

# Approximation of a Spherical Tessellation by the Laguerre Voronoi Diagram

Supanut Chaidee\*

Kokichi Sugihara\*

## Abstract

This paper presents a method for approximating spherical tessellations, the edges of which are geodesic arcs, using spherical Laguerre Voronoi diagrams. The approximation method involves fitting the polyhedron corresponding to the spherical Laguerre Voronoi diagram to the observed tessellation using optimization techniques.

## 1 Introduction

There are many natural phenomena that can be represented with polygonal patterns on a sphere, such as the patterns found on fruit skins. If the tessellation is similar to a Voronoi diagram, then a mathematical model can be constructed to assist with understanding the polygonal pattern formation.

Since the ordinary Voronoi diagram does not provide a good representation of most naturally occurring tessellations, consideration needs to be given to a weighting for the cells. One generalization of the Voronoi diagram is the Laguerre Voronoi diagram, a weighted Voronoi diagram, the edges of which are straight lines. This concept was introduced by [7, 2]. In brief, for a set  $S$  of  $n$  spheres  $s_i = (\mathbf{x}_i, r_i)$  in  $\mathbb{R}^d$ , where  $\mathbf{x}_i$  is the center of the sphere and  $r_i$  is the sphere radius, which is interpreted as the generator weight, the Laguerre distance of  $\mathbf{x} \in \mathbb{R}^d$  from  $s_i$  is defined by

$$d_L(\mathbf{x}, s_i) = \|\mathbf{x} - \mathbf{x}_i\|^2 - r_i^2.$$

This concept was extended to the spherical Laguerre Voronoi diagram (SLVD) in [8].

Active areas of research related to Voronoi diagrams are Voronoi recognition and approximation. The recognition problem is the determination of whether or not a tessellation is the Voronoi diagram. If it is not, we approximate it using the Voronoi diagram that provides the best fit. Many studies have also focused on planar tessellation. Recently, we proposed a method for fitting planar photographic images of spike-containing objects containing the generators in polygons, using ordinary spherical Voronoi diagrams, and applied it to fruit skin pattern analysis in [3].

\*Graduate School of Advanced Mathematical Sciences, Meiji University, Japan, {schaidee, kokichis}@meiji.ac.jp

Regarding the Laguerre Voronoi diagram, Duan et al. studied a method for the recognition of planar tessellations in [6]. The SLVD recognition problem was recently proposed in [5] using the polyhedron corresponding to the SLVD. Some studies have focused on the 3D structure of the Laguerre approximation. The tessellation fitting problem, which was considered in [3], was solved using the SLVD in [4] by approximating the weights of the generators when the locations are known.

This study proposes a method for approximating the SLVD for a spherical tessellation when the generator locations cannot be derived by conventional methods, as in [5]. In this situation, it is necessary to approximate both generator locations and weights. The remainder of this paper is organized as follows. First, we recall some definitions and theorems related to the SLVD. The algorithms for recognizing the SLVD are then presented. For the case that the given tessellation is not represented exactly by the SLVD, the difference between the tessellation and the constructed SLVD is quantified, and an optimization method is employed to find the best fit SLVD. Since the SLVD corresponds to a convex polyhedron, the optimization is applied to adjust this polyhedron to fit the observed tessellation. Finally, we conduct experiments using simulated data to confirm the validity of our method.

## 2 Preliminaries

We assume that the tessellation and the SLVD are on the unit sphere  $U$  in  $\mathbb{R}^3$ , where the center of the sphere is located at the origin  $O(0, 0, 0)$  of the Cartesian coordinate system.

We assume a tessellation  $\mathcal{T} = \{T_1, \dots, T_n\}$  consisting of  $n$  cells is a 3-regular spherical tessellation, where  $T_i$  is a convex spherical polygon, i.e., the polygon edges are sections of geodesic arcs.

