Tuned terminal triangles centroid Delaunay algorithm for quality triangulation

Maria-Cecilia Rivara and Pedro A. Rodriguez-Moreno

Abstract An improved Lepp based, terminal triangles centroid algorithm for constrained Delaunay quality triangulation is discussed and studied. For each bad quality triangle t, the algorithm uses the longest edge propagating path (Lepp(t)) to find a couple of Delaunay terminal triangles (with largest angles less than or equal to 120 degrees) sharing a common longest (terminal) edge. Then the centroid of the terminal quadrilateral is Delaunay inserted in the mesh. Bisection of some constrained edges are also performed to assure fast convergence. We prove algorithm termination and that a graded, optimal size, 30 degrees triangulation is obtained, for any planar straight line graph (PSLG) geometry with constrained angles greater than or equal to 30 degrees.

1 Introduction

Lepp bisection algorithm [10, 3] is an efficient reformulation of previous longest edge algorithm for triangulation refinement, that for each target triangle follows the longest edge propagating path (Lepp) to find a couple of terminal triangles sharing a common longest edge (terminal edge), which are then refined by longest edge bisection. Consequently, local refinement operations are used, and conforming triangulations (where adjacent triangles either share a common edge or a common vertex) are maintained throughout the whole refinement process. Due to the properties of the iterative longest edge bisection of triangles, refined triangulations that maintain the triangulation quality (bounded smallest angle) are obtained, while the

Maria-Cecilia Rivara
Universidad of Chile, Department of Computer Science, Avenida Beauchef 851, Santiago, 8370456, Chile, e-mail: mcrivara@dcc.uchile.cl

Pedro A. Rodriguez-Moreno
Universidad del Bio-Bio, Department of Information Systems, Av. Collao 1202, Concepcion, 4051381, Chile, e-mail: prodrigu@ubiobio.cl

proportion of quality triangles increases as the refinement proceeds. Based on the properties of terminal triangles and terminal edges it was also proved that optimal size triangulations are obtained [3].

A Lepp Delaunay algorithm for quality Delaunay triangulation, based on the Delaunay insertion of the midpoint of the terminal edge, was introduced by Rivara [10] and studied by Bedregal and Rivara [2]. An algorithm based on computing the centroid Q of the terminal triangles which is Delaunay inserted, was presented in [11] without proving termination, neither optimal size property. In this paper we study a tuned, order independent algorithm (where the size of the refined triangulation is almost equal independently of the triangle processing order), based on the Lepp centroid algorithm discussed in [11].

Alternative Delaunay refinement algorithms, based on selecting the circumcenter (or a point close to the circumcenter) of each skinny triangle which is Delaunay inserted in the triangulation have been studied by Ruppert [14], Shewchuk [16], and by Erten an Üngor [5]. Lepp Delaunay algorithms and circumcenter based algorithms have analogous practical behavior, as shown in the empirical study of reference [11], where the Triangle software [16] (without later improvement criteria) was compared with Lepp Delaunay algorithms. It is worth noting however that Lepp based algorithms have the advantage of being order independent, in the sense that they construct triangulations of approximately the same size independently of the processing order of the bad quality triangles. Consequently they are simpler methods than circumcenter based algorithms, easy to implement and easy to parallelize. On the other hand, the implementation of circumcenter algorithms is rather cumbersome, and requires processing triangles in bad-quality order. Section 6.3 of reference [4] discusses several recommendations to implement Ruppert's algorithm efficiently, which include maintaining a queue of skinny and oversized triangles throughout the refinement process.

Lepp algorithms. These are longest edge algorithms formulated in terms of the concepts of terminal edges, terminal triangles and longest edge propagating path [10, 3, 2]. An edge E is a *terminal edge* in triangulation τ if E is the longest edge of every triangle that shares E. The triangles sharing E are called *terminal triangles* (edge AB in Fig 1 (a)). If E is shared by two terminal triangles then E is an interior edge; if E is shared by a single terminal triangle then E is a boundary edge.

For any triangle t_0 in τ, the *longest edge propagating path* of t_0, $\texttt{Lepp}(t_0)$, is the ordered sequence of increasing triangles $\{t_j\}_0^{N+1}$ such that t_j is the neighbor triangle on the longest edge of t_{j-1} and where longest_edge $t_j >$ longest_edge t_{j-1}, for $j = 1, \ldots, N$. The process ends by finding the terminal edge E and a couple of associated terminal triangles t_N, t_{N+1}. In Figure 1 (a), Lepp(t_0) = $\{t_0, t_1, t_2, t_3\}$.

For each target triangle t, the generic Lepp based algorithms find an associated local largest edge shared by a couple of terminal triangles. Then a point is selected inside the terminal triangles (terminal edge midpoint or terminal triangles centroid) and inserted in the mesh. In the Lepp bisection algorithm, the midpoint M of the terminal edge is inserted by longest edge bisection of the terminal triangles as shown in Figure 1 (b). In the Lepp centroid Delaunay algorithm, the centroid Q of the terminal triangles is selected and (constrained) Delaunay inserted in the mesh, as shown in Figure 1 (c). The process is repeated until the target triangle t is destroyed.

