Computing 2D Periodic Centroidal Voronoi Tessellation

Dong-Ming Yan
Project ALICE, INRIA
Nancy, France
dongming.yan@inria.fr

Kai Wang
Project ALICE, INRIA / Gipsa-lab, CNRS
Grenoble, France
kai.wang@gipsa-lab.inpg.fr

Bruno Lévy
Project ALICE, INRIA
Nancy, France
bruno.levy@inria.fr

Laurent Alonso
Project ALICE, INRIA
Nancy, France
laurent.alonso@inria.fr

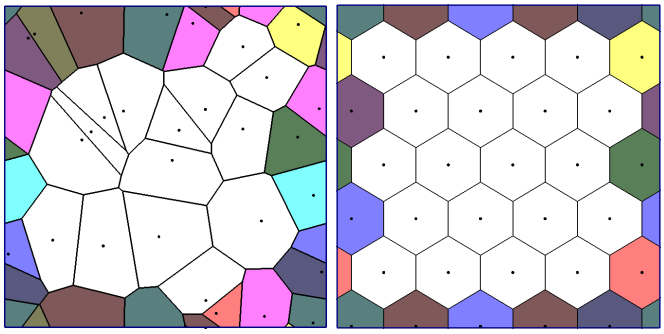
***Abstract*—In this paper, we propose an efficient algorithm to compute the centroidal Voronoi tessellation in 2D periodic space. We first present a simple algorithm for constructing the periodic Voronoi diagram (PVD) from a Euclidean Voronoi diagram. The presented PVD algorithm considers only a small set of periodic copies of the input sites, which is more efficient than previous approaches requiring full copies of the sites (9 in 2D and 27 in 3D). The presented PVD algorithm is applied in a fast Newton-based framework for computing the centroidal Voronoi tessellation (CVT). We observe that full-hexagonal patterns can be obtained via periodic CVT optimization attributed to the convergence of the Newton-based CVT computation.**

***Keywords*-Periodic Voronoi diagram, Delaunay triangulation, centroidal Voronoi tessellation, hexagonal pattern**

I. Introduction

The Voronoi diagram is a fundamental geometric structure which was first proposed in the 1900s [1]. For a given set of sites (or generators) in a domain, the Voronoi diagram is defined as a collection of Voronoi cells each of which contains the domain points that are closer to the corresponding site of the cell than to any other site. The Voronoi diagram has been extensively studied over one hundred years and successfully used in numerous applications of various disciplines, such as natural science, industry design, mathematics, geography and so on. There exist different variants of the Voronoi diagram, defined under different distance metrics (e.g. geodesic distance on surfaces, Euclidean distance, hyperbolic space distance and anisotropic distance), or with different constraints (e.g. diagram restricted to a compact domain, sites constrained on domain boundaries and diagram periodicity).

Centroidal Voronoi tessellation (CVT) is a special kind of Voronoi diagram that requires each site to coincide with the mass center of its corresponding Voronoi cell [2]. CVT has many nice properties which are useful in data clustering, optimal quadrature and quantization. One important property is that the cells of an optimal (isotropic) CVT are congruent to regular hexagons in asymptotic sense [3]. This property ensures a uniform and regular partition of the considered domain by CVT, which is desirable in many engineering applications such as mesh generation.



(a) (b)

Figure 1. Periodic centroidal Voronoi tessellation of 30 sites (the polygons with the same color belong to the same periodic Voronoi cell) : (a) PVD of initial sites; (b) periodic CVT.

When using Voronoi diagram for the generation of high-quality meshes, we can impose different application-related criteria. One particular yet important application is to generate mesh in a periodic space under a periodic distance metric. As an example, for a 2D squared domain, this spatial periodicity implies intuitively that when a particle leaves the domain crossing a certain edge, it immediately returns to the domain passing through the opposite edge. The Voronoi diagram constructed in such a periodic space is called a periodic Voronoi diagram (PVD, see Figure 1(a) for an example). PVD has been a useful tool in diverse application fields where spatial periodicity is a frequently encountered phenomenon, for instance the micro-structural modeling of materials [4]–[7] and the simulation of fluid dynamics [8], [9].

The main contributions of this paper include :

1. a simple algorithm for computing 2D periodic Voronoi diagram. Compared to the previous approaches, we use only a small number of *mirrors* (defined in Section III) instead of using the full copies;
2. a periodic CVT computation algorithm based on the quasi-Newton solver, which has faster convergence speeds compared to the classical Lloyd iteration;
3. the observation that full-hexagonal Voronoi cells can be obtained via periodic CVT optimization for some specific values of the number of sites.

A. Related work

The Voronoi diagram in a metric space is a special kind of tessellation of that space. The reader is referred to [10]–[12] for details of Voronoi diagram and its dual concept — Delaunay triangulation.

The definition of Voronoi diagram (and thus its optimized version CVT) incorporates the notion of distance between cell generators and domain points (see Section II). Different distance metrics have been studied. Peyré and Cohen [13] constructed geodesic CVT on mesh surfaces and used it for shape segmentation and remeshing. Alternatively, Yan et al. [14] computed the constrained and restricted CVT on mesh surfaces based on Euclidean distance metric, in the context of isotropic remeshing. Later, Lévy and Liu [15] generalized this Euclidean CVT to higher order, which is useful in 2D quad-dominant and 3D hex-dominant mesh generations. Rong et al. [16] introduced an algorithm which computes CVT in hyperbolic space. The advantage of doing computation in this space is that it allows us to easily generate high-quality meshes from high-genus models. Ju et al. [17] studied the properties of CVT on spherical domain and applied it to climate and global modeling. Finally, Tournois et al. [18] presented a method to compute Euclidean CVT in 2D constrained domain.

