GEOMETRIC SELF-ASSEMBLY OF RIGID SHAPES: A SIMPLE VORONOI APPROACH*

LISA J. LARSSON†, RUSTUM CHOKSI‡, AND JEAN-CHRISTOPHE NAVE‡

Abstract. Self-assembly of shapes from spheres to nonsmooth and possibly nonconvex shapes is pervasive throughout the sciences. These arrangements arise in biology for animal flocking and herding, in condensed matter physics with molecular and nano self-assembly, and in control theory for coordinated motion problems. While idealizing these often nonconvex objects as points or spheres aids in analysis, the effects of shape curvature and convexity are often dramatic, especially for short-range interactions. In this paper, we develop a general-purpose model for arranging rigid shapes in Euclidean domains and on flat tori. The shapes are arranged optimally with respect to minimization of a geometric Voronoi-based cost function which generalizes the notion of a centroidal Voronoi tessellation from point sources to general rigid shapes. Building upon our previous work in [L. J. Larsson, R. Choksi, and J.-C. Nave, *SIAM J. Sci. Comput.*, 36 (2014), pp. A792–A827], we present an efficient and fast algorithm for the minimization of this nonlocal, albeit finite-dimensional variational problem. The algorithm applies in any space dimension and can be used to generate self-assemblies of any collection of nonconvex, piecewise smooth shapes. We also provide a result which supports the intuition that self-assembled shapes should be centered in and aligned with their Voronoi regions.

Key words. self-assembly, rigid shapes, generalized Voronoi regions, centroidal Voronoi tessellation (CVT), distance functions, energy minimization, L-BFGS quasi-Newton method, Wasserstein distance, generalized Lloyd's method

AMS subject classifications. 65D18, 65K10, 68U05, 49M15

DOI. 10.1137/15M1034167

1. Introduction. Self-assembly, a process whereby a disordered system of components forms an organized, structured pattern solely as a consequence of particle interactions, is both ubiquitous in nature and important for the synthesis of designer materials. Self-assembly of structures is observed across the entire spectrum of length scales [67], from the microscopic to the macroscopic. Examples include molecular self-assembly [68], self-assemblies of nano particles [31, 32] (in particular, block copolymers [5, 28]), animal flocking, swarming, and herding [16, 17], and problems concerning the control of coordinated motion [2]. Models for such self-assemblies can be based on molecular dynamics [63], statistical physics [37], or variational problems with competing interactions [6, 56].

Many of these models are based upon short- and long-range interactions among the particle elements, and these interactions result in a propensity for structures to rearrange themselves subject to the confinements of the physical domain. The precise nature of the rearrangement is dictated by both the form of the interactions (e.g., Coulombic) and the geometric characteristics of the physical domain. Regardless of whether the interactions are modeled variationally or dynamically, the associated

nonlocal problems are analytically and computationally challenging and range from classical and quantum N-body problems to variational aspects of crystallization [7].

In this article we consider a simplified, finite-dimensional model for self-assembly of a finite number of rigid shapes. The interactions in our nonlocal model are all implicitly described via the distance functions and Voronoi regions of the shapes. If the shapes were points, then a standard, geometric approach would be to arrange them so that they form a *centroidal Voronoi tessellation* (cf. [23, 24]). Such centroidal Voronoi tessellations (CVTs) have a variational framework, and our model can be viewed as the generalization to rigid shapes. As with the CVT functional for points, our model favors a certain "centering" of the rigid shape within its Voronoi region. Specifically, we consider a given finite collection of shapes S_i in a convex, bounded domain $\Omega \subset \mathbb{R}^N$. These shapes, which can be thought of as compact submanifolds of codimension $1, \ldots, N$, can be moved by translation or rotation. The model is based on minimization of a geometric cost functional (energy), defined in (2.1), with respect to translation and rotation parameters. We refer to any critical point of the functional as an optimal arrangement (cf. Figure 1.1) and associate our notion of "self-assembly" with gradient descent. While the model can naturally be viewed as the generalization of CVT for shapes, it can also be directly motivated (cf. section 7) via shape interactions involving the 2-Wasserstein metric. Wasserstein metrics are widely used in the active field of optimal transport [66] and have recently appeared in models for pattern formation [8, 9].
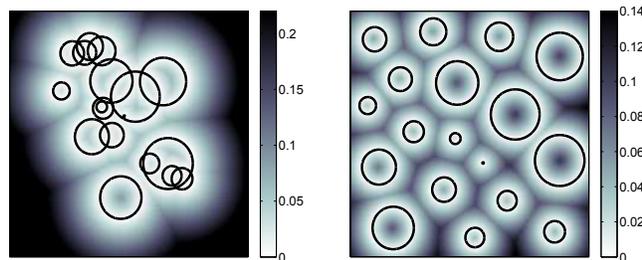
Fig. 1.1. Left: *Initial configuration of circles.* Right: *An optimal arrangement where the circles are arranged according to the cost functional* (2.1). *The distance to the closest circle is shown in gray.*

The primary goal of this paper is to study this finite-dimensional, nonlocal variational model by designing an efficient algorithm for its minimization. We also show how the model lends itself to a generalization of the celebrated Lloyd's method for point CVTs. As mentioned above, our model is a natural extension of the standard variational approach to CVT for points and is valid in any space dimension. Previous work has extended the CVT problem for line segments and other simple shapes with explicit distance functions [48, 57]. Our work is the first to consider the variational problem for general shapes in arbitrary dimensions and to show how its nonlocal structure is amenable to a flexible, fast algorithm for direct energy minimization.

2. The model. Consider arranging n shapes in a bounded, N-dimensional convex domain Ω. We assume that there is some continuous probability density function over our domain, $\rho \in L^1(\Omega)$. Areas of high density will cause shapes to cluster. We will denote the shapes by $S_1, \ldots, S_n$. Although many of the concepts presented here are well-defined for general subsets of Ω, we will assume throughout that the shapes S_i are piecewise-smooth compact manifolds embedded in Ω of codimension between 1

and N, inclusive. For instance, in $\mathbb{R}^2$, the shapes can be points and curves, and in $\mathbb{R}^3$, they can be points, curves, and surfaces. The *Voronoi* (or *generalized Voronoi*) *region* corresponding to shape S_i is the region of Ω that is closer to shape S_i than to any other shape. In order to achieve disjoint regions which partition Ω, we need a tie-breaking rule for points which are equidistant to two or more generators. Hence define

$$V_i := \left\{\mathbf{x} \in \Omega \ \middle|\ d(\mathbf{x}, S_i) < d(\mathbf{x}, S_j) \ \forall j \neq i\right\} \bigcup \left\{\mathbf{x} \in \Omega \ \middle|\ d(\mathbf{x}, S_i) = d(\mathbf{x}, S_j) \ i < j\right\},$$

where $d(\mathbf{x}, S_i) := \inf_{\mathbf{y} \in S_i} |\mathbf{x} - \mathbf{y}|$, the Euclidean distance from $\mathbf{x}$ to the closest point in the set S_i.

To model the rigid motion of the shapes, we parameterize each shape by its centroid and angles of rotation. We can then define our cost function in terms of this parametrization and study how it behaves under translations and rotations. Let $\mathbf{x}_i$ be the $N \times 1$-dimensional vector describing the centroid of shape S_i, and $\boldsymbol{\alpha}_i$ the $\binom{N}{2} \times 1$-dimensional vector of angles that describe the rotation of S_i about its center of mass. Moreover, let $\mathbf{X}$ denote the vector of stacked center locations for all n shapes, and $\boldsymbol{\alpha}$ the vector of stacked angles, while $x_i^{(k)}$ denotes the kth component of the translation vector for S_i, and similarly for $\alpha_i^{(k)}$. We denote the associated placement of shape S_i in space as given by $S_i(\mathbf{x}_i, \boldsymbol{\alpha}_i)$. We define the cost function that returns low values when the shapes are "well centered" within their Voronoi regions: Given a fixed integer $p \geq 1$ and letting $(\mathbf{X}, \boldsymbol{\alpha}) = (\mathbf{x}_1, \ldots, \mathbf{x}_n, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_n)$ be such that each $S_i(\mathbf{x}_i, \boldsymbol{\alpha}_i) \subset \overline{\Omega}$, we define the *arrangement energy (or cost)* of $S_1, \ldots, S_n$ by

$$(2.1) \qquad F(\mathbf{X}, \boldsymbol{\alpha}) := \sum_{i=1}^{n} \int_{V_i} d^p\big(\mathbf{y}, S_i(\mathbf{x}_i, \boldsymbol{\alpha}_i)\big) \rho(\mathbf{y}) \, d\mathbf{y}.$$

Note here that each V_i depends not only on the the parameters $\mathbf{x}_i$ and $\boldsymbol{\alpha}_i$, but also on the parameters of all neighboring Voronoi regions, $\mathbf{x}_j$ and $\boldsymbol{\alpha}_j$. This energy can be written without explicit mention of the Voronoi regions as

$$F(\mathbf{X}, \boldsymbol{\alpha}) = \int_{\Omega} d^p\Big(\mathbf{y}, \bigcup_{i=1}^{n} S_i(\mathbf{x}_i, \boldsymbol{\alpha}_i)\Big) \rho(\mathbf{y}) \, d\mathbf{y}.$$

We choose the form (2.1), as it both provides the right motivation and interpretation of the energy and, moreover, is directly linked to the analysis and computation of the associated variational problem (2.2).

When the shapes S_i are points and p is taken to be 2, criticality of the energy with respect to perturbations of $\mathbf{X}$ leads to the following geometric criterion: The points should be at the centroids of the Voronoi regions they generate [23]. Such a critical collection of points gives rise to what is known as a *centroidal Voronoi tessellation* (CVT) of the domain Ω (illustrated in Figure 2.1). This geometric characterization of critical points of the energy leads to a simple iterative algorithm for computing minimizers known as *Lloyd's method* [47, 23]. It is based upon an iterated step: project each point to the center of mass of its Voronoi region and then recompute the Voronoi regions for the new point set. The optimal arrangement is obtained at a fixed point of this projection. The globally optimal point configuration in $\mathbb{R}^2$ gives rise to regular hexagonal Voronoi regions [27, 53]; in $\mathbb{R}^3$, the dual of the body-centered cubic lattice has been shown numerically to give the lowest arrangement cost [25].

We now postulate a self-assembly of shapes S_i in Ω by minimizing F over all $(\mathbf{X}, \boldsymbol{\alpha})$ such that $S_i(\mathbf{x}_i, \boldsymbol{\alpha}_i) \subset \overline{\Omega}$, $i = 1, \ldots, n$. We call $S_1(\mathbf{x}_1, \boldsymbol{\alpha}_1), \ldots, S_n(\mathbf{x}_n, \boldsymbol{\alpha}_n)$ an
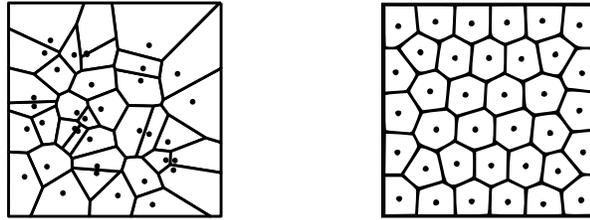
FIG. 2.1. *Left:* *The Voronoi diagram (or Voronoi tessellation) associated with a collection of points.* *Right:* *An arrangement of points which generated a centroidal Voronoi tessellation.*

optimal assembly (or *arrangement*) if the parameterization $(\mathbf{X}, \boldsymbol{\alpha})$ lies at a critical point of the arrangement cost F. It is a *ground state assembly* (or *arrangement*) if the arrangement cost F is at its global minimum. Such a minimizer exists due to the continuity of the cost function F (cf. the supplementary material).

For any fixed collection of shapes S_i, the variational problem

(2.2)
$$\min_{(\mathbf{X}, \boldsymbol{\alpha})} F(\mathbf{X}, \boldsymbol{\alpha})$$

is finite-dimensional. Two central issues prevail:

First, to treat this energy either analytically or numerically, we must integrate over the a priori unknown Voronoi regions V_i. Unlike with the case of point generators, the Voronoi regions V_i and the associated *generalized Voronoi diagram*

$$\cup_{i \neq j} \left(\overline{V_i} \cap \overline{V_j} \right) = \cup_{i \neq j} \left(\partial V_i \cap \partial V_j \right)$$

are not simple polygons. Here $\overline{V_i}$ and ∂V_i denote the closure and boundary of the set V_i, respectively. These regions do not have to be convex or simply connected. Numerically finding these regions explicitly, and then integrating over them, is computationally costly (cf. [44]). On the other hand, simple geometric algorithms like Lloyd's method for points [23] do not directly carry over for general shapes.

