

Guaranteed Quality Triangulation of Molecular Skin Surfaces

Ho-Lun Cheng *

Xinwei Shi †

School of Computing
National University of Singapore

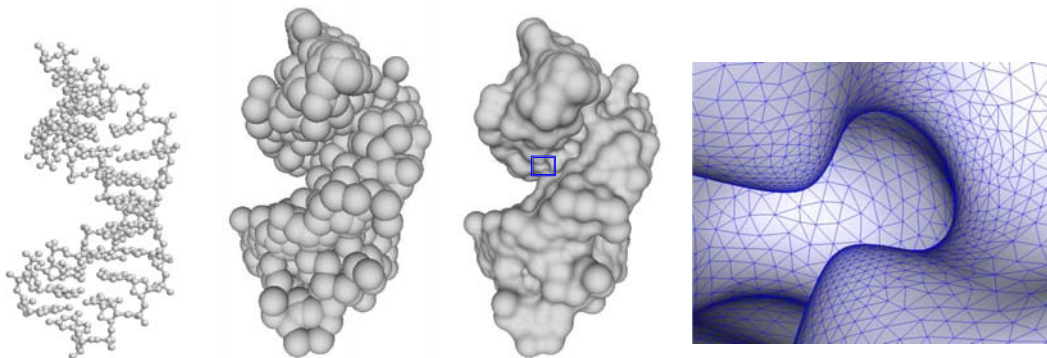


Figure 1: Molecular models of an A-DNA molecule. The leftmost shows the ball and stick model; the center left and right show the solvent accessible model and molecular skin model respectively; the rightmost shows the zoomed mesh details in the box of the center right figure.

ABSTRACT

We present an efficient algorithm to mesh the macromolecules surface model represented by the skin surface defined by Edelsbrunner. Our algorithm overcomes several challenges residing in current surface meshing methods. First, we guarantee the mesh quality with a provable lower bound of 21° on its minimum angle. Second, we ensure the triangulation is homeomorphic to the original surface. Third, we improve the efficiency of constructing the Restricted Delaunay Triangulation (RDT) of smooth surfaces. We achieve this by constructing the RDT using the advancing front method without computing the Delaunay tetrahedrization of the sample points on the surfaces. The difficulty of handling the front collision problem is tackled by employing the Morse theory. In particular, we construct the Morse-Smale complex to simplify the topological changes of the front. Our implementation results suggest that the algorithm decrease the time of generating high quality homeomorphic skin mesh from hours to a few minutes.

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, guaranteed quality triangulation, homeomorphism, Morse-Smale complex,

1 INTRODUCTION

Surface models are essential in many fields of computational science. For example, molecular surfaces are preferable in investigating the inter-molecular interactions such as the ligand docking. The

reason is that the surface model can represent the geometry of the molecules better than other models such as the ball and stick model. See Figure 1 as an example.

There are three existing molecular surface models [5]. The van der Waals surface (or VW) model is defined as the boundary of union of the atoms by considering each atom as a ball with its van der Waals radius. The other two surface models are defined through tracing a probe sphere that rolls over the VW model. The solvent accessible (or SA) model is the surface traced by the center of the probe sphere. The molecular surface (or MS) model is the surface traced by the inward-facing surface of the probe sphere. The MS model has a major advantage over the other two for its smoothness in most cases. Because smooth surfaces can be meshed with good quality triangles, MS model may facilitate accurate numerical computations. However, sharp corners still exist since the MS model can have self-intersections. These singularities result in unfaithful representations of the molecules and unrobust visualization software implementation [5].

Edelsbrunner [8] defines a paradigm of smooth surfaces for molecular modeling, namely, the *skin surface*. The skin model of a molecule is similar to its MS model. Moreover, it has a number of distinct properties such as smoothness, decomposability and complementarity and they are desirable in molecular modeling applications. In addition, the skin surface can deform freely with smooth transitions, which may be used for molecular dynamics simulations. Therefore, the skin surface model outruns the existing models for macromolecules such as DNAs and proteins.

However, there is still no efficient algorithm to mesh the skin surface for visualization and numerical computations. Although Cheng et al. [14] presented and implemented two algorithms, namely, a static and a dynamic skin triangulation algorithms, their performance are unsatisfactory. It will take hours to generate a skin surface of a protein molecule with about 1000 atoms. This paper proposes an efficient algorithm to mesh the skin surfaces.

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

†e-mail: shixinwe@comp.nus.edu.sg

1.1 Challenges in Surface Meshing

Surface meshing has been studied widely in the computer graphics and computational geometry literature [2, 4]. In order to support visualization and numerical simulation of surface models, the meshing algorithms are required to generate correct surface mesh with high triangle quality efficiently. However, several challenges still reside in existing methods. First, there should be provable bounds on the triangulation quality. Second, the output triangulation should be topological equivalent to the original surface. Both issues are important for the applications such as the numerical computations with finite element method (FEM). Finally, with the guarantees of mesh quality and topological correctness, the algorithm should be efficient and guaranteed to terminate.

Nevertheless, current works can not overcome all these challenges at the same time. The well-known marching cube algorithm [13] can achieve topological consistent implicit surface polygonization efficiently. However, it only ensure the triangulation is a manifold, which may not be homeomorphic to the original surface. On the other hand, Stander et al. [16] study the homeomorphic implicit surface polygonization by tracking the critical points of the implicit function using Morse theory. The surface reconstruction algorithm proposed by Amenta et. al [1] generate a homeomorphic mesh with sufficiently dense sample points. Nevertheless, the algorithm computes the Delaunay tetrahedrization of the sample points, which would be slow for large point set. Furthermore, the output mesh of all these works always includes bad shape triangles, that is, the triangles with extreme sharp or obtuse angles. Chew [4] proposes an algorithm to achieve guaranteed quality mesh for curved surfaces without addressing the topological equivalence problem.