Let  $p_i$  be a point on  $U$ . The sphere  $\tilde{c}_i$  centered at  $p_i$  is defined by

$$\tilde{c}_i = \{p \in U \mid \tilde{d}(p_i, p) = r_i\},$$

where  $\tilde{d}(p_i, p)$  is the geodesic distance between  $p_i$  and  $p$ . If  $0 \leq r_i < \pi/2$ ,  $r_i$  is defined as the sphere's radius. Otherwise,  $r_i$  is the imaginary sphere's radius, whose details were given in [5].

The Laguerre Proximity, the distance used for Voronoi construction, is defined by

$$\tilde{d}_L(p, \tilde{c}_i) = \frac{\cos \tilde{d}(p, p_i)}{\cos r_i}.$$

The Laguerre bisector of two circles  $\tilde{c}_i, \tilde{c}_j$  is defined by  $B_L(\tilde{c}_i, \tilde{c}_j) = \{p \in U \mid \tilde{d}_L(p, \tilde{c}_i) = \tilde{d}_L(p, \tilde{c}_j)\}$ .

For a set of  $n$  sphere circles  $\tilde{G} = \{\tilde{c}_1, \dots, \tilde{c}_n\}$  on  $U$ , the regions  $L_i := \tilde{R}(\tilde{G}, \tilde{c}_i) = \{p \in U \mid \tilde{d}_L(p, \tilde{c}_i) < \tilde{d}_L(p, \tilde{c}_j), j \neq i\}$  for all  $i$ , including their boundaries, constitute the SLVD. We denote the SLVD with  $n$  regions  $L_1, \dots, L_n$  as  $\mathcal{L} = \{L_1, \dots, L_n\}$ .

Sugihara presented algorithms for constructing the SLVD for a given set of sphere circles in [8]. Let  $\pi(\tilde{c}_i)$  be the plane passing through  $\tilde{c}_i$  and  $H(\tilde{c}_i)$  the half-space bounded by  $\pi(\tilde{c}_i)$  including the origin of the sphere. The intersection of  $\pi(\tilde{c}_i)$  and  $\pi(\tilde{c}_j)$  is denoted by  $\ell_{i,j}$ . The following theorem gives the relation between  $\ell_{i,j}$  and the Laguerre bisector.

**Theorem 1 ([8])** *The bisector  $B_L(\tilde{c}_i, \tilde{c}_j)$  is the intersection of  $U$  and the plane containing  $\ell_{i,j}$  and  $O$ .*

From Theorem 1, the SLVD can be constructed from the intersection of all halfspaces  $H(\tilde{c}_i)$  for all  $i$  to obtain the convex polyhedron, and by projecting the edges of the polyhedron onto the sphere with respect to the center  $O$ .

### 3 SLVD Recognition

For a given SLVD  $\mathcal{L}$ , there exists a polyhedron  $\mathcal{P}$  whose central projection onto the sphere coincides with  $\mathcal{L}$ . For any tessellation  $\mathcal{T}$ , we can construct the SLVD with the following algorithms, whose details were provided in [5].

Let  $\mathcal{V}$  be the set of tessellation vertices. Let  $\hat{e}_{i,j}$  be the tessellation edge separating cells  $i$  and  $j$ ,  $P_{i,j}$  be the plane passing through the edge  $\hat{e}_{i,j}$ ,  $v_{i,j,k}$  the tessellation vertex corresponding to cells  $i, j, k$ , and  $P_i := \pi(\tilde{c}_i)$  of the  $i$ -th cell.

The following algorithm is for the construction of the first three planes in the recognition process.

#### Algorithm 1 [5]: Plane Construction with Three Adjacent Sites

**Input:** The sphere  $\tilde{c}_i$  centered at  $p_i(x_i, y_i, \sqrt{x_i^2 + y_i^2})$  with radius  $r_i$ , tessellation edges  $\hat{e}_{i,j}, \hat{e}_{j,k}, \hat{e}_{i,k}$ , and tessellation vertex  $v_{i,j,k}$ .

