Automatic construction of quality nonobtuse boundary Delaunay triangulations *

Nancy Hitschfeld and María-Cecilia Rivara
Department of Computer Science, University of Chile,
casilla 2777, Santiago, CHILE
e-mail: nancy@dcc.uchile.cl, mcrivara@dcc.uchile.cl

Abstract. *In this paper we discuss the automatic construction of quality nonobtuse boundary Delaunay triangulations of polygons such as needed for control volume or finite element method applications. These are Delaunay triangulations whose smallest angles are bounded and, in addition, whose boundary triangles do not have obtuse angles opposite to any boundary or interface edge. The method we propose in this paper consists on: (1) The construction of a constrained (good quality) Delaunay triangulation of the polygon by using a Lepp-Delaunay algorithm (based on the longest-edge propagation path of target triangles); (2) A postprocess step which eliminates obtuse angles by Delaunay insertion of a finite number of adequate points on boundary or interface edges.*

keywords. Nonobtuse triangulation, Delaunay meshes, Voronoi diagram, control volume discretization method, box method.

1 Introduction

The numerical solution of partial differential equations (PDEs) is invaluable in design and optimization in many fields of engineering. The spatial discretization (mesh) of the structure to be simulated, i.e. its subdivision in cells, is key to the accuracy of the computed solution. An appropriate mesh should fulfill several requirements. First, it must provide a reasonable approximation of the geometry to be modeled, in particular of its boundary and internal material interfaces. Second, it is extremely important to accurately approximate all internal quantities relevant to the solution of the PDEs. Third, each cell must fulfill certain geometric constraints imposed by the numerical integration

*Submitted to a journal, Juni 1998. Technical Report TR/DCC-98-2.

method: if the PDEs are solved with the finite element method, no angle must be smaller than some bound supplied *a priori*. However, if the equations are solved using a control volume discretization method(CVM)[1], the center of the smallest circumcircle that surrounds each boundary element must be inside the region of the element [2]. For two dimensional geometries (2-D) this means that the angle opposite to a boundary edge must be a nonobtuse angle.

The CVM is very popular in the numerical simulation of semiconductor devices [1, 3, 4, 5]. In 2-D, both triangulations and mixed element meshes have been used. A review of previous work on this area can be found in [6, 7]. A more recently approach is the one presented in [8] based on the sphere packing technique [9]. All these approaches generate meshes without obtuse angles.

This paper presents a new algorithm to generate good quality 2-D meshes for both control volume discretization and finite element method which extends the Lepp-Delaunay algorithm introduced by Rivara in [10]. This kind of meshes are also very useful when problems are solved combining both methods. For example, Bürgler [3] uses the CVM method (voronoi diagram) to obtain the numerical solution of the Poisson equation and the finite element mesh for grid adaption and error estimation. This requires the combination of good quality meshes and well shaped Voronoi boxes. In particular, the minimum angle should be bounded and boundary triangles should not have obtuse angles opposite to any boundary edge or interface edge.

The method we propose in this paper, based on the use of longest-edge bisection techniques [11], consists on two steps: (1) The construction of a good quality (constrained) Delaunay triangulation (CDT) of the polygon having interior angles comprised between 30° and 120° [10]; (2) A postprocess step which eliminates boundary obtuse triangles by combining longest-edge insertion points, the Delaunay algorithm and a new treatment for a certain type of boundary obtuse triangles.

The construction of the good quality (constrained) Delaunay triangulation consists of: (a) The generation of an initial constrained Delaunay triangulation (which essentially uses the polygon vertices), and b) the use of an Lepp-Delaunay algorithm which improves the quality of the mesh so that the minimum angle is greater than or equal to 30°. The basic Lepp-Delaunay improvement strategy uses the Longest-Edge Propagation Path of the target triangles (to be either refined and/or improved in the mesh) in order to decide which is the best point to be inserted, to produce a good-quality distribution of points. This strategy is repeatedly used until the target triangle is destroyed.

The step designed to eliminate boundary obtuse triangles of polygonal regions considers three cases: (a) triangles with only one boundary edge which is opposite to the obtuse angle, (b) triangles with two boundary edges and one of them opposite to the obtuse angle, and (c) triangles with three boundary

edges. The case (a) is solved by inserting the midpoint at the boundary edge. Since the obtuse angle is smaller than or equal to 120°, the insertion of only one point is required. Some diagonal swapping might be necessary. For the case (b), a boundary isosceles triangle of the largest edges equal to half the smallest boundary edge of the target triangle is constructed. This construction can produce an obtuse triangle with one boundary edge, which is in turn eliminated by the Delaunay insertion of the midpoint of its longest edge. This point insertion can again produce a new boundary obtuse triangle, with largest angle smaller than the previous one and so on. The boundary obtuse triangles are eliminated after the insertion of a finite number of points. A triangle with three boundary edges (case (c)) is a particular case. Depending on the angles of the triangle, one or two isosceles triangles with two boundary edges are generated and a finite number of points inserted.

For obtuse triangles with one interface edge the same strategy as for case (a) is applied. For triangles with two or more interface edges adjacent to other obtuse triangles with two boundary edges, isosceles triangles for the new triangles which keep two interface edges are constructed. Since the boundary obtuse triangles are adjacent, the number of points inserted on shared edges is bounded by the edge that requires the highest number of point insertions.

The final mesh (having no boundary obtuse angles and without interface obtuse angles) is a Delaunay triangulation even for the triangles lying at the boundary and interface.

2 Basic concepts and definitions

Definition 1 *A boundary triangle is any triangle that has at least one edge on the geometric boundary or on a material interface (boundary edge).*

Definition 2 *A boundary obtuse triangle is any triangle that has a boundary edge opposite to its obtuse angle.*

Definition 3 *An interface obtuse triangle is any triangle that has an interface edge opposite to its obtuse angle.*

Definition 4 *An 1-edge boundary triangle is any triangle that has exactly one boundary or interface edge*

Definition 5 *An 2-edge boundary triangle is any triangle that has exactly two boundary or interface edges.*

Definition 6 *A boundary constrained angle is an angle that is defined by two boundary edges. This angle can not be modified.*

Definition 7 *Let P be a polygon with material interfaces. A tessellation T of P is appropriate for the* CVM *[1, 2] (well-shaped) if*
(i) T is a Delaunay tessellation,
(ii) The center of the circumcircle (Voronoi Point) that surrounds each boundary triangle lies in the same polygon as the boundary triangle.

