Guaranteed Quality Tetrahedral Delaunay Meshing for Medical Images

Panagiotis A. Foteinos[a,b,*], Andrey N. Chernikov[b], Nikos P. Chrisochoides[b]

[a] *Department of Computer Science, College of William & Mary, Williamsburg, 23185, Virginia, USA*
[b] *Department of Computer Science, Old Dominion University, Norfolk, 23569, Virginia, USA*

Abstract

In this paper, we present a Delaunay refinement algorithm for meshing 3D medical images. Given that the surface of the represented object is a smooth 2-manifold without boundary, we prove that (a) all the tetrahedra of the output mesh have radius-edge ratio less than $\sqrt{\sqrt{3}+2}\,(\approx 1.93)$, (b) all the boundary facets have planar angles larger than 30 degrees, (c) the symmetric (2-sided) Hausdorff distance between the object surface and mesh boundary is bounded from above by a user-specified parameter, and (d) the mesh boundary is ambient isotopic to the object surface. The first two guarantees assure that our algorithm produces elements of bounded radius-edge ratio. The last two guarantees assure that the mesh boundary is a good geometric and topological approximation of the object surface. Our method also offers control over the size of tetrahedra in the final mesh. Experimental evaluation of our algorithm on synthetic and real medical data illustrates the theory and shows the effectiveness of our method.

Keywords: Delaunay mesh generation; medical images; quality; fidelity;

1. Introduction

Delaunay meshing is a popular technique for generating tetrahedral meshes, since it is able to mesh various domains such as: polyhedral domains [1, 2],

*Corresponding author
Email addresses: pfot@cs.wm.edu (Panagiotis A. Foteinos), achernik@cs.odu.edu (Andrey N. Chernikov), nikos@cs.odu.edu (Nikos P. Chrisochoides)

domains bounded by surfaces [3, 4], or multi-labeled images [5, 6], offering at the same time mathematical guarantees on the quality and the fidelity of the final mesh.

In the literature, Delaunay refinement techniques have been employed to mesh objects whose surface is already meshed as a *Piecewise Linear Complex* (PLC) [1, 2, 7–9]. The challenge in this category of techniques is that the quality of the input PLC affects the quality of the final volume mesh. For example, if the input angles of the PLC are small, then even termination might be compromised [10]. For images, one way to alleviate this challenge is to consider the faces of each outer voxel as the input PLC, since these faces meet at large angles (90° or 180°). However, this would result in an unnecessarily large final mesh.

Another approach is to assume that the object Ω to be meshed is known only through an implicit function $f : \mathbb{R}^3 \to \mathbb{Z}$ such that points in different regions of interest evaluate f differently. This assumption covers a wide range of inputs used in modeling and simulation, such as parametric surfaces/volumes [3], level-sets, and segmented multi-labeled images [5, 6, 11], the focus of this paper. If the subsequent simulation permits sharp features of the domain to be rounded-off, such functions can be used to represent PLCs as well [11], a fact that renders this approach quite general. It should be noted that these methods do not suffer from any small input angle artifacts introduced by the initial conversion to PLCs, since the isosurface $\partial\Omega$ of the object Ω is recovered and meshed during refinement. In this paper, we deal with objects whose surface is a smooth 2-manifold (see Section 2). It is the algorithm's responsibility to mesh both the surface and the interior of the object such that the mesh boundary describes the object surface in a way that meets the predefined fidelity and quality requirements.

The quality of an element is traditionally measured in terms of its *circumradius-to-shortest-edge* ratio or *radius-edge* ratio for short. It is desirable that the mesh elements have radius-edge ratio bounded from above. Meshes satisfying that bounded ratio property are called *almost-good* meshes in the literature [12]. Miller *et al.* [13] show that almost-good meshes guarantee optimal convergence rates for approximate solutions of Poisson's equation.

Delaunay volume meshing algorithms extend the popular Delaunay surface meshing and reconstruction algorithms described in [14, 15], and they offer quality and fidelity guarantees [3, 4] under the assumption that the surface of the object is smooth [3, 15] or does not form input angles less than 90° [4]. However, the quality achieved by these algorithms is somewhat weak:

the upper bound for the elements' radius-edge ratio is larger than 4. In contrast, the upper bound guaranteed by our algorithm is $\sqrt{\sqrt{3}+2}\,(\approx 1.93)$. To our knowledge, our algorithm is the first volume Delaunay mesher for surfaces achieving such a small radius-edge ratio with these fidelity guarantees.

Almost-good meshes, however, might contain nearly flat elements, the so called *slivers*. The reason is that slivers can have a very small radius-edge ratio and at the same time a very small dihedral angle. In the literature, there are post-processing techniques that given an almost-good mesh, they are able to remove slivers. See for example the work of Li and Teng [12], the *exudation technique* of Cheng *et al.* [8], and the *sliver perturbation* of Tournois *et al.* [16]. In fact, the sliver removal technique of Li and Teng [12] requires a low radius-edge ratio, since the lower the radius-edge ratio, the larger the guaranteed bound on the minimum dihedral angles. This is another motivation for achieving low radius-edge ratio.

The success of Delaunay techniques to approximate the surface relies on the notion of ε-samples, first introduced by Amenta and Bern [17]. The construction of ε-samples directly from the surface is a challenging task. In the literature, however, it is assumed that either such a sample is known [15, 17, 18] or that an initial sparse sample is given on every connected component [3, 4, 19]. In this paper, we propose a method that starts directly from labeled images and computes the appropriate sample on the fly.

In summary, we present a Delaunay refinement algorithm which:

- operates directly on images, samples and meshes the surface and the volume of the represented biological object at the same time,
- is proved to generate tetrahedra with radius-edge ratio less than $\sqrt{\sqrt{3}+2}\,(\approx 1.93)$ and boundary facets with planar angles more than 30°,
- is proved to yield meshes of good fidelity, i.e., the mesh surface is guaranteed to be a faithful geometric and topological approximation of the object's surface, and
- offers control over the size and the grading of the final mesh.

In the literature, there are also non-Delaunay surface and volume meshing algorithms for 3D images. *Marching Cubes* [20] is a very popular technique for surface meshing; it guarantees, however, neither good quality triangular facets nor faithful surface approximation. Furthermore, since the cubes

have a very small size (close to the voxel size), Marching Cubes does not offer a way to control the size of the mesh. Molino *et al.* [21] develop the *Red-Green Mesh* (RGM) method. RGM starts by meshing an initial body-centric cubic (BCC) lattice which is then compressed such that its boundary fit on the surface. RGM gives, however, no quality or surface approximation guarantees. Another lattice-based method is the *Isosurface Stuffing* of Labelle and Shewchuk [11]. They prove that the graded version of the final mesh consists of elements with dihedral angles larger than 1.66°. The *Lattice Decimation* method proposed by Chernikov and Chrisochoides [22] is guaranteed to produce a good geometric approximation of the underlying object; however, topological faithfulness is not guaranteed. Alliez *et al.* [23] introduce a Delaunay-based optimization technique. Specifically, they iteratively compute the new locations of the points by minimizing a quadratic energy. The connectivity of these points is recalculated by finding their Delaunay triangulation each time. They show that this technique produces meshes that respect the boundary of the domain. Klingner and Shewchuk [24] extend the work of Freitag and Ollivier-Gooch [25] by proposing smoothing and topological transformations which improve the quality of the mesh substantially. The execution time, however, can be very high, even for small mesh size problems.

The rest of this paper is organized as follows: Section 2 provides the necessary definitions and Section 3 outlines our algorithm. Section 4 proves the quality, Section 5 proves the good grading, and Section 6 proves the fidelity guarantees. Finally, Section 7 elaborates on certain implementation details, Section 8 assesses the practical value of our work on both synthetic and real medical data, and Section 9 concludes our paper.

2. Preliminaries

Let $\mathcal{I} \subset \mathcal{R}^3$ be the (spatial) domain of a multi-tissue segmented image. $\mathcal{I}$ is the input of our algorithm that contains the object $\Omega \subset \mathcal{I}$ to be meshed. We assume that the object is partitioned into a finite number of n distinct tissues $\Omega = \bigcup_{i}^{n} \Omega_i$, $i = 1, \ldots, n$. Each Ω_i defines an interface $\partial\Omega_i$ that consists of the set of points that lie on the boundary between Ω_i and at least one more tissue or the background of the image. The *isosurface* $\partial\Omega$ of Ω is then the collection of all interfaces; that is, $\partial\Omega = \bigcup_{i}^{n} \partial\Omega_i$, $i = 1, \ldots, n$. We assume

that we are given a function $f : \mathcal{I} \to \{-1, 0, 1, \ldots, n\}$, which classifies every point $p \in \mathcal{I}$ appropriately. Specifically, p evaluates f to -1 if it lies on $\partial\Omega$, to 0 if it lies in the background (i.e., outside the object), or to a positive integer i if it belongs to the tissue Ω_i. The existence of such a function is a quite reasonable assumption: f can be constructed or approximated from the image voxels quite well for any segmented image (see Section 7 for details on f's implementation).

As is generally the case in the literature [3, 15, 19], we also assume that $\partial\Omega$ is a smooth (twice differentiable) 2-manifold without boundary.

Definition 1 (*medial axis*, Blum [26])**.** *The medial axis of $\partial\Omega$ is the closure of the set of those points having more than one closest point on $\partial\Omega$.*

Definition 2 (*local feature size*, Amenta and Bern [17])**.** *The local feature size of a point $p \in \partial\Omega$, denoted as $lfs_{\partial\Omega}(p)$, is the distance from p to the medial axis of $\partial\Omega$.*

We denote with $\mathrm{lfs}^{\inf}_{\partial\Omega}$ and $\mathrm{lfs}^{\sup}_{\partial\Omega}$ the infimum and the supremum of the local feature sizes of all the points on $\partial\Omega$ respectively, that is: $\mathrm{lfs}^{\inf}_{\partial\Omega} = \inf\{\mathrm{lfs}_{\partial\Omega}(p) : p \in \partial\Omega\}$ and $\mathrm{lfs}^{\sup}_{\partial\Omega} = \sup\{\mathrm{lfs}_{\partial\Omega}(p) : p \in \partial\Omega\}$. Note that since $\partial\Omega$ is assumed to be a smooth manifold, both $\mathrm{lfs}^{\inf}_{\partial\Omega}$ and $\mathrm{lfs}^{\sup}_{\partial\Omega}$ are positive real constants. Another useful property is that the local feature size is 1-Lipschitz, that is,

$$\mathrm{lfs}_{\partial\Omega}(p) \leq |pq| + \mathrm{lfs}_{\partial\Omega}(q). \tag{1}$$

Definition 3 (*ε-sample*, Amenta *et al.* [15])**.** *A point set $P \subset \partial\Omega$ is called an ε-sample of $\partial\Omega$, if for every point $p \in \partial\Omega$ there is a sample point $q \in P$, such that $|pq| \leq \varepsilon \cdot lfs_{\partial\Omega}(p)$.*

Next, we define a special *restriction*:

Definition 4 (*restricted Delaunay triangulation*, Boissonnat *et al.* [19])**.** *Let $\mathcal{D}(P)$ be the Delaunay triangulation of the point set P. The restriction of $\mathcal{D}(P)$ to $\partial\Omega$, denoted as $\mathcal{D}_{|\partial\Omega}(P)$, contains the facets in $\mathcal{D}(P)$ whose dual Voronoi edges intersect $\partial\Omega$.*

We shall refer to a facet whose dual Voronoi edge intersects $\partial\Omega$ as a *restricted* facet. We denote the Voronoi edge of a facet f with $\mathrm{Vor}(f)$.

In [19], the following useful theorem is proved:

Theorem 1 (Boissonnat *et al.* [19])**.** *If P is an ε-sample of $\partial\Omega$ with $\varepsilon < 0.09$, then:*

- $\mathcal{D}_{|\partial\Omega}(P)$ *is a 2-manifold ambient isotopic to* $\partial\Omega$ *and*
- *the 2-sided Hausdorff distance between* $\mathcal{D}_{|\partial\Omega}(P)$ *and* $\partial\Omega$ *is* $O(\varepsilon^2)$.

We next define the *surface ball* of a restricted facet:

Definition 5 (*surface ball*, Oudot *et al.* [3])**.** *Let* f *be a restricted facet and* e *be* f*'s dual Voronoi edge. The surface ball* $B_{surf}(f)$ *of* f *is a closed ball which is centered at a point* $p \in e \cap \partial\Omega$ *and passes through* f*'s vertices.*

In the rest of the paper, the center and radius of restricted facet f's surface ball $B_{\text{surf}}(f)$ are denoted by $c_{\text{surf}}(f)$ and $r_{\text{surf}}(f)$, respectively.

The following Remark follows directly from the fact that the center of restricted facet f's surface ball lies on its Voronoi edge:

Remark 1. *The surface ball of* f *contains no vertices in its interior.*

A real point p is called a *vertex*, if it has been already inserted into the mesh. Point p is called a *feature* point (or a feature vertex, if p is inserted into the mesh), if it is a surface point, i.e., $p \in \partial\Omega$. In the rest of the paper, $\text{cfp}(p)$ denotes the *Closest Feature Point* to p.

An *element* t is a tetrahedron, a (triangular) facet, or an edge. The *diametral ball* $B(t)$ of t is the set of points that lie inside or on t's smallest circumscribing sphere. The smallest circumscribing sphere of an element t will be sometimes called its *diametral sphere* and symbolized by $S(t)$. The center of t's diametral ball/sphere and the radius of t's diametral sphere are denoted by $c(t)$ and $r(t)$, respectively. The shortest edge of element t is denoted by $l_{\min}(t)$. Finally, the radius-edge ratio $\rho(t)$ of a tetrahedron or facet t is defined as $\rho(t) = \frac{|r(t)|}{|l_{\min}(t)|}$.

3. Algorithm

The user specifies as input the target upper radius-edge ratio $\bar{\rho}_t$ for the mesh tetrahedra, the target upper radius-edge ratio $\bar{\rho}_f$ for the mesh boundary facets, and parameter δ. It will be clear in Section 6 that the lower δ is, the better the mesh boundary will approximate $\partial\Omega$. For brevity, the quantity $\delta \cdot \text{lfs}_{\partial\Omega}(z)$ is denoted by $\Delta_{\partial\Omega}(z)$, where z is a feature point.

Our algorithm initially inserts the 8 corners of a cubical box B that contains the object Ω, such that the distance between a point p on the box and any feature point z is at least $2\Delta_{\partial\Omega}(z)$. Since $\text{lfs}_{\partial\Omega}(z) \leq \text{lfs}^{\sup}_{\partial\Omega}$, it suffices

to construct B such that it is separated from the minimum bounding box of Ω by a distance of at least $2 \cdot \delta \cdot \text{lfs}^{\text{sup}}_{\partial\Omega}$. Let d be the diagonal of the minimum bounding box of Ω. Clearly, constructing box B to be separated from the minimum bounding box by a distance of at least $\delta \cdot d$ fulfills the requirement, since $\text{lfs}^{\text{sup}}_{\partial\Omega}$ cannot be larger than $\frac{d}{2}$.

After the computation of this initial triangulation, the refinement starts dictating which extra points (also known as *Steiner* points) are inserted or which vertices are deleted. At any time, the Delaunay triangulation $\mathcal{D}(V)$ of the current vertices V is maintained. Note that by construction, $\mathcal{D}(V)$ always covers the entire object and that any point on the box is separated from $\partial\Omega$ by a distance of at least $2\Delta_{\partial\Omega}(z)$, where z is a feature point.