As mentioned previously and reported in [4]–[9], periodic Voronoi diagram has been widely used in diverse applications. However, in all these applications, the PVD is computed using full copies of primary sites, which is not an efficient solution especially when we have a large number of sites. This observation motivated our work on efficient computation of PVD, a problem not well addressed in the literature. In [19], Fu et al. discussed the efficient computation of Voronoi diagram on periodic graphs by exploiting the relation between nearest neighbor search on periodic graphs and some geometric convex-distance functions. However, their research problem is different from that studied in this paper. In particular, although in [19] the space is a periodic graph with repeated pattern, the Voronoi diagram is an ordinary one which is not defined under a periodic distance metric (its formal definition will be given later in Section II-A).

The dual of PVD — the periodic Delaunay triangulation — has also been studied. Caroli and Teillaud [20] presented an elegant solution for computing Delaunay triangulation in 3D periodic space based on directly defining the geometric predictors in periodic space (periodic kernel). This algorithm computes periodic Delaunay triangulation directly in 3D periodic space, and guarantees that the obtained triangulation is a simplicial complex in the 3D periodic space.

B. Outline

We first give some mathematical definitions in Section II. We present a simple and efficient algorithm for computing periodic Voronoi diagram in Section III. We show how the presented PVD computation algorithm can be applied in a CVT framework in Section IV. Experimental results are reported in Section V, and we draw our conclusion in Section VI.

II. Problem formulation

Given a set of sites $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$ in 2D, the Voronoi diagram of $\mathbf{X}$ is defined by a collection of n Voronoi cells $\{\Omega_i\}_{i=1}^n$, where

$$\Omega_i = \{\mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x} - \mathbf{x}_i\| \leq \|\mathbf{x} - \mathbf{x}_j\|, \forall j \neq i\}.$$

Each Voronoi cell Ω_i is the intersection of a set of 2D half-spaces, delimited by the bisecting planes of the Delaunay edges incident to the site $\mathbf{x}_i$.

A. Periodic Voronoi diagram

The 2D periodic space is a 2D flat torus $\mathbb{T}^2$, which is homeomorphic to the surface of a torus in 3D (as shown in Figure 2). The flat torus is defined as the quotient space

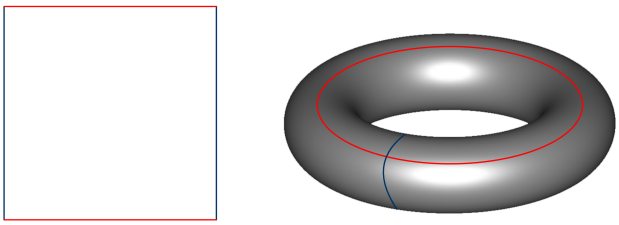


Figure 2. A periodic domain in 2D (left) is homeomorphic to the surface of a torus in 3D (right).

$\mathbb{T}^2 = \mathbb{R}^2/G$, where G is the group $(\mathbf{u} * \mathbb{Z}^2, +)$ and $\mathbf{u} := (1, 1)$. We use the unit square $\mathcal{D}_0 = [0, 1) \times [0, 1)$ as the representative (or primary) domain of the periodic space in 2D for simplicity. The elements of $\mathbb{T}^2$ are the equivalence classes under equivalence relation $\mathbf{p}_1 \sim \mathbf{p}_2 \leftrightarrow \mathbf{p}_1 - \mathbf{p}_2 \in \mathbf{u} * \mathbb{Z}^2$. The distance metric in $\mathbb{T}^2$ is defined by $d_{\mathbb{T}}(\mathbf{p}, \mathbf{q}) := \min(d_{\mathbb{R}}(\mathbf{p}', \mathbf{q}'))$, where $\mathbf{p}'$ and $\mathbf{q}'$ are equivalent to $\mathbf{p}$ and $\mathbf{q}$ under the quotient map, respectively. We refer the reader to [20] for more details of the definition of periodic space.

Assuming that the input sites $\mathbf{X}$ are located in the primary domain $\mathcal{D}_0 = [0, 1) \times [0, 1)$. Similar to the Euclidean Voronoi diagram, the 2D periodic Voronoi diagram of sites $\mathbf{X}$ is a collection of periodic Voronoi cells $\{\widetilde{\Omega}_i\}_{i=1}^n$ under the distance metric $d_{\mathbb{T}}$, where

$$\widetilde{\Omega}_i = \{\mathbf{x} \in \mathbb{R}^2 \mid d_{\mathbb{T}}(\mathbf{x}, \mathbf{x}_i) \leq d_{\mathbb{T}}(\mathbf{x}, \mathbf{x}_j), \forall j \neq i\}.$$

It is easy to see that the PVD in $\mathbb{R}^2$ is the periodic extension of the PVD in the primary domain $\mathcal{D}_0$. Therefore, it suffices to confine our study in $\mathcal{D}_0$. There are two types of Voronoi cells in a PVD, inner cells and boundary cells. An inner cell is the same as a Euclidean Voronoi cell, which is entirely inside $\mathcal{D}_0$. The cells across the periodic boundary are boundary cells. Each boundary cell is composed of several

visually disconnected polygons in $\mathcal{D}_0$ (cf. Figure 1, where the boundary cells are shaded).