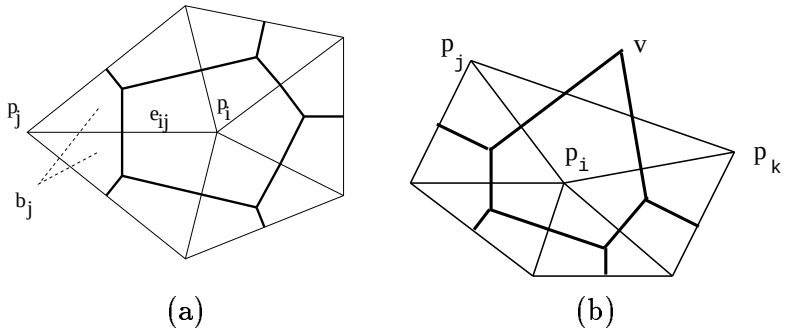

Figure 1: 2-D Delaunay triangulations and their Voronoi diagrams. Figure (a) shows an acceptable triangulation for the CVM, and Figure (b) shows an unacceptable triangulation (the Voronoi point v is outside the boundary obtuse triangle defined by p_j, p_k, p_i, where $p_j p_k$ is a boundary edge)

Theorem 1 *(Thales) Let t be a triangle defined by the vertices A,B, and C. If the triangle t lies on a circumcircle so that one of its edges is equal to the diameter of the circumcircle, then the triangle is a right triangle (Figure 2(a)).*

The Thales theorem can be also interpreted in the following way: If the circle with center in the midpoint M of AB and diameter AB includes the point C on its boundary, the angle α is a right angle and the center of the circumcircle of the triangle (C_1) coincides with M (Figure 2(a)); if the point C is outside this circle, α is a nonobtuse angle and the center of the circumcircle of the triangle (C_1) is inside the triangle (Figure 2(b)) and if the point C is inside the circle, α is an obtuse angle and the center of the circumcircle of the triangle (C_1) is outside the triangle (Figure 2(c)). This analysis will be used later to show if a particular Voronoi point C_1 is inside or outside a triangle ABC.

3 Preliminary concepts: the Lepp(t) and geometrical properties

This section reviews the Lepp concept [10] and summarizes some geometrical properties introduced in [12]:

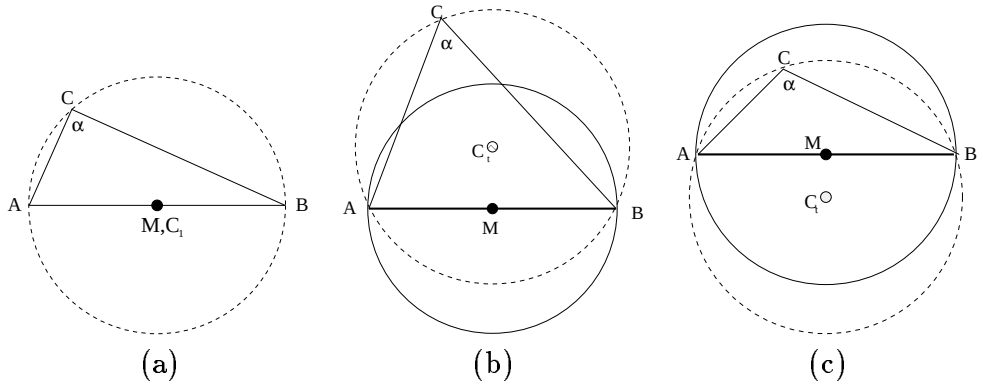

Figure 2: (a) Thales theorem: $\alpha = 90°$, (b) Center (C_1) of the circumcircle that surrounds t is inside the triangle (c) Center (C_1) of the circumcircle that surrounds t is outside the triangle ($\alpha > 90°$)

Definition 8 *For any triangle t_0 of any conforming triangulation T, the Longest-Edge Propagation Path of t_0 will be the ordered list of all the triangles t_0 , t_1, t_2, ...t_{n-1}, t_n, such that t_i is the neighbor triangle of t_{i-1} by the longest edge of t_{i-1}, for i = 1,2,.., n. In addition we shall denote it as the Lepp(t_0).*

Proposition 1 *For any triangle t_0 of any conforming triangulation of any bounded 2-dimensional geometry, the following properties hold: (a) for any t, the Lepp(t) is always finite; (b) The triangles t_0 , t_1,..., t_{n-1}, have strictly increasing longest edge (if $n > 1$); (c) For the triangle t_n of the Longest-Edge Propagation Path of any triangle t_0, it holds that either: (i) t_n has its longest edge along the boundary, and this is greater than the longest edge of t_{n-1}, or (ii) t_n and t_{n-1} share the same common longest edge.*

Definition 9 *Two adjacent triangles (t, t*) will be called a pair of terminal triangles if they share their respective (common) longest edge. In addition t will be a terminal boundary triangle if its longest edge lies along a boundary side.*

Note that the Longest-Edge Propagation Path of any triangle t corresponds to an associated polygon, which in certain sense measures the local quality of the current point distribution induced by t. To illustrate these ideas, see Figure 7(a), where the Longest-Edge Propagation Path of t_0 corresponds to the ordered list of triangles (t_0, t_1, t_2, t_3). Moreover the pair (t_2, t_3) is a pair of terminal triangles.

The definition 8 should be slightly modified to consider the case where the longest edge is not unique. In such a case, the longest edge that produces the shortest path should be selected.

Definition 10 *For any input PLSG (planar straightline graph) that defines a general polygon to be triangulated, the geodesic distance between two points of the polygon is defined as the shortest path that stays within the interior of the polygon. In addition points P and Q will be called geodesic interior points if the geodesic distance between both points is equal to the shortest Euclidean distance between points P and Q (see Figure 3). Otherwise they will be geodesic exterior points.*

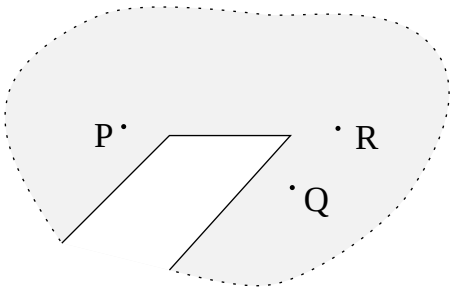

Figure 3: Points P and Q (P and R) are geodesic interior (exterior) points.

Definition 11 *For any input PLSG which defines a general polygon to be triangulated, three points A, B, C contribute to a valid (Delaunay constrained) triangle t in a CDT if (a) The vertices A, B, C, are geodesic interior points between them; and (b) The circumcircle through the points A,B,C contains no other geodesic interior point (with respect to the points A,B,C,) in its interior (see Figure 4).*

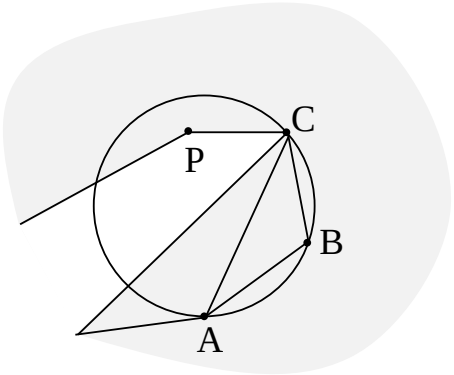

Figure 4: Triangle ABC is a valid Delaunay triangle

Proposition 2 *Let t be any triangle $t = t(A, B, C)$ of longest-edge AB. Then for any neighbor triangle t^* that shares side AB with t, the pair (t, t^*) forms a pair of Delaunay terminal triangles if and only if the third vertex of t^* belongs to the region $R = C_A \cap C_B - C_C - C_t$, where C_A, C_B, y C_C are circles of radius equal to the length of AB and respective centers A, B and C; and C_t is the circumcircle of triangle t.*

Proof. It follows from the fact that the pair (t,t*) is a pair of terminal triangles over a Delaunay triangulation and the conditions that such a pair of triangles hold.□

Figure 5 illustrates three different cases of regions R. A detailed proof can be found in [12].

