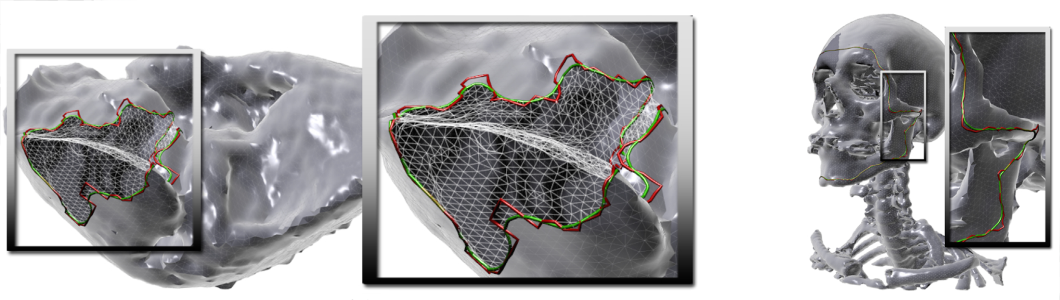
Curve Smoothing on Surface Meshes

Markus Pawellek

August 21, 2023

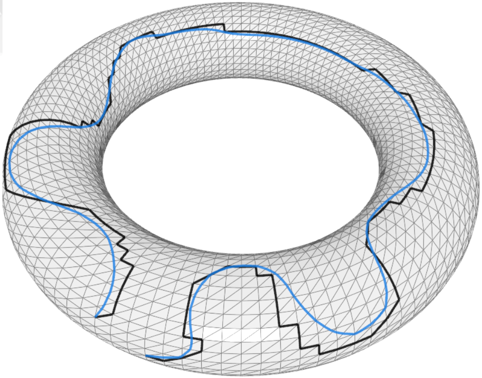
Introduction and Background

Introduction and Background: Reference

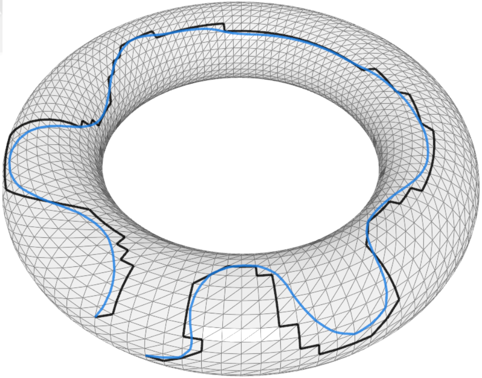


Kai Lawonn, Rocco Gasteiger, Christian Rössl, and Bernhard Preim. "Adaptive and Robust Curve Smoothing on Surface Meshes". In: *Computers & Graphics* 40 (2014), pp. 22–35. DOI: 10.1016/j.cag.2014.01.004

