Eurographics Symposium on Geometry Processing 2025
M. Attene and S. Sellán
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 44 (2025), Number 5

The Affine Heat Method

Yousuf Soliman[1] and Nicholas Sharp[2]

[1]Side Effects Software, Canada
[2]NVIDIA, USA

Figure 1: ***Logarithmic Maps.*** *The geodesic distance along a surface is central to many applications in computer graphics and machine learning, however extracting directional information from it requires numerical differentiation that decreases regularity. The logarithmic map instead encodes both direction and distance as a local parameterization about a point. Our Affine Heat Method computes the logarithmic map from a source point directly by heat diffusion (*left*). The radial and angular components of the parameterization give estimates of the geodesic distance (*middle*) and direction to any other point from the source (*right*).*

Abstract

This work presents the Affine Heat Method for computing logarithmic maps. These maps are local surface parameterizations defined by the direction and distance along shortest geodesic paths from a given source point, and arise in many geometric tasks from local texture mapping to geodesic distance-based optimization. Our main insight is to define a connection Laplacian with a homogeneous coordinate accounting for the translation between tangent coordinate frames; the action of short-time heat flow under this Laplacian gives both the direction and distance from the source, along shortest geodesics. The resulting numerical method is straightforward to implement, fast, and improves accuracy compared to past approaches. We present two variants of the method, one of which enables pre-computation for fast repeated solves, while the other resolves the map even near the cut locus in high detail. As with prior heat methods, our approach can be applied in any dimension and to any spatial discretization, including polygonal meshes and point clouds, which we demonstrate along with applications of the method.

1. Introduction

The *logarithmic map* from a point p in a curved domain is a local parameterization where all of the straight lines through the origin in the parameterization trace out geodesics through p. Also known as normal coordinates or geodesic polar coordinates, these are special parameterizations, well-known in differential geometry, that can dramatically simplify the analysis of geometric quantities, and they have proven themselves a useful addition to the discrete geometry processing toolbox. As these coordinates provide local parameterizations rooted at points on a surface (Fig. 1), they provide a straightforward interface to performing operations like surface decaling and sculpting [SGW06, SSC19, ML24].

A fundamental related concept is the notion of parallel transport of tangent vectors, describing how to move a tangent vector along a curve coherently as tangent spaces vary. This core operation is utilized in a variety of computer graphics and machine learning applications [Sch13, BYF*19, LDL*22]. Recent work exploits the relationship between parallel transport along shortest geodesics and the vector heat equation for direction field synthesis [SSC19] and generalized signed distance computations [FC24]. Both of these algorithms are so-called *heat methods* [CWW13], which perform geometric computations through the manipulation of solutions of short-time heat diffusion. These prior works apply parallel transport on the *tangent bundle*, which formalizes the intuitive notion that adjacent tangent vectors are similar if they point in the same direction and have the same magnitude—this is the most common notion of parallel transport in geometry processing.

In this work we observe that a simple change to the notion of parallel transport—introducing a homogeneous component to encode translations in addition to rotations—yields a new heat method which directly computes logarithmic maps. While past work has computed logarithmic maps indirectly as a derived quantity from heat flows [SSC19, HA19], we significantly improve accuracy and robustness by avoiding the need for imperfect, carefully-constructed initial conditions or approximate derivatives. We call this new approach the *Affine Heat Method* (AHM), since it utilizes diffusion with respect to an affine connection . We present two algorithmic variants: the first uses a single affine connection Laplacian for all heat flows, and second constructs an adapted Laplacian with respect to the source point, further increasing numerical accuracy at the cost of additional computation.

Since our affine heat methods do not involve the integration of vector fields, they are conceptually simpler than the approximation of the logarithmic maps introduced in [SSC19, HA19]. More importantly, the parameterization quality of the local coordinates are improved dramatically over the entire surface, especially near the source point, where the parameterizations produced by of [SSC19, HA19] introduces local distortion (see Figs. 2, 6 and 26), and the cut locus where our method averages the parameterization directly (see Fig. 12). We demonstrate the utility of our method on applications such as decaling (Fig. 17), UV flattening (Fig. 18), and the computation of high-quality and smooth stroke-aligned parameterizations (Fig. 19). While our logarithmic map (from points) can be used as a drop-in replacement for the logarithmic parameterizations computed by previous work, our stroke-aligned parameterizations make essential use of the affine diffusion and cannot be computed by any existing heat method.

2. Related Work

Surface parameterization is a fundamental topic in computer graphics and differential geometry that has garnered the attention of many researchers due to its importance in, *e.g.*, texturing workflows, surface registration, mesh generation, and manufacturing. For our discussion, we coarsely categorize parameterization methods into two types: local parameterizations that provide a map from a Euclidean coordinate system to a local region around a specified point, and global parameterizations that provide a map over the entire geometry. Global parameterization methods typically minimize a distortion energy subject to a variety of constraints on points and seams [KLS03, SSC18, GKK*24, AFSHA24]. Local parameterization methods, on the other hand, typically are unconstrained and advance the parameterization incrementally along an expanding wavefront. Note that past work in visual computing has used the terms "logarithmic" and "exponential" map interchangeably, in this text we will use the former. Schmidt and collaborators introduced discrete exponential maps [SGW06, Sch13] (DEM) to compute local parameterizations rooted at points or aligned with strokes. While it produces accurate local approximations, the parameterization quality falls off quickly when geometric complexity increases with distance. More recently, Madan and Levin [ML24] approximate the local parameterization on any surface representation given by a signed implicit function and compatible projection operator—their approach, also based on explicit radial tracing, parameterizes surface patches with geodesic splines.

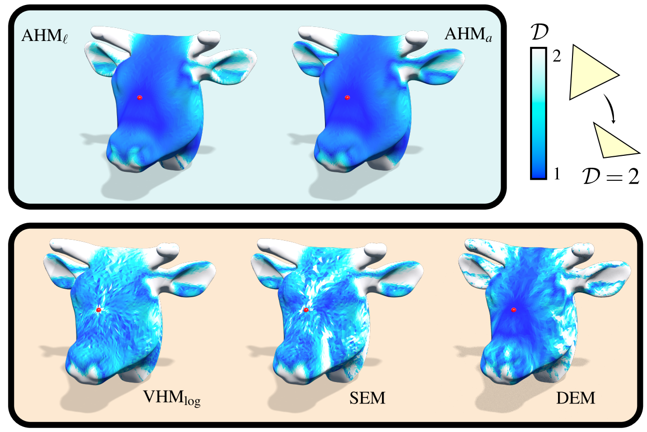


Figure 2: ***Metric Distortion.*** *Compared with prior methods VHM_{log} [SSC19], SEM [HA19], and DEM [SGW06] (*bottom row*), both variants of our affine heat method produce parameterizations with dramatically less metric distortion ($\mathcal{D}$) (Eqn. 18). We remark that our parameterizations are isometric at the source, just as the logarithmic map is in the smooth setting.*

The method we present lies somewhere in between local and global methods: we also approximate the logarithmic map, but we extend the parameterization globally to provide distance and direction from all points at once. In [CWW13], the heat method was introduced for approximating geodesic distance from short-time heat diffusion, and [BF15] extend a similar perspective as an iterative scheme. In subsequent work, [SSC19] introduced the vector heat method for approximating parallel transport of tangent vectors along shortest geodesics via short-time vector diffusion. They also utilized this operation to approximate surface logarithmic maps—we call this approximation of the logarithmic map $\text{VHM}_{\log}$. Unfortunately, the parameterization quality is suboptimal in part due to the inaccurate approximation of the radial vector field. In concurrent work, [HA19] also introduced a method for approximating the logarithmic map (SEM), computing the angular component of the parameterization separately by diffusion from the source neighborhood. The most recent incarnation of a heat method is the signed heat method of [FC24] for computing generalized signed distance functions to broken geometries.

3. Preliminaries

Our algorithm is best described using the language of vector bundles, in both the smooth and discrete setting; here we briefly review background and establish notation. We take care to properly formalize the algorithm in the smooth setting, but in practice both versions of our algorithm are straightforward to implement by assembling a Laplace-like matrix—the eager reader may proceed to Alg. 1.

3.1. Euclidean Motions and Homogeneous Coordinates

We make use of a homogeneous coordinate to represent affine transformations by linear ones. The subgroup of affine transformations we use to develop curves is the Lie group $\mathrm{SE}(n)$ of Euclidean motions in $\mathbb{R}^n$, with group multiplication given by composition. Every element $g \in \mathrm{SE}(n)$ is of the form

$$g(x) = Ax + b$$

for some rotation matrix $A \in \mathrm{SO}(n)$ and some translation $b \in \mathbb{R}^n$. Using homogeneous coordinates, g can be represented as a $(n+1) \times (n+1)$ matrix

$$g = \begin{pmatrix} A & b \\ 0 & 1 \end{pmatrix}$$

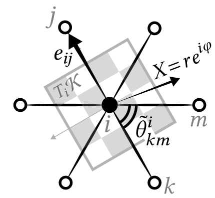
so that $g(x) = g\binom{x}{1}$. To simplify the notation, for a point $x \in \mathbb{R}^n$, we denote its lift into $\mathbb{R}^{n+1}$ by $[x] = \binom{x}{1} \in \mathbb{R}^{n+1}$.

3.2. Discrete Surfaces

We also need basic notions about the geometry of triangle meshes.

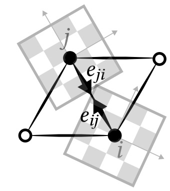
Combinatorics By a discrete surface we mean a two-dimensional orientable manifold simplicial complex $\mathcal{K} = (\mathcal{V}, \mathcal{E}, \mathcal{F})$, possibly with boundary. We denote oriented simplices by tuples of their vertex indices.

Discrete Metric The geometry of a triangle mesh is commonly encoded by a 3D position associated with each vertex. In the discrete setting, our algorithm actually depends only on weaker data: a discrete metric, that is a choice of positive edge lengths $\ell : \mathcal{E} \to \mathbb{R}_{>0}$ satisfying the triangle inequality on each face. In fact, this data is equivalent to the choice of a piecewise constant Riemannian metric in the faces that is compatible along the edges (see [KP16]). From edge lengths, one can easily compute all relevant geometric quantities like angle and area. Edge lengths are easily computed from vertex positions in the common case, but this generality gains us the ability to use intrinsic Delaunay triangulations for increased robustness, if desired [SGC21].



Discrete Tangent Bundle We will use the intrinsic tangent spaces at the vertices as our discrete tangent bundle. At each vertex $i \in \mathcal{V}$, we encode tangent vectors $X_i \in T_i\mathcal{K}$ in local polar coordinates (r_i, φ_i) following [KCPS13]: an arbitrary reference edge e_{ij} is chosen to be the complex unit. The remaining edge vectors are assigned polar coordinates by normalizing the angle sum $\Theta_i = \sum_{ijk} \theta^i_{jk}$ to 2π—the angles between tangent vectors are taken to be $\tilde{\theta}^i_{km} = 2\pi\theta^i_{km}/\Theta_i$. These normalized angles and edge lengths can be joined to provide coordinates of all of the outgoing edges.

Discrete Levi-Civita Connection The discrete Levi-Civita connection is the map $r^{\nabla}_{ij} : T_i\mathcal{K} \to T_j\mathcal{K}$ for each oriented edge ij describing parallel transport between the tangent spaces. The parallel transport is characterized by the property that the tangent vector $e_{ij} \in T_i\mathcal{K}$ is sent to $-e_{ji} \in T_j\mathcal{K}$. So after choosing a coordinate system in the tangent spaces, these vectors are represented by complex numbers and the parallel transport is given by



$$r^{\nabla}_{ij} := -e_{ji}/e_{ij}. \tag{1}$$

Discrete Vector Bundles To discretize the notion of affine vector diffusion we need to consider more general vector bundles over the vertices of a mesh. A *discrete vector bundle* of rank d is an assignment of vector spaces $V_i \cong \mathbb{R}^d$ for each vertex $i \in \mathcal{V}$. A *connection* ∇ is represented by an assignment of linear maps $r^{\nabla}_{ij} : V_i \to V_j$ for each oriented edge ij, satisfying $r^{\nabla}_{ji} \circ r^{\nabla}_{ij} = \mathrm{id}_{V_i}$. These maps are interpreted as parallel transport maps between the fibers of the bundle [LTGD16]. The discrete tangent bundle with the Levi-Civita connection is the standard example of such a discrete vector bundle endowed with a discrete connection.

