3. Methods

Molecular surfaces are usually represented by triangle meshes containing up to several thousand points. The comparison task is thus to find associations between the point sets of two different surfaces where similarity or complementarity is usually defined by the spatial arrangement of one or more equally defined properties. In this process it is not sufficient to associate every point on surface A with the point on B that gives the best match but it is necessary to consider also the similarities of the corresponding neighbors of A and B. Finally the spatial arrangement of the elements of the detected associations must also be taken into account. The problem of detecting similarities between 3D point sets is well known in cheminformatics. It has been shown earlier that it is equivalent to the maximum common subgraph problem and can be solved efficiently by maximum subgraph isomorphism detection [25;91] (see section 2.6). Unfortunately this is an NP-complete problem [71] which is not a critical limitation for the comparison of small molecular structures with some dozens of atoms, but which makes it inappropriate for the large point sets of complex surface objects. Consequently, if one wants to apply this algorithm to molecular surfaces the number of points has to be reduced and additional information about the chemical and geometrical environment should be represented in a way that is appropriate to dramatically simplify the association graph. In the following sections several heuristics will be discussed that can accomplish this simplification.

The initial point set usually contains a lot of redundant information. The situation around a particular surface element is not really different compared to the environment of its neighbors. But removing all the redundant points does not solve the problem, because it would not be a smooth and accurate representation of the molecule's shape which is needed in the evaluation and refinement process. It is therefore necessary to compare molecular surfaces on at least two different levels of detail: A coarse, non-redundant, representation may be used for the detection of general features that should be matched, and the detection of the correct alignment may be done on a high resolution basis.

Recently, several publications on molecular surface comparison reported successful applications of this idea [37;56]. In particular, Cosgrove et. al. [37] reported a graph-based method that utilizes this two-level approach. On the coarse level, they described the surfaces by patches of the same shape type (convex, concave, saddle shaped, cylindrical and flat). Local geometry parameters are used to decide which patches could overlap and to form an association graph. Matches between the surfaces are then established by clique detection and confirmed by a rigid body alignment at the high resolution level of the corresponding surface points. Their program, called SPAt, gives good results in reasonable time, but they do not consider the chemical environment of points on the surface.

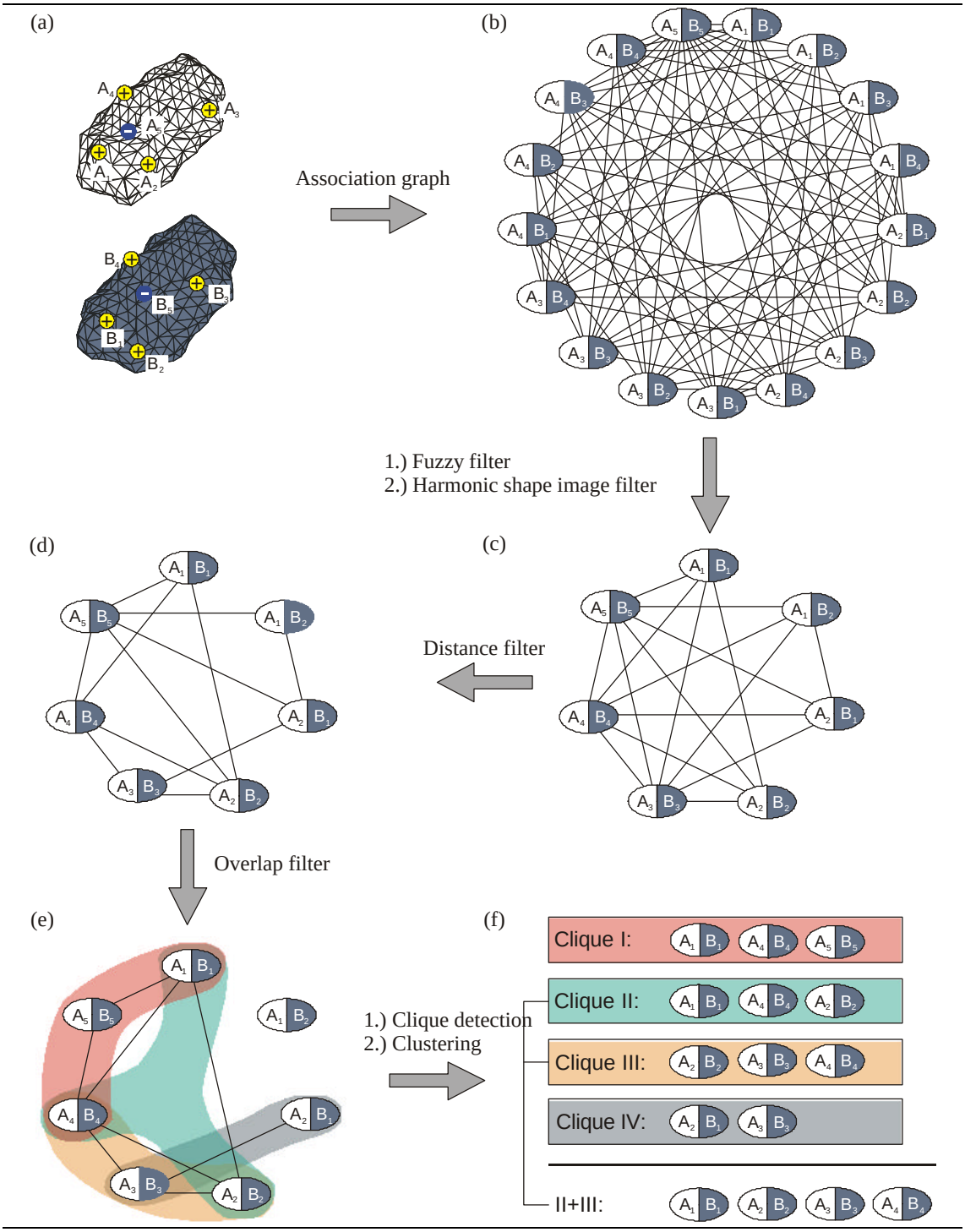


Figure 3-1: Overview of the surface similarity detection algorithm, developed in this thesis. Starting from two molecular surfaces the critical points are identified (a) and an initial association graph is built (b), which is then further simplified by the fuzzy and harmonic shape image filter (c), the distance filter (d) and the overlap filter (e). From the final association graph the cliques are detected (green, orange, blue and grey regions) and merged (clique II and III in this example) to yield the maximal surface similarity (f).

