LA-UR- 98-3949

Approved for public release; distribution is unlimited.

CONF-9810139--

Title:

Strategies for Nonobtuse Boundary Delaunay Triangulations

RECEIVED
MAY 03 1999
OSTI

Author(s):

Murphy, Michael, EES-5

Gable, Carl W. EES-5

Submitted to:

7th International Meshing Roundtable

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

Los Alamos
NATIONAL LABORATORY

Strategies for Nonobtuse Boundary Delaunay Triangulations

Michael Murphy* Carl W. Gable†

Abstract. *Delaunay Triangulations with nonobtuse triangles at the boundaries satisfy a minimal requirement for Control Volume meshes. We motivate this quality requirement, discuss it in context with others that have been proposed, and give point placement strategies that generate the fewest or close to the fewest number of Steiner points needed to satisfy it for a particular problem instance. The advantage is that this strategy places a number of Steiner points proportional to the combinatorial size of the input rather than the local feature size, resulting in far fewer points in many cases.*

1 Introduction

Techniques for numerically approximating the solution to partial differential equations typically have at least three distinct phases: *mesh generation*, where the domain is partitioned into a finite number of pieces; *discretization*, which takes the mesh and derives a system of linear equations, $Ax = b$, whose solution can be used to obtain an approximation to a PDE over the domain; and the *solution* phase, where the system of linear equations is solved[1]. The requirements of one phase can place constraints on the output of another. A frequent requirement of the solution phase is to restrict the matrix A to a class of matrices so that linear solution techniques which exploit special properties of this class to obtain fast, accurate, and stable performance can be employed. To satisfy this requirement, the discretization phase, which generates the matrix, often must impose geometric restrictions on the mesh. In this paper, we study the requirements of the *Control Volume* discretization technique. Our goal is to obtain algorithms that satisfy such geometric restrictions in the mesh generation phase.

We first give a geometric interpretation of the Control Volume Method, which will help motivate the need for mesh quality at some minimal level. We then argue why satisfying these minimal quality requirements alone can in some real world settings be more useful than satisfying them in conjunction with stricter quality requirements that have been proposed in the literature. Finally, we present algorithms to guarantee the minimal level of mesh quality necessary to perform computations on Delaunay triangulations of polygonal domains by adding either the fewest or close to the fewest number of Steiner points (*i.e.*, nodes placed to obtain quality triangles) necessary for a particular problem instance; we also address the difficulties that arise in more complicated geometric settings such as Planar Straight Line Graphs (PSLGs) (see [BE92] for a definition) and Piecewise Linear Complexes (PLCs) (see [MTT+96] for a definition) in higher dimensions.

2 The Control Volume Method

2.1 Overview

The Control Volume Method, also known as the "Control Volume Finite Element Method,"[For91], "Box Method," [BR87]) "Finite Volume," and "Integrated Finite Differences,"using "PEBI grids"[HBMC91], is a discretization technique for Delaunay meshes[2]. Regardless of the dimension, it discretizes a Delaunay mesh with respect to its edges. For each edge $e = (i, j)$ in the mesh, the Control Volume method creates an

*Department of Computer Science, University of Maryland—College Park and Los Alamos National Laboratory. Email: murphy@lanl.gov

†Geoanalysis Group, Earth and Environmental Science Division, Los Alamos National Laboratory. Email: gable@lanl.gov

[1] This is a simplification. Indeed, many variations exist depending on the application. For example, for time dependent applications this can be an iterative process—the output from the solver may induce changes in the mesh forcing a new discretization and thus a new system of equations to be solved. However, for our purposes, this view is complete.

[2] Control Volume can also be used for non-Delaunay meshes, but that is beyond the scope of this paper.

entry in the ith row and jth column in the matrix A associated with system of linear equations $Ax = b$ approximating the PDE to be solved (and vice versa —the matrix A is symmetric). The entry, call it $A_{i,j}$ is non–zero (except when there are degeneracies arising from co–circularity) because it is associated with the length (area in three dimensions and volume in higher dimensions) of the Voronoi face, $V_{i,j}$, separating i and j. The entry $A_{i,j}$ is computed by dividing $V_{i,j}$ by the the length of edge e; however, it is intuitive for the purposes of understanding quality requirements to ignore the division by the length of e and focus on the quantity $V_{i,j}$. For an edge in a constrained or conforming triangulation in a planar domain, (which we will work with from now on unless otherwise stated) $V_{i,j}$ can be computed as follows. If e is an internal edge, then because T is a triangulation, there are exactly two triangles incident upon e. The length of the segment formed by connecting the circumcenters of these two triangles c_1, c_2 is $V_{i,j}$. If e is a boundary edge, then there is only one triangle incident upon e and the length of the Voronoi edge is infinite. Thus, the length is truncated at the domain boundary by setting $V_{i,j}$ equal to the length of the line segment connecting the circumcenter of the incident triangle to the midpoint of e. This is illustrated in Figure 1.