The users can also define their own customized *Size Function* $\text{sf} : \Omega \mapsto \mathbb{R}^+$ and pass it as input to our mesher. The size function sets an upper bound on the radii of the circumballs of the tetrahedra, and thus offers the flexibility of controlling which parts of the domain need a denser representation.

During the refinement, some vertices are inserted exactly on the box; these vertices are called *box vertices*. The edges that lie precisely on one of the 12 edges of the bounding box are called *box edges*. We further divide the box vertices into two categories: *box-edge vertices* and *non-box-edge vertices*. The former vertices lie precisely on a box edge, while the latter do not. The facets that lie precisely on one of the 6 faces of the box are called *box facets*. For example, the initial triangulation contains just 8 box vertices (which are also box-edge vertices) and 12 box edges (among other edges). Note that the endpoints of a box edge are always box edge vertices, but the opposite is not always true. We shall refer to the vertices that are neither box vertices nor feature vertices as *free vertices*.

Next, we define two types of tetrahedra:

- **intersecting tetrahedra**: tetrahedra whose circumsphere intersects $\partial\Omega$ (i.e., there is at least one feature point in their circumball), and
- **interior tetrahedra**: tetrahedra whose circumcenter lies (strictly) inside Ω.

Note that a tetrahedron might be both intersecting and interior or might belong to neither type.

The algorithm inserts new vertices or removes existing ones for three reasons: to guarantee that the mesh boundary is close to the object surface, to remove tetrahedra or facets with large radius-edge ratio, and to satisfy

the sizing requirements. Specifically, let t be a tetrahedron and f a facet in $\mathcal{D}(V)$; the following five rules are checked in this order:

- **R1**: Let t be an intersecting tetrahedron and z be equal to the Closest Feature Point $\mathrm{cfp}(c(t))$ of t's circumcenter $c(t)$. If z is at a distance not closer than $\Delta_{\partial\Omega}(z)$ to any other feature vertex, then z is inserted and all the free vertices closer than $2\Delta_{\partial\Omega}(z)$ to z are deleted.

- **R2**: Let t be an intersecting tetrahedron and z be equal to $\mathrm{cfp}(c(t))$. If $r(t) \geq 2 \cdot \Delta_{\partial\Omega}(z)$, then $c(t)$ is inserted.

- **R3**: Let f be a restricted facet. If either $\rho(f) \geq \bar{\rho}_f$ or a vertex of f is not a feature vertex, then $z = c_{\mathrm{surf}}(f)$ is inserted. All the free vertices closer than $2\Delta_{\partial\Omega}(z)$ to z are deleted.

- **R4**: If t is an interior tetrahedron whose radius-edge ratio is larger than or equal to $\bar{\rho}_t$, then $c(t)$ is inserted.

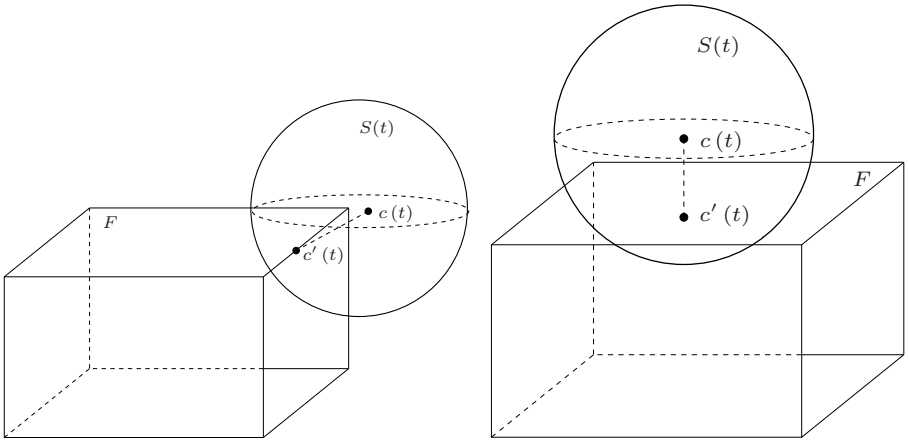
- **R5**: Let t be an interior tetrahedron. If $|r(t)| \geq \mathrm{sf}(c(t))$, where $\mathrm{sf}(\cdot)$ is the user-defined Size Function, then $c(t)$ is inserted.

Whenever there is no simplex for which R1, R2, R3, R4, or R5 apply, the refinement process terminates. *The final mesh reported is the set of tetrahedra whose circumcenters lie inside Ω (i.e., interior tetrahedra).* Thereafter, the final mesh is denoted by $\mathcal{M}$.

Definition 6 (Mesh boundary)**.** *Let f be a facet of the final mesh $\mathcal{M}$. Consider its two incident tetrahedra. If one tetrahedron has a circumcenter lying inside a tissue Ω_i and the other tetrahedron has a circumcenter lying either outside Ω_i or on $\partial\Omega_i$, then f belongs to the mesh boundary $\partial\mathcal{M}$.*

In Section 6, we prove that $\partial\mathcal{M}$ meshes all multi-tissue interfaces $\bigcup_i^n \partial\Omega_i$ ($= \partial\Omega$, see Section 2) accurately, in both geometric and topological sense.

To prove termination (see Section 4), no vertices should be inserted outside the bounding box. Notice, however, that vertices inserted due to R2 may lie outside the bounding box. To deal with such cases, we propose special **projections rules**. Their goal is to reject points lying outside the box and insert other points exactly on the box. They are simple to implement, computationally inexpensive, and do not compromise either quality or fidelity.



(a) $c'(t)$ is a box-edge point. (b) $c'(t)$ is a non-box-edge point.

Figure 1: The projection rule. The circumcenter $c(t)$ of a tetrahedron t (not shown) does not lie inside the box. $c(t)$ is rejected for insertion; rather, its projection $c'(t)$ precisely on the box is computed and inserted into the triangulation.

Note that the projection rules are different than the traditional *encroachment rules* described in [1, 2, 27].

Specifically, assume that R2 is triggered for a (intersecting) tetrahedron t and $c(t)$ lies outside the box. In that case, $c(t)$ is rejected for insertion. Instead, its projection $c'(t)$ on the box is inserted in the triangulation. That is, $c'(t)$ is the closest to $c(t)$ box point. Notice that $c'(t)$ can either lie exactly on a box edge (see Figure 1a) or in the interior of a box facet (see Figure 1b).

Recall that tetrahedra with circumcenters on $\partial\Omega$ or outside Ω are not part of the final mesh, and that is why rules R4 and R5 do not check them.

Algorithm 1 summarizes our mesh generation algorithm. Observe that at line 10, we ask for the closest feature point $\mathrm{cfp}(c)$ of a given circumcenter c. Also, given a feature point $z \in \partial\Omega$, the algorithm asks for its distance $\mathrm{lfs}_{\partial\Omega}(z)$ from the medial axis. The computation of $\mathrm{cfp}(\cdot)$ and $\mathrm{lfs}_{\partial\Omega}(\cdot)$ is explained in detail in Section 7. In the next section, we will prove that *Intersecting* $\cup$ *Interior* eventually will run out of elements, and the algorithm terminates.

Algorithm 1: The mesh generation algorithm.

Algorithm: Refine($\mathcal{I}$, δ, $\bar{\rho}_t$, $\bar{\rho}_f$, $sf(\cdot)$)

Input : $\mathcal{I}$ is the image containing Ω,
δ is the parameter that determines how dense the surface sampling will be,
$\bar{\rho}_t$ ($\geq \sqrt{\sqrt{3}+2}$) is the target radius-edge ratio for the tetrahedra,
$\bar{\rho}_f$ (≥ 1) is the target radius-edge ratio for the facets,
$\mathrm{sf}(\cdot)$ is the size function.

Output: A Delaunay mesh $\mathcal{M}$ that approximates $\partial\Omega$ well and is composed of tetrahedra with radius-edge ratio less than $\bar{\rho}_t$ and boundary facets with planar angles larger than 30°.

Let V be the set of vertices inserted into the triangulation;
Let $\mathcal{D}(V)$ be the triangulation of the set V;
Let *Intersecting* and *Interior* be the set of the intersecting and interior tetrahedra in $\mathcal{D}(V)$, respectively;
/* At this point, all the above sets are equal to the empty set. */
Insert the 8 vertices of a cubical box which contains Ω such that any point inserted on the box is separated from any point $z \in \partial\Omega$ by a distance of at least $2 \cdot \delta \cdot \mathrm{lfs}_{\partial\Omega}(z)$;
Update V, $\mathcal{D}(V)$, *Intersecting*, and *Interior*;
while *Intersecting* $\cup$ *Interior* $\neq \emptyset$ **do**
    if *Intersecting* $\neq \emptyset$ **then**
        Pick a tetrahedron $t \in$ *Intersecting*;
        Compute *steiner* $= \mathrm{cfp}(c(t))$;
        if *there is no feature vertex closer than* $\delta \cdot lfs_{\partial\Omega}$(*steiner*) *to steiner* **then**
            /* R1 applies. */
        else
            if $r(t) \geq 2 \cdot \delta \cdot lfs_{\partial\Omega}$(*steiner*) **then**
                Compute *steiner* $= c(t)$;
                /* R2 applies. */
                if *steiner lies outside the box* **then**
                    Compute *steiner* $= c'(t)$;
                    /* Projection rules apply. */
                end
            else
                if *t is adjacent to a restricted facet f, such that* $\rho(f) \geq \bar{\rho}_f$ *or f's vertices do not lie on* $\partial\Omega$ **then** /* Since f is a restricted facet, f is necessarily incident to at least one intersecting tetrahedron t. */
                    Compute *steiner* $= c_{\mathrm{surf}}(f)$;
                    /* R3 applies. */
                else
                    Intersecting $=$ *Intersecting* $\setminus t$;
                    /* No steiner point found. */
                    continue;
                end
            end
        end
    else
        /* *Interior* cannot be empty. */
        if $\rho(t) \geq \bar{\rho}_t$ **or** $r(t) \geq sf(c(t))$ **then**
            Compute *steiner* $= c(t)$;
            /* R4 or R5 apply. */
        else
            Interior $=$ *Interior* $\setminus t$;
            /* No steiner point found. */
            continue;
        end
    end
    Insert *steiner*;
    if *steiner is a feature vertex* **then**
        Delete all the free vertices that are closer than $2 \cdot \delta \cdot \mathrm{lfs}_{\partial\Omega}$(*steiner*) to *steiner*.
    end
    Update V, $\mathcal{D}(V)$, *Intersecting*, and *Interior*;
end
Let the final mesh $\mathcal{M}$ be equal to the set of the tetrahedra in $\mathcal{D}(V)$ whose circumcenter lies inside Ω;

4. Proof of Quality

In this section, we prove that if the target upper bound $\bar{\rho}_t$ for the radius-edge ratio is no less than $\sqrt{\sqrt{3}+2}$, then our algorithm terminates outputting tetrahedra with radius-edge ratio less than $\bar{\rho}_t$ and boundary facets with planar angles larger than 30° (see Theorem 2). Note that termination and quality are not compromised by any positive value of δ. Parameter δ affects only the fidelity guarantees (see Section 6).

Suppose that an element (tetrahedron or facet) t violates a rule Ri, where $i = 1, 2, 3, 4, 5$, proj, where Rproj denotes the projection rules. That is, if t violates R2, but its circumcenter lies on or outside the box, then we say that t violates Rproj instead. t is called an Ri element. Ri dictates the insertion of a point p (and possibly the removal of free points). Point p is called an Ri point. Although the initial 8 box corners inserted into the triangulation do not violate any rule, we shall refer to these corners as Rproj vertices as well.

Following similar terminology to [1, 27], we next define the *insertion radius* and the *parent* of a point p.

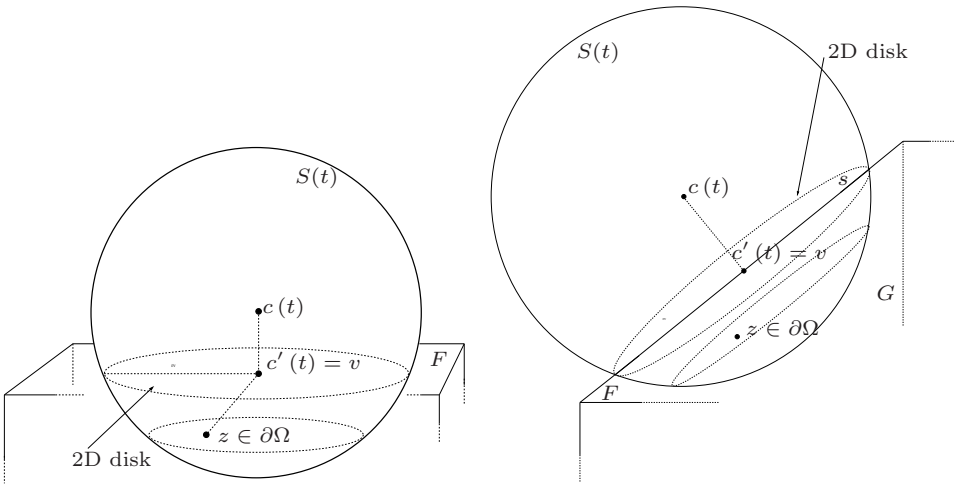
Definition 7 (Insertion radius)**.** *Let v be a vertex inserted into the triangulation. Right after the insertion of v (i.e., before any potential vertex removals), the insertion radius $R(v)$ of v is equal to $|vq|$, where q is:*

- *v's closest box vertex already inserted into the mesh, if v is a box vertex,*
- *v's closest feature vertex already inserted into the mesh, if v is an R1 vertex,*
- *v's closest vertex already inserted into the mesh, otherwise.*

Definition 8 (Parent)**.** *Let v be an Ri vertex inserted into the mesh because an element (tetrahedron or facet) t violated Ri. The parent $Par(v)$ of v is:*

- *an arbitrary box vertex, if t is a facet incident to at least one box vertex,*
- *the most recently inserted vertex of t, if t is a facet with $\rho(f) < \bar{\rho}_f$,*
- *the most recently inserted vertex of $l_{\min}(t)$, otherwise.*

The following two Lemmata relate the insertion radii of a vertex v with the distance between v and its neighbors.



(a) $c'(t)$ is not a box-edge vertex. (b) $c'(t)$ is a box-edge vertex.

Figure 2: Illustration to the proof of Lemma 3.

Lemma 1. *Let w be an R2, R3, R4, or an R5 vertex inserted into the triangulation and let x be an arbitrary vertex already in the triangulation. Then, $R(w) \leq |wx|$.*

Proof. According to Definition 7, $R(w)$ is the distance between w and its closest neighbor, say q. Therefore, $R(w) = |wq| \leq |wx|$. □

Lemma 2. *Let w be an R1 vertex inserted into the triangulation and let x be an arbitrary feature vertex already in the triangulation. Then, $R(w) \leq |wx|$.*

Proof. According to Definition 7, $R(w)$ is the distance between w and its closest feature vertex, say q. Therefore, $R(w) = |wq| \leq |wx|$. □

Lemma 3. *Let v be a box vertex. Then, $R(v) \geq 2\Delta_{\partial\Omega}(z)$, where z is the closest feature point of v.*

Proof. According to Definition 7, $R(v)$ is the distance between v and its closest box vertex.

