Available online at www.sciencedirect.com

SciVerse ScienceDirect

International Meshing Roundtable 00 (2016) 000–000

Procedia Engineering

www.elsevier.com/locate/procedia

25th International Meshing Roundtable (IMR25)

Discretized Riemannian Delaunay triangulations

M. Rouxel-Labbé[a,b,*], M. Wintraecken[b], J.-D. Boissonnat[b]

[a]*GeometryFactory, 1501 route des Dolines, 06560 Valbonne – Sophia Antipolis, France*
[b]*INRIA Sophia-Antipolis, 2004 Route des Lucioles, 06902 Valbonne, France*

Abstract

Anisotropic meshes are desirable for various applications, such as the numerical solving of partial differential equations and graphics. In this paper, we introduce an algorithm to compute discrete approximations of Riemannian Voronoi diagrams on 2-manifolds. This is not straightforward because geodesics, shortest paths between points, and therefore distances cannot in general be computed exactly.

Our implementation employs recent developments in the numerical computation of geodesic distances and is accelerated through the use of an underlying anisotropic graph structure.

We give conditions that guarantee that our discrete Riemannian Voronoi diagram is combinatorially equivalent to the Riemannian Voronoi diagram and that its dual is an embedded triangulation, using both approximate geodesics and straight edges. Both the theoretical guarantees on the approximation of the Voronoi diagram and the implementation are new and provide a step towards the practical application of Riemannian Delaunay triangulations.

Peer-review under responsibility of organizing committee of the 25th International Meshing Roundtable (IMR25).

Keywords: Anisotropic mesh generation; Delaunay triangulation; Voronoi diagram; Geodesic distance;

1. Introduction

Anisotropic simplicial meshes are triangulations whose elements are elongated along prescribed directions. Anisotropic meshes are known to be well suited when solving PDE's [1–3]. They can also significantly enhance the accuracy of a surface representation if the anisotropy of the mesh conforms to the curvature of the surface [4].

Many anisotropic mesh generation methods are based on Riemannian manifolds and create meshes whose elements adapt locally to the size and anisotropy prescribed by a metric field. Several authors have considered Voronoi diagrams using anisotropic distances to obtain triangulations adapted to an anisotropic metric field. These authors hoped to build on the well established concepts of the Euclidean Voronoi diagram and its dual structure, the Delaunay triangulation, and their many theoretical and practical results [5].

* Corresponding author.
E-mail address: mael.rouxel-labbe@inria.fr

Peer-review under responsibility of organizing committee of the 25th International Meshing Roundtable (IMR25).

The Voronoi diagram of a set of points $\mathcal{P} = \{p_i\}$ in a domain Ω is a partition of Ω in a set of regions $\{V_d(p_i)\}$. $V_d(p_i)$ is called the *Voronoi cell* of the *seed* p_i and is defined by

$$V_d(p_i) = \{x \in \Omega \mid d(p_i, x) \leq d(p_j, x),\ \forall p_j \in \mathcal{P} \backslash p_i\}.$$

Du and Wang [6] defined the anisotropic Voronoi cell of a point p_i in a domain Ω endowed with a metric field g as

$$V_{DW,g}(p_i) = \{x \in \Omega \mid d_{g(x)}(p_i, x) \leq d_{g(x)}(p_j, x),\ \forall p_j \in \mathcal{P} \backslash p_i\}.$$

Canas and Gortler [7,8] and Cheng et al. [9] proved that under sufficient sampling conditions, the dual of the DW-diagram is an embedded triangulation for planar and surface domains. Labelle and Shewchuk [10] introduced slightly different Voronoi cells:

$$V_{LS,g}(p_i) = \{x \in \Omega \mid d_{g(p_i)}(p_i, x) \leq d_{g(p_j)}(p_j, x),\ \forall p_j \in \mathcal{P} \backslash p_i\}.$$

While the LS-diagram obtained is easier to study and trace since the bisectors are quadrics, the associated theory only covers the planar setting and its extension to higher dimensions faces inherent difficulties.

Both these diagrams aim to approximate the Riemannian Voronoi diagram (RVD) whose cells are defined by

$$V_g(p_i) = \{x \in \Omega \mid d_g(p_i, x) \leq d_g(p_j, x), \forall p_j \in \mathcal{P} \backslash p_i\}. \tag{1}$$

where $d_g(p, q)$ is the geodesic distance on the Riemannian manifold. Judged too expensive, this Voronoi diagram has not been the subject of any practical work. However, the RVD provides a more favorable framework to develop a theoretically sound method. Additionally, it is devoid of orphan cells, which are regions of the domain disconnected from their seed. These orphans, which usually prevent the dual from being an embedded triangulation, appear in the other anisotropic Voronoi diagrams previously mentioned.

Leibon studied the dual of the Riemannian Voronoi diagram in 2D [11] and gave conditions such that the Riemannian Delaunay triangulation is an embedded triangulation. Together with Letscher [12], he extended that study to Riemannian Delaunay triangulations in any dimension, but their analysis was shown to be incomplete by Boissonnat et al. [13]. Dyer et al. [14] improved the 2D and surface work of Leibon with better bounds.

Contribution

We approach the Riemannian Voronoi diagram and its dual with a focus on both practicability and theoretical robustness. We introduce an algorithm to compute a discretized Riemannian Voronoi diagram that is generic with respect to the choice of the method used to compute geodesic distances. The construction of our diagram uses an anisotropic graph structure to greatly reduce the computational time.

Our algorithm is theoretically sound and builds upon the theoretical results of the Riemannian Delaunay triangulation. We show that, under sufficient conditions, the dual of the Riemannian Voronoi diagram can be embedded as a triangulation (mesh) of the point set, with edges drawn as geodesic arcs, or even as straight edges. These results are detailed in our companion report [15] and we only give here an overview.

Finally, we present an experimental study of our method in the planar and surface cases.

2. Related work

Due to its wide array of practical applications, anisotropic mesh generation has received a lot of attention and several classes of methods have been proposed.

Delaunay-based methods. The generation of isotropic Delaunay mesh is usually achieved through an iterative refinement algorithm. The common choice to refine a simplex is to choose its circumcenter as Steiner point. Borouchaki et al. [16] introduced the anisotropic Delaunay kernel, iteratively generating anisotropic meshes with a modified Bowyer-Watson algorithm. A different approach to anisotropic Delaunay triangulations was proposed by Boissonnat et al. [17], called locally uniform anisotropic meshes. In their algorithm, the star of each vertex v is composed of simplices that are Delaunay for the metric at v. Each star is built independently and the stars are stitched together in the hope of creating an anisotropic mesh. The algorithm works in any dimension, can handle complex geometries and provides guarantees on the quality of the simplices of the triangulation. However, the number of vertices required to achieve consistency in the star set can be very large for highly distorted metric fields, making the algorithm unattractive.

Energy-based methods. Optimal Delaunay triangulations (ODT) were introduced by Chen et al. [18,19] as the triangulation that minimizes the interpolation error $\|u - \hat{u}\|$ of a function u and its piecewise interpolation $\hat{u}$ over the mesh. First applied to isotropic mesh generation using $u(x) = x^2$ [20,21], ODT was naturally extended to anisotropic metric fields derived from the Hessian of a convex function through the use of weighted Delaunay triangulations [22,23]. However, as shown by Boissonnat et al. [24], a Riemannian metric field cannot always be expressed as the Hessian of a convex function. Chen et al. [25] attempted to remedy this restriction by locally approximating the metric field with local convex functions, but the algorithm may fail to converge or achieve the desired anisotropy. Fu et al. [26] tweaked this approach by including feedback from the neighboring vertices in the expression of the local energy, with better experimental results.