Second, even with the simple case of points, and after modding out symmetries, the energy landscape of F is highly nonconvex with many local minimizers. This leads to the important question of how one accesses low-energy self-assemblies, bypassing energy barriers.

To address the first point, we make the key observation that in order to pursue gradient descent or quasi-Newton iteration for the minimization of F, one does not need to find (resolve) the Voronoi regions V_i explicitly but rather simply evaluate integrals of certain functions over the V_i. In our previous paper [44], we presented an indirect fast iterative integration algorithm to evaluate such integrals over the Voronoi regions, which bypasses the explicit calculation of the Voronoi regions. Rather, it relied on the iteration of a Markov kernel operator until the input density is accumulated in a neighborhood of the generators. This approach transforms the problem completely: instead of integrating a density over a priori unknown regions, we evolve the input density to be able to integrate over known regions. As we show in section 4, this crucial step will enable us to produce an efficient and fast algorithm to simulate optimal arrangements via the minimization of F. Although algorithms exist for certain shape ansatzes, for example, [35], as far as we know this is the first work to present and implement an efficient algorithm to generate optimal arrangements of general shapes in two and three space dimensions.

In section 6, we use the model to address the analogue of Lloyd's method for shapes by incorporating higher order moments of the generalized Voronoi regions. In section 7, we note how the model can be conveniently rephrased in terms of the Wasserstein distance from optimal transport. We discuss some applications of the model and our algorithm in section 8.

3. The arrangement energy and its derivatives. When the shape S_i undergoes a translation or rotation, its Voronoi region will change. In this way, the arrangement cost function is a nonlinear function of the parametrization, as both the integrand and the domain of integration depend on $(\mathbf{X}, \boldsymbol{\alpha})$. Rigidly perturbing S_i will not only change the Voronoi region V_i, but will in fact change all neighboring Voronoi regions as well.

First we note that the energy F is a continuous function of its arguments $(\mathbf{X}, \boldsymbol{\alpha}) \in \mathbb{R}^{Nn} \times \mathbb{R}^{N^*n}$, where $N^* = \binom{N}{2} = \frac{N(N-1)}{2}$ (cf. the supplementary material). Hence a minimum of F must exist. Specifically, the continuity of F immediately implies the following.

THEOREM 3.1. *Suppose $\Omega \subset \mathbb{R}^N$ is either a bounded convex domain or a flat compact torus, and $\rho \in L^1(\Omega)$. Let the generators $S_1, \ldots, S_n$, $n < \infty$, be parameterized by $\mathbf{X}$ and $\boldsymbol{\alpha}$. Then a minimizer $(\mathbf{X}^*, \boldsymbol{\alpha}^*)$ of the energy F over all $(\mathbf{X}, \boldsymbol{\alpha})$ such that $S_i(\mathbf{x}_i, \boldsymbol{\alpha}_i) \subset \overline{\Omega}$, $i = 1, \ldots, n$, exists.*

Our approach to the minimization of F will be based upon gradient descent. In this section, we compute the necessary partial derivatives of the energy. It is convenient here to make some assumptions on the generating shapes S_i. Let us assume that the S_i are boundaries of simply connected domains in Ω which are Lipschitz and in fact piecewise C^1. It is not hard to see the boundaries of the Voronoi regions V_i will then also be Lipschitz and piecewise C^1. Indeed, the boundary of each V_i consists of a finite number of pieces which are the zero level set of the difference of two distance functions: ϕ_i, the distance to S_i, and ϕ_j, the distance to some neighboring generator S_j.

To compute partial derivatives of the energy,

$$F(\mathbf{X}, \boldsymbol{\alpha}) = \sum_{i=1}^{n} \int_{V_i} d^p(\mathbf{y}, S_i(\mathbf{X}, \boldsymbol{\alpha})) \rho(\mathbf{y}) \, d\mathbf{y},$$

let us focus on the partial derivative with respect to $\mathbf{x}_i^{(k)}$, where $k = 1, \ldots, N$ denotes the kth component of x_i. Note that with the other $\mathbf{x}_j$ ($j \neq i$) fixed, the only dependence of F on $\mathbf{x}_i^{(k)}$ comes from the integrals over V_i and the neighboring Voronoi regions. Let N_i denote the indices of the shapes that are neighbors to S_i. Note that the dependence on $\mathbf{x}_i^{(k)}$ in the integral over V_i comes in the integrand and the region of integration, while for the neighboring $V_j, j \in N_i$, it is only through the regions of integration. In the appendix (cf. the supplementary material), we show that we have sufficient regularity to invoke the Reynolds' transport theorem. In doing so, we find

$$(3.1) \qquad \frac{\partial F}{\partial x_i^{(k)}} = \int_{\partial V_i} d^p(\mathbf{y}, S_i) \left(\frac{\partial \mathbf{y}}{\partial x_i^{(k)}} \cdot \mathbf{n} \right) \rho(\mathbf{y}) \, d\mathcal{S}$$

$$+ \int_{V_i} \frac{\partial}{\partial x_i^{(k)}} d^p(\mathbf{y}, S_i) \, \rho(\mathbf{y}) \, d\mathbf{y}$$

$$(3.2) \qquad + \sum_{j \in N_i} \int_{\partial V_i \cap \partial V_j} d^p(\mathbf{y}, S_j) \left(\frac{\partial \mathbf{y}}{\partial x_i^{(k)}} \cdot (-\mathbf{n}) \right) \rho(\mathbf{y}) \, d\mathcal{S},$$

where $\mathbf{n}$ denotes the outer normal to V_i and $d\mathcal{S}$ denotes the surface measure. By definition, $d(\mathbf{y}, S_i) = d(\mathbf{y}, S_j)$ for $\mathbf{y} \in \partial V_i \cap \partial V_j$, and $\cup_{j \in N_i}(\partial V_i \cap \partial V_j) = \partial V_i$, so the boundary terms (3.1) and (3.2) are equal and opposite. Thus we have

$$(3.3) \qquad \frac{\partial F}{\partial x_i^{(k)}} = \int_{V_i} \frac{\partial}{\partial x_i^{(k)}} d^p(\mathbf{y}, S_i)\, \rho(\mathbf{y})\, d\mathbf{y}.$$

On the other hand, it is useful to have expressions for both $\mathbf{n}$ and the derivatives $\frac{\partial \mathbf{y}}{\partial x_i^{(k)}}$. To this end, note that any point along $\partial V_i \cap \partial V_j$ belongs to the set

$$\{\mathbf{y} \mid d(\mathbf{y}, S_i) - d(\mathbf{y}, S_j) = 0\},$$

and hence the unit outward normal to V_i for $\mathbf{y} \in \partial V_i \cap \partial V_j$ can be expressed as

$$\mathbf{n} = \frac{\nabla d(\mathbf{y}, S_i) - \nabla d(\mathbf{y}, S_j)}{|\nabla d(\mathbf{y}, S_i) - \nabla d(\mathbf{y}, S_j)|}.$$

For any $\mathbf{y} \in \partial V_i \cap \partial V_j$, direct calculation shows that

$$\frac{\partial \mathbf{y}}{\partial x_i^{(k)}} = \frac{1}{\nabla d(\mathbf{y}, S_i) - \nabla d(\mathbf{y}, S_j)} \; \frac{\partial}{\partial y^{(k)}} d(\mathbf{y}, S_i).$$

The derivative of F with respect to $\alpha_i^{(l)}$ is computed as

$$(3.4) \qquad \frac{\partial F}{\partial \alpha_i^{(l)}} = \int_{V_i} \frac{\partial}{\partial \alpha_i^{(l)}} d^p(\mathbf{y}, S_i)\rho(\mathbf{y})\, d\mathbf{y}.$$

In these first derivative calculations the flux (boundary) terms vanished. Hence they consist only of bulk integrals over the Voronoi regions which can be efficiently computed via an indirect kernel iteration method, which was presented in [44] and is discussed here in section 4.2. However, the second derivative calculations do involve nonvanishing flux integrals which are not taken over the full boundary of the Voronoi regions. Indeed, we have

$$\frac{\partial^2 F}{\partial x_i^{(k)} \partial x_i^{(l)}} = \int_{\partial V_i} \frac{\partial}{\partial x_i^{(k)}} d^p(\mathbf{y}, S_i) \left(\frac{\partial \mathbf{y}}{\partial x_i^{(l)}} \cdot \mathbf{n} \right) \rho(\mathbf{y})\, d\mathcal{S} + \int_{V_i} \frac{\partial^2}{\partial x_i^{(k)} \partial x_i^{(l)}} d^p(\mathbf{y}, S_i)\, \rho(\mathbf{y})\, dy,$$

and for $j \in N_i$,

$$(3.5) \qquad \frac{\partial^2 F}{\partial x_i^{(k)} \partial x_j^{(l)}} = -\int_{\partial V_i \cap \partial V_j} \frac{\partial}{\partial x_i^{(k)}} d^p(\mathbf{y}, S_i) \left(\frac{\partial \mathbf{y}}{\partial x_j^{(l)}} \cdot \mathbf{n} \right) \rho(\mathbf{y})\, d\mathcal{S}.$$

All other entries of the Hessian are zero. By replacing x_i with α_i and x_j with α_j above, the Hessian entries for $\boldsymbol{\alpha}$ can be computed. Simplified formulas in the case where the S_i are points can be found in [36, 46].

To finish calculating the energy derivatives (3.3)–(3.4), it only remains to calculate the derivatives of the function $d^p(\mathbf{y}, S_i)$. In some cases, this distance function can be written down and the derivatives explicitly computed. This is the case, for example, when dealing with lines, spheres, ellipsoids, and points. For all other shapes, however, the distance function $d(\mathbf{y}, S_i)$ is not explicitly available. We next describe how the derivative of the squared distance can be computed in terms of the solution of an

Eikonal equation. The Eikonal equation is a nonlinear boundary value problem that solves for the distance from a given zero level set. In particular, given a shape S_i, we solve for ϕ:

$$(3.6) \qquad \begin{cases} |\nabla \phi| \;=\; 1 & \text{in } \Omega \setminus S_i, \\ \phi(S_i) = 0. \end{cases}$$

The solution of this boundary value problem gives the minimum distance[1] to S_i from any point in Ω, that is, $d(\mathbf{y}, S_i) = \phi(\mathbf{y})$. Using this equality, and noting that rotations and translations are isometries, we are able to finish calculating the gradient of the cost function. In $\mathbb{R}^N$, the derivatives of the squared distance with respect to translation can be formulated in terms of the eikonal solution as follows:

$$(3.7) \qquad \frac{\partial d^p(\mathbf{y}, S_i)}{\partial x_i^{(k)}} = -p\, \phi(\mathbf{y})^{p-1} \left(\frac{\partial}{\partial y^{(k)}} \phi(\mathbf{y}) \right).$$

To evaluate the Hessian, the following second derivatives would be necessary:

$$\frac{\partial^2}{\partial x_i^{(k)} \partial x_i^{(l)}} d^p(\mathbf{y}, S_i) = p\,(p-1)\,\phi^{p-2}(\mathbf{y}) \left(\frac{\partial \phi(\mathbf{y})}{\partial y^{(k)}} \right) \left(\frac{\partial \phi(\mathbf{y})}{\partial y^{(l)}} \right) + p\, \phi^{p-1}(\mathbf{y}) \left(\frac{\partial^2 \phi(\mathbf{y})}{\partial y^{(k)} \partial y^{(l)}} \right).$$

In $\mathbb{R}^2$, the derivative with respect to rotation is

$$(3.8) \qquad \frac{\partial d^p(\mathbf{y}, S_i)}{\partial \alpha_i} = p\, \phi(\mathbf{y})^{p-1} \left(\nabla \phi(\mathbf{y}) \cdot \begin{pmatrix} y^{(2)} \\ -y^{(1)} \end{pmatrix} \right).$$

Similarly, the derivatives of the squared distance with respect to rotations in $\mathbb{R}^3$ are given by

$$(3.9) \qquad \frac{\partial d^p(\mathbf{y}, S_i)}{\partial \boldsymbol{\alpha}_i} = p\, \phi(\mathbf{y})^{p-1} \begin{pmatrix} 0 & -y^{(3)} & y^{(2)} \\ y^{(3)} & 0 & -y^{(1)} \\ -y^{(2)} & y^{(1)} & 0 \end{pmatrix} \nabla \phi(\mathbf{y}).$$

4. Direct energy minimization. For the case of points, there is a simple geometric consequence of criticality of the energy, which leads to Lloyd's method. For general shapes, such a characterization is not so straightforward (cf. section 6). Moreover, Lloyd-type algorithms converge only linearly. Hence, in this section we proceed with direct energy minimization via a quasi-Newton method. While at first sight this might seem straightforward for this finite-dimensional problem, it presents certain challenges, which we now summarize.

