Quality Mesh Generation for Molecular Skin Surfaces Using Restricted Union of Balls

Ho-Lun Cheng *   Xinwei Shi †

School of Computing
National University of Singapore
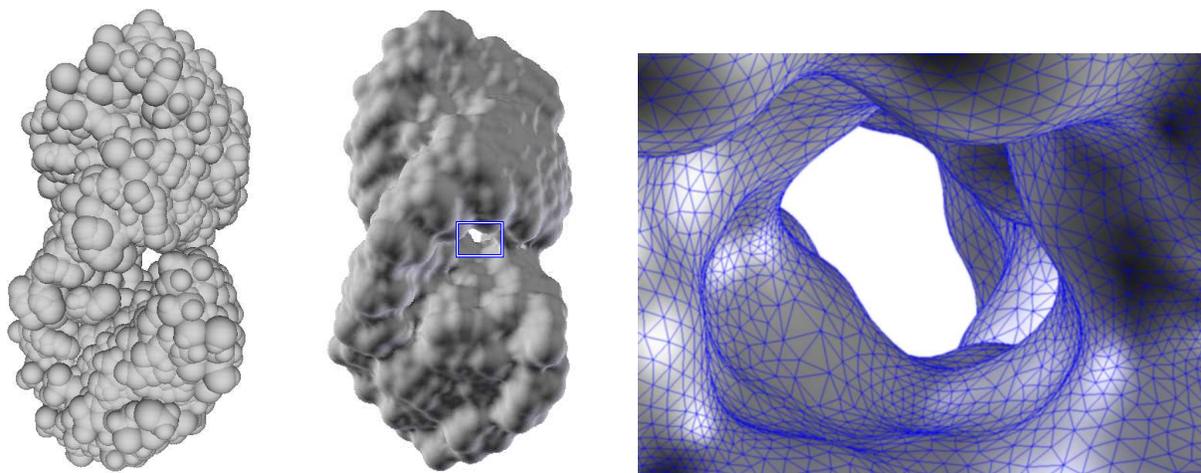
Figure 1: Molecular models of the molecule Helix. The leftmost shows the van der Waals surface model; the center shows the molecular skin model and the rightmost shows the magnified mesh details in the box of the center figure.

ABSTRACT

Quality surface meshes for molecular models are desirable in the studies of protein shapes and functionalities. However, there is still no robust software that is capable to generate such meshes with good quality. In this paper, we present a Delaunay-based surface triangulation algorithm generating quality surface meshes for the molecular skin model. We expand the *restricted union of balls* along the surface and generate an ε-sampling of the skin surface incrementally. At the same time, a quality surface mesh is extracted from the Delaunay triangulation of the sample points. The algorithm supports robust and efficient implementation and guarantees the mesh quality and topology as well. Our results facilitate molecular visualization and have made a contribution towards generating quality volumetric tetrahedral meshes for the macromolecules.

CR Categories: J.3 [Life and Medical Sciences]: Biology and Genetics; I.3.8 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, Surface, Solid, and Object Representation

Keywords: Smooth surfaces, meshing, restricted union of balls, Delaunay triangulation, guaranteed quality triangulation, homeomorphism

*e-mail: hcheng@comp.nus.edu.sg
†e-mail: shixinwe@comp.nus.edu.sg

1 INTRODUCTION

Molecular biology is receiving growing attention from computer scientists. With the recent success of the Human Genome Project, an important aspect of the biological study is the investigation of the three dimensional structures of molecules and their functionalities. A fundamental computational primitive supporting these studies is to construct digital representations of molecular surfaces. The polygonal mesh is one of the most popular representations because they facilitate not only visualizations but also numerical simulations on macromolecules. Wherein the mesh quality is important for the convergence and accuracy of the simulation. In addition, the homeomorphism between the mesh and the surface is essential to provide a correct boundary representation. In this study, we investigate efficient algorithms to generate quality and homeomorphic meshes for the molecular skin models.

1.1 Previous work

Molecular surface meshing has been much discussed in the recent literature. Amitabh et. al [4] presented a parallel algorithm for computing the triangulated molecular surface efficiently. The algorithm assumed an upper bound on the number of neighboring atoms to each atom and achieved linear computation over *n* processors. However, the paper offered little in the way of handling molecular surface self-intersections and guaranteeing the mesh quality. Akkiraju and Edelsbrunner [3] described algorithms for triangulating the surface of space filling (SF), solvent accessible (SA), and molecular surface (MS) models. The computation is based on the regular triangulation of a set of balls representing the atoms of a molecule. The implementation of the algorithm ran into robustness problem because of the inconsistency between the regular triangulation and

the surface meshes [3]. In addition, the author did not address the problem of molecular surface self-intersections nor offer any guarantees on the mesh quality. Laug and Borouchaki [14] applied surface mesh generation techniques to construct quality meshes for molecular surfaces. First, an analytical representation of the surface was computed and parameterized. Then, triangular meshes were constructed in the parametric space and mapped onto the molecular surface. The generated triangulation had good quality but was limited to a few small molecules free of self-intersections. By far, the molecular surface self-intersections have been the bottleneck in developing robust surface mesh generation tools for molecules.