**Output:** The three planes  $P_i, P_j, P_k$  with respect to polygons  $i, j, k$ .

**Procedure:**

1. Construct the plane  $P_i$  containing  $\tilde{c}_i$ .
2. Construct the planes  $P_{i,j}, P_{i,k}, P_{j,k}$ .
3. Find the intersections  $\ell_{i,j}$  of  $P_i$  and  $P_{i,j}$ , and  $\ell_{i,k}$  of  $P_i$  and  $P_{i,k}$ .

4. Construct a geodesic arc  $\hat{e}_{i,j}^c$  such that  $\hat{e}_{i,j}^c$  passes through  $p_i$  and is perpendicular to  $\hat{e}_{i,j}$ .
5. Choose a point  $q_j$  in polygon  $j$  on the arc  $\hat{e}_{i,j}^c$ .
6. Construct the plane  $P_j$  passing through  $\ell_{i,j}, q_j$ .
7. Find the intersection  $\ell_{j,k}$  of the planes  $P_j$  and  $P_{j,k}$ .
8. Construct the plane  $P_k$  passing through the lines  $\ell_{i,k}$  and  $\ell_{j,k}$ .

**end Procedure**

The following algorithm is for the generation of  $n$  planes for the tessellation.

#### Algorithm 2 [5]: Construction of $n$ Planes

**Input:** Spherical tessellation  $\mathcal{T}$  with tessellation vertices  $\mathcal{V}$ .

**Output:** The planes  $P_1, \dots, P_n$  with respect to the polygons  $1, \dots, n$ .

**Comment:**  $\mathbb{P}$  is the set of constructed planes.

**Procedure:**

1. make  $\mathbb{P}$  empty;
2. Choose an arbitrary vertex  $v_{i,j,k} \in \mathcal{V}$  and employ Algorithm 1 to construct planes  $P_i, P_j, P_k$ .
3. Add the planes  $P_i, P_j, P_k$  to the set  $\mathbb{P}$ .
4. Mark  $v_{i,j,k}$  as a used vertex.
5. **while** there exists an unmarked vertex  $v_{l,p,q} \in \mathcal{V}$  such that exactly two planes among  $P_l, P_p$  and  $P_q$  are included in  $\mathbb{P}$ .

**do**

Apply steps 2, 3, and 7 of Algorithm 1 to find

$\ell_{l,q}$

and  $\ell_{p,q}$ .

Construct a plane  $P_q$ .

Add  $P_q$  to the set  $\mathbb{P}$ .

Mark the vertex  $v_{l,p,q}$ .

**end while**

**end Procedure**

For each plane  $P_i \in \mathbb{P}$ , we consider the half-space  $H(P_i)$  which includes the sphere origin  $O$ , and find the intersection of all such halfspaces to obtain the convex polyhedron  $\mathcal{P}$ .

From Algorithm 1 the construction of the polyhedron  $\mathcal{P}$  depends on the initial sphere and the point  $q_j$ . However, even if we choose them arbitrarily, the polyhedron construction can still be carried out. This can be formalized with the following theorem.

**Theorem 2 ([5])** *For a given tessellation  $\mathcal{T}$ , the construction of a polyhedron corresponding to the SLVD  $\mathcal{L}$  is possible with an arbitrary choice of the initial plane  $P_i$  in step 1 and the point  $q_j$  in step 5 of Algorithm 1.*

The proof of this theorem uses the transformation of a polyhedron in the projective space of  $\mathbb{R}^3$  and the construction processes in Algorithm 1.

Let  $\mathcal{L}$  be the tessellation obtained from the intersection of all halfspaces bounded by  $P_1, \dots, P_n$  constructed by Algorithm 2 and projected onto a sphere. If the given tessellation  $\mathcal{T}$  is exactly the SLVD, the tessellation  $\mathcal{L}$  is identical to the tessellation  $\mathcal{T}$ , and arbitrary choice of Theorem 2 gives us the same SLVD. Otherwise, we get a different SLVD to  $\mathcal{T}$ .