3.1. General Concept

The general approach, which has been implemented in the SURFCOMP program, is to generate a representation of the surfaces using slightly overlapping circular patces and keep track only of a set of shape critical points (*CP*, coarse level) corresponding to

process step	section[a]	points[b] A	points[b] B	nodes	edges
at the beginning	-	*1131*	*1265*	$1.47 \cdot 10^6$	$2.04 \cdot 10^{12}$
After					
- critical point detection	3.2	27	29	553	274,841
- fuzzy property filter	3.5	24	27	162	17,982
- harmonic shape image filter	3.6	18	25	63	1,260
- distance filter	3.7	18	25	63	359
- overlap filter	3.8	18	25	60	93

Table 3-1: Complexity of the association graph
The number of graph nodes and edges is given for different steps of the filtering process shown for the comparison of 1THL (A) and 4TMN (B).
[a])the section of the text where the step is described
[b])the number of distinct surface points left in the nodes of the association graph

the centers of those patches. The idea of critical points was explored by Connolly's docking algorithm [35] which was later improved by Lin et. al. [86] and Wang [133]. It reduces the number of possible point pairs and associations by several orders of magnitude, so that it is possible to build an initial association graph. This graph is further simplified by several filters that compare the physicochemical properties, surrounding shape and local arrangement of the critical points on both surfaces. (Table 3-1 illustrates the complexity of the association graph at the initial stage and after every step of the algorithm.) In the final graph the similarities are then retrieved by clique detection and rigid body alignments are produced on a point-based (high resolution) level for every match. (see Figure 3-1)

For efficiency reasons SURFCOMP emphasizes the simplification of the association graph which results in a set of smaller cliques that represent only local surface similarities. Therefore, to get a picture of the total similarity between two surfaces, the cliques must be combined to reproduce the complete, global match. For that a hierarchical clustering was used to finally combine those cliques that represent the same geometrical transformation of one molecule onto the other. The final result can be a long list of possible alignments. To provide a faster access to the most promising matches a ranking mechanism was developed and several scoring functions were implemented to sort the results by significance. The alignments can be scored by their size, the root mean square deviation, and the correlations of property values on corresponding points. A ranking is then established by a consensus scoring similar to the methods used in molecular docking.

In a multi step filtering protocol, such as SURFCOMP, several heuristics are used to separate the significant from the insignificant similarities or complementarities. These heuristics are usually controlled by a set of parameters which demand a lot of experience and patience to be tuned properly. The SURFCOMP method needs 7 filtering parameters not including the variables that are involved in the generation of the surfaces and the surface patches. Some of these parameters have proper default values that can be applied to most of the problems, but some other values can have a big influence on the outcome of the calculations. This is not an unsolvable problem and multi step filtering procedures can produce good results (including this thesis), but it should be mentioned that there are alternative methods that have the potential to avoid at least some of these heuristics.

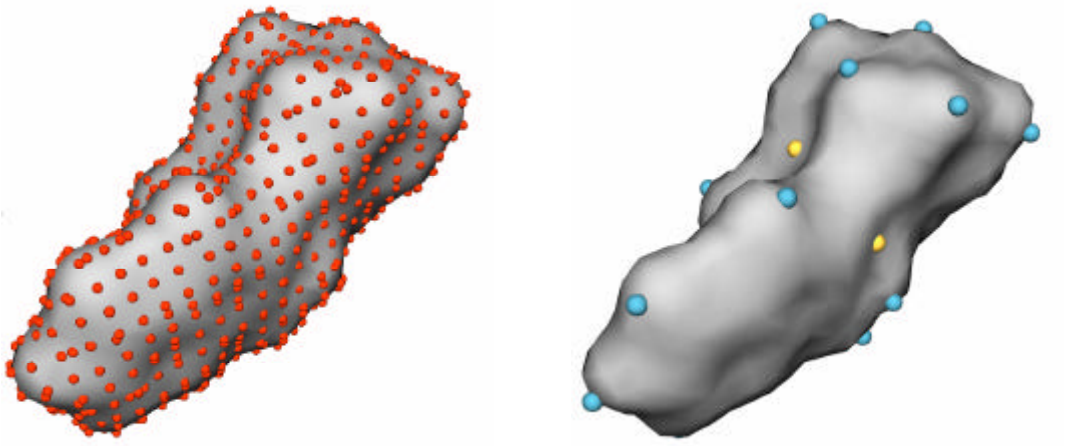
The first heuristic introduced into 3D object comparison is the selection of the coarse and fine representation of the surface. The use of shape critical points is a well understood technique which has been applied to many similarity- and docking-problems but it introduces an arbitrary resolution level that cannot be varied easily. Multi resolution analysis is a computer vision technique that allows the free specification of an object's resolution by means of regular triangulation meshes and wavelet decomposition [45;97]. This approach allows the automatic adjustment of an object's resolution for visualization or storage purposes. The resolution can always be increased or decreased if more or less wavelet coefficients are used. This technique or a similar approach could be used to perform a comparison between two molecular surfaces at a very low resolution, identify the best matching areas, increase the resolution and continue until the match cannot be improved anymore. The iterative procedure would reduce the number of heuristics to the absolute minimum and it would provide an automatic convergence criterion.

