

Defining, Computing, and Visualizing Molecular Interfaces

Amitabh Varshney*

Frederick P. Brooks, Jr.[†]

David C. Richardson[‡]

William V. Wright[†]

Dinesh Manocha[†]

*Department of Computer Science
State University of New York at Stony Brook
Stony Brook, NY 11794-4400

[†]Department of Computer Science
University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-3175

[‡]Department of Biochemistry
Duke University Medical Center
Durham, NC 27710

Abstract

A parallel, analytic approach for defining and computing the inter- and intra-molecular interfaces in three dimensions is described. The molecular interface surfaces are derived from approximations to the power-diagrams over the participating molecular units. For a given molecular interface our approach can generate a family of interface surfaces parametrized by α and β , where α is the radius of the solvent molecule (also known as the probe-radius) and β is the interface radius that defines the size of the molecular interface. Molecular interface surfaces provide biochemists with a powerful tool to study surface complementarity and to efficiently characterize the interactions during a protein-substrate docking. The complexity of our algorithm for molecular environments is $O(nk \log^2 k)$, where n is the number of atoms in the participating molecular units and k is the average number of neighboring atoms – a constant, given α and β .

1 Introduction

The driving problem that has motivated the research presented in this paper is the protein-substrate docking problem. This problem arises in the study of structure and interactions of proteins, particularly in rational drug design. For a better understanding of the fundamentals of protein structure the interested reader can see the book by Dickerson and Geis [4].

The *protein-substrate docking problem* is to determine the position and orientation of a protein molecule with respect to a given substrate (another molecule that may be a protein, a nucleic acid, or a drug

molecule) such that the energy of interaction of the two is minimized. This problem is useful in studying enzyme catalysis and antigen-antibody interactions. These interactions have been observed to be very specific in their occurrence. Even slight changes in the structure of a sidechain of one of the participants have been observed to inhibit such interactions.

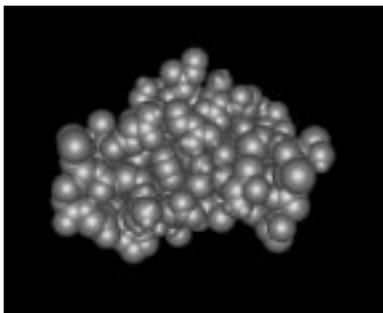
This docking of the protein with a substrate is characterized by geometric and electrostatic complementarity of the two surfaces and compatible hydrophilicity. Determination of the molecular surfaces at the interface of the two molecules thus plays a rather important role in solving this problem.

1.1 Molecular surfaces

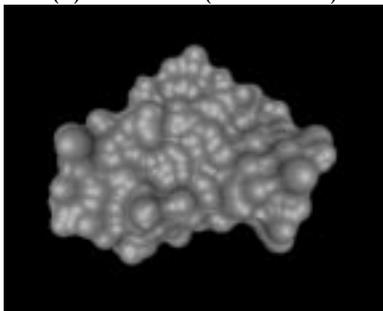
The *smooth surface* of a molecule is the surface which an exterior probe-sphere touches as it is rolled over the (assumed) spherical atoms of that molecule. This definition of a molecular surface was first proposed by Richards [17]. This surface is useful in studying the structure and interactions of proteins. In Figure 1(a) crambin is shown as a collection of spheres whose radii are the van der Waal's radii of the corresponding atoms. In Figure 1(b) the molecular surface of crambin is shown for a probe-sphere radius of 1.4Å (the radius of the spherical approximation to the water molecule).

1.2 Molecular interface surfaces

One of the important factors that influences the position and orientation of the protein with respect to the substrate in protein-substrate docking is the geometric fit or surface complementarity along the protein-substrate interface.



(a) Crambin (396 atoms)



(b) Crambin Surface, Probe Radius = 1.4 Å

Figure 1: Crambin and its molecular surface

Traditionally, the interface has been studied by using a clipping plane that is moved along the z -axis in the screen-space. This enables one to step-through the three-dimensional interface studying its cross-sections in a two-dimensional manner. This is a tedious process and does not readily convey the three-dimensional structure of the interface to the biochemist. The advantages of interface visualization in three dimensions are obvious when studying an interface that does not lie in a plane or when studying pairwise-interfaces across three or more molecular sub-units. Such cases often arise in practice; for example in studying the packing of α -helices in crystalline protein structures.

The definition of a molecular surface that appears in Section 1.1 is that of a *complete molecular surface*, i.e. a surface that completely envelopes a molecule. This surface is useful for visualizing the surface of a single molecule. However, when the objective is to study the interface between two or more molecules (or different sub-units within a single molecule), the complete molecular surface is a poor visualization tool. This is because the occlusion by the complete molecular surfaces (computed independently for each molecule) prevents the study of the inter-molecular contacts and cavities that are buried deep in the interior of the molecular interface. In this paper we define the concept of *molecular interface surfaces* and present an algorithm for their efficient computation.

