

Fundamentals of computer graphics

Luca Moschella

January 16, 2019

Contents

10 Mesh processing I	4
10.1 Adjacency matrices	4
10.1.1 Vertex-to-vertex	4
10.1.2 Vertex-to-triangle	4
10.1.3 Vertex-vertex co-occurrence	5
10.1.4 Triangle-to-triangle	6
10.1.5 Powers	6
10.1.6 Point clouds	7
10.2 Adjacency matrices as operators	7
10.3 Shortest paths on a graph	9
10.4 Graph Laplacian	9
10.5 Least squares meshes	10
11 Regular surfaces	12
11.1 Differential geometry	12
11.2 A union of charts	12
11.3 Regular surfaces	13
11.4 Parametrized curves	14
11.5 Charts	15
11.6 Tangent plane	16
11.7 Differential of a map	16
11.8 Mesh parametrization	19
12 Lengths and areas	20
12.1 Measuring lengths and areas	20
12.2 First fundamental form	20
12.2.1 Metric tensor	22
12.2.2 Generalized first fundamental form	23
12.3 Parametrization	23
12.4 Examples	24
12.4.1 The plane	24
12.4.2 The cylinder	28
12.4.3 The sphere	29
12.5 Length of a curve	30
12.5.1 Arc-length of a curve	30
12.5.2 Arc-length element	31
12.6 Area of a region	32
12.6.1 Area element	34
12.7 Summary	34
12.8 Integral of a function	34
12.9 Discretization	35
12.9.1 Metric tensor	35
12.9.2 Area element	36
12.9.3 Integral of a function	36

13 The gradient	39
13.1 Gradient of a function	39
13.2 The gradient in \mathbb{R}^2	39
13.3 The gradient on a surface	40
13.3.1 Geometric meaning	42
13.3.2 Representation theorem	42
13.3.3 Definition of the gradient	43
13.4 The gradient in local coordinates	44
13.5 The gradient in global coordinates	46
13.6 Summary	46
13.7 Gradient norm	47
13.8 Example: The sphere	48
13.8.1 Gradients in \mathbb{R}^2	49
13.8.2 Gradients on the surface	49
13.9 Discretization	51
13.9.1 The gradient	51
13.9.2 Gradient norm	52
13.9.3 Total variation	52
14 The Laplace-Beltrami operator	54
14.1 Motivation	54
14.2 Inner product on a manifold	55
14.2.1 Scalar functions	55
14.2.2 Tangent vector fields	56
14.3 Intuitions	57
14.4 Adjointness	58
14.5 Finite elements method (FEM)	59
14.5.1 Hat basis	60
14.5.2 Discretization	61
14.5.3 Discrete Laplace-Beltrami operator	62
14.5.4 Mass integral	63
14.5.5 Stiffness integral	66
14.5.6 Stiffness and mass matrices	66
15 Spectral geometry	69
15.1 Self-adjointness of Δ	69
15.2 Laplacian eigenvalues	69
15.3 Laplacian eigenfunctions	70
15.4 Spectral theorem	71
15.5 Discretization	72
15.6 Integral of eigenfunctions	73
15.7 Dirichlet energy	74
15.8 Discrete spectrum	75
15.9 Laplacian eigenvalues: Properties	76
15.10 Laplacian eigenfunctions	77
15.10.1 Fourier analysis	78

16 Heat diffusion	80
16.1 Heat equation	80
16.2 Heat kernel	80
16.3 Dirac initialization	81
16.4 Heat kernel: Properties	82
16.5 Heat kernel: Spectral decomposition	82
16.6 Heat kernel signature	86
16.7 Diffusion distance	86

10 Mesh processing I

10.1 Adjacency matrices

Let $|V| = n$, $|E| = e$, $|F| = m$ be the number of nodes, edges and faces for a mesh $M = (V, E, F)$. The mesh connectivity can be encoded in *adjacency matrices*. In general we have $m \approx 2n$, and with n in the order of several thousand these adjacency matrices can be *very large*, quadratic in n . It is advisable to use *sparse* data structures to store them.

10.1.1 Vertex-to-vertex

The *vertex-to-vertex* adjacency is defined as the $n \times n$ binary matrix:

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & \cdots & 1 \\ \vdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \vdots \\ 1 & 0 & \cdots & 1 \end{pmatrix}$$

where $a_{ij} = 1$ if vertex v_i is connected to v_j (that is $e_{ij} \in E$).

It has the following properties:

- The diagonal is always 0, since in meshes there aren't self-loops
- \mathbf{A} is *symmetric*, since in meshes edges are undirected
- Each row and column has at least one 1, it means every vertex must be connected to some other vertex. And in particular the sum of the elements of this matrix is equal to number of edges in the mesh $\sum_{ij} a_{ij} = e$

10.1.2 Vertex-to-triangle

The *vertex-to-triangle* adjacency is defined as the $n \times m$ binary matrix:

$$\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 1 & 1 \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ 0 & 1 & 0 & \cdots & 1 & 0 & 1 \end{pmatrix}$$

where $p_{ij} = 1$ if vertex v_i belongs to triangle t_j .

It has the following properties:

- Each *row* has at least one 1 (each vertex belongs to some triangle). It is exactly 1 only on pointy boundaries
- Each *column* sums up to 3 (each triangle has exactly 3 vertices)

10.1.3 Vertex-vertex co-occurrence

Consider the product:

$$\mathbf{P}\mathbf{P}^T = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 1 & 1 \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ 0 & 1 & 0 & \cdots & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & 1 \\ 0 & \cdots & \cdots & 0 \\ \vdots & \cdots & \cdots & \vdots \\ 0 & \cdots & \cdots & 1 \\ 1 & \cdots & \cdots & 0 \\ 1 & \cdots & \cdots & 1 \end{pmatrix}$$

$\mathbf{P}\mathbf{P}^T$ is a $n \times n$ matrix with the same zero pattern as \mathbf{A} .

The cell (i, j) counts the number of triangles where v_i and v_j co-occur. It is the sum of element-wise product:

$$(\mathbf{P}\mathbf{P}^T)_{ij} = [\mathbf{P}_{i1} \ \dots \ \mathbf{P}_{im}] \begin{bmatrix} \mathbf{P}_{j1} \\ \vdots \\ \mathbf{P}_{jm} \end{bmatrix}$$

It gives 1 each time $\mathbf{P}_{i\cdot}$ and $\mathbf{P}_{j\cdot}$ are 1, that is when two vertices co-occur in the same triangle.

Remark 1. If two vertices co-occur and aren't the same vertex, it means they are connected by an edge since we are dealing with triangles, thus, in the corresponding cell of the vertex adjacency matrix there would be 1. In the same way, if two vertex are connected they co-occur at least in one triangle, thus, in the corresponding cell of the vertex co-occurrence matrix there would be 1. This means that the vertex co-occurrence matrix has the same zero pattern of the vertex adjacency matrix.

It has the following properties:

- $\text{diag}(\mathbf{P}\mathbf{P}^T)$ contains the number of triangles for each vertex. It can be any number (think of a fan configuration)
- All other values are 0 ($e_{ij} \notin E$), 1 (boundary, e_{ij} appears in only one triangle) or 2 (interior, e_{ij} appears in two triangles). No other options are possible since in a manifold an edge can be at most in 2 triangles

10.1.4 Triangle-to-triangle

Consider the product:

$$\mathbf{P}^\top \mathbf{P} = \begin{pmatrix} 1 & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & 1 \\ 0 & \cdots & \cdots & 0 \\ \vdots & \cdots & \cdots & \vdots \\ 0 & \cdots & \cdots & 1 \\ 1 & \cdots & \cdots & 0 \\ 1 & \cdots & \cdots & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 1 & 1 \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ 0 & 1 & 0 & \cdots & 1 & 0 & 1 \end{pmatrix}$$

which is a $m \times m$ matrix.

The cell (i, j) counts the number of vertices that t_i and t_j share. It's like the co-occurrence for vertices.

It has the following properties:

- $\text{diag}(\mathbf{P}^\top \mathbf{P})$ is always 3 (the number of vertices per triangle)
- All other values are 0 (non-adjacent triangles), 1 (triangles share one vertex) or 2 (triangles share an edge). At most 2 triangles can share a single edge

Checking if the values in the matrix are at most two is a very nice and clean way to see if the given triangular mesh is a manifold mesh. Note that it is a necessary check, but not sufficient (e.g. the mesh could have two different triangles that intersect: this problem can't be detected by an adjacency matrix).

10.1.5 Powers

The k -th power of \mathbf{A} corresponds to composing \mathbf{A} with itself $k \geq 1$ times.

For $k = 2$:

$$\mathbf{A}^2 = \mathbf{A}\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 1 \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ 1 & 0 & 1 & \cdots & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & \cdots & 1 \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ 1 & 0 & 1 & \cdots & 1 \end{pmatrix}$$

The result is a $n \times n$ matrix encoding the *2nd order adjacency*, where in cell (i, j) there is the number of common neighbors between v_i and v_j .

For $k > 1$:

$$\mathbf{A}^k = \mathbf{A} \cdots \mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 1 \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ 1 & 0 & 1 & \cdots & 1 \end{pmatrix} \cdots \begin{pmatrix} 0 & 1 & 0 & \cdots & 1 \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ 1 & 0 & 1 & \cdots & 1 \end{pmatrix}$$

The results is a $n \times n$ matrix encoding the k -th order adjacency

Regarding the other types of adjacency matrices:

- The same applies to *triangle-to-triangle* adjacency
- The k -th power of *vertex-to-triangle* adjacency is given by $\mathbf{A}^k \mathbf{P}$

Manipulating adjacency is useful in many tasks relying upon *local context*.

10.1.6 Point clouds

Adjacency is a general notion that can be extended to *point clouds*. For example, use *Euclidean distance* within a threshold τ :

$$a_{ij} = \begin{cases} 1 & \text{if } \|\mathbf{v}_i - \mathbf{v}_j\|_2 \leq \tau \\ 0 & \text{otherwise} \end{cases}$$

Similarly to before, \mathbf{A}^k encodes k -th order adjacency.

Such notions of adjacency can of course be used on meshes as well.

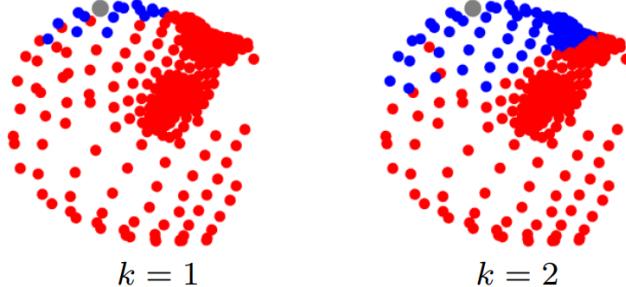


Figure 4: Example of 1-th and 2-th order adjacency on point clouds

10.2 Adjacency matrices as operators

We can see adjacency matrices as *operators* when applied to functions. For example, $\mathbf{g} = \mathbf{A}\mathbf{f}$ yields a *vertex-based* function g defined as:

$$g(v_i) = \sum_{e_{ij} \in E} f(v_j)$$

On this observation, one can construct new operators such as $\mathbf{g} = \mathbf{I} - \mathbf{A}$:

$$g(v_i) = f(v_i) - \sum_{e_{ij} \in E} f(v_j)$$

Similarly for *triangle-based* functions.