Introduction and Background: Goals

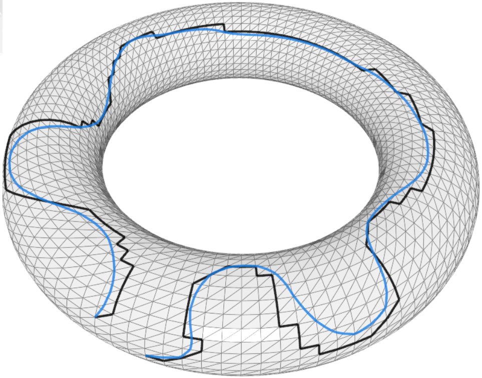


Introduction and Background: Goals

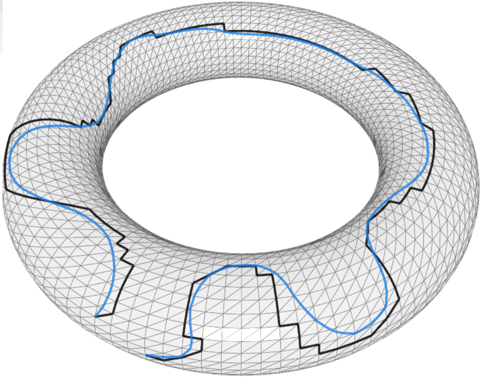


- Closeness to Initial Curve

Introduction and Background: Goals

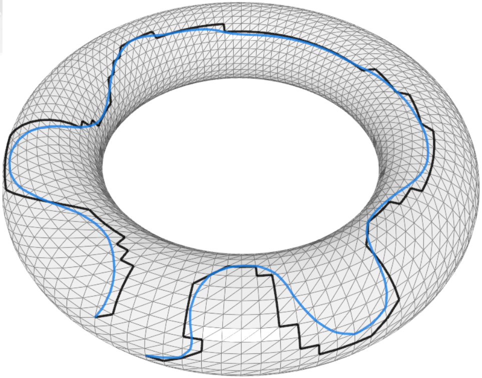


- Closeness to Initial Curve
- Balancing Closeness and Smoothness by Parameter



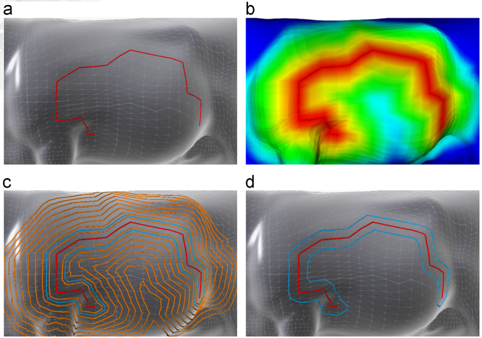
- Closeness to Initial Curve
- Balancing Closeness and Smoothness by Parameter
- Proof of Convergence and Smoothness

Introduction and Background: Goals

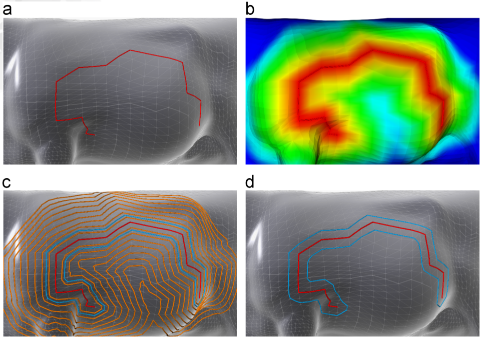


- Closeness to Initial Curve
- Balancing Closeness and Smoothness by Parameter
- Proof of Convergence and Smoothness
- Robust toward Geometric and Parametric Noise

Introduction and Background: Implementation

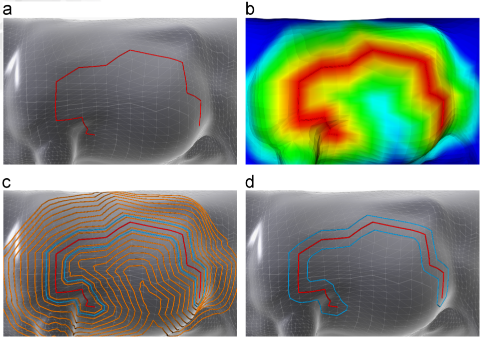


Introduction and Background: Implementation

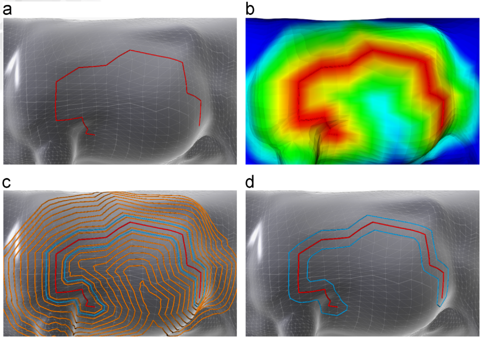


- Distance Envelope

Introduction and Background: Implementation



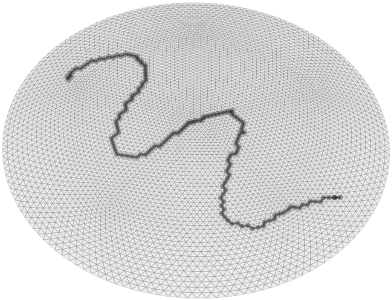
- Distance Envelope
- Weighted Relaxation of Geodesic Curvature to Desired Geodesic Curvature



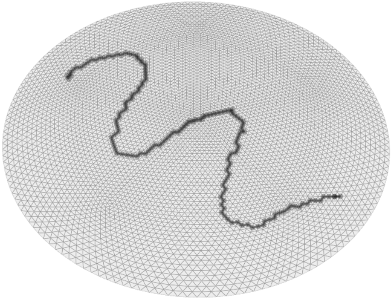
- Distance Envelope
- Weighted Relaxation of Geodesic Curvature to Desired Geodesic Curvature
- Convergence based on Length Reduction