A number of other studies have proposed new surface models for molecules. Zhang et.al [16] investigated the approximation of implicit solvation surfaces of molecules using the electron density maps. The author used the summation of Gaussian kernel functions to construct the density map and sample volumetric data. Surface meshes were extracted from the volumetric data as an isosurface using the dual contouring methods. However, the isosurface generated by the Gaussian density function is not smooth because the volume data could not be sufficiently fine. The error of the isosurface could be very large relative to the atom size and resulted in inaccurate representations.

Edelsbrunner [10] introduced the molecular skin model based on a framework of the Voronoi diagram and Delaunay triangulation of a set of weighted points. We argue that the skin model outperforms the existing surface models because the skin surface is smooth, free of self-intersections and capable of being parameterized, triangulated with good quality and deformed freely with smooth transitions [10]. Cheng et.al [6] developed an algorithm that constructed the skin triangulation by growing an initial Delaunay mesh dynamically. However, the efficiency is not satisfactory because the algorithm computes a large number of redundant intermediate meshes between the initial mesh and the final surface mesh.

Recently, Cheng and Shi [7] presented an advancing front meshing algorithm to generate quality meshes for the skin surface efficiently. In this algorithm, the critical points of a height function on the surface were used to detect front collision and the Morse-Smale complex was constructed to eliminate noisy critical points. This method improved the efficiency dramatically but resulted in robustness problem of the implementation. For example, this algorithm failed to triangulate the molecular skin surface illustrated in Figure 2 because the bumpy appearance on the surface resulted in a large number of noisy critical points and a complicated Morse-Smale complex. The accumulation of numerical errors in the approximation of Morse-Smale complex led to inconsistent critical points after eliminating noisy critical points inaccurately. An analytical representation of the Morse-Smale complex would avoid such a problem. However, the current methods for solving partial differential equations(PDEs) make it impossible for us to have analytical representations because the arcs in the Morse-Smale complex are solutions of partial differential equations with order 2 and it is difficult to obtain the analytical solutions for such equations.

Boissonnate and Oudot [5] proposed a Delaunay based algorithm for triangulating smooth surfaces with provable quality and topology guarantees. However, since the algorithm requires the computation of the local feature size of surface points, it is not applicable for the molecular skin surface because the media axis computation of the surface is hard.

To sum up, it is still challenging to build robust software to generate quality surface meshes for molecules. In this paper, we present a robust Delaunay-based meshing algorithm to generate quality meshes for the molecular skin models.

1.2 Our approach

We employed the Delaunay-based method to generate quality mesh for the molecular skin surface incrementally. In particular, we add
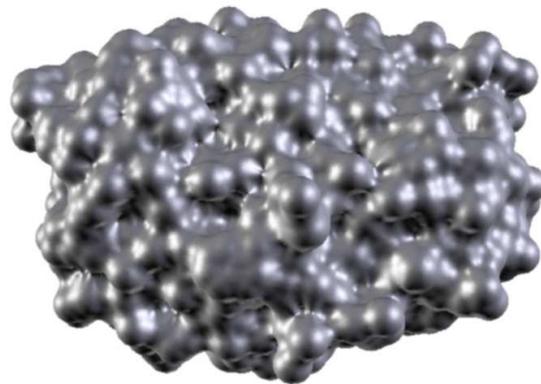
Figure 2: The molecular skin model of HIV-2 protease.

one sample point each time and maintain the Delaunay triangulation of the sample points with the incremental flipping algorithm. Then, we extract a subset of the Delaunay triangulation as the candidate surface triangles. This extraction only requires a local update because most of the candidate surface triangles remain the same. These candidate surface triangles forms a partial mesh and guides the future point sampling. We iteratively apply this procedure until we get an ε-sampling of the whole surface. At the same time, the surface mesh is extracted from the collection of all candidate surface triangles. Our Delaunay-based meshing algorithm can offer mathematical guarantees on the mesh quality as well as robustness in practice.

In order to guarantee the mesh quality and topology, we employ the *restricted union of balls*, which is introduced in Section 3.1. The restricted union of balls ensures that our algorithm generates an ε-sampling of the skin surface. Because the restricted Delaunay triangulation of an ε-sampling is homeomorphic to the skin surface if ε is sufficiently small, we can extract the restricted Delaunay triangles from the Delaunay triangulation to construct homeomorphic mesh [6]. At the same time, we can guarantee the length of the edges in the surface mesh is not too short and the circumradius of the triangle is small by utilizing the restricted union of balls. As a result, we can guarantee the termination of algorithm and achieve a lower bound on the minimum angle of each triangle in the surface mesh.

One point worth noting here is the efficiency of the algorithm. Since the incremental algorithm for Delaunay triangulation in three dimensions has quadratic complexity in the worst case, our algorithm may take quadratic time. Fortunately, we overcome this deficiency by speeding up the point location procedure in the computation of Delaunay triangulation. Instead of using a directed acyclic graph (DAG) to store the history of performed flips, we give a straightforward yet fast way to locate the tetrahedron enclose the newly inserted point by taking the advantage of our point sampling strategy. The results of our implementation provide evidence of the robustness, efficiency and quality guarantees of the algorithms.

