Discrete Comput Geom 10:47–65 (1993)

Discrete & Computational
Geometry

Edge Insertion for Optimal Triangulations*

M. Bern,[1] H. Edelsbrunner,[2] D. Eppstein,[3] S. Mitchell,[4] and T. S. Tan[5]

[1]Xerox Palo Alto Research Center, 3333 Coyote Hill Road,
Palo Alto, CA 94304, USA

[2]Department of Computer Science, University of Illinois,
Urbana, IL 61801, USA

[3]Department of Information and Computer Science, University of California,
Irvine, CA 92717, USA

[4]Center for Applied Mathematics, Cornell University,
Ithaca, NY 14853, USA

[5]Department of Information Systems and Computer Science,
National University of Singapore, Republic of Singapore

Abstract. Edge insertion iteratively improves a triangulation of a finite point set in $\Re^2$ by adding a new edge, deleting old edges crossing the new edge, and retriangulating the polygonal regions on either side of the new edge. This paper presents an abstract view of the edge insertion paradigm, and then shows that it gives polynomial-time algorithms for several types of optimal triangulations, including minimizing the maximum slope of a piecewise-linear interpolating surface.

1. Introduction

A *triangulation* of a finite set of points S in $\Re^2$ is a maximally connected, straight-line planar graph with vertex set S. Each bounded face is a triangle, and the triangulations includes the boundary of the convex hull. Triangulations find use in areas such as finite element analysis [2], [27], computational geometry [21],

* The research of the second author was supported by the National Science Foundation under Grant No. CCR-8921421 and under the Alan T. Waterman award, Grant No. CCR-9118874. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the view of the National Science Foundation. Part of the work was done while the second, third, and fourth authors visited the Xerox Palo Alto Research Center, and while the fifth author was on study leave at the University of Illinois.

[8], and surface approximation [7]. Applications typically require triangulations with "well-shaped" triangles, meaning—for example—that triangles with very small or large angles should be avoided. Taking a worst-case approach, the quality of a triangulation can be defined to be the quality of its worst triangle. Interesting algorithmic questions then arise when we ask for a triangulation of a given point set that optimizes some quality criterion. These questions take the form of *minmax* or *maxmin* problems, where the first quantifier is over all triangulations of the point set, and the second is over all triangles in the triangulation.

The problem of automatically generating optimal triangulations has been a subject for research since the 1960s (see, e.g., the discussion in [13]). In spite of this attention, very little is known about constructing optimal triangulations in polynomial time. Exhaustive search can be ruled out since a set of n points has, in general, exponentially many triangulations. Greedy approaches (such as eliminating triangles from worst to best) are ruled out by the NP-completeness of the following decision problem [20]: given a collection of points and edges, decide whether a subset of the edges defines a triangulation of the points.

Most positive results are related to the *Delaunay triangulation* [6]. It has been shown that among all triangulations of a given finite point set, the Delaunay triangulation optimizes various criteria. The Delaunay triangulation maximizes the minimum angle [26], minimizes the maximum circumscribing circle [5], and minimizes the maximum smallest enclosing circle [5], [22]. Efficient algorithms for constructing Delaunay triangulations are abundant in the literature and are based on such diverse algorithmic paradigms as edge-flipping [17], [18], divide-and-conquer [25], [15], geometric transformation [3], plane-sweep [12], and randomized incrementation [14].

Recently, Edelsbrunner, Tan, and Waupotitsch devised a polynomial-time algorithm that minimizes the maximum angle [10]. This algorithm constructs a minmax-angle triangulation by iteratively inserting a new edge, removing old edges crossed by the new edge, and then retriangulating the polygonal "holes" on either side of the new edge.

This paper presents an abstraction of the minmax-angle algorithm, which we call the *edge-insertion paradigm*, and applies it to obtain polynomial-time algorithms for some other optimal triangulation problems. The specific new results are an $O(n^2 \log n)$-time algorithm that constructs a triangulation with maxmin triangle height, an $O(n^3)$-time algorithm for minmax triangle eccentricity (distance from circumcenter), and—most significantly—an $O(n^3)$-time algorithm for finding a triangulated surface, interpolating given points in $\mathfrak{R}^3$, with minmax gradient. All three criteria are mentioned as open problems in a survey article on "systematic" triangulations [28].

Section 2 formulates the edge-insertion paradigm, which locally improves a triangulation according to a generic criterion. When instantiated to a specific criterion, the basic paradigm gives a local optimum in time $O(n^8)$. Section 3 states two abstract conditions for quality criteria, the first strictly weaker than the second. Section 4 proves that even the weaker condition suffices to show that the edge-insertion paradigm computes a global optimum; the argument is rather delicate. Section 5 discusses refinements of the basic paradigm with improved running times; here we show that the weaker condition implies an $O(n^3)$-time algorithm and the stronger condition implies an $O(n^2 \log n)$-time algorithm. (We do not yet know of any quality criteria globally optimized by the $O(n^8)$ basic algorithm, but not by the $O(n^3)$ algorithm.) Sections 6, 7, and 8 prove that the three specific optimization criteria mentioned above satisfy one or the other of the two conditions. Section 9 offers some concluding remarks.

2. The Edge-Insertion Paradigm

We start with some definitions. A *triangulation* of a finite point set S in $\mathfrak{R}^2$ is defined above as a maximally connected, straight-line planar graph with vertex set S. A *constrained triangulation* is a maximally connected, straight-line planar graph restricted to lie within a given connected polygonal region; the vertex set of the triangulation includes the vertices of the polygonal region along with any interior point "holes." Thus, a triangulation of a point set S is the special case in which the polygonal region is the convex hull of S. Another special case is a *polygon triangulation* in which there are no holes.

We denote by xy the relatively open line segment that connects the points $x, y \in \mathfrak{R}^2$. For $x, y, z \in \mathfrak{R}^2$, xyz is the open triangle with corners x, y, z. For a given finite point set S in $\mathfrak{R}^2$ and $x, y, z \in S$, we call xyz an *empty triangle* if all other points of S lie outside the closure of xyz.

Let μ be a function that maps each triangle xyz to a real value $\mu(xyz)$, called the *measure of* xyz. We restrict our attention to minmax criteria, that is, for each μ we consider the construction of a triangulation that minimizes the maximum $\mu(xyz)$ over all triangles xyz. Maxmin criteria can be simulated by considering $-\mu$. The measures of particular interest in this paper are largest angle, height (actually, negative height, since we desire maxmin height), eccentricity, and the gradient on a triangulated (nonplanar) surface.

The *measure* of a triangulation $\mathscr{A}$ is defined as $\mu(\mathscr{A}) = \max\{\mu(xyz) \mid xyz \text{ a triangle of } \mathscr{A}\}$. If $\mathscr{A}$ and $\mathscr{B}$ are two triangulations of the same point set, then $\mathscr{B}$ is called an *improvement* of $\mathscr{A}$, denoted $\mathscr{B} \prec \mathscr{A}$, if $\mu(\mathscr{B}) < \mu(\mathscr{A})$ or $\mu(\mathscr{B}) = \mu(\mathscr{A})$ and the set of triangles xyz in $\mathscr{B}$ with $\mu(xyz) = \mu(\mathscr{B})$ is a proper subset of the set of such triangles in $\mathscr{A}$. A triangulation $\mathscr{A}$ is *optimal* for μ if there is no improvement of $\mathscr{A}$.