Algorithm Generic Lepp-based algorithm
$Input$: triangulation τ, set S of triangles to be refined / improved
$Output$: Refined triangulation τ'

1: **for** each t in S **do**
2: **while** t remains in τ **do**
3: Find $\texttt{Lepp}(t)$, terminal triangles t_1, t_2 and terminal edge E (t_2 can be null)
4: Select point P inside terminal triangles, insert P in the mesh and update S
5: **end while**
6: **end for**

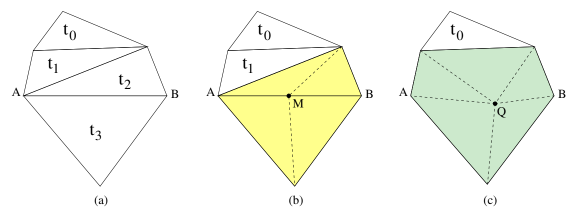


Fig. 1 (a) AB is a terminal edge shared by terminal triangles $\{t_2, t_3\}$ and $\texttt{Lepp}(t_0) = \{t_0, t_1, t_2, t_3\}$; (b) First step of Lepp-bisection algorithm for refining t_0; (c) First step of Lepp Delaunay centroid algorithm.

This paper presents improved and new results in the following senses:

i) In a previous paper, Rivara and Calderon [11] discussed a Lepp Delaunay centroid algorithm where for each selected couple of terminal triangles t_1, t_2 (with non constrained terminal edge) the centroid of the quadrilateral formed by t_1, t_2 is selected and Delaunay inserted in the mesh. In the tuned algorithm of this paper, if t_1 (or t_2) is a bad quality triangle with constrained second longest edge E, the midpoint of E is (constrained) Delaunay inserted in the mesh, (see section 3), which significantly reduces the number of points inserted close to the constrained edges.
ii) In this paper we present new rigorous results on algorithm termination and on the construction of optimal size triangulations, based on the properties of Lepp sequences proved in [3].
iii) We prove that the algorithm produces 30^o quality triangulations for any planar straight line graph (PSLG) geometry with constrained angles greater than or equal 30^o. This is a strong new result. Note that the proof in Ruppert's algorithm requires constrained angles $\geq 90^o$, while the modified algorithm of Shewchuk requires constrained angles $\geq 60^o$.
iv) We prove that the practical behavior of the tuned algorithm is independent of the triangles processing order, which is not the case of circumcircle based algorithms.

More specifically, in this paper we study a tuned Lepp Delaunay centroid algorithm by integrating previous, revisited and new results needed in the algorithm analysis. The following issues are considered:

- The simple insertion of the centroid Q over a couple of Delaunay terminal triangles t_1,t_2, obtained by joining Q with the vertices of t_1,t_2, improves the triangles obtained by longest edge bisection. This is an intermediate operation used in the algorithm analysis. In addition the Delaunay mesh insertion operation of Q improves even more the triangles involved.
- Most of the bad obtuse triangles have largest angle $> 120^o$, and are eliminated by edge swapping, assuming that an edge swapping Delaunay algorithm is used.
- The average Lepp size is small and tends to be 2 as the refinement proceeds. This result was proved for triangulations obtained by the Lepp bisection algorithm and extends to the algorithm of this paper.
- The constrained Delaunay triangulation of any PSLG data defines an intuitive edge distribution function which identifies edge details and non edge details in the PSLG geometry. We prove that the algorithm constructs a graded triangulation adapted to the geometry details. The edge details are not refined unless a close smaller detail induces its refinement.
- We use the simple (constrained) Delaunay triangulation associated with the PSLG data, as an intuitive edge distribution function, to prove termination and optimal size property, instead of using the local feature size function introduced by Ruppert [14].
- Our algorithm does not require the edge encroachment test used in Ruppert's algorithm, but a simple test based on triangle constrained edges.
- The mathematical properties of the mesh operations allows us to prove that 30^o triangulations are obtained for constrained angles $\geq 30^o$. Note that Ruppert algorithm requires 90^o constrained angles [14], and modified algorithm of Shewchuk requires constrained angles $\geq 60^o$ [16].

2 Previous results

The iterative longest edge bisection of individual triangles was studied by Rosenberg and Stenger [13] and by Stynes [18, 19]. This process produces a finite number of non-similar triangles with a bounded smallest angle, while the proportion of good triangles (quasiequilateral triangles) increases as the refinement proceeds.

Definition 1. Given a triangle $t(ABC)$ of vertices A, B, C, and edges $AB \geq BC \geq CA$, the longest-edge bisection of t (or simply bisection of t) is performed by joining the midpoint M of AB with the opposite vertex C (see Fig. 2 (a)).

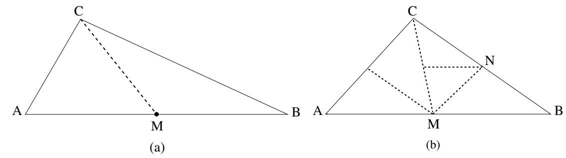


Fig. 2 (a) Longest-edge bisection of triangle $t(ABC)$ (b) First longest edge bisections that define a quasiequilateral triangle $t(ABC)$.

Definition 2. Triangle $t(ABC)$ of edges $AB \geq BC \geq CA$ is *quasiequilateral* if $AC \geq max\{AB/2, CM\}$ and $MC \geq BC/2$ (see Fig 2 (b)).

Note that for quasiequilateral triangles (see Figure 2 (b)) after the first median MC is introduced, the next longest edge bisections only produce medians parallel to the edges of the initial triangle *ABC*, which implies that at most, four similarly distinct triangles are produced. Furthermore the following results hold [13, 18, 19] :

A1. Given any triangle t_0 of smallest angle α_0, the iterative longest edge bisection of t_0 and its descendants produces a finite set $S(t_0)$ of similarly distinct triangles. Furthermore each triangle t in S(t_0) has smallest angle α_t such that $\alpha_t \geq \alpha_0/2$.
A2. For any quasiequilateral triangle t_{qeq}, the triangle set $S(t_{qeq})$ has at most, four similarly distinct triangles, all of which are also quasiequilateral.
A3. For any non quasiequilateral triangle t_0, consider the sequence of triangle sets Q_j defined as follows: $Q_0 = \{t_0\}$, and for j ≥ 1, Q_j is obtained by longest edge bisection of the triangles of Q_{j-1}. Then the triangle sets Q_j improve with j as follows: both the percentage of quasiequilateral triangles and the area of t_0 covered by these triangles, monotonically increase as the iterative refinement proceeds.

