

# Discrete Periodic Geodesics in a Surface

Anders Linnér and Robert Renka

## CONTENTS

- 1. Introduction
- 2. Specifics
- 3. New Method
- 4. Examples
- 5. Summary
- References

---

An alternative to the traditional curve-straightening flow on periodic curves in surfaces is introduced. The implementation of this flow produces periodic geodesics in minutes rather than hours. The flow is also simpler to initiate since its use of a penalty method permits initial curves that are not necessarily in the surface. Compact and noncompact examples are provided as well as examples with trivial and nontrivial free homotopy classes. The explicit curve-straightening flow on circles in Euclidean space is derived to help check the consistency of the implementations.

---

## 1. INTRODUCTION

### 1.1 Geodesics

Geodesics, i.e., the non-Euclidean generalization of straight lines, continue to be the subject of much research. The initial value problem that produces a so-called geodesic ray requires the solution of a second order nonlinear differential equation that typically cannot be solved explicitly. Fortunately, it is straightforward to numerically generate approximate geodesic rays. The problem of finding a length-minimizing curve connecting two given points also leads to geodesics. In this case, things are more involved since it is now necessary to solve a boundary value problem. Neither of these two problems is as demanding as the problem of numerically generating a periodic geodesic. A periodic geodesic is a parameterized curve  $\gamma$  of constant speed characterized by the fact that its acceleration has no tangential part and  $\gamma$  satisfies periodic boundary conditions.

### 1.2 Alternative Flows

Several different flow methods, such as a variety of curve-shortening [Grayson 89] and curve-straightening flows [Langer and Singer 87, Linnér 91], have been proposed as means to accomplish the task of producing periodic geodesics. As far as we know, only one of these methods, traditional curve-straightening, has an implementation capable of generating periodic geodesics in general surfaces that are diffeomorphic to a sphere; see

2000 AMS Subject Classification: Primary 49M25, 53C22, 58E10, 65K10

Keywords: curve-straightening, elastic energy, geodesic curvature, periodic geodesic, Sobolev gradient

[Linnér 04]. The tendency of the curve-shortening flow to shrink an initial periodic curve down to a point renders this technique less useful in such surfaces. Ideas to combat this through area-preserving modifications have been explored by Gage [Gage 90], but no general examples have been produced.

### 1.3 Curve-Straightening

The implementation of the curve-straightening flow in [Linnér 04] is interactive. It is also a hybrid since it utilizes both symbolic and numeric capabilities of Mathematica. As a consequence, it is a very time-consuming process to generate a periodic geodesic. The purpose of the present article is to report on a related but different approach now implemented in Fortran.

### 1.4 New Method

This new approach has the following in common with the previous method. It seeks to minimize the elastic energy, given by  $\int_{\gamma} \kappa^2 ds$ , in a suitable space of periodic curves confined to a given surface. Here  $\kappa$  is the so-called geodesic curvature and  $s$  is the arc length parameter. Both methods use Sobolev space gradients as their primary tool. There are different options as to how to deal with the three main difficulties: the constraint to stay in the surface, the demand that the curve is periodic, and the reduction of the elastic energy down to zero.

### 1.5 Existence and Convergence

A strong motivation behind the development of algorithms designed to produce periodic geodesics is the recent comprehensive existence results in the case of sphere-like surfaces. In the works by Franks [Franks 92] and Bangert [Bangert 93] it is shown that there are infinitely many geometrically distinct periodic geodesics in a sphere with an arbitrary Riemannian metric. Since the total squared geodesic curvature is zero at each periodic geodesic, it is very natural to use steepest descent as a tool to locate these global minima. It is, however, not known if the negative gradient trajectories always converge. The example in Section 2 illustrates divergence in the Euclidean plane where there are no periodic geodesics. It is shown in [Linnér 98] that divergence is possible even in the presence of a global minimum. A second motivation behind the development of these tools is the desire to learn more about the convergence or divergence along steepest descent.

### 1.6 Geometric Properties along Steepest Descent

In [Linnér 03] several examples illustrate how a change of Sobolev metric affects geometric properties of curves

along the flow. Little is known about the geometric influence of the Neuberger-Sobolev gradients [Neuberger 97] and the use of conjugate gradient steps in the setting of infinite-dimensional manifolds. The rest of this paper initiates an investigation into this. Few steepest descent flows in infinite-dimensional manifolds are explicit. The case of evolving circles in the Euclidean plane provides one example where the evolution of the length is explicit and not trivial, see Section 2.

### 1.7 Organization

This paper is organized as follows. Section 2 presents the necessary mathematical concepts and fixes the notation. It also contains a description of the curve-straightening flow in the Euclidean plane. The evolution of a unit length circle is described explicitly. Section 3 provides an overview of the implementation of the new method, which is based on the Neuberger-Sobolev gradient and the Polak-Ribiere nonlinear conjugate gradient method. Section 4 provides examples and a discussion of our computational experience. Section 5 summarizes the article by comparing the new method to the previous.

