Universität Bielefeld
Technische Fakultät

Dissertation
zur Erlangung des akademischen Grades
Doktor Ing.

# Soft Volume Models for Protein-Protein Docking

Steffen Neumann

Bielefeld, den 01. Dezember 2003

Betreut von:
Prof. Dr.-Ing Gerhard Sagerer,
Prof. Dr.-Ing. Franz Kummert
Arbeitsgruppe Angewandte Informatik
Technische Fakultät
Universität Bielefeld

## Erklärung

Hiermit erkläre ich, daß die vorliegende Arbeit von mir selbständig und nur unter Verwendung der erlaubten und aufgeführten Hilfsmittel erstellt wurde.

Ich erkenne ferner die momentan gültige Prüfungsordnung der Technischen Fakultät der Universität Bielefeld an.

Steffen Neumann Bielefeld, den 01. Dezember 2003

Gedruckt auf alterungsbeständigem Papier ISO 9706.

## Abstract

Protein docking is the question whether and how two proteins interact, starting from their 3D structure. For training and test of docking systems large data sets are needed. A method for automated Test Case generation based on combined searches and filters on the content of the Protein Data Bank (PDB) is described.

ElMaR is a distributed, modular and optionally parallel docking system. Fast Docking algorithms usually employ the rigid-body assumption and score geometric complementarity as well as physicochemical features. However, for unbound protein docking steric clashes might impose wrong penalisation if side chains change their conformation during the docking process. ElMaR incorporates protein flexibility obtained through statistics and force field calculation. Using a fast correlation technique, steric clash penalties are weighted according to the possibility of amino acid rotamer changes.

Results on the generated test sets are presented and discussed. The ability to distinguish between native and non-native contact sites is tested on interfaces in protein crystals. A performance exceeding published results has been achieved.



## Zusammenfassung

Der Begriff ¨Protein Docking¨ beschreibt die Frage, ob und wie zwei gegebene Proteine interagieren, ausgehend von der 3D Struktur. Für die Entwicklung von Protein Docking Systemen werden große Testsätze für Training und Validierung benötigt. Die manuelle Erstellung solcher Datensätze kann nicht mit dem exponentiellen Wachstum der Protein Datenbank (PDB) Schritt halten. Eine automatische Methode für die Erstellung von Testdatensätzen auf Basis kombinierter Suchverfahren und Filter wird vorgestellt.

Das Docking System ElMaR besteht aus verteilten, optional parallelisierten Modulen. Durch die Parallelisierung werden Ergebnisse in wenigen Minuten berechnet. Docking Hypothesen werden aufgrund der geometrischen Komplementarität, elektrostatischer Kräfte und der Oberflächenhydrophobizität bewertet. ElMaR berücksichtigt die Flexibilität von Seitenketten und führt dynamische Strafterme für die Bewertung von sterischer Überlappung ein. Die Flexibilitätsmaße werden einerseits aus Statistiken über der gesamten Protein Datenbank, andererseits für jedes Protein einzeln aus Kraftfeldern errechnet.

Für die erstellten Testdaten werden Docking Untersuchungen angestellt, und die Ergebnisse anschliessend diskutiert. Die Fähigkeit der Bewertungsfunktion, native von nicht-nativen Proteinkontakten zu unterscheiden wird anhand von Kristallkontakten erfolgreich untersucht.

# Contents

# Introduction 1

Most historical periods were characterised by major improvements in one of the natural sciences. The earlier centuries can be considered as the centuries of maths and physics, with understanding of geometry, calculus, classical mechanics and electricity. Understanding of chemical principles boosted in the 19th and 20th century, with invention of organic and inorganic compound analysis and -synthesis. These developments allowed biologists to look at the inner, molecular, mechanisms in living organisms and cells.

Major break-throughs in molecular biology were Mendel's observations on biological inheritance. The crystallisation of larger proteins like hemoglobin by Hoppe-Seyler in 1864 laid the foundation for the accurate X-RAY-crystallographic analysis of myoglobin by Perutz and Kendrew [Kendrew56] in the 1950's. The determination of the DNA double helix by Crick and Watson [Watson53] in 1953 and the first sequencing of a whole bacteriophage [Sanger77] in 1977 by later Nobel price winners Berg, Gilbert and Sanger marked the beginning of the field of molecular biology. Some of the aspects of molecular biology will be introduced in the next section, followed by a section on computational biology and protein docking in particular. Both disciplines complement each other: experiments provide data for training of computational models, and predictions or simulations suggest which experiments to conduct to verify a hypothesis.

## 1.1. Molecular Biology

Molecular biology is an interdisciplinary field between biology, chemistry and physics. The scale of the subject ranges from DNA bases to large networks of interacting proteins:

**Genomics** deals with aspects of the sequence of DNA and RNA not only in the human genome, including coding and non-coding areas and ribosomal DNA. Serious sequencing of DNA started

with the Maxim-Gilbert method and the Sanger “Primer extension”. First done manually by numerous lab assistants and students, methods for high throughput experiments were developed in the 1990s.

**Proteomics** describes the whole set of proteins in the living organism. Once they are synthesised in the cell, microarray techniques can show whether they are expressed under certain conditions or not. This helps determining the function of the protein. Structure determination gives further details on the chemical mechanisms. Also of interest is the location of proteins within the cell, since a protein might be active in certain cell compartments only.

**Metabolomics** is the “big picture”. The chemical pathways of living cells can fill large posters [Nicholson00]. Also of interest are the kinetics of reactions, needed to quantitatively simulate cells as in e.g. the eCell-Project [Tomira99].

Many more “-omics” have appeared in recent years. The term “Life Sciences” was formed to include all aspects of chemistry and biology which try to describe the workings and underlying principles and diseases in genomics, proteomics and metabolics. Next to scientific research, drug discovery and -development are the driving factors behind progress in this area.

Most of the actual work in the field was and is still done in the “wet” laboratories. Computers were first used to aid statistical analysis of experiments and interpretation of data. X-RAY crystallographers determining protein structures use programs to reconstruct 3D structure from scatter images.

Sequencing the human genome as done by Celera Inc. in the year 2000 was done on several hundred automatic sequencers in parallel running night and day in only nine months [Venter01]. A large compute farm was afterwards used to assemble the sequenced segments into the final genome sequence. Usage of computers is compulsory to manage and interpret those large amounts of data.

## 1.2. Computational Biology

Bioinformatics is the interdisciplinary area where algorithms, data management and -analysis, prediction, simulation and visualisation are developed and combined to interpret, predict or simulate modifications of processes in biological organisms.

The pharmaceutical industry uses computational biology techniques throughout the whole process of drug development. One common task is the search for interacting proteins. Given a target molecule, libraries of potential drugs need to be screened to find possible molecules to alter its (dys-)function. One example are inhibitors slowing down catalytic activity. Among the proteins in

the database several candidates might be available, so if at least one good candidate is found, false negatives are allowed.

Similarly, given a potential drug, possible side effects need to be ruled out. The search needs to find all of the potential interactions. False positives are allowed, those put an increased load on the following investigation stages, where false negatives could potentially harm patients.

Ab-Initio docking algorithms try to predict docking conformations based on the 3D structure of the two components. The primary focus of research in the protein protein docking area is to accurately predict the detailed interaction on a residue- or even atomic level. Care has to be taken to either keep the runtime requirements low, acceptable for database searches, or to achieve very accurate results.

Docking algorithms that consider the proteins having a fixed shape (rigid-body assumption) can be mislead by the so called induced fit. The term refers to molecules changing their conformation during the docking process. The search space in case of the rigid-body assumption is usually 6-dimensional, with a 3D translation and 3 rotational axes. More accurate docking algorithms also allow for movement of domains, side chains or individual atoms, increasing computing demands drastically. Their runtimes can be in the order of hours or even days for one putative complex.

A different approach is homology modelling, which becomes increasingly promising with a growing set of analysed protein structures: Given a complex conformation (experimentally derived or modelled in a previous step) the structure database is searched for similar docking sites. The difference to existing directories of protein domains like CATH [Orengo97] is the focus on surface regions.

At Bielefeld University the docking software BI was developed by Friedrich Ackermann et. al [Ackermann98] in the context of the BIOWEPRO project. It contains a C++ library for representation of proteins and molecules, as well as fast correlation algorithms used in the scoring function. It has been shown to be very efficient in terms of runtime complexity and produces accurate results on bound complex structures that can be used as a preprocessing step, cutting down the search space of potential docking conformations for compute-intensive docking programs working on atomic levels.

**Goal of this work** is to improve a docking system and scoring function to docking of unbound structures, incorporating flexibility information. The necessary infrastructure (test set creation, communication and database integration) is needed to keep track with the growing base of available protein structures.

This work is structured as follows:

The next chapter will give an introduction to the biological background and protein interaction in particular. The principles of protein docking and an overview of current docking systems will

be given in chapter 3. Chapter 4 describes published protein docking test sets, and automated methods to search for protein complexes with unbound components available in protein structure databases. The architecture and building blocks of the protein docking system ELMAR are shown in chapter 5. Several experiments for system evaluation and a discussion are presented in chapter 6. Chapter 7 concludes with the summary and an outlook.

# Biological Background: Proteins 2

This chapter gives a brief introduction to the chemical and biological background of proteins. First the hierarchy of protein structure and chemical properties are explained, followed by the principles of interaction, which have to be considered in the development of protein docking algorithms. The chapter ends with an overview of techniques for acquisition of interaction data, which provides the input and training data for simulation and prediction.

## 2.1. Structural and Chemical Properties

Proteins can be classified into various groups, depending on features of their structure, domains or based on their function within the metabolism such as signal transduction, cell skeleton or catalysis of metabolic reactions. The enzymes are classified in the hierarchical enzyme classification (EC) number scheme, controlled by the international union of biochemistry and molecular biology (IUBMB). Members of the same EC (sub-)group catalyse similar reactions.

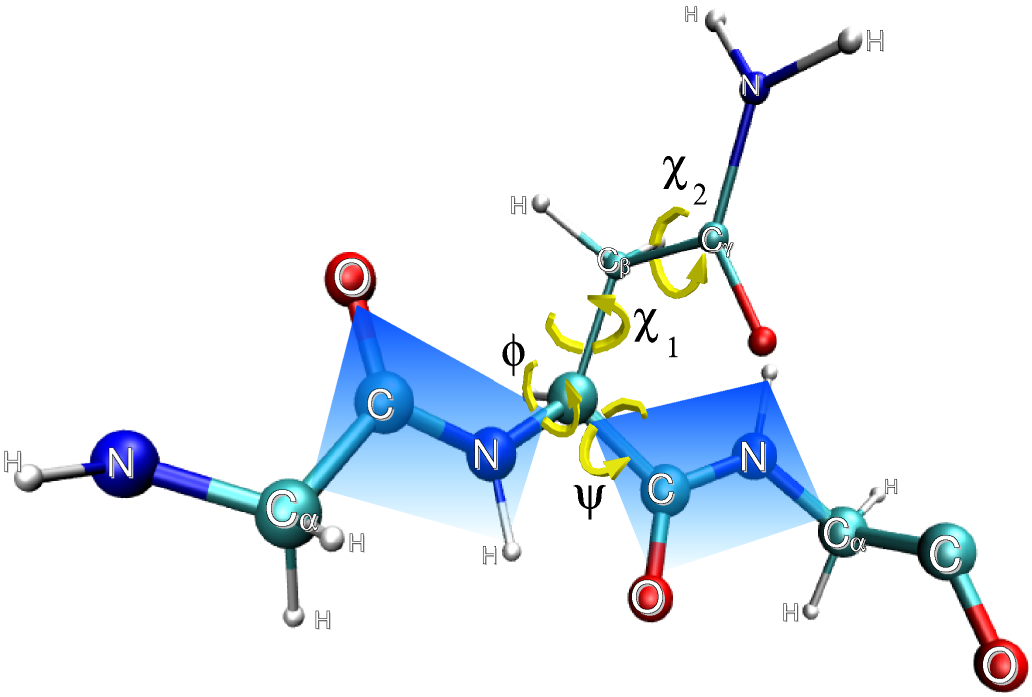
Proteins are described by their primary, secondary and tertiary structure. Some proteins come in larger biological units with a so called quaternary structure: several subunits are combined into functional complexes.

### 2.1.1. Structure

Proteins are molecules built from a linear sequence of 20 common amino acids[1] according to the information in the coding DNA. The length ranges from 50 up to 3000 residues and longer. Smaller molecules are referred to as polypeptides.

[1] There are several “unusual” amino acids like selenocysteine, the newly discovered pyrrolysine [Atkins02] or hydroxyproline, the latter being a post-translational modification of the normal α-amino acid.

(a) $\cdots - Val - Glu - Tyr - Phe - Gly - Leu - His - Asp - Gly - Pro - His - \cdots$




(b)

Figure 2.1.: Two representations of a protein: (a) the sequence using three-letter codes, (b) ball-and-stick model, showing amid-planes between two adjacent residues. Side chains (just a $H$ for the two $Gly$, $NH_2COCH_2$ for the $Asn$ in the centre) are attached to the $C_\alpha$. $\phi$ and $\psi$ angles determine the backbone or secondary structure, $\chi_{1\cdots n}$ the side chain conformation.

The sequence of amino acid types is called the primary structure (cf. figure 2.1(a)). Amino acids have a uniform base $NHC_\alpha RHCO$, the side chain (often denoted $R$) is attached to the $C_\alpha$ atom and determines the amino acid type and its physico-chemical properties. The side chains are linear, branched or contain a ring system. They have up to four degrees of freedom, called $\chi_1$ to $\chi_4$ (see figure 2.1(b)). These continuous angles are often divided into discrete rotamers that cover e.g. 120°. Amino acids can be grouped according to their properties into a matrix of hydrophob or hydrophile and positive/neutral/negatively charged classes.

Along the backbone only the bonds between $N - C_\alpha$ and $C_\alpha - C$ have a rotational degree of freedom. The angles are named $\phi$ and $\psi$ respectively (see figure 2.1(b)). Some combinations of $\phi$ and $\psi$ are more favourable and result in a regular secondary structure that can be classified as right/left handed $\alpha$-helix, parallel/antiparallel $\beta$-sheet or loop region. Examples are given in the figures 2.2(a) and 2.2(b).

While the secondary structure is tied to the sequence and divides it into stretches of helices, sheets and loop regions, the tertiary structure can only be described through 3D atomic coordinates[2] of all atoms in the protein.

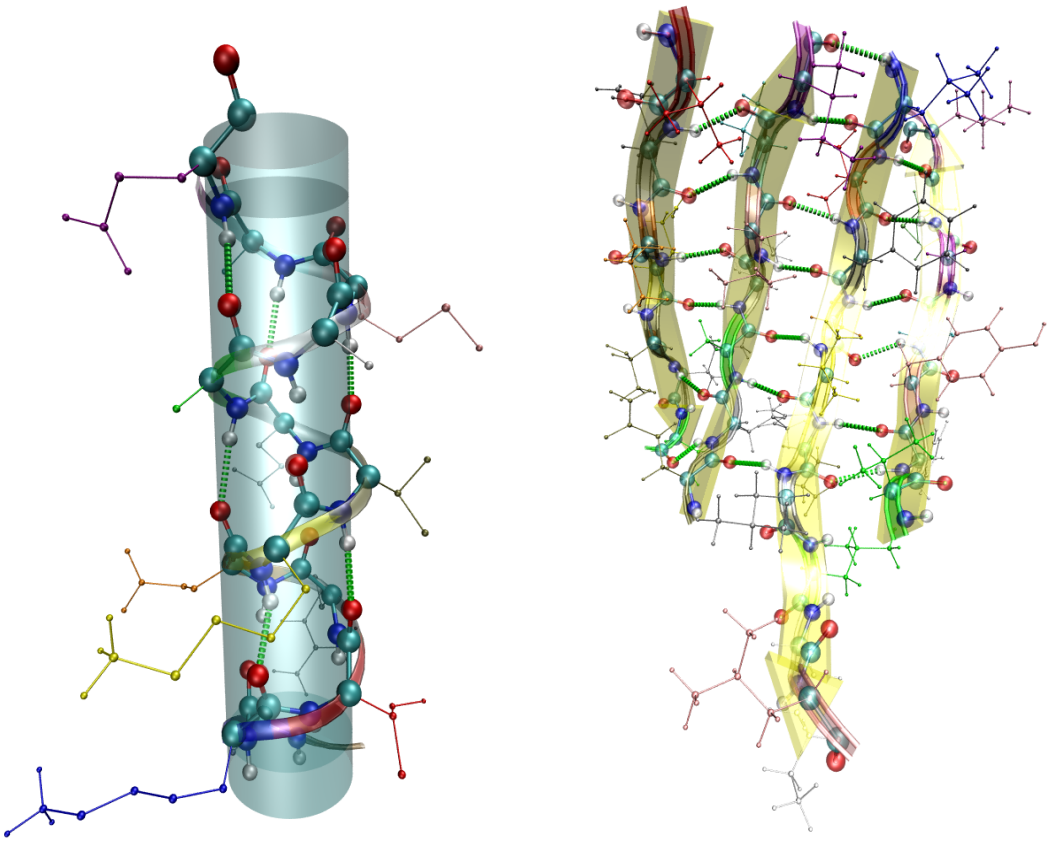
During their synthesis proteins fold into their secondary and tertiary structure and expose a part of the amino acids to the solvent and thus to potential docking partners. This solvent accessible surface (SAS) determines the specificity of a protein for docking partners, because the governing forces are mostly short range, weak forces as described in the next section.

### 2.1.2. Weak Forces

Within a chemical system, a number of attractive or repulsive forces determine the system's energy. Bonds[3] are established if they lead to a lower energy level and thus preferable state. Several kinds of forces differ in strength and distance of reach.

Two charged atoms, a charged atom and a dipole ($\delta^+$ or $\delta^-$) or two dipoles exert an attractive force between each other. The strength depends on the difference in charges and decays depending on the distance $R$ with $1/R$, $1/R^2$ or $1/R^3$ respectively. It also depends on the dielectric constant of the surrounding medium, which is higher for the solvent, opposed to the solvent excluded protein core or the also solvent excluded contact site of a complex. Polar or charged amino acids are usually located at the protein surface, whereas the core is usually hydrophobic. If a protein misfolds during synthesis, so called chaperons unfold them again to re-initiate folding. The recognition of misfolds depends on the incorrect surface configuration and untypically large hydrophobic surface patches.

[2] This includes equivalent notations such as bond angles or internal coordinates.
[3] In this context this applies to both covalently bound atoms, as well as non-covalent bonds.



(a) α-helix

(b) β-sheet

Figure 2.2.: The two common secondary structures. The backbone atoms are shown in CPK colours, the (down scaled) side chain atoms and the backbone trace have a colour for each residue. $H$-bonds are shown in green, they stretch from an oxygen in residue $n$ to a hydrogen in residue $n+4$ for helices, and between opposite residues in sheets.

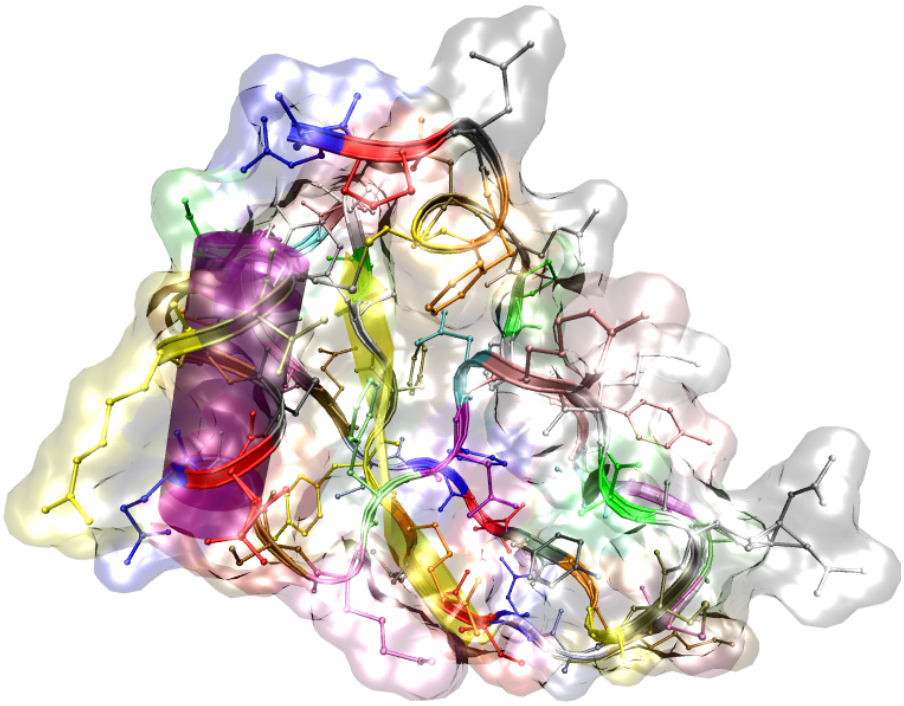


Figure 2.3.: Tertiary structure of a trypsin inhibitor (6pti). The backbone atoms are shown in CPK colours, the secondary structure elements as cartoons. The (down scaled) side chain atoms, backbone trace and the solvent accessible surface have a colour for each residue.

Even weaker forces are the van-der-Waals forces. Induced dipole moments in the electron clouds of neighbouring atoms cause attractive forces in the order of 0,1 - 0,2 kcal/mol, decreasing with distance as $1/R^6$. At the same time they induce a repulsive force proportional to $-1/R^{12}$. Combined they form a term $\frac{1}{R^6} - \frac{1}{R^{12}}$ also known as Lennard Jones potential [Stryer94]. The atom specific van-der-Waals radius is the radius where the energy minimum occurs, it measures between 1.2 Å (Hydrogen) to 2.0 Å (Carbon) for typical atoms in proteins. Though the van-der-Waals forces are weak in their strength, they play a fundamental role in the packing of proteins due to the large number of interacting atoms.

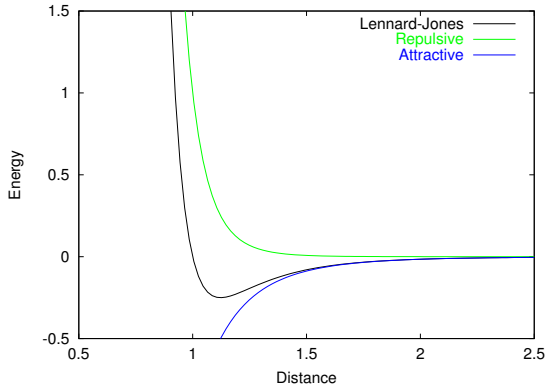



Figure 2.4.: Combination of attractive and repulsive van-der-Waals forces in the Lennard Jones potential, the minimum defines the van-der-Waals radius.

Hydrogen bonds are formed between neighbouring, non-bonded functional groups or atoms. They reduce the normal distance between unbound $H$ and $O$ or $H$ and $N$ of 2,6 Å and 2,7 Å respectively by about 0,8 Å. If a hydrogen bond is established, the donor (mainly $H$) borrows negative charge from the acceptors (electronegative atoms e.g. $N$ and $O$). The dipole forces act highly directed, so relative orientation of acceptor and donor is important.

## 2.2. Protein Structure Data

The model of β-Trypsin inhibitor (6tld in figure 2.3) is based on the 3D atomic coordinates of the protein. To obtain such 3D data, a solution of the proteins is treated with a variety of salt concentrations, temperature and concentration gradients to foster the formation of an ordered crystal structure (see figure 2.5). Using X-RAY "light", a scatter image is projected onto the detectors as a snapshot. The protein structure can be reconstructed from the phase and amplitude, where the phase has to be approximated, so several images need to be taken. X-RAY-diffraction provides the atomic structure of the molecules with a resolution of up to 0.5 Å.

Figure 2.5.: A crystal of protein substance, ready for X-RAY-diffraction. Courtesy of NASA [Horack97].

In the early 1970s protein structures were centrally collected by the Brookhaven National Laboratory (BNL) in the PDB database [Bernstein77; Bermann00], which was transferred to the Research Collaboration in Structural Biology (RCSB) in 1999. The PDB contains 3D coordinates of proteins and larger molecular assemblies as well as meta data such as literature references, their authors and the underlying X-RAY and NMR experiments. Today the RCSB collaborates with the European Biology Institute (EBI) in Cambridge and PDBJ in Japan to enhance the data model and consistency of the PDB.

Searches for proteins can be carried out by keyword or sequence search. Structural alignment methods and services like DALI [Holm93], CE-ALIGN [Bourne98] and PSI-BLAST [Schäffer01] allow to implement Query-by-content search facilities to retrieve homologous proteins or -folds. Built on top are hierarchical clustering schemes for domains like FSSP [Sander94] or CATH [Orengo97]. With a better coverage of protein space the PDB will be increasingly useful for statistical analysis of protein structures.

## 2.3. Protein Function and -Interaction

Protein function is defined as the role a protein plays in the large network of reactions in the metabolic pathways. One of the key concepts is the interaction between two or more proteins. During such an interaction the proteins are bound non-covalently into complexes. There is no single definition of a complex, which makes estimates about the number of different complexes difficult[4]. They can be binary complexes of small compounds, or large assemblies of proteins like the 70S ribosome in yeast [Sali03]. Complexes can either be stable, like the protein assembling ribosome, or transient, such as signalling or metabolic reactions (see below). Homo oligomerisation of single chains of a larger protein allows for very fine-grained concentration dependent gain control of overall activity, and allows for easier conformational changes upon ligand binding [Royer01]. A “malfunction” in protein interaction is involved in many diseases, such as Creutzfeld-Jacob or Alzheimer’s disease [Zhang97].

A well-known family of proteins are serine proteases, including digestive enzymes trypsin, chymotrypsin and elastase. They have 25-50% sequence identity, and an even closer related folded structure. The activity takes place if the substrate is bound at the active site via several hydrogen bonds, and in a sequence of transitional steps the substrate peptide is cleaved. Trypsin and chymotrypsin have specific cleavage sites, whereas elastases cleave behind smaller and hydrophobic residues.

[4] In an extreme view, a whole cell can be considered a giant complex [Sali03].

To prevent degradation of cells at the place of proliferation, they are synthesised as inactive precursors. Blood coagulation for example takes place if the extra residues at the N-terminus are cleaved at specific residues and the binding pocket changes into its active form. Inhibiting this process needs to (reversibly) block the active site from contacting the substrate. The inhibitors have strong binding capabilities, but a very low reaction rate. Other protein protein complexes are e.g. hydrolases, oxireductases and glycolases with their inhibitors [Zubay93].

Various properties of protein surfaces have been shown to be characteristic for complex interfaces. Depending on the protein family, the interface area usually buries a larger area of hydrophob residues, whereas in antigen interfaces they are very polar and thus hydrophil. It is also obvious that not too many contacting residues must have the same charge [Goede98; Thornton97].

A (now historic) method for modelling and investigating the properties of binding sites is described in [Walters86], where a negative model of the receptor site is built around a Corey-Pauling-Kolthun (CPK) model using a thermoplastic material. An actual photograph is shown in figure 2.6. Nevertheless, the method provides useful insights to the underlying principles of molecular interaction, e.g. for educational purposes.

Figure 2.6.: Hand modelled receptor site. Courtesy of Dr. Walters [Walters95].

### 2.3.1. Experimental Complex Determination

Several methods exist for experimentally determining protein interactions at different levels of detail.

The to date most powerful technique towards complex structure is X-RAY-crystallography. A solution of the (putative) complex is crystallised. Since the full 3D data is available, information about the stoichiometry ("how many ligands do bind?") and residue/atomic contacts are known. Modern X-RAY-crystallography sites can produce the data for a typical protein or complex (once synthesised and purified) using a highly automated pipeline within a day [Sali03].

Other structure determination methods like NMR, electron microscopy or electron tomography also show the shape of the subunits and subunit contacts, but e.g. NMR does not provide easy access to the stoichiometry, or they do not provide the structure on atomic levels [Sali03]. All of them need quantities of purified protein substance.

Protein arrays are one of the high-throughput methods, where probes are "printed" on a chip surface and exposed to sample protein [Phizicky03]. The results are read using fluorescent markers and image processing techniques. The knowledge about complex partners can then be used to determine the 3D structure either experimentally or theoretically using computational methods.

### 2.3.2. Interaction Data

The interaction data obtained experimentally is collected for several model organisms. The Yeast Proteome Database (YPD) contains some 11.000 interactions in yeast [Costanzo01]. It is located at the Munich information center for protein sequences (MIPS).

