Sink-insertion for Mesh Improvement *

Herbert Edelsbrunner[†] and Damrong Guoy[‡]

Abstract

We propose sink-insertion as a new technique to improve the mesh quality of Delaunay triangulations. We compare it with the conventional circumcenter-insertion technique under three scheduling regimes: incremental, in blocks, and in parallel. Justification for sink-insertion is given in terms of mesh quality, numerical robustness, and ease of parallelization.

Keywords. Mesh generation, Delaunay triangulations, dynamic data structures, mesh quality, experimentation, parallel meshing.

1 Introduction

This paper uses Delaunay triangulations to mesh geometric domains and studies the use of sinks in the iterative improvement of the mesh quality. As will be explained later, sinks are circumcenters of special Delaunay simplices. The main claim of this paper is that adding sinks is more effective than adding general circumcenters. Although Delaunay triangulations and sinks can be defined in arbitrary fixed dimensions, this paper focuses on the three-dimensional case.

Motivation. Physical simulation is often based on a decomposition of the spatial domain into simple elements. The collection of these elements is referred to as a mesh of the domain, and its usual purpose is to aid the numerical solution of differential equations defined over the domain. In most applications the mesh is assumed to have the face-to-face property, which means that any two elements are either disjoint or meet along a common face.

A special but popular type of mesh is the Delaunay triangulation. Given a set of points or vertices, it consists of all simplices spanned by the points that have an empty circumscribing sphere [3]. For points in general position in $\mathbb{R}^2$ the Delaunay triangulation consists of triangles and their shared edges and vertices. For points in general position in $\mathbb{R}^3$ it consists of tetrahedra and their shared triangles, edges, and vertices. The quality of the triangles and tetrahedra is usually measured in terms of angles and/or aspect-ratios. The quality depends solely on how the points that define the Delaunay triangulation are distributed. We obtain good quality either by judiciously choosing the points or by modifying the point set after constructing the initial Delaunay triangulation. The first approach seems difficult, but there was considerable success in following the second approach, which is commonly referred to as *Delaunay refinement.* Chew [2] and Ruppert [8] formulated Delaunay refinement algorithms in $\mathbb{R}^2$ that add circumcenters of triangles and achieve guaranteed lower and upper bounds on the size of angles. Dey et al. [4] and Shewchuck [9] extend these results to $\mathbb{R}^3$ by adding circumcenters of tetrahedra.

Results. In this paper we define the concept of sinks, which are special circumcenters of Delaunay tetrahedra, and we propose to substitute sinks for general circumcenters in the insertion process. We describe algorithms that improve mesh quality through sink-insertion, and we present experimental evidence that sink-insertion has advantages over circumcircle-insertion in the context of the Delaunay refinement algorithm.

We show that sink-insertion creates about the same mesh quality as circumcenter-insertion, but it does this in a more economical manner. One of the reasons is that Delaunay triangles and tetrahedra with small or large angles tend to cluster and share sinks. Instead of dealing with a large number of circumcenters we can therefore work with a small number of sinks. Another

*Research by both authors is partially supported by NSF under grants CCR-97-12088 and DMS 98-73945. Research of the first author is also supported by NSF under grants EIA-99-72879 and CCR-00-86013 and by ARO under grant DAAG55-98-1-0177.

[†]Department of Computer Science, Duke University, Durham, North Carolina 27708, and Raindrop Geomagic, Research Triangle Park, North Carolina 27709.

[‡]Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801.

advantage of sinks is that their definition in terms of the data is generally well-conditioned while circumcenters are sometimes ill-conditioned and pose numerical problems.

Outline. Section 2 introduces Delaunay triangulations. Section 3 uses Morse theoretic ideas to derive structure among Delaunay simplices. Section 4 studies the combinatorial properties of sink-insertion. Section 5 presents experimental results. Section 6 concludes the paper.

2 Delaunay Triangulations

This section introduces Delaunay triangulations as duals of Voronoi diagrams. We begin with some general terminology on simplicial complexes, which we borrow from combinatorial topology [1].

Simplicial complexes. A *simplex* is the convex hull of an affinely independent collection of points, $\sigma = \operatorname{conv} S$. Its *dimension* is one less than the number of points, $\dim \sigma = \operatorname{card} S - 1$. In $\mathbb{R}^3$ the maximum number of affinely independent points is four, so we have non-empty simplices of dimensions 0, 1, 2, and 3 referred to as *vertices*, *edges*, *triangles*, and *tetrahedra*. For every subset $T \subseteq S$ the simplex $\tau = \operatorname{conv} T$ is a *face* of σ and we write $\tau \leq \sigma$. Symmetrically, σ is a *coface* of τ. The simplex τ is a *proper face* of σ and σ is a *proper coface* of τ if T is a proper subset of S. The *interior* of σ is the set of points that belong to σ but not to any of its proper faces, $\operatorname{int} \sigma = \sigma - \bigcup \tau$.

A *simplicial complex* K is a finite set of simplices that is closed under taking faces and has no improper intersections between its simplices. More formally, $\sigma \in K$ and $\tau \leq \sigma$ implies $\tau \in K$, and $\sigma, \sigma' \in K$ implies $\sigma \cap \sigma' \leq \sigma, \sigma'$. The second condition allows for the case in which σ and σ' are disjoint because the empty set is considered to be the unique (-1)-dimensional simplex that is a face of every simplex. A *subcomplex* of K is a subset that is also a simplicial complex. The *underlying space* of a set of simplices $L \subseteq K$ is the union of interiors, $|L| = \bigcup_{\sigma \in L} \operatorname{int} \sigma$. $|L|$ is a topologically closed set iff L is a simplicial complex. The *closure* of L is the smallest simplicial complex that contains L, $\operatorname{Cl} L = \{\tau \leq \sigma \mid \sigma \in L\}$. The *star* of a vertex $p \in K$ is the collection of cofaces, $\operatorname{St} p = \{\sigma \in K \mid p \leq \sigma\}$. The *link* of p is the collection of faces of simplices in the star that do not belong to the star, $\operatorname{Lk} p = \operatorname{Cl} \operatorname{St} p - \operatorname{St} p$. The closed star and the link are simplicial complexes while the star is usually not closed.

