SIAM J. NUMER. ANAL.
Vol. 64, No. 1, pp. 224–250
© 2026 Society for Industrial and Applied Mathematics

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

A PRIMAL-DUAL LEVEL SET METHOD FOR COMPUTING GEODESIC DISTANCES*

HAILIANG LIU† AND LAURA ZINNEL†

Abstract. The numerical computation of shortest paths or geodesics on surfaces, along with the associated geodesic distance, has a wide range of applications. Compared to Euclidean distance computation, these tasks are more complex due to the influence of surface geometry on the behavior of shortest paths. This paper introduces a primal-dual level set method for computing geodesic distances. A key insight is that the underlying surface can be implicitly represented as a zero level set, allowing us to formulate a constraint minimization problem. We employ the primal-dual methodology, along with regularization and acceleration techniques, to develop our algorithm. This approach is robust, efficient, and easy to implement. We establish a convergence result for the high resolution PDE system, and numerical evidence suggests that the method converges to a geodesic in the limit of refinement.

Key words. geodesic, primal-dual, level set, convergence

MSC codes. 90C46, 93B40

DOI. 10.1137/24M1721086

1. Introduction.

A geodesic is a curve on a surface that minimizes local length, serving as a useful generalization of straight lines in Euclidean space. Geodesic distances play a crucial role in both theoretical and applied problems, including those in computer vision [10], conformal field theory [18], quantum field theory [7], and geophysical visualization [13]. Therefore, approximating the minimal geodesic path between two points on a surface is essential in many application domains.

The approximation tasks can be framed as constrained optimization problems; however, they are often highly nonconvex, and effective algorithms for computing geodesic distances are limited. For complex surfaces represented in 3D meshes, specialized algorithms such as the fast-marching method [19], the Dijkstra algorithm ([13, 21, 20]), or the heat flow method [11] are often employed to find the shortest path across the surface. Although effective, these methods often involve considerable complexity due to the construction of a mesh or graph representation. In contrast, our algorithm approximates geodesics on surfaces without requiring any surface discretization.

In this work, we adopt a different strategy: using the level set of a function to represent the surface, rather than employing a triangulated mesh. In the literature, methods based on implicit level set surfaces are referred to as level set methods, which are particularly effective in image science applications, including image segmentation and rendering [23]. In this work, we introduce a primal-dual method for approximating geodesic distances and minimal geodesic paths on surfaces represented by level set functions.

*Received by the editors December 27, 2024; accepted for publication (in revised form) October 29, 2025; published electronically February 6, 2026.
https://doi.org/10.1137/24M1721086
Funding: This work by second author is partially supported by the National Science Foundation grant DGE-2152117.
†Department of Mathematics, Iowa State University, Ames, IA 50011 USA (hliu@iastate.edu, lezinnel@iastate.edu).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

1.1. Background. In differential geometry [1], the *tangent space* to a surface Ω at a point p, is defined by

$$T_p\Omega = \{\dot{\xi}(0) \mid \xi : (-\varepsilon, \varepsilon) \to \Omega \text{ is } C^\infty \text{ and } \xi(0) = p\}.$$

A surface is considered *orientable* if there exists a continuous function N on Ω such that for each p, the vector $N(p)$ is orthogonal to $T_p\Omega$. This field of normal vectors N is referred to as an *orientation* of Ω. Notably, if Ω is the zero level set of function $\phi : \mathbb{R}^3 \to \mathbb{R}$, and $\gamma : [0,1] \to \Omega$ is a path on Ω, then

$$\phi(\gamma(t)) = 0 \Rightarrow \nabla\phi(\gamma(t)) \cdot \dot{\gamma}(t) = 0 \quad \text{for} \quad t \in [0,1].$$

Thus, $\dot{\gamma}(t) \in T_p(\Omega)$. If the surface is smooth and the vector field is defined as

$$N(p) := \frac{1}{|\nabla\phi(p)|}\nabla\phi(p)$$

for every point $p = \gamma(t)$ along the curve, then N provides an orientation of Ω.

A curve $\gamma : [0,1] \to \Omega$ is a geodesic if and only if $\ddot{\gamma}$ is orthogonal to the tangent plane of Ω at $\gamma(t)$ for all $t \in [0,1]$ [25]. Further, for a zero level set surface Ω with orientation N as defined above, if $\ddot{\gamma}$ is parallel to $N(p)$ for every point $p = \gamma(t)$ along the curve, then γ is a geodesic.

It is known that if γ is a shortest path, then γ is a geodesic; however, not every geodesic represents a shortest path [25, 10]. For example, if p and q are points on a sphere, then there is both a long and a short great circle arc joining these points. Both of these great circle arcs are geodesics, but the short great circle arc is the shortest geodesic [25]. The fact that not all geodesics are shortest paths, along with the fact that a shortest geodesic between two points is not necessarily unique, complicates the task of finding a minimal geodesic on a surface.

1.2. Problem setup. Our goal is to develop an efficient algorithm for iteratively computing a geodesic and the geodesic distance (shortest path) between two points on a surface. Let Ω be a surface represented as the zero level set $\{x \in \mathbb{R}^3 \mid \phi(x) = 0\}$ of a function $\phi : \mathbb{R}^3 \to \mathbb{R}$. For $p, q \in \Omega$ and a path $\gamma : [0,1] \to \mathbb{R}^3$ with $\gamma(0) = p$ and $\gamma(1) = q$, the **length** of γ is calculated by

$$\int_0^1 |\dot{\gamma}(t)|\, dt.$$

We are particularly interested in paths from p to q on Ω. That is, we seek path γ to be in the set $\Gamma(p,q,\Omega)$ defined by

$$\Gamma(p,q,\Omega) = \{\gamma \,:\, \gamma(0) = p, \gamma(1) = q, \text{ and } \phi(\gamma(t)) = 0\, \forall t \in [0,1]\}.$$

Recall that a path $\gamma : [0,1] \to \mathbb{R}^3$ is a *geodesic* if $\gamma \in \Gamma(p,q,\Omega)$ and $\ddot{\gamma}$ is orthogonal to the tangent plane of Ω at $\gamma(t)$ for all $t \in [0,1]$. Hence,

$$\ddot{\gamma} \cdot \dot{\gamma} = 0. \tag{1.1}$$

It has been shown that the shortest length path between any two points on a surface must be a geodesic [25]. The **geodesic distance** from p to q is the length of the *shortest length path* γ from p to q on the surface Ω. That is, the length of path γ such that

$$\gamma = \underset{\xi \in \Gamma(p,q,\Omega)}{\arg\min} \int_0^1 |\dot{\xi}(t)|\, dt. \tag{1.2}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Using this formulation, we propose determining the geodesic distance by solving the following optimization problem:

$$\min_{\gamma\in\Gamma(p,q,\Omega)} \frac{1}{2}\int_0^1 |\dot{\gamma}(t)|^2\, dt,$$

which can be shown to be equivalent to (1.2). To eliminate the level set constraint, a Lagrangian multiplier $\lambda(t) \in C^1[0,1]$ is introduced in an inf-sup problem:

$$\inf_{\gamma}\sup_{\lambda}\left\{\frac{1}{2}\int_0^1 |\dot{\gamma}(t)|^2\, dt + \int_0^1 \lambda(t)\phi(\gamma(t))\, dt\right\}.$$

A simple strategy is to alternate a gradient-like descent in γ and a gradient-like ascent in λ. This yields the following iterative scheme:

$$\begin{cases}\lambda_{k+1}(t) = \lambda_k(t) + \tau_\gamma \phi(\gamma_k(t)),\\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\lambda\left(-\ddot{\gamma}_k(t) + \lambda_{k+1}(t)\nabla\phi(\gamma_k(t))\right),\end{cases}$$

where $\tau_\gamma, \tau_\lambda$ are tunable step sizes. However, such basic updates may not perform well in all cases due to stability issues inherent in the formulation. To address this, we augment the functional with a regularization term of the form:

$$\mathcal{L}_\varepsilon[\gamma,\lambda] = \frac{1}{2}\int_0^1 |\dot{\gamma}|^2\, dt + \int_0^1 \lambda(t)\phi(\gamma(t))dt - \frac{\varepsilon}{2}\int_0^1 \lambda^2(t)\, dt$$

and enhance the gradient descent-ascent updates with a relaxation step, resulting in the following update scheme:

$$\begin{cases}\lambda_{k+1}(t) = \dfrac{1}{1+\varepsilon\tau_\lambda}(\lambda_k(t) + \tau_\lambda\phi(\gamma_k(t))),\\ \tilde{\lambda}_{k+1}(t) = \lambda_{k+1}(t) + \omega(\lambda_{k+1}(t) - \lambda_k(t)),\\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma(-\ddot{\gamma}_k(t) + \tilde{\lambda}_{k+1}(t)\nabla\phi(\gamma_k(t))),\end{cases}$$

where $\varepsilon > 0$ is a regularization parameter, $\omega \in [0,1]$ controls the degree of relaxation, and $\tau_\lambda, \tau_\gamma$ are tunable step sizes. This refinement has been shown to significantly improve stability and numerical performance. Notably, the use of relaxation in this manner has also been explored in the Primal-Dual Hybrid Gradient (PDHG) method [6, 15] which addresses saddle-point problems of the primal-dual form:

$$\min_{y\in Y}\max_{x\in X} g(x) + \left(\langle Ax, y\rangle - f^*(y)\right).$$

PDHG has proven to be an efficient algorithm for a wide range of problems, particularly in imaging applications, and has gained popularity due to its strong convergence guarantees. For further details, readers are referred to [6], specifically section 2.5 for a brief description.

Our problem setup, which employs a novel level set representation, has not been previously explored. This approach establishes a primal-dual connection between the curve γ and the multiplier λ, resulting in an algorithm with potential applications in various fields.

1.3. Contribution. Our contribution is threefold.

- In this paper, we introduce a simple and efficient algorithm for approximating the minimal geodesic path between two points on a level set surface.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

- We show that, under certain structural assumptions on the level set function, the continuous PDE system exhibits asymptotic convergence, thereby supporting the convergence of the discrete algorithm. Furthermore, we demonstrate that, upon convergence, the algorithm approximates a minimal geodesic curve.
- We also explored two variations of the base algorithm, both of which demonstrated promising performance in various examples. To illustrate, we present results for the base algorithm and its variations on examples such as the surfaces of a sphere, a torus, and the Stanford Bunny.

1.4. Related work.

Geodesic solvers. Several popular methods exist for computing geodesic distances. The fast-marching method has been used to approximate the distances between points on a triangulated mesh representation of a surface [19]. The Dijkstra algorithm, an algorithm developed for finding the shortest paths in graphs, has been used to approximate the shortest paths on polyhedral surfaces [22]. In addition, heat flow methods have been developed for the geodesic approximation using the heat kernel function [11]. These methods typically require the surface to be represented as a triangulated mesh or as a graph. In contrast, our method can be applied directly to level set surfaces without requiring prior discretizaiton. For graph-based problems, we refer the reader to [9, 4] for discussions on the use of Lipschitz embedding and Lipschitz learning to the study of shortest paths on graphs.

Level set methods. Level set surface representations have proven to be valuable tools for various image processing tasks [23]. Specifically, we are interested in using level set methods for brain imaging applications, such as sulci segmentation [27], cortical surface reconstruction [12], and cortical surface registration and labeling [17].

The application of level set methods involves initializing a level set function to represent a given surface. While this representation is not unique, a common choice is the signed distance function (SDF), where the value at each point is the distance to the nearest point on the surface, positive inside and negative outside. The SDF can be explicitly defined using a crossing-time approach or fast marching method, as described by [23]. If the surface is given as a set of scattered points, a level set function can be reconstructed by iteratively refining the level set function to minimize an energy functional based on the data's proximity and smoothness [28]. Alternatively, deep learning methods have also been implemented for generating level set surface representations directly from input data, such as point clouds or image features [24]. Refer to section 4.4 for an explanation of how the SDF satisfies the structural assumption outlined in section 3.3 for the example shown in Figure 9.

Level-set methods have been applied to model the motion of curves on surfaces. In one approach introduced by [8], a level set function represents the surface, while an additional level set function is used to define the curve. The curve is then represented by the intersection of the two zero level sets of these two functions. Our approach represents the curve $\gamma : [0,1] \to \mathbb{R}^3$ directly as a parametrized curve, rather than defining γ through the intersection of two zero level sets.

