

A Convex Optimization Framework for Regularized Geodesic Distances

Michal Edelstein

Technion - Israel Institute of Technology
Haifa, Israel
smichale@cs.technion.ac.il

Nestor Guillen

Texas State University
San Marcos, TX, USA
nestor@txstate.edu

Justin Solomon

Massachusetts Institute of Technology (MIT)
Cambridge, MA, USA
jsolomon@mit.edu

Mirela Ben-Chen

Technion - Israel Institute of Technology
Haifa, Israel
mirela@cs.technion.ac.il

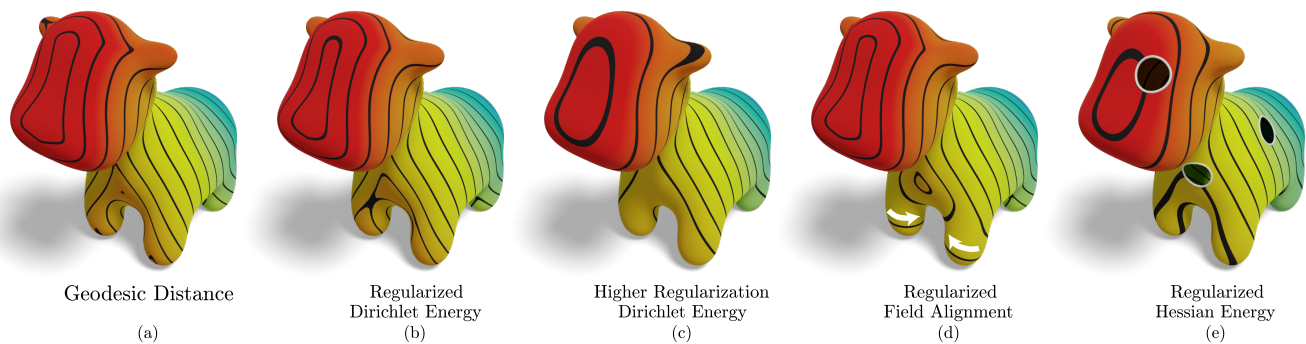


Figure 1: Geodesic distances (a) may not have desired properties such as smoothness. We present a general framework for regularized geodesic distances. Shown here are three examples of regularizers: (b,c) smoothness, (d) alignment to a vector field, and (e) boundary invariance.

ABSTRACT

We propose a general convex optimization problem for computing *regularized* geodesic distances. We show that under mild conditions on the regularizer the problem is well posed. We propose three different regularizers and provide analytical solutions in special cases, as well as corresponding efficient optimization algorithms. Additionally, we show how to generalize the approach to the *all pairs* case by formulating the problem on the product manifold, which leads to *symmetric* distances. Our regularized distances compare favorably to existing methods, in terms of robustness and ease of calibration.

CCS CONCEPTS

• Computing methodologies -> Shape analysis.

KEYWORDS

geodesic distance, convex optimization, triangle meshes

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
SIGGRAPH '23 Conference Proceedings, August 6–10, 2023, Los Angeles, CA, USA
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0159-7/23/08.
<https://doi.org/10.1145/3588432.3591523>

ACM Reference Format:

Michal Edelstein, Nestor Guillen, Justin Solomon, and Mirela Ben-Chen. 2023. A Convex Optimization Framework for Regularized Geodesic Distances. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings (SIGGRAPH '23 Conference Proceedings)*, August 6–10, 2023, Los Angeles, CA, USA. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3588432.3591523>

1 INTRODUCTION

Distance computation is a central task in shape analysis. Distances are required for many downstream geometry processing applications, including shape correspondence, shape descriptors and remeshing. In many cases, however, exact geodesic distances are not required, and a *distance-like* function suffices. Moreover, it is often required to *regularize* the distance-like function to improve performance of a downstream application.

The geometry processing community has proposed myriad methods for computing geodesic distances [Crane et al. 2020], including some regularized distances [Crane et al. 2013; Solomon et al. 2014]. However, a unified framework, including a controlled and easily calibratable approach to regularization is still missing.

We propose a flexible convex optimization framework for computing regularized geodesic distances. We show that under relatively mild conditions, our formulation has a minimizer that converges to the geodesic distance as the regularization weight vanishes. Furthermore, we propose a variety of regularizers, demonstrate their applicability, and provide corresponding efficient optimization algorithms. Finally, we formulate the *all-pairs* problem, as a special case of our framework on the *product manifold*. This formulation has the additional advantage that the resulting regularized distances are *symmetric* with respect to swapping the source and target points.

1.1 Related Work

The work on geodesic distances is vast, and a full review is out of scope. See the recent surveys [Crane et al. 2020; Peyré et al. 2010].

Geodesic Distances. Some approaches (e.g., MMP [Mitchell et al. 1987], MMP extension [Surazhsky et al. 2005], VTP [Qin et al. 2016], and others) compute the exact *polyhedral* geodesic distance on a triangle mesh. Other methods, e.g., Fast Marching [Kimmel and Sethian 1998], take a variational approach and compute approximate geodesic distances. More recently, convex optimization approaches have been suggested for computing approximate geodesic distances [Belyaev and Fayolle 2020, 2015] and for computing the cut locus [Générau et al. 2022a,b]. Our approach is also framed as a convex optimization problem, but incorporates an additional general regularization term.

Regularization. Exact geodesic distances have some shortcomings in applications, e.g., the geodesic distance from a point p is not smooth near the cut locus of p . The heat method [Crane et al. 2013] computes smoothed geodesic distances, where the smoothing is controlled by a time parameter. The earth mover’s distance (EMD) [Solomon et al. 2014] can also be used to compute geodesic distances, optionally smoothed by projecting on a reduced spectral basis. Compared to the heat and EMD methods, our framework allows for a more direct control on the smoothness parameter. For triangle meshes, another option is to compute the graph-based distances on the graph of the triangulation with a Dirichlet or Hessian regularization [Cao et al. 2020]. This approach is, however, triangulation dependent, and requires the use of 2-ring neighborhoods for accurate results. Furthermore, we provide theoretical results that guarantee that our optimization problem is well-posed for general regularizers under some mild conditions, providing mathematical footing required for future work to design additional regularizers.

1.2 Contributions

Our main contributions are:

- A convex optimization problem for extracting regularized geodesic distances, with theoretical results for a general regularizer under some mild conditions.
- Examples of regularizers with corresponding theoretical results and efficient optimization algorithms.
- The *all-pairs* generalization, with a scalable optimization scheme.

2 BACKGROUND

2.1 Geodesic Distances by Convex Optimization

Variational characterizations of the geodesic distance function are natural from several perspectives. From probability, they relate to large deviation estimates for the heat equation, as shown by Varadhan [1967]. From a purely PDE perspective, they can be constructed as the largest viscosity subsolution to the Eikonal equation, using Ishii’s extension of the Perron method to Hamilton-Jacobi equation [Ishii 1987]. Recently, it was shown [Belyaev and Fayolle 2020, 2015] how the geodesic distance u on a domain Ω from a source point x_0 can be computed by solving the convex optimization problem

$$\begin{aligned} \text{Minimize}_u \quad & -\int_{\Omega} u(x) \, d\text{Vol}(x) \\ \text{subject to} \quad & |\nabla u(x)| \leq 1 \text{ for all } x \in \Omega \setminus \{x_0\} \\ & u(x_0) = 0. \end{aligned} \quad (1)$$

As explained by Belyaev et al. [2020], we can thus use convex optimization methods, e.g. ADMM, to approximate geodesic distances.

Intuitively, since the function u is maximized, the gradient norm reaches the maximal allowed value, which is 1. Therefore, while not constraining it directly, the solution will fulfill $|\nabla u| = 1$ at every point in the domain and thus will be a geodesic distance. The big advantage of this formulation, as opposed to directly constraining $|\nabla u| = 1$, is that this optimization problem is *convex*. Furthermore, the point constraint $u(x_0) = 0$ may be relaxed to $u(x_0) \leq 0$ without changing the solutions to the problem. To see why, note that if $\phi : M \rightarrow \mathbb{R}$ is such that $|\nabla \phi(x)| \leq 1$ for all $x \in \Omega \setminus \{x_0\}$ and $\phi(x_0) < 0$, then the function $\tilde{\phi} := \phi - \phi(x_0)$ will satisfy the two constraints in (1) and have a strictly smaller objective functional than ϕ since $\phi < \tilde{\phi}$ everywhere.