The site http://binddb.org/ lists an overall of 62 so called "interaction databases". Most of them deal with qualitative description of interactions in metabolic pathways and return textual information or links to scientific papers. One of the exceptions is the BINDINGDB [Chen02a], which publishes quantitative data like binding free energy as well as a quantitative description of the experimental setup:

> "The Binding Database project aims to make experimental data on the non-covalent association of molecules in solution searchable via the WWW. The initial focus is on bimolecular systems, but data on host-guest and supra molecular systems are also important and will be included in time.
>
> It is expected that the enhanced access to data provided by this resource will facilitate drug-discovery, the design of self-assembling systems, and the development of predictive computer models of binding." [Chen02a]

Once the BINDINGDB has been populated with a large number of interaction data, quantitative calibration of docking programs can be performed.

Another recent effort is the INTACT project, located at the EBI and closely integrated with the SWISSPROT database. It aims to "define a standard for the representation and annotation of protein protein interaction data, provide a public repository with data from experiments or curated literature and related software" (http://intact.sf.net/). As explained in chapter 4, the quality and quantity of experimental data is vital for development and training of docking algorithms.

## 2.4. Conformational Changes

Proteins can undergo several kinds of conformational changes. There are large scale shear-like or hinge-bend domain movements [Gerstein94]. On a smaller scale, parts of the backbone can be flexible, especially in loop regions where restrictions are less tight than in helices or sheets.

The (structural) dynamics of the docking process cannot easily be determined, not the least because of their very small time scales. The molecular movement database [Echols03] provides close to 200 structures, for which 4000 movies and trajectories with various visual appearances have been generated. They are based on multiple crystal structures or simulations, and morph smoothly between different transitional states, but chemical realism was not the primary goal.

Small-scale flexibility occurs if amino acids change their side chain conformation. Depending on the residue's type, its rotamer and the context, residues have different probabilities of changing their rotamer. This will be discussed in detail in the next section.

Finally the side chains and atoms show stochastic movement, which does not affect the solvent accessible surface based docking approaches.

## 2.5. Rotamer Flexibility

Recent work [Koch00; Zöllner02; Koch02b] has shown a variable degree of flexibility depending on the residue type. The residues have differently sized side chains with varying physico-chemical properties. A rotamer change might be difficult due to the expected intramolecular steric clashes, or due to unfavourable energetic effects resulting from e.g. a breakage in intramolecular H-bonds. If flexibility occurs regardless, it is usually caused by an overall decrease in the system's free energy.

To assess flexibility, two approaches are possible, one descriptive examining available data, one predictive based on simulation.

### 2.5.1. Rotamer Statistics

For some complexes both the bound and unbound structures are available in the PDB. After a sequence alignment step the residues are unambiguously mapped between the bound and the unbound PDB entry, and their side chain angles (and differences thereof) can be examined.

Some PDB entries do not contain atom positions for all residues in the sequence (especially at the beginning or end of a chain). It can also happen that the same protein has a different numbering (like starting at 0 or 1) or residue IDs differ between different crystallographers in their positional identifiers[5] or include alternate locations for residues if the exact position is unsure. The comparison therefore has to be done on the basis of the position *within the sequence* which is – by definition – the same for sequence identical chains.

For these rotamer changes a modified rotamer library has been built [Koch01] which contains the probability for an amino acid in a given conformation to change into another rotamer.

The statistical tendencies for rotamer changes have been analysed in [Koch00]. The amino acids can be classified in groups with high probability for rotamer changes (e.g. 30% for Arginine) and those where the side chain structure remains fairly fixed. A larger solvent accessible surface

[5] If a protein is closely related to another one, the crystallographer might choose to denote an insertion with a residue ID like "188a", which is biologically sensible, but causes confusion for automated parsers.

also increases flexibility. Other influences include the secondary structure or the rotamericity[6]. A flexibility value can thus be assigned to each residue individually, or with respect to the surrounding context.

### 2.5.2. Energy Calculation

A predictive approach towards assessment of protein flexibility are energy calculations using the AMBER force field [Koch02b; Zöllner02]. It is based on the tendency of a system to occupy a state with minimum energy. Rather than a full prediction of the docked side chain conformation a measurement for the uncertainty of any single residue is calculated.

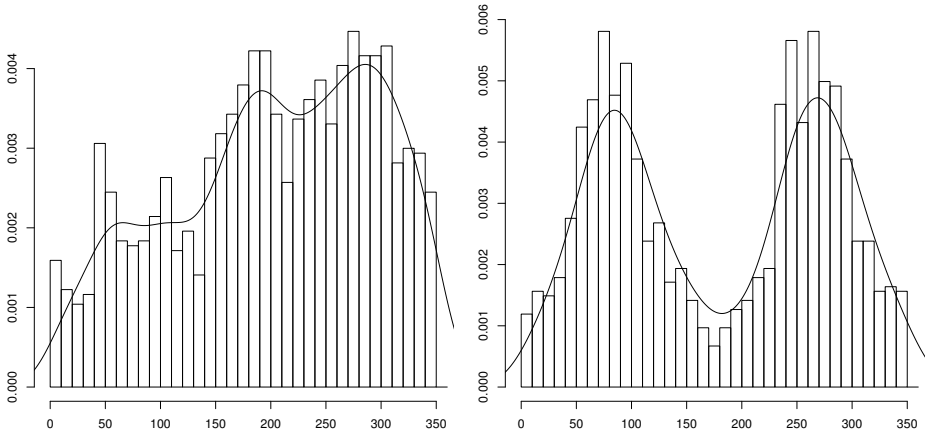



Figure 2.7.: Distribution of energy minima for $\chi_1$ of Arginine (trimodal) and the symmetric Phenyl (bimodal). In [Koch02b].

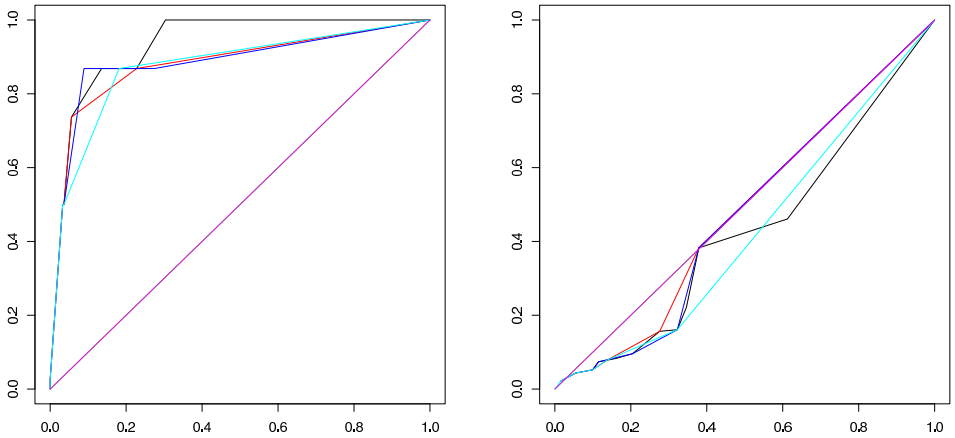
For every single residue and $\chi$ angle in a protein the bonds are rotated by 360° in steps of 5°. A combined sampling of the whole conformational space is not feasible[7]. For each of these conformations the total energy of the structure is calculated using the AMBER force field.

Even with this simplified model the distribution of angles where an energy minimum occurs (see figure 2.7) complies nicely with rotamer distribution found in the crystallographically determined PDB structures.

Taking the approach one step further, a conformational change can be predicted: if the system is in a state of a (local) energy minimum, changing into another (local) minimum involves crossing an energy barrier. If for a given residue a conformation can be found in the sampling set that has

[6] Distance to the center of the rotamer.

[7] Consider an average protein which might have 200 amino acids, with two $\chi$ angles each. The full search space at 5° sampling is $200^{72^{72}}$. Even a coarser sampling, e.g. restricted to three rotamers per angle yields $200^{3^3} = 5 \times 10^{20}$.




(a) LYS: $\int_0^1 ROC = 0.82$ (b) SER: $\int_0^1 ROC = 0.47$

Figure 2.8.: ROC plots for a) successful and b) unreliable classification of rotamer changes. Four different methods are used during normalisation of the input data. A perfect classification has a value of $1.0$, random prediction corresponds to $0.5$. The prediction is robust against variations in the normalisation step. In [Zöllner02].

a lower energy than the unchanged protein, a rotamer change is advantageous to the system. For several types of residues a conformational change can be predicted successfully. The rate of false positives or false negatives of these predictions depends on the threshold applied to the energy difference. The receiver operator characteristics (ROC, see figure 2.8) show the performance of a simple linear threshold classifier for all thresholds simultaneously. The larger the area below the ROC plot, the better the underlying classificator. The area reaches $1.0$ for a perfect classification and $0.5$ for random predictions. A value smaller than $0.5$ is a misprediction and usually reveals a flaw in the classifier training. A flexibility value can thus be assigned to each residue in a protein, weighted by the quality of the prediction.

The next chapter will discuss the application of computational techniques to the subject of protein interaction.

# Computational Protein Docking 3

Protein docking research can be divided into two areas depending on the size of the target molecule:

**Protein Ligand docking** is about docking small organic compounds or short polypeptides against a receptor protein and is widely used by pharmaceutical companies. Both the receptor's active site and the ligand can be modelled flexible using current methods [Jones97; Claussen01]. Screening of large virtual libraries is possible, see e.g. [Waszkowycz01].

**Protein Protein docking** deals with proteins docking to other proteins, (larger) polypeptide or DNA, see below.

In the remaining part of this work protein ligand docking will not be considered further. Since protein ligand interfaces are much smaller, the attractive and repulsive forces have to be modelled in much more detail using atomic representations, directed $H$-bonds etc. . However, due to the smaller size of the input the computational complexity is less a problem compared to protein protein docking. The following sections present different approaches to the protein protein docking problem.

The process of screening must be thought of as a hierarchical process, starting with the whole compound library. A set of filter modules removes candidates until the result set satisfies the search conditions. The modules need to be stacked in such a way that the selectivity increases as the complexity of the algorithms, and thus their runtime needs, increase.

For docking algorithms the first filter steps should select appropriate protein classes as candidates, followed by fast shape-based matching. In a step beyond the screening, energy based simulations analyse each candidate for the fitness regarding to the required biochemical function.

This chapter will give an overview of existing protein docking systems and scoring functions, including some novel approaches based on a combination of simulation and experiments that are

less complex than X-RAY-crystallography. A section on flexibility aspects in docking algorithms is followed by an introduction to the 1:N docking.

## 3.1. Existing approaches to Protein Docking

Tackling the protein docking problem using computational methods involves several steps and follows the approach of general pattern recognition systems (fig 3.1).

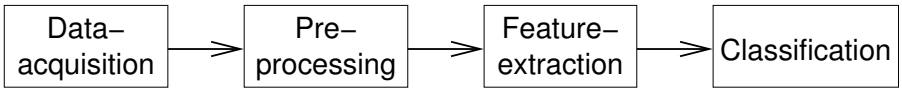



Figure 3.1.: General pattern recognition system

The data acquisition is usually not done in realtime, in contrast to image processing where the sensors (cameras) are usually connected to the system online. The crystallographic experiments deliver their data to databases like the PDB, where they are gathered through direct access or a local replica. An (optional) preprocessing stage checks for inconsistencies, performs sanitising actions and further enhancements. The feature extraction has to represent the biochemical properties associated with the interacting residues and atoms. The final classification step simulates whether and how the proteins interact.

The evaluation measures the difference between the prediction and the crystallographically resolved complex structure. The difference is usually measured with the Root Mean Square Deviation (RMSD), calculated over all atom pairs $(\vec{x}_i, \vec{y}_i)$ in the bijection $B$ between the two proteins as follows:

$$RMSD = \sqrt{\frac{1}{N}\sum_{i \in B}(\vec{x}_i - \vec{y}_i)^2} \tag{3.1}$$

The bijection $B$ can be constructed from the $C_\alpha$ atoms or can contain all heavy atoms ($C, N, O, S$) of the side chains.

### 3.1.1. Energy Calculation including Solvation free Energy

An accurate docking algorithm needs to compute the intra- and intermolecular forces with high accuracy. The approach by Lenhof et al. [Kohlbacher01] solves the Poisson-Boltzmann equations for prediction of polar contributions to solvation free energy.

To incorporate side chain flexibility a rotamer library is used to enumerate a discrete set of conformations to be considered. To avoid steric clashes side chain demangling is performed using the branch-and-cut algorithm [Althaus02] to avoid the computational complexity inherent to a search tree including all possible rotamer conformations.

### 3.1.2. Interactive Molecular Dynamics

A very intuitive to use system has been developed by Schulten et al [Stone01]. A force-feedback mouse with 6 degrees of freedom was combined with realtime molecular dynamics (MD) simulation on a small cluster containing 32 CPUs. The forces acting on a small molecule are computed in real-time, and are (scaled accordingly) transmitted to the feedback device. Together with a 3D representation on screen the user can perform a directed search with immediate feedback about its plausibility.

### 3.1.3. Geometry based Protein-Protein Docking

One of the existing docking systems is the docking suite 3D-DOCK by Sternberg et al. [Moont99]. It has historically evolved from the FTDock software and includes several modules. The proteins are first sampled into a discrete grid representation, and a geometric scoring function assesses shape complementarity. Afterwards the docking hypotheses are scored with a statistical residue-pair-potential, i.e. the trained possibilities of residue-type contacts are summed up. An additional filter program can apply manually constructed constraints, if e.g. the catalytic residues are known and their presence in the contact site is an absolute must.

A similar system called ZDOCK has been developed by Chen et al. described in [Chen02b]. It has a runtime requirement of 10-20 hours per complex on a SGI Origin 2000 computer.

### 3.1.4. Post-Docking Filters

The approach by Vera Grimm [Grimm02] is the calculation of a knowledge based pair-potential. Given a ranked list of docking hypotheses the correct contact can be extracted from a set of near-native ones.

The probability for a true docking hypothesis can be predicted from the atom distribution in the contact site. The training data for this knowledge based material is extracted from the contact sites of complexes existing in the PDB. The statistical approach avoids the explicit modelling of a formula for the energy function. The trained probabilities include all physico-chemical influences implicitly.

This filter requires the native or near-native conformation to be in the result set. If it is not present, the results are not meaningful.

### 3.1.5. Prediction of NMR Spectra

A new approach to protein docking is a combination of experimental and computational methods [Kohlbacher01].

The computational part of this approach consists of a search stage sampling translation and rotation of the docking partners. For each of these hypotheses an 1D NMR-spectrum is simulated. 1D NMR-spectra show characteristic peaks for atoms and functional groups of the molecule. Their spatial relations cause peak-shifts, relative to the position of the isolated atoms and groups.

The experimental setup measures the 1D NMR-spectrum for the existing complex. The predicted spectra are compared to those acquired by real NMR spectroscopy and scored by similarity.

Though the experimental part of this methods requires access to lab equipment and expertise in NMR measurements, it is by far less expensive both in equipment and labour compared to crystallography or 3D NMR spectrography.

## 3.2. Incorporation of Flexibility into Docking Algorithms

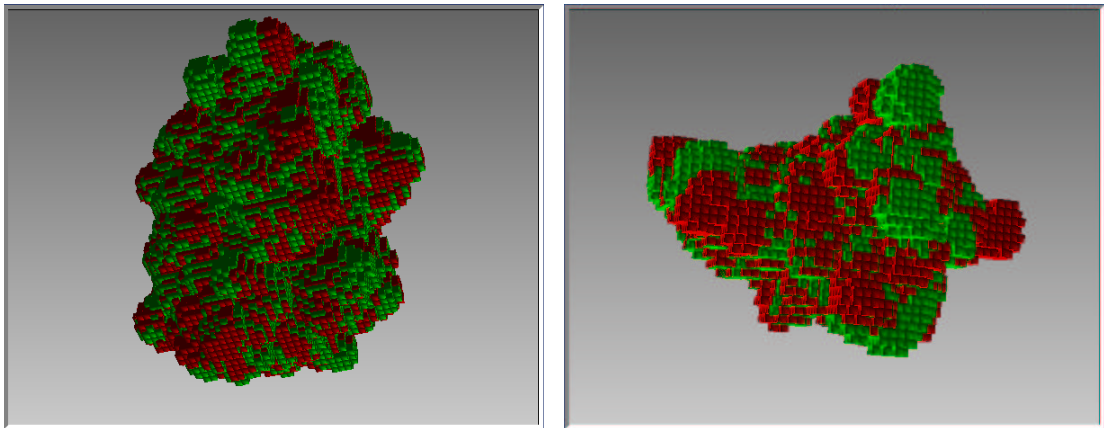
The main problem for a docking algorithm introduced by induced fit is that steric clashes are penalised, which might not occur after a change. Protein flexibility can be accommodated on several levels with increasing computational complexity:

a) rigid body docking without any flexibility
b) partially flexible docking with one molecule (usually the receptor) kept rigid and
c) with flexibility for both docking partners.

The most simple solution to flexibility in docking algorithms is to ignore it altogether. This has been shown for low-resolution (around 7 Å) models by Vakser et al. [Vakser99]. As a positive side effect the low-resolution scoring function is invariant towards small-scale conformational changes. The accuracy of the results is, however, limited by this low resolution, and contains a large number of false-positive results.

Rigid body approaches can also accommodate protein flexibility by allowing some penetration of the proteins, limited to a “soft belt” at the surface [Fernandez-Recio02; Jiang02]. Steric clash penalties are reduced if the clash occurs within the soft belt.

Both the low-resolution and the soft-belt approach assume the same (constant) flexibility regardless of the underlying amino acid. A smaller selectivity and thus higher rate of false positives is to be expected. The “softer” this shell is, the more false positives have to be expected. If no further constraints are applied, anything can be docked into anything.



(a) Enzyme (b) Inhibitor

Figure 3.2.: Structural alignment of the complex 2PTC (in green) and the two unbound partners 1TLD and 6PTI (both in red). Areas with a single colour determine the location of conformational changes with an effect on the solvent accessible surface. Areas with alternating red/green voxels do not change considerably.

A more elaborate approach is a non-uniform soft-shell which penalises steric clash according to the specific side chain flexibility. Since the flexibility parameters can be computed offline, this additional step does not influence the run-time of the actual docking algorithm. This has to date only been implemented in the ELMAR docking system, and will be presented in the chapter 5.

To model full flexibility all rotatable bonds (side chains and/or the backbone) can be allowed to change during the simulation. To reduce the search space, a subset of bonds can be selected, e.g. within the binding site[1], with the rest of the protein held fixed. Alternatively rotations can be restricted to a discrete set of rotamers, or a combination thereof.

A compartmentisation of the conformational search-space can be achieved if the side chain angles are only allowed to be in one out of three (or two, depending on the amino acid type) discrete rotamers, which cover 120° (or 180° respectively). The rotamer-combinations together with the associated probability of occurrence in a test set are combined in a rotamer library (see sec-

[1] The required location of the binding site may not always be available.

tion 2.5.1). The predictive approach described in section 2.5.2 also provides a set of conformations ranked by an energy-related scoring. During the docking run the query-proteins can be modified such that the most probable side chains are tried first.

Abagyan et. al. [Totrov94] report the docking of a lysozyme-antibody complex at 1.6 Å accuracy. However, this precision requires a runtime of >100 hours.

Instead of explicitely allowing flexible bonds, a larger set of structures can be combined to form an ensemble which is docked. One of the remaining problems is the generation of the ensembles. If multiple X-RAY- or NMR structures are known for the protein, they can be superimposed and merged into a single structure. Those parts that differ significantly are elements of an ensemble structure [Claussen01]. Compatibility graphs identify those substructures that can occur together and represent a valid conformation of the protein. Two examples with many known structures are the HIV-1 protease and DFH reductase. Ensembles can also be generated using MD simulation. The main benefit of this approach is that incompatible conformations of the flexible substructures are eliminated early in the process.

# Test and Training Data 4

Developing a protein docking problem can be interpreted as an optimisation problem, where the parameters of an algorithm need to be tuned to give optimal results. The optimisation has to be done on a training set of data, the performance is evaluated on a separate set of test data.

If only few training sets are available, the generalisation to new data cannot be guaranteed. The algorithm tends to overfit and “memorises” the data. The more parameters need to be optimised, the larger the training set has to be.

The protein structures are collected in the PDB, which now exists for 30 years and has been subject to some criticism over the years: . The following quote is taken from

> “The PDB structure entries, consisting of a collection of files having nondescript names, cannot be easily grasped in a biochemically meaningful context. Manually organising the structures based on the descriptive information in the files is becoming less and less practical as the database expands.” [Pearlstein96]

Several authors proposed attempts to re-organise the PDB content [Hashimoto94; Abdallah98]. Both systems contain the complete structural data, including the 3D positions of all atoms. The flexibility possible through the use of a database comes at the expense of complexity on access of the data. These approaches have not been widely adopted, most applications are still designed to read the PDB files.

This chapter reviews some of the test sets available in the literature. To overcome the bottleneck of hand crafted test sets, automated methods for discovering new test cases in the fast growing protein databases are proposed. As an improvement over “[...] Manually organising the structures based on the descriptive information [...]” a schema for a relational database containing the descriptive meta data and protein sequences will be described.

## 4.1. Available Benchmark Data Sets

For the bound docking case several test sets exist in the literature. They are usually hand-selected from the PDB, since no consistent labelling of complexes is done in the database.

Ackermann [Ackermann98] lists 51 complexes in the enzyme/inhibitor and antibody/antigen class and some homodimers. The set is suitable only in the case of bound docking.

In the unbound case both the two unbound docking partners as well as the resulting complex need to be resolved. Those test sets can be built on top of the bound test sets by searching for the unbound conformation of the docking partners.

If one partner has not been resolved in the unbound form, it can be extracted from the complex, softening the requirements on the data set. Induced fit cannot be modelled, though. In any case the structures need to fulfil further quality requirements, such as a maximum resolution or absence of small molecules in the crystal structure.

Nussinov [Norrel99] evaluates scoring functions on a test set of 9 receptor and 9 ligand molecules in their unbound form combined into 19 so called “mock complexes”. These are the structural superposition of the unbound molecules onto the respective complexed conformation. The corresponding complexes are not mentioned in the paper and need to be derived from other publications and PDB searches.

Sternberg [Betts99] has 31 test cases in enzyme/inhibitor and antibody/antigen class and some from various other classes. 23 of them use at least one partner in the complexed confirmation, only 8 employ two unbound proteins.

The review paper [Halperin02] lists an overall of 86 test cases for 32 complexes. Well known complexes like 2PTC where receptor and ligand are determined in 3 and 4 variants respectively count for $3 \times 4 = 12$ of the test cases.

| Complex ID | Chains |
|---|---|
| 2PTC | E+I |
| 1BDJ | A+B |
| 1CGI | E+I |
| 1WQ1 | G+R |
| 2TGP | Z+I |
| 1FSS | A+B |

Table 4.1.: Common test cases in the literature. For these complexes sequence identical unbound structures exist in the PDB. The remaining published test sets are collected in appendix A.

The set theoretical union of these test sets contains 197 combinations of bound and unbound structures, 160 of them are unique. They cover 75 different complexes.

Some of the available test cases do not comply with the requirements mentioned above. The unique set of the three test sets consists of 160 test cases, a quarter of them deliberately employs one chain in its complexed form. Among the 48 cases where true unbound structures with one chain each are used, 35 have differences in one or even both sequences between the complexed and unbound state. The remaining complex entries are shown in table 4.1. Some of the differences result from point mutations or possibly sequencing glitches. Others are length differences in the number of sequenced amino acids of up to ten residues. Careful inspection is needed to ensure that these mismatches do not impose artefacts on the prediction. They can be safely included in the test set if neither the active site nor the overall folding pattern is affected. Mistyped PDB IDs in the published data sets complicate this problem even further.

The data set COMBASE by Vakser et. al. [Glaser01] contains a large list of PDB entries with chains that have an interface with each other, but no unbound data is available. These complexes can be used during the training of post-docking filters.

A special case are blind tests like the international competition CAPRI (Critical Assessment of Predicted Interactions, http://capri.ebi.ac.uk/) [Vajda02; Janin03]. A target complex is selected before it is deposited in the PDB. It's unbound components are made publically available and docking hypotheses can be submitted. The correct (complexed) structure will be held back until the closing date of the contest. The evaluation discusses not only the raw performance of the algorithms, but also their strengths and weaknesses.

Despite the size of the PDB, the resulting data sets are relatively small. As shown in figure 4.1 the growth of the PDB is exponential, and will grow faster with the use of high throughput experimental methods. The need for semi-automated test set creation is obvious. The following sections will explain heuristics to gather test sets based on sequence identity or the available meta data.

## 4.2. Automated Test Set Creation

For semi-automated test set creation two schemes have been developed, both querying the PDB for tuples of complexes and the associated unbound conformation of the proteins. The first starts from unbound chains and searches for corresponding complexes, the second uses the available meta data to find complexes, for which unbound structures are retrieved.

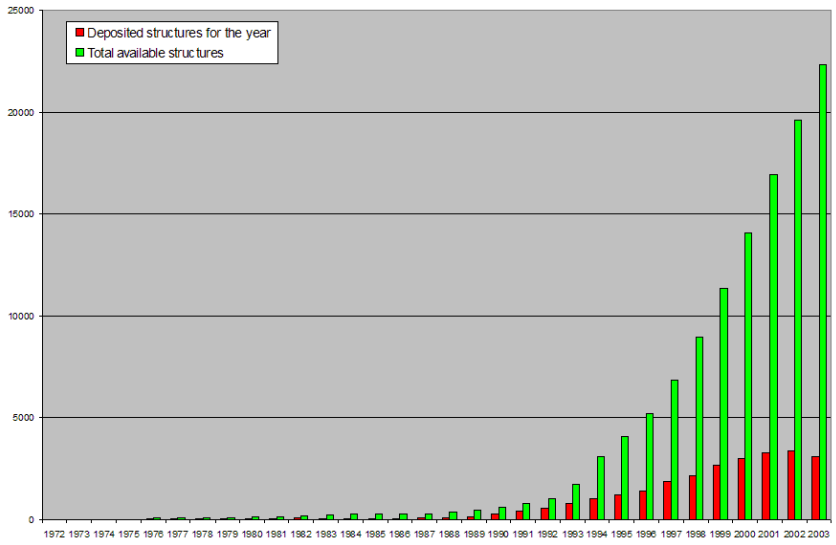



Figure 4.1.: PDB content growth between 1972 and september 2003. Courtesy of Rutgers University www.rcsb.org.

### 4.2.1. Two-chain Complexes

A straightforward definition of a binary protein protein complex is a PDB entry consisting of two chains, with each chain being crystallographically resolved individually as well. Multiple unbound chains can be combined using their cross product, as depicted in figure 4.2. To avoid small polypeptides a minimum chain length of at least 35 residues is required, this number has been taken from [Zubay93]. A resolution between 0.5 Å and 2.5 Å ensures that neither theoretical models[1] nor low-resolution entries are selected. If for each chain a sequence identical one can be found in the PDB, this is a valid training/test candidate.

Those requirements can easily be described as SQL statements on the given database schema, as shown in listing 4.3. The query selects a total number of 324 entries with two chains which are considered complexes, and from 1 to 781 sequence identical unbound chains for each part of the complex for a total of 82034 test cases.

The results show a number of different test cases for each complex, ranging from one available pair of unbound proteins to 5041 (=71$\times$71) for some homodimers like 1LKR. For 2PTC chain E and I there are 66 and 4 unbound entries respectively, for a total of 264 test cases.

[1] Theoretical models are assigned a resolution of 0 Å. Some modelled structures are optimised using an energy function, if a docking algorithm used the same function in the scoring stage, the result is not meaningful.

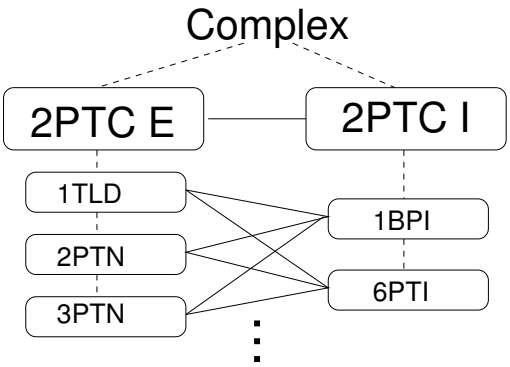



Figure 4.2.: Cross product of unbound chains that are sequence identical to the complex 2ptc.

Due to limitations of the PDB format[2] not all of these test cases are usable for unbound docking: some contain small organic molecules stored not as a protein chain, but as a set of `HETATOM` records and can influence the conformation of the protein. Closer inspection reveals that 17 of the single chain enzyme entries are inhibited by small organic molecules. It is clear that the dataset needs further cleaning. The intersection between the results of several heuristics will be addressed in the next sections.

