Chapter 5

Pairwise Surface Alignment

In this chapter we describe in detail how pairwise alignments of molecules, based on the surface representation presented in Chapter 4, are computed. The aim is to compute good local alignments as well as good "global" alignments, if such global alignments exist. The pairwise alignments should have properties that allow the computation of feasible multiple alignments. Since we do not know in advance which pairwise alignments a good multiple alignment will be based on, we need to compute diverse pairwise alignments. That is to say, it is not sufficient to compute the optimal pairwise alignment only, but we also need subordinate alignments. Even though the computation of pairwise alignments is a useful method in its own right, the alignment algorithm presented in this chapter has been designed to allow an easy and meaningful computation of multiple matchings.

We have chosen a surface representation using points distributed on the molecular surface. Our aim is to find *one-to-one correspondences* between surface points of distinct molecules, such that the transformations computed according to these one-to-one correspondences yield good alignments of the molecular surfaces. In general, the one-to-one correspondences will be partial, but global or "near-global" correspondences can be established as well. Informally, we consider an alignment to be good, if the number of points brought close to each other by the alignment is large. In the case of surface points this means that large parts of the molecular surfaces of the two molecules get close to each other. We use a scoring function that balances the distance of the corresponding points and the number of corresponding points. This allows us to identify near-global correspondences with larger point distances as well as partial correspondences with smaller points distances. A more formal notion of a good alignment will be given in Section 5.1.3.

The problem of determining the optimal alignment of two point sets w.r.t. a given objective function is assumed to be NP-complete, which so far could not be proved. However, Kirchner [96] has shown that there exists a *fully polynomial time approximation scheme* (FPTAS) for this problem, which means that for each $\epsilon > 0$ a solution to the problem with an error at most ϵ can be found in polynomial time bounded in n and $1/\epsilon$, where n is the number of points. The run time of this approach [96], however, is a polynomial in n of degree 9, which is impractical for most real-world problems.

Several authors [6, 83, 97] have therefore proposed heuristic algorithms using a two-

step approach for the pairwise alignment of molecular structures based on the coordinates of the molecules' atomic nuclei. For the alignment of molecular surfaces, we follow this two-step approach, whereby we base the alignment on surface points. The two-step approach works as follows. First, a set of initial transformations is generated, which are used to overlay the molecular structures. Second, for each initial transformation a local optimization is carried out to improve the initial alignment. In each such optimization step, a one-to-one correspondence between points representing the two molecular surfaces is computed. This one-to-one correspondence is used to compute a new transformation. Given a one-to-one correspondence, the transformation minimizing the *root mean square* (rms) distance between corresponding points can be found using least-squares fitting [89]. For this transformation a new one-to-one correspondence is computed which again is improved using least-squares fitting, and so forth until no improvement is gained.

Since the applied optimization scheme is a local one, the success of the described two-step approach greatly depends on the generation of good initial transformations. Thus, one important question to be answered is how to generate initial transformations such that all or at least most good alignments can be found by the subsequent optimization procedure. This problem will be dealt with in Section 5.2. Even though the generation of initial transformations is crucial, the optimization step is equally important. The quality and efficiency of the approach depends in large parts on the choice of the optimization scheme. We use the simple, yet efficient, greedy optimization scheme proposed by Kirchner [97]. This optimization scheme together with our extensions to surface point matching will be considered in depth in Section 5.3. With Section 5.4, in which we describe how to filter out similar pairwise alignments, we conclude the algorithmic part of this chapter. In order to evaluate our surface alignment approach, we use two sets of molecules, which are introduced in Section 5.5. Results for these sets of molecules are then presented in Section 5.6, and the chapter is concluded in Section 5.7 with a summary and conclusion about the proposed surface alignment approach. We begin this chapter by introducing some basic notations and concepts in Section 5.1.

5.1 Basic Notations

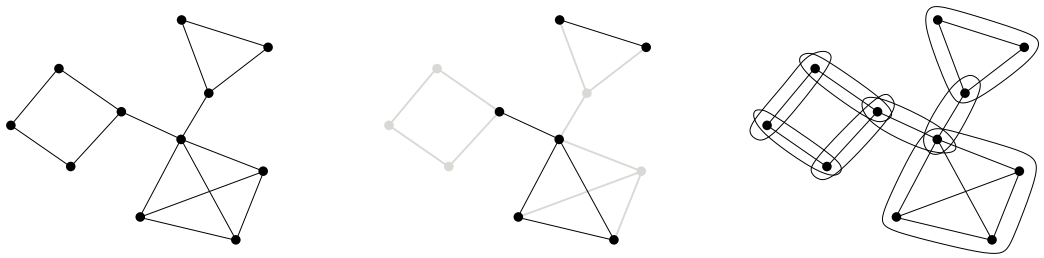
For the generation of initial transformations, we use clique detection on the distance compatibility graph of two point sets. Thus, for the reader not familiar with graph theory, we will define the notion of a clique and some related concepts in Section 5.1.1.

Section 5.2 deals with the generation of initial transformations. We propose two different ways, one of which is based on surface points. This method uses local surface shape features which are introduced in Section 5.1.2.

In Section 5.1.3, we will then give some basic concepts and notations about matchings and transformations that will play a major role in the optimization step.

5.1.1 Graphs

A general introduction to graph theory can be found, e.g., in [48]. Here, we only present some concepts that are needed to define a clique of a graph.



(a) (b) (c)

Figure 5.1: (a) Some arbitrary graph G. (b) The highlighted subgraph represents the subgraph of G induced by the black vertices. (c) All cliques of G. There exists one clique of size 4, one clique of size 3, and 6 cliques of size 2.

Definition 5.1.1 (Induced Subgraph)**.** Let $G = (V, E)$ be a graph with vertex set V and edge set $E \subseteq V \times V$. Let $V' \subseteq V$ be a subset of V. Then, the subgraph of G *induced* by V' is denoted by $G(V') = (V', E(V'))$, where $E(V')$ is defined as follows:

$$E(V') := \{(u, v) \in E \mid u, v \in V'\} .$$

An example of an induced subgraph is shown in Figure 5.1(b).

Definition 5.1.2 (Complete Graph)**.** Let $G = (V, E)$ be a graph. G is called *complete* if

$$(u, v) \in E \ , \ \forall u, v \in V, \ u \neq v \ .$$

Definition 5.1.3 (Clique)**.** Let $G = (V, E)$ be a graph. Then the vertex set $C \subseteq V$ is called a *clique*, if its induced subgraph, $G(C)$, is complete and maximal, i.e., there does not exist a vertex set $\widetilde{C} \subset V$, $C \subset \widetilde{C}$, such that $G(\widetilde{C})$ is complete. Figure 5.1(c) depicts all cliques of the graph displayed in Figure 5.1(a).

5.1.2 Local Surface Shape Features

The notion of local shape at some point on a 2-manifold surface can be captured using the two constructs *principal curvature* and *principal direction* from classical differential geometry. We will shortly describe these constructs in an informal way following the article of Goldman and Wipke [62]. A more mathematical description can be found, e.g., in [84].

The curvature κ is measured at a fiducial point on a planar curve. It describes how the normal to the curve changes when moving along the curve. The curvature is negative for clockwise and positive for counter-clockwise rotations. A large absolute value of κ indicates a large curvature. If κ is 0, then the curve at the fiducial point is a straight line. A more formal description follows.

Let $\mathcal{S}$ be a surface patch embedded in $\mathbb{R}^3$. Let p be a point on $\mathcal{S}$ with coordinates $\mathbf{x}(p)$ and let $\mathbf{n}(p)$ be the normal vector to $\mathcal{S}$ at $\mathbf{x}(p)$. Let further be t the *tangent plane* to $\mathcal{S}$ at $\mathbf{x}(p)$. Then each plane orthogonal to t and containing $\mathbf{n}(p)$ is called a *normal*

plane at $\mathbf{x}(p)$. Each normal plane intersects $\mathcal{S}$ in a plane curve called *normal section*. The curvature κ of the normal section at $\mathbf{x}(p)$ is called the curvature at $\mathbf{x}(p)$. If we start with an arbitrary normal plane at $\mathbf{x}(p)$ and rotate it around the normal vector up to an angle of π, all curvatures at $\mathbf{x}(p)$ are generated. The minimum and maximum values of these curvatures are called the principal curvatures and are denoted by κ_{min} and κ_{max}, respectively. If κ_{min} and κ_{max}, and hence all normal curvatures at $\mathbf{x}(p)$, are equal, then p is called an *umbilic*. The principal directions are the vectors defined by the intersection of the tangent plane and the normal planes corresponding to the principal curvatures. If p is not an umbilic, then the principal directions are orthogonal to each other.

In classical differential geometry, principal curvature and direction are defined for an infinitesimal area around the fiducial point. In the context of molecular surfaces, however, one is interested in more "global" shape features. In order to overcome this limitation, Goldman and Wipke introduced *quadratic shape descriptors* (QSD) [62]. A similar approach is described by Zachmann et al. [176]. The quadratic shape descriptor for a point p on the surface is computed by least-squares fitting a paraboloid to a small patch around p. Since a paraboloid has a quadratic form, the shape descriptor is termed quadratic shape descriptor. Thus, instead of computing the principal curvatures and directions, *local range curvatures and directions* are computed, which are the curvatures and directions of the fitted paraboloid. The local range curvature can be considered as some kind of averaged curvature.

The coordinates of each point within a small surface patch around some point p can be described by the displacement from the tangent plane at $\mathbf{x}(p)$, which is orthogonal to $\mathbf{n}(p)$. Such a map is called *monge patch* [100] and has the following form:

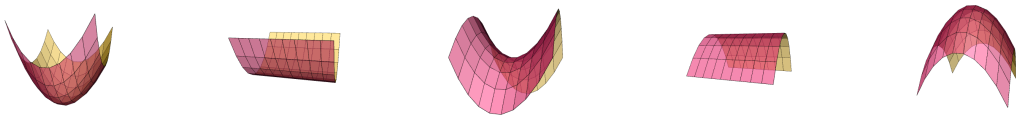
$$x(u,v) = u\vec{i} + v\vec{j} + w(u,v)\vec{k} \ ,$$

where $\vec{i}$, $\vec{j}$, and $\vec{k}$ are orthogonal unit vectors constructed such that $\vec{k}$ points in the direction of $\mathbf{n}(p)$, and u, v, and $w(u,v)$ are the transformed coordinates of the points on the surface patch. The function $w(\cdot,\cdot)$ is called the height function of u and v in the transformed coordinate system. Using the definition of the monge patch, we can express the local Hessian matrix II, which is the second fundamental form of the surface patch, as [100]

$$\mathrm{II} = \begin{pmatrix} \frac{\partial^2 w}{\partial u^2} & \frac{\partial^2 w}{\partial u \partial v} \\ \frac{\partial^2 w}{\partial u \partial v} & \frac{\partial^2 w}{\partial v^2} \end{pmatrix} \ .$$

The eigenvalues and eigenvectors of II give us the local range curvatures and directions, respectively [62, 100]. To distinguish the local range curvatures from the principal curvatures, we denote the minimum local range curvature by k_{min} and the maximum local range curvature by k_{max}.

According to the local range curvatures, the shape of a surface patch can be classified by the *shape index* which states the degree of convexity and concavity of the patch. It is



-1.0	-0.5	0.0	0.5	1.0
concave umbilic	rut spinode	minimal saddle	ridge spinode	convex umbilic

Figure 5.2: Five parabolic surface patches with their corresponding shape indices SI.

defined by

$$SI := \begin{cases} -\frac{2}{\pi} \arctan \frac{k_{max}+k_{min}}{k_{max}-k_{min}} & \text{, if } k_{max} \neq k_{min} \\ 1 & \text{, if } k_{max} = k_{min} < 0 \\ -1 & \text{, if } k_{max} = k_{min} > 0 \\ 0 & \text{, if } k_{max} = k_{min} = 0 \end{cases} \quad (cf.\ [100]). \tag{5.1}$$

The shape index maps the shape of some local patch to the interval $[-1, 1]$. Five parabolic surface patches with their corresponding shape indices are shown in Figure 5.2. Note, that the shape index does not measure the degree of curvature but the ratio between minimum and maximum curvature. In the context of solvent excluded surfaces, this is not problematic, since the degree of curvature is roughly equal across the whole solvent excluded surface. This is due to the fact, that the probe sphere and the van der Waals spheres have similar radii.

Based on the shape index, Goldman and Wipke [62] define a similarity measure as follows. Let SI_1 and SI_2 be the shape indices of two surface patches. Then, the shape similarity between these patches can be defined as

$$\text{shapeSim}(SI_1, SI_2) := \frac{2.0 - |SI_1 - SI_2|}{2.0} \ . \tag{5.2}$$

This shape similarity measure yields a value of 1.0, if the shape indices are identical, and a value of 0.0, if the shape indices are maximally dissimilar. For all other shape indices, the similarity value will be somewhere between 0.0 and 1.0.

5.1.3 Matchings

The pairwise alignment algorithm presented in this chapter is based on the computation of pairwise matchings. In this subsection we formally introduce the notion of a pairwise matching together with further basic notations related to the computation of pairwise alignments.

To align one molecular surface to another one, it needs to be transformed. So we begin this subsection with the definition of a rigid body transformation, i.e. a transformation that does not change the shape of any object, but only its position and orientation.

Definition 5.1.4 (Rigid Body Transformation [129])**.** A mapping $T : \mathbb{R}^3 \to \mathbb{R}^3$ is a *rigid body transformation* if it satisfies the following properties:

1. Length is preserved: $\|T(x) - T(y)\| = \|x - y\|$, for all points $x, y \in \mathbb{R}^3$, where $\|\cdot\|$ is the Euclidean norm.
2. The cross product of free vectors is preserved: $T_*(\vec{v} \times \vec{w}) = T_*(\vec{v}) \times T_*(\vec{w})$, for all vectors $\vec{v}, \vec{w} \in \mathbb{R}^3$, with $T_*(\vec{v}) := T(x) - T(y)$, where $x, y \in \mathbb{R}^3$ are arbitrary points satisfying $\vec{v} = x - y$.

We denote the set of all rigid body transformations by $\mathcal{T}$.

From Definition 5.1.4 follows that rigid body transformations preserve the inner vector product. This means, that orthogonal vectors are transformed to orthogonal vectors. Coupled with property 2 of Definition 5.1.4 it also follows that orthonormal vectors are transformed to orthonormal vectors and, hence, orthonormal coordinate frames to orthonormal coordinate frames [129].

Every rigid body transformation T can be written in the form

$$T : \mathbb{R}^3 \to \mathbb{R}^3,\ x \mapsto Rx + t,\ x \in \mathbb{R}^3\ ,$$

where $R \in \mathbb{R}^{3\times3}$ is an orthogonal matrix with $\det R = 1$, and $t \in \mathbb{R}^3$. We call R the *rotation* and t the *translational vector* (or simply translation) of transformation T.

Next, we want to give a precise definition of the informal term "one-to-one correspondence" used earlier in this chapter. This leads us to the definition of a *pairwise matching* between arbitrary finite sets, which do not necessarily have to be point sets. A pairwise matching can also be considered as a matching in a bipartite graph with bipartition $\{P, Q\}$ [48].

Definition 5.1.5 (Pairwise Matching)**.** A *pairwise matching* defined on two finite sets P and Q is a bijective function $f : \tilde{P} \to \tilde{Q}$, with $\tilde{P} \subseteq P$ and $\tilde{Q} \subseteq Q$. The set of all pairwise matchings that have a domain which is a subset of P and a range which is a subset of Q is denoted by $\mathcal{F}(P, Q)$. A pairwise matching f can be written as a set f^* of pairs from $P \times Q$ with $(p, q) \in f^* \Leftrightarrow f(p) = q$. The number of pairs $|f^*|$ will be referred to as the *size* of f. We further define

$$\begin{aligned} f^P :=& \{p \in P \mid \exists q \in Q : f(p) = q\}\ \text{, and} \\ f^Q :=& \{q \in Q \mid \exists p \in P : f(p) = q\}\ . \end{aligned}$$

In order to measure the quality of a matching, it is common to compute the *root mean square* (rms) distance of the matched points. This is defined as follows.