Primal-dual algorithms. Primal-dual formulations for constrained optimization have been extensively studied, with applications ranging from imaging to broader convex optimization problems (see [26, 16, 14, 6, 3]). These methods for managing constraints have played a significant role in the development of primal-dual algorithms. Building on this foundation, our work incorporates a level-set constraint into the framework.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Acceleration. While gradient descent/ascent is a reliable first-order optimization method, it often suffers from slow convergence in many practical problems. This limitation has motivated the development of accelerated variants that incorporate momentum, such as Polyak's heavy ball method (1964) and Nesterov's accelerated gradient descent (1983). In the context of calculus of variations problems defined over functions, a PDE-based acceleration framework was introduced by Benyamin, Calder, Sundaramoorthi, and Yezzi [2], further analyzed in [5]. These works show improved efficiency via wave-equation dynamics and explicit time-stepping schemes. In this work, we adopt the relaxation method, common used in the primal-dual hybrid gradient (PDHG) algorithm [6, 15].

1.5. Organization of the paper. In section 2, we define the geodesic approximation problem. Section 3 introduces an algorithm to solve it, followed by an analysis of the algorithm's behavior in section 4. Finally, concluding remarks and discussions are provided at the end.

2. Method. The uniqueness of geodesics in three-dimensional spaces depends on the specific geometry and topology of the space – particularly aspects such as curvature, singularities, and the presence of antipodal points or a cut locus. For most pairs of points on a given surface, the geodesic connecting them is unique. Therefore, we will focus on the case where a unique geodesic exists and defer discussion of nonunique geodesics to a later section.

2.1. Optimization problem. From now on, we define

$$L(p,q) = \int_0^1 |\dot{\gamma}(t)| dt$$

as the **geodesic length** from p to q, provided $\gamma(t)$ is a geodesic curve which is a solution to the minimization problem (1.2).

We want to find the geodesic distance $L(p,q)$ from p to q on surface Ω, represented by $\Omega = \{x \mid \phi(x) = 0\}$ for some function $\phi : \mathbb{R}^3 \to \mathbb{R}$. Define

$$d^2(p,q) = \min_{\gamma \in \Gamma(p,q,\Omega)} \int_0^1 |\dot{\gamma}(t)|^2 \, dt.$$

By Cauchy–Schwarz,

$$|p-q| \le \int_0^1 |\dot{\gamma}(t)| \, dt \le \left(\int_0^1 1 \, dt \right)^{1/2} \left(\int_0^1 |\dot{\gamma}(t)|^2 \, dt \right)^{1/2} = \left(\int_0^1 |\dot{\gamma}(t)|^2 \, dt \right)^{1/2},$$

hence we have

$$|p-q| \le L(p,q) \le d(p,q).$$

Note that for a geodesic, $L(p,q) = d(p,q)$. More precisely, we have the following result.

LEMMA 1. *If γ is a geodesic on Ω from p to q, then $L(p,q) = d(p,q)$. Moreover, the following statements are equivalent:*

(i) *γ is a constant-speed geodesic.*
(ii) *$\gamma \in C^1[0,1]$ and $|\dot{\gamma}(t)| = d(\gamma(0), \gamma(1))$ for all $t \in [0,1]$.*
(iii) *Let $m > 1$, γ solves a class of minimization problems*

$$\min_{\gamma \in \Gamma(p,q,\Omega)} \left\{ \int_0^1 |\dot{\gamma}(t)|^m dt \right\}.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Proof. (i) $\Leftrightarrow$ (ii): First, recall that by the definition of a geodesic, the path γ minimizes arc length, i.e.,

$$L(p,q) = \int_0^1 |\dot{\gamma}(t)| dt = d(p,q).$$

Since γ is a geodesic, it satisfies the Euler–Lagrange equations associated with arc length minimization. In particular, we compute:

$$\frac{d}{dt}|\dot{\gamma}(t)| = \frac{1}{|\dot{\gamma}(t)|}(\dot{\gamma}(t)\cdot\ddot{\gamma}(t)) = 0.$$

Therefore, $|\dot{\gamma}|$ is constant in t. Therefore, γ moves at constant speed. Thus,

$$L(p,q) = \int_0^1 |\dot{\gamma}(t)|\, dt = |\dot{\gamma}(t)| = \left(\int_0^1 |\dot{\gamma}(t)|^2\, dt\right)^{1/2} = d(p,q).$$

The equivalent between (i) and (ii) follows directly (see also a proof in [25] on page 216).

(ii) $\Rightarrow$ (iii): Assume $\gamma \in C^1[0,1]$ and $|\dot{\gamma}(t)| = d(\gamma(0),\gamma(1))$ for all $t \in [0,1]$. For $m > 1$, choose m^* such that $\frac{1}{m} + \frac{1}{m^*} = 1$, then Hölder's inequality gives

$$\int_0^1 |\dot{\gamma}(t)|\, dt \le \left(\int_0^1 1^{m^*}\, dt\right)^{\frac{1}{m^*}} \left(\int_0^1 |\dot{\gamma}(t)|^m\, dt\right)^{\frac{1}{m}} = \left(\int_0^1 |\dot{\gamma}(t)|^m\, dt\right)^{\frac{1}{m}}.$$

This implies

$$\left(\int_0^1 |\dot{\gamma}(t)|\, dt\right)^m \le \int_0^1 |\dot{\gamma}(t)|^m\, dt.$$

Since $|\dot{\gamma}(t)| = d(p,q)$ is a constant, the inequality becomes an equality

$$\int_0^1 |\dot{\gamma}(t)|^m\, dt = d(p,q)^m = \left(\int_0^1 |\dot{\gamma}(t)|\, dt\right)^m.$$

Therefore, γ achieves the minimum of the functional $\int_0^1 |\dot{\gamma}(t)|^m dt$ over $\Gamma(p,q,\Omega)$, proving (iii).

(iii) $\Rightarrow$ (i): Assume γ minimizes $\int_0^1 |\dot{\gamma}(t)|^m dt$. This means Hölder's inequality must be tight as an equality, which only occurs when $|1|^{m^*} = 1$ and $|\dot{\gamma}^*(t)|^m$ are linearly dependent. This implies $|\dot{\gamma}(t)|$ must be constant almost everywhere. Therefore, γ is a constant-speed geodesic. ∎

Considering the smoothness of the underlying functional defined by $d(p,q)$, we propose to find the geodesic distance from p to q by solving the following optimization problem:

$$\min_{\gamma\in\Gamma(p,q,\Omega)} \frac{1}{2}\int_0^1 |\dot{\gamma}(t)|^2\, dt.$$

To eliminate the level-set constraint, we introduce a Lagrangian multiplier $\lambda(t) \in C^1[0,1]$, so that the constraint optimization problem can be reformulated as an inf-sup problem:

$$\inf_{\gamma}\sup_{\lambda} \mathcal{L}[\gamma,\lambda],$$

where $\mathcal{L}$ is defined by

$$\mathcal{L}[\gamma,\lambda] = \frac{1}{2}\int_0^1 |\dot{\gamma}(t)|^2\, dt + \int_0^1 \lambda(t)\phi(\gamma(t))\, dt.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

2.2. Optimality conditions. At the optimal pair (γ, λ), the variational derivative $\delta\mathcal{L} = 0$. Observe that

$$\begin{aligned}\delta\mathcal{L} &= \int_0^1 \dot{\gamma}(t)\delta\dot{\gamma}(t)\,dt + \int_0^1 \lambda(t)\nabla\phi(\gamma)\cdot\delta\gamma\,dt + \int_0^1 \phi(\gamma(t))\delta\lambda\,dt \\ &= \dot{\gamma}\delta\gamma\big|_0^1 - \int_0^1 \ddot{\gamma}\delta\gamma\,dt + \int_0^1 \lambda(t)\nabla\phi(\gamma)\cdot\delta\gamma\,dt + \int_0^1 \phi(\gamma(t))\delta\lambda\,dt \\ &= \dot{\gamma}\delta\gamma\big|_0^1 + \int_0^1 (-\ddot{\gamma} + \lambda(t)\nabla\phi(\gamma))\cdot\delta\gamma\,dt + \int_0^1 \phi(\gamma(t))\delta\lambda\,dt.\end{aligned}$$

Note that $\gamma(0) = p$ and $\gamma(1) = q$ are fixed, which implies that $\delta\gamma = 0$ at $t = 0$ and $t = 1$. Consequently, the optimal solution must satisfy the following conditions:

$$\begin{cases} -\ddot{\gamma}(t) + \lambda(t)\nabla\phi(\gamma(t)) = 0, \\ \phi(\gamma(t)) = 0, \quad \forall t \in [0,1], \\ \gamma(0) = p, \quad \gamma(1) = q. \end{cases}$$

According to [25, Definition 9.1.1], a curve γ on a surface $\mathcal{S}$ is called a *geodesic* if its acceleration vector $\ddot{\gamma}(t)$ is either zero or perpendicular to the tangent plane of the surface at the point $\gamma(t)$; that is, $\ddot{\gamma}(t)$ is parallel to the unit normal at that point, for all values of the parameter t. We now state the following result.

LEMMA 2. *A point (λ,γ) satisfies the optimality conditions if and only if γ is a geodesic.*

Proof. ($\Rightarrow$) Suppose (λ,γ) satisfies the optimality conditions. Then, by definition, $\phi(\gamma(t)) = 0$ for all $t \in [0,1]$, $\gamma(t)$ lies on the surface $\phi = 0$. Further, the conditions $\gamma(0) = p$ and $\gamma(1) = q$ imply γ is a path from p to q on the surface. The last condition says $\ddot{\gamma}(t) = \lambda(t)\nabla\phi(\gamma(t))$ for all $t \in (0,1)$, that is, the acceleration vector $\ddot{\gamma}$ is parallel to $\nabla\phi(\gamma(t))$ for all $t \in (0,1)$. Since $\nabla\phi(\gamma(t))$ is normal to the surface at $\gamma(t)$, $\ddot{\gamma}(t)$ is parallel to its unit normal for all $t \in (0,1)$. Hence, γ has zero tangential acceleration and lies entirely on the surface, satisfying the definition of a geodesic.

($\Leftarrow$) Now assume γ is a geodesic on the surface defined by $\phi = 0$, connecting points p and q. Then, $\phi(\gamma(t)) = 0$ for all $t \in (0,1)$, $\gamma(0) = p$, and $\gamma(1) = q$ must be satisfied. Since γ is a geodesic, its acceleration vector $\ddot{\gamma}(t)$ is orthogonal to the surface at each point $\gamma(t)$, i.e., normal to the tangent plane. Because $\nabla\phi(\gamma(t))$ is normal to the tangent plane at $\gamma(t)$, there exists a scalar function $\lambda(t)$ such that $\ddot{\gamma}(t) = \lambda(t)\nabla\phi(\gamma(t))$ for all $t \in (0,1)$.Therefore, the pair (λ, γ) satisfies the optimality conditions. ∎

2.3. Gradient descent-ascent. For the inf-sup problem, we can use a Gradient Descent-Ascent approach to update $\gamma(t)$ and $\lambda(t)$ for all $t \in [0,1]$. In the $(k+1)th$ iteration, the updates are defined by

$$\begin{aligned}\lambda_{k+1}(t) &= \lambda_k(t) + \tau_\lambda\left(\frac{\delta}{\delta\lambda}\mathcal{L}[\gamma,\lambda]\right), \\ \gamma_{k+1}(t) &= \gamma_k(t) - \tau_\gamma\left(\frac{\delta}{\delta\gamma}\mathcal{L}[\gamma,\lambda]\right),\end{aligned}$$

where τ_γ and τ_λ are tunable step sizes. Note that

$$\begin{aligned}\frac{\delta}{\delta\lambda}\mathcal{L}[\gamma,\lambda] &= \phi(\gamma(t)), \\ \frac{\delta}{\delta\gamma}\mathcal{L}[\gamma,\lambda] &= -\ddot{\gamma}(t) + \lambda(t)\nabla\phi(\gamma(t)).\end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

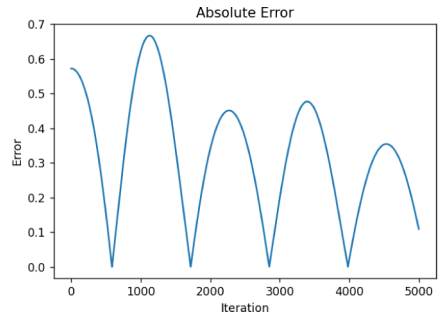

FIG. 1. *Absolute error of the approximated geodesic distance between antipodal points on the sphere, plotted against the number of iterations for the update scheme described above.*

This results in the following updates:

$$\begin{cases} \lambda_{k+1}(t) = \lambda_k(t) + \tau_\lambda \phi(\gamma_k(t)), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma \left(-\ddot{\gamma}_k(t) + \lambda_{k+1}(t)\nabla\phi(\gamma_k(t))\right). \end{cases} \tag{2.1}$$

One issue with the algorithm is that the maximization of $\mathcal{L}$ with respect to λ may not be well-defined, potentially causing a lack of convergence for $\lambda_k(t)$. When this update scheme was applied to approximate the geodesic distance between antipodal points on the surface of a unit sphere, the algorithm failed to converge. Figure 1 illustrates the absolute error in the approximate geodesic distance relative to the number of iterations.