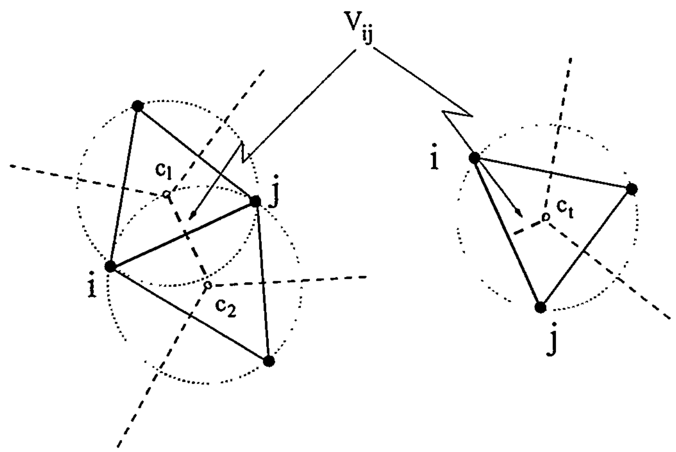

Figure 1: The entry $A_{i,j}$ in the matrix A corresponding to edge $e = (i, j)$ is associated with the length of the edge of the Voronoi face, $V_{i,j}$, shared by edge (i, j). When (i, j) is a boundary edge (right), this length is the length of the line segment with endpoints at the bisector of (i, j) and the circumcenter of the unique triangle incident upon (i, j).

2.2 Minimal Quality Requirements for the Control Volume Technique

Although it is well known that the Delaunay Triangulation is optimal with respect to many interesting criteria[Law77] [Raj94] [Mus97], not all Delaunay meshes are appropriate for the purpose of solving PDEs with this discretization technique[For91]. Indeed, the *quality* of a triangulation has been a major research topic in mesh generation, resulting in many definitions. This diversity stems in part from the variety of applications, differing levels of expectations about what can be obtained from a mesh, and requirements of competing discretization methods and solvers. It is not our intention to attempt the authoritative definition. However, by restricting our attention to the Control Volume setting, a minimal requirement for a Delaunay mesh to be suitable for numerical computation can be precisely stated. That is, from the solver's perspective, it is required that the matrix A in the associated linear system $Ax = b$ be an M–matrix. M–matrices are square matrices whose off-diagonal elements are either zero or negative and whose diagonal elements are strictly positive. M–matrices are also non-singular and have strictly positive inverses. As it turns out, the diagonal element $A_{i,i}$ in the control volume discretization is the sum of the absolute values of the other entries in row i, causing it to be diagonally dominant, implying (with some minor additional assumptions) the latter two criteria via standard theorems in numerical analysis. Consequently, we need only focus on obtaining negative off diagonal entries in the manner explained below. With that, the solution of the resulting system of linear equations using an iterative technique is possible, allowing a tremendous advantages in accuracy, stability, and running time over direct solvers; see [Saa96] for more background on M–matrices and iterative linear solution techniques.

The M–matrix restriction also avoids physically nonsensical results. To see why, consider a general model problem that could be solved using this discretization technique. The equation given by $F = -\nabla G$, where F is the flux, governs the diffusion of heat if G corresponds to temperature or fluid flow in a porous material if G corresponds to pressure. Energy (fluid) flows from high to low temperature (pressure) so the equation has a minus sign to account for heat (fluid) flowing opposite the gradient. When constructing the geometric coefficients for the discrete form of the equations, the minus sign is associated with the geometric terms. The requirement that all the geometric coefficients are negative insures there is no non-physical transport up gradient.

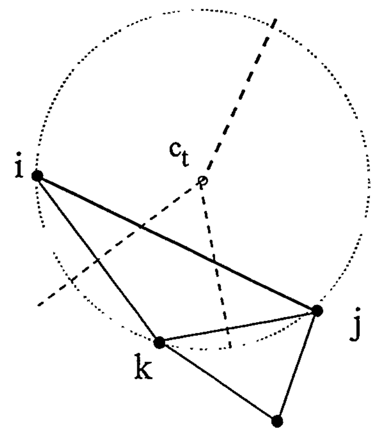

Figure 2: An obtuse boundary triangle in a Delaunay mesh. The edge (i, j) is a boundary. The circumcenter c_t of the triangle (i, j, k) falls outside the domain. Situations like this lead to geometrically nonsensical results and fail to give an M–matrix when using the Control Volume discretization scheme.

To translate the M–matrix requirement of the solver into a geometric restriction for the mesh generator, we must first point out a subtlety of the discretization technique. Since the non-diagonal elements of an M–matrix should be zero or negative, the length of the Voronoi segments associated with edges, used in the matrix entry computation, should be a signed quantity. When computing this signed length, it is necessary to use the orientation of the triangles incident upon it in the following manner. Suppose that edge $e = (i, j)$ is incident upon a triangle $t = (i, j, k)$ with circumcenter c_t. Then we compute the signed distance from c_t to the midpoint of e. If c_t is on the same side of e as k (the other vertex of the triangle), then this distance is negative, by convention. Otherwise, it is non-negative. For an external boundary edge, if the circumcenter is outside the domain then the length of the circumcenter to the bisector must be positive with this convention; an example is shown in Figure 2. However, this violates the M–matrix requirement. For an internal edge in a Delaunay mesh, the sum of the contributions of the two triangles incident upon it will always be non-positive, satisfying the M–matrix requirement, even if one of the circumcenters falls outside the domain. Nevertheless, circumcenters outside the domain in any case are undesirable[3]. Fortunately, as a consequence of the following straightforward theorem, to obtain an M–matrix with the Control Volume discretization we need only focus on triangles incident upon boundary edges.