Initially, only the 8 box vertices of the bounding box are inserted. By construction, no matter the order they are inserted, no box point is closer

than $2\delta \text{lfs}^{\text{sup}}_{\partial\Omega} \geq 2\delta \text{lfs}_{\partial\Omega}(z)$ for any $z \in \partial\Omega$. Therefore, the initial edges are definitely larger than $4\Delta_{\partial\Omega}(z)$ for any $z \in \partial\Omega$, and the statement holds.

During the course of refinement, a box point v is inserted either because the circumcenter $c(t)$ of an intersecting tetrahedron t lies on or outside the box. According to the projection rules, $c(t)$ is ignored, and its projection $c'(t)$ is inserted instead.

See Figure 2 for two examples illustrating the insertion of a non box-edge vertex and a box-edge vertex. In both cases, consider the 2D disk (drawn in both Figure 2a and Figure 2b) of the t's sphere $S(t)$ that contains $c'(t)$ and is perpendicular to the segment $c(t)\,c'(t)$. This disk partitions t's circumball in two parts: the *upper* part that contains $c(t)$ and the *lower* part that intersects the interior of the box. From the empty ball property, we know that the insertion radius of $c'(t)$ cannot be less than the radius of the 2D disk. Let z be the closest feature point to $c'(t)$. Since t is an intersecting tetrahedron, z has to lie in the lower part of t's circumball, which means that $R(c'(t)) \geq |c'(t)\,z|$. By construction, however, $|c'(t)\,z|$ is larger than $2\delta \text{lfs}^{\text{sup}}_{\partial\Omega} \geq 2\Delta_{\partial\Omega}(z)$, and the proof is complete.

□

Lemma 4. *Let v be a box vertex. Then $|v\,cfp(v)| \leq R(v)\sqrt{3}$.*

Proof. If v is a box vertex inserted because the circumcenter of an R2 element lies on or outside the box, then the statement holds, because the proof of Lemma 3 directly suggests that $|\text{cfp}(v)\,v| \leq R(v)$.

Consider the case where the box vertex v is one of the initially inserted 8 box corners. Note that the circumballs of all the resulting tetrahedra are the same with the circumscribed ball of the box. Let us denote with r the length of the radius of that ball. Since that ball contains the whole box, we have that $|v\text{cfp}(v)| \leq 2r$. It is easy to show that $r = \frac{\sqrt{3}}{2}L$, where L is the box's edge length edge. From Definition 7, we know that $R(v) = L$, and therefore, we obtain that $|v\text{cfp}(v)| \leq 2r = L\sqrt{3} = R(v)\sqrt{3}$. □

Lemma 5. *Let f be a facet and wx be its second smallest edge. If $\rho(f) \leq \bar{\rho}_f$, then $|r(f)| \geq \frac{|wx|}{\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}}$.*

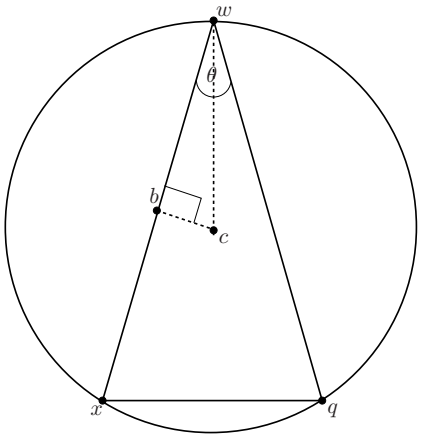


Figure 3: Illustration to the proof of Lemma 5. The facet f is shown in bold together with its circumcircle. The radius of the circumcircle of f is bounded from below by $\frac{|wx|}{\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}}$, where wx is the second smallest edge of f.

Proof. Let f be a facet defined by the vertices w, x, and q. Assume that $xq = l_{\min}(f)$ and wq is the largest edge. If there are more than one smallest edges and/or largest edges, choose arbitrarily one edge as the smallest and/or one edge as the largest. In this example, wx is the second smallest edge of f.

Keeping the circumball fixed, move w along the circumcircle such that the length of wx becomes equal to the length of wq. In such a configuration (which is shown in Figure 3), the length of wx (i.e., the length of the second smallest edge) is maximized. Also observe, the radius wc bisects angle θ. From the right triangle $\triangle cbw$, we get that $\cos\frac{\theta}{2} = \frac{|wx|}{2|r(f)|}$. Since θ is still the angle opposite to f's shortest edge, θ and $\rho(f)$ are related through the following equality: $\sin\theta = \frac{1}{2\rho(f)}$ [27]. This fact and basic trigonometry yield that $\cos\frac{\theta}{2} = \frac{\sqrt{\sqrt{1-\frac{1}{4\rho(f)^2}}+1}}{\sqrt{2}}$. Note that the right-hand side is maximized when $\rho(f)$ gets its highest value. Since the Lemma assumes that $\rho(f) \leq \bar{\rho}_f$, we obtain that $\cos\frac{\theta}{2} \leq \frac{\sqrt{\sqrt{1-\frac{1}{4\bar{\rho}_f^2}}+1}}{\sqrt{2}}$. Therefore, we finally conclude that $|r(f)| \geq \frac{|wx|}{\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}}$. □

The following lemma sets a lower bound on the shortest edge introduced into the mesh after the insertion of a point according to the five rules.

Lemma 6. *Let v be inserted as dictated by the five rules and w be its parent. Then,*

- $R(v) \geq \Delta_{\partial\Omega}(z)$*, if v is an R1 or an R2 vertex, where z is the closest feature point to v,*
- $R(v) \geq \min\left\{\frac{R(w)}{\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}}, \bar{\rho}_f R(w), \Delta_{\partial\Omega}(v)\right\}$*, if v is an R3 vertex and w is a free or a box vertex,*
- $R(v) \geq \min\left\{\bar{\rho}_f R(w), \Delta_{\partial\Omega}(w)\right\}$*, if v is an R3 vertex and w is a feature vertex,*
- $R(v) \geq \bar{\rho}_t R(w)$*, if v is an R4 vertex,*
- $R(v) \geq sf(v)$*, if v is an R5 vertex.*

Proof. We separate cases according to the type of v.

- **Case 1:** v is an R1 or an R2 vertex.

  If R1 is triggered, then v is equal to z. According to Definition 7, $R(v)$ is the distance between v and its closest feature vertex. Vertex v, however, is inserted only if v is separated from any other feature vertex by a distance of at least $\Delta_{\partial\Omega}(v) = R(v)$, and the statement holds.

  Otherwise, R2 applies for a tetrahedron t and v is equal to $c(t)$. According to Definition 7, $R(v)$ is the distance between v and its closest neighbor. Because of the empty ball property, $R(v)$ is at least $r(t) \geq 2\Delta_{\partial\Omega}(\mathrm{cfp}(v))$, and the statement holds.

- **Case 2:** v is an R3 vertex.

  In this case, v is equal to the center $c_{\mathrm{surf}}(f)$ of f's surface ball, where f is the restricted facet that violates R3. According to Definition 7, $R(v)$ is the distance between v and its closest neighbor. Since any surface ball is empty of vertices in its interior (Remark 1), we know that $R(v) = |r_{\mathrm{surf}}(f)|$. The rest of this proof attempts to bound $|r_{\mathrm{surf}}(f)|$ from below. We separate three scenaria:

(a) First, consider the case where f is incident to at least on box vertex. According to Definition 8, this box vertex can be the parent w of v. By construction, the distance between $c_{\text{surf}}(f)$ and w is at least $2\Delta_{\partial\Omega}(z)$ for any feature point z. Therefore, the surface radius is at least $2\Delta_{\partial\Omega}(v) > \Delta_{\partial\Omega}(v)$, and the statement holds.

(b) Second, consider the case where $\rho(f) \geq \bar{\rho}_f$. According to Definition 8, w is the most recently inserted vertex incident to $wq = l_{\min}(f)$. Since $\rho(f)$ is no less than $\bar{\rho}_f$, $|r(f)| = \rho(f)\,|l_{\min}(f)| \geq \bar{\rho}_f\,|l_{\min}(f)|$.

If w is not an R1 vertex, from Lemma 1, we get that $|r(f)| \geq \bar{\rho}_f\,|l_{\min}(f)| \geq \bar{\rho}_f R(w)$.

If w is an R1 vertex and q is a feature vertex (that is, q is either an R1 or an R3 vertex), then from Lemma 2, we get that $|r(f)| \geq \bar{\rho}_f\,|l_{\min}(f)| \geq \bar{\rho}_f R(w)$.

If w is an R1 vertex and q is a free vertex then q has to be separated from w by a distance of at least $2\Delta_{\partial\Omega}(w)$, because R1 deleted all the free points closer than $2\Delta_{\partial\Omega}(w)$ to w. That means that $|wq| \geq 2\Delta_{\partial\Omega}(w)$, a fact that also bounds $|r(t)|$ from below by $\frac{|wq|}{2} \geq \Delta_{\partial\Omega}(w)$.

(c) Lastly, consider the case where f's radius-edge ratio is less than $\bar{\rho}_f$. Since R3 is triggered, f has to be incident to at least one free vertex q. According to Definition 8, w is the most recently inserted vertex of f. If w is a feature vertex (i.e., w is either an R1 or an R3 vertex), then q must be separated from w by a distance of at least $2\Delta_{\partial\Omega}(w)$, because w was inserted after q, and by R1 and R3, all the free points closer than $2\Delta_{\partial\Omega}(w)$ to w were deleted. Since wq is an edge of f, the radius of any surface ball of f has to be at least $\frac{2\Delta_{\partial\Omega}(w)}{2} = \Delta_{\partial\Omega}(w)$ in length. Otherwise, w is in fact a free vertex (i.e., it is an R2, R4, or R5 vertex).

Any vertex w of f is incident to f's shortest edge (say L_1), or f's second shortest edge (say L_2), or both. From Lemma 5, we have that $|r(f)| \geq \frac{|L_2|}{\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}} \geq \frac{|L_1|}{\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}}$. From Lemma 1, we finally get that: $|r(f)| \geq \frac{R(w)}{\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}}$.

- **Case 3:** v is an R4 vertex.

There has to be a tetrahedron t that violates R4, and therefore, $|r(t)| \geq \bar{\rho}_t |l_{\min}(t)|$. According to Definition 7, $R(v)$ is the distance between v and its closest neighbor. Because of the empty ball property, $R(v) = |r(t)| \geq \bar{\rho}_t |l_{\min}(t)|$. According to Definition 8, the parent w of v is the most recently inserted vertex of $wq = l_{\min}(t)$.

If w is not an R1 vertex, from Lemma 1, we get that $\bar{\rho}_t |l_{\min}(t)| \geq \bar{\rho}_t R(w)$.

If w is an R1 vertex and q is a feature vertex (that is, q is either an R1 or an R3 vertex), then from from Lemma 2, we get that $\bar{\rho}_t |l_{\min}(t)| \geq \bar{\rho}_t R(w)$.

If w is an R1 vertex and q is a free vertex then q has to be separated from w by a distance of at least $2\Delta_{\partial\Omega}(w)$, because R1 deleted all the free points closer than $2\Delta_{\partial\Omega}(w)$ to w. That means that $|wq| \geq 2\Delta_{\partial\Omega}(w)$, a fact that also bounds $|r(t)|$ from below by $\frac{|wq|}{2} \geq \Delta_{\partial\Omega}(w)$.

- **Case 4:** v is an R5 vertex.

  v is the circumcenter of a tetrahedron t with radius no less than $\mathrm{sf}(c(t)) = \mathrm{sf}(v)$. According to Definition 7 and the empty ball property, however, the radius of t is equal to $R(v)$.

□

The next Lemma shows that the boundary facets of the output mesh are in fact restricted facets.

Lemma 7. *Let V be the set of vertices inserted into the triangulation. The set $\partial\mathcal{M}$ of the boundary facets of the final mesh $\mathcal{M}$ is a subset of $\mathcal{D}_{|\partial\Omega}(V)$.*

Proof. It follows directly from Definition 6. A facet f is a facet of the mesh boundary, if it is incident upon a tetrahedron t_1 whose circumcenter lies inside Ω_i (see Section 2) and upon a tetrahedron t_2 whose circumcenter lies either outside Ω_i or on its surface $\partial\Omega_i$. However, this means that the dual Voronoi edge e of f intersects $\partial\Omega_i$, and as a subsequence, e also intersects $\partial\Omega\,(\supseteq \partial\Omega_i)$. Hence, f belongs to $\mathcal{D}_{|\partial\Omega}(V)$. □

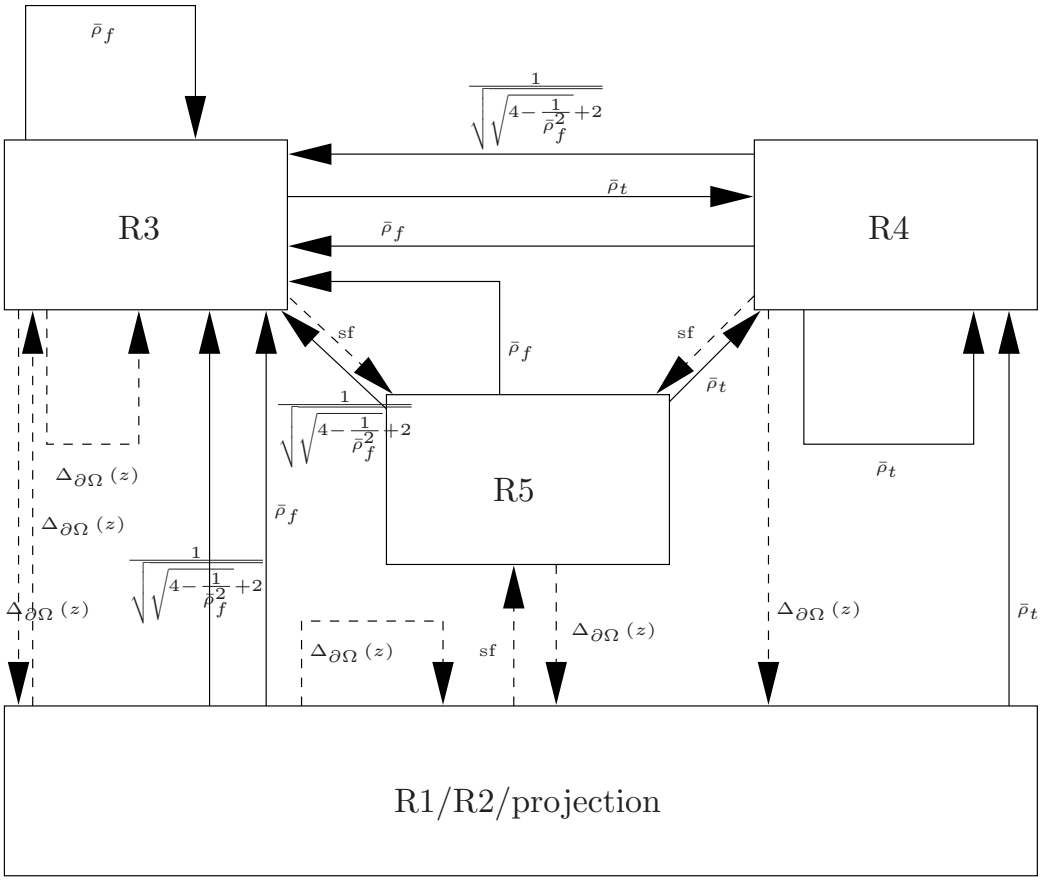


Figure 4: Flow diagram depicting the relationship among the insertion radii of the vertices inserted because of the rules, where the arrows point from parents to their offspring. A solid arrow from Ri to Rj with label x implies that the insertion radius of an Rj point v is at least x times larger than the insertion radius of its Ri parent w. The label of the dashed arrows is the absolute value of $R(v)$. No solid cycle should have a product less than 1. The dashed arrows break the cycle.

