out of 73). The number of targets which have FH, SH-two and SH-one templates in KBDOCK are summarised in Table 6.2. This table shows that, even with date filtering, the KBDOCK database contains a good number of candidate cases for the selected queries. This suggests that CBR-based docking could provide a useful approach to the protein docking problem.

Table 6.1 also shows that KBDOCK sometimes finds more than one distinct FH interface for a given pair of query domains. In other words, the instances of two domain families can sometimes interact *via* more than one combination of binding sites. For example, KBDOCK retrieves two FH templates for three of the Enzyme targets (1dfj, 1eaw, 2pcc) without date filtering, and for just one target (1eaw) when date filtering is applied. Similarly, KBDOCK retrieves two FH templates for two of the Other targets (1fqj, 1ml0) without date filtering, and just one target (1ml0) when date filtering is applied. Table 6.3 shows that most domains involved in the 73 benchmark targets (a total of 146 domains) have just one or two DFBSs, as was generally observed for all Pfam domain families (Chapter 4). The enzyme Trypsin has the largest number of DFBSs (five or six).

Target PDB	Dep. year	Target Class[a]	D1 Pfam name	D1 DFBSs[b]	D2 Pfam name	D2 DFBSs[b]	FH DDIs	FH DFBSs[c]
Enzymes (36)								
1avx	1997	RB	Trypsin	5 (5)	Kunitz legume	4 (1)	2 (0)	1 (0)
1ay7	1997	RB	Ribonuclease	1 (1)	Barstar	1 (1)	7 (1)	1 (1)
1cgi	1991	RB	Trypsin	6 (2)	Kazal 1	2 (1)	4 (2)	1 (1)
1d6r	1999	RB	Trypsin	6 (5)	Bowman-Birk leg	2 (1)	4 (1)	1 (1)
1dfj	1996	RB	RnaseA	3 (0)	LRR 1	3 (0)	2 (0)	2 (0)
1e6e	2000	RB	Pyr redox 2	3 (1)	Fer2	3 (2)	0 (0)	0 (0)
1eaw	2001	RB	Trypsin	6 (6)	Kunitz BPTI	2 (2)	24 (14)	2 (2)
1ezu	2000	RB	Trypsin	6 (5)	Ecotin	1 (1)	7 (2)	1 (1)
1f34	2000	RB	Asp	1 (1)	Pepsin-I3	0 (0)	0 (0)	0 (0)
1fle	1996	RB	Trypsin	6 (3)	WAP	1 (0)	1 (0)	1 (0)
1gl1	2001	RB	Trypsin	6 (6)	Pacifastin I	0 (0)	0 (0)	0 (0)
1hia	1996	RB	Trypsin	6 (3)	Antistasin	0 (0)	1 (0)	1 (0)
1jtg	2001	RB	BLIP	1 (0)	Beta-lactamase	1 (0)	3 (0)	1 (0)
1mah	1995	RB	COesterase	2 (1)	Toxin 1	1 (1)	3 (1)	1 (1)
1n8o	2002	RB	Trypsin	6 (6)	Ecotin	1 (1)	8 (5)	1 (1)
1oc0	2003	RB	Somatomedin B	0 (0)	Serpin	1 (1)	0 (0)	0 (0)
1oph	2003	RB	Trypsin	6 (6)	Serpin	2 (2)	4 (2)	1 (1)
1ppe	1991	RB	Trypsin	6 (2)	Squash	1 (0)	6 (0)	1 (0)
1r0r	2003	RB	Peptidase S8	2 (2)	Kazal 1	2 (1)	1 (0)	1 (0)
1yvb	2005	RB	Peptidase C1	2 (1)	Cystatin	1 (1)	2 (2)	1 (1)
2j0t	2006	RB	Peptidase M10	2 (2)	TIMP	2 (2)	3 (2)	1 (1)
2o8v	2006	RB	PAPS reduct	1 (1)	Thioredoxin	2 (2)	0 (0)	0 (0)
2oul	2007	RB	Peptidase C1	2 (1)	Chagasin I42	1 (1)	3 (1)	1 (1)
2pcc	1993	RB	peroxidase	2 (0)	Cytochrom C	4 (0)	4 (0)	2 (0)
2sic	1991	RB	Peptidase S8	2 (1)	SSI	1 (0)	1 (0)	1 (0)
2sni	1988	RB	Peptidase S8	2 (1)	potato inhibit	1 (1)	6 (1)	1 (1)
3sgq	1999	RB	Trypsin	6 (4)	Kazal 1	2 (1)	4 (4)	1 (1)
7cei	1998	RB	Colicin Pyocin	0 (0)	E2R135	0 (0)	0 (0)	0 (0)
1acb	1991	MD	Trypsin	5 (2)	potato inhibit	1 (1)	0 (0)	0 (0)
1nw9	2003	MD	Peptidase C14	1 (1)	BIR	1 (0)	0 (0)	0 (0)
4cpa	1982	MD	Peptidase M14	1 (0)	CarbpepA inh	0 (0)	0 (0)	0 (0)
1f6m	2000	D	Pyr redox 2	4 (0)	Thioredoxin	2 (1)	0 (0)	0 (0)
1fq1	2000	D	CDKN3	0 (0)	Pkinase	4 (3)	0 (0)	0 (0)
1pxv	2003	D	Peptidase C47	1 (0)	Staphostatin B	0 (0)	1 (0)	1 (0)
1zli	2005	D	Peptidase M14	1 (1)	Inhibitor I68	1 (0)	2 (0)	1 (0)
2o3b	2006	D	Endonuclease NS	0 (0)	NuiA	0 (0)	0 (0)	0 (0)

Table 6.1: Overall docking template results for the 73 docking benchmark targets (continued on the following page).

Target PDB	Dep. year	Target Class[a]	D1 Pfam name	D1 DFBSs[b]	D2 Pfam name	D2 DFBSs[b]	FH DDIs	FH DFBSs[c]
Others (37)								
1ak4	1997	RB	Pro isomerase	1 (0)	Gag p24	0 (0)	0 (0)	0 (0)
1buh	1998	RB	Pkinase	4 (2)	CKS	0 (0)	0 (0)	0 (0)
1e96	2000	RB	Ras	6 (3)	TPR 1	0 (0)	0 (0)	0 (0)
1efn	1996	RB	F-protein	0 (0)	SH3 1	3 (0)	0 (0)	0 (0)
1ffw	2000	RB	Response reg	3 (2)	CheY-binding	1 (1)	1 (1)	1 (1)
1fqj	2000	RB	G-alpha	2 (2)	RGS	3 (1)	7 (1)	2 (1)
1gcq	2000	RB	SH3 1	3 (1)	SH3 2	0 (0)	0 (0)	0 (0)
1gpw	2001	RB	His biosynth	1 (1)	GATase	2 (2)	1 (1)	1 (1)
1h9d	2001	RB	Runt	0 (0)	CBF beta	0 (0)	0 (0)	0 (0)
1he1	2000	RB	Ras	5 (3)	YopE	1 (1)	1 (1)	1 (1)
1j2j	2003	RB	Arf	3 (2)	GAT	1 (0)	0 (0)	0 (0)
1kac	1999	RB	Adeno knob	3 (0)	V-set	4 (1)	3 (0)	1 (0)
1ktz	2002	RB	TGF beta	2 (1)	ecTbetaR2	0 (0)	0 (0)	0 (0)
1ml0	2002	RB	M3	2 (0)	IL8	3 (0)	2 (0)	2 (0)
1s1q	2004	RB	UEV	1 (0)	ubiquitin	5 (2)	1 (0)	1 (0)
1xd3	2004	RB	Peptidase C12	1 (1)	ubiquitin	5 (3)	1 (1)	1 (1)
2a9k	2005	RB	Ras	5 (5)	Binary toxA	2 (1)	1 (1)	1 (1)
2btf	1994	RB	Actin	4 (0)	Profilin	1 (0)	3 (0)	1 (0)
2g77	2006	RB	TBC	0 (0)	Ras	5 (5)	0 (0)	0 (0)
2hle	2006	RB	Ephrin lbd	1 (1)	Ephrin	1 (1)	3 (2)	1 (1)
2oob	2007	RB	UBA	3 (2)	ubiquitin	5 (4)	5 (4)	1 (1)
1grn	1998	MD	Ras	5 (2)	RhoGAP	1 (1)	2 (2)	1 (1)
1mq8	2002	MD	ICAM N	2 (0)	VWA	4 (2)	1 (0)	1 (0)
1r6q	2003	MD	Clp N	1 (1)	ClpS	1 (1)	1 (1)	1 (1)
1syx	2004	MD	DIM1	0 (0)	GYF	0 (0)	0 (0)	0 (0)
1wq1	1997	MD	Ras	5 (2)	RasGAP	0 (0)	0 (0)	0 (0)
2ayo	2005	MD	UCH	1 (1)	ubiquitin	4 (3)	3 (1)	1 (1)
2cfh	2006	MD	TRAPP	2 (0)	TRAPP	2 (0)	0 (0)	0 (0)
2h7v	2006	MD	Ras	5 (5)	Rac1	0 (0)	0 (0)	0 (0)
2nz8	2006	MD	Ras	5 (5)	RhoGEF	2 (1)	10 (7)	1 (1)
2oza	2007	MD	Pkinase	3 (3)	Pkinase	3 (3)	0 (0)	0 (0)
2z0e	2007	MD	Peptidase C54	1 (0)	MAP1 LC3	1 (0)	1 (0)	1 (0)
3cph	2008	MD	GDI	2 (2)	Ras	5 (4)	3 (2)	1 (1)
1r8s	2003	D	Arf	3 (3)	Sec7	1 (0)	2 (0)	1 (0)
1y64	2004	D	Actin	4 (4)	FH2	0 (0)	0 (0)	0 (0)
2ido	2006	D	RNase-T	1 (1)	DNA pol3 theta	0 (0)	0 (0)	0 (0)
2ot3	2007	D	Ras	5 (5)	VPS9	1 (0)	1 (0)	1 (0)

Table 6.1: D1 denotes the first query domain; D2 denotes the second query domain. Figures in brackets show the results when PDB deposition date filtering is applied. [a]The Docking Benchmark target class – RB: Rigid Body; MD: Medium Difficulty; D: Difficult. [b] The number of DFBSs involving the query domain (excluding the target domains of the target structure). [c]The number of pairwise DFBSs calculated for full homology (FH) DDIs.

Target class	Total targets	FH templates	SH-two templates	SH-one templates	No templates
No date filtering					
Enzyme	36	24	5	5	2
Other	37	21	3	11	2
Total	73	45	8	16	4
With date filtering					
Enzyme	36	13	5	11	7
Other	37	13	1	15	8
Total	73	26	6	26	15

Table 6.2: Summary of the number of FH, SH-two, and SH-one binding site templates which KBDOCK finds for the 73 targets in our single-domain docking test set.

Target class	No. target domains	No. Homologous DFBSs in KBDOCK 0	1	2	3	4	5	6
No date filtering								
Enzyme	72	11	23	17	4	4	2	11
Other	74	15	19	10	11	6	12	1
Total	146	26	42	27	15	10	14	12
With date filtering								
Enzyme	72	25	26	10	3	1	3	4
Other	74	31	17	11	7	3	5	0
Total	146	56	43	21	10	4	8	4

Table 6.3: Distribution of the number of possible DFBS templates for the 146 individual domains of the 73 selected Protein Docking Benchmark targets.

6.6 The KBDOCK Case Adaptation

In Section 6.2, we described some different kinds of case adaptation in CBR (Aamodt and Plaza, 1994, Watson and Marir, 1994). For example, *substitution adaptation* re-instantiates parts of a previous case by applying a domain-specific *transformation* operator to map it onto the problem case. Here, we wish to apply this principle to the protein docking problem, but due to the spatial nature of this task and the fact that we may have to deal with multiple combinations of domains, we also need to take into account the possibility that not all of the retrieved DDI cases will be mutually compatible, and that similar docking problems may have *different* solutions. Hence, to model a protein complex, we need to consider multiple possible adaptations from the CB, and we need to rank them according to how well they collectively correspond to the target complex.

6.6.1 Modelling FH Problems Using Substitution Adaptation

Previous studies have shown that similar pairs of domains often interact in the same way (Aloy and Russell, 2003). Therefore, FH cases are very likely to provide good homology docking models

and they require a minimum adaptation. In Section 6.5.3, we saw that KBDOCK sometimes retrieves more than one FH case which correspond to different pairs of DFBSs. Thus, we collect the instances in $FH(d1, d2)$ into groups having distinct pairs of DBFSs, and we rank the members of each group by their overall sequence similarity to the concatenated sequences of $q(d1, d2)$. We then select the most similar member (to the query) of each group, and superpose its domains onto the query using the ProFit program in order to give a final ranked list of substitution-adapted solutions.

6.6.2 Modelling SH Problems Using Transformation Adaptation

Calculating an Initial Docking Pose Using Known DFBSs

For SH-two problems, we assume that the target complex may be modelled using a pair of existing DFBSs. We therefore take the Cartesian product $P(d1/B1, d2/B2) = SH(d1/B1) \times SH(d2/B2)$ to enumerate all possible candidate pairs of DFBSs. However, this does not form 3D interfaces between the DFBSs, it only gives a set of symbolic associations. Therefore, for each instance of P, we construct a putative DDI, $p(d1/b1, d2/b2)$, using the coordinates of the stored centre of gravity and central interface residues in order to locate the two domains on the global z axis with their central residues facing each other near the origin, and with $d1$ on the negative z axis and $d2$ on the positive z axis. This is illustrated in Figure 6.4. These transformations have been implemented using call-outs to functions from the Hex docking program (Ritchie and Kemp, 2000). Up to a small translation and an undetermined twist about the z axis, each pair of such configurations defines a putative pair-wise interface which could be refined by a rigid body docking search. However, since the aim is to find solutions for the given query, we then superpose the domains in $q(d1, d2)$ onto the oriented pair $p(d1/b1, d2/b2)$ in order to obtain a set of candidate solutions. For SH-one problems, in which one or more DFBSs are known for just one of the query domains, the query domains are oriented on the z axis as described above, using a random surface residue for the uninstantiated binding site centre residue. Here, we chose the z axis because we used the Hex docking program to perform focused docking (see next section) and Hex uses the z-axis as the intermolecular axis. Given the coordinates of the centre of binding site and centre of mass, this is sufficient for Hex to perform a focused docking search.

6.6.3 Evaluating the FH and SH Cases

This section describes the results when KBDOCK is applied to the 73 single-domain benchmark targets (Section 6.5.2). Table 6.4 lists the templates retrieved and the extent to which DFBSs are re-used for both FH and SH cases. Here, a FH template is considered to be correct if the root mean squared deviation (RMSD) between it and the native complex is less than 10Å. This is similar to the CAPRI criteria for an “acceptable” docking prediction (Mendez *et al.*, 2005). When only SH templates are retrieved, we assess their quality by comparing each proposed binding site with that of the native complex and if our binding site vector clustering algorithm would group them together, we consider the retrieved template to be correct. The final two columns of Table 6.4 show the outcome of this test. Similarly, Table 6.5 reports the corresponding results when date filtering is applied. A summary of the overall case retrieval results from a template modelling point of view is given in

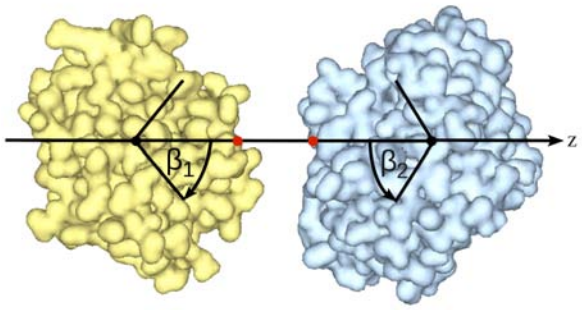


Figure 6.4: Illustration of an initial docking pose between a pair of candidate SH domains. Black spheres represent the centre of mass of each domain, and red spheres represent the central residue of each binding site. The angles β_1 and β_2 represent the Hex docking search range angles which will be applied to focus the docking search around the putative interface as will be described in Section 6.7.

Table 6.6.4.

Evaluation of FH Cases

Table 6.4 shows that the FH cases retrieved by KBDOCK provide good models of the target complex. For example, the first row of this table shows that the target 1avx can be modelled using the template 1avw with a very low of RMSD of 0.49 Å. However, in this particular case, given the similarity of the PDB file names, it is very likely that the template and the target structures were both solved at the time. The next row shows a more interesting example with sequence identities of 30.4% between the first query domain (1rgh_B) and the first template domain (1b2s_C), and 97.6% between the second query domain (1a19_B) and the second template domain (1b2s_F). In this case, a FH template with a RMSD of 2.89 Å is obtained in which the corresponding binding sites of the template domains are re-used (indicated as tick marks in the final two columns).

As described in Section 6.5.3, KBDOCK sometimes finds more than one distinct FH template for a given pair of query domains. For example, KBDOCK retrieves two FH templates for three of the Enzyme targets (1dfj, 1eaw, 2pcc) without date filtering, and for just one target (1eaw) when date filtering is applied. Subsequent visual inspection of the calculated templates for the matriptase/BPTI target (PDB code 1eaw) showed that the first trypsinogen/BPTI DDI (2r9p) provides a very good template (with an overall RMSD between the template and target of 0.79 Å), whereas the second (lower sequence identity) FH template corresponds to a different inhibitor orientation found in the prothrombin/boophilin complex (2ody; 8.54 Å RMSD). This is illustrated in Figure 6.5.

On the other hand, the two DDIs (1z7x and 2bex) calculated for the large *RnaseA/LRR 1* Enzyme complex (1dfj) were seen to overlap considerably, and the two large binding sites calculated for the *RnaseA* domain should have been clustered as a single binding site. We believe that such clustering artefacts sometimes arise due to different assignments of core and rim residues in different instances of homologous DDIs.

The peroxidase/cytochrome C Enzyme target (2pcc) is another interesting case. Although the two DDIs calculated for this target are quite distinct, further investigation revealed that one of the

DDIs (1s6v) arises because the crystal contact between these domains was larger than the biological contact in the structure. Consequently, two binding sites instead of one were also calculated for these domains.

KBDOCK retrieves two FH templates for two of the Other targets (1fqj, 1ml0) without date filtering, and just one target (1ml0) when date filtering is applied. Visual inspection of the calculated templates for the first two targets confirmed that these FH templates correspond to two different binding modes for the G-protein complex (1fqj), and the M3-protein complex (1ml0), and that the first (highest sequence identity) template best matches the target (0.70 Å and 1.03 Å RMSD, respectively). However, as discussed above, our binding site vector algorithm tends to overestimate the number of binding sites. For example, two of the Other targets (1mq8, 2ayo) are calculated to have two "new" binding sites although visual inspection again suggests these are known binding sites. The FH template proposed for these two targets gives a good model of the target complex (RMSD 2.93 and 1.76 Å, respectively). Thus, KBDOCK can successfully retrieve alternate FH binding modes when they exist in the database, but it can also be seen that its clustering algorithm has a slight tendency to overestimate the number of distinct binding sites.

Evaluation of SH Cases

Table 6.4 also shows that SH-two templates exist for 5 of the Enzyme targets (1e6e, 2ov8, 1acb, 1nw9, 1f6m). For example, the target PDB 1e6e involves a DDI between the *Pyr_redox_2* and *Fer2* domain family. Each of these domain families has three DFBSs according to our binding site vector algorithm. As illustrated in Figure 6.5, the DFBS vectors of both domains point to distinct positions. The target 1e6e reuses the DFBS of 2v3b_A (*Pyr_redox_2*) and 1ffv_D (*Fer2*). Similarly, the targets 1acb and 1f6m reuse their known DFBSs. The target 2o8v involves a DDI between *PAPS reduct* and *Thioredoxin* domain family. The target 2o8v reuses one of the known DFBSs on its *Thioredoxin* domain but does not reuse their known DBFS on the *PAPS reduct* domain. The target 1nw9 is a case where each domain does not reuse their known DFBSs. There are three Other targets (2ayo, 2cfh, 2oza) for which SH-two templates exist but none of the known DFBSs are reused in the targets. In other words, these three targets have DDI binding modes which are not known in KBDOCK.

For those targets where only SH-one templates exist, one of the known DFBSs is found to be re-used in a total of 3 out of 5 Enzyme targets (1gl1, 4cpa, 1fq1) and 5 out of 11 Other targets (1ktz, 2g77, 1wq1, 2h7v, 1y64). In order to assist any subsequent docking calculation that might use these templates, Table 6.4 and 6.5 show the proposed PDB template along with the name of the query residue calculated to be at the centre of the binding site. These residues may be used to set up a focused docking calculated as explained in the previous section.

Target PDB	Query D1[a]	Query D2[a]	Template D1[b]	%-Seq (RMSD) D1[c]	Template D2[b]	%-Seq (RMSD) D2[c]	Key Res D1[d]	Key Res D2[d]	FH RMSD[e]	DFBS D1[f]	DFBS D2[f]
Enzymes (36)											
1avx_A_B	1qqu_A	1ba7_B	1avw_A (97)	99.1 (0.40)	1avw_B (97)	99.4 (0.45)	S-195	S-60	0.49	✓	✓
1ay7_A_B	1rgh_B	1a19_B	1b2s_C (98)	30.4 (1.59)	1b2s_F (98)	97.6 (0.64)	Y-86	A-36	2.89	✓	✓
1cgi_E_I	2cga_B	1hpt_A	1cgj_E (91)	100.0 (0.69)	1cgj_I (91)	98.0 (0.67)	Y-146	C-16	1.56	✓	✓
1d6r_A_I	2tgt_A	1k9b_A	2iln_B (06)	100.0 (0.53)	2iln_I (06)	72.0 (0.90)	W-215	A-42	1.60	✓	✓
1dfj_E_I	9rsa_B	2bnh_A	1z7x_X (05)	69.4 (0.71)	1z7x_W (05)	18.9 (1.60)	N-67	S-91	1.42	✓	✓
			2bex_D (04)	33.6 (1.14)	2bex_B (04)	18.9 (1.58)	P-42	S-86	3.86	×	×
1e6e_A_B	1e1n_A	1cje_D	1y56_A (04)	10.3 (1.95)	2jqr_B (07)	97.7 (0.58)	T-87	D-76	–	×	×
			2eq9_K (07)	13.8 (1.69)	1ffv_A (00)	17.8 (1.87)	L-181	I-94	–	×	×
			2v3b_A (07)	13.6 (1.89)	1ffv_D (00)	17.8 (1.80)	V-47	V-33	–	✓	✓
1eaw_A_B	1eax_A	9pti_A	2r9p_B (07)	41.1 (0.79)	2r9p_F (07)	98.1 (0.26)	S-195	C-14	0.79	✓	✓
			2ody_D (06)	34.4 (1.03)	2ody_F (06)	45.3 (0.70)	G-216	Y-23	8.54	✓	×
1ezu_C_B	1trm_A	1ecz_A	1slu_B (96)	95.4 (0.47)	1slu_A (96)	95.3 (0.79)	W-215	S-82	0.95	✓	✓
1f34_A_B	4pep_A	1f32_A	1htr_B (94)	50.9 (1.23)	–	–	F-15	–	–	×	–
1fle_E_I	9est_A	2rel_A_4	2z7f_E (07)	38.6 (1.02)	2z7f_I (07)	51.1 (1.58)	F-215	C-23	1.47	✓	✓
1gl1_A_I	1k2i_1	1pmc_A_6	1acb_E (91)	98.6 (0.34)	–	–	M-192	–	–	✓	–
			1tab_E (90)	41.8 (0.93)	–	–	W-215	–	–	✓	–
			1l4z_A (02)	38.4 (0.98)	–	–	T-144	–	–	×	–
			1bml_B (99)	38.4 (0.83)	–	–	T-62	–	–	×	–
			1bui_B (98)	38.9 (1.12)	–	–	G-25	–	–	×	–
			1wss_H (04)	35.6 (1.01)	–	–	S-164	–	–	×	–
1hia_B_I	2pka_Y	1bx8_A	1c9p_A (99)	26.5 (1.11)	1c9p_B (99)	30.6 (1.61)	S-195	C-29	1.24	✓	✓
1jtg_B_A	3gmu_B	1zg4_A	2b5r_D (05)	100.0 (0.75)	2b5r_B (05)	98.8 (0.38)	E-73	Y-105	1.89	✓	✓
1mah_A_F	1j06_B	1fsc_A	1ku6_A (02)	99.2 (0.41)	1ku6_B (02)	82.0 (0.78)	V-73	V-34	0.76	✓	✓
1n8o_C_E	8gch_F	1ifg_A	1ezu_C (00)	24.0 (0.82)	1ezu_B (00)	96.2 (0.91)	H-57	S-82	1.31	✓	✓
1oc0_A_B	2jq8_A_4	1b3k_A	–	–	1k9o_I (01)	30.1 (1.44)	–	R-346	–	–	×
1oph_A_B	1utq_A	1qlp_A	1k9o_E (01)	71.7 (0.55)	1k9o_I (01)	28.4 (1.40)	Q-192	L-353	0.65	✓	✓
1ppe_E_I	1btp_A	1lu0_A	2btc_E (98)	100.0 (0.42)	2btc_I (98)	89.7 (0.44)	Q-192	G-29	0.44	✓	✓
1r0r_E_I	1scn_E	2gkr_I	1yu6_A (05)	100.0 (0.33)	1yu6_C (05)	100.0 (0.76)	L-126	T-17	1.05	✓	✓
1yvb_A_I	2ghu_A	1cew_I	1stf_E (93)	33.8 (1.13)	1stf_I (93)	19.8 (1.51)	G-40	L-54	2.98	✓	✓
2j0t_A_D	966c_A	1d2b_A_20	1uea_C (97)	59.0 (0.72)	1uea_D (97)	70.5 (1.28)	A-182	T-2	2.01	✓	✓
2o8v_A_B	1sur_A	2trx_A	1zun_A (05)	17.9 (1.61)	2ajq_B (05)	100.0 (0.63)	P-167	I-75	–	×	×
			–	–	2puk_C (07)	46.2 (0.97)	–	I-75	–	–	✓
2oul_A_B	3bpf_A	2nnr_A	2nqd_B (06)	36.4 (1.12)	2nqd_A (06)	96.0 (0.72)	H-174	P-30	6.31	✓	✓
2pcc_A_B	1ccp_A	1ycc_A	1u74_A (04)	100.0 (0.39)	1u74_B (04)	99.0 (0.59)	N-196	R-13	0.40	✓	✓
			1s6v_A (04)	98.3 (0.41)	1s6v_D (04)	98.0 (0.50)	K-29	K-86	15.4	×	×

Table 6.4: The calculated docking templates and key binding site residues for the 73 benchmark targets without date filtering (continued on next page)

Target PDB	Query D1[a]	Query D2[a]	Template D1[b]	%-Seq (RMSD) D1[c]	Template D2[b]	%-Seq (RMSD) D2[c]	Key Res D1[d]	Key Res D2[d]	FH RMSD[e]	DFBS D1[f]	DFBS D2[f]
Enzymes (contd)											
2sic_E_I	1sup_A	3ssi_A	3sic_E (91)	100.0 (0.27)	3sic_I (91)	98.9 (0.64)	L-126	C-71	0.18	✓	✓
2sni_E_I	1ubn_A	2ci2_I	1sbn_E (91)	99.6 (0.34)	1sbn_I (91)	35.9 (1.40)	L-126	T-58	1.70	✓	✓
3sgq_E_I	2qa9_E	2ovo_A	3sgb_E (83)	99.4 (0.54)	3sgb_I (83)	98.0 (0.43)	G-215	C-16	0.22	✓	✓
7cei_A_B	1unk_B	1m08_B	–		–	–	–	–	–	–	–
1acb_E_I	2cga_A	1egl_A	1cgi_E (91)	100.0 (0.71)	2tec_I (90)	98.4 (1.02)	R-145	V-43	–	✓	✓
			1tab_E (90)	41.8 (1.02)	–	–	W-215	–	–	✓	–
			1o5f_H (03)	38.6 (1.04)	–	–	C-122	–	–	×	–
			1l4z_A (02)	38.5 (1.01)	–	–	P-152	–	–	×	–
			1bml_A (99)	38.5 (1.02)	–	–	T-62	–	–	×	–
1nw9_B_A	1jxq_A	2opy_A	1i4e_B (01)	35.3 (0.97)	2pop_B (07)	34.8 (0.72)	R-341	Q-283	–	×	×
4cpa_A_I	8cpa_A	1h20_A_9	1pyt_B (95)	97.1 (0.40)	–	–	F-279	–	–	✓	–
1f6m_A_C	1cl0_A	2tir_A	2eq9_K (07)	23.7 (1.37)	2ajq_B (05)	99.0 (0.62)	Q-214	I-75	–	✓	✓
			2f5z_C (05)	18.8 (2.05)	2o8v_B (06)	98.1 (0.69)	E-209	C-32	–	✓	×
			2v3b_A (07)	17.3 (2.00)	–	–	W-52	–	–	×	–
			1y56_A (04)	15.0 (1.49)	–	–	R-95	–	–	×	–
1fq1_A_B	1fpz_F	1b39_A	–	–	1bi8_A (98)	13.8 (1.88)	–	N-91	–	–	×
			–	–	3gni_B (09)	13.7 (1.87)	–	H-28	–	–	×
			–	–	2qvs_E (07)	13.4 (1.81)	–	S-159	–	–	✓
			–	–	2f2c_B (05)	11.0 (1.93)	–	S-33	–	–	×
1pxv_A_C	1x9y_A	1nyc_A	1y4h_A (04)	98.8 (0.73)	1y4h_C (04)	100.0 (0.78)	H-340	S-92	0.70	✓	✓
1zli_A_B	1kwm_A	2jto_A_6	1zlh_A (05)	38.3 (0.85)	1zlh_B (05)	100.0 (1.84)	R-127	G-45	1.24	✓	✓
2o3b_A_B	1zm8_A	1j57_A	–	–	–	–	–	–	–	–	–
Others (38)											
1ak4_A_D	2cpl_A	1e6j_P	1mzw_A (02)	53.4 (0.58)	–	–	T-41	–	–	×	–
1buh_A_B	1hcl_A	1dks_A	1jsu_A (96)	87.6 (0.94)	–	–	V-79	–	–	×	–
			2iw9_A (06)	85.7 (1.01)	–	–	I-52	–	–	×	–
			2bkz_A (05)	85.2 (1.01)	–	–	R-122	–	–	×	–
			1fq1_B (00)	84.8 (1.17)	–	–	E-208	–	–	×	–
1e96_A_B	1mh1_A	1hh8_A	1g4u_R (00)	100.0 (0.78)	–	–	G-12	–	–	×	–
			2h7v_B (06)	100.0 (0.88)	–	–	R-68	–	–	×	–
			1hh4_B (00)	100.0 (0.80)	–	–	R-68	–	–	×	–
			2dfk_B (06)	71.3 (1.03)	–	–	T-125	–	–	×	–
			2v55_D (08)	45.7 (0.97)	–	–	E-171	–	–	×	–
			2c5l_A (05)	30.9 (1.19)	–	–	D-38	–	–	×	–

Table 6.4: (continued on next page)

Target PDB	Query D1[a]	Query D2[a]	Template D1[b]	%-Seq (RMSD) D1[c]	Template D2[b]	%-Seq (RMSD) D2[c]	Key Res D1[d]	Key Res D2[d]	FH RMSD[e]	DFBS D1[f]	DFBS D2[f]
Others (contd)											
1efn_B_A	1avv_A	1g83_A	–	–	1m27_C (02)	100.0 (0.54)	–	D-100	–	–	×
			–	–	1ju5_C (01)	42.9 (1.16)	–	W-139	–	–	×
			–	–	1u5s_A (04)	32.0 (1.06)	–	V-88	–	–	×
1ffw_A_B	3chy_A	1fwp_A	1a0o_E (97)	100.0 (0.54)	1a0o_F (97)	100.0 (1.48)	I-95	V-53	0.52	✓	✓
1fqj_A_B	1tnd_C	1fqi_A	1fqk_A (00)	90.3 (0.47)	1fqk_B (00)	95.7 (0.82)	T-178	N-364	0.47	✓	✓
			2bcj_Q (05)	51.3 (0.89)	2bcj_A (05)	14.8 (1.48)	H-252	A-356	17.0	×	×
1gcq_B_C	1gri_B	1gcp_B	2jt4_A (07)	40.8 (1.16)	–	–	P-49	–	–	×	–
			1m27_C (02)	35.4 (0.83)	–	–	E-16	–	–	×	–
			1u5s_A (04)	26.5 (1.31)	–	–	I-4	–	–	×	–
1gpw_A_B	1thf_D	1k9v_F	1ka9_F (01)	58.5 (0.90)	1ka9_H (01)	37.1 (1.17)	D-98	M-421	3.57	✓	✓
1h9d_A_B	1ean_A	1ilf_A_1	–	–	–	–	–	–	–	–	–
1he1_C_A	1mh1_A	1he9_A	1g4u_R (00)	100.0 (0.77)	1g4u_S (00)	30.0 (1.43)	G-12	R-146	0.98	✓	✓
1j2j_A_B[g]	1o3y_A	1oxz_A	2j59_D (06)	100.0 (0.56)	1wr6_D (04)	45.9 (1.03)	T-31	L-227	–	–	–
			1re0_A (03)	99.4 (0.98)	–	–	G-50	–	–	–	–
			3bh7_A (07)	46.0 (0.77)	–	–	A-27	–	–	–	–
1kac_A_B	1nob_F	1f5w_B	1p69_A (03)	99.4 (0.51)	1p69_B (03)	97.5 (0.68)	P-418	E-56	0.10	✓	✓
1ktz_A_B	1tgk_A	1m9z_A	2p6a_A (07)	33.1 (1.14)	–	–	V-92	–	–	✓	–
			1m4u_L (02)	31.6 (1.22)	–	–	Y-50	–	–	×	–
1ml0_A_D	1mkf_A	1dol_A	2nyz_B (06)	100.0 (0.73)	2nyz_E (06)	27.7 (0.97)	L-273	C-12	1.03	✓	✓
			2nz1_A (06)	100.0 (0.62)	2nz1_E (06)	90.8 (0.96)	L-174	I-20	24.3	×	×
1s1q_A_B	2f0r_A	1yj1_A	1uzx_A (04)	23.0 (1.14)	1uzx_B (04)	87.8 (0.66)	F-44	F-45	1.97	✓	✓
1xd3_A_B	1uch_A	1yj1_A	1cmx_A (99)	23.5 (1.42)	1cmx_B (99)	87.8 (0.66)	A-11	V-70	1.12	✓	✓
2a9k_A_B	1u90_A	2c8b_X	2a78_A (05)	99.4 (0.63)	2a78_B (05)	33.0 (0.80)	A-70	I-97	1.03	✓	✓
2btf_A_P	1ijj_B	1pne_A	1hlu_A (97)	92.5 (1.45)	1hlu_P (97)	100.0 (0.54)	C-774	N-99	0.75	✓	✓
2g77_A_B	1fkm_A	1z06_A	–	–	1ukv_Y (03)	41.3 (1.17)	–	Y-103	–	–	×
			–	–	3cue_R (08)	41.9 (1.29)	–	W-87	–	–	×
			–	–	2fu5_D (06)	36.3 (1.52)	–	R-71	–	–	×
			–	–	1wq1_R (97)	35.3 (1.25)	–	S-42	–	–	✓
			–	–	1wa5_A (04)	26.9 (1.32)	–	D-161	–	–	×
2hle_A_B	2bba_A	1iko_P	1kgy_C (01)	44.2 (1.07)	1kgy_G (01)	97.9 (0.88)	A-186	G-126	2.01	✓	✓
2oob_A_B	2ooa_A	1yj1_A	2g3q_A (06)	30.6 (0.90)	2g3q_B (06)	83.8 (0.93)	R-964	V-70	7.59	×	✓
1grn_A_B	1a4r_A	1rgp_A	2ngr_A (98)	99.4 (0.68)	2ngr_B (98)	95.3 (0.64)	Q-61	V-198	1.74	✓	✓
1mq8_A_B	1iam_A	1mq9_A	3bn3_B (07)	26.1 (1.46)	3bn3_A (07)	93.8 (0.53)	E-34	S-141	2.93	×	×
1r6q_A_C	1r6c_X	2w9r_A	1mg9_B (02)	100.0 (0.82)	1mg9_A (02)	97.6 (0.42)	T-26	V-80	0.77	✓	✓
1syx_A_B	1qgv_A	1l2z_A_1	–	–	–	–	–	–	–	–	–

Table 6.4: (continued on next page)

Target PDB	Query D1[a]	Query D2[a]	Template D1[b]	%-Seq (RMSD) D1[c]	Template D2[b]	%-Seq (RMSD) D2[c]	Key Res D1[d]	Key Res D2[d]	FH RMSD[e]	DFBS D1[f]	DFBS D2[f]
Others (contd)											
1wq1_R_G	6q21_D	1wer_A	1xd2_A (04)	99.4 (0.66)	–	–	V-44	–	–	×	–
			3brw_D (07)	57.1 (0.91)	–	–	G-60	–	–	✓	–
			3cf6_R (08)	55.6 (1.04)	–	–	D-57	–	–	×	–
			1ukv_Y (03)	37.4 (1.03)	–	–	Y-71	–	–	×	–
			2dfk_B (06)	31.4 (1.18)	–	–	A-122	–	–	×	–
2ayo_A_B	2ayn_A	2fcn_A	1nbf_A (02)	22.1 (1.35)	1nbf_D (02)	90.5 (0.68)	S-275	L-69	1.76	×	×
2cfh_A_C	1sz7_A	2bjn_A	2j3w_D (06)	95.4 (0.62)	2j3t_B (06)	53.0 (0.95)	V-116	M-107	–	×	×
			2j3t_A (06)	96.0 (0.78)	2j3w_F (06)	21.9 (1.26)	L-60	L-135	–	×	×
2h7v_A_C	1mh1_A	2h7o_A	1g4u_R (00)	100.0 (0.77)	–	–	G-12	–	–	×	–
			1hh4_B (00)	100.0 (0.80)	–	–	R-68	–	–	✓	–
			1i4d_D (01)	99.4 (0.70)	–	–	S-71	–	–	✓	–
			1e96_A (00)	98.9 (0.61)	–	–	P-29	–	–	×	–
			2v55_D (08)	45.7 (0.98)	–	–	E-171	–	–	×	–
2nz8_A_B	1mh1_A	1nty_A	2vrw_A (08)	98.9 (0.93)	2vrw_B (08)	22.4 (1.66)	T-58	Q-1368	2.84	✓	✓
2oza_B_A	3hec_A	3fyk_X	1jsu_A (96)	35.1 (1.31)	2qvs_E (07)	27.4 (1.58)	V-52	S-265	–	×	×
			1fq1_B (00)	34.8 (1.42)	2vgo_B (07)	27.6 (1.63)	H-228	V-137	–	×	×
			1w98_A (04)	34.8 (1.21)	3ddq_C (08)	20.8 (1.53)	R-70	E-104	–	×	×
2z0e_A_B	2d1i_A	1v49_A_1	2zzp_A (09)	95.7 (0.60)	2zzp_B (09)	99.1 (1.46)	S-316	F-80	0.82	✓	✓
3cph_G_A	3cpi_G	1g16_A	1ukv_G (03)	99.1 (1.34)	1ukv_Y (03)	51.9 (0.76)	R-248	Y-89	1.86	✓	✓
1r8s_A_E	1hur_A	1r8m_E	1r8q_A (03)	100.0 (1.02)	1r8q_E (03)	98.9 (1.15)	I-46	I-193	2.27	✓	✓
1y64_A_B	2fxu_A	1ux5_A	1lot_B (02)	93.4 (0.59)	–	–	M-283	–	–	×	–
			2pbd_A (07)	93.1 (1.01)	–	–	R-116	–	–	×	–
			2ff3_B (05)	92.8 (0.54)	–	–	Y-143	–	–	✓	–
			1mvw_Y (02)	92.2 (0.74)	–	–	Y-53	–	–	×	–
2ido_A_B	1j54_A	1se7_A_1	1zbu_A (05)	17.8 (1.52)	–	–	E-85	–	–	×	–
2ot3_B_A	1yzu_A	1txu_A	2efe_D (07)	39.1 (1.17)	2efe_C (07)	43.9 (1.42)	D-74	L-316	3.50	✓	✓

Table 6.4: [a] The unbound structures of the query domains, as given by Hwang *et al.*, (2010). D1 denotes the first query domain; D2 denotes the second query domain. D1 and D2 are given in Table 6.1. [b]The best template found by KBDOCK with which to model the target complex. The PDB deposition year of the template is given in parentheses. [c] The percentage sequence identity between the query and the template domain sequences. The RMSD between the C_α atoms of the target complex and the selected template is given in parentheses. [d] The residue shown is the query residue calculated to be at the centre of the interface in the target complex. [e] When a FH template is found, the unbound query structures are superposed onto the template and the overall RMSD between the superposed query domains and those of the target complex is calculated. [f] The selected template and target DFBSs are the same according to the KBDOCK DFBS clustering threshold. [g] The template DFBS could not be compared with the target in this case (1j2j) because 3DID does not provide DDIs for this structure. A hyphen indicates “no data found”.