### 4.2.2. Keyword based Classification Scheme

The PDB AT A GLANCE database [Pearlstein96] is a set of dynamically generated HTML pages that use keyword searches to classify proteins. Classification is done into different families (including enzymes, blood coagulation proteins or viral molecules). With a proper set of keywords it is also possible to distinguish between (non-) complexed, inhibited or activated molecules. The search terms cover the `Compnd, Expdat, Header` and `Source` records in the PDB entries.

The keyword search also finds multi-chain complexes. Those can be complexes between two or more single domain proteins or include at least one multi domain protein with domains individually assigned a chain. For docking experiments exactly two complex parts are needed. In these cases heuristics have to combine the chains into two assemblies before further processing. One example is the complex 2KAI, with the unbound enzyme 2PKA chains AB and inhibitor 4PTI. From all PDB entries classified as complex by PDB AT A GLANCE a sample rule and PDB IDs are shown in figure 4.4.

[2] The `HETATOM` records contain various small molecules, like water, salt or small inhibitors. It is not possible to easily distinguish between inhibiting and “other” `HETATOM` records.

```
select  *

from    Protein as P1,                  # Unbound Chain
        Chain as C1,
        Protein as P2,                  # second Unbound Chain
        Chain as C2,

        Protein as P3,                  # Complex Chains
        Chain as C3,
        Chain as C4

where

        P1.Entry = C1.Entry             # Tie Unbound Protein and Chain tables
and     P2.Entry = C2.Entry             #             dito

and     P3.Entry = C3.Entry             # ... and Complex Chains in single Entry
and     P3.Entry = C4.Entry

and     C3.Chain_Id < C4.Chain_Id       # use one of C3/C4 and C4/C3 pairs

and     C1.Sequence  = C3.Sequence      # Sequence identical Complex/Unbound
and     C2.Sequence  = C4.Sequence      # Sequence identical Complex/Unbound

and     P1.Chain_No = 1                 # Unbound has one and only one chain
and     P2.Chain_No = 1                 # Unbound has one and only one chain
and     P3.Chain_No = 2                 # Complexes with *exactly* two chains

and     C3.Res_No > 35                  # Avoid small Polypeptides
and     C4.Res_No > 35

and     P1.Resolution between 0.1 and 2.5       # Avoid Models/NMR
and     P2.Resolution between 0.1 and 2.5       #
and     P3.Resolution between 0.1 and 2.5       # and bad Resolution
```

Figure 4.3.: Database retrieval of complex/unbound test cases. The query is a triple join of the `Protein` table. All entries from `P3` with two chains are selected, and all entries `P1,P2` which contain a single sequence identical chain.

```
select Entry, Header

from Protein where

Header like "%peptide%"
 AND Header like "%hydrolase%"
  OR Header like "%protease%"
 AND NOT Compnd  like "%inhibitor%"
```

| Entry | Header |
|---|---|
| 1APZ | Complex (Hydrolase/Peptide) |
| 1BLL | Hydrolase(α-aminoacylpeptide) |
| 1IBC | Complex (hydrolase/peptide) |
| 1LAM | Hydrolase (α-aminoacylpeptide) |
| 1LAP | Hydrolase(α-aminoacylpeptide) |
| 1NS3 | Complex (Hydrolase/Peptide) |
| 1PTT | Complex (Hydrolase/Peptide) |

Figure 4.4.: Keyword search for complex entries, the SQL-query is shown on the left, an excerpt of the result set on the right. These Proteins are members of EC class 3.4.x.x .

The PDB AT A GLANCE result set contains an overall set of 1642 (partially multichain) complexes, from which 184 match the chain-name heuristics and have the unbound components in the PDB. The cross product of the respective unbound proteins combines into 180133 test cases.

The search discovers several entries with only one chain, which are classified as complexes. This happens usually if the protein is complexed with a small ligand encoded in PDB's `HETATOM` records. Further constraints (e.g. `Chain_No > 1`) eliminate this problem.

## 4.3. Combining the Test Sets

The results of the different heuristics can be combined to obtain a consensus test set which satisfies multiple requirements. The combination can easily be described as a `join` operation on the relational test set table. The predicates are part of the conjunction in the `where` clause. Figure 4.5 has a graphical representation of the result sets.

The intersection between the 2-chain complexes from the above section 4.2.1 and those published in the literature covers 23 test cases covering the four complexes 2PTC, 1CGI, 1WQ1 and 2TGP. The intersection between the keyword based classification scheme and those published in the literature also covers 23 test cases on four complexes, 1WQ1 is not part of the set, instead the entry 4HVP is included.

The 2-chain complexes from section 4.2.1 and those from the keyword based classification scheme have 66 complexes in common, with 13005 test cases. These are those entries that have been declared as complexes in the PDB entry, and have single-chain unbound entries for each of the two complex chains.

Finally, the intersection between all three sets contains the three common complexes, together with 22 combinations of the respective unbound chains.

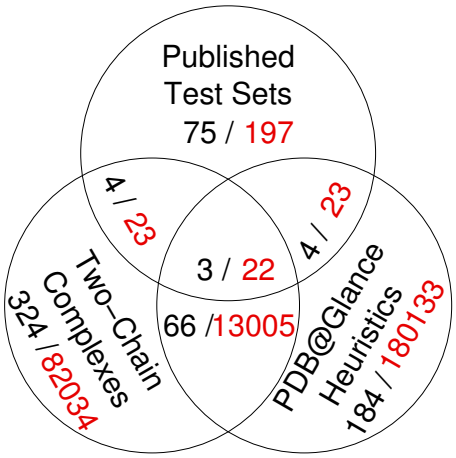



Figure 4.5.: Intersection of test sets. For each test set or intersection between sets the number of complexes and unbound combination is shown.

The reason for this very small overlap lies in the fact that the published sets deliberately soften some of the requirements for the test cases, as mentioned above. Only 6 of the published complexes in the test sets have their unbound data sequence identical to the bound structure, which is a requirement for the test case generation of the database driven approaches.

Those complexes that are not member of the intersection between the database driven approaches do have more than two chains or chain names not covered by the heuristics. The latter can be corrected by adding more heuristic rules in the developed framework.

The data model of the underlying database has only been described implicitly. The next section will give a detailed view of the schema.

## 4.4. A Database Schema for Complexes and Unbound Test Cases

As part of the overall database modelling a schema for the PDB entries, surface segments and complex–test case associations has been developed. For efficient data storage and flexible query facilities the system uses a relational database. The MYSQL software was chosen as underlying RDBMS[3]. The software excels with its speed and simple administration.

The system modules access the data through several database abstraction layers, such as `Perl::DBI`, ODBC and the C++-API MYSQL++. The backend database can therefore be changed easily to other free or commercial RDBMS servers.

[3] Relational Database Management System

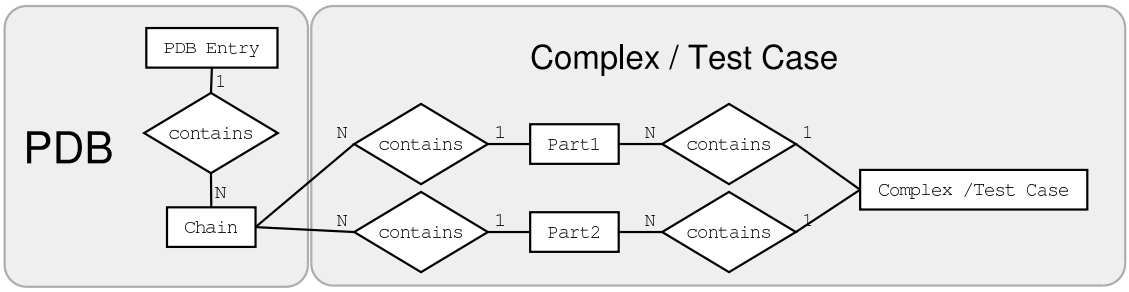



Figure 4.6.: Overview of the database schema that covers Test Case generation. The attributes of the PDB related tables reflect the information stored in the flatfile data.

In figure 4.6 the database schema is given as entity-relationship model diagram. The part of the schema that directly covers the contents of the PDB is modelled after the 3D INSIGHT database [An98]. The flatfiles that make the PDB contents are parsed according to their record identifier and placed into the appropriate relational table.

An abstraction `Part` has to be inserted between the complexes and individual chains, because even a binary complex might contain three chains, if one of the docking partners is a known dimer, such as the entry 2KAI-AB. In that case `Part1` would contain the dimer chains AB, `Part2` the inhibitor.

The tables carrying test set information about the biologically active units and resulting complexes are populated from published data sets and the heuristics from section 4.2 that determine complexes and unbound counterparts.

## 4.5. Creating Synthetic Complexes

Once the unbound conformations have been identified as test cases, the question arises how good the results from a docking tool can be in the optimal case. This also characterises the “difficulty” of the test case. A validation process needs to compare the hypotheses to the best achievable result.

If conformational changes occur during the docking process a rigid-body docking algorithm cannot produce a hypothesis with an RMSD of 0.0 Å. Since neither side chain nor backbone angles are modified, the superposition of the unbound proteins onto the complex optimises for a minimal RMSD. The result is also called a “mock complex”, the term has been formed in [Halperin02]. An example is shown in figure 4.7.

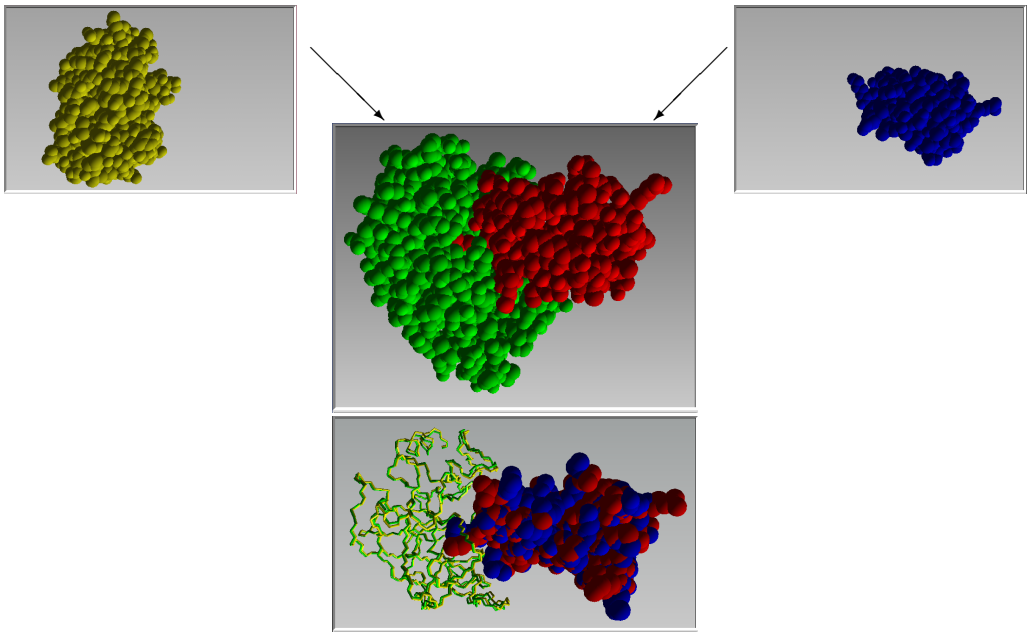


Figure 4.7.: Producing mock complexes. The isolated proteins (yellow/blue) are mapped onto the complexed conformation (green/red). Differences between the real and mock complex are shown using wireframe (enzyme) and spacefill (inhibitor) visualisation.

This structural alignment will – by definition – return the best solution achievable by a rigid body docking algorithm with regard to the RMSD. Rotamer changes or steric clashes will not be modelled and would need full flexibility in the docking process.

The bijection is a result of a sequence alignment between the crystallographically resolved residues and matching all atoms with the same names therein. A sequence alignment is necessary because residues might be missing from the 3D structure of one of the proteins, where a 1:1 mapping would fail.

| Unbound Chains | | Complex | RMSD/Å |
|---|---|---|---|
| 1SUP_ / 3SSI_ | → | 2SICE+I | 0.63 |
| 1SUP_ / 2CI2_ | → | 2SNIE+I | 0.64 |
| 2PTN_ / 6PTI_ | → | 2PTCE+I | 0.64 |
| 3PTN_ / 6PTI_ | → | 2PTCE+I | 0.68 |
| 1BTY_ / 6PTI_ | → | 2PTCE+I | 0.83 |
| 1AKZ_ / 1UGIA | → | 1UGHE+I | 0.90 |
| 1BRA_ / 6PTI_ | → | 1BRBE+I | 0.91 |
| 1CHG_ / 1HPT_ | → | 1CGIE+I | 1.63 |
| 1TGN_ / 6PTI_ | → | 2TGPZ+I | 2.23 |

Table 4.2.: Unbound RMSD. The unbound chains have been superimposed onto the complex.

Using an RMSD considering only the $C_\alpha$ atoms as described in [Halperin02] would not reveal the differences in side chain placement, and is therefore not meaningful in this case. Instead, the RMSD needs to be computed across all heavy atoms $C, N$ and $O$. Table 4.2 shows the range of RMSD results for the mock complexes on some of the test cases.

# System Design and -Components 5

## 5.1. System Architecture

As described in chapter 3 protein docking is a search problem with a high computational complexity. Starting from the huge search space the irrelevant docking hypotheses have to be eliminated, until a ranking of the remaining solutions is computationally feasible.

The solution is a hierarchical system with increasing specificity at the cost of increasing complexity. A modular software architecture also allows to exchange individual functions to test new algorithms and approaches or improve speed. One way of achieving speed improvements is the parallelisation of the most compute intensive tasks.

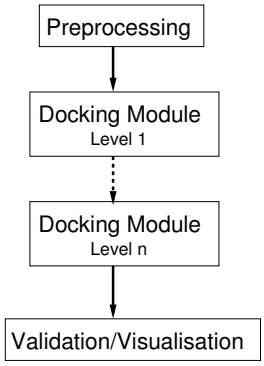



Figure 5.1.: The pipelined ELMAR system architecture. The docking modules can be stacked, they filter hypotheses with increasing specificity.

Most of the modules are implemented in a client/server fashion, providing a stream of (processed

or filtered) data to the downstream modules. An overview over the stages of the system is shown in figure 5.1. The preprocessing step reads protein data, computes a discrete surface representation and calculates physical and chemical features. A stack of docking modules eliminates hypotheses that are implausible with respect to the applied cost function. Finally, the results can be visualised and validated against the correct solution if available.

The core of the system consists of the search stage and the scoring function in the docking modules. The modules will be described in the following section, communication and infrastructure will be considered afterwards. The description of individual modules follows the flow of data as shown in figure 5.1, starting with the preprocessing steps.

### 5.1.1. Preprocessing Protein Data

Preprocessing of protein data starts with the 3D structures taken from the PDB, which is subsequently transformed into a less complex representation. A segmentation step classifies putative docking sites.

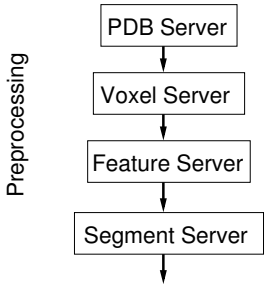



Figure 5.2.: A detailed view of the preprocessing module of figure 5.1. The preprocessing steps are applied to each individual protein chain.

All these steps can be computed on the whole protein data bank in advance. This persistent storage makes the data immediately available if it is needed for an actual docking search.

#### The PDB Server

The module PDB server in figure 5.2 provides the data of individual PDB entries. The PDB content is mirrored as a set of individual flat-files from one of the official sites such as `www.rcsb.org` or in Europe `pdb.ccdc.cam.ac.uk/pdb/`. The mirroring is either done periodically or on demand using the PERL program `mirror`.

The PDB server opens the requested files stored in the local repository and provides them as demand stream (which will be explained in section 5.3.2) to one or several clients. This mechanism

works across different computer networks, even if they are within different administrative domains[1]. This would not be possible using normal networked file systems, such as NFS, and avoids data replication with the associated data redundancy.

### Voxel and Feature Extraction

The natural representation of a protein conformation is the set of atoms, including their position, bonds and associated charges as stored in the PDB. For fast protein docking this representation is too complex because of the large number of calculations[2] that are needed to compute e.g. steric clashes. A simpler representation is therefore required. M. Connolly described in [Connolly83] the determination of the solvent accessible surface by rolling a "probe sphere" with a diameter of 1.4 Å (the approximate size of a water molecule) along the surface. Narrow cavities, though on the surface, are thus not solvent accessible and cannot be accessed by a ligand.

In the first preprocessing stage the protein surface is discretised into an equidistant, 3D voxel model. An example is shown in figure 5.3. The underlying physico-chemical properties such as hydrophobicity and charge are attached to the surface as components of a feature vector. Their values are extracted from the parameters of the AMBER-forcefield developed by Weiner et al. [Weiner86].

By converting the protein data into a regularly spaced 3D model the representation is computationally easier to handle. A surface point is reachable through its index in the grid, expressed as an integer triplet. Via the lattice constants a point in space directly addresses a grid index. The cross correlation, which will be described later, can be done in discrete fourier space, reducing the complexity even further.

### Segment Server

The full 3 dimensional search space of all possible translations is too large to be sampled completely by the docking modules. Therefore the surface is segmented into potential active sites, which are used as seed points for the docking modules.

The segment server processes a PDB entry together with the voxel representation of the surface. The convex hull of the surface points is computed, covering the whole protein. The convex hull is a triangulation of the complete set of surface voxels such that no triangle is placed in a "valley". The spat product $c$ of a facets' three neighbours' surface normals $\vec{n}$ is positive: $c = (\vec{n}_1 \times \vec{n}_2) \cdot \vec{n}_3 \geq 0$. The algorithm `qhull` by Barber et al [Barber96] is used.

[1] This can be the case if CPU time is provided by different faculties or even external data centers.

[2] Especially float point arithmetics and trigonometric functions, which are slower than integer calculations on general purpose hardware.

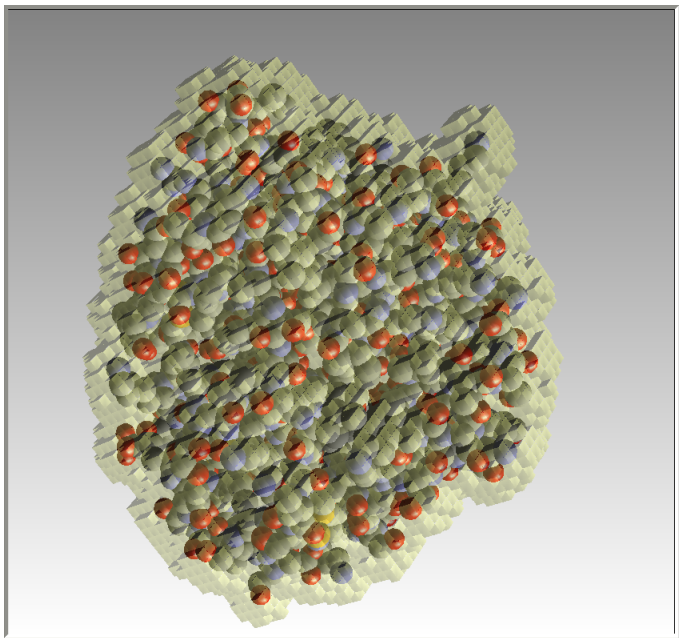


Figure 5.3.: Voxel representation of 1TGSZ chymotrypsin. The solvent accessible surface grid representation is overlaid in light shade.

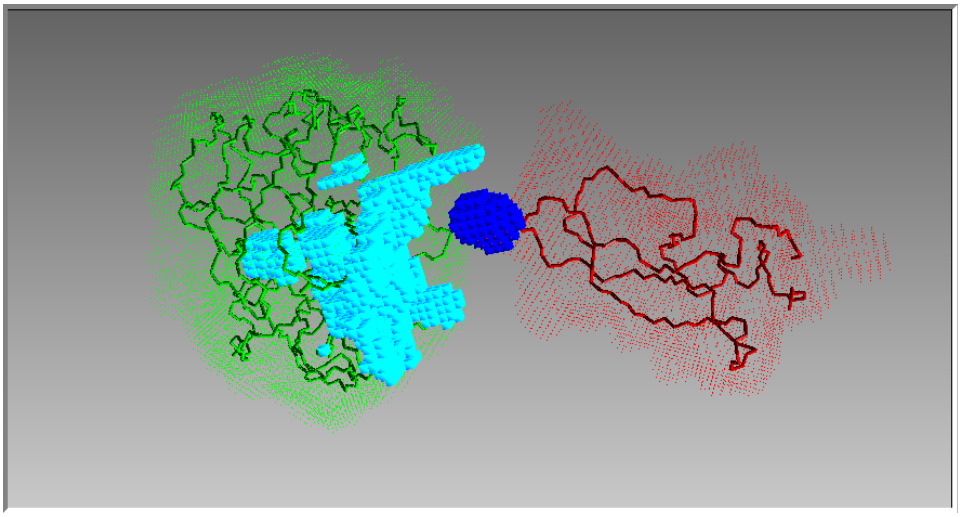


Figure 5.4.: Backbone of 2PTC and inhibitor, the surface is overlaid as dots. The surface regions matching the real contact site best are shown in blue shades.

A facet is convex if the spat product $c$ is large and the size is small, which means it covers surface with high local curvature. Similarly, large facets in an almost planar neighbourhood ($c$ is small) cover concave parts of the surface.

In the following docking steps two of these segments (one from each docking partner) are paired. A naive approach would be to pair each segment from the first protein with all segments from the second. Since two convex segments cannot form a large contact area, only convex-concave segments need to be paired. Additionally, concave-concave areas form a saddle shaped contact, and hence are also included. An example of two paired surface segments is shown in figure 5.4.

### 5.1.2. Docking Modules

During a docking run the list of initial hypotheses is created and passed to the next module(s). The larger of the two proteins is held fixed, while the position of the other molecule is modified with a translation $\vec{t} = (t_1, t_2, t_3)^T$ and a rotation specified in eulerian coordinates[3] $\vec{r} = (\phi, \theta, \psi)^T$. A single complete hypotheses is described by $\vec{h} = (\vec{t}, \vec{r}, \vec{d})^T$, where $\vec{d}$ is the associated meta data, including results of the scoring function.

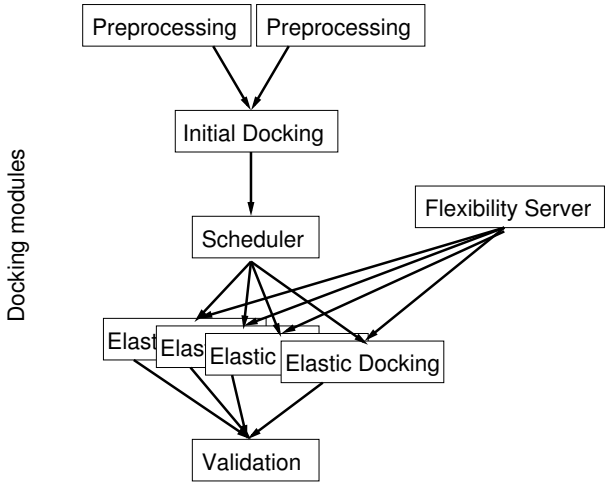



Figure 5.5.: Two stacked docking modules and the associated infrastructure: the scheduler and an external provider for flexibility data.

The next sections will describe the docking modules used in the ELMAR system.

[3] The eulerian rotation is widely used in classical mechanics [Goldstein89]. The body is rotated three consecutive times, each time around one of the axis of the (intermediate) coordinate system. Several small variations (left/right handed coordinate system or different nomenclature/order of rotations) between british and continental literature cause occasional disturbance.

### 5.1.3. Initial Docking

Once the putative active sites have been extracted, they can be paired to form the initial docking hypotheses. A pairing between two convex surface segments is not necessary, because a sufficiently large geometric complementarity cannot be achieved. The geometry of docking sites can either be convex↔ concave as in the classical lock-and-key model, or concave↔ concave for saddle-shaped contacts.

Statistics on cocrystallised contact sites have determined cut-off values that can be applied to the size of these regions and their hydrophobicity. Surface regions that do not match the cut-off criteria can be pruned from the list of hypotheses early. In the later stages each surface region is paired with approximately ten other segments, and each pair is sampled at five different angles. Each region therefore participates in roughly 50 hypotheses.

In a crystallised complex both contact sites of the molecules share the same surface points, therefore the principal axes of the surface points are parallel. The eigenvector analysis of the two candidate regions reveals a transformation matrix that determines how the coordinate system of one surface region (and the whole protein) has to be transformed to align the principal axes to obtain the corresponding complex conformation. The translation vector moves the mobile protein such that the center of both contact sites are at the same position.

### 5.1.4. Elastic Docking and Scoring

The previous module created a list of preliminary docking hypotheses $\vec{h} = (\vec{t}, \vec{r})^T$. The scoring function $e(\vec{h})$ has to sort this list of hypotheses according to the expected fitness of the resulting complex.

The binding forces introduced in section 2.1.2 are short range forces. Since no detailed energy calculations can be performed for all hypotheses in reasonable time, several approximations are used. During the preprocessing steps the protein surface is labelled with the physico-chemical features such as hydrophobicity and electrostatics. The combination of those features reflects the interaction forces introduced in section 2.1.2:

**Geometry and Complementarity** The term is larger the larger the binding site is, i.e. the more surface points are matched between the docking partners. This reflects the van-der-Waals attraction forces between atoms. Steric clash (overlap) is penalised with the empirically derived parameter $\rho = -9$, reflecting the repulsive forces at short distances.

$$
\begin{aligned}
P_1 \bullet P_2(i,j,k) &= \sum_{i',j',k'} P_1(i',j',k') \cdot P_2(i+i', j+j', k+k') \cdot El(i',j',k') \qquad (5.1)\\
P_{1(i,j,k)} &= \begin{cases} 1, & (i,j,k) \in \text{Protein 1} \\ 0 & else \end{cases} \\
P_{2(i,j,k)} &= \begin{cases} 1, & (i,j,k) \in \text{Surface of Protein 2} \\ -\rho, & (i,j,k) \in \text{Interior of Protein 2} \\ 0 & else \end{cases} \\
\text{with} \qquad & El(i,j,k) = \text{Elasticity at grid index} \quad i,j,k
\end{aligned}
$$

The parameter $El$ describes the elasticity assigned to an amino acid, it is used to decrease the effect of the penalty term $-\rho$. Elasticity and determining $El$ will be described in detail in the next section.

**Hydrophobicity** The correlation of hydrophobic surface patches corresponds to the effect of an entropy increase if the solvent is expelled from the active site. Hydrophobic surface patches force the solvent to build a hydrogen cage, where the solvent builds a network of hydrogen bonds. Hydrophilic surfaces allow bonds between the solvent and the solute, increasing the system's degrees of freedom. This component of the scoring function favours patches with a hydrophobicity $h$ larger than average $\bar{h}$. The convolution $H_1 \bullet H_2(i,j,k)$ is defined using:

$$
\begin{aligned}
H_1 \bullet H_2(i,j,k) &= \sum_{i',j',k'} H_1(i',j',k') \cdot H_2(i+i', j+j', k+k') \qquad (5.2)\\
H_{1(i,j,k)} &= \begin{cases} 1+h_1, & h_1 > \bar{h} \ \wedge \ (i,j,k) \in \text{Surface of Protein 1} \\ 1, & h_1 \leq \bar{h} \ \wedge \ (i,j,k) \in \text{Surface of Protein 1} \\ 0 & else \end{cases} \\
H_{2(i,j,k)} &= \begin{cases} 1+h_2, & h_2 > \bar{h} \ \wedge \ (i,j,k) \in \text{Surface of Protein 2} \\ 1, & h_2 \leq \bar{h} \ \wedge \ (i,j,k) \in \text{Surface of Protein 2} \\ 0 & else \end{cases}
\end{aligned}
$$

**Charge** Charges are assigned to the surface according to empirical values $q_1, q_2$ from the AMBER force field and multiplied. Opposite charges are favourable, so the whole term is negated the charge $Q_2$ are negated. Analogous to $P_1 \bullet P_2(i,j,k)$ the correlation $Q_1 \bullet Q_2(i,j,k)$ is defined as:

$$
\begin{aligned}
Q_1 \bullet Q_2(i,j,k) &= \sum_{i',j',k'} Q_1(i',j',k') \cdot Q_2(i+i', j+j', k+k') \\
Q_{1(i,j,k)} &= \begin{cases} q_1, & (i,j,k) \in \text{Surface of Protein 1} \\ 0 & else \end{cases} \\
Q_{2(i,j,k)} &= \begin{cases} -q_2, & (i,j,k) \in \text{Surface of Protein 2} \\ 0 & else \end{cases}
\end{aligned} \tag{5.3}
$$

The computation of these functions is performed as a convolution between the fourier-transformed voxel representations of the grids, reduced to the volume of a search cube. The search cube consists of the smallest possible volume which contains the whole (hypothetical) contact site of both molecules to reduce runtime even further.