Definition 5.1.6 (Root Mean Square Distance)**.** Let $P = \{p_i\}_{i=1}^m$ and $Q = \{q_i\}_{i=1}^n$ be two finite point sets and let $\mathbf{X}(P) := \{\mathbf{x}(p_i)\}_{i=1}^m \subset \mathbb{R}^3$ and $\mathbf{X}(Q) := \{\mathbf{x}(q_i)\}_{i=1}^n \subset \mathbb{R}^3$ denote their respective coordinate sets. Then we define the *root mean square distance* of P and Q w.r.t. a matching $f \in \mathcal{F}(P, Q)$ as

$$\mathrm{rmsd}(P, Q; f) := \sqrt{\frac{\sum_{(p,q)\in f^*} \|\mathbf{x}(p) - \mathbf{x}(q)\|^2}{|f^*|}}.$$

During the iterative alignment procedure, the second point set is constantly transformed to overlay it with the first point set. Hence, we are more interested in the rms distance of the matched points w.r.t. a given transformation.

Definition 5.1.7 (Root Mean Square Distance (2))**.** Let $P, Q, \mathbf{X}(P)$, and $\mathbf{X}(Q)$ be as in Definition 5.1.6. Then we define the rms distance of P and Q w.r.t. a matching $f \in \mathcal{F}(P,Q)$ and a rigid body transformation T as

$$\mathrm{rmsd}(P,Q;f;T) := \sqrt{\frac{\sum_{(p,q)\in f^*} \|\mathbf{x}(p) - T(\mathbf{x}(q))\|^2}{|f^*|}}.$$

Given a one-to-one correspondence between points of two points sets P and Q, i.e. a pairwise matching, we are looking for the transformation that minimizes the rms distance.

Definition 5.1.8 (Matching Transformation)**.** Let P and Q be two point sets. For a matching $f \in \mathcal{F}(P,Q)$ we define its *matching transformation* to be the transformation $T_f \in \mathcal{T}$ minimizing $\mathrm{rmsd}(P,Q;f;T)$, i.e.,

$$T_f := \arg\min_{T\in\mathcal{T}} \mathrm{rmsd}(P,Q;f;T)$$

The matching transformation T_f can be computed in $\mathcal{O}(|f^*|)$ time using singular value decomposition of the matrix product $\widetilde{X}\widetilde{Y}^T$, $\widetilde{X}, \widetilde{Y} \in \mathbb{R}^{3\times|f^*|}$, where $\tilde{x}_{\cdot i} := \mathbf{x}(\tilde{p}_i)$, $\tilde{p}_i \in f^P$, $i \in \{1,\ldots,|f^*|\}$, and $\tilde{y}_{\cdot i} := \mathbf{x}(f(\tilde{p}_i))$, with $\tilde{x}_{\cdot i}$ and $\tilde{y}_{\cdot i}$ being the i'th column vectors of matrices $\widetilde{X}$ and $\widetilde{Y}$, respectively [89].

Definition 5.1.9 (Matching Score)**.** Let P, Q, $f \in \mathcal{F}(P,Q)$, and T be defined as in Definition 5.1.7, and let $\alpha \in \mathbb{R}^+$. We define the matching score of f w.r.t. T as

$$\mathrm{score}(P,Q;f;T) := \frac{|f^*|}{\min(|P|,|Q|)} \cdot e^{-\alpha\cdot\mathrm{rmsd}(P,Q;f;T)}. \tag{5.3}$$

The function *score*, which yields values between 0.0 and 1.0 including, is used to measure the quality of an alignment. A value of 0.0 states that the two point sets are maximally dissimilar under the current transformation. A score of 1.0 states that the best possible alignment has been found. The function score is not only applied to measure the quality of the final alignment but is also used as objective function in the local optimization procedure. The objective function consists of two terms which are antipodal. The first term measures the size of the matching. In order to obtain a high matching score, the size of a matching should be large. However, the goodness of a matching also depends on the distance of the matched points which is measured by the second term. This term is larger for small matchings which, in general, have a smaller rms distance than larger ones. Hence, a smaller matching might in fact yield a larger score value if its rms distance is considerably smaller than that of the larger one. Thus, the two terms weight the size of a matching against the distance of matched points. This weighting can be influenced by the parameter α. The function *score* with $\alpha = 1.0$ was previously used in [97, 163]. We have added the parameter α to be able to weight the importance of the rms distance. If not otherwise mentioned, we also use a value of 1.0.

If we fix the matching f we can determine the optimal matching score for f by determining the transformation T maximizing the *matching score* (cf. Equation (5.3)). This can be formulated as follows.

Definition 5.1.10 (Optimal Matching Score)**.** Let P and Q be two point sets. Then, for a matching $f \in \mathcal{F}(P,Q)$ we define its *optimal matching score* as

$$\text{score}(P,Q;f) := \max_{T \in \mathcal{T}} \text{score}(P,Q;f;T) \ .$$

It is easy to see that for each matching $f \in \mathcal{F}(P,Q)$ the following property holds:

$$\text{score}(P,Q;f) = \text{score}(P,Q;f;T_f) \ .$$

Similarly, if we fix the transformation T, we can ask for the optimal matching f_T, i.e., the matching which maximizes the *matching score* (Equation (5.3)). This can be formulated by the following definition.

Definition 5.1.11 (Optimal Matching)**.** Let P and Q be two point sets. Then, for a rigid transformation $T \in \mathcal{T}$ we define the *optimal matching* w.r.t. T as

$$f_T := \arg \max_{f \in \mathcal{F}(P,Q)} \text{score}(P,Q;f;T) \ .$$

For the generation of initial transformations, matchings satisfying pairwise distance constraints play an important role. These matchings are termed *d-bounded matchings* and are defined next.

Definition 5.1.12 (d-Bounded Matching)**.** Let P and Q be two point sets. Let $f \in \mathcal{F}(P,Q)$ be a matching and let $d > 0$, $d \in \mathbb{R}$, be a constant. We call f *d-bounded*, if f satisfies the following property:

$$\big| \|\mathbf{x}(p') - \mathbf{x}(p'')\| - \|\mathbf{x}(f(p')) - \mathbf{x}(f(p''))\| \big| \leq d \quad , \quad \forall p', p'' \in f^P \ .$$

5.2 Initial Transformations

In [6], Akutsu proposes three strategies for the generation of initial transformations for the alignment of two molecules. The first strategy uses exhaustive search w.r.t. all triples of atoms in both molecules. Here, for each pair of triples a transformation is computed by least-squares fitting the triples. Second, instead of using all triples, for one molecule a random subset of atoms of some predefined size is selected, and the triples of this subset are used only. The probability of finding the best alignment using this random strategy depends on the size of the selected subset. The third strategy uses small chain fragments of the proteins being considered. This last strategy is not applicable to the alignment of drug molecules, since these, in general, do not have a single chain.

Kirchner [97] also considers triples of atoms for initial transformation generation. However, he reduces the number of triples to be tested by considering only triples having similar

inter-atomic distances, i.e., he only considers pairs of atom triples representing d-bounded matchings of size 3, where d is some predefined constant.

However, even if the number of pairs of triples is restricted to d-bounded pairs of triples, this number can still be rather large depending on the size of the molecules under consideration. We therefore use d-bounded matchings of size c, $c \geq 3$, to generate initial transformations. Let P and Q be two point sets representing two molecules, respectively. Then, d-bounded matchings of size larger than 3 can be computed using clique detection on the *distance compatibility graph* of P and Q, which is defined as follows.

Definition 5.2.1 (Distance Compatibility Graph)**.** Let $d > 0$, $d \in \mathbb{R}$, be a constant and let P and Q be two point sets. Then, the *distance compatibility graph* of P and Q w.r.t. d is defined on the vertex set $P \times Q$ and has the following property: two vertices (p', q') and (p'', q'') are connected by an edge, if the matching f given by $f^* = \{(p', q'), (p'', q'')\}$ is d-bounded, i.e.,

$$\big|\|\mathbf{x}(p') - \mathbf{x}(p'')\| - \|\mathbf{x}(q') - \mathbf{x}(q'')\|\big| \leq d \ .$$

We denote the distance compatibility graph of P and Q w.r.t. d by $G(P, Q)_d = (P \times Q, E_d)$.

It is obvious, that each clique of the distance compatibility graph corresponds to a maximal d-bounded matching f of P and Q, where maximal means that f cannot be extended to a matching $\tilde{f}$, such that $f^* \subset \tilde{f}^*$ and $\tilde{f}$ is d-bounded. Thus, by identifying all cliques of the distance compatibility graph, we find all maximal d-bounded matchings. The matching transformations (cf. Definition 5.1.8) of these d-bounded matchings will be used as initial transformations for the computation of pairwise alignments. We can now easily reduce the number of initial transformations by increasing the minimum size c. The number of vertices in the graph can also be reduced, which might be accomplished by introducing further constraints. For example, we could require each two points representing a vertex in $G(P, Q)_d$ to have some common properties. For example, in the case of atoms, we could force the atoms to be of the same type; for surface points, other constraints can be used, such as similar shape properties, etc.

Unfortunately, determining all cliques of some arbitrary graph is NP-complete. Nevertheless, for sparse graphs of moderate size, such as distance compatibility graphs of drug-sized molecules, there exist efficient algorithms that solve this problem quickly. The most widely used algorithm applied to identifying cliques in the compatibility graph of molecules is the Bron-Kerbosch algorithm [30], which enumerates all cliques of the graph. It applies a branch-and-bound technique to prune the search tree and is very efficient in doing so.

Several algorithms have applied clique detection for molecular alignment (see, e.g., [64, 103, 29, 115, 127, 35, 81]). However, there is one major drawback when using clique detection directly and solely for molecular alignment. Clique detection on the distance compatibility graph identifies substructures satisfying *overall* distance constraints. While this allows identification of very similar substructures, it is difficult to utilize it for identifying less similar substructures. To do so, one would have to increase the value of d. Increasing the value of d, however, very quickly leads to an enormous increase in the number of d-bounded matchings.

Due to these observations, we only use clique detection for the generation of initial transformations. When using clique detection on the distance compatibility graph to generate initial transformations, we have a choice between different representations of the molecules. In the following two subsections we want to look at the merits and demerits of two such representations, i.e. the molecule's atoms, and points generated on the molecular surface.

5.2.1 Atom-Based Initial Transformations

The atomic structure of a molecule can be considered as the *skeleton* of the volume enclosed by the solvent excluded surface (SES) (cf. Section 2.3.2). Most points on the SES lie on the van der Waals sphere (cf. Section 2.2.3) of some atom, hence, their distance from an atomic nucleus is equal to the atom's van der Waals radius. However, there exist points in concave regions that are not located on the van der Waals sphere of any atom. These points lie on the surface of a probe sphere, which is in contact with the van der Waals spheres of at least two atoms (cf. Figure 2.4). Since the radius of the probe sphere is comparable to the van der Waals radii of the molecule's atoms, the distance of all points located on the SES to the closest atomic nucleus is less than two times the radius of the closest van der Waals sphere. Hence, we can argue, that for each point on the SES there exists an atom "close-by". Due to this fact we can also reason that a good alignment of molecular surfaces, i.e. an alignment positioning many points on the molecular surfaces close to each other, will also position the atomic nuclei of the nearest atoms close to each other. This is the rationale for using the coordinates of the atomic nuclei for the generation of initial transformations.

This approach has advantages and disadvantages. A clear advantage is that the number of atoms in a molecule is much smaller than the number of surface points with distances similar to those of the bonded atoms. Since clique detection is NP-complete, a reduction of the number of points and hence the size of the graph has great influence on the applicability of the method. One disadvantage of this method can be seen from the following observation. A good surface alignment will in most cases result in a good volume overlap, whereas a good volume, i.e. atom, overlap might result in a very poor surface alignment. For example, consider an initial transformation due to the overlap of two terminating rings. If the rest of the molecules point in different directions, there will be hardly any surface overlap. Hence, using the coordinates of the atomic nuclei for initial transformation generation might result in some very poor initial alignments.

In Section 5.6 we compare the results of atom-based initial transformation generation and surface-based initial transformation generation. The latter of which will be shortly described next.

5.2.2 Surface-Based Initial Transformations

Instead of using the coordinates of atomic nuclei for generating initial transformations, we can also use surface points. Cosgrove et al. [35] and Hofbauer [81] use surface points in combination with clique detection to compute surface alignments (cf. Section 3.3.4).

However, they do not use clique detection as initial step, as we do, but to generate final alignments. Furthermore, they do not use homogeneously distributed points, but identify surface regions with similar properties, e.g. shape or physico-chemical properties, and represent each such region by a single point. In contrast, we use homogeneously distributed surface points generated by applying Algorithm 4.7 (see Chapter 4). Varying the point density allows us to determine the size of the point sets. Larger point sets will lead to a larger number of d-bounded matchings for the same values of c and d. Thus, the point density constitutes one possibility of controlling the number of initial transformations. Note, however, that we have to ensure to use the same point density for all molecular surfaces to be aligned.

Using surface points, the number of vertices in the distance compatibility graph can be reduced by requiring that matching points need to satisfy some shape similarity. Hofbauer [81] uses harmonic shape image filters [178] to compare the shapes of small patches. Cosgrove et al. [35] directly use shape indices as defined in Equation (5.1). We use the shape similarity measure proposed by Goldman and Wipke [62], defined in Equation (5.2). Furthermore, we consider the surface normals at the surface points. Thus, in addition to requiring the matching points to satisfy the distance constraint given by the value of d, we also require the angular distance of the normals of the matched points to satisfy some constraint given by the angular distance threshold β. This leads to the following redefinition of the distance compatibility graph for surface points.

Definition 5.2.2 (Distance Compatibility Graph for Surface Points)**.** Let $d \in \mathbb{R}^+$, and $\beta \in [0, \pi]$ be constants and let P and Q be two surface point sets. Then, the *distance compatibility graph* of P and Q w.r.t. d and β is defined on the vertex set $P \times Q$ and has the following property: two vertices (p', q') and (p'', q'') are connected by an edge, if the matching f given by $f^* = \{(p', q'), (p'', q'')\}$ is d-bounded, i.e.,

$$\big|\|\mathbf{x}(p') - \mathbf{x}(p'')\| - \|\mathbf{x}(q') - \mathbf{x}(q'')\|\big| \leq d \ ,$$

and satisfies the angular distance constraint, i.e.,

$$\big|\measuredangle\big(\mathbf{n}(p'), \mathbf{n}(p'')\big) - \measuredangle\big(\mathbf{n}(q'), \mathbf{n}(q'')\big)\big| \leq \beta \ ,$$

where $\measuredangle(\vec{u}, \vec{v})$ denotes the angle between the vectors $\vec{u}$ and $\vec{v}$. We denote the distance compatibility graph of P and Q w.r.t. d and β by $G(P, Q)_{d,\beta} = (P \times Q, E_{d,\beta})$.

5.3 Alignment Optimization

5.3.1 Point Matching

In this section we consider the problem of determining a locally optimal or near-optimal pairwise matching of two finite points sets (cf. Definition 5.1.5) for a given initial transformation w.r.t. the objective function *score*, i.e. the matching score defined in Equation (5.3). One solution to this problem, as proposed by several authors [6, 83, 97], is an iterative procedure in which alternately the matching and the transformation corresponding to the

current optimal matching (cf. Definition 5.1.11) are improved. Fortunately, the problem of determining the optimal transformation for a given matching can be efficiently solved using singular value decomposition of a matrix composed from the coordinates of the matching points [89] (cf. Section 5.1.3). Thus, the problem of determining a locally optimal pairwise matching reduces to determining the optimal matching for a given transformation.

