# Robust Triangulation Algorithm for Geometry with Acute Angle Features

Xianhai Meng and Guoyi Mei

***Abstract*—A new algorithm is presented to automatic generate conforming Delaunay triangulation of non-manifold geometric domains with acute angle features. The algorithm is based on Delaunay refinement technique, which often failed to terminate when there are small angles in input geometry. By assigning proper weights to vertices on sharp-angled elements and take place Delaunay triangulation with weighted Delaunay triangulation, the algorithm can accept any inputs without any bound on angle and without setting any protected area and adding any new vertices near the sharp-angled elements. The algorithm also guarantees bounded circumradius to shortest edge length for all elements except the ones near small input angles. The proof of terminator and some computation results are also presented.**

***Index Terms*—Algorithm, computational geometry, triangulation, conforming delaunay triangulation, weighted vertex, acute angles.**

## I. Introduction

Conforming Delaunay triangulation (CDT) is an important topic both in theoretically computational geometry and in practical applications of finite element method, scientific visualization, geometric modeling etc. Now most CDT algorithms are based on local transformations like edges/faces swapping (or flipping) [1-3] or Delaunay refinement technique [4-5]. For the several advantages, such as easy to state and implement, can offer good mathematical guarantee on mesh quality, and convenient for being extended to high dimensions, the Delaunay refinement technique become more and more popular both in theory and in practice.

Chew[4] and Ruppert[5] first present the Delaunay refinement technique in 2D in early of 1990's, which is used to mesh an input PSLG (*Planar Straight Line Graph*)[6] in 2D with Delaunay triangles, and Shewchuk[7] extended it to input PLC (*Piecewise Linear Complex*) [7] in 3D. But, there are still some shortcomings for Delaunay refinement. First, above Delaunay algorithms works only with input domain where no angle is smaller than a bounded value. Second, in 3-dimension, a kind of poor quality tetrahedron called "Sliver" couldn't be eliminated by bounded radius-ratio, which is the ratio of element's circumradius to its smallest edge length and can be guaranteed in Delaunay refinement. Especially, the first problem limited the applicability of the Delaunay refinement technique.

So, for this reason, some remarkable efforts have been done to handle the input small angle for Delaunay refinement. In the planar case, Ruppert[5] provided two heuristic methods, one is a preprocessing step called "corner-lopping" to isolate small angle, and another is called "concentric circular shells" which split segment at its vertex's concentric circular shells whose radius are power of two. Based on above concentric circular shells, Shewchuk[9] presented an alternation called "Terminator", which produces the output mesh has most angles are greater than $arcsin\left(1/2\sqrt{2}\right)$, has poor quality triangle only in the vicinity of small input angles and also offers guaranteed termination. Then, Miller *et al*[10] showed an "adaptive" version of Ruppert's algorithm with concentric shell splitting, the algorithm outputs meshes where all output angles are greater than $26.45^{\circ}$, except those whose shortest edge subtends an small input angle, and also has termination and grading guarantees. In those algorithms, all the segments incident to the endpoints that bear the small input angle need to be split to power of two lengths and when a new inserted point encroached those segment, some rules on splitting encroached segment are applied due to the relative position of point to the endpoint. Furthermore, Yang Qin[11] also presented an alternative method based on setting control circle around the small input angles, which gives a bounded angle about 20.7º except for the small input angles .

In 3D case, Shewchuk[8] first proposed the directional object for solving this problem and presented a method for non-manifold input without guarantees on mesh quality. Murphy[12] also presented a algorithm claimed for any PLC input, but it produces too many additional points around the small input angles, then Cohen-Steiner *et al* [13] improved it but without any quality guarantee. Finally in 2004, Steven Pav *et al*[14] keep on improving that algorithm and presented an good Delaunay refinement for non-manifold inputs. In addition, S.W.Cheng[15] has ever presented a significant theoretical improved algorithm for small angles, but it is hard to implement, so, a simplified version of this algorithm[16] is given with constant bounded radius-edge ratio and some implemented examples, but it handles only 2-manifold boundaries. Yang Qin [11] also proposed a improvement method on 3D case for any input angles with good guarantee for termination.

