Meshing Volumes with Curved Boundaries

Steve Oudot, Laurent Rineau, Mariette Yvinec

Engineering with Computers manuscript No.
(will be inserted by the editor)

Meshing Volumes with Curved Boundaries *

Steve Oudot[1], **Laurent Rineau**[2], **Mariette Yvinec**[3]

[1] INRIA, BP 93 06902 Sophia Antipolis, France. Steve.Oudot@sophia.inria.fr
[2] ENS, 45 rue d'Ulm, 75005 Paris, France. Laurent.Rineau@ens.fr
[3] INRIA, BP 93 06902 Sophia Antipolis, France. Mariette.Yvinec@sophia.inria.fr

July 31, 2013

Abstract This paper introduces a three-dimensional mesh generation algorithm for domains whose boundaries are curved surfaces, possibly with sharp features. The algorithm combines a Delaunay-based surface mesher with a Ruppert-like volume mesher, resulting in a greedy scheme to sample the interior and the boundary of the domain simultaneously. The algorithm constructs provably-good meshes, it gives control on the size of the mesh elements through a user-defined sizing field, and it guarantees the accuracy of the approximation of the domain boundary. A notable feature is that the domain boundary has to be known only through an oracle that can tell whether a given point lies inside the object and whether a given line segment intersects the boundary. This makes the algorithm generic enough to be applied to domains with a wide variety of boundary types, such as implicit surfaces, polyhedra, level-sets in 3D gray-scaled images, or point-set surfaces.

Key words Mesh generation, reconstruction, sampling, Delaunay refinement

1 Introduction

Simplicial meshes are one of the most popular representations for surfaces, volumes, scalar fields and vector fields, in applications such as Geographic Information Systems (GIS), computer graphics, virtual reality, med-

*Work partially supported by the IST Programme of the EU as a Shared-cost RTD (FET Open) Project under Contract No IST-006413 (ACS - Algorithms for Complex Shapes) and by the European Network of Excellence AIM@shape (FP6 IST NoE 506766).

ical imaging and finite element analysis. However, constructing discrete representations of continuous objects can be time-consuming, especially when the geometry of the object is complex. In this case, mesh generation becomes the pacing phase in the computational simulation cycle. Roughly speaking, the more the user is involved in the mesh generation process, the longer the latter is. An appealing example is given in [1], where the mesh generation time is shown to be 45 times that required to compute the solution. This motivates the search for fully-automated mesh-generation methods, which inherently require the use of guaranteed-quality meshing algorithms.

Related work

Delaunay refinement is recognized as one of the most powerful techniques for generating meshes with guaranteed quality. It allows the user to get an easy control on the sizes of the mesh elements, for instance through a (possibly non-uniform) sizing field. Moreover, it constructs meshes with a good grading, able to conform to quickly varying sizing fields. The pioneer work on Delaunay refinement is due to Ruppert [2], who proposed a mesh generator for planar domains with piecewise linear boundaries and constraints. Provided that the boundaries and constraints do not form angles smaller than $\frac{\pi}{3}$, Ruppert's algorithm guarantees a lower bound on the smallest angle in the mesh. Furthermore, this bound is achieved by adding an asymptotically optimal number of Steiner vertices. Later on, Shewchuk improved the handling of small angles in two dimensions [3] and generalized the method to the meshing of three-dimensional domains with piecewise linear boundaries [4]. The handling of small angles is more puzzling in three dimensions, where dihedral angles and facet angles come into play. Cohen-Steiner *et al.* [5] proposed to use protecting spheres around sharp edges. Cheng and Poon [6] carried on this idea, and provided a thoroughful handling of small input angles formed by boundaries and constraints. Cheng *et al.* [7] turned the same idea into a simpler and more practical meshing algorithm.

In three-dimensional space, Delaunay refinement is able to produce tetrahedral meshes with an upper bound on the radius-edge ratios of the tetrahedra, the radius-edge ratio of a tetrahedron being the ratio between its circumradius and the length of its shortest edge. This eliminates from the mesh all kinds of badly-shaped tetrahedra, except the ones called *slivers*. A sliver can be described as a tetrahedron formed by four vertices close to the equatorial circle of a sphere and roughly equally spaced on this circle. Cheng *et al.* [8], and later on Cheng and Dey [9], proposed to exude slivers from the mesh by turning the Delaunay triangulation into a weighted Delaunay triangulation with carefully-chosen small weights applied to the vertices. Li and Teng [10] proposed to avoid slivers by relaxing the choice of Steiner vertices inside small areas around the circumcenters of the elements to be refined.

The main drawback of the above techniques is that they deal exclusively with domains with piecewise linear boundaries, whereas in many applica-

tions objects have curved boundaries. In such applications, time is spent discretizing the boundary B of the object into a polyhedron P, before the interior of the object can be sampled. Then, the original boundary B is dropped away and replaced by its discretized version P:

- Mesh generation algorithms based on advancing front methods [11], as well as some Delaunay refinement techniques, like the unit edge mesher of [12,13], construct meshes that conform strictly to the discrete boundary P.
- In contrast, Ruppert-like methods [4] allow themselves to refine P. However, whenever a point should be inserted on B, it is in fact inserted on P.

For both families of algorithms, the quality of the resulting mesh and the accuracy of the boundary approximation depend highly on the initial surface mesh P.

Several methods have been proposed for meshing two-dimensional or three-dimensional domains with curved boundaries. Most of them deal only with specific types of boundaries (parametric, implicit etc.) [14], or they simply come with no guarantee regarding the topology of the output mesh, or the quality of its elements, or even the termination of the process [15–17]. One notable exception is [18], where the algorithm is able to handle any two-dimensional domain bounded by piecewise smooth curves, of any type, provided that a small number of geometric quantities can be estimated, such as the curvature of a given curve at a given point or the total variation of the unit tangent vector between two points on a given curve. The problem with this approach is that it is designed exclusively for the two-dimensional case. Moreover, the geometric information required by the algorithm, such as surface curvature, may not always be readily available and can be time-consuming (if ever possible) to retrieve on certain types of input data.

Contributions

In this paper, we take advantage of recent results on the front of smooth or Lipschitz surface meshing and approximation using Delaunay refinement [19, 20], to build a fully-automated algorithm that can mesh three-dimensional domains bounded by smooth or Lipschitz surfaces. Intuitively, a surface is Lipschitz if it is locally the graph of a Lipschitz bivariate function. The class of Lipschitz surfaces includes in particular all piecewise smooth surfaces with bounded normal deviation at singular points.

Our approach combines the surface mesher of [19,20] with a Ruppert-like volume mesher, resulting in a greedy Delaunay-based scheme to sample the interior and the boundary of the domain simultaneously. The algorithm is guaranteed to terminate and to construct good-quality meshes for domains whose boundaries are smooth or Lipschitz surfaces of any topological type. In particular, boundaries are allowed to have sharp features, but on the other

hand they must be manifold. The sizes of the mesh elements are controlled through a user-defined sizing field. Moreover, the size of the output mesh is bounded.

A noticeable feature of the algorithm is that the boundary of the object has to be known only through an oracle that can answer two simple geometric questions: whether a given point lies inside the object, and whether a given line segment intersects the boundary. This makes the algorithm generic enough to be applied to objects with a wide variety of boundary types, such as implicit surfaces, polyhedra, level-sets in 3D gray-scaled images, or point-set surfaces. Note that, when the boundary B of the domain is a polyhedron, we do not constrain the mesh to conform to the edges of B, however we control the approximation error given by the Hausdorff distance between B and the boundary of the mesh.

Overview

For simplicity, the core of the paper focuses on the case where the domain to mesh has smooth boundaries. Section 2 recalls a few known facts about restricted Delaunay triangulations and smooth surface approximation. Section 3 describes the main algorithm. Section 4 deals with the accuracy of the approximation of the object by the output mesh. In Section 5, we prove that the meshing algorithm terminates, and we bound the number of vertices of the output mesh.

Section 6 addresses the more general case where the domain boundary is Lipschitz. Section 7 discusses the practicality of the algorithm: it gives some details about the choice of the sizing field, and it explains how to remove slivers. Finally, Section 8 provides a few examples and experimental results.

2 Preliminary definitions

In the sequel, $\mathcal{O}$ denotes a bounded open subset of Euclidean space $\mathbb{R}^3$. We call respectively $\bar{\mathcal{O}}$ and $\partial\mathcal{O}$ the topological closure and the boundary of $\mathcal{O}$. In this section, as well as in Sections 3, 4 and 5, we assume that $\partial\mathcal{O}$ is $C^{1,1}$, which means that it is continuous and that its normal vector field is continuous and satisfies a Lipschitz condition.

Definition 1

• *The* medial axis M *of* $\partial\mathcal{O}$ *is the topological closure of the set of points of* $\mathbb{R}^3$ *that have at least two nearest neighbors on* $\partial\mathcal{O}$*. Every point of* M *is the center of an open ball that is maximal with respect to inclusion among the set of open balls included in* $\mathbb{R}^3 \setminus \partial\mathcal{O}$*. Such a ball is called a* medial ball.

• *Given a point* $x \in \mathbb{R}^3$*, we call* distance to the medial axis at x*, or* $d_M(x)$*, the Euclidean distance from* x *to* M*.*

It is well-known [21] that, since $\partial\mathcal{O}$ is $C^{1,1}$, the infimum of d_M over $\partial\mathcal{O}$ is positive. This infimum is called the *reach* of $\partial\mathcal{O}$. The class of surfaces with positive reach has been extensively studied in the recent years, and the distance to the medial axis has played a prominent role in the development of a sampling theory for smooth surfaces. In particular, Amenta and Bern [22] introduced the notion of ε-sample of a surface, and showed that such samples are convenient for reconstructing surfaces with guarantees.

Definition 2 *Given a positive function ε defined over $\partial\mathcal{O}$, a finite point set $\mathcal{P}$ is an ε-sample of $\partial\mathcal{O}$ if $\mathcal{P} \subset \partial\mathcal{O}$ and if $\forall x \in \partial\mathcal{O}$, $d(x, \mathcal{P}) \leq \varepsilon(x)$.*

Amenta and Bern proved that, if the function ε is sufficiently small compared to the distance to the medial axis of $\partial\mathcal{O}$, then the so-called *restricted Delaunay triangulation* of $\mathcal{P}$, introduced below in Definition 4, is a triangulated surface that provably approximates $\partial\mathcal{O}$ in a topological and in a geometric sense. This idea has been carried on to do surface meshing and surface reconstruction – see [19, 23, 24] and the references therein for a survey.

Definition 3 *Let $\mathcal{P}$ be a finite set of points.*

• *The* Voronoi cell *of $p \in \mathcal{P}$ is the set of all points of $\mathbb{R}^3$ that are closer to p than to any other $p' \in \mathcal{P}$.*

• *The* Voronoi diagram *of $\mathcal{P}$, $\mathcal{V}(\mathcal{P})$, is the cellular complex formed by the Voronoi cells of the points of $\mathcal{P}$.*

It is well known that, if the points of $\mathcal{P}$ are in general position, then the dual complex of $\mathcal{V}(\mathcal{P})$ is a tetrahedrization of the convex hull of $\mathcal{P}$, called the *Delaunay triangulation* (or $\mathcal{D}(\mathcal{P})$ for short).