The triangulations obtained by the Lepp bisection algorithm are conforming and inherit properties A1, A2, A3 as follows: the iterative local/global use of the Lepp bisection algorithm (and previous longest edge algorithms) produces sequences of nested, refined and conforming triangulations $\{\tau_j\}$ such that B1, B2 hold [10, 3]:

B1. For any triangle t_0 in τ_0, the refined triangles nested in t_0 belong to a finite set $S(t_0)$ of similarly distinct triangles, all of which have smallest angle $\alpha \geq \alpha_0/2$, where α_0 is the smallest angle of t_0.
B2. The refined triangulations $\{\tau_j\}$ improve with j in the following senses: both the percentage of quasiequilateral triangles, and the area covered by these triangles, increase as the refinement proceeds.

More recently Bedregal and Rivara [3] proved that there exists a close relationship between quasiequilateral triangles and terminal triangles (the proportion of terminal triangles increases as quasiequilateral triangles increases), which imply B3. Furthermore bounds on the number of triangle partitions performed inside a triangle in a Lepp sequence [3] (assertion B4) together with B3, implies B5.

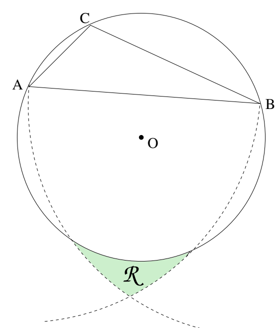
B3. The proportion of terminal triangles increases (approaching 1) as the refinement proceeds and the average length of Lepp(t) tends to be 2 as the refinement proceeds. Furthermore the Lepp Delaunay algorithms inherit the same properties.

B4. The number of longest edge bisections performed in the interior of a triangle t to make it conforming in a refining Lepp sequence, is constant and less than 3 in most cases. This constant is bounded by $O(\log^2(1/\alpha))$ for triangles with arbitrary smallest angle α.
B5. Lepp bisection algorithm produces optimal size triangulations.

Finally, the properties of Delaunay terminal triangles [10, 17], play a crucial role in Lepp Delaunay algorithms, and specifically in the algorithm of this paper. Couples of Delaunay terminal triangles ABC, ABD (see Figure 3) are neighbor triangles that simultaneously satisfy that AB is the common longest edge of the both triangles, and that triangles ABC and ABD are locally Delaunay which implies that vertex D is outside the circumcircle of triangles ABC. Both conditions together imply that vertex D must belong to the shadowed region $\mathscr{R}$ limited by the circumcircle of triangle ABC and the circles of vertices A, B and radius AB. In the case that $\measuredangle ACB = 120^o$, $\mathscr{R}$ reduces to one point D' (triangle $AD'B$ is equilateral). Consequently for $\measuredangle ACB > 120^o$, $\mathscr{R}$ is empty and the following results hold:

Theorem 1. *For any pair of Delaunay terminal triangles t_1, t_2 sharing a terminal edge AB it holds:*
a) Largest angle (t_i) $\leq 2\pi/3$ for i=1,2
b) At most one of the triangles t_1, t_2 is obtuse

Fig. 3 Delaunay terminal triangles ABC, ABD; vertex D belongs to region R.



Since the algorithm of this paper inserts points in the interior of couples of Delaunay terminal triangles, only triangles with largest angle less than or equal to 120^o can become a terminal triangle throughout the algorithm processing.

Definition 3. We will say that t is a PD terminal triangle (potentially a Delaunay terminal triangle) if the $largest\ angle(t) \leq 120^o$.

3 The tuned algorithm

We first introduce the following mesh operation: for bad quality terminal triangle t with constrained second longest edge CB, the constrained Delaunay insertion of the midpoint of CB is performed (see Figure 4). This operation reduces the number of interior points inserted close to the constrained edges. Note that the previous Lepp Delaunay midpoint algorithm requires this operation to guarantee convergence [10]. The previous Lepp Delaunay centroid algorithm does not use this operation since the centroid selection avoids the introduction of collinear points, but introduces more points than the tuned algorithm of this paper [11].

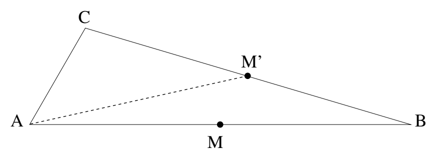


Fig. 4 For constrained second longest edge CB, the midpoint of CB is constrained Delaunay inserted in the mesh.

Algorithm Tuned Terminal_Triangles_Centroid_Delaunay_Algorithm
Input: CDT τ associated with PSLG data, angle tolerance θ_{tol}
Output: Refined triangulation τ_f with angles $\geq \theta_{tol}$.

```
Find S set of bad quality triangles
for each t in S (while S ≠ ∅) do
    while t remains unrefined do
        Use Lepp(t) to find Delaunay terminal triangles t1, t2 and terminal edge E
        if E is constrained (this includes t2 null) then
            Perform Constrained Delaunay insertion of midpoint of E
        else
            if there exists t (t1 or t2) such that α_t < θ_tol and
               second longest edge L is constrained then
                Perform constrained Delaunay insertion of midpoint of L
            else
                Compute centroid Q of terminal triangles, and perform constrained
                Delaunay insertion of Q
            end if
        end if
        Update S
    end while
end for
```

4 Better angle bounds on first bisections of triangles

We show that the first longest edge bisection of a triangle produces a better triangle t_B (see Figure 5) and a bad obtuse triangle t_{OB} [17]. Assume the triangle of Figure 5 where $AB \geq BC \geq AC$ with the notation shown in this figure.