Theorem 1 *In a Delaunay Triangulation, if the circumcenters of each triangle with an edge incident upon the boundary are contained inside the boundary, then the circumcenter of every triangle in the triangulation is contained inside the boundary.*

A proof is given in[CFH93].

We investigate algorithms that satisfy Theorem 1 by adding points to insure that each triangle incident upon a boundary edge is nonobtuse because a nonobtuse triangle contains its circumcenter.

[3]There are point sets where the circumcenter of every triangle in the Delaunay Triangulation of these sets falls outside the boundary. For example, a finite point set sampled from the "moment curve" in the plane parameterized by (t, t^2).

3 Related Work

Recently, quality requirements imposed by the Control Volume method have been discussed in the theoretical computer science community [MTTW95] (and followed up in [MTT⁺96] and [MTTW98]). Furthermore, the work of [BMR94] is motivated in part by the problem of generating quality meshes applicable to the Control Volume method. Finally, Rivara and Hitschfeld [HR97] recently have addressed a problem very similar to the one we consider. We review these works below.

3.1 The Radius–Edge Condition

One might ask why we are focusing upon avoiding a "show stopper" by requiring only that the mesh yield a discretization resulting in an M–matrix when algorithms have been given to satisfy loftier quality requirements that not only guarantee convergence but also improve the rate of convergence. Specifically, a recent result in quality Control Volume meshing [MTTW95] states that the Control Volume method will converge at a rate depending on the the largest ratio of the circumradius of each triangle to its shortest edge (abbreviated "radius–edge"). The smaller this ratio, the better. Shewchuk explores in his thesis [She97] how Ruppert's Delaunay Refinement algorithm [Rup93] can be used to achieve such a bounded ratio for every triangle in the triangulation.

To see why this quality measure can be too restrictive for some applications, it is worthwhile to examine some rather extreme yet real-world, applications that we face. One such application is modeling fluid flow in geological systems. A rock layer may extend for many kilometers yet may only be a few meters thick. Thus, high aspect ratios are the norm rather than an occasional nuisance. Another real world situation where very high aspect ratios arise is in semiconductor modeling when the ability to model the effect of thin layers of film placed in the semiconductors is desired. In these situations, more stringent quality triangulation requirements can produce unsatisfactory results because the number of triangles required to satisfy them is too large. For example, in two dimensions, the bounded radius–edge ratio quality measure requires that element size be on the order of the local feature size (see [Rup93] for a definition). However, any algorithm that outputs elements on the order of the local feature size will generate entirely too many elements in these high aspect ratio settings because the local feature size is small everywhere[4].

In contrast, the quality requirements we impose, although far less stringent, require far fewer Steiner points to satisfy them. Indeed, we can obtain "combinatorial results" — those that show that the number of Steiner points depends on the combinatorial size of the initial input (*i.e.*, the number of points and line segments), rather than local feature size, a geometric measure.

3.2 Nonobtuse Triangulation of Polygonal Domains

The work of Bern, Mitchell, and Ruppert [BMR94], which shows how to triangulate a polygonal domain with nonobtuse triangles using $O(n)$ Steiner points (where n is the number of vertices in the polygon) is a prime example of a combinatorial result applicable to Control Volume meshing. Nonobtuse triangles are appropriate for the Control Volume method because they imply that the Voronoi face associated with an edge e, crosses e. They are also interesting for general interpolation problems because they imply that for any point inside a triangle, its nearest neighbor in the set of all mesh vertices to that point is one of the triangle vertices. Meshes with bounded radius–edge ratios, or any triangulation with no small angles, do not have these properties unless the guarantee says that no angle will be less than 45 degrees or the bounded edge ratio is $\leq \sqrt{2}$; guarantees at this level are beyond the current state of the art.

There are some differences in our approaches. As we only require the boundary elements to be nonobtuse, we can use far fewer Steiner points. Bern, Mitchell, and Ruppert report that in practice, their algorithm generates roughly $25n$ Steiner points per instance. In contrast, our algorithms generate the fewest or close to the fewest number of Steiner points needed *for a particular problem instance.* In the worst case, this can be bounded at $3n$, for the reasons that one can never pack more than three obtuse angles around a vertex and that it is possible to place a Steiner point to resolve an obtuse angle without creating a new obtuse angle in the Delaunay Triangulation of the point set. In situations where we are not as concerned about the

[4] It is also worthwhile to note that these high aspect ratio problems tend to frustrate many mesh generation heuristics as well.

quality of interior triangles, or when we wish to have no interior points at all, our approach can be more useful.