Definition 4 *Let $\mathcal{P}$ be a finite point set.*

• *The Delaunay triangulation of $\mathcal{P}$* restricted to *$\mathcal{O}$, or $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ for short, is the subcomplex of $\mathcal{D}(\mathcal{P})$ formed by the tetrahedra whose dual Voronoi vertices lie in $\mathcal{O}$.*

• *The Delaunay triangulation of $\mathcal{P}$* restricted to *$\partial\mathcal{O}$, or $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ for short, is the subcomplex of $\mathcal{D}(\mathcal{P})$ formed by the triangles whose dual Voronoi edges intersect $\partial\mathcal{O}$.*

Given a facet f of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ and its dual Voronoi edge e, every point of $e \cap \partial\mathcal{O}$ is the center of an open ball containing no point of $\mathcal{P}$, and whose bounding sphere passes through the vertices of f. See Figure 1. This ball is called a *surface Delaunay ball of $\mathcal{P}$*.

The main idea of our algorithm is to sample $\mathcal{O}$ and $\partial\mathcal{O}$ greedily and simultaneously, using $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ and $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ to drive the choice of the next point to insert. The output is a point set whose restriction to $\partial\mathcal{O}$ is a *loose ε-sample* of $\partial\mathcal{O}$ [19]:

Definition 5 *Given a positive function ε defined over $\partial\mathcal{O}$, a finite point set $\mathcal{P}$ is a* loose ε-sample *of $\partial\mathcal{O}$ if the following conditions hold:*

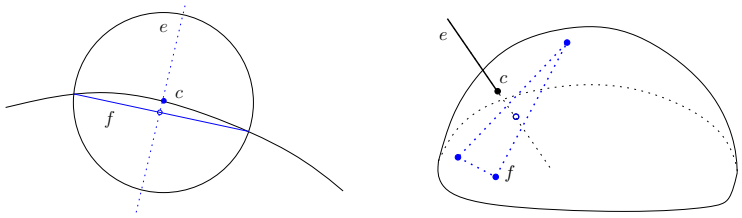

Figure 1 Left: the equivalent of a surface Delaunay ball for a curve in dimension two. Right: the center of a surface Delaunay ball of a facet $f \in \mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ is the intersection of the Voronoi edge e dual of f with the surface $\partial\mathcal{O}$.

(L1) $\mathcal{P} \subset \partial\mathcal{O}$;
(L2) $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ *has vertices on every connected component of* $\partial\mathcal{O}$;
(L3) *the center* c *of every surface Delaunay ball of* $\mathcal{P}$ *is closer to* $\mathcal{P}$ *than* $\varepsilon(c)$.

Observe that ε-samples satisfy Assertions L1 and L3. Moreover, if $\varepsilon < 0.1\ \mathrm{d}_M$, then L2 is satisfied as well, by Theorem 2 of [22]. It follows that any ε-sample is a loose ε-sample, for $\varepsilon < 0.1\ \mathrm{d}_M$. Loose ε-samples share many beautiful properties with ε-samples, including:

Theorem 1 (from [19]) *If* $\mathcal{P}$ *is a* loose ε-sample *of* $\partial\mathcal{O}$, *with* $\varepsilon \leq 0.09\ d_M$, *then the following statements hold:*
(i) $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ *is a closed* 2*-manifold ambient isotopic to* $\partial\mathcal{O}$,
(ii) *the Hausdorff distance between* $\partial\mathcal{O}$ *and* $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ *is* $O(\mu^2\ \mathrm{diam}(\mathcal{O}))$, *where* $\mu = \sup_{x\in\partial\mathcal{O}} \frac{\varepsilon(x)}{d_M(x)}$ *and* $\mathrm{diam}(\mathcal{O})$ *is the diameter of* $\mathcal{O}$,
(iii) *the normals of* $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ *approximate the normals of* $\partial\mathcal{O}$ *within an error of* $O(\mu)$,
(iv) $\mathcal{P}$ *is an* $\varepsilon(1 + 8.5\ \mu)$*-sample of* $\partial\mathcal{O}$.

Ambient isotopy and Hausdorff distance are most interesting for our problem. As for normal approximation, it is useful in applications that require to estimate differential quantities of surfaces from point samples.

3 Main algorithm

The algorithm takes as input the domain $\mathcal{O}$ to mesh, a sizing field σ, and three parameters: α, ϱ_f and ϱ_t. The domain is known through an oracle that can tell whether a given point lies inside $\mathcal{O}$ or not. The oracle can also detect whether a given segment intersects $\partial\mathcal{O}$ and, in the affirmative, return all the points of intersection (which are finitely many, generically). The sizing field is a positive function $\sigma : \bar{\mathcal{O}} \to \mathbb{R}^+$ defined over $\bar{\mathcal{O}}$ and assumed to be 1-Lipschitz.
The algorithm first constructs an initial point set $\mathcal{P}_i \subset \partial\mathcal{O}$ that satisfies the following conditions:

(I1) $\mathcal{P}_i$ is a $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$, namely: $\forall x \in \partial\mathcal{O}$, $\mathrm{d}(x, \mathcal{P}_i) \leq 0.09\ \mathrm{d}_M(x)$;

(I2) $\mathcal{P}_i$ is $0.03 d_M$-*sparse*, that is: $\forall p \in \mathcal{P}_i$, $\mathrm{d}(p, \mathcal{P}_i \setminus \{p\}) \geq 0.03\ \mathrm{d}_M(p)$.

The construction of such a point set is described extensively in [19], therefore we skip it here. Once $\mathcal{P}_i$ is built, the algorithm constructs $\mathcal{P}$ iteratively, starting with $\mathcal{P} = \mathcal{P}_i$, and inserting one point in $\mathcal{P}$ per iteration. In the meantime, the restricted Delaunay triangulations $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ and $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ are maintained, using the oracle.

At each iteration, one element of the mesh (a facet of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ or a tetrahedron of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$) is refined. To refine a tetrahedron, the algorithm inserts its circumcenter in $\mathcal{P}$. To refine a facet f of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$, the algorithm inserts in $\mathcal{P}$ the center of some surface Delaunay ball circumscribing f. At each iteration, the choice of the next point to insert is driven by the following rules, considered in this order:

R1 if a facet f of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ does not have its three vertices on $\partial\mathcal{O}$, or if its radius-edge ratio is greater than ϱ_f, then insert in $\mathcal{P}$ the center of any of the surface Delaunay balls that circumscribe f;

R2 if a facet f of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ has a surface Delaunay ball $B(c, r)$ such that $r > \alpha\ \sigma(c)$, then insert c in $\mathcal{P}$;

R3 if a tetrahedron t of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ has a circumradius greater than $\sigma(c)$, where c is the circumcenter of t, or if t has a radius-edge ratio greater than ϱ_t, then consider the circumcenter c of t:

- **R3.1** if c is not included in any surface Delaunay ball, then insert c in $\mathcal{P}$;
- **R3.2** else, insert in $\mathcal{P}$ the center of one surface Delaunay ball containing c.

The algorithm terminates when the triggering conditions of rules R1, R2 and R3 are no longer met. The point set $\mathcal{P}$ is then renamed $\mathcal{P}_f$ and returned, as well as $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$. Note that every facet f of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ has its three vertices on $\partial\mathcal{O}$ and a radius-edge ratio bounded by ϱ_f (rule R1). In addition, every surface Delaunay ball $B(c, r)$ circumscribing f has a radius $r \leq \alpha\ \sigma(c)$ (rule R2). Finally (rule R3), every tetrahedron t of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ has a circumradius $r \leq \min\{\sigma(c),\ \varrho_t\ l_{min}\}$, where c is the circumcenter of t and l_{min} is the length of the shortest edge of t.

It appears from the above description that our algorithm is similar in spirit to the Delaunay refinement scheme for polyhedra proposed in [4]. At least, the strategies used to select the tetrahedra to be refined and to choose the refinement points to insert, are quite similar. Nevertheless, an obvious difference is that we have no rule to conform the mesh to the input edges. Also, our algorithm does not handle constrained facets embedded in the facets of the bounding polyhedron but extract from the current triangulation the restricted Delaunay facets approximating the domain boundary. This forces us to explicitly check (through rule R1) that every facet of the Delaunay triangulation restricted to $\partial\mathcal{O}$ has its three vertices on $\partial\mathcal{O}$. The notion of smallest circumscribing ball of a constrained facet (called the Gabriel ball) is replaced by that of a surface Delaunay ball.

4 Approximation accuracy

In this section, we assume that the algorithm terminates, and we focus on the quality of its output. Termination will be discussed in Section 5. Let $\mathcal{P}_{f\,|\partial\mathcal{O}} = \mathcal{P}_f \cap \partial\mathcal{O}$.

Theorem 2 *$\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is a 3-manifold ambient isotopic to $\bar{\mathcal{O}}$, at Hausdorff distance $O(\mu^2\ \mathrm{diam}(\mathcal{O}))$ from $\bar{\mathcal{O}}$, where $\mu = \min\{0.09,\ \sup_{x\in\partial\mathcal{O}} \frac{\alpha\,\sigma(x)}{d_M(x)}\}$. Moreover, $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is a $\min\{0.09\ d_M,\ \alpha(1+8.5\mu)\ \sigma\}$-sample of $\partial\mathcal{O}$.*

The rest of Section 4 is devoted to the proof of this theorem and can be skipped in a first reading. Since $\mathcal{P}_i$ is a $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$, $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is also a $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$, since no point is deleted during the course of the algorithm. Thus, $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$ is a closed 2-manifold with the same topology type as $\partial\mathcal{O}$, by Theorem 1 (i). Therefore, to provide theoretical guarantees on the topology of the output of the algorithm, it suffices to prove that the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is equal to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. We will proceed in two steps:

- First, in Lemmas 1–2 and Corollary 1, we will show that the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ coincides with the Delaunay triangulation restricted to the boundary of the domain, $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$.
- Next, in Lemmas 3–4, we will show that $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ itself coincides with $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$, which, as mentioned above, is ambient isotopic to $\partial\mathcal{O}$.

Finally, in Lemma 5, we will show that the surface Delaunay balls of $\mathcal{P}_f$ are the same as the ones of $\mathcal{P}_{f\,|\partial\mathcal{O}}$, which will enable us to conclude the proof of Theorem 2 by bounding the Hausdorff distance between $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ and $\mathcal{O}$.

Lemma 1 *The boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is a subcomplex of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$. Moreover, if every edge of the Voronoi diagram $\mathcal{V}(\mathcal{P}_f)$ intersects $\partial\mathcal{O}$ at most once, and transversally, then the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is equal to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$.*

Proof Since $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is a union of Delaunay tetrahedra, its boundary is a union of Delaunay facets. Let f be a facet of the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$. By definition, f belongs to two Delaunay tetrahedra, one of which has its dual Voronoi vertex inside $\mathcal{O}$, whereas the other one has its dual Voronoi vertex outside $\mathcal{O}$ (and possibly at infinity). It follows that the Voronoi edge dual to f intersects $\partial\mathcal{O}$, which means that $f \in \mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$.

