Homework 3: Intrinsic Geometry of Surfaces

Differential Geometry for Computer Science (Spring 2013), Stanford University

Due Monday, May 20, in the course mailbox

Problem 1 (40 points). *Let $f : [a, b] \to \mathbb{R}$ be a smooth, positive function. The surface of revolution in $\mathbb{R}^3$ obtained by rotating the graph $y = f(z)$ in the (y, z)-plane about the z-axis can be parametrized by*

$$\phi(z, \theta) := \big(f(z)\cos(\theta), f(z)\sin(\theta), z\big) \qquad (z, \theta) \in [a, b] \times [-\pi, \pi]\,.$$

(a) Draw an informative picture.

(b) Find the components of the induced metric and its inverse in the parameter domain.

(c) Write down an integral for the surface area of the surface.

(d) Find the Christoffel symbols (in the form Γ_{ijk}) in the parameter domain.

(e) Write down the geodesic equation in the parameter domain. When is a z = constant curve a geodesics? It is also true that the θ = constant curves are geodesics. Bonus: can you find $z(t)$ so that the curve $\gamma(t) := (z(t), \theta_0)$, where θ_0 is a constant, is a geodesic?

(f) Derive expressions for the gradient, divergence and Laplace operators in the parameter domain.

(g) Derive an expression for the Gauss curvature using intrinsic calculations only. When is the Gauss curvature equal to zero, and is this expected?

Problem 2 (20 points). *The cotangent Laplacian is a key object of study in the world of applied differential geometry. We will introduce several different derivations of the* same *formula with different starting points! Thus, it's worth making sure you can build your own cotangent Laplacian from the ground up.*

(a) We will first encounter the cotangent Laplacian in lecture 12 in our discussion of first-order finite elements. Write code for `FEMLaplacian.m` to compute the cotangent Laplacian L and "consistent" (non-diagonal) mass matrix A. Once again, all your matrices must be sparse to receive credit, and ideally you should find ways to write this method without any loops (they're slow in Matlab!).

(b) Recall that the vibration modes of a surface are given by eigenfunctions f of the form $\Delta f = \lambda f$. Propose a finite elements "weak" formulation of this problem in terms of L and A paralleling our adaptation of the Poisson equation. Complete `problem2b.m` to implement your formulation, and show the eigenfunctions corresponding to the five smallest eigenvalues.

Problem 3 (20 points). *Recall that the heat equation $\frac{\partial u}{\partial t} = -\Delta u$ governs the diffusion of an initial distribution of heat u_0 over time. In this problem, we'll derive a discretization of the heat equation on surfaces, which is a building block for many smoothing algorithms.*

(a) *First discretize "in space" to derive a version of the heat equation on a mesh in terms of u, u_t, L, and A.*

(b) *To discretize our system in time, we'll use a "backward Euler" scheme, a more stable alternative to the "forward Euler" scheme we explored in homework 1. Suppose we choose a time step $T > 0$. Then, we can approximate the left hand side of the heat equation with the divided difference $\frac{1}{T}(u_T - u_0)$. Substituting u_0 into the right hand side mimics the forward Euler scheme. Instead, substitute u_T to the right hand side, and after applying the substitution from part (a) give a formula for u_T in terms of u_0, L, and A.*

(c) *Complete `problem3b.m`, which will iterate your scheme from part (b) to solve the heat equation.*

(d) *What happens as $t \to \infty$? Why?*

(e) *You can think of (x, y, z) coordinates on a surface M as a function $\vec{x}$ from M to $\mathbb{R}^3$. Since heat flow is dispersive, one way you can smooth a surface is to apply heat flow to the function $\vec{x}$ itself. In other words, you would solve $\frac{\partial \vec{x}_t}{\partial t} = -\Delta_t \vec{x}_t$. Notice that Δ is now a function of t – the Laplacian changes as the geometry smooths itself. Adapt your time stepping scheme from (b) by recomputing the Laplacian at each time step (so, you use u at $t = T$ but L and A at $t = 0$), and complete `problem3e.m` to implement this smoothing technique, known as "Implicit Fairing."*

Problem 4 (20 points)**.**

(a) *Justify the following relationship left as an exercise in Lecture 8 (slides 36-37):*

$$\int_V K\,dA = 2\pi - \sum_j \varepsilon_j = 2\pi - \sum_j \theta_j$$

(b) *[Open-ended] Implement two different formulae for Gaussian curvature on a triangle mesh. At least one formula should be from Lecture 8; feel free to choose the second from a published paper. Compare the output of the two methods and discuss when they might be useful.*

We will defer problems involving geodesics to the fourth problem set.