Robust Search Methods for Rational Drug Design Applications

by

Bashir S. Sadjad

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2009

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

The main topic of this thesis is the development of computational search methods that are useful in drug design applications. The emphasis is on exhaustiveness of the search method such that it can guarantee a certain level of geometric accuracy. In particular, the following two problems are addressed: (i) Prediction of binding mode of a drug molecule to a receptor and (ii) prediction of crystal structures of drug molecules.

Predicting the binding mode(s) of a drug molecule to a target receptor is pivotal in structure-based rational drug design. In contrast to most approaches to solve this problem, the idea in this work is to analyze the search problem from a computational perspective. By building on top of an existing docking tool, new methods are proposed and relevant computational results are proven. These methods and results are applicable for other *place-and-join* frameworks as well. A fast approximation scheme for the docking of rigid fragments is described that guarantees certain geometric approximation factors. It is also demonstrated that this can be translated into an energy approximation for simple scoring functions.

A polynomial time algorithm is developed for the matching phase of the docked rigid fragments. It is demonstrated that the generic matching problem is NP-hard. At the same time the optimality of the proposed algorithm is proven under certain scoring function conditions. The matching results are also applicable for some of the fragment-based *de novo* design methods.

On the practical side, the proposed method is tested on 829 complexes from the PDB. The results show that the closest predicted pose to the native structure has the average RMS deviation of 1.06 Å.

The prediction of crystal structures of small organic molecules has significantly improved over the last two decades. Most of the new developments, since the first blind test held in 1999, have occurred in the lattice energy estimation subproblem. In this work, a new efficient systematic search method that avoids random moves is proposed. It systematically searches through the space of possible crystal structures and conducts search space cuts based on statistics collected from the structural databases. It is demonstrated that the fast search method for rigid molecules can be extended to include flexible molecules as well. Also, the results of some prediction experiments are provided showing that in most cases the systematic search generates a structure with less than 1.0Å RMSD from the experimental crystal structure. The scoring function that has been developed for these experiments is described briefly. It is also demonstrated that with a more accurate lattice energy estimation function, better results can be achieved with the proposed robust search method.

Acknowledgements

It is always difficult to list all the people that should be acknowledged for their role in achievements of any big project. My thesis is no exception and I am sure that the list of names below is not complete. I like to thank Simulated Biomolecular Systems for supporting this work. I always wanted to do my Ph.D. in a more practical setting compared to a pure academic work. SimBioSys provided this opportunity and I am grateful for that. I have benefitted significantly from the countless discussions with my co-supervisor at SimBioSys, Zsolt Zsoldos. He is one of those few real interdisciplinary people who can understand chemistry very well while being a good programmer. I like to thank my other colleagues there as well for their help, especially Aniko Simon, Mahdi Mirzazadeh, Darryl Reid, Orr Ravitz, James Law, Sing Yong Khew, Antony Williams, and A. Peter Johnson.

At the School of Computer Science of University of Waterloo, I was lucky to do my Masters with Timothy Chan in Computational Geometry. Although he was not involved directly in my Ph.D. but some of the results of my Masters research directly influenced the design and implementation of both projects discussed in this thesis. I like to thank my co-supervisor at school, Ming Li, for his support especially in making this industrial setting for my Ph.D. possible. I thank Ali Ghodsi, Babak Alipanahi, and other members of the Bioinformatics group for their comments. I also benefited from comments made by my Ph.D. advisory and thesis committees members, Brendan McConkey (Biology), Forbes Burkowski (Computer Science), Brian Ingalls (Applied Math), and I am especially thankful to Nicolas Moitessier (Chemistry, McGill), for his careful reading and comments.

I am grateful for my family who have been great supporters in difficult times of my life. My wife, Masoomeh, was a great patron especially during the difficult periods in my graduate studies. She left many opportunities behind and came with me to Canada for my graduate studies which I will remember forever. My parents were my first role models; I will never forget the hard working of my father and sacrifices of my mother.

There have been countless friends in my life and especially in my graduate studies who have influenced my thinking. It is not possible to name them all but I am very grateful for their companionship. One of the best fruits of having such friends and the countless discussions with them was to be critical about all aspects of life, either scientific or not.

And last but not least, there was a little buddy who made the last year of my studies full of joy. He is my son, Rashed, one of the best gifts that God has given us.

Dedication

To my wife, Masoomeh, and to my parents

... and to the martyrs who stood up against oppression in Iran recently.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis summarizes some of my contributions in the development of computational tools for rational drug design. These contributions are collectively labeled as *robust search methods* which reflects the philosophy behind the development of these methods, as described below.

I joined Simulated Biomolecular Systems (SimBioSys) after doing my Masters research in Computational Geometry. I started learning about the mechanisms of actions of drugs and the drug discovery process by working on the *docking* tool eHiTS [140, 141]. A docking tool is supposed to predict how a drug molecule binds to a biological target (usually a protein) as described in Section 1.1. Like many problems in rational drug design, the search space is huge and the *scoring function* is a non-convex goal function with many local minima. The philosophy behind development of eHiTS was to exhaustively traverse the search space. Chapter 2 of this thesis, which is also going to be published as a separate paper [110] deals with this problem. The new algorithms that are designed and implemented to address some of the shortcomings of eHiTS are described in Chapter 2. A docking software is a useful tool for the scientists working in the discovery stages of the drug design process. Specifically, such a tool can help in the *Hit Identification* stage by ranking the drug candidates based on their estimated binding affinities and in the *Lead Optimization* stage by helping the medicinal chemist in modifications of a drug candidate or a family of drug candidates. High Throughput Screening methods are used for differentiating between actives and non-active compounds in massive scales. As described in Section 1.1, making such differentiations with a computer program, or a Virtual High Throughput Screening tool, means significant savings of resources for a drug design project.

Chapter 3 of this thesis is the report of the design and implementation of a new search method for prediction of crystal structures of drug-like molecules. The main contributions of this chapter are also included in another paper [109]. This tool which is called eCrySP is also designed according to the same philosophy as in eHiTS, i.e., exhaustive search and avoiding stochastic methods. There are several modules that are shared between the docking and crystal prediction tools, namely perceiving chemical properties of molecules, the fragmentation process of flexible molecules, surface calculation, components of scoring functions, etc. However the most important common part that is one of the contributions of this thesis is the very fast method for sampling of the placement of two rigid objects besides each other such that there is no clash between these objects and the contact surface between them is beyond a certain threshold. This method is described in details in Section 2.3 in the context of the docking problem, followed by corresponding accuracy proofs. It is also demonstrated how such an idea can be useful in crystal structure prediction as well in Section 3.1.

As described using some important examples in Section 1.2, certain properties of a drug molecule that is marketed as a crystalline solid depends on the placement pattern of molecules in the crystal structure and the lattice energy. The most important of which are probably solubility and dissolution rate. These properties are usually more important in the later stages of drug development rather than earlier discovery stages. However, more and more, drug companies are realizing that it is important to look into these properties earlier in the drug design pipeline [49]. In fact poor solubility or dissolution rate is an important cause for drug failures in stages after Lead Optimization. These kind of failures are expensive because significant resources are already spent for such a candidate. Therefore it is important to include optimization of these properties in earlier stages like Lead Optimization. This is where a system similar to High Throughput Screening is needed for crystal structure determination and in fact examples of such a machinery do exist [49]. Having a computer program that can predict crystal structures reliably is similar to a docking tool in the context of binding affinity prediction. Another important goal of a crystal structure prediction tool is to predict possible polymorphs of a drug candidate as described in Section 1.2.

It is noteworthy that in implementation of both parts of this thesis, there were several software components developed in SimBioSys that have been used but not mentioned explicitly. Examples are processing of different input formats like PDB or Mol2, splitting of a target protein and a bound ligand in a PDB file, etc. It is obvious that such a foundation was very useful in implementing the ideas proposed in this thesis. On the other hand there are basic functionalities that have been implemented during the course of this project and now are being used by other projects in SimBioSys. For example the design and implementation of a component to handle space groups was done during the crystal structure prediction project and now is used in the visualization tool CheVi as well.

In the rest of this chapter we look at some of the previous works that are related to the contributions of this thesis. For the docking problem some of the related publications are mentioned in Section 1.1 with the focus on search algorithms. This is a huge area and by no means we claim that a complete review of the field is done here, instead some useful surveys written by experts in the field are mentioned. In the case of crystal structure prediction, the main focus is on the search methods proposed for this problem and the state of the art is reviewed in Section 1.2. Later in Chapter 3 we show that our new search method is novel and significantly different than the methods proposed so far for this problem.

1.1 Protein-Ligand Docking

As mentioned above, one of the key areas that computational methods can help the drug discovery process is the identification of lead compounds and the lead optimization process. A standard procedure for finding active compounds for a known biological target is High-Throughput Screening (HTS) of a library of thousands or millions of small molecules (the ligands). This procedure filters out a significant number of ligands that are unlikely to have high binding affinity to that target. Although HTS is very useful, it requires fairly expensive labs to screen practical size libraries in a reasonable time frame [12, 73].

A Virtual High-Throughput Screening (VHTS) tool is a computer program that has the ultimate goal of replacing an HTS lab, although currently it is mainly used as a complementary tool. Such a tool differentiates between active and non-active compounds, possibly by predicting the binding affinities; and different approaches exist for making such predictions. One rational approach is to simulate the thermodynamic effects of the target environment and find the conformation and position of the ligand that minimizes a function approximating the changes in the *free energy of binding*. Finding that binding configuration is called *docking.* The biological target could be a protein or a nucleic acid but we are mainly interested in proteins, although in theory, the proposed methods are applicable to other targets as well. There are also different types of protein targets such as enzymes and membrane receptors. Therefor, for simplicity, we use the term receptor to collectively call all these biological targets.

The prediction of the ligand binding modes presents two problems: First, the changes in the free energy of binding as a function of the protein-ligand configuration should be approximated with an acceptable accuracy. These approximations are usually called *scoring functions.* There are many previous and ongoing attempts to develop accurate scoring functions. A thorough comparison of nine such functions has been conducted by Ferrara et al. [43]. They used a set of 189 protein-ligand complexes. Another comparison among 37 different scoring functions is carried out by Warren et al. [132]. In fact, there are several other similar comparative studies of scoring functions and docking methods as listed by Moitessier et al. [89]. Developing an accurate and fast scoring function is a very difficult problem because accurate quantum mechanics level calculations are not feasible for such large molecular complexes and it is very difficult to make the right approximation decisions to simplify such calculations. Examples of shortcomings that are common among many of these scoring functions are the modeling of the entropy or the solvent effects [89, 79]. Detailed descriptions of scoring function features is beyond the scope of this thesis. However some common terms such as the repulsion part of the Lennard-Jones potential are employed [80]. A common form of a simple *empirical* scoring function is given in Section 2.2. For the practical experiments in Section 2.5, the scoring function developed for the eHiTS docking package [140, 141] is chosen.

The second issue in developing a docking tool is a reliable optimization method that can find the global minimum of a given scoring function. This is also a difficult problem because there are usually a huge number of local minima in the search space. The primary focus in Chapter 2 of this thesis is this optimization problem.

The ligand conformation, together with its positioning relative to the receptor, is called a ligand *pose.* In fact, docking is the problem of finding the best pose. Note that the best pose is the global minimizer of the scoring function. Due to the approximations in that function, the best pose is not necessarily close to the native pose which is discovered by experimental methods such as X-ray scattering techniques or Nuclear Magnetic Resonance (NMR) spectroscopy.

One of the very first docking programs was DOCK, in the early 80s [78] which was treating both ligand and receptor as rigid objects. Since then many docking tools have been developed, in fact over 60 of them with corresponding publications are listed by Moitessier et al. [89]. Other examples of good reviews of the docking methods and scoring functions used in them are the work of Kitchen et al. [73], Sousa et al. [118] and Taylor et al. [121]. The review of both search algorithms and scoring functions in docking by Halperin et al. [61] is also noteworthy, however it goes beyond drug-like ligands and addresses other types of docking as well.

From the search algorithm perspective, a categorization is given in a previous paper [141]. The range is from stochastic and purely heuristic search methods, including simulated annealing (MCDOCK [82], AutoDock2 [56]), genetic algorithms (GOLD [68, 127, 126], AutoDock3 [92]), and other stochastic approaches (ICM [6, 7, 123]), to more directed and deterministic ones such as DOCK [41, 42, 94], FlexX [106, 105], and eHiTS [140, 141]. There are also methods that combine heuristics with a systematic search, for example Glide [45, 60, 46] do a systematic search which is followed by a Monte-Carlo minimization. Another example of hybrid methods is the three tools DAIM-SEED-FFLD used together. In this framework, the ligand is fragmented into mainly rigid fragments (DAIM [74]), then three anchor fragments are selected and docked independently (SEED [86]). In fact this systematic docking is done in preprocessing for different types of fragments. Eventually with a genetic algorithm the whole molecule is reconstructed and optimized (FFLD [21, 23]).

No accuracy level is guaranteed for the stochastic search methods used in docking, e.g., simulated annealing and genetic algorithms. In contrast, the goal on the methods proposed in this thesis is to exhaustively search the pose space by using fine sampling. Several docking methods have the same approach. However, to be exhaustive, major interactions should not be missed; and this implies certain requirements for the sampling procedure that is not satisfied by many of the current docking methods [141]. In particular, a 15+ degree sampling of the dihedral angles is too crude for an exhaustive docking algorithm. For example in FlexX which is a well established fragment-based docking method that uses incremental construction to handle ligand flexibility, a set of dihedral angles that are most common in crystal structure of small molecules are used to sample the conformational space. However this approach has two drawbacks, first there are examples that a small error in sampling a dihedral angle of the ligand will result in missing of a key interaction, e.g., the ligand of PDB code 1CX2 as described in [141]. Another problem is that in some cases a dihedral angle of the ligand might be far from the angles in optimized conformations and that is because interactions with the receptor environment compensate for it; again as described in more details previously [141]. Glide [45] also suffers from these sampling problems because it relies on initial ligand conformations that are minima of the ligand conformational landscape. Even the flexibility handling idea of recent versions of DOCK [42, 94] in which an anchor rigid fragment is first docked and then the other fragments are added incrementally suffers from this drawback. This is because during the reconstruction, partial poses are ranked and those with bad energy values are removed. However the bad energy contributions can later be compensated by other fragments, which is not modeled as described below.

To overcome the problem of the large pose search space, a well-known ligand fragmentation method [39] also known as the *place-and-join* method [118] is adopted in Chapter 2. The idea is to split the input ligand into rigid fragments and solve the sampling problem in two steps: Rigid docking of the fragments, generating many acceptable poses, and matching the poses to reconstruct close to optimal poses for the entire ligand. Note that this method is fundamentally different than incremental fragment-based methods in which a base fragment is docked and other fragments are added incrementally. It is well known that in the ligand native pose, some of the fragments might be in far from optimum positions to compensate for the placement of other fragments; such positions are not considered in incremental methods. For example this problem was observed many years ago in the development of a *de novo design* program called SPROUT [55] and the above place-and-join method was chosen partly because of that experience [141].

A geometric shape descriptor structure is developed to model the molecular surface. The advantage of this approach is reflected in Section 2.3. Although the proposed descriptor structure with its properties is unique, many other attempts have been made to use geometric descriptors to model ligands and receptors; for example, (i) using spheres of different sizes to model the empty space of the binding site cavity followed by a matching with ligand spheres [115, 41]; (ii) the method of Fingerprints for Ligands And Proteins (FLAP) by Baroni et al. [13] in which the points of possible energetic interactions are marked on both the receptor surface and the ligand, then a geometric matching is done between quadruples of these points; (iii) the shape descriptor of Weisel et al. [133] that is a grid based method mainly used to find the possible binding sites of a receptor. Although used for a different purpose, but the underlying idea of Weisel et al. is similar to the one used in the present work in the sense that they also use vectors in different directions to measure the empty space around a grid point inside receptor cavities (see Section 2.3).

It is noteworthy that the ultimate docking solution should deal with the flexibility of both the ligand and the receptor. However, the receptors was assumed to be rigid in most of the major docking methods [68, 92, 106, 45] until a few years ago [46, 89, 10], and it is the approach that is followed in most of this work, as well. This approach is, in fact, the first step to solve the final problem. If one cannot predict the native ligand configuration with the native receptor structure in the bound state, there is not much point in addressing protein flexibility.

Teague offers an interesting review of protein conformational changes in the ligand binding process [122]. As it was mentioned, this problem is yet to be addressed properly in current docking software [79]. One method for addressing this problem is applied by Sherman et. al. [114]. This approach uses the Glide docking package and the protein structure determination package Prime. The main drawback of this approach is that ligand binding and protein flexibility are modeled in independent steps. In another approach, Glide is extended to address protein flexibility to some extent by reducing the energy penalty of clashing atoms [46]. A similar method is also applied by Ferrari et al. using DOCK [44]. They compare this method with the method of using multiple receptor conformations. Another method is to use a grid that represents the binding site cavity of different conformations of the receptor which is called an ensemble of receptors. One example of such methods is the approach of Sotriffer and Dramburg [117]. AutoDock3 was also used in a similar grid based method [98]. In this method a weighted average of the energy grids built for different protein conformations was used. The ensemble of receptors may come from different NMR or crystal structures of the same receptor or be the result of taking snapshots in molecular dynamics simulations, like the approach of Amaro et al. [10].

Another way to model receptor flexibility is to extend the optimization variables to the ones modeling receptor conformations, e.g., binding site side chain dihedral angles. This was the approach from the early versions of ICM [123], however its search method is mainly a random walk through the search space which is a huge space with many local minima even with a rigid receptor. Moitessier et al. have extended a genetic algorithm approach by including variables modeling receptor flexibility [90, 32]. In their approach, receptor conformation is modeled discretely in the sense that a set of possible conformations for side chains and the backbone are combined. FlexX is also extended to FlexE to address the problem of receptor flexibility [30]. In this approach possible conformations for binding site patches (side chains or backbone loops) are extracted from different receptor structures and are combined during the docking of the flexible ligand. Frimurer et al. tried to generate many receptor conformations by changing the side chain conformations in the binding site. They then used FlexX to dock a ligand to all of these conformations and selected the minimum energy receptor-ligand pairs [47]. Of course one major drawback of this approach is its resource requirement in terms of CPU time. In other words for m receptor conformations the running time is multiplied by m. One idea to decrease this linear running time factor is to include the choice of the receptor conformation as an optimization variable [67]. Although if the search algorithm is exhaustive this will not remove the linear dependency [67]. The methods and ideas for including multiple receptor conformations in docking were recently reviewed by Totrov and Abagyan [124]; an older review is also done by Carlson [22].

In Section 4.1 it is shown that the general docking method of Chapter 2 can be extended to include flexible side chains as well. Modeling the full flexibility of the protein is very difficult and, in some sense, a connection to the folding problem. This occurs because, in some cases, the binding of the small ligands can cause a significant change in the protein conformation. One such example is the binding of the drug trifluoperazine, with the Ca^{2+}-calmodulin protein [122]. One benefit of our proposed method is that it models ligand and protein side chain flexibility simultaneously.

Some of the ideas of Section 4.1 are implemented and tested on a few targets with known flexible side chains. These results are mainly for a proof of concept and more investigation is left for future works. Note that in most of the cases protein conformational changes upon drug binding are limited to a few side chains. In fact, as it is shown in the statistical study of Najmanovich et al. [95], in 85% of the cases, the protein conformational changes are limited to three side chains only.

Two final introductory notes on docking: First, a completely different set of approaches to estimate affinities exist that are *ligand-based*. These methods do not require any structural information about the binding site. The only data that is provided is a list of active molecules and their affinities. The goal is to rank the input library of the ligand molecules, based on their similarity to the actives, and estimate their affinity that way. One example is a joint work of the present author [107]. However this kind of approach is beyond the scope of this thesis. Secondly, it is evident that the ligand molecules here are small drug-like molecules. This is significant because the term, docking, is used in other contexts as well, for example, the binding of two proteins. To decide what molecules are drug-like, we follow some of the criteria determined by Lipinski et al. [81] called *Lipinski's rule of five*. These are rules of thumb that are observed in many orally active drugs. One of these rules limits the molecular weight to 500 daltons or less. It is noteworthy that these constraints also apply to the scope of the search method we have developed for crystal structure prediction, which is discussed in Chapter 3.

1.2 Crystal Structure Prediction

The other major contribution of this thesis is the new search method that is proposed and implemented for the prediction of crystal structure of drug-like molecules. Many of the properties of a crystal structure are determined by the arrangement of the molecules in that structure. For example, the solubility or dissolution rate of a drug, marketed as a crystalline solid, might be changed, if it crystallizes in a different form [33, 66]. A typical example of such an effect was seen in the production lines of *ritonavir*, an inhibitor of HIV-1 protease in the 1990s. In 1998, after two years of being on the market, some of the ritonavir capsules failed the dissolution test due to a new crystal form which was not known at the time of the drug approval. The new form, called Form II, was less soluble and more stable than the original Form I. This effect was soon propagated to other production lines of ritonavir, and eventually, after an expensive process, a new formulation of ritonavir was submitted to the FDA for approval [28].

Most solid drugs are marketed as crystalline solids rather than amorphous solids [33], and it is crucial to determine all the possible crystal forms, i.e., all the polymorphs and their lattice energies. The crystal structure and the lattice energy can be used to predict physical properties of a drug, e.g., solubility and dissolution rate which are important factors in the drug bioavailability [66, 49, 131]. Although there are interesting attempts to predict the solubility of a drug from its molecular structure [138, 113, 131, 112, 130] but there are many examples that show an accurate estimation of the lattice energy is needed for such predictions. The ritonavir case is one example. Other interesting examples are shown by Hancock and Parks [62]; in one particular case the solubility of an amorphous form of indomethacin falls well below the initial solubility after an hour because of formation of the crystalline solids in the solution. Examples of solubility differences of up to 4 folds between different crystalline forms are also surveyed by them and by Pudipeddi and Serajuddin [103].

Experimental methods for crystal structure determination, such as X-ray scattering techniques, are time consuming and expensive. Our goal here is to develop a computational method to predict possible crystal structures of a molecule. This problem, known as crystal structure prediction (CSP), has had a rather long history of improvements. From the early arguments in the late 1980s about the difficulties in making such predictions [85, 31, 63], even calling the failure to make such predictions a "continuing scandal" [85], to the success of the latest blind test of crystal structure prediction [35], it has been a long way. This progress has been facilitated by improvements in two major areas: (i) the emergence of better models for estimating the lattice energies, and (ii) better search methods, coupled with the significant increase in computational power which made it possible to search the structure space more thoroughly.

It should be noted that the general approach in CSP is to find the structure that minimizes the lattice energy (or free energy), i.e., the most thermodynamically stable crystal formation. In other words, the kinetic effects of the crystallization process are usually ignored. These kinetic effects are important and the presence of polymorphs formed under different crystallization conditions is a reason to question the above approach. In fact, polymorphism is a more common phenomenon than it is traditionally perceived. For example, Stahly has shown that 50% of organic molecules used in 245 polymorph screens exhibit polymorphism [119]. Also, other studies have been conducted to determine cases that this approach of lattice energy minimization is not successful and perhaps kinetic effects have to be considered [34]. However, it is believed that different crystal forms of an organic compound should have close lattice energies [102]. Also, there are many reports on the success of the lattice energy minimization approach, including the four blind tests of CSP, hosted by the Cambridge Crystallographic Data Center [35, 37, 93, 83]. Therefore, this lattice energy minimization approach has been adopted in the design of our CSP search method. This method is presented in Chapter 3 and is called "electronic Crystal Structure Prediction", or *eCrySP* for short.

Our key target in this area has been to design and implement a new *search method* that is more robust than existing methods. The development of a new energy function has been only secondary. However, it is clear that an accurate model for estimating lattice energies is an essential part of any successful CSP project. In fact, most of the new developments since the first blind test [83], have been in the area of energy estimation models. A simple model similar to the W99 force field [134, 135, 136], has been developed as the default scoring function of eCrySP. Models similar to W99 force field, which consists of a point charge model and a 6-exp or 6-12 component modeling orbital overlap repulsion and attractive dispersion forces, have been used in many CSP experiments [37]. More elaborate ideas have also been tested, including the use of multipoles instead of point charges [36, 19, 137], fluctuating charges modeling the polarization effects in the crystalline environment, instead of fixed charges [20], quantum-mechanical methods [20, 97, 88], the use of many "pixel" charges to model the electron distribution [51, 52, 53], and hybrid methods combining two or more of the above categories [71]. It is also important to include the conformational energy for flexible molecules [38]. The approaches employed by one of the most successful groups in the fourth blind test of CSP [35] even included force field parameterization based on the input molecule, called a "tailor-made" force field [96]. Implementation and improvements of these energy calculation techniques is beyond the scope of this thesis since the main contribution of eCrySP is its search algorithm. In Section 3.2 we briefly describe our approach for selecting a W99-like scoring function. More details about statistics collection from CSD toward improving this scoring function is given in Section 4.2.

Beside the energy estimation function, the other key part of any CSP project is the choice of a search method. The responsibility of this module is to find the global minimum of the lattice energy landscape (or other close local minima). Different approaches have been used for this search problem ranging from random structure generation coupled with local minimization to systematic approaches. Since our main focus is on this subproblem, the search methods used by the four more successful groups in the third blind test, CSP2004 [37], as well as two other systematic approaches, are described here with some details. These methods cover the basic ideas of most of the methods that are currently used in CSP. For a survey of these and other methods see the review of Verwer and Leusen [128] or CSP1999 report [83]. The report of CSP2007 (the fourth blind test), published recently, shows that the search methods are not fundamentally different than CSP2004. It is interesting that many of the groups simply used different variations of random search strategies [35].

From the three categories in CSP2004, the only successful predictions (i.e., a correct structure in the first three submissions) are in the simplest category of small rigid molecules. The four successful groups are Day et al. [34] for both molecules in this category; and van Eijck [125], Karamertzanis and Pantelides [69, 70], and Bazterra et al. [14, 15, 16] for only one of the molecules.

The method of Day et al. [34] is dependent on the Polymorph Predictor module of the Accelrys Cerius^2 software package. The principal contribution of Day et al. is the choice of the scoring function and the work they have done on that front. Polymorph Predictor is the descendant of one of the very first successful methods for predicting crystal structures by Karfunkel and Gdanitz [72, 54]. Several other participants in CSP2004 and CSP2007 have adopted this tool. The search method of Polymorph Predictor is a *simulated annealing* approach, which is a greedy down-hill method with a temperature-dependent probability for taking up-hill steps to avoid trapping in local minima.

Bazterra et al. employed a *genetic algorithm* approach to search the crystal structure space [14]. The idea of a genetic algorithm is to simulate the genetic evolution. Each structure is coded as a vector and using crossover and mutation operators, the vectors evolve in relation to their energies. Those with smaller energy values have a higher chance to survive.

Simulated annealing and genetic algorithms are heuristic methods without any accuracy guarantee. In fact, to make sure that all the relevant low energy structures are generated the search has to be repeated several times [101]. There are also other examples of stochastic methods used for crystal structure prediction. For example the approach of Pillardy et al. is an extension of simulated annealing in which instead of one structure, a family of structures are maintained [99]. Again no level of accuracy can be guaranteed for the structures found.

The search method of van Eijck [125] stems from an earlier tool developed in the mid 1990s, called UPACK [91] which uses a grid sampling. As described with details in Section 3.1, given a rigid molecule and a fixed space group, there are 12 parameters that should be determined to define a crystal structure. UPACK samples these parameters by using a 12-dimensional grid. The drawback of this approach is that to achieve a reasonable accuracy, a fine enough grid should be selected, but in that case the search is usually prohibitively slow. Such methods can guarantee a certain accuracy level. The developers of UPACK have conducted a comparison with Polymorph Predictor and have shown comparable performances [91].

The method of Karamertzanis and Pantelides [69, 70] is also based on sampling the search space, similar to UPACK. However, instead of using an exhaustive grid-based sampling, they use the sampling method of Sobol [116]. This sampling is similar to a random sampling of the search space but relies on a deterministic sequence. These authors had interesting observation about the estimated number of local minima of the scoring function. For the four small rigid molecules that they have examined, the number of these minima is in the range of several tens of thousands [69].

All of these approaches use general purpose search strategies, e.g., simulated annealing, genetic algorithms, and grid sampling. There are two other approaches that are noteworthy because they are more specific to crystal structure generation and use insights from crystal packing patterns and the crystallization process. The first is PROMET, by Gavezzotti. It is based on the nucleation phase of the crystallization process. Pairs or clusters of molecules with strong interactions are built and are extended to full crystal structures [50]. The other method is MOLPAK by Holden et al. [65]. In this approach the patterns of molecules in the neighborhood of a central molecule are analyzed in many crystal structures in the Cambridge Structural Database (CSD) [9]. For each space group, frequent patterns are extracted and applied to guide the systematic crystal structure generation.

The eCrySP approach is described in Chapter 3. The shape descriptor method to model molecular surface that was originally developed for docking is modified and used by eCrySP. The sampling method based on these descriptors is a key step of eCrySP. Some of the important assumptions of the search method are based on statistical observations of structures in CSD. It is outlined how such statistics, collected from thousands of crystal structures, can help prune the search space without losing low-energy structures. This structural database, or similar ones[1] can also be used to adjust the parameters of the force fields, as explained later in this thesis. A significant property of eCrySP is that the conformation flexibility is modeled during both the sampling stage and local optimization, rather than during the final local optimization only.

[1] One example of such databases is CrystalEye [40] which is updated automatically. However, the quality of the data stored in CSD is superior.

1.3 Guaranteed Geometric Accuracy

Both in the case of the docking problem and the crystal structure prediction, we assert that deterministic systematic search methods that guarantee a certain level of accuracy are superior to stochastic and heuristic methods, because of the complexity of the energy surface. It is true that the scoring functions used in structure prediction problems usually have many local minima and so it is very difficult to come up with analytical methods for finding the global minima of such energy surfaces. However, this is exactly an indication that off-the-shelf optimization methods which are usually heuristics will not be able to solve these problems. Examples of these methods are simulated annealing, genetic algorithms, and tabu search. The most important philosophy throughout the development of the computational methods presented in this thesis is to do systematic searches that can guarantee a certain level of accuracy. In other words they should be able to guarantee that they can find the global minima of the scoring function within a certain approximation threshold. Although this is the ultimate goal but working directly with energy values is too difficult. Instead we have tried to guarantee a certain level of *geometric accuracy*. In Section 2.2 we have justified that a geometric approximation of the global minimizer structure can lead to an energy approximation too.

The following generic definitions describe the meaning of a guaranteed geometric accuracy in structure prediction problems:

Definition 1 *Given an ordered set of n atoms $A = (a_1, a_2, \ldots, a_n)$, an* atomic configuration *is an ordered set of 3D coordinates $P = (p_1, p_2, \ldots, p_n)$, i.e., p_i is the coordinates assigned to a_i in configuration P.*

Definition 2 *Given two atomic configurations $P = (p_1, p_2, \ldots, p_n)$ and $Q = (q_1, q_2, \ldots, q_n)$ of the same n atoms, they are said to be ψ-*close *iff*

$$\max_{1 \leq i \leq n} \{||p_i - q_i||\} \leq \psi.$$

In the context of drug binding with a fixed receptor binding site, P and Q are two positions for the same drug in the protein binding site (Chapter 2). In the context of crystal structures, P and Q consist of one molecule and a certain number of its neighbors in two crystal structures of the same molecule (Chapter 3). The systematic search method for the drug docking problem is mathematically formalized in Chapter 2 based on more specific definitions similar to Definition 2. The above philosophy of exhaustive search is also followed in the development of eCrySP.

It is important to note that the concept of ψ-closeness of Definition 2 is significantly different than root mean square deviation or RMSD that is usually used to assess the quality of structure prediction tools. Following the notation of Definition 1, the RMSD of two configurations P and Q is defined as:

$$\mathrm{RMSD}(P, Q) = \sqrt{\frac{\sum_{1 \leq i \leq n} ||p_i - q_i||^2}{n}}. \tag{1.1}$$

In other words, RMSD is an indication of the *average* error between atom locations in P and Q. However P and Q are said to be ψ-close if the error between atom locations is *limited* by ψ. That is the kind of geometric accuracy that we like to achieve. Of course this is the kind of accuracy that can be formally translated to energy approximation not small RMSD values.