Let us now assume that every edge of $\mathcal{V}(\mathcal{P}_f)$ intersects $\partial\mathcal{O}$ at most once, and transversally. Let f be a facet of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$. By definition, the Voronoi edge dual to f intersects $\partial\mathcal{O}$. Since this edge intersects $\partial\mathcal{O}$ only once, and transversally, one of its vertices lies inside $\mathcal{O}$ whereas the other one (which may be at infinity) lies outside $\mathcal{O}$. It follows, by definition of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$, that one of the Delaunay tetrahedra incident to f belongs to $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$, while the other one does not. Hence, f belongs to the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$. □

In our case, $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ is precisely the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$, by virtue of the following result:

Lemma 2 *Every edge of $\mathcal{V}(\mathcal{P}_f)$ that intersects $\partial\mathcal{O}$ does it in only one point, and its vertices belong to different connected components of $\mathbb{R}^3 \setminus \partial\mathcal{O}$.*

Proof Among the edges of $\mathcal{V}(\mathcal{P}_f)$, only those whose dual Delaunay facets have their three vertices on $\partial\mathcal{O}$ can intersect $\partial\mathcal{O}$, thanks to rule R1. Let e be such an edge. It is included in an edge e' of $\mathcal{V}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. Since $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is a $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$, Lemma 3.6 of [19] tells that e' intersects $\partial\mathcal{O}$ at most once, and transversally, which yields the lemma. □

Corollary 1 *The boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ coincides with $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$.*

It follows from Corollary 1 that, if we can prove that $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f) = \mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$, then the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ will be equal to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. We need an intermediate result:

Lemma 3 *$\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ has vertices on all the connected components of $\partial\mathcal{O}$.*

Proof By rule R1, every edge e of $\mathcal{V}(\mathcal{P}_f)$ that intersects $\partial\mathcal{O}$ has a dual Delaunay facet f whose three vertices are in $\mathcal{P}_{f\,|\partial\mathcal{O}}$. Since $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is a $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$, the point $c = e \cap \partial\mathcal{O}$ lies at distance at most $0.09\ \mathrm{d}_M(c)$ from the vertices of f. It follows, by Lemma 8 of [22], that c and the vertices of f lie on the same connected component of $\partial\mathcal{O}$. As a consequence, to prove the lemma, it suffices to show that every connected component of $\partial\mathcal{O}$ is intersected by at least one Voronoi edge.

Note that every connected component $\mathcal{C}$ of $\partial\mathcal{O}$ is the border between two connected components Ω_1 and Ω_2 of $\mathbb{R}^3 \setminus \partial\mathcal{O}$, such that every connected path from Ω_1 to Ω_2 crosses $\mathcal{C}$. Therefore, to prove that $\mathcal{C}$ is intersected by a Voronoi edge, it suffices to prove that Ω_1 and Ω_2 both contain Voronoi vertices, since the graph made of the Voronoi vertices and edges is connected.

Let us assume for a contradiction that some component Ω of $\mathbb{R}^3 \setminus \partial\mathcal{O}$ contains no Voronoi vertex. Let x be a point of Ω farthest from $\partial\mathcal{O}$. As a local maximum, x is a critical point of the distance to $\partial\mathcal{O}$. Therefore, x belongs to the medial axis of $\partial\mathcal{O}$. Since the Delaunay balls centered at the Voronoi vertices (including the ones at infinity) cover $\mathbb{R}^3$, at least one such ball (say $B(c,r)$) contains x. Since c lies outside Ω while x lies inside, the line segment $[c,x]$ intersects the boundary of Ω (which is part of $\partial\mathcal{O}$). Let y be a point of intersection – see Figure 2. The open ball centered at y, of radius $\mathrm{d}(x,y)$, is contained in the interior of $B(c,r)$. Therefore, it contains no point of $\mathcal{P}_f$. Now, its radius is $\mathrm{d}(x,y)$, which is at least the distance from y to M since $x \in M$. It follows that y is farther from $\mathcal{P}_f$ than $\mathrm{d}_M(y)$, which contradicts the fact that $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is a $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$. Hence, Ω contains at least one Voronoi vertex, which ends the proof of Lemma 3. □

We can now prove that $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f) = \mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$, by using the fact that $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ is the boundary of a three-dimensional object, namely $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ (Corollary 1).

Lemma 4 *$\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f) = \mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$.*

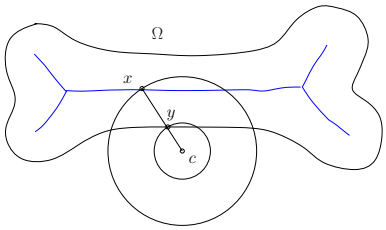

Figure 2 For the proof of Lemma 3

Proof Thanks to rule R1, all the facets of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ have their three vertices in $\mathcal{P}_{f\,|\partial\mathcal{O}}$, hence their dual Voronoi edges are included in edges of $\mathcal{V}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. It follows that $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ is a subcomplex of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$.

Since, from Corollary 1, $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ is the boundary of a three dimensional object, namely $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$, it is the union of surface mesh components without boundary. From what precedes, we know that each component C of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ is included in a component C' of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$ and because C' is manifold and without boundary, C and C' coincides. Furthermore, Lemma 3 implies that $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$ has as many components as $\partial\mathcal{O}$ and $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$ and therefore is equal to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. □

It follows from the previous results that the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is equal to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$, which is ambient isotopic to $\partial\mathcal{O}$, by Theorem 1 (i). In addition to this topological result, we would like to give a bound on the Hausdorff distance between $\partial\mathcal{O}$ and the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$, depending on the input sizing field σ. Let $\mu = \min\{0.09,\ \sup_{x\in\partial\mathcal{O}} \frac{\alpha\,\sigma(x)}{\mathrm{d}_M(x)}\}$. Our bound will depend on μ. So far, we know that $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is a $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$.

Lemma 5 *The surface Delaunay balls of $\mathcal{P}_f$ and those of $\mathcal{P}_{f\,|\partial\mathcal{O}}$ are the same.*

Proof Since every edge of $\mathcal{V}(\mathcal{P}_f)$ that intersects $\partial\mathcal{O}$ is included in an edge of $\mathcal{V}(\mathcal{P}_{f\,|\partial\mathcal{O}})$, the surface Delaunay balls of $\mathcal{P}_f$ are also surface Delaunay balls of $\mathcal{P}_{f\,|\partial\mathcal{O}}$. Let us show that the converse is true. Let e be an edge of $\mathcal{V}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. If $e \cap \partial\mathcal{O} \neq \emptyset$, then $|e \cap \partial\mathcal{O}| = 1$, by Lemma 3.6 of [19]. Moreover, the Delaunay facet dual to e belongs to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_f)$, by Lemma 4. This means that e contains an edge e' of $\mathcal{V}(\mathcal{P}_f)$, such that $|e' \cap \partial\mathcal{O}| \geq 1$. Hence, $e \cap \partial\mathcal{O} = e' \cap \partial\mathcal{O}$. □

Thanks to Lemma 5, we know that rule R2 controls the radii of all the surface Delaunay balls of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. As a consequence, upon termination of the algorithm, $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is a loose $\alpha\sigma$-sample of $\partial\mathcal{O}$, in addition to being a $0.09\mathrm{d}_M$-sample (recall that $\mathcal{P}_i \subseteq \mathcal{P}_{f\,|\partial\mathcal{O}}$ is a $0.09\mathrm{d}_M$-sample). We can now conclude the proof of Theorem 2:

Proof of Theorem 2. By Corollary 1 and Lemma 4, the boundary of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is equal to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f\,|\partial\mathcal{O}})$. Since $\mathcal{P}_{f\,|\partial\mathcal{O}}$ is a (loose) $0.09\mathrm{d}_M$-sample of $\partial\mathcal{O}$,

Theorem 1 (i) states that there exists an ambient isotopy $h : [0,1] \times \mathbb{R}^3 \to \mathbb{R}^3$ that maps $\partial\mathcal{O}$ to $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f|\partial\mathcal{O}})$. The map $h(1,.) : \mathbb{R}^3 \to \mathbb{R}^3$ is an ambient homeomorphism that maps the compact 3-manifold $\bar{\mathcal{O}}$ to a compact 3-manifold bounded by $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f|\partial\mathcal{O}})$. Now, the only compact 3-manifold bounded by $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f|\partial\mathcal{O}})$ is $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ itself[1]. Thus, we have $h(1,\bar{\mathcal{O}}) = \mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$, which means that $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is ambient isotopic to $\bar{\mathcal{O}}$.

Since $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ and $\bar{\mathcal{O}}$ are both compact, their Hausdorff distance is achieved by a pair of points lying on their boundaries. Hence, we have $\mathrm{d}_H(\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f),\ \bar{\mathcal{O}}) = \mathrm{d}_H(\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_{f|\partial\mathcal{O}}),\ \partial\mathcal{O})$, which by Theorem 1 (ii) is $O(\mu^2\ \mathrm{diam}(\mathcal{O}))$ since $\mathcal{P}_{f|\partial\mathcal{O}}$ is a loose min $\{0.09\ \mathrm{d}_M,\ \alpha\ \sigma\}$-sample of $\partial\mathcal{O}$. As for the fact that $\mathcal{P}_{f|\partial\mathcal{O}}$ is a min $\{0.09\ \mathrm{d}_M,\ \alpha(1+8.5\mu)\ \sigma\}$-sample of $\partial\mathcal{O}$, it is a direct consequence of Theorem 1 (iv) and of the fact that $\mathcal{P}_{f|\partial\mathcal{O}}$ is both a loose min $\{0.09\ \mathrm{d}_M,\ \alpha\ \sigma\}$-sample and a 0.09-sample of $\partial\mathcal{O}$. □

Observe that the results of this section do not rely on rule R3. Hence, they hold not only upon termination, but also during the course of the algorithm, each time neither rule R1 nor rule R2 can be applied. In particular, Theorem 2 holds every time rule R3 is triggered. This observation will be instrumental in proving Lemma 6 of Section 5.

5 Termination and size of the output

In this section, we provide conditions on parameters α, ϱ_f and ϱ_t, to ensure that the algorithm terminates. We assume that the sizing field σ is 1-Lipschitz over $\bar{\mathcal{O}}$.

Our strategy is to work out an upper bound on the size of the point sample constructed by the algorithm. The termination of the algorithm will then follow. Our upper bound is expressed in terms of a sizing field σ', which depends on d_M and on the parameters used in the algorithm, and which reflects the influence of each of these quantities on the size of the output. Let us first extend d_M over $\bar{\mathcal{O}}$:

$$\forall p \in \bar{\mathcal{O}},\ \sigma_0(p) = \inf\ \{\mathrm{d}(p,q) + \mathrm{d}_M(q) \mid q \in \partial\mathcal{O}\} \tag{1}$$

As proved in [15,25,26], σ_0 is a 1-Lipschitz function, equal to $\mathrm{d}_M(p)$ on $\partial\mathcal{O}$. In fact, σ_0 is the pointwise maximal 1-Lipschitz function which is at most d_M on $\partial\mathcal{O}$. We now define σ' as follows:

$$\forall p \in \bar{\mathcal{O}},\ \sigma'(p) = \min\ \{\alpha\ \sigma(p),\ 0.03\ \sigma_0(p)\}. \tag{2}$$

Let $\gamma = \max\ \{\alpha,\ 0.03\}$. Since σ and σ_0 are 1-Lipschitz, σ' is γ-Lipschitz.

