International Journal of Computational Geometry
& Applications
Vol. 15, No. 1 (2005) 25–54
© World Scientific Publishing Company

World Scientific
www.worldscientific.com

WHEN AND WHY DELAUNAY REFINEMENT ALGORITHMS WORK

GARY L. MILLER
*Computer Science Department, Carnegie Mellon University, 5000 Forbes Avenue
Pittsburgh, PA 15213, USA
glmiller@cs.cmu.edu*

STEVEN E. PAV
*Department of Mathematics, University of California at San Diego, MC0112
La Jolla, CA 92093-0112, USA
spav@ucsd.edu*

NOEL J. WALKINGTON
*Department of Mathematics, Carnegie Mellon University, 5000 Forbes Avenue
Pittsburgh, PA, 15213-3890, USA
noelw@andrew.cmu.edu*

Received 9 December 2003
Revised 19 November 2004
Communicated by Alper Üngör, Guest Editor

ABSTRACT

An "adaptive" variant of Ruppert's Algorithm for producing quality triangular planar meshes is introduced. The algorithm terminates for arbitrary Planar Straight Line Graph (PSLG) input. The algorithm outputs a Delaunay mesh where no triangle has minimum angle smaller than about 26.45° except "across" from small angles of the input. No angle of the output mesh is smaller than $\arctan\left[(\sin\theta^*)/(2-\cos\theta^*)\right]$ where θ^* is the minimum input angle. Moreover no angle of the mesh is larger than about 137°, independent of small input angles. The adaptive variant is unnecessary when θ^* is larger than 36.53°, and thus Ruppert's Algorithm (with concentric shell splitting) can accept input with minimum angle as small as 36.53°. An argument is made for why Ruppert's Algorithm can terminate when the minimum output angle is as large as 30°.

Keywords: Mesh generation; Ruppert's Algorithm; computational geometry; triangular.

1. Introduction

The success of the finite element method depends in part on the quality of the mesh of the problem domain. A good lower bound on minimum angle of the mesh can guarantee the well-conditioning of the stiffness matrix, thus ensuring reasonable convergence of iterative-based solvers.[1] In the case where a large minimum output angle cannot be guaranteed, a bound on maximum angle of the mesh is desirable, since this criterion is essential for interpolation accuracy.[2]

The Delaunay Refinement Algorithm is one of the first algorithms for generating meshes of guaranteed quality. First described by Ruppert, the algorithm accepts a set of points and a set of segments, augments the point set with Steiner points, and returns the Delaunay Triangulation of the augmented set. For suitable input, the triangulation will conform to the input, has no angle smaller than some parameterizable κ (which is no larger than $\arcsin \frac{1}{2\sqrt{2}} \approx 20.7°$), and will exhibit "good grading," *i.e.*, short edges in the triangulation are attributable to nearby input features which are close together. The number of triangles in the output is within a constant of optimal.[3]

The algorithm has the advantage of being relatively easy to state and implement, and has been the object of great scrutiny and interest. Since its introduction, the algorithm and the analysis of the algorithm have been improved and modified: the class of known acceptable input has been expanded[4]; a variant algorithm has been developed to handle small input angles[5]; the algorithm has been adapted to accept curved input[6,7]; it also has been generalized to higher dimensions.[4,8,9,10,11]

Ruppert's original analysis required that no input segments meet at acute angles, and guaranteed that no angle in the output was smaller than a parametrizable $\kappa < \arcsin \frac{1}{2\sqrt{2}}$. As $\kappa \nearrow \arcsin \frac{1}{2\sqrt{2}}$, the proved bound on the number of Steiner Points approaches infinity,[3] though this behaviour is *not* seen experimentally; rather, the Delaunay Refinement Algorithm is often run with κ as large as $\pi/6$, or larger, without diverging. The input condition has since been relaxed to a $\pi/3$ lower bound on input angles.[5,8] The algorithm has been observed to terminate on some input with smaller (in some cases much smaller) input angles.

Shewchuck demonstrated an alteration of the algorithm, the so-called "Terminator," which accepts input with arbitrary minimum angle, θ^*, producing Delaunay meshes with no output angle smaller than $\arcsin \left[\sin\left(\frac{\theta^*}{2}\right)/\sqrt{2}\right]$. This variant is adaptive in the sense that it leaves some small angles in the output mesh, while most angles are larger than $\arcsin \frac{1}{2\sqrt{2}}$. The location of the small output angles cannot be determined very much beyond the statement that they are "near input angles less than ... 60°." Moreover, the analysis of this scheme comes without grading guarantees, and thus no optimality claim.[5]

We describe an alteration of the algorithm which produces meshes where the minimum angle of each output triangle is greater than $\arcsin 2^{-7/6} \approx 26.45°$, except possibly triangles with a short edge opposite[a] an input angle $\theta < 36.53°$; in this case, the output angle is no less than $\arctan\left(\frac{\sin\theta}{2-\cos\theta}\right)$. Moreover, in spite of the potential of arbitrarily small output angles, this algorithm can guarantee that no

[a] In this context, "opposite" means the endpoints of the edge are on distinct input segments subtending angle less than about 36.53°. That is, *e.g.*, triangle Δpqr could be in the output mesh with $\angle pqr < \arcsin 2^{-7/6}$ only if p and r are points on distinct input segments subtending angle smaller than 36.53°. See Lemma 7.

output angle is *larger* than around $\pi - 2\arcsin\frac{\sqrt{3}-1}{2} \approx 137.1°$. In this sense the algorithm contrasts favorably with the Terminator, which has no upper bound other than the naïve one of $\pi - 2\arcsin\left[\sin\left(\frac{\theta^*}{2}\right)/\sqrt{2}\right]$, which deteriorates when θ^* is small. Moreover, our algorithm comes with grading and optimality guarantees, and is fairly simple.

In the case where $\theta^* \geq 36.53°$, our analysis shows that the variant algorithm is unnecessary, and that Ruppert's original algorithm with circular shell splitting comes with the same output and optimality guarantees.

In this work the strategy of Shewchuk is employed, *i.e.*, termination is proved without showing good grading.[4] The output of the algorithm *is* well-graded, and thus the number of Steiner Points is optimal[12]; however, the simpler termination result is here presented since a relatively accessible and complete proof may be given in a small amount of space. The interested reader is encouraged to refer to the more detailed exposition in Ref. [12].

2. The Meshing Problem

The meshing problem is described in terms of the input to the algorithm and the expected conditions on the output. The input to the mesher is defined as follows:

Assumption 1 (Input). The input to the meshing problem consists of a finite set of points, $\mathcal{P} \subseteq \mathbb{R}^2$, and a set of segments $\mathcal{S}$ such that

(i) the two endpoints of any segment in $\mathcal{S}$ are in $\mathcal{P}$,
(ii) any point of $\mathcal{P}$ intersects a segment of $\mathcal{S}$ only at an endpoint,
(iii) the intersection of two segments of $\mathcal{S}$ is either the empty set or a common endpoint, and
(iv) the boundary of the convex hull of $\mathcal{P}$ is the union of segments in $\mathcal{S}$.

Let Ω denote the convex hull of the input, and let $0 < \theta^* \leq \pi/3$ be a lower bound on the angle between any two intersecting segments of the input.

Items (i)-(iii) characterize $(\mathcal{P}, \mathcal{S})$ as a Planar Straight Line Graph (PSLG); item (iv) can always be satisfied by augmenting an arbitrary PSLG which does not satisfy it with a bounding polygon (typically a rectangle). The restriction that $\theta^* \leq \pi/3$ is merely for convenience; asserting a larger lower bound does not give better results.

Assumption 2 (Output). The algorithm outputs sets of points, segments, triangles, $\mathcal{P}', \mathcal{S}', \mathcal{T}'$, respectively, satisfying:

(i) **Complex:** The output collectively forms a simplicial complex.
(ii) **Delaunay:** Each triangle of $\mathcal{T}'$ has the Delaunay property with respect to $\mathcal{P}'$.
(iii) **Conformality:** $\mathcal{P} \subseteq \mathcal{P}'$, and for every $s \in \mathcal{S}$, s is the union of segments in $\mathcal{S}'$.
(iv) **Quality:** There are few or no "poor-quality" triangles in $\mathcal{T}'$.

(v) **Cardinality:** Few Steiner points have been added, *i.e.*, $|\mathcal{P}' \setminus \mathcal{P}|$ is small.

One passable definition of item (iv) is that there are some reasonably large constants $0 < \alpha \leq \omega \leq \frac{\pi+\alpha}{4}$ such that for every triangle $t \in \mathcal{T}'$, no angle of t is smaller than α or larger than $\pi - 2\omega$. However, such a guarantee is not consistent with conformality of the triangulation (item (iii)) when the input contains angles less than α. A weaker definition is that most triangles satisfy the above condition, and those that do not (i) are describably near an input angle of size θ, (ii) have no angle smaller than $\theta - \mathcal{O}\left(\theta^2\right)$, and (iii) have no angle larger than $\pi - 2\omega$. This definition is presented because it is satisfied by some algorithm herein described.

3. The Algorithm

We describe a whole class of algorithms, which we collectively refer to as "the" Delaunay Refinement Algorithm. This class contains Ruppert's original formulation,[3] as well as the "incremental" version.[8]

We introduce the algorithm along with some terminology: we suppose that the algorithm maintains a set of "committed" points, initialized to be the set of input points, $\mathcal{P}$. The algorithm also maintains a set of "current" segments, initialized as the input set, $\mathcal{S}$. The algorithm will "commit" points to the set of committed points. At times the algorithm will choose to "split" a current segment; this is achieved by removing the segment from the set of current segments, adding the two half-length subsegments which comprise the segment to the set of current segments, and committing the midpoint of the segment. The word "midpoint" should be taken to mean one of these segment midpoints for the remainder of this work, to distinguish them from the other kind of Steiner Point, which will be called "circumcenters."

The algorithm has two high-level operations, and will continue to perform these operations until it can no longer do so, at which time it will output the committed points, the current segments and the Delaunay Triangulation of the set of committed points. For convenience, we say that a segment is "encroached" by a point p if p is inside the diametral circumball of the segment. Then the two major operations are as follows:

> (CONFORMALITY) If s is a current segment, and there is a committed point that encroaches s, then split s.
>
> (QUALITY) If a, b, c are committed points, the circumcircle of the triangle Δabc contains no committed point, triangle Δabc has an angle smaller than the global *minimum output angle*, κ, and the triangle's circumcenter, p, is in Ω, then attempt to commit p. If, however, the point p encroaches any current segment, then do not commit to point p, rather in this case split one, some, or all of the current segments which are encroached by p.

It should be clear that if the algorithm terminates then every segment of the set $\mathcal{S}$ has been decomposed into current segments, none of which are encroached by committed points, and thus have the Delaunay property with respect to the final point set, and are thus present in the output Delaunay Triangulation. The algorithm clearly never adds any points outside Ω. It is simple to show that if the algorithm terminates, no triangle in the Delaunay Triangulation has an angle smaller than the minimum output angle κ, though we omit the proof.[12]

In the Adaptive Delaunay Refinement Algorithm, the operation (QUALITY) is replace by the operation (QUALITY$'$):

> (QUALITY$'$) If a, b, c are committed points, the circumcircle of the triangle Δabc contains no committed point, $\angle acb < \hat{\kappa}$, the circumcenter, p, of the triangle is inside Ω and either (i) both a, b are midpoints on distinct input segments sharing input endpoint x, and $\angle axb > \pi/3$, or (ii) a, b are not midpoints on adjoining input segments, then attempt to commit p. If, however, the point p encroaches any current segment, then do not commit to point p, rather in this case split one, some, or all of the current segments which are encroached by p.

