ALIASING-FREE SIMPLIFICATION OF SURFACE MESHES

Jean-Luc Peyrot, Frédéric Payan and Marc Antonini, IEEE member

Laboratory I3S - University Nice - Sophia Antipolis and CNRS (France) - UMR 7271
peyrot@i3s.unice.fr, fpayan@i3s.unice.fr, am@i3s.unice.fr

ABSTRACT

We propose in this paper a robust simplification technique, which preserves geometric features such as sharp edges or corners from original surfaces. To achieve this goal, our simplification process relies on a detection tool that enables to preserve the sharp features during the three subsequent steps: a Poisson disk sampling that intelligently reduces the number of vertices of the initial mesh; the meshing of the samples that aligns the edges along the feature lines; and a constrained relaxation step that improves the shape of the triangles of our final simplified mesh. Experimental results show that our method always produces valid meshes without aliasing artifacts, and without giving up the shape fidelity and quality of the mesh elements.

***Index Terms*—** Mesh simplification, meshing, Poisson disk sampling, feature-preserving, GPU

1. INTRODUCTION

Nowadays, 3D acquisition systems are widespread used to generate numeric representations of surfaces. In order to capture small details, current systems generate dense point clouds. Consequently, the meshes generated from such point clouds are dense, and often oversampled. Moreover, they may also suffer from bad-shaped triangles (*i.e.* poor aspect-ratio), which is not desirable for many applications.

To overcome this problem, many simplification techniques have been developed for the last decades. They aim at reducing the number of vertices, while possibly optimizing their positions, and ensuring the fidelity with respect to the original shapes. It can be also desirable to generate high-quality simplified meshes, in other words, to improve the aspect-ratio of the triangles.

The literature relative to mesh simplification is large [1,2]. The two main categories are the methods based on incremental simplification, such as the well-known *Qslim* algorithm of Garland and Heckbert [3], and the methods based on partitioning, such as the *VSA* algorithm of Cohen-Steiner *et al.* [4].

Furthermore, numerous sampling methods for surfaces emerged in recent years. Among them, the methods that generate Poisson disk distributions are particularly relevant for many applications such as rendering [5] or texturing [6]. Fu *et al.* [7] showed that a Poisson disk distribution is also a relevant way to create high-quality meshes. Indeed, their overall idea is to position and triangulate a set of samples on a given surface according to their Poisson disk sampling method, and then to relax their positions to improve the quality of the output triangles. Recently, we developed an efficient feature-preserving Poisson disk sampling method for surface meshes [8]. Our method has the advantage to handle any kind of topology, to preserve sharp features, and to be relatively easy to implement. Inspired by [7], we propose in this paper our own simplification method for triangular meshes. Our main objective is to propose a robust simplification technique that handles complex surfaces having saliences, genus, and which generates valid meshes, without outliers or triangle flips.

Starting from a set of samples generated with [8], we explain here how these samples can be meshed and better positioned with respect to the original shape, to finally obtain a simplified mesh without any geometric aliasing artifacts (*i.e.* notches along features for instance), and whose triangles have a good aspect-ratio. The main contributions of this paper are a weighted geodesic Voronoi diagram and a constrained relaxation which are driven by a vertex classification to preserve sharp features.

The paper is organized as follows: Section 2 gives an overview of our simplification method. Then, Section 3 introduces the notion of Geodesic Voronoi Diagram (GVD), the key component of our method, presents typical meshing techniques based on GVD, and our contribution. Next, Section 4 exposes our feature-constrained relaxation method, and Section 5 provides experimental results and comparisons with prior simplification techniques. Finally, Section 6 concludes this paper.

2. SIMPLIFICATION OVERVIEW

The main steps of our simplification method are illustrated in Figure 2, and an example of data generated all along the process is presented in Figure 1.