Theorem 3 *If $\alpha < \frac{1}{5}$, $\varrho_f \geq \frac{1}{1-\gamma}$ and $\varrho_t \geq \frac{4}{1-5\gamma}$ (where $\gamma = \max\{\alpha,\ 0.03\}$), then*

$$|\mathcal{P}_f| \ =\ O\left(\iiint_{\mathcal{O}} \frac{dx}{\sigma_0^3(x)}\ +\ \frac{1}{\alpha^3}\ \iiint_{\mathcal{O}} \frac{dx}{\sigma^3(x)}\right),$$

[1] $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is compact because it is a finite union of tetrahedra.

where σ_0 depends only on $\mathcal{O}$ (not on σ).

Since σ_0 and σ are both positive and continuous over $\bar{\mathcal{O}}$, which is compact, the bound given in the theorem is finite. It follows that the algorithm inserts finitely many points. Since one point is inserted at each iteration, and since no point is ever removed, the algorithm terminates.

The rest of Section 5 is devoted to the proof of Theorem 3 and can be skipped in a first reading. In the sequel, $\mathcal{P}_{|\partial\mathcal{O}}$ stands for $\mathcal{P} \cap \partial\mathcal{O}$. Our first task is to provide a lower bound on the so-called *insertion radius* of a point inserted by the algorithm.

Definition 6 *Given a point p inserted in $\mathcal{P}$ by the algorithm, the* insertion radius *of p, or $r(p)$ for short, is the Euclidean distance from p to $\mathcal{P}$ right before its insertion*[2]*. The insertion radius of a point p of the initial point set $\mathcal{P}_i$ is the Euclidean distance from p to $\mathcal{P}_i \setminus \{p\}$.*

In fact, we prove a stronger result:

Lemma 6 *If $\alpha < \frac{1}{5}$, $\varrho_f \geq \frac{1}{1-\gamma}$ and $\varrho_t \geq \frac{4}{1-5\gamma}$, then, at each iteration of the algorithm, one has:*
(C1) *$\forall p \in \mathcal{P}$, $r(p) \geq \sigma'(p)$;*
(C2) *$\forall p \in \mathcal{P} \setminus \mathcal{P}_{|\partial\mathcal{O}}$, $\delta(p) \geq \frac{1}{1-\gamma}\,\sigma'(p)$, where $\delta(p)$ is the Euclidean distance from p to $\partial\mathcal{O}$.*

Proof We prove the lemma by induction. Initially, we have $\mathcal{P} = \mathcal{P}_i$, and every point p of $\mathcal{P}_i$ satisfies C1, since p is farther than $0.03\ \mathrm{d}_M(p)$ from $\mathcal{P}_i \setminus \{p\}$. Moreover, the points of $\mathcal{P}_i$ belong to $\partial\mathcal{O}$, thus C2 is also satisfied. Let us now assume that C1 and C2 are satisfied by every point of $\mathcal{P}$, up to a certain step where point c is inserted in $\mathcal{P}$. We will prove that c also satisfies C1 and C2.

• If rule R1 is being applied, then c is the center of a surface Delaunay ball $B(c,r)$ of $\mathcal{P}$ circumscribing a facet f of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$, such that:

1. either the bounding sphere of $B(c,r)$ passes through a point $p \in \mathcal{P}\setminus\mathcal{P}_{|\partial\mathcal{O}}$,
2. or $r > \varrho_f\,|e|$, where e is the shortest edge of f.

In case 1., $r(c)$ is at least the distance $\delta(p)$ from p to $\partial\mathcal{O}$, which by induction is at least $\frac{1}{1-\gamma}\,\sigma'(p)$. Since σ' is γ-Lipschitz, we have $\sigma'(p) \geq (\sigma'(c) - \gamma\,\mathrm{d}(c,p))$, hence

$$\mathrm{d}(c,p) \geq \frac{1}{1-\gamma}\,(\sigma'(c) - \gamma\,\mathrm{d}(c,p))\,,$$

which implies that $r(c) = \mathrm{d}(c,p) \geq \sigma'(c)$. It follows that C1 is satisfied for c.

[2]Note that it is also the length of the shortest Delaunay edge that is created when p is inserted.

In case 2., we have $|e| \geq r(p)$, where p is the vertex of e inserted last. By induction, $r(p) \geq \sigma'(p)$, which is at least $\sigma'(c) - \gamma\ \mathrm{d}(c,p) = \sigma'(c) - \gamma\ r(c)$ since σ' is γ-Lipschitz. It follows that $r(c) \geq \frac{\varrho_f}{1+\gamma\ \varrho_f}\ \sigma'(c)$. Hence, C1 is satisfied for c if ϱ_f satisfies:

$$\varrho_f \geq \frac{1}{1-\gamma} \tag{3}$$

As for C2, it is satisfied in both cases since c belongs to $\partial\mathcal{O}$.

• If rule R2 is applied, then c is the center of a surface Delaunay ball of radius greater than $\alpha\ \sigma(c) \geq \sigma'(c)$, thus the insertion radius of c is at least $\sigma'(c)$, hereby satisfying C1. Moreover, C2 is satisfied since c belongs to $\partial\mathcal{O}$.

• If rule R3.1 is applied, then c is the circumcenter of a tetrahedron t, and the insertion radius $r(c)$ is the circumradius r of t. According to rule R3.1, r is either greater than $\sigma(c)$ or greater than $\varrho_t\ l_{min}$, where l_{min} is the length of the shortest edge of t. In the first case, we have $r > \sigma(c) > \alpha\ \sigma(c) \geq \sigma'(c)$, since $\alpha < 1$. In the second case, we have $r > \varrho_t\ l_{min}$. Among the vertices of the shortest edge of t, let p be the one inserted last. We have $r(p) \leq l_{min}$, thus $r > \varrho_t\ r(p)$. Moreover, by induction, we have $r(p) \geq \sigma'(p)$. Hence, $r \geq \varrho_t\ \sigma'(p)$. Since σ' is γ-Lipschitz, $\varrho_t\ \sigma'(p)$ is at least $\varrho_t\ (\sigma'(c) - \gamma\ \mathrm{d}(c,p)) = \varrho_t\ (\sigma'(c) - \gamma r)$. It follows that $r \geq \frac{\varrho_t}{1+\varrho_t\gamma}\ \sigma'(c)$, which means that C1 is satisfied for c if ϱ_t satisfies:

$$\varrho_t \geq \frac{1}{1-\gamma} \tag{4}$$

To check C2, we notice that, in both cases ($r > \sigma(c)$ and $r > \varrho_t\ l_{min}$), $r(c)$ is bounded from below by $\frac{\varrho_t}{1+\varrho_t\gamma}\ \sigma'(c)$. Let q be a point of $\partial\mathcal{O}$ closest to c. We have $\delta(c) = d(c,q) \geq \mathrm{d}(c, \mathcal{P}_{|\partial\mathcal{O}}) - \mathrm{d}(q, \mathcal{P}_{|\partial\mathcal{O}})$, where $\mathrm{d}(c, \mathcal{P}_{|\partial\mathcal{O}}) \geq r(c) \geq \frac{\varrho_t}{1+\varrho_t\gamma}\ \sigma'(c)$.

Since rule R3 is applied only when R1 and R2 are fulfilled, Theorem 2 holds right before c is inserted. Hence, $\mathcal{P}_{|\partial\mathcal{O}}$ is a min $\{0.09\ \mathrm{d}_M,\ \alpha(1 + 8.5\mu)\ \sigma\}$-sample of $\partial\mathcal{O}$, which means that $\mathrm{d}(q, \mathcal{P}_{|\partial\mathcal{O}}) \leq \min\ \{0.09\ \mathrm{d}_M(q),\ \alpha(1+ 8.5\mu)\ \sigma(q)\} \leq 3\ \sigma'(q)$. Therefore, $\delta(c) = d(c,q) \geq \frac{\varrho_t}{1+\varrho_t\gamma}\ \sigma'(c) - 3\ \sigma'(q)$. Since σ' is γ-Lipschitz, we have $\sigma'(q) \leq \sigma'(c) + \gamma\ \mathrm{d}(c,q)$, thus $\delta(c) = d(c,q) \geq \frac{1}{1+3\gamma} \left(\frac{\varrho_t}{1+\varrho_t\gamma} - 3\right) \sigma'(c)$. It follows that C2 is satisfied for c if ϱ_t

satisfies[3]:

$$\frac{1}{1+3\gamma}\left(\frac{\varrho_t}{1+\varrho_t\gamma}-3\right)\geq\frac{1}{1-\gamma}\ ,\quad i.e.\ \ \varrho_t\geq\frac{4}{1-5\gamma} \tag{5}$$

• If rule R3.2 is applied, then c is the center of a surface Delaunay ball B, of radius $r=r(c)$, containing the circumcenter c' of a tetrahedron t' of circumradius $r'\geq\frac{\varrho_t}{1+\varrho_t\gamma}\,\sigma'(c')$ (see case R3.1). Since σ' is γ-Lipschitz, we have $\sigma'(c')\geq\sigma'(c)-\gamma\,r(c)$. Moreover, the circumsphere of t' is empty, thus $r'\leq\mathrm{d}(c',p)$, for any point p of $\mathcal{P}$ lying on the bounding sphere of B. Since B contains both p and c', $\mathrm{d}(c',p)$ is at most $2r(c)$. Hence,

$$2r(c)\geq\mathrm{d}(c',p)\geq r'\geq\frac{\varrho_t}{1+\varrho_t\gamma}\left(\sigma'(c)-\gamma\,r(c)\right),\quad i.e.\ \ r(c)\geq\frac{\varrho_t}{2+3\varrho_t\gamma}\,\sigma'(c)$$

Therefore, C1 is satisfied for c if ϱ_t satisfies:

$$\varrho_t\geq\frac{2}{1-3\gamma} \tag{6}$$

Moreover, C2 is satisfied because $c\in\partial\mathcal{O}$.

• To conclude, Conditions C1 and C2 are satisfied for c if Eqs. (3), (4), (5), (6) hold, which is granted if we choose $\gamma<\frac{1}{5}$ (and hence $\alpha<\frac{1}{5}$), $\varrho_f\geq\frac{1}{1-\gamma}$, and $\varrho_t\geq\frac{4}{1-5\gamma}$. □

From now on,we assume that $\alpha<\frac{1}{5}$, $\varrho_f\geq\frac{1}{1-\gamma}$ and $\varrho_t\geq\frac{4}{1-5\gamma}$, where $\gamma=\max\{\alpha,\ 0.03\}$. Given any point p included in $\mathcal{P}_i$ or inserted by the algorithm, we define $B(p)$ as the open ball centered at p, of radius $\rho(p)=\frac{1}{2(1+\gamma)}\,\sigma'(p)$.

Lemma 7 *The balls $\{B(p)\}$ are pairwise disjoint.*

Proof Let p,q be two points of $\mathcal{P}_f$. If p and q belong to $\mathcal{P}_i$, then Definition 6 and Lemma 6 (C1) tell us that $\mathrm{d}(p,q)\geq\frac{1}{2}(r(p)+r(q))\geq\frac{1}{2}(\sigma'(p)+\sigma'(q))$, which is greater than $\frac{1}{2(1+\gamma)}(\sigma'(p)+\sigma'(q))$. It follows that $B(p)$ and $B(q)$ are disjoint.