The edge-insertion paradigm uses a natural local improvement operation, not surprisingly called an "edge-insertion." Given a triangulation $\mathscr{A}$ of a point set S, the *edge-insertion* of qs, for $q, s \in S$, goes as follows:

Function EDGE-INSERTION ($\mathscr{A}$, qs): triangulation.

1. $\mathscr{B} := \mathscr{A}$.
2. Add qs to $\mathscr{B}$ and remove from $\mathscr{B}$ all edges that intersect qs.
3. Retriangulate the polygonal regions P and R constructed in step 2.
4. **return** $\mathscr{B}$.

For now we assume that regions P and R (see Fig. 2.1) are retriangulated in an optimal fashion (minimizing the maximum μ), e.g., by dynamic programming [16].

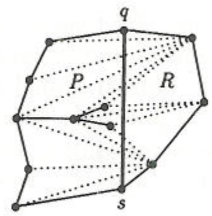

Fig. 2.1. Inserting qs leaves two polygonal regions P and R.

The basic, most general, version of the *edge-insertion paradigm* is given below; it tries all possible edge-insertions and halts when no edge-insertion improves the current triangulation.

Input. A set S of n points in $\Re^2$.
Output. An optimal triangulation $\mathscr{T}$ of S.

Algorithm
Construct an arbitrary triangulation $\mathscr{A}$ of S.
repeat $\mathscr{T} := \mathscr{A}$;
  for all pairs $q, s \in S$ **do**
    $\mathscr{B} :=$ EDGE-INSERTION ($\mathscr{A}$, qs);
    if $\mathscr{B} \prec \mathscr{A}$ **then** $\mathscr{A} := \mathscr{B}$; **exit** the for-loop **endif**
  endfor
until $\mathscr{T} = \mathscr{A}$.

The edge-insertion paradigm can be viewed as a generalization of the edge-flipping paradigm that computes a Delaunay triangulation [17], [18]. An edge-flip inserts the diagonal of a convex quadrilateral formed by two neighboring triangles; the process halts when no edge-flip improves the current triangulation. The simpler edge-flipping paradigm, however, fails to compute global optima for maximum angle, height, eccentricity, and slope, as we show in later sections of this paper.

We now argue that the basic algorithm above terminates after time $O(n^8)$. A single edge-insertion operation takes time $O(n^3)$ when retriangulation is done by dynamic programming [16], assuming the measures of any two triangles can be compared in constant time. The **for** loop thus takes time $O(n^5)$ per iteration of the repeat loop. Finally, the **repeat** loop is iterated at most $O(n^3)$ times, because there are only $\binom{n}{3}$ triangles spanned by S, and each iteration permanently discards at least one of them when it finds an improvement of the current triangulation.

Remark. The edge-insertion paradigm can be extended to constrained triangulations by limiting the edge-insertion operation to edges ab that lie in the interior of the restricting polygonal region. As a consequence, a triangulation that lexicographically minimizes the decreasing vector of triangle measures can be constructed in the nondegenerate case, that is, when $\mu(abc) \neq \mu(xyz)$ unless $abc = xyz$. Details can be found in [10].

3. Two Sufficient Conditions

We now formulate two conditions on measures μ, sufficient to show that the edge-insertion paradigm computes a global optimum (i.e., minmax μ). They are also sufficient to imply algorithms much faster than $O(n^8)$; these are given in Section 5.

Let S be a set of n points in $\Re^2$, let $\mathcal{B}$ be a triangulation of S, and let xyz be an empty triangle in S. We say that $\mathcal{B}$ *breaks* xyz *at* y if it contains an edge yt with $yt \cap xz \neq \varnothing$. Note that if $\mathcal{B}$ breaks xyz at y, then it cannot break xyz at x or z.

We call vertex y an *anchor* of an empty triangle xyz in point set S, if every triangulation $\mathcal{B}$ of S, with $\mu(\mathcal{B}) \leq \mu(xyz)$, either contains xyz or breaks xyz at y. For example, if $\mu(xyz)$ is the measure of the largest angle in xyz, and the largest angle has vertex y, then y is an anchor. Intuitively speaking, if a triangle has an anchor, it will be the triangle's "worst vertex." We can now give the two conditions on quality measures μ.

(I) (Weak Anchor Condition) For each triangulation $\mathcal{A}$, and each triangle xyz of $\mathcal{A}$ with $\mu(xyz) = \mu(\mathcal{A})$, there is an anchor vertex of xyz.

In other words, $\mathcal{B}$ can be an improvement of $\mathcal{A}$ only if it breaks a worst triangle of $\mathcal{A}$ at its anchor. Since $\mathcal{B}$ cannot break a triangle at two vertices, a triangle's anchor is unique in triangulations $\mathcal{A}$ with $\mu(\mathcal{A})$ larger than the minimum. Thus, if xyz is an empty isosceles triangle with two largest angles, then no triangulation can have minmax angle less than this largest angle.

(II) (Strong Anchor Condition) For each triangulation $\mathcal{A}$ and each triangle xyz of $\mathcal{A}$, there is an anchor vertex of xyz.

Notice that μ equal to the measure of the largest angle satisfies (II), since the largest angle in *any* triangle xyz—not just a worst triangle—must either appear in a triangulation $\mathcal{A}$ with $\mu(\mathcal{A}) \leq \mu(xyz)$, or be subdivided by it. An important difference between the weak and strong conditions is that in (I) the triangulation $\mathcal{A}$ that contains xyz plays an important role, while in (II) $\mathcal{A}$ is insignificant.

4. Proof of Correctness

The Cake Cutting Lemma (below) asserts that if $\mathcal{A}$ is not yet optimal for measure μ satisfying condition (I), then there is an edge whose insertion leads to an improvement, specifically an edge breaking a worst triangle at its anchor. In [10] this lemma is proved for the maximum angle measure using an argument that rotates edges of an optimal triangulation of S. While this argument works for angles, we need a different argument for the general class of measures that satisfy (I).

Before continuing, we remark that the regions P and R (created in step 2 of an edge-insertion) are not necessarily simple polygons in the usual meaning of the term. Although their interiors are always simply connected, there can be edges contained in the interiors of their closures, as shown in Fig. 2.1. Nevertheless, each such edge can be treated as if it consisted of two edges, one for each side, which then allows us to treat P and R as if they were simple polygons.

As usual, a *diagonal* of a simple polygon is a line segment that connects two vertices and—except at its endpoints—lies interior to the polygon. An *ear* is a triangle bounded by two polygon edges and one diagonal.

Lemma 4.1 (Cake Cutting). *Assume μ satisfies condition* (I). *Let $\mathcal{T} \prec \mathcal{A}$ be two triangulations of point set S. Let pqr be a triangle in $\mathcal{A}$ but not in $\mathcal{T}$ with $\mu(pqr) = \mu(\mathcal{A})$; let q be an anchor of pqr; and let qs be an edge in $\mathcal{T}$ that intersects pr. Let P and R be the polygons generated by adding qs to $\mathcal{A}$ and removing all edges that intersect qs. Then there are triangulations $\mathcal{P}$ and $\mathcal{R}$ of P and R with $\mu(\mathcal{P}) < \mu(pqr)$ and $\mu(\mathcal{R}) < \mu(pqr)$.*

Proof. We prove the assertion for P, and by symmetry it follows for R. The plan is to use the edges of $\mathcal{T}$ to locate ears of P with a small μ value, thereby obtaining $\mathcal{P}$. Each connected component of an edge of $\mathcal{T}$ intersected with P (that is, a segment seen through the "window" P) is called a *clipped edge*. As P is not necessarily convex, several clipped edges can belong to the same edge of $\mathcal{T}$. A clipped edge partitions P into two polygons, the *near side* supported by qs and the *far side* not supported by qs.