In all above algorithm in 3D, A similarly method can be found. They all need to construct a protecting region around the vertices and the edges where exist the small input angle, such as Cohen-Steiner's protected area, S.W.Cheng's buffer zone and Yang Qin's control column *etc.*, and when a new

Manuscript received March 18, 2012; revised March 28, 2012. This work was supported in part by the Fundamental Research Funds for the Central Universities of China under grant No.YWF-10-02-057 and the Natural Science Foundation of China with No. 61003110.

Xianhai Meng and Guoyi Mei are with School of Computer Science & Engineering, Beihang University, 100083, Beijing, China (e-mail: mengxh@ nlsde.buaa.edu.cn; meiguoyi@ nlsde.buaa.edu.cn).

point is added inside those protecting regions, some new splitting rules which is different from traditional Delaunay refinement are applied to insure conformality of the mesh to the input. Those new splitting rules for protecting region around small input angle not only complicate the implement of those algorithms and take more time consume, also import instability to algorithms especially when they are used for complex non-manifold PLC inputs. The later problem is also the reason of why there is only one effective example for their algorithms. So, now, how to mesh 3D domain with small angles, especially with complex non-manifold geometry, and with guaranteed quality and without adding many additional points is still an open question.

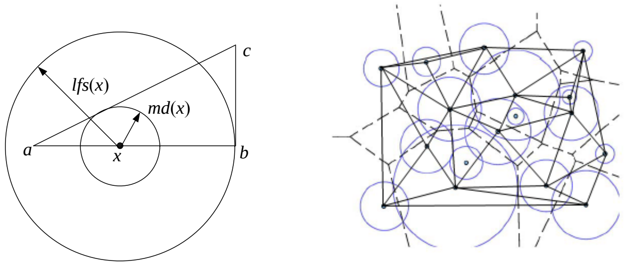
In this paper, we present a new algorithm for non-manifold domains with small angles. The algorithm is also based on the Delaunay refinement technique. But, it just simply assign some weight value to vertices that located on the sharp-angled input, and replace the origin Delaunay refinement algorithm with weighted Delaunay refinement. And we also give a proof of termination and shape quality, and some implemented complex examples. In our algorithms we bound only the orthoradius -to-shortest-edge ratio of the output mesh that will leave behind poor quality “sliver” but there are still have some methods proffered to deal with slivers[17-18].

## II. Preliminaries

There are some definitions which will be used later in content, and they have been introduced in earlier other’s works.

### A. Local Feature Size & Minimum Distance

This concept is first introduced by Ruppert[5], and it is used to formalize the idea of sparest possible spacing and is a basis definition in this area. Given a PSLG or PLC *P*, the local feature size at a point *p*, *lfs* (*p*), is the radius of the smallest disk or sphere centered at *p* that intersect two non-incident elements of P. Also, we define the *minimum distance* at a point p, *md*(*p*), is the radius of the smallest disk or sphere center at *p* that intersect two elements of P, but at least one of the two elements is non-incident with *p*. Here if the intersection of two elements is non-empty, we call the two elements is incident. Shown in Fig.1a




(a) (b)

Fig.1. (a) Local feature size and minimum size, (b) Weighted Delaunay triangulation

### B. Weighted Points & Weight Delaunay Triangulation

A weighted point $\hat{p}=(p,P^2)$ is the disk or sphere with center *p* and radius *P*. The weighted or power distance between two weighted points $\hat{p}$ and $\hat{m}$ is $\pi(\hat{p},\hat{m})=\|p-m\|^2-P^2-M^2$ . We call $\hat{p}$ and $\hat{m}$ *orthogonal* if $\pi(\hat{p},\hat{m})=0$. That mean the two spheres meet in a circle and the two tangent planes at every point of this circle form a right angle.

If $\pi(\hat{p},\hat{m})>0$, we call $\hat{p}$ and $\hat{m}$ *further than orthogonal*. Let $\tau$ be a simplex of dimension one or more, the smallest orthoshpere of $\tau$ is the smallest sphere of all spheres that are orthogonal to each vertex of $\tau$ . If all vertices of $\tau$ are unweighted, its smallest orthosphere is equal to its diametral sphere. And for a triangle in 2D or a tetrahedron in 3D, its orthosphere is unique. We also call the center and radius of the smallest orthosphere of any simplex as its orthocenter and orthoradius respectively.