3 REGULARIZED GEODESIC DISTANCES

Given a compact surface M and a closed set $E \subset M$ (typically, $E = \{x_0\}$), our goal is to compute a function $u : M \rightarrow \mathbb{R}$ which is “as-geodesic-as-possible,” but has some additional property. Depending on the application, one may require the function to be smooth, or to be aligned to an input direction at some points on the surface. We assume that this additional information is encoded in a *regularizer* functional of the form

$$\mathcal{E}(u) = \int_M F(\nabla u(x), x) \, d\text{Vol}(x), \quad (2)$$

where F is convex in the first argument.

Generalizing Eq.(1), we consider the following convex optimization problem

$$\begin{aligned} \text{Minimize}_u \quad & \alpha \mathcal{E}(u) - \int_M u(x) \, d\text{Vol}(x) \\ \text{subject to} \quad & |\nabla u(x)| \leq 1 \text{ for all } x \in M \setminus E \\ & u(x) \leq 0 \text{ for all } x \in E. \end{aligned} \quad (3)$$

with some $\alpha > 0$. We discuss various options for \mathcal{E} in Sections 3.1, 4 and the supplemental.

The problem (3) has a long history in the case of a domain $\Omega \subset \mathbb{R}^d$ with $F(\nabla u(x), x) = |\nabla u(x)|^2$, which is known as the *elastic-plastic torsion problem*. This is a free boundary problem, i.e., a PDE involving an interface, unknown a priori, across which the PDE’s nature may change dramatically. In our case, this is reflected in two regions for the solution u_α , one where it solves a Poisson equation,

and one where it solves the Eikonal equation. Refer to [Caffarelli and Friedman 1979] and the book [Petrosyan et al. 2012] for more background. For the Riemannian setting, this problem was first studied in [Généreau et al. 2022a], discussed further below.

Even under these general conditions (see the supplemental for detailed assumptions), we show that (a) the optimization problem has a minimizer for every $\alpha > 0$, (b) the minimizer is unique, and (c) they converge uniformly to the exact geodesic distance as $\alpha \rightarrow 0$. We gather these results under the next two theorems.

THEOREM 3.1. *There is a unique minimizer for problem (3).*

THEOREM 3.2. *Let u_α denote the minimizer to the optimization problem (3). Then, as $\alpha \rightarrow 0$*

$$\max_{x \in M} |d(x, E) - u_\alpha(x)| \rightarrow 0,$$

where $d(x, E)$ is the geodesic distance from x to the set E .

The proofs of Theorem 3.1 and Theorem 3.2, are in Supp. 2 and 3.

The unique minimizer u_α provided by Thm. 3.1 is Lipschitz continuous by construction. In addition, it has two distinct regimes in the respective regions $\{|\nabla u_\alpha| = 1\}$ and $\{|\nabla u_\alpha| < 1\}$. For general second order elliptic regularizers, u_α will be smooth in the interior of $\{|\nabla u_\alpha| < 1\}$, there u_α will solve the unconstrained Euler-Lagrange equation corresponding to the objective functional in (3), which would be a nonlinear elliptic equation. Accordingly, standard elliptic theory guarantees that u_α will be smooth in the region where the gradient constraint is not active. In the other region $\{|\nabla u_\alpha| = 1\}$ the function will solve the Eikonal equation in the viscosity sense.

In the case $F = |\nabla u|^2$ (Sec. 3.1) it was proved in [Généreau et al. 2022a] that for all $\alpha < \alpha_0$ (α_0 depending on the geometry of Ω) the minimizer u_α agrees with the geodesic distance function in $\{|\nabla u| = 1\}$. Therefore, u_α coincides with the distance function everywhere save for a region around the cut locus of x_0 . In this region u_α solves the Poisson equation $\Delta u_\alpha = -1/\alpha$. As shown in [Généreau et al. 2022a], as $\alpha \rightarrow 0$, the open set $\{|\nabla u_\alpha| < 1\}$ shrinks and converges to the cut locus. We expect the theorems in [Généreau et al. 2022a] to hold for general elliptic functionals (such as the p -Laplace equation), but this entails pointwise estimates of nonlinear elliptic equations on manifolds beyond the scope of this work.

Our regularizing functionals (Sec. 3.1, 4) correspond to elliptic energy functionals that promote smoothness and other desirable properties (non-negativity, symmetries) in the minimizer of (3). Accordingly, the significance of Thm. 3.2 is in providing a smooth approximation to the geodesic distance function solution. Moreover, this approximation is in the L^∞ metric, so the approximation error can be made small for all $x \in M$ provided α is sufficiently small.

3.1 Dirichlet Regularizer

A natural regularizer (also considered in [Généreau et al. 2022a,b]) is the Dirichlet energy

$$\mathcal{E}_{\text{Dir}}(u) = \frac{1}{2} \int_M |\nabla u(x)|^2 \, d\text{Vol}(x). \quad (4)$$

First, we look at the simple example where the solution to problem (3) using the Dirichlet energy (4) is given by an explicit formula. We analyze the case where M is the circle \mathbb{S}^1 . Fix $\alpha > 0$. We parameterize \mathbb{S}^1 via the map $x \mapsto (\cos(x), \sin(x))$, i.e., by real numbers

modulo 2π . Moreover, we will use the group structure of \mathbb{S}^1 , which is given by $\mathbb{R}/2\pi\mathbb{Z}$. In this case the problem amounts to looking for a 2π -periodic function $u(x)$ that minimizes

$$\frac{\alpha}{2} \int_0^{2\pi} |u'(x)|^2 \, dx - \int_0^{2\pi} u(x) \, dx$$

among all such 2π -periodic functions satisfying the constraints

$$u(0) \leq 0 \text{ and } |u'(x)| \leq 1 \text{ for all } x \in (0, 2\pi).$$

The minimizer $u(x)$ for this problem has a simple analytical expression, given as follows: First, given $x \in \mathbb{R}$ we set $\hat{x} = x \bmod 2\pi$. Then,

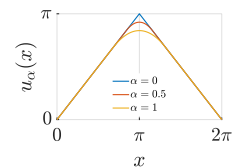
$$u_\alpha(x) = \begin{cases} \hat{x} & \text{if } 0 \leq \hat{x} \leq L \\ \pi - \frac{1}{2}\alpha - \frac{1}{2\alpha}(\hat{x} - \pi)^2 & \text{if } L \leq \hat{x} \leq 2\pi - L \\ 2\pi - \hat{x} & \text{if } \hat{x} \geq L \end{cases} \quad (5)$$

Here, $L = L(\alpha)$ is defined by

$$L(\alpha) = (\pi - \alpha)_+. \quad (6)$$

This expression approximates the distance to the point corresponding to $x = 0$. Observe that for $\alpha > \pi$ the functions u_α are all equal to u_π . In general, the solution u_α has two regimes or regions, one region where it matches the geodesic distance function exactly, and one where it is solving Poisson's equation $u_\alpha'' = -1/\alpha$ and therefore matches a concave parabola, with the condition that u_α is C^1 across these two regions. This is the standard condition for solutions to the obstacle problem (see [Petrosyan et al. 2012]), which is intrinsically related to (3) in this particular case (see Supplemental 1 for further discussion).

The inset figure shows the behavior of the function on the circle. Note the smoothing region, whose width depends on the smoothing parameter α and matches (5)-(6).



Thanks to the group structure of \mathbb{S}^1 and the invariance of the problem under the group action (in other words, by symmetry), we obtain a corresponding formula when the source point is any point $y \in \mathbb{S}^1$. In particular, if $u_\alpha(\cdot, y)$ represents the solution to the problem with source at y , then

$$u_\alpha(x, y) = u_\alpha(x - y), \quad \forall x, y \quad (7)$$

We highlight a notable fact about these functions in a theorem.

THEOREM 3.3. *For every $\alpha > 0$, the function $u_\alpha(x, y)$ given by (5)-(7) defines a metric on \mathbb{S}^1 .*

This theorem will be proved in the supplemental. For a general M , it is not clear whether one can expect $u_\alpha(x, y)$ to be a metric. At the very least, it might be that Theorem 3.3 may generalize to other groups or homogeneous spaces. In Section 6 we discuss an extension of problem (3) to the product manifold $M \times M$ that treats all pairs (x, y) at once, producing an approximation $U_\alpha(x, y)$ that we can prove will be symmetric in (x, y) . We do not prove this general formulation has the triangle inequality but Figure 12 provides some encouragement in that direction.