If no clipped edge exists in the window, then P has only three vertices and therefore must be a triangle of $\mathcal{T}$. This triangle is not in $\mathcal{A}$, which implies that its measure is less than $\mu(\mathcal{A})$, because any triangle of $\mathcal{T}$ with measure $\mu(\mathcal{A})$ is also a triangle of $\mathcal{A}$. So assume the existence of at least one clipped edge. Denote by $q = p_0, p_1, \ldots, p_k, p_{k+1} = s$ the sequence of vertices of P.

Claim 1. *For $1 \le j \le k$, if $\angle p_{j-1}p_jp_{j+1} < \pi$, then $p_{j-1}p_{j+1}$ is a diagonal of P.*

Proof of Claim 1. By construction of P, it is possible to find nonintersecting line segments $p_{j-1}x$ and $p_{j+1}y$, both inside P, so that x and y lie on qs. (If $j = 1$, then $x = p_{j-1} = q$; if $j = k$, then $y = p_{j+1} = s$.) The (possibly degenerate) pentagon $xp_{j-1}p_jp_{j+1}y$ is part of P, and, because the interior angles at p_j, x, and y measure less than π, edge $p_{j-1}p_{j+1}$ is a diagonal of the pentagon and therefore also of P. □

Claim 2. *There is at least one clipped edge whose far side is a triangle.*

Proof of Claim 2. Let xy be a clipped edge so that its far side, F, contains no further clipped edge. Consider the triangle in $\mathcal{T}$ that lies on the same side of xy as F. Polygon F must be a subset of this triangle, and since all vertices of F—except possibly x and y—are points in S, F must be a triangle xp_iy. □

The clipped edges xy that satisfy Claim 2 fall into four classes as illustrated in Fig. 4.1. An ear $p_{i-1}p_ip_{i+1}$ so that xy is a clipped edge with far side xp_iy can now be removed from P, leaving a polygon P' with one less vertex. Claims 1 and 2 remain

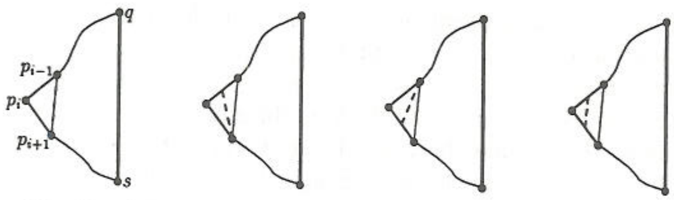

Fig. 4.1. A "maximally far" clipped edge locates a good ear of P.

true for P' because the removed ear is not supported by qs. Hence we can iterate and compute a triangulation $\mathscr{P}$ of P. Symmetrically, we get a triangulation $\mathscr{R}$ of R. Let $\mathscr{B}$ be the triangulation of S thus obtained.

Claim 3. *$\mu(abc) < \mu(pqr)$ for all triangles abc in $\mathscr{P}$ and $\mathscr{R}$.*

Proof of Claim 3. Let abc be a triangle in $\mathscr{P}$ or $\mathscr{R}$ with maximum μ. Assume without loss of generality that abc is a triangle of $\mathscr{P}$ and that $a = p_i$, $b = p_j$, $c = p_l$ with $i < j < l$. At the time immediately before abc was removed by adding the edge ac there was a clipped edge xy with far side xby, as shown in Fig. 4.2. Hence, $\mathscr{T}$ does not break abc at b, and, by construction, $\mathscr{A}$ breaks abc at b and therefore neither at a nor at c.

If $xy = ac$ (as in the leftmost picture in Fig. 4.1), then abc is a triangle in $\mathscr{T}$ that is not in $\mathscr{A}$, and therefore $\mu(abc) < \mu(pqr)$. So assume $xy \neq ac$, and assume for the sake of contradiction that $\mu(abc) \geq \mu(pqr) = \mu(\mathscr{A}) \geq \mu(\mathscr{T})$. Since we chose abc to have maximum μ in $\mathscr{P}$ or $\mathscr{R}$, this means that $\mu(abc) = \mu(\mathscr{B})$. Then condition (I) requires abc to have an anchor. However, b cannot be the anchor of abc, because $\mathscr{T}$ neither contains abc nor breaks abc at b. Similarly, neither a nor c can be an anchor of abc because $\mathscr{A}$ neither contains abc nor breaks abc at a or c. This contradiction completes the proofs of Claim 3 and Lemma 4.1. □

The Cake Cutting Lemma now shows that the basic edge-insertion paradigm cannot get stuck in a local optimum for μ satisfying condition (I).

Lemma 4.2. *Assume μ satisfies condition* (I). *Let $\mathscr{A}$ be a nonoptimal triangulation of point set S. Then there is an edge-insertion operation that improves $\mathscr{A}$.*

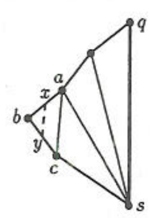

Fig. 4.2. Triangle abc cannot have an anchor.

Proof. Let $\mathscr{B}$ be an improvement of $\mathscr{A}$ and consider a triangle pqr in $\mathscr{A}$ with $\mu(pqr) = \mu(\mathscr{A})$ that is not in $\mathscr{B}$. Condition (I) requires pqr to have an anchor, say q, so $\mathscr{B}$ must contain an edge qs with $qs \cap pr \neq \emptyset$. Let P and R be the polygonal regions generated by adding qs and deleting the edges that intersect qs. The Cake Cutting Lemma implies that there are polygon triangulations $\mathscr{P}$ and $\mathscr{R}$ of P and R with $\mu(\mathscr{P})$ and $\mu(\mathscr{R})$ both smaller than $\mu(pqr)$. $\square$

Remark. Lemmas 4.1 and 4.2 remain true for constrained triangulations provided the optimization criterion satisfies (I) or (II) in this more general setting. This is indeed the case for all criteria considered in this paper.

5. Refinements of the Paradigm

The refined versions of edge-insertion differ from the basic paradigm in two major ways. First, edge-insertions are restricted to candidate edges qs that break a worst triangle pqr at its anchor q. Second, the two polygonal regions created by adding edge qs are retriangulated by repeatedly removing ears (as in the proof of the Cake Cutting Lemma), rather than by dynamic programming.

Outline of Refinements

Let $\mathscr{A}$ be a triangulation with worst triangle pqr, that is, $\mu(pqr) = \mu(\mathscr{A})$, and let q be the anchor of pqr. We denote by $qs_1, qs_2, \ldots$ the sequence of candidate edges. This order may be arbitrary for the $O(n^3)$ refinement, but, for criteria satisfying condition (II), a carefully chosen order speeds up the running time to $O(n^2 \log n)$. Both refinements are specializations of the algorithm given below. We use the notation $s_{i+1} = \text{NEXT}(s_i)$.