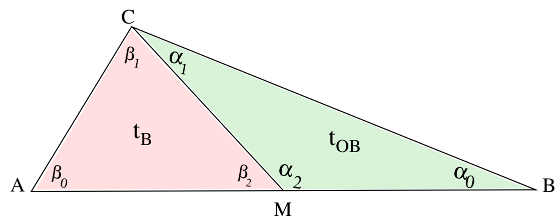


Fig. 5 Notation for longest edge bisection. Angles in longest edge bisection of triangle ABC with $AB \geq BC \geq AC$.

It is rather easy to see that if t is a right angled triangle then $\alpha_1 = \alpha_0$, $\beta_1 = \beta_0$, $AM = CM$, while if t is an acute triangle then $\alpha_1 < \alpha_0$, $\beta_1 < \beta_0$, $AM < CM$; and if t is an obtuse triangle then $\alpha_1 > \alpha_0$, $\beta_1 > \beta_0$, $AM > CM$. These properties allow proving most of the assertions of Lemma 1 [17]. The bound on α_1 follows from the strong property $A1$.

Lemma 1. *The following angle bounds hold [17].*
(a) $\alpha_1 \geq \alpha_0/2$, $\alpha_2 \geq 90^o$, $\beta_2 \leq 90^o$, $\beta_1 \geq \pi/6$, $\beta_1 \geq \alpha_1$
(b) $\beta_2 = \alpha_0 + \alpha_1 \geq 3\alpha_0/2$
(c) if t is obtuse, then $\alpha_1 > \alpha_0$ *and* $\beta_2 \geq 2\alpha_0$
(d) if t is acute, then $\alpha_1 < \alpha_0$ *and* t_B *is acute*

Next we introduce the taxonomy of Figure 6, which is a variation of those presented in references [17, 9]. This is obtained by fixing the longest edge AB of triangle ABC considering $AB \geq BC \geq CA$, and studying which is the longest edge of triangle AMC and the longest edge of triangle CMN (see Figure 2 (a)) according to the position of vertex C. Thus, the half circle of vertex M and radius AM separates obtuse and acute triangles. Arcs AR and MR respectively correspond to isosceles triangles with edges $AM = CM$ and edges $AC = AM$; while arc ZW corresponds to the circle of center $\tilde{N}$ (where $A\tilde{N} = AB/3$) and radious $A\tilde{N}$, corresponding to the triangles for which $CB = 2CM$. The set of quasiequilateral triangles is the union of region $\mathscr{R}_1$ (acute triangles) and region $\mathscr{R}_2$ (obtuse triangles). Finally arc AW corresponds to points C for which the largest angle is equal to $120°$, defined by the circle of center W' and radious WW', where points W, W' are symmetric with respect to line AB. By studying the boundaries of regions $\mathscr{R}_1$ and $\mathscr{R}_2$ it is easy to see that $\mathscr{R}_1 \bigcup \mathscr{R}_2$ correspond to quasiequilateral triangles and that most of these triangles (vertex C by above line SW) have the smallest angles $\geq 30°$. Only for vertex C in region SZW, smallest angle $> 27.88°$ (the worst case corresponds to $C = Z$ where $tg(\alpha_0(Z)) = \sqrt{7}/5$). Note that most of the triangles of $\mathscr{R}_3$ also have the smallest angle $\geq 30°$.

Lemma 2. *(a) For quasiequilateral triangles in region $TSWX$, $\alpha_0 \geq 30^o$; (b) For quasiequilateral triangles in region SZW, $\alpha_0 > 27.88$.*

Next we extend the notation of Figure 5. We will call $\alpha_0(t)$, $\alpha_1(t)$ to the angles α_0, α_1 obtained by longest edge bisection of $t(ABC)$; in addition we call $\alpha_0(t_B)$, $\alpha_1(t_B)$ to the α_0, α_1 angles obtained by longest edge bisection of t_B.

Lemma 3. *Given any triangle t, then:*
(a) If $\alpha_0 \leq 30^o$, then $\alpha_1 \geq 0.79\alpha_0$ and $\beta_2 \geq 1.79\alpha_0$. Furthermore, the ratio α_0/α_1 increases (α_1 approaching α_0) while α_0 decreases.
(b) If $\alpha_0 \geq 30^o$, then $\alpha_0(t_B) \geq 30^o$ and t_B is quasiequilateral.
(c) If t is quasiequilateral with $\alpha_0 \geq 30^o$, then $\alpha_0(t_B) \geq 30^o$ and $\alpha_0(t_{OB}) \geq 27.88^o$.
(d) If t is a PD-terminal triangle then t_B is a PD-terminal triangle, and $\alpha_0(t_B) \geq Min\{3/2\alpha_0(t), 30^o\}$. Furthermore, if $\alpha_0 \geq 20^o$, then smallest angle$(t_B) \geq 30^o$.

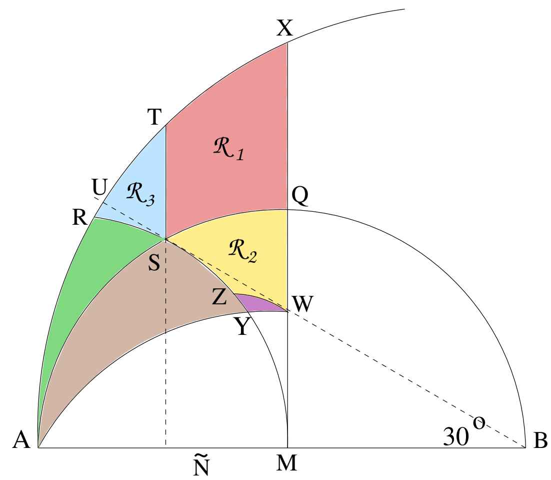


Fig. 6 Taxonomy on longest edge bisection of triangles $t(ABC)$ with $AB \geq BC \geq CA$.