Outline. Section 2 introduces some geometric fundamentals. Section 3 describe our Delaunay skin meshing algorithm. Section 4 illustrates some experimental results of our meshing algorithm and we conclude the paper in Section 5.

2 PRELIMINARIES

We first introduce some definitions, then describe a few previous results from [6] for the understanding of our algorithm.

2.1 Definitions

Skin surfaces. A skin surface F_B is defined by a set of spheres B in $\mathbb{R}^3$. It is the envelope of an infinite family of spheres derived from B by convex combination and shrinking. Even though the family of spheres is infinite, the skin surface can be decomposed to a collection of quadratic patches. Each patch is a portion of sphere or hyperboloid clipped within a polyhedron obtained by shrinking the Minkowski sum of the corresponding Voronoi and Delaunay polyhedron. The molecular skin model of a molecule is the skin surface specified by the set of spheres positioned at each atom coordinate and equipped with a radius as $\sqrt{2}$ times the summation of the atom's van der Waals radius and the radius of the probe sphere, which is usually chosen as 1.4 Angstrom to represent the water as solvent.

Curvature property. The skin surface is a C^1 continuous surface and its maximum curvature is continuous as well. We denote the maximum principal curvature at $x \in F_B$ as $\kappa(x)$. The reciprocal $1/\kappa(x)$ is called the *local length scale* at x, denoted as $\rho(x)$. For two points $x, y \in F_B$, the variation of the local length scale can be expressed in the Lipschitz condition,

$$|\rho(x) - \rho(y)| \leq \|x - y\|.$$

We use the local length scale to control the sampling density in the triangulation. In particular, we require the vertices of the triangulation to be an ε-sampling.

ε-sampling of skin. A finite subset $P \subset F_B$ is an ε-sampling of F_B if every point $x \in F_B$ has a point $p \in P$ such that their distance is at least $\varepsilon\rho(x)$. For an ε-sampling on the skin with a feasible ε value, its *restricted Delaunay triangulation* is homeomorphic to the skin surface.

Restricted Delaunay triangulation. The *restricted Voronoi polygon* of $a \in P$ is defined as $v_a' = v_a \cap F_B$, in which v_a is the Voronoi cell of a with respect to P in $\mathbb{R}^3$. The nerve of the restricted Voronoi polygons is the set of these polygons with non-empty common intersection. The collection of the convex hull of the nerve element is the *restricted Delaunay triangulation*, namely,

$$D_P' = \left\{ \text{conv}\,(U) \quad | \quad U \subseteq P, \quad F_B \cap \bigcap_{a \in U} v_a \neq \emptyset \right\}.$$

We aim to generate an ε-sampling of a skin surface and construct its restricted Delaunay triangulation. Next, we introduce several properties of the restricted Delaunay triangulation of an ε-sampling of the skin surface.

2.2 Homeomorphic Conditions

The restricted Delaunay triangulation is homeomorphic to the surface if it satisfies the closed ball property [11]. This implies that we need a dense enough sample on the surface. We refer to the results from Cheng et. al [6].

Thm. 2.1 Homeomorphism Theorem. If P is an ε-sampling of a skin surface F_B with $0 < \varepsilon < 0.279$, the restricted Delaunay triangulation D_P' is homeomorphic to F_B.

For each triangle $abc \in D_P'$, its Voronoi edge passes through its circumcenter o and intersects the skin surface at a point z. Moreover, the triangle abc have a small circumradius compared to local length scale at its vertices and the distance between o and z have an upper bound as well. These two properties are expressed in the following two Lemmas from [6].

Lemma 2.1 The cricumradius R_{abc} have an upper bound, namely,

$$R_{abc} < \frac{\varepsilon}{1 - \varepsilon}\rho_{abc}.$$

Lemma 2.2 The distance between o and z is

$$\|oz\| \leq \frac{\varepsilon^2}{2}\rho_{abc},$$

in which the local length scale of the triangle abc is defined as $\rho_{abc} = \min\{\rho(a), \rho(b), \rho(c)\}$. We use these two bounds as the conditions to select candidate surface triangles from the Delaunay triangulation of a partial sampling.

3 Delaunay Skin Meshing Using Restricted Union of Balls

In this section, we first introduce the definition of the restricted union of balls and the guarantees of mesh topology and quality. Then, we give an overview of our meshing algorithm followed by the detail description of the key operations in the algorithm.

3.1 Restricted Union of Balls

For each point $p \in P$, we define the γ-ball $\hat{p}$ of p as the open ball centered at p with a radius $\gamma\rho(p)$, in which γ is a positive constant less than 1 and the $\rho(p)$ is the local length scale at p. A γ-ball is *empty* if no other sample point in P is inside the γ-ball. Each γ-ball intersects the skin surface with a topological disk and the intersection of the skin and a union of a set of γ-balls is called *the restricted union of balls*. The boundary of the restricted union of balls is a set of closed curves consisting of a loop of arcs when the union of balls does not cover the whole surface. A γ-ball that contributes to the boundary of the restricted union of balls is called a *boundary ball*.