B. Centroidal Voronoi tessellation

Centroidal Voronoi tessellation (CVT) is a special kind of Voronoi tessellation, which is defined as a critical point of the energy function [2] :

$$F(\mathbf{X}) = \sum_{i=1}^{n} \int_{\Omega_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 \, \mathrm{d}\sigma, \tag{1}$$

where $\rho(\mathbf{x}) > 0$ is a user-defined density function, and $\mathrm{d}\sigma$ is the area differential.

The partial derivative of the CVT energy function with respect to each site is given by [21] :

$$\frac{\partial F}{\partial \mathbf{x}_i} = 2m_i(\mathbf{x}_i - \mathbf{g}_i), \tag{2}$$

where $m_i = \int_{\Omega_i} \rho(\mathbf{x}) \, \mathrm{d}\sigma$ is the mass of the cell Ω_i, and $\mathbf{g}_i = \frac{\int_{\Omega_i} \rho(\mathbf{x})\mathbf{x} \, \mathrm{d}\sigma}{\int_{\Omega_i} \rho(\mathbf{x}) \, \mathrm{d}\sigma}$ is the weighted centroid. The CVT is uniform when the density ρ is constant. The minimizer of this energy is obtained when each site coincides with the mass center of its Voronoi cell, i.e. when we have a CVT.

The CVT energy function in periodic space is defined by replacing the Euclidean distance metric in Eqn. (1) with the periodic distance metric $d_{\mathbb{T}}$:

$$F_{\mathbb{T}}(\mathbf{X}) = \sum_{i=1}^{n} \int_{\widetilde{\Omega}_i} \rho(\mathbf{x}) d_{\mathbb{T}}(\mathbf{x}, \mathbf{x}_i)^2 \, \mathrm{d}\sigma. \tag{3}$$

In this paper, we propose an algorithm to compute periodic CVT in $\mathbb{T}^2$. Our algorithm has two main components (a) compute the 2D periodic Voronoi diagram (Section III), and (b) minimize the periodic CVT energy function (Section IV). We shall explain the details of each component in the following sections. The uniform density will be used in the rest of this paper to study the properties of CVT in 2D periodic space.

III. Periodic Voronoi diagram computation

We first introduce the main idea of our approach, and then present the details of the PVD computation algorithm.

A. Overview

As discussed in Section I-A, existing methods need to duplicate 9 copies in 2D (or 27 copies in 3D, respectively) for computing the periodic Voronoi diagram (e.g. in [4]–[9]). After this full duplication, the PVD can be easily computed in the Euclidean space instead of the periodic space, as explained below.

Suppose that we now have n input sites $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$ located in the 2D primary domain $\mathcal{D}_0 = [0,1) \times [0,1)$. Existing methods first duplicate the sites $\mathbf{X}$ in all the 8 neighboring domains $\{\mathcal{D}_j\}_{j=1}^8$ of the primary domain. The duplicated sites (or mirror sites) are denoted as $\{\mathbf{x}_{i,j}\}, i = 1..n, j = 1..8$. The Euclidean Voronoi diagram of all the sites is then computed. The periodic Voronoi diagram of $\mathbf{X}$ is obtained by intersecting all the Voronoi cells with domain $\mathcal{D}_0$. The intersecting results of mirror cells are assigned to the periodic Voronoi cells of the corresponding primary sites. The basic idea of the PVD computation based on full copies is illustrated in Figure 3. This straightforward approach will become inefficient when the number of sites increases.

We notice that actually only a small set of mirror sites contributes to the periodic Voronoi diagram, which are the mirrors whose Voronoi cells intersect with the primary domain $\mathcal{D}_0$. Based on this observation, we derive a simple algorithm for computing PVD by using only the necessary periodic mirror sites.

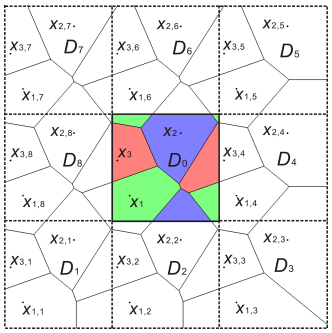
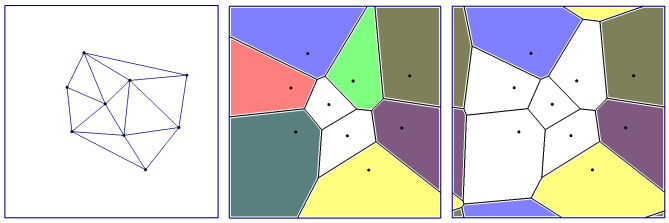


Figure 3. PVD computation based on full copies.

B. Algorithm description

Our PVD computation algorithm consists of three steps, as shown in Figure 4. First, we build the Euclidean Voronoi diagram of input sites $\mathbf{X}$ based on Delaunay triangulation. Next, we detect the sites whose Voronoi cells intersect with the periodic domain boundary. Finally, we repeatedly insert necessary mirror sites to build the PVD. We present in the following the details of these steps.

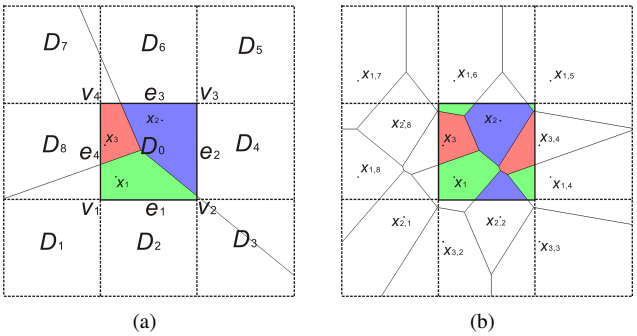


(a) (b) (c)

Figure 4. Main steps for computing periodic Voronoi diagram : (a) Delaunay triangulation; (b) compute clipped Voronoi diagram in the primary domain; (c) compute periodic Voronoi diagram by inserting periodic mirror sites.

