5A.1

Meshing 3D Domains Bounded by Piecewise Smooth Surfaces*

Laurent Rineau[1] and Mariette Yvinec[2]

[1] INRIA, BP 93 06902 Sophia Antipolis, France.
Laurent.Rineau@sophia.inria.fr
[2] INRIA, BP 93 06902 Sophia Antipolis, France.
Mariette.Yvinec@sophia.inria.fr

Summary. This paper proposes an algorithm to mesh 3D domains bounded by piecewise smooth surfaces. The algorithm may handle multivolume domains defined by non connected or non manifold surfaces. The boundary and subdivision surfaces are assumed to be described by a complex formed by surface patches stitched together along curve segments.

The meshing algorithm is a Delaunay refinement and it uses the notion of restricted Delaunay triangulation to approximate the input curve segments and surface patches. The algorithm yields a mesh with good quality tetrahedra and offers a user control on the size of the tetrahedra. The vertices in the final mesh have a restricted Delaunay triangulation to any input feature which is a homeomorphic and accurate approximation of this feature. The algorithm also provides guarantee on the size and shape of the facets approximating the input surface patches. In its current state the algorithm suffers from a severe angular restriction on input constraints. It basically assumes that two linear subspaces that are tangent to non incident and non disjoint input features on a common point form an angle measuring at least 90 degrees.

1 Introduction

Mesh generation is a notoriously difficult task. Getting a fine discretization of the domain of interest is the bottleneck of many applications in the areas of computer modelling, simulation or scientific computation. The problem of mesh generation is made even more difficult when the domain to be meshed is bounded and structured by curved surfaces which have to be approximated as well as discretized in the mesh. This paper deals with the problem of generating unstructured tetrahedral meshes for domains bounded by piecewise smooth surfaces. A common way to tackle such a problem consists in building

*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).

first a triangular mesh approximating the bounding surfaces and then refining the volume discretization while preserving the boundary approximation. The meshing of bounding surfaces are mostly performed through the highly popular marching cubes algorithm [LC87]. The marching cubes algorithm provides an accurate discretization of smooth surfaces but the output surface mesh includes poor quality elements and fails to recover sharp features. The marching cubes may be followed by some remeshing step to improve the shape of the elements and to adapt the sizing of the surface mesh to the required density, see [AUGA05] for survey on surface remeshing. Once a boundary surface mesh is obtained, the original surface is replaced by its piecewise linear approximation. The 3-dimensional mesh is then obtained through a meshing algorithm which either conforms strictly to the boundary surface mesh (see e.g. [FBG96, GHS90, GHS91]) or refine the surface mesh within the geometry of the piecewise linear approximation [She98, CDRR04]. See e.g. [FG00] for a survey on 3-dimensional meshing. In both cases, the quality of the resulting mesh and the accuracy of the boundary approximation highly depends on the initial surface mesh.

This paper proposes an alternative to the marching cubes strategy. In this alternative, the recovery of bounding curves and surfaces is based on the notion of restricted Delaunay triangulations and the mesh generation algorithm is a multi-level Delaunay refinement process which interleaves the refinement of the curve, surface and volume discretization.

Delaunay refinement is recognized as one of the most powerful method to generate meshes with guaranteed quality. The pioneering work of Chew [Che89] and Ruppert [Rup95] handles the generation of 2-dimensional meshes for domains whose boundaries and constraints do not form small angles. Shewchuk improves the handling of small angles in two dimensions [She02] and generalizes the method to generate 3-dimensional meshes for domains with piecewise linear boundaries [She98]. The handling of small angles formed by constraints is more puzzling in three dimensions, where dihedral angles and facet angles come into play. Using the idea of protecting spheres around sharp edges [MMG00, CCY04], Cheng and Poon [CP03] provide a thorough handling of small input angles formed by boundaries and constraints. Cheng, Dey, Ramos, and Ray [CDRR04] turn the same idea into a simpler and practical mesh generation algorithm for polyhedral domains. In 3-dimensional space, Delaunay refinement produces tetrahedral meshes free of all kind of degenerate tetrahedra except slivers. Further work [CDE$^+$00, CD03, LT01, CDRR05] was needed to deal with the problem of sliver exudation.

Up to now, little work has been dealing with curved objects. The early work of Chew [Che93] tackles the problem of meshing curved surfaces and Boivin and Ollivier-Gooch [BOG02] consider the meshing of 2-dimensional domains with curved boundaries. In [ORY05], we propose a Delaunay refinement algorithm to mesh a 3-dimensional domain bounded by a smooth surface. This algorithm relies on recent results on surface mesh generation [BO05, BO06]. It involves Delaunay refinement techniques and the notion of restricted Delaunay triangulations to provide a nice sampling of both the volume and the bounding surface. The restriction to the boundary surface of the Delaunay triangulation of the final mesh vertices forms an homeomorphic and accurate approximation of this surface. The present paper extends this mesh generation algorithm to handle 3-dimensional domains defined by piecewise smooth surfaces, i. e. patches of smooth surfaces stitched together along 1-dimensional smooth curved segments. The main idea is to approximate the 1-dimensional sharp features of boundary surfaces by restricted Delaunay triangulations. The accuracy of this approximation is controlled by a few additional refinement rules in the Delaunay refinement process. The algorithm produces a controlled quality mesh in which each surface patch and singular curved segment has a homeomorphic piecewise linear approximation at a controlled Hausdorff distance. The algorithm can handle multi-volume domains defined by non connected or non manifold piecewise smooth surfaces. The algorithm relies only on a few oracles able to detect and compute intersection points between straight segments and surface patches or between straight triangles and 1-dimensional singular curved segments. Therefore it can be used in various situations like meshing CAD-CAM models, molecular surfaces or polyhedral models. The only severe restriction with respect to the input features is an angular restriction. Roughly speaking, tangent planes at a common point of two adjacent surface patches are required to make an angle bigger than 90°.

Our work is very closed to a recent work [CDR07] where Cheng, Dey and Ramos propose a Delaunay refinement meshing for piecewise smooth surfaces. This algorithm suffers no angular restriction but uses topology driven refinement rules which involve computationally intensive and hard to implement predicates on the surface.

The paper is organized as follows. Section 2 precises the input of the algorithm and provides a few definitions. In particular we define a local feature size adapted to the case of piecewise smooth surfaces. The mesh generation algorithm is described in section 3. Before proving in section 5 the correctness of this algorithm, i. e. basically the fact that it always terminates, we prove in section 4 the accuracy, quality and homeomorphism properties of the resulting mesh. The algorithm has been implemented using the library CGAL [CGAL]. Section 6 provides some implementation details and shows a few examples. Section 7 gives some directions for future work, namely to get rid of the angular restriction on input surface patches.

2 Input, definitions and notations

The domain $\mathcal{O}$ to be meshed is assumed to be a union of 3-dimensional cells whose boundaries are piecewise smooth surfaces, i. e. formed by smooth surface patches joining along smooth curve segments.

More precisely, we define a regular complex as a set of closed manifolds, called faces, such that:

- the boundary of each face is the union of faces of the complex,
- any two faces have disjoint interior.

We consider a 3-dimensional regular complex whose 2-dimensional subcomplex is formed by patches of smooth surfaces and whose 1-dimensional skeleton is formed by smooth curve segments. Each curve segment is assumed to be a compact subset of a smooth closed curve, and each surface patch is assumed to be a compact subset of a smooth closed surface. The smoothness conditions on curves (resp. surfaces) is to be $C^{1,1}$, *i.e.* to be differentiable with a Lipschitz condition on the tangent (resp. normal) field.