Note that the operation (QUALITY$'$) may be applied with respect to a given triangle Δabc for any of its three angles. In summary, the algorithm removes angles smaller than $\hat{\kappa}$ except when the opposite edge spans a small angle in the input, in which case the small output angles are ignored. For this variant we call $\hat{\kappa}$ the *output angle parameter*; the output mesh may well contain angles smaller than $\hat{\kappa}$. We will let α be the minimum angle in the output mesh.

The heuristics involved with determining which operation to perform when and on which segment or poor-quality triangle are not relevant to our discussion. This is not to say that they might not affect ease of implementation, running time, cardinality of the final set of committed points, parallelizability, etc. A common heuristic (and the one chosen by Ruppert and others) is to prefer conformality operations over quality operations, which likely results in a smaller output, and which simplifies detecting that a circumcenter is outside of Ω. A description of a member of this class of algorithms would have to include some discussion of how to figure out which current segments are encroached, which triangles are suitable for removal via the quality operation, how to deal with degeneracy, etc. We do not concern ourselves with these details (though see Refs. [4, 8, 13, 14]).

3.1. *When is adaptivity necessary?*

We here make the claim that the Delaunay Refinement Algorithm is as good as its adaptive variant when the latter is used with a small output angle parameter $\hat{\kappa}$. Alternatively, we can claim that the Adaptive Delaunay Refinement Algorithm is not necessary when the minimum input angle, θ^*, is reasonably large.

These vague claims follow from the following *Gedankenexperiment*: Suppose the Adaptive Delaunay Refinement Algorithm is run with output angle parameter $\hat{\kappa}$. Furthermore suppose it can be shown that for a certain kind of input that the algorithm terminates, and that the output mesh has wonderful properties, including the wonderful property that no angle in the output mesh is less than $\hat{\kappa}$. Since the algorithm never attempts to remove a triangle with a minimum angle greater than

$\hat{\kappa}$, and because the algorithm *does* remove any triangle with minimum angle smaller than $\hat{\kappa}$, then the (QUALITY′) operation could essentially be restated, in this case, as "attempt to commit the circumcenter of any triangle with minimum angle smaller than $\hat{\kappa}$." But this is the (QUALITY) operation. Thus if the non-adaptive Delaunay Refinement Algorithm were applied to this kind of input, with $\kappa = \hat{\kappa}$, then it will terminate, outputting a mesh with the same wonderful properties, including a minimum angle of $\hat{\kappa}$.

In this notice we first examine the adaptive variant; our claim allows us to use the results to describe the behaviour of the regular Delaunay Refinement Algorithm. Thus, the following analysis should be read with a tacit understanding that it can be applied to the Delaunay Refinement Algorithm as well, if κ is set propertly. For example, it will be shown that if an input with $\theta^* \approx 36.53°$ conforms to Assumption 3, then the Adaptive Delaunay Refinement Algorithm with $\hat{\kappa} = 26.45°$ will terminate, leaving no angle in the output mesh smaller than $\hat{\kappa}$, and no angle larger than $\pi - 2\hat{\kappa}$. Then we can immediately claim that the Delaunay Refinement Algorithm (*i.e.*, Ruppert's Algorithm) with $\kappa = 26.45°$ will also terminate on the same input, and with the same output and grading guarantees.

Thus the adaptive variant is only necessary when θ^* is small, say smaller than about 36.53°. When θ^* *is* small, the adaptive variant will remove small angles where this is possible, *i.e.*, away from small input angles.

4. Preliminaries

Some preliminary definitions and results are essential to the exposition. First there is the matter of terminology: if p is a committed point that was the midpoint of a segment, we say this segment is the "parent" segment (or parent subsegment) of p; the "radius" of a segment is half its length, while the radius associated with a midpoint is the radius of its parent segment; any segment derived from a segment $s \in \mathcal{S}$ by splitting is a "subsegment" of (or on) s; segments in $\mathcal{S}$ which share an endpoint are nondisjoint; distinct nondisjoint segments are said to be "adjoining."

Throughout this work, we let $|x - y|$ denote the Euclidian distance between points x and y. For a segment S, we let $|S|$ denote the length of the segment. Local feature size is defined in terms of the input, and is the classical definition due to Ruppert:

Definition 1 (Local Feature Size). For a point $x \in \mathbb{R}^2$, the *local feature size* at x, relative to an input PSLG, $(\mathcal{P}, \mathcal{S})$, is the minimum r such that a closed ball of radius r centered at x intersects at least two disjoint features of $\mathcal{P} \cup \mathcal{S}$. The local feature size is a Lipschitz function, *i.e.*, $\text{lfs}(x) \leq |x - y| + \text{lfs}(y)$.

This definition is illustrated in Figure 1. For the proof we require an extra condition on the input:

Assumption 3. In addition to those of Assumption 1 we make the following assumption:

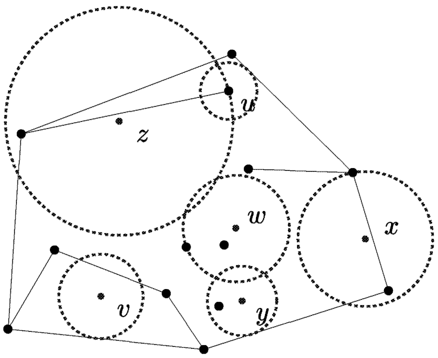


Fig. 1. For a number of points in the plane, the local feature size with respect to the given input is shown. About each of the points u, v, w, x, y, z is a circle whose radius is the local feature size of the center point. The point u is an input point.

(i) If S_1, S_2 are two adjoining input segments that meet at angle other than π, then they have the same length modulo a power of two, that is $\frac{|S_1|}{|S_2|} = 2^k$ for some integer k.

It is simple to show that this assumption can be satisfied by the addition of no more than $2\,|\mathbb{S}|$ augmenting points, effectively redefining the input; we will briefly consider this matter in Section 9. We will also argue that Ruppert's strategy of splitting on concentric circular shells obviates this additional assumption.[3]

5. Midpoint-Midpoint Interactions

Ruppert noted that one way his algorithm could fail was due to infinite cascades of segment midpoints each encroaching on an adjoining subsegment; the prescribed cure was concentric shell splitting, which puts input into a form which satisfies Assumption 3 on an as-needed basis.[3] To simplify the proof, we assume the input satisfies this assumption up-front, then ease the restriction later. In this section we show how this assumption can prevent infinite cascades of midpoints.

First we quote a useful lemma; the proof is purely geometric.[12]

Lemma 1. *Given two rays, R and R' from a point x with angle θ between them, suppose there is a ball of radius r with center p on ray R such that the ball does not contain x but does contain a point q of R'. Then if $\pi/4 \leq \theta < \pi/2$,*

$$\frac{|q-x|}{|p-x|} \leq \frac{|q-x|}{r} < \frac{|q-x|}{|p-q|} \leq 2\cos\theta.$$

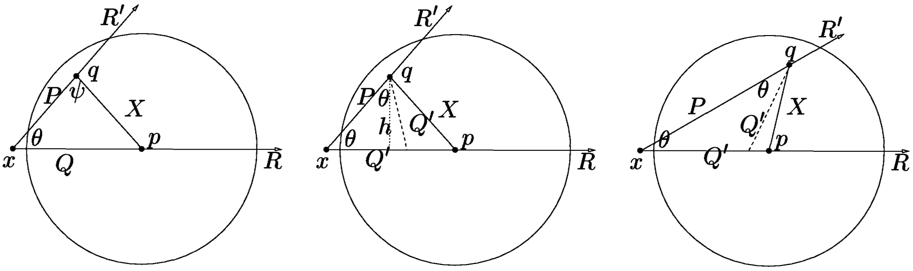
If $0 < \theta < \pi/4$, then only the inequality

$$\frac{|q-x|}{|p-x|} < 2\cos\theta$$

can be asserted.

Proof. Letting P, Q, X be as in Figure 2, first note that $X < r \leq Q$ because x is not inside the ball (which has radius r), but q is; thus for the "large" angle case, it suffices to show only that $P/X \leq 2\cos\theta$. Using the sine identity, we find that $\sin\theta \leq \sin\psi$, implying that $\theta \leq \psi$. We can then draw an isosceles triangle of base angle θ and base (x, q), with side lengths Q'. In the case where $\pi/4 \leq \theta$, the apex angle of this isosceles triangle will be acute, so the altitude h, the leg Q' and the leg X are ordered left to right as shown in Figure 2(b), and thus $Q' \leq X$. Using the cosine relation it is easy to show that $P/Q' = 2\cos\theta$ and thus $P/X \leq 2\cos\theta$, as desired.

In the "small" angle case, the apex angle may be obtuse. However, since $\theta < \psi$, we have $Q' < Q$. Thus $2\cos\theta = P/Q' > P/Q$, as desired. □



(a) as stated

(b) bounded by the isosceles

(c) the small angle case

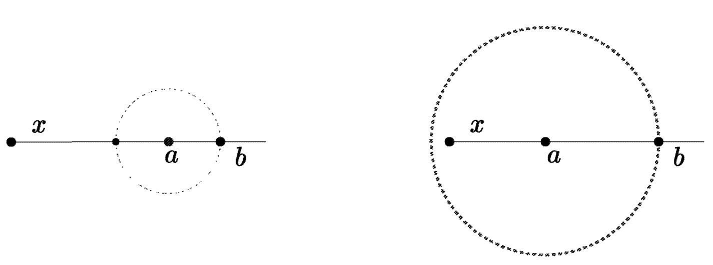
Fig. 2. Proof of Lemma 1; The lemma as stated is shown in (a). It can be shown that $\theta \leq \psi$, so we may draw the isosceles triangle, as in (b) with base angle θ to get the desired bound. The altitude is also drawn in (b), and both triangle legs will be to its right in the order shown. The case where $\theta < \pi/4$ is shown in (c); in this case the ordering of the legs relative to the altitude is not fixed, and only a weaker result is obtained.

A geometric argument follows which helps us establish that radii don't "dwindle" when one midpoint encroaches on an adjoining segment. The lemma can be seen as a mild improvement on Lemma 1 when the input satisfies Assumption 3.

We present a few geometric claims.

Claim 5.1. *Let (a, b) be a subsegment of an input segment which has endpoint x. Let $|x - a| < |x - b|$. Then either $x = a$ or $|a - b| \leq |x - a|$.*

Proof. See Figure 3. Suppose that a is distinct from x. Then a must be a midpoint of some subsegment of radius at least $|a - b|$. However, $|x - a|$ is at least this radius, *i.e.*, $|a - b| \leq |x - a|$. □



(a) Possible Case

(b) Impossible Case

Fig. 3. The argument of Claim 5.1 is shown. When (a, b) is a subsegment on an input segment with endpoint x, such that $0 < |x-a| < |x-b|$, we show that $|a-b| \leq |x-a|$, as shown in (a). The case illustrated in (b) is impossible since a would have to be the midpoint of a subsegment which actually contained the endpoint x. In both figures we show the diametral circle of the subsegment of which a is the center.