If now p or q has been inserted in $\mathcal{P}$ during the main loop of the algorithm, then we can assume without loss of generality that p was inserted in $\mathcal{P}$ after q. The distance between p and q is then at least $r(p)$.

[3]The bound on the circumradius to shortest edge ratio ϱ_t given in Eq. (5) corresponds to a rather poor guarantee on tetrahedral shapes, compared to the bound achieved *e.g.* in [4] for polyhedra. In fact, this bound comes from conditions I1 and I2 on the initial point sample, which are prescribed by the surface mesher of [19], and could certainly be improved. Furthermore, for any surface sizing field $\alpha\sigma$ that is small enough compared to the sizing field currently used to generate the initial surface mesh, the bound on ϱ_t can be reduced to $\varrho_t\geq\frac{1+\beta}{1-(2+\beta)\gamma}$, where β tends to 1 as α tends to 0.

By Lemma 6 (C1), we have $r(p) \geq \sigma'(p)$, which is at least $\frac{1}{1+\gamma}\ \sigma'(q)$ since σ' is γ-Lipschitz. Thus, $\mathrm{d}(p,q) \geq \frac{1}{1+\gamma}\ \max\{\sigma'(p), \sigma'(q)\}$. It follows that $\frac{1}{2(1+\gamma)}\ \sigma'(p) + \frac{1}{2(1+\gamma)}\ \sigma'(q) \leq \mathrm{d}(p,q)$, which means that $B(p)$ and $B(q)$ are disjoint. □

To compute an upper bound on the size of the output point sample, we need another result, which states that every ball $B(p)$ intersects $\mathcal{O}$, and that the volume of the part of $B(p)$ included in $\mathcal{O}$ can be lower-bounded.

Lemma 8 *For any $p \in \mathcal{P}_f$, $B(p) \cap \mathcal{O}$ contains a ball of radius $\frac{1}{2}\ \rho(p)$.*

Proof We distinguish between two cases:
• If p lies inside $\mathcal{O}$, then, according to Lemma 6 (C2), the distance $\delta(p)$ from p to $\partial\mathcal{O}$ is at least $\frac{1}{1-\gamma}\ \sigma'(p)$, which is greater than $\frac{1}{2}\ \rho(p)$. Hence, the ball centered at p, of radius $\frac{1}{2}\ \rho(p)$, is included in $\mathcal{O}$.
• Otherwise, p lies on $\partial\mathcal{O}$. There are two medial balls B_i and B_o tangent to $\partial\mathcal{O}$ at p. One of them (say B_i) is included in $\mathcal{O}$, whereas the other one is included in $\mathbb{R}^3 \setminus \mathcal{O}$. Since B_i is a medial ball, its radius is at least $\mathrm{d}_M(p) > \sigma'(p)$. Moreover, the radius of $B(p)$ is $\rho(p) < \sigma'(p)$. It follows that the intersection of $B(p)$ with B_i contains a ball of radius $\frac{1}{2}\ \rho(p)$. □

Using Lemmas 7 and 8, we can now conclude the proof of the theorem.

Proof of Theorem 3. We use a standard scheme [19] and bound the integral of $1/\sigma'^3$ over $\mathcal{O}$. Since $B(p) \cap \mathcal{O} \subseteq \mathcal{O}$ for any $p \in \mathcal{P}_f$, we have

$$\iiint_{\mathcal{O}} \frac{dx}{\sigma'^3(x)} \geq \iiint_{\bigcup_{p\in\mathcal{P}_f}(B(p)\cap\mathcal{O})} \frac{dx}{\sigma'^3(x)}.$$

Moreover, the balls $B(p)$ are pairwise disjoint, by Lemma 7, thus

$$\iiint_{\bigcup_{p\in\mathcal{P}_f}(B(p)\cap\mathcal{O})} \frac{dx}{\sigma'^3(x)} = \sum_{p\in\mathcal{P}_f} \iiint_{(B(p)\cap\mathcal{O})} \frac{dx}{\sigma'^3(x)}.$$

In addition, since σ' is γ-Lipschitz, for any point $x \in B(p)$ we have

$$\sigma'(x) \leq \sigma'(p) + \gamma\ \mathrm{d}(x,p) \leq \sigma'(p) + \gamma\ \rho(p) = \left(1 + \frac{\gamma}{2(1+\gamma)}\right)\ \sigma'(p)$$

It follows that

$$\sum_{p\in\mathcal{P}_f} \iiint_{(B(p)\cap\mathcal{O})} \frac{dx}{\sigma'^3(x)} \geq \sum_{p\in\mathcal{P}_f} \frac{\mathrm{Vol}\left(B(p)\cap\mathcal{O}\right)}{\left(1+\frac{\gamma}{2(1+\gamma)}\right)^3 \sigma'^3(p)}.$$

Now, by Lemma 8, the volume of $B(p) \cap \mathcal{O}$ is at least $\frac{4}{3}\pi\frac{1}{64(1+\gamma)^3}\ \sigma'^3(p)$. This implies that

$$\sum_{p\in\mathcal{P}_f} \frac{\mathrm{Vol}\left(B(p)\cap\mathcal{O}\right)}{\left(1+\frac{\gamma}{2(1+\gamma)}\right)^3 \sigma'^3(p)} \geq \sum_{p\in\mathcal{P}_f} \frac{\frac{\pi}{48(1+\gamma)^3}\ \sigma'^3(p)}{\left(1+\frac{\gamma}{2(1+\gamma)}\right)^3 \sigma'^3(p)} = \frac{\pi}{6\left(2+3\gamma\right)^3}\ |\mathcal{P}_f|,$$

which is at least $\frac{1}{34}\,|\mathcal{P}_f|$ since $\gamma < \frac{1}{5}$. Hence, $|\mathcal{P}_f|$ is at most 34 $\iiint_{\mathcal{O}} \frac{dx}{\sigma'^3(x)}$. Now, $\sigma'(x)$ is defined as the minimum of $\alpha\ \sigma(p)$ and of $0.03\ \sigma_0(p)$, which are positive functions. It follows that $\iiint_{\mathcal{O}} \frac{dx}{\sigma'^3(x)}$ is at most $\iiint_{\mathcal{O}} \frac{dx}{\alpha^3\ \sigma^3(x)}$ + $\iiint_{\mathcal{O}} \frac{dx}{0.03^3\ \sigma_0^3(x)}$, which ends the proof of the theorem.

6 The case of Lipschitz surfaces

Although Sections 4 and 5 focused on the smooth case, our theoretical results hold in a more general setting. In Section 6.1, we introduce the concept of Lipschitz radius and recall some properties of loose ε-samples of Lipschitz surfaces. In Section 6.2, we adapt the algorithm to the Lipschitz setting. We prove its correctness in Section 6.3.

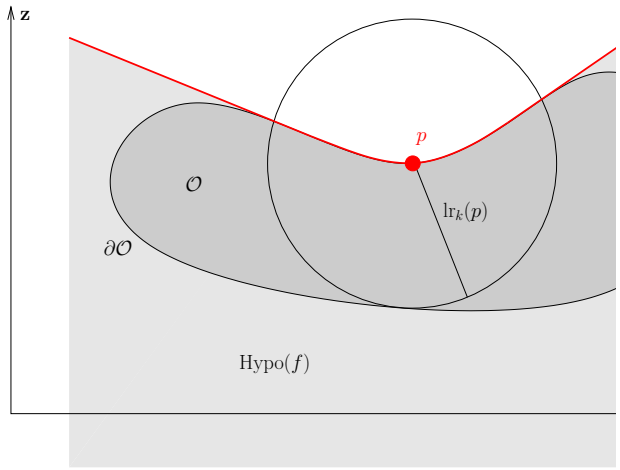

Figure 3 The k-Lipschitz bivariate function f and its associated oriented frame.

6.1 Lipschitz radius and Lipschitz surfaces

In [20], Boissonnat and Oudot introduced the notion of Lipschitz radius of a surface, defined here for $\partial\mathcal{O}$:

Definition 7 *Given a point $p \in \partial\mathcal{O}$, the k-Lipschitz radius of $\partial\mathcal{O}$ at p, or $\mathrm{lr}_k(p)$ for short, is the maximum radius r such that $\mathcal{O} \cap B(p,r)$ is the intersection of $B(p,r)$ with the hypograph of some k-Lipschitz bivariate function f.*

An illustration of this definition is given in Figure 3. Recall that the hypograph of a real-valued bivariate function f is the set of points $(x, y, z) \in \mathbb{R}^3$ such that $z < f(x, y)$. The function f is k-Lipschitz if

$$\forall p, q \in \mathbb{R}^2, \ \frac{|f(p) - f(q)|}{\|p - q\|} \leq k.$$

As proved in [20], $p \mapsto \mathrm{lr}_k(p)$ itself is a 1-Lipschitz function, and as such it is continuous. We call *k-Lipschitz radius of $\partial\mathcal{O}$*, or simply $\mathrm{lr}_k(\partial\mathcal{O})$, the minimum of lr_k over $\partial\mathcal{O}$. This minimum is positive if $\partial\mathcal{O}$ is a k-Lipschitz surface, *i.e.* if $\mathcal{O}$ can be described locally as the hypograph of some k-Lipschitz bivariate function. The class of Lipschitz surfaces is already known to other areas of Mathematics and Computer Science, such as non-smooth analysis [27, §7.3], elliptic PDE theory [28], or geometric measure theory [29, Ch. III]. It includes in particular all piecewise smooth surfaces with bounded normal deviation at singular points (the bound depending on k).

As emphasized in [20], when the surface $\partial\mathcal{O}$ is smooth, lr_k is everywhere at least a fraction of d_M, and it can be arbitrarily large compared to d_M. In the more general case where $\partial\mathcal{O}$ is k-Lipschitz, lr_k plays a role similar to that of d_M in the smooth setting. In particular, (loose) ε-samples share similar properties, provided that ε is sufficiently small compared to $\mathrm{lr}_k(\partial\mathcal{O})$, and that the restricted Delaunay facets are not too skinny:

Theorem 4 [20, Thm 7.1]
If $\partial\mathcal{O}$ is a $\tan\theta$-Lipschitz surface, for some $\theta < 19.1\deg$, and if $\mathcal{P}$ is a loose ε-sample of $\partial\mathcal{O}$, with $\varepsilon \leq 0.09\ \mathrm{lr}_k(\partial\mathcal{O})$ (where $k = \tan\theta$), such that the radius-edge ratios of the facets of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ are less than $\sin\left(\frac{\pi}{3} - \theta\right) / 2\sin\theta$, then the following assertions hold:

(i) *$\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ is a closed 2-manifold ambient isotopic to $\partial\mathcal{O}$,*
(ii) *The Hausdorff distance between $\partial\mathcal{O}$ and $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ is $O\left(\frac{\mu}{\cos^2\theta}\ \mathrm{lr}_k(\partial\mathcal{O})\right)$, where $\mu = \frac{1}{\mathrm{lr}_k(\partial\mathcal{O})}\ \sup_{x \in \partial\mathcal{O}} \varepsilon(x)$,*
(iii) *The normals of $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P})$ approximate the normals of $\partial\mathcal{O}$ within an error of $O(\theta)$,*
(iv) *$\mathcal{P}$ is an $\varepsilon\sqrt{1 + \frac{1}{\cos^4\theta}}$-sample of $\partial\mathcal{O}$.*

Assuming that the restricted Delaunay facets are not too skinny is mandatory to control their normals when the surface is non-smooth. An illustration is given in Figure 4, where the radius-edge ratio of a facet f is too big for the normal of f to be controlled.