Many other methods have been considered, such as isometric and conformal embeddings [27] and repulsive vertex repositioning [28,29] techniques, but they possess shortcomings in theoretical or practical aspects.

Geodesic Voronoi diagrams. Peyré and Cohen [30] used isotropic geodesic Voronoi diagrams on surfaces as a tool to achieve isotropic adaptive remeshing. Cao et al. [31] studied the digital Voronoi diagram, a discrete structure diagram obtained by flooding from the seed and proved that they obtain a geometrically and topologically correct dual triangulation. Their work is however limited to the 2D isotropic setting.

3. Background

3.1. The Riemannian metric

A metric in $\mathbb{R}^d$ is defined by a symmetric positive definite quadratic form represented by a $d \times d$ matrix G. The distance between two points p and q in Ω, as measured using the metric G, is defined by

$$d_G(x, y) = \sqrt{(x - y)^t G (x - y)}.$$

Given the symmetric positive definite matrix G, we denote by F any matrix such that $\det(F) > 0$ and $F^t F = G$. F is called a *square root* of G. Although not unique, we shall assume that F is fixed. Note that

$$d_G(x, y) = \sqrt{(x - y)^t F^t F (x - y)} = \|F(x - y)\|,$$

where $\|\cdot\|$ stands for the Euclidean norm.

A Riemannian *metric field* g, defined on a domain $\Omega \subset \mathbb{R}^d$, associates a metric $g(p) = G_p = F_p^t F_p$ to any point $p \in \Omega$. We always assume that g is Lipschitz continuous.

Labelle and Shewchuk [10] introduced the concept of *distortion* between two points p and q of Ω:

$$\psi(p, q) = \psi(G_p, G_q) = \max\left\{\left\|F_p F_q^{-1}\right\|, \left\|F_q F_p^{-1}\right\|\right\},$$

Note that $\psi(G_p, G_q) \geq 1$ with $\psi(G_p, G_q) = 1$ when $G_p = G_q$. A fundamental property of the distortion is to relate the two distances d_{G_p} and d_{G_q}. Specifically, for any pair x, y of points, we have

$$\frac{1}{\psi(p, q)}\, d_{G_p}(x, y) \leq d_{G_q}(x, y) \leq \psi(p, q)\, d_{G_p}(x, y).$$

3.2. Geodesics

In the following, we assume that our domain has no boundaries and is geodesically complete in order to avoid punctures. A path between two points x, y of Ω is a function $\gamma : [0, 1] \to \Omega$ such that $\gamma(0) = x$ and $\gamma(1) = y$. If Ω is endowed with the Riemannian metric field g, the Riemannian length of a path γ is defined as

$$l_g(\gamma) = \int_0^1 \sqrt{\gamma'(t)^t g(\gamma(t)) \gamma'(t)}\, dt.$$

Let $\Gamma(x, y)$ be the set of paths from x to y on Ω: $\Gamma(x, y) = \{\gamma : [0, 1] \to \Omega \mid \gamma(0) = x \text{ and } \gamma(1) = y\}$. The geodesic distance between two points is given by $d_g(x, y) = \min_{\gamma \in \Gamma(x,y)} l_g(\gamma)$.

We say that $\mathcal{P} = \{p_i\}$ is an (ϵ, μ)-net with respect to the metric field g if, for all $x \in \Omega$, there exists $p_i \in \mathcal{P}$ such that $d_g(p_i, x) < \epsilon$ and for all $p_i, p_j \in \mathcal{P}$, $d_g(p_i, p_j) \geq \mu$.

3.3. Nets

The structures of interest will be built from a finite set of points $\mathcal{P} \subset \Omega$, which we call *samples* or *seeds*. If Ω is a bounded domain, $\mathcal{P}$ is an *ϵ-sample set* for Ω with respect to g if $d_g(x, \mathcal{P}) < \epsilon$ for all $x \in \Omega$. The set $\mathcal{P}$ is μ-separated with respect to g if $d_g(p, q) \geq \mu$ for all $p, q \in \mathcal{P}$. We assume that the separation parameter μ is proportional to ϵ, thus there exists a positive λ, with $\lambda \leq 1$, such that $\mu = \lambda\epsilon$. If $\mathcal{P}$ is an ϵ-sample set that is μ-separated, we say that $\mathcal{P}$ is an *(ϵ, μ)-net*.

3.4. The Riemannian Voronoi diagram

The cell $V_g(p_i)$ of a seed p_i in the Riemannian Voronoi diagram (RVD) has been defined in Equation 1. A *Voronoi face* is the intersection of a set of Voronoi cells. If the dimension of the face is 0 or 1, we respectively use the terms *Voronoi vertex* and *Voronoi edge*. The *bisector* of two seeds p and q is the set $\mathcal{BS}(p, q) = \{x \in \Omega, d(p, x) = d(q, x)\}$.

The *nerve* of a finite covering C is $\text{Nrv}(C) = \{D \subseteq C \mid \cap D \neq \emptyset\}$. A geometric realization of the nerve is a (geometric) simplicial complex $\mathcal{K}$ and a bijection φ between $\text{Nrv}(C)$ and $\mathcal{K}$ that preserves the inclusion, that is τ and σ are simplices of $\mathcal{K}$ such that $\tau \subset \sigma$ if, and only if, there exists $D_\sigma = \varphi(\sigma)$ and $D_\tau = \varphi(\tau)$ in $\text{Nrv}(C)$ such that $D_\sigma \subset D_\tau$. In the following, we identify $\mathcal{K}$ and the realization of the nerve. The nerve of the RVD is $\text{Nrv(RVD)} = \{\{V_g(p_i)\} \mid \cap_i V_g(p_i) \neq \emptyset, p_i \in \mathcal{P}\}$. A geometric realization of Nrv(RVD) whose vertices are $\mathcal{P}$ is called a *dual* of the RVD. If there exists an abstract homeomorphism between the nerve $\text{Nrv}(C)$ and the domain, then we say that the realization of the nerve is a *triangulation*. If there exists an embedding, which provides a homeomorphism, of the dual whose edges are geodesic arcs (resp. straight edges in the ambiant space), such an embedding is called the *Riemannian Delaunay triangulation* (resp. *Straight Riemannian Delaunay Triangulation*) of $\mathcal{P}$.

3.5. Power protection

The concept of protection and power-protection of point sets and simplices were introduced by Boissonnat et al. [32] to provide lower bounds on the thickness and quality of simplices.

Let σ be a simplex whose vertices belong to $\mathcal{P}$ and whose (geodesic) circumscribing ball is noted $B_g(c, r)$, where c is the circumcenter and r is the circumradius. For $0 \leq \delta \leq r$, we associate to $B_g(c, r)$ the dilated ball $B_g^{+\delta} = B_g(c, \sqrt{r^2 + \delta^2})$. We say that σ is *δ-power protected* if $B_g^{+\delta}$ is empty of any vertex in $\mathcal{P} \setminus \text{Vert}(\sigma)$. If all the simplices of $\text{Del}(\mathcal{P})$ are δ-power protected then $\mathcal{P}$ is δ-power protected.