The domain $\mathcal{O}$ that we consider is a union of cells, i. e. 3-dimensional faces, in such a regular complex.

We denote by $\mathcal{F}$ the 2-dimensional regular complex which forms the boundaries of the cells in $\mathcal{O}$. The set of faces in $\mathcal{F}$ includes a set $\mathcal{Q}$ of vertices, a set $\mathcal{L}$ of smooth curve segments and a set $\mathcal{S}$ of smooth surface patches, such that $\mathcal{F} = \mathcal{Q} \cup \mathcal{L} \cup \mathcal{S}$.

In the following, $\bigcup \mathcal{F}$ denotes the domain covered by the union of faces in $\mathcal{F}$ and $\mathrm{d}(x, y)$ the Euclidean distance between two points x and y.

We assume that two elements in $\mathcal{F}$ which are neither disjoint nor incident do not form sharp angles. More precisely, we assume the following:

Definition 1 (The angular hypothesis). *There is a distance* λ_0 *so that, for any pair* (F, G) *of non disjoint and non incident faces in* $\mathcal{F}$*, if there is a point* z *on* $F \cap G$ *such that* $\mathrm{d}(x, z) \leq \lambda_0$ *and* $\mathrm{d}(y, z) \leq \lambda_0$*, then the following inequality holds:*

$$\mathrm{d}(x, y)^2 \geq \mathrm{d}(x, F \cap G)^2 + \mathrm{d}(y, F \cap G)^2. \tag{1}$$

In the special case of linear faces, the angular hypothesis holds when the *projection condition* [She98] holds. The projection condition states that if two elements F and G of $\mathcal{F}$ are neither disjoint nor incident, the orthogonal projection of G on the subspace spanned by F does not intersect the interior of F. For two adjacent planar facets, it means that the dihedral angle must be at least 90°.

Definition of the local feature size

To describe the sizing field used by the algorithm we need to introduce a notion of *local feature size* (lfs , for short) related to the notion of local feature size used for polyhedra [Rup95, She98] and also to the local feature size introduced [AB99] for smooth surfaces. To account for curvature, we first define a notion of *interrelated points*, (using an idea analogous to the definition of *intertwined* points for anisotropic metric in [LS03]).

Definition 2 (Interrelated points). *Two points* x *and* y *of* $\mathcal{S}$ *are said to be* interrelated *if:*

- *either they lie on a common face* $F \in \mathcal{F}$*,*
- *or they lie on non-disjoint faces,* F *and* G*, and there exists a point* w *in the intersection* $F \cap G$ *such that:* $\mathrm{d}(x, w) \leq \lambda_0$ *and* $\mathrm{d}(y, w) \leq \lambda_0$.

We first define a feature size $\mathrm{lfs}^{\mathrm{P}}(x)$ analog to the feature size used for a polyhedron. For each point $x \in \mathbb{R}^3$, $\mathrm{lfs}^{\mathrm{P}}(x)$ is the radius of the smallest ball centered at x that contains two non interrelated points of $\bigcup \mathcal{F}$.

We then define a feature size $\mathrm{lfs}_{F_i}(x)$ related to each feature in $\mathcal{L} \cup \mathcal{S}$. Let F_i be a curve segment or surface patch in $\mathcal{F}$. We first define the function $\mathrm{lfs}_{F_i}(x)$ for any point $x \in F_i$ as the distance from x to the medial axis of the smooth curve or the smooth surface including the face F_i. Thus defined, the function $\mathrm{lfs}_{F_i}(x)$ is a Lipschitz function on F_i. Using the technique of Miller, Talmor and Teng [MTT99], we extend it as a Lipschitz function $\mathrm{lfs}_{F_i}(x)$ defined on $\mathbb{R}^3$:

$\forall x \in \mathbb{R}^3,\ \mathrm{lfs}_{F_i}(x) = \inf\ \{\mathrm{d}(x, x') + \mathrm{lfs}_{F_i}(x')\ :\ x' \in F_i\}$.

The local feature size $\mathrm{lfs}(x)$ used below is defined as the pointwise minimum:

$$\mathrm{lfs}(x) = \min\left(\mathrm{lfs}^{\mathrm{P}}(x),\ \min_{F_i \in \mathcal{F}} \mathrm{lfs}_{F_i}(x)\right).$$

Being a pointwise minimum of Lipschitz functions, $\mathrm{lfs}(x)$ is a Lipschitz function.

3 The mesh generation algorithm

The meshing algorithm is based on the notion of Voronoi diagrams, Delaunay triangulations and restricted Delaunay triangulations. Let $\mathcal{P}$ be a set of points and p a point in $\mathcal{P}$. The Voronoi cell $V(p)$ of the point p is the locus of points that are closer to p than to any other point in $\mathcal{P}$. For any subset $\mathcal{T} \subset \mathcal{P}$, the Voronoi face $V(\mathcal{T})$ is the intersection $\bigcap_{p \in \mathcal{T}} V(p)$. The Voronoi diagram $\mathcal{V}(\mathcal{P})$ is the complex formed by the non empty Voronoi faces $V(\mathcal{T})$ for $\mathcal{T} \subset \mathcal{P}$.

We use $\mathcal{D}(\mathcal{P})$ to denote the Delaunay triangulation of $\mathcal{P}$. Let X be a subset of $\mathbb{R}^3$. We call restricted Delaunay triangulation of $\mathcal{P}$ to X, and denote by $\mathcal{D}_X(\mathcal{P})$, the subcomplex of $\mathcal{D}(\mathcal{P})$ formed by the faces in $\mathcal{D}(\mathcal{P})$ whose dual Voronoi faces have a non empty intersection with X. Thus a triangle pqr of $\mathcal{D}(\mathcal{P})$ belongs to $\mathcal{D}_X(\mathcal{P})$ iff the dual Voronoi edge $V(p, q, r)$ intersects X and an edge pq of $\mathcal{D}(\mathcal{P})$ belongs to $\mathcal{D}_X(\mathcal{P})$ iff the dual Voronoi facet $V(p, q)$ intersects X.

The algorithm is a Delaunay refinement algorithm that iteratively builds a set of sample points $\mathcal{P}$ and maintains the Delaunay triangulation $\mathcal{D}(\mathcal{P})$, its restriction $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ to the domain $\mathcal{O}$ and the restrictions $\mathcal{D}_{|S_k}(\mathcal{P})$ and $\mathcal{D}_{|L_j}(\mathcal{P})$ to every facet S_k of $\mathcal{S}$ and every edge L_j of $\mathcal{L}$. At the end of the refinement process, the tetrahedra in $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ form the final mesh and the subcomplexes $\mathcal{D}_{|S_k}(\mathcal{P})$ and $\mathcal{D}_{|L_j}(\mathcal{P})$ are accurate approximation of respectively S_k and L_j.

The refinement rules applied by the algorithm to reach this goal use the, hereafter defined, notion of encroachment for restricted Delaunay facets and edges.

Let L_j be an edge of $\mathcal{L}$. For each edge qr of the restricted Delaunay triangulation $\mathcal{D}_{|L_j}(\mathcal{P})$, there is at least one ball, centered on L_j, whose bounding sphere passes through q and r and with no point of $\mathcal{P}$ in its interior. Such a ball is centered on a point of the non empty intersection $L_j \cap V(q,r)$ and called here after a *restricted Delaunay ball.* The edge qr of $\mathcal{D}_{|L_j}(\mathcal{P})$ is said to be *encroached* by a point x if x is in the interior of a restricted Delaunay ball of qr.