## 2. SPECIFICS

To provide a gentle introduction to the techniques and the notation, we present all the details in the first case considered.

### 2.1 The Euclidean Case

In almost all cases, the flow along the negative gradient trajectories of the elastic energy restricted to periodic curves of arbitrary length is not explicit. In the case of periodic curves in the Euclidean plane, the flow formulas simplify so that at least some trajectories are completely explicit. To see this, consider regular curves  $\gamma : [0, 1] \rightarrow \mathbb{R}^2$  of length  $L_{\gamma}$  parameterized so that  $|\gamma'(s)| = L_{\gamma}$  for all  $s$  in  $[0, 1]$ . There is a unique tangent angle  $\theta : [0, 1] \rightarrow \mathbb{R}$  with  $\theta(0) \in [0, 2\pi)$ , and  $\gamma'(s) = (x(s), y(s)) = L_{\gamma}(\cos \theta(s), \sin \theta(s))$ . The signed curvature  $\kappa : [0, 1] \rightarrow \mathbb{R}$  is given by  $\kappa(s) = \theta'(s)/L_{\gamma}$ . It is now convenient to work with pairs  $(\theta, L)$ , and if necessary integrate to recover  $\gamma$ .

### 2.2 Elastic Energy

The elastic energy is proportional to the quantity

$$J(\theta, L) = \frac{1}{2L} \int_0^1 (\theta'(s))^2 ds.$$

Rotate  $\gamma$  so that the initial tangent is horizontal, and define

$$H_0 = \{ \theta \in W_1^2[0, 1] \mid \theta(0) = 0 \}.$$

This Sobolev space consists of absolutely continuous functions with derivatives in  $L^2[0, 1]$ . Let  $\mathbb{R}^+$  denote the positive real numbers and choose the domain so that  $J : H_0 \times \mathbb{R}^+ \rightarrow \mathbb{R}$ . We disallow curves with a tangent discontinuity by requiring  $\theta(1) = \eta \cdot 2\pi$ , where  $\eta$  is the integer known as the rotation number. The Riemannian structure on  $T(H_0 \times \mathbb{R}^+) = TH_0 \times T\mathbb{R}^+ = H_0 \times \mathbb{R}$  is the traditional

$$\langle (v_\theta, v_L), (w_\theta, w_L) \rangle = \int_0^1 v'_\theta(s)w'_\theta(s)ds + v_L w_L.$$

### 2.3 Gradient

In infinite dimensions it becomes important to distinguish between the “directional” derivative and the gradient. They are related by

$$DJ(\theta, L)(v_\theta, v_L) = \langle \nabla J(\theta, L), (v_\theta, v_L) \rangle,$$

where the right-hand side depends on the choice of Riemannian structure. Since

$$DJ(\theta, L)(v_\theta, v_L) = -v_L \frac{J(\theta, L)}{L} + \frac{1}{L} \int_0^1 \theta'(s)v'_\theta(s)ds,$$

it follows that the two components of the gradient are given by  $\nabla J(\theta, L)_\theta = \theta/L$ , and  $\nabla J(\theta, L)_L = -J(\theta, L)/L$ . Observe that

$$\nabla J(\theta, L)_\theta(1) - \nabla J(\theta, L)_\theta(0) = \theta(1)/L = \eta \cdot 2\pi/L,$$

so for nonzero rotation number the tangent direction will not stay periodic. More generally, it is necessary to project onto the tangent space of the manifold of all periodic curves given by

$$\Omega_\eta = \left\{ (\theta, L) \in H_0 \times \mathbb{R}^+ \mid \begin{array}{l} G_x(\theta, L) = G_y(\theta, L) = 0, \\ \Phi(\theta, L) = \eta \cdot 2\pi \end{array} \right\},$$

where

$$\begin{aligned} G_x(\theta, L) &= L \int_0^1 \cos \theta(s) ds, \\ G_y(\theta, L) &= L \int_0^1 \sin \theta(s) ds, \text{ and} \\ \Phi(\theta, L) &= \theta(1). \end{aligned}$$