1.2 Objective

Our goal is to design a skin triangulation algorithm to satisfy the following requirements:

- High efficiency;
- Good quality mesh;
- Homeomorphic mesh;
- Correctness and termination.

First, we aim to mesh the skin model of proteins within a few minutes on PC platforms. At the same time, the algorithm will guarantee the mesh quality with a lower bound on the minimum angle in the triangulation, that is, no angles in the mesh are less than 21° . Moreover, we can construct a homeomorphism between the triangulation and the skin surface. Finally, we require the algorithm is provable correct and terminates.

1.3 Our Approach

We approximate the skin surface by constructing its *restricted Delaunay triangulation* incrementally using the *advancing front method* without computing the Delaunay tetrahedrization of sample points. In particular, we use the *Morse-Smale complex* to handle the *front collision problem* in the advancing front meshing.

The advancing front method constructs the surface mesh by iteratively attaching triangles to the *front*, which is the boundary of partial triangulations. However, the newly added triangles may overlap existing ones as the front advancing and we call this the *front collision problem*. Conventional methods to handle this problem is time consuming because the potential collisions are detected by frequently checking the distance of any two vertices on the front [11]. Even though the collisions can be detected and the front stops

advancing at such regions, there will be cracks in the partial triangulations when no more triangles can be added. Fixing these cracks involves heuristics, which always lead to robustness problems of algorithms. Inspired by the ideas from *Morse theory*, We advance the front with the priority specified by a *Morse function* on skin surfaces. As a result, we can predict the potential front collisions and connect different parts of the front correctly according to topological changes of the front while it is advancing.

Nevertheless, difficulty arises for the topologically complex regions, that is, the front may change its topology in a sophisticated way corresponding to the small-scale details of the surface. With the recent results on the topological simplification [7], we compute the *Morse-Smale complex* to simplify the topological changes of the front. Finally, we can detect the front collisions efficiently and avoid cracks fixing. Our implementation suggests that our algorithm achieves high efficiency. The computation time of a skin surface decreases from hours to a few minutes.

Moreover, we achieve homeomorphic mesh with high quality by adapting the triangle size to the surface curvature. Specifically, we control the local size of the mesh to be proportional to the radius of the maximum principle curvature of the surface. Two conditions are maintained while attaching triangles and these conditions guarantee the mesh quality and homeomorphism to the surface. The operations of maintaining the conditions are proved correct and avoid infinite loops bouncing back and forth, which implies the correctness and the termination of the algorithm.

It should be noted that our application of Morse theory on skin triangulation differs from Stander et al. [16]. On one hand, we use a height function on the surface, a smooth 2-manifold, as the Morse function. Stander et al. use the function that defines the implicit surface as the Morse function, which could be considered as a height function on a 3-manifold. On the other hand, Stander et al. guarantee the homeomorphism between the mesh and the implicit surface by tracking the critical points of the implicit function. However, our homeomorphic triangulation is guaranteed by locally controlling the triangles size. Besides, a related concept to the Morse-Smale complex in our paper is the Reeb graph used in the surface reconstruction [10]. The Reeb graph is a compressed representation of the components of implicit surfaces. However, the Morse-Smale complex expresses the gradient flow on surfaces. We use the Morse-Smale complex of a height function to simplify the topological changes of the front.

Outline. Section 2 introduces the skin model of molecules and some geometric conditions of the skin triangulation. In Section 3, we handle the front collision problem by predicting the topological changes of the front according to Morse theory. We describe our algorithm in Section 4. Finally, we demonstrate some triangulation results in section 5 and conclude the paper with suggestions for the future works in Section 6.

2 BACKGROUND

In this section, we give a brief summary of the skin model of molecules. Readers who wish the details can refer to the works of Edelsbrunner [8]. Then, we present the conditions for constructing the homeomorphic skin triangulation with guaranteed quality, which are cited from Cheng et al. [3].

2.1 Molecular Skin

A skin surface is defined by a set of weighted points

$$B = \{b_i = (z_i, w_i) \in \mathbb{R}^3 \times \mathbb{R} \mid i = 1..n\},$$

where b_i is a weighted point with z_i as its position and w_i as its weight. The skin surface F_B defined by B is the boundary of an in-

finite family of spheres derived from B by convex combination and shrinking operation. Considering the shrinking factor as $\frac{1}{2}$, the skin surface is a closed C^1 continuous surface with continuous maximum principle curvature. It can be decomposed by the *mixed complex* M_B into a finite collection of quadratic patches, namely, sphere patches and hyperboloid patches. The mixed complex M_B is a collection of polyhedra that partition the \mathbb{R}^3 space. The portion of the skin surface in each polyhedron is a quadratic patch.

To model a molecule with the skin surface, we consider each atom as a weighted point $b_i \in B$. That is, the position z_i is the center of an atom, and its weight w_i is $\sqrt{2}$ times the summation of the atom's van der Waals radius with the radius of the probe sphere, which is usually chosen as 1.4 Angstrom to represent the water as solvent. Then, the skin surface F_B gives a model of the molecule. Figure 1 shows the SA model and the molecular skin model of a DNA molecule. The molecular skin uses hyperboloid to blend the spheres and forms a smooth surface model.

2.2 Skin triangulation

The *triangulation* of a skin surface is to form a simplicial complex whose underlying space is homeomorphic to the skin surface. We use the *restricted Delaunay triangulation* of a set of points, $T \subseteq F_B$, to approximate the skin surface, F_B .