Discrete Connection Laplacians Just as the cotan Laplacian is the quadratic form associated to the Dirichlet energy

$$\mathcal{E}(u) := \sum_{ij \in \mathcal{E}} w_{ij} |du_{ij}|^2,$$

where $du_{ij} := u_j - u_i$, we take the discrete connection Laplacian to be the quadratic form associated to the vector Dirichlet energy

$$\mathcal{E}^{\nabla}(X) := \sum_{ij \in \mathcal{E}} w_{ij} |d^{\nabla} X_{ij}|^2, \tag{2}$$

where $d^{\nabla} X_{ij} := X_j - r^{\nabla}_{ij} X_i$. This definition works for any discrete connection over a discrete vector bundle with fibers over the vertices. The discrete connection Laplacian of a connection ∇_0 over a discrete vector bundle of rank d is a matrix $\mathsf{L}^{\nabla_0} \in \mathbb{R}^{d|\mathcal{V}| \times d|\mathcal{V}|}$ that can be built facewise by assembling for each triangle $ijk \in \mathcal{F}$ the local $3d \times 3d$ matrix

$$\begin{bmatrix} (w_{ij} + w_{ki})\,\mathrm{id}_d & -w_{ij} r^{\nabla_0}_{ij} & -w_{ki} r^{\nabla_0}_{ik} \\ -w_{ij} r^{\nabla_0}_{ji} & (w_{ij} + w_{jk})\,\mathrm{id}_d & -w_{jk} r^{\nabla_0}_{jk} \\ -w_{ki} r^{\nabla_0}_{ki} & -w_{jk} r^{\nabla_0}_{kj} & (w_{jk} + w_{ki})\,\mathrm{id}_d \end{bmatrix} \tag{3}$$

and accumulating the entries in L^{∇_0} according to the triangle's vertex indices. In the case of the Levi-Civita connection, the parallel transport maps can be identified as multiplication by complex numbers allowing us to express the associated connection Laplacian as a complex matrix $\mathsf{L}^\nabla \in \mathbb{C}^{|\mathcal{V}|\times|\mathcal{V}|}$. We will also need the diagonal lumped mass matrix M to discretize the heat equation—it is either real with size $d|\mathcal{V}| \times d|\mathcal{V}|$ in the case of L^{∇_0} or complex with $|\mathcal{V}| \times |\mathcal{V}|$ in the case of L^∇.

3.3. The Vector Heat Method

The central idea of the vector heat method [SSC19] is that parallel transport along minimal geodesics can be computed by short time heat diffusion. The approach can be used to approximate parallel transport in arbitrary vector bundles with a connection, but in all cases it results in parallel transport along the shortest geodesics in M. To precisely describe the setup and the asymptotic expansion of the heat kernel upon which the algorithm is based, we need to briefly review the notion of vector bundles and connections. We refer the reader interested in a more complete introduction to the theory of connections to [Jos08, Chap. 4].

Vector Bundles Smooth vector bundles, as opposed to their discrete analogs, assign a family of vector spaces to every point on a smooth manifold—for example, the tangent spaces at different points can be joined together to form the tangent bundle. The vector bundle data is encoded in a smooth manifold E fibered over a base manifold M, and we denote it by $E \to M$ and write E_x for the vector subspace of E that projects down to a point $x \in M$. A choice of a vector in each fiber is called a *section* of the bundle, and we write ΓE for the space of all sections. For example, a section of the tangent bundle is a vector field and $\Gamma(TM) = \mathfrak{X}(M)$, and a section of a trivial bundle $\underline{\mathbb{R}^k} \to M$ is a smooth function $M \to \mathbb{R}^k$.

Connections To differentiate a section $\sigma \in \Gamma(E)$ along a tangent direction X requires the choice of a connection to identify the changing vector spaces. A *connection* ∇ on a vector bundle describes how to compute the directional (covariant) derivative $\nabla_X \sigma$ and gives rise to the notion of parallel transport by asking that the directional derivative vanishes along a curve. In the case of the tangent bundle on a Riemannian manifold, the Levi-Civita connection is the canonical choice. Similarly, there is a natural connection on any trivial bundle: the directional derivative (*i.e.*, the exterior derivative).

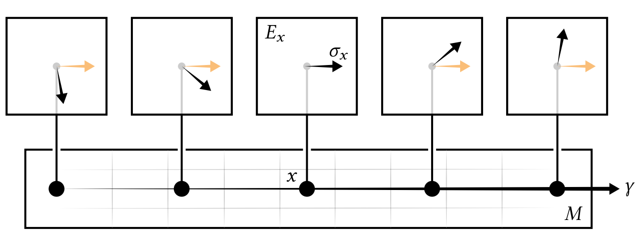


Figure 3: ***Vector Bundles with Connection.*** *A diagrammatic representation of a trivial vector bundle $E = \underline{\mathbb{R}^2}$ over a curve γ in M. The yellow vector field is a parallel section with respect to the trivial connection, while the black vector field is a parallel section with respect to the connection $\nabla = d - \alpha$ with matrix-valued 1-form $\alpha(\gamma') = \left(\begin{smallmatrix} 0 & 1 \\ -1 & 0 \end{smallmatrix}\right)$. This infinitesimal rotation introduces a twist into the identification of neighboring tangent spaces.*

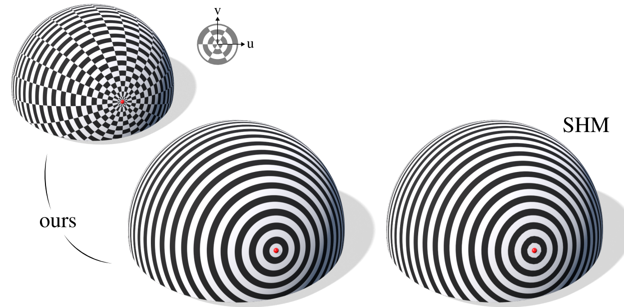


Figure 4: ***Boundary Behavior.*** *Since the approximation of parallel transport obtained from the short time asymptotics of the heat kernel is not strongly influenced by the presence of a boundary, both our method* (left) *and the signed heat method of [FC24]* (right) *realize the correct boundary behavior for geodesic distance.*

However, these are not the only meaningful connections. Any other connection ∇ on a trivial bundle $\underline{\mathbb{R}^k} \to M$ is of the form

$$\nabla_X f = d_X f - \alpha(X) f,$$

for some matrix-valued 1-form $\alpha \in \Omega^1(M; \mathbb{R}^{k\times k})$, and we write $\nabla = d - \alpha$ to denote this connection [Jos08]. In the general case, the space of connections is an affine space and the difference between two connection is given by an endomorphism valued 1-form $\alpha \in \Omega^1(M; \mathrm{End}\, E)$. The matrix-valued 1-form α has a geometric interpretation: it describes how a basis needs to change to remain parallel (Fig. 3). It is often useful to restrict the α to take values in a subgroup of all matrices (*e.g.*, infinitesimal rotations) so that the parallel transport preserves certain geometric properties. For example, integrating the rotation-valued connection in Fig. 3 between successive points produces a rotation-valued parallel transport map (*i.e.*, a discrete metric connection). We will use these representations with α taking values in the space of Euclidean transformations in the description of our algorithm in Sec. 4.2.

Vector Diffusion Heat diffusion is naturally described as the gradient flow of the Dirichlet energy of temperature, and this interpretation generalizes to the vectorial setting as well. For a section $\sigma \in \Gamma(E)$, we can define the *vector Dirichlet energy* (*cf.* , Eqn. 2)

$$\mathcal{E}^\nabla(\sigma) = \tfrac{1}{2} \int_M |\nabla\sigma|^2,$$

where the integral is with respect to the Riemannian volume form. Its L^2-gradient flow is the *vector heat equation*

$$(\partial_t - \Delta^\nabla)\sigma_t = 0,$$

where Δ^∇ is the *connection Laplacian* associated to ∇. Intuitively, this evolution smears the initial vectorial distribution, and the precise behavior can be understood by studying the heat kernel $k_t^\nabla(x,y) : E_x \to E_y$, which describes the simpler picture of how a single vector at a x diffuses to y in some time t. Short time asymptotics of the heat kernel reveal a deep relationship to parallel transport along shortest geodesics that can be used for geometric computation. [BGV92, Theorem 2.30] states that the magnitude of a vector falls off exponentially like $(4\pi t)^{-n/2}e^{-d(x,y)^2/4t}$, exactly as for the scalar heat equation, and the leading order term of the asymptotic expansion is equal to the parallel transport $\mathcal{P}_{x\to y} : E_x \to E_y$ along the unique geodesic connection x and y.

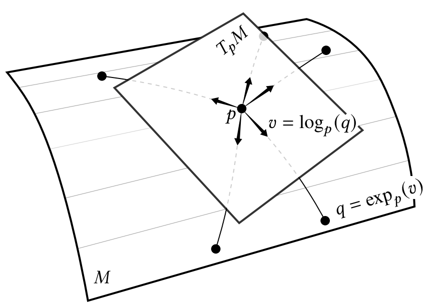
In practice, this means that we can approximate the parallel transport of a vector rooted at x along all geodesics emanating from x by normalizing the solution of a short time heat equation. This approximation algorithm is what we call the vector heat method, irrespective of the choice of base space, vector bundle, or connection. We remark that using Neumann boundary conditions with the vector heat method reproduces the correct boundary behavior of parallel transport along minimal geodesics (Fig. 4).

Time Discretization We follow the same approach of past heat methods [CWW13, SSC19, FC24] to obtaining a semidiscrete algorithm: apply a single step of a backward Euler approximation of the heat equation. Solutions of connection Laplacian based heat equations at small time $t = \tau$ are approximated by the linear equation

$$(\mathrm{id} - \tau\Delta^{\nabla})\sigma_\tau = \sigma_0, \tag{4}$$

where σ_0 is the initial condition used to start the diffusion[†].

3.4. Exponential and Logarithmic Maps



The exponential map on a Riemannian manifold (M, g) at a point $p \in M$ is a map $\exp_p : T_pM \to M$ defined by following a geodesic from p for a unit time along a specified tangent direction. The logarithmic map is the inverse of this map[‡]

$$\log_p : M \to T_pM,$$

that describes for any point $q \in M$ what tangent direction $X \in T_pM$ do we need to follow to reach q. The logarithmic map is technically a map into the tangent space, but we can turn it into a parameterization after choosing any basis in T_pM.

† σ_0 only needs to have the regularity of a vector-valued measure for the equation to make sense—we consider initial conditions given by Hausdorff measures supported on points and curves

‡ Technically, the logarithmic map is only defined on the subset of M where the exponential map is a diffeomorphism.

3.5. Visualizing Logarithmic Maps

Surface parameterizations in computer graphics are often visualized using a checkerboard pattern induced by a Cartesian coordinate system. However, since logarithmic maps describe distance and direction information to a source point, we find it more appropriate to visualize the parameterization using a checkerboard pattern induced by polar coordinates instead, which we will use throughout the paper (unless otherwise noted). This choice breaks the translational symmetry of the checkerboard pattern, revealing the location of the source point along with the measure of distance and direction back to the source. Even when using the analytical expression for the logarithmic map on the sphere (Fig. 5), the Cartesian checkerboard pattern looks highly distorted—the clover-esque distortion near the antipodal point is an artifact that can be understood by imagining pulling the four corners of a square to a single point.

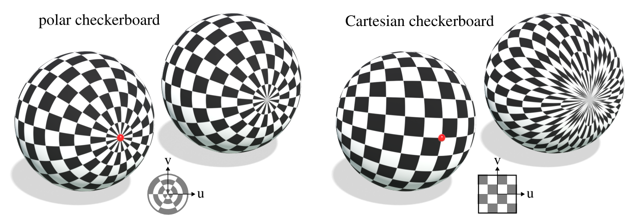


Figure 5: ***Spherical Polar Coordinates.*** *Using a Cartesian checkerboard pattern (*right*) to visualize the exact logarithmic map on the sphere results in highly distorted textures and artifacts unrelated to the parameterization. Using a polar checkerboard pattern instead (*left*) better visualizes the geometric information encoded in these local parameterizations.*

4. The Affine Heat Methods

We now describe two closely related heat methods for computing logarithmic maps on curved domains. Both methods use a notion of affine parallel transport to obtain the logarithmic map directly through diffusion. The localized variant (Sec. 4.1) will produce a radial vector field that approximates the gradient of geodesic distance—measured in a geodesic frame, this radial field is equal to the logarithmic map. Turning to the adaptive variant (Sec. 4.2), we use the same geodesic frame to instead build a diffusion operator adapted to the source point—the corresponding radial field will be the logarithmic map itself. Sec. 6 discusses the practical tradeoffs between these methods; in short both are fast and more accurate than prior approaches, but the localized variant allows prefactorization for fast repeated solves, while the adaptive variant further improves numerical accuracy.