Theorem 2 (Termination and quality)**.** *Let* $\bar{\rho}_f \geq 1$ *and let* $\bar{\rho}_t \geq \sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2}\left(\geq \sqrt{\sqrt{3}+2} \approx 1.93\right)$. *The algorithm terminates producing tetrahedra of radius-edge ratio less than* $\bar{\rho}_t$ *and boundary facets of planar angles larger than* $30°$.

Proof. Figure 4 shows the insertion radius of the inserted point as a fraction of the insertion radius of its parent, as proved in Lemma 3 and Lemma 6. An arrow from Ri to Rj with label x implies that the insertion radius of an Rj point v is at least x times larger than the insertion radius of its Ri parent w. The label of the dashed arrows is the absolute value of $R(v)$, with sf denoting that the insertion radius of v is no less than $\mathrm{sf}(v)$. Note that the

labels of the dashed arrows depend on the local feature size of $\partial\Omega$ and the size function sf, and as such are always positive constants.

Recall that during refinement, free vertices might be deleted (because of R1 or R3). Nevertheless, such deletions of vertices are always preceded by insertion of feature points. Considering the fact that feature vertices are never deleted from the mesh, termination is guaranteed if we prove that the insertion radii of the inserted vertices cannot approach zero. Clearly [1, 27], it is enough to prove that Figure 4 contains no solid cycle of product less than 1. By requiring $\bar{\rho}_f$ to be no less than 1 (cycle R3 $\rightarrow$ R3) and $\bar{\rho}_t$ to be no less than $\sqrt{\sqrt{4-\frac{1}{\bar{\rho}_f^2}}+2} \geq \sqrt{\sqrt{3}+2}$ (cycle R3 $\rightarrow$ R4 $\rightarrow$ R3), no solid cycle of Figure 4 has product less than 1, and termination is guaranteed.

Upon termination, the tetrahedra reported as part of the mesh have circumcenters that lie inside Ω and therefore they cannot be skinny, because otherwise R4 would apply. This implies that any mesh tetrahedron has radius-edge ratio less than $\bar{\rho}_t$.

Since a boundary facet f is a restricted facet (by Lemma 7), R3 guarantees that the radius-edge ratio $\rho(f)$ of f's diametral ball cannot be larger than or equal to $\bar{\rho}_f$. Also, note that $\rho(f)$ is equal to $\frac{1}{2\sin\theta}$, where θ is the smallest angle of f [27]. It follows that the planar angles are larger than 30°. □

5. Proof of Good Grading

In this Section, we show that the shortest edge connected to an inserted vertex v is proportional to v's local feature size, a property that is also known as *good grading*. Although good grading implies size optimality in 2D domains [28], the same does not hold in general 3D domains, a mishap associated to the fact that slivers are not provably eliminated. As explained and demonstrated in [1], there are cases such that the local feature size becomes arbitrary close to zero, while there is a way to mesh the domain with very few (and poor) elements. However, infinitesimal local feature size means that even a very large number of elements satisfy the good grading property, despite the fact that many fewer elements would suffice. This is a possible non size-optimal scenario.

Nevertheless, it is useful to show that parts of the domain with large local feature sizes are meshed with larger and fewer elements than parts of smaller local feature sizes. We also wish to show that dense size functions on certain

parts will not affect considerably the density of vertices on other parts of the domain.

Following similar terminology to [3], we define the *general local feature size* and the *general size function* on a vertex v as follows:

Definition 9 (General Local Feature Size)**.** *The general local feature size* $glfs_{\partial\Omega}(v)$ *on a vertex* v *is defined as*

$$glfs_{\partial\Omega}(v) = \inf_{z\in\partial\Omega} \left\{ |vz| + lfs_{\partial\Omega}(z) \right\} \tag{2}$$

Definition 10 (General Size Function)**.** *The general size function* $gsf(v)$ *on a vertex* v *is defined as*

$$gsf(v) = \inf_{p\in\Omega} \left\{ |pv| + sf(p) \right\} \tag{3}$$

The definition of $\mathrm{glfs}_{\partial\Omega}(\cdot)$ implies that vertices far from $\partial\Omega$ will tend to have large general local feature sizes. In the case where the vertex lies on the surface, the general local feature size coincides with the local feature size (see Definition 2) of the vertex, and it increases when the vertex lies far from the medial axis. The definition of $\mathrm{gsf}(\cdot)$ implies that vertices close to parts of the domain on which the user-defined size function is small will evaluate the general size function to a small number as well.

The following Remark states a few useful properties of the general local feature size and the general size function:

Remark 2 (From [23, 29])**.** $glfs_{\partial\Omega}(\cdot)$ *and* $gsf(\cdot)$ *are 1-Lipschitz. Moreover,* $glfs_{\partial\Omega}(z) = lfs_{\partial\Omega}(v)$*,* $\forall z \in \partial\Omega$*.*

Following the terminology of [1, 27], the density $D(v)$ of a vertex v is defined as:

$$D(v) = \frac{\min\{\mathrm{glfs}_{\partial\Omega}(v), \mathrm{gsf}(v)\}}{R(v)} \tag{4}$$

Our goal is to bound from above the density of all inserted vertices by a constant depending only on Ω and the input parameters. Notice that since $D(v) \leq \frac{\mathrm{glfs}_{\partial\Omega}(v)}{R(v)}$ and $D(v) \leq \frac{\mathrm{gsf}(v)}{R(v)}$, it is enough to bound from above either $\frac{\mathrm{glfs}_{\partial\Omega}(v)}{R(v)}$ or $\frac{\mathrm{gsf}(v)}{R(v)}$.

The following Lemma relates the insertion radius of a vertex with its distance from its parent:

Lemma 8. *Let v be an R3 or an R4 vertex inserted into the mesh and w be its parent. Then, $R(v) = |vw|$.*

Proof. If v is an R3 or R4 vertex, then v is the center of an element t's circumball or surface ball. According to Definition 8, the parent w of v is one vertex of t. Because of the empty circumball and surface ball property, Definition 7 implies that $R(v) = |vw|$. □

The following Lemma relates the density of a vertex v with that of its parent:

Lemma 9. *Let v be an R3 or R4 vertex and $R(v) \geq c \cdot R(w)$, where $w = Par(v)$. Then*

$$D(v) \leq 1 + \frac{D(w)}{c}. \tag{5}$$

Proof. The proof is similar to the proof of Lemma 6 in [27].

Let $w = Par(v)$. Since $\text{glfs}_{\partial\Omega}(\cdot)$ and $\text{gsf}(\cdot)$ are 1-Lipschitz (see Remark 2), we have that:

$$\begin{aligned}
\min\{\text{glfs}_{\partial\Omega}(v), \text{gsf}(v)\} &\leq \min\{|vw| + \text{glfs}_{\partial\Omega}(w), |vw| + \text{gsf}(w)\} \\
&= |vw| + \min\{\text{glfs}_{\partial\Omega}(w), \text{gsf}(w)\} \\
&= R(v) + \min\{\text{glfs}_{\partial\Omega}(w), \text{gsf}(w)\} && \text{(from Lemma 8)} \\
&= R(v) + R(w)D(w) && \text{(from Equation (4))} \\
&\leq R(v) + \tfrac{R(v)}{c} D(w),
\end{aligned}$$

and the result follows by dividing both sides by $R(v)$. □

Before we proceed to the proof of good grading, we need two auxiliary Lemmata:

Lemma 10. *Let v be an R2 vertex. Then, $|v\,cfp(v)| \leq R(v)$.*

Proof. Vertex v is an R2 vertex because of an intersecting tetrahedron t. Since t is an intersecting tetrahedron and v is the center of t, we have that $|v\text{cfp}(v)| \leq |r(t)|$. Definition 7, however, implies that $|r(t)| = R(v)$, and the statement holds. □

Lemma 11. *Let v be a vertex inserted into the mesh. Then,*

- $D(v) \leq \frac{1+\delta\sqrt{3}}{\delta}$*, if v is an R1, R2, or a box vertex,*
- $D(v) \leq \frac{1+\delta}{\delta}$*, if v is an R3 vertex and $R(v) \geq \min\{\Delta_{\partial\Omega}(w), \Delta_{\partial\Omega}(v)\}$, where $w = Par(v) \in \partial\Omega$, and*

- $D(v) \leq 1$, *if* v *is an R5 vertex.*

Proof. We separate cases.

Let v be an R1 vertex. According to the flow diagram of Figure 4, we have that $R(v) \geq \Delta_{\partial\Omega}(\text{cfp}(v)) = \Delta_{\partial\Omega}(v) = \delta \cdot \text{lfs}_{\partial\Omega}(z)$. From Remark 2, we have that $R(v) \geq \delta \cdot \text{lfs}_{\partial\Omega}(z) = \delta \cdot \text{glfs}_{\partial\Omega}(z)$, giving that $D(v) \leq \frac{1}{\delta} \leq \frac{1+\delta\sqrt{3}}{\delta}$, and the statement holds.

Let v be an R2 or a box vertex. According to the flow diagram, $R(v) \geq \Delta_{\partial\Omega}(\text{cfp}(v)) = \delta \cdot \text{lfs}_{\partial\Omega}(\text{cfp}(v)) = \delta \cdot \text{glfs}_{\partial\Omega}(\text{cfp}(v))$, and from the fact that the general local feature size is 1-Lipschitz, we have that $R(v) \geq \delta(\text{glfs}_{\partial\Omega}(v) - |v\text{cfp}(v)|)$. From Lemma 4 and Lemma 10, we know that $|v\text{cfp}(v)| \leq R(v)\sqrt{3}$. Therefore, we obtain that $R(v) \geq \delta\left(\text{glfs}_{\partial\Omega}(v) - R(v)\sqrt{3}\right)$. Dividing both sides by $R(v)$ finally gives that $D(v) \leq \frac{1+\delta\sqrt{3}}{\delta}$, and the statement holds.

Let v be an R3 vertex and $R(v) \geq \Delta_{\partial\Omega}(v) = \delta \cdot \text{lfs}_{\partial\Omega}(v)$. It follows directly that $D(v) \leq \frac{1}{\delta} < \frac{1+\delta}{\delta}$.

Let v be an R3 vertex and $R(v) \geq \Delta_{\partial\Omega}(w)$, where $w \in \partial\Omega$ is the parent of v. From Remark 2, we obtain that $R(v) \geq \Delta_{\partial\Omega}(w) = \delta \cdot \text{lfs}_{\partial\Omega}(w) = \delta \cdot \text{glfs}_{\partial\Omega}(w) \geq \delta(\text{glfs}_{\partial\Omega}(v) - |vw|)$. From Definition 8, w is one of the vertices of a restricted facet whose surface ball has v as the center. From the empty surface ball property and Definition 7, we know that $R(v) = |vw|$. Therefore, $R(v) \geq \delta(\text{glfs}_{\partial\Omega}(v) - R(v))$. Dividing both sides by $R(v)$ finally gives that $D(v) \leq \frac{1+\delta}{\delta}$, and the statement holds.

Let v be an R5 vertex. According to the flow diagram, all the arrows pointing to R5 are dashed and labeled as sf. The label of dashed arrows is the absolute value of $R(v)$ and therefore, $R(v) \geq \text{sf}(v)$. Since, however, $\text{gsf}(v) = \inf_{p \in \Omega} \{ |pv| + \text{sf}(p) \} \leq |vv| + \text{sf}(v) = \text{sf}(v)$, we get that $R(v) \geq \text{gsf}(v)$, and the proof is complete. □

Finally, the following Theorem proves that our algorithm achieves good grading:

Theorem 3 (Good Grading)**.** *Let* $\bar{\rho}_f$ *be strictly larger than* 1 *and let* $\bar{\rho}_t$ *be strictly larger than* $X = \sqrt{\sqrt{4 - \frac{1}{\bar{\rho}_f^2}} + 2}\left(\geq \sqrt{\sqrt{3}+2} \approx 1.93\right)$*. Let* v *be an Ri vertex inserted into the mesh,* $i = 1, 2, 3, 4, 5$*, proj. Then, right after its insertion, its density* $D(v)$ *is bounded from above by a fixed constant* $D_i > 0$*.*

Proof. This theorem will be proved via induction.

Initially, only the 8 box corners are inserted into the triangulation. According to Lemma 11, the induction basis holds, if

$$D_{\text{proj}} = \frac{1+\delta\sqrt{3}}{\delta}. \tag{6}$$

For the induction hypothesis, assume that the density $D(w)$ of v's parent R_j vertex w is bounded from above by D_j, where $j = 1, 2, 3, 4, 5, \text{proj}$. We need to show that one constant D_i bounds from above the density of R_i vertex v, where $i = 1, 2, 3, 4, 5, \text{proj}$.

We separate cases according to the type of v:

- v **is an R1, R2, or a box vertex**
  According to Lemma 11, the insertion radius of v is bounded from above by $\frac{1+\delta\sqrt{3}}{\delta}$. Therefore, no matter what the parent of v is, the induction step holds, if

$$D_1 = D_2 = D_{\text{proj}} = \frac{1+\delta\sqrt{3}}{\delta}. \tag{7}$$

- v **is an R5 vertex**
  Similarly to the case above, Lemma 11 suggests that no matter what the parent of v is, the induction step holds, if

$$D_5 = 1. \tag{8}$$

- v **is an R4 vertex**
  From the flow diagram, all the arrows pointing to R4 are labeled with $\bar{\rho}_t$. Therefore, from Lemma 9 and Lemma 6, we get that $D(v) \leq 1 + \frac{D(w)}{c}$, with c equal to $\bar{\rho}_t$ for any parent w. Thus, the induction step would be proved, if D_4 was set to a value that satisfied all the following inequalities:

$$1 + \frac{D_1}{\bar{\rho}_t} \leq D_4 \tag{9}$$

$$1 + \frac{D_3}{\bar{\rho}_t} \leq D_4 \tag{10}$$

$$1 + \frac{D_4}{\bar{\rho}_t} \leq D_4 \tag{11}$$

$$1 + \frac{D_5}{\bar{\rho}_t} = 1 + \frac{1}{\bar{\rho}_t} \leq D_4 \tag{12}$$

Observe that the D_5 term in Inequality (12) is replaced by 1, according to Equality (8).

- v **is an R3 vertex**
  According to Lemma 11, $D(v)$ is bounded from above by $\frac{1+\delta}{\delta}$ for the relationships of Figure 4 that are depicted by the dashed arrows pointing to R3. Therefore, for the induction step to be proved, D_3 has to satisfy at least the following inequality:

$$\frac{1+\delta}{\delta} \leq D_3 \tag{13}$$

For the rest of the relationships (i.e., solid arrows), we know from Lemma 9 and Lemma 6 that $D(v) \leq 1 + \frac{D(w)}{c}$, where c is equal to $\bar{\rho}_f$ if w is an R3 vertex or equal to $\min\left\{\frac{1}{X}, \bar{\rho}_f\right\}$ if w is an R1, R2, Rproj, R4 or R5 vertex.

