Sensitivity analysis and optimization of centroidal Voronoi tessellations with geometric constraints*

Ernesto G. Birgin† Juan S. C. Franco‡ Antoine Laurain§

February 3, 2025¶

Abstract

In this paper, we investigate the optimization of Centroidal Voronoi Tessellations (CVT) under geometric constraints. For this purpose, we minimize a linear combination of the standard CVT energy functional with terms involving geometric attributes such as area and perimeter. The derivative of the objective functional with respect to the position of the generators is computed using techniques of shape calculus and sensitivity analysis of minimization diagrams. Several numerical experiments are presented to explore the geometric constraints of cells with identical areas, cells without small edges, and density-based distributions of cells.

Keywords: Centroidal Voronoi Tessellations, Voronoi diagrams, shape optimization, bound-constrained minimization, numerical experiments.

Mathematics Subject Classification: 49Q10, 49J52, 49Q12

1 Introduction

A Voronoi tessellation divides a region into cells based on a given set of generating points, known as sites. When each site is positioned at the centroid of the Voronoi cell, one obtains a Centroidal Voronoi Tessellations (CVTs). CVTs are widely used in various fields of science and engineering, including image processing, data compression, and numerical approximations of partial differential equations, particularly for mesh generation. We refer to [7] for a comprehensive survey of applications. CVT has been generalized to very broad settings, such as CVT of surfaces or line segments, distance metrics, and discrete point sets [8].

*This work has been partially supported by the Brazilian agencies FAPESP (grants 2013/07375-0, 2022/05803-3, and 2023/08706-1) and CNPq (grant 302073/2022-1).

†Department of Computer Science, Institute of Mathematics and Statistics, University of São Paulo, Rua do Matão, 1010, Cidade Universitária, 05508-090, São Paulo, SP, Brazil. e-mail: egbirgin@ime.usp.br

‡Department of Applied Mathematics, Institute of Mathematics and Statistics, University of São Paulo, Rua do Matão, 1010, Cidade Universitária, 05508-090, São Paulo, SP, Brazil. e-mail: jcastanof@ime.usp.br

§Faculty of Mathematics, University of Duisburg-Essen, Thea-Leymann-Str. 9, 45127, Essen, Germany, e-mail: antoine.laurain@uni-due.de

¶Revised on August 30, 2025.

A standard approach to computing CVTs is the deterministic Lloyd's algorithm [7, 14], which is a fixed-point algorithm. The convergence of Lloyd's algorithm is slow, but several improvements have been introduced, such as the Lloyd-Newton method [6] and variants that employ quasi-Newton methods, such as the limited-memory BFGS (LBFGS) method [10, 13]. CVTs can also be constructed using probabilistic methods such as MacQueen's algorithm [15].

In [13], it was shown that a CVT can be seen as a critical point of a certain energy function, with C^2 regularity in convex domains. In this way, derivative-based optimization methods can be used to compute CVTs. The derivative of the energy and the Lloyd map were computed in [7] and used in subsequent papers such as [4, 13]. The use of these derivatives for gradient-based, Newton, and quasi-Newton methods is also discussed in [7].

It was shown in [3] that this type of sensitivity analysis for Voronoi diagrams can be recast in the much more general framework of minimization diagrams. Minimization diagrams are a broad class of diagrams whose cells are defined via the lower envelope of a set of graphs of functions. Many relevant diagrams, such as Voronoi or power diagrams, are special cases of minimization diagrams. These theoretical results allow us to consider the optimization of any differentiable criterion depending on the geometric properties of the minimization diagram. This includes CVT as a special case, but also various other criteria such as perimeter, area, angles, and other geometric quantities. A useful feature of this general theory is the handling of fixed elements of the geometry, such as the boundary of the domain, which are usually not considered in the sensitivity analysis of Voronoi diagrams. In [3], the authors explored various numerical applications for optimizing Voronoi diagrams, with the aim of obtaining specific properties such as equal cell areas or edges of equal length. These properties are desirable for improving mesh quality using quality measures, see [7, 11].

In this paper, we extend these results to the optimization of CVTs. We consider objective functions that are linear combinations of the CVT energy with additional terms that enforce geometric constraints. The goal is to demonstrate how one can control the geometric properties of CVTs. This can also be seen as minimizing the CVT energy with additional constraints in the form of penalizations. We focus on two types of geometric constraints. The first constraint is of the area type, forcing cells to have identical areas. The second type is a perimeter constraint, where the goal is to avoid cells with small edges.

Section 2 establishes the notation and essential tools for the sensitivity analysis of Voronoi diagrams. These results are then applied to compute the gradient of the CVT and other related shape functions. We present two distinct approaches for the computation of gradients. Numerical experiments are provided in Section 3. In the experiments, we show how to compute CVTs where all cells have the same area, cells without edges considered small, and cells whose area is governed by a given function. The final section presents conclusions and lines of future work.

2 The CVT energy function and its gradient

In this section, we first recall the main results of [3] that are necessary for the sensitivity analysis of Voronoi diagrams. We then introduce the CVT energy function and use these results to compute its gradient. Consider the set $A := \{x \in \mathbb{R}^2 : \varphi(x) < 0\}$ with $\varphi(x) := \min_{\ell \in \mathcal{K}_A} \varphi_\ell(x)$ and $\varphi_\ell \in \mathcal{C}^\infty(\mathbb{R}^2, \mathbb{R})$ for all $\ell \in \mathcal{K}_A := \{\kappa_0 + 1, \ldots, \kappa_0 + \kappa_1\}$ for some given κ_0, κ_1. For $\ell \in \mathcal{K}_A$,

introduce the set $\partial_\ell A := \{x \in \partial A : \ \varphi_\ell(x) = 0\}$; then we have $\partial A = \cup_{\ell \in \mathcal{K}_A} \partial_\ell A$. Note that the notation $\partial_\ell A$ refers to a subset of the boundary of the set A, whereas the similar-looking notation $\partial_t T$ represents further the partial derivative of the function T with respect to t. Denote by $\mathcal{T}_{\partial A}$ the finite set of corners of A. Let $\mathcal{K}_{\text{vor}} = \{1, \dots, \kappa_0\}$ be a set of indices, $\mathbf{a} = \{a_k\}_{k \in \mathcal{K}_{\text{vor}}}$ be a set of points in the plane, the so-called *sites*, and let $\boldsymbol{V}(\mathbf{a}) := \{V_i(\mathbf{a})\}_{i \in \mathcal{K}_{\text{vor}}}$ be the Voronoi diagram associated with $\mathbf{a}$, where the cells of the diagram are defined by

$$V_i(\mathbf{a}) := \{x \in A \text{ such that } \|x - a_i\| \leq \|x - a_j\| \text{ for all } j \in \mathcal{K}_{\text{vor}} \setminus \{i\}\}.$$

We will also need the following notation. The Euclidean norm is denoted by $\|\cdot\|$. For $x, y \in \mathbb{R}^n$, $x \cdot y = x^\top y \in \mathbb{R}$; $x \otimes y = xy^\top \in \mathbb{R}^{n \times n}$. For a counterclockwise, $\pi/2$-rotation matrix R, define $x^\perp := Rx$ for $x \in \mathbb{R}^2$. The transpose of a matrix M is denoted by $M^\top$. The gradient of a function $\psi : \mathbb{R}^2 \to \mathbb{R}$ with respect to $x \in \mathbb{R}^2$ is represented as $\nabla_x \psi$ and is expressed as a column vector. The Jacobian matrix of a function $\varphi : \mathbb{R}^2 \to \mathbb{R}^2$ is denoted by $D_x \varphi$. For a scalar-valued function F, its gradient with respect to $\mathbf{a}$ is written as $\nabla_{\mathbf{a}} F$ or simply ∇F if F only depends on $\mathbf{a}$. For a vector-valued function F, DF denotes its Jacobian matrix with respect to $\mathbf{a}$. For a sufficiently smooth set $S \subset \mathbb{R}^2$, its closure is written as $\overline{S}$, $|S|$ represents either its perimeter if S is one-dimensional, or its area if S is two-dimensional.

2.1 Sensitivity analysis of Voronoi diagrams

Our general objective is to compute the derivative of functions $\mathbf{a} \mapsto G(\mathbf{a})$ depending on the shape of the Voronoi diagram $\boldsymbol{V}(\mathbf{a})$. To this end, we introduce a perturbation of the sites $\mathbf{a} + t\delta\mathbf{a}$, where $\delta\mathbf{a} := \{\delta a_k\}_{k \in \mathcal{K}_{\text{vor}}}$ is a prescribed perturbation direction, and $t \in \mathbb{R}$ is a small scalar parameter. The corresponding directional derivative of G in direction $\delta\mathbf{a}$ is then defined as

$$\nabla G(\mathbf{a}) \cdot \delta\mathbf{a} = \lim_{t \to 0} \frac{G(\mathbf{a} + t\delta\mathbf{a}) - G(\mathbf{a})}{t}.$$

Our numerical algorithm is based on the gradient $\nabla G(\mathbf{a})$; see Section 3. This perturbation of the sites induces a perturbed Voronoi diagram, denoted $\boldsymbol{V}(\mathbf{a} + t\delta\mathbf{a}) := \{V_k(\mathbf{a} + t\delta\mathbf{a})\}_{k \in \mathcal{K}_{\text{vor}}}$. In [3], a theoretical framework for analyzing the sensitivity of minimization diagrams was developed, with the specific case of Voronoi diagrams examined in [3, §4]. The key idea of this framework is to construct a mapping $T(\cdot, t)$ that transports each reference cell $V_k(\mathbf{a})$ to its perturbed counterpart $V_k(\mathbf{a} + t\delta\mathbf{a})$; see Theorem 2.3. The derivative $\theta := \partial_t T(\cdot, 0)$ then plays an essential role in computing the gradient $\nabla G(\mathbf{a})$.

We now recall the results of [3], starting with some tools and notation.

Definition 2.1 (interior vertices and edges)**.** *For $\{i, j, k\} \subset \mathcal{K}_{vor}$, we define the set of inner vertices $Y_{ijk}(t) := \overline{V_i(\mathbf{a} + t\delta\mathbf{a})} \cap \overline{V_j(\mathbf{a} + t\delta\mathbf{a})} \cap \overline{V_k(\mathbf{a} + t\delta\mathbf{a})}$, and write $Y_{ijk} := Y_{ijk}(0)$. For $k \in \mathcal{K}_{vor} \setminus \{i\}$, interior edges of $\boldsymbol{V}(\mathbf{a} + t\delta\mathbf{a})$ are denoted by $E_{ik}(\mathbf{a} + t\delta\mathbf{a}) := \overline{V_i(\mathbf{a} + t\delta\mathbf{a})} \cap \overline{V_k(\mathbf{a} + t\delta\mathbf{a})}$.*

Definition 2.2 (boundary vertices and edges)**.** *For $\{i, j\} \subset \mathcal{K}_{vor}$ and $\ell \in \mathcal{K}_A$, we define the set of boundary vertices $X_{ij\ell}(t) := \overline{V_i(\mathbf{a} + t\delta\mathbf{a})} \cap \overline{V_j(\mathbf{a} + t\delta\mathbf{a})} \cap \partial_\ell A$, and write $X_{ij\ell} := X_{ij\ell}(0)$. For $k \in \mathcal{K}_{vor} \setminus \{i\}$ and $\ell \in \mathcal{K}_A$, boundary edges of $\boldsymbol{V}(\mathbf{a} + t\delta\mathbf{a})$ are denoted by $E_{ik}(\mathbf{a} + t\delta\mathbf{a}) := \overline{V_i(\mathbf{a} + t\delta\mathbf{a})} \cap \partial_\ell A$.*

A crucial aspect of the theory developed in [3] is to establish a set of geometric assumptions that prevent degenerate cases and ensure the feasibility of the sensitivity analysis. These assumptions guarantee that each interior vertex Y_{ijk} of the Voronoi diagram is shared by at most three cells. Furthermore, they prevent trivial situations in which two distinct-indexed cells are identical. These assumptions were formulated in the more general context of minimization diagrams, so here we provide a simpler formulation in the case of Voronoi diagrams, which summarizes the discussion in [3, §4].

Assumption 1. *Suppose that:*

- *(Non-degeneracy of interfaces) There holds* $\|\nabla_x \varphi_\ell(x)\| > 0$ *for all* $x \in \partial_\ell A$ *and for all* $\ell \in \mathcal{K}_A$*, and* $\|a_i - a_j\| > 0$ *for all* $\{i,j\} \subset \mathcal{K}_{vor}$*.*
- *(Non-degeneracy of vertices) For all* $\{i,j,k\} \subset \mathcal{K}_{vor}$ *such that* $Y_{ijk} \neq \emptyset$ *we have* $(a_j - a_i)^\perp \cdot (a_k - a_i) \neq 0$ *and* $Y_{ijk} \cap \overline{V_m(\mathbf{a})} = \emptyset$*, for all* $m \in \mathcal{K}_{vor} \setminus \{i,j,k\}$*. In addition, for all* $\{i,j\} \subset \mathcal{K}_{vor}$ *and* $\ell \in \mathcal{K}_A$ *and all* $v \in X_{ij\ell}$ *we have* $(a_j - a_i)^\perp \cdot \nabla \varphi_\ell(v) \neq 0$*,* $v \notin \overline{V_m(\mathbf{a})}$*, for all* $m \in \mathcal{K}_{vor} \setminus \{i,j\}$*, and* $v \notin \mathcal{T}_{\partial A}$*, where* $\mathcal{T}_{\partial A}$ *is the finite set of corners of* A*.*

Under these geometric assumptions, the set Y_{ijk} contains at most one point, but $X_{ij\ell}$ may contain multiple points. The key to the sensitivity analysis of the perturbed Voronoi diagram $\boldsymbol{V}(\mathbf{a} + t\delta\mathbf{a})$ is the computation of the derivatives of the vertices $Y_{ijk}(t)$ and $X_{ijk}(t)$. This is essentially an application of the implicit function theorem under Assumption 1.