2.4. Regularization. To enhance the algorithm's performance, we introduce a regularization term to the functional $\mathcal{L}$ to ensure a well-defined maximum of $\mathcal{L}$ with respect to λ. For $\varepsilon > 0$, we define

$$\mathcal{L}_\varepsilon[\gamma, \lambda] = \frac{1}{2}\int_0^1 |\dot{\gamma}|^2\, dt + \int_0^1 \lambda(t)\phi(\gamma(t))dt - \frac{\varepsilon}{2}\int_0^1 \lambda^2(t)\, dt.$$

We then have

$$\begin{aligned} \frac{\delta}{\delta\lambda}\mathcal{L}_\varepsilon[\gamma, \lambda] &= \phi(\gamma(t)) - \varepsilon\lambda(t), \\ \frac{\delta}{\delta\gamma}\mathcal{L}_\varepsilon[\gamma, \lambda] &= -\ddot{\gamma}(t) + \lambda(t)\nabla\phi(\gamma(t)). \end{aligned}$$

Thus, the gradient descent-ascent updates become

$$\begin{cases} \lambda_{k+1}(t) = \lambda_k(t) + \tau_\lambda(\phi(\gamma_k(t)) - \varepsilon\lambda_{k+1}(t)), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma(-\ddot{\gamma}_k(t) + \lambda_{k+1}(t)\cdot\nabla\phi(\gamma_k(t))). \end{cases}$$

This can be rewritten as

$$\begin{cases} \lambda_{k+1}(t) = \dfrac{1}{1+\varepsilon\tau_\lambda}(\lambda_k(t) + \tau_\lambda\phi(\gamma_k(t))), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma(-\ddot{\gamma}_k(t) + \lambda_{k+1}(t)\cdot\nabla\phi(\gamma_k(t))). \end{cases}$$

2.5. Acceleration. The primal-dual hybrid gradient (PDHG) algorithm is an efficient first-order optimization algorithm for solving saddle point problems [6].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Generally, if $\langle\cdot,\cdot\rangle$ represents an inner product on Hilbert spaces X and Y, with the norm defined as $\|\cdot\| = \langle\cdot,\cdot\rangle^{1/2}$, then the PDHG algorithm is employed to solve saddle-point problems of the primal-dual form

$$\min_{y\in Y}\max_{x\in X} g(x) + (\langle Ax, y\rangle - f^*(y)),$$

where functionals $g : X \to [0,+\infty]$ and $f : Y \to [0,+\infty]$ are proper, convex, and lower semicontinuous, $A : X \to Y$ is a continuous linear operator, and f^* is the convex conjugate of f. The PDHG algorithm uses gradient ascent and descent steps as follows:

$$\begin{cases} x_{k+1} = \mathrm{prox}_{\tau_x g}(x_k - \tau_x L^* y_k), \\ \tilde{x}_{k+1} = x_{k+1} + \omega(x_{k+1} - x_k), \\ y_{k+1} = \mathrm{prox}_{\tau_y f^*}(y_k - \tau_y L\tilde{x}_{k+1}), \end{cases}$$

where τ_y, τ_x, and ω are constants. The convergence of this algorithm is guaranteed when $\tau_x\tau_y\|A\|^2 < 1$. In [6], the convergence of the PDHG algorithm was analyzed for $\omega = 1$, demonstrating a convergence rate of $O(1/k)$ in finite dimensional spaces.

The relaxation step in the primal-dual hybrid gradient (PDHG) algorithm is designed to accelerate convergence by leveraging information from previous iterations to improve the current update – a concept similar to momentum in optimization. Inspired by such relaxation step, we introduce the following update:

$$\begin{cases} \lambda_{k+1}(t) = \dfrac{1}{1+\varepsilon\tau_\lambda}(\lambda_k(t) + \tau_\lambda\phi(\gamma_k(t))), \\ \tilde{\lambda}_{k+1}(t) = \lambda_{k+1}(t) + \omega(\lambda_{k+1}(t) - \lambda_k(t)), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma(-\ddot{\gamma}_k(t) + \tilde{\lambda}_{k+1}(t)\nabla\phi(\gamma_k(t))), \end{cases} \tag{2.2}$$

where $\omega \in (0,1]$ is the relaxation parameter. This creates a sort of "extrapolated" variable $\tilde{\lambda}$, which combines the current and previous values of the λ variable. Before implementing the algorithm, we conduct a primary analysis using ODE approach to gain deeper insights into the behavior and stability of the update.

Note that if ϕ is linear –for instance, $\phi(\gamma) = a\cdot\gamma$, where $a \in \mathbb{R}^3$ is a constant vector, then the objective functional falls into the class suitable for applying the PDHG method [6]. In the present infinite dimensional setting, using the optimal relaxation parameter $\omega = 1$, the update becomes

$$\begin{cases} \lambda_{k+1}(t) = \dfrac{1}{1+\varepsilon\tau_\lambda}(\lambda_k(t) + \tau_\lambda\phi(\gamma_k(t))), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma(-\ddot{\gamma}_{k+1}(t) + (2\lambda_{k+1}(t) - \lambda_k(t))\nabla\phi(\gamma_k(t))). \end{cases} \tag{2.3}$$

A detailed convergence rate analysis will be presented in section 5: the result asserts that a convergence rate of $O(1/k)$ is ensured if $\tau_\lambda\tau_\gamma|a|^2 < 1$. This is consistent with the known result for PDHG.

3. Analysis of dynamic behavior. One effective way for analyzing optimization algorithms is to first examine their continuous limit through differential equations. This method provides valuable insights into the convergence properties of the algorithm.

We begin by reformulating the proposed algorithm as a PDE system.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

3.1. PDE system. The discrete dynamics can be viewed as a discretization of the continuous PDE system. To illustrate this, we note that

$$\begin{aligned}\tilde{\lambda}_{k+1} &= (1+w)\lambda_{k+1} - w\lambda_k \\ &= \frac{1+w}{1+\varepsilon\tau_\lambda}(\lambda_k + \tau_\lambda\phi(\gamma_k)) - w\lambda_k \\ &= \frac{1-\varepsilon w\tau_\lambda}{1+\varepsilon\tau_\lambda}\lambda_k + \frac{(1+w)\tau_\lambda}{1+\varepsilon\tau_\lambda}\phi(\gamma_k) \\ &= \left(1 - \frac{1+w}{1+\varepsilon\tau_\lambda}\varepsilon\tau_\lambda\right)\lambda_k + \frac{(1+w)\tau_\lambda}{1+\varepsilon\tau_\lambda}\phi(\gamma_k).\end{aligned}$$

Hence,

$$\begin{aligned}\frac{\lambda_{k+1}-\lambda_k}{\tau_\lambda} &= -\frac{\varepsilon}{1+\varepsilon\tau_\lambda}\lambda_k + \frac{1}{1+\varepsilon\tau_\lambda}\phi(\gamma_k), \\ \frac{\gamma_{k+1}-\gamma_k}{\tau_\gamma} &= \ddot{\gamma}_k(t) - \nabla\phi(\gamma_k)\cdot\left[\left(1 - \frac{1+w}{1+\varepsilon\tau_\lambda}\varepsilon\tau_\lambda\right)\lambda_k + \frac{(1+w)\tau_\lambda}{1+\varepsilon\tau_\lambda}\phi(\gamma_k)\right].\end{aligned}$$

For any $k>0$, let $\tau := \tau_\gamma = \frac{\tau_\lambda}{1+\varepsilon\tau_\lambda}$ and $s_k = k\tau$. We assume that (λ_k, γ_k) correspond to values at $s = s_k$ for some sufficiently smooth functions $(\lambda(s,t), \gamma(s,t))$. As $\tau \to 0$, while keeping $\alpha = (1+\omega)\tau$, we have

$$\begin{cases}\lambda' = -\varepsilon\lambda + \phi(\gamma), \\ \gamma' = \ddot{\gamma} - \nabla\phi(\gamma)\cdot((1-\alpha\varepsilon)\lambda + \alpha\phi(\lambda)),\end{cases} \tag{3.1}$$

where $\lambda = \lambda(t,s)$ and $\lambda' = \frac{\partial}{\partial s}\lambda(t,s)$ for the training step index s and curve parameter t.

Regarding the PDE system, we state the following result:

THEOREM 1. *As $\tau_\lambda, \tau_\gamma \to 0$, the update in 2.2 can be expressed as a discrete-time update of the following PDE system:*

$$\begin{aligned}\lambda' &= -\varepsilon\lambda + \phi(\gamma), \\ \gamma' &= \ddot{\gamma} - \nabla\phi(\gamma)\cdot((1-\alpha\varepsilon)\lambda + \alpha\phi(\gamma))\end{aligned}$$

with the initial condition $\lambda(t,0) = \lambda_0(t)$ and $\gamma(t,0) = \gamma_0(t)$. If $\nabla\phi$ is Lipschitz continuous, then there exists a unique solution for the PDE system. An equilibrium of the PDE system satisfies:

$$\begin{cases}-\varepsilon\lambda^* + \phi(\gamma^*) = 0, \\ \ddot{\gamma}^* - ((1-\alpha\varepsilon)\lambda^* + \alpha\phi(\gamma^*))\nabla\phi(\gamma^*) = 0.\end{cases} \tag{3.2}$$

The goal is to show that, as $s \to +\infty$, the PDE system converges to the optimal solution $\lambda^*(t)$ and $\gamma^*(t)$ governed by (3.2). Additionally, as $\epsilon \to 0$, the curve $\gamma(t)$ approaches a geodesic.

3.2. Equilibrium curves.

LEMMA 2. *If γ^*, λ^* is an equilibrium solution to the system,*

$$\begin{cases}\lambda' = -\varepsilon\lambda + \phi(\gamma), \\ \gamma' = \ddot{\gamma} - ((1-\alpha\varepsilon)\lambda + \alpha\phi(\gamma))\nabla\phi(\gamma),\end{cases}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

then γ^* approximates a geodesic on Ω when ε is small. Specifically, as $\varepsilon \to 0$, both $\phi(\gamma(t)) \to 0$ and $\ddot{\gamma}^* \cdot \dot{\gamma}(t) \to 0$.

Proof. Assume γ^* and λ^* is an equilibrium solution to the above system. Equations 3.2 imply that

$$\varepsilon\lambda^* = \phi(\gamma^*).$$

That is, $\phi(\gamma^*) \to 0$, as $\varepsilon \to 0$, which means that $\gamma^* \to \Gamma(p, q, \Omega)$ for all $t \in [0, 1]$. Also, taking the derivative gives us

$$\varepsilon\dot{\lambda}^* = \dot{\gamma}^* \cdot \nabla\phi(\gamma^*). \tag{3.3}$$

Therefore, $\dot{\gamma}^* \cdot \nabla\phi(\gamma^*) \to 0$ as $\varepsilon \to 0$, so $\nabla\phi(\gamma^*)$ approximates the normal direction to the surface.

Further, 3.2 also implies $\ddot{\gamma}^* = ((1-\alpha\varepsilon)\lambda^* + \alpha\phi(\gamma^*))\nabla\phi(\gamma^*)$. Hence, $\ddot{\gamma}^*$ is parallel to $\nabla\phi(\gamma^*)$, an approximation of the normal direction to the surface. Thus, according to the definition 1.1 of a geodesic, γ^* approximates a geodesic when ε is small. ∎

It can also be shown that γ is a good approximation to the shortest geodesic from p to q on Ω.

