

Storing and analysing biomolecular contacts

Dissertation

zur Erlangung des Grades

des Doktors der Naturwissenschaften

der Naturwissenschaftlich-Technischen Fakultät III

Chemie, Pharmazie, Bio- und Werkstoffwissenschaften

der Universität des Saarlandes

von

Peter Walter

Saarbrücken,

Juni 2011

Tag des Kolloquiums: 16.01.2012

Dekan: Prof. Dr. Wilhelm F. Maier

Berichterstatter: Prof. Dr. Volkhard Helms  
Prof. Dr. Andreas Hildebrandt

Vorsitz: Prof. Dr. Rita Bernhardt

Akad. Mitarbeiter: Dr. Konstantin Schneider

## Acknowledgements

First and foremost, I would like to thank Prof. Volkhard Helms for offering me the opportunity to work in his group. I am also grateful to him for his constant support and encouragement. I also want to thank my former colleague Sam Ansari who gave me a lot of suggestions in the field of protein-protein interactions. I want to specially thank Christian Spaniol, Jennifer Metzger and Mazen Ahmad for reading my thesis and providing me with fruitful comments. I also thank my family for their moral support and their patience. Last but not least, I thank all the members in the Helms group for providing a pleasant working environment.

## Abstract

Biomolecular contacts play a crucial role in all areas of life. In particular, protein-protein (PP) interactions are essential for most processes in biological cells. Antigen-antibody recognition, enzyme substrate binding, hormone receptor binding, RNA splicing, DNA replication and signal transduction are just some examples for the rich variety of PP interactions. In the last years modern proteomic methods have helped to get a better understanding of the complexity within living cell and organism. More and more sequences of unknown proteins are deciphered, their function is revealed, structural details are detected and the interaction in the complex network of biological processes is uncovered. Contacts between proteins and small molecules (PL) describe the second important group for biomolecular contacts and play an essential role for drug design. The vast increase of such information necessitates the application of databases for easy handling and analysis of data. We created a database covering PP as well as PL interactions for which structural data are available. Using the database, we performed a number of analyses concerning features of protein-protein complexes, in particular the group of obligate and non-obligate interactions. Combining information from PP and PL complexes, we generated a prediction method for binding sites of small molecules on PP interface sites. Finally, we tested the applicability of features of PP interactions for the prediction of their kinetic parameters.

## Kurzfassung

Kontakte zwischen Biomolekülen spielen eine wichtige Rolle in allen Bereichen des lebenden Organismus. Insbesondere Protein-Protein Interaktionen (PP) sind für die meisten Prozesse innerhalb der Zelle erforderlich. Die Antigen-Antikörper Erkennung, Enzym-Substrat Bindung, Bindung eines Hormons an seinen Rezeptor, Spleißen von RNA, DNA Replikation und Signaltransduktion sind nur einige Beispiele für die große Vielfalt an PP Interaktionen. In den vergangenen Jahren haben moderne Methoden aus dem Bereich der Proteomik dazu beigetragen, das Verständnis der Komplexität innerhalb der lebenden Zelle und des Organismus zu verbessern. Die Zahl an entschlüsselten Proteinsequenzen steigt beständig an, ihre Funktion wird erfaßt, strukturelle Details werden aufgedeckt und ihr Beitrag im Netzwerk der biologischen Prozesse wird durchleuchtet. Kontakte zwischen Proteinen und kleinen Molekülen bzw. Liganden (PL) stellen die zweite wichtige Gruppe an biomolekularen Kontakten dar und spielen eine wesentliche Rolle für die Entwicklung neuer Arzneistoffe. Der immense Anstieg derartiger Informationen erfordert den Einsatz von Datenbanken zur einfachen Handhabung und Auswertung der Daten. Wir erstellten eine Datenbank, die sowohl PP als auch PL Interaktionen umfasst und für die strukturelle Informationen vorhanden sind. Unter Zuhilfenahme dieser Datenbank führten wir Analysen zu Eigenschaften von Protein-Protein Komplexen, insbesondere zur Gruppe der obligaten und nicht-obligaten Interaktionen, durch. Indem wir Informationen aus PP und PL Komplexen miteinander kombinierten, schufen wir eine Vorhersagemethode für Bindungsstellen von kleinen Molekülen auf PP Oberflächen. Schließlich untersuchten wir physiko-chemische Merkmale von PP Interaktionen zur Vorhersage ihrer kinetischen Parameter.

## Zusammenfassung

Kontakte zwischen Biomolekülen spielen eine wichtige Rolle in allen Bereichen des lebenden Organismus. Methoden wie z.B. die Kristallographie ermöglichen eine genaue Strukturaufklärung auf molekularer Ebene. Trotz der Schwierigkeiten, biologisch relevante Moleküle zu detektieren, wächst die Zahl an verfügbaren Strukturdaten stetig an. Kontakte zwischen Proteinen untereinander gelten als besonders interessanter Forschungsansatz wegen ihrer Bedeutung in den meisten biologischen Prozessen wie z.B. im Metabolismus oder in der Signaltransduktion. In diesem Zusammenhang werden beispielsweise Antworten auf die Frage gesucht, welche Charakteristika die Bindungsstellen an der Oberfläche von Proteinen aufweisen im Vergleich zu Nichtbindungsstellen oder ob eine Interaktionsstelle mit Hilfe von physikochemischen Parametern vorhergesagt werden kann. Desweiteren ist gerade für die pharmazeutische Forschung von Interesse, inwieweit die Kenntnisse über Protein-Protein Interaktionen zur Entwicklung neuer Arzneistoffe genutzt werden können. Dabei kommt den Interaktionen zwischen Proteinen und kleinen Molekülen eine besondere Bedeutung zu, da letztere zur wichtigsten Gruppe an pharmazeutisch relevanten Verbindungen gehören.

Aufgrund der Menge an verfügbaren Daten und in Hinblick auf den ständigen Anstieg von neuen Informationen bietet sich der Einsatz von Datenbanktechnologien an. Wir erstellten im Rahmen der Dissertation eine verbesserte Version der Analysing Biomolecular Contacts Datenbank (ABC<sup>2</sup>), die eine Vielzahl von Merkmalen von PP und PL Interaktionsdaten aufnehmen und abspeichern kann. Dazu gehören beispielsweise die Interface SASA, geometrische Merkmale des Interface wie die Planarität oder die Rundheit und Daten über die interagierenden Aminosäuren und Atome. Zwei verschiedene Definitionen eines Interface kommen hierbei zur Anwendung. Im distanzbasierten Ansatz werden Residuen oder Atome betrachtet, die innerhalb eines bestimmten Abstands zu Residuen oder Atomen des jeweiligen Bindungspartner liegen. Alternativ kann das oberflächenbasierte Kriterium angewendet werden, das all diejenigen Aminosäuren oder Residuen als zum Interface gehörig ansieht, wenn ihre Oberfläche im Verlauf der Komplexierung abnimmt, also vom Partnermolekül überlagert wird. Gegenwärtig beträgt die Grösse der Datenbank ca. 40 Gigabyte. Eine Webseite ist verfügbar, die Nutzern die Suche nach bestimmten PP oder PL Interaktionen ermöglicht und verschiedene Informationen bereitstellt.

Der Datenbestand diente als Grundlage für Analysen von PP und PL Interaktionen. PP Komplexe können aufgrund ihrer spezifischen Eigenschaften und Funktionen klassifiziert werden. Als Beispiel ist die Gruppierung nach obligaten und nicht-obligaten Interaktionen zu nennen wobei erstere PP Komplexe bezeichnen, deren Proteinketten nur im komplexierten Zustand stabil sind und letztere sich auf Komplexe beziehen, deren Bestandteile auch getrennt voneinander existieren können. Wir untersuchten Zusammenhänge zwischen der Struktur und Gruppierung von PP Interaktionen und ihrer Funktion. Eine Beobachtung hierbei war, dass Proteinketten von obligaten Interaktionen eher dazu tendieren, zum selben Kompartiment zu gehören als nicht-obligate Interaktionen.

Eine weitere Analyse bezog sich auf den Zusammenhang zwischen PP und PL Interaktionen. Speziell interessierten wir uns für PL Interaktionen, für die analoge PP Interaktionen existierten, d.h. die Interfaces der PP und PL -

Komplexe weisen zumindest eine teilweise Überlappung auf. Wir konnten einen Datensatz von 175 nicht-redundanten PP/PL Paaren generieren, den wir als Trainingssatz für eine Vorhersage von Bindungsstellen von kleinen Molekülen auf PP Interfaces verwendeten. Als Vorhersagemethode kamen hierbei random forests zum Einsatz.

Schließlich wurden Interaktionsmerkmale der ABC<sup>2</sup>-Datenbank dahingehend untersucht, mit welcher Genauigkeit Kinetiken zwischen Protein-Protein Interaktionen vorhergesagt werden können. Dazu wurde ein aus der Literatur gewonnener Datensatz aus PP Komplexen herangezogen, der kinetische Parameter wie  $k_{on}$  oder  $k_{off}$  beinhaltet. Dieser Datensatz wurde nach verschiedenen Kriterien gefiltert. Die resultierenden Komplexe dienten als Trainingssatz für eine support vector machine, wobei als Merkmale z.B. die Zahl der polaren Aminosäuren oder die Interface SASA angewendet wurden.

## List of Figures

1	Three dimensional structure of retinole molecule bound to a protein	13
2	Overview of bioactive small molecules. . . . .	14
3	Overview of interface criteria between protein-protein interactions.	16
4	Protein-protein interaction . . . . .	17
5	Oligomeric states of PP complexes. . . . .	18
6	Interface similarity calculation using fingerprints I . . . . .	21
7	X-Ray detection of molecules . . . . .	23
8	Number of innovative drugs approved by the FDA . . . . .	28
9	Docking workflow. . . . .	30
10	Non-covalent interaction in protein-ligand interactions . . . . .	31
11	Growth of RCSB database . . . . .	34
12	Architecture of a DBMS. . . . .	35
13	Relationships between relations. . . . .	37
14	Types of ER symbols . . . . .	38
15	Partial and transitive dependencies. Figure adapted from [1] . . . . .	43
16	Concatenated key . . . . .	45
17	Indexing methods. . . . .	46
18	ABC <sup>2</sup> : Basic relations. . . . .	51
19	ABC <sup>2</sup> : Connection between basic relations and RCSB identifiers.	52
20	ABC <sup>2</sup> : <i>abc_entity</i> and associated relations . . . . .	53
21	ABC <sup>2</sup> : <i>abc_entity</i> and statistics relations . . . . .	53
22	Fragmentation of SMILE strings. . . . .	55
23	Interface similarity calculation using fingerprints I . . . . .	55
24	ABC <sup>2</sup> : Distance criterion . . . . .	57
25	ABC <sup>2</sup> : Surface based criterion . . . . .	58
26	ABC <sup>2</sup> : <i>AAIndex</i> . . . . .	59
27	ABC <sup>2</sup> : Pharmacophores . . . . .	59
28	ABC <sup>2</sup> : <i>bioUnit</i> . . . . .	61
29	ABC <sup>2</sup> : amino acid chain representation. . . . .	62
30	ABC <sup>2</sup> : GO . . . . .	62
31	Overview of class packages. . . . .	65
32	Basic classes for the import process. . . . .	66
33	PDB representation for import process. . . . .	68
34	Object hierarchy for import process. . . . .	68
35	Object hierarchy for basic import workflow. . . . .	69
36	Object hierarchy for extended import workflow. . . . .	69
37	Object hierarchy for deleting data. . . . .	70
38	ABC <sup>2</sup> : User management . . . . .	71
39	Model View Controller (MVC) framework . . . . .	71
40	ABC <sup>2</sup> -website . . . . .	73
41	ABC <sup>2</sup> -website . . . . .	74
42	GO-tree . . . . .	77
43	Combination of protein chains from complexes . . . . .	79
44	Distribution of the GO similarity according to molecular function	81
45	Distribution of the GO similarities according to biological process	82
46	Distribution of the GO similarities according to cellular component	82
47	C-index values . . . . .	83
48	Unique clusters . . . . .	84

49	Features for clusters with 60% functional similarity. . . . .	87
50	Hydrophobicity of interface residues . . . . .	87
51	Distribution of sidechain and backbone contacts . . . . .	88
52	Schema of a pair of shared PP and PL interaction. . . . .	94
53	Distribution of pharmacophore groups. . . . .	95
54	Example of a PP/PL pair. . . . .	97
55	PP residues against PL residues. . . . .	98
56	Conservation of PP/PL pairs. . . . .	99
57	Protrusion of PP/PL pairs. . . . .	99
58	Distribution of surface fractions for overlap and non-overlap residues for PP interfaces. . . . .	100
59	Feature quality for random forest prediction. . . . .	101
60	Maximum overlap surface patches. . . . .	102
61	Number of surface patches per interface. . . . .	103
62	Collection and filtering of PPI and PLI. . . . .	109
63	Superposition of a PP/PL pair . . . . .	110
64	Pharmacophore assignment for three amino acids . . . . .	112
65	Surface patch generation. . . . .	113
66	Surface patch generation. . . . .	113
67	Association constants range. . . . .	116
68	The transition state for association . . . . .	117
69	Correlation between charge imbalance and association kinetics . . . . .	125
70	Plot between dissociation rate constant and mass imbalance. . . . .	126
71	Plot between inhibition constant and gap volume index. . . . .	126
72	Correlation among kinetic values. . . . .	127
73	Web page for prediction of kinetic values. . . . .	128
74	Maximum overlap patches with patch sizes 5,6 and 8. . . . .	130
75	ABC <sup>2</sup> database diagram, part one. . . . .	133
76	ABC <sup>2</sup> database diagram, part two. . . . .	134
77	Features for clusters with 60% functional similarity. . . . .	135

## List of Tables

1	Overview of experimental methods to detect protein-protein interactions . . . . .	24
2	Databases related to protein-protein and protein-small molecule interactions . . . . .	32
3	Overview of MySQL commands . . . . .	47
4	Data sources for ABC <sup>2</sup> . . . . .	50
5	Largest clusters for molecular function with similarity greater than 80% . . . . .	85
6	Set of complexes for NOXclass prediction . . . . .	89
7	Cluster bins for NOXclass prediction. . . . .	89
8	Examples of PP/PL pairs. . . . .	96
9	Confusion matrix for random forest prediction. . . . .	100
10	Term frequencies for GO function. . . . .	103
11	Term frequencies for GO biological process. . . . .	104
12	Term frequencies for GO cellular component. . . . .	104
13	Biological process terms referring to apoptosis. . . . .	105
14	PDBs related to apoptosis. . . . .	106
15	Visualization of complexes related to apoptosis. . . . .	107
16	Overview of pharmacophores. . . . .	111
17	Feature data on association and dissociation kinetics. . . . .	118
18	Feature data on inhibition kinetics. . . . .	120
19	List of features for prediction of PP-kinetics . . . . .	121
20	Grouping of amino acids. . . . .	121
21	Prediction accuracies for kinetic values. . . . .	124
22	List of PL/PP pairs. . . . .	132

## Listings

1	SQL query for getting surface size of an interface. . . . .	63
2	SQL query for accessing contact data based on distance criterion. . . . .	63
3	SQL query for finding all PDBs containing protein chains with uniprot-ID P00044. . . . .	63
4	SQL command defining a view. . . . .	64
5	Code example for a workflow class . . . . .	66

# Contents

<b>1</b>	<b>Introduction</b>	<b>13</b>
1.1	Biomolecular contacts . . . . .	13
1.2	Protein-protein interactions . . . . .	14
1.3	Types of protein-protein interactions . . . . .	16
1.4	Characteristics of protein-protein interactions . . . . .	18
1.5	Interface similarity . . . . .	20
1.6	Detection of protein-protein interactions . . . . .	22
1.7	Prediction methods . . . . .	24
1.7.1	Prediction of protein-protein interaction . . . . .	25
1.7.2	Prediction of protein-protein interfaces . . . . .	26
1.7.3	Prediction of protein-protein interface structure . . . . .	27
1.7.4	Function prediction . . . . .	27
1.8	Protein-small molecule interactions . . . . .	28
1.9	Protein-protein and protein-small molecule interaction databases	31
1.10	Aim of this work . . . . .	33
<b>2</b>	<b>Database management systems</b>	<b>34</b>
2.1	Overview . . . . .	34
2.2	Relational model . . . . .	36
2.2.1	Keys . . . . .	36
2.2.2	Entity relationship (ER) . . . . .	37
2.2.3	Relational algebra . . . . .	39
2.3	Design principles . . . . .	41
2.3.1	Anomalies . . . . .	41
2.3.2	Normalization . . . . .	42
2.3.3	Indices . . . . .	44
2.4	SQL . . . . .	47
<b>3</b>	<b>The ABC<sup>2</sup> database</b>	<b>49</b>
3.1	Preliminary considerations . . . . .	49
3.2	Structure of ABC <sup>2</sup> database . . . . .	50
3.3	Examples of SQL queries . . . . .	61
3.4	Implementation . . . . .	64
3.4.1	Data import . . . . .	64
3.4.2	Deleting and updating data . . . . .	67
3.5	Website . . . . .	70
<b>4</b>	<b>Function-structure relationships in obligate or non-obligate protein-protein interactions</b>	<b>75</b>
4.1	Data generation . . . . .	76
4.2	Assignment of GO-terms . . . . .	77
4.3	Clustering . . . . .	79
4.4	GO-term analysis for obligate and non-obligate interfaces . . . . .	80
4.5	Functional clustering of complexes . . . . .	81
4.6	Conclusions . . . . .	90

<b>5</b>	<b>Predicting where small molecules bind at protein-protein interfaces</b>	<b>92</b>
5.1	Results and Discussion . . . . .	93
5.2	Materials and methods . . . . .	106
5.2.1	Pharmacophore group assignment . . . . .	108
5.2.2	Feature generation . . . . .	111
5.2.3	Random forests . . . . .	112
<b>6</b>	<b>Prediction of kinetics of protein-protein interactions</b>	<b>115</b>
6.1	Methods . . . . .	117
6.1.1	Features . . . . .	120
6.1.2	Classification . . . . .	123
6.1.3	Model validation: . . . . .	123
6.2	Results and Discussion . . . . .	124
6.2.1	Conclusions . . . . .	128
6.3	Website . . . . .	128
<b>7</b>	<b>Outlook</b>	<b>129</b>
<b>8</b>	<b>Supplementary material</b>	<b>130</b>
	<b>References</b>	<b>136</b>

# 1 Introduction

## 1.1 Biomolecular contacts

Biomolecular contacts are structurally defined through their binding interfaces that are the contact area between two biomolecules that are non-covalently linked. Such an interaction has a biological meaning influencing a living cell or an organism. In particular, this work refers to protein-protein and protein-ligand interactions which belong to the most important group of biological contacts and offer numerous opportunities for therapeutic applications.

The basic principle of binding among biomolecules is the “Lock and Key Model” introduced by Emil Fischer in 1890 [2]. An example for this is shown in figure 1. The fulfilment of a biological function is achieved by the binding of two structures, the so called ligand and host, which are complementary among each other at the binding interface. A refinement of this principle is the induced fit model which has been favoured for more than 50 years [2]. Here, the interfacial areas of the binding partners do not have to be fully complementary. Upon complexation conformational changes take place optimizing the interaction. A further model has been suggested which is called ‘conformational selection’ and formulates several conformational ensembles of a protein in its unbound state. A binding partner selects a state for complexation which results in an energetically favourable complex [3].

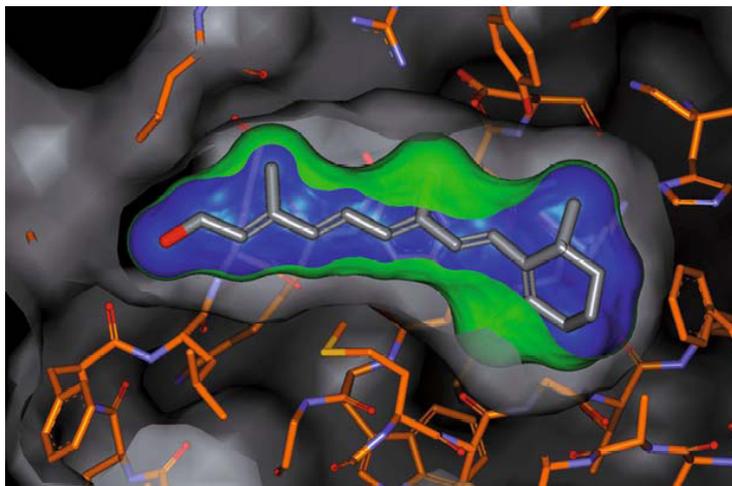


Figure 1: A retinole molecule (grey sticks) is embedded like a key within the binding site of its transport protein. Picture taken from [4].

Protein-protein interactions play a crucial role in all areas of life. In particular, they are essential for most processes in biological cells. Antigen-antibody recognition, enzyme substrate binding, hormone receptor binding, RNA splicing, DNA replication, transcription and signaling pathways just represent some examples of the processes in which protein-protein interactions are involved [5]. In the last years modern proteomic methods have helped to get a better understanding of the complexity within living cells and organisms. More and more sequences of unknown proteins are deciphered, their functions are revealed,

structural details are detected and their interaction in the complex network of biological processes are uncovered.

Despite of the growing number of experimentally determined three-dimensional structures of protein complexes, our understanding of the physico-chemical properties governing binding specificity and affinity is still limited. Elucidating the underlying principles of protein-protein interactions may contribute to advance and finally complete our understanding of the relationships between characteristics of protein-protein interactions and their function. Such a knowledge can be applied for predicting putative protein-protein contacts, including their structure and function. Moreover, due to their importance for the living cell, protein-protein interactions are attractive targets for novel drug therapies. One application is the mimicking of protein chains serving as binding partners in a complex by small ligands which also represents the link to the second group of biomolecular contacts, the so-called protein-small molecule interactions. Small molecules are found as natural substrates in many biological processes making them also attractive for drug design. Some typical examples for small molecules are shown in figure 2.

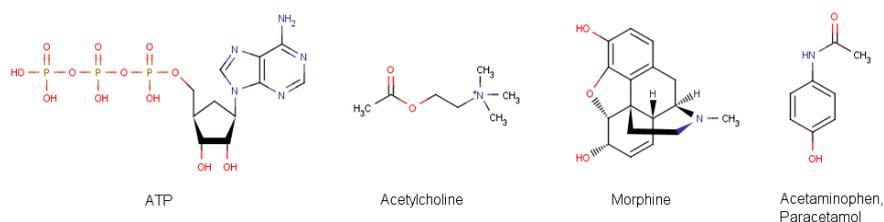


Figure 2: Overview of bioactive small molecules. ATP is a natural substrate for many biological processes, acetylcholine acts as neurotransmitter in organism, morphine belongs to the first synthesized drugs and was isolated from opium, paracetamol is a widely used drug belonging to the group of non-steroidal anti-inflammatory drugs (NSAD).

There is no clear definition for small molecules, even in terms of size. Usually, they cover a wide range of organic compounds and do not incorporate single atoms or ions. However, molecules that might be pharmaceutically relevant can be characterized by Lipinski's rule of five, defining some constraints with respect to physico-chemical properties [6]. In comparison to proteins, small molecules have certain advantages over proteins with respect to therapeutical applicability. The relatively large size of proteins makes oral administration difficult as they cannot be transported through the intestine membrane in general. Besides, proteins are decomposed by peptidases before reaching its target. From an economical point of view small molecules can be produced more easily and cheaper in many cases.

## 1.2 Protein-protein interactions

Protein-protein interactions turn out to be a valuable basis for many research areas. As a starting point for further considerations it is inevitable to give a clear definition of protein-protein interaction. Basically, one may distinguish between interactions within the same molecule (intra chain) and interactions between

separate protein molecules (inter chain). The former type is responsible for the stabilization of the monomer whereas the latter may be further subdivided into two different categories. A specific interaction exerts an effect which has a certain biological meaning within the living organism. Unspecific interactions have no functional meaning and occur *in vivo* due to the high concentration of macromolecules in the cell which is also denoted as macromolecular crowding [7]. *In vitro*, such associations may be observed when elucidating the structure of protein complexes by crystallography. The entire crystal is made of unit cells which are identical building blocks. A unit cell consists of identical asymmetric units that are grouped together by symmetric operations. Here, the asymmetric unit is the smallest unit which cannot be decomposed into subunits by applying symmetry operations. An asymmetric unit may contain one or several molecules but they do not necessarily represent a biological complex. A typical example is the hemoglobin molecule. The asymmetric unit consists of four identical chains but the actual biological complex is made of 8 chains which can be created by 2-fold symmetry operation of the asymmetric unit. This example shows that the existence of symmetry between molecules or groups of molecules does not indicate a crystal lattice but may represent a genuine biological molecule. These interactions are formed during crystallization and are denoted as crystal contacts.

For studies about protein-protein interactions from structural data it is inevitable to clearly distinguish between biologically relevant interactions and crystal contacts. Several investigations showed significant differences between these interaction types. It was found that the surface of biological meaningful interaction is much larger [8, 9]. Crystal contacts have a higher segmentation of the part of the protein participating in an interaction [10]. The conservation of residues at functional interfaces is higher in comparison to crystal contacts [11]. Conversely, some similarities were observed such as the amino acid composition of large crystal contacts and non-obligate protein-protein interfaces. Studying crystal contacts may be helpful for a better understanding of the crystallization process of proteins and may therefore provide hints for optimizing the procedure.

One focus of this work lies in the consideration of biologically relevant protein-protein interactions. Such an interaction is characterized by its interface which is defined as the area between two protein chains fulfilling a certain distance criterion or a surface-based criterion. The former is based on the distance of non-covalently connected atoms between residues. Every residue having one or more atoms whose distance with atoms from a residue from the partner chain lies within a certain range (4-8 Å) is considered as an interaction. The entirety of these residues finally makes up the interface of a protein complex. In the surface-based criterion the SASA of every residue in the bound and unbound states are measured. Only residues with a SASA value greater than zero are considered. If the absolute difference between complexed and uncomplexed form is greater than zero as well, then the residue is fully buried (i.e.  $\Delta(SASA_{bound}, SASA_{unbound}) = 0$ ) or partly buried (i.e.  $\Delta(SASA_{bound}, SASA_{unbound}) > 0$ ) upon complexation. In this case the residue is counted as interface residue. Both interface definitions are illustrated in figure 3.

Not all interactions are formed between proteins of the same organism. One example is the immune system, in particular antibody-antigen reactions. Figure 4 shows the binding of a major histocompatibility complex which is responsible

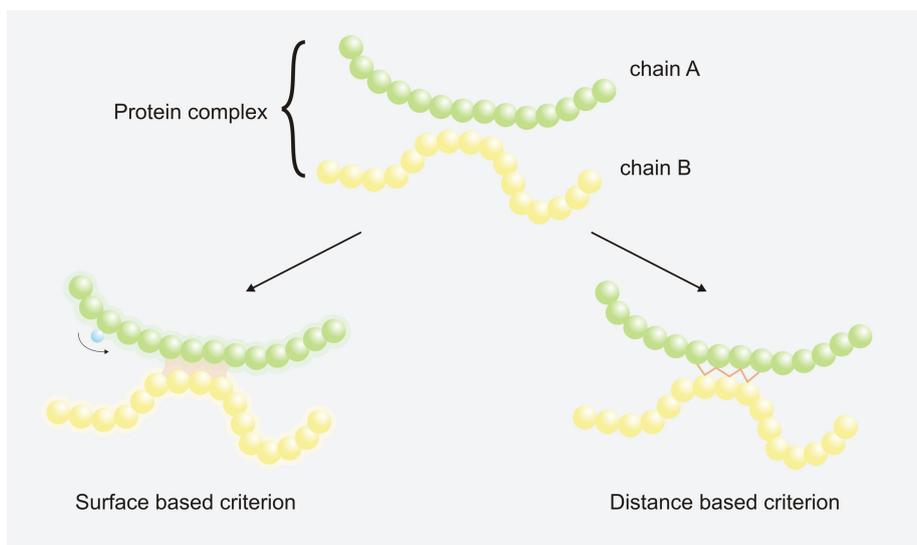


Figure 3: Overview of interface criteria between protein-protein interactions. For surface-based criterion, the sphere at the interface site indicates the volume which is not solvent-accessible any more upon complexation. The lines at the interface site for the distance-based criterion stand for contact distances between interface residues from both chains.

for recognition of exogenous compounds in favor of the immune system. The MHC II molecule consists of an  $\alpha$  and  $\beta$  chain which form a groove for the binding of antigens. Subsequently, the complex is recognized by T-helper cells activating the immune process. In the example the binding partner is a peptide which is derived from the herbal protein gluten. The antigen-antibody reaction is the reason for celiac disease which is an autoimmune disorder of the small intestine and leads to symptoms like diarrhoea, failure to thrive and also fatigue.

### 1.3 Types of protein-protein interactions

Protein-protein interactions play a pivotal role in most biological processes. In eucaryotic cells, the majority of proteins is involved in complex formation and it is believed that a protein has about six to eight interacting partners on average [12]. The vast number of different protein-protein interactions leads to an enormous variety of functions and effects in the cell. One of the most fundamental questions is whether there is a relationship between the characteristics of a protein-protein interaction and its function. Basically, in many biological processes, proteins recognize specific targets and bind to them in a highly regular manner. This observation leads to the assumption that specificity of interactions is determined by structural and physico-chemical features. One strategy to reveal these relations is to group interactions according to various aspects.

Several suggestions have been made in the literature for the classification of protein-protein interactions. In general, one can distinguish between three fundamental classes of protein-protein interactions [13]. The simplest is the distinction between homocomplexes and heterocomplexes. Usually, homocom-

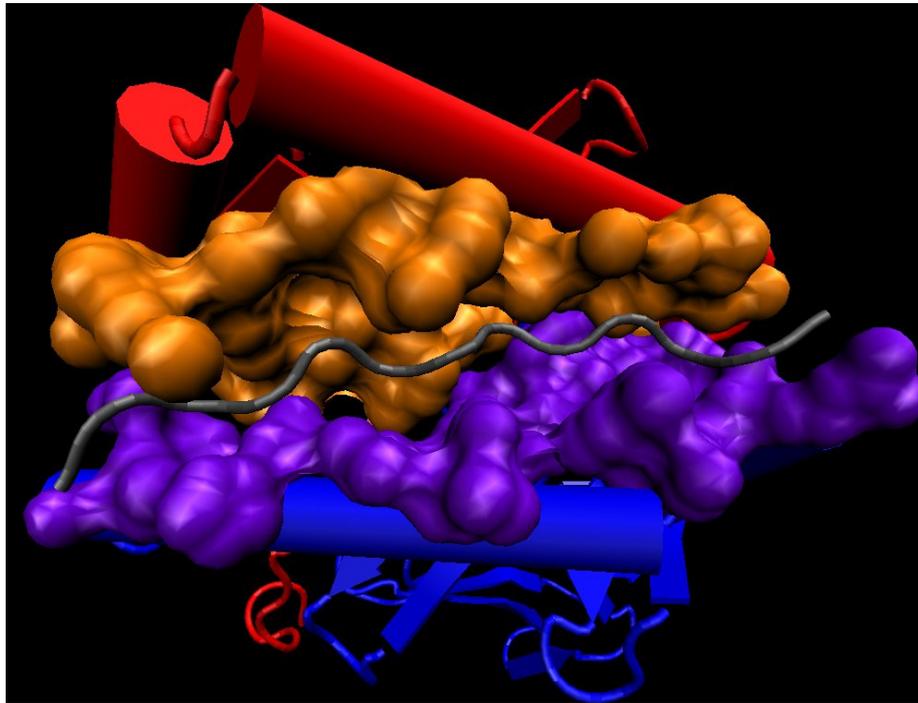


Figure 4: Protein protein interaction. An MHC molecule (red and blue) presents an antigen from gluten (grey). The interface surfaces of both chains are marked in orange and purple, respectively.

plexes are found in stable complexes and serve the purpose to form the structure of a protein complex whereas in heterocomplexes the binding of the proteins leads to a certain effect such as a signalling process. The binding partners can detach from each other allowing the regulation of the effect [14]. Next, the group of permanent and transient interactions refer to the lifetime of the complex. Permanent complexes are more stable in comparison to transient complexes and are existent for a longer time period. However, there is no clear threshold to separate both groups such as a particular kinetic or thermodynamic value. Besides, it may happen that an interaction switches from transient to permanent due to changes of cellular conditions [13]. This grouping can only be considered as a rough characterization of protein-protein interactions. On the contrary, an unambiguous classification can be made into obligate and non-obligate interactions which refer to the mere existence of binding partners in the bound and the unbound form. An obligate interaction is only stable in the bound form and leads to the constraint that the binding partners have to be co-localized upon expression.

Furthermore, one can also differ between protein complexes that form a complex to fulfill a biological reaction such as an enzymatic process and the ones whose interaction is responsible for stabilizing or forming a structure. An example for this is ceratin, a fibrous structural protein which can be found in the hair and nails of mammals. A newer approach of classification consists in incorporating structural information. It was investigated whether structurally

similar interfaces also tend to share the same function. Aloy *et al.* found out that similarity only in fold is only rarely associated with a similarity in interaction whereas close homologs also exhibit the same function [15]. Furthermore, Keskin *et al.* extended this paradigm to interfaces. They found complexes having similar structure at the interface but different functions [16].

In the literature, protein-protein interactions were investigated by many authors to find out whether the groups differ from each other with respect to structural or physico-chemical features. In the following, we list and discuss several important papers in this field: Ofran and Rost introduced six types of protein-protein interactions [17] which are based on the types mentioned above. Intra-domain and domain-domain contacts refer to interactions within the same chain taking place between residues of the same structural domain and between different domains respectively. The remaining types describe interactions between different protein chains. Permanent and transient contacts between equal chains are denoted as homo-oligomer and homo-complex. Analogously, contacts between different chains are called hetero-oligomer and hetero-complex. It was found that these types significantly differ from each other with respect to amino acid composition.

Another classification refers to a superior organisation of protein complexes. A protein may not only form a single interface with a partner structure but may also create several interface with different partners, leading to oligomeric interfaces [18]. The average number of members for an oligomeric complex has been estimated to be four [19]. Figure 5 shows different kinds of oligomeric states for proteins.

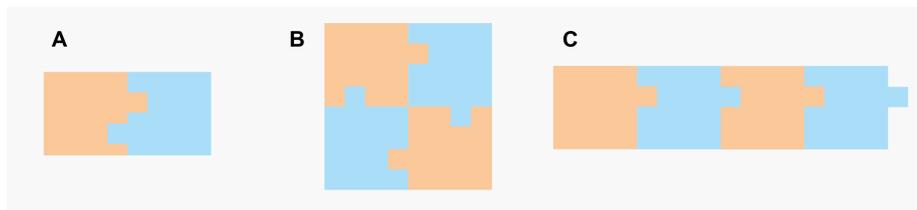


Figure 5: Oligomeric states of PP complexes. (A) Isologous dimer, (B) heterologous tetramer, (C) heterologous polymer. Figure adapted from [18].

#### 1.4 Characteristics of protein-protein interactions

As mentioned before, a number of research projects have dealt with elucidating characteristics about protein-protein interactions. Such information can then be applied for instance to find features that are suitable to distinguish interface areas from non-interface regions on the protein surface. Examples for such an approach are introduced in section 1.7.2.

Bahadur *et al.* compared obligate and non-obligate interfaces and observed several features allowing a discrimination between these types of interaction [20]. Obligatory chains have a higher number of contacts per interface than non-obligatory chains. Besides, the hydrophobicity of obligatory chains turns out to be much higher in comparison to non-obligatory interfaces, especially when focussing on amino acids located at the center of the interfaces (i.e. residues

that are fully buried). Comparing the amino acid preference revealed that polar contacts are preferred by non-obligate interactions whereas non-polar contacts are found in obligate interactions. This observation can be explained by the fact that obligate interactions are never exposed to the solvent in contrast to non-obligate complexes. Also, the residue-residue interaction patterns showed several dissimilarities. In general, obligate interfaces prefer interactions between non-polar interactions whereas the ratio between non-polar and polar interactions is more balanced for non-obligate complexes.

Several studies showed that obligate interactions tend to be homomeric whereas non-obligate interactions can be homomeric as well as heteromeric. A simple explanation can be made on the basis of the spatial constraints of obligate interactions. Due to the fact that the binding partners cannot exist in the unbound state, the protein chains have to be co-localized which means they have to be expressed in the same cellular compartment. Thus, binding of copies of the same protein chain is the easiest way to follow this principle. Intuitively, one may assume that obligate interactions are permanent, whereas non-obligate interactions may be transient or permanent. Even this is true in most of the cases but there is one exception to that rule. Antigen-antibody interactions are non-obligate and transient. The antigen originally exists separated from the antibody, but the interaction is usually stable after complexation has taken place.

Ansari and Helms studied the participation of sidechain and backbone interactions for a number of predominantly transient protein-protein interactions and showed that sidechain-sidechain and backbone-sidechain interactions dominate the protein interfaces [21].

An interesting aspect of protein-protein interaction was revealed in the study of Borgner *et al.* [22]. They compared the binding energies for a dataset of residues from heterodimeric protein-protein interfaces with each other using alanine scanning. It was found that not every interface residue plays an important role for the stability of a protein-protein interface. In fact, only a fraction of residues called *hot-spots* account for the majority of binding energy [22, 23]. In particular, tryptophan, arginine and lysine play important roles as energetically favorable residues. Despite the fact that these three amino acids belong to the largest ones no correlation between binding energy and size of buried sidechain surface was observed.

Hot spot residues are generally excluded from solvent and are located in the interior of the interface with a high packing density. Besides, hot spots tend to be highly conserved and are surrounded by residues exhibiting a more moderate conservation. Knowing about hot spots provides further insight into the nature of protein-protein interactions. However, binding data is available only for a small amount of complexes. Therefore, several methods for prediction of hot spots from structural information were developed using energy-based calculations [24, 25, 26]. Also MD-simulations were successfully applied for prediction, because hot spot residues tend to be restricted with respect to mobility [27]. All these methods however can not be applied to large scale analysis due to the computational costs. A more time-saving approach was taken by Tuncbag *et al.* [28]. They applied several empirical prediction methods using conservation, surface area and pair potentials for prediction of hot spot residues with an accuracy of about 70 percent.

Cohen *et al.* pointed out the differences and similarities between intermolec-

ular and intramolecular interactions of proteins [29]. To this end, four high resolution descriptors (H-bond contacts, interactions between polar atoms,  $\pi$ - $\pi$  and cation- $\pi$  interactions) were analyzed for both types of interaction. They found out that no difference exists in the chemistry or geometry of individual bonds leading to the suggestion that it might be feasible to incorporate intramolecular interactions for the overall analysis of protein-protein interactions.

Examining structural data of protein-protein interactions does not take into consideration flexibility of molecules. Not only that the observed complex corresponds the genuine structure *in vivo* but also for understanding the biological function one has to consider various conformations of the given protein. The function of a protein and its properties are also decided by the distributions and redistributions of the populations of its conformational and dynamic substates under different environments [30].

The role of water in interfaces is discussed in several works. In general, water plays a major role in polar interactions that stabilize complexes [31]. Janin distinguishes between dry interfaces, where water is only located in a ring around the interface area and wet interfaces containing cavities in the interior part of the interface that are filled with water molecules [31]. Thus, water increases atomic packing density in particular for wet interfaces. Teyra and Pisabarro examined the role of so-called wet-spots referring to residues that only participate in an interaction through water-mediation [32]. The existence of water molecules bridging residue interactions reduces the necessity for large shape complementarity in interface sites. Water also contributes significantly to the formation of protein-protein complexes [33, 34].

Protein disorder, which refers to the absence of stable conformations under physiological conditions, has an influence on the formation and characteristics of protein-protein interactions. It was found that the per-residues size and the surface areas of ordered proteins are significantly smaller than that of disordered proteins [35]. Disordered proteins feature the existence of several conformations which affects association of binding partners, modulates the lifetime of different conformers and influences the biological function [36].

Kinetic data complements understanding of the nature of protein-protein complexes and binding processes. Structural data alone just provide a static view of the complex whereas the kinetic nature of a protein-protein complex reveal its behaviour over time. For instance, the speed of association plays an important role in many cellular processes requiring a quick response to a stimulus such as signal transduction and immune response [37]. In such a case, rapid association or kinetic control prevails over thermodynamic control which refers to the stability of the complex [38]. Examples for kinetic control are the competition of proteins with the same receptor or different binding rates for related proteins. The experimental determination of binding constants of protein-protein interactions is a difficult task requiring tremendous efforts for preparation and performance. Consequently, only few data about kinetics for this kind of interaction is available.

## 1.5 Interface similarity

The comparison of protein-protein interfaces requires the definition of a similarity or dissimilarity scale. One may consider interfaces as similar if the protein chains from one complex are homologous to the protein complex from the part-

ner complex. However, it may happen that a complex contains more than a single interface which are distinct according to their function. Also, alignment of the interface region does not constitute a feasible method for similarity measurement. Depending upon the applied interface criterion, the interface area is subdivided into segments that are interrupted by non-interface stretches making the application of a common alignment procedure impossible. A number of research works suggested the following methods.

With increasing knowledge of structural details of protein complexes, methods were created on the basis of structural matches. Keskin *et al.* compiled a set of non-redundant protein-protein interfaces by structural comparison using geometric hashing technique [16]. This technique is applied in *Multiprot* that aims at finding similar protein structures by finding an optimal superimposition of the  $C_{\alpha}$  atoms of the query molecules [39]. The interface similarity is then calculated as the RMSD of the superimposed  $C_{\alpha}$  atoms from the query molecules.

Galinter explicitly compares the geometry of non-covalent interactions which are represented as vectors. An entire interface is modeled as graph with nodes consisting of vectors and edges revealing the geometric orientation between nodes. An alignment between two graphs, i.e. two protein-protein interfaces is performed by searching for the maximum common subgraph [40].

A combined method to derive interface similarity uses three different that are based on one sequence feature and two geometric features [41]. At first, an interface is considered as the combination of two faces, i.e. the regions on the protein chains that interact with each other. For calculating the similarity between two faces, their sequence is aligned with each other. Then the alignment is then transferred into a fingerprint representation. Every match of interface residues between the two faces is assigned one in the fingerprint whereas every mismatch is assigned a zero. The similarity calculation is depicted in figure 6

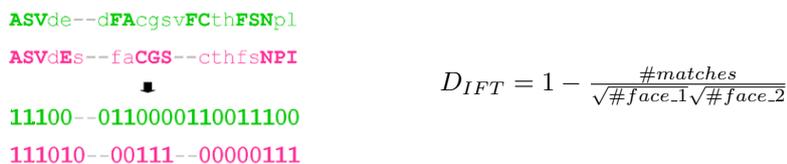


Figure 6: Left: Interface tag alignment and mapping between two faces (green and magenta) into fingerprint representation. Uppercase letters in the alignment refers to amino acids that belong to the interface, lowercase letters are located outside the interface region. Right: Formula for calculating the distance between two faces.

The two geometric features are calculated after structural alignment of the faces. The face overlap distance is based on superposition as described above whereas the face angle stands for the angle between the centres of the two faces and the common center after superimposition.

Another approach is described by Mintseris and Weng [42]. They considered an interface as pairs of atoms from different chains that are in close contact with each other. To this end, a pre-defined set of 18 atom types was applied leading to 171 atoms types altogether [43]. This number represents the dimension of the so-called *atomic contact vector* (ACV). Every element of the vector contains

the number of occurrences of a certain atom type pair. For two interfaces the similarity was then determined by calculating the Euclidian distance between their vector representation. The ACVs were used for statistical analysis and prediction methods. As an application, it was possible to distinguish a dataset of transient and permanent complexes with a success rate of 91%.

## 1.6 Detection of protein-protein interactions

Consideration of protein-protein interactions requires interaction data as a basis for further analysis. Such data are provided by numerous different experimental methods. Sometimes, computational methods such as homology comparison are utilized for detection of interaction but such approaches just infer an interaction but do not provide a direct proof. Thus, *in silico* methods are considered in this scope as methods for prediction of protein-protein interaction.

Experimental methods can be subdivided into methods that only detect the mere existence of binary or complex interaction between putative binding partners and methods providing structural characterization of interactions [44]. One major advantage of the first group is that these methods are high throughput methods which means the technical requirements are quite low and many molecules can be screened very rapidly. Structural resolution of interactions requires more work and time and is not applicable for every putative interaction due to technical limitations. In the following, representatives from both groups are discussed and are listed in table 1

The yeast two-hybrid system can detect interactions *in vivo*. Two proteins A and B are fused with two domains of a transcription factor that binds to the promoter region of a reporter gene encoding e.g. a fluorescent protein [45]. Transcription only takes place if A and B actually bind. Therefore, detection of the fluorescent protein proves that proteins A and B bind.

Tandem affinity purification together with mass spectrometry (TAP-MS) also allows obtaining information about protein-protein interaction *in vivo* and consists of two steps [46]. First, components of the cell like protein complexes are filtered out by TAP which is related to affinity chromatography. Then the putative protein complex is split into ionized fragments. Detection and identification of the fragments facilitates the derivation of the polypeptide sequence.

Gene co-expression that can be detected with microarrays is based on the idea that proteins acting together have to be expressed at the same time in the same spatial compartment. The co-expression can be measured as gene expression profile from cell cycle experiments or expression levels at different conditions. Consequently, profiles of proteins that interact with each other should be similar [44]. In fact this is true for permanent interactions as was shown for ribosomes and proteasomes but not necessarily for transient interactions as they are not constrained with respect to time and localization [47].

Protein arrays detect actual protein interactions [44]. The solid phase contains immobilized capturing proteins that are probed with fluorescently labeled proteins which may be putative interaction partners. The binding affinity can then be derived from the extent of fluorescence.

The synthetic lethality method embeds mutations in two different genes [44]. The observation that mutation of only one gene results in viable cells whereas the combination causes cell death allows the following conclusions. Either the gene products act in parallel redundant metabolic networks or they play an

important role in the same network. Furthermore a possible physical interaction between them may be assumed.

A detailed characterization such as the number and type of amino acids which are involved in a protein-protein interaction or the size of the interface area can be obtained with methods detecting the three dimensional structure of a complex. The disadvantage of those techniques is that they require time and money intensive experimental work. Therefore, the number of structurally resolved protein-protein complexes is relatively low. Due to technical limitations, not every kind of interaction can be detected such as large proteins or interactions of proteins in membrane layers. Consequently, one has to keep in mind that structural data are biased towards structures that can be easily detected. Thus, the quantity of certain classes of resolved protein structures does not necessarily reflect its biological importance. As an example, membrane proteins are underrepresented in structural database but they play an important role in any living cell and cover a wide range of functionalities.

The majority of protein-protein interactions in the RCSB database has been detected by X-ray crystallography. A schema of the process is visualized in figure 7 [48]. The crystallized protein is radiated with X-Rays resulting in a diffraction

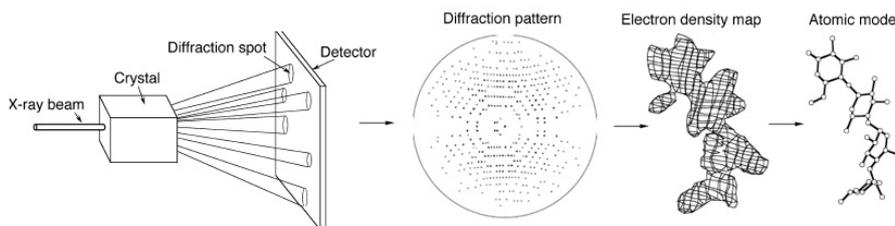


Figure 7: X-Ray detection of molecules.

pattern. With this information an electron density map can be generated which finally allows deducing a structure model of the molecule. The drawback of this method, however, is that the interface structure might have formed during crystallization and actually does not exist *in vivo* [8].