A related problem is the definition of the interface between two or more molecules. Even though the concept of the interface between molecular units is intuitive and obvious, a formal and precise definition of the *molecular interface region* has not been given thus far. In this paper we provide a definition of the molecular interface region.

2 Previous and related work

An algorithm describing analytic computation of molecular surfaces was first given by Connolly [2]. Edelsbrunner has suggested computing the molecular surfaces by using weighted α -hulls [5]. An implementation of α -hulls for spheres of equal radii (unweighted points) has been presented by Edelsbrunner and Mücke in [6].

Let k be the average number of neighbors per atom in a molecule consisting of n atoms. It has been shown by Varshney *et al* in [18] that under reasonable assumptions for proteins, the average number of neighbors k for a probe-radius of 1.4Å (the radius of the water molecule) will be less than 139, irrespective of n . In practice, k has been found to be around 45 for a probe-radius of 1.4Å. In [18] an approximation to a power cell [1] is computed for each atom in parallel and is used to determine the molecular surface contribution of that atom. This algorithm requires $O(n)$ sequential time and can be parallelized to work in $O(1)$ time (the constant is proportional to $k \log k$) over n processors. Halperin and Overmars in [9] use the fact that there are only a limited number of neighbors per atom to compute an $O(n)$ complexity decomposition of an arrangement of spheres representing the atoms of a molecule. The boundary of the union of spheres is computed from this decomposition in $O(n)$ time.

Connolly has implemented the concept of “buried surface area” [3]. The buried surface area at the interface of two molecules is defined as the sum of the surface areas of the each molecule that become inaccessible to the solvent due to the presence of the other molecule. This definition is geometrically equivalent to the work presented in this paper with the constraint $\alpha = \beta$, i.e. equal values of probe-radius and interface-radius.

Pattern-recognition-based ideas are used to determine surface complementarity for molecules by Katchalski-Katzir *et al* [10]. They map the molecules into a voxel representation and then use Fourier transforms to compute the correlation between two or more molecular sub-units. A similar approach has been taken by Walls and Sternberg in [19] where the molecules are subdivided into planar slices and com-

plementarity of the molecular interfaces is studied amongst the digital contours formed by the intersection of the molecular surface with each gridded plane. There also has been some other interesting geometry-based work on drug design and protein-substrate docking [12, 13, 15].

To the best of our knowledge, no prior work has been done in analytically defining the geometric interface of molecules based upon the concept of power diagrams.

3 Our approach

Let the complete molecular surfaces defined for a probe-radius α for the molecules A and B be represented by $\mathcal{S}(A, \alpha)$ and $\mathcal{S}(B, \alpha)$, respectively. We define the *molecular interface surface* $\mathcal{I}(A, B, \alpha, \beta)$ for a probe-radius α and an interface-radius β for the two molecules A and B as the subset of $\mathcal{S}(A, \alpha)$ and $\mathcal{S}(B, \alpha)$ that includes exactly those points of $\mathcal{S}(A, \alpha)$ that are within a distance β from the surface of some atom of B and exactly those points of $\mathcal{S}(B, \alpha)$ that are within a distance β from the surface of some atom of A . This is illustrated in Figure 2 where the interface surface $\mathcal{I}(A, B, \alpha, \beta)$ is shown in bold.

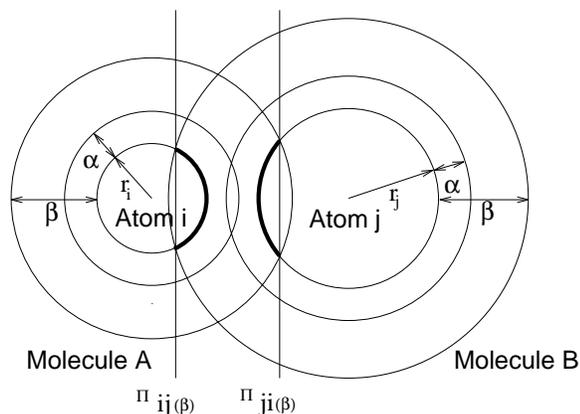


Figure 2: An example of $\mathcal{I}(A, B, \alpha, \beta)$

In Figure 4(a) four domains of transthyretin are shown and in Figure 4(b) the molecular interface surface amongst the transthyretin domains for a probe-sphere radius $\alpha = 2.4\text{\AA}$ and an interface-radius $\beta = 1.0\text{\AA}$ is shown.