The terms of the scoring function are combined into the final score in equation 5.4. The weights $\alpha, \beta \in [0..1]$ have been optimised using an exhaustive parameter search. The convex combination ensures that the weights of these components are bound by the triangle shown in figure 5.6. Each corner corresponds to using a single component exclusively. Every point within the triangle can be uniquely identified using the parameters $\alpha$ and $\beta$.

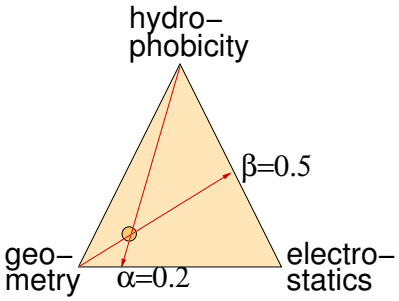



Figure 5.6.: Convex combination of features. The combination of $\alpha$ and $\beta$ defines the weights of individual feature components at their intersection. The corners of the triangle resemble exclusive influence of one of the components.

$$
\begin{aligned}
\text{score}(i,j,k) &= (1-\alpha)(1-\beta) \cdot (P_1 \bullet P_2)(i,j,k) \\
&+ \alpha(1-\beta) \cdot (H_1 \bullet H_2)(i,j,k) \\
&+ \beta \cdot (Q_1 \bullet Q_2)(i,j,k)
\end{aligned} \tag{5.4}
$$

The rotation angles have to be set before the cross correlation, sampled at e.g. 30°. The translation corresponding[4] to the grid index $(i, j, k)$ that satisfies

$$\underset{i,j,k}{argmax} \quad \text{score}(i, j, k) \tag{5.5}$$

has the highest score. The hypotheses and their scores are held in a sorted list.

The values of $\alpha$ and $\beta$ have to be trained on a training set of complexes for optimal classification results, they might not be optimal for a new and different query protein. If the user is very familiar with the docking algorithm, she can tweak the parameters for the particular class of protein in question, but usually an easier user interface is needed to eliminate artefacts of the scoring function related to the weights $\alpha$ and $\beta$. Another method to obtain $\alpha$ and $\beta$ using relevance feedback is currently being developed [Zöllner03]: a set of hypotheses is presented in a 3D GUI application. The plausibility of a subset of hypotheses is assessed by the user, applying marks ranging from "good", "neutral" to "irrelevant". According to the user's feedback the weights are modified such that hypotheses with a good mark are then ranked at the top of the result list.

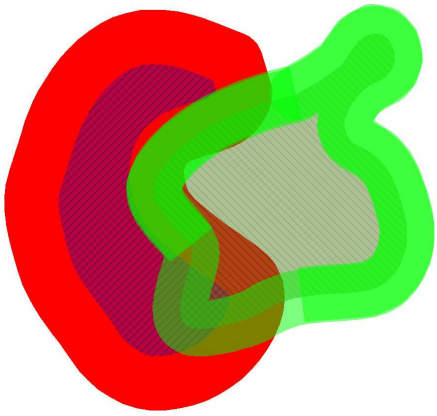


Figure 5.7.: Geometric fit with flexibility information; the unbound conformation of the ligand produces steric overlap. The lighter shade represents a decreased weight of the penalty.

### 5.1.5. Flexibility Calculations

The flexibility server is an interface to a set of various methods providing information about conformational flexibility of a given amino acid.

[4] Grid indices and $x, y, z$ coordinates are transformed using the lattice constant $l$ of the grid: $x = i \cdot l_x$, etc.

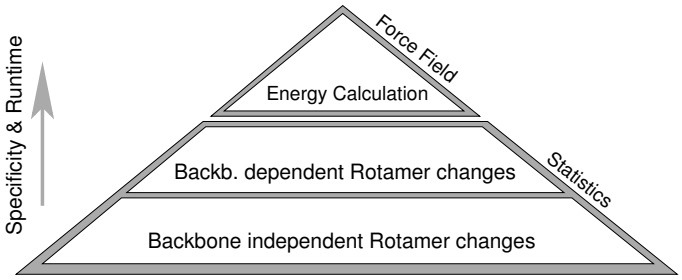



Figure 5.8.: Computation of flexibility. The rotamer change statistics are built on top of rotamer libraries, including the backbone conformation increases the specificity of the flexibility measure. Force Field calculations provide even more specific predictions of side chain conformational changes.

In the ELMAR system it is realised using a MYSQL database, hosted on a server along the PDB-data as will be described in section 5.3.1. Flexibility information can be requested at multiple levels of granularity, based on the amino acid type alone or a more selective tuple with a context sensitive combination of type, secondary structure or rotamericity [Koch02a].

The flexibility values are pre-computed on a training set of the PDB and instantly available for the docking run. The more dependent variables are incorporated, the more sensitive are the statistics with respect to small data sets (see section 2.4).

Without any knowledge about the context of a residue in a certain protein only the overall flexibility of the amino acid can be given. Amino acids like Arginine, Serine, Lysine and Glutamine have been shown to change their rotamer in 20-30% of all cases during complex formation, whereas others change in only 5%. The flexibility can be attributed to the size and polarity of the side chains.

Using the context of a residue in a protein structure (including its actual conformation) additional features can be considered. An increased probability for side chain rotamer changes has been shown for residues with a higher rotamericity, i.e. $\chi$-angles close to the boundary of the rotamer. The side chain conformation (or more generally the secondary structure $\alpha$-helix or $\beta$-sheet) decrease flexibility, whereas loop regions have a larger flexibility. Surface residues have also been shown to be flexible above average. For details see [Koch03].

As a flexibility measure for the dynamisation of the steric clash the probability of a rotamer change $p(\text{res})$ is used, where $p$ has been precomputed for amino acids of type “res” in the same context as the current residue “res”. To be used as weight the probabilities are scaled (see equation 5.6) in the range between $1 \pm \frac{\sigma}{2}$. For $\sigma = 0$ no elasticity weights are applied.

$$El = \left(1 - \frac{\sigma}{2}\right) + \sigma\left(1 - \frac{p - \min(p)}{\max(p) - \min(p)}\right) \tag{5.6}$$

The second source of flexibility, the energy based classification scheme, provides precalculated flexibility scores for all residues in the test set, also stored in the database [Zöllner02]. Scaling of these values is done as in equation 5.6, where $p = 1$ for residues predicted to change, and $p = 0$ otherwise. The simulation has to be started prior to the docking run if the query proteins were not already part of the test set.

Energy calculation is most specific to the protein in question, on the other hand the runtime requirements for the preparation increase, as figure 5.8 indicates. A SQL-query selects and combines those parameters from the database server.

During the development the system has to be evaluated and compared to other docking systems on a set of known (and solved) docking cases. The criteria will be explained in the next section.

## 5.2. Validation

Software continuously evolves during its life-cycle, competing systems are developed and with the growth of the PDB new test sets become available. A validation setup is therefore an integral part of the ELMAR system.

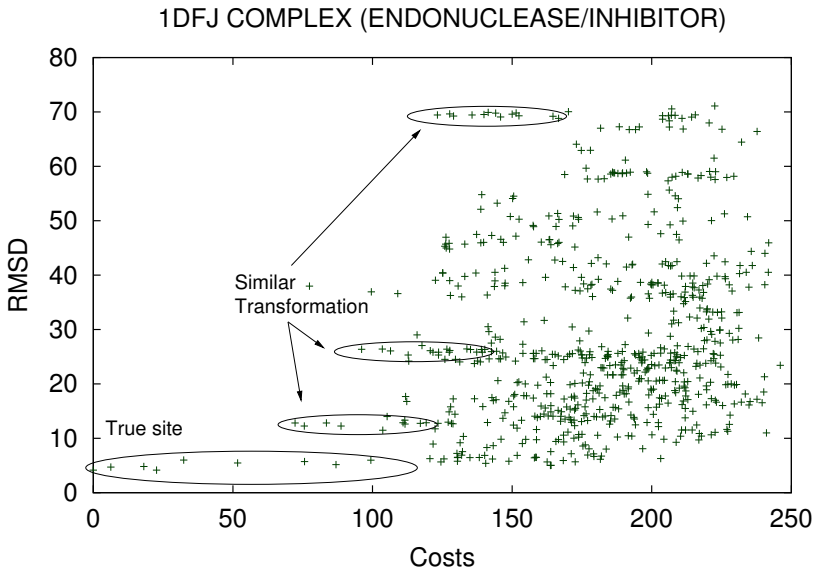
In the bound case validation takes place between a docking hypothesis and the complex that provided the individual chains as true solution. In the unbound (or semi-unbound) case the mock complex, constructed from the unbound chains and the true complex via structural alignment (see section 4.5) is used for verification.

Since several measures exist to assess the accuracy of a docking system, a flexible approach has to be chosen. For each hypothesis the RMSD is computed, and stored within the database. The following sections explain first a detailed representation of results, followed by a method for a summary of a whole docking run.

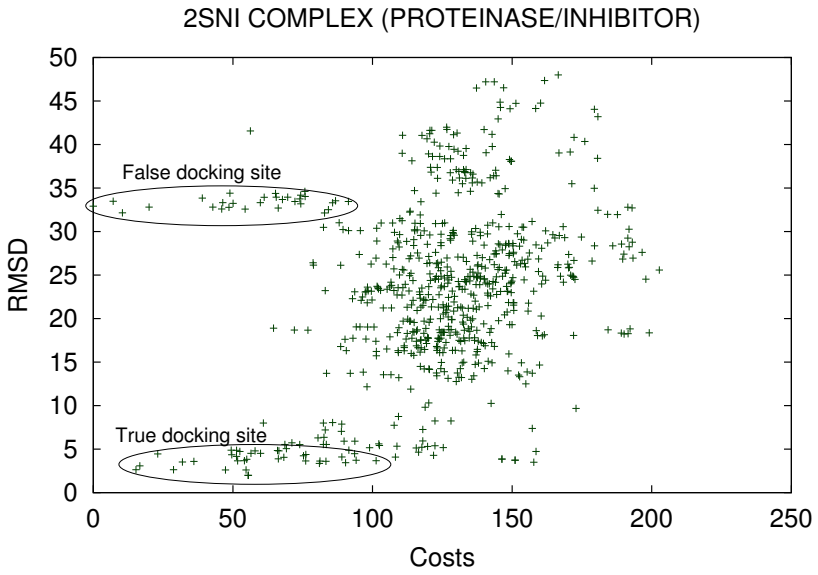
### 5.2.1. Graphical Representation of Docking Results

In a graphical representation of a docking run the RMSD for all hypotheses is plotted against the estimated score. Some examples are shown in figure 5.9.

Plotting the hypotheses against the score gives an overview over sensitivity and selectivity of the scoring function. For better readability the $x$-axis is plotted as $x = max(score) - score$, so that high scores with a low RMSD (i.e. good hypotheses) appear near the origin at $(0,0)$. Several other features can also be detected in the plots, e.g. the RMSD of the best ranked (leftmost) hypothesis, or the minimum RMSD that had been found. Sets of data points with an almost equal RMSD (encircled in figure 5.9) are likely to stem from very similar transformation of the second protein, and their scores are an indication how good the “fit” is in a larger neighbourhood of the parameters.




(a) Precise scoring, IPI=53.9, MinRMSD=1.1 Å, N100=52.




(b) One true, one false positive docking site: IPI=-2.5, MinRMSD=1.48 Å, N100=4

Figure 5.9.: Sample cost vs. error plots. Each dot represents a single hypothesis. Correct hypotheses have a low RMSD at low costs, they are located in the lower left corner.

### 5.2.2. Performance Indicators

From result plots like those in figure 5.9 the performance can be visually observed: is there a clear correlation between the costs and RMSD, are there several good constellations, and if so, are they rated consistently with regard to their score ? False positives can be identified easily, but there is no single number that can be assigned as quantitative value to the results of a docking run.

The DRUF approach (Docking Results unified Format) suggests several numbers for performance evaluation [Halperin02, Appendix A]. For unbound docking the superposition of the unbound receptor's $C_\alpha$ onto complexed $C_\alpha$ are used, and the RMSD of docking partner $C_\alpha$ is computed.

The performance of the search stage is determined by the smallest RMSD in the result set. The finer the sampling of the search space, the better the solution can be, at the expense of runtime needs. If a hierarchical approach is used for pruning the search space, this shows whether correct constellations are filtered out. These are the false negatives.

The performance of the scoring stage alone can be assessed if the correct solution – the crystallographed complex – is included in the result set. The scoring function should position this solution with RMSD=0 Å in a top position, giving a high ComplexRank.

DRUF suggests the following performance indicators to assess the system's overall performance:

- RMSD of solution at Rank #1
- Rank of first solution with <5 Å
- Rank of best RMSD hypothesis
- *N10:* # of solutions <3 Å with rank < 10
- *N50:* # of solutions <4 Å with rank < 50
- *N100:* # of solutions <5 Å with rank < 100

They can be extracted easily from the hypotheses database using appropriate SQL-statements[5]. Some examples are shown in figure 5.10.

These values use the absolute RMSD value and are comparable only between the same proteins. They do not incorporate the "difficulty" of a given docking task. The difficulty is closely related to the amount of structural flexibility upon docking and can be estimated using the *unbound* RMSD, as shown in table 4.2.

[Halperin02] also uses the difference between the unbound RMSD and lowest RMSD in the result set as normalising factor to obtain relative RMSD data. The difference represents the inaccuracy introduced by the docking system.

The handling of side chain flexibility is missing in DRUF. A docking suite that provides side chain placement does not benefit from correctly predicted rotamer changes. Multiple hypotheses with different side chain conformations cannot be compared using side chain $C_\alpha$ RMSD.

[5] Systems without sub-select statements need to use temporary tables in case of the "Rank of best RMSD hypothesis".

```
select Entry, rmsd from Hypothesis where rank=1;
select Entry, min(rank) from Hypothesis
       where rmsd<5 group by Entry;

select Entry, rank from Hypothesis
       where rmsd = (select min(rmsd) from Hypothesis as mins
                            where Hypothesis.Entry=mins.Entry
                            group by mins.Entry);

select Entry, count(*) from Hypothesis
       where rmsd<[3|4|5] and rank<[10|50|100] group by Entry;
```

Figure 5.10.: SQL-queries compute performance on whole test sets. From top: RMSD of solution at Rank #1, Rank of first solution with <5 Å, Rank of best RMSD hypothesis, N10, N50 and N100.

Also desirable would be a performance value in terms of CPU time and/or memory requirements. For relational database systems the Transaction Processing Performance Council (TPC) publishes defined benchmark data sets and collects absolute performance or performance-per-cost results.

A performance indicator independent from RMSD calculations is the residue contact count: the set of interacting residues (within a cut-off distance from each other) in the contact site is counted and compared to the hypothesis. Thus, the test is constraint to the atoms in the (hypothetical) binding site, but it cannot distinguish between two hypotheses that both miss the active site altogether[6]. The $C_\alpha$-distance count similarly checks for matching number of $C_\alpha$ pairs within a cut-off radius.

All the above measures extract single data points (N10, best RMSD, etc.) from the overall set, their expression power is not close to the full plots introduced earlier. An integrated performance indicator has been developed, which summarises the whole set of hypotheses.

### 5.2.3. Integrated Performance Indicator

A set of simple and meaningful performance indicators should give hints to the sensitivity and selectivity of the system. For binary decisions, this is usually done by evaluating the number of true/false positive and negative classifications.

For continuous values like RMSD and costs binary decisions are not possible without setting threshold values. As a threshold for a correct docking hypothesis an RMSD of less than 5 Å can

[6] By deliberately allowing artificially high steric clash (and moving more residues closer together), this measure can be mislead to indicate a good fit.

be accepted. For plots such as 5.9 there are no binary decisions, and a more elaborate approach has to be taken.

Ideally, the costs and error measures should be linearly dependent on each other. In reality this is not possible, since there are many (theoretically an unlimited number of) docking constellations with the same RMSD error on different parts of the surface, and thus the components of the cost function will rate the respective surface properties and come up with different estimates. In close vicinity of the active site, however, the scoring considers the same surface patches for different hypotheses.

If true docking positions are searched for, the scoring should have a high selectivity with as few false positives as possible. The exact ranking of (true or false) negatives is less important. Instead of absolute values the results should be comparable between different query proteins, to be usable in a 1:N docking scenario. A comprehensive quality measurement includes:

- RMSD of a hypothesis (the smaller the better). Figure 5.11(a) shows that conformations with an RMSD below 10 Å receive a good score.
- the rank of the hypotheses: the lower, the less important are outliers. Figure 5.11(b) shows the weighting across all results.
- few false positives (with low costs but high RMSD, the upper left area in figure 5.11(c)).

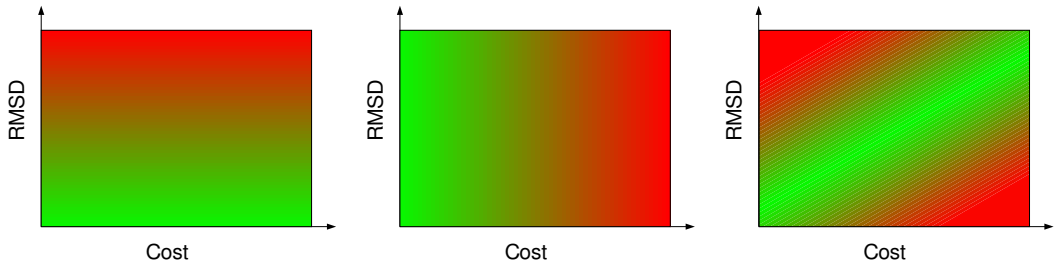
The Integrated Performance Indicator IPI summarises the performance of a docking run as a weighted sum of the scores $score_i$ across all hypotheses $i$. The constants $p_a$ and $p_b$ assess false positives by measuring the hypotheses above ($p_a$) and below ($p_b$) the diagonal of the plots.

$$\begin{aligned} \text{IPI} \quad = \quad & \sum_i \frac{Rank_{max} - Rank_i}{Rank_{max}} \cdot \frac{max(10\text{Å} - \text{RMSD}_i, 0)}{10\text{Å}} \\ & + \begin{cases} p_a & if \quad \frac{\text{RMSD}_i}{score_i} > \frac{\text{RMSD}_{max}}{score_{max}} \\ p_b & else \end{cases} \end{aligned} \tag{5.7}$$

Suitable values are $p_a = -0.025$ and $p_b = 0.05$. The transcription of equation 5.7 into a SQL-query is shown in figure 5.12. Examples for this measurement are given in figure 5.9 and throughout this chapter..

The preprocessing and docking modules are implemented as small command-line tools. Programs without graphical user interfaces can run on headless machines[7]. The next section deals with communication and infrastructure used to integrate the modules.

[7] I.e. compute clusters or a BOW (Bunch of workstations). GUI systems should always be designed such that the user interface is decoupled from the actual simulation, and both can run on separate machines.




(a) RMSD between 0-10 Å normalised to $[0..1]$.

(b) Rank normalised to the interval $[0..1]$.

(c) Exclusion of false positives.

Figure 5.11.: The components of the integrated performance indicator. Hypotheses that fall into the green area contribute to a good score.

```
select Test_Case_Id,
       sum( (maxrank-rank)/maxrank
           * greatest( (10-rmsd)/10, 0 )
           + if( rmsd/score < maxrmsd/maxscore, 0.05, -0.025)) as IPI,
from Hypothesis
group by Test_Case_Id
```

Figure 5.12.: Calculation of the integrated performance indicator using an SQL-statement.

## 5.3. Communication and Infrastructure

In addition to the ASCII flat files that are used by the BIOWEPRO-Software, the ELMAR modules use network communication and a relational database. This section describes the “glue” that is used to pass information between the modules and that provides persistent storage.

### 5.3.1. Database Integration

Closed[8] software can be limited to internal data formats and storage methods. An open, flexible software system needs to interact with other programs and read/write data in portable ways. Though XML (the eXtensible Markup Language) promises these properties, interoperability between diverse programs is still not ubiquitous.

A database system decouples data storage from the applications accessing the data. It provides access control and parallel query processing. Data access is independent of programming languages or operating systems. SQL offers flexible evaluation of results under different conditions, see figure 5.13. The diagram shows three different kinds of client applications: time critical applications such as the scoring of hypotheses are written in C++, the data is accessed using the MYSQL++-API. These clients create or consume the raw data records. An intermediate adapter layer (ODBC) allows to use standard office or statistics packages (such as SPSS, R or S) to access the raw data for visualisation, aggregation and to create chart diagrams. With a closed format input converters (written for each application) would be necessary to import the data. Alternatively scripting languages such as PERL or the Unix shell can be used for rapid application development (RAD) to aggregate and evaluate the data.

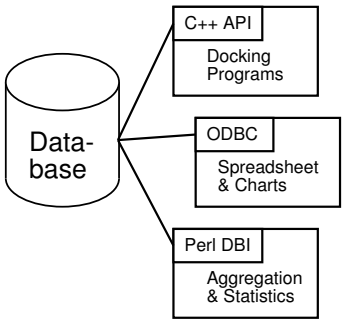



Figure 5.13.: A Database provides data storage for applications.

[8] This term refers to programs that have a minimal dependency on external modules or data. No interoperability is required and I/O functionality can be buried within the software.

As underlying RDBMS the MYSQL[9] database was chosen because of its ease of use, speed and reliability.

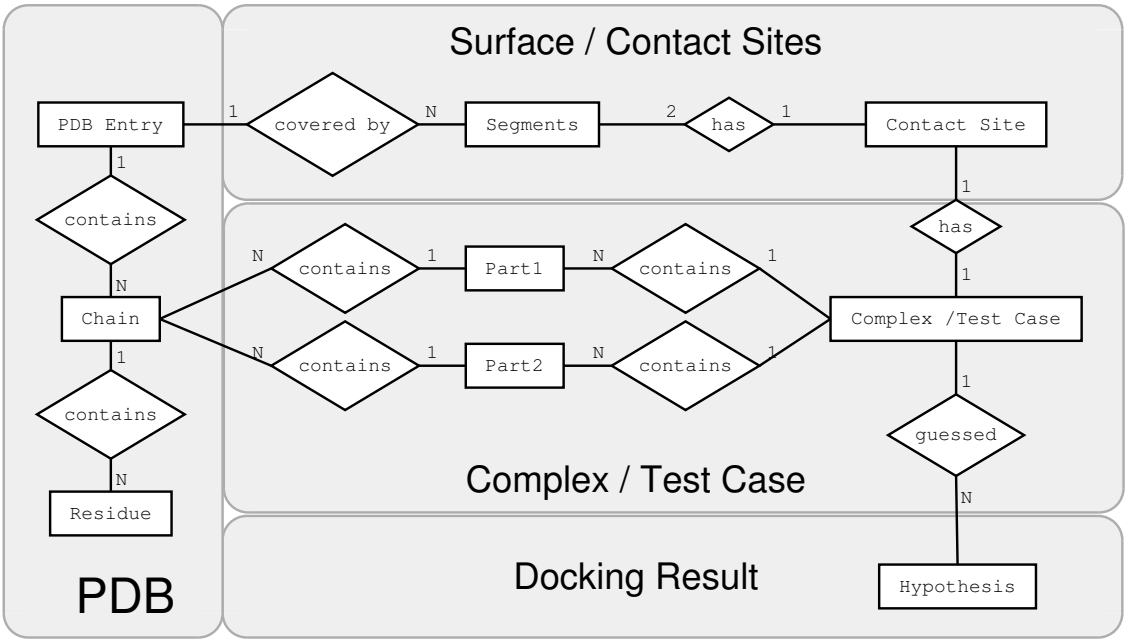



Figure 5.14.: The complete Entity-Relationship database schema, including intermediate results of preprocessing of protein surfaces.

The complete database schema (cf. figure 5.14) covers four domains and extends the one introduced in section 4.4. First, the content of the PDB is represented. The table `Protein` contains name, compound, organism, resolution and other meta data on the protein and the experiment found in the header and `remark` fields of the PDB. The amino acid sequence of the protein chains contained in a PDB entry (used to search for identical proteins in different conformations) is stored in the `Chain` table. Finally, the `Residue` table carries the conformation of the residues including the elasticity calculated.

A second set of tables contains the protein surface, segmented into areas of similar geometric properties, classified as convex or concave. Contact sites are modelled as two surface segments, one from each of the two interacting proteins. The lists of hypotheses including the scores of the individual components of the scoring function are stored in their own table `Hypothesis` and can be visualised or used for further classification, as in section 6.3.

[9] MYSQL has often been criticised for its incomplete implementation on database features. The current version (4.0.13) supports transactions, referential integrity and sub-selects.

The database has a very positive effect on ad-hoc evaluation of the data sets and interoperability between programs implemented in different languages.

### 5.3.2. Network Streams

The system heavily uses the C++-concept of IOSTREAM objects [Stroustrup98] for input/output and for (intermediate) data storage. Buffered streams (see figure 5.15) provide a sliding window into the stored data.

The buffer-object supplies pointers to the beginning/end of the buffer and the current position. If read/write requests can be executed within the buffer area the data is read or written to the current position and the positional pointer is updated. If a read buffer is empty, the `underflow()`-method of the buffer object is called, requesting the buffer to be filled again. Respectively, a full write buffer is written to the final destination if the `overflow()`-method is called.

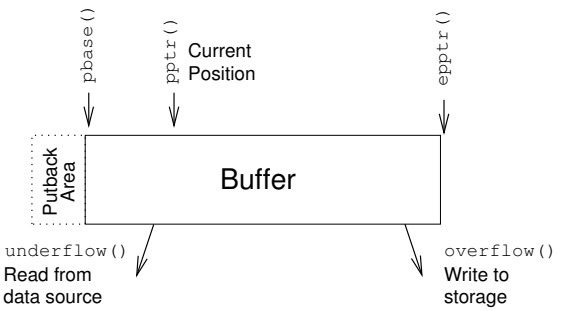



Figure 5.15.: Buffered IOSTREAM. Pointers denote the beginning/end of the buffer area, `pptr()` is the current position. The putback area is optional.

Using this mechanism the stream-oriented methods are completely decoupled from the low-level read/write operations. The required methods the `streambuf`-object has to provide are sketched in listing 5.16. This framework has been coupled to a distributed communication system.

The Distributed Application Communication System (DACS) [Fink95; Fink96] consists of a C-library and demon processes. The library provides a minimal API to C-programs. Users of the system usually do not have to interact with the DACS-demons or naming services.

An application provides resources under a logically unique name. Network (or if possible shared memory) communication is handled by the library and the DACS-demons, see figure fig:benutzersicht. For the ELMAR system three different communication concepts of the DACS have been incorporated into the IOSTREAM framework:

```
class dsstreambuf : public streambuf {
public:
  dsstreambuf();
  dsstreambuf* open( const char* sname, int open_mode);
  dsstreambuf* close();
  ~dsstreambuf();

  int is_open();
  int overflow(int c);                // write buffer to storage
  int underflow();                    // read next data
  int sync();
private:
  DACSstatus_t     dstatus;
  // ... data to access DACS demand streams
  char             *buffer;           // data buffer
};
```

```
dsstreambuf::dsstreambuf() : opened(0) { buffer = malloc(bufferSize); }

dsstreambuf::open( const char* sname, int open_mode) {
  if ( mode & ios::in) {
    dstatus = dacs_order_stream(dacs_entry, dsstream_name);
  } else if ( mode & ios::out) {
    dstatus = dacs_register_stream(dacs_entry, dsstream_name);
  }
  opened = 1;
  return this;
}

int dsstreambuf::is_open() { return opened; }

int dsstreambuf::overflow( int c ) {
  dstatus = dacs_recv(dacs_entry, dstream_name,
                      (NDRfunction_t*)ndr_bytes_marshalling,
                      (void**)&ndr_recv_bytesvector);
  // ... set pointers accordingly ...
  return *gptr();
}

int dsstreambuf::underflow() {
  ndr_bytesvector.data=pbase();
  dstatus = dacs_update_stream(dacs_entry, dsstream_name, &ndr_bytesvector);
  if ( dstatus == D_OK ) {
    setp( pbase(), epptr());
    return c;
  } else {
    return EOF;
  };
}
```

Figure 5.16.: Skeleton of a `streambuf`-object. The `open()` method registers the demand stream in the DACS-system or, for input buffers, subscribes to one. Most error handling code is omitted in this listing for brevity.