Observe that The upper bound on ε used in Theorem 4 is uniform. This is due to the analysis provided in [20], which deals with a uniform upper bound for simplicity. We will stick to this choice in the sequel. Please note however that it is no real loss of generality, since only the upper bound is required to be uniform, not the sampling density ε itself. Using the fact that lr_k is 1-Lipschitz, one can extend the analysis to the case of a non-uniform upper bound on ε, at the price of additional technical details.

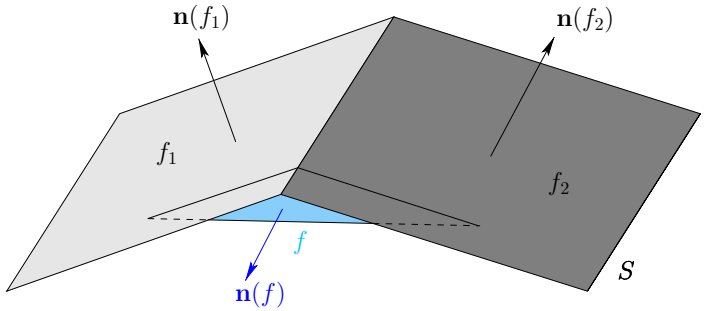

Figure 4 Controlling the normal of a facet.

6.2 Algorithm

The algorithm is mostly the same as in Section 3. The only difference is that d_M is replaced by $\mathrm{lr}_k(\partial\mathcal{O})$ in the initialization phase and in rules R2 and R3.

6.3 Theoretical guarantees

Theorem 5 *Assume that $\partial\mathcal{O}$ is a $\tan\theta$-Lipschitz surface, for some $\theta < 18.6\,\mathrm{deg}$. Assume further that parameters α, ϱ_f and ϱ_t are chosen so as to satisfy the following conditions, where $\gamma = \max\{\alpha,\ 0.03\}$:*

$$\begin{cases} \alpha < \frac{1}{5} \\ \frac{1}{1-\gamma} \leq \varrho_f < \frac{\sin\left(\frac{\pi}{3}-\theta\right)}{2\sin\theta} \\ \varrho_t \geq \frac{4}{1-5\gamma} \end{cases}$$

Then, the algorithm terminates, and the size of the output point set $\mathcal{P}_f$ is:

$$|\mathcal{P}_f| \ = \ O\left(\iiint_{\mathcal{O}} \frac{dx}{\sigma_0^3(x)} \ + \ \frac{1}{\alpha^3} \iiint_{\mathcal{O}} \frac{dx}{\sigma^3(x)}\right),$$

where σ_0 depends only on $\mathcal{O}$ (not on σ). Moreover, $\mathcal{D}_{|\mathcal{O}}(\mathcal{P}_f)$ is a 3-manifold ambient isotopic to $\bar{\mathcal{O}}$, at Hausdorff distance $O\left(\frac{\mu}{\cos^2\theta}\,\mathrm{lr}_k(\partial\mathcal{O})\right)$ from $\bar{\mathcal{O}}$, where $\mu = \min\{0.09,\ \frac{1}{\mathrm{lr}_k(\partial\mathcal{O})}\ \sup_{x\in\partial\mathcal{O}} \alpha\,\sigma(x)\}$.

Since $\gamma = \max\{\alpha,\ 0.03\}$, the hypotheses of the theorem are satisfiable by some α, ϱ_f, and ϱ_t, if and only if $\frac{\sin(\frac{\pi}{3}-\theta)}{2\sin\theta} > \frac{1}{1-0.03}$, which yields the bound of 18.6 deg on θ. In the case where $\partial\mathcal{O}$ is a piecewise smooth surface, Theorem 3.3 of [20] states that the normal deviation at singular points of $\partial\mathcal{O}$ must be less than 32 deg for θ to be less than 18.6 deg. As a result, our algorithm works provably well when applied to piecewise smooth surfaces with normal deviations up to 32 deg. As we will see in Section 8, this bound is rather pessimistic.

The proofs of Theorems 2 and 3 hold the same in the Lipschitz setting (hereby proving Theorem 5), provided that the following changes are made:

– replace d_M by $\mathrm{lr}_k(\partial\mathcal{O})$ (σ_0 is still 1-Lipschitz and σ' is still γ-Lipschitz),
– replace Theorem 1 by Theorem 4,
– replace Lemma 3.6 of [19] by Lemma 1.8 and Remark 1.9 of [30],
– replace Lemma 8 of [22] by Lemma 9 below,
– adapt the proofs of Lemmas 3 and 8 as described below.

Lemma 3.6 of [19] states that, if $\partial\mathcal{O}$ is a smooth surface and $\mathcal{P}$ is an ε-sample of $\partial\mathcal{O}$, with $\varepsilon < \frac{1}{7}\ \mathrm{d}_M$, then every edge of $\mathcal{V}(\mathcal{P})$ intersects S at most once, and transversally. Lemma 1.8 and Remark 1.9 of [30] state an equivalent result in the more general Lipschitz setting.

The rest of Section 6.3 gives more details about some of the changes listed above. It can be skipped in a first reading.

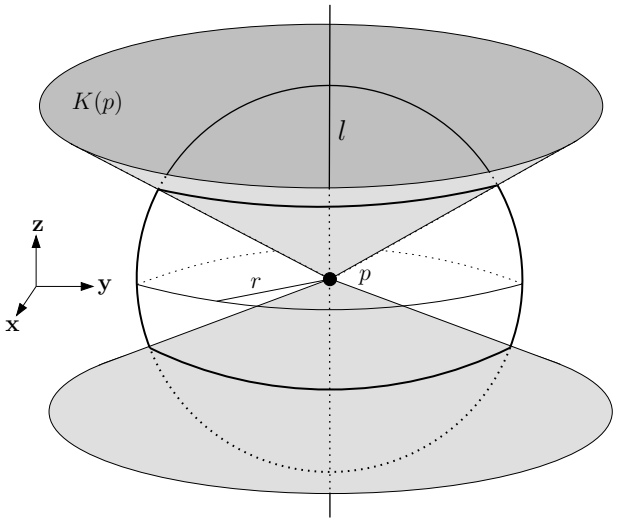

Figure 5 For the proof of Lemma 9.

Lemma 9 *If $\partial\mathcal{O}$ is a k-Lipschitz surface, with $k < 1$, then for any point $p \in \partial\mathcal{O}$ and any $r \leq \mathrm{lr}_k(p)$, $B(p,r) \cap \partial\mathcal{O}$ is a topological disk.*

Proof Let $\theta = \arctan k \in [0, \pi/4[$. Since $\partial\mathcal{O}$ is a k-Lipschitz surface, $B(p,r) \cap \partial\mathcal{O}$ coincides with the graph of a k-Lipschitz bivariate function, defined in some orthonormal frame $(\mathrm{x}, \mathrm{y}, \mathrm{z})$. Therefore, $B(p,r) \cap \partial\mathcal{O}$ is a surface of genus zero with boundaries, or equivalently, it is a set of pairwise-disjoint topological disks possibly with holes. Proving that $B(p,r) \cap \partial\mathcal{O}$ is a topological disk reduces then to showing that the boundary of $B(p,r) \cap \partial\mathcal{O}$ is connected.

Since $\partial\mathcal{O}$ has no boundary, the boundary of $B(p,r) \cap \partial\mathcal{O}$ lies on the sphere $\partial B(p,r)$. Moreover, since $B(p,r) \cap \partial\mathcal{O}$ coincides with the graph of a bivariate function passing through p, at least one component C of the

boundary of $B(p,r) \cap \partial\mathcal{O}$ is a cycle. We will prove that C is the only connected component of the boundary.

According to the so-called Cocone Lemma [20, Lemma 4.1], $B(p,r) \cap \partial\mathcal{O}$ lies outside the double cone $K(p)$ of apex p, of axis l aligned with the z-axis of the frame mentioned above, and of half-angle $\frac{\pi}{2} - \theta$ (see Figure 5). We call *poles* the two points of intersection of l with the bounding sphere of $B(p,r)$, and *meridian* any geodesic curve on the sphere whose endpoints are the poles. Given a meridian m, we know that m can intersect $\partial\mathcal{O}$ only outside $K(p)$. Now, for any two points p, q of $m \setminus K(p)$, the small angle between lines (p,q) and l is at most θ. Hence, p and q cannot both lie on $B(p,r) \cap \partial\mathcal{O}$, because the latter is the graph of a $\tan\theta$-Lipschitz function (with $\theta < \frac{\pi}{4}$) defined over the plane orthogonal to l. It follows that m intersects $\partial\mathcal{O}$ in at most one point. Since this is true for any meridian, and since C is a cycle, C intersects every meridian exactly once, and therefore the boundary of $B(p,r) \cap \partial\mathcal{O}$ cannot have any other connected component. □

Proof of Lemma 3. All we have to do here is to prove that every connected component of $\mathbb{R}^3 \setminus \partial\mathcal{O}$ contains at least one Voronoi vertex, the rest of the proof being the same as in Section 4.

Let us assume that some component Ω of $\mathbb{R}^3 \setminus \partial\mathcal{O}$ contains no Voronoi vertex. Let x be a point of Ω farthest from $\partial\mathcal{O}$. As a local maximum, x is a critical point of the distance to $\partial\mathcal{O}$. It is proved in [20, Thm 3.8] that, since $\partial\mathcal{O}$ is a k-Lipschitz surface, no critical point lies closer to $\partial\mathcal{O}$ than $\frac{1}{2}\ \mathrm{lr}_k(\partial\mathcal{O})$. Therefore, we have $\mathrm{d}(x, \partial\mathcal{O}) \geq \frac{1}{2}\ \mathrm{lr}_k(\partial\mathcal{O})$. Then, by the same argument as in Section 4, we can work out a contradiction with the fact that $\mathcal{P}_{f|\partial\mathcal{O}}$ is a $0.09\mathrm{lr}_k(\partial\mathcal{O})$-sample of $\partial\mathcal{O}$.

Proof of Lemma 8. Only the second part of the proof needs to be revised. Let p be a point of $\mathcal{P}_f \cap \partial\mathcal{O}$. According to the so-called Cocone Lemma [20, Lemma 4.1], $B(p) \cap \partial\mathcal{O}$ lies outside some double cone $K(p)$ of apex p and of half-angle $\frac{\pi}{2} - \theta$. Each half of $B(p) \cap K(p)$ contains a ball of radius $\frac{\cos\theta}{1+\cos\theta}\ \mathrm{lr}_k(\partial\mathcal{O})$, tangent to the boundaries of $B(p)$ and of $K(p)$. These two balls lie on different sides of $\partial\mathcal{O}$, hence one of them is located in $\mathcal{O}$. Therefore, $B(p) \cap \mathcal{O}$ contains a ball of radius $\frac{\cos\theta}{1+\cos\theta}\ \mathrm{lr}_k(\partial\mathcal{O})$, which is greater than $\frac{\cos(18.6\,\mathrm{deg})}{1+\cos(18.6\,\mathrm{deg})}\ \mathrm{lr}_k(\partial\mathcal{O}) > \frac{1}{4(1+\gamma)}\ \sigma'(p) = \frac{1}{2}\ \rho(p)$.