1) Voronoi diagram construction: We build the Delaunay triangulation of the input sites $\mathbf{X}$ (Figure 4(a)). The Voronoi diagram $\{\Omega_i\}$ of $\mathbf{X}$ is deduced from the Delaunay triangulation.

2) Boundary sites detection: In this step, we detect the initial candidates of boundary sites. Since both the Voronoi cells and the domain $\mathcal{D}_0$ are convex, the cells that have non-empty intersection with the domain can be easily detected. The Voronoi cells of the sites on the convex hull of $\mathbf{X}$ are infinite, and those cells are first marked as boundary cells.



(a) (b)

Figure 5. Illustration of mirror sites insertion : (a) clipped Voronoi diagram; (b) periodic Voronoi diagram after inserting 11 mirror sites.

Then we traverse the adjacent Delaunay triangles of each unmarked site. If a site has one or more adjacent triangles whose circumcenter is outside of the domain $\mathcal{D}_0$, then the corresponding cell is marked as a boundary cell, otherwise as an inner cell. We compute the clipped Voronoi diagram by clipping the boundary Voronoi cells against domain boundary using the Sutherland-Hodgman clipping algorithm [22], as shown in Figure 4(b). The sites of boundary cells are tagged as initial boundary sites.

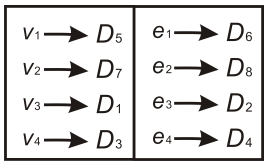
3) Periodic mirrors insertion: Once the initial boundary sites are tagged, we are ready to insert periodic mirrors of them to build the PVD. Actually, only the mirrors of the initial boundary sites may contribute to the final PVD. The reason is that we insert mirror sites in neighboring domains which are outside of the convex hull of the primary sites $\mathbf{X}$, therefore the initial inner cells are kept unchanged during the insertion process and do not contribute to the PVD. The insertion of mirror sites is guided by the following two rules :

(a) If the Voronoi cell of a site intersects with the horizontal periodic boundary edge e_1 (resp. e_3), we insert a mirror copy in the upper (resp. nether) neighboring domain of the site's belonging domain; similarly, if the cell intersects with the vertical edge e_2 (resp. e_4), we insert a mirror copy in the left (resp. right) neighboring domain of the site's belonging domain (see Figure 5).

(b) If the Voronoi cell of a site contains a periodic boundary vertex $v_{j=1..4}$, we insert a mirror copy in the diagonal neighboring domain of the site's belonging domain that is incident to the vertex $v_{[(j-1+2)mod4]+1}$, i.e. the diagonal opposite vertex of v_j in $\overline{\mathcal{D}_0}$.

The idea behind these two rules is that we (only) need to insert the mirrors whose cells may intersect the equivalent classes of the intersected periodic boundary elements. Also note that the above rules are applicable to the mirror sites insertion of both the primary sites and the already inserted mirrors.

The mapping between intersected boundary edge/vertex to neighboring domains for primary sites is given in the inset of this paragraph. For example, as shown in Figure 5(a),



the Voronoi cell of $\mathbf{x}_2$ intersects with e_2, e_3 and v_3, the corresponding mirror sites $\mathbf{x}_{2,8}, \mathbf{x}_{2,2}$ and $\mathbf{x}_{2,1}$ are inserted, respectively. Each time after inserting mirror sites, we update the Euclidean Voronoi diagram for all the sites (both primary sites and mirrors) based on the efficient incremental Delaunay triangulation algorithm of CGAL [23], and the intersections of the cells with $\mathcal{D}_0$ are computed. If the Voronoi cells of the inserted mirror sites intersect with the boundary edges $e_{i=1..4}$ or vertices $v_{j=1..4}$, we insert new mirrors of those intersecting mirror sites if the new mirrors are not in the diagram. The insertion process is still guided by the aforementioned two rules. However, the specific mapping between boundary elements and neighboring domains is relevant to the belonging domain of the intersecting mirror site and thus is different from that for the primary sites.

The insertion process terminates when there is no more mirror that can be inserted, as for example in Figure 5(b). The termination of our algorithm is guaranteed, since in the worst case each primary boundary site will be duplicated 8 times in all the neighboring domains. The resulting clipped Voronoi cells in $\mathcal{D}_0$ of the mirror sites are assigned to their corresponding primary sites, as shown in Figure 4(c). Since we use only a small set of mirror sites, our method is more efficient than existing methods based on full copies when the number of sites increases (see Section V). We give a brief proof of the correctness of our algorithm in Appendix.

IV. Periodic Centroidal Voronoi Tessellation

We use the L-BFGS method to minimize the periodic CVT energy function (Eqn. (3)). L-BFGS method needs to evaluate $F_{\mathbb{T}}(\mathbf{X}^{(k)})$ and $\partial F_{\mathbb{T}}(\mathbf{X}^{(k)})$ in a series of iterations [24]. The evaluation for an inner site is the same as in the minimization of conventional Euclidean CVT energy, since its Voronoi cell lies completely in the primary domain. For the boundary sites, we simply transform the periodic Voronoi cell to Euclidean space to form a single connected cell, so that the computation can be done efficiently without modifying the solver. This process is illustrated in Figure 6.