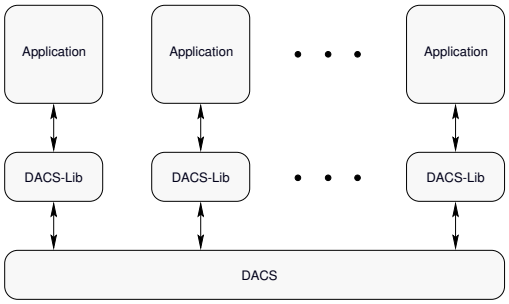



Figure 5.17.: DACS architecture. If the applications are running on several hosts, network communication and byte ordering are handled by the library transparently.

**Function Streams** provide a serialised 1:1 communication. The data source (or function server) waits for requests and sends the result to exactly one recipient. In the ELMAR system this functionality is used by the scoring modules to poll for work packages provided by the schedulers.

**Demand Streams** adopt the mechanism of DACS-streams to the IOSTREAM concept. One stream sender offers the data, multiple receivers can register or subscribe. This 1:N communication is used within the ELMAR system to propagate the 3D protein data.

**Channel Streams** follow the N:1 communication paradigm. A single receiver opens a channel, to which many senders can write data. The module DOCKHARVEST collects the scored hypotheses from several scoring modules.

Figure 5.18 shows their usage in the ELMAR system. Not limited to ELMAR these streams can be used wherever a C++-program expects an `istream` or `ostream` object. In addition a tool `dspipe` has been written, which connects `stdin` or `stdout` to any of the above communication concepts. Additionally, it can bridge between any two of the concepts. This command line tool can therefore be used in shell scripts or for easy testing of parts of the system.

The integration of the DACS framework allows the ELMAR-system to run on multiple hardware architectures on a possibly large number of hosts. The authorisation mechanism underlying the connection between the DACS-demons allows multiple users to connect to the system, but ensures that only registered users gain access. A naming service decouples the actual system configuration from the logical resources it provides, so that system updates and/or reorganisation (of both the underlying hardware and modules of the docking system) is of no concern.

The ELMAR system as described above has been used for bound and unbound docking of different complexes and for pairs of proteins that occur during the X-RAY-crystallography. The results of those experiments will be shown in the next chapter.

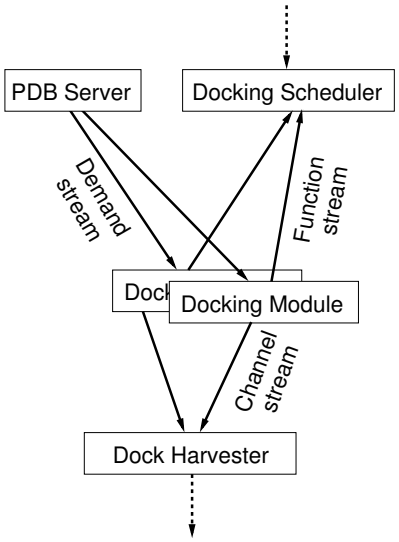



Figure 5.18.: Usage of different kinds of streams for communication. For function streams the arrows symbolise the polling nature of the communication.

# System Evaluation 6

The general term 'system evaluation' covers many aspects, including: "[...] performance studies of computers [...] and distributed systems (hardware), resource allocation and control methods and algorithms, [...] system architecture, design and implementation discussed from a performance viewpoint, case studies and model validations." [Bux03].

In this chapter the composition of the test set is described, followed by an apriori-assessment of the difficulties that can be expected if the unbound structures have to be docked. This is followed by the results of several test runs on selected docking cases, both in terms of runtime, memory consumption and quality of the results. The ability of the scoring function to discriminate between contact sites that occur between dimers and those artificially imposed through the crystallisation process is examined afterwards.

The possible speed-up that can be achieved by parallelisation of the scoring is shown in the last section. The chapter concludes with a discussion of the results, with regard to both the quality of the results, and the applicability to the 1:N docking database search.

## 6.1. Characterisation of the Test Set

The test set used for the docking evaluation has been created using the intersection of the top-down and bottom-up heuristics as described in chapter 4. They cover several enzyme families, the detailed composition is given in table 6.1. The set includes 66 complexes, for which 13005 test cases exist. A random selection of 1020 of them have been used for the unbound docking experiments.

All mock complexes have been generated for the intersection of the top-down and bottom-up heuristics.

| EC Group | Count | Description |
|---|---|---|
| 1.1.x.x | 4 | Oxidoreductases acting on the CH-OH |
| 1.5.x.x | 6 | Oxidoreductases acting on the CH-NH group |
| 2.1.x.x | 3 | Transferases transferring one-carbon groups |
| 2.6.x.x | 6 | Transferases transferring nitrogenous groups |
| 2.7.x.x | 1 | Transferases transferring phosphorus-containing groups |
| 3.1.x.x | 10 | Hydrolases acting on ester bonds |
| 3.2.x.x | 1 | Hydrolases / Glycosylases |
| 3.4.x.x | 30 | Hydrolases / Peptidases |
| 5.2.x.x | 1 | cis-trans-Isomerases |
| 5.3.x.x | 1 | Intramolecular Isomerases |

Table 6.1.: Enzyme Classes in the test set. Several enzyme families are included, with half of the complexes covering the hydrolase family.

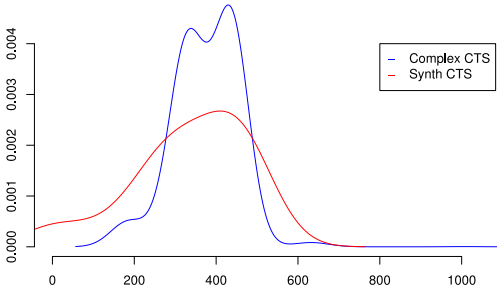
The “difficulty” of a given unbound docking can be assessed using the unbound RMSD, which is the RMSD-distance between the mock complex of unbound docking partners to the actual complex. Some of the values are shown in table 4.2.

Since steric clash poses the largest problem for rigid body docking algorithms, the contact sites of both the complex and the unbound proteins were compared. The number of voxels involved in a steric clash was counted. Each of these overlapping voxels would add $\rho = -9$ as penalty in the scoring function. Similar, the number of interacting surface voxels was counted. They contribute a higher score in the correlation function 5.1 to 5.3.

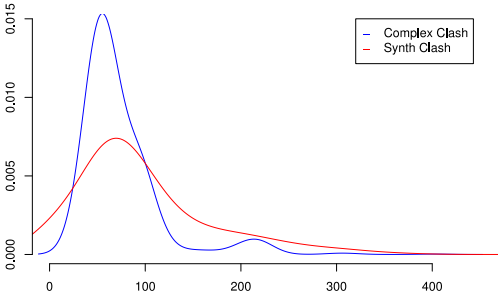
The histograms in figure 6.1 show that the real complexes have a 10% higher score for the contact site, and mock complexes have one third more steric clashes, reducing the geometric score. Still, the histograms have a similar shape such that both cases can share the same scoring parameters (especially $\rho$) in the scoring function. Those cases where the score dropped considerably cannot be assessed without considering full flexibility. The clash volumes between bound and mock complexes as well as their ratio are shown in table 6.2.

However, there are examples where the steric clash is not caused by side chain flexibility. For the unbound complex 1OXP/1AMA (a homo dimer) the steric clash nearly doubles. Figure 6.2 shows that a small twist in the backbone of a 6 residue tail of the chain causes the $N$-terminus to overlap with the globular part of the docking partner. This kind of backbone flexibility has not been included, since the flexibility parameters cannot be estimated in advance, and hand-modelling of hinge-motion would be necessary but is unfeasable for large-scale screenings.

The test set that has been created can be summarised as containing enzymes from a range of different families. It satisfies the strict constraints that have been proposed in chapter 4, and has both simple and difficult complexes.




(a) Contact sites are larger for real complexes (median 716) than synthetic complexes (median 645).




(b) Steric clash overlap are larger for synthetic complexes (median 139) than for real complexes (median 103)

Figure 6.1.: Comparison between cocrystallised complexes and sequence identical synthetic complexes created from unbound conformation. The size of contact site and steric clashes are plotted against the gaussian density function.

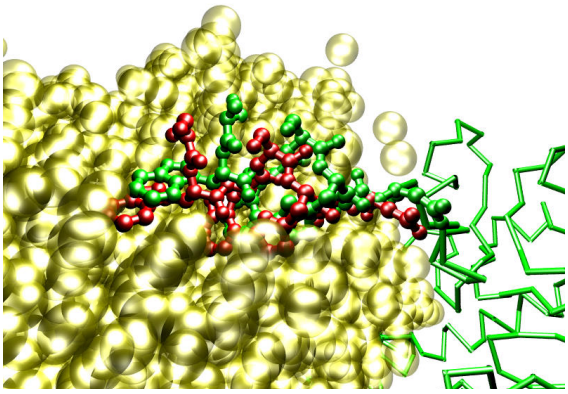


Figure 6.2.: Steric clashes for 1OXP/1AMA→9AAT. The two chains of the complex are coloured yellow/green. The structural superposition of the unbound molecule 1AMA (red) “dives” into the complex, causing the overlap.

| Unbound chains | Complex | Complex Clash | Unbound Clash | Ratio |
|---|---|---|---|---|
| 1G36A/ 1BPI_ | →1TPAE+I | 45 | 43 | 1.0 |
| 7DFR_/1DRH_ | →1JOLA+B | 66 | 66 | 1.0 |
| 9RAT_/1AQP_ | →9RSAA+B | 94 | 100 | 1.1 |
| 9RAT_/1AQP_ | →8RSAA+B | 53 | 59 | 1.1 |
| 1JWRA/ 1JSF_ | →1LZSA+B | 64 | 73 | 1.1 |
| 7DFR_/1DRH_ | →1DYIA+B | 58 | 61 | 1.1 |
| 9RNT_/1BU4_ | →1RGCA+B | 39 | 47 | 1.2 |
| 1OXP_/1AMA_ | →1TARA+B | 143 | 176 | 1.2 |
| 7DFR_/1DRH_ | →1DYHA+B | 61 | 76 | 1.2 |
| 1CQ8A/1AAW_ | →1ASNA+B | 200 | 265 | 1.3 |
| 1POD_/1BBC_ | →1POEA+B | 402 | 513 | 1.3 |
| 1OXP_/1AMA_ | →8AATA+B | 120 | 170 | 1.4 |
| 1NSP_/1NDC_ | →1NDPA+B | 959 | 1305 | 1.4 |
| 1I3HA/ 1C57A | →3ENRA+B | 218 | 323 | 1.5 |
| 1OXP_/1AMA_ | →7AATA+B | 120 | 189 | 1.6 |
| 1OXP_/1AMA_ | →9AATA+B | 124 | 216 | 1.7 |
| 1OHK_/1DRF_ | →1DHFA+B | 166 | 387 | 2.3 |
| 1GCD_/1HPT_ | →1CGIE+I | 53 | 148 | 2.8 |
| 1CQ8A/1AAW_ | →1ASMA+B | 72 | 221 | 3.1 |
| 1OHK_/1DRF_ | →2DHFA+B | 118 | 375 | 3.2 |

Table 6.2.: Voxel counts indicating steric clash between complex partners and the increase in unbound cases.

## 6.2. Experiments and Results

Several docking experiments have been conducted. The bound docking examines how well a given complex can be predicted with the fast scoring function.

### 6.2.1. Bound Docking

In a first step the available complexes are docked in their bound structure. In table 6.3 the results are summarised using the DRuF format. For each complex 700 hypotheses have been kept in a sorted list. The minimum RMSD of all hypotheses in the list is given in the column “Minimum RMSD”. In 19 cases no results with an RMSD below 5 Å have been found among the first 100 hypotheses. In ten of these, such a hypotheses is included with a higher rank, in 9 cases no sensible hypotheses have been found. The latter are those cases, where the actual binding site was not part of the initially segmented surface segments. Additionally, the correct solution (translation=$(0,0,0)^T$ and rotation=$(0,0,0)^T$ have been included in the list of hypotheses. If they score better than the generated hypotheses, their rank is included as the column “Complex Rank”.

| *Entry* | *ComplexRank* | *Min$R_{MSD}$* | *N10* | *N50* | *N100* | *IPI* |
|---|---|---|---|---|---|---|
| 1ASM | 1 | 1.92 | 2 | 4 | 4 | 17.0 |
| 1ASN | 1 | 1.84 | 6 | 24 | 39 | 25.3 |
| 1BRB | 147 | 1.75 | | 3 | 4 | 18.6 |
| 1BRC | 66 | 1.21 | | 22 | 48 | 60.0 |
| 1CBW | | 7.94 | | | | -3.5 |
| 1CGI | 1 | 1.23 | | 29 | 66 | 115.1 |
| 1DFJ | 5 | 1.09 | 7 | 29 | 52 | 53.9 |
| 1DHF | | 4.58 | | | | 20.4 |
| 1DQJ | | 3.43 | | | | 14.3 |
| 1DYH | 692 | 4.80 | | | | 10.8 |
| 1DYI | 607 | 0.97 | | | | 4.4 |
| 1DYJ | 698 | 1.72 | | | | 10.1 |
| 1JOL | 692 | 2.71 | | | | 4.1 |
| 1LZS | | 6.82 | | | | 8.2 |
| 1MLC | | 1.12 | | | | 29.6 |
| 1NDP | | 2.96 | | 2 | 2 | 9.9 |
| 1POE | | 6.03 | | | | -9.7 |

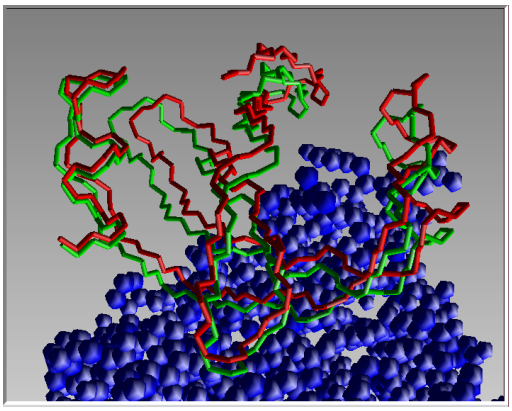
(continued)

| Entry | ComplexRank | $MinR_{MSD}$ | N10 | N50 | N100 | IPI |
|---|---|---|---|---|---|---|
| 1PPE | | 0.82 | 14 | 28 | 28 | 33.7 |
| 1RGC | | 3.76 | | | | 5.7 |
| 1STF | 1 | 5.26 | | | | 17.9 |
| 1TAB | | 3.88 | | | 18 | 26.7 |
| 1TAR | 1 | 5.81 | | | | 5.1 |
| 1TLC | 1 | 3.16 | | 4 | 5 | 24.3 |
| 1TPA | 1 | 2.56 | | 1 | 6 | 32.4 |
| 1UGH | 25 | 1.17 | 8 | 37 | 67 | 119.0 |
| 1WQ1 | 1 | 13.38 | | | | -10.2 |
| 1XZK | | 4.82 | | | | 32.0 |
| 2DHF | 218 | 1.90 | | 2 | 8 | 56.4 |
| 2KAI | 78 | 1.17 | 9 | 25 | 53 | 40.5 |
| 2PCB | | 28.43 | | | | -7.2 |
| 2PCC | 62 | 4.08 | | | | 28.5 |
| 2PTC | 1 | 3.01 | | | 8 | 28.6 |
| 2SIC | 1 | 2.83 | | | 6 | 36.2 |
| 2SNI | 1 | 1.48 | 1 | 2 | 4 | 20.9 |
| 2TEC | 665 | 1.49 | | 14 | 40 | 59.3 |
| 2TGP | 1 | 1.30 | 1 | 22 | 54 | 86.2 |
| 3ENR | 680 | 0.00 | | | 4 | 12.3 |
| 6CHA | 341 | 3.22 | | | 2 | 10.4 |
| 7AAT | 1 | 3.03 | | 4 | 12 | 9.7 |
| 8AAT | 1 | 1.65 | 9 | 12 | 12 | 7.4 |
| 8RSA | 1 | 10.44 | | | | -5.8 |
| 9RSA | 416 | 3.89 | | | | 5.9 |

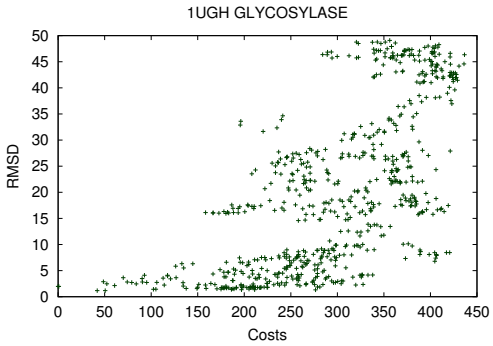
Table 6.3.: IPI and DRuF results for the complex test set. If the complex is not ranked within the first 700 hypotheses or no good results have been found the table entry is left blank.

Using the docking visualisation tool VIWISH [Klein96], some of the complexes have been examined in more detail. For the uracil-DNA glycosylase 1ugh the search stage finds 154 hypotheses with an RMSD of 5 Å or less. The list sorted by the scoring function contains 67 good hypotheses

among the first 100, and 8 that are better than 3 Å among the first 10. The correct solution is also ranked very high and appears among other hypotheses with an RMSD of less than 4 Å.



(a) Blue: Enzyme 1UGHE. Green: correct solution, inhibitor 1UGHI. Red: inhibitor hypothesis with RMSD=1.18 Å.




(b) Score for complex 1UGH. IPI=119, N10=8

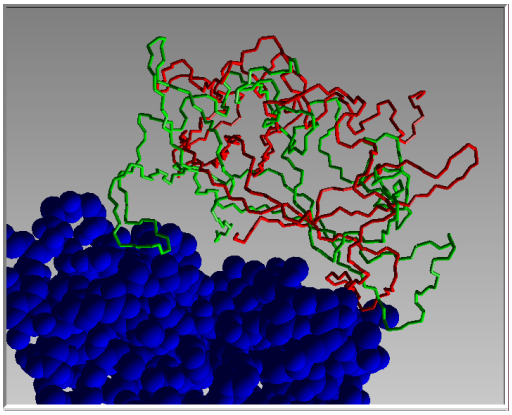
Figure 6.3.: Example 1ugh: since many good hypotheses (37 out of 50 with an RMSD below 4 Å) occur in the result set, the ComplexRank is “only” rank 25.

The hypotheses for the ribonuclease complex 8RSA achieve scores that range between 60 and 260. The docking has a rather high MinRMSD of 10 Å. A closer inspection shows that the complex has a contact that is distributed across two crystal interfaces. Several of the hypotheses have a very good overlap with the actual contact site(s), but the docking partner is rotated by 180° and thus has a high RMSD (see figure 6.4). The score of the correct solution reaches 286 and results in a top rank because this is 20% above the best estimated score for the generated hypotheses.

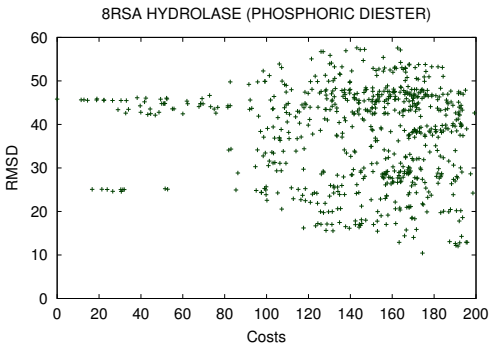
### 6.2.2. Characterisation of the Integrated Performance Indicator

The IPI values given in table 6.3 is large for docking runs with a large number of true positive (TP) results, and has a penalty for false positives which appear in the hypotheses plots above the diagonal. The correlation between the IPI and DRUF values such as the N100 is shown in figure 6.5.

The DRUF-N100 simply counts hypotheses with an RMSD<4 Å, whereas the IPI also considers the rank and the absolute RMSD of a hypothesis. The deviation of the IPI between several complexes with the same N100 stems either from a large number of false positives as in the case of entry 2SNI (IPI: 20.9) or several hypotheses with an RMSD below 3 Å.



(a) Green: correct solution, red: hypothesis with correct translation, but rotated by 180°.




(b) Score for complex 8rsa. IPI=-5.8, N100=0, MinRMSD=10.4 Å.

Figure 6.4.: Example 8rsa: The hypotheses with an RMSD around 25 Å correspond to the rotated solution, those with an RMSD around 45 Å to an incorrect docking site. Scoring the complex gives a ComplexRank of 1 if the contact site is included in the test set, but the search stage finds only a mirrored position of the inhibitor in the area of the correct active site.

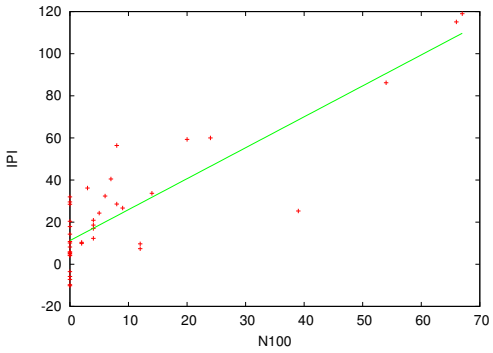



Figure 6.5.: Correlation between DRUF-N100 and the IPI value for the complexes from table 6.3. The top right complex is 1UGH as in figure 6.3(b), the complex from figure 6.4(b) is in the lower left corner.

### 6.2.3. Unbound Docking

The unbound docking experiments start from the isolated structures of the complex compounds. For each complex between one and 213 combinations of the unbound compounds have been tested, for an overall of 1880 test cases.

The MinRMSD value are shown in figure 6.6, where the boxplot shows the median, the range of the values and outliers. For most docking runs the median of the MinRMSD is below 5 Å, which is the cut-off value for the N100 DRuF score.

The oxireductases DYI, DYH and DYJ are a hard problem both in the complex (cf. table 6.3) and in the unbound case. Consequently, the scores that are assigned by the scoring function in figure 6.7 are very low, indicating a large amount of steric overlap for the correct solution.

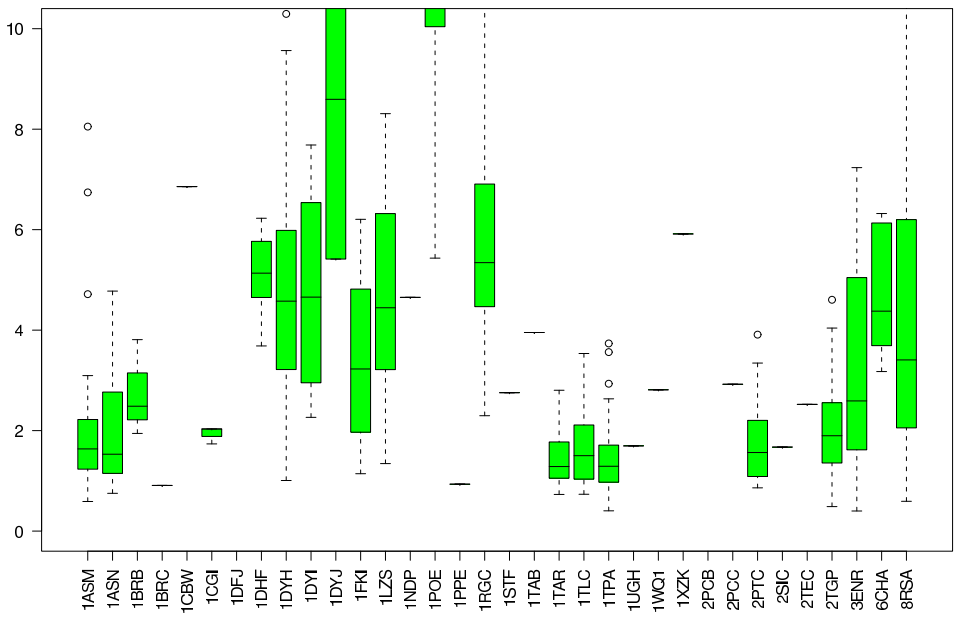



Figure 6.6.: Variation in MinRMSD for unbound docking. For each set of unbound structures the complex is given. The boxplot shows for each complex the median as a line within the box. The box is bound by the 25% and 75% margin, i.e. it contains 50% of all values. The whiskers at the end of the dotted lines encompass 90% of all values, with the remaining outliers plotted as circles. Those cases, where a good solution is found in the result set also have a small variance between several unbound combinations.

For 8rsa most of the test cases have a lower MinRMSD than the complex 8rsa. As explained in the previous section, the low performance in case of 8rsa was caused by the early pruning of the correct orientation within the active site. The unbound structures do not exhibit this problem, and have a median MinRMSD of 3.5 Å.

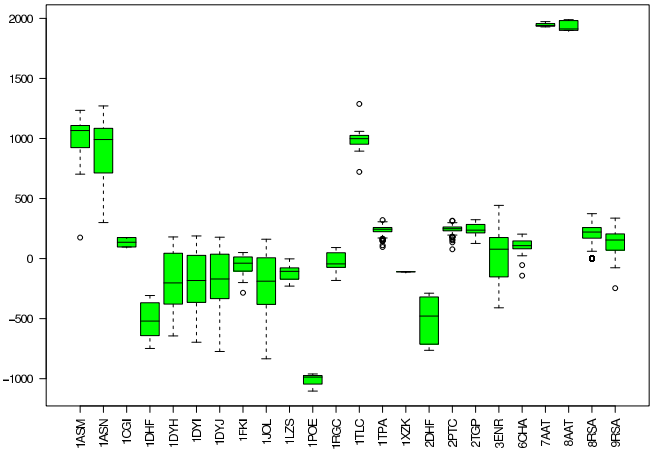



Figure 6.7.: Each of the mock complexes has been scored by the scoring function. The smaller the variance, the more reliable are the unbound structures.

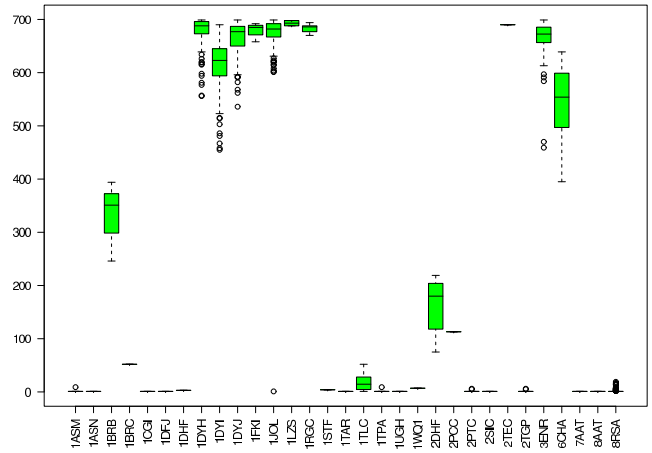



Figure 6.8.: The rank of the mock complex. Consistent with the result for the bound docking tests the oxireductases DYI, DYH and DYJ are not scored well.

### 6.2.4. Flexible Docking

The result in the unbound docking case have shown that for several complexes induced fit misleads the scoring function. This section will show that the flexibility term in the scoring function can improve the score.

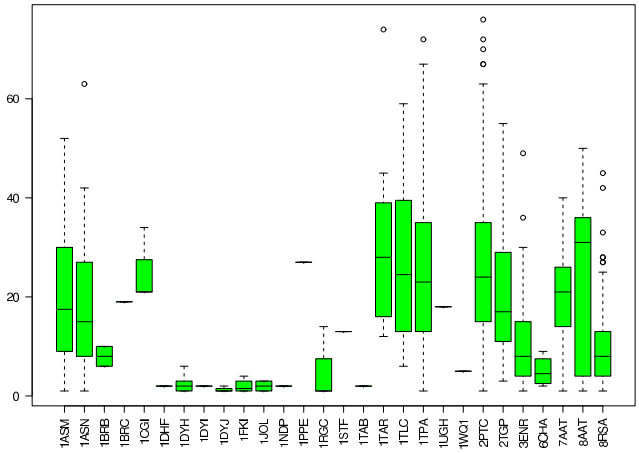



Figure 6.9.: The N100 value for unbound test cases. For all mock complexes where at least one good hypothesis is found, there are on average 20 hypotheses with an RMSD < 5 Å.

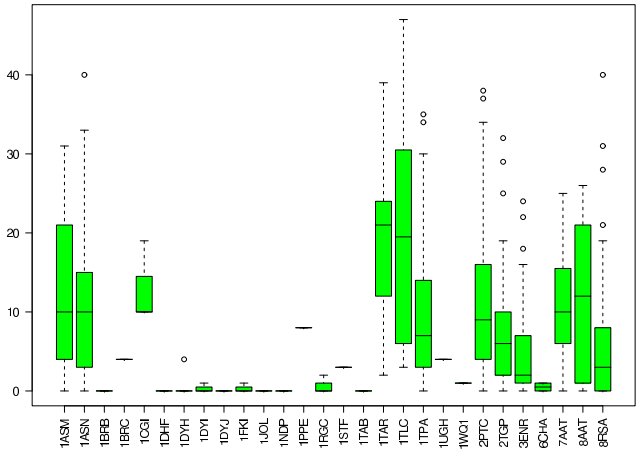



Figure 6.10.: The N50 value for unbound test cases, for which N100 is at least 1.