4. The discretized Riemannian Voronoi diagram

We now introduce our algorithm that approximates the Riemannian Voronoi diagram.

4.1. Computing geodesics

The computation of geodesic distances and paths on discrete domains is in itself a difficult task to which much work has been dedicated in the last decades. Historically, geodesic distances were first evaluated with the Euclidean metric naturally derived from the embedding of the domain in $\mathbb{R}^d$. Many approaches exist depending on the expectations on the geodesic distance, exact or approximate, heuristic or with theoretical guarantees [33].

Algorithms have more recently been introduced to compute geodesic distances when the domain is endowed with a prescribed metric field g that exhibits anisotropy. We detail some methods that we empirically evaluated. Konukoglu et al. [34] proposed a heuristic method to extend the fast marching method to the anisotropic setting by adding a recursive correction scheme. However, this implies loss of the theoretical convergence of the fast marching

method and the recursivity induces a heavy computational cost. Mirebeau [35] introduced a grid-based modification of the fast marching method with theoretical guarantees, but the underlying grid structure does not extend easily to manifolds. Campen et al. [36] proposed a heuristic approach based on a modified Dijkstra algorithm. While their algorithm does not offer guarantees, it is nevertheless fast and precise with little requirement on the quality of the discrete domain. Lastly, a very different approach was considered by Crane et al. [37], who solve linear elliptic equations to compute a global geodesic distance map, obtaining an approximate distance between any two points of the domain. The algorithm is extremely fast compared to front propagation-based methods, but its parameters require delicate tuning and the approximation was generally much worse than the methods previously mentioned, making it unattractive for our work.

4.2. Generating the diagram

We refer to the discretization of the domain Ω upon which geodesic distances are computed as the *canvas*, and denote it C. Notions related to the canvas explicitly carry *canvas* in the name (for example, an edge of C is a *canvas edge*). The easiest canvas to consider is an isotropic triangulation of the domain. The generation of this triangulation is performed through a classic refinement algorithm driven by a *sizing field* constraint s: if a simplex σ of C has a circumradius R_σ larger than $s(c_\sigma)$, where c_σ is the circumcenter of σ, then it is refined by inserting c_σ. We will show how the sizing field is chosen in Section 6.

The procedure used to generate our approximate diagram is similar to the classic Dijkstra algorithm: each seed is associated a color (an integer) and each canvas vertex is assigned three fields:

- a status: KNOWN, TRIAL or FAR (initialized at FAR),
- a distance: the current smallest geodesic distance to a seed (initialized at ∞),
- a color: the color of the closest Voronoi seed.

The initialization of the canvas with the seed set is done by locating each seed on the canvas. The vertices of the simplex containing the seed are then attributed TRIAL as status, their color is set to the color of the seed and their distance is set to the distance to the seed.

The vertices of the canvas are colored through simultaneous Dijkstra-like front propagations from all the seeds: the vertices are sorted in a priority queue Q that is kept ordered by increasing distance to the vertex's closest seed. At each iteration, the point with lowest distance to a seed becomes KNOWN and is removed from the queue. The distances at its neighbors that are not KNOWN are evaluated again, and the queue is updated.

When the queue is empty, the result is a partition of the vertices of the canvas. To each seed p_i, we associate a *discrete cell* $V^d(p_i)$ made of all canvas simplices who have at least one vertex of the color of p_i. We call the set of these cells the *discretized Riemannian Voronoi diagram* (DRVD).

Refinement of the DRVD

New seeds can easily be inserted in an existing DRVD: the Voronoi cell of a new point is obtained by spreading from this new seed with a single front propagation algorithm. When the sampling of the point set is already dense on the domain, the insertion is a local operation.

4.3. Extracting the nerve of the DRVD

To construct a geometric realization of the nerve $\mathcal{N} = \mathrm{Nrv}(\mathrm{DRVD})$, we must first extract the connectivity of the DRVD. Using a triangulation as canvas allows us to obtain $\mathcal{N}$ easily: each canvas k-simplex σ of C has $k+1$ vertices $\{v_0, \dots, v_k\}$ with respective colors $\{c_0, \dots, c_k\}$, which correspond to seeds $\{p_{c_0}, \dots, p_{c_k}\} \in \mathcal{P}$. If all colors c_i are different, then the intersection $D = \bigcap_{i=0\dots k} V^d(p_{c_i})$ is non-empty and D belongs to the nerve (see Figure 2). We say that the simplex σ *witnesses* (or is a witness of) D. If for all $D \in \mathcal{N}$ there exists $\sigma \in C$ such that σ witnesses D and if for all $\sigma \in C$ the simplex witnessed by σ belongs to $\mathcal{N}$, then we say that C *captures* $\mathcal{N}$.

5. Fast DRVD computations

The computation of geodesic distances constitutes the expected bottleneck in the construction of the DRVD. This cost can be reduced in two independent ways: improving the speed of geodesic computations and decreasing the amount of geodesic distance estimations. We rely on previous work for the estimation of the geodesic distances and improving the current methods goes beyond the scope of this paper. We can however dramatically decrease the number of geodesic computations required by changing the nature of the canvas, as we shall explain now.

Assume that a number of consecutive canvas edges form a line in the domain. If the metric were constant along this line, the accuracy of the distance computation would not be improved by adding edges. Extrapolating from this simple example, one can intuit that it is preferable for the canvas not to be an isotropic discretization of the domain: directions in which the metric field does not vary too much can be discretized only sparsely, as long as other constraints are honored (see the theoretical requirements in Section 6). Unfortunately, an anisotropic mesh is exactly what we want to construct. However, for geodesic distance computations, a triangulation is not required and a graph suffices. The locally Delaunay algorithm of Boissonnat et al. [17,38] gives such a graph based on the connectivity of its stars. We recall in the next section the main principles of this algorithm.

5.1. The local Delaunay algorithm

The local Delaunay algorithm, introduced by Boissonnat et al. [17,38], attempts to extend the concept of Delaunay triangulation to non Euclidean settings by computing independently at each vertex its connectivity to other vertices. The algorithm constructs for every vertex $v \in \mathcal{P}$ the Delaunay triangulation in the metric M_v of v, which is the image through the stretching transformation F_v^{-1} of the Euclidean Delaunay triangulation $\mathrm{Del}(F_v(\mathcal{P}))$. The *star* S_v of v in $\mathrm{Del}_v(\mathcal{P})$ is the sub-complex of $\mathrm{Del}_v(\mathcal{P})$ formed by the simplices that are incident to v. The collection of all the stars is called the (anisotropic) star set of $\mathcal{P}$, noted $\mathrm{S}(\mathcal{P})$.

Since the connectivity of each star is set according to the metric at the center of the star, *inconsistencies* may exist amongst the stars of the sites: a simplex σ that appears in $\mathrm{Del}_v(\mathcal{P})$ might not appear in $\mathrm{Del}_w(\mathcal{P})$ with $v, w \in \sigma$ (see Figure 1, left). The algorithm refines the set of sites $\mathcal{P}$ until each simplex conforms to a set of a criteria (size, quality, etc.) and there are no more inconsistencies amongst the stars, which is proven to happen.