3.3 Nonobtuse Boundary Triangulations

Hitschfeld and Rivara[HR97][HR98], have recently addressed a very similar problem. They note that no–small–angle quality Delaunay triangulation algorithms which guarantee that all angles are larger that 30 degrees, can allow obtuse boundary angles. They present an algorithm to be used as a "post–processing step" after such a quality triangulation algorithm on a polygonal domain is used to remove the obtuse boundary triangles; the postprocessing algorithm makes the explicit assumption that all angles in the triangulation not defined by two boundary edges are between 30 and 120 degrees and that the triangulation is constrained Delaunay. We simplify this algorithm so that it places fewer Steiner points and does not make any assumptions about the angles in the input triangulation. However, the price of our simplification is that a no–small–angle triangulation is not maintained. We dub this algorithm **Project–Flip**. Hints of a similar algorithm can be found in[For91].

PROJECT–FLIP

input : A Constrained or Conforming Delaunay Triangulation of a Polygonal Domain.
while there exists an obtuse triangle $t = (a, b, c)$ with boundary edge $e = (a, b)$ opposite the obtuse angle **do**
    Let d be the orthogonal projection of c onto e into the triangulation.
    Remove t from the triangulation
    Create two new triangles (a, c, d) and (b, c, d)
    Restore Delaunay Triangulation via Flip algorithm[Law77]
end-while

Project-Flip has the following properties:

- It terminates and produces triangles so that the circumcenters of each triangle are contained in the boundary.
- The number of points inserted can be bounded by $O(n)$.
- While it is useful for constrained or conforming Delaunay Triangulations of polygons, it cannot be generalized to arbitrary PSLGs.

The Project-Flip algorithm is not optimal in the sense that it will not place the minimum number of points in a particular instance. This will be shown below. Consequently, we wish to explore methods which are sensitive to the problem instance.

4 Nonobtuse Boundary Triangles with Few Steiner Points

4.1 Nonobtuse Triangles on a Boundary Edge

Let us first consider the following related problem. Suppose we are given a line segment in the plane denoted by edge $e = [a, b]$, $a, b, \in R^2$, and a set of points S all contained in a semicircle of the diametrical circle of e (*i.e.* the circle with diameter e). We wish to refine e into $k+1$ into non-overlapping subintervals $[a, s_1), \ldots, [s_{k-1}, s_k), [s_k, b]$, by placing Steiner points $S_e = \{s_1, \ldots, s_k\}$ along e such that the open diametrical circle of each subinterval is empty in the sense that it contains no point of S in its interior. This would guarantee that no circumcenters in the Delaunay Triangulation of the point set $S \cup \{a, b\} \cup S_e$ would fall on the other side of e. Equivalently, all the angles opposite the subintervals in the Delaunay Triangulation would be nonobtuse. Ideally, the number of Steiner points that we choose, k, should be the fewest number of Steiner points needed for a particular instance.

One trivial algorithm takes the orthogonal projection of all points inside e's diametrical circle onto e and uses these as the Steiner points that induce the subintervals. However, this can add too many points because each time we add a Steiner point to refine e, the area covered by the diametrical circles on each subsegment of e

decreases. Thus, some points that were initially contained in e's diametrical circle may no longer reside inside the diametrical circles of the newly created subintervals and therefore can be omitted from consideration. Indeed, the addition of one Steiner point may be all that is necessary. Although worst case optimal, it is not sensitive to the size of the output required for a particular instance and is thus unsatisfactory.

The Project-Flip algorithm could also be applied to this problem. By first taking the Delaunay Triangulation of the point set (in which e would be a boundary edge) the algorithm would terminate with no obtuse angles. In some cases, it would place far fewer points on e than the straightforward orthogonal projection algorithm described above. However, as shown in Figure 3, it does not place the fewest Steiner points for a particular instance. Moreover, there exists similar examples, identified by Mount[Mou97], which shows that the running time can be $O(n^2)$.

