Clustering protein interfaces and their analysis

by

Engin Çukuroğlu

A Thesis Submitted to the
Graduate School of Sciences and Engineering
in Fullfillment of the Requirements for
the Degree of

Doctor of Philosophy
In
Computational Sciences and Engineering

Koç University
January, 2015

Koc University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a doctoral dissertation by

Engin Cukuroglu

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

Ozlem Keskin, Ph. D. (Advisor)

Attila Gursoy, Ph. D. (Advisor)

Türkan Haliloglu, Ph. D.

Halil Kavakli, Ph. D.

Mehmet Sayar, Ph. D.

Burak Erman, Ph. D.

Elif Ozkirimli Olmez, Ph. D.

Date: 19 January 2015

It is good to have an end to journey toward;
but it is the journey that matters, in the end.

Abstract

Improvements in experimental techniques increasingly provide structural data relating to protein-protein interactions and across-interface interacting domains. Classification of structural details of protein-protein and inter-chain domain-domain interactions can provide valuable insights for modeling and abstracting design principles of protein-protein interactions. This dissertation mainly focuses on clustering protein-protein interactions by their interface structures, exploiting these clusters to obtain and analyze shared and distinct protein binding sites, and dissecting the domain-domain combinations in protein interface structures. 22604 unique interface structures in the PDB are found and these unique interface structures are used to find protein pairs which have multiple binding sites to interact with each other. For the first time, domain combinations in the interfaces both on the same and on the complementary monomer are analyzed, based on PFAM domains classification. The domain combinations are represented by constructing a network. In addition to PFAM domain analysis, fold, family and superfamily classification of SCOP are mapped to unique protein interface structures in order to find the SCOP domains in each classification level which are the mostly frequent ones used in protein interfaces and have the most different unique interface structures. This comprehensive study of protein interface structures and interface domain combinations can serve as a resource for the community.

Özet

Deneysel tekniklerdeki ilerlemeler protein-protein etkileşimleri ve arayüzdeki yapısal bölgeleriyle ilgili yapı verilerini sağlamıştır. Proteinler ve zincirler arasındaki yapısal bölge etkileşimlerini sınıflandırmak, protein-protein etkileşimlerini modelleme ve dizayn prensiplerini özetlemede kıymetli anlayış sağlayabilir. Bu tez ana olarak protein-protein etkileşimlerini arayüz yapılarını kullanarak sınıflandırma, sınıflandırılan kümeleri kullanarak farklı protein bağlanma yerlerini tespit ve analiz etme, ve yapısal bölge birleşmelerini incelemek üzerine odaklanmıştır. PDB'de 22604 adet eşsiz arayüz yapısı bulunmuş ve bu eşsiz yapılar kullanılarak çoklu bağlanma bölgesi bulunan protein çiftleri bulunmuştur. Aynı monomer veya tamamlayıcı monomer üzerindeki yapısal bölge birleşmeleri PFAM yapısal bölge sınıflandırmasına göre ilk defa analiz edilmiştir. Yapısal bölge birleşmeleri bir ağ oluşturularak temsil edilmiştir. PFAM bölge sınıflandırma analizine ek olarak, protein arayüzlerinde en çok kullanılan ve en çok farklı eşsiz yapılara sahip olan SCOP yapısal bölge türlerini herbir sınıflandırma seviyesinde bulmak için, kıvrım, aile ve süperaile SCOP yapısal bölgeleri eşsiz protein arayüz yapılarıyla eşleştirilmiştir. Bu geniş kapsamlı protein arayüz yapıları ve arayüz yapısal bölge birleşmeleri calışması bu konuda çalışan topluluk için kaynak niteliğindedir.

Acknowledgements

In the first place, I offer my sincerest gratitude to my supervisors, **Ozlem Keskin** and **Attila Gursoy** for the chance of being a member of their research group, supplying us a productive and comfortable research environment and their support during my work.

I offer my deep appreciation to **Ruth Nussinov** for her guidance and sharing her endless experience with us.

I would like to thank also my thesis committee, Turkan Haliloglu, Halil Kavaklı, Mehmet Sayar, Burak Erman and Elif Ozkirimli Olmez, especially for their precious time.

I thank my officemates and all members of the COSBI family for their friendship.

Finally, I thank my family for their patience and continuous support during every step of my education.

Table of Contents

List of Figures

List of Tables

Nomenclature

PDB	Protein Data Bank
ASA	Accessible Surface Area
RMSD	Root Mean Square Deviation
SCOP	Structural Classification of Proteins
PFAM	Protein Families Database
PRISM	Protein Interactions by Structural Matching

Chapter 1
INTRODUCTION

Proteins are main constituents of many cellular processes. They catalyze reactions, provide transportation, form the building blocks of viral capsids, traverse the membranes to yield regulated channels, and transmit the information from DNA to RNA [1]. Proteins interact with each other to perform their functions. Investigating the protein-protein interactions systematically is a key step to enlighten molecular mechanisms. Molecular mechanisms have a sophisticated harmony. Some proteins interact with their partners simultaneously using different interaction sites, some of them interact with their partners using the same interaction site in different time, and some of them interact only one protein [2]. How it can be possible to bind many different proteins using the same binding interface or how a protein binds many different proteins at the same time are some key questions. Cukuroglu *et al*. [3] show that the structural orientation of residues are the protagonist of protein-protein interactions and different residue conformations make multiple and simultaneous interactions possible.

Protein-protein interactions take place in order to perform their functions, but there is a limited number of specific binding sites which proteins can bind with each other [4]. These specific binding sites are called interfaces. Interface is formed by atoms that interacts the atoms of complementary molecules. The orientations of residues are important for binding and differences between these amino acid positions provide novel binding strategies for proteins. Hence, the first question that comes to mind is “do proteins prefer specific amino acid positions to bind other proteins?” In order to answer this question, protein interfaces should be classified according to their structural similarities and then, their interface analysis will be showed. Also, another question pops up after the analysis of interfaces, even if they prefer specific positions how can it be possible for some proteins to bind many different proteins using the same binding

site conformations? If some interfaces are used multiple times, all similar interfaces can be extracted by using 3D structure information of the protein complexes. Separation of the dissimilar interface structures with a clustering method forms unique interface types in the cell. These unique interface types are used commonly in template based docking studies [5, 6] and Kundrotas *et al.* claimed that these unique interface architectures (templates) are available to model nearly all complexes of structurally characterized proteins [6].

This dissertation primarily focuses on structural analysis of protein interfaces and its applications. All the chapters endeavor to address the question "*how the proteins interact*".

The outline of this dissertation is as follows:

In Chapter 2, a comprehensive and most recent review of the protein-protein interactions and interface clustering methods is presented. The chapter includes the corresponding works related to architectures of protein interfaces, protein interface similarities, hot spot residues and clustering similar interface architectures.

In Chapter 3, the materials and methods are presented. First, it explains protein interface structure extraction and then expresses "how the interface similarities are found for all the PDB entries". The details of clustering method are defined step by step for the clarification. The chapter is finalized with the extraction of multi-interface proteins.

Chapter 4 includes the results of the new approach to cluster protein-protein interface structures. The validation of the cluster results, the comparison with previous method and yearly entries of PDB are presented. The chapter is pointed out that these interface cluster representatives can be used as a template set for protein–protein interaction prediction. For the template based prediction, the target proteins surface should be extracted properly to fit the template structures, so overview of the template interface accessible surface areas are introduced and possible optimization method for template based prediction is presented at the end of the section.

Chapter 5 is important for one of the applications of protein interface clusters. Multi-binding proteins can use one or multiple binding sites using multiple interface architectures in order to bind their partner proteins. The interface clusters are used to find all multi-binding proteins in PDB. In this chapter, the analysis of the interfaces of the multi-binding proteins is revealed. The question in this chapter is "how the specificity of the interactions established during binding."

Chapter 6 is dedicated to domains in protein interfaces. Most of the protein interfaces have single domain but there are interfaces that have multi-domain structures. All the domain combinations in these multi-domain interfaces are extracted and the important domains are labeled. Domain specific interface architectures will be useful for protein-protein interaction analysis.

In Chapter 7, the database of protein interface clusters and interface domain information are presented. PIFACE database includes all the protein-protein interface structures that are derived for interface clustering. The user can retrieve all the information about the protein-protein interfaces including the domain information. The JSmol plug-in is used for the visualization of both the protein interface structures and protein interface structure comparisons.

This dissertation ends with a conclusion chapter discussing the results, explaining future directions, and finally with presenting major conclusions of the study.

Chapter 2
LITERATURE REVIEW

Proteins physically interact with each other through binding sites. Some proteins interact with their partners simultaneously using different interaction sites, some interact with their partners via the same interaction site at different times, and some appear to interact with only one protein [2]. How can many different proteins use the same binding interface, and how can a single protein bind many different proteins at the same time, are key questions that emerge in structurally-enriched protein-protein interaction networks and regulation. Within the framework of the general factors that bear on these intriguing questions is the landscape of residue conformations, particularly of key residues, making multiple and simultaneous interactions possible [1, 7-11]. While a vast number of protein-protein interactions can take place, there is a limited number of specific binding site conformations through which proteins can bind [4, 12, 13]. Studies of interfaces can be illuminating: they can address questions such as whether preferences of specific amino acids in certain positions can help binding site prediction, and on a different level, how some proteins can bind many different proteins using the same binding site conformations. Since binding and folding are similar events, they may also help understand hierarchical protein folding [14]. Obtaining a set of unique interface structures can be particularly useful in template-based docking [5, 6]. It is previously shown that template based docking can be fast and accurate if there exists a good set of template interfaces [15-17]. Kundrotas et al. posited that unique interface structures can serve as templates to model nearly all complexes of structurally characterized proteins, and that the existing interfaces already can achieve this aim [6]. Kundrotas and Vakser showed that structural similarities of the interfaces have the greatest influence in template-based docking [18].

2.1 Proteins interact through their interfaces

Proteins interact with proteins, peptides, DNA or RNA in order to form complexes. In Figure 2-1, some examples of these interactions are presented. These complex structures are the constituent of the many biological processes [19]. Actually, proteins use interfaces in order to build a complex and fulfill their functions. Interfaces are formed by residues whose properties determine binding specificity and affinity. As an interface example, shown is the interface between a multi-subunit E3 protein ubiquitin ligase (Figure 2-2 PDB Id: 1GQP). Protein interfaces have been studied for a long time and researchers deposited their findings to the protein interface databases to identify the general properties of them. Some of the available databases are PROTORP [20], InterPare [21], 3did [22-24], PIBASE [25] and PRINT [7, 26].

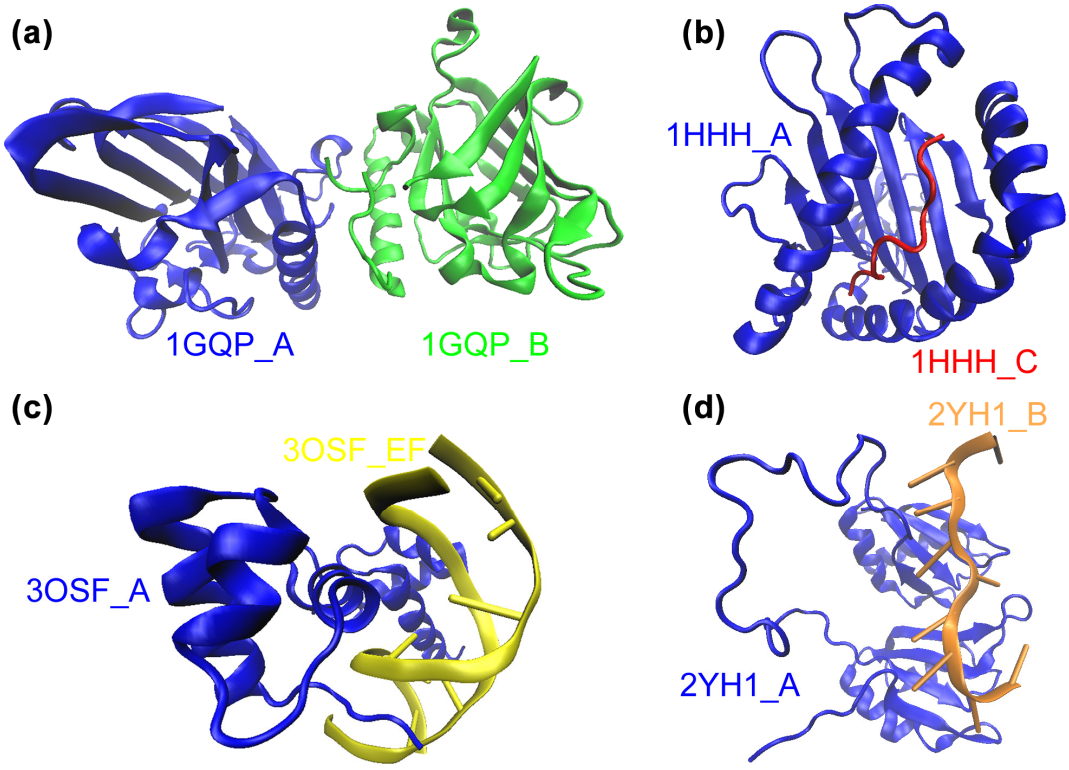


Figure 2-1 Protein interacts with different molecules. (a) Protein-protein interaction of 1GQP between chain A (blue) and chain B (green). (b) Protein-peptide interaction of 1HHH between chain A (blue, protein) and chain C (red, peptide). (c) Protein-DNA interaction of 3OSF between chain A (blue, protein) and chain EF (yellow, DNA). (d) Protein-RNA interaction of 2YH1 between chain A (blue, protein) and chain B (orange, RNA).

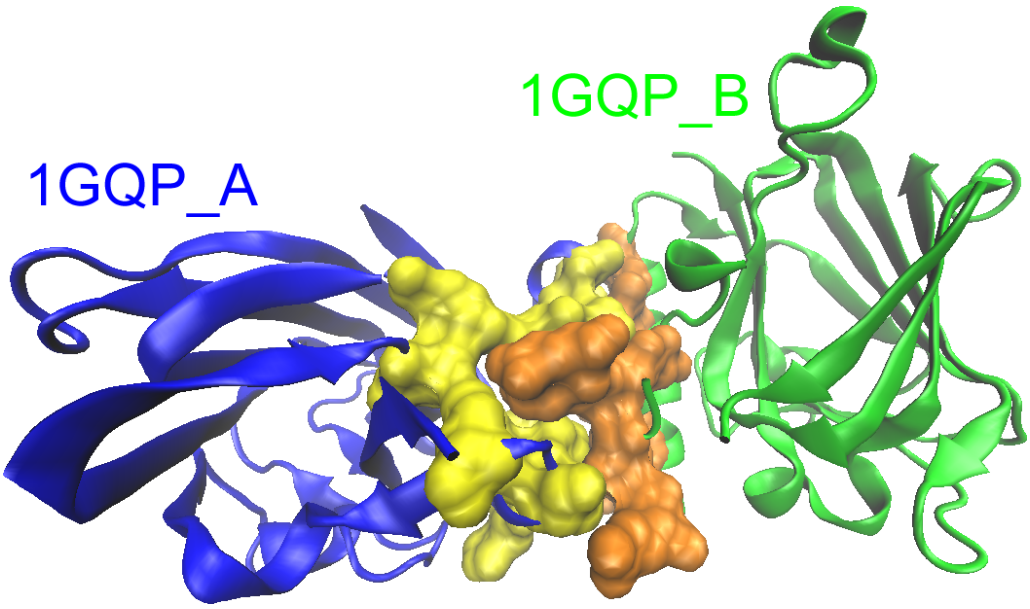


Figure 2-2 Protein interface of 1GQP between chain A (blue) and chain B (green). The interface residues of chain A are yellow and interface residues of chain B are orange.

2.2 Properties of protein interfaces

Proteins need an efficient mechanism to find their partners in a crowded cell and their amino acid properties can change their binding types. Protein interactions can occur between identical or non-identical chains. If a monomer interacts with an identical monomer it is called homo-oligomeric (homologous) complexes. If a monomer interacts with a non-identical monomer it is called hetero-oligomeric (heterologous) complex. These oligomeric structures can be classified for their protomer stability as obligate and non-obligate [27]. In an obligate PPI, the protomers are not found as stable structures on their own *in vivo*. Most of the hetero-oligomeric structures in the PDB are the example of the non-obligate interactions. They are mostly functioning in signaling pathways. Also, protein interactions can be classified as permanent or transient interactions based on the lifetime of the complex [27].

Permanent interactions are stable and only exist in complexed form whereas the transient interaction associates and dissociates *in vivo*. The interaction strength is mostly responsible for association and dissociation of the protein complexes and it depends on the salt bridges, hydrogen bonds, hydrophobic interactions and disulphide bonds [27-29]. The information about the transient complexes are limited due to crystallization limitations [30], hence, structural information of permanent residues are widely accessible than transient ones. All these interactions have been studied widely in order to understand the protein-protein interfaces [7, 28, 29, 31-40].

Protein interfaces are constituted by amino acids whose conformations are important for binding affinity and specificity. Previous protein-protein interface studies showed that size, shape and physicochemical properties are the key determinants of the protein-protein interactions [26, 28, 29, 32, 41]. Moreover, hydrophobic and electrostatic interactions are the major factors to stabilize protein-protein interfaces [28, 29, 32, 35, 42]. According to side chain information of the amino acids, they will be hydrophobic, hydrophilic, polar and nonpolar which changes the physicochemical properties of the protein interfaces. Also, side chain – side chain and backbone – backbone interactions of the amino acids are helpful to determine the secondary structures of protein interfaces (beta strand, alpha helixes and loops) [43, 44]. The secondary structures of the interfaces are important for the binding mode and studies showed that interface residues are more conserved than the surface residues [45-49]. Accessible surface area is another important identifier for protein interfaces. Protein interface sizes are changeable based on the change in their solvent accessible area when the association/dissociation process takes place. On average, interfaces has 1600(±400) $Å^2$ [29] and according to their interface sizes, their hydrophobic densities change.

In general, interfaces form planar or well packed structures according to type of interaction [13, 32]. Weak homodimers have flat, small and polar interfaces on average and their physicochemical and geometrical properties closely resemble those of non-obligate hetero-oligomeric complexes [50]. In contrast to the weak transient dimers,

strong transient dimers often confronted large conformational changes upon association/dissociation and have larger, less planar and sometimes more hydrophobic interfaces [50]. Interface residues of obligate complexes tend to evolve slower than the transient ones which is important for coevolving with their interacting partners [51].

A recent study divides the interfaces according to their interface structures [7]. The global structures of the monomers has similar interface structures and functions, it is called Type I. These types of interfaces are the expected ones because, in general, related proteins associate in similar ways. In Type II, the interfaces are similar but the overall interface structures and functions are different. In Type III, only one side of the interface is similar and the rest is different.

Proteins form biological complexes in vivo but sometimes there will be artificial interactions in PDB based on the crystallization artifacts. These complexes cannot occur as biological complexes and deciding which contacts are biological or not is a difficult process [52]. In order to differentiate these artificial contacts conservation score is used previously [45, 52]. Interface size is an important index for finding the non-biological contacts [31, 52, 53]. Henrick and Thornton used 400 Å2 as a threshold to identify biological complexes. Carugo and Argos compare the binding site and surface amino acid composition and if the composition is similar they label the interface as non-biological [54]. In order to find the biological interfaces, Zhu *et al.* use machine learning approaches with the interface features (interface area, interface area ratio, amino acid composition, correlation between surface and interface regions, gap volume index, and conservation score) [55]. Liu *et al.* use contact frequencies of the interface residues to predict non-biological interfaces [56]. Duarte et al. used evolutionary based approach based on the sequence entropy of the homolog sequences [57]. Recently, vibrational motion of the atom (B-factor) values are used to distinguish biological interfaces and crystal packing contacts [58].

2.3 Protein interface databases/servers

Protein interfaces have different properties as defined in previous section and these properties are generally deposited on the databases/servers.

ProtorP is a server that analyses the protein-protein interactions in 3D [20]. The server calculates the physical and chemical parameters of the interactions such as size, shape, intermolecular bonding, residue and atom composition, and secondary structure contributions.

InterPare is a protein domain interaction interface database [21]. It deposited protein interface data whose 3D structures are known. They supply geometric distance, ASA and Voronoi diagram of the protein interfaces. Also, they separated the data as inter-chain domain-domain and intra-chain domain-domain interaction interfaces.

PIBASE is a database of structurally defined protein interfaces [25]. The user can retrieve interface and domain related data with ASA, inter-atomic contact and sequence information. The database uses SCOP [59] and CATH [60] domain information.

PDBePISA is an interactive tool for the exploration of macromolecular interfaces [61]. The user can retrieve various data such as accessible/buried surface area, free energy of dissociation, and presence/absence of salt bridges and disulphide bonds.

SCOWLP is a database for characterization and visualization of the PDB protein interfaces [62]. The user can query SCOP/PDB Ids to get interfacial area, total interacting residues and number of wet spots.

2.4 Protein-protein interface similarity

The main step to achieve a unique interface set is to flag similar interfaces. Comparisons of two protein interfaces can detect similarities in amino acid sequences

of protein interfaces (sequence alignment) or similarities in 3D coordinates of amino acid positions in the proteins (non-sequential structural alignment). Protein interface clustering can be done in three different ways: using only sequential or only structural alignment scores of all protein interfaces, or a hybrid strategy which includes both sequential and structural alignment scores of protein interfaces. The PFAM [63] and SCOP [59] databases are commonly used for classification by sequence and structural alignments, respectively. Previous studies aiming to investigate binding properties showed that protein interfaces can be classified by their sequence similarities [4, 64] or, in other words, proteins with similar sequences often interact in similar ways [65, 66]. However, it has also been suggested that interactions can only be reliably inferred for close homologs [65, 67]. To decrease the computational cost, studies have also used a hybrid strategy of both sequence and structural comparison, and to increase the reliability of the classification, others used structural alignment of protein interfaces [68-72]. Bordner and Gorin clustered biologically relevant interfaces with a hybrid strategy to provide a reliable catalog of evolutionary conserved protein-protein interfaces with a diverse set of properties [73].

Detecting evolutionarily related proteins via structural similarities is more reliable than via sequence similarities since structure is more conserved. Sequence based methods are easy to derive and computationally cheap, so sequence based methods are generally the starting point of the studies; however, this has two limitations [74]. First, proteins generally use interfacial areas in order to form their interactions which do not necessarily contain short local sequence motifs. Secondly, if the sequence similarity of two proteins falls below 40% it is hard to make any inference about their functions. Schroder and his colleagues show that even if structural comparison is computationally costlier than other methods it gives more reliable results [68]. In order to find structural similarity between two protein interfaces, the 3D coordinates of the atoms must be known. The number of deposited structures increases exponentially with improvements in experimental techniques. Thus, it should be possible to identify novel binding strategies (if any) by examining recent deposited structures. Previous works

showed that the number of distinct interface structures grows with the increasing number of PDB structures [7, 26, 75].

2.5 Interface clustering methods

To classify protein interface structures by either sequence, or structural alignments, or both, some kind of clustering methods should be used. In most structural clustering studies, a hierarchical clustering algorithm is used. Ghoorah *et al.* compare structural alignment of interfaces by extracting dimensionless interface vectors using the center of mass of core and rim Cα coordinates [69]. Interface vectors are clustered using hierarchical clustering. Aloy and his colleagues classify domain based interactions using 3D structures of proteins and perform complete linkage hierarchical clustering to find global interfaces [22]. Tseng and Li generate Protein Surface Classification (PSC) library using pairwise local RMSD measures of protein surfaces [76]. They present 1974 surface types that include 25857 functional surfaces identified from 24170 bound structures. Also, Teyra *et al.* perform pairwise structural alignments of protein binding regions and classify them with agglomerative hierarchical clustering using the complete linkage method [77]. They also note that complete linkage is sensitive to zero similarity and expands the differences between the clusters.