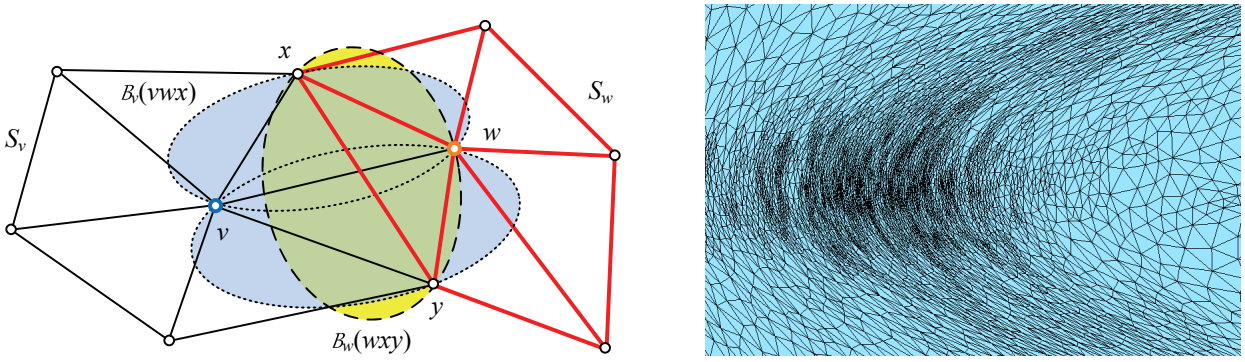

Fig. 1. On the left, two stars and an inconsistent configuration, [38]. On the right, the local Delaunay algorithm required 4430 vertices to construct a consistent star set of this region endowed with the hyperbolic shock.

Unfortunately, the algorithm requires in practice a large number of vertices to solve inconsistencies when dealing with highly-distorted metric fields, making it unattractive to use. Figure 1 (right) shows an example of metric field that is difficult for the star set. The algorithm required 4430 vertices to achieve consistency, a number considerably higher than our method in identical settings, as shown in Section 8.

5.2. Using star sets as canvasses

While the local Delaunay algorithm is not competitive, its star set structure provides, regardless of its consistency, a graph that can be used to compute geodesic distances. Since the stars conform to the metric field, this anisotropic

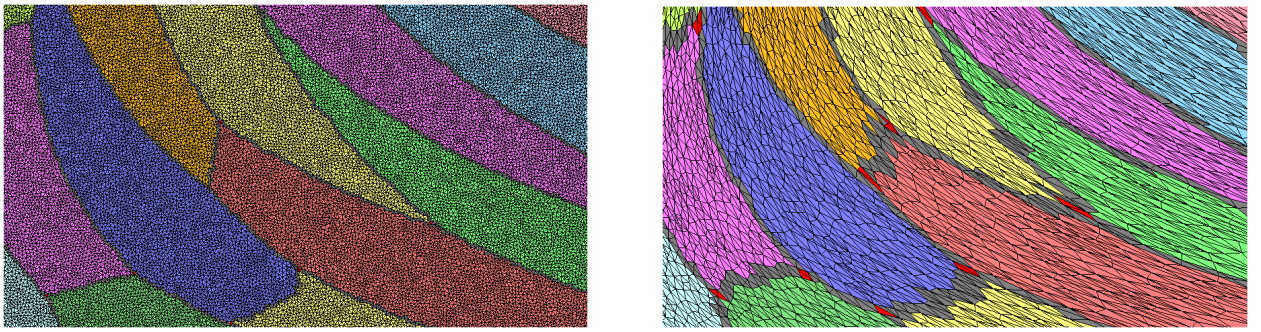

Fig. 2. Discretized Riemannian Voronoi diagram of the same seed set on two different canvas. On the left, an isotropic triangulation (1M vertices). On the right, an inconsistent star set (40k vertices). The nerves of both DRVD are identical. The canvas simplices colored in red are witnesses of Voronoi vertices. The canvas simplices colored in grey are witnesses of Voronoi edges.

canvas is much sparser than an equivalent isotropic triangulation. This difference in density is accentuated as the anisotropy ratio increases. Indeed, the Voronoi cells become very thin for extreme anisotropy ratios and require a very dense sampling when using isotropic canvasses. On the other hand, the amount of vertices in a star set-based canvas can be kept at around two orders of magnitude that of the point set regardless of the anisotropy. In other words, it suffices in practice that every Riemannian Voronoi cell contains around 100 canvas vertices to capture the nerve. Finally, it is fast to compute and can be trivially parallelized.

The use of the star set as canvas therefore greatly decreases the computational time as the amount of vertices that make up the canvas is drastically reduced without any change in the extracted nerve (see Figure 2 and Section 8). This approach is not unlike the stencil-based geodesic distance computation method of Mirebeau [35] previously described. Indeed, our stars can be considered as non-uniform stencils. Although they offer less guarantees than Mirebeau's stencils, Delaunay stars allow to handle easily the case of highly rotational manifolds.

6. Theory

We give in this section a summary of the theoretical background of our work. The results below can be found in detail in our companion report [15]; we indicate between parantheses the corresponding result in our companion paper.

We show that the DRVD and the RVD are combinatorial equivalent under some conditions and provide explicit requirements on the point set ($\mathcal{P}$ must be a power protected net) and on the density of the canvas such that both the Riemannian Delaunay Triangulation (RDT) and the Straight Riemannian Delaunay Triangulation (SRDT) are embedded triangulations in the domain.

Equivalence of the DRVD and RVD. The combinatorial equivalence of the RVD and the DRVD are obtained in steps. We begin with the basic case of a two dimensional flat domain endowed with the Euclidean metric field. The metric field is then made generic while the domain stays flat. Finally, curvature of the manifold is incorporated to prove the result in the case of a generic 2-manifold endowed with a generic metric field.

Theorem 6.1 (Report: Section 12)**.** *Let $\mathcal{P}$ be a δ-power protected point (ϵ,μ)-net with respect to g. Let C be the canvas, with a sizing field s. If ϵ and s are sufficiently dense and $\delta > \delta_0$, where $0 < \delta_0 < \epsilon$ is a fixed constant that depends on the metric field and on the manifold, then the RVD and the DRVD are combinatorially equivalent.*

Embedding of the RDT. We mostly rely on previously established work for the correctness of the RDT (see [11,39]). In the case of 2-manifolds, the embedding of the RDT is achieved with light requirements.

Theorem 6.2 (Report: Theorem 10.20)**.** *Let $\mathcal{V}_{\{p_i\}}$ be the Riemannian Voronoi diagram of the point set $\{p_i\}$ on an oriented Riemannian surface M. Suppose that no three vertices lie on a geodesic. Assume that all Voronoi cells V_{p_i} lie within balls of radius less than a sixth of the convexity radius. Then the Riemannian realization of the Delaunay*

complex of the Riemannian Voronoi diagram induces a piecewise smooth homeomorphism from the Delaunay complex to the surface M.

Since this result applies to the RVD, we must make sure that the conditions mentioned above such that the DRVD and the RVD are combinatorially equivalent are met.

Embedding of the SRDT. To prove of the correctness of the SRDT, we show that geodesic paths and straight edges between seeds are close when the distortion is small and that the deformation from one to another induces no inversion or intersection of simplices. The core lemma of this proof is presented below.