Chapter 2

Predicting the Binding Mode of a Drug Molecule

This chapter is devoted to the protein-ligand docking problem. Because of the accurate algorithmic proofs of this section, the concepts are formalized first in Section 2.1. Then based on these definitions, some new methods are proposed that are applicable to place-and-join docking frameworks. The performance of the proposed methods are shown in practice using an extensive test set of 829 protein-ligand complexes. Geometric accuracy properties and NP-Hardness results are proved as well. Most of the content of this chapter will be appeared in *IEEE/ACM Transactions on Computational Biology and Bioinformatics* [110].

2.1 Definitions and Contributions

Definition 3 *The* binding site *of the receptor is the location in which the ligand molecule is docked. This is sometime called the* cavity *or the* binding site*.*

There are algorithms to determine or predict the binding site but here it is assumed that the binding site is given. For the experimental results of Section 2.5 the binding site is chosen by finding a cavity inside the receptor atoms that are within 7.0 Å of the ligand atoms in the co-crystallized structure. An example of a binding site is shown in Figure 2.1 where the interaction of a sulfonamide drug with the carbonic anhydrase enzyme of PDB

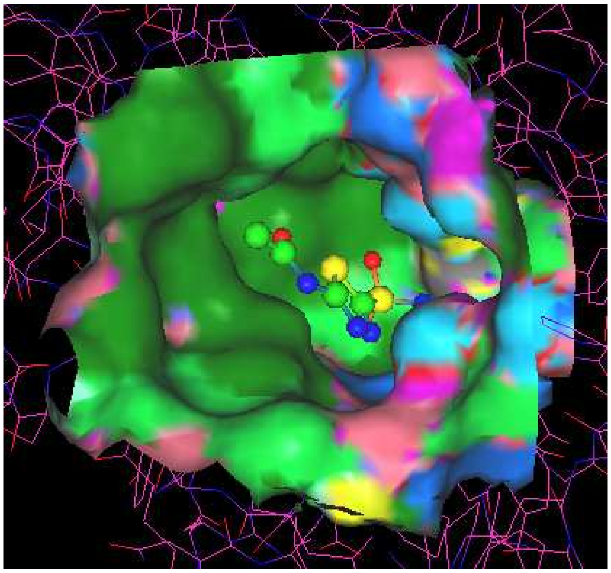


Figure 2.1: An example of a binding site. This is generated from the coordinates of the carbonic anhydrase enzyme of PDB code 1AZM [24] and its interaction with a sulfonamide drug. The receptor surface colors show the chemical features, perceived by the eHiTS scoring function.

code 1AZM [24] is represented. The receptor surface colors show the chemical features, perceived by the eHiTS scoring function.

Definition 4 *A covalent bond is called* rotatable*, if it is single and is not in any cycles. Bonds that are not single in at least one of the resonating structures are not considered rotatable. Any non-hydrogen atom is called a* heavy *atom. A* terminal *atom is a heavy atom that has, at most, one bond to another heavy atom. A molecular fragment is called* rigid *if there is no rotatable bond between any two heavy non-terminal atoms.*

Definition 5 *Any conformation of a molecule, along with a certain position of it in the binding site of the receptor is called a* pose *of that molecule. This can also be called* binding mode *or* binding configuration*. The actual bound pose, determined by experimental methods is called the* native pose *or the* native mode*.*

Definition 6 *Each pose P is a set of vectors $\{p_1, p_2, \ldots, p_n\}$, each representing the coordinates of one atom, where n is the number of atoms. Two poses P and Q are said to be*

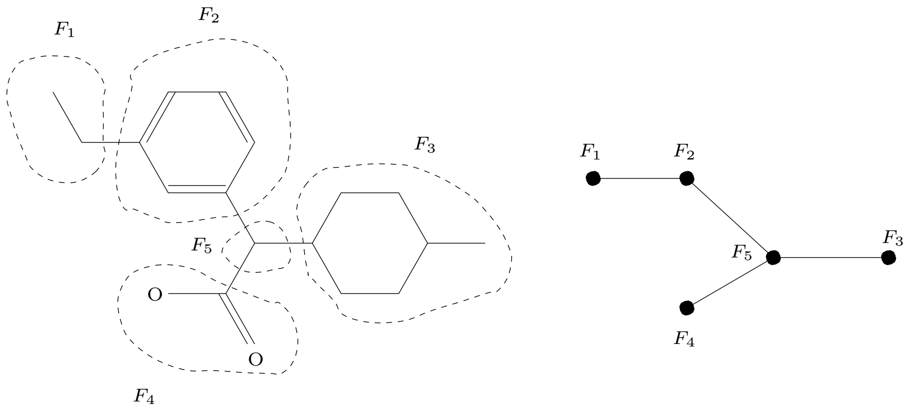


Figure 2.2: **Left:** The input ligand is fragmented into rigid fragments. **Right:** The corresponding graph of rigid fragments, which is always a tree denoted by T^{ligand}.

ψ-close*, which is denoted by* $P \approx_\psi Q$*, iff*

$$\max_{1 \leq i \leq n} \{\|p_i - q_i\|\} \leq \psi, \tag{2.1}$$

in which ψ *is the closeness threshold.*

As mentioned in Section 1.1, the goal is to guarantee a certain geometric accuracy in the search procedure. However, the pose search space is huge for flexible drug molecules. A well-known technique to handle this combinatorial explosion is to split the input molecule into rigid fragments, dock each rigid fragment into the binding site, and reconstruct the plausible poses for the entire molecule. This is called the place-and-join method.

One example of the fragmentation process is given in Figure 2.2. It is easy to see that the graph representing the connectivity of fragments is a tree, denoted by T^{ligand}. For the matching phase, the rotatable bonds connected to each fragment are included in that fragment. These bonds are called *join bonds*, and their atoms are called *join atoms*. Note that the term "pose" might be used for ligand fragments as well as the whole ligand and this should be clear from the context. The docking of rigid fragments is called, **RigiDock**, and the matching of rigid poses, **PoseMatch**. These two steps are the focus of this chapter and are illustrated by a simple example in Figure 2.3.

Fragments or poses are indicated by capital letters like F and P (for notation simplicity, small letters are used for the poses only, starting from Section 2.4). Note that the terms

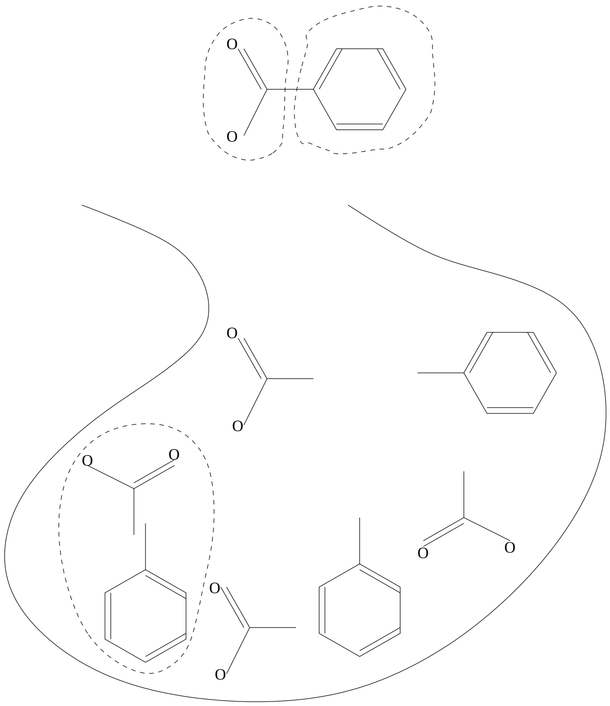


Figure 2.3: An imaginary two dimensional example of the algorithm flow. After fragmentation of the input ligand, the RigiDock step finds many poses for each rigid fragment. The PoseMatch step finds possible matches between the RigiDock output poses.

fragment and pose can be used interchangeably. In fact, the fragment F is also the input pose. Two fragments are called *neighbors*, if they are adjacent in T^{ligand}. The goal of the RigiDock step is to generate a set of poses in the receptor cavity such that at least one of the poses is close enough to the native pose. It is not possible to filter out poses based on their score value at the RigiDock level, because in the native pose, some of the fragments are not at optimal or even near optimal locations. However the significant penalty of any major *steric clash* of a fragment pose with the cavity is enough to reject such a pose even at the RigiDock level; this is described in Section 2.2.

The contributions in this chapter include the following:

1. A very fast method for the RigiDock step with a proven geometric accuracy (Section 2.3). It is also demonstrated how a geometric approximation can be translated into the scoring approximation in Section 2.2.

2. An algorithm for the PoseMatch step and proving its optimality under specific conditions for the ligand and the scoring function (Section 2.4).

3. A proof of the intractability of the PoseMatch problem for the general case (Section 2.6).

4. A thorough assessment of the newly developed method in practice, using 829 protein-ligand complexes from the Protein Data Bank [17] (PDB) which have drug-like ligands (Section 2.5).

An important point about the PoseMatch results is their wide application in structure-based drug design. These results are applicable to any place-and-join method. They also have applications in some of the fragment-based *de novo* design techniques, where different fragments are designed to interact with specific parts of the binding site, and then, these fragments are joined to form full drug molecules [111].

2.2 Geometric versus Scoring Accuracy

A detailed description of the advanced scoring functions is beyond the scope of this work. However, to justify some of the decisions in the search algorithm, scoring function features that are common to most of such functions are used. An *empirical* scoring function is usually a sum of different terms, combined by an appropriate weighting scheme. Some of the common terms follow:

- The effect of electrostatic forces between (partially) charged atoms, using the Coulomb law.
- The Lennard-Jones potential representing the van der Waals attractive and Pauli repulsion forces.
- Other terms for hydrogen-bonding or hydrophobic effects.
- Ligand internal energy, which might be computed by molecular mechanics techniques.

One simple general form is

$$E(M) = \sum_{i \in R, j \in L} k \frac{q_i q_j}{d_{ij}} + \sum_{i \in R, j \in L} \left(\frac{r_{ij}}{d_{ij}^{12}} - \frac{a_{ij}}{d_{ij}^{6}} \right) + E_s(M) + E_l(M), \tag{2.2}$$

in which M represents a binding mode or a ligand pose, R and L are the set of atoms of receptor and ligand, q_i is the partial charge of atom i, the distance between atoms i and j is denoted by d_{ij}, the Coulomb constant is k, and r_{ij} and a_{ij} are the experimentally determined, positive parameters of the Lennard-Jones potential [80]. The ligand internal energy is represented by $E_l(M)$. More sophisticated scoring functions does contain many other components. For example some treat *hydrogen bonds* and *metal interactions* differently than general electrostatic forces. These terms are collectively included under $E_s(M)$ and are out of the scope of the present discussion.

If two non-bonded heavy atoms are too close to each other, the repulsive component of the Lennard-Jones potential, i.e., the r_{ij}/d_{ij}^{12} term in (2.2) will be dominant. In other words, the large penalty of this component prevents any two atoms from being too close. The atoms are usually modeled by spheres with their corresponding van der Waals radii. By using this repulsion property, at the RigiDock step, the generation of any poses with a significant clash with the receptor is prevented. This repulsion property can also be used to justify how a geometric approximation is translated into a score approximation, at least, for simple scoring function terms. For example, consider the electrostatic potential between two atoms, which is represented by $e_{ij} = kq_iq_j/d_{ij}$ in (2.2). Let us say this component is

$e'_{ij} = kq_iq_j/d'_{ij}$ in the native pose. Then, if a pose is generated close enough to the native pose, it is guaranteed that e_{ij} in this pose is δ-approximation of e'_{ij}. To be more precise, a ψ-close pose with $\psi = \delta c$ can guarantee the following:

$$|d_{ij} - d'_{ij}| \leq \psi \Rightarrow |d_{ij} - d'_{ij}| \leq \delta c,$$

$$c \leq d_{ij}, c \leq d'_{ij} \Rightarrow 1 - \delta \leq \frac{e_{ij}}{e'_{ij}} \leq 1 + \delta,$$

in which c is the minimum clash distance between atoms i and j, and so d_{ij} and d'_{ij} are both greater than c.

It is obvious that proving similar approximation factors for the complex scoring functions, used in docking nowadays, is far more difficult. However, intuitively, two poses that are close enough to each other and do not have a significant clash with the receptor should have close score values. In the experiments, the eHiTS scoring function is used, an early version of which is described by Zsoldos et al. [141].

2.3 Rigid Fragment Pose Generation

It is not difficult to develop a brute force method to sample all the rigid body transformations with a certain accuracy level, apply each transformation to the input molecule, and test the resulting poses for steric clashes. However to be accurate enough, millions of poses should be tried, as reported in Section 2.5. This leads to an expensive procedure with many 3D transformation operations, and in a practical sense, impossible to do. Instead, a different method that does not need any transformation to be applied during the pose generation is selected. The idea is to represent the surface of each fragment with a set of *surface vectors*. Each vector measures the distance of the fragment center of mass to the surface in a certain direction. A similar structure is created for many points inside the cavity too (Figure 2.5). By using these structures, each clash check is reduced to several vector length comparisons without yielding any real transformations. A more precise description of this idea is given in Section 2.3.2. Let us first see how the poses are generated using these surface vectors.

2.3.1 Pose Search Space Fine Sampling

As we discussed in the previous section, the goal of the RigiDock step is to generate a pose set Π for each fragment. This set should cover the search space of the rigid body transformations of the fragment, inside the cavity. Covering indicates that for each valid pose P inside the cavity, there is at least one pose in Π that is close enough to P. If the binding site is open (which is usually the case) it is closed at some far enough points and some dummy atoms are assumed at the closings. Therefore it is assumed that the binding site cavity is closed and the inside and outside cavity are well defined. The closing algorithm works as follows: A 3D grid is placed on the receptor and a traversal algorithm from points on the bounding box is started, similar to a Breath-First; this kind of traversal is called a *flood* here. The points that are traversed are those that are not inside receptor atoms, i.e., the flood always stays out of the receptor atoms. The result of this flood is a *depth* value assigned to each grid point. The intuition of this depth value is the length of the minimum path from a grid point to the bounding box without crossing any receptor atoms. Then another similar flood is started from the deepest point D with depth d. For each grid point this flood finds a similar distance from D. The closing of the receptor happens at distance $d + \mathrm{cls}$ where cls is a user-defined threshold. For the experiments of Section 2.5, this threshold is the default value of 4.3 Å.

To be more precise, some notations are necessary. Let B be a rigid body transformation. Without loss of generality, it is assumed that B is the combination of two affine operators: A rotation component $R_{v,\alpha}$ which defines the rotation of angle α around vector v; and a translation component by vector w, denoted by T_w. Therefore, for vector x,

$$B(x) = T_w(R_{v,\alpha}(x)). \tag{2.3}$$

It is also assumed that the origin of the coordinate system is at the center of mass of the input fragment F. Since each pose P is a set of atom coordinates $\{p_1, p_2, \ldots, p_n\}$, then $B(P) = \{B(p_1), B(p_2), \ldots, B(p_n)\}$. Now each pose P inside the cavity is defined by a transformation B where $P = B(F)$. Given ψ, a method is developed that generates a set of poses Π, where for each pose P inside the cavity, $\exists Q \in \Pi : P \approx_\psi Q$. Beside the accuracy guarantee, the other significant feature of the proposed method is its efficiency. A small preprocessing step is performed, other than that the RigiDock solver, does not need any floating point operations to apply the transformations.

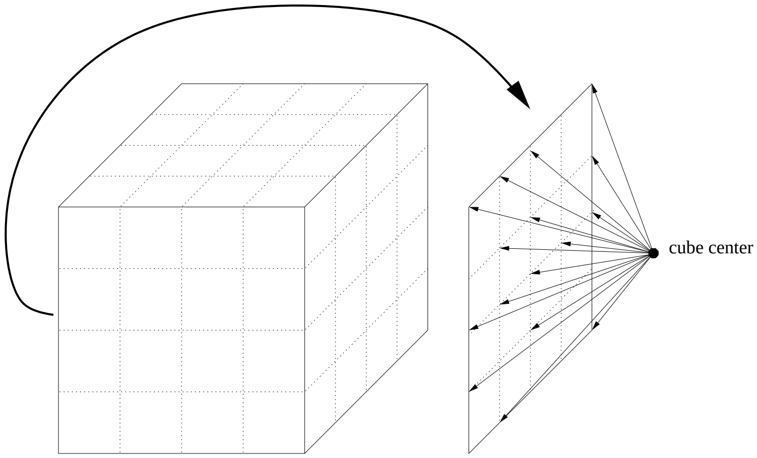


Figure 2.4: The construction pattern of a surface vector set in 3D. A sample of vectors toward the left face of the cube is shown.

First, a general method used in several geometric approximation algorithms [25, 26, 8, 27] is described starting by the following lemma:

Lemma 1 *For any positive $\epsilon \in \mathbb{R}$, there is a set L_d of $\Theta((\frac{1}{\epsilon})^{(d-1)/2}+1)$ vectors in $\mathbb{R}^d$, such that for each vector $v \in \mathbb{R}^d$, the angle between v and some $w \in L_d$ is at most* $\arccos(\frac{1}{1+\epsilon})$*.*

This lemma was first proved by Yao [139] but the wording above is similar to that of [27]. One way to construct such an L_d set is to employ a simple grid-based method [26]. Such a construction in three-dimensional space is depicted in Figure 2.4. Each cube edge is of size two and on each face a two-dimensional grid is placed. The set L_3 consists of all the vectors, originated from the cube center to a grid point. The unit cell size of the grids is chosen such that the conditions of Lemma 1 are satisfied. Assume that the cube faces are perpendicular to the main axes and note that the set of vectors in each of xy, yz, or xz plane is an L_2 set in that plane, satisfying Lemma 1. This property is used later in the proofs. These sets are shown by $L_3(xy)$, $L_3(yz)$, and $L_3(xz)$, respectively. L_d is called a set of *surface vectors* and ϵ is the parameter of L_d. It is clear that the length of the vectors in L_d is not important, and to simplify the argument they are assumed to be normalized.

A rotation around a vector passing through the origin is called a *centered rotation* (or around the origin in $\mathbb{R}^2$). The set of surface vectors are employed to discretize the space of all the centered rotations in 3D. To do the same for the translations space, a 3D grid is used which is created inside the bounding box of the binding site cavity. The unit cell size c of this grid is again dependent on ψ, the closeness threshold (Inequality 2.1). The RigiDock algorithm is shown in Algorithm 2.3.1. The dependency of parameters ϵ and c on ψ is determined by Theorem 4.

```
Data    : An input fragment $F$, a set of surface vectors $L_3$, and a receptor binding
          site.
Result  : A set $\Pi$ of poses covering inside the binding site.
for $\forall v \in L_3$ do
    Let $D$ be any rotation that maps $v$ to $s = (1, 0, 0)$;
    for $\forall w \in L_3(yz)$ do
        Let $S$ be the rotation around $s$ that maps $(0, 1, 0)$ to $w$;
        Let $R = D^{-1}SD$ ;
        for any grid point $g$ do
            Let $T$ be the translation corresponding to $g$;
            Let $P = T(R(F))$;
            if $P$ has no significant steric clash with the receptor then
                Add $P$ to $\Pi$;
            endif
        endfor
    endfor
endfor
```

Algorithm 2.3.1: RigiDock algorithm.

Lemma 2 *Let $v \in \mathbb{R}^2$, and for vector $w \in \mathbb{R}^2$, let R_w be the centered rotation that maps v to w. Then there exists $w \in L_2$ such that $\|R_w(p) - p\| \leq \sqrt{\frac{2\epsilon}{1+\epsilon}}\|p\|$ for any $p \in \mathbb{R}^2$.*

Proof: There exists $w \in L_2$ such that θ, the angle between w and v is, at most, $\arccos(\frac{1}{1+\epsilon})$. Now consider R_w;. It rotates any point by angle θ around the origin. By using the cosine law,

$$\|R_w(p) - p\|^2 = 2\|p\|^2 - 2\|p\|^2 \cos\theta \leq 2\|p\|^2(1 - \frac{1}{1+\epsilon})$$

$$\Rightarrow \|R_w(p) - p\| \leq \sqrt{\frac{2\epsilon}{1+\epsilon}}\|p\|.$$

□

Lemma 3 *For any centered rotation S, there exists a rotation R of the form of Line 1 of Algorithm 2.3.1, where $\|S(p) - R(p)\| \leq 3\sqrt{\frac{2\epsilon}{1+\epsilon}}\|p\|$ for any $p \in \mathbb{R}^3$.*

Proof: Rotation S is characterized by a rotation axis u and an angle α, i.e.i, $S = R_{u,\alpha}$. It is easy to see that in Algorithm 2.3.1, the rotation of Line 1 is, in fact, a rotation around v. Because of Lemma 1, there exists $v \in L_3$ such that the angle between v and u is, at most, $\arccos(\frac{1}{1+\epsilon})$. On the other hand, since $L_3(yz)$ is an L_2 in the yz plane, one of the rotations around v in Algorithm 2.3.1 has angle θ, where $\cos(\theta - \alpha) \geq \frac{1}{1+\epsilon}$. Let us denote this rotation by N. Finally, let O be a centered rotation that maps v to u.

Rotation N is $R_{v,\theta}$ and it is equal to $O^{-1}R_{u,\theta}O$. In other words, for an arbitrary $p \in \mathbb{R}^3$, $N(p) = R_{v,\theta}(p) = O^{-1}R_{u,\theta}O(p)$ and $S(p) = R_{u,\alpha}(p)$, the following inequalities are direct applications of Lemma 2 or its proof ($q \in \mathbb{R}^3$),

$$\|O(q) - q\| \leq \sqrt{\frac{2\epsilon}{1+\epsilon}}\|q\|,$$

$$\|R_{u,\theta}(q) - R_{u,\alpha}(q)\| \leq \sqrt{\frac{2\epsilon}{1+\epsilon}}\|q\|,$$

$$\|O^{-1}(q) - q\| \leq \sqrt{\frac{2\epsilon}{1+\epsilon}}\|q\|.$$

Since rotation is a rigid body transformation (i.e., preserves the distances), the previous three inequalities can be combined by using the triangle inequality such that

$$\|O^{-1}R_{u,\theta}O(p) - R_{u,\alpha}(p)\| \leq 3\sqrt{\frac{2\epsilon}{1+\epsilon}}\|p\|$$

which completes the proof. □

Theorem 4 *In Algorithm 2.3.1, if the set of surface vectors L_3 has parameter ϵ, and the grid T has unit cell size $c \leq \frac{2}{\sqrt{3}}(\psi - 3\sqrt{\frac{2\epsilon}{1+\epsilon}}\Delta)$ in which Δ is the maximum distance of an atom from the origin; then for each pose P, there exists $Q \in \Pi$ such that $P \approx_\psi Q$.*

Proof: Note that for each translation T there exists a translation T', corresponding to a grid point that approximates T with maximum error $\frac{\sqrt{3}}{2}c$, i.e., $\|T(p) - T'(p)\| \leq \frac{\sqrt{3}}{2}c$ for any $p \in \mathbb{R}^3$. With Lemma 3 and the triangle inequality, there exists a rigid body transformation tried in the Algorithm 2.3.1, with maximum error

$$E(\epsilon, c) = 3\sqrt{\frac{2\epsilon}{1+\epsilon}}\Delta + \frac{\sqrt{3}}{2}c. \tag{2.4}$$

Based on the ψ-closeness definition (2.1), this error must be less than ψ to guarantee the closeness claim. This is accomplished by solving $E(\epsilon, c) = \psi$ for c, which gives the c mentioned in the theorem statement. □

The significance of Theorem 4 is in (2.4) which gives an upper bound on the error threshold ψ, based on parameters ϵ and c. As a practical example, for a rigid fragment with $\Delta = 3$Å, if each edge of the cube in Figure 2.4 is divided into 12 segments (which gives $\epsilon \approx 0.0137$) and set $c = 0.6$Å, the maximum difference ψ is smaller than 2Å. This section is brought to a close by the following complexity theorem.

Theorem 5 *For a binding site of volume V and a fragment with farthest atom from the center at distance Δ, a pose ψ-close to the native pose can be found in time $\Theta((\Delta/\psi)^5(V/\psi^3))$, using Algorithm 2.3.1.*

Proof: The proof of this theorem follows from Lemma 1 and Theorem 4, here is the sketch. By setting $c = \psi/\sqrt{3}$ in Theorem 4, $1/\epsilon = \Theta((\Delta/\psi)^2)$. Note that in Algorithm 2.3.1 for each grid point, $\Theta((1/\epsilon)^{1.5})$ poses are tried (Lemma 1). The next section reveals how the check of Line 3 can be done in $\Theta(1/\epsilon)$. Also, $V/c^3 = \Theta(V/\psi^3)$ grid points are tried which completes the proof. □

2.3.2 Efficient van der Waals Filtering

So far the method for choosing the parameters of the RigiDock step, for any certain geometric accuracy, is demonstrated. However, the number of poses that are tried is huge and a brute force implementation of Algorithm 2.3.1 is very slow. In fact, the primary reason for using the surface vectors, to sample the rotation space, is to perform the steric clash checking of Line 3 of Algorithm 2.3.1 without explicitly transforming the rigid fragment.

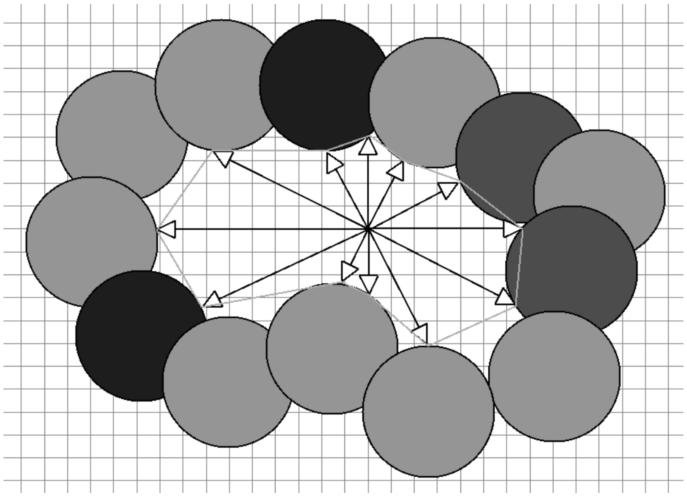


Figure 2.5: A two dimensional view of a cavity descriptor [141].

The steric clash checking step requires preprocessing of the rigid fragment and the cavity. Assume that the set of surface vectors is $L_3 = \{v_1, v_2, \ldots, v_n\}$. Let w_i be the farthest point along v_i that is on the input rigid fragment surface. On the other hand, for each grid point p inside the cavity, a translation is employed to map p to the origin and find the farthest point $w_{p,i}$ along v_i inside the cavity. A schematic two-dimensional view of such a procedure is depicted in Figure 2.5. Calculation of w_i's and $w_{p,i}$'s are done using an approximation method similar to a binary search. The set $\{w_1, w_2, \ldots, w_n\}$ is called the rigid fragment *descriptor*, and $\{w_{p,1}, w_{p,2}, \ldots, w_{p,n}\}$ is called the descriptor of the grid point p.

For a fast steric clash checking, the surface of the ligand and the cavity are approximated by the corresponding descriptors. Assume that the rotation part of a rigid body transformation B in Algorithm 2.3.1 is the identity, i.e., B is a translation of the center of mass of the rigid fragment to a grid point p. To see whether pose $P = B(F)$ clash with the cavity, the length of vectors in the descriptors are compared. An indicator of the extent of clash between the cavity and pose P is $\max_{1 \leq i \leq n}\{\|w_i\| - \|w_{p,i}\|\}$. The bigger this indicator, the bigger the clash. Now, to extend this idea to non-identity rotations, a rotation R is applied to the set of surface vectors and each $R(v_i)$ is approximated by the closest v_j. In other words the vector v_j that has the smallest angle with $R(v_i)$, denoted by $v_{R,i}$, is found. Since the set of rotations in Algorithm 2.3.1 is finite, $v_{R,i}$'s are computed for all rotations

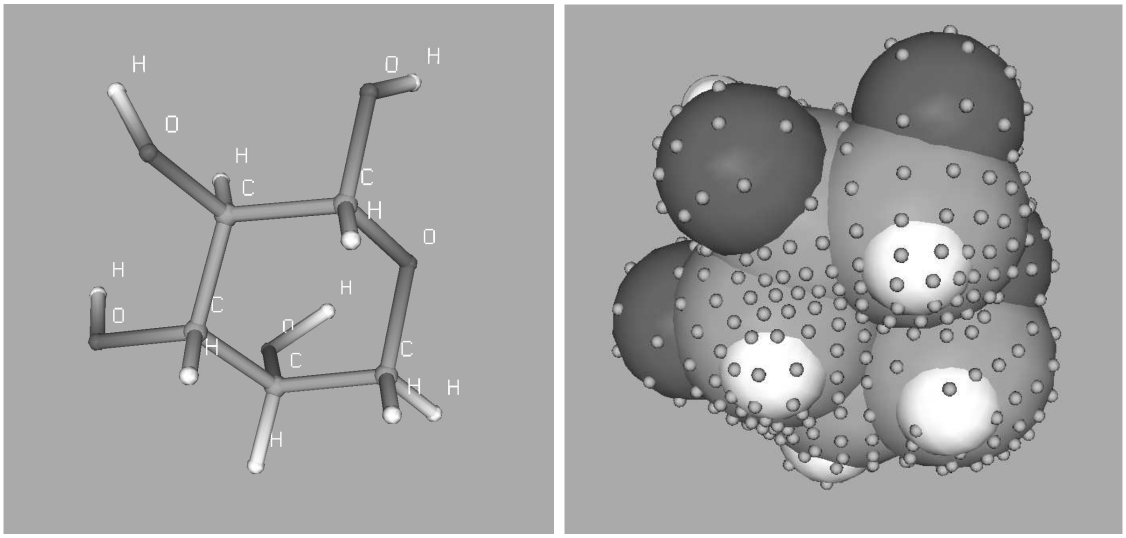


Figure 2.6: **Left:** ball-and-stick model of the L-arabinose molecule. The heavy atom coordinates are extracted from PDB code 1ABE [104]. The hydrogens are added based on the hybridisation of atoms. **Right:** the space-filling model with the end points of the surface vectors highlighted.

R. This clash checking procedure is an approximation and also a little overlap is possible between the ligand and the cavity, therefore a threshold greater than zero is used for the rejection of poses.

To establish an intuition of surface approximation by descriptors, the set of w_i's for a rigid fragment is signified in Figure 2.6. The fragment in the figure is the ligand of the PDB code 1ABE [104] which is treated as a single rigid fragment. The set of surface vectors in this figure is generated by the cube pattern in Figure 2.4 with 9×9 grids on each face, generating an L_3 with $\epsilon \approx 0.023$ and $\arccos(\frac{1}{1+\epsilon}) \approx 12.21°$.

The final note about the proposed algorithm is that in Line 2 of Algorithm 2.3.1, all the grid points are *not* tried in the final implementation. Instead, some statistics are collected from real cases in PDB, on the minimum distance between a fragment and the receptor surfaces. Based on these results, a threshold is chosen for the maximum distance and only poses are tried with an end-point of a surface vector within 2.0 Å of the cavity surface.

The proposed RigiDock method is tested on a set of 829 protein-ligand complexes. The average number of poses tried for each rigid fragment, the number of accepted poses, the accuracy in terms of RMS deviation from the native state, and the run-time of RigiDock are summarized in Table 2.1. More details of the experimental results are given in Section 2.5.

2.4 An Algorithm for PoseMatch

For notation simplicity, small letters such as p are used to indicate the poses in this section and the following sections. One formulation of the PoseMatch problem is as follows:

Problem 1 *Given sets $\Pi_1, \Pi_2, \ldots, \Pi_r$ of poses for each rigid fragment, and a score value $s(p, q)$ assigned to each pair of poses p and q, find a set of* compatible *poses $p_1, p_2, \ldots, p_r$ such that $p_i \in \Pi_i$ and $S(p_1, p_2, \ldots, p_r) = \sum_{1 \leq i \leq j \leq n} s(p_i, p_j)$ is minimized. A set of poses is compatible iff for each pair of neighbor rigid fragments R_i and R_j, the atom distances of join bond(s) of p_i and p_j, are close to the corresponding distance in the original ligand.*

The compatibility condition ensures that a full ligand pose can be constructed from the selected rigid fragment poses without moving them significantly. The acceptable distance error for two neighbor rigid fragments depends on the accuracy of the RigiDock step and can be determined by Theorem 4. A set of compatible poses is called a *matching set*, or simply a *match*. Note that in the above formulation, $s(p_i, p_i)$ is the interaction score between p_i and the receptor. Therefore the summation in Problem 1 is the sum of the intermolecular and intramolecular interaction energies. The ligand internal energy in (2.2), or $E_l(m)$, is the sum of $s(p_i, p_j)$ for $i \neq j$ in the summation of Problem 1, as described below.