The molecular interface surfaces, besides being a useful visualization tool, provide an accuracy-sensitive technique of characterizing the interactions during a protein-substrate docking. The molecular interface surfaces define a hierarchy of detail parametrized by α and β at the inter-molecular interface. Given a par-

ticular value of α , one can define the interface surface by choosing a suitable value for the parameter β based on the computing power available, the desired response time, and the modeling accuracy of the physical interactions. The molecular interface surfaces for different values of α and β for the domains of transthyretin are shown in Figure 5.

3.1 Defining molecular interfaces

Let us first consider the interface between two molecules A and B ; we shall later explain how to deal with cases with more than two molecules.

Let the two molecules A and B be represented as the sets of atoms, $A = \{a_1, a_2, a_3, \dots, a_n\}$ and $B = \{b_1, b_2, b_3, \dots, b_m\}$. We first construct a three-dimensional power-diagram $\mathcal{P}(A, B)$ of the atoms (assumed spherical) for the two molecules A and B . For molecules, the construction of such a power diagram will take time $O(n + m)$ since the number of neighbors for any given atom is bounded by a constant.

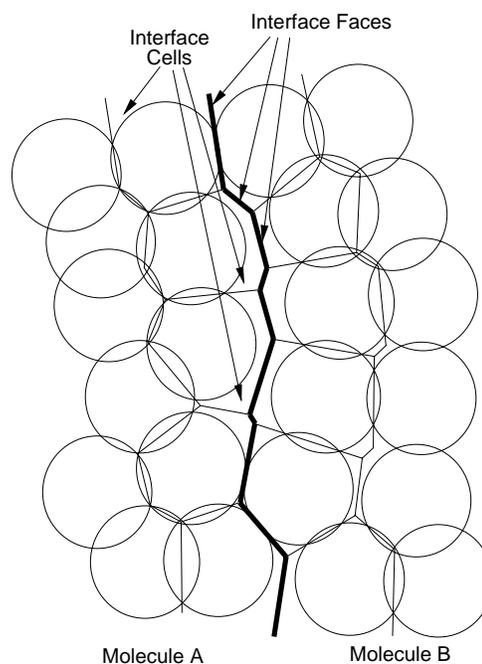


Figure 3: Interface-cells and interface-faces

Each face of the power-diagram $\mathcal{P}(A, B)$ is defined by two atoms. If the two atoms defining a face are $a_i, 1 \leq i \leq n$ and $b_j, 1 \leq j \leq m$, i.e. they belong to the different molecules, then we label such a face as an *interface-face*. A two-dimensional version of this concept is shown in Figure 3 where the interface-faces are shown as a bold polyline. Let us define an *interface-cell* as a cell in the power-diagram $\mathcal{P}(A, B)$ that has at

least one interface-face. Let us define *interface-atoms* to be those atoms whose cells are interface-cells. We note that the molecular interface between molecules A and B is completely defined by the piecewise-planar surface formed by the interface-faces.

This approach can be easily extended to handle cases where more than two molecules form an interface as follows. We construct a power-diagram over the atoms of all the molecules surrounding an interface, as before. We now note that every face of such a power-diagram is defined by exactly two atoms, regardless of how many molecules participate at the interface. We label a face as an interface-face if the two atoms defining it come from two different molecules. Interface-cells and interface-atoms can now be defined just as in the two-molecule case.