We use the restricted union of balls to generate an ε-sampling of the skin surface. We prove that the restricted union of balls without boundaries determines an ε-sampling of the skin surface when the value of γ is small. We have the following theorem.

Thm. 3.1 Sampling Density Theorem. For a sampling $P \subset F_B$, if its restricted union of balls with $\gamma \leq \frac{\varepsilon}{1+\varepsilon}$ covers the skin surface, P is an ε-sampling of the skin surface F_B.

PROOF. Let x be any point on the skin surface. Since the restricted union of balls covers the whole surface, we have at least one γ-ball $\hat{a}$ enclose x. We have the upper bound for γ when x is on the boundary of $\hat{a}$. In this case, $\|xa\| = \gamma\rho(a)$.

By definition of the ε-sampling, we require $\|xa\| \leq \varepsilon\rho(x)$. Together with the curvature property, we have $\gamma\rho(a) \leq \varepsilon(\rho(a) - \gamma\rho(a))$. That is, $\gamma \leq \frac{\varepsilon}{1+\varepsilon}$. ⊡

Theorem 3.1 implies that we can construct a restricted union of balls with $0 < \gamma < 0.218$ to obtain an ε-sampling with $0 < \varepsilon < 0.279$, which suggests that we can guarantee the surface mesh is homeomorphic to the skin surface.

If we require each γ-ball is empty, we can guarantee the length of an edge ab in the surface mesh have a lower bound, that is,

$$\|ab\| \leq \gamma\rho_{ab},$$

in which the local length scale of the edge ab is defined as $\rho_{ab} = \max\{\rho(a), \rho(b)\}$. Together with the Lemma 2.1, we can guarantee a lower bound on the minimal angle of the surface mesh, that is, $arcsin(\frac{\gamma}{2\varepsilon}(1 - \varepsilon))$. This suggests that smaller γ and ε mean better

mesh quality if we ensure that they satisfy the upper bound. We can choose the values of the γ and ε as small as possible. However, this would result in more surface triangles. We should keep a balance between the mesh quality and mesh size. For example, we chose $\gamma = 0.15$ and $\varepsilon = 0.18$ and achieved a lower bound of $20°$ on the minimum angle.

3.2 Overview of our algorithm

Our algorithm constructs the triangulation of a given skin surface by meshing each connected component individually. We can differentiate all the connected components with the alpha shape [12] and its Betti numbers [9] because each connected component of the skin surface F_B corresponds to a connected component of the alpha complex K_B specified by B. This corresponding relationship is based on the homotopy equivalence between the spaces enclosed by the skin surface F_B, namely, the skin body, and the underlying space of the alpha complex [10].

For each component, we start from a seed point and insert new points along the boundary of the restricted union of balls incrementally. Each newly inserted point is located outside the current restricted union of balls. After a new point is inserted, we compute the Delaunay triangulation and extract the small restricted Delaunay triangles as the candidates for the surface triangulation. At the same time, the restricted union of balls expands along the skin surface and the boundary is updated until the whole surface is covered. The algorithm terminates when the restricted union of balls covers the whole surface. Figure 3 illustrates a scenario of a vertex insertion.
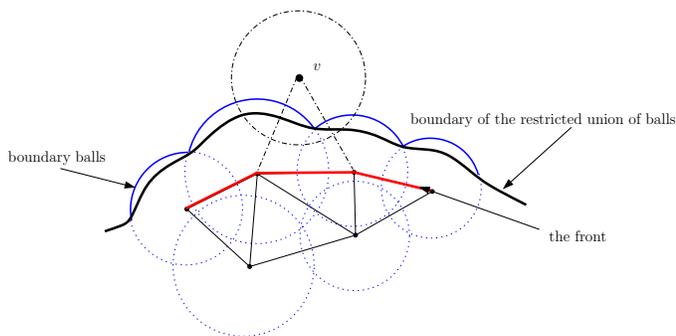
Figure 3: The vertex insertion in the algorithm.

Denote the current sampling point set as P_i and Delaunay triangulation of P_i as D_i. We choose all the small restricted Delaunay triangles in D_i as the *candidate surface triangles*, denoted as $S_i \subset D_i$. Ideally, S_i could be a piecewise 2-manifold with boundary that is an exact subset of the final surface mesh. However, we cannot achieve this from a partial sampling and the candidate surface triangles S_i is a superset of the 2-manifold, which may include some false surface triangles resulting in tetrahedra or non-manifold elements. These false surface triangles can be cleaned up when we obtain an ε-sampling.

With P_i, we insert a new point v to get P_{i+1}. Since the point v must be outside of the current restricted union of balls, we identify the boundary balls to locate v. The boundary balls are found by using a subset of the Delaunay edges in S_i, namely, the *front*. Each front edge is either a dangling edge in S_i or an edge shared by two candidate triangles whose normals form an angle larger than $90°$. The collection of all the γ-balls at the vertices of the front edges includes all the boundary balls and a small number of interior γ-balls. We can differentiate these interior balls in our point placement methods and locate the new point v on the restricted Voronoi edge of a front edge whose vertices γ-balls are boundary

balls. Then, we maintain the Delaunay triangulation of P_{i+1} and extract new candidate surface triangles. As a result, the collection of candidate surface triangles grows from S_i to S_{i+1} and the front advances to the unmeshed region. We store the *front* in a queue Q and iteratively apply the above procedure until the front Q is empty, that is, the restricted union of balls cover the whole surface. The pseudo code of the algorithm is illustrated as following.