The derivatives are given by

$$\begin{aligned} DG_x(\theta, L)(v_\theta, v_L) &= \\ v_L \int_0^1 \cos \theta(s) ds - L \int_0^1 v_\theta(s) \sin \theta(s) ds, \end{aligned}$$

$$DG_y(\theta, L)(v_\theta, v_L) =$$

$$v_L \int_0^1 \sin \theta(s) ds + L \int_0^1 v_\theta(s) \cos \theta(s) ds,$$

and  $D\Phi(\theta, L)(v_\theta, v_L) = v_\theta(1)$ . When these nonlinear functionals are restricted to  $\Omega_\eta$ , their derivatives no longer depend on  $v_L$ , and hence  $\nabla G_x(\theta, L)_L = \nabla G_y(\theta, L)_L = \nabla \Phi(\theta, L)_L = 0$ . Integration by parts further reveals that

$$DG_x(\theta, L)(v_\theta, v_L) = \int_0^1 v'_\theta(s)y(s)ds,$$

and

$$DG_y(\theta, L)(v_\theta, v_L) = - \int_0^1 v'_\theta(s)x(s)ds.$$

The gradients are given by

$$\begin{aligned} \nabla G_x(\theta, L)_\theta &= \int_0^s y(u)du, \\ \nabla G_y(\theta, L)_\theta &= - \int_0^s x(u)du, \text{ and} \end{aligned}$$

$$\nabla \Phi(\theta, L)_\theta = s.$$

### 2.4 Projection

The task at hand is to determine the scalar fields  $\lambda_x, \lambda_y$  and  $\mu$  so that  $\nabla J^\pi := \nabla J - \lambda_x \nabla G_x - \lambda_y \nabla G_y - \mu \nabla \Phi \in T\Omega_\eta$ . Specifically, one needs

$$\begin{aligned} \left( \frac{\theta(s)}{L} - \lambda_x \int_0^s y(u)du \right. \\ \left. + \lambda_y \int_0^s x(u)du - \mu s, -\frac{J(\theta, L)}{L} \right) \in T\Omega_\eta. \end{aligned}$$

Observe that, due to the second component of the projected gradient, there are no periodic critical points, and hence each negative gradient trajectory must diverge. Since the values of the three functionals are constant on  $\Omega_\eta$ , it must be that the three derivatives vanish in the directions given by  $T\Omega_\eta$ . This leads to three linear equations that, in the  $H_0$  component, correspond to the matrix equation

$$\begin{bmatrix} DG_x(\int_0^s y) & -DG_x(\int_0^s x) & DG_x(s) \\ DG_y(\int_0^s y) & -DG_y(\int_0^s x) & DG_y(s) \\ \int_0^1 y & -\int_0^1 x & 1 \end{bmatrix} \begin{bmatrix} \lambda_x \\ \lambda_y \\ \mu \end{bmatrix} = \begin{bmatrix} DG_x(\frac{\theta}{L}) \\ DG_y(\frac{\theta}{L}) \\ \eta \cdot \frac{2\pi}{L} \end{bmatrix},$$

where most arguments are suppressed.

### 2.5 Steepest Descent

The matrix equation simplifies to

$$\begin{bmatrix} \int_0^1 y^2 & -\int_0^1 xy & \int_0^1 y \\ -\int_0^1 xy & \int_0^1 x^2 & -\int_0^1 x \\ \int_0^1 y & -\int_0^1 x & 1 \end{bmatrix} \begin{bmatrix} \lambda_x \\ \lambda_y \\ \mu \end{bmatrix} = \begin{bmatrix} \int_0^1 \kappa y \\ -\int_0^1 \kappa x \\ \eta \cdot \frac{2\pi}{L} \end{bmatrix},$$

where the left-hand side involves the center of mass and second order inertia. The right-hand side is related to the so-called Steiner curvature centroid. The evolution along the negative gradient trajectory satisfies

$$\frac{d}{d\tau}(\theta_\tau, L_\tau) = -\nabla^\pi J(\theta_\tau, L_\tau)$$

in flow time  $\tau$ . This flow equation typically cannot be solved explicitly. There is, nonetheless, at least one example where things work out completely. Suppose the initial curve of the flow is a circle of unit length so that  $\theta_0 = 2\pi s$ ,  $L_0 = 1$ , and  $\eta = 1$ . A routine, albeit lengthy, calculation shows that  $(\lambda_x)_0 = (\lambda_y)_0 = 0$ , and  $\mu_0 = 2\pi$  initially. It follows that  $\nabla^\pi J(\theta_\tau, L_\tau)_\theta = 0$ , and hence  $(\lambda_x)_\tau = (\lambda_y)_\tau = 0$  with  $\mu_\tau = 2\pi/L_\tau$ . The curve evolves as a circle, i.e.,  $\theta_\tau = \theta_0$  for all times  $\tau$ . Since  $\nabla^\pi J(\theta_\tau, L_\tau)_L = -J(\theta_\tau, L_\tau)/L_\tau$ , it follows that the length evolves as the solution to the following initial value problem:

$$\frac{d}{d\tau} L_\tau = \frac{2\pi^2}{L_\tau^2}, \quad L_0 = 1.$$

The solution is  $L_\tau = \sqrt[3]{1 + 6\pi^2\tau}$ .