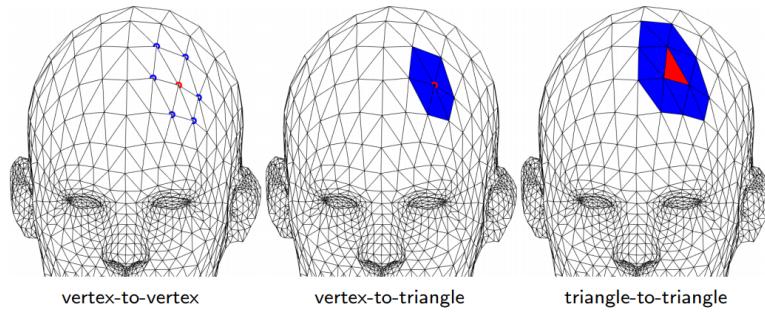


Figure 1: Example of 1-th order adjacency

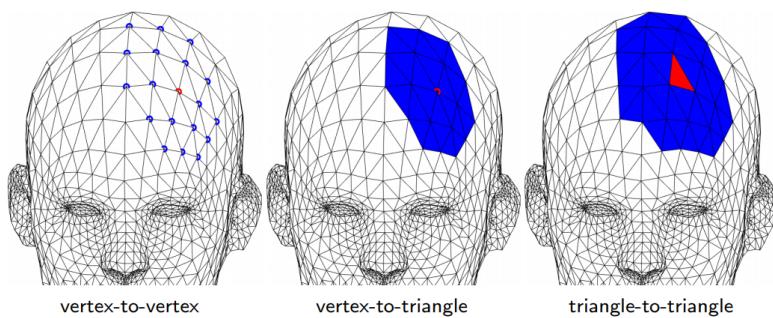


Figure 2: Example of 2-th order adjacency

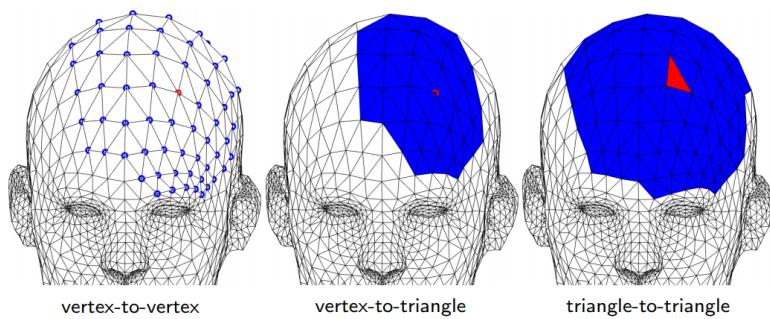


Figure 3: Example of 3-th order adjacency

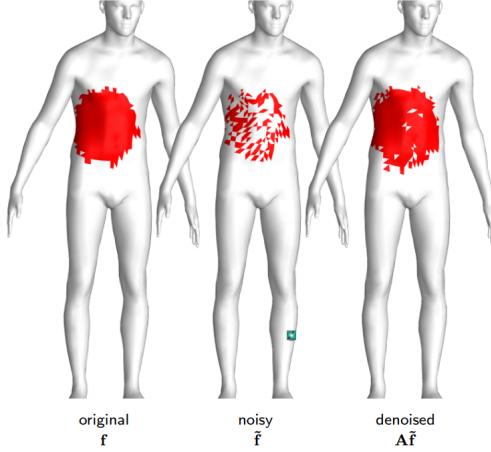


Figure 5: A possible application of adjacency operators is hole-filling

10.3 Shortest paths on a graph

Shortest paths along edges provide *upper bounds* to exact geodesics, referring to Figure 6:

$$\sqrt{2} = d(v_2, v_3) \neq d(v_1, v_4) = 2$$

However it is still useful with *high resolution* meshes or for *local* distances. It is solved by *Dijkstra's algorithm* on the mesh graph.

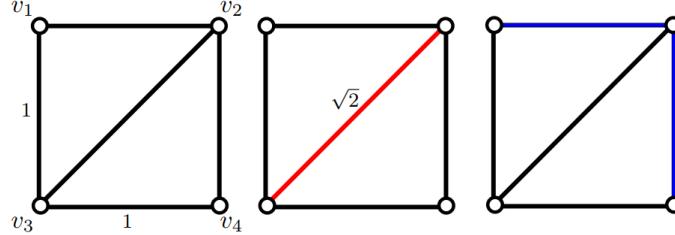


Figure 6: Visualization of shortest path on a mesh

10.4 Graph Laplacian

Given a *mesh graph* $G = (V, E)$ consider this condition on vertex v_i :

$$\mathbf{v}_i - \frac{1}{d_i} \sum_{j:(i,j) \in E} \mathbf{v}_j = 0 \quad (1)$$

where d_i is the valence (or degree) of v_i . The equation is satisfied if v_i lies in the *barycenter* of its neighbors.

Remark 2. It's interesting to note that in defining the graph Laplacian we are ignoring the triangles.

In matrix notation, we define the $n \times n$ matrix \mathbf{L} as:

$$L_{ij} = \begin{cases} 1 & \text{if } i = j \\ -\frac{1}{d_i} & \text{if } e_{ij} \in E \\ 0 & \text{otherwise} \end{cases}$$

also known as the *graph Laplacian* of G . We can rewrite the condition (1) as:

$$\mathbf{LV} = \mathbf{0}$$

The Graph Laplacian has the following properties:

- It is symmetric. If we think again at the definition in terms of the adjacency matrix, it is just the identity scaled by something minus the adjacency matrix, so the result is symmetric
- The sum of the rows, and the columns since it's symmetric, is zero. This comes directly from the definition of the Graph Laplacian

10.5 Least squares meshes

There are embeddings that satisfy $\mathbf{LV} = \mathbf{0}$? This condition forces the embeddings to have a more or less uniform tessellation.

There are some trivial solution that satisfy the condition:

- The trivial embeddings $\mathbf{V} = \mathbf{0}$
- One-dimensional subspace $\text{span}(\mathbf{1})$, i.e. \mathbf{V} equal to a constant vector

but, is there an *interesting* embedding satisfying $\mathbf{LV} = \mathbf{0}$?

We'll try to define a problem that forces the embedding to be interesting, defining some *anchor vertices* with fixed positions.

Optimization problem

Assume $m \geq 1$ anchor vertices $\mathbf{v}_s \in \mathcal{A}$ with a known 3D position. Then, consider the linear system:

$$\begin{pmatrix} \mathbf{L} \\ \mathbf{A} \end{pmatrix} \mathbf{V} = \mathbf{b}$$

where

$$a_{ij} = \begin{cases} 1 & \text{if } \mathbf{v}_j \in \mathcal{A} \\ 0 & \text{otherwise} \end{cases} \quad b_k = \begin{cases} (0, 0, 0) & k \leq n \\ \mathbf{v}_{s_{k-n}} & n < k \leq n + m \end{cases}$$

This problem may not have an exact solution in general, so we will try to minimize the error redefining the problem as follow:

$$\begin{pmatrix} \mathbf{L} \\ \mathbf{A} \end{pmatrix} \mathbf{V} \approx \mathbf{b} \implies \min_{\mathbf{v} \in \mathbb{R}^{n \times 3}} \| \begin{pmatrix} \mathbf{L} \\ \mathbf{A} \end{pmatrix} \mathbf{V} - \mathbf{b} \|_2^2 \quad (2)$$

$$\implies \min_{\mathbf{v} \in \mathbb{R}^{n \times 3}} \| \mathbf{L}\mathbf{V} \|_2^2 + \sum_{\mathbf{v}_i \in \mathcal{A}} \| \mathbf{v}_i - \mathbf{b}_i \|_2^2 \quad (3)$$

where in (2) we are imposing to minimize the approximation error in the L_2 square sense, in (3) we are writing more explicitly the problem that is encoded in the compact notation:

- $\mathbf{L}\mathbf{V} = \mathbf{0}$ since $\mathbf{b} = \mathbf{0}$ for the first n elements (initial condition) thus $\mathbf{L}\mathbf{V}$ should be as small as possible
- $\mathbf{v}_i - \mathbf{b}_i = \mathbf{0}$ since \mathbf{A} is a $m \times n$ matrix that selects one element per row, and it should be as small as possible

$$\mathbf{AV} - \mathbf{b} = \mathbf{0} \iff \begin{pmatrix} \mathbf{v}_{s_1} \\ \vdots \\ \mathbf{v}_{s_m} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_{n+1} \\ \vdots \\ \mathbf{b}_{n+m} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$$

The anchor constraints are not satisfied exactly, since we aren't optimizing only over them, however at higher resolution the error distributes better among the constraints (Figure 7).

It is possible to use this procedure to perform some primitive *shape modeling*, changing the anchor positions (Figure 8).

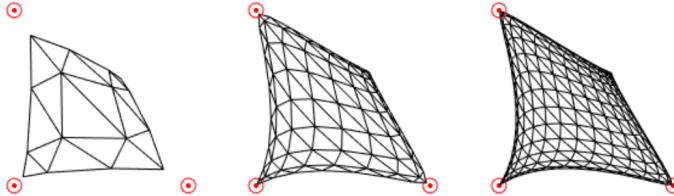


Figure 7: Approximation error when changing the resolution of the mesh

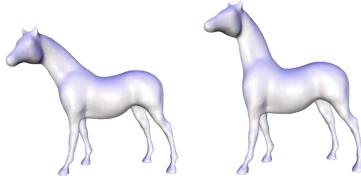


Figure 8: Shape modeling using least square meshes

11 Regular surfaces

11.1 Differential geometry

Differential geometry studies the *local properties* of curves and surfaces, that is, properties which depend on the behavior in the neighborhood of a point. We do have a notion of neighborhood because we know what distance means, so now we can make use of these ideas.

Differential geometry is about locality, usually it is not concerned with saying anything global about the shape

It gives us powerful tools to compute *lengths, areas, integrals, gradients*, etc. on *surfaces*.



Figure 9: Neighborhoods of points



Figure 10: Distances on a surface

11.2 A union of charts

We are all used to think of the world as a collection of maps (Figure 11), also called charts. There is not a unique way of charting the geography of the world, it depends on what kind of measure you want to minimize. So, we can think of the world, which is approximately a sphere, as a union of charts. These charts don't have to be disjoint, in Figure 11 they overlap.

In differential geometry *chart* indicates the mapping $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ and not the portion of the 2D plane involved in the mapping.

We know that these mapping can't be isometries in general, e.g. for spheres we have seen that a mapping to or from \mathbb{R}^2 can't be isometric.

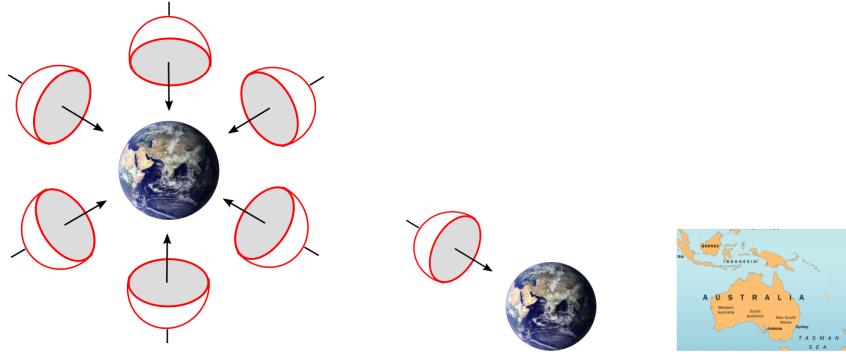


Figure 11: Surfaces as union of charts Figure 12: The chart is the *mapping*

11.3 Regular surfaces

Each regular surface can be seen as a union of multiple charts.

A surface in \mathbb{R}^3 is a subset of \mathbb{R}^3 . It's a *regular surface* if it can be generated with the following procedure (Figure 13):

1. *Cut* pieces of a *plane*
2. *Deform* these pieces
3. *Glue* them together in a shape so that there are no sharp points, edges, or self-intersections (*regularity*)

Regularity ensures that we can talk about *tangent planes* at each point. In non-regular surfaces (Figure 14) along the sharp points the tangent plane is not unique, so it is not well defined.

Remark 3. Our shapes have these kind of corners, we are dealing with 3-dimensional since meshes, so, we are full of sharp edges and sharp points. In fact we never deal with real manifolds, true differential surfaces. However we will see that all the notion that we will see in differential geometry can be translated to the discrete universe. There will be a way to generalize to triangle meshes.

In the language of differential geometry, we call these regular surfaces *2-dimensional Riemannian sub-manifolds*, “sub” because they are a subset of \mathbb{R}^3 , “2-dimensional” because our charts go from $\mathbb{R}^2 \rightarrow \mathbb{R}^3$, i.e. we cut pieces of the plane and that is what makes them two dimensional, “riemannian” if we equip these surfaces with a notion of distance, just because Riemann was the one having the idea. Equivalently, we can call these regular surfaces *differential manifolds* or just *manifolds*.

Remark 4. Our shapes are 2-dimensional because they can be mapped to the plane, we are only seeing a collections of distortion of the plane. We are looking at portions of the plane, and we visualize an embedding of them in \mathbb{R}^3 . Actually,

this is a two dimensional creature living in \mathbb{R}^3 . A 3-dimensional manifold, would be a surface with all its interior, a volumetric object.