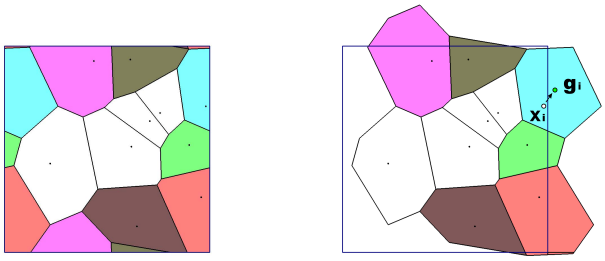


Figure 6. Periodic energy function evaluation and gradient computation.

After each iteration, the obtained new position $\mathbf{x}_i^{(k+1)}$ is mapped to the primary domain, and finally we set this primary point as the new position of the boundary site $\mathbf{x}_i$.

V. Experimental results

We present some results of our periodic CVT computation algorithm. All the experiments were conducted on a laptop PC with a 2.4GHz Intel Duo-Core processor and 3GB memory. We use CGAL [23] for computing Delaunay triangulation in Section III.

Efficiency. Our PVD algorithm is very efficient since it only uses a small set of periodic mirror sites, instead of using the full copies. Once the 2D Delaunay triangulation is built in the primary domain, it is ready to compute PVD efficiently. The timing curve of the PVD computation and that of the Delaunay triangulation against increasing number of sites are shown in Figure 7.

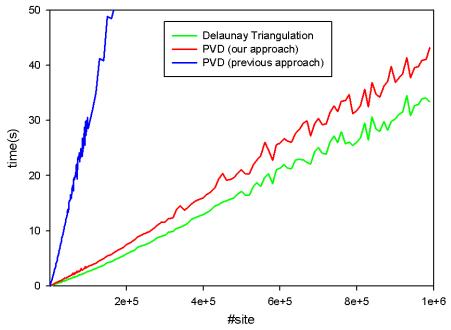


Figure 7. Timing of PVD computation against number of sites.

We use the presented PVD algorithm in a fast CVT computation framework [24] with a slight modification of the PVD for gradient computation and function evaluation under Euclidean metric. We observed that periodic CVT always has more regular patterns than standard CVT, since it has more freedom in periodic space and simpler homotopy. Several results are shown in Figure 8. The energy of periodic CVT is always smaller than that of standard CVT during iterations as shown in Figure 9.

Comparison. We compare our PVD algorithm with previous approaches that use full copies. The statistic result of the number of mirrors used by our PVD algorithm against increasing number of sites is shown in Figure 10(a). It can be seen that only a small number of mirror sites is used even for very large point sets. Figure 10(b) shows the percentage of the mirror sites with respect to the number of primary sites. The percentage decreases when the number of sites increases. Furthermore, it is not surprising to see that the number of copies is approximately of the same order as the squared root of the number of primary sites, since in our experiments the initial sites are uniformly distributed in the primary domain. Our algorithm is able to compute PVD for very large data set as efficiently as Delaunay triangulation, while the computation cost will increase quickly for the

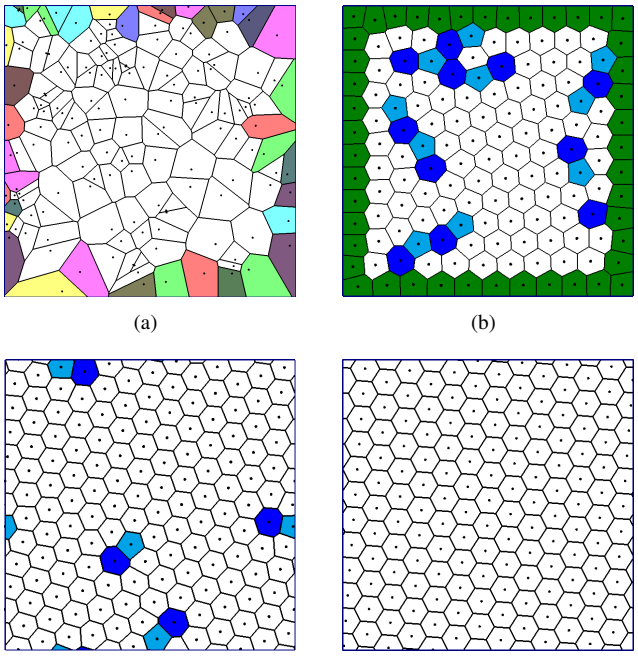


Figure 8. Results of CVT and periodic CVT for 150 sites. (a) Input sites; (b) result of standard CVT (green cells are clipped boundary cells and blue cells are non-hex cells); (c) periodic CVT result of Lloyd's method [25] after 3k iterations; (d) result of periodic CVT with 80 Newton iterations (see also the accompanied videos).

approaches using full copies, as illustrated in Figure 7. More precisely, our method can accelerate the PVD computation by roughly a factor of 8, as shown by the red and blue curves in the figure.

Note that one can also implement a 2D version of the 3D periodic Delaunay triangulation provided in CGAL [20] for computing PVD, but our algorithm computes PVD directly in Euclidean space, which is an alternative simpler solution.

Observation. With the fast convergence of our algorithm, we observed that for most values of the number of sites n, a full-hexagonal pattern is obtained (see Figure 11 at the end of the paper). We evaluate several minimizers of the periodic CVT energy function and compare them with the theoretical lower bound [26] in Table I. For some specific values of n, no regular patterns were obtained. The theoretical analysis of this observation will be published in a further paper.