Theorem 2.1 ([3, Theorem 7])**.** *Suppose Assumption 1 holds and* $|Y_{ijk}| = 1$ *for some* $\{i,j,k\} \subset \mathcal{K}_{vor}$*. Then, denoting* $v = Y_{ijk}$*, there exists* $\tau_1 > 0$ *and a unique smooth function* $z_v : [0,\tau_1] \to \mathbb{R}^2$ *satisfying* $z_v(0) = v$*,* $z_v(t) = Y_{ijk}(t)$ *for all* $t \in [0,\tau_1]$ *and*

$$z_v'(0) = M_v(j,k,i)\delta a_i + M_v(k,i,j)\delta a_j + M_v(i,j,k)\delta a_k, \tag{1}$$

where

$$M_v(i,j,k) \coloneqq \frac{(a_i - a_j)^\perp \otimes (v - a_k)^\top}{Q(i,j,k)}$$

and

$$Q(i,j,k) \coloneqq \det \begin{pmatrix} (a_j - a_i)^\top \\ (a_k - a_i)^\top \end{pmatrix}.$$

Theorem 2.2 ([3, Theorem 8])**.** *Suppose Assumption 1 holds and let* $\{i,j\} \subset \mathcal{K}_{vor}$*,* $\ell \in \mathcal{K}_A$*. Then* $X_{ij\ell}$ *is finite,* $X_{ij\ell} \in \partial A \setminus \mathcal{T}_{\partial A}$ *, and there exists* $\tau_1 > 0$ *such that for each* $v \in X_{ij\ell}$ *there exists a unique smooth function* $z_v : [0,\tau_1] \to \mathbb{R}^2$ *satisfying* $z(0) = v$*,* $\varphi_\ell(z_v(t)) = 0$ *for all* $t \in [0,\tau_1]$*, and*

$$X_{ij\ell}(t) = \bigcup_{v \in X_{ij\ell}} \{z_v(t)\} \text{ for all } t \in [0,\tau_1].$$

In addition we have

$$z_v'(0) = \mathscr{M}_v^\ell(j,i)\delta a_i + \mathscr{M}_v^\ell(i,j)\delta a_j \tag{2}$$

with

$$\mathscr{M}_v^\ell(j,i) \coloneqq \frac{-\nabla_x \varphi_\ell(v)^\perp \otimes (v - a_i)^\top}{\det \begin{pmatrix} (a_j - a_i)^\top \\ \nabla_x \varphi_\ell(v)^\top \end{pmatrix}}.$$

Under Assumption 1, the motion of each Voronoi cell $V_i(\mathbf{a}+t\delta\mathbf{a})$, $i \in \mathcal{K}_{\rm vor}$, can be parameterized by a bi-Lipschitz mapping $T(\cdot,t)$, such that its derivative $\theta := \partial_t T(\cdot,0)$ can be described explicitly as a function of the sites $\mathbf{a}$. The explicit expression of θ at the vertices is a consequence of Theorems 2.1 and 2.2. This parameterization is described in the following theorem, which is a particular case of [3, Theorem 5].

Theorem 2.3. *Let $i \in \mathcal{K}_{vor}$ and suppose Assumption 1 holds. Then there exist $t_0 > 0$ and $T : \overline{V_i(\mathbf{a})} \times [0,t_0] \to \mathbb{R}^2$ such that $T(V_i(\mathbf{a}),t) = V_i(\mathbf{a}+t\delta\mathbf{a})$, $T(E_{ik}(\mathbf{a}),t) = E_{ik}(\mathbf{a}+t\delta\mathbf{a})$ for all $k \in \mathcal{K}_{vor}\setminus\{i\}$, $T(E_{i\ell}(\mathbf{a}),t) = E_{i\ell}(\mathbf{a}+t\delta\mathbf{a})$ for all $\ell \in \mathcal{K}_A$, $T(\partial V_i(\mathbf{a}),t) = \partial V_i(\mathbf{a}+t\delta\mathbf{a})$ and $T(\cdot,t) : \overline{V_i(\mathbf{a})} \to \overline{V_i(\mathbf{a}+t\delta\mathbf{a})}$ is bi-Lipschitz for all $t \in [0,t_0]$. Let $\theta := \partial_t T(\cdot,0)$, we also have*

$$\begin{aligned}
\theta(x)\cdot\nu(x) &= \frac{\nabla_a\phi(x,a_k)\cdot\delta a_k - \nabla_a\phi(x,a_i)\cdot\delta a_i}{\|\nabla_x\phi(x,a_k)-\nabla_x\phi(x,a_i)\|} \text{ for all } x\in E_{ik}(\mathbf{a}) \text{ and all } k\in\mathcal{K}_{vor}\setminus\{i\},\\
\theta(x)\cdot\nu(x) &= 0 \text{ for all } x\in E_{i\ell}(\mathbf{a}) \text{ and for all } \ell\in\mathcal{K}_A,\\
\theta(v)\cdot\tau(v) &= (M_v(j,k,i)\delta a_i + M_v(k,i,j)\delta a_j + M_v(i,j,k)\delta a_k)\cdot\tau(v)\\
&\quad \text{for all } v\in Y_{ijk}, \{i,j,k\}\subset\mathcal{K}_{vor},\\
\theta(v)\cdot\tau(v) &= (\mathscr{M}_v^\ell(j,i)\delta a_i + \mathscr{M}_v^\ell(i,j)\delta a_j)\cdot\tau(v) \text{ for all } v\in X_{ij\ell}, \{i,j\}\subset\mathcal{K}_{vor}, \ell\in\mathcal{K}_A,
\end{aligned} \tag{3}$$

where ν is the outward unit normal vector to $V_i(\mathbf{a})$, $\phi(x,a) := \|x-a\|^2$, and τ is any tangent vector to $\partial V_i(\mathbf{a})$.

Theorems 2.1, 2.2, and 2.3 permit the computation of various relevant geometric quantities related to Voronoi diagrams. The derivative of area and edge integrals is of particular interest and has been performed in [3, Sections 3 and 4]; here we give a brief summary of these results. Starting with the case of an area integral, consider

$$G_1(\mathbf{a}+t\delta\mathbf{a}) \coloneqq \int_{V_i(\mathbf{a}+t\delta\mathbf{a})} f(x)\,dx, \tag{4}$$

where $f \in C^1(\overline{A},\mathbb{R})$. Then, using Theorem 2.3, we have $V_i(\mathbf{a}+t\delta\mathbf{a}) = T(V_i(\mathbf{a}),t)$ and we can apply a change of variable to transform the integral in (4) to an integral on $V_i(\mathbf{a})$. Next we can differentiate the obtained expression with respect to t, use the divergence theorem to transform the integral on $V_i(\mathbf{a})$ to an integral on $\partial V_i(\mathbf{a})$, and use the first equality in (3) to obtain

$$\begin{aligned}
\nabla G_1(\mathbf{a})\cdot\delta\mathbf{a} = &\sum_{E\in\mathcal{E}_i^{\rm int}} \frac{\delta a_i}{\|a_i - a_{k(i,E)}\|}\cdot\int_E f(x)(x-a_i)\,dx\\
&- \sum_{E\in\mathcal{E}_i^{\rm int}} \frac{\delta a_{k(i,E)}}{\|a_i - a_{k(i,E)}\|}\cdot\int_E f(x)(x-a_{k(i,E)})\,dx.
\end{aligned} \tag{5}$$

where $\mathcal{E}_i^{\rm int}$ is the set of the interior edges of the cell $V_i(\mathbf{a})$.

Next, consider the edge integral function defined by

$$G_2(\mathbf{a}+t\delta\mathbf{a}) \coloneqq \int_{E(\mathbf{a}+t\delta\mathbf{a})} f(x)\,dx, \tag{6}$$

where $f \in C^1(\overline{A}, \mathbb{R})$. In (6), the edge $E(\mathbf{a})$ can be either a boundary edge, given by $E(\mathbf{a}) = \overline{V_i(\mathbf{a})} \cap \partial_\ell A$ for $\ell \in \mathcal{K}_A$, or an interior edge, defined as $E(\mathbf{a}) = \overline{V_i(\mathbf{a})} \cap \overline{V_k(\mathbf{a})}$ for $\{i, k\} \subset \mathcal{K}_{\text{vor}}$. In a similar way as for the area integral, using Theorem 2.3 we obtain the following expression for the gradient:

$$\nabla G_2(\mathbf{a}) \cdot \delta\mathbf{a} = \mathcal{F}(i, w_E) \cdot \tau(w_E) - \mathcal{F}(i, v_E) \cdot \tau(v_E), \tag{7}$$

where v_E, w_E are the start and end points of $E(\mathbf{a})$ determined by a counterclockwise orientation on $\partial V_i(\mathbf{a})$, $\tau = \frac{w_E - v_E}{\|w_E - v_E\|}$ is a unit tangent vector to $\partial V_i(\mathbf{a})$, and

$$\mathcal{F}(i, v) = \begin{cases} M_v(j,k,i)\delta a_i + M_v(k,i,j)\delta a_j + M_v(i,j,k)\delta a_k, & \text{if } v \in Y_{ijk}, \\ \mathscr{M}_v^\ell(j,i)\delta a_i + \mathscr{M}_v^\ell(i,j)\delta a_j, & \text{if } v \in X_{ij\ell}, \\ 0, & \text{if } v \in \mathcal{T}_{\partial A}. \end{cases} \tag{8}$$

In (8), the indices j, k in Y_{ijk} and j in $X_{ij\ell}$ depend on i and on the vertex v. These indices can be uniquely determined by adopting a counterclockwise orientation for the cells surrounding v.

2.2 CVT energy function

Let $\rho : A \to \mathbb{R}$, $\rho > 0$, be a given density function. The mass centroid of a Voronoi cell $V_i(\mathbf{a})$ is defined by

$$c_i := \frac{\displaystyle\int_{V_i(\mathbf{a})} x\,\rho(x)\,dx}{\displaystyle\int_{V_i(\mathbf{a})} \rho(x)\,dx}.$$

In general, we have $c_i \neq a_i$. When $c_i = a_i$ for all $i = 1, \ldots, \kappa_0$, $\boldsymbol{V}(\mathbf{a})$ is called a Centroidal Voronoi Tessellations (CVT). CVTs enjoy several desirable properties that make them attractive for applications. The resulting cells are typically well-shaped, naturally adapted to the density ρ, and well-suited for the generation of high-quality meshes in finite element and finite volume methods. A natural way to characterize CVTs is through the CVT energy functional [4, 7, 13], defined as

$$G(\mathbf{a}) \coloneqq \frac{1}{\kappa_0} \sum_{i=1}^{\kappa_0} G_i(\mathbf{a}) \text{ with } G_i(\mathbf{a}) = \int_{V_i(\mathbf{a})} \rho(x)\|x - a_i\|^2\,dx, \tag{9}$$

recalling that $\mathbf{a} = \{a_i\}_{i=1}^{\kappa_0}$ are the sites generating the cells $V_i(\mathbf{a})$. In [7, 13] it is shown that the critical points of the function $G(\mathbf{a})$ generate a CVT. Despite these advantages, several challenges remain. Cell shapes may deteriorate for certain density functions or near the boundary of irregular domains, making additional geometric control necessary. Moreover, $G(\mathbf{a})$ is highly non-convex: it may possess many local minima, and the final tessellation can be sensitive to initialization, sometimes yielding distorted configurations. In this work, we address these difficulties by incorporating additional geometric constraints to the CVT energy functional, designed to improve robustness and enforce better cell regularity.

The calculation of the gradient of $G(\mathbf{a})$ is standard and has been performed, for example, in [7, 13]. Here, in Sections 2.3 and 2.4 we present the calculation of the gradient of $G(\mathbf{a})$ using our framework, which will also allow us to supplement $G(\mathbf{a})$ with other geometric terms

in the numerical experiments. We present two ways to calculate the gradient of $G(\mathbf{a})$. First, we compute $\nabla G(\mathbf{a})$ directly using (5), and second, we provide first an explicit formula for $G_i(\mathbf{a})$ and $G(\mathbf{a})$ as a function of the vertices and a_i, and compute the gradient of that formula. We then discuss the differences between these two formulas for applications.

For numerical purposes, (9) needs to be evaluated. In the general case, quadrature rules need to be employed. In some particular cases, an explicit expression of $G(\mathbf{a})$ can be obtained. Let $T(a_i, v, w)$ be the triangle with vertices a_i, v, w, where w is the neighbor vertex of v in counterclockwise direction, and define $\mathcal{V}_i$ as the set of vertices of $V_i(\mathbf{a})$. Then we have the partition $V_i(\mathbf{a}) = \bigcup_{v\in\mathcal{V}_i} T(a_i, v, w)$. Thus

$$G_i(\mathbf{a}) = \int_{V_i(\mathbf{a})} \rho(x)\|x - a_i\|^2\, dx = \sum_{v\in\mathcal{V}_i} \int_{T(a_i,v,w)} \rho(x)\|x - a_i\|^2\, dx.$$

Now, we consider the transformation $\phi : \mathbb{R}^2 \to \mathbb{R}^2$ defined as follows,

$$\phi(\xi, \lambda) = a_i + \xi(v - a_i) + \lambda(w - a_i) \text{ with } \xi, \lambda \geq 0 \text{ and } \xi + \lambda = 1.$$

Using a change of variables, we have

$$\int_{T(a_i,v,w)} f(x)\, dx = \int_{\xi=0}^{1} \int_{\lambda=0}^{1-\xi} f(\phi(\xi, \lambda))|J(v, w, a_i)|\, d\lambda\, d\xi,$$

with the Jacobian matrix $J(v, w, a_i) = D\phi = (v - a_i|w - a_i) \in \mathbb{R}^{2\times 2}$ and

$$\begin{aligned} |J(v, w, a_i)| &= \begin{vmatrix} v_1 - a_{i,1} & w_1 - a_{i,1} \\ v_2 - a_{i,2} & w_2 - a_{i,2} \end{vmatrix} \\ &= (v_1 - a_{i,1})(w_2 - a_{i,2}) - (v_2 - a_{i,2})(w_1 - a_{i,1}) \\ &= -(v - a_i) \cdot (w - a_i)^{\perp}. \end{aligned} \tag{10}$$

For $f(x) = \rho(x)\|x - a_i\|^2$, we get

$$\begin{aligned} &\int_0^1 \int_0^{1-\xi} \rho(\phi(\xi, \lambda))\|\phi(\xi, \lambda) - a_i\|^2 |J(v, w, a_i)|\, d\lambda d\xi \\ &= |J(v, w, a_i)| \int_0^1 \int_0^{1-\xi} \rho(\phi(\xi, \lambda))\|\xi(v - a_i) + \lambda(w - a_i)\|^2\, d\lambda d\xi \\ &= |J(v, w, a_i)| \int_0^1 \int_0^{1-\xi} \rho(\phi(\xi, \lambda))(\xi^2\|v - a_i\|^2 + \lambda^2\|w - a_i\|^2 + 2\lambda\xi(v - a_i) \cdot (w - a_i))\, d\lambda d\xi. \end{aligned}$$

To simplify this expression further, we need to choose a specific class of functions ρ such as polynomials. Let us consider the particular case $\rho \equiv 1$, which results in

$$\int_0^1 \int_0^{1-\xi} \|\phi(\xi, \lambda) - a_i\|^2 |J(v, w, a_i)|\, d\lambda d\xi = \frac{|J(v, w, a_i)|}{12}(\|v - a_i\|^2 + (v - a_i) \cdot (w - a_i) + \|w - a_i\|^2).$$

Finally,

$$G_i(\mathbf{a}) = \sum_{v \in \mathcal{V}_i} \frac{|J(v,w,a_i)|}{12}(\|v-a_i\|^2 + (v-a_i)\cdot(w-a_i) + \|w-a_i\|^2),$$

and this yields the following explicit formula of $G(\mathbf{a})$ in the case $\rho \equiv 1$:

$$G(\mathbf{a}) = \frac{1}{\kappa_0}\sum_{i=1}^{\kappa_0}\left(\sum_{v \in \mathcal{V}_i} \frac{|J(v,w,a_i)|}{12}(\|v-a_i\|^2 + (v-a_i)\cdot(w-a_i) + \|w-a_i\|^2)\right). \tag{11}$$

2.3 Computing the gradient of the integral form of $G(\mathbf{a})$

Here, we compute the value of the gradient of the integral form (9) of $G(\mathbf{a})$ using (5). First, we have

$$\nabla G(\mathbf{a}) = \nabla\left(\frac{1}{\kappa_0}\sum_{i=1}^{\kappa_0} G_i(\mathbf{a})\right) = \frac{1}{\kappa_0}\sum_{i=1}^{\kappa_0}\nabla G_i(\mathbf{a}).$$

Recall that $\mathcal{E}_i^{\text{int}}$ is the set of the interior edges of the cell $V_i(\mathbf{a})$ and let $k(i,E)$ be the index such that $E = \overline{V_i(\mathbf{a})} \cap \overline{V_{k(i,E)}(\mathbf{a})}$. Applying (5), we get

$$\begin{aligned}
\nabla G_i(\mathbf{a})\cdot\delta\mathbf{a} &= \nabla\left(\int_{V_i(\mathbf{a})}\rho(x)\|x-a_i\|^2\,dx\right)\cdot\delta\mathbf{a} \\
&= \sum_{E\in\mathcal{E}_i^{\text{int}}}\frac{\delta a_i}{\|a_i - a_{k(i,E)}\|}\cdot\underbrace{\int_E \rho(x)\|x-a_i\|^2(x-a_i)\,dx}_{=:I_E^1} \\
&\quad - \sum_{E\in\mathcal{E}_i^{\text{int}}}\frac{\delta a_{k(i,E)}}{\|a_i - a_{k(i,E)}\|}\cdot\underbrace{\int_E \rho(x)\|x-a_i\|^2(x-a_{k(i,E)})\,dx}_{=:I_E^2} \\
&\quad - 2\delta a_i\cdot\int_{V_i(\mathbf{a})}\rho(x)(x-a_i)\,dx.
\end{aligned}$$

Note that the last term is not present in (5) but appears as the integrand $\rho(x)\|x-a_i\|^2$ depends on a_i.