Claim 5.2. *Let (a, b) be a subsegment of an input segment which has endpoint x. Suppose p is a point on an input segment which shares the endpoint x that encroaches on the diametral circle of (a, b). Assume that $|x-a| < |x-b|$, and let θ be the angle between the two input segments. Then $|x-a| < |x-p| \cos\theta$, and $\theta < \frac{\pi}{2}$. Also we can claim $|x-p| < |x-b|$. Moreover, if m is the midpoint of (a, b), and r is its radius, then $|x-m| \sin\theta < r$.*

Proof. The gist of this claim is shown in Figure 4. Since p encroaches the diametral circle of (a, b), then so does it's projection onto the line containing the segment, p'. Thus $|x-a| < |x-p'|$. But $|x-p'| = |x-p| \cos\theta$. This implies that $\cos\theta$ is strictly positive, so $\theta < \frac{\pi}{2}$. Since the circle centered at x of radius $|x-b|$ contains the diametral circle of (a, b), it contains the point p, so then $|x-p| \leq |x-b|$.

Since p encroaches (a, b), the radius of the diametral circle must be at least the distance from m to the line segment containing p, which is $|x-m| \sin\theta$. □

The proof of the following claim is by simple induction and is omitted.

Claim 5.3. *Let (a, b) be a current segment on an input segment (x, y). Then $\log_2 \frac{|x-y|}{|a-b|}$ is a nonnegative integer. Moreover $\frac{|x-a|}{|a-b|}$ is either zero or is an integral power of two, as is $\frac{|y-a|}{|a-b|}$, $\frac{|x-b|}{|a-b|}$, and $\frac{|y-b|}{|a-b|}$.*

Lemma 2. *Given three noncollinear points, x, p, q, with $|x-q| \leq |x-p|$ then $|p-q| \geq 2|q-x| \sin\frac{\theta}{2}$, where $\theta = \angle qxp \leq \pi$.*

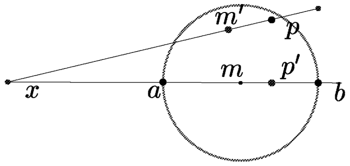


Fig. 4. The argument of Claim 5.2 is shown. Letting $\theta = \angle axp$, by definition of sine and cosine, $|m - m'| = |x - m| \sin\theta$, and $|x - p'| = |x - p| \cos\theta$, where m', p' are projections of the points m, p, onto the opposing segment. Thus $|x - a| \le |x - p| \cos\theta$, and the radius of the circle is at least $|x - m| \sin\theta$. Part of the circle $\mathcal{C}_2$ centered at x of radius $|x - b|$ is shown. Since $\mathcal{C}_2$ contains $\mathcal{C}_1$, the diametral of (a, b), then $|x - p| \le |x - b|$.

Proof. Let $L = \frac{|x-p|}{|x-q|} \ge 1$. Using the cosine rule on Δxpq,

$$\begin{aligned}
|p-q|^2 &= |x-p|^2 + |x-q|^2 - 2\,|x-p|\,|x-q|\cos\theta. \\
&= (1+L^2)\,|x-q|^2 - 2L\,|x-q|^2\cos\theta \\
&\ge 2L\,|x-q|^2 - 2L\,|x-q|^2\cos\theta \\
&= 2L\,|x-q|^2\,(1-\cos\theta),
\end{aligned}$$

where we have used that $1+L^2 \ge 2L$. Using $L \ge 1$, we obtain $\frac{|p-q|}{|x-p|} \ge \sqrt{2(1-\cos\theta)}$. It is a simple exercise to show that $2\sin\frac{\theta}{2} = \sqrt{2(1-\cos\theta)}$ for $\theta \in [0,\pi]$. □

Lemma 3. *Suppose that the input conforms to Assumption 3. Let p be the midpoint of a segment which is encroached by a committed point, q, on an adjoining input segment. Let r_p be the radius associated with p, and r_q that of q. Then $r_q \le r_p$, and moreover,*

$$|p-q| \ge 2r_q \sin\frac{\theta}{2},$$

where θ is the angle between the two input segments.

Proof. Let $(x, y), (x, z)$ be the two input segments containing, respectively, p, q. Let (a, b) be the subsegment of which p is midpoint. Let (c, d) be that for which q is midpoint. Assume that a is closer to x than b is, and assume c is closer to x than d is. It may be the case that $x = a$, or $x = c$.

By Claim 5.3, $\log_2 \frac{|x-y|}{|a-b|}$, and $\log_2 \frac{|x-z|}{|c-d|}$ are nonnegative integers. By Assumption 3, and since $\theta \ne \pi$, $\log_2 \frac{|x-y|}{|x-z|}$ is an integer. Thus $\log_2 \frac{|a-b|}{|c-d|} = \log_2 \frac{r_p}{r_q} = j$ is also an integer. We wish to show that it is nonnegative.

By Claim 5.2, $|x-a| < |x-q| < |x-b|$, so that $|x-a| < |x-c| + r_q < |x-a| + 2r_p$. Using Claim 5.3 shows that $k = \frac{|x-a|}{|a-b|} = \frac{|x-a|}{2r_p}$ is a nonnegative integer, as is, *mutatis mutandis*, $l = \frac{|x-c|}{2r_q}$. Thus $2kr_p < (2l+1)r_q < 2(k+1)r_p$, or $2^{j+1}k < (2l+1) < 2^{j+1}(k+1)$, and so

$$\frac{2l+1}{2^{j+1}} - 1 < k < \frac{2l+1}{2^{j+1}}$$

If j is a negative integer, then 2^{j+1} is a power of two no greater than 1; in particular it divides any integer, thus $\frac{2l+1}{2^{j+1}} = m$ is an integer. This gives the contradiction that $m - 1 < k < m$ for integer m, k. Thus j is a nonnegative integer, or $r_p \geq r_q$.

For the second part, by Lemma 2, $|p - q| \geq 2(|x - q| \wedge |x - p|) \sin \frac{\theta}{2}$. Clearly $|x - p| \geq r_p \geq r_q$, and $|x - q| \geq r_q$, so the result $|p - q| \geq 2r_q \sin \frac{\theta}{2}$ holds, as desired. □

Recall that our termination proof will show there is some global lower bound on nearest neighbor distance of newly committed points. Thus Lemma 3 takes care of the case of midpoint-midpoint interactions because it assures that the radius associated with p is no smaller than that associated with q; one way of viewing this is to say that radii do not "dwindle" when a midpoint "causes" another midpoint to be committed.

6. Circumcenter Sequences

We next consider the problem of circumcenters. Our proof will look at circumcenters as just another way that segment midpoints are committed. That is we will consider a maximal sequence of circumcenter commissions as somehow "causing" a midpoint to be committed. The following definition formalizes this notion.

Definition 2. A *circumcenter sequence* is a sequence of points, $\{b_i\}_{i=0}^{l-1}$ such that for $i = 1, 2, \ldots, l - 1$, b_i is the circumcenter of a triangle in which b_{i-1} is the more recently committed endpoint of an edge opposite an angle less than $\hat{\kappa}$. The point b_0 may be an input point or segment midpoint.

For $i = 0, 1, \ldots, l - 2$, let a_i be the *other* endpoint of the short edge of which b_i is the more recently committed endpoint. In the case where a_0, b_0 are both input points, they are committed simultaneously; we imagine a total order on input points which determines the tie. Both a_0, b_0 may be midpoints on distinct, nondisjoint input segments. In this case we assume that the triangle with circumcenter b_1 was removed by a (QUALITY′) operation because of a small angle opposite a_0, b_0. In particular this means that we assume the angle subtended by the input segments containing a_0, b_0 is at least $\pi/3$ in this case.

When talking about such sequences, for $i = 1, 2, \ldots, l - 1$, let $\tilde{r}_i$ be the circumradius of the triangle associated with b_i. Note that $\tilde{r}_i = |b_i - b_{i-1}| = |b_i - a_{i-1}|$, and that $|a_i - b_i| \geq \tilde{r}_i$. We let $\tilde{r}_0 = |b_0 - a_0|$, *i.e.*, the length of the first short edge.

Note that for a circumcenter sequence, $\{b_i\}_{i=0}^{l-1}$, the points $b_1, b_2, \ldots, b_{l-2}$ are circumcenters which have been committed, b_{l-1} is a circumcenter, though it may be rejected, and b_0 may be any type of point. If b is a triangle circumcenter, there is always a circumcenter sequence ending with b, although it may be a trivial sequence of two elements. Any circumcenter sequence whose first element, b_0, is a triangle circumcenter may be extended to a maximal sequence whose first element is either a segment midpoint or an input point.

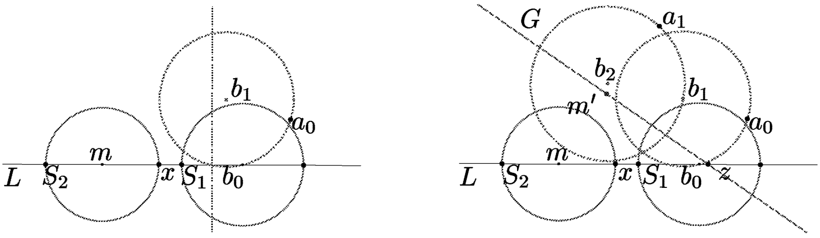
The following geometric lemma is the key result which allows us to make the $\arcsin 2^{-7/6}$ output guarantee. It essentially states that only circumcenter sequences longer than a certain length can "turn" around a 180° feature.

Lemma 4.

Let S_1, S_2 be segments with disjoint interiors on a common line, L. Assume that $|S_2| < |S_1|$, i.e., *S_2 is shorter than S_1. Let b_0 be the midpoint of S_1, and let $\{b_i\}_{i=0}^{l-1}$ be a circumcenter sequence such that b_{l-1} is inside the diametral circle of S_2. Then $l \geq 4$.*

Note that in this lemma can be proven in slightly more generality, which we do not need. In particular the order in which a_0, b_0 were committed is irrelevant to this result.

Proof. The argument is sketched in Figure 5. Point b_1 is the circumcenter of a triangle whose circumcircle does not contain the point x, which is the endpoint of S_1 closer to S_2. However, this circumcircle has b_0 on it, so b_1 must be in the closed halfspace defined by the bisector of x and b_0 and which does not contain x, as shown in Figure 5(a). Thus b_1 cannot be in the diametral circle of S_2, which is in the open halfspace on the other side of this bisector.



(a) b_1 does not encroach S_2.

(b) b_2 does not encroach S_2.

Fig. 5. The head of a circumcenter sequence is shown; the point b_1 must be to the right of the bisector of b_0 and x, and so it cannot encroach S_2, which is on the other side of this bisector, as shown in (a). In (b) the bisector of b_1 and the point x is shown. Since b_2 cannot be closer to x than to b_1, and since the diametral circle of S_2 is on the opposite side the bisector, b_2 cannot encroach S_2. In this case, a_0 is shown to be outside the diametral circle of S_1. This is not a necessary hypothesis for this lemma.