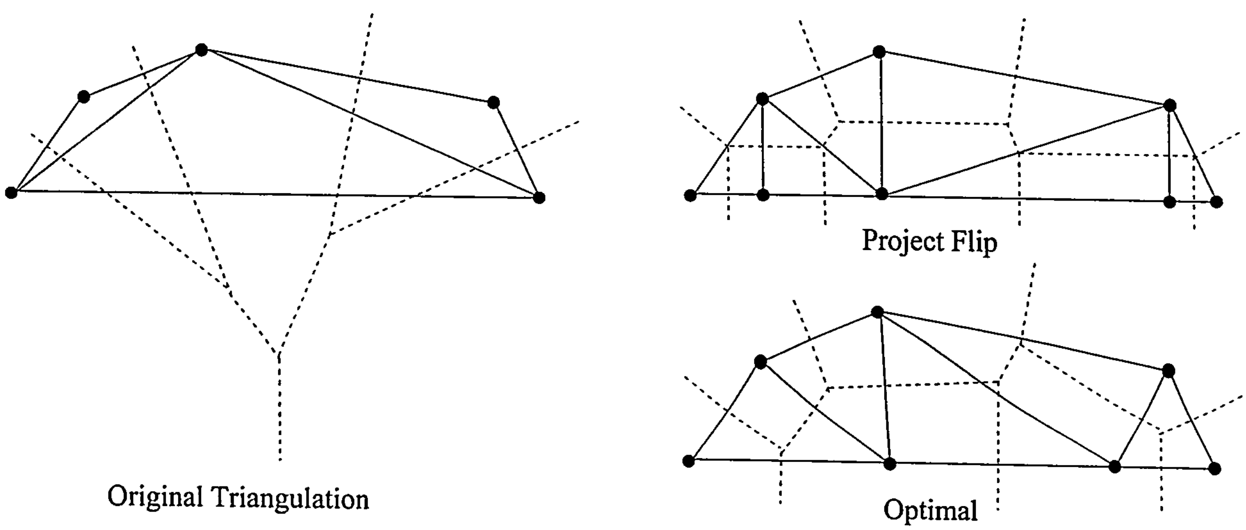

Figure 3: Project-Flip, which in this case has the same output as simple orthogonal projection of all points inside e's diametrical circle onto e, can be suboptimal in terms of the number of Steiner points added to obtain nonobtuse boundary triangles.

It turns out that a simple greedy strategy, illustrated in Figure 4, gives an optimal solution to this easier problem.

Greedy–Refine(e,S)

input : An edge $e = [a, b]$ and a set of points S all contained in a semicircle of the diametrical circle of e.
Starting at a find the largest subsegment of e, $e' = [a, a')$ such that the open diametrical disk of e' contains no point of S
if $a' \neq b$ **then**
    Refine e by placing a Steiner point at a'.
    Let S' be the points in the diametrical circle of $[a', b]$
    Greedy–Refine ($[a', b]$,S');
end-if

Theorem 2 *The greedy algorithm for an edge generates the fewest number of Steiner points required for a particular instance.*

Proof: The greedy algorithm's optimality can be proven by induction on the minimum number of Steiner points k needed for a particular interval. That is, suppose we are given an edge $e = [a, b]$ and a set S such that k Steiner points are required to refine e so that it satisfies the empty diametrical circle criterion. Obviously, $k \leq n$, for as noted, orthogonally projecting each point in S onto e would do the job. The base case, $k = 0$ is trivial.

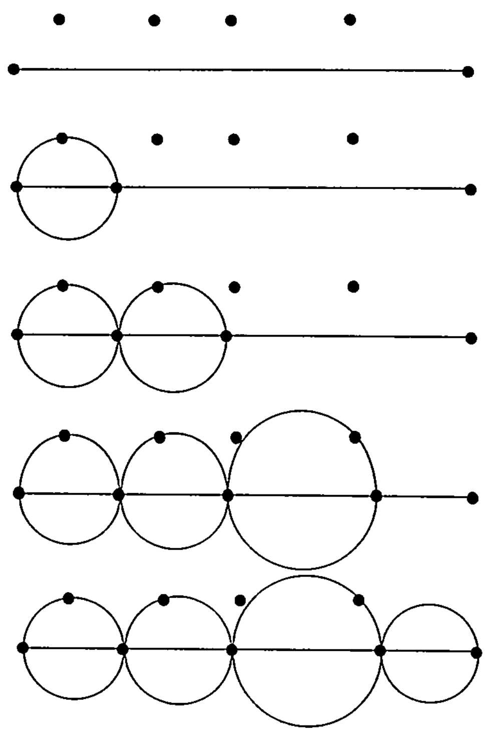

Figure 4: Execution of the greedy algorithm for a line segment, e. It starts at an endpoint and grabs the largest empty circle it can get.

Without loss of generality, transform the plane so that the edge (a, b) coincides with the interval [0,1] along the x-axis. Among all optimal Steiner placements achieving k points, consider the one whose leftmost Steiner point is as far right as possible (*i.e.*, has the largest x-coordinate). More formally let x_0 be the *supremum* of the set of initial optimal Steiner placements. Let x_g be the leftmost endpoint of the greedy triangulation. We claim that $x_g = x_0$. Suppose not. If $x_g < x_0$, then the closed diametrical circle bounded at x_g is interior to the circle bounded at x_0 and hence would be further expanded by the algorithm, a contradiction. On the other hand, if $x_g > x_0$, then we can modify the optimum point placement strategy so that it uses the same number of Steiner points. This is done by shrinking the second optimum disk by moving its leftmost point to x_g. The modified second disk is contained in the original second disk and consequently remains empty. This contradicts the hypothesis on x_0.

4.2 Convex Polygon

We now consider generalizing our problem to a slightly more complex situation. Given a convex polygon P, we wish to decompose each polygonal edge into subsegments that satisfy the point–free diametrical circle property. This problem cannot be solved by applying the above greedy algorithm for edges to each polygonal edge individually, using the vertices of the P inside the diametrical circle as the set of points. This is because *interference* can arise. It is possible to deem an edge's diametrical circle point–free only later to violate the diametrical circle when adding a Steiner point on another edge. This leads to conflicts and suboptimality, illustrated in Figure 5.