In [97], Kirchner presented an exact algorithm for the computation of the optimal matching for a given transformation T. This algorithm is based on the identification of augmented paths with minimal weight in bipartite graphs with edge weights. He gives an incremental procedure for the computation of the optimal matching with size $k+1$ from the optimal matching with size k. This makes the algorithm very efficient. Nevertheless, the run time for this exact algorithm is still $\mathcal{O}(n^3)$, where n is the maximal number of points in the point sets to be matched. The algorithm computes for each possible value of k the matching f, $|f^*| = k$, with minimal rms distance (cf. Definition 5.1.7), and among these matchings selects the one with largest score as optimal matching.

Kirchner proposes two strategies to improve this run time. First, he introduces a maximal distance δ which two points are allowed to be apart from each other to be matched. This improves the run time drastically, but is still too slow to be applied to a large number of initial transformations. An additional, second, strategy proposed by him is therefore to use a greedy strategy, which first computes the optimal matching of size 1. From this, a matching of size 2 is computed greedily by adding that pair of points closest to each other such that neither of the two points is in the current matching yet. This step is repeated until no matching of greater size can be found. In detail, the procedure is given in Algorithm 5.1.

5.3.2 Surface Point Matching

For the computation of pairwise matchings of surface point sets we could directly apply Algorithm 5.1. However, for the alignment of surfaces it is not sensible to allow the matching of all points located close to each other. For example, if two surface points, $p \in P$ and $q \in Q$, are close to each other but their normal vectors point in opposite directions, matching points p and q will not lead to an alignment with high similarity in the surface regions corresponding to p and q. Matching p and q might have two undesired effects. First, the score for this alignment might be larger than it actually should be due to the additional matching pair which does not present a local surface similarity. Second, a better alignment by the iterative optimization procedure might be prevented, because the computation of the transformation might be such that p and q will stay close to each other.

We therefore introduce an angular distance threshold γ, which we generally set to $60°$. In contrast to the distance threshold δ, which reduces the overall number of point pairs to be considered, γ is used to eliminate undesired matching pairs. For two surface points p and q to be matched, from now on we require the following additional constraint to be satisfied:

$$\measuredangle\big(\mathbf{n}(p), T_*(\mathbf{n}(q))\big) < \gamma \ , \tag{5.4}$$

where $T_*(\cdot)$ is defined as in Definition 5.1.4.

Algorithm 5.1 Greedy Point Matching

Input: finite point sets P and Q, rigid transformation T

Input: distance threshold δ

Output: near-optimal matching f_{opt}

$f_0^* \leftarrow \emptyset$
$P' \leftarrow P$, $Q' \leftarrow Q$
$k \leftarrow 0$
while $\exists p \in P', q \in Q' : \|\mathbf{x}(p) - \mathbf{x}(T(q))\| < \delta$ **do**
    $(p, q) \leftarrow \arg \min_{p \in P', q \in Q'} \|\mathbf{x}(p) - \mathbf{x}(T(q))\|$
    $f_{k+1}^* \leftarrow f_k^* \cup \{(p, q)\}$
    $P' \leftarrow P' \setminus \{p\}$
    $Q' \leftarrow Q' \setminus \{q\}$
    $k \leftarrow k + 1$
end while
$f_{opt} \leftarrow \arg \max_k \text{score}(P, Q; f_k; T)$

5.3.3 Handling Multiple Properties

At this stage, we are able to compute surface point matchings with Algorithm 5.1, if we consider the additional constraint given by Equation (5.4). However, in Chapter 4 we described the generation of distinct point sets representing distinct properties of the molecular surface. Points representing distinct properties should not get matched. Let N be the number of properties we want to consider. Then, we denote the point sets representing property i by P_i and Q_i, respectively. These considerations lead us to the definition of *combined pairwise matchings* w.r.t. N distinct properties.

Definition 5.3.1 (Combined Pairwise Matching)**.** Let P and Q be two finite sets and let $\{P_i\}_{i=1}^N$ and $\{Q_i\}_{i=1}^N$ be decompositions of P and Q, respectively, i.e. $P = \bigcup_{i=1}^N P_i$ and $P_i \cap P_j = \emptyset$, $\forall i \neq j$, and equivalently for Q. Furthermore, let $f_i \in \mathcal{F}(P_i, Q_i)$, be pairwise matchings on P_i and Q_i. Then we define the *combined pairwise matching* $f \in \mathcal{F}(P, Q)$ w.r.t. the decompositions $\{P_i\}_{i=1}^N$ and $\{Q_i\}_{i=1}^N$ by

$$f(p) := f_i(p), \text{ if } p \in P_i \ .$$

It directly follows that

$$f^* = \bigcup_{i=1}^N f_i^* \ .$$

Analogously to the matching score of "simple" matchings (cf. Equation (5.3)), we can

define the matching score for a combined pairwise matching as

$$\text{score}^*(P,Q;f;T) := \frac{|f^*|}{\sum_{i=1}^{N}\min(|P_i|,|Q_i|)} \cdot e^{-\alpha\cdot\text{rmsd}(P,Q;f;T)} \ .$$

With this definition we can adjust Algorithm 5.1 to handle distinct property sets. The whole algorithm for the computation of optimal or near-optimal surface point matchings which considers distinct property sets P_i and Q_i is given in Algorithm 5.2.

Algorithm 5.2 Greedy Surface Point Matching Handling Multiple Properties

Input: finite surface point sets P_i and Q_i, $i = 1,\ldots,N$, rigid transformation T

Input: distance threshold δ, angular distance threshold γ

Output: near-optimal combined matching f_{opt}

1: $f_0^* \leftarrow \emptyset$

2: $\forall i = 1,\ldots,N : P_i' \leftarrow P_i,\ Q_i' \leftarrow Q_i$

3: $k \leftarrow 0$

4: **while** $\exists i \in \{1,\ldots,N\} \wedge \exists p \in P_i', q \in Q_i' :$

5: $\quad \|\mathbf{x}(p) - \mathbf{x}(T(q))\| < \delta \ \wedge \ \measuredangle\big(\mathbf{n}(p), T_*(\mathbf{n}(q))\big) < \gamma$ **do**

6: $\quad j \leftarrow \arg\min_{i=1,\ldots,N} \min_{p\in P_i', q\in Q_i'} \{\|\mathbf{x}(p) - \mathbf{x}(T(q))\| \mid \measuredangle\big(\mathbf{n}(p), T_*(\mathbf{n}(q))\big) < \gamma\}$

7: $\quad (p,q) \leftarrow \arg\min_{p\in P_j', q\in Q_j'} \{\|\mathbf{x}(p) - \mathbf{x}(T(q))\| \mid \measuredangle\big(\mathbf{n}(p), T_*(\mathbf{n}(q))\big) < \gamma\}$

8: $\quad f_{k+1}^* \leftarrow f_k^* \cup \{(p,q)\}$

9: $\quad P_j' \leftarrow P_j' \setminus \{p\}$

10: $\quad Q_j' \leftarrow Q_j' \setminus \{q\}$

11: $\quad k \leftarrow k+1$

12: **end while**

13: $f_{opt} \leftarrow \arg\max_k \text{score}^*(P,Q;f_k,T)$

5.3.4 Locating Close Points

Up to now we have not given any information on how to determine the closest pair of points to be added next to the current matching (cf. Algorithm 5.2, line 7). However, a fast determination of close points is crucial for the efficiency of the alignment algorithm. In this section, we therefore develop an efficient data structure for the fast determination of close points.

Let P and Q be two finite surface point sets with decompositions $\{P_i\}_{i=1}^N$ and $\{Q_i\}_{i=1}^N$, respectively. Then, one solution to the problem would be to compute the distances of all pairs of surface points belonging to the same property, remove all those pairs with distance larger than δ and sort the remaining pairs according to their distances. Then we could go through the sorted list, starting with the pair of points having the smallest

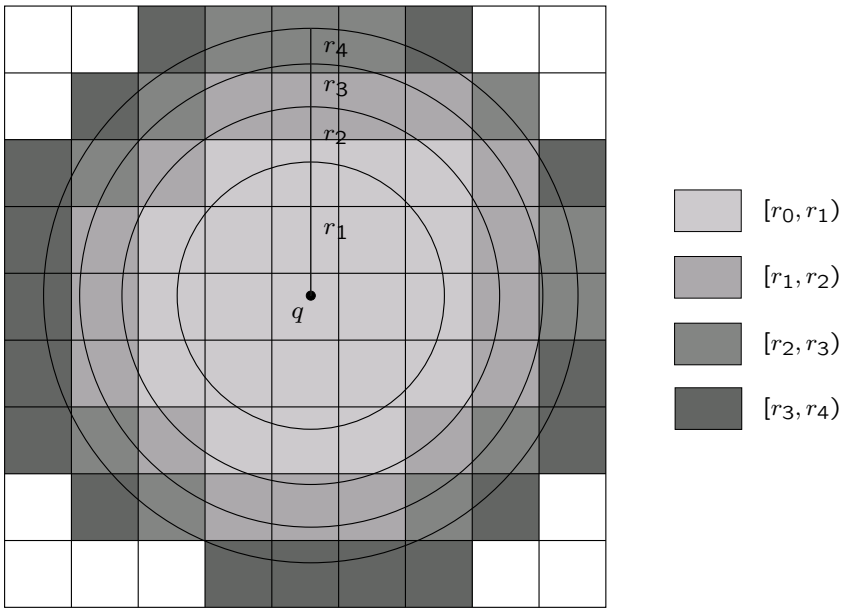


Figure 5.3: Point Distance Grid with variable distance ranges from some point q. The gray-scales of the grid cells denote the membership to the corresponding distance interval. Those grid cells left white are outside the distance threshold δ from q.

distance, and successively add the next pair if none of its two points is in the matching yet. This approach has a run time of $\mathcal{O}(\sum_i m_i n_i + s \ln s)$, where m_i and n_i are the sizes of the points sets P_i and Q_i, respectively, and s is the number of point pairs with distance smaller than δ.

Since we use greedy matching and only consider pairs of points with distance smaller than some δ, we can do a lot better. In order to do so, however, we need a data structure that is able to answer the following query quickly. Given some point $q \in Q_i$ with transformed coordinates $T(\mathbf{x}(q))$, which points of point set P_i are within some distance range $[r_i, r_{i+1})$ from $T(\mathbf{x}(q))$? If we had such a data structure, then for a set of distances $\{r_i\}_{i=0}^{k}$ with $0 = r_0 < r_1 < \ldots < r_k = \delta$, we could apply the following approach. For each property i and each point $q \in Q_i$, identify all points $p \in P_i$ within a distance range of $[r_0, r_1)$ from $T(\mathbf{x}(q))$. From the set of these point pairs, compute all matchings f_0 to f_{l^0} using Algorithm 5.2. Then, for each property i and each point $q \in Q_i, q \notin f_{l^0}^{Q}$, identify all points $p \in P_i, p \notin f_{l^0}^{P}$ within a distance range of $[r_1, r_2)$ from $T(\mathbf{x}(q))$. Using this set of pairs, enlarge the matching f_{l^0} greedily up to some matching f_{l^1}. Continue like this until distance threshold δ has been reached or until no larger matching can be found. From all matchings f_0 up to $f_{l^{k-1}}$ select the one with maximal score as f_{opt}.

In order to answer the above mentioned query quickly, we developed a data structure, which we call *point distance grid*. A detailed description of the point distance grid and its application follows.

Point Distance Grid

Let P be the target point set, i.e. the point set which is not transformed during the alignment procedure, and let Q be the query point set, i.e. the point set that is transformed by the rigid body transformation T. Let further denote by $\mathfrak{B}_\mathfrak{i}$ the bounding box of $\mathbf{X}(P_i)$ expanded in each direction by δ, and let $\mathfrak{G}_i$ denote the smallest uniform grid enclosing $\mathfrak{B}_\mathfrak{i}$, having grid cell size $\mathfrak{d}$ in each dimension. Let $\{r_i\}_{i=0}^k$ be a set of distances with $0 = r_0 < r_1 < \ldots < r_k = \delta$, which will be used as described above. For each grid cell c of $\mathfrak{G}_i$ we maintain k lists, such that the j'th list contains all points $p \in P_i$ with minimal distance to the bounding box of c, denoted by $d_{min}(\mathbf{x}(p), c)$, in $[r_j, r_{j+1})$. In order to build up these lists, for each property i and each point $p \in P_i$ we determine grid cell c containing $\mathbf{x}(p)$ and do a breadth first search on $\mathfrak{G}_i$ starting from c to determine all grid cells $\tilde{c}$ with minimal distance from $\mathbf{x}(p)$ to the bounding box of $\tilde{c}$ smaller than δ. Point p is added to the j'th list of cell $\tilde{c}$, if $d_{min}(\mathbf{x}(p), \tilde{c}) \in [r_j, r_{j+1})$. An example is shown in Figure 5.3. The grid $\mathfrak{G}_i$ together with the k neighbor lists maintained for each grid cell $c \in \mathfrak{G}_i$ is termed the point distance grid of point set P_i w.r.t. distance threshold δ, grid cell size $\mathfrak{d}$, and distances $\{r_i\}_{i=0}^k$. Since P_i is never transformed during the alignment procedure, the point distance grids needs to be built up only once.

Algorithmic Details

Using point distance grids, we can render Algorithm 5.2 more precisely, leading to Algorithm 5.3. The algorithm basically works as follows. Instead of looking at all pairs of points with distance smaller than δ at once, we look at the closest points first. Having computed the matchings for these points, we look at the next closest points. Now, we only need to consider those points that have not yet been matched. In lines 6 through 19 of Algorithm 5.3 we determine in each step the next closest points. Note, however, that some point pairs might already have been determined during one of the previous steps (cf. lines 13 and 14). This is due to the fact that we consider the minimum distance from some point $p \in P$ to the bounding box of some grid cell c, which might contain points with distances to p outside the distance range into which the minimum distance falls. In order not to loose those point pairs, they need to be added to the appropriate list, E_t. Note that there might also be points with larger distance than δ. These are discarded. The number of such points depends on the grid cell size $\mathfrak{d}$.

As yet, we have not given any details on how to choose the distances $\{r_i\}_{i=0}^k$. Setting the values of r_i such that the size of the intervals are equal, we get the undesired effect, that for larger i the number of points to be tested might increase, since the surface points are distributed on a two-dimensional surface and not on a one-dimensional line. Hence, it is more sensible to set the distances according to equation

$$r_i = \sqrt{i} \cdot \frac{\delta}{\sqrt{k}} \ .$$

Since we do not allow an arbitrary packing of the surface points, the number of points within a small neighborhood of some point will be bounded by some constant M. Thus, the theoretic run time of our algorithm is $\mathcal{O}(\sum_i M \cdot n_i + s \ln s)$, i.e., it is no longer quadratic.