The remaining part of this section describes each stage of the method in detail and the steps that are necessary to prepare the input data and evaluate the output. The implementation aspects for each step are given. For the theoretical and algorithmic details the reader is referred to Chapter 2 (Theory). At the end of this chapter a description is given how the application of this method can be extended to the comparison of protein surfaces.

3.2. Definition of Critical Points

The shape of a surface is mainly determined by the location of convex, concave and saddle shaped features. If two objects match, the features of their surfaces have to be similar and should be aligned by the same rigid body transformation. A single feature is usually formed by many surface points which have similar curvature patterns. Hence it is reasonable to take the feature level as the low and the point level as the high resolution for the comparison process. To get an appropriate representation of the surface features a subset of so called shape critical points is extracted from the initial complete set of points. Shape critical points are characteristic points where the properties that define the feature are a maximum.

Shape features can be identified by the signs of the canonical curvatures (cc_1, cc_2; p.



(a) (b)

Figure 3-2: Surface of the thermolysin inhibitor L-valyl-L-tryptophan
(a) all points, (b) distribution of peak (blue) and valley (yellow) critical points over the surface (PDB entry 3TMN).

12): convex regions have two negative, concave two positive and saddle shaped ones display one positive and one negative curvature. Hence two classes of critical points can be defined: A point **p** is a *peak*, if it is a convex point with maximum negative curvature and a *valley*, if it is a concave point with maximum positive curvature in a certain neighborhood N(**p**, r_{cp}) defined by the on-surface radius r_{cp}. This corresponds to a "dip" or "cleft" on the surface (eq. 3-1). To keep the initial set of critical points as small as possible do not consider saddle points are not considered.

$$\begin{aligned} \mathbf{p} &:= peak \quad if\ \forall \mathbf{q} \in \mathrm{N}(\mathbf{p}, r_{cp}) \left| |cc_{1\mathbf{p}}| > |cc_{1\mathbf{q}}| \right. \\ \mathbf{p} &:= valley \quad if\ \forall \mathbf{q} \in \mathrm{N}(\mathbf{p}, r_{cp}) \left| |cc_{1\mathbf{p}}| < |cc_{1\mathbf{q}}| \right. \end{aligned} \qquad \text{eq. 3-1}$$

At the beginning of a comparison process, before the initial association graph is formed, the peaks and valleys of both surfaces are determinated. The *CP* algorithm investigates every convex or concave point on the surface and adds it to the *peaks* or *valleys* if it meets the appropriate criteria. Figure 3-2b shows the peak and valley critical points of a thermolysin inhibitor molecule. It can be seen that there are many more convex than concave *CP*s. This is due to the fact that most "valleys" are not concave but saddle shaped regions.

3.3. The Association Graph

An initial version of the association graph is formed from the critical points of both surfaces. In this graph the **vertices** correspond to pairs of critical points, $pp_{ij} = (CP_{iA}, CPj_B)$, from the two surfaces that are compared. Hence all the convex and concave critical points of the first surface are paired with the convex and concave *CP*s of the second surface to form the initial set of vertices. This means that at this stage all the critical points with the same curvature attribute are considered similar.

According to the definition of an association graph, **edges** should be drawn between every two pairs that do not have a critical point in common (see Figure 3-1b), but for computational reasons no edges are considered before the application of the distance filter described below.

3.4. Generation of Surface Patches

Since SURFCOMP's coarse level representation of molecular surfaces is the set of their critical points together with their neighborhoods, it is necessary to have a consistent definition of the vicinity of a surface point also known as the *patch*. A patch is a continuous piece of the surface centered on a point **c** that includes all points around that center within a certain *on-surface* distance, henceforth called the patch radius. (An *on-surface* distance of two points is the shortest path between them over the surface, not straight through 3D space).

In this work patches, like surfaces, are represented by triangulated meshes (p. 9). But unlike surfaces which are completely closed objects with no boundaries, a patch has a border that should be defined by an unambiguous sequence of triangle edges. For the harmonic image construction (p. 17) all the points on the border must be passed exactly once before the point where the walk was started is reached again. In such a traversal the iteration from one border point to the other is controlled by the counter clockwise order of the triangle points: In all triangles that contain the active point its successors are examined to find a border point that has not yet been reached. To make this mechanism work, every triangle must not have more than one border edge (because then there would be more than one unvisited border point among the successors, and the