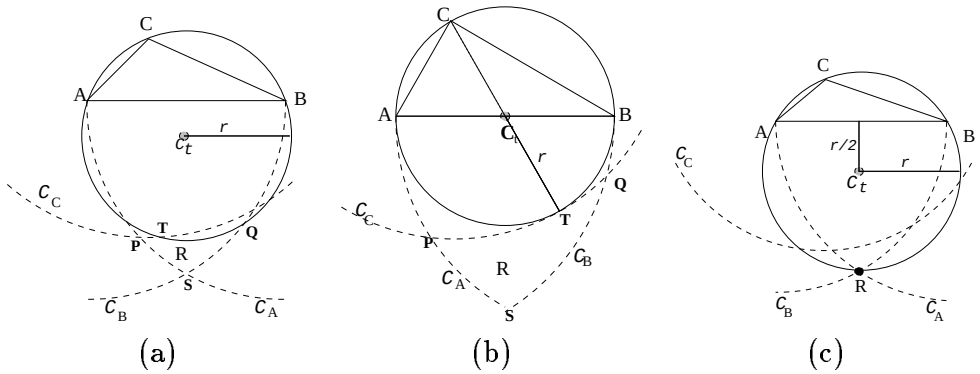

Figure 5: Geometrical place for the 4th vertex (denoted by region R) in a pair of terminal triangles

The following Theorem [12] states the geometrical conditions which assure that (t, t^*) is a pair of Delaunay terminal triangles. In this case the third vertex of t^* must belong to $R \neq \phi$.

Theorem 2 *Let τ be any CDT. Then for any pair of Delaunay terminal triangles (t, t^*) in τ, the following property holds: t is an obtuse triangle if and only if the distance d between the circumcenter P_t of t and the longest-edge of t satisfies that $0 < d \leq \frac{r}{2}$, where r is the circumradius of t.*

Proof. The result follows by finding the limit case where R reduces to one point (C_A intersects C_t only in one point), which holds for d=r/2 (See Figure 5(c)).□

Corollary 3 *For any pair of Delaunay terminal triangles (t, t^*), t is an obtuse triangle if and only if its largest angle γ holds that $\gamma \leq 120°$; and t^* is an acute-angled triangle.*

4 Lepp-Delaunay improvement triangulation algorithm and properties

In this section we use an improved version of the Lepp-Delaunay algorithm (introduced in [10]) which allows the quality improvement of any triangulation in the sense that a minimum angle of 30° is obtained for any angle that is a non-boundary constrained triangle.

The basic Backward-LE-Delaunay improvement procedure uses the Longest-Edge Propagation Path of the target triangles (to be either refined and/or improved in the mesh) in order to decide which is the best point to be inserted, in order to produce a good-quality distribution of points. This procedure is repeatedly used until the triangle t is destroyed. Note that this basic algorithm does not consider the fact that t could be a boundary triangle.

```
Basic Backward-LE-Delaunay-Improvement (t, T){
while  t remains without being modified do
      Find the Longest-Edge Propagation Path of t
      Perform a Delaunay insertion of the point p
           (midpoint of the longest edge of the last triangle
               in the Lepp(t))
end while
}
```

Figure 6: Backward-LE-Delaunay improvement procedure

We have used the word improvement instead of bisection or refinement. This is to explicit the fact that one step of the procedure does not necessarily produce a smaller triangle. More important however, is the fact that the procedure improves the triangle in the sense of Theorem 4. The proof of this theorem can be found in [10].

Theorem 4 *For any Delaunay triangulation T, the repetitive use of the Backward-LE-Delaunay-Improvement technique over the worst triangles of the mesh with smallest angle* $< 30°$ *produces a quality triangulation of smallest angles greater than or equal to* $30°$.

Corollary 5 *The use of Lepp-polygon triangulation algorithm with* $\varepsilon = 30°$ *produces a Delaunay triangulation such that obtuse triangles have angles smaller than or equal to* $120°$.

For an illustration of this idea see Figure 7 where the triangulation (a) is this initial Delaunay triangulation with $\text{Lepp}(t_0) = t_0, t_1, t_2, t_3$, and the triangulation (b), (c) and (d) illustrate the complete sequence of point insertions

needed to improve t_0. In this example, the improvement (modification) of t_0 implies the automatic Delaunay insertion of three additional Steiner points. Each one of these points is the midpoint of the last triangle of the current Lepp(t_0). It should be pointed out here that each Delaunay point insertion essentially improves the local point distribution in the current Lepp(t_0).

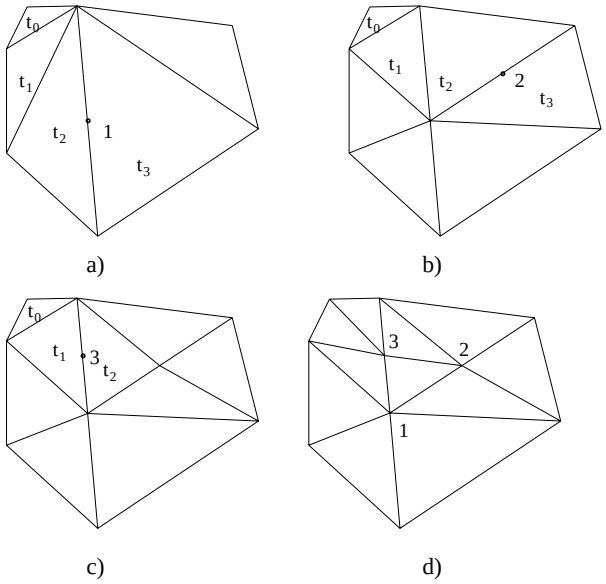

Figure 7: Backward Longest-Edge Delaunay improvement of triangle t_0

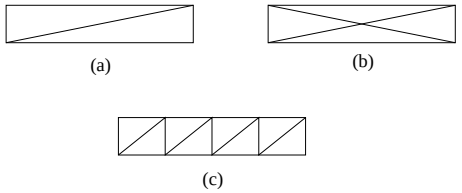

Figure 8: Boundary treatment technique

By combining the basic Lepp-procedure and adequate boundary considerations, a simple 2-dimensional quality-triangulation algorithm is obtained. The special boundary treatment technique is to avoid the insertion of undesirable interior points. To illustrate this idea consider the simple example of Figure 8(a). In this case the naive use of the Lepp point insertion algorithm would produce undesirable interior points (as shown in Figure 8(b)).

The Lepp-improvement algorithm including the special boundary treatment can be formulated as shown in Figure 9.

```
Quality-Polygon-Triangulation ( P, δ ) {
Input: A general polygon P (defined by a set of vertices
   and edges);  and a tolerance parameter (δ  <  30^o)
Construct T, a constrained (boundary) Delaunay
   triangulation of P.
Find S, the set of the worst triangles t of T (of smallest
     angle < δ)
for  each t in S do
     Backward-LE-Delaunay-Improvement (T, t)
     Update the set S (by adding the new small-angled
     triangles and eliminating those destroyed throughout
     the process)
end for
}

Backward-LE-Delaunay-Improvement (T, t) {
while  t remains without being modified do
    if  (t* has a boundary edge l, and l is not the
         smallest edge of t,)
         select p, the midpoint of l
    else
          Find the Lepp(t),  and t* the last triangle in
          the Lepp(t)
          select p midpoint of the longest edge of t*
    end if
    Perform the Delaunay insertion of p
end while
}
```

Figure 9: Lepp-procedure with boundary considerations

Note that: (1) δ is a threshold parameter less than or equal to 30° that can be easily adjusted; (2) in practice we have worked with a constrained Delaunay triangulation of the 2-dimensional geometry (Chew, 1989); (3) the quality-triangulation algorithm maintains and processes the triangles of the set S in any order.