### C. Quality Measures

For a triangle or tetrahedron, we know one of its quality factor is radius-edge ratio, which is the ration of it circumradius to its shortest edge length. In our paper, for the triangle or tetrahedron whose vertices may be a weighted point, we define the orthoradius-edge ratio $\rho(t)=OR/L_{\min}$ as its new quality factor, where *OR* and $L_{\min}$ are the orthoradius and the shortest edge length of *t*.

## III. Small Angles Problem

According to the theory of Voronoi diagram and its dual Delaunay triangulation, it can be found that the fundamental idea of Delaunay refinement for conformity is from next two lemmas:

**LEMMA 3.1** *For an edge e of a PSLG or PLC P with vertex set V, if there exists a diametral circle/sphere of e containing no points of V in its interior, and then e must be an edge of Delaunay triangulation of V.*

**LEMMA 3.2** *For a triangular face f of a PLC P with vertex set V, if there exist a equatorial sphere of f containing no points of V in its interior, and then f must be a face of Delaunay triangulation of V.*

So, in the process of Delaunay refinement, each edge is initially represented by one sub-segment, and if one sub-segment doesn’t have empty diametral circle/sphere, which is called as encroached, it need to be split at its midpoint to two sub-segments, this process will stop until all sub-segments have empty diametral circle/sphere. Similarly, each input facet is first subdivided into triangular faces called sub-facets, if one sub-facet doesn’t have empty equatorial sphere, it also need to be subdivided by inserted its circumcenter until no such sub-facet exist. As a result, when there are some small angles in input PSLG or PLC, an endless cycle of mutual encroachment may produce ever-shorter sub-segments incident to the apex of the small angle, as shown in Fig.2.

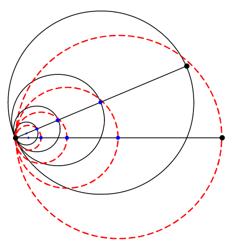


Fig. 2. Endless mutual encroachments

That problem has been found for three types of input angles, edge-edge angle, edge-facet angle and facet-facet angle. The edge-edge angle is defined as the minimum angle in 2D or 3D between two incident edges. The edge-facet angle is the angle between an edge *e* and a facet *f* which sharing a common point *o* in 3D, and it is the minimum angle of all possible angle $\angle mon$, where $m \in e$, $n \in f$, $n \neq o$, as shown in Fig.3. For the facet-facet angle, if the two facets f1 and f2 shared an common edge e, the facet-facet angle is the minimum dihedral angle between two facets, and if f1 and f2 just shared a common point p, the facet-facet angle actually is the minimum angle of all possible angles among the two facet's boundaries and the two facets, as shown in Fig.4.

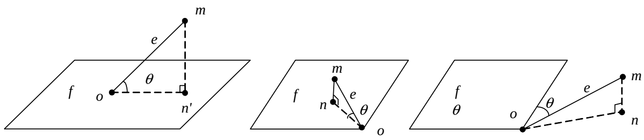


Fig. 3. The definition of edge-facet angle

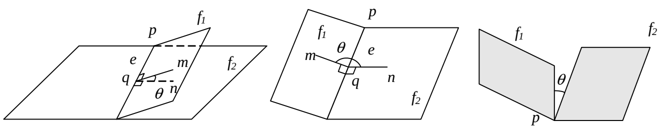


Fig. 4. The definition of facet-facet angle

In some previous works, Ruppert has presented the minimum input angle accepted by origin Delaunay refinement method is $\pi/2$, and Shewchuk extend it to $\pi/3$ for 2D case, and $\pi/2$ for 3D case. Yang Qin[11] has carefully analyzed this threshold, and presented a sufficient condition for termination of Ruppert's Delaunay refinement method:

**LEMMA 3.3** *In input PSLG or PLC, if all edge-edge angles are greater than and equal to* $\pi/3$*, all edge-facet angles are greater than and equal to* $\arccos\left(\sqrt{2}/4\right)$*, and all facet-facet angles are greater than and equal to* $\pi/2$*, then the Delaunay refinement must terminate.*

Because lemma 3.3 has been proved as a sufficient condition for Delaunay refinement, so in paper, for any input point, if there are some input edge and facet incident to this point having edge-edge angles less than $\pi/3$, or having edge-facet angles less than $\arccos\left(\sqrt{2}/4\right)$, we call this input point as ***sharp vertex***. For any input edge, if there are some incident facets have facet-facet angles less than $\pi/2$, we call it as ***sharp edge***, and the endpoint of sharp edge is also set as sharp vertex.