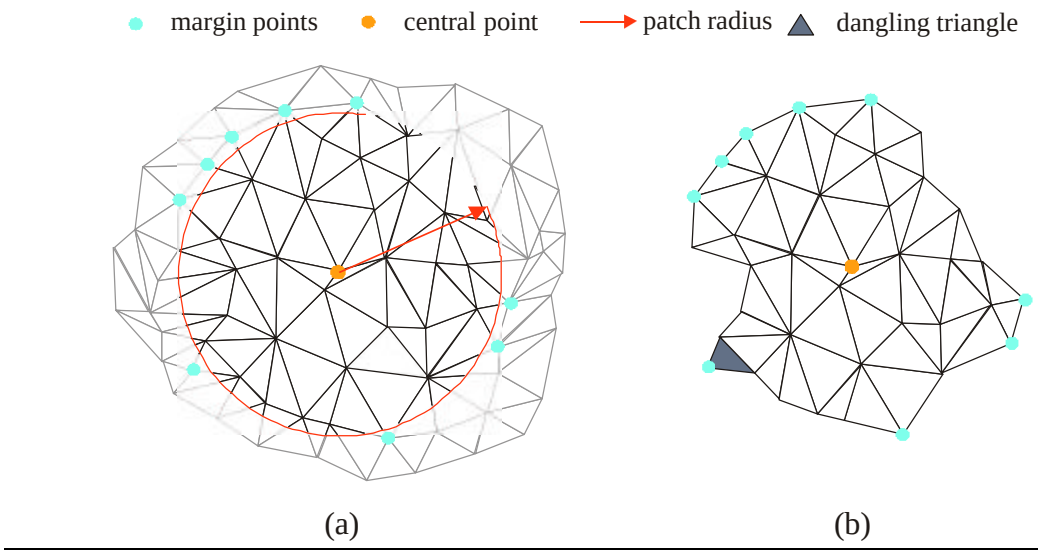


Figure 3-3: Illustration of the patch generation process.
In the first steps the points within and in close contact to the patch radius are selected (a) and extracted from the surface. There may be dangling triangles (b) which have to be removed in the refinement to ensure consistency.

walk could end up in a loop, a shortcut or a dead end). Therefore triangles with two or three edges exposed to the border (dangling triangles) should be removed.

Holes in surface patches present another problem. They are formed if a molecular surface contains “pillar-like” cylindrical areas in the close vicinity to the central point. In this case, only the base of the pillar lies within the patch radius and the upper part or the head is not included. Automatically including all the points in the hole is not appropriate, because if the pillar is a bridge to the rest of the surface, all the other points would be included and the patch would be equal to the complete surface. Fortunately the harmonic shape images are robust with respect to holes and missing parts in patches [144], therefore it was decided not to fill them.

According to the considerations above, the patch around a central point **c** is created as follows (see also Figure 3-3):

1. A subset of the surface points which have an Euclidean distance to **c** that is less than the patch radius is preselected to avoid on-surface distance calculation on the complete surface.
2. The on-surface distances between **c** and all the points in the subset are calculated by the Dijkstra shortest path algorithm [41] with the edge-weights set to the Euclidean distances between neighboring points.
3. All points around **c** within the patch radius are extracted plus any points connected to them that lie within a 5% margin off this radius.
4. Every triangle on the surface that contains three selected points is copied to the patch.
5. To preserve a correct clockwise or counter clockwise walk, all triangles in the patch with more than one border edge (dangling triangles) are removed.
6. If there are any points that do not belong to a triangle, they are removed and step 5. is repeated until consistency is achieved.
7. For each remaining point a reference to the original point in the surface is stored.

3.5. Fuzzy Filter

In order to reduce the complexity of the problem, it is necessary to remove those critical point pairs from the association graph that do not have a similar chemical environment. Each vertex of the graph must thus be checked by a chemical filter to ensure that the corresponding critical points have similar chemical properties. Fuzzy sets and linguistic variables [142] were used to express the similarity between chemical properties mapped onto the surface, and applied a defuzzification function, introduced by Exner et. al. [48] as a similarity measure (p. 15).

According to Figure 2-5 five fuzzy sets were defined for each physicochemical property in the experiments that correspond to common classifications (Table 3-2). An important issue in the application of that technique is the scaling of the compared properties. For every possible value of the property the contribution to each of the predefined fuzzy sets must be specified in advance (see also eq. 2-18 on p. 16). But many quantities that can be mapped onto the molecular surface vary too much to apply fixed boundaries and relations between these sets. Especially the electrostatic potential depends strongly on the total charge of the molecule and consequently it is meaningless to compare surface ESP patterns directly between molecules with different total net charges unless an appropriate normalization is carried out. For instance the absolute difference between the ESP around the adenosine 3-H in ATP^{-4} and ATP^{-3} is about 50 kcal/mol while the relative difference between the ESP over the center of the adenosine 6-ring and the 3-H in the 3 and 4 minus species is only 10 kcal/mol. For a general interpretation of the membership functions it will therefore be necessary to use normalized (mean-centered or autoscaled) surface properties. In the fuzzy filtering autoscaled functions were used. The rationale behind this is that in surface similarity searches, the aim is to find the region on one surface that fits *most likely* to a patch on the other one. Hence the most positive or negative values will fit best to each other regardless of their absolute difference. Furthermore an autoscaled property provides natural classifications for the membership functions.

property	high –	–	neutral	+	high +
ESP	highly negative	negative	neutral	Positive	highly positive
LP	highly hydrophilic	hydrophilic	amphiphilic	Hydrophobic	highly hydrophobic

Table 3-2: Definition of fuzzy qualitative classes
These classes are used as fuzzy sets in the linguistic variables of the fuzzy filter.

The fuzzy filter is the first filtering step in the surface comparison process. It takes every vertex of the initial association graph, and calculates the fuzzy dissimilarity function for a certain chemical property of its points according to eq. 2-19 (p. 16). Every vertex whose points are more dissimilar than a certain fuzzy threshold F is then removed from the graph and its points are considered to be chemically dissimilar. By this filter, the number of the associations can be reduced by approximately 80% (see also Table 3-1).

3.6. Harmonic Shape Image Filter

The fuzzy chemical filtering checks for similar physicochemical properties between both surfaces, but it does not consider the shape of the molecules around the critical points. It is important, however, to consider the surface-patches around the critical

points and compare them with each other to establish whether two *CP*s are embedded in similar regions and how their neighborhoods are best oriented relative to each other. In the present surface comparison process harmonic shape images (HSI) [145] (p. 17) provide the methodology to compare the patches and to define a relative orientation between them.