Consider the general form of a scoring function in (2.2). For a moment, assume that the $E_l(m)$ component is always zero, i.e., ignore the ligand internal energy. Since the underlying graph of the rigid fragments, T^{ligand}, is a tree, as seen in Section 2.1, the PoseMatch problem can be solved by picking a rigid fragment F as the root of T^{ligand}, solving the problem recursively for all poses of each child node of F, and then for the poses of F, itself, as seen in Algorithm 2.4.1.

To address $E_l(m)$, note that the significant elements in a typical ligand internal energy function are:

1. the stretching energy corresponding to the bond lengths

2. the bending energy of the bond angles

3. the torsional energy of the dihedral angles

4. the interaction energy of the non-bonded ligand atoms.

Since the bond angles and lengths are not changed, the first two components are irrelevant. The torsional energy is consistently between two neighbor rigid fragments. It is later demonstrated that if the torsional energy is the only component in $E_l(m)$, then the minimum match can be found in polynomial time. Note that in the wording of Problem 1, this means that $s(p, q) = 0$ for p and q being poses of two non-neighbor fragments.

The drug-like ligands are small molecules, usually with fewer than 10 rigid fragments. In most cases, there are none or very few non-bonded ligand atoms from two non-neighbor rigid fragments that make significant intramolecular interactions (such as a hydrogen bond). Even in such cases the contribution of those interactions to the final energy value is not substantial due to many other interactions between the ligand and the receptor. This explains why ignoring those interactions remains a good approximation. However, there is one type of interaction which can be as significant as all the other ligand-receptor interactions, and that is the repulsion term of the Lennard-Jones potential. In fact, regardless of all the other interactions, a match in which two non-bonded atoms have significant clash with each other should not be accepted. However, with this condition, the PoseMatch problem is NP-hard, as proved in Section 2.6.

Now, it comes the main heuristic: The same algorithm that finds the exact solution in the case of torsional-only $E_l(m)$ is applied for the general case. However, during the recursive procedure, the poses of each match in each sub-problem are kept, and matches with a severe clash between their poses are rejected. A more precise description is given in Algorithm 2.4.1. A compatible set of poses is considered *valid*, iff no two poses clash with each other.

It is easy to check that without the clash condition of Line 1 and when $s(p, q) = 0$ for non-neighbor poses p and q, this algorithm finds the best match (by applying the typical proof of greedy algorithms). However, with the clash condition, the proposed algorithm can find an approximate minimum.

```
Algorithm: PoseMatch($T$)
Data   : A rooted tree $T$ of rigid fragments $F_1, F_2, \ldots, F_r$ with root $F_1$.
         The set of poses $\Pi_i$ for each fragment $F_i$.
Result : For each $p \in \Pi_1$, the valid set of poses $C_p = \{p, p_2, \ldots, p_r\}$ that minimizes
         $S(p, p_2, \ldots, p_r)$ or $\emptyset$ if no such set exists.
for each child node $F_i$ of $F_1$ do
    Run PoseMatch($T_i$) where $T_i$ is the subtree of $T$ with root $F_i$;
endfor
for $\forall p \in \Pi_1$ do
    $C_p = \emptyset$;
    for each child node $F_i$ of $F_1$ do
1       Let $Q$ be the set of poses $q \in \Pi_i$ which are: compatible with $p$ and $C_q \neq \emptyset$
        and no pose in $C_q$ clash into $p$;
        if $Q = \emptyset$ then
            $C_p = \emptyset$;
            break;
        else
            Let $q = \text{argmin}_{q \in Q} S(p, C_q)$;
            $C_p = C_p \cup C_q$;
        endif
    endfor
    if $C_p \neq \emptyset$ then
        $C_p = C_p \cup \{p\}$
    endif
endfor
```

Algorithm 2.4.1: The PoseMatch algorithm.

2.5 Experimental Results

The experimental results of the RigiDock and the PoseMatch algorithms are presented in this section using the eHiTS scoring function [141]. For each pose of the root fragment in T^{ligand}, Algorithm 2.4.1 finds a match. In this set of matches, the *closest match* is the one closest to the native pose. Also, the closest match to the native pose, in a set of 300 matches selected based on score values and geometric diversity, is compared to the native pose. This is because a few hundred matches should be selected at the PoseMatch level for further local minimization. The final decision on the ranking of these solutions is conducted after the local minimization. This set of size 300 is called the *selected matches*.

In the experiments, the RMS deviation (RMSD) from the native pose is used to measure the accuracy of the proposed method. RMSD is not the best measure. A more important measure is whether the key interactions are similar to the native pose or not; such as the approach of Kontoyianni et al. [75]. Since such measures need a significant amount of manual inspection, RMSD is used for automatic evaluation. It is noteworthy that the RMSD from the native pose of a match, found by Algorithm 2.4.1, is not only a function of the sampling accuracy of our method but also the quality of the scoring function used in Algorithm 2.4.1. In other words, some of the matches, very close to the native pose, might not be selected (or even generated) because of scoring function deficiencies. One extreme case of this is denoted in Figure 2.7 in which two matches are shown for a sulfonamide drug. The native pose is exhibited by thick bonds. Although, the match on the top is much closer to the native pose, the match on the bottom has a better score. (The hidden receptor is the carbonic anhydrase from the PDB code 1AZM [24], the binding site is shown in Figure 2.1.)

In the experiments in this paper, drug-like ligands are the focus. To decide which ligand is drug-like, the *Lipinski's rule of five* is employed, a set of features commonly observed in orally active drugs [81]. 829 ligand-receptor complexes are selected from the PDB, where the ligands are all drug-like. The list of these codes is given in the Appendix. The average resolution of these PDB codes is 1.98 Å with standard deviation of 0.33 Å. Table 2.1 summarizes the results at the RigiDock level. The accuracy and speed in this table are notable.

Table 2.2 contains the results summary of the PoseMatch step. The results show that with the scoring function of eHiTS, one PoseMatch solution can be delivered to the local

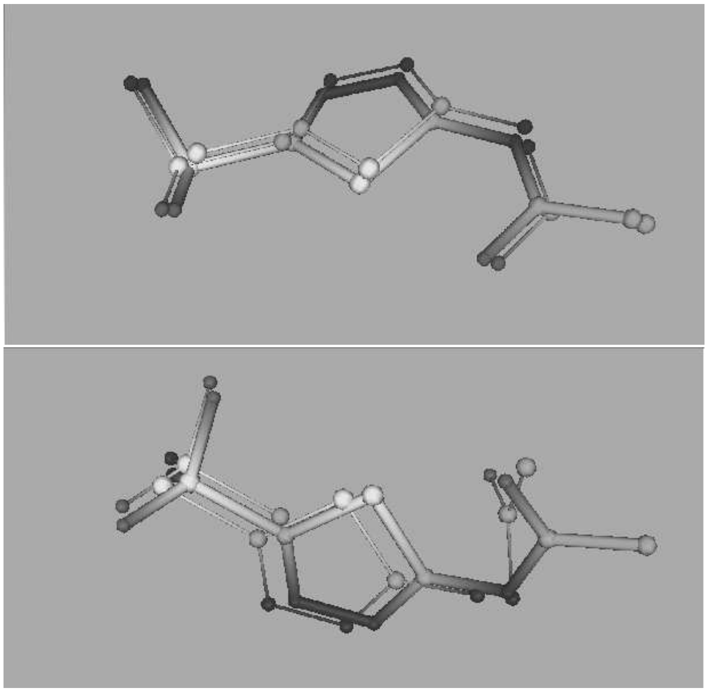


Figure 2.7: Two matching sets of poses are shown for a sulfonamide drug. Native pose is shown with thick bonds. While the match on the top is much closer to the native pose the one on the bottom has a better score due to scoring function deficiencies (the hidden receptor is the carbonic anhydrase from the PDB code 1AZM [24])

#poses tried	#poses accepted	RMSD	run-time	#frag/lig
2.28×10^7	1.46×10^5	0.66 Å	5.0 sec	4.08

Table 2.1: The validation of RigiDock method over a set of 829 protein-ligand complexes. All numbers are averages over the whole set. The first four values are per rigid fragment. The last column shows the average number of rigid fragments per ligand.

#frags	average RMSD (Å) closest match	average RMSD (Å) closest selected	average RMSD (Å) top-rank	run-time (sec)	% of cases
1	0.47	0.76	2.324	3.4	8
2	0.78	1.17	2.372	20.9	15
3	0.91	1.38	2.237	371.4	22
4	1.06	1.58	2.925	792.3	22
5	1.14	1.72	3.658	986.3	12
6	1.43	1.97	3.342	2216.8	10
7	1.57	2.10	4.102	2378.0	5
8	1.64	2.21	4.827	3133.2	3
9	2.11	2.71	6.235	5147.8	2
10	2.58	2.97	3.403	4247.0	1
total avrg.	1.06	1.54	2.94	941.7	100

Table 2.2: PoseMatch results for a set of 829 protein-ligand complexes. Each row shows the averages for ligands with certain number of rigid fragments. The last row shows the overall results.

minimization step, with an average RMSD of 1.54 Å (in a reasonable time). Note that at least 0.5 Å of this is mainly due to the scoring function. This is demonstrated by the RMSD values of the closest match found in the PoseMatch (which is not necessarily selected); the average RMSD here is 1.06 Å. It is also shown that the match with the best score (top-rank) has an average RMSD of 2.94 Å. It is noteworthy that the standard deviation of the closest selected match RMSDs is 0.66 Å while that of the top-rank is 2.47 Å. The jobs were run on a cluster of 114 CPUs. Each CPU is an Intel Xeon 2.40GHz processor, the reported times are measured on such a CPU.

The percentage of cases with the closest match below a certain RMSD value is illustrated in Figure 2.8. Again, note the difference between the closest match found and the one in the selected 300 matches. Another important point, shown in this graph, is the improvement of the PoseMatch output after local optimization. The dotted line in this graph reflects

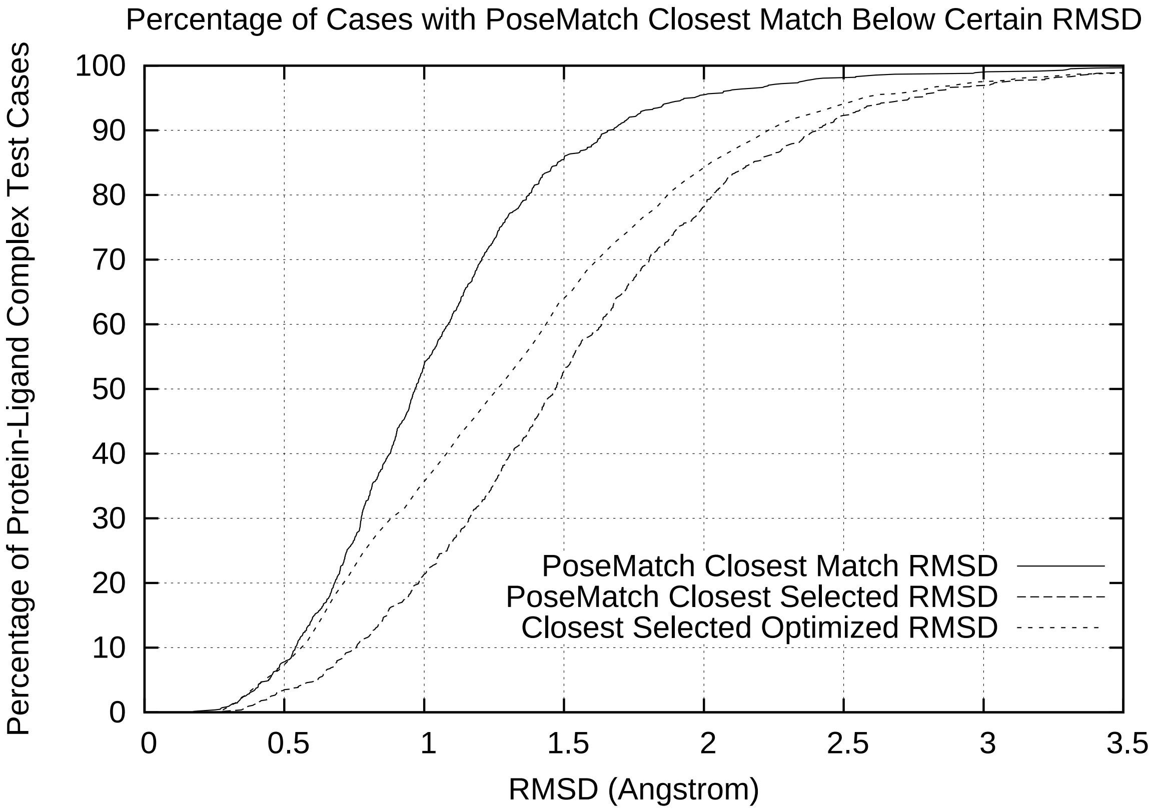


Figure 2.8: The percentage of cases with the closest match below certain RMSD from native pose. Number of selected matches is 300. The local optimization usually brings the PoseMatch closest output closer to the native pose.

the closest final pose after local optimization (among the 300 PoseMatch output poses). The novel heuristic for the PoseMatch problem does not work well for too flexible ligands, as shown in Table 2.2.

After the local minimization a more accurate scoring function is chosen to rank the final outputs. Usually more accurate scoring functions are time consuming and cannot be used at the PoseMatch level with hundreds of thousands of poses for each rigid fragment.

The CCDC/Astex dataset of 305 protein-ligand complexes [1] is the extended version of the dataset that was originally used to evaluate the GOLD program [68] and is a standard test set of docking evaluation. From the above 829 PDB codes, 202 are also in this set. The analysis of the above results over this subset shows that the average RMSD of the closest selected match is 1.55 Å while that of the top-rank is 2.91 Å.

As a minimal comparison with other programs, it is noteworthy that from the 200 PDB codes used in an evaluation of the FlexX docking program [76], 117 codes are among the 829 codes used for the above experiment. Among these codes the average RMSD of the closest selected match and the top-rank match are 1.50 Å and 2.65 Å respectively. In the case of FlexX, the specific results for this subset is unknown but for the whole set of 200, the average RMSD of the top-rank pose is 3.97 Å and that of the closest output pose is 2.16 Å [76]. However it should be emphasized that among the 200 ligands, some are very flexible with more than 10 rigid fragments. Those codes are of course not present in the above dataset of 829 and they are in general more difficult to predict.

2.6 Intractability

In this section, the NP-hardness results are summarized. These results apply to the PoseMatch problem. There are two reasons why the results of this section are important:

1. These results indicate that finding an exact polynomial algorithm to solve Problem 1, with any scoring function having a Lennard-Jones repulsive term, is unlikely.
2. As mentioned in Section 1.1, Problem 1 is a generic problem that arises in any place-and-join method and some of the fragment-based *de novo* design methods. Therefore the results in this section apply to all such methods.

All of the reductions are derived from the well-known NP-complete problem 3SAT, to the target problem. Given a set of boolean variables $x_1, x_2, \ldots, x_n$ and a set of clauses $C_1, C_2, \ldots, C_m$, where each clause is a disjunction of three variables (x_i or $\bar{x_i}$), the 3SAT problem is to decide whether there is a true-false assignment of x_i's such that all the clauses are satisfied (i.e., at least one *true* in each clause). First a decision version of Problem 1 is proved to be NP-complete:

Theorem 6 *Consider the PoseMatch Problem 1. Assume the scoring function s has the simple form that for two poses p and q of two rigid fragments, $s(p, q) = 0$ if they do not clash and $s(p, q) = \infty$ otherwise. Then deciding whether there is a compatible pose set with a finite score value is NP-complete.*

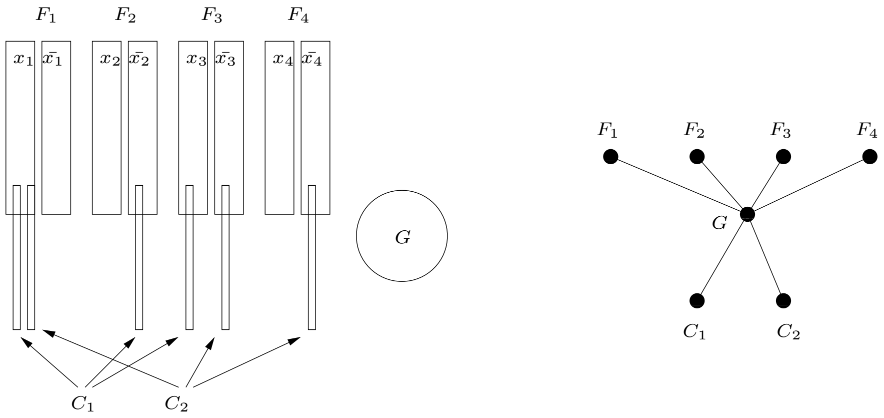


Figure 2.9: **Left:** The set of poses corresponding to the 3SAT problem with variable set $\{x_1, x_2, x_3, x_4\}$ and two clauses $C_1 = \bar{x_1} + x_3 + x_4$ and $C_2 = \bar{x_1} + x_2 + \bar{x_3}$. **Right:** The corresponding graph of rigid fragments T^{ligand}.

Proof: 3SAT is reduced to this PoseMatch problem. Consider an instance of the 3SAT problem with n variables and m clauses. A ligand with $n+m+1$ fragments is constructed. For each variable x_i consider a fragment F_i with two poses; one corresponding to x_i and one to $\bar{x}_i$. For each clause $C_i = y_1+y_2+y_3$ (where y_i is x_j or $\bar{x_j}$ for some j), consider a fragment C_i with three poses p_1, p_2, and p_3, where p_i is clashing with the pose corresponding to $\bar{y}_i$. Also, consider a fragment G with only one pose and assume T^{ligand} consists of all the edges from G to other fragments. One example is demonstrated in Figure 2.9.

Assume that the distance upper bound of the compatibility conditions are all big enough such that all the poses are compatible with the single pose of G. It is easy to verify that the 3SAT instance is satisfiable iff the PoseMatch problem above has a solution with a finite score. □

Note that in the proof of Theorem 6, it can be argued that a few of the properties of the constructed molecule are not generally true for a typical ligand. First, in the compatibility conditions, a large upper bound is used which is not the case for real examples. Secondly, the rigid fragment G has too many neighbors. In a typical rigid fragment connectivity tree, T^{ligand}, the number of neighbors of each node is very limited. In the following theorem it is demonstrated that enforcing the upper bounds on the compatibility threshold and the number of neighbors each node can have, does not change the NP-completeness.

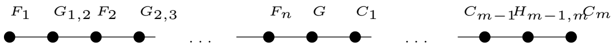


Figure 2.10: T^{ligand} in the proof of Theorem 7.

Theorem 7 *In Theorem 6, the decision problem remains NP-complete even if the maximum degree $\Delta(T^{\text{ligand}})$ is, at most, two and the upper bound limit on the join atom distances of two neighbor compatible poses is arbitrarily small.*

Proof: Here is the proof sketch: The idea is similar to the proof of Theorem 7. However, T^{ligand} is just a path, consisting of $2n + 2m - 1$ fragments. The F_i and C_i fragments (as before) and $n + m - 1$ fragment connecting them exist, as shown in Figure 2.10. The placement of the poses of F_i and C_i is as before. For each connecting fragment G_i or H_i, it is assumed that enough poses exist that can handle any pose selection for its neighbors. For example, for the pose selection of fragments F_i and F_{i+1}, there are four possibilities, therefore four poses for $G_{i,i+1}$ are considered, corresponding to each possibility. To remove the clashes between the H_i poses, the z-axis is used and they are placed at different z-values. □

Remark: Theorem 7 is valid even with the extra condition of convexity of all the rigid fragments.

Finally the next theorem can be proved with a 3SAT reduction, again, by constructing both the cavity and ligand, based on the instance of 3SAT.

Theorem 8 *Deciding whether a flexible ligand fits inside the closed binding pocket of a receptor (such that no two atoms overlap) is NP-complete.*

2.7 Concluding Remarks

In this chapter we proposed new methods for two of the main steps in the place-and-join docking frameworks, namely docking of rigid fragments (RigiDock) and matching of the generated poses (PoseMatch). In the development of the RigiDock and PoseMatch algorithms, the goal was to produce a pose close enough to the native pose, such that, by a local minimization, the native pose can be reproduced. It was shown that the output of RigiDock always contains a pose close to the native pose. The NP-hardness of the PoseMatch problem was also proved. Also, some justification for the polynomial-time heuristic used in PoseMatch was provided plus its optimality under certain conditions. The promising performance of the proposed method was also demonstrated in practice. It is noteworthy that not all the methods explained here are integrated in the current release of eHiTS.

On the practical side the results of an extensive testing of the proposed approach on 829 protein-ligand complexes from the PDB was provided. It was shown that among many poses generated at the PoseMatch level, there is one very close to the native pose with an average RMSD of 1.06Å.

Chapter 3

Predicting the Crystal Structure of a Drug Molecule

In this chapter the new search method for prediction of crystal structures of drug-like molecules is described. This method is called “electronic Crystal Structure Prediction”, or *eCrySP*. In Section 3.1, key steps of the search method are described and supporting statistics are provided. The statistics of this section are collected from the Cambridge Crystal Structural Database (CSD) [9]. The main use of these statistics in this chapter is the pruning criteria that are inferred from them. However in Section 3.2 and later in Section 4.2 it is outlined how these can be used for improving the scoring function as well. Finally the results of the applications of eCrySP for some real examples are demonstrated in Section 3.3. The results are also compared with the structures predicted by the Polymorph Predictor module of Accelrys Inc.’s Materials Studio [2].

3.1 Structure Search Method

In this section, the details of the new search method for crystal structure prediction (CSP) are described. Structure prediction of rigid molecules is explained first and then the extension to flexible molecules is described. Different pruning criteria are used to enhance the performance. Most of these criteria are statistically justified by using data collected from CSD. Several pictures are used to simplify the description.

3.1.1 Scope of the Search

The only internal degrees of freedom included in the eCrySP search method are the rotation around rotatable bonds; all other bond angles and lengths are kept fixed. This implies that the input molecule is assumed to have 3D coordinates. A bond is called *rotatable* if it is an exocyclic single bond between two non-terminal heavy atoms. A heavy atom is called terminal if it has at most one neighbor heavy atom. (The rotation of a terminal group of one heavy atom and several hydrogens is modeled separately during the energy calculation.)

For a rigid molecule, i.e., a molecule without any rotatable bonds, and a given space group, 12 parameters are required to fully determine a crystal structure. There are different ways to choose these parameters and the following has been adopted in this work:

- the base vectors defining the unit cell (3×3 variables)
- the coordinates of the origin (3 variables).

This means that during the search the coordinates of the input rigid molecule is fixed as shown in Figure 3.1 (the picture has been generated by the *mercury* [84] visualization package). This fixed molecule is sometimes called the *central* or the *main* molecule in the rest of this chapter. The experimental crystal structure is sometimes called, the *target* crystal structure here, because that is the structure that the search is supposed to find.

For a flexible molecule, the number of parameters is $12 + r$, where r is the number of rotatable bonds. The r parameters determine the dihedral angles which fix the conformation. In the current implementation of eCrySP, multiple molecular units in the asymmetric unit cell are not modeled, i.e., $Z' = 1$. This also indicates that, currently, eCrySP cannot be used to predict structures of cocrystals, hydrates, etc. The extension to multiple components in the asymmetric unit cell, i.e., $Z' > 1$ is a future work. It should be noted that according to a recent study, more than 72% of the organic crystal structures stored in CSD are single component and about 7% are hydrates [129].

The following notation is used hereafter. The letters a, b, c signify the unit cell base vectors and $\hat{a}, \hat{b}, \hat{c}$ represent their normal vectors. The origin is denoted by o and the individual coordinates of any vector in the Cartesian system are denoted by x, y, z subscripts, e.g., o_x, o_y, o_z. Of course, the same crystal structure can be described by different choices of unit cell vectors. There are standard ways to choose one, such as the *reduced unit*

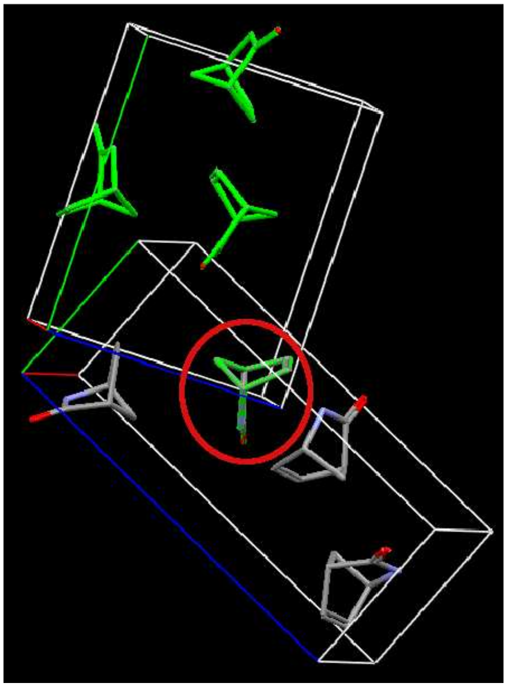


Figure 3.1: Generating crystal structures of a rigid molecule by choosing the base vectors and the origin. Two different structures of the same molecule are shown with the main central molecule fixed in both (the grey structure is CSD refcode RUVZEN [120]).

cell [77, 57]. Different criteria has been followed in eCrySP to eliminate duplicate crystal structures, but for reasons that will become clear later, all the conditions of a reduced cell cannot be imposed. Instead, the conditions of the more relaxed Buerger cells are imposed, and although they do not eliminate duplication problem completely but do reduce it significantly [58].

As mentioned in the Section 1.2, eCrySP is intended to be a systematic search that can guarantee a certain level of accuracy. One simple idea for achieving this goal is to directly sample the $12 + r$ search variables with a grid sampling method. Ideas, similar to this, have been developed before for CSP [91, 69, 70]. However, to ensure that vital interactions are not missed, a very fine sampling is needed, even for rather small molecules, as shown in Figure 3.2. Indeed, the development of eCrySP began with such approaches, for rigid molecules, but soon it was realized that an acceptable level of accuracy requires several days of computation time on a typical CPU, even for rigid molecules. This time scale is unrealistic for practical purposes, especially when it is extended to flexible molecules.

3.1.2 Growing Pairs to Crystal Structures

An alternative to grid sampling is to start with a pair of molecules, and to *grow* that pair to all possible crystal structures. For the main molecule, M, the key observation is that, at least one of its neighbor molecules, say N, in the target crystal structure exists such that

- there is no significant clash between M and N,
- close contact exists between significant surface areas of M and N,

where clash and contact between molecules are defined as intersection or contact of the surface determined by the van der Waals spheres around atoms. Adjusting the van der Waals radii based on CSD statistics will be discussed. The first item of the aforementioned conditions is obvious: A significant clash is associated with high repulsive energy, and hence cannot occur in a realistic target structure. For the second item, quantification of the minimal contact must be provided. Surface contact thresholds have been determined by collecting relevant statistics from CSD (the results of a similar study for the more generic case of a flexible molecule is presented in the following subsection in Figure 3.10).

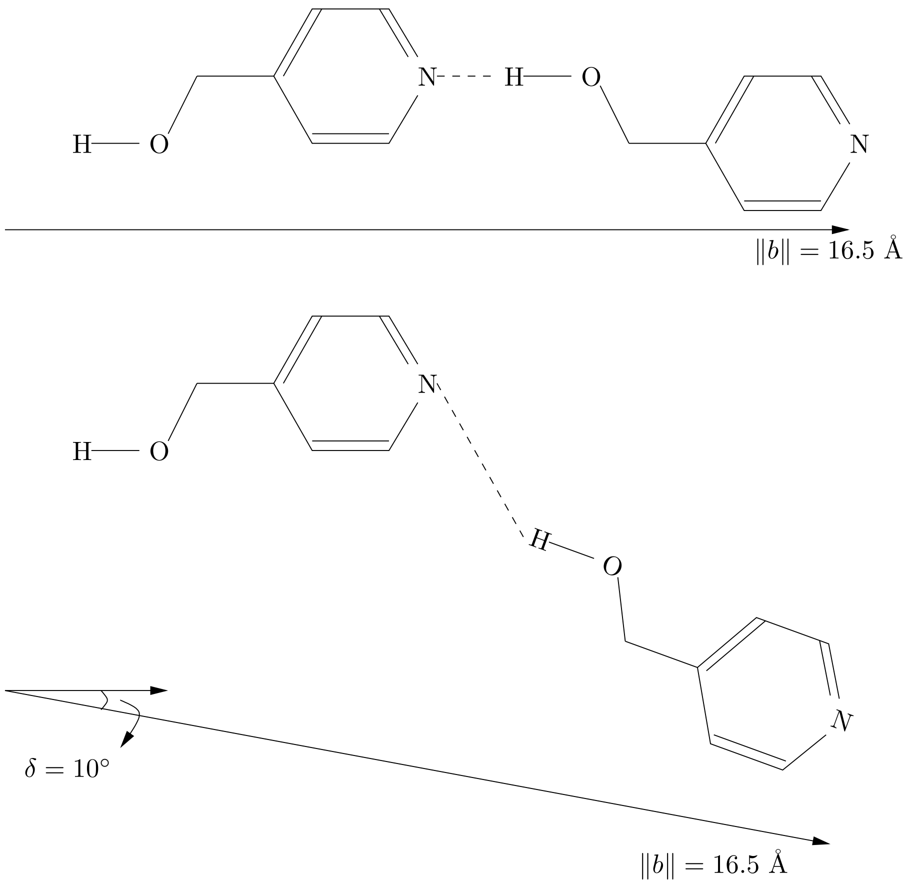


Figure 3.2: The effect of sampling accuracy of base vector angles on key interactions. A 10 degree error in sampling of a base vector can significantly distorts a perfect hydrogen bond in the target crystal structure. This example shows a neighbor generated by a combination of a rotation and a translation.

There are two main reasons for selecting this approach of growing pairs. First, once the information about a pair in a crystal structure is given, some of the parameters of that structure can be inferred. For example Figure 3.3 depicts a neighbor, N, which is a 180° rotation of M. This relative orientation imposes the following constraints on the target structure:

- The space group should have a symmetry operation, consisting of a 180° rotation (compatible screw operations are included).
- One of the base vectors should be parallel to the rotation axis.
- The other two vectors should be perpendicular to this vector.

These constraints will be manifested in the values of $\hat{a}, \hat{b}, \hat{c}$. In addition, once the lattice translation component of the symmetry operation is chosen, some of the base vector lengths can be fixed too. For example, suppose that the base vector b is chosen to be the rotation axis in Figure 3.3. If the corresponding space group operation is a 2_1 screw operation and the lattice translation along b is zero, then the translation along b can only be caused by the $\|b\|/2$ translation of the screw operation. This means that the length of b can be fixed. Finally if, for example, the other two lattice translation components are zero too then two linear constraints result with respect to the possible values of the origin coordinates o_x, o_y, o_z. This is due to the fact that the origin should satisfy the condition of being on the rotation axis.

The second reason for choosing this idea of fixing pairs is that a fast method for sampling *rigid* molecules that are close to each other but do not clash has already been developed, as discussed in Section 2.3. This method was originally designed for the molecular docking software eHiTS [141, 110] and has been instrumental in the development of eCrySP. The key idea is to represent the molecule surface with a set of *surface vectors*, as shown in Figure 3.4. Each vector measures the distance between the molecule centroid and its surface in a specific direction. Similarly a set of vectors is placed on any grid point around the molecule to measure the empty space in its vicinity. One of such grid points is portrayed in Figure 3.4.

A realistic example of the endpoints of surface vectors are shown for a lactam in Figure 3.5 (the coordinates are taken from CSD refcode RUVZEN). The picture is generated

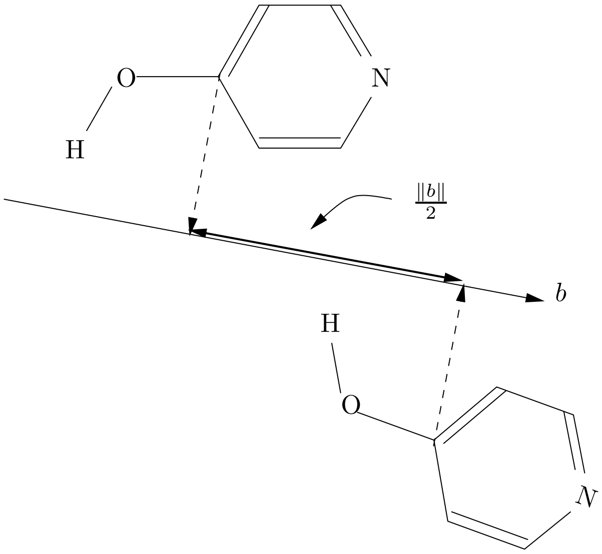


Figure 3.3: For a 2_1 screw symmetry operation, the neighbor molecule transformation forces some of the search variables to take specific values. Here, the rotation axis forces the direction of a base vector. For a given lattice translation, it also forces the length of this vector.