Restricted Delaunay triangulation. Let $T \subseteq F_B$ be a finite subset of points on the skin surface. The *restricted Voronoi polygon* of $a \in T$ is defined as $v_a' = v_a \cap F_B$, in which v_a is the Voronoi cell of a with respect to T in \mathbb{R}^3 . The dual of the restricted Voronoi polygons is the *restricted Delaunay triangulation*, namely,

$$D_T' = \left\{ \text{conv}(U) \mid U \subseteq T, F_B \cap \bigcap_{a \in U} v_a \neq \emptyset \right\}.$$

Conditions. For a restricted Delaunay triangulation D_T' to be homeomorphic to the skin surface, F_B , the point set T has to satisfy the closed ball property, which requires the restricted Voronoi polygon is a topologically closed ball of appropriate dimension.[9]. Cheng et al. [3] propose to generate such a point set by adapting the sizes of the edges and triangles in D_T' to the local surface curvature. At any point $x \in F_B$, let $\kappa(x)$ be the maximum principle curvature at x . Call $\rho(x) = 1/\kappa(x)$ the local length scale at x . Then, the D_T' needs to satisfy two conditions,

$$[L] R_{ab} > \frac{C}{Q} \rho_{ab}, \text{ for every edge } ab,$$

$$[U] R_{abc} < CQ \rho_{abc} \text{ for every triangle } abc.$$

In Condition [L], notation R_{ab} is the size of an edge $ab \in D_T'$ which is equal to $\|ab\|/2$. Similarly, the size of a triangle $abc \in D_T'$ is defined as R_{abc} that is the circumradius of the triangle abc . The local length scale of the edge ab is defined as $\rho_{ab} = \max\{\rho(a), \rho(b)\}$ and that of the triangle abc is defined as $\rho_{abc} = \min\{\rho(a), \rho(b), \rho(c)\}$. C and Q are judiciously chosen positive constants. C controls how closely the mesh approximates the surface and Q controls the quality of the mesh.

Cheng et al. [3] proved that the restricted Delaunay triangulation D_T' has the closed ball property if these two conditions are satisfied with $C = 0.08$ and $Q = 1.65$. This implies D_T' is homeomorphic to the skin surface F_B [9]. At the same time, the minimum angle in D_T' is larger than $\arcsin \frac{1}{Q^2}$, which is around 21° for $Q = 1.65$.

In our algorithm, we will control the triangle sizes to maintain these two conditions while constructing the skin triangulation incrementally. As a result, we achieve homeomorphic skin triangulation with a lower bound on its minimum angle.

3 FRONT COLLISION HANDLING

We employ the advancing front method to construct the surface mesh. However, the *front collision problem* still arises. In this situation, the newly added triangles may overlap other existing triangles. We introduce Morse theory to handle this problem. Firstly, we introduce the *Morse function* and its relationship with the topological changes of the front. Then, we construct the *Morse-Smale complex* and utilize it to simplify the topological changes of the front.

3.1 Topological Changes of the Front

Morse theory on surfaces describes the topological changes of the partial surface boundary specified by a function, namely, the Morse function [15]. Firstly, we introduce the Morse function and its critical points on surfaces. Then, we describe their relationships with the topological changes of the front.

Morse function. We denote \mathbb{M} as a smooth, compact 2-manifold without boundary in \mathbb{R}^3 and a function $f : \mathbb{M} \rightarrow \mathbb{R}$. We assume a local coordinate system (x_1, x_2) in a neighborhood of a point $p \in \mathbb{M}$. The point p is a *critical point* of function f if all its partial derivatives vanish with respect to the local coordinate system, that is, $\frac{\partial f}{\partial x_1}|_p = \frac{\partial f}{\partial x_2}|_p = 0$. Otherwise, it is a *regular point*. If p is a critical point, $f(p)$ is the *critical value* of f at p . The critical point p is *non-degenerate* if the *Hessian* of f at p , $H(p)$, which is the matrix of the second derivatives, is non-singular, namely, $\det H(p) \neq 0$. The function f is called a *Morse function* if all its critical points are non-degenerate.

For a critical point $p \in \mathbb{M}$, we can choose an appropriate local coordinate system (x_1, x_2) in the neighborhood of p such that the Morse function f is expressed in the form $f(x_1, x_2) = \pm x_1^2 \pm x_2^2 + f(p)$. The number of minuses is called the *index* of f at p . Thus, there are three types of critical points on a smooth 2-manifold in \mathbb{R}^3 , namely, minima with index 0, saddle points with index 1 and maxima with index 2.

In this paper, we employ the height function $h(p) = y_p$ on \mathbb{M} as the Morse function if $p = (x_p, y_p, z_p)$. The critical points of h are the points with horizontal tangent planes.

Topological changes of the front. Let $\mathbb{M}_a = \{x \in \mathbb{M} \mid h(x) \leq a\}$ be the partial surface for some $a \in \mathbb{R}$. Denote $L_a = \{x \in \mathbb{M} \mid h(x) = a\}$ as the *level curve* at a . The curve L_a is the boundary of \mathbb{M}_a and it is the intersection of \mathbb{M} and the horizontal plane $\mathbb{T}_a : y = a$.

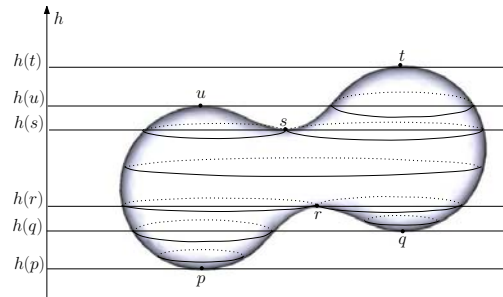


Figure 2: Critical points and level curves of height function h on a smooth 2-manifold. The points p, q are minima, r, s are saddle points and t, u are maxima of h . Solid curves are the level curves at different height.