Lemma 6.3 (Report: Lemma 11.1)**.** *Let $U \subset \Omega$ be open and g and g' two Riemannian metric fields on U, with g' constant. Let $\psi_0 \geq 1$ be a bound on the metric distortion. Suppose that U is included in a ball $B_g(p_0, r_0)$, with $p_0 \in U$ and $r_0 \in \mathbb{R}^+$, such that $\forall p \in B(p_0, r_0), \psi(g(p), g'(p)) \leq \psi_0$. Let us further denote the geodesics connecting x and y with respect to g and g' as γ_g and $\gamma_{g'}$. If $z \in \gamma_g$ then $d_{g'}(z, \gamma_{g'}) \leq a$, where*

$$a = \frac{1}{2}\sqrt{\psi_0^2 - \frac{1}{\psi_0^2}}\, d_g(x, y).$$

This result can be visualized as the existence of a protecting tube around the straight edge in which the geodesic path lies.

7. Constructing the point set $\mathcal{P}$

Our theory exposes requirements on the canvas C (through its sizing field) and on the point set $\mathcal{P}$ (through its density and protection) such that the duals of the DRVD are embedded triangulations. The former is easily achieved through refinement. We now explain how a δ-power protected ϵ-sample can be built iteratively through a refinement algorithm that is similar to that of the locally uniform Delaunay method proposed by Boissonnat et al. [17].

In the local Delaunay algorithm, the obstacle that potentially prevents the star set from being a triangulation is the presence of inconsistencies: a configuration where adjacent stars whose connectivity is incompatible. It is shown that, once the distortion is small enough, inconsistencies are only created by *quasi-cosphericities*, the existence $n + 1$ points almost living on a sphere. Steiner points can then be chosen in a small zone around the circumcenter called the *picking region* such that no quasi-cosphericities are created (see Figure 3) as the thickened Delaunay spheres encroaching on the picking region do not cover it.

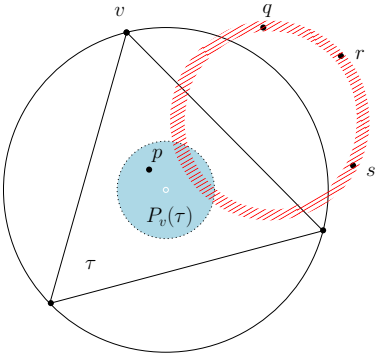

Fig. 3. In blue, the picking region of the simplex τ, [38].

Power protection and the absence of quasi-cosphericities are two aspects of the same concept. Indeed, both can be thought of as thickened circumscribing balls that must be empty of any point other than the vertices of the circumscribed simplex. In Boissonnat et al.'s setting, this thickening depends on the metric distortion and goes to 0 as the distortion becomes small, which makes the core of their proof of the (eventual) existence of valid Steiner points. In our context, the thickening is explicitly given by the δ coefficient of the power protection and must be bigger than a value δ_0. However, this bound δ_0 also goes to 0 as the distortion becomes small, which happens as the sampling gets denser. The same smart refinement algorithm that prevents quasi-cosphericity in their method will therefore create here a power protected point set.

The initialization of the point set $\mathcal{P}$ is made with a few random points on the domain. The refinement algorithm is made of two criteria:

- A sizing criterion to obtain an ϵ-sample. Simplices are refined by the insertion of their circumcenter.
- A protection criterion to obtain a δ-power protected point set. The Steiner point is chosen with a pick_valid routine that selects a *valid* refinement point (see [40] for details). If no good refinement point exists (yet) in the

picking region, the circumcenter is instead inserted, which lowers the distortion and increases the probability of finding a valid Steiner point at a further step.

The μ-separation of the point set is implicitly obtained with this choice of Steiner points and can be computed explicitly through the insertion radius, as done in [40]. In similar fashion to the local Delaunay algorithm, the refinement algorithm can be shown to terminate and to produce a δ-power protected (ϵ, μ)-net.

In practice, neither the sampling nor the canvas need to be as dense as our theory requires them to be and a farthest point refinement can be used to generate the point set.

8. Results

We now present some results for 2D and planar domains. The different geodesic distance computation algorithms described in Section 4.1 were implemented and tested and while the method proposed by Campen et al.'s does not offer any theoretical guarantee, its precision and convergence were in practice satisfactory and its speed a significant advantage. CGAL [41] was used to generate isotropic Delaunay canvases and our newer implementation of the star set algorithm for star sets.

We consider two types of metric fields: artificial ones, chosen for their difficulty, and curvature-based metric fields as more realistic examples. These metric fields are precisely defined in our companion report [15].

Computation time

The extraction of the nerve is performed on the fly during the coloring the canvas. This operation and the construction of the duals are negligible in cost compared to computing geodesic distances.

Figure 5 uses an isotropic canvas of a 4 by 4 square and a swirly metric field. The canvas is composed of 1, 286, 862 vertices and took 29 seconds to generate. The computation of the RDT and SRDT took almost 6 hours. Comparatively, computing the RDT and SRDT using a star set as canvas only required 121, 264 stars (vertices) and took 6 minutes, including 3 minutes to generate the star set. This improvement becomes even more significant as the anisotropy increases.

Duals

The swirly and hyperbolic shock metric field (see Figures 5 and 8) are good examples of metric fields that are typically difficult for the non-geodesic Voronoi diagrams of Du and Wang and Labelle and Shewchuk. Indeed, they are characterized by both noticeable anisotropy ratios and a highly rotational vector field (vectors given at a point by the eigenvectors of the metric at this point). These regions are easily populated by orphan cells and an improper dual for the VS or LS diagram if the point set is not sufficiently dense. On the other hand, this issue cannot appear in the RVD and both the RDT and the SRDT quickly become triangulations.

Preprocessing the metric field, typically with a Laplacian smoothing, is a technique also often used (see for example [26]) that lowers the fidelity to the input metric field and can even nullify the anisotropy, for example in the highly rotational region of the hyperbolic shock, resulting in isotropic elements. On the other hand, our algorithm produces a DRVD that honors the metric field closely and produces triangulations whose anisotropic elements conform to the input metric field with even very few points for both the RDT and the SRDT (see Figure 8).

Fig. 4. the DRVD of 1020 seeds on the "chair" surface, with a curvature induced metric field. The RDT is traced in black.

Geodesic edges. The geodesic paths are traced by backtracking along the gradient of the geodesic distance map, as described by Yoo et al. [42]. In the case of planar and surface domains, tracing the geodesic paths between the seeds provides the complete triangulation as the Riemannian simplices are simply the interior of the region formed by the three geodesic edges on the surface. Figure 4 shows the DRVD and the RDT of 1020 seeds for the "chair" surface endowed with a curvature-induced anisotropic metric field. The anisotropy of the regions near parabolic lines is well captured with few Riemannian simplices.

Straight edges. By definition, the RVD captures the metric field more accurately than other methods that typically only consider the metric at the vertices. This allows us to construct Riemannian triangulations, but also has a positive influence on the straight edge realization of the DRVD. Figure 7 shows the SRDT of a surface endowed with a curvature induced metric field. The straight edge realization of the DRVD also rapidly become an embedded triangulation during our refinement process. This can for example be seen on the "Fertility" surface where as few as 100 vertices are enough to create an anisotropic triangulation that roughly follows the curvature metric field.