Table I
Periodic CVT Evaluation. F_{hex} is the theoretical lower bound $\frac{5}{18\sqrt{3}n}$ [26].

n	F_{CVT}	F_{PCVT}	F_{hex}
3	6.6180×10^{-2}	5.9671×10^{-2}	5.3458×10^{-2}
30	5.5454×10^{-3}	5.3482×10^{-3}	5.3453×10^{-3}
56	2.9520×10^{-3}	2.8639×10^{-3}	2.8638×10^{-3}
150	1.0919×10^{-3}	1.0696×10^{-3}	1.0692×10^{-3}
168	9.7361×10^{-4}	9.5464×10^{-4}	9.5461×10^{-4}
224	7.2769×10^{-4}	7.1598×10^{-3}	7.1596×10^{-4}
300	5.4255×10^{-4}	5.3481×10^{-4}	5.3458×10^{-4}

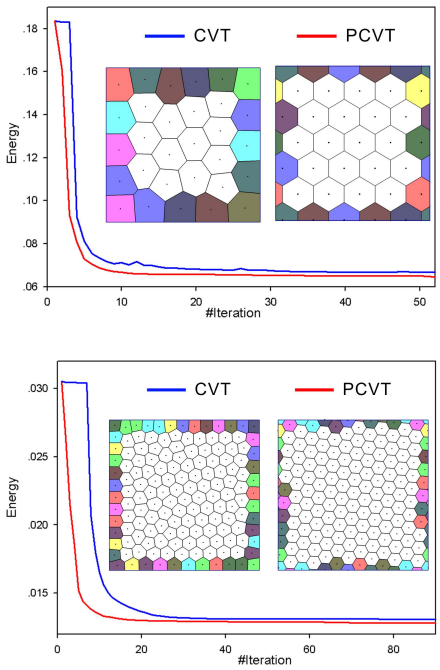


Figure 9. The CVT energy of 30 sites (top) and 150 sites (bottom).

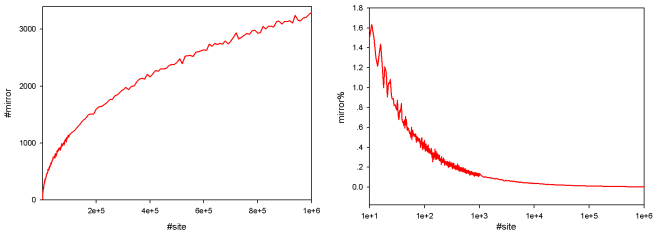


Figure 10. The number of mirror sites against number of sites (left), and the percentage of mirror sites (right).

VI. Future work

In the future, we would like to study the properties of the periodic CVT, which is also related to the problem on finding the global minimal of CVT energy function. We aim to find the connection with Gersho's conjecture and study the dependence on the number n. Another direction is the extension of our approach to 3D periodic space, and the development of practical algorithms for periodic surface/volume mesh generation.

Acknowledgments

This work is partially supported by the European Research Council (GOODSHAPE FP7-ERC-StG-205693) and ANR/NSFC (60625202,60911130368) Program (SHAN Project).

Appendix

In this appendix, we will prove the correctness of the presented PVD computation algorithm (Section III). The key point is to prove that our algorithm actually inserts all the mirror sites whose Voronoi cells intersect with the primary domain $\mathcal{D}_0$. Our proof is based on the following property of the Voronoi diagram.

Incremental property. *Assume that we have established a Euclidean Voronoi diagram of sites* $\{\mathbf{x}_i\}_{i=1}^k$*, whose cells are denoted as* $\{\Omega_i\}_{i=1}^k$*. We now insert a new site* $\mathbf{x}_{k+1}$ *and obtain an updated Voronoi diagram composed of cells* $\{\Omega_i^{'}\}_{i=1}^{k+1}$*. Then, it can be deduced that the updated Voronoi cells of the old sites* $\{\mathbf{x}_i\}_{i=1}^k$ *are subsets of their initial cells. We have the relation* $\Omega_i^{'} \subseteq \Omega_i$ *for* $i = 1..k$*.*

In order to make it easy to understand, we prefer to take a single boundary cell as example to proceed the proof. Suppose that a PVD is correctly constructed based on full copies of the primary sites and that $\mathbf{x}_1$ is a boundary site whose cell Ω_1 intersects with the domains $\mathcal{D}_0$, $\mathcal{D}_5$ and $\mathcal{D}_6$, as shown in Figure 12(a). This means that the cells of mirrors $\mathbf{x}_{1,1}$ and $\mathbf{x}_{1,2}$ intersect with $\mathcal{D}_0$ and that we need to insert these two copies. In our PVD algorithm, we first compute the Voronoi diagram of the primary sites, without any mirrors. Based on the above incremental property of the Voronoi diagram, we know that after this first step, the obtained cell of $\mathbf{x}_1$, denoted by $\Omega_1^{(1)}$ (cf. Figure 12(a)), is a superset of Ω_1. Therefore, $\Omega_1^{(1)}$ certainly intersects with e_3, and accordingly our algorithm inserts $\mathbf{x}_{1,2}$ in $\mathcal{D}_2$. Next, based on the same argument, we can deduce that in the updated diagram, the cell of $\mathbf{x}_{1,2}$, denoted by $\Omega_{1,2}^{(2)}$, certainly intersects with e_2, and $\mathbf{x}_{1,1}$ is thus inserted. Notice that $\mathbf{x}_{1,1}$ may also be inserted in the first step if $\Omega_1^{(1)}$ intersects with e_2.