Therefore, the induction step would be proved, if D_3 was set to a value that satisfied also the following inequalities:

$$1 + D_1 \max\left\{X, \frac{1}{\bar{\rho}_f}\right\} = 1 + D_1 X \leq D_3 \tag{14}$$

$$1 + \frac{D_3}{\bar{\rho}_f} \leq D_3 \tag{15}$$

$$1 + D_4 \max\left\{X, \frac{1}{\bar{\rho}_f}\right\} = 1 + D_4 X \leq D_3 \tag{16}$$

$$1 + D_5 \max\left\{X, \frac{1}{\bar{\rho}_f}\right\} = 1 + X \leq D_3 \tag{17}$$

Observe that X is always larger than $\frac{1}{\bar{\rho}_f}$ when $\bar{\rho}_f > 1$ and that is why the $\frac{1}{\bar{\rho}_f}$ term is eliminated from Inequalities (14), (16), and (17). Also, the D_5 term in Inequality (17) is replaced by 1, according to Equality (8).

Putting it all together and simplifying the results, Inequalities (9)- (17) above are simultaneously satisfied by choosing:

$$D_4 = \max\left\{\frac{\bar{\rho}_t + 1}{\bar{\rho}_t - X}, \frac{\delta\left(\bar{\rho}_t + 1\right) + X\left(1 + \delta\sqrt{3}\right)}{\delta\bar{\rho}_t}, \frac{\bar{\rho}_t\left(\bar{\rho}_f - 1\right) + \bar{\rho}_f}{\bar{\rho}_t\left(\bar{\rho}_f - 1\right)}\right\} \tag{18}$$

and

$$D_3 = \max\left\{\frac{\delta + X\left(1 + \delta\sqrt{3}\right)}{\delta}, \frac{\delta\bar{\rho}_t\left(1 + X\right) + X\left(1 + \delta\sqrt{3}\right)}{\delta\bar{\rho}_t}, \frac{\bar{\rho}_f}{\bar{\rho}_f - 1}, \frac{\bar{\rho}_t\left(1 + X\right)}{\bar{\rho}_t - X}\right\} \tag{19}$$

Equalities (6), (7), (8), (18), and (19) satisfy both the induction basis and the induction step for any number and type of vertices, and therefore, the proof is complete.

□

6. Proof of Fidelity

In this section, we derive an upper bound for δ, such that the boundary of the final mesh is a provably good topological and geometric approximation of $\partial\Omega$. Our goal is to prove that the mesh boundary $\partial\mathcal{M}$ (see Definition 6) is equal to $\mathcal{D}_{|\partial\Omega}(E)$ for E a 0.09-sample of $\partial\Omega$ (see Theorem 4 of this section). To see why this is enough, recall that from Theorem 1, the restriction of a 0.09-sample of $\partial\Omega$ to $\partial\Omega$ is a good topological and geometric approximation of $\partial\Omega$.

First, we show that δ directly controls the density of the feature vertices. Let V be the set of vertices in the triangulation and E be equal to $V \cap \partial\Omega$.

Lemma 12. *Let $\delta < \frac{1}{4}$. Then E is a $\frac{5\delta}{1-4\delta}$-sample of $\partial\Omega$.*

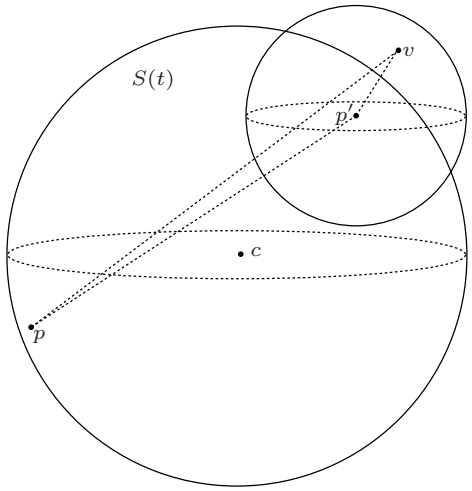


Figure 5: Illustration to the proof of Lemma 12.

Proof. Recall that upon termination, there is no tetrahedron for which R1, R2, R3, R4, or R5 apply.

See Figure 5. Let p be an arbitrary point on $\partial\Omega$. Since $\mathcal{D}(V)$ covers all the domain, point p has to lie on or inside the circumsphere of a tetrahedron t (not shown). Hence, t is an intersecting tetrahedron. Let point p' be the feature point closest to $c(t)$. Note that $|c(t)\,p| \geq |c(t)\,p'|$ and therefore p' lies on or inside t's circumsphere. We also know that t's circumradius has to be less than $2\Delta_{\partial\Omega}(p')$, since otherwise R2 would apply for t. Therefore, we have the following:

$$\begin{array}{rcll} |pp'| & < & 2r(t) & \text{(because both } p \text{ and } p' \text{ lie on or inside } B(t)) \\ & < & 4\Delta_{\partial\Omega}(p') & \text{(because of R2)} \\ & \leq & 4\delta\left(|pp'| + \mathrm{lfs}_{\partial\Omega}(p)\right) & \text{(from Inequality (1)),} \end{array}$$

and by reordering the terms, we obtain that:

$$|pp'| < \frac{4\delta}{1-4\delta}\mathrm{lfs}_{\partial\Omega}(p), \text{ with } \delta < \frac{1}{4}. \tag{20}$$

Moreover, there must exist a feature vertex v in the triangulation closer than $\Delta_{\partial\Omega}(p') = \delta \cdot \mathrm{lfs}_{\partial\Omega}(p')$ to p', since otherwise R1 would apply for t. Hence, $|vp'| < \delta \cdot \mathrm{lfs}_{\partial\Omega}(p')$, and using Inequality (1), we have that:

$$|vp'| < \delta\left(|pp'| + \mathrm{lfs}_{\partial\Omega}(p)\right) \tag{21}$$

Applying the triangle inequality for $\triangle pvp'$ yields the following:

$$\begin{aligned}
|pv| &\leq |pp'| + |vp'| \\
&< |pp'| + \delta\left(|pp'| + \mathrm{lfs}_{\partial\Omega}(p)\right) && \text{(from Inequality (21))} \\
&= |pp'|(1+\delta) + \delta \cdot \mathrm{lfs}_{\partial\Omega}(p) \\
&< \tfrac{4\delta}{1-4\delta}\mathrm{lfs}_{\partial\Omega}(p)(1+\delta) + \delta \cdot \mathrm{lfs}_{\partial\Omega}(p) && \text{(from Inequality (20))} \\
&= \left(\tfrac{4\delta(1+\delta)}{1-4\delta} + \delta\right)\mathrm{lfs}_{\partial\Omega}(p) \\
&= \tfrac{5\delta}{1-4\delta}\mathrm{lfs}_{\partial\Omega}(p),
\end{aligned}$$

and the proof is complete. □

Recall from Section 2 that the multi-tissue object Ω could be described as a union of materials $\Omega = \bigcup_{i}^{n} \Omega_i$. Let us denote by Ω_i^j, the j^{th} connected component of a specific tissue Ω_i, $j = 1, \ldots, m$.

Similar to Definition 4, $\mathcal{D}_{|\partial\Omega_i^j}(V)$ denotes the set of those facets in the Delaunay triangulation of the vertices in V whose dual Voronoi edge intersects the surface $\partial\Omega_i^j$ of Ω_i^j. Also, note that $\partial\Omega = \bigcup_{i}^{n}\bigcup_{j}^{m} \partial\Omega_i^j$.

From Lemma 12 and Definition 3, the following Corollary follows:

Corollary 1. *Let $\delta \leq \frac{0.09}{5.36} \approx 0.0168$ and let $E_i^j = V \cap \partial\Omega_i^j$. Then, E_i^j is a 0.09-sample of $\partial\Omega_i^j$.*

As we have already mentioned in Section 3, the final mesh $\mathcal{M}$ reported consists of tetrahedra whose circumcenter lies inside Ω. Let $\mathcal{M}_i^j$ be the set of tetrahedra whose circumcenter lies inside Ω_i^j.

Similar to Definition 6, $\partial\mathcal{M}_i^j$ denotes the set of the boundary facets of submesh $\mathcal{M}_i^j$. That is, $\partial\mathcal{M}_i^j$ contains the facets incident to two tetrahedra such that one tetrahedron has a circumcenter lying inside Ω_i^j and the other has a circumcenter lying either outside Ω_i^j or on $\partial\Omega_i^j$.

Lemma 13. *Let t be an intersecting tetrahedron whose circumball $B(t)$ contains a point m of $\partial\Omega$'s medial axis. Then, $\delta > \frac{1}{4}$.*

Proof. Upon termination, rule R2 cannot apply for any tetrahedron. There-

fore, we have the following:

$$\begin{array}{rcll} 2 \cdot \delta \cdot \mathrm{lfs}_{\partial\Omega}\left(\mathrm{cfp}\left(c\left(t\right)\right)\right) & > & |r(t)| & \text{(from R2)} \\ & \geq & \frac{|\mathrm{cfp}(c(t))m|}{2} & \text{(since } m \text{ and } \mathrm{cfp}\left(c\left(t\right)\right) \text{ lie inside } B(t)\text{)} \\ & \geq & \frac{\mathrm{lfs}_{\partial\Omega}(\mathrm{cfp}(c(t)))}{2} & \text{(since } m \text{ is on the medial axis)} \\ \delta & > & \frac{1}{4}. & \end{array} \Rightarrow$$

□

Lemma 14. *Let $\delta \leq \frac{1}{4}$. Any facet $f \in \partial\mathcal{M}_i^j$ belongs to $\mathcal{D}_{|\partial\Omega_i^j}\left(V\right)$ and has its vertices on $\partial\Omega_i^j$.*

Proof. Since f belongs to $\partial\mathcal{M}_i^j$, f is incident to two tetrahedra $t_1, t_2 \in \mathcal{D}\left(V\right)$, such that the circumcenter of t_1 lies inside Ω_i^j and the circumcenter of t_2 lies outside Ω_i^j or $\partial\Omega_i^j$. However, this means that the Voronoi edge of f intersects $\partial\Omega_i^j$, and therefore, $f \in \mathcal{D}_{|\partial\Omega_i^j}\left(V\right)$. This completes the first part.

For the second part and for the sake of contradiction, assume that there is at least one vertex v of f that does not lie on $\partial\Omega_i^j$, but on another $\partial\Omega_{i'}^{j'}$. Consider the tetrahedron t_1, one of the two tetrahedra incident to f with circumcenter lying inside Ω_i^j. Since v lies on $\partial\Omega_{i'}^{j'}$, the circumball $B(t_1)$ of t_1 intersects $\partial\Omega$ in more than one connected component. According to Lemma 7 of Amenta and Bern [17], this implies that $B(t)$ contains a point m of the medial axis of $\partial\Omega$. Moreover, observe that t_1 is in fact an intersecting tetrahedron. From Lemma 13, we finally get that $\delta > \frac{1}{4}$. However, this raises a contradiction, since δ is assumed to be no larger than $\frac{1}{4}$. □

The next two Lemmas prove a few useful properties for the mesh $\mathcal{M}$ and its boundary $\partial\mathcal{M}$. Our goal is to show that $\partial\mathcal{M}_i^j$ is always non-empty and does not have boundary (Lemma 16), a fact that will be used for proving the fidelity guarantees (Theorem 4).

Lemma 15. *Let $\delta \leq \frac{1}{4}$. Then, $\mathcal{M}_i^j \neq \emptyset$.*

Proof. For the sake of contradiction, assume that $\mathcal{M}_i^j$ is empty. That means that there is no tetrahedron whose circumcenter lies inside Ω_i^j. Since the triangulation $\mathcal{D}\left(V\right)$ covers all the domain, the circumballs of the tetrahedra in $\mathcal{D}\left(V\right)$ also cover the tissue Ω_i^j. Therefore, there has to be a circumball $B(t)$ ($t \in \mathcal{D}\left(V\right)$) which contains a point m on the medial axis of $\partial\Omega_i^j$, such

that m lies inside Ω_i^j. By our assumption, the circumcenter $c(t)$ cannot lie inside Ω_i^j. Therefore, t is an intersecting tetrahedron. From Lemma 13, we finally get that $\delta > \frac{1}{4}$. However, this raises a contradiction, since δ is assumed to be no larger than $\frac{1}{4}$. □

Lemma 16. *Let $\delta \leq \frac{1}{4}$. Then $\partial\mathcal{M}_i^j$ is a non-empty set and does not have boundary.*

Proof. The fact that $\partial\mathcal{M}_i^j$ is a non-empty set follows directly from Lemma 15: since $\mathcal{M}_i^j$ cannot be empty, its boundary $\partial\mathcal{M}_i^j$ cannot be empty too. For the other part, since $\partial\mathcal{M}_i^j$ is the boundary of a set of tetrahedra, it cannot have boundary.

□

The following Theorem proves the fidelity guarantees achieved by our algorithm:

Theorem 4. *Let $\delta = 0.0168$. Then the mesh boundary $\partial\mathcal{M}$ is a 2-manifold ambient isotopic to $\partial\Omega$ and the 2-sided Hausdorff distance between the mesh boundary and $\partial\Omega$ is $O(\delta^2)$.*

Proof. By Theorem 1, it is enough to prove that $\partial\mathcal{M}$ is the restriction to $\partial\Omega$ of the Delaunay triangulation of a 0.09-sample. We will, in fact, show that the boundary $\partial\mathcal{M}_i^j$ of the submesh $\mathcal{M}_i^j$ is equal to $\mathcal{D}_{|\partial\Omega_i^j}\left(E_i^j\right)$ (recall that E_i^j is equal to $V \cap \partial\Omega_i^j$) which is the restriction to $\partial\Omega_i^j$ of the Delaunay triangulation of a 0.09-sample of $\partial\Omega_i^j$, by Corollary 1. This is enough, since this would prove that the boundary of each submesh $\mathcal{M}_i^j$ is an accurate representation of the interface $\partial\Omega_i^j$, for any i and j.

Let f be a facet in $\partial\mathcal{M}_i^j$. From Lemma 14, we know that $f \in \mathcal{D}_{|\partial\Omega_i^j}(V)$ that f's vertices lie on $\partial\Omega_i^j$. Let B be the surface ball of f. From Definition 5, the interior $\text{int}(B)$ of B is empty of vertices in V. Therefore, $\text{int}(B)$ is empty of vertices in $V \cap \partial\Omega_i^j$ also. Without loss of generality, assume that the vertices in V are in general position. Since there is a ball B that circumscribes f and does not contain vertices of $V \cap \partial\Omega_i^j$ in its interior, f has to appear as a simplex in $\mathcal{D}\left(V \cap \partial\Omega_i^j\right)$. Since the center of B lies on $\partial\Omega_i^j$, then the Voronoi dual of f intersects $\partial\Omega_i^j$ in $\mathcal{D}_{|\partial\Omega}\left(V \cap \partial\Omega_i^j\right)$, as well. Hence, $\partial\mathcal{M}_i^j \subseteq \mathcal{D}_{|\partial\Omega_i^j}\left(V \cap \partial\Omega_i^j\right)$.

For the other direction, we will prove that $\partial\mathcal{M}_i^j$ cannot be a proper subset of $\mathcal{D}_{|\partial\Omega_i^j}\left(V \cap \partial\Omega_i^j\right)$, and therefore, equality between these two sets is forced.

Toward this direction, we will prove that any proper non-empty subset of $\mathcal{D}_{|\partial\Omega_i^j}\left(V \cap \partial\Omega_i^j\right)$ has boundary; this is enough, because we have proved in Lemma 16 that $\partial\mathcal{M}_i^j$ is non-empty and does not have boundary.