3.2 Computing interface surfaces

Let us define an atom a_i of the molecule A to be the ball (spherical volume) $\sigma(c_{a_i}, r_{a_i})$, where c_{a_i} is the center and r_{a_i} is the van der Waal’s radius of the atom a_i . Let us define an α -extended-radius ball $\Psi(a_i, \alpha)$ corresponding to the atom a_i , as a ball concentric to atom a_i whose radius is α more than the radius of a_i . Mathematically, $\Psi(a_i, \alpha) = \sigma(c_{a_i}, r_{a_i} + \alpha)$. Let us define $A(+\beta)$ to be the union of all β -extended-radii balls corresponding to the atoms of A , i.e. $A(+\beta) = \bigcup_{a_i \in A} \sigma(c_{a_i}, r_{a_i} + \beta) = \bigcup_{a_i \in A} \Psi(a_i, \beta)$. An atom a_j is considered to be an α -neighbor of atom a_i if the distance between their centers, $d(c_{a_i}, c_{a_j})$, is less than or equal to $r_{a_i} + 2\alpha + r_{a_j}$. Let $N(a_i, \alpha)$ be the list of atoms in A that are α -neighbors to atom a_i . Thus, $N(a_i, \alpha) = \{a_j | d(c_{a_i}, c_{a_j}) \leq r_{a_i} + 2\alpha + r_{a_j}\}$ where $a_i, a_j \in A$. Terms involving the molecule B are defined similarly. We define a *chordale* Π_{ij} of two atoms i and j represented by $\sigma(c_i, r_i)$ and $\sigma(c_j, r_j)$, respectively, as the plane passing through their circle of intersection. Thus, $\Pi_{ij} = \{x | 2x(c_j - c_i) = r_i^2 - r_j^2 - c_i^2 + c_j^2\}$. We further adopt the notation that the chordale of two extended-radii balls corresponding to the atoms i and j , such that their radii have been extended by α and β respectively, is denoted by: $\Pi_{i(\alpha)j(\beta)} = \{x | 2x(c_j - c_i) = (r_i + \alpha)^2 - (r_j + \beta)^2 - c_i^2 + c_j^2\}$. Continuing with this notation further, it is easy to see that $\Pi_{ij(\beta)}$ corresponds to the chordale of atom i with the β -extended-radius ball corresponding to atom j , i.e. $\Pi_{ij(\beta)} = \{x | 2x(c_j - c_i) = r_i^2 - (r_j + \beta)^2 - c_i^2 + c_j^2\}$. This is illustrated in Figure 2.

Algorithm overview. Let the set of all atoms of molecule A that intersect $B(+\beta)$ be represented by $A_{B(+\beta)}$. We first generate the surface patches of $\mathcal{S}(A, \alpha)$ that are contributed by every atom $a_i \in$

$A_{B(+\beta)}$. The details of surface generation for a given atom are given in [18]. Every surface patch thus generated is clipped by $B(+\beta)$. All the clipped surface patches that lie within $B(+\beta)$ are retained and form that subset of $\mathcal{S}(A, \alpha)$ that belongs to $\mathcal{I}(A, B, \alpha, \beta)$. The subset of $\mathcal{S}(B, \alpha)$ that belongs to $\mathcal{I}(A, B, \alpha, \beta)$ is computed analogously.

Algorithm details. We only give the details of generation of the subset of $\mathcal{S}(A, \alpha)$ that belongs to $\mathcal{I}(A, B, \alpha, \beta)$; the subset of $\mathcal{S}(B, \alpha)$ that belongs to $\mathcal{I}(A, B, \alpha, \beta)$ is computed analogously. To efficiently compute the molecular interface surface we start with the interface-atoms and proceed outwards from the interface.

We first compute the power diagram $\mathcal{P}(A, B)$ over the atoms of the molecules A and B . This is efficiently done by using a global grid to localize the atoms of A and B and then generating approximations to the cells of the power diagram as described in [18]. Once the cells of the power diagram are available we identify the interface-faces, interface-cells, and interface-atoms according to their definitions given in Section 3.1. Then we initialize $A_{B(+\beta)}$ to be the set of all the interface-atoms of A .

Next, for each atom $a_i \in A_{B(+\beta)}$, we generate the closed curves ∂c_{i_p} on $\Psi(a_i, \alpha)$ due to the chordales generated by it and its neighbors: $\Pi_{a_i(\alpha)a_j(\alpha)}$, $a_j \in N(a_i)$. The closed curves ∂c_{i_p} are piecewise-circular arcs (they are in fact the union of a collection of circular arcs on $\Psi(a_i, \alpha)$) and can be generated in time $O(k \log k)$, where k is the size of $N(a_i)$, the set of neighbors of a_i . The details are given in [18].

The complete molecular surface is composed of convex spherical, toroidal, and concave spherical triangular patches depending on whether the probe-sphere defining the surface is in contact with one, two, or three atoms respectively as outlined by Connolly in [2]. As explained in [18], the vertices of the piecewise-circular closed curves ∂c_{i_p} correspond to the positions of the center of the probe-sphere when it is in contact with three atoms and are used to generate the concave spherical triangular patches. The circular arcs defining the piecewise-circular curves ∂c_{i_p} correspond to the locus of the center of the probe-sphere while it is in contact with two atoms and are used to generate the toroidal patches. The interiors of the closed curves ∂c_{i_p} correspond to the locus of the center of the probe-sphere when it is tangential to only atom a_i and are used to generate the convex spherical patches. In this manner we can compute the molecular surface patches that a_i contributes to $\mathcal{S}(A, \alpha)$.

The surface patches contributed by atom a_i are clipped by the union of those spheres from $B(+\beta)$

that overlap it as follows. We generate the chordales $\Pi_{a_i b_j(\beta)}$ due to the ball $\sigma(c_{a_i}, r_{a_i})$ and each sphere in the set $B(+\beta)$. The planes $\Pi_{a_i b_j(\beta)}$ are used to clip the complete molecular surface contributed by atom a_i .