There are methods other than hierarchical clustering, such as centroid models (k-means clustering), distribution models (multivariate normal distributions), density models, subspace models (biclustering), and graph based models. Graph theory has also started to become popular in the last decade for analyzing the relationship between events. Barabasi and his colleagues presented various properties of networks by tracking the internet routes [78]. They discovered the power law distribution of the networks and the importance of the most highly connected nodes in lethality and centrality [79]. Other researchers focus on topological properties of networks. Girvan and Newman highlighted the betweenness property (first proposed by Freeman [80]) of the nodes and edges of the network, and emphasized the community structure of the

network [81]. Edge (node) betweenness is defined as the number of shortest paths passing through this edge (node) between all pairs of other nodes. It is a measure of the influence of edge (node) on the flow of information between other edges (nodes). If a network has communities, few edges which have higher betweenness connect these communities. Removing the edges with highest betweenness value from the network means separating the communities from one another [81]. Girvan and Newman presented this method to sidestep the shortcomings of the hierarchical clustering.

2.6 Hot spot residues in protein interfaces

Residues in protein-protein interfaces do not equally contribute to the binding energies. There are critical residues called hot spots which contribute most to the binding energy [82, 83]. Hot spot residues can be detected by alanine scanning mutagenesis experiments [82]. If the binding energy difference is more than 2 kcal/mol after mutating a residue to an alanine, it is labeled as a hot spot. Bogan and Thorn [83] analyzed amino acid compositions of hot spots and concluded that some residues are more favorable as hot spots in protein-protein interfaces. Tyr, Arg and Trp are the most frequent ones which are critical due to their size and conformation. They also reported that hot spots are surrounded by a set of residues, that are energetically less important resembling to an O-ring in a pipe fitting, to occlude hot spots from water molecules. There is a correlation between change in the accessible surface area and energy contribution of residues [84]. Moreira et al. [85] also supported O-ring hypothesis using Molecular Dynamic (MD) simulations. Further, these hot spots are not randomly distributed in the interfaces but rather clustered. Hot spots are assembled within densely packed regions. These modular assembly regions are called hot regions [86]. This binding site organization justify how a given protein molecule may bind to different protein partners. Kleanthous and coworkers [87] showed that a limited number of mutations at the interface of cognate complex of colicin E9 endonuclease and immunity protein 9 provide high-affinity binding of E9 to immunity protein 2, although at a

weaker affinity compared to the cognate complex. These experimental findings were also studied by computational hot spot organizations [88]. The organization of hot spot residues provides a mechanism to obtain binding affinity and specificity to different partners. Therefore, cooperativity of these residues can reveal the complex binding organizations in specificity [89, 90].

Determining hot spot residues by experimental techniques is costly and time consuming, so computational methods have been developed to predict hot spot residues in bound and unbound protein structures [85, 91-100]. Prediction results show that prediction accuracies are improving over the years. Prediction methods will be a complement to experimental studies to find hot spot residues which have an important role on binding affinity and specificity. In addition, computational methods can guide experimental mutational studies and explain functional and mechanistic aspects of protein binding [101].

Chapter 3
MATERIALS AND METHODS

3.1 Definition of a protein-protein interface

An interface is described as contact region between two interacting proteins. Previous works show that there are different approaches to find contact residues in a complex. Distance based approaches use atomic distances between two proteins to extract interacting residues [7, 75, 97, 102-110], surface area based approaches use accessible surface area (ASA) values [20, 111-113]. Some of the interfaces are derived by using Voronoi diagrams [114].

Two residues are defined as contacting if the distance between any two atoms of the two residues from different chains is less than the sum of their corresponding van der Waals radii plus 0.5 Å [7, 75]. Nearby residues are defined as if the distance between alpha carbon atoms of noninteracting residues and interacting residues in the same chain is smaller than 6 Å. Previous studies showed that nearby residues are important to represent a more complete architecture of interfaces, such that interface residues are not isolated [7, 12, 13, 75, 115].

An interface is labeled with the PDB ID plus the chain IDs. For instance, if the PDB ID of a protein structure is 1GQP and there is an interface between chains A and B, then this interface is named 1GQPAB as in previous studies [7, 26, 75].

In order to generate a protein-protein interface dataset, all possible binary interactions of the protein structures are extracted and check regardless of whether they interact. All PDB entries are used to generate the interface set. As of January 14th 2012, there were 78477 PDB entries and 45491 of them were complex structures which resulted in 622321 possible binary interactions, assuming all protein chains interact with each other. However, not all chains in a protein complex interact physically with

each other, necessitating chain pair detection. Extracting interface residues in complexes by distance thresholds needs excessive time. Hence, the accessible surface area (ASA) of the possible interface pairs are first calculated by using NACCESS [116] in order to eliminate interface candidates whose interface ASA values of the complex structure are smaller than 1 $Å^2$. NACCESS calculates the monomer and complex ASA values of the proteins and the interaction surface is calculated simply subtracting complex ASA value from the monomer ASA values of two proteins. In order to find ASA values of the interface structures, NACCESS is run for monomer and complex form of the structures as described in naccessDriver method in Appendix B. Then, interfaceExtractorUsingNaccess method in Appendix B is called to extract interface structures using ASA values of the structures. When small interfaces were eliminated, 184342 interface candidates remained. (For NMR structures the first model is used. RNA and DNA chains are eliminated. Chains which have residues different than usual 20 amino acids are eliminated (e.g. selenomethionin)). The remaining interface candidates were processed according to interface definition. As a result, there were 130209 interfaces where each binding site had at least five residues. These interface structures are extracted using interfaceExtractorUsingVDW method in Appendix B.

3.2 Structural comparison

MultiProt [117] was used to compare these 130209 protein interfaces. MultiProt, which performs structural alignment regardless of the order of the residues on proteins, is an appropriate tool to use for comparison of interface structures which are generally composed of discontinuous segments of protein chains. No sequence alignment was performed. MultiProt uses PDB structures as input to calculate the binary similarities of each protein in the dataset. In each comparison step, two interface files which have contact and nearby residues in PDB format are given as an input to MultiProt in order to perform structural alignment.

3.3 Interface similarity

Interface similarity is calculated based on the structurally matched residues of the two interfaces. The similarity formula is:

$$InterfaceLength\_1 = numberOfInterfaceResiduesOfInterface\_1$$

$$InterfaceLength\_2 = numberOfInterfaceResiduesOfInterface\_2$$

numberOfStructurallyMatchedResidues are found by using MultiProt [117]. MultiProt aligns the interfaces structures using the geometric hashing algorithm. 3 Angstroms as the RMSD threshold in the structural comparisons are used.
The Interface similarity is defined as

$$InterfaceSimilarity = \frac{numberOfStructurallyMatchedResidues}{\min(InterfaceLength\_1, InterfaceLength\_2)} \quad \text{(Equation 3.1)}$$

Pairwise comparison of all interfaces is a time consuming process. Comparing a small interface with a big interface is unnecessary. Gao and Skolnick's [118] work on the structural space of protein interfaces shows that native interfaces find a match with a significant score among random interfaces and their mean interface residues coverage is 86% with a standard deviation 10%, and their mean contact residues coverage is 52% with a standard deviation 9%. Thus, to decrease the number of interface comparisons, interface comparisons are eliminated where the number of contacting residues in one interface is larger by 25% than that of the small interface and the number of contacting and nearby residues is larger by 50% as compared to the smaller interface. As a result, nearly 2 billion protein interface comparisons are done by MultiProt which used approximately 2500 cpu days to finish the comparisons. Interface similarities are found by using multiprotDriver method in Appendix B.

3.4 Clustering algorithm

Girvan and Newman [81] used graphs in order to find communities in a network. A community is defined by dense inter-connectivity (Figure 3-1a). Communities are clusters in a network which have similar properties. Communities are extracted using the betweenness property of the edges. An edge between communities has the highest betweenness value in the graph. The method of Girvan and Newman starts with removing the edge with the highest betweenness value and then, recalculates all the betweenness values of the edges and removes the edge from the graph that has the new highest betweenness value. Communities in a network are found applying this process until the stopping criteria are reached. Therefore, similar protein interfaces can be clustered using community finding algorithm. In order to find the clusters of similar interfaces, the network of interfaces should be formed first.

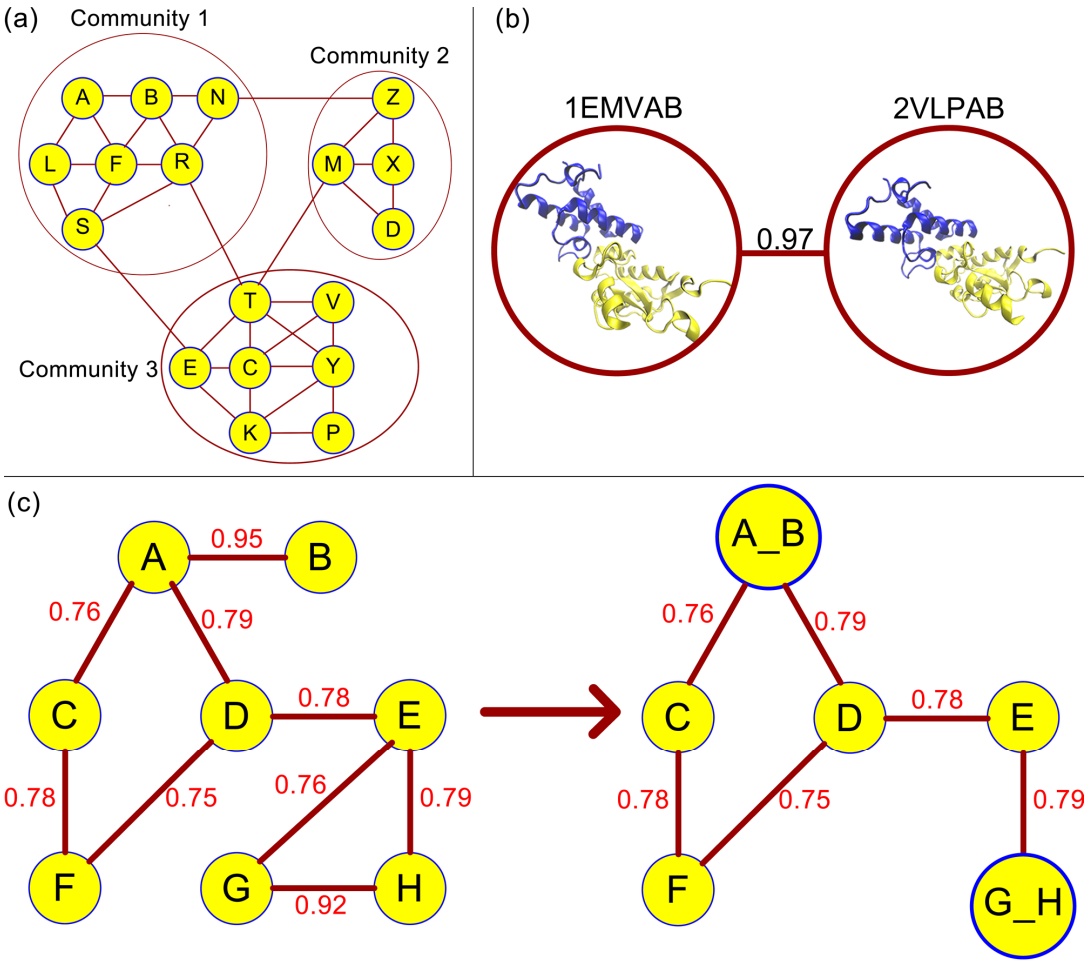


Figure 3-1 (a) Community structure. (b) Node and edge representation in protein interface network. (c) The nodes in the left network which have similarity values higher than 0.80 are grouped as a single node in the right network. The new node and the neighbors' similarity values are chosen as the maximum similarity value of the edges between the two nodes which were grouped and the neighbor nodes (e.g. Node G and node H).

3.4.1 Network of interfaces

A network is formed by nodes and edges. The network of protein interfaces is constructed by using interface structures as nodes. If two interfaces are similar (the similarity is calculated as explained above, Equation 3.1), an edge is drawn between the two corresponding nodes in the network (Figure 3-1b). This differs from the earlier

strategy of constructing the protein interface network [119] where protein monomers are considered as nodes. There, an edge is drawn if the two monomers form an interface. Here each interface structure is represented as a node and similarity between interface structures as edges. In order to generate network of interfaces, interfaceNetworkBuilder method in Appendix B is used. The known similarities between the interface structures are used as an edge in an undirected graph where the nodes are the interfaces.

3.4.2 Finding communities

Dividing a network structure to community structures starts with removing the edge which has the highest betweenness value. If the network is separated into two distinct networks, the edge removing process keeps going recursively for each network until one of the stopping criteria is reached. Stopping criteria are based on two network properties, clustering coefficient and minimum cluster size. The clustering coefficient is a measure of degree for nodes in a graph which tend to cluster together. The clustering coefficient [81] is calculated by

$$C=\frac{3*numberOfTrianglesOnTheGraph}{numberOfConnectedTriplesOfVertices} \qquad \text{(Equation 3.2)}$$

As a stopping criterion, different clustering coefficient values are used during clustering (1, 0.95, 0.90, 0.85 ..., and 0.5) in order to find the best possible clusters. If the network has a clustering coefficient value higher than the criterion the network separation process stops.

Minimum cluster size criterion, set at 5, is used because network nodes to fall apart as a network of size 1 are not wanted. After a network separation, if one of the clusters has less than 5 interfaces, this cluster is no more divided into two clusters.

In order to increase the speed of network separation, the nodes in the network are clustered in 5 steps according to the interface similarity values (increasing similarity values with 5%, similarity values higher than 0.80, 0.85, 0.90, 0.95 and 1.00 are processed respectively). At first, nodes which have similarity values higher than 0.80 are grouped as a single node (starting with the nodes which have the highest similarity) and their neighbors are linked to this new node (Figure 3-1c). A method called "combiningNetworkNodesWithSimilarity" in Appendix B is used to generate smaller network which is easier to calculate betweenness centrality values. There are two possible selections for the similarity value between the new node and the neighbors. First of them is selecting the maximum similarity value of the edges between the two nodes which were grouped and the neighboring nodes. Conversely, the second choice is selecting the minimum similarity value of the edges between the two nodes which were grouped and the neighbor nodes. When the final clustering results are analyzed selecting the maximum similarity value for the new node and the neighbor node is better than selecting the minimum similarity value (shown by validation of the clusters with different criteria in the Results section).

The new networks generated by the community finding algorithm are processed again, combining nodes which have similarity values higher than 0.85, 0.90, 0.95 and 1. After five cycles, all generated networks are reprocessed by the community finding algorithm without any node grouping to obtain the final results. Communities extracted at the final stage are the interface clusters. As explained in the method communityFinder in Appendix B, the network of edges are deleted one by one starting with the maximum betweenness centrality value until the network are divided to two connected components. Then, the average coefficient value and their component size are checked, if they passed the defined criteria the communities are finalized, otherwise, the component recursively processed again until it fits the criteria.

This grouping of the nodes according to their similarity values procedure is used to minimize the run time of the community finding algorithm. Calculating edge betweenness centrality of a network has $O(n^3)$ complexity. The largest connected component of the network has 1638090 edges which restrict applying the community finding algorithm because of the runtime. Edge betweenness centrality values of the new network should be calculated after each edge removing process, so this step is the bottleneck of the community finding algorithm. Therefore, the nodes are grouped according to their similarities, then the separation step is applied which speeds up the removal of the edges from the network, and simplifies finding the edge betweenness values of the network in the last step. The main steps of the methods are shown in Figure 3-2.

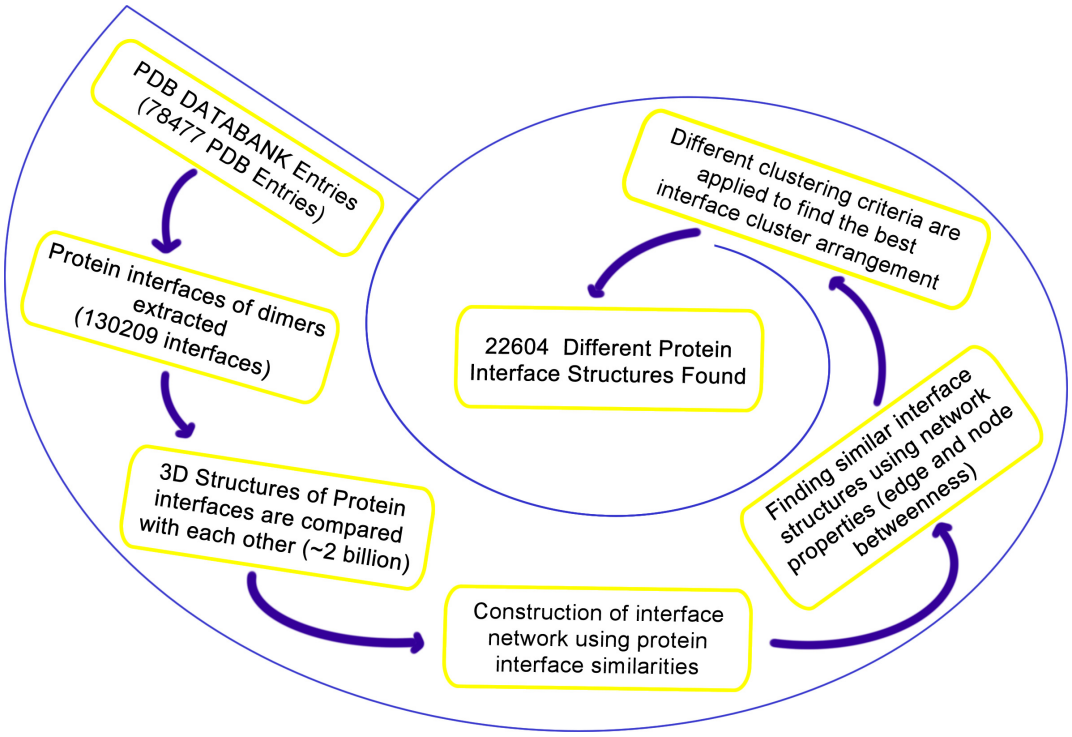


Figure 3-2 Flowchart of the methodology.

3.4.3 Evaluating communities

The clustering results are evaluated using the Silhouette index [120]. The performance of the Silhouette index has been reviewed in Handl et al. [121]. The silhouette index is calculated as:

$$S(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \qquad \text{(Equation 3.3)}$$

where

dissimilarity = 1 - similarity

a(i) = the average dissimilarity between interface i and all other interfaces in its cluster

b(i) = the minimum of the average dissimilarities between interface i and the interfaces in the other clusters.

SilhouetteIndexCalculator method in Appendix B is used to calculate all the silhouette values.

3.4.4 Interface clustering software

The clustering procedure was performed using an in-house software. Codes are written in Python, and the betweenness and clustering coefficient values are found by using the NetworkX package [122].

3.5 Multi-interface extraction of proteins

Some proteins have more than one interaction partner and they bind their partners using the same or different binding sites. The interface clusters are used to extract these proteins which have multi-interface structure.

To find proteins with multiple interfaces, firstly, pairwise sequence alignments of all the monomers in the PDB are downloaded from PDB FTP Services (22

November 2012) and labeled with a unique ID. All monomers which have the same unique ID have 100% sequence identity. This leads to 48669 different unique IDs. Then, two monomers which have an interface are labeled as first monomer unique ID underscore second monomer unique ID (e.g. 423_1002). According to their monomer sequence identities, 26825 distinct interface pairs are extracted. These pairs are compared with the structural clusters extracted from interface similarity. 7962 protein pairwise interactions out of 26825 have more than one interface in the PDB.

3.6 Type I, II and III interface extraction

Interfaces have different types according to their function and structural properties. The types of the proteins can be extracted combining protein interface cluster and SCOP Family [59] information.

Type I interfaces are similar and they should have similar global protein folds. Similar interfaces are grouped in the same cluster of the protein interface dataset which means that the members of the clusters have similar interface architecture with the representatives. Also, if the global folds of the proteins in the complex have same SCOP Family information with the representative complex, they can be labeled as Type I interface. If at least one of the proteins in the complex has more than one domain information and same global fold with the representative complex it is labeled as Type 1 multi-domain interfaces.

Type II interfaces are similar but they have dissimilar global protein folds. In order to find Type II interfaces, the members of the protein interface clusters are compared with their representative complex and if they don't share same SCOP Family information they have dissimilar global folds. Hence, the member with a dissimilar global fold with its representatives is labeled as Type II interfaces. If at least one of the proteins in the complex has more than one domain information with a dissimilar global fold, it is labeled as Type II multi-domain interfaces. The interface types are found by using type_I_II_finder method in Appendix B.

In this dissertation, Type III interfaces are defined using global folds of the protein structures of the complexes. If a protein structure with a same SCOP Family has different interface architecture across the interface clustering dataset, interface with this protein structure is called as Type III. In order to find these interfaces, SCOP Families of all the global folds are found and all SCOP Family combinations of the interface structures in the different clusters are compared. If there is a same SCOP Family combination in one of the proteins of the interfaces, they are labeled as Type III. Type_III_finder method in Appendix B is used to generate Type III interfaces from interface clustering dataset.

Chapter 4
RESULTS OF INTERFACE CLUSTERING

4.1 Validation of the clusters with different criteria

There are various clustering methods and they have different pros and cons. In this work, two step approaches are used for a reliable clustering result. First, grouping the nodes according to their similarities iteratively is used to provide compactness (small intra-cluster variation) of the clusters. Secondly, the edge betweenness property of the protein interface network is used to cluster similar interfaces, providing the connectedness of the clusters.

Choosing the clustering method is the starting point of the data analysis but the important part is finding appropriate criteria for generating the best results. Different clusters are obtained by using different parameters and all results from different clusters are evaluated to select the best cluster set. The parameters for generating different clusters are the clustering coefficient for the stopping criteria, the similarity value adjustments (increasing similarity values by 5% or 1% for grouping similar interfaces as a single node), and choosing the maximum or minimum similarity values for the new node and the neighboring nodes during node formation.

For the evaluation of a clustering method, there are two main measurements; external and internal [121]. For external evaluation, a gold standard protein interface clusters dataset from the PDB is needed, but unfortunately there are no standard clusters for protein interfaces. For internal measurement, Silhouette index [120] is used for comparing different clusters generated by different thresholds such as various clustering coefficient values for stopping criteria (1, 0.95, 0.90, …, 0.50) and different similarity values for grouping the nodes (1% or 5% increment of similarity value for grouping the similar interfaces). As a result, the silhouette index shows that using 1 as a clustering coefficient, 5% increment of similarity value for grouping similar interfaces, and

choosing the maximum similarity value between the combined nodes and neighbor node gives the best protein-protein interface clustering results. In Figure 4-1, the silhouette index for 100_5_max bar (clustering coefficient of 1, 5% increment of similarity value for grouping similar interfaces are chosen, and the maximum similarity value between the combined nodes and neighbor node is selected) gives the best silhouette values. This corresponds to 22604 clusters and 11088 of these clusters are single-membered. These 22604 clusters are used for further analysis which included both single and multi-membered clusters.