5 Nonobtuse boundary Delaunay triangulations

In this section we shall show that by using a postprocess step over the quality mesh generated with the Lepp-polygon triangulation algorithm described in the previous section (with an $\varepsilon = 30°$), the unacceptable triangles having obtuse angles opposite to a boundary or interface edge are eliminated.

Furthermore, we shall show that the resulting triangulation is a Delaunay triangulation and not a constrained Delaunay triangulation.

5.1 Non-obtuse boundary triangles for polygons without interfaces

Boundary obtuse triangles with one, two or three boundary edges require different strategies to eliminate the obtuse angle.

5.1.1 Triangles with one boundary edge

Theorem 6 *Let τ be any improved Delaunay triangulation of any PSLG geometry (with smallest angle greater than or equal to 30°). Let t be a boundary obtuse triangle of τ and e the unique boundary edge of t. Then (a) the obtuse triangle is eliminated by inserting the midpoint of e and (b) the new generated boundary triangles are nonobtuse triangles.*

Proof. Let t be any boundary obtuse triangle of τ of vertices A,B,P where AB is the unique boundary edge of t. In order to prove part (a) of the theorem consider the extreme case of the isosceles obtuse triangle ABC of longest edge equal to AB, largest angle equal to 120° and smallest angles equal to 30° that restrict the geometry of t (See Figure 10(a)). In effect the vertex P of t must belong to the region limited by the prolongation of the edges BC and AC and the circle of diameter AB. We shall show that for the extreme triangle ABC, (1) the insertion of the midpoint M of AB generates two nonobtuse boundary triangles CBM and CMA and (2) the Delaunay point insertion step (diagonal swapping) does not introduce obtuse angles of vertex C (triangles CBM and CMA are boundary nonobtuse triangles). Let suppose that triangle CBM has an obtuse angle of vertex C. In such a case the point C of this triangle should be inside of the circle with center M_1 and diameter BM. However, this is not possible because according to theorem 2, for the specific triangle ABC, the distance between C and M is $r/2$ (where r is the radius of the circumcircle that surrounds t) and the radius of the circles with centers M_1 and M_2 is less than r. Thus, region CDE does not intersect these circles, which implies that the circles do not contain the vertex P of t, for any valid vertex P.

To proceed with the second part of the proof consider the region Ω in Figure 10(b) which identifies the location of a vertex D so that a diagonal swapping is required after the insertion of the point M. The diagonal swapping generates two new triangles where one of those, the triangle MDA, is a boundary triangle. Since the smallest angle of triangle ACD is 30°, the new boundary triangle is nonobtuse because the circle with diameter AM does not include D. The shortest distance between D and the edge MA is r and the circle with center M_2 has a radius less than r.□

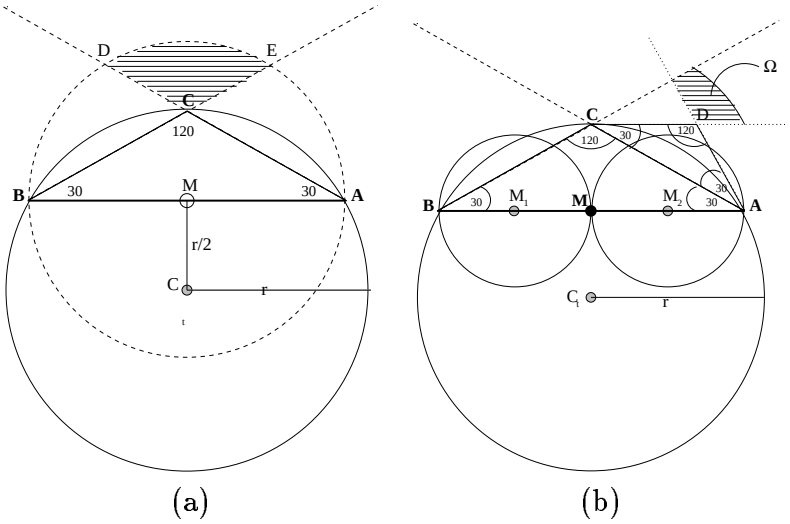

Figure 10: (a)The shadow region shows the possible location of the P vertex so that ABP is an obtuse triangle on P, (b) Diagonal interchange (AC to MD) does not produce a new boundary obtuse triangle

Corollary 7 *For any improved Delaunay triangulation of any PLSG geometry (without interfaces) the number of point insertions (N_{1b}) required to eliminate N 1-edge boundary obtuse triangles is equal to N.*

5.1.2 Triangles with two boundary edges

The elimination of 2-edge boundary obtuse triangles can be divided into two cases:

1. The smallest edge and the longest edge of the triangle are boundary edges (edges BC and BA in Figure 11). Note that in this case, the boundary constrained angle is β with $\beta \geq \alpha \geq 30°$ (The angle of vertex A is smaller than or equal to the angle of vertex B). The strategy of the previous section also applies to this case because the insertion of the AB midpoint

does not create a new boundary obtuse triangle. In addition, notice that β must be greater than or equal to 30° because if β is less than 30°, α would be less than 30° too and then, the Lepp improvement procedure would not have finished yet.

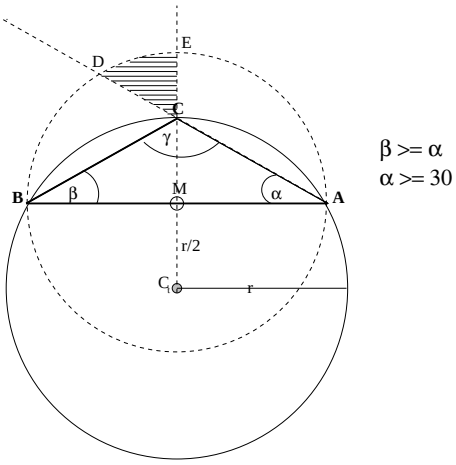

Figure 11: Region of valid points for C

2. The smallest edge is an interior edge. In this case, the boundary constrained angle β must be less than α. We can not apply the same strategy as for the 1-edge boundary obtuse triangles because after two applications of the strategy, a new triangle similar to the original one will be obtained, as shown in Figure 12 (t_o is similar to t_4). One additional problem is that because of the boundary restrictions the minimum angle of this triangle can be less than 30° and consequently, the obtuse angle can be greater than 120°.