The Idea

The Idea: Surfaces in 2D

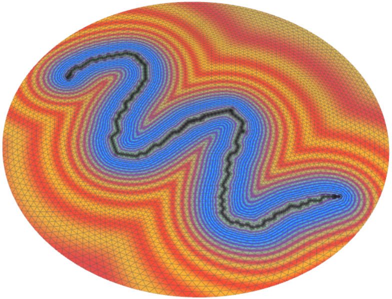


The Idea: Surfaces in 2D



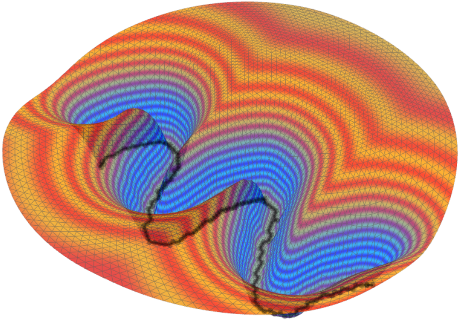
1. Start with initial curve

The Idea: Surfaces in 2D



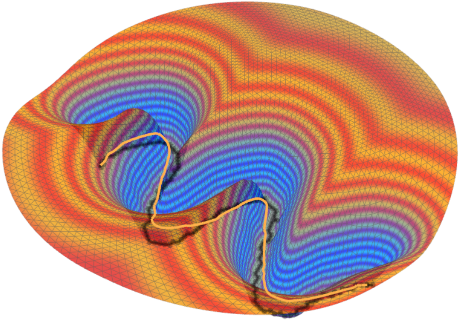
1. Start with initial curve
2. Generate distance field and penalty potential

The Idea: Surfaces in 2D



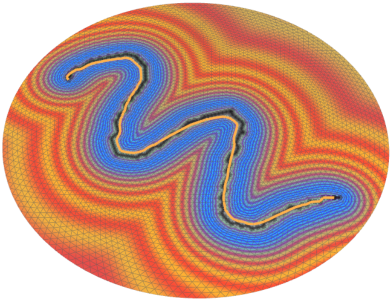
1. Start with initial curve
2. Generate distance field and penalty potential
3. Lift surface to 3D by adding potential to coordinates

The Idea: Surfaces in 2D



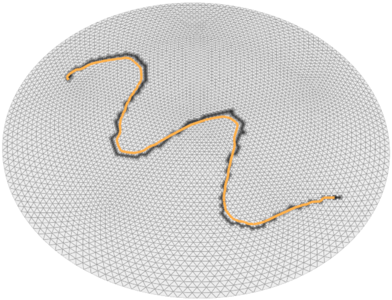
1. Start with initial curve
2. Generate distance field and penalty potential
3. Lift surface to 3D by adding potential to coordinates
4. Generate geodesic for lifted surface

The Idea: Surfaces in 2D



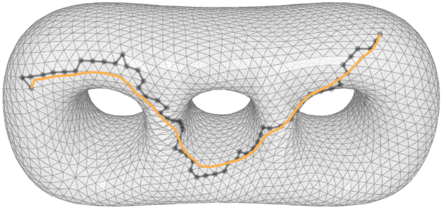
1. Start with initial curve
2. Generate distance field and penalty potential
3. Lift surface to 3D by adding potential to coordinates
4. Generate geodesic for lifted surface
5. Project geodesic back to original surface

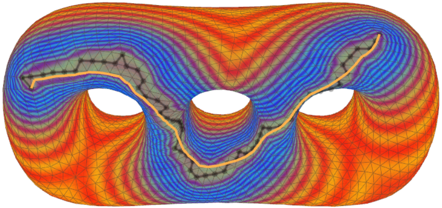
The Idea: Surfaces in 2D



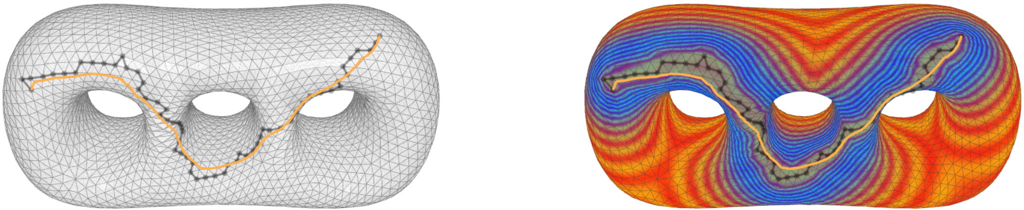
1. Start with initial curve
2. Generate distance field and penalty potential
3. Lift surface to 3D by adding potential to coordinates
4. Generate geodesic for lifted surface
5. Project geodesic back to original surface