Voronoi polyhedra. Let S be a finite set of points in $\mathbb{R}^3$. The *Voronoi region* of a point $a \in S$ is the set of points no further from a than from any other point in S, $V_a = \{x \in \mathbb{R}^3 \mid \|x - a\| \leq \|x - b\|, b \in S\}$. Here closeness is measured using the Euclidean distance. The Voronoi region of a is thus a three-dimensional convex polyhedron, which may or may not reach to infinity. Voronoi regions have disjoint interiors but may overlap along common portions of the boundary. A *Voronoi polyhedron* is the common intersection of Voronoi regions, $V_A = \bigcap_{a \in A} V_a$. It is a convex polyhedron of dimension three or less. Another Voronoi polyhedron V_B is a *face* of V_A if $V_B \subseteq V_A$, and it is a *proper face* if furthermore $V_B \neq V_A$. The *interior* of a Voronoi polyhedron is the set of points that belong to the polyhedron but not to any of its proper faces, $\operatorname{int} V_A = V_A - \bigcup V_B$. The *Voronoi diagram* is the set of Voronoi polyhedra, $\operatorname{Vor} S = \{V_A \mid A \subseteq S\}$. It is convenient but not necessary to assume non-degeneracy, which means that

(1) no four points lie in a common plane,

(2) no five points lie on a common sphere, and

(3) no three points lie on a great-circle of a sphere that passes through four points.

This assumption can be justified computationally using simulated perturbation as described in [5]. Assumption (1) implies that as long as there are at least four points, each Voronoi polyhedron has at least one vertex. Assumption (2) implies that if $V_A \neq \emptyset$ then $V_B \neq V_A$ for every proper superset B of A. Assumption (3) has more subtle consequences related to the concept of sinks as discussed in Section 3.

Delaunay simplices. The *Delaunay triangulation* of S is the set of simplices dual to Voronoi polyhedra. It is well-defined if S satisfies assumption (2). In this case every Voronoi polyhedron V_A has dimension $\dim V_A = 4 - \operatorname{card} A$. The *dual Delaunay simplex* is $\tau = \operatorname{conv} A$ whose dimension is $\dim \tau = \operatorname{card} A - 1 = 3 - \dim V_A$. In words, (three-dimensional) Voronoi polyhedra correspond to Delaunay vertices, Voronoi polygons correspond to Delaunay edges, Voronoi edges correspond to Delaunay triangles, and Voronoi vertices correspond to Delaunay tetrahedra. The face relationship in the Delaunay triangulation is the precise reverse of that in the Voronoi diagram, namely V_B is a face of V_A iff $\tau = \operatorname{conv} A$ is a face of $\sigma = \operatorname{conv} B$.

Let B be a set of four points in S. Because of assumptions (1) and (2) there is a unique sphere that passes through the four points and every other point lies either inside or outside but not on the sphere. We call the sphere *empty* if no point of S lies inside. If $\sigma = \operatorname{conv} B$ is a Delaunay tetrahedron then the circumscribing sphere is centered at the Voronoi vertex $x = V_B$ and it is empty because $\|x - a\| = \|x - b\| < \|x - c\|$ for all $a, b \in B$ and $c \in S - B$. The reverse is also true.

This implies that τ is a Delaunay simplex iff there is an empty sphere that passes through the vertices of τ and contains no other points of S. We call the circumsphere of a Delaunay tetrahedron a *Delaunay sphere* and the closed ball bounded by that sphere a *Delaunay ball*.

3 Flow and Relation

This section introduces sinks as the circumcenters of tetrahedra without successors in a relation motivated by the construction of a continuous vector field. We begin by defining two square distance functions.

Square distances. For a point $a \in \mathbb{R}^3$ consider the square distance function that maps every point $x \in \mathbb{R}^3$ to $\pi_a(x) = \|x-a\|^2$. Given a finite set $S \subseteq \mathbb{R}^3$, let $\pi_S : \mathbb{R}^3 \to \mathbb{R}$ be the pointwise minimum of the square distance functions,

$$\pi_S(x) \quad = \quad \min\{\pi_a(x) \mid a \in S\}.$$

The function $\pi = \pi_S$ is piecewise quadratic, and the pieces decompose $\mathbb{R}^3$ into the Voronoi regions of S.

Consider now a sphere (z, r) with center $z \in \mathbb{R}^3$ and radius $r \in \mathbb{R}$. The *weighted square distance* of a point $y \in \mathbb{R}^3$ from (z, r) is $\varphi_z(y) = \|y-z\|^2 - r^2$. For zero radius the weighted square distance is equal to the square distance from the center. Given S let Z be the collection of Delaunay spheres and $\varphi_S : \mathbb{R}^3 \to \mathbb{R}$ the pointwise minimum of the weighted square distances to all Delaunay spheres,

$$\varphi_S(y) \quad = \quad \min\{\varphi_z(y) \mid (z, r) \in Z\}.$$

The function $\varphi = \varphi_S$ is piecewise quadratic and it decomposes the convex hull of S into the Delaunay tetrahedra. Level sets of the functions π and φ defined for points in $\mathbb{R}^2$ are shown in Figure 1.

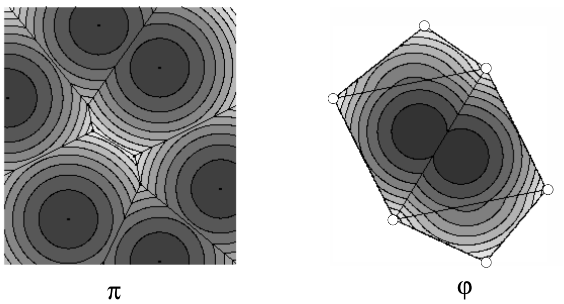


Figure 1: To the left, the level sets of π are overlaid with the Voronoi diagram. To the right, the level sets of φ are overlaid with the Delaunay triangulation.