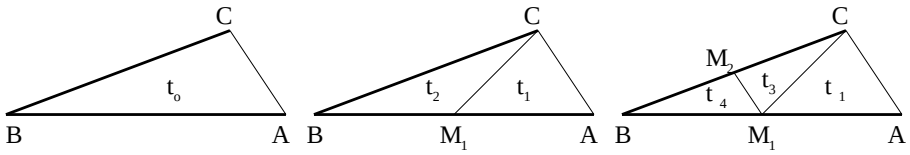

Figure 12: t_o is similar to t_4

The essential ideas of the algorithm to handle case 2 are the followings: An 2-edge boundary isosceles triangle of the largest edges equal to half the smallest boundary edge of the target triangle is constructed (Figure 13(b)). This construction can produce an 1-edge boundary obtuse triangle t_1, which

is in turn eliminated by the Delaunay insertion of the midpoint of the longest edge of t_1 (Figure 13(c)). This can again produce a new boundary obtuse triangle t_1, with largest angle smaller than the previous one and so on. The boundary obtuse triangles are eliminated after the insertion of a finite number of points. Note however that, since the boundary constrained angle can be less

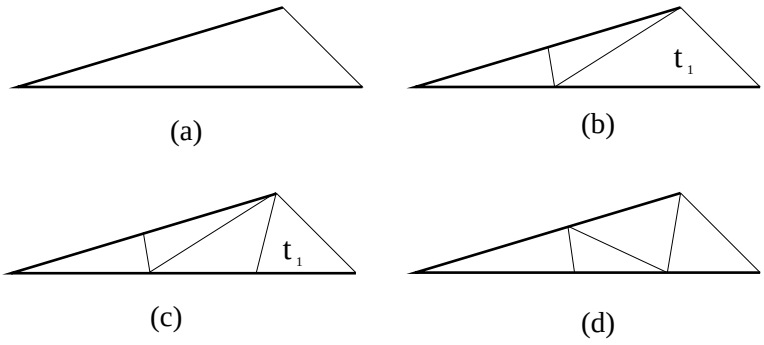

Figure 13: Elimination of 2-edge boundary obtuse triangles

than 30°, some 1-edge boundary obtuse triangles with obtuse angle greater than 120° can be produced. To illustrate see Figure 13.

The algorithm to handle 2-edge boundary obtuse triangles where the smallest edge is an interior edge can be schematically described as shown in Figure 14.

Theorem 8 *Let t be a 2-edge boundary obtuse triangle with interior smallest edge. (1) If the angle of vertex B is greater than β_0 (where β_0 is a constant to be determined later), the obtuse angle is eliminated by insertion of exactly two points by creating a 2-edge boundary isosceles triangle (2) If the angle of vertex B is less than β_0 as shown in Figure 16, an isosceles triangle is created as in point (1) and if 1-edge boundary obtuse triangles are generated, they are eliminated by inserting a number of points bounded by $i \leq 2\frac{\overline{C-M_0}}{\overline{M_0-N}}$.*

Proof. In order to eliminate 2-edge boundary obtuse triangles with interior smallest edge (β is the smallest angle of the triangle), we build a 2-edge boundary isosceles triangle as shown in Figure 15. The construction of an isosceles triangle avoids the propagation of obtuse angles opposite to a boundary edge. The insertion of only two points is required to eliminate the obtuse angle if the angle β is greater than or equal to β_0 because in this case $\gamma_1 \leq 90°$ (see Figure 15). The value of β_0 that produces $\gamma_1 = 90°$ can be found by using the isosceles properties of triangle BNM and the cosines theorem. Thus, the following three equations that relate δ and β are obtained. They allow to compute δ giving values to β.

```
Input: t is a 2-edge boundary obtuse triangle  with smallest
interior angle and T is the current triangulation

Compute the midpoint M of the smallest boundary
edge of t ( See Figure 15)
Compute the point N so that the length of segment BM
is equal to the length of  segment BN
Perform the Delaunay insertion of N and M
(This reduces to join points N and M_j; and points N and C)

S = Φ
if  triangle t_1 of vertices NAC is a 1-edge boundary obtuse angle
     S = {t_1 }
end if

while   S is not empty
     Get one of the triangles t of S
     Perform the Delaunay insertion of the longest edge midpoint
of t
     if  a new triangle t_1 is an 1-edge boundary obtuse triangle
          S = S U {t_1}
     end if
end while
```

Figure 14: Algorithm to eliminate 2-edge boundary obtuse triangles

$$m^2 = 2d^2 - 2d^2\cos(\beta)$$

$$x^2 = m^2 + d^2 2md\cos(90 + \frac{\beta}{2})$$

$$m^2 = d^2 + x^2 - 2dx\cos(\delta)$$

In addition, since α must be greater than β, the condition $\alpha = \beta_0$ is imposed to compute the maximum value of γ_1 for any triangle that satisfies the conditions of this theorem and has a boundary constrained angle β_0. The following relation is thus obtained:

$$\gamma_1 = 180 - 2\beta_o - \delta \leq 90^\circ$$

Numerically, we obtain that if β_o is greater than 32.54°, δ is greater or equal to 24.93°, and then γ_1 is less than 90°.

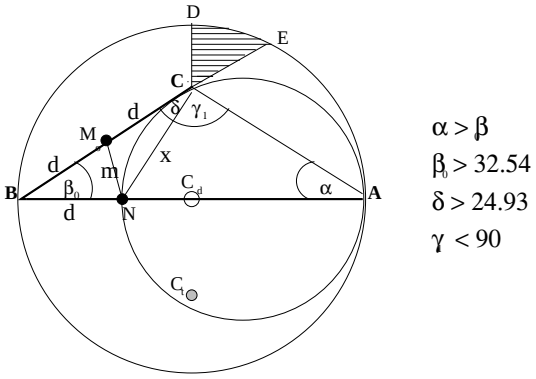

Figure 15: Minimum value of β so that it is not necessary to insert additional points

In case $\beta \leq \beta_0$ as shown in Figure 16(a), the new 1-edge boundary triangle NCA might be obtuse. If this triangle is a boundary obtuse triangle, a boundary edge midpoint is inserted and the Delaunay criteria is applied. After this insertion, new 1-edge boundary obtuse triangles might appear. An upper bound of the number of point insertions can be obtained if we consider that the point insertions finish when the boundary edge is the smallest edge of the 1-edge boundary triangles. Since the smallest edge of the quadritateral $NACM_0$ is NM_0, the number of point insertions on each boundary edge is bounded by $\lceil \frac{\overline{C-M_0}}{\overline{M_0-N}} \rceil$.$\Box$

Corollary 9 *The number of points inserted (V_{2b}) to eliminate N 2-edge boundary obtuse triangles with smallest interior edge is:*

$$V_{2b}(t_j) \leq 2N + 2\sum_{j=1}^{N} \lceil \frac{\overline{C_j - M_{0j}}}{\overline{M_{0j} - N_j}} \rceil$$

where t_j is the 2-edge boundary obtuse triangle j.