Another simple example for which we can compute the analytical solution is the disk. We discuss it in the Supplemental, Section 1.

For general triangle meshes, we provide the discrete formulation in Section 5. Figure 5 shows the behavior of our computed distance

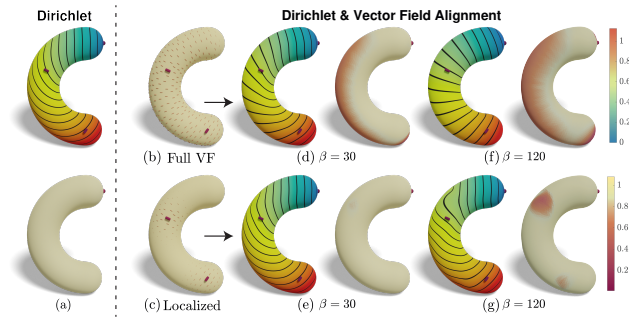


Figure 2: Vector field alignment regularization. (a) Dirichlet regularized distance. The two marked vector directions are not aligned with the regularized distance. (b) An interpolated and (c) localized vector field based on the two directions. (d-g) The corresponding regularized distance, using both Dirichlet and vector field alignment (Sec. 4.1), using two regularization weights.

using the Dirichlet energy as α changes. We plot the normalized error between u_α and u_0 , showing the solution converges smoothly towards u_0 . We also show the result for three α values. For each α , we see both the level sets of the distance function (right), and the norm of the gradient $|\nabla u|$ (left). As α increases the smoothing area around the cut locus becomes larger. Note that far from the cut locus the norm of the gradient is exactly one, showing that our regularized distance is exactly a geodesic distance function there.

4 REGULARIZERS

In Section 3.1 and Supplemental 1 we discussed two analytical examples using the Dirichlet energy (4) as the regularizer. In this section we discuss other possible regularizers in various geometries (not using analytical formulas). The first of those functionals is included in the class (2) covered by our theorems and is motivated by the question of alignment to a given vector field. The other one is a Hessian functional that falls outside the theorems in our work but for which we make several numerical experiments and which raises interesting theoretical questions. Lastly, we discuss one more example of a regularizer in Supplemental 6, with a non-quadratic regularizer that takes advantage of the general form of \mathcal{E} in (3).

4.1 Vector Field Alignment

In addition to smoothing, one might want to align the isolines of the distance function to a given direction. We can align ∇u with the line field $V(x)$ represented as a 3D vector at each point x using the following regularizer:

$$\mathcal{E}(u) = \frac{1}{2} \int_M |\nabla u(x)|^2 + \beta \langle V(x), \nabla u(x) \rangle^2 d\text{Vol}(x), \quad (8)$$

Here, $\langle V(x), \nabla u(x) \rangle$ refers to the Riemmanian metric in M , which amounts to the usual inner product between vectors when M is a surface in \mathbb{R}^3 , for example. Note the additional parameter β that gives the relative weight between alignment to the vector field $V(x)$ versus general smoothness. This is equivalent to computing a distance function using an anisotropic smoothing term, where the

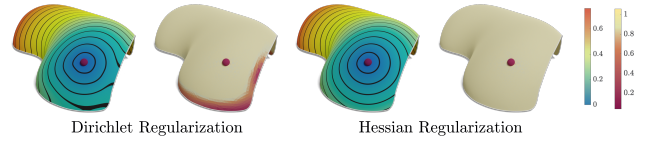


Figure 3: The distance computed using the Dirichlet energy regularizer (left) and the curved Hessian (right). Note the differences near the boundaries.

anisotropic metric at each point on the surface is represented in \mathbb{R}^3 using the following matrix: $I + \beta \hat{V}$ where I is the identity matrix, and $\hat{V} = VV^T$. In terms of the Lagrangian $F(\xi, x)$ in the regularizer functional, this problem corresponds to choosing

$$F(\xi, x) = |\xi|^2 + \beta \langle V(x), \xi \rangle^2 = |A(x)\xi|^2$$

where $A(x) = I + \beta \hat{V}(x)$.

We allow the user to either provide a line field at each point, or specify a sparse set of directions. If needed, we interpolate the sparse constraints to a smooth line field, as suggested by Pluta et al. [2021, Section 5.5.4]. Optionally, we scale the interpolated line field with a geodesic Gaussian (the geodesic distance is computed with our method without regularization).

Figure 2 shows the results. Starting from two vectors (a), we interpolate a line field (b), or a localized line field (c), and compute the resulting vector field aligned regularized distance for two regularization parameters β (d-g). The gradient norm shows where the function deviates from being a geodesic distance as its isolines align to the prescribed directions.

4.2 Hessian for Natural Boundary Conditions

For the Dirichlet regularizer, if we do not impose any boundary conditions on the problem, the minimizer will have zero Neumann boundary values (sometimes called “natural boundary conditions” in FEM). Recently, Stein et al. [2020] suggested using the Hessian energy instead, given by

$$\mathcal{E}(u) = \frac{1}{2} \int_M |\nabla^2 u(x)|^2 d\text{Vol}(x). \quad (9)$$

Here, we use the Frobenius norm of the matrix $\nabla^2 u$ (accordingly, this norm relies on the Riemannian metric of M). This energy yields natural boundary conditions, making the result more robust to holes or mesh boundaries. In the Supplemental, Section 4, we show the analytical solution for the simple case of the circle.

Figure 3 demonstrates this, using the Dirichlet energy (left) and the curved Hessian (right). For each method, the left image shows the level sets of the function, and the right image shows the norm of the gradient $|\nabla u|$. Note the difference near the mesh boundaries, where using the Dirichlet energy leads to zero Neumann conditions, meaning that the isolines are perpendicular to the boundary, and the distance is smoothed, whereas when using the Hessian energy the distance is unaffected by the boundary.

5 OPTIMIZATION VIA ADMM

5.1 Notation

Discretely, we represent surfaces using triangle meshes $M = (\mathcal{V}, \mathcal{F})$, where \mathcal{V} are the vertices, \mathcal{F} are the faces, and $n = |\mathcal{V}|$, $m = |\mathcal{F}|$. We use a piecewise-linear discretization of functions on the mesh with one value per vertex; hence, functions are represented as vectors of length n . Vector fields are piecewise constant per face and can be represented in the trivial basis in \mathbb{R}^3 or in a local basis per face; we represent them using vectors of length \mathbb{R}^{3m} or \mathbb{R}^{2m} , respectively.

Vertex and face areas are denoted by $A_{\mathcal{V}} \in \mathbb{R}^n$, $A_{\mathcal{F}} \in \mathbb{R}^m$, where the area of a vertex is a third of the sum of the areas of its adjacent faces. The diagonal matrices $M_{\mathcal{V}} \in \mathbb{R}^{n \times n}$, $M_{\mathcal{F}} \in \mathbb{R}^{3m \times 3m}$ contain $A_{\mathcal{V}}$, $A_{\mathcal{F}}$ on their corresponding diagonals (repeated 3 times for $A_{\mathcal{F}}$). The total area of the mesh is A . We use standard differential operators [Botsch et al. 2010, Chapter 3]. In particular, our formulation requires the cotangent Laplacian $W_D \in \mathbb{R}^{n \times n}$, the gradient $G \in \mathbb{R}^{3m \times n}$, and the divergence $D = G^T M_{\mathcal{F}} \in \mathbb{R}^{n \times 3m}$.

5.2 Optimization Problem

In this setting, the optimization problem in Eq. (3), becomes:

$$\begin{aligned} \text{Minimize}_u \quad & -A_{\mathcal{V}}^T u + \alpha F_M(Gu) \\ \text{subject to} \quad & |(Gu)_f| \leq 1 \quad \text{for all } f \in \mathcal{F} \\ & u_i \leq 0 \quad \text{for all } i \in E, \end{aligned} \quad (10)$$

where E here is a subset of vertex indices where the distance should be 0, and F_M is a convex function that acts on the gradient of u . Note that this problem is convex whenever F is convex, since the objective will be convex, the first constraint is a second-order cone constraint, and the second constraint is a linear inequality.