Throughout this section, we will consider an n-dimensional Riemannian manifold M with metric g and fix a point $p \in M$ about which we will construct our logarithmic maps. So that we may focus on the essential geometric picture, we further assume that for every point $q \in M$ there is a unique geodesic from p to q. To keep the discrete picture concrete, we only consider two-dimensional triangle meshes $\mathcal{K} = (\mathcal{V}, \mathcal{E}, \mathcal{F})$ until Sec. 4.3.

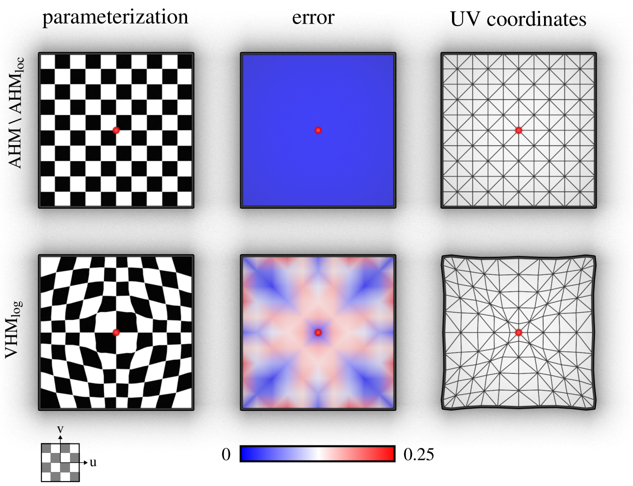


Figure 6: ***A Flat Comparison.*** *Computing logarithmic maps using the vector heat method approach introduced in [SSC19] results in a large amount of distortion near the source point (*bottom row*); both versions our affine heat method result in much more accurate parameterizations, in the Euclidean case reproducing the ground truth (*top row*).*

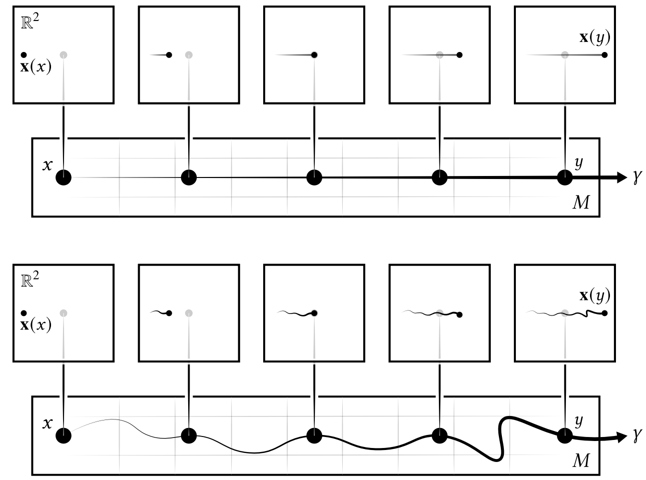


Figure 7: ***Affine Parallel Transport.*** *Using translations in the identification of neighboring tangent spaces via a connection can be used to integrate arbitrary curves in the base space isometrically in the fibers.*

Parameterizing Euclidean Space Before presenting the general algorithm on curved domains, we start in the Euclidean setting where the essential feature of our method can already be explained. The goal is to construct a connection $\overline{\nabla}$ that encodes the logarithmic map in $\mathbb{R}^n$ through parallel transport along straight lines through a point. While the resulting parameterization will be the trivial one, *i.e.*, the identity map $x \in \mathbb{R}^n \mapsto x$, our construction generalizes naturally to any curved manifold where no trivial parameterization exists.

Given a curve γ and a map $\mathbf{x} : \mathbb{R}^n \to \mathbb{R}^n$ we want to encode

$$d_{\gamma'}\mathbf{x} = \gamma' \tag{5}$$

as the condition that $\mathbf{x}$ is a parallel section along γ of some connection. This would imply that we could compute the parameterization $\mathbf{x}$ by the corresponding parallel transport along the geodesics (straight lines) through a point. However, if we only consider connections $\nabla = d - \alpha$ on the trivial bundle $\underline{\mathbb{R}^n} \to \mathbb{R}^n$ the parallel sections along a curve γ will satisfy the differential equation

$$d_{\gamma'}\mathbf{x} = \alpha(\gamma')\mathbf{x}.$$

It is impossible to express Eqn. 5 in this way since $\alpha(\gamma')$ acts linearly on $\mathbf{x}$. The crucial observation motivating the introduction of a homogeneous coordinate is that we can use a translation to resolve this if we allow ourselves to work with more general affine transformations of the form $\mathbf{x} \mapsto \alpha(\gamma')\mathbf{x} + \beta(\gamma')$.

On the trivial bundle $\underline{\mathbb{R}^{n+1}} \to \mathbb{R}^n$ we consider the connection

$$\overline{\nabla} := d - \begin{pmatrix} 0 & \mathrm{id} \\ 0 & 0 \end{pmatrix}. \tag{6}$$

Here, we treat the identity map as the 1-form $\mathrm{id} \in \Omega^1(\mathbb{R}^n; \mathbb{R}^n)$ defined by $\mathrm{id}(X) = X$. The matrix-valued 1-form we add to d describes an infinitesimal translation (with no rotation) in the direction being differentiated (Fig. 7). Using that a section is a map $(\mathbf{x}, \lambda) : \mathbb{R}^n \to \mathbb{R}^{n+1}$, we can verify that $\overline{\nabla}$ encodes Eqn. 5 by computing the $\overline{\nabla}$-covariant directive along a curve γ:

$$\overline{\nabla}_{\gamma'} \begin{pmatrix} \mathbf{x} \\ \lambda \end{pmatrix} = d_{\gamma'} \begin{pmatrix} \mathbf{x} \\ \lambda \end{pmatrix} - \begin{pmatrix} 0 & \gamma' \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \lambda \end{pmatrix} = \begin{pmatrix} d_{\gamma'}\mathbf{x} - \lambda\gamma' \\ d_{\gamma'}\lambda \end{pmatrix}.$$

If $(\mathbf{x}, \lambda)$ is parallel along a curve γ then $\nabla_{\gamma'} \binom{\mathbf{x}}{\lambda} = 0$ and so

$$d_{\gamma'}\mathbf{x} = \lambda\gamma', \qquad d_{\gamma'}\lambda = 0.$$

The second equation implies that the homogeneous parameter must be constant along γ. Therefore, since λ is constant, the first equation can be explicitly integrated to obtain that $\mathbf{x} = \lambda\gamma + \mathbf{x}_0$ with some integration constant $\mathbf{x}_0 \in \mathbb{R}^n$. Dividing out the homogeneous coordinate, we see that if $(\mathbf{x}, \lambda)$ is parallel along γ then it describes a translate of γ in $\mathbb{R}^n$.

We can now conclude that if $\mathbf{x}(0) = 0$ and $\lambda(0) = 1$ and $(\mathbf{x}, \lambda)$ is parallel along *all* geodesics through the origin then

$$\begin{pmatrix} \mathbf{x}(x) \\ \lambda(x) \end{pmatrix} = \begin{pmatrix} x \\ 1 \end{pmatrix}$$

for all $x \in \mathbb{R}^n$. We approximate the parallel transport along all of these geodesics using the vector heat method—in particular, after dividing out the homogeneous coordinate of a short-time affine heat diffusion we obtain the trivial parameterization of Euclidean space centered around the origin. Both the localized and adaptive variants of the affine heat method will reduce to the picture we just described when the underlying manifold is Euclidean space. Remarkably, even in the discrete our method reproduces the exact solution on Euclidean domains, up to floating point precision. See Rem. 2.

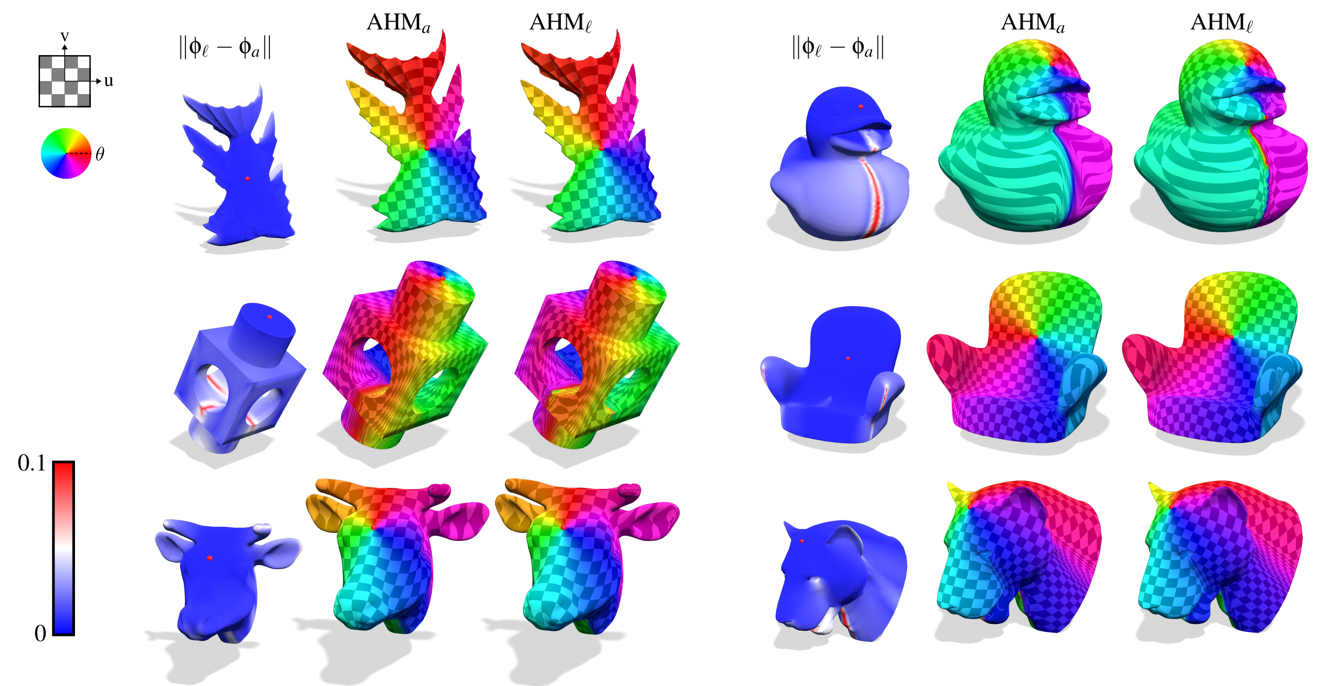


Figure 8: ***Gallery.*** *Within the injectivity radius, the parameterization from the localized affine heat method (ϕ_ℓ) is nearly indistinguishable to the more accurate results provided by the adaptive affine heat method (ϕ_a). We normalize the surfaces so that the maximum of the geodesic distance to the source is equal to 1. Even near the cut locus, where the ground truth is ill-defined, AHM produces a continuous approximation of the angular component of the parameterization. Visualized with Cartesian checkerboards.*

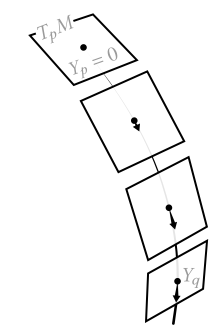
4.1. The Localized Affine Heat Method

The *Localized Affine Heat Method* (AHM$_\ell$) follows an approach similar to VHM$_{\log}$: to construct the angular component of the parameterization we measure the angle between a "radial field" and a "horizontal field" (Fig. 9). The former being tangent to all geodesics from the source, while the latter making a constant angle with each such geodesic. Ensuring that the angular and distance components of the parameterization, computed separately, are compatible and produce a low-distortion map is difficult in practice. We instead propose to compute a radial field whose length is equal to the geodesic distance (*i.e.*, half the gradient of geodesic distance squared). With our homogeneous connection, we can use short-time heat flow to easily and accurately compute this quantity via parallel transport along shortest geodesics. Below, we start in the smooth setting, before discretizing AHM$_\ell$.