We illustrate the topological changes of the level curve L_a at the three types of critical points of h with the example in Figure 2. Let ε be a small positive number. While we sweep a horizontal plane \mathbb{T}_a upwards from $a = -\infty$, the partial surface, \mathbb{M}_a , is empty

when $a < h(p)$. After \mathbb{T}_a passes the minimum p and when $a = h(p) + \varepsilon$, the level curve L_a changes from $L_{h(p)-\varepsilon} = \emptyset$ to $L_{h(p)+\varepsilon}$, which is a topological circle. When the horizontal plane rises to $\mathbb{T}_{h(r)}$, the two topological circles in $L_{h(r)-\varepsilon}$ touch at r . At the level $h(r) + \varepsilon$, the curve $L_{h(r)+\varepsilon}$ becomes one topological circle. As the height increases to $h(s)$, $L_{h(s)}$ changes to two topological circles with their contact point at s . When the horizontal plane \mathbb{T}_a arrives at the maximum u , one topological circle converges to the point u . Finally, $L_{h(t)+\varepsilon}$ becomes empty after the horizontal plane passes the maximum t .

By this observation, we are inspired to sweep the partial triangulation over the skin surface in a similar manner. The topological changes of the level curve correspond to the topological changes of the front. Therefore, the front will collide around the critical points on the skin surfaces. We set a *protecting sphere* with the center at each critical point to detect the potential collisions of the front. Thus, we only manage the topological changes of the front within each protecting sphere. There are three kinds of topological changes corresponding to the three types of critical points, namely, “creation” at the minima, “bridge” at the saddle points and “seal” at the maxima. For the radii of the protecting spheres, we will discuss them in Section 4.2.

However, two critical points may be close to each other and their protecting spheres have non-empty intersection. More sophisticated solutions are needed in this case. One possible solution is to decrease the size of all the triangles in the triangulation. The reason is that the radii of the protecting spheres decrease as the triangle sizes decrease. We can reduce the triangle sizes until no two protecting spheres intersect. However, it results in huge number of triangles, which would decrease the efficiency of the algorithm and increase the storage.

On the other hand, we can remove the critical points that are close to each other without influencing the construction of homeomorphic triangulation. We will use the Morse-Smale complex to remove these critical points that are close to each other in the next section.

3.2 Simplification

The critical points that are close to each other can be interpreted as the *noisy critical points*, which can be eliminated if they are connected by an arc in the Morse-Smale complex [7].

Firstly, we describe the structure of the Morse-Smale complex. Then, we describe how to construct the Morse-Smale complex on skin surfaces. Finally, we show how to simplify the topological changes of the front by eliminating the noisy critical points.

Morse-Smale complex. The critical points of the height function are connected in the Morse-Smale complex via the *integral lines*. An integral line, γ , is a curve on \mathbb{M} . For each point $p \in \gamma$, its tangent vector on γ agrees with the gradient of the height function h at p . The *gradient* of the height function h at a point p can be defined as $\nabla h = (\frac{\partial f}{\partial x_1}(p), \frac{\partial f}{\partial x_2}(p))$ in an orthogonal local coordinate system (x_1, x_2) . The gradient vanishes at the critical points. For a regular point, the gradient is the tangent vector in the steepest ascending direction.

The Morse-Smale complex is a collection of vertices, arcs and regions. Each vertex is a critical point, and each arc is an integral line connecting a saddle point to a minimum or a maximum critical point. The arcs divide the manifold \mathbb{M} into quadrangular regions. Each region is surrounded by two saddle points, a minimum and a maximum critical point [6]. See Figure 10(a) for an example.

Construction. The first step to construct the Morse-Smale complex of h on the skin surface F_B is to locate all the critical points. In

the skin decomposition, the critical points can be easily computed analytically by exploring every single sphere or hyperboloid patch.

Next, we compute the arcs in the Morse-Smale complex, which are the integral lines connected to the saddle points. At each saddle point, there are two integral lines of the steepest ascent and another two of the steepest descent. First, we compute the tangent vectors of each saddle point that indicates the steepest ascending and descending directions. Because each saddle point lies on an one-sheeted hyperboloid, we can get the vectors analytically by computing the tangent vectors of a vertical cross section of the hyperboloid.

Then, we compute the gradient of the height function h at the regular points. For each regular point $p \in F_B$, we denote the gradient vector of the height function h at p as g_p and the unit surface normal vector of p as n_p . The gradient vector is $g_p = \frac{V}{n_p \cdot V} - n_p$, in which the unit vector $V = (0, 1, 0)$ represents the height direction.

Finally, we approximate the integral lines with piecewise linear curves because the integral lines on the skin surface are difficult to compute analytically. At each saddle point, starting from the tangent vectors indicating the steepest directions, we trace four integral lines by iteratively stepping forward following the gradient until we meet a maximum or a minimum critical point. The accuracy of the approximation is guaranteed by choosing sufficiently small step sizes adaptive to the surface curvature. The Figure 10(a) illustrates an example of the Morse-Smale complex.

Elimination of the noisy critical points. For the critical points whose protecting spheres intersect each other, we remove them by contracting the short arcs in the Morse-Smale complex. An arc is contracted through deleting its two ending critical points and re-connecting their neighbors. We describe the contraction of an arc with the example illustrated in Figure 3.

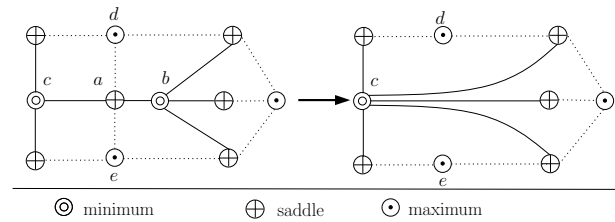


Figure 3: The contraction of the ab in the Morse-Smale complex.

Let a be a saddle point and b be a minimum critical point. Suppose that the arc ab is a short ascending integral line from b to a . We can view the contraction of the arc ab as merging critical points a, b into another minimum critical point c that connected to a . The four integral lines connected to a are removed and all the integral lines start from b are extended to c . After the contraction of ab , each region is still a quadrangle with minimum, saddle, maximum and saddle in order. The short arcs between the saddle points and maxima can be contracted in a similar way.