Algorithm 5.3 Algorithm 5.2 Rewritten Using Point Distance Grids

Input: finite surface point sets P_i and Q_i, $i = 1, \ldots, N$, and transformation T
Input: angular distance threshold γ
Input: point distance grids $\mathfrak{G}_i$ and distances $\{r_i\}_{i=0}^{K}$
Output: near-optimal combined matching f_{opt}

$f_0^* \leftarrow \emptyset$
$\forall i = 1, \ldots, N : P_i' \leftarrow P_i,\ Q_i' \leftarrow Q_i$
$\forall i = 1, \ldots, K : E_i \leftarrow \emptyset$
$l \leftarrow 0$
for $i \in \{0, \ldots, K\}$ **do**
  for $j \in \{1, \ldots, N\}$ **do**
    for $q \in Q_j'$ **do**
      determine c in $\mathfrak{G}_j$ containing $T(\mathbf{x}(q))$
      for all $p \in i$'th list of c **do**
        if $\measuredangle\big(\mathbf{n}(p), T_*(\mathbf{n}(q))\big) < \gamma\}$ **then**
          if $\|\mathbf{x}(p) - T(\mathbf{x}(q))\| \in [r_i, r_{i+1})$ **then**
            $E_i \leftarrow E_i \cup \{(p, q)\}$
          else if $\exists t \in \{0, \ldots, K-1\} : \|\mathbf{x}(p) - T(\mathbf{x}(q))\| \in [r_t, r_{t+1})$ **then**
            $E_t \leftarrow E_t \cup \{(p, q)\}$
          end if
        end if
      end for
    end for
  end for
  $E_i \leftarrow \text{sortMinFirst}(E_i)$
  for $j \leftarrow 1, |E_i|$ **do**
    $(p, q) \leftarrow E_i[j]$
    $s \leftarrow \text{ propertySetIndex}(p, q)$
    if $p \in P_s' \wedge q \in Q_s'$ **then**
      $f_{l+1}^* \leftarrow f_l^* \cup \{(p, q)\}$
      $P_s' \leftarrow P_s' \setminus \{p\}$
      $Q_s' \leftarrow Q_s' \setminus \{q\}$
      $l \leftarrow l + 1$
    end if
  end for
end for
$f_{opt} \leftarrow \arg\max_l \text{score}^*(P, Q; f_l, T)$

This can be be achieved due to the preprocessing step, in which we build up the point distance grids, and due to an additional cost in terms of memory. For drug-sized molecules, however, this memory cost can be neglected.

5.4 Reduction of Pairwise Matchings

As result of the point matching algorithm described in the previous sections, we get many similar pairwise alignments, particularly around pairwise alignments with high score. This makes it difficult to quickly grasp the most promising alignments. It is therefore often desired to filter out similar pairwise alignments and only keep a set of distinct alignments with large score. In order to avoid confusion, we want to stress that we only compare pairwise alignments of the same two molecular structures, i.e. the same two surface point sets. The two point sets will be referred to as the reference point set and the query point set. While the reference point set is fixed, the query point set changes its position in space according to a particular pairwise matching. Determination of diverse pairwise matchings requires the definition of a similarity measure. We define the similarity measure based on the matching transformation.

5.4.1 Similarity of Matching Transformations

Let us recall, that for a given pairwise matching f, the relative position of the query point set to the reference point set is defined by the *matching transformation* T_f (cf. Definition 5.1.8) which can be computed efficiently. Now we can be more precise about what we mean by diverse pairwise matchings. Two matchings are considered as being diverse, if the distance between their corresponding matching transformations is larger than some predefined threshold δ_T w.r.t. the distance measure defined below.

Definition 5.4.1 (Distance between Matching Transformations)**.** Let P be the reference point set and let Q be the query point set. Let f_1 and f_2 be two pairwise matchings on P and Q and let T_{f_1} and T_{f_2} be their corresponding matching transformations. Then the distance between T_{f_1} and T_{f_2} w.r.t. the coordinates of Q can be defined as the root mean square distance of the transformed coordinates of Q, i.e.,

$$d(T_{f_1}, T_{f_2}; Q) := \sqrt{\frac{\sum_{q \in Q} \| T_{f_1}(\mathbf{x}(q)) - T_{f_2}(\mathbf{x}(q)) \|}{|Q|}} \; .$$

Determining a set of diverse pairwise matchings with respect to some threshold δ_T requires the determination of all pairs of pairwise matchings whose matching transformations have a distance smaller than δ_T. The straightforward approach is to compute the distance between the matching transformations for all pairs of pairwise matchings. The run time of this approach, however, is quadratic and in practice it might be very time-consuming for a large number of pairwise matchings. We can improve this run time by exploiting the following observation. For a given matching transformation, we can eliminate a large number of matching transformations by looking at the translational part of the transformation only. If the translations of two transformations differ by more than δ_T, then the

distance between the transformations is also larger than δ_T. This is formulated in the following lemma.

Lemma 5.4.2 (Lower Bound for Distance of Two Transformations)**.** Let $Q = \{q_1, \ldots, q_n\}$ be a set of points with coordinates $\mathbf{X}(Q) := \{\mathbf{x}(q_i)\}_{i=1}^n \subset \mathbb{R}^3$, $n > 0$. Let $\widetilde{T}$ and T be two matching transformations with rotational parts $\widetilde{R}$ and R, and translational parts $\tilde{t}$ and t, respectively (cf. Definition 5.1.4). Then the following inequality holds

$$d(\widetilde{T}, T; Q) \geq \|\tilde{t} - t\| \ , \text{ i.e.,}$$

the distance between $\tilde{t}$ and t constitutes a lower bound for $d(\widetilde{T}, T; Q)$.

Proof. Without loss of generality, we assume that $\widetilde{T}$ is the identity transformation, i.e. $\widetilde{R} = I$ and $\tilde{t} = 0$, and that $\mathbf{X}(Q)$ is centered at the origin, i.e. $\frac{1}{n}\sum_{i=1}^n \mathbf{x}(q_i) = 0$. If we apply transformation T to the points $\mathbf{x}(q_i) \in \mathbf{X}(Q)$, then we get the transformed points $\tilde{\mathbf{x}}(q_i) = R\mathbf{x}(q_i) + t$. We have to show that

$$\sqrt{\frac{\sum_{i=1}^n (\mathbf{x}(q_i) - \tilde{\mathbf{x}}(q_i))^2}{n}} \geq \|\tilde{t} - t\| = \|t\| \ , \text{ i.e.}$$

$$\sum_{i=1}^n (\mathbf{x}(q_i) - \tilde{\mathbf{x}}(q_i))^2 \geq nt^2 \tag{5.5}$$

We can convert $\tilde{\mathbf{x}}(q_i)^2$ to

$$\begin{aligned}
\tilde{\mathbf{x}}(q_i)^2 &= (R\mathbf{x}(q_i) + t)^\mathsf{T}(R\mathbf{x}(q_i) + t) \\
&= (\mathbf{x}(q_i)^\mathsf{T}R^\mathsf{T} + t^\mathsf{T})^\mathsf{T}(R\mathbf{x}(q_i) + t) \\
&= \mathbf{x}(q_i)^\mathsf{T}R^\mathsf{T}R\mathbf{x}(q_i) + 2(t^\mathsf{T}R\mathbf{x}(q_i)) + t^\mathsf{T}t \\
&= \mathbf{x}(q_i)^\mathsf{T}I\mathbf{x}(q_i) + 2(t^\mathsf{T}R\mathbf{x}(q_i)) + t^\mathsf{T}t \\
&= \mathbf{x}(q_i)^2 + 2(t^\mathsf{T}R\mathbf{x}(q_i)) + t^2 \ .
\end{aligned} \tag{5.6}$$

Using Equation (5.6) we get

$$\begin{aligned}
\sum_{i=1}^n (\mathbf{x}(q_i) - \tilde{\mathbf{x}}(q_i))^2 &= \sum_{i=1}^n (\mathbf{x}(q_i)^2 - 2\mathbf{x}(q_i)^\mathsf{T}\tilde{\mathbf{x}}(q_i) + \tilde{\mathbf{x}}(q_i)^2) \\
&= \sum_{i=1}^n (\mathbf{x}(q_i)^2 - 2\mathbf{x}(q_i)^\mathsf{T}(R\mathbf{x}(q_i) + t) + \mathbf{x}(q_i)^2 + 2(t^\mathsf{T}R\mathbf{x}(q_i)) + t^2) \\
&= 2\sum_{i=1}^n \mathbf{x}(q_i)^2 - 2\sum_{i=1}^n \mathbf{x}(q_i)^\mathsf{T}(R\mathbf{x}(q_i) + t) + 2\sum_{i=1}^n (t^\mathsf{T}R\mathbf{x}(q_i)) + nt^2 \\
&= 2\underbrace{\sum_{i=1}^n (\mathbf{x}(q_i)^2 - \mathbf{x}(q_i)^\mathsf{T}R\mathbf{x}(q_i))}_{\geq 0} - 2t^\mathsf{T}\underbrace{\sum_{i=1}^n (\mathbf{x}(q_i) - R\mathbf{x}(q_i))}_{=0} + nt^2 \ .
\end{aligned} \tag{5.7}$$

□

5.4.2 Determination of Diverse Pairwise Matchings

Using Lemma 5.4.2 we can easily filter out most pairs of matching transformations that do not have to be considered. This can be achieved using a three-dimensional grid similar to the *point distance grid* described in Section 5.3.4. Into this grid we insert the translations of all matching transformations. Then, for a given matching transformation we can query the grid for all those matching transformations with a translation being close enough to fulfill the distance threshold constraint.

Given a set F of pairwise matchings of two point sets P and Q, the set $F_{\delta_T} \subseteq F$ of diverse pairwise matchings w.r.t. δ_T is generated by the following greedy method given by Algorithm 5.4. This set has the following two properties.

1. $\forall f', f'' \in F_{\delta_T} : d(T_{f'}, T_{f''}; Q) > \delta_T$.
2. $\forall f \in F \, \exists f' \in F_{\delta_T} : d(T_f, T_{f'}; Q) \leq \delta_T$.

Algorithm 5.4 Determination of diverse pairwise matchings.

Input: set of pairwise matchings F on P and Q, and distance threshold δ_T
Output: set of diverse pairwise matchings F_{δ_T}

1: $F_{\delta_T} \leftarrow \emptyset$
2: **while** $F \neq \emptyset$ **do**
3: $\quad f_{max} \leftarrow \arg\max_{f \in F} \text{score}^*(P, Q; f)$
4: $\quad F_{\delta_T} \leftarrow F_{\delta_T} \cup \{f_{max}\}$
5: $\quad N \leftarrow \{f \in F \mid f \neq f_{max} \wedge d(T_{f_{max}}, T_f; Q) \leq \delta_T\}$ ▷ Neighbored matchings.
6: $\quad F \leftarrow F \setminus N$
7: **end while**

5.5 Molecular Data

For the evaluation of our surface alignment algorithm, we used two sets of molecules, namely 8 *thermolysin inhibitors* and 7 *HIV-1 protease inhibitors*. For both sets of molecules, the complex structures were available from the PDB [1]. An alignment of the complex structures can be easily done, since the protein structures are well preserved. This gives us an "experimental alignment" of the inhibitors, too, which enables us to quantify the quality of the algorithm. In the following, the active conformers from the complex structures will be referred to as *experimental conformers*.

We also generated ensembles of conformers for each molecule to evaluate how well the algorithm is able to identify *active conformers* (cf. Definition 2.4.2), i.e. conformers that are similar to the experimental conformers w.r.t. the structural features responsible for binding to the proteins.

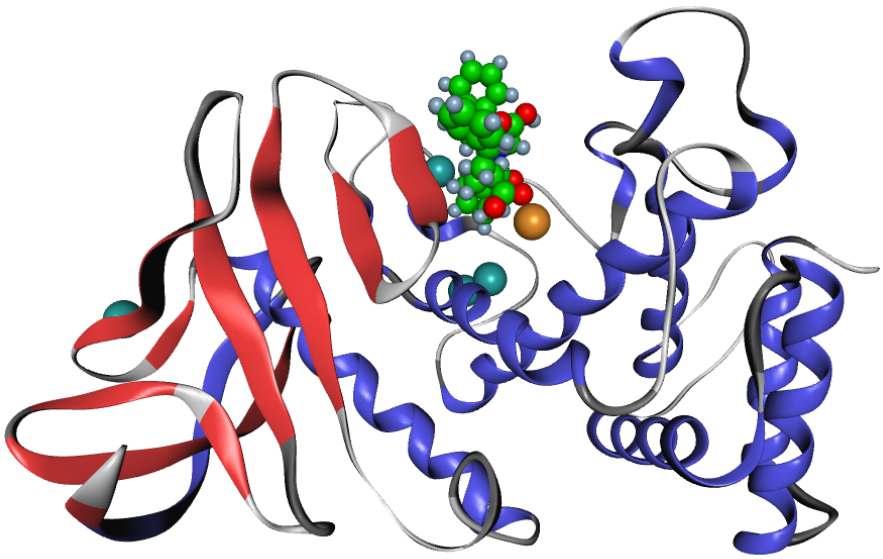


Figure 5.4: Metalloendopeptidase thermolysin complexed with the inhibitor from crystal structure 1THL. Thermolysin is depicted by its secondary structure representation consisting of helices (blue) and β-sheets (red). The inhibitor is shown in its bound state. The radii of its atomic spheres are half the van der Waals radii. The van der Waals sphere of the zinc atom is depicted in brown color, while the van der Waals spheres of the calcium atoms are shown in cyan.

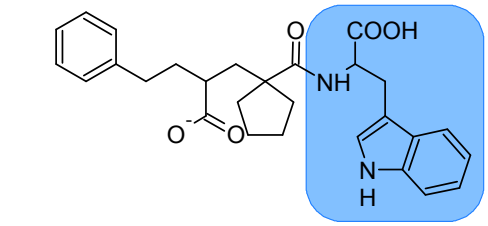
5.5.1 Thermolysin Inhibitors

Crystal Structures

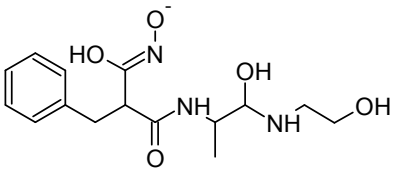
Thermolysin (TLN, EC-number 3.4.24.27 [132]) is a thermostable *endopeptidase* from *Bacillus thermoproteolyticus* [133]. Endopeptidases chemically break peptide bonds somewhere in the middle of the molecule, in contrast to *exopeptidases*, which remove amino acids at the end of the amino acid chain. Thermolysin belongs to the family of extracellular proteases. It has a zinc ion in its active site [99], thus, it belongs to the class of metalloendopeptidases. Furthermore, thermolysin also contains 4 calcium atoms, which are not located in the active site. A complex structure of thermolysin with an inhibitor (1THL), obtained from the PDB [1], is shown in Figure 5.4.

There exist many complex structures of TLN with different inhibitors in the PDB [1]. Cosgrove et al. [35] and Hofbauer [81] used a subset of 8 inhibitors to test their surface alignment programs SPAT and SURFCOMP, respectively (cf. Section 3.3.4). In order to be able to compare the results obtained with our approach, we use the same subset. These 8 inhibitors were extracted from the following complex structures: 1THL, 1TLP, 1TMN, 3TMN, 4TMN, 5TMN, 6TMN, and 5TLN. We will use the PDB keys of the complex structures to denote their corresponding inhibitors. The inhibitors' structural formulas are shown in Figure 5.5.