First, we must compute the level set (distance) function for the generators if an explicit formula does not exist.

Next, the integrals associated with these gradients are defined over a priori unknown sets (Voronoi regions). Regardless of an explicit or numerically generated computation of the integrands in the first-order gradients, these integrals are defined over a priori unknown sets. To first resolve these regions at every step and then integrate is computationally intensive. Luckily, these a priori unknown sets do not need to be computed explicitly: rather, one only needs a quick and efficient way of

---

[1]In the literature, often the signed distance function is considered. Note that the solution to (3.6) gives the unsigned distance function.

computing integrals over the Voronoi regions, and such a method is the centerpiece of our earlier work [44]. For completeness, we describe this method below in section 4.2.

Finally, while our indirect integration algorithm will allow us to quickly compute gradients, the second derivatives include flux integrals over certain parts of the boundary of the Voronoi regions. Hessian calculations are thus costly, and quasi-Newton methods are preferable over direct Newton methods. To this end, we invoke the limited memory BFGS method (L-BFGS) described below in section 4.3.

4.1. Computation of the distance function. The integrands of the arrangement cost derivatives $\partial F/\partial x_i^{(k)}$ and $\partial F/\partial \alpha_i^{(l)}$ are given in terms of the distance function $d(\mathbf{y}, S_i)$ and its derivatives (see (3.3)–(3.4) and (3.7)–(3.9)). The minimal distance function to a shape is only given explicitly in very special cases. Computationally, the function $d(\mathbf{y}, S_i)$ can be obtained as the solution of the eikonal equation (3.6). To obtain $d(\mathbf{y}, \cup_{i=1}^n S_i)$, the distance from any point $\mathbf{y}$ in the domain Ω to the closest point on any shape, one can simply solve the eikonal equation with boundary condition $\phi(\cup_{i=1}^n S_i) = 0$.

To solve the eikonal equation numerically, a fast marching method (FMM) [61, 58] or a fast sweeping method (FSM) [69, 64] can be used. Whereas the FMM uses the characteristics of the nonlinear eikonal equation explicitly, the FSM leverages the fact that the upwind finite difference scheme ensures information propagates correctly along characteristics to obtain a slightly better overall complexity. An FSM iterates through all the grid points in alternating directions, updating the solution at each point as the minimum of the current and previous solution values. After a number of these "sweeps," a stable solution is obtained. The complexity of this algorithm given m grid points is $\mathcal{O}(m)$. For the numerical results generated in this paper, a first-order FSM was used.

The gradient of the solution to the eikonal equation can be computed using upwind finite differences, and one-sided finite differences at the boundary of the domain. To ensure the correct derivative is taken near the shocks (the nonsmooth points of the distance function), we take the largest of the two possible one-sided differences (in absolute value).

4.2. Integration over Voronoi regions: The indirect iterative approach. Obtaining an optimal placement of shapes reduces to efficient and accurate integration over generalized Voronoi regions. As we mentioned in [44], this problem is nontrivial, and a direct approach whereby one first computes the Voronoi regions is computationally costly. Rather, in [44] we present and analyze an algorithm to integrate directly, without explicitly calculating the generalized Voronoi diagram. Let us recap the essential features and order of our algorithm. Given a collection of shapes S_i which are distributed in Ω (we suppress here the argument $(\mathbf{x}_i, \boldsymbol{\alpha}_i)$ for S_i) and some function $f(\mathbf{y})$ defined on Ω, our goal is to compute the integral

$$(4.1) \qquad \int_{V_i} f(\mathbf{y}) \, d\mathbf{y}$$

without first computing V_i, the Voronoi region of S_i. The algorithm is based upon the following four steps:

(i) Using an FSM [69], solve the eikonal equation for $\phi : \Omega \to \mathbb{R}$, with boundary condition $\phi(\cup_{i=1}^n S_i) = 0$ (see (3.6)).

(ii) With the eikonal solution ϕ, construct a kernel $k(\mathbf{x}, \mathbf{y})$ and an operator:

$$P[f](\mathbf{x}) = \int_\Omega k(\mathbf{x}, \mathbf{y}) f(\mathbf{y}) \, d\mathbf{y}.$$

The operator P moves "mass" (given by f) toward the closest shape. To obtain the kernel k, first compute

$$\widetilde{k}(\mathbf{x}, \mathbf{y}) := \begin{cases} (\phi(\mathbf{x}) - \phi(\mathbf{y}))_+ & \text{if } |\mathbf{x} - \mathbf{y}| \leq h, \\ 0 & \text{otherwise.} \end{cases}$$

Then k is the normalization of $\widetilde{k}$ such that k integrates to one in $\mathbf{x}$. This normalization preserves the L^1 norm of f: $\int_\Omega P[f](\mathbf{x}) \, d\mathbf{x} = \int_\Omega f(\mathbf{y}) \, d\mathbf{y}$. See [44] for details.

(iii) Then iterate the operator until the support of $P^m[f](\mathbf{x})$ is concentrated in an h-neighborhood of each shape, and integrate over these (a priori known) regions to obtain solutions to (4.1) for $i = 1, \ldots, n$.

(iv) Numerically, a finite volume method is used to discretize the operator P based upon a spatial discretization of size h, and a summation along the grid points closest to each shape S_i will yield the solution to (4.1).

This algorithm is a discrete-space analogue of evolving f in the direction of the negative eikonal gradient. In [44], we analyzed the theoretical and numerical properties of this algorithm: in particular, we proved the algorithm was first order in h and presented several convergence results.

4.3. Quasi-Newton methods for CVT energy minimization. To compute the critical points of the arrangement cost (2.1), we will use nonlinear solvers that find $(\mathbf{X}^*, \boldsymbol{\alpha}^*)$ where the energy gradients are zero. For simplicity, we will use $\widetilde{\mathbf{X}}$ to denote the parameter vector (which contains $\mathbf{X}$ and $\boldsymbol{\alpha}$, stacked) in this section. Three natural algorithms used to find these critical points are gradient descent, Newton's method, and quasi-Newton algorithms. Starting at some initial condition $\widetilde{\mathbf{X}}^{(0)}$, gradient descent takes steps in the direction of the negative gradient and terminates when the norm of the gradient is sufficiently small. The descent step is

$$\widetilde{\mathbf{X}}^{(k+1)} = \widetilde{\mathbf{X}}^{(k)} - \gamma \nabla F(\widetilde{\mathbf{X}}^{(k)}),$$

where $\gamma > 0$ is a step length chosen using line search algorithms such as the one proposed in [52]. The advantage of using this method is that it is straightforward to implement, the disadvantage being that it converges linearly to a critical point.

To achieve quadratic convergence to a critical point, Newton's method is used. This method requires not only gradient information but also Hessian information, as it relies on a locally quadratic approximation to the objective function to find a search direction. The Newton update is

$$\widetilde{\mathbf{X}}^{(k+1)} = \widetilde{\mathbf{X}}^{(k)} - \gamma \left(\nabla^2 F(\widetilde{\mathbf{X}}^{(k)}) \right)^{-1} \nabla F(\widetilde{\mathbf{X}}^{(k)}),$$

where $\gamma \in (0, 1)$ is a step length parameter chosen via a line search algorithm. In the current optimization problem, we can efficiently compute the cost gradient, but not the Hessian. The Hessian terms contain surface integrals over the generalized Voronoi regions, which are costly to compute. Moreover, the Hessian has many nonzero elements, which can be inefficient to populate.

To obtain superlinear convergence to a critical point while using only gradient information, quasi-Newton methods have been developed. Quasi-Newton methods use a secant approximation of the Hessian, which requires only gradient information at each step. Under the original BFGS method, an initial guess, $\mathbf{H}^{(0)} \approx \left(\nabla^2 F(\widetilde{\mathbf{X}}^{(0)}) \right)^{-1}$, of the inverse Hessian of the objective function is specified, and subsequent iterations

update the inverse Hessian using the inverse BFGS formula [54, 55]. The BFGS formula for the inverse Hessian is

$$\mathbf{H}^{(k)} = \left(\mathbf{U}^{(k-1)}\right)^{\top} \mathbf{H}^{(k-1)} \mathbf{U}^{(k-1)} + \boldsymbol{\rho}^{(k-1)} \mathbf{s}^{(k-1)} \left(\mathbf{s}^{(k-1)}\right)^{\top},$$

where

$$\mathbf{s}^{(k-1)} := \widetilde{\mathbf{X}}^{(k)} - \widetilde{\mathbf{X}}^{(k-1)}, \qquad\qquad \mathbf{y}^{(k-1)} := \mathbf{G}^{(k)} - \mathbf{G}^{(k-1)},$$

$$\boldsymbol{\rho}^{(k-1)} := 1/\left(\left(\mathbf{y}^{(k-1)}\right)^{\top} \mathbf{s}^{(k-1)}\right), \quad \mathbf{U}^{(k-1)} := \mathbf{I} - \boldsymbol{\rho}^{(k-1)} \mathbf{y}^{(k-1)} \left(\mathbf{s}^{(k-1)}\right)^{\top}$$

such that $\mathbf{G} = \nabla F(\widetilde{\mathbf{X}})$ is the gradient of the arrangement cost, k is the index the iteration of the algorithm, and $\mathbf{I}$ denotes the identity matrix of dimension $n(N + N^*)$.

A limited memory version of BFGS (L-BFGS) [55] was developed for large-scale problems that uses only gradient information from M previous steps to calculate the approximate inverse Hessian. Typically, M is set between 3 and 20 [45]. In the supplementary videos, gradient descent with a small step size was used to illustrate the gradient flow of the arrangement energy. To obtain efficient optimal arrangements, the L-BFGS algorithm was used with the Moré–Thuente line search algorithm [52].

4.4. The full algorithm. To conclude, the energy derivatives from section 3 can be computed using a combination of the FSM [69] and a generalized Voronoi integration algorithm [44]. These derivatives are used by L-BFGS to iteratively compute search directions that efficiently locate optimal arrangements. We summarize the full algorithm in the supplementary material, where we also present some results on convergence and CPU times.

5. Results for shapes in two and three dimensions. Here we present some results of our algorithm for certain rigid shapes in two and three dimensions. In all cases the density ρ is taken to be constant (see the supplementary material for examples with nonuniform densities). We choose to focus on the case $p = 2$ for direct comparison with existing results on CVTs. L-BFGS was run with parameter $M = 6$, stopping criterion $|\nabla F| \leq 5 \times 10^{-4}$, and the Moré–Thuente line search [52]. As we shall see, this stopping criterion is readily achieved with only a few quasi-Newton iterations, typically anywhere between 10 and 30. In each iteration, the energy and energy derivatives were calculated using the integration algorithm in section 4.2 with uniform grid size, typically $h = \frac{1}{151}$ for two dimensions. We present examples for shapes with explicit distance functions (line segments, circles, spheres). Then we present examples of general shapes for which a distance function must be computed via the solution to the eikonal equation (3.6). In all the two-dimensional depictions, the minimum distance function to the shapes is shown in grayscale in the background to illustrate the Voronoi regions. We start with a simple test simulation for points.