## IV. Weight for Small Input Angle

In previous works, different but similar methods have been developed to deal with small angle problem, in such method, they all set some protected area around the sharp element, such as Rupper and Shewchuk's concentric circular shells, David Cohen-Steiner's protecting balls, S.W.Cheng's buffer zone and apply new split rule when new added located in protected area, such as Shewchuk's splitting sub-segment cluster into same length, Cohen-Steiner's "split-on-a-sphere" strategy and S.W.Cheng's splitting shield sub-segments or sub-facets. All of above techniques have only one purpose, and that is to avoid the endless cycle by prevented the new point from being added too near the sharp elements.

In our research work, we have found there are some good properties of weighted point and its weight circle, and those properties is similar with the above improvements and can be used to solve simply the small angle problem. We first introduce those properties below:

**LEMMA 4.1** *For an edge e, if the intersection of two weight circles/spheres of its two endpoints a and b is empty, then the orthocenter o of e must be located on e and never be contained by any weight circle/sphere of a and b.*

**LEMMA 4.2** *For a triangle t, if there is one vertex of t whose weight sphere don't intersect with weight spheres of other two vertices of t, then the orthocenter of t must be on the same plane with t and not be contained by weight sphere of any vertex of t.*

According to related definition in section 2, above two lemmas is easy to be proved.

The case of mutual encroachment often occurs for the input elements which incident to a sharp angle, so how to stop this process and avoid an endless cycle is must. With the above two properties of weighted simplex, we can found that if we assign a weight to the point in the sharp element, and when the refinement on that element began, the new add point never will be more and more close to the sharp element, so the endless cycle will never happen.

According above thought, we get a weight assignment rule for sharp element in input PSLG or PLC:

1. For the sharp input point a, we assign its weight $w_a$ as: $w_a=(w \cdot lfs(a))^2$, where *lfs*(a) is the local feature of a in input PSLG or PLC and w is an constant factor and should less than 0.5.
2. For the sharp input edge e, when it or its sub-segment need to divided by its orthocenter, we set the current divided sub-segment as s, its orthoradius is $r_s$, its orthocenter point is o, so we assign o's weight $w_o$ as: $w_o = min(r_s^2,(w \cdot md(o))^2)$, where md(o) is minimum distance of o, and *w* should be less than 0.5.

## V. Algorithm

Typically, we give the description of our weighted Delaunay refinement method for small input angle in 3D case below:

Firstly, we get an input PLC *X* and compute all sharp vertices and sharp edges, for sharp vertices, we set the corresponding weighted to them as above mentioned, and for sharp edges, we just assign a sharp flag to them.

Because there are some weighted vertices in input after above setting, we need to use weighted Delaunay triangulation in later algorithm instead origin Delaunay triangulation under Euclidean distance. In weighted algorithm, a sub-segment or a triangular sub-facet is encroached if a vertex other than its endpoints or its vertices has weighted or power distance less than or equal to the power distance between its orthocenter and its vertex.

After initialized weighted setting, another important thing is to subdivide the facets in input PLC into triangular faces in advance. That subdivision actually is a 2D process of our algorithm in the plane the facet located. In that process, the facet and other coplanar vertices and edges in PLC are treat as an input PSLG for 2D algorithm, and It is worth to notice that when splitting the boundary of facet and this boundary edge is a sharp edge, its orthocenter edge must be set proper weight before inserting it to mesh.