LEMMA 3. *If γ^*, λ^* is an equilibrium solution to the system*

$$\begin{cases} \lambda' = -\varepsilon\lambda + \phi(\gamma), \\ \gamma' = \ddot{\gamma} - \nabla\phi(\gamma)((1-\alpha\varepsilon)\lambda + \alpha\phi(\gamma)), \end{cases}$$

then, when ε is small, γ^* approximates a minimal geodesic for the surface represented by $\{x| \quad \phi(x) = 0\}$. Specifically, γ^* minimizes $\int_0^1 |\dot{\gamma}^*(t)|^2\, dt$ under the constraint $\phi(\gamma(t)) = \varepsilon\lambda^*(t)$ (and as $\varepsilon \to 0$, both $\phi(\gamma(t)) \to 0$ and $\ddot{\gamma}^* \cdot \dot{\gamma}(t) \to 0$.)

Proof. It suffices to show that $\gamma^*(t)$ is the solution to the constrained minimization:

$$min \int_0^1 |\dot{\gamma}(t)|^2 dt \quad s.t. \quad \phi(\gamma(t)) = \epsilon\lambda^*(t), \quad t \in [0, 1].$$

To do so, we let $\delta \in \mathbb{R}$ be a small parameter, and define

$$J(\delta) = \left\{ \int_0^1 |\dot{\gamma}^*(t) + \delta\dot{a}(t)|^2\, dt, \quad \phi(\gamma^*(t) + \delta a(t)) = \varepsilon\lambda^*(t) \quad \forall t \in [0, 1] \right\},$$

where $a \in C^1[0, 1]$ is an arbitrary perturbation that satisfies $a(0) = a(1) = 0$. Observe that

$$\frac{d}{d\delta} \int_0^1 |\dot{\gamma}^*(t) + \delta\dot{a}(t)|^2\, dt = \int_0^1 2(\dot{\gamma}^*(t) + \delta\dot{a}(t)) \cdot \dot{a}(t)\, dt.$$

Thus,

$$\begin{aligned} \lim_{\delta \to 0} \frac{d}{d\delta} \int_0^1 |\dot{\gamma}^*(t) + \delta\dot{a}(t)|^2\, dt &= \int_0^1 2\dot{a}(t) \cdot \dot{\gamma}^*(t)\, dt \\ &= -2 \int_0^1 a(t) \cdot \ddot{\gamma}^*(t) dt. \end{aligned}$$

This relies on the fact that $a(0) = a(1) = 0$. On the other hand, the constraint $\phi(\gamma^*(t) + \delta a(t)) = \varepsilon\lambda^*(t)$ for any δ yields

$$\nabla\phi(\gamma^*(t) + \delta a(t)) \cdot a(t) = 0.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Since $\ddot{\gamma}^* = ((1-\alpha\varepsilon)\lambda^* + \alpha\phi(\gamma^*))\nabla\phi(\gamma^*(t))$, it follows that

$$\ddot{\gamma}^*(t)\cdot a(t) = 0.$$

Consequently, we have

$$\lim_{\delta\to 0}\frac{d}{d\delta}\int_0^1 |\dot{\gamma}^*(t)+\delta\dot{a}(t)|^2\,dt = -2\int_0^1 a(t)\cdot\ddot{\gamma}^*(t)dt = 0.$$

Therefore, $\delta = 0$ is a critical point for $J(\delta)$. Now, taking the second derivative,

$$\frac{d^2}{d\delta^2}\int_0^1 |\dot{\gamma}^*(t)+\delta\dot{a}(t)|^2\,dt = \int_0^1 2|\dot{a}(t)|^2 dt \geq 0.$$

Therefore, $J(\delta)$ is convex, so $\delta = 0$ is a global minimizer of $J(\delta)$. Thus, γ^* minimizes $\int_0^1 |\dot{\gamma}^*(t)|^2\,dt$ under the constraint $\phi(\gamma(t)) = \varepsilon\lambda^*(t)$, and when ε is small, γ^* approximates one of the shortest geodesics from p to q. ∎

3.3. Convergence analysis. In this section, we present the main theoretical result of this work, providing the convergence analysis in continuous-time PDE systems. Let $U^* = (\lambda^*(t), \gamma^*(t))$ be the steady solution, then

$$\begin{cases} -\varepsilon\lambda^*(t) + \phi(\gamma^*(t)) = 0, \\ \ddot{\gamma}^*(t) - [(1-\alpha\varepsilon)\lambda^*(t) + \alpha\phi(\gamma^*(t))]\nabla\phi(\gamma^*(t)) = 0. \end{cases} \tag{3.4}$$

It is reasonable to consider the domain in which ϕ is suitably small. More precisely, we make the following assumption.

Assumption A. There exists a constant $\nu > 0$ and a constant $a > 0$ such that

$$|\nabla\phi(x)|^2 \geq \nu \quad \text{and} \quad 2a\|D^2\phi(x)\| \leq \nu$$

for $x \in \Sigma$, where

$$\Sigma = \{x \mid |\phi(x)| \leq a\}.$$

THEOREM 4. *Consider the PDE system*

$$\begin{cases} \lambda'(t,s) = -\varepsilon\lambda(t,s) + \phi(\gamma(t,s)), \\ \gamma'(t,s) = \ddot{\gamma}(t,s) - [(1-\alpha\varepsilon)\lambda(t,s) + \alpha\phi(\gamma(t,s))]\nabla\phi(\gamma(t,s)), \end{cases}$$

with initial condition $\gamma_0(t)$ satisfying $\phi(\gamma_0(t)) = 0$ for $t = 0$ and $t = 1$. Suppose Assumption A holds. Then, for appropriate choices of ϵ and α, satisfying $1/2 < \alpha\epsilon < 1$, the system converges exponentially to the equilibrium curve governed by (3.4).

The proof presented below comprises two steps:

Step 1: We introduce a Lyapunov function defined for $U = (\lambda, \gamma)$ as

$$\begin{aligned} J(U) := &\int_0^1 (-\varepsilon\lambda + \phi(\gamma(t,s))^2\,dt \\ &+ \mu\int_0^1 |\ddot{\gamma}(t,s) - [(1-\alpha\varepsilon)\lambda(t,s) + \alpha\phi(\gamma(t,s))]\nabla\phi(\gamma(t,s))|^2\,dt, \end{aligned}$$

where the parameter μ is to be determined so that J along the PDE dynamics is nonincreasing in s, at least for large s. Thus, all solutions will lie strictly in the sublevel set defined by

$$Z = \{U, \quad J(U) \leq J(U_0) < \infty\}.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Step 2: We fix μ and show that there is a $\beta > 0$ such that the decay rate of J is

$$\frac{d}{ds}J \leq -\beta J.$$

After these two essential steps, we will be able to prove the time-asymptotic convergence result expressed as

$$J(U) \sim O(1)e^{-\beta s}$$

as s becomes large.

Proof. When evaluating J along the PDE system, it can be rewritten as

$$J = \int_0^1 \lambda'^2\, dt + \mu \int_0^1 |\gamma'|^2\, dt,$$

with $\mu > 0$ to be determined. Taking the time-derivative of J, we have

$$\begin{aligned}
\frac{d}{ds}J &= 2\int_0^1 \lambda'\lambda''\, dt + 2\mu \int_0^1 \gamma' \cdot \gamma''\, dt \\
&= 2\int_0^1 \lambda'(-\varepsilon\lambda' + \nabla\phi(\gamma)\cdot\gamma')\, dt \\
&\quad + 2\mu \int_0^1 \gamma' \cdot (\ddot{\gamma}' - [(1-\alpha\varepsilon)\lambda' + \alpha\nabla\phi(\gamma)\cdot\gamma']\nabla\phi(\gamma) \\
&\quad - [(1-\alpha\varepsilon)\lambda + \alpha\phi(\gamma)]D^2\phi(\gamma)\gamma')\, dt \\
&= 2\int_0^1 -\varepsilon\lambda'^2 + \lambda'\nabla\phi(\gamma)\cdot\gamma'\, dt \\
&\quad + 2\mu \int_0^1 \gamma' \cdot \ddot{\gamma}' - [(1-\alpha\varepsilon)\lambda' + \alpha\nabla\phi(\gamma)\cdot\gamma']\nabla\phi(\gamma)\cdot\gamma' \\
&\quad - [(1-\alpha\varepsilon)\lambda + \alpha\phi(\gamma)]\gamma' \cdot D^2\phi(\gamma)\gamma'\, dt \\
&= 2\int_0^1 -\varepsilon\lambda'^2 + \lambda'\nabla\phi(\gamma)\cdot\gamma'\, dt + 2\mu \int_0^1 \gamma' \cdot \ddot{\gamma}' - \alpha|\nabla\phi(\gamma)|^2|\gamma'|^2\, dt \\
&\quad + 2\mu \int_0^1 -(1-\alpha\varepsilon)\lambda'\nabla\phi(\gamma)\cdot\gamma' - [(1-\alpha\varepsilon)\lambda + \alpha\phi(\gamma)]\gamma' \cdot D^2\phi(\gamma)\gamma'\, dt \\
&= 2\int_0^1 -\varepsilon\lambda'^2 - \mu\alpha|\nabla\phi(\gamma)|^2|\gamma'|^2\, dt - 2\mu\int_0^1 |\dot{\gamma}'|\, dt + 2\mu\gamma'\cdot\dot{\gamma}'|_0^1 \\
&\quad + 2\int_0^1 (1-\mu(1-\alpha\varepsilon))\lambda'\nabla\phi(\gamma)\cdot\gamma' - \mu(\lambda + \alpha\lambda')\gamma' \cdot D^2\phi(\gamma)\gamma'\, dt.
\end{aligned}$$

Notice that $\gamma(0,s) = p$ and $\gamma(1,s) = q$ for all s, so γ' is zero at $t = 0$ and $t = 1$. Therefore, $\mu\gamma'\cdot\dot{\gamma}'|_0^1 = 0$. Thus,

$$\begin{aligned}
\frac{d}{ds}J &= 2\int_0^1 -\varepsilon\lambda'^2 - \mu\alpha|\nabla\phi(\gamma)|^2|\gamma'|^2\, dt - 2\mu\int_0^1 |\dot{\gamma}'|\, dt \\
&\quad + 2\int_0^1 (1-\mu(1-\alpha\varepsilon))\lambda'\nabla\phi(\gamma)\cdot\gamma' - \mu(\lambda + \alpha\lambda')\gamma' \cdot D^2\phi(\gamma)\gamma'\, dt.
\end{aligned}$$

Therefore, if $0 < \alpha\varepsilon < 1$, then we can choose $\mu = \frac{1}{(1-\alpha\varepsilon)}$. Then,

$$\begin{aligned}
\frac{d}{ds}J &= 2\int_0^1 -\varepsilon\lambda'^2 - \mu\alpha|\nabla\phi(\gamma)|^2|\gamma'|^2\, dt - 2\mu\int_0^1 |\dot{\gamma}'| + (\lambda + \alpha\lambda')\gamma' \cdot D^2\phi(\gamma)\gamma'\, dt \\
&\leq -2\varepsilon\int_0^1 \lambda'^2 dt - 2\mu\alpha\int_0^1 |\nabla\phi|^2|\gamma'|^2\, dt + 2\mu\int_0^1 |\lambda + \alpha\lambda'|\|D^2\phi(\gamma)\||\gamma'|^2 dt.
\end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

We now need to bound the last term. Observe that

$$\lambda(t,s) = \lambda(t,0)e^{-\varepsilon s} + \int_0^s \phi(\gamma)e^{\varepsilon\tau}\, d\tau \cdot e^{-\varepsilon s}.$$

This gives

$$\begin{aligned}|\lambda(t,s)| &\le |\lambda(t,0)|e^{-\varepsilon s} + \max_{[0,s]} |\phi(\gamma)|e^{-\varepsilon s}\frac{1}{\varepsilon}(e^{\varepsilon s}-1)\\ &\le |\lambda(t,0)|e^{-\varepsilon s} + \frac{\max_{[0,s]}|\phi(\gamma)|}{\varepsilon}.\end{aligned}$$

Therefore, using $\epsilon\alpha < 1$,

$$\begin{aligned}|\lambda + \alpha\lambda'| &= |(1-\alpha\varepsilon)\lambda + \alpha\phi(\gamma)|\\ &\le (1-\alpha\varepsilon)|\lambda(t,0)|e^{-\varepsilon s} + \frac{1-\alpha\varepsilon}{\varepsilon}\max_{[0,s]}|\phi(\gamma)| + \alpha|\phi(\gamma)|\\ &= (1-\alpha\varepsilon)|\lambda(t,0)|e^{-\varepsilon s} + \frac{\max_{[0,s]}|\phi(\gamma)|}{\varepsilon}.\end{aligned}$$