As $c_i \coloneqq \dfrac{\int_{V_i(\mathbf{a})}\rho(x)x\,dx}{\int_{V_i(\mathbf{a})}\rho(x)\,dx}$ is the centroid of the cell $V_i(\mathbf{a})$, we obtain

$$\nabla G_i(\mathbf{a})\cdot\delta\mathbf{a} = 2\delta a_i\cdot(a_i-c_i)\int_{V_i(\mathbf{a})}\rho(x)\,dx + \sum_{E\in\mathcal{E}_i^{\text{int}}}\frac{\delta a_i\cdot I_E^1 - \delta a_{k(i,E)}\cdot I_E^2}{\|a_i - a_{k(i,E)}\|}.$$

This yields

$$\nabla G(\mathbf{a})\cdot\delta\mathbf{a} = \frac{1}{\kappa_0}\sum_{i=1}^{\kappa_0}\sum_{E\in\mathcal{E}_i^{\text{int}}}\frac{\delta a_i\cdot I_E^1 - \delta a_{k(i,E)}\cdot I_E^2}{\|a_i - a_{k(i,E)}\|} + 2\delta a_i\cdot(a_i-c_i)\int_{V_i(\mathbf{a})}\rho(x)\,dx.$$

Now we rearrange the term

$$\sum_{i=1}^{\kappa_0} \sum_{E \in \mathcal{E}_i^{\text{int}}} \frac{\delta a_{k(i,E)} \cdot I_E^2}{\|a_i - a_{k(i,E)}\|}$$

by summing over all fixed indices k such that $k = k(i, E)$ for some index i and edge E. Then a_i becomes $a_{i(k,E)}$ which yields

$$\sum_{i=1}^{\kappa_0} \sum_{E \in \mathcal{E}_i^{\text{int}}} \frac{\delta a_{k(i,E)} \cdot I_E^2}{\|a_i - a_{k(i,E)}\|} = \sum_{k=1}^{\kappa_0} \sum_{E \in \mathcal{E}_k^{\text{int}}} \frac{\delta a_k \cdot \int_E \rho(x) \|x - a_{i(k,E)}\|^2 (x - a_k)\, dx}{\|a_{i(k,E)} - a_k\|}.$$

Using the property $\|x - a_{i(k,E)}\| = \|x - a_k\|$ on E and changing the notation for indices, we end up with

$$\sum_{i=1}^{\kappa_0} \sum_{E \in \mathcal{E}_i^{\text{int}}} \frac{\delta a_{k(i,E)} \cdot I_E^2}{\|a_i - a_{k(i,E)}\|} = \sum_{i=1}^{\kappa_0} \sum_{E \in \mathcal{E}_i^{\text{int}}} \frac{\delta a_i \cdot I_E^1}{\|a_i - a_{k(i,E)}\|}.$$

Thus, the terms depending on I_E^1 and I_E^2 cancel out in $\nabla G(\mathbf{a}) \cdot \delta \mathbf{a}$ and

$$\nabla G(\mathbf{a}) \cdot \delta \mathbf{a} = \frac{1}{\kappa_0} \sum_{i=1}^{\kappa_0} 2\delta a_i \cdot (a_i - c_i) \int_{V_i(\mathbf{a})} \rho(x)\, dx, \tag{12}$$

which is the standard formula in the literature, see for instance [4, 7, 12]. The fact that I_E^1 and I_E^2 cancel out is also observed in [12, p. 278] for instance.

2.4 Explicit gradient computation in the case $\rho \equiv 1$

To obtain the expression (12) of $\nabla G(\mathbf{a})$, we have used (5), which is based on Theorem 2.3, using the shape calculus techniques of [3]. In the special case where $\rho \equiv 1$, formula (11) for $G(\mathbf{a})$ is obtained, and $\nabla G(\mathbf{a})$ can be computed in a simpler way, without using Theorem 2.3, by directly differentiating (11). The case $\rho \equiv 1$ already covers a wide range of applications, as demonstrated by our numerical experiments, see Section 3. The purpose of this section is to perform this calculation and discuss the resulting formula. Note that for certain classes of functions ρ, such as polynomials, one can similarly derive an explicit formula analogous to (11) and differentiate it to obtain $\nabla G(\mathbf{a})$.

Let $\mathcal{E}_i$ be the set of edges of $V_i(\mathbf{a})$. In (11) the sum is over $\mathcal{V}_i$, the set of vertices of $V_i(\mathbf{a})$. We transform it into a sum over $E \in \mathcal{E}_i$ in order to use the results of Section 2.1. We also write v_E, w_E instead of v,w, recalling that w is the neighbor vertex of v in counterclockwise orientation. Differentiating (11), we thus obtain

$$\nabla G(\mathbf{a}) \cdot \delta \mathbf{a} = \frac{1}{\kappa_0} \sum_{i=1}^{\kappa_0} \left(\frac{1}{12} \sum_{E \in \mathcal{E}_i} \sigma \nabla \gamma + \gamma \nabla \sigma \right) \cdot \delta \mathbf{a}, \tag{13}$$

where $\gamma \coloneqq |J(v_E, w_E, a_i)|$ and $\sigma \coloneqq \|v_E - a_i\|^2 + (v_E - a_i) \cdot (w_E - a_i) + \|w_E - a_i\|^2$. First, we compute

$$\nabla\gamma \cdot \delta\mathbf{a} = \nabla(|J(v_E, w_E, a_i)|) \cdot \delta\mathbf{a} = \operatorname{sign}(J(v_E, w_E, a_i))\nabla(J(v_E, w_E, a_i)) \cdot \delta\mathbf{a}.$$

Using (10), we have

$$\begin{aligned}
\nabla(J(v_E, w_E, a_i)) \cdot \delta\mathbf{a} &= -\nabla((v_E - a_i) \cdot (w_E - a_i)^\perp) \cdot \delta\mathbf{a} \\
&= -((D(v_E - a_i))^\top (w_E - a_i)^\perp + (D(w_E - a_i)^\perp)^\top (v_E - a_i)) \cdot \delta\mathbf{a} \\
&= -D(v_E - a_i)\delta\mathbf{a} \cdot (w_E - a_i)^\perp - D(w_E - a_i)^\perp \delta\mathbf{a} \cdot (v_E - a_i) \\
&= -D(v_E - a_i)\delta\mathbf{a} \cdot (w_E - a_i)^\perp + D(w_E - a_i)\delta\mathbf{a} \cdot (v_E - a_i)^\perp \\
&= -Dv_E\delta\mathbf{a} \cdot w_E^\perp + Dv_E\delta\mathbf{a} \cdot a_i^\perp + Da_i\delta\mathbf{a} \cdot w_E^\perp \\
&\quad + Dw_E\delta\mathbf{a} \cdot v_E^\perp - Dw_E\delta\mathbf{a} \cdot a_i^\perp - Da_i\delta\mathbf{a} \cdot v_E^\perp.
\end{aligned}$$

We have $Da_i\delta\mathbf{a} = \delta a_i$ and in view of (1), (2), (8) we have that $Dv_E\delta\mathbf{a} = \mathcal{F}(i, v_E)$ and $Dw_E\delta\mathbf{a} = \mathcal{F}(i, w_E)$, thus

$$\nabla\gamma\cdot\delta\mathbf{a} = \operatorname{sign}(J(v_E, w_E, a_i))(\mathcal{F}(i, v_E)\cdot(-w_E^\perp + a_i^\perp) + \mathcal{F}(i, w_E)\cdot(v_E^\perp - a_i^\perp) - \delta a_i(-w_E^\perp + v_E^\perp)). \quad (14)$$

Now, we compute

$$\nabla\sigma \cdot \delta\mathbf{a} = [\nabla(\|v_E - a_i\|^2) + \nabla((v_E - a_i) \cdot (w_E - a_i)) + \nabla(\|w_E - a_i\|^2)] \cdot \delta\mathbf{a}.$$

We have

$$\begin{aligned}
\nabla((v_E - a_i) \cdot (w_E - a_i)) \cdot \delta\mathbf{a} &= [(w_E - a_i)\nabla(v_E - a_i) + (v_E - a_i)\nabla(w_E - a_i)] \cdot \delta\mathbf{a} \\
&= (w_E - a_i) \cdot [(Dv_E - Da_i) \cdot \delta\mathbf{a}] + (v_E - a_i) \cdot [(Dw_E - Da_i) \cdot \delta\mathbf{a}] \\
&= (w_E - a_i) \cdot (\mathcal{F}(i, v_E) - \delta a_i) + (v_E - a_i) \cdot (\mathcal{F}(i, w_E) - \delta a_i) \\
&= \mathcal{F}(i, v_E) \cdot (w_E - a_i) + \mathcal{F}(i, w_E) \cdot (v_E - a_i) - \delta a_i(w_E + v_E - 2a_i),
\end{aligned}$$

and in a similar way

$$\begin{aligned}
\nabla(\|v_E - a_i\|^2) \cdot \delta\mathbf{a} &= 2\mathcal{F}(i, v_E) \cdot (v_E - a_i) - 2\delta a_i \cdot (v_E - a_i) \\
\nabla(\|w_E - a_i\|^2) \cdot \delta\mathbf{a} &= 2\mathcal{F}(i, w_E) \cdot (w_E - a_i) - 2\delta a_i \cdot (w_E - a_i).
\end{aligned}$$

Combining these results, we get

$$\begin{aligned}
\nabla\sigma \cdot \delta\mathbf{a} &= 2\mathcal{F}(i, v_E) \cdot (v_E - a_i) - 2\delta a_i \cdot (v_E - a_i) + \mathcal{F}(i, v_E) \cdot (w_E - a_i) + \mathcal{F}(i, w_E) \cdot (v_E - a_i) \\
&\quad - \delta a_i(w_E + v_E - 2a_i) + 2\mathcal{F}(i, w_E) \cdot (w_E - a_i) - 2\delta a_i \cdot (w_E - a_i) \\
&= \mathcal{F}(i, v_E) \cdot (2v_E + w_E - 3a_i) + \mathcal{F}(i, w_E) \cdot (v_E + 2w_E - 3a_i) \quad (15) \\
&\quad - 3\delta a_i \cdot (v_E + w_E - 2a_i).
\end{aligned}$$

Replacing (14) and (15) in (13), we obtain

$$\begin{aligned}\nabla G_i(\mathbf{a})\cdot\delta\mathbf{a} = \frac{1}{12}\sum_{E\in\mathcal{E}_i}\kappa_i(E)(\mathcal{F}(i,v_E)\cdot(-w_E^\perp+a_i^\perp)+\mathcal{F}(i,w_E)\cdot(v_E^\perp-a_i^\perp)-\delta a_i(-w_E^\perp+v_E^\perp))\\ +|J(v_E,w_E,a_i)|(\mathcal{F}(i,v_E)\cdot(2v_E+w_E-3a_i)+\mathcal{F}(i,w_E)\cdot(v_E+2w_E-3a_i)\\ -3\delta a_i\cdot(v_E+w_E-2a_i)),\end{aligned}$$

where

$$\kappa_i(E) := \operatorname{sign}(J(v_E,w_E,a_i))(\|v_E-a_i\|^2+(v_E-a_i)\cdot(w_E-a_i)+\|w_E-a_i\|^2).$$

Finally, rearranging the last expression for $\nabla G_i(\mathbf{a})\cdot\delta\mathbf{a}$, we get

$$\nabla G(\mathbf{a})\cdot\delta\mathbf{a} = \frac{1}{\kappa_0}\sum_{i=1}^{\kappa_0}\nabla G_i(\mathbf{a})\cdot\delta\mathbf{a}, \tag{16}$$

with

$$\begin{aligned}\nabla G_i(\mathbf{a})\cdot\delta\mathbf{a} = \frac{1}{12}\sum_{E\in\mathcal{E}_i}\mathcal{F}(i,v_E)\cdot(\kappa_i(E)(-w_E^\perp+a_i^\perp)+|J(v_E,w_E,a_i)|(2v_E+w_E-3a_i))\\ +\mathcal{F}(i,w_E)(\kappa_i(E)(v_E^\perp-a_i^\perp)+|J(v_E,w_E,a_i)|(v_E+2w_E-3a_i)\\ -\delta a_i\cdot(\kappa_i(E)(-w_E^\perp+v_E^\perp)+3|J(v_E,w_E,a_i)|(v_E+w_E-2a_i)).\end{aligned} \tag{17}$$

Formulas (16,17) are equivalent to (12) for the special case $\rho\equiv 1$. As explained above, formula (17) demonstrates a direct method for computing the gradient of objective functions that rely on Voronoi diagrams, without having to differentiate a parameterized integral. However, one drawback is that such explicit calculation only works for specific densities ρ, such as ρ constant or polynomial. Also, we expect this formula to be more computationally expensive than (12), since (12) results from a simplification (I_E^1 and I_E^2 canceling out). This means that (16,17) could be further simplified for more efficiency, but this reduces the advantage of the explicit calculation.