Then, our algorithm can begin with the weighted Delaunay tetrahedralization $\mathcal{WDT}(X)$ of the vertices of $X$ which include input vertices, endpoints of input edges and mesh vertices of facet's triangular mesh. Some segments and facets of X may be missing from the tetrahedralization, in which case their recovery is aided by the insertion of additional vertices (while maintaining the weighted Delaunay property). Vertex insertion is governed by two rules:

1. If a sub-segment is encroached and is missing in the $\mathcal{WDT}(X)$, as it should be split immediately by inserting its orthorcenter, and if the sub-segment is a belong to a sharp edge, its orthocenter should have weight as mentioned in section 4.
2. If a triangular sub-facet is encroached and is missing in the $\mathcal{WDT}(X)$, its orthocenter also needs to be inserted into $\mathcal{WDT}(X)$. However, if the new vertex would encroach upon any sub-segment, it is not inserted, and instead all the sub-segment it would encroach upon is split as step 1.

For quality, our algorithm iteratively inserts orthocenter of poor quality tetrahedron that have orthoradius-edge ratio above a threshold. However, if the new vertex would encroach upon any sub-segment or sub-facet, then it is rejected and instead all the encroached elements need to be subdivided. As sequence above, all encroached sub-segments are given priority over encroached sub-facets, which have priority over poor quality tetrahedron.

Once every missing segment is represented by a contiguous linear sequence of edges of tetrahedralization, the missing facets can be recovered by ensuring conformality for all segments, and just after all missing segments and facets have been recovered, the poor quality tetrahedron can be eliminated by inserting its orthocenter with operating in conjunction with above two ensue. The weighted Delaunay property of the tetrahedralization is maintained throughout all the remaining vertex insertions.

## VI. Algorithm Analysis

According to above illustration about weighted method, because the sharp vertices or sharp edges will be covered by weight circle/sphere of vertices on its at the extreme case, there aren't points too near to the sharp element and the endless cycle of mutual encroachment will never happen. So, the processing of recovery for input edges or facets will terminate.

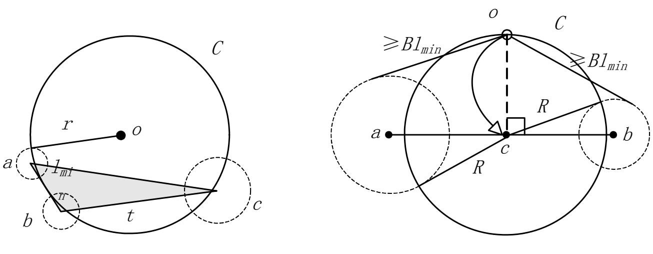
In this part, we focus on the termination of mesh quality improvement, and gives quality bounded orthoradius-edge ratio for termination.

**THEROEM 6.1** *For 2D or 3D case of our algorithms, if the threshold B for orthoradius-edge ratio of triangles /tetrahedrons is larger than* $\sqrt{2}$ *, then during the quality improvement, our algorithm will terminate with no triangulation edge shorter than the shortest edge.*

**Proof**. For 2D algorithm, we have two cases when inserting the orthocenter of poor quality triangle *t*:

1) If the orthocenter doesn't encroach any segment, it will insert into the triangulation. Now, let *C* be the smallest *t*, *o* is its orthocenter and *r* is its orthoradius, $l_{\min}$ is the smallest edge length of t. So we have $r/l_{\min} > B$. When point *o* is inserted into the triangulation, according to the property of weighted Delaunay triangulation and definition of weighted distance, the smallest length of all new edges is larger than orthoradius r, so we get all new edges with their length are larger than $l_{\min}$ when B is larger than $\sqrt{2}$. As illustration in Fig.5a.

2) If the orthocenter encroach some segments, it will be rejected and the encroached segment's orthocenter will be inserted. Let *e* be the encroached segment, a and b is its tow end-points, and C be its smallest orthosphere, c is its orthocenter and R is its orthoradius. $l_{\min}$ is also the smallest edge length of t. Because the power distance between o and a is equal to the power distance between o and b, and C contains point o, for general, we can let $\angle oca \le \pi/2$, and we get $\|o-a\|^2 \le \|o-c\|^2+\|c-a\|^2$ and $\|o-c\| \le R$, meanwhile, weighted point *a* and *c* is orthogonal, so we get $\|c-a\|^2-w_a^2=R^2$. Integrating above equations with the power distance equation between *o* and *a* $pl_{oa}=\|o-a\|^2-w_a^2$, we get $R \ge pl_{oa}/\sqrt{2}$. According to above analysis, we have $pl_{oa} \ge Bl_{\min}$, so we get $R \ge B \cdot l_{\min}/\sqrt{2}$. So we get all new edges with their length are larger than $l_{\min}$ when B is larger than $\sqrt{2}$ after inserting the orthocenter c into the triangulation, as illustration in Fig.5b.