Since $V \cap \partial\Omega_i^j$ meets the requirements of Theorem 1, $\mathcal{D}_{|\partial\Omega_i^j}\left(V \cap \partial\Omega_i^j\right)$ is a 2-manifold without boundary. Therefore, any edge in $\mathcal{D}_{|\partial\Omega_i^j}\left(V \cap \partial\Omega_i^j\right)$ is incident to exactly two facets of $\mathcal{D}_{|\partial\Omega_i^j}\left(V \cap \partial\Omega_i^j\right)$. Since any proper non-empty subset $\mathcal{A}$ of $\mathcal{D}_{|\partial\Omega_i^j}\left(V \cap \partial\Omega_i^j\right)$ has fewer facets, $\mathcal{A}$ contains at least an edge e incident to only one facet. However, this implies that e belongs to the boundary of $\mathcal{A}$, and the proof is complete. □

7. Implementation details

We used the *Insight Toolkit* (ITK) for image processing [30]. ITK provides, among others, the implicit function f that describes object Ω to be meshed (see Section 2). Specifically, given a real point p, f returns 0 if the voxel enclosing p is in the background, or it returns the identifier i of the tissue Ω_i if that voxel belongs to Ω_i, $i = 1, \ldots, n$. In order to compute the closest feature point function $\mathrm{cfp}\,(p)$ and identify the cloud of points lying on $\partial\Omega$, we make use of the Euclidean Distance Transform (EDT) as implemented in ITK and presented in [31]. Specifically, the EDT returns the boundary voxel p' which is closest to p. Then, we traverse the ray $\overrightarrow{pp'}$ and we compute the intersection between the ray and $\partial\Omega$ by interpolating the positions where f changes value [20]. The actual mesh generator was built on top of the *Computational Geometry Algorithms Library* (CGAL) [32]. CGAL offers flexible data structures for Delaunay point insertions and removals and robust exact geometric predicates.

The rest of this section describes important implementation aspects.

7.1. Medial Axis Approximation

Recall that rules R1 and R2 make an extensive use of $\mathrm{lfs}_{\partial\Omega}\,(\cdot)$, and therefore, knowledge about the medial axis is needed.

Since the computation of the exact medial axis is a difficult problem [33, 34], we seek a good (for our purposes) approximation of it. Precisely, we are interested in computing $\widetilde{\mathrm{lfs}}_{\partial\Omega}\,(p)$: the approximation of $\mathrm{lfs}_{\partial\Omega}\,(p)$, where $p \in \partial\Omega$.

Remark 3. *In this subsection, we do not alter the fidelity guarantees of Theorem 4, since the theorem assumes that $lfs_{\partial\Omega}(\cdot)$ is known and accurate; in this subsection, we attempt to provide a fast way to approximate $lfs_{\partial\Omega}(\cdot)$.*

For an excellent review of image-based medial axis approximation methods, see the work of Coeurjolly and Montanvert [35]. The authors also describe an optimal algorithm (MAEVA[1]) for the computation of the medial axis, which is a free implementation to download. We found out, however, that although the method is fast, the resulted discrete medial axis was not accurate enough for our purposes. We attribute this behavior to the fact that image-based methods do not realize the underlying shape; they compute the medial axis of volumetric data, which contains discontinuities and thus, renders the computation unstable.

Amenta *et al.* [18] and Dey and Zhao [33] (and the references therein) consider methods that given a set of sample points on the surface, they approximate the medial axis from their Voronoi diagram. Their key concept is the *Pole* of a feature vertex, a technique that we integrate into our algorithm in order to compute $\widetilde{\text{lfs}}_{\partial\Omega}(\cdot)$. Boissonnat and Oudot [19] describe a two-phase algorithm that is able to approximate the medial axis based on the notion of the *Lambda-Medial Axis* [36]. The Lambda-Medial Axis makes weaker assumptions about the sample and as such, it is suitable for noisy data. Nevertheless, we found that the Pole technique is easier to implement and quite robust for our purposes in all the input images we tried. It should also be mentioned that both the Pole and the Lambda-Medial Axis technique focus on surface recovery and not volume meshing. That means that they assume that only vertices on the isosurface are allowed (i.e., the sample). This is not the case in this work, since the quality criteria might dictate the insertion of vertices in the interior of the domain. As we explain below, this difference necessitates the simultaneous maintenance of a second triangulation.

Let E be a vertex set on $\partial\Omega$ and consider the voronoi vertices of the voronoi cell of a feature vertex $v \in E$. The voronoi vertices inside Ω (if any) are called *internal* and the rest (if any) *external*. Amenta *et al.* [18] shows that if E is dense, the internal pole (i.e., the furthest from v internal voronoi vertex) is close to the medial axis contained in Ω, and the external pole (i.e., the furthest from v external voronoi vertex) is close to the medial

[1] `http://liris.cnrs.fr/david.coeurjolly/doku/doku.php?id=code:maeva`

axis contained in the complement of Ω. Therefore, the poles of each sample point form a good discrete approximation of the medial axis.

The problem with the poles (as a good approximation of the medial axis) is that E has to be a dense sample of the surface; however, our algorithm needs the approximation of the medial axis, so it can create a graded sample E. Recall that we do not assume that a starting sample set is known a priori. In fact, when the algorithm starts, there is not a even a single feature vertex inserted into the triangulation. In order to resolve this cyclic dependency, our algorithm alternates between two modes: a "uniform" and a "graded".

Specifically, the algorithm maintains a second triangulation $\mathcal{D}(Z)$ (together with the triangulation $\mathcal{D}(V)$, see Section 3) which contains only feature vertices. To compute $\widetilde{\text{lfs}}_{\partial\Omega}(z)$, z is inserted into the current set of feature vertices Z, and $\mathcal{D}(Z)$ is updated. Next, the poles of z are computed from $\mathcal{D}(Z)$, and the distance from z to its closest pole is returned as the approximation of the distance from z to the medial axis. Clearly, in the early stages of the refinement, Z is a very sparse sample set, and, therefore, the poles of $z \in Z$ are not to be trusted as a good approximation of the medial axis. Note, however, that these poles can only be further from z, than the poles computed at a much denser sample set. In other words, when Z is sparse, $\widetilde{\text{lfs}}_{\partial\Omega}(z)$ gives a larger value than it should (i.e, $\widetilde{\text{lfs}}_{\partial\Omega}(z)$ is larger than $\text{lfs}_{\partial\Omega}(z)$). This has severe consequences, since it is possible for the resulting sample not to be as dense as it should.

For this reason, instead of returning just the value of $\widetilde{\text{lfs}}_{\partial\Omega}(z)$, we choose to return the following quantity: $\min\{\lambda, \widetilde{\text{lfs}}_{\partial\Omega}(z)\}$, where λ will be specified shortly. When $\widetilde{\text{lfs}}_{\partial\Omega}(z)$ is too large (i.e., larger than λ), the value of λ is returned. Parameter λ acts as a safety net and simulates the uniform mode of the algorithm: in the worst case, a uniform sample set will be generated, whose density depends on λ. Note that the uniform mode is triggered mostly in the early stages of the algorithm. Later on, more and more feature vertices are inserted into the triangulation, and the medial axis is sufficiently described by the poles; and this is when the graded mode of the algorithm is activated.

Specifying a value for λ is not intuitive. If λ is small, then the approximation of the medial axis would be more accurate, but the graded mode would be activated fewer times, sacrificing in this way a well-graded surface mesh. On the other hand, if λ is large, then we would expect to see better grading, but it is likely for the approximation of medial axis to be so bad (i.e., it is

likely that $\widetilde{\text{lfs}}_{\partial\Omega}(\cdot)$ is too large), such that the graded mode would fail to capture the curvature of $\partial\Omega$. Nevertheless, extensive experimental evaluation on both synthetic and real medical images has shown that in most cases, setting λ to a value 12 times the size of the voxel suffices.

Note that if the users are not interested in achieving grading along the surface, the second triangulation $\mathcal{D}(Z)$ is not needed at all, since they could define $\widetilde{\text{lfs}}_{\partial\Omega}(p)$ to be simply equal to λ.

7.2. Dihedral angle improvement

Provable theoretical guarantees on the minimum and maximum dihedral angles are outside the scope of this paper. Nevertheless, for practical purposes, we felt that the issue of sliver removal and dihedral angle improvement should be addressed.

We could apply the *sliver exudation* technique [8] in order to improve the dihedral angles. Edelsbrunner and Guoy [37], however, have shown that in most cases sliver exudation does not remove all poor tetrahedra: elements with dihedral angles less than 5° survive. The *random perturbation* technique [12] offers very small guarantees and sometimes requires many (random) trials for the elimination of a single sliver as reported in [16].

A straightforward and inexpensive way to eliminate slivers is to try to split them by inserting their circumcenter. Shewchuk [1] shows that this technique works when the slivers are far away from the mesh boundary. However, when slivers are close to the mesh boundary, the newly inserted points alter the boundary triangles. In fact, the boundary triangles might not have their vertices on the surface any more, or might not even belong to the restricted triangulation. In this subsection, we propose *point rejection strategies* that prevent the insertion of points which hurt fidelity.

Our algorithm first tries to convert *illegal* facets to *legal* ones. We define legal facets to be those restricted facets whose vertices lie precisely on $\partial\Omega$. Conversely, a restricted facet with at least one vertex not lying on $\partial\Omega$ is called an illegal facet.

Let f be an illegal facet and e its voronoi edge (see Figure 6a). Recall that e has to intersect $\partial\Omega$ (see Section 2) at a point p. Any vertex v of f which do not lie precisely on $\partial\Omega$ is deleted from the triangulation, while point p is inserted.

In addition, the algorithm tries to keep in the Delaunay triangulation as many legal facets as possible. Let c be the circumcenter of a sliver considered for insertion. If the insertion of c eliminates a legal facet f (see Figure 6b),

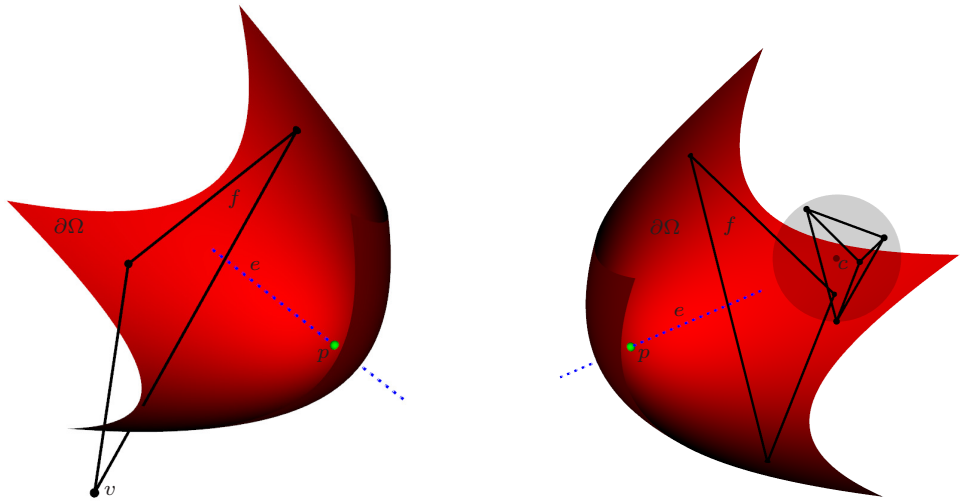


Figure 6: The point rejection strategies. (a) f is an illegal facet. (b) f is a legal facet.

then c is not inserted. Instead, a point p on the intersection of $\partial\Omega$ and f's voronoi edge e is inserted.

In summary, we cope with slivers by augmenting our algorithm (see Section 3) with the following two rules:

R6: If an illegal facet f appears, then all its vertices that do not lie on the surface are deleted and a point p on $\mathrm{Vor}\,(f) \cap \partial\Omega$ is inserted (Figure 6a).

R7: Let t be a sliver and c its circumcenter. If c eliminates a legal facet f, then c is rejected. Instead, a point on p on $\mathrm{Vor}\,(f) \cap \partial\Omega$ is inserted (Figure 6b).

We define slivers via the optimization metric η, as described by Liu and Joe [38]. Specifically, for a tetrahedron t, $\eta\,(t) = \dfrac{12\,(3v)^{\frac{2}{3}}}{\sum_{i=1}^{6} l_i^2}$, where v is the volume of t, and l_i are the lengths of t's edges. We chose η, because its computation is robust even when t is an almost flat element. In [38], it is proved that $0 \leq \eta\,(t) \leq 1$. Moreover, η is 0 for a flat element, and 1 for the regular tetrahedron.

We consider a tetrahedron t to be a sliver, if $\eta\,(t)$ is less than 0.06. The reason we chose this value is because: (a) it introduces a small size increase

(about 15%) over the mesh obtained without our sliver removal heuristic (i.e., without rules R6 and R7), and (b) it introduces a negligible time overhead. In the Experimental Section 8, we show that this 0.06 bound corresponds to meshes consisting of tetrahedra with dihedral angles between 4.6° and 171°.

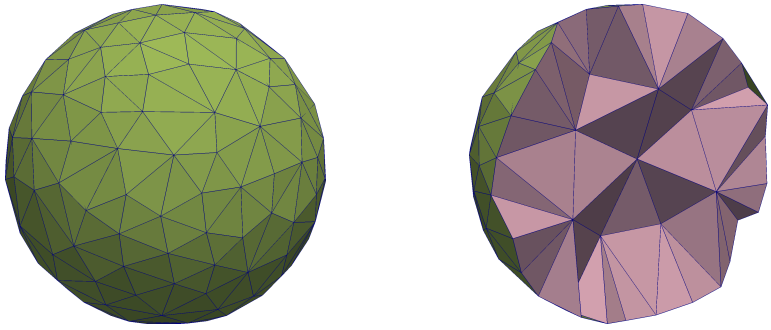
Note that R6 and R7 never remove feature vertices; on the contrary, they might insert more to "protect" the surface. Hence, they do not violate Theorem 4: the mesh boundary continues being equal to the restricted Delaunay triangulation $\mathcal{D}_{|\partial\Omega}(V \cap \partial\Omega)$, and therefore a good approximation of the surface. In order not to compromise termination (and the guarantees we give for the radius-edge ratio and the boundary planar angles), if R6 or R7 introduce an edge shorter than the shortest edge already present in the mesh, then the operation is rejected and the sliver in question is ignored.

We experimentally found that the point rejection strategies were able to generate tetrahedra with angles more than 4.5° and less than 171°. We emphasize that neither fidelity (see Theorem 4) nor termination (see Theorem 2) is compromised with this heuristic.

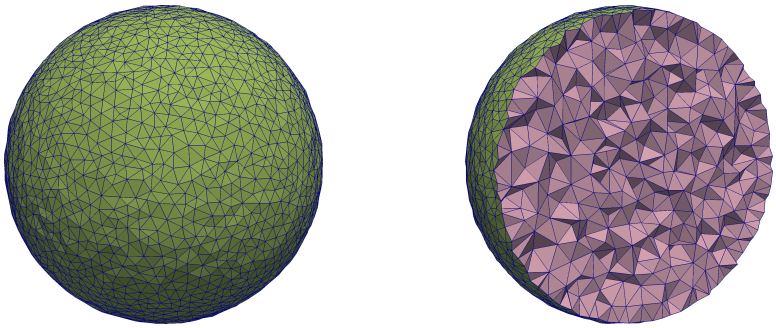
8. Experimental Evaluation

This section presents the final meshes generated by our algorithm on synthetic and real medical data. All the experiments were conducted on a 64 bit machine equipped with a 2.8 GHz Intel Core i7 CPU and 8 GB of main memory. For the 3D visualization of the final meshes, we used ParaView [39], an open source visualization application.