Let G be the bisector of the points b_1 and x. The point b_2 is the center of a circle which does not contain x, but has b_1 on its boundary, since b_1 is one of the vertices of the triangle which b_2 is added to remove. Thus b_2 must be either on the line G, or in the open halfspace defined by G that is closer to the point b_1. In Figure 5(b), this is the halfspace to the upper right of G.

It then suffices to show that the closure of the diametral ball of S_2 is contained in the other open halfspace defined by G, and thus b_2 cannot encroach S_2.

Let z be the intersection of L and G; take m to be the midpoint of S_2, and m' is its projection onto G. Let x' be the projection of x onto G. Let y be the projection of b_1 onto L. See Figure 6. The point x is clearly between m and z, otherwise x would be in the halfspace closer to b_1 than to x, a contradiction. Thus $|m - z| = |m - x| + |x - z|$.

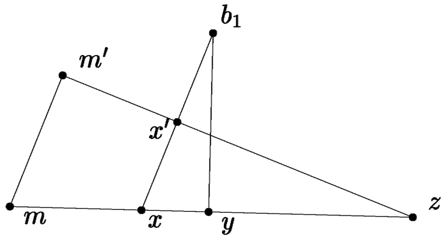


Fig. 6. The geometric heart of the argument is shown, with three similar triangles, $\Delta mm'z, \Delta xx'z, \Delta xyb_1$. Note that y may actually be to the right of z; this does not affect our argument.

By similarity of the three triangles of Figure 6,

$$\frac{|m - m'|}{|m - z|} = \frac{|x - x'|}{|x - z|} = \frac{|x - y|}{|x - b_1|}.$$

Let $r = |S_2|/2 < |S_1|/2$, by assumption. Since S_1, S_2 have disjoint interiors, $|m - x| \geq r$. Then $|m - z| \geq r + |x - z|$, so

$$\begin{aligned} |m - m'| &= \frac{|x - x'|\,|m - z|}{|x - z|}, \\ &\geq \frac{|x - x'|\,(r + |x - z|)}{|x - z|}, \\ &\geq \frac{|x - x'|}{|x - z|} r + |x - x'| \\ &= \frac{|x - y|}{|x - b_1|} r + |x - x'|. \end{aligned}$$

As noted above, b_1 is to the right of the bisector of x and b_0, so $|x - y| \geq |x - b_0|/2 = |S_1|/4 > r/2$. Note also that $|x - b_1| = 2|x - x'|$. Then

$$|m - m'| > \frac{r^2}{4|x - x'|} + |x - x'|.$$

The right hand side is minimized when $|x - x'| = r/2$, where the right hand side has value r.

Thus $|m - m'| > r$, and the distance from m to G, which is $|m - m'|$, is greater than the radius of the diametral circle of S_2. Then the closed diametral circle of S_2 is contained in the open halfspace opposite b_1, as desired. □

This lemma allows us to prove a better output angle for the Delaunay Refinement Algorithm. Previous proofs required $2\sin\tilde{\kappa} \leq \frac{1}{\sqrt{2}}$; by the lemma, the following proof only requires that $(2\sin\tilde{\kappa})^3 \leq \frac{1}{\sqrt{2}}$. A better output angle could be guaranteed if the lemma could be improved; this would have to be via some alteration of the algorithm, as the example of Figure 7 shows the lemma cannot be extended in the naïve setting. We return to this matter later.

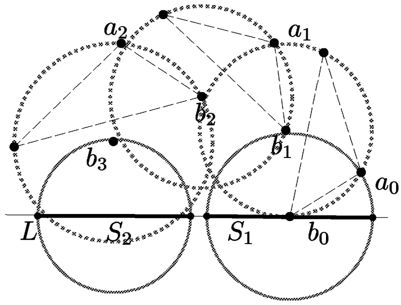


Fig. 7. A circumcenter sequence, $\{b_i\}_{i=0}^{3}$, is displayed, which shows that Lemma 4 cannot be extended. The segments S_1, S_2 are shown, with their diametral circles. The points b_1, b_2, b_3 are circumcenters of triangles (shown) with an angle smaller than $\pi/6$. The point b_3 encroaches S_2.

Since $\tilde{\kappa} < \pi/6$, we can establish a geometric series which gives the following lemma and its corollary. The corollary describes how a segment midpoint which is not caused by a midpoint encroaching the segment is caused by some other midpoint or input point.

Lemma 5.
Suppose $\{b_i\}_{i=0}^{l-1}$ is a circumcenter sequence. For $i > 0$, let $\tilde{r}_i$ be the circumradius associated with b_i. Then for $i = 1, 2, \ldots, l-1$,

- *$\tilde{r}_{i-1} < 2\tilde{r}_i \sin\tilde{\kappa}$ and therefore $\tilde{r}_i < (2\sin\tilde{\kappa})^{l-1-i}\tilde{r}_{l-1}$, and*
- *$|b_{l-1} - b_i| < \frac{\tilde{r}_{l-1}}{1-2\sin\tilde{\kappa}}$, and $|b_{l-1} - a_i| < \frac{\tilde{r}_{l-1}}{1-2\sin\tilde{\kappa}}$.*

Proof. By definition, b_i is the circumcenter of a triangle of radius $\tilde{r}_i$, which has a short edge no shorter than $\tilde{r}_{i-1}$ opposite an angle less than $\tilde{\kappa}$. By the sine rule, then $2\tilde{r}_i \sin\tilde{\kappa} > \tilde{r}_{i-1}$.

Using this repeatedly gives $\tilde{r}_i < (2\sin\tilde{\kappa})^{l-1-i}\tilde{r}_{l-1}$. Since $2\sin\tilde{\kappa} < 1$, we may bound the distance from b_i to b_{l-1} by the geometric series, as follows:

$$\begin{aligned}
|b_{l-1} - b_i| &\leq |b_{l-1} - b_{l-2}| + |b_{l-2} - b_{l-3}| + \ldots \\
&\quad + |b_{i+1} - b_i|, \\
&\leq \tilde{r}_{l-1} + \tilde{r}_{l-2} + \ldots + \tilde{r}_{i+1}, \\
&< \tilde{r}_{l-1} + (2\sin\tilde{\kappa})\tilde{r}_{l-1} + \ldots
\end{aligned}$$

$$+(2\sin\hat{\kappa})^{l-i-2}\tilde{r}_{l-1},$$
$$< \frac{1}{1-2\sin\hat{\kappa}}\tilde{r}_{l-1}.$$

The bound for $|b_{l-1} - a_i|$ follows since $|b_{i+1} - a_i| = |b_{i+1} - b_i| = \tilde{r}_{i+1}$, and the above analysis suffices. □

Corollary 1. *Suppose that segment s_p with midpoint p and radius r_p was split, but the segment was not encroached by a committed point. Then there is some maximal circumcenter sequence $\{b_i\}_{i=0}^{l-1}$ such that b_{l-1} "yielded" to p, causing it to be committed. Moreover, $\tilde{r}_i < (2\sin\hat{\kappa})^{l-1-i}\sqrt{2}r_p$, $|p - b_i| \leq \eta r_p$, and $|p - a_i| \leq \eta r_p$, for $i = 0, 1, \ldots, l-1$, with $\eta = 1 + \frac{\sqrt{2}}{1-2\sin\hat{\kappa}}$.*

Proof. Since b_{l-1} was the center of an empty circumcircle, but encroached s_p, then $\tilde{r}_{l-1} \leq \sqrt{2}r_p$. Using the lemma gives the desired bound on $\tilde{r}_i$. By the lemma, and since $\hat{\kappa} < \pi/6$, $\tilde{r}_i \leq \tilde{r}_{l-1}$. Then

$$|p - b_i| \leq |p - b_{l-1}| + |b_{l-1} - b_i| \leq r_p + \frac{\tilde{r}_{l-1}}{1-2\sin\hat{\kappa}}$$
$$\leq \left(1 + \frac{\sqrt{2}}{1-2\sin\hat{\kappa}}\right) r_p = \eta r_p.$$

The bound on $|p - a_i|$ follows, *mutatis mutandis*, as above. □

7. Proving Termination

We prove termination not by showing that output mesh edges are well-graded, rather by showing that the algorithm can create no mesh edge smaller than dictated by the minimum local feature size of the input. Towards this end we define

$$\mathrm{lfs}_{min} = \min\{\mathrm{lfs}(x) \mid x \in \Omega\}.$$

Theorem 4 (Radius Bounds). *Suppose that the input to the Adaptive Delaunay Refinement Algorithm conforms to Assumption 3. Suppose that $\hat{\kappa} \leq \arcsin 2^{-7/6}$. Then there is a constant, μ, depending on θ^* and $\hat{\kappa}$ such that if p is the midpoint of a segment, s, of radius r that is committed by the algorithm, then $\mathrm{lfs}_{min} \leq \mu r$.*

Proof. We consider why the segment was split. If there was an input point or a point on a disjoint input sequence that encroached s, then clearly $\mathrm{lfs}(p) \leq r$, so it suffices to take $\mu \geq 1$.

Suppose a midpoint q on a nondisjoint input sequence encroached s. Using this result inductively we know that $\mathrm{lfs}_{min} \leq \mu r_q$, where r_q is the radius associated with q. By Lemma 3, $r_q \leq r$, which suffices.

Suppose that s was not encroached by an input point or midpoint, rather it was split when a circumcenter "yielded" to the segment split. Consider a maximal circumcenter sequence, $\{b_i\}_{i=0}^{l-1}$ ending in the circumcenter b_{l-1} which yielded to the split of s. By maximality, b_0 is not a circumcenter. Consider the identity of b_0.

If b_0 is an input point or a midpoint on an input feature disjoint from the segment containing s, then $\text{lfs}(p) \leq |p - b_0| \leq \eta r$, by Corollary 1. Thus it suffices to take $\eta \leq \mu$.

The only remaining possibility is that b_0 is a midpoint on an input feature nondisjoint from the one containing s. Let r_b be the radius associated with b_0. This radius may be larger or smaller than $\tilde{r}_0 = |b_0 - a_0|$. We consider the possibilities:

- Suppose $r_b \leq \tilde{r}_0$. Using this result inductively we have $\text{lfs}_{min} \leq \mu r_b$. By Corollary 1, $\tilde{r}_0 \leq (2 \sin \hat{\kappa})^{l-1} \sqrt{2} r$. If b_0 is a midpoint on the same input segment as p or on a distinct input segment subtending an angle other than π, then by Assumption 3, $\log_2 \frac{r}{r_b}$ is an integer. But since $r_b \leq \sqrt{2} r$, it must be a nonnegative one, thus $r \geq r_b$, so $\text{lfs}_{min} \leq \mu r$. The only alternative is b_0 is a midpoint on a distinct input segment subtending angle π with the one containing p. Then either $r_b \leq r$, in which case immediately $\text{lfs}_{min} \leq \mu r$, or $r < r_b$, in which case by Lemma 4, $l \geq 4$, so $r_b \leq (2 \sin \hat{\kappa})^3 \sqrt{2} r$. This yields a contradiction when $\hat{\kappa} \leq \arcsin 2^{-7/6}$, as assumed.
- Suppose $r_b > \tilde{r}_0$. This means that a_0 encroached the diametral circle of the subsegment associated with b_0, and thus, since b_0 was committed after a_0, a_0 is not a circumcenter.