Generation of molecular surface patches from the components ∂c_{i_p} on $\Psi(a_i, \alpha)$ and then clipping them by $\Pi_{a_i b_j(\beta)}$ can be done efficiently in the following manner. We compute the union of the circles formed by the chordales $\Pi_{a_i b_j(\beta)}$ on the sphere $\Psi(a_i, \alpha)$ and then intersect it with the closed, piecewise-circular components ∂c_{i_p} already on the sphere $\Psi(a_i, \alpha)$ from the previous step. By using a central projection, discs on the sphere $\Psi(a_i, \alpha)$ can be mapped to discs on a plane. Thus, the computation of union and intersections of circles on a sphere can be mapped to equivalent operations on a plane. This problem has been studied by Kedem *et al* in [11] and it is shown there that the complexity of the union of k circles on a plane is $O(k)$ and can be computed in time $O(k \log^2 k)$. For our case, k is the total number of α and β -neighbors for a given atom.

Once such a union of circles is available on the plane, it is mapped back to the surface of the sphere $\Psi(a_i, \alpha)$ as a collection of faces with circular arc edges. Depending on their source, we can classify the circular arc edges into two kinds – those that arise due to the chordales $\Pi_{a_i b_j(\beta)}$ (let us refer to them collectively as ∂c_{i_q}) and those that arise due to the chordales $\Pi_{a_i(\alpha) a_j(\alpha)}$, $a_j \in N(a_i)$ (we refer to them collectively as ∂c_{i_p}). Generation of the final molecular interface proceeds from the arcs ∂c_{i_p} , as outlined above. However, clipping due to any plane $\Pi_{a_i b_j(\beta)}$ can now be done efficiently by locating its corresponding circular-arc amongst ∂c_{i_q} and doing the clipping only along the length of the arc (and not over the entire surface contribution of a_i to $\mathcal{S}(A, \alpha)$). Thus the generation of the clipped surface patches now takes time $O(k \log^2 k)$ for every interface atom.

After the processing of a_i is over, we check if any neighbor of a_i overlaps with $B(+\beta)$. We include it in the list $A_{B(+\beta)}$ if it is not already there. In this manner we can incrementally update the list $A_{B(+\beta)}$ for an efficient sequential implementation.

Our approach to computing the smooth molecular surface can be parallelized over all the atoms of the molecule. Each of the steps as described above can be carried out independently for each atom.

4 Implementation and results

Our implementation has been done on Pixel-Planes 5 [7], although it is general enough to be easily portable to any other parallel architecture. In our current im-

plementation we perform the clipping of the surface patches of $\mathcal{S}(A, \alpha)$ (and $\mathcal{S}(B, \alpha)$) by the chordales $\Pi_{a_i b_j(\beta)}$ (and $\Pi_{b_j a_i(\beta)}$) directly in three dimensions instead of mapping the entire computations on to a plane, using a union-of-circles approach, and mapping the results back in three-dimensional space as described in the previous section. Although both of these approaches are mathematically equivalent, it is easier to understand and describe the problem using a union-of-circles approach on a plane. Once the approach is clear, it is easier to implement the clipping directly in three dimensions.

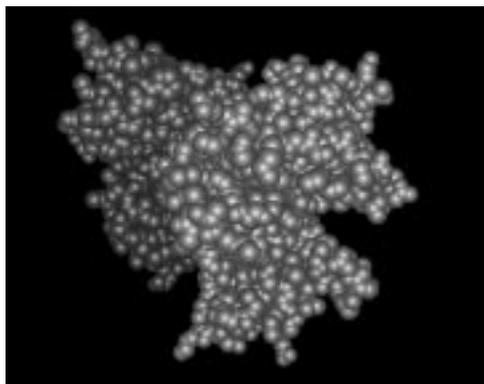
Table 1 shows our timings, in seconds, for computation and display of molecular interface surfaces for molecules pt04-10, pglu10, and rt308, for varying probe-radii α and interface-radii β . The four domains of the molecule pt04-10 contain 915, 896, 915, and 896 atoms, those of pglu10 contain 914, 896, 914, and 896 atoms, and those of rt308 contain 924, 924, 924, and 924 atoms, respectively. The three molecules represent the prealbumin molecules in humans and rats. We chose these three molecules as they are very similar in structure but have important differences arising at their interfaces that lead to different biochemical properties. Defining molecular interface surfaces in the manner that we have done allows one to study the differences amongst these molecules at the interface of their domains.