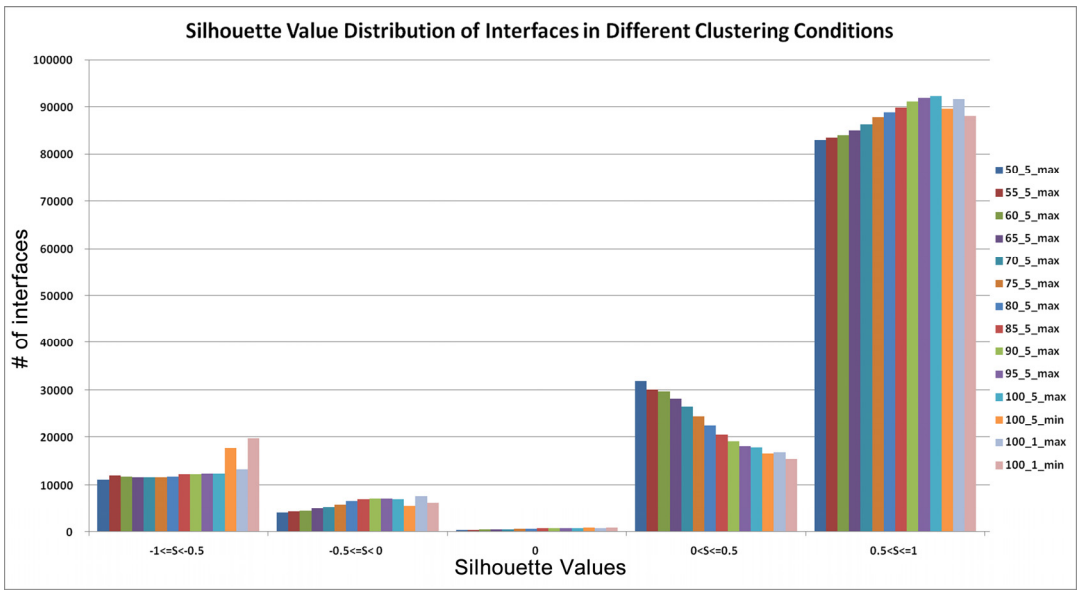


Figure 4-1 Silhouette value distribution of interfaces in different clustering conditions. For example 60_5_max means that clustering coefficient for stopping criteria is 0.6, similarity increment for grouping is 5 and maximum similarity value is used for edges from the combined node to their neighbor nodes.

Investigation of the interface clusters reveals that protein interactions can occur between homo- or heterodimeric chains as expected. Overall 87080 interfaces are homodimers and 43129 interfaces are heterodimers. 17523 of the interfaces out of 22604 representative interfaces are homodimers and the rest are heterodimers. There are 9123 homodimer interfaces out of 11088 are present in the cluster representatives without a member.

4.2 Type I, Type II and Type III interfaces

Further, when members of a single interface cluster are investigated, different types of interfaces are observed. Previously two interface types is labeled as: Type I: Similar interfaces, similar global protein folds. In most cases, if the interfaces are similar, the overall protein folds are also similar. Such similar interface, similar fold clusters contain a single family.Here, it is observed that protein interface structures can be similar when global folds of the complexes are similar, as expected An example is shown in Figure 4-2a. Bence-Jones Kappa I Protein Bre complex (1BRECF) and Immunoglobulin Light and Heavy chain complex (43C9AC) have 79% interface similarity as shown in the figure. Type II: *Similar interfaces, dissimilar global protein folds.* Even if the global folds of proteins are different, these proteins can interact by similar interface structures as shown in Figure 4-2b. Asportakinase complex (3AB2CG) and Thioredoxin complex (3O6TCD) which have different global folds have 77% interface similarity as shown in Figure 4-2.

Automatic type detection is done across all the protein interface clusters. If representative and member interface have SCOP Family information in order to find their global similarity interfaces are labeled as Type I, Type I multi-domain, Type II and Type II multi-domain interfaces. 34355 member interfaces are labeled as Type I, 3828 member interfaces are labeled as Type I multi-domain, 4692 member interfaces are labeled as Type II and 213 member interfaces are labeled as Type II multi-domain. Overall accessible surface and interface size values are shown in Table 4-1. Type I interfaces are generally have higher ASA and interface size values. If the global structure has multiple domains, it has higher ASA and interface size as expected.

There are 2121 proteins with different SCOP Family information which take a role to construct interfaces with different architectures. These proteins interact with other proteins in order to form 14351 Type III interfaces.

Table 4-1 Properties of Type I and Type II interfaces

	ASA		interface size	
	mean	std	mean	std
Type I	2186.616	1774.542	46.73893	36.83791
Type I multi-domain	3356.509	2610.34	71.4663	55.18918
Type II	1663.769	1453.219	36.06863	30.50729
Type II multi-domain	3209.135	2918.996	68.34742	62.8341

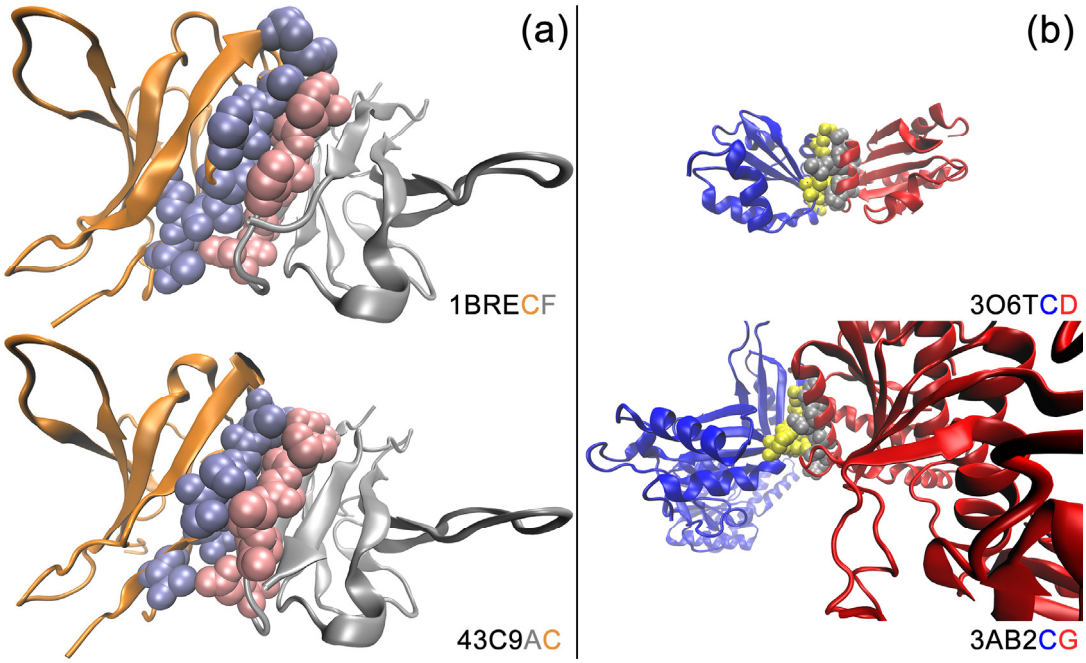


Figure 4-2 Complexes are shown in cartoon representation and interface residues are shown in ball representation. (a) Bence-Jones Kappa I Protein Bre complex (1BRECF) and Immunoglobulin Light and Heavy chain complex (43C9AC) have 79% interface similarity. (b) Asportakinase complex (3AB2CG) and Thioredoxin complex (3O6TCD) which have different global fold have 77% interface similarity.

4.3 Comparison with previous data

In order to show the performance of the current dataset, it is compared with previously derived non-redundant protein interface dataset [26]. Previous dataset is derived by using hierarchical clustering method and the important innovation of the

current dataset is replacing the hierarchical clustering algorithm with a community finding algorithm based on graph theory. Further, the previous clustering operated with the requirement that the maximum interface size difference between cluster members should be maximum fifty residues; the current clustering method uses a percentage-based approach. There will be similarity between the interfaces if the bigger interface size does not exceed the 1.25 times of the smaller interface size. This strategy also outperformed with smaller interfaces. Hence, the new method extracted different interface structures than the previous published method. 45176 interface structures are clustered in previous dataset and 7240 unique interface structures were found. The new clustering algorithm is applied to these 45176 interface structures and 8648 unique interface structures are obtained. When their silhouette values are examined, the previous clusters overall silhouette value is 0.52 and the new clusters overall silhouette value is 0.70. The distribution of silhouette values over the number of interfaces explicitly showed that the community finding algorithm based on graph theory outperformed the hierarchical clustering (Figure 4-3).

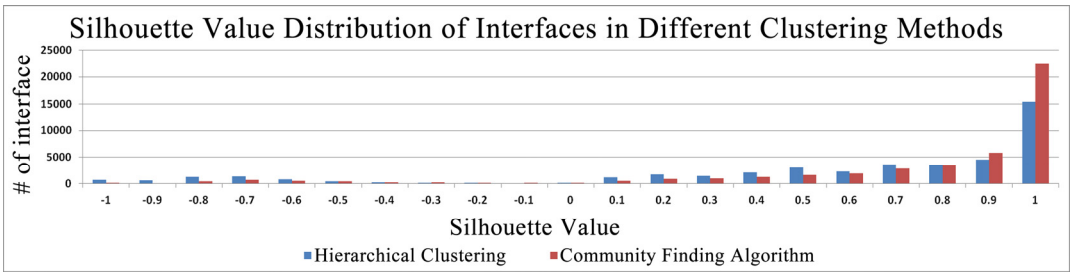


Figure 4-3 The silhouette value distribution over the protein interfaces generated by hierarchical clustering and community finding algorithm is presented. The new clusters are better clustered according to the silhouette values.

New dataset is compared with the work of Kim W.K. *et al.* [68]. The benchmark data used in their work is generated by using domain information of the complexes. In this case, interactions between protein monomers are investigated. Therefore, in these protein-protein interactions there are multi-domain interactions at the same time. For

example, in the benchmark set, the interface between the chain A and B of PDB ID "1A22" exists in two different clusters because chain B has two SCOP domains. The SCOP domain in Chain A interacts with two different SCOP domains in Chain B. However, in this dataset, these three domain interactions are labeled as one interface structure. Multiple domain entries always cause problems in comparison. Another problem is the difference between the minimum number of residue interactions in order to form an interface structure. An interface should have at least 5 interacting residues in each monomer, however, in the benchmark, some domain-domain interactions have less than 5 interacting residues in the monomer. Therefore, these results could not be compared with the benchmark dataset of Kim et al.

4.4 Surface extraction of monomers for template based docking

Structurally non-redundant interface architectures are obtained by extracting representatives of all clusters. These representative interfaces are a valuable resource for template based docking studies. The main purpose of template based docking is matching monomer surface to a template interface structure; thus surface extraction of the monomers is an important step. In order to obtain reliable docking results, the same method should be used to extract the surface of monomers with the templates. Jones and Thornton [32] used relative accessible area (RASA) of the residues to define the interior and exterior residues. They defined exterior residues as having RASA value >5%, and interior residues as those with RASA value <=5%. However, when the distribution of the RASA values over the interface and its nearby residues is analyzed, the mean of the average RASA values of each representative interface residues was 52.58 with a standard deviation of 7.84. The mean of the average RASA values of each representative nearby residues was 29.72 with a standard deviation of 9.12. Hence, using 40% RASA value which corresponded to 99% of the average interface RASA values is suggested in order to extract interface residues using RASA values of the

residues. Both interface and nearby residue distributions of representative interfaces are shown in Figure 4-4.

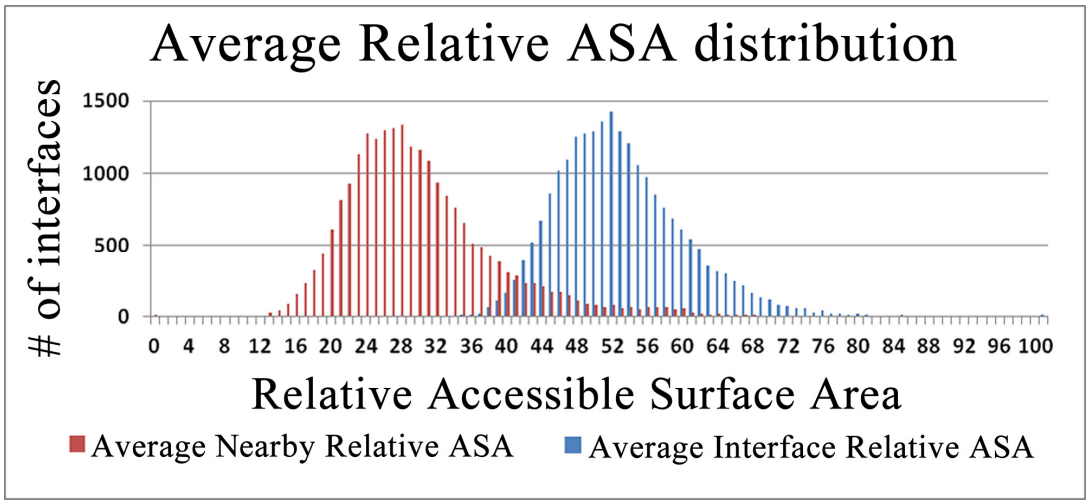
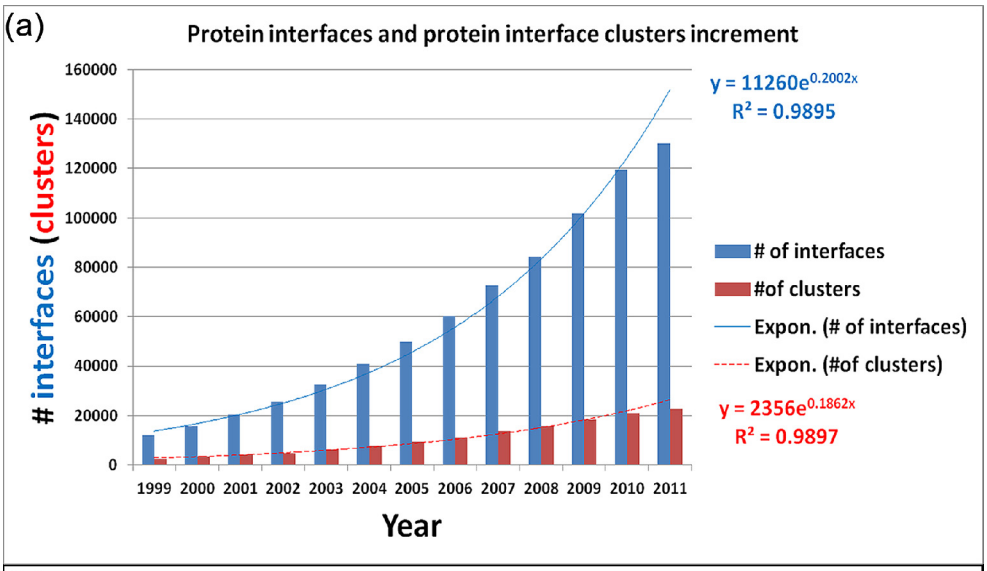
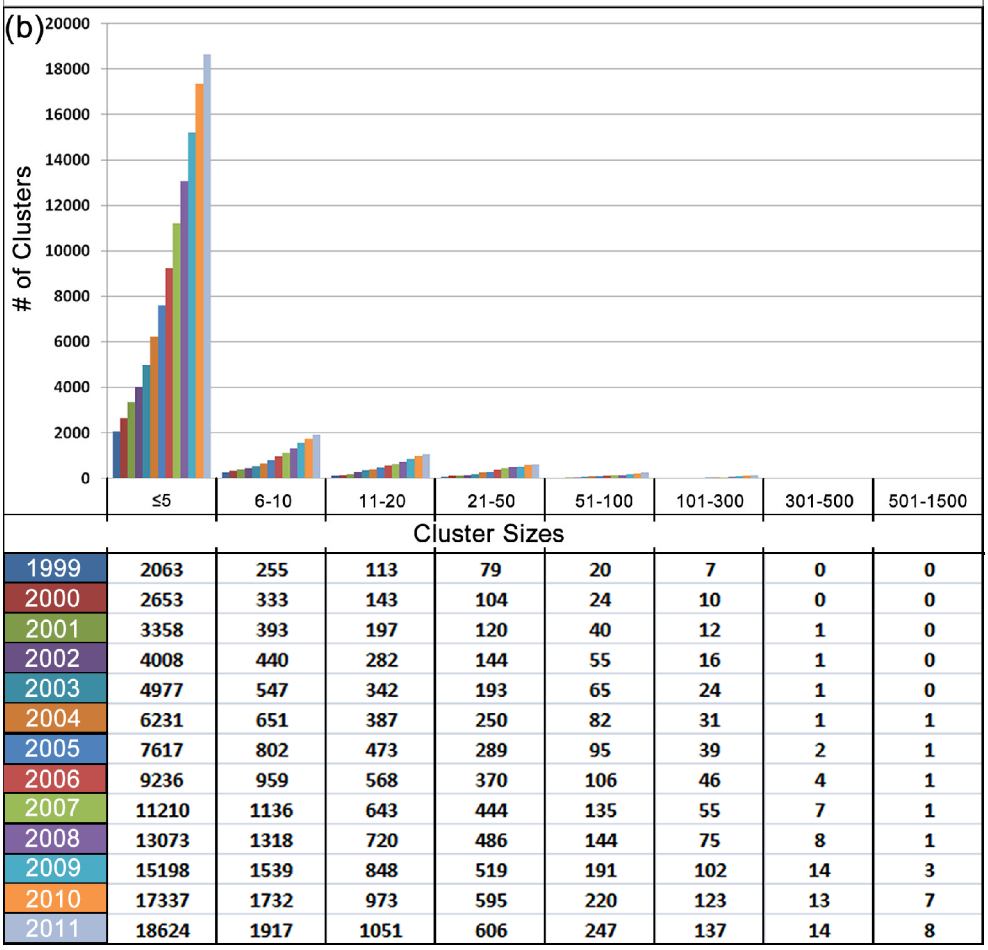


Figure 4-4 The average relative accessible surface area of the representative interfaces and their nearby residues are showed. It is suggested using 40% RASA value which corresponded to 99% of the average interface RASA values in order to extract interface residues using RASA values of the residues.

4.5 Protein interfaces and interface clusters based on years

The numbers of both available protein interfaces and their interface clusters increased exponentially in recent years. However, interface clusters have smaller increment than protein interfaces as shown in Figure 4-5a. In addition to the increase in the number of clusters during the years, there is also an increase in the clusters size. The distribution of protein interface cluster sizes are presented in Figure 4-5b. The largest cluster in 1999 had 238 members that increased to 1361 in 2011. Moreover, in 2011, there were 8 clusters with more than 500 members which were not present before 2004.





	≤5	6-10	11-20	21-50	51-100	101-300	301-500	501-1500
1999	2063	255	113	79	20	7	0	0
2000	2653	333	143	104	24	10	0	0
2001	3358	393	197	120	40	12	1	0
2002	4008	440	282	144	55	16	1	0
2003	4977	547	342	193	65	24	1	0
2004	6231	651	387	250	82	31	1	1
2005	7617	802	473	289	95	39	2	1
2006	9236	959	568	370	106	46	4	1
2007	11210	1136	643	444	135	55	7	1
2008	13073	1318	720	486	144	75	8	1
2009	15198	1539	848	519	191	102	14	3
2010	17337	1732	973	595	220	123	13	7
2011	18624	1917	1051	606	247	137	14	8

Figure 4-5 (a) Protein interface and interface clusters evaluation during years. (b) Distribution of protein interface cluster sizes throughout years.

4.6 Interface clustering with different similarity values

In this study, it is expected to find if there is a method which uses smaller number of representatives as templates than actual 22604 protein interface representatives to predict protein-protein interactions. In PDB, there are 130209 interface structures which will be also a template for protein-protein interaction prediction but it is computationally costly and comparing with the similar interface structures repeatedly is an unnecessary process. Structurally similar interfaces are clustered in order to eliminate possible extra comparisons with the same interface structure while predicting two monomers are interacting or not. According to clustering results, 22604 distinct protein interface structures are present in PDB. The representatives are grouped hierarchically as shown in Table 4-2 in order to optimize the number of comparisons during prediction of protein-protein interaction. Table 4-2 represents the interface clusters with different similarity levels. 22604 interface representatives are obtained using at least 75% interface similarity criterion and using these representatives, 19950 interface clusters are obtained using at least 70% similarity criterion (17674 interface clusters with 65% similarity, 14192 interface clusters with 60% similarity, 11624 interface clusters with 55% similarity and 8589 interface clusters with 50% similarity). A bottom up interface representatives finding process is applied and in order to validate this optimization a top down protein-protein prediction process will be applied.

Table 4-2 Hierarchical clustering of representatives using different similarity values

Similarity	# of interfaces	# of clusters	Silhouette values
s75	130209	22604	0.56
s75_s70	22604	19950	0.52
s75_s70_s65	19950	17674	0.47
s75_s70_s65_s60	17674	14192	0.40
s75_s70_s65_s60_s55	14192	11624	0.33
s75_s70_s65_s60_s55_s50	11624	8589	0.26

4.7 Concluding remarks and future directions

A non-redundant dataset of protein-protein interfaces are generated, containing 22604 unique interface structures. The new dataset has not been used for prediction as yet; however, older datasets have been extensively used for predictions of protein-protein interactions [15, 16, 123, 124]. Protein-protein interactions reflect functional and structural information. This interface dataset can be analyzed with respect to all of these. The dataset is functionally unique in two ways: first, it provides a rich resource of structural data of protein-protein interactions, allowing using these for knowledge-based protein-protein interaction predictions, for constructing structural pathways, for studies of drug side effects, where drugs may bind to 'unintended' similar interfaces, for alternative pathways in drug resistance, and broadly for protein function. Second, the analysis is performed with respect to interface residues. Extraction of the surface of monomers can be easily done by using RASA values of interface residues. Surface residues identified by the RASA values will work well with the current template set in template-based docking.

As a future direction, protein interface clusters can be optimized in order to predict protein-protein interactions using knowledge-based method. First of all, two monomers which have 3D structures available in PDB are taken as targets. A template based protein-protein interaction prediction tool should be used for the protein-protein interaction prediction and the optimized templates should be used as prior knowledge. PRISM [5, 125-127] is one the template based protein-protein interaction prediction tool which can be modified for using the optimized interface clusters. The search of similar interface architectures are started with relax similarity criterion (50%) using 8589 representatives shown in previous section. If there is possible interaction for these targets, the used templates are selected for the second level comparison with similarity value 55%. The similar interface architectures to the selected interface architectures in the second level (similarity value equals to 55%) are used as templates in order to find possible interactions for these targets. The templates which have similar architectures

with previous selected templates in the first level are used rather than comparing with 3035 interface structures (11624-8589, first level minus second level). This procedure will improve to avoid unnecessary calculations during prediction. Whole process will finish after six iterations which is the sixth level with 22604 interface representatives of the hierarchical grouping as shown in Table 4-2.

Chapter 5
RESULTS OF MULTI-BINDING PROTEINS

5.1 Multi-interface interaction

A protein can interact with its partners using the same or different interfaces. Finding possible interaction sites is challenging and it is critical to predict possible interaction modes. Docking, homology modeling, and template based docking can be used for this purpose. When all PDB entries are investigated, some proteins with multiple partners are observed to exploit multiple interaction sites with their partners. These experimentally found structures can be extracted using distinct interface structures of the complexes (representative interface structures) present in PDB. A general view of the database is shown in Figure 5-1a.

