VoroCrust Illustrated: Theory and Challenges

Ahmed Abdelkader
University of Maryland, College Park MD, USA
akader@cs.umd.edu

Chandrajit L. Bajaj[1]
University of Texas, Austin TX, USA

Mohamed S. Ebeida
Sandia National Laboratories, Albuquerque NM, USA

Ahmed H. Mahmoud
University of California, Davis CA, USA

Scott A. Mitchell
Sandia National Laboratories, Albuquerque NM, USA

John D. Owens
University of California, Davis CA, USA

Ahmad A. Rushdi
University of California, Davis CA, USA

Abstract

Over the past decade, polyhedral meshing has been gaining popularity as a better alternative to tetrahedral meshing in certain applications. Within the class of polyhedral elements, Voronoi cells are particularly attractive thanks to their special geometric structure. What has been missing so far is a Voronoi mesher that is sufficiently robust to run automatically on complex models. In this video, we illustrate the main ideas behind the VoroCrust algorithm, highlighting both the theoretical guarantees and the practical challenges imposed by realistic inputs.

2012 ACM Subject Classification Theory of computation → Randomness, geometry and discrete structures → Computational geometry

Keywords and phrases sampling, surface reconstruction, polyhedral meshing, Voronoi

Digital Object Identifier 10.4230/LIPIcs.SoCG.2018.77

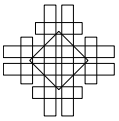
Category Multimedia Contribution

Related Version See the corresponding paper in these proceedings [2].

Funding This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research (ASCR), Applied Mathematics Program.

Acknowledgements Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

[1] Supported in part by a contract from Sandia, #1439100, and a grant from NIH, R01-GM117594



34th International Symposium on Computational Geometry (SoCG 2018).
Editors: Bettina Speckmann and Csaba D. Tóth; Article No. 77; pp. 77:1–77:4

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Mesh generation is a fundamental problem in computational geometry, geometric modeling, computer graphics, scientific computing and engineering simulations. There has been a growing interest in polyhedral meshes as an alternative to tetrahedral or hex-dominant meshes [17]. Polyhedra are less sensitive to stretching, which enables the representation of complex geometries without excessive refinement. In addition, polyhedral cells have more neighbors even at corners and boundaries, which offers better approximations of gradients and local flow distributions. Even compared to hexahedra, fewer polyhedral cells are needed to achieve a desired accuracy in certain applications. This can be very useful in several numerical methods [7]. In particular, the accuracy of a number of important solvers, e.g., the two-point flux approximation for conservation laws [14], greatly benefits from a conforming mesh which is *orthogonal* to its dual as naturally satisfied by Voronoi meshes. Such solvers play a crucial role in hydrology [19] and computational fluid dynamics [8].

A conforming volume mesh exhibits two desirable properties *simultaneously*: (1) a decomposition of the enclosed volume, and (2) a reconstruction of the bounding surface. Conforming Delaunay meshing is well-studied [11], but Voronoi meshing is less mature. A common practical approach to polyhedral meshing is to dualize a tetrahedral mesh and *clip* each cell by the bounding surface [15]. Unfortunately, clipping does not guarantee the convexity and connectedness of cells [13]. Another approach is to *mirror* Voronoi sites across the surface [20], but we are not aware of any approximation guarantees in this category.

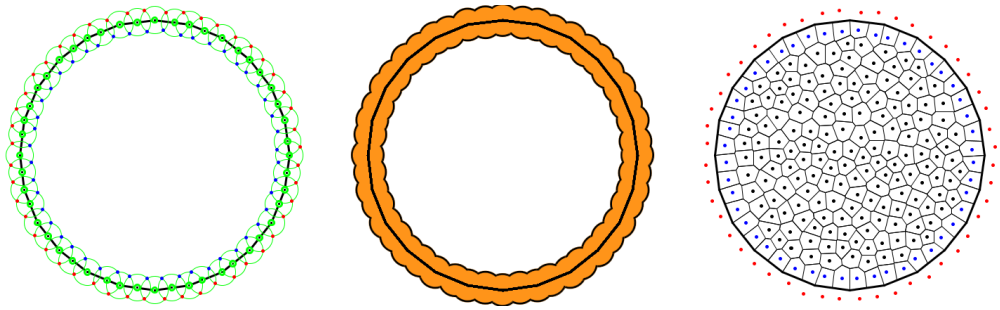
The desired mesher takes as input a description of the geometric model for which a conforming Voronoi mesh is to be generated, e.g., a set of surface samples or a complete computer-aided design (CAD) model. Having access to a CAD model allows the mesher to accurately sample new points as needed; a crucial requirement for high quality meshing. Deferring the sampling problem, we start in an idealized setting that allows us to establish strong theoretical guarantees on the correctness of the underlying approach. In a forthcoming paper [3], we describe the design and implementation of the complete VoroCrust algorithm, which can handle realistic inputs possibly having sharp features, can estimate a sizing function and generate samples, and can guarantee the quality of the output mesh.