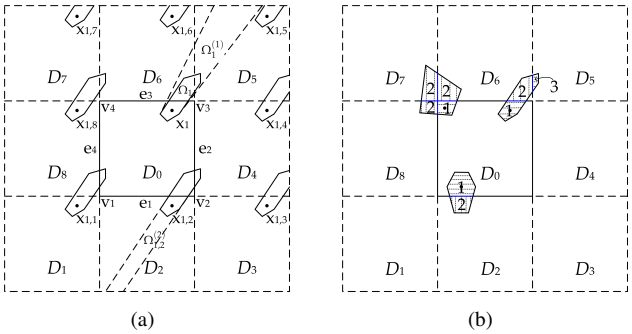


Figure 12. (a) Our algorithm can insert all the necessary mirrors of a boundary site for a correct PVD computation (for clarity we show only the cells of $\mathbf{x}_1$ and its mirror copies); (b) Some typical boundary cells (they are not necessarily in the same diagram) and the process of "polygon flooding".

Actually, one can easily prove that for any boundary cell (cf. Figure 12(b) for some typical examples), our algorithm can insert all the necessary mirror sites for computing a correct PVD, as explained in the following. As shown in

Figure 12(b), a boundary cell correctly computed with full copies is split by the domains $\{\mathcal{D}_j\}$ into several polygons. In the first step of our algorithm, we compute the intersection between boundary cell and $\mathcal{D}_0$, which is a superset of the polygon numbered by "1" in Figure 12(b). The algorithm then inserts at least the mirrors whose cells contain the primary domain images of all the incident polygons of the polygon "1", i.e. those numbered by "2" in Figure 12(b). The reason is that the cell-$\mathcal{D}_0$ intersection certainly contains the periodic boundaries shared by polygon "1" and its neighbors. The process continues until all the polygons are "flooded". This is guaranteed due to a) the incremental property of the Voronoi diagram, b) the guiding rules for the mirror sites insertion of our algorithm, and c) the fact that PVD cells in our case are convex and connected. □

REFERENCES

[1] G. Voronoi, "Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Deuxième mémoire: Recherches sur les parallelohèdres primitifs," *J. Reine Angew. Math.*, vol. 134, pp. 198–287, 1908.

[2] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: Applications and algorithms," *SIAM Review*, vol. 41, no. 4, pp. 637–676, 1999.

[3] Q. Du and D. Wang, "The optimal centroidal Voronoi tessellations and the Gersho's conjecture in the three-dimensional space," *Computers and Mathematics with Applications*, vol. 49, pp. 1355–1373, 2005.

[4] L. Decker and D. Jeulin, "Simulation 3D de matériaux aléatoires polycristallins," *La Revue de Métallurgie*, vol. 96, no. 2, pp. 271–275, 2000.

[5] M. Nygårds and P. Gudmundson, "Three-dimensional periodic Voronoi grain models and micromechanical FE-simulations of a two-phase steel," *Computational Materials Science*, vol. 24, pp. 513–519, 2002.

[6] F. Fritzen, T. Böhlke, and E. Schnack, "Periodic three-dimensional mesh generation for crystalline aggregates based on Voronoi tessellations," *Computational Mechanics*, vol. 43, no. 2, pp. 701–713, 2009.

[7] F. Fritzen and T. Böhlke, "Periodic three-dimensional mesh generation for particle reinforced composites with application to metal matrix composites," *International Journal of Solids and Structures*, vol. 48, no. 8, pp. 706–718, 2010.

[8] C. Börgers and C. S. Peskin, "A Lagrangian method based on the Voronoi diagram for the incompressible Navier Stokes equations on a periodic domain," *Journal of Computational Physics*, vol. 70, no. 2, pp. 397–438, 1987.

[9] K. E. Thompson, "Pore-scale modeling of fluid transport in disordered fibrous materials," *AIChE Journal*, vol. 48, no. 7, pp. 1369–1389, 2002.

[10] F. Aurenhammer, "Voronoi diagrams – a survey of a fundamental geometric data structure," *ACM Computing Surveys*, vol. 23, pp. 345–405, 1991.

[11] S. Fortune, "Voronoi diagrams and Delaunay triangulations," in *Computing in Euclidean Geometry*, 1992, pp. 193–233.

[12] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed. Wiley, 2000.

[13] G. Peyré and L. D. Cohen, "Geodesic methods for shape and surface processing," in *Advances in Computational Vision and Medical Image Processing: Methods and Applications*, 2009, pp. 29–56.

[14] D.-M. Yan, B. Lévy, Y. Liu, F. Sun, and W. Wang, "Isotropic remeshing with fast and exact computation of restricted Voronoi diagram," *Computer Graphics Forum (Proc. SGP)*, vol. 28, no. 5, pp. 1445–1454, 2009.

[15] B. Lévy and Y. Liu, "L_p centroidal Voronoi tessellation and its applications," *ACM Trans. on Graphics (Proc. SIGGRAPH)*, vol. 29, no. 4, p. Article No. 119, 2010.

[16] G. Rong, M. Jin, and X. Guo, "Hyperbolic centroidal Voronoi tessellation," in *Symposium of Solid & Physical Modeling (SPM)*, 2010, pp. 117–126.

[17] L. Ju, T. Ringler, and M. Gunzburger, "Voronoi diagrams and their application to climate and global modeling," in *Numerical Techniques for Global Atmospheric Models*, 2011.

[18] J. Tournois, P. Alliez, and O. Devillers, "2D centroidal Voronoi tessellations with constraints," *Numerical Mathematics: Theory, Methods and Applications*, vol. 3, no. 2, pp. 212–222, 2010.