The results from our approach for molecule pt04-10 (transthyretin) are shown in Figure 5. We have also demonstrated the interface amongst alpha-helices for cytochrome (Protein Data Bank file `pdb256b.ent`) in Figure 6. For these results we were using a configuration of 28 Intel i860 processors.

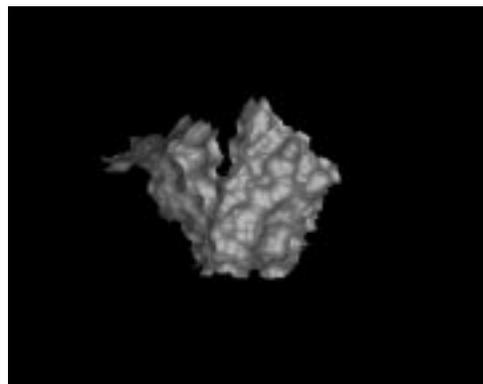
Molecule	α	β			
		(times are in seconds)			
		0.5 Å	1.0 Å	2.0 Å	4.0 Å
pt04-10	1.0 Å	0.53	0.71	1.02	1.93
	1.4 Å	0.55	0.72	1.04	1.94
	2.0 Å	0.57	0.74	1.05	1.95
pglu10	1.0 Å	0.55	0.68	1.00	1.98
	1.4 Å	0.55	0.68	0.99	2.00
	2.0 Å	0.58	0.71	1.05	2.03
rt308	1.0 Å	0.53	0.69	1.05	2.19
	1.4 Å	0.52	0.68	1.05	1.98
	2.0 Å	0.55	0.72	1.06	2.03

Table 1: Molecular Interface Generation Times

We tessellate the molecular interface surface and display it using triangles. If visualization of the interface surfaces is the only goal then an interesting alternative for implementing interface surfaces could be

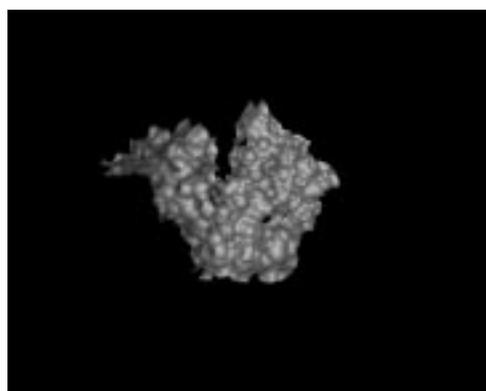


(a) Transthyretin domains

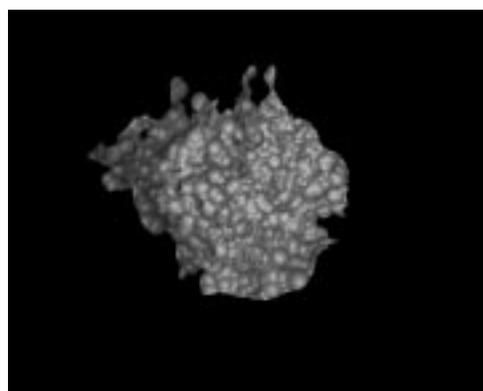


(b) Interface surface for $\alpha = 2.4 \text{ \AA}$, $\beta = 1.0 \text{ \AA}$

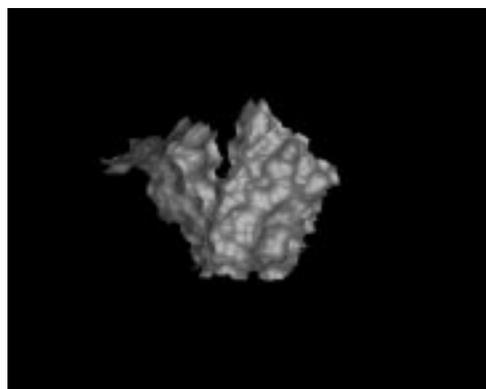
Figure 4: Transthyretin and its interface surface



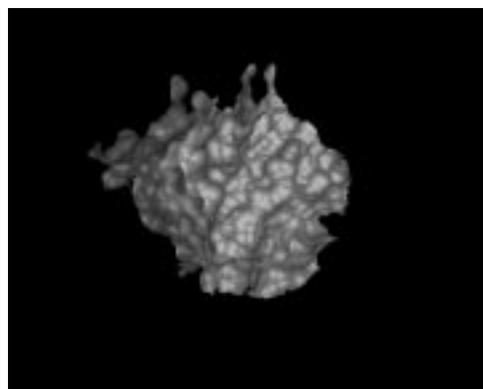
(a) $\alpha = 1.0 \text{ \AA}$, $\beta = 1.0 \text{ \AA}$