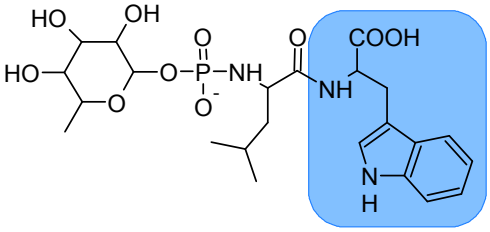
A detailed look at the structural formulas of the 8 inhibitors (cf. Figure 5.5) reveals that



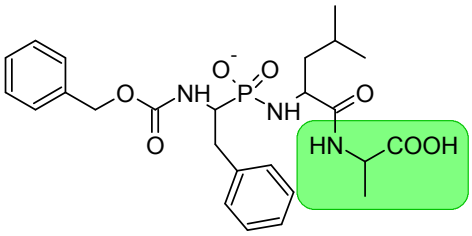
1THL



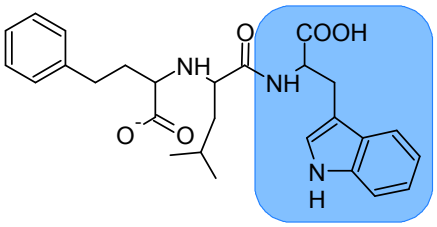
5TLN



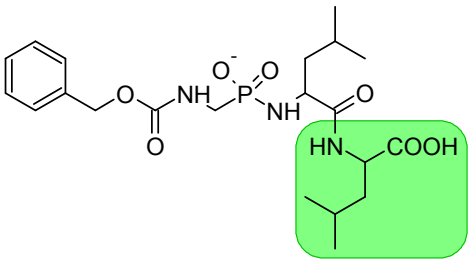
1TLP



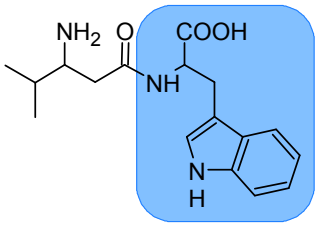
4TMN



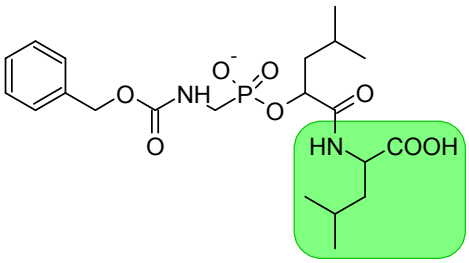
1TMN



5TMN



3TMN



6TMN

Figure 5.5: Structural formulas of eight thermolysin inhibitors. The blue boxes in the left column highlight the amino acid tryptophan, while the green boxes in the right column highlight aliphatic amino acids (alanine in 4TMN, leucine in 5TMN and 6TMN).

Table 5.1: Information about conformer ensembles of the eight thermolysin inhibitors. *First column:* inhibitor name. *Second column:* number of conformers generated by ConFlow. *Third column:* global energy minimum E_{pot}^{min}. *Forth column:* number of selected conformers in the representative ensemble, denoted by $\#C_{ens}$. *Fifth column:* rms distance of conformer closest to experimental conformer (hydrogen atoms were not considered). *Sixth column:* energy of conformer closest to experimental conformer, denoted by $E_{pot}^{closest}$.

	no. sampled conformers	E_{pot}^{min}	$\#C_{ens}$	$\min_{q \in C_{ens}} \text{rmsd}(q, q_{act})$	$E_{pot}^{closest}$
1THL	11437	279 kJ	119	0.42 Å	313 kJ
1TLP	9797	86 kJ	80	0.45 Å	129 kJ
1TMN	8485	317 kJ	116	0.24 Å	361 kJ
3TMN	21480	147 kJ	98	0.11 Å	175 kJ
5TLN	22103	79 kJ	152	0.64 Å	117 kJ
4TMN	9490	116 kJ	80	0.51 Å	155 kJ
5TMN	11147	57 kJ	68	0.29 Å	100 kJ
6TMN	11119	104 kJ	77	0.39 Å	129 kJ

they can be divided into three groups [81]. The first group consists of the inhibitors 1THL, 1TLP, 1TMN, and 3TMN, all of which contain the amino acid *tryptophan*, highlighted by the light blue box in Figure 5.5. The second group consists of 4TMN, 5TMN, and 6TMN, all of which have an *aliphatic* amino acid (alanine or leucine) at the C-terminal end, highlighted by the light green box in Figure 5.5. The third group consists of only one inhibitor, namely 5TLN, which is quite different from the other inhibitors. All inhibitors except 3TMN and 5TLN are complexed to the zinc ion in the active site of thermolysin via a negatively charged deprotonated carboxyl- or phosphate-like group [81]. While 3TMN is not complexed to the zinc ion at all, 5TLN has a charged hydroxamic acid group, which complexes with the zinc ion. Among the 8 inhibitors there are two that are almost identical, namely 5TMN and 6TMN, where the NH-group of 5TMN has been replaced by a single oxygen atom in 6TMN.

Conformers of the Thermolysin Inhibitors

Using the experimental conformer of each thermolysin inhibitor as starting point, conformation analysis (cf. Section 2.4) was carried out with the program ConFlow [125]. ConFlow uses a systematic approach to sample the configurational space. To each thus generated conformer, energy minimization is applied using the molecular force field MMFF [73] and the minimizer Rprop [27]. The systematically sampled conformers as well as the conformers generated during the minimization procedure are stored by ConFlow. The numbers of conformers generated by ConFlow are given in Table 5.1, second column.

From these conformers, a representative set of low-energy conformers was generated by applying the following two steps.

1. Select all conformers with potential energy within an energy window of 50 kJ from the global energy minimum (cf. Table 5.1, third column). An energy window of 50 kJ has been generally accepted [26].

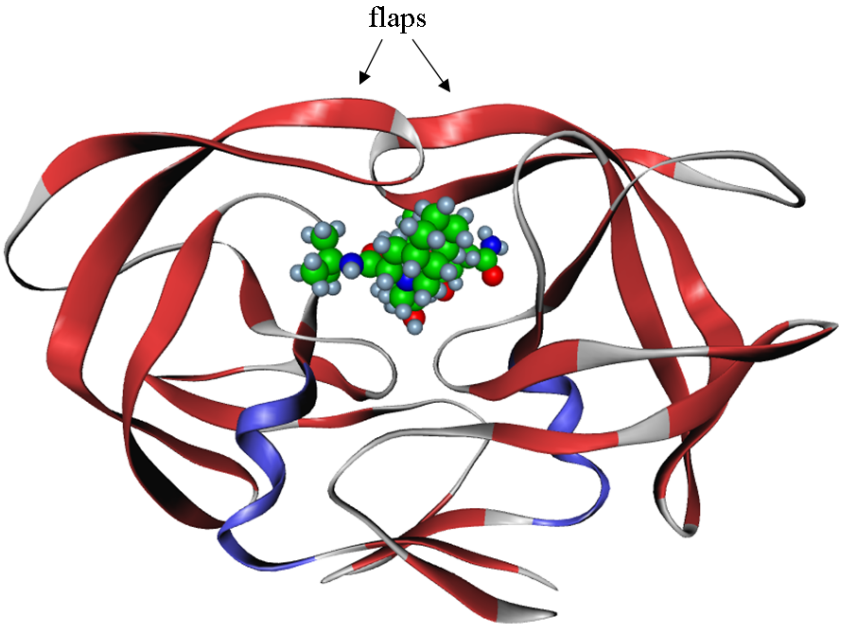


Figure 5.6: HIV-1 protease complexed with inhibitor saquinavir (PDB-entry 1MTB). The protease is depicted by its secondary structure representation consisting of helices (blue) and β-sheets (red). The van der Waals spheres of the inhibitor's atoms are colored according to their atom types, and the spheres have been scaled to half their size. The protease consists of two monomers, left and right. The two loops on top act as flaps, i.e., they need to open to let an amino acid chain or an inhibitor in.

2. From the low-energy conformers, select a number of conformers, such that at least one conformer is close to the experimental conformer (rms distance $< 1.0\,\text{Å}$), and such that the selected conformers cover the whole configurational low-energy subspace identified by ConFlow.

The number of conformers in each ensemble was chosen w.r.t. the number of low energy conformers and the size of the molecules. The details of the ensembles are given in Table 5.1.

5.5.2 HIV-1 Protease Inhibitors

Human Immunodeficiency Virus

The *human immunodeficiency virus* (HIV) is one of the greatest current health threats. The first infections with HIV were noticed in 1981 in the USA [8]. Since then, HIV has spread all over the world and has emerged as a pandemia. HIV is a retrovirus which causes depletion of helper T-lymphocytes [8]. The gradual depletion of these cells finally causes the *acquired immune deficiency syndrome* (AIDS). At this stage, the patient is increasingly susceptible to infections of bacterial, viral or fungal origin as well as to certain

types of cancer. There exist two types of HIV, namely HIV-1 and HIV-2. HIV-1 is more virulent than HIV-2 and is transmitted more easily than HIV-2. Hence, while HIV-2 is largely confined to West-Africa [146], HIV-1 has reached all continents and amounts for the majority of infections throughout the world.

As all viruses, HIV is not able to replicate itself but needs the help of a host cell for replication. To enter the cell, the virus first attaches to the surface of the cell. The next step is the fusion of the virus membrane with the host cell membrane with the result that the viral content is released into the host cell cytosol [8]. In the host cell cytosol, the single-stranded RNA of HIV complexes with the *HIV reverse transcriptase*, which catalyzes a double-stranded DNA by reverse transcription. Subsequently, the DNA is permanently integrated into the host genome. As a cause of this, the host cell helps to replicate the HI virus by building all of its components. However, before the new virus can separate from the host cell, the synthesized amino acid chains need to be cut. This is realized by the *HIV protease*.

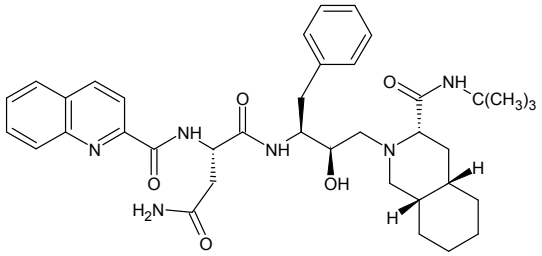
In order to hinder HIV to replicate, in principal, every step in the replication cycle is a potential target for anti-viral treatment. The two targets most widely aimed at are the reverse transcriptase and the protease. If one of them can be inhibited, the virus can not be completely replicated. Currently, there exist about 20 drugs inhibiting either the reverse transcriptase or the protease. In order to be effective, a combination of two, three, or four drugs seems to be necessary [19]. However, HIV has a very high mutation rate, which quickly leads to mutations that are resistant to certain drugs. Hence, there is a large need to develop new drugs [8], in particular, drugs which are less likely to enforce the development of drug-resistant mutants.

Crystal Structures

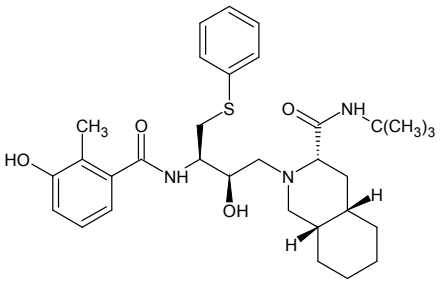
As mentioned above, the HIV-1 protease is one of the two main targets for anti-viral treatment. The HIV-1 protease is a symmetric dimer with each monomer having 99 amino acids. Figure 5.6 shows the 3-dimensional structure of the HIV-1 protease complexed with the inhibitor saquinavir. Note the two flaps in the upper part of the image, which need to open to let an amino acid chain or an inhibitor in [151].

During the last one and a half decades, intensive studies on compounds being able to inhibit the HIV-1 protease have been accomplished. The first approved inhibitor was *saquinavir*, which has been in clinical use since 1995 [8]. Further peptidic inhibitors followed, like *nelfinavir*, *ritonavir*, *lopinavir*, *indinavir*, and *amprenavir*. All of these inhibitors show a high similarity in the molecular scaffold, i.e. in the central part of the structural formulas (cf. Figure 5.7). In 2005, the first non-peptidic inhibitor, *tipranavir* [4], was approved. Compared to the peptidic inhibitors, this inhibitor shows clear distinctions in the structural formula, in particular in the molecule's scaffold.

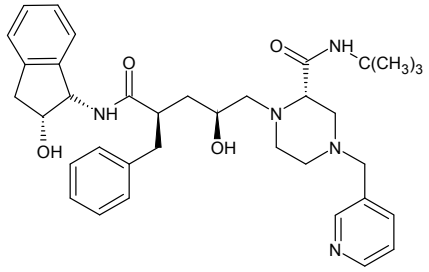
For all of these inhibitors, one or more crystallized complexes with the wild-type HIV-1 protease or some mutant are available: 1C6X, 1C6Y, 1C6Z, 1C70, 1FB7 and 1MTB for saquinavir; 1HPV for amprenavir; 1C6Y, 1HSG, 1HSH, 1K6C, 1K6P, 1K6T, 1K6V, 1SDT, 1SDU, 1SDV, and 1SGU for indinavir; 1MUI and 1RV7 for lopinavir; 1OHR for nelfinavir; 1HXW, 1N49, and 1SH9 for ritonavir; and 1D4Y for tipranavir. Note, that this



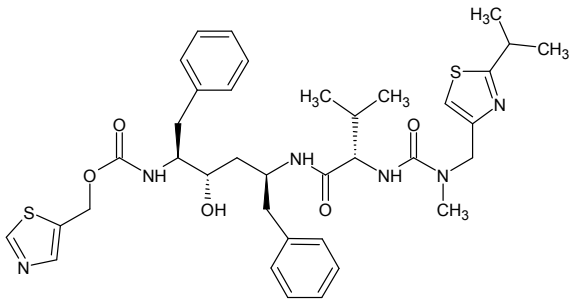
saquinavir (SQV)



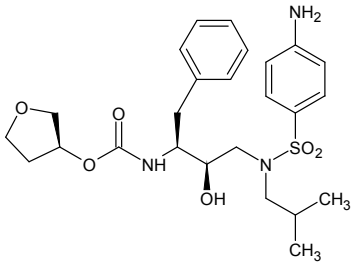
nelfinavir (NFV)



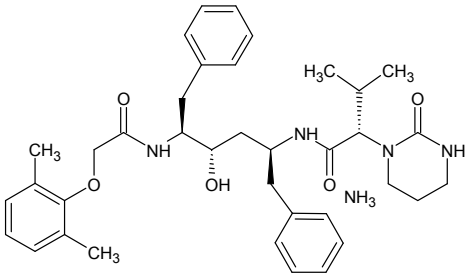
indinavir (IDV)



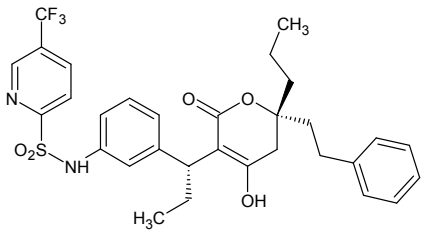
ritonavir (RTV)



amprenavir (APV)



lopinavir (LPV)



tipranavir (TPV)

Figure 5.7: Structural formulas of seven HIV-1 protease inhibitors.

Table 5.2: Information about conformer ensembles of the seven HIV-1 protease inhibitors. *First column:* inhibitor name. *Second column:* number of conformers generated by the conformation analysis program. *Third column:* number of selected conformers in the representative ensemble. *Fourth column:* rms distance of conformer closest to experimental conformer (hydrogen atoms were not considered).

	no. sampled conformers	$\#C_{ens}$	$\min_{q \in C_{ens}} \mathrm{rmsd}(q, q_{act})$
SQV	624	250	0.64 Å
IDV	521	250	0.31 Å
APV	309	120	0.84 Å
NFV	221	100	0.29 Å
RTV	2165	350	0.58 Å
LPV	800	150	0.25 Å
TPV	226	100	0.36 Å

list may not be complete.

Conformers of the HIV-1 Protease Inhibitors

Similarly to the thermolysin inhibitors, conformation analysis for the HIV-1 protease inhibitors was carried out with the experimental conformers as starting points. In contrast to the thermolysin inhibitors, however, we did not use the conformation analysis program ConFlow [125], but a recent improvement of the program ZibMol [58], which is a combination of Hybrid Monte Carlo sampling (cf. Section 2.4.1) and an efficient perturbation of metastability, which allows to generate a complete decomposition of configurational space into metastable subsets.