[19] N. Fu, H. Imai, and S. Moriyama, "Voronoi diagrams on periodic graphs," in *Proc. of 7th International Symposium on Voronoi Diagrams in Science and Engineering (ISVD)*, 2010, pp. 189–198.

[20] M. Caroli and M. Teillaud, "Computing 3D periodic Triangulations," in *Proc. of 17th European Symposium on Algorithms (ESA)*, 2009, pp. 59–70.

[21] M. Iri, K. Murota, and T. Ohya, "A fast Voronoi diagram algorithm with applications to geographical optimization problems," in *Proc. of the 11th IFIP Conference on System Modelling and Optimization*, 1984, pp. 273–288.

[22] I. E. Sutherland and G. W. Hodgman, "Reentrant polygon clipping," *Communications of the ACM*, vol. 17, no. 1, pp. 32–42, 1974.

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

[24] Y. Liu, W. Wang, B. Lévy, F. Sun, D.-M. Yan, L. Lu, and C. Yang, "On centroidal Voronoi tessellation — energy smoothness and fast computation," *ACM Trans. on Graphics*, vol. 28, no. 4, p. Article No. 101, 2009.

[25] S. Lloyd, "Least square quantization in PCM," *IEEE Trans. Inform Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[26] D. J. Newman, "The hexagon theorem," *IEEE Trans. Inform Theory*, vol. 28, no. 2, pp. 137–139, 1982.

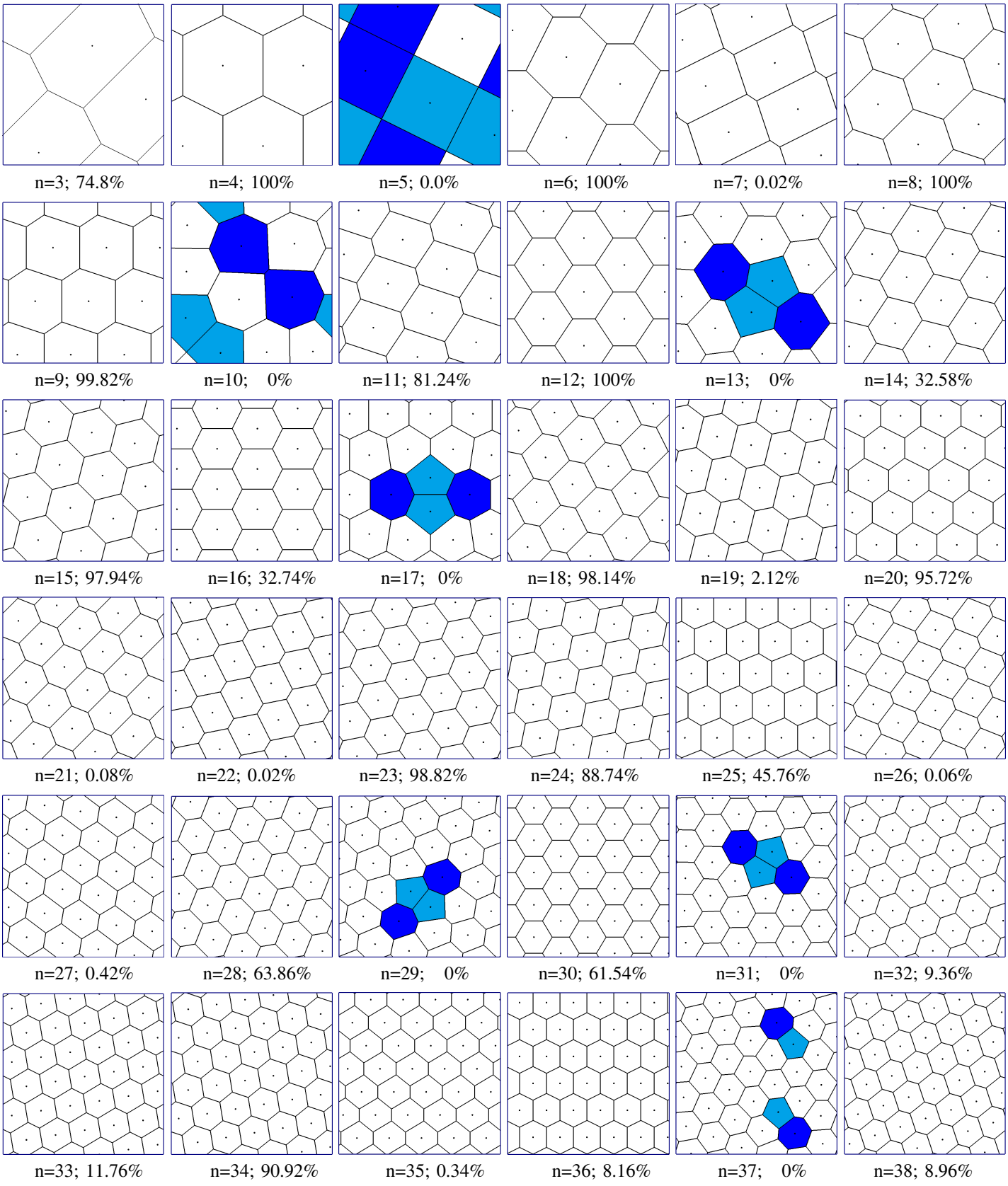


Figure 11. Hexagonal patterns of periodic CVT for different number of sites. The blue cells are non-hexagonal cells. For each number of n we perform 5000 runs. The percentage of the appearance of full-hexagonal patterns is also given.