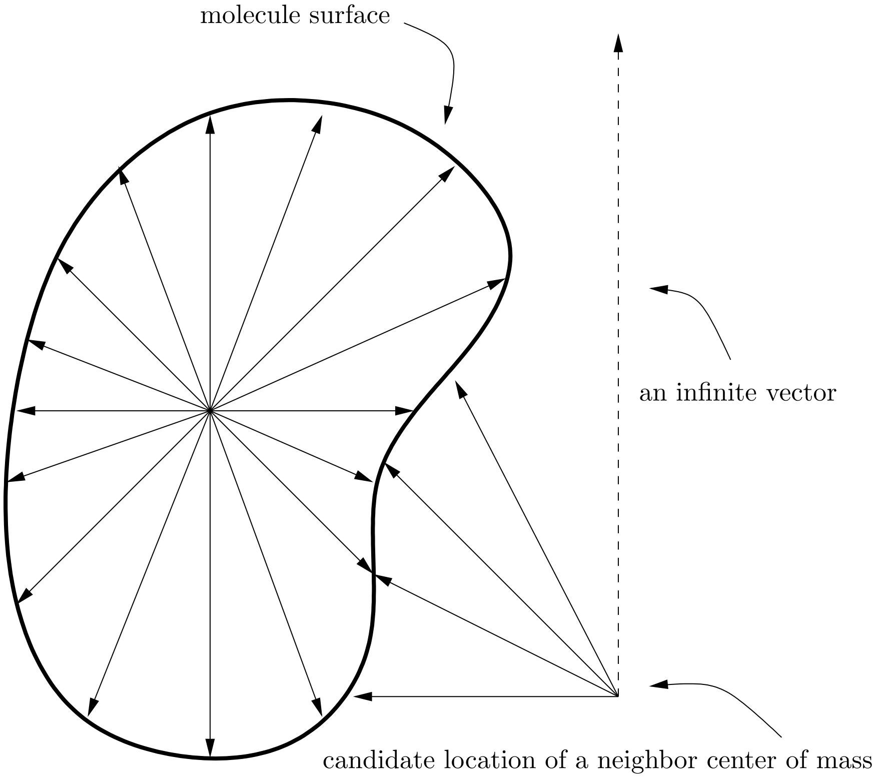


Figure 3.4: A conceptual imaginary 2D example showing the surface vectors representing a molecular surface. The external vectors are computed for a set of grid points around the molecule, one of which is shown here.

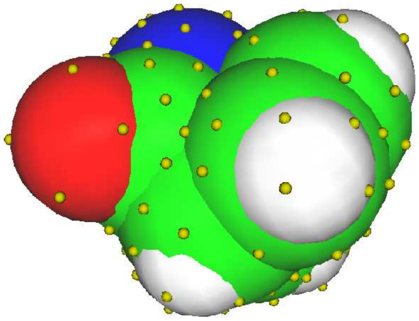


Figure 3.5: The endpoint of surface vectors for a lactam (coordinates from CSD refcode RUVZEN). For the bond structure, see Figure 3.1.

by the visualization package CheVi. The number of vectors depends on the chosen accuracy level. The default accuracy level of eCrySP is used for this example. The bond structure of this molecule is given in Figure 3.1.

For now assume the main molecule M is rigid (the extension to flexible molecules is described in the next subsection). Once these vector measurements are completed, the possible locations for a neighbor molecule are sampled and filtered based on the clash and surface contact criteria mentioned above. Each neighbor molecule, N, in the target structure is generated by a specific rigid body transformation (i.e., a combination of a proper or improper rotation and a translation) of the main molecule, M. To check the clash and the surface contact of N and M, the surface vector lengths should be compared. A fixed set of rotations is chosen to sample the neighbor molecule space. For each rotation R, the mapping between the surface vectors is found in a preprocessing step. In other words, for a surface vector v, the rotation $R(v)$ is computed and v is mapped to the closest surface vector to $R(v)$. This produces a very fast neighbor generator procedure which requires not even a single transformation after the preprocessing step. An example of the mapping between the vectors is shown in Figure 3.6.

Sampling the space of neighbor molecules is done by sampling the space of possible translations and rotations. There is a rigid body transformation B such that for an atom with coordinates x in the main molecule M, the coordinates of the corresponding atom in

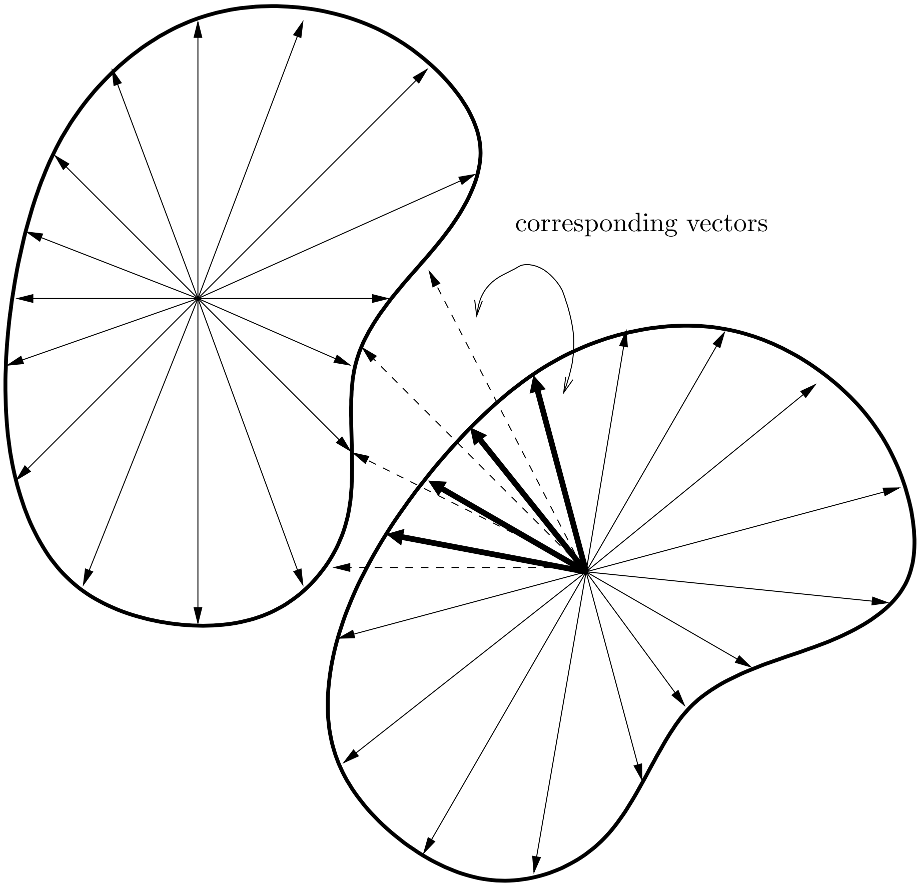


Figure 3.6: The clash and surface contact measurement for a rigid body transformation of the main molecule. After preprocessing, the only check in this step is vector length comparisons. The corresponding vectors are shown by thick or dashed lines. The center of mass is placed at the same grid point illustrated in Figure 3.4.

the neighbor molecule N is $B(x)$ where:

$$B(x) = T_w(R_{v,\alpha}(x)), \tag{3.1}$$

or

$$B(x) = T_w(R'_{v,\alpha}(x)), \tag{3.2}$$

in which T_w is a translation by vector w and $R_{v,\alpha}$ is a rotation around vector v by angle α and R' is an improper rotation. The values of α come from a certain set of angles based on the space group being searched. For example for space group $P2_1$, the only values for α are 0 or 180 degrees. To sample the space of the translations T_w a 3D grid with cell size c is used. To sample the rotation space $R_{v,\alpha}$ the vectors v should be sampled. This is done using the aforementioned surface vectors. These vectors are chosen such that a certain level of accuracy, as defined in Definition 2, can be guaranteed. This is done using the following lemma which is proved to be a useful tool in several computational geometry applications [27, 25, 26, 8]. In fact this is the same lemma that is the basis of accuracy proofs of the docking method in Section 2.3:

Lemma 9 *For any positive $\epsilon \in \mathbb{R}$, there is a set L_d of $\Theta((\frac{1}{\epsilon})^{(d-1)/2} + 1)$ vectors in $\mathbb{R}^d$, such that for each vector $v \in \mathbb{R}^d$, the angle between v and some $w \in L_d$ is at most* $\arccos(\frac{1}{1+\epsilon})$.

This lemma was first proved by Yao [139] but the above modified version is closer to that of [27]. There are simple ways to construct such a set of vectors for a given ϵ [26]. To guarantee the closeness threshold ψ of Definition 2, suitable values of c and ϵ can be chosen as shown in Section 2.3.

Algorithm 3.1.1 summarizes the search method for the rigid molecules. With the above discussion, it is easy to see why the checks of of Lines 1 and 2 of this algorithm are efficiently done. The fine sampling of the crystal forms space produces a large set of candidate structures. A subset of structures is selected at the end of sampling (Line 4) and will be subject to local optimization.

3.1.3 Extension To Flexible Molecules

To extend the previous idea of Section 3.1.2 to flexible molecules, the flexibility is modeled during the pair generation process. This means that the dihedral angle sampling is included

```
Data   : An input rigid molecule $M$;
         A positive surface contact threshold $t$;
Result : A set of candidate crystal structures.
  Fix the position of one copy of $M$ and call it $M_1$;
  Sample possible crystallographic positions of a neighbor molecule of $M_1$;
1 Reject neighbors having significant clash with $M_1$;
2 Reject neighbors where the surface in contact with $M_1$ is less than $t$;
3 Let $P$ be the set of accepted positions;
  for each implemented space group $G$ do
      for each position $M_2 \in P$ do
          for each set of base vector directions $\hat{a}, \hat{b}, \hat{c}$ compatible with $G$ and $M_2$ do
              Determine $\|a\|, \|b\|, \|c\|$ based on: location of $M_2$, volume of the unit
              cell, clashes between neighbor molecules, and other pruning criteria[a];
              if acceptable crystal structure then
                  Estimate lattice energy;
              endif
          endfor
      endfor
  endfor
4 Select a subset of generated structures based on energy and geometric diversity;
  Run local optimization on the selected subset;
```

[a] These criteria are used to speed up the search and to prevent duplicate structures; some of them are explained in the next sections.

Algorithm 3.1.1: The crystal structure prediction algorithm for rigid molecules.

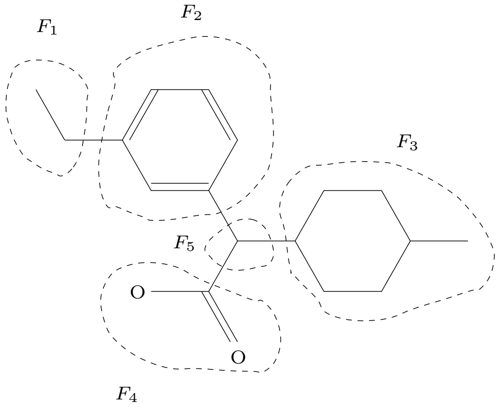


Figure 3.7: The input flexible molecule is fragmented into rigid fragments.

in the pair generation, but from that point on, i.e., Line 3 of Algorithm 3.1.1, the same method as before is followed.

To sample the dihedral angles during the pair generation process, first, the input flexible molecule is divided into rigid fragments by cutting the rotatable bonds, as demonstrated in Figure 3.7. The largest rigid fragment, F, is chosen and the possible locations for the same fragment, in a neighbor molecule, is generated as before. Then, for each of these generated fragment pairs, other fragments are added one by one until the entire molecule is constructed in both copies. In the process of adding these fragments the dihedral angles are sampled too.

To clarify the pair generation process, let us look at one example. For the molecule in Figure 3.7, the algorithm starts by F_2 (note that the rotatable bonds are included in rigid fragments and so, in this case, F_2 or F_3 might be chosen as the first fragment). For each of the generated pairs for fragment F_2, other fragments are added, avoiding any significant clash. The other criteria to limit the number of pairs generated is that the amount of surface contact between the two pairs should be above a certain threshold at each step (Figure 3.8 and 3.9). This threshold is determined according to statistics collected from CSD.

The contact ratio threshold is pivotal in keeping the number of generated pairs within a practical limit. On the other hand, this threshold should be set such that at least one of the pairs in the experimental structure is guaranteed to be generated. To set this threshold

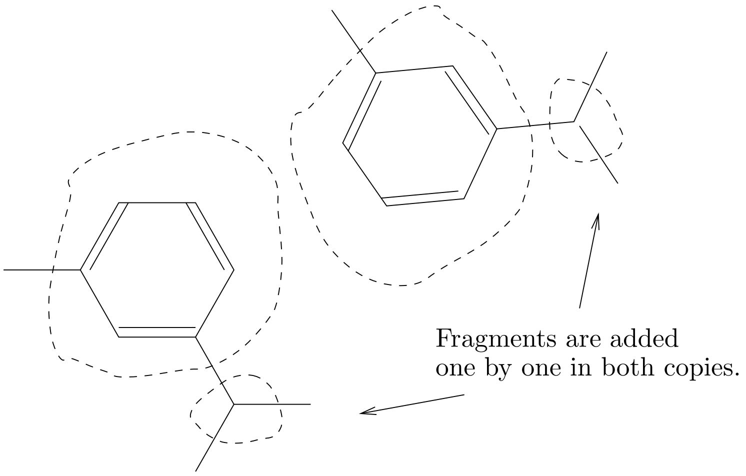


Figure 3.8: The fragments are added while the dihedral angles are being sampled.

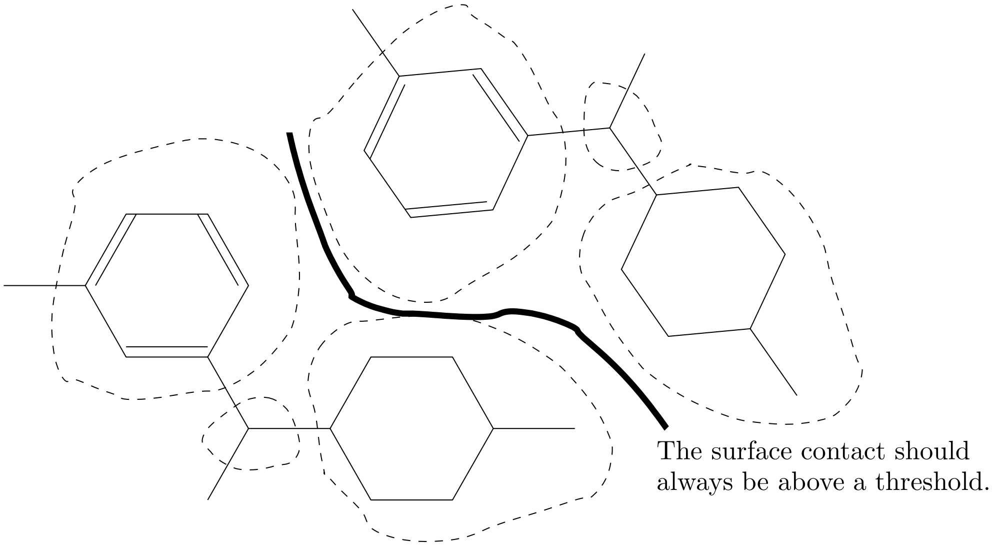


Figure 3.9: At each stage, the surface contact area should be above a statistically determined threshold.

2,649 structures, randomly selected from the CSD, that satisfy the following criteria have been analyzed (these criteria are used in all of the statistical studies of this chapter unless otherwise stated):

- No cocrystals, hydrates, or salts, i.e., the asymmetric unit contains a single connected molecule.
- R-factor, at most, 5%.
- No errors or disorders.

The following definitions are instrumental in clarifying the contact ratio concept:

Definition 7 *Let C be a crystal structure of a flexible molecule with rigid fragments $F_1, \ldots, F_r$, where F_1 is the largest. Let M and N be two molecules in C and $\pi = (\pi_1, \ldots, \pi_r)$ be a permutation of fragments corresponding to a traversal of the tree of fragments starting from F_1, i.e., $\pi_1 = 1$. The i^{th}* contact surface *of M and N over fragment order π or $S_i(M, N, \pi)$ is the surface in contact between M and N when only fragments $F_{\pi_1}, \ldots, F_{\pi_i}$ are considered. The* minimum contact ratio $\mathrm{MC}(M, N, \pi)$ *of M and N over π is*

$$\mathrm{MC}(M, N, \pi) = \min_{1 \leq i \leq r} \{S_i(M, N, \pi)\}. \tag{3.3}$$

The guaranteed contact ratio *in C or* $\mathrm{GCR}(C)$ *is the maximum of* $\mathrm{MC}(M, N, \pi)$ *over all choices of M, N, and π in C divided by the total surface of a molecule.*

For each structure C, all possible pairs of neighbor molecules are selected, and all the orders of adding fragments are analyzed. For each pair of neighbor molecules and each fragment order, the minimum surface contact ratio during the fragment addition is calculated as defined in Equation 3.3 which leads to the calculation of GCR(C) as defined in Definition 7. This guaranteed contact ratio shows that in the crystal structure C, there exists a pair of neighbor molecule and a specific order of adding fragments, such that the contact ratio is never less than GCR(C) during the fragment addition process. Of course, when the flexibility increases GCR(C) decreases as graphed in Figure 3.10. Indeed, the eCrySP performance decreases significantly when the flexibility increases. Therefore it is impractical, to use eCrySP, at its present form, for molecules with more than six or seven

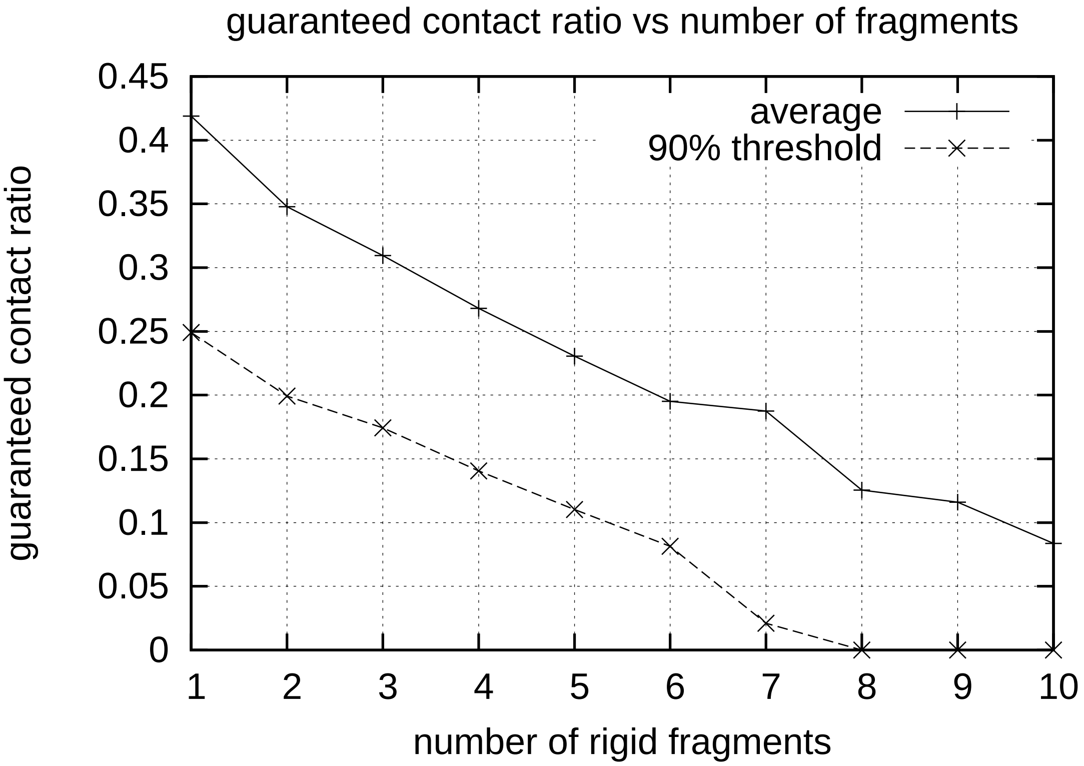


Figure 3.10: Summary of analysis of surface contact ratio between neighbor molecules in 2649 structures of CSD. The average values of *guaranteed contact ratio* (Definition 7) is shown as a function of number of rigid fragments. The value that is less than 90% of the cases is plotted as well.

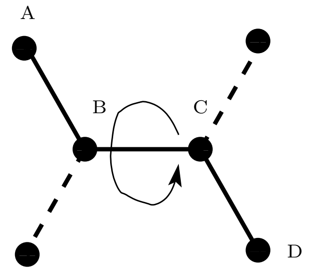


Figure 3.11: The dihedral angles are categorized based on the properties of the four defining atoms A, B, C, and D.

rigid fragments. In Figure 3.10, the average values of GCR(C) are plotted as a function of the number of fragments. In addition, the value that is less than 90% of the cases is graphed as well.

The other factor in flexible pair generation is the choice of the dihedral angle sampling steps. For this part, again, a thorough analysis of the structures in CSD has been carried out. Each dihedral angle is defined by four atoms, as illustrated in Figure 3.11. Atoms B and C are the endpoints of the rotatable bond. For each of them, a neighbor atom should be selected, for which the heaviest neighbor is selected. Only the cases where A, B, C, and D are all heavy atoms (i.e., non-hydrogen) are looked at, since it is relevant to the flexibility modeling of eCrySP. In this study, dihedral angles are categorized based on the hybridization of these four atoms. Because an automatic molecular perception is used, structures that contain only atoms C, N, O, H are considered to be more reliable. This selection results in 15,100 structures from CSD with a total of 61,946 dihedral angles.

One goal of this analysis is to measure the effect of the crystalline environment on the choice of dihedral angles, compared to the gas-phase minimum conformation. The analysis of the dihedral angle statistics indicates that for some hybridization categories, there is a strong preference for specific values. The results for important categories are graphed in Figure 3.12. For example, when B and C are both sp^3, a significant jump is visible at $-180°$, $-60°$, $60°$, and $180°$ (`hybrd-2-3-3-3` and `hybrd-3-3-3-3` in the graph). These values, of course, correspond to the staggered conformation. At the same time, in some of the other categories there is very little or almost no preference; such as when one of B and C is sp^2 and the other is sp^3 (`hybrd-2-2-3-2` and `hybrd-2-2-3-3` in the graph). It is also found that in most cases, the hybridization of the neighbor atoms is not significant in the

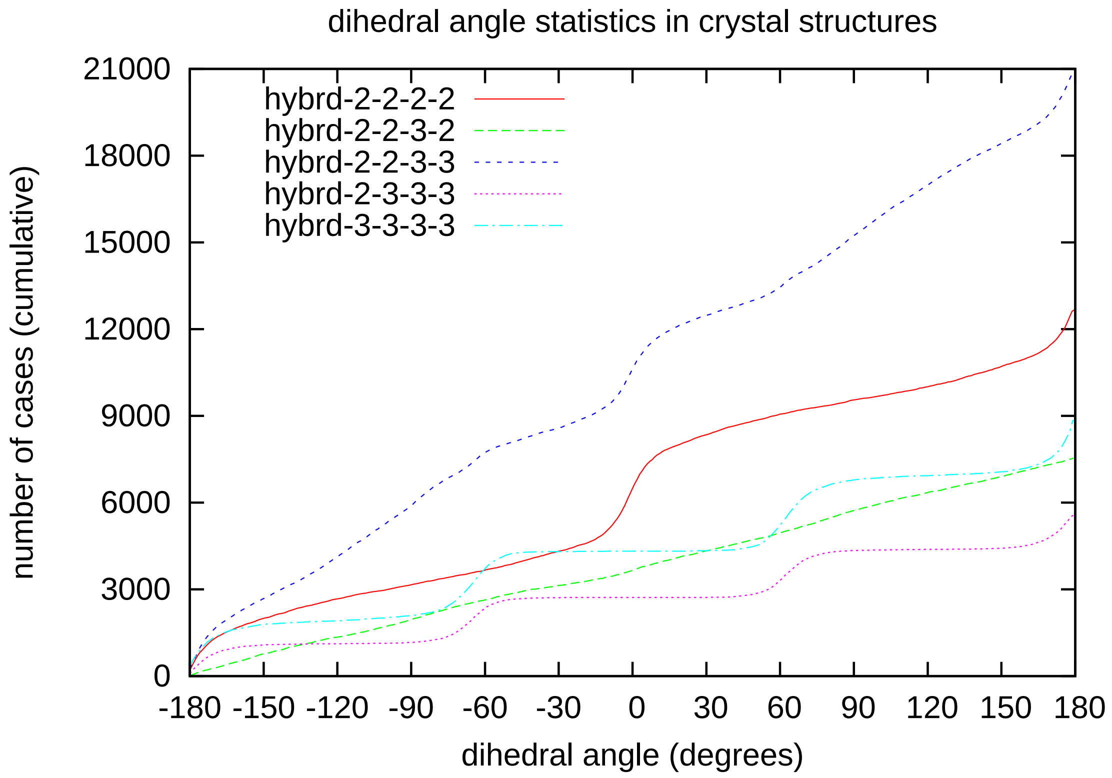


Figure 3.12: The results of dihedral angle analysis of 15,100 structures from CSD with 61,946 rotatable bonds.

dihedral angle preference, as seen in the aforementioned examples.

The dihedral angle sampling in eCrySP is now set to a default value of 30 degrees. However, a more sophisticated approach would be to establish a distribution of the angles based on these graphs. In fact, these data about the dihedral angles can be used to add a stronger conformation dependent energy term to the scoring function used, which is discussed later. However, in the present form, these enhancements are not included in eCrySP. All the results presented here for flexible molecules are with the default 30° sampling.

Some practical examples are shown in Figure 3.13. In this figure, from thousands of pairs generated by eCrySP, the one that is the closest to a pair in the target crystal structure is selected and shown. The Root Mean Square Deviation (RMSD) values are measured

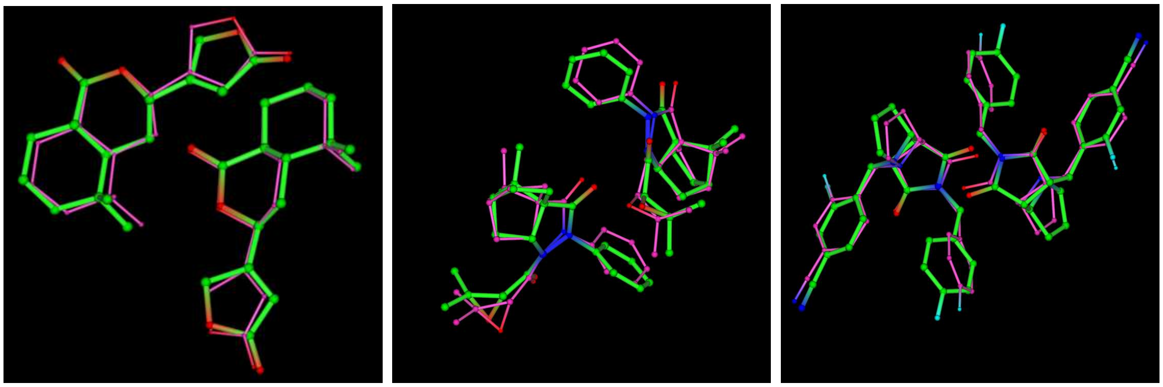


Figure 3.13: The closest pair predicted by eCrySP for three crystal structures. The pair from the target structure is shown by thin bonds. (CSD refcodes are LUBZIR, AQEBED, and BETMAP and the RMSD values are 0.41Å, 0.50Å, and 0.52Å respectively).

between the two pairs, i.e., two molecules from the target structure and two molecules from the predicted pair. In each case, the pair from the target structure is shown by thin bonds. The corresponding CSD refcodes are LUBZIR, AQEBED, and BETMAP, the RMSD values are 0.41Å, 0.50Å, and 0.52Å and they have two, three, and four rigid fragments, respectively. Note that as explained before the conformation of rigid fragments is taken from the experimental crystal structure, but no information about dihedral angles is used in the pair generation.

3.1.4 Other Pruning Criteria

As demonstrated in Section 3.3, the number of structures that are generated by eCrySP is in the range of millions for a typical drug-like molecule. This is a small fraction of the billions of structures, that are examined during the pair generation process and are filtered for various reasons in Algorithm 3.1.1. This amount of computation is large, and unless clever criteria are used for pruning the search space, the required CPU time would be impractical. Some of these criteria were explained in previous sections. Additional criteria can be devised using mathematical proofs or statistical analysis of experimental data. Examples of the mathematically proven criteria are:

1. The origin should be inside or close to the main molecule surface.

2. The centroid of the main molecule should be in the positive octant of the unit cell coordinate system, i.e., the coordinate system in which the base vectors are a, b, c.

3. The unit cell vectors a, b, c should satisfy the conditions of a Buerger unit cell.

Each of these constraints can be proven by starting from a crystal structure that does not satisfy it and change the origin and/or lattice vectors to satisfy it without changing the actual crystal structure. It is noteworthy that the angle conditions of the reduced unit cell conditions contradicts the second criteria above, which is why only the Buerger unit cell conditions are enforced.

Examples of statistically driven criteria include the contact surface threshold explained in Sections 3.1.2 and 3.1.3. Another criterion is the *volume*, which conveys that too much vacuum in a structure is not physical. More precisely, depending on the choice of the van der Waals radii, there is a lower bound on the ratio between the sum of the volumes of the molecules in the unit cell and the total volume of the unit cell. This ratio is called V_f for the volume that is *filled* by molecules. The claim is that V_f should be close to 1. This resembles the well known principle of *closest packing*, described half a century ago. One of the conclusions of this principle is that the minimum energy structure should also have the highest density. Of course there are exceptions to this principle, specially when the hydrogen bonds play a vital role in formation of a crystal structure [18].

Based on the data from 37,925 crystal structures in CSD, less than 0.4% of the crystals have V_f less than 0.75, as shown in the graph of Figure 3.14. In the volume calculations, an estimate of the van der Waals radii of atoms was adopted, (as there are different tables for such radii in the literature). Some of these radii are listed in the *Non-adjusted Radius* column of Table 3.1. The volume graph, corresponding to this column, is represented by a solid line in Figure 3.14.

In some cases, the volume ratio is greater than one. This is an artifact of the sphere model. For some of the important interactions, adjacent atomic spheres can overlap. For example, in a hydrogen bond of N—H$\cdots$O the distance of H and O is less than the sum of their van der Waals radii, which means their hypothetical spheres are intersecting. A set of *adjusted atom radii* are used which is based on the activity of the atoms. These values are listed in the *Adjusted Radius* column of Table 3.1. The volume graph corresponding to this column is shown by the dashed line in Figure 3.14. Some statistics are also collected

Element	Non-adjusted Radius (Å)	Adjusted Radius (Å)	
H	0.8	non H-bond	H-bond donor
		0.8	0.2
C	1.6	hydrophobic	neutral
		1.6	1.5
N	1.5	non-polar	polar
		1.5	1.3
O	1.4	non-polar	polar
		1.4	1.2
F	1.47	1.47	
S	1.8	1.8	
Cl	1.6	1.6	

Table 3.1: Representative atom radii used for molecule volume calculations.

on the atom distances by using the structures in CSD to adjust these radii according to the real crystal structures (for a similar attempt, see [108]). It has been also experimentally shown by many other researchers that the most stable structure, usually has one of the largest densities among other possible crystal structures, e.g., [69].

3.1.5 Selection and Local Minimization

Due to computational cost, from the many structures that are generated by eCrySP at the sampling level, only a small subset is selected for further local optimization (Line 4 of Algorithm 3.1.1). The selection is done, using an online geometric clustering of the structures. From each cluster, a representative is selected based on the estimated lattice energies. At the local optimization level, a more accurate energy estimation method can be used. The local optimization method stems from the Powel algorithm implemented by Press et al. [100].