The numbers of conformers generated by the conformation analysis program were in the range of 221 and 2165. For each inhibitor, a representative set of conformers was selected by reducing the given conformers to a minimum of 100 (for NFV and TPV) and a maximum of 350 conformers (for RTV), such that at least one conformer was close to the experimental conformer (rms distance < 1.0 Å). Note, that the large number of conformers generated for RTV is due to its larger number of degrees of freedom. The reduction of conformers was achieved similarly to the reduction of the thermolysin inhibitor conformers by selecting conformers that cover the whole configurational subspace identified by the conformation analysis program. The number of conformers in each ensemble was chosen w.r.t. the original number of conformers and the size of the molecules. The details of the ensembles are given in Table 5.2.

5.6 Experimental Results

Several experiments were conducted to evaluate the pairwise surface alignment approach presented in Sections 5.2 through 5.4. All experiments were done with both sets of inhibitors described in the previous section. Before we describe the experiments in detail,

Table 5.3: Numbers of property points used for representing molecular surface shape and physicochemical properties of experimental conformers of thermolysin inhibitors.

	shape	donor	acceptor	negative MEP	positive MEP
1THL	117	41	115	147	7
1TLP	122	110	203	148	8
1TMN	122	56	137	150	8
3TMN	77	73	98	16	18
5TLN	85	87	147	126	2
4TMN	139	47	171	165	9
5TMN	119	52	177	148	7
6TMN	119	41	177	146	6

Table 5.4: Numbers of property points used for representing molecular surface shape and physicochemical properties of experimental conformers of HIV-1 protease inhibitors.

	shape	donor	acceptor	negative MEP	positive MEP
SQV	160	38	89	76	51
IDV	153	33	66	69	48
APV	125	28	70	52	46
NFV	135	31	60	74	46
RTV	171	23	105	60	50
LPV	155	23	62	59	47
TPV	139	20	90	46	62

we shortly want to recall the algorithmic framework and its parameters.

5.6.1 Algorithmic Framework and Parameters

Algorithm

The algorithmic approach for pairwise surface alignment consists of the following four steps:

1. Generation of surface points (cf. Chapter 4).
2. Generation of initial transformations (cf. Section 5.2).
3. Alignment optimization for each initial transformation (cf. Section 5.3).
4. Selecting a set of diverse pairwise alignments (cf. Section 5.4).

For each step of the algorithm, a few decisions had to be made, which shall be described shortly.

For the first step we need to decide how dense the points representing each property should be distributed on the molecular surface. Point distances between 1 and 2 Å seem to be appropriate. We weighted the properties such that the numbers of points used for the representation of all properties were of comparable magnitude while favoring

physico-chemical properties over shape. Points representing shape were distributed on the molecular surfaces with an approximate geodesic distance of 2 Å, while points representing donor/acceptor points were distributed within the respective regions with an approximate geodesic distance of 1 Å. The geodesic distance of the points representing the molecular electrostatic potential (MEP) varied according to the MEP scalar field given on the molecular surface. The smallest geodesic distance observed between MEP points was about 1 Å. The number of property points that were generated for the experimental conformers can be found in Table 5.3 for the thermolysin inhibitors, and in Table 5.4 for the HIV-1 protease inhibitors.

The second step offers a choice between atom-based and surface-based generation of initial transformations. For the first type of experiments, i.e. the comparison of experimental conformers only, we compared both approaches and obtained similar results. Hence, for the other types of experiments, we only used the surface-based approach, since it is more general.

In the third step we had to make a decision on the properties to be considered. In our experiments we investigated the influence of the used properties on the alignment results.

Finally, the fourth step required a choice of δ_T (cf. Section 5.4), i.e. of the distance threshold which determines if two transformations are too close to each other. In all our experiments we set δ_T to 1.0 Å.

Algorithmic Parameters

The parameters used in our experiments are shortly described below and summarized in Table 5.5. We did not optimize the parameters for each set of molecules in order to show that with the same set of parameters, different sets of molecules can be successfully aligned.

Generation of Initial Transformations. The first parameter for the generation of initial transformations is the *distance constraint* d (cf. Definition 5.2.1), used for building up the *distance compatibility graph*. Two nodes in the distance compatibility graph are connected with each other by an edge, if the distances between the pairs of points corresponding to the nodes in the graph deviate by a value of at most d.

The second parameter is the minimum size c (cf. Section 5.2), substructures used to compute initial transformations need to have. This parameter needs to be adjusted to the size of the molecules to be aligned. Together with the parameter d it determines the number of initial transformations being generated.

If the initial transformations are generated using surface points, three more parameters are used. The first of which is the *geodesic distance* D (cf. Chapter 4), two neighbored surface points should be approximately apart from each other. In our experiments, we used a value of 4.0 Å. The second parameter used for surface-based generation of initial transformations is the *angular distance constraint* β (cf. Definition 5.2.2). This value specifies how much the angles of the surface normals of two pairs of surface points may deviate to allow a matching of these pairs of surface points. Thus, if the angular distance is smaller than β, the nodes in the distance compatibility graph representing these two pairs of surface

Table 5.5: Experimental Parameters. Experiments T1-T5 are applied to the thermolysin inhibitors, while experiments H1-H5 are applied to the HIV-1 protease inhibitors. Experiments T1, T2, H1, and H2 align the experimental conformers pairwisely, while experiments T3, T4, T5, H3, H4, and H5 align the query molecule conformers, taken from an ensemble of calculated conformers, to the experimental conformers of the reference molecules. The letters "a" and "s" denote "atom" and "surface", respectively.

Experiment	Generation of Initial Transformations						Alignment Optimization			Results in Table
	general			surface-based			general		surface-based	
	type	d	c	D	β	shapeSim$_{min}$	type	δ	γ	
T1a	a	0.20 Å	4	–	–	–	s	2.0 Å	60°	A.3
T1b	a	0.20 Å	4	–	–	–	s	2.0 Å	60°	A.4
T1c	a	0.20 Å	4	–	–	–	s	2.0 Å	60°	A.5
T1d	a	0.20 Å	4	–	–	–	s	2.0 Å	60°	A.6
T2a	s	0.50 Å	4	4.0 Å	60°	0.5	s	2.0 Å	60°	A.7
T2b	s	0.50 Å	4	4.0 Å	60°	0.5	s	2.0 Å	60°	A.8
T2c	s	0.50 Å	4	4.0 Å	60°	0.5	s	2.0 Å	60°	A.9
T2d	s	0.50 Å	4	4.0 Å	60°	0.5	s	2.0 Å	60°	A.10
T3	a	0.20 Å	4	–	–	–	a	2.0 Å	–	A.11
T4	s	0.50 Å	4	4.0 Å	60°	0.5	s	2.0 Å	60°	A.12
T5	a	0.20 Å	4	–	–	–	s	2.0 Å	60°	A.13
H1a	a	0.20 Å	5	–	–	–	s	2.0 Å	60°	A.17
H1b	a	0.20 Å	5	–	–	–	s	2.0 Å	60°	A.18
H1c	a	0.20 Å	5	–	–	–	s	2.0 Å	60°	A.19
H1d	a	0.20 Å	5	–	–	–	s	2.0 Å	60°	A.20
H2a	s	0.50 Å	5	4.0 Å	60°	0.5	s	2.0 Å	60°	A.21
H2b	s	0.50 Å	5	4.0 Å	60°	0.5	s	2.0 Å	60°	A.22
H2c	s	0.50 Å	5	4.0 Å	60°	0.5	s	2.0 Å	60°	A.23
H2d	s	0.50 Å	5	4.0 Å	60°	0.5	s	2.0 Å	60°	A.24
H2e	s	0.55 Å	5	4.0 Å	60°	0.5	s	2.0 Å	60°	A.25
H3	s	0.50 Å	5	4.0 Å	60°	0.5	a	2.0 Å	–	A.26
H4	a	0.20 Å	5	–	–	–	s	2.0 Å	60°	A.27
H5	a	0.20 Å	5	–	–	–	a	2.0 Å	–	A.28

points will be connected by an edge, if they additionally satisfy the distance constraint d. A value of 60° seems to be appropriate for β. The last parameter, shapeSim_{min} (cf. Equation 5.2), gives the minimal shape similarity, two surface points need to have to get matched. That is to say, two surface points, one from each surface, will be represented by a single node in the distance compatibility graph only if their surface shape similarity is larger than shapeSim_{min}. In all experiments we used a value of 0.5.

Alignment Optimization. Apart from the properties to be represented by surface points and their respective weights, there are only two parameters used in the alignment optimization step. These are the *distance threshold* δ (cf. Section 5.3.4), and the *angular distance threshold* γ (cf. Section 5.3.2). Both thresholds determine whether or not two surface points are allowed to get matched in the alignment optimization step. The first parameter states that the distance between two surface points to get matched may be at most δ. Similarly, the second parameter states that the angle between the surface normals of two surface points to get matched may be at most γ, which we always set to 60°.

5.6.2 Experiments

In order to be able to verify the computed alignments, we aligned the complex structures of each group of protein-ligand complexes. Aligning these complex structures could be easily done, since the protein structures vary only slightly. This gave us an "experimental alignment" of the inhibitors, which could be used to compare the results of the computed alignments with. For evaluating the quality of the obtained alignments, we computed the root mean square (rms) distance between the conformers aligned by the algorithm and their corresponding experimental conformers in the experimentally given alignment. We considered all atoms except the hydrogen atoms. For the experimental conformers, including the hydrogen atoms should not make a large difference. However, the influence on the rms distances between experimental conformers and conformers from the ensemble might be considerable, since in the conformation analysis, dihedral angles of terminating atoms (mostly hydrogens) were neglected. Furthermore, we also neglected the hydrogen atoms in the alignment procedure.

Before computing the rms distances, we filtered the pairwise alignments such that only distinct pairwise alignments remained (cf. Section 5.4.2). From these pairwise alignments we only considered the 10 top-ranked alignments. All other alignments were neglected. For these top-ranked alignments, the rms distances were computed and the results were collected in tables (cf. Appendix A). Each table entry gives two values, the first of which represents the rms distance of the optimal alignment w.r.t. the scoring function. The second value gives the smallest rms distance from the 10 top-ranked alignments. The first column gives the name of the inhibitor whose experimental conformer was used as reference structure for the alignments given in the respective row. The other inhibitors were used as query structures.

All tables giving numerical results, i.e. the rms distances, were moved to Appendix A for clarity. In the succeeding sections we have tried to summarize the tables and to highlight the most interesting results.

Aims of the Experiments

We conducted several experiments to evaluate the quality of the pairwise surface alignment algorithms described in this chapter. These experiments were of different type and were carried out with different intentions. Basically, four types of experiments were conducted:

I. *Dependence on used properties.* In order to test the dependence of the alignment results on the choice of properties, we performed tests on the experimental conformers using varying properties. That is to say, we aligned the experimental conformers using

   (a) shape points, donor and acceptor points, and MEP points,
   (b) shape points only,
   (c) donor and acceptor points only, and
   (d) MEP points only.

II. *Atom-based versus surface-based generation of initial transformations.* In this type of experiment, we investigated the dependence of the alignment results on the way the initial transformations are generated.

III. *Atom versus surface alignment.* This type of experiment was carried out to show, that surface alignment indeed has preferable properties over atom alignment. We compared the surface alignment algorithm with the atom alignment approach described in [15], which applies point matching to the positions of the atomic nuclei.

IV. *Identification of active conformers from ensembles of conformers.* Finally, we compared how well surface and atom alignment are able to identify active conformers from an ensemble of conformers generated with conformation analysis.

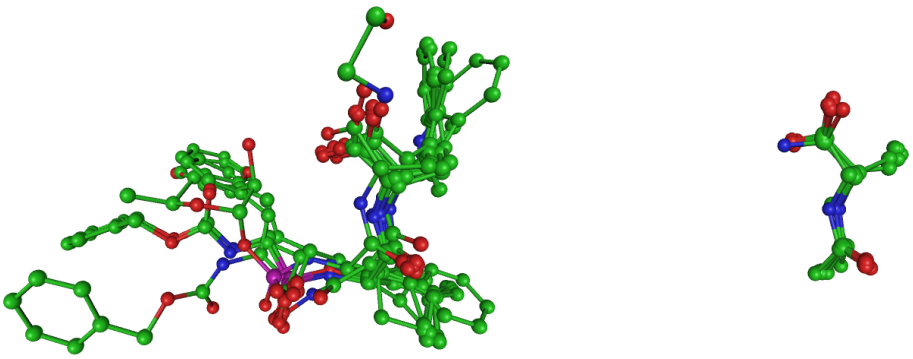
Note that the experiments of type IV are more complicated than finding the experimental conformers, since these were not contained in the ensembles, but only conformers close to the experimental one. Hence, when looking at the results of the type IV experiments, two important points should be considered.

1. The best possible rms distance cannot be better than the smallest rms distance of the conformer closest to the experimental one (cf. Tables 5.1 and 5.2).

2. The rms distance is not always a good measure, since the rms distance can be small even though some (possibly very important) atoms may be completely disoriented, and it can be large even though the important parts of the molecules, i.e. those parts responsible for the binding, are aligned very well. This emphasizes the need for a visual inspection of the alignments.

Thermolysin Inhibitors

The thermolysin inhibitors do not constitute an ideal set of drug-sized molecules to demonstrate the capabilities of a surface alignment algorithm, since all inhibitors have a common



(a) (b)

Figure 5.8: (a) Experimental alignment of all thermolysin inhibitors. (b) Common scaffolds of the thermolysin inhibitors, aligned by least squares-fitting.

Table 5.6: Molecular properties used in experiments T1 and T2.

Experiment	Initial Transformations	Shape	H-Bonding	MEP	Results
T1a	atom-based	×	×	×	Table A.3
T1b	atom-based	×	–	–	Table A.4
T1c	atom-based	–	×	–	Table A.5
T1d	atom-based	–	–	×	Table A.6
T2a	surface-based	×	×	×	Table A.7
T2b	surface-based	×	–	–	Table A.8
T2c	surface-based	–	×	–	Table A.9
T2d	surface-based	–	–	×	Table A.10

scaffold, as can be seen in Figure 5.8. Due to this scaffold, atom alignment performed almost equally well as surface alignment in the comparison of the experimental conformers. But, as we will see later, the atom alignment approach had problems with the identification of active conformers from an ensemble of conformers.

Experimental Conformers. The first experiment that was carried out for the thermolysin inhibitors was to align the experimental conformers with our surface alignment algorithm based on different property sets, namely shape points, donor/acceptor points, and MEP points. The alignment was done both with atom-based initial transformations (experiments T1a-d) and surface-based initial transformations (experiments T2a-d). In both experiments, T1 and T2, the number of initial transformations were approximately the same, as can be seen in Tables A.1 and A.2. The molecular properties that were considered in the respective experiments are given in Table 5.6, while the numerical results can be found in Appendix A.1.1 in Tables A.3 through A.10.

The alignments obtained by experiments T1 and T2, i.e. surface alignment with atom-

Table 5.7: Run times in seconds for experiments T1, T2 and T3.

Experiment	1THL	1TLP	1TMN	3TMN	5TLN	4TMN	5TMN	6TMN
T1a	15.7	28.0	15.4	7.6	13.0	20.8	11.7	11.6
T1b	7.8	9.0	6.9	3.9	4.0	9.9	5.1	5.2
T1c	4.4	9.0	4.8	2.6	3.8	5.4	3.7	3.7
T1d	6.1	8.0	5.7	1.7	3.9	8.4	4.0	4.2
T2a	15.0	29.9	21.9	9.2	16.7	23.7	16.2	17.0
T2b	10.8	9.4	10.6	5.3	6.0	10.6	8.1	8.2
T2c	4.1	9.8	6.8	3.1	5.0	6.0	5.6	5.2
T2d	6.4	7.8	8.0	1.8	5.0	8.9	6.1	6.0
T3	1.6	1.9	1.2	0.7	0.8	1.7	1.1	1.1

based and surface-based initial transformations, respectively, are very similar with a slight preference to the atom-based initial transformations. In both experiments it can be observed, that the alignment algorithm performed best, when all properties were considered (experiments T1a and T2a, cf. Tables A.3 and A.7). This result confirms the necessity to simultaneously consider several molecular properties for the alignment of molecules.