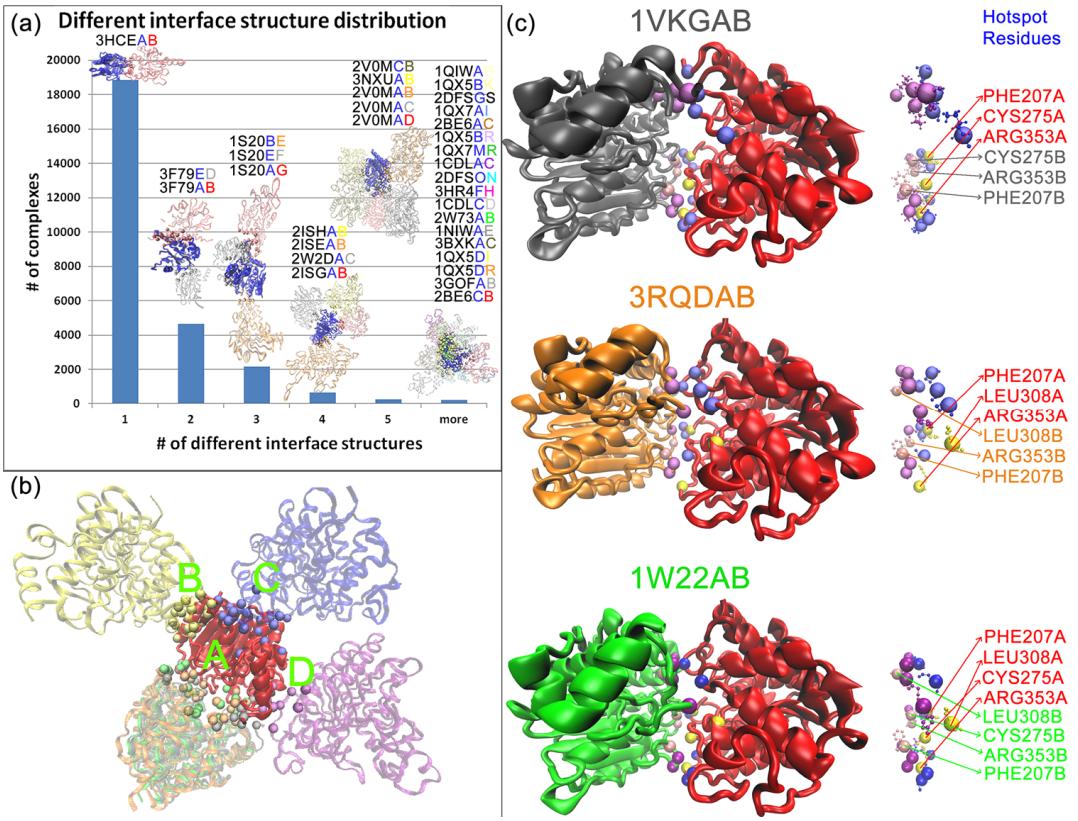


Figure 5-1 (a) Histogram of protein-protein interactions which have different binding structures at the same shared site or different binding sites. (b) Multiple interaction sites of Histone deacetylase 8 (red-1VKGA). (c) Histone deacetylase 8 (red monomer) uses the same binding site to bind different partners. On the left hand-side, protein complexes are shown and in the center, interface structures are shown in ball and stick. Blue and yellow balls are the interface residues of histone deacetylase 8 (red), and yellow balls also showed the hotspot residues of the interface. Pink and purple balls are the interface residues of the partner monomer shown in gray, orange and green, and pink balls also show the hotspot residues of the interface. In the right side, hotspot residues of the interfaces are given.

Proteins prefer different conformations to bind other proteins. Analysis of those 7962 monomer pairs, which are extracted from PDB shown in Chapter 3 in section "Multi-interface extraction of proteins", illustrates that these proteins interact with their partners using slightly different conformations to bind their partners at the same shared site or at different binding sites. In Figure 5-1a, complex with one interface structure is pair of Phenylethanolamine N-methyltransferase with Phenylethanolamine N-

methyltransferase, with two interface structures are pairs of probable two-component response regulator with probable two-component response regulator, with three interface structures are pairs of Hypothetical oxidoreductase yiaK with Hypothetical oxidoreductase yiaK, with four interface structures are pairs of Neurotoxin BoNT/A with Neurotoxin BoNT/A, with five different interface structures are pairs of Cytochrome P450 3A4 with Cytochrome P450 3A4 and eighteen interface structures are pairs of Calmodulin 2 with Calmodulin 2. 3500 out of the 7962 use the same shared site to bind their partners and the rest employ different binding sites. For example in Figure 5-1b, the red labeled monomer, which is a histone deacetylase 8, is shown with multiple interaction partners at four different binding sites (a shared site and three different binding sites). Histone deacetylase 8 (gray, orange, green, yellow, blue, and purple) can interact with other histone deacetylase 8. It can pair up with another histone deacetylase 8 at four different binding sites, dependent on cellular conditions, phosphorylation states and mutations. Interestingly, these histone deacetylase 8 pairs (taken from the protein interface clusters with their properties listed in Figure 5-1) exist in six different interface architectures.

Table 5-1 Six different interfaces for interaction of histone deacetylase 8 with another histone deacetylase 8 show that both at the same and different locations, the interface size varies across binding sites.

Interface Name	Interface ASA (Å^2)	# of interface residue	Binding Site
1VKGAB	1152.59	25	A
3RQDAB	907.84	20	A
1W22AB	1089.86	24	A
3F0RAC	753.88	19	B
3F07CB	1076.17	21	C
1T64AB	423.46	10	D

Analysis of Figure 5-1b illustrates that the binding site of histone deacetylase 8 is used multiple times. Six different interface architectures of interaction between

Histone deacetylase 8 and Histone deacetylase 8 are shown. One of the binding sites is shared by three partners. The others are different. Gray-1VKGB, orange-3RQDB, green-1W22B are at binding site A, yellow-3F0RC is at binding site B, blue-3F07B is at binding site C, purple-1T64B is at binding site D. The balls represent the carbon alpha atoms of the interface residues of the complexes. Carbon alpha atoms are labeled according to interaction partners of the 1VKGA. However, as can be seen in Figure 5-1c these have different interface architectures (these interfaces are in different interface clusters because their interface similarities are lower than 75% similarity. The RMSD scores of the pairwise structural alignments of the complexes (over 672 aligned residues) generated by PDBeFOLD [128] are as follows: 1W22AB-1VKGAB:1.126 Å, 1W22AB-3RQDAB:2.305 Å and 1VKGAB-3RQDAB:1.756 Å (RMSD values are calculated by using alpha carbons)). Thus, despite the similarity of monomers and binding sites, small conformational changes in the binding sites provide different interface architectures (like interologs [129]) and different relative energy distributions among the residues (Figure 5-1c) [13]. The energy distributions of the residues are extracted using the HotRegion server [88]. HotRegion gives information about hotspot residues which contribute more to binding energy [82, 83, 130]. Hence, generating templates using sequence similarity is not a good choice for docking. On the other hand, templates prepared with interface structure similarities are sensitive leading to more reliable docking.

5.2 Multi-interface binding and hot spot residues

Some proteins can interact with multiple partners using the same or different interfaces on their surfaces. It is impossible for these proteins to interact with many partners using different interfaces, because they have limited surface area for binding. Hence, they use some binding sites repeatedly [13, 131]. Thangudu *et al.* [132] showed that this multiple binding sites have at least one hot spot residue and these sites usually overlap. The affinity and specificity of multiple interactions can originate from

cooperativity of different hot spot residues [88]. An experimental study also highlighted that the mutations of different hot spot residues of the hub protein SMAD3 changed the affinity of the interactions with its partners [133]. Hot spot residue analysis can reveal the complex binding strategies of the proteins with multi-interface binding. Cukuroglu *et al.* [134] extracted all possible proteins with multi-interface binding strategies in PDB [135]. These structures were all from experimental non-redundant structures. According to the results of an analysis, actin used 40 different binding modes while interacting with another actin monomer as shown in Figure 5-2 (40 different complexes are superimposed). The blue monomer on the center of the Figure 5-2 is the actin monomer in common and the other actin monomers represented with different colors are the interaction partners. When the hot spot residues of the complexes are analyzed (hot spots and hot regions are found by Hotregion [88]), only two interfaces did not have any computational hot spots. 35 interfaces used 67 different hot spot residues on the actin surface (colored blue in Figure 5-2) and none of the hot spot residue combinations were repeated in any interaction which might generate the binding affinity and specificity. In three complexes, the actin monomer (colored blue in Figure 5-2) did not have any hot spot residues even if their partner monomers had hot spot residues (Supplementary Table 1).

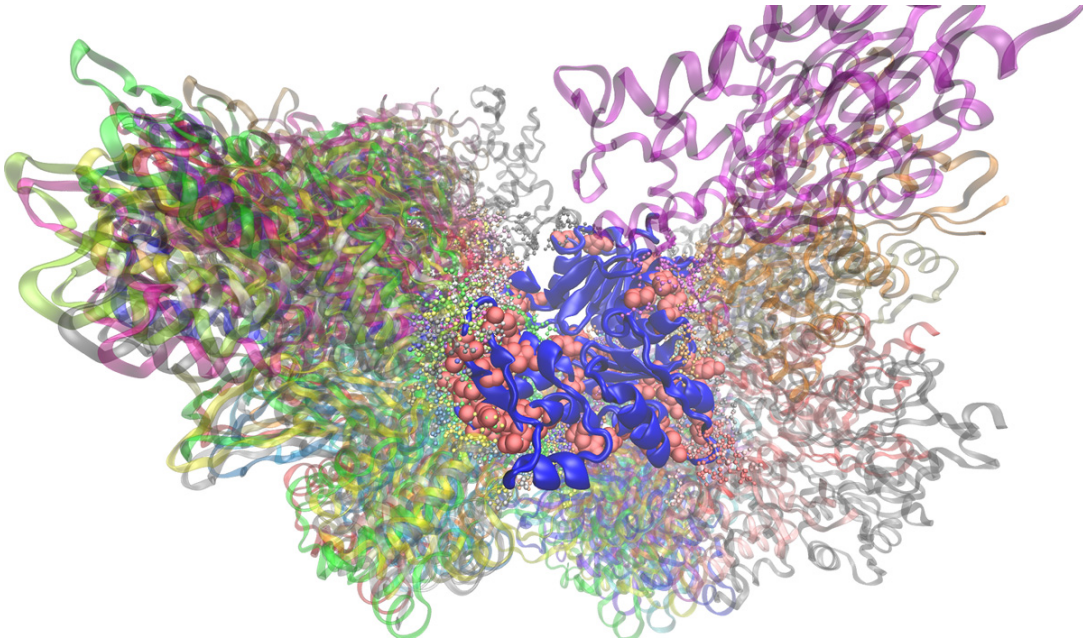


Figure 5-2 Actin monomer is shown in blue cartoon representation. The other colors are the interaction partners of the Actin monomer (Actin – actin interactions). Ball and stick representations show the interface residues and big red ball representations show the residues which are used as a hot spot at least in one of the interaction by actin monomer as shown in blue cartoon representation.

Ubiquitin used 22 different binding strategies while interacting with another ubiquitin monomer as shown in Figure 5-3 (22 different complexes are superimposed). The computational hot spots were tabulated in Supplementary Table 2 and the complexes with computational hot spots are shown separately in Figure 5-4. The ubiquitin monomer (colored blue in Figure 5-3) used 17 different residues as hot spots and these hot spot residues were used in 13 complexes. In four complexes, the ubiquitin monomer (colored blue in Figure 5-3) did not have any hot spot residues even if their partner monomers had hot spot residues. Only five complexes did not have any computational hot spots ant the other 17 complexes with hot spots residues are shown in Figure 5-4. The coloring of the Figure 5-4 is the same as Figure 5-3. The blue colored ubiquitin monomer used different hot spot residue combinations with their partners in order to provide binding affinity and specificity. For these two examples, the hot spot

residues in the interfaces are formed mostly a hot region. For actin complexes, hot spot residues in 27 complexes out of 32 complexes that have hot spot residues both on the target and the partner proteins make contacts with each other, also in the ubiquitin example this case occurred in 10 complexes out of 11 complexes. These complementary hot spot interactions might be a promising target for drug studies. However, even if hot spots do not have any complementary hot spot residues in partner protein they will also be a promising target for inhibiting protein-protein interactions.

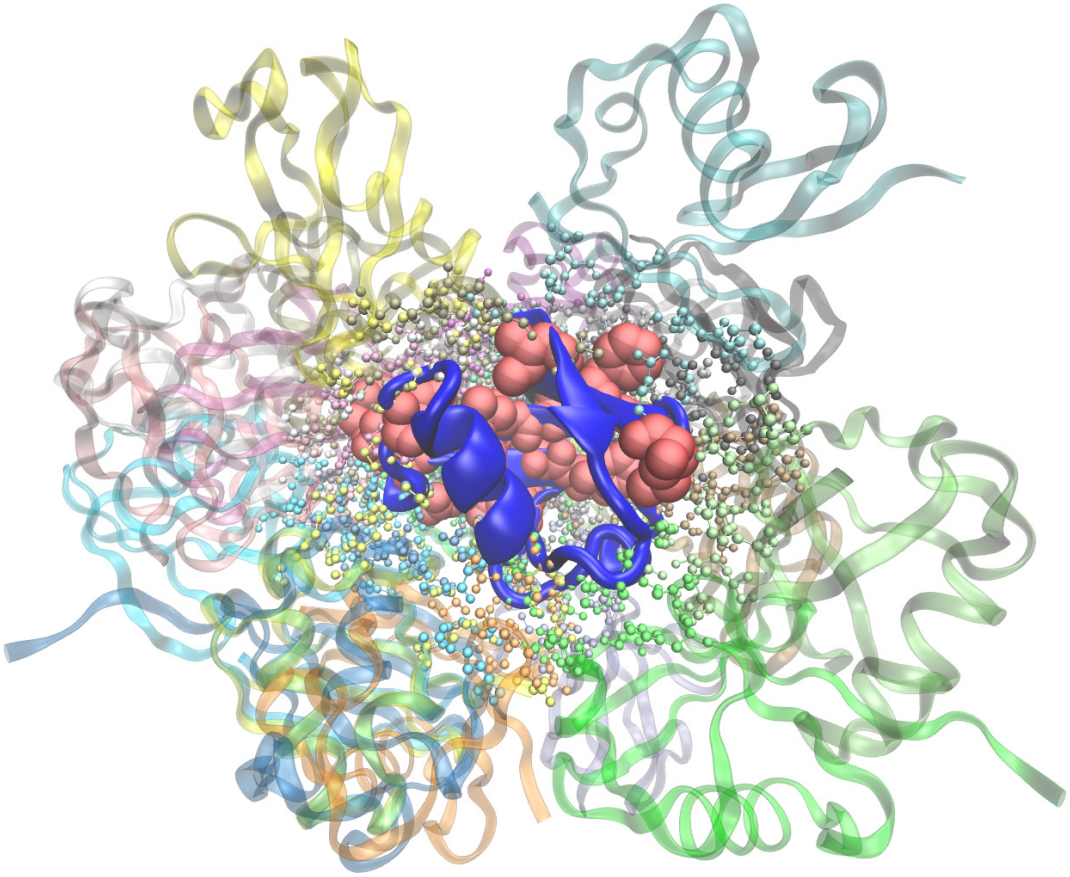


Figure 5-3 Ubiquitin monomer is shown in blue cartoon representation. The other colors are the interaction partners of the Ubiquitin monomer (Ubiquitin – Ubiquitin interactions). Ball and stick representations show the interface residues and big red ball representations show the residues which are used as a hot spot at least in one of the interaction by ubiquitin monomer as shown in blue cartoon representation.

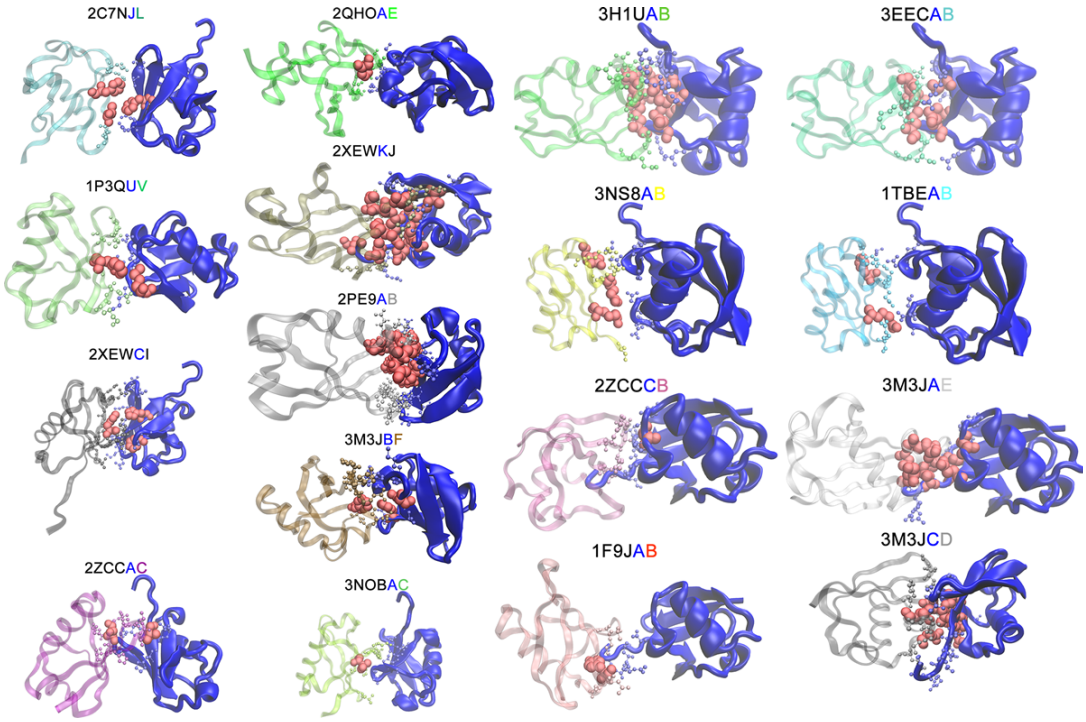


Figure 5-4 Ubiquitin- Ubiquitin interactions with hot spot residues are represented seperately. The hot spot residues are shown in Supplementary Table 2. The colors of the cartoon representations are the same as Figure 5-3.

5.3 Concluding remarks

Analysis of these multi-binding protein interfaces illustrates that 7962 of them are shared multi-partnered binding sites. Since the interfaces are derived from the PDB, it suggests that over one third of the protein-protein complexes have proteins bound to different partners. Some may be subunits; others may be signaling hub proteins. This allows dissecting protein-protein interactions to address questions such as how similar (or, different) are interfaces of partners binding to the same shared regions, on a large scale. It is observed that interface size may vary substantially among partners, as do the hot spot residues in the interaction. Such shared binding site cases may help in addressing questions of binding specificity.

Chapter 6
RESULTS OF DOMAIN INTERFACE RELATIONSHIP

Proteins are formed by domains and domain based analysis can help in preparing domain specific templates for protein-protein interaction predictions. In this part, the protein domains are analyzed according to their contribution to the interface structures. PFAM [63] and SCOP [59] domains are analyzed.

6.1 PFAM domains

PFAM [63] is a collection of functional regions of proteins (usually corresponding to domains). The organization of functional units on protein surfaces can play important roles in binding partner proteins. The analysis of PFAM domains in one side of the interfaces shown in Figure 6-1 indicates that most interfaces are formed by monomers in the PDB with single domains. Also, the most representative interfaces are formed by monomers with single domains as shown in Figure 6-2

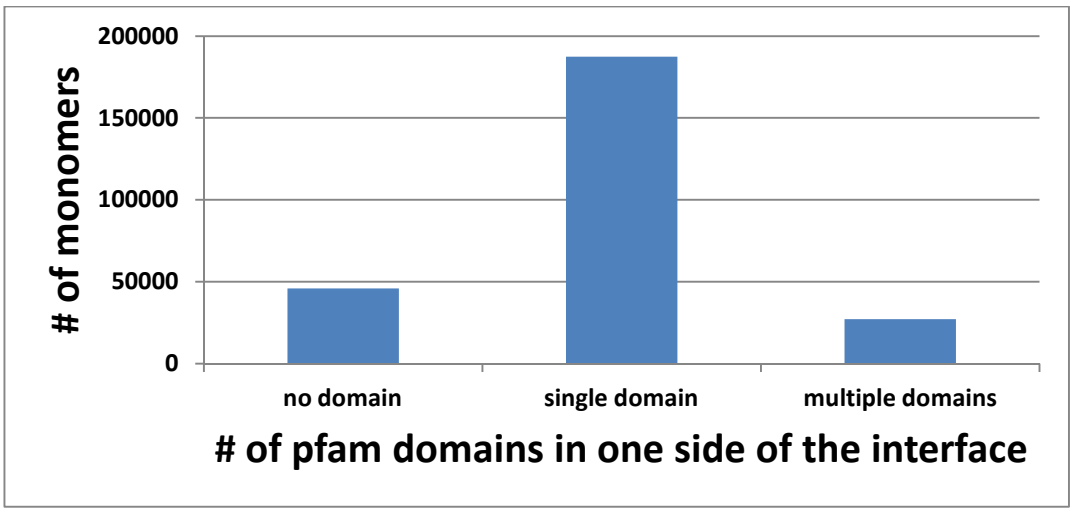


Figure 6-1 The distribution of the PFAM domains in monomers of the interfaces.

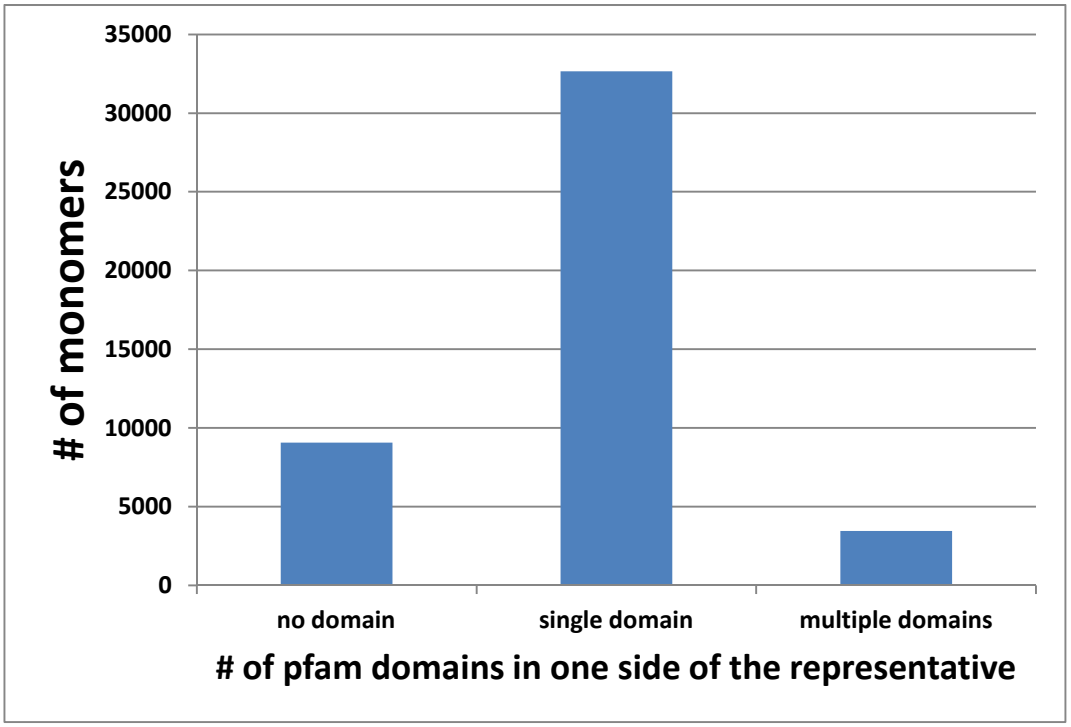


Figure 6-2 The distribution of the PFAM domains in monomers of the representative interfaces.

If all the domains in the interface of a monomer are enumerated and compared with the number of domains in their binding partner, the network representation of the binding preferences will be as shown in Table 6-1. Binding preference is defined as the percentage of occurrences of an interface built by a monomer with x number of PFAM domains and another monomer with y number of PFAM domains in their interface architecture. According to the table, if a monomer has six PFAM domains in its interface, it binds 100% with a monomer which has five PFAM domains in its interface. The overall preferences are gathered in the diagonal as shown in Table 6-1, so the interfaces are mostly constructed by the binding sites which has similar number of domains. The overall binary interaction preferences are tabulated in Table 6-2 as the number of interactions (e.g. Monomers with one domain are mostly prefer to interact

with monomers having one domain, 83777 interface structures are found from PDB) and in Table 6-1 as the percentage of the preferences.