The construction of the polyhedron  $\mathcal{P}$  corresponding to the tessellation  $\mathcal{T}$  shown in Algorithm 2 requires time complexity  $O(n \log n)$ .

## 4 SLVD Approximation Method

Note that almost all real world spherical tessellations cannot be represented exactly with an SLVD. In such instances, there exists a difference between  $\mathcal{T}$  and  $\mathcal{L}$ . In this section, we define an index for this discrepancy, and provide a method for minimizing the discrepancy to obtain the best fit SLVD.

### 4.1 Discrepancy

For the tessellations  $\mathcal{T}$  and  $\mathcal{L}$ , suppose that  $T_i$  corresponds to  $L_i$  for all  $i$ . For the  $i$ -th cell, let  $A_i = T_i \cap L_i$  be the intersecting convex spherical polygon. Suppose that the polygon is  $k_i$ -gon, which we denote by the anticlockwise sequence of vertices  $A_i = (A_{i,1}, \dots, A_{i,k_i})$ . Also, let  $\alpha_{i,1}, \dots, \alpha_{i,k_i}$  be the angles between two adjacent spherical  $k$ -gon edges. The area of the spherical polygon  $A_i$  is denoted by  $\text{area}(A_i)$ . If  $A_i = \emptyset$ ,  $\text{area}(A_i) = 0$ . Otherwise,  $\text{area}(A_i) = \sum_{j=1}^{k_i} \alpha_{i,j} - (k_i - 2)\pi$ .

Let  $A_{\mathcal{T}}, A_{\mathcal{L}}$  be the areas of spherical tessellations  $\mathcal{T}$  and  $\mathcal{L}$ , respectively. The difference between the areas for the tessellations  $\mathcal{T}$  and  $\mathcal{L}$  are defined by  $D_{\mathcal{T}} = A_{\mathcal{T}} - A$  and  $D_{\mathcal{L}} = A_{\mathcal{L}} - A$ , where  $A = \sum_{i=1}^n \text{area}(A_i)$ .

The discrepancy between  $\mathcal{T}$  and  $\mathcal{L}$  is defined by

$$\begin{aligned} \Delta_{\mathcal{T}, \mathcal{L}} &= (D_{\mathcal{T}} + D_{\mathcal{L}}) / (A_{\mathcal{T}} + A_{\mathcal{L}}) \\ &= 1 - \frac{1}{4\pi} \sum_{i=1}^n \left( \sum_{j=1}^{k_i} \alpha_{i,j} - (k_i - 2)\pi \right). \end{aligned} \quad (1)$$

### 4.2 The Procedure for Obtaining an SLVD Corresponding to a Given Tessellation

For the tessellation  $\mathcal{T}$ , we employ Algorithms 1 and 2 to construct the polyhedron and the SLVD. We compute the discrepancy by the following procedure.

1. For the tessellation  $\mathcal{T} = \{T_1, \dots, T_n\}$ , determine the area of each cell. The set of all areas is denoted  $\mathcal{A}_{\mathcal{T}} = \{\text{area}(T_1), \dots, \text{area}(T_n)\}$ .
2. Choose the cell  $i$  such that  $\text{area}(T_i) := \max \text{area}(T_j)$ ,  $j = 1, \dots, n$ .
3. Starting from the  $i$ -th cell, define the center of the first generator from the centroid  $p_i$  of the cell. Define the weight of the cell as zero which, is the plane tangent to the  $i$ -th cell at  $p_i$ .

4. Without loss of generality, choose the location of the second generator from a point inside this cell.
5. Employ Algorithms 1 and 2 to construct a polyhedron  $\mathcal{P}$  and project  $\mathcal{P}$  onto the center of the sphere  $U$ .
6. For each  $i$ , find the intersection  $A_i$  of two spherical polygons  $T_i, L_i$ , and compute the discrepancy  $D(\mathbf{x}) := \Delta_{\mathcal{T}, \mathcal{L}}$ .