From the experiments which only considered a single property, the donor/acceptor points gave the best results (experiments T1c and T2c, cf. Tables A.5 and A.9), while with the MEP points, problems were only encountered with inhibitor 3TMN (cf. Tables A.6 and A.10). This is due to the fact that inhibitor 3TMN misses the negatively charged group which, in the other molecules, is responsible for binding to the positively charged zinc ion. Surface alignment with shape points only performed worst (experiments T1b and T2b, cf. Tables A.4 and A.8). Here, the main problems were with inhibitor 5TLN which is least similar to the other molecules in terms of surface similarity. These results indicate, that donor/acceptor and MEP points play the most important role in determining the correct alignment.

The fact that the results of the alignments with atom-based initial transformations were slightly better than those with surface-based initial transformations may be due to the common scaffold of the inhibitors. In the case of atom-based initial transformations, this common scaffold already aligns the molecules very close to the experimental positions.

When all properties were considered, both experiments (T1a and T2a) successfully aligned all experimental conformers. In all but eight of 56 pairwise alignments (cf. Table A.7), the rms distance of the top-ranked alignment was below 1.0 Å, and even in these 8 cases the closest rms distance among the 10 top-ranked alignments was below 1.36 Å.

A comparison of these results with the results from atom alignment [15] reveals that even in the case of the experimental conformers, surface alignment performed better (cf. Table A.11). In four cases, the rms distance of the top-ranked atom alignment was above 5.0 Å. In two cases, no rms distance smaller than 1.56 Å could be found among the 10 top-ranked alignments.

The run times of the algorithm for experiments T1, T2 and T3 can be seen in Ta-

ble 5.7. The average run time for the alignment of two molecular surfaces with initial transformations as given in Tables A.1 and A.2 was therefore about 2.5 seconds. With 0.2 seconds on average, the atom alignment algorithm was about 10 to 15 times faster, which might be due to the ten-fold number of points considered in the surface alignment approach.

As mentioned before, the set of thermolysin inhibitors was also used to compare our results with the programs SPAt [35] and SurfComp [81] (cf. Section 3.3.4). A comparison of the results reveals, that our algorithm performed as least as good as SPAt and SurfComp, while it gave much better results for 3TMN and 5TLN. Both programs had problems with aligning these two inhibitors. This superiority of our approach has several reasons. First, our approach considers several properties at once, which, as has been shown, produced much better results than a single property. Second, we represent the surface shape much better due to using more surface points. And third, we optimize the alignment score (matching score) during the alignment procedure, in contrast to SPAt and SurfComp, which only use the score to measure the alignment after the alignment has been performed. In addition to giving better results, our approach is also faster, in particular in the preprocessing step, which takes considerably longer for SPAt and SurfComp.

Ensemble of Conformers. The last experiment we conducted with the thermolysin inhibitors was to identify active conformers from an ensemble of conformers. Here, we used the experimental conformers of all inhibitors in turn as reference structures, and for the query molecules we used the ensembles of conformers (cf. Table 5.1).

Since for the alignment of the experimental conformers the best results were obtained by considering all available properties, in this experiment we also considered all properties. Furthermore, we used the surface-based generation of initial transformations only, since it is more general than the atom-based one. Hence, only two experiments were carried out in the ensemble test, namely one for surface alignment and one for atom alignment. The results can be found in Tables A.12 and A.13, respectively.

In this experiment, the observed rms distances were, in general, larger than those for the experimental conformers. This is mainly due to the problem that the rms distance is often not a good measure, as explained earlier. In most cases where a large rms distance was found, the surface regions important for the binding were well overlaid. The surface alignment algorithm failed to identify a sensible alignment only in two cases, whereas with the atom alignment algorithm, six pairs or molecules could not be aligned correctly. Furthermore, the rms distances were considerably larger for atom alignment. The reason for this lies in the identification of rather small well-fitting scaffolds. While for the experimental conformers, this small scaffold allows a "correct" positioning, since the rest of the molecule is fixed, for the ensemble test, it is important to align larger regions, even though these are less conserved. Our surface alignment algorithm is able to consider less conserved regions much better.

The inhibitor 3TMN performed worst as reference structure for both atom and surface alignment, which can be explained by the small size of the molecule and, in particular, by the missing negatively charged group which is an important factor strongly influencing

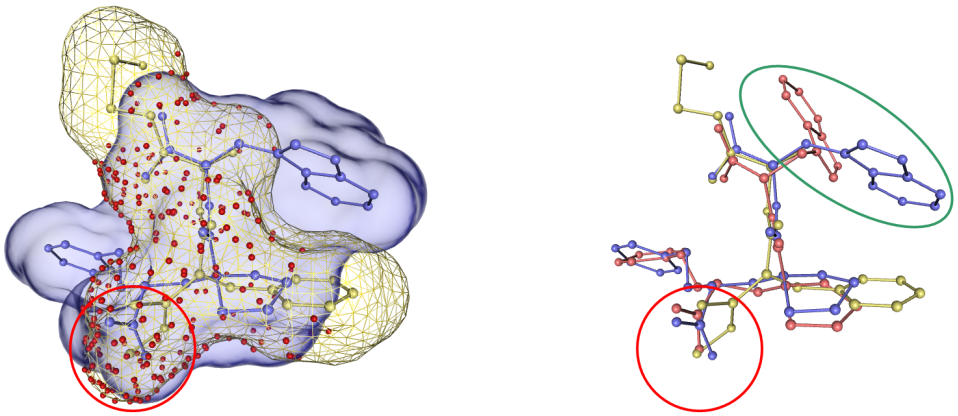


Figure 5.9: Surface alignment of experimental conformer of 5TLN (yellow) with an ensemble conformer of 1THL (blue). This alignment has the highest score and an rms distance of 2.73 Å (cf. Table A.12). *Left:* The image shows the overlaid surfaces together with the matched surface points (red). The negatively charged groups binding to the zinc ion are well overlaid (red circle). *Right:* Comparison of aligned ensemble conformer of 1THL with experimental conformer in experimental position (light red). The large rms distance is due to the wrong orientation of the indole moiety in the right upper corner (green ellipse).

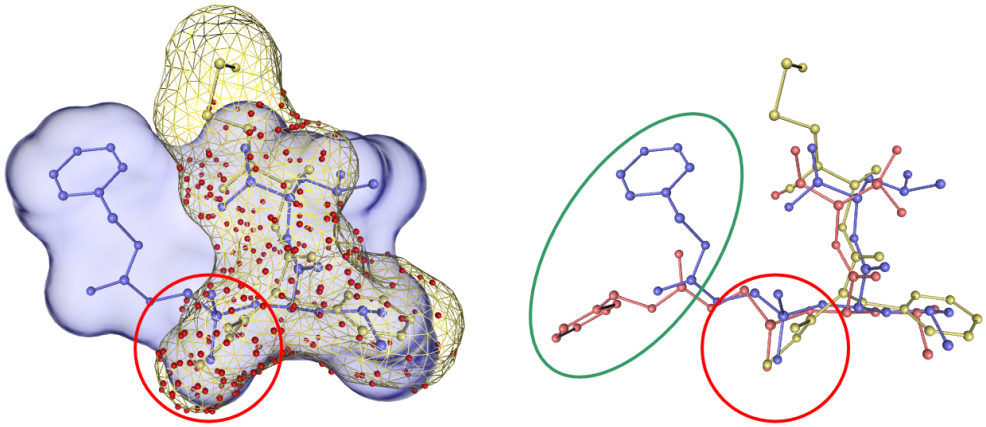


Figure 5.10: Surface alignment of experimental conformer of 5TLN (yellow) with an ensemble conformer of 6TMN (blue). This alignment was the closest alignment with an rms distance of 3.25 Å (cf. Table A.12). *Left:* The image shows the overlaid surfaces together with the matched surface points (red). Note, that the matched surface region is very similar to that shown in Figure 5.9. The negatively charged groups binding to the zinc ion are well overlaid (red circle). *Right:* Comparison of aligned ensemble conformer of 6TMN with its experimental conformer in the experimental position (light red). The large rms distance is due to the wrong orientation of the benzyloxycarbonyl moiety, which can be seen on the left-hand side (green ellipse).

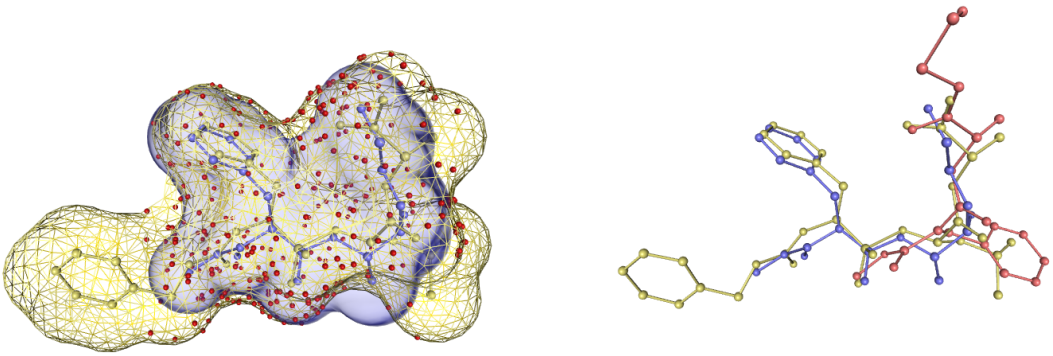


Figure 5.11: Surface alignment of experimental conformer of 4TMN (yellow) with ensemble conformer of 5TLN (blue). This *miss-alignment* has the highest score and an rms distance of 6.56 Å (cf. Table A.12). Even though this alignment is wrong, surface shape and properties are well matched. This exemplifies the difficulty of determining an active conformer from an ensemble. *Left:* The left image shows the overlaid surfaces together with the matched surface points (dark red). *Right:* Comparison of aligned ensemble conformer of 1THL with its experimental conformer in the experimental position (red).

the binding of the other inhibitors.

Example alignments are depicted in Figures 5.9, 5.10, and 5.11. The former two depict successful alignments, yet with a large rms distance. These two alignments exemplify that good alignments with large rms distance exist and that the rms distance is a problematic measure. The last example represents a miss-alignment if compared to the experimental alignment, yet overlays molecular shape and properties very well.

HIV-1 Protease Inhibitors

The set of HIV-1 protease inhibitors (HIVPIs) differs from the set of thermolysin inhibitors in several respects. First, while all thermolysin inhibitors had a common scaffold, among the HIVPIs there exists one inhibitor, tipranavir, that shows a distinct scaffold. This can clearly be seen in Figure 5.12. Second, the HIVPIs are almost completely enclosed by the active site of the HIV-1 protease. This is due to the flaps of the HIV-1 protease (cf. Figure 5.6), which cover the inhibitors during the binding. And third, for each inhibitor there exist two equally likely positions in the active site, since the HIV-1 protease is a symmetric dimer. While the active site, therefore, is also rotationally symmetric, the inhibitors are not.

Experimental Conformers. With the experimental conformers of the HIVPIs we conducted the same experiments (cf. Tables 5.8) as with the experimental conformers of the thermolysin inhibitors. The results of these experiments are given in Tables A.17 through A.26.

Similarly to the thermolysin inhibitors, the best results with surface alignment were

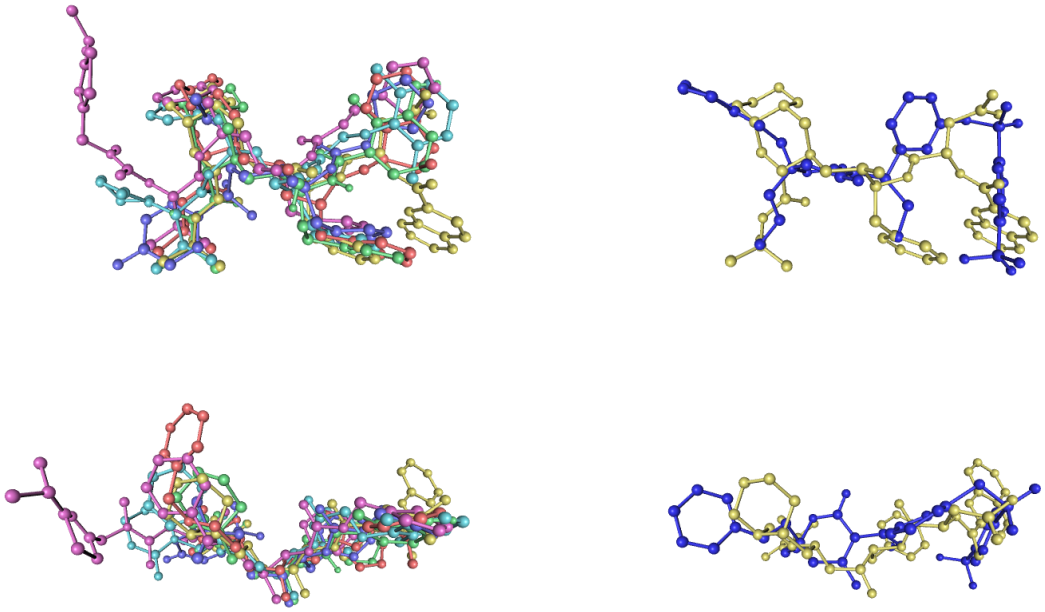


Figure 5.12: Experimental alignments of HIV-1 protease inhibitors. *Left:* All inhibitors except tipranavir. These six inhibitors show a rather similar scaffold. *Right:* Saquinavir (yellow) and tipranavir (blue). The differences in the scaffold can clearly be seen, in particular in the bottom image.

obtained, when all properties were considered. However, in contrast to the thermolysin inhibitors, where the donor/acceptor properties clearly gave the best results, here, all single properties performed equally well, but notably worse than when all properties were considered simultaneously.

Looking at the results of experiments H1a and H2a reveals (cf. Tables A.17 and A.21), that the surface alignment with surface-based initial transformations performed worse than with atom-based initial transformations. The reason for this lies in the considerably smaller number of initial transformations generated with the chosen parameters. We therefore carried out another experiment, H2e (cf. Table A.25), in which we slightly increased the distance constraint parameter d, resulting in an increased number of initial transformations roughly equal to the number of atom-based initial transformations. With this increased number of initial transformations, we obtained results similar to those of experiment H1a.

A comparison of the results of atom alignment, experiment H3 (cf. Table A.26), with surface alignment, experiments H1a and H2e, shows similar results for all HIVPIs except for tipranavir (TPV). For TPV, atom alignment performed considerably worse than surface alignment. With atom alignment, all top-ranked alignments with TPV as reference structure gave an rms distance above 4.9 Å, and in four cases no alignment among the 10

Table 5.8: Molecular properties used in experiments H1 and H2.

Experiment	Initial Transformations	Shape	H-Bonding	MEP	Results
H1a	atom-based	×	×	×	Table A.17
H1b	atom-based	×	–	–	Table A.18
H1c	atom-based	–	×	–	Table A.19
H1d	atom-based	–	–	×	Table A.20
H2a	surface-based	×	×	×	Table A.21
H2b	surface-based	×	–	–	Table A.22
H2c	surface-based	–	×	–	Table A.23
H2d	surface-based	–	–	×	Table A.24
H2e	surface-based	×	×	×	Table A.25

Table 5.9: Run times in seconds for experiments H1, H2 and H3.