Figure 13: Union of charts

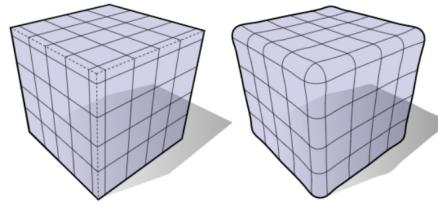


Figure 14: Non-regular and regular surfaces

11.4 Parametrized curves

A *parametrized curve* (Figure 15) is a differentiable map $\alpha : (t_0, t_1) \rightarrow \mathbb{R}^3$

$$\alpha(t) = (x(t), y(t), z(t))$$

where t is called a *parameter*; $x(t)$, $y(t)$, $z(t)$ are differentiable; and the *tangent* vector at t is: $\alpha'(t) = (x'(t), y'(t), z'(t))$

The fact that the curve is differentiable means basically that you can move to close by points in space by increasing the value of t by an infinitesimal amount. You can't increase t by an infinitesimal amount and move sharply in space.

Remark 5. Parametrized curves are not functions in general.

Example 1 (parametrized curve). A parametrization for the helix (Figure 17) is $\alpha(t) = (a \cos(t), a \sin(t), bt)$

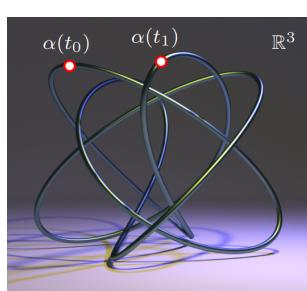


Figure 15: A parametrized curve

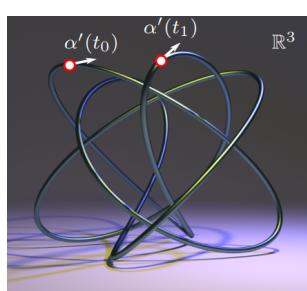


Figure 16: Tangent vectors on a parametrized curve

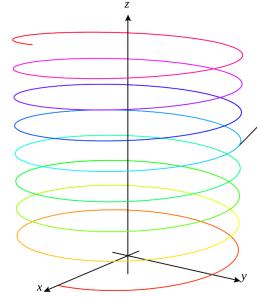


Figure 17: A parametrized helix

11.5 Charts

Definition 1 (chart). A chart ϕ is an *invertible smooth* map with a *smooth* inverse:

$$\phi : U \rightarrow S \quad \phi(u, v) = (x(u, v), y(u, v), z(u, v))$$

where $U \subset \mathbb{R}^2$ and $S \subset \mathbb{R}^3$

Remark 6. A smooth, invertible map with a smooth inverse on manifolds is called diffeomorphism.

In general the chart can't be an isometry and the parametrization is *not unique* (Figure 18)



Figure 18: Two different chart parametrizations for the same surface, the grid highlights the deformations that the maps perform

Definition 2 (regular chart). A chart $\phi : U \rightarrow S$ is *regular* if there exists a *tangent plane* at all points of S .



Figure 19: The tip of the cone is a *singular point*, as with the curves, a *tangent plane* can't be defined at p

Remark 7. A single regular surface may be parametrized by multiple charts. The charts may overlap but they can't create irregularities. Each chart must have an inverse function, by definition, independently to the other charts.

11.6 Tangent plane

We'll do a quick recap of all the definition needed to define the tangent plane.

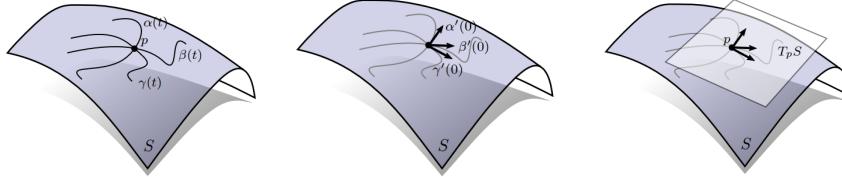
Definition 3 (derivative). The derivative of a function represents an infinitesimal change in the function with respect to one of its variables. The derivative of a function $f(x)$ with respect to the variable x is defined as:

$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h}$$

Definition 4 (tangent line). A straight line is tangent to a given curve $f(x)$ at a point x_0 on the curve if the line passes through the point $(x_0, f(x_0))$ on the curve and has slope $f'(x_0)$, where $f'(x)$ is the derivative of $f(x)$. This line is called tangent line, or sometimes simply a tangent.

Definition 5 (tangent vector). A *tangent vector* is a vector that is tangent to a curve or surface at a given point, thus it lies on the tangent line at that point.

Definition 6 (tangent plane). Given a point p on a surface S , the tangent plane of S at point p is the set of *tangent vectors* to all possible parametrized curves on S passing through p . The tangent plane is denoted by $T_p S$ or $T_p(S)$. Moreover, the curves are parametrized so that $p = \alpha(0) = \beta(0) = \dots$



All the tangent vectors of curves on the surface S lie on the same plane since the curves are defined on S (Theorem 2).

Note that every point on the surface has its own tangent plane, it doesn't make sense to talk about the "tangent plane of the surface". We can talk about the tangent bundle of the surface.

Definition 7 (tangent bundle). The collections of tangent planes of a surface $\{T_p(S) \mid p \in S\}$ is called *tangent bundle* of S .

11.7 Differential of a map

Definition 8 (differential of a map). The *differential* of the map ϕ at point p is defined as:

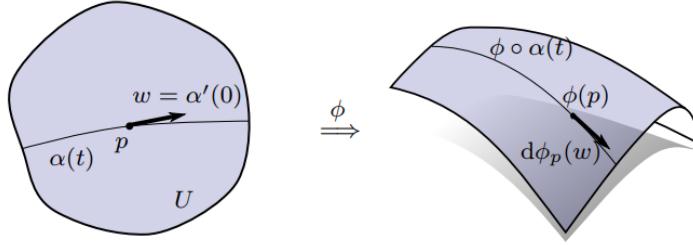
$$d\phi_p(w) = (\phi \circ \alpha)'(0)$$

where

$$\alpha(t) = (u(t), v(t)) \in \mathbb{R}^2 \quad (\phi \circ \alpha)(t) = \phi(u(t), v(t)) \in \mathbb{R}^3$$

It is a *linear map* that takes tangent vectors to tangent vectors:

$$d\phi_p : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$



Recall, for a curve $\alpha(t)$ in \mathbb{R}^2 :

$$\alpha(t) = \begin{pmatrix} u(t) \\ v(t) \end{pmatrix} \quad \alpha'(t) = \begin{pmatrix} \frac{du}{dt} \\ \frac{dv}{dt} \end{pmatrix} \implies \alpha'(0) = \begin{pmatrix} \frac{du}{dt} \Big|_{t=0} \\ \frac{dv}{dt} \Big|_{t=0} \end{pmatrix} := \begin{pmatrix} \alpha_u \\ \alpha_v \end{pmatrix}$$

$$\phi(u, v) = (x(u, v), y(u, v), z(u, v))$$

Now we want to compute $d\phi_p(w)$:

$$\mathbb{R}^3 \ni d\phi_p(w) = d\phi_p \begin{pmatrix} \alpha_u \\ \alpha_v \end{pmatrix} = (\phi \circ \alpha(t))'(0) \quad (4)$$

$$= \frac{d}{dt}(\phi \circ \alpha(t))|_{t=0} = \frac{d}{dt}(\phi(u(t), v(t)))|_{t=0} \quad (5)$$

$$= \frac{d}{dt} \left(\begin{matrix} x(u(t), v(t)) \\ y(u(t), v(t)) \\ z(u(t), v(t)) \end{matrix} \right) \Big|_{t=0} = \left(\begin{matrix} \frac{d}{dt}x(u(t), v(t)) \Big|_{t=0} \\ \frac{d}{dt}y(u(t), v(t)) \Big|_{t=0} \\ \frac{d}{dt}z(u(t), v(t)) \Big|_{t=0} \end{matrix} \right) \quad (6)$$

$$= \begin{pmatrix} \frac{\partial x}{\partial u} \frac{\partial u}{\partial t} + \frac{\partial x}{\partial v} \frac{\partial v}{\partial t} \\ \frac{\partial y}{\partial u} \frac{\partial u}{\partial t} + \frac{\partial y}{\partial v} \frac{\partial v}{\partial t} \\ \frac{\partial z}{\partial u} \frac{\partial u}{\partial t} + \frac{\partial z}{\partial v} \frac{\partial v}{\partial t} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial u} \alpha_u + \frac{\partial x}{\partial v} \alpha_v \\ \frac{\partial y}{\partial u} \alpha_u + \frac{\partial y}{\partial v} \alpha_v \\ \frac{\partial z}{\partial u} \alpha_u + \frac{\partial z}{\partial v} \alpha_v \end{pmatrix} \quad (7)$$

$$= \begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} \end{pmatrix} \begin{pmatrix} \alpha_u \\ \alpha_v \end{pmatrix} = \underbrace{\begin{pmatrix} | & | \\ \frac{\partial \phi}{\partial u} & \frac{\partial \phi}{\partial v} \\ | & | \end{pmatrix}}_{Jacobian} \begin{pmatrix} \alpha_u \\ \alpha_v \end{pmatrix} \quad (8)$$

where in (4) we are writing explicitly the tangent vector $w \in \mathbb{R}^2$ and applying the definition of the differential of a map, in (5) we are changing the notation of the derivative and expanding the function composition with the definition

of α , in (6) we are using the definition of ϕ and the linearity of the derivative, in (7) we are using the multivariate chain rule (Definition 10) and the compact notation we defined for $\alpha'(0)$, in (8) we are decomposing the matrix into two matrix and we notice that one of those it's the Jacobian of ϕ and the other is just w , thus the differential of a map is the Jacobian.

In differential geometry the Jacobian is the matrix representation of the differential of a map, and has rank at most 2. The Jacobian for a 2-dimensional manifold is a 3×2 matrix.

Notice that multiplying the Jacobian and a vector of the standard basis just selects a column of the Jacobian. Actually, the Jacobian is mapping basis vectors to basis vectors, of course this new basis isn't necessary orthogonal and even the lengths may change. It is interesting to note that the coefficients (α_u, α_v) of the tangent vector $w \in \mathbb{R}^2$ in the standard basis are *the same coefficients* of the tangent vector $d\phi_p(w) \in \mathbb{R}^3$ in the new basis $\frac{\partial \phi}{\partial u}, \frac{\partial \phi}{\partial v}$ of the tangent plane, as we can see in (8), this property comes from the linearity of the differential.

Thus, the vectors $\frac{\partial \phi}{\partial u}, \frac{\partial \phi}{\partial v}$ span the tangent plane at $\phi(p)$ if the rank of the Jacobian is 2 (Figure 21), because it means they are linearly independent and so they form a basis. If the rank is 1 it means that the two vectors are not linearly independent, are one the scale of the other, and so you can't define a tangent plane but only a line. So, whenever you get that the rank of a Jacobian is not 2 you are at a singular point, you can't define a tangent plane, thus the surface is not regular.

Remark 8. Without ulterior constraints (injectivity), a linear map doesn't preserve linear independence or dependence.

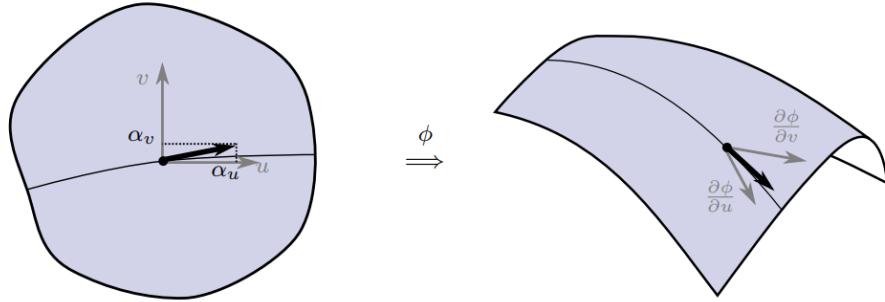


Figure 21: The columns of the Jacobian span the tangent plane

Remark 9. The chart ϕ doesn't have one differential. It has as many differentials as it has points in U , the differential of ϕ is actually the differential of ϕ_p , it is computed for point p and it's its own differential which knows how to map tangent vectors at $p \in \mathbb{R}^2$ to tangent vectors at $p \in \mathbb{R}^3$.