5.3 Quadratic Objectives

In practice, the objectives we consider are quadratic, leading to the following optimization problem

$$\begin{aligned} \text{Minimize}_u \quad & -A_{\mathcal{V}}^T u + \frac{\alpha}{2} u^T W u \\ \text{subject to} \quad & |(Gu)_f| \leq 1 \quad \text{for all } f \in \mathcal{F} \\ & u_i \leq 0 \quad \text{for all } i \in E. \end{aligned} \quad (11)$$

Different functionals correspond to different weight matrices W . To use the *Dirichlet energy* in Eq. (4), we set W to the cotangent Laplacian matrix W_D . For the *vector field alignment* objective in Eq. (8) we construct the anisotropic smoothing matrix $W_V = D(I + \beta \hat{V})G$, where $I \in \mathbb{R}^{3m \times 3m}$, and $\hat{V} \in \mathbb{R}^{3m \times 3m}$ is block diagonal, with the 3×3 block of face $f \in \mathcal{F}$ given by $V_f V_f^T$ (see also Section 4.1). Finally, to use the *Hessian regularizer* in Eq. (9) we take W to be the curved hessian matrix [Stein et al. 2020], denoted by W_H .

5.4 Efficient Optimization Algorithm

We derive an alternating direction method of multipliers (ADMM) algorithm [Boyd et al. 2011] to solve the optimization problem in Eq. (11) efficiently. We reformulate the optimization problem, adding an auxiliary variable $z \in \mathbb{R}^{3m}$ representing the gradient of the distance function Gu . This leads to:

$$\begin{aligned} \text{Minimize}_u \quad & -A_{\mathcal{V}}^T u + \frac{\alpha}{2} u^T W u + \sum_{f \in \mathcal{F}} \chi(|z_f| \leq 1) \\ \text{subject to} \quad & (Gu)_f = z_f \quad \text{for all } f \in \mathcal{F} \\ & u_i \leq 0 \quad \text{for all } i \in E, \end{aligned}$$

where $\chi(\cdot)$ is the indicator function, i.e., $\chi(|z_f| \leq 1) = \infty$ if $|z_f| > 1$ and 0 otherwise.

The corresponding augmented Lagrangian is:

$$\begin{aligned} L(u, y, z) \quad & = -A_{\mathcal{V}}^T u + \frac{\alpha}{2} u^T W u + \sum_{f \in \mathcal{F}} \chi(|z_f| \leq 1) + \\ & \sum_{f \in \mathcal{F}} a_f y_f^T ((Gu)_f - z_f) + \frac{\rho \sqrt{A}}{2} \sum_{f \in \mathcal{F}} a_f |(Gu)_f - z_f|^2, \end{aligned}$$

where a_f is the area of the face f , $\rho \in \mathbb{R}$ is the penalty parameter, and $y \in \mathbb{R}^{3m}$ is the dual variable or lagrange multiplier.

The ADMM algorithm consists of iteratively repeating three steps [Boyd et al. 2011, Section 3]. First, we perform u -minimization, then z -minimization, and finally the dual variable, y , is updated. The full derivation of the three steps appears in Supplemental 8, and the resulting algorithm in Algorithm 1.

Algorithm details. Note that the first step, the u -minimization, includes solving a linear system with a fixed coefficient matrix, which is pre-factored and used for all the ADMM iterations, as well as all distance computations. To enforce the constraint $u_E \leq 0$, we eliminate the relevant columns from the linear system and solve for u_i for all $i \in \mathcal{V} \setminus \{E\}$. We project intermediate z values to the unit ball, i.e. $\text{Proj}(z_f \in \mathbb{R}^3, \mathbb{B}^3)$ is equal to $z_f/|z_f|$ if $|z_f| > 1$, and z_f otherwise. We use the stopping criteria suggested by Boyd et al. [2011, Section 3.3.1], formulated for our problem. See Supplemental 8 for details.

Efficiency. We compare our approach with a solution implemented using commercial software, i.e., CVX [Grant and Boyd 2008, 2014] with the MOSEK solver [ApS 2019]. Table 1 provides a comparison of the running times, measured on a desktop machine with an Intel Core i9. For the optimization using CVX and MOSEK, we report both the total running times and the solver running times. Our optimization scheme yields at least an order of magnitude improvement.

Figure 4 shows our result for multiple sources: three isolated points, a vertex sampling of a path, and the boundary. Additional results are shown in Section 7 and in the Supplemental.

ALGORITHM 1: ADMM.

```

input :  $M, \alpha, W, E$ 
output :  $u \in \mathbb{R}^n$  - distance to  $E$ 
initialize  $\rho \in \mathbb{R}$ ; // penalty parameter
            $z \leftarrow \mathbf{0}^{3m}$ ; // auxiliary variable,  $Gu = z$ 
            $y \leftarrow \mathbf{0}^{3m}$ ; // dual variable
            $\rho \leftarrow \rho \sqrt{A}$ 
while algorithm did not converge do // See Supp. 8
    solve  $(\alpha W + \rho W_D)u = A_{\mathcal{V}} - Dy + \rho Dz$  s.t.  $u_E = 0$ 
     $z_f \leftarrow \text{Proj}(\frac{1}{\rho} y_f + (Gu)_f, \mathbb{B}^3)$  for all  $f \in \mathcal{F}$ 
     $y \leftarrow y + \rho(Gu - z)$ 
end

```

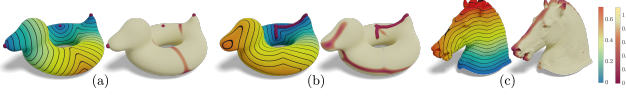


Figure 4: Distance to multiple sources: (a) 3 points, (b) a vertex sampling of a path, and (c) the boundary. We show the distance and the gradient norm.

Table 1: Running times for computing the distance from a single source.

Model	\mathcal{F}	ADMM (sec)	CVX (sec)	
			Total	MOSEK
Pipe, Fig 5	10K	0.075	1.16	0.36
Moai, Fig 8	43K	1.01	5.93	2.25
Armadillo, Supp. 10	346K	1.89	37.3733	21.88
Gardet, Supp. 10	989K	5.22	132.91	87.83
Dragon, Supp. 10	2349K	7.66	347.73	230.02
Sea star, Supp. 10	3500K	11.16	572.56	380.36

6 SYMMETRIC ALL-PAIRS FORMULATION

In Section 3.1 we observed how the problem (3) produces a new metric on \mathbb{S}^1 . It is not clear if problem (3) produces such a result in general. Part of the problem is how (3) treats the x and the source y (if $E = \{y\}$) differently. If one is interested in approximating the full geodesic distance function $d(x, y)$ it is of interest to have a formulation that works directly in all of $M \times M$ and which is naturally symmetric. This leads to the following variation on problem (3).

Consider the manifold $M \times M$ with the product metric inherited from M . Naturally, given a function $U : M \times M \rightarrow \mathbb{R}$ we can fix $y \in M$ and consider the function $x \mapsto U(x, y)$ or fix $x \in M$ and consider the function $y \mapsto U(x, y)$. We define $\nabla_1 U(x, y)$ and $\nabla_2 U(x, y)$ to be the respective gradients for these functions in M . Equivalently, from the decomposition $T(M \times M)_{(x, y)} = (TM)_x \oplus (TM)_y$ we see that $\nabla_{M \times M} U(x, y) = (\nabla_1 U, \nabla_2 U)$.

With this notation, consider the minimization problem

$$\begin{aligned} & \text{Minimize}_U \quad \alpha \mathcal{E}_{M \times M}(U) - \int_{M \times M} U(x, y) \, d\text{Vol}(x, y) \\ & \text{subject to} \quad |\nabla_1 U(x, y)| \leq 1 \text{ in } \{(x, y) \mid x \neq y\} \\ & \quad \quad \quad |\nabla_2 U(x, y)| \leq 1 \text{ in } \{(x, y) \mid x \neq y\} \\ & \quad \quad \quad U(x, y) \leq 0 \text{ on } \{(x, y) \mid x = y\} \end{aligned} \quad (12)$$

Here, we are focusing on the Dirichlet energy functional

$$\mathcal{E}_{M \times M}(U) := \frac{1}{2} \int_{M \times M} |\nabla_1 U(x, y)|^2 + |\nabla_2 U(x, y)|^2 \, d\text{Vol}(x, y)$$

The optimization problem (12) is a natural extension of (3) if one is interested in the full geodesic distance function. Indeed, for $\alpha = 0$, problem (12) has only one minimizer, the geodesic distance function.

