Delaunay Refinement

`Topics:` ratio property, polyhedra, refinement algorithm, local density, termination, graded meshes.

Ratio property. In this section we discuss the extension of the two-dimensional Delaunay refinement algorithm to three dimensions. A new idea that enters is the expression of shape quality of a tetrahedron σ as the ratio of circumradius over the length of the shortest edge. This ratio can also be defined for triangles. The circumradius of a tetrahedron is at least as large as the circumradii of its triangles, and its shortest edge is at most as long as the shortest edges of its triangles. It follows that the ratio of a tetrahedron is at least as large as the ratios of its four triangles.

For tetrahedra the minimum circumradius over length of shortest edge ratio is achieved by the regular tetrahedron. Assuming the edges have length $L = 1$, the ratio is equal to the circumradius R, which in Figure 11 is the distance between z and any one of the four vertices. For a regular tetrahedron $abcd$ the circumcen-

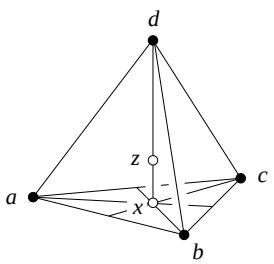

Figure 11: A regular tetrahedron $abcd$, the barycenter x of abc and the barycenter z of $abcd$.

ter is also the barycenter, and the same is true for each of its triangles:

$$z \;=\; \frac{1}{4}\cdot(a+b+c+d) \;=\; \frac{3}{4}\cdot x + \frac{1}{4}\cdot d,$$

where $x = \frac{1}{3}\cdot(a+b+c)$ is the circumcenter of abc. The circumradius of abc is $\|a-x\| = \frac{2}{3}\cdot\frac{\sqrt{3}}{2} = \frac{\sqrt{3}}{3}$, and the circumradius of $abcd$ is

$$\|d-z\| \;=\; \frac{3}{4}\cdot\sqrt{1-\frac{3}{9}} \;=\; \frac{\sqrt{6}}{4} \;\approx\; 0.61237.$$

All other tetrahedra have larger ratio. A 3-complex D has the *ratio property* for ϱ_0 if $R/L \le \varrho_0$ for all tetrahedra and triangles in D. Regular tetrahedra cannot fill the space around an edge, so every 3-complex that has at least one interior edge has ϱ_0 strictly larger than $\sqrt{6}/4$. Although smaller values of ϱ_0 generally correspond to better quality complexes, we will have technical reasons to choose ϱ_0 strictly larger than 2.

Polyhedra. The input to the mesh generation problem is a 3-dimensional *polyhedron*, which we define as the union of convex polyhedra,

$$P \;=\; \bigcup_{i\in\Lambda}\bigcap H_i,$$

where Λ is a finite index set and each H_i is a finite set of closed half-spaces. For example the polyhedron in Figure 12 can be specified as the union of four convex polyhedra. As we can see, faces are not necessarily sim-

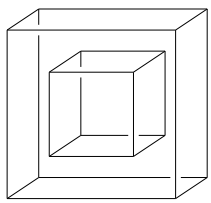

Figure 12: The window-frame polyhedron. Eight of its ten facets are rectangles and the front and fack facets are congruent annuli.

ply connected. We use a definition that permits faces that are not even connected. Let b be the open ball with unit radius centered at the origin of $\mathbb{R}^3$. For a point x we consider a sufficiently small neighborhood $N_\varepsilon(x) = (x + \varepsilon\cdot b)\cap P$. The *face figure* of x is the enlarged version of this neighborhood:

$$\mathrm{ff}\, x \;=\; x + \bigcup_{\lambda>0}\lambda\cdot(N_\varepsilon(x)-x).$$

A *face* of P is the closure of a maximal collection of points with identical face figure. To distinguish the faces of P from the edges and triangles of the Delaunay complex we call 1-faces of P *segments* and 2-faces of P *facets*. Observe that the window-frame polyhedron in Figure 12 has 16 vertices, 24 segment, 10 facets, and two 3-faces, namely the inside with face figure $\mathbb{R}^3$ and the outside with face figure $\emptyset$. We assume the input to the mesh generation problem is a suitable data structure that represents the face structure of the polyhedron P.

For technical reasons we need to restrict ourselves to bounded polyhedra without interior angles smaller than $\frac{\pi}{2}$. The condition applies to angles between two segments, between a segment and a facet, and between two facets. The window-frame in Figure 12 satisfies the condition, but in general the angle constraint is rather limiting and it would be nice to remove it.