Harmonic shape images compare surface patches by a local shape descriptor mapped onto their points. Several such descriptors were introduced in section 2.2.3. In the present work the surface topology index (STI) [23] was used to compare the shape of two surface patches, but any other scalar shape descriptor could be applied as well. While in general possible, multiple scalar values, such as the canonical curvatures, are not used in the HSI comparison because the Pearson correlation function is susceptible to leverage effects. Such effects can be easily introduced, if two variables do not occupy the same space or if the dissimilarities of one type neutralize the similarities of the other or vice versa.

HSI generation. The transformation of a surface patch into a harmonic shape image is a multi-step process that involves (i) the detection and mapping of the patch border, (ii) the correct mapping of the patch's interior points, and (iii) the sampling of the shape descriptors from the point-based mesh to a regular grid (see also p. 17). Especially the sampling step is computationally intensive and special techniques must be applied to improve its speed.

The detection of the patch's border is done by the patch generation algorithm (section 3.4) when removing the dangling triangles. For the mapping of the border points a continuous, counterclockwise walk along this border will provide us with the correct sequence for the determination of the position angles θ_i on the unit circle (eq. 2-20). The traversal is done by following the triangle edges from one border point to its successor at the boundary. The positions of the border points are obtained in polar form with the radius r=1 and the angles θ_i and must be transformed into Cartesian coordinates for the interior mapping (p. 17).

After the position of the border is fixed, the interior points can be mapped into the unit disk. According to section 2.5.3, these positions are defined by a pair of systems of linear equations (eq. 2-28). The matrices **A** and $\mathbf{b_x}\mathbf{b_y}$ must be assembled according to eq. 2-29 and eq. 2-30 with the spring constants k_{ij} set to

$$\frac{1}{\left\|\mathbf{p}_i - \mathbf{p}_j\right\|} \qquad \text{eq. 3-2}$$

and the actual positions of the border points on the unit circle (p. 17). The fact that the systems use the same coefficient matrix **A** and different constrain vectors $\mathbf{b_x}$ and $\mathbf{b_y}$ for the x and y position, makes it suitable to solve the equations by LU-decomposition [108].

The last step is the generation of the shape image by resampling of the descriptors from the mapped points to a regular grid. To perform the sampling it is necessary to identify the triangle beneath every position at the grid. The actual value for the grid point is then calculated by an interpolation of the triangle's vertices (p. 20, eq. 2-32). The search for the active triangles is of order $O(N^3)$ where N is the number of grid points which is equal to the number of points in the patch. Hence the resampling is the rate determining step of the HSI generation. The process can be accelerated if a geometric hashing algorithm is used to investigate only the triangles in the vicinity of a grid point. To this end the whole image is divided into fields in a way that each field

contains approximately 5 triangles. All triangles are assigned to those fields, where at least a part of the triangle is present. Later, during the resampling, only those triangles of the field of the current grid point are considered.

Image comparison. As discussed in section 2.5.4 the images can be rotated against each other. It is thus necessary to perform a full angular scan when determining the similarity between two harmonic shape images. Usually one of the images is fixed, and the other one is rotated by a predefined angular increment δ (usually 2°) and resampled for each rotated position around the circle. To avoid unnecessary sampling the grids for the flexible image are precompiled for every possible angular position and persistently stored with the harmonic map data.

The harmonic shape image filter is invoked for every critical point pair in the association graph that is left after the fuzzy filtering step. In this filter step the patches around the critical points are computed, if they are not yet available, and transformed into the corresponding harmonic images. Then the two images of the *CPs* of the associated point pair are compared as described above and on p. 21. If the detected similarity is better than the shape threshold *R* the point pair passes the filter otherwise it is removed from the association graph.

3.7. Distance Filter

Up to this point only single pairs of critical points (*pp*) have been considered which are represented by the vertices of the association graph. However, the aim is to find groups of *CP* pairs which represent a similarity between the compared surfaces. Thus it is necessary to form edges between the point pairs in the association graph, to identify those which can overlap at the same time.

A simple but effective criterion is the difference of the distances of two point pairs on surface A and B. Considering two point pairs $pp_1 = (CP_{A1}, CP_{B1})$ and $pp_2 = (CP_{A2}, CP_{B2})$ with the positions of their critical points $\mathbf{p}_{A1}$, $\mathbf{p}_{B1}$ and $\mathbf{p}_{A2}$, $\mathbf{p}_{B2}$, the distances d_A and d_B are

$$d_A = \|\mathbf{p}_{A1} - \mathbf{p}_{A2}\| \quad \text{(A)}$$
$$d_B = \|\mathbf{p}_{B1} - \mathbf{p}_{B2}\| \quad \text{(B)} \qquad \text{eq. 3-3}$$

the Euclidean distances between the two critical points on surfaces A and B (see also Figure 3-4). Martin et. al. used the same criterion for the identification of pharmacophore patterns [91].

Two pairs are connected in the association graph only if the distances d_A and d_B are within a certain distance tolerance $t \geq |d_A - d_B|$ and d_A, d_B are larger than the minimum

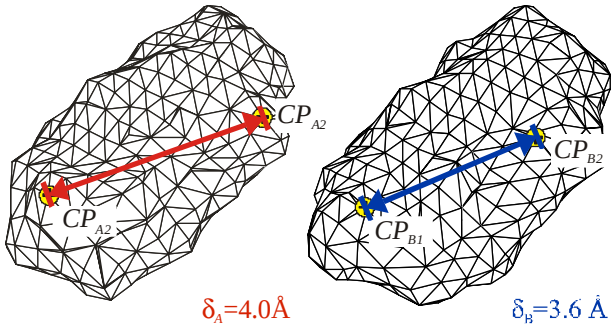


Figure 3-4: Distance filter

distance d_{min}. The minimum distance is introduced to avoid connections between very close critical point pairs which represent essentially the same regions. It should also be noticed, that no connections must be drawn between critical point pairs that share the same point on either of the two surfaces (i.e. $CP_{A1} \neq CP_{A2}$ and $CP_{B1} \neq CP_{B2}$).

