The Farthest Point Delaunay Triangulation Minimizes Angles

David Eppstein
Department of Information and Computer Science
UC Irvine, CA 92717

November 20, 1990

Abstract

We show that the planar dual to the Euclidean farthest point Voronoi diagram for the set of vertices of a convex polygon has the lexicographic minimum possible sequence of triangle angles, sorted from sharpest to least sharp. As a consequence, the sharpest angle determined by three vertices of a convex polygon can be found in linear time.

1. Introduction

A celebrated result in computational geometry is that the Delaunay triangulation of a planar point set maximizes the minimum angle in any triangle [7]. More specifically, if the points are in general position (by which we mean no four points are cocircular), then the sequence of triangle angles, sorted from sharpest to least sharp, is lexicographically maximized over all such sequences constructed from triangulations of the points.

In this paper we study a similar result for the *farthest point Delaunay triangulation*; that is, the planar dual of the farthest point Voronoi diagram of a planar point set. Since the farthest point Voronoi diagram only has regions corresponding to the vertices of the convex hull of the original point set, we concentrate our attention on points that are the vertices of a convex polygon. We show that, for such point sets in general position, the sequence of triangle angles, sorted from sharpest to least sharp as before, is lexicographically minimized.

Instead of Delaunay triangulations, we may view our results in terms of three dimensional convex hulls. If we map each point (x, y) to $(x, y, x^2 + y^2)$, the plane is mapped to a paraboloid in space, and every circle in the plane is mapped to a plane cutting the paraboloid. This transformation has been used, e.g., in data structures for circular range queries [8]. It is a curious fact that both the nearest and farthest point Voronoi diagrams can be obtained as the planar projections of the upper and lower portions of the convex hull of the transformed point set [1, 2]. Our result provides another relation between the upper and lower hulls, in terms of the angles of their corresponding planar triangles.

As a consequence of our results, the sharpest angle determined by three vertices of a convex polygon can be determined in linear time, using the linear time convex hull algorithm of Aggarwal et al. [1]. The sharpest angle determined by three of an arbitrary set of n points corresponds to a line segment bounding one of the faces of the dual line arrangement, and so can be found in $O(n^2)$ time [3]. A faster solution would also speed up the detection of degeneracies in a configuration (i.e. three collinear points, or equivalently three concurrent lines). Detecting degeneracy faster than $O(n^2)$ is an outstanding open problem in the field of computational geometry. Perhaps alternate views of the problem, such as the minimum angle problem studied here, will lead to an improved solution, but we have been unable to come up with such a result.

There remains the question of minimizing the angle sequence for points not in general position. The problem arises when a face of the Delaunay triangulation has more than three vertices; this can happen when several points are cocircular. The sharpest angle can still be found in linear time in this case, but the lexicographic minimum angle triangulation is harder to find. The corresponding problem for maximizing the angle sequence was solved by Mount and Saalfeld [6], who showed that the optimal triangulation for cocircular points could be found in time $O(n \log n)$. We show that, for our problem, with input points on a circle but otherwise in general position, an $O(n)$ time algorithm exists. Finally, when two or more pairs of points have the same distance, the problem can be solved in polynomial time by dynamic programming. In contrast to the close relation between nearest point and farthest point Delaunay triangulations for the general position case, our methods for cocircular points are quite different from those of Mount and Saalfeld.

2. Sharpest Angles in Convex Polygons

We now describe how the sharpest angle in a point set may be found, if the points form the vertices of a convex polygon. This will lead to our characterization of the farthest point Delaunay triangulation in terms of its angle sequence. We assume that the order of the points around the polygon is known; otherwise all algorithms require an additional $O(n \log n)$ convex hull step. First we need two lemmas; the first is obvious and the second can be found in Euclid [5].