Algorithm 1 DeloneSkinMesh()

1: **while** The boundary of restricted union of balls is not empty **do**
2:   Find a front edge ab;
3:   Locate a new point v according to ab;
4:   Compute the Delaunay triangulation of $P_i \cup \{v\}$;
5:   Extract the candidate surface triangles;
6:   Update the front;
7: **end while**
8: Clean up the non-restricted Delaunay triangles from S;

The algorithm consists of four main steps, namely, locating a new vertex, updating the Delaunay triangulation, extracting candidate surface triangles and updating the front. When the restricted union of balls covers the surface and no more points can be inserted, we get an ε-sampling and the collection of candidate triangles includes the restricted Delaunay triangulation of the ε-sampling. We clean up the non-restricted Delaunay triangles and get the final surface mesh. Next, we discuss these steps in details.

3.3 Point Placement

Assume that we have a front edge ab whose γ-balls $\hat{a}$ and $\hat{b}$ are boundary balls. We aim to locate a new sample point v satisfying the following requirements:

- the point v should be outside the restricted union of balls;

- the γ-ball $\hat{v}$ should be free of any sample point in P_i;

- the γ-ball $\hat{v}$ should intersect the boundary balls $\hat{a}$ and $\hat{b}$ deeper than tangentially;

- the point v will form at least one candidate surface triangles with the points in P_i.

The first two requirements maintains the empty property of the γ-balls, which avoids short edges in the mesh and implies that the algorithm will terminate. The third requirement is to ensure the restricted union of balls cover the whole surface once the algorithm terminates. The last one aims to maintain a valid front to guide our future vertex insertion. These requirements imply that the distance from the point v to a and b should be larger than $\gamma \rho_{ab}$ but not too far from a and b. This suggests that the points along the intersection between the skin and the boundary balls $\hat{a}$ and $\hat{b}$ would be a good choice. We argue that there is always a space to find a point v satisfying these requirements before the algorithm terminates. In particular, we can locate the point v on the restricted Voronoi edge of the front edge ab. See Figure 4 as an example.

In Figure 4, ab is a front edge and its restricted Voronoi edge is the curve wy, which is the intersection of the skin and the Voronoi polygon of ab with respect to the points in P_i. The solid line circles represent the γ-balls and the point m is the intersection point of the skin surface and the boundary of $\hat{a}$ and $\hat{b}$ that is outside the restricted union of balls. The point v is the point we intend to insert and edge va and vb are the Delaunay edges of D_{i+1} after v is inserted. We demonstrate that point v can be found on the curve wy around the point m. Since ab is a front edge, the endpoint w is on the Voronoi
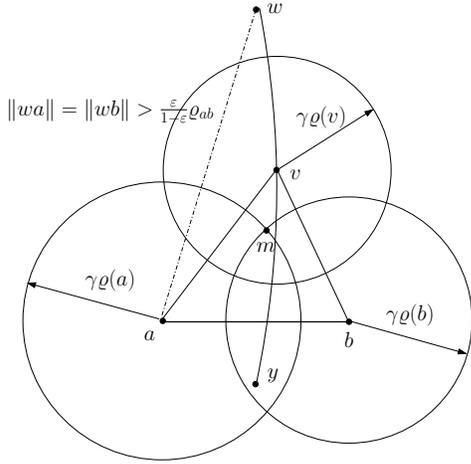
Figure 4: Locate the new point v correspond to a front edge ab.

edge of a Delaunay triangle abx in D_i and abx is not a candidate surface triangle. Thus, the circumradius R_{abx} of the triangle abx is greater than $\frac{\varepsilon}{1-\varepsilon}\rho_{abx}$. It implies that $\|wa\| = \|wb\| > \frac{\varepsilon}{1-\varepsilon}\rho_{abx}$, which means the point w is outside $\hat{a} \cup \hat{b}$ and far from the corner point m. As a result, we can find the point v starting from m toward w along the curve wy.

The computation of the intersection curve wy between the skin and the Voronoi polygon of edge ab is costly. Therefore, we use the projection of wy on the tangent plane of the point a, namely, $w'y'$, to locate the point v. We denote the projection of m on the line $w'y'$ as m' and $w'y'$ is a line segments passing through the middle point t of a and b. We find a point v' on $w'y'$ start from t with a length slightly longer than tm'. For instance, $\|tv'\| = \frac{1+\sqrt{3}}{2}\|tm'\|$ is a feasible choice in practice. We get the point v by projecting v' to the skin surface.

In the case where the vertices γ-balls of front edge ab is not boundary balls, the point v we compute using the previous procedure must be inside the restricted union of balls. We simply discard the point v and cancel the candidate surface triangles attached to the edge ab. Next, we compute the Delaunay triangulation, D_{i+1}, of $P_i \cup \{v\}$.