Target PDB	Query D1[a]	Query D2[a]	Template D1[b]	%-Seq (RMSD) D1[c]	Template D2[b]	%-Seq (RMSD) D2[c]	Key Res D1[d]	Key Res D2[d]	FH RMSD[e]	DFBS D1[f]	DFBS D2[f]
Enzymes (36)											
1avx_A_B	1qqu_A	1ba7_B	1mct_A (92)	100.0 (0.46)	1ava_C (97)	30.1 (1.23)	S-195	P-168	–	✓	×
			1tab_E (90)	81.5 (0.50)	–	–	W-215	–	–	✓	–
			2hpp_H (93)	37.6 (0.96)	–	–	F-94	–	–	×	–
			1hut_H (93)	37.5 (1.05)	–	–	G-203	–	–	×	–
			1avg_H (97)	37.2 (0.46)	–	–	F-82	–	–	×	–
1ay7_A_B	1rgh_B	1a19_B	1brs_C (94)	30.4 (1.70)	1brs_F (94)	98.8 (0.49)	H-85	D-39	3.24	✓	✓
1cgi_E_I	2cga_B	1hpt_A	1tgs_Z (82)	41.8 (1.04)	1tgs_I (82)	66.0 (1.06)	Y-146	C-16	1.62	✓	✓
1d6r_A_I	2tgt_A	1k9b_A	1tab_E (90)	100.0 (0.59)	1tab_I (90)	36.0 (1.05)	W-215	C-41	1.65	✓	✓
1dfj_E_I	9rsa_B	2bnh_A	–	–	–	–	–	–	–	–	–
1e6e_A_B	1e1n_A	1cje_D	1f6m_F (00)	12.3 (1.73)	1ffv_A (00)	17.8 (1.71)	A-180	I-94	–	×	×
			–	–	1ffu_D (00)	17.8 (1.71)	–	C-46	–	–	×
1eaw_A_B	1eax_A	9pti_A	1bzx_E (98)	39.6 (0.84)	1bzx_I (98)	98.1 (0.26)	S-195	C-14	0.69	✓	✓
			1kig_H (97)	33.2 (0.99)	1kig_I (97)	24.1 (1.33)	G-219	R-42	7.60	×	×
1ezu_C_B	1trm_A	1ecz_A	1slx_B (96)	96.8 (0.50)	1slx_A (96)	95.3 (0.78)	W-215	S-82	0.97	✓	✓
1f34_A_B	4pep_A	1f32_A	1htr_B (94)	50.9 (1.23)	–	–	F-15	–	–	×	–
1fle_E_I	9est_A	2rel_A_4	1pyt_C (95)	53.8 (0.78)	–	–	F-215	–	–	✓	–
			1hrt_H (93)	30.0 (1.14)	–	–	Y-207	–	–	×	–
			2hpp_H (93)	29.1 (1.09)	–	–	W-94	–	–	×	–
1gl1_A_I	1k2i_1	1pmc_A_6	1acb_E (91)	98.6 (0.34)	–	–	M-192	–	–	✓	–
			1slv_B (96)	42.1 (1.00)	–	–	S-195	–	–	✓	–
			1bui_B (98)	38.9 (1.12)	–	–	G-25	–	–	×	–
			1bml_B (99)	38.4 (0.83)	–	–	T-62	–	–	×	–
			1j9c_H (01)	35.6 (1.00)	–	–	S-164	–	–	×	–
			1sgf_X (97)	29.5 (0.96)	–	–	G-187	–	–	×	–
1hia_B_I	2pka_Y	1bx8_A	2kai_B (84)	100.0 (0.55)	–	–	W-215	–	–	✓	–
			2hpp_H (93)	18.7 (1.06)	–	–	H-101	–	–	×	–
			1ets_H (92)	18.7 (1.09)	–	–	G-203	–	–	×	–
1jtg_B_A	3gmu_B	1zg4_A	–	–	–	–	–	–	–	–	–
1mah_A_F	1j06_B	1fsc_A	1fss_A (95)	58.9 (0.77)	1fss_B (95)	82.0 (0.79)	Y-72	V-34	0.79	✓	✓
1n8o_C_E	8gch_F	1ifg_A	1ezu_C (00)	24.0 (0.82)	1ezu_B (00)	96.2 (0.76)	H-57	S-82	1.31	✓	✓
1oc0_A_B	2jq8_A_4	1b3k_A	–	–	1k9o_I (01)	30.1 (1.44)	–	R-346	–	–	×
1oph_A_B	1utq_A	1qlp_A	1k9o_E (01)	71.7 (0.55)	1k9o_I (01)	28.4 (1.40)	Q-192	L-353	0.65	✓	✓
1ppe_E_I	1btp_A	1lu0_A	1tab_E (90)	100.0 (0.44)	–	–	W-215	–	–	✓	–
			1hgt_H (91)	36.3 (0.97)	–	–	G-203	–	–	×	–
1r0r_E_I	1scn_E	2gkr_I	1cse_E (88)	100.0 (0.30)	3sgb_I (83)	100.0 (0.46)	L-126	C-16	–	✓	✓
			1scj_A (98)	68.2 (0.51)	–	–	Y-104	–	–	×	–
1yvb_A_I	2ghu_A	1cew_I	1stf_E (93)	33.8 (1.13)	1stf_I (93)	19.8 (1.51)	G-40	I-58	2.98	✓	✓
2j0t_A_D	966c_A	1d2b_A_20	1uea_C (97)	59.0 (0.72)	1uea_D (97)	70.5 (1.28)	A-182	T-2	2.01	✓	✓
2o8v_A_B	1sur_A	2trx_A	1zun_A (05)	17.9 (1.61)	2ajq_B (05)	100.0 (0.63)	P-167	I-75	–	×	×
			–	–	2ipa_A (06)	46.2 (1.32)	–	I-72	–	–	×

Table 6.5: The calculated docking templates and key binding site residues for the 73 benchmark targets with date filtering (continued on next page).

Target PDB	Query D1[a]	Query D2[a]	Template D1[b]	%-Seq (RMSD) D1[c]	Template D2[b]	%-Seq (RMSD) D2[c]	Key Res D1[d]	Key Res D2[d]	FH RMSD[e]	DFBS D1[f]	DFBS D2[f]
Enzymes (contd)											
2oul_A_B	3bpf_A	2nnr_A	2nqd_B (06)	36.4 (1.12)	2nqd_A (06)	96.0 (0.72)	H-174	P-30	6.31	✓	✓
2pcc_A_B	1ccp_A	1ycc_A	–	–	–	–	–	–	–	–	–
2sic_E_I	1sup_A	3ssi_A	1cse_E (88)	69.0 (0.55)	–	–	L-126	–	–	✓	–
2sni_E_I	1ubn_A	2ci2_I	1cse_E (88)	68.6 (0.54)	1cse_I (88)	35.9 (1.39)	L-126	V-57	1.96	✓	✓
3sgq_E_I	2qa9_E	2ovo_A	3sgb_E (83)	99.4 (0.54)	3sgb_I (83)	98.0 (0.43)	G-215	C-16	0.57	✓	✓
7cei_A_B	1unk_B	1m08_B	–	–	–	–	–	–	–	–	–
1acb_E_I	2cga_A	1egl_A	1cgj_E (91)	100.0 (0.73)	2tec_I (90)	98.4 (1.02)	R-145	V-43	–	✓	✓
			1hgt_H (91)	35.1 (1.10)	–	–	W-207	–	–	×	–
1nw9_B_A	1jxq_A	2opy_A	1i4e_B (01)	35.3 (0.97)	–	–	R-341	–	–	×	–
4cpa_A_I	8cpa_A	1h20_A_9	–	–	–	–	–	–	–	–	–
1f6m_A_C	1cl0_A	2tir_A	–	–	1t7p_B (97)	99.0 (0.55)	–	I-75	–	–	✓
1fq1_A_B	1fpz_F	1b39_A	–	–	1bi8_C (98)	13.8 (1.72)	–	N-91	–	–	×
			–	–	1buh_A (98)	10.9 (1.77)	–	L-156	–	–	×
			–	–	1b6c_D (99)	10.8 (1.72)	–	S-69	–	–	×
1pxv_A_C	1x9y_A	1nyc_A	–	–	–	–	–	–	–	–	–
1zli_A_B	1kwm_A	2jto_A_6	2bo9_C (05)	39.0 (0.98)	–	–	Y-198	–	–	✓	–
2o3b_A_B	1zm8_A	1j57_A	–	–	–	–	–	–	–	–	–
Others (38)											
1ak4_A_D	2cpl_A	1e6j_P	–	–	–	–	–	–	–	–	–
1buh_A_B	1hcl_A	1dks_A	1fin_C (96)	85.2 (0.96)	–	–	I-52	–	–	×	–
			1jsu_A (96)	87.6 (0.94)	–	–	V-79	–	–	×	–
1e96_A_B	1mh1_A	1hh8_A	1foe_D (00)	96.6 (0.95)	–	–	D-57	–	–	×	–
			1ds6_A (00)	94.8 (0.80)	–	–	R-68	–	–	×	–
			1rrp_A (99)	27.4 (1.24)	–	–	Y-23	–	–	✓	–
1efn_B_A	1avv_A	1g83_A	–	–	–	–	–	–	–	–	–
1ffw_A_B	3chy_A	1fwp_A	1a0o_E (97)	100.0 (0.54)	1a0o_F (97)	100.0 (1.48)	I-95	V-53	0.52	✓	✓
1fqj_A_B	1tnd_C	1fqi_A	1agr_A (97)	62.9 (0.77)	1agr_E (97)	35.3 (1.16)	T-178	N-364	0.78	✓	✓
1gcq_B_C	1gri_B	1gcp_B	1ycs_B (96)	32.6 (0.85)	–	–	F-47	–	–	×	–
1gpw_A_B	1thf_D	1k9v_F	1ka9_F (01)	58.5 (0.90)	1ka9_H (01)	37.0 (1.17)	D-98	M-421	3.57	✓	✓
1h9d_A_B	1ean_A	1ilf_A_1	–	–	–	–	–	–	–	–	–
1he1_C_A	1mh1_A	1he9_A	1g4u_R (00)	100.0 (0.77)	1g4u_S (00)	30.0 (1.43)	G-12	R-146	0.98	✓	✓
1j2j_A_B[g]	1o3y_A	1oxz_A	1ksj_A (02)	45.3 (0.87)	–	–	V-53	–	–	–	–
			1m2o_B (02)	36.6 (0.97)	–	–	T-48	–	–	–	–
1kac_A_B	1nob_F	1f5w_B	–	–	1akj_E (97)	16.7 (1.37)	–	L-73	–	–	✓
1ktz_A_B	1tgk_A	1m9z_A	1es7_A (00)	30.8 (1.26)	–	–	R-52	–	–	×	–
1ml0_A_D	1mkf_A	1dol_A	–	–	–	–	–	–	–	–	–
1s1q_A_B	2f0r_A	1yj1_A	–	–	1cmx_B (99)	87.8 (0.48)	–	V-70	–	–	×
			–	–	1otr_B (03)	83.8 (1.25)	–	I-44	–	–	×
1xd3_A_B	1uch_A	1yj1_A	1cmx_A (99)	23.5 (1.42)	1cmx_B (99)	87.8 (0.66)	A-11	V-70	1.12	✓	✓
2a9k_A_B	1u90_A	2c8b_X	2a78_A (05)	99.4 (0.63)	2a78_B (05)	32.9 (0.80)	A-70	I-97	1.03	✓	✓

Table 6.5: (continued on next page).

Target PDB	Query D1[a]	Query D2[a]	Template D1[b]	%-Seq (RMSD) D1[c]	Template D2[b]	%-Seq (RMSD) D2[c]	Key Res D1[d]	Key Res D2[d]	FH RMSD[e]	DFBS D1[f]	DFBS D2[f]
Others (contd)											
2btf_A_P	1ijj_B	1pne_A	–	–	–	–	–	–	–	–	–
2g77_A_B	1fkm_A	1z06_A	–	–	1ukv_Y (03)	41.3 (1.17)	–	Y-103	–	–	×
			–	–	2d7c_A (05)	38.4 (1.11)	–	V-68	–	–	✓
			–	–	2fu5_D (06)	36.3 (1.06)	–	R-71	–	–	×
			–	–	1xd2_B (04)	35.3 (1.13)	–	D-88	–	–	✓
			–	–	1wa5_A (04)	26.9 (1.32)	–	D-161	–	–	×
2hle_A_B	2bba_A	1iko_P	1kgy_D (01)	44.2 (1.04)	1kgy_H (01)	97.9 (0.84)	A-186	G-126	2.02	✓	✓
2oob_A_B	2ooa_A	1yj1_A	2g3q_A (06)	30.6 (0.90)	2g3q_B (06)	83.8 (0.93)	R-964	V-70	7.59	×	✓
1grn_A_B	1a4r_A	1rgp_A	1am4_D (97)	94.4 (0.83)	1am4_A (97)	96.0 (0.85)	Q-61	V-198	1.74	✓	✓
1mq8_A_B	1iam_A	1mq9_A	–	–	1ijk_A (01)	21.7 (1.41)	–	K-276	–	–	×
			–	–	1m10_A (02)	21.7 (1.52)	–	Q-167	–	–	×
1r6q_A_C	1r6c_X	2w9r_A	1mg9_B (02)	100.0 (0.82)	1mg9_A (02)	97.6 (0.42)	T-26	V-80	0.77	✓	✓
1syx_A_B	1qgv_A	1l2z_A_1	–	–	–	–	–	–	–	–	–
1wq1_R_G	6q21_D	1wer_A	1gua_A (96)	57.7 (0.93)	–	–	Y-40	–	–	×	–
			1am4_D (97)	30.2 (1.25)	–	–	Q-61	–	–	✓	–
2ayo_A_B	2ayn_A	2fcn_A	1nbf_B (02)	22.1 (1.31)	1nbf_C (02)	90.5 (0.68)	S-275	L-69	1.76	×	✓
2cfh_A_C	1sz7_A	2bjn_A	–	–	–	–	–	–	–	–	–
2h7v_A_C	1mh1_A	2h7o_A	1hh4_B (00)	100.0 (0.80)	–	–	R-68	–	–	✓	–
			1g4u_R (00)	100.0 (0.77)	–	–	G-12	–	–	×	–
			1i4d_D (01)	99.4 (0.70)	–	–	S-71	–	–	✓	–
			1e96_A (00)	98.9 (0.61)	–	–	P-29	–	–	×	–
			1wa5_A (04)	27.4 (1.26)	–	–	Q-141	–	–	×	–
2nz8_A_B	1mh1_A	1nty_A	1foe_D (00)	96.6 (0.95)	1foe_C (00)	22.8 (2.01)	D-57	I-1364	3.61	✓	×
2oza_B_A	3hec_A	3fyk_X	1fq1_B (00)	34.8 (1.32)	2bfx_B (04)	27.0 (1.63)	H-228	V-137	–	×	×
			1jsu_A (96)	35.1 (1.32)	1f3m_C (00)	21.3 (1.43)	V-52	S-265	–	×	×
			1w98_A (04)	34.8 (1.21)	2g9x_C (06)	21.1 (1.50)	R-70	E-104	–	×	×
2z0e_A_B	2d1i_A	1v49_A_1	–	–	–	–	–	–	–	–	–
3cph_G_A	3cpi_G	1g16_A	1ukv_G (03)	99.1 (1.34)	1ukv_Y (03)	51.9 (0.76)	R-248	Y-89	1.86	✓	✓
1r8s_A_E	1hur_A	1r8m_E	1r4a_B (03)	55.2 (0.90)	–	–	Y-81	–	–	✓	–
			1ksh_A (02)	44.3 (0.88)	–	–	T-55	–	–	✓	–
			1m2o_D (02)	35.6 (1.05)	–	–	D-67	–	–	×	–
1y64_A_B	2fxu_A	1ux5_A	1t44_A (94)	92.8 (0.68)	–	–	Y-143	–	–	✓	–
			1rgi_A (03)	91.4 (0.55)	–	–	E-93	–	–	×	–
			1kxp_A (02)	91.4 (0.53)	–	–	L-171	–	–	×	–
			1tyq_A (04)	34.5 (1.39)	–	–	R-256	–	–	×	–
2ido_A_B	1j54_A	1se7_A_1	1zbu_A (05)	17.8 (1.52)	–	–	E-85	–	–	×	–
2ot3_B_A	1yzu_A	1txu_A	2hv8_A (06)	38.3 (1.17)	–	–	A-54	–	–	✓	–
			1ukv_Y (03)	37.0 (1.04)	–	–	Y-89	–	–	×	–
			2fu5_D (06)	34.0 (1.24)	–	–	L-57	–	–	×	–
			2a78_A (05)	32.5 (1.24)	–	–	G-77	–	–	×	–
			1wa5_A (04)	31.5 (1.27)	–	–	Q-145	–	–	×	–

Table 6.5: The column headings are described in Table 6.4.

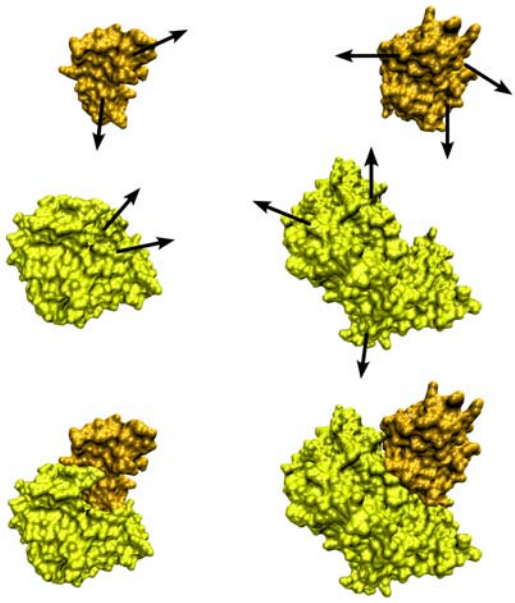


Figure 6.5: This figure illustrates the domain family binding sites of the Trypsin / Kunitz BPTI and Pyr_Redox_2 / Fer2 docking targets. The example on the left shows the trypsin (yellow) and Kunitz BPTI (orange) query domains with their binding site direction vectors (calculated from the 2r9p and 2ody FH templates) shown as arrows. The target complex (PDB code 1eaw) shown below may be modelled to within 0.79 Å RMSD by re-using the DFBS of the highest sequence identity FH template, 2rp9_B / 2rp9_F (see Table 6.4). The example on the right shows the three binding site direction vectors of each of the Pyr_Redox_2 (yellow) and Fer2 (orange) query domains, with the target complex (PDB code 1e6e) shown below. This target has no FH templates, but the complex may still be modelled by re-using the SH binding sites of 2v3b_A and 1ffv_D (Table 6.4).

6.6.4 Summary of KBDOCK Case Retrieval Results

Table 6.6 summarises the results given in Table 6.4 and Table 6.5. Overall, it can be seen that KBDOCK can provide high quality FH docking templates for a total of 45 of the 73 targets (or 26 out of 73 with date filtering). Even when no FH templates exist, KBDOCK can still find useful binding site information for at least one of the domain partners for 12 of the remaining 28 targets (or 18 out of 47 with date filtering). More specifically, KBDOCK finds good FH templates for a total of 24 out of 36 Enzyme target complexes, although this number falls to 13 when PDB deposition date filtering is applied. A further 10 Enzyme targets have SH DDIs involving one or both of the target domains, and just 2 Enzyme targets have no hetero DDI information. Similarly, Table 6.6 shows that 21 (or 13 with date filtering) of the 37 Other targets may be modelled using FH templates, and a further 14 targets have SH DDIs involving one or both of the target domains. Like the Enzyme targets, only 2 of the Other targets have no hetero DDI information.

Table 6.6 also shows that all of the retrieved FH templates are correct according to the 10Å RMSD threshold. Thus, if KBDOCK retrieves a FH template, there is a high probability that it represents a good model of the target complex. Half of the number of targets for which only SH templates are available were found to reuse their known DFBSs. This suggests that in these cases focused

docking would give better predictions than blind docking. These results demonstrate that the approach embodied in KBDOCK provides a useful way to find protein docking templates.

Target class	Total targets	FH templates	SH-two templates	SH-one template	No templates
No date filtering					
Enzyme	36	24 / 24	(3 + 1) / 5	3 / 5	2
Other	37	21 / 21	(0 + 0) / 3	5 / 11	2
Total	73	45 / 45	(3 + 1) / 8	8 / 16	4
With date filtering					
Enzyme	36	13 / 13	(2 + 1) / 5	7 / 11	7
Other	37	13 / 13	(0 + 0) / 1	8 / 15	8
Total	73	26 / 26	(2 + 1) / 6	15 / 26	15

Table 6.6: Summary of the KBDOCK template modeling results for the 73 selected Protein Docking Benchmark targets. This table shows the number of docking targets for which the proposed templates are correct compared to the total number of templates retrieved. When SH-two templates are found, the figures in brackets give the number of cases in which both binding sites are modeled correctly *plus* the number in which only one binding site is modeled correctly.

6.7 The KBDOCK Case Refinement

Section 6.6.2 has shown that the CBR approach can often find potentially useful cases with which to model SH-two and SH-one problems but it cannot fully specify the proposed 3D binding mode. In this section, we describe how we use Hex rigid body docking program to refine the poses proposed by KBDOCK and we compare predictions obtained from such docking refinement against blind docking.

Using Hex for Fast Docking Refinement

The Hex rigid-body docking algorithm (Ritchie and Kemp, 2000) is used to refine and rank the SH initial poses generated using the procedure described in Section 6.6.2. Since Hex can perform all-versus-all docking of multiple PDB model structures, it is relatively straight-forward to prepare a Hex script to perform a rigid-body docking search around each putative DDI in the list $P(d1/b1, d2/b2)$. For SH-two cases, the docking search is focused around the given pair of binding site centre residues using two angular constraints, β_1 and β_2 (Ritchie *et al.*, 2008), as shown in Figure 6.4. On the other hand, for SH-one problems, just one angular constraint is applied to the known binding site, and the other domain is allowed to spin freely in order to search over its entire surface. If no DFBSs match the query, unconstrained blind docking is applied. Here, each pair-wise Hex docking run used 3D FFT shape-based correlation searches with range angles of β_1=β_2=45°, as appropriate, and 40 translational steps of 0.5 Å along the z axis with respect to each given starting orientation. This generates approximately 60, 360, and 2,000 million trial rigid body orientations for each pair-wise SH-two, SH-one, and blind docking run, respectively, of which the top 2,000 are re-scored using the

DARS potential (Chuang *et al.*, 2008) within Hex. For docking runs involving multiple combinations of DFBSs, individual pair-wise docking results are merged and sorted by DARS energy to give the final ranked list of solutions. Algorithm 1 shows some high-level pseudo-code which summarises the modelling choices of the KBDOCK approach.

Algorithm 1 High-level pseudo-code for modelling a DDI by CBR.

if $|FH(d1, d2)| > 0$ **then**
  apply substitution adaptation
else if $|SH(d1, D2)| > 0 \wedge |SH(D1, d2)| > 0$ **then**
  apply two substitutions and focused docking
else if $|SH(d1, D2)| > 0 \vee |SH(D2, d2)| > 0$ **then**
  apply one substitution and loosely constrained docking
else
  apply blind docking
end if

6.7.1 The Extended Docking Test Set

In the first test set described above, we considered only single-domain targets from the Protein Docking Benchmark. Here, we wish to test the KBDOCK CBR approach against a larger test set and therefore we include both single and multi-domain targets. The Protein Docking Benchmark 4.0 (Hwang *et al.*, 2010) consists of 176 targets in total. Here again, we exclude the 25 antigen-antibody complexes because the antibody binding sites are known *a priori* and because the antigen binding sites generally do not entail homology. We also exclude 11 targets for which no Pfam domain could be identified in one or both of their subunits (PDB codes 1clv, 1e6e, 1udi, 2abz, 2b42, 2uuy, 1f51, 1j2j, 1qa9, 1xqs, 2hrk), leaving 140 target complexes. In order to simulate a scenario in which trivial solutions of each target do not already exist, all targets were modelled using only structures with PDB deposition dates earlier than those of the query structures. Because we want to compare predictions obtained with blind docking with those obtained using the KBDOCK CBR approach, we exclude targets which have no DDI information in the KBDOCK database. This removed a further 39 targets from consideration, leaving 102 targets for which non-trivial homologues exist in the KBDOCK database for one or both partners. Of these 102 targets, 54 are single-domain targets and the remaining 48 are multi-domain targets.

6.7.2 Docking Refinement Results for Single-Domain Targets

Table 6.7 compares the CBR-based modelling results with blind Hex docking for the 54 single-domain target complexes of the extended docking test set (Section 6.7.1). Except for the angular search range constraints, the blind Hex runs used the same docking parameters as for the SH problems described above. For the target 1eaw, three FH template cases were found. Hence, for this target, the template with the highest overall similarity was used to build the final model by substitution

adaptation. As can be seen from the Rank and RMSD columns in Table 6.7, this procedure allowed 23 out of the 24 single-domain targets to be modelled accurately, with known DFBSs being re-used in all but one of the targets (indicated with an asterisk). The final two columns of Table 6.7 show the results obtained using Hex blind docking. Here, relatively low accuracy solutions were found for only 8 of the 24 targets, with six of those being ranked in the top 10 by the Hex scoring function. A hyphen in this table indicates that no "acceptable" solution within 10Å RMS from the native complex was found within the first 2,000 solutions using blind docking. These contrasting results highlight the utility of modelling DDIs using known DFBSs.

The next two sections of Table 6.7 show that 30 of the docking targets may be treated as somewhat more challenging SH problems (26 SH-one and 4 SH-two problems). Of the 26 SH-one problems, 15 of the known DBFSs turn out to be re-used, whereas 11 targets use previously unknown binding sites (according to the PDB date filter). It is unsurprising that KBDOCK gives no acceptable solution when the known DFBS is not reused in the target. Of the 15 targets that re-use known DFBSs, KBDOCK finds 8 good solutions, of which 6 are ranked within the top 4. For the remaining 7 targets (out of the 15), KBDOCK retrieves the template DFBSs, but the angular constraint passed to the docking stage is too tight to include a near-native orientation in the search space. Blind docking finds slightly more (11 targets) solutions overall, but the ranks for two of these (1kac and 1s1q) are too low to be useful. There are two SH-one targets (1kac and 2g77) for which blind docking finds a solution but KBDOCK does not.

On the other hand, for two of the four SH-two cases, the DFBSs on both partners are re-used and KBDOCK finds two good solutions at rank 1 and 2 for targets 1r0r and 1acb, respectively. Rigid body docking with Hex finds only one rather poor solution for these four targets. Overall, these results suggest that focused docking gives better predictions when the known DFBSs are reused in the targets.

6.8 Modelling Multi-Domain Docking Problems

So far we have considered only the relatively straightforward task of modelling single-domain protein complexes. This section describes how we extend the KBDOCK approach to deal with multi-domain PPIs. Here, we consider the problem of aggregating DDI-level CBR cases to model the 3D structure of a multi-domain protein complex.

6.8.1 Aggregating Multiple DDIs

In general, each protein to be docked may consist of multiple domains. Therefore, if $X = (d1, d2, ..., dN)$ and $Y = (e1, e2, ..., eM)$ represent the proteins X and Y and their component domains, then the preceding analysis of DDIs might suggest that we should take the Cartesian product of all possible pairs of component domains, apply the above KBDOCK CBR modelling procedure to each pair of domains, and then aggregate the solutions. However, because rigid body docking inevitably produces multiple false-positive solutions, it is important to consider computational docking as a measure of last resort, and to defer any docking calculation for as long as possible in the

Target PDB	Target Class	Template PDB	%-Seq 1	%-Seq 2	DFBS Pairs	DFBSs Re-used	CBR ±*Hex* Rank	CBR ±*Hex* RMSD	Blind *Hex* Rank	Blind *Hex* RMSD
Single-domain FH (no *Hex* docking)										
1ay7	E / R	1brs	29.4	94.1	1	Y + Y	1	2.89	–	–
1cgi	E / R	3sgb	20.6	34.7	1	Y + Y	1	1.56	2	9.07
1eaw	E / R	1tfx	41.5	39.6	3	Y + Y	1	0.93	7	8.06
1mah	E / R	1fss	58.9	96.7	1	Y + Y	1	0.76	2	4.63
1n8o	E / R	1ezu	41.3	95.3	1	Y + Y	1	1.31	–	–
1oph	E / R	1jmo	28.9	35.6	1	Y + Y	1	0.65	–	–
1yvb	E / R	1stf	33.8	19.8	1	Y + Y	1	2.98	–	–
2j0t	E / R	1bqq	46.7	31.2	1	Y + Y	1	2.01	5	9.07
2oul	E / R	2nqd	36.4	94.0	1	Y + Y	1	6.31	2	9.61
2sni	E / R	1cse	63.1	35.9	1	Y + Y	1	1.70	–	–
3sgq	E / R	1cgj	18.4	34.0	1	Y + Y	1	0.22	–	–
1ffw	O / R	1a0o	98.3	98.5	1	Y + Y	1	0.52	411	9.72
1fqj	O / R	1agr	63.6	35.3	1	Y + Y	1	0.47	–	–
1gpw	O / R	1ka9	58.5	37.6	1	Y + Y	1	3.57	5	8.78
1he1	O / R	1g4u	99.4	30.0	1	Y + Y	1	0.98	–	–
1xd3	O / R	1cmx	21.8	94.2	1	Y + Y	1	1.12	–	–
2a9k	O / R	2a78	98.8	33.2	1	Y + Y	1	1.03	–	–
2hle	O / R	1shw	44.2	26.7	1	Y + Y	1	2.01	–	–
2oob	O / R	2bwe	20.5	33.3	1	Y + N	*	*	335	9.90
1grn	O / M	1am4	93.8	96.0	1	Y + Y	1	1.74	–	–
1r6q	O / M	1r6o	98.0	98.8	1	Y + Y	1	0.77	–	–
2ayo	O / M	1nbf	22.1	97.1	1	Y + Y	1	1.76	–	–
2nz8	O / M	2dfk	24.0	71.3	1	Y + Y	1	2.84	–	–
3cph	O / M	1vg0	19.4	37.9	1	Y + Y	1	1.86	–	–
Single-domain SH-one (with *Hex* docking)										
1f34	E / R	1htr	51.0	–	1	N + U	*	*	5	8.64
1fle	E / R	1pyt	53.8	–	3	Y + U	1	7.24	–	–
1gl1	E / R	1acb	98.6	–	6	Y + U	4	8.01	6	7.32
1hia	E / R	2kai	64.3	–	3	Y + U	52	9.56	11	8.45
1oc0	E / R	1k9o	30.1	–	1	N + U	*	*	–	–
1ppe	E / R	1tab	100.0	–	2	Y + U	1	3.54	1	3.29
2sic	E / R	1cse	63.4	–	1	Y + U	1	9.69	1	7.33
1nw9	E / M	1i4e	34.9	–	1	N + U	*	*	23	6.49
1fq1	E / D	1buh	98.6	–	3	N + U	*	*	–	–
1zli	E / D	2bo9	7.5	–	1	Y + U	2	7.77	2	7.13
1buh	O / R	1fin	98.6	–	2	N + U	*	*	–	–
1e96	O / R	1foe	97.2	–	3	N + U	*	*	–	–
1gcq	O / R	1ycs	32.7	–	1	N + U	*	*	–	–
1i4d	O / R	1g4u	99.4	–	3	N + U	*	*	–	–
1kac	O / R	1akj	–	17.0	1	U + Y	–	–	265	8.76

Table 6.7: Summary of the CBR-based docking results (continued).

reasoning process. Furthermore, since each of the target proteins will normally be provided as complete 3D structures, it is not necessary to model any internal DDIs because these are given as part of the problem. Indeed, such internal interactions will obviously “consume” a certain number of DFBSs, thus blocking them from interacting with the domains of the other protein. Therefore,

Target	Target	Template	%-Seq	%-Seq	DFBS	DFBSs	CBR ±*Hex*		Blind *Hex*	
PDB	Type	PDB	1	2	Pairs	Re-used	Rank	RMSD	Rank	RMSD
Single-domain SH-one (with *Hex* docking; contd)										
1s1q	O / R	1otr	–	89.9	2	U + N	*	*	190	9.59
1z0k	O / R	1ukv	49.4	–	5	Y + U	2	3.00	2	7.06
2g77	O / R	1ukv	–	40.7	5	U + Y	–	–	55	8.58
1lfd	O / M	1wq1	99.4	–	2	Y + U	48	8.59	–	–
1mq8	O / M	1ijk	–	21.7	2	U + N	*	*	–	–
1wq1	O / M	1gua	57.7	–	2	Y + U	–	–	–	–
2h7v	O / M	1g4u	99.4	–	5	Y + U	–	–	–	–
1r8s	O / D	1r4a	55.6	–	3	Y + U	–	–	–	–
1y64	O / D	1nm1	89.3	–	4	Y + U	–	–	–	–
2ido	O / D	1zbu	17.8	–	1	N + U	*	*	–	–
2ot3	O / D	2hv8	38.3	–	5	Y + U	–	–	–	–
Single-domain SH-two (with *Hex* docking)										
1avx	E / R	1mct + 1ava	100.0	30.1	5	Y + N	*	*	–	–
1r0r	E / R	1cse + 1ppf	89.5	100.0	2	Y + Y	2	2.61	61	9.90
2o8v	E / R	1zun + 2bto	17.9	99.0	2	N + Y	–	–	–	–
1acb	E / M	1cgj + 2tec	100.0	100.0	2	Y + Y	1	8.60	–	–

Table 6.7: Summary of the KBDOCK versus blind docking results. A – denotes no solution found within the top 2,000 docking predictions. An * denotes no solution was found by KBDOCK because a DFBS was not re-used. Other abbreviations used: E, O, R, M and D stand for 'Enzyme', 'Other', 'rigid body', 'medium difficulty' and 'difficult' target, respectively. Y, N and U stand for 'Yes' (DFBS re-used), 'No' (DFBS not re-used), and 'Unknown' binding site, respectively.

our first strategy is to remove from consideration any DDIs that are implicitly blocked by the other components of the query. This is done by identifying the binding site centres of each domain in the query protein (by querying the CB as described above) and by striking out any binding sites whose centre residues are buried in the query protein. This reduces the number of DFBSs that should be considered as possible docking sites. The overall procedure is illustrated in Figure 6.6.

The next step is to form a Cartesian product of the surviving DFBSs of each protein, and to query the CB with each putative pair of such DFBSs in order to collect sets of FH and SH cases from the CB. If no cases are retrieved, then Hex blind docking is applied directly. Otherwise, if any FH cases are retrieved we assume that the problem can be modelled by superposing the query structures onto the best FH template, as before (Section 6.6.1). The only difference from the single DDI procedure is that now all of the atoms of each protein are transformed by the superposition transformation. On the other hand, if no FH case and no SH-two cases are retrieved the proteins are docked and ranked by applying the SH-one procedure to the set of available DBFSs (Section 6.6.2). Otherwise, all available domains are cross-docked and ranked using the SH-two procedure. Here again, the main difference from the single DDI SH modelling procedure is that now all of the atoms of each protein are transformed when making a docking pose.

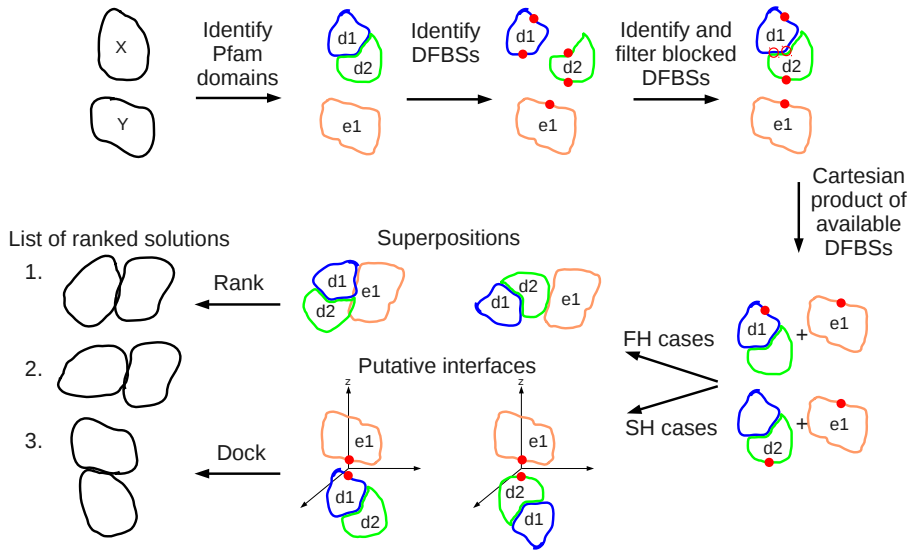


Figure 6.6: Overview of the KBDOCK approach for modelling multi-domain PPIs. Given two proteins, KBDOCK uses PfamScan to identify the Pfam domains. For each Pfam domain, KBDOCK retrieves one or more DFBSs and it filters out any DFBS which is consumed by intra DDIs. From the remaining DFBSs, KBDOCK enumerates all the possible pairings using a Cartesian product. Depending on the type of the DFBS case (FH or SH), a coordinate transformation is applied and a docking refinement is used if necessary. If FH cases are found, the predictions are ranked according to sequence similarity. Otherwise, the Hex rigid-body docking program is used to refine and rank the initial docking poses.

6.8.2 KBDOCK Modelling Results for Multi-Domain Targets

Table 6.8 shows the results obtained for the 48 multi-domain targets. Like the single-domain targets, the first group of entries shows a very satisfying pattern of results for six multi-domain FH problems. For the three targets 1d6r, 1ezu, and 1wdw, one of the partners is a large symmetric homodimer. In all three of these cases, the KBDOCK approach correctly retrieves very low RMSD solutions for each monomer of the dimer (shown as three rank-1 solutions in the table). The next three targets, 2vdb, 1h1v, and wi9b, involve three assymetric repeats of the same domain in one of the partners, and KBDOCK again finds acceptable or better quality docking for these three targets (two of which are “Difficult”). In contrast, for these six targets, blind docking found only one solution (for 1d6r) with a very poor rank.

On the other hand, only a relatively small number of the 34 multi-domain SH-one targets are predicted well either by KBDOCK or by blind docking. Of the 34 targets, 13 targets do not reuse their known DFBSs and therefore focused docking does not find any acceptable solution. Blind docking finds acceptable solutions for one of these 13 targets. The remaining 21 targets reuse one of their known DFBS but focused docking finds acceptable solutions for only 5 of them (2 targets within the top 10 rank). For these 5 targets, blind docking performs better (3 targets within the top 10 rank). This is because 9 of the 21 targets for which their DFBSs are reused are intrinsically difficult to score and rank using a rigid body docking algorithm.

Target PDB	Target Type	Template PDB	%-Seq 1	%-Seq 2	DFBS Pairs	DFBSs Re-used	CBR ±*Hex* Rank	CBR ±*Hex* RMSD	Blind *Hex* Rank	Blind *Hex* RMSD
Multi-domain FH (no *Hex* docking)										
1d6r	E / R	1tab	100.0	65.4	2	Y + Y	1	1.60	379	8.22
1ezu	E / R	1azz	38.6	99.2	2	Y + Y	1	0.95	–	–
1wdw	E / R	2tys	60.0	30.4	2	Y + Y	1	0.71	–	–
2vdb	O / R	1tf0	25.7	96.0	3	Y + Y	1	3.59	–	–
1h1v	O / D	1d4x	92.5	96.4	3	Y + Y	1	8.79	–	–
2i9b	O / D	2fd6	20.5	100.0	3	Y + Y	1	6.02	–	–
Multi-domain SH-one (with *Hex* docking)										
1bvn	E / R	1viw	50.2	–	1	Y + U	1	9.76	1	4.19
1ewy	E / R	1ep3	19.6	–	1	Y + U	1	7.82	3	5.98
1oyv	E / R	2sec	89.5	–	2	Y + U	–	–	–	–
1tmq	E / R	1dhk	52.5	–	2	Y + U	17	6.25	3	7.73
1ijk	E / M	1hyr	–	24.6	2	U + N	*	*	–	–
1jiw	E / M	1smp	–	36.9	1	U + Y	–	–	–	–
1kkl	E / M	1ggr	–	34.5	1	U + Y	–	–	–	–
1a2k	O / R	1wq1	27.2	–	1	Y + U	–	–	–	–
1azs	O / R	1gg2	–	39.1	1	U + Y	–	–	–	–
1b6c	O / R	1fin	–	21.5	3	U + N	*	*	1	6.16
1fcc	O / R	1frt	19.5	–	2	U + N	*	*	–	–
1jwh	O / R	1buh	–	29.4	3	U + N	*	*	–	–
1ktz	O / R	1es7	–	34.0	1	U + N	*	*	–	–
1kxp	O / R	1hlu	92.5	–	4	Y + U	–	–	–	–
1ofu	O / R	1ia0	–	14.3	1	U + N	*	*	–	–
1pvh	O / R	1f6f	26.6	–	2	Y + U	–	–	–	–
1rv6	O / R	1pdg	24.4	–	2	N + U	*	*	–	–
1t6b	O / R	1v7p	–	16.2	3	U + N	*	*	–	–
1xu1	O / R	1oqd	37.0	–	6	Y + U	109	7.56	–	–
2b4j	O / R	1c6v	63.0	–	2	N + U	*	*	–	–
2fju	O / R	1g4u	–	99.4	4	U + Y	–	–	–	–
3bp8	O / R	1o2f	–	94.3	1	U + Y	–	–	–	–
3d5s	O / R	1ghq	99.6	–	2	Y + U	28	8.93	64	3.29
1he8	O / M	1gua	57.1	–	2	Y + U	–	–	–	–
1i2m	O / M	1ibr	98.1	–	3	Y + U	–	–	–	–
1k5d	O / M	1rrp	99.2	–	3	Y + U	–	–	–	–
1n2c	O / M	3min	99.8	–	4	Y + U	–	–	–	–
1bkd	O / D	1wq1	99.4	–	2	Y + U	–	–	–	–
1de4	O / D	1qqd	44.2	–	5	Y + U	–	–	–	–
1eer	O / D	1axi	–	14.4	2	U + N	*	*	–	–
1fak	O / D	1kig	41.5	–	4	N + U	*	*	–	–
1jmo	O / D	2hpp	–	99.6	6	U + Y	–	–	–	–
1zm4	O / D	1aip	15.7	–	3	N + U	*	*	–	–
2c0l	O / D	1e96	16.7	–	1	N + U	*	*	–	–
Multi-domain SH-two (with *Hex* docking)										
1gxd	E / R	1uea + 1smp	27.5	41.3	2	N + N	*	*	–	–
1m10	E / M	1ijk + 1dfj	98.3	36.0	1	N + N	*	*	147	8.85
1akj	O / R	1hhg + 1agf	100.0	23.9	3	N + Y	*	*	150	7.55
1hcf	O / R	1sgf + 1evt	48.7	20.7	2	N + N	*	*	–	–
1klu	O / R	1aqd + 1klg	100.0	98.1	44	Y + Y	–	–	–	–
1sbb	O / R	1rvf + 2seb	31.3	81.0	3	N + N	*	*	–	–
1gp2	O / M	1got + 1got	68.9	98.3	13	Y + Y	148	9.11	–	–
1ibr	O / D	1qbk + 1rrp	98.1	18.3	3	Y + Y	–	–	–	–

Table 6.8: Summary of the KBDOCK versus blind docking results for multi-domain targets. See Table 6.7 for abbreviations used.