Lemma 1. *If angle xyz is the sharpest angle formed by the vertices of a convex polygon, then points x and z are adjacent vertices on the boundary of the polygon.* □

Lemma 2. *Given points x and z in the plane, and an angle θ, then the locus of points y such that angle xyz is equal to θ forms a circle, touching both x and z, and such that the center of the circle forms angle 2θ with x and z.* □

Theorem 1. *The sharpest angle of a set of n points in general position, which form the vertices of a convex polygon, can be found as an angle of a triangle in the farthest point Delaunay triangulation.*

Proof: Let θ be the minimum angle xyz in a convex polygon, and let C be the circle tangent to all three points. Then by lemma 2 there can be no point of the polygon outside C and on the same side of xz as y, or else that point would lead to a sharper angle. By lemma1 there can be no point in or out of C on the other side of xz. Therefore C contains the entire polygon, and touches three points of it. The center of C is equidistant from each of x, y, and z, and no further from any of the other input points; therefore it is contained in the intersection of the farthest point Voronoi diagram regions corresponding to x, y, and z. Dually, x, y, and z are vertices of a common face in the farthest point Delaunay triangulation. Since the points were assumed to be in general position, this face is a triangle. □

We can now use this result to show that the farthest point Delaunay triangulation indeed minimizes the angle sequence.

Theorem 2. *Given a set of points in convex but otherwise general position, the triangulation lexicographically minimizing the sequence of angles,* *sorted from sharpest to least sharp, of its triangles, is the farthest point Delaunay triangulation.*

Proof: By the assumption of general position, all angles differ. Let xyz be the sharpest angle determined by the points. Then xyz must be an angle of a triangle in the minimum triangulation; since x and z are adjacent, the only possible such triangle is xyz. This triangle is indeed included in the farthest point Delaunay triangulation. Let A be the set of points on the side of xy, and B be the points on the side of yz. Then clearly a minimum angle triangulation of the whole point set is also minimum when restricted to either of A or B. By induction these triangulations are the farthest point Delaunay triangulations of A and B respectively. But putting them together gives the farthest point Delaunay triangulation of the whole set. $\square$

We can also find the sharpest angle quickly, even when the points are not in general position.

Theorem 3. *The sharpest angle determined by a set of vertices of a convex polygon can be found in time $O(n)$.*

Proof: We can construct the farthest point Delaunay triangulation in linear time, using the parabolic transformation described in the introduction and the fast convex hull algorithm of Aggarwal et al. [1]. As we have seen, the sharpest angle is determined by two adjacent points and a third point on the same face of the triangulation. Lemma 2 can be used to show that if more than three points share a face, any of them will lead to the same angle. Therefore for each adjacent pair of points x and z, we can find the sharpest angle xyz; the sharpest angle overall must be one such. $\square$

3. Minimum Angle Triangulation for Cocircular Points

We have shown how to compute the triangulation minimizing the lexicographically ordered sequence of angles for point sets in convex but otherwise general position, using the farthest point Delaunay triangulation. The assumption of general position was necessary, because the farthest point Delaunay triangulation can be ambiguous if some four or more points are cocircular. In this case the input can be partially triangulated as above, leaving some faces of the resulting planar graph having more than three sides. Each such face has as its vertices a set of cocircular points. The overall minimum angle sequence triangulation can be found by piecing together the minimum angle sequence triangulations of these faces. Thus the general minimum angle sequence triangulation problem reduces to that for sets of cocircular points. In the rest of this section, we assume that our input points are cocircular; we now discuss how their minimum angle sequence triangulation can be found.

Consider an angle θ in a triangle formed by three of these points. Then, if $\theta \leq 90°$, the section of circle between the two opposite points covers an angle of $\theta/2$ [5]. Otherwise, the chord covers an angle of $(180° - \theta)/2$. Thus, the minimum angle sequence triangulation problem can be transformed into one of minimizing the lengths of the segments corresponding to the diagonals of the triangulation.

Also note that, since each interior diagonal has two sides, each angle θ corresponding to such a diagonal can be paired with another angle $180° - \theta$ in the same triangulation. The angles corresponding to the boundary of the convex hull must appear in any triangulation. Therefore, the minimum angle sequence triangulation also maximizes the sequence of largest angles in the triangles. Similarly, Mount and Saalfeld [6] noted that their max min angle triangulation algorithm for cocircular points also supplies the min max angle triangulation computed for more general input (but more slowly) by Edelsbrunner et al. [4].