### 2.6 General Surfaces

The rest of this article concerns general regular surfaces of the form  $M = \{p \in \mathbb{R}^3 \mid f(p) = 0\}$  where  $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ , and  $\nabla f(p) \neq 0$  for all  $p$  in  $M$ . Observe, that  $\kappa$  now stands for the geodesic curvature, and the angle  $\theta$  measures the signed difference between the tangent direction and a reference direction that is parallel-transported along the curve. When  $\gamma$  is a curve of unit speed in  $M$  and  $n = \nabla f / |\nabla f|$  is the unit normal to the surface, then the following holds:

$$|\kappa| = |\gamma''(s) - \gamma''(s) \cdot n(\gamma(s))n(\gamma(s))|.$$

The implementation described in [Linnér 04] handles this general case, but at a sluggish pace. For instance, the explicit case described in the previous paragraph takes 100 steps in 6 minutes using Mathematica 5.0 on a Windows PC running an AMD Athlon at 1.2-GHz. A more complicated surface with nonzero Hessian  $Hf$  and derivative  $DHf$  slows the method to about an hour. Incidentally, the circles associated with the pair  $(\theta_\tau, L_\tau)$  now

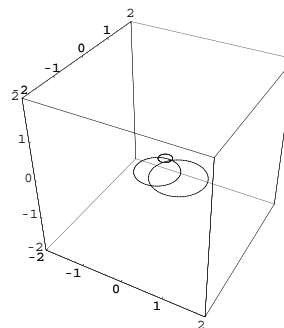


FIGURE 1. Circles of varying length in the Euclidean plane.

spin around the fixed initial point as their radii grow; see Figure 1. Unlike in the special explicit case, here the tangent angle is measured with respect to the parallel-transported initial tangent vector. The initial tangent vector is free to change in this general case, and it does so. This spinning is an example of a general effect due to the choice of Riemannian structure; see [Linnér 03] for details.

### 3. NEW METHOD

In this section we describe a new method for constructing a discrete approximation to a periodic geodesic in a regular surface  $M$ . The method is based on Neuberger’s theory of Sobolev gradients [Neuberger 97], and is similar to the methods used in [Renka 03] and [Renka 04a]. The primary innovation here is the treatment of nonlinear equality constraints. We found that a simple penalty method was remarkably effective, at least for our test cases.

As mentioned in the introduction, curves will now be treated without the help of the tangent angle. As a consequence, the length of the curve is no longer explicit. The curves are parameterized by arc length. Specifically, denote by  $\gamma : [0, L] \rightarrow M$  a parametric representation of a unit speed curve in  $M$  with total length  $L$ :  $f(\gamma(s)) = 0$  and  $|\gamma'(s)| = 1$  for all  $s$  in  $[0, L]$ . The elastic energy based on the geodesic curvature of  $\gamma$  is proportional to the  $L_2$ -norm squared of the tangential curvature component:

$$J(\gamma) = \int_0^L |\mathbf{K}_g(s)|^2 ds,$$

where

$$\mathbf{K}_g(s) = \gamma''(s) - \langle \gamma''(s), n(\gamma(s)) \rangle n(\gamma(s))$$

is a vector equal to the projection of the acceleration vector  $\gamma''(s)$  onto the orthogonal complement of the surface

normal  $n$  at  $\gamma(s)$ . The  $J$  here has a different domain compared to before and the factor  $1/2$  is dropped. A geodesic is characterized by  $J(\gamma) = 0$ . The periodic end conditions are  $\gamma(0) = \gamma(L)$  and  $\gamma'(0) = \gamma'(L)$ .

### 3.1 Discretization

Let  $\gamma$  now denote the polygonal curve defined by a cyclically ordered sequence of  $m$  vertices  $\gamma_i, i = 1, \dots, m$ . Note that our discrete representation of  $\gamma$  includes the derivative approximations defined below along with the vertices. We denote segment lengths by  $\Delta s_i$ , midpoint unit tangent vectors by  $\Delta \gamma_i$ , and vertex curvature vectors by  $\Delta^2 \gamma_i$ :

$$\begin{aligned}\Delta s_i &= \|\gamma_i - \gamma_{i-1}\|, \\ \Delta \gamma_i &= \frac{\gamma_i - \gamma_{i-1}}{\Delta s_i},\end{aligned}$$

and

$$\Delta^2 \gamma_i = \frac{\Delta \gamma_{i+1} - \Delta \gamma_i}{(\Delta s_i + \Delta s_{i+1})/2}$$

for  $i = 1, \dots, m$ , where  $\gamma_0 = \gamma_m$ ,  $\Delta s_{m+1} = \Delta s_1$ , and  $\Delta \gamma_{m+1} = \Delta \gamma_1$ .