5.2 Nonobtuse boundary for PLSG geometries

Boundary obtuse angles opposite to a polygon interface as shown in Figure 17 can be handled in the same way as 1-edge boundary obtuse triangles. Analogously to this previous case, the insertion of the midpoint of the interface

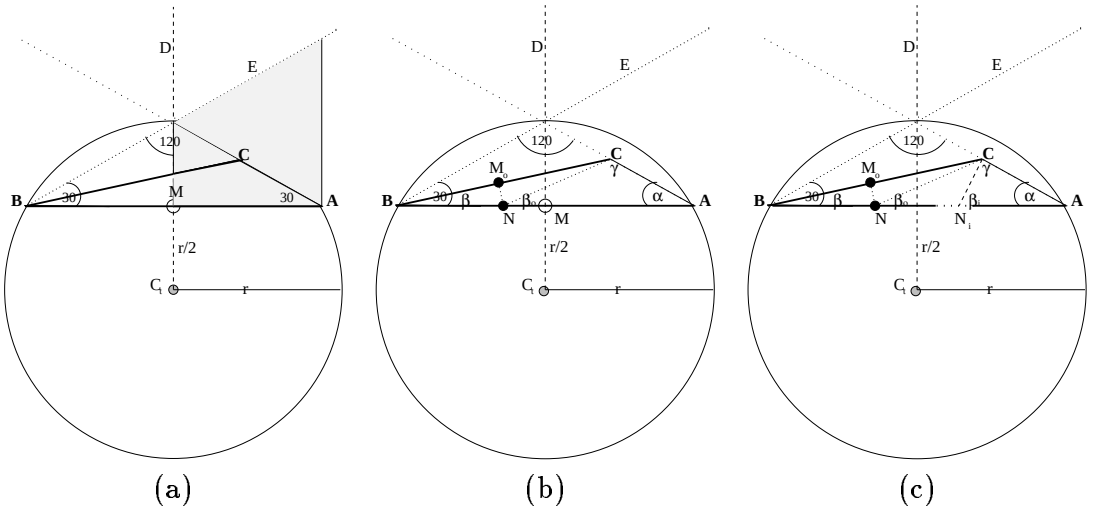

Figure 16: Boundary obtuse triangle with a constrained angle less than β_0

(common) edge AB destroys both obtuse angles and does not generate obtuse angles opposite to the interface edges.

Corollary 10 *The number of vertices (V_{1i}) inserted to eliminate N 1-edge interface obtuse triangles is bounded as follows:* $\frac{N}{2} \leq V_{1i} \leq N$

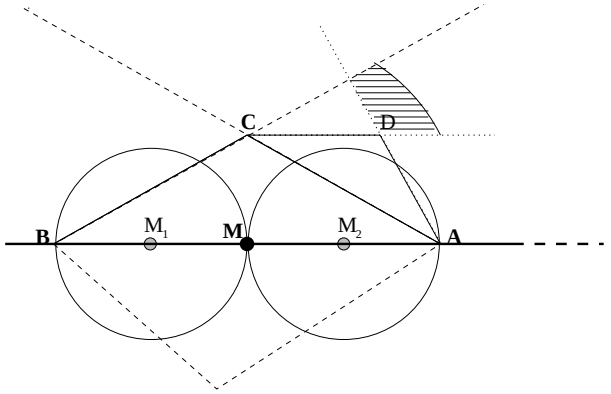

Figure 17: Obtuse angles opposite to a material interface

Figure 18 illustrates the more complex case arising when interfaces whith several interface edges converge to a common vertex A. In this case we eliminate the obtuse angles by inserting the midpoint M of the smallest interface edge and a point N_j on each edge j so that the distance between N_j and A is

equal to the distance between M and A. Thus, isosceles triangles are generated around A. We then eliminate the 1-edge boundary obtuse triangles using the same elimination strategy applied to 1-edge boundary obtuse triangles shown in Figure 14. Since the boundary obtuse triangles are adjacent, the number of points inserted on shared edges is defined by the triangle that requires the highest number of point insertions.

The previous strategy is also applied if some of the triangles of the group of adjacent 2-edge boundary triangles are nonobtuse triangles. Otherwise, the insertion of points to destroy only the 2-edge boundary obtuse angles of the group can produce new 2-edge boundary obtuse triangles in the adjacent triangles that were 2-edge boundary nonobtuse triangles.

Corollary 11 *The number of vertices (V_{2a}) inserted to eliminate N convergent boundary obtuse triangles is:*

$$NV(j) = \lceil \frac{\min(\overline{B_j N_j}, \overline{B_{j+1} N_{j+1}})}{\overline{N_{j+1} N_j}} \rceil, 1 \leq j \leq N$$

$$V_{2a}(A) <= N + 1 + (N+1) \max_{1 \leq j < N} (NV(j), NV(j+1))$$

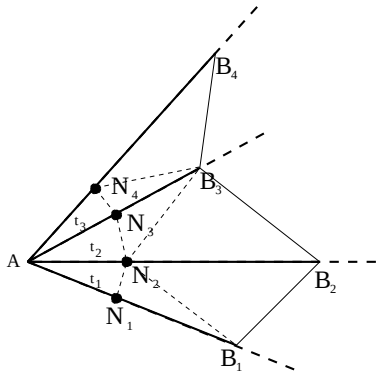

Figure 18: Adjacent boundary obtuse triangles

Proposition 3 *The number of point insertions to eliminate N boundary obtuse angles is $O(N)$.*

Proof. Let be N_{1b} the number of 1-edge boundary obtuse triangles, N_{1i} the number of -edge interface obtuse triangles, N_{2b} the number of isolated 2-edge boundary obtuse triangles and N_{2a} the number of nodes that concentrates

adjacent 2-edge boundary obtuse triangles. The total number of inserted points V is:

$$V <= N_{1b} + N_{1i} + \sum_{j=1}^{N_{2b}} V_{2b}(t_j) + \sum_{j=1}^{N_{2a}} V_{2a}(A_j)$$

Let be N^k the number of triangles associated to the node A_k and t_o be the 2-edge boundary obtuse triangle that requires the highest number of point insertions to eliminate its obtuse angle. In order to identify t_o, we consider each 2-edge boundary obtuse triangle independently. Then, the previous expression can be bound as follows:

$$V <= N_{1b} + N_{1i} + (N_{2b} + \sum_{j=1}^{N_{2a}} N^k) V_{2b}(t_o) = O(N)$$

Corollary 12 *Nonobtuse boundary and interfaces triangles => Delaunay triangulations.*

6 Examples

This section discusses the results obtained by applying the algorithm presented in this paper to several test examples with different geometrical complexity. To illustrate the practical behavior of the algorithm, four test problems of different geometrical complexity have been considered: the right angled spiral of Figure 19(a); the strip geometry with "interior" interface edge of Figure 20(a), the two circle polygon with additional interior interface edges of Figure 21(a) and the polygon with several constrained angles of Figure 22(a).

Tables 1, 2, 3 and 4 summarize the geometrical information of the meshes generated throughout the automatic improvement process. Each table contains information about the number of vertices (vertices), the number of triangles (triangles), the minimum angle (min. angle), the average value of the minimum angles, the maximum angle (max. angle), the average value of the maximum angles and the number of boundary obtuse triangles (b-obtuse triangles) that still remains after applying a Delaunay algorithm (Delaunay), after applying the Lepp-Delaunay strategy (Lepp-Delaunay) and after applying the strategy to eliminate boundary obtuse angles (Final mesh). In particular, when the mesh has 2-edge boundary obtuse triangles with smallest interior edge (triangles whose quality can be only partially improved by the Lepp-improvement procedure because of their boundary constrained angles), the rows that give angle information contain two values: the left one corresponds to the set of Lepp-improvable triangles and the right one considers the set of triangles with boundary constrained angles.