Likewise, for each triangle qrs of the restricted Delaunay triangulation $\mathcal{D}_{|S_k}(\mathcal{P})$, there is at least one ball, centered on the patch S_k, whose bounding sphere passes through q, r and s and with no point of $\mathcal{P}$ in its interior. Such a ball is called a restricted Delaunay ball (or a *surface Delaunay ball* in this case). The triangle qrs of $\mathcal{D}_{|S_k}(\mathcal{P})$ is said to be encroached by a point x if x is in the interior of a surface Delaunay ball of qrs.

The algorithm takes as input

- the piecewise smooth complex $\mathcal{F}$ describing the boundary of the volume to be meshed.
- A sizing field $\sigma(x)$ defined over the domain to be meshed. The sizing field is assumed to be a Lipschitz function such that for any point $x \in \mathcal{F}$, $\sigma(x) \le \mathrm{lfs}(x)$.
- Some shape criteria in the form of upper bounds β_3 and β_2 for the radius-edge ratio of respectively the tetrahedra in the mesh and the boundary facets.
- Two parameters α_1 and α_2 such that $\alpha_1 \le \alpha_2 \le 1$ whose values will be discussed in section 5.

The algorithm begins with a set of sample points $\mathcal{P}_0$ including $\mathcal{Q}$, at least two points on each segment of $\mathcal{L}$ and at least three points on each patch of $\mathcal{S}$.

At each step, a new sample point is added to the current set $\mathcal{P}$. The new point is added according to the following rules, where rule R_i is applied only when no rule R_j with $j < i$ can be applied. Those rules issue calls to sub-procedures, respectively called `refine-edge`, `refine-facet-or-edge`, and `refine-tet-facet-or-edge`, which are described below.

R1 If, for some L_j of $\mathcal{L}$, there is an edge e of $\mathcal{D}_{|L_j}(\mathcal{P})$ whose endpoints do not both belong to L_j, call `refine-edge`(e).

R2 If, for some L_j of $\mathcal{L}$, there is an edge e of $\mathcal{D}_{|L_j}(\mathcal{P})$ with a restricted Delaunay ball $B(c_e, r_e)$ that does not satisfy $r_e \le \alpha_1 \sigma(c_e)$, call `refine-edge`($e$).

R3 If, for some S_k of $\mathcal{S}$, there is a facet f of $\mathcal{D}_{|S_k}(\mathcal{P})$ whose vertices do not all belong to S_k, call `refine-facet-or-edge`(f).

R4 If, for some S_k of $\mathcal{S}$, there is a facet f of $\mathcal{D}_{|S_k}(\mathcal{P})$, and a restricted Delaunay ball $B(c_f, r_f)$ of f with radius-edge ratio ρ_f, such that

R4.1 either the size criteria, $r_f \le \alpha_2 \sigma(c_f)$, is not met,

R4.2 or the shape criteria, $\rho_f \leq \beta_2$, is not met,
call `refine-facet-or-edge`(f).

R5 If there is some tetrahedron t in $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$, with Delaunay ball $B(c_t, r_t)$ and radius edge ratio ρ_t, such that

R5.1 either the size criteria, $r_t \leq \sigma(c_t)$, is not met,

R5.2 or the shape criteria $\rho_t \leq \beta_3$ is not met,
call `refine-tet-facet-or-edge`(t).

refine-edge. The procedure `refine-edge`(e) is called for an edge e of the restricted Delaunay triangulation $\mathcal{D}_{|L_j}(\mathcal{P})$ of some edge L_j in $\mathcal{L}$. The procedure inserts in $\mathcal{P}$ the center c_e of the restricted Delaunay ball $B(c_e, r_e)$ of e with largest radius r_e.

refine-facet-or-edge. The procedure `refine-facet-or-edge`(f) is called for a facet f of the restricted Delaunay triangulation $\mathcal{D}_{|S_k}(\mathcal{P})$ of some facet S_k in $\mathcal{S}$. The procedure considers the center c_f of the restricted Delaunay ball $B(c_f, r_f)$ of f with largest radius r_f and performs the following:
- if c_f encroaches some edge e in $\cup_{L_j \in \mathcal{L}} \mathcal{D}_{|L_j}(\mathcal{P})$, call `refine-edge`$(e)$,
- else add c_f in $\mathcal{P}$.

refine-tet-facet-or-edge. The procedure `refine-tet-facet-or-edge`(t) is called for a cell t of $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$. It considers the circumcenter c_t of t and performs the following:
- if c_t encroaches some edge e in $\cup_{L_j \in \mathcal{L}} \mathcal{D}_{|L_j}(\mathcal{P})$, call `refine-edge`$(e)$,
- else if c_t encroaches some facet f in $\cup_{S_k \in \mathcal{L}} \mathcal{D}_{|S_k}(\mathcal{P})$,
  call `refine-facet-or-edge`(f),
- else add c_t in $\mathcal{P}$.

4 Output Mesh

At the end of the refinement process, the tetrahedra in $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ form the final mesh and the features of $\mathcal{L}$ and $\mathcal{S}$ are approximated by their respective restricted Delaunay triangulation. In this section we assume that the refinement process terminates, and we prove that after termination, each connected component O_i of the domain O is represented by a submesh formed with well sized and well shaped tetrahedra and that the boundary of this submesh is an accurate and homeomorphic approximation of $\mathrm{bd}\, O_i$.

Theorem 1. *If the meshing algorithm terminates, the output mesh $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ has the following properties.*

Size and shape. *The tetrahedra in $\mathcal{D}_{|\mathcal{O}}(\mathcal{P})$ conform to the input sizing field and are well shaped (meaning that their radius-edge ratio is bounded by β_3).*

Homeomorphism. *There is an homeomorphism between $\bigcup \mathcal{F}$ and $\mathcal{D}_{\bigcup \mathcal{F}}(\mathcal{P})$ such that each face F of $\mathcal{F}$ is mapped to its restricted Delaunay triangulation $\mathcal{D}_F(\mathcal{P})$. Furthermore, for each connected component O_i of the domain*

O, the boundary $\operatorname{bd} O_i$ *of* O_i *is mapped to the boundary* $\operatorname{bd} \mathcal{D}_{O_i}(\mathcal{P})$ *of the submesh* $\mathcal{D}_{O_i}(\mathcal{P})$*.*

Hausdorff distance. *For each face (curve segment or surface patch) F in $\mathcal{F}$, the Hausdorff distance between the restricted Delaunay triangulation $\mathcal{D}_F(\mathcal{P})$ and F is bounded.*

Proof. The first point is a direct consequence of rules R5.2 and R5.1. The rest of this section is devoted respectively to the proof of the homeomorphism properties and to the proof of Hausdorff distance.

The extended closed ball property.

To prove the homeomorphism between $\bigcup \mathcal{F}$ and $\mathcal{D}_{\bigcup \mathcal{F}}(\mathcal{P})$ we make use of the Edelsbrunner and Shah theorem [ES97]. In fact, because the union $\bigcup \mathcal{F}$ is not assumed to be a manifold topological space, we make use of the version of Edelsbrunner and Shah theorem for non manifold topological spaces. This theorem is based on an extended version of the closed balled property recalled here for completeness.