Nuclear magnetic resonance (NMR) only has a slightly lower resolution in comparison to X-ray method and additionally allows the detection of hydrogen atoms. However, preparation of samples is very complex. The method requires substantial quantities of purified proteins. Also, the method is currently technically limited to samples of ca. 60 kDa size which is too low for most protein complexes.

Other spectroscopic methods allow the detection of protein-protein interactions *in vivo* such as fluorescence resonance energy transfer (FRET) which uses labelling of target proteins with fluorophores. Surface plasmon resonance does not require this labelling.

Some experimental method are suitable for another characterization of protein-protein interactions. Isothermal titration calorimetry measures the enthalpy of binding which can be used for assignment of kinetic values to protein-protein interactions [49]. However, data about protein-protein interaction kinetics are still quite rare making it difficult to use this information for statistical analysis.

Method	HT	Type	Characterization
Yeast2Hybrid	+	binary	identification
Affinity purification	+	complex	identification
DNA microarrays / Gene coexpression	+	functional	identification
Protein microarrays	+	complex	identification
Synthetic lethality	+	functional	identification
Phage display	+	complex	identification
X-ray crystallography	-	complex	structural
NMR spectroscopy	-	complex	identification, structural, kinetic
Fluorescence resonance energy transfer	-	binary	biological
Surface plasmon resonance	-	complex	kinetic
Atomic force microscopy	-	binary	mechanical
Electron microscopy	-	complex	structural

Table 1: Overview of experimental methods to detect protein-protein interactions. A plus-sign in the second column indicates *high* throughput techniques. The third column provides information about which type of interaction can be detected. 'Binary' refers to only pairwise interaction whereas 'complex' allows detection of arbitrarily complex modes of interaction. 'Functional association' indirectly reveals interaction by observing effects of putative binding partners.

## 1.7 Prediction methods

Prediction methods which are related to protein-protein interactions can be grouped into three different categories. First are methods providing an estimation whether two putative binding partners may interact or not. Usually, protein sequence data is applied in this context allowing for large scale analyses. The information can then be exploited for further applications such as protein-protein interaction networks or metabolic pathways. Second are methods that predict residues on a protein sequence forming an interface. To this end, either non-structural as well as structural data are applicable. Third are methods that determine a potential 3D structure for complexes for which the existence of interaction is already known. Nonetheless these methods are also a kind of protein-protein interaction prediction as these methods aim at revealing the highest level of interaction details. The difference between second and third group is that the methods and features from the second group can be applied to one chain in a protein interface whereas for the third group both chains are required.

The predicted protein binding sites may not only be candidates for protein-protein interactions but might be also suitable for binding of small molecules making these concepts an interesting research area for drug design.

The following section discusses features and computational methods which have been used for the prediction of protein-protein interactions:

### 1.7.1 Prediction of protein-protein interaction

The methods in this section only detect the mere existence of a protein-protein interaction. Further details such as the residues that participate in the interaction or the shape of the interface are not provided.

**Phylogenetic profile:** The basic assumption for this method is that putative interacting proteins which are functionally linked also co-evolve and have orthologs in the same subset of sequenced organisms. Thus the systematic simultaneous presence or absence of proteins in many different genomes which are extracted from the phylogenetic tree infers a correlation between them which means they might be potential interacting partners. The concept is related to co-expression as protein chains sharing the same expression profile are maintained during evolution in order to preserve the functionality of the complex [50, 51]. One drawback is the high computational cost for the creation of the phylogenetic profile. Besides, the assumption is not always true as there might be ubiquitous proteins which occur in all genomes [52].

**Gene neighbourhood:** Genes with closely related functions whose transcripts might interact with each other may be located close to each other in the sequence and are transcribed simultaneously. In bacteria, these proteins are often transcribed as a single unit or an operon whereas in eukaryotes they are just co-regulated [50].

**Gene fusion:** As a special case of gene neighbourhood, gene fusion takes place if two proteins A and B from organism 1 form a single protein in organism 2. Such an observation leads to the conclusion that the separate proteins in organism 1 might interact with each other [53].

**Gene co-expression similarity:** As mentioned above, strong co-expression of proteins suggests that they might interact with each other. The expression profile similarity can be calculated as a correlation coefficient between relative expression of two genes or the gene products [44]. Several studies showed that interacting proteins tend to share their expression profile in comparison to non-interacting proteins [52, 50]. However, this method is only applicable if the proteins are dependent upon each other. As it was discussed earlier, this is mainly true for permanent complexes as the involved protein chains are not stable alone and thus require complex formation just after expression.

**Correlated mutation:** The idea behind correlated mutation analysis is that a mutation at the binding interface of one protein involved in an interaction might result in the loss of complex formation [54]. For compensation, the binding partner may also mutate the complementary residue(s) at its binding interface to restore the interaction. This is the case for proteins A and B, when a mutation at certain position in protein A goes along with a mutation at a certain position with protein B. So the existence of correlated mutations in multiple sequence alignments of the two proteins can serve as indicator for protein-protein interaction.

### 1.7.2 Prediction of protein-protein interfaces

In comparison to the previous section the following prediction methods provide the information which residues of the binding partners participate in the interaction.

**Conservation of interface/non-interface residues:** Interface residues are believed to be more evolutionary conserved in comparison to non-interface residues. The reason for this is that a certain biological function which is encoded in an interface is worth being maintained during evolution. Mutation at the interface may result in a loss of its functionality and may reduce the efficiency of the organism which represents an evolutionary disadvantage [11]. The meaning of conservation is controversially discussed. Caffrey *et al.* postulated that the difference of conservation between interface residues and other residues at the surface of a protein is too small to be a suitable predictor [55]. A more recent analysis of a large scale dataset of protein complexes revealed a higher importance of conservation in the interface region than expected before [56].

**Sequence information:** The distribution of amino acids at the interface might be different in comparison to rest of the protein. For example, it is commonly known that histidine, serine and tyrosine residues are enriched at active sites of enzymes as they may easily change their protonation states. Glaser *et al.* found that some residues are preferentially found at interface sites [57]. In their analysis, large hydrophobic residues such as leucine and tryptophan were preferentially found at interface sites whereas small amino acids like glycine and alanine rarely occur. Interface prediction from sequence alone exhibits two advantages: The analysis can be performed on a much larger dataset or even genome-wide, as the number of structurally known protein complexes is rather low in comparison to the number of known sequences [58]. Besides, analysis of sequence data is easier to perform and usually requires less computer power than investigating more detailed information such as structural data. But this prediction method has several limitations. For example, some approaches use neighboring information to derive the probability for a residue to lie at the interface or not [59]. However, interface sequence is usually interrupted by stretches of residues which are located outside the interface. Using sequence information in connection with further data such as conservation may increase the success rate [58].

**Hydrophobicity:** Considering the distribution of hydrophobic and hydrophilic residues located at the interface site can be regarded as a generalization of the amino acid composition. Here, the amino acids are grouped into polar and non-polar ones. Besides, hydrophobicity scales like the Kyte-Doolittle index may be applied that consider the quantitative extent of hydrophobicity per residue. It was found that hydrophobic interactions are important for the stability of a protein-protein interface [20]. In particular, obligate interfaces tend to be more hydrophobic in comparison to non-obligate interfaces. The latter interface type even appears to have a similar distribution of hydrophobic residues as non-interfacial surface areas. The reason is that too many hydrophobic residues would be unfavorable as non-obligate interface areas can be exposed to solvent in the non-bound state of the protein binding partners.

### 1.7.3 Prediction of protein-protein interface structure

**Protein-protein docking:** Protein-protein docking is the most common method for computer aided prediction of interface structures. The main principle is the tight complementarity of the two associating interfaces. In most cases, this will require conformational adaption of side-chain rotamers and, in about 10% of the known cases, also rearrangements of the protein backbone [60]. However, taking into account the full conformational flexibility of all residues costs too much computation time and thus docking algorithms carry out generalizations to reduce the search space. An important strategy is to keep the protein chains rigid during the docking process so that rotation and translation are the only degrees of freedom. Such approaches work very well for structures that undergo small conformational changes upon complexation. Following the concept of CASP (Critical Assessment of Techniques for Protein Structure Prediction) the CAPRI competition aims at providing a basis for comparison of several prediction techniques with each other to evaluate their reliability. Besides, the quality of docking results can be estimated with benchmark sets [61]. Despite continuous refinement of docking methods the induced fit problem makes prediction of protein-protein interactions difficult. Besides, the computational cost for all approaches is quite large [62].

**Homology modeling:** The protein interaction prediction through tertiary structure webserver (Interprets) provides structures of protein-protein interactions which are based on homology modeling [63]. The approach is similar to homology modeling of single proteins. Given two sequences, homologous sequences are searched for which structural data as complex is available. The residues which are known to participate in the formation of the complex are compared with the residues of the sequence to derive a score providing an estimation whether there exists an interaction for the input sequences or not. Kundrotas *et al.* tested the reliability of homology modeling with entries from a protein-protein interaction database and achieved a false-true positive prediction rate between 5:1 and 7:1 [64].

**Threading:** This method is based on single chain threading and outputs suitable structures from a template database for a target sequence. To this end, each residue of the input sequence is placed on a position of a template structure followed by an evaluation how well the target fits the template. In contrast to homology modeling, threading additionally applies structural information for the prediction. Liu *et al.* created a protocol to derive a complex structure with threading which is included in Multiprospector [65]. With the method, protein-protein interactions in yeast were successfully predicted. Similar to homology modeling, threading suffers from the same drawback. The reliability is strongly dependent upon the applied template structure, if one is available at all.

### 1.7.4 Function prediction

In many cases the functions of predicted or structurally resolved protein-protein interactions are not known. Thus, function prediction of protein-protein interactions is an important subject of current research. Knowing the biological

meaning of an interaction does not only serve as a feature for a better characterization of an interaction but it also helps to understand the principles of how interactions cooperate in metabolic pathways. One way to derive the functionality of a protein complex is to detect homologous sequences for which the function is known. The approach is based on the idea that orthologous proteins having the same interaction with other orthologous proteins also share the same function. Jaeger *et al.* used structural conservation of interaction networks to infer functions for protein-protein interactions [66]. Further studies reported successful predictions of protein-protein interactions in species like *C. albicans*, *A. thaliana* and *H. sapiens* [67].

The structure of a protein complex and especially its interface area is also subject for functional analysis. Several methods and programs like Multiprot or Galinter allow the calculation of a similarity score for structural comparison [39, 40]. It is feasible to assume that the structure of a protein or protein complex reflects its function. Nonetheless, it is not necessarily unambiguous to derive a biological meaning from the structure alone. Keskin *et al.* clustered structurally similar interfaces into three groups [16]. It was found that similar interfaces may have different functions. An explanation for this observation is that nature re-uses favorable conformations and applies them for many different purposes. Thus, structural similarity alone does not appear to be well suited for function prediction. Another problem is that different folds can perform the same function [68].

## 1.8 Protein-small molecule interactions

As mentioned in the initial chapter, protein-small molecule interactions represent the most important group of interactions which are used for therapeutic purposes. Figure 8 gives an overview of the new drugs which were approved by FDA in the last few years [69]. Despite the fact that the expenses for pharmaceutical research increased largely over the past years the number of novel drugs declined demonstrating the difficulties and hurdles in pharmaceutical research. In 2007, the expenses for drugs in Germany accounted for about 42 billion €.



Figure 8: Number of innovative drugs approved by the Food and Drug Administration (FDA).

The costs for drug development are estimated to vary from 500 million US\$

to 2000 million US\$ depending upon the kind of drug and the pharmaceutical company [69]. Only about 8% of drug candidates gains a FDA approval and is eventually brought to market [70]. These facts illustrate the necessity to optimize the process of drug development which is still a challenging research subject.

The classical drugs were based on substances from plants or microorganisms that appeared to be suitable for therapeutic application. Discoveries were often made by chance, for instance the finding of penicillin by Alexander Fleming or by trial and error methods which is comparable to finding a needle in the haystack. At the end of the 19th century, drug development was optimized by incorporating scientific methods and systematic search for new compounds. Also, known drugs could be isolated and purified from natural and new substances were synthesized due to progress in chemistry. By the end of the 20th century, a further breakthrough was the introduction of high throughput screening methods allowing for finding new biologically active compounds by performing thousands up to millions of pharmacological, biological and genetic tests [71]. As methods are time-consuming and cost-intensive, filtering techniques are applied to reduce the search space to a smaller area of highly promising compounds for further testing.

A typical binding site or pocket for small molecules encompasses the following features. The shape of the binding site usually resembles a cleft in which the small molecule fits. This is in accordance to the lock and key model as described in the introduction. Besides, the contact regions between the protein and the ligand should be complementary with respect to physicochemical features [72]. Usually, the lipophilic part of the ligand is in contact with the lipophilic parts of the protein (side chains of the amino acids Ile, Val, Leu, Phe, and Trp, perpendicular contact to amide bonds). In addition, several hydrogen bonds are formed and some of them can be charge assisted. Cation- $\pi$  interactions and metal complexation can also play a significant role in individual cases. Unfavorable contacts occur quite rarely such as unpaired polar groups. Figure 10 gives an overview of the different types of atomic contacts in protein ligand interactions. The conformation of the binding region should be energetically favorable which can be achieved by structural changes upon binding as described with the induced fit model.

Computational methods greatly support the drug design process. First, there are methods to detect pockets on proteins that may be binding sites for small molecules provided that the receptor for a protein is unknown. Either geometrical features of the protein surface are considered like Surfnet [73], Ligsite [74] and PASS [75] or energetic criteria are applied like Q-site finder [76]. Such a binding site or pocket may not exist continuously [77]. Thus, it may happen that not all possible binding sites can be detected from the crystal structure of a protein which just represents the time-averaged conformation. In this context, molecular modeling methods can help to disclose more conformations with pockets. Second, ligands are compiled that fit into the putative binding sites. Given its geometrical and physico-chemical features, a pattern of attributes for the ligand can be defined, the so-called "pharmacophor". It is then used as query input for a compound database. The hits are filtered according to certain criteria and are applied together with the target protein for molecular docking tools such as AutoDock or FlexX. It is noteworthy to mention that docking results in finding binding modes with the lowest interaction energy for a given

ligand [78]. However, this does not necessarily mean that the ligand actually binds to the target. Either the binding feature must be proven experimentally or evaluation methods have to be applied to assess the affinity of the ligand.

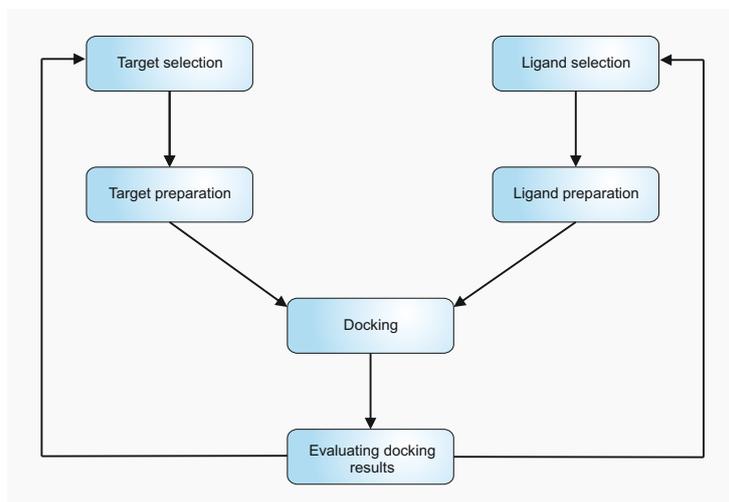


Figure 9: Docking workflow. First, the structure of the target molecule and an appropriate ligand must be chosen, followed by preparation steps which depend on the applied docking method. The output usually consists of several docking results that are scored and filtered. Diagram adapted from [78]

Development of a compound acting as ligand for a binding site only represents one step towards creation of a novel drug. Pharmacokinetic aspects have to be taken into consideration which are described with the LADME model. Here, aspects are considered e.g. whether the compound can be dissolved under physiological conditions and remains stable. For oral application the drug needs to be capable to cross biological membranes such as the blood-brain barrier for drugs acting in the central nervous system. Within the organism the drug should not be decomposed by enzymes, e.g. unspecific esterases before reaching its target. Eventually the drug should be removable from the organism to prevent accumulation of the substance. Another important aspect is the toxicity of the drug which often occurs due to low specificity of the compound resulting in a number of undesirable side effects. The optimization of the lead compound presents a great challenge to pharmaceutical research. There are many ways to modify a lead compound such as addition or removal of hydrophilic or hydrophobic groups, variation of substituents or incorporation or cutting of rings [4]. Every change in the molecule affects its physico-chemical features, its structure and its biological activity, making ligand design a complex task. In the following some design principles are listed [4, 79]. Increasing lipophilic contacts between protein and ligand often results in a higher binding affinity. However, a highly lipophilic compound is less water-soluble. Usually, additional H-bonds only increase binding affinity if the interaction is stronger than in water. Rigid ligands form a stronger interaction in comparison to more flexible ligands due to lower loss of degrees of freedom. The incorporation of chirality may enhance selectivity of the molecule. Combinatorial chemistry helps to generate variants

of ligands and to find suitable candidates [80].

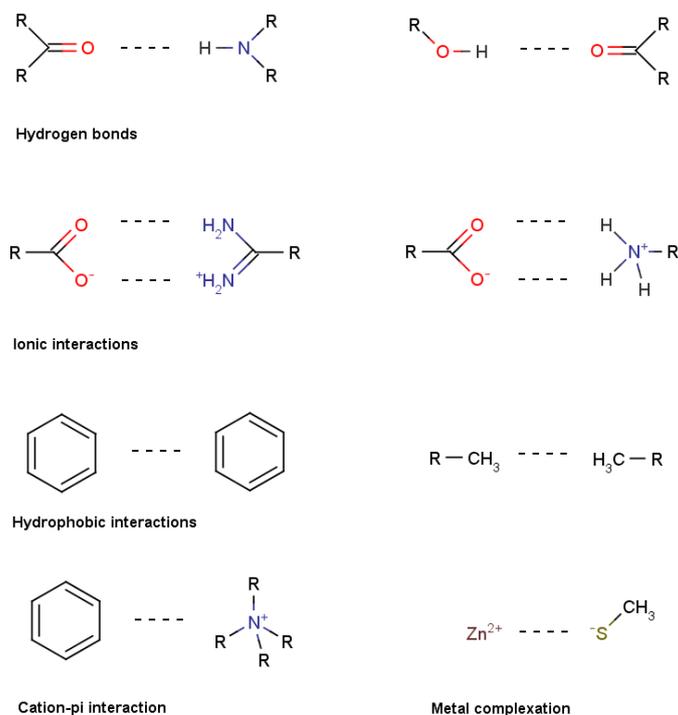


Figure 10: Non-covalent interaction found in protein-ligand interactions.

Design of small molecules for therapeutic applications poses numerous problems. Up till now only very few drugs are available that were created from *de-novo* design. One challenge is to find appropriate target proteins for which also knowledge about the structural details is available. However, many putative targets belong to the class of membrane proteins for which only very few structures are known. Another difficulty is the generation of ligands from the huge search space of chemical variations having the desired physico-chemical and pharmacological effects. Depending on the standards of counting, the number of target proteins varies between 218 and 324 [81] which is quite low in comparison the entirety of potential pharmacologically interesting targets, which is estimated to be around 6000 [82]. The statistics gives indication of the existence of many therapeutically relevant protein-ligand interactions that remain to be discovered.

## 1.9 Protein-protein and protein-small molecule interaction databases

In this section a number of databases are presented dealing with biomolecular interactions, in particular protein-protein interactions [83]. An overview of the most important databases which are all available through the internet is shown in table 2. Generally, they differ from each other in two aspects, namely the source of interacting data and the information content. Databases containing

Database	Type	Source	Number of datasets
DIP	P	E,S	57,683
BIND	P	E,C,S	188,517
MPact/MIPS	P	E,C,F	15,488
MINT	P	E,C	111,518
HPRD	P	E,C	38,806
String	P	E,C,F	>50 Mill.
ProtCom	P,D	S,H	1,770
3did	D	S,H	115,559
Pibase	D	S,H	216,739
Biogrid	P	E,C	240,207
Scoppi	D	S	102,084
PISA	P,L	S	Structures from RCSB
PDBBind	L	S	3214
AffinDB	L	S	474
Binding MOAD	L	S	14720

Table 2: Databases related to protein-protein and protein-small molecule interactions. The Type column describes the type of interaction; P stands for an interaction between protein chains, D refers to domain-domain interactions, L for interactions between protein and small molecule. Source refers to the sources for interaction data. E stands for high throughput experiments, S structural data, C manual curation and H interface homology modeling.

structural information aim at providing physicochemical features to characterize protein-protein interactions. Scoppi for instance classifies interfaces in PDB files from the RCSB database according to SCOP domain definition and extracts information such as interface size and amino acids involved in the interaction [84]. Pibase applies a similar approach using SCOP and CATH for a structural classification of protein-protein interactions at domain level [85]. As the name indicates, the database of 3D interacting domains (3DID) comprises a collection of domain domain interactions for which the structure is known. Besides, it also contains a hand curated set of peptide mediated interactions. Both groups are believed to represent the prevalent type of interaction in signaling and regulatory networks [86]. Beside two-chain structural data from the RCSB, the 3DID contains artificially created domain domain interactions which were generated from single chains. Commonly, data from these types of databases can be used for further areas of application such as derivation of scores for evaluation of protein-protein docking procedures or as template database for homology modeling.

Other databases incorporate high-throughput and/or low-throughput data. Usually these resources focus on the mere existence of an interaction between proteins and do not provide further information based on structural details. Such data can be highly beneficial for the setup of protein-protein interaction networks. A typical representative is the Biomolecular interaction network database (BIND) which comprises a collection of molecular interactions derived from high-throughput data submissions and hand-curated information gathered from the scientific literature [87]. The database includes interactions between proteins, DNA, RNA and small molecules. Besides it defines pathways

as collection of interactions which occur in a defined order in living organisms. STRING represents the most comprehensive database for protein-protein interactions so far. It covers about 2.5 million proteins from 630 organisms and contains more than 50 million interactions [88]. The database of interacting proteins (DIP) considers its interactions like a network. The involved protein sequences are regarded as nodes whereas the interactions represent the edges. DIP is restricted to sequential data and provides the existence of a binary or complex interaction between proteins [89]. The molecular interaction database (MINT) has access to a wide variety of detection methods but limits the interactions to the ones that are experimentally verified and were mined from literature by expert curators [90]. The database also offers a network view of the interactions. Similarly, Biogrid lists interactions detected by high throughput experiments from *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster* and *Homo sapiens* [91]. The Human Protein Reference Database focuses on protein-protein interaction exclusively from human being and only comprises hand curated dataset which explains the relatively low number of datasets in comparison to similar databases [92]. PISA provides interface information for different kinds of contacts (protein, small molecules, RNA, DNA) about the structure data from the RCSB [93]. PDBind and AffinDB focus on binding constants for protein-ligand complexes [94, 95]. Binding MOAD contains data about biologically relevant ligands that were extracted from RCSB [96].

Though many databases overlap with each other in their information content or the type of data presentation, they all appear useful as there is no general standard database for storage and retrieval of protein-protein interactions.

### 1.10 Aim of this work

The amount of biological data is constantly soaring such as protein data for which structural information is available. Figure 11 illustrates the number of new structures which were included in the PDB database during the last decades. The diagram reveals two aspects. First, the fast growth of available structures that also correlates to an increase of complex data allows a more and more subtle and comprehensive analysis of biomolecular contacts with respect to structure dependent features. Second, former research work only covered structures that were available at that time which is far less in comparison to the amount of data from today's view. This circumstance raises the question how much former scientific results will change when applied to larger datasets. The aim of this work was therefore to incorporate a modern database system for structural data on biomolecular interactions that is capable of doing automatic updates in order to preserve a nearly full coverage of data that can be exploited for any kind of analyses. Using the database, we performed a number of analyses concerning features of protein-protein interactions, in particular the group of obligate and non-obligate interactions. Combining information from PP and PL complexes, we generated a prediction method for binding sites of small molecules on PP interface sites. Finally, we tested the applicability of features of PP interactions for the prediction of their kinetic parameters.

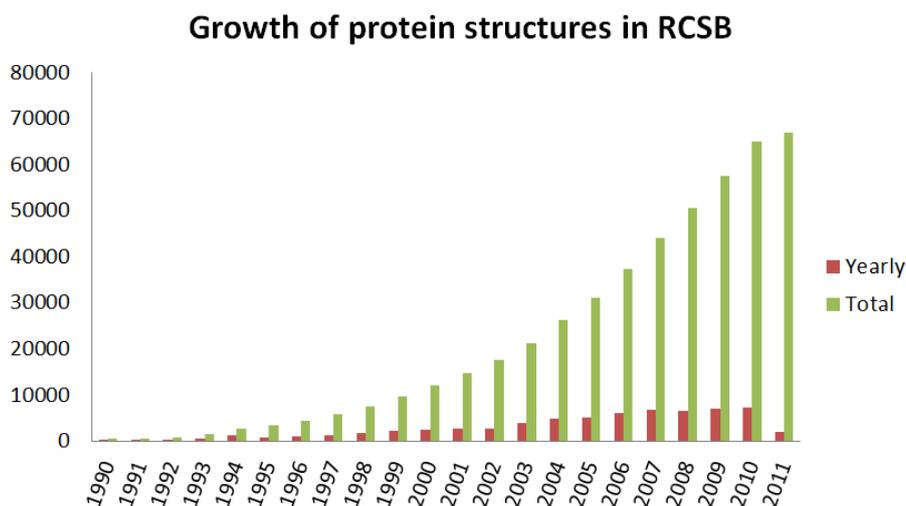


Figure 11: Number of PDB structures that were included into RCSB database per year.

## 2 Database management systems

### 2.1 Overview

Typically a database is applied whenever large amounts of data are processed in electronic computing. In its general meaning the term 'database' is very unspecific. It may be an Excel table storing lists of addresses or a database server such as Oracle or MySQL. In information technology a **database** is just a collection of data which is organized in a certain way [97]. The software which allows using, creating and modifying a database is called **database management system** (DBMS). A DBMS offers the following features demonstrating its usefulness in particular for large amounts of data:

- **Efficiency:** The main task of the DBMS is to store information in an efficient manner allowing fast access and modification of the data.
- **Data independence:** The user or an external program should not have to worry about the details of organization and physical storage of the data. The only way to access the data is through the DBMS. In other words, it is not dependent upon any specific external application logic.
- **Safety:** The DBMS is also responsible for consistency of the data and recovery of data after system crashes.
- **Data administration:** User management within the database increases safety and allows for better control of data access. For experts fine tuning of the database allows optimizing the database speed.

The architecture of a typical DBMS is illustrated in figure 12. A request to the database can be sent in different ways. A normal user enters a query in a web form or he uses the GUI of an application. Instead of a human being, a

program may also interact with a database. For this kind of communication a database driver is required which handles the requests and submits them to the database. Database management systems also offer a direct interaction which is command-line based and is intended for sophisticated users, programmers and database administrators.

The input for the DBMS consists of SQL commands which is a language standard especially for databases. As first step, the query is analysed syntactically and semantically by the query evaluation engine. After that a query evaluation plan is generated. To this end, an optimizer tries to transform the query into an equivalent term which takes less time and computation power during execution. Finally, database management applies the query evaluation plan and the desired data is retrieved from the physical data storage. A further element in a database management system is the concurrency control which is responsible to maintain data consistency. As an example, the concurrency control handles situations like the manipulation of the same data by more than one user at the same time. Due to the frequent transfer and manipulation of data in a DBMS, an abrupt system crash may also lead to data loss and inconsistencies. The recovery manager contains several mechanisms to restore the database as far as possible.

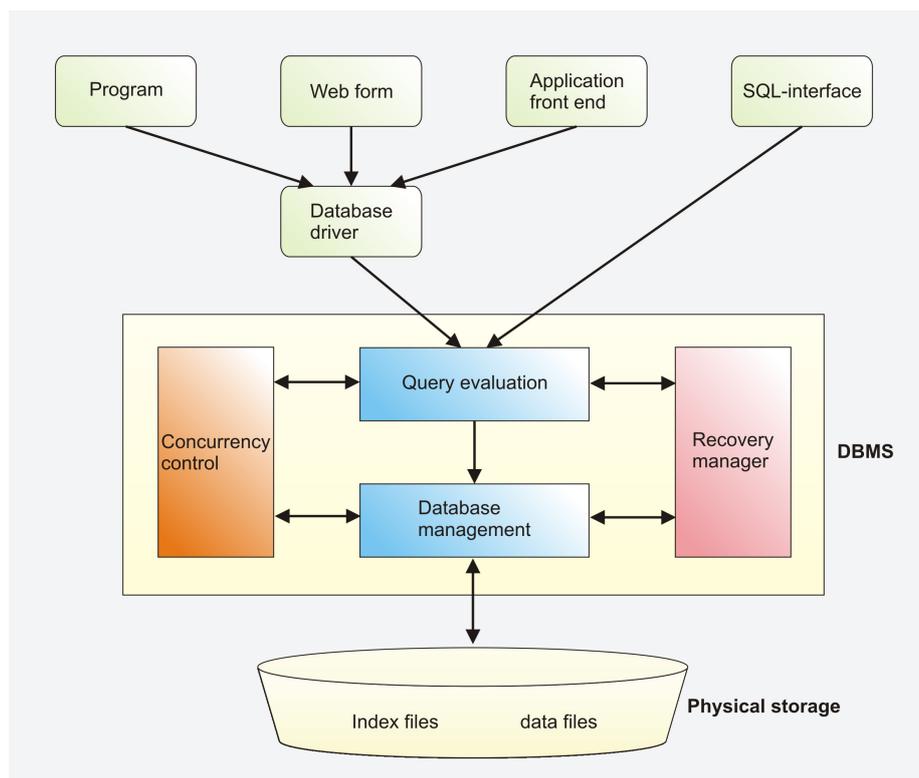


Figure 12: Architecture of a DBMS.

As application of computational power became more and more popular since the early sixties of the last century, several approaches were developed for the

storage and management of large amount of information. In 1970, E.F. Codd developed the relational model for the representation of data which has been the favorite model up till now. The major advantage of this model is its simplicity making it the basis for most of present database systems.

## 2.2 Relational model

A relation denotes the structure in which data is stored [98]. An instance of a relation is called relation variable. It can be thought of as a set of records containing the actual information. Every relation has a heading and a body. The heading is a set of attributes and the body is a set of tuples that conform to that heading. The most common way for representation of a relation is table form. The headlines that denote the columns commensurate the attributes in a relation. A table row is comparable to a tuple. Relations are based on the theory of sets. Consequently there is no order of the tuples and of the attributes in the relation. Besides there must not exist identical tuples in a relation. Every tuple has to be different from each other in at least one attribute value. A special feature of the tuple value is their atomicity which means that a value cannot be decomposed into pieces of information. This property is known as the first normal form. Further normal forms are listed in chapter 2.3.2.

In mathematical terms a relation schema  $R$  is a set of domain names  $f_i, 1 \leq i \leq n$  for a domain  $D_i$  [1]:

$$R(f_1 : D_1, \dots, f_n : D_n)$$

For every  $f_i$  let  $Dom_i$  be the set of values associated with the domain  $D_i$ . A relation variable is then defined as a set of tuples whose elements originate from domains  $D_1 \dots D_n$ :

$$\{\langle f_1 : d_1, \dots, f_n : d_n \rangle \mid d_1 \in Dom_1, \dots, d_n \in Dom_n\}$$

Thus, a relation variable consists of tuples with degree (or arity)  $n$ . The number of tuples in this set is denoted as cardinality. A relational database consists of relations with distinct relation names. Relations are not isolated from each other but may share a certain relationship. The most common relationships are binary ones which are detailed below and visualized in figure 13.

- one to one: Two relations  $R$  and  $S$  follow the one to one relation if one element from  $R$  is assigned to at most one element from  $S$  and vice versa.
- one to many, many to one: An element from  $R$  points to an arbitrary number of elements from  $S$  whereas one element from  $S$  may have a relation with at most one element from  $R$ .
- many to many: For an element from  $R$  the number of relations with elements from  $S$  lies between zero and many. The same holds for elements from  $R$ .

### 2.2.1 Keys

Keys are crucial for the identification of tuples in a relation. Let  $R$  be a relation schema and  $K := A_1, \dots, A_n \subseteq R$  so that for every relation variable  $r$  the

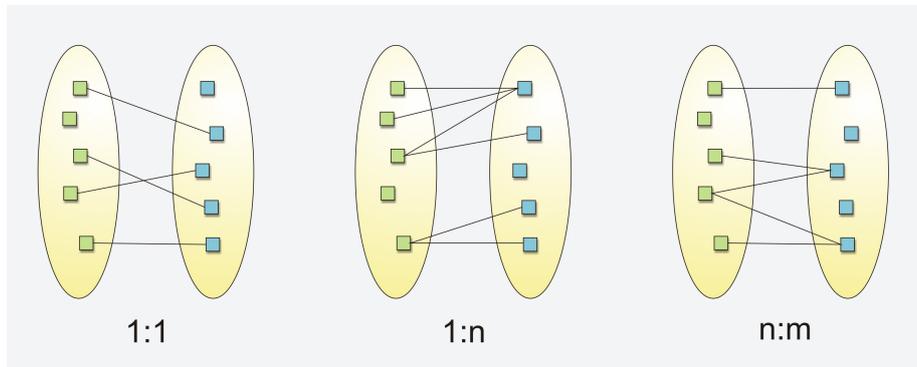


Figure 13: Relationships between relations.

following condition holds [99]:

$$\forall t_1, t_2 \in r [t_1 \neq t_2 \Rightarrow \exists B \in K : t_1(B) \neq t_2(B)]$$

This means that no two tuples may have the same values for the attributes in  $B$  (uniqueness constraint). If no proper subset of an element of  $K$  has the uniqueness property (irreducibility constraint) we call this key a candidate key, else it represents a superkey. A chosen candidate key is denoted a primary key. It is noteworthy to mention that a relation always contains at least one key because otherwise there might exist duplicate tuples. Additionally, primary keys can never have NULL as an entry in the tuple because this leads to loss of the uniqueness constraint. To conclude, the primary key is a minimal set of attributes which identifies a certain tuple in a relation. For a database keys are important for fast retrieval of data as primary keys are used for indexing allowing a faster access to the desired information.

A foreign key is defined as follows. Let  $P$  be the set of attributes in a relation  $R_1$  representing the primary key. Then a set  $FK$  in a relation  $R_2$  is a foreign key if and only if every tuple from  $R_2$  with attributes from  $FK$  has the same tuple value in  $R_1$ . A foreign key is not necessarily a primary key, so a tuple from  $R_1$  may reference one or several tuples from  $R_2$ . A foreign key establishes a link between relations. Combining tuples from  $R_1$  and  $R_2$  having the same primary key and foreign key, respectively, bring together data lines sharing the same meaning.

### 2.2.2 Entity relationship (ER)

The basic task for the design of a database is collecting the various types of data and establishing connections between them. To represent these domains together with their relationships, the entity relationship model was introduced and is widely used in the field of database design. The basic elements of ER are shown in figure 14. In (a) - (c) the upper diagram shows a more conceptual diagram for database representation. Here, boxes represent an entity containing information in terms of attributes. Entities are related to each other through relationships that are visualised as diamonds. The lower model is typically applied for physical design of database tables. In the examples, a relation between

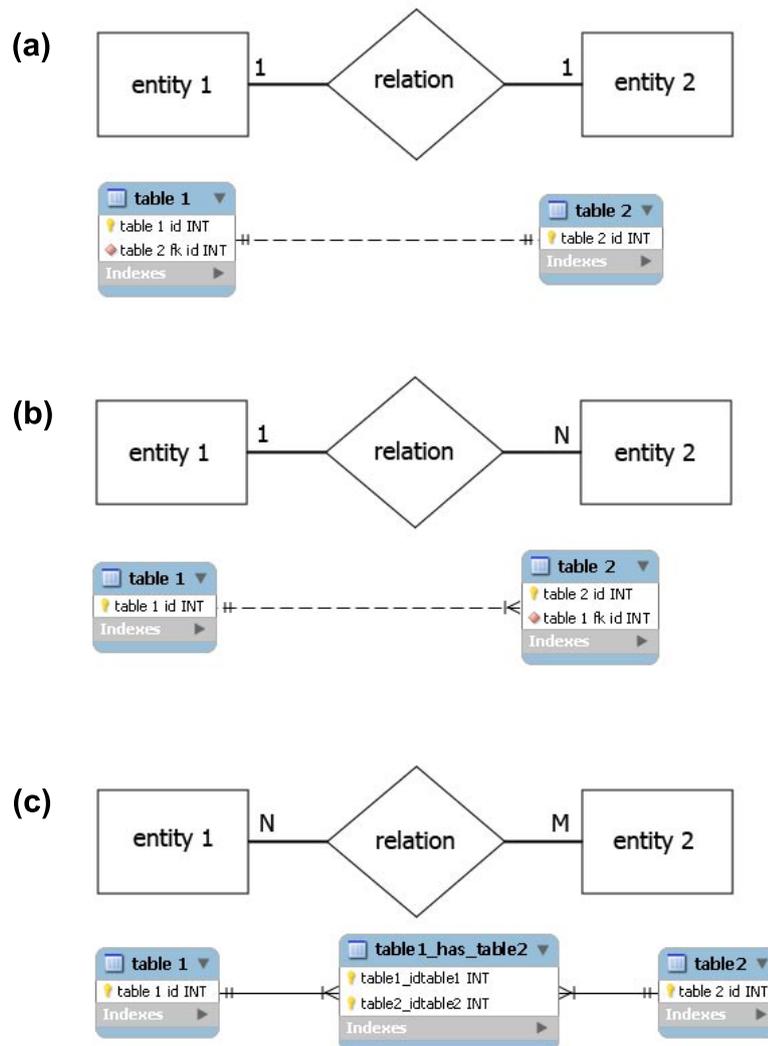


Figure 14: Two types of entity relationship (ER) visualizations. (a) describes 1:1, (b) 1:n, and (c) n:m relationships. For the lower model in (a) - (c), a primary key is illustrated with a key icon and the foreign key with a red diamond.

*table 1* and *table 2* is shown. (a) and (b) describe non-identifying relations that is visualised by the dashed line which indicates that there is no overlap between the foreign key (here: *table 2 fk id*) and the primary key. In (c) the  $n : m$  relationship between *table 1* and *table 2* is achieved using a mapping table *table\_1\_has\_table\_2*. Here, we have an identifying relation because the primary key in *table\_1\_has\_table\_2* also contains the foreign key for *table 1* and *table 2* respectively.

### 2.2.3 Relational algebra

The relational algebra comprises a set of operators that can be applied to relations as input and provide a relation as output as well. In practice, database query languages allow the formulation of search queries including operations such as search for equality, consideration of certain relations, combination of several columns etc. Unfortunately, there is no unique standard for the implementation of these operations and thus the possibilities for the formulation among the database query languages are different from each other. For example, in Oracle, subtracting of set of tuples from another set of tuples can be achieved with the MINUS keyword whereas MySQL does not bear such an operation so far. The most important relation operations are described in the following. Many of them are based on the theory of sets such as union or difference whereas others are more specialized methods such as renaming or projection.

**Renaming:** In queries relations can be repeatedly used. Sometimes the user desires an alternative name for a relation for better understanding and a clearer overview. In these cases it makes sense to rename relations. For this, we define the rename operator  $\rho$ :

$$\rho_{newname}(oldname)$$

This operation replaces the denotation *oldname* by *newname*.

**Selection:** Selection restricts the tuples from  $R$  to the ones fulfilling a given predicate. We define the selection operator  $\sigma$  as follows:

$$\sigma_{predicate}(R)$$

In general, the predicate consists of the following elements:

- Attribute names from  $R$  or constants
- Comparison operators:  $=, \neq, \leq, \geq, <, >$
- Logical operators:  $\vee, \wedge, \neg$

**Projection:** Whereas selection selects certain tuples, projection selects given attributes. The resulting set of tuples only consists of the attributes which contain the desired information. The projection is defined using the  $\pi$  operator:

$$\pi_{attributes}(R)$$

**Union:** Two relations  $R$  and  $S$  can be merged provided that they both have the same attribute names and attribute types.

$$R \cup S$$

**Difference:** The resulting schema formed by the difference operation contains tuples that occur in  $R$  but not in  $S$

$$R - S$$

**Cartesian product:** Cartesian product is the basic operation for establishing a connection between different relations. The following operation creates all possible pairs  $(|R| \cdot |S|)$  between relations  $R$  and  $S$ :

$$R \times S$$

The attribute set of the resulting relation consists of the union of the relation schemes from  $R$  and  $S$ :  $sch(R) \cup sch(S)$ .

**Join:** The preceding operators suffice for a formal definition of the relational algebra. From this point of view a join can be considered as syntactical sugar. In practice it plays an important role in the combination of data which are derived from more than one relation. In most cases the Cartesian product does not make sense in a query formulation because most of the tuples created by Cartesian product do not share the same context with each other and are therefore without any meaning. Joins, however, merge relations with respect to attributes they have in common. Let us suppose that  $R$  has  $m + k$  attributes  $A_1, \dots, A_m, B_1, \dots, B_k$  and  $S$  consists of  $n + k$  attributes  $B_1, \dots, B_k, C_1, \dots, C_n$ . Altogether,  $R \bowtie S$  has  $m + n + k$  attributes. We assume that attributes  $A_i$  and  $C_i$  do not share the same names. There are different join operators that differ from each other with respect to the way which tuples to use as output. The **natural join** or **inner join** only takes combinations of tuples from L and R for which the values of the shared attribute are the same. In the example below this is the case for the first tuples from L and R respectively.

L		R	$\bowtie$		=	Result																												
<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 5px;">A</td><td style="padding: 2px 5px;">B</td><td style="padding: 2px 5px;">C</td></tr> <tr><td style="padding: 2px 5px;"><math>a_1</math></td><td style="padding: 2px 5px;"><math>b_1</math></td><td style="padding: 2px 5px;"><math>c_1</math></td></tr> <tr><td style="padding: 2px 5px;"><math>a_2</math></td><td style="padding: 2px 5px;"><math>b_2</math></td><td style="padding: 2px 5px;"><math>c_2</math></td></tr> </table>	A	B	C	$a_1$	$b_1$	$c_1$	$a_2$	$b_2$	$c_2$		<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 5px;">C</td><td style="padding: 2px 5px;">D</td><td style="padding: 2px 5px;">E</td></tr> <tr><td style="padding: 2px 5px;"><math>c_1</math></td><td style="padding: 2px 5px;"><math>d_1</math></td><td style="padding: 2px 5px;"><math>e_1</math></td></tr> <tr><td style="padding: 2px 5px;"><math>c_3</math></td><td style="padding: 2px 5px;"><math>d_2</math></td><td style="padding: 2px 5px;"><math>e_2</math></td></tr> </table>	C	D	E	$c_1$	$d_1$	$e_1$	$c_3$	$d_2$	$e_2$				<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 5px;">A</td><td style="padding: 2px 5px;">B</td><td style="padding: 2px 5px;">C</td><td style="padding: 2px 5px;">D</td><td style="padding: 2px 5px;">E</td></tr> <tr><td style="padding: 2px 5px;"><math>a_1</math></td><td style="padding: 2px 5px;"><math>b_1</math></td><td style="padding: 2px 5px;"><math>c_1</math></td><td style="padding: 2px 5px;"><math>d_1</math></td><td style="padding: 2px 5px;"><math>e_1</math></td></tr> </table>	A	B	C	D	E	$a_1$	$b_1$	$c_1$	$d_1$	$e_1$
A	B	C																																
$a_1$	$b_1$	$c_1$																																
$a_2$	$b_2$	$c_2$																																
C	D	E																																
$c_1$	$d_1$	$e_1$																																
$c_3$	$d_2$	$e_2$																																
A	B	C	D	E																														
$a_1$	$b_1$	$c_1$	$d_1$	$e_1$																														

In left outer join all the tuples from the left relation are used for the resulting tuple set. The output consists of the natural join between L and R and additionally the remaining tuples from L. Non-existing values in the tuples are filled with NULL.

L		R	$\bowtie_{\text{left}}$		=	Result																																	
<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 5px;">A</td><td style="padding: 2px 5px;">B</td><td style="padding: 2px 5px;">C</td></tr> <tr><td style="padding: 2px 5px;"><math>a_1</math></td><td style="padding: 2px 5px;"><math>b_1</math></td><td style="padding: 2px 5px;"><math>c_1</math></td></tr> <tr><td style="padding: 2px 5px;"><math>a_2</math></td><td style="padding: 2px 5px;"><math>b_2</math></td><td style="padding: 2px 5px;"><math>c_2</math></td></tr> </table>	A	B	C	$a_1$	$b_1$	$c_1$	$a_2$	$b_2$	$c_2$		<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 5px;">C</td><td style="padding: 2px 5px;">D</td><td style="padding: 2px 5px;">E</td></tr> <tr><td style="padding: 2px 5px;"><math>c_1</math></td><td style="padding: 2px 5px;"><math>d_1</math></td><td style="padding: 2px 5px;"><math>e_1</math></td></tr> <tr><td style="padding: 2px 5px;"><math>c_3</math></td><td style="padding: 2px 5px;"><math>d_2</math></td><td style="padding: 2px 5px;"><math>e_2</math></td></tr> </table>	C	D	E	$c_1$	$d_1$	$e_1$	$c_3$	$d_2$	$e_2$				<table style="width: 100%; border-collapse: collapse;"> <tr><td style="padding: 2px 5px;">A</td><td style="padding: 2px 5px;">B</td><td style="padding: 2px 5px;">C</td><td style="padding: 2px 5px;">D</td><td style="padding: 2px 5px;">E</td></tr> <tr><td style="padding: 2px 5px;"><math>a_1</math></td><td style="padding: 2px 5px;"><math>b_1</math></td><td style="padding: 2px 5px;"><math>c_1</math></td><td style="padding: 2px 5px;"><math>d_1</math></td><td style="padding: 2px 5px;"><math>e_1</math></td></tr> <tr><td style="padding: 2px 5px;"><math>a_2</math></td><td style="padding: 2px 5px;"><math>b_2</math></td><td style="padding: 2px 5px;"><math>c_2</math></td><td style="padding: 2px 5px;">NULL</td><td style="padding: 2px 5px;">NULL</td></tr> </table>	A	B	C	D	E	$a_1$	$b_1$	$c_1$	$d_1$	$e_1$	$a_2$	$b_2$	$c_2$	NULL	NULL
A	B	C																																					
$a_1$	$b_1$	$c_1$																																					
$a_2$	$b_2$	$c_2$																																					
C	D	E																																					
$c_1$	$d_1$	$e_1$																																					
$c_3$	$d_2$	$e_2$																																					
A	B	C	D	E																																			
$a_1$	$b_1$	$c_1$	$d_1$	$e_1$																																			
$a_2$	$b_2$	$c_2$	NULL	NULL																																			

Analogously to the previous operation, right outer join keeps all elements from R for the resulting tuple set.