Starting from a dense triangular mesh (a), a feature detection tool is applied to classify its vertices into two categories

THIS WORK IS SUPPORTED BY A GRANT FROM *RÉGION PROVENCE ALPES CÔTE D'AZUR.*

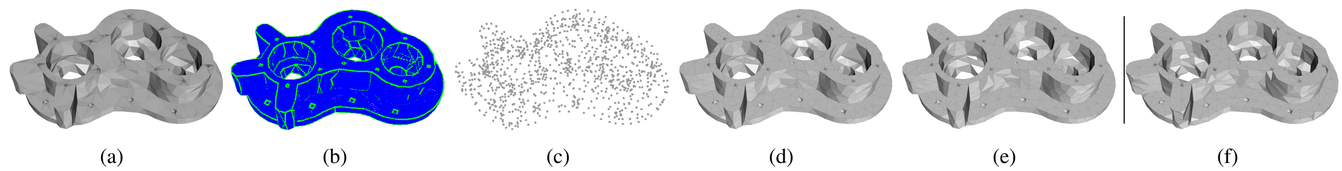

Fig. 1: Simplification overview: (a) dense input mesh (around 240k vertices); (b) classification result (green: *sharp edge*; blue: *smooth region*); (c) Poisson disk samples (around 1k samples); (d) output of the meshing with the proposed weighted GVD; (e) output of the relaxation: the simplified mesh (without geometric aliasing artifacts). (f) For comparison, output of the meshing with the usual GVD: the geometric aliasing damages the sharp features.

(b): each vertex belongs to a *smooth region* or to a *sharp edge*. According to this classification, a set of Poisson disk samples is generated (c). This latter is then triangulated, according to our meshing based on a weighted Geodesic Voronoi Diagram (GVD) to produce a first mesh without geometric aliasing (d). Finally a relaxation is applied on this mesh to modify the position of the vertices, which permits to reduce the global approximation error and to improve the aspect-ratio of the triangles of the simplified mesh (e). Constrained by the feature detection, the two last steps do not damage the features. For comparison, (f) presents the output of our simplification technique if the usual GVD is used: the geometric aliasing damages the sharp features.

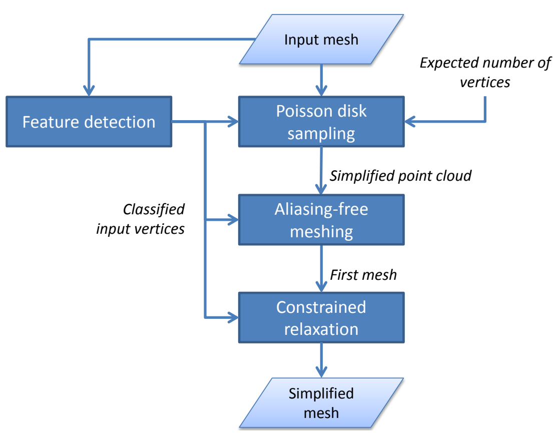

Fig. 2: Proposed simplification algorithm.

3. PROPOSED MESHING TECHNIQUE

In this section we present our first contribution, a meshing (or reconstruction) technique based on a Geodesic Voronoi Diagram (GVD) [9–11].

3.1. Geodesic Voronoi Diagram (GVD)

We consider a surface S embedded in $\mathbb{R}^3$, and a set $X = \{x_i\}$ of p seeds initialized on the surface S. The GVD of X on S is the union of the Voronoi cells $\{c_i\}$ associated to the seeds $\{x_i\}$. Each cell c_i is defined on S by

$$c_i = \{x \in S / \ \forall j \neq i, d(x, x_i) < d(x, x_j)\}, \qquad (1)$$

where $d(.,.)$ is a geodesic distance. In our case, the surface S being defined by a mesh M, the cell c_i corresponds to the set of vertices of M which are closer to the seed x_i than all other seeds (in terms of geodesic distance).

3.2. Meshing based on GVD

Such a meshing technique consists in connecting the seeds $\{x_i\}$ to generate a mesh M_{out}. Usually, this process is done by linking the seeds whose Voronoi cells share a common border [12].

This technique is simple and efficient for smooth surfaces, but may generate artifacts on saliences like sharp edges: see Figure 1(f). Those artifacts, often called the *geometric aliasing artifacts*, appear when the edges of the output mesh are not aligned along the feature lines. When a GVD is used during meshing, these artifacts occur when two consecutive seeds on a feature line have no common border. This is due to the fact that these seeds are too far from each other, and consequently the associated cells are disjoint. Figure 3(b) illustrates this problem, where the red points represent the seeds that do not lie on the feature edge, while blue ones do. The GVD is calculated using equation (1). We observe that the orange and the red cells associated to the blue seeds are disjoint. Consequently, a notch appears since the generated edge connects the red seeds.

