topics

- covariant derivative
- laplacian
- poisson equation
- discrete laplacian

covariant derivative

- recall the derivative of a function $\nabla_a f = \partial_a f = d_a f$
- we contract with a direction to get directional derivative $v^a \nabla_a f$
- we would like to take directional derivatives of tangent vectors, co-vectors and tensors as well.

outline

- in the plane we can take the directional derivative of a vector field (field of number pairs)
- the result is another 2d vector field, which we can think of as a tangent vector field
- we would like to generalize this to a tangent vector field on a surface
- there are some difficulties
- if we work on the surface in 3D, the derivatives of these vectors (number triples) are not tangent vectors
- if we work in a parameterization, we get parametereization dependence.
- so we need to fix this up, either using some 3d geometry, or using the metric information.

covariant derivative in embedding

- suppose we have an embedded surface
- suppose we have a tangent vector field w^a in the surface
- we can think of these as vectors in 3d, and take their derivative along a curve in some tangent direction v^a.
- problem: this derivative vector will not be a tangent vector.
- solution: project it orthogonally back to the tangent plane.
- this is called the directional covariant derivative : $v^a \nabla_a w^b$
- which defines the covariant derivative : $\nabla_a w^b$
- it turns out that it can be calcuated using only the partial derivatives in the parametric domain of the w^i along with the Christoffel tensor
  - the christoffel tensor (of the parameterization) can be computed using partial derivatives of the metric tensor coefficients g_{ij}.
  - no additional 3d information is needed

parallel transport

- we say that a vector field w^a parallel transports in the direction t^a if $t^a \nabla_a w^b = 0$
- so covariant derivative "connects" tangent vectors between neighboring tangent planes
- we say that a vector field w^a parallel transports along a curve with tangent t^a if $t^a \nabla_a w^b = 0$
- it can be shown that if i have two vector fields v^a and w^a, that are parallel transports along a curve, then their dot products remain constant along the curve.

covariant derivative without embedding

- suppose we tried the following
- we represent a vector field w^a in coordinates (2 for surface)
- we think of this as two scalar functions over the surface w^i.
- we then take the derivative of each of these functions $v^a \partial_a w^i$
- we interpret these two numbers as the coordinates of the directional "derivative": $v^a \partial_a w^b$
- this is the same as taking the directional derivatives of a vector field in the parameter domain, and then pushing the answer back onto the surface.
- this gives us a well defined operator ∂_a that when applied to w^b returns a 1,1 tensor.
- if we used this to define parallel transport we will see parameterization dependence
  - think of an all vertical field in one parameterization.
- so $\partial_a \neq \nabla_a$ when applied to vectors on a curved surface

axiomatic derivation of covariant

- define "axioms" that any derivative operator should satisfy
- given a metric g_{ab}
- given two vectors v^a, w^a parallel transported along a curve with tangent t^a
- we want their inner product to stay constant

$$\begin{aligned} 0 &= t^a \nabla_a g_{bc} v^b w^c \\ &= v^b w^c t^a \nabla_a g_{bc} + g_{bc} w^c t^a \nabla_a v^b + g_{bc} v^b t^a \nabla_a w^c \\ &= v^b w^c t^a \nabla_a g_{bc} \\ 0 &= \nabla_a g_{bc} \end{aligned}$$

- this locks down ∇_a
- can be computed using ∂_a and an associated set of "christoffel symbols": $\Gamma^c_{ab} = 1/2 g^{cd}(\partial_a g_{bd} + \partial_b g_{ad} - \partial_d g_{ab})$
- and then $\nabla_a v^c = \partial_a v^c + \Gamma^c_{ab} v^b$
- we will not go into this

laplace-beltrami

- define the laplace beltrami operator as $\triangle f := g^{ab} \nabla_a \nabla_b f$
- in a flat space, this is just the laplacian $f_{uu} + f_{vv}$
- if we have an embedded surface and we take the laplace-beltrami of the three coordinate functions, and we assemble the answer back as a normal vector, we get the mean curvature, as before $2Hn^i := g^{ab} \nabla_a \nabla_b x^i$
- define the divergence of a vector field as the following trace: $div(v^a) := \nabla_a v^a$

poisson equation

- we will look at a sovling a certain optimization problem over covector fields.
- it will turn out the solution will satisfy something called a Poission equation.
- this is an important problem, and equation which will come up this semester.
- the expression will require our laplace beltrami.
- we will also get a glimpse at standard techniques used for these kinds of optimization problems

setup

- suppose i wish to minimize the energy

$$\int_s dA_g g^{ab} \left[(\nabla_a f) - r_a\right]\left[(\nabla_b f) - r_b\right]$$