Contraction of the short arcs between a saddle point and a minimum or a maximum critical point is the only operation required in eliminating the noisy critical points. With a sequence of contraction operations, it is sufficient to get a simplified Morse-Smale complex in which no two protecting spheres intersect each other [6].

4 ADAPTIVE SWEEPING TRIANGULATION

When the front is advancing, the new triangles are always attached to the lowest vertex on the front, which is called *departure vertex*. Thus, the partial triangulation sweeps from the bottom to the top of the skin surface until the whole surface is covered. In each step of

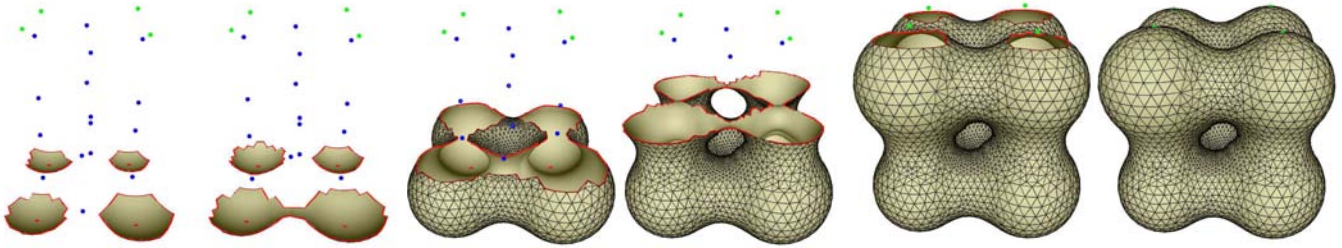


Figure 4: Six snap-shots of the growing mesh from the bottom to top of the skin surface defined by 8 weighted points at the corners of a cube. The red points are the minima, the blue points are the saddle points and the green points are the maxima; The red portion of the partial mesh is the front.

adding new triangles, their sizes are adaptive to the surface curvature and the triangulation is guaranteed to be homeomorphic to the surface.

We first give a top-down description of the algorithm. Then, we discuss the curvature adaptation schema to control the sizes of triangles. Refinement operations that maintain the closed ball property are introduced at the end of the section.

4.1 Algorithm

The input of the algorithm is a set of weighted points, B , and the output is a triangulation, K , which approximate the skin surface F_B . The algorithm is divided into two stages, namely, initialization stage and sweeping triangulation.

Initialization. In this stage, we construct two combinatorial structures, namely, the mixed complex M_B of the set of weighted point B and the Morse-Smale complex of the height function h on the skin surface F_B . These two complexes guide the mesh generation in the next stage. We first compute the mixed complex, M_B , of B by constructing the weighted Delaunay triangulation of B . For each polyhedron μ in M_B , it is the Minkowski sum of a simplex in the weighted Delaunay triangulation of B and its dual Voronoi cell, scaled by $1/2$. Then, we locate all the critical points on each quadratic patch within μ . Finally, we compute the Morse-Smale complex.

We set a protecting sphere for each the critical point according to their curvature, which will be described in Section 4.2. If two protecting spheres intersect, we eliminate the critical points by contracting the short arcs in the Morse-Smale complex.

Sweeping Triangulation. Next, we add triangles to the mesh incrementally. The partial triangulations sweep over the skin surface along the height direction. Figure 4 demonstrates six snap-shots of the growing mesh of a skin surface. The sweeping triangulation stage is divided into three steps, namely, *creating the initial front*, *creeping triangles* and *sealing holes*, and they are implemented according to the topological changes of the front at the three types of critical points, namely, minima, saddle point and maxima respectively.

First, we create the initial front at the minima with the “creation” operations. Then we advance the front along the height direction by attaching triangles to the lowest vertex on the front iteratively. During this process, the different portions of the front connect around the saddle points within its protecting sphere, and we call this the “bridge” operation. When the front is in the protecting spheres of the maxima and no more triangles can be attached, we close the holes with the “seal” operation and the algorithm terminates.

Initial front. We first construct the initial front by applying “creation” operation at each minimum critical point. The “creation” operation creates a “bowl” of 6 neighboring triangles that share one common vertex. See Figure 5.

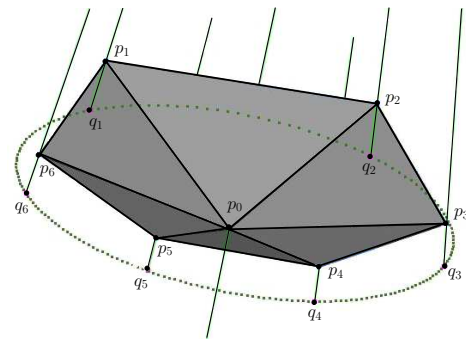


Figure 5: Create a bowl at the minimum point p_0 .

Let p_0 be the minimum critical point. We first draw a tangent disk with p_0 as the center. Then we locate six points $q_i, i = 1..6$ on the boundary of the tangent disk. See the dashed circle in Figure 5. These six points divide the tangent circle into six arcs uniformly, that is, $\angle q_i p_0 q_{i+1} = 60^\circ$. We add the new vertices p_i by projecting q_i to the skin surface and form the six triangles. The boundary of each bowl is a cycle of 6 edges, which is a *front polygon*. After applying *creation()* operation to all minima, we get the initial front, which consists of a set of front polygons. We push all the edges in the front polygons to a priority queue, Q , which enables us to get the departure vertex with the minimum height in the next step.

Creeping triangles. In this step, we add more triangles iteratively from the initial front. There are two operations in this process: the *creep()* operation to attach triangles to the front from a departure vertex, and the *bridge()* operation to connect two portions of the front around the saddle points. The step of creeping triangles is implemented with *CreepTriangles()*, which is illustrated in the following pseudo code.