3.3. Our contribution

To overcome the aforementioned geometric aliasing, we propose to apply a weighting on the GVD. The objective is to stretch the Voronoi cells along features. For this purpose, we first reuse the classification results: a vertex v of M is part of the class *sharp edge* if it lies on a feature edge, otherwise it belongs to the class *smooth region*. Then, to create the GVD, we replace in equation (1) the geodesic distance function $d(.,.)$

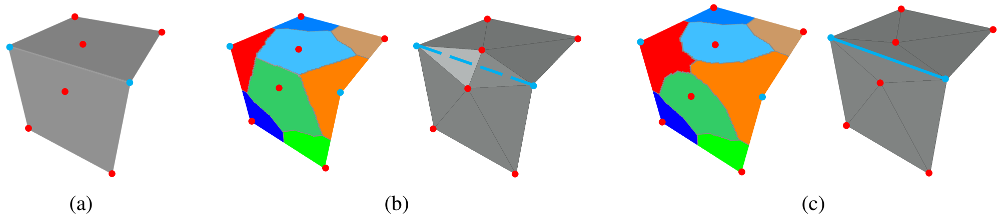

Fig. 3: Geometric aliasing and resulting solution of our feature-aware weighted meshing: (a) original surface; (b) and (c) GVD and the resulting triangulation obtained without and with the proposed weighting, respectively.

by the following weighted geodesic distance $d_{pond}(.,.)$:

$$d_{pond}(v1, v2) \; = \; \alpha.d(v1, v2). \tag{2}$$

$v1$ and $v2$ are two neighbor vertices of M. α is used to warp the Voronoi cells: α is smaller than 1 if $v1$ and $v2$ belong to the class *sharp edge*, otherwise, α is equal to 1. The role of the weighted distance $d_{pond}(.,.)$ is to speed the growing process up along saliences and thus to ensure the creation of a common border between seeds on the feature lines. Figure 3(c) illustrates this fact: the red and the orange cells now share a border, and consequently, the aliasing artifact does not occur. Empirically, α is set to 0.3, which is efficient for all our experimentations, whatever the expected number of vertices.

3.4. Implementation

Before detailing the implementation of the proposed meshing technique, we recall that the two first steps of our simplification are the feature detection and the sampling (see Figure 2). The latter generates the samples (see Figure 1(c)) that will be considered as the seeds $\{x_i\}$ for the meshing step. For these two first steps, we choose to use the dart throwing technique proposed in [8]. This technique generates Poisson disk sampling on surfaces of arbitrary topology, while preserving the sharp features. It is based on the tensor voting theory [13], to detect the feature lines, and Dijkstra's algorithm [14] to measure the geodesic distances between samples. Please read [8] for more details.

The pseudo-code of our meshing is given by Algorithm 1, see below.

Algorithm 1 Implementation details of our meshing method

Consider the samples generated with [8] as seeds $\{x_i\}$ and the classified input vertices;
Step 1: Compute the GVD of X on M *via* a region growing process using the weighted geodesic distance (2);
Step 2: Triangulate the samples $\{x_i\}$ according to the resulting GVD.

To accelerate the generation of the GVD, we use a GPU parallelized version of the multi-sources Dijkstra's algorithm [15]. The creation of a Voronoi diagram indeed is well suited to parallelization, by using one thread per seed. However, special attention must be paid along the borders. Indeed, two threads might add the same vertex to different cells at the same time. Consequently, a mutual exclusion technique has been implemented to retrieve reliable cells over the surface of M, that allows only one thread at a time to access a vertex. The principle is depicted in Figure 4: (a) two cells are growing simultaneously on the surface of M. (b) and (c) after two iterative propagations, their borders are in contact and the surrounded red area is critical because the update of the vertex distances to their respective nearest seeds, must be done iteratively, and not in parallel (*i.e.* one thread after the other).

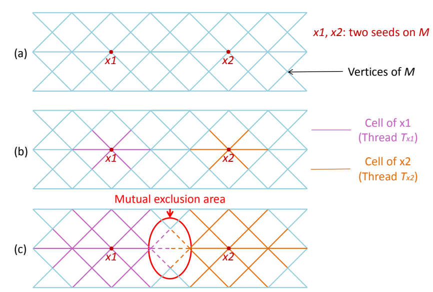

Fig. 4: GPU parallelized GVD. The red circle highlights the area where false values might appear if the two threads T_{x1} and T_{x2} modify at the same time the distance from the same vertex to their respective seeds.