3 Numerical experiments

Numerical experiments on constructing centroidal Voronoi tessellations with special desired properties are presented in this section. CVTs are always constructed by minimizing a composite function $f(\mathbf{a}) \coloneqq \omega\, G(\mathbf{a}) + J(\mathbf{a})$, where $G(\mathbf{a})$ is the energy function given by (9), $J(\mathbf{a})$ varies depending on the specific geometry constraint imposed, and ω is a positive weight. In the minimization problem, sites a_k for $k=1,\ldots,\kappa_0$ are restricted to the domain $A=[0,\sqrt{\kappa_0}]^2$. Thus, optimization problems are bound-constrained.

In Section 3.1 we perform a numerical experiment in which we compare in practice the expressions (12) and (16,17) of the gradient of $G(\mathbf{a})$. In Section 3.2, we address the construction of CVTs with cells of identical area. In Section 3.3, we tackle the problem of constructing cells

without small edges. In Section 3.4, we deal with the construction of CVTs with cells of different sizes for different regions of the domain A.

The entire code is written in Fortran 90. The implementation provided in [1, 2, 3] is used to construct the Voronoi diagrams. All experiments were performed on a computer with an Apple M1 processor and 8 GB of RAM, running MacOS Sonoma (version 14.6.1). The code was compiled using the GCC 14.1.0 GFortran compiler with the -O3 optimization directive enabled.

3.1 Centroidal Voronoi tessellation

In this section, we minimize $G(\mathbf{a})$, defined in (9), without additional geometric terms. For general densities ρ, quadrature rules must be used to evaluate (9). In all our experiments, we focus on the case $\rho \equiv 1$ and the explicit form (11) of $G(\mathbf{a})$; this allows us to preserve the exactness of the function and to save computational time in the optimization process. Optimization problems are solved with the quasi-Newton method L-BFGS-B [5, 16, 17]. Default values are used for all its parameters. The stopping criterion is set to obtain an infinity norm of the continuous projected gradient less than or equal to $\epsilon = 10^{-8}$, i.e., to find an iterate $\mathbf{a}^\ell$ such that

$$\left\| P_\Omega \left(\mathbf{a}^\ell - \nabla G(\mathbf{a}^\ell) \right) - \mathbf{a}^\ell \right\|_\infty \leq \epsilon = 10^{-8},$$

where P_Ω represents the (orthogonal) projection operator onto the (convex) feasible set $\Omega = \{a_k \in A \text{ for } k = 1, \ldots, \kappa_0\}$. The initial point $\mathbf{a}^0$ of the optimization process consists of uniformly distributed random points in the domain A. This type of starting point is only used in this experiment, where only the function $G(\mathbf{a})$ is minimized. In Sections 3.2, 3.3, and 3.4, where the combination of $G(\mathbf{a})$ with an extra term is minimized, we take as starting point $\mathbf{a}^0$ the approximate solution $\mathbf{a}^\star$ to the problem of minimizing $G(\mathbf{a})$.

Tables 1 and 2 show some details of the optimization process on a set of instances with $\kappa_0 \in \{5, 10, 50, 100, 1000, 10000, 25000, 50000\}$, considering the expressions (12) and (16) for computing the gradient, respectively. In the tables, κ_0 is the number of sites considered. The column $G(\mathbf{a}^\star)$ denotes the value of the objective function at the last iterate $\mathbf{a}^\star$; while the column $\|\nabla G(\mathbf{a}^\star)\|_\infty$ denotes the sup-norm of its continuous projected gradient. The columns labeled "it", "fcnt", and "Time" represent the iteration count, the number of objective function evaluations, and the CPU execution time in seconds, respectively. The "fcnt/it" column shows the average number of function evaluations per iteration. From these tables we can see that, for each κ_0, the values of $G(\mathbf{a}^\star)$ are very similar when using both gradient expressions, and that in all cases it was possible to achieve the stopping criterion imposed. The tables also show that the performance of the method was slightly faster when using the expression (12) than when using the expression (16). This confirms the observation of Section 2.4, where it is explained that (16) is expected to be more computationally demanding than (12). For this reason, in Sections 3.2, 3.3, and 3.4 the expression (12) is used to compute the gradient of G. Figure 1 shows the resulting diagrams for the cases $\kappa_0 \in \{500, 1000\}$.

It is worth noting that derivatives exist under Assumption 1, which assumes the non-degeneracy of interfaces and vertices. However, changes in the topological structure of the diagram are allowed and occur frequently during the optimization process. This is a desirable feature of the procedure; otherwise, it would depend heavily on the initial guess. There is no

κ_0	$G(\mathbf{a}^\star)$	$\|\nabla G(\mathbf{a}^\star)\|_\infty$	it	fcnt	Time	fcnt/it
5	1.76349E−01	6.8E−09	24	30	0.001	1.25
10	1.69930E−01	8.6E−09	31	35	0.002	1.13
50	1.65505E−01	9.6E−09	88	95	0.022	1.08
100	1.63885E−01	9.9E−09	115	124	0.050	1.08
500	1.62358E−01	9.5E−09	254	278	0.320	1.09
1000	1.62019E−01	8.7E−09	477	530	1.208	1.11
10000	1.61780E−01	8.8E−09	494	505	12.915	1.02
25000	1.61708E−01	8.3E−09	753	780	60.151	1.04
50000	1.61661E−01	8.7E−09	1221	1271	472.639	1.04

Table 1: Details on the optimization process and resulting solutions to minimize $G(\mathbf{a})$ as κ_0 increases, using the gradient calculation described in (12).

κ_0	$G(\mathbf{a}^\star)$	$\|\nabla G(\mathbf{a}^\star)\|_\infty$	it	fcnt	Time	fcnt/it
5	1.76349E−01	6.8E−09	24	30	0.002	1.25
10	1.69930E−01	8.6E−09	31	35	0.004	1.13
50	1.65443E−01	9.6E−09	107	120	0.034	1.12
100	1.63885E−01	9.9E−09	115	124	0.061	1.08
500	1.62241E−01	4.9E−09	239	258	0.376	1.08
1000	1.62170E−01	8.5E−09	202	208	0.621	1.03
10000	1.61783E−01	9.1E−09	507	522	16.301	1.03
25000	1.61719E−01	8.8E−09	736	762	63.164	1.04
50000	1.61649E−01	8.6E−09	1285	1343	598.587	1.05

Table 2: Details on the optimization process and resulting solutions to minimize $G(\mathbf{a})$ as κ_0 increases, using the gradient calculation described in (16).

need to impose constraints to prevent topological changes. Only degenerate configurations are undesirable, but these never occur in practice, as they represent a set of measure zero within the search space.

3.2 Centroidal Voronoi tessellation with cells of identical area.

In this section, we consider the merit function given by

$$f_1(\mathbf{a}) := \omega\, G(\mathbf{a}) + J^1(\mathbf{a}),$$

where

$$J^1(\mathbf{a}) := \frac{1}{\kappa_0}\sum_{i=1}^{\kappa_0}[J_i^1(\mathbf{a})]^2 \text{ with } J_i^1(\mathbf{a}) := \left(\int_{V_i(\mathbf{a})} dx\right) / \left(\frac{1}{\kappa_0}\int_A dx\right) - 1,$$

and $\omega \geq 0$ is given. The function $J^1(\mathbf{a})$ quantifies the deviation of the area of each Voronoi cell $V_i(\mathbf{a})$ from the average area of the cells within the domain A. The purpose of minimizing $f_1(\mathbf{a})$ is to find CVTs with cells of similar area, i.e., what is expected in an approximate solution $\mathbf{a}$ is

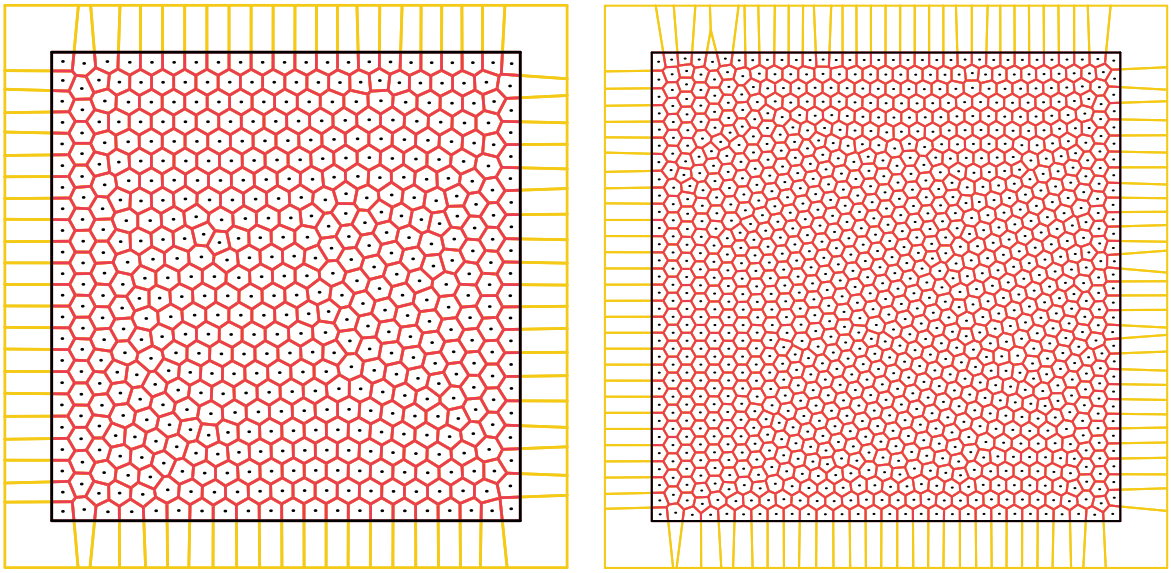


Figure 1: Centroidal Voronoi tessellations with $\kappa_0 \in \{500, 1000\}$. Results obtained using the gradient formula (12).

that for all i, $|V_i(\mathbf{a})| \approx \frac{1}{\kappa_0}\int_A dx = 1$, because $A = [0, \sqrt{\kappa_0}]^2$. Since J_i^1 corresponds to (4) with $f \equiv 1$, applying (5), we have that

$$\nabla J_i^1(\mathbf{a}) \cdot \delta\mathbf{a} = \frac{\kappa_0}{|A|} \sum_{E \in \mathcal{E}_i^{\text{int}}} \frac{|E|}{\|a_i - a_{k(i,E)}\|}[\delta a_i \cdot (p_E - a_i) - \delta a_{k(i,E)} \cdot (p_E - a_{k(i,E)})],$$

where $p_E \coloneqq (v_E + w_E)/2$ and $k(i, E)$ is the index such that $E = \overline{V_i(\mathbf{a})} \cap \overline{V_{k(i,E)}(\mathbf{a})}$.

The choice of the parameter ω is important for the optimization process to obtain the desired results. We will present an appropriate choice for ω that was found in the particular case where $\kappa_0 = 10$ is used, and then show the results obtained when optimizing the cases with $\kappa_0 \in \{500, 1000\}$ using the same value for ω. Figure 2(a) shows the solution of minimizing $G(\mathbf{a})$, while Figures 2(b-e) show the solutions obtained by minimizing $f_1(\mathbf{a})$ with $\omega \in \{1, 0.1, 0.01, 0.001\}$. In the figures, the cells $V_i(\mathbf{a})$ that satisfy $|J_i^1(\mathbf{a})| > 10^{-3}$ are colored green. Therefore, the uncolored cells V_i satisfy $|J_i^1(\mathbf{a})| \leq 10^{-3}$, which means that the uncolored cells have an area that is very close to the desired area (the relative error of the cell area to the ideal area is less than or equal to 0.1%). The desired goal is reached when all cells are uncolored, which is the case for $\omega = 0.001$. The details of the solutions found are shown in Table 3. In particular, it shows the area of each cell. It is interesting to compare the cell areas of the diagrams constructed by minimizing G alone and f_1 with $\omega = 0.001$. Table 4 summarizes key aspects of the optimization process, providing the objective function value $f_1(\mathbf{a}^\star)$ at the last iterate $\mathbf{a}^\star$, the sup-norm of the continuous projected gradient $\|f_1(\mathbf{a}^\star)\|_\infty$, the value of the CVT energy function $G(\mathbf{a}^\star)$ and the value of $J^1(\mathbf{a}^\star)$, as well as the iteration count, number of function evaluations, and CPU execution time in seconds. The numbers in the table show that, regardless of the value of ω,

the problems were easily solved, with the optimization process stopping in all cases due to the imposed stopping criterion. The value of G in the solution $\mathbf{a}^\star$ obtained by minimizing G alone is $G(\mathbf{a}^\star) = 1.69930\text{E-}01$; see Table 2. As ω decreases, Table 4 shows that $J^1(\mathbf{a}^\star)$ improves by at least an order of magnitude, while $G(\mathbf{a}^\star)$ deteriorates only slightly, remaining close to the value obtained by minimizing G alone. This means that minimizing f_1 succeeds in designing a diagram with the desired geometric properties at the cost of only a small increase in the CVT energy function, which is the function that a CVT diagram should minimize. In short, minimizing f_1 succeeds in designing a CVT diagram with the desired characteristics.

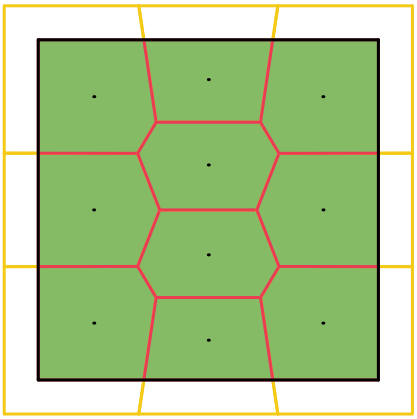
Cell	$G(\mathbf{a})$		$f_1(\mathbf{a})$ with $\omega = 1$		$f_1(\mathbf{a})$ with $\omega = 0.1$		$f_1(\mathbf{a})$ with $\omega = 0.01$		$f_1(\mathbf{a})$ with $\omega = 0.001$	
i	$\lvert V_i(\mathbf{a}^\star)\rvert$	$\lvert J_i^1(\mathbf{a}^\star)\rvert$	$\lvert V_i(\mathbf{a}^\star)\rvert$	$\lvert J_i^1(\mathbf{a}^\star)\rvert$	$\lvert V_i(\mathbf{a}^\star)\rvert$	$\lvert J_i^1(\mathbf{a}^\star)\rvert$	$\lvert V_i(\mathbf{a}^\star)\rvert$	$\lvert J_i^1(\mathbf{a}^\star)\rvert$	$\lvert V_i(\mathbf{a}^\star)\rvert$	$\lvert J_i^1(\mathbf{a}^\star)\rvert$
1	8.31E−01	1.69E−01	9.97E−01	3.28E−03	9.99E−01	7.09E−04	9.98E−01	1.58E−03	1.00E+00	1.60E−04
2	1.08E+00	8.10E−02	9.97E−01	3.28E−03	9.99E−01	7.09E−04	1.00E+00	9.40E−04	1.00E+00	9.50E−05
3	1.09E+00	8.75E−02	1.04E+00	4.08E−02	1.01E+00	6.30E−03	1.00E+00	9.86E−04	1.00E+00	9.95E−05
4	1.09E+00	8.75E−02	1.04E+00	4.08E−02	1.01E+00	6.30E−03	1.00E+00	9.86E−04	1.00E+00	9.95E−05
5	9.13E−01	8.72E−02	9.55E−01	4.47E−02	9.94E−01	6.43E−03	9.99E−01	1.33E−03	1.00E+00	1.34E−04
6	1.09E+00	8.75E−02	1.01E+00	1.04E−02	1.00E+00	1.55E−03	1.00E+00	9.86E−04	1.00E+00	9.95E−05
7	9.13E−01	8.72E−02	9.55E−01	4.47E−02	9.94E−01	6.43E−03	9.99E−01	1.33E−03	1.00E+00	1.34E−04
8	1.08E+00	8.10E−02	9.97E−01	3.28E−03	9.99E−01	7.09E−04	1.00E+00	9.40E−04	1.00E+00	9.50E−05
9	8.31E−01	1.69E−01	9.97E−01	3.28E−03	9.99E−01	7.09E−04	9.98E−01	1.58E−03	1.00E+00	1.60E−04
10	1.09E+00	8.75E−02	1.01E+00	1.04E−02	1.00E+00	1.55E−03	1.00E+00	9.86E−04	1.00E+00	9.95E−05

Table 3: Obtained values for the area $|V_i(\mathbf{a}^\star)|$ of each cell and $J^1(\mathbf{a}^\star)$ after minimizing the functions $G(\mathbf{a})$ and $f_1(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^1(\mathbf{a})$ with $\omega \in \{1, 0.1, 0.01, 0.001\}$.