Table 6-1 Fractions of # of domains in monomer vs # of domains in partner monomer

		# of domains in partner monomer						
# of domains in monomer		0	1	2	3	4	5	6
	0	0.68	0.29	0.03	0.00	0.00	0.00	0.00
	1	0.07	0.89	0.03	0.00	0.00	0.00	0.00
	2	0.05	0.26	0.67	0.01	0.00	0.00	0.00
	3	0.02	0.12	0.15	0.72	0.00	0.00	0.00
	4	0.02	0.04	0.38	0.05	0.28	0.23	0.00
	5	0.00	0.02	0.00	0.00	0.21	0.29	0.48
	6	0.00	0.00	0.00	0.00	0.00	1.00	0.00

Table 6-2 # of domains in monomer vs # of domains in partner monomer

		# of domains in partner monomer						
# of domains in monomer		0	1	2	3	4	5	6
	0	15612	13339	1282	37	3	0	0
	1	13339	83777	6297	270	5	3	0
	2	1282	6297	8198	343	51	0	0
	3	37	270	343	844	6	0	0
	4	3	5	51	6	19	31	0
	5	0	3	0	0	31	22	70
	6	0	0	0	0	0	70	0

The preferences of the representative interfaces are tabulated in Table 6-3. Monomers with six domains are not present in the interface representatives, so the information about six domains does not available. When the results of all the interfaces and the representatives are compared, they almost similar which mean that interfaces are mostly constructed by the binding sites which have similar number of domains.

Table 6-3 Fractions of # of domains in monomer vs # of domains in partner monomer and # of domains in monomer vs # of domains in partner monomer in paranthesis.

		# of domains in partner monomer					
# of domains in monomer		0	1	2	3	4	5
	0	0.79 (3243)	0.20 (797)	0.01 (48)	0.00 (0)	0.00 (0)	0.00 (0)
	1	0.10 (1612)	0.88 (14731)	0.02 (344)	0.00 (8)	0.00 (1)	0.00 (0)
	2	0.08 (133)	0.24 (405)	0.67 (1113)	0.01 (18)	0.00 (3)	0.00 (0)
	3	0.04 (6)	0.18 (24)	0.21 (28)	0.57 (77)	0.01 (1)	0.00 (0)
	4	0.20 (2)	0.10 (1)	0.20 (2)	0.10 (1)	0.20 (2)	0.20 (2)
	5	0.00 (0)	0.00 (0)	0.00 (0)	0.00 (0)	0.50 (1)	0.50 (1)

Previous studies extended the domain analysis by examining combination of different domains [136-138]. They suggested that the combination of domains increases the specificity of enzymes, generates new regulatory functions and builds new molecular recognition sites. Thus, all the domain combinations in interfaces are explored. The combinations of PFAM domains in interfaces, including whether the domains are on the same monomer side of the interface or on the complementary side, can be shown using a network representation (Figure 6-3a). The interactions of PFAM domains generate a huge network. The combinations of PFAM domains on the same side of the interfaces are showed in Figure 6-3b and across interface sides are shown in Figure 6-3c. Both the same side and opposite side combination networks have 0.001 graph density which means that most of the domains combine with a limited number of domains. The average degrees of the networks are 1.57 for same side and 1.90 for the opposite side combination network which explains the general choices for interaction partners. The interfaces of proteins are constructed by domains which mostly prefer the same domains for combination both in the same and across monomers. However, some domains like ATPases, GHKL domain, alpha amylase etc. (Table 6-4) have higher preference rates than average for an interface. These domains assist other domains to build an interface for specific functions. Figure 6-3c shows the preferred opposite side

combination network of interacting partners in different RNA polymerase domains (Table 6-5).

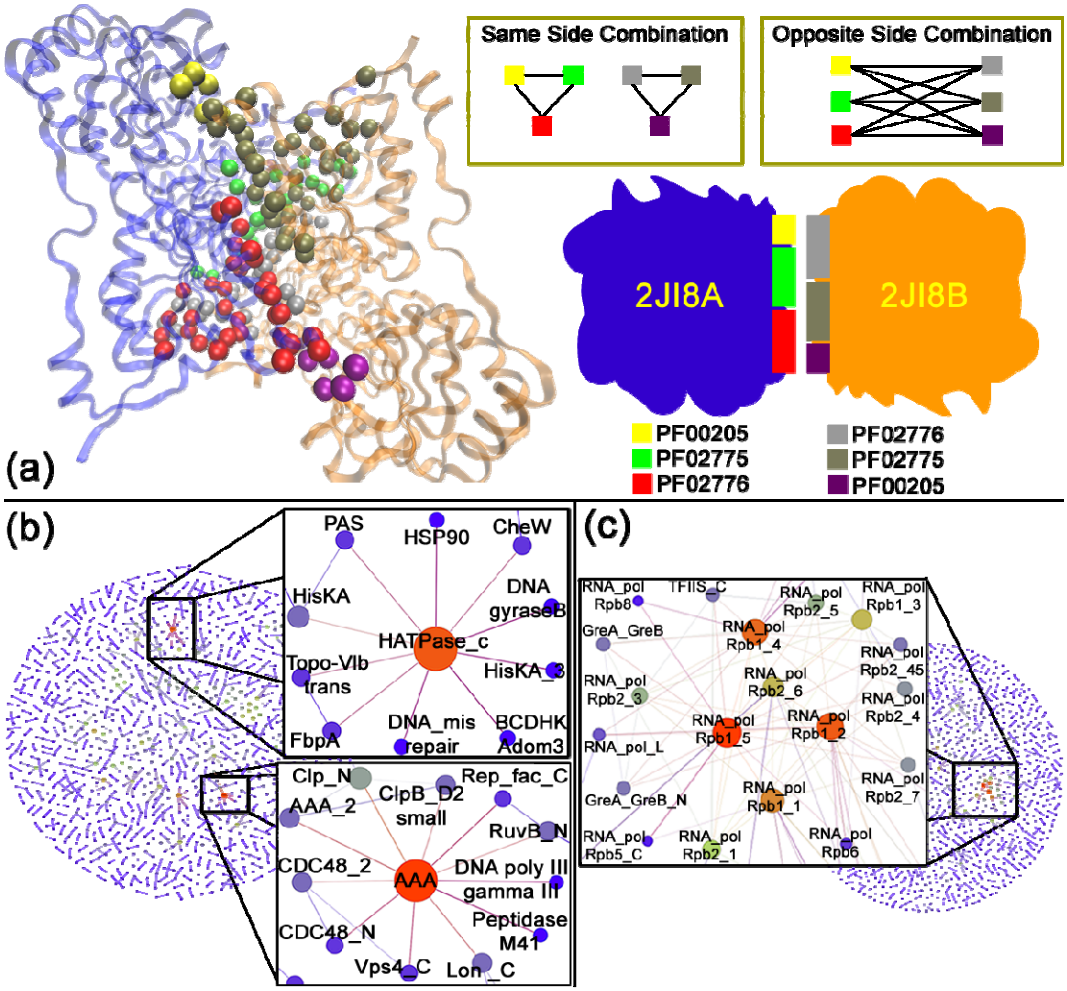


Figure 6-3 (a) A combination network example using the interaction between oxaly-CoA decarboxylase and oxaly-CoA decarboxylase (2JI8AB interface). The interface has 6 domains, 3 domains in chain A and 3 domains in chain B. Same side and opposite side combination networks are constructed as shown in upper right part of the figure. (b) Same Side PFAM Graph. On the upper right hand-side the GHKL domain, on the lower right hand site the ATPase family associated with various cellular activities domain and their combination domains in the same side of the interface structure are presented. The GHKL domain works with PAS, HSP90, CheW, DNA GyraseB, DNA_mis repair etc. domains to diversify its functions. (c) Opposite Side PFAM Graph. RNA polymerase Rpb1 domain 5 and its combination domains in the opposite side of the interface structure are presented. In the complementary monomer interface structures, RNA polymerase Rpb1 domain 5 makes combinations with mostly RNA polymerase domains to fulfill its functions.

Table 6-4 Top 18 PFAM domains that have the most domain partners in the same monomer. Their functional informations are listed at the last column.. PF00004 has 11 different domain combinations in a monomer. It chooses its domain partner according to their functional activities.

PFAM ID	# of combination domains	# of monomers	Functional Info
PF00004	11	390	ATPase family associated with various cellular activities
PF02518	10	132	The **GHKL domain** (**G**yrase, **H**sp90, Histidine **K**inase, **M**utL)
PF00128	8	133	Alpha amylase, catalytic domain
PF00560	7	50	Leucine Rich Repeat
PF00111	7	220	2Fe-2S iron-sulfur cluster binding domain
PF03810	6	30	Importin-beta N-terminal domain
PF00069	6	131	Protein kinase domain
PF01833	6	74	IPT/TIG domain
PF04563	6	833	RNA polymerase beta subunit
PF04560	6	562	RNA polymerase Rpb2, domain 7
PF04565	6	502	RNA polymerase Rpb2, domain 3
PF00072	6	17	Response regulator receiver domain
PF00207	6	56	Alpha-2-macroglobulin family
PF01751	6	61	Toprim domain
PF00440	6	142	Bacterial regulatory proteins, tetR family
PF00562	6	746	RNA polymerase Rpb2, domain 6
PF00293	6	16	NUDIX domain
PF01842	6	118	ACT domain

Table 6-5 Top PFAM domains that prefer the most number of different domains in the partner monomer in order to fulfill their functions. Their functional information are listed at the last column.

PFAM ID	# of domain partners	# of interfaces	Functional Info
PF04998	19	837	RNA polymerase Rpb1, domain 5
PF00623	17	834	RNA polymerase Rpb1, domain 2
PF05000	16	775	RNA polymerase Rpb1, domain 4
PF04997	14	595	RNA polymerase Rpb1, domain 1
PF00004	12	351	ATPase family associated with various cellular activities
PF05193	11	291	Peptidase M16 inactive domain
PF04983	11	527	RNA polymerase Rpb1, domain 3
PF00562	11	698	RNA polymerase Rpb2, domain 6
PF00089	10	59	Trypsin
PF00560	10	57	Leucine Rich Repeat
PF00271	10	50	Helicase conserved C-terminal domain
PF00128	9	135	Alpha amylase, catalytic domain
PF00069	9	268	Protein kinase domain
PF04563	9	471	RNA polymerase beta subunit
PF00071	9	45	Ras family
PF00129	9	771	Class I Histocompatibility antigen, domains alpha 1 and 2
PF00006	9	1566	ATP synthase alpha/beta family, nucleotide-binding domain
PF00009	9	45	Elongation factor Tu GTP binding domain

PFAM domains are mapped to interface clusters in order to find the distribution of interfaces in PFAM domains and how many different interface structures they use to interact with their partners (Figure 6-4a). Some domain interactions are heavily studied. As shown in Table 6-6, examples include the proteasome subunit, ferritin-like domain and trypsin. Others such as the all-important protein kinase domain may also use different interface structures to bind their partners (Table 6-7). Protein kinase domains are utilized in phosphorylation. Hence, protein kinases domains appear in many different protein interface structures. In addition to the Bourne and Scheeff study [139] who grouped protein kinases according to their structural evolution, structural

differences in their protein interface structures is found. 297 different interface structures of protein kinase domains are present in PDB.

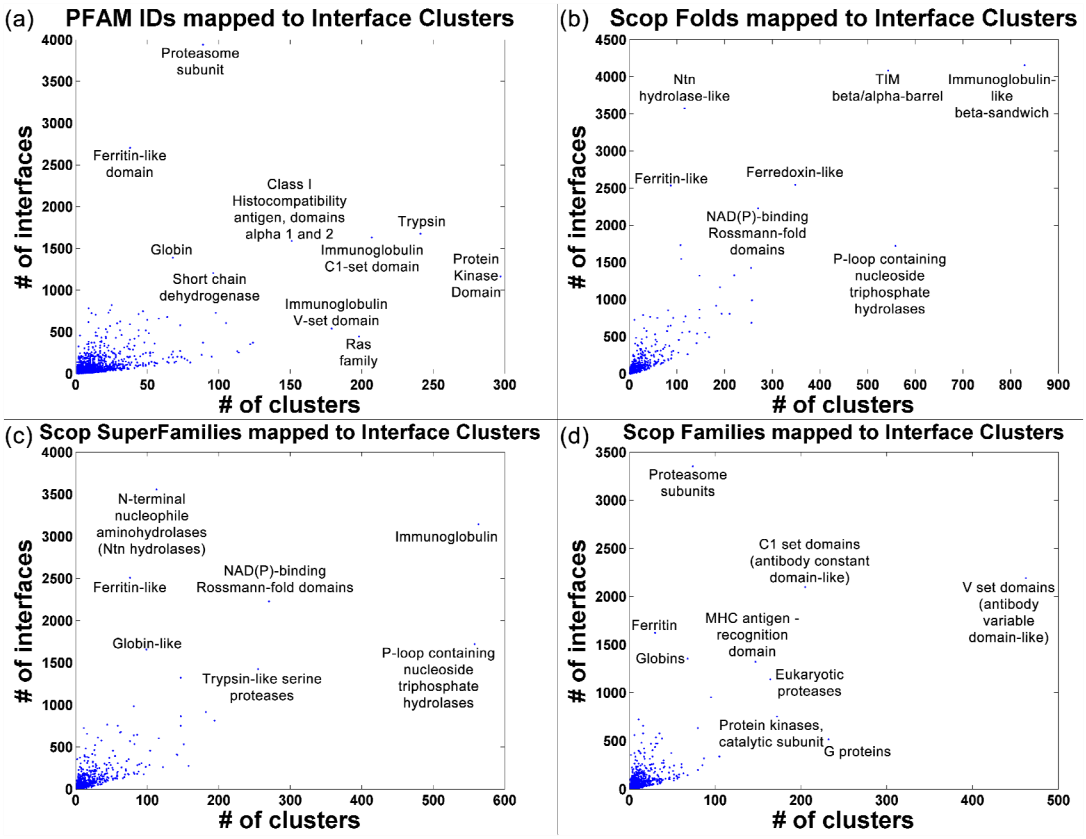


Figure 6-4 (a) PFAM domains are mapped to protein interface clusters. The figure shows the number of interfaces vs number of clusters for each PFAM domain. Protein kinase domain uses the highest number of different interface architectures to fulfill its function even if it does not have the highest number of interfaces in PDB. (b) Scop Folds mapped to interface clusters. (c) Scop SuperFamilies mapped to interface clusters. (d) Scop Families mapped to interface clusters

Table 6-6 Top 10 PFAM domains which appear in the interfaces

PFAM ID	# of interfaces	Functional Info
PF00227	3941	Proteasome subunit
PF00210	2705	Ferritin-like domain
PF00089	1676	Trypsin
PF07654	1632	Immunoglobulin C1-set domain
PF00129	1592	Class I Histocompatibility antigen, domains alpha 1 and 2
PF00042	1390	Globin
PF00106	1206	Short chain dehydrogenase
PF00069	1166	Protein kinase domain
PF00016	822	Ribulose bisphosphate carboxylase large chain, catalytic domain
PF10584	785	Proteasome subunit A N-terminal signature

Table 6-7 Top 10 PFAM domains which use different interface structures to bind their partners

PFAM ID	# of clusters	Functional Info
PF00069	297	Protein kinase domain
PF00089	241	Trypsin
PF07654	207	Immunoglobulin C1-set domain
PF00071	198	Ras family
PF07686	179	Immunoglobulin V-set domain
PF00129	151	Class I Histocompatibility antigen, domains alpha 1 and 2
PF00036	124	EF hand
PF00595	122	PDZ domain (Also known as DHR or GLGF)
PF00067	114	Cytochrome P450
PF07714	113	Protein tyrosine kinase

There are 1166 interfaces that have kinase domains in their structure and these interfaces have 297 different interface architecture. Hence, Kinase domains use different interaction strategies in order to fulfill their functions. The four examples of these Kinase domain interactions using different structural architecture are shown in

Figure 6-5. In Figure 6-5a, two monomers with Kinase domains (PF00069) are formed an interface structure. In Figure 6-5b, a monomer with Kinase domain and a monomer with a transforming growth factor beta type I GS-motif (PF08515), which regulates cell growth and differentiation, are used different interface architectures then previous example. In Figure 6-5c, each monomer has two different domain structures (Kinase and Endoribonuclease domains (PF06479)) in their interfaces and these domains collaborations form the interface architecture. In Figure 6-5d, the monomer highlighted with grey has the kinase domain and the complementary monomer has two different domains (Cyclin N terminal (PF00134) and Cyclin C terminal (PF02984) domains). These four examples of kinase domain interactions with different partners are important for the functionality perspective. The monomers with kinase domain use different interface architectures in order to fulfill its functions.

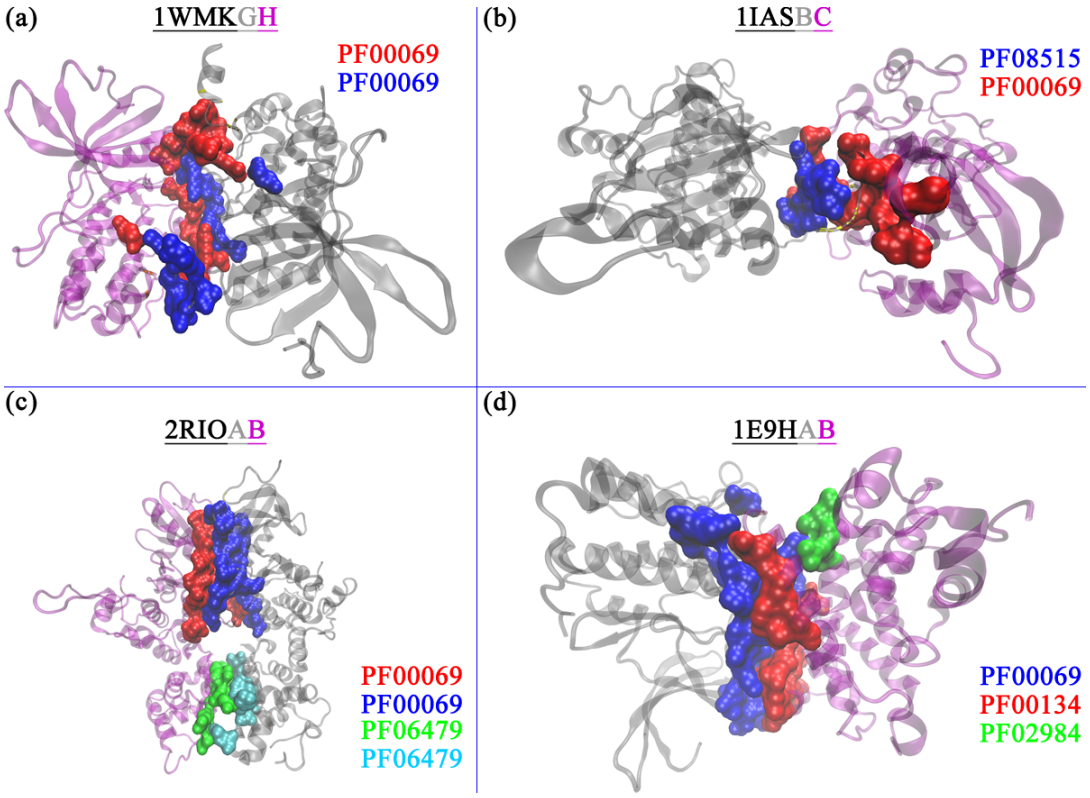


Figure 6-5 Examples of interfaces with Kinase domains (PF00069). (a) Two monomers with Kinase domains (PF00069) are formed an interface structure. (b) The interaction between a monomer (purple) with Kinase domain (red) and a monomer (grey) with a transforming growth factor beta type I GS-motif (PF08515) (blue). (c) Each monomer (purple and grey) has two different domain structures (Kinase (red and blue) and Endoribonuclease domains (PF06479) (green and cyan)) in their interfaces and these domains collaborations form the interface architecture. (d) The monomer highlighted with grey has the kinase domain (blue) and the complementary monomer (purple) has two different domains (Cyclin N terminal (PF00134) (red) and Cyclin C terminal (PF02984) (green) domains).

6.2 SCOP domains

SCOP [59] is a database classifying protein structures at different levels: Species, Protein, Family, SuperFamily, Fold and Class. In this work, the classification levels of Fold, SuperFamily and Family are mapped to protein interface clusters in order to reveal any possible rules in the binding strategies of protein structures. For different levels of structural classification, the numbers of interfaces vs the numbers of interface clusters are plotted in Figure 6-4b, Figure 6-4c and Figure 6-4d. Also, the

classification members of folds, superFamilies and families which have the most different binding strategies are listed in Table 6-8, Table 6-9 and Table 6-10. Structural classification of interfaces from the structurally classified domains generated by SCOP can reveal shared or different binding sites of the same SCOP hierarchy, unlike PFAM. Immunoglobulins have the most diverse interface architectures even though the structural classifications of their domains are the same. Protein kinases have the most diverse interface architectures according to the PFAM classification; however, when viewed from the standpoint of the structural similarities of the monomers rather than the functional similarity, kinases fall behind the immunoglobulins. Eventually, these unique interface architectures can be used to reveal different interface architectures of structural or functional similar domains which can be adapted to specific protein-protein interactions like immunoglobulins, kinases, trypsins, PDZ domains, ubiquitins etc.

Table 6-8 Scop Folds that have the most different binding strategies are listed

Scop fold	# of clusters	Functional Info
48725	829	Immunoglobulin-like beta-sandwich
52539	558	P-loop containing nucleoside triphosphate hydrolases
51350	543	TIM beta/alpha-barrel
54861	348	Ferredoxin-like
51734	270	NAD(P)-binding Rossmann-fold domains
54235	257	beta-Grasp (ubiquitin-like)
46688	256	DNA/RNA-binding 3-helical bundle
50493	255	Trypsin-like serine proteases
52171	220	Flavodoxin-like
48370	210	alpha-alpha superhelix

Table 6-9 Scop SuperFamilies that have the most different binding strategies are listed

Scop superFamily	# of clusters	Functional Info
48726	563	Immunoglobulin
52540	558	P-loop containing nucleoside triphosphate hydrolases
51735	270	NAD(P)-binding Rossmann-fold domains
50494	255	Trypsin-like serine proteases
56112	194	Protein kinase-like (PK-like)
52833	182	Thioredoxin-like
46785	158	"Winged helix" DNA-binding domain
47473	151	EF-hand
54452	147	MHC antigen-recognition domain
49899	147	Concanavalin A-like lectins/glucanases
53474	147	alpha/beta-Hydrolases

Table 6-10 Scop Families that have the most different binding strategies are listed

Scop family	# of clusters	Functional Info
48727	462	V set domains (antibody variable domain-like)
52592	232	G proteins
48942	205	C1 set domains (antibody constant domain-like)
88854	172	Protein kinases, catalytic subunit
50514	164	Eukaryotic proteases
54453	147	MHC antigen-recognition domain
54237	105	Ubiquitin-related
51751	95	Tyrosine-dependent oxidoreductases
47502	87	Calmodulin-like
53851	85	Phosphate binding protein-like

6.3 Concluding remarks

As a functional perspective, PFAM domains of all protein interfaces are extracted. It is shown that majority of the interfaces are built from monomers which have one domain. There are small amount of multi-domain protein interactions. Functional mapping of proteins are done and collaboration networks of the domains are

listed. When the collaboration in the monomer is investigated ATPase family associated, GHKL and Alpha amylase domains are the top three domains which are the most collaborated domains in order to form an interface. When the collaboration of domains in the partner monomer is scrutinized RNA polymerase domains are the most collaborated domains in order to form an interaction between the monomers. Also, the interaction network of the interfaces according to their domain numbers showed that the monomers are preferred interaction partners which have the same number of domains in their interfaces with them. For a final functional analysis, the PFAM domains are mapped to interface clusters in order to find the domains which exist in many different interface structures. Protein kinase, trypsin and immunoglobin C1-set domains appeared in the most different interface structures.

The SCOP classification results are also mapped to the interface clusters for different hierarchical levels such as folds, super families and families. Immunoglobulin domains present in the most different interface structures for each different hierarchical levels which should be provided by their structural properties.