Algorithm
  Construct an arbitrary triangulation $\mathscr{A}$ of S.
  repeat $\mathscr{T} := \mathscr{A}$;
    find a worst triangle pqr in $\mathscr{A}$, let q be its anchor, and set $s := s_1$;
    while s is defined **do**
      $\mathscr{B} := \mathscr{A}$, add qs to $\mathscr{B}$, and remove all edges that intersect qs;
      (partially) triangulate the two polygonal regions P and R
        by cutting off ears xyz with $\mu(xyz) < \mu(pqr)$;
      if P and R are completely triangulated **then** $\mathscr{A} := \mathscr{B}$; **exit** the while-loop
                                else $s := \text{NEXT}(s)$
      endif
    endwhile
  until $\mathscr{T} = \mathscr{A}$ and all worst triangles pqr in $\mathscr{A}$ have been tried.

In an implementation of the algorithm we would not really copy entire triangulations. Instead of the assignment $\mathscr{T} := \mathscr{A}$, we would use a flag to check whether an iteration of the repeat-loop produced an improved triangulation. The assignment $\mathscr{B} := \mathscr{A}$ can be avoided by making changes directly in $\mathscr{A}$ and undoing them to the extent necessary. The remainder of this section explains some of the steps in greater detail and analyzes the complexity of the two refinements.

Triangulating by Ear Cutting

Suppose an edge qs has been added to $\mathscr{B}$ and the edges that intersect qs have been removed, thus creating two polygonal regions P and R. Let $q = p_0, p_1, \ldots, p_k, p_{k+1} = s$ be the sequence of vertices of P and let $q = r_0, r_1, \ldots, r_m, r_{m+1} = s$ be the corresponding sequence for R. As in the proof of the Cake Cutting Lemma, the two regions are (partially) triangulated by repeatedly removing ears with measures less than $\mu(pqr)$. As implied by the proof, the sequence in which the ears are removed is immaterial as long as only the last is supported by qs. This method may be implemented using a stack for the vertices of P (R), so that it runs in time linear in the size of P (R). In the case of P, the stack is initialized by pushing p_0 and p_1. After that, for $i := 2$ to $k + 1$ we push vertex p_i, and whenever the three topmost vertices, $z = p_i, y, x$, define a triangle with $\mu(xyz) < \mu(pqr)$ we pop y, the second vertex from the top. The triangulation is complete if, at the end of the process, $p_{k+1} = s$ and $p_0 = q$ are the only two vertices on the stack.

Theorem 5.1. *Let S be a set of n points in $\mathfrak{R}^2$, and let μ be a measure that satisfies* (I) *so that given a worst triangle its anchor can be computed in constant time.*

(1) *A constrained or unconstrained triangulation of S that minimizes the maximum triangle measure can be constructed in time $O(n^3)$ and storage $O(n^2)$.*
(2) *In the nondegenerate case* (i.e., *when* $\mu(xyz) \neq \mu(abc)$ *unless* $xyz = abc$) *the (unique) triangulation that lexicographically minimizes the decreasing vector of triangle measures can be constructed in the same amount of time and storage.*

Proof. To achieve the claimed bounds, we use the algorithm above, along with two data structures requiring a total of $O(n^2)$ storage. First, the quad-edge data structure of Guibas and Stolfi [15] stores the triangulation in $O(n)$ memory and admits common operations, such as removing an edge, adding an edge, and walking from one edge to the next in constant time each.

Second, to record the status of candidate edges, we use an n-by-n bit array whose elements correspond to the edges defined by S. If the insertion of a candidate edge qs is unsuccessful, that is, the triangulation of P or R cannot be completed, then we know by the Cake Cutting Lemma that qs cannot be in any improvement of the current triangulation. We then set the bit for qs, so that we do not attempt the insertion of qs again. If the insertion of qs is successful, we set the bit for the edge pr; because every improvement breaks pr (by condition (I)), it cannot be in any later improvement. The bit array can also be used to compute the sequence of candidate edges $qs_1, qs_2, \ldots$: scan the row corresponding to q and take all edges qs that intersect pr and whose flag has not yet been set.

Each edge-insertion, whether successful or not, causes a new flag set for one of the $\binom{n}{2}$ edges defined by S. Therefore, at most $\binom{n}{2}$ edge-insertions are carried out taking a total of $O(n^3)$ time. Part (1) of the claim follows because an initial triangulation can be constructed in time $O(n \log n)$, most straightforwardly by plane-sweep (see Section 8.3.1 of [8]).

To obtain a triangulation that lexicographically minimizes the entire vector of triangle measures we solve a sequence of constrained triangulation problems as in [10]. The first constraining region is defined by the points and edges on the boundary of the convex hull of S with the other points forming holes. After computing an optimal triangulation as in (1), we remove the worst triangle (which is unique by nondegeneracy assumption) from the constraining region and iterate until the region is empty. The time is still $O(n^3)$ because each edge needs to be inserted at most once during the entire process. □

A Special Order of Insertions for Condition (II)

For measures μ that satisfy (II) we define a special sequence $qs_1, qs_2, \ldots, qs_l$ of edge-insertions, as in [10]. The first edge, qs_1, has the property that it intersects pr, but otherwise it intersects as few edges as possible. As we explain below, each subsequent $s_{i+1} = \text{NEXT}(s_i)$ lies on a particular side of qs_i, and, on this side, the set of edges in the current triangulation $\mathscr{B}$ that intersect qs_{i+1} is the smallest proper superset of the edges that intersect qs_i. The index l is the smallest integer for which qs_l leads to an improvement or s_{l+1} is undefined.

On the insertion of qs_i, the retriangulation process either completes its task or it gets stuck because all ears of the remaining regions have measure at least $\mu(pqr)$. Let us now consider the case where the triangulation of P cannot be completed, as this is the case for which we need to define $\text{NEXT}(s_i)$. In this case the stack contains $k + 2 \geq 3$ vertices $q = p_0, p_1, \ldots, p_k, p_{k+1} = s_i$ defining the remaining region $P' \subseteq P$; each ear $p_{j-1}p_jp_{j+1}$ of P' has measure at least $\mu(pqr)$.

Lemma 5.2. *Let $\mathscr{T}$ be an improvement of $\mathscr{B}$ for μ satisfying condition* (II), *and let P' be the uncompleted part of P as above. Then all edges of $\mathscr{T}$ that intersect P' also intersect qs_i. In particular, all edges of $\mathscr{T}$ incident to q avoid P'.*

Proof. As in the proof of the Cake Cutting Lemma we consider P' as a "window" through which we see clipped edges of $\mathscr{T}$. Now suppose the claim is not true, that is, there is a clipped edge that does not have one of its endpoints on qs_i. Then, as in the proof of the Cake Cutting Lemma, we can find a clipped edge xy whose far side is a triangle xp_jy. However, now condition (II) implies $\mu(\mathscr{T}) > \mu(p_{j-1}p_jp_{j+1})$ if p_j is an anchor of the ear $p_{j-1}p_jp_{j+1}$, and $\mu(\mathscr{B}) > \mu(p_{j-1}p_jp_{j+1})$ if p_{j-1} or p_{j+1} is an anchor. This contradicts the assumption that P' has no such ear. □

It is interesting to observe that the proof of Lemma 5.2 breaks down if we assume that μ satisfies only (I), since $p_{j-1}p_jp_{j+1}$ need not be a worst triangle.