Definition 9. The Jacobian matrix is the matrix of all first-order partial derivatives of a vector-valued function.

Theorem 1. *The differential of a map is a linear operation.*

Proof. Since the differential of a map is representable as the Jacobian matrix, it must be a linear operation since any matrix describes a linear map. \square

Theorem 2. All the tangent vectors S at point p of all the curves defined on the surface that pass through p , lie on the same plane (i.e. $T_p(S)$).

Proof. The differential of a map maps the tangent vectors $\mathbb{R}^2 \rightarrow \mathbb{R}^3$ and it can be encoded in a 3×2 matrix, the Jacobian. This matrix has rank at most 2, so the image of this mapping is at most 2, that is, a plane. So the tangent vectors on the surface at point p are all on the same tangent plane $T_p(S)$. \square

11.8 Mesh parametrization

We deal with *discrete* surfaces (Figure 22):



Figure 22: A discrete surface

We parametrize the surface with a chart for each triangle. We have a chart for each triangle, so we can “hardcode” the vertices positions in each chart and then perform a linear interpolation. For a mesh with m triangles, we have m charts $\phi_j : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ with $j = 1 \dots m$:

$$\phi_j(u, v) = \begin{pmatrix} x_1^j \\ y_1^j \\ z_1^j \end{pmatrix} + u \left(\begin{pmatrix} x_2^j \\ y_2^j \\ z_2^j \end{pmatrix} - \begin{pmatrix} x_1^j \\ y_1^j \\ z_1^j \end{pmatrix} \right) + v \left(\begin{pmatrix} x_3^j \\ y_3^j \\ z_3^j \end{pmatrix} - \begin{pmatrix} x_1^j \\ y_1^j \\ z_1^j \end{pmatrix} \right)$$

with $u \in [0, 1]$, $v \in [0, 1 - u]$.

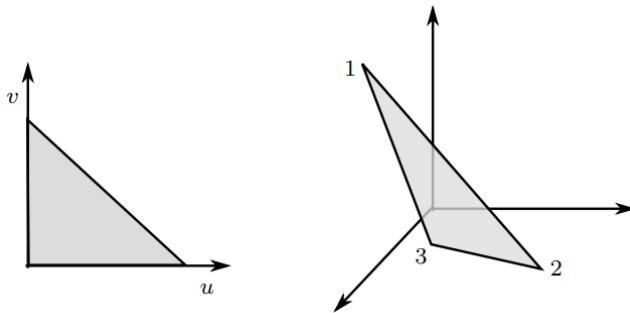
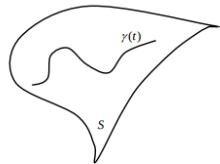


Figure 23: Parametrization of a 3D triangle from the canonical 2D triangle

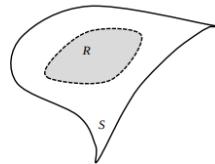
12 Lengths and areas

12.1 Measuring lengths and areas

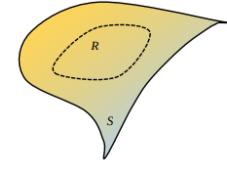
How can we compute the length of a curve, the area of a region or the integral of a function $f : S \rightarrow \mathbb{R}$ on top of surfaces?



Length of a curve



Area of a region



Integral of a function
 $f : S \rightarrow \mathbb{R}$

12.2 First fundamental form

The quadratic form $I_p : T_p(S) \rightarrow \mathbb{R}$ given by:

$$I_p(w) = \langle w, w \rangle_p = \|w\|^2$$

is called the *first fundamental form* of the regular surface S at point p .

It depends on the point in which we are operating, since the tangent plane is defined given a point, and obviously the vector w that we are considering.

It is the key ingredient for computing *lengths* and *areas* of surfaces.

Derivation

Let us denote by $\{\mathbf{x}_u, \mathbf{x}_v\}$ the basis spanning the tangent plane $T_p(S)$, i.e. the columns of the Jacobian.

Any vector $\mathbf{w} \in T_p(S)$ is the tangent vector to a curve $\alpha(t) = \mathbf{x}(u(t), v(t))$ which lies on the surface, with $p = \alpha(0)$.

Then we can write:

$$I_p(\mathbf{w}) = I_p(\alpha'(0)) \tag{9}$$

$$= \langle \alpha'(0), \alpha'(0) \rangle_p \tag{10}$$

$$= \langle \mathbf{x}_u u' + \mathbf{x}_v v', \mathbf{x}_u u' + \mathbf{x}_v v' \rangle_p \tag{11}$$

$$= \langle \mathbf{x}_u, \mathbf{x}_u \rangle_p (u')^2 + 2 \langle \mathbf{x}_u, \mathbf{x}_v \rangle_p u' v' + \langle \mathbf{x}_v, \mathbf{x}_v \rangle_p (v')^2 \tag{12}$$

$$:= E (u')^2 + 2F u' v' + G (v')^2$$

where (9) is due to the definition of tangent vector \mathbf{w} at p (derivative of the curve at p), (10) is due to the definition of the first fundamental form, (11) is due

to the *chain rule*:

$$\begin{aligned} \text{let } \beta(t) &= (u(t), v(t)) & \alpha(t) &= \mathbf{x}(u(t), v(t)) = \mathbf{x}(\beta(t)) \\ \beta : \mathbb{R} &\rightarrow \mathbb{R}^2 & \mathbf{x} : \mathbb{R}^2 &\rightarrow \mathbb{R}^3 \\ \beta : t &\rightarrow (u, v) & \mathbf{x} : (u, v) &\rightarrow (x, y, z) \end{aligned}$$

applying the chain rule

$$\begin{aligned} J_\alpha &= J_{\mathbf{x}}(u, v) \cdot J_\beta(t) \\ &= \begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} \end{pmatrix} \begin{pmatrix} \frac{\partial u}{\partial t} \\ \frac{\partial v}{\partial t} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial u} \frac{\partial u}{\partial t} + \frac{\partial x}{\partial v} \frac{\partial v}{\partial t} \\ \frac{\partial y}{\partial u} \frac{\partial u}{\partial t} + \frac{\partial y}{\partial v} \frac{\partial v}{\partial t} \\ \frac{\partial z}{\partial u} \frac{\partial u}{\partial t} + \frac{\partial z}{\partial v} \frac{\partial v}{\partial t} \end{pmatrix} \\ &= \begin{pmatrix} \frac{\partial x}{\partial u} u' + \frac{\partial x}{\partial v} v' \\ \frac{\partial y}{\partial u} u' + \frac{\partial y}{\partial v} v' \\ \frac{\partial z}{\partial u} u' + \frac{\partial z}{\partial v} v' \end{pmatrix} = (\mathbf{x}_u u' + \mathbf{x}_v v') \end{aligned}$$

in (12) we are applying the definition of inner product for vector values $\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{i=1}^n a_i b_i$ and the linearity of the sum:

$$\begin{aligned} \langle \mathbf{a} + \mathbf{b}, \mathbf{a} + \mathbf{b} \rangle &= \sum_{i=1}^n (a_i + b_i)(a_i + b_i) = \sum_{i=1}^n (a_i + b_i)^2 \\ &= \sum_{i=1}^n a_i^2 + 2a_i b_i + b_i^2 = \sum_{i=1}^n a_i^2 + \sum_{i=1}^n 2a_i b_i + \sum_{i=1}^n b_i^2 \\ &= \langle \mathbf{a}, \mathbf{a} \rangle + 2\langle \mathbf{a}, \mathbf{b} \rangle + \langle \mathbf{b}, \mathbf{b} \rangle \end{aligned}$$

Remark 10. u' and v' are the coefficients of the tangent vector of the curve in parameter space (as we saw, they are also the coefficients of \mathbf{w} , the tangent vector on the surface)

Remark 11 (derivative of multivariable function). The derivative of $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, with $n \neq 1$ or $m \neq 1$ is *not a number* and it is represented by the Jacobian of the function J_f

Definition 10 (multivariable chain rule). The Jacobian of a composite function is the product of the Jacobians of the composed functions, evaluated at the appropriate points.

Let $\beta : \mathbb{R} \rightarrow \mathbb{R}^2$ $\beta(t) = (x(t), y(t))$ be differentiable at point t , let $\alpha : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ $\alpha(t) = f(\beta(t))$ be differentiable at point $(x(t), y(t))$. Then $\alpha = (f \circ \beta)(t) = f(\beta(t))$ is differentiable at t and

$$\frac{d\alpha}{dt} = \frac{\partial \alpha}{\partial x} \frac{dx}{dt} + \frac{\partial \alpha}{\partial y} \frac{dy}{dt}$$

Or, equivalently, expressing the derivatives in terms of the Jacobians:

$$J_\alpha = J_f|_{x,y} \cdot J_\beta|_t$$

12.2.1 Metric tensor

$$\begin{aligned}
I_p(\mathbf{w}) &= E (u')^2 + 2Fu'v' + G (v')^2 \\
E &= \langle \mathbf{x}_u, \mathbf{x}_u \rangle_p \\
F &= \langle \mathbf{x}_u, \mathbf{x}_v \rangle_p \\
G &= \langle \mathbf{x}_v, \mathbf{x}_v \rangle_p \\
\implies I_p(\mathbf{w}) &= E (u')^2 + 2Fu'v' + G (v')^2 \\
&= (u' \quad v') \begin{pmatrix} Eu' + Fv' \\ Fu' + Gv' \end{pmatrix} \\
&= (u' \quad v') \underbrace{\begin{pmatrix} E & F \\ F & G \end{pmatrix}}_{\text{called } g} \begin{pmatrix} u' \\ v' \end{pmatrix}
\end{aligned}$$

The first fundamental form $I_p(\mathbf{w})$ is also called the *metric tensor* (sometimes just g is called metric tensor).

$E(u, v)$, $F(u, v)$ and $G(u, v)$ are the *components* (or the coefficients) of the first fundamental form. These play important roles in many intrinsic quantities of the surface. If $E(u, v)$, $F(u, v)$ and $G(u, v)$ are smooth, we have a *Riemannian manifold*.

Remark 12. E , F and G will change depending on where we are, because they are defined as the inner products of tangent vectors at that point. We can think of them, actually, as functions of the point. So E , F and G depend on the position on the surface, so they are functions on the surface. Thus E , F and G are function of u and v , and we can write $E(u, v)$, $F(u, v)$ and $G(u, v)$.

Metric tensor and Jacobian

We have seen the Jacobian matrix:

$$J_{\mathbf{x}} := D_{\mathbf{x}} = \begin{pmatrix} | & | \\ \frac{\partial \phi}{\partial u} & \frac{\partial \phi}{\partial v} \\ | & | \end{pmatrix} = \begin{pmatrix} | & | \\ \mathbf{x}_u & \mathbf{x}_v \\ | & | \end{pmatrix}$$

Then, it is easy to see that:

$$\begin{aligned}
J_{\mathbf{x}}^\top J_{\mathbf{x}} &= \begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial y}{\partial u} & \frac{\partial z}{\partial u} \\ \frac{\partial x}{\partial v} & \frac{\partial y}{\partial v} & \frac{\partial z}{\partial v} \end{pmatrix} \begin{pmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \\ \frac{\partial z}{\partial u} & \frac{\partial z}{\partial v} \end{pmatrix} \\
&= \begin{pmatrix} \frac{\partial x}{\partial u} \frac{\partial x}{\partial u} + \frac{\partial y}{\partial u} \frac{\partial y}{\partial u} + \frac{\partial z}{\partial u} \frac{\partial z}{\partial u} & \frac{\partial x}{\partial u} \frac{\partial x}{\partial v} + \frac{\partial y}{\partial u} \frac{\partial y}{\partial v} + \frac{\partial z}{\partial u} \frac{\partial z}{\partial v} \\ \frac{\partial x}{\partial v} \frac{\partial x}{\partial u} + \frac{\partial y}{\partial v} \frac{\partial y}{\partial u} + \frac{\partial z}{\partial v} \frac{\partial z}{\partial u} & \frac{\partial x}{\partial v} \frac{\partial x}{\partial v} + \frac{\partial y}{\partial v} \frac{\partial y}{\partial v} + \frac{\partial z}{\partial v} \frac{\partial z}{\partial v} \end{pmatrix} \\
&= \begin{pmatrix} \langle \mathbf{x}_u, \mathbf{x}_u \rangle & \langle \mathbf{x}_u, \mathbf{x}_v \rangle \\ \langle \mathbf{x}_v, \mathbf{x}_u \rangle & \langle \mathbf{x}_v, \mathbf{x}_v \rangle \end{pmatrix} = g
\end{aligned}$$