6.1 Theoretical Results

Our discussion suggests that the solutions to (12), to the extent they exist, should converge to the geodesic distance as $\alpha \rightarrow 0$. The next two theorems, counterparts to Theorems 3.1 and 3.2, address this.

THEOREM 6.1. *There is a unique minimizer for problem (12).*

(See Supplemental 5 for a proof.)

We will denote the unique minimizer for this problem by $U_\alpha(x, y)$. The motivation for (12) was in part finding a way to guarantee the

symmetry of the resulting regularization, and so we have the following theorems, proved in Supplemental 5.

THEOREM 6.2. *The function $U_\alpha(x, y)$ is symmetric in x and y .*

THEOREM 6.3. *As $\alpha \rightarrow 0$, we have*

$$\|d(x, y) - U_\alpha(x, y)\|_{L^\infty(M \times M)} \rightarrow 0.$$

Analogously to Theorem 3.2, this last theorem guarantees the functions U_α provide a uniform approximation to the full geodesic distance $d(x, y)$ provided α is chosen adequately.

6.2 Scalable Optimization

Discretely, we represent U as an $n \times n$ matrix. We also express $\nabla_1 U(x, y), \nabla_2 U(x, y)$ as gradients over the row and columns of U , i.e., GU and GU^T . The optimization problem in Eq. (12) becomes:

$$\begin{aligned} & \text{Minimize}_U \quad -A_V^T U A_V + \\ & \quad \quad \quad \frac{1}{2} \alpha \text{Tr} \left(M_V (U^T W_D U + U W_D U^T) \right) \\ & \text{subject to} \quad |(\nabla U_{(i, \cdot)})_f| \leq 1 \quad \text{for all } f \in \mathcal{F}, i \in \mathcal{V} \\ & \quad \quad \quad |(\nabla U_{(\cdot, j)})_f| \leq 1 \quad \text{for all } f \in \mathcal{F}, j \in \mathcal{V} \\ & \quad \quad \quad U_{i,i} \leq 0 \quad \quad \quad \text{for all } i \in \mathcal{V}, \end{aligned} \quad (13)$$

where $X_{i,j}$ denotes the (i, j) -th element of a matrix X , $X_{(i, \cdot)}$ denotes the i -th row, and $X_{(\cdot, j)}$ the j -th column.

The complexity here is significantly higher than computing the distance of all points to a closed set. A naive formulation of the ADMM for this problem leads to a per-iteration linear solve with a system matrix of size $n^2 \times n^2$. To reduce it to n solves with a system matrix of size $n \times n$, we derive a second ADMM algorithm, Alg. 2 (see Supp), scalable to larger meshes. The symmetric formulation in Equation (12) arises naturally in this derivation.

Figure 6 shows an example of distances computed using this approach. The fixed source formulation (Alg. 1) (left) is not symmetric. We can symmetrize the distance matrix (center), but this leads to

ALGORITHM 2: Symmetric All-Pairs ADMM.

```

input :  $M, \alpha$ 
output:  $U \in \mathbb{R}^{n \times n}$ ; // dual consensus variable
initialize  $\rho_1, \rho_2 \in \mathbb{R}$ ; // penalty parameters
 $Z, Q \leftarrow \mathbf{0}^{3m \times n}$ ; // auxiliary variables  $GX = Z, GR = Q$ 
 $Y, S \leftarrow \mathbf{0}^{3m \times n}$ ; // dual variables
 $H, K \leftarrow \mathbf{0}^{n \times n}$ ; // dual consensus variables
 $\rho_1 \leftarrow \rho_1 \sqrt{A}$ ,  $\rho_2 \leftarrow \rho_2 \sqrt{A^{-1}}$ 
 $W_P \leftarrow (\alpha + \rho_1) W_D + \rho_2 M_V$ ,  $M_P \leftarrow \frac{1}{2} A_V A_V^T M_V^{-1}$ 
while algorithm did not converge do // See Supp. 9
  solve for  $X$ 
   $W_P X = M_P - DY + \rho_1 DZ - M_V H + \rho_2 M_V U$ 
  solve for  $R$ 
   $W_P R = M_P - DS + \rho_1 DQ - M_V K + \rho_2 M_V U^T$ 
   $(Z_{(\cdot, i)})_f \leftarrow$ 
  Proj  $\left( \frac{1}{\rho_1} (Y_{(\cdot, i)})_f + (GX_{(\cdot, i)})_f, \mathbb{B}^3 \right)$  for all  $i \in \mathcal{V}, f \in \mathcal{F}$ 
   $(Q_{(\cdot, i)})_f \leftarrow$ 
  Proj  $\left( \frac{1}{\rho_1} (S_{(\cdot, i)})_f + (GR_{(\cdot, i)})_f, \mathbb{B}^3 \right)$  for all  $i \in \mathcal{V}, f \in \mathcal{F}$ 
   $U = \max \left( \frac{H+K^T}{2\rho_2} + \frac{X+R^T}{2}, 0 \right)$ ;  $U_{i,i} = 0$  for all  $i \in \mathcal{V}$ 
   $Y \leftarrow Y + \rho_1 (GX - Z)$ ;  $S \leftarrow S + \rho_1 (GR - Q)$ 
   $H \leftarrow H + \rho_2 (X - U)$ ;  $K \leftarrow K + \rho_2 (R - U^T)$ 
end

```

visible noise in the gradient norm. The all-pairs formulation (Alg. 2) is both symmetric and has a smooth gradient norm.

7 EXPERIMENTAL RESULTS

7.1 Scale-Invariant Parameters

The parameter α controls the size of the smoothing area. Therefore, scaling the mesh requires changing its value. To avoid that, and enable more intuitive control of the smoothing area, we define a scale-invariant smoothing parameter $\hat{\alpha}$ that is independent of the mesh area or resolution. For the Dirichlet and vector field alignment energies, we achieve that by setting $\alpha = \hat{\alpha}\sqrt{A}$. For the Hessian energy, we set $\alpha = \hat{\alpha}\sqrt{A^3}$. We note that the parameter β is already scale-invariant, i.e., $\beta = \hat{\beta}$. For our ADMM algorithms (Sec. 5, 6.2) to be scale-invariant, we normalize the penalty variables, residual and feasibility tolerances. Figure 7 demonstrates this. We uniformly rescale an input mesh, and use the same smoothing parameter $\hat{\alpha}$. Note that while the distances are different between the meshes, the scale of the smoothed region, i.e., the area where the norm of the gradient is not 1, is similar. For all our experiments we use the scale invariant formulation, unless stated otherwise.

7.2 Comparison

In Fig. 8 we compare our Dirichlet regularized distances to “Geodesics in Heat” [Crane et al. 2013] and regularized EMD [Solomon et al. 2014], with two smoothing parameters for each. In addition, we show the exact geodesics computed using MMP [Mitchell et al. 1987] for reference. Note that while all approaches lead to a smoother solution compared to the exact geodesics, our approach is more stable, in the sense that the same scale of regularization is observed on all meshes, for the same parameters. Thus, we conjecture that for our approach the regularization parameter is easier to tune.

Table 2 compares the running times, and the maximal error w.r.t the MMP distance (as a % of the maximal distance). The distances are computed with Geometry Central [Sharp et al. 2019] for the heat method and MMP, and with a Matlab implementation of our ADMM Algorithm 1. Note that both our method and the heat method have comparable errors, and for both smoother solutions have larger errors. A timing comparison for the all-pairs case is in the Supp.

We additionally show in the supplemental material a comparison of the representation error of the different approaches in a reduced basis (providing a quantitative measure of smoothness).

7.3 Robustness

Meshing. We demonstrate that our method is invariant to meshing, and is applicable to non-uniform meshing *without* modifying α . Fig. 9 compares our result with the heat method, for 3 remeshings of the same shape. Note that for the heat method with the default smoothing parameter (left), the half-half mesh fails. This is remedied by using a different parameter (center), however there are still differences between the different meshing (note especially the gradient norm). Using our approach (right) we get very similar distance functions and gradient norm for all 3 meshings. Fig. 5 in the supplemental shows additional results with bad triangulations.

Noise. Fig. 10 shows robustness to noise and bad meshing. We add Gaussian normal noise with $\sigma = 0.5, 0.8$ of the mean edge

length, and use a remesh with highly anisotropic triangles and self-intersections. We show the distances and the gradient norm, all with the same α . Note that the results are consistent between the different meshes.