Smooth Picture To compute the radial field, we take inspiration from the Euclidean setting and consider the connection ∇^ℓ on the vector bundle $TM \oplus \mathbb{R}$ with block-decomposition

$$\nabla^\ell = \begin{pmatrix} \nabla & 0 \\ 0 & d \end{pmatrix} - \begin{pmatrix} 0 & \mathrm{id} \\ 0 & 0 \end{pmatrix},$$

where $\mathrm{id} \in \Omega^1(M;TM)$ is the tautological 1-form defined by $\mathrm{id}(X) = X$. Our choice of connection ensures that parallel transport of the zero vector $(0,1) \in T_pM \oplus \mathbb{R}$, in the tangent space of the source p, along geodesics through p produces the radial vector field $\frac{1}{2}\,\mathrm{grad}(d_p^2)$:



Lemma 1 Suppose that $(Y,\lambda) \in \Gamma(TM \oplus \mathbb{R})$ satisfies $Y_p = 0 \in T_pM$ and $\lambda_p = 1$ and that (Y,λ) is ∇^ℓ-parallel along all geodesics through p. Then for any such geodesic γ through p we have that $Y = d_p(\gamma)\,\frac{\gamma'}{|\gamma'|}$ and $\lambda \equiv 1$.

Proof Consider a unit-speed parameterized geodesic $\gamma(t)$ satisfying $\gamma(0) = p$. Since (Y,λ) is parallel along γ we have that

$$\nabla^\ell_{\gamma'} \begin{pmatrix} Y \\ \lambda \end{pmatrix} = \begin{pmatrix} \nabla_{\gamma'} Y \\ d_{\gamma'}\lambda \end{pmatrix} - \begin{pmatrix} 0 & \gamma' \\ 0 & 0 \end{pmatrix} \begin{pmatrix} Y \\ \lambda \end{pmatrix} = \begin{pmatrix} \nabla_{\gamma'} Y - \lambda\gamma' \\ d_{\gamma'}\lambda \end{pmatrix} = 0.$$

The second equation implies that λ is constant along γ, and so $\lambda|_\gamma \equiv 1$. The first equation then implies that $\nabla_{\gamma'} Y = \gamma'$. Letting n be any parallel section along γ that is orthogonal to γ' we have

$$\langle Y,n\rangle' = \langle \nabla_{\gamma'} Y,n\rangle + \langle Y,\nabla_{\gamma'} n\rangle = 0$$

since the Levi-Civita connection ∇ is a metric connection. The first term vanishes since $\langle \nabla_{\gamma'} Y,n\rangle = \langle \gamma',n\rangle$ and n is orthogonal to γ', while the second term vanishes since n is parallel. This implies that $Y = L\gamma'$. To determine the length, we compute

$$\langle Y,Y\rangle' = 2\langle \nabla_{\gamma'} Y,Y\rangle = 2\langle \gamma',L\gamma'\rangle = 2L$$

since $|\gamma'| = 1$. This implies that $(L^2)' = 2L' = 2L$ and so $L'(t) = 1$. Since $L(0) = 0$ we deduce that $L(t) = t = d_p(\gamma(t))$ and the result follows. $\square$

Since the angle between parallel vector fields along a curve is constant, we can compute the logarithmic map $\mathbf{x} : M \to \mathbb{R}^n$ by measuring the coordinates of the radial field Y in a geodesic frame $E_1, \ldots, E_n$ adapted to p (App. A).

$$\mathbf{x}(p) = (\langle E_i(p), Y_p \rangle)_{i=1}^n. \tag{7}$$

For future reference, we define the identification induced by E_i as

$$\Phi : TM \to \mathbb{R}^n, \qquad X \in T_pM \mapsto (\langle E_i(p), X \rangle)_{i=1}^n, \tag{8}$$

so that $\mathbf{x}(p) = \Phi(Y_p)$.

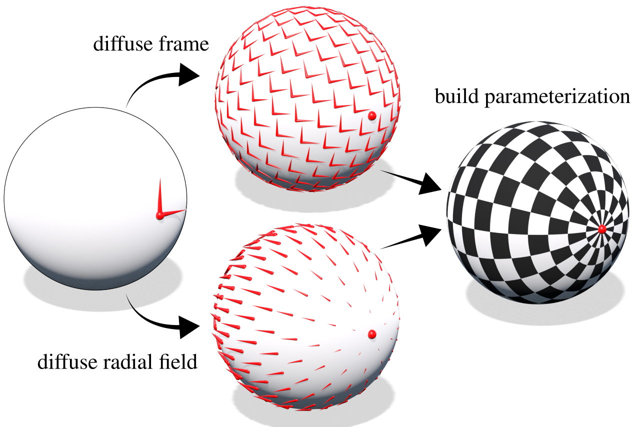
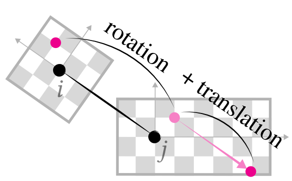


Figure 9: ***Localized Variant Overview.*** *The three steps of the localized affine heat method.*

Discrete Radial Field We discretize the bundle $TM \oplus \mathbb{R}$ by $T\mathcal{K} \oplus \mathbb{R}$ by adding a homogeneous coordinate to the vertex tangent spaces. In the discrete setting we can also understand the computation of the affine parallel transport corresponding to ∇^ℓ directly. The per-edge entries of the Laplacian matrix encode a notion of similarity between values at adjacent vertices: in a scalar Laplacian, values across an edge ij are considered similar when they are identical, while in a tangent vector connection Laplacian they are considered similar if they are equal after aligning the tangent spaces by a rotation r_{ij}^{∇} (Eqn. 1). Here we go one step further to discretize ∇^ℓ and build an operator that treats values as similar after a rotation and a translation between neighboring tangent spaces; similarity is defined after aligning the tangent spaces with the local Euclidean transformation:



$$r_{ij}^{\nabla^\ell} : T_i\mathcal{K} \to T_j\mathcal{K}, \qquad r_{ij}^{\nabla^\ell}(X_i) = r_{ij}^{\nabla} X_i + e_{ji}, \tag{9}$$

where $e_{ji} \in T_j\mathcal{K}$ is the edge vector. To build the corresponding connection Laplacian we need to represent these affine transformations by linear maps using a real homogeneous coordinate:

$$r_{ij}^{\nabla^\ell} = \begin{bmatrix} r_{ij}^{\nabla} & e_{ji} \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3},$$

which is an approximation of the parallel transport with respect to ∇^ℓ along the edge ij. Since the discretization of the ∇^ℓ connection is given by matrices in SE(2), the discrete ∇^ℓ connection Laplacian is a real matrix $\mathsf{L}^{\nabla^\ell} \in \mathbb{R}^{3|\mathcal{V}|\times 3|\mathcal{V}|}$. To approximate the radial field, we apply the vector heat method and diffuse the zero vector supported at a single source point p by solving

$$(\mathsf{M} + \tau \mathsf{L}^{\nabla^\ell}) \begin{pmatrix} \mathsf{Y} \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \delta_p \tag{10}$$

for $\mathsf{Y} : \mathcal{V} \to T\mathcal{K}$ and $\lambda : \mathcal{V} \to \mathbb{R}$. We obtain the radial field by dividing out the homogeneous coordinate. We choose the short time τ to be proportional to the mean edge-length squared (see Sec. 4.3).

ALGORITHM 1: The Localized Affine Heat Method

Input: A source point p with a unit vector $\mathsf{U}_p \in T_p\mathcal{K}$

Output: A parameterization $\phi : \mathcal{V} \to \mathbb{R}^2$.

1. Solve the affine diffusion equation $(\mathsf{M} + \tau \mathsf{L}^{\nabla^\ell})\binom{\mathsf{Y}}{\lambda} = \binom{0}{1}\delta_p$ (Eqn. 10).
2. Solve the tangent diffusion equation $(\mathsf{M} + \tau \mathsf{L}^{\nabla})\tilde{\mathsf{U}} = \mathsf{U}_p \delta_p$ (Eqn. 11).
3. Evaluate the parameterization $\phi_v := \Phi(\mathsf{Y}/\lambda)$ (Eqn. 12).

Discrete Geodesic Frame To obtain the final parameterization, we need to measure the coordinates of the radial field with respect to a geodesic frame adapted to p. As opposed to the arbitrary identification of the tangent spaces induced by choosing an arbitrary edge as the complex unit, the identification induced by a geodesic frame provides a consistent way to orient the tangent spaces relative to p (Fig. 10). We approximate the adapted geodesic using the vector heat method with respect to the Levi-Civita connection (*cf.* [SSC19]): diffuse and normalize a tangent vector $\mathsf{U}_p \in T_p\mathcal{K}$ to extend it to a discrete vector field U. Denoting the discrete tangent connection Laplacian $\mathsf{L}^{\nabla} \in \mathbb{C}^{|\mathcal{V}|\times|\mathcal{V}|}$, the discrete geodesic frame is obtained by normalizing the solution of

$$(\mathsf{M} + \tau \mathsf{L}^{\nabla})\tilde{\mathsf{U}} = \mathsf{U}_p \delta_p. \tag{11}$$

The coordinates induced by $\mathsf{U}_i := \tilde{\mathsf{U}}_i / |\tilde{\mathsf{U}}_i|$ defines an identification $\Phi_i : T_i\mathcal{K} \to \mathbb{C}$ adapted to the point p:

$$\Phi_i(X) := X/\mathsf{U}_i = (\langle X, \mathsf{U}_i \rangle, \langle X, \imath \mathsf{U}_i \rangle), \tag{12}$$

in the second equation we denote the imaginary unit $\imath$ and the real inner products $\langle \cdot, \cdot \rangle$. Following the smooth theory (Eqn. 7), we obtain the logarithmic map of p by evaluating the coordinates of radial field using Φ. A concise overview of the algorithm is given in Alg. 1.

Remark 1 Rather than also building and factoring the Levi-Civita connection Laplacian, we can exploit the fact that the upper left component of $r_{ij}^{\nabla^\ell}$ is the r_{ij}^{∇}—setting the homogeneous coordinate to zero, solving

$$(\mathsf{M} + \tau \mathsf{L}^{\nabla^\ell}) \begin{pmatrix} \tilde{\mathsf{U}} \\ 0 \end{pmatrix} = \begin{pmatrix} \mathsf{U}_p \delta_p \\ 0 \end{pmatrix}, \tag{13}$$

is equivalent to Eqn. 11.

4.2. The Adaptive Affine Heat Method

We now present the *Adaptive Affine Heat Method* (AHM$_a$) that uses the geodesic frame in the construction of the connection itself to

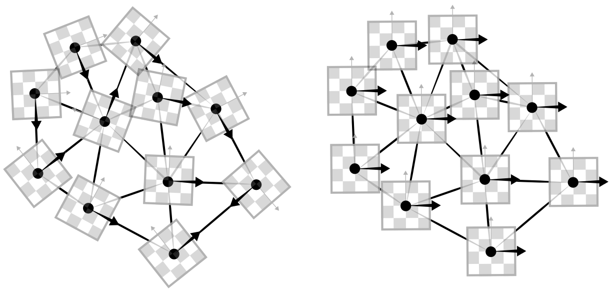


Figure 10: ***Tangent Space Coordinates.*** Left: *the identification given by choosing an arbitrary reference edge is useful for encoding tangent vectors numerically, but has no geometric significance.* Right: *a vector field obtained by the vector heat method induces a consistent orientation of the tangent spaces with respect to the frame at p.*

identify the tangent spaces with a fixed Euclidean space. The analogous construction of the radial field will produce the logarithmic map directly. Intuitively, this adaptive variant provides additional benefits in highly distorted regions since it averages the parameterization values directly.