(a) (b)

Fig. 5. Inserted new point in 2D algorithm

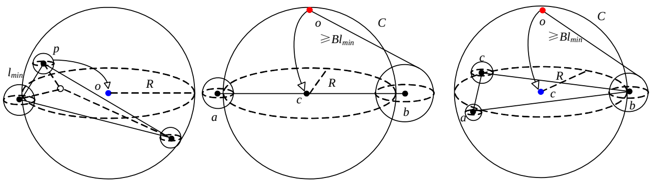
For 3D algorithm, we have three cases when inserting the orthocenter of poor quality tetrahedron *t*:

1) If the orthocenter doesn't encroach any segment or facet, it will insert into the tetrahedralization. This case is same as the above first case, as illustration in Fig.6a.

2) If the orthocenter encroached some segments, that is similar with above second case, as illustration in Fig.6b.

3) If the orthocenter encroached some triangles of facets, it will be rejected and the orthocenter *v* of encroached triangle *t* will be inserted into tetrahedralization. Let a, b and c is the vertices of t, and C is its smallest orthoshpere, c and R is its orthocenter and orthoradius. Now, let H is a plane containing point a and o and is perpendicular to t. The intersection of H and C is a circle called C′ whose center is $c'$. So, for generation, we can let $\angle oc'a \le \pi/2$, and we get $pl_{oa}^2 = \|o-a\|^2 - w_a^2 \le \|c'-o\|^2 + \|c'-a\|^2 - w_a^2$. Because $c'$ is the projection of c on H, we have $\|c'-o\| \le \|c-o\|$ and $\|c'-a\| \le \|c-a\|$. Also, we Integrate above equations with

the power distance equation between o and a $pl_{oa}$=||o-a||$^2$-w$_a^2$ ≥ $Bl_{min}$. we get $R \geq Bl_{min} / \sqrt{2}$ , so we get all new edges with their length are larger than $l_{min}$ when $B$ is larger than $\sqrt{2}$ after inserting the orthocenter $c$ into the tetrahedralizaion, as illustration in Fig.6c.



(a) (b) (c)

Fig. 6. Inserted new point in 3D algorithm

From above analysis, we get a bounded orthoradius-edge ration for the output mesh of our algorithms, as we known, for mostly triangles or tetrahedrons which don't contain weighted points as vertices, their orthoradius-edge ration are the same as their circumradius-edge ratio. So we conclude our algorithm can guarantees bounded circumradius-edge for all elements except the ones near small input angles.

## VII. Examples and Conclusion

We have implemented above weighted Delaunay refinement algorithm both in 2D & 3D with C++ language, and we use the weighted version Boyer/Watson algorithm to implement weighted Delaunay triangulation of vertices. All of those cases have the setting with quality threshold of orthoradius-edge ratio is $\sqrt{2}$ and the constant factor for sharp elements is 1/3. Fig.7 shows a result mesh produced by our 2D algorithm for a PSLG with nine small angles and this case is similar to the example presented by Shewchuk[9], and the bold line in left figure is the segment of input PSLG, and the circles show is the weight circle of sharp vertices. The right figure gives the column plot of angle of mesh. In the final mesh, no angle is less than the smallest input angle 2.49 º. Fig.8 is an input PLC of pyramid with many faces and the result of tetrahedralization. It's all segment are sharp edges and the circles on the edges are the weighted circles of vertices.Fig.9-10shows the result of the experiment on a typical non-manifold geometry, which is a box with ten oddly shaped facets that have a common segment. The origin input PLC is present in the left of Fig.9, and the right part is the final result triangular mesh on all facets. Fig.10 is the final Delaunay tetrahedral mesh. And their weight sphere on the sharp edge.