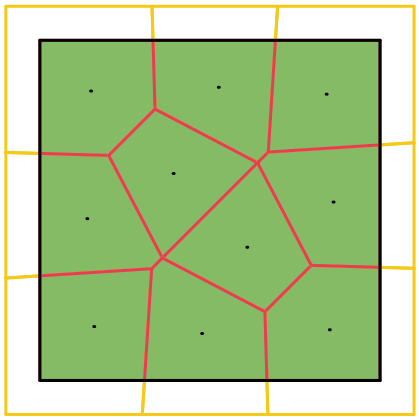
ω	$f_1(\mathbf{a}^\star)$	$\lVert\nabla f_1(\mathbf{a}^\star)\rVert_\infty$	$G(\mathbf{a}^\star)$	$J^1(\mathbf{a}^\star)$	it	fcnt	Time
1	1.73112E−01	1.9E−09	1.72354E−01	7.57491E−04	44	63	0.004
0.1	1.74133E−02	9.1E−09	1.73964E−01	1.68787E−05	94	119	0.005
0.01	1.82952E−03	9.7E−09	1.82810E−01	1.41961E−06	29	32	0.002
0.001	1.83081E−04	2.7E−09	1.83066E−01	1.44641E−08	35	38	0.002

Table 4: Details of the process of minimizing the function $f_1(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^1(\mathbf{a})$ varying ω.

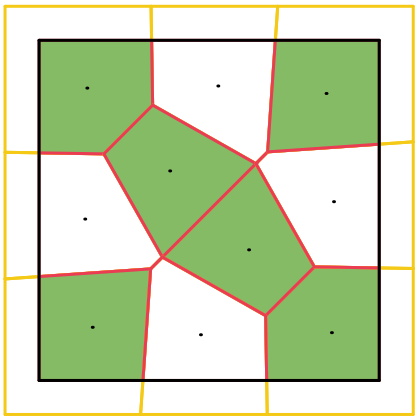
Taking into account $\kappa_0 = 1000$, we also perform the experiment of minimizing only G and f_1 with $\omega \in \{1, 0.1, 0.01, 0.001\}$. Figure 3 shows the distribution of cell areas in the four different approximate solutions $\mathbf{a}^\star(\omega)$ found by minimizing the function $f_1(\mathbf{a})$ varying ω. Consider the solutions $\mathbf{a}^\star(\omega)$ for the different values of ω and let $\nu_1 = \min_{\{\omega,i\}} |V_i(\mathbf{a}^\star(\omega))|$, $\nu_{101} = \max_{\{\omega,i\}} |V_i(\mathbf{a}^\star(\omega))|$, $\Delta\nu = (\nu_{101} - \nu_1)/100$, and $\nu_j = \nu_1 + j\,\Delta\nu$ for $j = 2, \ldots, 100$. The figure shows the distribution of the values of $|V_i(\mathbf{a}^\star(\omega))|$ for $\omega \in \{1, 0.1, 0.01, 0.001\}$ and $i = 1, \ldots, \kappa_0$ over the intervals $[\nu_j, \nu_{j+1}]$ for $j = 1, \ldots, 100$. More specifically, there is a graph for each $\mathbf{a}^\star(\omega)$ and, for each value $\frac{1}{2}(\nu_j + \nu_{j+1})$ in the abscissa, the graph shows in the ordinate the proportion of cells $V_i(\mathbf{a}^\star(\omega))$ whose area $|V_i(\mathbf{a}^\star(\omega))|$ is in the interval $[\nu_j, \nu_{j+1}]$. The plot clearly shows that, the smaller the value of ω, the larger the proportion of cells in the solution $\mathbf{a}^\star(\omega)$ whose area is close to 1. Figure 4(a-b) shows the diagrams obtained by minimizing G alone and f_1 with $\omega = 0.001$, respectively. In both figures, cells $V_i(\mathbf{a})$ such that $|J_i^1(\mathbf{a})| > 10^{-3}$ are painted green. Basically speaking, this means that when minimizing G alone, almost none of the cells have the desired area, while when minimizing f_1 with $\omega = 0.001$, all have the desired area.



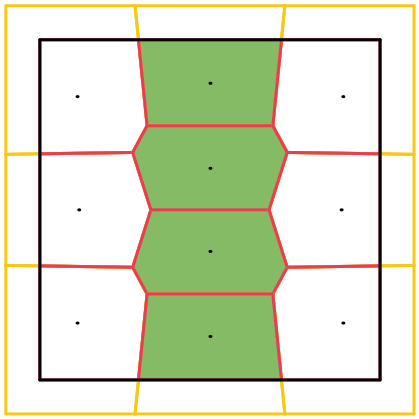
(a) $G(\mathbf{a})$



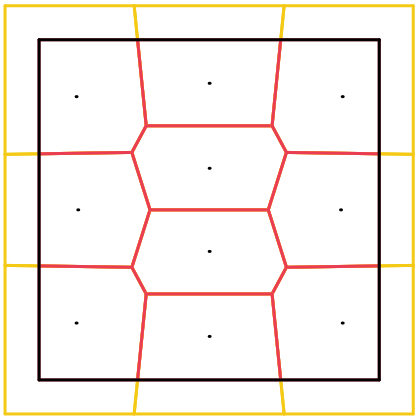
(b) $f(\mathbf{a})$ with $\omega = 1$



(c) $f(\mathbf{a})$ with $\omega = 0.1$



(d) $f(\mathbf{a})$ with $\omega = 0.01$



(e) $f(\mathbf{a})$ with $\omega = 0.001$

Figure 2: Centroidal Voronoi tessellations with $\kappa_0 = 10$. In (a) we show the result of minimizing the function $G(\mathbf{a})$. In (b-e) we show the diagrams obtained by minimizing $f_1(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^1(\mathbf{a})$ with decreasing values of ω.

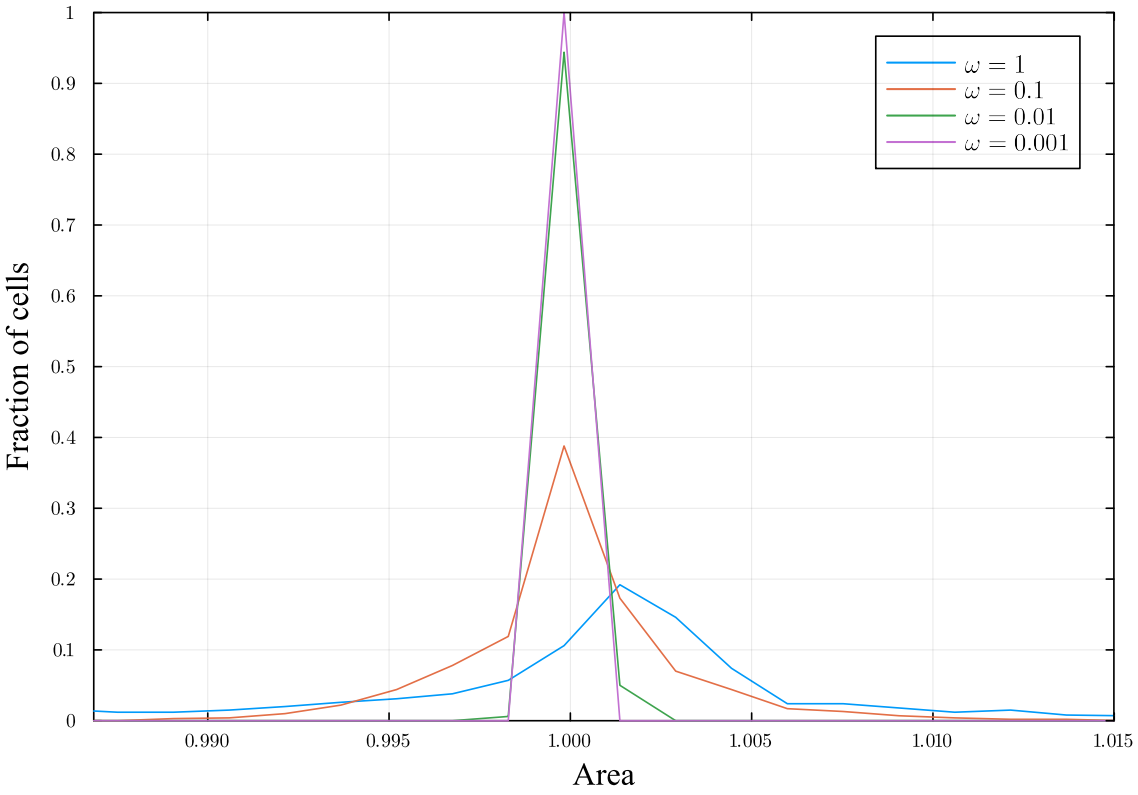


Figure 3: Distribution of the cell areas in the approximate solutions found when minimizing the function $f_1(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^1(\mathbf{a})$ with $\kappa_0 = 1000$ considering $\omega \in \{1, 0.1, 0.01, 0.001\}$.

3.3 Centroidal Voronoi tessellations with edge length constraints

This section focuses on cell edge length considerations. Scrutinizing the cells in Figure 4(a), we observe that small-edged cells are present in the tessellation.

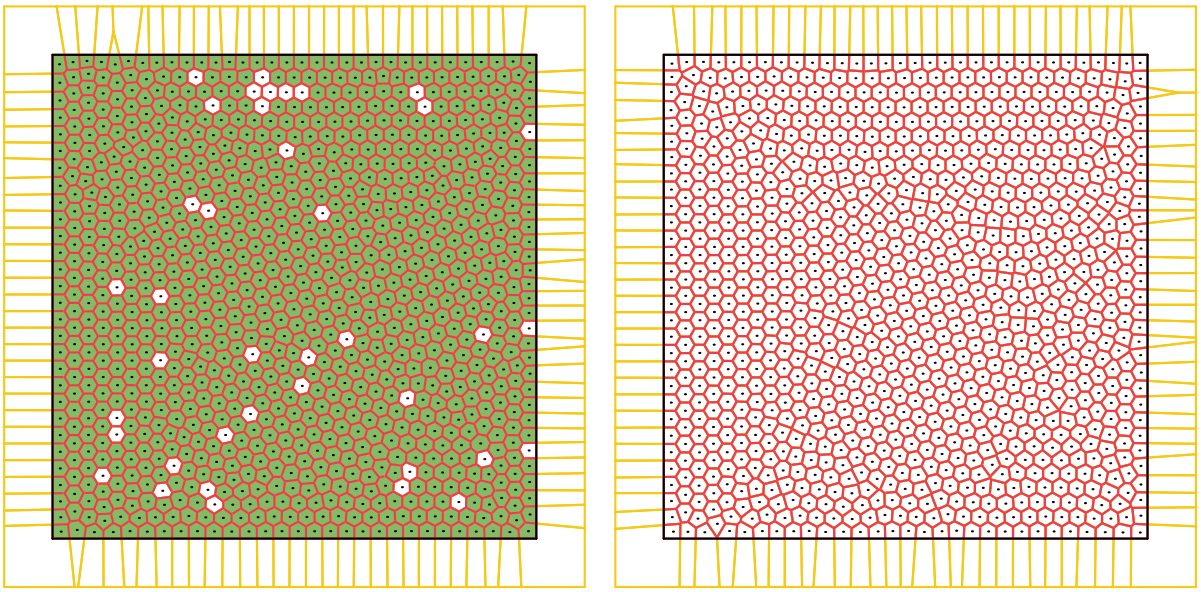
Given a fraction $c_2 \in (0, 1)$, an edge E of cell $V_i(\mathbf{a})$ is considered small if its length $|E|$ is less than $c_2\bar{E}_i$, where $\bar{E}_i$ represents the average edge length of the cell, calculated as the cell perimeter P_i divided by the number of edges n_i. To construct CVTs that avoid cells with small edges, for a given tolerance $c_2 \in (0, 1)$, we minimize

$$f_2(\mathbf{a}) := \omega\, G(\mathbf{a}) + J^2(\mathbf{a}),$$

where

$$J^2(\mathbf{a}) := \sum_{i=1}^{\kappa_0} J_i^2(\mathbf{a}) \text{ with } J_i^2(\mathbf{a}) := \frac{1}{n_i} \sum_{E \in \mathcal{E}_i} \min\left\{0, \frac{|E|}{\bar{E}_i} - c_2\right\}^2 \tag{18}$$

and $\omega \geq 0$ is a given constant. In (18), $\mathcal{E}_i$ is the set of edges of cell $V_i(\mathbf{a})$ and $n_i = |\mathcal{E}_i|$. Given $c_2 \in (0, 1)$, if all edges $E \in \mathcal{E}_i$ of a cell $V_i(\mathbf{a})$ satisfy $|E| \geq c_2\bar{E}_i$, i.e., if the size of each cell edge is at least $100\% \times c_2$ of the average size, then $|E|/\bar{E}_i - c_2 \geq 0$ for all E, and hence J_i^2 vanishes.



(a) $G(\mathbf{a})$ (b) $f_1(\mathbf{a})$ with $\omega = 0.001$

Figure 4: Centroidal Voronoi tessellation with $\kappa_0 = 1000$. In (a) we show the diagram resulting from minimizing the function $G(\mathbf{a})$. In (b) we show the diagrams obtained by minimizing $f_1(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^1(\mathbf{a})$ with $\omega = 0.001$.