Minima and maxima. The function $\pi : \mathbb{R}^3 \to \mathbb{R}$ has local minima and maxima similar to smooth Morse functions [7]. The minima of π are the points of S, and the maxima are the Voronoi vertices that are contained inside their dual Delaunay tetrahedra. Symmetrically, the maxima of φ are the points in S, and the minima are the circumcenters of Delaunay tetrahedra that are contained in their tetrahedra. We see that π and φ have the same set of local extremes, only that the minima and maxima are exchanged.

We define a *sink* as a local maximum of π, or equivalently a local minimum of φ. In three dimensions, a sink is therefore a circumcenter that is contained in the interior of its tetrahedron. For a point set S that satisfies the non-degeneracy assumption (3) there are no ambiguous cases of circumcenters lying on the boundary of their tetrahedra. Figure 2 visualizes sinks in two dimensions. Figure 3 visualizes the corresponding function φ.

Figure 2: Triangles that contain their own circumcenters are shown in grey; their circumcenters are the sinks.

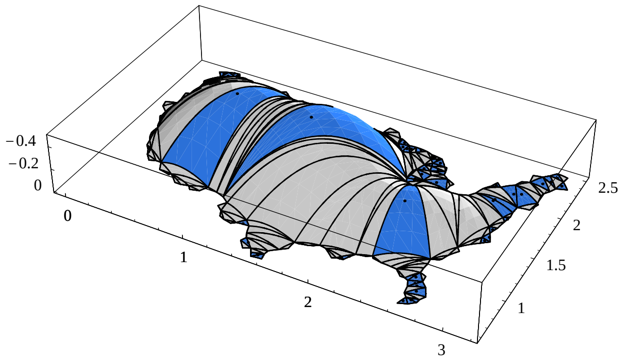


Figure 3: Visualization of φ with function values decreasing from bottom to top.

Partial order. We introduce bump functions in small neighborhoods of Delaunay triangles to make φ everywhere smooth. The bump functions can be chosen to preserve the set of maxima and minima, and as we shrink the bump functions we regain the original map φ in the limit.

Using the smoothness assumption we can define the *negative gradient* of φ, which maps every point $x \in \mathbb{R}^3$ to the vector $-\nabla\varphi(x)$ of the three partial derivatives. We think of $-\nabla\varphi$ as a vector or flow field and, following the example of [6], we use it as the guiding intuition in the construction of a partial order over the Delaunay tetrahedra. Specifically, define $\tau \prec \sigma$ if the negative gradient leads from the interior of τ through the common triangle into the interior of σ. Equivalently, $\tau \prec \sigma$ if the two tetrahedra share a common triangle and the circumcenter of τ lies beyond the plane spanned by that triangle. An example is shown in Figure 4. We introduce a dummy element ω that represents the space outside the convex hull of S and define $\tau \prec \omega$ for every Delaunay tetrahedron τ that has a triangle on the convex hull boundary and its circumcenter beyond the plane of that triangle.

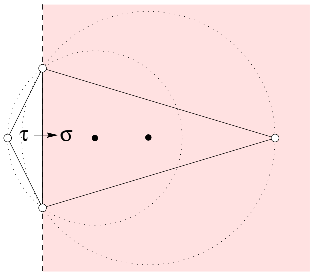


Figure 4: We have $\tau \prec \sigma$ because τ and σ share a common edge and the circumcenter of τ lies on σ's side of the line that passes through that edge.

Note that $\tau \prec \sigma$ implies that the circumradius of τ is less than that of σ. This is also true for pairs $\tau \prec \omega$ if we stipulate that the circumradius of ω is infinity. It follows that the relation is acyclic and its transitive closure is therefore a partial order. The tetrahedra without successor in the relation are exactly the ones that contain their own circumcenters. These circumcenters are the sinks as defined above. Let A_σ be the set of ancestors of σ under the relation, which includes σ itself. A_σ is maximal iff σ has no successor. Since one tetrahedron τ can have more than one descendent, it can also belong to more than one ancestor set. We therefore define the *set of sinks* of τ as the collection of circumcenters of tetrahedra that have no successors and whose ancestor sets contain τ.

4 Sink-insertion

Given a Delaunay triangulation, we improve its mesh quality by adding sinks to the vertex set. This section studies the combinatorics of that operation.

Prestars and flowers. Let $x \notin S$ be a point in the interior of the convex hull of S, and assume for convenience that $S \cup \{x\}$ is non-degenerate. Let D be the Delaunay triangulation of S and D_x the Delaunay triangulation of $S \cup \{x\}$. We need a name for the set of simplices in D that cover the same portion of space as the star of x in D_x. This is the *prestar* of x formally defined as $\mathrm{Pt}\, x = \mathrm{Cl}\, I - \mathrm{Cl}\, E$, where I is the set of Delaunay tetrahedra whose circumspheres enclose x and E is the set of remaining tetrahedra. We can obtain D_x from D by substituting the star for the prestar,

$$D_x \quad = \quad D - \mathrm{Pt}\, x \cup \mathrm{St}\, x.$$

This is essentially Watson's algorithm [10] for constructing Delaunay triangulations. The *flower* of a subset $L \subseteq D$ is the union of Delaunay balls of tetrahedra in L, denoted as $\mathrm{Fl}\, L$. We are only interested in the flowers of the prestar and the star of a vertex x, $\mathrm{Fl}\,\mathrm{Pt}\, x$ and $\mathrm{Fl}\,\mathrm{St}\, x$. Figure 5 illustrates the concept by showing the two flowers for a point x in a two-dimensional Delaunay triangulation.

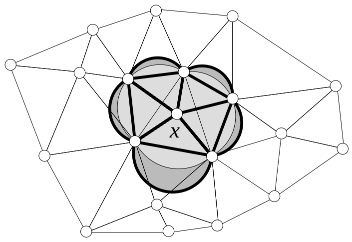


Figure 5: The bold edges bound triangles in the star of x. The light shading indicates the flower of the prestar, and the dark shading shows the difference between the flowers of the star and of the prestar.