12.2.2 Generalized first fundamental form

Some use a generalized first fundamental form $I_p : T_p(S) \times T_p(S) \rightarrow \mathbb{R}$

Given $\mathbf{w}_1 \in T_p(S)$ and $\mathbf{w}_2 \in T_p(S)$ it's defined as:

$$I_p(\mathbf{w}_1, \mathbf{w}_2) = \langle \mathbf{w}_1, \mathbf{w}_2 \rangle_p$$

Derivation

Let us denote by $\{\mathbf{x}_u, \mathbf{x}_v\}$ the basis spanning the tangent plane $T_p(S)$. Let $\mathbf{w}_1 \in T_p(S)$ be tangent to the curve $\alpha(t)$ and $\mathbf{w}_2 \in T_p(S)$ be tangent to the curve $\beta(t)$, both the curves lie on the surface with $p = \alpha(0) = \beta(0)$

$$\begin{aligned} I_p(\mathbf{w}_1, \mathbf{w}_2) &= I_p(\alpha'(0), \beta'(0)) \\ &= \langle \alpha'(0), \beta'(0) \rangle_p \\ &= \langle \mathbf{x}_u u'_1 + \mathbf{x}_v v'_1, \mathbf{x}_u u'_2 + \mathbf{x}_v v'_2 \rangle_p \\ &= \langle \mathbf{x}_u, \mathbf{x}_u \rangle_p u'_1 u'_2 + \langle \mathbf{x}_u, \mathbf{x}_v \rangle_p (u'_1 v'_2 + u'_2 v'_1) + \langle \mathbf{x}_v, \mathbf{x}_v \rangle_p v'_1 v'_2 \\ &= E u'_1 u'_2 + F (u'_1 v'_2 + u'_2 v'_1) + G v'_1 v'_2 \end{aligned}$$

where the derivation is basically the same as before.

Metric tensor

Even the generalized first fundamental form can be expressed in terms of g :

$$\begin{aligned} I_p(\mathbf{w}_1, \mathbf{w}_2) &= E u'_1 u'_2 + F (u'_1 v'_2 + u'_2 v'_1) + G v'_1 v'_2 \\ E &= \langle \mathbf{x}_u, \mathbf{x}_u \rangle_p \\ F &= \langle \mathbf{x}_u, \mathbf{x}_v \rangle_p \\ G &= \langle \mathbf{x}_v, \mathbf{x}_v \rangle_p \\ \implies I_p(\mathbf{w}_1, \mathbf{w}_2) &= E u'_1 u'_2 + F (u'_1 v'_2 + u'_2 v'_1) + G v'_1 v'_2 \\ &= (u'_1 \quad v'_1) \begin{pmatrix} Eu'_2 + Fv'_2 \\ Fu'_2 + Gv'_2 \end{pmatrix} \\ &= (u'_1 \quad v'_1) \underbrace{\begin{pmatrix} E & F \\ F & G \end{pmatrix}}_{\text{called } g} \begin{pmatrix} u'_2 \\ v'_2 \end{pmatrix} \end{aligned}$$

12.3 Parametrization

Remark 13. E, F and G depend on $\{\mathbf{x}_u, \mathbf{x}_v\}$, the basis of the tangent plane $T_p(S)$ at a given point.

Depending on the values of E, F and G there are some special names to the resulting parametrization:

generic if E, F and G can assume generic values, we have a generic parametrization (Figure 24)

orthogonal if $F \equiv 0$ it means that $\langle \mathbf{x}_u, \mathbf{x}_v \rangle = 0$, hence the basis of the tangent plane $T_p(S)$ must be orthogonal (Figure 25)

conformal if $F \equiv 0$ and $E = G$ it means that, beside being orthogonal, the square lengths of the basis vectors are the same (Figure 26)

isometric if $F \equiv 0$ and $E = G = 1$, intuitively it means that the transformation is an isometry. Hence, the vectors in \mathbb{R}^2 can be only translated, rotated or reflected into the tangent plane $T_p(S)$ (Figure 27). Note that we are talking about isometries even without talking of distance functions, we'll see that the two definitions are equivalent

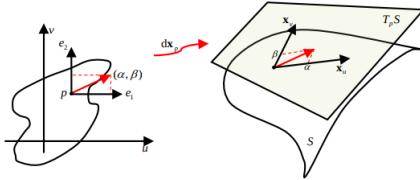


Figure 24: Generic parametrization

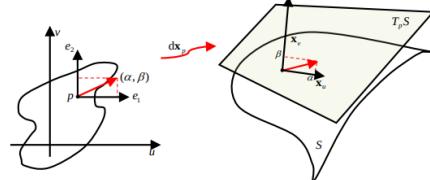


Figure 25: Orthogonal parametrization: $F \equiv 0$

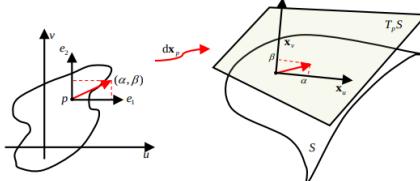


Figure 26: Conformal parametrization: $F \equiv 0$ and $E = G$

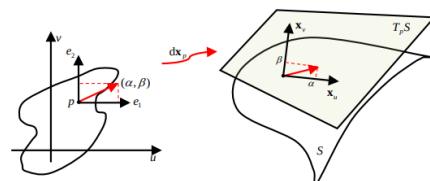


Figure 27: Isometric parametrization: $F \equiv 0$ and $E = G = 1$

12.4 Examples

12.4.1 The plane

First example

Consider a plane $S \subset \mathbb{R}^3$ passing through q_0 and containing the *orthonormal* vectors $\tilde{\mathbf{w}}_1$ and $\tilde{\mathbf{w}}_2$, we define the chart:

$$\begin{aligned}\tilde{\mathbf{x}}(u, v) &= q_0 + u\tilde{\mathbf{w}}_1 + v\tilde{\mathbf{w}}_2 & \Rightarrow & \tilde{\mathbf{x}}_u = \frac{\partial \tilde{\mathbf{x}}}{\partial u} = \tilde{\mathbf{w}}_1 \\ &&& \tilde{\mathbf{x}}_v = \frac{\partial \tilde{\mathbf{x}}}{\partial v} = \tilde{\mathbf{w}}_2\end{aligned}$$

This chart $\tilde{\mathbf{x}}$ takes a point from the plane in \mathbb{R}^2 and gives a point on a plane that lives in \mathbb{R}^3 . Now we have an explicit expression for the chart, so we can explicitly compute all these things that we saw (e.g. the Jacobian, the metric tensor, E , F , G)

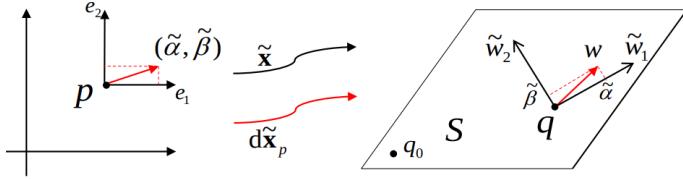
Remark 14. If we use the standard base in \mathbb{R}^2 , in general we have that the base of $T_p(S)$ is:

$$\tilde{\mathbf{x}}_u = d\tilde{\mathbf{x}}_p \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \tilde{\mathbf{x}}_v = d\tilde{\mathbf{x}}_p \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

In this case given the parametrization we have that:

$$d\tilde{\mathbf{x}}_p \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \tilde{\mathbf{w}}_1 \quad d\tilde{\mathbf{x}}_p \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \tilde{\mathbf{w}}_2$$

We want to compute the *first fundamental form* for an arbitrary point q in S , which means we want to compute the squared length of tangent vectors at q .



Remark 15. Note how the coefficients $\tilde{\alpha}, \tilde{\beta}$ of the tangent vector in the parameter space are also the coefficients of this vector on the surface.

We can explicitly compute \tilde{g} since the relevant information are given:

$$\tilde{g} = \begin{pmatrix} \tilde{E} & \tilde{F} \\ \tilde{F} & \tilde{G} \end{pmatrix} = \begin{pmatrix} \langle \tilde{\mathbf{x}}_u, \tilde{\mathbf{x}}_u \rangle & \langle \tilde{\mathbf{x}}_u, \tilde{\mathbf{x}}_v \rangle \\ \langle \tilde{\mathbf{x}}_v, \tilde{\mathbf{x}}_u \rangle & \langle \tilde{\mathbf{x}}_v, \tilde{\mathbf{x}}_v \rangle \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

since $\tilde{\mathbf{x}}_u = \frac{\partial \tilde{\mathbf{x}}}{\partial u} = \tilde{\mathbf{w}}_1$ and $\tilde{\mathbf{x}}_v = \frac{\partial \tilde{\mathbf{x}}}{\partial v} = \tilde{\mathbf{w}}_2$, and, we know that $\{\tilde{\mathbf{w}}_1, \tilde{\mathbf{w}}_2\}$ form an orthonormal basis.

Remark 16. Every vector $\mathbf{w} \in T_p(S)$ is in the form $\mathbf{w} = \tilde{\alpha}\tilde{\mathbf{x}}_u + \tilde{\beta}\tilde{\mathbf{x}}_v$. Note that $\tilde{\alpha}$ and $\tilde{\beta}$ are the coefficients of \mathbf{w} along the base vectors, they do not define the base vectors.

Thus, the first fundamental form of $\mathbf{w} \in T_p(S)$ at p is

$$I_p((\tilde{\alpha}, \tilde{\beta})) = (\tilde{\alpha} \quad \tilde{\beta}) \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \tilde{\alpha} \\ \tilde{\beta} \end{pmatrix} = (\tilde{\alpha} \quad \tilde{\beta}) \begin{pmatrix} \tilde{\alpha} + 0 \\ 0 + \tilde{\beta} \end{pmatrix} = \tilde{\alpha}^2 + \tilde{\beta}^2$$

It is the squared length of the tangent vector \mathbf{w} in parameter space, by the Pythagorean theorem, and, it is also the squared length of the tangent vector on the surface.

The fact that we get the same number doesn't depend on the choice of the chart. After all, we can have different parametrizations for the same surface and this choice should not give a different result. The length of a vector must be the same all the time.

The interesting thing is that we could compute this length even on a curved surface.

Second example

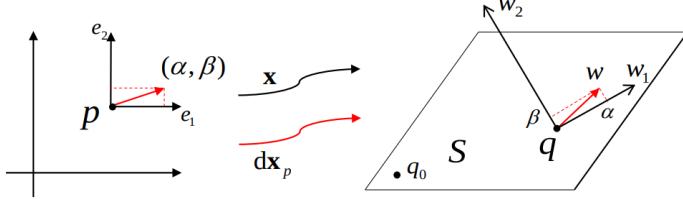
This time we are changing the parametrization $\tilde{\mathbf{x}}$, in particular we are changing the vectors \mathbf{w}_1 and \mathbf{w}_2 , but still we expect that the lengths of vectors in $T_p(S)$ do *not* change (as they are a *property of the surface*, not a property of the chart).

Let $\|\mathbf{w}_1\| = 1$ and $\|\mathbf{w}_2\| = 2$, the vectors \mathbf{w}_1 and \mathbf{w}_2 continue to be orthogonal.

As before, we have $\mathbf{x}_u = \mathbf{w}_1$, $\mathbf{x}_v = \mathbf{w}_2$, so we can compute g :

$$g = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \langle \mathbf{x}_u, \mathbf{x}_u \rangle & \langle \mathbf{x}_u, \mathbf{x}_v \rangle \\ \langle \mathbf{x}_v, \mathbf{x}_u \rangle & \langle \mathbf{x}_v, \mathbf{x}_v \rangle \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}$$

Remark 17. Every vector $\mathbf{w} \in T_p(S)$ is in the form $\mathbf{w} = \alpha \mathbf{x}_u + \beta \mathbf{x}_v$. In the previous example it was $\mathbf{w} = \tilde{\alpha} \tilde{\mathbf{x}}_u + \tilde{\beta} \tilde{\mathbf{x}}_v$. The two bases are *different* in the two example, and thus even the coefficients for the *same* \mathbf{w} are different.