Definition 3 (Extended closed ball property). *A CW complex is a regular complex whose faces are topological balls. A set of point $\mathcal{P}$ is said to have the extended closed ball property with respect to a topological space X of $\mathbb{R}^d$ if there is a CW complex $\mathcal{R}$ with $X = \bigcup \mathcal{R}$ and such that, for any subset $\mathcal{T} \subseteq \mathcal{P}$ whose Voronoi face $V(\mathcal{T}) = \bigcap_{p \in \mathcal{T}} V(p)$ has a non empty intersection with X, the following holds.*

1. *There is a CW subcomplex $\mathcal{R}_\mathcal{T} \subset \mathcal{R}$ such that $\bigcup \mathcal{R}_\mathcal{T} = V(\mathcal{T}) \cap X$.*
2. *Let $\mathcal{R}^0_\mathcal{T}$ be the subset of faces in $\mathcal{R}_\mathcal{T}$ whose interior is included in the interior of $V(\mathcal{T})$. There is a unique face $F_\mathcal{T}$ of $\mathcal{R}_\mathcal{T}$ which is included in all the faces $\mathcal{R}^0_\mathcal{T}$.*
3. *If $F_\mathcal{T}$ has dimension j, $F_\mathcal{T} \cap \operatorname{bd} V(\mathcal{T})$ is a $j-1$-sphere.*
4. *For each face $F \in \mathcal{R}^0_\mathcal{T} \setminus \{F_\mathcal{T}\}$ with dimension k, $F \cap \operatorname{bd} V(\mathcal{T})$ is a $k-1$ ball.*

Furthermore, $\mathcal{P}$ is said to have the extended generic intersection property for X if for every subset $\mathcal{T} \subseteq \mathcal{P}$ and every face $F' \in \mathcal{R}_\mathcal{T} \setminus \mathcal{R}^0_\mathcal{T}$ there is a face $F \in \mathcal{R}^0_\mathcal{T}$ such that $F' \subseteq F$.

Theorem 2 ([ES97]). *If X is a topological space and $\mathcal{P}$ a finite point set that has the extended generic intersection property and the extended closed ball property for X, X and $\mathcal{D}_X(\mathcal{P})$ are homeomorphic.*

In the following, we consider the final sampling $\mathcal{P}$ produced by the meshing algorithm and we show that $\mathcal{P}$ has the extended closed ball property with respect to $\bigcup \mathcal{F}$. For this, we need a CW complex $\mathcal{R}$ whose domain coincides with $\bigcup \mathcal{F}$. We define $\mathcal{R}$ as $\mathcal{R} = \{V(\mathcal{T}) \cap F : \mathcal{T} \subseteq \mathcal{P}, F \in \mathcal{F}\}$. and our first goal is therefore to prove that each face in this complex is a topological ball.

Surface sampling

The following lemmas are borrowed from the recently developed surface sampling theory [ACDL02, BO05]. They are hereafter adapted to our setting where the faces $S_k \in \mathcal{S}$ are patches of smooth closed surfaces.

Lemma 1 (Topological lemma). *[ACDL02] For any point $x \in \bigcup \mathcal{F}$, any ball $B(x,r)$ centered on x and with radius $r \leq \mathrm{lfs}(x)$ intersects any face F of $\mathcal{F}$ including x according to a topological ball.*

Lemma 2 (Long distance lemma). *[Dey06] Let x be a point in a face S_k of $\mathcal{S}$. If a line l through x makes a small angle $(l, l(x)) \leq \eta$ with the line $l(x)$ normal to S_k at x and intersects S_k in some other point y, $\mathrm{d}(x,y) \geq 2\mathrm{lfs}(x)\cos(\eta)$.*

Lemma 3 (Chord angle lemma). *[ACDL02] For any two points x and y of S_k with $\mathrm{d}(x,y) \leq \eta\,\mathrm{lfs}(x)$, with $\eta \leq 2$, the angle between xy and the tangent plane of S_k at x, is at most $\arcsin\frac{\eta}{2}$.*

Lemma 4 (Normal variation lemma). *[ACDL02] Let x and y be two points of S_k with $\mathrm{d}(x,y) \leq \eta \min(\mathrm{lfs}(x), \mathrm{lfs}(y))$, with $\eta \leq 2$. Let $n(x)$ and $n(y)$ be the normal vectors to S_k at x and y respectively. Assume that $n(x)$ and $n(y)$ are oriented consistently, for instance toward the exterior of the smooth closed surface including S_k. Then the angle $(n(x), n(y))$ is at most $2\arcsin\frac{\eta}{2}$.*

Lemma 5 (Facet normal lemma). *[ACDL02] Let p, q, r be three points of S_k, such that the circumradius of triangle pqr is at most $\eta\,\mathrm{lfs}(p)$. The angle between the line l_f normal to triangle pqr and the line $l(p)$ normals to S_k in p is at most $\arcsin(\eta\sqrt{3}) + \arcsin\frac{\eta}{1-\eta}$.*

At the end of the algorithm, the sampling $\mathcal{P}$ is such that for any patch S_k of $\mathcal{S}$, the subset $\mathcal{P} \cap S_k$ is a *loose α_2-sample* of S_k. This means that any restricted Delaunay ball $B(c,r)$, circumscribed to a face of the restricted Delaunay triangulation $\mathcal{D}_{|S_k}(\mathcal{P} \cap S_k)$, has its radius r bounded by $\alpha_2\mathrm{lfs}(c)$. It is proved in [BO06] that any loose ε-sample of a patch S_k is a ε'-sample of S_k for $\varepsilon' = \varepsilon(1 + O(\varepsilon))$. The following lemma are proved in [Boi06] for closed smooth surfaces, but their proof can be easily adapted in the case of surface patches.

Lemma 6 (Projection lemma). *[Boi06] Let $\mathcal{P}_k$ be a loose ε-sample of the smooth surface patch S_k for $\varepsilon < 0.24$. Any pair f and f' of two facets of $\mathcal{D}_{|S_k}(\mathcal{P}_k)$ sharing a common vertex p, have non overlapping orthogonal projections onto the tangent plane at p i.e. the projections of the relative interiors of f and f' do not intersect.*

Lemma 7 (Small cylinder lemma). *[Boi06] Let $\mathcal{P}_k$ be a loose $\varepsilon-$*sample *of the smooth surface patch S_k. For any point $p \in \mathcal{P}_k$, if $V_{\mathcal{P}_k}$ denotes the cell* *of p in the diagram $\mathcal{V}(\mathcal{P}_k)$, the intersection $V_{\mathcal{P}_k}(p) \cap S_k$ is contained in a small cylinder with axis $l(p)$, height $h = O(\varepsilon^2)\mathrm{lfs}(p)$ and radius $O(\varepsilon\,\mathrm{lfs}(p))$, where $l(p)$ is the line normal to S_k at p.*

The same result holds if the function lfs *is replaced by any Lipschitz sizing field σ with $\sigma(x) \leq \mathrm{lfs}(x)$, both in the definition of loose $\varepsilon-$*sample *and in the description of the small cylinder.*

Proof of the homeomorphism properties

The following lemmas are related to the final sampling $\mathcal{P}$. They assume that the sizing field $\sigma(x)$ is less than $\mathrm{lfs}(x)$ for any point $x \in \mathcal{F}$ and that the constant α_1 and α_2 used in the algorithm are small enough.

Lemma 8.
- *A facet $V(p,q)$ of $\mathcal{V}(\mathcal{P})$ intersects a curve segment L_j of $\mathcal{L}$ iff p and q are consecutive vertices on L_j. Any facet in $\mathcal{V}(\mathcal{P})$ intersects at most one curve segment in $\mathcal{L}$. If non empty, the intersection $V(p,q) \cap L_j$ is a single point, i. e. a 0-dimensional topological ball.*
- *For any cell $V(p) \in \mathcal{P}$, the intersection $V(p) \cap L_j$ is empty if $p \notin L_j$ and a single curve segment, i.e a 1-dimensional topological ball, otherwise.*

Proof. Proof omitted in this abstract.