### 4.3 Tessellation Fitting

To find the best fit SLVD, we find the minimum discrepancy from Equation (1). The discrepancy  $\Delta_{\mathcal{T}, \mathcal{L}}$  is related to the angle of the intersecting spherical polygons and the number of spherical polygon vertices which will change when the SLVD changes.

The main factor which affects the SLVD  $\mathcal{L}$  is the alignment of planes  $P_1, \dots, P_n$  composing the polyhedron  $\mathcal{P}$  of  $\mathcal{L}$ .

Let the plane  $P_i$  be

$$P_i : a_i x + b_i y + c_i z = d_i. \quad (2)$$

Since the plane  $P_i$  does not pass through the sphere's origin, the plane equation (2) can be expressed as

$$P_i : A_i x + B_i y + C_i z = 1. \quad (3)$$

The parameters  $A_i, B_i, C_i$  involve the alignment of the plane  $P_i$ . Therefore, the adjustment of the SLVD of  $n$  planes requires the parameters  $\mathbf{x} = (A_1, \dots, A_n, B_1, \dots, B_n, C_1, \dots, C_n)$ .

We define the discrepancy function of  $\mathbf{x}$  using the procedure in Section 4.2 by  $D(\mathbf{x}) := \Delta_{\mathcal{T}, \mathcal{L}}$ .

However, it is complicated to find the relation between the planes and the angles as defined in (1). Therefore, we employ the Nelder-Mead method to find  $\min D(\mathbf{x})$  numerically, where  $D(\mathbf{x})$  is computed in a pointwise manner. The details of the method are provided in Chapter 18 of [1].

In brief, we first construct a simplex  $\mathcal{S} = \{S_1, \dots, S_{m+1}\}$  of  $m$  dimensional parameter space composed of  $m + 1$  vertices and compute the discrepancy function value for each simplex vertex. For each iteration, the worst vertex which yields the maximum discrepancy will be replaced with a new vertex by reflection, expansion, or contraction of the centroid points among the remaining  $m$  vertices with ratios  $\alpha_R, \alpha_E, \alpha_C$ . If we cannot replace the worst vertex, we shrink the simplex to the vertex that has the smallest discrepancy, with ratio  $\alpha_S$ . Therefore, the direction of the simplex is moved to the local minimum of the discrepancy function. The iteration is terminated if it meets the convergence criteria.

For a tessellation of  $n$  cells, the number of parameters considered is  $m = 3n$ . Therefore, we consider a simplex of  $3n + 1$  vertices. The convergence condition is determined by the number of iterations.

We choose the first vertex  $S_1$  of the initial simplex from the parameters of the planes constructed by the procedure provided in Section 4.2. The remaining  $3n$  vertices are determined by  $S_i = S_1 + \beta \mathbf{e}_i$ , where  $\mathbf{e}_i$  is the  $i$ -th standard basis of  $\mathbb{R}^m$ , and  $\beta$  is the positive number.

## 5 Experiments and Numerical Results

We conducted the experiments using simulated data. We used Wolfram Mathematica®10.3 to implement the algorithms.

To validate Algorithms 1 and 2, we generated the SLVD for the tessellation  $\mathcal{T}$ . From the experiments, we can find the tessellation  $\mathcal{L}$  which coincides with the tessellation  $\mathcal{T}$ . The accuracy was measured using the discrepancy function value.

For the approximation, we separated the experiment into 2 parts, testing the validity of the framework, and fitting an SLVD to an arbitrary tessellation. The tessellation had 10 cells. We set  $\alpha_R = 1, \alpha_E = 2, \alpha_C = -0.5, \alpha_S = -0.5$ , and iterated 4,000 times.