  - r_a is given as input, and f is the unkown function

- same as minimizing

$$\int_s dA_g g^{ab} \left[1/2(\nabla_a f)(\nabla_b f) - (\nabla_a f) r_b\right]$$

- take "variational derivative"
- for all test functions h
- consider energy of $F = f + \epsilon h$.
- take $dE(F)/d\epsilon$ at $\epsilon = 0$
- if this is zero, then adding in h cannot improve (at first order) the energy
- setting this to zero, we get

$$\int_s dA_g g^{ab}(\nabla_a f)(\nabla_b h) = \int_s dA_g g^{ab}(\nabla_a h) r_b$$

stokes theorem (Gauss form)

$$\int_s dA_g \nabla_a v^a = \int_{\partial s} dl_g n_a v^a$$

basically we have a ∇ on one side and a ∂ on the other
the boundary normal n_a is used to evaluate v^a.
if the surface is closed, there is no boundary, and the rhs is 0.
if v^a vanishes on the boundary, then the rhs is 0
let $v^a = g^{ab} v_b$, then stokes says

$$\int_s dA_g g^{ab} \nabla_a v_b = \int_{\partial s} ...$$

let $v_b = r_b h$, then stokes says

$$\int_s dA_g g^{ab}(\nabla_a r_b) h + \int_s dA_g g^{ab}(\nabla_a h) r_b = \int_{\partial s} ...$$

so

$$\int_s dA_g g^{ab}(\nabla_a r_b) h = -\int_s dA_g g^{ab}(\nabla_a h) r_b + \int_{\partial s} ...$$

apply stokes

- now apply stokes theorem to both sides of our expression above and ignore the boundary
  - either the surface was closed, or we constained our answer to a known value at boundary, so the relevant test functions h are zero at the boundary
- for all h

$$\int_s dA_g (g^{ab} \nabla_a \nabla_b f)(h) = \int_s dA_g (g^{ab} \nabla_a r_b)(h)$$

- sice this is true for all test functions, then at all points we should have

$$(g^{ab}\nabla_a\nabla_b f) = (g^{ab}\nabla_a r_b)$$

- this is the poisson equation $\triangle f = \text{div}(g^{ab}r_b)$

discrete version

- work in orthogonal coordinate system per triangle
- f will represented as one scalar per vertex
- d will map from a p.l. function to a p.c. co-vector, 2 numbers per triangle
- r will represent a rhs covector, 2-numbers per triangle
- M will represent the integration
  - M_2 will represent the area per triangle. it will repeat the number twice, since we are working with co-vectors with two coords
  - M_1 will represent the area of the vornoi region per vertex.
- minimize energy

$$min_f 1/2(df - r)^t M_2(df - r)$$

$$min_f 1/2 f^t d^t M_2 df - f^t d^t M_2 r + 1/2 r^t M_2 r$$

- set gradiant to zero

$$d^t M_2 df \quad = \quad d^t M_2 r$$

discrete laplacian

- roughly we want to think of $d^t M_2 df$ as a discrete laplacian of f, measured at each vertex. and $d^t M_2 r$ as a discrete divergence, and the above as a poisson equation.
  - but (1) the above measurement would be invariant to uniform scales of the mesh by a factor of λ
  - and (2) a continuous laplacian would scale by $\frac{1}{\lambda^2}$
- so for consistency with continuous case, we often pre multiply by the inverse of a per-vertex area.