A simple modification to the greedy algorithm places the fewest number of Steiner points. For an edge e that has a nonempty diametrical circle, we need to be sensitive to the regions on e in which adding a Steiner point will create problems for other edges. That is, those regions where the diametrical disks of other edges intersect e. We call these the "forbidden intervals" of e. The idea behind this "sensitive greedy strategy" is to start at an endpoint and find the largest segment whose diametrical circle is empty, with the added constraint that an endpoint is never placed inside a forbidden interval. Specifically, if the greedy strategy

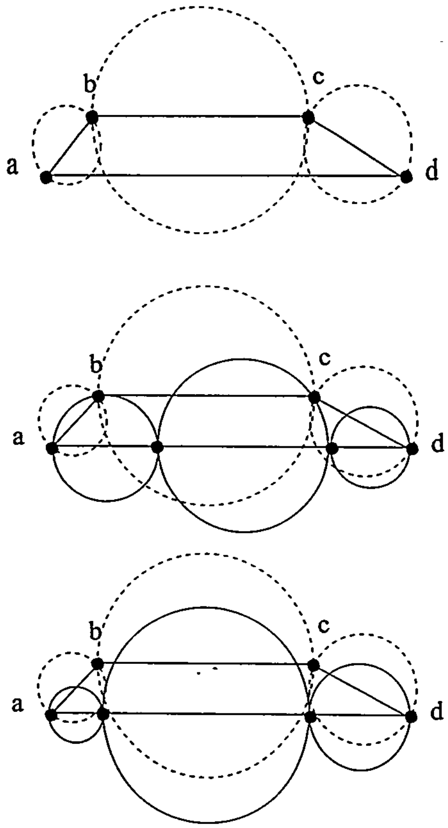

Figure 5: In the top figure edges (a,b), (b,c) and (c,d) have point–free diametrical circles and edge (a,d) is in need of refinement. The middle figure illustrates that the greedy algorithm for edges is deficient in the sense that it can add Steiner points that violate previously point–free diametrical circles. The solution is to never add a Steiner point that violates a diametrical circle of another edge, regardless of whether that diametrical circle is empty or not.

places a point inside a forbidden interval, then place the Steiner point at the start of the forbidden interval.

```
SENSITIVE–GREEDY–REFINE(P)
input : A convex polygon P
Let E be the list of edges of P without point–free diametrical circle property w.r.t. the vertices of P
while E is not empty do
        Let e = [a, b] be an edge in E
        Remove e from E
        Starting at a find the largest subsegment of e, e' = [a, a') such that:
                        The open diametrical disk of e' contains no point of S and
                        a' is not contained in the open diametrical disk of another edge f.
        Refine e by placing a Steiner point at a'.
        if the diametrical circle of [a', b] is not point–free then
           Add subsegment [a', b] to E
        end-if
end-while
```

Theorem 3 *For a convex polygon, the sensitive greedy strategy places the fewest Steiner points to obtain nonobtuse boundary triangles.*

Before proving this, we need the following two lemmas:

Lemma 1 *The diametrical circles of the forbidden regions on an edge are point-free.*

The proof follows immediately from the convexity of the polygon.

Lemma 2 *The forbidden regions on an edge do not overlap.*

Proof Sketch: Let f and g be two edges on a convex polygon P whose open diametrical circles overlap. We consider the lune, the overlapping region, of the two open circles. If f and g share a unique common vertex, then one apex of the lune is at this common vertex. We now analyze the location of the other apex and its relation to e. Let h be the segment connecting the two vertices not common to f and g. Notice that due to convexity, h must either be e or be contained in the halfspace of e containing P. The other apex is on the orthogonal projection of the common vertex onto the line containing h. Consequently, it is either contained in the same halfspace of e as P, if it falls onto h or it is outside the domain if the orthogonal projection does not fall onto h because f and g are boundary edges. In the former case, since we are considering open diametrical circles, this single point is the gap. In the latter case, it implies by convexity that e cannot intersect the lune at all. In case f and g do not share a common vertex, then it is possible to show that the lune is contained in halfspaces of the lines containing the two segments connecting e and f to form a convex quadrilateral. Since both of these segments must be contained in a halfspace of e, it implies that e cannot intersect the lune at all. □

Proof (Of Theorem 3): Let e be an edge which is under refinement. Let f be an edge whose diametrical circle intersects e. Two cases can arise: (1) the diametrical circle of f is empty and (2) the diametrical circle of f is non-empty.

In case (1), placing a point in this forbidden interval will violate f, since f's diametrical circle is empty. Would one benefit from refining in the forbidden region of f with respect to e? The answer is no. Placing a point in the forbidden region of e with respect to f will necessarily induce a Steiner point on f to cope with the one we just added. However, we could spend two Steiner points (one from refining e and the one placed on f to cope with the one placed on e) at least as well, if not better, by just refining e and leaving f alone. Two Steiner points placed on e are enough to skip over the forbidden region of e with respect to f and create an empty diametrical circle on e: place one Steiner point at the beginning of the forbidden region of f and the other as far as possible along e so that it does not violate another forbidden region and creates a subsegment with an empty diametrical circle. By Lemma 1, we know that the second point will be beyond the forbidden region of f with respect to e and consequently will leave f alone.