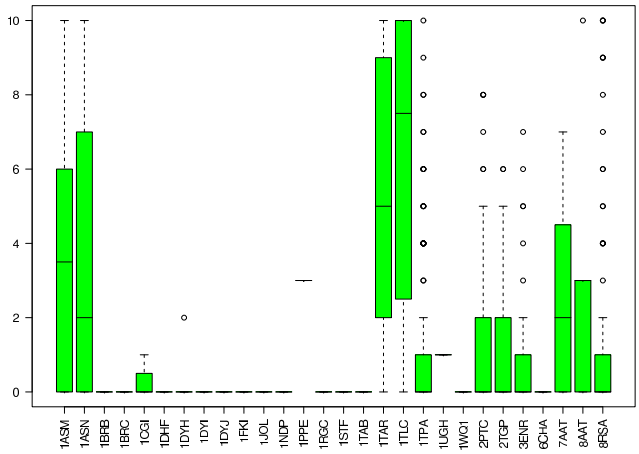
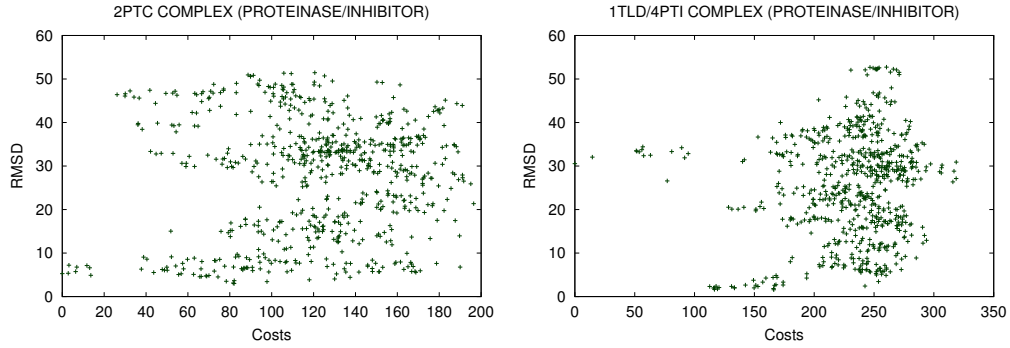



Figure 6.11.: The N10 value for unbound test cases, for which N100 $\geq$ 1.

One example is the complex between trypsin and its inhibitor 1BPTI, as shown in figure 3.2. Other complexes which are sequence identical to 2PTC are 1TPA and 3BTK, all from bos taurus (cow) with a resolution around 1.9 Å. One of the side chains with conformational changes is Lysine at position 15. In these complexes this residue consistently[1] has the $\chi_{1-4}$ angles in the 3rd, 2nd, 2nd and 2nd rotamer respectively. In the unbound form the 2nd or 4th rotamer change.

For the mock complexes 1TLD-4PTI and 1TLD-5PTI the geometric score drops from -87 to -129 and -193, reducing the (mock) complex rank from 1 to 2 and 11, respectively.




(a) Docking results for complex 2PTC. IPI=28.6, MinRMSD=3.0 Å, N100=8

(b) Results for the mock complex created from unbound 1TLD and 4PTI. IPI=27.5, MinRMSD=1.6 Å, N50=19.

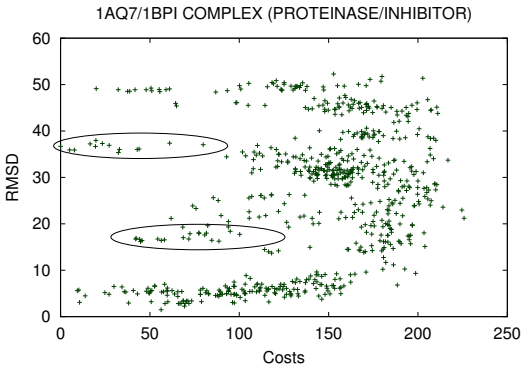
Figure 6.12.: Direct comparison of bound and unbound docking results. In the unbound case a set of hypotheses with an RMSD of around 32 Å is ranked better than the actual docking site if no elasticity information is used.

### Energy based Flexibility

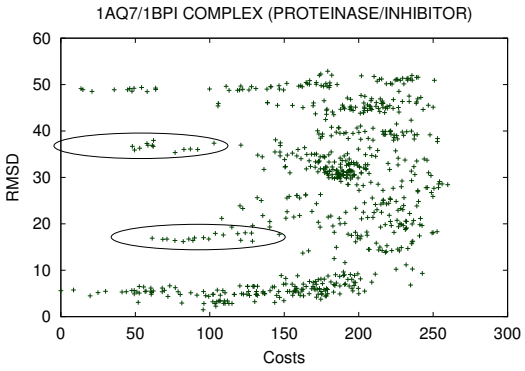
For the test with energy based predictions a preliminary set of 303 protein annotations has been provided by F. Zöllner [Koch02b]. The data (cf. table 6.4 for a summary) contains predictions for 50.000 residues. About 20% of the residues are predicted to change the rotamer. No docking tests have been done for those mock complexes for which the energy predictions are available for only one unbound partner.

Figure 6.13 shows the results for the mock complex 1AQ7-1BPI corresponding to the entry 1TPA (trypsin and its inhibitor). Four clusters of hypotheses with a similar translation/rotation and costs below 100 can be identified in the plots. One of them contains a wrong solution on rank #1. Enabling the flexibility term in the scoring function improves the scoring: the cluster with an RMSD between 15 and 20 Å drops from rank 28 to 37, the one between 35 and 40 Å from 1 to rank 16.

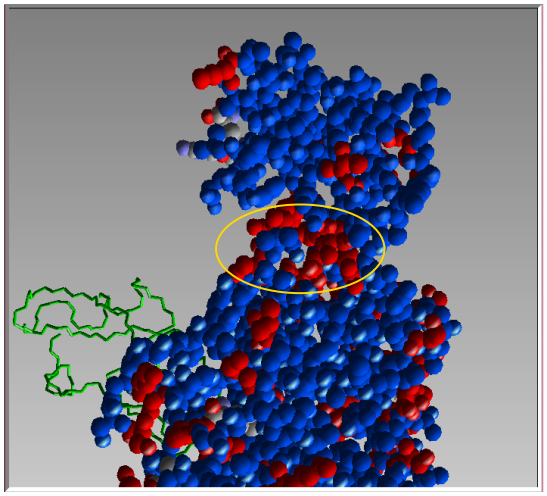
[1] With the exception of 1bhc, which has been crystallised in a decameric state.




(a) Unbound Docking without flexibility.




(b) Energy based flexibility used in scoring function.



(c) Residues predicted to change (red), the ellipse marks the interface. Correct solution shown in green.

Figure 6.13.: Comparison of unbound docking with and without flexibility information. Similar clusters (with respect to translation and rotation) of hypotheses can be identified. The position of the two ellipses is held fixed. Figure 6.13(c): Hypothesis ranked #1 drops to #16 with flexibility enabled. The active site contains residues marked flexible.

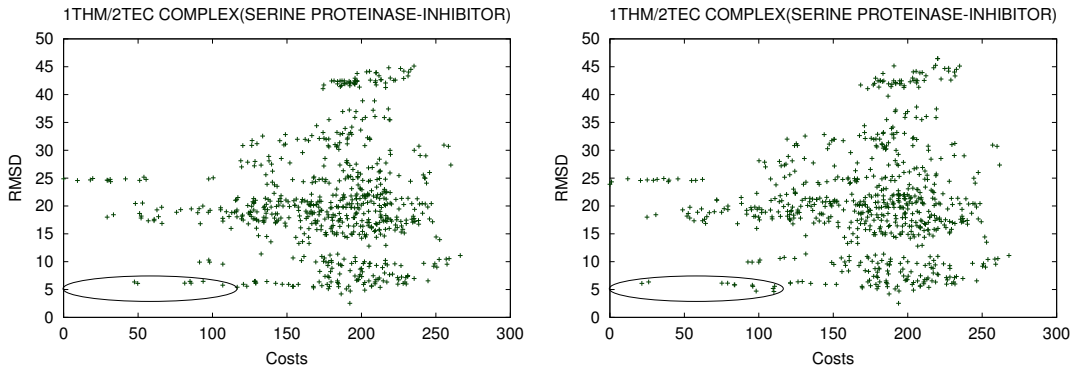
| | # Residues |
|---|---|
| Rotamer changes | 8280 |
| Rotamer unchanged | 41046 |
| Sum | 49326 |

Table 6.4.: Data set for energy based rotamer change prediction.

### Statistics based Flexibility

For the test with flexibility parameters based on rotamer statistics the data from [Koch03] has been used. For each combination of amino acid and secondary structure a rotamer change probability has been calculated and applied to the geometric fit term introduced in section 5.1.4.

The results for the mock complex 1THM/2TECI is shown in figure 6.14. Since the ellipses are kept in the same position between the two plots, kit can be seen that good hypotheses are ranked up in figure 6.14(b), including the two hypotheses advancing from rank #12 and #14 to rank #5 and #8 respectively.




(a) Unbound Docking without flexibility. MinRMSD=2.5 Å, IPI=29.0, N100=0.

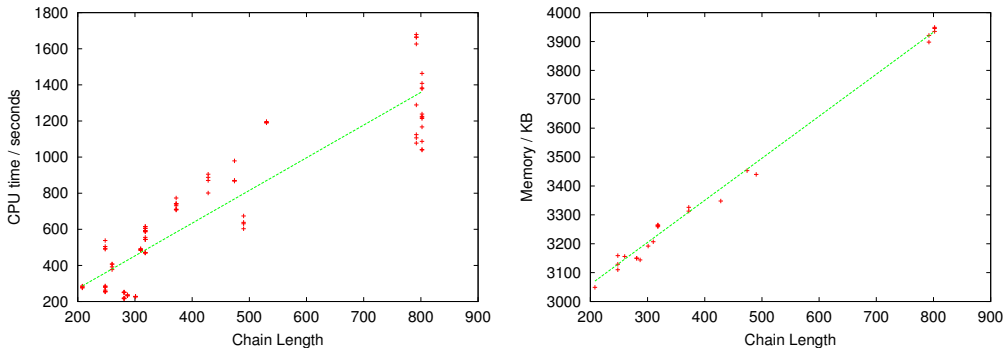
(b) Rotamer statistics based flexibility used in scoring function. IPI=29.9, MinRMSD=2.5 Å, N100=2.

Figure 6.14.: Comparison of unbound docking with and without flexibility information. The IPI values improves, most visible is the advance of two hypotheses from rank 12 and 14 up to 5 and 8.

Since ELMAR is intended to scan several docking candidates with many hypotheses each, the runtime and potential speed increases through parallelisation have been evaluated in the next sections.

### 6.2.5. Memory and Runtime Requirements

The runtime of a docking run is influenced by size of the molecules that are processed. The number of voxels in the discrete 3D representation is $O(n)$ with the chain length $n$ of the protein. The following experiments are conducted using the test set given in appendix A. First the runtimes[2] and the memory requirements for a docking run are measured in relation to the chain length of the docking problem, see figure 6.15.




(a) CPU time

(b) Average Memory footprint

Figure 6.15.: Correlation between chain length and resource utilisation. The runtime measurements have been run three times to eliminate performance differences.

Both memory and CPU time depend linearly on the size of the input. This is in contrast to methods calculating the pairwise contribution of atoms or residues to the interaction forces, which are NP-complete without the application of further heuristics.

The time needed for the cross correlation of the surface features also depends on the size of the search cube that encloses the contact site in a hypothesis. The correlation between chain length and -volume is shown in figure 6.16. Mostly the relationship is linear as well, with some exceptions if the proteins depart from globular shape. The ribonuclease inhibitor 1DFHI for example is a long, arc shaped structure with a length of 457 residues and volume 27786, whereas the neuroaminidase 1NMBN has a very compact shape with length 470 and volume 22125.

[2]The runtime is the actual CPU time needed on a 2.4 Ghz Intel P4 with 512-1024MB main memory.

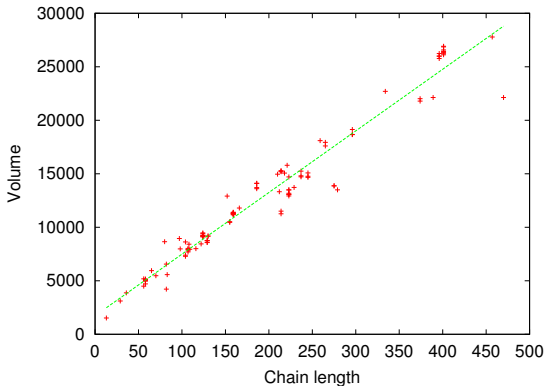



Figure 6.16.: Correlation between chain length and -volume. The relationship is mostly linear for the globular proteins in the test set.

### 6.2.6. Parallelisation Results

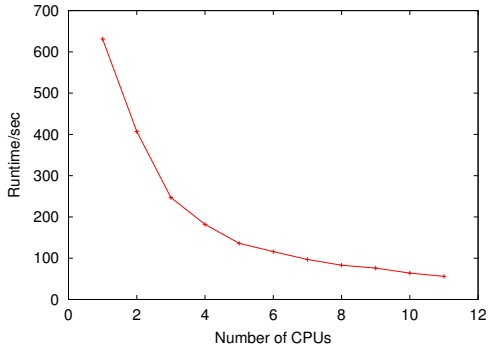
The most compute intensive part of the system is the scoring of hundreds, in case of a 1:N search thousands of hypotheses. In order to improve the latency of the system until all results have been processed, the main scoring loop was parallelised using the DACS communication system.

The efficiency of the parallelisation depends on a good balance between the increased number of CPUs on the one hand, and the increasing overhead of network communication on the other. To evaluate the efficiency, the execution times of a typical docking run have been measured on a varying number of approximately equally fast workstations. The times given are the “wallclock time” measuring the real world runtime, in contrast to the actual CPU time that would neglect the communication.

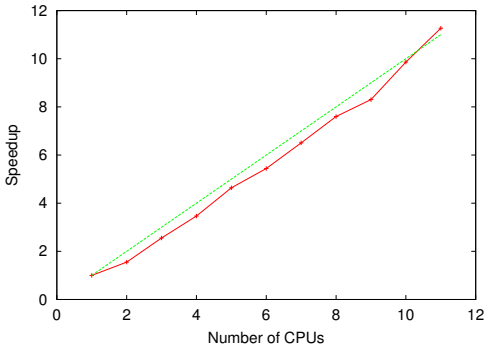
The scoring stage of the final docking module scales almost linearly on a small to medium number of CPUs. With this result an interactive use of the system becomes feasible.

## 6.3. Determining the biologically active Contact Site

A very different task for the docking system is the discrimination between monomeric and (homo-) dimeric and proteins in the crystal state, that is to discriminate native and artificial protein interfaces of densely packed proteins. Next to the biological significance of this problem it also shows that the scoring function is able to discriminate biologically sensible contacts and non-native interfaces.




(a) Wallclock execution time




(b) Speedup factor (red), theoretical limit (green).

Figure 6.17.: Parallel Execution of the scoring stage. All workstations use the Compaq Alpha CPU, ranging from XP1000 (CPU 21264) at 500 MHz to DS10 (CPU 21264A) at 667 MHz. These differences cause the above-theoretical speedup for the last machine on the right.

### 6.3.1. Crystal Packing

The X-RAY-diffraction patterns show the superposition of all molecules in the sample crystal. For the X-RAY-analysis proteins are forced into a crystal state, where the molecules are repeatedly placed in a regular – possibly symmetric – arrangement.

It is very difficult to distinguish between binding sites that occur in the biological active dimer and the contact areas that are caused by the crystal packing. The examples (see figure 6.18) show that a dimer might have a flat interface that does not have any obvious features that would explain its specificity, or some monomers with cavities that would otherwise suggest a binding pocket.

### 6.3.2. Scoring Monomer and Dimer Contact Areas

A test set of monomers and dimers has been described in [Ponstingl00]. Those PDB-files contain only the two chains in question, without any bound ligands or water molecules. For monomers, the second chain has been added by duplicating the protein as stored in the PDB and applying symmetry operations that conform to crystal parameters given in the PDB-file. The data set consists of 183 non-homologous PDB files containing pairs of proteins at a resolution of 3.0 Å or less. 95 entries contain the biologically active dimer, and 88 entries are two monomers that have a contact surface caused by the crystal packing. The classification has been taken from either the PDB itself,

or the corresponding SWISSPROT-entry. The pairwise sequence identity within members of one class is 25% or less. Thus the remaining entries cover a wide range of different proteins.

Since the 3D structures already have the bound conformation, no elasticity is to be used in these cases. The search space for the docking programs has been fixed to a translation and rotation of $T = (0,0,0)^T$ and $R = (0,0,0)^T$, so that the scoring function evaluates the given conformation only.

The results are shown in figure 6.19. The histograms show the distribution of scores for individual components of the scoring function. The geometrical surface complementarity (figure 6.19(b)) does not aid in the separation of mono- and dimeric complexes and is not considered in the following classification.

The scores for the components' geometry, hydrophobicity and charge have been used to train a support vector machine [Chang01] to solve the two-class classification problem. Support vector machines have a performance that is superior to e.g. linear models. To assess the separability of the problem and the performance of the classificator, the bootstrap method [Efron97] repeatedly performs the training on a subset of the original data, and uses the remaining cases as test set.

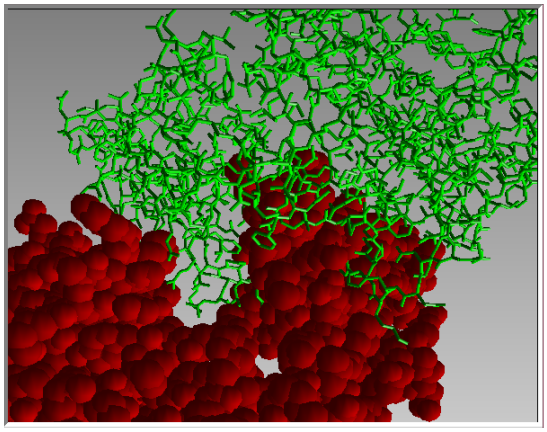
Using this classification scheme, the monomers can be predicted correctly with 95%, the dimers at 82% correctness. The overall error rate reaches 11%. Published performances [Henrick98; Ponstingl00] have error rates of 18%, 15.4% and 12.5%, depending on the scoring function and data set used.

| | | Predicted | |
|---|---|---|---|
| | | Monomer | Dimer |
| True | Monomer | 83% (73) | 17% (15) |
| | Dimer | 5% ( 5) | 95% (90) |

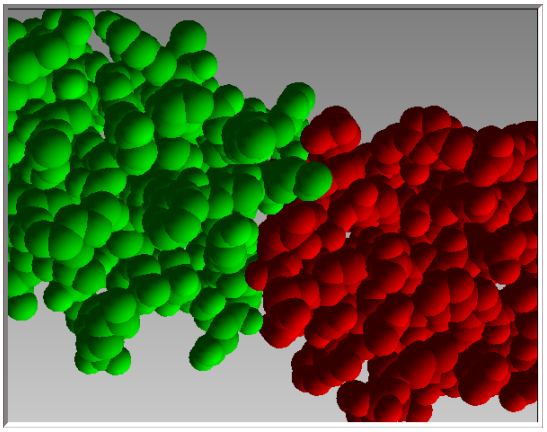
Table 6.5.: Classification results on the set of 188 monomer and dimer structures. The total accuracy of a ten-fold cross validation is 89%.

The performance is summarised in table 6.5. Most errors arise from dimers that have wrongly been classified as monomers, in the plot in figure 6.19(d) in the overlap region of both classes. An example of a mis-classification is e.g. the PDB entry 1AUA, a phospholipid-binding protein, whose biologically active unit is a monomer. In the crystal structure two α-helices interface with each other, which might contribute for the "good" score leading to the false prediction. A detailed discussion of the proteins prone to mis-classification and their individual properties is given in [Ponstingl00].

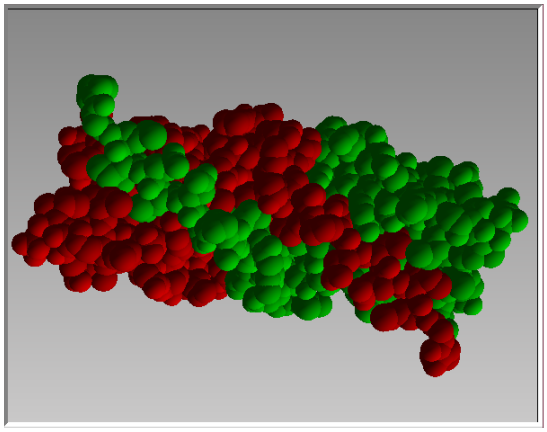
These results show that the scoring function is able to distinguish between contact sites due to biological function and contacts that arise from the experimental conditions during crystallisation.



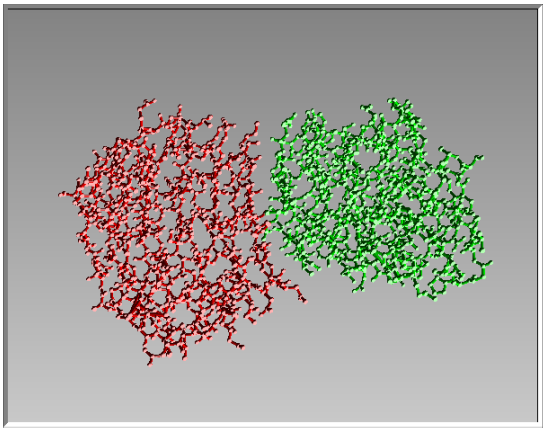
(a) Monomer 1CKM



(b) Monomer 2ACY

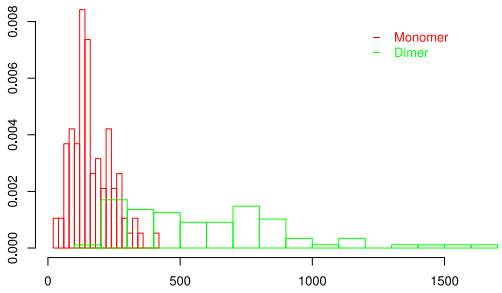


(c) Dimer 1BUO

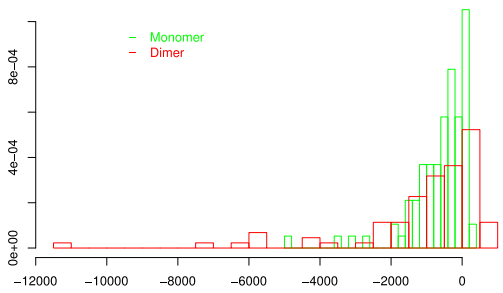


(d) Ball-and-Stick model of Dimer 1SLT

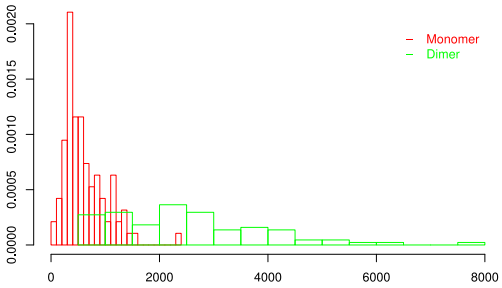
Figure 6.18.: Examples of monomers (first row) and dimers (second row) in the data set [Ponstingl00]. The two chains of dimer (c) are very entangled, but also monomer (a) appears to be bound within a binding pocket. Both mono- and dimers in the right column have a flat, loosely packed interface.



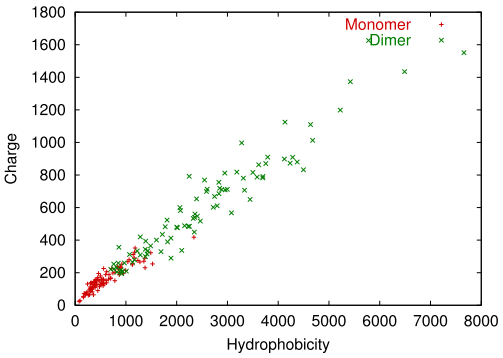

(a) Correlation of surface charges.




(b) Surface complementarity




(c) Correlation of hydrophobicity




(d) $(x,y)$ plot of figures 6.19(a) and 6.19(c)

Figure 6.19.: Individual components of the scoring function can be used for monomer/dimer discrimination. Histograms in (a) and (c) allow a discrimination between the two classes, especially in the 2D plot (d). Surface complementarity (b) is similarly distributed between both classes and not useful as discriminator.

## 6.4. Discussion of Results

The currently available test sets have been collected by various authors and been published over the years. The automated procedures presented in chapter 4 collect a large set of available PDB structures that can be used for training and validation of both bound and unbound docking systems.

The rather large number of test cases in contrast to the small overlap with the published test might suggest that the automatic method is not suitable to the task. However, the constraints placed on the search queries (resolution, sequence identity, etc.) are more stringent than those applied to the manually collected data. The intersection of the two database driven approaches leads to a reasonable set of training data. Though a manual inspection can also ensure the integrity of the unbound candidates, the automated system scales with the growing content of the PDB, and additional filters can be applied independently instead of re-iterating over the published papers on the PDB entries.

In the majority of the docking experiments on the cocrystallised complexes a good solution is found. In some cases no satisfying results can be obtained. These are mostly those cases, where the correct active site is not included in the initial result set. However, if the correct solution is inserted into the hypotheses list it scores well in some of these cases. If the constraints during the initial docking would be less strict and allow for a larger number of hypotheses during the later stages, those complexes would be solved as well, but the runtime increases for all test cases.

Energy calculations and rotamer statistics provide a measure of potential side chain conformational changes. Both approaches can improve the score of the correct solution as in figure 6.14(b) or decrease the score of incorrect hypotheses as in figure 6.13(b).

Because the flexibility parameters are calculated from observed rotamer changes and individual energy calculations, the resulting soft shell around the protein is more specific than a homogenous soft shell such as in [Fernandez-Recio02; Jiang02].

The integrated performance indicator (IPI) allows a finer assessment which of two given docking runs yields better results. The DRUF measures return e.g. the number of correct hypotheses within the first N results, but they do not consider at which position they occur.

The correlation between the IPI and N100 confirms that the IPI can express the quality of a docking run. Closer inspection of the variance shows that a large number of false positives decreases the score. The IPI is calculated as weighted sum of rank and RMSD. Such an accurate error measure is needed during systematic tests during training to find optimal parameter sets.

The task of discriminating between naturally occurring contact sites and those that appear only in the crystal state of the protein is very important during structure determination. Furthermore this also shows that the scoring function is able to discriminate biologically sensible contacts and non-native interfaces. The classification capabilities of the scoring function have been proven in

section 6.3 on the Monomer/Dimer test set published in [Henrick98; Ponstingl00]. In contrast to the hypotheses generated by the search stage, all these protein contacts are *real* conformations that exist in the crystal.

The protein structures have been scored by the ELMAR system. The search stage has been bypassed and flexibility information was disabled. Since the geometric complementarity of both natural and artificial contacts is good, using the hydrophobicity and charge as features was sufficient to discriminate 89% of the contacts correctly in a bootstrap evaluation setup. The results show that despite the simplicity of the scoring function compared to energy based approaches, it can be used in a 1:N docking scenario.

Another requirement in 1:N docking is a sufficiently short execution time. A linear speedup has been shown. Given a small cluster of 10 machines approximately 1500 candidates can be examined within a day. A larger cluster can therefore process the PDB overnight. The tradeoff between speed and accuracy will not allow a single sorted list to be created, but a vast portion of the immense search space can be eliminated. In most cases the results contain a sufficiently correct solution. This manageable number of hypotheses can be checked using e.g. energy calculations such as the time consuming approaches mentioned in chapter 3. Since only a fraction of the original search space has to be examined, reasonable runtime can be expected.

# Conclusion 7

Life sciences have seen an enormous boost during the late 1990's, with more still to come. In the postgenomic era the understanding of underlying mechanisms of interaction in the metabolic system will be important to interpret the huge amount of genomic sequence data.

This chapter will briefly summarise the thesis, and give an outlook to further research that may be conducted in the area.