5.1. A test case: CVT of points. For point generators, an optimal self-assembly corresponds to a CVT. In this case, it is straightforward to generate the polygonal Voronoi regions (the Voronoi tessellation), and there are many algorithms[2] to compute CVTs. Whereas the computational framework (the eikonal solver of section 4.1 and the integration algorithm of section 4.2) is not needed for point generators,

---

[2]These range from direct quasi-Newton minimization of the arrangement energy (there called the CVT energy), the elegant Lloyd's method (discussed and generalized in section 6), to probabilistic algorithms, such as MacQueen's sampling method [49], and later accelerated and parallel versions [38]. See also [19, 21, 26, 40, 41, 46, 60, 65].

this setting provides a good test case for our method. In Figure 5.1, we run our algorithm on an initial configuration of points in the unit square and present our resulting self-assembled configuration together with the centroids of their associated Voronoi regions. The average distance of these points from the actual Voronoi centroids is 6.50×10^{-4} with a standard deviation of 2.73×10^{-4}.
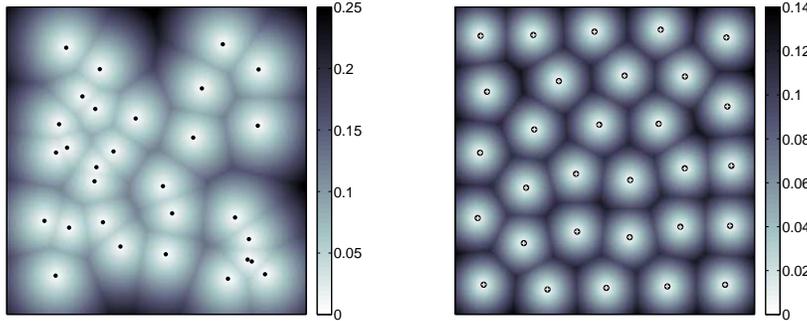
FIG. 5.1. Left: *Initial configuration of points.* Right: *Comparison of the final converged self-assembly of point generators after* 16 *L-BFGS iterations (*black dots*) with the centroids of the associated Voronoi regions (*gray crosses*).*

5.2. Circles, lines, and spheres. In this section, results for generators that have explicit distance functions are presented in $\mathbb{R}^2$ and $\mathbb{R}^3$. These include lines, circles, ellipses, spheres, and ellipsoids. See [43] for exact forms of the respective distance functions and their partial derivatives with respect to position and rotation. Circles of different size in a square were previously presented in Figure 1.1 with the initial and the final converged configuration subject to a stopping criterion. It is natural to ask about the stability of the final configuration in order to verify that it does indeed represent a local minimizer. Whereas the Hessian involves boundary integrals over parts of the Voronoi diagram which our method does not seek to resolve, one can approximate it with finite differences of gradients (the basis for the BFGS method). To this end, we computed the smallest and largest eigenvalues of the Hessian to be 0.0028 and 0.0871, respectively. In principle this computation can be done for any self-assembled arrangement of any shapes to gain numerical support that one has achieved a local minimizer of (2.1).

Figures 5.2 and 5.3 show a self-assembly for an initial configuration of line segments and spheres (see the captions for details).

5.3. General shapes. We present the results of several runs using leaves and animal cartoons of different sizes in two dimensions followed by two three-dimensional examples. These shapes were chosen as simple and readily available examples—in terms of their level sets—of nonconvex shapes. Our first example in Figure 5.4 shows an initial configuration of 18 leaves of various sizes and the final converged self-assembly. A plot of the energy and the energy gradient per iteration is included. Note here that energy minimization does not prevent shapes from overlapping, and converged assemblies can include nested shapes. Our second example in Figure 5.5 shows initial configurations for 60 mixed shapes and the final converged self-assemblies. A three-dimensional example is presented in Figure 5.6, showing nine rabbit-shaped objects in the unit cube. A video of these initializations and assemblies are provided in the supplementary material to visualize these assemblies from various angles.
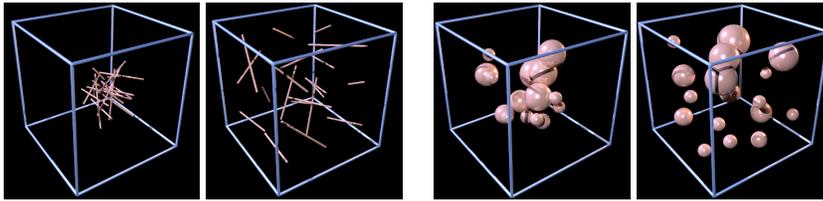
FIG. 5.2. *Left two: Initialization of line segments in* $\mathbb{R}^3$ *and its resulting optimal self-assembly. Right two: Initialization of* 18 *spheres in* $\mathbb{R}^3$ *and its resulting optimal self-assembly.*
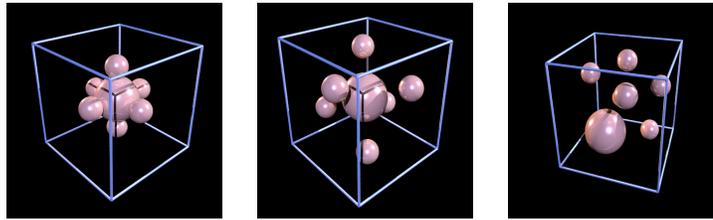
FIG. 5.3. *Left: The initialization of spheres in* $\mathbb{R}^3$. *Middle: A constrained optimal self-assembly wherein the large center sphere is fixed at* $(0.5, 0.5, 0.5)$. *Right: The unconstrained optimal self-assembly.*
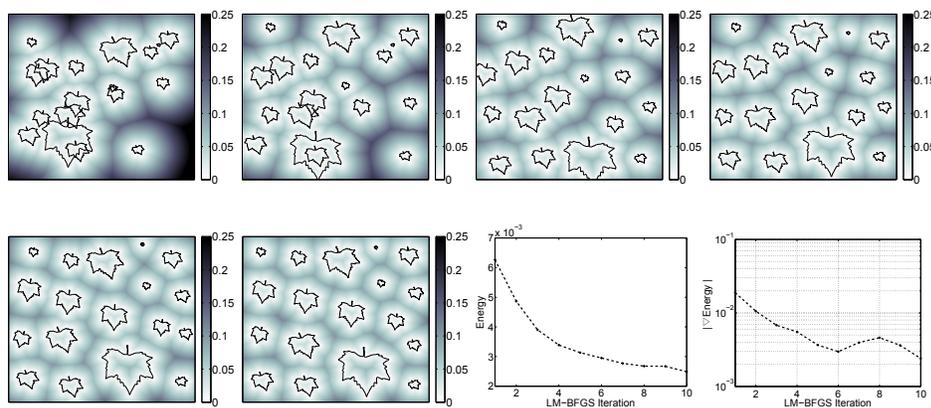
FIG. 5.4. *Top: The initial configuration and the third, fifth, and seventh iterations. Bottom: The ninth and twelfth iterations, the energy per iteration, and* $|\nabla Energy|$ *per iteration (note the vertical axis has been appropriately scaled).*

We make a few concluding remarks concerning these and many other simulations which were performed using our algorithm (cf. the thesis of Larsson [43]). The algorithm converges quickly (with respect to the number of quasi-Newton iterations) for any collection of shapes. Initial configurations can include overlapping shapes. Self-assembled configurations can include nested shapes when sufficient empty space in Ω is lacking. There is strong evidence that the converged configuration is close to an optimal arrangement and in fact a local minimizer of the energy. See section 8.1 for a discussion of the energy landscape. Not surprisingly there are many optimal arrangements depending on the initial configuration of the shapes, and energy barriers are abundant, particularly with respect to rotations. The closer the shapes are, the less they tend to rotate with each quasi-Newton iteration. Optimal arrangements
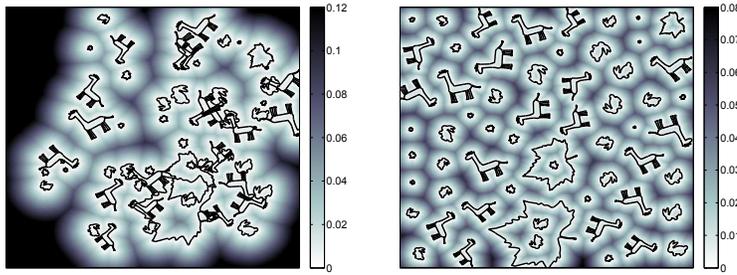
FIG. 5.5. *A mixture of* 60 *shapes.* Left: *Initialization.* Right: *Converged self-assembly after* 40 *L-BFGS iterations.*
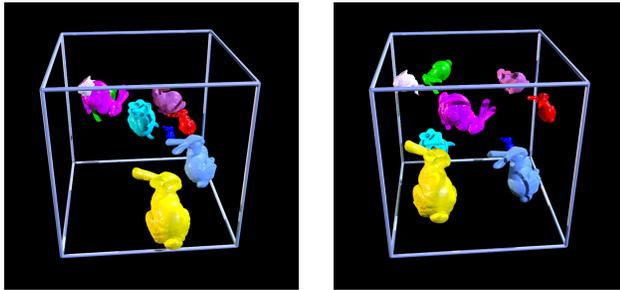
FIG. 5.6. Left: *Initialization of* 9 *bunnies of various sizes in three dimensions.* Right: *Assembly of bunnies.*
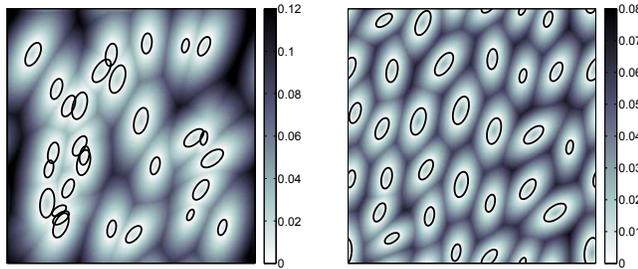
FIG. 5.7. *Periodic boundary conditions.* Left: *An initial configuration of ellipses on the torus.* Right: *The self-assembled arrangement.*

tend to lead to shapes which are both well centered and well oriented in their Voronoi regions. We explore this tendency in section 6.

We also remark that the model and numerical method can readily accommodate periodic boundary conditions. To generate results with periodic boundary conditions, one generator is fixed to break symmetry. To compute the energy gradient, a distance function with periodic boundary conditions was also obtained using the FSM. This can be done by updating boundary points in a periodic way and does not increase the complexity of the algorithm. Then the same first derivatives and optimization algorithms presented in section 4 can be applied. See Figure 5.7 for one example of a periodic optimal arrangement.

6. A generalized Lloyd's approach. Lloyd's method for point generators gives a simple geometric characterization of critical points of the energy for the case of $p = 2$, and leads to an elegant and simple fixed point algorithm to compute optimal

arrangements of points. Recall that the generators $\mathbf{X}$ give rise to a CVT when they are at the centers of mass of the Voronoi regions they generate, and for the case of $p = 2$, this is equivalent to being a critical point of the energy F. Lloyd's method is a fixed point iteration used to arrive at a CVT that successively projects the generators to the centers of mass of the Voronoi cells, then recomputes the Voronoi regions. More precisely, Lloyd's method is a simple iterative method to compute a fixed point of the following map:

$$(6.1) \qquad T_{ik}\left(x_i^{(k)}\right) = \frac{\int_{V_i} y^{(k)} \rho(\mathbf{y})\, d\mathbf{y}}{\int_{V_i} \rho(\mathbf{y})\, d\mathbf{y}} \quad \text{for } k = 1, \ldots, N, \ i = 1, \ldots, n.$$

So in the case $p = 2$ where the shapes are all points, the energy F is minimized when the points are at the centroids of their Voronoi regions. Moreover, this is the geometric criterion that inspired us to use this notion of arrangement energy as the basis for self-assembly of shapes. Hence it is interesting to ask whether a similar geometric intuition holds for optimal self-assemblies of shapes. The answer is yes.