Again in case (2), one can never benefit from refining in a forbidden region. This is because if f's diametrical circle is not empty and intersects e, then one of the endpoints of e is one of the endpoints of the forbidden region of e with respect of f as well. Since by Lemma 1, the diametrical circle of this forbidden region is empty, we can simply place a Steiner point at the start of the forbidden region. □

4.3 Set of Points

Given a set of points S in the plane, we wish to refine each edge of the convex hull of S so that the Delaunay triangulation of S taken with the augmented points has no obtuse angles opposite the boundaries. One interpretation of this criterion is that the nearest neighbor in S and the augmented points to a point outside the convex hull should be one of the vertices of the convex hull edge closest to that point. To achieve this, we divide the convex hull edges into two groups: those that have empty diametrical circles and those that do not. The strategy is never to violate an empty diametrical circle of another edge when refining an edge, as in the case of the convex polygon. However, the strategy changes when dealing with a non-empty diametrical circle of another edge. The answer is to refine only if it greedy strategy gets snagged on a point internal to the convex hull. If it gets snagged on a boundary point, the strategy is to refine the other violated edge first. This insures that a Steiner point is never needed to be placed to resolve an obtuse angle around a previously added Steiner point. We do not have an optimality proof at this time, although we can be sure that in many situations, this will place fewer Steiner points than Project-Flip.

4.4 Non–convex polygonal domains

The above algorithms can be generalized to non–convex polygons, with or without interior points, by changing the notion of forbidden regions slightly to take into account visiblity. This allows us to obtain constrained Delaunay Triangulations with nonobtuse boundary triangles, but not conforming ones.

5 The Difficulty of More Complicated Domains

5.1 Planar Straight Line Graph

The problem of nonobtuse boundary triangulations is significantly more difficult on a PSLG, which allows for much more complicated input geometries with multimaterial interfaces. The problem is to ensure that for each edge on a given PSLG, its diametrical circle is empty. As a solution to this problem implies the solution to the problem of conforming Delaunay Triangulation, the lower bound of $\Omega(n^2)$ Steiner points given in [ET93] applies. However, it is not even known if the number of Steiner points required to satisfy the edges of a PSLG can be bounded by a polynomial function of the input size. Work with a similar spirit can also be found in the study of triangulations that have no large angles[Mit93][Tan96]. The goal in these studies is to refine a triangulation using a number of Steiner points as close to $\Omega(n^2)$ as possible so that the largest angle in the triangulation is bound by some constant. Of course, the closer to nonobtuse this angle is, the better. We suggest relaxing the problem so that the angles opposite boundary edges are the only ones taken into consideration. However, many of the difficulties faced in these problems still apply. The salient problem is that the interference caused by refining one edge can have global reprocussions. Mitchell's paper[Mit93] contains some very interesting examples of the global problems encountered.

Recently, Rivara and Hitschfeld [HR98] have announced that they can solve this problem on constrained Delaunay triangulated PSLGs if all angles not constrained by two boundary edges are bounded between 30 and 120 degrees. We will refer to these as "quality" PSLGs. However, their technique, which appears very practical, falls prey to some "worst case" scenarios that invalidate their general claims about its performance. Specifically, Proposition 3 (p. 18) which states that a such a quality PSLG with k obtuse boundary triangles needs only $O(k)$ Steiner points to make the PSLG satisfy the nonobtuse boundary property. We present figure 6 as evidence that this is incorrect. In this counterexample, one bad triangle can induce as many as $\Omega(n)$ Steiner points to satisfy the nonobtuse boundary Delaunay property, where n is the total number of triangles. Furthermore, it is not even clear if their algorithm will terminate on all such quality PSLGs. This illustrates the insidious nature of the the global interference problem, and the resulting propogation of Steiner points, which makes this problem so difficult in general.

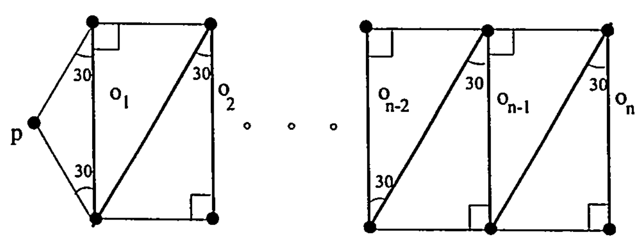

Figure 6: A PSLG satisfying all the angle critera assumed by Rivara and Hitschfeld's method [HR98] that contradicts their Proposition 3 (p. 18). The leftmost triangle has a 120 degree angle at vertex p and is the only obtuse triangle in the triangulation of the PSLG, whose edges are highlighted. Refining edge o_1 to remove the obtuse angle will necessarily induce a total of $\Omega(n)$ Steiner points, one on each of the highlighted edges.

5.2 M–Matrices in Three Dimensions

As is the case in the plane, not all three dimensional Delaunay Triangulations will yield a M–matrix under the Control Volume Method[Let92]. The problem of obtaining an M–matrix in three dimensions can be geometrically formulated most generally as follows: Given a PLC, ensure that the diametrical sphere of each object contains no other vertex. For a line segment in three dimensions, not part of any facet, the diametrical sphere is simply the sphere whose diameter is the line segment. For a triangular facet, the diametrical sphere is the minimum–diameter sphere containing that triangular facet. This can be obtained by computing the circumcircle of the triangular facet and expanding it into a sphere with the same radius. Using the three–dimensional version of Theorem 1 (also proven in [CFH93]), facets refined to satisfy these conditions will result in an M–matrix using the Control Volume discretization technique in three dimensions. Unfortunately, this implies a solution to Conforming Delaunay Triangulation, an "easier" problem for which a general point placement strategy which terminates correctly for PLCs in dimensions higher than two is not known to exist.