$$M_1^{-1} d^t M_2 df \quad = \quad M_1^{-1} d^t M_2 r$$

discrete stokes

- look at

$$(h^t d^t) M_2 (df)$$

- we can think of the action of M_2 here to be the area integral

$$\begin{aligned} (h^t d^t) M_2 (df) &= (h^t)(d^t M_2 df) \\ &= (h^t) M_1 (M_1^{-1} d^t M_2 df) \end{aligned}$$

- the last move was done so that we have an M_1 around to stand for the area integral.
- note that the integrated energy does not change under mesh scaling

cotangents

- for hat basis, lets look at $(f^t d^t M_2 df)$

- thm: this can be evaluated per triangle as

$$1/2 \sum_t \sum_{he_{vw}} (\cot \alpha_{vw})(f_v - f_w)^2$$

- so no need to use a parameter dependent d matrix

matrix form

- we can rewrite

$$\begin{aligned} L &= M_1^{-1} d^t M_2 d \\ &= M_1^{-1} \hat{d}^t \hat{C} \hat{d} \end{aligned}$$

- where $\hat{d}$ is the 2e by v incidence matrix, and $\hat{C}$ is an 2e by 2e diagonal matrix with the contangents.

full edges

$$f^t d^t M_2 df = 1/2 \sum_t \sum_{he_{vw}} (\cot \alpha_{vw})(f_v - f_w)^2$$

- since each half edge appears twice with the same value, we can also write this as

$$1/2 \sum_{e_{vw}} (\cot \alpha_{wv} + \cot \alpha_{vw})(f_v - f_w)^2$$

- can fix orientation of e arbitrarily

matrix form

- we can factor the laplacian as

$$\begin{aligned} L &= M_1^{-1} d^t M_2 d \\ &= M_1^{-1} \bar{d}^t C \bar{d} \end{aligned}$$

- where $\bar{d}$ is the e by v incidence matrix, and C is an e by e diagonal matrix with the contangents.

Laplacian

- given f, what is $Lf = M_1^{-1} \bar{d}^t C \bar{d} f$ at vertex v.

$$1/A_v \sum_{w \in N(v)} (\cot \alpha_{wv} + \cot \alpha_{vw})(f_v - f_w)$$

rhs

- lets look at $(df - r)^t M_2 (df - r)$
- this can be evaluated per triangle as

$$1/2 \sum_t \sum_{he_{vw}} (\cot \alpha_{vw})((f_v - f_w) - r_t(he_{vw}))^2$$
$$1/2 \sum_t \sum_{he_{vw}} (\cot \alpha_{vw})((f_v - f_w) - r_t^t(x_v - x_w))^2$$

- in the first expression $r_t(he)$ is the one form integrated along the half edge. in the second expression, this is computed by dotting the coordinates of the covector with the (local orthonromal) coordinates of the edge.
- assuming that there is no consistancy in r_t between two triangles that share an edge, we cannot get a full-edge version of this expression.

div of r

- in matrix form, we can write

$$M_1^{-1} d^t M_2 r \quad = \quad M_1^{-1} \hat{d}^t \hat{C} \hat{r}$$

- where $\hat{r}$ is the integral of r along each half edge.
- at each vertex, this gives us

$$\begin{aligned} 1/A_v & \sum_{w \in N(v)} (\cot \alpha_{vw}) r_t^t (x_v - x_w) \\ -1/A_v & \sum_{w \in N(v)} (\cot \alpha_{wv}) r_t^t (x_w - x_v) \end{aligned}$$

alternative discrete covector data type

- if r came as the derivative of a continuous, PL function, then $r(he)$ would negate across matching half edges.
- so we only would need to store one number (and a prefered, but arbitrary, edge direction)
- we would call such a data type $\bar{r}$
- for this data type, we could take its divergence as

$$M_1^{-1} \bar{d}^t \bar{C} \bar{r}$$

- which expresses

$$1/A_v \sum_{w \in N(v)} (\cot \alpha_{vw} + \cot \alpha_{wv}) \bar{r}_{vw}$$

discrete one-form

- at this point, if we want to generalize away from derivatives of functions, we could even remove the constraint that the (signed) sum of $\bar{r}$ around a face has to vanish.
- this is often a very natural defintion for a discrete co-vector on a mesh.