As we search for an insertion, we maintain an open wedge W containing all the remaining candidate edges. Initially, W is the wedge between the ray $\overrightarrow{qp}$ (starting at q and passing through p) and the ray $\overrightarrow{qr}$. If the edge-insertion of qs_i turns out to be unsuccessful because the triangulation of P cannot be completed, then Lemma 5.2 allows us to redefine W as the part of the old W on R's side of $\overrightarrow{qs_i}$. Similarly, if the triangulation of R cannot be completed, then W can be narrowed down to P's side of $\overrightarrow{qs_i}$. (As a consequence, if neither P nor R can be completely triangulated, then it is impossible to improve the current triangulation by breaking pqr at q.)

As soon as one of P or R has been found to be noncompletable, wedge W is updated and an edge-insertion is attempted with $s_{i+1} = \text{NEXT}(s_i)$. If it is P that could not be completed (the R case is symmetrical), then we choose s_{i+1} by looking first at the triangle on the far side of $r_m r_{m+1}$ (the last edge of R) from q. If the third vertex s of this triangle lies in wedge W, then we choose s_{i+1} to be s. If this is not the case, then we move on to the next triangle sharing an edge with $r_m r_{m+1} s$, and test whether its far vertex z lies in the wedge. We eventually either run out of triangles (then no edge-insertion at q is possible) or we find a vertex s_{i+1} such that the set of edges in $\mathscr{B}$ that intersect qs_{i+1} is the smallest proper superset of the edges that intersect qs_i. See Fig. 5.1.

When we move from qs_i to qs_{i+1}, most of the work done to triangulate P and R can be saved. Assume that qs_i has failed because P could not be completely triangulated. Because qs_{i+1} intersects $r_m r_{m+1}$ all ears cut off P remain the same and do not have to be reconsidered. On the other hand, r_{m+1} is no longer a vertex of R, so all ears cut off R that are incident to r_{m+1} must be returned to R's territory. When we move to qs_{i+1} some additional edges are removed from $\mathscr{B}$ which, in effect, expands P and R. The new vertices can be pushed on their respective stacks, one by one, so that the triangulation process can continue where it left off.

The only place where we waste time in this process (i.e., where time spent is not proportional to good ears found) is when ears cut off R are returned to R. Since ears are returned for only one polygon, we can limit the waste by strictly alternating between cutting an ear of P and one of R. This way, for each returned

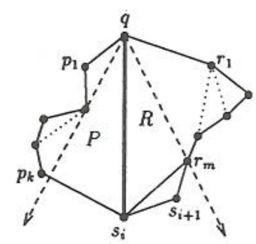

Fig. 5.1. The two rays define the current W, and the dotted line segments indicate those ears removed from P and R. If P is found to be noncompletable, then the next candidate edge qs_{i+1} lies in the updated W defined by $\overrightarrow{qs_i}$ and the ray passing through R.

ear (except maybe the last) there is a permanently removed ear. Therefore, the total number of operations performed while inserting $qs_1, qs_2, \ldots, qs_l$ is linear in the number of edges in $\mathscr{B}$ that intersect qs_l.

As in the proof of Theorem 5.1, a successful edge-insertion, complete with retriangulation, takes time linear in the number of old edges intersected by the new edge. We now prove that the old edges removed will never be reinserted in any later successful edge-insertion.

Lemma 5.3. *Assume μ satisfies condition* (II), *let $\mathscr{A}$ be a triangulation of S with worst triangle pqr, and let $\mathscr{B}$ be obtained from $\mathscr{A}$ by the successful insertion of edge qs_l. Then no edge xy in $\mathscr{A}$ that intersects qs_l can be an edge of any improvement of $\mathscr{B}$.*

Proof. Lemma 5.2 implies that every improvement of $\mathscr{B}$ has an edge qw that lies inside the wedge W computed when qs_l is inserted into $\mathscr{A}$. Every edge xy in $\mathscr{A}$ that intersects qs_l also intersects every other edge qt with $t \in W$. In particular, $xy \cap qw \neq \emptyset$ which implies that xy is neither in $\mathscr{B}$ nor in any improvement of $\mathscr{B}$. □

Theorem 5.4. *Let S be a set of n points in $\Re^2$ and let μ be a measure that satisfies* (II) *so that given a triangle its anchor can be computed in constant time.*

(1) *A constrained or unconstrained triangulation of S that minimizes the maximum triangle measure can be constructed in time $O(n^2 \log n)$ and storage $O(n)$.*
(2) *In the nondegenerate case (i.e., when $\mu(xyz) \neq \mu(abc)$ unless $xyz = abc$) the (unique) triangulation that lexicographically minimizes the decreasing vector of triangle measures can be constructed in the same amount of time and storage.*

Proof. As before, the algorithm uses the quad-edge data structure of [15] to store the triangulation. The bit array, however, is replaced by a priority queue that holds the triangles of $\mathscr{A}$ ordered by measure. It admits inserting and deleting triangles and finding a triangle with maximum measure in logarithmic time [4]. Lemma 5.3 implies that only $O(n^2)$ edges and triangles are manipulated in the main loop of the algorithm, which thus takes time $O(n^2 \log n)$. Lemma 5.3 also implies a quadratic upper bound on the number of iterations of the repeat-loop, which implies that the total time needed to find worst triangles pqr is also $O(n^2 \log n)$. This proves part (1), and part (2) follows from the same argument as in Theorem 5.1. □

6. Maximizing the Minimum Height

The *height* $\eta(xyz)$ of triangle xyz is the minimum distance from a vertex to the opposite edge. A *maxmin height triangulation* of S maximizes the smallest height of its triangles, over all triangulations of S. Although the maxmin height,

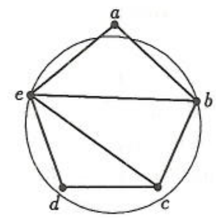

Fig. 6.1. Flipping either *be* or *ce* locally decreases the minimum height. Thus, the edge-flip method cannot change this triangulation into the optimal one.

the maxmin angle, and the minmax angle criteria all tend to avoid thin and elongated triangles, they do not necessarily define the same optima. Indeed, four-point examples can be constructed to show that the three criteria are pairwise different.

The edge-flipping strategy [17], [18] applied to the maxmin height criterion does not always succeed in computing an optimal triangulation. Consider a regular pentagon *abcde* and the circle through the five points. Perturb a slightly to a point outside the circle and c and d slightly to points inside the circle so that $h(c, db) < h(d, ec) < h(b, ca) = h(e, ad) < h(a, be)$, where we write $h(x, yz)$ for the minimum distance between a point x and a line through points y and z. See Fig. 6.1. The maxmin-height triangulation uses diagonals ac and ad. If the current triangulation uses be and ce, however, no edge-flip can result in a better triangulation.

We now show that $-\eta$ satisfies condition (II), when we define the vertices of xyz with maximum angle to be anchors. It follows that maxmin height triangulations can be constructed by the $O(n^2 \log n)$-time implementation of the edge-insertion paradigm.