7 Practicality of the algorithm

7.1 Sizing field

The meshing algorithm presented in the previous sections takes as input a sizing field $\sigma : \bar{\mathcal{O}} \to \mathbb{R}^+$ which, for the purpose of the analysis, is assumed to be 1-Lipschitz. In this section, we explain how to deal with user-defined sizing fields that are not 1-Lipschitz or not defined everywhere in $\bar{\mathcal{O}}$.

Let us assume that the user wants a mesh whose grading conforms to a sizing field σ_u that is not 1-Lipschitz. Then we can use the technique of Miller, Talmor and Teng [25] to derive from σ_u a new sizing field σ'_u that is 1-Lipschitz:

$$\forall p \in \bar{\mathcal{O}},\ \sigma'_u(p) = \inf\ \{\mathrm{d}(p,q) + \sigma_u(q) \mid q \in \bar{\mathcal{O}}\}$$

Note that $\sigma'_u(p) \leq \sigma_u(p)$, $\forall p \in \bar{\mathcal{O}}$. The field σ'_u is the best 1-Lipschitz approximation of σ_u [15], because any 1-Lipschitz function that is pointwise at most σ_u is also pointwise at most σ'_u.

The meshing algorithm can be run using the sizing field σ'_u, however it is not necessary to compute σ'_u inside $\mathcal{O}$. Indeed, the algorithm requires an evaluation of the sizing field at internal points only in rule R3, when refining a tetrahedron. A tetrahedron t is refined either when its circumradius is greater than the value of σ at its circumcenter, or when its radius-edge ratio is greater than ϱ_t. A careful look at the proof of termination shows that a positive 1-Lipschitz lower bound on the circumradii of tetrahedra is sufficient for the proof to hold. Since $\sigma'_u(p) \leq \sigma_u(p)$ for any $p \in \mathcal{P}$, the proof still holds if rule R3 is applied only when $r > \sigma_u(p)$. Besides saving some evaluations of σ_u, this variant of the algorithm constructs sparser meshes whose densities conform to the user-defined sizing field, with a grading controlled by the bounds ϱ_f and ϱ_t on the radius-edge ratios of the mesh elements.

In the case where the user has no particular sizing requirements, the 1-Lipschitz sizing field used in the analysis is the field σ_0 defined in Eq. (1), at the beginning of Section 5. Here again, the algorithm does not need to evaluate σ_0 inside $\mathcal{O}$. It may simply skip the size test for tetrahedra and consider for refinement only the tetrahedra with a radius-edge ratio greater than ϱ_t. Since there is less chance that this variant of the algorithm refines a tetrahedron than the original version, it is clear that this variant also terminates. Its output is a mesh whose elements size is a fraction of d_M or $\mathrm{lr}_k(\partial\mathcal{O})$ on $\partial\mathcal{O}$ and grows accordingly with the bounds ϱ_f and ϱ_t when moving towards the inside of the object.

In all cases, the algorithm needs to evaluate d_M or $\mathrm{lr}_k(\partial\mathcal{O})$ to check whether rule R2 is to be applied or not. In the Lipschitz case, we need also to evaluate θ if we want to see whether the input parameters α, ϱ_f and ϱ_t meet the requirements of Theorem 5. These issues are addressed in [19, 20].

7.2 Sliver removal

Optimizing radius-edge ratios prevents our output mesh from containing any bad tetrahedra, except possibly *slivers*. Recall that a sliver is a tetrahedron whose vertices are close to a great circle of its circumsphere and equally spaced along this circle.

Cheng *et al.* [8], and later on Cheng and Dey [9], proposed to exude slivers from the mesh by assigning carefully-chosen weights to the vertices, so that their weighted Delaunay triangulation contains as few slivers as

possible. Li and Teng [10] proposed to avoid slivers by relaxing the choice of refinement vertices inside small areas around the circumcenters of the elements to be refined.

In our context, we use the sliver exudation algorithm of [8] as a post-process. The output mesh is no longer a Delaunay triangulation, but a weighted Delaunay triangulation. Although the theoretically-guaranteed bound on aspect ratios is known to be miserably low [8], the method is efficient in practice and generates almost sliver-free meshes [31].

8 Implementation and results

The algorithm has been implemented in C++, using the geometric library CGAL [32], which provided us with an efficient and flexible implementation of the three-dimensional Delaunay triangulation.

8.1 Smooth case

Figures 6 and 7 show two meshes generated by our algorithm from smooth implicit surfaces. The algorithm is coupled with the post-processing step described in Section 7.2. Each figure is composed of two views of the output mesh: one shows the boundary (top left), the other shows a zoom on the interior, cut by a plane[4] (right). The bottom-left corner of each figure shows the distribution of the inverse aspect ratios of the tetrahedra. The inverse aspect ratio of a tetrahedron compares the radius of the inscribed sphere to the circumradius. It is a fair measure of the tetrahedral shape, tending to zero for any kind of degenerate tetrahedra. In our histograms, the inverse aspect ratios are represented on a linear scale ranging from 0 to $\frac{1}{3}$ (which corresponds to the inverse of the aspect ratio of a regular tetrahedron). The histograms are normalized with respect to area, so that we can make fair comparisons between meshes of different sizes.

In Figure 6, the boundary of the domain is a level set in a 3D gray-scaled image. Its diameter is about 280 millimeters, and its reach approximately 1 millimeter. Although our theoretical results require strict conditions on α, ϱ_f and ϱ_t, in practice the algorithm works well under weaker conditions. For instance, in this example we used a uniform sizing field of 2 millimeters, with $\alpha = 1$, $\varrho_f = 1$ and $\varrho_t = 2$, which is far beyond the theoretical limits. Note that the topology of the domain has been captured, and that the boundary has been accurately approximated.

The inverse aspect ratios distribution of our algorithm (in medium gray) has been superimposed with those obtained by two other algorithms: the unit edge mesher of [12, 13] (in dark gray), and the variational mesher of [15] (in light gray). These two programs, run with our initial surface mesh $\mathcal{D}_{|\partial\mathcal{O}}(\mathcal{P}_i)$ (33,012 vertices) as input, generated approximately the

[4]The screenshots were obtained using Medit [33].

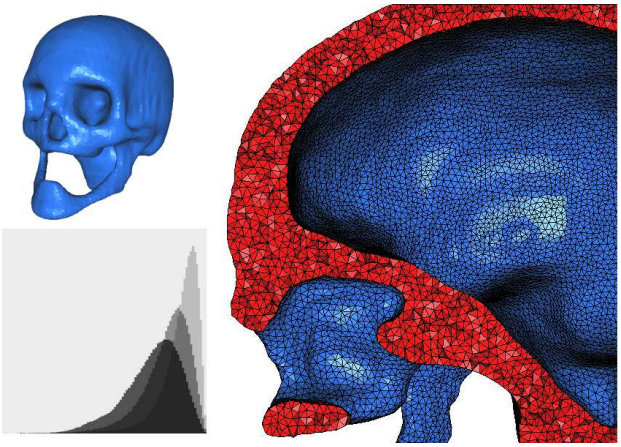

Figure 6 Skull model: 89,245 vertices (among which 35,483 lie on $\partial\mathcal{O}$) and 442,542 tetrahedra.

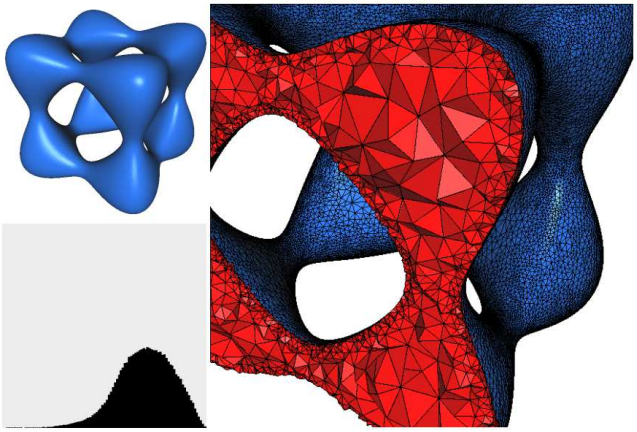

Figure 7 Tanglecube model: 57,293 vertices and 226,010 tetrahedra.

same number of vertices as our mesher. Their running times, on a Pentium IV at 1.7 GHz, are respectively 10 seconds and 10 minutes. The running time of our algorithm on this example is 20 seconds to insert the 53,762 vertices lying inside $\mathcal{O}$ and the 2,471 remaining vertices on $\partial\mathcal{O}$, and 1 minute to exude slivers from the mesh. Compared to the other mesh generators, our algorithm makes a reasonable trade-off between running time and quality of the output. In our mesh, the minimal dihedral angle is 5 deg.

In Figure 7, the boundary of the domain is an algebraic surface of degree four and genus five, called *tanglecube*. We used no sizing field inside the domain and $\sigma_0 = 0.09\ \mathrm{d}_M$ on its boundary, as described in Section 7.1. The bound ϱ_t on the radius-edge ratios was set to 2 (while $\varrho_f = 1$), which enforced the grading of the output mesh. Although the overall appearance

of the inverse aspect ratios distribution is deteriorated due to the non-uniformity of the sizing field, the quality of the output mesh remains quite acceptable. The minimal dihedral angle is 1.1 deg.

8.2 Non-smooth case

Figure 8 shows the output of our algorithm on a non-smooth surface. The boundary of the input domain is a polyhedral surface, shown in Figure 9. The output mesh does not conform exactly to the input polyhedron, but it approximates it for the Hausdorff distance and carries the same topological type. Note also that the boundary of the output mesh is smoother than the input polyhedron, with fewer vertices. We ran the algorithm with a uniform size field and parameter α set to 1. The minimal dihedral angle of the output mesh is 3.2 deg.

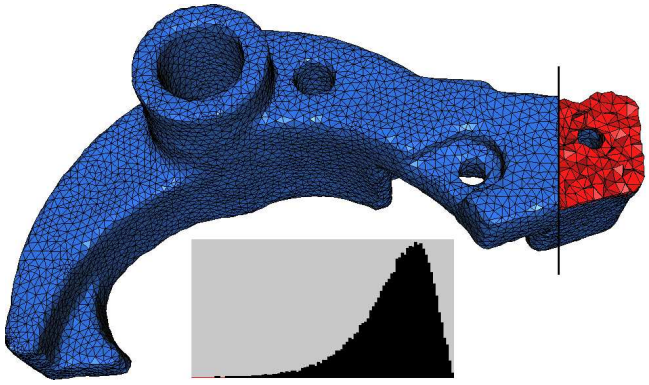

Figure 8 Mechanical model: 15,687 vertices and 71,757 tetrahedra.