We'll try to express the computation in terms of $\tilde{\alpha}$ and $\tilde{\beta}$. Writing \mathbf{w} in base $\{\tilde{\mathbf{w}}_1, \tilde{\mathbf{w}}_2\}$ we have:

$$\text{first example: } \mathbf{w} = \tilde{\alpha} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \tilde{\beta} \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\text{second example: } \mathbf{w} = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 2 \end{pmatrix}$$

Since the two vectors are linearly independent (they form a base), we can set:

$$\alpha = \tilde{\alpha}$$

Thus:

$$\tilde{\beta} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \beta \begin{pmatrix} 0 \\ 2 \end{pmatrix} \implies \tilde{\beta} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = 2\beta \begin{pmatrix} 0 \\ 1 \end{pmatrix} \implies \tilde{\beta} = 2\beta \implies \beta = \frac{\tilde{\beta}}{2}$$

We can now compute the first fundamental form:

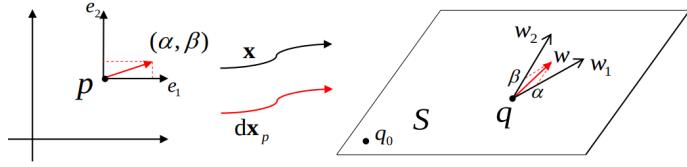
$$\begin{aligned} I_p((\alpha, \beta)) &= (\alpha \quad \beta) \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \\ &= \begin{pmatrix} \tilde{\alpha} & \frac{\tilde{\beta}}{2} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix} \begin{pmatrix} \frac{\tilde{\alpha}}{2} \\ \frac{\tilde{\beta}}{2} \end{pmatrix} \\ &= \begin{pmatrix} \tilde{\alpha} & \frac{\tilde{\beta}}{2} \end{pmatrix} \begin{pmatrix} \tilde{\alpha} \\ 2\tilde{\beta} \end{pmatrix} = \tilde{\alpha}^2 + \tilde{\beta}^2 \end{aligned}$$

Third example

Now let $\|\mathbf{w}_1\| = 1$, $\|\mathbf{w}_2\| = 1$ and $\langle \mathbf{w}_1, \mathbf{w}_2 \rangle = \frac{1}{\sqrt{2}}$, so the vectors \mathbf{w}_1 and \mathbf{w}_2 are no longer orthogonal.

We have $\mathbf{x}_u = \mathbf{w}_1$, $\mathbf{x}_v = \mathbf{w}_2$ and we can compute g as

$$g = \begin{pmatrix} E & F \\ F & G \end{pmatrix} = \begin{pmatrix} \langle \mathbf{x}_u, \mathbf{x}_u \rangle & \langle \mathbf{x}_u, \mathbf{x}_v \rangle \\ \langle \mathbf{x}_v, \mathbf{x}_u \rangle & \langle \mathbf{x}_v, \mathbf{x}_v \rangle \end{pmatrix} = \begin{pmatrix} 1 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 1 \end{pmatrix}$$



Remark 18. Again, we have that $\alpha \mathbf{x}_u + \beta \mathbf{x}_v = \mathbf{w} = \tilde{\alpha} \tilde{\mathbf{x}}_u + \tilde{\beta} \tilde{\mathbf{x}}_v$ the basis vector are different in the two example, and thus even the coefficients are different.

Remark 19. Remember that in Euclidean space:

$$\langle \mathbf{a}, \mathbf{b} \rangle = \|\mathbf{a}\| \|\mathbf{b}\| \cos(\theta)$$

Remark 20. The bisector of \mathbf{a} and \mathbf{b} is given by $\mathbf{a} + \mathbf{b}$.

We will express, again, the computation in terms of $\tilde{\alpha}$ and $\tilde{\beta}$. The first thing to observe is that $\langle \mathbf{w}_1, \mathbf{w}_2 \rangle = \frac{1}{\sqrt{2}}$, so the vectors \mathbf{w}_1 and \mathbf{w}_2 form an angle of 45° .

$$\mathbf{w}_1 = \tilde{\mathbf{w}}_1 \tag{13}$$

$$\mathbf{w}_2 = \frac{\tilde{\mathbf{w}}_1 + \tilde{\mathbf{w}}_2}{\sqrt{2}} \tag{14}$$

in (13) we are fixing \mathbf{w}_1 and $\tilde{\mathbf{w}}_1$ to be equal, this means that \mathbf{w}_2 must be the bisector of $\tilde{\mathbf{w}}_1$ and $\tilde{\mathbf{w}}_2$ (14). Since $\tilde{\mathbf{w}}_1$ and $\tilde{\mathbf{w}}_2$ are the canonical vectors we get $\|\tilde{\mathbf{w}}_1 + \tilde{\mathbf{w}}_2\| = \sqrt{2}$, so to enforce $\|\mathbf{w}_2\| = 1$ we must divide by $\sqrt{2}$.

Thus, we can express \mathbf{w}_1 and \mathbf{w}_2 in base $\{\tilde{\mathbf{w}}_1, \tilde{\mathbf{w}}_2\}$ as:

$$\mathbf{w}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \mathbf{w}_2 = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}$$

So, we must solve for α and β this:

$$\begin{cases} \mathbf{w} = \tilde{\alpha} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \tilde{\beta} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \mathbf{w} = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \beta \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \end{cases} \Rightarrow \begin{cases} \mathbf{w} = \tilde{\alpha} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \tilde{\beta} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \mathbf{w} = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \frac{\beta}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \frac{\beta}{\sqrt{2}} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{cases}$$

Let $\beta = \sqrt{2}\tilde{\beta}$

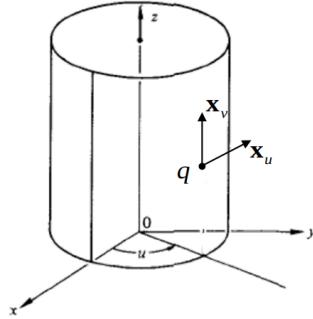
$$\begin{cases} \mathbf{w} = \tilde{\alpha} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \tilde{\beta} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ \mathbf{w} = \alpha \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \tilde{\beta} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \tilde{\beta} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{cases} \implies \alpha = \tilde{\alpha} - \tilde{\beta}$$

So we get:

$$\begin{aligned} I_p((\alpha, \beta)) &= (\alpha \quad \beta) \begin{pmatrix} 1 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = (\tilde{\alpha} - \tilde{\beta} \quad \sqrt{2}\tilde{\beta}) \begin{pmatrix} 1 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 1 \end{pmatrix} \begin{pmatrix} \tilde{\alpha} - \tilde{\beta} \\ \sqrt{2}\tilde{\beta} \end{pmatrix} \\ &= (\tilde{\alpha} - \tilde{\beta} \quad \sqrt{2}\tilde{\beta}) \begin{pmatrix} \tilde{\alpha} \\ \frac{\tilde{\alpha} - \tilde{\beta}}{\sqrt{2}} + \sqrt{2}\tilde{\beta} \end{pmatrix} = (\tilde{\alpha}^2 - \tilde{\alpha}\tilde{\beta}) + (\tilde{\alpha}\tilde{\beta} - \tilde{\beta}^2 + 2\tilde{\beta}^2) \\ &= \tilde{\alpha}^2 + \tilde{\beta}^2 \end{aligned}$$

Remark 21. The length of the vector is equal in the three examples, because the length doesn't depend on the parametrization.

12.4.2 The cylinder



A valid parametrization for an open cylinder is \mathbf{x} , and it yields the set U :

$$\begin{aligned} \mathbf{x}(u, v) &= (\cos u, \sin u, v) \\ U &= \{(u, v) \in \mathbb{R}^2 \mid 0 < u < 2\pi, -\infty < v < \infty\} \end{aligned}$$

We can compute \mathbf{x}_u and \mathbf{x}_v (the columns of the Jacobian, the basis of $T_p(S)$) as:

$$\mathbf{x}_u = \frac{\partial \mathbf{x}}{\partial u} = (-\sin u, \cos u, 0) \quad \mathbf{x}_v = \frac{\partial \mathbf{x}}{\partial v} = (0, 0, 1)$$

Note how $(0, 0, 1)$ is a vector pointing up, and it makes sense.

Thus:

$$\begin{aligned} E &= \langle \mathbf{x}_u, \mathbf{x}_u \rangle = \sin^2 u + \cos^2 u = 1 \\ F &= \langle \mathbf{x}_u, \mathbf{x}_v \rangle = 0 \quad \implies g = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\ G &= \langle \mathbf{x}_v, \mathbf{x}_v \rangle = 1 \end{aligned}$$

The components of the first fundamental forms of the cylinder are equal to the components of the plane. It is weird since we have a non flat surface and we get the same metric tensor as the plane. Getting the same metric tensor as the plane means that we can compute lengths in the same way we did on the plane.

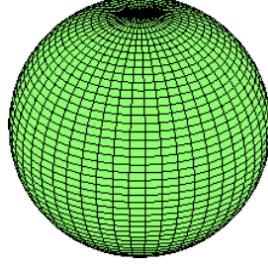
Whenever this happens we say the two surfaces are *locally isometric*.

The way to see this intuitively is that we can take a cylinder, cut it on a side and we can isometrically put it on the plane (without having to stretch it).

They are not *globally isometric*, because the cylinder is not actually open, and a closed cylinder wouldn't have the same metric tensor as the plane, since it isn't isometric to the plane.

12.4.3 The sphere

A complete sphere can't be parametrized with a single chart, since that chart wouldn't be smooth. Intuitively, given a ball it is impossible to stretch and squeeze it so that it stays on the plane, since it is a closed surface. If we make an hole in the sphere it becomes possible.



A valid parametrization for the sphere without the north pole is the following:

$$\mathbf{x} : (0, 2\pi) \times (-\frac{\pi}{2}, \frac{\pi}{2}) \rightarrow \mathbb{R}^3 \quad \mathbf{x}(u, v) = \begin{pmatrix} \cos(u) \cos(v) \\ \sin(u) \cos(v) \\ \sin(v) \end{pmatrix}$$

We can compute the Jacobian of this chart (\mathbf{x}_u and \mathbf{x}_v are the columns):

$$D\mathbf{x} = J_{\mathbf{x}} = \begin{pmatrix} | & | \\ \mathbf{x}_u & \mathbf{x}_v \\ | & | \end{pmatrix} = \begin{pmatrix} -\sin(u) \cos(v) & -\cos(u) \sin(v) \\ \cos(u) \cos(v) & -\sin(u) \sin(v) \\ 0 & \cos(v) \end{pmatrix}$$

Thus:

$$g = D\mathbf{x}^\top D\mathbf{x} = \dots = \begin{pmatrix} \cos^2(v) & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} E & F \\ F & G \end{pmatrix}$$

Here it is evident that E , F and G are indeed *differentiable* functions $E(u, v)$, $F(u, v)$ and $G(u, v)$, thus, the sphere without the pole is a Riemannian manifold.

Thus, if $\mathbf{w} = \alpha\mathbf{x}_u + \beta\mathbf{x}_v$ is the tangent vector to the sphere at point $\mathbf{x}(u, v)$, then its squared length is given by the first fundamental form:

$$\|\mathbf{w}\|^2 = I(w) = E(\alpha)^2 + 2F\alpha\beta + G(\beta)^2 = \dots = \alpha^2 \cos^2(v) + \beta^2$$

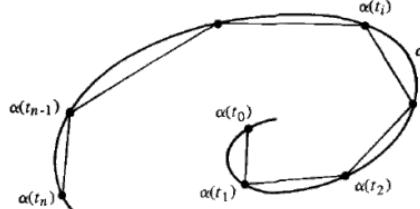
12.5 Length of a curve

Now we have all we need to compute lengths, areas and integrals on surfaces. All we needed to do was a way to compute squared lengths of tangent vectors.

With the first fundamental form, we can treat metric questions on a regular surface without further references to the ambient space.