In this multimedia contribution, we illustrate the basic constructions underlying the proposed algorithm and briefly explain why it works. A number of meshes generated by a preliminary implementation is shown. Finally, we highlight the main technical challenges imposed by realistic inputs along with an illustration of each. The rest of this abstract follows the same structure and concludes by a summary of the methods used in preparing the video content.

2 The abstract algorithm

An abstract version of the proposed algorithm, geared towards volumes bounded by smooth surfaces, can be described as follows. The points used in defining the Voronoi diagram are referred to as *seeds*. Figure 1 illustrates the steps in 2D.

1. Take as input a sample $\mathcal{P}$ on the surface $\mathcal{M}$ bounding the volume $\mathcal{O}$.
2. Define a ball B_i centered at each sample p_i, with a suitable radius r_i, and let $\mathcal{U} = \cup_i B_i$.
3. Initialize the set of seeds $\mathcal{S}$ with the corners of $\partial\mathcal{U}$, $\mathcal{S}^{\uparrow}$ and $\mathcal{S}^{\downarrow}$, on both sides of $\mathcal{M}$.
4. Optionally, generate additional seeds $\mathcal{S}^{\downarrow\downarrow}$ in the interior of $\mathcal{O}$, and include $\mathcal{S}^{\downarrow\downarrow}$ into $\mathcal{S}$.
5. Compute the Voronoi diagram $\mathrm{Vor}(\mathcal{S})$ and retain the cells with seeds in $\mathcal{S}^{\downarrow} \cup \mathcal{S}^{\downarrow\downarrow}$ as the volume mesh $\hat{\mathcal{O}}$, where the facets between $\mathcal{S}^{\uparrow}$ and $\mathcal{S}^{\downarrow}$ yield a surface approximation $\hat{\mathcal{M}}$.



(a) Seeds from union of balls. **(b)** Isotopic reconstruction. **(c)** Conforming decomposition.

Figure 1 Screen captures from the video illustrating the algorithmic steps on a planar curve.

Under appropriate conditions on the sampling, the union of balls U is isotopic to the bounding surface $\mathcal{M}$ [10]. In addition, the Voronoi facets in $\mathrm{Vor}(\mathcal{S})$ separating $\mathcal{S}^{\uparrow}$ and $\mathcal{S}^{\downarrow}$, which constitute the surface reconstruction $\hat{\mathcal{M}}$, are in fact the medial axis of U [6], which is isotopic to $\mathcal{M}$ if the balls have suitable spacing and radii [9]. The sparsity condition helps in bounding the quality of mesh elements; the cells are fat. Finally, the enclosed volume can easily be refined without disrupting the surface reconstruction.

In particular, we assume $\mathcal{P}$ is an ϵ-sample with a weak σ-sparsity condition dictating $\mathbf{d}(p_i, p_j) \geq \sigma\epsilon \cdot \mathrm{lfs}(p_j)$ whenever $\mathrm{lfs}(p_i) \geq \mathrm{lfs}(p_j)$. The union of balls $\mathcal{U}$ is defined by setting $r_i = \delta \cdot \mathrm{lfs}(p_i)$. Fixing $\delta = 2\epsilon$, the sampling conditions for non-uniform approximation are satisfied when ϵ is sufficiently small [10]. We also fix $\sigma = 3/4$ to give $\mathcal{U}$ a particularly nice structure which guarantees all samples appear as vertices in the surface reconstruction.

In this setting, we establish a number of geometric properties of the construction. This allows us to bound the distance between seeds in $\mathcal{S}^{\uparrow} \cup \mathcal{S}^{\downarrow}$ and their separating facets, which in turns bounds the inradius of Voronoi cells in the resulting decomposition $\hat{\mathcal{O}}$. To bound the outradius of cells, we proceed to seed the interior of $\mathcal{O}$. Assuming $\mathcal{O}$ is shifted and scaled to fit in the root box of the octree, we refine octree boxes until their sizes match an extension of lfs. Once refinement terminates, we insert into $\mathcal{S}^{\downarrow\downarrow}$ additional seeds at the centers of all boxes which are empty of seeds in $\mathcal{S}^{\uparrow} \cup \mathcal{S}^{\downarrow}$. This scheme allows us to bound the ratio of outradius to inradius for all cells, and also to bound the number of seeds. See [2] for the details and [4] for some earlier related work using a different scheme.