3.4 Computation of Delaunay triangulation

We adapt the incremental flip algorithm to construct the Delaunay triangulation D_{i+1} efficiently. The incremental flip algorithm was initially proposed by Lawson [15]. The basic idea of the algorithm is the following. Let P be a set of n points in $\mathbb{R}^3$, $4 < i < n$ and assume that the Delaunay triangulation of the first i points in P is already constructed, called D_i. Add the $(i+1)$-th point to triangulation and restore the Delaunayhood by flipping, this results in D_{i+1}. Repeat this process until $i = n$. A crucial step in the algorithm is the point location, which occurs when a new point is added in to the triangulation. A directed acyclic graph (DAG) with the history of all performed flips were used to speed up the point location. However, the DAG structure results in large memory usage and complicated implementation. In our algorithm, we employ a straightforward yet fast way to locate point by taking the advantage of our point sampling strategy.

Since the newly inserted point v corresponds to a front edge ab and is not far from a and b, we simply search the tetrahedra in a certain region around ab. Particularly, we choose a Delaunay triangle abd in the star of edge ab whose normal has minimal difference with the normal of point v at the skin surface as a starting point.

The point v only belongs to one of the half spaces divided by the plane pass through triangle abd, denote as H. If the triangle abd is a facet on the convex hull and there is no points in P_i belonging to the half space H, the point v would be on the convex hull also and we return the artificial tetrahedron connecting abd with the point at infinite. Otherwise, there must be a tetrahedron enclosing v and we can find it by walking through the triangles enclosed by a sphere centered at a and with a radius of 1.5 times of the length of edge va along the direction toward v. In general, the point v is either a convex hull point or is enclosed by a tetrahedron attached to the start point triangle. That is, we only need a constant time to locate the tetrahedron enclosing v, which accelerate the construction of the Delaunay triangulation very much. Then, we connect v to the vertices of the tetrahedron and perform flips to restore the Delaunay property of the triangulation. Our implementation is based on the results of alpha shape software [1, 12].

After the Delaunay triangulation D_{i+1} is accomplished, we extract the candidate surface triangles. The Delaunay triangulation D_{i+1} differs from D_i with the star of v, which is a set of tetrahedra that include v as one of their vertices. Therefore, we can only consider the triangles in the star of v for labeling new candidate surface triangles.

3.5 Extraction of Candidate Surface Triangles

We select candidate surface triangles according to two conditions, namely, the small circumradius condition and restricted Delaunay condition. By "small" we mean the circumradius R_{abc} of a Delaunay triangle abc is smaller than $\frac{\varepsilon}{1-\varepsilon}\rho_{abc}$. For the restricted Delaunay property, we not only require the Voronoi edge V_{abc} intersects the skin surface but also need the distance between the intersection point z and the circumcenter o is less than $\frac{\varepsilon^2}{2}\rho_{abc}$. The following pseudo code shows the procedure $ExtractCandidateTringles()$.

Algorithm 2 ExtractCandidateTriangles()

1: unmark all the triangles in $Star(v)$
2: **while** there is an unmarked triangle $abc \in Star(v)$ **do**
3:     **if** $R_{abc} < \frac{\varepsilon}{1-\varepsilon}\rho_{abc}$ **then**
4:         **if** V_{abc} intersect the skin surface with a closest point z and $\|oz\| < \frac{\varepsilon^2}{2}\rho_{abc}$ **then**
5:             mark abc as a candidate surface triangle;
6:         **else**
7:             mark abc as a non candidate surface triangle;
8:         **end if**
9:     **else**
10:         mark abc as a non candidate surface triangle;
11:         **for** the edge $xy \in \{ab, bc, ac\}$ **do**
12:             **if** $\|xy\| > \frac{2\varepsilon}{1-\varepsilon}\rho_{abc}$ **then**
13:                 mark all the triangles in $Star(xy)$ as non candidate surface triangles;
14:             **end if**
15:         **end for**
16:     **end if**
17: **end while**

The small circumradius condition and restricted Delaunay condition ensure that the candidate surface triangles include all the restricted Delaunay triangles in the final restricted Delaunay triangulation of the ε-sampling. This argument is based on the following two observations. First, the small Delaunay triangles would be always Delaunay if we did not insert any point into its smallest circumsphere. It implies that the points sampled in the future will not invalidate most existing candidate surface triangles. Second, the restricted Voronoi edge of a Delaunay triangle can only become shorter after more points are sampled, which means that a

restricted Delaunay triangle in a partial mesh could become a non-restricted Delaunay triangle but a non-restricted Delaunay triangle would never be a restricted Delaunay later.

We can accelerate the candidate surface extraction procedure by checking the edge length of a large triangle. Since the longest edge of a triangle is as long as 2 times of its circumradius, we can distinguish whether a Delaunay triangle is not a candidate surface triangle if its edge is too long.

Finally, we update the front from the new candidate surface triangles by checking each edge of a candidate triangle. If the edge is a dangling edge or shared by two triangles with large normal angles, then we put the edge to the queue Q.