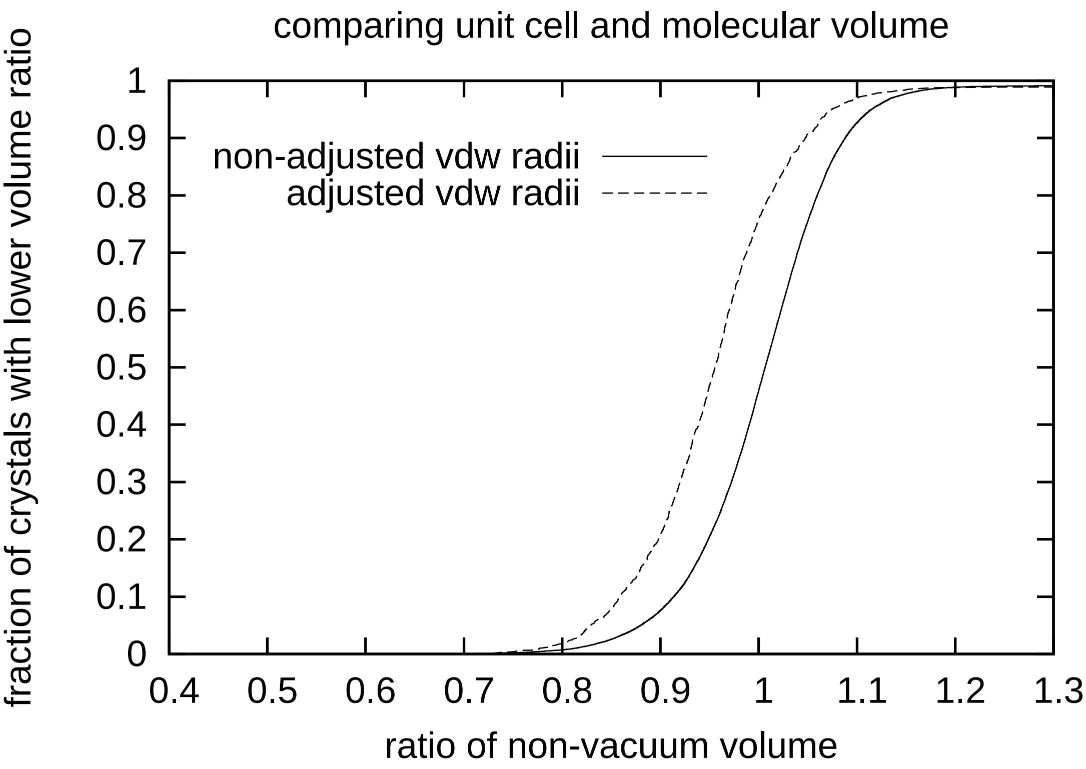


Figure 3.14: The graph of $F(x)$, where $F(x)$ is the fraction of crystal structures with V_f less than x (see the text for the definition of V_f). The two graphs compare adjusted and non-adjusted van der Waals radii.

3.1.6 Parallelization

To utilize the clusters of many CPUs, a parallelization method is implemented in eCrySP. The assigned number of CPUs, n, is simply set, and then the search engine divides the search space into n regions. The division is initiated at the pair generation step.

3.2 Scoring Function

A scoring function has been developed for eCrySP to compare the lattice energy of different structures. The principal components are similar to the ones of W99 force field [134, 135, 136], i.e., a combination of a van der Waals and an electrostatic term. Since the function values are not scaled into an energy range using sublimation energies, the term score is used here instead of the lattice energy. The scoring function is a major component that can be improved significantly, as displayed in Section 3.3. In view of this, eCrySP has been designed such that replacing the scoring function is easy. However, since the number of the structures compared is huge, any scoring function used prior to local minimization should be efficient. The total score is the sum of the interacting atom-pair scores. For a pair, a and b, of atoms at distance $d_{a,b}$ with charges q_a and q_b, the interaction score is

$$S(a,b) = C_v \mathrm{vdw}(a,b) + C_e \mathrm{es}(a,b), \tag{3.4}$$

where

$$\mathrm{vdw}(a,b) = \epsilon_{a,b}\left(\frac{r_{a,b}}{d_{a,b}}\right)^6\left(\left(\frac{r_{a,b}}{d_{a,b}}\right)^6 - 2\right), \tag{3.5}$$

and

$$\mathrm{es}(a,b) = k_e\frac{q_a q_b}{d_{a,b}}. \tag{3.6}$$

Several different forms were tested to model the dispersion-repulsion forces, e.g., the 8-4 form, but eventually (3.5) was chosen. In (3.5), $\epsilon_{a,b}$ and $r_{a,b}$ are the minimum energy and the ideal distance for atoms a and b, respectively. In other words, at distance $r_{a,b}$ this term is at its minimum $-\epsilon_{a,b}$. A knowledge-based approach is applied to choose these values by analyzing the interactions in about 90 thousand structures from CSD; for this training, cocrystals were not excluded. The distance range is divided into n intervals by choosing $d_0 = 0 < d_1 < d_2 < \cdots < d_n$. If the selected set of structures is called Λ, the main idea is to calculate the probability $\Pr_{a,b}(d_{i-1} \leq d < d_i)$, which is the probability of two interacting atoms of types a and b being at distance $[d_{i-1}, d_i)$ in a random interaction in Λ. Of course, longer distances have a higher chance, because the spherical shell $d_{i-1} \leq d < d_i$ has a larger volume. Therefore, these probabilities are normalized by using the volume of these shells, and the most likely interval is selected; $r_{a,b}$ is set this way. For $\epsilon_{a,b}$ a Boltzman-like distribution is employed to assign energies to these normalized probabilities, similar to the approach of Grzybowski et al. [59]. It is noteworthy that $r_{a,b}$ and $\epsilon_{a,b}$ are most reliable when a significant number of (a, b) interactions occur in Λ, although, these values are estimated for underrepresented pairs as well. More details about setting these parameters and possible future improvements of the scoring function are discussed in Section 4.2.

For partial charges different methods were tried, e.g., methods developed based on accurate calculation of charge distribution for functional groups using the quantum mechanics calculations done by the Gaussian 03 software package [48]. Details of these methods are given in [109]. Gasteiger charges were also used, mainly to compare the effect of charge assignment in final crystal prediction. The Gasteiger charges were calculated by OpenBabel [3]. These different approaches does not seem to improve the results of Section 3.3 significantly.

In the electrostatic equation (3.6), the constant k_e or the Coulomb's constant is $1/4\pi\epsilon$ in which ϵ is the dielectric constant of the medium. Determination of ϵ is not trivial and one simple way to do it is to use a distance dependent constant. Therefore, $C_e q_a q_b / d_{a,b}^2$ is used as the electrostatic term of the whole scoring function (3.4). The details of this approach and the reasoning behind it is described in [109].

Finally, the constant C_v is used to adjust the weight of the dispersion-repulsion term compared with the electrostatic term.

3.2.1 Atom Type Extensions

There are a few extensions to the standard atom types in eCrySP. The first is to use a special atom type for the lone-pair electrons which is denoted by LP. For example, the oxygen of the carbonyl has two LPs connected to it. This atom type significantly improves modeling of the Hydrogen bonds. If a single charge is placed at the atom nucleus only then a Bohr atom model is used which ignores the electron density distribution. As shown in

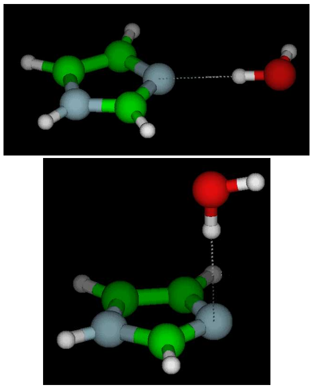


Figure 3.15: The two geometries for the demonstrated hydrogen-bonds have an energy difference of about 4.5 Kcal/mol [109]. This can be modeled by assigning charges to the lone-pairs but cannot be captured by the tradictional point charge model.

Figure 3.15 this can result in significant errors in calculation of the energy of a Hydrogen bond. In this figure the interaction between the imidazole ring and a water molecule is illustrated in two different relative geometries with an about 4.5 Kcal/mol difference. This difference is ignored in the Bohr atom model.

The other extension is to divide the most frequent atom types into several sub-types. These atom types are C, N, O, H, and LP; and the extra types are listed in Table 3.2.

3.2.2 Rotamer Optimization

As discussed in Section 3.1.1, the conformation sampling of the eCrySP search does not include rotatable bonds connected to terminal heavy atoms. For example, the rotation around the single bond, connecting a hydroxyl to a carbon, is not included in the dihedral sampling. This type of rotation is modeled on-the-fly during the score calculation. When the interactions of a molecule with its environment in the crystal structure is being calculated, a rotamer sampling procedure optimizes the rotation for each of these terminal rotamers, according to the energy values. Two examples are highlighted in Figure 3.16.

Type	Description
C_{ar}	carbon in aromatic ring or resonance, e.g., benzene
N_{ar}	nitrogen in aromatic ring or resonance, e.g., histidine
O_{ar}	oxygen in aromatic ring or resonance
H_{lipo}	H on sp3 hydrophobic carbon, e.g. aliphatic chain, cyclohexane
H_{ar}	H on hydrophobic carbon in aromatic ring (non-polarized), e.g. H on benzene
LP_{lipo}	LP on hydrophobic Halogen, e.g. F, Cl, Br, I
H_{don}	hydrogen bond donor H (polar-atom-H), e.g., proton of peptide -NH
LP_{acc}	hydrogen bond acceptor LP, e.g. on ketone =O

Table 3.2: Some of the extra atom types used in the scoring function.

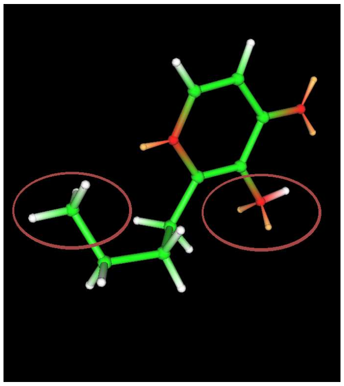


Figure 3.16: Rotamer sampling and optimization is done in each call of the scoring function. The rotamers are highlighted for a generated conformation of CSD refcode SABMAK. The calculated lone-pairs are also shown.

3.3 Experimental Results and Discussion

In this section, the results of several crystal structure prediction experiments done by eCrySP are compared with the known experimental structures. As discussed before, the input molecules are given in 3D coordinates, and the bond lengths and angles are not changed during the search. No information about the dihedral angles of the rotatable bonds is used for CSP. In addition comparisons with the Polymorph Predictor module of Materials Studio version 4.4 [2] are also conducted. The principal criteria for the comparisons of this section is the RMSD of the predicted structure from the experimental structure, which is described next.

3.3.1 RMSD Calculation

The key idea for RMSD calculation in this section is similar to the COMPACK program [29], which is also used in the blind tests of CSP [37]. To calculate the RMSD between two structures A and B, one center molecule with ten of its neighbor molecules are selected from A first. Then the center molecule is overlaid on a molecule in B, and for each of the neighbors in A, a closest molecule from B is selected. These two sets of 11 molecules are overlaid again, and the distances between corresponding atoms are measured to calculate RMSD between A and B. The method of the COMPACK program in CSP2004 is more accurate, since it compares the interatomic distances between two structures and it also finds the best matching set of the possible sets of the neighbor molecules. However, a much faster method is needed to compare a large number of structures in a reasonable amount of time. Therefore, the faster and less accurate method, described above, is used. Generally, the estimated RMSD values by this method should be greater than that of COMPACK for the same pair of structures. The number of neighbor molecules used to calculate RMSD is also another difference. In CSP2004 this number was in the 12-16 range. However for some cases, neighbors beyond 10 or 11 are too far from center molecule and so only 10 neighbors are used for all cases here. Usually adding a few more neighbor molecules will not change the calculated RMSD values significantly.

When the RMSD of a crystal structure is referenced, it is implied that it is the RMSD from the experimental structure.

3.3.2 Rigid Molecules

The first test set consists of 24 structures from CSD. The procedure for selecting these structures is as follows: First, a small number of structures of CSD, satisfying the general conditions described in the previous sections, were selected randomly. From these, flexible or too large rigid structures were removed, especially the ones with large cyclic fragments. A 2D view of each molecule, along with their space group, is given in Table 3.3. The representations have been generated using the `molconvert` utility of ChemAxon [4]. The goal of starting from a random set is to simulate a blind test.

For this experiment, the six most frequent space groups in CSD were searched, i.e., $P2_1/c$, $P\overline{1}$, $P2_12_12_1$, $C2/c$, $P2_1$, and $Pbca$. Therefore, the crystal structures with the space group other than these six space groups were removed from the test set. Some other space groups are also implemented and can optionally be searched but the searching in these six groups is the default setting in eCrySP. These groups cover more than 82% of the structures in CSD [5]. Note that the space groups with inversion-like operators were not excluded for the chiral molecules. This means that the racemic crystal structures were also searched, and no assumption is made on pure enantiomers. Another important note is that eCrySP does not sample cycle conformations, as discussed before.

The results of this experiment are given in Table 3.4. The tests were conducted by using 40 nodes of a cluster of Intel Xeon 2.40GHz processors. As discussed in Section 3.1.6, the search was automatically divided between these nodes. The second column indicates the number of structures that were accepted at the end of the sampling level, i.e., before the selection of Line 4 of Algorithm 3.1.1. As demonstrated, tens of thousands of structures are generated at this level. Since the local optimization is slow, at the end of the sampling, a small subset of structures should be selected which does not necessarily mean that the structure closest to the experimental structure is chosen. This is mainly due to the fact that the scoring function is not perfect and a close structure can be rejected because there are many other structures with better scores. In this experiment, each computing node selected 200 structures and after the local optimization, a central process selected 300 structures with best scores for the output. The third column of Table 3.4 shows the RMSD of the structure closest to the experimental structure among the output of 300 structures; the average RMSD is 1.16Å.

To check the effect of the selection procedure of Line 4 of Algorithm 3.1.1, the following

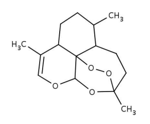
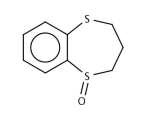
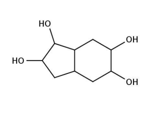
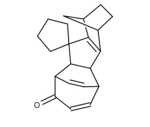
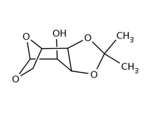
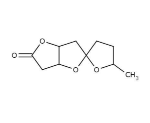
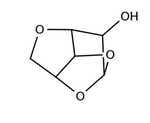
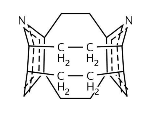
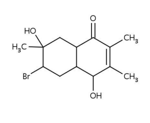
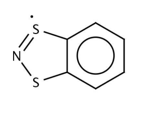
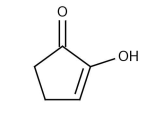
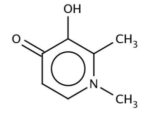
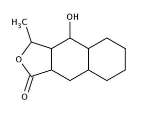
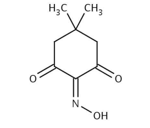
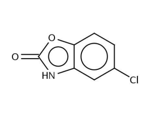
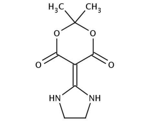
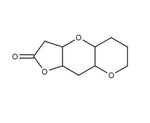
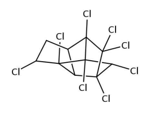
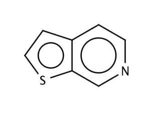
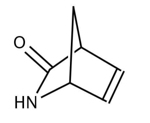
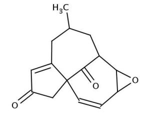
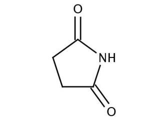
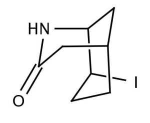
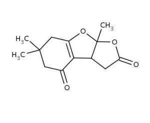
CSD refcode	2D diagram	space group	CSD refcode	2D diagram	space group	CSD refcode	2D diagram	space group
ADAHAP		$P2_12_12_1$	AQIGOW		$P2_12_12_1$	CERMOB		$P2_1$
CIYKOK		$P2_1/c$	COHJIS		$P2_12_12_1$	DAJSIQ		$P2_12_12_1$
DANGLP		$P2_12_12_1$	DAZCLA01		$Pbca$	EHULEY		$P2_1$
FAGRIO		$Pbca$	FUGJUM01		$P2_12_12_1$	GALDEC		$Pbca$
HULPUZ		$P2_12_12_1$	NANGEO		$P2_1/c$	NEWKOP		$P\overline{1}$
ODOPUS		$P\overline{1}$	PIGRAY		$P2_1/c$	PTCHLD		$P2_1/c$
RAVTAJ		$Pbca$	RUVZEN		$P2_12_12_1$	SIBGAL		$Pbca$
SUCCIN04		$Pbca$	WIPBOM		$P2_12_12_1$	YAMXEQ		$P2_12_12_1$

Table 3.3: List of molecules used in the rigid experiments. Ring conformations are not changed during the search.

experiment was carried out: A similar search was conducted, but this time, only in the space group of the experimental structure. Also, the RMSD values of all of the generated structures were calculated (without local optimization). From these RMSD values, the smallest was found and is reported in the fourth column of Table 3.4. The average value is 0.93Å, which is an indication of the accuracy of the sampling phase. Then, only the structures within 2.0Å RMSD from the experimental structure were selected and local optimization was carried out on them. The fifth column of Table 3.4 contains the final RMSD value of the closest structure. Local optimization improves the RMSD with an average of 0.21Å. The improvement is evident in almost every case. Note that the average in this column is 0.72Å, whereas the average minimum RMSD in the 300 output is 1.16Å. This means that if at the selection of Line 4 of Algorithm 3.1.1 the best choice was made, the average RMSD improves by 0.44Å. Finally, in a few cases, e.g., GALDEC, where the RMSD reported in the fifth column is worse than the RMSD of the third column, i.e., the RMSD of the closest structure in the 300 output. One reason for these differences is that the later experiment was done on a different CPU architecture and therefore the numerical errors, specially at the local minimization level, can cause such differences.

The local optimization was also conducted on the experimental structure itself, and the RMSD of the locally optimized structure is also reported in Table 3.4. This RMSD value is also a measure of the quality of the scoring function, because in an ideal case, the experimental structure should already be at a local minimum. Of course, measurement errors always exist in experimental methods used in determination of crystal structures. The results of this experiment are reported in the last column of Table 3.4.

To illustrate an example, the closest output structure for refcode RUVZEN is overlaid with the experimental structure in Figure 3.17. The hydrogen-bonding network of some of the molecules are also depicted in this figure. The hydrogen-bonding in this structure is discussed in the original structure paper [120].

3.3.3 Comparison with Polymorph Predictor

Polymorph Predictor is one of the common tools used for crystal structure prediction. Many participants in the previous blind tests of CSP adopted Polymorph Predictor as one of the computational tools, either as part of the Accelrys Cerius^2 software toolkit or the later Materials Studio version [2]. The ancestor of this tool is the simulated annealing

Column Index	1	2	3	4	5
CSD refcode	Number of Generated Structures	RMSD in 300 (Å)[a]	Best Gen. RMSD[b]	Best Gen. Local-opt RMSD[c]	Exp. Str. Local-opt RMSD[d]
ADAHAP	43663	1.01	0.84	0.43	0.60
AQIGOW	59544	0.53	0.80	0.45	0.37
CERMOB	49178	0.58	0.79	0.60	0.19
CIYKOK	28812	1.64	1.61	1.87	0.35
COHJIS	60194	0.73	0.67	0.62	0.38
DAJSIQ	53473	1.82	0.86	0.38	0.26
DANGLP	132905	1.52	0.74	0.60	0.31
DAZCLA01	24844	2.39	0.74	0.56	0.44
EHULEY	30901	2.00	1.04	0.91	0.37
FAGRIO	242910	0.73	0.59	0.50	0.47
FUGJUM01	241000	0.77	0.82	0.36	0.39
GALDEC	140600	0.49	0.99	0.66	0.41
HULPUZ	74321	0.51	0.88	0.60	0.48
NANGEO	73872	0.38	0.71	0.49	0.23
NEWKOP	142287	1.20	0.82	0.35	0.24
ODOPUS	59543	0.85	0.94	0.75	0.43
PIGRAY	72995	1.67	1.14	0.83	0.45
PTCHLD	7201	2.21	2.31[e]	2.31[f]	0.24
RAVTAJ	273827	0.62	0.85	0.45	0.54
RUVZEN	173182	0.69	0.60	0.52	0.36
SIBGAL	40879	0.89	1.39	1.13	0.76
SUCCIN04	242096	1.17	0.57	0.35	0.26
WIPBOM	92239	2.46	0.71	0.56	0.35
YAMXEQ	30733	1.04	0.85	0.87	0.38
average	99633.3	1.16	0.93	0.72	0.38

[a] Among the 300 output structures after local optimization, the closest to the experimental structure was selected and the RMSD from the experimental structure was calculated.

[b] The RMSD of every structure generated at the sampling level in the same space group of the experimental structure was calculated and the minimum is reported. No local optimization was done in this case.

[c] From all the structures generated at the sampling level, those within a certain geometric threshold of the experimental structure were selected and locally optimized. The minimum RMSD after local minimization is reported.

[d] The RMSD of the locally optimized experimental structure.

[e] In this specific case a small increase in the clash threshold generates a structure with 1.67 Å RMSD.

[f] None of the generated structures were within the range to be locally optimized.

Table 3.4: Results of the eCrySP predictions for the set of 24 rigid molecules of Table 3.3.

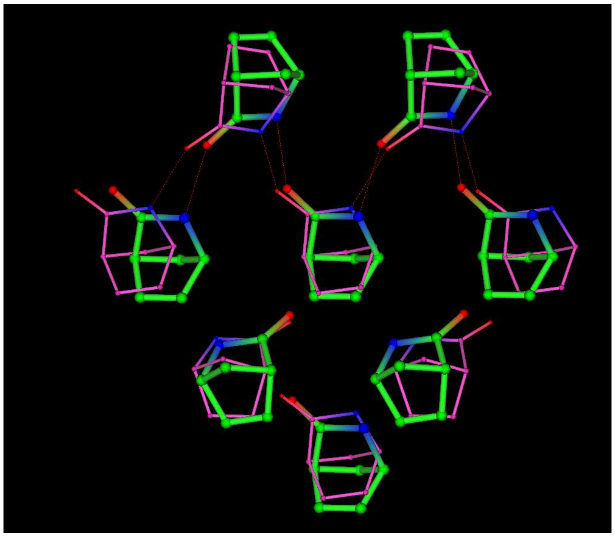


Figure 3.17: The closest eCrySP predicted structure (thick bonds) compared to the experimental structure of CSD refcode RUVZEN (thin bonds) among 300 output structures. The RMSD is 0.68Å and the dotted lines indicate hydrogen bonds.

method of Karfunkel and Gdanitz [72, 54], as described in the Section 1.2. A computational experiment, similar to the one in the previous section, was carried out by using Polymorph Predictor, to make a comparison with eCrySP.

The current implementation of Polymorph Predictor in version 4.4 of Materials Studio, has four steps. The first step is the simulated annealing step which treats the input molecule as a rigid body, and by changing the 12 parameters of the crystal structure, attempts to minimize the lattice energy. After this step, a clustering is done to remove duplicate structures, i.e., structures within a certain geometric threshold from each other. Then, a local optimization is carried out on one structure from each cluster. Finally, a second clustering is done to remove the duplicates again. The local optimization step can handle the molecular flexibility, but for a fair comparison with the experiment of the previous section, the input molecule is kept rigid throughout the whole process, and of course the conformation from the experimental crystal structure is employed.

Similar to eCrySP, different accuracy levels can be used for Polymorph Predictor. For a fair comparison, an accuracy level is chosen to satisfy two constraints:

- The CPU time used should be comparable to that used by eCrySP in the experiment of the previous section.
- The final number of output structures should be close to 300.

Doing some measurements on simple cases, the *Medium* setting of Polymorph Predictor was selected and the number of clusters was limited to 60 for each space group. Since the same six space groups were searched, the total number of output structures should be at most 360. The number of clusters is an upper bound and Polymorph Predictor may generate fewer clusters. After finishing the experiment, it was found that the average number of output structures was 283. The total time spent by all the cluster nodes in the eCrySP experiment of the previous section was summed up and compared to the total time used by Polymorph Predictor. The average total runtime of eCrySP was 321.2 minutes and that of Polymorph Predictor was 309.1 minutes, as reported in Table 3.5. The Polymorph Predictor experiment was conducted on a different CPU, because Materials Studio was not installed on the cluster. Therefore, the times reported for Polymorph Predictor were scaled to a CPU similar to the ones used in the eCrySP experiment.

The Dreiding force field [87] and Gasteiger charges were selected for energy calculation, as implemented in Materials Studio. From the set of output structures of Polymorph Predictor, the closest structure was chosen by using the aforementioned RMSD calculation method. These RMSD values are reported in the second column of Table 3.5 and should be compared with the third column which is the eCrySP closest structure in the 300 output. As demonstrated in this table, the average RMSD of closest structure predicted by the Polymorph Predictor is 1.1Å, whereas eCrySP closest RMSD is 1.16Å, although eCrySP performed better in several cases. It is important to note that multiple runs of Polymorph Predictor can return better or worse results because of the stochastic nature of the algorithm. Also, note that without any changes in the search algorithm and with a better scoring function at the selection level, the eCrySP results can improve significantly, as indicated in the previous section. As discussed before, the eCrySP runs where distributed between 40 nodes, each returning 200 structures. From the 8000 structures returned, 300 with the best scores were selected for output. The closest structure in the whole set of 8000 eCrySP structures was also found for each case and the RMSD values are reported in the fourth column of Table 3.5. The interesting point is that the average RMSD in this column is 0.79Å. The significance of this finding is that with a better scoring function employed only for the final ranking (and not even during the search), the final results can improve significantly.

3.3.4 Flexible Molecules

The same pseudo-random procedure, as the one for the rigid case, was employed to select a few flexible test structures. From the selected set, the first with one, two, and three flexible bonds were chosen (based on the alphabetical order). The results for those three cases are reported in this section. As demonstrated, the eCrySP runtime increases significantly with the number of rotatable bonds. Consequently, it is not practical to use the conformation sampling feature for molecules with more than five or six rotatable bonds, i.e., six or seven rigid fragments. One main reason for this is that the surface contact pruning criteria is weakened as the number of rigid fragments increases, as demonstrated in Figure 3.10.

Besides the runtime issue, when the input molecule is too flexible the number of generated pairs and consequently the number of generated structures is enormous. As was demonstrated for rigid molecules, the procedure for selecting a diverse set of structures for local minimization, is responsible for a significant drop in the accuracy of the final output structures. This selection problem is even more serious when the number of generated structures explodes for a very flexible molecule.

The selected CSD refcodes for computational experiments were LUBZIR, AQEBED, and, BETMAP, with one, two, and three rotatable bonds, respectively. For each of these three cases, at least one of the pairs in the experimental crystal structure was successfully generated, as demonstrated in Figure 3.13. The same type of experiments as in the rigid cases were conducted for these experimental structures and the results are reported in Table 3.6. The format of this table has changed such that each structure information is represented in a column instead of a row. Additional data are also reported in this table, e.g., number of rigid fragments and the total CPU time. The experiment setting was exactly as before, i.e., the jobs were running on 40 nodes of a cluster of Intel Xeon 2.40GHz processors. Each node selected 200 structures from the generated structures at the sampling level, for local optimization. At the end 300 structures were selected for output by a central process.

For flexible molecules, it is not possible to make a direct comparison with Polymorph Predictor because its simulated annealing step treats the input molecule as a rigid body.

CSD refcode	Polymorph Predictor Closest RMSD (Å)	eCrySP Closest RMSD (Å)	eCrySP Closest RMSD in 8000 (Å)	Polymorph Predictor CPU Time (minutes)	eCrySP CPU Time (minutes)
ADAHAP	0.23	1.01	0.63	384.6	472.3
AQIGOW	2.71	0.53	0.53	472.0	206.4
CERMOB	0.42	0.58	0.58	529.3	344.0
CIYKOK	1.06	1.64	1.14	508.0	482.3
COHJIS	0.21	0.73	0.73	411.0	361.5
DAJSIQ	0.89	1.82	1.23	159.1	283.9
DANGLP	1.17	1.52	0.76	306.1	219.5
DAZCLA01	1.64	2.39	0.50	461.8	366.2
EHULEY	1.29	2.00	0.97	382.8	597.7
FAGRIO	2.00	0.73	0.73	230.3	205.2
FUGJUM01	0.18	0.77	0.42	193.1	166.7
GALDEC	0.18	0.49	0.49	326.8	319.1
HULPUZ	1.26	0.51	0.51	302.1	407.4
NANGEO	0.43	0.38	0.38	249.5	325.9
NEWKOP	0.20	1.20	1.03	135.0	259.4
ODOPUS	1.38	0.85	0.85	435.5	348.5
PIGRAY	0.79	1.67	1.63	397.8	290.2
PTCHLD	1.46	2.21	1.76	80.5	555.6
RAVTAJ	1.84	0.62	0.62	227.0	206.1
RUVZEN	1.05	0.69	0.69	207.0	164.1
SIBGAL	0.84	0.89	0.89	256.3	330.2
SUCCIN04	1.35	1.17	0.41	222.1	155.1
WIPBOM	1.67	2.46	0.73	276.5	246.1
YAMXEQ	2.16	1.04	0.85	264.8	395.2
average	1.10	1.16	0.79	309.1	321.2

Table 3.5: Comparison between the structures generated by Polymorph Predictor and eCrySP.

1	CSD refcode	LUBZIR	AQEBED	BETMAP
2	Number of Generated Structures	365794	254868	529801
3	RMSD in 300 Output (Å)	1.37	2.37	1.59
4	Best Generated RMSD (Å)	0.83	0.91	1.51
5	Best Gen. Local-opt RMSD (Å)	0.88	0.50	1.52
6	Experimental Str. Local-opt RMSD (Å)	0.37	0.54	0.35
7	Number of Rigid Fragments	2	3	4
8	Total CPU Time (minutes)	1907.5	2717.0	7380.7

Table 3.6: Results of the eCrySP predictions for the flexible molecules of Figure 3.13.

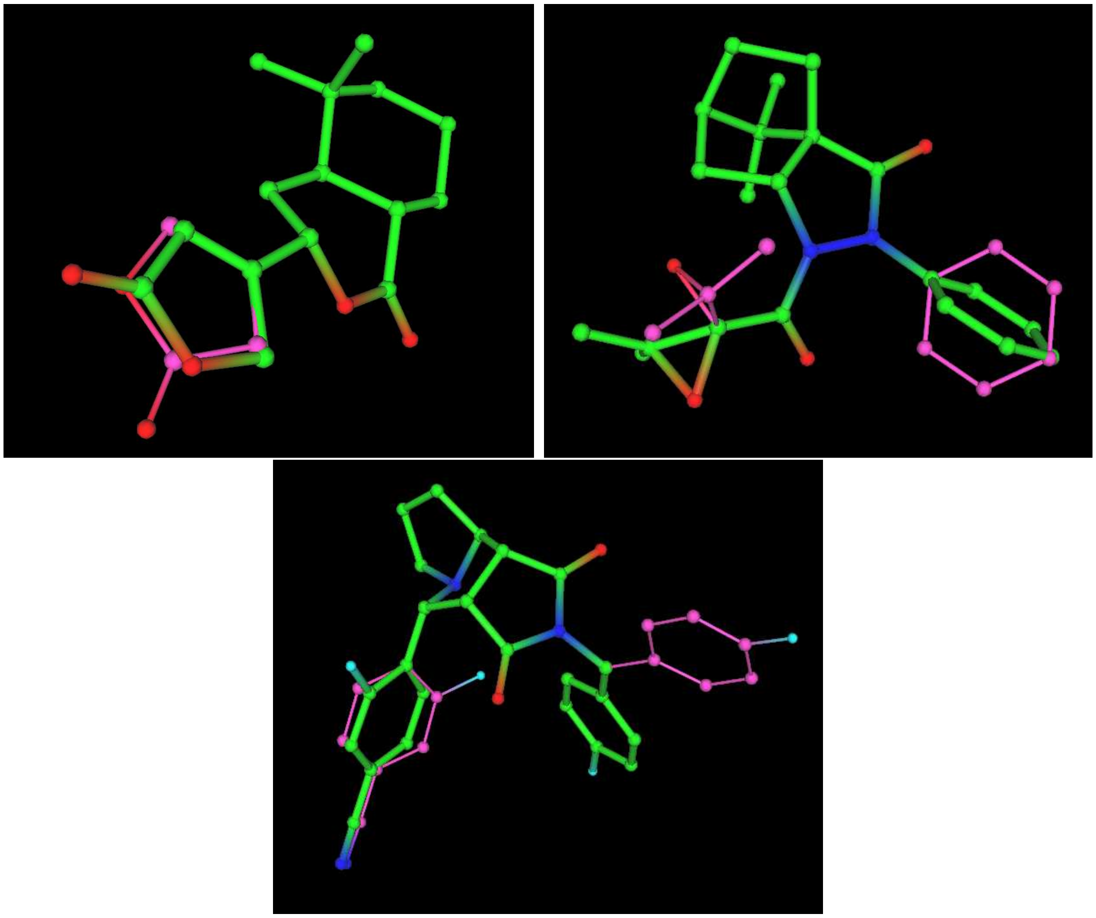


Figure 3.18: The target crystal structure conformation (thick-green) overlaid on the decoy conformation (thin-purple) for refcodes LUBZIR, AQEBED, and BETMAP.