6.9 Discussion and Conclusion

This chapter has described a systematic CBR-based approach to model the 3D structures of protein complexes from structural DDIs, and we have tested it on a well known benchmark dataset. By working at the Pfam domain level, we are able to draw upon a non-redundant set of almost all currently known hetero DDIs to model pair-wise DDIs and furthermore to aggregate pair-wise DDIs to predict larger complexes. The results in Tables 6.7 and 6.8 show that for FH problems, in which previously solved cases may be re-used, our approach provides a near-perfect way to retrieve good 3D templates and to build high quality models of the target complexes.

To our knowledge, none of the previous homology docking methods have grouped binding sites in domain families according to their location in 3D space. The closest related work is probably the work of Korkin *et al.* (2006). As described in Section 2.3.3, the approach of Korkin *et al.* needs multiple docking runs and there is a need for an additional step of merging the predictions and ranking them. For example, given a target pair of domains (d1, d2) which have N and M instances of SCOP domains which are involved in an interaction, the number of docking runs performed is N$\times$M. This number can be large for pairs of SCOP families which have many DDI instances. The KBDOCK CBR approach can also involve running several docking calculations but because we initially cluster the DDIs into distinct DFBSs, the number of docking runs in our approach is normally much less.

Using PDB deposition date filtering to exclude trivial FH solutions from consideration caused a good proportion of the benchmark targets to be treated as more difficult SH problems. For the single-domain SH targets, we find that known binding sites are re-used in approximately half of the benchmark targets, and this knowledge can usefully be exploited to guide rigid-body docking. Hence our case-based method of re-using DDIs extends the reach of current homology modelling techniques. Indeed, our results show that if known binding sites are in fact re-used, a good model is often ranked within the first handful of solutions. Although our current implementation can miss some SH-one solutions which are found by blind docking, we believe this is mainly because the angular constraint passed to the rigid docking stage is too strict.

On the other hand, our results for multi-domain SH targets show that it is difficult to use computational docking to identify the correct pair of binding sites in multi-domain complexes. This is perhaps not surprising given that each trial pair of initial docking poses will produce a large number of false-positive docking predictions which will mask any near-native solutions. Indeed, if both partners have multiple binding sites, then without additional biological knowledge we essentially revert to a blind docking problem (as in the SH-two targets 1klu and 1gp2 examples, with 44 and 13 candidate DDIs to be docked, respectively). Hence, it would be desirable to incorporate more sophisticated restraints derived from other kinds of biological evidence and to use a more powerful flexible docking algorithm such as HADDOCK (de Vries *et al.*, 2010), especially when dealing with difficult targets involving conformational flexibility.

Chapter 7

Conclusions

7.1 Summary of the Main Contributions

This thesis has presented a systematic KDD-based approach for representing, describing, and comparing 3D protein-protein interactions on a large scale and to facilitate knowledge-based modelling of protein-protein complexes. We have used KDD techniques to integrate protein and PPI data from several sources in order to extract structural knowledge in a useful form. Particular emphasis was put on deriving 3D PPI knowledge that can be reused easily because, in our opinion, current PPI resources do not provide an easy way to incorporate homology information into computational methods for predicting 3D protein-protein complexes.

The main contributions of this thesis are the following: (1) we have developed an integrated database of non-redundant 3D hetero domain-domain interactions; (2) we have described a novel method of describing and clustering DDIs according to the spatial orientations of their binding partners, thus introducing the notion of domain family binding sites; (3) we have proposed a structural classification of DFBSs similar to the CATH classification of protein folds, and we have presented a study of secondary structure propensities of DFBSs and interaction preferences; (4) we have introduced a systematic case-base reasoning approach to model automatically the 3D structures of protein complexes from existing structural DDIs. All of these contributions have been made publicly available through a web server (http://kbdock.loria.fr). The research contributions of this thesis have been described in one published article and in three further manuscripts which are currently being revised for submission. Copies of these articles are provided in Appendices B and C.

7.1.1 The KBDOCK Database of 3D Non-Redundant Hetero DDIs

The KBDOCK database provides the foundation of this thesis. KBDOCK integrates DDI and residue contact information from the 3DID database, domain family information from the Pfam database, and structural information from the PDB. The data in KBDOCK are organised according to Pfam domain families to facilitate the analysis of 3D interactions. Thus, for each Pfam domain family, KBDOCK calculates and stores a non-redundant set of hetero DDIs which are placed in a common coordinate frame using least-squares fitting based on a Pfam consensus sequence align-

ment. Chapter 3 has shown that this provides a useful way to explore the structural relationships between the members of a given Pfam family. The current version of KBDOCK contains a total of 2,721 non-redundant hetero DDIs involving 1,029 different Pfam domain families. Hence, KBDOCK constitutes one of the largest collections of 3D hetero DDIs to date. It therefore provides a useful framework for performing statistical analyses and applying data mining algorithms to 3D DDI data in order to discover trends and patterns at domain-domain interfaces.

7.1.2 The Domain Family Binding Site Concept

The remarkable consistency of many of the DDI superpositions observed in Figure 3.12, for example, led us to organize the DDIs in protein domain families according to the spatial position of their binding sites. To do this, we calculated a binding site direction vector from the all-atom centre of mass of a domain to the centre of its binding site for each superposed domain using a weighted average of its core and rim residues. For each Pfam family, we clustered the binding site direction vectors of the family members using a hierarchical clustering algorithm. Each of the obtained clusters represents a group of binding sites belonging to the same Pfam family and for which the binding partners appear in similar spatial positions. We used this grouping to define the concept of a domain family binding site (DFBS). This concept has provided a natural basis with which to perform large-scale studies of the spatial arrangements of DDIs by Pfam family. For example, we explored whether binding sites are conserved within Pfam families and whether binding sites might be promiscuous. Chapter 4 has shown that nearly 70% of the 1,029 Pfam domain families in KBDOCK have only one DFBS, and most of the remaining families have from just two to four DFBSs. This confirms previous studies that DDIs often re-use their binding sites (Korkin *et al.*, 2005, Shoemaker *et al.*, 2006). Additionally, we found that over 80% of DFBSs (out of a total of 1,439 DFBSs) interact with just one type of Pfam domain family, and that very few DFBSs interact with more than three different Pfam domain families. This indicates that most DFBSs are primarily monogamous in their structural relationships with other domains. These results have been described in an article published in *Bioinformatics* (Ghoorah *et al.*, 2011). The DFBS concept has played a central role in the other contributions of this thesis, as summarized below.

7.1.3 Structural Classification and Study of Domain Family Binding Sites

Another question that arises when studying interactions between different protein domain families is whether the binding sites of different Pfam families might have common features. Therefore, we analysed the secondary structure content of DFBSs, and the secondary structure preferences between pairs of DFBSs. To achieve this, we proposed a structural classification of binding sites similar to the CATH domain family classification (Cuff *et al.*, 2009). Our classification identifies seven main secondary structure types, each consisting of different proportions of the three main secondary structure elements (SSEs). Chapter 5 has shown that helices and irregular secondary structures are the most common types of SSE in DFBSs. These results confirm some previous findings (Keskin and Nussinov, 2007, Guharoy and Chakrabarti, 2007). However, our classification also showed that there is no specific "α + β" type and that DFBSs always contain a considerable fraction of γ SSEs (average 34.6%). In addition, our study showed that α-α interactions and γ-γ interactions are rather

frequent, whereas α-β and β-β interactions are rather strongly disfavoured. Chapter 5 has also shown that there is almost no difference in the SSE character of single partner binding sites and multi-partner binding sites. Thanks to KBDOCK, our study of structural interactions represents the most comprehensive one to date. A manuscript describing this work was submitted to *ISMB 2012* and is being revised to take into account the referees' comments.

7.1.4 Case-Based Protein Docking

This thesis has also shown how to use DFBSs within a case-based reasoning framework in order to guide protein docking calculations. By exploiting the fact that pairs of similar domains often interact in the same way and the fact that the binding sites within domain families are often conserved, we have designed and implemented a CBR approach within KBDOCK to propose templates and key binding site residues for modelling single and multi-domain protein-protein complexes. The DFBS notion forms the basis for case indexing and retrieval in our CBR approach and thus provides a convenient way to retrieve distinct DDIs from the KBDOCK database. Chapter 6 has shown that when FH DDIs exist, KBDOCK provides a near-perfect way to retrieve good 3D templates and to build high quality models of the target complexes using a simple structural superposition. Otherwise, when only SH DDIs exist, KBDOCK uses their DFBS information to propose a starting orientation for focused docking. Our results using the Hex docking algorithm show that when known DFBSs are re-used in the query domain structures, focused docking can improve significantly the docking predictions. A manuscript describing this approach was submitted to *ECCB 2012* and received favorable comments. A revised manuscript is in preparation.

7.1.5 The KBDOCK Web Server

We have developed a web server (http://kbdock.loria.fr) to provide open access to the KBDOCK resource. Since it was first published on-line in March 2011, it has received 3,115 visits (as of October 2012). As shown in Appendix A, the KBDOCK web server has an easy-to-use form-based interface, and it can be queried in two main ways. In *browse* mode, the user may find and visualise a list of hetero DDIs involving either one or a pair of given Pfam query domains. The retrieved DDIs are grouped by their DFBS and the user can pick a DFBS of interest for further analysis. In *docking* mode, the user provides two structures. The user is then provided with a non-redundant list of candidate DDI templates. Even if no matching DDIs exist in the database, KBDOCK can often still provide useful binding site information for one or both query domains using knowledge of their known binding sites. This knowledge may then be used automatically to provide starting orientations which can then be docked and refined using the Hex server (Macindoe *et al.*, 2010). All of the results of queries against the database may be visualised in a common coordinate frame using the Jmol plug-in, and all queries may be expressed using Pfam IDs or by providing the PDB codes or amino-acid sequences of the domains of interest. A manuscript describing the KBDOCK server is ready for submission in the next web server issue of the *Nucleic Acid Research* journal.

7.2 Timeliness and Novelty

Given the growing amount of structural data in public databases, the KBDOCK system represents a timely contribution. The recent interest in using docking algorithms to predict protein interaction partners (Sacquin-Mora *et al.*, 2008, Wass *et al.*, 2011, Melquiond *et al.*, 2012), suggests the need for reliable docking predictions. As discussed in Chapter 1, the reliability of docking predictions can be improved if experimental information from related PPIs is incorporated into docking algorithms (van Dijk *et al.*, 2005, Qin and Zhou, 2007*b*, Lensink and Wodak, 2010). However, until now, it has been difficult to find relevant PPI information automatically.

Compared to previous PPI databases, our KBDOCK database was designed right from the beginning to identify automatically structural templates with which to guide protein docking calculations. Hence, KBDOCK has many features which distinguish it from existing structural PPI database approaches: (i) it uses the Pfam consensus sequence to guide structural alignments, (ii) it places all of the complexes involving a given Pfam domain family into a common coordinate frame in order to locate the interaction partners consistently in 3D space, (iii) it uses the notion of "core" and "rim" binding site residues to group the complexes by the spatial position of their binding site, (iv) it finds automatically the best available DDI template to use to model by homology a complex of two given structures, (v) if more than one interface is found, it proposes a model for each, (vi) if no suitable DDI template exists, it can still propose candidate binding sites for one or both interaction partners, (vii) it calculates a centre residue for each proposed binding site which may be used to initialise a docking calculation and finally (viii) using its proposed starting orientations, it prepares and submits a focused docking job using the Hex docking server Macindoe *et al.* (2010). Hence, KBDOCK provides a novel and useful resource for analysing the 3D structures of DDIs within and between Pfam domain families, thus helping to provide a sharper picture of the 3D interactome.

7.3 Future Extensions to KBDOCK

The current KBDOCK database is built from the November 2009 version of the 3DID database. We are updating KBDOCK to use the latest version of 3DID. In the new version, we will include intra and homo interactions because these may provide useful information for predicting multi-domain hetero interactions. Our binding site direction vector algorithm can be applied to homo and intra DDIs, and therefore we will define DFBSs for homo and intra DDIs as well as the existing hetero DFBSs. We will then apply our secondary structure analyses to homo and intra DFBSs. This will represent the largest study of domain binding sites in hetero, homo and intra DDIs. We will also use knowledge of homo and intra DDIs to improve our DDI aggregation method of predicting multi-domain PPIs. We will extend the KBDOCK CBR case retrieval algorithm to handle a wider range of semi-homology problems. For example, we will use either the SCOP or CATH hierarchy to generalise docking problems. Furthermore, we plan to explore ways to compare and re-use 3D peptide fragments within binding sites.

Regarding the KBDOCK web server, we are currently extending it to provide the starting docking orientations and the corresponding interaction restraints (AIRs) which the user can download in

order to perform focused docking with other popular programs such as HADDOCK. Furthermore, we are extending the server to deal with multi-domain complexes. Other possible technical extensions could include providing a web service access to KBDOCK. We could also distribute the KBDOCK database and programs to allow users to launch queries locally.

7.4 Future Prospects

Given the rapid growth in the number of 3D structures in the PDB, there is a growing need to be able to study and predict 3D interactions on a large scale. We expect that KBDOCK may be used: (i) to analyse binding sites shapes and geometries within and across protein domain families; (ii) to provide supporting evidence (by cross-docking) for PPIs coming from HTT data; (iii) to investigate the structural nature of hub proteins; and (iv) to calculate the 3D structures of all the interactions in a PPI network. In the future, we hope that other PPI databases could provide cross-references to KBDOCK to enrich PPI networks with 3D structural information. In conclusion, we believe the KBDOCK system will be useful to anyone interested in studying protein-protein interactions, template-based modelling of 3D protein-protein complexes, and docking by homology. More generally, we expect that KBDOCK will provide a useful foundation for further KDD-based studies of the 3D interactome.

Research Outputs

Published Article

- Ghoorah A W, Devignes M-D, Smaïl-Tabbone M, and Ritchie D W. Spatial clustering of protein binding sites for template-based protein docking. *Bioinformatics 2011: 27(20): 2820-7*.

Manuscripts in Preparation

- Ghoorah A W, Devignes M-D, Smaïl-Tabbone M, and Ritchie D W. Using case-based reasoning to model 3D protein complexes from structural domain-domain interactions.
- Ghoorah A W, Devignes M-D, Smaïl-Tabbone M, and Ritchie D W. A systematic structure-based classification and analysis of protein domain family binding sites.
- Ghoorah A W, Devignes M-D, Smaïl-Tabbone M, and Ritchie D W. KBDOCK: A new resource for knowledge-based protein docking.

Conference Presentations

- Spatial Clustering of Protein Binding Sites for Template-Based Protein Docking. *BeNeLux Bioinformatics Conference (BBC 2011)*, Luxembourg, 2011.
- Extracting and visualising protein-protein interfaces for knowledge-based docking. *Journées Scientifiques de la Fédération Charles Hermite*, Nancy, 2010.

Conference Posters

- Extracting and visualizing protein-protein complexes for knowledge-based docking. *LIX Bioinformatics Colloquium 2010*, Paris, 2010 and *9th ECCB*, Ghent, 2010.
- Developing a Knowledge-Based Approach to Protein-Protein Docking. *4th CAPRI Evaluation Meeting*, Barcelona, 2009.

Software and Web Servers

- The KBDOCK web server is available at *http://kbdock.loria.fr*.

Appendix A

The KBDOCK Web Server

A.1 Introducing the KBDOCK Web Server

To provide open access to the KBDOCK database, I have implemented the KBDOCK web server (http://kbdock.loria.fr/). The server implements much of the functionality described in the previous chapters. It has an easy-to-use form-based interface (Figure A.1), and it can be queried in two main ways. In *browse* mode, the user may find and visualise a list of hetero DDIs involving a given Pfam query domain. In *docking* mode, the user provides two query domains, and is provided with a non-redundant list of candidate DDI docking templates. Even if no matching DDIs exist in the database, KBDOCK can often still provide useful binding site information for one or both query domains using knowledge of their known binding sites. In either case, the user can select a pair of binding sites to launch a focused docking calculation using the Hex docking server (Macindoe *et al.*, 2010). All of the results of queries against the KBDOCK database may be visualised in a common coordinate frame using the Jmol plug-in, and all queries may be expressed using Pfam IDs or by providing the PDB codes or amino-acid sequences of the domains of interest. Thus, the KBDOCK server provides a novel and easy way to explore and visualise known DDIs and for finding knowledge-based templates with which to model unsolved protein complexes. The following sections describe the implementation details and the functionality of the KBDOCK server in more detail.

A.2 Implementation Details

The KBDOCK server was written mainly in the PHP scripting language. Some JavaScript was used for handling certain mouse events. Database queries are processed using PrologScript[27] and Linux shell scripts. The Jmol [28] plug-in is used for graphical visualisation. The Jmol plugin requires Java version 1.4 or later. The web interface has been tested using several popular browsers for the Windows, Linux, and Mac OS X operating systems.

[27] http://www.swi-prolog.org/
[28] http://jmol.sourceforge.net

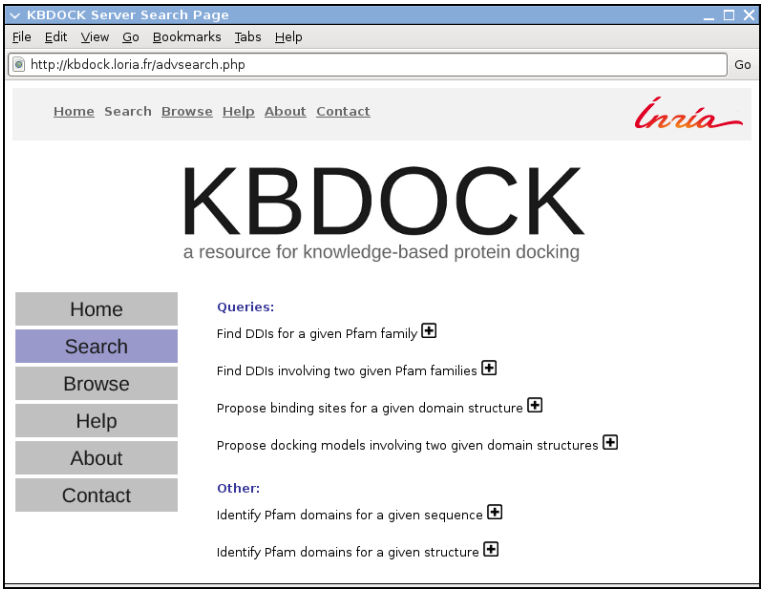


Figure A.1: A screenshot of the KBDOCK "Search" page.

A.3 Analysing Binding Sites by Pfam Family

In order to analyse the binding sites of a given Pfam family, the user may use the KBDOCK "Search" page (Figure A.1) to enter either a Pfam identifier (e.g. *Kunitz_legume*), a Pfam accession number (e.g. PF00197), a keyword (e.g. inhibitor), an amino acid sequence, or a PDB file of a protein structure. If a sequence or a structure is entered, the PfamScan utility [29] is used to determine the Pfam accession number. Otherwise, the accession number is found directly from the KBDOCK database. KBDOCK then retrieves a non-redundant list of hetero DDIs involving the query domain, grouped by their domain family binding site.

Figure A.2 shows the results page when KBDOCK is queried using the *Kunitz_legume* protease inhibitor domain (PF00197). The Jmol plugin shows the retrieved DDIs in the coordinate frame of the query domain. By convention, the query domain is shown in black and interacting residues are shown as wire sticks. The user may choose to view the DDIs together or individually. A Pfam consensus-based sequence alignment of the retrieved domains is also provided, in which each sequence is colour-coded according to the calculated core, rim and centre residues. A link to download the superposed PDB files as a single compressed file for further analysis is also available. A unique feature of KBDOCK is that it can also be queried with two query domains simultaneously. For example, Figure A.3 shows the output when KBDOCK is queried using the *Kunitz_BPTI* protease inhibitor domain (PF00014) and the *Trypsin* domain (PF00089). The results show that *Kunitz_BPTI* domain interact with *Trypsin* in two different ways.

[29] ftp://ftp.sanger.ac.uk/pub/databases/Pfam/Tools/

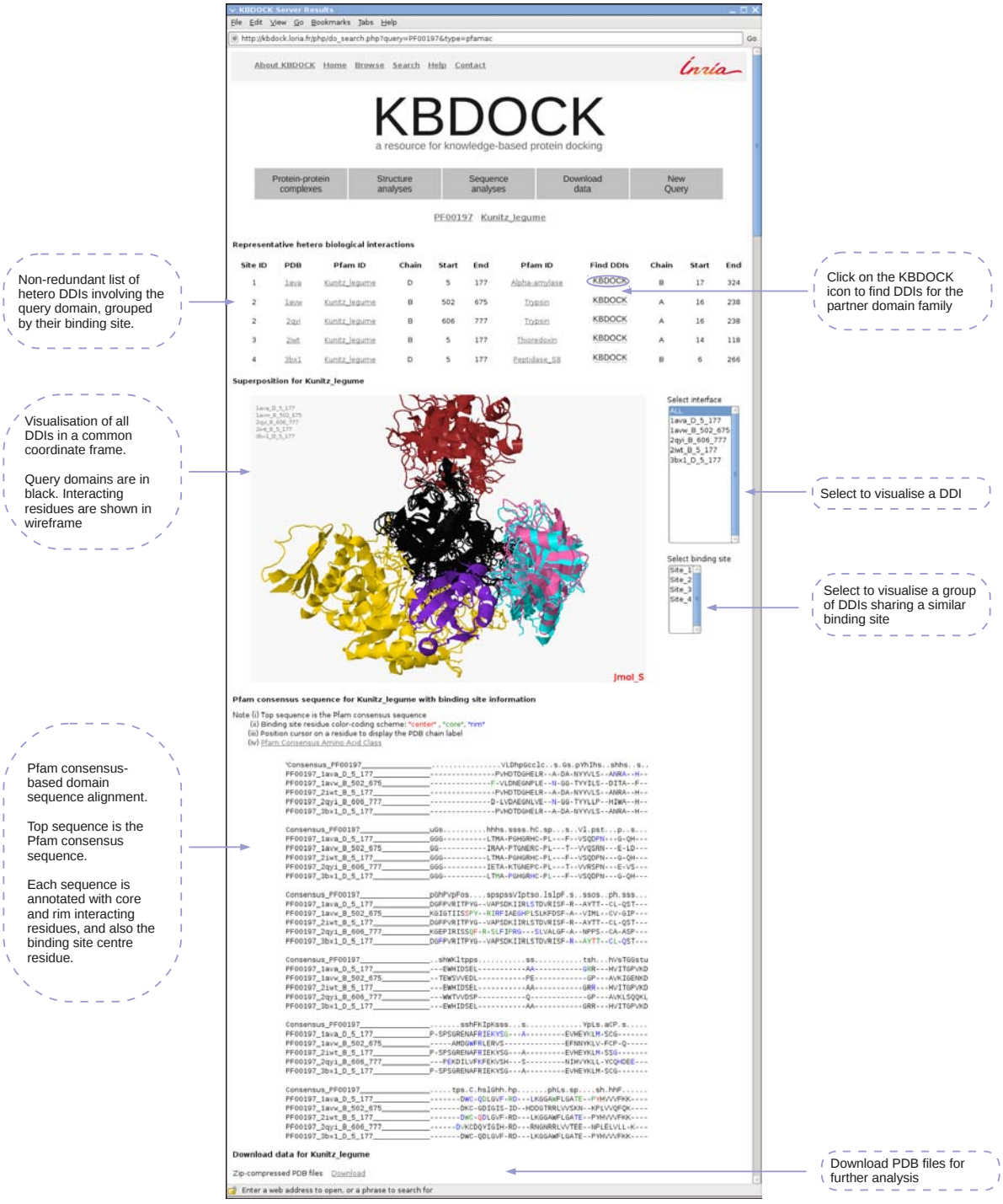


Figure A.2: A screenshot of the KBDOCK results page for the query domain family, *Kunitz_legume*. The results page consists of four sections: (i) a non-redundant list of hetero DDIs grouped by their binding site, (ii) a Jmol view of the DDIs in the coordinate frame of the query domain (the query domain is shown in black and interface residues are shown in wireframe), (iii) a Pfam consensus-based sequence alignment of the domains annotated with the core (green), rim (blue), and centre (red) binding site residues, (iv) a link to download the superposed PDB files as a single compressed file for further analysis.

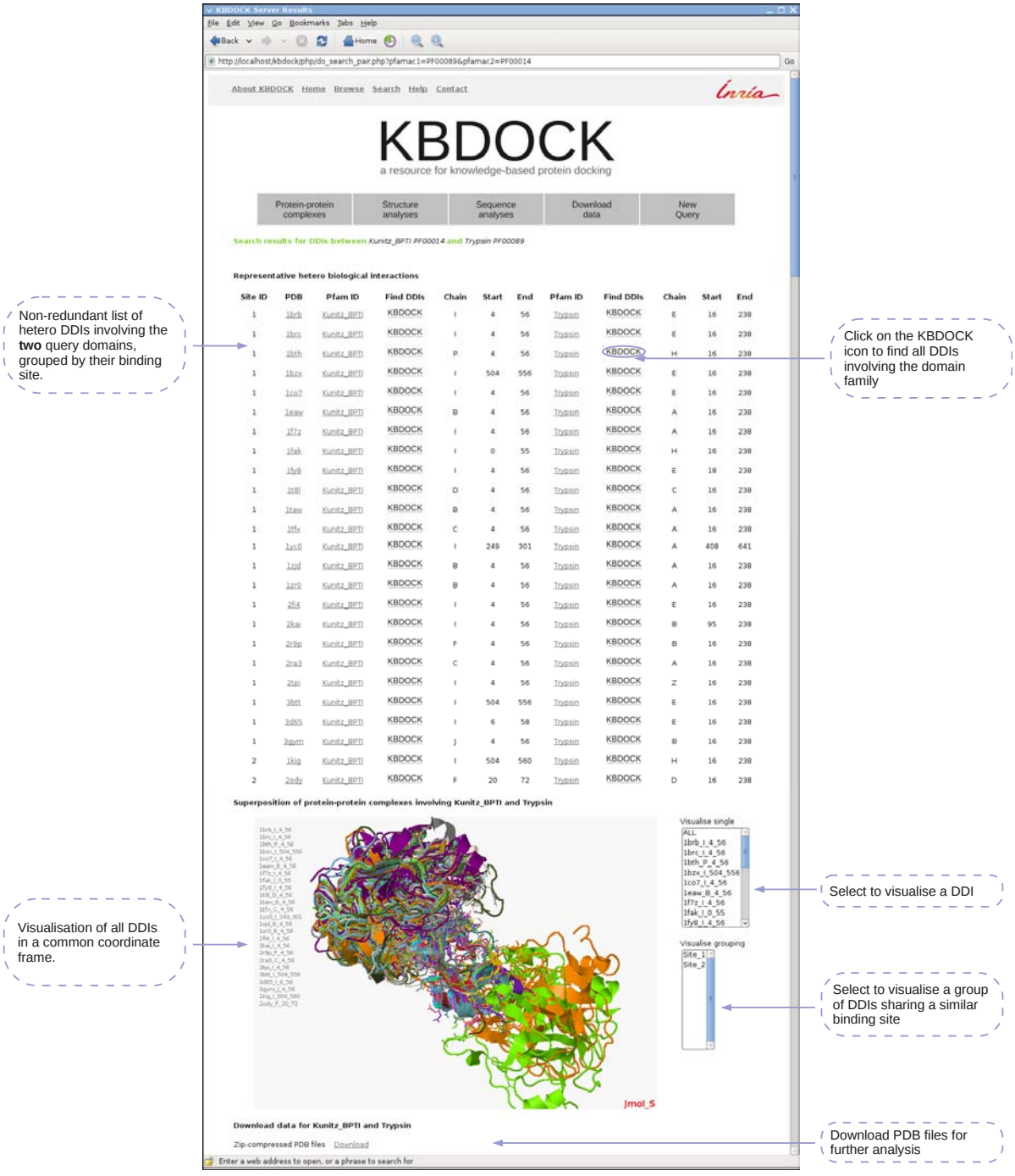


Figure A.3: A screenshot of the KBDOCK results page for the pair of query domain families, *Kunitz_legume* and *Trypsin*.

A.4 Proposing Binding Sites for a Query Domain Structure

KBDOCK can propose the locations of possible binding sites for a given query domain structure using the known binding sites of the corresponding Pfam family. For example, Figure A.4 shows a

screenshot of the results page obtained when querying KBDOCK using PDB code 1nty. This structure involves the *RhoGEF* domain family for which KBDOCK identifies two DFBSs. To illustrate a DFBS, KBDOCK shows a Jmol view of the DDI template with the query domain structure superposed onto it. In addition, to indicate the approximate centre of the DFBS of the query domain, the calculated centre residue is shown in wire frame. For each DFBS found, KBDOCK shows a colour-coded sequence alignments of the query and template domains showing the core, rim, and centre binding site residues.

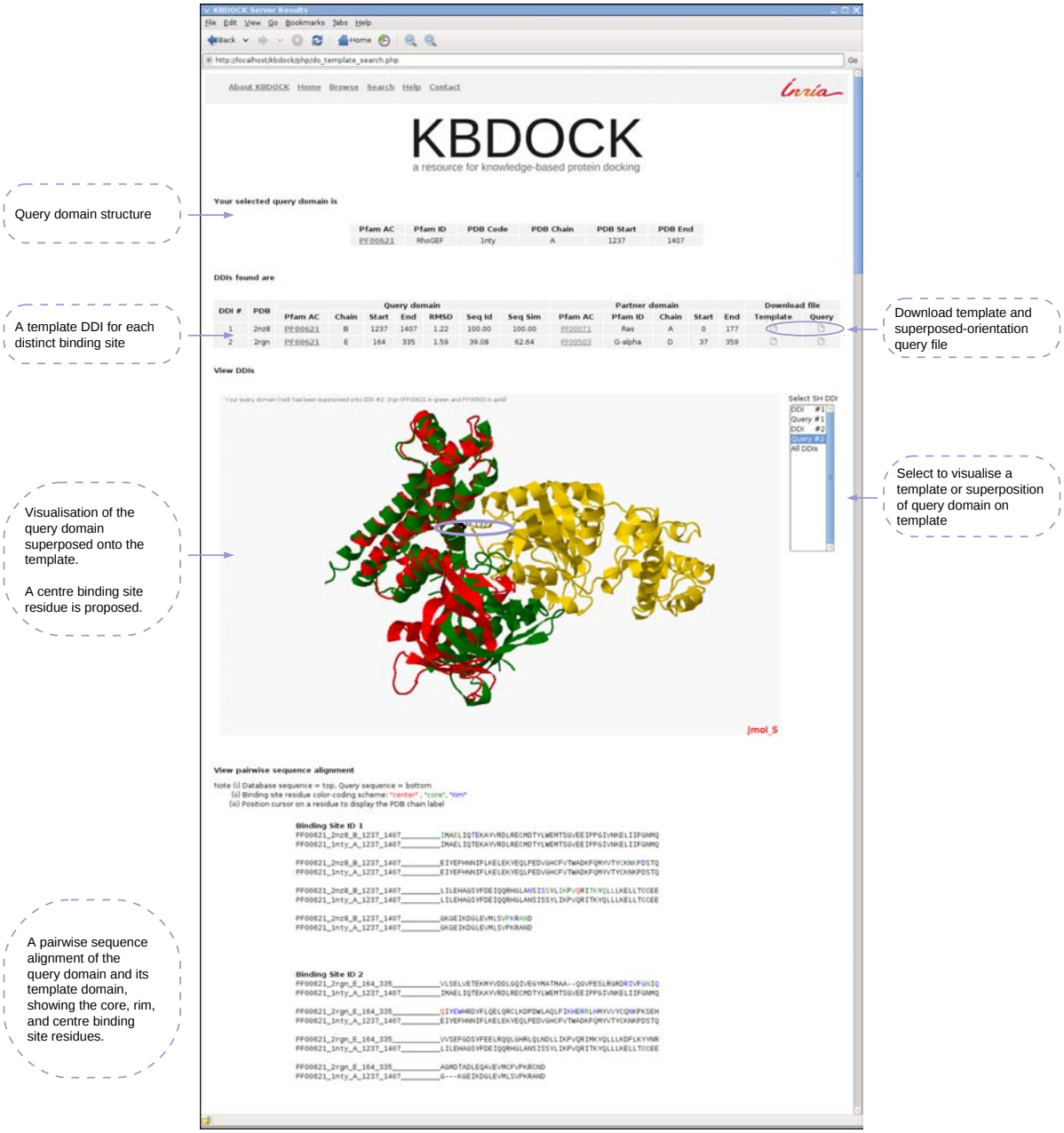


Figure A.4: A screenshot showing the KBDOCK results page when querying KBDOCK with a single domain structure to propose DFBSs.

A.5 Finding Docking Templates

As discussed in previous chapters, a protein complex may often be successfully modelled using the known binding sites of homologous domains. Given two query domain structures, KBDOCK searches for full-homology (FH) DDIs involving the same Pfam families as the query domains. The query domains are then superposed onto the FH template(s) in order to propose a docking model of the complex. If several FH DDIs exist in the database and if they correspond to different pairs of binding sites, KBDOCK outputs a possible docking template for each distinct pair of binding sites.

On the other hand, if no FH templates are found, KBDOCK searches for and outputs semi-homology (SH) DDIs containing the individual query domains because these can still provide useful information for a docking calculation. In these cases, the query domain is superposed onto each template in turn in order to propose a binding site on the query domain. If several SH templates are found for a given query domain, KBDOCK selects as a template the domain with the highest sequence similarity to the query. The overall approach is illustrated schematically in Figure A.5.

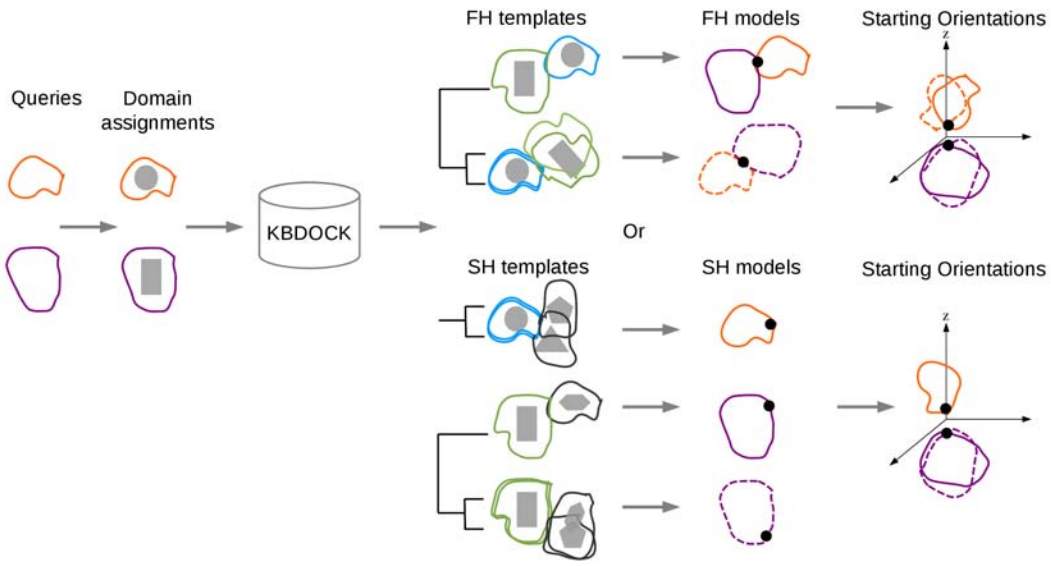


Figure A.5: Schematic illustration of how KBDOCK processes a docking query. Here, dots represent the calculated binding site centre residues. When a FH template is available, KBDOCK superposes the query domains onto the template to propose a docking model. When only SH templates exist, KBDOCK proposes one or more binding site(s) on each query domain. For each selected template, KBDOCK calculates the core, rim, and binding site centre residues.

A.5.1 Full-Homology Templates

To find docking templates, the user enters two PDB codes or uploads two PDB files and he then specifies which pair of Pfam domains in those structures should be used as queries. Currently, the KBDOCK web server supports only one pair of query domains. If KBDOCK finds one or more FH DDI templates for the query domains, it shows a Jmol view of the superposed query and FH

template(s), along with colour-coded sequence alignments of the query and template domains as described before. As before, the user may download the query and template structures for later analysis. Figure A.6 shows a screenshot of the results page obtained when querying KBDOCK using PDB codes 1thf and 1k9v (which correspond to the *His_biosynth* and *GATase* Pfam domains, respectively). According to KBDOCK, these two domains interact via a single interaction mode. Hence, KBDOCK proposes only one FH template. In this example, the best FH template (selected using overall sequence identity) is PDB code 1gpw (an imidazole glycerol phosphate synthase).

A.5.2 Semi-Homology Templates

Even if no pair-wise DDI homologues can be found, KBDOCK can often propose binding sites for the individual domains, which can be useful for conventional docking calculations. Hence, if no FH DDIs exist in the database, KBDOCK will output in a similar way a non-redundant list of SH templates and their colour-coded sequence alignments along with a Jmol view of the superpositions. Figure A.7 shows a screenshot of the results page obtained when querying KBDOCK using PDB codes 1o3y and 1oxz (which correspond to the *Arf* and *GAT* Pfam domains, respectively). Here, KBDOCK found SH DDIs for both query domains. The *Arf* domain family has three DFBSs. Hence, KBDOCK proposes three DDI templates, each corresponding to a distinct binding site according to our spatial clustering algorithm. According to KBDOCK, the *GAT* domain family has one DFBS only.

A.6 Focused Docking Using Hex

In Section 6.7, we described how the template DDI and DFBS information retrieved from the KBDOCK database can be used to set up a focused docking calculation. Hence, it is natural to connect the KBDOCK server with a protein docking server to avoid the user to download and upload data from one server to another. To start with, we link the KBDOCK server with the Hex docking server (Macindoe *et al.*, 2010). The Hex docking server provides web access to the Hex docking program (Ritchie and Kemp, 2000).

Even though the KBDOCK server can currently provide DDI templates only for single-domain proteins, multiple DFBSs on each query structure can be still specified for focused docking by Hex because Hex can perform all-versus-all docking of multiple PDB structures (see Figure A.5 and Section 6.7 for details). For example, in Section A.5.2, the pair of query structures involves the *Arf* and *GAT* domains, which have three and one DFBS, respectively. Hence, the user can select which DFBS to use on the *Arf* domain as shown in Figure A.7a. In addition, the user can specify the Hex docking search range angle as well as other Hex docking parameters. Based on the user selection, the KBDOCK server prepares a PDB model structure file for each query domain and a Hex script containing all the selected parameter values. The KBDOCK server sends the docking job to the Hex server which provides a link to the Hex results page (Figure A.7b). The user can then download the Hex predictions and view them in any visualisation tool.