First we consider the case that all distances between pairs of input points are distinct. In this case, a linear time algorithm follows easily from the following facts.

Lemma 3. *Let ab be the shortest distance among any two non-consecutive points in a set of vertices of a convex polygon. Then segment ab appears in the minimum angle sequence triangulation of the points.*

Proof: Any triangulation containing ab is lexicographically smaller than any triangulation not containing ab; therefore ac must be in the minimum triangulation. $\square$

Lemma 4. *Let points a, b, c, d, and e appear consecutively in a cocircular set of points. If $bd < ac$, $bd < ce$, and triangle bcd does not contain the center of the circle, then segment bd appears in the minimum angle sequence triangulation of the points.*

Proof: By lemma 3, we can imagine removing the smallest diagonals one by one, while preserving the property that the result is the minimum triangulation. At some point we will have removed all diagonals of length less than bd. We cannot have removed any diagonal touching point c, because all such diagonals have length at least $\min(ac, ce)$. Therefore bd will be the smallest remaining diagonal and so part of the minimum triangulation. □

Theorem 4. *Given a set of points in cocircular but otherwise general position, the triangulation lexicographically minimizing the sequence of angles, sorted from sharpest to least sharp, of its triangles, can be found in linear time.*

Proof: Simply remove locally minimal diagonals one by one, as indicated in lemma 4. The shortest diagonal is always such a local minimum, so at least one such diagonal can be found. The set of local minima can be kept in a linked list; whenever a diagonal is removed two new diagonals need to be considered as possible local minima, and two old diagonals may also become local minima. Therefore the total work per diagonal is $O(1)$, and the algorithm takes linear time. □

Finally, for completeness, we describe a polynomial time dynamic programming algorithm for computing the minimum angle sequence triangulation for cocircular points not necessarily in general position. Let $S_{i,j}$ be the sequence of segment lengths in the minimum angle sequence triangulation for the polygon formed by cutting off diagonal ij in the input set, on the side containing the circle's center. i, including the boundary segments. Denote by $M(S_1, S_2)$ the sequence formed by merging two sorted sequences of segment lengths. Then

$$S_{i,j} = \min_{i<k<j} M(S_{i,k}, S_{k,j}) \cup \{ij\}.$$

Each segment ik and kj is shorter than ij, so we can compute each $S_{i,j}$ by using this formula for the diagonals in order by length.

Theorem 5. *Given a set of n points in cocircular position, the triangulation lexicographically minimizing the sequence of angles, sorted from sharpest to least sharp, of its triangles, can be found in time $O(n^4)$.*

Proof: Each choice of k takes time $O(n)$ to compute the merge, there are $O(n)$ possible values k can take, and there are $O(n^2)$ pairs (i, j) for which to compute $S_{i,j}$. □

References

[1] A. Aggarwal, L.J. Guibas, J. Saxe, and P.W. Shor, A linear time algorithm for computing the Voronoi diagram of a convex polygon, 19th ACM Symp. Theory of Computing, 39–47, 1987.

[2] K.Q. Brown, Voronoi diagrams from convex hulls, Inform. Process. Lett. 9:223–228, 1979.

[3] H. Edelsbrunner, J. O'Rourke, and R. Seidel, Constructing arrangements of lines and hyperplanes with applications, SIAM J. Comput. 15:341–363, 1986.

[4] H. Edelsbrunner, T.S. Tan, and R. Waupotitsch, A polynomial time algorithm for the minmax angle triangulation, 6th ACM Symp. Computational Geometry, 44–52, 1990.

[5] Euclid, Elements, book III props. 20–21. Trans. Sir T.L. Heath, Dover, 1956.

[6] D.M. Mount and A. Saalfeld, Globally-equiangular triangulations of cocircular points in $O(n \log n)$ time, 4th ACM Symp. Computational Geometry, 143–152, 1988.

[7] R. Sibson, Locally equiangular triangulations, Computer J. 21:243–245, 1978.

[8] F.F. Yao, D.P. Dobkin, H. Edelsbrunner, and M.S. Paterson, Partitioning space for range queries, SIAM J. Comput. 18(2):371–384, 1989.