The integral defining geodesic curvature is approximated by the composite trapezoidal rule giving the discretized energy functional

$$E(\gamma) = .5 \sum_{i=1}^m (\|\mathbf{u}_i\|^2) \Delta a_i,$$

where

$$\begin{aligned}\Delta a_i &= \Delta s_i + \Delta s_{i+1}, \\ \mathbf{u}_i &= (I - \mathbf{n}_i \mathbf{n}_i^t) \Delta^2 \gamma_i,\end{aligned}$$

and

$$\mathbf{n}_i = \nabla f(\gamma_i) / \|\nabla f(\gamma_i)\|.$$

### 3.2 Minimization Problem

We have the constrained optimization problem of minimizing  $E(\gamma)$  subject to the end conditions and the  $m$  nonlinear constraints  $F(\gamma) = 0$ , where the  $i$ th component of  $F(\gamma)$  is  $f(\gamma_i)$  for  $i = 1, \dots, m$ . The periodic end conditions are implicit in the expressions for  $\Delta s_1$ ,  $\Delta \gamma_1$ , and  $\Delta^2 \gamma_m$ .

For the nonlinear constraints, we add a penalty function to the energy functional, giving

$$\phi(\gamma) = E(\gamma) + .5wF(\gamma)^t F(\gamma),$$

where  $w$  is a positive penalty weight. As  $w$  increases, the solution to the unconstrained problem of minimizing  $\phi$  approaches the solution to the constrained problem. Unfortunately, the Hessian of  $\phi$  also becomes more nearly

singular, and the optimization problem becomes increasingly ill-conditioned, as  $w$  increases. We therefore solve a sequence of problems with increasing values of  $w$ , using each solution as the initial estimate for the subsequent problem.

### 3.3 Descent

The functional  $\phi$  is minimized by a gradient descent method using the discretized Sobolev gradient defined below. The method of steepest descent, although notoriously slow with the ordinary gradient, is effective with our gradient, but a nonlinear conjugate gradient method is faster, with a typical speedup factor of about three. Our code and test results are based on the Polak-Ribiere variant of the Fletcher-Reeves conjugate gradient method [Polak 71].

The iteration is

$$\gamma_{k+1} = \gamma_k + \alpha_k \mathbf{d}_k \quad (k = 0, 1, 2, \dots),$$

where

$$\mathbf{d}_k = -\mathbf{g}_k + \frac{(\mathbf{g}_k - \mathbf{g}_{k-1})^t \mathbf{g}_k}{\mathbf{g}_{k-1}^t \mathbf{g}_{k-1}} \mathbf{d}_{k-1},$$

$\mathbf{g}_k$  is the Sobolev gradient of  $\phi$  at  $\gamma_k$ ,  $\gamma_0$  is an initial solution estimate, and  $\alpha_k$  is computed by a line search—minimization of  $\psi(\alpha_k) = \phi(\gamma_k + \alpha_k \mathbf{d}_k)$ . We use Brent's univariate optimization algorithm which combines golden section search with parabolic interpolation [Brent 73]. The iteration is restarted with a steepest descent iteration ( $\mathbf{d}_k = -\mathbf{g}_k$ ) after every  $n$  steps, where  $n$  was (somewhat arbitrarily) chosen to be 3 in our tests.

### 3.4 Sobolev Gradient

Denote by  $S_0$  the set of perturbations for  $\gamma$  which preserve the end conditions. This is the space of all  $m$ -vectors of vertices. A Sobolev inner product associated with a curve (the current approximation to the solution)  $\gamma$  is

$$\langle \mathbf{g}, \mathbf{h} \rangle_\gamma = \int \langle \mathbf{g}''(s), \mathbf{h}''(s) \rangle ds + w_0^2 \int \langle \mathbf{g}(s), \mathbf{h}(s) \rangle ds,$$

where  $s$  is the arc length associated with  $\gamma$ , and  $w_0$  is a positive weight. Note that this is positive (defines an inner product) on functions with square integrable second derivatives and is intrinsic to the curve (independent of the parametrization). The purpose of the second term is both to force positivity and to allow some control over the conditioning of the smoothing operator defined below.

The discretized inner product on  $S_0$  is

$$\begin{aligned} \langle \mathbf{g}, \mathbf{h} \rangle_\gamma &= .5 \sum_{i=1}^m (\langle \Delta^2 \mathbf{g}_i, \Delta^2 \mathbf{h}_i \rangle + w_0^2 \langle \mathbf{g}_i, \mathbf{h}_i \rangle) \Delta a_i \\ &= \langle D\mathbf{g}, D\mathbf{h} \rangle_{2m}, \end{aligned} \quad (3-1)$$

where  $D = \begin{pmatrix} D_2 \\ D_0 \end{pmatrix}: (\mathbb{R}^3)^m \rightarrow (\mathbb{R}^3)^{2m}$  is the discrete differential operator defined by

$$\begin{aligned} D_2 \mathbf{g}_i &= \sqrt{.5 \Delta a_i} \Delta^2 \mathbf{g}_i, \\ D_0 \mathbf{g}_i &= w_0 \sqrt{.5 \Delta a_i} \mathbf{g}_i, \end{aligned}$$

and, for  $\mathbf{r}, \mathbf{s} \in (\mathbb{R}^3)^m$ , the discretized  $L_2$  inner product is

$$\langle \mathbf{r}, \mathbf{s} \rangle_m = \sum_{i=1}^m \langle \mathbf{r}_i, \mathbf{s}_i \rangle.$$

Note that  $\Delta^2 \mathbf{g}_i$  involves the second divided difference of  $\mathbf{g}$  values and values of  $\Delta s_i = \|\gamma_i - \gamma_{i-1}\|$ , not  $\|\mathbf{g}_i - \mathbf{g}_{i-1}\|$ , which would involve  $\mathbf{g}$  in a nonlinear fashion.