Acknowledgements

The first author thanks Professor David Mount for many useful discussions.

References

[BE92] M. Bern and D. Eppstein. Mesh generation and optimal triangulation. In D.-Z. Du and F. K. Hwang, editors, *Computing in Euclidean Geometry*, volume 1 of *Lecture Notes Series on Computing*, pages 23–90. World Scientific, Singapore, 1992.

[BMR94] M. Bern, S. Mitchell, and J. Ruppert. Linear-size nonobtuse triangulation of polygons. In *Proc. 10th Annu. ACM Sympos. Comput. Geom.*, pages 221–230, 1994.

[BR87] R. Bank and D. Rose. Some error estimates for the box method. *SIAM J. Numer. Analysis*, 24:777–787, 1987.

[CFH93] P. Conti, W. Fichtner, and N. Hitschfeld. Mixed elements trees: A generalization of modified octrees for the generation of meshes for the simulation of complex 3-d semiconductor devices. *IEEE Trans. on CAD/ICAS*, 12(11):1714–1725, November 1993.

[ET93] H. Edelsbrunner and T. S. Tan. An upper bound for conforming Delaunay triangulations. *Discrete Comput. Geom.*, 10(2):197–213, 1993.

[For91] P.A. Forsyth. A control volume finite element approach to NAPL groundwater contamination. *SIAM J. Sci. Stat. Comput.*, 12:1029–1057, 1991.

[HBMC91] Z.E. Heinemann, C.W. Brand, M. Munka, and Y.M. Chen. Modeling reservoir geometry with irregular grids. *SPE Reservoir Engineering*, pages 225–232, 1991.

[HR97] N. Hitschfeld and M. C. Rivara. Non-obtuse boundary Delaunay triangulations. In *6th Intl. Meshing Roundtable*, 1997. http://www.andrew.cmu.edu/user/sowen/imr6.html.

[HR98] N. Hitschfeld and M. C. Rivara. Automatic creation of non-obtuse boundary Delaunay triangulations. Technical Report TR DCC-98-2, Dept. Comput. Sci., University of Chile, Santiago, Chile, 1998. http://www.dcc.uchile.cl/~nancy/.

[Law77] C. L. Lawson. Software for C^1 surface interpolation. In J. R. Rice, editor, *Math. Software III*, pages 161–194, New York, NY, 1977. Academic Press.

[Let92] F. Letniowski. Three-dimensional Delaunay triangulation for finite element approximations to a second-order diffusion operator. *SIAM J. Sci. Stat. Comput.*, 13:765–770, 1992.

[Mit93] Scott A. Mitchell. Refining a triangulation of a planar straight-line graph to eliminate large angles. In *Proc. 34th Annu. IEEE Sympos. Found. Comput. Sci. (FOCS 93)*, pages 583–591, 1993.

[Mou97] D.M. Mount, 1997. Personal Communications.

[MTT+96] G.L. Miller, D. Talmor, S.-H. Teng, N. Walkington, and H. Wang. Control volume meshes using sphere packing: Generation, refinement, and coarsening. In *Proc. 5th International Meshing Roundtable*, Albuquerque, NM, 1996. Sandia National Laboratories.

[MTTW95] G. L. Miller, D. Talmor, S.-H. Teng, and N. Walkington. A Delaunay based numerical method for three dimensions: generation, formulation, and partition. In *Proc. 27th Annu. ACM Sympos. Theory Comput.*, pages 683–692, 1995.

[MTTW98] G.L. Miller, D. Talmor, S.-H. Teng, and N. Walkington. On the radius edge condition in the control volume method. *SIAM J. Numer. Analysis*, 1998. Submitted for publication. Preprint available at http://www.cs.cmu.edu/afs/cs.cmu.edu/user/tdafna/www/publications.html.

[Mus97] O.R. Musin. Properties of the Delaunay triangulation. In *Proc. 13th Annu. ACM Sympos. Comput. Geom.*, pages 424–426, 1997.

[Raj94] V.T. Rajan. Optimality of the Delaunay triangulation in R^d. *Disc. and Comput. Geom.*, 12:189–202, 1994.

[Rup93] J. Ruppert. A new and simple algorithm for quality 2-dimensional mesh generation. In *Proc. 4th ACM-SIAM Sympos. Discrete Algorithms*, pages 83–92, 1993.

[Saa96] Y. Saad. *Iterative Methods for Sparse Linear Systems.* PWS Publishing Company, Boston,MA, 1996.

[She97] J. R. Shewchuk. *Delaunay refinement mesh generation.* PhD thesis, School of Computer Science, Carnegie Mellon University, 1997. Available at http://www.cs.cmu.edu/~jrs/jrspapers.html.

[Tan96] T.-S. Tan. An optimal bound for high quality conforming triangulations. *Discrete Comput. Geom.*, 15:169–193, 1996.