Chapter 7
PIFACE: A CLUSTERED PROTEIN-PROTEIN INTERFACE DATABASE

In this chapter, a tutorial of the Piface Database (http://prism.ccbb.ku.edu.tr) that deposited all the results of interface clustering and domain analysis is presented. The main page of the database is shown in Figure 7-1.

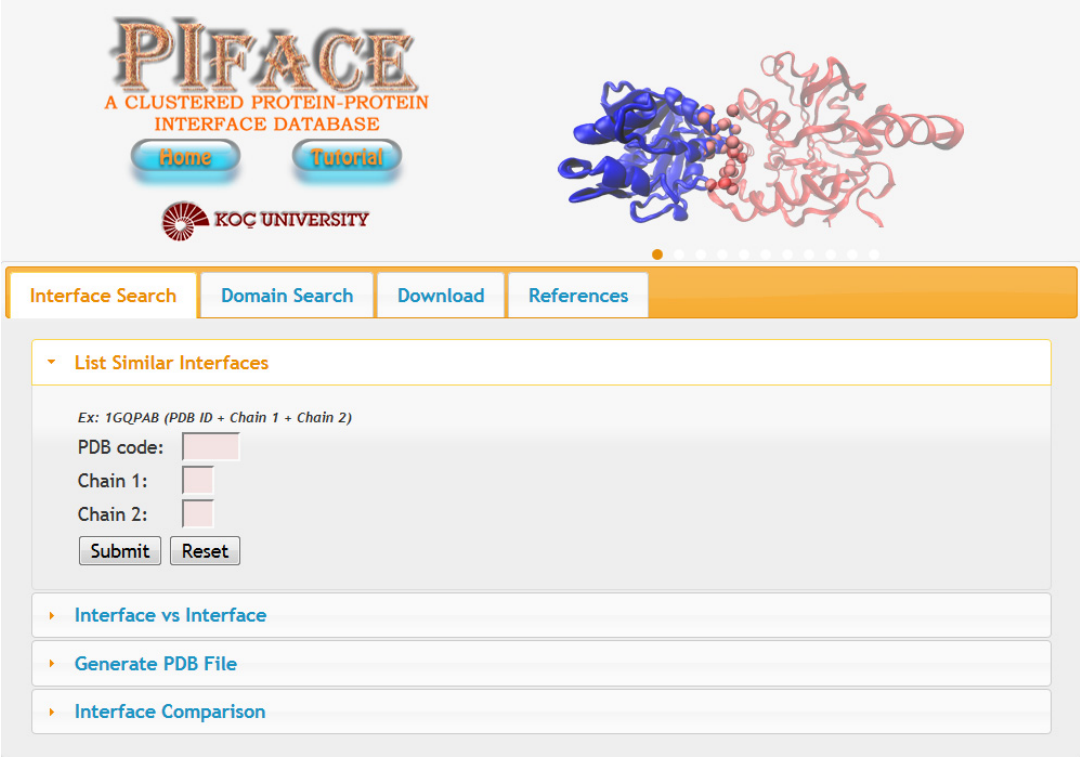


Figure 7-1 Homepage of the Piface Database

There are two main tab in the database. First one is the “Interface Search” tab which the user can retrieve similar interfaces and compare two interface structures in the database. Second one is the “Domain Search” which the user can query the domain information in interfaces.

7.1 Interface search

Interface Search section has four different search options:

1) List Similar Interfaces

2) Interface vs Interface

3) Generate PDB File

4) Interface Comparison

7.1.1 List similar interfaces

User can find similar interface structures of queried interface. The interface structure can be queried just submitting the PDB ID of the protein structure and its two consecutive chain IDs. For example, if you search the interface between the chains A and B of the 1KLM, the PIFACE database retrieve the information of the similar interface as shown in Figure 7-2.

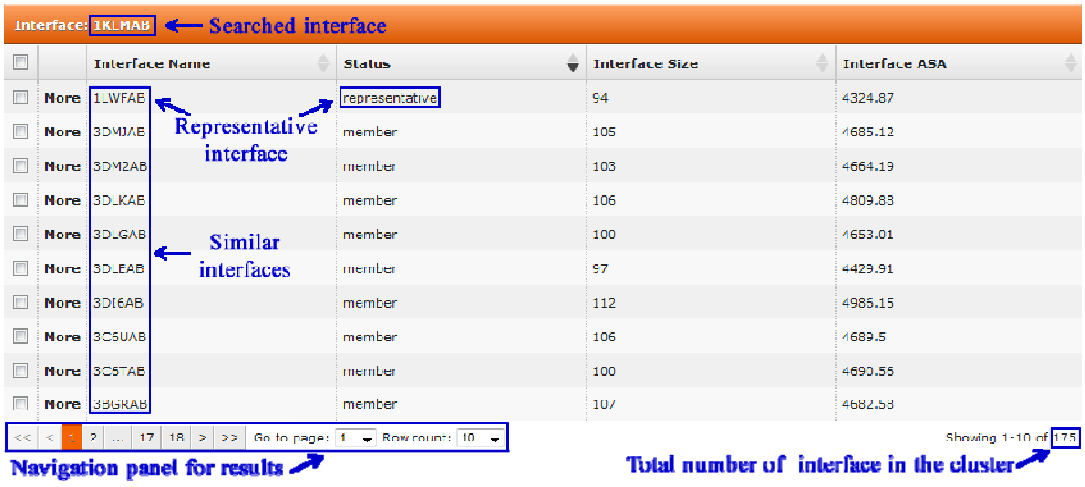


Figure 7-2 Similar interface search results

You can find the monomer properties by clicking "More" as shown in Figure 7-3.

Figure 7-3 The molecule information tab can be reached by clicking "More".

User can add/remove interface properties shown in table using popup menu by clicking right mouse button onto the header of the table (Figure 7-4). Also, when you select the interfaces a new panel will show the selected interface properties at the bottom of the page. You can sort the table by clicking the header, the arrows in the table header shows the ordering (increment/decrement). (Popup menu abbreviations: Interface Size = # of interface residues, Interface ASA = Interface Accessible Surface Area, C_1_I_Size = Chain 1 Interface Size, C_2_I_Size = Chain 2 Interface Size, C_1_MASA = Chain 1 Monomer Accessible Surface Area, C_2_MASA = Chain 2 Monomer Accessible Surface Area, C_1_CASA = Chain 1 Complex Accessible Surface Area, C_2_CASA = Chain 2 Complex Accessible Surface Area)

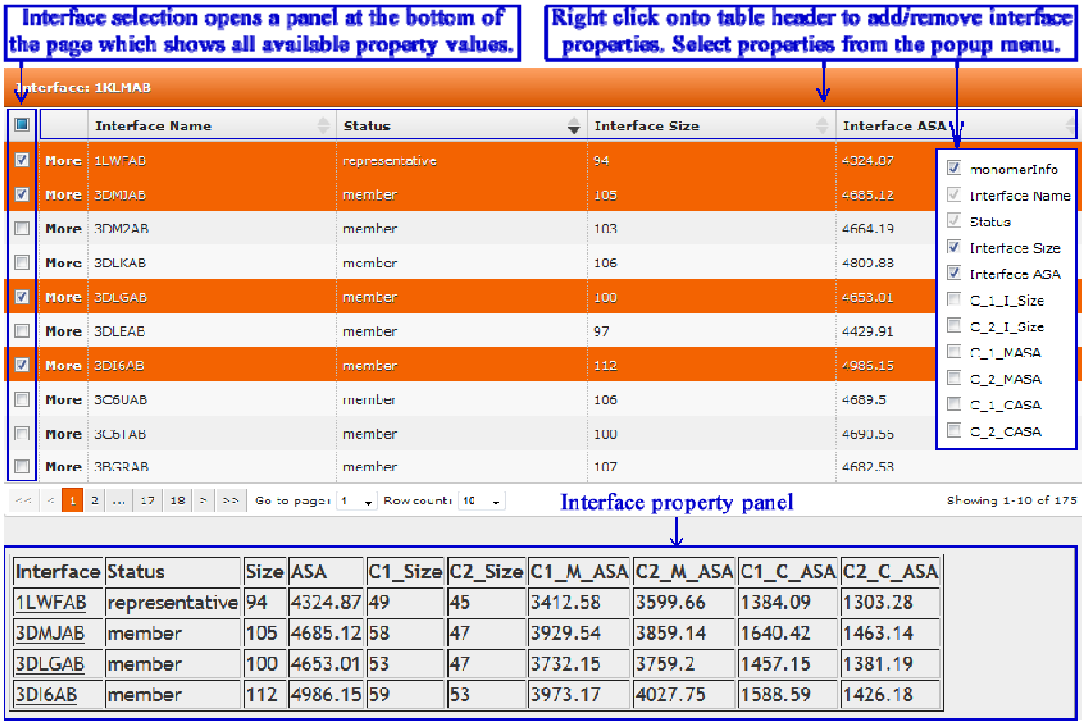


Figure 7-4 Properties of similar interface results table

7.1.2 Interface vs interface

User can compare two interfaces from Interface vs Interface Tab. If interfaces have at least 75% interface similarity, their structural alignment results can be retrieved from database Figure 7-5. The first row of the result table is dedicated to the interface residue properties and the second row shows the properties of the interface residues with nearby residues. It has also the RMSD value of the structural alignment because all the alignments are performed with interface and nearby residues. So, there is not any RMSD value for interface residues. A panel below the result table shows the complex structures of the proteins using JSmol [25].

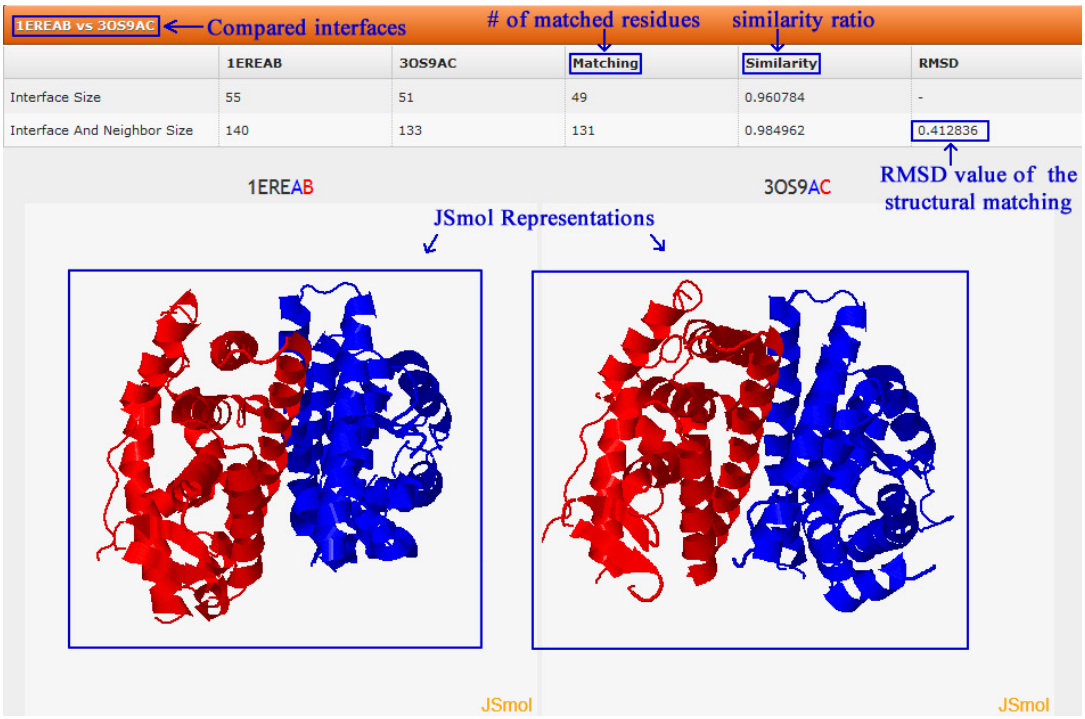


Figure 7-5 Comparison result of the two interfaces.

7.1.3 Generate PDB file

User can download complex structures with interface, nearby, hot spot and hot region residues information (Figure 7-6). Hot spot and hot region information are taken from HotRegion [88]. For example, if you search the interface between the chains L and V of the 1H2I, the PIFACE database generate your PDB file according to your choices.

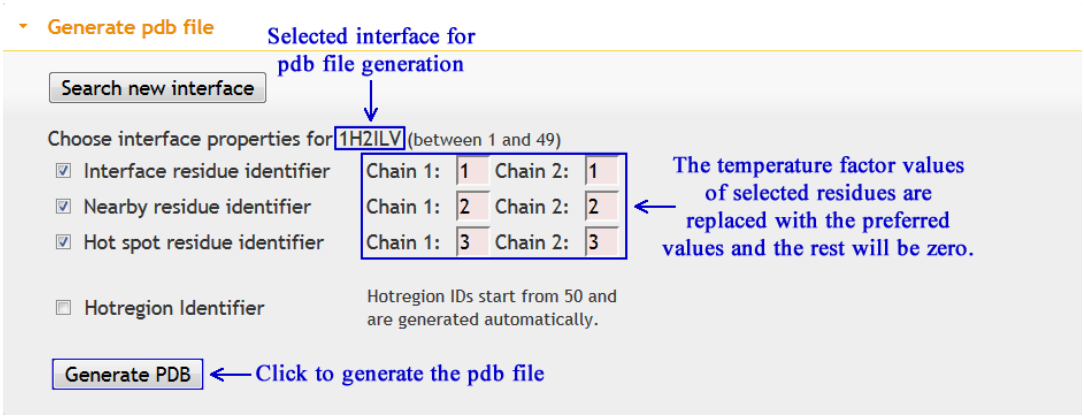


Figure 7-6 Generate PDB File Tab

User can both download and visualize the PDB file (Figure 7-7).

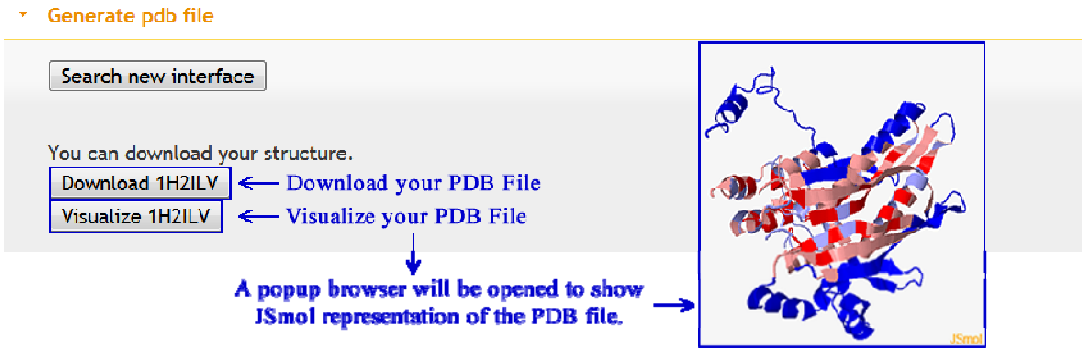


Figure 7-7 Results of PDB generation

7.1.4 Interface comparison

User can also compare two interfaces if two interfaces have lower than 75% similarity. In the PIFACE database, the interfaces which have similarity higher than 75% are stored and can be searched from "Interface vs Interface" tab under the "Interface Search" section. In "Interface Comparison" tab, user can obtain the residue alignments of the interfaces, the superimposed complex structures and the similarity value of the structural alignment of the interfaces. The comparison results of the interface between

the chains A and B of the 1GQP and the interface between the chains A and B of the 1KLM are presented in Figure 7-8.

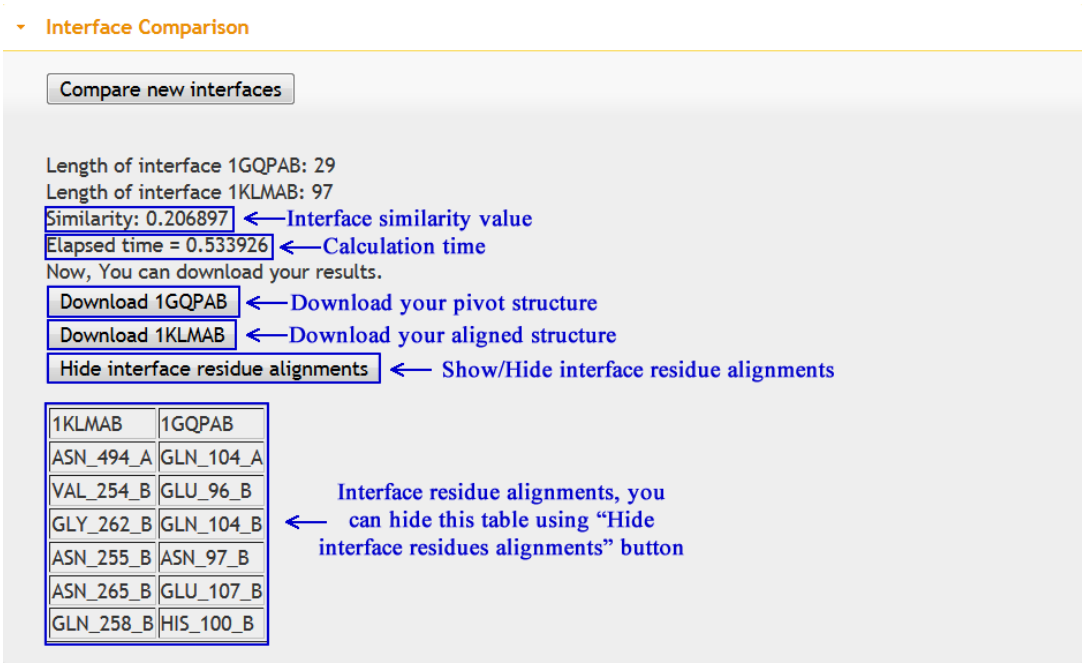


Figure 7-8 Interface comparison results of 1GQPAB and 1KLMAB.

7.2 Domain search

Domain Search section has two different search options (PFAM [63] and SCOP [59]) with three subsections.

1. PFAM Domain

   1.1 Search PFAM domains in interface

   1.2 Search interface for PFAM domain

   1.3 Search PFAM domain combinations

2. SCOP Domain

   2.1 Search SCOP domains in interface

2.2 Search interface for SCOP domain

2.3 Search SCOP domain combinations

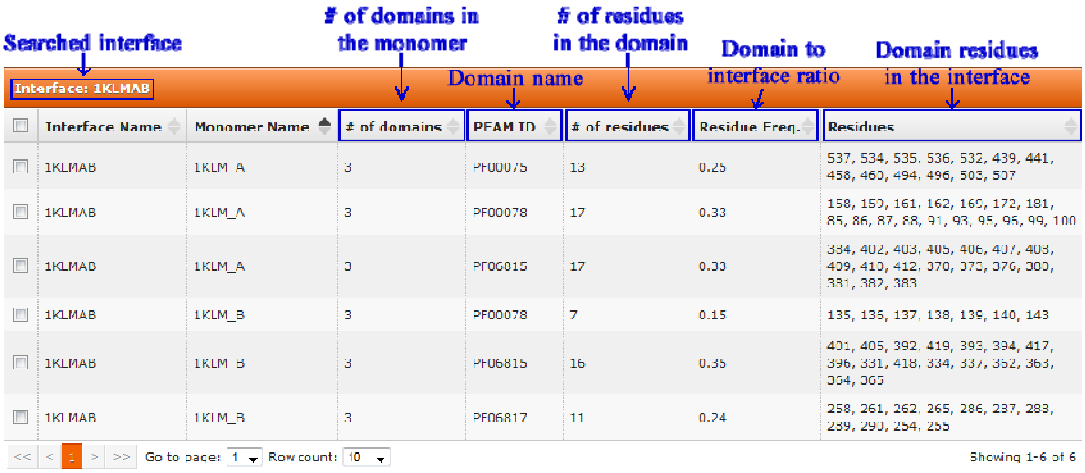
*SCOP domain results are also represented like PFAM domain results. The user should keep in mind that SCOP has different hierarchy for classifying, so using the correct SCOP type for the search option is important.

7.2.1 PFAM domain

Users can find the answers of which PFAM domains are in the queried interface, which interfaces have the queried PFAM domains or is there another domain in the same or partner monomer of the interface.

7.2.1.1 Search PFAM domains in interface

The results of example search using 1KLMAB interface is in Figure 7-9. Both chain A and chain B have three different PFAM domains in their interface structure. The chain A has PF00075, PF00078 and PF06815 domains and the residues of these domains are shown under the "Residues" column. The number of residues is presented in the "# of residues" column. The "Residue Freq." of the domains is calculated by dividing number of residues of the domain to the interface residues of the monomer.

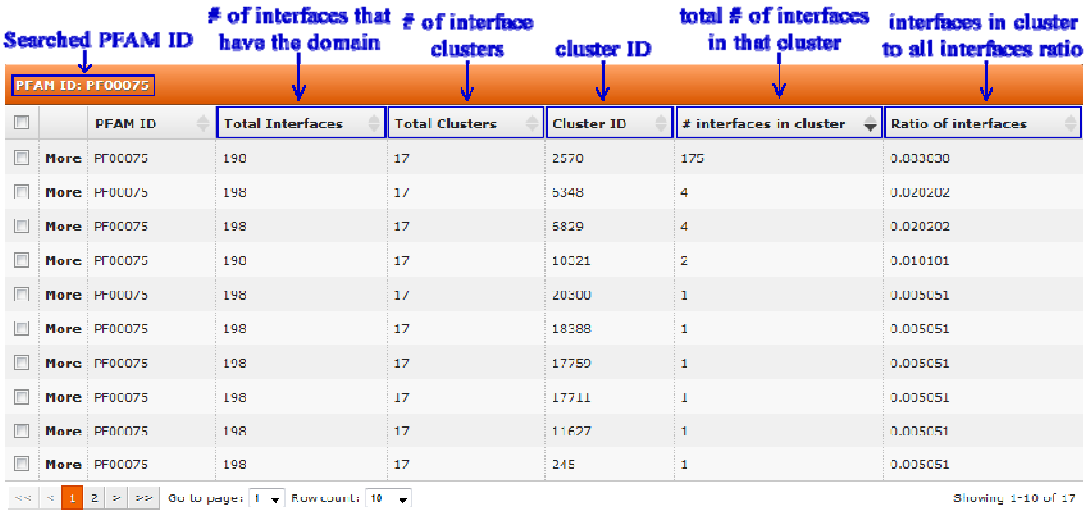


Interface Name	Monomer Name	# of domains	PFAM ID	# of residues	Residue Freq.	Residues
1KLMAB	1KLM_A	3	PF00075	13	0.25	537, 534, 535, 536, 532, 439, 441, 458, 460, 494, 496, 503, 507
1KLMAB	1KLM_A	3	PF00078	17	0.33	158, 159, 161, 162, 165, 172, 181, 85, 86, 87, 88, 91, 93, 95, 96, 99, 100
1KLMAB	1KLM_A	3	PF06015	17	0.33	384, 402, 403, 405, 406, 407, 408, 409, 410, 412, 370, 373, 376, 380, 381, 382, 383
1KLMAB	1KLM_B	3	PF00078	7	0.15	135, 136, 137, 138, 139, 140, 143
1KLMAB	1KLM_B	3	PF06815	16	0.35	401, 405, 392, 419, 393, 394, 417, 396, 331, 418, 334, 337, 352, 353, 354, 355
1KLMAB	1KLM_B	3	PF06817	11	0.24	258, 261, 262, 265, 286, 287, 288, 289, 290, 254, 255

Figure 7-9 Results of PFAM domain entries in the 1KLMAB interface.

7.2.1.2 Search interface for PFAM domain

The results of example search using PF00075 domain is in Figure 7-10. PF00075 present in 198 interfaces in the database. It is situated in 17 different interface architectures. Number of interfaces in the cluster and the ratio of the interfaces in the cluster to all interfaces that contain the domain values are presented in the result table.