Hence, we have

$$\begin{aligned}|\lambda + \alpha\lambda'|\|D^2\phi(\gamma)\| &\le (1-\alpha\varepsilon)|\lambda(t,0)|e^{-\varepsilon s}\|D^2\phi(\gamma)\| + \frac{\max_{[0,s]}|\phi(\gamma)|}{\varepsilon}\|D^2\phi(\gamma)\|\\ &\le (1-\alpha\varepsilon)|\lambda(t,0)|e^{-\varepsilon s}\|D^2\phi(\gamma)\| + \frac{a}{\varepsilon}\|D^2\phi(\gamma)\|.\end{aligned}$$

Thus, the last term can be bounded

$$\begin{aligned}2\mu\int_0^1 |\lambda+\alpha\lambda'|\|D^2\phi(\gamma)\||\gamma'|^2\, dt &\le 2\mu(1-\alpha\varepsilon)\max_t|\lambda(t,0)|e^{-\varepsilon s}\max_t\|D^2\phi(\gamma)\|\int_0^1|\gamma'|^2 dt\\ &\quad + \frac{2a\mu}{\varepsilon}\max_t\|D^2\phi(\gamma)\|\int_0^1|\gamma'|^2\, dt\\ &\le \left(\delta + \frac{2a\mu}{\varepsilon}\max_t\|D^2\phi(\gamma)\|\right)\int_0^1|\gamma'|^2 dt,\end{aligned}$$

where we have used the fact that $|O(1)e^{-\varepsilon s}| \le \delta$ for $s > S$ when S is large enough. Thus,

$$\begin{aligned}\frac{d}{ds}J &\le -2\varepsilon\int_0^1 \lambda'^2 dt - 2\mu\alpha\int_0^1|\nabla\phi(\gamma)|^2|\gamma'|^2\, dt + 2\mu\int_0^1|\lambda+\alpha\lambda'||D^2\phi(\gamma)||\gamma'|^2 dt\\ &\le -2\varepsilon\int_0^1\lambda'^2 - \left(2\mu\alpha\nu - \delta - \frac{\mu\nu}{\varepsilon}\right)\int_0^1|\gamma'|^2\, dt.\end{aligned}$$

Note that we can choose ε and α so that $\varepsilon\alpha > 1/2$, and take S suitably large so that

$$\delta \le \delta_0 := (\alpha\varepsilon - 1/2)\frac{\mu\nu}{\varepsilon} = \frac{\nu(\alpha\varepsilon - 1/2)}{\varepsilon(1-\alpha\varepsilon)}.$$

Hence

$$J' \le -\beta J,$$

where $\beta = \min\{2\varepsilon, \delta_0\}$. Thus,

$$J(U(s)) \le J(U(S))e^{-\beta(s-S)}, \quad \forall s > S. \qquad \square$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Remark 5. Assumption A may appear unconventional, it provides a reasonable constraint that keeps γ close to the underlying surface. For instance with $\phi(x) = \frac{1}{2}(|x|^2 - 1)$, we have

$$\nabla\phi(x) = x, \quad D^2\phi(x) = I.$$

To satisfy assumption A in the domain $\Sigma = \{x||\phi(x)| \le a\} = \{x| \quad 1 - 2a \le |x|^2 \le 1 + 2a\}$, it suffices to set $\nu = 1 - 2a$ with a satisfying $2a \le \nu$. This implies $a \le 1/4$ and $\nu \ge 1/2$. We can also consider $\phi(x)$ as a signed distance function, which, in this case, takes the form:

$$\phi(x) = 1 - |x|.$$

In the domain $\Sigma = \{x| \, |\phi(x)| \le a\} = \{x| \quad 1 - a \le |x| \le 1 + a\}$, both

$$\nabla\phi(x) = -\frac{x}{|x|}, \quad \text{and} \quad D^2\phi(x) = \frac{1}{|x|^3}\left[x \otimes x - |x|^2 I\right]$$

satisfy Assumption A with $\nu = 1$, provided $\frac{2a}{1-a} \le 1$, which requires $0 < a \le \frac{1}{3}$. Note that for the signed distance function, the condition $|\nabla\phi| = 1$ always holds. This property also explains why the signed distance function is highly effective in various applications.

3.4. Discrete system. Note that the update described in (2.2) can be reformulated as

$$\begin{aligned}
\frac{\lambda_{k+1} - \lambda_k}{\tau_\lambda} &= -\frac{\varepsilon}{1 + \varepsilon\tau_\lambda}\lambda_k + \frac{1}{1 + \varepsilon\tau_\lambda}\phi(\gamma_k) \\
\frac{\gamma_{k+1} - \gamma_k}{\tau_\gamma} &= \ddot{\gamma}_k(t) - \nabla\phi(\gamma_k) \cdot \left[\left(1 - \frac{1 + w}{1 + \varepsilon\tau_\lambda}\varepsilon\tau_\lambda\right)\lambda_k + \frac{(1 + w)\tau_\lambda}{1 + \varepsilon\tau_\lambda}\phi(\gamma_k)\right],
\end{aligned}$$

for which the high-resolution PDE system (3.1) serves as a more accurate time-continuous counterpart. By some standard truncation error analysis we can conclude the following:

THEOREM 6. *Denote $U_k(t) = (\lambda_k(t), \gamma_k(t))$ as the semidiscrete update defined in* (2.2)*, and $U(s,t) = (\lambda(s,t), \gamma(s,t))$ as the solution provided in Theorem* 4 *with initial data $U(0,t) = U_0(t)$. Then for $\tau = \tau_\gamma = \frac{\tau_\lambda}{1+\varepsilon\tau_\lambda}$, we have:*

$$\lim_{\tau\to 0} \sup \max_{0 \le k \le S/\tau} \|U_k(t) - U(k\tau, t)\| = 0$$

for any fixed $S > 0$.

The key question is: can we relate convergence properties of high-resolution PDEs to those of the corresponding discrete update scheme. The answer is yes, as Theorem 6 holds true for $S = \infty$ under the structural assumptions on ϕ. A full proof would involve carefully estimating a discrete Lyapunov functional such as

$$J_k = \int_0^1 (\lambda_{k+1} - \lambda_k)^2 dt + \mu \int_0^1 (\gamma_{k+1} - \gamma_k)^2 dt.$$

Rather than going deeper with technical estimates, we now turn our attention to how the scheme behaves under discretization in the parameter t, and how it performs numerically. This also brings up a subtle issue: in cases where the geodesic between states is not unique, what does convergence actually mean – and how should the scheme behaves.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

3.5. Discussion on the case with nonunique geodesics. When the geodesic curve between points p and q on the level set surface is not unique, the initialization of γ can significantly influence the algorithm's convergence. In such cases, we introduce randomness into the initialization of γ. Our numerical experiments indicate that this randomized initialization improves the likelihood of convergence to a valid geodesic curve. Intuitively, this may be because the algorithm tends to closest to the initial curve. Therefore, the initial choice of γ plays a key role in determining which geodesic the algorithm ultimately finds in nonunique scenarios.

4. Numerical results.

4.1. Initialization. There are several ways to initialize the iteration algorithm. One straightforward approach is to set γ as a straight line connecting p to q, with λ initialized to 0. This provides a simple initial guess

$$\begin{cases} \gamma_0(t) = p(1-t) + qt & \text{such that } \gamma_0(0) = p, \gamma_0(1) = q, \\ \lambda_0(t) = 0. \end{cases}$$

When the geodesic curve is not unique, for instance, when $\gamma(0) = p$ and $\gamma(1) = q$ are antipodal points on the surface of a sphere, multiple geodesic curves exist between p and q. To address this, we introduce randomness into the initialization of the curve γ by selecting a random point r on the surface and setting

$$\gamma_0(t) = p(1-t) + \tau_r r(1-t)t + qt,$$

where τ_r is a constant. In our numerical experiments, this initialization method enables the algorithm to converge to a geodesic curve, even when p and q are antipodal. Since determining the uniqueness of a geodesic between points p and q can be challenging, this initialization approach is applicable for any pair of points. Figure 2 shows the resulting curve after 3500 iterations, when γ is initialized to be a straight line (left) and when γ is initialized with some randomness (right). Both results in Figure 2 are generated with the PDHG-inspired algorithm using regularization described by 2.2. We use $\varepsilon = 0.01$ and $\omega = 1$.

4.2. Numerical solution. To compute the solution of the marching scheme, we use finite differences. For some integer $m > 0$, we define $\Delta t = 1/m$. For $i \in \{1, \ldots, m\}$, let $t_i = i\Delta t$ and for each $i \in \{1, \ldots, m-1\}$, we do

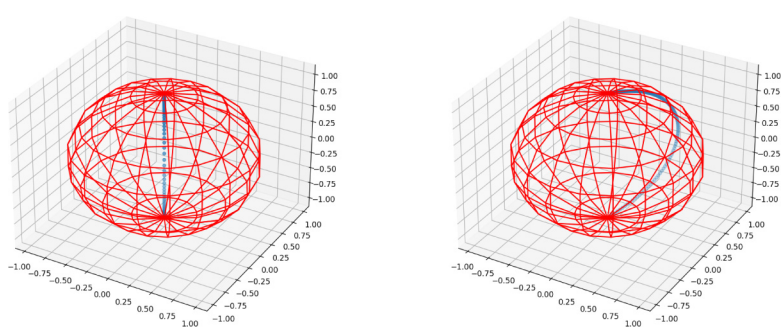

FIG. 2. *Left: Resulting curve γ (in blue) after $3,500$ iterations of the algorithm to find the path between antipodal points p and q, using a straight line as initial guess. Right: Resulting curve after $3,500$ iterations for the same antipodal points p and q, using random initialization.*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

TABLE 1

Average results for algorithm 4.1 *for* 10 *pairs of points,* 100 *time points,* $\tau_\gamma = 0.00001$, $\tau_\lambda = 0.7$, $\varepsilon = 0.01$, *and* $\omega = 1$.

	Avg Absolute Error	Avg Relative Error	Avg Comp. Time
100 iterations	0.169	9.08 %	0.376 s
1000 iterations	0.107	5.719 %	4.175 s
2000 iterations	0.057	2.978 %	8.279 s

$$
\begin{cases}
\lambda_{k+1}(t_i) = \frac{1}{1+\varepsilon\tau_\lambda}\left(\lambda_k(t_i) + \tau_\lambda \phi(\gamma_k(t_i))\right), \\
\tilde{\lambda}_{k+1}(t_i) = \lambda_{k+1}(t_i) + \omega(\lambda_{k+1}(t_i) - \lambda_k(t_i)), \\
\gamma_{k+1}(t_i) = \gamma_k(t_i) - \tau_\gamma \left(-\frac{\gamma_k(t_{i+1}) - 2\gamma_k(t_i) + \gamma_k(t_{i-1})}{\Delta t^2} + \tilde{\lambda}_{k+1}(t_i)\nabla\phi(\gamma_k(t_i))\right), \\
\gamma_k(t_0) = p, \quad \gamma(t_m) = q.
\end{cases}
\tag{4.1}
$$

To evaluate the results, two error metrics were used. The first, termed "absolute error", is defined as $||\gamma| - d|$, where d represents the true geodesic distance. The second metric, referred to as "surface error", is given by

$$|\lambda^T \phi(\gamma)|,$$

which reflects the extent to which the path γ deviates from the surface. Surface error is a useful metric when the true geodesic distance can not be calculated.

Table 1 shows the average absolute error, average relative error, and average computation time for the algorithm 4.1 for 10 pairs of points on the unit sphere. Here, $\tau_\gamma = 0.00001$, $\tau_\lambda = 0.7$, $\varepsilon = 0.01$, and $\omega = 1$ were used.

4.3. Effect of regularization coefficient ε. The choice of ϵ significantly affects both numerical stability and the quality of the computed geodesic. We tested a range of values (i.e., $\epsilon = 10^{-4}, \ldots 1$), as shown in Figure 3. Small ε leads to instability, while large ε increases absolute error. Regularization is introduced to make sure the ascent step is well-defined. Step size choices alone are not enough to prevent λ from drifting, and this can be seen in Figures 4 and 5. Figure 4 shows that, without regularization, all tested values of τ_λ lead to instability, as reflected in both absolute and surface error.

