Geodesic Paths and Distances

Report on *A Survey of Algorithms for Geodesic Paths and Distances*

Matthieu Pierre Boyer*
École Normale Supérieure
Paris, France
matthieu.boyer@ens.fr

Antoine Groudiev*
École Normale Supérieure
Paris, France
antoine.groudiev@ens.fr

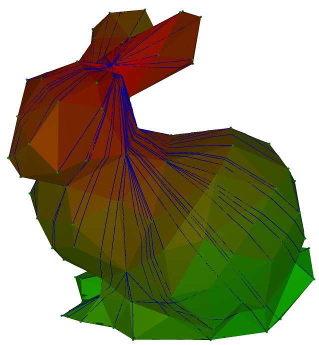

Figure 1: Geodesic paths on a mesh, computed with our implementation of the Improved Chen-Han algorithm. Faces are colored from red to green according to the distance of their barycenter to the source point. The geodesic path to each vertex is drawn in blue.

ABSTRACT

In this report, we investigate different methods to compute shortest-paths on meshed 2-manifolds embedded in $\mathbb{R}^3$, based on [CLPQ20]. We will most notably compare different types of methods, either coming from the resolution of PDEs on the manifold, or through the unfolding of the embedding to $\mathbb{R}^2$. We also provide from-scratch implementations of three methods, namely the heat method, fast marching, and the Improved Chen-Han algorithm, and compare their performances on several datasets.

INTRODUCTION

1 MATHEMATICAL REMINDERS

A 2-manifold (without boundary) is a topological space in which all points have nieghbourhoods homeomorphic to disks (without boundary) in $\mathbb{R}^2$. This means that zooming enough on every point looks like the plane.

In our context, we will be given a 2-manifold already meshed[1], that is, a finite set $\mathbb{V} \subseteq \mathbb{R}^3$ of vertices (of cardinal $n_{\mathbb{V}}$) and a finite set $\mathcal{F} \subseteq [\![1, n_{\mathcal{F}}]\!]^3$ of faces, given by the indices of the associated vertices. The edges E of the manifold are given by any subsets of size two of $f \in \mathcal{F}$. Because we can't have continuous functions, functions on the manifold will be represented as functions on $\mathbb{V}$, on E or on $\mathcal{F}$. As such, we will take any function and interpolate it linearly on each face, giving us a piecewise-linear function.

A path on a piecewise-linear manifold can then be understood as interpolating on pieces of the manifold, or directly computing the curves on the mesh. The quality of the approximation by the mesh of the 2-manifold will never be taken into account in the quality results.

2 PDE-BASED METHODS

In this section we will study methods inspired by partial differential equations that arise from models of physical phenomena. Indeed, many physical phenomena propagate along the surfaces over time, and dissipate over space, thus allowing us to retrieve geodesics from solutions to the equations.

2.1 General Theory

Consider the parabolic heat equation

$$\frac{\mathrm{d}}{\mathrm{d}t} u_t = \Delta u_t.$$

Here u_t is the temperature profile at time t and Δ is the laplacian operator (or divergence of the gradient operator). However, on a piecewise-linear manifold, because functions on vertices are interpolated linearly to become functions on the manifold, the gradient is piecewise-constant and can be computed explicitly from the values at each vertex. Consider a face $f \in \mathcal{F}$ with vertices $p_i, p_j, p_k \in \mathbb{R}^3$. Let $e_1 = p_j - p_i$ and $e_2 = p_k - p_i$. The face normal is

$$n_f = \frac{e_1 \times e_2}{|e_1 \times e_2|}$$

where $\times$ is the cross product in $\mathbb{R}^3$ and thus the gradient, being perpendicular to level curves is

$$\nabla u_{|f} = \frac{1}{2A_f} \sum_{l \in f} u_l (n_f \times e_l)$$

*Both authors contributed equally.
[1] [CLPQ20] gives a few methods to create such a meshing.

where $A_f = \frac{1}{2}|e_1 \times e_2|$ is the area of face f and e_l is the **opposite** to vertex l. Note that we can represent the gradient as a matrix $G \in \mathbb{R}^{n_{\mathbb{V}} \times 3n_{\mathcal{F}}}$, although this representation is really inefficient for practical computation. The definition of the divergence then comes from the Gauss-Ostrogradski theorem by intregration by parts

$$(\nabla \cdot X)_i = \frac{1}{A_i} \sum_{f \ni i} \sum_{e \in f} \frac{1}{2} \cot\left(\alpha_e^f\right) \langle X_f, e \rangle$$

where A_i is the Voronoi area associated with vertex i and α_e^f is the angle at the vertex opposite to edge e in f.

Finally, we can define the Laplace-Beltrami operator (the piecewise-linear version of the continuous laplacian) as $\Delta = (\nabla \cdot) \circ \nabla : \mathbb{V} \to \mathbb{R}$ or simply

$$(\Delta u)_i = \frac{1}{2A_i} \sum_{e=(i,j)} \left(\cot \alpha_{i,j} + \cot \beta_{i,j}\right) \left(u_i - u_j\right)$$

where $\alpha_{i,j}, \beta_{i,j}$ are the two angles opposite edge (i, j). One could then see the Laplace-Beltrami operator as a matrix $L \in \mathbb{R}^{n_{\mathbb{V}} \times n_{\mathbb{V}}}$.

After the spatial discretization of the laplace operator we just described, operating a time discretization in a single backward Euler step for some fixed time t will give us approximate solutions to the equation. If we want to find the distance maps from some set $X \subseteq \mathbb{V}$, we simply solve the linear equation associated to the continuous equation we need to solve.