Flowers expand. A plane decomposes a ball into two *segments*. Each triangle of a tetrahedron τ spans a plane that decomposes the ball bounded by the circumsphere into two segments. Of these the one disjoint from τ is referred to as the *exterior segment* of the triangle. Let L be a set of tetrahedra in $\mathrm{Del}\, S$. It is not hard to see that the flower of L is equal to the union of tetrahedra and exterior segments of triangles in $\mathrm{Cl}\, L$ that belong to only one tetrahedron in L.

As before assume that $x \notin S$ is a point in the interior of the convex hull of S. Equivalently, x is an interior vertex of D_x. The boundary triangles in the star of x are the same as those of the prestar. However, the presence of x pushes the circumspheres of the incident tetrahedra outwards. As a consequence, the flower of the star is strictly bigger than that of the prestar. This is illustrated in Figure 5.

FLOWER LEMMA. $\mathrm{Fl}\,\mathrm{Pt}\, x \subseteq \mathrm{Fl}\,\mathrm{St}\, x$.

PROOF. Because the tetrahedra of $\mathrm{Pt}\, x$ and $\mathrm{St}\, x$ cover the same space, it suffices to consider only exterior segments of boundary triangles. The new point x lies inside the circumsphere of the tetrahedron incident to such a triangle in $\mathrm{Pt}\, x$ and it lies on the circumsphere of the tetrahedron incident to the same triangle in $\mathrm{St}\, x$. It follows that after inserting x, the new exterior segments contain the old ones. ◻

Point insertion. The purpose of inserting points into a Delaunay triangulation is to locally improve the mesh quality. We use the ratio r/ℓ to measure the quality of a triangle or tetrahedron, where r is the radius of the circumsphere and ℓ is the length of the shortest edge. For tetrahedra the smallest ratio is obtained by the regular tetrahedron for which $r/\ell = \sqrt{3/8} = 0.612\ldots$. The Delaunay refinement algorithms published in the literature are designed to create triangulations where the maximum ratio is bounded from above by a constant.

The basic idea of our algorithms is to eliminate tetrahedra with high ratio by adding their sinks as new vertices to the Delaunay triangulation. We call one such addition a *sink-insertion*. Figure 6 illustrates the idea for the `USA` data set. We compare the performance of the algorithm to the traditional approach of using *circumcenter-insertions*. In the experimental study we consider three scheduling regimes: adding points incrementally, in blocks, and in parallel. We use the constant ϱ_0 as the threshold above which we attempt to remove triangles or tetrahedra. We need $\varrho_0 \geq 1.0$ in order to prove our algorithms halt after a finite amount of time. Before explaining the three regimes in detail we discuss geometric and combinatorial concepts which we use to avoid infinite loops that create accumulations of new vertices.

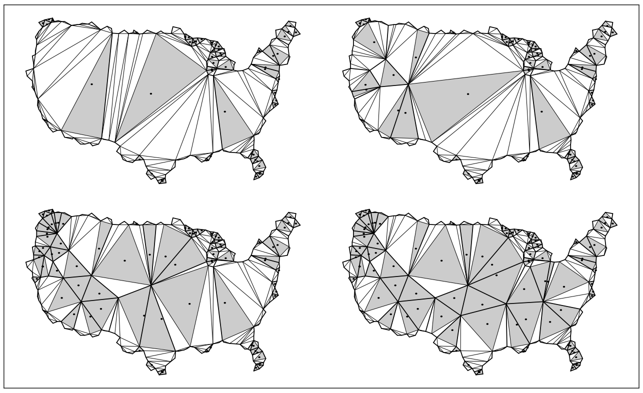


Figure 6: `USA` data set before adding any sinks and after incrementally adding 4, 12, and 15 sinks.

Minimum separation. Let $\delta > 0$ be the minimum distance between any two points in S. For each scheduling regime we make sure that all new vertices are placed within the underlying space of the original triangulation and not closer than δ to any of the other vertices. A straightforward packing argument then implies that the total number of vertices cannot exceed $6/\pi\delta^3$ times the volume of the underlying space expanded by $\delta/2$ in every direction.

The circumcenter z of a tetrahedron with ratio $r/\ell > \varrho_0 \geq 1.0$ has distance $r > \ell$ from the four vertices. The empty sphere criterion for Delaunay tetrahedra implies that the distance between z and any original or previously added vertex exceeds $\ell \geq \delta$. Call two spheres (y, q) and (z, r) *orthogonal* if $\|y - z\|^2 = q^2 + r^2$ and *further than orthogonal* if $\|y - z\|^2 > q^2 + r^2$. If two spheres are orthogonal or further than orthogonal then the distance between their two centers is at least as large as the larger of the two radii. We will use circumspheres of Delaunay tetrahedra with ratios exceeding $\varrho_0 \geq 1.0$. Hence $q, r > \delta$ and the orthogonality constraint on the two spheres implies $\|y - z\| > \delta$.

We also use combinatorial conditions to guarantee the minimum separation bound. Let $x \neq y$ be two points not in S but in the interior of the underlying space. We say that x and y are *independent* if the closures of their prestars are disjoint. Equivalently, the closed stars of x and y in the Delaunay triangulation of $S \cup \{x, y\}$ are disjoint. The following implications of independence will be useful. We state them using D, D_x, D_y, D_{xy} to denote the Delaunay triangulations of S, $S \cup \{x\}$, $S \cup \{y\}$, $S \cup \{x, y\}$.

INDEPENDENCE LEMMA. Statement I implies II, II implies III, and III implies IV.

I. Points x and y are independent.

II. The edge xy does not belong to D_{xy}.

III. The point x does not lie in the interior of the flower $\mathrm{Fl}\,\mathrm{St}\, y$ in D_y.

IV. The point x does not lie in the interior of the flower $\mathrm{Fl}\,\mathrm{Pt}\, y$ in D.

PROOF. I $\Longrightarrow$ II. The independence of x and y implies the disjointness of the stars of x and y in D_{xy}. Hence $xy \notin D_{xy}$.