Figure 5 shows absolute error and surface error for various τ_λ values with regularization ($\varepsilon = 0.01$). The results demonstrate that, with regularization, an appropriate choice of τ_λ yields stable and accurate behavior—unlike the unregularized case in Figure 4.

4.4. Examples. The algorithm 4.1 was used to approximate geodesic distances between points on several surfaces represented in $\mathbb{R}^3$. Figure 6 shows the absolute error and surface error with respect to iterations of the algorithm when approximating the geodesic distance between antipodal points $p = [0, 0, 1]$ and $q = [0, 0, -1]$. The final path obtained after 5000 iterations is plotted on a mesh sphere. For this example, the final absolute error was 0.265 and the final relative error was 8.445%.

Figure 7 shows the surface error against iterations of the algorithm 4.1 when approximating the geodesic distance between two points on a torus. The approximated path after 5000 iterations is plotted on a mesh torus.

4.5. Variation of the base algorithm. Note that the update in (2.2) represents a discretization of the continuous PDE system. Alternatively, we can employ different discretizations of the PDE system to create various updates. For instance, we update λ one more time before updating γ, resulting in the following

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

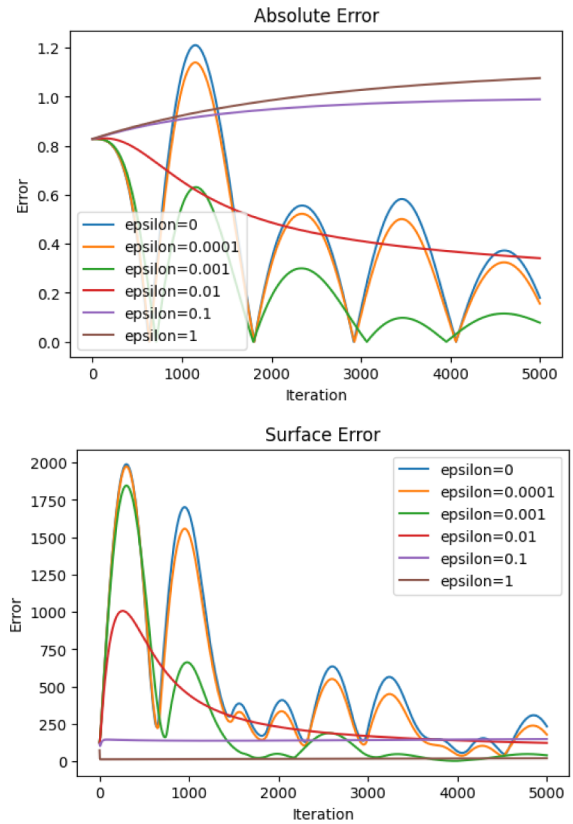

FIG. 3. *Plot of absolute error (left) and surface error (right) over iterations of the algorithm for finding a path γ between antipodal points on a sphere. Each colored curve represents results for a different value of the regularization coefficient ε.*

$$\begin{cases} \lambda_{k+1}(t) = \dfrac{1}{1+\varepsilon\tau_\lambda}(\lambda_k(t) + \tau_\lambda \phi(\gamma_k(t))), \\ \tilde{\lambda}_{k+1}(t) = \lambda_{k+1}(t) + \omega(\lambda_{k+1}(t) - \lambda_k(t)), \\ \bar{\lambda}_{k+1}(t) = \dfrac{1}{1+\varepsilon\tau_\lambda}(\tilde{\lambda}_{k+1}(t) + \tau_\lambda \phi(\gamma_k(t))), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma(-\ddot{\gamma}_k(t) + \tilde{\lambda}_{k+1}(t)\nabla\phi(\gamma_k(t))). \end{cases} \tag{4.2}$$

In Figure 8, the plot on the left shows the absolute error against iterations of the algorithm 4.2 when used to approximate distance between antipodal points on the sphere, and the final path γ is shown on the right.

Figure 9 shows the surface error over time when a variation 4.2 of the base algorithm is used to approximate geodesic distance between points on the surface of the "Stanford bunny." For this surface, the signed distance function ϕ was approximated for a point x by computing

$$\phi(x) = \min\{\|p - x\| \,:\, p \in P \subset \Omega\},$$

where $P \subset \Omega$ is a set of points on the surface. The approximated path after 100 iterations is plotted on a reconstruction of the surface of the Stanford bunny.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

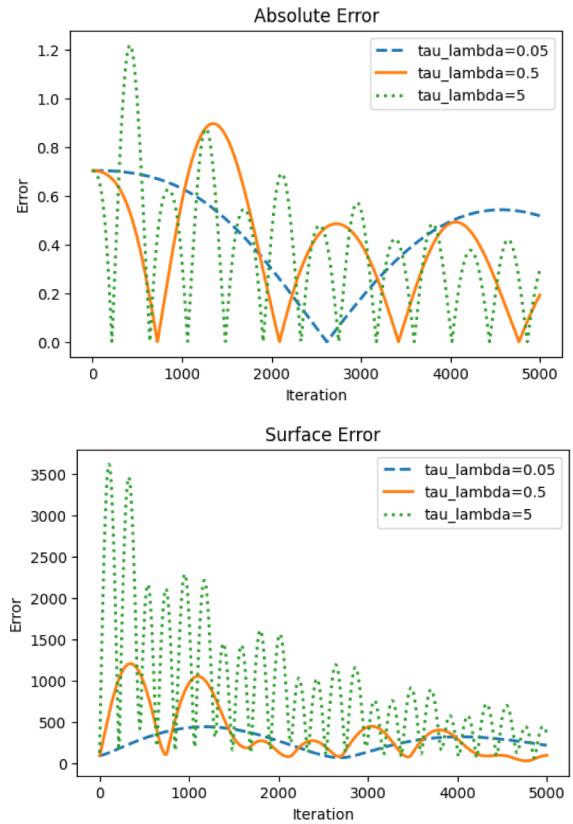

FIG. 4. *No regularization. Absolute error (left) and surface error (right) over iterations of the algorithm for finding a path γ between antipodal points on the sphere. Different colors indicate different values of τ_λ.*

Another algorithm, also grounded in the PDE formulation, showed promising performance:

$$\begin{cases} \lambda_{k+1}(t) = \frac{1}{1+\varepsilon\tau_\lambda}(\lambda_k(t) + \tau_\lambda \phi(\gamma_k(t))), \\ \tilde{\lambda}_{k+1}(t) = (1-\alpha\varepsilon)\lambda_{k+1}(t) + \alpha\phi(\gamma_k(t)), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma(-\ddot{\gamma}_k(t) + \tilde{\lambda}_{k+1}(t)\nabla\phi(\gamma_k(t))). \end{cases} \tag{4.3}$$

This scheme is a direct Euler discretization of the PDE system, where λ' and γ' are replaced by finite differences with time step $\frac{\tau_\lambda}{1+\varepsilon\tau_\lambda}$ and τ_γ, respectively. This is in sharp contrast to scheme (2.2), where

$$\tilde{\lambda}_{k+1}(t) = \lambda_{k+1}(t) + \omega(\lambda_{k+1}(t) - \lambda_k(t)).$$

In algorithm 4.3, we can rewrite $\tilde{\lambda}_{k+1}$ in terms of ω instead of α based on the calculations in section 3.1, where $\alpha = (1+\omega)\frac{\tau_\lambda}{1+\varepsilon\tau_\lambda}$,

$$\tilde{\lambda}_{k+1} = \lambda_{k+1} + \frac{\alpha}{\tau_\lambda}(\lambda_{k+1} - \lambda_k) = \lambda_{k+1} + \frac{(1+\omega)}{1+\varepsilon\tau_\lambda}(\lambda_{k+1} - \lambda_k).$$

In Figure 10, the plot on the left shows the absolute error against iterations of the algorithm 4.3 when used to approximate distance between antipodal points on the sphere, and the final path γ is shown on the right.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

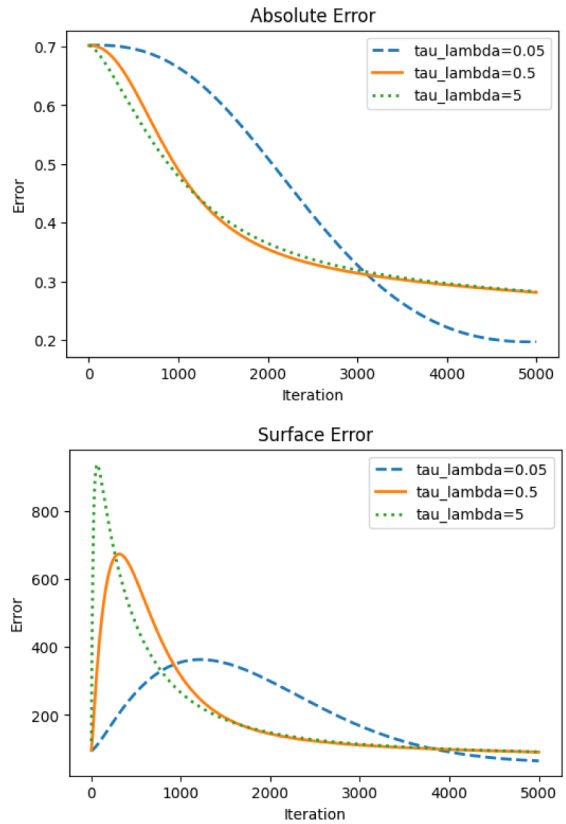

FIG. 5. *With regularization ($\varepsilon = 0.01$). Absolute error (left) and surface error (right) vs. iterations for computing a path γ between antipodal points on the sphere. Different colors indicate different values of τ_λ.*

5. Convergence rate for the planar surface case. Consider the case where $\phi(\gamma) = a \cdot \gamma$, with $a \in \mathbb{R}^3$. Then the variational derivatives of the Lagrangian are

$$\frac{\delta}{\delta\gamma}\mathcal{L}_\varepsilon[\gamma,\lambda] = -\ddot{\gamma} + \lambda a, \quad \frac{\delta}{\delta\lambda}\mathcal{L}_\varepsilon[\gamma,\lambda] = a \cdot \gamma - \varepsilon\lambda.$$

The PDHG scheme becomes

$$\begin{cases} \lambda_{k+1} &= \lambda_k + \tau_\lambda(a \cdot \gamma_k - \varepsilon\lambda_{k+1}), \\ \tilde{\lambda}_{k+1} &= \lambda_{k+1} + \omega(\lambda_{k+1} - \lambda_k), \\ \gamma_{k+1} &= \gamma_k - \tau_\gamma(-\ddot{\gamma}_{k+1} + \tilde{\lambda}_{k+1} a). \end{cases}$$

Setting $\omega = 1$, the update for γ simplifies to

$$(1 - \tau_\gamma \partial^2)\gamma_{k+1} = \gamma_k - \tau_\gamma(2\lambda_{k+1} - \lambda_k)a.$$

Solving the 2nd order equation with zero boundary conditions yields

$$\begin{aligned} \gamma_{k+1} &= (1 - \tau_\gamma \partial^2)^{-1}(\gamma_k - \tau_\gamma(2\lambda_{k+1} - \lambda_k)a) \\ &= \int_0^1 G(t,s)\,(\gamma_k(s) - \tau_\gamma(2\lambda_{k+1}(s) - \lambda_k(s))a)\,ds, \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

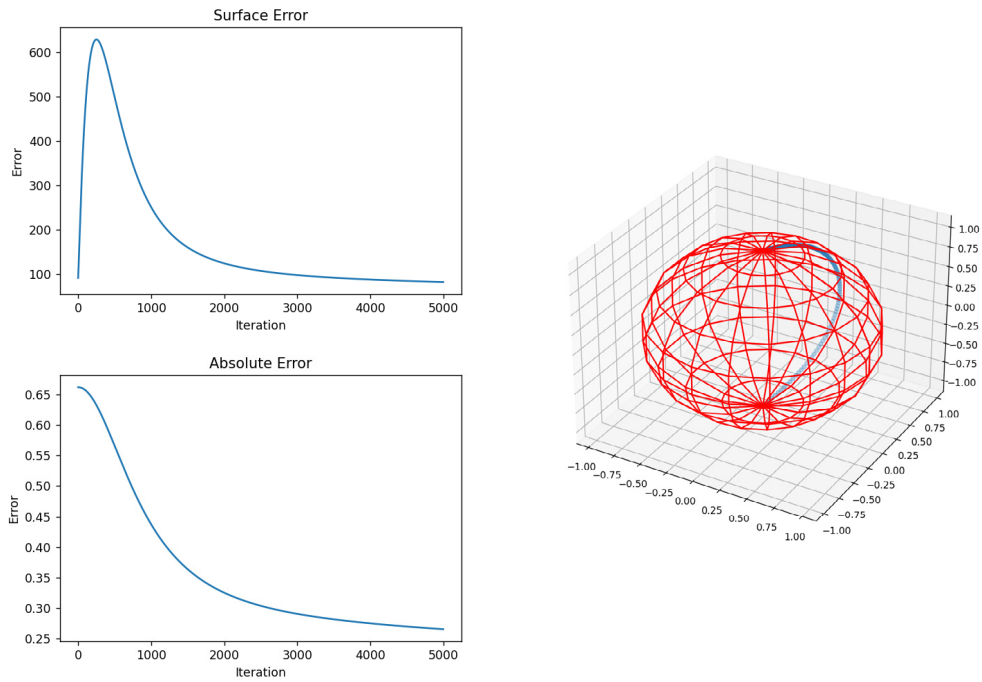

FIG. 6. *The plots on the left shows the absolute error and surface error with respect to number of iterations of the algorithm* 4.1, *where* $\varepsilon = 0.01$ *and* $\omega = 1$. *On the right, the approximated path* γ *after* 5,000 *iterations is plotted on a mesh unit sphere. For this example, the final absolute error was* 0.265 *and the final relative error was* 8.445%.