The flexibility is handled in the local optimization step though. It is easy to demonstrate that such an approach is very sensitive to the input conformation. To show this, a conformation far from the one in the target crystal structure was generated by changing the dihedral angles of the rotatable bonds. These conformations, which are called decoy conformations are illustrated in Figure 3.18.

For each case, two experiments where done using Polymorph Predictor. In the first one the native conformation, i.e., the conformation in the target crystal structure was used. Then same prediction experiment was repeated using the decoy conformation. For these experiments the more accurate *Fine* setting was used instead of the *Medium* used for

	CSD refcode	LUBZIR	AQEBED	BETMAP
Native Conformation	Number of Output Structures	194	114	213
	RMSD of the closest output (Å)	0.38	0.46	1.51
Decoy Conformation	Number of Output Structures	219	178	246
	RMSD of the closest output (Å)	2.85	1.16	3.58

Table 3.7: Results of the Polymorph Predictor predictions for the flexible molecules of Figure 3.13.

rigid molecules of the previous section. Also the local optimization step was set to modify dihedral angles too. The results of these experiments are reported in Table 3.7. As it can be seen, when the decoy conformation is used as input, the closest structure is too far from the target structure, especially in the cases of LUBZIR and BETMAP. On the other hand the target structure is found when the native conformation is used. To see if more accurate predictions are possible with a better energy calculation method, the ESP-fitted charges were used in the case of LUBZIR. The charges were calculated using the DMol^3 module of Materials Studio which employes density functional theory to model the electrostatic structure of molecules [2].

As the final point about flexible molecules, it is noteworthy that if enough different conformations are used, methods like Polymorph Predictor that do not handle flexibility in the sampling phase, might be able to find the target structure. One idea to set these conformations is to do a conformation sampling followed by an internal energy minimization. This idea was tested for the simplest case of LUBZIR with one rotatable bond. With a one degree sampling, 360 conformations were generated using the Conformer module of Materials Studio. For each conformation, a local optimization based on the internal energy was done with a constraint of retaining the dihedral angle. The internal energies are plotted in Figure 3.19. The three conformations corresponding to the three local minima of this graph are overlaid on the native conformation in Figure 3.20. With three Polymorph Predictor runs, each using one of these conformations, 234 structures were generated with

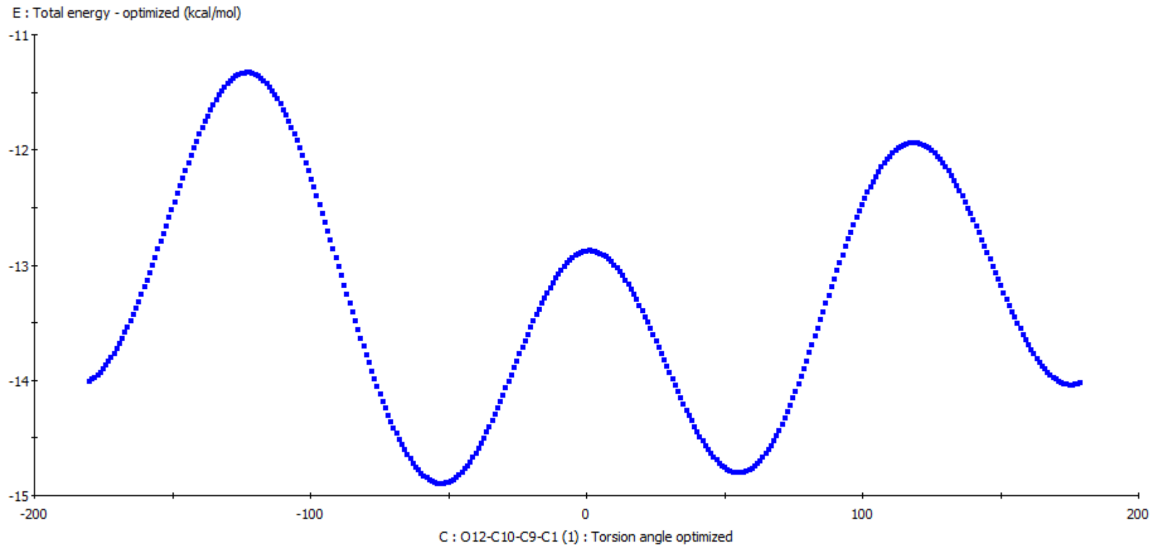


Figure 3.19: The internal energies of 360 conformations generated for the molecule of the refcode LUBZIR. The three local minima are at -53, 55, and 175 degrees.

a minimum RMSD of 0.29Å.

3.4 eCrySP Concluding Remarks

In this chapter, eCrySP, a new search method for crystal structure prediction, has been described, along with the default scoring function that is used with it. The most significant feature of this new method is its systematic approach that can guarantee a certain level of geometric accuracy in the sampling phase. It has been demonstrated that in most prediction experiments, at least some of the structures generated during the sampling are close enough to the experimental structure such that a local minimization from those can lead to the experimental structure.

The search space of possible crystal structures is large, especially when the conformation sampling and unit cell parameters sampling are handled simultaneously, as in eCrySP. To reduce the search space, several pruning criteria are implemented in eCrySP. Some of these criteria are based on the results of massive statistical analysis of CSD. In fact, the general

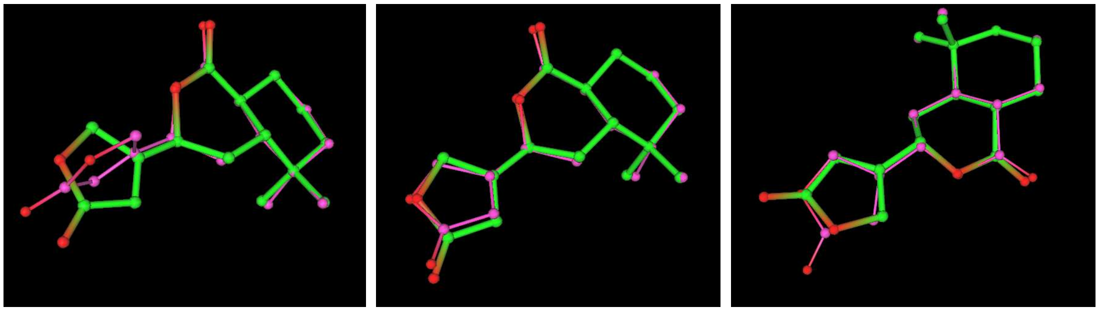


Figure 3.20: The three conformations (thin-purple) corresponding to the local minima of Figure 3.19 are overlaid on the native conformation (thick-green).

framework of the search is based on a few key observations of crystal structures.

The current implementation of eCrySP has been tested on a set of rigid and flexible molecules, and the predicted structures have been compared with those of the experimental structures. In addition, a comparison with the widely used CSP program, Polymorph Predictor, was also carried out.

In the experimental results, it was demonstrated that the most important reason for failure in finding the correct predictions is not the sampling of eCrySP, but the selection of a few structures after the sampling for local optimization or output. This is an indication that with a more accurate lattice energy estimation function, better results can be expected with the current search method.

Chapter 4

Discussion and Future Works

In this chapter several ideas are discussed for improving or extending the approaches proposed in the previous chapters of this thesis. In most of the cases we have done some preliminary investigations and experiments; the results of such efforts are also reported here.

First, in Section 4.1, we revisit the docking problem and propose ideas to address the receptor conformational changes caused by ligand binding. As it was mentioned in Section 1.1, the ultimate docking solution should model receptor flexibility at least to some extent. Although the concept of *induced fit* has been known for a long time, most of the current leading docking programs cannot handle binding site flexibility well. In fact this is one of the key issues that is currently researched and developed by different protein-ligand docking software teams, as discussed in Section 1.1. Some preliminary implementations of our proposed methods are done and the results are reported for the specific case of binding different ligands to the human carbonic anhydrase in Section 4.1.4. This is mainly done as a proof of concept.

The main contribution of this thesis is in the different search algorithms proposed for structure prediction problems. We have analyzed them from an algorithmic point of view and have shown their promising performance in practice. However as it was mentioned in Section 2.5 in the context of docking and was shown to a greater extent in Section 3.3 in the context of crystal structure prediction, the main obstacle in getting excellent results is the scoring function performance. In Section 4.2 we discuss some of the difficulties in developing scoring functions. We look at the problem of determining scoring function parameters as an optimization problem (Problem 2). Then we discuss some of our efforts in improving the scoring function for crystal structure prediction and propose ideas to improve it further.

4.1 Protein Flexibility and the Docking Problem

As it is stated by Teague in a survey of protein conformational changes upon drug binding [122] there are two types of conformational changes:

1. The main structure of the backbone is preserved and only the conformation of a few side chains interacting with the ligand are changed.
2. The protein undergoes a significant change by *hinge* and *shear* motions.

As a first step we consider the first type of changes which is easier to address. In fact, as it is shown in the statistical study of Najmanovich et al. [95], in 85% of the cases, the protein conformational changes upon ligand binding is limited to three side chains only. Therefor even with the assumption of a rigid backbone, most of the protein conformational changes in real cases are covered.

The idea of this section is based on the place-and-join methods described in Chapter 2. As it was mentioned, the input ligand is fragmented into rigid fragments, each fragment is independently docked, and then matching poses are evaluated as shown in Figure 4.1.

4.1.1 General Overview of Receptor Flexibility Handling

To extend the method of Chapter 2 to include side chain flexibility, the candidate flexible side chains should be identified first. Then, the same fragmentation method is applicable to sample their conformational space. This is shown in Figure 4.2 for a histidine residue.

We first note that a flexible side chain usually have less contact with other parts of the protein and is well exposed to make interactions with ligand and solvent. This is an intuitive observation and we have tested this with some of the reported experimental results. The details of how we identify these side chains is given in Section 4.1.2. Once

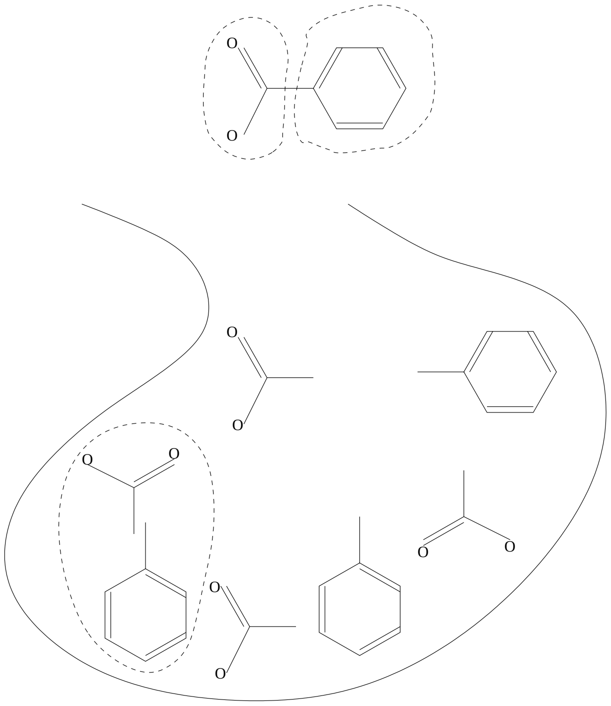


Figure 4.1: Review of the docking steps: (i) the input molecule is fragmented into rigid fragments (ii) RigiDock: each fragment is independently docked (iii) PoseMatch: matching fragment sets with good scores are selected (iv) the selected poses are locally optimized.

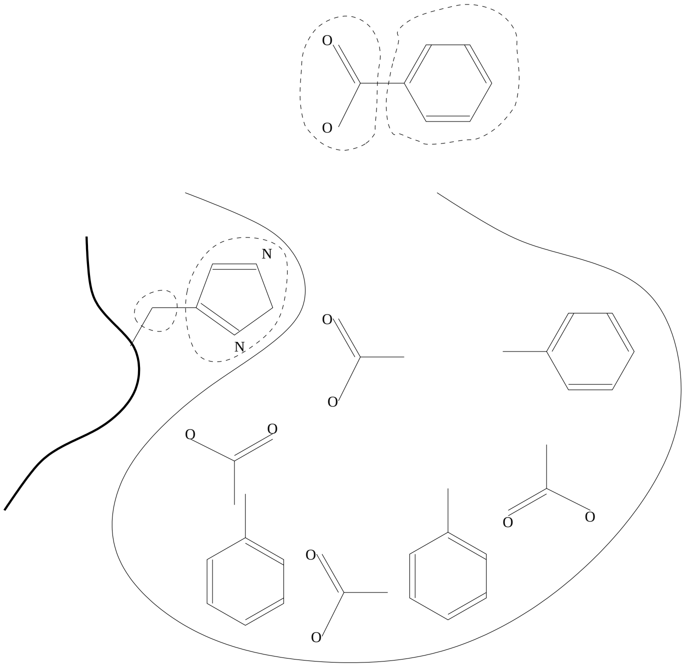


Figure 4.2: The inclusion of flexible side chains in the modeling of the docking process.

the candidate chains are identified we can model their flexibility the same way as we do for ligand. Each side chain can be broken into rigid fragments and the same RigiDock and PoseMatch steps can be done for each chain (Figure 4.2). In this case we should remove these chains from the receptor and make a *trimmed receptor*, because the ligand may now occupy the location of these chains. Of course, extra distance constraints should be included in the RigiDock and PoseMatch steps for the poses generated for side chains to make sure that they are always close to their C_α backbone atom.

The second step is to include rotatable bonds of the flexible side chains in the local optimization step. For this step the scoring should include:

- Interactions of ligand and flexible side chains with trimmed receptor.
- Interactions between ligand and flexible side chains.
- Interactions of flexible side chains with each other.

The scoring function may need a new tuning with this flexible side chain model. Since our main focus here is not the final energy values or the ranking of the poses we ignore this step.

Note that in this method the receptor and ligand flexibility are handled simultaneously which is the proper way to solve this problem. There are other approaches that treat ligand and receptor flexibility in different iterations and not at the same time [114].

From the above proposed method, we have implemented the flexible side chain detection and simultaneous optimization steps and they are integrated into eHiTS for the case studies of Section 4.1.4. The integration of flexible side chains in RigiDock and PoseMatch steps are not implemented yet and we leave it as a future work.

4.1.2 Detecting Flexible Side Chains

As we discussed earlier a side chain that has significant interaction with the rest of the protein should not be influenced very much by ligand binding. Therefore to determine candidate flexible side chains we count the number of atoms that are within 2.0Å of the cavity surface. If the ratio of number of these atoms to the total number of side chain atoms is greater than 0.8 we mark that side chain as flexible unless it has a disulfide bond or is interacting with a metal ion of the protein.

We tested this method on a set of receptors reported in Table 1 of Teague survey [122]. One particular set consists of 1HW8, 1HW9, 1HWI, 1HWJ, 1HWK, 1HWL PDB codes. The surface of the receptor in 1HW9 is shown in Figure 4.3. Different subunits are colored with different colors and one of the candidate side chains which is ASN-658 is highlighted by red. As one can see, this residue is well exposed to the solvent. It should be noted that all of the experimentally flexible residues are not necessarily determined by our method and to achieve that goal more parameter tuning based on statistics collection from PDB is needed. In Figure 4.4 all candidate residues are highlighted. A more sophisticated approach for predicting flexible side chains is given by Anderson et al. [11].

4.1.3 Simultaneous Optimization of Ligand and Receptor

The last step of eHiTS is the local optimization of ligand conformation. We added a new step after this which is the simultaneous optimization of ligand and flexible side chains

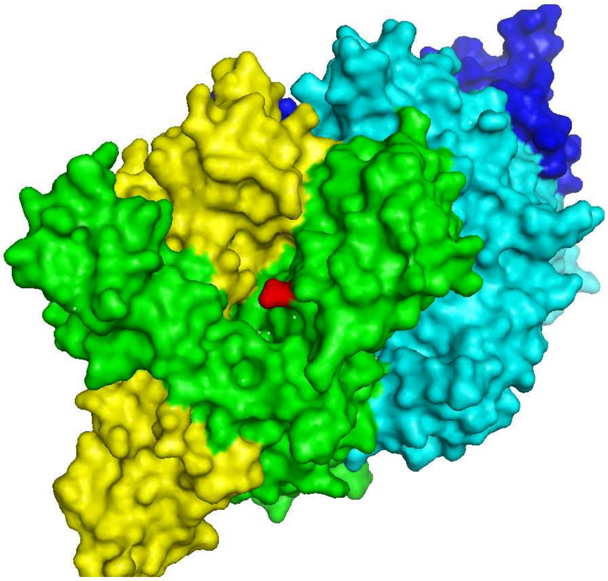


Figure 4.3: The structure of an oxidoreductase (from PDB code 1HW9) with a candidate flexible residue highlighted (image generated by PyMOL).

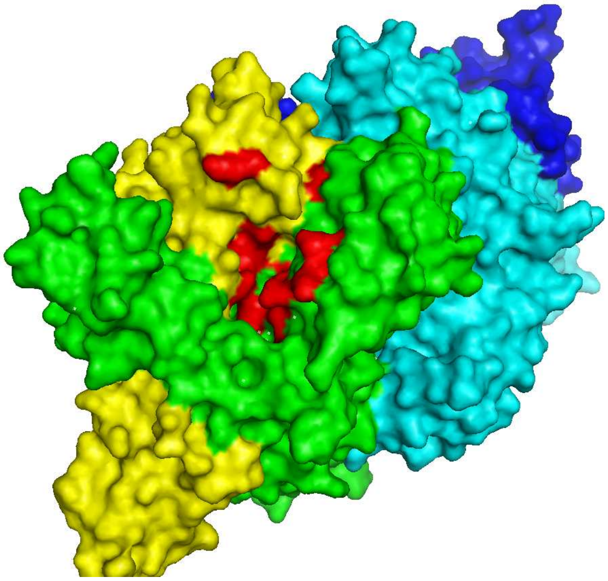


Figure 4.4: Same receptor of Figure 4.3 with all candidate flexible residues highlighted (image generated by PyMOL).

together. To do this we add extra variables for rotation around flexible bonds of each candidate side chain. We use the same local optimization engine used in eHiTS. We have also changed the scoring for this part to account for the flexibility of side chains.

4.1.4 Case Study: Carbonic Anhydrase

In this section we have demonstrated the applicability of the above proposed method for a simple case. Of course to do a thorough analysis of this new method first the side chain flexibility handling in RigiDock and PoseMatch levels should also be implemented and we have to try a big enough test set.

The test case is the human carbonic anhydrase II receptor bound to two different ligands. The relevant PDB codes are 1CIN and 1CIL. The significant change is in the HIS-64 residue. This residue is marked as flexible by the flexible side chain finder method of Section 4.1.2. The surface of the receptor from 1CIN is shown in Figure 4.5. The highlighted residue is HIS-64 and the structure of the bound ligand is also shown.

Figure 4.6 shows the same receptor with two different bound ligands. It is easy to see the clash between HIS-64 and the new ligand. In fact the structure of this new ligand is from PDB code 1CIL but the receptor structure is extracted from 1CIN. Binding of this ligand causes a conformational change in the receptor that is shown in Figure 4.7. The HIS-64 residue is moved. The difference between the two ligands is just the extra carbon atom in the ligand of 1CIL.

With the above flexible side chain detection procedure, we found 10 flexible residues which are: ASN-62, HIS-64, ASN-67, GLU-69, PHE-131, VAL-135, LEU-198, THR-200, CYS-206, ASN-244. Let us first have a closer look at the steric clash between 1CIL-ligand and 1CIN-receptor. The side chains close to the binding pocket are shown in Figure 4.8. The two receptor structures of 1CIN (blue carbons) and 1CIL (green carbons) are superimposed. The rotation of HIS-64 is visible in this figure. The shown ligand is from 1CIL. Note that the HIS-64 of 1CIN is too close to this ligand.

In our experiment we use the receptor 3D structure in 1CIN and use the ligand of 1CIL as the inputs of eHiTS. We already know that in the native structure, 1CIL, the HIS-64 residue is moved compared to 1CIN. The best output (i.e. highest score) ligand of our method is shown in Figure 4.9. In this figure that ligand with green carbons is

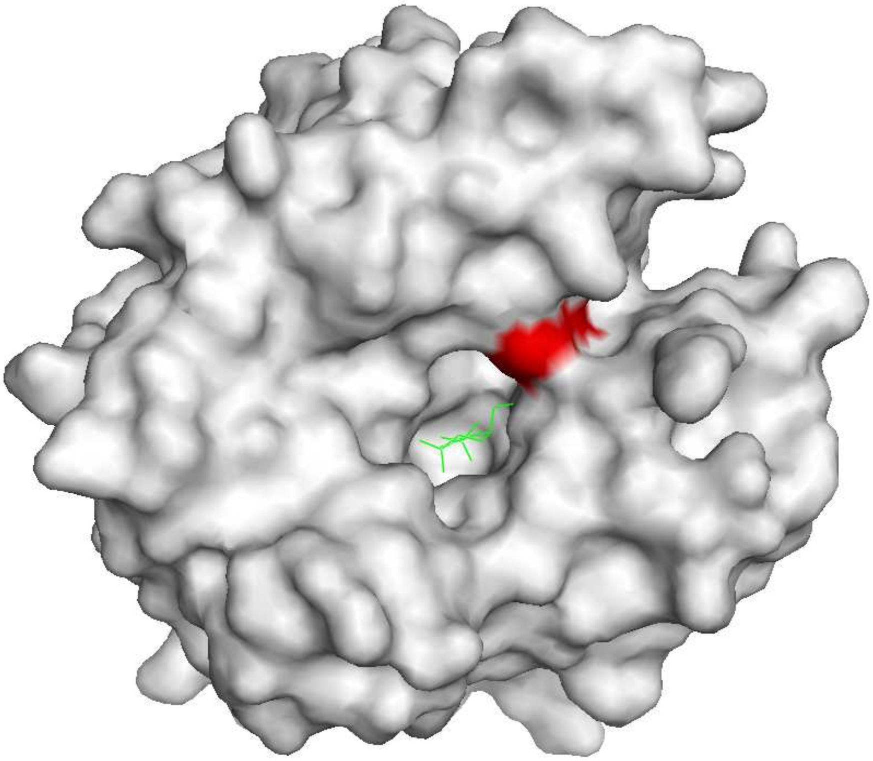


Figure 4.5: The human carbonic anhydrase II surface and a bound ligand (structures from PDB code 1CIN). The highlighted residue is HIS-64 (image generated by PyMOL).

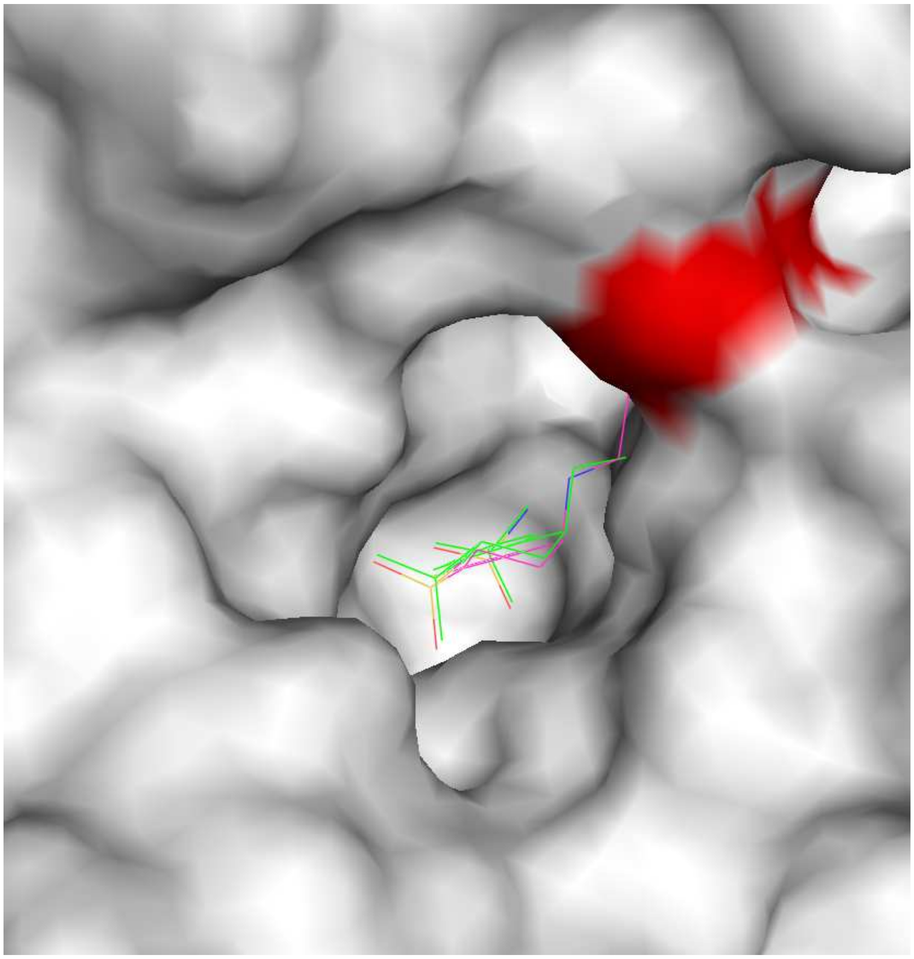


Figure 4.6: Binding of a similar ligand to carbonic anhydrase. The ligand from PDB structure 1CIL is overlaid on the receptor and ligand from 1CIN. All residues other than HIS-64 stay at the same location in the receptor of 1CIL, see Figure 4.7 (image generated by PyMOL).

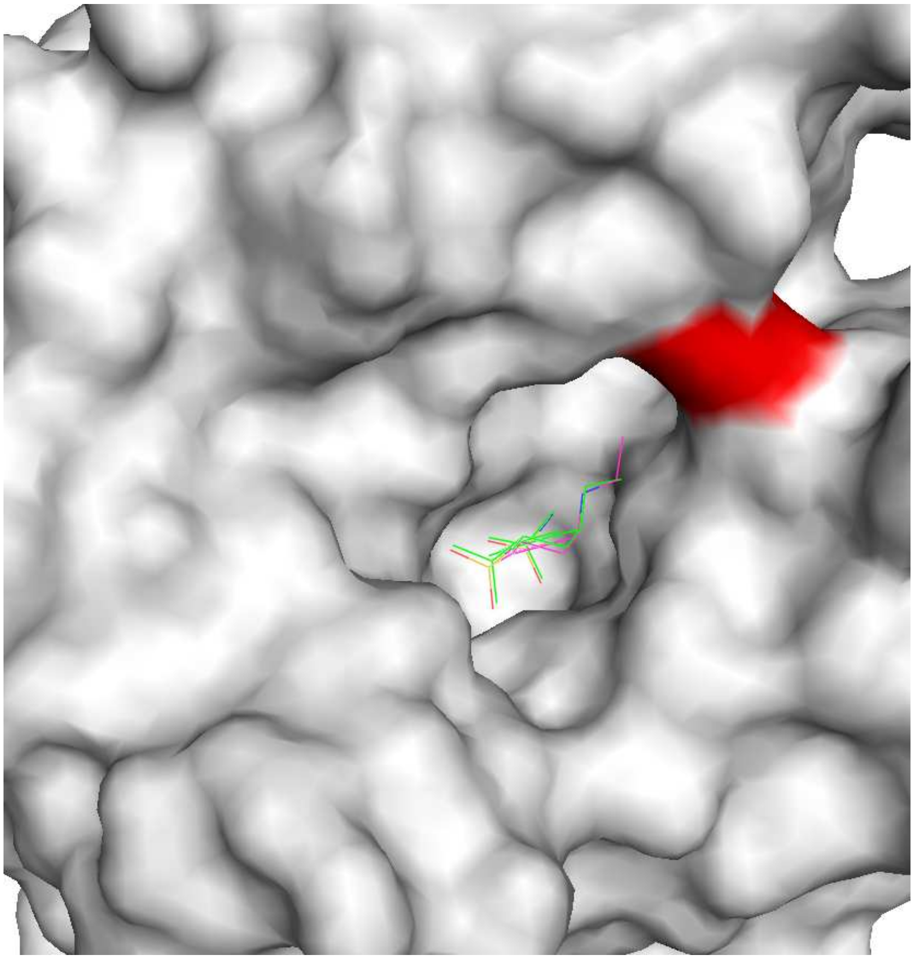


Figure 4.7: The location of HIS-64 is changed to accommodate for an extra carbon atom (the ligand of 1CIN is overlaid on the receptor and the ligand of 1CIL PDB; the image generated by PyMOL).

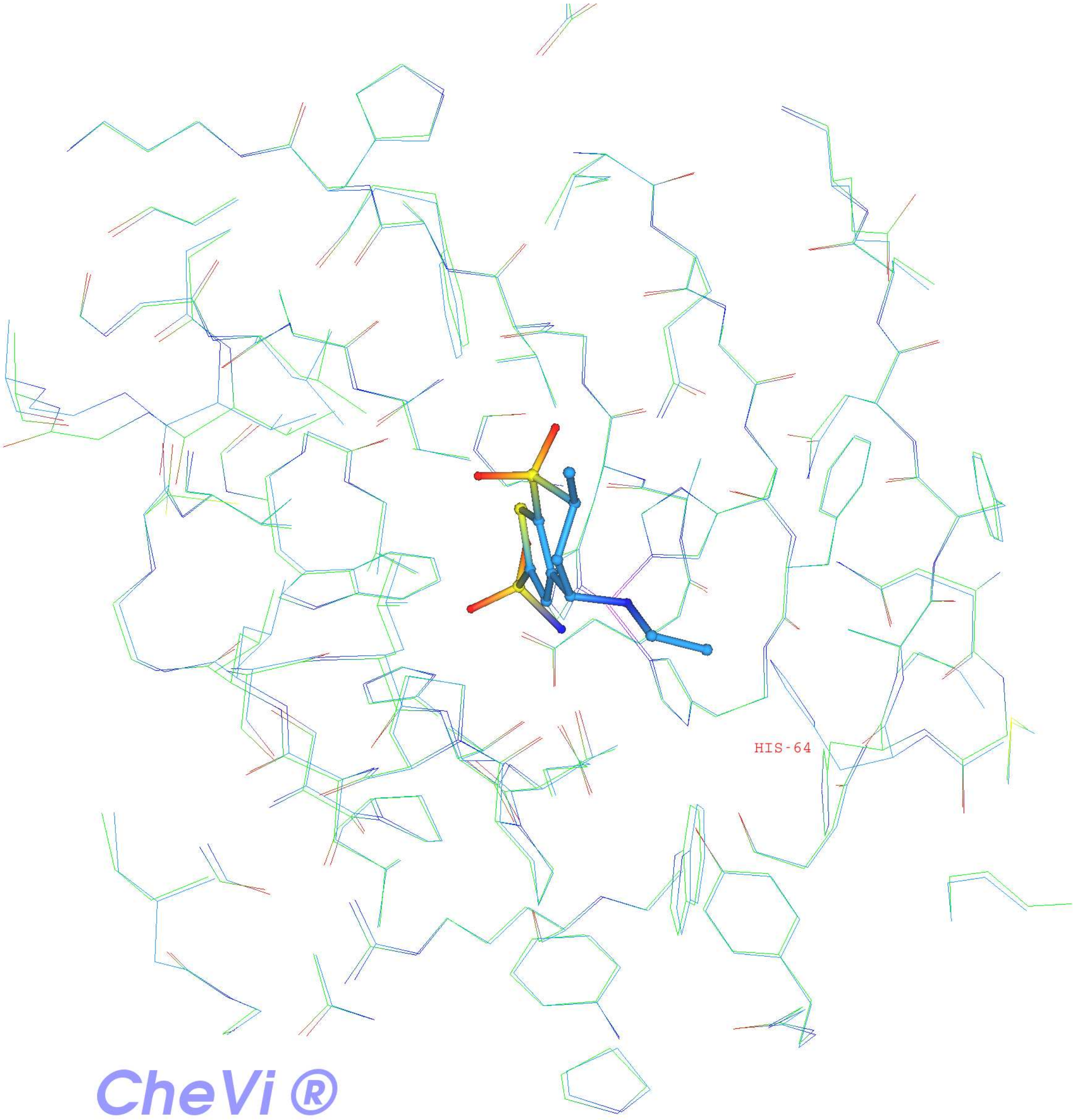


Figure 4.8: The binding site residues of carbonic anhydrase. The receptor structures of 1CIN (blue carbons) and 1CIL (green carbons) are superimposed on each other. The ligand is from 1CIL PDB code. The change in the conformation of the HIS-64 side chain is visible.

the best output and the one with blue carbons is from the native structure, 1CIL. The receptor atoms with blue carbons are the original 1CIN coordinates. Note the structural change in HIS-64 which is predicted by our method. Of course both the ligand and HIS-64 conformations are different than the native structure. There are other output structures (not the top-rank) that might be closer but the point we are trying to show here is the capability of this method in handling ligand and side chain flexibility together. There are other residues which are modified as well; green receptor carbon residues show these residues. One interesting change is in the LEU-198 residue. Note the difference between native ligand conformation and the predicted one. The hydrophobic rings of the predicted ligand is moved and the LEU-198 conformation change very well matches that move.