L		R		Result				
A	B	C	$\bowtie$	A	B	C	D	E
$a_1$	$b_1$	$c_1$		$a_1$	$b_1$	$c_1$	$d_1$	$e_1$
$a_2$	$b_2$	$c_2$		NULL	NULL	$c_3$	$d_2$	$e_2$

The outer join is a combination between natural join, left outer join and right outer join.

L		R		Result				
A	B	C	$\bowtie$	A	B	C	D	E
$a_1$	$b_1$	$c_1$		$a_1$	$b_1$	$c_1$	$d_1$	$e_1$
$a_2$	$b_2$	$c_2$		$a_2$	$b_2$	$c_2$	NULL	NULL
				NULL	NULL	$c_3$	$d_2$	$e_2$

The semi join between L and R outputs the same tuples as the natural join but restricts the attributes in the resulting relation to the ones from L.

L		R		Result		
A	B	C	$\bowtie$	A	B	C
$a_1$	$b_1$	$c_1$		$a_1$	$b_1$	$c_1$
$a_2$	$b_2$	$c_2$		$a_2$	$b_2$	$c_2$

The semi join between R and L outputs the same tuples as the natural join but restricts the attributes in the resulting relation to the ones from R.

L		R		Result		
A	B	C	$\bowtie$	C	D	E
$a_1$	$b_1$	$c_1$		$c_1$	$d_1$	$e_1$
$a_2$	$b_2$	$c_2$		$c_3$	$d_2$	$e_2$

## 2.3 Design principles

The manageability and performance are strongly dependent upon the structure of the database. Basically, the main aim of database design is to reduce the magnitude of redundancy, which means that a certain piece of information should be stored only once in the database. An efficient design decreases the consumption of storage capacity and possibly reduces computation time for queries because less amount of data has to be processed. The negative effects of redundancy are known as anomalies which are described in the next section. In database design there are general principles giving an indication how to avoid redundancy and to establish an efficient database structure.

### 2.3.1 Anomalies

The existence of anomalies in databases result in difficulties maintaining the consistency of the data. In general we distinguish between three anomalies mentioned below:

- **Insert anomaly:** Let us assume that a database table is filled with a new entry. However, not all data are available so that for some table cells no information is inserted. If this information is crucial for a clear identification of the record, there might be redundancy.

- **Update anomaly:** A bad design principle is to store different kinds of data in a single table. This may result in redundant pieces of data. During an update process all of these records have to be updated else inconsistent data occur.
- **Deletion anomaly:** If records are deleted from a table containing several kinds of information then additional data might be lost as an undesired side-effect.

### 2.3.2 Normalization

The normalization procedure helps to avoid anomalies and to improve the design of the database structure. So far, in database theory six normal forms are known (first, second, third and so on). If a relation conforms to a normal form then it is said that the relation is in the  $n$ 'th normal form. If a relation is in the  $(n + 1)$ st normal form then it is also in the  $n$ 'th normal form. The most relevant normal forms which are applied from a design point of view are the third normal form and the Boyce-Codd normal form which are detailed below. The basic concept for normal forms starting from number two is the functional dependency (FD) [100]. Let  $\alpha$  and  $\beta$  be sets of attributes from a relation  $R$ . Then

$$\alpha \rightarrow \beta$$

means that for all pairs of tuples  $r, t \in R$  the following condition holds: if  $r.\alpha = t.\alpha$  then  $r.\beta = t.\beta$ . In plain words this means that the *alpha*-values unambiguously determine the *beta*-values. It can be said the higher the normal form the stricter the conditions for the functional dependencies. Note that a key can be defined with the help of functional dependency. A set  $\alpha$  constitutes a superkey for a relation  $R$  if

$$\alpha \rightarrow R$$

Additionally if  $\alpha$  is minimal which means:

$$\forall A \in \alpha : \alpha - \{A\} \not\rightarrow \beta$$

then  $\alpha$  is a candidate key for relation  $R$ . The finding of functional dependencies is based on the meaning of the data in the relation and the relationship among each other. This basic set of functional dependencies is denoted as  $F$ . There might exist further functional dependencies which can be derived from inference rules providing the closure  $F^+$  for the set  $F$ . Three inference rules (*Armstrong axioms*) suffice to formulate the closure whereby  $\alpha$ ,  $\beta$  and  $\gamma$  are subsets from a relation  $R$ :

- Reflexivity: If  $\beta \subseteq \alpha$  then  $\alpha \rightarrow \beta$
- Enhancement: If  $\alpha \rightarrow \beta$  then  $\alpha \cup \gamma \rightarrow \beta \cup \gamma$
- Transitivity: If  $\alpha \rightarrow \beta$  and  $\beta \rightarrow \gamma$  then  $\alpha \rightarrow \gamma$

A reflexive FD is also denoted as *trivial FD*. The following derived rules are meant for better understanding:

- Union: If  $\alpha \rightarrow \beta$  and  $\alpha \rightarrow \gamma$ , then  $\alpha \rightarrow \beta \cup \gamma$

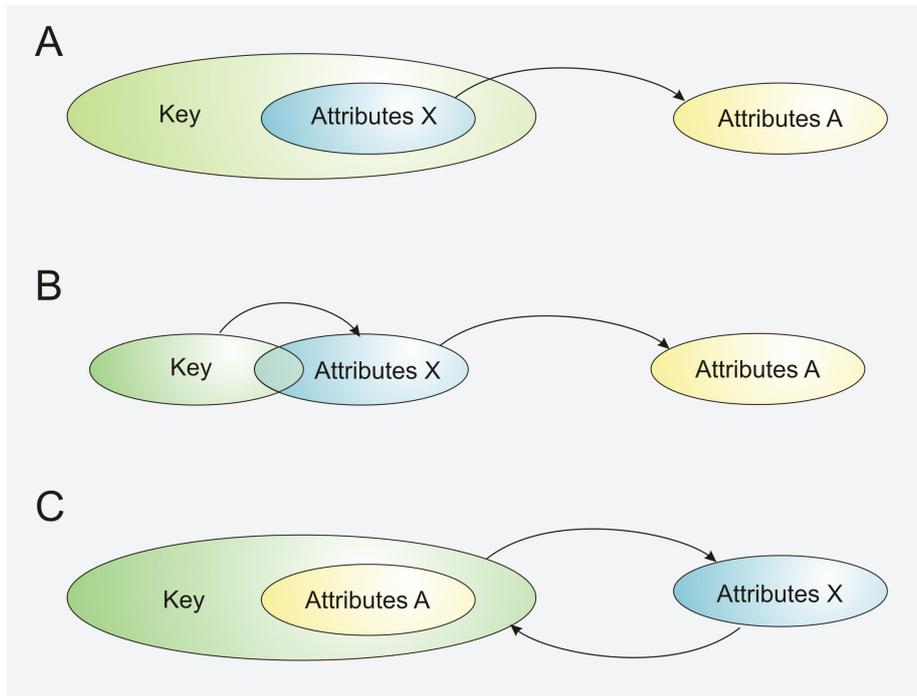


Figure 15: Partial and transitive dependencies. Figure adapted from [1]

- Decomposition: If  $\alpha \rightarrow \beta \cup \gamma$ , then  $\alpha \rightarrow \beta$  and  $\alpha \rightarrow \gamma$

Figure 15 shows two special dependencies playing a role in causing redundancy. Case A describes an example for partial dependency that occurs when a subset of a key (attributes  $X$ ) which itself is not a key anymore has a functional dependency for attributes  $A$ . As a consequence this represents the redundant information in the relation because for different keys  $A \rightarrow B$  might contain identical data. Cases B and C show two kinds of transitive dependency. In case B let us suppose that there is a functional dependency between  $X$  and the key. At the same time, there exists a functional dependency  $X \rightarrow A$ . Due to transitivity a functional dependency between the key and  $A$  can be inferred. A similar situation is shown in case C with the only difference that attribute set  $A$  is part of a key. Transitive dependencies lead to the same redundancy problem as shown above. In this example, the data in  $A$  might be stored redundantly in the relation.

**First normal form:** The first normal form is related to the concept of data atomicity. Every entry in all tuples in all relations contains a single value. The database cannot decompose these values into smaller pieces. However it does not mean that the value per se is indivisible. For instance a character string represents a single value and the database considers the string as a whole. Nonetheless a string can be decomposed into single characters. The same observation can be made with many other types. Integers can be split into prime factors and a data type consists of smaller entities such as day, month and year.

**Second normal form:** A relation schema violates the second normal form whenever it comprises data from different semantic realms. In mathematical terms a relation  $R$  is in second normal form if all non-key attributes are fully dependent on every candidate key in  $R$ . This means in particular that in 2NF there is no partial dependency.

**Third normal form:** Let  $X$  be an attribute set in  $R$  and  $A$  represents an attribute in a relation  $R$ .  $R$  is in third normal form, if for every functional dependency  $X \rightarrow A$  **at least one** of the following conditions is true:

- $A \in X$  (trivial FD)
- $X$  is a superkey
- $A$  is part of some key for  $R$

In other words, in third normal form there is no functional dependency  $A \rightarrow B$  where  $A$  only contains non-key attributes. In comparison to 2NF, 3NF does not allow transitive dependencies.

**Boyce-Codd normal form:** The Boyce-Codd normal form (BCNF) is similar to the third normal form but the requirements are stricter. A relation  $R$  is in BCNF if every functional dependency  $X \rightarrow A$  fulfills **at least one** of the following conditions:

- $A \in X$  (trivial FD)
- $A$  is a superkey of  $R$

The main point in BCNF is that every functional dependency is based on keys. Thus, any redundancy caused by FDs is avoided.

It is noteworthy to mention that the definitions not only cover single attributes for the right side of the functional dependency. Due to decomposition every functional dependency with a set of attributes on the right side can be transformed into FDs with single attributes.

For practical database design, violation of normal forms or denormalization may be a means to increase efficiency in certain situations. As an example, existence of redundant data in the database may facilitate formulation of faster queries to access this data.

### 2.3.3 Indices

In the preceding sections design aspects were covered for improving the consistency of a database allowing for an easier maintenance and extensibility. Additionally a good design also improves the computational performance of queries. A decisive improvement in performance, however, can be achieved by applying indices. An index is defined as a data structure for allowing a faster search for data in a database. An index can be represented by a search key such as a number or a character string. Note that a search key is unrelated to the concept of a primary key. The latter is always unique whereas the search key might be found several times during a query process. Whenever a user sends a request to a database, the query analyzer determines which relations to use in order to

compile the desired output. If the chosen relation(s) do not exhibit any indices the program examines row by row compiling the list of resulting tuples where the search criteria are applicable. Depending on the size of the relation this process may take only milliseconds or hours or longer. Even after application of design principles to reduce the amount of redundancy a relation may contain hundreds of thousands or millions of entries. The maximum size of a relation is only limited by the DBMS.<sup>1</sup>

Careful selection of indices is crucial for the technical aspects of a good database design. The crux of the matter is that indices only contribute to improved performance if the search criterion is based on attributes which are represented by indices. It does not make sense to create an index for every attribute because indices create an additional overhead and possibly computation time for maintenance. So the aim is to focus on few indices covering a wide range of possible query formulations. The search key for an index is not only based on a single attribute but can contain several ones. Such search keys are called composite search keys or concatenated keys. Figure 16 shows an example of a concatenated key. Here the index consists of the attributes protein

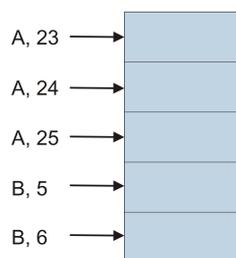


Figure 16: Example for a concatenated key consisting of two elements.

chain denotation and position number of the residue. Interestingly, such an index works for all queries containing prefixes of this key as search criterion. This means that protein and amino acid and protein alone are applicable but not amino acid alone. The reason is that with prefixes the order of the key is maintained which is not the case for non-prefixes. As a further advantage, if an query only retrieves data which are indexed, the query can be executed only in the index structure without examining the actual data which reduces the search time considerably. Such a query is denoted as index-only query.

In general, one can distinguish between a hash-based index and a tree-based index. They differ from each other with respect to performance of queries. In the following, the advantages and disadvantages of these index types are discussed.

Hashing uses a hash function  $h$  to map a key from  $S$  onto a bucket from  $B$ :

$$h : S \rightarrow B$$

$S$  represents the indices for all datasets of a database and  $B$  is the numbering of the available buckets. A hash-based index is illustrated in figure 17. Usually, the number of putative keys is much smaller than the number of possible data entries,  $|S| \gg |B|$ . Therefore, a bucket may contain more than one index.

<sup>1</sup>In the current version of MySQL, for instance, the maximum size per table is 65.536 Terabyte. In practice, this amount of space is (still) far more than required. Besides, most operating systems do not support such large quantities.

Hashing has a good performance for equality searches which means queries containing only one index value as search criterion because this only requires execution of the hash function plus searching the corresponding bucket. For range scans, which means a searching data entries fitting to a number of index values, this indexing type is not suitable at all. In this case, tree-based indices have a clear advantage.

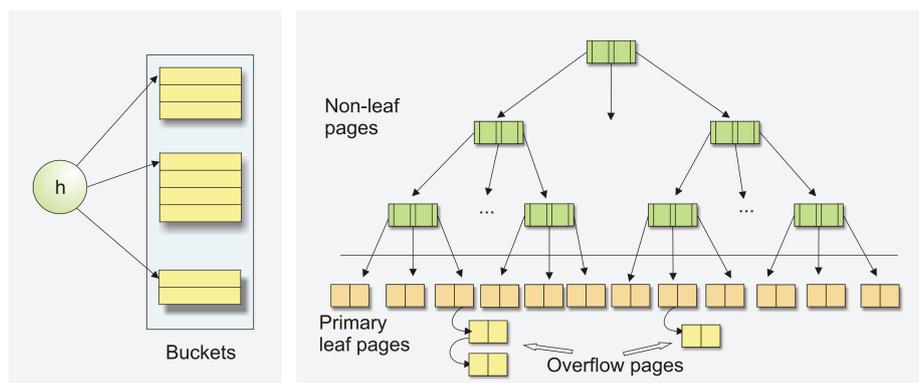


Figure 17: Indexing methods, left: Hashing, right: indexed sequential access method (ISAM).

The most common approaches for tree-based indices are the indexed sequential access method (ISAM)<sup>2</sup> and B+-trees. An example is shown in figure 17. The idea behind the indexed sequential access method is that the data and the index are located in different files. In particular, several data files are concentrated in one page. In the simplest case a single level index file points to the corresponding page of the data file. The index file is likely to be much smaller than the data file so that a binary search of the index file is more efficient than a binary search of the data file. A performance improvement is achieved by splitting up the index in a tree-like structure resulting in a multi-level index structure reducing the number of indices to be traversed to find the desired data. Only the leaf pages contain data. An important feature of ISAM is its static structure of the indices which means that once an index has been created it will not be changed. Any new data will be inserted into the data pages. If the maximum capacity of such a page is exceeded, a so called overflow page is created and linked with the existing primary page.

B+-trees also use a tree structure for the organization of the indices but in contrast to ISAM the structure may grow and shrink dynamically. The data entries are exclusively located in the leaf nodes. The search time for an entry is dependent upon the number of nodes that have to be traversed from root to the data entry. Therefore, it is desirable to keep the tree structure as flat as possible. For this reason, increasing the fanout, i.e. the number of children per node, is preferred over the splitting of nodes. Modification of the index structure becomes necessary during insertion or deletion of data entries.

The differences between ISAM and B+-trees also determine their usefulness in different application ranges. B+-trees generally show a better performance

<sup>2</sup>MySQL uses an improved version, called MyISAM [101]

in databases whose data is modified frequently due to the dynamic adjustment of the index structure. In ISAM, however, frequent updating of the data may result in more and more overflow pages if the existing slots are not sufficient. If data is kept unchanged in the database, ISAM may be faster as it does not apply time-consuming modifications of the index.

SQL command	Description
CREATE TABLE <table> (<attribute <sub>1</sub> -definition>, ... ,<attribute <sub>n</sub> -definition>)	The CREATE command defines a new table in the database. As parameters all attributes in this table are listed and their type (Integer, float, character,...) and special properties such as index assignment or primary key definition.
UPDATE <table> set <column> = <value> where <condition>	The update command changes the value for a certain attribute. If a condition is given only the tuples of the attribute are modified for which the condition is true.
INSERT INTO <table> (<column <sub>1n(&lt;value<sub>1n <td>Creating new entries is done with INSERT.</td> </sub></sub>	Creating new entries is done with INSERT.
ALTER <relation> {ADD   DROP   MODIFY }... <parameter>	The ALTER command modifies existing relations in a database. Numerous operations can be performed of which only the most important are listed here. Either a new column can be added (ADD) or an existing column may be deleted (DROP) or modified (MODIFY). Besides, ALTER also comprises operations such as renaming tables or dropping indices for columns.

Table 3: Overview of MySQL commands. Note that this list is incomplete. Other DBMS may have different syntax.

## 2.4 SQL

For managing and modifying a database several languages are available. The most widely used database language is SQL (Structured Query Language) which was introduced by IBM in the late seventies. SQL can be subdivided into the following parts:

- The Data Manipulation Language (DML): This subset comprises all commands to insert, delete and modify rows within database tables. Besides it allows the formulation of queries.
- The Data Definition Language (DDL): These commands allow the creation, deletion and modification of database tables.

- The Data Control Language (DCL): This subset refers to more technical issues. It provides mechanisms to control the access of tables and databases and for transaction management. In this context transactions are methods to ensure the data consistency. For instance a transaction regulates how data is updated when two or more users access the same data at the same time.

It is noteworthy to mention that SQL is not a real language but a standard. There are a number of database management systems using an implementation of the SQL standard. But in many cases there are differences between the implementation and the standard which means that not all features are included in the language. For MySQL which is the database management system used in this work a number of relational operators are missing such as `UNION`.

The most important SQL command is the `select`-command which is responsible for the retrieval of data. The following simple query example tells the database to look for a sequence of a protein having the PDB identifier 2PCC in the `protein` table:

```
select sequence from protein where pdb_id = '2PCC'
```

The result of this query is a table containing a single column with name 'sequence' comprising all sequences from the PDB structure with identifier 2PCC. A query covering more than one table is formulated as follows:

```
select * from table_1, table_2 where table_1.key = table_2.key
```

Here all attributes (\*) from `table_1` and `table_2` are retrieved. An overview of more SQL commands including a short description is listed in table 3.

### 3 The ABC<sup>2</sup> database

The ABC<sup>2</sup> database is the successor of the ABC database developed in my diploma thesis. The basic idea of this database is to provide a comprehensive data source for the analysis of features and properties of biomolecular contacts, in particular protein-protein and protein-small molecule interfaces at large scale. Currently, the database contains about 49,200 protein-protein and 37,900 protein-small molecule interactions. Interfaces are characterised by a number of features which are described in section 3.2. For in-house purposes, a user may directly access the database using SQL commands whereas for external users a web interface is available allowing for search, analysis and managing of interfaces. An automatic import function searches for new complexes in the RCSB database for continuous extension of the dataset. The following section covers the basic requirements of storing interaction data in a database.

#### 3.1 Preliminary considerations

The most important object for the database design is the interface, representing the contact between two biological objects. Consequently, in dimers we can observe at most one single interface, trimers with protein chains A, B and C may contain interfaces between chain A and B, A and C, and B and C. An interface may belong to a superior object. All current interfaces are derived from the RCSB database so that every interface can be assigned a PDB identifier. Conversely, an interface can be subdivided into smaller parts. For instance, an interface between two proteins consists of two separate protein chains. The chains are made of amino acids and the amino acids are made of atoms as smallest units in this consideration. The information assigned to an interface can be organized into two categories, see table 4:

- Structure-dependent data refers to any kind of data that can be extracted from the three-dimensional model of the complex. Examples are the surface of interface residues, distance information between amino acids, the volume of the gap between interacting objects and the number of contacts formed by residues.
- Structure-independent data refers to every kind of information that can be derived without knowledge of the structure such as the sequence of the protein chains or the functional characterization of the complex (with Gene ontology). This kind of information is typically obtained from sources other than the RCSB.

Clearly, different types of interfaces provide different kinds of information. For instance, one can formulate amino acids being in contacts with amino acids in a partner chain in a protein-protein interaction whereas in a protein ligand interaction there are contacts between amino acids and atoms of the small molecule. Due to lack of data or due to technical reasons an interface may not possess all information that it may have theoretically. As an example, conservation values for protein sequences are derived from the ConSurf-webserver in this work which are not available for all PDB chains. One reason is that the sequence length is too short for deriving a reliable conservation index.

Kind of data	Origin
Structure data	RCSB
Swissprot identifiers	Swissprot webserver
Conservation indices	Consurf webserver
GO ontology	GO webserver
SASA values	Naccess program
Empty space within interface area	Surfnct program
Circularity value	PCA analysis
Planarity value	Princip program
CATH data	CATH webserver
SCOP data	SCOP webserver
EC data	RCSB webservice
Obligate/non-obligate class prediction	NOX class
Obligate/non-obligate class assignment	Literature search
Kinetic values	Literature search
AA indices	Web resource

Table 4: Origin of data which are included in ABC<sup>2</sup> database.

### 3.2 Structure of ABC<sup>2</sup> database

The current version of the ABC<sup>2</sup> database contains 115 relations. A complete overview about the database diagram is available in the supplementary material, see figure 75 and 76. *MyISAM* was selected as storage engine for most of the relations as it offers good performance for fast searching of datasets. One reason for this is the lack of special features causing large overhead such as transaction management. However, these special properties are not required in the ABC<sup>2</sup> database.

The backbone of the ABC<sup>2</sup> is made of 4 relations as shown in figure 18. In this diagram, the meaning of these relations is clarified using a protein complex between chains A and B (the same holds for a protein-ligand interaction). An *abcEntity* entry stands for a certain interface between chains A and B. If A also participates into an interaction with another chain C, then a further *abcEntity* entry is created for representation of this interface even if there is an overlap of interface residues from AB and AC. The *interacts\_with* relation incorporates an ordering of the interaction and describes the contact between chain A and chain B as well as between chain B and chain A. An *interacts\_with* entry refers to n (more precisely 2) *participant\_ids* from *Participant* which represent the single structures from the RCSB file, in this case the amino acid chains. Eventually, a participant refers to one *bioUnit* entry which represents the structure-independent aspects of a participant. In the example, the *bioUnit* stands for the sequential information of the protein molecules. As the same sequence may occur in different interfaces or even in different PDBs, a *bioUnit* may refer to more than one *participant\_id*.

*abcEntity*, *interacts\_with* and *Participant* are closely related to structure data from RCSB. Figure 19 describes, how the PDB identifiers are assigned to these relations. There are three different types of identifiers. The *pdb*-relation stores the 4 letter code of the PDB structure and some more information concerning the structure. *pdb\_object* refers to the identifier of a molecule, either a single

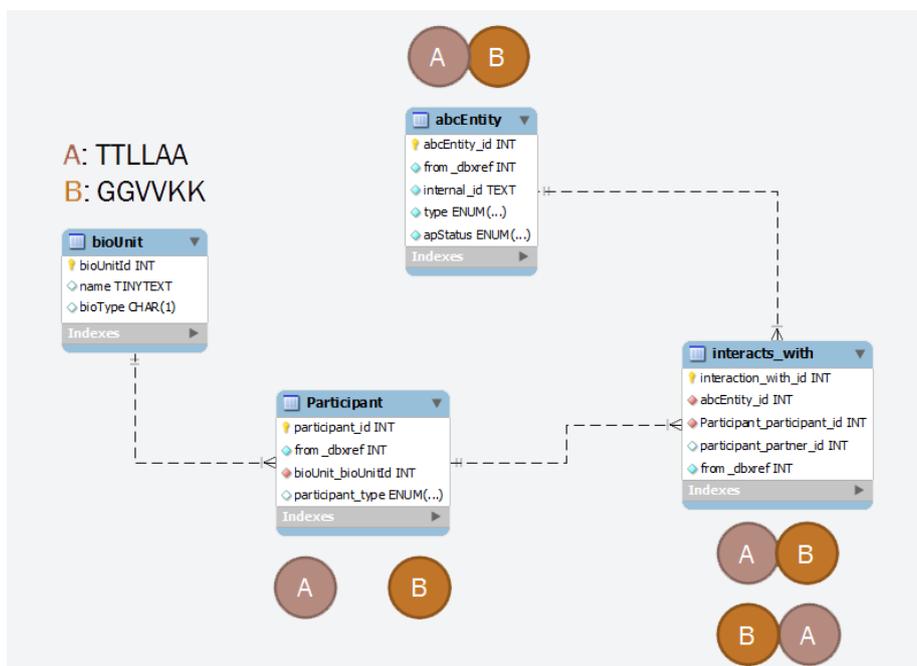


Figure 18: Basic relations of ABC<sup>2</sup> database.

character for an amino acid chain or a three letter name for a small molecule. A *pdb\_combination* entry contains both structure names identifying an interface. The PDB assembly identifies the structures that belong to one or more biological units. Additionally, there exist connections among the relations with each other. As an example, one structure or element from *pdb\_object* is contained in several *pdb\_combinations* whereas a *pdb\_combination* refers to  $n$  (or more precisely 2) entries from *pdb\_object*, which can be described as a  $n : m$ -relation between both relations. The *DBxref*-relation establishes  $n : m$  relations between the backbone relations and these identifier relations. *abcEntity* and *interacts\_with* are connected through *DBxref* to entries in *pdb*, *pdb\_object* and *pdb\_combination*. *Participant* is only connected to *pdb* and *pdb\_object*.

The central relation is the *abcEntity* which is responsible for the representation and identification of an interface (see figures 20 and 21). An interface belongs to a PDB structure which may contain more than one interface. It is noteworthy to mention that the same PDB with the same interface may occur twice or more often in the database with different *abcEntity* identifiers which happens when the same PDB file is stored in the database with different modifications. Such a case, for instance, is found in the OPM database offering modified versions of the original PDB (e.g. 1E54) [102]. This fact is taken into account by the *modding*-relation having an  $n : m$  relation with *abcEntity*, indicating the relationship among interfaces even though their *abcEntity\_ids* are distinct from each other.

An interface as a whole can be described with many characteristics that are stored in a number of relations that are associated with *abcEntity*. The following relations exclusively refer to protein-protein interactions:

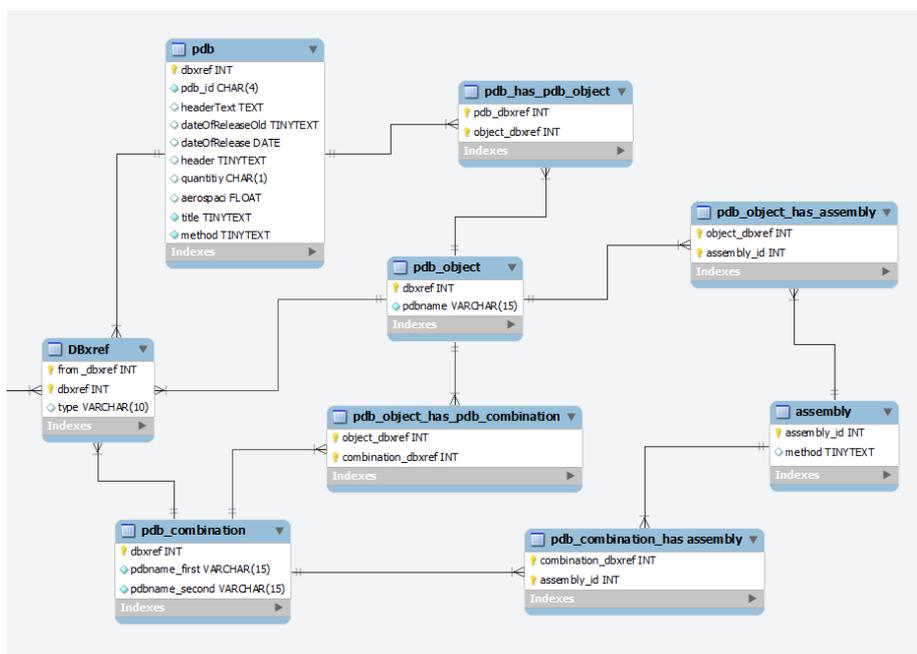


Figure 19: Connection between basic relations and RCSB identifiers.

- *NOXclass*: The NOXclass program [103] is based on a support vector machine and outputs a probability about the biological relevance of the interaction and the tendency to be obligate or non-obligate.
- *classStructure*: This relation reveals whether the interface is derived from two identical protein chains (homo) or different ones (hetero).
- *classObligateNonObligate*: For a small subset of about 500 interfaces the type of obligate and non-obligate interactions respectively was obtained from literature. These types are stored in this relation for the known cases.

The following features can be applied to protein-protein as well as to protein-small molecule interactions:

- *ifacePPConservationSurf*, *ifacePLConservationSurf*, *ifacePPConservationDist*, *ifacePLConservationDist*: The conservation score was derived from the Consurf webserver and was calculated as the average of all amino acids participating in a certain interaction using either the surface or distance-based interface criterion
- *ifacePPSurface*, *ifacePLSurface*: They contain the surface of the protein-protein interfaces and protein-small molecule interfaces, respectively. To this end, the surface area of the interface of every chain or small molecule was computed with Naccess [104] in its complexed and the unbound state. The values for two interacting structures A and B were applied in the following formula:

$$interface\ surface(\text{\AA}^2) = \frac{surface_A + surface_B - surface_{AB}}{2} \quad (1)$$

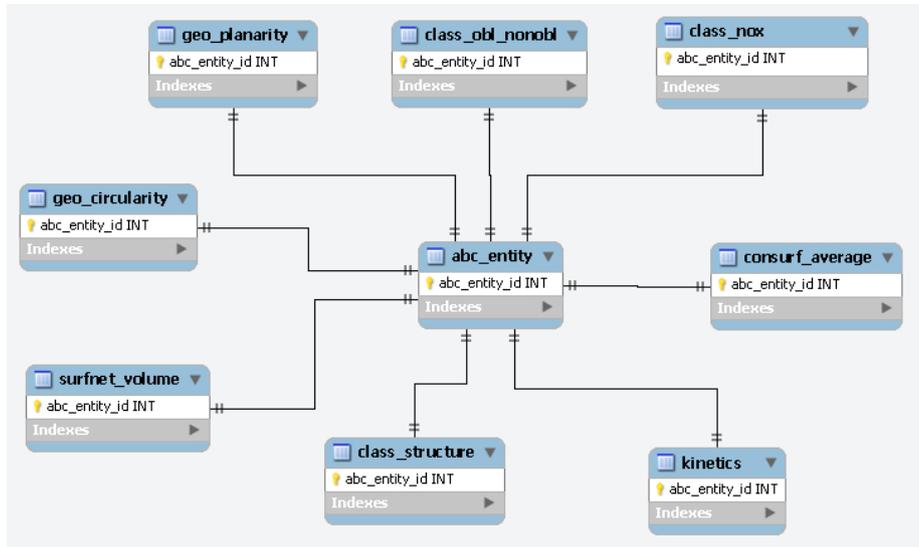


Figure 20: *abc\_entity* and associated relations.

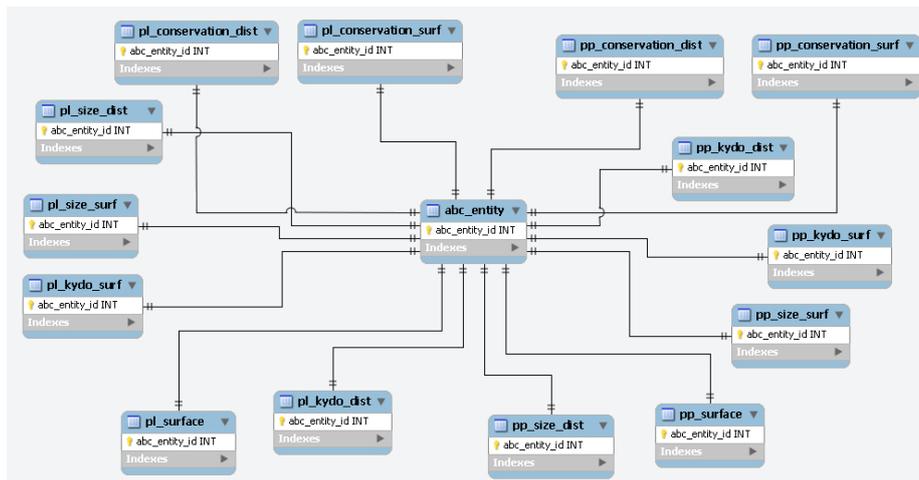


Figure 21: *abc\_entity* and associated relations containing interface statistics data.

- *ifacePPDistKydo*, *ifacePLDistKydo*, *ifacePPSurfKydo*, *ifacePLSurfKydo*: The hydrophobicity score derived with the Kyte and Doolittle scale [105] describes the hydrophobic character at the interface area [105]. Every amino acid is assigned a value reflecting its hydrophobicity. The more hydrophobic the molecule is the lower is the value. For every interface the score was calculated as the average of all its interface residues determined either with the distance-based or surface-based interface criterion. Beside the raw score, also hydrophobicity scores which were normalized against the surface of the residues are available. Please note that for protein-ligand interactions the score is calculated for the protein chain only.
- *surfnetVolume*: The volume of the gap between interaction partners is calculated with the program surfnet [73]. A normalized value, which is denoted as gap volume index was obtained by the following formula:

$$gap\ volume\ index(\text{\AA}) = \frac{gap\ volume(\text{\AA}^3)}{interface\ ASA(\text{\AA}^2)} \quad (2)$$

- *geoCircularity*: The circularity value gives an estimation how close or how distinct the interface shape is from being a perfect circle. To this end, a principal component analysis (PCA) was performed for all amino acids that are located at the interface. As standard the distance-based criterion is used with a 5Å cutoff range. The first two eigenvalues of the PCA analysis reflect the lengths of the axes exhibiting the largest extent in the interface structure. The circularity score is obtained by dividing the eigenvalues. It is noteworthy to mention that due to the descending order of the eigen values the result is always greater or equal to one. A score equal to or near one reveals that the axes have about the same length and thus the shape is close to a circle. Analogously, the more distinct the value is from one, the more elliptical the shape of the interface area is. The so called eccentricity is calculated as follows:

$$eccentricity = \sqrt{1 - \frac{breadth^2}{length^2}} \quad (3)$$

Breadth and length of the interface section are the first two eigenvalues from the PCA. The eccentricity has a value range between 0 (perfect circle) and 1 (straight line) allowing a better handling than the usual circularity value.

- *geoPlanarity*: Another value to characterize the overall interface shape is the planarity which describes whether the area is flat or whether the outlook is more jagged. Similar to circularity, a score is calculated based on a PCA of the interface atoms. The plane which is spanned by the first two eigenvectors represents a section through the interface. For every atom above and below that plane a perpendicular is dropped and the RMS out of these distances was calculated. The larger the value, the less planar the interface area is.

Further functionalities refer to similarity measures in interfaces. For protein-small molecule complexes, a ligand in its SMILE representation can be queried

0-bind paths	C	O	N
1-bind paths	OC	C=C	CN
2-bind paths	OC=C	C=CN	
3-bind paths	OC=CN		

Figure 22: Fragmentation of SMILE string bottom up. In this example, oxoacetonitril is decomposed into substructures. Each layer contains fragments of same bond length.

against all other PL complexes to find the ones exhibiting a similarity according to a given similarity range. Internally, the similarity is calculated using the CDK package [106]. To this end, the SMILE strings, which are compared with each other, are gradually decomposed into substructures until the atomar level, also see figure 22 [107]. For two compounds  $A$  and  $B$ , the number of identical and non-identical fragments are inserted into Tanimoto equation [108]:

$$Tanimoto = \frac{Fragments_{smile\ A} \cap Fragments_{smile\ B}}{Fragments_{smile\ A} \cup Fragments_{smile\ B}} \quad (4)$$

A value near 1 refers to highly similar smile strings, a value closer to zero means large dissimilarity. Even though this approach focuses on structural connections and does not take into account factors like bioisotery, the method is assumed to perform quite well for comparison of small molecules.

For similarity of protein-protein interfaces, the following approach was implemented. The sequences of two given protein chains  $P_1$  and  $P_2$  were restricted to residues that participate into an interaction, as illustrated in figure 23. Then, a sequence alignment of interface residues was made. A match is found if an alignment position in  $P_1$  and  $P_2$  contains the same amino acid. The number of

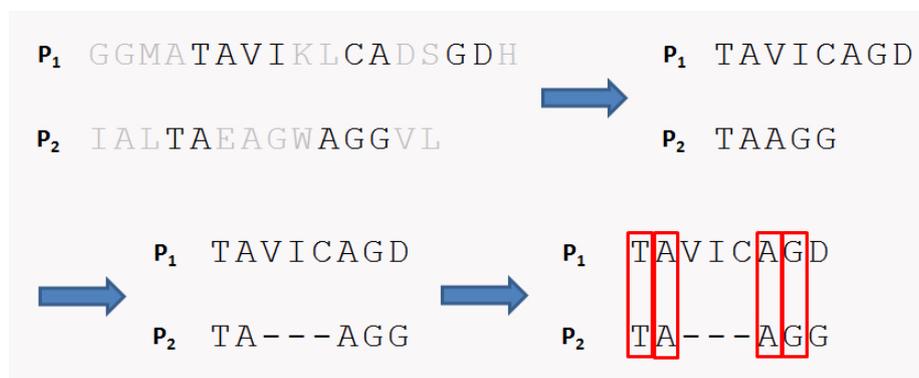


Figure 23: Non-interface residues (shown as grey letters) are excluded from the original sequence. The interface sequences from protein chains  $P_1$  and  $P_2$  are aligned with each other. The number of matching amino acids, here indicated in red boxes is counted for calculation of the fingerprint score.

matches and the number of mismatches between protein chains  $P_1$  and  $P_2$  was used to calculate a fingerprint score which is based on the Tanimoto coefficient:

$$T(P_1, P_2) = \frac{\# \text{ interface residues matches between } P_1 \text{ and } P_2}{\text{length of interface sequence alignment}} \quad (5)$$

Eventually, the similarity score for two protein complexes  $P_aP_b$  and  $P_cP_d$  was calculated as:

$$sim(P_aP_b, P_cP_d) = \frac{max(T(P_aP_c), T(P_aP_d)) + max(T(P_bP_c), T(P_bP_d))}{2} \quad (6)$$

As mentioned in the preliminary considerations, an interface represents a binary interaction between two biological objects. The *interacts\_with* relation constitutes the basis for all binding information of interfaces and contains the identifiers of the structures that are involved in an interaction. *abcEntity* and *interacts\_with* are connected via a 1 :  $n$  relationship (in this special case  $n$  equals 2, the reason for this is mentioned below). An entry in *interacts\_with* describes one biological structure with the *Participant\_participant\_id* attribute whereas the other structure involved in the interaction is represented as *participant\_partner\_id*. So the order of the interaction partners is fixed. Another entry describes exactly the same interface from the other way round which means the participant identifiers are swapped. This explains why in this relationship  $n$  equals 2. The *Participant\_participant\_id* and the *participant\_partner\_id* attributes refer to the *Participant* table that stand for a single structure within a PDB. *Participant* refers to at least one entry in *interacts\_with* and *interacts\_with* refers to exactly one entry from *Participant*. At first glance this appears confusing as *interacts\_with* contains two different participants but formally only the attribute *Participant\_participant\_id* is the foreign key from the *Participant* relation.

An interaction between two participants can be described either as a distance-based or a surface-based interaction. The definitions were given in the first chapter. In the current release of ABC<sup>2</sup>, two kinds of interacting pairs are considered, namely interactions between atoms and interactions between amino acids. It is evident that the latter case is a kind of generalization of the former case. Thus, protein-protein interfaces can be represented either as interactions between single atoms or as interactions between amino acid residues whereas protein ligand interfaces can only be described as atomic pair interactions.

An interaction at atomic level is described in the database by the following relations. An interface from *interacts\_with* refers to its interacting atoms from *atomDistance* which are denoted as *atom\_id* and *partner\_atom\_id*. The former is located on the structure which is represented as *Participant\_participant\_id* and the latter is located on *participant\_partner\_id*. The *atomDistance*-relation contains the id for an atom originating from one PDB structure and its partner atom from the other molecule including the distance between both atoms in Å. The *atom\_id* refers to the corresponding entry from *atom* which contains the denotation of the atom and its coordinates from the PDB file. The attributes *small\_molecule\_atom\_atom\_id* and *aminoacid\_id* determine whether the atom is originated from an amino acid or from a small molecule. The actual type is assigned an identifier whereas the other type is set to 'NULL'. Depending upon the type, the identifier refers to *aminoacid* that stands for a unique amino acid in the protein chain or to the *small\_molecule\_atom* relation which represents an atom from a small molecule in the interaction. The scenario for an interaction which is based on the distance criterion is shown in figure 24.

Figure 25 illustrates an interaction which is based on the surface based interface criterion. In comparison to the distance based definition the *atomDistance* relation is replaced by the *surfaceAtomContact* relation. It comprises all atoms

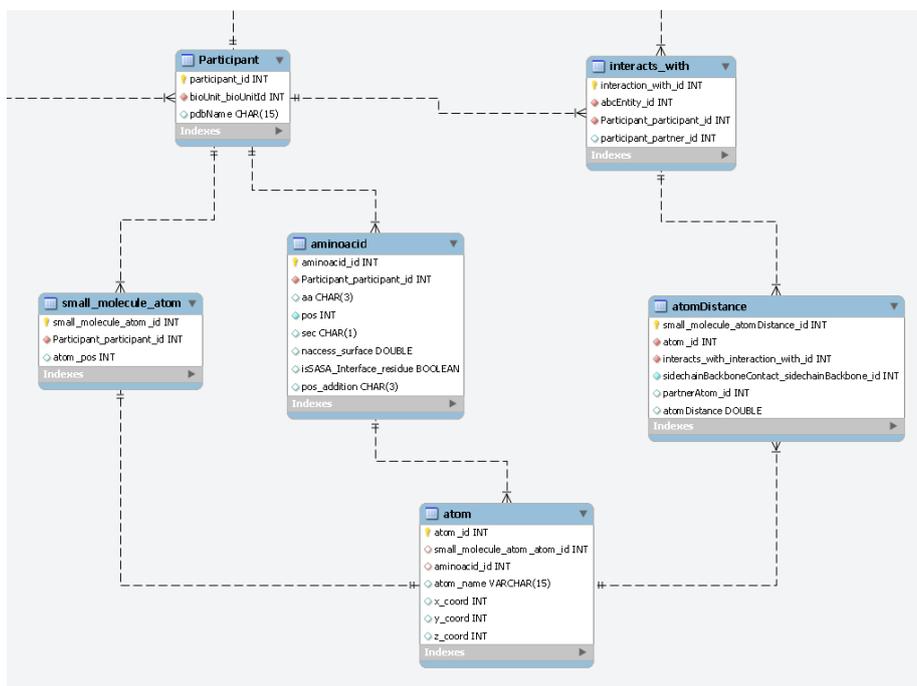


Figure 24: Relations describing an interaction based on atom contacts according to the distance criterion.

for which a loss in SASA occurs upon complexation. The relationships of the remaining relations are the same as with the previous interaction type.

The schema for interaction representation holds for protein-protein as well as for protein-ligand interaction. For protein-protein interactions exclusively there are two further relations describing the interaction between amino acids by means of distance-based and the surface-based interface criterion. As the dataset size of these relation is much smaller in comparison to the atomic interactions, performing queries of this kind of interface representation requires far less computation time.

As just mentioned, the relation *interacts\_with* also serves as starting point for the description of interactions between residues. The participant from this relation refers to one or many different residues from *residue\_residue\_contact*. One entry in *residue\_residue\_contact* represents an interaction between an amino acid that is located on the chain from *interacts\_with* (identified by the attribute *Participant\_participant\_id*) and the interacting amino acid (*partner\_aminoacid\_id*) from the partner chain (*participant\_partner\_id*). The interaction is specified in the *residue\_residue\_contact\_data* relation. It contains the information how many atom pairs between the interacting amino acids lie within a defined range.

In the context of amino acids, three further relations are subjects of interest. *AAIndex* assigns to every of the twenty amino acids numerous different values which were derived from AAindex database [109]. The indices represent various physico-chemical and biochemical properties of amino acids. Figure 26 illustrates how this information is stored in the database. The *AAIndex*-relation is the base relation for an amino acid index. It refers to twenty entries from

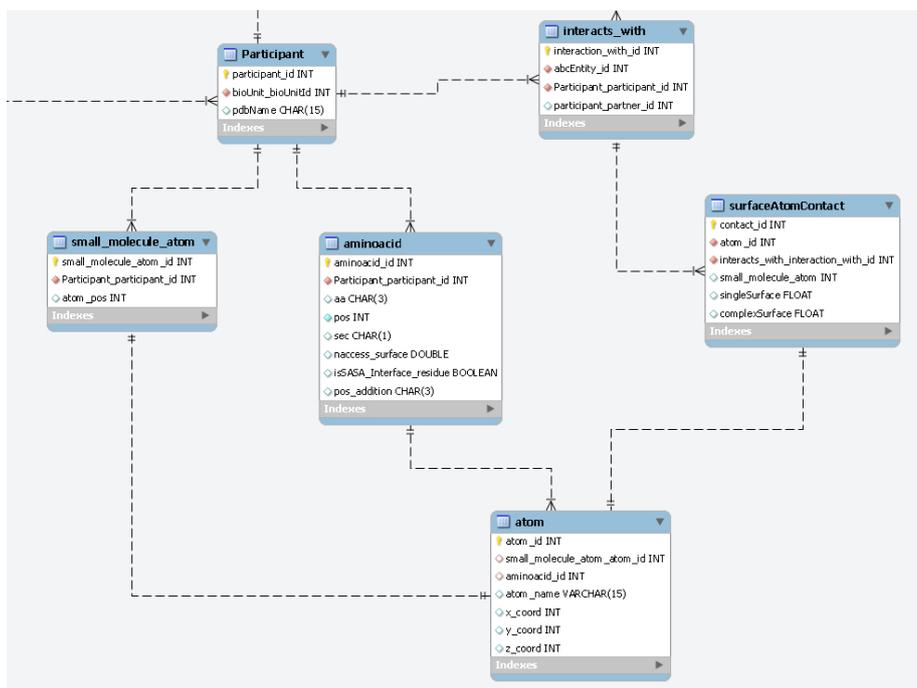


Figure 25: Relations describing an interaction based on atom contacts according to the surface based criterion.

*AAIndexValues* that contains the assignment of a score to each amino acid. *AA* indices can be referenced among each other. If references for an *aa* index are available they are stored in the *AAIndexCorrelation* relation.

Additionally, an amino acid can be characterized by its pharmacophores which are detailed in section 5 (see methods). The assignment of a pharmacophore group is dependent upon the kind of amino acid and the atom type. Both attributes are contained in the corresponding relation shown in figure 27 and refer to their foreign keys in the *aminoacid* and *atom* relation.