## 7.1. Summary

Protein docking can be thought of both as a tool to simulate the chemical principles during protein interaction, as well as a Query-by-Content approach to search databases containing 3D protein structures for matching candidates. In the former case the quality of the simulation is the prime goal. Where no annotation or other meta data is available, a database search using a full scan over the data set is necessary. The required CPU time per docking candidate has to be limited in order to allow the scan to complete in a reasonable time.

After presenting the relevant biological background, an overview of current approaches to protein-docking has been given.

The semi automatic test set creation uses a data driven approach to extract proteins from the PDB for which both the complex and the unbound structures of individual protein chains are known. Combined with a heuristic combination of search terms on the meta data it can mine the PDB for test cases for protein interaction studies. Further heuristics can easily be added. The mechanism is especially important with the exponential growth of the protein databases.

Using the ELMAR system protein docking has been applied to a set of cocrystallised protein complexes. The system performed well on the test cases. In some cases the correct docking site was not found by the segmentation step. The scoring function, however, would still rank the correct

solution in a top position. The unbound docking experiments showed a similar performance for the mock complexes. Using flexibility parameters the scoring function has been shown to eliminate false positive docking sites and to increase the scoring of the true docking site.

In addition to a search of the conformational space another task in the area of structural protein informatics has been investigated. The classification capabilities of the scoring function have been proven on the Monomer/Dimer test set published in [Henrick98; Ponstingl00]. The system was able to discriminate between the correct active site and the imposed interfaces with 89% accuracy.

A major aspect of a large system is flexibility, especially in terms of reconfiguration or adding new modules. The ELMAR system (based on the C++-library developed in the BIOWEPRO project) uses the DACS communication system to integrate the individual modules. Users need no knowledge about the current configuration, which is handled by the DACS naming service. The integration of a relational database allows different systems to access (potentially intermediate) results for own purposes without the need for individually written custom IO functions.

## 7.2. Outlook

The ELMAR docking system has been developed to process the early stages during a search for protein interactions. It has to be combined with other tools to identify potential drug targets for further processing in a laboratory. Several modules can be thought of to improve the performance or extend the potential uses of this docking system.

### 7.2.1. Additional Post processing

Energy calculations are very time consuming and are used within the system only during the preprocessing stage that classifies the flexibility of a given residue. A module that scores all hypotheses resulting from the initial docking stage with energy functions would have a runtime that is not feasible for an 1:N docking scenario. Instead, such a module can be added to the end of the ELMAR-pipeline: Another instance of the scheduler assigns the well-scoring solutions for an evaluation using the AMBER or CHARMM force fields. This also adds an estimation of the free energy of the bound system.

### 7.2.2. User Interface for Navigation

Navigation within the potentially large result set can be improved. Currently the IPHEX system is being developed [Zöllner03]. The user judges the (im)plausibility of hypotheses on a scale between -2 and +2. The system modifies the weight parameters in the scoring function according to the user's feedback and then re-ranks the hypotheses.

Humans still have some superior capability of discriminating and recognising patterns in complex data sets, if they are properly presented. In addition to visual input recent advances in sonification [Hermann02] improve perception of dynamic data, such as during a “walk” through a high dimensional feature space. Interesting hot spots can be detected while “passing by” and revisited for closer inspection.

Another improvement at the level of human-machine communication would be a haptic interface, providing real-time feedback of the score calculated for conformations visualised on a 3D display: the user can “try” some docking positions. Whereas the computer has to go through the whole search space, human experts try educated guesses first, borrowing from their expertise and previous experience. The distributed nature of the ElMaR modules allows to sample the surrounding of the current position in an anticipatory way. To reduce latency several CPUs can be used in parallel.

### 7.2.3. 1:N Protein Docking

Most protein protein docking studies start from (a set of) pairs of molecules to dock. 1:N docking refers to screening of (potentially large) databases of possible docking partners for a given receptor. In the area of protein ligand docking several approaches exist for database screening, e.g. [Waszkowycz01]. For protein protein docking the algorithms employed have to identify non-docking conformations and have low runtime-requirements.

The main problem remains in identifying non-docking partners. For a given pair of proteins all existing approaches return the most probable docked configuration. Post-docking filters, such as mentioned in section 3.1.4 have to be trained and applied to discriminate between docking and non-docking molecules.

A different approach is depicted in figure 7.1. Starting from a known or hypothesised complex configuration (upper right), the contact site can be extracted and used to search for similar active sites. The contact sites can be modelled as feature graphs, covering the “points of interest” on the protein surface, and retrieved from the database using fast index structures, such as [Kriegel03; Ciaccia97]. The process can be refined iteratively to use the result set, with a scoring function to limit the entries according to their interface complementarity. Since the active site is known in advance, the translational and rotational search space is reduced to the small neighbourhood around the known site.

### 7.2.4. Scheduling for Any-Time Evaluation

A typical docking run searching a large database of docking partners can take several hours. The throughput of the system cannot be increased unless more CPU power is added to the parallelised

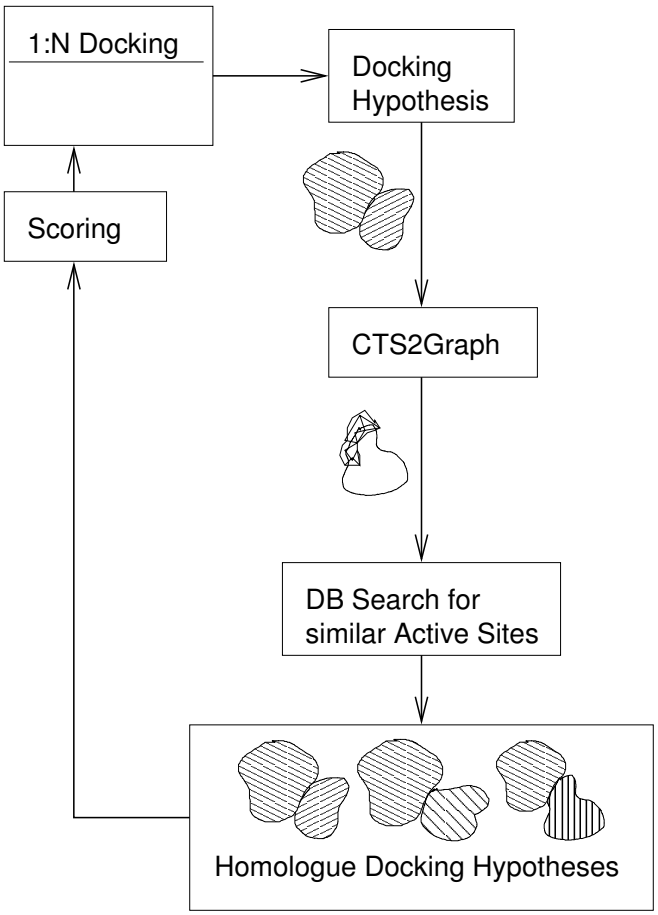



Figure 7.1.: Finding docking hypotheses homologue to a given one. Starting from given (or hypothetical) complex, similar docking partners are searched for. The active site at the receptor surface is known, reducing the size of the search space.

modules, but the latency of the system until first results are available can be reduced if early results leave the pipeline right after they are completed. This “any-time” property of a system allows to query the system at any time for the partial results. Though the results are not guaranteed to be globally optimal, they are the best seen so far and have to suffice under given time constraints. They are possibly superseded by a more complete and better result set afterwards.

The quality of the partial results depends on the order in which the hypotheses are considered, which is directly influenced by the sampling strategy. Because the first scoring function in the pipeline has a heuristic character it can be mislead, such as placing a good hypotheses at a low rank. The scheduling modules are prepared to use these scores not exclusively for the scheduling decisions towards later stages. They can be complemented by a random term, which also selects seemingly mediocre hypotheses which can turn out to be a good choice in the final scoring. A full search will eventually find those solutions as well, but the random approach is likely to process them earlier.

This strategy is similar to the access optimisation of high performance hard disks [Sagerer94], which use a stochastic access pattern that outperforms optimisation strategies that imposed a huge penalty in case of a misprediction.

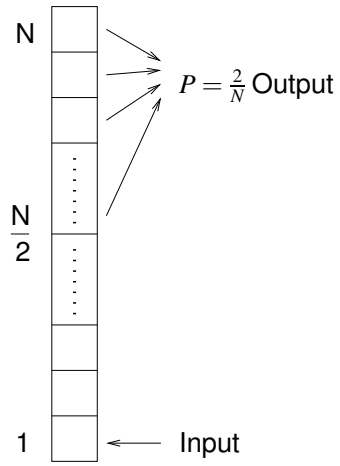



Figure 7.2.: Randomised queue: the probability $P$ can be chosen to select hypotheses from the upper half of the FIFO queue.

The scheduling between the individual modules in the ELMAR system is prepared to use a modified version of a FIFO queue: instead of the first-in first-out principle the next item is selected randomly from the upper half of the queue. The necessary infrastructure is already in place.

An evaluation is needed to assess the benefit in real-world applications. The degradation of the preliminary result sets under different time constraints indicates a necessary minimum latency of the system.

In the post-genomic era the genome of several organisms is known. By answering one question (What does our DNA look like ?) several more appear: when are genes translated ? how do proteins interact ? where are the regulatory knobs and dials that affect diseases ? To answer such questions we do not only need more but more reliable data. Even with the data itself, turning it into information requires efficient methods for analysis and interpretation of large genomic (and proteomic) data sets. Biologists and computer scientists will have to continue their work in this vast field in the coming years.

# Test Sets

# A

The following are test sets collected from several publications. If errors were found, they have been corrected if possible.

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1BRB(E:I) | 1BRA | 1BPI | COMPLEX(PROTEINASE/INHIBITOR) |
| 1CGI(E:I) | 1CHG | 1HPT | SERINE PROTEASE/INHIBITOR COMPLEX |
| 2KAI(AB:I) | 2PKA AB | 1BPI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2PTC(E:I) | 1BTY | 1BPI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SIC(E:I) | 1SUP | 2SSI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SNI(E:I) | 1SUP | 2CI2 | COMPLEX (PROTEINASE/INHIBITOR) |

Table A.1.: Enzyme/Inhibitor Complexes in [Betts99]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1ACB(E:I) | 5CHA A | 1ACB I | HYDROLASE(SERINE PROTEASE) |
| 1BRC(E:I) | 1BRA | 1BRC I | COMPLEX(PROTEINASE/INHIBITOR) |
| 1CHO(E:I) | 5CHA A | 1CHO I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1CSE(A:I) | 1SCD | 1CSE I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1PPE(E:I) | 1BTY | 1PPE I | HYDROLASE(SERINE PROTEINASE) |
| 1SBN(E:I) | 1SUP | 1SBN I | COMPLEX(PROTEINASE/INHIBITOR) |
| 1STF(E:I) | 1PPN | 1STF I | HYDROLASE(SULFHYDRYL PROTEINASE) |
| 1TAB(E:I) | 1BTY | 1TAB I | HYDROLASE (SERINE PROTEINASE) |
| 1TGS(Z:I) | 1TGT | 1TGS I | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TEC(E:I) | 1THM | 2TEC I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 4HTC(LH:I) | 2HNT | 4HTC I | HYDROLASE(SERINE PROTEASE) |
| 1UDI(E:I) | 1UDH | 1UDI I | COMPLEX (HYDROLASE/INHIBITOR) |

Table A.2.: Enzyme/Inhibitor Complexes with partially unbound components in [Betts99]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1MLC(AB:E) | 1MLB | 1LZA | COMPLEX (ANTIBODY/ANTIGEN) |
| 1VFB(AB:C) | 1VFA AB | 1LZA | IMMUNOGLOBULIN/HYDROLASE(O-GLYCOSYL) |

Table A.3.: Antibody/antigen Complexes in [Betts99]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1NCA(LH:N) | 1MCA AB | 7NN9 | HYDROLASE(O-GLYCOSYL) |
| 1NMB(LH:N) | 1NMB AB | 7NN9 | COMPLEX (HYDROLASE/IMMUNOGLOBULIN) |
| 1IGC(LH:A) | 1IGC LH | 1IGD | COMPLEX (ANTIBODY/BINDING PROTEIN) |
| 1JEL(LH:P) | 1JEL LH | 1POH | |
| 3HFL(LH:Y) | 3HFL LH | 1LZA | COMPLEX (ANTIBODY/ANTIGEN) |

Table A.4.: Antibody/antigen Complexes (1JEL has been superseeded by 2JEL since Feb 28, 1998) with partially unbound components in [Betts99]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1ATN(D:A) | 3DNI | 1ATN A | ENDODEOXYRIBONUCLEASE |
| 1GLA(G:F) | 1GLA G | 1F3G | PHOSPHOTRANSFERASE |
| 1SPB(S:P) | 1SUP | 1SPB P | COMPLEX (SERINE PROTEINASE/PROSEGMENT) |
| 2BTF(P:A) | 1PNE | 2BTF A | ACETYLATION AND ACTIN-BINDING |
| 3HHR(A:BC) | 1HGU | 3HHR BC | HORMONE/RECEPTOR |
| 1MDA(LH:A) | 1MDA LH | 1AAN | ELECTRON TRANSPORT |

Table A.5.: Other Complexes with partially unbound components in [Betts99]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1CGI(E:I) | 1CHG | 1HPT | SERINE PROTEASE/INHIBITOR COMPLEX |
| 1CHO(E:I) | 5CHA A | 2OVO | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 2PTC(E:I) | 2PTN | 6PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 1TGS(Z:I) | 2PTN | 1HPT | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SNI(E:I) | 1SUP | 2CI2 I | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SIC(E:I) | 1SUP | 3SSI | COMPLEX (PROTEINASE/INHIBITOR) |
| 1CSE(E:I) | 1SCD | 1ACB I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 2KAI(AB:I) | 2PKA XY | 6PTI | COMPLEX (PROTEINASE-INHIBITOR) |
| 1BRC(E:I) | 1BRA | 1AAP A | COMPLEX(PROTEINASE/INHIBITOR) |
| 1ACB(E:I) | 5CHA A | 1CSE I | HYDROLASE(SERINE PROTEASE) |
| 1BRS(A:D) | 1A2P B | 1A19 A | ENDONUCLEASE |
| 1MAH(A:F) | 1MAA B | 1FSC | COMPLEX (HYDROLASE/TOXIN) |
| 1UGH(E:I) | 1AKZ | 1UGI A | GLYCOSYLASE |
| 1DFJ(E:I) | 2BNH | 7RSA | COMPLEX (ENDONUCLEASE/INHIBITOR) |

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1FSS(A:B) | 2ACE E | 1FSC | COMPLEX (SERINE ESTERASE/TOXIN) |
| 1AVW(A:B) | 2PTN | 1BA7 A | COMPLEX (PROTEINASE/INHIBITOR) |

Table A.6.: Enzyme/Inhibitor complexes in [Chen02b]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1PPE(E:I) | 2PTN | 1PPE I | HYDROLASE(SERINE PROTEINASE) |
| 1TAB(E:I) | 2PTN | 1TAB I | HYDROLASE (SERINE PROTEINASE) |
| 1UDI(E:I) | 1UDH | 1UDI I | COMPLEX (HYDROLASE/INHIBITOR) |
| 1STF(E:I) | 1PPN | 1STF I | HYDROLASE(SULFHYDRYL PROTEINASE) |
| 2TEC(E:I) | 1THM | 2TEC I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 4HTC(LH:I) | 2HNT LCEF | 4HTC I | HYDROLASE(SERINE PROTEASE) |

Table A.7.: Enzyme/Inhibitor complexes with partially unbound components in [Chen02b]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1MLC(AB:E) | 1MLB AB | 1LZA | COMPLEX (ANTIBODY/ANTIGEN) |
| 1WEJ(LH:F) | 1QBL LH | 1HRC | COMPLEX (ANTIBODY/ELECTRON TRANSPORT) |
| 1AHW(DE:F) | 1FGN LH | 1BOY | COMPLEX (IMMUNOGLOBULIN/TISSUE FACTOR) |
| 1DQJ(AB:C) | 1DQQ LH | 3LZT | IMMUNE SYSTEM/HYDROLASE |
| 1BVK(DE:F) | 1BVL LH | 3LZT | COMPLEX (HUMANIZED ANTIBODY/HYDROLASE) |

Table A.8.: Antibody/antigen complexes in [Chen02b]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1FBI(LH:X) | 1FBI LH | 1HHL | COMPLEX (ANTIBODY/ANTIGEN) |
| 2JEL(LH:P) | 2JEL LH | 1POH | COMPLEX (ANTIBODY/ANTIGEN) |
| 1BQL(LH:Y) | 1BQL LH | 1DKJ | COMPLEX (ANTIBODY/ANTIGEN) |
| 1JHL(LH:A) | 1JHL LH | 1GHL A | COMPLEX(ANTIBODY-ANTIGEN) |
| 1NCA(LH:N) | 1NCA LH | 7NN9 | HYDROLASE(O-GLYCOSYL) |
| 1NMB(LH:N) | 1NMB LH | 7NN9 | COMPLEX (HYDROLASE/IMMUNOGLOBULIN) |
| 1MEL(B:M) | 1MEL B | 1LZA | COMPLEX (ANTIBODY/ANTIGEN) |
| 2VIR(AB:C) | 2VIR AB | 2VIU A | COMPLEX (HEMAGGLUTININ/IMMMUNOGLOBULIN) |
| 1EO8(LH:A) | 1EO8 LH | 2VIU A | VIRUS/VIRAL PROTEIN |
| 1QFU(LH:A) | 1QFU LH | 2VIU A | VIRAL PROTEIN/IMMUNE SYSTEM |
| 1IAI(MI:LH) | 1AIF LH | 1IAI LH | COMPLEX (IMMUNOGLOBULIN IGG1/IGG2A) |

Table A.9.: Antibody/antigen complexes with partially unbound components in [Chen02b]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 2PCC(A:B) | 1CCA | 1YCC | OXIDOREDUCTASE/ELECTRON TRANSPORT |
| 1WQ1(G:R) | 1WER | 5P21 | COMPLEX (GTP-BINDING/GTPASE ACTIVATION) |
| 1AVZ(B:C) | 1AVV | 1SHF A | COMPLEX (MYRISTYLATION/TRANSFERASE) |
| 1MDA(LH:A) | 2BBK LH | 1AAN | ELECTRON TRANSPORT |

Table A.10.: Other complexes in [Chen02b]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1IGC(LH:A) | 1IGC LH | 1IGD | COMPLEX (ANTIBODY/BINDING PROTEIN) |
| 1ATN(A:D) | 1ATN A | 3DNI | ENDODEOXYRIBONUCLEASE |
| 1GLA(G:F) | 1GLA G | 1F3G | PHOSPHOTRANSFERASE |
| 1SPB(S:P) | 1SUP | 1SPB P | COMPLEX (SERINE PROTEINASE/PROSEGMENT) |
| 2BTF(A:P) | 2BTF A | 1PNE | ACETYLATION AND ACTIN-BINDING |
| 1A0O(A:B) | 1CHN | 1A0O B | CHEMOTAXIS |

Table A.11.: Other complexes with partially unbound components in [Chen02b]

| Complex | Description |
|---|---|
| 1CHO(E:E) | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1FDL(LH:Y) | COMPLEX (ANTIBODY-ANTIGEN) |
| 1TEC(E:I) | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1TGS(Z:I) | COMPLEX (PROTEINASE/INHIBITOR) |
| 2HFL(LH:Y) | |
| 2KAI(AB:I) | COMPLEX (PROTEINASE-INHIBITOR) |
| 2MHB(A:B) | OXYGEN TRANSPORT |
| 2PTC(E:I) | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SEC(E:I) | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 2SNI(E:I) | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TGP(Z:I) | COMPLEX (PROTEINASE/INHIBITOR) |
| 3HFM(LH:Y) | COMPLEX(ANTIBODY-ANTIGEN) |
| 4CPA(?:I) | HYDROLASE (C-TERMINAL PEPTIDASE) |
| 4HVP(A:B) | HYDROLASE(ACID PROTEINASE) |
| 4SGB(E:I) | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 4TPI(Z:I) | COMPLEX (PROTEINASE/INHIBITOR) |
| 1ABI(H:L) | HYDROLASE(SERINE PROTEINASE) |
| 1ACB(E:I) | HYDROLASE(SERINE PROTEASE) |
| 1CSE(E:I) | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1TPA(E:I) | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SIC(E:I) | COMPLEX (PROTEINASE/INHIBITOR) |
| 5HMG(E:F) | INFLUENZA VIRUS HEMAGGLUTININ |
| 6TIM(A:B) | ISOMERASE(INTRAMOLECULAR OXIDOREDUCTASE) |
| 8FAB(A:B) | IMMUNOGLOBULIN |

| | |
|---|---|
| 9LDT(A:B) | OXIDOREDUCTASE(CHOH(D)-NAD+(A)) |
| 9RSA(A:B) | HYDROLASE (PHOSPHORIC DIESTER) |

Table A.12.: Complex test set, 2HFL has been superseeded by 3HFL May 2nd, 1995 in [Norrel99]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 3HFM(LH:Y) | 1HFM | 1LYM A | COMPLEX(ANTIBODY-ANTIGEN) |
| 3HFM(LH:Y) | 1HFM | 1LYM B | COMPLEX(ANTIBODY-ANTIGEN) |
| 2TGP(Z:I) | 1TGN | 4PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TGP(Z:I) | 1TGN | 5PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TGP(Z:I) | 1TGN | 6PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1TLD | 4PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1TLD | 5PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1TLD | 6PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 1FDL(LH:Y) | 2HFL | 1LYZ | COMPLEX (ANTIBODY-ANTIGEN) |
| 1FDL(LH:Y) | 2HFL | 6LYZ | COMPLEX (ANTIBODY-ANTIGEN) |
| 2KAI(AB:I) | 2PKA | 4PTI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2KAI(AB:I) | 2PKA | 5PTI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2KAI(AB:I) | 2PKA | 6PTI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2SNI(E:I) | 2SBT | 2CI2 | COMPLEX (PROTEINASE/INHIBITOR) |
| 1CHO(E:I) | 5CHA A | 2OVO | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1CHO(E:I) | 5CHA B | 2OVO | COMPLEX(SERINE PROTEINASE-INHIBITOR) |

Table A.13.: Unbound test cases. The complex entries are estimated from other publications since they are not mentioned in [Norrel99]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1BRB(E:I) | 1BRA | 1BPI | COMPLEX(PROTEINASE/INHIBITOR) |
| 1BRB(E:I) | 1BRA | 4PTI | COMPLEX(PROTEINASE/INHIBITOR) |
| 1BRB(E:I) | 1BRA | 5PTI | COMPLEX(PROTEINASE/INHIBITOR) |
| 1BRB(E:I) | 1BRA | 6PTI | COMPLEX(PROTEINASE/INHIBITOR) |
| 1CGI(E:I) | 1CHG | 1HPT | SERINE PROTEASE/INHIBITOR COMPLEX |
| 1CGI(E:I) | 2CHG AB | 1HPT | SERINE PROTEASE/INHIBITOR COMPLEX |
| 2KAI(AB:I) | 2PKA | 1BPI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2KAI(AB:I) | 2PKA | 4PTI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2KAI(AB:I) | 2PKA | 5PTI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2KAI(AB:I) | 2PKA | 6PTI | COMPLEX (PROTEINASE-INHIBITOR) |
| 2PTC(E:I) | 3PTN | 1BPI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 3PTN | 4PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 3PTN | 5PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 3PTN | 6PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1TLD | 1BPI | COMPLEX (PROTEINASE/INHIBITOR) |

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 2PTC(E:I) | 1TLD | 4PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1TLD | 5PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1TLD | 6PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1BTY | 1BPI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1BTY | 4PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1BTY | 5PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2PTC(E:I) | 1BTY | 6PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SIC(E:I) | 1SUP | 3SSI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SIC(E:I) | 2STL | 3SSI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SNI(E:I) | 2SBT | 2CI2 | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SNI(E:I) | 2SBT | 1YPC | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SNI(E:I) | 1SUP | 2CI2 | COMPLEX (PROTEINASE/INHIBITOR) |
| 2SNI(E:I) | 1SUP | 1YPC | COMPLEX (PROTEINASE/INHIBITOR) |
| 1BRS(A:D) | 1A2P | 1A19 | ENDONUCLEASE |
| 1BRS(A:D) | 1A2P | 1BTA | ENDONUCLEASE |
| 1BRS(A:D) | 1BAO | 1A19 | ENDONUCLEASE |
| 1BRS(A:D) | 1BAO | 1BTA | ENDONUCLEASE |
| 1BVN(P:T) | 1PIF | 2AIT | HYDROLASE/HYDROLASE INHIBITOR |
| 1CAO(ABCFGH:DI) | 5CHA AB | 1APP AB | LYASE(OXO-ACID) |
| 2TGP(Z:I) | 1TGN | 1BPI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TGP(Z:I) | 1TGN | 4PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TGP(Z:I) | 1TGN | 5PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TGP(Z:I) | 1TGN | 6PTI | COMPLEX (PROTEINASE/INHIBITOR) |
| 1CBW(ABCFGH:DI) | 5CHA AB | 1BPI | COMPLEX (SERINE PROTEASE/INHIBITOR) |
| 1CBW(ABCFGH:DI) | 5CHA AB | 4PTI | COMPLEX (SERINE PROTEASE/INHIBITOR) |
| 1CBW(ABCFGH:DI) | 5CHA AB | 5PTI | COMPLEX (SERINE PROTEASE/INHIBITOR) |
| 1CBW(ABCFGH:DI) | 5CHA AB | 6PTI | COMPLEX (SERINE PROTEASE/INHIBITOR) |
| 1CHO(E:I) | 5CHA AB | 2OVO | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1HIA(ABXY:IJ) | 1AO5 | 1BX8 | COMPLEX (PROTEASE/INHIBITOR) |
| 1UGH(E:I) | 1AKZ | 1UGI A | GLYCOSYLASE |
| 1BRC(E:I) | 1BRA | 1AAP AB | COMPLEX(PROTEINASE/INHIBITOR) |
| 1DFJ(E:I) | 1BNH | 7RSA | COMPLEX (ENDONUCLEASE/INHIBITOR) |

Table A.14.: Enzyme/Inhibitor complexes, the original publication has a typo, where 1AL9 should be 1A19, and 4PTI has no chain “Z” in [Halperin02]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1ACB(E:I) | 4CHA | 1ACB I | HYDROLASE(SERINE PROTEASE) |
| 1BRC(E:I) | 1BRA | 1BRC I | COMPLEX(PROTEINASE/INHIBITOR) |
| 1CHO(E:I) | 4CHA | 1CHO I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1CSE(E:I) | 1SCD | 1CSE I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 1PPE(E:I) | 3PTN | 1PPE I | HYDROLASE(SERINE PROTEINASE) |
| 1SBN(E:I) | 1SUP | 1SBN I | COMPLEX(PROTEINASE/INHIBITOR) |

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1STF(E:I) | 1PPN | 1SFT E | HYDROLASE(SULFHYDRYL PROTEINASE) |
| 1TAB(E:I) | 3PTN | 1TAB I | HYDROLASE (SERINE PROTEINASE) |
| 1TGS(Z:I) | 1TGT | 1TGS I | COMPLEX (PROTEINASE/INHIBITOR) |
| 2TEC(E:I) | 1THM | 2TEC I | COMPLEX(SERINE PROTEINASE-INHIBITOR) |
| 4HTC(E:I) | 2HNT | 4HTC I | HYDROLASE(SERINE PROTEASE) |
| 1UDI(E:I) | 1UDH | 1UDI I | COMPLEX (HYDROLASE/INHIBITOR) |