3.8. Overlap Filter

The distance filter checks if two pairs are at an appropriate distance for simultaneous overlap, but harmonic image matching provides additional information about the optimal orientation of each *CP* patch pair. Using this information the number of connections in the association graph can be further reduced.

The idea is to check the simultaneous overlap of both pairs via the relative orientations of the connecting axes on surface *A* and *B*. In Figure 3-5 the axes between the two critical points on each surface are projected onto the harmonic maps of the patches and the closest points on the borders of the patches are determined. a_1, a_2 and b_1, b_2 denote the angles between the optimum orientation (alignment axis) and the closest points to the *CP* axes on surface *A* and *B* respectively. The a and b angles thus describe the heading from one critical point patch to the other with respect to the alignment axis.

The filter computes the heading differences j_1, j_2 for both *CP* patch pairs and removes the connection between them, if none of them is within a certain angular tolerance j_{tol}:

$$j_1 = |b_1 - a_1|$$
$$j_2 = |b_2 - a_2| \qquad \text{eq. 3-4}$$

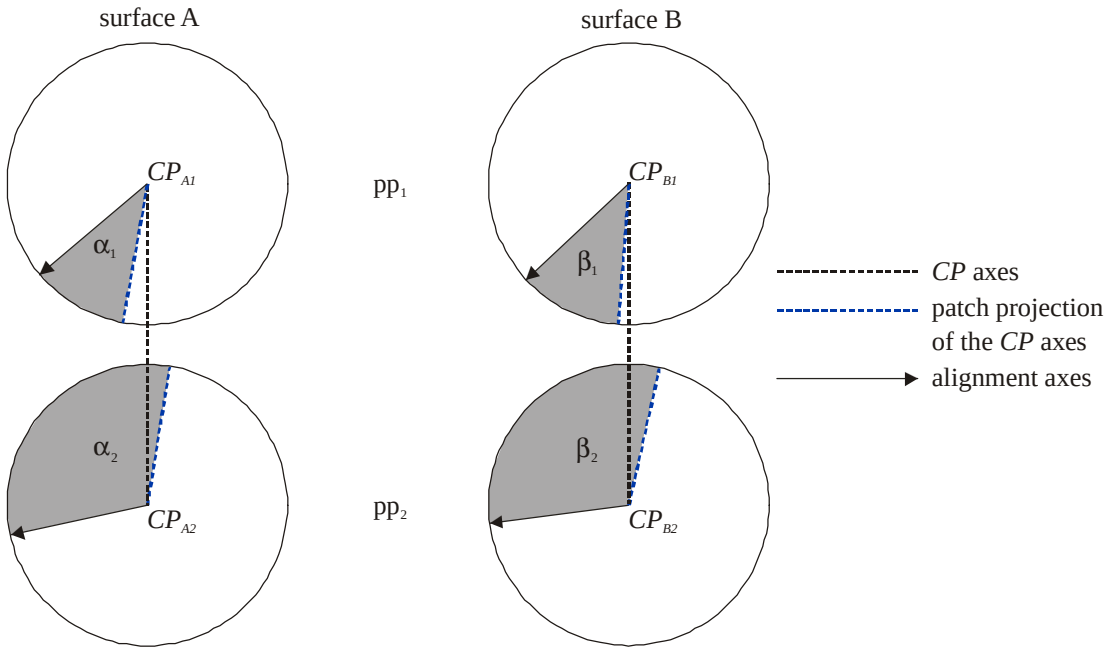


Figure 3-5: Illustration of the overlap filter.
The axes between the two patches on both surfaces (black stippled lines) are projected onto the harmonic map of the surface patch, and the angles between that projections and the axes that define "north" (0°) in the optimal alignment of the patchpairs pp_1 and pp_2 are determined as the bearing from one patch to the other patch on the same surface.

3.9. Clique Detection and Clustering

Having applied all the filters, the size of the association graph is reduced so that it is possible to search for cliques in it. The algorithm of Bron and Kerbosch [26] was used to find all cliques which are present in the association graph. This usually results in a large set of cliques consisting of two to four critical point pairs. They only represent partial similarities and must be combined to get the largest possible local surface alignment between the two molecules. Therefore, these small primary cliques are combined into larger clusters that represent different sets of corresponding points on both surfaces.

For each cluster a rigid body transformation was generated based on all the correspondences detected by the harmonic shape image matching for the patches around the critical points. The transformation matrices **T** are calculated by a least squares fit [93] of the two point sets superimposed over their centers of gravity. The root mean square deviation (RMSD) of this transformation serves as a quality criterion for the cluster:

$$RMSD = \sqrt{\frac{\sum_{i=1}^{N} \left\| \mathbf{T} \cdot \mathbf{q}_i - \mathbf{p}_i \right\|^2}{N}} \qquad \text{eq. 3-5}$$

where $\mathbf{p}_i$ is one of the N fixed and $\mathbf{q}_i$ is one of the N transformed corresponding points. From the large set of initial small clusters, those with high RMSD values are eliminated (above 4.0 Å) and the remaining clusters are subject to a stepwise hierarchical-linkage clustering as follows.

For all pairs of clusters in the list that can be combined, the RMS deviations for the transformation of cluster A with the transformation matrix of cluster B and vice versa are calculated; the smaller value (single linkage) is stored as the distance between A and B. Two clusters A, B cannot be combined if a critical point is paired with a different *CP* in A and B. At each step the algorithm takes the two closest clusters and merges them into a new one while updating the distances to the remaining clusters. The new one replaces the merged clusters in the list and the algorithm is repeated until no more clusters can be merged. The result is a set of possible local surface alignments.