12.5.1 Arc-length of a curve

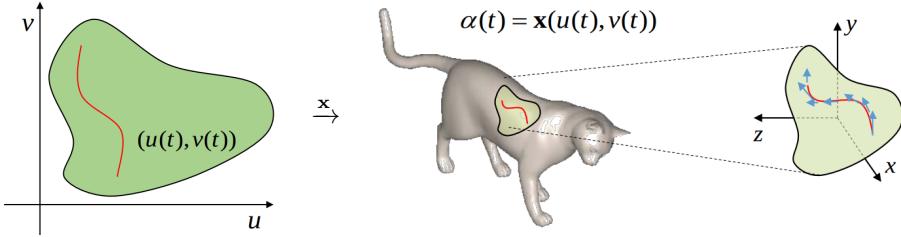
We compute the length of a curve approximating the curve with some vectors and summing up the length of the vectors.



Increasing the number of vectors, we get a more and more accurate representation of the curve. As we keep refining we get to very tiny segments, getting the definition of the derivative of the curve. The total length of the curve is given by the sum of those tiny segments, i.e. the integral seen as a sum on infinitesimal steps.

$$\begin{aligned} \alpha : (0, T) &\rightarrow S \\ s(t) &= \int_0^t \|\alpha'(x)\| dx = \int_0^t \sqrt{I(\alpha'(x))} dx \end{aligned}$$

This is the connection between measuring lengths and the first fundamental form, this is how we use the fact that we can measure the lengths of tangent vectors to measure lengths on a surface.



Thus, if $\alpha(t) = \mathbf{x}(u(t), v(t))$ is contained in a surface element parametrized by $\mathbf{x}(u, v)$, we can compute the length as:

$$s(t) = \int_0^t \sqrt{E(u')^2 + 2F u'v' + G(v')^2} dt$$

Remark 22. Remember that E , F and G are actually functions of (u, v) , so in general they are changing along the curve. So this integral makes sense, because as we go along the curve, as we integrate from 0 to t the components E , F and G change.

12.5.2 Arc-length element

Now, we are simply going to introduce new notation that is the standard notation used in the literature.

We have that:

$$\text{length } s(t) = \int_0^t \|\alpha'(x)\| dx$$

The *first fundamental theorem of calculus* gives us that the integral is the inverse of the derivative, and then, we are doing a notational trick bringing dt to the right (actually it has a profound meaning that we won't cover):

$$\frac{ds}{dt} = \|\alpha'(t)\| \implies ds = \|\alpha'(t)\| dt$$

We get to the compact notation:

$$\text{length}(\alpha) = \int_{\alpha} ds$$

the fact that the integral goes from 0 to t is still present but now it is implicit, and it must be written somewhere else or clear from the context.

In terms of the metric tensor, the *arc length element* ds is given by:

$$ds = \sqrt{E du^2 + 2F du dv + G dv^2} dt$$

May be necessary to deepen the 'first fundamental theorem of calculus'

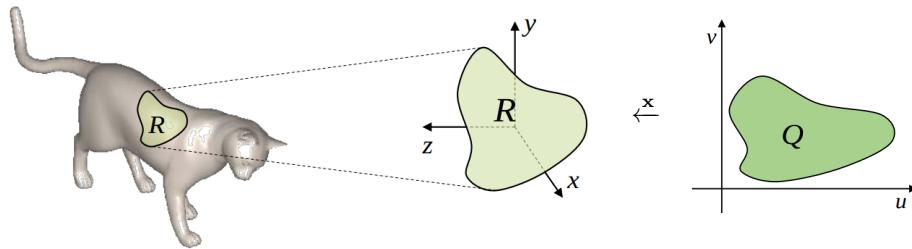
May be necessary to deepen the notational trick that brings dt on the right side

12.6 Area of a region

If $R \subset S$ is contained in the image of the parametrization $\mathbf{x} : U \subset \mathbb{R}^2 \rightarrow S$, the area of R is defined by:

$$A(R) = \iint_Q \|\mathbf{x}_u \times \mathbf{x}_v\| dv du \quad Q = \mathbf{x}^{-1}(R)$$

May be necessary to deepen the definition and notation of 'surface integral'

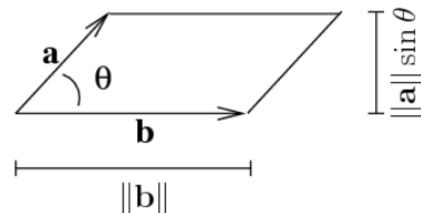


Remark 23 (norm cross product).

The norm of the cross product can be proved to be:

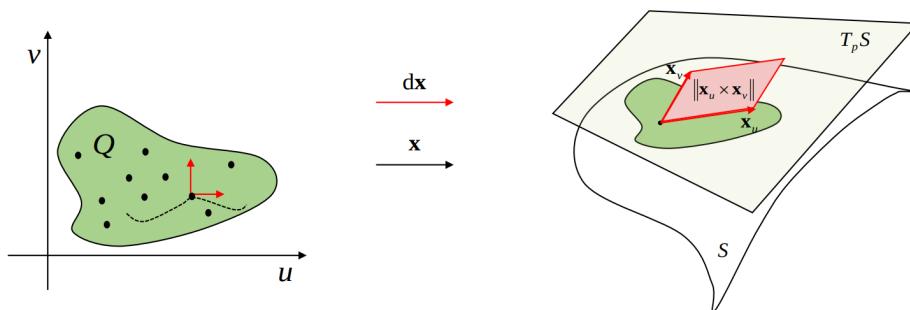
$$\|\mathbf{a} \times \mathbf{b}\| = \|\mathbf{a}\| \|\mathbf{b}\| \sin \theta$$

i.e. it is the area of the parallelogram described by a and b :



May be necessary to prove that the norm of the cross product is the area of the parallelogram

The *area of a region* on the surface is defined as the sum of the areas of *parallelograms* tangent to that surface region, it is the same principle used to measure lengths on surfaces.



So we move all points (u, v) on the parameter domain, for any point we construct a parallelogram \mathbf{x}_u and \mathbf{x}_v , take the norm of the cross product, integrate over all of them and sum them up. The double integral is actually summing up all these tiny parallelograms, it is the continuous way to say “sum over u and sum over v ”, in a infinitesimal small steps.

Now that we understand the intuition, we want to write this expression in a more readable way. We can observe that:

$$\|\mathbf{x}_u \times \mathbf{x}_v\|^2 = \|\mathbf{x}_u\|^2 \|\mathbf{x}_v\|^2 \sin^2 \omega \quad (15)$$

$$= \|\mathbf{x}_u\|^2 \|\mathbf{x}_v\|^2 (1 - \cos^2 \omega) \quad (16)$$

$$= \|\mathbf{x}_u\|^2 \|\mathbf{x}_v\|^2 - \|\mathbf{x}_u\|^2 \|\mathbf{x}_v\|^2 \cos^2 \omega \\ = \|\mathbf{x}_u\|^2 \|\mathbf{x}_v\|^2 - \langle \mathbf{x}_u, \mathbf{x}_v \rangle^2 \quad (17)$$

where (15) is a property of the norm of the cross product (Remark 23), (16) is a known equivalence and (17) is the definition of inner product on a Euclidean vector space $\langle \mathbf{a}, \mathbf{b} \rangle = \|\mathbf{a}\| \|\mathbf{b}\| \cos \theta$.

We can then rewrite:

$$\begin{aligned} \|\mathbf{x}_u \times \mathbf{x}_v\| &= \sqrt{\|\mathbf{x}_u\|^2 \|\mathbf{x}_v\|^2 - \langle \mathbf{x}_u, \mathbf{x}_v \rangle} \\ &= \sqrt{EG - F^2} \end{aligned} \quad (18)$$

$$= \sqrt{\det g} \quad (19)$$

where (18) is due to Theorem 3 and (19) is the formula for the determinant in a 2×2 matrix.

Thus $\sqrt{\det g}$ is the area of the parallelogram described by \mathbf{x}_u and \mathbf{x}_v .

Theorem 3. Let \mathbf{u} be a vector in the real vector space \mathbb{R}^n . Then:

$$\langle \mathbf{u}, \mathbf{u} \rangle = \|\mathbf{u}\|^2$$

Proof. Let $\mathbf{u} = (u_1, u_2, \dots, u_n)$, then:

$$\begin{aligned} \langle \mathbf{u}, \mathbf{u} \rangle &= u_1 u_1 + u_2 u_2 + \dots + u_n u_n \\ &= u_1^2 + u_2^2 + \dots + u_n^2 \\ &= \left(\sqrt{\sum_{i=1}^n u_i^2} \right)^2 \\ &= \|\mathbf{u}\|^2 \end{aligned}$$

□

We get the compact expression:

$$A(R) = \iint_Q \sqrt{\det g} \, dv \, du$$

12.6.1 Area element

We define the *area element* da as:

$$da = \sqrt{\det g} dv du \quad (20)$$

Leading to:

$$A(R) = \int_R da$$

The area element is also called (*Riemannian*) *volume form*. “Volume” because it works even with high dimensional manifold, and the general name for area is volume. In the case of 2-dimensional manifolds, volume corresponds to *area*.

12.7 Summary

To wrap-up, we have two alternative expression for measuring lengths and areas: one in *parameter space* the other directly on the *surface*

$$\text{Parameter space} \quad \text{length}(\alpha) = \int_0^T \|\alpha'(t)\| dt = \int_0^T \sqrt{E du^2 + 2F dv du + G dv^2} dt$$

$$\text{Surface} \quad \text{length}(\alpha) = \int_\alpha ds \quad ds = \sqrt{E du^2 + 2F dv du + G dv^2} dt$$

$$\text{Parameter space} \quad A(R) = \iint_Q \|\mathbf{x}_u \times \mathbf{x}_v\| dv du \quad Q = \mathbf{x}^{-1}(R)$$

$$\text{Surface} \quad A(R) = \int_R da \quad da = \sqrt{\det g} dv du$$

12.8 Integral of a function

We have the *definition*:

$$\int_R f(x) dx = \iint_Q f(\mathbf{x}(u, v)) \sqrt{\det g} dv du \quad Q = \mathbf{x}^{-1}(R)$$

where x is a point on the surface, that comes from some point (u, v) . So instead of x we can write $\mathbf{x}(u, v)$, instead of dx we have the area element $\sqrt{\det g} dv du$, and we have the double integral in the parameter domain.

There isn't nothing much to understand here, it makes sense, but, it is a definition.

In general, one takeaway message, is that when we look at integrals over surfaces, whenever we see dx we can think of it as the area element, the area of a tiny parallelogram at the point given by $\sqrt{\det g}$.

Remark 24. Generalizes the substitution rule in classical multivariate calculus.

12.9 Discretization

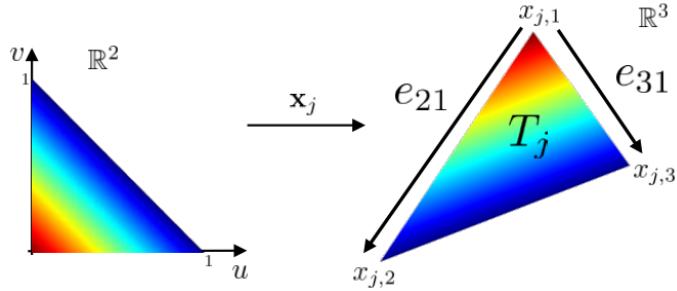
Now we will see how to compute these things on triangle meshes.

12.9.1 Metric tensor

We gave a discretization for the charts. Our charts for triangle meshes are one per triangle, and we have an explicit expression:

$$\mathbf{x}_j(u, v) = x_{j,1} + u(x_{j,2} - x_{j,1}) + v(x_{j,3} - x_{j,1})$$

that takes a point (u, v) from some canonical triangle in \mathbb{R}^2 and gives points in \mathbb{R}^3 .



Now we have an explicit expression for the chart, so we can compute explicitly the discrete \mathbf{x}_u and \mathbf{x}_v , computing the partial derivatives:

$$\begin{aligned}\mathbf{x}_{j_u} &= \frac{\partial \mathbf{x}_j}{\partial u} = x_{j,2} - x_{j,1} = e_{21} \\ \mathbf{x}_{j_v} &= \frac{\partial \mathbf{x}_j}{\partial v} = x_{j,3} - x_{j,1} = e_{31}\end{aligned}$$

As a simple check: \mathbf{x}_u and \mathbf{x}_v should span the tangent space at that point. They are the edges of a triangle, and obviously they span the tangent space where the triangle is contained. So this works.