Proof. The proof of item (a) follows by studying the case of acute triangles of region UAS in Figure 6, where the worst case corresponds to point U for which $\alpha_1 \approx 23.79^o$. To prove assertion (d) we consider the case where t is acute. Note that in Figure 5, $\alpha_0(t_B) = Min\{\beta_0, \beta_1, \beta_2\}$, where $\beta_2 \geq 3/2\alpha_0(t)$. On the other hand, β_0 is the smallest angle of t_B in Figure 6, when edge AM is the shortest edge of the t_B, which occurs for acute t with C either in region $\mathscr{R}_1$ or in region $\mathscr{R}_2$; and the worst case occurs for the equilateral triangle where $\beta_2 = 30^o$ (see Figure 5). Assertion (b) follows from Lemma 1, while assertion (c) follows from Lemma 2. □

In Lemma 4 we further quantify the notion that the triangle t_B in Figure 5 is a better triangle than t following the ideas of reference [17]. In Lemma 5 we further characterize PD-terminal triangles. The non PD terminal triangles correspond to very obtuse triangles of region WAM in Figure 6

Lemma 4. *If t_B is acute and $\alpha_0 \leq 30^o$, then $\alpha_1(t_B) \geq min\{1.4\alpha_0(t), 30^o\}$.*

The proof is rather complex and can be found in reference [17].

Lemma 5. *Given any PD-terminal triangle t. Then:*
a) If t is acute and $\alpha_0 \leq 30^o$, then t_{OB} is a non-PD terminal triangle.
b) If t is obtuse and $\alpha_0 > 22^o$, then t_{OB} can be a PD terminal triangle.

Proof. Part a) follows from the fact that for acute triangles $\alpha_1 < \alpha_0$, which in turn implies $\alpha_1 + \alpha_2 < 60^o$ and consequently t_{OB} is a non-PD terminal triangle. Part b) follows from the fact that for obtuse triangles, $\alpha_1 > \alpha_0$. In [12] it was proved that largest angle equal to 120^o and $\alpha_1 + \alpha_0 = 60^o$ implies that $\alpha_0 > 22^o$. Thus, only for some triangles with $\alpha_0 > 22^o$ it can hold $\alpha_1 + \alpha_0 > 60^o$ and t_{OB} can be a PD terminal triangle. □

5 Improvement properties of the centroid insertion

In what follows we consider that triangle t is good if $\alpha_0 \geq 30^o$. The algorithm of this paper in general performs (constrained) Delaunay insertion of the centroid Q of couples of Delaunay terminal triangles (both triangles with largest angle $\leq 120^o$). To analyze triangle improvement, the following intermediate simple centroid insertion is needed: consider the centroid Q of a couple of Delaunay terminal triangles as shown in Figure 7. The simple centroid insertion is then performed by joining Q with the four vertices, instead of performing longest edge bisections. This operation corresponds to a Laplacian smoothing of the terminal edge midpoint M and improves the triangles obtained by longest edge bisection. Note that the Laplacian smoothing works very well for convex geometries [6, 7], and couples of terminal triangles always define a convex quadrilateral.

Lemma 6. *The simple centroid insertion has the following properties: (i) It improves the worst angle obtained by the longest edge bisection; (ii) It avoids the reproduction of a bad quality triangle; (iii) It improves the lightly bad angles obtained by the longest edge bisection of good triangles.*

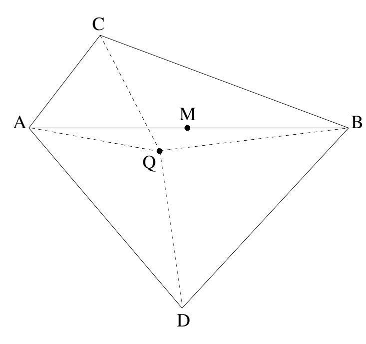


Fig. 7 Centroid refinement of terminal triangles ABC, ADB.

6 Algorithm analysis

Consider a general PSLG (planar straight line graph) geometry, defined by a set of points, edges and eventually polygonal objects defining exterior boundaries and interior holes. Any PSLG geometry has edge details and non-edge details. Edge details are small edges in the PSLG data, while non edge details are defined by two close isolated interior points, an isolated point close to an input edge, two edges with close points, constrained angles either over the boundaries or interior to the geometry, and vertices over these angles. For an illustration see Figure 8 (a).

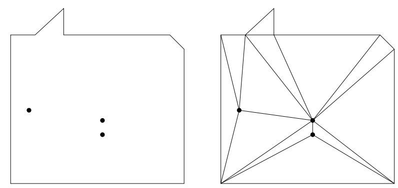


Fig. 8 (a) PSLG geometry; (b) constrained Delaunay triangulation identifies edge details and non edge details.