4. PROPOSED CONSTRAINED RELAXATION

The output of the reconstruction step is a first mesh without aliasing artifacts. However, the quality of the triangles and its fidelity with respect to the initial shape may be improved using a relaxation. The proposed relaxation is based on our algorithm proposed in [16] for generating Centroidal Voronoi tessellation. The principle is, for a given diagram, to compute the centroid of each cell, and to move the seed to this specific position. A new tessellation is then computed, and these two stages are iterated until convergence. Such an approach cre-

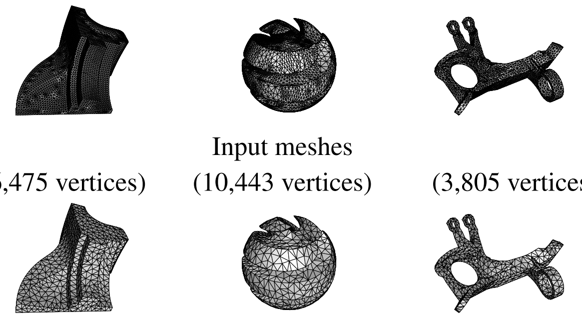

Fig. 5: Three meshes and their simplified versions, obtained with our method. From left to right: FANDISK, SHARP SPHERE and AXLE.

ates dual triangulation of high quality: see [16] for more details. The main differences between the relaxation proposed here and [16] are: (i) the weighted geodesic distance (equation 2) is used as metric to compute the Voronoi cells (instead of the euclidean distance, absolutely not suitable for dealing with shapes of arbitrary topology, sharp features, etc.); (ii) at the end of each relaxation stage, each "centroidal" seed is moved to the closest vertex of the input mesh M that belongs to the same class as this seed before relaxation (*smooth region* or *sharp edge*). The interest of this second modification is to ensure that the vertices remain positioned on the feature lines despite the relaxation. Hence, the sharp features of the input shape are also preserved during relaxation.

5. RESULTS

Figure 5 depicts three original manifold meshes (on top), and their simplified versions created with our method (on bottom).

We observe that no geometric aliasing occurs during our simplification and that each mesh well approximates its original shape. To verify the fidelity of our simplified meshes to the original shapes (low approximation error) and the quality of the output triangles, we compare our results in terms of geometric error (RMSE) [17], and minimum angle of triangles, with those produced by the prior simplification techniques *Qslim* [3] and *ACVD* [18] (see Table 1). As expected, *Qslim* is more efficient than our method on the smoother surfaces (SHARP SPHERE and FERTILITY). On the other hand, our method gives similar or lower approximation errors than *Qslim* and *ACVD* for non-smooth meshes such as AXLE or FANDISK. Also, we observe that, as expected, our method always outperforms *Qslim* in terms of quality of triangles, and most of times outperforms *ACVD*. Moreover, in the contrary of the two prior methods, our simplification always provides valid meshes (*i.e.* manifold meshes) without aliasing artifacts, and without giving up the shape fidelity and quality of the mesh elements. Indeed, see the three simplified meshes presented in Figure 6, generated by *Qslim* [3] and *ACVD* [18]: they contain triangle flips (FANDISK and AXLE) or outliers (CASTING). Finally, our results are very satisfactory since our objective was to propose a robust simplification technique dedicated for such complex meshes.

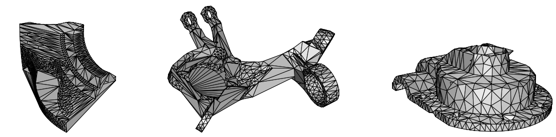

Fig. 6: Three meshes produced by *Qslim* [3] and *ACVD* [18], which present triangle flips or outliers. From left to right: FANDISK, AXLE and CASTING.

Models	Methods	Manifold	RMSE	Min. angle
FERTILITY	*Qslim* [3]	Yes	0.00068	31.3°
	ACVD [18]	Yes	0.00278	47.2°
	Our method	Yes	0.00242	39.4°
FANDISK	*Qslim* [3]	Yes	0.00390	26.6°
	ACVD [18]	Yes	0.00023	35.0°
	Our method	Yes	0.00040	39.2°
SHARP SPHERE	*Qslim* [3]	Yes	0.00068	33.8°
	ACVD [18]	No	0.00119	37.6°
	Our method	Yes	0.00238	39.7°
AXLE	*Qslim* [3]	No	0.00924	30.6°
	ACVD [18]	No	0.00115	35.9°
	Our method	Yes	0.00090	36.1°

Table 1: RMSE and minimum angle of triangles in degrees, for our method and two prior simplification algorithms (*Qslim* [3] and *ACVD* [18]). It is also indicated either the produced meshes are manifold or not.