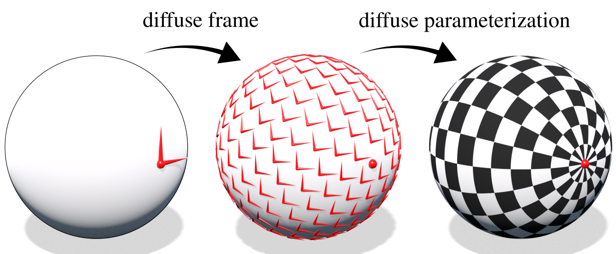


Figure 11: ***Overview.*** *The two steps of the adaptive affine heat method. After each diffusion, the resulting quantity is normalized to produce either a unit direction field or point in* $\mathbb{R}^2$.

Smooth Picture Exactly as in the Euclidean case, we work with the trivial bundle $\underline{\mathbb{R}^{n+1}} \to M$, but instead we consider a connection that determines the translation in an adapted geodesic frame:

$$\overline{\nabla}^{\Phi} := d - \begin{pmatrix} 0 & \Phi \circ \mathrm{id} \\ 0 & 0 \end{pmatrix}, \tag{14}$$

where $\Phi : TM \to \mathbb{R}^n$ is the geodesic frame obtained by parallel transport (Eqn. 8). Now consider a unit speed geodesic curve γ emanating from p and a section $(\mathbf{x}, \lambda) : M \to \mathbb{R}^{n+1}$ that is $\overline{\nabla}^{\Phi}$-parallel along γ. Just as in the Euclidean and localized cases, the homogeneous coordinate is constant along γ. Thus, without loss of generality, we can assume that $\lambda \equiv 1$. Since $(\mathbf{x}, \lambda)$ satisfies the ordinary differential equation

$$d_{\gamma'} \mathbf{x} = \Phi(\gamma')$$

along γ, and since $\Phi(\gamma')$ is constant (App. A), we conclude that $\mathbf{x}|_\gamma$ is a straight line. Since this characterizes the logarithmic map, we have shown that the logarithmic map can also be computed by $\overline{\nabla}^{\Phi}$-parallel transport of the initial condition [0] along the geodesics emanating from p:

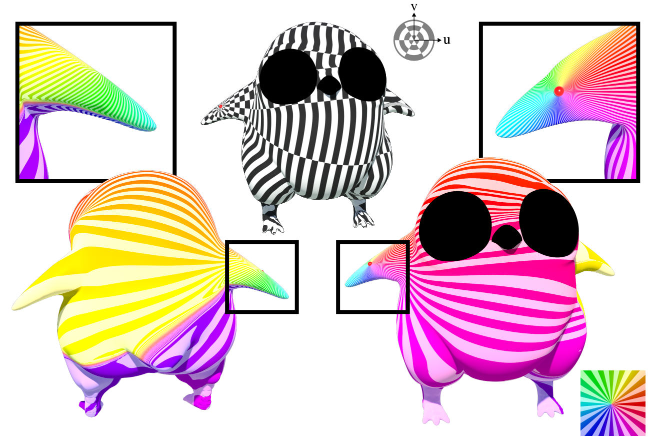


Figure 12: ***Colliding Geodesics.*** *Surface logarithmic maps are only well defined within the injectivity radius of the source point, where geodesics are guaranteed to never collide. Nevertheless, we produce smooth polar coordinates globally, even near the cut locus, rooted at points with extremely small injectivity radius.*

ALGORITHM 2: The Adaptive Affine Heat Method

Input: A source point p with a unit vector $\mathsf{U}_p \in T_p\mathcal{K}$
Output: A parameterization $\phi : \mathcal{V} \to \mathbb{R}^2$.

1. Solve the tangent diffusion equation $(\mathsf{M} + \tau \mathsf{L}^{\nabla})\tilde{\mathsf{U}} = \mathsf{U}_0$. (Eqn. 11)
2. Solve the affine diffusion equation $(\mathsf{M} + \tau \mathsf{L}^{\overline{\nabla}})\binom{\mathbf{x}}{\lambda} = \binom{0}{1}\delta_p$. (Eqn. 16)
3. Evaluate the parameterization $\phi_v := \mathbf{x}_v / \lambda_v$. (Eqn. 17)

Lemma 2 Suppose that $(\mathbf{x}, \lambda) : M \to \mathbb{R}^{n+1}$ satisfies $\mathbf{x}_p = 0$ and $\lambda_p = 1$ and that $(\mathbf{x}, \lambda)$ is $\overline{\nabla}^{\Phi}$-parallel along all geodesics through p. Then $\mathbf{x}(x) = \Phi(\log_p(x))$.

We will suppress the dependence of $\overline{\nabla}$ on Φ since the identification will always be clear from context. Now that we have constructed the connection $\overline{\nabla}$, encoding the logarithmic map in its parallel sections, we can simply apply the vector heat method to compute a parameterization of our domain rooted at $p \in M$. Since we also need to compute the identification Φ, the adaptive variation of our algorithm consists of two applications of the vector heat method (Fig. 11).

Discrete $\overline{\nabla}$ The discretization of the $\overline{\nabla}$ connection (Eqn. 14) is a linear map between the fibers of the trivial $\mathbb{R}^3$-bundle over the vertices $r_{ij}^{\overline{\nabla}} : \mathbb{R}^3_i \to \mathbb{R}^3_j$ that describes the parallel transport with respect to $\overline{\nabla}$. Since $\overline{\nabla}$ describes an infinitesimal translation in the direction that is being differentiated (after being identified with $\mathbb{R}^2$ using Φ computed from the result of Eqn. 11), the discrete parallel transport is given by the finite translation in the direction being differentiated—in this case, in the edge direction e_{ij}. As a matrix, we take

$$r_{ij}^{\overline{\nabla}} := \begin{pmatrix} \mathrm{id} & \Phi_i(e_{ij}) \\ 0 & 1 \end{pmatrix} \tag{15}$$

This discrete connection has the property that the development of any edge path is isometric, exactly as in the smooth setting. To approximate the logarithmic map, we now apply the vector heat method and diffuse the zero vector supported at a single point p and solve the discrete affine diffusion equation (*cf.* Eqn. 10):

$$(\mathsf{M}+\tau\mathsf{L}^{\overline{\nabla}})\begin{pmatrix}\mathbf{x}\\ \lambda\end{pmatrix}=\begin{pmatrix}0\\ 1\end{pmatrix}\delta_p. \tag{16}$$

The parameterization values $\phi:\mathcal{V}\to\mathbb{R}^2$ are obtained by normalizing the homogeneous coordinate:

$$\phi_v := \mathbf{x}_v/\lambda_v. \tag{17}$$

This procedure is summarized in Alg. 2.

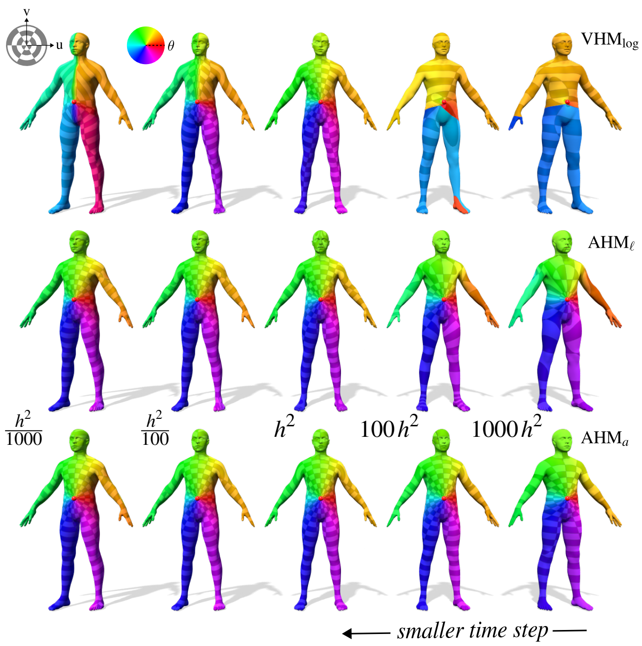


Figure 13: ***Very Short and Long Time Diffusion.*** *Compared to the method from [SSC19] (top), our affine heat method (bottom) provides accurate estimates of the log map across orders of magnitude of the time step used to estimate the heat kernels.*

Remark 2 On a planar triangle mesh Φ can be taken to be the identity map and so the affine parallel transport (Eqn. 15) has no curvature: for any closed loop of edges the parallel transport around it is given by the translation by the sum of the edge vectors, and this sum vanishes because the loop is closed. The absence of curvature implies that the corresponding affine heat diffusion is essentially the same as scalar diffusion (see App. B). We point out that, in this case, we can change the coordinates of the vertex tangent spaces so that discrete Levi-Civita connection is the identity and the localized variant is transformed into the adaptive one.

4.3. Details and Generalizations

Short Time Since our discrete algorithms amount to solving sparse linear systems representing short time heat diffusion, we need to choose an appropriate time step $\tau > 0$. Following work on prior heat methods [CWW13, SSC19], we let $\tau = h^2$ where h is the mean edge length. This ensures that the resulting algorithm is scale invariant. The improved robustness of the Affine Heat Method extends to the choice of τ: unlike $\mathrm{VHM}_{\log}$, it produces reasonable parameterizations across a large range of time steps used to approximate the heat kernel (see Fig. 13).

Source Discretization We discretize the right hand side of the time-discrete heat equations with values $(\mathsf{U}_0)_v \in \mathbb{C}$ and $(\mu_0)_v \in \mathbb{R}^3$ at vertices $v \in \mathcal{V}$. We first consider the situation when the source point coincides with a mesh vertex $i \in \mathcal{V}$, and we take

$$(\mathsf{U}_0)_v = \begin{cases} \mathsf{U}_p \in T_pM & \text{if } v = i,\\ 0 & \text{otherwise,}\end{cases}$$

and

$$(\mu_0)_v = \begin{cases} [0] & \text{if } v = i,\\ 0 & \text{otherwise.}\end{cases}$$

When the source point lies inside a triangle, we split the source to the three neighboring vertices according to the barycentric coordinates. For the vector field initial condition U_0, we need to make sure to express the initial tangent vector inside the face in the tangent spaces of its vertices—using the coordinates of an edge vector ij in both the tangent space of the face $X_{ijk} \in T_{ijk}\mathcal{K} \cong \mathbb{C}$ and in the tangent space of its source vertex $X_i \in T_i\mathcal{K} \cong \mathbb{C}$, we consider the parallel transport defined by the complex multiplication (with respect to a basis representation) by

$$r_{ijk\to i} = X_i/X_{ijk} \in \mathbb{C}.$$

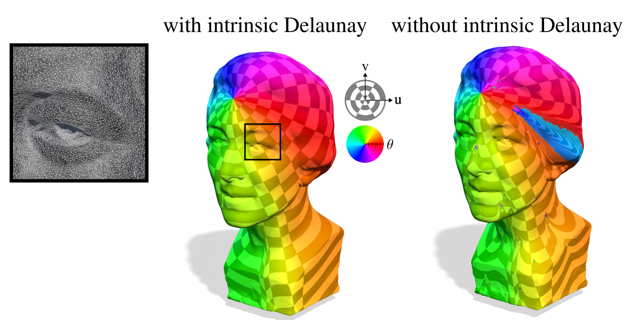


Figure 14: ***Intrinsic Delaunay Triangulations.*** *Our parameterization method only depends on the intrinsic geometry of the domain, enabling us to utilize the intrinsic Delaunay triangulation to ensure high quality parameterizations irrespective of the underlying quality of the input mesh.*

Intrinsic Triangulations The affine heat method relies solely on intrinsic quantities. This makes it possible to improve accuracy by using the intrinsic Delaunay triangulation [BS07] (Fig. 14). This is an important preprocessing step, as negative edge weights in the Laplacian may yield degenerate parameterization regions. Flipping to the intrinsic Delaunay triangulation resulted in continuous parameterizations (away from the cut locus) on all examples we encountered—we use the integer coordinates introduced in [GSC21] to represent intrinsic triangulations, and apply this intrinsic preprocessing on all examples and experiments.