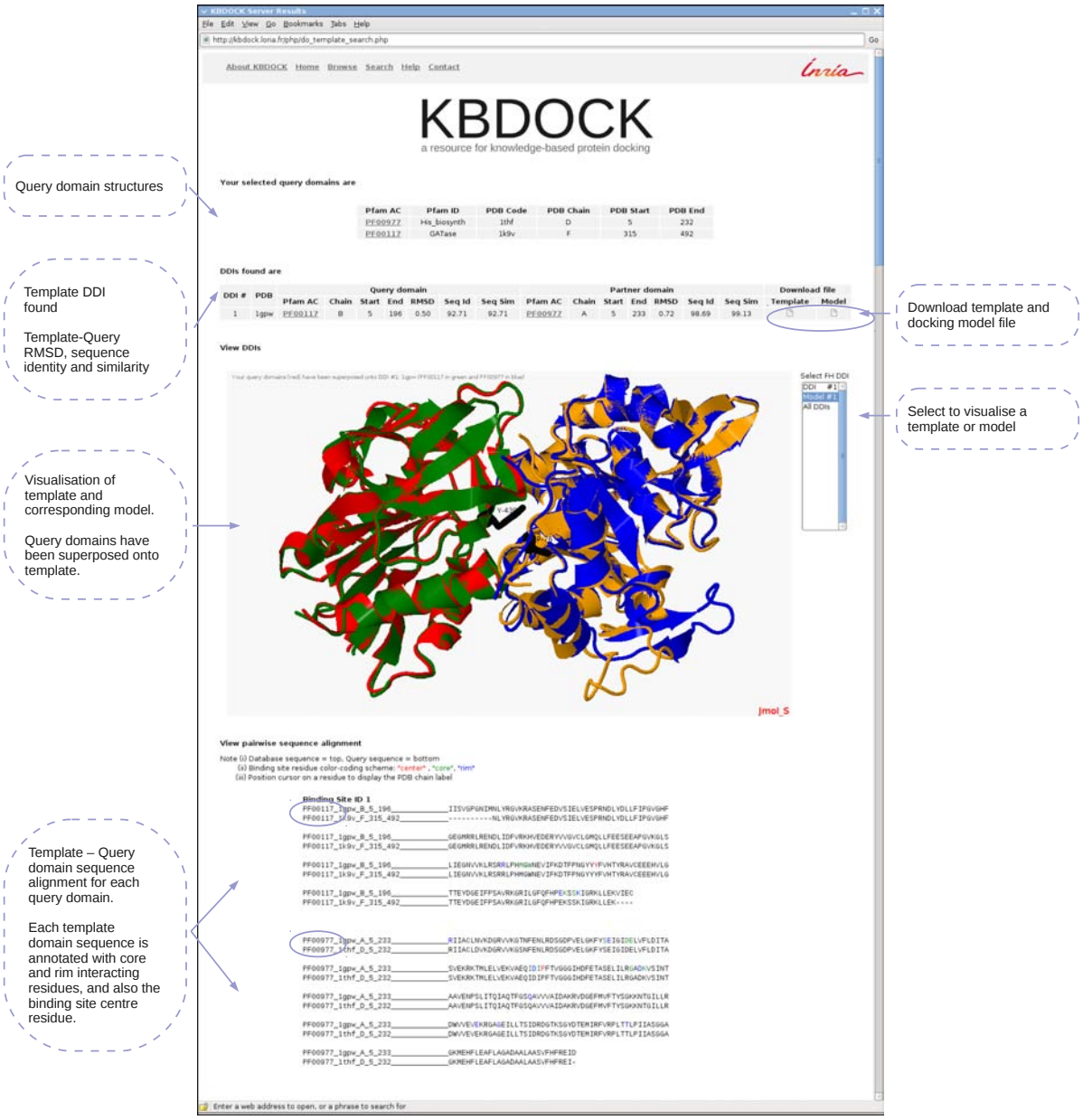


Figure A.6: A screenshot showing (a) the selected two domains for which a docking model is requested, (b) a Jmol view of the resulting FH template with the query domains superposed onto the template, (c) sequence alignments of the query and template domains highlighting the template core, rim, and centre binding site residues.

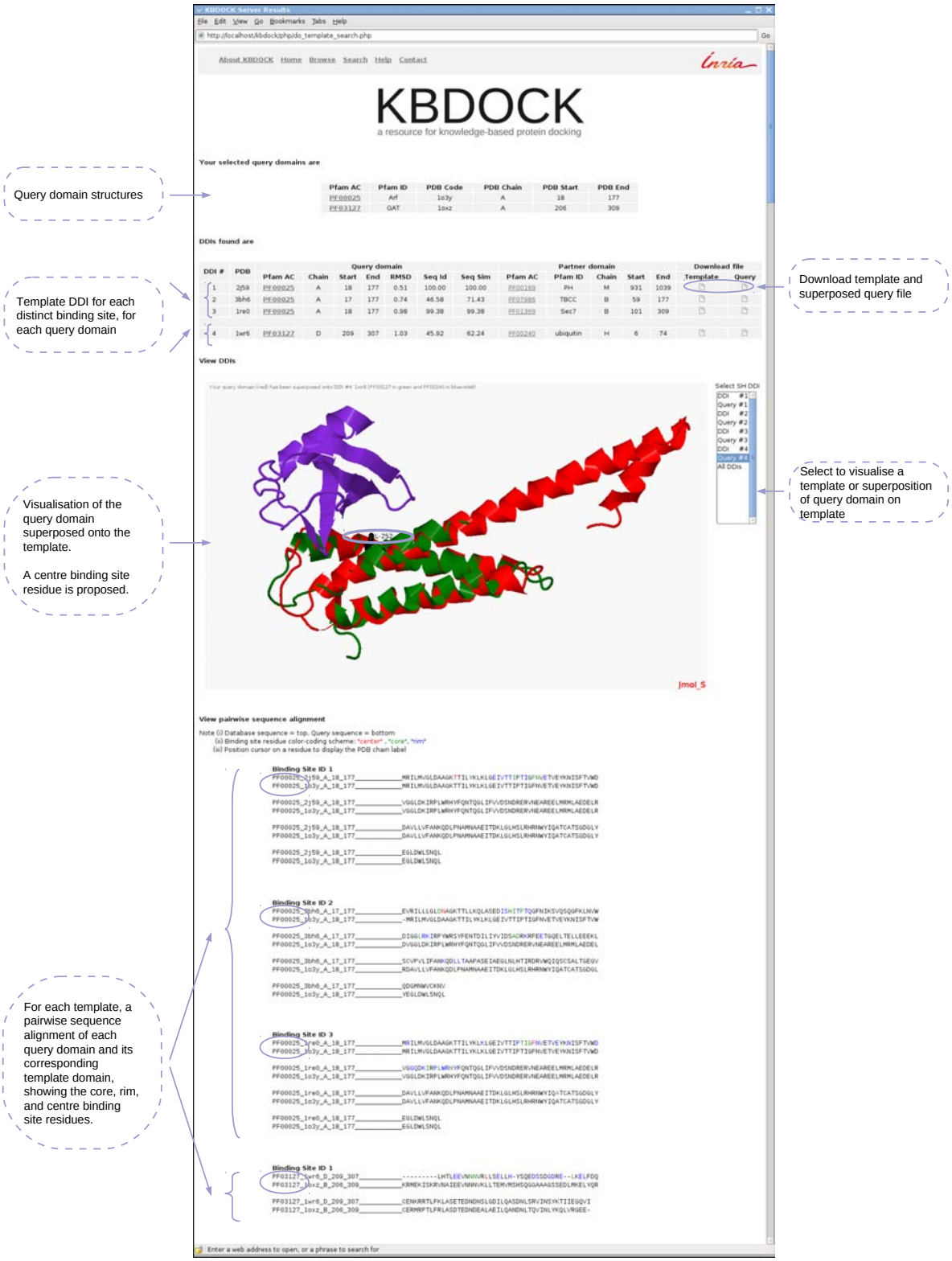


Figure A.7: A screenshot showing the KBDOCK results page when SH templates are found for both query domains.

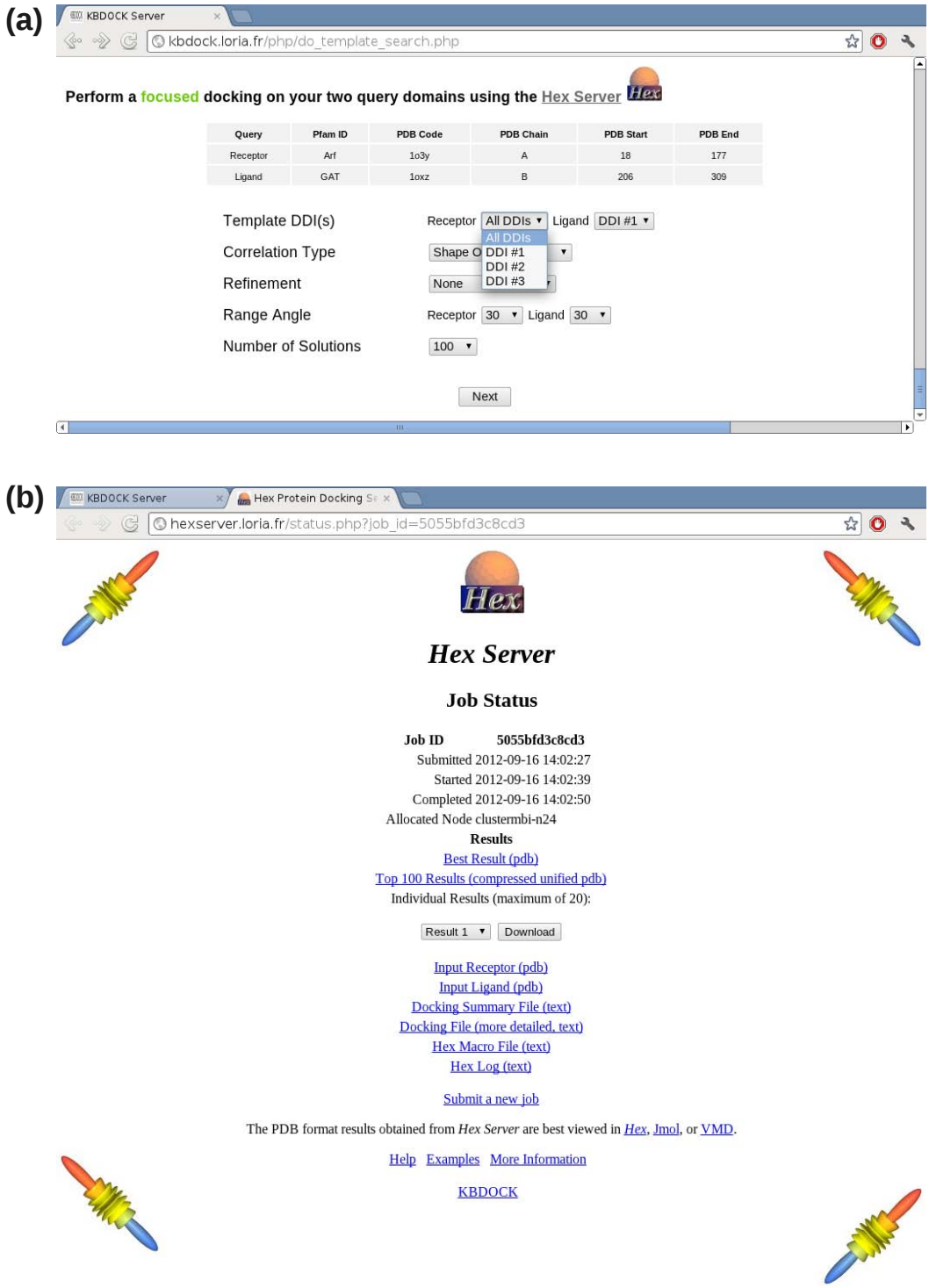


Figure A.8: A screenshot showing (a) the form where the user selects the DFBS to focus the docking calculations, and (b) the Hex docking results page.

Appendix B

Published Article

B.1 Spatial Clustering of Protein Binding Sites for Template-Based Protein Docking

The following article was published in *Bioinformatics* (impact factor 5.5) in August 2011. It has already received three citations (Google Scholar).

BIOINFORMATICS ORIGINAL PAPER Vol. 27 no. 20 2011, pages 2820–2827
doi:10.1093/bioinformatics/btr493

Structural bioinformatics

Advance Access publication August 27, 2011

Spatial clustering of protein binding sites for template based protein docking

Anisah W. Ghoorah[1], Marie-Dominique Devignes[2], Malika Smaïl-Tabbone[3] and David W. Ritchie[1,*]

[1]INRIA, [2]CNRS and [3]Nancy Université, Orpailleur Team, LORIA, Campus Scientifique, BP 239, 54506 Vandoeuvre-lès-Nancy, France

Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: In recent years, much structural information on protein domains and their pair-wise interactions has been made available in public databases. However, it is not yet clear how best to use this information to discover general rules or interaction patterns about structural protein–protein interactions. Improving our ability to detect and exploit structural interaction patterns will help to provide a better 3D picture of the known protein interactome, and will help to guide docking-based predictions of the 3D structures of unsolved protein complexes.

Results: This article presents KBDOCK, a 3D database approach for spatially clustering protein binding sites and for performing template-based (knowledge-based) protein docking. KBDOCK combines residue contact information from the 3DID database with the Pfam protein domain family classification together with coordinate data from the Protein Data Bank. This allows the 3D configurations of all known hetero domain–domain interactions to be superposed and clustered for each Pfam family. We find that most Pfam domain families have up to four hetero binding sites, and over 60% of all domain families have just one hetero binding site. The utility of this approach for template-based docking is demonstrated using 73 complexes from the Protein Docking Benchmark. Overall, up to 45 out of 73 complexes may be modelled by direct homology to existing domain interfaces, and key binding site information is found for 24 of the 28 remaining complexes. These results show that KBDOCK can often provide useful information for predicting the structures of unknown protein complexes.

Availability: http://kbdock.loria.fr/

Contact: Dave.Ritchie@inria.fr

Supplementary Information: Supplementary data are available at *Bioinformatics* online.

Received on June 11, 2011; revised on August 5, 2011; accepted on August 22, 2011

1 INTRODUCTION

Protein–protein interactions (PPIs) are central to many cellular processes. Proteins often perform their function by interacting with other proteins to form protein–protein complexes. In order to understand and predict PPIs reliably, and to relate such interactions to biological function, knowledge of the 3D structures of protein–protein complexes is vitally important. To date, over 65 000 protein structures have been deposited in the Protein Data Bank (PDB; Berman *et al.*, 2002). However, it has been estimated recently that only ∼12% of these structures correspond to heteromeric complexes (Stein *et al.*, 2011). Therefore, to bridge this gap, there is much interest in developing computational techniques to predict how two proteins fit together to form a complex (Aloy *et al.*, 2005). However, until recently, many of the hetero complexes in the PDB have been enzyme-inhibitor complexes, which are relatively easy to model directly. Hence, so-called template-based protein docking has not yet attracted much attention from the research community (Kundrotas *et al.*, 2008).

Since it is well known that protein folds are often more evolutionarily conserved than their sequences (Chothia and Lesk, 1986), and since it has been shown that proteins with similar sequences often interact in similar ways (Aloy *et al.*, 2003), it follows that close structural homologues should also be expected to interact in similar ways. Several studies have found that the locations of protein interaction sites are often conserved, especially within domain families, regardless of the structures of their binding partners (Gunther *et al.*, 2007; Keskin *et al.*, 2005; Korkin *et al.*, 2005, 2006). Additionally, it has also been observed that many protein families employ only one or a small number of binding sites (Keskin and Nussinov, 2007; Shoemaker *et al.*, 2006), suggesting that the same surface patch is often re-used. Indeed, it has been demonstrated previously that the structure of an unknown protein complex may often be successfully modelled using the known binding sites of homologous domains (Kundrotas *et al.*, 2008; Launay and Simonson, 2008). This may be described as template-based docking or docking by homology (Korkin *et al.*, 2006; Kundrotas and Alexov, 2006).

In recent years, much structural information on protein domains and on PPIs has been made available in on-line databases (Tuncbag *et al.*, 2009). However, beyond listing the residues observed at the interface between a given pair of proteins or protein domains, there is no generally accepted way to define what actually constitutes a protein binding site or to quantify whether or not two binding sites are structurally similar. For example, recent methods to compare structural interfaces have used techniques based on e.g. geometric hashing of cliques of interface Cα atoms (Keskin *et al.*, 2004), combining geometric hashing with a physicochemical complementarity scoring function (Shulman-Peleg *et al.*, 2004), geometric overlap and face angle scores of residue contact vectors (Kim *et al.*, 2006), principal component analysis of residue contact matrices (Aung *et al.*, 2008), complete linkage hierarchical clustering of groups of interface residues (Stein *et al.*, 2010) and dynamic programming-based fragment assembly (Gao and Skolnick, 2010).

Some recent examples of structural PPI databases are PIBASE (Davis and Sali, 2005), SCOPPI (Winter *et al.*, 2006), SCOWLP (Teyra *et al.*, 2006), 3D Complex (Levy *et al.*, 2006), 3D-partner (Chen *et al.*, 2007), PiSite (Higurashi *et al.*, 2009), IBIS (Shoemaker *et al.*, 2010) and 3DID (Stein *et al.*, 2010). Several of these databases describe PPIs in terms of domain–domain interactions (DDIs) because protein domains may often be identified as structural and functional units. Three widely used domain definitions are Pfam (Finn *et al.*, 2010), SCOP (Murzin *et al.*, 1995) and CATH (Cuff *et al.*, 2009). Pfam defines domain using sequence similarities, while the SCOP and CATH domain definitions are based on both sequence and structural similarities. Current structural PPI databases clearly constitute useful bioinformatics resources. However, it is not yet straightforward to use them to extract general patterns that describe the spatial nature of PPIs at the family level. Furthermore, because the question of how best to compare and cluster protein interfaces remains an open problem, it is not yet clear how best to use structural databases to propose suitable templates for homology-based docking predictions.

Here we present KBDOCK, a 3D database approach for spatially clustering protein binding sites and for performing template-based (knowledge-based) protein docking. KBDOCK combines residue contact information from the 3DID database with the Pfam protein domain family classification and protein coordinate data from the PDB in order to superpose and spatially cluster all known hetero DDIs for each Pfam family. The main features that distinguish KBDOCK from existing structural PPI databases are that: (i) it uses the Pfam consensus sequence to guide structural alignments; (ii) it places all of the complexes involving a given Pfam domain family into a common coordinate frame in order to locate the interaction partners consistently in 3D space; (iii) it uses the notion of 'core' and 'rim' interface residues to help define the geometric centre of a binding site; (iv) for each domain of interest, it spatially clusters a weighted combination of the core and rim interface residues of all DDIs involving that domain in order to define domain family binding sites; (v) it may be used to identify automatically the best available DDI template to use to model by homology a complex of two given domains; and (vi) even when no suitable DDI template exists, it can still propose candidate binding sites on one or both interaction partners as potential constraints for computational docking. Thus, KBDOCK represents a novel knowledge-based approach for proposing structural templates for protein docking.

Our approach is illustrated using 10 example query domains, each having multiple hetero interactions that may be clustered into a small number of domain family binding sites. The utility of the approach for template-based protein docking is demonstrated using 73 complexes from the Protein Docking Benchmark (version 4). Overall, up to 45 out of 73 complexes may be modelled by direct homology to existing domain interfaces, and key binding site information is found for 24 of the 28 remaining complexes. There are only four targets for which no homologous hetero DDIs exist. These results show that KBDOCK can often provide useful information for predicting the structures of unknown protein complexes.

*To whom correspondence should be addressed.

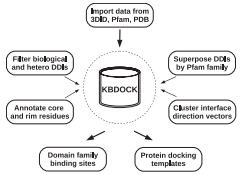


Fig. 1. Overview of the main KBDOCK data sources and processing steps.

2 METHODS

2.1 Overview of KBDOCK

KBDOCK is built from three main data sources. Multiple sequence alignments and consensus sequences are provided by Pfam, residue contact data and Pfam domain assignments are extracted from the 3DID database, and protein coordinates are obtained from the PDB. We use 3DID as our source of DDIs because it uses the Pfam classification to describe domains, and because it is one of the most complete and up-to-date structural PPI databases currently available. 3DID considers an interface to exist between two domains whenever five or more contacts (hydrogen bonds, electrostatic, or van der Waals atomic interactions) exist between the two domains. This means that 3DID contains both permanent and transient DDIs which may arise from both biologically relevant and non-biological (i.e. crystal contact) interactions. The version of 3DID used here (November 2009) contains a total of 140 612 DDIs drawn from 29 922 PDB structures. A total of 3755 different Pfam families are involved in at least one DDI.

The KBDOCK database is implemented using the MySQL relational database (http://www.mysql.com). All calculations and queries against the database are made using a small set of Prolog programs (http://www.swi-prolog.org/) and R scripts (http://www.r-project.org/). A web interface (http://kbdock.loria.fr) has been implemented using the PHP scripting language (http://php.net) and the Jmol plug-in for visualization (http://jmol.sourceforge.net). Figure 1 summarizes the processing steps used to populate the KBDOCK database. These are described in further detail below. The current version of KBDOCK stores Pfam domain family binding site information for a total of 2721 non-redundant (NR) hetero DDIs involving 1029 Pfam domain families. A MySQL dump of the database is available from the authors on request.

2.2 Selecting non-redundant hetero DDIs

Although the 3DID database stores all known DDIs, our main goal is to predict the 3D structures of heteromeric PPIs, as these are often the most difficult structures to solve experimentally (Ezkurdia *et al.*, 2009). Therefore, for each protein domain present in 3DID, all DDIs involving that domain are extracted and classified as either 'intra', 'homo' or 'hetero'. We consider a DDI to be intra if the interacting domains belong to a single protein chain, and homo if the interacting domains belong to different instances of the same protein chain in a given PDB structure. Otherwise, the interaction is considered to be hetero. Figure 2 illustrates these types of domain interactions schematically. Here, only hetero DDIs are considered further, although in principle the approach could also be used to model homo dimers.

Next, non-biological hetero interactions are filtered out. It has been shown that biological interactions usually have larger interfacial areas than non-biological interactions (Janin and Rodier, 1995). Hence, we use the DSSP program (Kabsch and Sander, 1983) to calculate the solvent accessible surfaces (SASs) buried within each domain interface. If a given domain has multiple interactions with other identical domains, e.g. due to crystal packing,

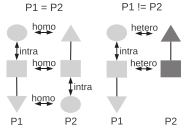


Fig. 2. Schematic illustration of the different types of DDI that may occur between two protein chains, P1 and P2. Protein chains can contain one or more domains connected by linker regions (straight lines). Each shape (circle, rectangle, triangle) represents a different Pfam domain. Lines with arrowheads represent DDIs.

we assume that the interaction with the largest buried SAS corresponds to the biological interaction, and only this DDI is retained.

It is also important to detect and eliminate duplicate or near-duplicate DDIs that may arise in other ways. For example, the same protein complex might have been solved under different crystallographic conditions, or a single crystal structure can sometimes contain different copies of the same complex. In order to deal with such cases, the sequences of the DDI partners are concatenated, and the NRDB90 program (Holm and Sander, 1998) is used with a similarity threshold of 99% to collect a final list of distinct NR DDIs.

It is worth noting that because we consider every structure to be useful, a high similarity threshold is used in order to retain as many non-duplicate structures as possible. This does not introduce any bias because here binding sites are defined by spatial clustering and not by counting residue frequencies.

2.3 Annotating DDI interfaces

As discussed above, the residues in PPI sites are often conserved across domain families. Indeed, due to evolutionary pressure, active site residues are often less likely to undergo mutation than other residue positions (Zvelebil *et al.*, 1987), and this phenomenon has been exploited previously to predict molecular interaction sites (Aytuna *et al.*, 2005; Lichtarge *et al.*, 1996). Therefore, the representative sets of hetero DDIs stored in KBDOCK are annotated with both 1D sequence information from Pfam and 3D structure information calculated using DSSP, respectively. For each Pfam domain family, the Pfam database provides a multiple sequence alignment and a consensus sequence of all UniProt sequences belonging to that family. We follow the Pfam convention of considering a residue to be conserved if at least 60% of the amino acids at a given position in the multiple sequence alignment are of the same amino acid type. However, because Pfam uses UniProt sequences rather than PDB structures, and because PDB structures may contain gaps or unresolved regions, we align each PDB sequence with its Pfam/UniProt sequence in order to map every PDB residue to its corresponding Pfam consensus position. This mapping allows the 1D Pfam consensus information to be transferred to each PDB residue position.

In order to enhance the 1D domain family information with 3D interaction information, DSSP is used to calculate the change in solvent accessibility for each interaction residue (as defined by 3DID) between the separate and complexed structures of each domain. Here, we use the notion of 'core' and 'rim' residues, as defined by Chakrabarti and Janin (2002). An interaction residue is considered to be a core interface residue if it loses at least 75% of its accessible surface area on going from the isolated to the complexed structure. Otherwise, it is considered to be a rim interface residue.

2.4 Defining protein domain family binding sites

Our mapping between the Pfam consensus sequence and PDB residue numbers provides a convenient way to identify the conserved residue positions of all domains stored in KBDOCK. Hence, it is straightforward to retrieve the C_α coordinates of the conserved residue positions in a given domain of interest along with the structures of all of the corresponding DDI partners, and to place these in a common coordinate frame using the ProFit (http://bioinf.org.uk) least-squares fitting program.

Superposing all the DDIs involving a given Pfam domain in this way provides a straightforward way to cluster individual binding sites and to identify automatically distinct PPIs in 3D space. For example, for each superposed DDI, the centre of mass, $\underline{C}$, of each binding site is calculated as a weighted average of the corresponding core (75%) and rim (25%) C_α coordinates. By also calculating the all-atom centre of mass $\underline{D}$ for each domain, an interface direction vector, $\underline{V}$, may then be calculated as

$$\underline{V} = (\underline{C} - \underline{D})/|\underline{C} - \underline{D}|. \quad (1)$$

In order to define domain family binding sites automatically, we cluster the dimensionless interface vectors using Ward's hierarchical clustering algorithm (Ward, 1963). This is illustrated in Supplementary Figure S1. From visual inspection of several example interfaces, we find that a clustering threshold of 0.4 often gives acceptable clusters.

2.5 Finding docking templates

In order to predict the 3D interaction between a pair of proteins, we need to query the database with two or more domains and to calculate the intersection of the results. This broadly corresponds to calculating a spatial join in a conventional relational database. Although one PPI can involve several DDIs, for simplicity only pair-wise DDIs are considered here. This leads to four possible outcomes, namely that the database is found to contain DDIs involving (i) both query domains together; (ii) both domains individually; (iii) just one domain; or (iv) neither domain.

In the first case, which we call a full homology (FH) DDI, the database DDI would be very likely to provide a good template with which to model the unknown interaction. The two query domains could be docked by homology simply by superposing them onto the FH template. If several such DDIs exist in the database, and if they correspond to different binding sites on the query domain(s), KBDOCK selects for each site the DDI with the highest overall sequence identity to the query domains. On the other hand, if homologous DDIs exist in the database for both of the query domains individually (case ii), it is reasonable to suppose that their binding sites might be re-used in the target complex, thus providing a rational way to initialize a more exhaustive computational docking calculation. Similarly, if just one of the target domains has known binding sites (case iii), these could still be used to constrain a computational docking run. These two cases may be termed docking by 'semi-homology' (SH) in analogy to the notion of a semi-join in relational algebra. In such cases, KBDOCK selects the best available homologous DDI for one or both query domains, as appropriate, and it identifies the residue(s) on the query domain(s) which lie closest to the centre of the corresponding binding site(s). These residue identities could then be used to define computational docking constraints. Clearly, if the database contains no homologous interactions, the target complex must be modelled by *ab initio* docking. However, as this study is primarily concerned with exploring a new knowledge-based approach for finding docking templates, the use of computational docking techniques is not considered here.

2.6 The Protein Docking Benchmark

In order to explore the utility of using KBDOCK to find homology templates for protein docking, our approach was used to predict a subset of the protein docking targets in version 4 of the Protein Docking Benchmark (Hwang *et al.*, 2010). The Docking Benchmark is a non-redundant expert-curated set of 176 protein complexes for which the bound complex structures, and most of the unbound component structures, have been solved by X-ray crystallography to a resolution of 3.25 Å or better. Since KBDOCK works at the domain level, we selected all single domain complexes belonging to the 'Enzyme-Inhibitor' (here called 'Enzyme') and 'Other' categories of the Docking Benchmark for this preliminary experiment. In other words, for simplicity we exclude the Benchmark 'Antibody' complexes (because apart from involving the antibody hypervariable loops, antibody–antigen interactions generally do not entail homology) and we exclude all other complexes involving multiple domains. This gives a test set of 36 Enzyme and 37 Other target complexes.

It should be noted that the Docking Benchmark complexes do not necessarily provide an unbiased set of homology modelling targets. Because several of the benchmark proteins have been relatively well studied, it is possible that the PDB could contain more homologues of those complexes than randomly selected complexes. In order to take into account this possible source of bias, a stringent test would be to exclude as templates all structures with more recent PDB deposition dates than the target structure. However, filtering complexes by target date often excludes a large proportion of the database. Therefore, in order to provide upper and lower bounds on the utility of template-based modelling, and to try to quantify the growing usefulness of knowledge-based approaches, we report results both with and without date filtering.

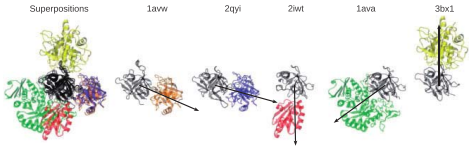


Fig. 3. This figure shows the superpositions and interface direction vectors [Equation (1)] of the five DDIs of the *Kunitz legume* Pfam family. Here, the 1avw (porcine trypsin/soybean trypsin inhibitor) and 2qyi (bovine trypsin/trypsin inhibitor) complexes share a common binding site, and clearly have very similar interface vectors.

3 RESULTS

3.1 Defining domain family binding sites

Superposing families of related DDIs in a common coordinate frame and clustering their interface direction vectors [Equation (1)] provide a straightforward way to analyse structural relationships between the members of a given query domain. Figure 3 shows the superpositions and interface vectors calculated for the five DDIs involving the *Kunitz legume* Pfam family. This figure clearly shows that this domain has four distinct interaction sites, one of which is common to two different trypsin/inhibitor complexes. Supplementary Figure S1 shows the spatial clustering dendrogram for this family, and for a further three example Pfam families (namely, *Kunitz BPTI*, *Ribonuclease* and *Actin*).

Spatial clusters have been calculated and stored in KBDOCK for all the 1029 Pfam domain families which are involved in hetero interactions. Superposing and clustering all Pfam domain binding sites in KBDOCK takes ~8 CPU hours on a 64-bit 2.8 GHz Q9550 processor. Table 1 summarizes the number of hetero DDI partners and calculated binding sites for 10 example Pfam domain families, including the four examples considered above. This table shows that these Pfam domains typically have from one to four binding sites, according to our spatial clustering algorithm. It is interesting to note that even domains involved in many DDIs such as *Kunitz BPTI*, *Trypsin* and *Actin* still have only a relatively small number of distinct binding sites.

Figure 4 shows the DDI superpositions for the 10 Pfam families listed in Table 1. In most cases, visual inspection of the complexes in this figure readily confirms the calculated number of binding sites given in Table 1. For example, the *Potato inhibit* domain interacts with eight other domains (all serine proteases) using a single binding site. On the other hand, the *Kunitz BPTI* domain has two inhibitory binding sites, and, as shown in Table 1, the *Kunitz legume* inhibitor has four binding sites that form distinct interfaces with four different domain families, namely *Trypsin*, *Thioredoxin*, *Alpha-amylase* and *Peptidase S8*. Conversely, *Thioredoxin* interacts with eight different Pfam families, but it does so using just two overlapping binding sites.

Table 1. Summary of the number of DDIs and calculated binding sites for 10 example Pfam domains stored in KBDOCK

Pfam ID	Pfam name	Function	No. of DDIs	No. of binding sites
PF00197	Kunitz legume	Protease inhibitor	5	4
PF00014	Kunitz BPTI	Protease inhibitor	27	2
PF00280	Potato inhibit	Protease inhibitor	8	1
PF00089	Trypsin	Protease	98	6
PF00062	Lys	Hydrolase	10	5
PF00545	Ribonuclease	Hydrolase	9	1
PF00022	Actin	Protein binding	24	4
PF00059	Lectin C	Glycoprotein binding	14	4
PF00111	Fer2	Ferredoxin	14	3
PF00085	Thioredoxin	Redox protein	8	2

For domains that have multiple binding sites and which interact with several different domain partners (e.g. *Fer2*, *Lectin C*, *Lys*, *Actin* and *Trypsin*), it can be difficult to distinguish all the interactions visually. Hence, KBDOCK allows the user to select and display only those DDIs involving a given binding site. Comparing the DDIs of binding sites selected in this way using 3D graphical visualization software such as Jmol often shows that our clustering algorithm calculates acceptable clusters in almost all cases.

Figure 5 shows the distribution and the change with time of the number of binding sites per domain family (excluding the very large *C1-set* immunoglobulin domain family) of all NR hetero DDIs in KBDOCK. This figure confirms that most domains typically have from one to four hetero binding sites, and only a very small number of domains such as *Trypsin* (six binding sites) have more than this. Indeed, over 60% of all hetero domains in KBDOCK have just one binding site, which supports the notion that domain binding sites are often re-used in different DDIs. It is interesting to note that despite the growing number of Pfam domains for which KBDOCK contains

[15:20 22/9/2011 Bioinformatics-btr493.tex]
Page: 2822 2820–2827
[15:20 22/9/2011 Bioinformatics-btr493.tex]
Page: 2823 2820–2827

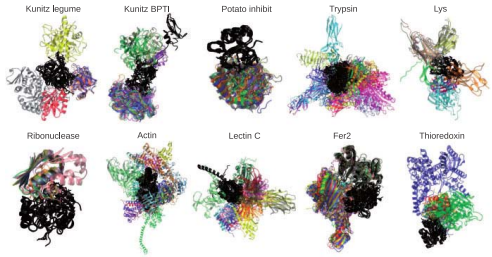


Fig. 4. DDI superpositions for 10 example Pfam domains (Table 1) in the coordinate frame of the query. In each case, the query domain is shown in black.

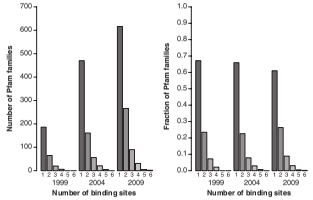


Fig. 5. The calculated number of hetero binding sites per domain family by PDB deposition date for all Pfam families except the *C1-set* immunoglobulin domains.

hetero complexes, the relative proportion of domains having 1, 2, 3, or 4 binding sites seems remarkably stable.

3.2 Docking by homology

Table 2 summarizes the results of querying KBDOCK to find docking templates for the 73 target complexes, both with and without filtering DDIs by PDB deposition date. Full details of these results are given in Supplementary Tables S1–S3. Supplementary Figure S2 shows two examples of docking targets for which KBDOCK finds FH and SH templates, respectively. For all targets, the structures of the unbound domains given by Hwang *et al.* (2010) were used as query domains, and the corresponding crystallographic complex (i.e. the expected solution) was excluded from the modelling procedure. Here, a FH template is considered to be correct if the root mean squared deviation (RMSD) between it and the native complex is <10 Å. This is similar to the CAPRI criteria for an 'acceptable' docking prediction (Mendez *et al.*, 2005). According to this criterion, Table 2 shows that KBDOCK finds good FH templates for a total of 24 out of 36 Enzyme target complexes, although this number falls to 13 when PDB deposition date filtering is applied. A further 10 targets have SH DDIs involving one or both of the target domains, and just two Enzyme targets have no hetero DDI information. Similarly, Table 2 shows that 21 (or 13 with date filtering) of the 37 Other targets may be modelled using FH templates, and a further 14 targets have SH DDIs involving one or both of the target domains. Like the Enzyme targets, only two of the Other targets have no hetero DDI information.

Table 2. Summary of the KBDOCK template modelling results for the 73 selected Protein Docking Benchmark targets[a]

Target class	Total targets	FH templates	Two SH templates	One SH template	No templates
No date filtering					
Enzyme	36	24/24	(3 + 1)/5	3/5	2
Other	37	21/21	(0 + 0)/3	5/11	2
Total	73	45/45	(3 + 1)/8	8/16	4
With date filtering					
Enzyme	36	13/13	(2 + 1)/5	7/11	7
Other	37	13/13	(0 + 0)/1	8/15	8
Total	73	26/26	(2 + 1)/6	15/26	15

[a]This table shows the number of docking targets for which the proposed templates are correct compared to the total number of templates retrieved. When SH templates are found for both domains individually ('Two SH templates'), the figures in brackets give the number of cases in which both binding sites are modelled correctly *plus* the number in which only one binding site is modelled correctly. Full details are presented in Supplementary Tables S1–S3.

Table 2 shows that all the retrieved FH templates are correct according to the 10 Å RMSD threshold. Thus, if KBDOCK retrieves a FH template, there is a high probability that it represents a good model of the target complex. However, Supplementary Table S1 shows that KBDOCK sometimes finds more than one distinct FH interface for a given pair of query domains. In other words, the instances of two domain families can sometimes interact via more than one combination of binding sites. For example, KBDOCK retrieves two FH templates for three of the Enzyme targets (1dfj, 1eaw, 2pcc) without date filtering, and for just one target (1eaw) when date filtering is applied. Subsequent visual inspection of the calculated templates for the matriptase/BPTI target (PDB code 1eaw) showed that the first trypsinogen/BPTI DDI (2r9p) provides a very good template (with an overall RMSD between the template and target of 0.79 Å), whereas the second (lower sequence identity) FH template corresponds to a different inhibitor orientation found in the prothrombin/boophilin complex (2ody; 8.54 Å RMSD).

Similarly, visual inspection of the calculated templates for two of the Other targets (1fqj, 1ml0) confirmed that these FH templates correspond to two different binding modes for both the G-protein complex (1fqj) and the M3-protein complex (1ml0), and that the first (highest sequence identity) template best matches the target (0.70 and 1.03 Å RMSD, respectively). On the other hand, the two DDIs (1z7x and 2bex) calculated for the large *RnaseA/LRR 1* Enzyme complex (1dfj) were seen to overlap considerably, and the two large binding sites calculated for the *RnaseA* domain should have been clustered as a single binding site. Similarly, two of the Other DDIs (1mq8, 2ayo) are calculated to have two distinct binding sites, although visual inspection again suggests that these should have been clustered as a single binding site. We believe that such clustering artefacts sometimes arise due to different assignments of core and rim residues in different instances of homologous DDIs. The peroxidase/cytochrome C Enzyme target (2pcc) is another interesting case. Although the two DDIs calculated for this target are quite distinct, further investigation revealed that one of the DDIs arises, because the crystal contact between these domains was larger than the biological contact in the 1s6v structure. Consequently, two binding sites instead of one were also calculated for these domains. Thus, KBDOCK can successfully retrieve alternate FH binding modes when they exist in the database, but it can also be seen that its clustering algorithm has a slight tendency to overestimate the number of distinct binding sites.

As might be expected, fewer FH templates are available when PDB data filtering is applied, and this causes an increase in the number of proposed SH templates. When only SH templates are retrieved, we assess their quality by comparing each proposed binding site with that of the native complex and if our interface clustering algorithm would group them together, we consider the retrieved template to be correct. The final two columns of Supplementary Tables S2 and S3 show the outcome of this test, and Table 2 summarizes the overall results. For example, SH DDIs involving the two individual query domains exist for five of the Enzyme targets. Supplementary Table S2 shows that three of these targets (1e6e, 1acb, 1f6m) may be modelled correctly by re-using their Pfam domain binding sites, and one further target (1ov8) may be partially modelled by re-using one of the two proposed SH templates. On the other hand, there are three Other targets for which the two query domains both have SH templates, but none of these lead to acceptable models. For those cases where only one SH template exists for a given target, the binding sites of one of the queries is found to be re-used in the target DDI in a total of three out of five Enzyme targets (1gl1, 4cpa, 1fq1) and five out of 11 Other targets (1ktz, 2g77, 1wq1, 2h7v, 1y64). In order to assist any subsequent computational docking calculation that might use these templates, Supplementary Tables S2 and S3 show the proposed PDB template along with the name of the query residue calculated to be at the centre of the binding site.

Overall, it can be seen that KBDOCK can provide high-quality FH docking templates for a total of 45 of the 73 targets (or 26/73 with date filtering). Even when no FH templates exist, KBDOCK can still find useful binding site information for at least one of the domain partners for 12 of the remaining 28 targets (or 18/47 with date filtering). These results demonstrate that the approach embodied in KBDOCK provides a useful way to find protein docking templates.

4 DISCUSSION

4.1 Comparison with previous approaches

Because the main aim of KBDOCK is to facilitate automatic docking by homology, it has several novel aspects that have not been explored in previous studies of structural PPIs. In particular, because protein docking is inherently a spatial problem (with six degrees of freedom in the simplest rigid body assumption), KBDOCK was designed from the start to consider the relative spatial arrangements of interacting protein domains, and to deal with cases where a full homology template is not necessarily available. This is in contrast to the most previous PPI classification approaches, which generally apply clustering techniques to groups of residues belonging to both partners of existing interfaces. For example, 3DID defines a domain interface by applying complete linkage hierarchical clustering to identify groups of shared interface residues within a Pfam domain family (Stein *et al.*, 2009). It then labels each distinct occurrence of a domain/interface pair as an 'interaction topology', and it applies a further round of hierarchical clustering to define 'global interface clusters' that group together individual interfaces (Stein *et al.*, 2010). This gives an average of about 10 global interfaces per Pfam domain [see Figure 3 of Stein *et al.* (2010)].

Kim *et al.* (2006) represent an interface as a pair of 'face vectors', each of which contains a list of ones and zeros to represent the contacting and non-contacting residues of each domain, respectively. The face vectors within a SCOP domain are then grouped according to the cosine similarity between their face vectors, and interfaces with similar faces are superposed and clustered according to their face overlap and the angle between the centroids of pairs of faces (Kim *et al.*, 2006; Winter *et al.*, 2006). Pairs of faces are then combined to define interface types. This procedure is reported to give on average about 5.4 distinct interface types per SCOP domain family (Winter *et al.*, 2006). Other approaches such as 3DID, PPiClust (Aung *et al.*, 2008) and I2I-SiteEngine (Shulman-Peleg *et al.*, 2004) also cluster and analyse pair-wise interfaces rather than individual binding sites. Clearly, the interface direction vectors used in KBDOCK share a similar inspiration to the face angle measure of Kim *et al.* (2006). However, Kim *et al.* (2006) focused on studying the diversity of domain interfaces, the evolution of hub proteins and gene fusion events (often manifested as intra-domain interactions), whereas our study focuses on finding docking templates for hetero domain interactions. Thus, our study complements and extends previous work.