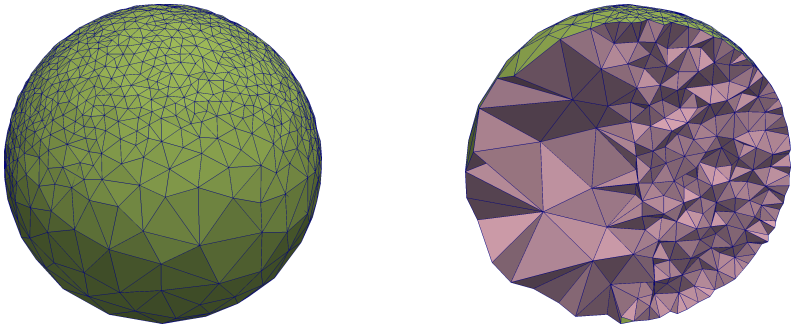
Although the fidelity guarantees we give hold for a very small value of δ (see Theorem 4), we wanted to see if our algorithm works well for much larger values of δ. Specifically, for all the experiments, we set δ to 2, i.e., we set δ to a value about 200 times larger than Theorem 4 recommends. A larger value of δ also implies that the size of the output mesh is smaller. Small-size meshes are desirable for two reasons: first, because the mesh generation execution time is considerably less (as it can be seen below, see Table 1a), and second, because finite element simulations [40, 41] on them run faster. We observed that even though the fidelity guarantees proved in Section 6 do not hold for large δ, the results in fact are pretty good. (We should also note that in some applications fidelity is not that important. For instance, a study on the impact of δ for the non-rigid registration problem [42] shows that the accuracy and speed of the solver is not very sensitive to fidelity.)



(a) $\mathrm{sf}_1(\cdot)$: the radii are smaller than 5mm.



(b) $\mathrm{sf}_2(\cdot)$: the radii are smaller than 1mm.



(c) $\mathrm{sf}_3(\cdot)$: the radii are smaller than 5mm for $z \geq 0$ and smaller than 1mm for $z < 0$.

Figure 7: Demonstrating the use of size functions. The whole mesh and a cross section of it is displayed. λ is set to about 12 times the spacing length of the image (i.e., $\lambda = 12\sqrt{3 \cdot 0.04^2} \approx 0.83$), δ to 2, $\bar{\rho}_t$ to $\sqrt{\sqrt{3}+2}$, and $\bar{\rho}_f$ to 1.

Table 1: Performance achieved by our algorithm and CGAL.

(a) Our algorithm. λ is set to 12 times the spacing length of the corresponding image, δ to 2, $\bar{\rho}_t$ to $\sqrt{\sqrt{3}+2}$, and $\bar{\rho}_f$ to 1.

	Input							
	Embedded Spheres	Torus	Brain	Stomach	Skeleton	Colon	Knee atlas	Head-neck atlas
achieved radius-edge ratio	1.93	1.93	1.93	1.84	1.93	1.93	1.93	1.93
achieved planar boundary angles (degrees)	30.0	30.0	30.0	30.0	30.0	30.0	30.0	30.0
#Vertices	2,428	2,299	6,023	4,712	50,759	5,917	102,330	36,174
#Boundary facets	3,626	3,898	10,080	8,254	95,778	11,040	146,460	74,768
#Elements	8,774	7,727	21,136	15,796	163,120	18,545	426,592	112,778
shortest mesh edge (mm)	0.45	0.61	2.83	4.78	3.9	3.19	2.2	8.2
achieved dihedral angles (degrees)	13.13 – 155.34	12.2 – 155.6	12.0 – 155.7	12.3 – 155.2	10.8 – 156.0	11.3 – 155.8	4.6 – 170.1	4.7 – 170.0
Time (secs)	1.4	1.3	4.8	2.6	41.8	5.3	43.9	20.3

(b) Performance achieved by CGAL on the same set of images. $\bar{\rho}_t$ is set to $\sqrt{\sqrt{3}+2}$, and $\bar{\rho}_f$ to 1.

	Input							
	Embedded Spheres	Torus	Brain	Stomach	Skeleton	Colon	Knee atlas	Head-neck atlas
achieved radius-edge ratio	1.41	1.30	1.86	1.34	2.63	1.40	2.34	2.71
achieved planar boundary angles (degrees)	30.0	30.0	16.5	30.0	20.3	22.6	30.0	10.6
#Vertices	7,099	1,684	3,973	3,447	42,603	4,834	81,753	35,755
#Boundary facets	3,682	1,450	2,554	2,866	54,340	5,980	53,186	60,674
#Elements	37,718	7,937	20,271	16,442	173,858	19,524	430,827	127,684
shortest mesh edge (mm)	0.56	1.42	2.63	3.67	0.01	3.19	0.26	0.15
achieved dihedral angles (degrees)	11.87 – 161.30	14.3 – 159.3	11.3 – 161.5	11.0 – 163.8	10.0 – 165.7	12.0 – 160.5	2.1 – 176.1	6.5 – 169.8
Time (secs)	1.0	0.2	0.6	0.5	10.0	1.0	13.7	8.8

As mentioned in Section 7.1, in all the following experiments, λ is set to 12 times the voxel size (i.e., length of the image spacing). Recall that λ is used so that we can compute an approximation of $\mathrm{lfs}_{\partial\Omega}(p)$, for $p \in \partial\Omega$.

For all the experiments, we set $\bar{\rho}_t$ to $\sqrt{\sqrt{3}+2}$ and $\bar{\rho}_f$ to 1, and therefore (from Theorem 2) termination is certain, all the output tetrahedra are guaranteed to have radius-edge ratio less than $\sqrt{\sqrt{3}+2}$, and all the boundary facets are guaranteed to have planar angles larger than 30°. Recall that quality is not affected by any value of δ. Although these parameters imply infinite grading constants (Theorem 3), grading is much better in practice, an observation that is also reported in [27] and demonstrated in this Section as well.

The first set of experiments demonstrates the use of custom size functions. Note that the use of any size function alters neither the quality nor the fidelity guarantees, since it is incorporated in Theorem 2 (see Section 4) and Theorem 4 (see Section 6).

Table 2: Information about the input images.

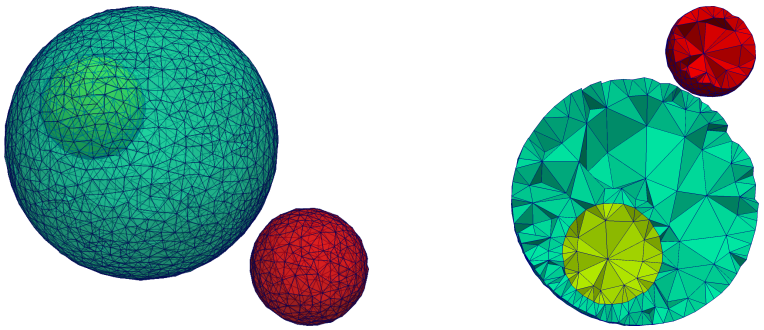
Image	Resolution	Spacing (mm^3)	Tissues
Single Sphere	$416 \times 416 \times 416$	$0.04 \times 0.04 \times 0.04$	1
Embedded Spheres	$634 \times 416 \times 416$	$0.04 \times 0.04 \times 0.04$	3
Torus	$147 \times 147 \times 67$	$0.25 \times 0.25 \times 0.25$	1
Brain	$316 \times 316 \times 188$	$0.93 \times 0.93 \times 1.5$	1
Stomach	$140 \times 186 \times 86$	$0.96 \times 0.96 \times 2.4$	1
Skeleton	$359 \times 265 \times 218$	$0.96 \times 0.96 \times 2.4$	1
Colon	$296 \times 167 \times 117$	$0.96 \times 0.96 \times 1.8$	1
Knee atlas	$413 \times 400 \times 116$	$0.27 \times 0.27 \times 1$	49
Head-neck atlas	$241 \times 216 \times 228$	$0.97 \times 0.97 \times 1.4$	60

We synthetically created the image of a sphere of radius 10mm and center $(0, 0, 0)$. See Table 2 for information about this image. We ran our algorithm on the sphere image three times, each of which with a different size function: $\mathrm{sf}_1(\cdot), \mathrm{sf}_2(\cdot)$, and $\mathrm{sf}_3(\cdot)$. $\mathrm{sf}_1(\cdot)$ restricts the radii of the elements to be smaller than 5mm, while $\mathrm{sf}_2(\cdot)$ restricts the radii of the elements to be smaller than 1mm. $\mathrm{sf}_3(\cdot)$ is a non-uniform size function. Specifically, it behaves as $\mathrm{sf}_1(\cdot)$ for $z \geq 0$ and as $\mathrm{sf}_2(\cdot)$ for the other part of the sphere.

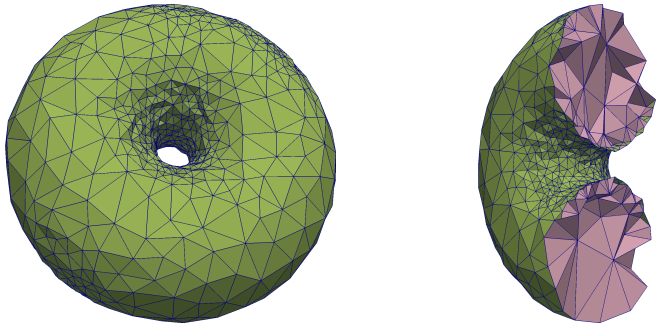
Figure 7 depicts the results. In all these three experiments, the achieved radius-edge ratio is less than $\sqrt{\sqrt{3}+2}$, and the planar angles are larger than $30°$, as theory dictates. Moreover, the dihedral angles of the output tetrahedra are between $12.9°$ and $155.8°$.

Observe that although parameters δ and λ (the ones directly responsible for the sampling density) are fixed for all three runs, the sample density varies. In fact, small size functions (i.e., size functions that take low values) make the boundary vertices denser (compare Figure 7a and Figure 7b for example). Figure 7c shows better exactly that: the surface is sampled more where the size function takes low values, and less otherwise. This indirect effect is expected and it is due to R3. Because of a small size function, more free vertices are inserter close to the surface. This, in turn, is likely to invalidate more restricted facets; that is, more restricted facets will not have their vertices on the surface, and thus, R3 is triggered dictating the insertion of more feature vertices to protect the restricted facets.

The next set of experiments shows the output of our method on difficult geometries both manifold and non manifold. Although the fidelity guarantees



(a) embedded spheres



(b) torus

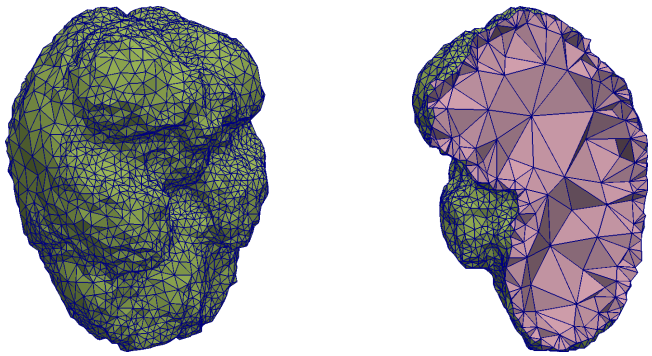
Figure 8: The final meshes produced by our algorithm for the embedded spheres and the torus. The first mesh of each row illustrates the whole mesh and the second a cross-section of it.

about the topology of the output mesh are proved only for manifold domains, in this Section we show that our method behaves fairly well for non-manifold cases (see Figure 11 for example) as well.

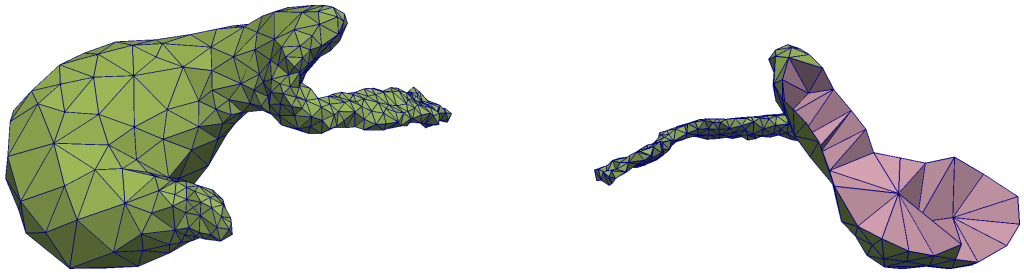
The first couple of images are the embedded spheres and a torus we synthetically created. The third is an MRI brain image obtained from Huashan Hospital[2]. The next three images are CT segmented scans of a skeleton, a colon, and a stomach, obtained from IRCAD Laparoscopic Center[3]. The last two images are the MRI knee atlas [43] and the CT head-neck [44] atlas obtained from the Surgical Planning Laboratory of Brigham and Women's

[2]Huashan Hospital, 12 Wulumuqi Zhong Lu, Shanghai, China.

[3]`http://www.ircad.fr/`



(a) brain



(b) stomach

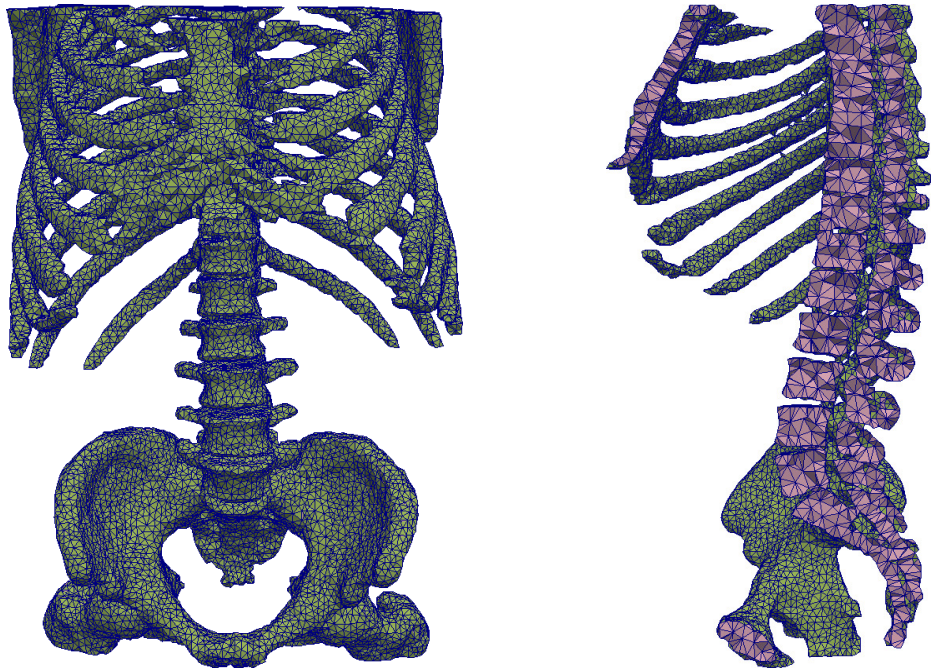
Figure 9: The final meshes produced by our algorithm for the brain and the stomach.