	PFAM ID	Total Interfaces	Total Clusters	Cluster ID	# interfaces in cluster	Ratio of interfaces
More	PF00075	198	17	2570	175	0.883838
More	PF00075	198	17	5348	4	0.020202
More	PF00075	198	17	5829	4	0.020202
More	PF00075	198	17	10221	2	0.010101
More	PF00075	198	17	20300	1	0.005051
More	PF00075	198	17	18388	1	0.005051
More	PF00075	198	17	17750	1	0.005051
More	PF00075	198	17	17711	1	0.005051
More	PF00075	198	17	11627	1	0.005051
More	PF00075	198	17	245	1	0.005051

Figure 7-10 Results of interface entries which have the PFAM domain PF00075.

The user can access the interfaces and their properties of the each cluster which has the searched domain by clicking the “More” button (Figure 7-11). The interface 1HVUAB has PF00075 domain in chain A. The residues of chain A which belong to PF00075 are 544, 439, 460 and 543. These residues are 22% of the interface residues of chain A which interact with chain B.

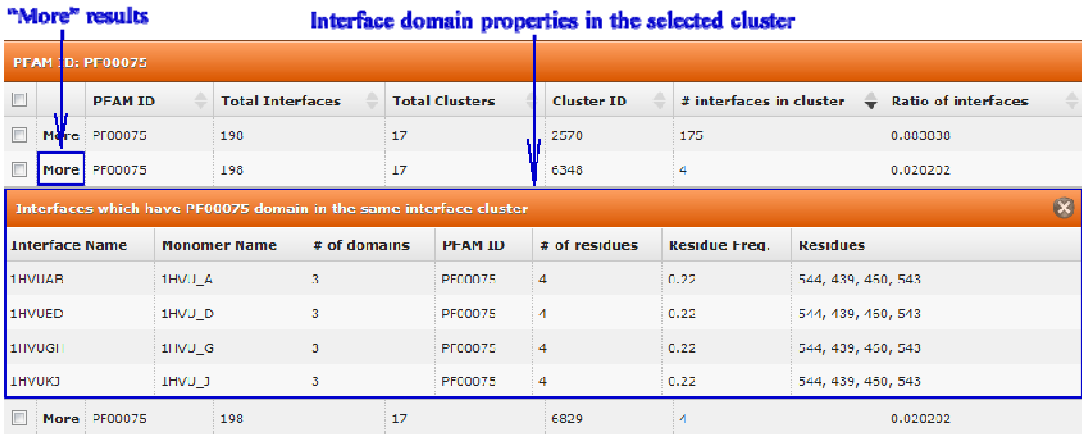


Figure 7-11 Interface properties of the selected cluster.

7.2.1.3 Search PFAM domain combinations

The domain organizations in the interface structures are gathered using all interfaces in the PIFACE database. Some domains built up the protein interfaces with other domains. These combinations are found over all the interfaces. The domains combinations are classified into two groups. The domains in the same monomer can be searched by choosing the radio button "Same", or choose "Opposite" for the domains in the partner monomer. These combinations types are explained in Chapter 6. The result of domain combination of the PF00205 in the same monomer is represented in Figure 7-12.

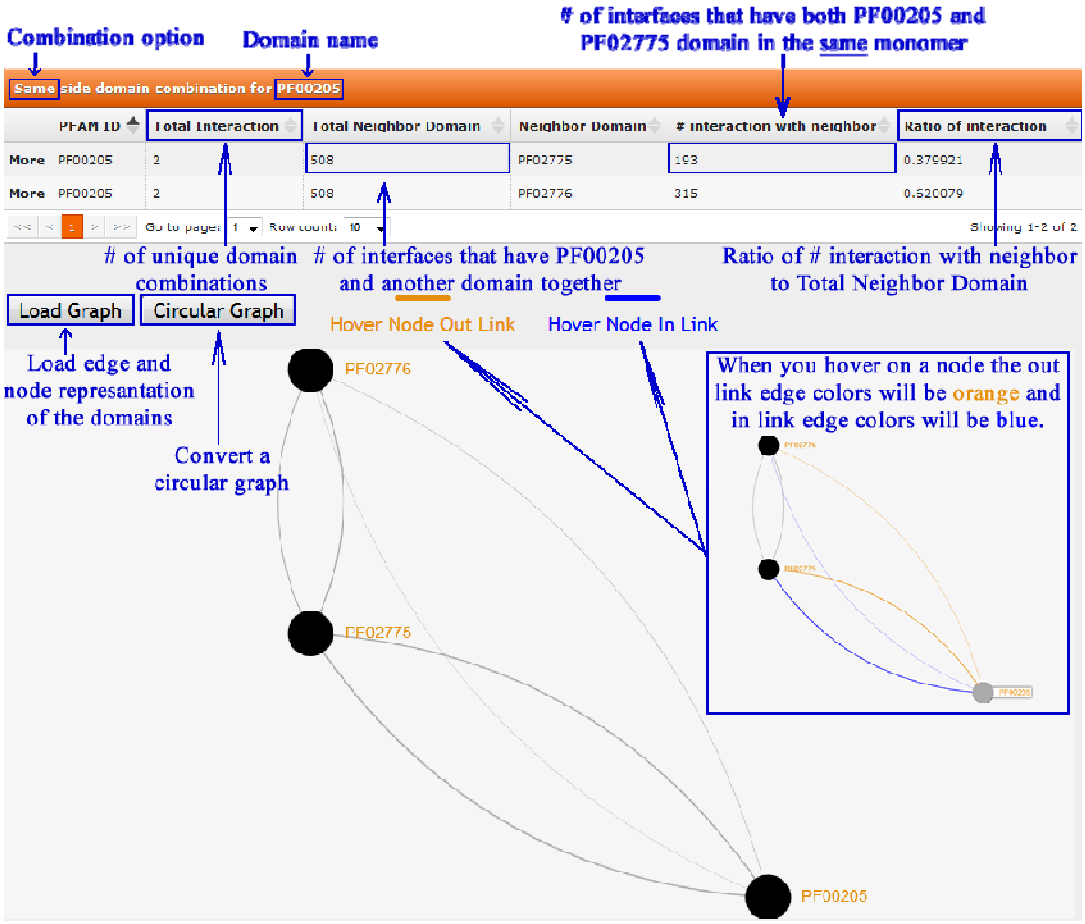


Figure 7-12 Result of domain combination of the PF00205 in the same monomer.

Chapter 8
CONCLUSION

Revealing the protein-protein interaction mechanism is important for obtaining biological functions of the proteins. Proteins are used their specific surface architectures in order to form complex structures with other proteins. These specific surface architectures, defined as interfaces, are the main characters for this dissertation. In this dissertation, the first aim is defining the interface structures in protein-protein interactions and finding the specific properties of these architectures. In order to obtain reliable results, redundancy in the structural data should be eliminated. Therefore, a new clustering method for protein interface structures is devised. The method is clearly stated from the extraction of the interface structures from protein-protein interactions to final clustering evaluation step. As a result of this clustering method, a new non-redundant dataset of protein interface architectures are found. The second aim of this dissertation is using the valuable non-redundant protein interface architecture dataset in order to extract valuable information for the question “how do proteins interact with their partner protein?”. These non-redundant protein interface architectures assist to extract multi-binding monomer proteins from PDB. Also, these interface structures can be used to predict protein-protein interactions as shown in previous studies [5, 17, 125]. Furthermore, domain information in these protein interface structures is investigated and the importance of the binding preferences among the domain partners for protein interactions is illustrated.

Overall, this dissertation presents methods to analyze protein-protein interactions using structural data and useful applications for biological problems. The non-redundant dataset is important for the knowledge based methods which predict protein-protein interactions via known protein structures. Also, domain-domain collaborations in the interfaces will be a guide for the researchers to predict protein-

protein interactions. Prediction of protein-protein interactions are important to complete the missing parts of the protein-protein interaction networks that derived using experimental results and this work will assist to increase the reliability of these predictions. Moreover, the multiple-binding monomer proteins presented in this thesis will lead to find important nodes in these biological networks which are significant to understand biological mechanism, cellular processes and drug targeting.

BIBLIOGRAPHY

1. Keskin, O., et al., *Principles of protein-protein interactions: what are the preferred ways for proteins to interact?* Chem Rev, 2008. **108**(4): p. 1225-44.
2. Han, J.D., et al., *Evidence for dynamically organized modularity in the yeast protein-protein interaction network.* Nature, 2004. **430**(6995): p. 88-93.
3. Cukuroglu, E., et al., *Non-redundant unique interface structures as templates for modeling protein interactions.* PLoS One, 2014. **9**(1): p. e86738.
4. Aloy, P. and R.B. Russell, *Ten thousand interactions for the molecular biologist.* Nat Biotechnol, 2004. **22**(10): p. 1317-21.
5. Tuncbag, N., et al., *Predicting protein-protein interactions on a proteome scale by matching evolutionary and structural similarities at interfaces using PRISM.* Nat Protoc, 2011. **6**(9): p. 1341-54.
6. Kundrotas, P.J., et al., *Templates are available to model nearly all complexes of structurally characterized proteins.* Proc Natl Acad Sci U S A, 2012. **109**(24): p. 9438-41.
7. Keskin, O., et al., *A new, structurally nonredundant, diverse data set of protein-protein interfaces and its implications.* Protein Sci, 2004. **13**(4): p. 1043-55.
8. Tsai, C.J., B. Ma, and R. Nussinov, *Protein-protein interaction networks: how can a hub protein bind so many different partners?* Trends Biochem Sci, 2009. **34**(12): p. 594-600.
9. Yogurtcu, O.N., et al., *Restricted mobility of conserved residues in protein-protein interfaces in molecular simulations.* Biophys J, 2008. **94**(9): p. 3475-85.
10. Rajamani, D., et al., *Anchor residues in protein-protein interactions.* Proc Natl Acad Sci U S A, 2004. **101**(31): p. 11287-92.
11. Moreira, I.S., P.A. Fernandes, and M.J. Ramos, *Hot spots--a review of the protein-protein interface determinant amino-acid residues.* Proteins, 2007. **68**(4): p. 803-12.
12. Keskin, O. and R. Nussinov, *Favorable scaffolds: proteins with different sequence, structure and function may associate in similar ways.* Protein Eng Des Sel, 2005. **18**(1): p. 11-24.
13. Keskin, O. and R. Nussinov, *Similar binding sites and different partners: implications to shared proteins in cellular pathways.* Structure, 2007. **15**(3): p. 341-54.
14. Tsai, C.J., et al., *Protein-protein interfaces: architectures and interactions in protein-protein interfaces and in protein cores. Their similarities and differences.* Crit Rev Biochem Mol Biol, 1996. **31**(2): p. 127-52.
15. Tuncbag, N., et al., *Fast and accurate modeling of protein-protein interactions by combining template-interface-based docking with flexible refinement.* Proteins, 2012. **80**(4): p. 1239-49.
16. Kuzu, G., et al., *Exploiting conformational ensembles in modeling protein-protein interactions on the proteome scale.* J Proteome Res, 2013. **12**(6): p. 2641-53.
17. Aytuna, A.S., A. Gursoy, and O. Keskin, *Prediction of protein-protein interactions by combining structure and sequence conservation in protein interfaces.* Bioinformatics, 2005. **21**(12): p. 2850-5.
18. Kundrotas, P.J. and I.A. Vakser, *Global and local structural similarity in protein-protein complexes: Implications for template-based docking.* Proteins, 2013.

19. Kleanthous, C., *Protein–Protein Recognition, Frontiers in Molecular Biology*. 2000: Oxford University Press.
20. Reynolds, C., D. Damerell, and S. Jones, *ProtorP: a protein-protein interaction analysis server.* Bioinformatics, 2009. **25**(3): p. 413-4.
21. Gong, S., et al., *A protein domain interaction interface database: InterPare.* BMC Bioinformatics, 2005. **6**: p. 207.
22. Stein, A., A. Ceol, and P. Aloy, *3did: identification and classification of domain-based interactions of known three-dimensional structure.* Nucleic Acids Res, 2011. **39**(Database issue): p. D718-23.
23. Stein, A., A. Panjkovich, and P. Aloy, *3did Update: domain-domain and peptide-mediated interactions of known 3D structure.* Nucleic Acids Res, 2009. **37**(Database issue): p. D300-4.
24. Stein, A., R.B. Russell, and P. Aloy, *3did: interacting protein domains of known three-dimensional structure.* Nucleic Acids Res, 2005. **33**(Database issue): p. D413-7.
25. Davis, F.P. and A. Sali, *PIBASE: a comprehensive database of structurally defined protein interfaces.* Bioinformatics, 2005. **21**(9): p. 1901-7.
26. Tuncbag, N., et al., *Architectures and functional coverage of protein-protein interfaces.* J Mol Biol, 2008. **381**(3): p. 785-802.
27. Nooren, I.M. and J.M. Thornton, *Diversity of protein-protein interactions.* EMBO J, 2003. **22**(14): p. 3486-92.
28. Jones, S. and J.M. Thornton, *Analysis of protein-protein interaction sites using surface patches.* J Mol Biol, 1997. **272**(1): p. 121-32.
29. Lo Conte, L., C. Chothia, and J. Janin, *The atomic structure of protein-protein recognition sites.* J Mol Biol, 1999. **285**(5): p. 2177-98.
30. Vakser, I.A., *Protein-protein interfaces are special.* Structure, 2004. **12**(6): p. 910-2.
31. Chakrabarti, P. and J. Janin, *Dissecting protein-protein recognition sites.* Proteins, 2002. **47**(3): p. 334-43.
32. Jones, S. and J.M. Thornton, *Principles of protein-protein interactions.* Proc Natl Acad Sci U S A, 1996. **93**(1): p. 13-20.
33. Tsai, C.J., et al., *Studies of protein-protein interfaces: a statistical analysis of the hydrophobic effect.* Protein Sci, 1997. **6**(1): p. 53-64.
34. Argos, P., *An investigation of protein subunit and domain interfaces.* Protein Eng, 1988. **2**(2): p. 101-13.
35. Chothia, C. and J. Janin, *Principles of protein-protein recognition.* Nature, 1975. **256**(5520): p. 705-8.
36. Janin, J. and C. Chothia, *The structure of protein-protein recognition sites.* J Biol Chem, 1990. **265**(27): p. 16027-30.
37. Korn, A.P. and R.M. Burnett, *Distribution and complementarity of hydropathy in multisubunit proteins.* Proteins, 1991. **9**(1): p. 37-55.
38. Lijnzaad, P., H.J. Berendsen, and P. Argos, *Hydrophobic patches on the surfaces of protein structures.* Proteins, 1996. **25**(3): p. 389-97.
39. Sheinerman, F.B., R. Norel, and B. Honig, *Electrostatic aspects of protein-protein interactions.* Curr Opin Struct Biol, 2000. **10**(2): p. 153-9.
40. Xu, W. and F.E. Regnier, *Protein-protein interactions on weak-cation-exchange sorbent surfaces during chromatographic separations.* J Chromatogr A, 1998. **828**(1-2): p. 357-64.

41. Levy, E.D. and J.B. Pereira-Leal, *Evolution and dynamics of protein interactions and networks.* Curr Opin Struct Biol, 2008. **18**(3): p. 349-57.
42. Sheinerman, F.B. and B. Honig, *On the role of electrostatic interactions in the design of protein-protein interfaces.* J Mol Biol, 2002. **318**(1): p. 161-77.
43. Gavenonis, J., et al., *Comprehensive analysis of loops at protein-protein interfaces for macrocycle design.* Nat Chem Biol, 2014. **10**(9): p. 716-22.
44. Azzarito, V., et al., *Inhibition of alpha-helix-mediated protein-protein interactions using designed molecules.* Nat Chem, 2013. **5**(3): p. 161-73.
45. Elcock, A.H. and J.A. McCammon, *Identification of protein oligomerization states by analysis of interface conservation.* Proc Natl Acad Sci U S A, 2001. **98**(6): p. 2990-4.
46. Teichmann, S.A., *The constraints protein-protein interactions place on sequence divergence.* J Mol Biol, 2002. **324**(3): p. 399-407.
47. Valdar, W.S. and J.M. Thornton, *Protein-protein interfaces: analysis of amino acid conservation in homodimers.* Proteins, 2001. **42**(1): p. 108-24.
48. Eames, M. and T. Kortemme, *Structural mapping of protein interactions reveals differences in evolutionary pressures correlated to mRNA level and protein abundance.* Structure, 2007. **15**(11): p. 1442-51.
49. Franzosa, E.A. and Y. Xia, *Structural determinants of protein evolution are context-sensitive at the residue level.* Mol Biol Evol, 2009. **26**(10): p. 2387-95.
50. Nooren, I.M. and J.M. Thornton, *Structural characterisation and functional significance of transient protein-protein interactions.* J Mol Biol, 2003. **325**(5): p. 991-1018.
51. Mintseris, J. and Z. Weng, *Structure, function, and evolution of transient and obligate protein-protein interactions.* Proc Natl Acad Sci U S A, 2005. **102**(31): p. 10930-5.
52. Valdar, W.S. and J.M. Thornton, *Conservation helps to identify biologically relevant crystal contacts.* J Mol Biol, 2001. **313**(2): p. 399-416.
53. Mintseris, J. and Z. Weng, *Atomic contact vectors in protein-protein recognition.* Proteins, 2003. **53**(3): p. 629-39.
54. Carugo, O. and P. Argos, *Protein-protein crystal-packing contacts.* Protein Sci, 1997. **6**(10): p. 2261-3.
55. Zhu, H., et al., *NOXclass: prediction of protein-protein interaction types.* BMC Bioinformatics, 2006. **7**: p. 27.
56. Liu, Q. and J. Li, *Propensity vectors of low-ASA residue pairs in the distinction of protein interactions.* Proteins, 2010. **78**(3): p. 589-602.
57. Duarte, J.M., et al., *Protein interface classification by evolutionary analysis.* BMC Bioinformatics, 2012. **13**: p. 334.
58. Liu, Q., Z. Li, and J. Li, *Use B-factor related features for accurate classification between protein binding interfaces and crystal packing contacts.* BMC Bioinformatics, 2014. **15 Suppl 16**: p. S3.
59. Andreeva, A., et al., *Data growth and its impact on the SCOP database: new developments.* Nucleic Acids Res, 2008. **36**(Database issue): p. D419-25.
60. Sillitoe, I., et al., *New functional families (FunFams) in CATH to improve the mapping of conserved functional sites to 3D structures.* Nucleic Acids Res, 2013. **41**(Database issue): p. D490-8.
61. Krissinel, E. and K. Henrick, *Inference of macromolecular assemblies from crystalline state.* J Mol Biol, 2007. **372**(3): p. 774-97.

62. Teyra, J., et al., *SCOWLP: a web-based database for detailed characterization and visualization of protein interfaces.* BMC Bioinformatics, 2006. **7**: p. 104.
63. Punta, M., et al., *The Pfam protein families database.* Nucleic Acids Res, 2012. **40**(Database issue): p. D290-301.
64. De, S., et al., *Interaction preferences across protein-protein interfaces of obligatory and non-obligatory components are different.* BMC Struct Biol, 2005. **5**: p. 15.
65. Tyagi, M., et al., *Homology inference of protein-protein interactions via conserved binding sites.* PLoS One, 2012. **7**(1): p. e28896.
66. Aloy, P. and R.B. Russell, *InterPreTS: protein interaction prediction through tertiary structure.* Bioinformatics, 2003. **19**(1): p. 161-2.
67. Dayhoff, J.E., et al., *Evolution of protein binding modes in homooligomers.* J Mol Biol, 2010. **395**(4): p. 860-70.
68. Kim, W.K., et al., *The many faces of protein-protein interactions: A compendium of interface geometry.* PLoS Comput Biol, 2006. **2**(9): p. e124.
69. Ghoorah, A.W., et al., *Spatial clustering of protein binding sites for template based protein docking.* Bioinformatics, 2011. **27**(20): p. 2820-7.
70. Winter, C., et al., *SCOPPI: a structural classification of protein-protein interfaces.* Nucleic Acids Res, 2006. **34**(Database issue): p. D310-4.
71. Xu, Q. and R.L. Dunbrack, Jr., *The protein common interface database (ProtCID)--a comprehensive database of interactions of homologous proteins in multiple crystal forms.* Nucleic Acids Res, 2011. **39**(Database issue): p. D761-70.
72. Garma, L., et al., *How many protein-protein interactions types exist in nature?* PLoS One, 2012. **7**(6): p. e38913.
73. Bordner, A.J. and A.A. Gorin, *Comprehensive inventory of protein complexes in the Protein Data Bank from consistent classification of interfaces.* BMC Bioinformatics, 2008. **9**: p. 234.
74. Gao, Y., R. Wang, and L. Lai, *Structure-based method for analyzing protein-protein interfaces.* J Mol Model, 2004. **10**(1): p. 44-54.
75. Tsai, C.J., et al., *A dataset of protein-protein interfaces generated with a sequence-order-independent comparison technique.* J Mol Biol, 1996. **260**(4): p. 604-20.
76. Tseng, Y.Y. and W.H. Li, *PSC: protein surface classification.* Nucleic Acids Res, 2012. **40**(Web Server issue): p. W435-9.
77. Teyra, J., et al., *SCOWLP classification: structural comparison and analysis of protein binding regions.* BMC Bioinformatics, 2008. **9**: p. 9.
78. Barabasi, A.L. and R. Albert, *Emergence of scaling in random networks.* Science, 1999. **286**(5439): p. 509-12.
79. Jeong, H., et al., *Lethality and centrality in protein networks.* Nature, 2001. **411**(6833): p. 41-2.
80. Freeman, L.C., *A Set of Measures of Centrality Based on Betweenness.* Sociometry, 1977. **40**(1): p. 35-41.
81. Girvan, M., et al., *Simple model of epidemics with pathogen mutation.* Phys Rev E Stat Nonlin Soft Matter Phys, 2002. **65**(3 Pt 1): p. 031915.
82. Clackson, T. and J.A. Wells, *A hot spot of binding energy in a hormone-receptor interface.* Science, 1995. **267**(5196): p. 383-6.
83. Bogan, A.A. and K.S. Thorn, *Anatomy of hot spots in protein interfaces.* J Mol Biol, 1998. **280**(1): p. 1-9.