Previous template-based docking approaches have used comparative patch analysis, threading and sequence alignment techniques (Chen and Skolnick, 2008; Korkin *et al.*, 2006; Kundrotas and Vakser, 2010; Kundrotas *et al.*, 2008; Launay and Simonson, 2008), for example. Hence, at a conceptual level, KBDOCK shares a similar inspiration with the comparative patch analysis approach of Korkin *et al.* (2006). This approach defines and clusters binding sites of interacting SCOP domains using a scalar 'localisation index' calculated as a sum of contact residue frequencies in the context of the superposed domains of a given

[15:20 22/9/2011 Bioinformatics-btr493.tex]
Page: 2824 2820–2827
[15:20 22/9/2011 Bioinformatics-btr493.tex]
Page: 2825 2820–2827

SCOP family (Korkin *et al.*, 2005). This index serves as a kind of fuzzy set membership measure, and does not consider the directional nature of the interface, whereas KBDOCK explicitly clusters binding sites according to the spatial orientation of their core and interface residues.

From a template docking point of view, KBDOCK is somewhat similar to the HOMBACOP approach of Kundrotas *et al.* (2008). HOMBACOP begins by using PSI-BLAST to identify candidate structural templates for a given pair of sequences, and these are refined using a further round of sequence-based template matching using a position-specific scoring matrix enriched with interface information of the known templates. In a similar spirit, Launay and Simonson (2008) use the solvent accessibility of interface residues to enhance their Needleman–Wunsch alignment of candidate templates. However, they then use an energy function to select the final template, whereas HOMBACOP uses sequence similarity and KBDOCK uses structural similarity to the target domains as the final selection criteria. Compared with HOMBACOP that used the PROTCOM database (Kundrotas and Alexov, 2007) without date filtering to produce 19 models for 43 targets (44%) from the Docking Benchmark version 2 (Mintseris *et al.*, 2005), KBDOCK finds good FH templates for 26 (36%) and 45 (62%) out of 73 targets with and without date filtering, respectively. Hence, KBDOCK appears to be rather competitive compared to the earlier approach.

Overall, KBDOCK provides high-quality FH docking templates for 62% of the targets studied here, and it finds useful binding site information for a further 39% (11/28) of the remaining targets. Following these very promising results, we are extending KBDOCK to deal with multi-domain complexes, and to link it directly to our rigid body docking software (Ritchie and Kemp, 2000).

4.2 Implications for the 3D interactome

There is growing interest in using docking techniques to predict large-scale structural PPIs (Kundrotas *et al.*, 2010; Launay and Simonson, 2008; Mosca *et al.*, 2009; Sinha *et al.*, 2008; Wass *et al.*, 2011). However, results from the CAPRI docking experiment (Lensink and Wodak, 2010) show that current docking algorithms still face the problem of how to distinguish a good solution from a list of feasible but mostly incorrect predicted docking orientations. On the other hand, exploiting biochemical or biophysical knowledge in data-driven docking (van Dijk *et al.*, 2005) can often help to constrain the scope of a docking calculation and considerably improve the quality of the results (Korkin *et al.*, 2006; Lensink and Wodak, 2010; Ritchie, 2008). Hence, if prior biological knowledge is available in a suitable form, it would be desirable to be able to incorporate it automatically in a docking calculation.

Kim *et al.* (2006) note that many of the currently known interface types only started to become available in the mid-1990s. Hence, early docking and interface studies only had a small repertoire of interface types to work with. They also found that although the number of interface types continues to grow, the rate of growth is currently much less than the growth in the total number of multi-domain structures that are being solved (Figure 5 of Kim *et al.*, 2006). Our analysis of the rate of growth in the number of hetero binding sites since 1999 (Fig. 5) also shows only a modest increase in the number of Pfam families having multiple hetero binding sites, despite over a 3-fold increase in the number of Pfam families for which hetero complexes are now available. This strongly supports the notion that protein binding sites are very often re-used. Of course, the hetero complexes available in the PDB are not necessarily representative of the whole structural interactome. Nonetheless, if the very small numbers of hetero protein binding sites found here do indeed turn out to be typical, this will have considerable implications for future data-driven and template-based docking approaches, and for populating 3D PPI networks on a genomic scale.

5 CONCLUSION

KBDOCK provides a systematic way to store and analyse the 3D structures of protein domain binding sites. By superposing the structures of all hetero DDIs involving a given query domain, and by using the simple notion of an interface direction vector to define the central region a protein binding site, a small number of spatially distinct binding sites may be identified for each Pfam domain family. Using this approach, we find that the majority of the 1029 Pfam domain families have a small number (up to four) of hetero binding sites, and over 60% have just one hetero binding site.

KBDOCK can be used to find automatically homologous hetero DDIs with which to model the unknown 3D structure of given protein complex. In 60% of the docking benchmark examples studied, KBDOCK finds a small number of high quality DDI templates with which to model the target complex. Furthermore, one of the unique strengths of KBDOCK is that it can find semi-homologous templates even when no full homology template is available. Hence, KBDOCK provides a useful knowledge-based approach for template-based protein docking and for helping to describe and understand structural PPIs on a genomic scale.

ACKNOWLEDGEMENT

We thank Matthieu Chavent for useful discussions.

Funding: Agence Nationale de la Recherche, grant reference ANR-08-CEXC-017-01.

Conflict of Interest: none declared.

REFERENCES

Aloy,P. *et al.* (2003) The relationship between sequence and interaction divergence in proteins. *J. Mol. Biol.*, **332**, 989–998.

Aloy,P. *et al.* (2005) Protein complexes: structure prediction challenges for the 21st century. *Curr. Opin. Struct. Biol.*, **15**, 15–22.

Aung,Z. *et al.* (2008) PPIClust: efficient clustering of 3D protein-protein interaction interfaces. *J. Bioinformatics Comput. Biol.*, **6**, 415–433.

Aytuna,A.S. *et al.* (2005) Prediction of protein-protein interactions by combining structure and sequence conservation in protein interfaces. *Bioinformatics*, **21**, 2850–2855.

Berman,H.M. *et al.* (2002) The protein data bank. *Acta Crystallogr. Sect. D Biol. Crystallogr.*, **58**, 899–907.

Chakrabarti,P. and Janin,J. (2002) Dissecting protein-protein recognition sites. *Proteins Struct. Funct. Genet.*, **47**, 334–343.

Chen,H. and Skolnick,J. (2008) M-TASSER: an algorithm for protein quaternary structure prediction. *Biophys. J.*, **94**, 918–928.

Chen,Y.C. *et al.* (2007) 3D-partner: a web server to infer interacting partners and binding models. *Nucleic Acids Res.*, **35**, W561–W567.

Chothia,C. and Lesk,A.M. (1986) The relation between the divergence of sequence and structure in proteins. *EMBO J.*, **5**, 823–826.

Cuff,A.L. *et al.* (2009) The CATH classification revisited–architectures reviewed and new ways to characterize structural divergence in superfamilies. *Nucleic Acids Res.*, **37**, D310–D314.

Davis,F.P. and Sali,A. (2005) PIBASE: a comprehensive database of structurally defined protein interfaces. *Bioinformatics*, **21**, 1901–1907.

Ezkurdia,I. *et al.* (2009) Progress and challenges in predicting protein-protein interaction sites. *Brief. Bioinformatics*, **10**, 233–246.

Finn,R.D. *et al.* (2010) The Pfam protein families database. *Nucleic Acids Res.*, **38**, D211–D222.

Gao,M. and Skolnick,J. (2010) iAlign: a method for the structural comparison of protein-protein interfaces. *Bioinformatics*, **26**, 2259–2265.

Gunther,S. *et al.* (2007) Docking without docking: ISEARCH – prediction of interactions using known interfaces. *Proteins Struct. Funct. Bioinformatics*, **69**, 839–844.

Higurashi,M. *et al.* (2009) PiSite: a database of protein interaction sites using multiple binding states in the PDB. *Nucleic Acids Res.*, **37**, D360–D364.

Holm,L. and Sander,C. (1998) Removing near-neighbour redundancy from large protein sequence collections. *Bioinformatics*, **14**, 423–429.

Hwang,H. *et al.* (2010) Protein-protein docking benchmark version 4.0. *Proteins Struct. Funct. Bioinformatics*, **78**, 3111–3114.

Janin,J. and Rodier,F. (1995) Protein-protein interaction at crystal contacts. *Proteins*, **23**, 580–587.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure - pattern-recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**, 2577–2637.

Keskin,O. and Nussinov,R. (2007) Similar binding sites and different partners: implications to shared proteins in cellular pathways. *Structure*, **15**, 341–354.

Keskin,O. *et al.* (2004) A new, structurally nonredundant, diverse data set of protein-protein interfaces and its implications. *Protein Sci.*, **13**, 1043–1055.

Keskin,O. *et al.* (2005) Hot regions in protein-protein interactions: the organization and contribution of structurally conserved hot spot residues. *J. Mol. Biol.*, **345**, 1281–1294.

Kim,W.K. *et al.* (2006) The many faces of protein-protein interactions: a compendium of interface geometry. *PLoS Comput. Biol.*, **2**, 1151–1164.

Korkin,D. *et al.* (2005) Localization of protein-binding sites within families of proteins. *Protein Sci.*, **14**, 2350–2360.

Korkin,D. *et al.* (2006) Structural modeling of protein interactions by analogy: application to PSD-95. *PLoS Comput. Biol.*, **2**, e153.

Kundrotas,P.J. and Alexov,E. (2006) Predicting 3D structures of transient protein-protein complexes by homology. *BBA Proteins Proteomics*, **1764**, 1498–1511.

Kundrotas,P.J. and Alexov,E. (2007) PROTCOM: searchable database of protein complexes enhanced with domain-domain structures. *Nucleic Acids Res.*, **35**, D575–D579.

Kundrotas,P.J. and Vakser,I.A. (2010) Accuracy of protein-protein binding sites in high-throughput template-based modeling. *PLoS Comput. Biol.*, **6**, e1000727.

Kundrotas,P.J. *et al.* (2008) Homology-based modeling of 3D structures of protein-protein complexes using alignments of modified sequence profiles. *Int. J. Biol. Macromol.*, **43**, 198–208.

Kundrotas,P.J. *et al.* (2010) GWIDD: genome-wide protein docking database. *Nucleic Acids Res.*, **38**, D513–D517.

Launay,G. and Simonson,T. (2008) Homology modelling of protein-protein complexes: a simple method and its possibilities and limitations. *BMC Bioinformatics*, **9**, 427.

Lensink,M.F. and Wodak,S.J. (2010) Docking and scoring protein interactions: CAPRI 2009. *Proteins Struct. Funct. Bioinformatics*, **78**, 3073–3084.

Levy,E.D. *et al.* (2006) 3D complex: a structural classification of protein complexes. *PLoS Comput. Biol.*, **2**, 1395–1406.

Lichtarge,O. *et al.* (1996) An evolutionary trace method defines binding surfaces common to protein families. *J. Mol. Biol.*, **257**, 342–358.

Mendez,R. *et al.* (2005) Assessment of CAPRI predictions in rounds 3–5 shows progress in docking procedures. *Proteins Struct. Funct. Bioinformatics*, **60**, 150–169.

Mintseris,J. *et al.* (2005) Protein-protein docking benchmark 2.0: An update. *Proteins Struct. Funct. Bioinformatics*, **60**, 214–216.

Mosca,R. *et al.* (2009) Pushing structural information into the yeast interactome by high-throughput protein docking experiments. *PLoS Comput. Biol.*, **5**, e1000490.

Murzin,A.G. *et al.* (1995) SCOP – a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.

Ritchie,D.W. (2008) Recent progress and future directions in protein-protein docking. *Curr. Protein Peptide Sci.*, **9**, 1–15.

Ritchie,D.W. and Kemp,G.J.L. (2000) Protein docking using spherical polar Fourier correlations. *Proteins Struct. Funct. Genet.*, **39**, 178–194.

Shoemaker,B.A. *et al.* (2006) Finding biologically relevant protein domain interactions: conserved binding mode analysis. *Protein Sci.*, **15**, 352–361.

Shoemaker,B.A. *et al.* (2010) Inferred biomolecular interaction server-a web server to analyze and predict protein interacting partners and binding sites. *Nucleic Acids Res.*, **38**, D518–D524.

Shulman-Peleg,A. *et al.* (2004) Protein-protein interfaces: Recognition of similar spatial and chemical organizations. *Proc. Algorithms Bioinformatics*, **3240**, 194–205.

Sinha,R. *et al.* (2008). Docking by structural similarity at protein-protein interfaces. *Proteins Struct. Funct. Bioinformatics*, **78**, 3235–3241.

Stein,A. *et al.* (2009) 3did update: domain-domain and peptide-mediated interactions of known 3D structure. *Nucleic Acids Res.*, **37**, D300–D304.

Stein,A. *et al.* (2010) 3did: identification and classification of domain-based interactions of known three-dimensional structure. *Nucleic Acids Res.*, **39**, D718–D723.

Stein,A. *et al.* (2011) Three-dimensional modeling of protein interactions and complexes is going 'omics. *Curr. Opin. Struct. Biol.*, **21**, 200–208.

Teyra,J. *et al.* (2006) SCOWLP: a web-based database for detailed characterization and visualization of protein interfaces. *BMC Bioinformatics*, **7**, 104.

Tuncbag,N. *et al.* (2009) A survey of available tools and web servers for analysis of protein-protein interactions and interfaces. *Brief. Bioinformatics*, **10**, 217–232.

van Dijk,A.D. *et al.* (2005) Data-driven docking for the study of biomolecular complexes. *FEBS J.*, **272**, 293–312.

Ward,J.H. (1963) Hierarchical grouping to optimize an objective function. *J. Am. Stat. Assoc.*, **58**, 236–244.

Wass,M.N. *et al.* (2011) Towards the prediction of protein interaction partners using physical docking. *Mol. Syst. Biol.*, **7**, 469.

Winter,C. *et al.* (2006) SCOPPI: a structural classification of protein–protein interfaces. *Nucleic Acids Res.*, **34**, D310–D314.

Zvelebil,M.J. *et al.* (1987) Prediction of protein secondary structure and active sites using the alignment of homologous sequences. *J. Mol. Biol.*, **195**, 957–961.

Appendix C

Manuscripts in Preparation

C.1 A Systematic Structure-Based Classification and Analysis of Protein Domain Family Binding Sites

The following article was submitted to the prestigious and highly competitive (13% acceptance rate) conference on Intelligent Systems for Molecular Biology (ISMB) 2012 and it received positive feedback from the reviewers. We are planning to submit a revised manuscript elsewhere as a full journal article.

BIOINFORMATICS

Vol. 00 no. 00 0000
Pages 1–7

A systematic structure-based classification and analysis of protein domain family binding sites

Anisah W. Ghoorah,[1,3] Marie-Dominique Devignes,[2] Malika Smaïl-Tabbone,[3] and David W. Ritchie [1]*

[1]INRIA, [2]CNRS, [3]Nancy Université
LORIA, Campus Scientifique, BP 239, 54506 Vandoeuvre-lès-Nancy, France

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: There is considerable interest in understanding how experimentally observed protein-protein interaction (PPI) networks can be explained in terms of 3D structural interactions. There is a rapidly growing number of 3D structures which have been solved. However it is not yet clear how best to understand and exploit knowledge of the 3D interactions between these structures. Here, we classify and analyse the secondary structural features of a comprehensive and non-redundant set of representative DDIs drawn from 140,612 structural DDIs provided by the 3DID database. Our dataset consists of 947 Pfam domain families involving a total of 1,439 hetero domain binding sites and 1,009 distinct domain family interactions.

Results: We have classified Pfam domain binding sites into seven structural types based on different combinations of three secondary structural elements (SSEs). We find that α-α domain interactions and irregular-irregular interactions are quite probable, whereas α-β and β-β interactions are rather strongly disfavoured. Furthermore, we find there is very little difference between the SSE character of single partner binding sites and multi-partner binding sites. However, two-partner binding sites are found to have significantly smaller surface areas than single partner binding sites. Additionally, we find that over 80% of Pfam domain binding sites interact with just one other type of Pfam domain, and very few domain binding sites interact with more than three different Pfam domains. Although our analysis can successfully identify known hub proteins such as the protein kinase domain (and can possibly explain their promiscuity), we find that most domains are primarily monogamous in their physical relationships with other domains, and this finding could have considerable implications for the interpretation of large-scale PPI networks and for drug targeting.

Availability: http://kbdock.loria.fr/

Contact: Dave.Ritchie@inria.fr

1 INTRODUCTION

Protein-protein interactions (PPIs) are central to many biological processes. At the three-dimensional (3D) structural level, proteins often perform their function by interacting with other proteins to form protein-protein complexes. These complexes may consist of homo-dimers or higher order homo-multimers, or they may involve heteromeric interactions between different protein chains. While homo-interactions are observed relatively often in crystal structures, most processes of biological interest involve hetero interactions, and the corresponding structures are normally much more difficult to determine experimentally and to predict computationally (Ezkurdia *et al.*, 2009). Hence, although the number of solved 3D protein structures appears to be growing exponentially (Berman, 2008), there is an equally growing need to be able to classify and analyse the structural repertoire of known hetero PPIs using computational modeling and analysis techniques.

Following the principle of homology, it is often assumed that proteins with similar 3D structures will have similar biological functions. Several studies have shown that proteins often interact via just one or a small number of binding sites (Shoemaker *et al.*, 2006; Keskin and Nussinov, 2007; Ghoorah *et al.*, 2011), and several groups have studied the biophysical properties of protein binding sites and interfaces. For recent reviews, see e.g., de Vries and Bonvin (2008), Keskin *et al.* (2008), Ezkurdia *et al.* (2009), and Fernandez-Recio (2011). For example, Keskin *et al.* (2008) summarise the factors which have been examined by several groups in order to characterise protein binding sites. These include, residue conservation, the types of secondary structures present, the shape and surface area of the binding site, the number of water molecules buried on binding, the number of polar and non-polar residues, the number of available hydrogen bonds and salt bridges, and the presence of so-called "hot spot" residues, for example.

In one of the earliest structural studies of PPIs, Janin and Chothia (1990) observed that the binding sites of protease inhibitors and antibody proteins often have similar structural properties. On the other hand, Jones and Thornton (1996) found that many interfaces have roughly equal proportions of helix, sheet, and loop residues, with some interfaces containing only one type of secondary structure, but most being mixed. Lo Conte *et al.* (1999) found that the size of the recognition site is related to the conformational changes. Xu *et al.* (1997) suggest that hydrogen bonds are weaker at the interface and that salt bridges play a significant role in protein association. Other studies based on alanine scanning mutagenesis have shown that certain called hot spot interface residues can contribute a large proportion to the total binding energy (Bogan and Thorn, 1998), and that large amino acids such as tryptophan, arginine and tyrosine are the frequent hot spot residues. Ma *et al.* (2003) found that hot spot residues such as tryptophan, and to a lesser extent phenylalanine and methionine, are more structurally conserved in PPI interfaces than in other surface regions, and they proposed that this tendency might be used to help predict the locations of unknown binding sites. Caffrey *et al.* (2004) found that the residues at protein interfaces are usually more conserved than other surface residues, particularly in homodimers, but they found that such differences are not sufficient to predict interface patches by conservation alone. Thus, even though some general trends have been observed, it is not clear how best to use this knowledge to differentiate and classify different types of binding sites. Indeed, we recently argued that there is still no universally accepted definition of what actually constitutes a protein binding site *per se* (Ghoorah *et al.*, 2011).

In order to relate the structure and function of different proteins in a systematic way, PPIs are often described in terms of domain-domain interactions (DDIs) because protein domains may often be identified as structural and functional units. Three widely used domain definitions are Pfam (Finn *et al.*, 2010), SCOP (Murzin *et al.*, 1995), and CATH (Cuff *et al.*, 2009). Pfam defines domains using multiple sequence alignments in order to identify families of sequences which will often correspond to distinct functional and structural regions. The SCOP and CATH classifications use both sequence and structural similarities to collect and relate protein domains in a hierarchical system of related domain families.

It should be noted that both the SCOP and CATH classifications can contain many instances of a given Pfam domain family, and that DDIs may be described and analysed both in terms of individual pairs of 3D domain structures (i.e. one instance of a DDI) and in terms of interactions at the domain family level (i.e. one or more instances of interactions between two Pfam domain families which have been observed experimentally).

In a previous study we defined protein domain family binding sites (DFBSs) using spatial clustering of hetero DDIs (Ghoorah *et al.*, 2011). Here, we analyse the secondary structural features of DFBSs themselves and within domain family interactions (DFIs) using our collection of 947 different Pfam domain families involving a total of 1,439 domain family binding sites (DFBSs) and 1,009 distinct domain family interactions (DFIs) which we derived from the 3DID database Stein *et al.* (2010). Hence, the present work represents one of the largest systematic studies of structural domain interactions to have been described to date and, to our knowledge, the first study to have considered quantitatively the nature of such interactions at the domain family level.

Here, we show that hetero domain binding sites can be clustered into seven main groups, each consisting of different proportions of the main secondary structure elements (SSEs), namely helices, sheets, and irregular structures. We therefore propose a structural classification of domain binding sites which naturally extends the top level of the CATH domain family classification (i.e., α, β, $\alpha+\beta$, and irregular), as originally defined by Levitt and Chothia (1976). We use our binding site classification to determine whether there are any general relationships between the SSEs of pairs of binding sites, and to investigate whether there are any differences in the structural features of domain binding sites which have just one domain partner (the majority) and those with more than one domain partner.

Our results confirm previous findings that α-helices are found more often at interfaces than β-sheets. More specifically, we find that α-α interactions and irregular-irregular interactions are quite probable, whereas α-β and β-β interactions are rather strongly disfavoured. On the other hand, we find there is very little difference, if any, between the SSE character of single partner binding sites and multi-partner binding sites. However, two-partner binding sites are found to have significantly smaller surface areas than single partner binding sites. Although our analysis can successfully identify known hub proteins such as the protein kinase domain, we find that most domains are primarily monogamous in their physical relationships with other domains, and this finding could have considerable implications for the interpretation of large-scale PPI networks and for drug targeting.

2 METHODS

2.1 The KBDOCK database

The KBDOCK database has been described previously (Ghoorah *et al.*, 2011). Briefly, KBDOCK combines DDI information from the November 2009 version of the 3DID database (Stein *et al.*, 2010) with the Pfam protein domain classification (Finn *et al.*, 2010) together with coordinate data from the Protein Data Bank (PDB; Berman *et al.*, 2002) for all known structural DDIs. The version of 3DID used here consists of a total of 140,612 DDIs drawn from 29,922 PDB structures and involving a total of 3,755 different Pfam domain families. However, many of these DDIs result from multiple copies of a given DDI appearing in a single crystal structure or from the same complex solved under different crystallographic conditions. Therefore, to achieve a robust classification and statistics, KBDOCK first filters the 3DID database to select DDIs involving only hetero interactions using a sequence similarity threshold of 99% for the concatenated sequences of each DDI. It then superposes and spatially clusters hetero DDIs in order to identify a small number of DFBSs for each Pfam domain (Ghoorah *et al.*, 2011). Finally, the DDI instances involving to each DFBS are filtered again, this time using a 60% sequence similarity threshold, in order to retain only distinct pairs of domains associated with any given DFBS. For example, 3DID has 23 DDIs for the Kunitz_legume domain (Pfam accession no. PF00197) which KBDOCK reduces to 5 non-redundant hetero DDIs, and which it then clusters spatially to identify 4 DFBSs on the Kunitz_legume domain. The 60% filter then reduces the 5 DDI instances to 4 representative DFIs. Overall, the KBDOCK filtering and clustering procedures give a total of 1,439 Pfam DFBSs located on 947 different Pfam domain families, and which are involved in a total of 1,009 distinct DFIs. The KBDOCK web server is available at *http://kbdock.loria.fr*. This provides an easy-to-use public interface to explore and analyse domain family interactions and their binding sites and to propose protein docking templates. A full dump of the database is available from the authors on request.

2.2 Structural annotation of DFBSs

We use the DSSP program (Kabsch and Sander, 1983) to annotate domain and DFBS residues with secondary structural information. DSSP defines eight types of SSE: α-helix (H), 3/10-helix (G), π-helix (I), residue in isolated β-bridge (B), extended strand (E), hydrogen bonded turn (T), bend (S), and loop/irregular (L). However, because several of these types are broadly quite similar,

*to whom correspondence should be addressed

and because only a few instances of turns and bends were found in the KBDOCK database, the eight DSSP types were grouped into three main SSE classes which we denote here as α (H, G and I), β (B and E), and γ (T, S, and L). The SSE propensity, $P_{f,b}(s)$, of each DFBS, (f, b), is calculated for each SSE class, s, as the average of the DSSP frequencies in the corresponding member domain binding sites:

$$P_{f,b}(s) = \frac{1}{M} \sum_{m=1}^{M} \frac{N^s_{m,b}}{N^\alpha_{m,b} + N^\beta_{m,b} + N^\gamma_{m,b}}, \qquad (1)$$

where M is the total number of non-redundant DDIs involving Pfam family f, and $N^s_{m,b}$ is the count of the number of residues of type s at the b^{th} binding site of the m^{th} DDI member of family f. Each SSE propensity value calculated in this way is automatically normalised to fall within the range $[0, 1]$.

2.3 Classifying and analysing DFBSs

In order to examine whether DFBSs might exhibit any preferred combinations of secondary structures, we first used Ward's hierarchical clustering algorithm (Ward, 1963), as implemented in the R software (*http://www.r-project.org*), to cluster the 1,439 DFBSs on the selected three classes of SSE. Visual inspection of the resulting dendrogram indicated that using 7 clusters would be the most parsimonious. However, because we believe hierarchical clustering is not necessarily the most robust clustering technique to use with smooth continuous functions, we next applied the expectation-maximization (EM) algorithm as implemented in the Weka data mining toolkit (*http://www.cs.waikato.ac.nz/ml/weka/*), to re-cluster the DFBSs using this number of target clusters. Weka's "JRip" propositional rule learning algorithm was then applied to generate a set of rules able to map DFBS instances to the selected clusters. In the following sections we refer to the clusters described by these rules as "binding site types." Finally, several specialised KBDOCK queries were implemented in Prolog to analyse the DFBSs and their interactions with respect to these binding site types. VMD (*http://www.ks.uiuc.edu/Research/vmd/*) was used to generate the molecular graphics figures.

3 RESULTS

3.1 Rule-based classification of DFBSs

Table 1 shows the mean and standard deviations (SDs) of the DFBS clusters obtained from EM clustering. Because these clusters are seen to describe biologically interesting combinations of SSE classes (e.g. "mainly α", etc.), and because each cluster has a broadly similar number of members, we adopted these clusters as a useful classification of the secondary structural composition of DFBSs. Visual inspection of Table 1 suggests that these clusters may be labeled as "α" (mainly α), "$\alpha + \gamma$" (approximately equal α and γ with almost no β), "$\beta + \gamma$" (mainly β plus some γ), "$\gamma + \alpha$" (mainly γ plus some α), "$\gamma + \beta$" (mainly γ plus some β), "γ" (nearly all γ), and "$\alpha + \beta + \gamma$" (approximately equal α, β, and γ). It is interesting to note that there is no specific "$\alpha + \beta$" type in this classification. Although binding sites containing both α and β SSEs are observed quite frequently (cluster 7, 161 instances), they always contain a considerable fraction of γ SSEs (average 34.6%). Indeed, Table 1 shows that each of the binding site types contains a significant γ component.

Figure 1 shows how our dataset of 1,439 DFBSs is distributed over the seven types of binding site. This figure confirms that helices and irregular SSEs are the most common types of SSE in domain binding sites. It is worth noting that despite the fact that a "mainly α" DFBS requires a considerably higher proportion of α SSEs than the proportion of β SSEs in a "$\beta + \gamma$" DFBS, yet Figure 1 still shows that the most common type of DFBS are those that involve α SSEs. Although Table 1 shows the observed SSE propensities, it does not provide a convenient way to classify a new instance. We therefore used JRip to generate a set of rules able to map DFBS instances to the selected clusters. These rules are able to classify correctly 96% of the DFBS instances. However, because some of these rules are rather complex, they were manually simplified by rounding each threshold to the nearest 5%. Table 2 shows the simplified rules obtained in this way.

Table 2. Simplified relationships between binding site SSE propensities (per cent units) and assigned binding site types.

Binding Site SSE Propensity Rule	Binding Site Type
$P(\alpha) >= 70$	α
$P(\beta) >= 45$	$\beta + \gamma$
$P(\beta) >= 20$ & $P(\alpha) <= 15$	$\gamma + \beta$
$P(\gamma) >= 80$	γ
$P(\alpha) >= 45$ & $P(\gamma) >= 35$	$\alpha + \gamma$
$P(\alpha) >= 20$ & $P(\gamma) >= 55$	$\gamma + \alpha$
Otherwise	$\alpha + \beta + \gamma$

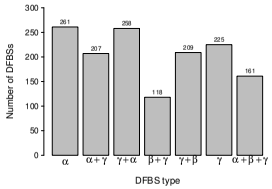


Fig. 1. The distribution of DFBSs over the 7 selected binding site types.

3.2 Do DFIs have SSE pairing preferences?

Figure 2 shows some examples of DDIs involving various associations of DFBS types. With 7 structural types of DFBS, there are $7 * 6/2 + 7 = 28$ possible pairs of binding site types. Table 3 lists the numbers of occurrences of DFIs for each pair of binding site types (total 1,009 DFIs). Given that the DFBS types are rather uniformly populated, with the exception of the $\beta + \gamma$ and $\alpha + \beta + \gamma$ types which are under-represented (Figure 1), a random distribution

Table 1. Mean and SDs (per cent units) of the SSE propensities for the 7 selected DFBS clusters.

Cluster	1	2	3	4	5	6	7
$P(\alpha)$	80.1 ± 11.5	53.7 ± 5.7	29.6 ± 8.2	4.5 ± 6.5	3.8 ± 5.4	4.3 ± 6.2	41.1 ± 16.6
$P(\beta)$	0.0 ± 0.0	0.4 ± 1.2	5.9 ± 6.8	61.7 ± 13.6	30.3 ± 9.4	2.6 ± 4.4	24.2 ± 13.2
$P(\gamma)$	19.3 ± 11.5	46.0 ± 5.8	64.5 ± 8.7	33.8 ± 13.9	65.9 ± 9.7	93.0 ± 7.4	34.6 ± 12.9
No. DFBSs	261	207	258	118	209	225	161

of our DFIs would give an average of about 36 DFIs (1009/28) per pair of DFBSs. Hence, Table 3 shows that the most frequent types of DFI consist of interactions between pairs of α (62/1009) and $\gamma + \alpha$ (55/1009), and between $\gamma + \alpha$ and γ (64/1000). Also frequent are DFIs involving $\gamma + \alpha$ with α, $\alpha + \gamma$, and $\gamma + \alpha$, giving 47, 55, and 59 interactions, respectively. DFIs in which $\gamma + \beta$ pairs with $\gamma + \beta$, or γ are also relatively frequent with 57 and 58 interactions, respectively. Thus, as might be expected from Table 1, the γ SSE is present in all frequent associations of binding site types.

In order to compare these frequencies more readily, Table 4 shows the marginal probabilities derived from Table 3. For example, if a given DFBS has been classified as α type, Table 4 shows that the probability that any partner of that domain will also have a mainly α binding site is 23% (compared to a random value of 100/7 $\simeq$ 14%). On the other hand, the probability that any partner of that domain will have at least some α SSEs is $23 + 19.7 + 17.4 + 8.2 = 68.3\%$. Similarly, if a given DFBS has been classified as $\beta + \gamma$, then the probability of observing $\beta + \gamma$ is only 9.6%. More generally, this table shows that interactions between pairs of α-rich DFBSs and also those between pairs of γ-rich DFBSs are quite probable, whereas α-β and β-β interactions are rather strongly disfavoured.

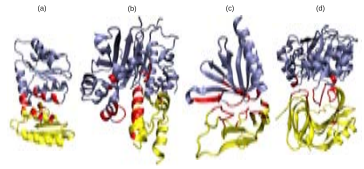


Fig. 2. Examples of different types of DFI: (a) α with α (PDB code 1VRC, chains A, C); (b) $\alpha + \gamma$ with α (PDB code 2CG5, chains A, B); (c) $\beta + \gamma$ with γ (PDB code 2YIB, chains F, B); (d) $\gamma + \alpha$ with $\gamma + \beta$ (PDB code 1TE1, chains A, B). Binding site SSEs are shown in red.

3.3 Are binding site surfaces special?

In a similar manner, we wished to investigate whether the mean SSE propensity of a domain binding site is different from the SSE propensity of the rest of the domain's accessible surface. Table 5 shows the marginal probabilities regarding the prediction

Table 3. The numbers of DFIs observed for each pair of binding site types.

	α	$\alpha+\gamma$	$\beta+\gamma$	$\gamma+\alpha$	$\gamma+\beta$	γ	$\alpha+\beta+\gamma$
α	62	53	21	47	33	32	22
$\alpha+\gamma$		30	24	55	22	33	26
$\beta+\gamma$			15	18	32	27	20
$\gamma+\alpha$				59	51	64	25
$\gamma+\beta$					57	58	35
γ						34	21
$\alpha+\beta+\gamma$							33

Table 4. The marginal probabilities (per cent units) of observing each type of "partner" binding site for a given "query" binding site.

	Partner						
Query	α	$\alpha+\gamma$	$\beta+\gamma$	$\gamma+\alpha$	$\gamma+\beta$	γ	$\alpha+\beta+\gamma$
α	23.0	19.7	7.8	17.4	12.3	11.6	8.2
$\alpha+\gamma$	21.8	12.3	9.9	22.6	9.1	13.6	10.7
$\beta+\gamma$	13.4	15.3	9.6	11.5	20.4	17.2	12.6
$\gamma+\alpha$	14.7	17.2	5.6	18.5	16.0	20.1	7.9
$\gamma+\beta$	11.5	7.6	11.1	17.7	19.8	20.1	12.2
γ	12.2	12.2	10.0	23.8	21.1	12.1	7.8
$\alpha+\beta+\gamma$	12.1	14.3	11.0	13.7	19.2	11.5	18.2

of a domain's binding site given knowledge of the SSE character of the rest of the domain's surface. Comparing the main diagonal and off-diagonal elements of this table suggests that the type of SSE in the binding site is well correlated with the SSE type of the domain's surface as a whole. In other words, there is often little or no significant difference between the SSE character of a domain's binding site and that of the rest of the domain's surface. However, it is worth noting that there seems to be a negative correlation for γ-rich surfaces, i.e., given a γ-rich domain surface it is rather unlikely that its binding site is also γ-rich.

3.4 Are multi-partner binding sites special?

We also wished to assess whether there are any significant differences between single partner binding sites and binding sites

Table 5. The marginal probabilities (per cent units) of observing a particular type of binding site with respect to the SSE type of the whole of the corresponding domain's surface.

	Binding Site						
Surface	α	$\alpha+\gamma$	$\beta+\gamma$	$\gamma+\alpha$	$\gamma+\beta$	γ	$\alpha+\beta+\gamma$
α	29.5	25.3	0.4	18.0	1.9	0.4	24.5
$\alpha+\gamma$	16.9	20.8	0.5	37.2	1.5	0.0	23.2
$\beta+\gamma$	1.9	1.4	42.6	17.7	6.7	0.5	29.2
$\gamma+\alpha$	6.6	11.2	10.9	39.2	0.8	2.7	28.7
$\gamma+\beta$	0.9	0.0	42.4	17.8	6.8	0.9	31.4
γ	3.1	6.2	24.9	35.6	3.1	9.3	21.8
$\alpha+\beta+\gamma$	4.3	3.1	13.7	29.2	1.2	0.0	48.5

that interact with more than one domain. Figure 3 shows the distribution of the number of distinct Pfam partners for both the 947 Pfam domain families and the 1,439 DFBSs stored in the KBDOCK database. This figure shows that some 62% (584/947) of these Pfam domains interact with just one type of domain partner, 21% interact with two types of domain, and only 17% interact with three or more different Pfam families. When considering DFBSs, the trend is even stronger, with over 80% (1,186/1,439) of DFBSs having just one type of domain partner. Only 17.5% (252/1,439) of DFBSs have more than one type of partner, and very few (in fact just 42) have more than three different domain partners.

In a similar manner, Table 6 shows the DFBS frequency and marginal probabilities of the various binding site types according to the number of their domain partners. This table indicates that multi-partner DFBSs tend to be slightly depleted in α-containing SSEs and richer in γ-containing SSEs, although these tendencies are not especially strong.

Table 7 lists the number of distinct Pfam domain partners for the 10 Pfam domains having the greatest numbers of DFBSs and domain partners. It is interesting to note that the Trypsin domain currently has the most interactions, presumably due to its rich variety of substrates and because interactions involving proteases have been heavily studied as therapeutic targets. As might be expected, most of the other domains such as Ras, PKinase, ubiquitin, V-set, and C1-Set are central to cell regulation, signaling, and the immune system, for example. Thus, the identity and function of the domains listed in Table 7 are rather consistent with their known function and with evidence from high throughput experiments. For example, Patil *et al.* (2010) report that kinase domains are frequently observed in PPI network hubs, with some 405 hubs having some kind of kinase activity.

Finally, in order to estimate whether there are any gross physical differences between DFBSs with just one binding site partner and those with more than one partner, we used DSSP to calculate the solvent accessible surface (SAS) of each DFBS (these calculations assume that each binding site contributes equally to the buried SAS at a DDI interface). Table 8 shows the average surface areas and number of binding site residues of DFBSs according to the number of Pfam partner domains. This table suggests that smaller DFBSs tend to have *more* interaction partners. Applying a Wilcoxon signed rank test to the difference in size between single-partner and two-partner DFBSs shows that this difference is statistically significant at the 5% level ($p = 0.014$). However, there are too few instances of three-partner DFBSs to confirm this trend statistically.

Table 6. Marginal probability comparison of binding site type with respect to the number of Pfam partners.

DFBSs	DFIs	α	$\alpha+\gamma$	$\beta+\gamma$	$\gamma+\alpha$	$\gamma+\beta$	γ	$\alpha+\beta+\gamma$
1186	1	18.8	14.3	8.8	16.8	13.6	16.1	11.6
159	2	15.1	13.2	4.4	25.8	18.9	12.6	10.1
52	3	19.2	25.0	3.9	11.5	17.3	17.3	5.8
42	>3	9.5	7.1	11.9	28.6	21.4	9.5	9.5

Table 7. The numbers of distinct Pfam domain partners for the 10 Pfam domains having the greatest numbers of DFBSs.

Pfam domain	Partners per DFBS					Total Partners
Peptidase S8	1	5				6
Cytochrome C1	1	3	5			9
Ubiquitin conjugating enzyme	1	2	2	5		10
Photosynthetic reaction centre	2	3	3	5		13
Protein kinase domain	2	4	4	5		15
Immunoglobulin C1-set	1	1	5	6	7	20
Ubiquitin	2	2	3	6	9	22
Immunoglobulin V-set	5	6	10	10		31
Ras 4	2	7	8	8	11	36
Trypsin	3	4	5	14	15	41

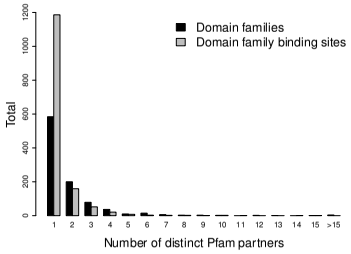


Fig. 3. Histogram of the number of different Pfam domain partners by Pfam domain and by DFBS.

Table 8. Average DFBS sizes calculated with respect to the number of their Pfam domain partners.

DFBSs	DFIs	Binding Site SAS (Å^2)	Binding Site Residues
1186	1	709 ± 560	15 ± 11
159	2	576 ± 377	12 ± 7
52	3	559 ± 340	12 ± 7
42	>3	670 ± 253	14 ± 6

4 DISCUSSION

There is considerable interest in understanding how experimentally observed PPI networks can be explained in terms of 3D structural interactions (Tsai *et al.*, 2009; Patil *et al.*, 2010). However, it is difficult to rationalise or consolidate the results from different groups because different investigators often use different terminologies to describe different types of interaction (e.g. homo *vs* hetero, transient *vs* permanent, obligate *vs* non-obligate, biological *vs* non-biological contact, date hub *vs* party hub, sociable *vs* non-sociable, "singlish" interface *vs* multi-interface). Such variety of language highlights the need for a well-defined terminology and methodology with which to describe, classify, and analyse the structural nature of PPIs and protein binding sites.

As a natural extension of the CATH domain classification, we have classified domain binding sites into seven SSE-based types, and we have proposed simple rules with which to classify new instances of binding sites. This classification was possible thanks to the well-characterized and non-redundant set of binding sites that we identified and stored previously in KBDOCK using our spatial clustering algorithm (Ghoorah *et al.*, 2011). Because KBDOCK is built from 3DID, which we consider to be the largest and most up-to-date DDI database, our SSE-based classification of binding sites may be considered as the most comprehensive one to date.

Our three SSE propensity "coordinates" (α, β, γ) are compatible with those of the top level of the CATH domain family classification. However, the continuous nature of these coordinates mean that any derived rules may need to evolve as new DDIs are added to the structural interactome. It should also be emphasised that a CATH class does not necessarily entail structural homology. Similarly, our classification of binding sites does not imply any kind of structural homology. It aims only to provide a practical framework in which to describe different combinations of domain binding site SSEs in a way that reflects well the observed SSE propensities. Nevertheless, for present purposes, we believe that the very large coverage and the lack of redundancy of our dataset make our classification quite reliable.