Lemma 6.1. *Let xyz be a triangle of a triangulation $\mathscr{A}$ of S and let $\eta(xyz) = h(y, zx)$. Then $\eta(\mathscr{T}) < \eta(xyz)$ for any triangulation $\mathscr{T}$ of S that neither contains xyz nor breaks xyz at y.*

Proof. The height $\eta(xyz) = h(y, zx)$ is the distance between y and a point $s \in zx$. Assume that xyz is not in $\mathscr{T}$ and that $\mathscr{T}$ does not break xyz at y. Therefore, there exists a triangle uyv in $\mathscr{T}$ so that either $u = x$ and $uv \cap yz \neq \varnothing$ (rename vertices if necessary), or uv intersects both yx and yz. In both cases, $\eta(uyv) \leq h(y, uv) < \eta(xyz)$ because $uv \cap ys \neq \varnothing$. □

It should be clear that Lemma 6.1 also holds for constrained triangulations of S. Theorem 5.4 then implies that a maxmin height triangulation, and in the nondegenerate case a triangulation lexicographically maximizing the increasing vector of heights, can be computed in time $O(n^2 \log n)$ and storage $O(n)$.

7. Minimizing the Maximum Eccentricity

Consider a triangle xyz and let (c_1, ρ_1) be its circumcircle, with center c_1 and radius ρ_1. The *eccentricity* of xyz, $\varepsilon(xyz)$, is the infimum over all distances between c_1 and points of xyz. Clearly, $\varepsilon(xyz) = 0$ iff c_1 lies in the closure of xyz. Note that eccentricity is related to the size of the maximum angle, $\alpha(xyz)$, only with large triangles counting more. Specifically, unless $\varepsilon(xyz) = \varepsilon(abc) = 0$,

$$\alpha(xyz) < \alpha(abc) \qquad \text{iff} \quad \frac{\varepsilon(xyz)}{\rho_1} < \frac{\varepsilon(abc)}{\rho_2},$$

where ρ_2 is the radius of the circumcircle of abc. The triangulation of the pentagon in Fig. 6.1 can be used to show that edge-flipping does not always succeed in minimizing the maximum eccentricity.

Eccentricity is our first example of a measure satisfying condition (I), but not (II). Consider Fig. 7.1. In this figure, vertices u and v lie very close to yx and yz, respectively, so that the circumcircle of uyv is significantly smaller than the one of xyz, and $\varepsilon(uyv) < \varepsilon(xyz)$. In fact, $\varepsilon(xyz)$ exceeds the eccentricity of every triangle of the minmax-eccentricity triangulation $\mathcal{T}$, even though $\mathcal{T}$ does not break xyz at any of its vertices. We now show that ε satisfies the weaker condition (I). It turns out that y is an anchor of xyz only if a largest angle in xyz is at y.

Lemma 7.1. *Let xyz be a triangle of a triangulation $\mathcal{A}$ of S, such that $\varepsilon(xyz) > 0$, and let y be a vertex with maximum angle in xyz. Then* $\max\{\varepsilon(\mathcal{A}), \varepsilon(\mathcal{T})\} > \varepsilon(xyz)$ *for every triangulation $\mathcal{T}$ of S that neither contains xyz nor breaks xyz at y.*

Proof. Assume that $\mathcal{T}$ neither contains xyz nor breaks it at y. Therefore, $\mathcal{T}$ must contain a triangle uyv so that $u = x$ and $uv \cap yz \neq \varnothing$ (renaming vertices if necessary), or uv intersects yx and yz, as in Fig. 7.2. Let (c_1, ρ_1) be the circumcircle of xyz. If neither u nor v are enclosed by this circle, then $\varepsilon(xyz) < \varepsilon(uyv) \le \varepsilon(\mathcal{T})$. Otherwise, assume that v is enclosed by (c_1, ρ_1) and consider the line segment $c_1 v$.

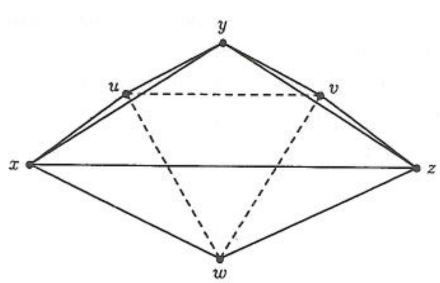

Fig. 7.1. $\mathcal{T}$ is the triangulation with diagonals uv, vw, and wu, and $\mathcal{A}$ the one with diagonals xy, yz, and zx. Then $\varepsilon(\mathcal{T}) < \varepsilon(xyz) \le \varepsilon(\mathcal{A})$, but $\mathcal{T}$ does not break xyz at any of its vertices, in contradiction to condition (II).

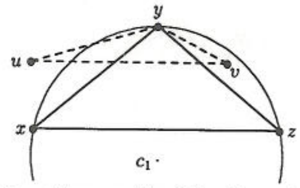

Fig. 7.2. The triangle xyz in $\mathscr{A}$ is neither contained in $\mathscr{T}$ nor is it broken at y by $\mathscr{T}$. Therefore, $\mathscr{T}$ contains a triangle uyv that intersects xyz as shown. There must be a triangle in $\mathscr{A}$ with eccentricity greater than $\varepsilon(xyz)$ intersecting c_1v.

It intersects a sequence of edges of $\mathscr{A}$, ordered from c_1 to v. For an edge ab in this sequence let abc be the supporting triangle so that c and c_1 lie on different sides of ab. Assume that ab is the first edge in the sequence so that (c_1, ρ_1) encloses c but not a and not b. Then $\varepsilon(\mathscr{A}) \geq \varepsilon(abc) > \varepsilon(xyz)$. □

Theorem 5.1 thus implies that a minmax-eccentricity triangulation of n points can be constructed in time $O(n^3)$ and storage $O(n^2)$. In the nondegenerate case the same time and storage suffice to construct a triangulation lexicographically minimizing the decreasing vector of eccentricities.

8. Minimizing the Maximum Slope

Consider a function $f: \Re^2 \to \Re$ defining a surface $x_3 = f(x_1, x_2)$ in $\Re^3$. The *gradient* of f is the vector $\nabla f = (\partial f/\partial x_1, \partial f/\partial x_2)$, each component of which is itself a function from $\Re^2$ to $\Re$. Define $\nabla^2 f = (\partial f/\partial x_1)^2 + (\partial f/\partial x_2)^2$, and call $\sqrt{\nabla^2 f}$ at a point (x_1, x_2) the *slope* at this point.

Let S be a point set in $\Re^2$ and let $\hat{S}$ be the corresponding set in $\Re^3$ where each point of S has a third coordinate called *elevation.* For a point x of S, we write $\hat{x}$ for the "lifted" point, that is, the corresponding point in $\hat{S}$. Analogous to the definitions in $\Re^2$, $\hat{x}\hat{y}$ denotes the relatively open line segment with endpoints $\hat{x}$ and $\hat{y}$, and $\hat{x}\hat{y}\hat{z}$ denotes the relatively open triangle with corners $\hat{x}, \hat{y}, \hat{z}$. We can think of $\hat{x}\hat{y}\hat{z}$ as a partial function f on $\Re^2$, defined within xyz. At each point in xyz, the gradient is well defined and the same as for any other point in xyz. We can therefore set $\sigma(xyz)$ equal to the slope at any point of xyz, and call it the *slope* of xyz. For a triangulation $\mathscr{A}$ of S define $\sigma(\mathscr{A}) = \max\{\sigma(xyz) \mid xyz \text{ a triangle of } \mathscr{A}\}$, as usual. A *minmax-slope triangulation of* S minimizes the maximum σ of any triangle.