Besides single linkage complete and average linkage were examined too, but they did not cause any differences in the quality of the results. Because single-linkage can be implemented more efficiently than complete and average linkage, it was used in all experiments.

3.10. Scoring and Ranking

The hierarchical clustering provides the results of a surface comparison as a tree, where the largest alignments are found in the elements closest to the root and the original cliques are placed in the leaves. This representation can be very useful when one wants to examine how the larger alignments are composed and how strong the different elements of the clusters are related to each other. However, in most of the cases the primary question is which alignment is the best in respect to percentage of the covered surface, quality of the rigid body fit and chemical similarity. Especially when the molecules are large and the comparison produces a number of possible top-level alignments, this task is difficult to do by visual inspection of the alignments even though the pure RMSD value of the rigid body fit provides a good initial guess of the quality of the clusters.

Another problem lies in the nature of the hierarchical clustering: The algorithm subsequently combines two clusters into a new one either until only one cluster is left or the new one cannot be combined with another one because of ambiguous critical point combinations. Thus the top-level clusters often represent poor alignments if two clusters that practically do not fit together are combined because all their *CP* pairs are correct. The consequence is that the best alignments are often placed in the levels beneath the top. It will therefore be necessary to find an alternative ranking that will sort out the promising alignments by a combination of patch-size, geometrical and chemical similarity criteria.

Ranking is a well known problem in molecular docking, where the large lists of possible ligand/receptor conformations must be scored and sorted to simplify and speed up the manual search for the best structure. The usual strategy is to improve the energy score, which is by far the most important criterion, with several other heuristics [29]. Wang and Wang [135] identified three different classes of consensus scoring methodologies: *"rank-by-number"* (eq. 3-6) uses the average scoring value, *"rank-by-rank"* (eq. 3-7) takes the average rank and *"rank-by-vote"* (eq. 3-8) counts how often an entry is sorted into the top x% by each scoring function:

$$r_i = \frac{1}{N}\sum_{j=1}^{N} \mathrm{SF_j}(x_i) \qquad \text{eq. 3-6}$$

$$r_i = \frac{1}{N}\sum_{j=1}^{N} \mathrm{rank}(\mathrm{SF_j}(x_i)) \qquad \text{eq. 3-7}$$

$$r_i = \sum_{j=1}^{N} \mathrm{top}(n, \mathrm{SF_j}(x_i)) \qquad \text{eq. 3-8}$$

where r_i is the rank of the docking result x_i, $\mathrm{SF_j}$ is the j^{th} scoring function. $\mathrm{rank}(\mathrm{SF_j}(x_i))$ returns the rank of x_i and $\mathrm{top}(n, \mathrm{SF_j}(x_i))$ yields true if x_i is among the top n% according to $\mathrm{SF_j}$ or false otherwise.

A consensus scoring algorithm was implemented in SURFCOMP based on the *rank-by-rank* scheme. The algorithm calculates the average ranks determined by (a) the RMSD value of the rigid body fit, (b) the number of corresponding surface points that build the alignment and (c) the chemical correlation of these points. Thereafter it sorts the results in a way that places the most promising clusters at the top of the list. All clusters are evaluated and ranked independently of their position in the hierarchy.

ad a. The ***RMSD*** value is provided by the clustering algorithm eq. 3-5. The clusters are ranked in ascending order because the quality of a rigid body transformation is inversely proportional to the RMSD value.

ad b. The number of corresponding points ($\boldsymbol{N_{points}}$) reflects the size of the detected surface similarity. The larger the common surface area the better the cluster is ranked by this quantity. It is somehow complementary to the RMSD because larger point sets are more likely to produce larger RMSD values so combining the RMSD and size of the similarity reflects a kind of trade off between accuracy and size.

ad c. RMSD and the number of corresponding points are responsible for the evaluation of the geometric fit. Besides that a check of the chemical similarity should not be neglected. This can be easily performed by the calculation of a Pearson correlation coefficient $\boldsymbol{R_{chem}}$ (eq. 2-35 on p. 21) between the physicochemical properties

of the corresponding points. For the ranking all the clusters are sorted by descending order.

The final ranking value is the weighted average rank over all terms (eq. 3-9) and a sort in ascending order will bring the must promising clusters to the top of the list. Usually the three contributions are weighted equally and the weights are set to 1. Once a good cluster is identified in that list its hierarchical position can be examined to check whether its parent or one of its children may provide a better representation of that particular surface similarity.

$$consensusrank = \frac{1}{3}\left[\text{rank}(RMSD) + \text{rank}(N_{points}) + \text{rank}(R_{chem})\right] \qquad \text{eq. 3-9}$$

3.11. Treatment of Protein Surfaces

The first goal of this project was to establish a surface comparison algorithm for low molecular weight compounds. However, in the course of the studies the ability of the program to compare surfaces of large biomolecules such as proteins was investigated too. The main problem with large molecules is that even at a lower level of resolution the number of points is approximately one order of magnitude greater than for small compounds. Because most of the algorithms in the comparison process scale approximately quadratically with the number of surface points, this implies a massive increase in computational time, memory and the number of candidate alignments that have to be evaluated.