Hospital[4]. Information about the input images is shown in Table 2. Figure 8, Figure 9, Figure 10, and Figure 11 show the meshes produced by our algorithm on these input images.

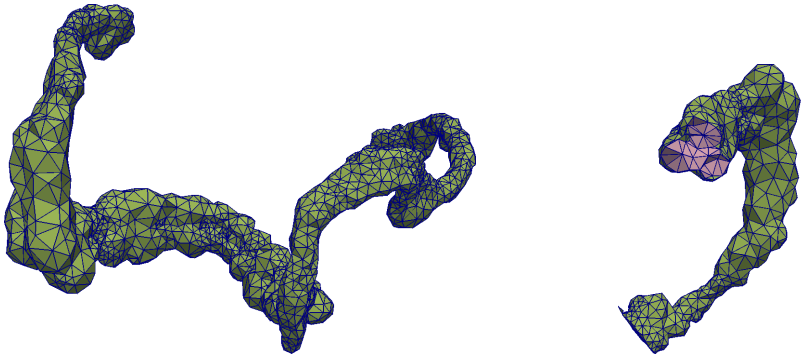
Table 1a reports some statistics for the meshes generated by our algorithm. The observed largest radius-edge ratio in all the meshes is no more than $\sqrt{\sqrt{3}+2}$ and the observed planar angles of the boundary facets in all meshes is no smaller than 30° corroborating in this way the theory.

Also, for the meshes of Figure 8, Figure 9, Figure 10, and Figure 11, notice that: (a) the interior of the object (i.e. the part away from the surface) is meshed with fewer and bigger elements (*volume grading*), and (b) in most cases, more and smaller boundary triangles mesh parts of the surface close to the medial axis (*surface grading*). Graded meshes greatly reduce the total

[4] `http://www.spl.harvard.edu/`

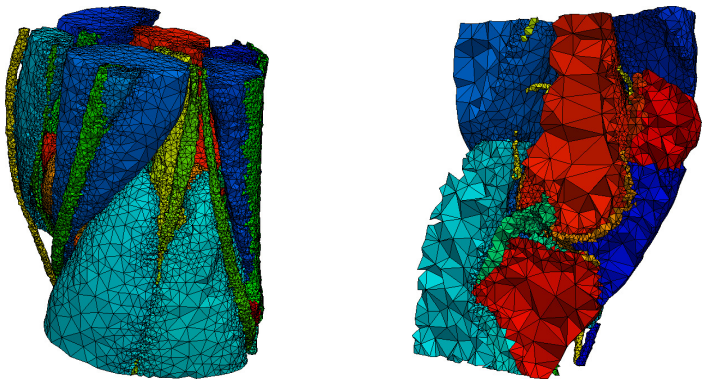


(a) skeleton

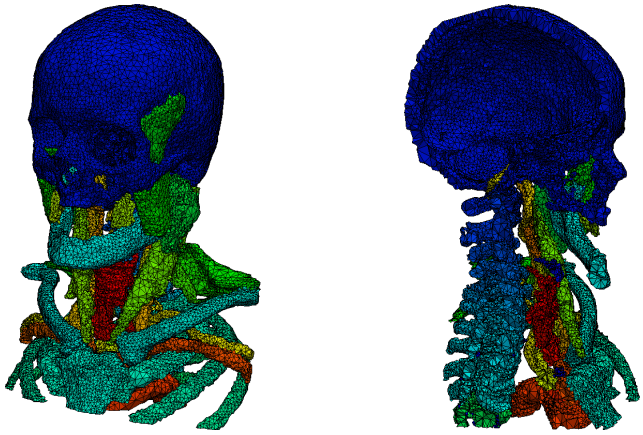


(b) colon

Figure 10: The final meshes produced by our algorithm for the skeleton and the colon.



(a) knee atlas



(b) head-neck atlas

Figure 11: The final meshes produced by our algorithm for the multi-tissue knee and head-neck atlases.

number of elements, representing, at the same time, difficult geometries (i.e., geometries with high curvature and/or non-manifold parts) accurately.

For comparison, Table 1b shows the meshes generated by CGAL [32], the state of the art mesh generation tool we are aware of, able to operate directly on images as well. We set the quality parameters to the same values with the ones used in our algorithm. Note, however, that CGAL does not offer surface grading according to the local feature size. Nevertheless, we were able to set an upper limit on the radii of all the tetrahedra, so that the resulting meshes have similar number of elements to the meshes produced by our algorithm.

Indeed, observe that both Table 1a and Table 1b report similar mesh

sizes on the same input image, with one exception: the mesh size on the Embedded Spheres generated by CGAL is more than 4 times larger than the one generated by our method. The reason for this mismatch is the fact that CGAL found it difficult to recover the red ball (see Figure 8a) with a small number of elements. We had to considerably increase the size of the whole mesh so that CGAL could represent both connected components.

Table 1a and Table 1b suggest that the quality achieved by our method is comparable to CGAL's. The execution time of our method is much higher, but this is expected since the surface grading offered by our algorithm necessitates the computation of the poles and the maintenance of a second mesh, slowing down the overall meshing time. Improving the speed of our algorithm is outside the scope of this paper and is left for future work.

9. Conclusions

In this paper, we presented a 3D Delaunay refinement image-to-mesh conversion algorithm that operates directly on segmented images. It is able to create an appropriate sample set on the object's surface, and to mesh the volume and the surface at the same time. This flexibility (besides the fact that it solves three distinct problems, that is, sampling, surface recovery, and volume meshing) results in a generally lower number of vertices than in the case where the surface is meshed without considering the rules of quality. For instance, the insertion of points that improve the quality might help the density requirement at the same time.

Apart from the fidelity guarantees we give (namely, that the mesh boundary is a good geometric and topological approximation of the object's surface), our algorithm provably achieves very low radius-edge ratio without sacrificing fidelity. The planar angles of the boundary facets are also guaranteed to be larger than 30°. Moreover, by slightly relaxing the quality guarantees, our algorithm provably exhibits good grading.

Experimental evaluation on various images shows that the final meshes are free of slivers and exhibit both volume and (in most cases) surface grading, a fact that greatly reduces the size of the mesh making the subsequent FEM analysis [40, 41, 45] faster. Lastly, demonstration of the use of custom size functions shows that our algorithm allows for additional flexibility to meet user-defined mesh density.

Acknowledgments

The authors are deeply grateful to the reviewers for improving the manuscript with their constructive suggestions and comments. This work is supported in part by NSF grants: CCF-1139864, CCF-1136538, and CSI-1136536 and by the John Simon Guggenheim Foundation and the Richard T. Cheng Endowment.

References

[1] J. R. Shewchuk, Tetrahedral mesh generation by Delaunay refinement, in: Proceedings of the 14$^{\text{th}}$ ACM Symposium on Computational Geometry, ACM, Minneapolis, MN, 1998, pp. 86–95.

[2] A. Chernikov, N. Chrisochoides, Three-Dimensional Semi-Generalized Point Placement Method for Delaunay Mesh Refinement, in: Proceedings of the 16$^{\text{th}}$ International Meshing Roundtable, Elsevier, Seattle, WA, 2007, pp. 25–44.

[3] S. Oudot, L. Rineau, M. Yvinec, Meshing volumes bounded by smooth surfaces, in: Proceedings of the International Meshing Roundtable, Springer-Verlag, 2005, pp. 203–219.

[4] L. Rineau, M. Yvinec, Meshing 3D domains bounded by piecewise smooth surfaces, in: Proceedings of the International Meshing Roundtable, pp. 443–460.

[5] D. Boltcheva, M. Yvinec, J.-D. Boissonnat, Mesh Generation from 3D Multi-material Images, in: Medical Image Computing and Computer-Assisted Intervention, Springer, 2009, pp. 283–290.

[6] J.-P. Pons, F. Ségonne, J.-D. Boissonnat, L. Rineau, M. Yvinec, R. Keriven, High-Quality Consistent Meshing of Multi-label Datasets, in: Information Processing in Medical Imaging, Springer Berlin Heidelberg, 2007, pp. 198–210.

[7] L. P. Chew, Guaranteed-quality Delaunay meshing in 3D, in: Proceedings of the 13$^{\text{th}}$ ACM Symposium on Computational Geometry, Nice, France, pp. 391–393.

[8] S.-W. Cheng, T. K. Dey, H. Edelsbrunner, M. A. Facello, S.-H. Teng, Sliver exudation, Journal of the ACM 47 (2000) 883–904.

[9] P.-L. George, H. Borouchaki, Delaunay triangulation and meshing, Application to finite elements, HERMES, 1998.

[10] H. Si, Constrained Delaunay tetrahedral mesh generation and refinement, Finite Elements in Analysis and Design 46 (2010) 33–46.

[11] F. Labelle, J. R. Shewchuk, Isosurface stuffing: fast tetrahedral meshes with good dihedral angles, ACM Transactions on Graphics 26 (2007) 57.1–57.10.

[12] X.-Y. Li, S.-H. Teng, Generating Well-Shaped Delaunay meshed in 3D, in: Proceedings of the 12$^{\text{th}}$ annual ACM-SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001, pp. 28–37.

[13] G. L. Miller, D. Talmor, S.-H. Teng, N. Walkington, A Delaunay based numerical method for three dimensions: generation, formulation, and partition, in: Proceedings of the 27th Annu. ACM Sympos. Theory Comput, ACM, 1995, pp. 683–692.

[14] L. P. Chew, Guaranteed-quality mesh generation for curved surfaces, in: Proceedings of the 9$^{\text{th}}$ ACM Symposium on Computational Geometry, San Diego, CA, pp. 274–280.

[15] N. Amenta, S. Choi, T. K. Dey, N. Leekha, A Simple Algorithm for Homeomorphic Surface Reconstruction, International Journal of Computational Geometry and Applications 12 (2002) 125–141.

[16] Jane Tournois, Rahul Srinivasan, Pierre Alliez, Perturbing Slivers in 3D Delaunay Meshes, in: Proceedings of the 18$^{\text{th}}$ International Meshing Roundtable, Sandia Labs, Salt Lake City, Utah, USA, pp. 157–173.

[17] N. Amenta, M. Bern, Surface reconstruction by Voronoi filtering, in: SCG '98: Proceedings of the fourteenth annual symposium on Computational geometry, ACM, New York, NY, USA, 1998, pp. 39–48.

[18] N. Amenta, S. Choi, R. K. Kolluri, The power crust, in: Proceedings of the sixth ACM symposium on Solid modeling and applications, SMA '01, ACM, New York, NY, USA, 2001, pp. 249–266.

[19] J.-D. Boissonnat, S. Oudot, Provably good sampling and meshing of surfaces, Graphical Models 67 (2005) 405–451.

[20] W. E. Lorensen, H. E. Cline, Marching cubes: A high resolution 3D surface construction algorithm, SIGGRAPH Computer Graphics 21 (1987) 163–169.

[21] N. Molino, R. Bridson, J. Teran, R. Fedkiw, A crystalline, red green strategy for meshing highly deformable objects with tetrahedra, in: Proceedings of the 12$^{\text{th}}$ International Meshing Roundtable, Sandia National Laboratories, 2003, pp. 103–114.

[22] A. Chernikov, N. Chrisochoides, Tetrahedral image-to-mesh conversion for biomedical applications, in: ACM Conference on Bioinformatics, Computational Biology and Biomedicine, Chicago, IL, pp. 125–134.

[23] P. Alliez, D. Cohen-Steiner, M. Yvinec, M. Desbrun, Variational tetrahedral meshing, ACM Trans. Graph. 24 (2005) 617–625.

[24] B. M. Klingner, J. R. Shewchuk, Aggressive tetrahedral mesh improvement, in: Proceedings of the International Meshing Roundtable, Springer, 2007, pp. 3–23.

[25] L. A. Freitag, C. Ollivier-Gooch, Tetrahedral mesh improvement using swapping and smoothing, International Journal for Numerical Methods in Engineering 40 (1997) 3979–4002.

[26] H. Blum, A Transformation for Extracting New Descriptors of Shape, in: W. Wathen-Dunn (Ed.), Models for the Perception of Speech and Visual Form, MIT Press, Cambridge, 1967, pp. 362–380.

[27] J. R. Shewchuk, Delaunay refinement algorithms for triangular mesh generation, Computational Geometry: Theory and Applications 22 (2002) 21–74.

[28] S. A. Mitchell, Cardinality bounds for triangulations with bounded minimum angle, in: CCCG, pp. 326–331.

[29] G. L. Miller, D. Talmor, S.-H. Teng, Data generation for geometric algorithms on non-uniform distributions, Int. J. Comput. Geometry Appl. 9 (1999) 577–598.

[30] ITK, Insight Segmentation and Registration Toolkit, http://www.itk.org, v4.1.0.

[31] C. R. Maurer, Q. Rensheng, V. Raghavan, A linear time algorithm for computing exact euclidean distance transforms of binary images in arbitrary dimensions, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (2003) 265 – 270.

[32] CGAL, Computational Geometry Algorithms Library, http://www.cgal.org, v4.0.

[33] T. K. Dey, W. Zhao, Approximate medial axis as a voronoi subcomplex, Computer-Aided Design 36 (2004) 195–202.

[34] P. Giblin, B. B. Kimia, A formal classification of 3D medial axis points and their local geometry, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2004) 238–251.

[35] D. Coeurjolly, A. Montanvert, Optimal separable algorithms to compute the reverse euclidean distance transformation and discrete medial axis in arbitrary dimension, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2007) 437–448.

[36] F. Chazal, A. Lieutier, Stability and homotopy of a subset of the medial axis, in: Proceedings of the ninth ACM symposium on Solid modeling and applications, SM '04, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2004, pp. 243–248.

[37] H. Edelsbrunner, D. Guoy, An experimental study of sliver exudation, Engineering with Computers 18 (2002) 229–240.

[38] A. Liu, B. Joe, On the shape of tetrahedra from bisection, Math. Comput. 63 (1994) 141–154.

[39] A. H. Squillacote, ParaView Guide, A Parallel Visualization Application, Kitware Inc., 2008.

[40] M. A. Audette, H. Delingette, A. Fuchs, O. Burgert, K. Chinzei, A topologically faithful, tissue-guided, spatially varying meshing strategy for computing patient-specific head models for endoscopic pituitary surgery simulation, Computer Aided Surgery 12 (2007) 43–52.

[41] M. Audette, M. Miga, J. Nemes, K. Chinzei, T. Peters, A Review of Biomechanical Modeling of the Brain for Intrasurgical Displacement Estimation and Medical Simulation, Biomechanical Systems, General Anatomy (2007) 83–112.

[42] P. Foteinos, Y. Liu, A. Chernikov, N. Chrisochoides, An Evaluation of Tetrahedral Mesh Generation for Non-Rigid Registration of Brain MRI, in: Computational Biomechanics for Medicine V, 13th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI) Workshop, Springer, 2010, pp. 126–137.

[43] J. Richolt, M. Jakab, R. Kikinis, SPL Knee Atlas (2011). Available at: `http://www.spl.harvard.edu/publications/item/view/1953`.

[44] M. Jakab, R. Kikinis, Head and neck atlas (2012). Available at: `http://www.spl.harvard.edu/publications/item/view/2271`.

[45] O. Clatz, H. Delingette, I.-F. Talos, A. J. Golby, R. Kikinis, F. Jolesz, N. Ayache, S. Warfield, Robust non-rigid registration to capture brain shift from intra-operative MRI, IEEE Transactions on Medical Imaging 24 (2005) 1417–1427.