Note that the constrained Delaunay triangulation of the input PSLG data intuitively defines an edge distribution function to which an optimal size good quality triangulation should be adapted. More specifically this identifies edge details and non-edge details by means of skinny triangles with associated (constrained or non constrained) small edges, very obtuse triangles with largest angled vertex close to an edge data, and triangles with constrained smallest angle. Figure 8 (b) shows the constrained Delaunay triangulation of the example of Figure 8 (a). We will prove that the algorithm of section 5 produces a graded quality mesh with smaller good quality triangles around the PSLG geometry details. The following Lemma assures that 30^o constrained angles always produce quality triangles:

Lemma 7. *Let t be any triangle with* 30^o *constrained angle. Then (a) If t is obtuse, the longest edge bisection of t produces quality triangles (part (c) of Lemma 1); (b) If t is acute the tuned algorithm inserts three points in the constrained edges, as shown in Figure 9 to produce quality triangles (*δ *is the worst angle* $> 34^o$*).*

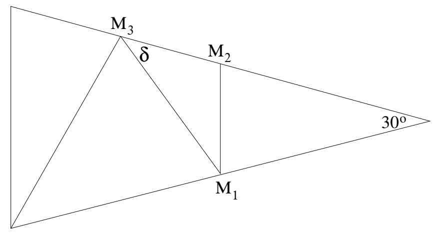


Fig. 9 Worst case of acute isosceles 30^o triangle.

Theorem 2. *Consider any PSLG geometry with constrained angles* $\geq 30^o$ *and the input constrained Delaunay triangulation* τ_0 *associated with the PSLG data. Then for angle tolerance* $\theta_{tol} = 30^o$*,*
(a) The algorithm finishes with a graded 30^o *constrained Delaunay triangulation.*
(b) The final triangulation is size optimal.

Proof. Given $\theta_{tol} = 30^o$, consider the bad triangles with angles less than 30^o. To prove part (a), we will study five cases of triangles processing:

Case 1. Non PD terminal triangles. Each bad quality triangle t (with largest angle $> 120^o$ either with one or two bad angles) is a non PD terminal triangle and consequently is eliminated by swapping edge AB either by processing t or by processing a Lepp-neighbor bad quality triangle. This operation produces locally more equilateral triangles.

Case 2. Bad PD terminal triangles. Consider a couple of non constrained Delaunay terminal triangles. Let $t(ABC)$ with $AB \geq BC \geq AC$ be the worst triangle in the couple with $\alpha_0 < 30^o$. Then:
• According to part(a) of Lemma 3, the longest edge bisection of t introduces midpoint M of AB and a better triangle $t_B(ACM)$ with $\alpha_0(t_B) \geq 1.79\alpha_0$, and a bad obtuse triangle t_{OB}. The simple centroid insertion of Q corresponds to the Laplacian smoothing of point M, which improves the worst angles of t_{OB} (introduced by the longest edge bisection) and avoids the repetition of a triangle similar to triangle ABC. This operation can be seen as a first step of the Delaunay insertion of point Q.

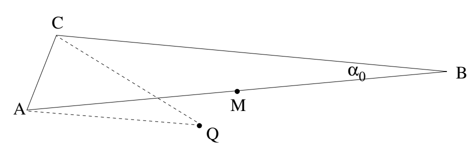


Fig. 10 Triangle ABC with $\alpha_0 < 30^o$. Better triangle ACQ and CQB are obtained with respect to the longest edge bisection.

• The centroid Q is Delaunay inserted in the mesh (see Figure 10). If triangle CQB is a non PD terminal triangle, then triangle CQB is eliminated (and improved) by swapping edge CB, either when Q is Delaunay inserted (if there exists a vertex inside the (big) circumcircle of triangle CMB), or by later processing CBQ, or by processing a neighbor bad quality triangle. If triangle CQB is a PD terminal triangle and still bad, then by processing triangle CQB this can become a terminal triangle and the centroid $\tilde{Q}$ of CQB and its neighbor triangle is inserted, which improves the angles.
• According to part (d) of Lemma 3, for $\alpha_0 < 20^o$, CAQ can still be bad. Then for small α_0, a finite sequence of points Q_i can be inserted in the mesh until a good triangle CAQ_n is obtained (see Figure 11). The process finishes without refining edge AC (AC is a local smallest edge), unless a close smaller edge induces neighbor refinement. See the termination analysis for more details.

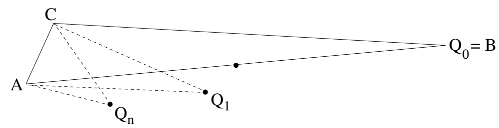


Fig. 11 Points Q_i are introduced until triangle CAQ_n is good.

Case 3. Terminal triangles with constrained edges. For Delaunay terminal triangles with constrained terminal edge, the constrained Delaunay insertion of the terminal edge midpoint is performed and the improvement process continues. For bad triangles with constrained second edge E, the simple constrained Delaunay insertion of midpoint of E is performed, which accelerates convergence (section 3).
Case 4. Couples of good Delaunay terminal triangles. For couples of good quality Delaunay terminal triangles with smallest angles $\geq 30^o$, the centroid Q of the terminal quadrilateral is inserted, which produces more equilateral triangles than those obtained by longest edge bisection. This is equivalent to a Laplacian smoothing of the terminal edge midpoint introduced by the longest edge bisection of the terminal triangles. This operation improves eventual angles less than 30^o that could have been introduced by the longest edge bisection.
Case 5. Triangles with 30^o constrained angles. Here, good quality triangles are obtained inside t by inserting one or three points over the constrained edges.

Termination. The proof on termination is based on the fact that for skinny triangles, the smallest edge AC is never refined, unless there exists a smaller bad quality triangle t^* such that Lepp(t^*) contains triangle AQ_nC (see Figure 12). Thus the algorithm stops when every triangle of local smallest edge in τ_0 becomes good (smallest angle $\geq 30^o$), and every remaining intermediate bad quality triangle t is processed or eliminated by edge swapping; and every intermediate almost good terminal triangle is improved by centroid insertion. This produces a good quality triangulation graded around the *PSLG* geometry details.