If the restricted union of balls covers the whole surface and no more points can be added, the sample points P is an ε-sampling of the skin surface. We walk through all the candidate surface triangles and clean up all the non-restricted Delaunay triangles. The remaining triangles form the restricted Delaunay triangulation of the ε-sampling P, which is a quality surface mesh approximating the skin surface.

4 RESULTS

We implemented the algorithm on the PC platform with C++ using OpenGL as the graphics library. We partially reuse prior software on alpha shapes [1] and Betti numbers [9]. We triangulate a few large molecular skin models of proteins, especially several molecules that our previous algorithm fails to generate the mesh because of the noisy critical points in the Morse-Smale Complexes.

Figure 2 shows the molecular skin model of a protease molecule of the Human Immunodeficiency Virus (HIV). Our algorithm triangulates this surface and produce a mesh with a minimum angle 20.35°. Since the surface of the molecule has a very bumpy appearance, our previous algorithm failed to triangulate this surface [7]. In contrast to this, our current algorithm can handle the bumpy surface robustly. This can be explained by two factors: first, we did not use the Morse-Smale complex in this algorithm so that there are no accumulation of numerical error involved in the computation; second, the robust and stable implementation of the alpha shape software relieve us from the robustness worries associated with the Delaunay triangulation computation.

Figure 5 and 6 show the molecular skin model of two molecules we randomly choose from the protein data bank [2]. These results demonstrate that our meshing algorithm generates a correct and precise representation of the surfaces of molecules. We verify this by comparing the Betti numbers of the alpha shape and that of the surface. The topological features such as genus and tunnels are preserved and detail geometric feature like depressions and protrusions are approximated accurately. For example, Figure 5(b) shows the magnified view of a small genus and Figure 6(b) illustrates the magnified view of a cavity on the surface. The homeomorphism between the mesh and the original surface and the accurate approximation are due to the ε-sampling generated by the restricted union of balls. This also supports our arguments that the molecular skin model is a better geometric model for the molecules such as proteins and DNAs because it can enable us to achieve these two goals.

molecular name	no. triangles in the mesh	minimum angle in the mesh	computing time
$200D(Figure5)$	65,162	19.28°	00:01:35
$1FG1(Figure6)$	94,390	20.82°	00:02:41
$Helix(Figure1)$	98,017	21.20°	00:05:45
$HIV2(Figure2)$	226,758	20.35°	00:15:43

Table 1: Performance of our meshing algorithm.

Table 1 lists the statistics of the running time and mesh quality of

the surface meshes illustrated in this paper. All these experiments were run on a Pentium 4 PC. These results indicate that our implementation performs robustly and achievs good mesh quality with reasonable speed.

5 DISCUSSION

This paper describes an incremental mesh generation algorithm using restricted union of balls for triangulating the molecular skin surfaces. The restricted union of balls is the intersection of the skin surface and the union of a set of empty open balls centered at the surface sample points. By judiciously choosing the radius of the balls to adapt to the surface curvature, we obtain an even ε-sampling and the surface mesh is the restricted Delaunay triangulation of the sampling, which is guaranteed to be homeomorphic to the original surface. At the same time, we achieve guaranteed mesh quality and reasonable efficiency.

Compared with the sweeping skin meshing algorithm, our current meshing algorithm performed more robustly but had a lower efficiency because it computes the Delaunay triangulation of all the sample points on the surface. Although we improve the efficiency of locating the tetrahedron in the incremental construction of the Delaunay triangulation in a fast way, the edge flipping is still costly since the complexity of the Delaunay triangulation of surface sample points can be $O(n^2)$, as shown by Jeff Erickson[13]. Actually, any surface meshing algorithm using Delaunay triangulation of the sample points would face this problem and perform even worse because they have to use the DAG structure to locate the new vertex.

The mesh quality we achieved is worse than the previous methods, namely, the sweeping skin meshing algorithm and dynamic skin triangulation algorithm in a small scale. The main reason is that we do not use the costly refinement operations to improve the mesh quality. We can achieve a better quality by choosing smaller γ values. However, it would result in a large number of triangles in the mesh. Applying the refinement operations on our result would be an option to achieve better qualities. Indeed, after some minor refinement, our mesh will be as good as the results given by the previous method. However, we skip the refinement part in this paper because we aim to focus on the new way of mesh generation.

Our algorithm generates surface meshes and outputs the Delaunay triangulation of these sample points at the same time. An immediate future work would be the generation of quality volumetric tetrahedral mesh for the molecules on the base of our results. The Delaunay refinement algorithm in [8] would be a good choice. With the quality volumetric tetrahedral mesh, we can approximate the electrostatic potentials by solving the Poisson-Boltzmann equations using finite element methods. We believe the result would be more accurate and close to the experimental results because the mesh quality is an essential factor for the accuracy and convergence of finite element methods.

REFERENCES

[1] Alpha shape software http://biogeometry.duke.edu/software/alphashapes/.

[2] Protein data bank http://www.rcsb.org.

[3] Nataraj Akkiraju and Herbert Edelsbrunner. Triangulating the surface of a molecule 71 (1996), 5-22. *Discrete Appl. Math.*, 71:5–22, 1996.