Table A.15.: Enzyme/Inhibitor complexes, the original publication has a typo, where 1SFT should be 1STF in [Halperin02]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1MLC(ABCD:EF) | 1MLB | 1LZA | COMPLEX (ANTIBODY/ANTIGEN) |
| 1MLC(ABCD:EF) | 1MLB | 1LYZ | COMPLEX (ANTIBODY/ANTIGEN) |
| 1MLC(ABCD:EF) | 1MLB | 6LYZ | COMPLEX (ANTIBODY/ANTIGEN) |
| 1MLC(ABCD:EF) | 1MLB | 3LZT | COMPLEX (ANTIBODY/ANTIGEN) |
| 1VFB(AB:C) | 1VFA | 1LZA | IMMUNOGLOBULIN/HYDROLASE(O-GLYCOSYL) |
| 1VFB(AB:C) | 1VFA | 1LYZ | IMMUNOGLOBULIN/HYDROLASE(O-GLYCOSYL) |
| 1VFB(AB:C) | 1VFA | 6LYZ | IMMUNOGLOBULIN/HYDROLASE(O-GLYCOSYL) |
| 1VFB(AB:C) | 1VFA | 3LZT | IMMUNOGLOBULIN/HYDROLASE(O-GLYCOSYL) |
| 1FDL(LH:Y) | 3HFL LH | 1LZA | COMPLEX (ANTIBODY-ANTIGEN) |
| 1FDL(LH:Y) | 3HFL LH | 1LYZ | COMPLEX (ANTIBODY-ANTIGEN) |
| 1FDL(LH:Y) | 3HFL LH | 6LYZ | COMPLEX (ANTIBODY-ANTIGEN) |
| 1FDL(LH:Y) | 3HFL LH | 3LZT | COMPLEX (ANTIBODY-ANTIGEN) |
| 3HFM(LH:Y) | 1HFM | 5LYM A | COMPLEX(ANTIBODY-ANTIGEN) |
| 3HFM(LH:Y) | 1HFM | 5LYM B | COMPLEX(ANTIBODY-ANTIGEN) |
| 1AHW(ABDE:CF) | 1FGN LH | 1BOY | COMPLEX (IMMUNOGLOBULIN/TISSUE FACTOR) |
| 1BVK(AB:C) | 1BVL AB | 1LZA | COMPLEX (HUMANIZED ANTIBODY/HYDROLASE) |
| 1BVK(AB:C) | 1BVL AB | 1LYZ | COMPLEX (HUMANIZED ANTIBODY/HYDROLASE) |
| 1BVK(AB:C) | 1BVL AB | 6LYZ | COMPLEX (HUMANIZED ANTIBODY/HYDROLASE) |
| 1BVK(AB:C) | 1BVL AB | 3LZT | COMPLEX (HUMANIZED ANTIBODY/HYDROLASE) |
| 1DQI(AB:C) | 1DQQ AB | 1LZA | OXIDOREDUCTASE |
| 1DQI(AB:C) | 1DQQ AB | 1LYZ | OXIDOREDUCTASE |
| 1DQI(AB:C) | 1DQQ AB | 6LYZ | OXIDOREDUCTASE |
| 1DQI(AB:C) | 1DQQ AB | 3LZT | OXIDOREDUCTASE |

Table A.16.: Antibody/Antigen complexes in [Halperin02]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1NCA(LH:N) | 1NCA LH | 7MN9 | HYDROLASE(O-GLYCOSYL) |
| 1NMB(LH:N) | 1NMB LH | 7MN9 | COMPLEX (HYDROLASE/IMMUNOGLOBULIN) |
| 1IGC(LH:A) | 1IGC LH | 1IGD | COMPLEX (ANTIBODY/BINDING PROTEIN) |
| 1JEL(LH:P) | 1JEL LH | 1POH | |
| 3HFL(LH:P) | 3HFL LH | 1LZA | COMPLEX (ANTIBODY/ANTIGEN) |

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 3HFM(LH:Y) | 3HFM LH | 1LZA | COMPLEX(ANTIBODY-ANTIGEN) |

Table A.17.: Antibody/Antigen complexes in [Halperin02]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1MDA(LH:A) | 2BBK | 1AAN | ELECTRON TRANSPORT |
| 4HVP(A:B) | 3HVP | 3HVP | HYDROLASE(ACID PROTEINASE) |
| 1FSS(A:B) | 2ACE | 1FSC | COMPLEX (SERINE ESTERASE/TOXIN) |
| 2PCB(AC:B) | 1CCP | 1HRC | OXIDOREDUCTASE/ELECTRON TRANSPORT |
| 2PCB(AC:B) | 1CCP | 1YCC | OXIDOREDUCTASE/ELECTRON TRANSPORT |
| 1WEJ(LH:F) | 1QBL LH | 1HRC | COMPLEX (ANTIBODY/ELECTRON TRANSPORT) |
| 1AVZ(A:C) | 1AVV | 1SHF A | COMPLEX (MYRISTYLATION/TRANSFERASE) |
| 1WQ1(G:R) | 1WER | 5P21 | COMPLEX (GTP-BINDING/GTPASE ACTIVATION) |
| 1BDJ(A:B) | 3CHY | 2A0B | COMPLEX (CHEMOTAXIS/TRANSFERASE) |

Table A.18.: Other complexes, the original publication has entry entry1WQ1GR mistyped as entry1WQLQR. in [Halperin02]

| Complex | Receptor | Ligand | Description |
|---|---|---|---|
| 1ATN(D:A) | 3DNI | 1ATN A | ENDODEOXYRIBONUCLEASE |
| 1GLA(G:F) | 1GLA G | 1F3G | PHOSPHOTRANSFERASE |
| 1SPB(S:P) | 1SUP | 1SPB P | COMPLEX (SERINE PROTEINASE/PROSEGMENT) |
| 2BTF(P:A) | 1PNE | 2BTF A | ACETYLATION AND ACTIN-BINDING |
| 3HHR(A:BC) | 1HGU | 3HHR BC | HORMONE/RECEPTOR |

Table A.19.: Other complexes in [Halperin02]

# Curriculum Vitae

Steffen Neumann

## Address

Fahrenheitweg 9
33613 Bielefeld
Germany
Phone: +49 (0) 521 106 2949
Fax: +49 (0) 521 106 2992
Email: sneumann@TechFak.Uni-Bielefeld.DE
Homepage: www.TechFak.Uni-Bielefeld.DE/~sneumann/

## Personal Details

Gender: Male
Date of birth: 04th of August, 1972
Place of birth: Düsseldorf, Germany
Present Citizenship: German
Parents: Edeltraud and Manfred Neumann

## Education

| | |
|---|---|
| 1979-1983 | Primary School "Geschwister Scholl", Monheim, Germany |
| 1983-1992 | Secondary School "Otto Hahn", Monheim, Germany. Leaving with the A-Level. |
| 1992-1999 | Undergraduate Studies Bielefeld University in "Computing in the natural sciences".<br>Specialisation: Applied Computer Science, Neurobiology and Neuropsychology.<br>Thesis: *Scoring Protein Docking Hypotheses using elastic Matching*; Supervisors: Prof. Dr. G. Sagerer and Prof. Dr. S. Posch |
| 10/1994-05/1995 | Erasmus Studies in Computer Science and Biotechnology at the Dublin City University. |
| Since 05/1999 | Ph.D. Student Bielefeld University, Germany.<br>Project title: *1:n Protein Docking* Supervisors: Supervisor: Prof. Dr. G. Sagerer and Prof. Dr. F. Kummert. |

## Thesis: Soft volume models for protein-protein docking

Protein docking is the question whether and how two proteins interact, starting from their 3D structure. For training and test of docking systems large data sets are needed. Protein Data Base (PDB) automated test case generation is needed. A method for automated Test Case generation based on combined searches and filters on the content of the PDB is described.

ElMaR is a distributed, modular and optionally parallel docking system. Fast Docking algorithms usually employ the rigid-body assumption and score geometric complementarity as well as physico-chemical features. However, for unbound protein docking steric clashes might impose wrong penalization if side-chains change their conformation during the docking process. ElMaR incorporates protein flexibility obtained through statistics and force field calculation. Using a fast correlation technique steric clash penalties are weighted according to the possibility of amino acid rotamer changes.

The ability to distinguish between native and non-native contact sites is tested on interfaces in protein crystals. A performance exceeding published results has been achieved, Results on the generated test sets are presented and discussed.

## Working Experience

| | |
|---|---|
| 10/1995–12/1996 | Student worker in the inter faculty project "Studienberatung Online": deployment of a Linux network and Web Servers. Teaching UNIX and HTML. |
| 02/1997-04/1997 | Internship at the Ministry for Science and Education, Düsseldorf, Germany |
| 4/1997–4/1998 | Student worker in the "Research Focus 360" `http://www.sfb360.Uni-Bielefeld.DE` with Dr. N. Jungclaus on the Distributed Application Communication System. |
| Since 05/1999 | Research Assistant at the University of Bielefeld, Germany. |

## Teaching Experience

| | |
|---|---|
| 1999 | Seminar "Proteins and computer science" |
| 2000 | Lab Course "Operating Systems" |
| 2001 | Seminar "Proteins and computer science" |
| 2001 | Part of Lecture "Databases" |
| 2002 | Lab Course "Applied computer science and Proteins" |
| 2003 | Seminar "Pervasive Computing" |

## Systems & Programming Languages

| | |
|---|---|
| **Solaris/Tru64/Linux** | very good |
| **Windows** | good |
| **C/C++** | very good |
| MySQL | |
| **(several APIs)** | very good |
| **perl** | good |
| **python, Tcl/Tk** | fair |

## Languages

| | |
|---|---|
| **German** | native |
| **English** | fluently |
| **French** | fair |

## References

These persons are familiar with my professional qualifications and my character:


**Prof. Gerhard Sagerer**
Ph.D. Thesis supervisor
University of Bielefeld,
Technical Faculty
33501 Bielefeld
Germany
Phone: +49 (0) 521 106 2935
Fax: +49 (0) 521 106 2992
Email: sagerer@TechFak.Uni-Bielefeld.DE

**Prof. Stefan Posch**
Diploma Thesis supervisor
University of Halle,
Institute of Computer Science
D-06099 Halle/Saale
Germany
Phone: +49 (0) 345 55 24728 (-24711)
Fax: +49 (0) 345 55 27033
Email: posch@informatik.uni-halle.de

**Dr. Ing. Nils Jungclaus**
Former member of SFB360
Perfact Innovation GmbH
D-33615 Bielefeld
Germany
Phone: +49 (0) 521 968 792 62
Fax: +49 (0) 521 968 792 66
Email: nils@perfact-innovation.de


Bielefeld, 12/01/2003

## Papers

[1] K. Koch, F. Zöllner, S. Neumann, F. Kummert, and G. Sagerer. Comparing bound and unbound protein structures using energy calculation and rotamer statistics. *In Silico Biology*, 2:32, 2002.

## Other Publications

[1] S. Neumann. Bewertung von Dockinghypothesen durch elastisches Matching. Master's thesis, Universität Bielefeld, 1999.

[2] S. Neumann, S. Posch, and G. Sagerer. Towards evaluation of docking hypotheses using elastic matching. In *Computer Science and Biology: Proceedings of the German Conference on Bioinformatics*, page 220, October 1999.

[3] K. Koch, S. Neumann, and G. Sagerer. Towards a protein-protein docking optimized rotamer library. In *GCB 2000, Poster Abstracts*, page 41, October 2000.

[4] K. Koch, S. Neumann, G. Sagerer, and F. Zöllner. Side chain flexibility for 1:n protein-protein docking. 2002. poster 92A, ISMB 2002.

[5] S. Neumann, F. Zöllner, K. Koch, F. Kummert, and G. Sagerer. Elmar: A protein docking system usind flexibility information. 2002. poster 113, ECCB 2002.

[6] F. Zöllner, K. Koch, S. Neumann, F. Kummert, and G. Sagerer. Calculating residue flexibility information from statistics and energy based prediction. 2002. poster 190, ECCB 2002.

# List of Figures

# List of Tables

# Bibliography


[Abdallah98] Abdallah, F., Zimmermann, O., Leven, O., Schneckener, S. and Kim, J. A Relational Macromolecular Structure Database. In D. Schomburg and O. Zimmermann, eds., *German Conference on Bioinformatics, GCB'98*. University of Cologne, 1998.

[Ackermann98] Ackermann, F., Hermann, G., Posch, S. and Sagerer, G. Estimation and filtering of potential protein-protein docking positions. *Bioinformatics*, vol. 14(2):196–205, August 1998.

[Althaus02] Althaus, E., Kohlbacher, O., Lenhof, H.-P. and Müller, P. A combinatorial approach to protein docking with flexible side-chains. *Journal of Computational Biology*, vol. 9:597 – 612, August 2002.

[An98] An, J., Nakama, T., Kubota, Y. and Sarai, A. 3DinSight: An Integrated Relational Database and Search Tool for Structure, Function and Property of Biomolecules. *Bioinformatics*, vol. 14:188–195, 1998.

[Atkins02] Atkins, J. F. and Gesteland, R. Biochemistry: The 22nd Amino Acid. *Science*, vol. 5572:1409–, 2002. ISSN 0036-8075.

[Barber96] Barber, C., Dobkin, D. and Huhdanpaa, H. The Quickhull algorithm for convex hulls. *ACM Trans. on Mathematical Software*, Dec 1996.

[Bermann00] Bermann, H. M., Westbrook, J., Zukang, F., Gilliland, G., Bhat, T. N. *et al.* The Protein Data Bank. *Nucleic Acids Research*, vol. 28:235–242, 2000.

[Bernstein77] Bernstein, F. C., Koetzle, T. F., Williams, G. J., Meyer, E. J., Brice, M. *et al.* The Protein Data Bank: a computer–based archival file for macromolecular structures. *Journal of Molecular Biology*, vol. 112:535–542, 1977.

[Betts99] Betts, M. J. and Sternberg, M. J. An analysis of conformational changes on protein-protein association: implications for predictive docking. *Protein Engineering*, vol. 12(4):271–283, 1999.

[Bourne98] Bourne, P. E. and Shindyalov, I. N. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Engineering*, vol. 11(9):739–, 1998. ISSN 0269-2139.

[Bux03] Bux, W., Borst, S. and Herzog, U., eds. *Performance Evaluation*. Elsevier, 2003.

[Chang01] Chang, C.-C. and Lin, C.-J. Training nu-Support Vector Classifiers: Theory and Algorithms. *Neural Computation*, vol. 13(9):2119–2147, 2001.

[Chen02a] Chen, X., Lin, Y., Liu, M. and Gilson, M. The Binding Database: Overview and User's Guide. *Biopolymers Nucleic Acid Science*, 2002.

[Chen02b] Chen, Z., R. Weng. Docking Unbound Proteins Using Shape Complementarity, Desolvation, and Electrostatics. *Proteins: Structure, Function, and Genetics*, vol. 47(3):281–294, 2002. ISSN 0887-3585.

[Ciaccia97] Ciaccia, P., Patella, M. and Zezula, P. M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. In M. Jarke, M. J. Carey, K. R. Dittrich, F. H. Lochovsky, P. Loucopoulos and M. A. Jeusfeld, eds., *VLDB'97, Proc. of 23rd International Conference on Very Large Databases, August 25-29, 1997, Athens, Greece*, pp. 426–435. Morgan Kaufmann, 1997. ISBN 1-55860-470-7.

[Claussen01] Claussen, H., Buning, C., Rarey, M. and Lengauer, T. FLEXE: Efficient Molecular Docking Considering Protein Structure Variations. *Journal of Molecular Biology*, vol. 308(2):377–395, 2001. ISSN 0022-2836.

[Connolly83] Connolly, M. L. Solvent–Accessible Surfaces of Proteins and Nucleic Acids. *Science*, vol. 221:709, 1983.

[Costanzo01] Costanzo, M. C., Crawford, M. E., Hirschman, J. E., Kranz, J. E., Olsen, P. *et al.* YPD, PombePD and WormPD: model organism volumes of the BioKnowledge library, an integrated resource for protein information. *Nucleic Acids Res.*, vol. 29:75–79, 2001.

[Echols03] Echols, M., Milburn, D. and Gerstein, M. MolMovDB: analysis and visualization of conformational change and structural flexibility. *Nucleic Acids Res*, vol. 31:478–482, 2003.

[Efron97] Efron, B. and Tibshirani, R. The .632+ Bootstrap Estimator. *Journal of the American Statistical Association*, vol. 92(438):548–560, 1997.

[Fernandez-Recio02] Fernandez-Recio, J., Totrov, M. and Abagyan, R. Soft protein-protein docking in internal coordinates. *Protein Science*, vol. 11(2):280–291, 2002. ISSN 0961-8368.

[Fink95] Fink, G. A., Jungclaus, N., Ritter, H. and Sagerer, G. A Communication Framework for Heterogeneous Distributed Pattern Analysis. In *International Conference on Algorithms And Architectures for Parallel Processing*, pp. 881–890. Brisbane, 1995.

[Fink96] Fink, G. A., Jungclaus, N., Kummert, F., Ritter, H. and Sagerer, G. A Distributed System for Integrated Speech and Image Understanding. In *International Symposium on Artificial Intelligence*, pp. 117–126. Cancun, Mexico, 1996.

[Gerstein94] Gerstein, M., Lesk, A. and Chothia, C. Structural mechanisms for domain movements in proteins. *Biochemistry*, vol. 33:6739–7649, 1994.

[Glaser01] Glaser, F., Steinberg, D. M., Vakser, I. A. and Ben-Tal, N. Residue frequencies and pairing preferences at protein-protein interfaces. *Proteins: Structure, Function, and Genetics*, 2001.

[Goede98] Goede, A., Preissner, R. and Froemmel, C. Dictionary of Interfaces in Proteins (DIP). Data Bank of Complementary Molecular Surface Patches. *Journal of Molecular Biology*, vol. 280(3):535–, 1998. ISSN 0022-2836.

[Goldstein89] Goldstein, H. *Klassische Mechanik*. AULA-Verlag Wiesbaden, 10 edn., 1989.

[Grimm02] Grimm, V. *Untersuchung eines wissensbasierten Potentials zur Bewertung von Protein-Protein-Docking Studien*. Ph.D. thesis, Universität Köln, 2002.

[Halperin02] Halperin, I., Ma, B., Wolfson, H. and Nussinov, R. Principles of Docking: An Overview of Search Algorithms and a Guide to Scoring Functions. *Proteins: Structure, Function, and Genetics*, vol. 47(4):409–443, 2002. ISSN 0887-3585.

[Hashimoto94] Hashimoto, A., Nakanishi, M. and Ito, M. An Object-Oriented Database of Protein Structure Data. In *Proc. of the first Int. Conf. on Applications of Databases*, vol. 819 of *Lecture Notes in Computer Science*, pp. 336–350. Springer, June 1994.

[Henrick98] Henrick, K. and Thornton, J. PQS: a protein quaternary structure file server. *Trends in Biochemical Sciences*, vol. 23(9):358–361, 1998. ISSN 0968-0004.

[Hermann02] Hermann, T., Krause, J. and Ritter, H. Real-Time Control of Sonification Models with an Audio-Haptic Interface. In R. Nakatsu and H. Kawahara, eds., *Proc. of the Int. Conf. on Auditory Display*, pp. 82–86. Int. Community for Auditory Display, Int. Community for Auditory Display, 2002.

[Holm93] Holm, L. and Sander, C. Protein Structure comparison by Alignment of distance Matrices. *Journal of Molecular Biology*, vol. 233:123–138, 1993.

[Horack97] Horack, J. Protein Crystal Growth. `http://science.nasa.gov/msl1/pcg_why.htm`, 1997.

[Janin03] Janin, J., Henrick, K., Moult, J., Eyck, L. T., Sternberg, M. J. E. *et al.* CAPRI: A Critical Assessment of PRedicted Interactions. *Proteins: Structure, Function, and Genetics*, vol. 52(1):2–9, 2003. ISSN 0887-3585.

[Jiang02] Jiang, F., Lin, W. and Rao, Z. SOFTDOCK: understanding of molecular recognition through a systematic docking study. *PROTEIN ENGINEERING -OXFORD-*, vol. 2002, Vol. 15, T. 4, S. 257-264, 2002. ISSN 0269-2139.

[Jones97] Jones, G., Willet, P., Glen, R. C., Leach, A. and Taylor, R. Development and Validation of a Genetic Algorithm for flexible Docking. *Journal of Molecular Biology*, vol. 267:727–748, 1997.

[Kendrew56] Kendrew, J. C. and Parrish, R. G. The crystal structure of myoglobin. III. Sperm-whale myoglobin. *Proc. Roy. Soc. London*, vol. A238:305, 1956.

[Klein96] Klein, T., Ackermann, F. and Posch, S. viwish: A visualisation server for protein modelling and docking. *Gene–COMBIS*, vol. Gene 183:GC51–GC58, 1996.

[Koch00] Koch, K., Neumann, S. and Sagerer, G. Towards a protein-protein docking optimized Rotamer library. In *Poster Abstracts of the German Conference on Bioinformatics*, p. 41. Oct 2000.

[Koch01] Koch, K., Zöllner, F. and Sagerer, G. Building a new Rotamer Library for Protein-Protein Docking using energy calculations and statistical approaches. Tech. Rep., TR-2001-03, Technical Faculty, Bielefeld Univeristy, 2001.

[Koch02a] Koch, K., Neumann, S., Sagerer, G. and Zöllner, F. Side chain flexibility for 1:n protein-protein docking. 2002. Poster 92A, ISMB 2002.

[Koch02b] Koch, K., Zöllner, F., Neumann, S., Kummert, F. and Sagerer, G. Comparing bound and unbound protein structures using energy calculation and rotamer statistics. *In Silico Biology*, vol. 2:32, 2002.

[Koch03] Koch, K. *Statistical analysis of amino acid side chain flexibility for 1:n Protein-Protein docking*. Dissertation, Technical Faculty, Bielefeld University, 2003.

[Kohlbacher01] Kohlbacher, O., Burchardt, A., Moll, A., Hildebrandt, A., Bayer, P. and Lenhof, H.-P. Structure prediction of protein complexes by a NMR-based protein docking algorithm. *Journal of Biomolecular NMR*, vol. 20:15–21, 2001.

[Kriegel03] Kriegel, H.-P. and Schönauer, S. Similarity Search in Structured Data. In *Proc. 5th Int. Conf. on Data Warehousing and Knowledge Discovery*. Prague, 2003.

[Moont99] Moont, G., Gabb, H. A. and Steinberg, M. J. E. Use of Pair Potentials Across Protein Interfaces in Screening Predicted Docked Complexes. *Proteins: Structure, Function, and Genetics*, vol. 35(3):364–, 1999.

[Nicholson00] Nicholson, D. IUBMB Metabolic Pathways Chart. Sigma Chemical Co., St.Louis, MO, 2000.

[Norrel99] Norrel, R., Petrey, D., Wolfson, H. J. and Nussinov, R. Examination of Shape Complementarity in Docking of Unbound Proteins. *Proteins: Structure, Function, and Genetics*, vol. 36(3):307–317, 1999.

[Orengo97] Orengo, C., Michie, A., Jones, S., Jones, D., Swindells, M. and Thornton, J. CATH-A Hierarchic Classification of Protein Domain Structures. *Structure*, vol. 5(8):1093–1108, 1997.

[Pearlstein96] Pearlstein, R. and FitzGerald, P. PDB–at–a–glance. `http://cmm.info.nih.gov/modeling/pdb_at_a_glance.html`, 1996. Link 11.12.2001.

[Phizicky03] Phizicky, E., Bastiaens, P. I. H., Zhu, H., Michael, S. and Fields, S. Protein analysis on a proteomic scale. *Nature*, vol. 422:208–215, March 2003.

[Ponstingl00] Ponstingl, H., Henrick, K. and Thornton, J. M. Discriminating Between Homodimeric and Monomeric Proteins in the Crystalline State. *Proteins*, vol. 41:47–57, 2000.

[Royer01] Royer, C. A. *Proteins*, chap. Protein Interactions. biophysics.org, 2001.

[Sagerer94] Sagerer, G. Vorlesung Technische Informatik, 1994.

[Sali03] Sali, A., Glaeser, R., Earnest, T. and Baumeister, W. From words to literature in structural proteomics. *Nature*, vol. 422:216–225, March 2003.

[Sander94] Sander, C. and Holm, L. The FSSP database of structurally aligned protein fold families. *Nucleic Acids Research*, vol. 22(17):3600, 1994.

[Sanger77] Sanger, F., Air, G. M., Barrell, B. G., Brown, N. L., Coulson, A. R. *et al.* Nucleotide sequence of bacteriophage $\Phi$ X174 DNA. *Nature*, vol. 265:687–695, 1977.

[Schäffer01] Schäffer, A. A., Aravind, L., Madden, T. L., Shavirin, S., Spouge, J. L. *et al.* Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. *Nucleic Acids Research*, vol. 29(14):2994–3005, 2001.

[Stone01] Stone, J. E., Gullingsrud, J., Schulten, K. and Grayson, P. A System for Interactive Molecular Dynamics Simulation. In J. F. Hughes and C. H. Sequin, eds., *2001 ACM Symposium on Interactive 3D Graphics*, pp. 191–194. ACM SIGGRAPH, New York, 2001.

[Stroustrup98] Stroustrup, B. *The C++ Programming Language*. Addison-Wesley, Reading, Massachusetts, USA, 1998.

[Stryer94] Stryer, L. *Biochemistry*. W. H. Freeman, 1994.

[Thornton97] Thornton, J. M. and Jones, S. Analysis of Protein-Protein Interaction Sites using Surface Patches. *Journal of Molecular Biology*, vol. 272:121–132, 1997.

[Tomira99] Tomira, K., M., Hashimoto, K., Takahashi, T. S., Shimizu, Y., Matsuzaki, Y. *et al.* E-CELL: software environment for whole-cell simulation. *Bioinformatics*, vol. 15(1):72–84, 1999.

[Totrov94] Totrov, M. and Abagyan, R. Detailed ab initio prediction of lysozyme-antibody complex with 1.6 Å accuracy. *Nature Structural Biology*, vol. 1:259–263, 1994.

[Vajda02] Vajda, S., Vakser, I. A., Sternberg, M. J. and Janin, J. CAPRI: Critical Assessment of PRediction of Interactions. *Proteins: Structure, Function, and Genetics*, vol. 47(4):444–446, 2002.

[Vakser99] Vakser, I. A., Matar, O. G. and Lam, C. F. A systematic study of low-resolution recognition in protein- protein complexes. *PROCEEDINGS- NATIONAL ACADEMY OF SCIENCES USA*, vol. 1999, Vol. 96, Nr. 15, S. 8477 -, 1999. ISSN 0027-8424.

[Venter01] Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J. *et al.* The Sequence of the Human Genome. *Science*, vol. 5507:1304–1351, 2001. ISSN 0036-8075.

[Walters86] Walters, D., Pearlstein, R. and Krimmel, C. A Procedure for Preparing Models of Receptor Sites. *J. Chem.*, vol. 63:869–872, 1986.

[Walters95] Walters, D. E. When There is No Receptor Crystal Structure: Building Useful Models of Receptor Sites. http://www.netsci.org/Science/Compchem/, August 1995.

[Waszkowycz01] Waszkowycz, B., Perkins, T. D. J., Sykes, R. A. and Li, J. Large-scale virtual screening for discovering leads in the postgenomic era. *IBM SYSTEMS JOURNAL*, vol. 2001, Vol. 40, T. 2, S. 360-378, 2001. ISSN 0018-8670.

[Watson53] Watson, J. D. and Crick, F. H. C. Molecular Structure of Nucleic Acids. *Nature*, vol. 171(4356):737, April 1953.

[Weiner86] Weiner, S. J., Kollmann, P. A., Nguyen, D. T. and Case, D. A. An all atom force field for simulations of proteins and nucleic acids. *Journal of Computational Chemistry*, vol. 7:230–252, 1986.

[Zhang97] Zhang, Z., Lee, C.-H., Mandiyan, V., Borg, J.-P., Margolis, B., Schlessinger, J. and Kuriyan, J. Sequence-specific recognition of the internalization motif of the Alzheimer's amyloid precursor protein by the X11 PTB domain. *EMBO Journal*, 1997.

[Zöllner02] Zöllner, F., Neumann, S., Koch, K., Kummert, F. and Sagerer, G. Calculating Residue Flexibility Information from Statistics and Energy based Prediction. In *European Conference on Computational Biology 2002, Poster Abstracts*, pp. 275–276. Saarbrücken, October 2002.

[Zöllner03] Zöllner, F., Neumann, S., Koch, K., Kummert, F. and Sagerer, G. IPHEX: A System for Evaluating Protein Docking Hypotheses Using User Feedback. In *European Conference on Computational Biology 2003*, pp. 214–216. Paris, October 2003.

[Zubay93] Zubay, G. *Biochemistry*. WCB, 1993. ISBN 0-697-14267-1.

## Acknowledgements

I wish to express my sincere thanks to both of my supervisors, Prof. Gerhard Sagerer and Prof. Franz Kummert for their help in planning and executing this work. The confidence Gerhard Sagerer had in the work requires no elaboration.

Over the time the interdisciplinary bioinformatics strike force – Kerstin, Frank, Matthias, Markus, Michaela, Petra and Thomas – grew to a respectable size and provided feedback and know how on both biology and computer science. The cooperation I received from other members of the group is gratefully acknowledged, not limited to, but especially to my proof-readers.

Thanks to my wife Steffi for the moral support she provided throughout my work. Her patience was tested to the utmost during the last weeks. Our son Jannick was so kind to wait until this work went to press.

Finally, I would like to thank all whose direct and indirect support helped me completing my thesis, and to the DFG (German research community) for funding the project.

Steffen Neumann