Several previous studies of structural PPIs have identified potentially interesting relationships between the shapes and physical properties of protein-protein interfaces, and most such studies have suggested that it might be possible to use such properties predictively. However, these earlier studies of PPIs have been somewhat limited by the relatively small numbers of hetero protein-protein complexes available, and by the problem of how to select a suitable sub-set of protein binding sites to work with. The study described here is based on a set of 1,009 DFIs between a total of 947 different Pfam domains, representing the first systematic study of DDIs at the Pfam domain family level.

As well as supporting previous findings regarding the preference for α-rich features in binding sites, our analysis indicates that irregular-irregular interactions are also favoured, but that interactions rich in α-β and β-β are rather strongly disfavoured. Knowledge of these secondary structure pairing propensities could be useful for the prediction of unknown DDIs, especially if combined with other near-orthogonal physical properties (de Vries and Bonvin, 2008).

Perhaps more significantly, our results show that some 60% of domain families and 80% of the DFBSs for which 3D structural DDI information is available interact with just one type of Pfam domain, and that very few DFBSs interact with more than three different Pfam domains. Interestingly, these DFBSs are always found in domains containing more than one DFBS (see Table 7). This suggests there may exist a "two-level multiplication" in the number of different partners for these domains. This multiplication of partners could be further enhanced when such multiple-partner domains are assembled in multi-domain protein architectures. Thus, such a combinatorial expansion of partners could explain the structural basis for the observed promiscuity of some hub proteins. We are planning future analyses to verify whether the small set of multi-partner DFBSs and their corresponding domains are actually involved in the architecture of these proteins.

5 CONCLUSION

Our structural classification of DFBSs provides a useful way to classify and analyse the secondary structure propensities of DDIs, and it highlights some SSE pairing preferences which might be useful for the prediction of unknown DDIs. We have used this classification to analyse the structural interactions of a large set of 1,439 domain family binding sites located on 947 Pfam domain families, and involving 1,009 distinct hetero domain interactions. We find that α-α interactions and irregular-irregular interactions are rather frequent, whereas α-β and β-β interactions are rather strongly disfavoured. We find there is very little difference, if any, between the SSE character of single partner binding sites and multi-partner binding sites. However, two-partner binding sites are found to have statistically significantly smaller surface areas than single partner binding sites. Additionally, we find that over 80% of those Pfam domains for which 3D structural DDI information is available interact with just one other type of Pfam domain, and that very few domain binding sites interact with more than three different Pfam domains. This indicates that most domains are primarily monogamous in their physical relationships with other domains. Hence, the results from this early glimpse of the structural interactome could have considerable implications for the interpretation of large-scale PPI networks and for drug targeting.

ACKNOWLEDGEMENT

Part of this work was funded by the Agence Nationale de la Recherche, grant reference ANR-08-CEXC-017-01.

REFERENCES

Berman, H. M. (2008). The Protein Data Bank: a historical perspective. *Acta Crystallographica*, **A38**, 88–95.

Berman, H. M., Battistuz, T., Bhat, T. N., Bluhm, W. F., Bourne, P. E., Burkhardt, K., Iype, L., Jain, S., Fagan, P., Marvin, J., Padilla, D., Ravichandran, V., Schneider, B., Thanki, N., Weissig, H., Westbrook, J. D., and Zardecki, C. (2002). The protein data bank. *Acta Crystallographica Section D-Biological Crystallography*, **58**, 899–907.

Bogan, A. A. and Thorn, K. S. (1998). Anatomy of hot spots in protein interfaces. *Journal Of Molecular Biology*, **280**(1), 1–9.

Caffrey, D. R., Somaroo, S., Hughes, J. D., Mintseris, J., and Huang, E. S. (2004). Are protein-protein interfaces more conserved in sequence than the rest of the protein surface? *Protein Science*, **13**(1), 190–202.

Cuff, A. L., Sillitoe, I., Lewis, T., Redfern, O. C., Garratt, R., Thornton, J., and Orengo, C. A. (2009). The CATH classification revisited-architectures reviewed and new ways to characterize structural divergence in superfamilies. *Nucleic Acids Research*, **37**, D310–D314.

de Vries, S. J. and Bonvin, A. M. J. J. (2008). How proteins get in touch: Interface prediction in the study of biomolecular complexes. *Current Protein & Peptide Science*, **9**(4), 394–406.

Ezkurdia, I., Bartoli, L., Fariselli, P., Casadio, R., Valencia, A., and Tress, M. L. (2009). Progress and challenges in predicting protein-protein interaction sites. *Briefings in Bioinformatics*, **10**(3), 233–246.

Fernandez-Recio, J. (2011). Prediction of protein binding sites and hot spots. *Wiley Interdisciplinary Reviews-Computational Molecular Science*, **1**(5), 680–698.

Finn, R. D., Mistry, J., Tate, J., Coggill, P., Heger, A., Pollington, J. E., Gavin, O. L., Gunasekaran, P., Ceric, G., Forslund, K., Holm, L., Sonnhammer, E. L. L., Eddy, S. R., and Bateman, A. (2010). The Pfam protein families database. *Nucleic Acids Research*, **38**, D211–D222.

Ghoorah, A. W., Devignes, M.-D., Smail-Tabbone, M., and Ritchie, D. W. (2011). Spatial clustering of protein binding sites for template based protein docking. *Bioinformatics*, **27**(20), 2820–2827.

Janin, J. and Chothia, C. (1990). The structure of protein-protein recognition sites. *The Journal of biological chemistry*, **265**(27), 16027–30.

Jones, S. and Thornton, J. M. (1996). Principles of protein-protein interactions. *Proceedings Of The National Academy Of Sciences Of The United States Of America*, **93**(1), 13–20.

Kabsch, W. and Sander, C. (1983). Dictionary of protein secondary structure - pattern-recognition of hydrogen-bonded and geometrical features. *Biopolymers*, **22**(12), 2577–2637.

Keskin, O. and Nussinov, R. (2007). Similar binding sites and different partners: Implications to shared proteins in cellular pathways. *Structure*, **15**, 341–354.

Keskin, O., Gursoy, A., Ma, B., and Nussinov, R. (2008). Principles of protein-protein interactions: What are the preferred ways for proteins to interact? *Chemical Reviews*, **108**(4), 1225–1244.

Levitt, M. and Chothia, C. (1976). Structural patterns in globular proteins. *Nature*, **261**(5561), 552–8.

Lo Conte, L., Chothia, C., and Janin, J. (1999). The atomic structure of protein-protein recognition sites. *Journal Of Molecular Biology*, **285**(5), 2177–2198.

Ma, B. Y., Elkayam, T., Wolfson, H., and Nussinov, R. (2003). Protein-protein interactions: Structurally conserved residues distinguish between binding sites and exposed protein surfaces. *Proceedings Of The National Academy Of Sciences Of The United States Of America*, **100**(10), 5772–5777.

Murzin, A. G., Brenner, S. E., Hubbard, T., and Chothia, C. (1995). SCOP – a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology*, **247**(4), 536–540.

Patil, A., Kinoshita, K., and Nakamura, H. (2010). Hub promiscuity in protein-protein interaction networks. *International Journal Of Molecular Sciences*, **11**(4), 1930–1943.

Shoemaker, B. A., Panchenko, A. R., and Bryant, S. H. (2006). Finding biologically relevant protein domain interactions: Conserved binding mode analysis. *Protein Science*, **15**(2), 352–361.

Stein, A., Ceol, A., and Aloy, P. (2010). 3did: identification and classification of domain-based interactions of known three-dimensional structure. *Nucleic Acids Research*, **39**, D718–D723.

Tsai, C.-J., Ma, B., and Nussinov, R. (2009). Protein-protein interaction networks: how can a hub protein bind so many different partners? *Trends In Biochemical Sciences*, **34**(12), 594–600.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, **58**, 236–244.

Xu, D., Tsai, C. J., and Nussinov, R. (1997). Hydrogen bonds and salt bridges across protein-protein interfaces. *Protein Engineering*, **10**(9), 999–1012.

C.2 Using Case-Based Reasoning to Model 3D Protein Complexes from Structural Domain-Domain Interactions

The following article was submitted to the European Conference on Computational Biology (ECCB) 2012 (14% acceptance rate) and it received positive feedback from the reviewers. We are planning to submit a revised manuscript as a full journal article.

BIOINFORMATICS

Vol. 00 no. 00 0000
Pages 1–7

Using case-based reasoning to model 3D protein complexes from structural domain-domain interactions

Anisah W. Ghoorah,[1,3] Marie-Dominique Devignes,[2] Malika Smaïl-Tabbone,[3] and David W. Ritchie [1]*

[1]INRIA, [2]CNRS, [3]Nancy Université
LORIA, Campus Scientifique, BP 239, 54506 Vandoeuvre-lès-Nancy, France

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Associate Editor: XXXXXXX

ABSTRACT

Motivation: The protein docking problem is the problem of how to calculate the three-dimensional (3D) structure of a protein complex from its unbound components. Although *ab initio* docking algorithms are improving, there is a growing need to use homology modeling techniques to exploit the rapidly increasing volumes of structural information that now exist. However, previous homology modeling approaches have been applied either only at the domain level, or they are limited to using the complete single-chain structures of a homologous protein complex as a 3D template. Here we wish to avoid such limitations by modeling protein complexes directly from known domain-domain interactions (DDIs). However, modeling 3D protein complexes from multiple DDIs is a non-trivial problem.

Results: We have developed a case-based reasoning approach for modelling the structures of 3D protein complexes by systematically identifying and re-using domain family binding sites from our database of non-redundant DDIs. When tested on 101 protein complexes from the Protein Docking Benchmark, our approach provides a near-perfect way to model single-domain and multi-domain protein complexes when suitable templates are available, and it extends our ability to model more difficult problems when only partial or incomplete templates exist. This represents a first step towards automatically reasoning about PPI networks at a structural level.

Availability: http://kbdock.loria.fr/

Contact: Dave.Ritchie@inria.fr

1 INTRODUCTION

The protein docking problem is the problem of how to calculate the three-dimensional (3D) structure of a protein complex starting from the unbound components. This problem was first described some thirty years ago (Wodak and Janin, 1978), and since then many computational docking algorithms have been developed (Halperin *et al.*, 2002). Nowadays, there is growing interest in using docking techniques to model the structural basis of protein-protein interaction (PPI) networks (Aloy *et al.*, 2005). Some recent studies have started to use computational docking to predict protein interaction partners and structural PPI networks (Sacquin-Mora *et al.*, 2008; Mosca *et al.*, 2009; Wass *et al.*, 2011). In the last ten years, the "CAPRI" (Critical Assessment of Predicted Interactions) blind docking experiment has stimulated many improvements (Janin *et al.*, 2003; Wodak and Mendez, 2004). However, while some good progress has been made (Lensink and Wodak, 2010), it still remains challenging to produce a satisfactory 3D model of a protein complex using *ab initio* docking algorithms, and cross-docking multiple proteins can produce many false-positive interactions. On the other hand, several studies have shown that using experimental information to guide and constrain docking calculations can improve the reliability of the predictions significantly (van Dijk *et al.*, 2005; Lensink and Wodak, 2010). Given that the number of solved structures in the Protein Data Bank (PDB) appears to be growing exponentially (Berman, 2008), the possibility to exploit existing structural knowledge of proteins and their interactions seems to be an increasingly promising way to model the 3D structures of unknown complexes.

Because protein domains may often be identified as structural and functional units, the 3D structures of protein complexes are often analysed in terms of their component domain-domain interactions (DDIs). Several structural interaction databases describe PPIs in terms of DDIs (Tuncbag *et al.*, 2009). Since it has been shown that proteins with similar sequences often interact in similar ways (Aloy *et al.*, 2003), and since it is well known that protein folds are often more conserved than their sequences (Chothia and Lesk, 1986), it follows that homologous families of protein domains could be expected to interact in structurally similar ways. Indeed, several studies have found that the locations of protein interaction sites are often conserved within domain families (Korkin *et al.*, 2005; Shoemaker *et al.*, 2006).

Several groups, including ourselves, have described homology-based PPI modeling approaches. For example, Lu *et al.* (2002), Grimm *et al.* (2006), Launay and Simonson (2008), and Mukherjee and Zhang (2011) used threading techniques to predict the 3D structure of a complex starting from the sequences of the component proteins and a library of known complexes. Korkin *et al.* (2006) used their comparative patch analysis technique together with knowledge of known binding sites to define docking restraints for modeling a selection of protein complexes. The ISEARCH algorithm of Gunther *et al.* (2007) models protein complexes by comparing the local surface similarity of pair of query domains with a library of DDIs derived from the PDB. The HOMBACOP approach of Kundrotas *et al.* (2008) uses BLAST to retrieve complex templates

*to whom correspondence should be addressed

which are similar to the given query sequences. Movshovitz-Attias *et al.* (2010) recently carried out a detailed study on the utility of using structural templates when modeling conformational flexibility in a number of enzyme-inhibitor and antibody-antigen docking problems.

However, to our knowledge, all such approaches have either been applied only at the domain-domain level, (Gunther *et al.*, 2007; Launay and Simonson, 2008; Ghoorah *et al.*, 2011), or they need to have available the complete single-chain structures of a homologous protein-protein complex. We believe that even if a complete protein-protein template does not exist, by applying the principles of homology all the way from sequence families to DDIs and PPIs, it should still be possible to propose 3D models of protein complexes by reasoning about the available structural knowledge of the component domains. However, given that a protein may consist of one or more domains, and knowing that domain families may have several binding sites, it follows that multiple possible combinations of DDIs should be considered when modeling PPIs. Hence, predicting PPIs from DDIs is a non-trivial problem.

From a computational point of view, docking by homology can be considered as a kind of case-based reasoning (CBR). In general, CBR algorithms aim to solve new problems by adapting the solutions found for similar previous cases (Kolodner, 1992). CBR is a very broadly defined method of problem-solving, and many types of CBR systems have been implemented in many different ways (Aamodt and Plaza, 1994). Nonetheless, most CBR systems typically maintain a "case base" (CB) of previous cases, and they solve problems (new cases) by applying four main steps (de Mántaras *et al.*, 2005), namely, (i) retrieve the most similar case or cases from the CB, (ii) re-use or adapt those cases in order to better match the problem and to propose a solution, (iii) revise the proposed solution if necessary, and (iv) retain the solved case in the CB for future use.

In docking by homology, it might at first seem natural to build a collection of solved protein complexes to serve as the CB, and to predict the structure of a new complex by matching its component proteins to the structures in the CB. However, as explained above, this would allow only a limited number of full-length single-chain complexes to be modeled. In order to be able to deal with a much wider and more diverse range of protein-protein modeling problems, we therefore treat individual domains and their associated domain-level interactions as the structural units of knowledge.

Here, we present a CBR-inspired approach for modelling protein complexes which extends our previously described KBDOCK system for modelling pair-wise DDIs (Ghoorah *et al.*, 2011). Briefly, KBDOCK collects and clusters structural DDI information from the 3DID database (Stein *et al.*, 2010) in order to define a non-redundant set of domain family binding sites (DFBSs) using the Pfam (Finn *et al.*, 2010) domain definition. The version of KBDOCK used here was built using a total of 140,612 DDIs from 29,922 PDB structures. These DDIs were filtered, superposed, and spatially clustered (Ghoorah *et al.*, 2011) to give a total of 1,439 Pfam DFBSs which are involved in 1,009 distinct domain family interactions. Hence, the public KBDOCK database contains the largest non-redundant set of 3D hetero DDIs currently available. We found that nearly 70% of domain families have just one hetero DFBS, and that very few have four or more DFBSs (Ghoorah *et al.*, 2011). Our statistics confirm previous indications (Korkin *et al.*, 2005; Shoemaker *et al.*, 2006) that domain binding sites are often re-used in DDIs. Our basic assumption, therefore, is that the 3D structures of unsolved protein complexes can be predicted by aggregating the interactions observed between their component DFBSs.

Our main aim here is to use some basic principles from CBR to lay down a systematic way to exploit partial or incomplete structural DDI information in order to extend our ability to model multi-domain protein complexes. In other words, we wish to develop a formal way to reason about and combine knowledge of existing DDIs in order to predict the 3D structures of protein complexes *without* requiring full-length homology templates to exist. Thus, our approach for modelling a protein complex aims to be a versatile hybrid between template-based and *ab initio* docking which will automatically adapt itself according to the structural knowledge available for each target. However, in order to ensure that all of our predictions are derived only from experimentally solved and validated 3D structures, we do not apply the final CBR step of storing the generated solutions in the case base.

2 METHODS

2.1 Homology modeling in the language of CBR

In the language of CBR, a case is a collection of attributes or features which describe a solved problem (here, the experimentally determined structures of a pair of interacting domains). In general, each case may be described by a number of *indexed* and *non-indexed* attributes. Indexed attributes are used for case retrieval, whereas non-indexed attributes provide useful contextual information. Here, the Pfam domain identifiers of the query structures serve as the main indexed attributes, whereas the non-indexed attributes include PDB codes, PDB chain identifiers, amino acid sequences and atomic coordinates. If necessary, indexed attributes may be derived from the non-indexed attributes. For example, KBDOCK uses PfamScan (Finn *et al.*, 2010) to determine the Pfam identifiers of the problem domains automatically from their sequences.

As shown in Figure 1, the information associated with each case includes instance-specific information such as the lists of residues of each domain which participate in a specific DDI, along with other derived instance-specific information such as the calculated geometric centre of the binding site, and the residue of each domain which KBDOCK assigned as the central residue of that particular binding site. By spatially clustering binding sites within Pfam families, KBDOCK also stores a family-level binding site identifier for each instance of a binding site. Thus, instances of DDIs in the CB may be grouped and retrieved according to both the Pfam families and the family-level binding sites involved. For example, the *Kunitz_legume* domain family has five non-redundant hetero DDI cases involving four distinct DFBSs. Because we define binding sites at the Pfam family level, KBDOCK identifies each binding site using a compound identifier, *PfamAC/BindingSite*. Thus, for example, PF00197/1 refers to the first DFBS of the *Kunitz_legume* family.

2.2 Pfam-based case retrieval

By denoting a pair of Pfam DFBSs as $d1/b1$ and $d2/b2$, we use the notation $c(d1/b1, d2/b2)$ to represent a DDI case in the CB. Similarly, by using upper case identifiers to represent unknown or uninstantiated instances, we denote a new problem (query) as

PDB	*1avw*	Structure	
Deposition date	*27-Sep-97*		
Expt. Technique	*X-ray diffraction*	Resolution	*1.75 Å*
Chain_1	*A*	Chain_2	*B*
Sequence_1	*IVGGYTCAANSI..*	Sequence_2	*DFVLDNEGNPL..*
PfamID_1	*Trypsin*	PfamID_2	*Kunitz_legume*
PfamAC_1	*PF00089*	**PfamAC_2**	*PF00197*
Region_1	*16-238*	Region_2	*502-675*
BindingSite_1	*2*	**BindingSite_2**	*1*
BS_res_1	*{Phe-502, ...}*	BS_res_2	*{His-57, ..}*
BS_centre_res_1	*Ser-195*	BS_centre_res_2	*Ser-560*
BS_centre_xyz_1	*(x, y, z)*	BS_centre_xyz_2	*(x, y, z)*

Fig. 1. An example of a DDI case in KBDOCK. Each case consists of a collection of attributes or features. Indexed attributes which may be used for case retrieval (the Pfam accession codes and the binding site identifier), and are shown in bold. For cases that match a given problem, the non-indexed attributes such as the domain sequences or the pre-computed binding site centre residues are used to guide the case adaptation and refinement steps, and to rank the proposed solutions.

$q(d1/B1, d2/B2)$, or often just simply $q(d1, d2)$. This notation allows known binding sites to be given as part of the query if such knowledge is available, and it allows partial or incomplete matches with the CB to be represented in a consistent way. Naturally, the overall aim is to find cases in the CB which match (or, more generally, which can be unified with) the given query specification. If both of the query domains can be unified with cases in the CB, we call this a full-homology (FH) problem, and we denote the set of matching cases as $FH(d1, d2)$. It is worth noting that even for the most favourable problems in which FH cases exist in the CB, the stored cases may involve more than one pair of DFBSs. For example, a recent CAPRI target concerned the complex between trypsin and arrowhead protease inhibitor A (Lensink and Wodak, 2010) which has two solutions involving two different inhibitor binding sites (PDB code 3E8L). On the other hand, it is also possible for one or both of the given query domains to match individually one half of a known DDI in the case base. We call such problems semi-homology (SH) problems, and we let $SH(d1, D2)$ and $SH(D1, d2)$, where $D1 \neq d1$ and $D2 \neq d2$, denote the two possible sets of SH cases for a given query. Furthermore, we call a problem for which both $|SH(d1, D2)| > 0$ and $|SH(D1, d2)| > 0$ a SH-two problem, and we call a problem in which only one query domain has matching cases a SH-one problem. This distinction becomes significant at the docking refinement stage. Of course, if $|FH(d1, d2)| = |SH(d1, D2)| = |SH(D1, d2)| = 0$, then no homologous cases exist, and it is necessary to adapt other more distantly related cases or to use *ab initio* docking. In the present study, we do not consider the possibility of applying adaptation beyond the Pfam level.

2.3 Case adaptation in CBR

Many different kinds of adaptation in CBR have been described (Aamodt and Plaza, 1994; Watson, 1997; de Mántaras *et al.*, 2005). For example, *substitution adaptation* re-instantiates parts of a previous case by applying a domain-specific *transformation* operator to map it onto the problem case. Here, we wish to apply this principle to the protein docking problem, but due to the spatial nature of this task and the fact that we have to may have to deal with multiple combinations of domains, we also need to take into account the possibility that not all of the retrieved DDI cases will be mutually compatible, and that similar docking problems may have *different* solutions. Hence, to model a protein complex, we need to consider multiple possible adaptations from the CB, and we need to rank them according to how well they collectively correspond to the target complex. Figure 2 illustrates this approach. Although this may seem obvious in the context of protein docking, it seems that the notion of *case amalgamation* has only recently started to be considered in the CBR community (Ontañón and Plaza, 2010).

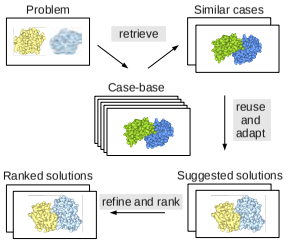


Fig. 2. Overview of the KBDOCK approach for modelling DDIs.

2.4 Modeling FH problems

Since we know from previous experience (Ghoorah *et al.*, 2011) that FH cases very often provide good homology docking models, we collect the instances in $FH(d1, d2)$ into groups having distinct pairs of DBFSs, and we rank the members of each group by their overall sequence similarity to the concatenated sequences of $q(d1, d2)$. We then select the most similar member of each group, and superpose its domains onto the query using the ProFit least-squares fitting program (http://bioinf.org.uk) in order to give a final ranked list of substitution-adapted solutions.

2.5 Modeling SH problems

For SH-two problems, we assume that the target complex may be modeled using a pair of existing DFBSs. We therefore take the Cartesian product $P(d1/B1, d2/B2) = SH(d1/B1) \times SH(d2/B2)$ to enumerate all possible pairs of candidate DFBSs. This broadly corresponds to generating amalgamations of cases in CBR (Ontañón and Plaza, 2010). However, this does not form 3D interfaces between the DFBSs, it only gives a set of symbolic associations. Therefore, for each instance of P, we construct a putative DDI, $p(d1/b1, d2/b2)$, using the coordinates of the stored centre of gravity and central interface residues in order to locate the two domains on the global z axis with their central residues facing each other near the origin, and with $d1$ on the negative z axis and $d2$ on the positive z axis, as illustrated in Figure 3. These transformations have been implemented using functions from the *Hex* docking program (Ritchie and Kemp, 2000). Up to a small translation and an undetermined twist about the z axis, each pair

of such configurations defines a putative pair-wise interface which could be refined by a rigid body docking search. However, since the aim is to find solutions for the given query, we then superpose the domains in $q(d1, d2)$ onto the oriented pair $p(d1/b1, d2/b2)$ in order to obtain a set of candidate solutions.

For SH-one problems, in which one or more DFBSs are known for just one of the query domains, the query domains are oriented on the z axis as described above, using a random surface residue for the uninstantiated binding site centre residue.

2.6 Refining and ranking SH problems

Our *Hex* rigid-body docking algorithm (Ritchie and Kemp, 2000) is used to rank the putative SH solutions generated using the above procedure. Since *Hex* can perform all-versus-all docking of multiple PDB model structures, it is relatively straight-forward to prepare a *Hex* script to perform a rigid-body docking search around each putative DDI in $p(d1/b1, d2/b2)$. For SH-two cases, the docking search is focused around the given pair of binding site centre residues using two angular constraints, β_1 and β_2 (Ritchie *et al.*, 2008), as shown in Figure 3. On the other hand, for SH-one problems, just one angular constraint is applied to the known binding site, and the other domain is allowed to spin freely in order to search over its entire surface. If no DFBSs match the query, unconstrained blind docking is applied. Algorithm 1 shows some high-level pseudo-code which summarises these modelling choices.

Here, each pair-wise *Hex* docking run used 3D FFT shape-based correlation searches with range angles of β_1=β_2=45, as appropriate, and 40 translational steps of 0.5 Å along the z axis with respect to each given starting orientation. This generates approximately 60, 360, and 2,000 million trial rigid body orientations for each pair-wise SH-two, SH-one, and blind docking run, respectively, of which the top 2,000 are re-scored using the DARS (Decoys as Reference State) potential (Chuang *et al.*, 2008). For docking runs involving multiple combinations of DFBSs, individual pair-wise docking results are merged and sorted by DARS energy to give the final ranked list of solutions.

Algorithm 1 High-level pseudo-code for modeling a DDI by CBR.

```
if |FH(d1, d2)| > 0 then
    apply substitution adaptation
else if |SH(d1, D2)| > 0 ∧ |SH(D1, d2)| > 0 then
    apply two substitutions and focused docking
else if |SH(d1, D2)| > 0 ∨ |SH(D2, d2)| > 0 then
    apply one substitution and loosely constrained docking
else
    apply blind docking
end if
```

2.7 Modeling multi-domain docking problems

We now consider the problem of aggregating DDI-level CBR cases to model the 3D structure of a multi-domain protein complex. In general, each protein to be docked may consist of multiple domains. Therefore, if $X = (d1, d2, ..., dN)$ and $Y = (e1, e2, ..., eM)$ represent the proteins X and Y and their component domains, then the preceding analysis of DDIs might suggest that we should take the Cartesian product of all possible pairs of component domains, and apply the above CBR modeling procedure to each pair of domains, and then aggregating the solutions. However, because rigid body docking inevitably produces multiple false-positive solutions, it is important to consider computational docking as a measure of last resort, and to defer any docking calculation for as long as possible in the reasoning process. Furthermore, since each of the target proteins will normally be provided as complete 3D structures, it is not necessary to model any internal DDIs because these are given as part of the problem. Indeed, such internal interactions will obviously "consume" a certain number of DFBSs, thus blocking them from interacting with the domains of the other protein. Therefore, our first strategy is to remove from consideration any DDIs that are implicitly blocked by the other components of the query. This is done by identifying the binding site centres of each domain in the query protein (by querying the CB as described above) and by striking out any binding sites whose centre residues are buried in the query protein. This reduces the number of DFBSs that should be considered as possible docking sites.

The next step is to form a Cartesian product of the surviving DFBSs of each protein, and to query the CB with each putative pair of such DFBSs in order to collect sets of FH and SH cases from the CB. If no cases are retrieved, then *Hex* blind docking is applied directly. Otherwise, if any FH cases are retrieved we assume that the problem can be modeled by superposing the query structures onto the best FH template, as before (see Section 2.4). The only difference from the single DDI procedure is that now all of the atoms of each protein are transformed by the superposition transformation. On the other hand, if no FH case and no SH-two cases are retrieved the proteins are docked and ranked by applying the SH-one procedure to the set of available DBFSs (Section 2.5). Otherwise, all available domains are cross-docked and ranked using the SH-two procedure. Here again, the main difference from the single DDI SH modeling procedure is that now all of the atoms of each protein are transformed when making a docking pose.

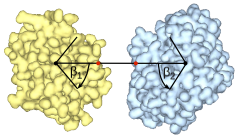


Fig. 3. Illustration of an initial docking pose between a pair of candidate SH domains. The angles β_1 and β_2 represent the *Hex* docking search range angles which will be applied to focus the docking search around the putative interface. Black spheres represent the centre of mass of each domain, and red spheres represent the central residue of each binding site.

3 RESULTS

We used the Protein Docking Benchmark 4.0 (Hwang *et al.*, 2010) as our test dataset. This consists of 176 protein-protein complexes for which the bound structures and the unbound components of at least one of the docking partners are available. Hwang *et al.*

divided the benchmark into 52 enzyme-inhibitor complexes (here called "Enzyme"), 25 antigen-antibody complexes, and 99 "Other" complexes, and they classified each target as "Rigid", "Medium", and "Difficult" according to the degree of conformational changes between the bound and unbound structures. Targets in the Rigid class should be amenable to rigid body docking algorithms, whereas Difficult targets normally require a flexible docking algorithm to be used in conjunction with prior knowledge about the binding mode. Here, we exclude the 25 antigen-antibody complexes because the antibody binding sites are known *a priori* and because the antigen binding sites generally do not entail homology. We also exclude 11 targets for which no Pfam domain could be identified in one or both of their subunits (PDB codes 1clv, 1e6e, 1udi, 2abz, 2b42, 2uuy, 1f51, 1j2j, 1qa9, 1xqs, 2hrk), leaving 140 target complexes.

However, it is worth noting that when the structure of a complex is solved by X-ray crystallography, the structures of the unbound proteins are often solved and deposited in the PDB by the same laboratory at the same time. Therefore, in order to simulate a scenario in which trivial solutions of each target do not already exist, all targets were modeled using only structures with PDB deposition dates earlier than those of the query structures. This removed a further 39 targets from consideration, leaving 101 targets for which non-trivial homologues exist in the CB for one or both partners.

Table 1 compares the CBR-based modeling results with blind *Hex* docking for the 101 target complexes used in the current study. Except for the angular search range constraints, the blind *Hex* runs used the same docking parameters as for the SH problems described above. Overall, 30 of the benchmark targets are FH problems, of which 24 are single-domain targets and six are multi-domain targets. These were modeled by retrieving cases involving the same Pfam families as the query structures, and by ranking the cases with respect to the query sequences and by applying superposition adaptation on the highest ranking cases. For the third target (1eaw), three FH template cases were found. Hence, for this target, the template with the highest overall similarity was used to build the final model. As can be seen from the Rank and RMSD columns in Table 1, this procedure allowed 23 out of the 24 single-domain targets targets to be modeled accurately, with known DFBSs being re-used in all but one of the targets (indicated with an asterisk). A hyphen in this table indicates that no "acceptable" solution within 10Å RMS from the native complex was found within the first 2,000 solutions. The final two columns of Table 1 show the results obtained using *Hex* blind docking. Here, relatively low accuracy solutions were found for only eight of the 24 targets, with only four of those being ranked in the top 10 by the *Hex* scoring function. These contrasting results highlight the utility of modeling DDIs using known DFBSs.

The next two sections of Table 1 show that 30 of the docking targets may be treated as somewhat more challenging SH problems (26 SH-one and four SH-two problems). Of the 26 SH-one problems, 15 of the known DBFSs turn out to be re-used, whereas 12 targets use previously unknown binding sites (according to the PDB date filter). Of the 15 targets that re-use DFBSs, KBDOCK finds eight good solutions, of which six are ranked within the top 4. Blind docking finds slightly more (11) solutions overall, but the ranks for two of these (1kac and 1s1q) are too low to be useful. There are two SH-one targets (1kac and 2g77) for which blind docking finds a solution but CBR does not. Here, KBDOCK retrieves the template DFBSs, but the angular constraint passed to the docking stage is too tight to include a near-native orientation in the search space. On the other hand, for two of the four SH-two cases, the DFBSs on both partners are re-used and CBR finds two good solutions at rank 1 and 2 for 1r0r and 1acb, respectively. Rigid body docking with *Hex* finds only one rather poor solution for these four targets.

The right-hand part of Table 1 shows the results obtained for the multi-domain targets. Like the single-domain targets, the first group of entries show a striking pattern of results for six multi-domain FH problems. For the three targets 1d6r, 1ezu, and 1wdw, one of the partners is a large symmetric homodimer. In all three of these cases, the CBR approach correctly retrieves very low RMSD solutions for each monomer of the dimer (shown as three rank-1 solutions in the table). The next three targets, 2vdb, 1h1v, and wi9b, involve three assymetric repeats of the same domain in one of the partners, and KBDOCK again finds acceptable or better quality docking these three targets (two of which are "Difficult"). In contrast, for these six targets, blind docking found only one solution (for 1d6r) with a very poor rank.

On the other hand, only a relatively small number of the multi-domain SH targets are predicted well either by CBR or by blind docking. This is because several of these targets involve binding sites which are in fact not re-used in the complex (marked with "N" and "*" in Table 1), and because several of the targets (the final 11 SH-one rows marked as "M" and "D") are intrinsically difficult to score and rank using a rigid body docking algorithm.

4 DISCUSSION AND CONCLUSION

We have described a systematic CBR-based approach to model the 3D structures of protein complexes from structural DDIs, and we have tested it on a well known benchmark dataset. Previous homology modelling-based approaches have either focused on docking only DDIs, or they require full-length homology templates to exist. However, it is difficult to compare our results with previous work because each of the previous studies used a different dataset. Here, by working at the Pfam domain level, we are able to draw upon a non-redundant set of almost all currently known hetero DDIs, and we can avoid the above limitations by aggregating pairwise DDIs to predict larger complexes. The results in Table 1 show that for FH problems, in which previously solved cases may be re-used, our approach provides a near-perfect way to retrieve good 3D templates and to build high quality models of the target complexes.

Using PDB deposition data filtering to exclude trivial FH solutions from consideration caused a good proportion of the benchmark targets to be treated as more difficult SH problems. For the single-domain SH targets, we find that known binding sites are re-used in approximately half of the benchmark targets, and this knowledge can usefully be used to guide rigid-body docking. Hence our case-based method of re-using DDIs extends the reach of current homology modeling techniques. Indeed, our results show that if known binding sites are in fact re-used, a good model is often ranked within the first handful of solutions. Although our current implementation can miss some SH-one solutions which are found by blind docking, we believe this is mainly because the angular constraint passed to the rigid docking stage is too strict. On the other hand, our results for multi-domain SH targets show that it is difficult to use computational docking to identify the correct pair

Table 1. Summary of the CBR-based docking results.*

Target PDB	Target Type	Source PDB	SID 1	SID 2	DFBS Pairs	DFBSs Re-used	CBR + *Hex* Rank	CBR + *Hex* RMSD	Blind *Hex* Rank	Blind *Hex* RMSD
Single-domain FH (no *Hex* docking)										
1ay7	E / R	1brs	29.4	94.1	1	Y + Y	1	2.89	–	–
1cgi	E / R	3sgb	20.6	34.7	1	Y + Y	1	1.56	2	9.07
1eaw	E / R	1tfx	41.5	39.6	3	Y + Y	1	0.93	7	8.06
1mah	E / R	1fss	58.9	96.7	1	Y + Y	1	0.76	2	4.63
1n8o	E / R	1ezu	41.3	95.3	1	Y + Y	1	1.31	–	–
1oph	E / R	1jmo	28.9	35.6	1	Y + Y	1	0.65	–	–
1yvb	E / R	1stf	33.8	19.8	1	Y + Y	1	2.98	–	–
2j0t	E / R	1bqq	46.7	31.2	1	Y + Y	1	2.01	5	9.07
2oul	E / R	2nqd	36.4	94.0	1	Y + Y	1	6.31	2	9.61
2sni	E / R	1cse	63.1	35.9	1	Y + Y	1	1.70	–	–
3sgq	E / R	1cgj	18.4	34.0	1	Y + Y	1	0.22	–	–
1ffw	O / R	1a0o	98.3	98.5	1	Y + Y	1	0.52	411	9.72
1fqj	O / R	1agr	63.6	35.3	1	Y + Y	1	0.47	–	–
1gpw	O / R	1ka9	58.5	37.6	1	Y + Y	1	3.57	5	8.78
1he1	O / R	1g4u	99.4	30.0	1	Y + Y	1	0.98	–	–
1xd3	O / R	1cmx	21.8	94.2	1	Y + Y	1	1.12	–	–
2a9k	O / R	2a78	98.8	33.2	1	Y + Y	1	1.03	–	–
2hle	O / R	1shw	44.2	26.7	1	Y + Y	1	2.01	–	–
2oob	O / R	2bwe	20.5	33.3	1	Y + N	*	*	335	9.90
1grn	O / M	1am4	93.8	96.0	1	Y + Y	1	1.74	–	–
1r6q	O / M	1r6o	98.0	98.8	1	Y + Y	1	0.77	–	–
2ayo	O / M	1nbf	22.1	97.1	1	Y + Y	1	1.76	–	–
2nz8	O / M	2dfk	24.0	71.3	1	Y + Y	1	2.84	–	–
3cph	O / M	1vg0	19.4	37.9	1	Y + Y	1	1.86	–	–
Single-domain SH-one (with *Hex* docking)										
1f34	E / R	1htr	51.0	–	1	N + U	*	*	5	8.64
1ffe	E / R	1pyt	53.8	–	3	Y + U	1	7.24	–	–
1gl1	E / R	1acb	98.6	–	6	Y + U	4	8.01	6	7.32
1hia	E / R	2kai	64.3	–	3	Y + U	52	9.56	11	8.45
1oc0	E / R	1k9o	30.1	–	1	N + U	*	*	–	–
1ppe	E / R	1tab	100.0	–	2	Y + U	1	3.54	1	3.29
2sic	E / R	1cse	63.4	–	1	Y + U	1	9.69	1	7.33
1nw9	E / M	1j4e	34.9	–	1	N + U	*	*	23	6.49
1fq1	E / D	1buh	98.6	–	3	N + U	*	*	–	–
1zli	E / D	2bo9	7.5	–	1	Y + U	2	7.77	2	7.13
1buh	O / R	1fin	98.6	–	2	N + U	*	*	–	–
1e96	O / R	1foe	97.2	–	3	N + U	*	*	–	–
1gcq	O / R	1ycs	32.7	–	1	N + U	*	*	–	–
1i4d	O / R	1g4u	99.4	–	3	N + U	*	*	–	–
1kac	O / R	1akj	–	17.0	1	U + Y	–	–	265	8.76
1s1q	O / R	1otr	–	89.9	2	U + N	*	*	190	9.59
1z0k	O / R	1ukv	49.4	–	5	Y + U	2	3.00	2	7.06
2g77	O / R	1ukv	–	40.7	5	U + Y	–	–	55	8.58
1lfd	O / M	1wq1	99.4	–	2	Y + U	48	8.59	–	–
1mq8	O / M	1ijk	–	21.7	2	U + N	*	*	–	–
1wq1	O / M	1gua	57.7	–	2	Y + U	–	–	–	–
2h7v	O / M	1g4u	99.4	–	5	Y + U	–	–	–	–
1r8s	O / D	1r4a	55.6	–	3	Y + U	–	–	–	–
1y64	O / D	1nm1	89.3	–	4	Y + U	–	–	–	–
2ido	O / D	1zbu	17.8	–	1	N + U	*	*	–	–
2ot3	O / D	2hv8	38.3	–	5	Y + U	–	–	–	–
Single-domain SH-two (with *Hex* docking)										
1avx	E / R	1mct + 1ava	100.0	–	5	Y + N	*	*	–	–
1r0r	E / R	1cse + 1ppf	89.5	–	2	Y + Y	2	2.61	61	9.90
2o8v	E / R	1zun + 2bto	–	99.0	2	N + Y	–	–	–	–
1acb	E / M	1cgj + 2tec	100.0	–	2	Y + Y	1	8.60	–	–

(continued)

Table 1. (continued).