Optimal size property. This follows from the termination reasoning together with the fact that the average Lepp size tends to be 2 as the refinement proceeds. □

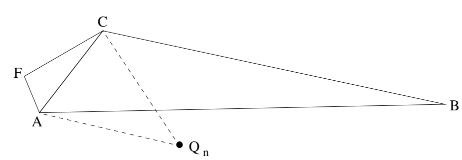


Fig. 12 Neighbor triangle ACF induces refinement of triangle AQ_nC to obtain a graded refined triangulation around edge FA.

Theorem 3. *The algorithm is order independent, where the mesh size is approximately the same by processing the bad triangles in arbitrary order.*

Proof. The set of terminal edges introduces a mesh partition so that every triangle in the partition reaches the same terminal edge. □

7 Empirical study and concluding remarks

In Table 1 we compare our algorithm with results reported by Shewchuk [16] on Ruppert's algorithm (without the off-center preprocess of Üngor). Next we present results on the behavior of the Delaunay centroid algorithm for the six geometries of the Figure 13. Table 2 includes final mesh sizes for $\theta_{tol} = 30^o, 33^o, 34^o, 35^o$ obtained with our algorithm. See the final triangulations for $\theta_{tol} = 30^o$ for these examples in Figure 13. Table 3 compares the number of triangles obtained with our software, with respect to those obtained with the current version of Triangle [15] which processes skinny and oversized triangles in order, and includes a boundary preprocess technique due to Üngor [5] to minimize the size of the final triangulation. A negative number means our software introduces less triangles than Triangle, while the -∞ symbol means that Triangle does not converge.

Table 1: Algorithms comparison, Key test case, $\theta_{tol} = 33^o$

	Del centroid algorithm	Ruppert's algorithm [16]	
Triangle processing	without order	without order	Ordering triangles
Final Mesh size	229	450	249

It should be noted that: (i) our results are not far from those obtained by the current optimized version of Triangle; (ii) our software works properly until $\theta_{tol} = 35^o$ for all the test cases, while Triangle fails for 50% of the test cases (-∞ symbol) for $\theta_{tol} = 35^o$; iii) Note that for the key test case and $\theta_{tol} = 33^o$, our algorithm produces a final triangulation with 229 triangles against 450 triangles obtained with pure Ruppert algorithm (first-come first split bad quality triangle) and 249 triangles by always processing the worst existing triangle, as reported by Shewchuk [16].

Table 2: Mesh sizes for Delaunay centroid algorithm as a function of θ_{tol}

	Superior lake	Neuss geometry	Square	Chesapeake bay	Long rectangle	Key geometry
	size(τ_0)	size(τ_0)	size(τ_0)	size(τ_0)	size(τ_0)	size(τ_0)
	528	3,070	9	14,262	2	54
θ_{tol}	size(τ_f)	size(τ_f)	size(τ_f)	size(τ_f)	size(τ_f)	size(τ_f)
30	1,835	8,338	54	36,803	19	170
33	2,273	9,939	65	45,883	22	229
34	2,512	11,054	70	52,027	25	262
35	3,017	12,742	81	63,138	27	349

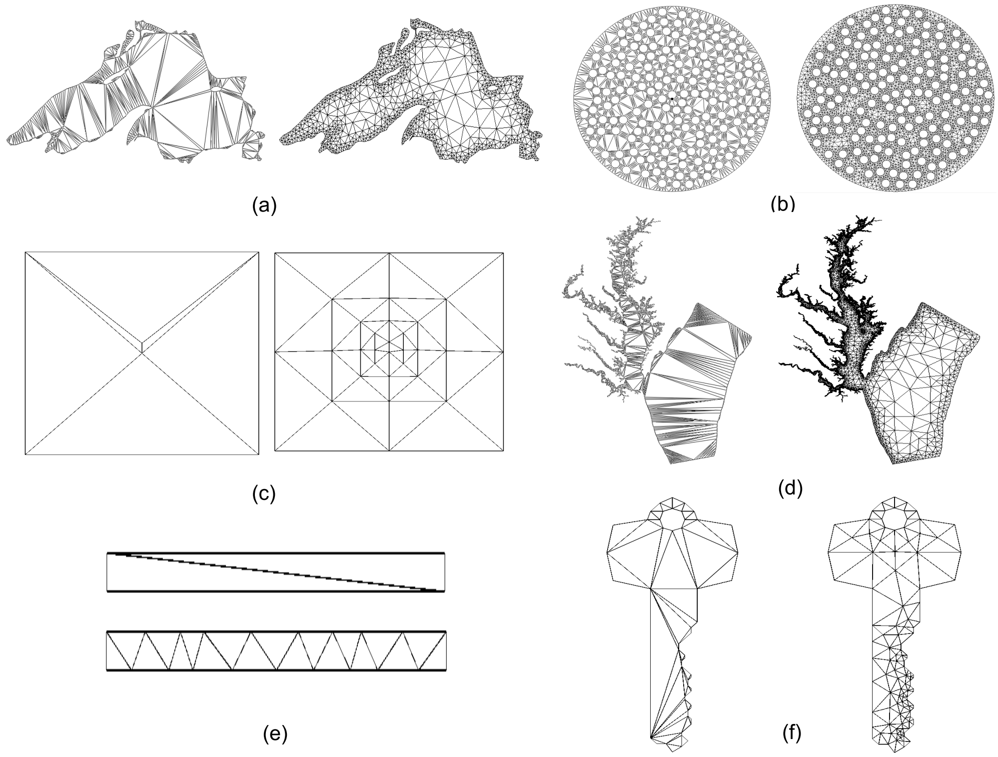


Fig. 13: Quality meshes for $\theta_{tol} = 30^o$ (a) Superior lake shape; (b) Neuss shape; (c) Square with skinny triangles; (d) Chesapeake bay shape; (e) Long rectangle; (f) Key shape.