(b) $\alpha = 1.0 \text{ \AA}$, $\beta = 2.4 \text{ \AA}$



(c) $\alpha = 2.4 \text{ \AA}$, $\beta = 1.0 \text{ \AA}$



(d) $\alpha = 2.4 \text{ \AA}$, $\beta = 2.4 \text{ \AA}$

Figure 5: Interface surfaces amongst domains in transthyretin

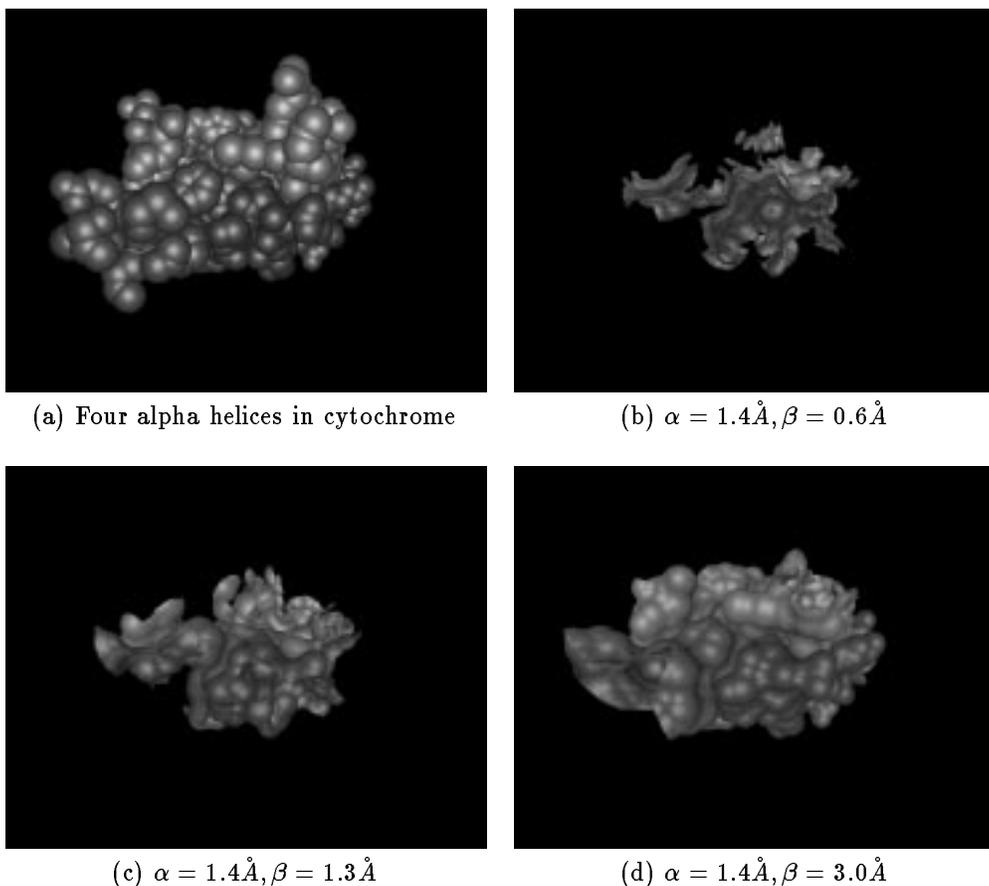


Figure 6: Interface surfaces showing the packing of alpha helices in cytochrome

to use the hardware texture mapping by using boolean textures for clipping the complete molecular surfaces [14, 8, 16]. However, the advantage of explicitly computing a tessellated molecular interface surface is that besides visualization it can be also used for localizing the interactions occurring across the molecular interface. This is outlined further in Section 5.

In general we expect the biochemists to arrive at a reasonable value for the interface-radius β through their knowledge of the molecular interface characteristics and by interactively visualizing the interface with different values of β .

5 Conclusions

We have defined the concept of molecular interface surfaces and have presented an analytic linear-time sequential as well as an efficient parallel algorithm for computing them. The interface surfaces as defined in this paper localize and reduce the set of possible inter-

actions at a molecular interface. Therefore these surfaces should also find use in increasing the efficiency of the algorithms that process the electrostatic and hydrophilic interactions at the interface. Thus, these results provide a general framework for further research on the geometric and electrostatic complementarity and hydrophilic compatibility in the protein-substrate docking problem. We hope that this will allow one to characterize and visualize the molecular interface complementarity better and encourage further research in the area of molecular docking.

Acknowledgements

We would like to acknowledge the valuable discussions we had with Pankaj Agarwal and Jan Prins during various stages of this work. We would also like to acknowledge the useful comments from the anonymous referees. The data for molecules pt04-10, pglu10, and rt308 was provided to us by Dr. Vivian Cody, Medical Foundation of Buffalo.