Lemma 9.
- *An edge $V(p,q,r)$ of $\mathcal{V}(\mathcal{P})$ may intersect the surface patch S_k only if p, q and r belong to S_k. An edge $V(p,q,r)$ intersects at most one surface patch of $\mathcal{S}$. If non empty, the intersection $V(p,q,r) \cap S_k$ is a single point.*
- *A facet $V(p,q)$ of $\mathcal{V}(\mathcal{P})$ may intersect the surface patch S_k only if p and q belong to S_k. If non empty, the intersection $V(p,q) \cap S_k$ is a single curve segment, i.e a 1-dimensional topological ball .*
- *For any cell $V(p) \in \mathcal{P}$, the intersection $V(p) \cap S_k$ is empty if $p \notin S_k$ and a topological disk if $p \in S_k$.*

Proof. Proof omitted in this abstract.

Lemma 10. *The set $\mathcal{R} = \{V(\mathcal{T}) \cap F : \mathcal{T} \subset \mathcal{P}, F \in \mathcal{F}\}$ forms a CW complex and $\mathcal{P}$ has the extended closed ball property for $\bigcup \mathcal{F}$.*

Proof. Lemma 8 and 9 show that each element in $\mathcal{R}$ is a topological ball. Obviously, the boundary of each face in $\mathcal{R}$ is a union of faces in $\mathcal{R}$ and the intersection of two faces in $\mathcal{R}$ is either empty or a face in $\mathcal{R}$. Therefore $\mathcal{R}$ is a CW complex. For any subset $\mathcal{T} \in \mathcal{P}$, $\mathcal{R}_{\mathcal{T}} = \{V(\mathcal{T}) \cap F : F \in \mathcal{F}\}$ and $\mathcal{R}^0_{\mathcal{T}}$ is the subset of $\mathcal{R}_{\mathcal{T}}$ obtained with the faces $F \in \mathcal{F}$ whose interior intersects the interior of $V(\mathcal{T})$. Therefore condition 1 in definition 3 is satisfied and we next show that conditions 2, 3 and 4 are also satisfied.

Let $V(p,q,r)$ be an edge of $\mathcal{V}(\mathcal{P})$. For $\mathcal{T} = \{p,q,r\}$, $\mathcal{R}_{\mathcal{T}}$ is empty except if p, q, r belong to the same facet $S_k \in \mathcal{S}$. In this last case S_k is the unique face of $\mathcal{F}$ intersected by $V(p,q,r)$ so that $\mathcal{R}^0_\mathcal{T}$ reduces to $\{F_\mathcal{T} = V(p,q,r) \cap S_k\}$, thus conditions 2 and 4 are trivial and condition 3 is satisfied because $F_\mathcal{T} \cap \operatorname{bd} V(p,q,r)$ is the empty set which is a 0-dimensional sphere.

Let $V(p,q)$ be a facet of $\mathcal{V}(\mathcal{P})$ and assume that $V(p,q) \cap \mathcal{F}$ is not empty. It results from lemmas 8 and 9 that, either $V(p,q)$ intersects a single surface patch S_k of $\mathcal{S}$ and no curve segment in $\mathcal{L}$ or $V(p,q)$ intersects a single segment L_j of $\mathcal{F}$ and all the patches of $\mathcal{S}$ incident to L_j.

In the first case $\mathcal{R}^0_\mathcal{T}$ reduces to $\{F_\mathcal{T} = V(p,q) \cap S_k\}$, and conditions 2 and 4 are trivial. Condition 3 is satisfied because $V(p,q)$ does not intersect the boundary of S_k and therefore, $\operatorname{bd} V(p,q) \cap F_\mathcal{T} = \operatorname{bd} V(p,q) \cap S_k = \operatorname{bd}(V(p,q) \cap S_k)$, which, owing to lemma 9, is 1-dimensional sphere.

In the second case, $\mathcal{R}^0_\mathcal{T} = \{V(p,q) \cap L_i\} \cup \{V(p,q) \cap S_k : S_k \in \mathcal{S}, L_i \subset S_k\}$, and condition 2 holds with $F_\mathcal{T} = V(p,q) \cap L_j$. Furthermore, $\operatorname{bd} V(p,q) \cap F_\mathcal{T}$ is empty so that conditions 3 holds. For any $F = V(p,q) \cap S_k \in \mathcal{R}^0_\mathcal{T}$, $\operatorname{bd} V(p,q) \cap F = \operatorname{bd}(V(p,q) \cap S_k) - V(p,q) \cap L_j$ which is a single point (from lemmas 8 and 9), so that condition 4 holds.

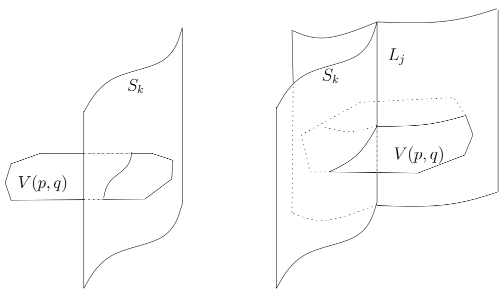

Fig. 1. Intersection with a Voronoi facet $V(p,q)$. Left $V(p,q)$ intersects no curved segment L_j. Right $V(p,q)$ intersects a single curved segment L_j.

Let $V(p)$ be a cell of $\mathcal{V}(\mathcal{P})$. If $p \notin \mathcal{F}$, $V(p)$ intersects no face of $\mathcal{F}$ and $\mathcal{R}_\mathcal{T}$ is empty.

If p belongs to some facet S_k of $\mathcal{S}$ but to no edge of $\mathcal{L}$, S_k is the only face in $\mathcal{F}$ intersected by $V(p)$. Thus, $\mathcal{R}^0_\mathcal{T}$ reduces to $\{F_\mathcal{T} = V(p) \cap S_k\}$ and conditions 2 and 4 are trivial. Condition 3 is satisfied because $V(p)$ does not intersect $\operatorname{bd} S_k$ and therefore $\operatorname{bd} V(p) \cap F_\mathcal{T} = \operatorname{bd} V(p) \cap S_k = \operatorname{bd}((V(p) \cap S_k)$, which, owing to lemma 9 and is a 1-dimensional topological sphere.

If p belongs to some edge L_i but is not a vertex of $\mathcal{F}$, $V(p)$ intersects L_i and all facets S_k containing L_i. Therefore, $\mathcal{R}^0_\mathcal{T} = \{V(p) \cap L_i\} \cap \{V(p) \cap S_k : S_k \in \mathcal{S}, L_i \subset S_k\}$ and condition 2 holds with $F_\mathcal{T} = V(p) \cap L_j$. The intersection $\operatorname{bd} V(p) \cap F_\mathcal{T}$ is $\operatorname{bd} V(p) \cap L_i$, which is two points, i. e. a 0-sphere, from lemma 8. For any $F = V(p) \cap S_k$ in $\mathcal{R}^0_\mathcal{T}$, $\operatorname{bd} V(p) \cap F = \operatorname{bd}(V(p) \cap S_k) - V(p) \cap L_i$ which is from lemmas 9 and 8 a 1-dimensional topological sphere minus a 1-dimensional topological ball, i. e. a 1-dimensional topological ball.

The case where p is a vertex of $\mathcal{F}$ is omitted in this abstract.