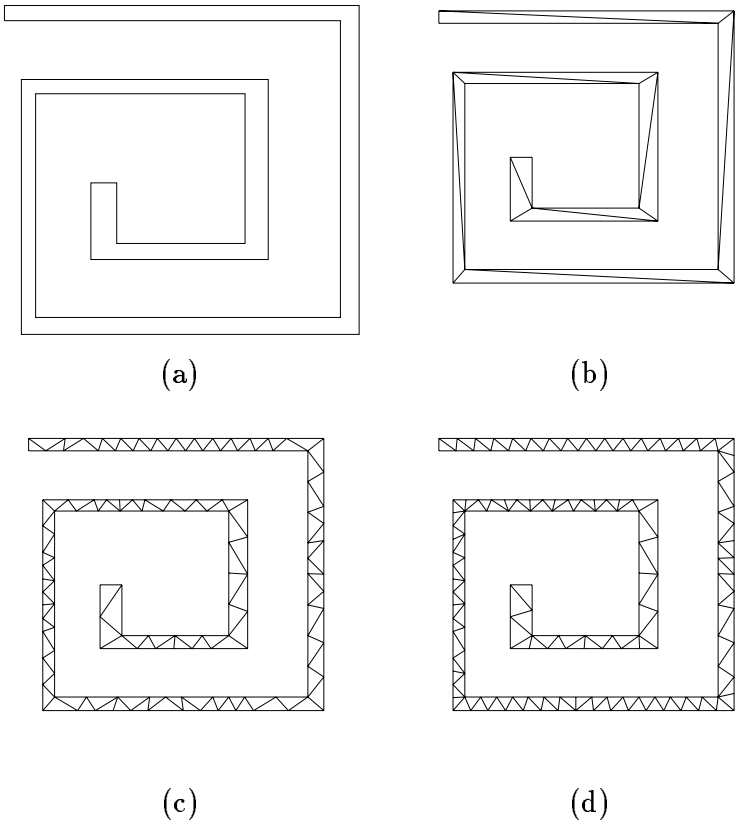

Figure 19: Example 1

Some triangles with angles less than 30° can be introduced while eliminating 2-edge boundary obtuse triangles with smallest interior edge. They are located in the neighborhood of the original boundary obtuse triangles. The number of triangles with minimum angle less than 30° is shown table 3 and 4 close to the number of triangles of the final mesh. For example, in example 3, the number of triangles with boundary constrained angle less than 30° are 4 and the number of triangles with angle less than 30° generated while eliminating the 2-edge boundary obtuse triangles are 16. The number of involved triangles depends on the number of point insertions and on the number of diagonal swapping made to eliminate the boundary obtuse angle.

	Example 1		
	Delaunay	Lepp-Del	Final mesh
vertices	16	130	158
triangles	18	128	156
min. angle	2.59	30.53	30.53
aver. min. angle	6.62	40.76	43.31
max. angle	145.53	111.03	112.52
aver. max. angle	126.00	83.96	80.00
b-obtuse triangles	8	28	0

Table 1: Statistical information for the example 1 (Figure 19)

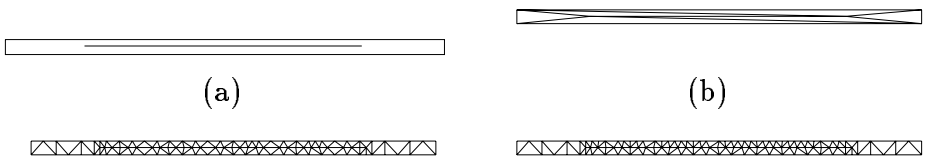

Figure 20: Example 2

	Example 2		
	Delaunay	Lepp-Del.	Final mesh
vertices	6	99	116
triangles	6	128	149
min. angle	1.00	30.77	30.77
aver. min. angle	4.10	43.53	44.72
max. angle	175.52	108.16	106.60
aver. max. angle	144.80	83.65	81.68
b-obtuse triangles	2	21	0

Table 2: Statistical information for the example 2 (Figure 20)

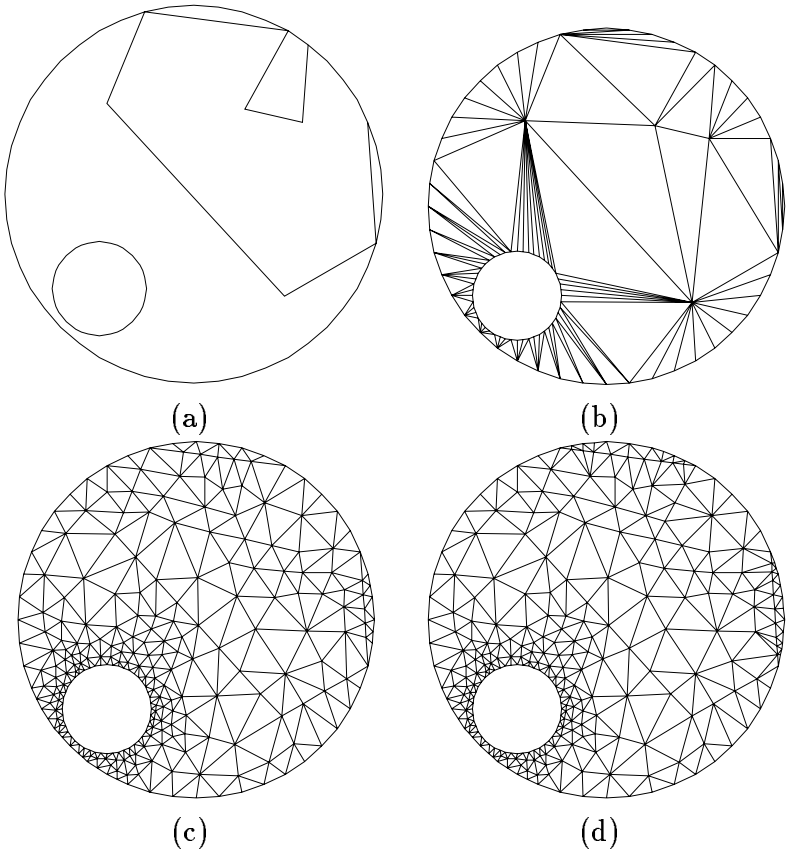

Figure 21: Example 3

	Example 3		
	Delaunay	Lepp-Del	Final mesh
vertices	100	272	291
triangles	104	434	463 (16,4)
min. angle	0.84	30.06-14.99	12.40-14.99
aver. min. angle	15.73	43.15-16.87	42.39-16.87
max. angle	172.49	115.17-129.32	126.82-82.50
aver. max. angle	111.87	79.80-124.87	80.49-82.56
b-obtuse triangles	9	8	0

Table 3: Statistical information for the example 3 (Figure 21)