  If a_0 is an input point or on an input segment disjoint from the one containing b_0, then $\text{lfs}_{min} \leq \tilde{r}_0$, so it suffices to take $\mu \geq 1$.

  The alternative is that a_0 is a midpoint on an input segment adjoining the one containing b_0. By the definition of the (QUALITY′) operation and circumcenter sequences, it must be the case that θ, the angle between the two input segments is as least $\pi/3$. Using Lemma 3, we know that $\tilde{r}_0 = |a_0 - b_0| \geq r_a$, where r_a is the radius associated with a.

  If the input segment containing a_0 is disjoint from the one containing p, then using Corollary 1 again it suffices to take $\eta \leq \mu$.

  Otherwise arguments as above show that $r \geq r_a$, and using this result inductively suffices.

In all it suffices to take $\mu = \eta = 1 + \frac{\sqrt{2}}{1 - 2 \sin \hat{\kappa}}$. □

The following corollary gives termination:

Corollary 2. *Suppose the Adaptive Delaunay Refinement Algorithm considers committing point p. Let q be the closest point that has already been committed. Then* $\text{lfs}_{min} \leq \frac{\mu}{2 \sin \frac{\theta *}{2}} |p - q|$.

Proof. Consider the identity of p.

- Suppose p is a midpoint, and let r be the associated radius. If $r \leq |p - q|$, then the theorem gives $\text{lfs}_{min} \leq \mu |p - q|$. If, however, $r > |p - q|$, then q encroaches the subsegment of p, so it cannot be a circumcenter (which would have yielded). If q is an input point or on a disjoint input feature then $\mathrm{lfs}_{min} \leq |p-q|$, which suffices. Otherwise q is a midpoint on a nondisjoint input segment. Then, using, Lemma 3, $|p-q| \geq 2r_q \sin \frac{\theta^*}{2}$, where r_q is the radius associated with q. Using the theorem on q, we have $\mathrm{lfs}_{min} \leq \mu r_q$; which gives the desired result.

- Suppose p is a circumcenter with associated radius r. Then $r = |p-q|$, since the triangle is Delaunay. Then p can be considered the last circumcenter in a circumcenter sequence, and by Lemma 5 $r > \tilde{r}_0$. Then using this corollary inductively on the point b_0, the first point of the circumcenter sequence, gives the desired result. □

That this implies termination is simply seen: all committed points are in the bounded set Ω, but when the algorithm attempts to commit a point, its nearest committed neighbor cannot be too close. Thus the algorithm must terminate.

Note that this proof entirely ignores the issue of grading. The skeptic might object that all the edges in the final mesh could have size $\Theta(\mathrm{lfs}_{min})$. However, the algorithm actually *does* exhibit good grading; the proof of this fact is fairly involved.[12]

It is of interest that that this uniform grading constant does not diverge as $\hat{\kappa}$ reaches its limit value of $\arcsin 2^{-7/6}$, but *does* diverge as $\hat{\kappa}$ approaches $\pi/6$. We note that the limitation $\hat{\kappa} < \arcsin 2^{-7/6}$ comes from the case of collinear subsegments connected by a circumcenter sequence; in this situation Lemma 4 gives a lower bound on the length of the circumcenter sequence. A greater lower bound would relax the restriction on $\hat{\kappa}$, but this is not theoretically possible without changing the algorithm, as shown by the counterexample of Figure 7.

This does illustrate, however, why the Adaptive Delaunay Refinement Algorithm *might* work for a given input with $\hat{\kappa}$ as large as 30°: constructing a counterexample such as Figure 7 where collinear subsegments are connected by a circumcenter sequence is difficult work. Moreover, such counterexamples require a few committed points noncollinear with the subsegments, points which have to be perfectly aligned to make the counterexample work. Thus it seems unlikely that one could construct a counterexample where setting $\hat{\kappa} = 30°$ could cause the algorithm to fall into an infinite loop; such a counterexample would likely have to exhibit a structure which is scaled and repeated by repeated action of circumcenter sequences between collinear subsegments.

8. Output Quality

Recall that the Adaptive Delaunay Refinement Algorithm ignores some angles smaller than the parameter $\hat{\kappa}$. We will show that small output angles are not too much smaller than a nearby small input angle. The following simple geometric claim gives the output quality guarantee; the idea is to use it with facts about midpoints, the definition of (QUALITY′), and the Delaunay property to get the bound on output angles. The omitted proof is simple trigonometry.[12] A more general version appears in Ref. [7], which suggests a means of improving the output angle guarantee.

Lemma 6. *Let x, s, q be three distinct noncollinear points. Let p be a point on the open line segment from x to s. Suppose that $|p-s| \leq |x-p| \leq |x-q|$. Let $\theta = \angle pxq$, and $\phi = \angle psq$. Then*

$$\phi \geq \arctan\left(\frac{\sin\theta}{2-\cos\theta}\right).$$

The following claim is a simple consequence of Thales' theorem, and the definition of the Delaunay Triangulation.

Claim 8.1 (Edge-Apex Rule). *Given a triangle Δpqr in the Delaunay Triangulation of a set of points, $\mathcal{P}$, with L the line through p, q, then $\angle prq \geq \angle pr'q$ for every $r' \in \mathcal{P}$ that is on the same side of L as p, with equality only holding in the case of degeneracy.*

We can now state the output guarantee.

Lemma 7. *Suppose the Adaptive Delaunay Refinement Algorithm terminates for a given input. Let Δpqr be a triangle in the output triangulation. Then either*

(i) The angle $\angle prq > \hat{\kappa}$, or

(ii) the points p and q are midpoints on adjoining input segments which meet at angle $\theta < \pi/3$ and

$$\angle prq \geq \arctan\left(\frac{\sin\theta}{2-\cos\theta}\right).$$

The same alternative, mutatis mutandis, *holds for $\angle rqp$ and $\angle qpr$. Consequently no angle in the output mesh is smaller than* $\min\left\{\hat{\kappa}, \arctan\left(\frac{\sin\theta^*}{2-\cos\theta^*}\right)\right\}$.

Proof. Supposing that $\angle prq \leq \hat{\kappa}$, by the definition of the Adaptive Delaunay Refinement Algorithm, it must be that p, q are midpoints on an adjoining input segment, meeting at an angle, θ, less than $\pi/3$. Let x be the input point common to these segments. Without loss of generality, assume that $|x-p| \leq |x-q|$. The midpoint p is the endpoint of two subsegments of this input segment; let the one farther from x be (p, s). By Claim 5.1, $|p-s| \leq |p-x|$. Then by Lemma 6, $\angle psq \geq \arctan\left(\frac{\sin\theta}{2-\cos\theta}\right)$. Letting L be the line through p, q, consider the location of r:

- Suppose r is on the same side of L as x. By Claim 8.1, $\angle prq \geq \angle pxq = \theta > \arctan\left(\frac{\sin\theta}{2-\cos\theta}\right)$.
- If r is on the same side of L as s, by Claim 8.1, $\angle prq \geq \angle psq \geq \arctan\left(\frac{\sin\theta}{2-\cos\theta}\right)$. □

We note briefly that $\arctan\left[(\sin\theta)/(2-\cos\theta)\right] = \theta + \mathcal{O}\left(\theta^2\right)$, which makes this lower bound much better than that of $\arcsin\left[\sin\left(\frac{\theta}{2}\right)/\sqrt{2}\right] = \frac{\theta}{2\sqrt{2}} + \mathcal{O}\left(\theta^2\right)$ achieved by Shewchuk's Terminator.[5]

The following corollary gives an *upper* bound on output angles that depends on the output angle parameter, $\hat{\kappa}$, but *not* on the minimum output angle. Given $\hat{\kappa} = \arcsin 2^{-7/6} \approx 26.45°$, it guarantees no output angle is bigger than about $\pi - 2\arcsin \frac{\sqrt{3}-1}{2} \approx 137.1°$. The proof relies on the location of small output angles and uses the fact that diametral circles of subsegments are not encroached in the final mesh.

Corollary 3. *If Δpqr is a triangle in the output triangulation produced by the Adaptive Delaunay Refinement Algorithm, then*

$$\angle pqr \leq \max\left\{\pi - 2\hat{\kappa}, \pi - 2\arcsin \frac{\sqrt{3}-1}{2}\right\}.$$

Proof. Without loss of generality, assume that $\angle prq$ is the smallest angle of triangle Δpqr. We first prove that

$$\angle pqr \leq \min_{\kappa \leq \hat{\kappa}} \left[(\pi - 2\kappa) \vee \frac{2}{3}(\pi + \arcsin 2\sin\kappa - \kappa)\right].$$

Pick $\kappa \leq \hat{\kappa}$. Considering the two alternatives of Lemma 7: in the first case $\angle prq \geq \hat{\kappa} \geq \kappa$, and thus, since it is the smallest angle of the triangle, then $\angle pqr \leq \pi - 2\hat{\kappa} \leq \pi - 2\kappa$; So suppose the second case holds, *i.e.*, that p, q are midpoints on adjoining input segments which meet at angle $\theta < \pi/3$, and $\angle prq \geq \arctan\left(\frac{\sin\theta}{2-\cos\theta}\right)$. By trigonometry, if $\theta \geq \arcsin 2\sin\kappa - \kappa$, then $\arctan\left(\frac{\sin\theta}{2-\cos\theta}\right) \geq \kappa$, in which case, again, $\angle pqr \leq \pi - 2\kappa$. So assume otherwise. We will show that $\angle pqr \leq \frac{2}{3}(\pi + \theta)$.

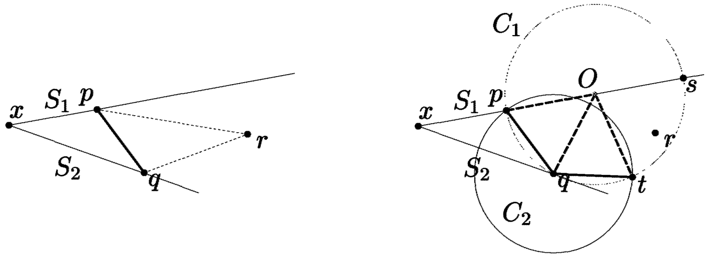
Let p be on input segment S_1, let q be on S_2; let x be the input point shared by S_1,S_2. Assume that r is not on the same side of the line segment (p, q) as x. In the case where r *is* on the same side of the segment as x, an argument similar to that which follows can show that $\angle pqr \leq \frac{2}{3}(\pi - \theta)$.

Because the output of the mesh respects the input segments, it must be the case that $\angle pqr$ is smaller than the angle subtended by (p, q) and S_2. That is, r is "between" S_1 and S_2, as shown in Figure 8(a). This is an external angle of triangle Δxpq at q, thus has magnitude $\theta + \angle xpq$. Then if $\angle xpq \leq \pi/2$, we can bound

$$\angle pqr \leq \theta + \angle xpq \leq \theta + \pi/2 = \frac{2}{3}\theta + \frac{1}{3}\theta + \pi/2 \leq \frac{2}{3}\theta + \frac{\pi}{9} + \pi/2 = \frac{2}{3}\theta + \frac{11}{18}\pi,$$

because $\theta < \pi/3$, which suffices. So assume $\angle xpq$ is obtuse.