The Idea: Surfaces in 3D



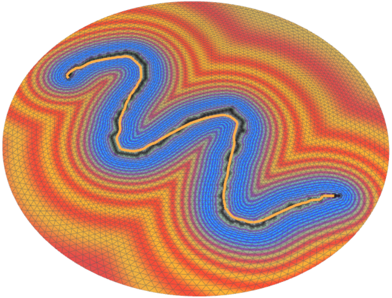


The Idea: Surfaces in 3D

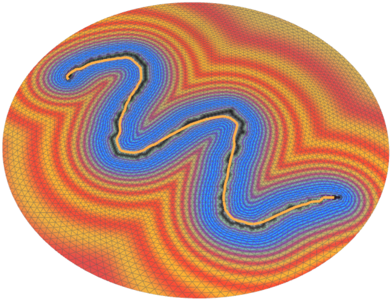


- Surfaces embedded in 3D are lifted into 4D by appending potential to fourth coordinate of vertex positions

The Idea: Balancing Closeness and Smoothness

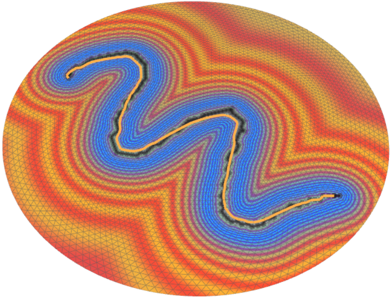


The Idea: Balancing Closeness and Smoothness



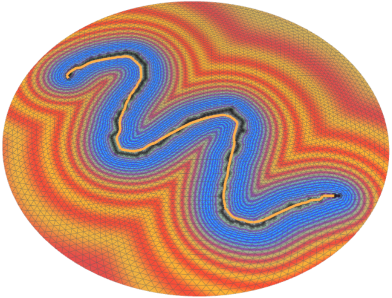
- No restrictions when choosing penalty potentials

The Idea: Balancing Closeness and Smoothness



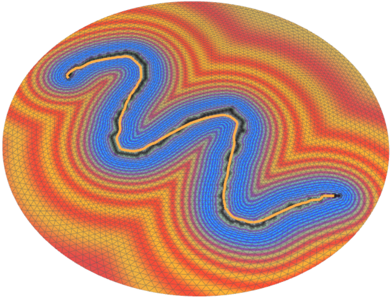
- No restrictions when choosing penalty potentials
- Here, potentials depend on distance d to initial curve

The Idea: Balancing Closeness and Smoothness



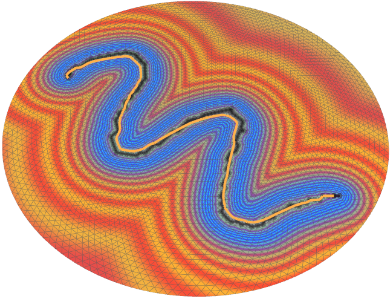
- No restrictions when choosing penalty potentials
- Here, potentials depend on distance d to initial curve
- Using parameterized family of functions allows for balancing

The Idea: Balancing Closeness and Smoothness



- No restrictions when choosing penalty potentials
- Here, potentials depend on distance d to initial curve
- Using parameterized family of functions allows for balancing

The Idea: Balancing Closeness and Smoothness

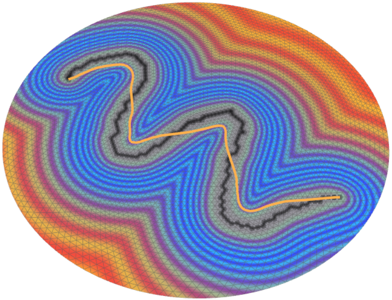


- No restrictions when choosing penalty potentials
- Here, potentials depend on distance d to initial curve
- Using parameterized family of functions allows for balancing

$$\varphi_{\alpha,\tau} : [0,\infty) \to \mathbb{R}, \quad \alpha, \tau \in (0,\infty)$$