Table 3: Percentage of triangles added with respect to current version of Triangle[a].

θ_{tol}	Superior lake	Neuss geometry	Square	Chesapeake bay	Long rectangle	Key geometry
30	0.44	13.18	24.07	4.82	-15.79	22.94
33	-5,28	12,01	16.92	2.41	0.00	10.92
34	-5.29	-2.70	20.00	3.61	-68.00	-8.78
35	-∞	-∞	24.69	-∞	-207.41	5.44

[a] Triangle processes skinny and oversized triangles in order and uses a boundary preprocess step.

Furthermore, for all the test cases, the average Lepp size is less than 3 from the beginning and quickly becomes less than 2.5, as the refinement proceeds. The algorithm is an easy to implement, order independent, robust method, suitable for use in adaptive finite element methods where good quality meshes are needed to assure convergence. With an adequate triangle data structure that keeps information on neighbor triangle, the refinement is of cost $O(N)$ where N is the number of points inserted.

In three dimensions, Balboa, Rodriguez-Moreno and Rivara [1] have introduced a simple and effective mesh improvement algorithm for tetrahedral meshes, which generalizes some of the ideas presented in this paper. Note that for any tetrahedron t, Lepp(t) corresponds to a submesh with several (more than two) terminal edges, and associated terminal stars. We call terminal star to a set of tetrahedra that share a common (terminal) largest edge in the mesh. Two new terminal star operations are alternatively used in the mesh improvement algorithm: the simple centroid insertion of the terminal star (that generalizes the simple centroid insertion of section 5 in two dimensions), and swapping of the terminal edge as described by Freitag and Olliver-Gooch [8], but selectively applied to the terminal stars. The operation that most improves the mesh is performed whenever significant improvement is achieved. For more details see reference [1]

Acknowledgements Work partially supported by Departamento de Ciencias de la Computación, Universidad de Chile, Departamento de Sistemas de Información, Research Group GI150115/EF, and Research Project DIUBB 172115 4/R, Universidad del Bio-Bio.

References

1. Balboa, F., Rodriguez-Moreno, P., Rivara, M.C.: Terminal star operations algorithm for tetrahedral mesh improvement. In Proceedings 27h International Meshing Roundtable, Albuquerque, USA (2018)
2. Bedregal, C., Rivara, M.C.: New results on Lepp-Delaunay algorithm for quality triangulations. Procedia Eng. 124, 317–329 (2014)
3. Bedregal, C, Rivara, M.C.: Longest-edge algorithms for size-optimal refinement of triangulations. Computer-Aided Des. 46, 246–251 (2014)
4. Cheng, S.W., Dey, T.K., Shewchuk, J.R.: Delaunay mesh generation. CRC computer and information science series, CRC Press, Computational Geometry. 22(1-3), 21–74 (2013)
5. Erten, H., Üngor, A.: Quality triangulations with locally optimal steiner points. SIAM J. Sci. Comput. 31, 2103–2130, (2009)
6. Field, D.A.: Laplatian smoothing and Delaunay triangulations, Commun. Appl. Numer. Meth. 4(6), 709–712 (1988)
7. Freitag, L.A.: On combining laplatian and optimization-based mesh smoothing techniques, ASME Applied Mechanics Division-Publications-AMD. 220, 37–44 (1997)
8. Freitag, L.A., Olliver-Gooch, C.: Tetrahedral mesh improvement using swapping and smoothing, Int. J. Numer. Methods Eng. 40, 3979–4002 (1997)
9. Gutierrez, C., Gutierrez, F., Rivara, M.C.: Complexity of the bisection method. Theoretical Computer Science. 382(2), 131–138 (2007)
10. Rivara, M.C.: New longest-edge algorithms for the refinement and/or improvement of unstructured triangulations. Int. J. Mumer. Methods Eng. 40(18), 3313–3324 (1997)

11. Rivara, M.C., Calderon, C.: Lepp terminal centroid method for quality triangulation. Computer-Aided Des. 42(1), 58–66 (2010)
12. Rivara, M.C., Hitschfeld, N., Simpson, R.B.: Terminal edges Delaunay (smallest angle based) algorithm for the quality triangulation problem, Computer-Aided Des. 33(3), 263-277 (2001)
13. Rosenberg, I.G., Stenger, F.: A Lower Bound on the Angles of Triangles Constructed by Bisecting the Longest Side. Math. Comput. 29(130), 390–395 (1975)
14. Ruppert, J.: A delaunay refinement algorithm for quality 2-dimensional mesh generation. J. Algorithms, 18(3), 548–585 (1995)
15. Shewchuk, J.R.: Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator, In Applied Computational Geometry (M.C. Lin and D. Manocha eds.), Lecture Notes in Computer Science, 1148, 203–222 (1996)
16. Shewchuk, J.R.: Delaunay refinement algorithms for triangular mesh generation. Computational Geometry, 22(1-3), 21–74 (2002)
17. Simpson, R.B., Rivara, M.C.: Geometrical mesh improvement properties of Delaunay terminal edge refinement, Technical report CS-2006-16, The David Cheriton School of Computer Science, University of Waterloo, 536–544 (2006)
18. Stynes, M.: On Faster Convergence of the bisection method for certain triangles. Math. Comput. 33(146), 717–721 (1979)
19. Stynes, M.: On faster convergence of the bisection method for all triangles. Math. Comput. 35, 1195–1201 (1980)