Let s be the point on the line containing S_1 such that $\angle pqs = \pi/2$; because we have assumed $\angle xpq$ is obtuse, there is such a point and it is on the same side of (p, q) as r. Let C_1 be the circumcircle of p, q, s; it has center O on the line through S_1. Because line segments in the output are not encroached it must be the case that there is a vertex of the mesh on the line segment (p, s), since $\angle pqs = \pi/2$. It could be the case that this point is s itself. By Claim 8.1 it must be the case that r is inside or on C_1, as otherwise, by Thales' theorem, Δpqr would not have the Delaunay property.



(a) Input

(b) With Circles

Fig. 8. The proof of Corollary 3 is shown, for the case where r is opposite (p, q) from x. The point r must be between S_1,S_2, so $\angle pqr$ is smaller than the angle subtended by (p, q) and S_2. The point r must be inside the circle C_1, as otherwise some point on S_1 subtends a larger angle to (p, q) than r does, violating the Delaunay property of the output. By assumption, (p, q) is the shortest edge of the triangle Δpqr so r is outside the circle C_2 . Thus $\angle pqr \leq \angle pqt$. These two bounds together give $\angle pqr \leq \frac{2}{3}(\pi + \theta)$.

Since $\angle prq$ is the smallest angle of the triangle, then (p, q) is the shortest edge of the triangle. Then if C_2 is the circle centered at q of radius $|p - q|$, it must be the case that r is not inside C_2. The point p is a point of intersection of C_1, C_2; let t be the other. See Figure 8(b).

Then $\angle pqr \leq \angle pqt$. Looking at the two similar isosceles triangles of Figure 8(b), it is clear that $\angle pqt$ is twice the external angle of Δxpq at p, that is $\angle pqt = 2(\pi - \angle xpq)$. We have already seen that $\angle pqr \leq \theta + \angle xpq$, thus

$$\angle pqr \leq \min \{2(\pi - \angle xpq), \theta + \angle xpq\}.$$

We have assumed that $\pi/2 \leq \angle xpq \leq \pi - \theta$. Over this range the terms cross at $\angle xpq = \frac{2\pi - \theta}{3}$, and thus $\angle pqr \leq \frac{2}{3}(\pi + \theta)$, as desired.

Now we note that $\pi - 2\kappa$ is decreasing with increasing κ, while $\frac{2}{3}(\pi + \arcsin 2 \sin \kappa - \kappa)$ is increasing. A calculation shows that they cross when $\kappa = \arcsin \frac{\sqrt{3}-1}{2}$. Thus if $\hat{\kappa} \leq \arcsin \frac{\sqrt{3}-1}{2} \approx 21.47°$, then $\angle pqr$ is smaller than $\pi - 2\hat{\kappa}$. If $\hat{\kappa} \geq \arcsin \frac{\sqrt{3}-1}{2}$, then $\angle pqr$ is no larger than $\pi - 2 \arcsin \frac{\sqrt{3}-1}{2} \approx 137.1°$. □

9. Input Grooming

Our analysis so far has required that input meet Assumption 3. We will consider how to make input which satisfies Assumption 1 conform to the extra assumption. We view this transformation as a preprocessor which puts the input into the appropriate form for the algorithm. This preprocessor breaks an input segment into pieces not too much shorter than the original.

Definition 3. A *γ-Bounded Reduction Augmenter* is a procedure that takes an input $(\mathcal{P}, \mathcal{S})$, and produces an output $(\mathcal{P}', \mathcal{S}')$ such that

(i) $\mathcal{P} \subseteq \mathcal{P}'$,
(ii) every segment of $\mathcal{S}$ is the union of segments in $\mathcal{S}'$,
(iii) every point of $\mathcal{P}' \setminus \mathcal{P}$ and every segment of $\mathcal{S}'$ is on a segment of $\mathcal{S}$, and
(iv) if $S' \in \mathcal{S}'$ is a segment on segment $S \in \mathcal{S}$, then $|S| \leq \gamma\,|S'|$.

Note the definition requires that $\gamma \geq 2$, otherwise the augmenting procedure is idempotent, *i.e.*, leaves $(\mathcal{P}, \mathcal{S})$ unchanged. Also note that for a γ-Bounded Reduction Augmenter that $|\mathcal{P}' \setminus \mathcal{P}| \leq \lfloor \gamma - 1 \rfloor\,|\mathcal{S}|$.

The following theorem bounds the decrease in local feature size caused by a γ-Bounded Reduction Augmenter.

Theorem 5.

Let $\operatorname{lfs}(x)$ *be the local feature size on* $(\mathcal{P}, \mathcal{S})$, *and let* $\operatorname{lfs}'(x)$ *be the local feature size on* $(\mathcal{P}', \mathcal{S}')$, *where* $\mathcal{P} \subseteq \mathcal{P}'$, *every segment of* $\mathcal{S}$ *is the union of segments in* $\mathcal{S}'$, *and every point of* $\mathcal{P}' \setminus \mathcal{P}$ *and segment of* $\mathcal{S}'$ *is on a segment of* $\mathcal{S}$.

Furthermore suppose that coincident segments of $\mathcal{S}$ *meet at an angle no less than some* $\theta^* \leq \pi/2$, *and that there is a constant* $\gamma \geq 2$ *such that if* $S' \in \mathcal{S}'$ *is a segment on the segment* $S \in \mathcal{S}$, *that* $|S| \leq \gamma\,|S'|$.

Then there is a constant σ *such that* $\operatorname{lfs}'(x) \geq \sigma \operatorname{lfs}(x)$ *for every* $x \in \mathbb{R}^2$. *Additionally,* $\sigma = \frac{\sin\theta^*}{2\gamma-1}$ *suffices.*

Note that although we have been assuming that $\theta^* \leq \pi/3$, this theorem only requires that θ^* is no greater than $\pi/2$.

This theorem bounds the possible decrease of lfs_{min} when preprocessing the input to make it conform to Assumption 3. In light of Corollary 2, this is a bound on the decrease of minimum edge length in the mesh output by the Adaptive Delaunay Refinement Algorithm.

Proof. We will refer to those features of $(\mathcal{P}, \mathcal{S})$ as being "input."

Let $r = \operatorname{lfs}'(x)$, and let B be the open ball centered at x of radius r. By definition there are two disjoint features touching the closure of B. Since features of $(\mathcal{P}', \mathcal{S}')$ are on features of the input, we can assume there are two (not necessarily distinct) features of the input, say X, Y that intersect the closure of B. We consider the cases:

- If X, Y are disjoint features of the input, then by definition $\operatorname{lfs}(x) \leq r = \operatorname{lfs}'(x)$, since the closure of B intersects two features of the input.
- If $X = Y$, then the closure of B contains two points of $\mathcal{P}'$ placed on some segment $(a, b) \in \mathcal{S}$. If both these points are in $\mathcal{P}$, then by definition $\operatorname{lfs}(x) \leq \operatorname{lfs}'(x)$. So assume that there is one, call it p, that is in $\mathcal{P}'$. Our assumptions on the lengths of segments in $\mathcal{S}'$ give us

$$|a - p| \vee |b - p| \leq \frac{(\gamma - 1)\,|a - b|}{\gamma}.$$

By definition of the local feature size, $\operatorname{lfs}(x) \leq |x - a| \vee |x - b|$. Using the triangle inequality gives $\operatorname{lfs}(x) \leq r + (|p - a| \vee |p - b|) \leq r + \frac{(\gamma-1)|a-b|}{\gamma}$.

So

$$\frac{\mathrm{lfs}'(x)}{\mathrm{lfs}(x)} \geq \frac{\mathrm{lfs}'(x)}{\mathrm{lfs}'(x) + \frac{(\gamma-1)|a-b|}{\gamma}}.$$

The two terms involved in the right hand side are both positive, but only one of them depends on x. It is easy to see that the right hand side is minimized when $\mathrm{lfs}'(x)$ is minimized. But the new local feature size, $\mathrm{lfs}'(x)$ must be at least half the distance from p to the other point, *i.e.*, at least half the length of a segment on (a, b), so $\mathrm{lfs}'(x) \geq \frac{|a-b|}{2\gamma}$. Thus

$$\begin{aligned}\frac{\mathrm{lfs}'(x)}{\mathrm{lfs}(x)} &\geq \frac{\frac{|a-b|}{2\gamma}}{\frac{|a-b|}{2\gamma} + \frac{(\gamma-1)|a-b|}{\gamma}} \\ &\geq \frac{1}{1+2(\gamma-1)},\end{aligned}$$

which suffices.

- If X, Y are non-disjoint input features, then we may assume they are segments, as the case where one is an endpoint of the other is treated in the previous case. Let the two segments be $(a, b), (a, c)$. Without loss of generality assume there is a feature of $(\mathcal{P}', \mathcal{S}')$, call it Z, that is on (a, b) and intersects the closure of B, as does the segment (a, c). Furthermore we may assume that Z is disjoint from every feature of $(\mathcal{P}', \mathcal{S}')$ which is on (a, c). Then there is a point $p \in \mathcal{P}' \setminus \mathcal{P}$ which is on (a, b) which is no farther from a than Z. See Figure 9.

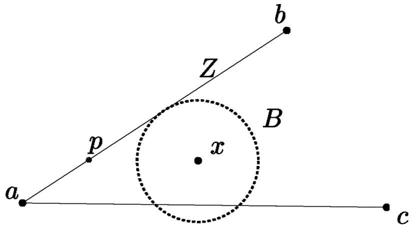


Fig. 9. The "segment-segment" case for the proof of Theorem 5 is shown. The segments (a, b),(a, c) are in the input, while p is an augmenting point.

This then gives us the lower bound: $\mathrm{lfs}'(x) \geq \frac{|a-p| \sin(\theta \wedge \pi/2)}{2}$, where θ is the angle subtended by the segments $(a, b), (a, c)$. This lower bound holds because the distance from Z to (a, c) is at least the distance from a to Z times $\sin(\theta \wedge \pi/2)$. Using the bounds from the previous case, this gives $\mathrm{lfs}'(x) \geq \frac{|a-b| \sin(\theta \wedge \pi/2)}{2\gamma}$.