Thus, J_i^2 measures the violation of the constraints on the minimum size of the edges of the cell $V_i(\mathbf{a})$. The gradient ∇J_i^2 depends on $\nabla|E|$ and $|E|$ corresponds to (6) with $f \equiv 1$. Therefore, by (7,8), we have that

$$\nabla J_i^2(\mathbf{a}) \cdot \delta\mathbf{a} = \sum_{E \in \mathcal{E}_i} \mu(E)([\mathcal{F}(i, w_E) - \mathcal{F}(i, v_E)] \cdot \tau_E)$$

with

$$\mu(E) := \frac{2}{P_i} \left(\min\left\{0, \frac{|E|}{\bar{E}_i} - c_2\right\} - \sum_{\tilde{E} \in \mathcal{E}_i} \frac{|\tilde{E}|}{P_i} \min\left\{0, \frac{|\tilde{E}|}{\bar{E}_i} - c_2\right\} \right).$$

To obtain an appropriate value for ω, we analyze the results of minimizing $f_2(\mathbf{a})$ with $c_2 = 0.5$ and $\kappa_0 = 10$, varying $\omega \in \{1, 0.1, 0.01\}$. Table 5 shows the results. The table indicates that all problems were efficiently solved, meeting the specified stopping criterion without difficulty. Furthermore, we can see that, as expected, $J^2(\mathbf{a}^\star(\omega))$ decreases as ω decreases. We can also see that as ω decreases, the value of G increases. However, it increases only slightly and always remains close to the value obtained by minimizing the function G alone, which is 1.69930E-01 (see Table 2). This suggests that the solutions found, which satisfy the desired geometric properties, also preserve the property of minimizing the CVT energy function. Figures 5(a-d) show the diagrams obtained when minimizing $G(\mathbf{a})$ and $f_2(\mathbf{a})$ with $c_2 = 0.5$ and varying $\omega \in \{1, 0.1, 0.01\}$. Considering $c_2 = 0.5$ means that we expect diagrams in which no cell has an edge smaller than 50% of the average size of its edges. In the figures, colored cells are those that

do not satisfy the desired property. This means that satisfactory results were obtained when minimizing $f_2(\mathbf{a})$ with $\omega = 0.1$ and $\omega = 0.01$. Table 6 analyzes the obtained solutions in detail. For each cell of each solution, the table shows the size of its edges, the average size of the edges, the ratio of the smallest edge to the average, and the value of J_i^2. The table clearly shows that as ω decreases, the smallest edge of each cell approaches the smallest required proportion, that is, 50%.

ω	$f_2(\mathbf{a}^\star)$	$\|\nabla f_2(\mathbf{a}^\star)\|_\infty$	$G(\mathbf{a}^\star)$	$J^2(\mathbf{a}^\star)$	it	fcnt	Time
1	1.70217E−01	1.9E−09	1.70079E−01	1.37640E−04	8	9	0.002
0.1	1.70437E−02	7.3E−09	1.70393E−01	4.34693E−06	17	21	0.002
0.01	1.70479E−03	9.9E−09	1.70474E−01	5.12211E−08	27	30	0.003

Table 5: Details of the process of minimizing $f_2(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^2(\mathbf{a})$ with $c_2 = 0.5$ varying ω.

Upon determining the value $\omega = 0.01$, additional experiments with $\kappa_0 \in \{500, 1000\}$ were performed. The experiments consisted of minimizing $G(\mathbf{a})$ as well as $f_2(\mathbf{a})$ with $\omega = 0.01$, varying c_2. The results are presented in Tables 7 and 8 and illustrated in Figures 6 and 7. According to the table results, increasing the value of c_2 leads to a moderate increase in the computational cost of the optimization process. However, in all cases, the problems were easily solved. The interesting observation is that when c_2 increases, solutions $\mathbf{a}^\star$ with more restrictive geometric conditions are calculated with a very small increase in the value of $G(\mathbf{a}^\star)$. Moreover, the values of G remain close to the value obtained when minimizing G alone (see Table 2). We can also see that as c_2 increases, $J^2(\mathbf{a}^\star)$ also increases and, at $c_2 = 0.9$, the geometric condition is not met. In the figures, seven shades of blue were used to paint the cells. The darkest cells are those with side sizes between 10-20% of the mean. The lighter cells are those with side sizes between 70 and 80% of the mean. The figures show that, as the value of c_2 increases, cells with edge sizes smaller than 80% of the average size disappear.

Figure 8 analyzes ten different diagrams with $\kappa_0 = 1000$ obtained by minimizing G alone and the function f_2 with $\omega = 0.01$ and $c_2 \in \{0.1, 0.2, \ldots, 0.9\}$. For a given solution $\mathbf{a}^\star$, the figure shows the proportion of cells $V_i(\mathbf{a}^\star)$ satisfying $J^2(\mathbf{a}^\star) = 0$ as a function of $c \in [0, 1]$. The zoom in the figure shows that minimizing $f_2(\mathbf{a})$ with $c_2 \leq 0.8$ ensures that every cell satisfies the condition "all my edges are at least $100\% \times c_2$ the average size of my edges". When $c_2 = 0.9$, the geometric constraints are too restrictive and the solution does not satisfy the desired property. This corresponds to the fact that the corresponding curve falls below 1 for $c < 0.9$. As a general observation, all the ten curves look very similar when c varies from 0 to 0.35. This is because, in general, when a diagram is built by minimizing G alone, there are only a few cells with small edges. The "difference" between the curves shows that these few undesired edges are eliminated when f_2 is minimized for increasing values of c_2. The curve relative to the diagram obtained by minimizing G overlaps with that of minimizing f_2 with $c_2 = 0.1$. These curves are equal because, when G is minimized, no cell has an edge that is less than 10% of the average.

	Cell i	Edges sizes						$\bar{E}_i$	$100(\lvert E_i^{\min}\rvert/\bar{E}_i)\%$	$J_i^2(\mathbf{a}^\star)$
Min $G(\mathbf{a})$	1	1.20E+00	7.75E−01	9.70E−01	7.75E−01	-	-	9.30E−01	83.4%	0.00E+00
	2	1.05E+00	9.24E−01	5.65E−01	5.65E−01	9.24E−01	-	8.06E−01	70.0%	0.00E+00
	3	3.36E−01	7.75E−01	9.82E−01	1.05E+00	9.24E−01	-	8.14E−01	**41.3%**	1.51E−03
	4	9.82E−01	1.05E+00	9.24E−01	3.36E−01	7.75E−01	-	8.14E−01	**41.3%**	1.51E−03
	5	3.36E−01	9.70E−01	3.36E−01	5.65E−01	9.02E−01	5.65E−01	6.12E−01	54.9%	0.00E+00
	6	9.24E−01	1.05E+00	9.82E−01	7.75E−01	3.36E−01	-	8.14E−01	**41.3%**	1.51E−03
	7	9.02E−01	5.65E−01	3.36E−01	9.70E−01	3.36E−01	5.65E−01	6.12E−01	54.9%	0.00E+00
	8	5.65E−01	9.24E−01	1.05E+00	9.24E−01	5.65E−01	-	8.06E−01	70.0%	0.00E+00
	9	7.75E−01	1.20E+00	7.75E−01	9.70E−01	-	-	9.30E−01	83.4%	0.00E+00
	10	1.05E+00	9.82E−01	7.75E−01	3.36E−01	9.24E−01	-	8.14E−01	**41.3%**	1.51E−03

	Cell i	Edges sizes						$\bar{E}_i$	$100(\lvert E_i^{\min}\rvert/\bar{E}_i)\%$	$J_i^2(\mathbf{a}^\star)$
Min $f_2(\mathbf{a})$ with $\omega = 1$	1	1.22E+00	7.54E−01	9.47E−01	7.54E−01	-	-	9.18E−01	82.2%	0.00E+00
	2	1.07E+00	9.22E−01	5.56E−01	5.56E−01	9.22E−01	-	8.05E−01	69.1%	0.00E+00
	3	3.73E−01	7.54E−01	9.73E−01	1.05E+00	9.22E−01	-	8.14E−01	**45.9%**	3.44E−04
	4	9.73E−01	1.05E+00	9.22E−01	3.73E−01	7.54E−01	-	8.14E−01	**45.9%**	3.44E−04
	5	3.73E−01	9.47E−01	3.73E−01	5.56E−01	9.01E−01	5.56E−01	6.18E−01	60.4%	0.00E+00
	6	9.22E−01	1.05E+00	9.73E−01	7.54E−01	3.73E−01	-	8.14E−01	**45.9%**	3.44E−04
	7	9.01E−01	5.56E−01	3.73E−01	9.47E−01	3.73E−01	5.56E−01	6.18E−01	60.4%	0.00E+00
	8	5.56E−01	9.22E−01	1.07E+00	9.22E−01	5.56E−01	-	8.05E−01	69.1%	0.00E+00
	9	7.54E−01	1.22E+00	7.54E−01	9.47E−01	-	-	9.18E−01	82.2%	0.00E+00
	10	1.05E+00	9.73E−01	7.54E−01	3.73E−01	9.22E−01	-	8.14E−01	**45.9%**	3.44E−04

	Cell i	Edges sizes						$\bar{E}_i$	$100(\lvert E_i^{\min}\rvert/\bar{E}_i)\%$	$J_i^2(\mathbf{a}^\star)$
Min $f_2(\mathbf{a})$ with $\omega = 0.1$	1	1.23E+00	7.40E−01	9.29E−01	7.40E−01	-	-	9.10E−01	81.3%	0.00E+00
	2	1.08E+00	9.20E−01	5.50E−01	5.50E−01	9.20E−01	-	8.04E−01	68.4%	0.00E+00
	3	4.01E−01	7.40E−01	9.66E−01	1.04E+00	9.20E−01	-	8.14E−01	**49.3%**	1.09E−05
	4	9.66E−01	1.04E+00	9.20E−01	4.01E−01	7.40E−01	-	8.14E−01	**49.3%**	1.09E−05
	5	4.01E−01	9.29E−01	4.01E−01	5.50E−01	9.02E−01	5.50E−01	6.22E−01	64.4%	0.00E+00
	6	9.20E−01	1.04E+00	9.66E−01	7.40E−01	4.01E−01	-	8.14E−01	**49.3%**	1.09E−05
	7	9.02E−01	5.50E−01	4.01E−01	9.29E−01	4.01E−01	5.50E−01	6.22E−01	64.4%	0.00E+00
	8	5.50E−01	9.20E−01	1.08E+00	9.20E−01	5.50E−01	-	8.04E−01	68.4%	0.00E+00
	9	7.40E−01	1.23E+00	7.40E−01	9.29E−01	-	-	9.10E−01	81.3%	0.00E+00
	10	1.04E+00	9.66E−01	7.40E−01	4.01E−01	9.20E−01	-	8.14E−01	**49.3%**	1.09E−05

	Cell	Edges sizes						$\bar{E}_i$	$100(\lvert E_i^{\min}\rvert/\bar{E}_i)\%$	$J_i^2(\mathbf{a}^\star)$
Min $f_2(\mathbf{a})$ with $\omega = 0.01$	1	1.23E+00	7.37E−01	9.26E−01	7.37E−01	-	-	9.08E−01	81.1%	0.00E+00
	2	1.08E+00	9.20E−01	5.48E−01	5.48E−01	9.20E−01	-	8.04E−01	68.2%	0.00E+00
	3	4.06E−01	7.37E−01	9.65E−01	1.04E+00	9.20E−01	-	8.14E−01	**49.9%**	1.28E−07
	4	9.65E−01	1.04E+00	9.20E−01	4.06E−01	7.37E−01	-	8.14E−01	**49.9%**	1.28E−07
	5	4.06E−01	9.26E−01	4.06E−01	5.48E−01	9.02E−01	5.48E−01	6.23E−01	65.2%	0.00E+00
	6	9.20E−01	1.04E+00	9.65E−01	7.37E−01	4.06E−01	-	8.14E−01	**49.9%**	1.28E−07
	7	9.02E−01	5.48E−01	4.06E−01	9.26E−01	4.06E−01	5.48E−01	6.23E−01	65.2%	0.00E+00
	8	5.48E−01	9.20E−01	1.08E+00	9.20E−01	5.48E−01	-	8.04E−01	68.2%	0.00E+00
	9	7.37E−01	1.23E+00	7.37E−01	9.26E−01	-	-	9.08E−01	81.1%	0.00E+00
	10	1.04E+00	9.65E−01	7.37E−01	4.06E−01	9.20E−01	-	8.14E−01	**49.9%**	1.28E−07

Table 6: Details of cells after minimizing the objective functions $G(\mathbf{a})$ and $f_2(\mathbf{a})$ with $\omega \in \{1, 0.1, 0.01\}$.

3.4 Density-based centroidal Voronoi tessellations

This section focuses on constructing CVTs where cell sizes are determined by a function $\psi : A \to \mathbb{R}$. For this, following [3], we define our objective function as

$$f_3(\mathbf{a}) = G(\mathbf{a}) + \omega\, J^3(\mathbf{a}),$$

c_2	$f_2(\mathbf{a}^\star)$	$\|\nabla f_2(\mathbf{a}^\star)\|_\infty$	$G(\mathbf{a}^\star)$	$J^2(\mathbf{a}^\star)$	it	fcnt	Time
0.2	1.62359E−03	6.4E−09	1.62359E−01	1.36588E−11	64	101	0.141
0.3	1.62361E−03	8.4E−09	1.62361E−01	6.44011E−11	114	145	0.186
0.4	1.62368E−03	9.2E−09	1.62368E−01	2.48965E−10	233	311	0.350
0.5	1.62384E−03	9.3E−09	1.62384E−01	8.14196E−10	335	454	0.498
0.6	1.62424E−03	6.1E−09	1.62423E−01	2.89568E−09	345	427	0.480
0.7	1.62564E−03	8.6E−09	1.62562E−01	1.46599E−08	507	558	0.638
0.8	1.63419E−03	8.6E−09	1.63398E−01	2.15309E−07	633	642	0.781
0.9	1.74145E−03	9.0E−09	1.69973E−01	4.17186E−05	722	753	0.997

Table 7: Details on the optimization process and the resulting solutions for finding CVTs that prevent cells with relatively small edges for $\kappa_0 = 500$.

c_2	$f_2(\mathbf{a}^\star)$	$\|\nabla f_2(\mathbf{a}^\star)\|_\infty$	$G(\mathbf{a}^\star)$	$J^2(\mathbf{a}^\star)$	it	fcnt	Time
0.3	1.62019E−03	9.3E−09	1.62019E−01	3.01081E−12	48	63	0.179
0.4	1.62022E−03	9.8E−09	1.62022E−01	7.05410E−11	81	110	0.284
0.5	1.62032E−03	6.9E−09	1.62032E−01	3.57745E−10	235	300	0.711
0.6	1.62063E−03	4.6E−09	1.62063E−01	1.33134E−09	310	387	0.918
0.7	1.62174E−03	8.7E−09	1.62173E−01	8.84976E−09	475	512	1.235
0.8	1.62862E−03	9.1E−09	1.62850E−01	1.23288E−07	777	787	1.970
0.9	1.71539E−03	9.5E−09	1.67819E−01	3.72017E−05	1357	1450	4.031

Table 8: Details on the optimization process and the resulting solutions for finding CVTs that prevent cells with relatively small edges for $\kappa_0 = 1000$.