We have  $\langle \mathbf{g}, \mathbf{h} \rangle_\gamma = \langle D\mathbf{g}, D\mathbf{h} \rangle_{2m} = \langle D^t D\mathbf{g}, \mathbf{h} \rangle_m = \langle D^t D\mathbf{g}, \mathbf{h} \rangle_{S_0}$ . Now let  $\mathbf{g}$  be the Sobolev gradient ( $\gamma$ -gradient) of  $\phi$  at  $\gamma$ . Then by the Riesz Representation Theorem,

$$\phi'(\gamma)\mathbf{h} = \langle \nabla \phi(\gamma), \mathbf{h} \rangle_{S_0} = \langle \mathbf{g}, \mathbf{h} \rangle_\gamma = \langle D^t D\mathbf{g}, \mathbf{h} \rangle_{S_0}$$

for all  $\mathbf{h} \in S_0$ . Thus the Sobolev gradient  $\mathbf{g}$  is defined by

$$\mathbf{g} = (D^t D)^{-1} \nabla \phi(\gamma).$$

The 4th order smoothing operator  $(D^t D)^{-1}$  may be thought of as a preconditioner for the gradient descent method, but it is chosen in a far less ad hoc manner than is usually the case with preconditioners.

An expression for the discretized  $L_2$  gradient  $\nabla \phi(\gamma)$  is obtained by simply differentiating the expression for  $\phi$ . The operator  $D$  is represented by a  $2m$  by  $m$  matrix which is applied to each of the three components of an element of  $S_0$ . Note that  $D_2$  is tridiagonal with nonzeros in the upper right and lower left corners. Hence, the symmetric positive definite order- $m$  matrix  $D^t D$  is pentadiagonal except for three nonzero elements in the corners. The linear systems are solved by a direct method (using a Cholesky  $R^t R$  factorization for upper triangular matrix  $R$ ).

### 3.5 Software

A well-documented portable Fortran-77 implementation of the new method is available from the second author. An interactive surface plotting program is also available

[Renka 04b]. The Mathematica implementation of the previous method is available from the first author in the notebook format. There is also Mathematica code available to help import the Fortran output into a notebook.

## 4. EXAMPLES

### 4.1 The Explicit Example Revisited

In order to verify the explicit expression for evolution of length in the planar circles, as well as to test the integrity of our implementation of the new method, we ran some tests using steepest descent with fixed constant step-size ( $10^{-5}$ ) and the ordinary  $L_2$ -gradient. Using  $m = 22$  and beginning with a discretization of a circle of circumference 1, we computed 500,000 iterations. All iterates correspond to circles centered at the origin, and Table 1 demonstrates that the curve lengths evolve at the predicted rate. The four columns contain the number of iterations, the computed curve length, the flow time, and the predicted curve length based on  $\tau$ , respectively. The association of the time increment  $\Delta\tau = 3.54$  with an iteration count of  $10^5$  was obtained by evaluating the inverted formula  $\tau(L) = (L^3 - 1)/(6\pi^2)$  at the computed value  $L = 5.951$ .

### 4.2 Speed Improvement

The second example (Figure 2) compares the computational performance of the implementation of the curve-straightening flow in [Linnér 04] with the new algorithm and its Fortran implementation. It is not really a fair comparison since this new implementation uses compiled code and the previous method does not. Regardless, one goal is the development of a much faster generator of periodic geodesics. The surface is given by  $2x^2 + 3y^2 + 5z^2 + x^3z + yz^3 = 1$ . The fourth order terms perturb the ellipsoid so that symmetries are removed. With initial curve the ellipse with  $z = 0$  using the same hardware (AMD 1.2-GHz Athlon PC), a periodic geodesic is generated in less than 2 seconds, compared to

Iterations	$L$	$\tau$	$L_\tau$
0	.9966	0.00	1.000
$10^5$	5.951	3.54	5.950
$2 \times 10^5$	7.491	7.08	7.490
$3 \times 10^5$	8.573	10.62	8.572
$4 \times 10^5$	9.434	14.16	9.434
$5 \times 10^5$	10.162	17.70	10.161

TABLE 1. Computed and predicted curve lengths.