Symmetry error. Our Alg. 1 is not symmetric. Figure 11 shows the symmetry error $\frac{1}{\sqrt{A}}|d(x, y) - d(y, x)|$ for 3 source points for our method and the heat method. Note that for all three points, the symmetry error is higher for the heat method.

Triangle inequality error. Our method does not guarantee that triangle inequality holds, while EMD does. However, experimentally it does hold for higher values of $\hat{\alpha}$. Fig. 12 shows the triangle inequality error of a fixed pair of vertices with respect to every other vertex. We compare the heat method, Alg. 1, and Alg. 2. For the first two, we symmetrize the computed distance matrix. We show the results for three t values for the heat method, and three values of $\hat{\alpha}$ for our approach. We visualize the distance from the chosen point using isolines. Note the difference in the error scaling between the two methods. Further, note that for higher values of $\hat{\alpha}$ our approach has no violations of the triangle inequality. Table 3 shows the percentage of triplets violating the triangle inequality for the same data. Note, that also when considering all the possible triplets, higher values of the smoothing parameter lead to less violations.

7.4 Volumetric Distances

Our framework can compute distances on *tetrahedral meshes*. We replace the standard mass matrix, gradient, and divergence operators with their volumetric versions, as implemented in *gptoolbox* [Jacobson et al. 2021]. Figure 13 demonstrates our Dirichlet regularized volumetric distance on a human shape (a). We show the distance from a point on the shoulder on two planar cuts (b,c), and the distance from the boundary using two $\hat{\alpha}$ values (d,e).

7.5 Example Application: Distance Function for Knitting

Some approaches for generating knitting instructions for 3D models require a function whose isolines represent the knitting rows [Edelstein et al. 2022; Narayanan et al. 2018]. Using the geodesic distance to an initial point (or a set of points) is a good choice since the stitch heights are constant, as are the distances between isolines. On the other hand, this choice limits the design freedom significantly, as designers and knitters have no control over the knitting direction on different areas of the shape. Using regularized distances with vector alignment solves this problem. For example, see Figure 2 and the C model. Using geodesic distances to the starting point will result in a non-symmetric shape (a) (see also [Edelstein et al. 2022, Figure 10]). By adding 2 directional constraints (f), we obtain a function whose isolines respect the symmetries of the shape. Note that for the regularized distances, the gradient norm is no longer 1 everywhere, and thus the distances between isolines is not constant. This can be addressed when knitting by using stitches of different heights. Figure 14 shows how adding alignment to the teddy’s arm and legs aligns the knitting rows with the creases. Crease alignment leads to better shaping [Edelstein et al. 2022, Section 9.3], and prevents over-smoothing of the knit model.

Table 2: Comparison of run-times (T) and the maximal error (ϵ) of the computed distance (in % of the maximal distance) for the models in Figure 8.

Model	\mathcal{F}	MMP T (sec)	Heat $t = \hat{\epsilon}^2$		Heat $t = 20\hat{\epsilon}^2$		EMD d_W^0		EMD d_W^{100}		Ours $\hat{\alpha} = 0.02$		Ours $\hat{\alpha} = 0.1$	
			T (sec)	ϵ (%)	T (sec)	ϵ (%)	ϵ (%)	ϵ (%)	T (sec)	ϵ (%)	T (sec)	ϵ (%)		
Homer	23K	0.255	0.031	2.47	0.030	7.76	30.33	30.31	0.3664	3.40	0.1503	11.90		
Elephant	10K	0.061	0.011	4.84	0.011	11.36	11.36	15.82	0.191	2.00	0.138	5.72		
Armadillo	29K	0.279	0.041	2.83	0.041	9.33	20.55	14.82	0.454	2.35	0.192	6.78		
Moai	43K	1.133	0.101	1.71	0.102	5.34	26.32	26.43	0.683	1.84	0.390	8.60		
Koala	9K	0.063	0.013	2.30	0.010	7.17	55.26	41.15	0.124	2.35	0.059	8.64		

Table 3: Percentage of triplets violating triangle inequality for the data in Fig. 12. Note that for our approach higher values of α lead to less violations.

	Heat - Symmetrized	Fixed-Source - Symmetrized	All-Pairs
(a)	1.84	2.04	1.23
(b)	2.20	1.28	0.88
(c)	2.20	0.25	0.09

8 CONCLUSIONS AND FUTURE WORK

We presented a novel framework for constructing regularized geodesic distances on triangle meshes. We demonstrated the versatility of our approach by presenting three regularizers, analyzing them, and providing an efficient optimization scheme, as well as a symmetric formulation on the product manifold. The theoretical results and experiments in this work raise a number of interesting questions for future research. One of them is whether the functions $U_\alpha(x, y)$ provide metrics in general, i.e., whether they satisfy the triangle inequality; we are not aware of results where geodesic distances can be regularized to have smooth metrics in $M \times M$. Another theoretical question involves convergence of the minimizers in the Hessian energy-regularized problem, as discussed in Section 4. Algorithmically, the ADMM algorithm from Section 5.4 easily generalizes to other convex functions F_M (e.g., L^1 norms) in Equation 10; recent theory on nonconvex ADMM also suggests that Algorithm 1 can be effective for nonconvex regularizers, possibly requiring large augmentation weights ρ [Attouch et al. 2010; Gao et al. 2020; Hong et al. 2016; Ouyang et al. 2020; Stein et al. 2022; Wang et al. 2019; Zhang et al. 2019; Zhang and Shen 2019].

ACKNOWLEDGMENTS

Michal Edelstein acknowledges funding from the Jacobs Qualcomm Excellence Scholarship and the Zeff, Fine and Daniel Scholarship. Nestor Guillen was supported by the National Science Foundation through grant DMS-2144232. The MIT Geometric Data Processing group acknowledges the generous support of Army Research Office grants W911NF2010168 and W911NF2110293, of Air Force Office of Scientific Research award FA9550-19-1-031, of National Science Foundation grants IIS-1838071 and CHS-1955697, from the CSAIL Systems that Learn program, from the MIT-IBM Watson AI Laboratory, from the Toyota-CSAIL Joint Research Center, from a gift from Adobe Systems, and from a Google Research Scholar award. Mirela Ben-Chen acknowledges the support of the Israel Science Foundation (grant No. 1073/21), and the European Research Council (ERC starting grant no. 714776 OPREP). We use the repositories SHREC'07, SHREC'11, Windows 3D library, AIM@SHAPE, and Three D Scans, and thank Keenan Crane, Jan Knippers, Daniel Sonntag, and Yu Wang for providing additional models. We also

thank Hsueh-Ti Derek Liu for his Blender Toolbox, used for the visualizations throughout the paper.