Because the distance to a general shape is not a polynomial expression, the arrangement cost function does not immediately lead to a geometric characterization of the minimizer, as was the case for points. However, one can approximate the squared geometric distance to a shape with an appropriate second-order polynomial function which takes into account rotations of the generators. In principle, a whole family of methods can be derived using higher-order polynomials. For second-order polynomials the natural geometric object to work with is an ellipsoid. To this end, let r_i, b_i, c_i be positive constants for $i = 1, \ldots, n$. Define $\mathbf{A}_i = \operatorname{diag}(1, \frac{r_i}{b_i})$ in $\mathbb{R}^2$ and $\mathbf{A}_i = \operatorname{diag}(1, \frac{r_i}{b_i}, \frac{r_i}{c_i})$ in $\mathbb{R}^3$. In $\mathbb{R}^N$, $\mathbf{A}_i$ is an $N \times N$ diagonal matrix, with positive diagonal entries. The equation of an ellipsoid centered at the origin and aligned with the axes is

$$\left\{ \mathbf{y} \in \mathbb{R}^N \,\middle|\, E(\mathbf{y}) = 0 \right\}, \qquad \text{where } E(\mathbf{y}) := (\mathbf{A}_i \mathbf{y})^\top (\mathbf{A}_i \mathbf{y}) - r_i^2.$$

So, for example, if $N = 2$, $r_i = 2$, $b_i = 1$, then $E(\mathbf{y}) = y_1^2 + 4y_2^2 - 4$. These ellipses approximate the shapes S_i. Given a placement $(\mathbf{X}, \boldsymbol{\alpha}) = (x_1, \ldots, x_n, \alpha_1, \ldots, \alpha_n)$ of the S_i, the approximating ellipsoids are given by

$$\left\{ \mathbf{y} \in \mathbb{R}^N \,\middle|\, E(R_i^{-1}(\mathbf{y} - \mathbf{x}_i)) = 0 \right\},$$

where R_i denotes the rotation matrix parameterized by $\boldsymbol{\alpha}_i$. Let the *algebraic distance*, $d_a(\mathbf{y}, S_i)$, be defined by

$$(6.2) \qquad d_a^2(\mathbf{y}, S_i) := |E(R_i^{-1}(\mathbf{y} - \mathbf{x}_i))|.$$

Note that the algebraic distance is defined explicitly in terms of the parameters $\mathbf{x}_i$ and $\boldsymbol{\alpha}_i$ of the shape S_i and is different from the geometric distance. Indeed, even in the case where the shapes S_i are ellipses, the algebraic distance differs from the geometric.

The algebraic distance leads to the following approximating cost function:

$$(6.3) \qquad G(\mathbf{X}, \boldsymbol{\alpha}) := \sum_{i=1}^n \int_{V_{a,i}} d_a^2(\mathbf{y}, S_i) \rho(\mathbf{y})\, d\mathbf{y}.$$

As the ellipsoid tends to a point, $d_a^2(\mathbf{y}, S_i) \to |\mathbf{y} - \mathbf{x}_i|^2$, and the arrangement cost function is recovered. Note that the Voronoi regions $V_{a,i}$ in (6.3) are computed based on the algebraic distance (6.2). For the approximate energy (6.3), we have the following result.

PROPOSITION 6.1. *Assume that the ellipsoidal approximations to the shapes S_1, $\ldots, S_n$ are disjoint. Let $\mathbf{X}$ be in $\Omega \subset \mathbb{R}^N$ and, for simplicity, let $\rho(\mathbf{y})$ be constant. Let R be a d-dimensional rotation matrix parameterized by $\boldsymbol{\alpha}$. If the following conditions hold at $(\mathbf{X}, \boldsymbol{\alpha})$:*

$$x_i^{(k)} = \frac{\int_{V_{a,i}} y^{(k)} \rho(\mathbf{y}) \, d\mathbf{y}}{\int_{V_{a,i}} \rho(\mathbf{y}) \, d\mathbf{y}} \quad and \quad R_i = \mathbf{U}_i^\top \quad for \ k = 1, \ldots, N,$$

where $\mathbf{U}_i$ is the matrix of eigenvectors of the moment matrix $\mathbf{M}_i = (m_{j,k})_{j,k=1}^N$, given by

$$(6.4) \qquad m_{j,k} = \int_{V_{a,i}} (y^{(j)} - x_i^{(j)})(y^{(k)} - x_i^{(k)}) \rho(\mathbf{y}) \, d\mathbf{y},$$

then $(\mathbf{X}, \boldsymbol{\alpha})$ is a critical point of (6.3).

The first condition on the centroids is also a necessary condition for criticality. Our proof shows that the second moment condition is also necessary for criticality in $N = 2$. We expect that it is also a necessary condition for higher space dimensions as well, but our proof yields only the sufficiency.

Proof. Define $E_i := \{\mathbf{y} \mid E(R_i^{-1}(\mathbf{y} - \mathbf{x}_i)) < 0\}$, which are simply the interiors of the ellipsoids that, by assumption, are nonoverlapping. Hence we note that

$$\widetilde{G}(\mathbf{X}, \boldsymbol{\alpha}) = \sum_{i=1}^n \int_{V_{a,i}} E(R_i^{-1}(\mathbf{y} - \mathbf{x}_i)) \rho(\mathbf{y}) \, d\mathbf{y},$$

$$= G(\mathbf{X}, \boldsymbol{\alpha}) - \underbrace{2 \sum_{i=1}^n \int_{E_i} E(R_i^{-1}(\mathbf{y} - \mathbf{x}_i)) \rho(\mathbf{y}) \, d\mathbf{y}}_{=:K_0}.$$

Since by assumption, ρ is constant, the interior integrals over E_i are independent of $(\mathbf{X}, \boldsymbol{\alpha})$, and hence $\widetilde{G}(\mathbf{X}, \boldsymbol{\alpha})$ and $G(\mathbf{X}, \boldsymbol{\alpha})$ differ by a constant K_0. Therefore, it is sufficient to consider the minimizers of $\widetilde{G}$.

For the proof, we will relax the slaving of the tessellation $\{V_{a,i}\}_{i=1}^n$ to $(\mathbf{X}, \boldsymbol{\alpha})$ and consider the criticality conditions with respect to $\mathbf{X}$ and $\boldsymbol{\alpha}$ at an arbitrary tessellation $\mathcal{V} = \{\widehat{V}_i\}_{i=1}^n$ of Ω into n regions. These will suffice, as fixing $(\mathbf{X}, \boldsymbol{\alpha})$ and minimizing with respect to all tessellations will yield the criticality condition $\mathcal{V} = \{V_{a,i}\}_{i=1}^n$. This step is identical to the case of point generators and was proven in [23]. Thus we consider the energy

$$\mathcal{H}(\mathbf{X}, \boldsymbol{\alpha}, \mathcal{V}) = \sum_{i=1}^n \int_{\widehat{V}_i} E(R_i^{-1}(\mathbf{y} - \mathbf{x}_i)) \rho(\mathbf{y}) \, d\mathbf{y},$$

which coincides with $\widetilde{G}(\mathbf{X}, \boldsymbol{\alpha})$ when $\mathcal{V} = \{V_{a,i}\}_{i=1}^n$.

First fix $\boldsymbol{\alpha}$ and $\mathcal{V}$, and for simplicity denote $\mathcal{H}(\mathbf{X}, \boldsymbol{\alpha}, \mathcal{V})|_{\boldsymbol{\alpha}, \mathcal{V}} = \mathcal{H}(\mathbf{X})$. Let $\varepsilon > 0$, and choose $\mathbf{v} = (v^{(1)}, \ldots, v^{(N)})^\top$ such that $\mathbf{x}_i + \varepsilon \mathbf{v}$ remains in the interior of Ω. Denote by $\mathbf{e}_k$ the vector in $\mathbb{R}^{Nn}$ that is all zeros except in the kth entry. The first variation is then

$$\lim_{\varepsilon \to 0} \frac{\mathcal{H}(\mathbf{X} + \varepsilon \sum_{k=0}^{N-1} \mathbf{e}_{N-k} \, v^{(N-k)}) - \mathcal{H}(\mathbf{X})}{\varepsilon} = \sum_{k=1}^N v^{(k)} \frac{\partial \mathcal{H}(\mathbf{X})}{\partial x_i^{(k)}} = 0.$$

The first variation is zero if all partial derivatives are zero, which in matrix form means

$$\frac{\partial \mathcal{H}(\mathbf{X})}{\partial \mathbf{x}_i} = 2 \int_{\widehat{V}_i} R_i \mathbf{A}_i^\top \mathbf{A}_i R_i^\top (\mathbf{x}_i - \mathbf{y}) \rho(\mathbf{y}) \, d\mathbf{y} = \mathbf{0}.$$

To show that $\int_{\widehat{V}_i} (\mathbf{x}_i - \mathbf{y}) \rho(\mathbf{y}) \, d\mathbf{y} = \mathbf{0}$, it suffices to show that the matrix $R_i^\top \mathbf{A}_i^\top \mathbf{A}_i R_i$ is invertible. Observe:

$$\det(R_i \mathbf{A}_i^\top \mathbf{A}_i R_i^\top) = \underbrace{\det(R_i)}_{=1} \underbrace{\det(\mathbf{A}_i^\top)}_{>0} \underbrace{\det(\mathbf{A}_i)}_{>0} \underbrace{\det(R^\top)}_{=1} > 0.$$

Therefore the kernel of $R_i \mathbf{A}_i^\top \mathbf{A}_i R_i^\top$ is empty, and the location vectors are minimized when placed at the center of mass of the regions $\widehat{V}_i$.

Now fix each $\mathbf{x}_i$ at the centroids of the region $\widehat{V}_i$ for $i = 1, \ldots, n$; keep the tessellation $\mathcal{V}$ fixed as well. For ease of notation denote $\mathcal{H}(\mathbf{X}, \boldsymbol{\alpha}, \mathcal{V})|_{\mathbf{X}, \mathcal{V}} = \mathcal{H}(\boldsymbol{\alpha})$. The first variation with respect to the angle constrains that for all l, $\frac{\partial \mathcal{H}(\boldsymbol{\alpha})}{\partial \alpha_i^{(l)}} = 0$. Each partial derivative is

$$\frac{\partial \mathcal{H}(\boldsymbol{\alpha})}{\partial \alpha_i^{(l)}} = \int_{\widehat{V}_i} \left((\mathbf{y} - \mathbf{x}_i)^\top R_i \mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R^\top}{\partial \alpha_i^{(l)}} (\mathbf{y} - \mathbf{x}_i) + (\mathbf{y} - \mathbf{x}_i)^\top \frac{\partial R_i}{\partial \alpha_i^{(l)}} \mathbf{A}_i^\top \mathbf{A}_i R_i^\top (\mathbf{y} - \mathbf{x}_i) \right) \rho(\mathbf{y}) \, d\mathbf{y},$$

$$= 2 \int_{\widehat{V}_i} (\mathbf{y} - \mathbf{x}_i)^\top R_i \mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} (\mathbf{y} - \mathbf{x}_i) \rho(\mathbf{y}) \, d\mathbf{y}.$$

The expression $\frac{\partial R_i^\top}{\partial \alpha_i^{(l)}}$ denotes the componentwise partial derivative of the matrix $R_i^\top$. The trace function can be applied to the scalar $\frac{\partial \mathcal{H}(\boldsymbol{\alpha})}{\partial \alpha_i^{(l)}}$, and the cyclic property of the trace can be applied to find

$$\frac{\partial \mathcal{H}(\boldsymbol{\alpha})}{\partial \alpha_i^{(l)}} = \text{tr}\left(2 \int_{\widehat{V}_i} (\mathbf{y} - \mathbf{x}_i)^\top R_i \mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} (\mathbf{y} - \mathbf{x}_i) \rho(\mathbf{y}) \, d\mathbf{y} \right)$$

$$(6.5) \qquad = 2 \int_{\widehat{V}_i} \text{tr}\left((\mathbf{y} - \mathbf{x}_i)^\top R_i \mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} (\mathbf{y} - \mathbf{x}_i) \right) \rho(\mathbf{y}) \, d\mathbf{y}$$

$$(6.6) \qquad = 2 \int_{\widehat{V}_i} \text{tr}\left(R_i \mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} (\mathbf{y} - \mathbf{x}_i)(\mathbf{y} - \mathbf{x}_i)^\top \right) \rho(\mathbf{y}) \, d\mathbf{y}$$

$$(6.7) \qquad = 2 \, \text{tr}\left(R_i \mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} \int_{\widehat{V}_i} (\mathbf{y} - \mathbf{x}_i)(\mathbf{y} - \mathbf{x}_i)^\top \rho(\mathbf{y}) \, d\mathbf{y} \right)$$

$$(6.8) \qquad = 2 \, \text{tr}\left(R_i \mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} \mathbf{M}_i \right)$$

$$(6.9) \qquad = 2 \, \text{tr}\left(\mathbf{A}_i^\top \mathbf{A}_i \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} R_i R_i^\top \underbrace{\mathbf{M}_i}_{= \mathbf{U}_i^\top \Lambda \mathbf{U}_i} R_i \right)$$

$$(6.10) \qquad = 2 \, \text{tr}\left(\underbrace{\left(\frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} R_i \right)}_{=: S} \underbrace{R_i^\top \mathbf{U}_i^\top \Lambda \mathbf{U}_i R_i \mathbf{A}_i^\top \mathbf{A}_i}_{=: B} \right).$$