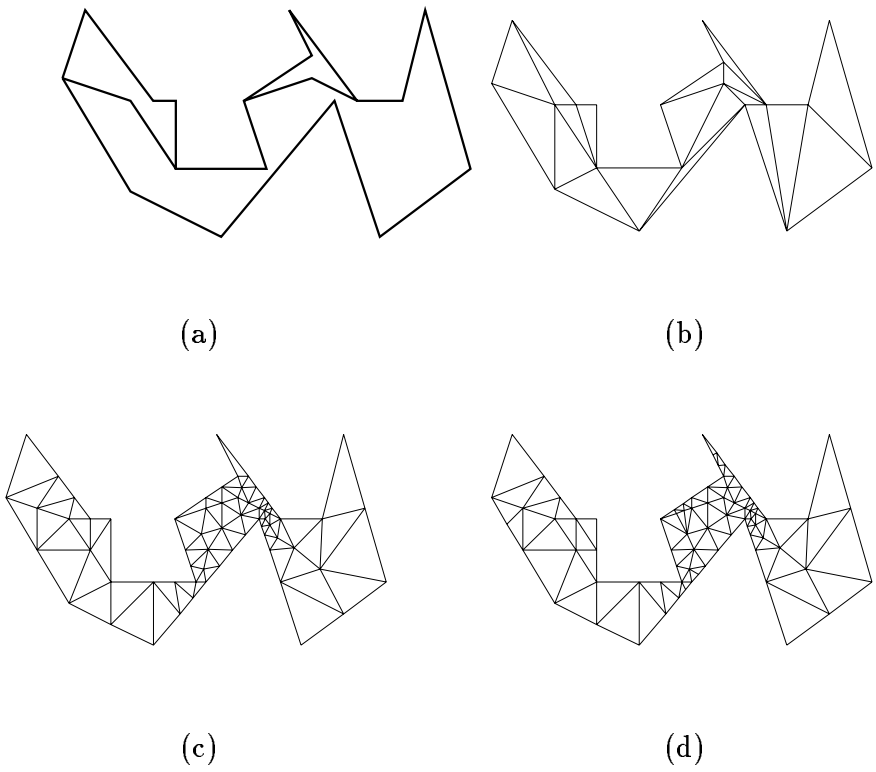

Figure 22: Example 4

	Example 4		
	Delaunay	Lepp-Del.	Final mesh
vertices	19	65	77
triangles	18	80	94 (5,4)
min. angle	5.19	30.34-10.30	17.91-10.30
aver. min. angle	25.91	43.34-20.52	41.88-20.52
max. angle	168.69	112.61-116.56	113.62-90.00
aver. max. angle	105.85	80.29-100.77	80.78-86.32
b-obtuse triangles	10	6	0

Table 4: Statistical information for the example 4 (Figure 22)

Tabla 5 compares the theoretically expected number of point insertions of the postprocess algorithm to eliminate boundary obtuse angles with the number of point insertions obtained in practice. The table shows that the implemented algorithm confirm the expected theoretical results.

Number of point insertions during the elimination of boundary obtuse triangles					
	N_{1b}	N_{1i}	N_{2b}	expected	inserted
Example 1	28	0	0	28 (Cor. 7)	28
Example 2	13	8	0	$10.5 \leq N \leq 21$ (Cor. 10)	17
Example 3	4	0	4	20 (Prop. 3)	19
Example 4	3	0	3	16 (Prop. 3)	12

Table 5: Number of point insertions while eliminating boundary obtuse angles

7 Conclusions

In this paper we present a new automatic algorithm to generate good quality meshes for the control volume discretization and the finite element methods. The resulting triangulations are quality Delaunay triangulations, whose boundary triangles do not have obtuse angles opposite to boundary or interface edges. The algorithm consists of two steps: (1) The generation of good quality constrained Delaunay triangulation. The quality of any mesh is improved using the Lepp-Delaunay strategy: the angles are bounded by 30° and 120°. In practice, the 2-dimensional triangulations obtained is size-optimal [13]. The use of this improvement technique simplifies very much the next step. In addition, the quality mesh has very few boundary obtuse angles. (2) The elimination of boundary obtuse triangles. The postprocess to eliminate boundary obtuse triangles introduces a linear number of points with respect to the number of boundary obtuse triangles. For meshes whose domain geometry does not have boundary constrained angles less than 32.54°, the number of inserted points is bounded by the number of boundary obtuse triangles. Otherwise, the postprocess inserts a finite number of points that is proportional to the number of boundary obtuse triangles.

The postprocess that eliminates boundary obtuse angles guarantees that: (1) if after the Lepp improvement algorithm, the mesh has only 1-edge boundary or interface obtuse triangles, the angles of the triangulation are bounded by 30° and 120°. (2) If the mesh has 2-edge boundary obtuse triangle with boundary constrained angles greater than $\beta_0 = 32.54°$, the angles of the triangulation are also bounded by 30° and 120° except in a number of triangles

equal to the number of 2-edge boundary obtuse triangles. (3) For meshes with any type of boundary obtuse triangles, most of the angles are bounded by 30° and 120°. The few triangles that are not bounded are in the neighborhood of the 2-edge boundary obtuse triangles.

8 Acknowledgment

The first author thanks to Norbert Strecker for a valuable interaction about the subject. The programming of the algorithm was done by Mauricio Palma. This work was supported by Fondecyt project No 1960735 and Fondap AN-1 project.

References

[1] M. R. Pinto. *Comprehensive Semiconductor Device Simulation for Silicon ULSI.* PhD thesis, Stanford University, 1990.

[2] N. Hitschfeld, P. Conti, and W. Fichtner. Mixed Elements Trees: A Generalization of Modified Octrees for the Generation of Meshes for the Simulation of Complex 3-D Semiconductor Devices. *IEEE Trans. on CAD/ICAS*, 12:1714–1725, November 1993.

[3] J. F. Bürgler. *Discretization and Grid Adaptation in Semiconductor Device Modeling.* PhD thesis, ETH Zürich, 1990. published by Hartung-Gorre Verlag, Konstanz, Germany.

[4] G. Heiser. *Design and Implementation of a Three Dimensional, General Purpose Semiconductor Device Simulator.* PhD thesis, ETH Zürich, 1991. published by Hartung-Gorre Verlag, Konstanz, Germany.

[5] Stephan Müller. *An object-oriented approach to multidimensional semiconductor device simulation.* PhD thesis, Swiss Federal Institute of Technology, 1994.

[6] N. Hitschfeld. *Grid Generation for Three-dimensional Non-Rectangular Semiconductor Devices.* PhD thesis, ETH Zürich, Series in Microelectronics, Vol. 21, 1993. PhD thesis published by Hartung-Gorre Verlag, Konstanz, Germany.

[7] G. Garretón, L. Villablanca, N. Strecker, and W. Fichtner. A new approach for 2-d mesh generation for complex device structures. In *NUPAD V - Technical Digest*, Honolulu, USA, June 1994.

[8] Gary L. Miller, Dafna Talmor, Shang-Hua Teng, Noel Walkington, and Han Wang. Control volume meshes using sphere packing: generation, refinement and coarsening. In *Proceedings of the 5th International meshing Roundtable*, pages 47–61, Pittsburgh, Pennsylvania, 1996.

[9] Jim Ruppert Marshall Bern, Scott Mitchell. Linear nonobtuse triangulation of polygons. In *Proc. 10th annu. ACM sympos. computational geometry*, pages 231–241, St.Louis, 1994.

[10] M.C. Rivara. New longest-edge algorithms for the refinement and/or improvement of unstructured triangulations. *International journal for numerical methods in Engineering*, 40:3313–3324, 1997.

[11] M.C. Rivara and G. Iribarren. The 4-triangles longest-side Partition of Triangles and linear Refinement Algorithms. *Mathemathics of Computation*, 65(216):1485–1501, october 1996.

[12] M.C. Rivara and N. Hitschfeld. Geometrical properties of the lepp-delaunay algorithms for the quality triangulation problem. *Department of Computer Science, U. de Chile*, 1997.

[13] J Ruppert. A delaunay refinement algorithm for quality 2-d mesh generation. *Journal of algorithms*, 18:548–585, 1995.