Experiment	SQV	IDV	APV	NFV	RTV	LPV	TPV
H1a	13.5	10.5	5.2	8.6	11.8	10.9	11.6
H1b	8.6	6.5	3.3	5.5	8.3	7.1	6.4
H1c	3.9	3.0	1.7	2.5	3.8	3.3	2.6
H1d	2.3	1.9	1.1	1.6	2.1	2.0	1.7
H2a	10.4	6.1	5.7	6.2	9.2	8.5	9.8
H2b	6.3	3.4	3.2	3.5	6.0	5.0	4.4
H2c	1.7	1.4	1.3	1.3	2.0	1.9	1.7
H2d	1.0	0.8	0.7	0.7	1.0	1.0	0.8
H2e	14.3	7.8	7.7	7.4	12.4	13.4	13.3
H3	2.7	2.3	1.3	1.8	2.5	2.4	2.2

top-ranked alignments could be found with an rms distance below 2.7 Å. This observation can be explained by the fact that the molecules were aligned w.r.t. the atom centers and not the outward-facing surfaces. Our surface alignment algorithm, in contrast, did not have any problems with TPV.

The run times for experiments H1, H2, and H3 are given in Table 5.9. With an average of 1 second, the surface alignment algorithm is approximately 3 to 4 times slower than the atom alignment algorithm with an average of 0.3 seconds. The smaller difference in the run times compared to the thermolysin inhibitors might be due to the smaller number of iterations in the optimization step, though we did not confirm this.

Ensemble of Conformers. In the last type of experiment (IV), we again compared the experimental conformer of each HIVPI with the ensembles of the other molecules. The experimental conformers were used as reference structure, while the conformers from the ensembles served as query structures. Details about the ensembles of conformers are given in Table 5.2.

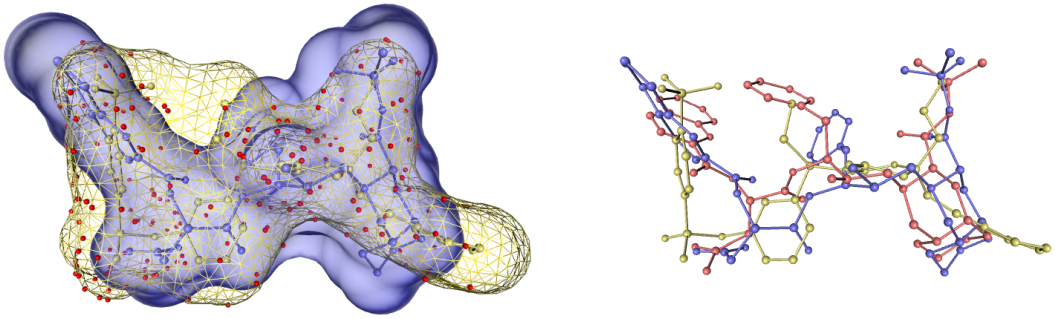


Figure 5.13: Surface alignment of experimental conformer of TPV (yellow) with an ensemble conformer of SQV (blue). This alignment has the highest score and an rms distance of 2.59 Å (cf. Table A.27). *Left:* The image shows the overlaid surfaces together with the matched surface points (red). *Right:* Comparison of aligned ensemble conformer of SQV with its experimental conformer in the experimental position (light red). Even though the rms distance is rather large the general orientation is correct.

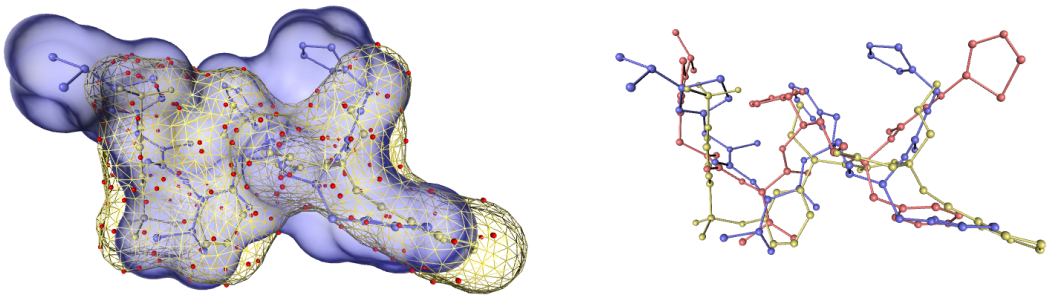


Figure 5.14: Surface alignment of experimental conformer of TPV (yellow) with an ensemble conformer of RTV (blue). This alignment was the closest alignment with an rms distance of 2.80 Å (cf. Table A.27). *Left:* The image shows the overlaid surfaces together with the matched surface points (red). *Right:* Comparison of aligned ensemble conformer of RTV with its experimental conformer in the experimental position (light red).

We compared atom alignment with surface alignment, whereas for surface alignment all properties, i.e. shape, hydrogen-bonding, and MEP, were considered. The results of surface and atom alignment are given in Tables A.27 and A.28, respectively.

In all but five of 42 cases (cf. Table A.27), a conformer and a position could be found among the 10 top-ranked pairwise alignments such that the rms distance to the experimental position was below 1.85 Å. Only in one case, the alignment failed completely. In the remaining four cases, a conformer and a position with rms distance below 3.0 Å could be found. For this size of molecules, an rms distance below 3.0 Å generally means that the correct overall position was identified, presumably with a slight shift and a disorientation of some of the terminating groups (cf. Figures 5.13 and 5.14). We can therefore constitute, that the surface alignment algorithm was well capable of identifying active conformers together with the "correct" orientation. The worst results were obtained for tipranavir, which, because of its different scaffold also has the lowest surface similarity to the other molecules.

In order to investigate if the results for tipranavir would have been better, if the experimental conformers had been included in the ensemble of conformers, we compared the scores of this experiment with the scores of experiment H2a. The results led us to the conclusion, that for all inhibitors but amprenavir an alignment with rms distance around 1.0 Å or below would have been among the 10 top-ranked alignments, if the experimental conformers had been included. This again emphasizes the importance of conformation analysis and the dependence of the alignment results on the ensemble quality.

In Figures 5.13 and 5.14 two alignments of the experimental conformer of TPV with conformers of two of the other HIVPIs are depicted. These two alignments show, that among the 10 top-ranked alignments there was at least one alignment in which the general form and orientation of the inhibitors was correctly identified.

The results presented in Table A.28 show that atom alignment failed to produce meaningful alignments for tipranavir in all but two cases, in which the general position was correctly identified. In these two cases, tipranavir acted as reference structure. Hence, the atom alignment algorithm was unable to identify an active conformer of tipranavir, no matter which HIVPI was used as reference molecule.

Metastable Molecular Conformations. As mentioned earlier, our surface alignment approach is not limited to solvent excluded surfaces but can also be applied to isodensity surfaces, such as those of conformational densities (cf. Section 2.4.3). In Figure 5.16 we show the alignment results of the conformational densities of two metastable conformations, one from tipranavir (TPV) and one from amprenavir (APV). The metastable conformations were selected from 99 (TPV) and 59 (APV) metastable conformations, respectively, computed with the program ConfJump [169]. Since we do not know the "correct" alignment, it was not possible to quantify the quality of these results. However, by visual inspection of the alignments depicted in Figure 5.16 we can judge, that both shape as well as physico-chemical properties are matched very well.

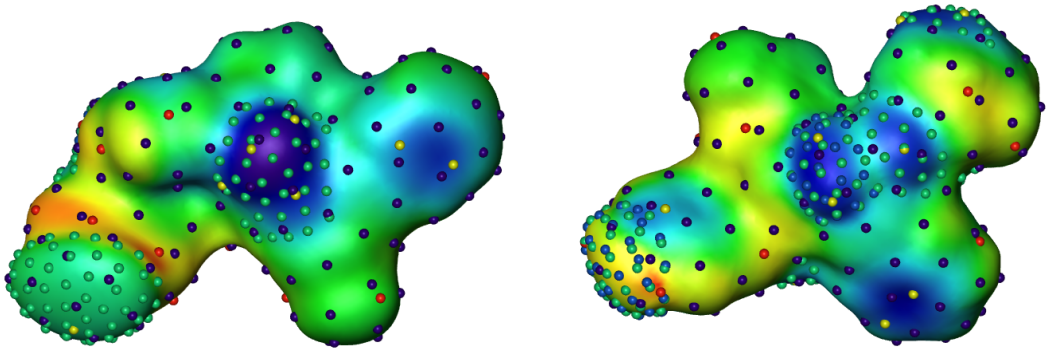


Figure 5.15: Isodensity surfaces of conformational densities, based on molecular surfaces, of two metastable conformations, one from TPV (left), the other one from APV (right). The electrostatic potential (blue/green: negative, red/yellow: positive) was averaged over all time steps belonging to the conformational ensemble of the metastable conformation. Points with distinct colors represent distinct properties: shape (dark blue), donor (light blue), acceptor (green), negative MEP (yellow), and positive MEP (red).

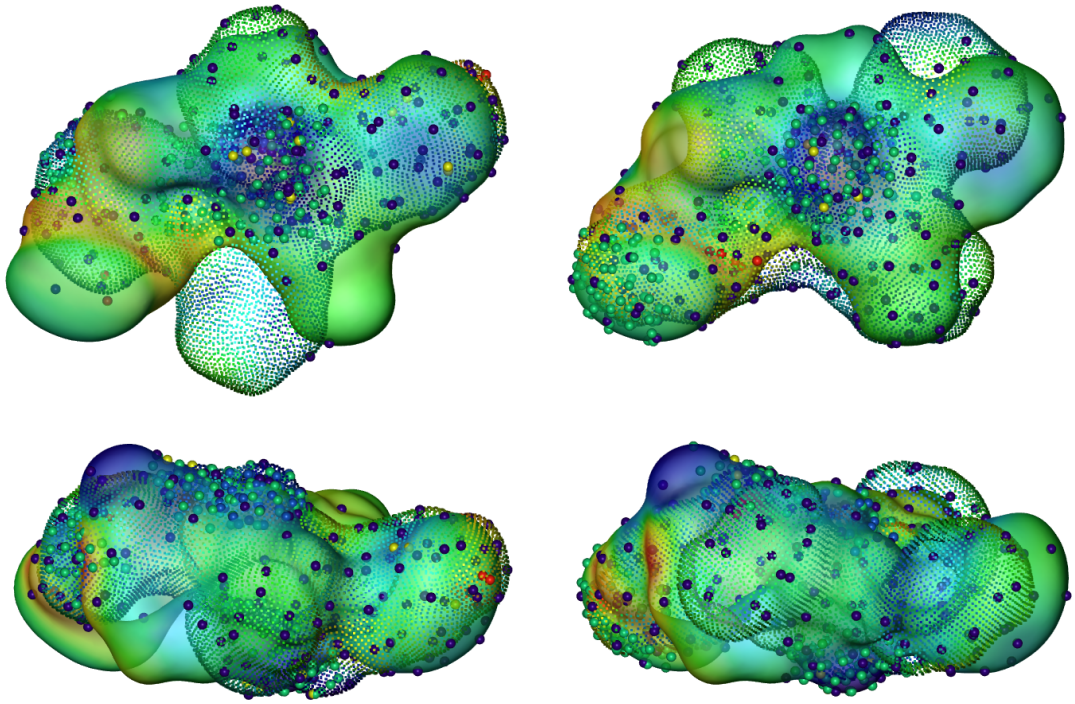


Figure 5.16: Best (left column) and second best (right column) alignments of isodensity surfaces of conformational densities represented by the surface points shown in Figure 5.15. The isodensity surface of tipranavir is shown as semi-transparent surface, while the isodensity surface of amprenavir is depicted by dots. The spheres denote the matched surface points. Colors have the same meaning as in Figure 5.15. In both alignments, shape as well as physico-chemical properties were matched well.

5.7 Summary and Conclusion

In this chapter, we presented a new approach to pairwise molecular surface alignment based on the identification of partial matchings between two molecular surfaces. The partial surface matchings are based on partial point matchings which take into account both molecular shape and physico-chemical properties. A weighting of these properties is given by the ratio of the points per property distributed on the molecular surface.

The presented algorithmic approach consists of two steps. In a first step, initial transformations are computed, each of which are optimized in the second step. The second step uses an iterative point matching scheme, which has been adjusted to the needs of surface points. A three-fold speed up was gained by introducing an efficient spatial data structure, called point distance grid.

The pairwise surface alignment algorithm was evaluated on two sets of molecules, namely a set of 8 thermolysin inhibitors, and a set of 7 HIV-1 protease inhibitors. For all of these molecules, experimental conformers were available from crystallized protein-ligand complexes. These conformers allowed a quantitative evaluation of the quality of the results. In a first series of tests, we investigated whether our surface algorithm is able to align the experimental conformers to each other similar to the "experimental alignments" given by the protein-ligand complexes. A second series of tests was performed to examine the algorithm's capability of identifying active conformers in ensembles of conformers calculated by conformation analysis programs. In addition to our surface alignment algorithm, we applied an atom alignment algorithm to the same sets of molecules. The results of all our tests are summarized below.

Selection of Properties. In order to examine the dependence of the alignment results on the choice of properties, we performed several runs of our algorithm using different properties and a combination thereof. Interestingly, for both sets of molecules the alignment results were considerably better when all properties were used, i.e. shape, hydrogen bonding, and MEP. This confirms the need to consider shape and physico-chemical properties simultaneously.

Atom-based versus Surface-based Initial Transformations. We proposed two methods for the generation of initial transformations. To investigate the dependence of the alignment results on the choice of method for initial transformation computation, we compared the results obtained with atom-based and surface-based initial transformations. The differences were small and suggest, that both methods can be used interchangeably. However, because the surface-based method is more general, since it is applicable to arbitrary surfaces without the need for an underlying "skeleton", we prefer the surface-based method. More important than the choice of method for initial transformation generation is the number of initial transformations, which can be easily modified by a few parameters. As a rule of thumb, at least a few thousand initial transformations should be used.

Alignment of Experimental Conformers. Using a large enough number of initial transformations, we obtained successful results for all experimental conformers. In most

cases the rms distance of the top-ranked alignment was below 1.0 Å. In the small number of exceptions, an rms distance below 2.0 Å could be observed, which states that at least the general position was correctly identified. For the set of thermolysin inhibitors, we were able to compare our results with the results of two other programs, namely SPAt and SurfComp. The comparison yielded, that our approach performed at least as good as SPAt and SurfComp. Also, for two inhibitors which show less similarity to the other 6 inhibitors, we achieved considerably better results than the other two programs. In addition to giving better results, our algorithm is also faster.

Identification of Active Conformers out of an Ensemble. In this test, an experimental conformer of some molecule was used as reference structure to identify an active conformer from an ensemble of conformers of a second molecule. In all but three cases, among the 10 top-ranked alignments a conformer close to the experimental one together with the correct general position could be found. Considering the large number of conformers in each ensemble, this, too, represents a very good result.

Surface Alignment versus Atom Alignment. A comparison of our surface alignment approach with an earlier atom alignment approach demonstrated that surface alignment indeed is superior to atom alignment, in particular if the scaffolds of the molecular structures are dissimilar. This could be shown for both the comparison of experimental conformers, but even more for the identification of active conformers from an ensemble of conformers. The price that has to be paid for this better quality is a 5 to 15 times increased run time.

Final Conclusion. The given results show a high quality of the presented pairwise surface alignment approach. The results indicate superiority to atom alignment as well as previously published approaches. We can therefore conclude, that our approach represents an important contribution to 3-dimensional molecular alignment.

Besides being a versatile tool in its own right, we use the pairwise surface alignment as prerequisite for the computation of multiple alignments, from which we hope to gain more information about a set of active molecules. The computation of multiple from pairwise alignments will be described in the following chapter.