In (6.5), the linearity of the trace and the integral are used to take the trace inside. In (6.6), the cyclic property of the trace is used to move the first vector to the end. Then,

in (6.7), the linearity of the trace and integral are used again to take the integral inside. In (6.8), the integral is denoted by $\mathbf{M}_i$. As this second moment matrix is positive semidefinite, it is diagonalized as $\mathbf{U}_i^\top \Lambda \mathbf{U}_i$, for $\mathbf{U}_i$ orthogonal and Λ diagonal. This decomposition is used in (6.10). In (6.9), the cyclic property of the trace is used to move the first matrix R_i to the end of the expression, and the term $R_i R_i^\top$ is added, as this is just the identity (rotation matrices are orthogonal). In the last line, (6.10), $S = \frac{\partial R_i^\top}{\partial \alpha_i^{(l)}} R_i$ is a skew-symmetric matrix. Observe: Because rotation matrices are orthogonal, $R_i^\top R_i = I$. Then, by the product rule, $\frac{\partial R^\top}{\partial \alpha_i^{(l)}} R_i + R_i^\top \frac{\partial R_i}{\partial \alpha_i^{(l)}} = 0$, and rearranging, $S = -S^\top$. Since the trace of the product of a skew-symmetric matrix (S) and a symmetric matrix (B) is zero, setting $R_i = \mathbf{U}_i^\top$ implies that (6.10) is 0. $\square$

Proposition 6.1 recovers the geometric intuition that in order to form an optimal arrangement, the shapes should be both "well centered" and "well oriented" within their Voronoi regions. For example, in $\mathbb{R}^2$ the condition on the rotation matrix R_i reduces to the following formula for the shape rotation angle:

$$\alpha_i = \frac{1}{2} \arctan \left(\frac{2m_{1,2}}{m_{2,2} - m_{1,1}} \right).$$

This angle is known as the *moment of inertia* of the algebraic Voronoi region. This means that the principal axis of the shape should be aligned with the principal axis of its algebraic Voronoi region. In higher dimensions, the principal axis of the shape should always be aligned with the eigenvector corresponding to the largest eigenvalue of $\mathbf{U}_i^\top$. Thus assuming we can compute the parameters of the approximating ellipsoids needed for the approximate energy (6.3), we can formulate a *generalized Lloyd's method* in $\mathbb{R}^N$ as follows: First, project $\mathbf{x}_i$ to the center of mass of $V_{a,i}$, $i = 1, \ldots, n$. Then project R_i, the N-dimensional rotation matrix to $\mathbf{U}_i^\top$, the transpose of the eigenvector matrix of the second moments $\mathbf{M}_i$, $i = 1, \ldots, n$. Repeat until a fixed point is obtained.

However, note that the parameters of the approximating ellipsoids (for example, r_i, b_i, c_i in $\mathbb{R}^3$) do not appear in these moment conditions but are required to compute the algebraic Voronoi regions. This suggest a *simplified generalized Lloyd's method* for which these parameters need not be estimated: use the original geometric Voronoi region V_i instead of the algebraic Voronoi region $V_{a,i}$ to compute these moments. Specifically, given a collection of shapes S_i and an initial placement $(\mathbf{X}, \boldsymbol{\alpha})$, with R being the d-dimensional rotation matrix parameterized by $\boldsymbol{\alpha}$,

(i) compute the centroids of V_i, $i = 1, \ldots, n$, and project $\mathbf{x}_i$ to the respective centroids;

(ii) compute $\mathbf{U}_i^\top$, the transpose of the eigenvector matrix of the second moments $\mathbf{M}_i$, $i = 1, \ldots, n$, given by (6.4) (based on geometric V_i), and project R_i to $\mathbf{U}_i^\top$;

(iii) recompute the Voronoi regions and repeat until a fixed point is obtained.

In $\mathbb{R}^2$ this algorithm was already proposed by [35], and results were presented with application to nonphotorealistic rendering. The present work extends that algorithm to $\mathbb{R}^N$ and gives an energetic interpretation of the results. It is important to note that the only basis for presenting these two generalized Lloyd's algorithms was the motivation provided by Proposition 6.1. These algorithms warrant both numerical experimentation and a contraction/convergence study. Indeed, convergence of the regular Lloyd's algorithm is nontrivial (see, for example, [22]), and hence convergence of these more complicated geometric integration schemes is far from obvious.

7. A relation to optimal transport. Many readers will have noticed the similarity between the structure of our arrangement cost (2.1) and notions from the active mathematical field of *optimal transport* [66]. Here we make the simple remark that the energy for $p = 2$ can be written in terms of the Wasserstein-2 distance. In doing so we provide an elementary illustration of aspects of Brenier's theorem [10].

The field of optimal transport arose as a way to solve the problem of minimizing the transport cost to haul ore from a set of mines to a collection of factories [66]. This engineering problem was later reformulated and rediscovered in many contexts. Let μ be a probability measure on $\Omega \subset \mathbb{R}^N$ that is absolutely continuous with respect to Lebesgue measure; where applicable, denote its density by ρ. Let ν be any Borel probability measure. Define measures on the product space $\Omega \times \Omega$ as follows:

$$\mathcal{M}(\mu, \nu) = \big\{\text{All Borel measures } \pi : \Omega \times \Omega \to [0,1] \,\big|\, \pi(\mathbf{y}, \Omega) = \mu(\mathbf{y}) \text{ and } \pi(\Omega, \mathbf{z}) = \nu(\mathbf{z})\big\}.$$

That is, we consider product measures whose projections in each component are the respective given marginal.

Kantorovich's formulation of optimal transportation for Euclidean cost is given as

$$(7.1) \qquad W_2^2(\mu, \nu) = \inf_{\pi} \left\{ \int_{\Omega \times \Omega} |\mathbf{y} - \mathbf{z}|^2 d\pi(\mathbf{y}, \mathbf{z}) \,\bigg|\, \pi \in \mathcal{M}(\mu, \nu) \right\},$$

where $W_2(\mu, \nu)$ is called the Wasserstein-2 distance between the measures μ and ν. This modern formulation of the problem is useful because $\mathcal{M}$ is always nonempty; it contains the product measure $\mu \times \nu$ [66]. In particular, this guarantees a minimizer always exists. The minimizing measure on the product space, π, is called a transport plan. A more classical formulation of the problem, going back to Monge, is via a *transport map* $q : \text{supp}(\mu) \to \text{supp}(\nu)$. Given a Borel probability measure μ and a Borel map $q : \mathbb{R}^N \to \mathbb{R}^N$, we define the *push forward* probability measure $q_\# \mu$ by

$$(q_\# \mu)(\mathcal{A}) = \mu\left(q^{-1}(\mathcal{A})\right) \qquad \forall \text{ Borel-measurable subsets } \mathcal{A} \subset \mathbb{R}^N.$$

Denoting $\psi(\mathbf{y}) = (\mathbf{y}, q(\mathbf{y}))$, the transport plan π can be recovered as $\pi = \mu \circ \psi^{-1}$ [42]. In these cases, one can formulate the Wasserstein-2 distance (7.1) in the more classical way:

$$(7.2) \qquad W_2^2(\mu, \nu) = \inf_{q} \left\{ \int_{\Omega} |\mathbf{y} - q(\mathbf{y})|^2 d\mu(\mathbf{y}) \,\bigg|\, \nu = q_\# \mu \right\}.$$

Brenier's theorem and its extensions [50, 10] prove that the formulations (7.1) and (7.2) are equivalent, and in fact a unique transport map q exists that minimizes (7.2), and is, up to sets of measure zero, the gradient of a convex function.

Now let us make a simple connection with the arrangement energy (2.1). For simplicity of notation, let us assume $|\Omega| = 1$ and take the density $\rho \equiv 1$. Let μ be given by N-dimensional Lebesgue measure $d\mathbf{y}$. To define the singular measure ν, let S_i be a fixed collection of shapes which are compact subsets of Ω of codimension in $\{1, 2, \ldots, N\}$. For any set $T = \{T_i\}$, $i = 1, \ldots, n$, of affine isometries such that $T_i(S_i) \subset \overline{\Omega}$, we define the singular measure ν_T supported on $\bigcup_{i=1}^{n} T_i(S_i)$ as follows:[3] For any Borel-measurable set $\mathcal{A} \subset \Omega$,

$$(7.3) \quad \nu_T(\mathcal{A}) = \frac{\left|\left\{ \mathbf{z} \in \Omega \,\big|\, d\left(\mathbf{z}, \bigcup_{i=1}^{n} T_i(S_i)\right) = |\mathbf{z} - \mathbf{y}| \text{ for some } \mathbf{y} \in \mathcal{A} \bigcap T_i(S_i) \right\}\right|}{|\Omega|}.$$

---

[3] For nonconvex generators $S_1, \ldots, S_n$, the definition of ν requires a tie-breaking rule for points that are equidistant to two or more points along some S_i.

In other words, $\nu_T(\mathcal{A})$ is the percent of volume of the Voronoi regions which are associated with the part of the generators $\bigcup_{i=1}^{n} T_i(S_i)$ which lie in $\mathcal{A}$. For example, if the S_i were points and T was represented by their positions $\mathbf{x} = \{x_i\}$, then $\nu_T = \sum_{i=1}^{n} m_i \delta_{x_i}$ where $m_i = |V_i|$ with $\sum_{i=1}^{n} m_i = |\Omega| = 1$. This point case has been studied within the framework of optimal transport in [51, 8, 9].

For target measures given by (7.3), Monge's formulation (7.2) is rather simple as the unique optimal push forward is, up to a set of measure 0, simply a projection onto the closest point in $\bigcup_{i=1}^{n} T_i(S_i)$, that is,

$$q_*(\mathbf{y}) = \mathbf{y} - d\left(\mathbf{y}, \bigcup_{i=1}^{n} T_i(S_i)\right) \nabla d\left(\mathbf{y}, \bigcup_{i=1}^{n} T_i(S_i)\right).$$

Moreover in this case, if we parameterize the family of isometries T by the respective position and angle vectors $(\mathbf{X}, \boldsymbol{\alpha})$, we find

$$\begin{aligned} W_2^2(\mu, \nu_T) &= \inf_q \left\{ \int_\Omega |\mathbf{y} - q(\mathbf{y})|^2 d\mathbf{y}, \quad \mu \circ q^{-1} = \nu_T \right\} \\ &= \int_\Omega |\mathbf{y} - q_*(\mathbf{y})|^2 d\mathbf{y} \\ &= \int_\Omega d^2\left(\mathbf{y}, \bigcup_{i=1}^{n} T_i(S_i)\right) d\mathbf{y} = \sum_{i=1}^{n} \int_{V_i} d^2(\mathbf{y}, T_i(S_i)) d\mathbf{y} = F(\mathbf{X}, \boldsymbol{\alpha}). \end{aligned}$$

Thus our optimal arrangement problem of finding $(\mathbf{X}, \boldsymbol{\alpha})$ that minimizes the arrangement energy $F(\mathbf{X}, \boldsymbol{\alpha})$ given by (2.1) is equivalent to finding T that minimizes $W_2^2(\mu, \nu_T)$.

8. A few applications.

8.1. Empirical explorations of ground states and the energy landscape.
We have defined an *optimal assembly* as a critical point of the arrangement cost function (2.1) and presented an algorithm which arrives at a critical point of F. However, even after modding out by domain symmetry effects, the energy landscape of the arrangement function F is highly nonconvex with many local minimizers. Questions as to the nature of ground state arrangements, a global minimum for (2.1), and characterizations of the stability of optimal arrangements are most natural but, from a purely analytical point of view, very difficult.

Even for the cases of points, where optimal arrangements directly correspond to CVTs, questions as to the nature of the ground state remain open. *Gersho's conjecture* [29] was originally stated for optimal vector quantizers used in data compression and transmission. In essence, the conjecture asserts that asymptotically as the number of generators gets larger, the ground state CVT is a periodic tessellation based upon a single polytope (cell) dictated by the space dimension. The basic Voronoi cell of the ground state CVT was shown to be the regular hexagon in two dimensions [27] (see also [53] for a simple proof). For the three-dimensional space with a constant density function, it was proved that among all lattice-based CVTs, the CVT corresponding to the body-centered cubic lattice (BCC) is the optimal one [4]. While numerical simulations suggest the optimality of BCC with respect to general CVTs, the conjecture remains open [25].