Refinement algorithm. The goal is to construct a Delaunay complex D with a subcomplex $K \subseteq D$ that subdivides P, that is, $|K| = P$. The first step of the algorithm computes D as the Delaunay complex of the set of vertices of P. Unless we are lucky there will be segments that are not covered by edges of D, and there will be facets that are not covered by triangles of D. To recover these segments and facets, we add new points and update the Delaunay complex using e.g. the incremental flipping algorithm of the previous section. The points are added using the three rules below. We need some definitions. A segment of P is decomposed into *subsegments* by vertices of the Delaunay complex that lie on the segment, and a facet is decomposed into (triangular) *subfacets* by the two-dimensional Delaunay triangulation of the vertices on the facet or its boundary. A vertex *encroaches upon* a subsegment if it is enclosed by the diametral sphere of that subsegment, and it *encroaches upon* a subfacet if it is enclosed by the equatorial sphere of that subfacet.

`Rule 1.` If a subsegment is encroached upon we split it by adding its midpoint as a new vertex to the Delaunay complex. The new subsegments may or may not be encroached upon, and splitting continues until none of the subsegments is encroached upon.

`Rule 2.` If a subfacet is encroached upon we split it by adding its circumcenter x as a new vertex to the Delaunay complex. However, if x encroaches upon one or more subsegments then we do not add x and instead split the subsegments.

`Rule 3.` If a tetrahedron inside P has ratio $R/L > \varrho_0$ then we split the tetrahedron by adding its circumcenter x as a new vertex to the Delaunay complex. However, if x encroaches upon any subsegments or subfacets we do not add x and instead split the subsegments and subfacets.

Rule 1 takes priority over Rule 2, which takes priority over Rule 3. At the time we add a point on a facet, the prioritization guarantees that the boundary segments of the facet are subdivided by edges of the Delaunay complex. Similarly, at the time we add a point in the interior of P, the boundary of P is subdivided by triangles in the Delaunay complex. A point considered for addition to the Delaunay complex has a *type*, which is the number of the rule that considers it or equivalently the dimension of the simplex it splits. Points of type 1 split subsegments and are always added once they are considered. Points of type 2 and 3 may be added or rejected.

Local density. Just like in two dimensions, the *local feature size* is crucial in understanding the Delaunay refinement algorithm. It is the function $f : \mathbb{R}^3 \to \mathbb{R}$ with $f(x)$ the radius of the smallest closed ball with center x that intersects at least two disjoint faces of P. Note that f is bounded away from zero by some positive constant. It is easy to show that f satisfies the Lipschitz condition

$$f(x) \quad \leq \quad f(y) + \|x - y\|.$$

This implies that f is continuous over $\mathbb{R}^3$, but more than that the condition says that f varies only slowly with x.

The local feature size is related to the *insertion radius* r_x of a point x, which is the length of the shortest Delaunay edge with endpoint x right after adding x. If x is a vertex of P then r_x is the distance to the nearest other vertex of P. If x has type 1 or 2 then r_x is the distance to the nearest encroaching vertex, and if that vertex does not exist because it was rejected, then r_x is twice the length of the subsegment or the circumradius of the subfacet x splits. If x has type 3 then r_x is the circumradius of the tetrahedron x splits. We define the insertion radius also for points that are considered for addition but are rejected because they encroach upon subsegments or subfacets.

Points are added in a sequence, and for each new point there are predecessors that we can make responsible for the addition. If x has type 1 or 2 then we define the responsible *parent* $p = p_x$ as the encroaching point that triggers the event. The point p may be a Delaunay

vertex or a rejected midpoint or circumcenter. If there are several encroaching points then p is the one closest to x. If x has type 3 then p is the most recently added endpoint of the shortest edge of the tetrahedron x splits. The relation between the insertion radius and the local feature size stated in Claim 5 below is the result of a short case analysis.

CLAIM 5. Let x be a vertex of D and p its parent, if it exists. Then $r_x \geq f(x)$ or $r_x \geq c \cdot r_p$, where $c = 1/\sqrt{2}$ if x has type 1 or 2 and $c = \varrho_0$ if x has type 3.

Termination. Claim 5 limits how quickly the insertion radius can decrease. We aim at choosing the only independent constant, which is ϱ_0, such that the insertion radii are bounded from below by a positive constant. Once this is achieved we can prove termination of the algorithm using a standard packing argument. Figure 13 illustrates the possible parent-child relations between the three types of points added by the algorithm. We follow an arc of the digraph whenever the

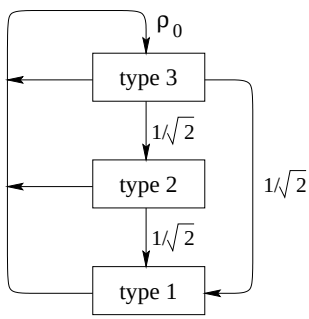

Figure 13: The directed arcs indicate possible parent-child relations, and their labels give the worst case factors relating the insertion radii.

insertion radius of a point x is determined by its parent and not directly by the local feature size at x. The arc is labeled by the smallest possible factor relating the insertion radius of x to that of its parent. Note that there is no arc from type 1 to type 2 and there are no loops from type 1 back to type 1 and from type 2 back to type 2. This is because the angle constraint on the input polyhedron prevents parent-child relations for points on segments and facets with non-empty intersection. If there is a relation between points on segments and facets with empty intersection then $r_x \geq f(x)$ and there is no need to follow an arc in the digraph.