Graphs The affine heat methods generalize to a wide variety of spatial discretizations with incidence relationships modeled by graphs. To implement the affine heat method on a graph $\Gamma = (\mathcal{V}_\Gamma, \mathcal{E}_\Gamma)$ we need the following quantities: (1) for each vertex $i \in \mathcal{V}_\Gamma$ a tangent space $T_i\Gamma \cong \mathbb{R}^n$ along with a trivialization $\Phi_i : T_i\Gamma \to \mathbb{R}^n$, (2) a local logarithmic map that assigns for each oriented edge $ij \in \mathcal{E}_\Gamma$ a tangent vector $e_{ij} \in T_i\Gamma$ describing the position of the vertex j in the tangent space of the source point, and (3) edge weights $w : \mathcal{E}_\Gamma \to \mathbb{R}$ used to build a graph Laplacian and vertex masses $m : \mathcal{V}_\Gamma \to \mathbb{R}$ to build a corresponding mass matrix. From this information, we can construct a discrete connection $r_{ij}^{\overline{\nabla}} : T_i\Gamma \to T_j\Gamma$ exactly as we did for a simplicial surface:

$$r_{ij}^{\overline{\nabla}} := \begin{pmatrix} \mathrm{id} & \Phi_i(e_{ij}) \\ 0 & 1 \end{pmatrix},$$

build the affine connection Laplacian, and apply the discrete algorithm from Alg. 2 verbatim.

We apply this generalization to polygonal meshes (Fig. 16) and point cloud scans (Fig. 15). So that we approximate geodesic polar coordinates, we use the trivialization given by approximating a geodesic frame using the vector heat method, and we refer to [SSC19] for further details on this step. For polygonal meshes, we use the Laplacian defined by [BHKB20], and define the connection Laplacian on polygonal meshes following [FC24, Sec. 8.3]. For point clouds, we use the tufted Laplacian defined by [SC20]. In both cases, we define the local logarithmic map by projecting the extrinsic edge vectors between neighboring vertices onto their tangent spaces.

5. Applications

Below, we consider several illustrative applications to demonstrate the utility of the high quality local parameterizations generated by the affine heat method. As logarithmic maps are general purpose tools, they have further applications beyond those we present (*e.g.*, computing the Riemannian center of mass [Wei37]—these are often called “Karcher means”, a term of unclear origins [Kar14]).

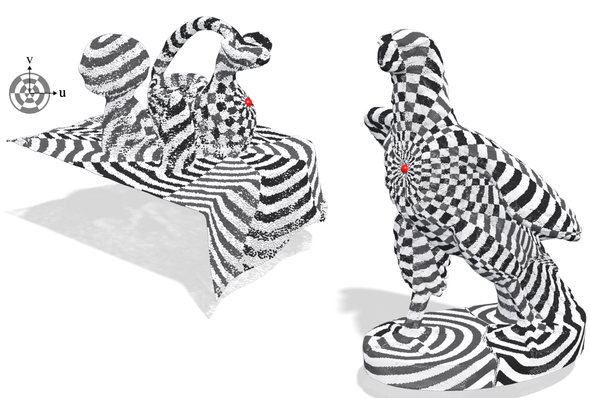


Figure 15: ***Point Cloud Scans.*** *Our method generalizes naturally to point clouds.*

Figure 16: ***Polygonal Meshes.*** *Our method generalizes naturally to quad meshes, and more general polygonal meshes.*

Decaling The approximation of logarithmic maps computed by our method, rooted at a chosen point p, provide ideal parameterizations for decaling applications where one is interested in applying detailed textures or displacements not only locally, but well outside the injectivity radius of p. Their inherent geometric structure allows textures, particularly those designed with radial or circular features, to be mapped accurately and predictably onto curved surfaces (Fig. 17). Here, the improved accuracy of our method near the source (Fig. 6) is of great importance, as distortions near the center of a decal are visually displeasing.

Figure 17: ***Decaling Bowls.*** *Using the logarithmic map, we can effortlessly place a geometric pattern onto a bowl.*

UV Flattening Beyond decaling, surface parameterization is crucial for tasks requiring a mapping of the entire surface to a 2D domain, often referred to as UV flattening. Our approach offers flexibility in this context. Even on some surfaces with a highly nontrivial topology we can generate a low distortion parameterization ϕ_p originating from just a single source point p that covers the entire surface (Fig. 18, *left*). In general, the single-source global parameterizations inevitably suffer from high distortion far from the source, limiting their utility for applications demanding low metric error everywhere. For generating high-quality parameterizations suitable for traditional texturing workflows, a patchbased approach is often more effective. We leverage AHM within such a framework as follows: select a set of source points $\{p_i\}$, compute their geodesic Voronoi diagram $\{C_i\}$ on the surface and define the final parameterization by restricting the logarithmic map ϕ_{p_i} (computed via AHM rooted at p_i) to the corresponding cell C_i. Given approximations of geodesic distance from each source $\{d_i : \mathcal{V} \to \mathbb{R}\}$, we take the geodesic Voronoi cells to be the set of faces closest (measured at the barycenter) to a single source: $C_i := \{f \in \mathcal{F} \mid \sum_{v\in f} d_i(v) \leq \sum_{v\in f} d_j(v)\ \forall j \neq i\}$—we use the heat method to approximate geodesic distances [CWW13]. By combining the local accuracy and robustness of AHM near each source with Voronoi partitioning we can produce a set of low-distortion charts that cover the entire surface (Fig. 18, *right*). Determining an optimal configuration of points automatically (for different measures of optimality) is an interesting question for further research—one simple possibility would be to use the landmarks generated by furthest point sampling proposed in [SSC19, Sec. 8.5]

Stroke-Aligned Parameterizations The parameterization diffusion approach we use to generate geodesic polar coordinates can be generalized to compute stroke aligned parameterizations, similar to those of [Sch13]. Starting from a curve $\gamma : [0,1] \to M$ endowed with a parameterization $z : [0,1] \to \mathbb{R}^n$, we can extend this parameterization to all of M by making two modifications to the affine heat method. We first construct a frame aligned with γ and extend it to M by parallel transport along shortest geodesics—in the case $n = 2$, the frame along the curve is determined completely by γ (*i.e.*, $\gamma', J\gamma'$), but in higher codimension this frame needs to be specified by the user—the natural choice would be the Bishop frame computed via parallel transport along the curve [BWR*08].

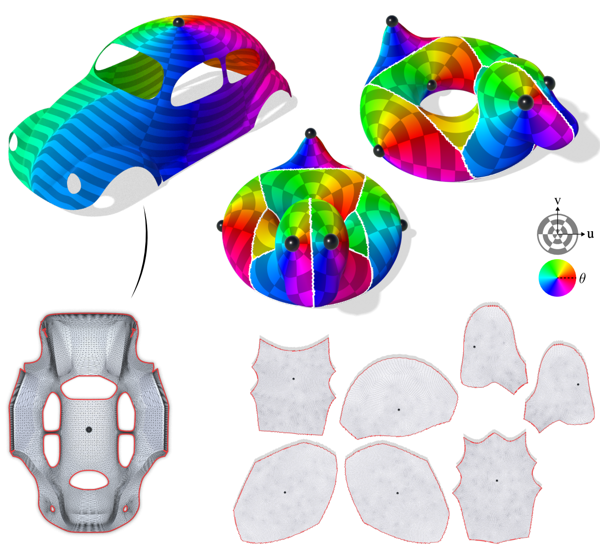


Figure 18: ***UV Flattening.*** *Our logarithmic maps provide low distortion parameterizations even on complicated surfaces (*left*). On surfaces that do not admit any non-degenerate parameterization without seams, a user can specify a collection of points to compute polar coordinates on the associated Voronoi cells (*right*).*

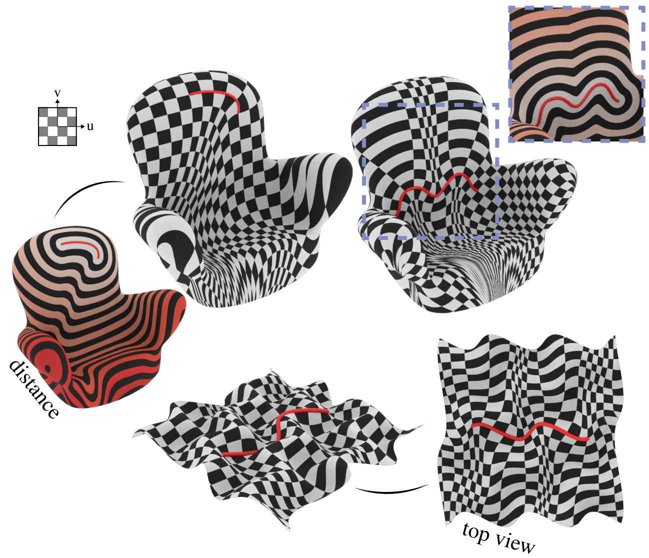


Figure 19: ***Stroke-Aligned Parameterizations.*** *Diffusing a parameterization specified along a curve provides a straightforward method for computing aligned surface parameterizations. Measuring the distance to the curve in UV space provides an estimate of the geodesic distance to the curve.*

We then construct the affine connection Laplacian (Eqn. 15) using the identification induced by this frame, and diffuse the initial parameterization that is now specified along all of γ to obtain the extended parameterization (Fig. 19). In these cases, it is no longer meaningful to visualize the parameterization using a polar checkerboard pattern. If a parameterization is not already given, the natural choice for an open curve is to consider the arclength parameterization of the curve along the $+u$-axis. For closed curves, a more natural choice would be a parameterization along a circle. More generally, the procedure just described generalizes to extend a parameterization defined on a submanifold of arbitrary codimension to the entire manifold (*e.g.*, extending a parameterization of an entire surface to an aligned parameterization of the ambient space). We leave the exploration of the utility of such surface aligned parameterizations to future work.

Distance and its Gradient AHM$_\ell$ and AHM$_a$ define novel heat based approximations of geodesic distance as the length of their approximation of the logarithmic map $|\phi|$. In the localized variant, the distance approximation is equivalently given by the length of the radial field $|\mathsf{Y}|$. In the vicinity of the source point, we observed that these approximations of geodesic distance are more accurate than even heat methods tailored for geodesic distance (Fig. 20). At the cut locus, however, these approximations are inaccurate as our method approximates a ground truth parameterization that is not continuous.

The localized variant of the affine heat method also computes a radial field Y from a point p as an intermediate step in computing the logarithmic map $\log_p$. This radial vector is equal to half the gradient of squared geodesic distance, and is of independent interest

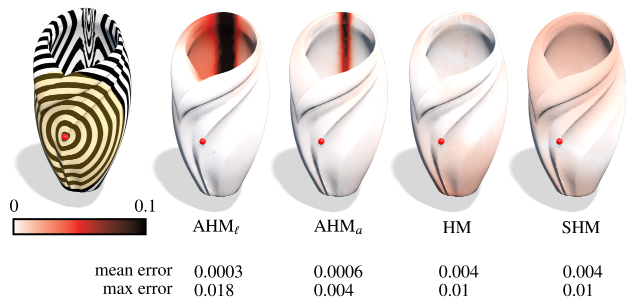


Figure 20: ***Geodesic Distance.*** *Our distance approximations are more accurate in a neighborhood around the source (highlighted in orange) than both the original heat method of [CWW13] (HM) and the signed heat method of [FC24] (SHM). We measure the error against the exact polyhedral distance in the highlighted region.*

as it is the evaluation of the logarithmic map to p from every other point $q \in M$:

$$Y_q = -\log_q(p).$$

Unlike previous approaches, AHM_ℓ computes this radial vector field without differentiating the distance approximation, or approximating the Hausdorff measure supported on a small (or infinitesimal) circle (Fig. 21).

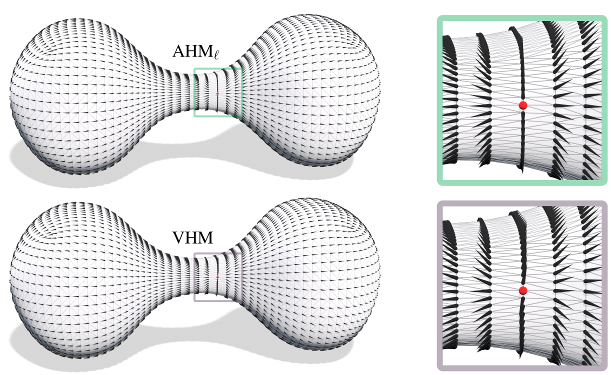


Figure 21: ***Radial Vector Fields.*** *On anisotropic meshes, normalizing the radial vector field approximated via affine diffusion (AHM_ℓ) can produce a noticeably more isotropic approximation than that computed via vector diffusion* à la *[SSC19].*

6. Evaluation

We consider a diverse set of numerical experiments to evaluate the performance, accuracy, and tradeoffs of our methods.