While the case of general shapes will naturally be far more complex, our method does allow for empirical study of ground states and the overall energy landscape of (2.1). We present two illustrations.
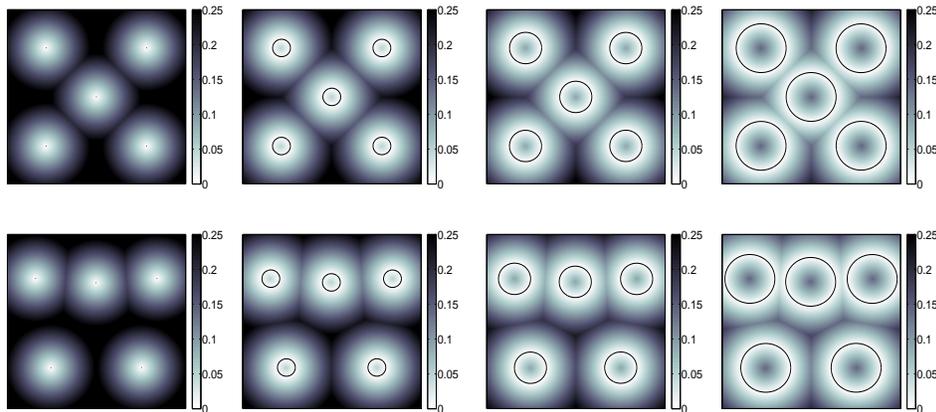
FIG. 8.1. *Two critical points for the arrangement energy* (2.1) *with five circles of equal radius; $r = 0$, 0.05, 0.09, and 0.14 from left to right. The top geometry has the lowest energy until $r = 0.134$, after which the bottom geometry is optimal.*

A phase transition with circles of equal radii. We consider five circles of the same radius. This case is particularly simple since as long as the circles do not overlap, the Voronoi regions generated by the circles coincide with the Voronoi regions generated by the center points of the circles. Here we show that as the radii increase there is a critical value at which there is a phase transition in the morphology of the global minimizer. To this end, we let $\Omega = [0, 1]^2$ with uniform density, and S_i, $i = 1, \ldots, 5$, be circular generators of radius r. The gradient flow for the centers is given by the following 10-dimensional coupled dynamical system: For $i = 1, \ldots, 5$,

$$\frac{\partial \mathbf{x}_i}{\partial t} = -\nabla_{\mathbf{x}_i} F = -2 \int_{V_i} (\mathbf{x}_i - \mathbf{y}) \left(1 - \frac{r}{|\mathbf{x}_i - \mathbf{y}|}\right) \rho(\mathbf{y}) \, d\mathbf{y}, \qquad \mathbf{x}_i(0) = \mathbf{x}_i.$$

Many runs (3,000) were performed with the initial condition $\mathbf{x}_i(0)$ taken from a uniform distribution, and the converged state with the lowest energy is recorded. We loosely refer to this as the "global minimizer." For $r = 0$, the global minimizer of the CVT energy is a $4 + 1$ configuration, with four points in each corner, and one point in the center of the unit square. This configuration is displayed in the first configuration of the top of Figure 8.1. Another critical point for the $r = 0$ case is shown[4] in the first configuration of the bottom of Figure 8.1. Figure 8.1 also displays the analogous critical points for $r = 0.05$, 0.09, and 0.14. Because the Voronoi regions are polygonal, one can compute the critical centers and corresponding energies analytically and to machine precision (cf. [43]). In doing so, one finds a phase transition wherein the bottom configuration represents the global minimizer at around $r = 0.134$.

Energy histogram. Another application is to generate energy histograms by uniformly sampling initial centers and rotation angles; applying our algorithm to convergence; recording the final arrangement energy; and plotting the histogram. We give one such example in Figure 8.2. Here we also plot a sample of the lowest energy arrangement—a good candidate for a ground state. Note that the smaller oscillations (bumps) in the energy distribution of Figure 8.2 (right) should be ignored as they are

---

[4]Because of the symmetries of the domain, there are three other critical points with the same energy under the rotations $\theta = \pi/2$, π, and $3\pi/2$.
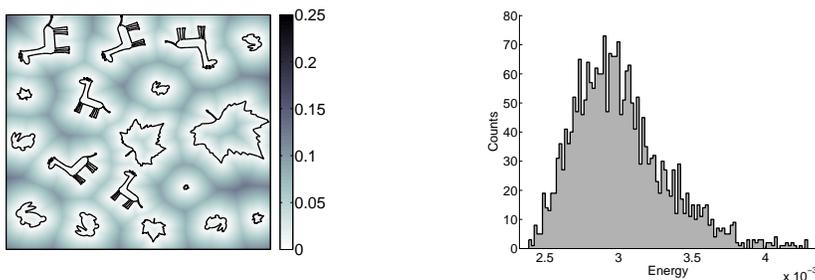
FIG. 8.2. *Energy distribution for* 1,000 *runs of* 18 *shapes with M = 6 in L-BFGS.* Left: *An example with the lowest energy.* Right: *Approximate energy histogram.*

most probably numerical effects: the energies differences are sufficiently small with respect to the tolerance of the method.

8.2. Packing of shapes. The self-assembly algorithm we have outlined here offers a natural way to spread shapes out in a domain. Then it is natural to ask, whether by shrinking the domain or growing the shapes, if a *packing* can be obtained whereby the volume outside the shapes is minimized. Obtaining optimal packing densities for circles has been well studied (see, for example, the monograph [15]). In $\mathbb{R}^2$, the optimal packing lattice for circles is the hexagonal lattice. In $\mathbb{R}^3$, the optimal packing lattice for spheres is the body-centered cubic. Few theoretical results are known for packing spheres in high dimensions [14]. For ellipsoids, it has been shown numerically that nonlattice packings lead to better packing densities [20]. Recently, algorithms for tetrahedral packings have been considered [33]. Algorithms for packing shapes typically use hard constraints for enforcing no overlap: at each iteration, it must be checked whether any shape intersects another shape.

To obtain a packing using our self-assembly algorithm, a natural approach is to begin with small particles and grow them until a packing is obtained. Then there are two things to consider: first, an appropriate rate of growth must be chosen, and second, a stopping criterion must be set so that the algorithm terminates when a packing is obtained. Here, we sketch these two steps.

Starting with a collection of small shapes (relative to the domain volume), obtain a self-assembly. Then alternate between growing the shapes and translating and rotating the shapes as to minimize the assembly energy. If the rate of growth is too fast, the shapes may not be able to spread out quickly enough, leading to suboptimal results. We propose to grow the generators at a rate that is based on the assembly energy. Previously, energy derivatives were calculated for rotations and translations of the shapes; to choose the growth rate, we will use the derivative of the energy with respect to *scaling* the shapes. For example, in the case of spherical generators, the scale of the shape is given by the radius parameter, and so

$$\frac{\partial F}{\partial r_i} = \frac{\partial}{\partial r_i}\left(\sum_{i=1}^{n}\int_{V_i}\Big(d(\mathbf{y},\mathbf{x}_i)-r_i\Big)^2\rho(\mathbf{y})\,d\mathbf{y}\right) = -2\int_{V_i}\Big(d(\mathbf{y},\mathbf{x}_i)-r_i\Big)\rho(\mathbf{y})\,d\mathbf{y}.$$

For points $\mathbf{y} \in V_i$ outside the sphere, the integrand $d(\mathbf{y},\mathbf{x}_i) - r_i$ is the distance to the sphere. For points $\mathbf{y} \in V_i$ interior to the sphere, the integrand is the negative distance. Therefore, when the energy inside and outside of the sphere is equal, this partial derivative will be zero. If this derivative is negative, there is a surplus of space

around the sphere, and the radius should increase. If the derivative is positive, then there is more energy inside the sphere, and the radius should shrink. To consider only the energy outside of the spheres, one can use

$$\frac{\partial \widetilde{F}}{\partial r_i} := -2 \int_{V_i \setminus \Gamma_i} \Big(d(\mathbf{y}, \mathbf{x}_i) - r_i \Big) \rho(\mathbf{y}) \, d\mathbf{y},$$

where Γ_i is the disk with boundary S_i. We then calculate $\frac{\partial F}{\partial r_i}$ for each i and choose the smallest derivative as the growth rate. As the shapes get closer together, the growth rate will naturally slow, allowing for more translation and rotation so that the shapes can pack well.

The second consideration is the stopping criterion. Whereas previous methods use hard constraints to avoid overlap, we propose a soft constraint. Instead of directly assessing whether shapes are overlapping, we suggest adjusting the arrangement energy directly to penalize overlap. This involves changing the energy interior to each shape. Currently, powers of the distance function are used. However, the signed distance function would assign negative energy to the interior of the shape (for odd powers). Then overlap would increase the energy, so minimizing the energy would promote nonoverlapping shapes. Phase transitions, such as highlighted in section 8.1, could occur during the growth of the shapes. The interesting question of which optima can be accessed using a growth-based algorithm is very relevant to the packing application. In fact, the ground state may only be accessible by starting from a suboptimal local minimum for the small, initialized shapes. Assessing the complex nature of the energy landscape for packing problems is a rich area for future research. The novelty of trying to adapt the current algorithm to packing is that the class of shapes it can accommodate is large.

8.3. Energy-driven pattern formation. Pattern formation of modulated phases can often be explained via minimization of energy functionals involving competing short- and long-range interactions [62]. The subject has proven to be a driving force for much research in the modern calculus of variations [39]. A good case in point is the Ohta–Kawasaki functional [56, 13], which provides one of the simplest models for self-assembly of diblock copolymers. Here the ubiquitous Ginzburg–Landau free energy is augmented with long-range interactions of Coulombic type, presenting a rich set of mathematical problems as to the nature of global and local minimizers (see, for example, [11, 59] and the references therein). In the regime of small volume fraction, wherein small spherical phases are present, there are two temporal regimes in the gradient flow dynamics [34, 30]: the formation of the spherical structures and their coarsening until an intrinsic length scale is reached, followed by pattern formation of the spheres dictated by the long-range interactions. These dynamic regimes are directly connected to the first few terms in the Gamma-limit expansion of the free energy [12]. The problem considered in this article addresses the later stage of pattern formation. While for Ohta–Kawasaki these interactions are of Coulombic type, here the interactions involve Wasserstein distances (cf. section 7). Recently Bourne et al. [8, 9] considered a simplified model in the dilute phase limit based upon Wasserstein norms. This results in a variational problem defined over sums of weighted delta functions with support in $\Omega \subset \mathbb{R}^N$, $\nu = \sum_{i=1}^{n} m_i \delta_{x_i}$ with $\sum_{i=1}^{n} m_i = |\Omega| = 1$, which takes the form

$$E(\nu) = \lambda \sum_{i=1}^{n} (m_i)^{\frac{N-1}{N}} + W_2^2(d\mathbf{y}, \nu),$$

where $d\mathbf{y}$ denotes N-dimensional Lebesgue measure on Ω, and W_2 is the Wasserstein distance defined in section 7. Minimization is made computationally tractable by setting the problem in the framework of CVT and adopting the simple approach of Lloyd's algorithm [8]. It would be interesting to adopt the methods contained here to explore shape and orientation effects in, for example, related simplified models for hard ellipses/ellipsoids.

8.4. Constrained optimal transport. For the rigid CVT problem, the optimal transportation problem becomes computationally tractable by relying on the tools of computational geometry: generalized Voronoi regions and eikonal distances. However, there is a large body of work for point generators on the constrained CVT problem: how can one minimize the average squared distance to the closest post office while ensuring the number of people serviced by each post office is equal? The constraint is that the measure of each Voronoi region with respect to a density, ρ, is equal. Some work on efficient algorithms to compute constrained CVTs is contained in [1, 3, 18, 51, 70]. Efficient computation in the constrained case relies on a characterization of the optimal transport plan in terms of power Voronoi diagrams [1, 51]. Accommodating constraints could also be pursued in the case of general shapes.

REFERENCES

[1] F. Aurenhammer, F. Hoffmann, and B. Aronov, *Minkowski-type theorems and least-squares clustering*, Algorithmica, 20 (1998), pp. 61–76.

[2] G. Baldassarre, V. Trianni, M. Bonani, F. Mondada, M. Dorigo, and S. Nolfi, *Self-organized coordinated motion in groups of physically connected robots*, IEEE Trans. Syst. Man Cybernet. B, 37 (2007), pp. 224–239.