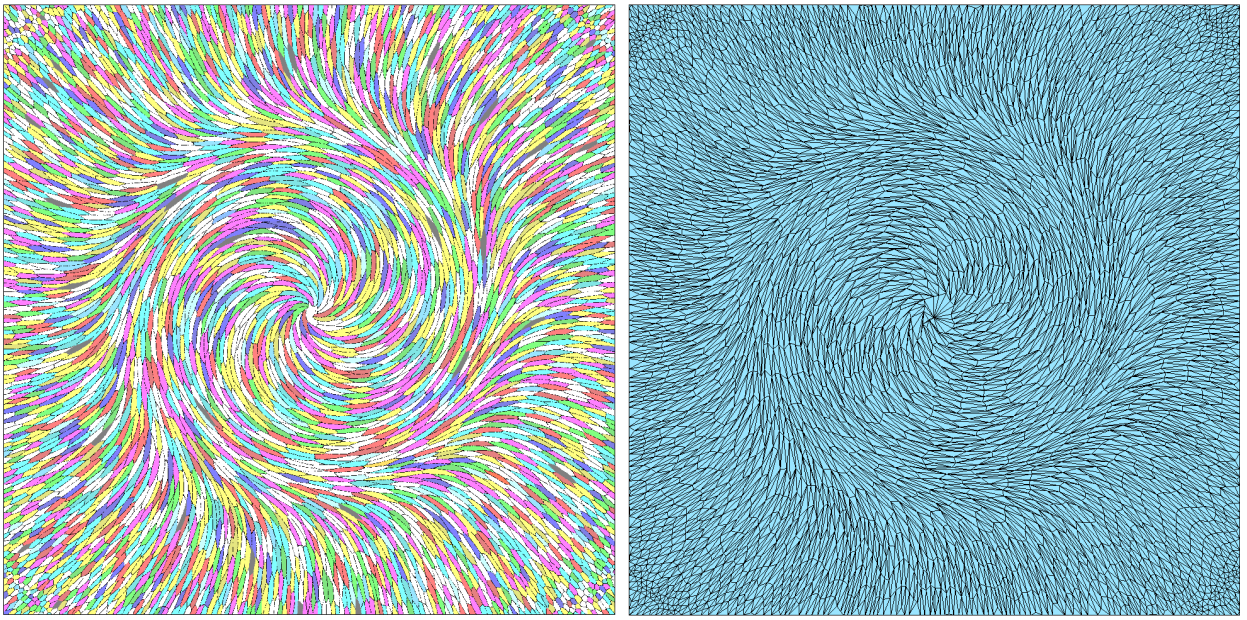

Fig. 5. On the left, the DRVD of 4010 seeds in a square endowed with a swirly metric field. On the right, the straight edge dual triangulation of the DRVD.

Edge lengths and angles. The quality of a mesh and its conforming to the metric field can be evaluated through the measure, in the metric, of the lengths of its edges and of its angles. We evaluate the edge lengths and the angles of the point set shown on Figure 8 (the SRDT is shown) for both the RDT and the SRDT. Note that the point sets are generated with a sizing criterion intended to create unit meshes, that is RDTs whose geodesic edges are ideally of length 1. Results for the edge lengths of the RDT and the angles of the SRDT are shown in Figure 6 and show good control on the edge lengths and the angles. We also observed that while the results are naturally better for the RDT, since the point set is generated with respect to the geodesic distance, the quality of the elements of the SRDT stays relatively close to that of the RDT.

Optimization

Optimization is often used to improve the quality of a triangulation. Centroidal Voronoi tessellations are Voronoi diagrams whose generators are also the centroids (centers of mass) of their respective cells [43]. The famous Lloyd algorithm [44] iteratively moves the seeds to the center of mass of their respective cell and recomputes the Voronoi diagram of this new seed set.

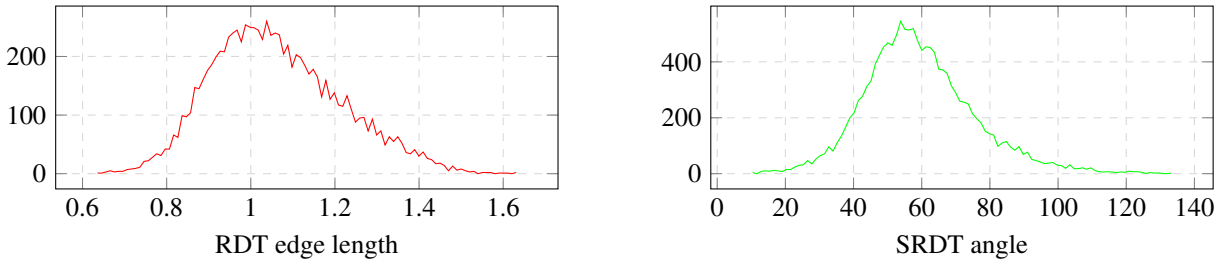

Fig. 6. Histograms for the optimized mesh shown on Figure 7. On the left, RDT edge lengths; on the right, SRDT angles values.

We approximate the Riemannian Voronoi center of mass of a cell V with the following formula:

$$c_g = \frac{\sum_i c_i \, |t_i|_{g_i}}{\sum_i |t_i|_{g_i}}. \tag{2}$$

where $|t_i|_{g_i}$ is the area of the triangle t_i in the metric g_i and the t_i make a partition of V. The canvas conveniently provides this decomposition of a geodesic Voronoi cell in small triangles, making the approximation of c_g accurate. The formula in Equation (2) does not extend to surfaces as the result of the weighted sum might not lie on the domain. In that setting, we use a process similar to Wang et al. [45]. As its Euclidean counterpart, this algorithm comes with no guarantees (not even for termination) but works well in practice. In Figure 8, the initial SRDT of 4000 seeds has been optimized with 100 iterations. The metric field is well captured with few elements, especially in the rotational region.