We checked the validity of the approximation framework by generating the SLVD. After that, we perturbed some of the initial plane parameters, which yielded a different SLVD to the initial one. After that, we employed the Nelder-Mead method to optimize the discrepancy function. From the experiment, we found that we can find the local minimum that has the smallest discrepancy, and the estimated SLVD converges to the tessellation. The results for the discrepancy minimization are shown in Figure 1 (left).

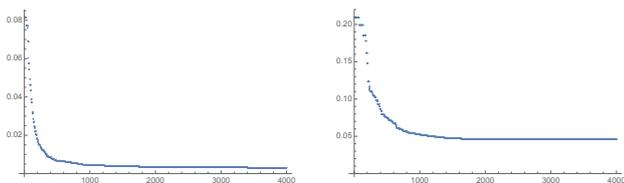


Figure 1: The change in the discrepancy of the initial simplex when (left) the initial simplex vertex was perturbed; (right) arbitrary spherical tessellation

After that, we conducted the experiment for an arbitrary spherical tessellation and employed the Nelder-Mead method. From the experiment, we can find the parameter that best fits the given tessellation. The change in the discrepancy is shown in Figure 1 (right), and the results for the fitted SLVD is shown in Figure 2.

## 6 Concluding Remarks

We proposed a framework for finding an SLVD that can be fitted to a given spherical tessellation. The

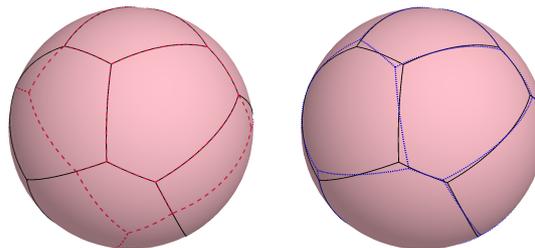


Figure 2: (Left) Solid lines: the spherical tessellation; dashed lines: SLVD from the  $S_1$  parameter; (Right) dotted lines: the fitted SLVD from the optimization

optimization of the discrepancy function was shown to rely on the orientation of the polyhedron of the SLVD.

The proposed framework can be used for recognizing whether the given tessellation is close to the Voronoi diagram. For the 3D real world spherical tessellation, we can extract the tessellation, project it onto a sphere and employ our framework. However, similar to [3, 4], this can be considered as a new problem when we use a planar photographic image instead of the information of 3D tessellation.

## Acknowledgments

S.C. acknowledges the MIMS Ph.D. Program of the Meiji Institute for Advanced Study of Mathematical Sciences, Meiji University. This research was partly supported by the Grant-in-Aid for Basic Research No. 24360039 and Exploratory Research No.15K12067 of MEXT.

## References

- [1] Arora, J. S.: Introduction to optimum design (Third Edition). Elsevier, USA (2012)
- [2] Aurenhammer F.: Power Diagram: Properties, Algorithms, and Applications, SIAM J. Comput., 16, 78–96 (1987)
- [3] Chaidee S., and Sugihara K.: Approximation of Fruit Skin Patterns Using Spherical Voronoi Diagram, Pattern Anal Appl, DOI: 10.1007/s10044-016-0534-2 (2016)
- [4] Chaidee S., and Sugihara K.: Fitting Spherical Laguerre Voronoi Diagrams to Real-World Tessellations Using Planar Photographic Images, accepted
- [5] Chaidee S., and Sugihara K.: Recognition of the Spherical Laguerre Voronoi Diagram, in preparation
- [6] Duan Q., Kroese D. P., Brereton T., Spettl A., Schmidt V.: Inverting Laguerre Tessellations, Comput. J. 57, 1431–1440 (2014)
- [7] Imai H., Iri M., Murota K.: Voronoi Diagram in the Laguerre Geometry and its Applications. SIAM J. Comput. 14, 93–105 (1985)
- [8] Sugihara K.: Laguerre Voronoi Diagram on the Sphere. J. Geom. Graph. 6, 69–81 (2002)