[3] M. Balzer, T. Schlömer, and O. Deussen, *Capacity-constrained point distributions: A variant of Lloyd's method*, ACM Trans. Graph., 28 (2009), 86.

[4] E. S. Barnes and N. J. A. Sloane, *The optimal lattice quantizer in three dimensions*, SIAM J. Algebraic Discrete Methods, 4 (1983), pp. 30–41, doi:10.1137/0604005.

[5] F. S. Bates and G. H. Fredrickson, *Block copolymers: Designer soft materials*, Phys. Today, February 1999, pp. 32–38.

[6] A. J. Bernoff and C. M. Topaz, *A primer of swarm equilibria*, SIAM J. Appl. Dyn. Syst., 10 (2011), pp. 212–250, doi:10.1137/100804504.

[7] X. Blanc and M. Lewin, *The Crystallization Conjecture: A Review*, preprint, arXiv: 1504.01153, 2015.

[8] D. P. Bourne, M. A. Peletier, and S. M. Roper, *Hexagonal patterns in a simplified model for block copolymers*, SIAM J. Appl. Math., 74 (2014), pp. 1315–1337, doi:10.1137/130922732.

[9] D. P. Bourne, M. A. Peletier, and F. Theil, *Optimality of the triangular lattice for a particle system with Wasserstein interaction*, Comm. Math. Phys., 329 (2014), pp. 117–140.

[10] Y. Brenier, *Polar factorization and monotone rearrangement of vector-valued functions*, Comm. Pure Appl. Math., 44 (1991), pp. 375–417.

[11] R. Choksi, *On global minimizers for a variational problem with long-range interactions*, Quart. Appl. Math., 70 (2012), pp. 517–537.

[12] R. Choksi and M. A. Peletier, *Small volume fraction limit of the diblock copolymer problem: I. Sharp-interface functional*, SIAM J. Math. Anal., 42 (2010), pp. 1334–1370, doi:10.1137/090764888.

[13] R. Choksi and X. Ren, *On the derivation of a density functional theory for microphase separation of diblock copolymers*, J. Stat. Phys., 113 (2003), pp. 151–176.

[14] H. Cohn and N. Elkies, *New upper bounds on sphere packings* I, Ann. of Math. (2), 157 (2003), pp. 689–714.

[15] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*, 3rd ed., Springer-Verlag, New York, 1999.

[16] I. Couzin, *Collective minds*, Nature, 445 (2007), p. 715.

[17] F. Cucker and S. Smale, *Emergent behavior in flocks*, IEEE Trans. Automat. Control, 52 (2007), pp. 852–862.

[18] F. de Goes, K. Breeden, V. Ostromoukhov, and M. Desbrun, *Blue noise through optimal transport*, ACM Trans. Graph., 31 (2012), 171.

[19] Z. Di, M. Emelianenko, and S. G. Nash, *Truncated Newton-based multigrid algorithm for centroidal Voronoi diagram calculation*, Numer. Math. Theory Methods Appl., 5 (2012), pp. 242–259.

[20] A. Donev, I. Cisse, D. Sachs, E. A. Variano, F. H. Stillinger, R. Connelly, S. Torquato, and P. M. Chaikin, *Improving the density of jammed disordered packings using ellipsoids*, Science, 303 (2004), pp. 990–993.

[21] Q. Du and M. Emelianenko, *Uniform convergence of a nonlinear energy-based multilevel quantization scheme*, SIAM J. Numer. Anal., 46 (2008), pp. 1483–1502, doi:10.1137/050648699.

[22] Q. Du, M. Emelianenko, and L. Ju, *Convergence of the Lloyd algorithm for computing centroidal Voronoi tessellations*, SIAM J. Numer. Anal., 44 (2006), pp. 102–119, doi:10.1137/040617364.

[23] Q. Du, V. Faber, and M. Gunzburger, *Centroidal Voronoi tessellations: Applications and algorithms*, SIAM Rev., 41 (1999), pp. 637–676, doi:10.1137/S0036144599352836.

[24] Q. Du, M. Gunzburger, and L. Ju, *Advances in studies and applications of centroidal Voronoi tessellations*, Numer. Math. Theory Methods Appl., 3 (2010), pp. 119–142.

[25] Q. Du and D. Wang, *The optimal centroidal Voronoi tessellations and the Gersho's conjecture in the three-dimensional space*, Comput. Math. Appl., 49 (2005), pp. 1355–1373.

[26] M. Emelianenko, *Fast multilevel CVT-based adaptive data visualization algorithm*, Numer. Math. Theory Methods Appl., 3 (2010), pp. 195–211.

[27] L. Fejes Tóth, *Lagerungen in der Ebene, auf der Kugel und im Raum*, Springer, New York, 1972.

[28] G. H. Fredrickson, *The Equilibrium Theory of Inhomogeneous Polymers*, Oxford Science Press, Oxford, 2006.

[29] A. Gersho, *Asymptotically optimal block quantization*, IEEE Trans. Inform. Theory, 25 (1979), pp. 373–380.

[30] K. Glasner and R. Choksi, *Coarsening and self-organization in dilute diblock copolymer melts and mixtures*, Phys. D, 238 (2009), pp. 1241–1255.

[31] M. Grzelczak, J. Vermant, E. M. Furst, and L. M. Liz-Marzán, *Directed self-assembly of nanoparticles*, ACS Nano, 4 (2010), pp. 3591–3605.

[32] B. A. Grzybowski, C. E. Wilmer, J. Kim, K. P. Browne, and K. J. M. Bishop, *Self-assembly: From crystals to cells*, Soft Matter, 5 (2009), pp. 1110–1128.

[33] A. Haji-Akbari, M. Engel, A. Keys, X. Zheng, R. Petschek, P. Palffy-Muhoray, and S. Glotzer, *Disordered, quasicrystalline and crystalline phases of densely packed tetrahedra*, Nature, 462 (2009), pp. 773–777.

[34] M. Helmers, B. Niethammer, and X. Ren, *Evolution in off-critical diblock copolymer melts*, Netw. Heterog. Media, 3 (2008), pp. 615–632.

[35] S. Hiller, H. Hellwig, and O. Deussen, *Beyond stippling—methods for distributing objects on the plane*, Computer Graphics Forum, 22 (2003), pp. 515–522.

[36] M. Iri, K. Murota, and T. Ohya, *A fast Voronoi-diagram algorithm with applications to geographical optimization problems*, in System Modelling and Optimization, Lecture Notes in Control and Inform. Sci. 59, P. Thoft-Christensen, ed., Springer, Berlin, Heidelberg, 1984, pp. 273–288.

[37] E. Jankowski and S. C. Glotzer, *Calculation of partition functions for the self-assembly of patchy particles*, J. Phys. Chem. B, 115 (2011), pp. 14321–14326.

[38] L. Ju, Q. Du, and M. Gunzburger, *Probabilistic methods for centroidal Voronoi tessellations and their parallel implementations*, Parallel Comput., 28 (2002), pp. 1477–1500.

[39] R. V. Kohn, *Energy-driven pattern formation*, in Proceedings of the International Congress of Mathematicians, Vol. I, Zürich, Eur. Math. Soc., 2007, pp. 359–383.

[40] Y. Koren and I. Yavneh, *Adaptive multiscale redistribution for vector quantization*, SIAM J. Sci. Comput., 27 (2006), pp. 1573–1593, doi:10.1137/040607769.

[41] Y. Koren, I. Yavneh, and A. Spira, *A multigrid approach to the scalar quantization problem*, IEEE Trans. Inform. Theory, 51 (2005), pp. 2993–2998.

[42] W. Kreitmeier, *Optimal vector quantization in terms of Wasserstein distance*, J. Multivariate Anal., 102 (2011), pp. 1225–1239.

[43] L. J. Larsson, *Generalized Voronoi Regions: Integration and Optimization*, Ph.D. thesis, McGill University, 2014.

[44] L. J. Larsson, R. Choksi, and J.-C. Nave, *An iterative algorithm for computing mea-*

sures of generalized Voronoi regions, SIAM J. Sci. Comput., 36 (2014), pp. A792–A827, doi:10.1137/130935598.

[45] D. C. LIU AND J. NOCEDAL, *On the limited memory BFGS method for large scale optimization*, Math. Programming, 45 (1989), pp. 503–528.

[46] 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. Graphics, 28 (2009), 101.

[47] S. LLOYD, *Least squares quantization in PCM*, IEEE Trans. Inform. Theory, 28 (1982), pp. 129–137.

[48] L. LU, B. LEVY, AND W. WANG, *Centroidal Voronoi tessellation of line segments and graphs*, Computer Graphics Forum, 31 (2012), pp. 775–784.

[49] J. MACQUEEN, *Some methods for classification and analysis of multivariate observations*, in Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281–297.

[50] R. MCCANN AND N. GUILLEN, *Five lectures on optimal transport*, in Analysis and Geometry of Metric Measure Spaces, CRM Proc. Lecture Notes 56, 2010, pp. 145–180.

[51] Q. MÉRIGOT, *A multiscale approach to optimal transport*, Computer Graphics Forum, 30 (2011), pp. 1583–1592.

[52] J. J. MORÉ AND D. J. THUENTE, *Line search algorithms with guaranteed sufficient decrease*, ACM Trans. Math. Software, 20 (1994), pp. 286–307.

[53] D. NEWMAN, *The hexagon theorem*, IEEE Trans. Inform. Theory, 28 (1982), pp. 137–139.

[54] J. NOCEDAL, *Updating quasi-Newton matrices with limited storage*, Math. Comp., 35 (1980), pp. 773–782.

[55] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, 2nd ed., Springer Series in Operations Research and Financial Engineering, Springer, New York, 2006.

[56] T. OHTA AND K. KAWASAKI, *Equilibrium morphology of block copolymer melts*, Macromolecules, 19 (1986), pp. 2621–2632.

[57] A. OKABE, B. BOOTS, AND K. SUGIHARA, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, John Wiley and Sons, 1992.

[58] S. J. OSHER AND R. P. FEDKIW, *Level Set Methods and Dynamic Implicit Surfaces*, Springer-Verlag, New York, 2002.

[59] X. REN AND J. WEI, *Spherical solutions to a nonlocal free boundary problem from diblock copolymer morphology*, SIAM J. Math. Anal., 39 (2008), pp. 1497–1535, doi:10.1137/070690286.

[60] G. RONG, Y. LIU, W. WANG, X. YIN, X. D. GU, AND X. GUO, *GPU-assisted computation of centroidal Voronoi tessellation*, IEEE Trans. Visualization Comput. Graphics, 17 (2011), pp. 345–356.

[61] J. A. SETHIAN, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, Cambridge, UK, 1999.

[62] M. SEUL AND D. ANDELMAN, *Domain shapes and patterns: The phenomenology of modulated phases*, Science, 267 (1995), p. 467.

[63] G. SRINIVAS, D. E. DISCHER, AND M. L. KLEIN, *Self-assembly and properties of diblock copolymers by coarse-grain molecular dynamics*, Nature Materials, 3 (2004), pp. 638–644.

[64] Y.-H. R. TSAI, L.-T. CHENG, S. OSHER, AND H.-K. ZHAO, *Fast sweeping algorithms for a class of Hamilton–Jacobi equations*, SIAM J. Numer. Anal., 41 (2003), pp. 673–694, doi:10.1137/S0036142901396533.

[65] C. N. VASCONCELOS, A. SÁ, P. C. CARVALHO, AND M. GATTASS, *Lloyd's algorithm on GPU*, in Advances in Visual Computing, Springer, 2008, pp. 953–964.

[66] C. VILLANI, *Optimal Transport: Old and New*, Grundlehren Math. Wiss. 338, Springer, Berlin, 2009.

[67] G. M. WHITESIDES AND B. GRZYBOWSKI, *Self-assembly at all scales*, Science, 295 (2002), pp. 2418–2421.

[68] G. M. WHITESIDES, J. P. MATHIAS, AND C. T. SETO, *Molecular self-assembly and nanochemistry: A chemical strategy for the synthesis of nanostructures*, Science, 254 (1991), pp. 1312–1319.

[69] H. ZHAO, *A fast sweeping method for Eikonal equations*, Math. Comp., 74 (2005), pp. 603–627.

[70] X. ZHAO, Z. SU, X. D. GU, A. KAUFMAN, J. SUN, J. GAO, AND F. LUO, *Area-preservation mapping using optimal mass transport*, IEEE Trans. Visualization Comput. Graphics, 19 (2013), pp. 2838–2847.