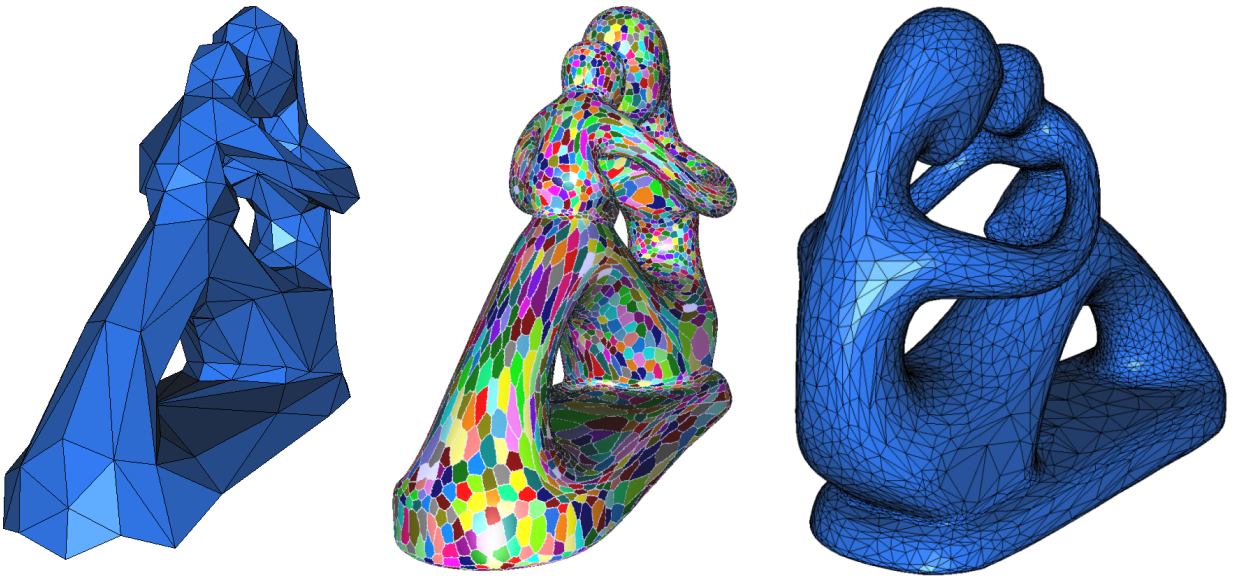

Fig. 7. The “Fertility” surface endowed with a curvature induced metric field, SRDT with 220 seeds (left), DRVD (center) and SRDT (right) with 6020 seeds.

9. Conclusion

We have introduced a practical approach to computing Riemannian Voronoi diagrams and exposed theoretical conditions on the point set and the canvas such that geodesic and straight edge realizations are embedded triangulations

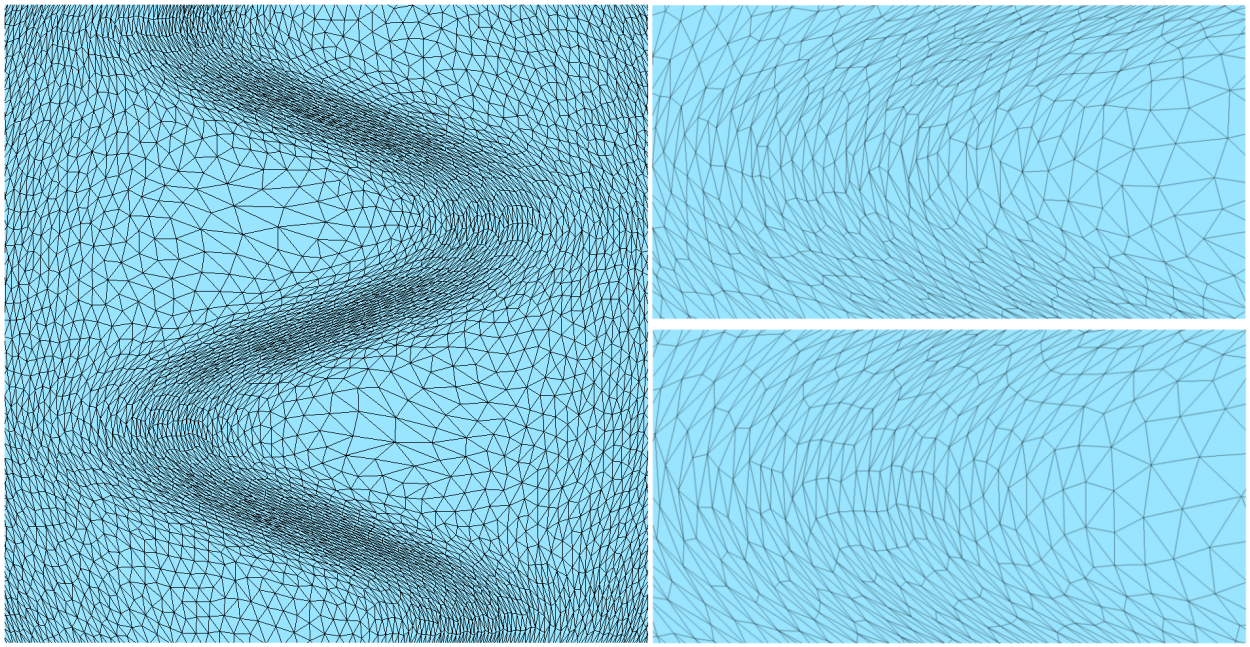

Fig. 8. The optimized SRDT of 4000 seeds in a planar domain endowed with a hyperbolic shock induced metric field (left). On the right, a zoom on a rotational region of the metric field shows the difference between pre- (above) and post- (bottom) optimization.

of the domain in planar and surface cases. The theoretical requirements are demanding, but in practice both the geodesic and straight edge duals require relatively few points to become embedded triangulations, even with non-trivial, highly distorted metric fields. The RVD and its duals are shown to be particularly well suited to capture the metric field in regions where it is both anisotropic and rotational. No preprocessing or smoothing of the metric field is required, a technique that is often used and results in noticeable loss of anisotropy in this type of region.

Future work

The use of the star set greatly increases the speed of the algorithm, and additional techniques are still to be explored. A first technique to improve the computation time is to observe that during most of the computation of the DRVD, there is very little interaction between the different fronts. Parallelization of the algorithm should therefore yield significant gains. If the metric we consider were almost constant in significant parts of our domain, a hybrid diagram that would only use geodesics in zones where the metric varies a lot would decrease the running time significantly.

Another interesting topic to explore would be the applied uses of Riemannian Delaunay triangulations. Indeed, much work has been committed to the theoretical study of these triangulations, but it still stays untouched in practice. Finite elements can be for example defined on the RDT in our context as the canvas naturally offers a piecewise linear transformation between a simplex of the RDT and the traditional reference element of the finite element method. One can also intuit that Riemannian simplices adapted to a metric field could be used to further decrease numerical error.

Finally, the work presented here focused on planar and surface domains. A future paper will provide results for Riemannian manifolds of arbitrary dimension.

References

[1] E. F. D'Azevedo, R. B. Simpson, On optimal interpolation triangle incidences, SIAM J. Sci. Statist. Comput. 10 (1989) 1063–1075.
[2] J. R. Shewchuk, What is a good linear finite element? - interpolation, conditioning, anisotropy, and quality measures, 2002.
[3] J.-M. Mirebeau, Optimal meshes for finite elements of arbitrary order, Constructive approximation 32 (2010) 339–383.
[4] M. Garland, P. S. Heckbert, Surface simplification using quadric error metrics, in: ACM SIGGRAPH, 1997, pp. 209–216.
[5] F. Aurenhammer, R. Klein, Voronoi diagrams, in: J. Sack, G. Urrutia (Eds.), Handbook of Computational Geometry, 2000, pp. 201–290.

[6] Q. Du, D. Wang, Anisotropic centroidal Voronoi tessellations and their applications, SIAM Journal on Scientific Computing 26 (2005).

[7] G.-D. Canas, S.-J. Gortler, Orphan-free anisotropic Voronoi diagrams, Discrete and Computational Geometry 46 (2011).

[8] G. D. Cañas, S. J. Gortler, Duals of orphan-free anisotropic Voronoi diagrams are embedded meshes, in: SoCG, ACM, New York, NY, USA, 2012, pp. 219–228.

[9] S.-W. Cheng, T. K. Dey, E. A. Ramos, R. Wenger, Anisotropic surface meshing, in: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm, Society for Industrial and Applied Mathematics, 2006, pp. 202–211.

[10] F. Labelle, J. R. Shewchuk, Anisotropic Voronoi diagrams and guaranteed-quality anisotropic mesh generation, in: SoCG, 2003, pp. 191–200.