Observe that every cycle in the digraph contains the arc labeled ϱ_0 leading into type 3. We choose $\varrho_0 \geq 2$ so the products of arc labels for all cycles are 1 or larger. In cases where r_x is not at least $f(x)$ there exist other points q with $r_x \geq r_q/2$ and $r_q \geq f(q)$. Since $f(q)$ is bounded away from zero by some positive constant, we conclude that the insertion radii cannot get arbitrarily small. It follows that the Delaunay refinement algorithm terminates. For $\varrho_0 < 2$ there are cases where the algorithm does not terminate.

Graded meshes. With little additional effort we can show that for ϱ_0 strictly larger than 2, insertion radii are directly related to local feature size, and not just indirectly through chains of ancestors. We begin with a relation between the local feature size over insertion radius ratio of a vertex and of its parent.

CLAIM 6. Let x be a Delaunay vertex with parent p and assume $r_x \geq c \cdot r_p$. Then

$$\frac{f(x)}{r_x} \leq 1 + \frac{f(p)}{c \cdot r_p}.$$

PROOF. We have $r_x = \|x - p\|$ if p is a Delaunay vertex and $r_x \geq \|x - p\|$ if p is a rejected midpoint or circumcenter. Starting with the Lipschitz condition we get

$$\begin{aligned} f(x) &\leq f(p) + \|x - p\| \\ &\leq \frac{f(p) \cdot r_p}{r_p} + r_x \\ &\leq \frac{f(p)}{c \cdot r_p} \cdot r_x + r_x, \end{aligned}$$

and the result follows after dividing by r_x. □

To prepare the next step we assume $\varrho_0 > 2$ and define constants

$$\begin{aligned} c_1 &= \frac{(3+\sqrt{2}) \cdot \varrho_0}{\varrho_0 - 2}, \\ c_2 &= \frac{(1+\sqrt{2}) \cdot \varrho_0 + \sqrt{2}}{\varrho_0 - 2}, \\ c_3 &= \frac{\varrho_0 + 1 + \sqrt{2}}{\varrho_0 - 2}. \end{aligned}$$

Note that $c_1 > c_2 > c_3 > 1$.

INVARIANT 7. If x is a type i vertex in the Delaunay complex, for $1 \leq i \leq 3$, then $r_x \geq f(x)/c_i$.

PROOF. If the parent p of x is a vertex of the input polyhedron P then $r_x \geq f(x)$ and we are done. Otherwise, assume inductively that the claimed inequality

holds for vertex p. We finish the proof by case analysis. If x has type 3 then $r_x \geq \varrho_0 \cdot r_p$ by Claim 5, and

$$\frac{f(x)}{r_x} \;\leq\; 1 + \frac{c_1}{\varrho_0} \;\leq\; c_3$$

by Claim 6. If x has type 2 we have $r_x \geq f(x)$ unless p has type 3. Then $r_x \geq r_p/\sqrt{2}$ by Claim 5, and

$$\frac{f(x)}{r_x} \;\leq\; 1 + \sqrt{2} \cdot c_3 \;\leq\; c_2$$

by Claim 6. If x has type 1 we have $r_x \geq f(x)$ unless p has type 2 or 3. Then $r_x \leq 1 + r_p/\sqrt{2}$ by Claim 5, and

$$\frac{f(x)}{r_x} \;\leq\; 1 + \sqrt{2} \cdot c_2 \;\leq\; c_1$$

by Claim 6. ∎

Because c_1 is the largest of the three constants, we can simplify Invariant 7 to $r_x \geq f(x)/c_1$ for every Delaunay vertex x. From this we conclude

$$\|a - b\| \quad \geq \quad \frac{f(x)}{1 + c_1}$$

for any two vertices in the Delaunay complex, see also Lemma 10 in Chapter I.

Bibliographic notes. The bulk of the material in this section is taken from a paper by Jonathan Shewchuk [6]. In that paper he extend the algorithm of Jim Ruppert [5] described in Section 5 from two to three dimensions. The idea of measuring the quality of a tetrahedron by its circumradius to length of shortest edge ratio is due to Miller and coauthors [4]. The definition of a polyhedron as the union of convex polyhedra goes back at least to Hadwiger [2]. The right way to define a face of a polyhedron is a debated issue in the geometry literature, see [3], and the definition in this section is taken from [1].

[1] H. Edelsbrunner. Algebraic decomposition of non-convex polyhedra. *In* "Proc. 36th Ann. IEEE Sympos. Found. Comput. Sci., 1995", 248–257.

[2] H. Hadwiger. *Vorlesungen über Inhalt, Oberfläche und Isoperimetrie.* Springer, Berlin, 1957.

[3] B. Grünbaum and Shephard.

[4] 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 Ann. ACM Sympos. Theory Comput., 1995", 683–692.

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

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