Although above results of our algorithm still have some uncompleted work on the quality of final mesh, but our algorithm is easy to implement and will have good stability and reliability. It is not only an algorithm in theory but also an algorithm for engineering application. In the future work, we will continue to research for the improvement on the quality of mesh, including the method to avoid subdividing the input small angle to smaller angle and also including the method to eliminate "sliver" tetrahedron.

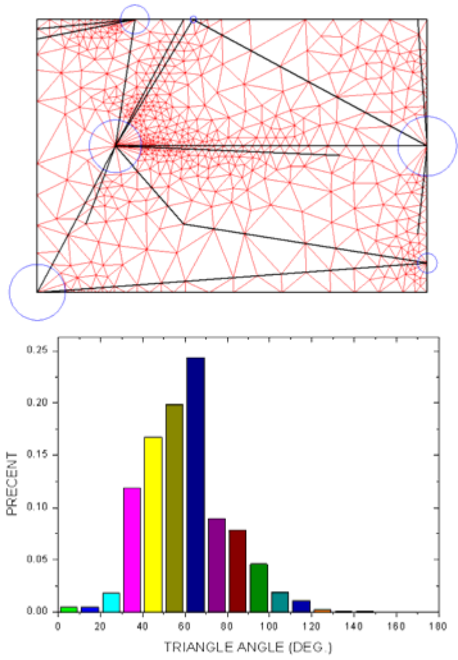



Fig. 7. Result meshes and plot of angle

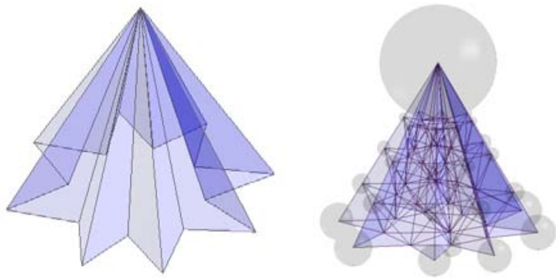


Fig. 8. Pyramid with many face

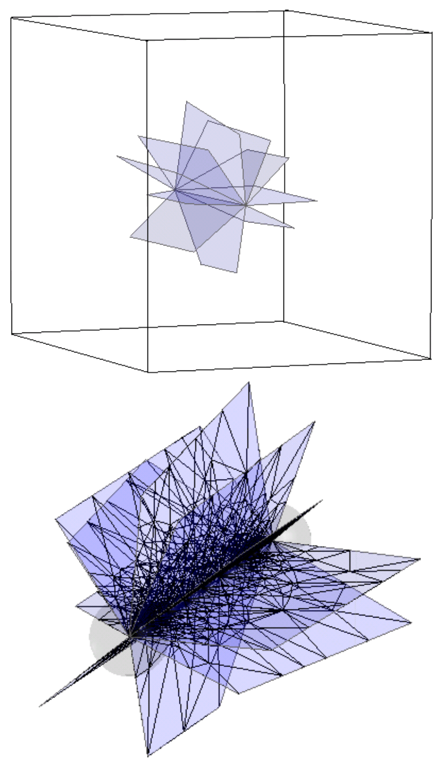


Fig. 9. No-manifold input PLC and resulting triangular mesh on the input facets

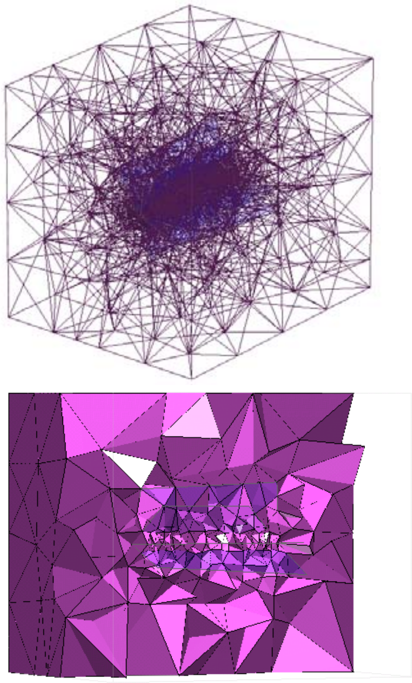


Fig. 10. Final tetrahedral mesh and a section view

REFERENCES

[1] H. Borouchaki, P. L. George, "Aspects of 2-D Delaunay mesh generation," *International Journal for Numerical Methods In Engineering*. vol.40, no. 11, pp.1957-1975, 1997.
[2] P. L. George, F. Hecht, E. "Saltel, Automatic mesh generation with specified boundary," *Computer Methods in. Applied Mechanics and Engineering*. vol.92, pp. 169-188, 1991.
[3] N. Weatherill, O. Hassan, "Efficient three dimensional Delaunay triangulation with automatic point creation and imposed boundary constraints," *International Journal for Numerical Methods in Engineering*. vol. 37, pp. 2005-2039, 1994.
[4] Chew L. P. "Guaranteed-quality triangular meshes." Tech. Rep. 89-983, Computer Science Department, Cornell University, 1989
[5] Ruppert J. "A Delaunay refinement algorithm for quality 2-dimensional mesh generation." *J.Algorithms*, vol. 18, no. 3, pp.548-585, 1995.
[6] Preparata F P, Shamos M I. *Computational Geometry: An Introduction*. Springer, New York, 1985
[7] J. R. Shewchuk. "Tetrahedral mesh generation by Delaunay refinement." In: *Proc. 14th ACMSympos. Comput. Geom.*, 1998,pp. 86-95.
[8] Shewchuk J. R. "Mesh Generation for Domains with Small Angles". In: *Proceedings of the 6th Annual Symposium on Computational Geometry*, 2000, pp.1-10