[4] Jr. William V. Wright Amitabh Varshney Frederick P. Brooks. Linearly scalable computation of smooth molecular surfaces. *IEEE Computer Graphics and Applications*, 14(5):19–25, 1994.

(a)

(a)

(b)
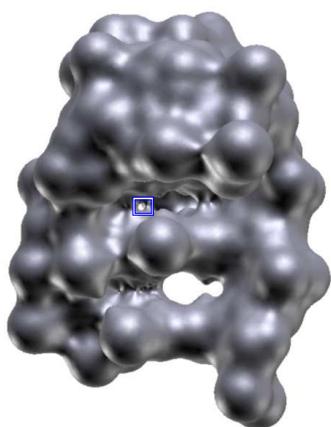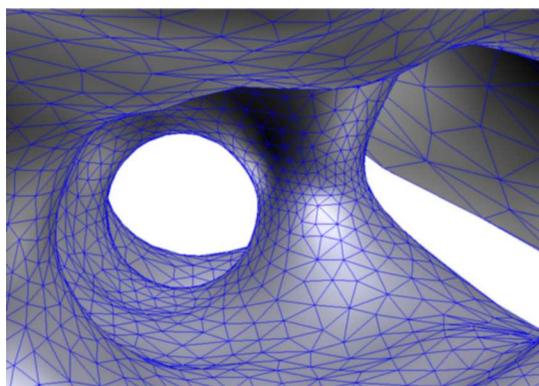
(b)

Figure 5: The molecular skin model of the molecule with PID:200D. Figure (a) shows its surface model and (b) illustrates the zoomed mesh details of a small genus in the box of (a).
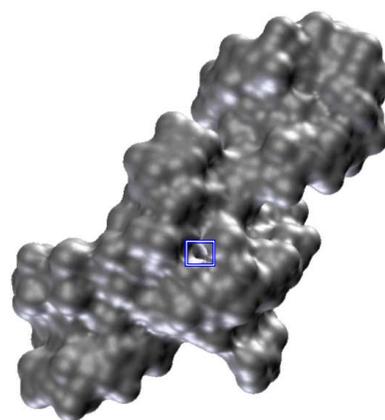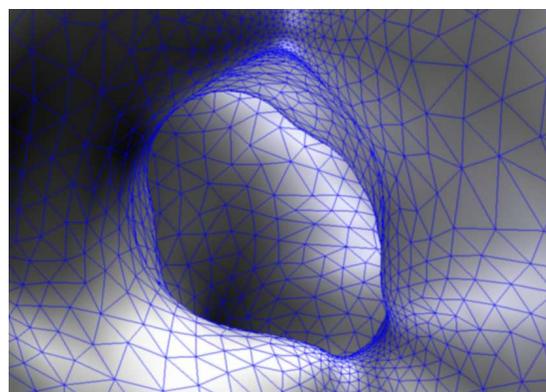
Figure 6: The molecular skin model of the molecule with PID:1FG1. Figure (a) shows its surface model and (b) illustrates the zoomed mesh details of a cavity in the box of (a).

[5] J. D. Boissonnat and S. Oudot. Provably good surface sampling and approximation. In *Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 9–18. Eurographics Association, 2003.

[6] H. Cheng, T. K. Dey, H. Edelsbrunner, and J.Sullivan. Dynamic Skin Triangulation. *Discrete Comput. Geom.*, 25:525–568, 2001.

[7] Ho-Lun Cheng and Xinwei Shi. Guaranteed quality triangulation of molecular skin surfaces. In *Proceedings of IEEE Visualization*, 2004.

[8] Siu-Wing Cheng and Tamal K. Dey. Quality meshing with weighted delaunay refinement. In *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 137–146. Society for Industrial and Applied Mathematics, 2002.

[9] Cecil Jose A. Delfinado and Herbert Edelsbrunner. An incremental algorithm for betti numbers of simplicial complexes on the 3-sphere. *Computer Aided Geometric Design*, 12(7):771–784, November 1995.

[10] H. Edelsbrunner. Deformable smooth surface design. *Discrete Computational Geometry*, 21:87–115, 1999.

[11] H. Edelsbrunner and N. Shah. Triangulating topological spaces. *Proceedings 10th ACM Symposium on Computational Geometry.*, pages 285–292, 1994.

[12] Herbert Edelsbrunner and Ernst P. Mucke. Three-dimensional alpha shape. *ACM Transaction on Graphics*, 13(1):43–72, January 1994.

[13] Jeff Erickson. Nice point sets can have nasty delaunay triangulations. In *Proceedings of the seventeenth annual symposium on Computational geometry*, pages 96–105. ACM Press, 2001.

[14] Patrick Laug and Houman Borouchaki. Molecular surface modeling and meshing. *Proceedings 10th International Meshing Roudable*, pages 31–41, Oct 2001.

[15] Charles L. LAWSON. Software for c^1 surface interpolation. *Mathematical Software III*, pages 161–194, 1977.

[16] Yongjie Zhang, Guoliang Xu, and Chandrajit Bajaj. Quality meshing of implicit solvation models of biomolecular structures. Ices technical report, University of Texas at Austin, 2004.