II $\Longrightarrow$ III. If x belongs to $\mathrm{int}\,\mathrm{Fl}\,\mathrm{St}\, y$ in D_y then there is an empty sphere that passes through x and y. Then xy is an edge in D_{xy}, which contradicts II.

III $\Longrightarrow$ IV. By the Flower Lemma the flower of the prestar of y in D is contained in the flower of the star of y in D_y. ◻

Scheduling. Incremental insertion is the simplest form of scheduling. We pick an arbitrary triangle or tetrahedron with ratio $r/\ell > \varrho_0$ and insert its sinks or its circumcenter. After insertion we use local flip operations to restore the Delaunay triangulation. The minimum separation of δ between vertices is maintained because $\varrho_0 \geq 1.0$ and therefore $r > \ell \geq \delta$.

Currently most implementations of three-dimensional Delaunay triangulations are offline. By this we mean that the set of vertices has to be known in advance and cannot be changed without recomputing the entire triangulation. In such a software environment we cannot afford to retriangulate for every new vertex. We thus insert points in blocks, where a *block* is a maximal set of sinks or circumcenters of tetrahedra with $r/\ell > \varrho_0$, and all points in the blocks are centers of Delaunay spheres that are mutually orthogonal or further than orthogonal. The condition on r/ℓ guarantees the minimum separation of δ between vertices in the block and previously added vertices. Orthogonality implies the same minimum separation between vertices in the block. The schedule is therefore finite.

For parallel insertion we find the sinks or circumcenters of all tetrahedra whose ratios exceed ϱ_0. For each point we search the simplices in the closure of the prestar and we select a maximal subset with pairwise disjoint closed prestars. By definition the points in this subset are pairwise independent. We can therefore add the points in parallel even within the fairly restrictive exclusive read/exclusive write (EREW) model of the parallel random access machine (PRAM). As proved in the Independence Lemma, the points in the maximal subset are on or outside each others circumspheres. This implies that the minimum separation of δ between the vertices is maintained and the schedule is finite.

5 Experiments

This section presents the experimental results we obtained by applying the sink- and circumcenter-insertion algorithms to improve the mesh quality of two- and three-dimensional Delaunay triangulations. We begin by describing the assumptions and the measured parameters and continue by discussing the five data sets included in our experimental study. The results for each data set are summarized by displaying the triangulations, tabulating crucial complexity and quality parameters, and plotting the evolution of size and quality parameters during the iteration of sink-insertions. All time measurements are done on a Pentium II 450MHz processor with 128 MB of memory.

Assumptions and parameters. Each triangulation in this experiment is a *stable subcomplex* of a Delaunay triangulation, which by definition is a subcomplex whose circumcenters all lie in the interior of its underlying space. We are thus able to triangulate rather general and not just convex domains without having to represent the domain boundary in a separate data structure. We maintain the boundary and the stability of the subcomplex by disallowing the insertion of circumcenters that lie within the diametral sphere of a boundary triangle. This is the unique sphere that passes through the three vertices and whose center lies on the plane of the triangle. (Correspondingly in $\mathbb{R}^2$ we disallow the insertion of circumcenters inside diametral circles of boundary edges.) Our algorithms thus include a boundary test that rejects a point if the closure of its prestar contains a boundary triangle that contains the point inside its diametral sphere.

Recall that we add sinks or circumcenters for tetrahedra with ratio $r/\ell > \varrho_0$ and we use $\varrho_0 \geq 1.0$ in all cases. For ease of reference we say that a triangle or tetrahedron with ratio larger than ϱ_0 has *low quality*. Our experiments suggest that sink- and circumcenter-insertion lead to about the same mesh quality. Furthermore, the different scheduling regimes do not have much of an effect on the size and quality of the final triangulation. Sinks are typically shared by many low quality triangles and tetrahedra. If there are many low quality triangles or tetrahedra then they tend to share their sinks. As a result there are significantly fewer sinks than circumcenters to be considered by the algorithm. Another advantage of sink-insertion is the numerical robustness of the computation of sinks. Tetrahedra whose circumcenters are not sinks sometimes define ill-conditioned matrices causing floating-point errors in the circumcenter computation.

USA data. Figure 2 shows the `USA` data set with the maximal stable subcomplex of its Delaunay triangulation. Figure 7 show the output triangulations after block scheduled sink- and circumcenter-insertion with ratio threshold $\varrho_0 = 1.0$. The table below the graphics show that the quality of the two displayed triangulations as well as the ones obtained by incremental scheduling is about the same. The two graphs below the table plot the evolution of various parameters during the block iteration of sink-insertions. The first graph shows the number of low quality triangles, the number of their sinks, the number of sinks passing the boundary test, and the number of sinks per block. The second graph shows the maximum, average, and minimum ratios of the triangles. Initially, many low quality triangles share the same sinks and the triangles with lowest quality (highest ratios) tend to be eliminated first. After the eleventh iteration, when the average ratio is about 0.8, most low quality triangles contain their own sinks. Many of them fail the boundary test, which is why no significant improvement happens after that stage.

Cuboctahedron data. The graphics in Figure 8 shows the centrally symmetric input surface of the `Cuboctahedron` data set. It is a triangulated surface with 7,496 triangles. The minimum, average, and maximum ratios of these triangles are 0.58, 0.70, and 1.05. The table in Figure 8 shows the quality of the initial triangulation and the triangulations after sink- and after circumcenter-insertion.