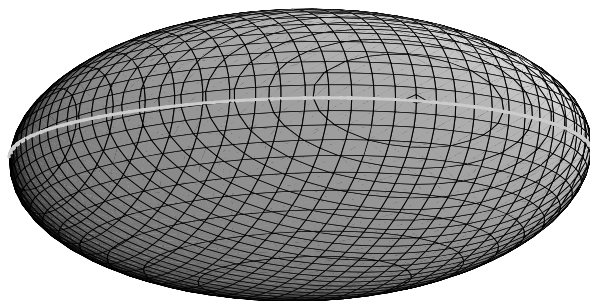


FIGURE 2. Periodic geodesic on  $2x^2 + 3y^2 + 5z^2 + x^3z + yz^3 = 1$ .

several hours using Mathematica. The two implementations produce the exact same periodic geodesic. Observe that the  $z$  component of the geodesic is no longer constant.

### 4.3 Mountain Pass

The third example (Figure 3) illustrates how the flow may carry two distinct initial curves to two distinct periodic geodesics. The surface is a torus of revolution that satisfies  $(\sqrt{x^2 + y^2} - 2)^2 + z^2 = 1$ . The circles with  $z = 0$  and radii 1 and 3, respectively, are periodic geodesics. For  $0 \leq t \leq \pi/2$ , the initial curves with  $z = \cos(t)$  and radii  $2 - \sin(t)$  have the following, perhaps surprising, behavior. If  $t < 0.2$ , then the limit periodic geodesic has radius 3 with  $z = 0$ . If  $t > 0.3$ , then the limit periodic geodesic has radius 1 with  $z = 0$ . In particular, the circle with  $t = 0$ , which is a curve where each point has zero Gaussian curvature, is not exceptional. It is tempting to speculate that a nongeodesic circle for some  $t$  in  $(.2, .3)$  is in fact an elastic curve that appears as a mountain pass critical point. This belief is strengthened when the variational equation  $\kappa_{ss} + \kappa^3/2 + \kappa G = 0$  for free elastic curves is analyzed; see [Langer and Singer 84]. Here,  $G$  is the Gaussian curvature of the surface along the curve. The circles considered here have negative constant geodesic curvature  $\kappa = \cos(t)/(\sin(t) - 2)$ . The Gaussian curvature is the constant  $G = \sin(t)/(\sin(t) - 2)$  along the circles; see page 157 in [do Carmo 76]. For nonzero  $\kappa$  the variational equation simplifies to a quadratic equation with one admissible solution  $\sin(\hat{t}) = 2 - \sqrt{3}$  and  $\hat{t} \approx 0.271263753$ .

### 4.4 Noncompact Surface

The last example (Figure 4) involves a surface with six “ends” given by  $x^2y^2 + y^2z^2 + z^2x^2 = 1$ . Unlike sphere-like surfaces, in this example it is possible to shrink some initial curves without the curve degenerating to a point.

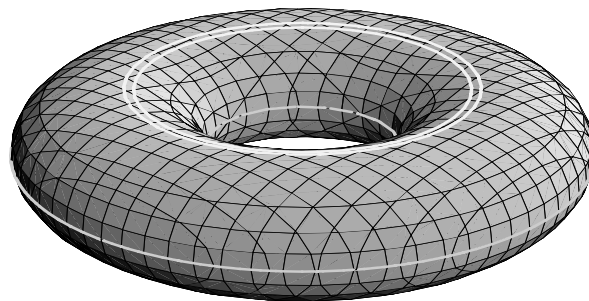


FIGURE 3. Two initial curves that tend to different periodic geodesics on  $(\sqrt{x^2 + y^2} - 2)^2 + z^2 = 1$ .

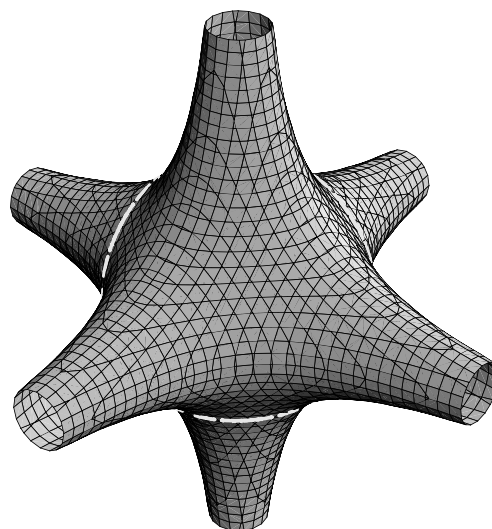


FIGURE 4. Periodic geodesic on  $x^2y^2 + y^2z^2 + z^2x^2 = 1$ .

It is far from obvious how to give an initial curve, with this property, that is in the surface. The new method does not require the initial curve to be in the surface, so here a slanted circle is used. Finally, recall that a standard right circular cylinder is a much simpler example of a noncompact surface with periodic geodesics.