6. CONCLUSION

We presented a new simplification technique for surface meshes, that has the advantage to handle surfaces of arbitrary topology, and to preserve the sharp features. The main contributions are an efficient meshing and a relaxation technique that take into account the feature lines in order to avoid geometric aliasing. Experimentations show that our resulting simplified meshes are also of *high quality*, since they approximate well the input shapes in terms of geometric error and triangle aspect-ratio. Furthermore, they are valid meshes and do not present outliers or triangle flips.

7. REFERENCES

[1] Paolo Cignoni, Claudio Montani, and Roberto Scopigno, "A comparison of mesh simplification algorithms," *Computers & Graphics*, vol. 22, no. 1, pp. 37 – 54, 1998.

[2] David P. Luebke, "A developer's survey of polygonal simplification algorithms," *IEEE Computer Graphics and Applications*, vol. 21, no. 3, pp. 24–35, 2001.

[3] Michael Garland and Paul S. Heckbert, "Surface simplification using quadric error metrics," *Computer Graphics*, vol. 31, pp. 209–216, 1997.

[4] David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun, "Variational shape approximation," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 905–914, 2004.

[5] Ewen Cheslack-Postava, Rui Wang, Oskar Akerlund, and Fabio Pellacini, "Fast, realistic lighting and material design using nonlinear cut approximation," in *ACM SIGGRAPH Asia 2008 Papers*, 2008, SIGGRAPH Asia '08, pp. 128:1–128:10.

[6] Ares Lagae and Philip Dutré, "A procedural object distribution function," *ACM Transactions on Graphics*, vol. 24, no. 4, pp. 1442–1461, Oct. 2005.

[7] Yan Fu and Bingfeng Zhou, "Direct sampling on surfaces for high quality remeshing," in *Proceedings of the ACM Symposium on Solid and Physical Modeling*, New York, NY, USA, 2008, p. 115–124, ACM.

[8] Jean-Luc Peyrot, Frédéric Payan, and Marc Antonini, "Feature-preserving direct blue noise sampling for surface meshes," in *Eurographics (Short Papers)*, may 2012, pp. 9–912.

[9] Ron Kimmel and James A. Sethian, "Fast voronoi diagrams and offsets on triangulated surfaces," in *Proc. of AFA Conf. on Curves and Surfaces*. 1999, pp. 193–202, University Press.

[10] G. Peyré and L. D. Cohen, "Geodesic remeshing using front propagation," *International Journal of Computer Vision*, vol. 69, no. 1, pp. 145–156, 2006.

[11] Dong-Ming Yan, Bruno Lévy, Yang Liu, Feng Sun, and Wenping Wang, "Isotropic remeshing with fast and exact computation of restricted voronoi diagram," in *Proceedings of the Symposium on Geometry Processing*. 2009, p. 1445–1454, Eurographics Association.

[12] Shi-Qing Xin, Shuang-Min Chen, Ying He, Guo-Jin Wang, Xianfeng Gu, and Hong Qin, "Isotropic mesh simplification by evolving the geodesic delaunay triangulation," in *Proceedings of the 2011 Eighth International Symposium on Voronoi Diagrams in Science and Engineering*, Washington, DC, USA, 2011, p. 39–47, IEEE Computer Society.

[13] Hyun Soo Kim, Han Kyun Choi, and Kwan H. Lee, "Feature detection of triangular meshes based on tensor voting theory," *Journal of Computer-Aided Design and Computer Graphics*, vol. 41, no. 1, pp. 47–58, Jan. 2009.

[14] Edsger. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.

[15] A. Munshi, B.-R. Gaster, T.-G. Mattson, J. Fung, and D. Ginsburg, *OpenCL Programming Guide*, Prentice Hall, 2011.

[16] A. Kammoun, F. Payan, and M. Antonini, "Adaptive semi-regular remeshing: A voronoi-based approach," in *Proceedings of IEEE international workshop on MultiMedia Signal Processing*, 2010, p. 350–355.

[17] Paolo Cignoni, Claudio Rocchini, and Roberto Scopigno, "Metro: Measuring error on simplified surfaces," *Computer Graphics Forum*, vol. 2, no. 17, pp. 167–174, 1998.

[18] S. Valette, J.-M. Chassery, and R. Prost, "Generic remeshing of 3d triangular meshes with metric-dependent discrete voronoi diagrams," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 369–381, 2008.