2.2 Implementations

In our implementation, we compared multiple methods based on physical phenomena allowing to trace geodesics, or at least geodesic-like curves. Indeed, not all of those compute the true geodesic distance, but some sense of distance that can be drawn and integrated to find shortest paths.

Heat Method. This method is based on the heat equation $\nabla u_t = \frac{\mathrm{d}}{\mathrm{d}t} u_0$, which models the evolution of temperature profiles u_t in time in a given material, which here will be our surface, from the initial profil u_0. It can be derived We discretize it as:

$$(\mathrm{id} - t\Delta)u_t = u_0$$

Retrieving the true geodesic distance can the be done by first normalizing the gradient $X = -\nabla u_t / |\nabla u_t|$ of the solution of the above equation that points along geodesics, then solving the Poisson equation $\Delta\phi = \nabla \cdot X$ to retrieve the true distance function. This fact comes from the Varadhan formula $\phi(x, y) = \lim_{t \to 0} \sqrt{-4t \log k_{t,x}(y)}$.

[CWW17] suggest that the proper value of t to use for computations here is around h^2 with h the mean spacing between adjacent nodes, as $h\Delta$ is invariant with respect to scale.

Poisson Equations. The equations in this paragraph all allow to draw geodesics-like curves, although they do not give the actual metric on the manifold. They are all based on the Poisson equation

$$\Delta u = u_0,$$

which can be derived, for example, from the Maxwell equations to compute the electrostatic potential along a charge distribution, or from the momentum equation to compute the pressure in a incompressible fluid given its velocity.

Wavefront Propagation. The hyperbolic wavefront propagation equation

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2} u_t = c^2 \Delta u_t, \text{ or, } \Box u = 0,$$

where $\Box$ is the d'Alembert operator, models the propagation of a wave in a material, which will again be our surface here. It arises for example the response of the surface to some elastic deformation u considering the stress tensor $T = E\nabla u$ with E the homogenous modulus of elasticity, and considering the inertial force $\rho \frac{\partial^2 u}{\partial t^2}$ caused by the local acceleration.

3 COMPUTATIONAL GEOMETRY METHODS

While PDE-based methods provide a general framework to compute geodesic distances on arbitrary, possibly non-meshed manifolds, we can leverage the fact that our manifold is meshed to design more efficient algorithms. A whole class of such algorithms, arising from computational geometry, can be classified according to two main caracteristics.

Global vs. local. Global methods compute globally optimal paths, up to numerical precision. Local methods start from an initial non-geodesic path and iteratively improve it until convergence to a local minimum.

Exact vs. approximate. Exact methods compute the exact geodesic distance, while some approximate methods trade accuracy for speed, and compute only an approximation of the geodesic distance within some guaranteed error bound.

In the following, we focus on a global and exact method, the Improved Chen-Han algorithm [CH90, XW09].

3.1 Continuous Dijkstra

The Improved Chen-Han (ICH) algorithm is based on the continuous Dijkstra paradigm introduced by Mitchell et al. for the Mitchell-Mount-Papadimitriou (MMP) algorithm [MMP87]. The main idea is to apply Dijkstra's algorithm in the continuous setting of a piecewise-linear manifold.

A first natural approach could be to consider the graph formed by the vertices and edges of the mesh, and run Dijkstra's algorithm on it. However, this would only yield paths that follow the edges of the mesh, which can be arbitrarily longer than the true geodesic paths on the surface.

Instead, we would like to consider each point on the surface as a potential node in the graph. However, since the shortest path on a single face is always a straight line, it is sufficient to only consider points on the edges of the mesh as potential nodes. This still yields an infinite number of potential nodes; a key concept of the MMP algorithm is to use instead a data structure called *windows*, which encapsulate a continuous interval of points on an edge, along with the information needed to propagate the shortest path from these points to adjacent faces.

Formally, a window

$$w = (s, p, e, b_0, b_1, d_0, d_1, \sigma)$$

is defined on edge e, between points b_0 and b_1 (see Figure 2). The window represents shortest paths originating from a "pseudosource" p, at a distance σ from the actual source s. Distances from p to the

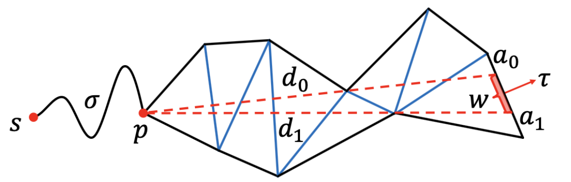

Figure 2: Illustration of a window $w = (s, p, e, b_0, b_1, d_0, d_1, \sigma)$**.**

endpoints b_0 and b_1 of the window, respectively d_0 and d_1, are also stored.

REFERENCES

[CH90] Jindong Chen and Yijie Han. Shortest paths on a polyhedron. In *Proceedings of the sixth annual symposium on Computational geometry*, pages 360–369, 1990.

[CLPQ20] Keenan Crane, Marco Livesu, Enrico Puppo, and Yipeng Qin. A survey of algorithms for geodesic paths and distances, 2020.

[CWW17] Keenan Crane, Clarisse Weischedel, and Max Wardetzky. The heat method for distance computation. *Commun. ACM*, 60(11):90–99, October 2017.

[MMP87] Joseph SB Mitchell, David M Mount, and Christos H Papadimitriou. The discrete geodesic problem. *SIAM Journal on Computing*, 16(4):647–668, 1987.

[XW09] Shi-Qing Xin and Guo-Jin Wang. Improving chen and han's algorithm on the discrete geodesic problem. *ACM Transactions on Graphics (TOG)*, 28(4):1–8, 2009.