Performance The affine heat method's runtime is dominated by solving two sparse linear systems. AHM_ℓ requires factoring one matrix, reusable for the two linear systems and across multiple sources—this results in $7-30\times$ speedups for AHM_ℓ on successive solves (Fig. 22, *bottom left*). The source-dependence of the connection in AHM_a prevents prefactoring of one matrix; reusing the symbolic factorization yields $3-10\times$ speedups on successive

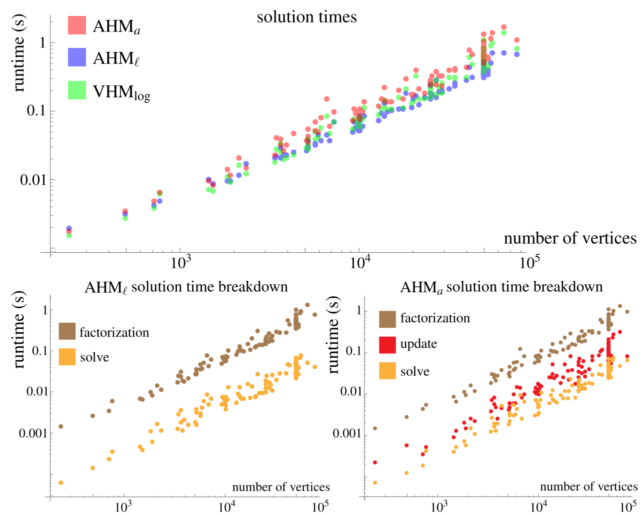
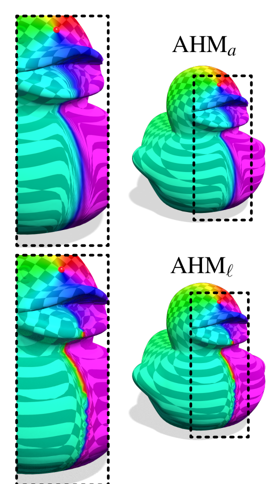


Figure 22: ***Runtime.*** Top: *simulating affine diffusion is slightly more expensive than vector diffusion.* Bottom row: *The full factorization of AHM_ℓ can be reused to speedup successive solves. AHM_a additionally requires updating the numerical factorization of a single matrix (red) when the source is changed.*

solves (Fig. 22, *bottom right*). On the [MZ13] dataset, AHM_ℓ is on average 27% slower than VHM_{log}, while AHM_a is 70% slower due to requiring the solution of an additional linear system (Fig. 22, *top*). On a typical model of about 100k vertices, AHM_a takes $\approx$ 1s. All heat methods share the same asymptotic time complexity dominated by the time it takes to solve connection Laplace-type equations. Our code is implemented in C++; timings are reported on an Intel i7-14700K CPU.



Localized vs. Adaptive The localized variant of the affine heat method is faster, while the adaptive variant delivers higher accuracy. However, they are almost indistinguishable within the injectivity radius, as quantified by difference in the parameterization measured and visualized in Fig. 8. They primarily differ in their behavior near the cut locus (see inset)—the estimate of the angular coordinate of the logarithmic map estimated by AHM_a is smoother and provides an accurate measure of the direction back to the source. To assess convergence we measure the error against the analytical solution on a refined sequence of discretizations of the unit sphere (Fig. 24)[§]. Both variants of the affine heat method appear

§ The analytical solution is obtained by combining the exact geodesic distance function $d(x,y) = \cos^{-1}(x \cdot y)$ with the angle formed by the geodesic from the source p and a fixed reference direction

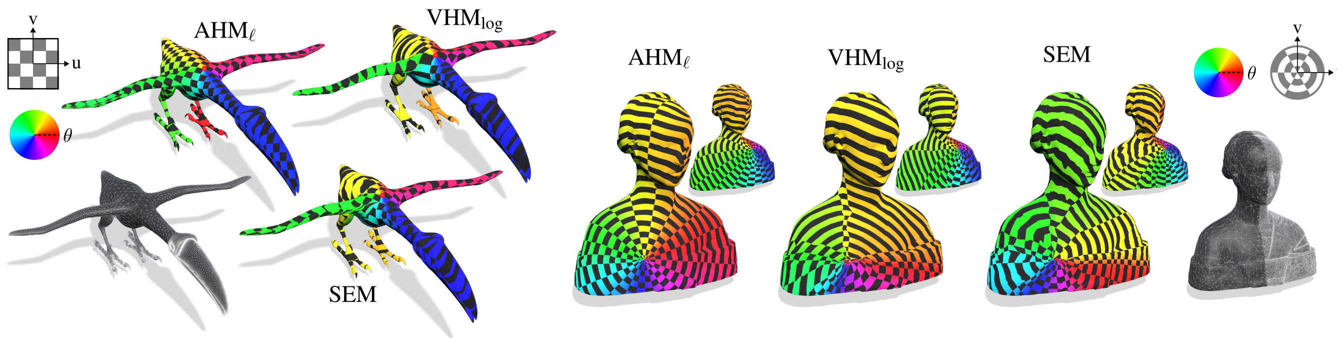


Figure 23: ***Robustness.*** *Our affine method more robustly produces accurate logarithmic maps than prior methods based on heat diffusion (VHM$_{log}$, SEM), especially when the the underlying geometry is coarsely tesselated. All methods use diffusion operators built on the intrinsic Delaunay triangulation.*

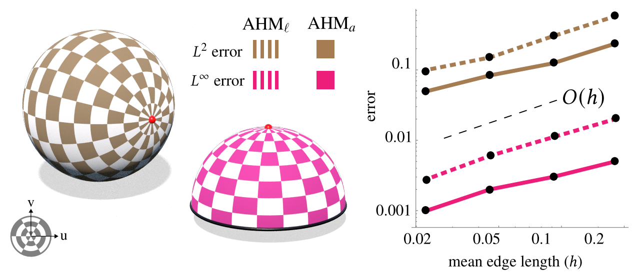


Figure 24: ***Convergence on S^2.*** *We observe a linear convergence rate to the analytical solution on S^2 in both the L^2 and L^∞ norms.*

to exhibit a linear convergence rate $O(h)$, where h is the mean edge length. We measure the convergence in both L^2 and L^∞. Due to the inherent discontinuity of the true logarithmic map at the cut locus (antipodal point), we measure the L^∞ error only in the hemisphere while measuring the L^2 error over the entire surface. The adaptive variant does produce, to a slight extent, a more accurate parameterization.

Comparisons with Prior Work To evaluate the numerical accuracy of our method we compare the parameterization accuracy with existing methods for computing logarithmic maps: namely, the discrete exponential maps (DEM) from [SGW06], the smoothed exponential maps (SEM) from [HA19], and the vector heat method log maps (VHM$_{log}$)from [SSC19]. We also utilize the intrinsic Delaunay triangulation when applying VHM$_{log}$ and SEM.

While metric distortion is not an explicit objective of any of these methods, it is an important practical indicator of the usability of the parameterization for downstream tasks. In Fig. 2, we compute the metric distortion of the parameterizations computed both by our methods and by the previous work. The affine heat methods outperform the alternatives. On each triangle we measure the distortion as

$$\mathcal{D} = \max\left({}^1\!/\sigma_1, \sigma_2\right), \tag{18}$$

where $\sigma_1 \leq \sigma_2$ are the singular values of the gradient of the parameterization inside the triangle. The distortion of a triangle is minimized if the parameterization is isometric with $\mathcal{D} = 1$. The figure shows that near the source point VHM$_{log}$ and SEM introduce local distortion. DEM, on the other hand, resolves the parameterization directly near the source point, but is overly distorted across the surface. Compared with these methods, AHM$_\ell$ and AHM$_a$ produce parameterizations with less distortion globally, and with smoother distortion distributions.

Fig. 25 highlights that while the discrete exponential maps method of [SGW06] provides accurate coordinates locally, distortion accumulates significantly far from the source point as indicated by the irregular checkerboard pattern. In contrast, AHM$_a$ computes a smooth parameterization, even up to and including the cut locus. Even in the Euclidean setting where the ground truth parameterization is trivial, Figs. 6 and 26 shows that VHM$_{log}$ and SEM produce parameterization with distortion concentrated near the source point, respectively. Both AHM$_\ell$ and AHM$_a$ not only resolve this local distortion, but reproduces the trivial solution over the entire mesh. Finally, we observe that even AHM$_\ell$ exhibits strong robustness to various forms of low-quality tessellations where both VHM$_{log}$ and SEM struggle (Fig. 23).

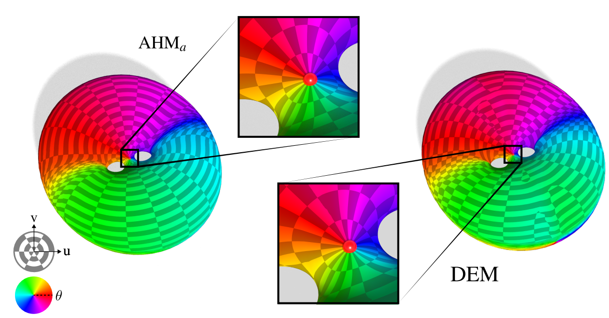


Figure 25: ***Discrete Exponential Maps Comparison.*** *The DEM approach of [SGW06] produces distorted parameterizations as the deviations in the normals accumulate.*

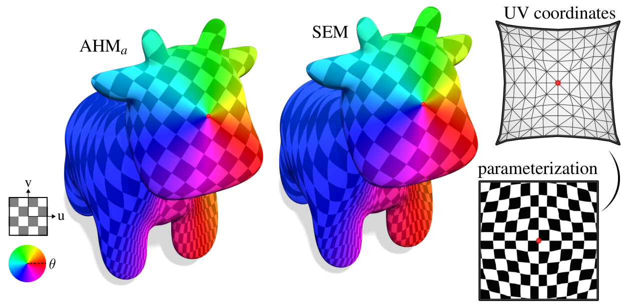
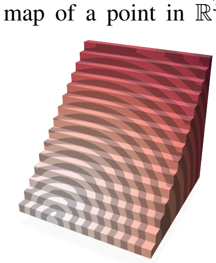


Figure 26: ***Smoothed Exponential Maps Comparison.*** *The heat method approach for computing smoothed exponential maps of [HA19] also produces distorted parameterizations, even in the trivial case of a flat grid.*

6.1. Limitations and Future Work

While our method offers significant advantages in terms of parameterization quality relative to prior work, it also presents certain challenges and suggests avenues for future research. The main computational challenge of the adaptive variant is that the dependence of the connection Laplacian operator on the source point implies that the symbolic factorization needs to be recomputed across multiple solves. While the localized affine heat method we presented does not suffer from this problem, finding a reformulation that computes identical parameterizations using a fixed universal operator would be desirable, if one exists. Inspired by the fact that the heat method can be understood as the first iteration in a fixed point method for solving a nonlinear PDE [BF15], we are also interested in understanding whether our localized and adaptive variants admit a similar formulation. Turning our attention to controlling the local parameterizations, we note that since the principal symbol of a Laplacian is the Riemannian metric, vector heat methods can only ever compute parallel transport along shortest geodesics as defined by the metric (or equivalently the cotan weights). In particular, such a formulation cannot be generalized to compute the logarithmic map associated to an arbitrary affine connection. Nevertheless, some control over the geodesics can be obtained by changing the underlying metric (*e.g.*, using a conformally equivalent metric). Generalizing heat methods beyond the setting of induced metrics and developing intuitive handles for modifying the notion of geodesics would make for interesting future work. Finally, while we presented applications of computing surface logarithmic maps, our methodology generalizes to higher-dimensional domains. For example, we can use the affine heat method on a regular grid to compute the logarithmic map of a point in $\mathbb{R}^3$ (see inset for the distance component computed this way)—in this case, the resulting parameterization is trivial, but more complicated boundary conditions or sources (*i.e.*, strokes and surfaces) can produce more interesting 3D parameterizations—we leave applications of our method in higher dimensions to future work.