Triangulations are commonly used to compute surfaces interpolating point set data with elevations. Rippa [23] recently proved that, regardless of elevations, the Delaunay triangulation minimizes the integral (over the convex hull of S) of $\nabla^2 f$ among all triangulations of S. See [7] for other interesting optimization criteria.

The five-point example of Fig. 6.1 again shows that the edge-flipping strategy does not in general compute a minmax-slope triangulation. Just imagine that points a, b, c, d, e are not perturbed and thus form a regular pentagon. Let the

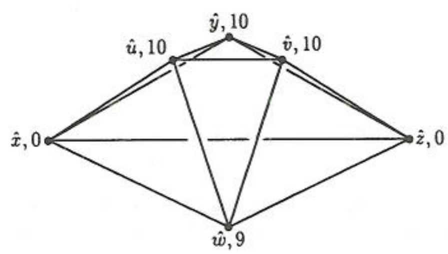

Fig. 8.1. Triangulation $\mathscr{T}$ with diagonals uv, vw, and wu is an improvement of $\mathscr{A}$ with diagonals xy, yz, and zx. $\mathscr{T}$ has no triangle with slope as large as $\sigma(xyz)$, but does not break xyz at any of its vertices.

elevations of a, b, c, d, e be 5, 11, 0, 10, 0, in this sequence. The optimal triangulation is defined by the diagonals ac and ad, and the current triangulation (with diagonals be and ce as shown) cannot be improved by a single edge-flip.

As in the case of eccentricity, we can show that σ does not satisfy the strong condition (II). Figure 8.1 gives a six-point example in which an improvement $\mathscr{T}$ of $\mathscr{A}$, does not break a triangle xyz with $\sigma(\mathscr{T}) < \sigma(xyz)$.

Observe that the direction of steepest descent at any point on a triangle xyz is given by $\Delta = -\nabla f$ at that point. We call the vertex y a *peak* of xyz unless the line $y + \lambda\Delta$, $\lambda \in \Re$, intersects the closure of xyz only at y. In other words, a peak is a vertex first hit when sweeping with a line perpendicular to the direction of steepest descent. In the nondegenerate case xyz has only one peak, but if Δ is parallel to an edge, then there are two peaks. Call the intersection of the closure of $\hat{x}\hat{y}\hat{z}$ with the plane parallel to the x_3-axis through $y + \lambda\Delta$ the *descent line* $\ell(xyz)$ of xyz, assuming y is an anchor of xyz.

The remainder of this section shows that σ does satisfy the weak condition (I). In fact, each peak of a worst triangle is an anchor. For technical reasons it is necessary to assume that no four points of $\hat{S}$ are coplanar. Indeed, the strict inequality in Lemma 8.1 is incorrect without this assumption. (This general position assumption, however, does not diminish the generality of our algorithm, because a simulated perturbation of the points can be used to enforce general position [9]).

Lemma 8.1. *Let xyz be a triangle of a triangulation $\mathscr{A}$ of S, and let the intersection of line $y + \lambda\Delta$ with the closure of xyz be strictly larger than point y, and let y be a peak of xyz. Then* $\max\{\sigma(\mathscr{A}), \sigma(\mathscr{T})\} > \sigma(xyz)$ *for every triangulation $\mathscr{T}$ of S that neither contains xyz nor breaks xyz at y.*

Proof. The slope of xyz, $\sigma(xyz)$, is also the slope of the descent line $\ell_1 = \ell(xyz)$. Assume without loss of generality that ℓ_1 descends from $\hat{y}$ down to where it meets the closure of $\hat{x}\hat{z}$. (If it ascends, we use the same argument only with the x_3-axis reversed.) Assume also that $\mathscr{T}$ neither contains xyz nor breaks it at y. It follows that $\mathscr{T}$ contains an edge uv so that either $u = x$ and $uv \cap yz \neq \varnothing$ (rename vertices if necessary) or uv intersects both yx and yz. If $\sigma(uyv) > \sigma(xyz)$, then $\sigma(\mathscr{T}) > \sigma(xyz)$ and there is nothing to prove.

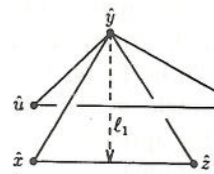

Fig. 8.2. The triangle xyz with peak y in $\mathscr{A}$ is neither contained in $\mathscr{T}$ nor is it broken at y by $\mathscr{T}$. Therefore, $\mathscr{T}$ contains a triangle uyv that intersects xyz as shown. It is possible that $u = x$ or $v = z$, but not both at the same time.

Otherwise, the edge $\hat{u}\hat{v}$ must pass *above* ℓ_1 in $\mathfrak{R}^3$. By this we mean that there is a line parallel to the x_3-axis that meets $\hat{u}\hat{v}$ and ℓ_1 and the elevation of its intersection with $\hat{u}\hat{v}$ exceeds the elevation of its intersection with ℓ_1, as in Fig. 8.2. Then at least one of $\hat{u}$ and $\hat{v}$ must lie above the plane h_1 through points $\hat{x}, \hat{y}, \hat{z}$; say $\hat{v}$ lies above h_1. Consider the triangle yvz, and note that it is not necessarily a triangle of $\mathscr{A}$ or $\mathscr{T}$, nor even an empty triangle of S. We have $\sigma(yvz) > \sigma(xyz)$ because the x_3-parallel projection of ℓ_1 onto the plane h_2 through $\hat{y}, \hat{v}, \hat{z}$ is steeper than ℓ_1 but not steeper than $\ell_2 = \ell(yvz)$. We distinguish three cases depending on which vertex is the peak of yvz, that is, through which one a line of steepest descent of $\hat{y}\hat{v}\hat{z}$ passes.

Case 1: v is a peak of yvz. Then ℓ_2 connects $\hat{v}$ with a point on the closure of $\hat{y}\hat{z}$. Consider the intersection of $\mathscr{A}$ with a plane parallel to the x_3-axis through ℓ_2. This intersection includes a polygonal chain that connects $\hat{v}$ with that same point on the closure of $\hat{y}\hat{z}$ (since yz is an edge in $\mathscr{A}$). One of the segments in the chain must have slope at least the average slope of the chain; hence one of the triangles abc in $\mathscr{A}$ has $\sigma(abc) \geq \sigma(yvz) > \sigma(xyz)$, and therefore $\sigma(\mathscr{A}) > \sigma(xyz)$.

Case 2: z is a peak of yvz. Then ℓ_2 connects $\hat{z}$ with a point on the closure of $\hat{y}\hat{v}$. Then we use the same argument as in Case 1, only applied to $\mathscr{T}$. Since yv is an edge in $\mathscr{T}$ at least one of the triangles abc in $\mathscr{T}$ that intersect the projection of ℓ_2 has $\sigma(abc) \geq \sigma(yvz) > \sigma(xyz)$, and therefore $\sigma(\mathscr{T}) > \sigma(xyz)$.