The *Participant*-relation stands for a molecule that is involved in an interaction. In the current version of ABC<sup>2</sup> this can be either an protein/peptide chain or a small molecule or ligand respectively. As a certain participant may be involved in interactions with many other participants, the *Participant*-relation is connected with *interacts\_with* via a 1 : n relationship. A participant is dependent upon the structure of the molecule. In particular, this holds for protein/peptide chains for which the following subordinate relations of *Participant* contain further information:

- *consurf*: The *consurf*-relation comprises conservation data for every residue of a protein chain. Conservation data was derived from the consurf-webserver. The data serves as the source for the conservation values stored in the relations that describe conservation features of interfaces.
- *naccessLocation*: It provides a summary of the location of any kind of residues. In this context, the location of a residue is defined as being on

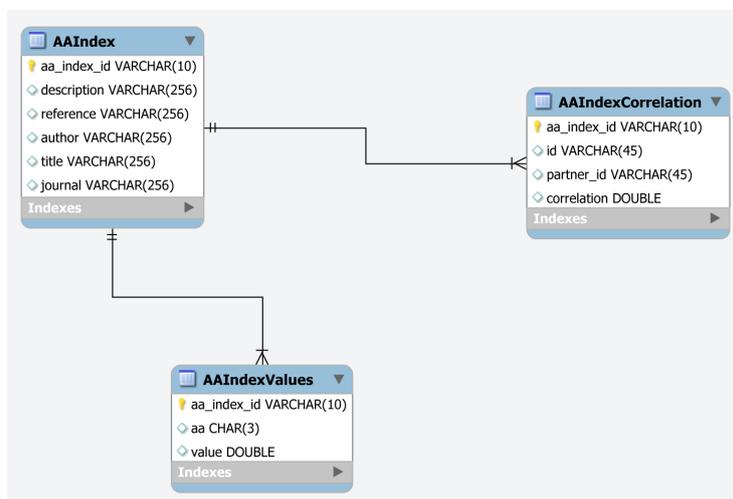


Figure 26: *AAIndex* defines *aa* indices that assign scores to any of the twenty amino acids.

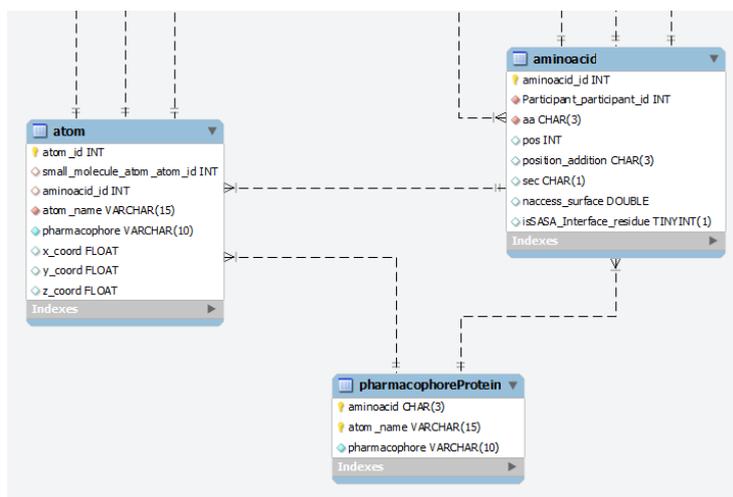


Figure 27: Pharmacophore group assignment for amino acids.

the surface or being in the core. The assignment is done with *Naccess* that calculates the SASA-value for every amino acid. A SASA value greater than zero indicates that the corresponding amino acid is located at the surface whereas a value equals zero reveals that the residue is buried in the interior of the molecule. One row in *naccessLocation* contains the numbers of occurrence for every type of residue to be located either at the surface and the interior of the participant respectively. Every participant is assigned exactly two entries: One for the residues that are located at the surface of the interface and one which lists the number of buried residues.

- *participantSequence*: In this relation, the sequence of the entire protein chain including position numbers is listed. It is noteworthy to mention that due to modifications the sequence in the structure does not necessarily match the corresponding swissprot sequence which is encoded by a *bioUnit* identifier.
- *proteinParticipant*: The relation provides details about the numbering of sequence positions for a *Participant* for which different standards may exist in a PDB file.

As the name reveals, a participant is involved in a certain interaction which is identified by the *abcEntity\_id*. The participant is related to the *bioUnit*-relation that represents structure-independent data. The *bioUnit* relation focuses on all aspects that are common to a group of participants. Therefore, a participant is always bound to a certain PDB structure whereas a *bioUnit*-object may refer to several different participants from one or several PDBs. A *bioUnit* acts as superordinate entity for three different concepts, either an amino acid chain or a small molecule or a nucleotide, see figure 28. These are described in the same named relations *aa\_chain*, *small\_molecule* and *nucleotide*. The latter relation is meant for later implementation. The difference between a *bioUnit* and its corresponding *Participant* consists in structure-specific aspects. For instance, some residues of the structure were enzymatically cleaved or one or more amino acids are mutated. However, the swissprot identifier remains the same as the overall sequence is still the same.

*aa\_chain* is a general representation for an amino acid chain and can be further classified as either a protein or a peptide relation. In this context, an amino acid chain is considered as protein if a swissprot identifier can be found for the sequence, else it is defined as peptide. Besides, *aa\_chain* has connections to further relations characterizing polypeptidic sequences, see figure 29. These comprise assignment of EC and GO classification as well as a mapping of homologous sequences for a given sequence for fast access of homologous sequences.

The enzyme classification scheme (EC) has a tree-like hierarchy. As an example, the EC-code 2.1.1 stands for transferase (code **2**), which transfers one-carbon groups (code 2.1) and is a methyltransferase (code 2.1.1). The tree structure is mapped into tables as follows: A *bioUnit* entry may have several different EC-codes. The three constituent parts of this code (see the bold digits in the example above) are represented in *bioUnit\_has\_EC* with internal identifiers. *ec\_class\_id* stands for the first component, *ec\_subclass\_id* for the second one and *ecsubclass\_id* for the last one. Any of these components refer to an entry in *ec\_tree* that stores the actual number of the EC code component and

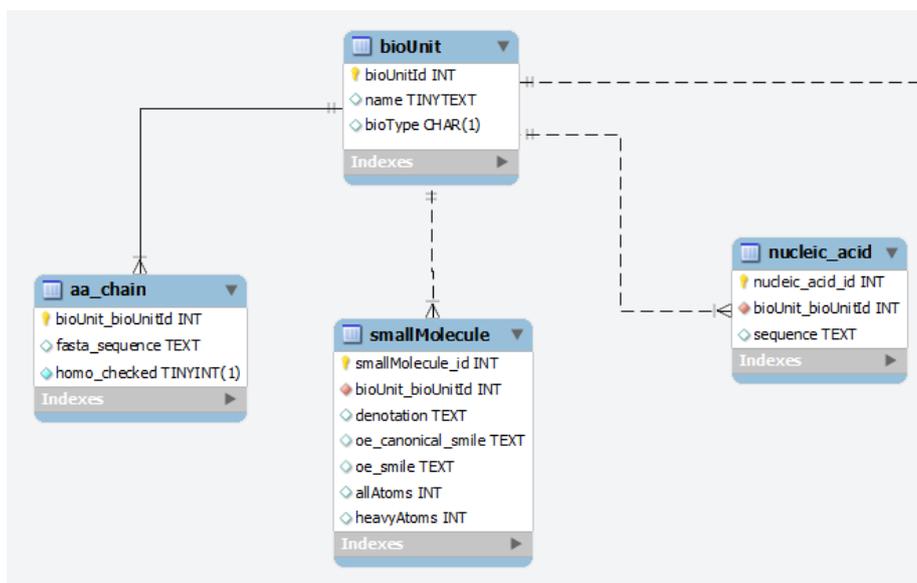


Figure 28: A *bioUnit* entry either represents an amino acid chain, a small molecule or a nucleotide.

contains the identifier for the higher level component. If the current component already stands for the root, the *parent\_id* is NULL. *ec.tree* allows descending the tree by following the *ec.tree\_child\_id* starting from a *parent\_id* or climbing by following the *parent\_id* starting from a *ec.tree\_child\_id*.

Gene ontology (GO) describes a sequence with respect to its functional meaning, its occurrence in biological processes and its location in cellular components. Figure 30 details how the hierarchic structure is modeled in ABC<sup>2</sup> database. The relations *bioUnit*, *GO\_has\_bioUnit* and *GO* form an  $n : m$  relationship meaning that a *bioUnit* entry may refer to several *GO* entries and vice versa. The *GO* relation contains the GO identifier (a name starting with 'GO:' followed by a six digits with leading zeros) and the GO ontology origin (either 'molecular function', 'biological process' or 'cellular component'). The GO-tree is encoded by the relations *term* and *term2term*. A term entry stands for a certain GO identifier and stores its full denotation. It is associated with the *GO* relation via the *acc*-attribute acting as foreign key. A *GO* entry may have children in the tree unless it is not a child node. All children of a term are listed as *term2\_id* attribute in *terms2terms*. Another  $n : m$  relation is formed between the tables *GO\_has\_bioUnit*, *GO\_has\_bioUnit\_has\_GO\_evidence* and *GO\_evidence*. The *evidence*-attribute refers to the data source of the GO assignment.

### 3.3 Examples of SQL queries

In this subsection, some examples of typical SQL queries are given that are used for the ABC<sup>2</sup>-database. To begin with, a simple query for accessing the surface size of a protein interface is formulated. Here, the *surface* attribute from the relation *ifacePPSurface* for a certain interface is provided that is identified through its *abcEntity\_id*.

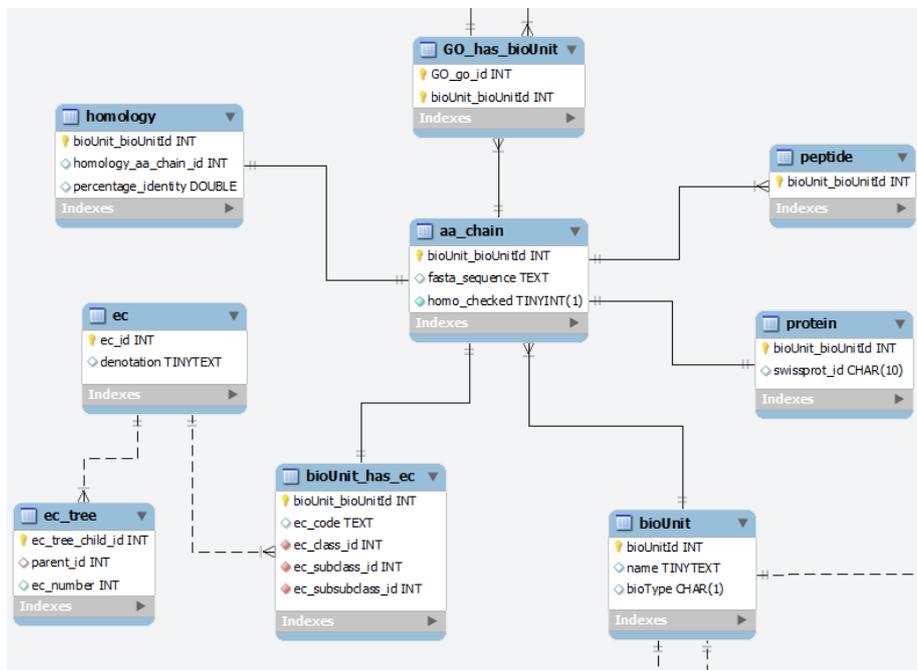


Figure 29: The aa\_chain is the central relation for sequence-based information.

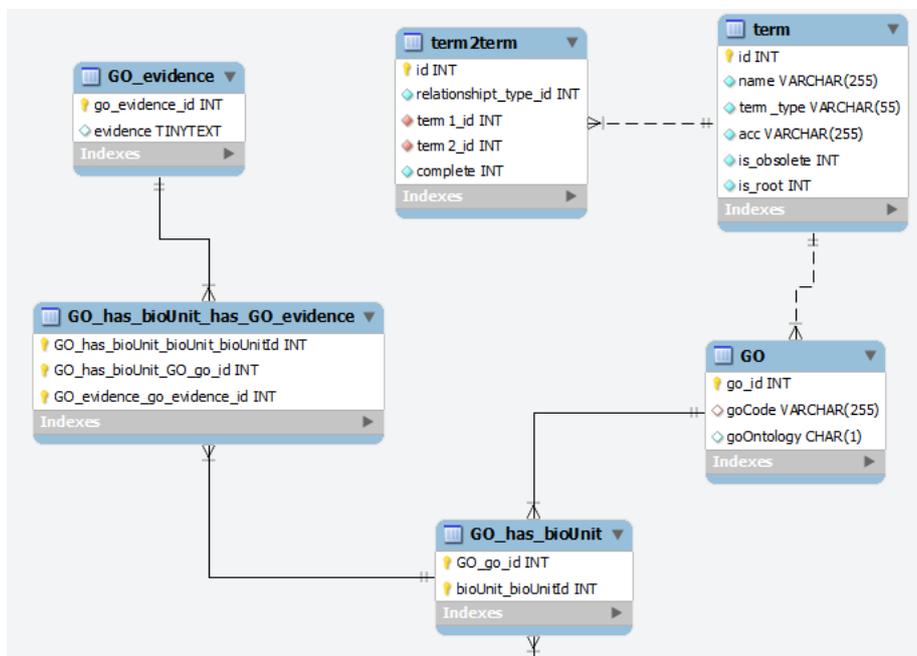


Figure 30: Relations covering Gene ontology data including tree-like ordering of GO-terms.

```
select surface from ifacePPSurface
where abcEntity_id = 1234
```

Listing 1: SQL query for getting surface size of an interface.

One of the most important types of queries refers to the search for interaction data either between proteins among each other or a protein with a small molecule. For instance, the participating interface residues for a certain *abcEntity\_id* is compiled with the following command:

```
select r1.aa as aa1, r1.pos as pos1, a1.atom_name as
    atom1, r2.aa as aa2, r2.pos as pos2, a2.atom_name as
    atom2
from interacts_with, atomDistance, atom a1, aminoacid
    r1, atom a2, aminoacid r2
where interacts_with.interaction_with_id =
    atomDistance.interacts_with_interaction_with_id
and a1.atom_id = atomDistance.atom_id
and r1.aminoacid_id = a1.aminoacid_id
and atomDistance.partnerAtom_id = a2.atom_id
and a2.aminoacid_id = r2.aminoacid_id
and abcEntity_id = 1234 and interaction_with_id = 4321
and atomDistance <= 5 order by r1.pos, r1.aa
```

Listing 2: SQL query for accessing contact data based on distance criterion.

This query follows the distance-based interface criterion. To this end, several relations are joined. The *atom* and the *aminoacid* relation occur twice with different names (*a1*, *r1* and *a2*, *r2* respectively) and represent the amino acid and the atoms for the protein chains that interact with each other. The *interaction\_with* relation characterizes the interaction by the order of its participants whereas the actual interface is identifier by the *abcEntity\_id*. *atomDistance* is required to determine the contacting atoms between the protein chains. Besides, it defines the distance limit (in this case 5Å). The output consists of amino acid denotations including positions from one protein chain (*r1.aa*, *r1.pos*) and the analogous data for the binding partner (*r2.aa*, *r2.pos*).

The next SQL command demonstrates the usefulness to distinguish molecules according to structure-dependent and structure-independent properties. Here, all PDB identifiers are accessed exhibiting protein chains having the uniprot identifier P00044. It is noteworthy to mention that due to the uniqueness of the key *bioUnit\_bioUnitId* the relations *protein* and *Participant* can be joined directly without incorporating the relations *aa\_chain* and *bioUnit*.

```
select pdb_id from protein, Participant, DBxref, pdb
where protein.bioUnit_bioUnitId =
    Participant.bioUnit_bioUnitId
and Participant.from_dbxref = DBxref.from_dbxref
and DBxref.dbxref = pdb.dbxref
and swissprot_id = 'P00044'
```

Listing 3: SQL query for finding all PDBs containing protein chains with uniprot-ID P00044.

Besides, for easier access of data, in particular PDB identifiers, several views were defined. The following example creates a view that gives an overview of all chain identifiers that are assigned to a PDB. The following command creates a relation *pdbchains* with attributes *PDB* and *chain*.

```
create view pdbchains as
select pdb.pdb_id AS PDB, pdb_object.pdbname AS chain
from pdb, pdb_has_pdb_object , pdb_object
where pdb.dbxref = pdb_has_pdb_object.pdb_dbxref
and pdb_object.dbxref = pdb_has_pdb_object.object_dbxref
```

Listing 4: SQL command defining a view.

### 3.4 Implementation

The main implementation was done using Java. Figure 31 gives an overview of the class packages containing the code for website layout and management as well as the code for database management, query handling and import functions. Besides, the package also comprises some packages for bioinformatic-specific algorithms. The code for the website can be subdivided according to the model-view-controller (MVC) framework [110] into classes representing the controller, classes acting as views and classes which are called model and contain the actual logic. In addition to self-generated packages, a number of external Java libraries were used such as BioJava [111] providing functions such as parsing of PDB files. Beside Java code, scripting languages were applied for minor tasks such as Python and Tcl/Tk. Eventually, external programs such as Naccess [104] were run under Java using a library for execution of system processes.

#### 3.4.1 Data import

The design of the import functionality was driven by the following considerations. First, due to the large number of different methods and techniques that have to be applied for importing data from different sources, the framework should allow a clear organisation of functionalities which also supports code reusability. Second, it has to be flexible allowing for easy modifications and extensions. Third, it is also desirable to have to proper organisation of error handling and logging.

The implementation covering these features is based on a composite design pattern. Figure 32 shows the basic classes. As the name implies, *APWorkflow* represents a class for managing one or more processes which are in turn *APWorkflows* and *APTasks* respectively. An *APTask* stands for a single job to be executed. Such a framework allows for a tree-like hierarchical order of classes. *APWorkflow* classes can be considered as node within this tree whereas *APTask* classes represent the leaves. An example workflow code is detailed in listing 5. The *MyWorkflow* instance obtains a reference to an instance of *DataContainer* which carries relevant data for processing and is responsible for exchanging data among other workflow and task classes. The start method executes the functionality of the class. Here, the class compiles a list of further workflows and tasks all having the type *APComponent* in an array. Finally, the instances are started one after the other. Such a design allows for a well structured organisation of processes which can be easily extended just by inserting new classes in

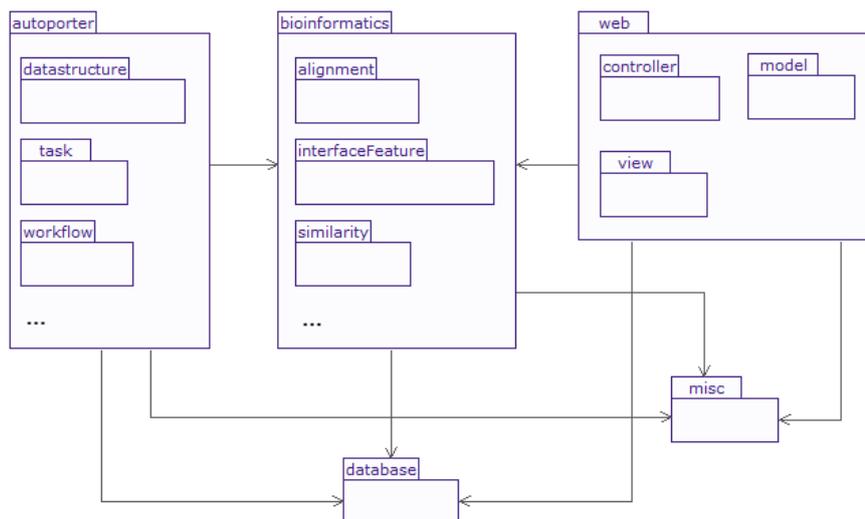


Figure 31: Overview of class packages. The *autoporter* package refers to all classes that are responsible for import of data into the database from external sources such as the RCSB. A more detailed description is given in subsection 3.4.1. The *web* package comprises all classes with reference to the website of ABC<sup>2</sup> database. It contains three important sub-packages which are designed in accordance to the MVC framework which is explained in subsection 3.5. The *bioinformatics* package harbours code for tasks like alignment or tools for the calculation of similarity scores between SMILES strings. Both *autoporter* and *web* require functionality from this package which is indicated by dependency arrows in the diagram. Database comprises routines for establishing a database connection, management of connection pooling and query handling. In this context, MySQL provides the so-called JDBC class library which handles any communication between a MySQL server and Java. *Misc* contains code for common tasks such as execution of system calls or classes for string formatting. Classes from *bioinformatics*, *Misc* and *database* packages were also used for the projects from sections 4-6.

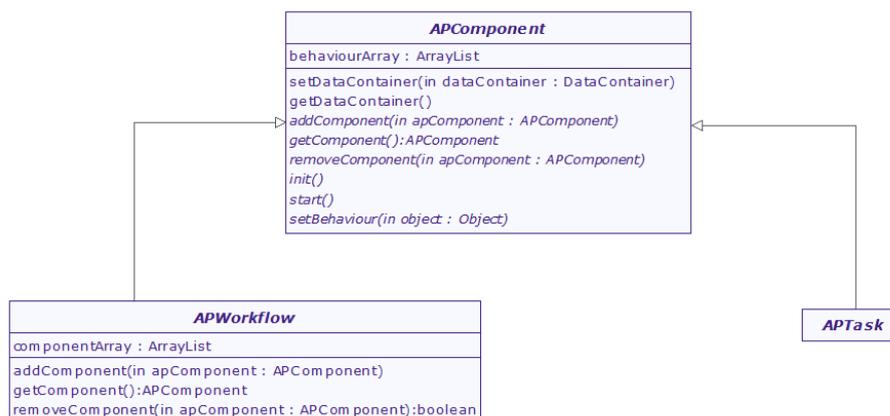


Figure 32: Basic classes for the import process.

the array. Besides, it facilitates easier application of logging and error handling mechanisms.

```

public class MyWorkflow extends APWorkflow {

    public MyWorkflow (DataContainer dataContainer) {
        super(dataContainer);
    }

    public void start() throws Exception {
        List<APComponent> components = new
            ArrayList<APComponent>;
        components.add(new Task(dataContainer));
        components.add(new Workflow(dataContainer));
        components.add(new Task(dataContainer, new
            ModifiedBehaviour()));

        for (APComponent component : components) {
            component.start();
        }
    }
}
  
```

Listing 5: Code example for a workflow class

The hierarchical structure also allows a dynamic compilation of classes according to the requirements of the import process. For instance, the user only wants protein-protein data to be imported, so only classes dealing with protein-protein interaction data are instantiated.

As an example, the main import process for biomolecular data is shown in figure 34. Here, the workflow *ImportBiomolecularContacts* is the starting point for importing biomolecular contact data into the database and manages all subsequent processes. Initially, the *RCSBReader* is executed establishing a connection to RCSB database through a webservice and accesses a list of PDB

structures that may contain biomolecular contacts. As the number of PDBs and corresponding interfaces might be large, the *DataManager* class is responsible for separating the data into smaller pieces that are processed in one iteration to prevent extensive usage of memory. *DataMapper* then converts the information read from the PDB into an internal data representation, which is shown in figure 33. Additionally, *DataFilter* generates all possible binary combinations of elements in the PDB structure. Referring to figure 33, this class creates the data for *PDBCombination* and *PDBInteractsWith*. As a first filtering step, the exist filter tries to find all interfaces or PDBs that are already available in the database and excludes these cases from further consideration.

The actual import is subdivided into three large workflows. *BasicImport* refers to a minimum set of data, which is required for an interface to be accepted as an entry in the ABC<sup>2</sup> database. As can be seen in figure 35, the workflow can be further subdivided into workflows that handle the import of PP interfaces as well as PL interfaces and can be easily extended with further biomolecular contacts. Given the PP import, the initial step consists in generating contact data using the *PPDist* workflow which is based on the distance-based interface criterion. Using the contact information, a filter task decides whether the corresponding interface is valid or not. In the former case, the subsequent workflows and task respectively write the data in the database tables. *ElementData* is responsible for all information referring to the single molecules with respect to structural and structure-independent features, *PPData* incorporates all interface data, among them the storage of the contact information, *RCB-SData* contains information about the PDB structure, which is read from the RCSB database in XML file format [112]. *AssignAssembly* assigns molecules to their biological units if more than one are found in the PDB file.

After completing the import of basic data for a new interface, *ExtendedImport*, which is shown in figure 36, generates and adds further data. Here, there are conceptual differences from the previous workflows. Beside workflows for PP and PL complexes, further subworkflows are implemented dealing with *BioUnit* or structure-independent data and with *Participant* or structure-dependent information. Additionally, the ordering of the present subworkflows is arbitrary.

Eventually, *DerivedImport* covers all relations storing data which is derived from data from existing relations in the database. Thus, no external source is required in this context. Mainly, the workflow generates statistical data such as average consurf values for entire interfaces and stores them in individual relations for faster access. Besides, the workflow also provides for PP complexes contact information among residues which is compiled from the relations containing atom contacts. On average, the *residue.residue.contact* relation only consumes 5% of the space in comparison to the *atomDistance* relation facilitating faster access of protein-protein contact data.

### 3.4.2 Deleting and updating data

Deletion of data is related to the import process because the same design principles and codes from there are used. In principle, removing data from the database refers to deletion of an interface. Removing a PDB can then be considered as the deletion of all its interfaces. Eventually, cleaning of the entire database is achieved by removing all PDBs.

Figure 37 depicts the workflow for deleting an interface. An interface to be

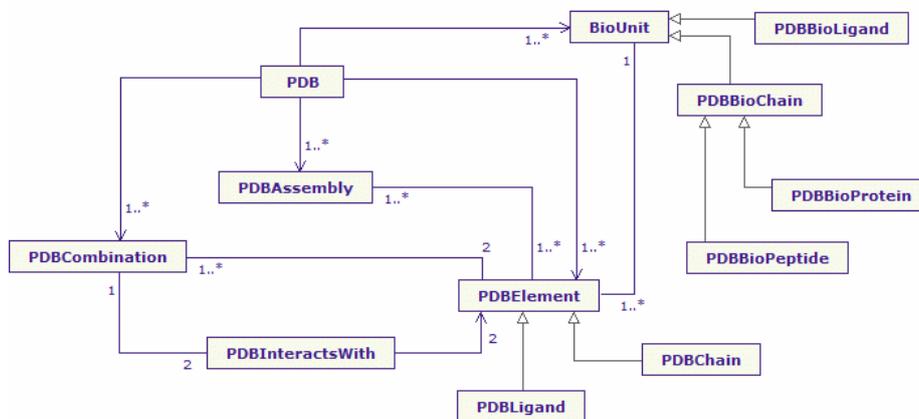


Figure 33: Internal representation of PDB data within the import process. The main purpose of these classes is to store identifiers and references that are required during data import. The associations among the classes are equivalent to the relations in the database. As an example, a PDB structure, which is represented by *PDB*, contains one to many *PDBCombinations*, which refer to interfaces between proteins and proteins or proteins and ligands. One of these combinations in turn are made of exactly two *PDBElements* which refer to single molecules, either a protein or a ligand.

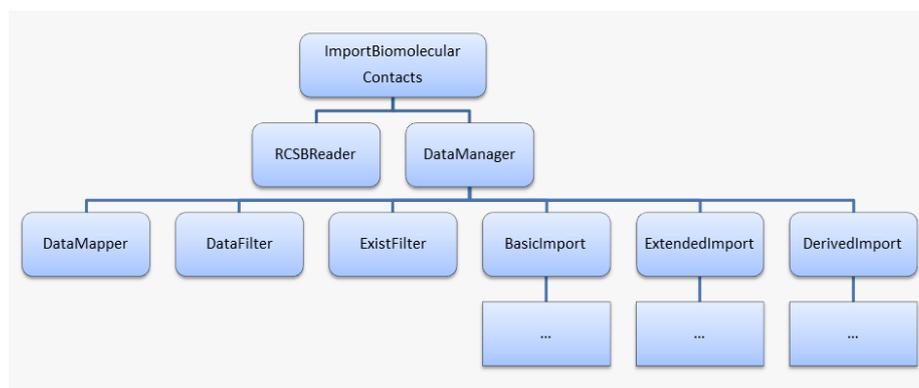


Figure 34: Class diagram for the representation of PDB data within the import framework. The node objects stand for APWorkflow classes. They are linked with subsequent objects that are started from left to right in this illustration.

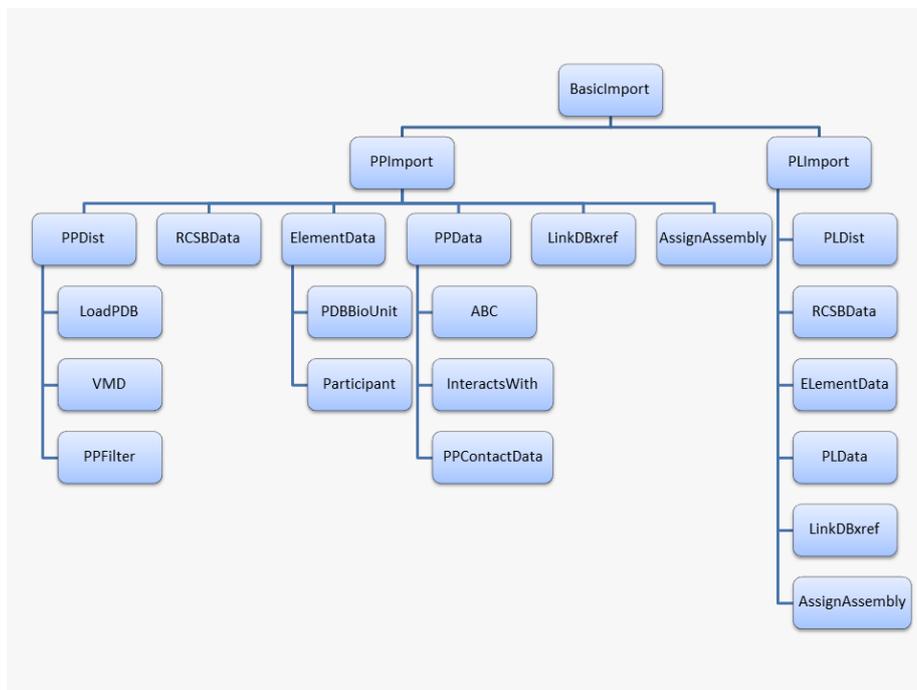


Figure 35: Object hierarchy for basic import process. The structure of the diagram is equivalent to figure 34

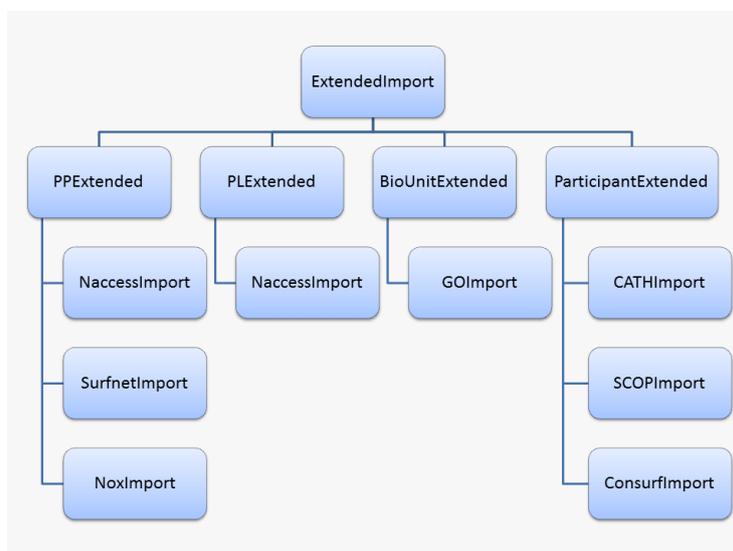


Figure 36: Object hierarchy for extended import process.

deleted is defined by its *abcEntity* identifier which represents as an individual element or as array of different elements the input for the main workflow. The subworkflow *deleteAbcEntity* removes all common data that are related to an interface such as classification predicted by NOXclass for an interface or geometrical properties such as the gap volume. Any contact information for the interface is deleted by *deleteInteractsWith* and affects relations like *surfaceAtomContact* and *atomDistance*. *DeleteParticipant* removes the individual binding partners that are involved in the interaction. In contrast to the previous tasks, there may exist dependencies if a binding partner is also found in another interface. In such a case, the participant itself has to be preserved, only the residues or small atom molecules that exclusively interact in the interaction are removed from the corresponding relations. Equivalently, *deleteBioUnit* is responsible for deleting the binding partners with reference to their structure-independent properties and also considers putative dependencies. Finally, *deletePDB* deletes irrelevant data from the relations that refer to identifiers from RCSB and decides whether the entire PDB has to be deleted provided that no interface exists for this entry any more.

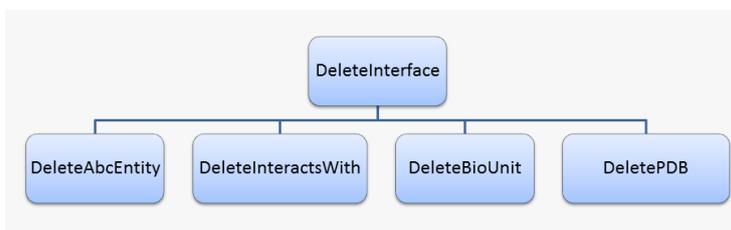


Figure 37: Object hierarchy for deleting data.

In general, modification of existing data in ABC<sup>2</sup> is very rarely required because most of the information in the database can be considered as static information which is not subjected to frequent modifications. Therefore, no special routines are designed for updating existing information. Whenever an update is required, the corresponding interfaces which are affected are deleted first and then re-imported into the database, using the functionalities from the framework as described above.

### 3.5 Website

For accessing interaction data from the Internet, a website was created. Figure 40 shows the search menu. A user may perform queries for protein-protein interfaces and protein-ligand interfaces according to criteria such as the hydrophobicity of the interface, functional meaning and conservation. After completion of a query task, the result page is displayed showing a list of interfaces fitting to the search criteria. For every interface, further information can be accessed like the visualization of the interface, conservation scores and binding preferences for every interface residue and so on, see figure 41. The website was programmed under Java using servlets and Java server pages [113] which are libraries facilitating the creation and management of web pages. It is hosted by a Tomcat webserver [114], which is an open-source implementation for usage with servlet and JSP technology.

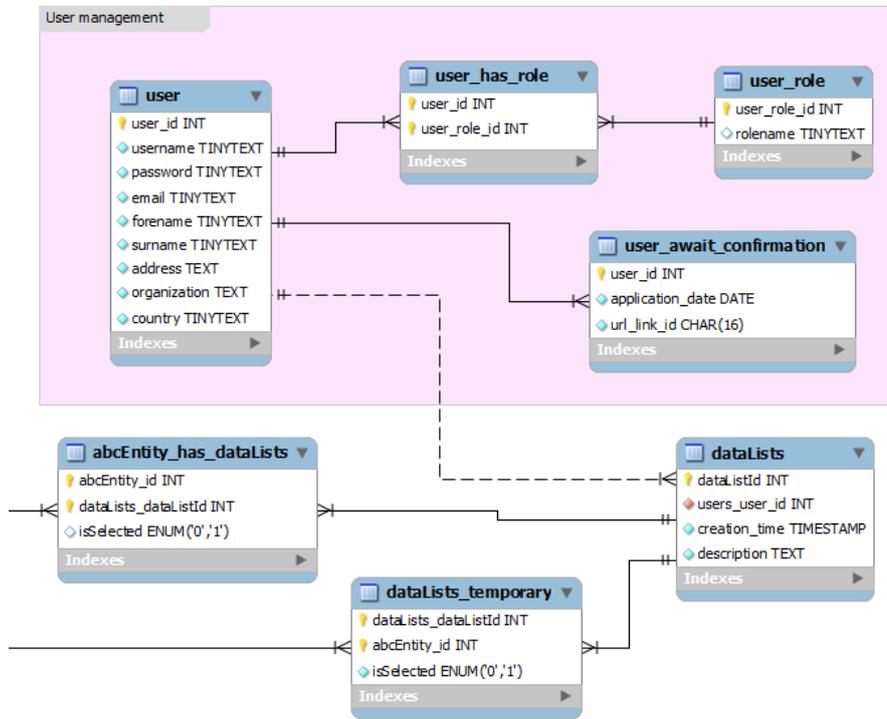


Figure 38: Database relations which are required for the website implementation.

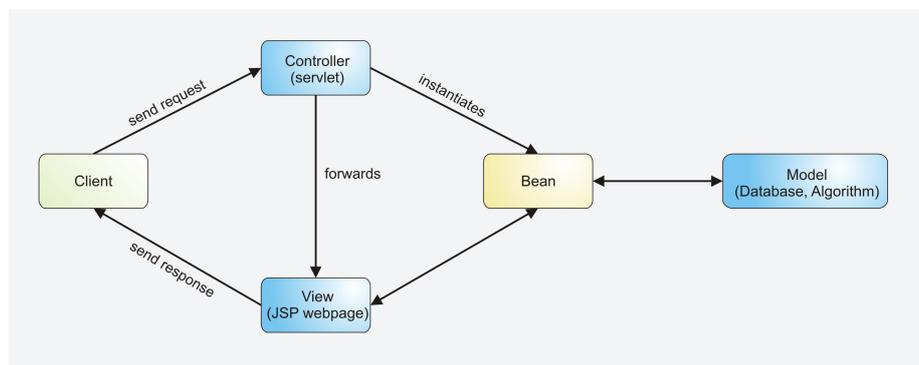


Figure 39: Model View controller (MVC) framework.

The structure of the website follows the model-view-controller (MVC) approach which is described in figure 39 [110, 115]. The basic idea of MVC is a strict separation of presentation and data. All aspects covering mere presentation and layout of data is the responsibility of the view object. Any logic that can be executed is compiled in the model object. The controller serves as mediator between these two objects. It decides which model to use and manages the data transfer from the model to the view object.

Let us assume that a user sends a database request to the webserver. Internally the controller which is represented by a servlet receives the request, checks whether the query appears valid and prepares an appropriate SQL command which is send via Bean to the database acting as the model. A bean is a java specific object that is responsible for storing and providing data by getter and setter methods. The bean receives the answer from the database and the view containing all layout elements that form the website is filled with the data. Eventually the server sends the final webpage as response to the user.

Figure 38 shows some tables that are related to user management of the website. In particular, the interfaces that were found upon a query can be permanently stored by a registered user and are located in the *abcEntity\_has\_dataLists* relation. The *abcEntity\_temporary* relation stores query results temporarily for an active web session for registered as well for unregistered users. This relation acts faster than *abcEntity\_has\_dataLists* because it is entirely kept in memory and not on physical storage.

## Search for protein-protein interactions:

**PDB** | Class | Interface features | GO ontology | Geometry

Interface definition:  
 Surface based  
 Distance based  
4

Hydrophobicity range  
Average hydrophobicity score range: -0.996 to 0.154

Surface size of interface (surface based interface definition only) (Å<sup>2</sup>)  
Average surface size score range: 500 to 1000

Size of interface  
Number of residues per interface: 25 to 70

Conservation of interface  
Average conservation score range: -0.2 to 0.2

Conservation of interface  
Average conservation score range: -0.2 to 0.2

## Search for protein-ligand interactions:

**PDB** | Interface features | GO ontology | Geometry | Small molecules

Search geometric features

Volume Å<sup>3</sup>: 658.364 to 2011.98

Eccentricity (0-1): 0.4 to 0.6

Planarity: 5.183 to 69.904

Figure 40: Query menu of the ABC database.

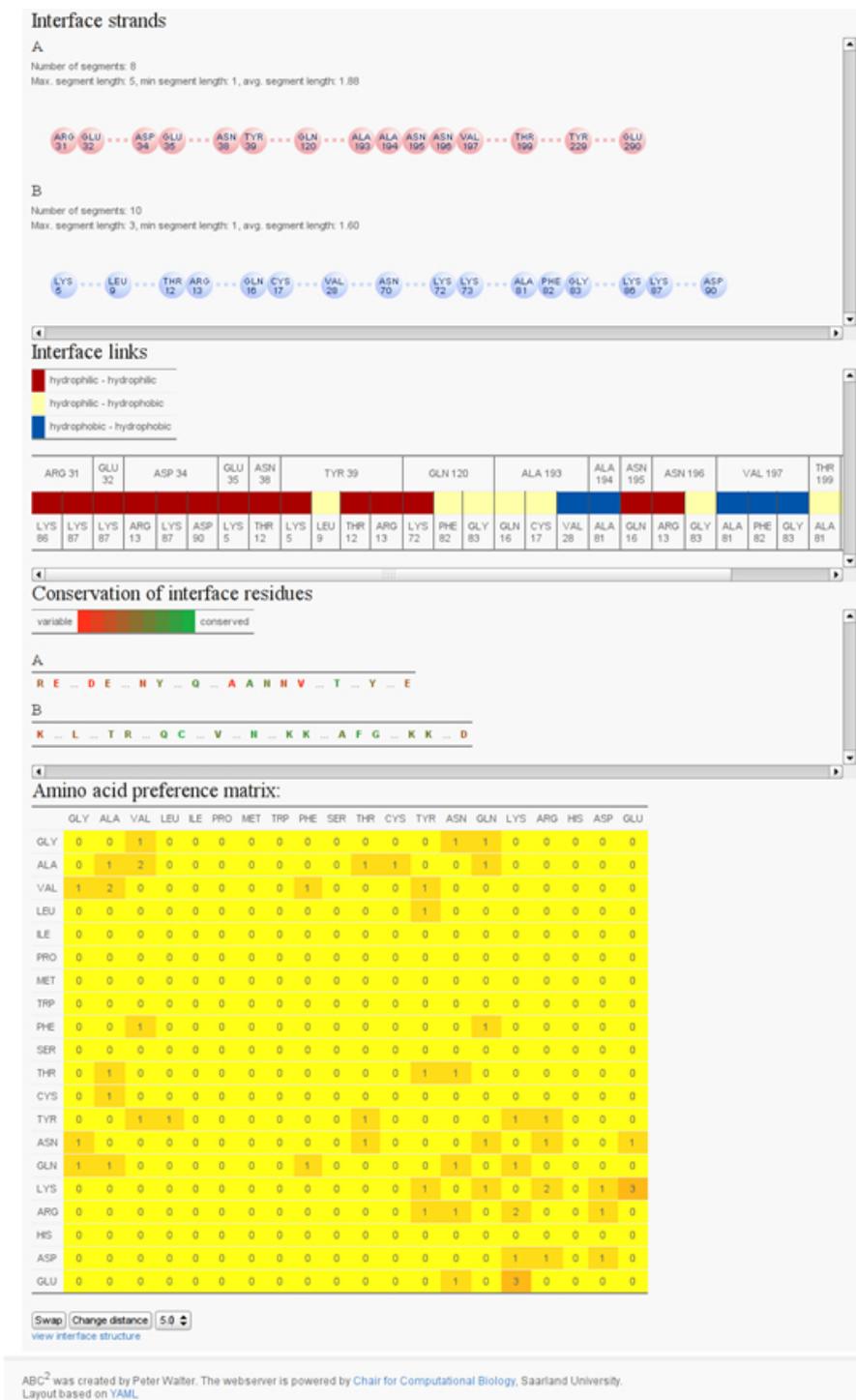


Figure 41: Result page for a query request showing information about an interface.

## 4 Function-structure relationships in obligate or non-obligate protein-protein interactions

Elucidating the principles determining the function of protein-protein interactions for example allows predicting the functions of little studied protein monomers or complexes [59, 116, 117] and contributes to the evaluation of putative interactions suggested by docking methods. Here, we investigated the two classes of obligate and non-obligate interactions and clustered all complexes according to the similarity of their annotations in the Gene Ontology. The main aim of this project was to introduce the pairwise comparison of protein-protein complexes based on functional GO-annotations in an automated fashion. We found that partners involved in obligate interactions share a high similarity of molecular function, whereas those of non-obligate interactions do not. Along the same lines, partners of obligate interactions have a high degree of co-localization whereas partners of non-obligate interactions may belong to the same or different compartments. When two different complexes show more than 80% similarity of molecular function, they are either both obligate or both non-obligate interactions. We also classify functional clusters of interfaces according to four physico-chemical properties. Incorporating functional annotations from GO-ontology is suggested as another useful means for the characterization of obligate and non-obligate complexes. The findings of this study allow making predictions of the interaction type of putative protein complexes in the absence of knowledge on the structure of the complex.

An interaction is considered as obligate if the binding partners of the complex do not exist separated from each other, whereas in a non-obligate interaction the chains can be found in the bound as well as in the unbound conformation [13]. In contrast, the groups of transient and permanent interactions are based on the lifetimes of a protein-protein complex. Another very simple classification is the group of heteromeric and homomeric complexes that are closely related to obligate and non-obligate interactions. It has been found that many obligate interactions take place between identical chains whereas non-obligate interactions emerge from different chains [13].

To fully understand the underlying principles and the meaning of protein interactions, it is desirable to find physico-chemical features that are characteristic for the classifications mentioned above. Examples of such features are the surface area of the interface, the amino acid composition at the interface, the residue propensities, the gap volume of the interface area, the hydrophobicity and the conservation of interface residues [14, 55]. These features can then be used to characterise various interface types. It was found that, in general, obligate interfaces are more hydrophobic than non-obligate interfaces and tend to involve a larger number of interface residues [20, 13, 118]. Besides, residues from obligate interfaces were found to be more conserved than non-obligate residues [119]. Also, significant differences were identified in amino acid composition and residue propensity for several types of interactions [17]. The concept of an obligate / non-obligate or a transient / permanent interaction gives a rough indication about possible functions of a protein complex [120]. For example, an enzyme inhibitor complex exhibits a high binding affinity resulting in a non obligatory permanent interaction. A structural protein such as actin turns out to be obligate and permanent whereas an antigen-antibody complex

is non-obligate and permanent due to the fact that the binding partners exist separated from each other but bind strongly upon complexation. In this work we employ an updated, larger data set to investigate the relationship between protein function and complex stability in more detail. As mentioned before, Noreen and Thornton [120] already pointed out the tendency for co-localization of obligate interfaces but they did not apply gene ontology to underline this observation. Moreover, Mintseris and Weng [119] already formulated the relationship between interface type and GO-terms. However, as shown below, we have achieved a more subtle differentiation of GO-terms by applying a similarity measure in an automated fashion. This allowed us to introduce the putative suitability of functional characterization for a prediction of obligate and non-obligate interfaces in the absence of structural data.

A further key focal point to enhance our understanding of protein-protein interactions would be to find a correspondence between structure features and interaction mode. For example, Aloy *et al.* found that two complexes with a close sequence homology often interact in the same way, whereas complexes that only have similar structural folds often use different contact interfaces [15]. Other studies pointed out that proteins with globally similar structures may have different functions [68, 121]. Furthermore, Keskin *et al.* described that due to evolutionary processes (divergent or convergent) even structurally similar interfaces may be functionally dissimilar [16]. When focussing on physico-chemical parameters of interfaces, characteristic properties were found for different functional classes. For instance, serine proteases prefer main-chain-main-chain interactions whereas the interfaces of antigen-antibody complexes are enriched in sidechain-sidechain interactions [122]. Besides, cell surface receptors differ from other homo- and heterocomplexes with respect to charge complementarity and shape complementarity [123]. These examples show that our understanding of the structural determinants for protein-protein interactions is still limited.