4.2 Crystal Structures Scoring Improvements

We have implemented a new search method for crystal structure prediction which was described in Chapter 3 (eCrySP). Although this tool is very well able to generate a structure close to the target in many of the cases, however such a structure is usually not very high in the score ranking. This is the biggest problem in selecting that structure for output as shown in Section 3.3.

We have spent a significant amount of time trying to improve the lattice energy estimation of crystal structures. However the results of Section 3.3 show that still there is a long way to go. We first started with the default eHiTS scoring function that is developed for protein-ligand binding. In this section we show our efforts in retraining the statistical weights of this scoring function based on the structures in Cambridge Structural Database (CSD). We describe why we decided to simplify this 4-dimensional scoring function and use the significantly simpler scoring function of Section 3.2. We show how we set the parameters of this function statistically. Finally we have argued that a more advanced function should be used to get better results and this is a future work. The experiments and implementations of this section is a joint work of the author and his PhD co-supervisor Zsolt Zsoldos.

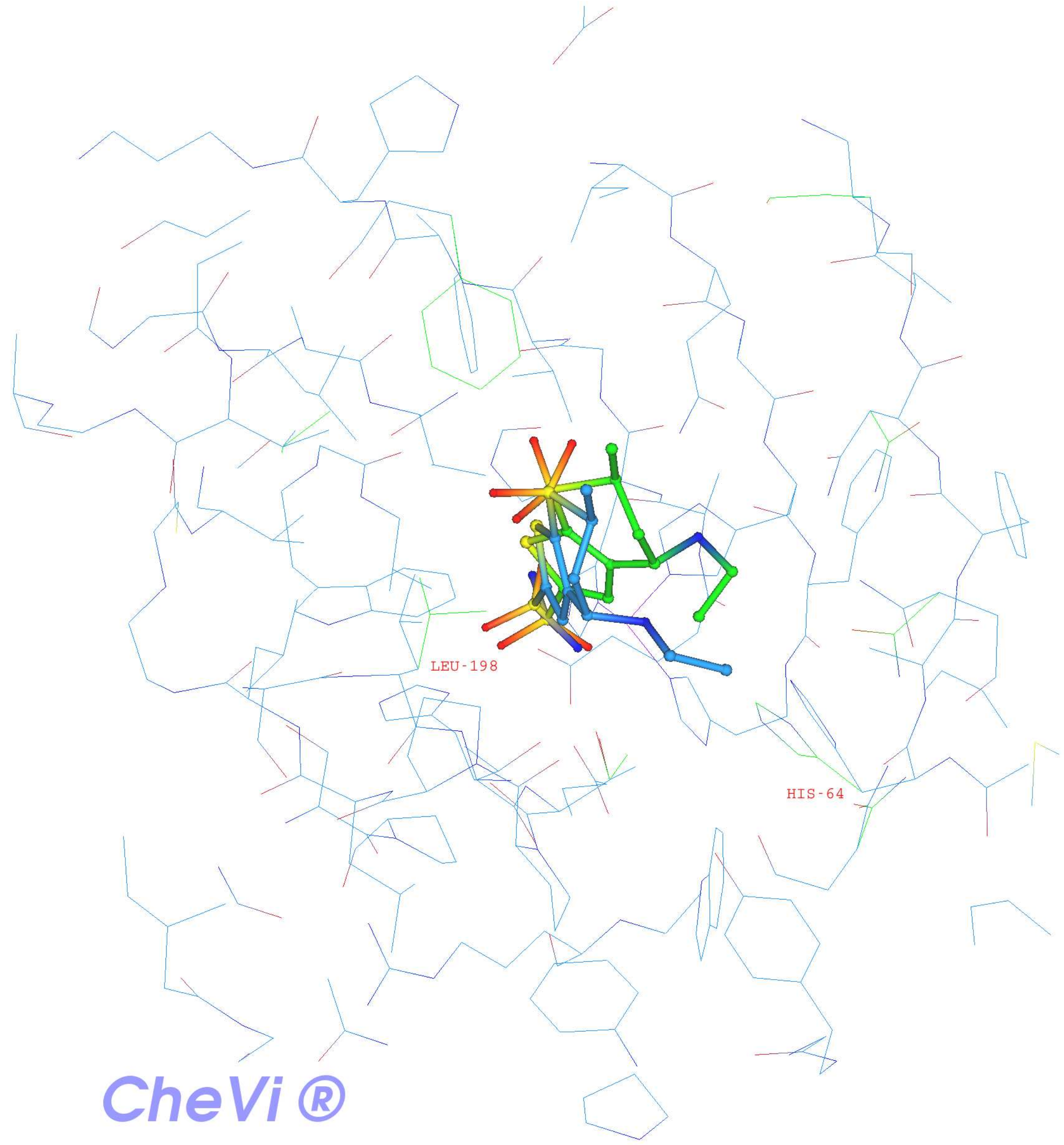


Figure 4.9: The best predicted ligand pose with the corresponding predicted conformational changes of the receptor residue.

4.2.1 Recognition of Real Crystal Structures among Decoys

Here is the main challenge for any scoring function: If the real target crystal structure and many decoy structures are ranked based on their score value, we are expecting the real crystal structure to be the top-rank. Of course there are always errors in crystallography data, so at least if we locally optimize the real crystal using the scoring function, we expect that optimized structure to be the top-rank. To simplify the descriptions we may use the target structure and the locally optimized target structure interchangeably here. Also we limit the scope of the work to rigid molecules only. As it was shown in Section 3.3 the results for rigid molecules are more reliable.

The main problem is that in many cases even the original structure itself may have a score higher than many of the structures generated by eCrySP which is an indication of problems in the scoring function. This was the motivation for trying to solve the next problem:

Definition 8 *For the scoring function s let $s(c)$ be the score of a proposed crystal structure c of molecule m. Also let p be a* rigid molecule crystal structure predictor (RCP) *that for an input molecule m and the scoring function s generates a set of candidate crystal structures $p_s(m)$. Then $r(m, s, p)$ is defined as the ratio of output structures $c_i \in p_s(m)$ that have $s(c_i) < s(c^\star)$, where $c^\star$ is the original crystal structure of m.*

Problem 2 *For a given RCP, p, find the scoring function s that minimizes $R(s) = \sum_{m \in M} r(m, s, p)$ where M is a set of molecules with fixed given conformations.*

Of course finding the minimizer of $R(s)$ is a vague goal because we never can even imagine all possible scoring functions. In fact our goal is to improve $R(s)$ and we used mainly this criteria to assess different scoring function ideas. The RCP engine p we used is mainly kept fixed and is based on the method described in Chapter 3. The following sections show the steps that we took to improve $R(s)$, starting from retraining of eHiTS scoring function (Sections 4.2.2 and 4.2.3) toward fundamental changes described in Section 4.2.4. We show the improvement for a small set of selected crystal structures from the Cambridge Structural Database (CSD) in Section 4.2.6.

There is an important point about Problem 2: Assessing scoring functions based on their ability in ranking a real structure among decoys is a common approach. However we believe that working with a fixed set of decoys is fundamentally wrong. Because even if a scoring function s ranks the real structure the highest among a set of decoys, it is usually very easy to use s in a structure optimizer engine and generate many decoy structures with better score values than the real structure. Therefor it is necessary to have a dynamic set of decoys that is generated by an accurate structure optimizer using the scoring function in question. Hofmann and Apostolakis have done an interesting scoring function training using similar data mining approaches to ours here [64]. However the above argument about the decoy generation also applies to their approach because they try to fit parameters such that the output scoring function can differentiate real crystal structures from a fixed set of decoys.

4.2.2 eHiTS Scoring Function

In the development of eCrySP we started with the scoring function used in SimBioSys's docking software eHiTS at the time. This is the scoring function that is used in the docking experiments of Section 2.5. That scoring function is based on recognizing interacting heavy-atom pairs and scoring each interaction. One interaction is described by two heavy-atoms (non-hydrogens) and two other points which are generally called *dummies*. These dummies could be hydrogen atoms, lone-pairs, π-electrons, etc. To fully describe the geometry of an interaction, the four parameters shown in Figure 4.10 are used: Distance d between the two heavy atoms, the two angles α and β between heavy-dummy vectors and the line connecting heavy atoms, and the dihedral angle δ. The relative interaction geometry of these four points cannot be fully described with less than four parameters but other options are available, for example distances between heavy and dummies from opposite sides may also fully describe the interaction geometry.

An interaction *configuration* consists of these four variables plus the types of heavy atoms and dummies participating in that interaction. The eHiTS scoring function is statistical-based, meaning that for each configuration, it assigns an energy value based on the number of times that configuration is observed in a database of structural data. In the case of protein-ligand binding this structure database was the protein-ligand complexes in PDB.

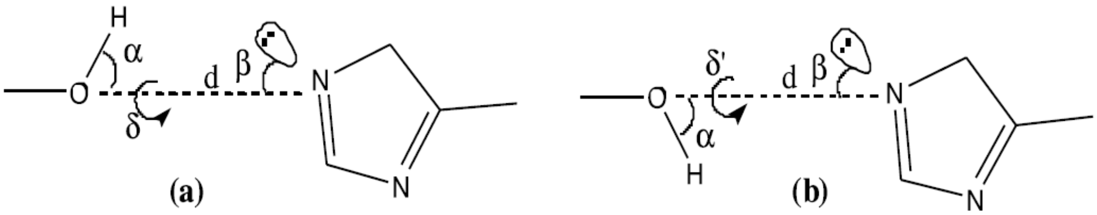


Figure 4.10: **(a).** The four geometric parameters to describe an interaction: Distance d, dummy angles α and β, and the dihedral angle δ. **(b).** The effect of changing δ while keeping other parameters fixed. (Image created by Zsolt Zsoldos and used by permission.)

4.2.3 Retraining with CSD Data

The original eHiTS scoring function with the weights trained for protein-ligand binding did not perform well for crystal structures, as expected. The first step in improving this scoring function was to retrain it with small molecule crystal structure data instead of proteins and bound ligands of PDB. For this purpose we used the crystal structures stored in CSD. One main advantage of structures in CSD is that in most cases the hydrogens are also stored and in many of them their placement is correct (we found obvious errors in some of the cases thought). Figure 4.11 shows sample graphs of the statistics collected for two types of interactions. The data set here is a subset of CSD containing around 27,000 structures with no metals and no ions. Probabilities show the likelihood of observing the corresponding configuration if we select an interaction in the whole set randomly. The distances are between the surface of two heavy atoms (using a generally shortened radii) not the actual nuclei.

One of the first observations in the graphs of Figure 4.11 is that the likelihood of a hydrogen-bond donor versus hydrogen-bond acceptor interaction is much higher than hydrogen-bond donor versus hydrogen-bond donor. This is an obvious fact but the point is that without any prior knowledge used in statistics collection, these facts can be inferred.

Figure 4.11 shows how the score value changes when one single configuration parameter is variable while others are kept fixed. Note that the probabilities shown in these graphs

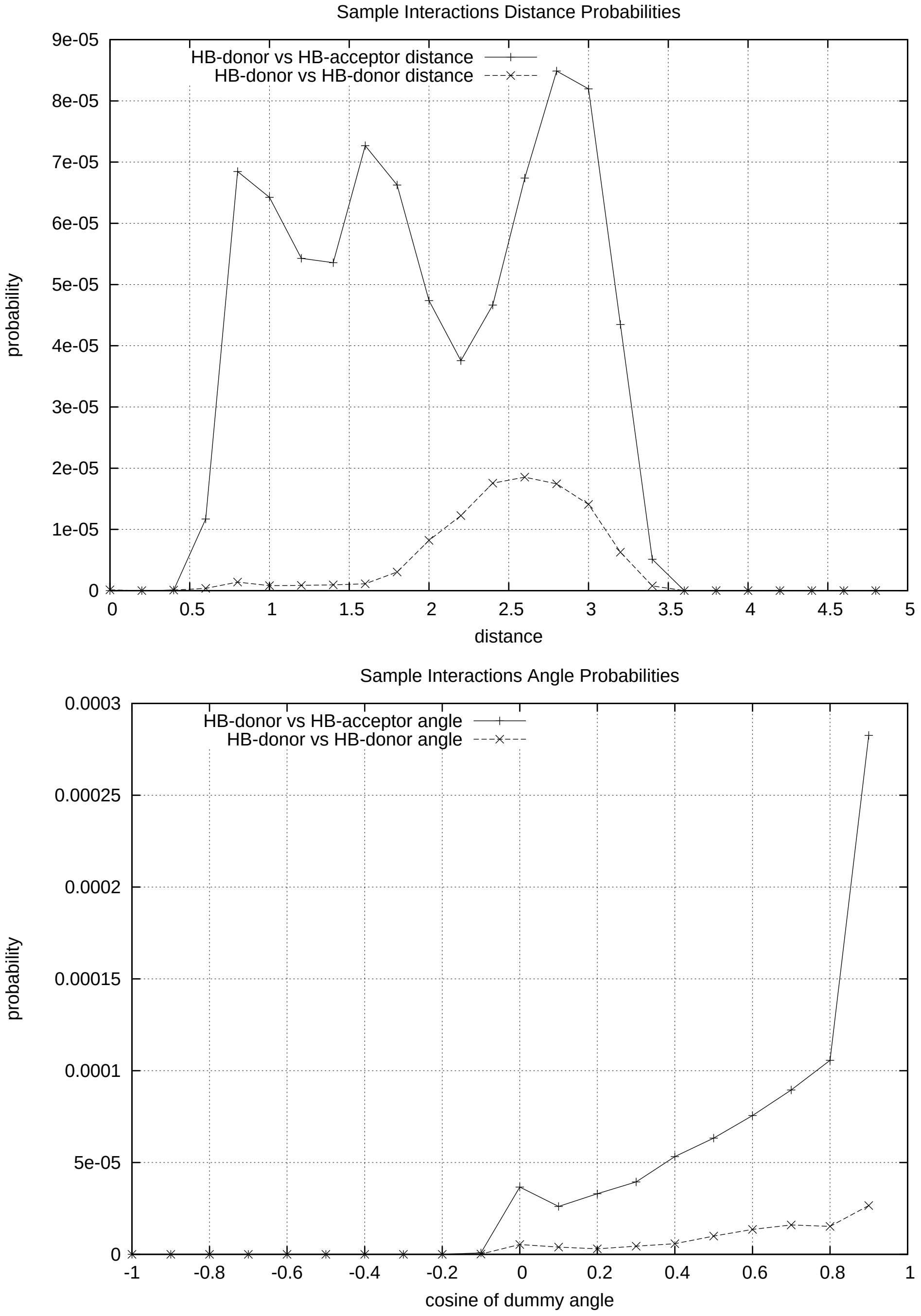


Figure 4.11: The likelihood of certain interaction configurations happening in a subset of structures from CSD (see the text for the description of variables).

do not directly translate to score values. Instead for each configuration the probability of observing that configuration for *random* crystal structures should also be calculated and then the logarithm of the ratio will be translated to a score value (this is inspired by Boltzman equation). Since this is not the final scoring function we came up with, we skip many details of how the statistics collection works. Instead we just talk about some of the drawbacks of this method.

If the bins we use for statistics collection are too big, the resulting scoring function would be too crude to differentiate between real and decoy structures. On the other hand if the bins are too small then the number of bins in the whole configuration space is huge and we may have an *over-training* problem because the number of interactions should be way more than the number of configurations to have a smooth realistic scoring function. In the case of protein-ligand binding, we solved this problem by using the *temperature factors* stored in PDB files. This way we could generate many interactions from a single one by using a probability distribution based on the temperature factors. For many reasons we fundamentally changed the scoring function for crystal structures: Firstly, this temperature factor based approach have some statistical drawbacks. Secondly we didn't find similar measures in CSD entries, and thirdly and most importantly during many trial and errors for training this scoring function, we came up with a different way of scoring which was giving more promising results in terms of the measure defined in Definition 8 and Problem 2 and that is the scoring function described in Section 3.2.

4.2.4 Fundamental Changes in Scoring

Following the poor performance of the eHiTS scoring function even with CSD-based training, we simplified the scoring function significantly. As it was shown in Section 3.2, the main components of this function is a van der Waals 6-12 component and an electrostatic term based on point charges. After tuning different weights of this function the results were significantly better. It is noteworthy that this simple model is similar to the W99 force field [134, 135, 136]. This and other similar scoring functions have been used by some other CSP projects as well [35, 37, 93, 83].

Here we report our efforts in setting the parameters of the van der Waals term (3.5).

Let us look at this term again:

$$\mathrm{vdw}(a,b) = \epsilon_{a,b}\left(\frac{r_{a,b}}{d_{a,b}}\right)^6\left(\left(\frac{r_{a,b}}{d_{a,b}}\right)^6 - 2\right). \tag{4.1}$$

In this equation $r_{a,b}$ is the ideal distance between atoms a and b, i.e., the distance at which the minimum $-\epsilon_{a,b}$ is reached. Note that $r_{a,b}$ is sometime approximated as $r_a + r_b$, i.e., the sum of the van der Waals radii of the two atoms. While we also use similar statistical methods to set each atom radius, however we do not impose such constraint here. One benefit of this approach is that in some case, e.g., a hydrogen bond donor and acceptor pair, the two atoms may go closer than the sum of their van der Waals radii and we should not penalize such interactions.

4.2.5 Finding Interacting Pairs

Let us denote actual atoms by letters $a, b, c, \ldots$ and atom types by letters $t, u, v, \ldots$ As it was briefly mentioned in Section 3.2, to find the ideal distance between two atom types t and u (say a carbon and an oxygen), the idea is to look at close atom pairs of type t and u in our dataset (which is a subset of CSD) and determine the likelihood of a certain distance happening (to be more precise the likelihood of a distance *range* is determined). In other words we find out $\Pr(d_1 \le d_{t,u} < d_2)$ in our dataset. Then we compare this with the approximate likelihood of a certain distance (range) happening in a random structure, i.e., $\Pr(d_1 \le d'_{t,u} < d_2)$. Based on the ratio of these two probabilities we determine the ideal distance range. The ratio in the best distance range is also used to set $\epsilon_{t,u}$ as described in Section 3.2. It is important to include the random variable $d'_{t,u}$ in our calculations since bigger distances simply have a higher chance of occurring in a crystal structure. This is because for $d_2 > d_1$, the sphere shell between two spheres of radii d_1 and $d_1 + \delta$ is smaller than that of radii d_2 and $d_2 + \delta$.

One of the issues is what pairs to consider when counting distances; in other words what constitutes an interacting pair? For example if atom b is between a and c, then should the interaction between a and c be also considered? For this simplistic case the answer is probably no but it is easy to imagine cases when the distinction is not so obvious. We tried different methods which are explained in Section 4.2.6 along with the corresponding experiments evaluating the goal function of Problem 2.

The other issue is how to approximate the probabilities in a random structure. One idea is to generate random neighbor molecules around a fixed molecule and count the interactions happening, we followed this idea mainly in eHiTS scoring function retraining of Section 4.2.3. Another idea that we have used here is to estimate that probability based on the exposed surface of each atom. In other words calculate the surface of an atom that is not buried inside the molecule. Then the interaction probability of two atoms a and b is proportional to their exposed surfaces. One minor point here is that how we determine the surface of an atom because this in fact is dependent on what radius we assign to each atom. This problem can be solved by iteratively estimating atom radii, calculating best pair distances, and readjusting atom radii again.

4.2.6 Experiments

In this section we review some of the steps in improving the parameters of term (4.1) of the scoring function. The criteria we use in comparing the results of these experiments is the goal function in Problem 2. We should emphasize that this was not the only criteria we used in our decisions. In fact at many steps we were also looking at the shape of the statistics graphs to see whether they make sense from a physical chemistry point of view, however we won't get into those details here.

Let us look at Problem 2 again: The RCP or the search engine was almost fixed in the experiments of this section. In fact it was the version of eCrySP for rigid molecules available at the time of these experiments. This RCP generates many (in some cases hundreds of thousands) of structures and based on the scoring function, it selects a small subset of them for local optimization as explained in Chapter 3. This subset was of size 100 in most of our experiments here. We add the real structure to this subset, locally optimize all structures and sort them based on their score values. We expect the original structure to be the first for all cases in our test set p. Therefor if the scoring function s is ideal then $R(s) = 0$. This is the idea behind our evaluation method that is based on Problem 2.

Following is the list of different methods in determining interacting atoms and estimating the lowest energy $\epsilon_{a,b}$ in (4.1). A name is assigned to each experiment for easier references to them. This list shows the step by step improving of the statistics collection method.

- HEAVY: The first experiment that only includes heavy atoms (i.e., no hydrogen or lone-pair included). Between two neighbor molecules, all pairs of atoms where included for statistics collection. The parameter $\epsilon_{a,b}$ was set to 1 for all pairs.

- DUMMY: The same experiment as HEAVY by including dummies (hydrogens and lone-pairs) in statistics collection and score calculation.

- THRESH: Same as DUMMY but ignoring pairs beyond a certain distance threshold.

- LOG: Using Boltzman-like distributions to determine $\epsilon_{a,b}$, i.e., logarithm of the ratio of the experimental and expected values in the best distance range was used.

- LONG: Increasing the distance threshold.

- NO_OFFSET: Same as LONG but with an offset added to $\mathrm{vdw}(a, b)$ of (4.1) such that the value of $\mathrm{vdw}(a, b)$ is automatically zero at the distance threshold used to find interacting atoms. Prior to this experiment there was a jump to zero at the threshold.

- DIRVECT [final]: Same as NO_OFFSET but with a more sophisticated method used in finding interacting atom-pairs. In this method, a set of vectors similar to the ones shown in Figure 3.4 was used for each atom. For an atom a these vectors were placed on atom nucleus. An atom b was considered to be interacting with a, if there was a vector from a's nucleus that was hitting b's van der Waals surface without hitting any other atom surface at a shorter distance.

Table 4.1 summarizes the results of these experiments based on the criteria of Problem 2. The set of structures used for this table is a superset of rigid structures listed in Table 3.4. The final method is DIRVECT in which a set of vectors from atom center to many directions around the atom is used for finding interacting atoms. The scales $\epsilon_{a,b}$ are also calculated as in the LOG experiment. As it can be seen, on average, there are less than 2% of the generated structures that have a score better than the original structure (after local optimization). This is a pretty good result, however note that the quality of the search engine RCP has a direct effect on this number. To check this effect we used a 0.5Å grid in the sampling of neighbor molecules instead of the default 1Å grid (see Section 3.1.2). This means that the number of structures that are visited is almost 8 times (i.e., $(1/0.5)^3$) and

Experiment Name	$R(s)$ % (see Problem 2)	optimized rmsd Å
HEAVY	20.12	0.85
DUMMY	3.54	0.53
THRESH	2.84	0.39
SCALE	3.37	0.39
LOG	2.92	0.37
LONG	2.30	0.33
NO_OFFSET	2.09	0.32
DIRVECT	1.83	0.23

Table 4.1: Scoring function tuning experiments summary; advances in determination of interacting atom pairs and the way the score is calculated.

the accuracy of the search engine is higher. Using this new RCP, the $R(s)$ was increased to 3.92%. That is a clear indication of the fact that the search engine used to generate decoy structures is very important in improving a scoring function. As this experiment shows, we should not be too optimistic about the final results of DIRVECT row in Table 4.1.

The last column of Table 4.1 shows how much the real structure from CSD changes after the local optimization. This is another measure of how good the scoring function is. In fact in the ideal case this number should be very close to zero (there are always errors in the experimental structure determination methods too so this could never be exactly zero).

4.3 Conclusion

The search method proposed in Chapter 2 for the docking problem works quite well when the binding site structure is known and rigid. It is also a well known fact that proteins undergo structural changes in the ligand binding process [122]. Therefor the next natural step in extending the search method is to include receptor flexibility. In Section 4.1, we showed how the ideas of Chapter 2 can be extended to address side-chain flexibility. Also the results of some very preliminary implementations were also demonstrated.

Both for the docking problem and crystal structure prediction, one of the major problems is that the current search methods are able to find structures very close to the target structures, however these structure are not ranked high enough in the ordered list of score values. This was demonstrated in Section 2.5 and Section 3.3.

We discussed some of our efforts in improving our scoring function for the crystal structure prediction problem. We defined a quantitative measure to evaluate different scoring functions and with that measure we showed the improvement gained in this process of changing and retraining the scoring function. Although the final scoring function coming out of this process is performing significantly better in ranking structures close to the target, however the final results could be significantly better with a more accurate energy estimation function. Table 3.4 of Section 3.3 clearly shows this problem. Therefor we think that the next step in advancing eCrySP is yet to improve the scoring function. More sophisticated functions similar to the ones mentioned in Section 1.2 should be used because probably we have already reached the limits of W99-like methods.

Another major area to extend eCrySP is to model multiple molecular units in the asymmetric unit cell, i.e., $Z' > 1$. This will enable us to predict not just the structure of co-crystals but also hydrates and salts.

Appendix A: The List of 829 PDB Codes Used in Experiments

10gs 1ssq 1tnh 1tnk 1uio 1utj 1uvs 1a28 1bap 1bl7 1c5s 1c84 1dbb 1drj 1dwb
1eap 1i9n 1l2s 1o2r 1o3p 1stp 1uto 1uz1 1v0n 1v2r 3mct 3pcc 3pcg 3tmn 1a99
1akw 1mmp 1o0m 1o3k 1owh 1w8l 1wdn 1xbb 5std 6tmn 1a9u 1ctt 1dhj 1mrs 1nc1
1nc3 1nc9 1nw5 1rdj 1sm2 1tkb 1ttm 1utl 1wei 1wm1 2cht 2qwg 5xia 1abe 1aha
1aqw 1i80 1i9p 1kdk 1ksn 1o33 1oar 1okn 1q54 1qb9 1srj 1tet 1utn 1uv6 1v0l
1v2l 1v2v 1v79 2ak3 8abp 8cpa 1abf 1aj7 1c3x 1c5n 1c5y 1cea 1d7j 1dhf 1kui
1sbr 1thl 1acj 1b32 1db1 1ecv 1f9g 1ghy 1gi7 1k22 1mrk 1n1m 1n3i 1n4h 1o0n
1o86 1acl 1f57 1g3e 1hn2 1hri 1jt1 1ngp 1njd 1xid 1zsb 2phh 2pri 2xis 8icd
1acm 1cin 1dd7 1gi8 1kr3 1lah 1lcp 1mts 1rt2 1u1w 1ugp 1ulb 1uou 1utp 1wap
1xie 1xjd 2h4n 2sim 9aat 1aco 1b74 1e70 1epb 1g52 1ha2 1koj 1mbi 1mcq 1niu
1nsd 1p1n 1p57 1qaw 1qft 1qh7 1qk3 1r1j 3fx2 1add 1bgq 1bnw 1cnw 1n46 1n5r
1pu8 1pxp 1qf1 2pk4 2tmn 5yas 6abp 6rnt 1ado 1cla 1gj6 1hyo 1if7 1kyv 1nje
1pxo 1q8u 1qpe 1r1h 1ro6 1trk 1vfn 1vrh 220l 223l 25c8 6enl 1a42 1cps 1fh8
1fkw 1fl3 1g4j 1j16 1jap 1n1v 1o3l 1q8t 1qf2 3amv 1af2 1bra 1bzm 1c88 1cru
1dy4 1f8b 1gj9 1lyl 1mh5 1stc 1tmt 1ag9 1gj5 1k1l 1o2j 1o3i 1oss 1oyt 1q4w
1q7a 1r5y 1r9l 1rbp 1rpj 1s38 1s39 1sqa 1ai4 1ax0 1bnn 1cnx 1fv0 1g53 1hdy
1hlk 1imb 1ivd 1jqd 1okm 1ptv 1q95 1qbo 1uw6 1uz4 1v2j 1v2m 1v78 2drc 1ai5
1bn4 1ebg 1ejn 1fkh 1kv1 1lag 1lan 1m5w 1ndw 1o2u 1tsy 1ai7 1ajq 1azl 1los
1o3d 1owe 1sqo 1trd 1xkb 3pce 3pgh 4fab 6upj 8xia 1aid 1gi9 1o2g 1o8b 1rdm
1rej 1srg 1tng 1tnj 1toi 1v2k 1ydt 3hvt 3nos 7tim 1ajn 1b46 1cbx 1drf 1gdo
1gi2 1hsb 1l7s 1ldm 1lgt 1lke 1lrh 1m0n 1mnc 1o2y 1xka 3cpa 4tln 9abp 1ajp
1b3g 1d09 1drk 1e2k 1e5a 1erb 1g32 1jet 1lpz 1o2n 1o2z 1tka 1uj6 5tln 1akt
1m6p 1o2s 1o39 1ogx 1om1 1p19 1tok 1utm 1ux7 1v2s 1vot 1w8m 1yej 5cpp 5icd
1alw 1byt 1c1e 1c1r 1c83 1dr1 1f5l 1fj4 1fkg 1kc7 1lpk 1oe7 1xbo 1a4k 1dg5
1etz 1iy7 1k4g 1m2q 1me8 1mup 1n2v 1nvq 1o2w 1v7a 3pck 456c 4tpi 7abp 7std
1aoe 1bnu 1c1u 1f0r 1gj7 1l83 1laf 1lbf 1nfy 1o3e 1oe8 1yei 2usn 1apb 1b3l
1d5r 1f8e 1gjd 1ndz 1pr5 1q63 1qf0 1apu 1o3b 1ow4 1p1o 1p1q 1pb8 1pb9 1pbd
1pfu 1qq9 1rdl 1rdn 1rgk 1swn 1txr 2ypi 3ert 1atl 1g46 1gpn 1gyy 1ik4 1ikt
1n9m 1nvr 1o2x 1udt 1yds 2dri 2gss 1avn 1br5 1eoc 1ew8 1f2o 1fki 1fpu 1g36
1ghv 1iup 1j07 1o37 1phf 1pkx 1q91 1rob 1tnl 1tph 2aad 2std 3cla 1azm 1byg
1c2t 1gca 1gcz 1ghw 1hp0 1hyt 1k9s 1kv5 1lbl 1ndv 1nvs 1o2k 1o3g 1okl 1rgl
1tdb 1b0h 1d6v 1g48 1if8 1lyx 1m2p 1mfi 1o2q 1o32 1oba 1oko 1os5 2r07 1b1h
1efy 1gaf 1ii5 1j01 1j15 1jaq 1m5j 1mjj 1qk4 1rm8 830c 1b3h 1d7i 1dbm 1g3d
1gyx 1h46 1kuk 1tog 1tyl 1ukz 1v2u 1vpo 1xff 1ydb 2cpp 2ctc 2lgs 3gpb 3kiv
5abp 5cna 6rsa 1b40 1cqp 1hfc 1m2x 1nis 1nli 1no6 1pa9 1pme 1re8 1upf 1v2n
1yda 2bza 2dbl 4cox 4lbd 6std 8atc 1a4m 1axz 1bn3 1ckp 1ctu 1d1p 1ett 1nfw
1phd 1phg 1pot 1ps3 1pu7 1pxn 1q65 1ta2 1b42 1bnq 1gj4 1j14 1j17 1ndy 1o2t
1o3h 1owd 2csn 3pcb 3pch 3tpi 4sga 1b6h 1dg9 1f5k 1f8d 1gj8 1m0q 1mmq 1nu3
1o2p 1o34 1o5r 1qpb 1qy1 1qy2 1r0p 1rd4 3erd 3mth 3pcj 3std 4rsk 1b6n 1e2p
1f2p 1fmo 1ftm 1g1d 1gjb 1n1t 1o30 2r04 6cpa 1b6o 1die 1e2l 1e6q 1f3e 1f4x
1h4n 1jao 1m1b 1mdr 1mld 1moq 1mtw 1o35 1pph 2amv 1b7h 1df8 1f4e 1ghb 1ghz
1gpk 1h1s 1jgl 1jmi 1pzp 1qbv 4cts 4std 5enl 1b8y 1d4p 1hak 1kug 1sqt 1uml
2tsc 1b9j 1gi6 1h4w 1h9z 1qx1 1qxl 2gbp 2izl 2tpi 3pcf 3ptb 4cla 1b9v 1f74
1mu6 1o3j 1oxq 1ta6 1bcd 1bnt 1c5q 1c5x 1c86 1com 1dbj 1dhi 1o2h 1o2v 1w3j
1wht 1yee 1bcj 1cim 1coy 1ctr 1gi5 1ivb 1jmf 1mmr 1njc 1qcf 1qkb 1sw1 1swk
1toj 1uvt 1a4q 1fig 1fkx 1flr 1g45 1hsl 1i7z 1iih 1m2r 1mcr 1o3f 1ofz 1oim
1os0 1bcu 1bnv 1dvz 1ecq 1f0s 1gi1 1gja 1ive 1jys 1k4h 1li2 1li6 1lna 1o2o
1o38 1p28 1pgp 1bky 1cx2 1dog 1dzk 1e66 1ezq 1li3 1lnm 1lst 1m0o 1mu8 1o3c
1osv 1q8w 1qpq 1qy5 1tuf 1ydd 2aac 2ada 2cmd 2qwc 2yhx 1bm7 1br6 1f3d 1fhd
1fsa 1g85 1gi4 1grp 1lgw 1lhw 1lqe 1nf8 1nja 1pbq 1pr1 1q1g 1q66 1sqn 1sre
1swp 1uj5 2pcp 2qwd 3mag 3pcn 4aah 4dfr 1bn1 1cbs 1fgi 1g4o 1gpy 1n43 1nfu
1oif 1onz 1rnt 1swr 1tni 1uho 2ans 2qwk 5upj 6tim 1a50 1c5o 1c5p 1c5t 1c87
1d3h 1did 1e2n 1ec9 1ew9 1f8c 1gjc 1ndj 1nw7 1ydr 2adm 1a69 1akr 1f0u 1inc
1qbq 1tom 4ts1 966c 1a6w 1c12 1e3v 1ghx 1ivc 1jmg 1mmb 1qan 1rql 1rzy 2ack
2cgr 2xim 4tim 1a7t 1c5c 1ce5 1ceb 1cil 1dl7 1g54 1hwr 1k21 1sw2 1swg 1v48
2mas 2mcp 2rkm 4fbp

References

[1] CCDC/Astex test set, http://www.ccdc.cam.ac.uk/products/life_sciences/gold/validation/astex/.