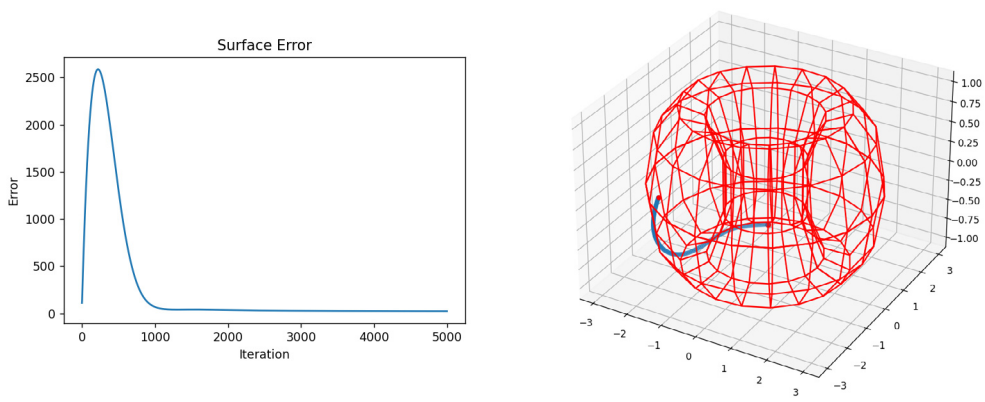

FIG. 7. *The plot on the left shows the surface error with respect to number of iterations of algorithm* 4.1 *when used to approximate the geodesic distance between two points on a torus. On the right, the approximated path gamma after* 5,000 *iterations is plotted on a mesh torus.*

where G is the Green function defined by

$$G(t,s) = \frac{\sqrt{\tau_\gamma}}{\sinh(1/\sqrt{\tau_\gamma})} \begin{cases} \sinh((1-s)/\sqrt{\tau_\gamma})\sinh(t/\sqrt{\tau_\gamma}), & 0 \le t \le s, \\ \sinh(s/\sqrt{\tau_\gamma})\sinh((1-t)/\sqrt{\tau_\gamma}), & s < t \le 1. \end{cases}$$

We can prove the following result.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

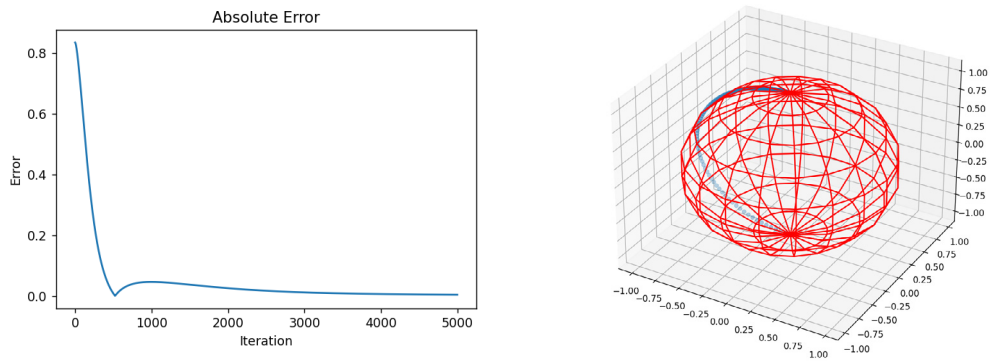

FIG. 8. *The plot on the left shows the absolute error with respect to iterations of the algorithm* 4.2 *described above, where* $\varepsilon = 0.00001$ *and* $\omega = 1000$. *The approximated path* γ *after* 5,000 *iterations is plotted on a mesh sphere on the right. For this example, the final absolute error was* 0.004 *and the final relative error was* 0.118%.

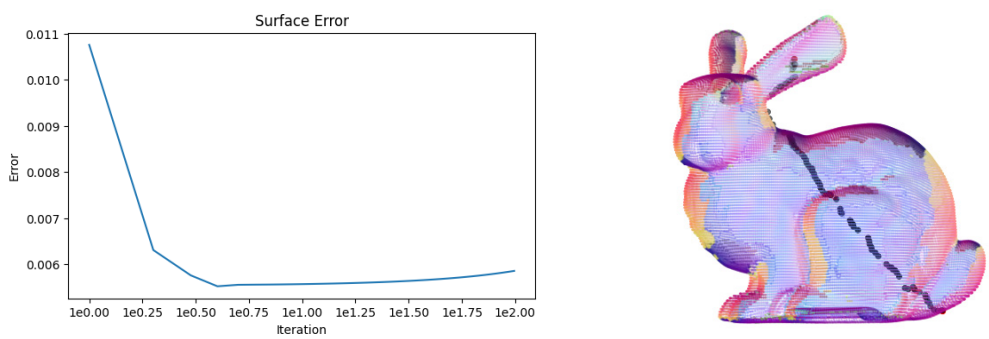

FIG. 9. *The figure shows the surface error and final gamma plot for a path on the surface of the stanford bunny after* 100 *iterations. These plots were created using the update scheme* 4.2 *with* $\varepsilon = 0.0000001$ *and* $\omega = 100000$. *Note that a log scale is used for the* x*-axis for the surface error plot.*

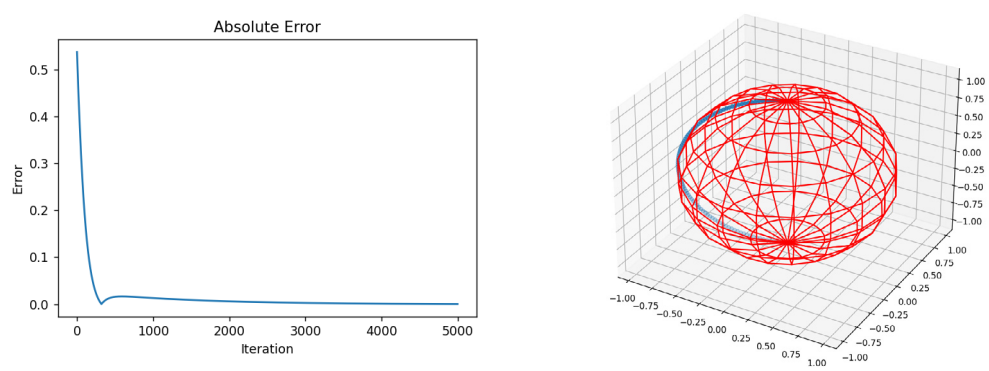

FIG. 10. *The plot on the left shows the absolute error with respect to iterations of the algorithm* 4.3 *described above, where* $\varepsilon = 0.00001$ *and* $\alpha = 1000$. *The approximated path* γ *after* 5,000 *iterations is plotted on a mesh sphere on the right. For this example, the final absolute error was* 0.001, *and the final relative error was* 0.005%.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

THEOREM 1. *Assume $\phi(\gamma) = a \cdot \gamma$ for some $a \in \mathbb{R}^3$, and let λ_k and γ_k be the iterates generated by the scheme*

$$\begin{cases} \lambda_{k+1} &= \lambda_k + \tau_\lambda(a \cdot \gamma_k - \varepsilon\lambda_{k+1}) \\ \gamma_{k+1} &= (1 - \tau_\gamma \partial^2)^{-1}(\gamma_k - \tau_\gamma(2\lambda_{k+1} - \lambda_k)a). \end{cases}$$

Denote the matrix and variables

$$A = \begin{bmatrix} \frac{1}{\tau_\lambda} & a^T \\ a & \frac{1}{\tau_\gamma} I_{3\times 3} \end{bmatrix}, \quad \xi = \begin{bmatrix} \lambda \\ \gamma \end{bmatrix}, \bar{\xi}_k = \begin{bmatrix} \bar{\lambda}_k \\ \bar{\gamma}_k \end{bmatrix} = \frac{1}{k}\sum_{i=1}^{k} \xi_i.$$

Then the system converges for any step sizes $\tau_\lambda, \tau_\gamma$ satisfying

$$\tau_\lambda \tau_\gamma < \frac{1}{|a|^2} = \frac{1}{a^T a},$$

with an ergodic convergence rate of $\mathcal{O}(1/k)$. Specifically, for any $k \geq 1$ and any (γ, λ),

$$\mathcal{L}_\varepsilon(\bar{\gamma}_k, \lambda) - \mathcal{L}_\varepsilon(\gamma, \bar{\lambda}_k) \leq \frac{1}{2k}\|\xi_0 - \xi\|_A^2.$$

Proof. From the update scheme, we have

$$\begin{cases} \dfrac{\lambda_k - \lambda_{k+1}}{\tau_\lambda} + a \cdot (\gamma_k - \gamma_{k+1}) &= \varepsilon\lambda_{k+1} - a \cdot \gamma_{k+1} = -\dfrac{\delta}{\delta\lambda}\mathcal{L}_\varepsilon^{k+1} \\ \dfrac{\gamma_k - \gamma_{k+1}}{\tau_\gamma} + (\lambda_k - \lambda_{k+1})a &= -\ddot{\gamma}_{k+1} + \lambda_{k+1}a = \dfrac{\delta}{\delta\gamma}\mathcal{L}_\varepsilon^{k+1}. \end{cases}$$

Now consider the difference in Lagrangian values:

$$\mathcal{L}_\varepsilon(\gamma_{k+1}, \lambda) - \mathcal{L}_\varepsilon(\gamma, \lambda_{k+1}) = \Delta_1 + \Delta_2 + \Delta_3.$$

Term-by-term analysis gives:

- Kinetic term difference:

$$\begin{aligned} \Delta_1 &= \frac{1}{2}\int_0^1 \left(|\dot{\gamma}_{k+1}|^2 - |\dot{\gamma}|^2\right) dt \\ &= \int_0^1 (\gamma - \gamma_{k+1}) \cdot \ddot{\gamma}_{k+1}\, dt - \frac{1}{2}\int_0^1 |\dot{\gamma}_{k+1} - \dot{\gamma}|^2\, dt, \end{aligned}$$

- By-linear term difference:

$$\begin{aligned} \Delta_2 &= \int_0^1 \lambda a \cdot \gamma_{k+1}\, dt - \int_0^1 \lambda_{k+1} a \cdot \gamma\, dt \\ &= \int_0^1 (\lambda - \lambda_{k+1}) a \cdot \gamma_{k+1}\, dt + \int_0^1 \lambda_{k+1} a \cdot (\gamma_{k+1} - \gamma)\, dt, \end{aligned}$$

- Regularization term difference:

$$\begin{aligned} \Delta_3 &= -\frac{\varepsilon}{2}\int_0^1 \lambda^2\, dt + \frac{\varepsilon}{2}\int_0^1 \lambda_{k+1}^2\, dt \\ &= \int_0^1 (\lambda_{k+1} - \lambda)\varepsilon\lambda_{k+1}\, dt - \frac{\varepsilon}{2}\int_0^1 (\lambda - \lambda_{k+1})^2\, dt. \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Combining all:

$$\begin{aligned}
&\mathcal{L}_\varepsilon(\gamma_{k+1},\lambda) - \mathcal{L}_\varepsilon(\gamma,\lambda_{k+1}) \\
&\quad = \int_0^1 (\gamma-\gamma_{k+1})\cdot(\ddot{\gamma}_{k+1}-\lambda_{k+1}a)\,dt + \int_0^1 (\lambda-\lambda_{k+1})(a\cdot\gamma_{k+1}-\varepsilon\lambda_{k+1})\,dt \\
&\qquad - \frac{1}{2}\int_0^1 |\dot{\gamma}-\dot{\gamma}_{k+1}|^2\,dt - \frac{\varepsilon}{2}\int_0^1 |\lambda-\lambda_{k+1}|^2\,dt.
\end{aligned}$$