Target PDB	Target Type	Source PDB	SID 1	SID 2	DFBS Pairs	DFBSs Re-used	CBR + *Hex* Rank	CBR + *Hex* RMSD	Blind *Hex* Rank	Blind *Hex* RMSD
Multi-domain FH (no *Hex* docking)										
1d6r	E / R	1tab	100.0	65.4	2	Y + Y	1	1.60	379	8.22
1ezu	E / R	1azz	38.6	99.2	2	Y + Y	1	0.95	–	–
1wdw	E / R	2tys	60.0	30.4	2	Y + Y	1	0.71	–	–
2vdb	O / R	1tf0	25.7	96.0	3	Y + Y	1	3.59	–	–
1h1v	O / D	1d4x	92.5	96.4	3	Y + Y	1	8.79	–	–
2i9b	O / D	2fd6	20.5	100.0	3	Y + Y	1	6.02	–	–
Multi-domain SH-one (with *Hex* docking)										
1bvn	E / R	1viw	50.2	–	1	Y + U	1	9.76	1	4.19
1ewy	E / R	1ep3	19.6	–	1	Y + U	1	7.82	3	5.98
1oyv	E / R	2sec	89.5	–	2	Y + U	–	–	–	–
1tmq	E / R	1dhk	52.5	–	2	Y + U	17	6.25	3	7.73
1ijk	E / M	1hyr	–	24.6	2	U + N	*	*	–	–
1jiw	E / M	1smp	–	36.9	1	U + Y	–	–	–	–
1kkl	E / M	1ggr	–	34.5	1	U + Y	–	–	–	–
1a2k	O / R	1wq1	27.2	–	1	Y + U	–	–	–	–
1azs	O / R	1gg2	–	39.1	1	U + Y	–	–	–	–
1b6c	O / R	1fin	–	21.5	3	U + N	*	*	1	6.16
1fcc	O / R	1frt	19.5	–	2	U + N	*	*	–	–
1jwh	O / R	1buh	–	29.4	3	U + N	*	*	–	–
1ktz	O / R	1es7	–	34.0	1	U + N	*	*	–	–
1kxp	O / R	1hlu	92.5	–	4	Y + U	–	–	–	–
1ofu	O / R	1ia0	–	14.3	1	U + N	*	*	–	–
1pvh	O / R	1f6f	26.6	–	2	Y + U	–	–	–	–
1rv6	O / R	1pdg	24.4	–	2	N + U	*	*	–	–
1t6b	O / R	1v7p	–	16.2	3	U + N	*	*	–	–
1xu1	O / R	1oqd	37.0	–	6	Y + U	109	7.56	–	–
2b4j	O / R	1c6v	63.0	–	2	N + U	*	*	–	–
2fju	O / R	1g4u	–	99.4	4	U + Y	–	–	–	–
3bp8	O / R	1o2f	–	94.3	1	U + Y	–	–	–	–
3d5s	O / R	1ghq	99.6	–	2	Y + U	28	8.93	64	3.29
1he8	O / M	1gua	57.1	–	2	Y + U	–	–	–	–
1i2m	O / M	1ibr	98.1	–	3	Y + U	–	–	–	–
1k5d	O / M	1rrp	99.2	–	3	Y + U	–	–	–	–
1n2c	O / M	3min	99.8	–	4	Y + U	–	–	–	–
1bkd	O / D	1wq1	99.4	–	2	Y + U	–	–	–	–
1de4	O / D	1qqd	44.2	–	5	Y + U	–	–	–	–
1eer	O / D	1axi	–	14.4	2	U + N	*	*	–	–
1fak	O / D	1kig	41.5	–	4	N + U	*	*	–	–
1jmo	O / D	2hpp	–	99.6	6	U + Y	–	–	–	–
1zm4	O / D	1aip	15.7	–	3	N + U	*	*	–	–
2c0l	O / D	1e96	16.7	–	1	N + U	*	*	–	–
Multi-domain SH-two (with *Hex* docking)										
1gxd	E / R	1ueu + 1smp	27.5	41.3	2	N + N	*	*	–	–
1m10	E / M	1ijk + 1dfj	98.3	36.0	1	N + N	*	*	147	8.85
1akj	O / R	1bhg + 1agf	100.0	23.9	3	N + Y	*	*	150	7.55
1hef	O / R	1sgf + 1evt	48.7	20.7	2	N + N	*	*	–	–
1klu	O / R	1aqd + 1klg	100.0	98.1	44	Y + Y	–	–	–	–
1sbb	O / R	1rvf + 2seb	31.3	81.0	3	N + N	*	*	–	–
1gp2	O / M	1got	68.9	98.3	13	Y + Y	148	9.11	–	–
1ibr	O / D	1qbk + 1rrp	98.1	18.3	3	Y + Y	–	–	–	–

*For several targets, multiple cases are retrieved and considered ("DFBS Pairs"). This table shows only a single template or pair of templates for each docking target, being the one with the highest percentage sequence identity (SID) to the query structures. In the table, a hyphen (–) denotes no solution found within the top 2,000 orientations. An asterisk (*) denotes no solution was found by CBR because a binding site was not re-used. Abbereviations used: E: Enzyme-inhibitor target; O: Other target; R: Rigid body target; M: Medium difficulty target; D: Difficult target; Y: Yes (binding site re-used); N: No (binding site not re-used); U: Unknown binding site.

of binding sites in multi-domain complexes. This is perhaps not surprising given that each trial pair of initial docking poses will produce a large number of false-positive docking predictions which will mask any near-native solutions. Indeed, if both partners have multiple binding sites, then without additional biological knowledge we essentially revert to a blind docking problem (as in e.g. the SH-two targets 1klu and 1gp2, with 44 and 13 candidate DDIs to be docked, respectively). Hence, it would be desirable to incorporate more sophisticated restraints derived from other kinds of biological evidence and to use a more powerful flexible docking algorithm such as e.g. HADDOCK (de Vries *et al.*, 2010), especially when dealing with difficult targets involving conformational flexibility.

Nonetheless, given the rapid growth in the number of 3D structures in the PDB, and the growing volumes of other biochemical and biophysical data which can usefully be applied to help model macromolecular complexes (van Dijk *et al.*, 2005), it is becoming more and more time-consuming and tedious to study and model thoroughly even individual protein complexes. Therefore, in order to understand more fully the complex biomolecular networks and mechanisms which drive the cell, we need to develop more systematic and automated ways to reason about and model the 3D structural interactions which occur between the macromolecular components. We believe that the CBR approach presented here demonstrates a practical first step towards this goal.

ACKNOWLEDGEMENT

Part of this work was funded by the Agence Nationale de la Recherche, grant references ANR-08-CEXC-017-01 and ANR-11-MONU-006-02.

REFERENCES

Aamodt, A. and Plaza, E. (1994). Case-based reasoning; foundational issues, methodological variations, and system approaches. *AI COMMUNICATIONS*, **7**(1), 39–59.

Aloy, P., Ceulemans, H., Stark, A., and Russell, R. B. (2003). The relationship between sequence and interaction divergence in proteins. *Journal of Molecular Biology*, **332**(5), 989–998.

Aloy, P., Pichaud, M., and Russell, R. B. (2005). Protein complexes: structure prediction challenges for the 21(st) century. *Current Opinion In Structural Biology*, **15**(1), 15–22.

Berman, H. M. (2008). The Protein Data Bank: a historical perspective. *Acta Crystallographica*, **A38**, 88–95.

Chothia, C. and Lesk, A. M. (1986). The relation between the divergence of sequence and structure in proteins. *EMBO Journal*, **5**, 823–826.

Chuang, G.-Y., Kozakov, D., Brenke, R., Comeau, S. R., and Vajda, S. (2008). DARS (decoys as the reference state) potentials for protein-protein docking. *Biophysical Journal*, **95**(9), 4217–4227.

de Mántaras, R. L., McSherry, D., Bridge, D. G., Leake, D. B., Smyth, B., Craw, S., Faltings, B., Maher, M. L., Cox, M. T., Forbus, K. D., Keane, M. T., Aamodt, A., and Watson, I. D. (2005). Retrieval, reuse, revision and retention in case-based reasoning. *Knowledge Eng. Review*, **20**(3), 215–240.

de Vries, S. J., Melquiond, A. S. J., Kastritis, P. L., Karaca, E., Bordogna, A., van Dijk, M., Rodrigues, J. P. G. L. M., and Bonvin, A. M. J. J. (2010). Strengths anbd weaknesses of data-driven docking in critical assessment of predicted interactions. *Proteins: Struct. Func. Bioinf.*, **78**, 3242–3249.

Finn, R. D., Mistry, J., Tate, J., Coggill, P., Heger, A., Pollington, J. E., Gavin, O. L., Gunasekaran, P., Ceric, G., Forslund, K., Holm, L., Sonnhammer, E. L. L., Eddy, S. R., and Bateman, A. (2010). The Pfam protein families database. *Nucleic Acids Research*, **38**, D211–D222.

Ghoorah, A. W., Devignes, M.-D., Smail-Tabbone, M., and Ritchie, D. W. (2011). Spatial clustering of protein binding sites for template based protein docking. *Bioinformatics*, **27**(20), 2820–2827.

Grimm, V., Zhang, Y., and Skolnick, J. (2006). Benchmarking of dimeric threading and structure refinement. *Proteins: Struct. Func. Bioinf.*, **63**, 457–464.

Gunther, S., May, P., Hoppe, A., Frommel, C., and Preissner, R. (2007). Docking without docking: ISEARCH – prediction of interactions using known interfaces. *Proteins: Structure Function and Bioinformatics*, **69**(4), 839–844.

Halperin, I., Ma, B., Wolfson, H., and Nussinov, R. (2002). Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins: Struct. Func. Genet.*, **47**, 409–443.

Hwang, H., Vreven, T., Janin, J., and Weng, Z. (2010). Protein-protein docking benchmark version 4.0. *Proteins: Structure Function and Bioinformatics*, **78**(15), 3111–3114.

Janin, J., Henrick, K., Moult, J., Ten Eyck, L., Sternberg, M. J. E., Vajda, S., Vakser, I., and Wodak, S. J. (2003). CAPRI: a critical assessment of predicted interactions. *Proteins: Structure Function and Genetics*, **52**, 2–9.

Kolodner, J. L. (1992). An introduction to case-based reasoning. *Artificial Intelligence Reviews*, **6**, 3–34.

Korkin, D., Davis, F. P., and Sali, A. (2005). Localization of protein-binding sites within families of proteins. *Protein Science*, **14**, 2350–2360.

Korkin, D., Davis, F. P., Alber, F., Luong, T., Shen, M.-Y., Lucic, V., Kennedy, M. B., and Sali, A. (2006). Structural modeling of protein interactions by analogy: application to PSD-95. *PLoS Computational Biology*, **2**(11), el 53.

Kundrotas, P. J., Lensink, M. F., and Alexov, E. (2008). Homology-based modeling of 3D structures of protein-protein complexes using alignments of modified sequence profiles. *International Journal of Biological Macromolecules*, **43**(2), 198–208.

Launay, G. and Simonson, T. (2008). Homology modelling of protein-protein complexes: a simple method and its possibilities and limitations. *BMC Bioinformatics*, **9**, 427.

Lensink, M. F. and Wodak, S. J. (2010). Docking and scoring protein interactions: CAPRI 2009. *Proteins: Structure Function and Bioinformatics*, **78**(15), 3073–3084.

Lu, L., Lu, H., and Skolnick, J. (2002). Multiprospector: An algorithm for the prediction of protein-protein interactions by multimeric threading. *Proteins-Structure Function And Genetics*, **49**(3), 350–364.

Mosca, R., Pons, C., Fernandez-Recio, J., and Aloy, P. (2009). Pushing structural information into the yeast interactome by high-throughput protein docking experiments. *PLoS Computational Biology*, **5**(8), e1000490.

Movshovitz-Attias, D., London, N., and Schueler-Furman, O. (2010). On the use of structural templates for high-resolution docking. *Proteins: Structure Function and Bioinformatics*, **78**(8), 1939–1949.

Mukherjee, S. and Zhang, Y. (2011). Protein-protein complex structure predictions by multimeric threading and template recombination. *Structure*, **19**(7), 955–966.

Ontañón, S. and Plaza, E. (2010). Amalgams: A formal approach for combining multiple case solutions. In *ICCBR*, pages 257–271.

Ritchie, D. W. and Kemp, G. J. L. (2000). Protein docking using spherical polar Fourier correlations. *Proteins: Structure Function and Genetics*, **39**(2), 178–194.

Ritchie, D. W., Kozakov, D., and Vajda, S. (2008). Accelerating and focusing protein-protein docking correlations using multi-dimensional rotational FFT generating functions. *Bioinformatics*, **24**(17), 1865–1873.

Sacquin-Mora, S., Carbone, A., and Lavery, R. (2008). Identification of protein interaction partners and protein-protein interaction sites. *Journal of Molecular Biology*, **382**(5), 1276–1289.

Shoemaker, B. A., Panchenko, A. R., and Bryant, S. H. (2006). Finding biologically relevant protein domain interactions: Conserved binding mode analysis. *Protein Science*, **15**(2), 352–361.

Stein, A., Ceol, A., and Aloy, P. (2010). 3did: identification and classification of domain-based interactions of known three-dimensional structure. *Nucleic Acids Research*, **39**, D718–D723.

Tuncbag, N., Kar, G., Keskin, O., Gursoy, A., and Nussinov, R. (2009). A survey of available tools and web servers for analysis of protein-protein interactions and interfaces. *Briefings In Bioinformatics*, **10**(3), 217–232.

van Dijk, A. D., Boelens, R., and Bonvin, A. M. (2005). Data-driven docking for the study of biomolecular complexes. *FEBS Journal*, **272**(2), 293–312.

Wass, M. N., Fuentes, G., Pons, C., Pazos, F., and Valencia, A. (2011). Towards the prediction of protein interaction partners using physical docking. *Molecular Systems Biology*, **7**, 469.

Watson, I. D. (1997). *Applying case-based reasoning - techniques for the enterprise systems*. Morgan Kaufmann.

Wodak, S. J. and Janin, J. (1978). Computer analysis of protein-protein interaction. *J. Mol. Biol.*, **124**, 323–342.

Wodak, S. J. and Mendez, R. (2004). Prediction of protein-protein interactions: the CAPRI experiment, its evaluation and implications. *Current Opinion In Structural Biology*, **14**(2), 242–249.

C.3 KBDOCK: A New Resource for Knowledge-Based Protein Docking

The following article was submitted to Nucleic Acid Research (NAR) database server annual issue (2011) and it received favourable comments from the reviewers. We are going to submit an updated version to the 2013 issue of the journal.

Published online X XXXX XXXX

Nucleic Acids Research, XXXX, Vol. XX, No. X **1–5**
doi:10.1093/nar/gkn000

KBDOCK: A new resource for knowledge-based protein docking

Anisah W. Ghoorah,[1] Marie-Dominique Devignes,[2] Malika Smaïl-Tabbone,[3] and David W. Ritchie[1*]

[1]INRIA, [2]CNRS, [3]Nancy Université
LORIA, Campus Scientifique, BP 239, 54506 Vandoeuvre-lès-Nancy, France

Received XXXX X, XXXX; Revised XXXX X, XXXX; Accepted XXXX X, XXXX

ABSTRACT

KBDOCK is a 3D database system that defines and spatially clusters protein binding sites for knowledge-based protein docking. KBDOCK integrates protein domain-domain interaction (DDI) information from 3DID, sequence alignments from Pfam, and structural information from the PDB. This allows the spatial arrangements of DDIs within Pfam families to be analysed and predicted. KBDOCK can be queried using Pfam domain identifiers, protein sequences, or 3D protein structures. For a given query domain, KBDOCK provides a non-redundant list of its DDIs and their binding sites in a common coordinate frame. A unique feature of KBDOCK is that it can also be queried with two query domains simultaneously. In this case, KBDOCK provides the best DDI for both query domains, if available, for use as a docking template. Even if no pair-wise DDI homologues can be found, KBDOCK can often propose binding sites for the individual domains, which can be useful for conventional docking calculations. The KBDOCK web interface is freely available at http://kbdock.loria.fr/.

INTRODUCTION

Protein-protein interactions play a central role in many cellular processes. To date, over 65,000 protein structures have been deposited in the Protein Data Bank (PDB (1)). However, it has been estimated recently that only about 12% of these structures correspond to heteromeric complexes (2). Therefore, to bridge this gap, there is much interest in developing computational docking techniques to build models of protein complexes. There currently exist many protein docking algorithms (3), but *ab initio* docking can be difficult, and it is often advantageous to exploit biological knowledge about the nature of the interaction (4, 5, 6). In recent years, much structural information on protein interactions has been made available in on-line databases (7). These databases constitute useful bioinformatics resources, but there remains a need to develop automated ways to exploit the three-dimensional (3D) nature of this data more effectively.

Here we present KBDOCK, a new resource for knowledge-based protein docking. KBDOCK integrates domain-domain interaction (DDI) information from 3DID (8) and sequence alignments from Pfam (9) together with structural information from the PDB in order to analyse the spatial arrangements of DDIs by Pfam family, and to propose structural templates for protein docking. KBDOCK has an easy-to-use web interface which can be queried by Pfam domain, protein sequence or structure information. For a given query domain, KBDOCK finds a list of non-redundant hetero DDIs, and allows the user to visualise them interactively in the coordinate frame of the query. A unique feature of KBDOCK is that it can also be queried with two query domains simultaneously. In this case, KBDOCK provides the best DDI for both query domains, if available, for use as a docking template. Even if no pair-wise DDI homologues can be found, KBDOCK can often find binding sites involving the individual domains, which can be useful for conventional docking calculations. Thus, KBDOCK provides a novel and easy way to analyse known DDIs, and to find structural templates to help model unsolved protein complexes.

MATERIALS AND METHODS

Defining Pfam family-level binding sites

Figure 1 summarizes the data sources and processing steps to populate the KBDOCK database. Briefly, for each of the 3,713 Pfam families in 3DID, DDIs are extracted and are classified into "intra," "homo," and "hetero" interactions. Currently, only hetero interactions are retained. Next, biologically relevant interfaces are distinguished from crystal contacts using interface areas calculated by the DSSP program (10), and interface residues are annotated as "core" or "rim" depending on their solvent accessibility (11). To obtain a set of non-redundant interactions, the NRDB90 program (12) is used with a sequence similarity threshold of 99%. The sequences of the selected domains are then aligned with the Pfam consensus sequence in order to obtain a mapping between the UniProt and PDB sequence numbers. This mapping is used to superpose DDIs using the ProFit least squares fitting program (http://bioinf.org.uk). For each superposed domain, an interface direction vector is calculated from the all-atom centre of mass of each domain to the centre of its binding site using a weighted average of its core (75%) and rim (25%) residues. Finally, in order to define domain family binding sites, interface direction vectors are clustered using a hierarchical clustering algorithm. The results of all of the above calculations are stored in the KBDOCK database.

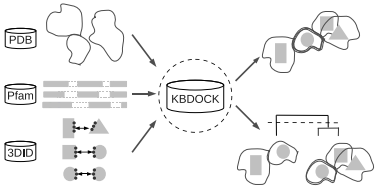


Figure 1. The three primary data sources are PDB for protein structures, Pfam for protein domain assignments and multiple sequence alignments, and 3DID for DDIs and residue contacts. Different shapes such as circles and triangles represent different Pfam domains. Processing this data involves four main steps: (i) selecting hetero DDIs, (ii) mapping PDB sequences to the Pfam consensus and identifying core and rim residues, (iii) superposing DDIs and calculating interface direction vectors, and (iv) clustering interface vectors to define Pfam family-level binding sites.

The data model

Figure 2 shows a simplified view of the KBDOCK relational data model. The *Pfam_PDB_domain* entity contains the mapping of all the PDB domain sequences to their corresponding Pfam consensus sequence. The *DDI* entity contain DDIs classified into intra, homo, or hetero. For every hetero DDI, the *binding_site* entity contains 3D information about the interaction, such as the atomic coordinates of the interaction in the superposed orientation, the centre of mass of each domain, the calculated center of the interface, along with the binding site direction vector, and the residue calculated to be closest to the centre of the binding site. The *core_rim_residues* entity lists the interacting residues annotated as either core or rim.

The physical database is implemented using the MySQL database engine (http://www.mysql.com). Operations such as parsing and processing the data are implemented in Prolog (http://www.swi-prolog.org/). Binding site direction vectors are clustered using R scripts (http://www.r-project.org/). The current version of KBDOCK contains 2,721 non-redundant hetero DDIs involving 1,029 Pfam families, and was built using Pfam version 24.0 and the November 2009 version of 3DID.

The web interface

The KBDOCK web interface is written mainly in the PHP scripting language (http://php.net). Queries are processed using Prolog and Linux shell scripts. The Jmol plug-in is used for graphical visualisation (http://jmol.sourceforge.net). The web interface has been tested using several popular browsers for the Windows, Linux, and Mac OS X operating systems.

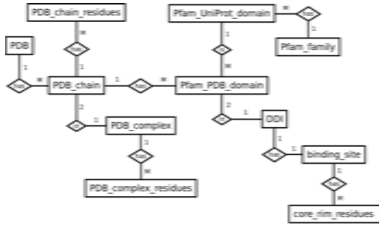


Figure 2. A simplified view of the entity-relationship diagram of the KBDOCK database.

Finding homology docking templates

Given two query domain structures, KBDOCK searches for DDIs involving the same Pfam families as the query domains. We call any DDIs that satisfy this search "full-homology" (FH) templates. As discussed above, a protein complex may often be successfully modeled using the known binding sites of homologous domains. Hence, the query domains are superposed onto the FH template(s) in order to propose a docking model of the complex. If several FH DDIs exist in the database and if they correspond to different pairs of binding sites, KBDOCK outputs a proposed docking model for each distinct pair of binding sites.

On the other hand, if no FH templates are found, KBDOCK searches for and outputs DDIs containing the individual query domains because these can still provide useful information for a docking calculation. We call such DDIs "semi-homology" (SH) templates in analogy to a semi-join in relational algebra. In these cases, the query domain is superposed onto each template in turn in order to propose a binding site on the query domain. If several SH templates are found for a given query domain, KBDOCK selects as a template the domain with the highest sequence similarity to the query. The overall approach is illustrated schematically in Figure 3.

RESULTS

Analyzing binding sites by Pfam family

In order to analyse the binding sites of a given Pfam family, the user may use the KBDOCK "Search" page (Figure 4) to enter either a Pfam identifier (e.g. Kunitz_legume), a Pfam accession number (e.g. PF00197), a keyword (e.g. inhibitor), an amino acid sequence, or a PDB file of a protein structure. If a sequence or a structure is entered, the PfamScan utility (ftp://ftp.sanger.ac.uk/pub/databases/Pfam/Tools/) is used to determine the Pfam accession number. Otherwise, the accession number is found directly from KBDOCK database. KBDOCK then retrieves a non-redundant list of hetero DDIs involving the query domain, grouped by their binding site. Figure 5 shows the output when KBDOCK is queried using

*To whom correspondence should be addressed. Tel: +33 3 83 59 30 45; Fax: +33 3 83 59 30 79; Email: dave.ritchie@loria.fr

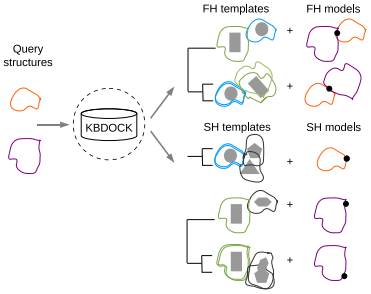


Figure 3. Schematic illustration of how KBDOCK processes a docking query. Here, dots represent the calculated binding site centre residues. When a FH template is available, KBDOCK superposes the query domains onto the template to propose a docking model. When only SH templates exist, KBDOCK proposes one or more binding site(s) on each query domain. For each selected template, KBDOCK calculates the core, rim, and binding site centre residues.

the Kunitz_legume protease inhibitor domain (PF00197). The Jmol plugin shows the retrieved DDIs in the coordinate frame of the query domain. The query domain is shown in black and interacting residues are shown as wire sticks. The user may choose to view the DDIs together or individually. A Pfam consensus-based sequence alignment of the retrieved domains is also provided, in which each sequence is colour-coded according to the core, rim and centre residue assignments. A link to download the superposed PDB files as a single compressed file for further analysis is also available.

Figure 4. A screenshot of the KBDOCK "Search" page.

Finding docking templates

To find docking templates, the user enters two PDB codes or uploads two PDB files and he then specifies which pair of

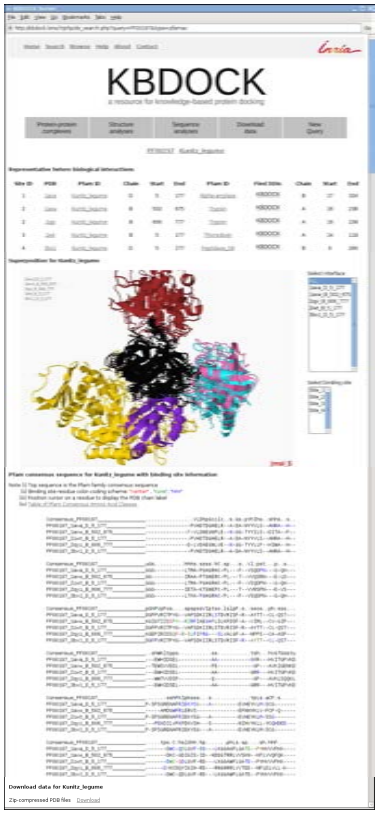
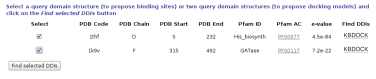


Figure 5. A screenshot of the KBDOCK results for the query domain family, Kunitz_legume. The results page consists of four sections: (i) a non-redundant list of hetero DDIs grouped by their binding site, (ii) a Jmol view of the DDIs in the coordinate frame of the query domain (the query domain is shown in black and interface residues are shown in wireframe), (iii) a Pfam consensus-based sequence alignment of the domains annotated with the core (green), rim (blue), and centre (red) binding site residues, (iv) a link to download the superposed PDB files as a single compressed file for further analysis.

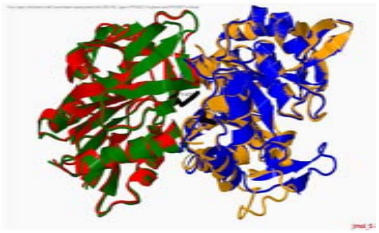
Pfam domains in those structures should be used as queries. If KBDOCK finds one or more FH DDI templates for the query

domains, it shows a Jmol view of the superposed query and FH template(s), along with colour-coded sequence alignments of the query and template domains showing the core, rim, and centre binding site residues. As before, the user may download the query and template structures for later analysis. Figure 6 shows a screenshot of the results page obtained when querying KBDOCK using PDB codes 1thf and 1k9v (which correspond to the His_biosynth and GATase Pfam domains, respectively). In this example, the best FH template is PDB code 1gpw (an imidazole glycerol phosphate synthase). If no FH DDIs exist in the database, KBDOCK will output in a similar way a non-redundant list of SH templates and their colour-coded sequence alignments along with a Jmol view of the superpositions (details not shown).

(a)



(b)



(c)

Figure 6. A screenshot showing **(a)** the selection of two domains for which a docking model is requested, **(b)** a Jmol view of the resulting FH template with the query domains superposed onto the template, **(c)** sequence alignments of the query and template domains highlighting the template core, rim, and centre binding site residues.

Protein Docking Benchmark results

KBDOCK has been evaluated using 73 targets from the Protein Docking Benchmark (13). Overall, KBDOCK produced high quality FH docking templates for 45 of the 73 complexes and it found useful SH templates for 24 of the 28 remaining complexes. Thus, KBDOCK represents a novel and useful way to find structural templates for knowledge-based protein docking.

DISCUSSION

Comparison with other structural databases

Although several structural PPI databases have been described recently, to our knowledge, KBDOCK is the only one which has been specifically designed to facilitate template-based protein docking. Some recent examples of structural PPI databases are SCOPPI (14), IBIS (15), and 3DID (8). SCOPPI classifies domain-domain interfaces based on geometric overlap and face angle scores of residue contact vectors (16). For a given SCOP family, SCOPPI outputs all PDB complexes involving the query. The DDIs are grouped according to their partner domain. For each group, multiple sequence alignments with interacting residues marked are available for both the query and partner family. Other information available includes their in-house interface type, interface area and volume, screenshot of the interface, and links to related publications. 3DID classifies domain-domain and domain-peptide interacions using hierarchical complete linkage clustering of groups of interface residues. For a given Pfam family, 3DID outputs a list of all its partner domains grouped by their interface profile. IBIS stores experimentally determined and inferred interactions between proteins, peptides, DNA, and RNA, and other small molecules. Like 3DID, IBIS classifies DDIs using hierarchical complete linkage clustering of groups of interface residues. For a given query protein, IBIS outputs a list of all its partner proteins. The interactions are listed as DDIs, which are grouped by their partner domain and their binding site. The identities of binding site residues on the query protein are also proposed.

Although these databases are useful, they cannot be used to provide docking templates directly because they cannot be queried with two domains simultaneously. Furthermore, they do not allow the binding sites and domain interactions of a given query domain to be visualised interactively in a common coordinate frame. In contrast, KBDOCK was designed right from the beginning to identify automatically structural templates with which to guide protein docking calculations. Hence, KBDOCK has many features which distinguish it from existing structural PPI databases: (i) it uses the Pfam consensus sequence to place all of the complexes involving a given Pfam domain family into a common coordinate frame, (ii) it uses the notion of "core" and "rim" interface residues to group the complexes by the spatial position of their binding site, (iii) it finds automatically the best available DDI template to use to model by homology a complex of two given structures, (iv) if more than one interface is found, it proposes a model for each, (v) if no suitable DDI template exists, it can still propose candidate binding sites for one or both interaction partners, and (vi) it calculates a centre residue for each proposed binding site

which may be used to initialise a docking calculation using, e.g., HexServer (17). Additionally, thanks to the Jmol plug-in, the KBDOCK web server provides a convenient way to view and compare Pfam binding sites and calculated docking templates.

Perspectives

We are currently extending KBDOCK to deal with multi-domain complexes. We are also developing machine learning techniques which aim to discover symbolic rules to describe protein binding sites and interfaces at the domain family level. We hope that such developments will provide further ways to help guide protein docking calculations.

CONCLUSIONS

KBDOCK was designed specifically to identify structural templates in order to guide protein docking calculations. KBDOCK therefore provides a novel and useful resource for analysing the 3D structures of DDIs within and between Pfam domain families, and for proposing knowledge-based templates to help predict the structures of unknown protein complexes.

ACKNOWLEDGEMENTS

This work is funded by Agence Nationale de la Recherche, grant reference ANR-08-CEXC-017-01.

Conflict of interest statement. None declared.

REFERENCES

1. H. M. Berman, T. Battistuz, T. N. Bhat, W. F. Bluhm, P. E. Bourne, K. Burkhardt, L. Iype, S. Jain, P. Fagan, J. Marvin, D. Padilla, V. Ravichandran, B. Schneider, N. Thanki, H. Weissig, J. D. Westbrook, and C. Zardecki. The protein data bank. *Acta Crystallographica Section D-Biological Crystallography*, 58:899–907, 2002.
2. A. Stein, R. Mosca, and P. Aloy. Three-dimensional modeling of protein interactions and complexes is going 'omics. *Current Opinion in Structural Biology*, 21:200–208, 2011.
3. I. Halperin, B. Ma, H. Wolfson, and R. Nussinov. Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins: Struct. Func. Genet.*, 47:409–443, 2002.
4. A. D. van Dijk, R. Boelens, and A. M. Bonvin. Data-driven docking for the study of biomolecular complexes. *FEBS Journal*, 272(2):293–312, 2005.
5. G. Launay and T. Simonson. Homology modelling of protein-protein complexes: a simple method and its possibilities and limitations. *BMC Bioinformatics*, 9:427, 2008.
6. P. J. Kundrotas, M. F. Lensink, and E. Alexov. Homology-based modeling of 3D structures of protein-protein complexes using alignments of modified sequence profiles. *International Journal of Biological Macromolecules*, 43(2):198–208, 2008.
7. N. Tuncbag, G. Kar, O. Keskin, A. Gursoy, and R. Nussinov. A survey of available tools and web servers for analysis of protein-protein interactions and interfaces. *Briefings In Bioinformatics*, 10(3):217–232, 2009.
8. A. Stein, A. Ceol, and P. Aloy. 3did: identification and classification of domain-based interactions of known three-dimensional structure. *Nucleic Acids Research*, 39:D718–D723, 2010.
9. R. D. Finn, J. Mistry, J. Tate, P. Coggill, A. Heger, J. E. Pollington, O. L. Gavin, P. Gunasekaran, G. Ceric, K. Forslund, L. Holm, E. L. L. Sonnhammer, S. R. Eddy, and A. Bateman. The Pfam protein families database. *Nucleic Acids Research*, 38:D211–D222, 2010.
10. W. Kabsch and C. Sander. Dictionary of protein secondary structure - pattern-recognition of hydrogen-bonded and geometrical features. *Biopolymers*, 22(12):2577–2637, 1983.
11. P. Chakrabarti and J. Janin. Dissecting protein-protein recognition sites. *Proteins: Structure Function and Genetics*, 47(3):334–343, 2002.
12. L. Holm and C. Sander. Removing near-neighbour redundancy from large protein sequence collections. *Bioinformatics*, 14(5):423–429, 1998.
13. H. Hwang, T. Vreven, J. Janin, and Z. Weng. Protein-protein docking benchmark version 4.0. *Proteins: Structure Function and Bioinformatics*, 78(15):3111–3114, 2010.
14. C. Winter, A. Henschel, W. K. Kim, and M. Schroeder. SCOPPI: a structural classification of protein - protein interfaces. *Nucleic Acids Research*, 34:D310–D314, 2006.
15. B. A. Shoemaker, D. C. Zhang, R. R. Thangudu, M. Tyagi, J. H. Fong, A. Marchler-Bauer, S. H. Bryant, T. Madej, and A. R. Panchenko. Inferred biomolecular interaction server-a web server to analyze and predict protein interacting partners and binding sites. *Nucleic Acids Research*, 38:D518–D524, 2010.
16. W. K. Kim, A. Henschel, C. Winter, and M. Schroeder. The many faces of protein-protein interactions: A compendium of interface geometry. *PLoS Computational Biology*, 2:1151–1164, 2006.
17. G. Macindoe, L. Mavridis, V. Venkatraman, M-D. Devignes, and D. W. Ritchie. HexServer: an FFT-based protein docking server powered by graphics processors. *Nucleic Acids Res*, 38 Suppl:W445–9, 2010.

Acknowledgments

I thank my three supervisors Marie-Dominique Devignes, Malika Smaïl-Tabbone and Dave Ritchie for their patient guidance, advice, support and encouragement during the last three and half years. I am particularly indebted to Dave Ritchie for transforming my scientific manuscripts into smooth and clear English but mostly for his unfailing source of reassurance, for his availability and for his gentle way of keeping me on track.

I also thank David Sherman, Pierre Tufféry, Pascale Kuntz-Cosperec, Anne Poupon and Bernard Girau who kindly accepted to be part of the jury of my thesis defence.

I thank Amedeo Napoli for allowing me to do this thesis in his Orpailleur team. I thank Emmanuelle Deschamps for all her help throughout my stay at the Loria. I thank all the members of the Orpailleur team, in particular, Matthieu Chavent for the useful discussions we had during the first year of my thesis and for our involvement in the CAPRI experiment.

Finally, I am grateful for funding from the French National Research Agency (ANR).

Bibliography

Aamodt,A. and Plaza,E. (1994) Case-based reasoning; foundational issues, methodological variations, and system approaches. *AI Communications,* **7** (1), 39–59.

Alberts,B. (1998) The cell as a collection of protein machines: preparing the next generation of molecular biologists. *Cell,* **92** (3), 291–294.

Aloy,P., Ceulemans,H., Stark,A. and Russell,R.B. (2003) The relationship between sequence and interaction divergence in proteins. *Journal of Molecular Biology,* **332** (5), 989–998.

Aloy,P., Pichaud,M. and Russell,R.B. (2005) Protein complexes: structure prediction challenges for the 21(st) century. *Current Opinion in Structural Biology,* **15** (1), 15–22.

Aloy,P. and Russell,R.B. (2002) Interrogating protein interaction networks through structural biology. *Proceedings of the National Academy of Science,* **99** (9), 5896–5901.

Aloy,P. and Russell,R.B. (2003) InterPreTS: protein interaction prediction through tertiary structure. *Bioinformatics,* **19** (1), 161–162.

Aloy,P. and Russell,R.B. (2004) Ten thousand interactions for the molecular biologist. *Nature Biotechnology,* **22** (10), 1317–1321.

Aloy,P. and Russell,R.B. (2006) Structural systems biology: modelling protein interactions. *Nature Reviews Molecular Cell Biology,* **7** (3), 188–197.

Anfinsen,C.B., Haber,E., Sela,M. and White,F.H. (1961) The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain. *Proceedings of the National Academy of Science,* **47**, 1309–1314.

Apweiler,R., Bairoch,A. and Wu,C.H. (2004*a*) Protein sequence databases. *Current Opinion in Chemical Biology,* **8** (1), 76–80.

Apweiler,R., Bairoch,A., Wu,C.H., Barker,W.C., Boeckmann,B., Ferro,S., Gasteiger,E., Huang,H.Z., Lopez,R., Magrane,M., Martin,M.J., Natale,D.A., O'Donovan,C., Redaschi,N. and Yeh,L.S.L. (2004*b*) UniProt: the universal protein knowledgebase. *Nucleic Acids Research,* **32**, D115–D119.

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. *Nature Genetics,* **25** (1), 25–29.

Aung,Z., Tan,S.H., Ng,S.K. and Tan,K.L. (2008) PPiClust: efficient clustering of 3d protein-protein interaction interfaces. *Journal of Bioinformatics and Computational Biology,* **6** (3), 415–433.

Bader,G.D. and Hogue,C.W.V. (2000) BIND - a data specification for storing and describing biomolecular interactions, molecular complexes and pathways. *Bioinformatics,* **16** (5), 465–477.

Bartlett,G.J., Porter,C.T., Borkakoti,N. and Thornton,J.M. (2002) Analysis of catalytic residues in enzyme active sites. *Journal of Molecular Biology,* **324** (1), 105–121.

Bateman,A., Birney,E., Cerruti,L., Durbin,R., Etwiller,L., Eddy,S.R., Griffiths-Jones,S., Howe,K.L., Marshall,M. and Sonnhammer,E.L.L. (2002) The Pfam protein families database. *Nucleic Acids Research,* **30** (1), 276–280.

Bergmann,R., Kolodner,J.L. and Plaza,E. (2005) Representation in case-based reasoning. *Knowledge Engineering Review,* **20(3)**, 209–213.

Berman,H.M., Battistuz,T., Bhat,T.N., Bluhm,W.F., Bourne,P.E., Burkhardt,K., Iype,L., Jain,S., Fagan,P., Marvin,J., Padilla,D., Ravichandran,V., Schneider,B., Thanki,N., Weissig,H., Westbrook,J.D. and Zardecki,C. (2002) The protein data bank. *Acta Crystallographica,* **D58**, 899–907.

Berman,H.M., Kleywegt,G.J., Nakamura,H. and Markley,J.L. (2012) The Protein Data Bank at 40: reflecting on the past to prepare for the future. *Structure,* **20** (3), 391–396.

Bernauer,J., Bahadur,R.P., Rodier,F., Janin,J. and Poupon,A. (2008) DiMoVo: a voronoi tessellation-based method for discriminating crystallographic and biological protein-protein interactions. *Bioinformatics,* **24** (5), 652–658.

Bogan,A.A. and Thorn,K.S. (1998) Anatomy of hot spots in protein interfaces. *Journal of Molecular Biology,* **280** (1), 1–9.

Bonvin,A.M. (2006) Flexible protein-protein docking. *Current Opinion in Structural Biology,* **16** (2), 194–200.

Bradford,J.R. and Westhead,D.R. (2005) Improved prediction of protein-protein binding sites using a support vector machines approach. *Bioinformatics,* **21** (8), 1487–1494.

Branden,C.I. and Tooze,J. (1999) *Introduction to Protein Structure*. Garland Publishing.

Brown,K.R. and Jurisica,I. (2005) Online predicted human interaction database. *Bioinformatics,* **21** (9), 2076–2082.

Bru,C., Courcelle,E., Carrere,S., Beausse,Y., Dalmar,S. and Kahn,D. (2005) The ProDom database of protein domain families: more emphasis on 3D. *Nucleic Acids Research,* **33**, D212–D215.

Caffrey,D.R., Somaroo,S., Hughes,J.D., Mintseris,J. and Huang,E.S. (2004) Are protein-protein interfaces more conserved in sequence than the rest of the protein surface? *Protein Science,* **13** (1), 190–202.

Carugo,O. and Argos,P. (1997) Protein-protein crystal-packing contacts. *Protein Science,* **6** (10), 2261–2263.

Castrignano,T., De Meo,P.D., Cozzetto,D., Talamo,I.G. and Tramontano,A. (2006) The PMDB protein model database. *Nucleic Acids Research,* **34**, D306–D309.

Chakrabarti,P. and Janin,J. (2002) Dissecting protein-protein recognition sites. *Proteins: Structure Function and Genetics,* **47** (3), 334–343.

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 Research,* **35**, D572–D574.

Chen,J.Y., Mamidipalli,S. and Huan,T.X. (2009) HAPPI: an online database of comprehensive human annotated and predicted protein interactions. *BMC Genomics,* **10**, S16.

Chen,R. and Weng,Z.P. (2002) Docking unbound proteins using shape complementarity, desolvation, and electrostatics. *Proteins: Structure Function and Genetics,* **47** (3), 281–294.

Choi,Y.S., Yang,J.S., Choi,Y., Ryu,S.H. and Kim,S. (2009) Evolutionary conservation in multiple faces of protein interaction. *Proteins: Structure Function and Bioinformatics,* **77** (1), 14–25.

Chothia,C. (1992) Proteins. One thousand families for the molecular biologist. *Nature,* **357** (6379), 543–544.

Chothia,C. and Lesk,A.M. (1986) The relation between the divergence of sequence and structure in proteins. *EMBO Journal,* **5**, 823–826.

Chuang,G.Y., Kozakov,D., Brenke,R., Comeau,S.R. and Vajda,S. (2008) DARS (decoys as the reference state) potentials for protein-protein docking. *Biophysical Journal,* **95** (9), 4217–4227.

Copley,R.R., Doerks,T., Letunic,I. and Bork,P. (2002) Protein domain analysis in the era of complete genomes. *FEBS Letters,* **513** (1), 129–134.

Cuff,A.L., Sillitoe,I., Lewis,T., Redfern,O.C., Garratt,R., Thornton,J. and Orengo,C.A. (2009) The CATH classification revisited-architectures reviewed and new ways to characterize structural divergence in superfamilies. *Nucleic Acids Research,* **37**, D310–D314.

Davis,F.P. and Sali,A. (2005) PIBASE: a comprehensive database of structurally defined protein interfaces. *Bioinformatics,* **21**, 1901–1907.