In each iteration, we get the departure vertex from Q to attach new triangles. A fan of triangles around the departure vertex are created by the *creep()* operation. See Figure 6 as an example. Let p_t be the departure vertex. We first draw the tangent disk with p_t as the center. The dashed circle in Figure 6 is the boundary of the tangent disk. Let p_0, p_n be the neighbors of p_t on the front and the points q_0, q_n are the projections of p_0, p_n on the tangent plane of p_t . We add new vertices p_i and form new triangles $p_{i-1} p_t p_i$ from $i = 1$ to $i = n$, in which $n = \lceil \frac{\angle q_0 p_t q_n}{60^\circ} \rceil$. To get p_i , we first locate q_i that satisfies $\angle q_i p_t q_{i+1} \simeq 60^\circ$. Then, we project each q_i to a point p_i on the skin surface. After each *creep()* operation, we push the

Algorithm 1 CreepTriangles()

```

1: while  $Q \neq \emptyset$  do
2:    $t = \text{ExtractMin}(Q)$ ;
3:   if  $t$  falls in the protecting sphere of a saddle point  $c$  then
4:     if  $c$  is ready to be bridged then
5:       bridge();
6:     else
7:       Mark the triangle  $t$  with red;
8:     end if
9:   else if  $t$  falls in the protecting sphere of a maximum point  $c$  then
10:    Mark the triangle  $t$  with green;
11:   else
12:     creep();
13:   end if
14:   Refine the newly added triangles;
15: end while
  
```

new front edges to the priority queue Q . While the priority queue Q is not empty, we repeat the *creep()* operation to advance the front towards the untriangulated part of the surface.

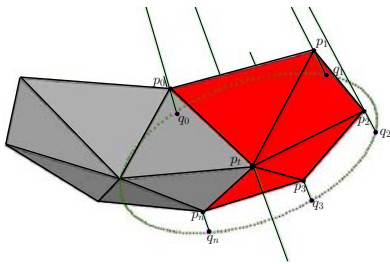


Figure 6: Creep triangles from a departure vertex p_r .

During the front advancing by the *creep()* operation, the small angle between two front edges may cause overlap between the newly added triangles and the existing triangles. See the Figure 7 (a).

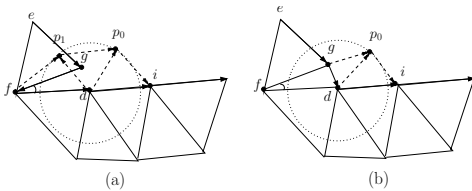


Figure 7: Wing a small angle to avoid the overlapping triangles in *creep()* operation. Figure (a) and (b) illustrate the result of *creep()* operation before and after fixing the small angle $\angle gfd$.

In Figure 7, the angle $\angle gfd$ is a small angle and the vertex d is the departure vertex. The dashed circle is the boundary of the tangent disk in the current *creep()* operation. In the Figure 7 (a), two newly added mesh vertices are p_0 and p_1 . Because $\angle gfd$ is small, the vertex p_1 lies in the existing triangle efg . As a result, the new triangles fp_1d and p_1p_0d overlap the existing triangle efg . We can solve this problem by fixing the angle $\angle gfd$ before we apply *creep()* to vertex d . As illustrated in the Figure 7 (b), the newly added triangles from d do not overlap the existing ones after adding a triangle gdf .

We call fixing small angles the *wing()* operation and apply it at the end of each *creep()* operation to fix all the small angles. It en-

sures that there are no angles smaller than 90° between any two adjacent front edges during front advancing.

However, the newly added triangles in the *creep()* operation may not maintain the closed ball property in Section 2.2. We need refinement operations to maintain the Condition [L] and Condition [U] for the closed ball property. We will introduce the refinement operations in Section 4.3.

Bridging the front. As the partial mesh grows, the fronts will get closer to each other around the saddle points or the maxima. The protecting spheres for the critical points will detect such events. Once the front falls into the protecting sphere of a saddle point, we stop the front advancing at this portion because it would lead to front collision. If the front reaches the protecting sphere of a saddle point on two different sides, we use a *bridge()* operation to connect the two sides of the front. Figure 8 illustrates the result a bridge operation.

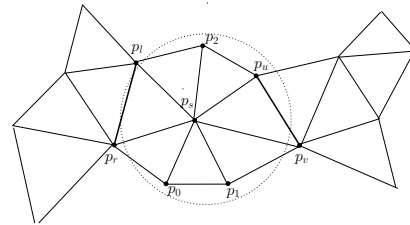


Figure 8: Bridge the front at a saddle point p_s .

In Figure 8, the vertex p_s is a saddle point and the thin dotted line represents the protecting sphere of p_s . The front edges $p_l p_r$ and $p_u p_v$ fall in the protecting sphere on two different sides of p_s . We add new triangles to bridge the front together at the edges $p_l p_r$ and $p_u p_v$. Then, we update the priority queue Q after a *bridge()* operation and resume the loop of *creep()* operation to advance the front. See the snapshots number 2, 3 and 4 in Figure 4 for an example.

Sealing holes. Once the front falls into the protecting sphere of a maximum critical point, we simply stop the front advancing at that portion. When the priority queue Q is empty, the surface, F_B , is covered by an almost finished partial triangulation with holes only around maxima. The boundary of each hole is a topological circle consisted of a cycle of edges in the protecting sphere of a maximum. We seal the hole by repeatedly adding triangles whose two edges are neighbors in the circle. The snapshots number 5 in Figure 4 illustrates the holes around each maximum before the sealing operation.

Finally, we get the triangulation K of the skin surface F_B after refining the newly added triangles in these *seal()* operations.

4.2 Curvature Adaptation

In the *creation()* and *creep()* operations, we obtain new mesh vertices by projecting the points from a tangent circle to the surface. In order to guarantee the closed ball property, we need to adjust the radius of the tangent disk to ensure that the new triangles satisfy the two Conditions in Section 2.2. Also, we use the same radii for the protecting spheres at the critical points. The reason is that each step of the front advancing must not cross to the other side of the spheres.