In this work we therefore investigated the relationship between GO-similarity and the classification into obligate and non-obligate interactions. Our results show that particular biological functions are enriched either in obligate or in non-obligate interactions.

## 4.1 Data generation

The data on protein-protein interfaces was derived from our ABC<sup>2</sup>-database [124] that contains structural and physico-chemical information for protein-protein interfaces extracted from PDB files at the Protein Data Bank [125]. Among other things it contains 536 interfaces of which the obligate/non-obligate classification has been described in the literature [42, 103, 20]. Depending on the features to be analysed, we applied two different interface definitions to collect the interface residues of a complex. In the distance based approach, a residue belongs to an interface when it is near another residue of a different chain. The standard distance cutoff used by many authors is 5Å [21, 59, 126]. The surface based approach refers to the surface difference of a residue between the bound and the unbound states. A major problem when dealing with protein interfaces is the separation of biologically relevant interfaces from crystal contacts that only occur in the crystal environment [8]. Therefore, we only considered interfaces within a biological unit as classified in the RCSB database. It comprises all protein chains from a PDB file that are shown or believed to be biologically

functional. The structural considerations of this study are limited to the interface part of the protein structure. To compile a list of non-redundant interfaces we first calculated the Pearson-correlation of the interfacial amino acid composition between every pair of interfaces A and B in our data set according to the following formula:

$$\text{corr}(A, B) = \frac{\frac{1}{20} \sum_{i=1}^{20} (a_i - \bar{a})(b_i - \bar{b})}{\sqrt{\frac{1}{20} \sum_{i=1}^{20} (a_i - \bar{a})^2} \cdot \sqrt{\frac{1}{20} \sum_{i=1}^{20} (b_i - \bar{b})^2}} \quad (7)$$

$a_i$  and  $b_i$  stand for the  $i = 1 \dots 20$  amino acids from interface A and interface B. Every interface was then represented as a node in a network [127]. Two nodes with a correlation value greater than 90% were linked with an edge. Subsequently we sorted the nodes according to their degree in descending order and removed nodes starting from the node with largest degree until no edges were left. This resulted in a data set consisting of 441 non-redundant interfaces. For some analyses a non-redundant set of complexes was required. Using the same network as described above, all those nodes were linked with an edge where the sequence identity of the chains that include the interface was greater than 30%. 298 interfaces remained after the reduction. Even though our data sets contain several multiprotein complexes consisting of more than one interface, interactions were always binary i.e. we did not distinguish between interfaces from distinct complexes or from the same one.

## 4.2 Assignment of GO-terms

To assign a well-defined function to each protein we used the Gene ontology (GO) that represents a controlled vocabulary to describe genes and their corresponding products [128]. It provides three annotations. The molecular function refers to the activity exerted by a protein, such as acting as an enzyme, or to its abilities such as being a structural protein. A biological process comprises a set of molecular functions in which several proteins are involved. The cellular component describes the localization of the protein inside the cell. An important aspect of GO is the DAG structure of the ontologies. An excerpt of the molecular function tree is shown in figure 42.

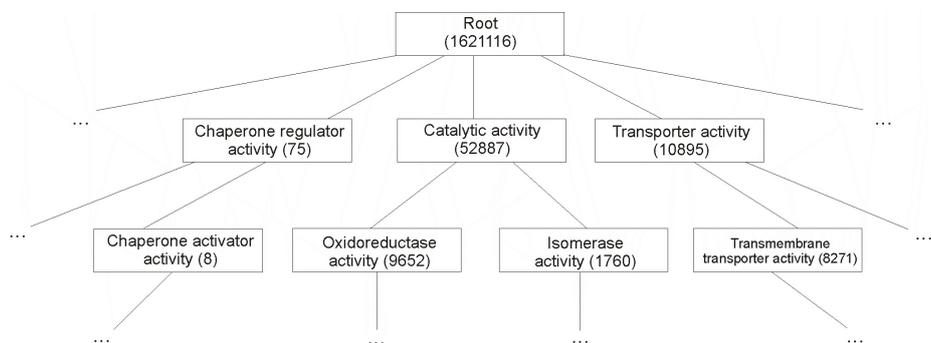


Figure 42: GO-tree for molecular functions. The numbers in brackets stand for the annotations for the corresponding GO-term.

Starting from the root it refers to the terms that constitute the most general denotation of a certain function, process or cellular localization. The next child nodes refer to more specialized terms and so on and so forth. Every node is assigned the number of annotations that fit to this GO-term, allowing to infer the level of generalization. A node that is annotated very often may represent a very common GO-term, whereas a node that is rarely annotated may stand for a specialised expression. The allocation of GO-terms to interfaces in our data set was based on two sources. The Gene Ontology Annotation (GOA) database contains a data set for the assignment of PDB files with their GO-terms [129]. However, the data is incomplete. Whenever information was missing the quickgo table was also taken into account which maps Uniprot-identifiers to GO-terms and is also a part of the GOA project. The relationship of GO-terms allows the application of mathematical methods to determine a scale for the similarity of GO-terms among each other. A number of such similarity scales for GO-terms are available [130, 131, 132] and were applied, for instance, for the prediction of functions of proteins [66].

In this work the similarity of protein function according to the gene ontology classification was determined by using a measure introduced by Schlicker *et al.* [133].

$$sim_{ij} = \frac{2\log(p_{LCA})}{\log(p_i) + \log(p_j)}(1 - p_{LCA}) \quad (8)$$

Here,  $sim_{ij}$  is the similarity between GO-terms  $i$  and  $j$ .  $p_i$  and  $p_j$  are the probabilities of the gene ontology terms  $i$  and  $j$  reflecting their relative frequency among all annotations.  $p_{LCA}$  stands for the probability of the last common ancestor (LCA). The LCA is the nearest parent node that both GO-nodes have in common. In this formula two crucial factors contribute to the overall similarity between the GO-terms. The last common ancestor (LCA) indicates the distance of the GO-terms in the hierarchy. The farther away the LCA lies, the less similar the GO-terms are. Besides, the number of annotations reflects the level of generalization. According to information theory a term occurring with a high frequency is believed to be less significant in comparison to terms which are more specific. Thus GO-terms that are rarely annotated obtain a higher similarity than GO-terms which are very common.

This formula was applied to calculate the semantic similarity of two separate proteins A and B. However, a protein chain may have more than a single GO-term. For instance, a protein kinase may have annotated molecular functions of 'Protein kinase activity' and 'ATP-binding'. In order to derive the similarity in such a case, we calculated the similarity for every GO-term from chain A with all GO-terms from chain B. For every GO term in A we selected the maximum similarity with a GO term from B and calculated the average of these maximum values. The same procedure was repeated for chain B. The average of the both maxima is denoted as  $sim_{rel}$ .

Further, we compared the similarity between binary complexes. From all possible chain combinations of the complexes AB and CD shown in figure 43 the GO-similarity for two protein complexes was calculated as:

$$GOFace_{avg} = \max\{0.5 \cdot (sim_{AC} + sim_{BD}), 0.5 \cdot (sim_{AD} + sim_{BC})\} \quad (9)$$

In other words, we considered every permutation of chain combinations between the complexes and took the one that gives the maximum value. For every com-

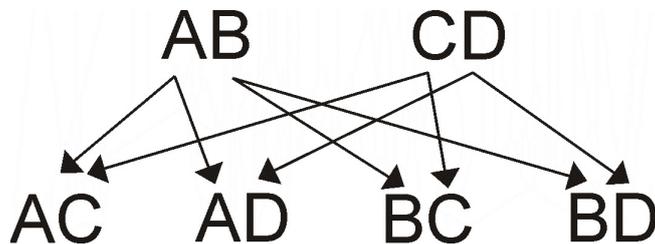


Figure 43: Combination between the protein chains to find the optimal pairing of similar chains.

plex, we extracted the terms for the molecular function, the biological process and the cellular component. Based on the non-redundant dataset we obtained 200 interfaces for molecular function (75 obligate, 125 non-obligate), 151 interfaces for biological process (78 obligate, 73 non-obligate) and 58 interfaces for cellular component (29 obligate, 29 non-obligate). The different numbers reflect the incomplete nature of GO annotations of the respective property.

### 4.3 Clustering

Then we applied hierarchical clustering implemented in the R package [134] to find groups of protein-protein complexes that are similar according to molecular function. We compiled lists of clusters with decreasing similarity by defining a cutoff every 10 steps of similarity percentage and collected all the subtrees below such a cutoff. These subtrees represent clusters whose members have a similarity among each other larger or equal than the given cutoff value.

Silhouette values were calculated in order to estimate the degree of dissimilarity between the clusters. Let us consider one cluster  $A$  among a set of clusters  $C$  and  $B = C \setminus A$ .  $a(i)$  is the average dissimilarity of an object  $i$  to all other objects of  $A$  and  $b(i) = \min\{d(i, B)\}$  stands for the lowest average dissimilarity of  $i$  to all other clusters in  $B$ . Then  $s(i)$  is defined as follows:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (10)$$

$s(i)$  takes on values between -1 and 1. A more positive value indicates that the object  $i$  fits well into its cluster whereas a more negative value suggests that the object  $i$  better fits into a different cluster. In other words, a high  $s(i)$  value represents well separated clusters.

Alternatively, the C-index provides an indication about the overall quality of the clustering. Let  $P$  be the set of pairs located in the same cluster and  $p$  the size of this set.  $S$  is the sum of distances of every entry in  $P$ .  $A$  is the set of pairs between all elements of the clusters. With  $S_{min}$  as the sum of the  $p$ -smallest distances of  $A$  and  $S_{max}$  as the sum of the  $p$ -largest distances of  $A$ ,  $C$  is computed as:

$$C = \frac{S - S_{min}}{S_{max} - S_{min}} \quad (11)$$

The C-index becomes small when  $S$  is small, meaning that the distances of pairs within the same cluster are small. Hence, a low value reflects a good clustering.

For the characterisation of the physico-chemical properties of a protein complex four features were employed, which are all related to the interface region. The interface surface area was calculated with the program NACCESS [104]. The gap volume is the volume between two molecules in the contact area and was calculated using the program SURFNET [73]. These values were then used to derive the gap volume index as the ratio between the two [14]:

$$GapIndex(\text{\AA}) = \frac{gapVolume(\text{\AA}^3)}{interfaceSurface(\text{\AA}^2)} \quad (12)$$

The nature of amino acids in the interface was characterized by their hydrophobicity using the Kyte-Doolittle scale [105].  $i$  runs over the interface residues and  $Surface_i$  refers to the interface area of  $i$ :

$$Kydo(interface) = \sum Surface_i \cdot KD_i \quad (13)$$

The value was calculated separately for amino acids that are located in the core region (defined as those residues that are fully buried upon complexation) or in the border region (defined as those residues that are only partially buried upon complexation) of the interface. Finally, the numbers of contacts between sidechain/sidechain, sidechain/backbone and backbone/backbone atoms were determined using our protein-protein interaction database [124].

#### 4.4 GO-term analysis for obligate and non-obligate interfaces

At first we investigated the correspondence between GO semantics and interface types on the subset of non-redundant 75 obligate and 125 non-obligate heteromeric complexes. This work is related to the work of Mintseris *et al.* who analyzed the relation between obligate/non-obligate interface type and function derived from GO-ontology [119]. In their work the comparison was done by counting the number of identical GO-terms shared by protein chains. In our analysis we apply a similarity measure allowing a more subtle differentiation of structures which are functionally similar. Another difference is that only terms from the fourth or deeper level were taken into consideration by Mintseris *et al.* Due to the structure of the GO-ontology, this is only a rough estimation of the specificity of particular GO terms as there are no clear levels in GO. In contrast, the measure of Schlicker *et al.* [133] we used in our work naturally incorporates the level of generality for a term. Therefore we consider our approach as less subjective and more sensitive. Figure 44 shows that most non-obligate interactions exhibit a very low or even no similarity and obligate interactions quite a high one. A typical example of a functionally dissimilar non-obligate interaction is 1TGP, consisting of chains Z and I. The former acts as serine endopeptidase enzyme whereas the latter is an enzyme inhibitor. From the functional point of view, these are very distinct roles resulting in very low overall similarity. Obligate interactions, however, do not show such a large diversity because of the tight contacts formed.

The same analysis was performed for the biological process annotations using 78 obligate and 73 non-obligate interfaces. In this case, a slightly different outcome was obtained, see figure 45. Again obligate interactions have high

### Molecular function similarity of obligate/non-obligate complexes

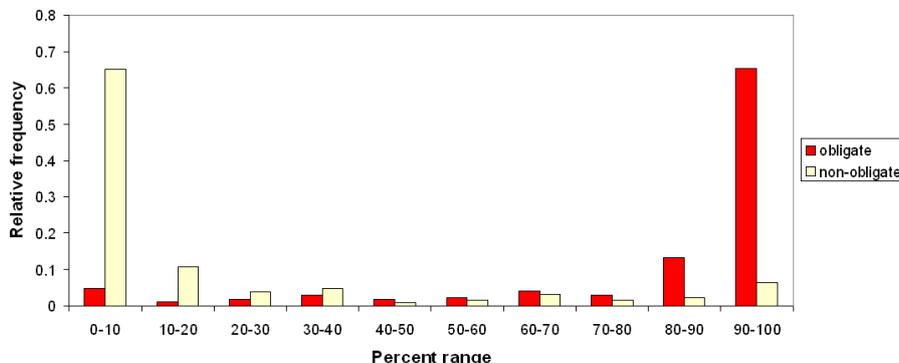


Figure 44: Distribution of the GO similarity according to molecular function between obligate and non-obligate complexes.

similarities whereas non-obligate interactions may have low to high similarity. Basically a biological process can be considered as a collection of molecular functions working together, which implies a certain relationship between these two ontologies at a more general level.

We also calculated the similarity of the cellular components of 29 obligate and 29 non-obligate complexes, see figure 46. About half of the non-obligate interfaces have a zero-similarity, whereas obligate interfaces tend to have a similarity range between 60 and 100 percent. These results are to be expected as protein chains involved in obligate interactions have to be localized in the same compartment, as the monomers alone are not stable. Nonetheless, a few complexes (1EXB, 1F3U, 1GPW) exhibit poorly corresponding annotations of their cellular components. The reason is that the respective GO-terms are quite general and contain only a little amount of information. This results in a low similarity value. Non-obligate interactions, by contrast, are not subjected to co-localization, so that the occurrence of the protein chains may be more or less distinct. This observation seems quite noteworthy when considering that co-localization has been used in the past as a criterion to judge the validity of experimental protein-protein interaction data [135]. An important aspect is that obligate structures tend to be more homomeric than non-obligate ones. As monomers of obligate interactions cannot exist alone, they should be co-expressed so that they can bind immediately with each other. Expression of identical chains is the easiest way of co-expression.

The observations made so far indicate that there is a preference for obligate interactions to have similar GO-ontologies whereas non-obligate interactions show a wider variety.

## 4.5 Functional clustering of complexes

The previous section characterized the functional similarity of two proteins forming one complex. In the following, however, we will compare the similarity and

### Biological process similarity of obligate/non-obligate complexes

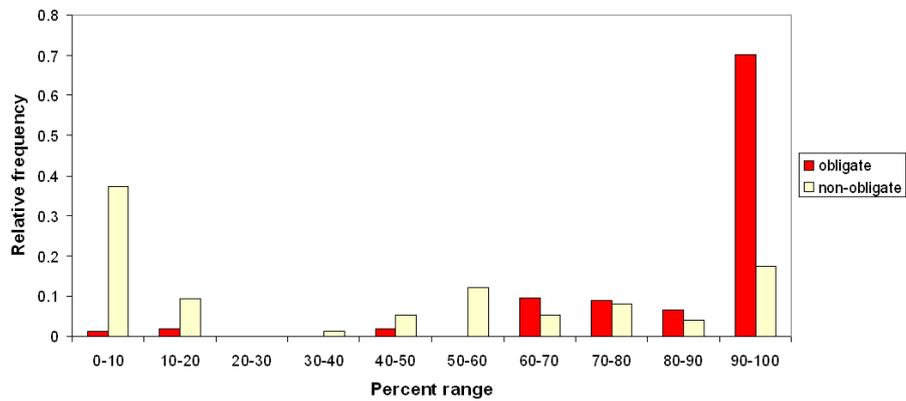


Figure 45: Distribution of the GO similarities according to biological process between obligate and non-obligate complexes.

### Cellular component similarity of obligate/non-obligate interactions

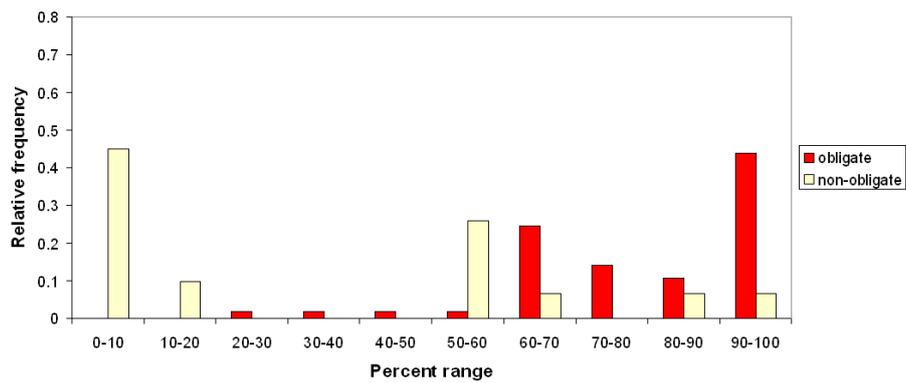


Figure 46: Distribution of the GO similarities according to cellular component between obligate and non-obligate complexes.

physico-chemical properties of pairs of complexes. To this end, the following analyses were performed after hierarchical clustering of complexes according to eq. (9). To validate the clustering, C-index was calculated according to eq. (11). Figure 47 shows that with decreasing similarity cutoff the C-index value goes up which means that the cluster differences converge due to accumulation of the groups. In comparison, C-index values of the random clustering reside near one indicating poor clustering. Both curves converge for low similarity thresholds because very few clusters involving many members exist there.

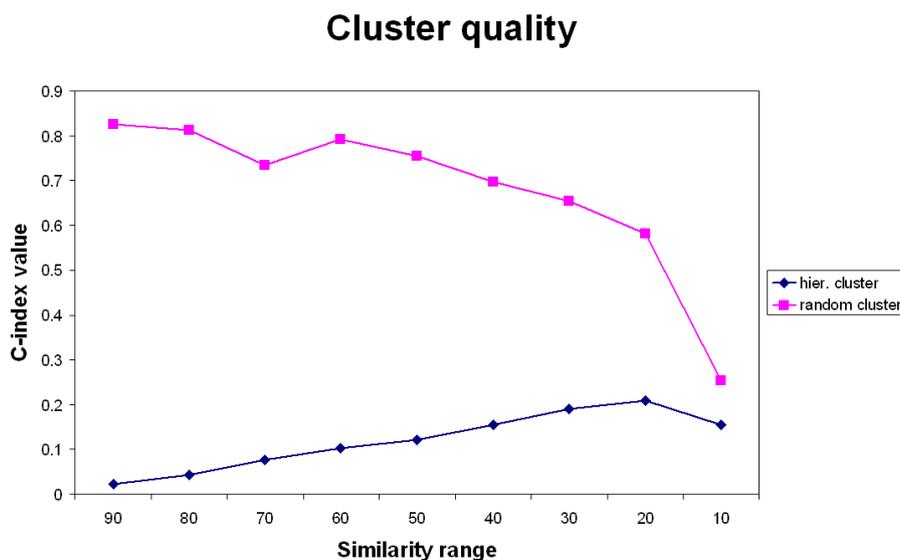


Figure 47: C-index values against clusters with decreasing similarity cutoff. The blue line represents the index values for the clustered data whereas the pink line stands for the same cluster whose members were re-distributed randomly.

Next, the numbers of obligate and non-obligate interface types were counted for every cluster. Figure 48 shows that clusters with a high functional similarity almost invariably contain complexes with only one interface type.

With decreasing similarity the number of mixed clusters containing obligate and non-obligate complexes increases rapidly. This observation suggests a possible relationship between interface type and function. Based on these findings we focused in the following on the clustering with 80% functional similarity as the fraction of cluster groups containing exclusively obligate or non-obligate complexes is still very high in this case. The members within a group represent complexes having similar GO-terms with each other. For every cluster, we counted the most frequent GO-terms, providing an overall semantic description of the corresponding cluster. Table 5 shows the clusters having at least three members obtained after hierarchical clustering with a cutoff range of 80% functional similarity. For comparison, results at 60% functional similarity are included in the supplementary material, see figure 77.

After the original clustering the separation was partially improper, which means that some functionally similar groups appeared in distinct clusters. For

### Relative frequency of obligate/non-obligate clusters

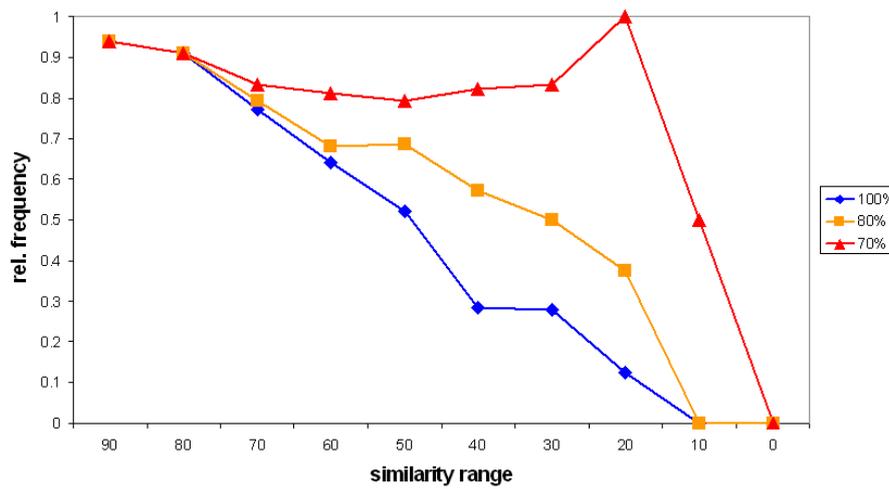


Figure 48: The schema depicts the distribution of interface types within the clusters. The 100 % curve  $\triangle \diamond$  stands for the fraction of cluster groups containing exclusively either obligate or non-obligate clusters. For the 80 % curve the proportion of one interface type is at least 80 %. The meaning of the 70 % curve is analogous.

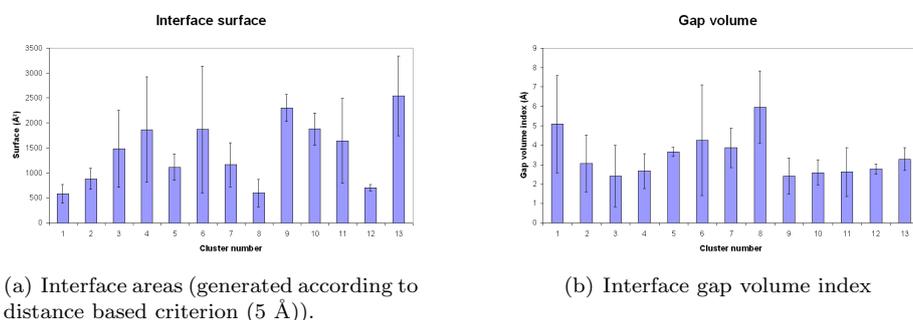
Id	Cluster members	Molecular function	obl.	non-obl.	a.s.v.
1	1AZZAC, 1AZZAD, 1CAOBD, 1PPFEI, 1ACBEI, 1HJABI, 1BRCEI, 4SGBEI, 1K90EI, 1HJACI, 1HIAAI, 1CHOEI, 3SGBEI	Serine-type endopepti- dase inhibitor activity	0	13	0.86
2	1UEAAB, 1SMPAI, 1KIGHI, 1FLEEI, 1EAIAC, 1SFIAI, 1DTDAB	Calcium ion binding, serine-type endopepti- dase inhibitor activity	0	7	0.21
3	1VKXAB, 1IHFAB, 1BFTAB, 1SMTAB, 1CMBAB	Transcription factor ac- tivity	5	0	0.84
4	1A4UAB, 1QSGEG, 1QSGFG, 1VFRAB, 2AE2AB	Oxidoreductase activity	5	0	0.62
5	3HHRAB, 3HHRAC, 1BP3AB, 1AXIAB	Hormone activity, Hematopoietin/interferon- class (D200-domain), cytokine receptor activ- ity	0	4	0.46
6	1F8RAC, 1F37AB, 2NACAB, 1GPEAB	NAD binding,FAD bind- ing	4	0	0.59
7	1LFDAB, 1FOEAB, 1HE1AC, 1WQ1GR	GTP binding	0	4	0.81
8	2PCFAB, 2PCBAB, 2PCCAB	Electron carrier activity	0	3	0.32
9	1HBNBE, 1HBNBC, 1MROAC	Coenzyme-B sul- foethylthiotransferase activity, Transferase activity	3	0	0.70
10	1FFVAB, 1QLBBC, 1KQFBC	Metal ion binding	3	0	0.42
11	1K8KCG, 1K8KDF, 1QFHAB	Actin binding	3	0	0.63
12	1B2SAD, 1AY7AB, 1BRSAD	Protein binding	0	3	0.63
13	1AJSAB, 1BJNAB, 1C7NAB	Catalytic activity, pyri- doxal phosphate binding	3	0	0.80

Table 5: Largest clusters for molecular function with similarity greater than 80%. Clusters with at least three members were taken into consideration. The data set was generated using hierarchical clustering. In the Cluster members column, the first four characters of every entry represent the PDB identifier, the last two characters stand for the protein chains forming the complex. The functional denotation of every cluster group is the most frequent GO-term all the cluster members have in common. The silhouette values in the last column represent the averaged silhouette value for all the members of a cluster (a.s.v.=average silhouette value).

this reason the cluster groups were subsequently merged when the similarity between the groups was at least 80%. Note that all clusters in table 5 are 'pure' meaning that their members are either all obligate or all non-obligate. Before merging the silhouette value was 0.61 and the average similarity between elements of the same cluster was 0.92. After merging, the silhouette value increased to 0.72 while the average similarity remained almost the same, namely 0.91. Clusters containing serine protease inhibitor complexes were merged most often (three times). Groups related to 'calcium ion binding', 'hormone activity', 'transcription factor binding', 'oxidoreductase activity' and 'GTP-binding' were merged once in each case. After the merging process, the silhouette values varied from 0.16 up to 0.99. The absence of negative values indicates that all elements fit quite well to the cluster which they belong to. The lowest silhouette value was found for 'nitrogenase activity'. This function is a subgroup of 'oxidoreductase activity', which is a very common group in the GO currently containing about 10.000 gene products. Hence, 'nitrogenase activity' tends to have quite a high similarity with other groups related to 'oxidoreductase activity' though the difference is still large enough for a separation into distinct clusters. Other groups with low silhouette values are calcium ion binding and electron carrier activity. The former is closely related to group 1. Both represent serine protease inhibitor complexes but group 2 is more specific because the complexes additionally contain a metal binding functionality resulting in a separation from the usual inhibitor complexes. The term 'electron carrier activity' (group 8) is a common expression and often comprises further functions such as 'iron ion binding' or 'peroxidase activity'. Removing another group (not listed in the table) which is also annotated the 'electron carrier activity' functionality results in an increase of the silhouette value from 0.32 to 0.66 for group 8. In this analysis we have neglected the domain character of the individual proteins. This means that we have used all annotated GO-terms for multi-domain proteins even if they only use one of these domains to interact with other proteins. This was mostly done for practical reasons as it turned out that assignment of GO-terms according to a certain domain definition results in a considerable loss of data.

For every cluster group, we investigated four physico-chemical properties. These features are shown in figure 49 (a), (b) and figures 50 - 51 for the clusters with 80% similarity cutoff. Some general tendencies are noticeable. As expected, the largest interface surface areas (groups 3,4,6,9,10,11,13) belong to obligate complexes, whereas smaller ones are found for non-obligate interactions. The interface areas of obligate complexes show a particular large variability for cluster 6. The average relative standard deviation for obligate and non-obligate interactions is 0.41 and 0.28 respectively. For example, in the first group containing serine protease inhibitors, the interface area values range from 283 to 941 Å<sup>2</sup>. Interestingly, the interfaces originating from the multiprotein complex 1AZZ vary noticeably from 503 Å<sup>2</sup> (chains A,D) to 942 Å<sup>2</sup> (chains A,C) even though the serine protease (chain A) is bound by identical inhibitor chains C and D. The second group is closely related to the first group. The difference is that these serine proteases are related to metal binding. Altogether the interface size tends to be larger within this group compared to the first one.

Despite the small size of the interface area for all members of group 1, the gap volume is quite large and shows a larger spread. A possible explanation for this is that the serine proteases cover a broad functional spectrum including



(a) Interface areas (generated according to distance based criterion (5 Å)).

(b) Interface gap volume index

Figure 49: Features for clusters with 80% functional similarity, see table 5)

digestion, inflammation, immunity and blood clotting. This apparently requires a certain amount of flexibility of the physico-chemical patterns of the interfaces. The smallest gap volume stems from 1K90EI that is an inhibitor complex from the serpin family, which form extremely stable complexes [136]. The largest gap volume is represented by 1CA0BD, containing Alzheimer's amyloid beta-protein precursor (APPI) which is a strong inhibitor of trypsin and many other serine proteinases [137].

### Hydrophobicity

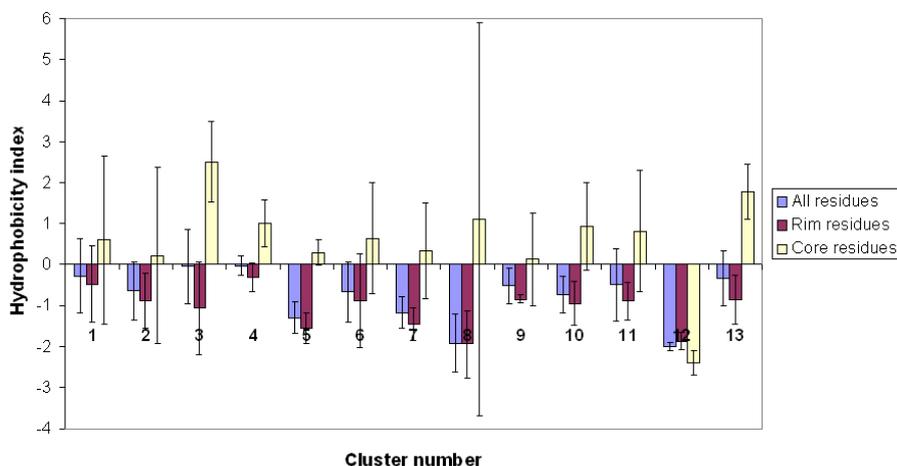


Figure 50: Hydrophobicity of interface residues calculated using Kyte-Doolittle indices.

The results from figure 50 show that the hydrophobicity in the border region clearly differs from the hydrophobicity in the core region. The former is more hydrophilic as the residues are at least partially exposed to the solvent, whereas the latter is shielded against the exterior. In general, functional groups containing obligate interactions tend to be more hydrophobic, whereas the non-obligate complexes are more hydrophilic. Cluster 3 contains complexes that are

## Sidechain backbone binding

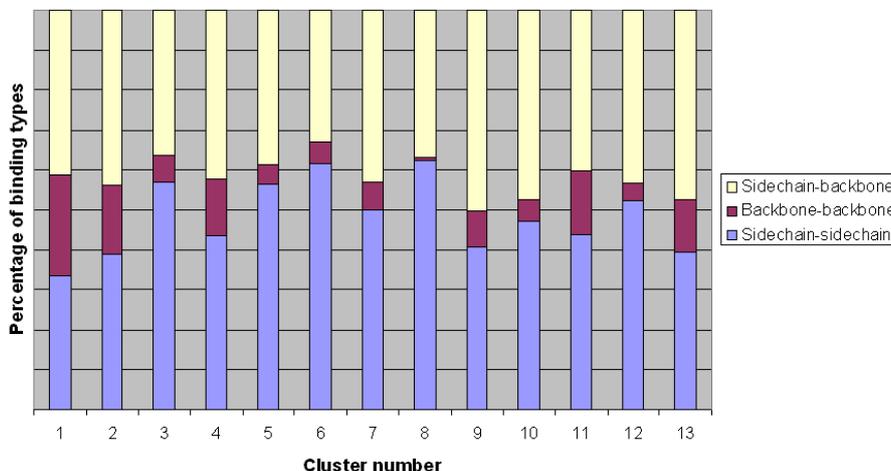


Figure 51: Relative distribution of sidechain / sidechain, sidechain / backbone and backbone / backbone contacts at the interface.

responsible for transcription factor activity. Their interfaces show the largest tendency to be hydrophobic in the core region. The contacts take place between homodimeric chains forming the actual complex which exerts the function as a whole. Obviously, as follows from their definition, obligate interfaces are not in contact with the solvent. Cluster 12, which contains endoribonucleases, comprises the most hydrophilic interfaces in the border region as well as in the core region. It is well-known that these complexes form numerous hydrogen bonds in the contact area [138]. This explains the high accessibility of the interface region to the solvent as well as the specificity for putative binding partners.

Group 1 exhibits the largest fraction of backbone/backbone interactions resulting in strong inhibitor complexes. In fact, the ratio varies with the volume and ranges from 19% for the broadest interface gap to 24% for the tightest one. This shows that with increasing gap size backbone atoms have fewer interactions with each other. Other groups with decreased backbone/backbone interaction show an increase in sidechain/sidechain contacts indicating a looser coupling between the binding partners. The ranges for the physico-chemical properties at 60% threshold of functional similarity are quite similar to the ones for 80% threshold (see figure 77 in supplementary material). Note, however, that at 60% clusters are often mixtures of obligate and non-obligate complexes, see figure 48.

The clear grouping of obligate and non-obligate interactions according to the function of their interfaces suggests that comparing the functional similarity between complexes might be helpful for predicting the type of interface. To test this assumption we compiled a set of interfaces from the RCSB for the group of serine protease inhibitors as a typical representative of non-obligate interactions and for the transcription factor complexes which stand for the obligate interactions. The complexes of these sets (listed in table 6) share functional

serine protease inhibitor complexes	1FY8EI, 1TM1EI, 1SGDEI, 1TMQAB, 1GOVAB
transcription factor complexes	1S3JAB, 1SGMAB, 1T33AB, 1UI5AB, 1R1TAB, 1P6ZNR, 1LJ9AB, 1KU2AB, 1KU9AB, 1HW5AB, 1FIAAB, 1FT9AB, 1GVJAB

Table 6: Set of complexes for NOXclass prediction. A set of non-redundant complexes was compiled for two functional classes. Serine protease inhibitor complexes (cluster 1 in table 5) are non-obligate whereas transcription factor complexes (cluster 3 in table 5) stand for obligate complexes.

id	0-10%	10-20%	20-30%	30-40%	40-50%
1	1.01	1.22	1.14	0.97	0.99
3	0.91	1.21	1.28	0.72	1.26
id	50-60%	60-70%	70-80%	80-90%	90-100%
1	0.84	0.47	-	-	-
3	0.73	0.71	0.47	0.95	1.37

Table 7: Set of interfaces for NOXclass prediction. The interface types 'obligate' and 'non-obligate' were predicted with NOXclass for about 7000 interfaces. For every cluster these interfaces were assigned into any of five bins according to the degree of their functional similarity with cluster 1 (serine protease inhibitors) or cluster 3 (transcription factor binding). The values given for each bin indicate the ratio of the quotient of the relative occurrence of obligate and non-obligate interfaces. For the 70%-100% range no interfaces were available for calculating the quotient for cluster 1.

similarity with groups 1 and 3 from the clustering mentioned above, but are non-homologous with its members.

For predicting the interface type in the absence of literature annotation, we applied the NOXclass program for the prediction of the interface types [103]. NOXclass uses physico-chemical parameters such as the interface size for the assignment of obligate and non-obligate interactions and applies machine learning techniques for the classification. The program outputs how probable the interface types are. As expected, all of the serine protease inhibitor complexes were predicted to be non-obligate (probabilities between 94.62% and 99.65%), whereas most of the transcription factor complexes turned out to be obligate (probabilities between 64.06% and 99.62%). The only exception is 1UI5AB. Besides we compiled a set of about 7000 interfaces for which we assigned the interface types as mentioned before and grouped them according to their functional similarity with cluster 1 or 3. The ratio of obligate and non-obligate interfaces in particular similarity range normalized by the total occurrence of obligate and non-obligate interfaces is listed in table 7.

A value lower than one indicates that there is a surplus of non-obligate interfaces whereas a value greater than one stands for a larger number of obligate interfaces. For low similarity ranges the value is near one which means an equal distribution of the interface types. For serine protease complexes there is a clear tendency towards more non-obligate interfaces whereas for transcription factor

complexes there is no clear distinction except for the largest similarity range. This might be due to a weakness in correctly predicting the interface type for transcription factor interfaces or means that transcription factors have no clear preference for a certain type. These observations show that there might indeed be a correlation between the functional role of certain interfaces and their type. The question is how similar the interfaces have to be to share the same interface type and whether this criterion is valid for all functional entities or only for some of them. Unfortunately, not enough validated data is currently available so that we could not test the statistical significance of this relationship. One reason is the uncertainty about obligate and non-obligate interactions. Another reason for this is that GO terms are still missing for many proteins. The manual curation takes time and requires a comprehensive understanding of the structure of GO to avoid redundancies and inconsistencies which becomes more and more difficult due to the steady growth of the GO [139]. However, GO is not the only source for the assignment of semantic meanings. Possibly a clearer and finer partition could be achieved by combining GO and other classification systems such as FunCat which may be the subject of future work [140]. However, it might be challenging to find a common denominator for the integration and establishing a similarity scale.

## 4.6 Conclusions

In this project, we examined the relationship between the interface types obligate / non-obligate and the semantic meaning of the corresponding protein complexes derived from the Gene Ontology database. First, we assigned GO-terms to protein chains and compared these terms between chains within the same complex. We found several characteristic relationships. For example, chains of obligate interfaces tend to be more functionally similar in comparison to chains of non-obligate interfaces. Also, partners involved in obligate interactions are very likely to be localized to the same compartments whereas those of non-obligate interactions are likely not co-localized. These findings underline the relationship between these interface types and the classification according to the GO-ontology.

When clustering pairs of complexes according to their functional similarity, at 80% threshold, a clear separation into 'pure' clusters was obtained containing either obligate complexes or non-obligate complexes. The physico-chemical properties revealed characteristic features for some groups, for instance serine protease inhibitor complexes all had very small interface surfaces. For other groups such as number 6 (NAD/FAD binding) interface surface area and gap volume showed a large variation. These observations suggest that interface features might be function-specific.

Overall, promising results were obtained from the prediction of interface types for two sets of functionally similar obligate and non-obligate interfaces suggesting that this method might allow simple predictions of obligate and non-obligate interfaces even if no structural information is available about particular complexes.

So far we focused on the data set of obligate and non-obligate interactions for which about 500 complexes are available. However, this data set only covers a small spectrum of functions, processes and localizations in the GO. Thus the next step will be to examine the interfaces of a larger data set of protein-protein

complexes generated from our protein-protein interaction database.

## 5 Predicting where small molecules bind at protein-protein interfaces

The project described in this section was done in collaboration with Jennifer Metzger from our group. The contribution of Jennifer was the design and training of the statistical classifier with random forests. My part was the the generation of interface features, the compilation of the input data set, and the patch analysis. Protein-protein interactions play important roles in most cellular processes [14, 120]. In the yeast *S. cerevisiae*, for example, interaction partners have been reported for more than 5000 of the 6000 yeast proteins [141]. In human cells protein interactions are involved in signaling processes such as in the MAPK cascade, and in regulatory processes for example the G-protein activated processes of hormone detection. Therefore, protein interactions are of vital interest for pharmaceutical intervention.

Currently, the main approach for designing inhibitors and modulators of protein-protein interactions comprised peptidomimetics. These are short peptides that mimic parts of the interface and compete with the natural protein binding partner for the same interface [4]. As some of these binding interfaces can also bind small molecule ligands, modulating the activities of protein-protein complexes by competitive or allosteric small molecule protein-protein inhibitors (SMPPIs) has become an area of very active interest in current pharmaceutical research [142].

Our structural understanding of the interaction of proteins with other proteins and with small molecule ligands originates largely from the atomistic three-dimensional protein structures deposited in the Protein Data Bank [125]. Statistical analysis of these complexes has allowed deriving some general principles about the binding interfaces of protein complexes. For example, permanent complexes tend to have large and hydrophobic interfaces whereas transient interactions often involve binding via smaller and more polar interfaces [36]. Besides, some binding interfaces resemble an O-ring where a hydrophobic interior is surrounded by a ring formed of polar and charged residues [22]. Protein binding interfaces are rather flat, on average, particularly when compared to those involved in binding small ligands that often bind into pronounced clefts on the protein surface. Yet, binding of ligands and/or the natural conformational dynamic fluctuations of proteins may induce the formation of binding pockets of suitable size and polarity as shown for several systems such as IL2-IL2-R, p53-MDM2, and Bcl3 XL [77, 142].

Interestingly, not all interface residues play the same role for the stability (binding affinity) of the complex. There often exists a subset of interface residues, the so-called hot spots, that are mainly responsible for the binding affinity [143, 22] and these may be promising locations for binding of a small molecule. These hot-spot residues are generally not spread over the entire interface but located in clusters. Thus, one could expect that successful SMPPIs preferentially bind in regions where hot spots residues are enriched. Several fast computational prediction-algorithms are available for the identification of hot-spot residues of protein-protein interfaces [28, 144]. Rational design of SMPPIs presents a considerable challenge. On the one hand, only few natural ligands or substrates are available for protein-protein interaction sites in comparison to the conventional protein targets, and this severely limits the template-based design

of appropriate small molecules [142]. Moreover, the interaction site of protein complexes is rather flat in contrast to typical protein-ligand interaction areas that contain more clefts and pockets. This poses a problem for achieving affinity and selectivity of a putative inhibitor [145, 146, 76]. Finally, protein-protein interface sites are often much larger than the average size of small molecule inhibitors which raises the question where the ligands should bind to ensure efficient inhibition. Despite all these difficulties, promising progress has been made towards finding drugs efficiently inhibiting protein-protein interactions such as for the p53 MDM2 system that plays an essential role in cancer therapy [145]. Only few studies have so far compared the general properties of protein-protein and protein-ligand complexes [145], [147, 148, 149]. The Timbal database contains structural data for a small number of protein-protein complexes and their complementary protein-ligand inhibitor complexes [150]. It was argued that the mere existence of pockets is not sufficient for determining the druggability of a protein-protein interface [150].

A more general approach was presented by Davis and Sali [151] who compiled a dataset of protein-protein and protein-ligand complexes. They classified surface residues at the binding interface into ‘bifunctional sites’ that contain residues which are involved in protein, as well as in ligand binding, and ‘monofunctional sites’ that only interact with other proteins. In order to maximize the amount of data, the proteins involved in either PP or PL interactions just had to be homologous to each other. This study did not distinguish cases where the ligand is integrated in the protein-protein interface or where it competes against the protein partner for the same binding site. Thus, the complexes containing bifunctional sites covered a wide functional range, including protein-protein inhibitor complexes, enzyme-substrate complexes and complexes involved in regulatory tasks or structural interaction.

In this work we focused on PP/PL pairs in which a ligand  $L_j$  and a second protein  $P_{i2}$  compete for the same binding interface on the surface of the first protein  $P_{i1}$ . Due to the scarcity of available complex structures, we adopted the same approach as Davis and Sali, and we considered all homologous pairs  $P_{i1}$ ,  $P_{i3}$  that share at least 40% sequence identity. According to Aloy and Russell this level of homology typically guarantees binding via the same binding interface [15]. We successfully identified several evolutionary and physicochemical features allowing a distinction between residues on a protein surface that participate in protein as well as in ligand binding and residues preferring protein binding only. The former are denoted as overlapping residues, the latter as non-overlapping residues. Non-overlapping residues can in principle originate either from the protein-protein complex or from the protein-ligand complex. Figure 52 shows an example for this classification. Based on these features, we then derived a statistical classifier that can identify promising ligand binding residues inside protein-protein interfaces with an accuracy of about 67%.

## 5.1 Results and Discussion

This study aims at characterizing the nature of protein residues at overlapping protein-protein and protein-ligand binding interfaces. More precisely, given the three-dimensional structure of such a protein-protein interface, we aimed at developing a method for predicting to which place of this interface small molecule ligands would bind most likely. In a drug design project targeting a known pro-

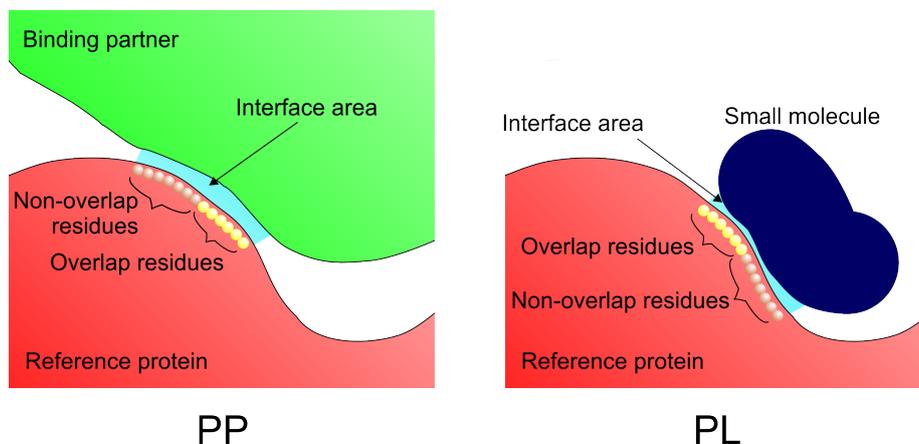


Figure 52: Schema of a pair of shared PP and PL interaction.

tein interface, this would allow focusing the virtual or experimental screening efforts on ligands with physico-chemical properties that are complementary to this portion of the binding interface.

As explained in the Materials and Methods section, a dataset of 175 tuples  $(P_{i1}, P_{i2}), (P_{i3}, L_j)$  was derived, where  $P_{i1}$ ,  $P_{i2}$  and  $P_{i3}$  are three proteins and  $L_j$  is a small molecule ligand,  $P_{i1}$  and  $P_{i3}$  share at least 40% sequence identity, and the aligned positions in the binding interfaces of  $P_{i1}$ - $P_{i2}$  and  $P_{i3}$ - $L_j$  have at least two residues in common. First, we compared the binding patterns of the protein-protein complex with its corresponding protein-small molecule partner. To this end, we mapped all binding atoms of the overlapping residues to their pharmacophore groups and counted which pharmacophore groups from the reference proteins bind to pharmacophore groups from the corresponding partner. We obtained a list of binding pharmacophores for PP and PL and calculated the similarity of the binding patterns using Tanimoto coefficient [108]. The average value over all coefficients was 0.27. The main reason for the low value is that for PP in general, the number of binding atom pairs is much larger than for the corresponding PL due to the larger number of putative binding partners in the interfacial area. To estimate the relevance, we shuffled the partner pharmacophore groups from the dataset and re-assigned them randomly to the pharmacophore groups of the reference proteins. Applying this method, we obtained an average Tanimoto coefficient of 0.20 indicating a slightly larger similarity of the binding patterns between PP and PL pairs. In figure 53, we compare the distribution of pharmacophore groups for the overlap and non-overlap residues on reference proteins from PP and PL.