Although the quality of the input triangles is good, the quality of initial triangulation is rather poor with r/ℓ as high as 22.80. This makes sense because the absence of any interior vertices forces skinny tetrahedra. We improve the quality by parallel scheduling with ratio threshold $\varrho_0 = 2.0$. The final triangulations after sink- and circumcenter-insertion have about the same quality, but the running time for circumcenter-insertion is considerably larger. The graphs in Figure 8 illustrate the reason. The first graph shows the evolution of the total number and the number of low quality tetrahedra. At the beginning nearly all tetrahedra have low quality. The first ten iterations reduce the total number of tetrahedra from 14,670 to 9,840 while increasing the number of vertices by 105. The second graph plots the evolution of the number of sinks and the size of the independent subsets of sinks. We observe that low quality tetrahedra tend to share sinks. They do not share circumcenters, and as a consequence sink-insertion spends significantly less time on boundary and independence tests. The third graph shows the evolution of the maximum ratio r/ℓ. Similar to the two-dimensional case, the tetrahedra with the lowest quality (highest ratios) tend to be eliminated first.

Paraboloid data. The graphics in Figure 9 shows the input surface of the `Paraboloid` data set. It is a triangulated surface with 6,272 triangles. The minimum, average, maximum ratios of these triangles are 0.58, 0.80, 1.31. As in the previous example the quality of the surface triangles is good but that of the initial tetrahedra is poor. We improve the quality by parallel scheduling with $\varrho_0 = 2.0$. The quality of the tetrahedra in the initial triangulation and in the triangulations after sink-insertion and after circumcenter-insertion is shown in the table in Figure 9. The latter two triangulations have about the same quality but the running time for circumcenter-insertion is again significantly larger than that for sink-insertion. The evolution of size and quality parameters shown in the three graphs of Figure 9 is similar to that observed for the `Cuboctahedron` data set.

Star data. The graphics in Figure 10 shows the input surface of the `Star` data set. It models a cavity inside the forward segment of the Reusable Solid Rocket Motor for NASA's Space Transportation System. The cavity is where the combustion starts. It has a shape of a star with eleven fins and is four meters long. The central cylinder has a radius of 0.5 meter. Each fin is 1.2 meters wide and 0.14 meters thick. The input surface is triangulated with 53,508 triangles. Their minimum, average, maximum ratios are 0.578, 1.315, 1.601. The quality of the surface triangles is good, but the initial triangulation contains tetrahedra with ratio as high as 14.09. We improve the quality by parallel scheduling with ratio threshold $\varrho_0 = 2.0$. The table in Figure 10 shows the quality of the initial triangulation and of the triangulations after sink-insertion and after circumcenter-insertion. Both algorithms give about the same quality, but sink-insertion is again faster, by a factor of about five. The graphs in Figure 10 show the evolution of size and quality parameters during the iteration. The graphs are different from the previous two examples, most likely because of the different shape characteristics. The `Star` model is relatively thin, which causes less clustering of low quality tetrahedra and therefore a different ratio between the numbers of such tetrahedra and the number of their sinks.

Head data. The graphics in Figure 11 shows the input surface of the `Head` data set. It models a solid propellant inside the forward segment of the Reusable Solid Rocket Motor. Geometrically it is a four meter long cylinder with the star of the previous example removed. The input surface is triangulated with 67,940 triangles. The surface triangulation around the star cavity is the same as that in the data set `Star`. The minimum, average, maximum ratios of the surface triangles are 0.577, 1.206 and 1.601. The quality of triangles is good, but the initial triangulation has again tetrahedra of poor quality. We improve the triangulation by parallel scheduling with ratio threshold $\varrho_0 = 2.0$. After improvement the number of tetrahedra nearly doubles, which should be compared to less than 50% increase in the previous three examples. Sink-insertion is about four times faster than circumcenter-insertion. The difference in speed is less than in the previous examples, most likely because the initial triangulation has better quality than before.

6 Conclusions

This paper studies the effect of using sinks instead of circumcenters in the Delaunay refinement algorithm for improving the mesh quality of three-dimensional Delaunay triangulations. While the restriction to sinks implies a dramatic improvement of running time, it has surprisingly little effect on the mesh quality. Mechanisms for refining the boundary and for removing slivers are not included in this paper as they shed no light on the primary subject of the study. Notwithstanding we plan to enhance the Delaunay refinement algorithm with such capabilities in order to reach the full potential of the Delaunay refinement method.

Acknowledgement

The authors thank Alan Cheng for his skin triangulation software used to generate the triangulated surface of the `cuboctahedron` and Xiangmin Jiao from the Center for Simulation of Advanced Rockets for the specification of the Reusable Solid Rocket Motor. The second author thanks Timothy J. Baker for a helpful discussion that inspired the study of the sink-insertion algorithm in a parallel setting.

References

[1] P. S. Alexandrov. *Combinatorial Topology, Volumes 1, 2 and 3.* Dover, Mineola, New York, 1998.

[2] L. P. Chew. Guaranteed-quality triangular meshes. Report TR 89-983, Comput. Sci. Dept., Cornell Univ., Ithaca, New York, 1989.

[3] B. Delaunay. Sur la sphère vide. *Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk* **7** (1934), 793–800.

[4] T. K. Dey, C. Bajaj and K. Sugihara. On good triangulations in three dimensions. *Internat. J. Comput. Geom. Appl.* **2** (1992), 75–95.

[5] 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.

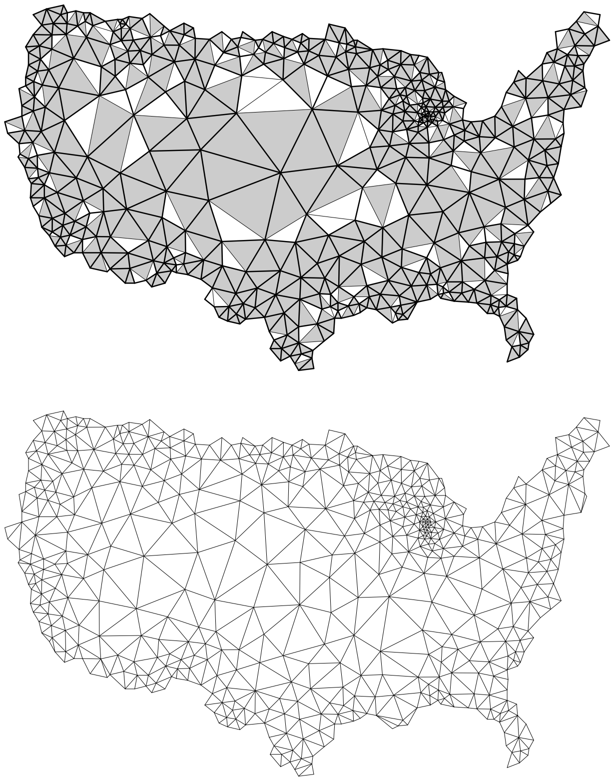
[6] H. Edelsbrunner, M. A. Facello and J. Liang. On the definition and the construction of pockets in macromolecules. *Discrete Appl. Math.* **88** (1998), 83–102.