Edge length constraints. Let the departure vertex be a and a new mesh vertex be b . Denote the radius of the tangent disk as r_a centered at a . Our consideration is to ensure the edge length of ab is not too long nor too short for both vertices a and b according to the

two Conditions in Section 2.2. We first derive the lower bound and upper bound of the size of edge ab , R_{ab} , relative to the local length scale $\rho(a)$.

Cheng et al. [3] prove that the rate of changes of the local length scale on the skin surface satisfy the following Lipschitz condition,

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

in which x, y are two points on the skin surface F_B . From this condition, we can determine the bound of $\rho(b)$ in terms of $\rho(a)$. As a result, we get the lower bound and upper bound of the size of edge ab according to the two Conditions, that is,

$$\frac{C}{Q-2C}\rho(a) \leq R_{ab} \leq \frac{CQ}{1+2CQ}\rho(a).$$

The inequality shows that we need a slightly longer edge than the edge length in Condition [L] in order to make sure the edge is long enough for both vertices a and b . With the edge length constraints, we can determine the radius of the tangent disk.

Radius of the tangent disk. To ensure the edge is not too short, we can use the bound of $r_a > \frac{2C}{Q-2C}\rho(a)$ since $R_{ab} > r_a/2$. For the upper bound of r_a , we know that R_{ab} increases when b gets further from the tangent disk. However, there is a *sandwiching ball* with radius $\rho(a)$ for every vertex a [14]. The sandwiching ball is tangent to the skin surface at the point a and the skin surface does not penetrate into this ball. Figure 9 shows the cross section of the skin surface and the sandwiching ball at a .

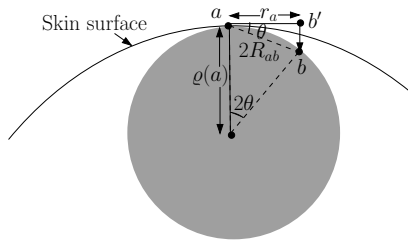


Figure 9: Cross section of the scenario of projecting the point from the tangent disk with radius r_a .

Then, the longest edge will be created if b is on the surface of the sandwiching ball of a . In Figure 9, we compute the angle $\theta = \arcsin(R_{ab}/\rho(a))$ and thus

$$\begin{aligned} r_a &= 2R_{ab} \cos \theta \\ &= 2R_{ab} \sqrt{1 - \left(\frac{R_{ab}}{\rho(a)}\right)^2} \end{aligned}$$

Substituting the upper bound of $R_{ab} = \frac{CQ\rho(a)}{1+2CQ}$ into the above equation, the upper bound of the tangent disk radius is

$$r_a = \left(\frac{CQ}{(1+2CQ)^2} \sqrt{(1+CQ)(1+3CQ)} \right) \cdot \rho(a).$$

For $C = 0.08$ and $Q = 1.65$, the numerical value of $r_a/\rho(a)$ is around 0.208, which is better than the lower bound $\frac{2C}{Q-2C} = 0.107$.

4.3 Local Refinement

The curvature dependent radius of the tangent disk r_a ensures that the edge length is not too long nor too short in most cases. However,

the partial triangulations may not maintain the closed ball property during sweeping. It is mainly due to three reasons. Firstly, the local region of newly added triangles in each step may not keep the restricted Delaunay property, we use *edge flipping* to maintain the restricted Delaunay triangulation. Secondly, the short edges added in the *wing()* operation may violate the Condition [L], we use the *edge contraction* to remove these short edges. Finally, edge contraction may lead to violation of the Condition [U]. Thus, we need the *vertex insertion* operation to destroy the big triangles.

We refer to the implementation of the refinement operations introduced in the dynamic skin triangulation algorithm [3]. After each step of adding new triangles, these refinement operations are applied. As a result, we maintain the closed ball property for the sweeping mesh. Cheng et al. proved that these refinement operations terminate, which implies the termination of our algorithm. Experiments with our implementation show that the refinement operations only influence a very small number of triangles around the new attached ones in each *creep()* operation. It means the refinement operations are efficient. At the same time, these refinement operations guarantee the minimal angle in the triangulation is larger than 21° , which implies a good quality mesh.

5 RESULTS

We demonstrate the skin meshes constructed by our algorithm in this section. Figure 10(a) shows the surface of a simple molecule named “pdb7tmn”, which is a binding inhibitor of protein molecule. Figure 11(a) shows the molecular skin surface model of “Gramicidin A” molecule and Figure 11(b) illustrates its partial mesh details included in the box of Figure 11(a). Table 1 lists the statistical results of the examples shown in this paper, along with a comparison of computation time with the dynamic skin triangulation algorithm [3]. A Pentium 4 processor running at 2.54GHZ is used in the test. It is easily noticed that our approach can greatly improve the efficiency. Moreover, we achieve a better bound on the minimum angle of the triangulation than the theoretical result on the bound of the minimum angle, that is, 21° .

molecular name	no. of triangles in the mesh	minimum angle in the mesh	computing time	
			our approach	Dynamic skin
<i>pdb7tmn</i>	24,336	27.02°	00:00:05	00:10:00
<i>A-DNA</i>	114,316	24.12°	00:00:51	00:35:12
<i>GramicidinA</i>	305,186	24.37°	00:03:22	01:35:23

Table 1: Performance of our meshing algorithm.

The dramatic improvement of the performance over the dynamic skin triangulation depends on the efficient solution of the front collision handling and avoidance of computing 3D tetrahedrization of the sample points. Moreover, the computation time is almost linear to the number of the triangles in the triangulation since it takes constant time in each iterative step of attaching triangles. The strategy of the initial points placement and the refinement operations can explain the achievement of high triangle quality.

6 DISCUSSION

This paper introduces a new method for visualizing the macromolecules such as proteins and DNAs. We construct high quality mesh of the molecular skin models with guarantees on the topology and a lower bound of the minimum angle. The surface triangulation can support not only the protein visualization but also numerical simulations of the protein interactions. We achieve high efficiency by employing the Morse theory to handle the front collision problem in advancing front method.