[2] Materials Studio, version 4.4, 2009, http://accelrys.com/products/materials-studio.

[3] OpenBabel, version 1.2, 2009 http://openbabel.org.

[4] ChemAxon, version 5.2, 2009, http://www.chemaxon.com.

[5] CSD-Statistics, 2008, http://www.ccdc.cam.ac.uk/products/csd/statistics/.

[6] R. Abagyan and M. Totrov. Biased probability Monte Carlo conformational searches and electrostatic calculations for peptides and proteins. *Journal of Molecular Biology*, 235(3):983–1002, 1994.

[7] R. Abagyan, M. Totrov, and D. Kuznetsov. ICMa new method for protein modeling and design: applications to docking and structure prediction from the distorted native conformation. *Journal of Computational Chemistry*, 15(5):488–506, 1994.

[8] P. K. Agarwal, S. Har-Peled, and R. Varadarajan. Approximating extent measures of points. In *Proceedings of the Twelfth ACM-SIAM Symposium on Discrete Algorithms*, pages 148–157, 2001.

[9] F. H. Allen. The cambridge structural database: a quarter of a million crystal structures and rising. *Acta Crystallographica Section B*, 58(1 Part 3):380–388, 2002.

[10] R. E. Amaro, R. Baron, and J. A. McCammon. An improved relaxed complex scheme for receptor flexibility in computer-aided drug design. *Journal of Computer-Aided Molecular Design*, 22(9):693–705, 2008.

[11] A. C. Anderson, R. H. ONeil, T. S. Surti, and R. M. Stroud. Approaches to solving the rigid receptor problem by identifying a minimal set of flexible residues during ligand docking. *Chemistry & Biology*, 8(5):445–457, 2001.

[12] J. Bajorath. Integration of virtual and high-throughput screening. *Nature Reviews Drug Discovery*, 1(11):882–894, 2002.

[13] M. Baroni, G. Cruciani, S. Sciabola, F. Perruccio, and J. S. Mason. A common reference framework for analyzing/comparing proteins and ligands. fingerprints for ligands and proteins(FLAP): Theory and application. *Journal of Chemical Information and Modeling*, 47(2):279–294, 2007.

[14] V. E. Bazterra, M. B. Ferraro, and J. C. Facelli. Modified genetic algorithm to model crystal structures. I. Benzene, naphthalene and anthracene. *The Journal of Chemical Physics*, 116:5984, 2002.

[15] V. E. Bazterra, M. B. Ferraro, and J. C. Facelli. Modified genetic algorithm to model crystal structures. II. Determination of a polymorphic structure of benzene using enthalpy minimization. *The Journal of Chemical Physics*, 116:5992, 2002.

[16] V. E. Bazterra, M. B. Ferraro, and J. C. Facelli. Modified genetic algorithm to model crystal structures: III. Determination of crystal structures allowing simultaneous molecular geometry relaxation. *International Journal of Quantum Chemistry*, 96(4):312–320, 2004.

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

[18] J. Bernstein. *Polymorphism in molecular crystals.* Oxford University Press, USA, 2002.

[19] S. Brodersen, S. Wilke, F. J. J. Leusen, and G. Engel. A study of different approaches to the electrostatic interaction in force field methods for organic crystals. *Physical Chemistry Chemical Physics*, 5(21):4923–4931, 2003.

[20] S. Brodersen, S. Wilke, F. J. J. Leusen, and G. E. Engel. Comparison of static and fluctuating charge models for force-field methods applied to organic crystals. *Crystal Growth & Design*, 5(3):925–933, 2005.

[21] N. Budin, N. Majeux, and A. Caflisch. Fragment-based flexible ligand docking by evolutionary optimization. *Biological Chemistry*, 382(9):1365–1372, 2001.

[22] H. A. Carlson. Protein flexibility and drug design: how to hit a moving target. *Current Opinion in Chemical Biology*, 6(4):447–452, 2002.

[23] M. Cecchini, P. Kolb, N. Majeux, and A. Caflisch. Automated docking of highly flexible ligands by genetic algorithms: A critical assessment. *Journal of Computational Chemistry*, 25(3):412–422, 2004.

[24] S. Chakravarty and K. K. Kannan. Drug-protein interactions. Refined structures of three sulfonamide drug complexes of human carbonic anhydrase I enzyme. *Journal of Molecular Biology*, 243:298–309, 1994.

[25] T. M. Chan. Approximating the diameter, width, smallest enclosing cylinder, and minimum-width annulus. In *Proceedings of the 16th Annual Symposium on Computational Geometry*, pages 300–309, 2000.

[26] T. M. Chan. Faster core-set constructions and data stream algorithms in fixed dimensions. In *Proceedings of the 20th Annual Symposium on Computational Geometry*, pages 152–159, 2004.

[27] T.M. Chan and B.S. Sadjad. Geometric optimization problems over sliding windows. *International Journal of Computational Geometry and Applications*, 16:145–157, 2006.

[28] S. R. Chemburkar, J. Bauer, K. Deming, H. Spiwek, K. Patel, J. Morris, R. Henry, S. Spanton, W. Dziki, W. Porter, et al. Dealing with the impact of ritonavir polymorphs on the late stages of bulk drug process development. *Organic Process Research & Development*, 4(5):413–417, 2000.

[29] J. A. Chisholm and S. Motherwell. *COMPACK*: a program for identifying crystal structure similarity using distances. *Journal of Applied Crystallography*, 38(1):228–231, 2005.

[30] H. Claußen, C. Buning, M. Rarey, and T. Lengauer. FlexE: efficient molecular docking considering protein structure variations. *Journal of Molecular Biology*, 308(2):377–395, 2001.

[31] M.L. Cohen. Novel materials from theory. *Nature*, 338(6213):291–292, 1989.

[32] C. R. Corbeil, P. Englebienne, and N. Moitessier. Docking ligands into flexible and solvated macromolecules. 1. Development and validation of FITTED 1.0. *Journal of Chemical Information and Modeling*, 47(2):435–449, 2007.

[33] S. Datta and D. J. W. Grant. Crystal structures of drugs: advances in determination, prediction and engineering. *Nature Reviews Drug Discovery*, 3(1):42–57, 2004.

[34] G. M. Day, J. Chisholm, N. Shan, W. D. S. Motherwell, and W. Jones. An assessment of lattice energy minimization for the prediction of molecular organic crystal structures. *Crystal Growth & Design*, 4:1327–1340, 2004.

[35] G. M. Day, T. G. Cooper, A. J. Cruz-Cabeza, K. E. Hejczyk, H. L. Ammon, S. X. M. Boerrigter, J. S. Tan, R. G. Della Valle, E. Venuti, J. Jose, et al. Significant progress in predicting the crystal structures of small organic molecules-a report on the fourth blind test. *Acta Crystallographica Section B: Structural Science*, 65(2):107–125, 2009.

[36] G. M. Day, W. D. Motherwell, and W. Jones. Beyond the isotropic atom model in crystal structure prediction of rigid molecules: atomic multipoles versus point charges. *Crystal Growth & Design*, 5(3):1023–1033, 2005.

[37] G. M. Day, W. D. S. Motherwell, H. L. Ammon, S. X. M. Boerrigter, R. G. Della, T. C. Lewis, H. Nowell, A. Torrisi, H. A. Scheraga, Y. A. Arnautova, et al. A third blind test of crystal structure prediction. *Acta Crystallographica Section B*, 61(5):511–527, 2005.

[38] G. M. Day, W. D. S. Motherwell, and W. Jones. A strategy for predicting the crystal structures of flexible molecules: the polymorphism of phenobarbital. *Physical Chemistry Chemical Physics*, 9(14):1693–1704, 2007.

[39] R. L. DesJarlais, R. P. Sheridan, J. S. Dixon, I. D. Kuntz, and R. Venkataraghavan. Docking flexible ligands to macromolecular receptors by molecular shape. *Journal of Medicinal Chemistry*, 29(11):2149–2153, 1986.

[40] J. Downing, N. Day, and P. Murray-Rust. CrystalEye: From Desktop To Data Repository . In *Third International Conference on Open Repositories*, Southampton, UK, April 2008.

[41] T. J. A. Ewing and I. D. Kuntz. Critical evaluation of search algorithms for automated molecular docking and database screening. *Journal of Computational Chemistry*, 18(9), 1997.

[42] T. J. A. Ewing, S. Makino, A. G. Skillman, and I. D. Kuntz. DOCK 4.0: search strategies for automated molecular docking of flexible molecule databases. *Journal of Computer-Aided Molecular Design*, 15(5):411–428, 2001.

[43] P. Ferrara, H. Gohlke, D. J. Price, G. Klebe, and C. L. Brooks. Assessing scoring functions for protein-ligand interactions. *Journal of Medicinal Chemistry*, 47:3032–3047, 2004.

[44] A. M. Ferrari, B. Q. Wei, L. Costantino, and B. K. Shoichet. Soft docking and multiple receptor conformations in virtual screening. *Journal of Medicinal Chemistry*, 47(21):5076, 2004.

[45] R. A. Friesner, J. L. Banks, R. B. Murphy, T. A. Halgren, J. J. Klicic, D. T. Mainz, M. P. Repasky, E. H. Knoll, M. Shelley, J. K. Perry, et al. Glide: a new approach for rapid, accurate docking and scoring. 1. Method and assessment of docking accuracy. *Journal of Medicinal Chemistry*, 47(7):1739–1749, 2004.

[46] R. A. Friesner, R. B. Murphy, M. P. Repasky, L. L. Frye, J. R. Greenwood, T. A. Halgren, P. C. Sanschagrin, and D. T. Mainz. Extra precision glide: Docking and scoring incorporating a model of hydrophobic enclosure for protein-ligand complexes. *Journal of Medicinal Chemistry*, 49(21):6177–6196, 2006.

[47] T. M. Frimurer, G. H. Peters, L. F. Iversen, H. S. Andersen, N. P. H. Møller, and O. H. Olsen. Ligand-induced conformational changes: improved predictions of ligand binding conformations and affinities. *Biophysical Journal*, 84(4):2273–2281, 2003.

[48] M.J. Frisch, G.W. Trucks, H.B. Schlegel, G.E. Scuseria, M.A. Robb, J.R. Cheeseman, J.A. Montgomery Jr, T. Vreven, K.N. Kudin, J.C. Burant, et al. Gaussian 03, Revision A. 1, Gaussian Inc., Pittsburgh, PA. 2003.

[49] C.R. Gardner, C.T. Walsh, and Ö. Almarsson. Drugs as materials: valuing physical form in drug discovery. *Nature Reviews Drug Discovery*, 3(11):926–934, 2004.

[50] A. Gavezzotti. Generation of possible crystal structures from molecular structure for low-polarity organic compounds. *J. Am. Chem. Soc*, 113(12):4622–4629, 1991.

[51] A. Gavezzotti. Calculation of intermolecular interaction energies by direct numerical integration over electron densities. I. Electrostatic and polarization energies in molecular crystals. *Journal of Physical Chemistry B*, 106(16):4145–4154, 2002.

[52] A. Gavezzotti. Calculation of intermolecular interaction energies by direct numerical integration over electron densities. 2. An improved polarization model and the evaluation of dispersion and repulsion energies. *Journal of Physical Chemistry B*, 107(10):2344–2353, 2003.

[53] A. Gavezzotti. Towards a realistic model for the quantitative evaluation of intermolecular potentials and for the rationalization of organic crystal structures. Part I. Philosophy. *CrystEngComm*, 5(76):429–438, 2003.

[54] R. J. Gdanitz. Ab Initio prediction of possible molecular crystal structures. *Theoretical Aspects and Computer Modeling of the Molecular Solid State*, pages 185–201, 1997.

[55] V. J. Gillet, W. Newell, P. Mata, G. Myatt, S. Sike, Z. Zsoldos, and A. P. Johnson. SPROUT: Recent developments in the de novo design of molecules. *Journal of Chemical Information and Computer Sciences*, 34(1):207–217, 1994.

[56] D. S. Goodsell and A. J. Olson. Automated docking substrates proteins simulated annealing. *Proteins: Structure, Function, and Genetics*, 8:195–202, 1990.

[57] R.W. Grosse-Kunstleve, N.K. Sauter, and P.D. Adams. Numerically stable algorithms for the computation of reduced unit cells. *Acta Crystallographica Section A*, 60(1):1–6, 2004.

[58] B. Gruber. The relationship between reduced cells in a general Bravais lattice. *Crystal Physics, Diffraction, Theoretical and General Crystallography*, 29(4):7394, 1973.

[59] B. A. Grzybowski, A. V. Ishchenko, R. S. DeWitte, G. M. Whitesides, and E. I. Shakhnovich. Development of a knowledge-based potential for crystals of small organic molecules: calculation of energy surfaces for C=0..H-N hydrogen bonds. *Journal of Physical Chemistry B*, 104(31):7293–7298, 2000.

[60] T. A. Halgren, R. B. Murphy, R. A. Friesner, H. S. Beard, L. L. Frye, W. T. Pollard, and J. L. Banks. Glide: a new approach for rapid, accurate docking and scoring. 2. Enrichment factors in database screening. *Journal of Medicinal Chemistry*, 47(7):1750–1759, 2004.

[61] I. Halperin, B. Ma, H. Wolfson, and R. Nussinov. Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins: Structure, Function, and Genetics*, 47:409–443, 2002.

[62] B. C. Hancock and M. Parks. What is the true solubility advantage for amorphous pharmaceuticals? *Pharmaceutical Research*, 17(4):397–404, 2000.

[63] F. C. Hawthorne. Crystals from first principles. *Nature*, 345(6273):297–297, 1990.

[64] D. W. M. Hofmann and J. Apostolakis. Crystal structure prediction by data mining. *Journal of Molecular Structure*, 647(1-3):17–39, 2003.

[65] J. R. Holden, Z. Du, and H. L. Ammon. Prediction of possible crystal structures for C-, H-, N-, O-, and F-containing organic compounds. *Journal of Computational Chemistry*, 14(4):422–437, 1993.

[66] L. F. Huang and W. Q. Tong. Impact of solid state properties on developability assessment of drug candidates. *Advanced Drug Delivery Reviews*, 56(3):321–334, 2004.

[67] S. Y. Huang and X. Zou. Ensemble docking of multiple protein structures: considering protein structural variations in molecular docking. *Proteins: Structure, Function, and Bioinformatics*, 66(2), 2007.

[68] G. Jones, P. Willett, R. C. Glen, A. R. Leach, and R. Taylor. Development and validation of a genetic algorithm for flexible docking. *Journal of Molecular Biology*, 267(3):727–748, 1997.

[69] P. G. Karamertzanis and C. C. Pantelides. Ab initio crystal structure prediction – I. Rigid molecules. *Journal of Computational Chemistry*, 26(3):304–324, 2005.

[70] P. G. Karamertzanis and C. C. Pantelides. Ab initio crystal structure prediction. II. Flexible molecules. *Molecular Physics*, 105(2):273–292, 2007.

[71] P. G. Karamertzanis and S. L. Price. Energy minimization of crystal structures containing flexible molecules. *Journal of Chemical Theory and Computation*, 2(4):1184–1199, 2006.

[72] H. R. Karfunkel and R. J. Gdanitz. Ab Initio prediction of possible crystal structures for general organic molecules. *Journal of Computational Chemistry*, 13(10):1171–1183, 1992.

[73] D. B. Kitchen1, H. Decornez1, J. R. Furr1, and J. Bajorath. Docking and scoring in virtual screening for drug discovery: Methods and applications. *Nature Reviews Drug Discovery*, 3:935–949, 2004.

[74] P. C. A. Kolb and A. Caflisch. Automatic and efficient decomposition of two-dimensional structures of small molecules for fragment-based high-throughput docking. *Journal of Medicinal Chemistry*, 49(25):7384–7392, 2006.

[75] M. Kontoyianni, L. M. McClellan, and G. S. Sokol. Evaluation of docking performance: comparative data on docking algorithms. *Journal of Medicinal Chemistry*, 47(3):558–565, 2004.

[76] B. Kramer, M. Rarey, and T. Lengauer. Evaluation of the FlexX incremental construction algorithm for protein-ligand docking. *Proteins: Structure, Function, and Genetics*, 37(2):228–241, 1999.

[77] I. Krivy and B. Gruber. A unified algorithm for determining the reduced (Niggli) cell. *Acta Crystallographica Section A*, 32(2):297–298, 1976.

[78] I. Kuntz, J. Blaney, S. Oatley, R. Langridge, and T. Ferrin. A geometric approach to macromolecule-ligand interactions. *Journal of Molecular Biology*, 161:269–288, 1982.

[79] A. R. Leach, B. K. Shoichet, and C. E. Peishoff. Prediction of protein-ligand interactions. Docking and scoring: successes and gaps. *Journal of Medicinal Chemistry*, 49(20):5851–5855, 2006.

[80] J. E. Lennard-Jones. Cohesion. *Proceedings of the Physical Society*, 43(5):461–482, 1931.

[81] C. A. Lipinski, F. Lombardo, B. W. Dominy, and P. J. Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced Drug Delivery Reviews*, 46:3–26, 1997.

[82] M. Liu and S. Wang. MCDOCK: A monte carlo simulation approach to the molecular docking problem. *Journal of Computer-Aided Molecular Design*, 13(5):435–451, 1999.

[83] J. P. M. Lommerse, W. D. S. Motherwell, H. L. Ammon, J. D. Dunitz, A. Gavezzotti, D. W. M. Hofmann, F. J. J. Leusen, W. T. M. Mooij, S. L. Price, B. Schweizer, et al. A test of crystal structure prediction of small organic molecules. *Acta Crystallographica Section B*, 56(4):697–714, 2000.

[84] C. F. Macrae, P. R. Edgington, P. McCabe, E. Pidcock, G. P. Shields, R. Taylor, M. Towler, and J. Streek. Mercury: visualization and analysis of crystal structures. *Applied Crystallography*, 39:453–457, 2006.

[85] J. Maddox. Crystals from 1st principles. *Nature*, 335:201, 1988.

[86] N. Majeux, M. Scarsi, J. Apostolakis, C. Ehrhardt, and A. Caflisch. Exhaustive docking of molecular fragments with electrostatic solvation. *Proteins Structure Function and Genetics*, 37(1):88–105, 1999.

[87] S. L. Mayo, B. D. Olafson, and W. A. Goddard III. DREIDING: A generic force field for molecular simulations. *The Journal of Physical Chemistry*, 94(26):8897–8909, 1990.

[88] A. J. Misquitta, G. W. A. Welch, A. J. Stone, and S. L. Price. A first principles prediction of the crystal structure of C6Br2ClFH2. *Chemical Physics Letters*, 456(1-3):105–109, 2008.

[89] N. Moitessier, P. Englebienne, D. Lee, J. Lawandi, and C. R. Corbeil. Towards the development of universal, fast and highly accurate docking/scoring methods: a long way to go. *British Journal of Pharmacology*, 153(S1):S7–S26, 2008.

[90] N. Moitessier, E. Therrien, and S. Hanessian. A method for induced-fit docking, scoring, and ranking of flexible ligands. Application to peptidic and pseudopeptidic β-secretase (BACE 1) inhibitors. *Journal of Medicinal Chemistry*, 49(20):5885–5894, 2006.

[91] W. Mooij, B. van Eijck, S. Price, P. Verwer, and J. Kroon. Crystal structure predictions for acetic acid. *Journal of Computational Chemistry*, 19(4):459–474, 1998.

[92] G. M. Morris, D. S. Goodsell, R. S. Halliday, R. Huey, W. E. Hart, R. K. Belew, and A. J. Olson. Automated docking using a lamarckian genetic algorithm and an empirical binding free energy function. *Journal of Computational Chemistry*, 19(14):1639–1662, 1998.

[93] W. D. S. Motherwell, H. L. Ammon, J. D. Dunitz, A. Dzyabchenko, P. Erk, A. Gavezzotti, D. W. M. Hofmann, F. J. J. Leusen, J. P. M. Lommerse, W. T. M. Mooij, et al. Crystal structure prediction of small organic molecules: a second blind test. *Acta Crystallographica Section B*, 58(4):647–661, 2002.

[94] D. T. Moustakas, P. T. Lang, S. Pegg, E. Pettersen, I. D. Kuntz, N. Brooijmans, and R. C. Rizzo. Development and validation of a modular, extensible docking program: DOCK 5. *Journal of Computer-Aided Molecular Design*, 20(10):601–619, 2006.

[95] R. Najmanovich, J. Kuttner, V. Sobolev, and M. Edelman. Side-chain flexibility in proteins upon ligand binding. *Proteins: Structure, Function, and Genetics*, 39(3), 2000.

[96] M. A. Neumann. Tailor-made force fields for crystal-structure prediction. *Journal of Physical Chemistry B*, 112(32):9810–9829, 2008.

[97] M. A. Neumann and M. A. Perrin. Energy ranking of molecular crystals using density functional theory calculations and an empirical van der Waals correction. *Journal of Physical Chemistry B*, 109(32):15531–15541, 2005.

[98] F. Osterberg, G. M. Morris, M. F. Sanner, A. J. Olson, and D. S. Goodsell. Automated docking to multiple target structures: incorporation of protein mobility and structural water heterogeneity in AutoDock. *Proteins: Structure, Function, and Genetics*, 46(1), 2002.

[99] J. Pillardy, Y. A. Arnautova, C. Czaplewski, K. D. Gibson, and H. A. Scheraga. Conformation-family Monte Carlo: A new method for crystal structure prediction. *Proceedings of the National Academy of Sciences*, 98(22):12351–12356, 2001.

[100] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C++, the art of scientific computing.* Cambridge University Press, 2002.

[101] S. L. Price. The computational prediction of pharmaceutical crystal structures and polymorphism. *Advanced drug delivery reviews*, 56(3):301–319, 2004.

[102] S. L. Price. From crystal structure prediction to polymorph prediction: interpreting the crystal energy landscape. *Physical Chemistry Chemical Physics*, 10(15):1996–2009, 2008.

[103] M. Pudipeddi and A. T. M. Serajuddin. Trends in solubility of polymorphs. *Journal of Pharmaceutical Sciences*, 94(5), 2005.

[104] F. A. Quiocho and N. K. Vyas. Novel stereospecificity of the l-arabinose-binding protein. *Nature*, 310:381–386, 1984.

[105] M. Rarey, B. Kramer, and T. Lengauer. Docking of hydrophobic ligands with interaction-based matching algorithms. *Bioinformatics*, 15:243–250, 1999.

[106] M. Rarey, B. Kramer, T. Lengauer, and G. A. Klebe. A fast flexible docking method using incremental construction algorithm. *Journal of Molecular Biology*, 261:470–489, 1996.

[107] D. Reid, B. S. Sadjad, Z. Zsoldos, and A. Simon. LASSO – ligand activity by surface similarity order: a new tool for ligand based virtual screening. *Journal of Computer-Aided Molecular Design*, 22(6):479–487, 2008.

[108] R. S. Rowland and R. Taylor. Intermolecular nonbonded contact distances in organic crystal structures: comparison with distances expected from van der Waals radii. *Journal of Physical Chemistry*, 100(18):7384–7391, 1996.

[109] B. S. Sadjad and Z. Zsoldos. ecrysp: A robust search method for crystal structure prediction of drug-like molecules. preprint.

[110] B. S. Sadjad and Z. Zsoldos. Toward a Robust Search for the Protein-Drug Docking Problem. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, to appear.

[111] G. Schneider and U. Fechner. Computer-based de novo design of drug-like molecules. *Nature Reviews Drug Discovery*, 4(8):649–663, 2005.

[112] T. S. Schroeter, A. Schwaighofer, S. Mika, A. Ter Laak, D. Suelzle, U. Ganzer, N. Heinrich, and K. R. Müller. Estimating the domain of applicability for machine learning QSAR models: a study on aqueous solubility of drug discovery molecules. *Journal of Computer-Aided Molecular Design*, 21(12):651–664, 2007.

[113] A. Schwaighofer, T. Schroeter, S. Mika, J. Laub, A. Ter Laak, D. Sulzle, U. Ganzer, N. Heinrich, and K.R. Muller. Accurate solubility prediction with error bars for electrolytes: A machine learning approach. *Journal of Chemical Information and Modeling*, 47(2):407–424, 2007.

[114] W. Sherman, T. Day, M. P. Jacobson, R. A. Friesner, and R. Farid. Novel procedure for modeling ligand/receptor induced fit effects. *Journal of Medicinal Chemistry*, 49(2):534–553, 2006.

[115] B. K. Shoichet, I. D. Kuntz, and D. L. Bodian. Molecular docking using shape descriptors. *Journal of Computational Chemistry*, 13(3):380–397, 1992.

[116] I. M. Sobol. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86–112, 1967.

[117] C. A. Sotriffer and I. Dramburg. In situ cross-docking to simultaneously address multiple targets. *Journal of Medicinal Chemistry*, 48(9):3122–3125, 2005.

[118] S. F. Sousa, P. A. Fernandes, and M. J. Ramos. Protein-ligand docking: current status and future challenges. *Proteins: Structure, Function, and Bioinformatics*, 65(1), 2006.

[119] G. P. Stahly. Diversity in single-and multiple-component crystals. The search for and prevalence of polymorphs and cocrystals. *Crystal Growth & Design*, 7(6):1007–1026, 2007.

[120] B. Suchod, J. Lajzerowicz, and A. Collet. (-)-2-Azabicyclo [2.2. 1] hept-5-en-3-one (Lactam). *Crystal Structure Communications*, 53(12):2701, 1997.

[121] R. D. Taylor, P. J. Jewsbury, and J. W. Essex. A review of protein-small molecule docking methods. *Journal of Computer-Aided Molecular Design*, 16(3):151–166, 2002.

[122] S. J. Teague. Implications of protein flexibility for drug discovery. *Nature Reviews Drug Discovery*, 2(7):527–541, 2003.

[123] M. Totrov and R. Abagyan. Flexible protein-ligand docking by global energy optimization in internal coordinates. *Proteins: Structure, Function, and Genetics*, 29:215–220, 1997.

[124] M. Totrov and R. Abagyan. Flexible ligand docking to multiple receptor conformations: a practical alternative. *Current Opinion in Structural Biology*, 18(2):178–184, 2008.

[125] B. P. Van Eijck. Crystal structure predictions using five space groups with two independent molecules. The case of small organic acids. *Journal of Computational Chemistry*, 23(4):456–462, 2002.

[126] M. L. Verdonk, G. Chessari, J. C. Cole, M. J. Hartshorn, C. W. Murray, J. W. M. Nissink, R. D. Taylor, and R. Taylor. Modeling Water Molecules in Protein- Ligand Docking Using GOLD. *Journal of Medicinal Chemistry*, 48(20):6504–6515, 2005.

[127] M. L. Verdonk, J. C. Cole, M. J. Hartshorn, C. W. Murray, and R. D. Taylor. Improved protein-ligand docking using GOLD. *Proteins-Structure Function and Genetics*, 52(4):609–623, 2003.

[128] P. Verwer and F. J. J. Leusen. Computer simulation to predict possible crystal polymorphs. *Reviews in Computational Chemistry*, 12:327–365, 1998.

[129] P. Vishweshwar, J. A. McMahon, J. A. Bis, and M. J. Zaworotko. Pharmaceutical co-crystals. *Journal of Pharmaceutical Sciences*, 95(3), 2006.

[130] J. Wang, T. Hou, and X. Xu. Aqueous Solubility Prediction Based on Weighted Atom Type Counts and Solvent Accessible Surface Areas. *Journal of Chemical Information and Modeling*, 49(3):571–581, 2009.

[131] J. Wang, G. Krudy, T. Hou, W. Zhang, G. Holland, X. Xu, et al. Development of reliable aqueous solubility models and their application in druglike analysis. *Journal of Chemical Information and Modeling*, 47(4):1395–1404, 2007.

[132] G. L. Warren, C. W. Andrews, A. M. Capelli, B. Clarke, J. LaLonde, M. H. Lambert, M. Lindvall, N. Nevins, S. F. Semus, S. Senger, et al. A critical assessment of docking programs and scoring functions. *Journal of Medicinal Chemistry*, 49(20):5912–5931, 2006.

[133] M. Weisel, E. Proschak, and G. Schneider. PocketPicker: analysis of ligand binding-sites with shape descriptors. *Chemistry Central Journal*, 1(1):7, 2007.

[134] D. E. Williams. Improved intermolecular force field for crystalline hydrocarbons containing four-or three-coordinated carbon. *Journal of Molecular Structure*, 485:321–347, 1999.

[135] D. E. Williams. Improved intermolecular force field for crystalline oxohydrocarbons including O-H...O hydrogen bonding. *Journal of Computational Chemistry*, 22(1):1–20, 2001.

[136] D. E. Williams. Improved intermolecular force field for molecules containing H, C, N, and O atoms, with application to nucleoside and peptide crystals. *Journal of Computational Chemistry*, 22(11):1154–1166, 2001.

[137] D. J. Willock, S. L. Price, M. Leslie, and C. R. A. Catlow. The relaxation of molecular crystal structures using a distributed multipole electrostatic model. *Journal of Computational Chemistry*, 16(5):628–647, 1995.

[138] A. Yan, J. Gasteiger, M. Krug, and S. Anzali. Linear and nonlinear functions on modeling of aqueous solubility of organic compounds by two structure representation methods. *Journal of Computer-Aided Molecular Design*, 18(2):75–87, 2004.

[139] A. C. Yao. On constructing minimum spanning trees in k-dimensional space and related problems. *SIAM Journal on Computing*, 11:721–736, 1982.

[140] Z. Zsoldos, D. Reid, A. Simon, B. S. Sadjad, and A. P. Johnson. eHiTS: An innovative approach to the docking and scoring function problems. *Current Protein and Peptide Science*, 7(5):421–435, 2006.

[141] Z. Zsoldos, D. Reid, A. Simon, B. S. Sadjad, and A. P. Johnson. eHiTS: A new fast, exhaustive flexible ligand docking system. *Journal of Molecular Graphics and Modelling*, 26(1):198–212, 2007.