$$\varphi_{\alpha,\tau}(d) := \alpha \exp\left(-\frac{1}{\tau d}\right)$$

The Idea: Balancing Closeness and Smoothness

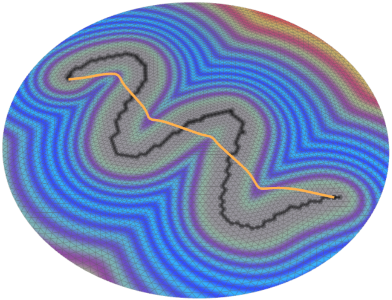


- No restrictions when choosing penalty potentials
- Here, potentials depend on distance d to initial curve
- Using parameterized family of functions allows for balancing

$$\varphi_{\alpha,\tau} : [0,\infty) \to \mathbb{R}, \quad \alpha,\tau \in (0,\infty)$$

$$\varphi_{\alpha,\tau}(d) := \alpha \exp\left(-\frac{1}{\tau d}\right)$$

The Idea: Balancing Closeness and Smoothness

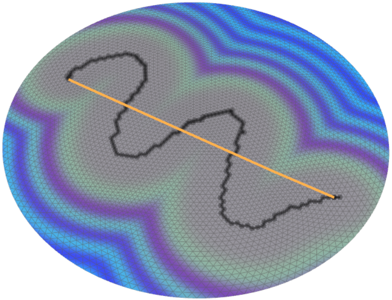


- No restrictions when choosing penalty potentials
- Here, potentials depend on distance d to initial curve
- Using parameterized family of functions allows for balancing

$$\varphi_{\alpha,\tau} \colon [0,\infty) \to \mathbb{R}, \quad \alpha, \tau \in (0,\infty)$$

$$\varphi_{\alpha,\tau}(d) := \alpha \exp\left(-\frac{1}{\tau d}\right)$$

The Idea: Balancing Closeness and Smoothness



- No restrictions when choosing penalty potentials
- Here, potentials depend on distance d to initial curve
- Using parameterized family of functions allows for balancing

$$\varphi_{\alpha,\tau} : [0, \infty) \to \mathbb{R}, \quad \alpha, \tau \in (0, \infty)$$

$$\varphi_{\alpha,\tau}(d) := \alpha \exp\left(-\frac{1}{\tau d}\right)$$

Properties, Limitations, and Future Work

Properties

Properties

- Easy implementation

Properties

- Easy implementation
- (Almost) any geodesics algorithm can be used

Properties

- Easy implementation
- (Almost) any geodesics algorithm can be used
- Inherits properties from geodesics algorithm (robustness, convergence)

Properties

- Easy implementation
- (Almost) any geodesics algorithm can be used
- Inherits properties from geodesics algorithm (robustness, convergence)
- Preserves closeness by definition

Properties

- Easy implementation
- (Almost) any geodesics algorithm can be used
- Inherits properties from geodesics algorithm (robustness, convergence)
- Preserves closeness by definition
- Other potentials allow for more general movement

Properties

- Easy implementation
- (Almost) any geodesics algorithm can be used
- Inherits properties from geodesics algorithm (robustness, convergence)
- Preserves closeness by definition
- Other potentials allow for more general movement
- Arbitrarily many parameters

Properties

- Easy implementation
- (Almost) any geodesics algorithm can be used
- Inherits properties from geodesics algorithm (robustness, convergence)
- Preserves closeness by definition
- Other potentials allow for more general movement
- Arbitrarily many parameters
- Potentials may also depend on other surface characteristics, such as mean curvature

Limitations and Future Work

Limitations and Future Work

- Scaling of potential

Limitations and Future Work

- Scaling of potential
- Good family of penalty potentials

Limitations and Future Work

- Scaling of potential
- Good family of penalty potentials
- Enlarging curves

Limitations and Future Work

- Scaling of potential
- Good family of penalty potentials
- Enlarging curves
- Measuring smoothness

Limitations and Future Work

- Scaling of potential
- Good family of penalty potentials
- Enlarging curves
- Measuring smoothness
- Further mathematical properties

Limitations and Future Work

- Scaling of potential
- Good family of penalty potentials
- Enlarging curves
- Measuring smoothness
- Further mathematical properties
- Applications

Limitations and Future Work

- Scaling of potential
- Good family of penalty potentials
- Enlarging curves
- Measuring smoothness
- Further mathematical properties
- Applications
- Other References: To best of my knowledge, it has not been done.