Table 8 shows four representative examples of such tuples. A list of all pairs is available in the supplementary material, see figure 22. The complex pairs 1MBQ-1BZX represent the typical relation between the ligand in the PL complex and the binding protein partner in the PP complex. Here, the ligand is denoted as inhibitor or antagonist and no binding partner from the corresponding PP pair is available in such a structure. The pair 1GZR-2DSQ refers to interactions with insuline molecules. In contrast to the first example the PP structure contains further protein chains which are colored grey in the figure,

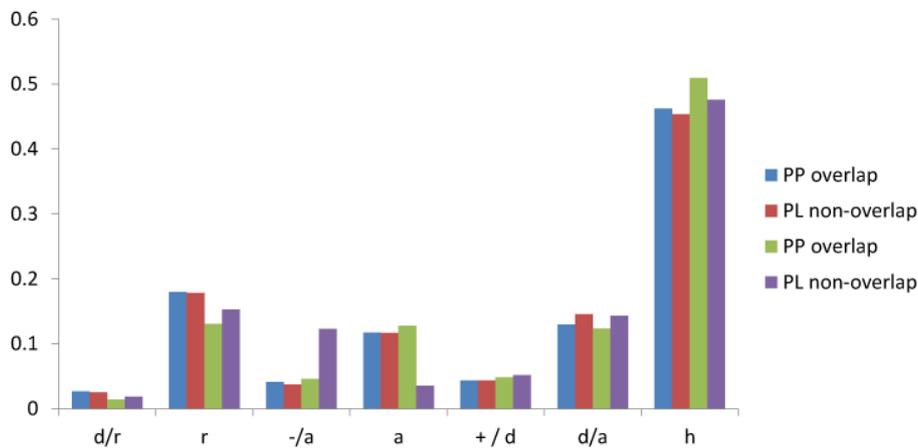


Figure 53: Distribution of pharmacophore groups from overlap and non-overlap residues from reference proteins.

forming a multimeric complex with four chains. In this case, the ligand is most likely a detergent molecule. The pair 1GOY-1X1U is the well-known barnase-barstar system. Here, the roles of ligand and partner protein are reversed. In the PL interaction, the ribonuclease barnase is bound to its natural ligand GMP whereas the protein binding partner barstar acts as inhibitor in the PP complex. In the automatic derivation of the dataset the function of the ligand was not considered. We believe that the mere existence of a small ligand binding site irrespective of the functional relationship provides valuable information for finding potential drug targets. The last example 1DBN-2DVG shows another combination in our dataset. In this case, the PL pair is formed by a sugar binding protein and its natural ligand N-Acetylglucosamine. Additionally, the complex contains another protein chain forming a homodimer. The ligand is integrated into this interface area. The corresponding PP complex is extracted from a quaternary homomeric structure. Both PDB files thus contain an equivalent protein-protein complex. Our workflow also picks up this kind of pairs because the ligand is tightly embedded in the interface area thus causing the collision as described in the method section. Our final dataset contains about 20 of such cases. We could have manually removed them but we argued that such data also provide valid information for the derivation of our prediction approach. Such a case reveals two facts making this kind of pair appropriate for our dataset. First, the ligand indeed exhibits overlapping residues with the reference protein regardless of the existence of further overlaps with different proteins. Second, the ligand only exists in the PL but not in the PP complex, telling us that the protein-ligand binding appears to be optional.

Figure 54 details the geometric relation between a PP and PL pair at the well-known example of the trypsin-benzamidine complex. The left picture shows the case observed in nature where the bovine pancreatic trypsin inhibitor (BPTI) binds to trypsin and blocks its active site. The right picture shows how a small benzamidine molecule binds into a cavity at the trypsin:BPTI interface and partially occupies the binding interface for trypsin inhibitor proteins.

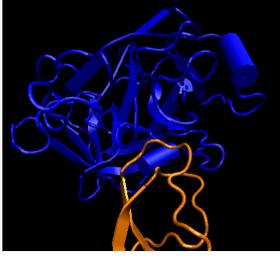
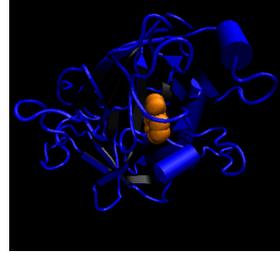
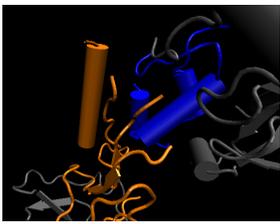
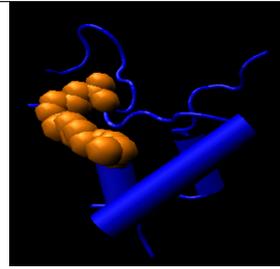
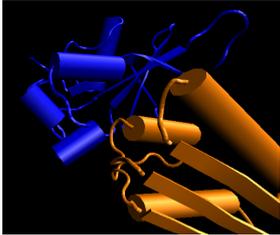
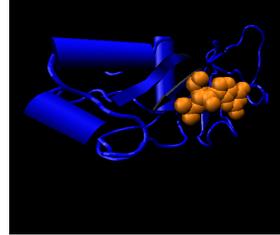
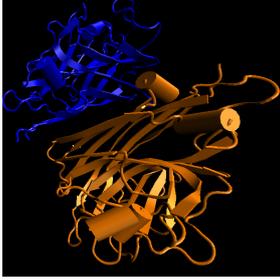
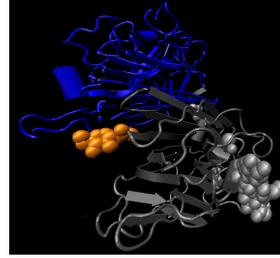
protein-protein complex	Protein ligand complex
 <p>1BZX <b>E</b>:I</p>	 <p>1MBQ <b>A</b>:BEN</p>
 <p>2DSQ <b>I</b>:G</p>	 <p>1GZR <b>B</b>:C15</p>
 <p>1X1U <b>A</b>:D</p>	 <p>1GOY <b>A</b>:3GP</p>
 <p>2DVG <b>C</b>:B</p>	 <p>1DBN <b>A</b>:NAG</p>

Table 8: Shown are pairs of protein-protein complexes (left) and the related protein-ligand complex (right). The identifier below the plot gives the name of the PDB entry and the chains or ligands used. The chains identifiers of the reference proteins are marked in bold.

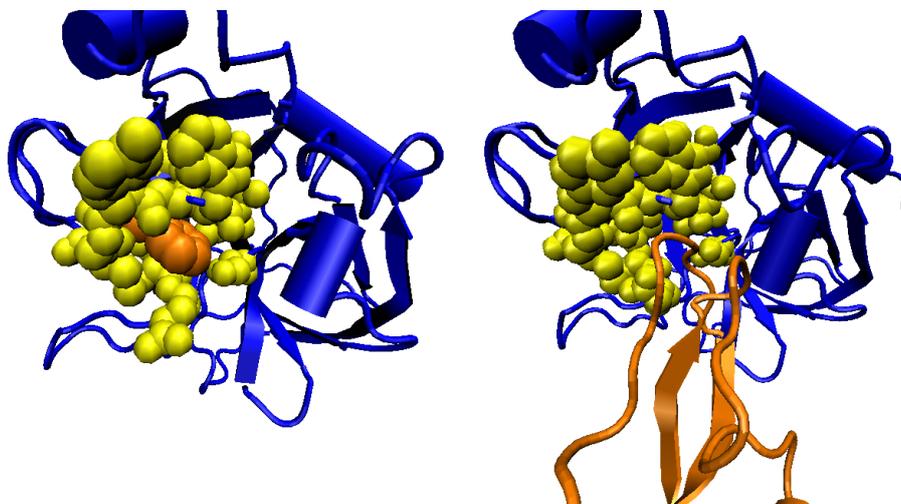


Figure 54: Example of a PP/PL pair 1MBQ A:BEN and 1BZX E:I. The reference proteins are marked in dark grey, the ligand and the binding protein partner in light grey. Additionally, the overlapping residues from the reference proteins are drawn as bright spheres.

The PP and PL interfaces in our dataset involve on average  $35.5 \pm 24.0$   $Pi1$  residues (PP) or  $8.7 \pm 6.1$   $Pi3$  residues (PL). We observed that the number of ligand atoms was not related to the size of the ligand binding interface (Pearson correlation coefficient equals -0.07). Next, figure 55 shows a plot of the number of overlap residues relative to the total number of interface residues for different interface sizes. On average, only 25% of the residues in a PP interface participate in the overlapping region whereas 79% of the residues of the corresponding PL interface belong to the overlap region. In fact, 64 PL interfaces out of the 175 PP/PL pairs (37%) were fully covered by the overlapping area compared to none of the PP interfaces.

For deriving the statistical classifier, we first needed to identify physico-chemical properties of residues at binding interfaces that display different distributions for residues in the overlapping part of PP/PL interfaces and for non-overlapping residues. For this, we tested several structural and evolutionary features of the interfaces that were discussed previously in the literature. An important feature for discriminating overlapping and non-overlapping regions turned out to be the evolutionary conservation of residues at the protein binding interfaces of homologous proteins. Due to their importance for the stability of the complex, residues at binding interfaces generally represent essential functional areas and thus may be preserved during evolution. It is an unsettled issue whether binding interfaces are more conserved than the rest of the protein surface [55, 56, 152]. Interestingly, we found that residues in the overlapping part of the protein-protein and protein-ligand interfaces are more conserved than non-overlapping residues, see figure 56. This finding differs from those reported by Davis and Sali who observed a lower conservation for overlap residues in comparison with non-overlap residues [151]. We do not expect that the difference can be ascribed to the different methods used to characterize evolutionary

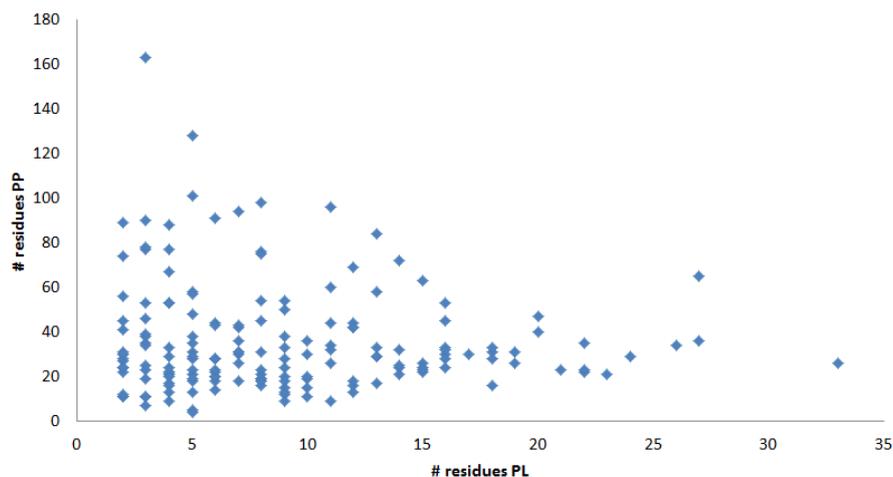


Figure 55: Shown is the absolute number of interface residues in the PP complex against the absolute number of interface residues in the corresponding PL complex.

conservation. Instead, a possible explanation may be that Davis and Sali analyzed a dataset that allowed for a larger evolutionary distance of corresponding protein-protein and protein-ligand pairs.

Secondly, it is well-known that not all interface residues are equally important for the stability of the interacting region of a protein complex [22]. Often, there exists a small subset of residues, termed hot-spots, that has a larger contribution to the binding affinity than the remaining amino acids. Consequently, such hot-spots are being considered as favorable targets for inhibition of protein complexation [153, 154]. They can be predicted by computational methods with typical accuracies around 70% [155]. We found that hot spots are underrepresented in the non-overlap region of a protein-protein interface (37% hot-spot, 63% non hot-spot) whereas they are equally abundant as non-hot spot residues in the overlap region (48% hot-spot, 52% non hot-spot). This observation can be interpreted that due to their relatively small size, ligands engage showing a relatively larger number of contacts with energetically important residues than corresponding protein binding partners.

The distribution of the protrusion index shows clearly distinct distributions for overlapping and non-overlapping residues, see figure 57. In agreement with previous results [76], the relatively lower values found for overlapping residues reflect that they tend to be located in concave structural clefts at the binding interfaces. In contrast, non-overlapping residues tend to have higher protrusion values indicative of exposed locations.

Figure 58 shows the relative surface fractions for overlap and non-overlap residues. Residues in the overlapping region have a slightly higher tendency for low surface accessibility, which can be explained by the higher preference of being located in pocket regions. The density feature was computed following the work of Illingworth *et al.* [156] who reported that residues within ligand binding sites tend to have a higher frequency of contact neighbors than surface

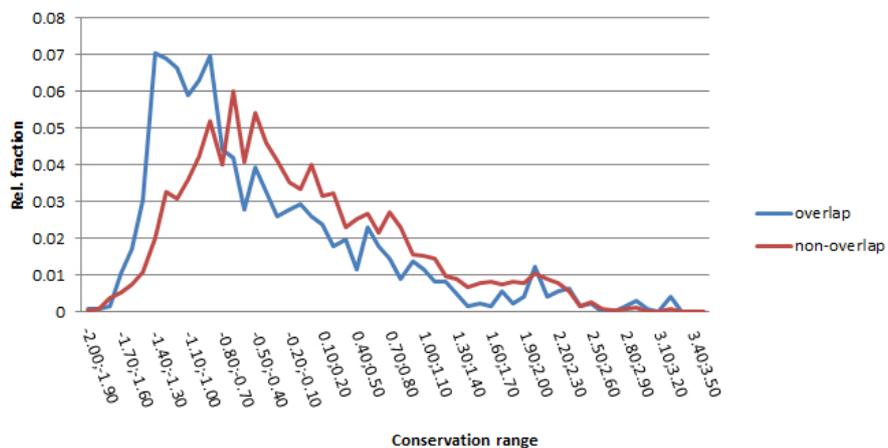


Figure 56: Distribution of conservation ranges obtained from the Consurf web-server for overlap and non-overlap residues for PP interfaces. Negative values indicate residues that are more conserved.

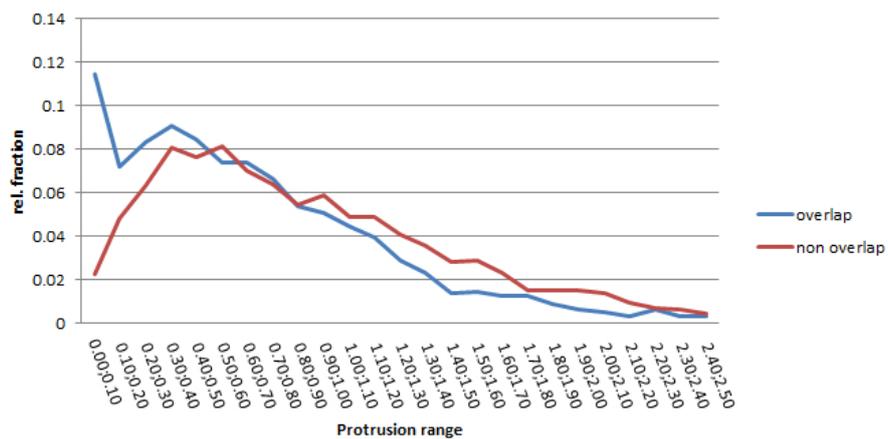


Figure 57: Distribution of protrusion ranges for overlap and non-overlap residues of PP interactions. Values close to zero indicate buried residues.

residues in general. A further study observed that the density values for hot-spot residues turned out to be significantly higher than for non hot-spot residues [144].

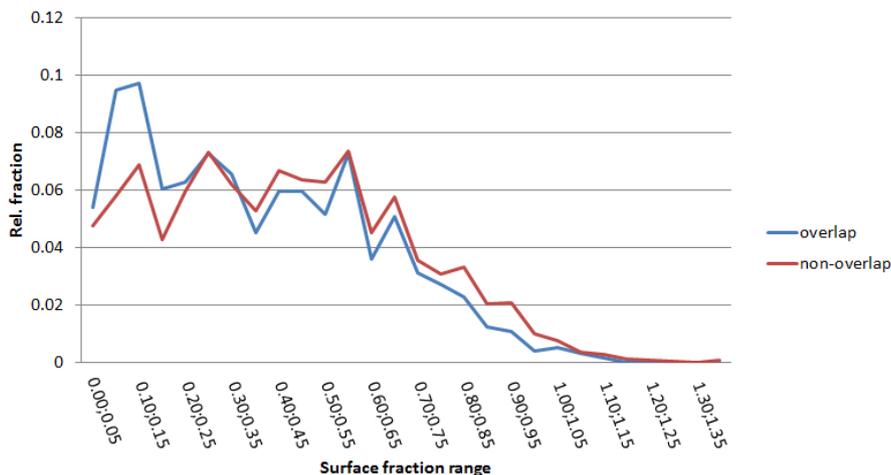


Figure 58: Distribution of surface fractions for overlap and non-overlap residues for PP interfaces.

In the following, we derived a statistical classifier using random forests [157]. Figure 59 shows how often individual features were used in the decision trees. In summary, the protrusion index, the conservation score and surface fraction turned out as promising features for distinguishing overlap and non-overlap residues. The accuracy of single residues according to this  $O/N$  classification was higher (67%) than the random value (50%) for a binary decision between two classes of the same size. Table 9 provides an overview of the assigned classifications.

	Observed overlap	Observed non-overlap
Predicted overlap	TP: 672	FP: 1533
Predicted non-overlap	FN: 424	TN: 3186

Table 9: Confusion matrix for our prediction. Here, TP (true positive) and TN (true negative) denote the number of correctly predicted overlap residues and the correctly predicted non-overlap residues respectively. FP (false positive) and FN (false negative) refer to wrong predictions of overlap and non-overlap residues.

Subsequently, we tested whether the values of neighboring individual residues can be used in a "patch" analysis to boost the accuracy of the prediction for the central residues of this patch. For this, we measured the coherence of an overlapping region in the PP interfaces. To this end, for every PP interface, the Euclidean distances between the heavy atoms of all residues were calculated and assigned to clusters. A cluster consists of a set of residues in which every residue has at least one neighboring residue in the same set within a distance of 5 Å. We found that 82% of the overlapping regions contained only one cluster, 13%

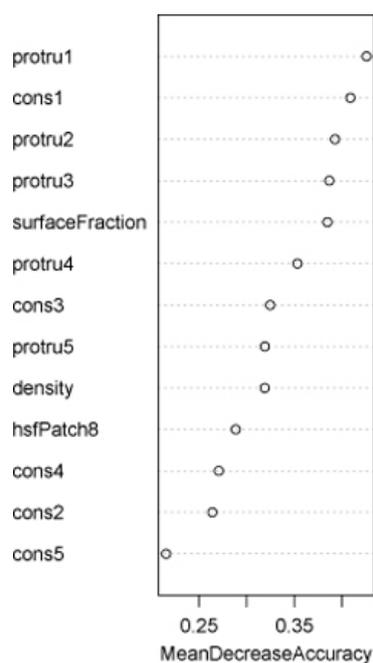


Figure 59: MeanDecreaseAccuracy reflects the suitability of a feature as a reliable predictor. In the diagram, this quality decreases from top to bottom. Cons#n refers to the conservation score of the n-th nearest surface residue starting from the central residue (n=1: central residue itself). Analogously, protr#n refers to the protrusion value of the n-th nearest residue starting from the central residue (n=1: central residue itself). Density and surfaceFraction describe the contact density and the surface fraction of the central residue. hsfPatch8 denotes the relative frequency of predicted hot-spots in a surface patch of size 8.

contained two clusters, and the remaining 5% contained three clusters. This observation indicates that overlap residues are not spread out over the interface but are located close to each other. Thus, we applied surface patches as described in the method section as a further means for characterizing this class of residues. We tested patches of  $n = 5$  to 8 residues around all central residues that were predicted as overlap residue ("O"). Consequently, the patches may contain  $O/(O + N)$  ratios of  $1/n$  to  $n/n$  of "O" residues. Figure 60 shows the frequency of 7-residue patches with different  $O/(O + N)$  ratios and the achieved coverage. The respective statistics for the other patch sizes are available in the supplementary material, see figure 77.

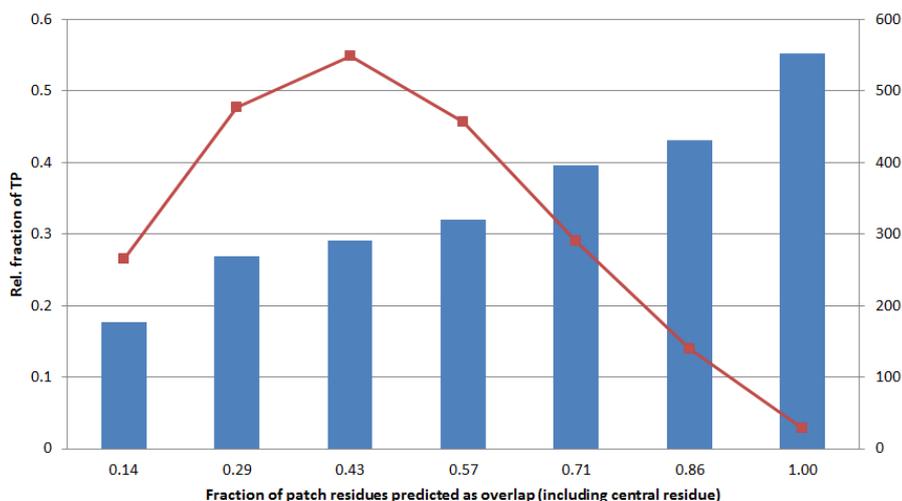


Figure 60: Ratio of positive classifications for patches with increasing amount of predicted overlap residues. The red line represents the coverage for every patch in absolute numbers, e.g. there are about 300 patches with a relative fraction of 0.71 for true positive overlap predictions.

Interestingly, there exists a non-negligible fraction of patches with a predicted  $O/(O + N)$  ratio of 1. Such patches boost the accuracy for the central residue to belong to the real overlap region up to 55%. Examples for structures in which central residues of patches with  $O/(O + N)$  ratio of 1 actually predict true overlap residues are 1Y48 E:I, 1OO9 A:B and 1FAK H:I. All of these pairs describe a complex between a protein and its protein inhibitor. In the corresponding PL complexes the inhibiting protein is replaced by a small molecule taking over the same role. Among the incorrect predictions with a  $O/(O + N)$  ratio of 1.0, the pair 2OL4 B - JPN and 1NHG B:D turned out to be a special case. Examining the PP structure revealed that the protein chains B and D were split into two chains due to residues without electron densities but obviously forming a single chain turning the supposed complex into a monomer. The prediction process recognized four patches with  $O/(O + N)$  ratio of 1 which were all wrong. Leaving out this pair improves the fraction of true positives for the maximum overlap up to 67%. Among the 175 PP/PL tuples considered, patches with 0.86 or 1.0 ratio were found in 99 cases indicating a coverage of

about 60%.

We then applied the prediction algorithm to the entire interface dataset that is currently stored in our ABC<sup>2</sup> database. To this end, we defined one chain for every interface as reference protein. We then generated surface patches of size 7 for the interface residues of the reference protein and focused on patches where the center residue is predicted as overlap and the total number of overlap residues is at least 5. Altogether we found 54809 of these patches belonging to 10156 interfaces. The distribution of surface patches is shown in figure 61.

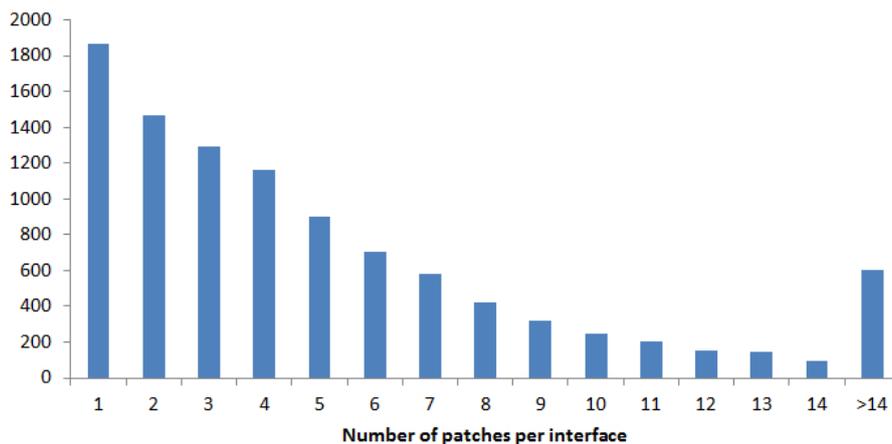


Figure 61: Distribution of the number of surface patches (here: at least 5 overlap residues out of 7) per protein-protein binding interface.

For a characterization of these interfaces with respect to their biological meaning, we clustered the corresponding sequences from all reference proteins to exclude redundancy to get an overview to which sorts of proteins the ones with overlap patches at the interface belong. The non-redundant set of sequences was analyzed regarding the Gene Ontology (GO)-terms for the involved proteins. In tables 10, 11 and 12 we list the most frequent GO-terms found for the three GO categories function, biological process and cellular component.

Function	Frequency
protein binding	341
protein homodimerization activity	53
identical protein binding	27
magnesium ion binding	20
DNA binding	18
zinc ion binding	17
sequence-specific DNA binding transcription factor activity	13
ATP binding	12
flavin adenine dinucleotide binding	12
transcription activator activity	12

Table 10: Term frequencies for GO function.

Biological Process	Frequency
signal transduction	39
blood coagulation	32
positive regulation of cell proliferation	20
platelet activation	20
transcription, DNA-dependent	19
protein phosphorylation	18
transcription from RNA polymerase II promoter	17
apoptosis	16
proteolysis	16
protein homotetramerization	16

Table 11: Term frequencies for GO biological process.

Cellular component	Frequency
cytoplasm	129
cytosol	121
nucleus	96
plasma membrane	69
mitochondrion	61
extracellular region	49
nucleoplasm	36
extracellular space	33
nucleolus	26
membrane	24

Table 12: Term frequencies for GO cellular component.

For a practical case study, we selected all terms from the biological process category containing the expression "apoptosis", see table 13 as these terms may be related to cancer therapy and collected all corresponding interfaces with their predicted overlap residues. Such a list may be an useful source for identifying potential drug targets.

Table 14 lists all PDB entries that contain at least 3 of the terms related to "apoptosis" and contain predicted overlap patches on one of the protein chains. Interestingly, all but the glyoxalase protein refer to very prominent proteins with central roles in the apoptosis machinery. Based on figure 60, the ratio of true positive ligand binding patches among should be at least 0.4.

Table 15 visualizes the predicted "overlap" surface patches for the eight complexes listed in table 14. Most of the predicted residues form a coherent area on the surface region. Only 1PYOD exhibits a cluster of overlap residues and a single overlap residue. The size of the overlap region ranges from rather small and compact as in 1PQ1A to large as shown in 1OLGA. Interestingly, some of the complexes also contain small ligands. In 1RE1B and 2C2ZB, a small molecule binds to the predicted overlap regions supporting the reliability of the prediction method. In the latter complex the small molecules beyond the overlap area are located in the border region of the protein-protein complex. In 1BH5B, the predicted overlap region does not correspond to the binding site for

Biological processes	Frequency
apoptosis	16
anti-apoptosis	12
induction of apoptosis	8
negative regulation of apoptosis	8
positive regulation of neuron apoptosis	2
induction of apoptosis by extracellular signals	7
regulation of apoptosis	5
cellular component disassembly involved in apoptosis	5
induction of apoptosis by intracellular signals	4
negative regulation of neuron apoptosis	4
positive regulation of neuron apoptosis	2
DNA damage response, signal transduction by p53 class mediator resulting in induction of apoptosis	2
positive regulation of anti-apoptosis	2
DNA damage response, signal transduction by p53 class mediator resulting in induction of apoptosis	2
positive regulation of anti-apoptosis	2
negative regulation of B cell apoptosis	1
transformed cell apoptosis	1
negative regulation of smooth muscle cell apoptosis	1
induction of apoptosis via death domain receptors	1
positive regulation of thymocyte apoptosis	1

Table 13: Biological process terms referring to apoptosis for the dataset of surface patches.

the small molecules. However, this does of course not exclude the possibility that other ligand molecules may bind to the predicted overlap region.

We emphasize that these predictions do not require the availability of a crystal structure of a given protein-protein complexes. Experimental knowledge about the binding interface, e.g. from chemical shift mapping by NMR or from accessibility measurements is a sufficient basis as input for a prediction by our method. As a caveat to this analysis we note that this analysis is of course limited by the amount of structural data on protein-protein and protein-ligand complexes currently available. This particularly affects the definition of non-overlap residues. It is clearly possible that these residues could be involved in binding alternative, possibly larger ligands. However, the clearly distinguishable properties of overlap and non-overlap residues derived in this study indicate that there may be only relatively few of such cases.

In conclusion, we have presented a new method that analyzes structural and physiochemical features of protein-protein binding interfaces. When given the three-dimensional structure of a protein-protein complex or the structure of a single protein with annotated PP interface, the method is able to identify to

PDB + chain	Protein name	Term frequency
1PQ1A	Apoptosis regulator Bcl-X	9
1RE1B	Caspase-3	9
1OLGA	Tumor suppressor P53	7
2C2ZB	Caspase-8 P10 subunit	7
2TNFB	Tumor necrosis factor alpha	5
1DU3D	apo2l/TRAIL	3
1BH5B	Glyoxalase I	3
1PYOD	Caspase-2	3

Table 14: PDBs related to apoptosis.

which parts of the PP interface small molecules will likely bind. In this regard our method differs from a related method recently presented by Davis [158] that transfers observed ligand positions bound to one protein to the surfaces of related, homologous proteins that may also bind other proteins.

## 5.2 Materials and methods

All interface data and features were retrieved from our ABC<sup>2</sup> database [124] that is based on the structures of biomolecular protein-protein and protein-ligand complexes taken from the PDB database [125]. Figure 62 depicts an overview of the data generation process. For each complex, the ABC<sup>2</sup> database also provides a list of interface residues. These were identified using a distance based approach which considers any surface residue as interface residue if at least one further residue from the partner chain can be found within radius of 5 Å. The main dataset for this study contained a list of PP/PL pairs where one protein may bind either a second protein or a small molecule ligand at the same interface. At first, we compiled a non-redundant set of tuples  $(P_{i1}, P_{i2})$ ,  $(P_{i3}, L_j)$  where  $P_{i1}$ ,  $P_{i2}$  and  $P_{i3}$  are three proteins and  $L_j$  is a small molecule ligand. Valid tuples were required to fulfill four conditions: (1)  $P_{i1}$  and  $P_{i2}$  are members of a protein-protein complex that is deposited in the PDB. (2)  $P_{i3}$  and  $L_j$  are members of a protein-ligand complex that is deposited in the PDB. (3)  $P_{i1}$  and  $P_{i3}$  share at least 40% sequence identity. (4) The aligned positions in the binding interfaces of  $P_{i1} - P_{i2}$  and  $P_{i3} - L_j$  have at least two residues in common. In the following we denote  $P_{i1}$  and  $P_{i3}$  as reference proteins as they determine the relation between the tuples  $(P_{i1}, P_{i2})$  and  $(P_{i3}, L_j)$ . For checking the last condition, the sequences of the reference proteins were aligned to each other. This resulted in a mapping of the respective interface regions. Residue pairs of proteins  $P_{i1}$  and  $P_{i3}$  that belong both to the  $P_{i1} : P_{i2}$  as well as to the  $P_{i3} : L_j$  interface were termed "overlapping" residues. The remaining interface residues of  $P_{i1}$  were termed "non-overlapping" residues. We did not consider PP/PL pairs that contained only one overlapping residue as this marginal overlap was considered being too small for deriving a meaningful statistical classifier. This procedure resulted in a dataset of about 10.000 pairs. However, this dataset may also contain PP/PL pairs where the ligand does not actually compete with the second protein for the interface on protein  $P_{i1}$ , but both  $L_j$  and  $P_{i2}$  may bind simultaneously, possibly in a cooperative manner. As this work focuses on identifying competitive binders, the reference proteins

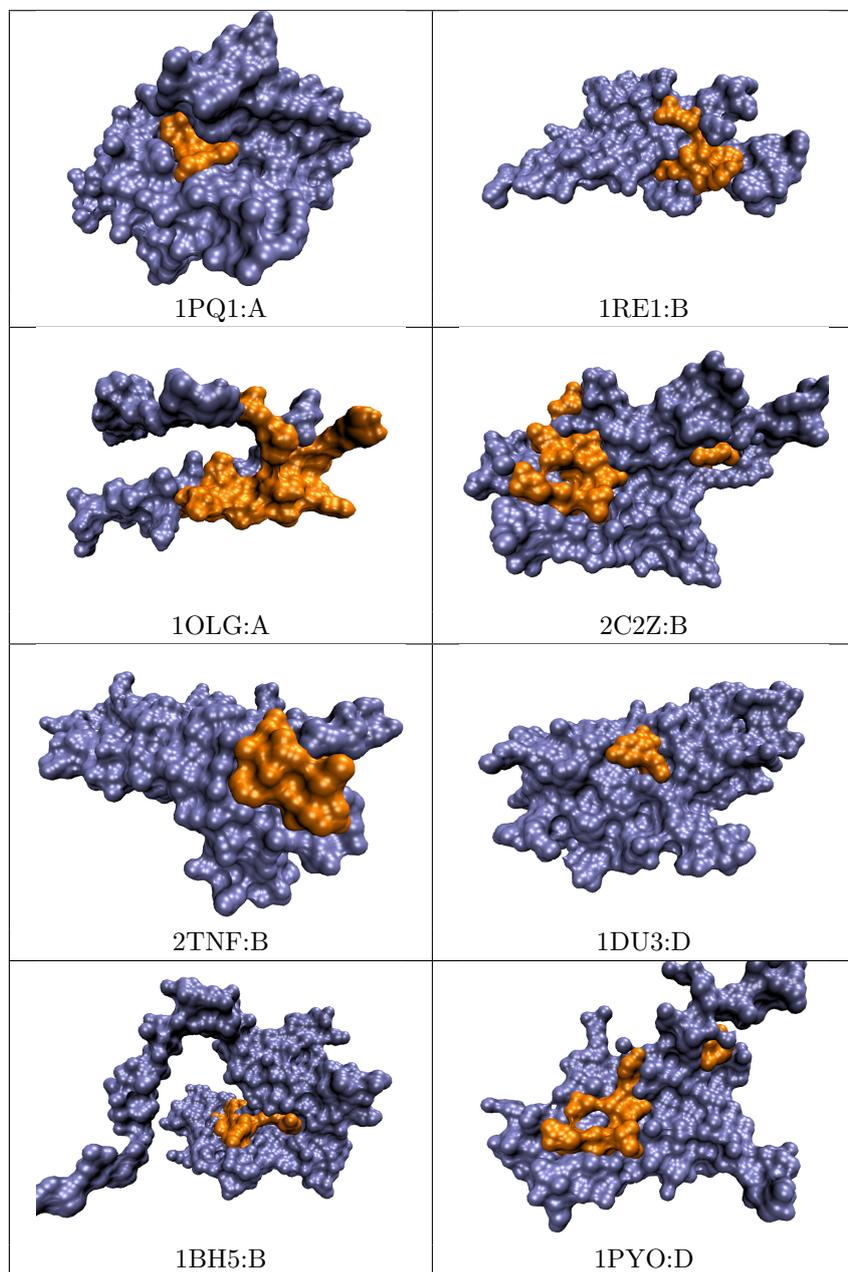


Table 15: Visualization of complexes related to apoptosis. Predicted overlap residues are colored in orange whereas all other surface residues appear in blue.

$P_{i3}$  were geometrically mapped onto the reference proteins  $P_{i1}$  using structural superposition, see figure 63. The resulting transformation and rotation matrices were then applied to the ligand  $L_j$  and all distances between any heavy atoms of  $L_j$  and  $P_{i2}$  were measured. If any of these distances was shorter than the sum of the two atomic radii this indicated a collision, thus  $L_j$  and  $P_{i2}$  are not likely to bind simultaneously to the same binding region on  $P_{i1}$ . Therefore, only competitive PP/PL pairs were kept for further refinement steps. This led to about 1000 pairs of PP and PL interfaces in total. In order to remove sequence redundancy among the PP/PL pairs we clustered the reference proteins  $P_{i1}$  of the remaining PP interfaces using the CD-hit program [159] with a sequence identity cut-off of 40%. This resulted in about 300 clusters. Every cluster contained one or several PP/PL pairs with homologous reference proteins. We did not want to consider short peptide fragments as representatives for the proteins  $P_{i2}$  as such peptides may only cover parts of the full PP interface. Therefore, we either selected from each cluster the tuple with the largest  $P_{i2}$  structure or excluded the entire cluster if it only contained small peptides shorter than 5 residues as reference proteins. Another requirement was that the two interfaces of a PP/PL pair should be similar to each other. To this end, we calculated the sequence identity for the sequence stretches consisting of a combination of overlapping and non-overlapping residues for the PP complex and the PL complex, respectively. Within a cluster we selected the representative with the highest identity score for these generated interface sequences. The final dataset comprised 175 PP/PL tuples. These are listed in the supplementary material, see figure 22.

For a characterization of the binding patterns between a PP/PL pair, we used assignments of pharmacophores. A pharmacophore is defined as a set of structural features in a molecule that is recognized at a receptor site and is responsible for that molecule’s biological activity [160]. For practical purposes, any atom can be assigned a pharmacophore. Table 16 provides an overview of the different types which are used in this work. It is noteworthy to mention that there are feasible combinations of single types. A donor in its protonated state gets the types donor and cationic (+/d). Analogously, a deprotonated acceptor is assigned anionic and acceptor (-/a). Non-carbon atoms in aromatic heterocycles may be a donor within an aromatic system (d/r). Also, they might act as acceptor such as nitrogen in indole-ring in tryptophan, see figure 64.

### 5.2.1 Pharmacophore group assignment

Pharmacophore group assignment is dependent upon two aspects. First, the atom type plays an essential role. Only nitrogen, sulfur and oxygen atoms are suitable to act as H-bond donor and/or acceptor whereas carbon-atoms tend to be hydrophobic. Second, the structure in which the atom is embedded also influences typification. As an example, nitrogen which is included in the backbone of the amino acid, is a secondary amine and may therefore act as donor and acceptor (d/a). An exception to this rule is proline, in which nitrogen is available as tertiary amine, so that it can only become an acceptor (a). The nitrogen atoms in the side chain of arginine form the guanidine group which is known to be a strong base, as the protonated form is stabilized by delocalized electron pairs. In this case, the corresponding pharmacophore type is cationic and donor (+/d). As nitrogen in the indole-structure of tryptophan is embedded in an aromatic system, its type is donor and aromatic (d/a).

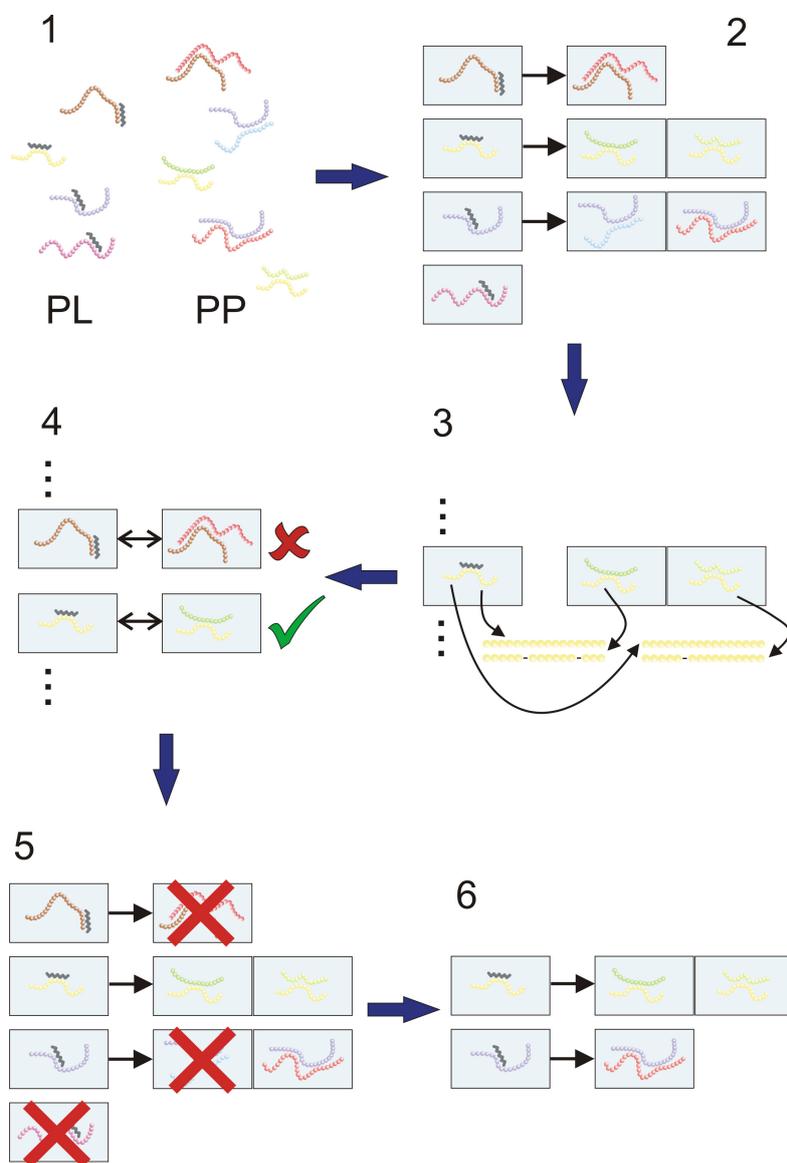


Figure 62: Collection and filtering of protein-protein/protein-small molecule interactions. (1) Data collection of PP and PL complexes. (2) Assignment of PP/PL pairs. (3) Alignment of reference proteins. (4) Search for overlaps. (5) Remove uneligible complexes. (6) Final dataset of PP/PL pairs.

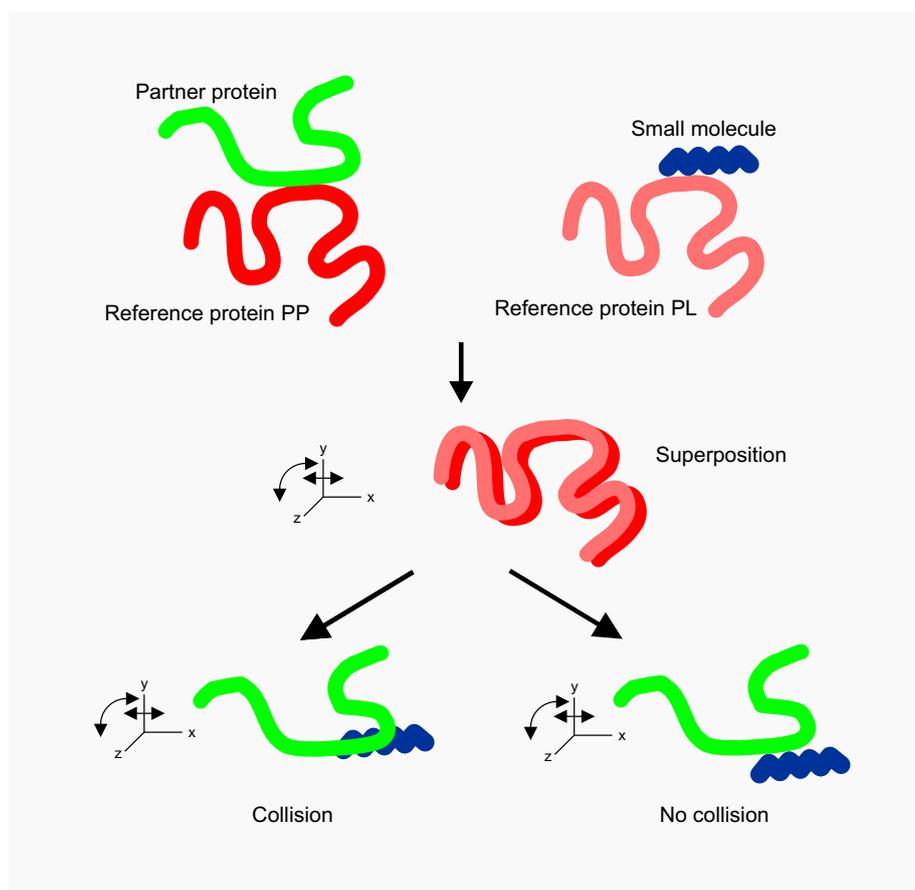


Figure 63: Superposition of a PP/PL pair

In this context, pharmacophores are assigned to protein atoms as well as to atoms of small molecules which are included in the ABC<sup>2</sup>-database. The pharmacophore definition is independent of the type of interface and allows us to compare protein-protein with protein-small molecule interfaces. For the twenty amino acids the assignment was done manually for every atom. The information can be retrieved from the relation *pharmacophoreProtein*. For the vast number of different functional groups in small molecules, an automated way for assignment is required. To this end, we applied pmapper, a program from the JChem package provided by Chemaxon that does assignment of pharmacophore according to a set of pre-defined rules [161]. These rules are based on SMARTS patterns that can be combined with logical operators. For comparison of binding patterns, all contacting atom-atom pairs for PP and PL were collected and converted into pharmacophore-pharmacophore groups. The overlap of the pharmacophore pairs for PP and PL was calculated using Tanimoto coefficient [108]:

$$Tanimoto = \frac{Pairs_{PP} \cap Pairs_{PL}}{Pairs_{PP} \cup Pairs_{PL}} \quad (14)$$

Pharmacophore	Abbreviation	assigned to amino acids
Cationic	+	no
Anionic	-	no
H-bond donor	d	no
H-bond acceptor	a	yes
Hydrophobic	h	yes
Aromatic	r	yes
Donor and acceptor	d/a	yes
Cationic and donor	+/d	yes
Anionic and acceptor	-/a	yes
Donor and aromatic	d/r	yes

Table 16: Overview of pharmacophores.

### 5.2.2 Feature generation

For all members of the final dataset, we computed structural and sequence features of the interfaces that reflect the physicochemical role of individual residues in the complex.

(1) The evolutionary conservation score for a single residue was obtained from the ConSurf-webserver [162]. According to this score, well conserved sequence positions have negative scores and flexible ones have positive scores.

(2) A measure for the energetic contribution of the residues at the binding interface was obtained from a hotspot prediction using an in-house implementation of two knowledge-based prediction algorithms [144, 28] in our ABC<sup>2</sup> database. Benchmarking this approach on a representative set of protein-protein interactions yielded a very similar accuracy as the webserver implementation by Keskin and co-workers [28]. Besides, we computed several structural features of the binding interfaces that characterize their packing density and curvature.

(3) A measure representing the level of burial of residues was quantified by the protrusion value. For this, we used the implementation from ref. [163] that calculates a protrusion value for the  $i$ th atom in a molecular structure as:

$$Protrusion(atom_i) = \frac{V_{empty}}{V_{atoms}} \quad (15)$$

Here,  $V_{atoms}$  denotes the volumes of the atoms within 10 Å radius around atom  $i$  and  $V_{empty}$  represents the value of the remaining empty space in this sphere. An atomic protrusion value of 0 refers to fully buried atoms. The larger the value, the more exposed the atom is to the solvent. The protrusion value for an entire amino acid was computed as the average of the values over all atoms of this residue. Figure 65 visualizes the concept of protrusion and shows a complex color encoded by protrusion values of the atoms.