□

As a conclusion, for small enough values of the constant α_1 and α_2 used by the algorithm, the final sample set $\mathcal{P}$ has the extended closed ball property for $\bigcup \mathcal{F}$. Adding the reasonable assertion that $\mathcal{P}$ has the extended generic intersection property, we conclude by Theorem 2 that $\bigcup \mathcal{F}$ and $\mathcal{D}_{\bigcup \mathcal{F}}(\mathcal{P})$ are homeomorphic. Moreover, because the proof of Theorem 2 construct the isomorphism step by step between each face $V(\mathcal{T}) \cap F \in \mathcal{R}$ and the corresponding face in $\mathcal{D}_F(\mathcal{P})$ in non decreasing order of dimension, the resulting isomorphism is such that each face F of $\mathcal{F}$ is mapped to its restricted Delaunay triangulation $\mathcal{D}_F(\mathcal{P})$.

The boundary $\operatorname{bd} O_i$ of each connected component O_i of the domain O is a union of faces $F \in \mathcal{F}$, each of which is homeomorphic to its restricted Delaunay triangulation $\mathcal{D}_F(\mathcal{P})$. Thus $\operatorname{bd} O_i$ is homeomorphic to the restricted Delaunay triangulation $\mathcal{D}_{\operatorname{bd} O_i}(\mathcal{P})$, which, from the following lemma, is just $\operatorname{bd} \mathcal{D}_{O_i}(\mathcal{P})$.

Lemma 11. *For each cell O_i of $\mathcal{O}$, $\mathcal{D}_{\operatorname{bd} O_i}(\mathcal{P}) = \operatorname{bd} \mathcal{D}_{O_i}(\mathcal{P})$.*

Proof. Let pqr be a facet of $\mathcal{D}_{\operatorname{bd} O_i}(\mathcal{P})$. Then the Voronoi edge $V(p,q,r)$ intersects $\operatorname{bd} O_i$ and it results from lemma 9 that $V(p,q,r)$ intersects at most one facet of $\operatorname{bd} O_i$ and in a single point. Therefore the two endpoints of $V(p,q,r)$ are Voronoi vertices one of which is inside O_i while the other is outside, which means that one of the tetrahedra incident to pqr belongs to $\mathcal{D}_{O_i}(\mathcal{P})$ while the other does not and therefore pqr belongs to $\operatorname{bd} \mathcal{D}_{O_i}(\mathcal{P})$. Reciprocally if pqr belongs to $\operatorname{bd} \mathcal{D}_{O_i}(\mathcal{P})$, the two endpoints of the Voronoi edge $V(p,q,r)$ are on a different side of $\operatorname{bd} O_i$ which means that $V(p,q,r)$ intersects $\operatorname{bd} O_i$ and therefore belongs to $\mathcal{D}_{\operatorname{bd} O_i}(\mathcal{P})$. □

Hausdorff distance

We prove here that the mesh generation algorithm provides a control on the Hausdorff distance between F and the approximating linear complex $\mathcal{D}_F(\mathcal{P})$ through the sizing field σ.

Let us first consider a curve segment L_j in $\mathcal{L}$. For each edge $e = pq$ in $\mathcal{D}_{L_j}(\mathcal{P})$, both edge e and the portion $L_j(p,q)$ of L_j joining p to q are included in the restricted Delaunay $B(c_e, r_e)$ circumscribed to e. The Hausdorff distance between $L_j(p,q)$ and e is therefore less than r_e which, from rule R2, is less $\alpha_1 \sigma(c_e)$ and the Hausdorff distance between L_j and $\mathcal{D}_{L_j}(\mathcal{P})$ is less than $2\alpha_1 \max_{x \in L_j} \sigma(x)$.

Let us then consider a surface patch S_k. Each triangle pqr in $\mathcal{D}_{S_k}(\mathcal{P})$ is included in its restricted Delaunay ball $B(c,r)$ with radius $r \leq \alpha_2 \sigma(c)$ and therefore each point of pqr is at distance less than $\alpha_2 \sigma(c)$ from S_k. From rule R4.1, and the small cylinder lemma (Lemma 7), we know that each point x in S_k is at distance $O(\alpha_2)\sigma(p)$ from its closest sample point p. The Hausdorff distance between S_k and $\mathcal{D}_{S_k}(\mathcal{P})$ is less than $O(\alpha_2) \max_{x \in S_k} \sigma(x)$.

5 Termination

This section proves that the refinement algorithm in section 3 terminates, provided that the constants α_1 and α_2, β_2 and β_3 involved in refinement rules are judiciously chosen. The proof of termination is, based on a volume argument. This requires to prove a lower bound for the distance between any two vertices inserted in the mesh.

For each vertex p in the mesh, we denote by $r(p)$ the insertion radius of p, that is the length of the shortest edge incident to vertex p right after the insertion of p, and $\delta(p)$ the distance from p to $\mathcal{F} \setminus \mathcal{F}_p$ where $\mathcal{F}_p$ is the set of features in $\mathcal{F}$ that contain the point p.

Recall that the algorithm depends on four parameters α_1, α_2, β_2 and β_3, and on a sizing field σ. Constants α_1 and α_2 are assumed to be small enough and such that $0 \leq \alpha_1 \leq \alpha_2 \leq 1$. The sizing field σ is assumed to be a Lipschitz function smaller than $\mathrm{lfs}(x)$ on $\mathcal{F}$. Let μ_0 be the maximum of σ over $\mathcal{F}$, and σ_0 the minimum of σ over $\mathcal{O}$.

Lemma 12. *For some suitable values of* α_1,α_2, β_2 *and* β_3, *there are constants* η_2 *and* η_3 *such that* $\alpha_1 \leq \eta_3 \leq \eta_2 \leq 1$ *and such that the following invariants are satisfied during the execution of the algorithm.*

$$\forall p \in \mathcal{P}, \quad r(p) \geq \alpha_1 \sigma_0 \tag{2}$$

$$\delta(p) \geq \begin{cases} \alpha_1\sigma_0 & \text{if } p \in \mathcal{L} \\ \frac{\alpha_1\sigma_0}{\eta_2} & \text{if } p \in \mathcal{S} \\ \frac{\alpha_1\sigma_0}{\eta_3} & \text{if } p \in \mathcal{P} \setminus \mathcal{S} \end{cases} \tag{3}$$

Proof. (Sketch) The proof is an induction. Invariants (2) and (3) are satisfied by the set $\mathcal{Q}$ and by the set $\mathcal{P}_0$ of initial vertices. We prove that invariants(2) and (3) are still satisfied after the application of any of the refinement rules R1-R5 if the following values are set:

$$\begin{array}{llll} & \alpha_1 = \frac{1}{(\sqrt{2}+2)\nu_0,(\nu_0+1)} & \alpha_2 = \frac{1}{\nu_0+1}, & \\ & \beta_2 = (\sqrt{2}+2)\nu_0, & \beta_3 = (\sqrt{2}+2)\nu_0\,(\nu_0+1)\,, & \\ & \frac{1}{\eta_2} = (\sqrt{2}+1)\nu_0, & \frac{1}{\eta_3} = (\sqrt{2}+2)\nu_0, & \\ \text{where} & \nu_0 = \frac{2\mu_0}{\sigma_0}. & & \end{array} \tag{4}$$

For lack of space and as an example, we only give here the proof for Rule R3 and omit the other cases.

Let us therefore assume that a new vertex c_f is added to $\mathcal{P}$ by application of Rule R3. The vertex c_f belongs to a surface patch S_k and is the center of a Delaunay ball $B(c_f, r_f)$, circumscribing a facet f of $\mathcal{D}_{|S_k}$. At least one of the vertices of f vertices, say p, does not belong to S_k. The insertion radius of c_f is r_f and, $r_f = \mathrm{d}(c_f, p) \geq \mathrm{d}(p, S_k)$. By induction hypothesis, $\mathrm{d}(p, S_k) \geq \alpha_1\sigma_0$, and invariant (2) is satisfied.