In molecular modeling and drug discovery the functionally most interesting part of a protein surface is where a substrate is converted catalytically (active sites of enzymes), a cofactor is bound or a signal molecule is recognized. Fortunately, such functional sites

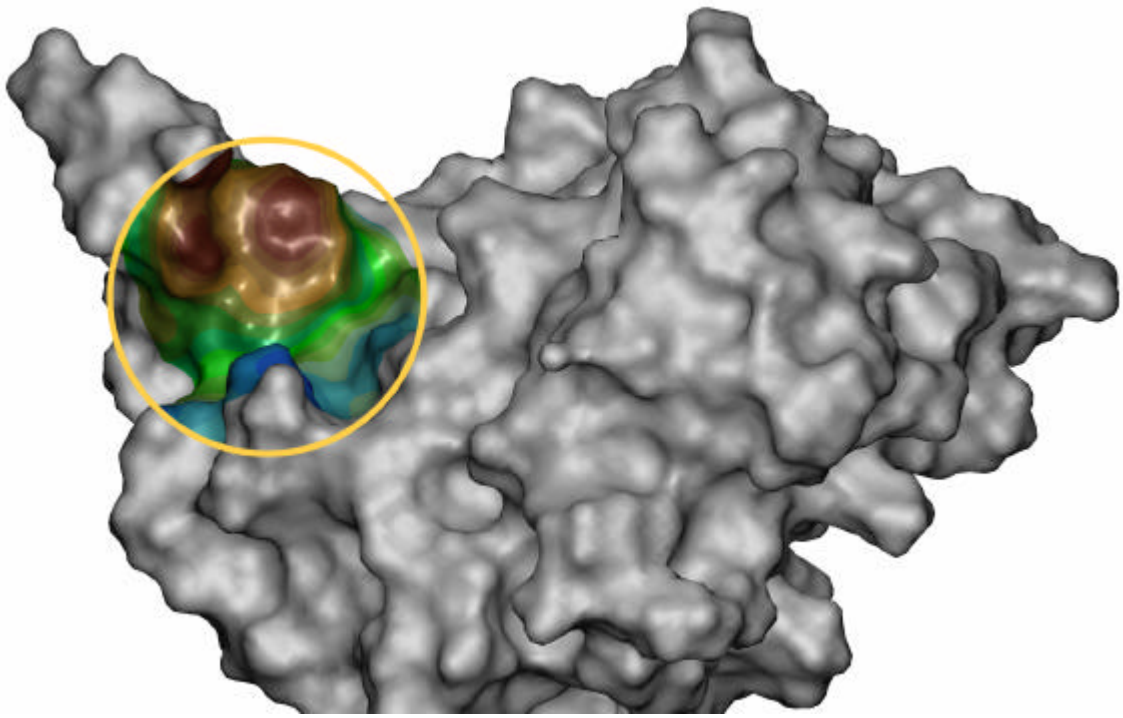


Figure 3-6: Surface of a bacterial ABC transporter protein (PDB: 1L2T).
The protein binds an ATP molecule in the pocket on the upper left corner. A site-sphere (represented by the yellow circle) was defined and the surface within this sphere (colored by the lipophilic potential) was used for the surface comparison.

usually cover only a small fraction of the total protein surface. Therefore the comparison of two proteins can be reduced in many cases to the comparison of their binding sites. This will allow the investigation of the relevant parts of the proteins' surfaces at the same resolution as low molecular weight compounds. A potential drawback of this approach is that information about the location of the functional site is needed, but that is usually available together with the 3D structure of the protein.

There are several ways to select the region of interest around a functional site. SURFCOMP applies one of the most popular strategies. A spherical region is defined such that it encompasses all amino acid residues that are known to be part of the site. Note that in the case of elongated binding pockets other choices (such as a union of overlapping spheres) would be possible. To restrict the surface comparison to the area around a site the initial association graph is built only from those critical points, which are included in the site-spheres. The rest of the process is performed as described above.

ESP Calculation. For the calculation of the electrostatic potential on the surface of a protein (see also section 2.2.1 on p. 11) it is usually not practical to use *ab initio* or semi-empirical calculations due to the large size of the systems. One viable compromise is to assign point charges derived from protein force fields. In the present investigations the charges of the AMBER force field [36] were used.

3.12. Implementation Details

The following section gives a general overview about the programming techniques, software and libraries that have been used for the various surface comparison experiments. If any experiment required different or additional tools it is described in the corresponding section of chapter 4, "Computations and Results".

3.12.1. Software

The complete surface comparison process, as described in the sections above, was implemented in the computer program SURFCOMP. The main program and all necessary libraries were written in C++ and binaries were compiled for Linux with the GNU compiler suite [53]. All matrix and vector manipulations have been coded with the RazorBack 2.0 library [8] and all graph operations were implemented using the Boost graph library [123].

The molecular surfaces were calculated by the MOLCAD module [24] in Sybyl 6.9 [2] or alternatively the molecular surface program MSMS by Michael Sanner [114;115]. All the surface properties were calculated by the MOLCAD module. For the electrostatic potential appropriate atomic point charges were either calculated at a semi-empirical level with MOPAC [40] or at the Hartree-Fock level with JAGUAR [119].

For the evaluation of the results a plug-in for the Geomview [1] software was developed that allows a real-time 3D visualization of the surface alignments, scoring and browsing of the ranking and preparation of several output formats for publishing the results. Pictures of the surfaces and surface alignment were prepared and computed using the rendering software POV-Ray [3].

3.12.2. Hardware and Computation

The actual surface comparisons were performed on a Linux cluster consisting of 22 nodes with two 2.4 GHz Intel Xeon processors and 2 GB of RAM. The generation of the surfaces and the calculation of semi-empirical atomic point charges were carried out on a four CPU SGI Origin 200 server with 4GB of RAM running under IRIX 6.5. HF

calculations and the evaluation of the results were executed on a Linux workstation with two 1.0 GHz Intel Pentium III processors with 512 MB of RAM.

Depending of the size of the problem a single surface comparison usually takes from about 75 s for low molecular weight compounds up to 2 hours for the comparison of large protein active sites. The calculation of the surfaces and their properties can take from 10 up to 120 seconds except for the calculation of atomic point charges which depend heavily on the computational level (HF, semi-empiric or force-field charges).