3 Dealing with realistic inputs

Unlike the idealized setting studied above, realistic inputs pose a number of challenges. Namely, sharp features do not fit well with the ϵ-sampling paradigm and require extra care to bound the quality of mesh elements in their neighborhoods. Even without sharp features, estimating the local feature size is impractical. In addition, for applications requiring surface approximations with good normals, extra steps are needed to iron out any irregularities that the basic approach might yield. We address these issues in a forthcoming publication [3].

4 Video production

We used *ParaView* [5] and the *Processing* programming language [18] to generate our animations. Speech was synthesized via the *Bing Speech API* [12] and the video was composed using *OpenShot* [16]. A low-quality version of the video will be hosted on the conference website [1].

References

1. A. Abdelkader, C. Bajaj, M. Ebeida, A. Mahmoud, S. Mitchell, J. Owens, and A. Rushdi. VoroCrust illustrated: theory and challenges (video content). http://computational-geometry.org/SoCG-videos/socg18video/videos/VoroCrust/.
2. A. Abdelkader, C. Bajaj, M. Ebeida, A. Mahmoud, S. Mitchell, J. Owens, and A. Rushdi. Sampling conditions for conforming Voronoi meshing by the VoroCrust algorithm. In *34th International Symposium on Computational Geometry (SoCG 2018)*, pages 1:1–1:16, 2018.
3. A. Abdelkader, C. Bajaj, M. Ebeida, A. Mahmoud, S. Mitchell, J. Owens, and A. Rushdi. VoroCrust: Voronoi meshing without clipping. *Manuscript*, In preparation.
4. A. Abdelkader, C. Bajaj, M. Ebeida, and S. Mitchell. A Seed Placement Strategy for Conforming Voronoi Meshing. In *Canadian Conference on Computational Geometry*, 2017.
5. J. Ahrens, B. Geveci, and C. Law. Paraview: An end-user tool for large-data visualization. In C. Hansen and C. Johnson, editors, *Visualization Handbook*, pages 717 – 731. Butterworth-Heinemann, 2005.
6. N. Amenta and R.-K. Kolluri. The medial axis of a union of balls. *Computational Geometry*, 20(1):25 – 37, 2001. Selected papers from the 12th Annual Canadian Conference.
7. N. Bellomo, F. Brezzi, and G. Manzini. Recent techniques for PDE discretizations on polyhedral meshes. *Mathematical Models and Methods in Applied Sciences*, 24(08):1453–1455, 2014.
8. T. Brochu, C. Batty, and R. Bridson. Matching fluid simulation elements to surface geometry and topology. *ACM Trans. Graph.*, 29(4):47:1–47:9, 2010.
9. F. Chazal and D. Cohen-Steiner. A condition for isotopic approximation. *Graphical Models*, 67(5):390 – 404, 2005. Solid Modeling and Applications.
10. F. Chazal and A. Lieutier. Smooth manifold reconstruction from noisy and non-uniform approximation with guarantees. *Computational Geometry*, 40(2):156 – 170, 2008.
11. S.-W. Cheng, T. Dey, and J. Shewchuk. *Delaunay Mesh Generation*. CRC Press, 2012.
12. Microsoft Corporation. Bing Speech API. https://azure.microsoft.com/en-us/services/cognitive-services/speech/.
13. M. Ebeida and S. Mitchell. Uniform random Voronoi meshes. In *International Meshing Roundtable (IMR)*, pages 258–275, 2011.
14. R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. In *Techniques of Scientific Computing (Part 3)*, volume 7 of *Handbook of Numerical Analysis*, pages 713 – 1018. Elsevier, 2000.
15. R.-V. Garimella, J. Kim, and M. Berndt. Polyhedral mesh generation and optimization for non-manifold domains. In *International Meshing Roundtable (IMR)*, pages 313–330. Springer International Publishing, 2014.
16. OpenShot Studios, LLC. OpenShot Video Editor. http://www.openshot.org/.
17. M. Peric and S. Ferguson. The advantage of polyhedral meshes. *Dynamics - Issue 24*, page 4–5, Spring 2005. The customer magazine of the CD-adapco Group, currently maintained by Siemens at http://siemens.com/mdx. The issue is available at http://mdx2.plm.automation.siemens.com/magazine/dynamics-24 (accessed March 29, 2018).
18. C. Reas and B. Fry. *Processing: A Programming Handbook for Visual Designers and Artists*. The MIT Press, 2014.
19. M. Sents and C. Gable. Coupling LaGrit Unstructured Mesh Generation and Model Setup with TOUGH2 Flow and Transport. *Comput. Geosci.*, 108(C):42–49, 2017.
20. M. Yip, J. Mohle, and J. Bolander. Automated modeling of three-dimensional structural components using irregular lattices. *Computer-Aided Civil and Infrastructure Engineering*, 20(6):393–407, 2005.