84. Guharoy, M. and P. Chakrabarti, *Conservation and relative importance of residues across protein-protein interfaces.* Proc Natl Acad Sci U S A, 2005. **102**(43): p. 15447-52.
85. Moreira, I.S., P.A. Fernandes, and M.J. Ramos, *Computational alanine scanning mutagenesis--an improved methodological approach.* J Comput Chem, 2007. **28**(3): p. 644-54.
86. Keskin, O., B. Ma, and R. Nussinov, *Hot regions in protein--protein interactions: the organization and contribution of structurally conserved hot spot residues.* J Mol Biol, 2005. **345**(5): p. 1281-94.
87. Meenan, N.A., et al., *The structural and energetic basis for high selectivity in a high-affinity protein-protein interaction.* Proc Natl Acad Sci U S A, 2010. **107**(22): p. 10080-5.
88. Cukuroglu, E., A. Gursoy, and O. Keskin, *HotRegion: a database of predicted hot spot clusters.* Nucleic Acids Res, 2012. **40**(Database issue): p. D829-33.
89. Shulman-Peleg, A., et al., *Spatial chemical conservation of hot spot interactions in protein-protein complexes.* BMC Biol, 2007. **5**: p. 43.
90. Cukuroglu, E., A. Gursoy, and O. Keskin, *Analysis of hot region organization in hub proteins.* Ann Biomed Eng, 2010. **38**(6): p. 2068-78.
91. Cho, K.I., D. Kim, and D. Lee, *A feature-based approach to modeling protein-protein interaction hot spots.* Nucleic Acids Res, 2009. **37**(8): p. 2672-87.
92. Darnell, S.J., D. Page, and J.C. Mitchell, *An automated decision-tree approach to predicting protein interaction hot spots.* Proteins, 2007. **68**(4): p. 813-23.
93. del Sol, A. and P. O'Meara, *Small-world network approach to identify key residues in protein-protein interaction.* Proteins, 2005. **58**(3): p. 672-82.
94. Grosdidier, S. and J. Fernandez-Recio, *Identification of hot-spot residues in protein-protein interactions by computational docking.* BMC Bioinformatics, 2008. **9**: p. 447.
95. Guerois, R., J.E. Nielsen, and L. Serrano, *Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations.* J Mol Biol, 2002. **320**(2): p. 369-87.
96. Huo, S., I. Massova, and P.A. Kollman, *Computational alanine scanning of the 1:1 human growth hormone-receptor complex.* J Comput Chem, 2002. **23**(1): p. 15-27.
97. Kortemme, T. and D. Baker, *A simple physical model for binding energy hot spots in protein-protein complexes.* Proc Natl Acad Sci U S A, 2002. **99**(22): p. 14116-21.
98. Li, J. and Q. Liu, *'Double water exclusion': a hypothesis refining the O-ring theory for the hot spots at protein interfaces.* Bioinformatics, 2009. **25**(6): p. 743-50.
99. Ofran, Y. and B. Rost, *Protein-protein interaction hotspots carved into sequences.* PLoS Comput Biol, 2007. **3**(7): p. e119.
100. Tuncbag, N., A. Gursoy, and O. Keskin, *Identification of computational hot spots in protein interfaces: combining solvent accessibility and inter-residue potentials improves the accuracy.* Bioinformatics, 2009. **25**(12): p. 1513-20.
101. Fernández-Recio, J., *Prediction of protein binding sites and hot spots.* WIREs Comput Mol Sci, 2011. **1**(5): p. 18.
102. Xu, Q., et al., *Statistical analysis of interface similarity in crystals of homologous proteins.* J Mol Biol, 2008. **381**(2): p. 487-507.
103. Lu, L., H. Lu, and J. Skolnick, *MULTIPROSPECTOR: an algorithm for the prediction of protein-protein interactions by multimeric threading.* Proteins, 2002. **49**(3): p. 350-64.

104. Hugo, W., et al., *SLiM on Diet: finding short linear motifs on domain interaction interfaces in Protein Data Bank.* Bioinformatics, 2010. **26**(8): p. 1036-42.
105. Zhu, H., et al., *Alignment of non-covalent interactions at protein-protein interfaces.* PLoS One, 2008. **3**(4): p. e1926.
106. Gunther, S., et al., *Docking without docking: ISEARCH--prediction of interactions using known interfaces.* Proteins, 2007. **69**(4): p. 839-44.
107. Benoit, V., et al., *PPIDD: an extraction and visualisation method of biological protein-protein interfaces.* Biochimie, 2008. **90**(4): p. 640-7.
108. Dafas, P., et al., *Using convex hulls to extract interaction interfaces from known structures.* Bioinformatics, 2004. **20**(10): p. 1486-90.
109. Vanhee, P., et al., *Protein-peptide interactions adopt the same structural motifs as monomeric protein folds.* Structure, 2009. **17**(8): p. 1128-36.
110. Aloy, P. and R.B. Russell, *Interrogating protein interaction networks through structural biology.* Proc Natl Acad Sci U S A, 2002. **99**(9): p. 5896-901.
111. Caffrey, D.R., et al., *Are protein-protein interfaces more conserved in sequence than the rest of the protein surface?* Protein Sci, 2004. **13**(1): p. 190-202.
112. Stein, A. and P. Aloy, *Novel peptide-mediated interactions derived from high-resolution 3-dimensional structures.* PLoS Comput Biol, 2010. **6**(5): p. e1000789.
113. Saha, R.P., R.P. Bahadur, and P. Chakrabarti, *Interresidue contacts in proteins and protein-protein interfaces and their use in characterizing the homodimeric interface.* J Proteome Res, 2005. **4**(5): p. 1600-9.
114. Headd, J.J., et al., *Protein-protein interfaces: properties, preferences, and projections.* J Proteome Res, 2007. **6**(7): p. 2576-86.
115. Keskin, O., et al., *Protein-protein interactions: organization, cooperativity and mapping in a bottom-up Systems Biology approach.* Phys Biol, 2005. **2**(2): p. S24-35.
116. Hubbard, S.J., S.F. Campbell, and J.M. Thornton, *Molecular recognition. Conformational analysis of limited proteolytic sites and serine proteinase protein inhibitors.* J Mol Biol, 1991. **220**(2): p. 507-30.
117. Shatsky, M., R. Nussinov, and H.J. Wolfson, *A method for simultaneous alignment of multiple protein structures.* Proteins, 2004. **56**(1): p. 143-56.
118. Gao, M. and J. Skolnick, *Structural space of protein-protein interfaces is degenerate, close to complete, and highly connected.* Proc Natl Acad Sci U S A, 2010. **107**(52): p. 22517-22.
119. Engin, H.B., et al., *A strategy based on protein-protein interface motifs may help in identifying drug off-targets.* J Chem Inf Model, 2012. **52**(8): p. 2273-86.
120. Rousseeuw, P.J., *Silhouettes: a graphical aid to the interpretation and validation of cluster analysis.* J. Comput. Appl. Math., 1987. **20**: p. 53-65.
121. Handl, J., J. Knowles, and D.B. Kell, *Computational cluster validation in post-genomic data analysis.* Bioinformatics, 2005. **21**(15): p. 3201-12.
122. A Hagberg, D.S., P Swart, *Exploring Network Structure, Dynamics, and Function using NetworkX*, in *Proceedings of the 7th Python in Science conference (SciPy 2008)*, T.V. Gäel Varoquaux, Jarrod Millman, Editor. 2008: Pasadena, CA USA. p. 11 - 15.
123. Kar, G., et al., *Human proteome-scale structural modeling of E2-E3 interactions exploiting interface motifs.* J Proteome Res, 2012. **11**(2): p. 1196-207.
124. Acuner Ozbabacan, S.E., et al., *Enriching the human apoptosis pathway by predicting the structures of protein-protein complexes.* J Struct Biol, 2012. **179**(3): p. 338-46.

125. Baspinar, A., et al., *PRISM: a web server and repository for prediction of protein-protein interactions and modeling their 3D complexes.* Nucleic Acids Res, 2014. **42**(Web Server issue): p. W285-9.
126. Keskin, O., R. Nussinov, and A. Gursoy, *PRISM: protein-protein interaction prediction by structural matching.* Methods Mol Biol, 2008. **484**: p. 505-21.
127. Ogmen, U., et al., *PRISM: protein interactions by structural matching.* Nucleic Acids Res, 2005. **33**(Web Server issue): p. W331-6.
128. Krissinel, E., *On the relationship between sequence and structure similarities in proteomics.* Bioinformatics, 2007. **23**(6): p. 717-23.
129. Hamp, T. and B. Rost, *Alternative protein-protein interfaces are frequent exceptions.* PLoS Comput Biol, 2012. **8**(8): p. e1002623.
130. Wells, J.A., *Systematic mutational analyses of protein-protein interfaces.* Methods Enzymol, 1991. **202**: p. 390-411.
131. Kim, P.M., et al., *Relating three-dimensional structures to protein networks provides evolutionary insights.* Science, 2006. **314**(5807): p. 1938-41.
132. Thangudu, R.R., et al., *Modulating protein-protein interactions with small molecules: the importance of binding hotspots.* J Mol Biol, 2012. **415**(2): p. 443-53.
133. Schiro, M.M., et al., *Mutations in protein-binding hot-spots on the hub protein Smad3 differentially affect its protein interactions and Smad3-regulated gene expression.* PLoS One, 2011. **6**(9): p. e25021.
134. Cukuroglu, E., et al., *Non-redundant unique interface structures as templates for modeling protein interactions.* PLoS One, 2014. **in press**.
135. Berman, H.M., et al., *The Protein Data Bank.* Nucleic Acids Res, 2000. **28**(1): p. 235-42.
136. Koide, S., *Generation of new protein functions by nonhomologous combinations and rearrangements of domains and modules.* Curr Opin Biotechnol, 2009. **20**(4): p. 398-404.
137. Bashton, M. and C. Chothia, *The generation of new protein functions by the combination of domains.* Structure, 2007. **15**(1): p. 85-99.
138. Cohen-Gihon, I., R. Nussinov, and R. Sharan, *Comprehensive analysis of co-occurring domain sets in yeast proteins.* BMC Genomics, 2007. **8**: p. 161.
139. Scheeff, E.D. and P.E. Bourne, *Structural evolution of the protein kinase-like superfamily.* PLoS Comput Biol, 2005. **1**(5): p. e49.

Appendix A
Supplementary Tables

Supplementary Table 1: The computational hot spot residues of actin – actin interfaces. (e.g. 3B5U:J,L means that the monomer chains J and L in the PDB ID 3B5U formed an interface.)

Interface	First Chain Hot spot residues	Second Chain Hot spot residues
3B5U:J,L	TYR198 VAL209 GLU205 LEU242 GLN246 LEU193 ILE248 ILE212 PHE200	CYS285 TYR294 ILE289 LEU171 ILE287 ASP286
2Y83:S,T	ARG39	ILE175 MET176 PHE266 ILE267 MET269
3B5U:I,J	ARG196 ILE64 GLU253 ASN252 SER199 GLU195	LYS113 ASN111 ARG177 LEU178 ILE75 LEU110
3G37:Q,R	ARG62 LYS61	GLU270 GLN263
2V51:B,D		
3B5U:E,G	GLU205	ILE287
3B5U:A,B	LEU67 LEU65	ILE267 LEU171 CYS285
3B5U:A,C	MET44	TYR166 TYR169
3B5U:G,F	HIS173 ARG177 LEU110	TYR198 ARG196 MET190
2Y83:S,R	ARG177 LEU110 CYS285	ARG39 ILE267 PHE266 ALA204 VAL201
3B5U:G,I		
3B5U:B,D	VAL45 MET44	ALA170 LEU171 ILE289 ILE165 PRO164
3B5U:D,F	VAL45 VAL43 ALA58 ILE64 GLY42 LEU65 MET44	VAL163 PRO164 ILE151 CYS285 ILE165 LEU171 GLY150 ILE287 ILE289 LEU293 TYR166 LEU142
1RFQ:A,B	ILE369 HIS371	CYS374 ALA365
2VCP:A,B	TYR53 TYR91 PRO98	GLN49 TYR53 GLN41 TYR91 PRO98
3B5U:K,J	HIS173 ALA174 LEU171 ARG177 CYS285	ARG62 PHE266
3B5U:K,M	PHE200 ILE208 LEU242	ILE287
3B5U:C,B		ARG62 GLY63 LEU65

3B5U:M,N	ALA58 HIS40 TYR53 GLY63 VAL35	ILE267 MET269 ALA272 LEU180 LEU171 GLU276 GLU270
1MVW:1,2	LEU110	ILE267 ARG196 ARG256
2Y83:O,P	ILE64	
3B5U:K,L	ARG62 ILE192 PHE200 TYR198 LEU65	HIS73 LEU110 ILE76 ILE75 LEU171 ARG177
3B63:K,J	LEU171 LEU110	ILE262 MET264
3B5U:E,D	ASP179 LEU110 ARG177 GLU270 ILE75 GLN263	TYR198 VAL201 GLU205 ARG39 ARG196
3B63:H,J	GLU205 ILE64	MET320 ILE282 ASP283 ILE284
1IJJ:A,B	ILE345 TYR143 ILE341	ILE464 MET444
3B5U:L,N	LEU242 ASP244	ARG290 TYR294
3B5U:I,K	ALA204	
3B5U:C,D	ARG62 PHE200	
3B5U:M,L	ARG177 ILE75 GLU276 GLN263 HIS73	ILE64 LEU65 MET44 ARG39 VAL201
1O19:1,Y	GLU205	ILE287 CYS285 ILE289
2W49:M,O	CYS285	LEU242 GLU205
1O1D:0,2	TYR166 ILE289	ILE64
2HMP:A,B		TYR143
3G37:T,V		ILE287 ALA170 CYS374 TYR169 ILE136 VAL139 CYS285
2GWK:A,B	ILE357 ILE369 MET355 TRP356	CYS374 ILE369 ILE357 MET355 TRP356
1O1D:1,0	GLU253 ARG256 ARG196	ILE75 LEU110
3B5U:E,F	ASP157 ARG206 ARG183 ARG62 GLN59 MET190 ILE208 LEU180 SER199 VAL209 GLU195 ILE192	MET227 LEU180 ILE282 ILE175 MET283 SER271 MET176 TYR188 ARG177 GLU270 LEU178 ILE289 ARG290 PHE223 THR277
2Y83:R,T	GLU205 ILE208 ILE248 LEU242	LEU171 ILE287 ARG290 ILE289 ALA170 TYR169 MET355 CYS374 TYR143 TYR294 LEU346 TYR166
3TPQ:A,D	MET283 ALA321 ILE287 TYR279	TYR337

Interface	Main Chain	Partner Chain	Main Chain Hot spot residues	Partner Chain Hot spot residues
3M3J:C,D	C	D	ILE44 ARG42 VAL70 GLN49	VAL70 GLN49 ILE44
3OJ3:C,B	C	B		
2O6V:E,G	E	G		
2XEW:K,J	K	J	ILE13 GLN40 ILE36 LEU69 VAL70 LEU71	ILE36 VAL70 LEU71 LEU73 ILE13 GLN40
2PE9:A,B	A	B	HIS68 PHE45 THR66 ALA46	ILE36 GLU34 GLN41 LEU71 GLN40
2QHO:A,E	A	E		PHE45
3M3J:A,E	A	E	ILE36 LEU73 GLN40 LEU71	GLN40 LEU71 ILE36 LEU73
1F9J:A,B	A	B		TYR159
2C7N:J,L	J	L	PHE4	PHE4 THR12
2ZCC:A,C	A	C	VAL70	VAL70
1P3Q:U,V	U	V	MET1	PHE4
2ZCC:C,B	C	B	ILE36	
3M3J:B,F	B	F	PHE45 SER65	PHE45
3M3J:B,E	B	E		
2XEW:C,I	C	I	PHE45 HIS68	PHE45
3NS8:A,B	A	B		TYR59 ARG42 GLN49
3NOB:A,C	A	C		ILE44
3H1U:A,B	A	B	ARG42 VAL70 GLN49 ILE44	LEU8 ILE36 LEU71 GLN40
3EEC:A,B	A	B	ILE44 VAL70 GLN49	LEU8 LEU71 ILE36
2XEW:B,E	B	E		
1TBE:A,B	A	B		PHE45 ARG42
2XEW:E,F	E	F		

Supplementary Table 2: The computational hot spot residues of ubiquitin – ubiquitin interfaces.

Interface	Main Chain	Partner Chain	Main Chain Hot spot residues	Partner Chain Hot spot residues
3M3J:C,D	C	D	ILE44 ARG42 VAL70 GLN49	VAL70 GLN49 ILE44
3OJ3:C,B	C	B		
2O6V:E,G	E	G		
2XEW:K,J	K	J	ILE13 GLN40 ILE36 LEU69 VAL70 LEU71	ILE36 VAL70 LEU71 LEU73 ILE13 GLN40
2PE9:A,B	A	B	HIS68 PHE45 THR66 ALA46	ILE36 GLU34 GLN41 LEU71 GLN40
2QHO:A,E	A	E		PHE45
3M3J:A,E	A	E	ILE36 LEU73 GLN40 LEU71	GLN40 LEU71 ILE36 LEU73
1F9J:A,B	A	B		TYR159
2C7N:J,L	J	L	PHE4	PHE4 THR12
2ZCC:A,C	A	C	VAL70	VAL70
1P3Q:U,V	U	V	MET1	PHE4
2ZCC:C,B	C	B	ILE36	
3M3J:B,F	B	F	PHE45 SER65	PHE45
3M3J:B,E	B	E		
2XEW:C,I	C	I	PHE45 HIS68	PHE45
3NS8:A,B	A	B		TYR59 ARG42 GLN49
3NOB:A,C	A	C		ILE44
3H1U:A,B	A	B	ARG42 VAL70 GLN49 ILE44	LEU8 ILE36 LEU71 GLN40
3EEC:A,B	A	B	ILE44 VAL70 GLN49	LEU8 LEU71 ILE36
2XEW:B,E	B	E		

1TBE:A,B	A	B		PHE45 ARG42
2XEW:E,F	E	F		

Appendix B

Supplementary Codes

Naccess Driver:

```
def naccessDriver(naccessRunPath, interfaceList, monomerStructureDirectory,
naccessResultFileDirectory):
        for each interface in interfaceList:
                run naccess for monomer_1
                run naccess for monomer_2
                run naccess for complex
                move all naccess results to the resultFileDirectory
end
```

Interface Extraction Using Naccess:

```
def interfaceExtractorUsingNaccess(interfaceList, deltaASACriteria, naccessResultFileDirectory,
naccessResultFileDirectory, naccessInterfaceFileDirectory):
        for each interface in interfaceList:
                monomer_1_ASA = naccessRSAFileReader(monomer_1, naccessResultFileDirectory)
                monomer_2_ASA = naccessRSAFileReader(monomer_2, naccessResultFileDirectory)
                interface_ASA = naccessRSAFileReader(interface, naccessResultFileDirectory)
                deltaASA = monomer_1_ASA+ monomer_2_ASA- interface_ASA
                if deltaASA > deltaASACriteria:
                        write interface coordinates to the naccessInterfaceFileDirectory
end
```

Interface Extraction Using VDW:

```
def interfaceExtractorUsingVDW(interfaceList, VDWCriteria, nearbyCriteria, pdbFileDirectory,
VDWInterfaceFileDirectory, VDWNearbyFileDirectory):
        interfaceResidueDictionary = {}
        for each interface in interfaceList:
                chain_1_atomList, chain_2_atomList = PDBFileReader(chain_1, chain_2,
pdbFileDirectory)
                for each atom_1 in chain_1_atomList:
                        for each atom_2 in chain_2_atomlist:
                                distanceAtom = dist(atom_1, atom_2)
                                distanceCriteria = atom_1 VDW radii + atom_2 VDW radii +
        VDWCriteria
                                if distanceAtom <= distanceCriteria:
                                        interfaceResidueDictionary[residue_1] = 1
                                        interfaceResidueDictionary[residue_2] = 1
                for each residue in interfaceResidueDictionary:
                        nearbyResiduesList.append(closest residues whose CA distance smaller than
                nearbyCriteria)
                        write residue coordinates to the VDWInterfaceFileDirectory

                for each residue in nearbyResiduesList:
```

```
                    write residue coordinates to the VDWNearbyFileDirectory
end
```

Multiprot Driver:

```
def multiprotDriver(interfaceList, multiprotRunPath, interfaceAndNearbyStructureDirectory,
multiprotSimilarityFileDirectory):
        for each interface_1 in interfaceList:
                for each interface_2 in interfaceList:
                        multiprotResults = run multiprot for interface_1 and interface_2
                        matchingResidues = readMultiprotOutput(multiprotResults)
                        similarityValue = matchingResidues / min(interface_1_size, interface_2_size)
                        write similarityValue to the multiprotSimilarityFileDirectory
end
```

Interface Network Builder:

```
def interfaceNetworkBuilder(interfaceList, interfaceSimilarities):
        G = undirectedGraph()
        for each interface in interfaceList:
                add interface as node to G
                for each similarityValue in interfaceSimilarities[interface]:
                        add similarityValue as edge between interface and its neighbor to G
        return G
end
```

Combine Interface Network With Similarity:

```
def combiningNetworkNodesWithSimilarity(interfaceNetwork, similarityCriteria):
        nodesToCombine = 1
        while nodesToCombine > 0:
                nodesToCombine = 0
                maxEdgeWeight = find maximum edge weight of the network
                if maxEdgeWeight >= similarityCriteria
                        nodesToCombine = nodesToCombine + 1

                if nodesToCombine > 0:
                        newNode = '%s_%s' %(interface_1, interface_2)
                        add newNode as node to interfaceNetwork
                        create edges between the newNode and both the interface_1 and interface_2
                        neighbors' nodes
                        ##################################################################
                        ## (If a neighbor has both edge between the interface_1 and interface_2,
                        ## choose the highest similarity value for the new edge between the newNode
                        ## and the neighbor node)
                        ##################################################################
                        delete both interface_1 and interface_2 nodes from interfaceNetwork
        return interfaceNetwork
end
```

Community Finder:

```
def communityFinder(interfaceNetwork, minClusterDivisionCriteria, communityList,
clusteringCoefficientCriteria):
        betweennessCentralityValues = edge_betweenness_centrality(interfaceNetwork)
        numberOfConnectedComponents = 1
        while numberOfConnectedComponents ==  1:
                remove the maximum edge value from the network
                connectedComponents = connected_component_subgraphs(interfaceNetwork)
                numberOfConnectedComponents = len(connectedComponents)
        clusteringCoefficient_0 = average_clustering(connectedComponents[0])
        clusteringCoefficient_1 = average_clustering(connectedComponents[1])
        if size(connectedComponents[0]) > minClusterDivisionCriteria and clusteringCoefficient_0 >=
clusteringCoefficientCriteria:
                communityList = communityFinder(connectedComponents[0],
minClusterDivisionCriteria, communityList)
        else:
                communityList.append(connectedComponents[0])
        if size(connectedComponents[1]) > minClusterDivisionCriteria and clusteringCoefficient_1 >=
clusteringCoefficientCriteria:
                communityList = communityFinder(connectedComponents[1],
minClusterDivisionCriteria, communityList)
        else:
                communityList.append(connectedComponents[1])
        return communityList
end
```

Silhouette Index Calculator:

```
def silhouetteIndexCalculator(clusterInfo, interfaceSimilarities):
        silhouetteDictionary = {}
        for each cluster in clusterInfo:
                for each interface_1 in cluster:
                        sumDissimilarityIntra = 0
                        for each interface_2 in cluster except interface_1:
                                sumDissimilarityIntra = sumDissimilarityIntra  + (1-
interfaceSimilarities(interface_1, interface_2))
                        averageDissimilarityIntra=
sumDissimilarityIntra/(numberOfMembersInCluster-1)
                minDissimilarityInter = 1-max(interface similarities between members in other
        clusters)
                silhouetteDictionary[interface_1] = (minDissimilarityInter – averageDissimilarityIntra)
/ max(minDissimilarityInter, averageDissimilarityIntra)
        return silhouetteDictionary
end
```

Type I and II Finder:

```
def type_I_II_finder(representativeProtein_1_domainDict, representativeProtein_2_domainDict,
memberProtein_1_domainDict, memberProtein_1_domainDict):
        similarDomainNumbers = find similar domain number of the member using representative
domains
        if similarDomainNumbers_protein_1 == representativeDomainNumbers_protein_1 and
similarDomainNumbers_protein_2 == representativeDomainNumbers_protein_2:
                if similarDomainNumber_protein_1 > 1 or  similarDomainNumber_protein_2 > 1:
                        memberType = 'Type_1_multidomain'
                else:
                        memberType = 'Type_1'
        else:
                if similarDomainNumber_protein_1 > 1 or  similarDomainNumber_protein_2 > 1:
                        memberType = 'Type_2_multidomain'
                else:
                        memberType = 'Type_2'
        return memberType
end
```

Type III Finder:

```
def type_III_finder(interfaceList, interfaceClusterInfo, monomerDomainInfo):
        type_3_list = []
        for each interface in interfaceList:
                interfaceClusterNo = interfaceClusterInfo[interface]
                for each interface_2 in interfaceList:
                        if interfaceClusterNo != interfaceClusterInfo[interface_2]:
                                if monomerDomainInfo[interface_monomer_1] ==
monomerDomainInfo[interface_2_monomer_1] or monomerDomainInfo[interface_monomer_1] ==
monomerDomainInfo[interface_2_monomer_2] or monomerDomainInfo[interface_monomer_2] ==
monomerDomainInfo[interface_2_monomer_1] or monomerDomainInfo[interface_monomer_2] ==
monomerDomainInfo[interface_2_monomer_2]:
                                        type_3_list.append(interface)
        return type_3_list
end
```