De,S., Krishnadev,O., Srinivasan,N. and Rekha,N. (2005) Interaction preferences across protein-protein interfaces of obligatory and non-obligatory components are different. *BMC Structural Biology,* **5**, 15.

de Lima Morais,D.A., Fang,H., Rackham,O.J., Wilson,D., Pethica,R., Chothia,C. and Gough,J. (2011) SUPERFAMILY 1.75 including a domain-centric gene ontology method. *Nucleic Acids Research,* **39**, D427–D434.

de Mántaras,R.L., McSherry,D., Bridge,D.G., Leake,D.B., Smyth,B., Craw,S., Faltings,B., Maher,M.L., Cox,M.T., Forbus,K.D., Keane,M.T., Aamodt,A. and Watson,I.D. (2005) Retrieval, reuse, revision and retention in case-based reasoning. *Knowledge Engineering Review,* **20** (3), 215–240.

de Vries,S.J. and Bonvin,A.M.J.J. (2008) How proteins get in touch: interface prediction in the study of biomolecular complexes. *Current Protein & Peptide Science,* **9** (4), 394–406.

de Vries,S.J., Melquiond,A.S.J., Kastritis,P.L., Karaca,E., Bordogna,A., van Dijk,M., Rodrigues,J.P.G.L.M. and Bonvin,A.M.J.J. (2010) Strengths anbd weaknesses of data-driven docking in critical assessment of predicted interactions. *Proteins: Structure Function and Bioinformatics,* **78**, 3242–3249.

de Vries,S.J., van Dijk,A.D. and Bonvin,A.M. (2006) WHISCY: what information does surface conservation yield? Application to data-driven docking. *Proteins: Structure Function and Bioinformatics,* **63** (3), 479–489.

Devore,J.L. (2008) *Probability and Statistics for Engineering and the Sciences*. Thomson Brooks/Cole.

Dominguez,C., Boelens,R. and Bonvin,A.M.J.J. (2003) HADDOCK: a protein-protein docking approach based on biochemical or biophysical information. *Journal of the American Chemical Society,* **125** (7), 1731–1737.

Douguet,D., Chen,H.C., Tovchigrechko,A. and Vakser,I.A. (2006) Dockground resource for studying protein-protein interfaces. *Bioinformatics,* **22**, 2612–2618.

Ezkurdia,I., Bartoli,L., Fariselli,P., Casadio,R., Valencia,A. and Tress,M.L. (2009) Progress and challenges in predicting protein-protein interaction sites. *Briefings in Bioinformatics,* **10** (3), 233–246.

Faure,G., Andreani,J. and Guerois,R. (2012) InterEvol database: exploring the structure and evolution of protein complex interfaces. *Nucleic Acids Research,* **40**, D847–D856.

Fayyad,U., Piatetsky-Shapiro,G. and Smyth,P. (1996) From data mining to knowledge discovery in databases. *AI Magazine,* **17**, 37–54.

Fedyukina,D.V. and Cavagnero,S. (2011) Protein folding at the exit tunnel. *Annual Review of Biophysics,* **40**, 337–359.

Fernandez-Recio,J. (2011) Prediction of protein binding sites and hot spots. *WIREs Computational Molecular Science,* **1** (5), 680–698.

Finn,R.D., Mistry,J., Schuster-Bockler,B., Griffiths-Jones,S., Hollich,V., Lassmann,T., Moxon,S., Marshall,M., Khanna,A., Durbin,R., Eddy,S.R., Sonnhammer,E.L. and Bateman,A. (2006) Pfam: clans, web tools and services. *Nucleic Acids Research,* **34**, D247–D251.

Finn,R.D., Mistry,J., Tate,J., Coggill,P., Heger,A., Pollington,J.E., Gavin,O.L., Gunasekaran,P., Ceric,G., Forslund,K., Holm,L., Sonnhammer,E.L.L., Eddy,S.R. and Bateman,A. (2010) The Pfam protein families database. *Nucleic Acids Research,* **38**, D211–D222.

Freddolino,P.L., Harrison,C.B., Liu,Y. and Schulten,K. (2010) Challenges in protein folding simulations: Timescale, representation, and analysis. *Nature Physics,* **6** (10), 751–758.

Gabb,H.A., Jackson,R.M. and Sternberg,M.J. (1997) Modelling protein docking using shape complementarity, electrostatics and biochemical information. *Journal of Molecular Biology,* **272** (1), 106–120.

Gao,M. and Skolnick,J. (2010) Structural space of protein-protein interfaces is degenerate, close to complete, and highly connected. *Proceedings of the National Academy of Science,* **107** (52), 22517–22522.

Ghoorah,A.W., Devignes,M.D., Smaïl-Tabbone,M. and Ritchie,D.W. (2011) Spatial clustering of protein binding sites for template based protein docking. *Bioinformatics,* **27** (20), 2820–2827.

Glaser,F., Steinberg,D.M., Vakser,I.A. and Ben-Tal,N. (2001) Residue frequencies and pairing preferences at protein-protein interfaces. *Proteins: Structure Function and Genetics,* **43** (2), 89–102.

Govindarajan,S., Recabarren,R. and Goldstein,R.A. (1999) Estimating the total number of protein folds. *Proteins: Structure Function and Genetics,* **35** (4), 408–414.

Gowri,V.S., Pandit,S.B., Karthik,P.S., Srinivasan,N. and Balaji,S. (2003) Integration of related sequences with protein three-dimensional structural families in an updated version of PALI database. *Nucleic Acids Research,* **31** (1), 486–488.

Grunberg,R., Nilges,M. and Leckner,J. (2007) Biskit - a software platform for structural bioinformatics. *Bioinformatics,* **23** (6), 769–770.

Guharoy,M. and Chakrabarti,P. (2005) Conservation and relative importance of residues across protein-protein interfaces. *Proceedings of the National Academy of Science,* **102**, 15447–15452.

Guharoy,M. and Chakrabarti,P. (2007) Secondary structure based analysis and classification of biological interfaces: identification of binding motifs in protein-protein interactions. *Bioinformatics,* **23** (15), 1909–1918.

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 Research,* **34**, D436–D441.

Günther,S., May,P., Hoppe,A., Frommel,C. and Preissner,R. (2007) Docking without docking: ISEARCH – prediction of interactions using known interfaces. *Proteins: Structure Function and Bioinformatics,* **69** (4), 839–844.

Halperin,I., Ma,B., Wolfson,H. and Nussinov,R. (2002) Principles of docking: an overview of search algorithms and a guide to scoring functions. *Proteins: Structure Function and Genetics,* **47**, 409–443.

Han,J. and Kamber,M. (2005) *Data mining : concepts and techniques*. Morgan Kaufmann, San Francisco.

Hand,D.J., Smyth,P. and Mannila,H. (2001) *Principles of data mining*. MIT Press, Cambridge, MA, USA.

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 Apweller,R. (2004) IntAct: an open source molecular interaction database. *Nucleic Acids Research,* **32**, D452–D455.

Higurashi,M., Ishida,T. and Kinoshita,K. (2008) Identification of transient hub proteins and the possible structural basis for their multiple interactions. *Protein Science,* **17** (1), 72–78.

Higurashi,M., Ishida,T. and Kinoshita,K. (2009) PiSite: a database of protein interaction sites using multiple binding states in the PDB. *Nucleic Acids Research,* **37**, D360–D364.

Holm,L. and Rosenstrom,P. (2010) Dali server: conservation mapping in 3D. *Nucleic Acids Research,* **38**, W545–W549.

Holm,L. and Sander,C. (1998) Removing near-neighbour redundancy from large protein sequence collections. *Bioinformatics,* **14** (5), 423–429.

Hoppe,A. and Frömmel,C. (2003) Needlehaystack: a program for the rapid recognition of local structures in large sets of atomic coordinates. *Journal of Applied Crystallography,* **36**, 1090–1097.

Hu,Z.J., Ma,B.Y., Wolfson,H. and Nussinov,R. (2000) Conservation of polar residues as hot spots at protein interfaces. *Proteins-Structure Function And Genetics,* **39** (4), 331–342.

Humphrey,W., Dalke,A. and Schulten,K. (1996) VMD: visual molecular dynamics. *Journal of Molecular Graphics,* **14** (1), 33–38.

Hunter,S., Jones,P., Mitchell,A., Apweiler,R., Attwood,T.K., Bateman,A., Bernard,T., Binns,D., Bork,P., Burge,S., de Castro,E., Coggill,P., Corbett,M., Das,U., Daugherty,L., Duquenne,L., Finn,R.D., Fraser,M., Gough,J., Haft,D., Hulo,N., Kahn,D., Kelly,E., Letunic,I., Lonsdale,D., Lopez,R., Madera,M., Maslen,J., McAnulla,C., McDowall,J., McMenamin,C., Mi,H., Mutowo-Muellenet,P., Mulder,N., Natale,D., Orengo,C., Pesseat,S., Punta,M., Quinn,A.F., Rivoire,C., Sangrador-Vegas,A., Selengut,J.D., Sigrist,C.J., Scheremetjew,M., Tate,J., Thimmajanarthanan,M., Thomas,P.D., Wu,C.H., Yeats,C. and Yong,S.Y. (2012) InterPro in 2011: new developments in the family and domain prediction database. *Nucleic Acids Research,* **40**, D306–D312.

Hwang,H., Vreven,T., Janin,J. and Weng,Z. (2010) Protein-protein docking benchmark version 4.0. *Proteins: Structure Function and Bioinformatics,* **78** (15), 3111–3114.

Janin,J. and Chothia,C. (1990) The structure of protein-protein recognition sites. *The Journal of Biological Chemistry,* **265** (27), 16027–30.

Janin,J. and Rodier,F. (1995) Protein-protein interaction at crystal contacts. *Proteins: Structure Function and Genetics,* **23** (4), 580–587.

Jefferson,E.R., Walsh,T.P., Roberts,T.J. and Barton,G.J. (2007) SNAPPI-DB: a database and API of Structures, iNterfaces and Alignments for Protein-Protein Interactions. *Nucleic Acids Research,* **35**, D580–D589.

Jensen,L.J., Kuhn,M., Stark,M., Chaffron,S., Creevey,C., Muller,J., Doerks,T., Julien,P., Roth,A., Simonovic,M., Bork,P. and von Mering,C. (2009) STRING 8 - a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Research,* **37**, D412–D416.

Jones,S., Marin,A. and Thornton,J.M. (2000) Protein domain interfaces: characterization and comparison with oligomeric protein interfaces. *Protein Engineering,* **13** (2), 77–82.

Jones,S. and Thornton,J.M. (1996) Principles of protein-protein interactions. *Proceedings of the National Academy of Science,* **93** (1), 13–20.

Jones,S. and Thornton,J.M. (1997) Analysis of protein-protein interaction sites using surface patches. *Journal of Molecular Biology,* **272** (1), 121–132.

Kabsch,W. and Sander,C. (1983) Dictionary of protein secondary structure - pattern-recognition of hydrogen-bonded and geometrical features. *Biopolymers,* **22** (12), 2577–2637.

Keskin,O., Gursoy,A., Ma,B. and Nussinov,R. (2008) Principles of protein-protein interactions: what are the preferred ways for proteins to interact? *Chemical Reviews,* **108** (4), 1225–1244.

Keskin,O., Ma,B.Y. and Nussinov,R. (2005) Hot regions in protein-protein interactions: the organization and contribution of structurally conserved hot spot residues. *Journal of Molecular Biology,* **345** (5), 1281–1294.

Keskin,O. and Nussinov,R. (2007) Similar binding sites and different partners: implications to shared proteins in cellular pathways. *Structure,* **15**, 341–354.

Keskin,O., Tsai,C.J., Wolfson,H. and Nussinov,R. (2004) A new, structurally nonredundant, diverse data set of protein-protein interfaces and its implications. *Protein Science,* **13** (4), 1043–1055.

Kim,W.K., Henschel,A., Winter,C. and Schroeder,M. (2006) The many faces of protein-protein interactions: a compendium of interface geometry. *PLoS Computational Biology,* **2**, 1151–1164.

Kolodner,J.L. (1992) An introduction to case-based reasoning. *Artificial Intelligence Reviews,* **6**, 3–34.

Korkin,D., Davis,F.P., Alber,F., Luong,T., Shen,M.Y., Lucic,V., Kennedy,M.B. and Sali,A. (2006) Structural modeling of protein interactions by analogy: application to PSD-95. *PLoS Computational Biology,* **2** (11), e153.

Korkin,D., Davis,F.P. and Sali,A. (2005) Localization of protein-binding sites within families of proteins. *Protein Science,* **14**, 2350–2360.

Kresge,N., Simoni,R.D. and Hill,R.L. (2006) The thermodynamic hypothesis of protein folding: the work of christian anfinsen. *Journal of Biological Chemistry,* **281** (14), e11.

Krissinel,E. and Henrick,K. (2004) Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. *Acta Crystallographica,* **D60**, 2256–2268.

Krissinel,E. and Henrick,K. (2007) Inference of macromolecular assemblies from crystalline state. *Journal of Molecular Biology,* **372** (3), 774–797.

Król,M., Tournier,A. and Bates,P. (2009) In-silico docking: predicting protein-protein interactions. In *Structure-Function Relation In Proteins*, (Roterman,I., ed.),. Transworld Research Network.

Kundrotas,P.J. and Alexov,E. (2006) Predicting 3D structures of transient protein-protein complexes by homology. *BBA – Proteins & Proteomics,* **1764** (9), 1498–1511.

Kundrotas,P.J. and Alexov,E. (2007) PROTCOM: searchable database of protein complexes enhanced with domain-domain structures. *Nucleic Acids Research,* **35**, D575–D579.

Kundrotas,P.J., Lensink,M.F. and Alexov,E. (2008) Homology-based modeling of 3D structures of protein-protein complexes using alignments of modified sequence profiles. *International Journal of Biological Macromolecules,* **43** (2), 198–208.

Kundrotas,P.J., Zhu,Z., Janin,J. and Vakser,I.A. (2012) Templates are available to model nearly all complexes of structurally characterized proteins. *Proceedings of the National Academy of Science,* **109** (24), 9438–9441.

Kundrotas,P.J., Zhu,Z.W. and Vakser,I.A. (2010) GWIDD: genome-wide protein docking database. *Nucleic Acids Research,* **38**, D513–D517.

Launay,G. and Simonson,T. (2008) Homology modelling of protein-protein complexes: a simple method and its possibilities and limitations. *BMC Bioinformatics,* **9**, 427.

Lees,J., Yeats,C., Redfern,O., Clegg,A. and Orengo,C. (2010) Gene3D: merging structure and function for a Thousand genomes. *Nucleic Acids Research,* **38**, D296–D300.

Lensink,M.F. and Wodak,S.J. (2010) Docking and scoring protein interactions: CAPRI 2009. *Proteins: Structure Function and Bioinformatics,* **78** (15), 3073–3084.

Letunic,I., Doerks,T. and Bork,P. (2009) SMART 6: recent updates and new developments. *Nucleic Acids Research,* **37**, D229–D232.

Levitt,M. and Chothia,C. (1976) Structural patterns in globular proteins. *Nature,* **261** (5561), 552–558.

Levy,E.D. (2010) A simple definition of structural regions in proteins and its use in analyzing interface evolution. *Journal of Molecular Biology,* **403** (4), 660–670.

Levy,E.D., Pereira-Leal,J.B., Chothia,C. and Teichmann,S.A. (2006) 3D complex: a structural classification of protein complexes. *PLoS Computational Biology,* **2** (11), 1395–1406.

Li,J.J., Huang,D.S., Wang,B. and Chen,P. (2006) Identifying protein-protein interfacial residues in heterocomplexes using residue conservation scores. *International Journal of Biological Macromolecules,* **38** (3-5), 241–247.

Li,W., Jaroszewski,L. and Godzik,A. (2001) Clustering of highly homologous sequences to reduce the size of large protein databases. *Bioinformatics,* **17** (3), 282–283.

Lichtarge,O., Bourne,H.R. and Cohen,F.E. (1996) An evolutionary trace method defines binding surfaces common to protein families. *Journal of Molecular Biology,* **257** (2), 342–358.

Lima,T., Auchincloss,A.H., Coudert,E., Keller,G., Michoud,K., Rivoire,C., Bulliard,V., de Castro,E., Lachaize,C., Baratin,D., Phan,I., Bougueleret,L. and Bairoch,A. (2009) HAMAP: a database of completely sequenced microbial proteome sets and manually curated microbial protein families in UniProtKB/Swiss-Prot. *Nucleic Acids Research,* **37**, D471–D478.

Lo Conte,L., Chothia,C. and Janin,J. (1999) The atomic structure of protein-protein recognition sites. *Journal of Molecular Biology,* **285** (5), 2177–2198.

Lu,L., Lu,H. and Skolnick,J. (2002) Multiprospector: an algorithm for the prediction of protein-protein interactions by multimeric threading. *Proteins: Structure Function and Genetics,* **49** (3), 350–364.

Ma,B.Y., Elkayam,T., Wolfson,H. and Nussinov,R. (2003) Protein-protein interactions: structurally conserved residues distinguish between binding sites and exposed protein surfaces. *Proceedings of the National Academy of Science,* **100** (10), 5772–5777.

Macindoe,G., Mavridis,L., Venkatraman,V., Devignes,M.D. and Ritchie,D.W. (2010) HexServer: an FFT-based protein docking server powered by graphics processors. *Nucleic Acids Research,* **38**, W445–W449.

Marchler-Bauer,A., Anderson,J.B., Chitsaz,F., Derbyshire,M.K., DeWeese-Scott,C., Fong,J.H., Geer,L.Y., Geer,R.C., Gonzales,N.R., Gwadz,M., He,S., Hurwitz,D.I., Jackson,J.D., Ke,Z., Lanczycki,C.J., Liebert,C.A., Liu,C., Lu,F., Lu,S., Marchler,G.H., Mullokandov,M., Song,J.S., Tasneem,A., Thanki,N., Yamashita,R.A., Zhang,D., Zhang,N. and Bryant,S.H. (2009) CDD: specific functional annotation with the Conserved Domain Database. *Nucleic acids research,* **37**, D205–D210.

Martin,A.C.R. (2005) Mapping PDB chains to UniProtKB entries. *Bioinformatics,* **21** (23), 4297–4301.

McDowall,M.D., Scott,M.S. and Barton,G.J. (2009) PIPs: human protein-protein interaction prediction database. *Nucleic Acids Research,* **37**, D651–D656.

McLachlan,A.D. (1982) Rapid comparison of protein structures. *Acta Crystallographica,* **A38** (6), 871–873.

Melquiond,A.S.J., Karaca,E., Kastritis,P.L. and Bonvin,M. (2012) Next challenges in protein-protein docking: from proteome to interactomeand beyond. *WIREs Compututational Molecular Sciences,* **2**, 642–651.

Mendez,R., Leplae,R., Lensink,M.F. and Wodak,S.J. (2005) Assessment of CAPRI predictions in rounds 3-5 shows progress in docking procedures. *Proteins: Structure Function and Bioinformatics,* **60**, 150–169.

Mi,H., Dong,Q., Muruganujan,A., Gaudet,P., Lewis,S. and Thomas,P.D. (2010) PANTHER version 7: improved phylogenetic trees, orthologs and collaboration with the Gene Ontology Consortium. *Nucleic Acids Research,* **38**, D204–D210.

Moont,G., Gabb,H.A. and Sternberg,M.J. (1999) Use of pair potentials across protein interfaces in screening predicted docked complexes. *Proteins: Structure Function and Genetics,* **35** (3), 364–373.

Moreira,I.S., Fernandes,P.A. and Ramos,M.J. (2007) Hot spots–a review of the protein-protein interface determinant amino-acid residues. *Proteins: Structure Function and Bioinformatics,* **68** (4), 803–812.

Moult,J., Fidelis,K., Kryshtafovych,A. and Tramontano,A. (2011) Critical assessment of methods of protein structure prediction (CASP)–round IX. *Proteins: Structure Function and Bioinformatics,* **79 Suppl 10**, 1–5.

Movshovitz-Attias,D., London,N. and Schueler-Furman,O. (2010) On the use of structural templates for high-resolution docking. *Proteins: Structure Function and Bioinformatics,* **78** (8), 1939–1949.

Mukherjee,S. and Zhang,Y. (2011) Protein-protein complex structure predictions by multimeric threading and template recombination. *Structure,* **19** (7), 955–966.

Murakami,Y. and Mizuguchi,K. (2010) Applying the Naïve Bayes classifier with kernel density estimation to the prediction of protein-protein interaction sites. *Bioinformatics,* **26** (15), 1841–1848.

Murzin,A.G., Brenner,S.E., Hubbard,T. and Chothia,C. (1995) SCOP – a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology,* **247** (4), 536–540.

Napoli,A. (2005). A smooth introduction to symbolic methods for knowledge discovery. Inria technical report inria-00001210.

Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology,* **48**, 443–453.

Neuvirth,H., Raz,R. and Schreiber,G. (2004) Promate: a structure based prediction program to identify the location of protein-protein binding sites. *Journal of Molecular Biology,* **338** (1), 181–199.

Nikolskaya,A.N., Arighi,C.N., Huang,H., Barker,W.C. and Wu,C.H. (2006) PIRSF family classification system for protein functional and evolutionary analysis. *Evolutionary Bioinformatics Online,* **2**, 197–209.

Nooren,I.M.A. and Thornton,J.M. (2003) Diversity of protein-protein interactions. *EMBO Journal,* **22** (14), 3486–3492.

Ofran,Y. and Rost,B. (2003*a*) Analysing six types of protein-protein interfaces. *Journal of Molecular Biology,* **325** (2), 377–387.

Ofran,Y. and Rost,B. (2003*b*) Predicted protein-protein interaction sites from local sequence information. *FEBS Letters,* **544** (1-3), 236–239.

Ofran,Y. and Rost,B. (2007*a*) ISIS: interaction sites identified from sequence. *Bioinformatics,* **23** (2), E13–E16.

Ofran,Y. and Rost,B. (2007*b*) Protein-protein interaction hotspots carved into sequences. *PLoS Computational Biology,* **3** (7), e119.

Orengo,C.A., Michie,A.D., Jones,S., Jones,D.T., Swindells,M.B. and Thornton,J.M. (1997) CATH - a hierarchic classification of protein domain structures. *Structure,* **5** (8), 1093–1108.

Pandey,A. and Mann,M. (2000) Proteomics to study genes and genomes. *Nature,* **405** (6788), 837–846.

Patil,A., Kinoshita,K. and Nakamura,H. (2010) Hub promiscuity in protein-protein interaction networks. *International Journal of Molecular Sciences,* **11** (4), 1930–1943.

Pazos,F., HelmerCitterich,M., Ausiello,G. and Valencia,A. (1997) Correlated mutations contain information about protein-protein interaction. *Journal of Molecular Biology,* **271** (4), 511–523.

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 Research,* **13** (10), 2363–2371.

Petrey,D., Xiang,Z., Tang,C.L., Xie,L., Gimpelev,M., Mitros,T., Soto,C.S., Goldsmith-Fischman,S., Kernytsky,A., Schlessinger,A., Koh,I.Y., Alexov,E. and Honig,B. (2003) Using multiple structure alignments, fast model building, and energetic analysis in fold recognition and homology modeling. *Proteins: Structure Function and Genetics,* **53 Suppl 6**, 430–435.

Petsko,G.A. and Dagmar,R. (2004) *Protein Structure and Function*. New Science Press.

Pieper,U., Eswar,N., Webb,B.M., Eramian,D., Kelly,L., Barkan,D.T., Carter,H., Mankoo,P., Karchin,R., Marti-Renom,M.A., Davis,F.P. and Sali,A. (2009) MODBASE, a database of annotated comparative protein structure models and associated resources. *Nucleic Acids Research,* **37**, D347–D354.

Ponstingl,H., Kabir,T. and Thornton,J.M. (2003) Automatic inference of protein quaternary structure from crystals. *Journal of Applied Crystallography,* **36**, 1116–1122.

Qin,S. and Zhou,H.X. (2007*a*) meta-PPISP: a meta web server for protein-protein interaction site prediction. *Bioinformatics,* **23** (24), 3386–3387.

Qin,S. and Zhou,H.X. (2007*b*) A holistic approach to protein docking. *Proteins: Structure Function and Bioinformatics,* **69** (4), 743–749.

Raghavachari,B., Tasneem,A., Przytycka,T.M. and Jothi,R. (2008) DOMINE: a database of protein domain interactions. *Nucleic Acids Research,* **36**, D656–D661.

Ritchie,D.W. (2008) Recent progress and future directions in protein-protein docking. *Current Protein & Peptide Science,* **9** (1), 1–15.

Ritchie,D.W. and Kemp,G.J.L. (2000) Protein docking using spherical polar Fourier correlations. *Proteins: Structure Function and Genetics,* **39** (2), 178–194.

Ritchie,D.W., Kozakov,D. and Vajda,S. (2008) Accelerating and focusing protein-protein docking correlations using multi-dimensional rotational FFT generating functions. *Bioinformatics,* **24** (17), 1865–1873.

Rousseeuw,P.J. (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics,* **20**, 53 – 65.

Sacquin-Mora,S., Carbone,A. and Lavery,R. (2008) Identification of protein interaction partners and protein-protein interaction sites. *Journal of Molecular Biology,* **382** (5), 1276–1289.

Sali,A. and Blundell,T.L. (1993) Comparative protein modelling by satisfaction of spatial restraints. *Journal of Molecular Biology,* **234** (3), 779–815.

Sali,A., Glaeser,R., Earnest,T. and Baumeister,W. (2003) From words to literature in structural proteomics. *Nature,* **422** (6928), 216–225.

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 Research,* **32**, D449–D451.

Schneidman-Duhovny,D., Inbar,Y., Nussinov,R. and Wolfson,H.J. (2005) Geometry-based flexible and symmetric protein docking. *Proteins: Structure Function and Bioinformatics,* **60** (2), 224–231.

Schneidman-Duhovny,D., Nussinov,R. and Wolfson,H.J. (2004) Predicting molecular interactions in silico: ii. protein-protein and protein-drug docking. *Current Medicinal Chemistry,* **11** (1), 91–107.

Selengut,J.D., Haft,D.H., Davidsen,T., Ganapathy,A., Gwinn-Giglio,M., Nelson,W.C., Richter,A.R. and White,O. (2007) TIGRFAMs and genome properties: tools for the assignment of molecular function and biological process in prokaryotic genomes. *Nucleic Acids Research,* **35**, D260–D264.

Shen,M.Y. and Sali,A. (2006) Statistical potential for assessment and prediction of protein structures. *Protein Science,* **15** (11), 2507–2524.

Shoemaker,B.A. and Panchenko,A.R. (2007*a*) Deciphering protein-protein interactions. Part II. Computational methods to predict protein and domain interaction partners. *PLoS Computational Biology,* **3** (4), e43.

Shoemaker,B.A. and Panchenko,A.R. (2007*b*) Deciphering protein-protein interactions. Part I. Experimental techniques and databases. *PLoS Computational Biology,* **3** (3), e42.

Shoemaker,B.A., Panchenko,A.R. and Bryant,S.H. (2006) Finding biologically relevant protein domain interactions: conserved binding mode analysis. *Protein Science,* **15** (2), 352–361.

Shoemaker,B.A., Zhang,D.C., Thangudu,R.R., Tyagi,M., Fong,J.H., Marchler-Bauer,A., Bryant,S.H., Madej,T. and Panchenko,A.R. (2010) Inferred biomolecular interaction server-a web server to analyze and predict protein interacting partners and binding sites. *Nucleic Acids Research,* **38**, D518–D524.

Shulman-Peleg,A., Mintz,S., Nussinov,R. and Wolfson,H.J. (2004) Protein-protein interfaces: recognition of similar spatial and chemical organizations. *LNCS Algorithms in Bioinformatics,* **3240**, 194–205.

Shulman-Peleg,A., Nussinov,R. and Wolfson,H.J. (2005) SiteEngines: recognition and comparison of binding sites and protein-protein interfaces. *Nucleic Acids Research,* **33**, W337–W341.

Sigrist,C.J., Cerutti,L., de Castro,E., Langendijk-Genevaux,P.S., Bulliard,V., Bairoch,A. and Hulo,N. (2010) PROSITE, a protein domain database for functional characterization and annotation. *Nucleic Acids Research,* **38**, D161–D166.

Sigrist,C.J., De Castro,E., Langendijk-Genevaux,P.S., Le Saux,V., Bairoch,A. and Hulo,N. (2005) ProRule: a new database containing functional and structural information on PROSITE profiles. *Bioinformatics,* **21** (21), 4060–4066.

Sikic,M., Tomic,S. and Vlahovicek,K. (2009) Prediction of protein-protein interaction sites in sequences and 3D structures by random forests. *PLoS Computational Biology,* **5** (1), e1000278.

Simons,K.T., Kooperberg,C., Huang,E. and Baker,D. (1997) Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. *Journal of Molecular Biology,* **268** (1), 209–225.

Smith,G.R. and Sternberg,M.J.E. (2002) Prediction of protein-protein interactions by docking methods. *Current Opinion in Structural Biology,* **12** (1), 28–35.

Stark,C., Breitkreutz,B.J., Reguly,T., Boucher,L., Breitkreutz,A. and Tyers,M. (2006) Biogrid: a general repository for interaction datasets. *Nucleic Acids Research,* **34**, D535–D539.

Stein,A., Ceol,A. and Aloy,P. (2010) 3did: identification and classification of domain-based interactions of known three-dimensional structure. *Nucleic Acids Research,* **39**, D718–D723.

Stein,A., Mosca,R. and Aloy,P. (2011) Three-dimensional modeling of protein interactions and complexes is going 'omics. *Current Opinion in Structural Biology,* **21**, 200–208.

Stein,A., Panjkovich,A. and Aloy,P. (2009) 3DID update: domain-domain and peptide-mediated interactions of known 3D structure. *Nucleic Acids Research,* **37**, D300–D304.

Stein,A., Russell,R.B. and Aloy,P. (2005) 3DID: interacting protein domains of known three-dimensional structure. *Nucleic Acids Research,* **33**, D413–D417.

Teyra,J., Doms,A., Schroeder,M. and Pisabarro,M.T. (2006) SCOWLP: a web-based database for detailed characterization and visualization of protein interfaces. *BMC Bioinformatics,* **7**, 104.

Teyra,J., Paszkowski-Rogacz,M., Anders,G. and Pisabarro,T.M. (2008) SCOWLP classification: structural comparison and analysis of protein binding regions. *BMC Bioinformatics,* **9**, 9.

Thorn,K.S. and Bogan,A.A. (2001) ASEdb: a database of alanine mutations and their effects on the free energy of binding in protein interactions. *Bioinformatics,* **17** (3), 284–285.

Tsai,C.J., Ma,B. and Nussinov,R. (2009) Protein-protein interaction networks: how can a hub protein bind so many different partners? *Trends in Biochemical Sciences,* **34** (12), 594–600.

Tuncbag,N., Kar,G., Keskin,O., Gursoy,A. and Nussinov,R. (2009) A survey of available tools and web servers for analysis of protein-protein interactions and interfaces. *Briefings in Bioinformatics,* **10** (3), 217–232.

Vajda,S. and Kozakov,D. (2009) Convergence and combination of methods in protein-protein docking. *Current Opinion in Structural Biology,* **19** (2), 164–170.

Vakser,I.A. and Aflalo,C. (1994) Hydrophobic docking: a proposed enhancement to molecular recognition techniques. *Proteins: Structure Function and Genetics,* **20** (4), 320–329.

Valencia,A. and Pazos,F. (2002) Computational methods for the prediction of protein interactions. *Current Opinion in Structural Biology,* **12** (3), 368–373.

van Dijk,A.D., Boelens,R. and Bonvin,A.M. (2005) Data-driven docking for the study of biomolecular complexes. *FEBS Journal,* **272** (2), 293–312.

Venkatesan,K., Rual,J.F., Vazquez,A., Stelzl,U., Lemmens,I., Hirozane-Kishikawa,T., Hao,T., Zenkner,M., Xin,X., Goh,K.I., Yildirim,M.A., Simonis,N., Heinzmann,K., Gebreab,F., Sahalie,J.M., Cevik,S., Simon,C., de Smet,A.S., Dann,E., Smolyar,A., Vinayagam,A., Yu,H., Szeto,D., Borick,H., Dricot,A., Klitgord,N., Murray,R.R., Lin,C., Lalowski,M., Timm,J., Rau,K., Boone,C., Braun,P., Cusick,M.E., Roth,F.P., Hill,D.E., Tavernier,J., Wanker,E.E., Barabasi,A.L. and Vidal,M. (2009) An empirical framework for binary interactome mapping. *Nature Methods,* **6** (1), 83–90.

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** (6887), 399–403.

Wallner,B. and Elofsson,A. (2005) All are not equal: a benchmark of different homology modeling programs. *Protein Science,* **14** (5), 1315–1327.

Wang,C., Bradley,P. and Baker,D. (2007*a*) Protein-protein docking with backbone flexibility. *Journal of Molecular Biology,* **373** (2), 503–519.

Wang,Y.L., Addess,K.J., Chen,J., Geer,L.Y., He,J., He,S.Q., Lu,S.N., Madej,T., Marchler-Bauer,A., Thiessen,P.A., Zhang,N.G. and Bryant,S.H. (2007*b*) MMDB: annotating protein sequences with entrez's 3D-structure database. *Nucleic Acids Research,* **35**, D298–D300.

Ward,J.H. (1963) Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association,* **58**, 236–244.

Wass,M.N., Fuentes,G., Pons,C., Pazos,F. and Valencia,A. (2011) Towards the prediction of protein interaction partners using physical docking. *Molecular Systems Biology,* **7**, 469.

Waterhouse,A.M., Procter,J.B., Martin,D.M., Clamp,M. and Barton,G.J. (2009) Jalview Version 2–a multiple sequence alignment editor and analysis workbench. *Bioinformatics,* **25** (9), 1189–1191.

Watson,I. and Marir,F. (1994) Case-based reasoning: a review. *The Knowledge Engineering Review,* **9**, 327–354.

Watson,I.D. (1997) *Applying case-based reasoning - techniques for the enterprise systems*. Morgan Kaufmann.

Winter,C., Henschel,A., Kim,W.K. and Schroeder,M. (2006) SCOPPI: a structural classification of protein - protein interfaces. *Nucleic Acids Research,* **34**, D310–D314.

Wodak,S.J. and Janin,J. (1978) Computer analysis of protein-protein interaction. *Journal of Molecular Biology,* **124** (2), 323–342.

Xu,D., Tsai,C.J. and Nussinov,R. (1997) Hydrogen bonds and salt bridges across protein-protein interfaces. *Protein Engineering,* **10** (9), 999–1012.

Yang,A.S. and Honig,B. (2000) An integrated approach to the analysis and modeling of protein sequences and structures. I. Protein structural alignment and a quantitative measure for protein structural distance. *Journal of Molecular Biology,* **301** (3), 665–678.

Yu,J., Guo,M., Needham,C.J., Huang,Y., Cai,L. and Westhead,D.R. (2010) Simple sequence-based kernels do not predict protein-protein interactions. *Bioinformatics,* **26** (20), 2610–2614.

Zhang,Y. (2008) Progress and challenges in protein structure prediction. *Current Opinion in Structural Biology,* **18** (3), 342–348.

Zhang,Y. and Skolnick,J. (2004) Automated structure prediction of weakly homologous proteins on a genomic scale. *Proceedings of the National Academy of Science,* **101** (20), 7594–7599.

Zhang,Y. and Skolnick,J. (2005*a*) The protein structure prediction problem could be solved using the current PDB library. *Proceedings of the National Academy of Science,* **102** (4), 1029–1034.

Zhang,Y. and Skolnick,J. (2005*b*) TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research,* **33** (7), 2302–2309.

Zhou,H.X. and Qin,S.B. (2007) Interaction-site prediction for protein complexes: a critical assessment. *Bioinformatics,* **23** (17), 2203–2209.

Zhou,H.X. and Shan,Y.B. (2001) Prediction of protein interaction sites from sequence profile and residue neighbor list. *Proteins: Structure Function and Genetics,* **44** (3), 336–343.

Zhu,H.B., Domingues,F.S., Sommer,I. and Lengauer,T. (2006) NOXclass: prediction of protein-protein interaction types. *BMC Bioinformatics,* **7**, 27.

Zvelebil,M.J., Barton,G.J., Taylor,W.R. and Sternberg,M.J. (1987) Prediction of protein secondary structure and active sites using the alignment of homologous sequences. *Journal of Molecular Biology,* **195** (4), 957–961.

Résumé

L'étude structurale de l'interactome cellulaire peut conduire à des découvertes intéressantes sur les bases moléculaires de certaines pathologies. La modélisation par homologie et l'amarrage de protéines ("protein docking") sont deux approches informatiques pour modéliser la structure tri-dimensionnelle (3D) d'une interaction protéine-protéine (PPI). Des études précédentes ont montré que ces deux approches donnent de meilleurs résultats quand des données expérimentales sur les PPIs sont prises en compte. Cependant, les données PPI ne sont souvent pas disponibles sous une forme facilement accessible, et donc ne peuvent pas être re-utilisées par les algorithmes de prédiction. Cette thèse présente une approche systématique fondée sur l'extraction de connaissances pour représenter et manipuler les données PPI disponibles afin de faciliter l'analyse structurale de l'interactome et d'améliorer les algorithmes de prédiction par la prise en compte des données PPI.

Les contributions majeures de cette thèse sont de : (1) décrire la conception et la mise en oeuvre d'une base de données intégrée KBDOCK qui regroupe toutes les interactions structurales domaine-domaine (DDI); (2) présenter une nouvelle méthode de classification des DDIs par rapport à leur site de liaison dans l'espace 3D et introduit la notion de site de liaison de famille de domaines protéiques ("domain family binding sites" ou DFBS); (3) proposer une classification structurale (inspirée du système CATH) des DFBSs et présenter une étude étendue sur les régularités d'appariement entre DFBSs en terme de structure secondaire; (4) introduire une approche systématique basée sur le raisonnement à partir de cas pour modéliser les structures 3D des complexes protéiques à partir des DDIs connus. Une interface web (http://kbdock.loria.fr) a été développée pour rendre accessible le système KBDOCK.

Le système KBDOCK couvre plus de 2,700 hetero DDIs non-redondantes correspondant à 1,439 DFBSs localisés sur 947 domaines Pfam distincts. KBDOCK a permis de réaliser plusieurs études étendues. Par exemple, KBDOCK a été utilisé pour montrer que: (1) près de 70% de familles de domaines protéiques n'ont qu'un seul DFBS et les autres familles en ont un petit nombre seulement, ce qui suggère que les DDIs re-utilisent souvent les mêmes sites de liaison; (2) plus de 80% de DFBSs interagissent avec une seule famille de domaines protéiques et les autres DFBSs interagissent avec un petit nombre de familles, ce qui indique que la plupart des DFBSs sont principalement monogames dans leur interactions avec les autres domaines protéiques; (3) les DFBSs impliqués dans des interactions présentent des régularités en terme de structure secondaire, ce qui pourrait servir comme un descripteur complémentaire dans la prédiction d'interaction; (4) lorsque les domaines re-utilisent leur DFBS, le docking orienté vient améliorer les prédictions. Ainsi, KBDOCK constitue une ressource unifiée qui permet d'enrichir les connaissances sur l'interactome structural.

Mots-clés: extraction de connaissances à partir des bases de données (ECBD); fouille de données; classification; base de données relationnelle; programmation logique; bioinformatique structurale; interaction protéine-protéine; protein docking; KBDOCK.

Abstract

Understanding how the protein interactome works at a structural level could provide useful insights into the mechanisms of diseases. Comparative homology modelling and *ab initio* protein docking are two computational methods for modelling the three-dimensional (3D) structures of protein-protein interactions (PPIs). Previous studies have shown that both methods give significantly better predictions when they incorporate experimental PPI information. However, in general, PPI information is often not available in an easily accessible way, and cannot be re-used by 3D PPI modelling algorithms. Hence, there is currently a need to develop a reliable framework to facilitate the reuse of PPI data. This thesis presents a systematic knowledge-based approach for representing, describing and manipulating 3D interactions to study PPIs on a large scale and to facilitate knowledge-based modelling of protein-protein complexes.

The main contributions of this thesis are: (1) it describes an integrated database of non-redundant 3D hetero domain interactions; (2) it presents a novel method of describing and clustering DDIs according to the spatial orientations of the binding partners, thus introducing the notion of "domain family-level binding sites" (DFBS); (3) it proposes a structural classification of DFBSs similar to the CATH classification of protein folds, and it presents a study of secondary structure propensities of DFBSs and interaction preferences; (4) it introduces a systematic case-base reasoning approach to model on a large scale the 3D structures of protein complexes from existing structural DDIs. All these contributions have been made publicly available through a web server (http://kbdock.loria.fr).

The KBDOCK database contains 2,721 non-redundant hetero DDIs corresponding to 1,439 DFBSs located in 947 distinct domain families. The KBDOCK database allows large-scale studies. For example, it was used to show that: (1) nearly 70% of protein domain families have just one binding site and the remaining families have a small number of binding sites which suggests that DDIs often re-use the same binding sites; (2) over 80% of DFBSs interact with just one other type of protein domain family, and very few DFBSs interact with more than three different Pfam domain families, which indicates that most DFBSs are primarily monogamous in their structural relationships with other domains; (3) Pfam families often have secondary structure pairing preferences, which might be useful for the prediction of unknown DDIs; (4) when DFBSs are in fact re-used, focused docking improves significantly the docking predictions. Thus, KBDOCK provides a useful framework for enriching our knowledge of the structural interactome.

Keywords: knowledge discovery in databases (KDD); data mining; classification; relational database; logic programming; structural bioinformatics; protein-protein interactions; protein docking; KBDOCK.

Équipe ORPAILLEUR – INRIA Nancy Grand Est

Laboratoire Lorrain de Recherche en Informatique et ses Applications (LORIA)

UMR 7503 - Campus Scientifique - BP 239 - 54506 Vandœuvre-lès-Nancy Cedex