The application of skin surfaces is not limited in modeling the molecules. Theoretically, every orientable closed surface has a skin representation [8]. The Figure 10(b) illustrates a human face modeled by the skin surface. Kruihof et.al [12] had stepped toward using the skin surface to approximate an arbitrary polygonal surface.

The choice of the height function in our paper might be questionable as the resulting critical points and Morse-Smale complex are obviously dependent on the choice of the height direction. However, our algorithm can generate correct mesh with different directions of the height function. Another issue is the number of the critical points. Since a large number of critical points may affect the efficiency of our algorithm, we tackle this by using an efficient point location data structure, namely, the *kd-tree*. On the other hand, it is possible to use other Morse functions instead of the height function and we change the priority of the front advancing accordingly.

We also propose to extend our algorithm to triangulate other smooth compact 2-manifolds, for example, implicit surfaces. A main challenge for this extension lies in finding the homeomorphism conditions to determine the curvature adaption schema. We are also interested in extending our algorithm for the triangulation of parametric surfaces such as Nonuniform Rational B-Splines (NURBS).

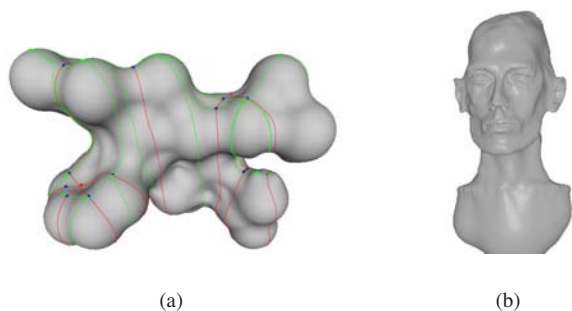
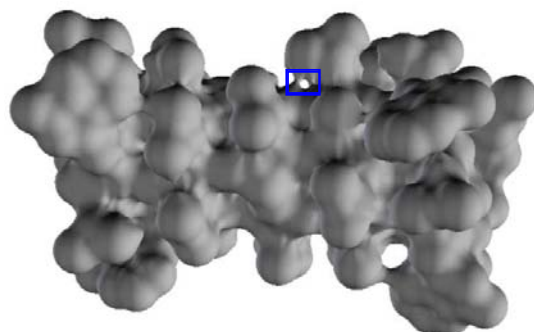


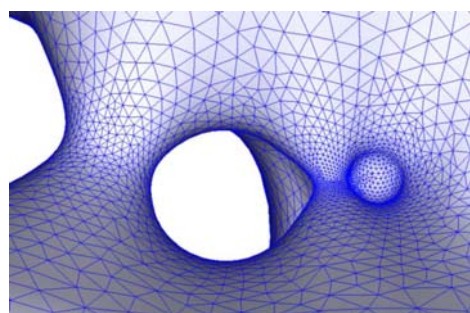
Figure 10: Skin surface models. Figure (a) illustrates the molecular skin model of “pdb7tmn” and its Morse-Smale complex before simplification. The red and green curves are the integral lines connecting minimum-saddle pairs and maximum-saddle pairs respectively. Figure (b) shows a human face modeled by the skin surface.

REFERENCES

- [1] N. Amenta, S. Choi, T. K. Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. In *Proceedings of the sixteenth annual symposium on Computational geometry*, pages 213–222. ACM Press, 2000.
- [2] Jules Bloomenthal. *Introduction to Implicit Surface: Surface Tiling*. Morgan Kaufmann, 1997.
- [3] Ho-Lun Cheng, Tamal K. Dey, Herbert Edelsbrunner, and John Sullivan. Dynamic skin triangulation. In *Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 47–56. Society for Industrial and Applied Mathematics, 2001.
- [4] L. Paul Chew. Guaranteed-quality mesh generation for curved surfaces. In *Proceedings of the ninth annual symposium on Computational geometry*, pages 274–280. ACM Press, 1993.
- [5] Michael L. Connolly. Molecular surfaces: A review. <http://www.netsci.org/Science/Compchem/feature14.html>, 1996.
- [6] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical Morse-Smale complexes for piecewise linear 2-manifolds. *Discrete Comput. Geom.*, 30:87–107, 2003.



(a)



(b)

Figure 11: The molecular skin model of “Gramicidin A”. Figure (a) shows its surface model and (b) illustrates the zoomed mesh details of the portion in the box of (a).

- [7] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete Comput. Geom.*, 28:511–533, 2002.
- [8] Herbert Edelsbrunner. Deformable smooth surface design. *Discrete Computational Geometry*, 21:87–115, 1999.
- [9] Herbert Edelsbrunner and Nimish R. Shah. Triangulating topological spaces. In *Proceedings of the tenth annual symposium on Computational geometry*, pages 285–292. ACM Press, 1994.
- [10] Anatolij T. Fomenko and Toshiyasu L. Kunii. *Topological Modeling for Visualization*. Springer-Verlag Tokyo, 1997.
- [11] Erich Hartmann. A marching method for the triangulation of surfaces. *The Visual Computer*, 14:95–108, 1998.
- [12] Nico Kruihof and Gert Vegter. Approximation by skin surfaces. In *Proceedings of the eighth ACM symposium on Solid modeling and applications*, pages 86–95. ACM Press, 2003.
- [13] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169. ACM Press, 1987.
- [14] Ho lun Cheng. *Algorithm for Smooth and Deformable Surfaces in 3D*. PhD thesis, UIUC, 2002.
- [15] Yukio Matsumoto. *An Introduction to Morse Theory*. American Mathematical Society, 2002.
- [16] Barton T. Stander and John C. Hart. Guaranteeing the topology of an implicit surface polygonization for interactive modeling. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 279–286. ACM Press/Addison-Wesley Publishing Co., 1997.