Work by Amitabh Varshney, Frederick P. Brooks,

Jr., and William V. Wright has been supported in part by NIH NCRG grant number 5-P41-RR02170. Work by Amitabh Varshney has also been supported in part by a University of North Carolina Dissertation Fellowship and NSF grant CCR-9502239. Work by Frederick P. Brooks, Jr. has also been supported in part by ARPA Contract DABT63-93-C-0048. Work by David C. Richardson has been supported in part by NIH grant number GM-15000. Work by Dinesh Manocha has been supported in part by a Sloan Foundation Fellowship, University Research Award, NSF Grant CCR-9319957, ONR Contract N00014-94-1-0738, and ARPA Contract DABT63-93-C-0048. Equipment support has been provided by NSF/ARPA, Science and Technology Center for Computer Graphics and Scientific Visualization, Agreement number ASC-8920219.

References

- [1] F. Aurenhammer. Power diagrams: Properties, algorithms and applications. *SIAM Journal of Computing*, 16(1):78–96, February 1987.
- [2] M. L. Connolly. Analytical molecular surface calculation. *Journal of Applied Crystallography*, 16:548–558, 1983.
- [3] M. L. Connolly. Solvent-accessible surfaces of proteins and nucleic acids. *Science*, 221(4612):709–713, 1983.
- [4] R. E. Dickerson and I. Geis. *The Structure and Action of Proteins*. Harper & Row, New York, NY, 1969.
- [5] H. Edelsbrunner. Weighted alpha shapes. Technical Report UIUCDCS-R-92-1760, Department of Computer Science, University of Illinois at Urbana-Champaign, 1992.
- [6] H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, 13(1):43–72, January 1994.
- [7] H. Fuchs, J. Poulton, J. Eyles, T. Greer, J. Goldfeather, D. Ellsworth, S. Molnar, G. Turk, B. Tebbs, and L. Israel. Pixel-planes 5: A heterogeneous multiprocessor graphics system using processor-enhanced memories. In *Computer Graphics: Proceedings of SIGGRAPH'89*, volume 23, No. 3, pages 79–88. ACM SIGGRAPH, July 1989.
- [8] Paul Haeberli and Mark Segal. Texture mapping as A fundamental drawing primitive. In Michael F. Cohen, Claude Puech, and Francois Sillion, editors, *Fourth Eurographics Workshop on Rendering*, pages 259–266. Eurographics, June 1993. held in Paris, France, 14–16 June 1993.
- [9] D. Halperin and M. H. Overmars. Spheres, molecules, and hidden surface removal. In *Proc. 10th Annu. ACM Sympos. Comput. Geom.*, pages 113–122, 1994.
- [10] E. Katchalski-Katzir, I. Shariv, M. Eisenstein, A. A. Friesem, C. Aflalo, and I. A. Vasker. Molecular surface recognition: Determination of geometric fit between proteins and their ligands by correlation techniques. *Proceedings, National Academy of Sciences, USA*, 89:2195–2199, March 1992.
- [11] K. Kedem, R. Livne, J. Pach, and M. Sharir. On the union of Jordan regions and collision-free translational motion amidst polygonal obstacles. *Discrete Comput. Geom.*, 1:59–71, 1986.
- [12] Irwin D. Kuntz. Structure-based strategies for drug design and discovery. *Science*, 257:1078–1082, 1992.
- [13] Irwin D. Kuntz, Jeffrey M. Blaney, Stuart J. Oatley, Robert Langridge, and Thomas E. Ferrin. A geometric approach to macromolecule–ligand interactions. *JMB*, 161:269–288, 1982.
- [14] W. E. Lorensen. Geometric clipping using boolean textures. In G. M. Nielson and D. Bergeron, editors, *IEEE Visualization '93 Proceedings*, pages 268–274, October 1993.
- [15] G.R. Marshall *et al.* The conformational parameter in drug design: The active analog approach. In ACS Symposium series. E. C. Olson and R.E. Christoffersen, editors, *Computer-Assisted Drug Design*, volume 112, pages 205–226, 1979.
- [16] A. Olson, January 1994. Private Communication.
- [17] F. M. Richards. Areas, volumes, packing and protein structure. *Ann. Rev. Biophys. Bioeng.*, 6:151–176, 1977.
- [18] A. Varshney, F. P. Brooks, Jr., and W. V. Wright. Computing smooth molecular surfaces. *IEEE Computer Graphics & Applications*, 15(5):19–25, September 1994.
- [19] P. H. Walls and M. J. E. Sternberg. New algorithm to model protein-protein recognition based on surface complementarity. *Journal of Molecular Biology*, 228:277–297, 1992.