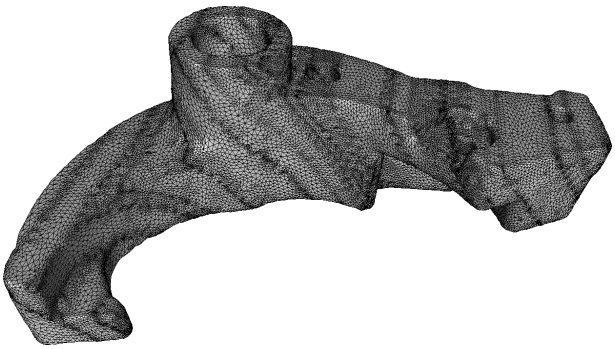

Figure 9 Mechanical model: input polyhedral surface (37,787 vertices).

Figure 10 (left) shows a polyhedral surface containing sharp edges with large normal deviations (about 90 deg). When run with this domain as input, our algorithm produces only crude approximations of the surface in the vicinity of sharp edges, as can be seen in Figure 10 (right), which shows the boundary of the output mesh without edges so that the bad normals approximation around sharp edges appears clearly. This example does not meet the requirements of Section 6 for our theoretical guarantees to hold. Nevertheless, the topology of the input domain is still captured, and the geometric approximation can be made arbitrarily accurate by increasing the sampling density. The normals approximation could be improved for instance by detecting sharp features in a preprocessing step, and then conforming the output mesh to these features using the technique of Cohen-Steiner *et al.* [5].

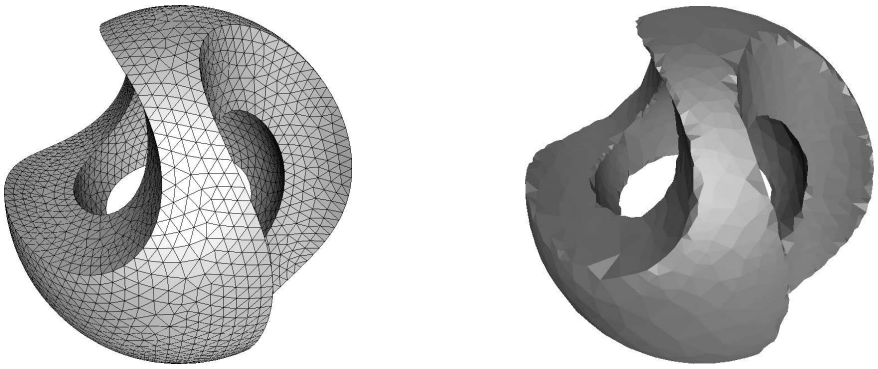

Figure 10 Sculpt model. Left: input surface mesh. Right: output surface mesh, shown without edges.

9 Conclusion and future work

We have introduced a new method for meshing three-dimensional domains bounded by curved surfaces. This method is a combination of existing approaches for meshing smooth or Lipschitz surfaces on the one hand, piecewise linear volumes on the other hand. We have given theoretical guarantees on the output of the algorithm, regarding its size, the quality of its elements, and the accuracy of the approximation of the original object. We have also provided experimental evidence that the algorithm works well in practice.

The main advantage of our method is that it samples the object $\mathcal{O}$ and its boundary $\partial\mathcal{O}$ simultaneously, which allows the user to specify any desired density inside $\mathcal{O}$ and on $\partial\mathcal{O}$. Moreover, the algorithm takes as input the object $\mathcal{O}$ itself, therefore it is independent from any original discretization of $\partial\mathcal{O}$ and it can approximate $\partial\mathcal{O}$ within any desired accuracy. In addition, the required prior knowledge of $\mathcal{O}$ is minimal, since the algorithm needs

only to know the object through an oracle capable of answering two basic geometric questions.

Note that our algorithm is also able to mesh domains with smooth or Lipschitz constraints. The difference between a constraint and a boundary is that both sides of the constraint have to be meshed, whereas only one side of the boundary has to. It turns out that our proofs hold for constraints as well.

Several questions remain open at this point, including:

- The bound in Theorem 3 depends highly on α, whereas the latter influences the density of the mesh only in the vicinity of $\partial\mathcal{O}$. It would be more relevant to devise a bound with two terms: one depending on α and on the integral of $1/\sigma^2$ over $\partial\mathcal{O}$, the other depending on the integral of $1/\sigma^3$ over $\mathcal{O}$.
- Our theoretical guarantees for domains with piecewise smooth boundaries hold as far as the normal deviation remains less than 32 deg. Our experiments show that the practical bound on the normal deviation lies closer to $\frac{\pi}{2}$. However, beyond this limit, our approach fails, even on simple examples. The same phenomenon was previously observed by others in the specific case where the domain to mesh has a polyhedral boundary. One suggested approach for dealing with small dihedral angles was to protect the sharp edges and corners with balls [5]. This approach might be transposable to our context, hereby allowing us to devise an algorithm capable of meshing volumes bounded by piecewise smooth surfaces with higher normal deviations.
- Another significant improvement to this work would be to handle domains with non-manifold constraints. Such an extension would find applications in medical simulation where the membranes of a human organ, as they are extracted from medical imaging data, may often appear as non manifold constraints.
- Finally, we would like to extend our analysis to volumes bounded by hypersurfaces in higher-dimensional Euclidean space. Although many of our arguments generalize straightforwardly to this setting, the keystone of the approach (namely, generating ε-samples of the boundary or constraint hypersurfaces) does not. Moreover, it is still an open question to know whether (loose) ε-samples of hypersurfaces in higher-dimensional space share the same properties as in 3-space.

10 Acknowledgements

This work was partially supported by the European Union through the Network of Excellence AIM@SHAPE Contract IST 506766 and through the IST Programme of the EU as a Shared-cost RTD (FET Open) Project under Contract No IST-006413 (ACS - Algorithms for Complex Shapes).

References

1. D. Mavriplis. Cfd in aerospace in the new millenium. *Canadian Aeronautics and Space Journal*, 46(4):167–176, 2000.
2. J. Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J. Algorithms*, 18:548–585, 1995.
3. J. R. Shewchuk. Delaunay refinement algorithms for triangular mesh generation. *Computational Geometry: Theory and Applications*, 22:21–74, 2002.
4. J. R. Shewchuk. Tetrahedral mesh generation by Delaunay refinement. In *Proc. 14th Annu. ACM Sympos. Comput. Geom.*, pages 86–95, 1998.
5. D. Cohen-Steiner, E. Colin de Verdière, and M. Yvinec. Conforming delaunay triangulations in 3d. *Computational Geometry: Theory and Applications*, pages 217–233, 2004.
6. S.-W. Cheng and S.-H. Poon. Graded conforming delaunay tetrahedralization with bounded radius-edge ratio. In *SODA '03: Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 295–304. Society for Industrial and Applied Mathematics, 2003.
7. S.-W. Cheng, T. K. Dey, E. A. Ramos, and T. Ray. Quality meshing for polyhedra with small angles. In *SCG '04: Proceedings of the twentieth annual symposium on Computational geometry*, pages 290–299. ACM Press, 2004.
8. S.-W. Cheng, T. K. Dey, H. Edelsbrunner, M. A. Facello, and S.-H. Teng. Silver exudation. *J. ACM*, 47(5):883–904, 2000.
9. S.-W. Cheng and T. K. Dey. Quality meshing with weighted delaunay refinement. In *SODA '02: Proceedings of the thirteenth annual ACM-SIAM symposium on discrete algorithms*, pages 137–146, 2002.
10. X.-Y. Li and S.-H. Teng. Generating well-shaped delaunay meshed in 3d. In *SODA '01: Proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms*, pages 28–37. Society for Industrial and Applied Mathematics, 2001.
11. P. J. Frey, H. Borouchaki, and P.-L. George. Delaunay tetrahedrization using an advancing-front approach. In *Proc. 5th International Meshing Roundtable*, pages 31–43, 1996.
12. P.-L. George, F. Hecht, and E. Saltel. Fully automatic mesh generator for 3d domains of any shape. *Impact of Computing in Science and Engineering*, 2:187–218, 1990.
13. P.-L. George, F. Hecht, and E. Saltel. Automatic mesh generator with specified boundary. *Computer Methods in Applied Mechanics and Engineering*, 92:269–288, 1991.
14. A. Scheffer and A. Ungor. Efficient adaptive meshing of parametric models. In *Proc. 6th ACM Sympos. Solid Modeling and Applications*, pages 59–70, 2001.
15. P. Alliez, D. Cohen-Steiner, M. Yvinec, and M. Desbrun. Variational tetrahedral meshing. In *Proceedings SIGGRAPH*, 2005.
16. S. Dey, R. O'Bara, and M. S. Shephard. Curvilinear mesh generation in 3d. In *Proc. 8th Internat. Meshing Roundtable*, pages 407–417, 1999.
17. P. Laug, H. Bourouchaki, and P.-L. George. Maillage de courbes gouverné par une carte de métriques. Technical Report RR-2818, INRIA Rocquencourt, 1996.
18. C. Boivin and C. Ollivier-Gooch. Guaranteed-quality triangular mesh generation for domains with curved boundaries. *International Journal for Numerical Methods in Engineering*, 55(10):1185–1213, 2002.

19. J.-D. Boissonnat and S. Oudot. Provably good sampling and meshing of surfaces. *Graphical Models*, 67(5):405–451, September 2005.
20. J.-D. Boissonnat and S. Oudot. Provably good sampling and meshing of Lipschitz surfaces. In *Proc. 22nd Annu. ACM Sympos. Comput. Geom.*, pages 337–346, 2006.
21. H. Federer. Curvature measures. *Trans. Amer. Math. Soc.*, 93:418–491, 1959.
22. N. Amenta and M. Bern. Surface reconstruction by Voronoi filtering. *Discrete Comput. Geom.*, 22(4):481–504, 1999.
23. S.-W. Cheng, T. K. Dey, E. A. Ramos, and T. Ray. Sampling and meshing a surface with guaranteed topology and geometry. In *Proc. 20th Annu. ACM Sympos. on Computat. Geom.*, pages 280–289, 2004.
24. F. Cazals and J. Giesen. Delaunay triangulation based surface reconstruction: ideas and algorithms. Research Report 5393, INRIA, November 2004.
25. G. L. Miller, D. Talmor, and S.-H. Teng. Data generation for geometric algorithms on non uniform distributions. *International Journal of Computational Geomety and Applications*, 9(6):577–599, 1999.
26. S.-H. Teng and C. W. Wong. Unstructured mesh generation: Theory, practice, and perspectives. *Int. J. Comput. Geometry Appl.*, 10(3):227–266, 2000.
27. F. H. Clarke. *Optimization and Nonsmooth Analysis*. Classics in applied mathematics. SIAM, 1990. Reprint.
28. J. Nečas. *Les méthodes directes en théorie des équations elliptiques*. Masson, 1967.
29. H. Federer. *Geometric Measure Theory*. Springer-Verlag, 1969.
30. S. Oudot. *Sampling and Meshing Surfaces with Guarantees*. Thèse de doctorat en sciences, École Polytechnique, Palaiseau, France, 2005. Preprint available at ftp://ftp-sop.inria.fr/geometrica/soudot/preprints/thesis.pdf.
31. H. Edelsbrunner and D. Guoy. An experimental study of sliver exudation. *Engineering With Computers, Special Issue on 'Mesh Generation' (10th IMR 2001)*, 18(3):229–240, 2002.
32. The CGAL Library. Release 3.1 (`http://www.cgal.org`).
33. Medit, a scientific visualization tool (`http://www-rocq.inria.fr/gamma/medit/medit.html`).