The local feature size (with respect to the input) of x is no greater than the distance from x to b, since b is outside the closure of B, which intersects an input feature disjoint from b, *i.e.*, the segment (a, c). Using the triangle inequality to bound the distance from x to b gives an upper bound: $\mathrm{lfs}(x) \leq r + |p - b|$. This holds because Z intersects the closure of B, but is no farther from b than p is. Thus

$$\frac{\mathrm{lfs}'(x)}{\mathrm{lfs}(x)} \geq \frac{\mathrm{lfs}'(x)}{\mathrm{lfs}'(x) + |p - b|}.$$

Again, the right hand side is minimized when $\mathrm{lfs}'(x)$ is minimized. Using the above obtained lower bound on $\mathrm{lfs}'(x)$ gives

$$\begin{aligned} \frac{\mathrm{lfs}'(x)}{\mathrm{lfs}(x)} &\geq \frac{\frac{|a-b| \sin(\theta \wedge \pi/2)}{2\gamma}}{\frac{|a-b| \sin(\theta \wedge \pi/2)}{2\gamma} + \frac{(\gamma-1)|a-b|}{\gamma}} \\ &\geq \frac{\sin(\theta \wedge \pi/2)}{\sin(\theta \wedge \pi/2) + 2(\gamma - 1)} \\ &\geq \frac{\sin(\theta \wedge \pi/2)}{2\gamma - 1} \geq \frac{\sin \theta^*}{2\gamma - 1}. \end{aligned}$$ □

In Algorithm 1 we present a 5-Bounded Reduction Augmenter which makes arbitrary input conform to Assumption 3. This preprocessor adds augmenting points around each vertex of degree at least two.

```
Algorithm 1:
Input: Input points and segments.
Output: An augmented set of points and segments.
BoundedReductionAugment($\mathcal{P}, \mathcal{S}$)
    foreach point $x \in \mathcal{P}$ which is the endpoint of at least two
    segments in $\mathcal{S}$
        To each segment $(x, y)$ in $\mathcal{S}$, add an augmenting point $p$,
        such that $0.2\,|x - y| \leq |x - p| \leq 0.4\,|x - y|$, and such
        that each new segment $(x, p)$ has the same length mod-
        ulo a power of two.
```

10. Adaptive Midpoint Splitting

Experimental observation has shown that using an algorithm like Algorithm 1 to preprocess input results in a mesh with a large number of Steiner Points. For example, consider Figure 10, the input to which is the outline of Lake Superior. The use of Algorithm 1 as a preprocessor results in a final mesh with approximately $10\,|\mathcal{P}|$ Steiner Points for this example, whereas using Ruppert's strategy of splitting on concentric circular shells results in approximately $2\,|\mathcal{P}|$ Steiner Points. These results are typical and illustrate that use of a preprocessor is not a practical idea, though it does provide a theoretical result. We here consider Ruppert's heuristic, which will remove the input restriction Assumption 3 while still yielding small point sets in practice.

Ruppert's strategy of splitting on concentric circular shells proceeds as follows: The first time an input segment is split, it is split by its midpoint, creating two subsegments each associated with one input point. When one of these subsegments is split, it is split by a point p closest to the midpoint of the subsegment such that $|p - x|$ is a power of two (in some global unit), where x is the input point associated with the subsegment. All further subsegment splits are committed at midpoints.[3]

Note that if this strategy was applied to split every input segment into three subsegments, we could describe it as a 6-Bounded Reduction Augmenter.

We will refer to these first three points on any segment as "off-center" points, even though they could be at the midpoint of the involved subsegment. It is simple to show that lfs_{min} is no greater than three times the length of the shortest subsegment created by an off-center split under this strategy. This follows since lfs_{min} is no greater than half the length of any input segment, and the fact that the off-center split must occur in the middle third of the subsegment.

Then Theorem 4 can be reproven for the Adaptive Delaunay Refinement Algorithm with concentric shell splitting for arbitrary input satisfying Assumption 1. The basic strategy is that if any of the midpoints involved in the proof are actually off-center points, they can be shown to be not far away by Corollary 1, and then the Lipschitz property of local feature size suffices; in the "endgame" none of the involved midpoints are off-center, and the input locally conforms to Assumption 3, so the arguments used previously may be reused.

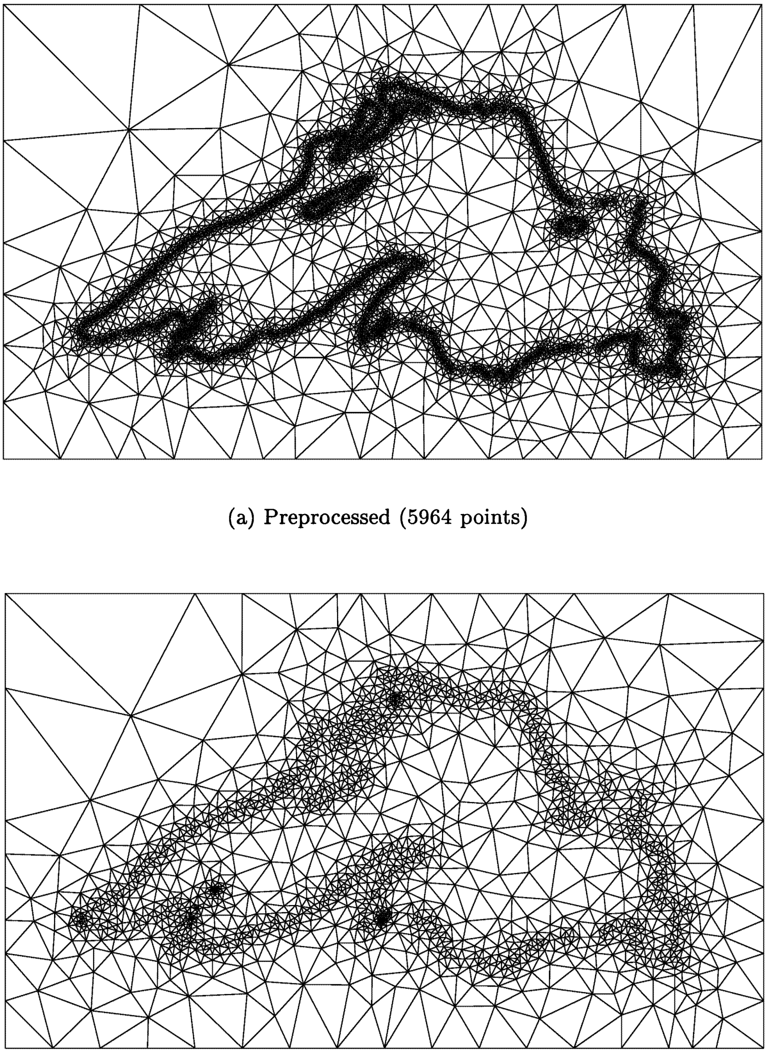
For the analysis to be valid, it is necessary that the algorithm treat off-center points as input points, not as midpoints. This makes a difference because the adaptive variant of the Delaunay Refinement Algorithm regards triangles differently if the shortest edge has midpoints as endpoints.

In light of the discussion in Subsection 3.1, we make the following

Claim 10.1. *Suppose an input conforming to Assumption 1 if given to Ruppert's Algorithm with concentric shell splitting. Then if* $\kappa < 26.45° \wedge \arctan\left[(\sin\theta^*)/(2 - \cos\theta^*)\right]$, *the algorithm will terminate with no angle smaller than* κ.

11. Results

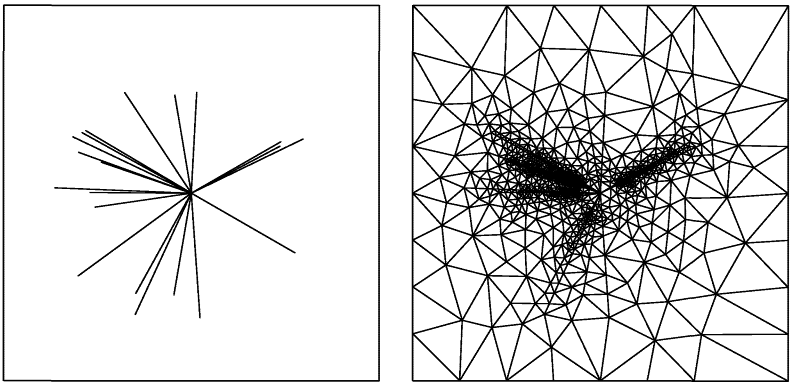
The Adaptive Delaunay Refinement Algorithm with splitting on concentric shells was implemented. In Figure 11, the code was tested on an input consisting of 20 randomly chosen "spokes" of length varying from 1/4 to 3/8 inside a unit box. The minimum input angle is approximately 0.92°. The output is a mesh on 1042 vertices, of which 1017 are Steiner Points, of these 559 are Steiner Points on input segments. This latter number is quite a bit larger than the figure $3|\mathbb{S}| = 72$ which is the number of "off-center" splits added by Ruppert's heuristic before all segments are put in the form of Assumption 3.



(a) Preprocessed (5964 points)

(b) Adaptive (1750 points)

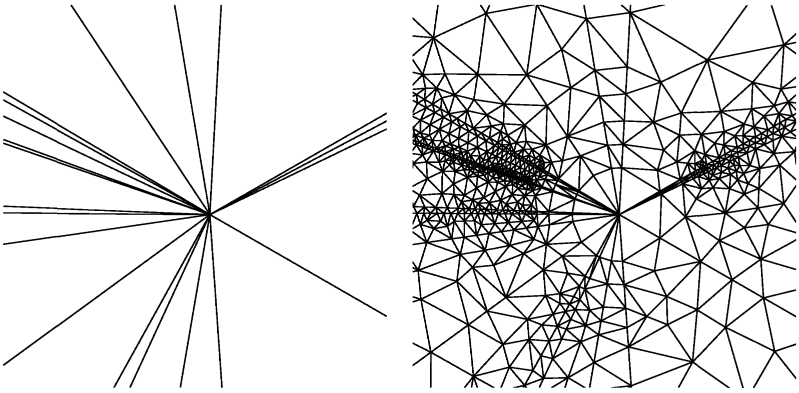
Fig. 10. The outline of Lake Superior, consisting of 522 points and 522 line segments, was processed by the Adaptive Delaunay Refinement Algorithm. In (a), the input was preprocessed with Algorithm 1, while in (b), Ruppert's heuristic of splitting on concentric circular shells was used instead. For the preprocessed case, 5442 Steiner Points are added (1044 during preprocessing), while for Ruppert's heuristic, only 1228 Steiner Points are added in total. In both cases $\hat{\kappa} \approx \arcsin 2^{-7/6}$ was used. The minimum input angle is $\theta^* \approx 15.02°$, while the minimum output angle for the meshes is around $14.07°$.



(a) Input

(b) Output

Fig. 11. An input consisting of 20 spokes has been processed by the Adaptive Delaunay Refinement Algorithm. The input is shown in (a), the output in (b). The input has minimum angle around 0.92°. In the output 1017 Steiner Points have been added, around half of which are on input segments.



(a) Input (detail)

(b) Output (detail)

Fig. 12. A detail of the input and output of Figure 11 is shown. The segments are all cut at the same distance from the "axle." This distance is approximately one eighth the length of the shortest input spoke. The mesh appears fairly regular near the axle.