To check invariant (3), we prove hereafter than $\mathrm{d}(c_f, \mathrm{bd}\, S_k) \geq \frac{\alpha_1\sigma_0}{\eta_2}$, and use the following claim to prove that the same bound holds for $\delta(c_f)$.

Claim. For any point x in a face $F \in \mathcal{F}$, $\delta(x) \geq \min(\mathrm{d}(x, \mathrm{bd}\, F), \mathrm{lfs}(x))$.

Proof. Omitted in this abstract.

Let y be the point of $\mathrm{bd}\, S_k$ closest to c_f, and let q be the sample point in $\mathcal{P} \cap \mathrm{bd}\, S_k$ closest to y. Then $\mathrm{d}(c_f, \mathrm{bd}\, S_k) = \mathrm{d}(c_f, y) \geq \mathrm{d}(c_f, q) - \mathrm{d}(y, q)$. The distance $\mathrm{d}(c_f, q)$ is at least r_f. The distance $\mathrm{d}(y, q)$ is at most $2\alpha_1\mu_0$ because, when Rule R3 is applied, the curve segments of $\mathcal{L}$ are covered by restricted Delaunay balls whose radii are at most $\alpha_1\mu_0$. Therefore

$$\mathrm{d}(c_f, \mathrm{bd}\, S_k) \geq r_f - 2\alpha_1\mu_0. \tag{5}$$

- If vertex p does not belongs to $\mathcal{F}$, $r_f = \mathrm{d}(c_f, p) \geq \frac{\alpha_1\sigma_0}{\eta_3}$ by induction hypothesis, and $\mathrm{d}(c_f, \mathrm{bd}\, S_k) \geq \frac{\alpha_1\sigma_0}{\eta_3} - 2\alpha_1\mu_0$ which is at least $\frac{\alpha_1\sigma_0}{\eta_2}$ with the choices in (4).
- If the vertex p lies on $\mathcal{F}$ but is not interrelated to c_f, $r_f = \mathrm{d}(c_f, p) \geq \mathrm{lfs}(c_f) \geq \sigma_0$, and $\mathrm{d}(c_f, \mathrm{bd}\, S_k) \geq \sigma_0 - 2\alpha_1\mu_0$ which is more than $\frac{\alpha_1\sigma_0}{\eta_2}$ with the choices in (4).
- If p lies on $F_i \in \mathcal{F}$ and is interrelated to c_f, we have by angular hypothesis, $r_f^2 = \mathrm{d}(c_f, p)^2 \geq \mathrm{d}(c_f, S_k \cap F_i)^2 + \mathrm{d}(p, S_k \cap F_i)^2$. Thus $\mathrm{d}(c_f, S_k \cap F_i) \leq r_f$, which proves that sample point p cannot lie in a curve segment of $\mathcal{L}$. Indeed if F_i was some curve segment in $\mathcal{L}$, $S_k \cap F_i$ would be a vertex in $\mathcal{Q}$ included in $B(c_f, r_f)$ which contradicts the fact that $B(c_f, r_f)$ is a Delaunay ball. Therefore p belongs to the interior of some surface patch $S_i \in \mathcal{S}$ and by induction hypothesis $\delta(p) \geq \frac{\alpha_1\sigma_0}{\eta_2}$ which implies that:

$$r_f^2 \geq \mathrm{d}(c_f, \mathrm{bd}\, S_k)^2 + \left(\frac{\alpha_1\sigma_0}{\eta_2}\right)^2. \tag{6}$$

  Equations (5) and (6) and the value chosen for $\eta 2$ in (4) implies that $\mathrm{d}(c_f, \mathrm{bd}\, S_k) \geq \frac{\alpha_1\sigma_0}{\eta_2}$.

□

Lemma 12 provides a constant lower bound, $\alpha_1\sigma_0$, for the insertion radius of any vertex in the mesh and therefore for the length of any edge in the final mesh. A standard volume argument then ensures that the Delaunay refinement algorithm of section 3 terminates.

6 Implementation and results

The algorithm has been implemented in C++, using the library CGAL [CGAL], which provided us with an efficient and flexible implementation of the three-dimensional Delaunay triangulation. Once the Delaunay refinement process described above is over, a sliver exudation [CDE+00] step is performed. This step does not move or add any vertex but it modifies the mesh by switching the triangulation into a weighted Delaunay triangulation with carefully chosen weights. As in [ORY05] the weight of each vertex in the mesh is chosen in turn so as to maximize the smallest dihedral angle of any tetrahedron incident to that vertex while preserving in the mesh any facet that belongs to the restricted triangulation of an input surface patch.

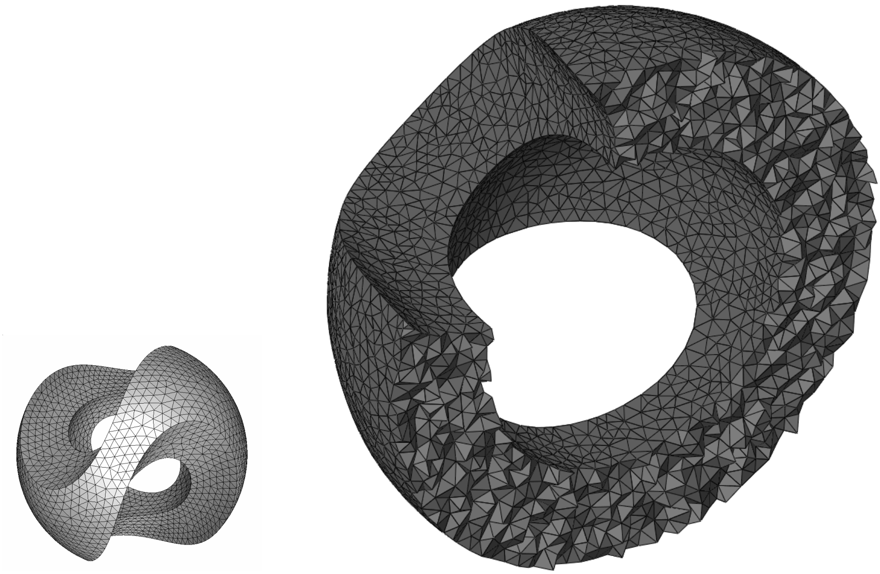

Fig. 2. Sculpt model. On the left: the input surface mesh. On the right: the output mesh (blue: facets of the surface mesh, red: tetrahedra of the volume mesh that intersect a given plan). The mesh counts 22923 tetrahedra.

Our mesh generation algorithm interacts with the input curve segments and surface patches through an oracle that is able to detect and compute intersections between planar triangles and curve segments and between straight segments and surface patches. Currently, we have only one implementation of such an oracle which handles input curve segments and surface patches described as respectively as polylines and triangular meshes. Thus, with respect to input features our algorithm currently act as a remesher but this is not a limitation of the method.