REFERENCES

- MOSEK ApS. 2019. *The MOSEK optimization toolbox for MATLAB manual. Version 9.0*. <http://docs.mosek.com/9.0/toolbox/index.html>
- Hédy Attouch, Jérôme Bolte, Patrick Redont, and Antoine Soubeyran. 2010. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Lojasiewicz inequality. *Mathematics of operations research* 35, 2 (2010), 438–457.
- Alexander Belyaev and Pierre-Alain Fayolle. 2020. An ADMM-based scheme for distance function approximation. *Numerical Algorithms* 84, 3 (2020), 983–996.
- Alexander G Belyaev and Pierre-Alain Fayolle. 2015. On variational and PDE-based distance function approximations. In *Computer Graphics Forum*, Vol. 34. Wiley Online Library, 104–118.
- Iwan Bokseveld and Amir Vaxman. 2022. High-Order Directional Fields. *ACM Transactions on Graphics (TOG)* 41, 6 (2022), 1–17.
- Mario Botsch, Leif Kobbelt, Mark Pauly, Pierre Alliez, and Bruno Lévy. 2010. *Polygon mesh processing*. CRC press.
- Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. 2011. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning* 3, 1 (2011), 1–122.
- Luis A Caffarelli and Avner Friedman. 1979. The free boundary for elastic-plastic torsion problems. *Trans. Amer. Math. Soc.* 252 (1979), 65–97.
- Luming Cao, Junhao Zhao, Jian Xu, Shuangmin Chen, Guozhu Liu, Shiqing Xin, Yuanfeng Zhou, and Ying He. 2020. Computing smooth quasi-geodesic distance field (QGDF) with quadratic programming. *Computer-Aided Design* 127 (2020), 102879.
- Keenan Crane, Marco Livesu, Enrico Puppo, and Yipeng Qin. 2020. A survey of algorithms for geodesic paths and distances. *arXiv preprint arXiv:2007.10430* (2020).
- Keenan Crane, Clarisse Weischedel, and Max Wardetzky. 2013. Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Transactions on Graphics (TOG)* 32, 5 (2013), 1–11.
- Michal Edelstein, Hila Peleg, Shachar Itzhaky, and Mirela Ben-Chen. 2022. AmiGo: Computational Design of Amigurumi Crochet Patterns. In *Proceedings of the 7th Annual ACM Symposium on Computational Fabrication*. 1–11.
- Wenbo Gao, Donald Goldfarb, and Frank E Curtis. 2020. ADMM for multiaffine constrained optimization. *Optimization Methods and Software* 35, 2 (2020), 257–303.
- François G enerau, Edouard Oudet, and Bozhidar Velichkov. 2022a. Cut locus on compact manifolds and uniform semiconcavity estimates for a variational inequality. *Archive for Rational Mechanics and Analysis* 246, 2 (2022), 561–602.
- François G enerau, Edouard Oudet, and Bozhidar Velichkov. 2022b. Numerical computation of the cut locus via a variational approximation of the distance function. *ESAIM: Mathematical Modelling and Numerical Analysis* 56, 1 (2022), 105–120.
- Michael Grant and Stephen Boyd. 2008. Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control*, V. Blondel, S. Boyd, and H. Kimura (Eds.). Springer-Verlag Limited, 95–110. http://stanford.edu/~boyd/graph_dcp.html.
- Michael Grant and Stephen Boyd. 2014. CVX: Matlab Software for Disciplined Convex Programming, version 2.1. <http://cvxr.com/cvx>.
- Mingyi Hong, Zhi-Quan Luo, and Meisam Razaviyayn. 2016. Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems. *SIAM Journal on Optimization* 26, 1 (2016), 337–364.
- Hitoshi Ishii. 1987. Perron’s method for Hamilton-Jacobi equations. *Duke Mathematical Journal* 55, 2 (1987), 369–384.
- Alec Jacobson et al. 2021. gptoolbox: Geometry Processing Toolbox. <http://github.com/alecjacobson/gptoolbox>.
- Ron Kimmel and James A Sethian. 1998. Computing geodesic paths on manifolds. *Proceedings of the national academy of sciences* 95, 15 (1998), 8431–8435.
- Joseph SB Mitchell, David M Mount, and Christos H Papadimitriou. 1987. The discrete geodesic problem. *SIAM J. Comput.* 16, 4 (1987), 647–668.
- Vidya Narayanan, Lea Albaugh, Jessica Hodgins, Stelian Coros, and James Mccann. 2018. Automatic machine knitting of 3D meshes. *ACM Transactions on Graphics (TOG)* 37, 3 (2018), 1–15.

- Wenqing Ouyang, Yue Peng, Yuxin Yao, Juyong Zhang, and Bailin Deng. 2020. Anderson acceleration for nonconvex ADMM based on Douglas-Rachford splitting. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 221–239.
- Arshak Petrosyan, Henrik Shahgholian, and Nina Nikolaevna Uraltseva. 2012. *Regularity of free boundaries in obstacle-type problems*. Vol. 136. American Mathematical Soc.
- Gabriel Peyré, Mickael Péchaud, Renaud Keriven, Laurent D Cohen, et al. 2010. Geodesic methods in computer vision and graphics. *Foundations and Trends® in Computer Graphics and Vision* 5, 3–4 (2010), 197–397.
- Kacper Pluta, Michal Edelstein, Amir Vaxman, and Mirela Ben-Chen. 2021. PH-CPF: planar hexagonal meshing using coordinate power fields. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–19.
- Yipeng Qin, Xiaoguang Han, Hongchuan Yu, Yizhou Yu, and Jianjun Zhang. 2016. Fast and exact discrete geodesic computation based on triangle-oriented wavefront propagation. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–13.
- Nicholas Sharp, Keenan Crane, et al. 2019. GeometryCentral: A modern C++ library of data structures and algorithms for geometry processing. <https://geometry-central.net/>. (2019).
- Justin Solomon, Raif Rustamov, Leonidas Guibas, and Adrian Butscher. 2014. Earth mover’s distances on discrete surfaces. *ACM Transactions on Graphics (ToG)* 33, 4 (2014), 1–12.
- Oded Stein, Alec Jacobson, Max Wardetzky, and Eitan Grinspun. 2020. A smoothness energy without boundary distortion for curved surfaces. *ACM Transactions on Graphics (TOG)* 39, 3 (2020), 1–17.
- Oded Stein, Jiajin Li, and Justin Solomon. 2022. A splitting scheme for flip-free distortion energies. *SIAM Journal on Imaging Sciences* 15, 2 (2022), 925–959.
- Vitaly Surazhsky, Tatiana Surazhsky, Danil Kirsanov, Steven J Gortler, and Hugues Hoppe. 2005. Fast exact and approximate geodesics on meshes. *ACM transactions on graphics (TOG)* 24, 3 (2005), 553–560.
- Sathamangalam R Srinivasa Varadhan. 1967. On the behavior of the fundamental solution of the heat equation with variable coefficients. *Communications on Pure and Applied Mathematics* 20, 2 (1967), 431–455.
- Yu Wang, Wotao Yin, and Jinshan Zeng. 2019. Global convergence of ADMM in nonconvex nonsmooth optimization. *Journal of Scientific Computing* 78 (2019), 29–63.
- Juyong Zhang, Yue Peng, Wenqing Ouyang, and Bailin Deng. 2019. Accelerating ADMM for efficient simulation and optimization. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–21.
- Tao Zhang and Zhengwei Shen. 2019. A fundamental proof of convergence of alternating direction method of multipliers for weakly convex optimization. *Journal of Inequalities and Applications* 2019, 1 (2019), 1–21.

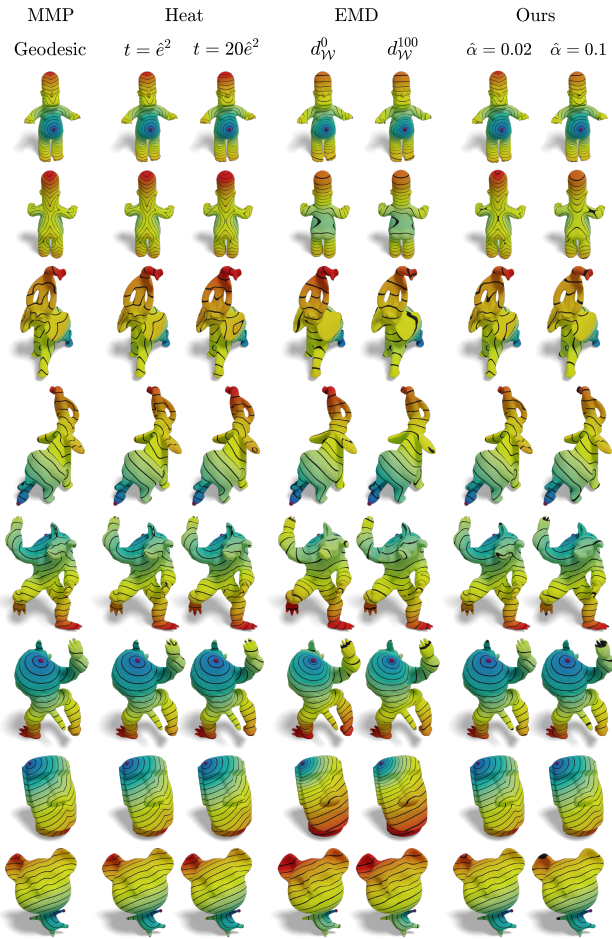


Figure 8: A qualitative comparison between our Dirichlet regularized distances, Heat method, and EMD, with two choices of smoothing parameter per method. See the text for details.