where

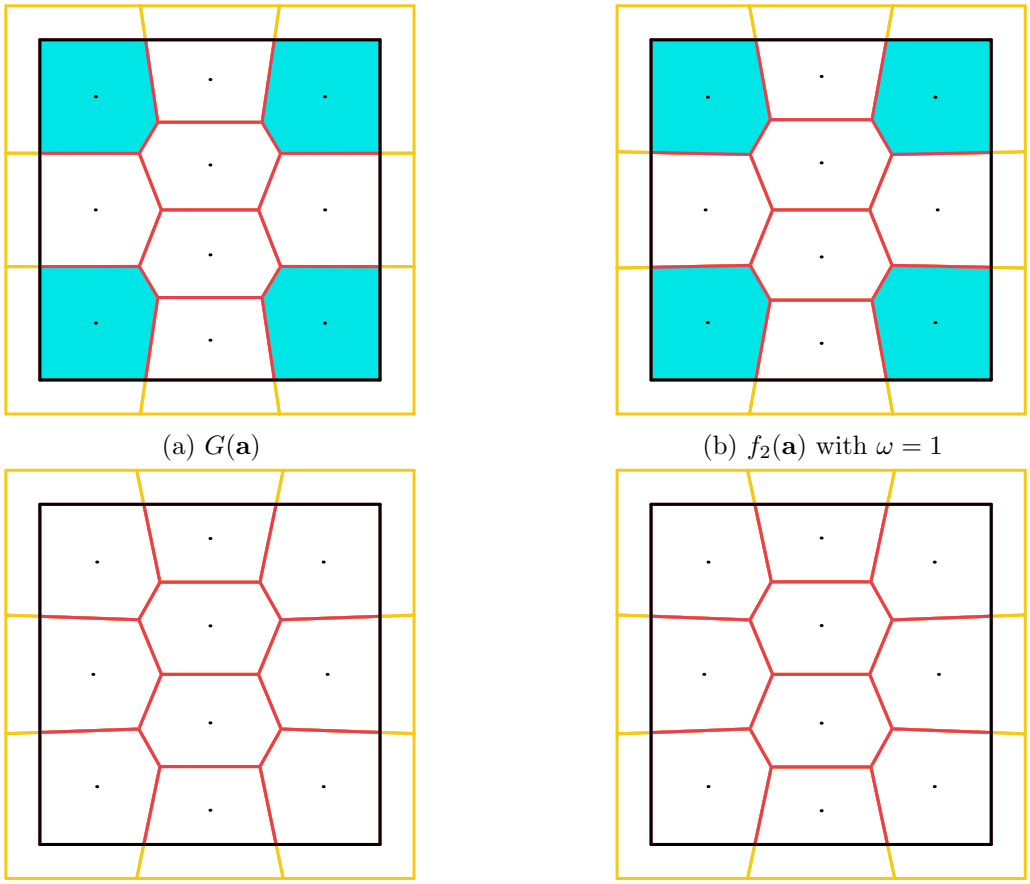
$$J^3(\mathbf{a}) := \frac{1}{\kappa_0}\sum_{i=1}^{\kappa_0}[J_i^3(\mathbf{a})]^2 \text{ with } J_i^3(\mathbf{a}) := \left(\int_{V_i(\mathbf{a})} dx\right) / \left(\frac{1}{\kappa_0}\int_A dx\right) - \psi(a_i),$$

where $\omega \geq 0$ is given.

The function ψ controls the desired proportion between the area of each cell $V_i(\mathbf{a})$ and the average area of all cells. The gradient $J^3(\mathbf{a})$ depends on $\nabla J_i^3(\mathbf{a})$ and $\nabla J_i^3(\mathbf{a}) = \nabla J_i^1(\mathbf{a}) - \nabla_{a_i}\psi(a_i)\cdot\delta a_i$. A challenge with the defined function J^3 is that the sum of the target cell areas may not match the total area of the region A. Consequently, it is not expected that J^3 vanish in its global minimizer, making it impossible to establish in practice that the global minimizer has been reached. Another option would be to consider a merit function like the one defined in Section 3.2, but with an arbitrary non-constant density function ρ. This was the approach considered in [7, 13]. Its disadvantage compared to our function J^3 is that it relies on quadrature rules to approximate the merit function and its derivatives, while our approach still allows an exact evaluation of the integrals.

In the numerical experiments, we considered $\kappa_0 = 1000$ and arbitrarily defined

(a) $\psi(z) = \psi_1(z) := a((\bar{z}_2 - (\bar{z}_1/4)^2)^2 + (\bar{z}_1/4 - 1)^2) + b$, where $\bar{z} = (2z - c)/5$, c is the center of the region A, $b = 1/4$, and $a = 19/16^2$;



(a) $G(\mathbf{a})$ (b) $f_2(\mathbf{a})$ with $\omega = 1$

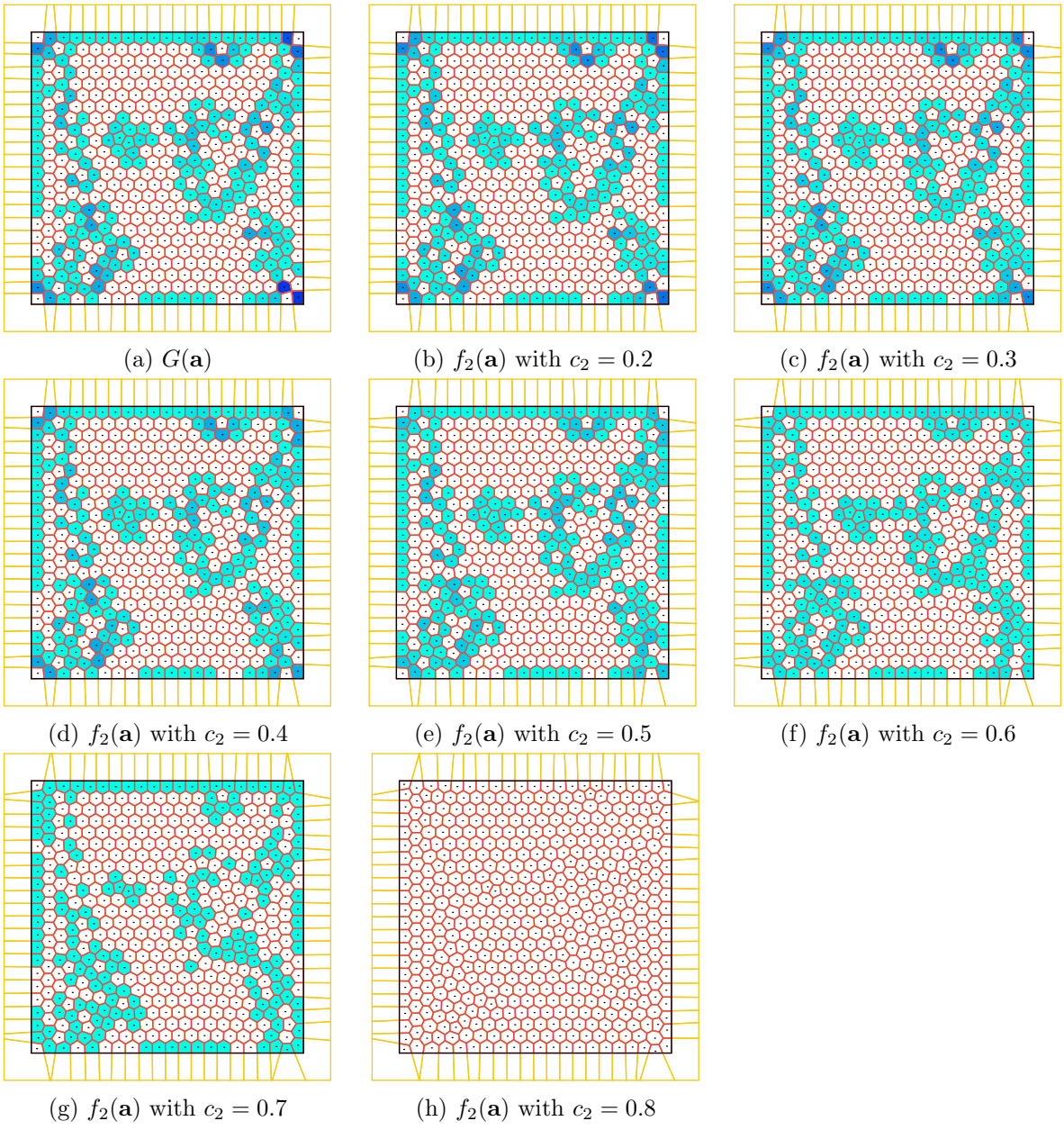
(c) $f_2(\mathbf{a})$ with $\omega = 0.1$ (d) $f_2(\mathbf{a})$ with $\omega = 0.01$

Figure 5: Centroidal Voronoi tessellation with $\kappa_0 = 10$. In (a) we show the result of minimizing $G(\mathbf{a})$. In (b-d) we show the result of minimizing $f_2(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^2(\mathbf{a})$ with $c_2 = 0.5$ and $\omega \in \{1, 0.1, 0.01\}$, respectively.

(b) $\psi(z) = \psi_2(z) := 0.1 + \frac{2.9}{\delta^2}\left(z_2 - 0.6\delta \sin(\frac{2\pi z_1}{\sqrt{\kappa_0}}) - \delta\right)^2$, where $\delta = \frac{\sqrt{\kappa_0}}{2}$;

(c) $\psi(z) = \psi_3(z) := 0.01 + 20\|z - c\|^2/r^2$, where c represents the center and r represents the radius of the circle that circumscribes the region A.

Table 9 shows the details and Figure 9 illustrates the solutions found for varying values of $\omega \in \{1, 0.1, 0.01\}$. As shown in the table, for certain cases, the method terminated because the norm of the merit function gradient reached the target value. In the remaining cases, the method was halted due to "lack of progress", meaning it continued to run as long as improvements in the objective function were observed, and stopped when no further progress was made; see [17] for details. This is not a concern in practical applications, and this stopping criterion is equally valid as any other, as the tolerance value $\varepsilon_{\text{opt}} = 10^{-8}$ used for stopping based on the gradient rule is chosen arbitrarily. In general, the figures in the table show that when $\omega = 0.01$, the value



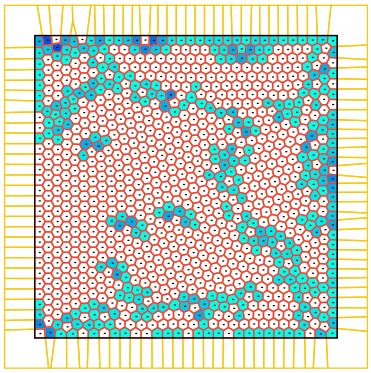
(a) $G(\mathbf{a})$ (b) $f_2(\mathbf{a})$ with $c_2 = 0.2$ (c) $f_2(\mathbf{a})$ with $c_2 = 0.3$

(d) $f_2(\mathbf{a})$ with $c_2 = 0.4$ (e) $f_2(\mathbf{a})$ with $c_2 = 0.5$ (f) $f_2(\mathbf{a})$ with $c_2 = 0.6$

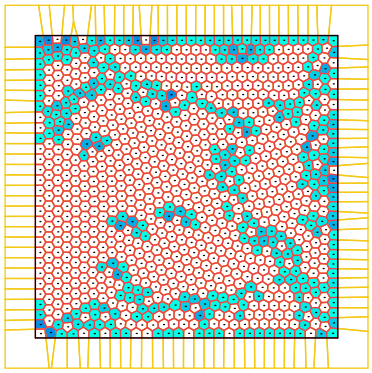
(g) $f_2(\mathbf{a})$ with $c_2 = 0.7$ (h) $f_2(\mathbf{a})$ with $c_2 = 0.8$

Figure 6: Centroidal Voronoi tessellation with $\kappa_0 = 500$. In (a) we show the result of minimizing the function $G(\mathbf{a})$. Cells with darker shading indicate a greater disparity in edge sizes. In (b-h), preserving what colors mean, we present the resulting diagrams by minimizing f_2 with $\omega = 0.01$ and varying $c_2 \in \{0.2, 0.3, \ldots, 0.8\}$.

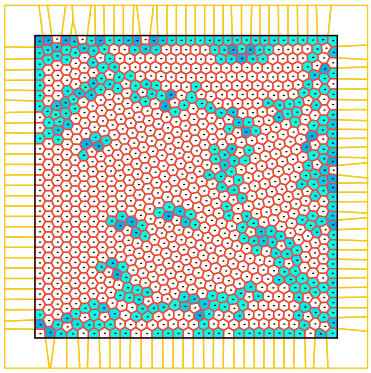
of $G(\mathbf{a}^\star)$ is of the same order as the value we found when minimizing G alone, i.e., 1.62E-01; see Table 2. On the other hand, for $\omega = 1$ we found values of $J^3(\mathbf{a}^\star)$ that are between 2 and 3 orders of magnitude smaller than those found with $\omega = 0.01$, with no significant deterioration



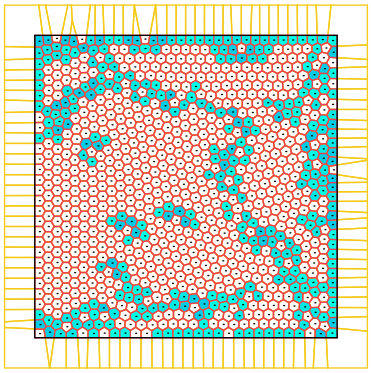
(a) $G(\mathbf{a})$



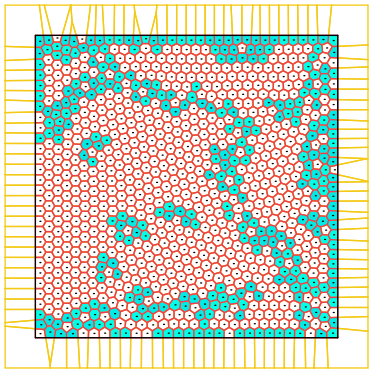
(b) $f_2(\mathbf{a})$ with $c_2 = 0.3$



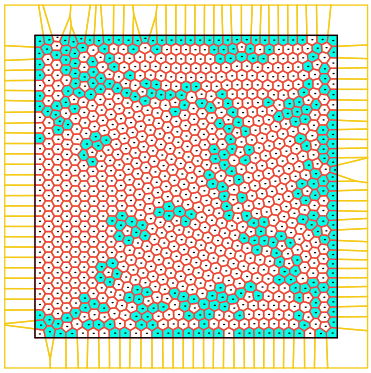
(c) $f_2(\mathbf{a})$ with $c_2 = 0.4$



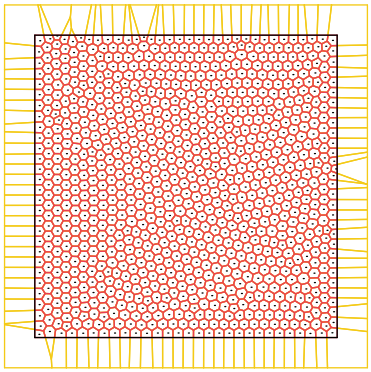
(d) $f_2(\mathbf{a})$ with $c_2 = 0.5$



(e) $f_2(\mathbf{a})$ with $c_2 = 0.6$



(f) $f_2(\mathbf{a})$ with $c_2 = 0.7$



(g) $f_2(\mathbf{a})$ with $c_2 = 0.8$

Figure 7: Centroidal Voronoi tessellations with $\kappa_0 = 1000$. In (a) we show the result of minimizing $G(\mathbf{a})$. Cells with darker shading indicate a greater disparity in edge sizes. In (b-g), preserving what colors mean, we present the diagrams that result from minimizing $f_2(\mathbf{a})$ with $\omega = 0.01$ and varying $c_2 \in \{0.3, 0.4, \ldots, 0.8\}$.