[7] J. Milnor. *Morse Theory.* Princeton Univ. Press, New Jersey, 1963.

[8] J. Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J. Algorithms* **18** (1995), 548–585.

[9] J. Shewchuck. Tetrahedral mesh generation by Delaunay refinement. *In* "Proc. 14th Ann. Sympos. Comput. Geom., 1998", 86–95.

[10] D. F. Watson. Computing the n-dimensional Delaunay tessellation with applications to Voronoi polytopes. *Computer Journal* **24** (1981), 167–172.



		ratio r/ℓ		
	#tri	min	avg	max
init	229	0.61	2.36	17.23
sink, block	879	0.58	0.76	2.58
circ, block	941	0.58	0.75	3.54
sink, incr	841	0.58	0.77	3.06
circ, incr	997	0.58	0.76	3.54

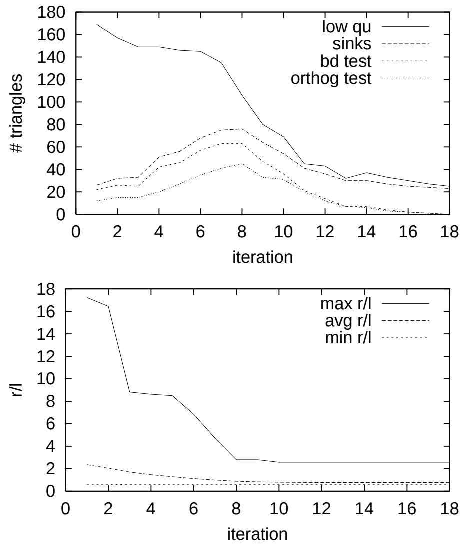
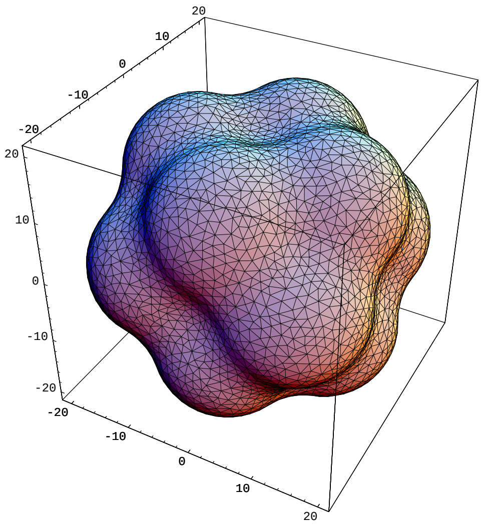


Figure 7: `USA` data set.



	time #min	#tetra	ratio r/ℓ min	avg	max
init		14,670	0.71	7.29	22.80
sink	6	15,948	0.63	1.27	2.00
circ	121	15,066	0.64	1.29	2.00

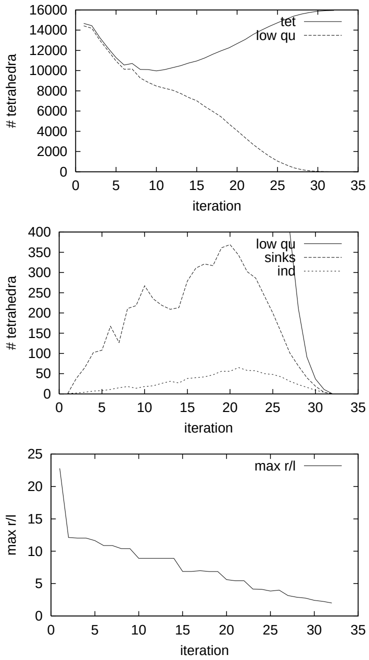
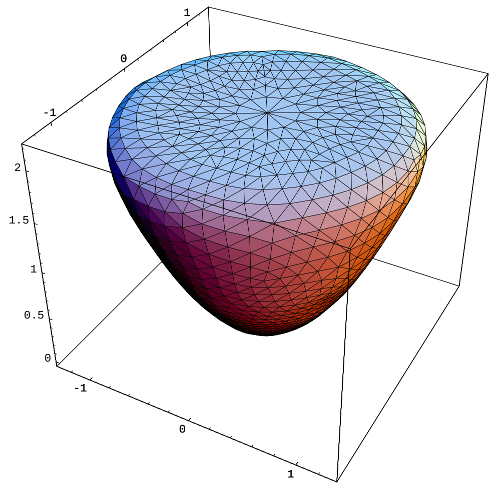


Figure 8: `Cuboctahedron` data set.



	time #min	#tetra	ratio r/ℓ min	avg	max
init		10,050	0.87	18.77	90.75
sink	7	14,793	0.65	1.24	2.00
circ	176	13,639	0.64	1.28	2.00