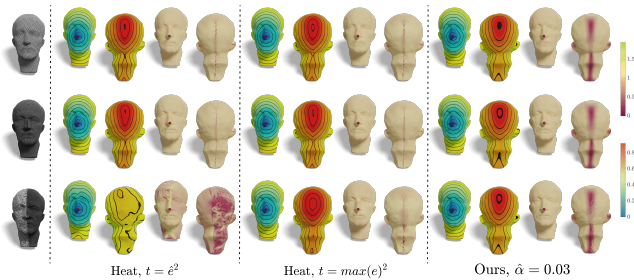


Figure 9: Our results v.s. the heat method for different remeshings, we show the distance function and the gradient norm. Note that our approach leads to distances which are very similar for the three meshes, *with the same smoothing parameter $\hat{\alpha}$* . See the text for details.

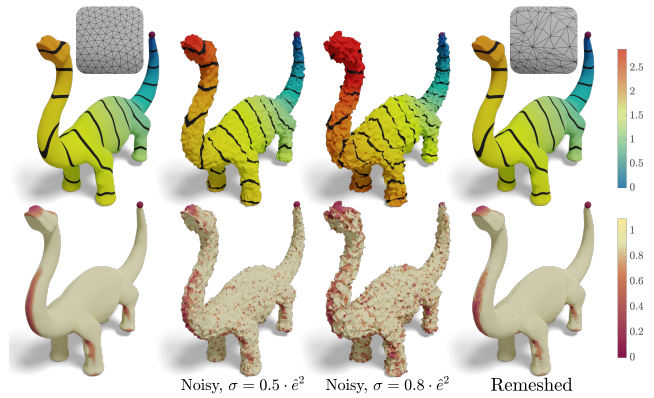


Figure 10: Robustness to noise and bad meshing, all distances computed with the same $\hat{\alpha}$. Note the similarities of the distances and gradient norm, despite the large normal noise and badly shaped triangles.

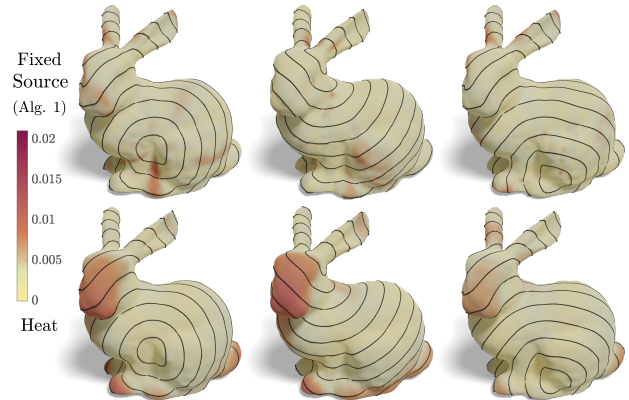


Figure 11: Violation of the symmetric property for 3 source points. Note that while our method is not symmetric by construction, the symmetry error is lower than the symmetry error for the heat method.

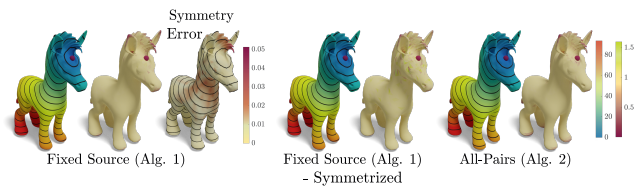


Figure 6: The all-pairs formulation, Alg. 2 (right), vs the fixed source formulation, Alg. 1 (left), and its symmetrized version (center). See the text for details.

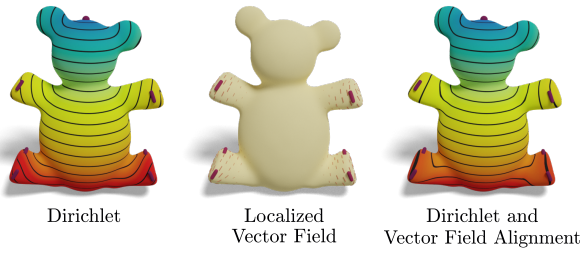


Figure 14: Vector field alignment of creases, useful for knitting applications.

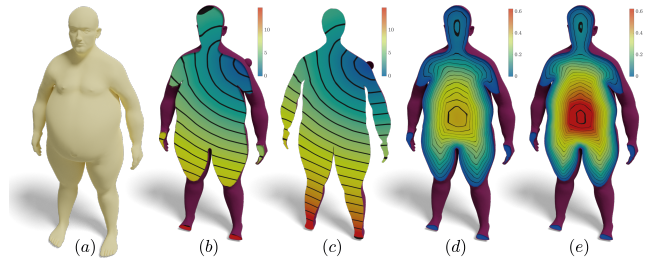


Figure 13: Dirichlet regularized volumetric distances. (a) The input tetrahedral mesh. (b,c) Two cuts showing the distance to a point on the shoulder. (d,e) Distance to the boundary, where (d) is more smoothed than (e), i.e. has a larger $\hat{\alpha}$ value.

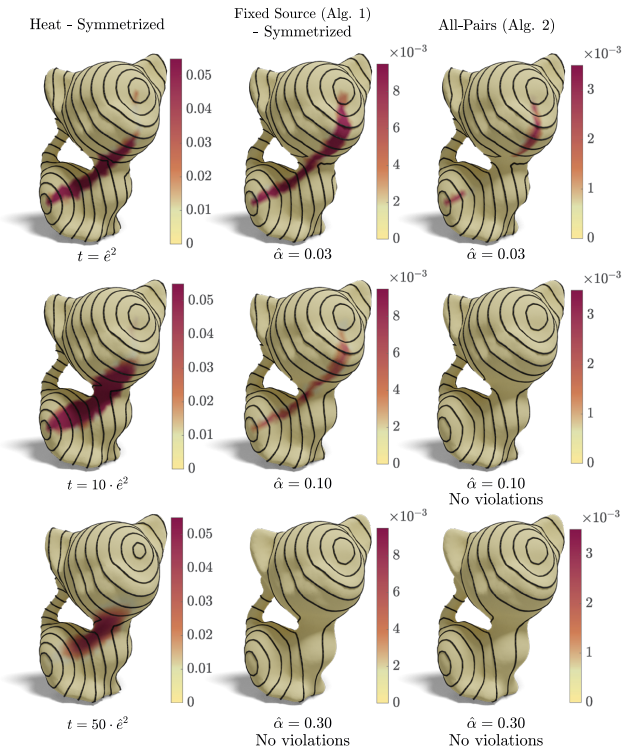


Figure 12: Triangle inequality violation. For a fixed pair of points (visualized with the distance isolines) we compute the triangle inequality error for all the other points. We compare the symmetrized heat method, our symmetrized method and the all-pairs formulation (which is symmetric by construction). We use a few smoothing weights for each approach. Note, that for our approach the violation reduces as the smoothing weight grows.

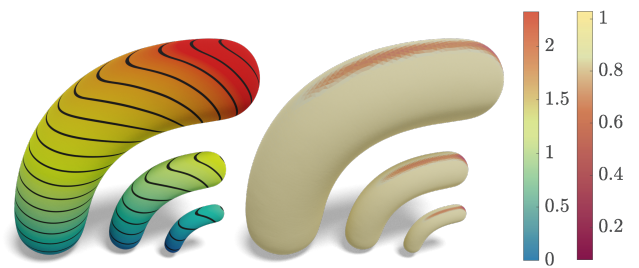


Figure 7: Scale invariance. While the distances are different between the uniformly scaled models, the area of the smoothing (where the norm of the gradient is not 1) is similar for all meshes. See the text for details.

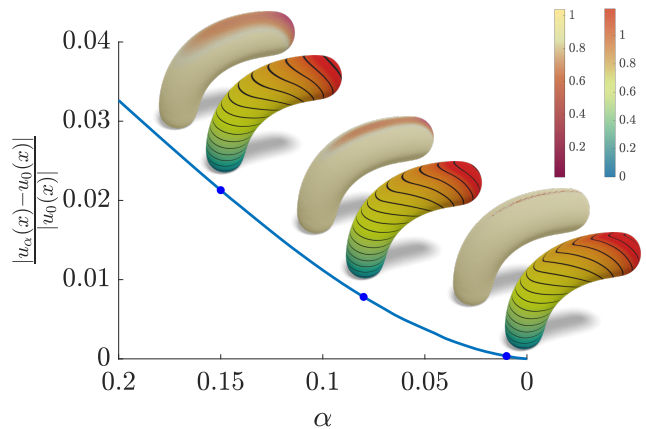


Figure 5: A numerical solution on triangle meshes. The distance converges towards u_0 as α approaches 0. Note the different smoothing regions, whose width depends on α .