The figure 2 shows a mesh generated by our algorithm. The input surface of the figure 2 is made of six curved edges, and four surface patches The input surface was given by a surface mesh. We have considered that an edge of the mesh is a sub-segment of a curved segment of the surface when the normal deviation at the edge is greater than 60 degree. The curved segments and surfaces patches of the input surface are nicely approximated, as well as the normals (the result surface of figure 2 has been drawn without any OpenGL smoothing). Each element of the result mesh has a Delaunay ball smaller than

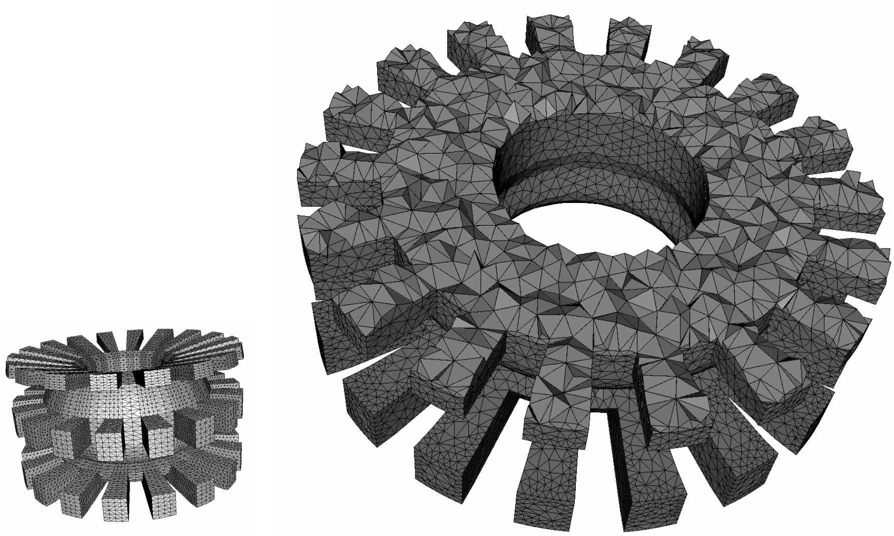

Fig. 3. ITER model. A mesh with 72578 tetrahedra.

a given sizing field. In the figure 2, the sizing field has been chosen uniform. After sliver exudation, the worst tetrahedra in the mesh has a dihedral angle of 1.6 degree.

Another example is shown on Figure 3 where a mechanical piece which is part of the International Thermonuclear Experimental Reactor (ITER) has been meshed.

7 Conclusion and future work

The algorithm provided in this paper is able to mesh volumes bounded by piecewise smooth surfaces. The output mesh has guaranteed quality and its granularity adapts to a user defined sizing field. The boundary surfaces and their sharp 1-dimensional features are accurately and homeomorphically represented in the mesh. The main drawback of the algorithm is the restriction imposed on dihedral angles made by tangent planes on singular points. Small angles are known to trigger an ever looping of the Delaunay refinement algorithm. The main idea to handle this problem is to define a protected zone around sharp features where the Delaunay refinement is restricted to prevent looping. This strategy assumes that the mesh already includes restricted Delaunay submeshes homeomorphic to the input surface patches and curved segments. This could be achieved using a strategy analog to the strategy proposed to conform Delaunay triangulation [MMG00, CCY04]. Another promising way to protect sharp features which is proposed by [CDR07] is to use weighted Delaunay triangulation with weighted points on sharp features.

References

[AB99] N. Amenta and M. Bern. Surface reconstruction by Voronoi filtering. *Discrete Comput. Geom.*, 22(4):481–504, 1999.

[ACDL02] N. Amenta, S. Choi, TK Dey, and N. Leekha. A simple algorithm for homeomorphic surface reconstruction. Internat. *J. Comput. Geom. & Applications*, 12:125–141, 2002.

[AUGA05] P. Alliez, G. Ucelli, C. Gotsman, and M. Attene. Recent advances in remeshing of surfaces. *Part of the state-of-the-art report of the AIM@SHAPE EU network*, 2005.

[BO05] J.D. Boissonnat and S. Oudot. Provably good sampling and meshing of surfaces. *Graphical Models*, 67(5):405–451, 2005.

[BO06] J.D. Boissonnat and S. Oudot. Provably good sampling and meshing of Lipschitz surfaces. *Proceedings of the twenty-second annual symposium on Computational geometry*, pages 337–346, 2006.

[BOG02] 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.

[Boi06] Jean-Daniel Boissonnat. *Voronoi Diagrams, Triangulations and Surfaces*, chapter 5. Inria, 2006.

[CCY04] D. Cohen-Steiner, É. Colin de Verdière, and M. Yvinec. Conforming Delaunay triangulations in 3d. *Computational Geometry: Theory and Applications*, pages 217–233, 2004.

[CD03] S.-W. Cheng and T. K. Dey. Quality meshing with weighted Delaunay refinement. *SIAM J. Comput.*, 33(1):69–93, 2003.

[CDE+00] S.-W. Cheng, T. K. Dey, H. Edelsbrunner, M. A. Facello, and S.-H. Teng. Sliver exudation. *J. ACM*, 47(5):883–904, 2000.

[CDR07] S.W. Cheng, T.K. Dey, and E.A. Ramos. Delaunay Refinement for Piecewise Smooth Complexes. *Proc. 18th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 1096–1105, 2007.

[CDRR04] 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.

[CDRR05] S.-W. Cheng, T. K. Dey, E. A. Ramos, and T. Ray. Weighted Delaunay refinement for polyhedra with small angles. In *Proceedings 14th International Meshing Roundtable, IMR2005*, 2005.

[CGAL] Cgal, Computational Geometry Algorithms Library. http://www.cgal.org.

[Che89] L. P. Chew. Guaranteed-quality triangular meshes. Technical Report TR-89-983, Dept. Comput. Sci., Cornell Univ., Ithaca, NY, April 1989.

[Che93] L.P. Chew. Guaranteed-quality mesh generation for curved surfaces. *Proceedings of the ninth annual symposium on Computational geometry*, pages 274–280, 1993.

[CP03] 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.

[Dey06] T.K. Dey. *Curve and Surface Reconstruction: Algorithms with Mathematical Analysis.* Cambridge University Press, 2006.

[ES97] H. Edelsbrunner and N.R. Shah. Triangulating Topological Spaces. *International Journal of Computational Geometry and Applications*, 7(4):365–378, 1997.

[FBG96] 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.

[FG00] P.J. Frey and P.L. George. *Mesh Generation: Application to Finite Elements.* Kogan Page, 2000.

[GHS90] 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.

[GHS91] 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.

[LC87] W.E. Lorensen and H.E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169, 1987.

[LS03] Francois Labelle and Jonathan Richard Shewchuk. Anisotropic voronoi diagrams and guaranteed-quality anisotropic mesh generation. In *SCG '03: Proceedings of the nineteenth annual symposium on Computational geometry*, pages 191–200, New York, NY, USA, 2003. ACM Press.

[LT01] X.-Y. Li and S.-H. Teng. Generating well-shaped Delaunay meshes 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.

[MMG00] M. Murphy, D.M. Mount, and C.W. Gable. A point-placement strategy for conforming Delaunay tetrahedralization. *Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms*, pages 67–74, 2000.

[MTT99] G.L. Miller, D. Talmor, and S.H. Teng. Data Generation for Geometric Algorithms on Non-Uniform Distributions. *International Journal of Computational Geometry and Applications*, 9(6):577, 1999.

[ORY05] Steve Oudot, Laurent Rineau, and Mariette Yvinec. Meshing volumes bounded by smooth surfaces. In *Proc. 14th International Meshing Roundtable*, pages 203–219, 2005.

[Rup95] J. Ruppert. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *J. Algorithms*, 18:548–585, 1995.

[She98] J. R. Shewchuk. Tetrahedral mesh generation by Delaunay refinement. In *Proc. 14th Annu. ACM Sympos. Comput. Geom.*, pages 86–95, 1998.

[She02] J. R. Shewchuk. Delaunay refinement algorithms for triangular mesh generation. *Computational Geometry: Theory and Applications*, 22:21–74, 2002.