Using the update expressions,

$$\begin{aligned}
&\le \int_0^1 (\gamma_{k+1}-\gamma)\cdot\left[\frac{\gamma_k-\gamma_{k+1}}{\tau_\gamma} + a(\lambda_k-\lambda_{k+1})\right]dt \\
&\quad + \int_0^1 (\lambda_{k+1}-\lambda)\left[\frac{\lambda_k-\lambda_{k+1}}{\tau_\lambda} + a\cdot(\gamma_k-\gamma_{k+1})\right]dt.
\end{aligned}$$

Let $A = \left[\begin{smallmatrix} \frac{1}{\tau_\lambda} & a^T \\ a & \frac{1}{\tau_\gamma} I_{3\times 3}\end{smallmatrix}\right]$, and define $\xi = \left[\begin{smallmatrix}\lambda\\ \gamma\end{smallmatrix}\right]$, then we obtain

$$\begin{aligned}
&= \int_0^1 A(\xi_k-\xi_{k+1})\cdot(\xi_{k+1}-\xi)\,dt \\
&= \frac{1}{2}\int_0^1 A(\xi_k-\xi)\cdot(\xi_k-\xi)\,dt - \frac{1}{2}\int_0^1 A(\xi_{k+1}-\xi)\cdot(\xi_{k+1}-\xi)\,dt \\
&\quad - \frac{1}{2}\int_0^1 A(\xi_k-\xi_{k+1})\cdot(\xi_k-\xi_{k+1})\,dt \\
&\le \frac{1}{2}\int_0^1 \|\xi_k-\xi\|_A^2\,dt - \frac{1}{2}\int_0^1 \|\xi_{k+1}-\xi\|_A^2\,dt.
\end{aligned}$$

Hence, the algorithm converges if

$$\tau_\lambda\tau_\gamma < \frac{1}{|a|^2} = \frac{1}{a^T a}.$$

Further, define ergodic averages, $\bar{\lambda}_k = \frac{1}{k}\sum_{i=1}^k \lambda_i$ and $\bar{\gamma}_k = \frac{1}{k}\sum_{i=1}^k \gamma_i$. By Jensen's inequality:

$$\int_0^1 |\dot{\bar{\gamma}}_k|\,dt \le \frac{1}{k}\sum_{i=1}^k \int_0^1 |\dot{\gamma}_i|\,dt.$$

Using the telescoping sum, we conclude

$$\begin{aligned}
\mathcal{L}_\varepsilon(\bar{\gamma}_k,\lambda) - \mathcal{L}_\varepsilon(\gamma,\bar{\lambda}_k) &\le \frac{1}{k}\sum_{i=0}^{k-1}(\mathcal{L}(\gamma_{i+1},\lambda) - \mathcal{L}(\gamma,\lambda_{i+1})) \\
&\le \frac{1}{2k}\|\xi_0-\xi\|_A^2.
\end{aligned}$$

Thus, the method achieves ergodic convergence at rate $O(1/k)$. ∎

6. Discussion and future directions. In this work, we introduce a primal-dual level set method for computing geodesic distances. The underlying surface is represented as the zero level set of a function, transforming the problem into a constrained minimization framework. Using the primal-dual methodology, we incorporate regularization and acceleration techniques to develop an efficient and robust algorithm.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

TABLE 2

Absolute and relative errors after $5,000$ *iterations of the base algorithm and the two variations of the base algorithm when used to approximate geodesic distance between antipodal points on the unit sphere. For all schemes,* $\varepsilon = 0.0001$, $\omega = 1000$, *and* $\alpha = 1000$.

Scheme	Absolute Error	Relative Error
PDHG 2.2	0.175	5.567%
PDHG var 1 4.2	0.004	0.118%
PDHG var 2 4.3	0.0001	0.005%

We analyze the long-term behavior of the high-resolution PDE system assicaited with the method by employing a Lyapunov function. Combined with structural conditions imposed on the level set function, this analysis establishes exponential convergence rates. Furthermore, relating these results to discrete-time settings, we gain further insights into the algorithm's performance. This proposed approach is computationally efficient, straightforward to implement, and offers several alternative extensions. Numerical examples illustrate that the method converges to geodesics as the level of refinement increases.

Despite its strengths, our study has limitations that point to future research directions. The convergence analysis relies on specific assumptions about the level set function. Relaxing this assumption to develop a more general framework would be a valuable extension of our work. Additionally, our current work is limited to static surfaces, whereas many practical applications require handling dynamically moving surfaces. Exploring the adaptability of our framework to moving surfaces will be an important avenue for future work.

We can generalize the current framework to handle a general Lagrangian $L(\gamma, \dot{\gamma})$ by solving the problem:

$$\min_{\Gamma(p,q,\Omega)} \int_0^1 L(\gamma, \dot{\gamma}) dt$$

provided that L is smooth and convex in its second argument. Using the saddle-point formulation

$$\inf_{\gamma} \sup_{\lambda} \left\{ \int_0^1 L(\gamma(t), \dot{\gamma}(t))\, dt + \int_0^1 \lambda(t)\phi(\gamma(t))\, dt \right\},$$

we derive with the same regularization the following iterative updates:

$$\begin{cases} \lambda_{k+1}(t) = \dfrac{1}{1+\varepsilon\tau_\lambda}(\lambda_k(t) + \tau_\lambda \phi(\gamma_k(t))), \\ \tilde{\lambda}_{k+1}(t) = \lambda_{k+1}(t) + \omega(\lambda_{k+1}(t) - \lambda_k(t)), \\ \gamma_{k+1}(t) = \gamma_k(t) - \tau_\gamma \left(L_{\gamma_k} - \dfrac{d}{dt} L_{\dot{\gamma}_k} + \tilde{\lambda}_{k+1}(t) \nabla \phi(\gamma_k(t)) \right), \end{cases}$$

where $\varepsilon > 0$ is a regularization parameter, $\omega \in [0, 1]$ controls relaxation, and $\tau_\lambda, \tau_\gamma$ are step sizes.

REFERENCES

[1] P. V. ARAÚJO, *Regular Surfaces*, in Differential Geometry, P. V. Araújo, ed., Springer International Publishing, Cham, 2024, pp. 31–57, https://doi.org/10.1007/978-3-031-62384-4_2.

[2] M. BENYAMIN, J. CALDER, G. SUNDARAMOORTHI, AND A. YEZZI, *Accelerated variational PDEs for efficient solution of regularized inversion problems*, J. Math. Imaging Vis., 62 (2020), pp. 10–36, https://doi.org/10.1007/s10851-019-00910-2.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

[3] S. Boyd, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Found. Trends Mach. Learn., 3 (2010), pp. 1–122, https://doi.org/10.1561/2200000016.

[4] L. Bungert, J. Calder, and T. Roith, *Uniform convergence rates for Lipschitz learning on graphs*, IMA J. Numer. Anal., 43 (2023), pp. 2445–2495, https://doi.org/10.1093/imanum/drac048.

[5] J. Calder and A. Yezzi, *PDE acceleration: A convergence rate analysis and applications to obstacle problems*, Res. Math. Sci., 6 (2019), 35, https://doi.org/10.1007/s40687-019-0197-x.

[6] A. Chambolle and T. Pock, *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vis., 40 (2011), pp. 120–145, https://doi.org/10.1007/s10851-010-0251-1.

[7] S. Chapman, D. A. Galante, E. Harris, S. U. Sheorey, and D. Vegh, *Complex geodesics in de Sitter space*, J. High Energy Phys., 2023 (2023), 6, https://doi.org/10.1007/JHEP03(2023)006.

[8] L.-T. Cheng, P. Burchard, B. Merriman, and S. Osher, *Motion of curves constrained on surfaces using a level-set approach*, J. Comput. Phys., 175 (2002), pp. 604–644, https://doi.org/10.1006/jcph.2001.6960.

[9] T. Chu, G. L. Miller, and D. Sheehy, *Exact computation of a manifold metric, via lipschitz embeddings and shortest paths on a graph*, in ACM-SIAM Symposium on Discrete Algorithms, 2017.

[10] K. Crane, M. Livesu, E. Puppo, and Y. Qin, *A Survey of Algorithms for Geodesic Paths and Distances*, https://doi.org/10.48550/arXiv.2007.10430, https://arxiv.org/abs/2007.10430, 2020.

[11] K. Crane, C. Weischedel, and M. Wardetzky, *Geodesics in heat: A new approach to computing distance based on heat flow*, ACM Trans. Graph., 32 (2013), pp. 152:1–152:11, https://doi.org/10.1145/2516971.2516977.

[12] R. S. Cruz, L. Lebrat, P. Bourgeat, C. Fookes, J. Fripp, and O. Salvado, *DeepCSR: A 3D Deep Learning Approach for Cortical Surface Reconstruction*, https://doi.org/10.48550/arXiv.2010.11423, https://arxiv.org/abs/2010.11423, 2020.

[13] E. Davis and S. Sethuraman, *Approximating geodesics via random points*, Ann. Appl. Probab., 29 (2019), pp. 1446–1486, https://arxiv.org/abs/1711.06952.

[14] E. Esser, X. Zhang, and T. F. Chan, *A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science*, SIAM J. Imaging Sci., 3 (2010), pp. 1015–1046, https://doi.org/10.1137/09076934X.

[15] T. Goldstein, M. Li, and X. Yuan, *Adaptive primal-dual splitting methods for statistical learning and image processing*, in Advances in Neural Information Processing Systems, 2015, pp. 2089–2097.

[16] T. Goldstein and S. Osher, *The split Bregman method for L1-regularized problems*, SIAM J. Imaging Sci., 2 (2009), pp. 323–343, https://doi.org/10.1137/080725891.

[17] A. Joshi, D. Shattuck, and R. Leahy, *A method for automated cortical surface registration and labeling*, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7359 LNCS, 2012, pp. 180–189, https://doi.org/10.1007/978-3-642-31340-0_19.

[18] J. Kastikainen and S. Shashi, *Structure of holographic BCFT correlators from geodesics*, Phys. Rev. D, 105 (2022), 046007, https://doi.org/10.1103/PhysRevD.105.046007.

[19] R. Kimmel and J. A. Sethian, *Computing geodesic paths on manifolds*, Proc. Natl. Acad. Sci., 95 (1998), pp. 8431–8435, https://doi.org/10.1073/pnas.95.15.8431.

[20] F. Mémoli and G. Sapiro, *Fast computation of weighted distance functions and geodesics on implicit hyper-surfaces*, J. Comput. Phys., 173 (2001), pp. 730–764, https://doi.org/10.1006/jcph.2001.6910.

[21] J. S. B. Mitchell, D. M. Mount, and C. H. Papadimitriou, *The discrete geodesic problem*, SIAM J. Comput., 16 (1987), pp. 647–668, https://doi.org/10.1137/0216045.

[22] R. Monneau, *Introduction to the Fast Marching Method*, 2010, https://hal.science/hal-00530910.

[23] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Applied Mathematical Sciences 153, Springer, New York, NY, 2003, https://doi.org/10.1007/b98879.

[24] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, *DeepSDF: Learning continuous signed distance functions for shape representation*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 165–174.

[25] A. Pressley, *Elementary Differential Geometry*, Springer Science & Business Media, 2001.

[26] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, 1970.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[27] M. Torkaman, L. Zhu, P. Karasev, and A. Tannenbaum, *Sulci segmentation using geometric active contours*, in Progress in Biomedical Optics and Imaging - Proceedings of SPIE, Vol. 10133, 2017, 101330I, https://doi.org/10.1117/12.2249716.

[28] H.-K. Zhao, S. Osher, and R. Fedkiw, *Fast surface reconstruction using the level set method*, in Proceedings IEEE Workshop on Variational and Level Set Methods in Computer Vision, 2001, pp. 194–201, https://doi.org/10.1109/VLSM.2001.938900.

Downloaded 04/23/26 to 129.186.192.138 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.