7. Conclusion

We introduced the affine heat method for computing logarithmic maps on surfaces. Utilizing Euclidean transformations to encode the parameterization in the parallel sections of a connection, we can efficiently approximate the parameterization via the solution of an affine diffusion equation. A localized and adaptive variant of the affine heat method were developed, providing practitioners with a tradeoff between computational cost and improved parameterization quality. Both variants of the algorithm are simple to implement, only requiring the solution of Laplace-type linear systems. Our approach generalizes to producing stroke-aligned parameterizations, and can be used to cover a surface with parameterized patches procedurally. Comparisons with prior work demonstrate that we obtain higher quality results by diffusing the parameterization directly instead of by computing the angular and distance components separately and then assembling the resulting parameterization.

8. Acknowledgements

Thanks to Chris Rydalch and Keenan Crane for early discussions, and Lorena E'Vers for the model in Fig. 12 and the dog skull model in Fig. 16.

References

[AFSHA24] AKALIN K., FINNENDAHL U., SORKINE-HORNUNG O., ALEXA M.: Mesh parameterization meets intrinsic triangulations. *Comp. Graph. Forum 43*, 5 (2024). 2

[BF15] BELYAEV A. G., FAYOLLE P.-A.: On variational and PDE-based distance function approximations. *Comp. Graph. Forum 34*, 8 (2015). 2, 15

[BGV92] BERLINE N., GETZLER E., VERGNE M.: *Heat kernels and Dirac operators*. 1992. 5

[BHKB20] BUNGE A., HERHOLZ P., KAZHDAN M., BOTSCH M.: Polygon Laplacian made simple. *Comp. Graph. Forum 39*, 2 (2020). 11

[BS07] BOBENKO A., SPRINGBORN B.: A discrete Laplace-Beltrami operator for simplicial surfaces. *Discrete Comput. Geom. 38*, 4 (2007). 10

[BWR*08] BERGOU M., WARDETZKY M., ROBINSON S., AUDOLY B., GRINSPUN E.: Discrete elastic rods. In *ACM SIGGRAPH 2008 papers*. 2008. 12

[BYF*19] BUDNINSKIY M., YIN G., FENG L., TONG Y., DESBRUN M.: Parallel transport unfolding: a connection-based manifold learning approach. *SIAM J. Appl. Algebra Geom 3*, 2 (2019). 2

[CWW13] CRANE K., WEISCHEDEL C., WARDETZKY M.: Geodesics in heat: a new approach to computing distance based on heat flow. *ACM Trans. Graph. 32*, 5 (2013). 2, 5, 10, 12, 13

[FC24] FENG N., CRANE K.: A heat method for generalized signed distance. *ACM Trans. Graph. 43*, 4 (2024). 2, 3, 4, 5, 11, 13

[GKK*24] GARANZHA V., KAPORIN I., KUDRYAVTSEVA L., PROTAIS F., SOKOLOV D.: In the quest for scale-optimal mappings. *ACM Trans. Graph. 43*, 1 (2024). 2

[GSC21] GILLESPIE M., SHARP N., CRANE K.: Integer coordinates for intrinsic geometry processing. *ACM Trans. Graph. 40*, 6 (2021). 11

[HA19] HERHOLZ P., ALEXA M.: Efficient computation of smoothed exponential maps. In *Comp. Graph. Forum* (2019), vol. 38, Wiley Online Library. 2, 14, 15

[Jos08] JOST J.: *Riemannian geometry and geometric analysis*. 2008. 4

[Kar14] KARCHER H.: Riemannian center of mass and so called karcher mean. *arXiv preprint arXiv:* (2014). 11

[KCPS13] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Globally optimal direction fields. *ACM Trans. Graph. 32*, 4 (2013). 3

[KLS03] KHODAKOVSKY A., LITKE N., SCHRÖDER P.: Globally smooth parameterizations with low distortion. *ACM Trans. Graph. 22*, 3 (2003). 2

[KP16] KNÖPPEL F., PINKALL U.: *Advances in Discrete Differential Geometry*. 2016, ch. Complex line bundles over simplicial complexes and their applications. 3

[LDL*22] LIN W., DURUISSEAUX V., LEOK M., NIELSEN F., KHAN M. E., SCHMIDT M.: Practical structured Riemannian optimization with momentum by using generalized normal coordinates. In *NeurIPS Workshop on Symmetry and Geometry in Neural Representations* (2022). 2

[LTGD16] LIU B., TONG Y., GOES F. D., DESBRUN M.: Discrete connection and covariant derivative for vector field analysis and design. *ACM Trans. Graph. 35*, 3 (2016). 3

[ML24] MADAN A., LEVIN D.: Local surface parameterization via geodesic splines. *arXiv preprint arXiv:2410.06330* (2024). 2

[MZ13] MYLES A., ZORIN D.: Controlled-distortion constrained global parametrization. *ACM Trans. Graph. 32*, 4 (2013). 13

[SC20] SHARP N., CRANE K.: A Laplacian for nonmanifold triangle meshes. *Comp. Graph. Forum 39*, 5 (2020). 11

[Sch13] SCHMIDT R.: Stroke parameterization. *Comp. Graph. Forum 32*, 2 (2013). 2, 12

[SGC21] SHARP N., GILLESPIE M., CRANE K.: Geometry processing with intrinsic triangulations. 3

[SGW06] SCHMIDT R., GRIMM C., WYVILL B.: Interactive decal compositing with discrete exponential maps. *ACM Trans. Graph. 25*, 3 (2006). 2, 14

[SSC18] SOLIMAN Y., SLEPČEV D., CRANE K.: Optimal cone singularities for conformal flattening. *ACM Trans. Graph. 37*, 4 (2018). 2

[SSC19] SHARP N., SOLIMAN Y., CRANE K.: The vector heat method. *ACM Trans. Graph. 38*, 3 (2019). 2, 4, 5, 6, 8, 10, 11, 12, 13, 14

[Wei37] WEISZFELD E.: Sur le point pour lequel la somme des distances de n points donnés est minimum. *Tohoku Math. J. 43* (1937). 11

Appendix A: Adapted Geodesic Frames

For Riemannian manifolds, parallel transport with respect to the Levi-Civita connection ∇ along geodesics is essentially characterized by the property that the angle between the parallel transported vector and the tangent vector of the curve remains constant: since the Levi-Civita connection is a metric connection, if X is parallel along a geodesic γ we have that

$$\langle X, \gamma' \rangle' = \langle \nabla_{\gamma'} X, \gamma' \rangle + \langle X, \nabla_{\gamma'} \gamma' \rangle = 0.$$

This gives us an approach to construct the frame Φ on the tangent bundle suitable for the construction of logarithmic maps.

Definition 1 A *geodesic frame adapted to* p is a collection of n vector fields $U_1, \dots, U_n \in \Gamma(TM)$ obtained by parallel transport along shortest geodesics from p of a fixed frame $(U_1)_p, \dots, (U_n)_p \in T_pM$.

More precisely, we fix an orthonormal frame $U_p^1, \dots, U_p^n \in T_pM$ at p to M that we extend to vector fields $U^1, \dots, U^n$ via parallel transport along the geodesics emanating from p. We then define $\Phi : TM \to \mathbb{R}^n$ via

$$X \in T_pM \mapsto \begin{pmatrix} \langle U^1, X \rangle \\ \vdots \\ \langle U^n, X \rangle \end{pmatrix} \in \mathbb{R}^n. \tag{19}$$

Notice that the identification Φ depends on p, in addition to the initial frame in T_pM.

Appendix B: Planar Domains

Since the affine parallel transport is trivial on a planar triangle mesh, we can show that the affine heat method is gauge-equivalent to a scalar diffusion equation:

Lemma 3 Let $\mathcal{K}$ be a planar triangle mesh with vertex positions $z : \mathcal{V} \to \mathbb{R}^2$. Let $\mathsf{L}^{\overline{\nabla}}$ be the affine connection Laplacian associated to $r^{\overline{\nabla}}$ and $\mathsf{L}^{\mathbb{R}^2}$ be the componentwise scalar cotan Laplacian. For any right hand side $\mathbf{b}, b$, the solution of the equation

$$(\mathsf{M} + \tau \mathsf{L}^{\overline{\nabla}}) \begin{pmatrix} \mathbf{x} \\ \lambda \end{pmatrix} = \begin{pmatrix} \mathbf{b} \\ b \end{pmatrix}$$

is given by $\mathbf{x}_i = \tilde{\mathbf{x}}_i + z_i$ and $\lambda_i = \tilde{\lambda}_i$ where $\tilde{\mathbf{x}}, \tilde{\lambda}$ are obtained as soltuions of

$$(\mathsf{M} + \tau \mathsf{L}^{\mathbb{R}^3}) \begin{pmatrix} \tilde{\mathbf{x}} \\ \tilde{\lambda} \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{b}} \\ b \end{pmatrix},$$

with $\tilde{\mathbf{b}}_i = \mathbf{b}_i - z_i$.

Proof Fix a vertex $i \in \mathcal{V}$. Using the fact that $e_{ij} = z_j - z_i$ we obtain

$$\begin{aligned}
\begin{pmatrix} \mathbf{b}_i \\ b_i \end{pmatrix} &= \mathsf{M}_{ii} \begin{pmatrix} \mathbf{x}_i \\ \lambda_i \end{pmatrix} + \tau \sum_{ij} w_{ij} \left(\begin{pmatrix} \mathbf{x}_i \\ \lambda_i \end{pmatrix} - r_{ji}^{\overline{\nabla}} \begin{pmatrix} \mathbf{x}_j \\ \lambda_j \end{pmatrix} \right) \\
&= \mathsf{M}_{ii} \begin{pmatrix} \mathbf{x}_i \\ \lambda_i \end{pmatrix} + \tau \sum_{ij} w_{ij} \left(\begin{pmatrix} \mathbf{x}_i \\ \lambda_i \end{pmatrix} - \begin{pmatrix} \mathbf{x}_j + z_i - z_j \\ \lambda_j \end{pmatrix} \right) \\
&= \begin{pmatrix} \mathrm{id} & z_i \\ 0 & 1 \end{pmatrix} \left[\mathsf{M}_{ii} \begin{pmatrix} \mathbf{x}_i - z_i \\ \lambda_i \end{pmatrix} + \tau \sum_{ij} w_{ij} \left(\begin{pmatrix} \mathbf{x}_i - z_i \\ \lambda_i \end{pmatrix} - \begin{pmatrix} \mathbf{x}_j - z_j \\ \lambda_j \end{pmatrix} \right) \right] \\
&= \begin{pmatrix} \mathrm{id} & z_i \\ 0 & 1 \end{pmatrix} \left[\mathsf{M}_{ii} \begin{pmatrix} \tilde{\mathbf{x}}_i \\ \tilde{\lambda}_i \end{pmatrix} + \tau \sum_{ij} w_{ij} \left(\begin{pmatrix} \tilde{\mathbf{x}}_i \\ \tilde{\lambda}_i \end{pmatrix} - \begin{pmatrix} \tilde{\mathbf{x}}_j \\ \tilde{\lambda}_j \end{pmatrix} \right) \right].
\end{aligned}$$

□

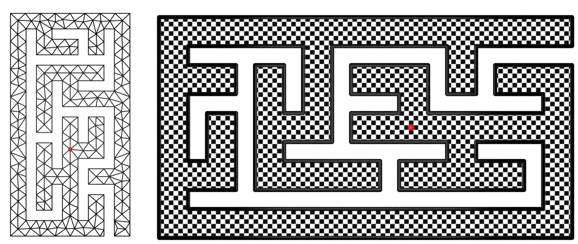


Figure 27: ***Mazes.*** *The affine heat method provides an extension of the logarithmic map outside the region where the exponential map is invertible. On planar domains, the extension produces the identity parameterization.*

The exact recovery of (a translate of) the vertex positions using the discrete affine heat method on planar domains follows since for the right hand side $\tilde{\mathbf{b}}_i = 0$ and $b = \delta_v$ for some vertex $v \in \mathcal{V}$ that the solution of the scalar diffusion is given by $\tilde{\mathbf{x}}_i = 0$. Therefore, after translation we obtain $\mathbf{x}_i = z_i$ for all vertices $i \in \mathcal{V}$. Observe that this property does not depend on the choice of edge weights nor the choice of diffusion time τ. Fig. 27 shows that our algorithm reproduces the trivial parameterization even on a complicated planar domain.