[11] G. Leibon, Random Delaunay triangulations, the Thurston-Andreev theorem, and metric uniformization, Ph.D. thesis, UCSD, 1999.

[12] G. Leibon, D. Letscher, Delaunay triangulations and Voronoi diagrams for Riemannian manifolds, in: SoCG, 2000, pp. 341–349.

[13] J.-D. Boissonnat, R. Dyer, A. Ghosh, Constructing Intrinsic Delaunay Triangulations of Submanifolds, Research Report RR-8273, INRIA, 2013. ArXiv:1303.6493.

[14] R. Dyer, H. Zhang, T. Möller, Surface sampling and the intrinsic Voronoi diagram, Computer Graphics Forum (Special Issue of Symp. Geometry Processing) 27 (2008) 1393–1402.

[15] M. Rouxel-Labbé, M. Wintraecken, J.-D. Boissonnat, Discretized Riemannian Delaunay triangulations, Research Report, INRIA Sophia Antipolis, 2016. URL: `https://www-sop.inria.fr/members/Mathijs.Wintraecken/Papers/IMR_Report.pdf`.

[16] H. Borouchaki, P. L. George, F. Hecht, P. Laug, E. Saltel, Delaunay mesh generation governed by metric specifications. part i. algorithms, Finite elements in analysis and design 25 (1997) 61–83.

[17] J.-D. Boissonnat, C. Wormser, M. Yvinec, Locally uniform anisotropic meshing, in: Proceedings of the twenty-fourth annual symposium on Computational geometry, ACM, 2008, pp. 270–277.

[18] L. Chen, J. Xu, Optimal Delaunay triangulations, Journal of Computational Mathematics 22 (2004) 299–308.

[19] L. Chen, P. Sun, J. Xu, Optimal anisotropic meshes for minimizing interpolation errors in L^p-norm, Mathematics of Computation 76 (2007) 179–204.

[20] P. Alliez, D. Cohen-Steiner, M. Yvinec, M. Desbrun, Variational tetrahedral meshing, ACM Trans. Graph. 24 (2005) 617–625.

[21] J. Tournois, C. Wormser, P. Alliez, M. Desbrun, Interleaving Delaunay refinement and optimization for practical isotropic tetrahedron mesh generation, ACM Trans. Graph. 28 (2009) 75:1–75:9.

[22] Y. Liu, H. Pan, J. Snyder, W. Wang, B. Guo, Computing self-supporting surfaces by regular triangulation, ACM Trans. Graph. 32 (2013).

[23] P. Mullen, P. Memari, F. de Goes, M. Desbrun, Hot: Hodge-optimized triangulations, ACM Trans. Graph. 30 (2011) 103:1–103:12.

[24] J.-D. Boissonnat, D. Cohen-Steiner, M. Yvinec, Comparison of algorithms for anisotropic meshing and adaptive refinement, Research Report ACS-TR-362603, INRIA, 2008.

[25] L. Chen, Mesh smoothing schemes based on optimal Delaunay triangulations, in: Proceedings of the 13th International Meshing Roundtable, Sandia National Laboratories, 2004, pp. 109–120.

[26] X.-M. Fu, Y. Liu, J. Snyder, B. Guo, Anisotropic simplicial meshing using local convex functions, ACM Trans. Graph. 33 (2014) 182:1–182:11.

[27] Z. Zhong, L. Shuai, M. Jin, X. Guo, Anisotropic surface meshing with conformal embedding, Graph. Models 76 (2014) 468–483.

[28] K. Shimada, A. Yamada, T. Itoh, Anisotropic triangulation of parametric surfaces via close packing of ellipsoids, International Journal of Computational Geometry & Applications (200).

[29] Z. Zhong, X. Guo, W. Wang, B. Lévy, F. Sun, Y. Liu, W. Mao, Particle-based anisotropic surface meshing, ACM Trans. Graph. 32 (2013).

[30] G. Peyré, L. Cohen, Geodesic remeshing using front propagation, International Journal on Computer Vision 69 (2006) 145–156.

[31] T.-t. Cao, H. Edelsbrunner, T.-s. Tan, Proof of correctness of the digital Delaunay triangulation algorithm, Computational Geometry: Theory and Applications 48 (2015).

[32] J.-D. Boissonnat, R. Dyer, A. Ghosh, S. Y. Oudot, Only distances are required to reconstruct submanifolds, Research Report, INRIA, 2014.

[33] G. Peyré, M. Péchaud, R. Keriven, L. D. Cohen, Geodesic methods in computer vision and graphics, Found. Trends. Comput. Graph. Vis. 5 (2010) 197–397.

[34] E. Konukoglu, M. Sermesant, O. Clatz, J.-M. Peyrat, H. Delingette, N. Ayache, A recursive anisotropic fast marching approach to reaction diffusion equation: Application to tumor growth modeling, in: Proceedings of the 20th International Conference, 2007, pp. 687–699.

[35] J.-M. Mirebeau, Anisotropic fast-marching on cartesian grids using lattice basis reduction, SIAM Journal on Numerical Analysis 52 (2014).

[36] M. Campen, M. Heistermann, L. Kobbelt, Practical anisotropic geodesy, in: Proceedings of the Eleventh Eurographics/ACMSIGGRAPH Symposium on Geometry Processing, SGP '13, Eurographics Association, 2013, pp. 63–71.

[37] K. Crane, C. Weischedel, M. Wardetzky, Geodesics in heat: A new approach to computing distance based on heat flow, ACM Trans. Graph. 32 (2013) 152:1–152:11.

[38] J.-D. Boissonnat, K.-L. Shi, J. Tournois, M. Yvinec, Anisotropic Delaunay meshes of surfaces, ACM Trans. Graph. 34 (2015) 14.

[39] R. Dyer, H. Zhang, T. Möller, A. Clements, An Investigation of the Spectral Robustness of Mesh Laplacians, Technical Report TR 2007-17, Simon Fraser University, 2007. SFU-CMPT.

[40] J.-D. Boissonnat, C. Wormser, M. Yvinec, Anisotropic Delaunay mesh generation, SIAM Journal on Computing 44 (2015) 467–512.

[41] CGAL, User and Reference Manual, 4.7 ed., CGAL Editorial Board, 2015. URL: `http://doc.cgal.org/4.7/Manual/packages.html`.

[42] S. W. Yoo, J.-K. Seong, M.-H. Sung, S. Y. Shin, E. Cohen, A triangulation-invariant method for anisotropic geodesic map computation on surface meshes, Visualization and Computer Graphics, IEEE Transactions on 18 (2012) 1664–1677.

[43] Y. Liu, W. Wang, B. Lévy, F. Sun, D.-M. Yan, L. Lu, C. Yang, On centroidal voronoi tessellation—energy smoothness and fast computation, ACM Trans. Graph. 28 (2009) 101:1–101:17.

[44] S. Lloyd, Least squares quantization in pcm, IEEE Trans. Inf. Theor. 28 (2006) 129–137.

[45] X. Wang, X. Ying, Y.-J. Liu, S.-Q. Xin, W. Wang, X. Gu, W. Mueller-Wittig, Y. He, Intrinsic computation of centroidal Voronoi tessellation (CVT) on meshes, Computer-Aided Design 58 (2015) 51–61.