(4) The number of contacts between an atom and its surrounding atoms was denoted as atomic packing density. We calculated the contact density at an atomic level according to the following formula:

$$Density(residue_i) = \frac{\sum_{j=1}^n contacts(atom_{ij})}{totalAtoms_i} \quad (16)$$

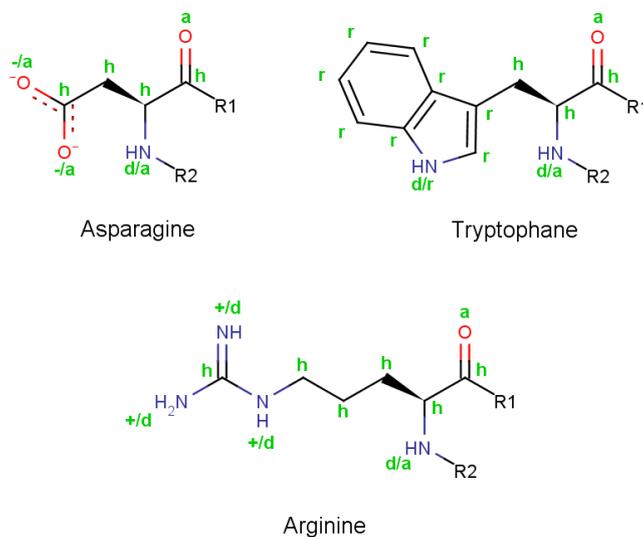


Figure 64: Pharmacophore assignment for three amino acids. The three examples cover all different pharmacophore types that are found in all amino acids.

$Contacts(atom_{ij})$  refers to the number of contacts between a surface accessible heavy atom  $j$  from residue  $i$  and other heavy atoms from residues from the same chain within radius of  $5\text{\AA}$ .  $TotalAtoms_i$  denotes the number of all heavy atoms in residue  $i$ . (5) Also, for all interface residues  $i$  the relative surface fraction was calculated using the program Naccess [104]:

$$rSASA(residue_i) = \frac{SASA(residue_i)}{Total\ surface(residue_i)} \quad (17)$$

Here, the solvent accessible surface area (SASA) was calculated by Naccess for an individual residue  $i$  in a PP or PL complex, whereby the total surface is the surface area of that residue located in the center of a tripeptide and surrounded by two alanines.

Another feature incorporates the direct neighbors of the residue of interest forming a small surface patch on the interface region. This approach was inspired by the work of Thornton *et al.* [164, 165]. A patch is made up of  $n$  surface residues, which consist of one central residue and  $n - 1$  neighboring residues. Thus, a patch describes the microenvironment for a central residue with respect to geometric parameters or physico-chemical properties. We applied a reimplement of the algorithm in ref. [164] and calculated patches for every surface residue in our dataset with sizes between 5 and 8. Figure 62 outlines the generation of surface patches.

### 5.2.3 Random forests

The binary statistical classification of overlapping and non-overlapping residues was based on random forests using a library from Breiman and Cutler implemented in R [157]. A random forest is a fast classifier consisting of a collection of decision trees. To obtain a single prediction, a majority vote is performed

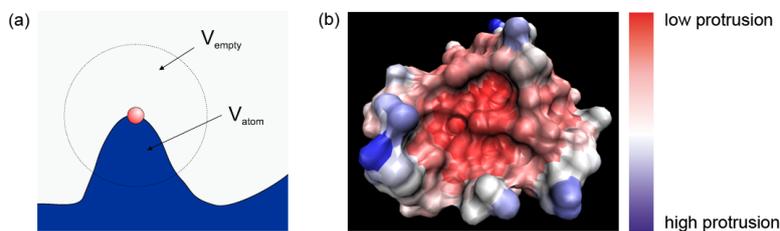


Figure 65: Illustration of protrusion. (a) visualizes formula 15 for  $atom_i$  (red), (b) shows the protrusion values for surface atoms that are encoded by color.

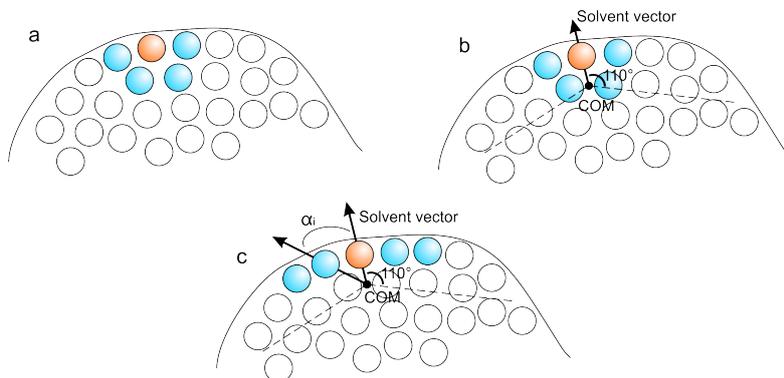


Figure 66: Surface patch generation. (a) Shown is a pre-patch of size  $n = 5$ , consisting of a central surface residue (orange) and  $n - 1 = 4$  nearest neighbors residues (light blue). (b) Using the coordinates of the  $C\alpha$ -atoms of the residues, the center of mass (COM) for this pre-patch is calculated. For any surface residue including the central residue, a solvent vector is defined using the coordinates of its  $C\alpha$ -atom and the COM. (c) The final surface patch contains the central residue and the closest  $n - 1$  surface residues (brown) for which the angle  $\alpha_i$  between their solvent vector and the solvent vector for the central residue is between  $0^\circ$  and  $110^\circ$ .

at the end. Each tree is trained using a different bootstrap sample from the original data set (which is obtained by random sampling with replacement). For each node of a tree a subset of the available features is randomly selected and the best split on these is chosen according to the training set by using the Gini impurity criterion. Each tree is fully grown and not pruned. Because of the bootstrap sampling, about one-third of the original cases are left out of the training set of a specific tree and thus, they are not used in the construction of that tree. This data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to obtain estimates for the importance of individual features. Because of that, there is no need for cross-validation or a separate test set to obtain an unbiased estimate of the test set error in random forests. The idea of such an ensemble classifier is to combine a number of weak learners to create a single strong learner. Here, the random forests were trained with the most promising features identified during this work. The default parameters were employed for the number of trees and

the number of features at each node.

Because of the large imbalance between the number of overlapping and non-overlapping residues in our data set (see table 9), we obtained a highly unbalanced prediction error between the two classes when using the whole data set. In order to balance the two class error rates we applied a down sampling procedure where we randomly drew the same number of data points from the majority class as from the minority class. This was repeated 1000 times.

## 6 Prediction of kinetics of protein-protein interactions

This study was made in collaboration with Naharajan Lakshmanaperumal, a student from our bioinformatics master program, who designed and performed the machine learning part as a research assistant (Hiwi) in our group where he was supervised by me.

In biological tissue protein-protein interactions are responsible for a wide variety of biological functions including signal transduction, catalysis, metabolism, transport and immunological reactions. Modern proteomic high-throughput methods such as yeast two-hybrid screens or tandem mass affinity purification (TAP) allow to map the “interactome” of biological cells [166]. However, these techniques only detect the existence/non-existence of particular biomolecular interactions. A thorough understanding of these interactions requires at least the structural characterization of their binding mode, the thermodynamic characterization of their binding constants, and the characterization of the kinetic association and dissociation constants. Clearly, cellular function is mainly regulated by controlled regulation of gene expression and subsequent splicing and translation. However, it has been estimated that cellular proteins of yeast are involved in about 5-10 interactions with other proteins or larger complexes [167]. Therefore, once a particular protein has been translated in the cell, its involvement in various possible cellular processes is controlled by the different kinetics of alternative interactions with its binding partners [168]. A typical example for this is the binding of an enzyme-substrate complex that determines the rate-limiting steps for reactions along a metabolic pathway. Besides, related proteins may exhibit different binding rates serving as a mechanism for specificity. Many cellular processes such as signal transduction or immunological reactions require a quick response to a stimulus [169], which is achieved by a fast association of the binding partners [38]. An example for this is the bacterial ribonuclease Barnase [170]. After synthesis it forms a tight complex with Barstar acting as inhibitor to prevent Barnase from damaging bacterial t-RNA. Strong binding affinity and fast association are related to each other but do not necessarily mean the same. A high binding affinity can also be achieved through slow dissociation. However, this option is not suitable for proteins that require a subtle control of their binding state as slow dissociation results in a more permanent interaction. Such a behavior is favorable for proteins serving as control instances for cellular processes. For instance, the complex between TATA binding protein (TBP) and eukaryotic promoters dissociates slowly representing a kinetic barrier for TATA binding [171].

A better understanding of protein-protein kinetics also helps improving our general understanding of protein-protein interactions. Common classifications of protein-protein interactions are those into obligate/non-obligate and transient/permanent interactions [13]. Many complexes are assigned to the former class and prediction methods exist for the automatic assignment based on physico-chemical features [103]. The latter class refers to the stability of the protein-protein complex which is related to the binding affinity. Due to lack of a clear definition, no proper assignment of this classification to protein-protein complexes exists so far. The reason for this is that for most complexes no data reflecting binding strength are available. A more comprehensive knowledge

of kinetics may contribute to a clearer definition of this classification. Common experimental techniques include surface plasmon resonance, stopped-flow, isothermal titration calorimetry, and NMR [172] for determining kinetic rate constants.

Members of any pair of protein types will regularly collide in biological cells due to thermal Brownian motion. The kinetic on-rate for such random encounters is given as solution of the Smoluchowski equation [173] as  $10^5$  to  $10^6 \text{ mol}^{-1} \text{ s}^{-1}$  depending on the relative diffusion constant of the two proteins which is related to their masses [170]. A more detailed overview of association constants is depicted in figure 67. Some protein pairs carry strong net

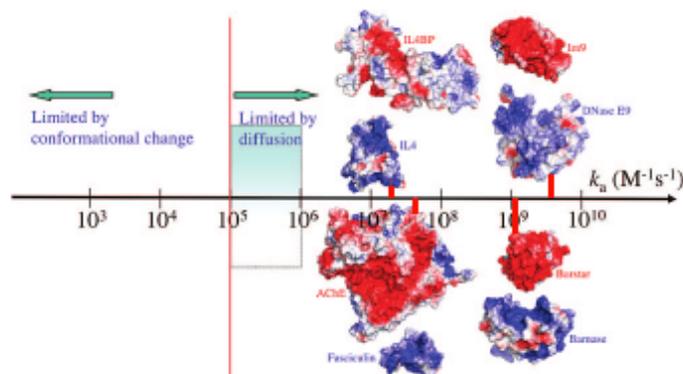


Figure 67: The red vertical line marks the start of the diffusion controlled regime. The shaded range marks the absence of long-range forces. The red bars indicate association constants for some well-known complexes. Picture taken from [38].

electrostatic charges due to an imbalance of positively and negatively charged residues, and sometimes due to additional co-factors such as heme groups. Protein pairs of opposite charges attract each other via a screened Coulomb law where the screening factor accounts for the screening effect of counter ions in the buffer solution. If the measured association constants depends strongly on the ionic strength of the buffer solution, this can be taken as a clear signal for the importance of electrostatic steering on the binding reaction.

Whereas long-range interactions are considered most important for the association reaction, the dissociation reaction is typically governed by short-ranged interactions. Therefore, the size of the binding interface formed during biomolecular assembly has an important role for the stability of the complex. Non-obligate interactions found in signal transduction or enzyme-inhibitor complexes typically involve relatively small interfaces of around  $10 \text{ nm}^2$  or less whereas obligate complexes will have two to three times larger interfaces [13].

Experimental determination of binding kinetics is time and cost-intensive. Thus, such data is only available for a limited number of protein-protein pairs. It is therefore highly desirable to find alternative ways that may allow fast and reliable predictions of kinetics values for protein-protein interaction data. To this end, several computational approaches have been developed. Schreiber *et*

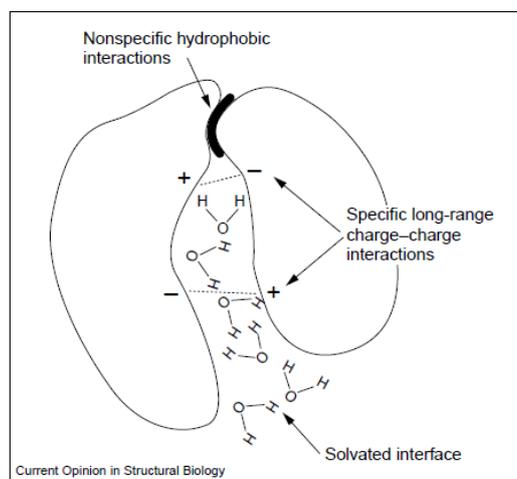


Figure 68: The transition state for association. Parts of the interface are still solvated. The complex is stabilised by specific long-range electrostatic interactions and unspecific hydrophobic and van der Waals interactions. Picture taken from [38].

*al.* implemented the HyPare algorithm that calculates association values based on the Debye-Huckel energy of interaction [170, 174]. Other, computationally more involved, studies aim at deriving kinetic values from biomolecular simulations such as Brownian dynamics simulations. Zhou and co-workers reported an interesting relationship between the electrostatic interaction free energy in the transition state and the association rate [175]. Figure 68 depicts the situation at the binding site in the transition state. However, these methods are mostly tailored at protein pairs that show electrostatic complementarity which is not a typical case [176].

The main aim of this study is to establish a relationship between the kinetics and thermodynamics of a particular protein-protein complex and characteristic features of the bound structure or of the protein compositions. We have focused here on such complexes for which a three-dimensional structure determined by X-ray crystallography was deposited in the Protein Data Bank. We manually extracted experimental data on association, dissociation and inhibition constants from the original literature and investigated the correlation of the above mentioned features with this data. Next, we applied support vector classification to classify a given complex into one of three classes for association and dissociation ("fast", "medium", or "slow") and for the inhibition constant, respectively.

## 6.1 Methods

We compiled a manually curated data set with experimental data on the kinetics and thermodynamics of protein-protein complexes with known three-dimensional structure in the RCSB data bank [125]. We considered dissociation rate constant ( $k_{off}$ ), association rate constant ( $k_{on}$ ) and inhibition constant

PDB id	chains	$k_{on}$	$k_{off}$	$Q_1$	$Q_2$	$pH$
1XDT	RT	7,90E+004	2,00E-003	3	-14	7,4
1EV2	AE	9,64E+004	5,96E-003	4	12	4,0
1KKL	AH	1,30E+005	5,80E-003	-3	-9	8,0
1AVA	AC	1,35E+005	2,50E-004	-12	-2	7,0
1HWH	AB	1,50E+005	N/A	-3	-6	5,0
1UEA	AB	2,00E+005	N/A	-10	4	7,5
3HHR	AB	3,00E+005	2,70E-004	-7	-4	-
1BUH	AB	3,20E+005	2,50E-002	-3	5	7,4
1LDT	LT	3,30E+005	6,60E-004	3	5	8,0
1C0F	AS	5,00E+005	3,20E-001	-13	-3	7,5
1SPB	PS	6,74E+005	3,50E-003	-1	4	7,0
1YDR	EI	8,30E+005	1,00E-001	1	3	6,8
1MCT	AI	9,90E+005	5,00E-008	5	1	8,2
1JSU	BC	1,12E+006	4,10E-002	-3	-2	8,0
1KIG	HI	2,85E+006	5,54E-004	-4	1	7,5
1FLE	EI	3,60E+006	5,80E-004	4	3	8,0
1ACB	EI	4,30E+006	3,60E-006	3	0	7,4
1STF	EI	9,90E+006	5,70E-007	9	1	7,4
1AVW	AB	1,00E+007	1,30E-005	-7	3	8,0
1EFU	AB	1,00E+007	3,00E-002	-16	-9	7,5
1HLU	AP	1,08E+007	7,00E-001	2	-12	7,0
1EAY	AC	1,40E+007	2,00E+001	-9	-4	7,5
1E44	AB	1,60E+007	2,40E-004	8	-13	7,0
1JST	AB	1,70E+007	1,70E+001	-3	4	7,5
1FIN	AB	1,90E+007	2,50E+001	4	-4	7,5
1IAR	AB	1,90E+007	2,00E-003	-5	7	7,4
1MAH	AF	2,70E+007	2,90E-004	4	-9	7,0
1BRS	AD	1,06E+008	1,00E-002	2	-5	6,0
1A4Y	AB	1,80E+008	1,30E-007	-22	10	6,0
1CM1	AB	2,70E+008	9,00E-005	4	-24	7,0
1DFJ	EI	2,80E+008	1,20E-005	4	-22	6,0
1BMM	HI	3,87E+008	6,20E-004	-5	7	7,8
1TBR	HR	7,60E+008	4,30E-004	-14	11	7,8

Table 17: Feature data on association and dissociation kinetics for protein complexes with known three-dimensional structure.

( $K_i$ ) of protein-protein complexes. As a first filtering step, for the complexes with multiple kinetic data in different experimental conditions, we selected the one that falls within the physiological range and if there are multiple entries meeting this criterion, we then selected randomly, resulting in a single experimental data for a complex. Then we selected those complexes where the experimental conditions were near the physiological range i.e. within a  $pH$  range of 5.0-9.0 and a temperature range of 15° - 35°C. If the pH value was not mentioned in the literature, we considered only the temperature. Every interaction involves two protomers which refer to the two polypeptide chains in the protein complex. When considering a protein-protein interaction, only the two protomers involved are relevant. The final data set contained association rate constants for 33 complexes, dissociation rate constants for 31 complexes and inhibition constants for 62 complexes, see table 17 and table 18.

PDB id	Chains	$K_i$	$pH$	Hydrophobic	Polar	Acidic
1GGR	AB	2,84E-05	-	104	71	36
1AVZ	BC	2,00E-05	7,5	67	57	21
1FGL	AB	1,60E-05	6,0	68	68	19
1GLA	FG	1,00E-05	7,0	281	209	91
1L0Y	AB	6,20E-06	7,5	156	198	53
1D2Z	AB	3,20E-06	7,4	112	80	30
1BBZ	AB	1,50E-06	-	28	31	5
1UDI	EI	1,30E-06	8,0	142	96	38
1A0O	AB	1,20E-06	6,5	97	47	34
1EAY	AC	9,00E-07	7,5	95	47	33
1A2K	AD	1,50E-07	7,5	143	111	34
1YDR	EI	8,30E-08	6,8	150	108	46
1BUH	AB	7,70E-08	7,4	166	105	42
1FIN	AB	4,80E-08	7,5	256	176	63
1KKL	AH	4,50E-08	8,0	110	83	36
1XDT	RT	2,70E-08	7,4	215	217	69
1KXV	AC	2,50E-08	7,4	236	259	62
1TMQ	AB	2,10E-08	6,9	224	254	67
1CBW	BD	1,60E-08	7,5	72	83	15
1D4V	AB	1,59E-08	7,4	81	127	38
1HLU	AP	1,30E-08	7,0	215	185	61
1MTN	CD	9,50E-09	8,0	59	71	7
1CA0	BD	7,10E-09	7,5	69	84	19
1VRK	AB	6,80E-09	7,5	61	48	37
1BCK	AC	5,50E-09	8,0	64	65	19
1SPB	PS	5,26E-09	7,0	143	141	24
1KXQ	AH	3,50E-09	7,4	231	265	62
1AHW	AC	3,40E-09	7,0	131	191	48
1AIP	AC	2,75E-09	7,4	250	157	88
1SGP	EI	2,63E-09	8,3	72	136	14
1GL0	EI	2,00E-09	5,0	106	133	15

*continued on next page*

PDB id	Chains	$K_i$	$pH$	Hydrophobic	Polar	Acidic
1TFX	AC	2,00E-09	7,4	96	143	21
1DPJ	AB	2,00E-09	5,0	138	145	49
1LDT	LT	1,97E-09	8,0	97	138	13
1AVA	AC	1,90E-09	7,0	246	204	74
1GPQ	AD	1,00E-09	6,4	93	112	23
1UEA	AB	6,30E-10	7,5	142	137	37
1AZZ	AC	5,10E-10	8,0	156	138	44
1SFI	AI	5,00E-10	8,0	83	125	11
3TGI	EI	4,40E-10	8,0	108	124	24
1HWH	AB	3,40E-10	5,0	146	142	51
3HHR	AB	3,00E-10	-	145	144	52
1VIW	AB	2,90E-10	-	249	297	80
1ACB	EI	2,00E-10	7,4	125	136	20
1KIG	HI	1,80E-10	7,5	105	117	41
1CSE	EI	1,52E-10	-	143	156	19
1BRS	AD	9,40E-11	6,0	74	72	26
1ITB	AB	3,50E-11	8,3	180	171	58
1TAW	AB	2,00E-11	7,5	94	146	17
2SIC	EI	1,80E-11	7,0	171	167	25
3SGB	EI	1,75E-11	8,3	183	224	61
1MAH	AF	1,10E-11	7,0	271	212	58
1FLT	VY	1,00E-11	-	72	71	25
1AVG	HI	3,00E-12	7,4	150	144	52
1BMM	HI	1,61E-12	7,8	107	88	35
1CM1	AB	3,00E-13	7,0	61	46	37
1TBR	HR	2,03E-13	7,8	132	131	51
1STF	EI	6,00E-14	7,4	113	140	23
1MCT	AI	5,00E-14	8,2	92	129	12
2PTC	EI	5,00E-14	8,2	97	144	14
1DFJ	EI	4,30E-14	6,0	218	244	68
1A4Y	AB	7,10E-16	6,0	225	226	72

Table 18: Feature data on inhibition kinetics for protein complexes with known three-dimensional structure.

### 6.1.1 Features

We chose a number of physico-chemical features for which we assume that they might influence the kinetics of a protein-protein complex. Table 19 lists all features that were tested for our prediction approach.

We analyzed several structural and sequence properties concerning the protein-protein interface and the whole complex as input features for the subsequent classification, see table 19.

**Amino acid composition:** The information about the amino acid compositions of the protein-protein interfaces and of the whole complexes were retrieved from the ABC<sup>2</sup> database [16]. The amino acid composition is defined as the frequency of each type of the 20 standard amino acids. The hydrophobic, neutral polar, basic and acidic amino acids were grouped separately according to their

Feature	Correl. with $k_{on}$	Correl. with $k_{off}$
Hydrophobicity of interface	-0.24	0.03
Number of Hydrogen bonds	-0.03	0.19
Interface Area	0.25	0.26
Gap volume index	0.05	-0.19
Total Molecular weight	-0.07	0.14
# hydrophobic amino acids in the interface	-0.05	0.23
# polar residues in the interface	0.14	-0.10
# basic residues in the interface	0.04	0.17
# acidic residues in the interface	0.42	0.17
# hydrophobic amino acids in the complex	-0.09	0.25
# polar residues in the complex	0.04	-0.03
# basic residues in the complex	0.10	0.19
# acidic residues in the complex	0.02	0.15
# hydrophobic contacts	-0.03	0.09
# polar contacts	0.17	-0.01
# oppositely charged contacts	0.24	0.31
Charge imbalance	0.61	-0.14
Mass imbalance		-0.32
# hot spot residues	0.17	0.34

Table 19: List of features that were used for the prediction of protein-protein kinetics.

physico-chemical properties into different groups as listed in table 20 and each one of those group frequencies was later used as an input feature.

**Charge imbalance:** Following [177], the net charge of a protein chain was defined here as the difference between the number of acidic and basic residues in that chain. If the net charges of both interacting proteins are different, there will be an electrostatic attraction between the two proteins. According to the Coulomb law, this attraction is proportional to the product of the two charges, but will be screened in the presence of ions. First, we computed the electrostatic attraction (IEA) as the product of the two charges  $Q_1$  and  $Q_2$ . For simplicity, we also devised an additive scoring function named charge imbalance (CI) that

Group	Amino acid residues
Hydrophobic amino acids	Alanine, Valine, Leucine, Isoleucine, Phenylalanine, Methionine, Proline, Tryptophane
Polar amino acids	Glycine, Serine, Threonine, Cysteine, Tyrosine, Asparagine, Glutamine and Histidine
Basic amino acids	Lysine and Arginine
Acidic amino acids	Aspartate and Glutamate

Table 20: Grouping of amino acids according to their physicochemical properties.

penalizes the complex if the net charges of both chains 1 and 2 have the same sign.

$$Q = \text{number of basic residues} - \text{number of acidic residues} \quad (18)$$

$$IEA = Q_1 \cdot Q_2 \quad (19)$$

$$CI = \begin{cases} |Q_1| + |Q_2| & \text{if } \text{sgn}(Q_1) \neq \text{sgn}(Q_2) \\ |Q_1 - Q_2| & \text{if } \text{sgn}(Q_1) = \text{sgn}(Q_2) \end{cases} \quad (20)$$

**Net Charged SASA:** The net charged SASA is a variation of the charge imbalance method as it considers the SASA of all the charged residues on protein chains 1 or 2.

$$CS = \sum SASA_i - \sum SASA_j \quad (21)$$

$$NCS_{1,2} = \begin{cases} |CS_1| + |CS_2| & \text{if } \text{sgn}(CS_1) \neq \text{sgn}(CS_2) \\ |CS_1 - CS_2| & \text{if } \text{sgn}(CS_1) = \text{sgn}(CS_2) \end{cases} \quad (22)$$

**Mass imbalance:** Mass imbalance is the difference of the masses of the individual interacting proteins. The molecular weights of the individual chains,  $mass_1$  and  $mass_2$ , were retrieved from the ABC<sup>2</sup> database [124].

$$MI_{1,2} = |mass_1 - mass_2| \quad (23)$$

**Gap volume index:** The gap volume index is one of the established features for interface complementarity. The gap volume is the sum of all empty cavities at the binding interface and was computed from the original PDB file using the SURFNET [73] program and keeping all crystallographic water positions. The minimum and maximum radii for the gap spheres were set to 1.0 and 5.0 Å, respectively. The grid separation was set to 2.0 Å.

$$\text{Gap volume index} = \frac{\text{Gap volume } (\text{Å}^3)}{\text{Interface area } (\text{Å}^2)} \quad (24)$$

**Hydrophobicity:** In order to describe the hydrophobic nature of the residues at the interface, we used the Kyte and Doolittle hydrophobicity values [105]. This is a widely used scale where negative values are assigned to hydrophilic residues and positive values to hydrophobic residues. The hydrophobicity of the interface was then calculated as the sum of the hydrophobicities  $h_i$  of all residues  $i$  in the interface.

**Number of contacts:** An amino acid residue of one protein is said to be in contact with a residue of another protein if the distance between the two is less or equal than 5Å. If the two residues involved are hydrophobic, then the contact is termed hydrophobic contact and if they are polar, they are termed polar contacts. The contacts between oppositely charged residues are counted as charged contacts. The mixed contacts between polar and hydrophobic residues were left out from our calculation. The contacts were weighted according to the Kyte and Doolittle scale [105] in order to account for their hydrophobicity.

### 6.1.2 Classification

We employed a support vector machine for building a regression model for our analysis. We employed a support vector machine [178, 179] to classify the complexes. In general, an SVM is a supervised learning algorithm for binary classification of data. For more than two classes of data, multi-class techniques are required. These techniques include "one-against-one" and "one-against-all" approaches [180]. The R package e1071 [181, 182] interfacing to libsvm [183] was used to perform the SVM classification. For multiclass-classification with  $k$  levels,  $k > 2$ , libsvm [183] uses the "one-against-one" approach, in which  $k \times (k - 1)/2$  binary classifiers were trained and the appropriate class was found by a voting scheme.

The complexes were first labeled as fast, medium and slow depending on their association rate and dissociation rate constants. If the value of the association rate constant is smaller than  $10^6 M^{-1} s^{-1}$ , the complexes were classified as slow and as fast for values larger than  $10^8 M^{-1} s^{-1}$ . All values between  $10^6 - 10^8 M^{-1} s^{-1}$ , were classified as medium. If the value of the dissociation rate constant is larger than  $10^{-3} s^{-1}$ , the complexes were classified as fast and if it is smaller than  $10^{-5} s^{-1}$ , they were classified as slow. All values between  $10^{-5} - 10^{-3} s^{-1}$ , were classified as medium. In the case of inhibition constants, the complexes were classified into high, medium and low affinity. The complexes having inhibition constants smaller than  $10^{-10} M$  were classified as high and as low for values larger than  $10^{-8} M$ . All values between  $10^{-8} - 10^{-10} M$ , were classified as medium. We employed three kernels namely the linear (equation (25)), polynomial (equation (26)) and radial basis kernel (equation (27)).

$$K(x_i, x_j) = x_i \cdot x_j \quad (25)$$

$$K(x_i, x_j) = (\gamma(x_i \cdot x_j) + a)^d \quad (26)$$

$$K(x_i, x_j) = \exp^{-\gamma \cdot |x_i - x_j|^2} \quad (27)$$

Here,  $x_i$  and  $x_j$  are the input vectors comprised of the featured vectors,  $\cdot$  is the dot product and  $\gamma$ ,  $a$  and  $d$  are kernel parameters. To obtain an SVM classifier with the optimal performance, the cost parameters  $C$  and were selected by leave one out cross validation (LOOCV) on the dataset based on the prediction accuracy.

### 6.1.3 Model validation:

In order to validate the model, a leave One Out Cross Validation (LOOCV) resampling method was used. In this technique, all members of the dataset except one are selected to be the training set and the remaining one is the test set. The selected model was evaluated using the total prediction accuracy of the model.

$$\text{Overall accuracy} = \frac{TP}{(TP + FP)} \quad (28)$$

The class accuracies represent the percentage of correctly predicted complexes in each class.

$$\text{Class accuracy}_{fast} = \frac{TP_{fast}}{(TP_{fast} + FP_{fast})} \quad (29)$$

$$Class\ accuracy_{slow} = \frac{TP_{slow}}{(TP_{slow} + FP_{slow})} \quad (30)$$

$$Class\ accuracy_{medium} = \frac{TP_{medium}}{(TP_{medium} + FP_{medium})} \quad (31)$$

## 6.2 Results and Discussion

We compiled from the original literature a dataset for kinetic association rate constants ( $k_{on}$ ), dissociation rate constants ( $k_{off}$ ) and inhibition constants ( $K_i$ ), see table 17 and 18, for complexes for which the three-dimensional structure of the complex is available in the Protein Data Bank [125]. The correlation between these three parameters are,  $k_{on}/k_{off} = -0.11$ ,  $k_{on}/K_i = -0.13$  and  $k_{off}/K_i = 0.63$ , see figure 72. This shows, on one hand, that the association and dissociation constants of protein-protein complexes are almost fully uncorrelated. Thus, one expects that they will be determined by different properties of the binding interfaces and/or of the full proteins. On the other hand,  $k_{off}$  and  $K_i$  are highly correlated. This is quite expected as  $k_{off}$  varies over 9 decades compared to 4 decades for  $K_i$ . Using our ABC<sup>2</sup> database [124], we analyzed geometric and global parameters of the bound conformations of these complexes and computed the correlation of these features with the experimental data, see table 19. We report the informative features and the prediction accuracies obtained by using those features or a combination of those features. Based on these results, we performed a comprehensive model selection procedure using single features and also combination of features. The prediction accuracy was measured in terms of Leave One Out Cross Validation (LOOCV) prediction accuracy and class accuracies. For each prediction accuracy, we analyzed whether the corresponding LOOCV class accuracies are balanced across the individual classes. In all cases, the best results were obtained using the radial basis kernel. Table 21 shows the prediction accuracies for the different kinetic classes.

Response	Cost	Gamma	Prediction acc.	Fast	Medium	Slow
$k_{on}$	7	0.5	72.7%	71.4%	69.2%	76.9%
$k_{off}$	14	8.5	61.3%	60.0%	71.4%	42.8%
$K_i$	86	0.17	72.6%	80.7%	68.0%	63.6%

Table 21: Prediction accuracies of the best model.

**Association kinetics:** The highest individual correlation of 0.61 was found between the association rate constant and the charge imbalance between the two proteins, see figure 69 (a). This is as expected: the higher the charge imbalance, the stronger is the electrostatic attraction between the oppositely charged proteins leading to faster association. The next important features are the number of acidic residues at the interface, which then lead to the formation of oppositely charged contacts, and to less hydrophobic interfaces. We then systematically tested combinations of this feature and 1-2 other features. The best model used the features charge imbalance, net charged SASA and the number of acidic residues in the interface. It gave a prediction accuracy of 72.7% when only considering the residues lysine and arginine as basic residues and 75.7% when also histidine residues were considered as basic residues in the

net charge and net charged SASA calculation further increased the prediction accuracy to 75.7%. The prediction accuracy fell to 69.7% if only residues at the protein surface instead of those of the whole protein were considered for the calculation of charge.

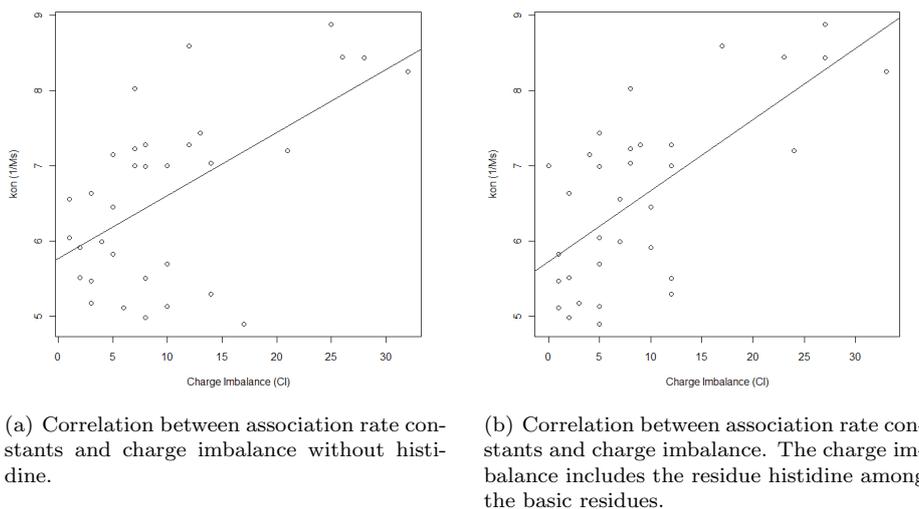


Figure 69: Correlation between charge imbalance and association kinetics

**Dissociation kinetics:** The best model was obtained based on the single feature of mass imbalance. The correlation coefficient between the mass imbalance and dissociation rate constant is -0.32 (figure 70). The model gave a prediction accuracy of 61.3%.

**Inhibition constant:** The correlation coefficient between the gap volume index and the inhibition coefficient is 0.53 (figure 71). The best model contained the gap volume index and certain aspects of the amino acid composition in the whole complex namely the numbers of basic, polar and acidic residues. With this model, we obtained a prediction accuracy of 72.6%.

Binding reactions that take place as random events following diffusion on a flat energy surface have association rate constants in the range from  $10^5$  to  $10^6 M^{-1}s^{-1}$ . Higher association rates require attractive interaction forces between the binding partners [184, 185]. It is well known that long-range attractive electrostatic interactions facilitate such higher rate constants. This effect is known as "electrostatic steering" [185]. Generally, electrostatic interactions are responsible for high rate constants in interactions for which speed plays a decisive role with respect to its function. However, analysis of interfaces by Schreiber *et al.* with his HyPare algorithm relativized the importance of electrostatics. The inclusion not only of the charged interface residues but also of all charged surface residues is not surprising given the fact that electrostatic interactions are long term interactions so that charged residues which are not located in the interface may also participate in stabilizing the interface. Besides,

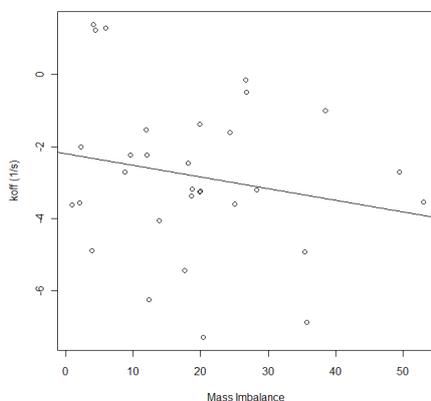


Figure 70: Plot between dissociation rate constant and mass imbalance.

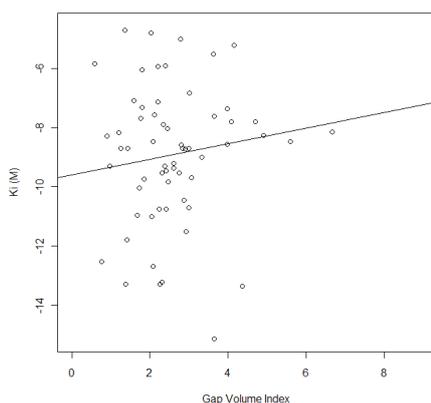


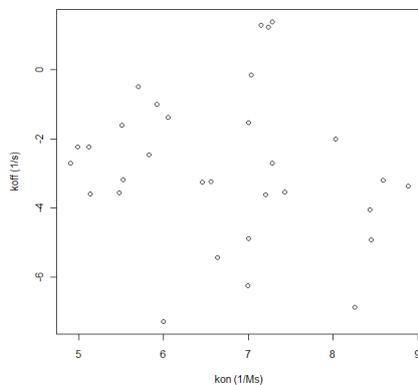
Figure 71: Plot between inhibition constant and gap volume index.

it is discussed that electrostatic interactions forwards formation of the encounter complex facilitating creation of the actual complex.

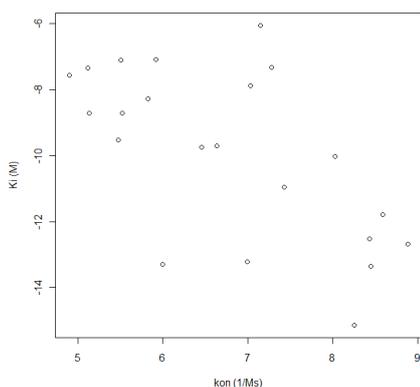
From Figure 72 (a) and (b), we see that the correlation between the association and dissociation rate constants as well as association rate constant and inhibition constants are very low. Figure 72 (c) shows a predictable correlation between the dissociation rate constant and the inhibition constant.

The inclusion of histidine in the basic residue category increases the correlation between the charge imbalance and the association rate constant as seen in figure 69 (b) compared to (a), where it was excluded for the analysis.

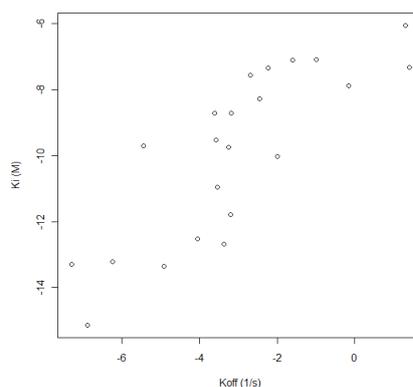
There are certain critical residues in the protein-protein interfaces that account for the majority of binding energy and these are termed the "Hot Spot Region". We calculated the number of hot spot regions in each complex using the HotPoint webserver [28]. The correlation coefficients with both the association and dissociation constants were quite low with 0.17 and 0.34 respectively.



(a) Correlation between association and dissociation rate constants.



(b) Correlation between association rate and inhibition constants.



(c) Correlation between dissociation rate and inhibition constant on a logarithmic scale.

Figure 72: Correlation among kinetic values.

In order to score the charge imbalance feature, we first had a scoring function based on the Coulomb's law, considering the product of the two charges of the proteins. But the prediction accuracy obtained with such a scoring function was found to be lower than that of the one obtained using the scoring function given in equation 3. The experimental dataset is limited and hence this method should be tested with a much larger dataset. We had to include experimental data obtained within a wide pH range although the pH certainly has an effect on the kinetic parameters by affecting the titration states of basic and acidic residues on the protein surface. It may also be problematic to mix different sorts of experimental data that were obtained by different experimental methods. Certainly, the residues at the binding interface will have the largest direct effect on the association kinetics. Still, some complexes such as the barnase-barstar complex also appear to involve other surface patches in the association process.

### 6.2.1 Conclusions

We compiled a novel dataset for the association and dissociation kinetics of protein-protein complexes with known structure. Furthermore, we developed a classification method for predicting the kinetics of protein protein interaction based on the structural and sequence properties of protein complexes with known structure. These structures could either be determined experimentally or result from protein-protein docking runs. We expect that the results will be highly useful for biochemists and pharmacologists as a first hand predictive method for the study of kinetics of protein-protein interaction.

### 6.3 Website

A website was created allowing a prediction of kinetic classes for an input structure of protein-protein interfaces based on PDB format. The main menu is shown in figure 73. The user can either search for a interface using a PDB and chain identifiers for pre-calculated predictions of kinetic values for interfaces from the ABC<sup>2</sup> database or he can upload an individual file in PDB format containing a protein-protein interface for which a prediction is conducted. A prediction request for an uploaded structure is forwarded to an external R-application under which the support vector machine is running [181]. Due to the simplicity of the classifiers for the machine learner, the result is available within a few seconds. The output consists of an assignment of one of the three classes "slow", "medium" or "fast".

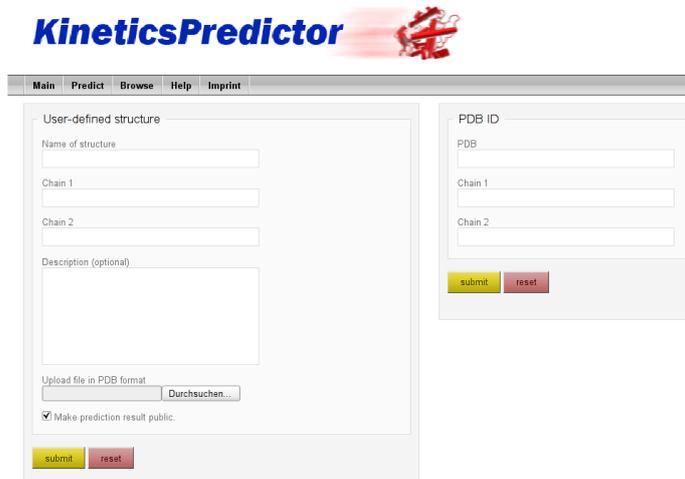


Figure 73: Web page for prediction of kinetic values.

Beside the execution of prediction queries, a user may also browse through the list of pre-calculated kinetic values for interfaces of the ABC<sup>2</sup> database. Some filter options allow for selection of certain interfaces.

The technical concept of the website follows the model-view-controller (MVC) framework [110] which is described in subsection 3.5.

## 7 Outlook

In the current version, the ABC<sup>2</sup> database considers amino acid sequences and ligands as interaction partners. In order to come closer to a complete picture about biomolecular contacts, further types of interactions should be incorporated such as protein-DNA or protein-RNA complexes. Also, these types of complexes become more and more interesting as the number of structural data is steadily growing. Besides, the number of complexes for which kinetic data is available is very low in the current version. It is desirable to extend such kind of data to collect a more comprehensive dataset allowing further analyses such as with machine learning methods.

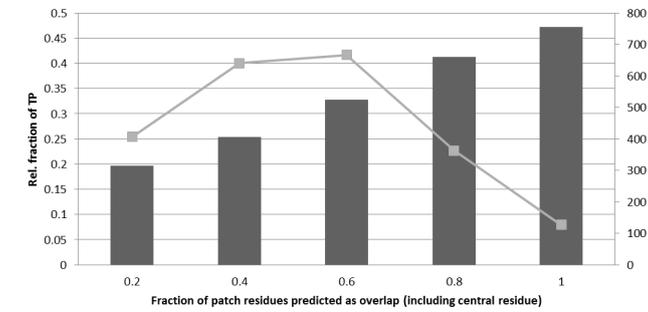
The website can be upgraded for more analysis methods to the user. However, incorporating such methods may raise technical difficulties as they usually require a lot of computation power. A possible solution to that problem might be to combine the database with a framework for conducting bioinformatics related tasks which is running locally on a user's machine. Such an application is currently under development in our workgroup. Besides, incorporating WSDL or SOAP protocol to allow external programs to directly access the server facilitates automation of query requests.

The current focus of the database lies on structurally resolved interfaces. An approach for further research might be to enrich the database with interface data for which no structural information is available. A combined analysis of structural data with interface domains may provide new insights into the relations of biomolecular contacts with each other.

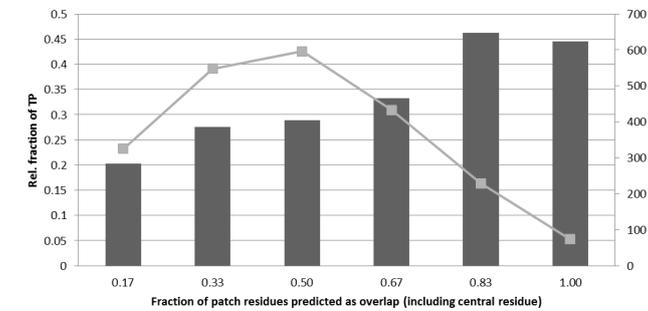
Also, the characterisation of protein-protein complexes using graphs becomes more and more popular. As an example, Liu and Li applied bipartite graphs to describe binding features among types of protein-protein interactions [186]. Tuncbag *et al.* analysed hot spot residues using minimum cut trees [187]. The ABC<sup>2</sup> database can be easily extended with network information about protein-protein interactions.

Referring to section 5, further research on PP and PL complexes may focus on features of ligands. One question to be answered is whether ligands that are prone to bind either at the interface of a protein-protein interaction are different from other small molecules.

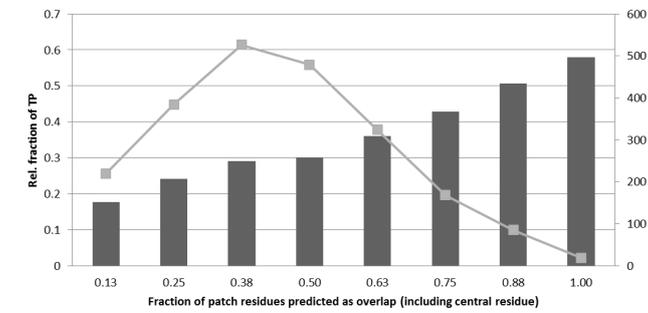
## 8 Supplementary material



(a) Maximum overlap patch for patch size 5.



(b) Maximum overlap patch for patch size 6.



(c) Maximum overlap patch for patch size 8.

Figure 74: Maximum overlap patches with patch sizes 5,6 and 8.