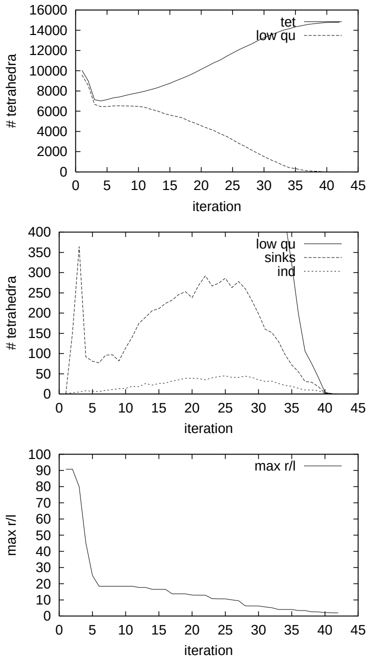
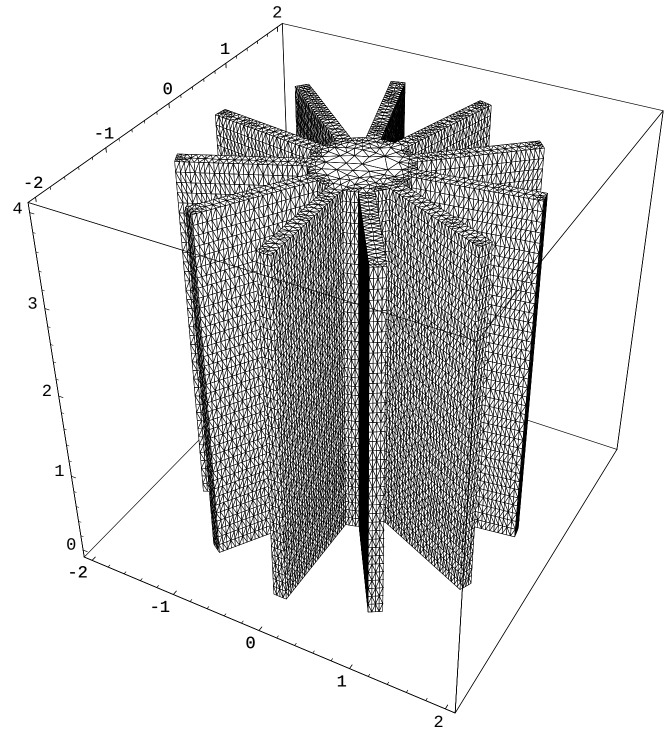


Figure 9: `Paraboloid` data set.



	time	#tetrahedra		ratio r/ℓ		
	#min	total	low qu	min	avg	max
init		78,114	58,147	0.68	2.40	14.09
sink	12	118,633	481	0.62	1.27	2.11
circ	58	119,442	1,867	0.63	1.27	2.17

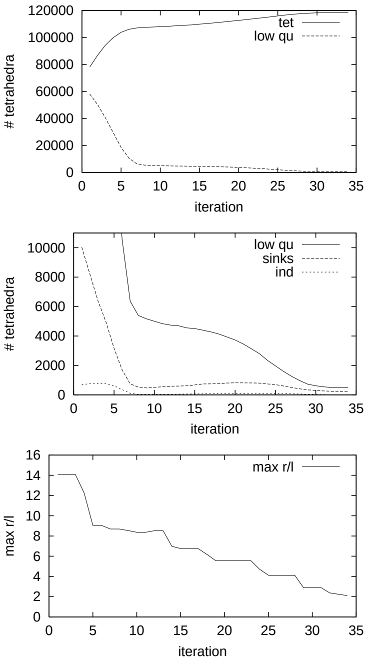
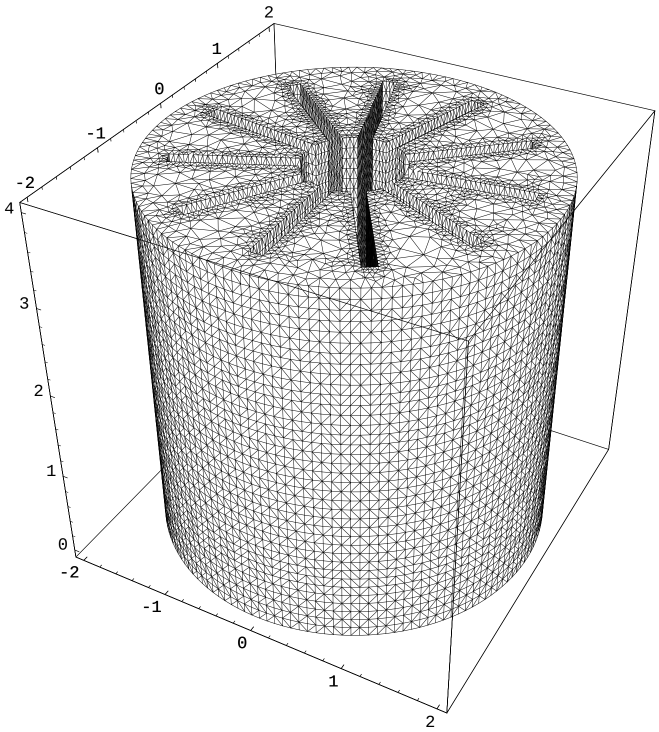


Figure 10: `Star` data set.



	time	#tetrahedra		ratio r/ℓ		
	#min	total	low qu	min	avg	max
init		100,454	79,238	0.67	4.11	8.67
sink	38	190,269	245	0.62	1.24	2.21
circ	177	183,027	104	0.62	1.27	2.15

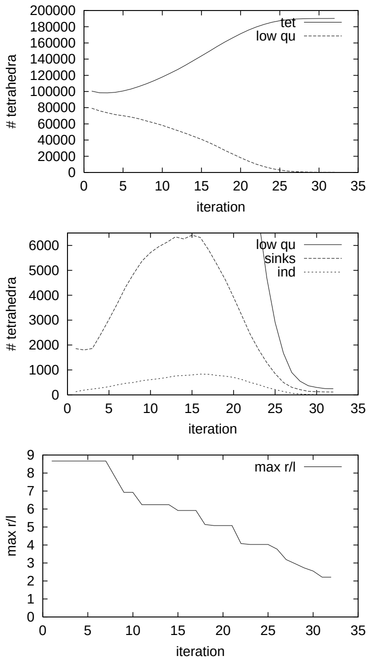


Figure 11: `Head` data set.