A detail of the "axle" of the spoke, Figure 12, shows that all spokes are split at the same distance from the axle. This distance is approximately one eighth the length of the shortest spoke. Note this is quite a bit better than the theoretical

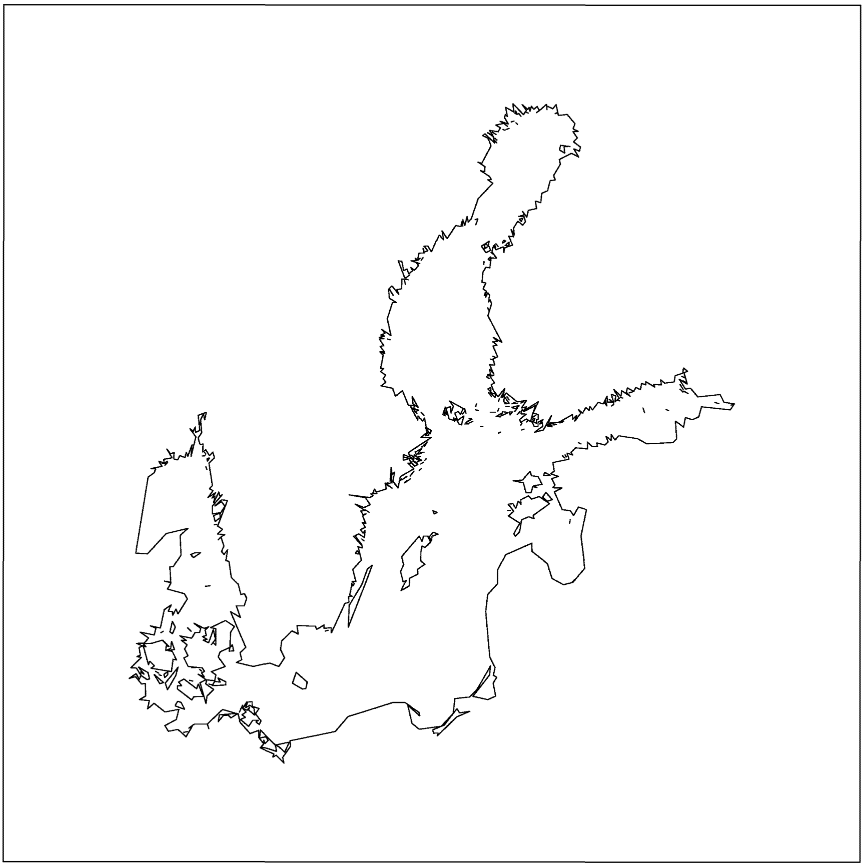
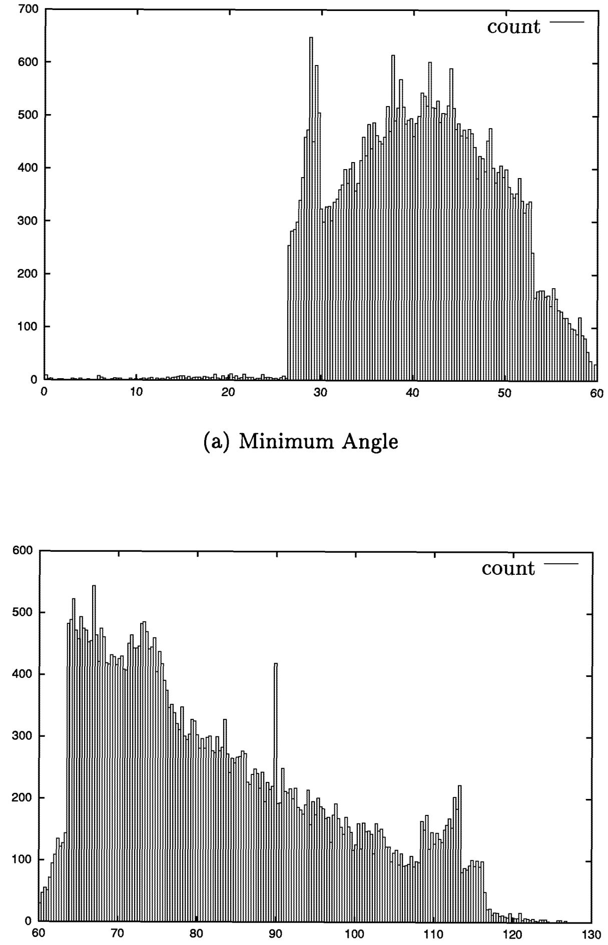


Fig. 13. The Baltic Sea input data. The input consists of 1401 points and 1301 line segments. There are a number of small angles and small segments present. The minimum angle, θ^* is approximately 0.052°.

guarantee that the shortest input spoke is no longer than 34000 times the length of these innermost subsegments, for this input (the bound depends on θ^*).[12] It is fortunate that the algorithm performs much better in practice than the dismal analysis can bound.

The code was also tested on the Baltic Sea, shown in Figure 13, with $\hat{\kappa} \approx \arcsin 2^{-7/6}$. The input has a number of small angles, the smallest being around 0.052°.

The output is shown in Figure 15, and is a mesh on 21775 vertices. The minimum and maximum angle histograms are shown in Figure 14. The minimum angle histogram shows that a small number of triangles have minimum angle less than 26.45°; these are all small input angles or "across" from small input angles, in accordance with Lemma 7. The largest angle in the output is about 126.9°, while the smallest angle is about 0.052°.



(a) Minimum Angle

(b) Maximum Angle

Fig. 14. The minimum- and maximum angle histograms are shown, respectively, in (a) and (b). In this figure triangles are counted, not angles, thus the total count is the number of triangles (in this case 43357), and not three times that number. In (a), those triangles with minimum angle smaller than $\hat{\kappa} \approx 26.45°$ are due to small input angles, in accordance with Lemma 7. The lack of large angles is guaranteed by Corollary 3.

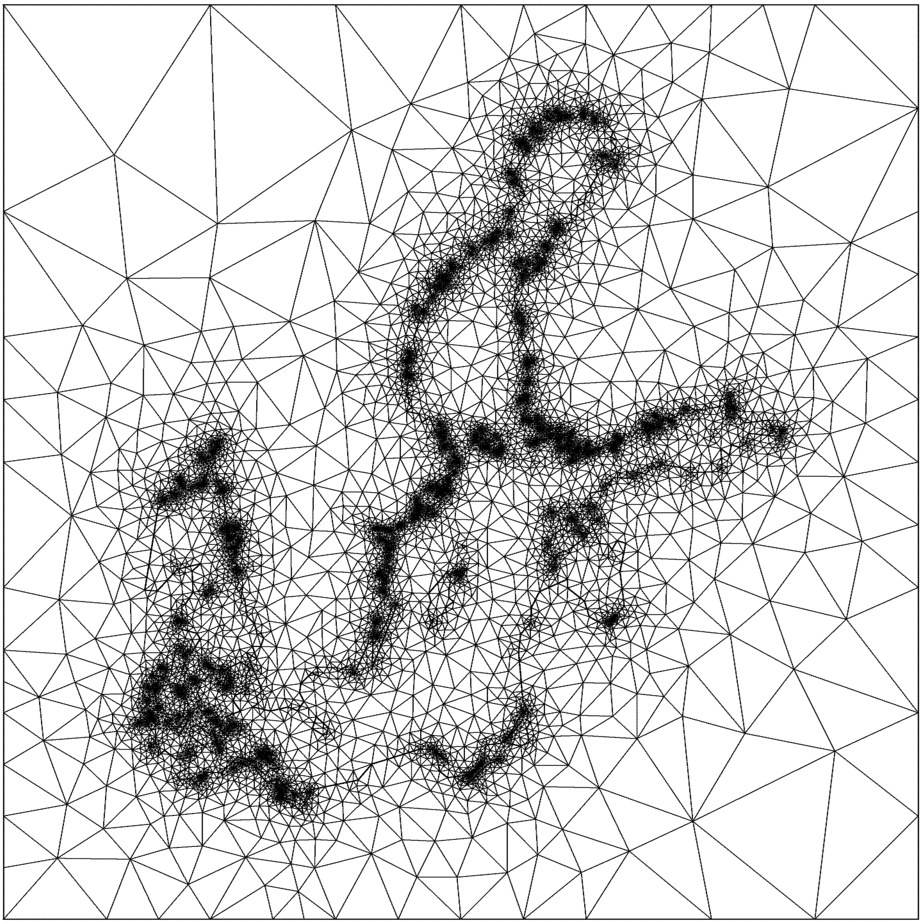


Fig. 15. The output mesh of the Baltic Sea input (Figure 13) with $\hat{\kappa} \approx \arcsin 2^{-7/6}$ is shown.

Acknowledgements

Supported in part by National Science Foundation Grants DMS–0208586 and ITR 0086093. This work was also supported by the NSF through the Center for Nonlinear Analysis.

References

1. P. G. Ciarlet, *The Finite Element Method for Elliptic Problems* (North–Holland, 1978).
2. I. Babuška and A. K. Aziz, On the angle condition in the finite element method, *SIAM J. Numer. Anal.* **13** (1976) (2) 214–226.
3. J. Ruppert, A Delaunay refinement algorithm for quality 2-dimensional mesh generation, *J. Algorith.* **18** (1995) (3) 548–585. ISSN 0196-6774, *Fourth Annual ACM-SIAM Symp. Discrete Algorithms* (*SODA*) (Austin, TX, 1993).

4. J. R. Shewchuk, *Delaunay Refinement Mesh Generation*, Ph.D. thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania (May 1997). Available as Technical Report CMU-CS-97-137.
5. J. R. Shewchuk, Delaunay refinement algorithms for triangular mesh generation, *Comput. Geom.* **22** (2002) (1-3) 21–74. ISSN 0925-7721, *16th ACM Symp. Computational Geometry* (Hong Kong, 2000).
6. C. Boivin and C. F. Ollivier-Gooch, Guaranteed-quality triangular mesh generation for domains with curved boundaries, *Int. J. Numer. Methods Eng.* **55** (2002) (10) 1185–1213.
7. S. E. Pav and N. J. Walkington, Corner lopping: A simple meshing algorithm suitable for domains with curved boundaries (2004), in preparation.
8. G. L. Miller, S. E. Pav and N. J. Walkington, An incremental Delaunay meshing algorithm, Technical Report 02-CNA-011, Center for Nonlinear Analysis, Carnegie Mellon University (2002). URL http://www.math.cmu.edu/cna
9. H. Si. Tetgen, A 3D Delaunay tetrahedral mesh generator. v.1.3 users manual (2004). URL http://www.wias-berlin.de/publications/technicalreports/9 URL http://tetgen.berlios.de/
10. S.-W. Cheng and S.-H. Poon, Graded conforming Delaunay tetrahedralization with bounded radius-edge ratio, in *Proc. Fourteenth Ann. ACM-SIAM Symp. Discrete Algorithms (Baltimore, MD, 2003)* (ACM, New York, 2003), pp. 295–304.
11. S. E. Pav and N. J. Walkington, Robust three dimensional Delaunay refinement, in *Proc. 13th Int. Meshing Roundtable* (Sandia National Laboratory, 2004), pp. 145–156.
12. S. E. Pav, *Delaunay Refinement Algorithms*, Ph.D. thesis, Department of Mathematics, Carnegie Mellon University, Pittsburgh, Pennsylvania (May 2003). URL http://www.andrew.cmu.edu/~spav/work
13. G. L. Miller, A time efficient Delaunay refinement algorithm, in *Proc. Fifteenth Annual ACM-SIAM Symp. Discrete Algorithms (SODA)* (2004), pp. 400–409.
14. E. P. Mücke, A robust implementation for three-dimensional Delaunay triangulations, *Int. J. Comput. Geom. Appl.* **8** (1998) (2) 255–276. ISSN 0218-1959.