## 5. SUMMARY

The total squared geodesic curvature of a curve is globally minimized if and only if the curve is a geodesic. This suggests generating its negative gradient trajectories inside the infinite-dimensional manifold of curves subject to the boundary conditions under consideration. There are several, rather different, ways to implement this approach. The new ideas presented here address several of the shortcomings with the method presented in [Linnér 04]. Although the two methods agree for some explicit Euclidean examples, there are fundamental differences.

### 5.1 Smoothness and Constraints

The new method avoids the tangent angle of [Linnér 04]. This complicates the algebraic form of the geodesic curvature, and a Sobolev space of second derivatives is required. An advantage is that it is now much easier to deal with the periodic boundary conditions. In particular, it is no longer necessary to repeatedly solve a pair of highly nonlinear six-dimensional initial value problems. The surface constraint enters through a penalty term, and a noteworthy consequence of this is the freedom to choose an initial curve that is not necessarily in the surface.

### 5.2 Steepest Descent

The two methods approximate the infinite-dimensional space of periodic curves differently. The previous method utilizes a Hilbert space basis and a suitable finite Fourier approximation. The new method tracks a finite number of points that belong to the curve throughout the descent. The flow takes place in the ambient Euclidean space and no “exponential projection” is necessary. The new method utilizes conjugate gradient steps along the descent.

### 5.3 Computational Demand

Finally, as mentioned above, the previous implementation is tied to Mathematica and does not, currently, perform at a speed acceptable for extensive exploration and experimentation. The new method is implemented in Fortran and the computations are completed in minutes rather than hours as before. To support visualization, a surface viewer is supplied that also shows the curve in the surface, and for Mathematica users a conversion interface to help import the Fortran output.

## REFERENCES

- [Bangert 93] V. Bangert. “On the Existence of Closed Geodesics on Two-Spheres.” *Internat. J. Math.* 4:1 (1993), 1–10.
- [Brent 73] Richard Brent. *Algorithms for Minimization without Derivatives*. Englewood Cliffs, NJ: Prentice Hall, Inc., 1973.
- [do Carmo 76] Manfredo P. do Carmo. *Differential Geometry of Curves and Surfaces*. Englewood Cliffs, NJ: Prentice Hall, Inc., 1976.
- [Franks 92] J. Franks. “Geodesics on  $S^2$  and Periodic Points of Annulus Homeomorphisms.” *Invent. Math.* 108:2 (1992), 403–418.
- [Gage 90] M. E. Gage. “Deforming Curves on Convex Surfaces to Simple Closed Geodesics.” *Indiana Univ. Math. J.* 39 (1990), 1037–1059.
- [Grayson 89] M. A. Grayson. “Shortening Embedded Curves.” *Ann. of Math.* 129 (1989), 71–111.
- [Langer and Singer 84] J. Langer and D. A. Singer. “The Total Squared Curvature of Closed Curves.” *J. Diff. Geom.* 20 (1984), 1–22.
- [Langer and Singer 87] J. Langer and D. A. Singer. “Curve-Straightening in Riemannian Manifolds.” *Ann. Global Anal. Geom.* 5 (1987), 133–150.
- [Linnér 91] A. Linnér. “Curve-Straightening in Closed Euclidean Submanifolds.” *Commun. Math. Phys.* 138 (1991), 33–49.
- [Linnér 98] A. Linnér. “Curve-Straightening and the Palais-Smale Condition.” *Trans. Amer. Math. Soc.* 350 9 (1998), 3743–3765.
- [Linnér 03] A. Linnér. “Symmetrized Curve-Straightening.” *Diff. Geo. Appl.* 18 (2003), 119–146.
- [Linnér 04] A. Linnér. “Periodic Geodesics Generator.” *Experimental Mathematics* 13:2 (2004), 199–213.
- [Neuberger 97] J. W. Neuberger. *Sobolev Gradients and Differential Equations*. Lecture Notes in Mathematics, 1670. New York: Springer, 1997.
- [Polak 71] E. Polak. *Computational Methods in Optimization*. New York: Academic Press, 1971.
- [Renka 03] R. Renka. “Algorithm 828. DNSPLIN1: Discrete Nonlinear Spline Interpolation.” *ACM Trans. Math. Softw.* 29:4 (2003), 1–11.
- [Renka 04a] R. Renka. “Constructing Fair Curves and Surfaces with a Sobolev Gradient Method.” *CAGD* 21 (2004), 137–149.
- [Renka 04b] R. Renka. “Algorithm 834. glsurf – An Interactive Surface Plotting Program using OpenGL.” *ACM Trans. Math. Softw.* 30:2 (2004), 212–217.

Anders Linnér, Department of Mathematical Sciences, Northern Illinois University, DeKalb, IL 60115  
(alinner@math.niu.edu)

Robert Renka, Department of Computer Science & Engineering, University of North Texas, Denton, TX 76203-1366  
(renka@cs.unt.edu)

Received May 4, 2004; accepted December 28, 2004.