in the value of $G(\mathbf{a}^\star)$. Graphically, when ω is "small" cells tend to have more uniform areas, whereas for larger values of ω we observe cells with different areas. The case of ψ_3 is a little different from the other two, and what we just mentioned would be better observed considering

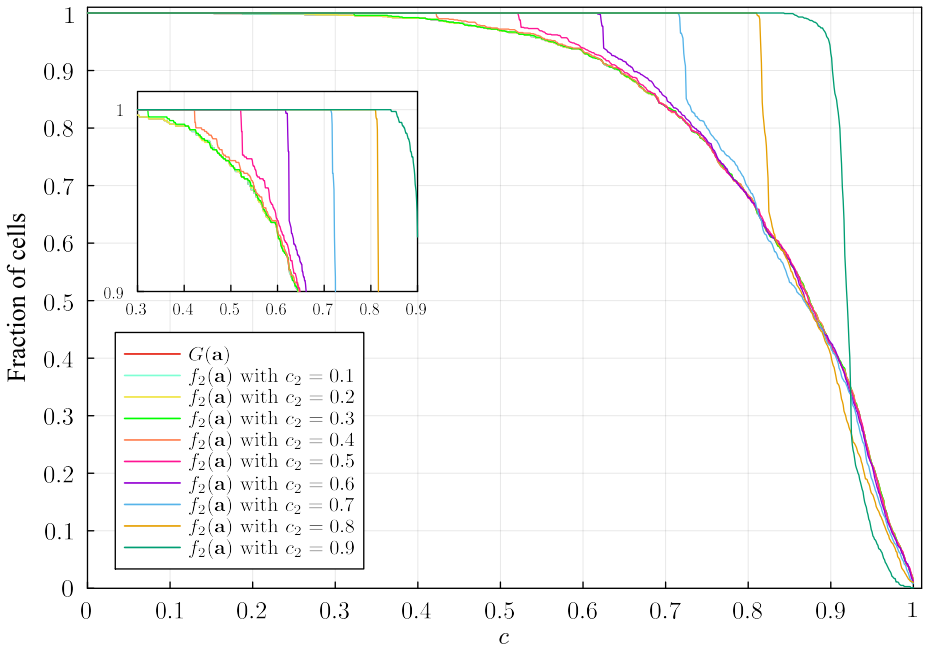
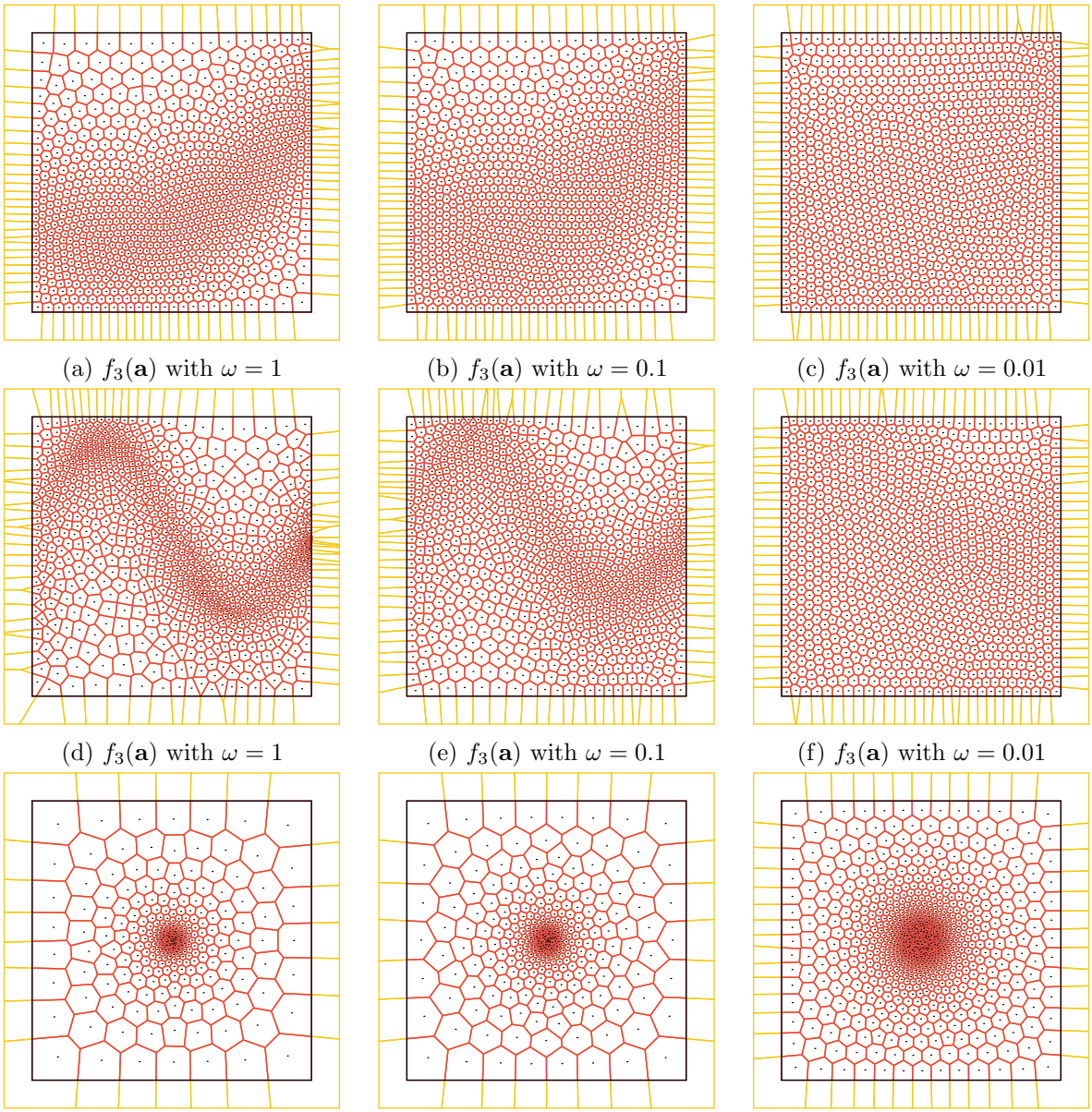


Figure 8: This figure compares solutions obtained by minimizing G alone and the combined function $f_2(\mathbf{a}) = \omega\, G(\mathbf{a}) + J^2(\mathbf{a})$ with $\omega = 0.01$, using $\kappa_0 = 1000$ and varying $c_2 \in \{0.1, \ldots, 0.9\}$. For each solution, the figure plots the percentage of cells that meet the condition "all my edges are at least 100% $\times$ c the average size of my edges" as a function of c.

smaller values of ω.

	ω	$f_3(\mathbf{a}^\star)$	$\|\nabla f_3(\mathbf{a}^\star)\|_\infty$	$G(\mathbf{a}^\star)$	$J^3(\mathbf{a}^\star)$	it	fcnt	Time
ψ_1	1	3.00640E−01	5.1E−09	2.75839E−01	2.48011E−02	768	792	2.839
	0.1	2.44043E−01	9.2E−09	2.14370E−01	2.96727E−01	478	490	1.140
	0.01	1.84559E−01	9.8E−09	1.66529E−01	1.80295E+00	337	355	0.835
ψ_2	1	5.45833E−01	8.5E−04	3.17750E−01	2.28084E−01	51	375	1.395
	0.1	2.75867E−01	1.6E−04	2.29055E−01	4.25556E−01	89	568	1.317
	0.01	1.88113E−01	1.8E−05	1.66878E−01	2.12344E+00	155	1281	2.845
ψ_3	1	2.07753E+00	7.2E−05	1.99442E+00	8.31023E−02	678	807	3.187
	0.1	1.82859E+00	3.8E−03	1.59789E+00	1.20786E+00	1396	1521	3.638
	0.01	8.37667E−01	9.4E−09	5.18745E−01	3.18922E+01	1990	2067	4.640

Table 9: Details of the process of minimizing $f_3(\mathbf{a}) = G(\mathbf{a}) + \omega\, J^3(\mathbf{a})$, considering the three different functions $\psi(z)$ and varying ω.



(a) $f_3(\mathbf{a})$ with $\omega = 1$ (b) $f_3(\mathbf{a})$ with $\omega = 0.1$ (c) $f_3(\mathbf{a})$ with $\omega = 0.01$

(d) $f_3(\mathbf{a})$ with $\omega = 1$ (e) $f_3(\mathbf{a})$ with $\omega = 0.1$ (f) $f_3(\mathbf{a})$ with $\omega = 0.01$

(g) $f_3(\mathbf{a})$ with $\omega = 1$ (h) $f_3(\mathbf{a})$ with $\omega = 0.1$ (i) $f_3(\mathbf{a})$ with $\omega = 0.01$

Figure 9: Centroidal Voronoi tessellation with $\kappa_0 = 1000$ constructed by seeking cells of different prescribed sizes, considering the three different functions $\psi(z)$. Pictures (a-c) correspond to ψ_1, (d-f) to ψ_2, and (g-i) to ψ_3.

4 Conclusions

In this work, we investigated the construction of centroidal Voronoi tesselations with geometric constraints. We have applied a specific case of the theory developed in [3], which provides a

sensitivity analysis for Voronoi diagrams. An advantage of this approach is its unified treatment of interior and boundary edges and vertices. This analysis, which enables the computation of the derivative of any differentiable function depending on the Voronoi diagram, relies on standard nondegeneracy assumptions about the geometry.

The resulting optimization problems were easily solved with a standard optimization method, L-BFGS-B, because two arbitrary choices simplified them. The first choice was to consider a constant density function, which allowed the integral to be computed with high accuracy, without incorporating noise in the evaluation of the objective function and its derivatives. The second choice was to define a square domain A, which implied bound-constrained minimization problems. However, the construction of CVTs has a wide range of applications in which domains extend beyond simple box-shaped regions. The simplest case beyond rectangular domains is to consider a convex set. In this case, methods based on gradient projection would be an alternative. However, CVTs are not well defined when two or more sites coincide, and the projection operation tends to construct such points. In any case, with domains given by convex regions or more complex regions, optimization methods for general nonlinear programming (NLP) problems would be required. The problems considered in the present work, extended to arbitrary domains, represent an interesting benchmark set for existing NLP methods. In the same vein, the problems in the present work, with the inclusion of non-constant density functions, are a challenge for existing methods that handle noise evaluations of the function to be minimized and/or its derivatives.

The numerical results indicate that it is possible to optimize the geometric features of CVTs while maintaining the centroidal property to a reasonable extent. In the numerical experiments, we have considered a single additional geometric constraint; if several simultaneous constraints are desired, a multi-objective optimization approach should be considered. A natural extension of this work is its application to large grids and surface grids [9]. Other research directions include exploring alternative mesh quality criteria to improve the convergence of finite-difference operators, for example, by minimizing the distance between the midpoint of a cell edge and the intersection points of grid segments with the cell edge, see [11]. For three-dimensional problems, the theoretical framework established in [3] must first be extended to three dimensions.

Ethics approval and consent to participate: not applicable.

Consent for publication: not applicable.

Funding: This work has been partially supported by the Brazilian agencies FAPESP (grants 2013/07375-0, 2022/05803-3 and 2023/08706-1) and CNPq (grant 302073/2022-1).

Availability of data and materials: The datasets generated and/or analyzed during the current study are available at `http://ime.usp.br/~egbirgin/`.

Competing interests: The authors declare that they have no competing interests.

Authors' contributions: All authors contributed equally to all phases of the development of this work.

Acknowledgements: not applicable.

References

[1] E. G. Birgin, A. Laurain, R. Massambone, and A. G. Santana. A shape optimization approach to the problem of covering a two-dimensional region with minimum-radius identical balls. *SIAM Journal on Scientific Computing*, 43(3):A2047–A2078, 2021.

[2] E. G. Birgin, A. Laurain, R. Massambone, and A. G. Santana. A shape-Newton approach to the problem of covering with identical balls. *SIAM Journal on Scientific Computing*, 44(2):A798–A824, 2022.

[3] E. G. Birgin, A. Laurain, and T. C. Menezes. Sensitivity analysis and tailored design of minimization diagrams. *Mathematics of Computation*, 92(344):2715–2768, 2023.

[4] B. Bogosel, G. Buttazzo, and E. Oudet. On the numerical approximation of Blaschke–Santaló diagrams using centroidal Voronoi tessellations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 58(1):393–420, 2024.

[5] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

[6] Q. Du and M. Emelianenko. Acceleration schemes for computing centroidal Voronoi tessellations. *Numerical Linear Algebra with Applications*, 13(2-3):173–192, 2006.

[7] Q. Du, V. Faber, and M. Gunzburger. Centroidal Voronoi tessellations: Applications and algorithms. *SIAM Review*, 41(4):637–676, 1999.

[8] Q. Du, M. Gunzburger, and L. Ju. Advances in studies and applications of centroidal Voronoi tessellations. *Numerical Mathematics: Theory, Methods and Applications*, 3(2):119–142, 2010.

[9] Qiang Du, Max D. Gunzburger, and Lili Ju. Constrained centroidal Voronoi tessellations for surfaces. *SIAM Journal on Scientific Computing*, 24(5):1488–1506, 2003.

[10] J. C. Hateley, H. Wei, and L. Chen. Fast methods for computing centroidal Voronoi tessellations. *Journal of Scientific Computing*, 63(1):185–212, 2015.

[11] R. P. Heikes, D. A. Randall, and C. S. Konor. Optimized icosahedral grids: Performance of finite-difference operators and multigrid solver. *Monthly Weather Review*, 141(12):4450–4469, 2013.

[12] M. Iri, K. Murota, and T. Ohya. A fast Voronoi-diagram algorithm with applications to geographical optimization problems. In P. Thoft-Christensen, editor, *System Modelling and Optimization*, pages 273–288, Berlin, Heidelberg, 1984. Springer Berlin Heidelberg.

[13] 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 Transactions on Graphics*, 28(4):Article 101, 2009.

[14] S. P. Lloyd. Least squares quantization in PCM. *Institute of Electrical and Electronics Engineers. Transactions on Information Theory*, 28(2):129–137, 1982.

[15] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, Berkeley, CA, 1967. University of California Press.

[16] J. L. Morales and J. Nocedal. Remark on "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization". *ACM Transactions on Mathematical Software*, 38(1):1–4, 2011.

[17] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software*, 23(4):550–560, 1997.