[9] Shewchuk J. R. "Delaunay Refinement algorithms for Triangular Mesh Generation". *Computational Geometry*, vol.22, no. 1-3, pp.21-74, 2002.
[10] Miller G. L., Pav S.E., Walkington N. J. "When and Why Ruppert's Algorithm Works". In: *Proceedings of 12th International Meshing Roundtable*, 2003, pp. 91−102
[11] Yang Qin, "Conforming Delaunay Triangulation", Ph.D. dissertation, Dept. Compu. Sci., Beihang University, 2001.
[12] Murphy M, Mount D M, Gable C W. "A point-placement strategy for conforming Delaunay tetrahedralization". In: *Proceeding of the 11th ACM-SIAM Symposium on Discrete Algorithms*, pp. 67-74, 2000.
[13] Cohen-Steiner D, De Verdiere E C, Yvinec M. "Conforming Delaunay Triangulations in 3D". In: *Proceeding of the 18th annual symposium on Computational Geometry*, pp. 199-208, 2002.
[14] Pav S. Walkington N. "Robust Three Dimensional Delaunay Refinement". In: *Proceedings of 13th International Meshing Roundtable*, pp.145-156, September 2004.
[15] Cheng S. W, Poon S. H. "Graded Conforming Delaunay Tetrahedralization with Bounded Radius-edge Ratio". In: *Proceedings of the 14th* annual *ACM-SIAM symposium on Discrete algorithms*[C], pp. 295-304, 2003
[16] Cheng S. W., Dey T. K., Ramos E.A et al. "Quality Meshing for Polyhedral withSmall Angles". In: *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, pp. 290-299, June 2004,
[17] Cheng S. W., Dey T. K., Edelsbrunner H., Facello M.A., Teng S.H. "Sliver exudation." In: *Proceedings of the _fteenth annual symposium on Computational geometry*, pp. 1-13, 1999
[18] Li X. Y., Teng S. H. "Generating well-shaped Delaunay meshes in 3D." In:*Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pp.28-37, 2001.

**Xianhai Meng** was born in December 1977. He has received the B.S. degree and M.S. degree in Department of aircraft design and applied mechanics from Beihang University, Beijing, China, in 1998 and 2001 and the Ph.D. degree in School of Computer Science & Engineering from Beihang University. His interested research area includes computer graphics, computational geometry and geometric modeling. He has been positioned as a Lecturer since 2006 in School of Computer Science & Engineering, Beihang University, Beijing, China.