PDB for PL	PDB for PP	PDB for PL	PDB for PP
1MFI B FHC	1GIF B A	1O9T B ATP	1P7L A B
1UBH S MPD	1YRQ A H	1KMH B TTX	1BMF F B
1LBC A CYZ	1S7Y B A	2C97 D MPD	1HQK E D
2HQU C DUP	2OL1 A C	1TXC A 2AN	1IFV A B
1A05 B IPM	2AYQ B A	2HA3 A P6G	1C2O D A
1JI5 B MPD	2CHP C D	2B0U B MPD	1S4Y B A
1C50 A CHI	1YGP A B	2JH0 D 701	1E0F E J
1GZR B C15	2DSQ I G	1O5D H CR9	1FAK H I
2BUQ A CAQ	3PCD A M	1SUP A PMS	1Y48 E I
1XKD A NAP	2D4V B A	1PFK A ADP	6PFK C D
1V84 A NAG	1KWS B A	1O6T A MES	1O6S A B
1LVW D TYD	1IIN A B	1RHM B NA4	1I3O B E
1DHK B NAG	1BJQ C A	1EST A TOS	1MCV A I
1HJ1 A PMB	1U3R A B	1HX0 A AC1	1BVN P T
1CY2 A TMP	1CYY A B	1LOJ A MPD	1I8F C B
1AR1 A LDA	2OCC A B	1CGY A MAL	1D7F B A
1ANK A AMP	4AKE A B	1BIW B S80	1OO9 A B
1USR A SIA	1E8U B A	2P95 A ME5	1P0S H E
1Q6Y A MPD	1VGQ B A	1BMQ A MNO	1SC1 A B
1U3T B CCB	1U3W B A	4VGC B SRD	1HJA B I
1QIW A DPD	2BE6 A D	1NIP B ADP	1G21 H G
1RYD A GLC	1H6A B A	1MPF A C8E	1OPF B A
1XR8 A PG4	2NX5 A D	1GG6 C APF	1N8O C E
2J9C C ATP	1HWU B A	1ICR A NIO	1KQD A B
1CPC A CYC	2J96 B A	1HVV A TAR	1JTH D A
1S57 B EPE	1B99 F C	1I9B D EPE	1YI5 C H
2IPF A TRS	2IPJ B A	2FNW A REP	1EZL C D
35C8 L NOX	15C8 L H	1WV7 T FUC	1AHW F A
1BCS B CST	1GXS D C	1G5N A SGN	1DM5 B D
1LOJ C MPD	1N9S C D	1XJI A D10	1BRR A C
1LOJ E MPD	1TH7 L K	2C4L A SIA	1NMA N H
2H6Y A MPD	2BTO T A	2I17 A CIT	1MI3 B A
1H48 C CDI	2AMT C B	1W5F B G2P	1RQ7 B A
1RE2 A NAG	1CKG B A	1ZOM A 339	1XX9 B D
1DBN B NAG	2DVG C B	1RTK A GBS	1DLE A B
2IWZ A 6NA	1W0I B A	1FQ6 A GSC	1DPJ A B
9RSA B ADU	1DFJ E I	4LIP D CCP	1QGE D E
2H0T A EPE	2G2U A B	1SPQ A PEG	7TIM B A
1IT6 A CYU	2O8A A I	1Y11 A 1PE	2EV4 B A
1YZW C PEG	1ZUX D B	1ZRK A 367	1ZJD A B
1GOY A 3GP	1X1U A D	1UX0 A THU	2FR6 C D
2AZ5 B 307	2TNF B A	1TB6 I MPD	2GD4 I H
1Q4J A GTX	1OKT B A	2DJH A UM3	2FHZ B A
2OIZ A TSR	2AGY B D	1RZH H CDL	1EYS H M
1SVL C ADP	2H1L E F	1BWO A LPC	1UVC A B
2APX A MLA	2AQ3 G A	2J6E B MPD	2IWG D E
2DQV A GAL	1SUV D F	2OM9 A AJA	2PRG A C
1G4I A MPD	1FX9 B A	1FLJ A GTT	1G6V A K
1LIN A TFP	2BL0 B A	1GMR A 2GP	1AY7 A B
1X29 B PMG	2AY5 A B	2FP7 A NDL	2IJO A I
2AY9 B 5PV	1ASL B A	2G7Y A MO9	1ICF A I
1YZW C PEG	1XMZ B A	1EKX C PAL	1GQ3 C B
2HG8 A MLE	2COG A B	1TR5 A THP	1SND A B
1L9B L HTO	2GMR L M	1Q60 B LG6	1XBY B A
1MBQ A BEN	1BZX E I	1L9H A HTO	1F88 B A
1YRX B D9G	2IYG B A	2OL4 B JPN	1NHG B D
1G8I A P6G	2I2R H D	2IWZ A 6NA	2GQD B A
2J8C M GGD	1PST M H	1O4H A 772	1A09 A B
1NGP L NPA	1P4I L H	1WV0 A BN4	1PYG B A
1KYN A KTP	1FI8 A C	6RNT A 2AM	1BVI A C
1XXS B STE	1PA0 B A	2J7L A XC2	1QPX A B
2IW6 A QQ2	1G3N A B	1IZ2 A SUM	2D26 A B
2NY0 A HEZ	2NY7 G H	2FP7 B NDL	2IJO B I

*continued on next page*

PDB for PL	PDB for PP	PDB for PL	PDB for PP
2FMH A TRS	1VLZ B A	1Y2F A WAI	1F46 B A
2PL7 B HTG	2GVM B A	2A01 A AC9	1AV1 B A
1M2Z A BOG	2AAX A B	1I5G A TS5	1O81 B A
1A8J L PME	1MCI B A	2YXJ A N3C	2P1L A B
1OAU J DNF	1A6U H L	1U0H B ONM	1AB8 B A
1TI1 A D12	2HI7 A B	2GJ6 D 3IB	1QRN D E
2C01 X ATP	2BEX C A	1BQI A SBA	1STF E I
1CLS D DEC	1G0A D B	2UUE B GVC	1JSU B C
1JTK A THU	1R5T A B	2B45 X EPE	1T6G C A
1KJ1 D MAN	1MSA B C	2DCY A TAR	2B42 B A
1XEY A GUA	1PMO D C	1F42 A MNB	1F45 A B
2CZ5 B CIT	1X1Z A B	2GUI A PEG	2IDO A B
3LJR A GGC	2C3N A B	1H1B A 151	1PPF E I
2C97 B MPD	1W29 C D	1BG9 A GLC	1AVA A C
1L7Z A MYR	1KQM C A	1G0T A PEG	1JZD B C
1RFX C PEG	1RH7 C B	2HXM A 302	1UGH E I
1RH7 C P6G	1RFX B C	2G2Z A COZ	2CUY B A
1GKA A D12	1OBQ B A	1WB8 A PMS	1B06 A B
2CLO X TRS	1HE8 B A	1BLC A CEM	1OME A B
1HUR A GDP	1R8Q A E	1EWY C FAD	2PVO D A
2FYD A PG4	1HFY A B	2OPY A CO9	1NW9 A B
1S9Q B CHD	2GPV D B	1ZL0 A TLA	1ZRS B A
1UTM A PEA	1SGF G B	1V3V B 5OP	2J3K A B
1SGC A CST	1SGR E I	2GOO C NDG	1NYS A C

Table 22: List of PL/PP pairs. The first chain identifiers for PP and PL respectively denote the reference proteins.



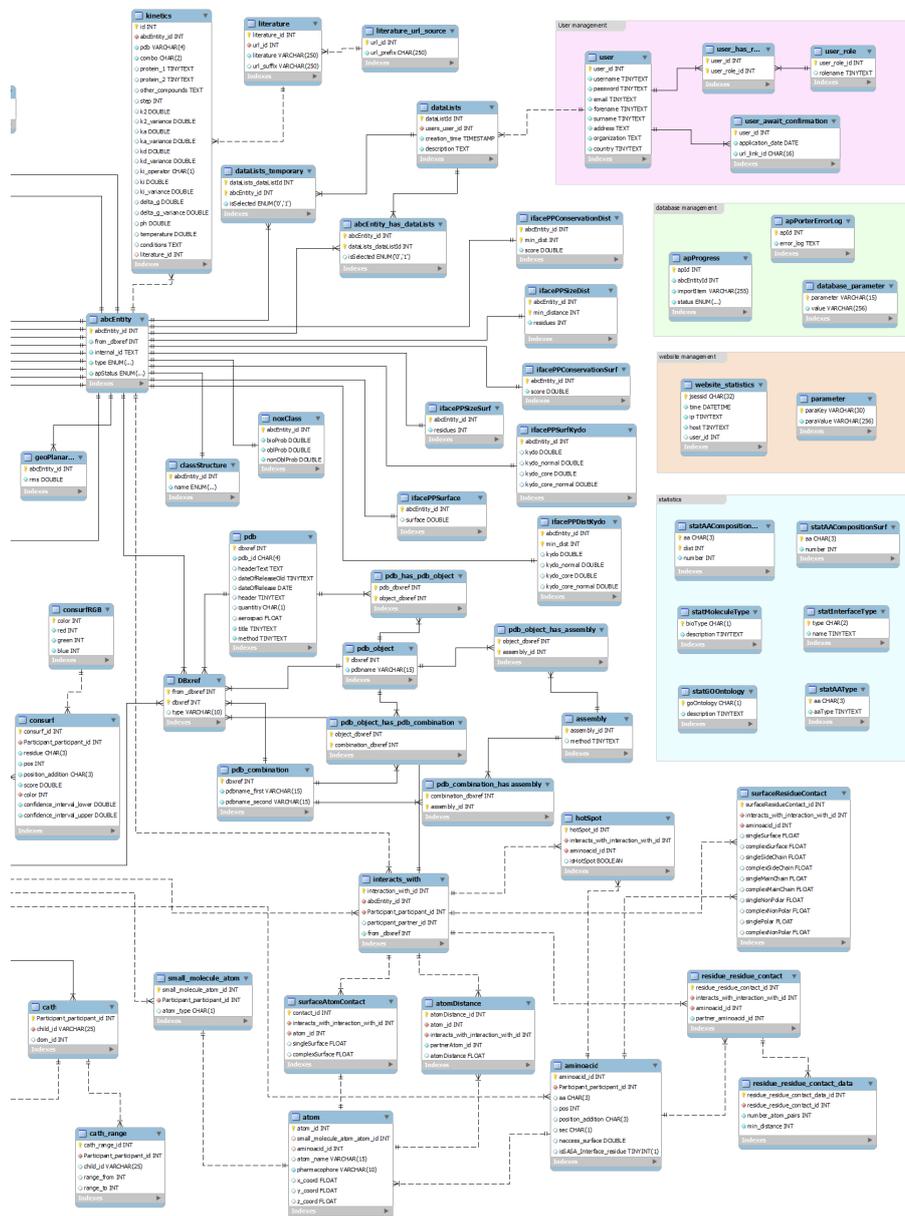
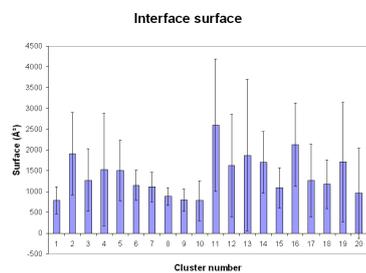
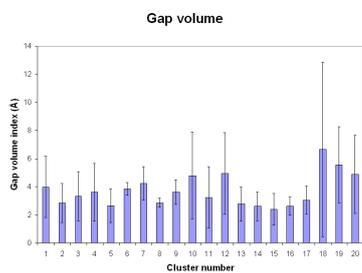


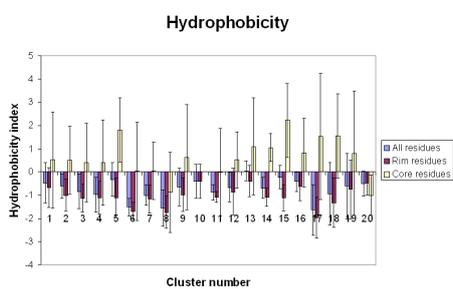
Figure 76: Database diagram from ABC<sup>2</sup> database, part two.



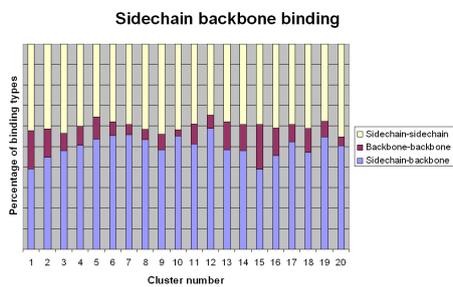
(a) Cluster 60% functional similarity - Interface surface



(b) Cluster 60% functional similarity - Interface gap volume index



(c) Cluster 60% functional similarity - Hydrophobicity



(d) Cluster 60% functional similarity - Sidechain/backbone contacts

Figure 77: Features for clusters with 60% functional similarity.

## References

- [1] R. Ramakrishnan and J. Gehrke. *Database management systems*. McGraw-Hill Science/Engineering/Math, 2003.
- [2] D.E. Koshland Jr. The key–lock theory and the induced fit theory. *Angewandte Chemie International Edition in English*, 33(23-24):2375–2378, 1995.
- [3] D.D. Boehr and P.E. Wright. BIOCHEMISTRY: How Do Proteins Interact? *Science*, 320(5882):1429, 2008.
- [4] G. Klebe. *Wirkstoffdesign: Entwurf und Wirkung von Arzneistoffen*. Spektrum Akademischer Verlag, 2nd edition, 2009. In: Springer-Online.
- [5] N. Tuncbag, A. GURSOY, E. GUNEY, R. NUSSINOV, and O. KESKIN. Architectures and functional coverage of protein–protein interfaces. *Journal of Molecular Biology*, 381(3):785–802, 2008.
- [6] C.A. Lipinski, F. Lombardo, B.W. Dominy, and P.J. Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced drug delivery reviews*, 23(1-3):3–25, 1997.
- [7] R.J. Ellis and A.P. Minton. Protein aggregation in crowded environments. *Biological Chemistry*, 387(5):485, 2006.
- [8] J. Janin and F. Rodier. Protein-protein interaction at crystal contacts. *Proteins: Structure, Function, and Genetics*, 23(4), 1995.
- [9] R. Prasad Bahadur, P. Chakrabarti, F. Rodier, and J. Janin. A dissection of specific and non-specific protein–protein interfaces. *Journal of molecular biology*, 336(4):943–955, 2004.
- [10] A. Pal, P. Chakrabarti, R. Bahadur, F. Rodier, and J. Janin. Peptide segments in protein-protein interfaces. *Journal of Biosciences*, 32(1):101–111, 2007.
- [11] W.S.J. Valdar and J.M. Thornton. Conservation helps to identify biologically relevant crystal contacts. *Journal of Molecular Biology*, 313(2):399–416, 2001.
- [12] G. Waksman. *Proteomics and protein-protein interactions: biology, chemistry, bioinformatics, and drug design*. Springer, 2005.
- [13] I.M.A. Nooren and J.M. Thornton. Diversity of protein–protein interactions. *The EMBO Journal*, 22:3486–3492, 2003.
- [14] S. Jones and J.M. Thornton. Principles of protein-protein interactions. *Proceedings of the National Academy of Sciences*, 93(1):13–20, 1996.
- [15] P. Aloy, H. Ceulemans, A. Stark, and R.B. Russell. The relationship between sequence and interaction divergence in proteins. *Journal of molecular biology*, 332(5):989–998, 2003.

- [16] O. Keskin, C.J. Tsai, H. Wolfson, and R. Nussinov. A new, structurally nonredundant, diverse data set of protein–protein interfaces and its implications. *Protein Science: A Publication of the Protein Society*, 13(4):1043, 2004.
- [17] Y. Ofra and B. Rost. Analysing Six Types of Protein–Protein Interfaces. *J. Mol. Biol.*, 325(2):377–387, 2003.
- [18] M.H. Ali and B. Imperiali. Protein oligomerization: how and why. *Bioorganic & medicinal chemistry*, 13(17):5013–5020, 2005.
- [19] D.S. Goodsell. Inside a living cell. *Trends in biochemical sciences*, 16(6):203–206, 1991.
- [20] S. De, O. Krishnadev, N. Srinivasan, and N. Rekha. Interaction preferences across protein-protein interfaces of obligatory and non-obligatory components are different. *BMC Structural Biology*, 5(1):15, 2005.
- [21] S. Ansari and V. Helms. Statistical analysis of predominantly transient protein-protein interfaces. *Proteins: Structure, Function, and Bioinformatics*, 61(2):344–355, 2005.
- [22] A.A. Bogan and K.S. Thorn. Anatomy of hot spots in protein interfaces. *Journal of Molecular Biology*, 280(1):1–9, 1998.
- [23] T. Clackson and J.A. Wells. A hot spot of binding energy in a hormone-receptor interface. *SCIENCE-NEW YORK THEN WASHINGTON-*, pages 383–383, 1995.
- [24] T. Kortemme and D. Baker. A simple physical model for binding energy hot spots in protein–protein complexes. *Proceedings of the National Academy of Sciences*, 99(22):14116, 2002.
- [25] Y. Gao, R. Wang, and L. Lai. Structure-based method for analyzing protein–protein interfaces. *Journal of Molecular Modeling*, 10(1):44–54, 2004.
- [26] R. Guerois, J.E. Nielsen, and L. Serrano. Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *Journal of molecular biology*, 320(2):369–387, 2002.
- [27] D. Gonzalez-Ruiz and H. Gohlke. Targeting protein-protein interactions with small molecules: challenges and perspectives for omputational binding epitope detection and ligand finding. *Current medicinal chemistry*, 13(22):2607–2625, 2006.
- [28] N. Tuncbag, A. Gursoy, and O. Keskin. Identification of computational hot spots in protein interfaces: combining solvent accessibility and inter-residue potentials improves the accuracy. *Bioinformatics*, 25(12):1513, 2009.
- [29] M. Cohen, D. Reichmann, H. Neuvirth, and G. Schreiber. Similar chemistry, but different bond preferences in inter versus intra-protein interactions. *Proteins: Structure, Function, and Bioinformatics*, 72(2):741–753, 2008.

- [30] S. Kumar, B. Ma, C.J. Tsai, N. Sinha, and R. Nussinov. Folding and binding cascades: dynamic landscapes and population shifts. *Protein Science*, 9(1):10–19, 2000.
- [31] J. Janin. Wet and dry interfaces: the role of solvent in protein-protein and protein-DNA recognition. *Structure*, 7(12):R277–R279, 1999.
- [32] J. Teyra and M.T. Pisabarro. Characterization of interfacial solvent in protein complexes and contribution of wet spots to the interface description. *Proteins: Structure, Function, and Bioinformatics*, 67(4):1087–1095, 2007.
- [33] M. Ahmad, W. Gu, and V. Helms. Mechanism of fast peptide recognition by SH3 domains. *Angewandte Chemie International Edition*, 47(40):7626–7630, 2008.
- [34] M. Ahmad, W. Gu, T. Geyer, and V. Helms. Adhesive water networks facilitate binding of protein interfaces. *Nature Communications*, 2:261, 2011.
- [35] K. Gunasekaran, C.J. Tsai, and R. Nussinov. Analysis of ordered and disordered protein complexes reveals structural features discriminating between stable and unstable monomers. *Journal of molecular biology*, 341(5):1327–1341, 2004.
- [36] O. Keskin, A. Gursoy, B. Ma, and R. Nussinov. Principles of protein-protein interactions: what are the preferred ways for proteins to interact? *Chemical reviews*, 108(4):1225–1244, 2008.
- [37] G. Schreiber. Kinetic studies of protein-protein interactions. *Current opinion in structural biology*, 12(1):41–47, 2002.
- [38] G. Schreiber, G. Haran, and H.X. Zhou. Fundamental Aspects of Protein-Protein Association Kinetics. *Chem. Rev*, 109(3):839–860, 2009.
- [39] M. Shatsky, R. Nussinov, and H.J. Wolfson. MultiProt-a multiple protein structural alignment algorithm. *Lecture notes in computer science*, 2452:235–250, 2002.
- [40] H. Zhu, I. Sommer, T. Lengauer, and F.S. Domingues. Alignment of non-covalent interactions at protein-protein interfaces. *PLoS ONE*, 3(4), 2008.
- [41] W.K. Kim, A. Henschel, C. Winter, and M. Schroeder. The many faces of protein-protein interactions: A compendium of interface geometry. *PLoS Comput Biol*, 2(9):e124, 2006.
- [42] J. Mintseris and Z. Weng. Atomic contact vectors in protein-protein recognition. *Proteins Structure Function and Genetics*, 53(3):629–639, 2003.
- [43] C. Zhang, G. Vasmatzis, J.L. Cornette, and C. DeLisi. Determination of atomic desolvation energies from the structures of crystallized proteins. *Journal of molecular biology*, 267(3):707–726, 1997.

- [44] BA Shoemaker and AR Panchenko. Deciphering protein-protein interactions. Part I. *Experimental techniques and databases. PLoS Comput. Biol.*, 3:e42, 2007.
- [45] A.J.M. Walhout, S.J. Boulton, and M. Vidal. Yeast two-hybrid systems and protein interaction mapping projects for yeast and worm. *Yeast*, 17(2):88–94, 2000.
- [46] O. Puig, F. Caspary, G. Rigaut, B. Rutz, E. Bouveret, E. Bragado-Nilsson, M. Wilm, and B. Séraphin. The tandem affinity purification (TAP) method: a general procedure of protein complex purification. *Methods*, 24(3):218–229, 2001.
- [47] A.C. Gavin, M. Bösch, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J.M. Rick, A.M. Michon, C.M. Cruciat, et al. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147, 2002.
- [48] B. Nölting. *Methods in modern Biophysics*. Springer, 2nd edition edition, 2006.
- [49] I. Jelesarov and H.R. Bosshard. Isothermal titration calorimetry and differential scanning calorimetry as complementary tools to investigate the energetics of biomolecular recognition. *Journal of molecular recognition*, 12(1):3–18, 1999.
- [50] V. Helms. *Principles of Computational Cell Biology*. Wiley VCH, 2003.
- [51] A. Valencia and F. Pazos. Computational methods for the prediction of protein interactions. *Current Opinion in Structural Biology*, 12(3):368–373, 2002.
- [52] B.A. Shoemaker and A.R. Panchenko. Deciphering protein-protein interactions. Part II. Computational methods to predict protein and domain interaction partners. *PLoS Comput Biol*, 3(4):e43, 2007.
- [53] A.J. Enright, I. Iliopoulos, N.C. Kyrioides, and C.A. Ouzounis. Protein interaction maps for complete genomes based on gene fusion events. *Nature*, 402(6757):86–90, 1999.
- [54] O. Noivirt, M. Eisenstein, and A. Horovitz. Detection and reduction of evolutionary noise in correlated mutation analysis. *Protein Engineering Design and Selection*, 18(5):247, 2005.
- [55] D.R. Caffrey, S. Somaroo, J.D. Hughes, J. Mintseris, and E.S. Huang. Are protein-protein interfaces more conserved in sequence than the rest of the protein surface? *Prot. Sci.*, 13(1):190, 2004.
- [56] Y.S. Choi, J.S. Yang, Y. Choi, S.H. Ryu, and S. Kim. Evolutionary conservation in multiple faces of protein interaction. *Proteins: Structure, Function, and Bioinformatics*, 77(1), 2009.
- [57] F. Glaser, D.M. Steinberg, I.A. Vakser, and N. Ben-Tal. Residue frequencies and pairing preferences at protein-protein interfaces. *Proteins Structure Function and Genetics*, 43(2):89–102, 2001.

- [58] Y. Ofra and B. Rost. Predicted protein–protein interaction sites from local sequence information. *FEBS letters*, 544(1-3):236–239, 2003.
- [59] H.X. Zhou and Y. Shan. Prediction of protein interaction sites from sequence profile and residue neighbor list. *Proteins Str. Fn. Gen.*, 44(3):336–343, 2001.
- [60] R. Chen, L. Li, and Z. Weng. ZDOCK: an initial-stage protein-docking algorithm. *Proteins Structure Function and Genetics*, 52(1):80–87, 2003.
- [61] J. Mintseris, K. Wiehe, B. Pierce, R. Anderson, R. Chen, J. Janin, and Z. Weng. Protein-protein docking benchmark 2.0: an update. *PROTEINS-NEW YORK-*, 60(2):214, 2005.
- [62] M. Topf and A. Sali. Combining electron microscopy and comparative protein structure modeling. *Current opinion in structural biology*, 15(5):578–585, 2005.
- [63] P. Aloy and R.B. Russell. InterPreTS: protein interaction prediction through tertiary structure. *Bioinformatics*, 19(1):161, 2003.
- [64] P.J. Kundrotas and E. Alexov. Predicting 3D structures of transient protein–protein complexes by homology. *BBA-Proteins and Proteomics*, 1764(9):1498–1511, 2006.
- [65] L. Lu, H. Lu, and J. Skolnick. MULTIPROSPECTOR: an algorithm for the prediction of protein-protein interactions by multimeric threading. *PROTEINS: Structure, Function, and Genetics*, 49(3), 2002.
- [66] S. Jaeger and U. Leser. High-precision function prediction using conserved interactions. In *German Conference on Bioinformatics (GCB)*, pages 145–162, 2007.
- [67] B. Lehner and A. Fraser. A first-draft human protein-interaction map. *Genome Biology*, 5(9):R63, 2004.
- [68] J.M. Thornton, A.E. Todd, D. Milburn, N. Borkakoti, and C.A. Orengo. From structure to function: approaches and limitations. *Nat. Struct. Biol.*, 7:991–994, 2000.
- [69] C.P. Adams and V.V. Brantner. Estimating the cost of new drug development: Is it really \$802 million? *Health Affairs*, 25(2):420, 2006.
- [70] G.M. Lee and C.S. Craik. Trapping moving targets with small molecules. *Science*, 324(5924):213, 2009.
- [71] Folkers G., Hüser J., and R. Mannhold. *High-Throughput Screening in Drug Discovery (Methods and Principles in Medicinal Chemistry)*. Wiley VCH, 2006.
- [72] H.J. Böhm and Schneider G. *Protein ligand interactions: From molecular recognition to drug design*. Wiley VCH, 2003.
- [73] R.A. Laskowski. SURFNET: A program for visualizing molecular surfaces, cavities, and intermolecular interactions. *J. Mol. Graph.*, 13(5):323–330, 1995.

- [74] M. Hendlich, F. Rippmann, and G. Barnickel. Ligsite: automatic and efficient detection of potential small molecule-binding sites in proteins. *Journal of Molecular Graphics and Modelling*, 15(6):359–363, 1997.
- [75] G.P. Brady and P.F.W. Stouten. Fast prediction and visualization of protein binding pockets with pass. *Journal of Computer-Aided Molecular Design*, 14(4):383–401, 2000.
- [76] A.T.R. Laurie and R.M. Jackson. Q-SiteFinder: an energy-based method for the prediction of protein-ligand binding sites. *Bioinformatics*, 21(9):1908, 2005.
- [77] S. Eyrisch and V. Helms. Transient Pockets on Protein Surfaces Involved in Protein- Protein Interaction. *J. Med. Chem*, 50(15):3457–3464, 2007.
- [78] A. Kukol. *Molecular Modeling of Proteins*. Humana press, 2008.
- [79] C.G. Wermuth. *The practice of medicinal chemistry*. Academic Press, 2008.
- [80] P.T. Corbett, J. Leclaire, L. Vial, K.R. West, J.L. Wietor, J.K.M. Sanders, and S. Otto. Dynamic combinatorial chemistry. *Chemical reviews*, 106(9):3652–3711, 2006.
- [81] Y. Landry and J.P. Gies. Drugs and their molecular targets: an updated overview. *Fundamental & clinical pharmacology*, 22(1):1–18, 2008.
- [82] J.P. Overington, B. Al-Lazikani, and A.L. Hopkins. How many drug targets are there? *Nature reviews Drug discovery*, 5(12):993–996, 2006.
- [83] N. Tuncbag, G. Kar, O. Keskin, A. GURSOY, and R. Nussinov. A survey of available tools and web servers for analysis of protein-protein interactions and interfaces. *Briefings in Bioinformatics*, 10(3):217, 2009.
- [84] C. Winter, A. Henschel, W.K. Kim, and M. Schroeder. SCOPPI: A structural classification of protein-protein interfaces. *Nucleic acids research*, 34(suppl 1):D310, 2006.
- [85] F.P. Davis and A. Sali. PIBASE: a comprehensive database of structurally defined protein interfaces. *Bioinformatics*, 21(9):1901–1907, 2005.
- [86] A. Stein, A. Panjkovich, and P. Aloy. 3did Update: domain-domain and peptide-mediated interactions of known 3D structure. *Nucleic Acids Research*, 37(Database issue):D300, 2009.
- [87] C. Alfarano, CE Andrade, K. Anthony, N. Bahroos, M. Bajec, K. Bantoft, D. Betel, B. Bobechko, K. Boutilier, E. Burgess, et al. The biomolecular interaction network database and related tools 2005 update. *Nucleic acids research*, 33(Database Issue):D418, 2005.
- [88] L.J. Jensen, M. Kuhn, M. Stark, S. Chaffron, C. Creevey, J. Muller, T. Doerks, P. Julien, A. Roth, M. Simonovic, et al. STRING 8—a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Research*, 37(Database issue):D412, 2009.

- [89] L. Salwinski, C.S. Miller, A.J. Smith, F.K. Pettit, J.U. Bowie, and D. Eisenberg. The database of interacting proteins: 2004 update. *Nucleic Acids Research*, 32(Database Issue):D449, 2004.
- [90] A. Zanzoni, L. Montecchi-Palazzi, M. Quondam, G. Ausiello, M. Helmer-Citterich, and G. Cesareni. MINT: a Molecular INTeraction database. *FEBS letters*, 513(1):135–140, 2002.
- [91] C. Stark, B.J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. BioGRID: a general repository for interaction datasets. *Nucleic acids research*, 34(Database Issue):D535, 2006.
- [92] G.R. Mishra, M. Suresh, K. Kumaran, N. Kannabiran, S. Suresh, P. Bala, K. Shivakumar, N. Anuradha, R. Reddy, T.M. Raghavan, et al. Human protein reference database–2006 update. *Nucleic acids research*, 34(Database Issue):D411, 2006.
- [93] E. Krissinel and K. Henrick. Inference of macromolecular assemblies from crystalline state. *Journal of molecular biology*, 372(3):774–797, 2007.
- [94] R. Wang, X. Fang, Y. Lu, and S. Wang. The PDBbind database: collection of binding affinities for protein-ligand complexes with known three-dimensional structures. *Journal of medicinal chemistry*, 47(12):2977–2980, 2004.
- [95] P. Block, C.A. Sotriffer, I. Dramburg, and G. Klebe. AffinDB: a freely accessible database of affinities for protein–ligand complexes from the PDB. *Nucleic acids research*, 34(suppl 1):D522, 2006.
- [96] M.L. Benson, R.D. Smith, N.A. Khazanov, B. Dimcheff, J. Beaver, P. Dresslar, J. Nerothin, and H.A. Carlson. Binding MOAD, a high-quality protein–ligand database. *Nucleic acids research*, 36(suppl 1):D674, 2008.
- [97] C.J. Date. *An introduction to database systems*. Springer, 2000.
- [98] CJ Date. *Database in depth: relational theory for practitioners*. O’Reilly Media, Inc., 2005.
- [99] A. Heuer and G. Saake. *Datenbanken, Konzepte und Sprachen*. mitp-Verlag, 2000.
- [100] A. Kemper and A. Eickler. *Datenbanksysteme: eine Einführung*. Oldenbourg Wissenschaftsverlag, 2006.
- [101] Sasha Pachev. *Understanding MySQL internals*. O’Reilly, 2007.
- [102] M.A. Lomize, A.L. Lomize, I.D. Pogozheva, and H.I. Mosberg. OPM: orientations of proteins in membranes database. *Bioinformatics*, 22(5):623, 2006.
- [103] H. Zhu, F.S. Domingues, I. Sommer, and T. Lengauer. NOXclass: prediction of protein-protein interaction types. *BMC bioinformatics*, 7(1):27, 2006.

- [104] SJ Hubbard and JM Thornton. NACCESS computer program. *Department of Biochemistry and Molecular Biology, University College London*, 2(9), 1993.
- [105] J. Kyte and R.F. Doolittle. A simple method for displaying the hydrophobic character of a protein. *J. Mol. Biol.*, 157(1):105–132, 1982.
- [106] C. Steinbeck, Y. Han, S. Kuhn, O. Horlacher, E. Luttmann, and E. Willighagen. The chemistry development kit (cdk): An open-source java library for chemo- and bioinformatics. *Journal of chemical information and computer sciences*, 43(2):493–500, 2003.
- [107] Daylight Inc. Fingerprints - screening and similarity. website, 2008. available online at [www.daylight.com/dayhtml/doc/theory/theory.finger.html](http://www.daylight.com/dayhtml/doc/theory/theory.finger.html).
- [108] TT Tanimoto. IBM Internal Report 17th Nov, 1957.
- [109] S. Kawashima, P. Pokarowski, M. Pokarowska, A. Kolinski, T. Katayama, and M. Kanehisa. AAindex: amino acid index database, progress report 2008. *Nucleic Acids Research*, 36(Database issue):D202, 2008.
- [110] E. Freeman, B. Bates, and K. Sierra. *Head first design patterns*. O’Reilly & Associates, Inc., 2004.
- [111] R.C.G. Holland, T.A. Down, M. Pocock, A. Prlić, D. Huen, K. James, S. Foisy, A. Dräger, A. Yates, M. Heuer, et al. Biojava: an open-source framework for bioinformatics. *Bioinformatics*, 24(18):2096, 2008.
- [112] J. Westbrook, N. Ito, H. Nakamura, K. Henrick, and H.M. Berman. Pdbml: the representation of archival macromolecular structure data in xml. *Bioinformatics*, 21(7):988, 2005.
- [113] M. Hall and L. Brown. *Core Servlets und JavaServer Pages*. Pearson Education, 2004.
- [114] Apache foundation. Apache tomcat website. website, 2011. available online at [tomcat.apache.org/](http://tomcat.apache.org/).
- [115] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design patterns: elements of reusable object-oriented software*, volume 206. Addison-wesley Reading, MA, 1995.
- [116] LJ Jensen, R. Gupta, H.H. Staerfeldt, and S. Brunak. Prediction of human protein function according to Gene Ontology categories. *Bioinformatics*, 19(5):635–642, 2003.
- [117] A. Shulman-Peleg, R. Nussinov, and H.J. Wolfson. Recognition of Functional Sites in Protein Structures. *J. Mol. Biol.*, 339(3):607–633, 2004.
- [118] A. Walker-Taylor and DT Jones. Computational Methods for Predicting Protein–Protein Interactions. In G. Waksman, editor, *Proteomics and Protein-protein interactions: Biology, Chemistry, Bioinformatics, and Drug Design*, pages 89–112. Springer, 2005.

- [119] J. Mintseris and Z. Weng. Structure, function, and evolution of transient and obligate protein-protein interactions. *Proc. Natl. Acad. Sci. USA*, 102(31):10930–10935, 2005.
- [120] I.M.A. Nooren and J.M. Thornton. Structural Characterisation and Functional Significance of Transient Protein-Protein Interactions. *J. Mol. Biol.*, 325(5):991–1018, 2003.
- [121] N. Nagano, C.A. Orengo, and J.M. Thornton. One Fold with Many Functions: The Evolutionary Relationships between TIM Barrel Families Based on their Sequences, Structures and Functions. *J. Mol. Biol.*, 321(5):741–765, 2002.
- [122] R.M. Jackson. Comparison of protein-protein interactions in serine protease-inhibitor and antibody-antigen complexes: Implications for the protein docking problem. *PRS*, 8(03):603–613, 1999.
- [123] J. Wang. Protein recognition by cell surface receptors: physiological receptors versus virus interactions. *TIBS*, 27(3):122–126, 2002.
- [124] P. Walter, Ansari S., and V. Helms. The ABC (Analysing Biomolecular Contacts)-database. *Journal of Integrative Bioinformatics*, 4(1):50, 2007.
- [125] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, TN Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The Protein Data Bank. *Nucl. Ac. Res.*, 28(1):235–242, 2000.
- [126] S.S. Negi and W. Braun. Statistical analysis of physical-chemical properties and prediction of protein-protein interfaces. *J. Mol. Mod.*, 13(11):1157–1167, 2007.
- [127] U. Hobohm and C. Sander. Enlarged representative set of protein structures. *Prot. Sci.*, 3(3):522, 1994.
- [128] M. Ashburner, CA Ball, JA Blake, D. Botstein, H. Butler, JM Cherry, AP Davis, K. Dolinski, SS Dwight, JT Eppig, et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, 25(1):25–9, 2000.
- [129] E. Camon, D. Barrell, C. Brooksbank, M. Magrane, and R. Apweiler. The Gene Ontology Annotation(GOA) project— application of GO in SWISS-PROT, TrEMBL and InterPro. *Comparative and Functional Genomics*, 4(1):71–74, 2003.
- [130] PW Lord, RD Stevens, A. Brass, and CA Goble. Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation. *Bioinformatics*, 19(10):1275–1283, 2003.
- [131] D. Lin. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, pages 296–304, 1998.
- [132] P. Resnik. Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. *Journal of Artificial Intelligence*, 11(11):95–130, 1999.

- [133] A. Schlicker, F. Domingues, J. Rahnenführer, and T. Lengauer. A new measure for functional similarity of gene products based on Gene Ontology. *BMC Bioinformatics*, 7(1):302, 2006.
- [134] R-project. website. available online at [www.r-project.org](http://www.r-project.org).
- [135] N. Lin, B. Wu, R. Jansen, M. Gerstein, and H. Zhao. Information assessment on predicting protein-protein interactions. *BMC Bioinformatics*, 5(154), 2005.
- [136] M. Wilczynska, M. Fa, J. Karolin, P.I. Ohlsson, L.B.A. Johansson, and T. Ny. Structural insights into serpin—protease complexes reveal the inhibitory mechanism of serpins. *Nat. Struct. Biol.*, 4(5):354–357, 1997.
- [137] AJ Scheidig, TR Hynes, LA Pelletier, JA Wells, and AA Kossiakkoff. Crystal structures of bovine chymotrypsin and trypsin complexed to the inhibitor domain of Alzheimer’s amyloid beta-protein precursor (APPI) and basic pancreatic trypsin inhibitor (BPTI): Engineering of inhibitors with altered specificities. *Prot. Sci.*, 6(9):1806–1824, 1997.
- [138] A.M. Buckle, G. Schreiber, and A.R. Fersht. Protein-protein recognition: Crystal structural analysis of a barnase-barstar complex at 2.0-Å resolution. *Biochemistry*, 33(30):8878–8889, 1994.
- [139] B. Smith, J. Williams, and S.K. Steffen. The Ontology of the Gene Ontology. In *AMIA... Annual Symposium proceedings [electronic resource]*, volume 2003, pages 609–614. American Medical Informatics Association, 2003.
- [140] A. Ruepp, A. Zollner, D. Maier, K. Albermann, J. Hani, M. Mokejcs, I. Tetko, U. Guldener, G. Mannhaupt, M. Munsterkotter, et al. The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. *Nucl. Ac. Res.*, 32(18):5539–5545, 2004.
- [141] T. Yamada and P. Bork. Evolution of biomolecular networks—lessons from metabolic and protein interactions. *Nature Reviews Molecular Cell Biology*, 10(11):791–803, 2009.
- [142] J.A. Wells and C.L. McClendon. Reaching for high-hanging fruit in drug discovery at protein–protein interfaces. *Nature*, 450(7172):1001–1009, 2007.
- [143] I.S. Moreira, P.A. Fernandes, and M.J. Ramos. Hot spots—A review of the protein–protein interface determinant amino-acid residues. *Proteins: Structure, Function, and Bioinformatics*, 68(4):803–812, 2007.
- [144] K. Cho, D. Kim, and D. Lee. A feature-based approach to modeling protein–protein interaction hot spots. *Nucleic acids research*, 37(8):2672, 2009.
- [145] D.C. Fry and L.T. Vassilev. Targeting protein–protein interactions for cancer therapy. *Journal of molecular medicine*, 83(12):955–963, 2005.

- [146] R.A. Laskowski, N.M. Luscombe, M.B. Swindells, and J.M. Thornton. Protein clefts in molecular recognition and function. *Protein Science: A Publication of the Protein Society*, 5(12):2438, 1996.
- [147] S. Zhong, A.T. Macias, and A.D. MacKerell. Computational identification of inhibitors of protein-protein interactions. *Current Topics in Medicinal Chemistry*, 7(1):63–82, 2007.
- [148] N. Sugaya and K. Ikeda. Assessing the druggability of protein-protein interactions by a supervised machine-learning method. *BMC bioinformatics*, 10(1):263, 2009.
- [149] P. Schmidtke and X. Barril. Understanding and predicting druggability. A high-throughput method for detection of drug binding sites. *Journal of medicinal chemistry*, 2010.
- [150] A.P. Higueruelo, A. Schreyer, G.R.J. Bickerton, W.R. Pitt, C.R. Groom, and T.L. Blundell. Atomic interactions and profile of small molecules disrupting protein-protein interfaces: the TIMBAL database. *Chemical Biology & Drug Design*, 74(5):457–467, 2009.
- [151] Fred P. Davis and Andrej Sali. The overlap of small molecule and protein binding sites within families of protein structures. *PLoS Comput Biol*, 6(2):e1000668, 2010. 20140189.
- [152] J. Gruber, A. Zawaira, R. Saunders, C.P. Barrett, and M.E.M. Noble. Computational analyses of the surface properties of protein-protein interfaces. *Acta Crystallographica Section D: Biological Crystallography*, 63(1):50–57, 2006.
- [153] B. Ma and R. Nussinov. Trp/Met/Phe hot spots in protein-protein interactions: potential targets in drug design. *Current topics in medicinal chemistry*, 7(10):999–1005, 2007.
- [154] SE Ozbabacan, A. Gursoy, O. Keskin, and R. Nussinov. Conformational ensembles, signal transduction and residue hot spots: Application to drug discovery. *Current opinion in drug discovery & development*, 13(5):527–537, 2010.
- [155] N. Tuncbag, O. Keskin, and A. Gursoy. Hotpoint: hot spot prediction server for protein interfaces. *Nucleic acids research*, 38(suppl 2):W402, 2010.
- [156] C.J.R. Illingworth, P.D. Scott, K.E.B. Parkes, C.R. Snell, M.P. Campbell, and C.A. Reynolds. Connectivity and binding-site recognition: Applications relevant to drug design. *Journal of Computational Chemistry*, 2010.
- [157] A. Liaw and M. Wiener. Classification and Regression by randomForest. *R news*, 2(3):18–22, 2002.
- [158] F. P. Davis. Proteome-wide prediction of overlapping small molecule and protein binding sites using structure. *Molecular Biosystems*, 7(2):545–557, 2011.

- [159] Y. Huang, B. Niu, Y. Gao, L. Fu, and W. Li. CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics*, 26(5):680, 2010.
- [160] P. Gund. Three-dimensional pharmacophoric pattern searching. *Prog. Mol. Subcell. Biol.*, 5:117–143, 1977.
- [161] Chemaxon. JChem package. website, 2009. available online at [www.chemaxon.com](http://www.chemaxon.com).
- [162] H. Ashkenazy, E. Erez, E. Martz, T. Pupko, and N. Ben-Tal. Consurf 2010: calculating evolutionary conservation in sequence and structure of proteins and nucleic acids. *Nucleic Acids Research*, 38(suppl 2):W529, 2010.
- [163] A. Pintar, O. Carugo, and S. Pongor. CX, an algorithm that identifies protruding atoms in proteins. *Bioinformatics*, 18(7):980, 2002.
- [164] S. Jones and J.M. Thornton. Prediction of protein-protein interaction sites using patch analysis. *Journal of molecular biology*, 272(1):133–143, 1997.
- [165] S. Jones and J.M. Thornton. Analysis of protein-protein interaction sites using surface patches. *Journal of molecular biology*, 272(1):121–132, 1997.
- [166] M.E. Cusick, N. Klitgord, M. Vidal, and D.E. Hill. Interactome: gateway into systems biology. *Human molecular genetics*, 14(suppl 2):R171, 2005.
- [167] G.T. Hart, A.K. Ramani, and E.M. Marcotte. How complete are current yeast and human protein-interaction networks. *Genome Biol*, 7(11):120, 2006.
- [168] M. Behar, H.G. Dohlman, and T.C. Elston. Kinetic insulation as an effective mechanism for achieving pathway specificity in intracellular signaling networks. *Proceedings of the National Academy of Sciences*, 104(41):16146, 2007.
- [169] F.B. Sheinerman, R. Norel, and B. Honig. Electrostatic aspects of protein-protein interactions. *Current Opinion in Structural Biology*, 10(2):153–159, 2000.
- [170] A. Spaar, C. Dammer, R.R. Gabdoulline, R.C. Wade, and V. Helms. Diffusional encounter of barnase and barstar. *Biophysical journal*, 90(6):1913–1924, 2006.
- [171] R.A. Coleman and B.F. Pugh. Slow dimer dissociation of the tata binding protein dictates the kinetics of dna binding. *Proceedings of the National Academy of Sciences*, 94(14):7221, 1997.
- [172] S. Lalonde, D.W. Ehrhardt, D. Loqué, J. Chen, S.Y. Rhee, and W.B. Frommer. Molecular and cellular approaches for the detection of protein-protein interactions: latest techniques and current limitations. *The Plant Journal*, 53(4):610–635, 2008.

- [173] M. Smoluchowski et al. Versuch einer mathematischen Theorie der Koagulationskinetik kolloider Lösungen. *Z. phys. Chem*, 92(2):129–168, 1917.
- [174] Y. Shaul and G. Schreiber. Exploring the charge space of protein–protein association: a proteomic study. *Proteins: Structure, Function, and Bioinformatics*, 60(3):341–352, 2005.
- [175] R. Alsallaq and H.X. Zhou. Prediction of protein-protein association rates from a transition-state theory. *Structure*, 15(2):215–224, 2007.
- [176] R.R. Gabdoulline, M. Stein, and R.C. Wade. qPIPSA: Relating enzymatic kinetic parameters and interaction fields. *BMC bioinformatics*, 8(1):373, 2007.
- [177] C.E. Felder, J. Prilusky, I. Silman, and J.L. Sussman. A server and database for dipole moments of proteins. *Nucleic acids research*, 35(suppl 2):W512, 2007.
- [178] V.N. Vapnik. *The nature of statistical learning theory*. Springer Verlag, 2000.
- [179] V.N. Vapnik. *Statistical learning theory*. Wiley-Interscience, 1998.
- [180] C.W. Hsu and C.J. Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.
- [181] R. Team. R: A language and environment for statistical computing. *R Foundation for Statistical Computing Vienna Austria ISBN*, 3(10), 2008.
- [182] E. Dimitriadou, K. Hornik, F. Leisch, D. Meyer, and A. Weingessel. Misc Functions of the Department of Statistics (e1071), TU Wien. *R package version*, 1:5–7, 2005.
- [183] C.C. Chang and C.J. Lin. LIBSVM: a library for support vector machines, 2001.
- [184] S.A. Allison and J.A. McCammon. Dynamics of substrate binding to copper zinc superoxide dismutase. *The Journal of Physical Chemistry*, 89(7):1072–1074, 1985.
- [185] H.X. Zhou. Brownian dynamics study of the influences of electrostatic interaction and diffusion on protein-protein association kinetics. *Biophysical journal*, 64(6):1711–1726, 1993.
- [186] Q. Liu and J. Li. Protein binding hot spots and the residue-residue pairing preference: a water exclusion perspective. *BMC bioinformatics*, 11(1):244, 2010.
- [187] N. Tuncbag, F.S. Salman, O. Keskin, and A. Gursoy. Analysis and network representation of hotspots in protein interfaces using minimum cut trees. *Proteins*, 78(10):2283, 2010.