

Calcul et application des diagrammes de Voronoi restreints

Vincent Nivoliers, LIRIS, Université Lyon 1

Le calcul de diagrammes de Voronoi restreints est récemment devenu un outil central pour de nombreuses applications, en particuliers en génération de maillages [8, 7], simulation de particules [10, 3] et en transport optimal [9, 6]. Avec ces applications sont nés des besoins, à la fois pour calculer efficacement de tels diagrammes et pour évaluer et optimiser des fonctions définies dessus. Dans cet exposé, nous nous attacherons particulièrement aux algorithmes de calcul de diagrammes de Voronoi restreints, et aux problèmes auxquels ils répondent.

1 Diagrammes de Voronoi restreints

Calculer un diagramme de Voronoi ou la triangulation de Delaunay qui lui est duale est un problème classique de géométrie algorithmique. Étant donné un ensemble de sites $\mathbf{V} = \{\mathbf{v}_k\}_{k=2}^n \subset \mathbb{R}^d$, la cellule de Voronoi d'un site \mathbf{v}_k dans le diagramme de \mathbf{V} est l'ensemble des points

$$\text{Vor}_{\mathbf{V}}(\mathbf{v}_k) = \{\mathbf{p} \in \mathbb{R}^d, d(\mathbf{v}_k, \mathbf{p}) < d(\mathbf{v}_\ell, \mathbf{p}) \text{ pour tout } \mathbf{v}_\ell \in \mathbf{V}, \mathbf{v}_\ell \neq \mathbf{v}_k\},$$

où $d(\mathbf{v}_k, \mathbf{p})$ est la distance entre le site \mathbf{v}_k et le point \mathbf{p} . Calculer le diagramme de Voronoi de \mathbf{V} dans \mathbb{R}^d consiste à calculer l'ensemble de ces cellules de Voronoi. Des algorithmes efficaces voire optimaux existent pour calculer ces diagrammes [2, 1].

Le diagramme de Voronoi restreint (Figure 1) consiste simplement à ne définir le diagramme que dans une région d'intérêt $\mathbf{S} \subset \mathbb{R}^d$:

$$\text{Vor}_{\mathbf{V}}(\mathbf{v}_k) \cap \mathbf{S} = \{\mathbf{p} \in \mathbf{S}, d(\mathbf{v}_k, \mathbf{p}) < d(\mathbf{v}_\ell, \mathbf{p}) \text{ pour tout } \mathbf{v}_\ell \in \mathbf{V}, \mathbf{v}_\ell \neq \mathbf{v}_k\}.$$

De tels diagrammes apparaissent notamment dans le cadre de la reconstruction de surfaces à partir de nuages de points à la fois comme objet théorique pour prouver la correction de la reconstruction [5], et comme outil pratique qu'il est nécessaire de calculer pour obtenir la reconstruction [4].

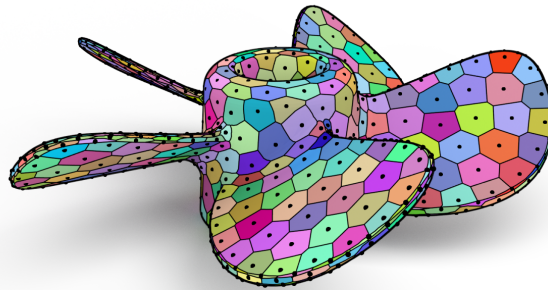


Figure 1: Un diagramme de Voronoi restreint.

2 Profiter la restriction

L'algorithme naïf et souvent le plus efficace pour calculer un diagramme de Voronoi restreint revient à calculer le diagramme de Voronoi complet puis à en calculer l'intersection avec le domaine d'intérêt. Cet algorithme fonctionne souvent très bien car les algorithmes de calcul de diagrammes de Voronoi sont efficaces. Il est toutefois limité lorsque :

- peu de sites contribuent au diagramme restreint dans la zone d'intérêt ;
- la zone d'intérêt est de dimension plus faible que l'espace ambiant.

Dans ces deux cas, calculer le diagramme complet est une perte de temps car la complexité du diagramme complet est potentiellement bien plus élevée que celle du diagramme restreint. Pour profiter de la restriction et ne calculer le diagramme que dans les zones utiles, il est nécessaire de déterminer, étant donné l'espace de restriction, l'ensemble des sites concernés. Deux méthodes utilisant des requêtes de plus proche voisinage et fondées sur le même critère ont émergé à partir de ce constat [10, 7]. Nous montrerons dans cet exposé les avantages et limites de ces méthodes, ainsi que des pistes d'améliorations et des parallèles avec des algorithmes classiques de calcul de diagramme de Voronoi. Nous proposerons également quelques problèmes ouverts pour poursuivre ces recherches.

Références

- [1] C Bradford Barber, David P Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483, 1996.
- [2] Jean-Daniel Boissonnat, Olivier Devillers, Monique Teillaud, and Mariette Yvinec. Triangulations in cgal. In *Proceedings of the sixteenth annual symposium on Computational geometry*, pages 11–18. ACM, 2000.
- [3] Fernando de Goes, Corentin Wallez, Jin Huang, Dmitry Pavlov, and Mathieu Desbrun. Power particles: an incompressible fluid solver based on power diagrams. *ACM Trans. Graph.*, 34(4):50, 2015.
- [4] Tamal K Dey and Joachim Giesen. Detecting undersampling in surface reconstruction. In *Discrete and Computational Geometry*, pages 329–345. Springer, 2003.
- [5] Herbert Edelsbrunner and Nimish R Shah. Triangulating topological spaces. In *Proceedings of the tenth annual symposium on Computational geometry*, pages 285–292. ACM, 1994.
- [6] Bruno Lévy. A numerical algorithm for l2 semi-discrete optimal transport in 3d. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(6):1693–1715, 2015.
- [7] Bruno Lévy and Nicolas Bonneel. Variational anisotropic surface meshing with voronoi parallel linear enumeration. *Proceedings of the 21st international meshing roundtable*, pages 349–366, 2013.
- [8] Bruno Lévy and Yang Liu. L p centroidal voronoi tessellation and its applications. In *ACM Transactions on Graphics (TOG)*, volume 29, page 119. ACM, 2010.
- [9] Quentin Mérigot. A multiscale approach to optimal transport. In *Computer Graphics Forum*, volume 30, pages 1583–1592. Wiley Online Library, 2011.
- [10] Chris Rycroft. Voro++: A three-dimensional voronoi cell library in c++. *Lawrence Berkeley National Laboratory*, 2009.