Case 3: y is a peak of yvz. In this case ℓ_2 connects $\hat{y}$ with a point $\hat{w}$ on the closure of $\hat{v}\hat{z}$. Furthermore, it is impossible that ℓ_2 descends from $\hat{y}$ to $\hat{w}$ because $\hat{w}$ lies above h_1, which contradicts $\sigma(yvz) > \sigma(xyz)$. Thus, it must be that ℓ_2 descends from $\hat{w}$ down to $\hat{y}$. Then $\sigma(uyv) > \sigma(yvz)$ because $\hat{u}\hat{v}$ passes above ℓ_2. However, $\sigma(yvz) > \sigma(xyz)$, so we have shown $\sigma(\mathscr{T}) > \sigma(xyz)$. □

Note that Lemma 8.1 also holds for constrained triangulations of S. We can therefore apply Theorem 5.1 and obtain an $O(n^3)$-time and $O(n^2)$-storage algorithm for constructing a minmax slope triangulation, and in the nondegenerate case for constructing a triangulation lexicographically minimizing the decreasing vector of slopes.

Remark. It would be interesting to find other optimality criteria for point sets with elevations, that are amenable to edge-insertion. However, we know that several natural measures, e.g., $\mu(xyz)$ equal to the maximum angle on the lifted triangle $\hat{x}\hat{y}\hat{z}$, do not satisfy either (I) or (II). A six-point counterexample can be formed with the vertices of a regular hexagon. There are two triangulations of the hexagon with an equilateral triangle in the middle; no single edge-insertion tranforms one into the other. By appropriately setting elevations, these two triangulations can be made local optima.

9. Conclusion

The main result of this paper is the formulation of the edge-insertion paradigm as a general method to compute optimal triangulations, and the identification of two classes of criteria for which the paradigm indeed finds the optimum. The paradigm is an abstraction of the algorithm introduced in [10] for computing minmax angle triangulations.

The algorithms of this paper have been implemented by Waupotitsch [11]. (The programs are currently available through anonymous ftp from the directory "/SGI/MinMaxer" at the site "ftp.ncsa.uiuc.edu.") The experience shows that the $O(n^2 \log n)$-time algorithm is fairly practical, also for large point sets. This is because its running time for most data sets is significantly less than the pessimistic worst-case prediction. This phenomenon, on the other hand, was not observed for the $O(n^3)$-time algorithm.

Though usually simple to verify, conditions (I) and (II) are somewhat restrictive. It would be interesting to find conditions weaker than (I) even though the price to pay may be implementations of the paradigm that take more than cubic time. Listings of optimality criteria can be found in [1], [2], [19], and [24]. Furthermore, implementations for criteria satisfying (I) and (II) that run in time $o(n^3)$ and $o(n^2 \log n)$ are sought.

Acknowledgment

The authors thank two anonymous referees for suggestions on improving the style of this paper.

References

1. R. E. Barnhill. Representation and approximation of surfaces. In *Math. Software III*, J. R. Rice, ed., Academic Press, New York, 1977, pp. 69–120.
2. M. W. Bern and D. Eppstein. Mesh generation and optimal triangulation. In *Computing in Euclidean Geometry*, D. Z. Du and F. K. Hwang, eds., World Scientific, Singapore, 1992, pp. 23–90.
3. K. Q. Brown. Voronoi diagrams from convex hulls. *Inform. Process. Lett.* **9** (1979), 223–228.
4. T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 1990.
5. E. F. D'Azevedo and R. B. Simpson. On optimal interpolation triangle incidences. *SIAM J. Sci. Statist. Comput.* **10** (1989), 1063–1075.

6. B. Delaunay. Sur la sphère vide. *Izv. Akad. Nauk SSSR, Otdel. Mat. Estestv. Nauk* **7** (1934), 793–800.
7. N. Dyn, D. Levin, and S. Rippa. Data dependent triangulations for piecewise linear interpolation. *IMA J. Numer. Anal.* **10** (1990), 137–154.
8. H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer-Verlag, Heidelberg, 1987.
9. H. Edelsbrunner and E. P. Mücke. Simulation of simplicity: a technique to cope with degenerate cases in geometric algorithms. *ACM Trans. Graphics* **9** (1990), 66–104.
10. H. Edelsbrunner, T. S. Tan, and R. Waupotitsch. An $O(n^2 \log n)$ time algorithm for the minmax angle triangulation. *SIAM J. Statist. Sci. Comput.* **13** (1992), 994–1008.
11. H. Edelsbrunner and R. Waupotitsch. Optimal two-dimensional triangulations. In *Animation of Geometric Algorithms: A Video Review*, M. Brown and J. Hershberger, eds. (Technical Report 87a), Systems Research Center, DEC, Palo Alto, CA, 1992, pp. 7–8.
12. S. Fortune. A sweepline algorithm for Voronoi diagrams. *Algorithmica* **2** (1987), 153–174.
13. J. A. George. Computer implementation of the finite element method. Ph.D. Thesis, Technical Report STAN-CS-71-208, Computer Science Dept., Stanford University, 1971.
14. L. J. Guibas, D. E. Knuth, and M. Sharir. Randomized incremental construction of Delaunay and Voronoi diagrams. *Algorithmica* **7** (1992), 381–413.
15. L. J. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Trans. Graphics* **4** (1985), 74–123.
16. G. T. Klincsek. Minimal triangulations of polygonal domains. *Ann. Discrete Math.* **9** (1980), 121–123.
17. C. L. Lawson. Generation of a triangular grid with applications to contour plotting. Technical Memo 299, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 1972.
18. C. L. Lawson. Software for C^1 surface interpolation. In *Math. Software III*, J. R. Rice, ed., Academic Press, New York, 1977, pp. 161–194.
19. D. A. Lindholm. Automatic triangular mesh generation on surfaces of polyhedra. *IEEE Trans. Magnetics* **19** (1983), 2539–2542.
20. E. L. Lloyd. On triangulations of a set of points in the plane. In *Proc. 18th Annual IEEE Symposium on the Foundations of Computer Science*, 1977, pp. 228–240.
21. F. P. Preparata and M. I. Shamos. *Computational Geometry—an Introduction*. Springer-Verlag, New York, 1985.
22. V. T. Rajan. Optimality of the Delaunay triangulation in $\Re^d$. In *Proc. 7th Annual Symposium on Computational Geometry*, 1991, pp. 357–363.
23. S. Rippa. Minimal roughness property of the Delaunay triangulation. *Comput. Aided Geom. Design* **7** (1990), 489–497.
24. L. L. Schumaker. Triangulation methods. In *Topics in Multivariate Approximation*, C. K. Chui, L. L. Schumaker, and F. L. Utreras, eds., Academic Press, New York, 1987, pp. 219–232.
25. M. I. Shamos and D. Hoey. Closest point problems. In *Proc. 16th Annual IEEE Symposium on the Foundations of Computer Science*, 1975, pp. 151–162.
26. R. Sibson. Locally equiangular triangulations. *Comput. J.* **21** (1978), 243–245.
27. G. Strang and G. Fix. *An Analysis of the Finite Element Method.* Prentice-Hall, Englewood Cliffs, NJ, 1973.
28. D. F. Watson and G. M. Philip. Systematic triangulations. *Comput. Vision Graphics Image Process.* **26** (1984), 217–223.

Received December 17, 1991, and in revised form September 21, 1992.