The coefficients for the metric tensor are thus given by:

$$g_j = \begin{pmatrix} E_j & F_j \\ F_j & G_j \end{pmatrix} = \begin{pmatrix} \langle \mathbf{x}_{j_u}, \mathbf{x}_{j_u} \rangle & \langle \mathbf{x}_{j_u}, \mathbf{x}_{j_v} \rangle \\ \langle \mathbf{x}_{j_v}, \mathbf{x}_{j_u} \rangle & \langle \mathbf{x}_{j_v}, \mathbf{x}_{j_v} \rangle \end{pmatrix} = \begin{pmatrix} \|e_{21}\|^2 & \langle e_{21}, e_{31} \rangle \\ \langle e_{21}, e_{31} \rangle & \|e_{31}\|^2 \end{pmatrix}$$

12.9.2 Area element

Now we want to discretize the area element. The area of the triangle is the area of a region:

$$\int_{T_j} da = \int_0^1 \int_0^{1-u} \sqrt{\det g_j} dv du \quad (21)$$

$$= 2A(T_j) \int_0^1 \int_0^{1-u} dv du \quad (22)$$

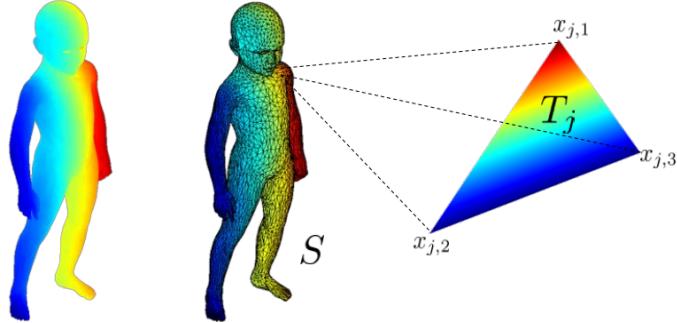
$$= 2A(T_j) \frac{1}{2} = A(T_j) \quad (23)$$

where

- in (21) we are decomposing the integral over the triangle T_j into two integrals in the parameter space, and, we are using the definition (20) of da . Note that with $u \in [0, 1]$ and $v \in [0, 1 - u]$ we can cover all the canonical triangle
- in (22) we are exploiting the fact that $\sqrt{\det g} = \|\mathbf{x}_u \times \mathbf{x}_v\|$ (proven in (19)). We know that the norm of the cross product is the area of the parallelogram described by \mathbf{x}_u and \mathbf{x}_v (Remark 23), thus, this quantity is two times the area of the triangle described by \mathbf{x}_u and \mathbf{x}_v
- in (23) we are just computing the integral. It is the area of a right-angled triangle of side 1, so it equals $\frac{1}{2}$

We get that the area of the triangle is equal to the area of the triangle, everything works.

12.9.3 Integral of a function



We have the usual assumption that our functions $f : S \rightarrow \mathbb{R}$ behaves piece-wise linearly on the mesh, meaning that they behave *linearly* within each triangle. Which means that the behavior within the triangle is *uniquely* determined by the values of f at the vertices of the triangle.

We get that:

$$\int_{T_j} f \, da = \int_0^1 \int_0^{1-u} f(\mathbf{x}(u, v)) \sqrt{\det g_j} \, dv \, du \quad (24)$$

$$= \int_0^1 \int_0^{1-u} f(x_{j,1})(1-u-v) + \\ + f(x_{j,2})u + f(x_{j,3})v \sqrt{\det g_j} \, dv \, du \quad (25)$$

$= \dots$

$$= \frac{1}{6}(f(x_{j,1}) + f(x_{j,2}) + f(x_{j,3}))2A(Tj) \quad (26)$$

$$= \frac{1}{3}(f(x_{j,1}) + f(x_{j,2}) + f(x_{j,3}))A(Tj) \quad (27)$$

where

- (24) is the definition of the integral of a function over a region, and the region is the triangle.
- (25) is the bilinear interpolation formula. If we have the value of f at the three vertices, we can get the value of f at intermediate points on our domain by applying this formula, where u and v vary within the triangle in the parameter domain. It's easy to check that for the three vertices ($v_1 = \{u = v = 0\}$, $v_2 = \{u = 1, v = 0\}$ and $v_3 = \{u = 0, v = 1\}$) it works, the other points in the middle are just linearly interpolated following our assumption on the functions.
- in (26) we are computing the integral. It is fairly easy to do since the integral is a linear operator, thus the integral of a sum is the sum of the integrals. Again the area element represent the area of a parallelogram, so two times the area of the triangle.

Basically, what this formula is saying is that the integral of a function within a triangle, if the functions behaves linearly within the triangle, is just given by the sum of $\frac{1}{3}$ the value of f at the three vertices times the area of the triangle.

This is a bit surprising, it's like separating the triangle in three regions and giving equal weight to each region. It's like having a function which is constant within this region and it's value is $\frac{1}{3}$ the value of f at the vertex.

This is the direct consequence that also the integral is a linear function, other than the function f within the triangle.



So, we know how to compute the integral of a function within a triangle, of course we want to compute integral of functions on a region of the entire mesh.

The integral of f over a region $R \subseteq S$ is just the sum:

$$\int_R f \, da = \sum_{j=1}^{|R|} \int_{T_j} f \, da$$

We are just computing the integral for each triangle in the region and then summing up.

There is a smart way to compute this quantity. Given the vertex p , we can take the region formed by connecting the barycenter of neighbors triangles, this is by the way the voronoi region of p . The area element of vertex p is just the sum of $\frac{1}{3}$ the area of each neighbouring triangle. Now we can form a diagonal matrix and placing along the diagonal these numbers, call this matrix \mathbf{A} . When we want to compute the integral of a function we take the function, discretize it as usual (with function values at each vertex), call this discretized function f . The operation \mathbf{Af} gives a vector, where each function value is weighted by the area element, and we sum up the vector of this vector $\mathbf{1}^\top \mathbf{Af}$. This is equivalent to the integral of a function over a region.

May be necessary
to add a paragr
'area element of
vertices'

Another way of seeing this, if we didn't know about areas and we were asked to compute the integral of a function, the simplest thing we would do is just sum up the values of this functions. The integral is a sum, so we just take the values of the function and then we sum up the values. Of course by doing this we are not taking in account the fact that we are operating on a surface, so we are not taking in account the local distortion or the local stretching and so on. The way to do this is just by introducing the local areas at each point.

Another way to realize this is that, if instead of a surface we are on the plane, we can still apply this operation, taking the sum of the function values weighted by the local areas, but then, we realize that on the plane local areas are all exactly the same. So all we are doing is just summing up the values of the function, and then applying some global scale to the result. If we are not on the plane but on a curved surface, it is not the same as applying a global scale, because each point has a different local area.

Note that this area element is, actually, the *discretized version of the area element* da . It can be seen from the equation (27). A naive discretization of the integral of a function on a triangle is the sum of that function of the vertices, in (27) there is also $\frac{1}{3}A(T_j)$, the area element.

We could compute the area element also using the discretized version of g , using the definition of da but it would be more unhandy. However, the result would be the same.

13 The gradient

In this section we will use symbols with the tilde, e.g. \tilde{f} , to indicate functions on the parametric domain and symbols without the tilde, e.g. f , to indicate functions on the surface.

13.1 Gradient of a function

Consider a surface S with parametrization $\mathbf{x} : U \rightarrow S$ and a *differentiable function* $f : S \rightarrow \mathbb{R}$.

We want to define the *gradient* $\nabla f(p)$ at a point $p \in S$, as in Figure 28.

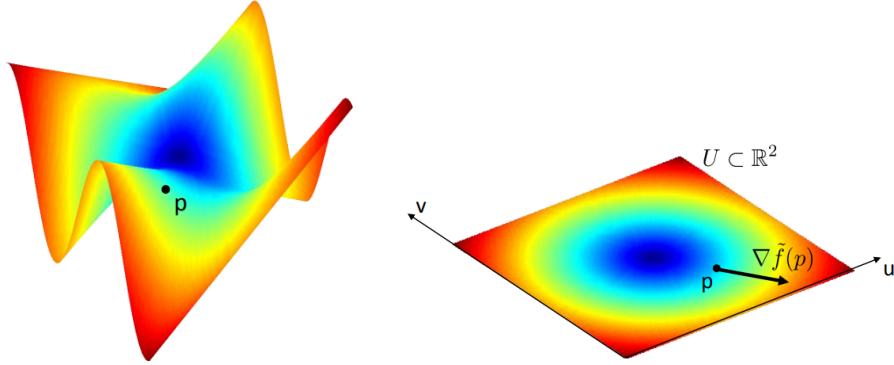


Figure 28: A point p on a surface

Figure 29: Vector field in \mathbb{R}^2

13.2 The gradient in \mathbb{R}^2

The gradient of a differentiable function $\tilde{f} : U \rightarrow \mathbb{R}$ is the vector field shown in Figure 29:

$$\nabla \tilde{f}(p) = \begin{pmatrix} \frac{\partial \tilde{f}}{\partial u}(p) \\ \frac{\partial \tilde{f}}{\partial v}(p) \end{pmatrix} \quad (28)$$

In Figure 29, assuming blue indicates small values and red high values, the partial derivative $\frac{\partial \tilde{f}}{\partial u}(p)$ points to the right and $\frac{\partial \tilde{f}}{\partial v}(p)$ points down. They define the coordinates of the gradient at p , that points in the direction of steepest increase.

13.3 The gradient on a surface

We will get to the definition of gradient on a surface through several -wrong-attempts to define it.

Idea 1

We could define the gradient using the same formula as before, but instead of \mathbb{R}^2 we are in \mathbb{R}^3 . So we consider the vector of partial derivatives in \mathbb{R}^3 :

$$\nabla f(p) = \begin{pmatrix} \frac{\partial f}{\partial x}(p) \\ \frac{\partial f}{\partial y}(p) \\ \frac{\partial f}{\partial z}(p) \end{pmatrix}$$

It isn't a good choice:

- In order to define the vectors in \mathbb{R}^3 we must define the coordinates of the embedding space at p (Figure 30). We are saying embedding space, so whatever I'm defining here will be a property of the embedding and not of the surface, and, we don't like this
- We have *no information* about f outside of the surface S (Figure 30), e.g. computing the partial derivative $\frac{\partial f}{\partial z}(p)$ means taking the value of f at p and the value of f slightly above p , but f is not defined above p since f is defined only on the surface.

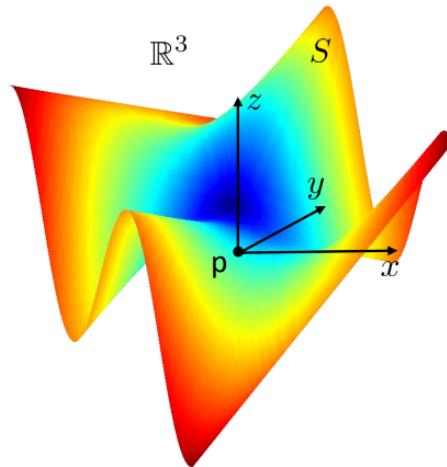


Figure 30: The classic gradient is not well defined on a surface

This definition only works when functions are defined entirely in \mathbb{R}^3 , i.e. with euclidean objects. In fact in multivariate calculus this is how we define the gradient of a function in \mathbb{R}^3 , but we are working on subsets of \mathbb{R}^3 , with surfaces.

Idea 2

Another possibility, now it is going in the right direction. We know that our surfaces are manifolds, so they are paratrizable by some parametrization function, by some chart \mathbf{x} . It means that for function f defined on the surface, we have a pre-image, according on the chart, defined on a subset of the plane (Figure 31).

We express our differentiable function f on the surface, in terms of a parametrization \mathbf{x} defining a function \tilde{f} on the parameter space:

$$\tilde{f}(u, v) = f(\mathbf{x}(u, v))$$

and then we could claim that our gradient is just the gradient computed on the parametric domain:

$$\nabla f(p) = \begin{pmatrix} \frac{\partial \tilde{f}}{\partial u}(p) \\ \frac{\partial \tilde{f}}{\partial v}(p) \end{pmatrix}$$

we are allowed to do this since the coefficients of tangent vectors are the same both in the parametric space and on the surface.

It doesn't work. It depends on the choice of the *parametrization* (Figure 31), since we are defining \tilde{f} in terms of the chart \mathbf{x} .

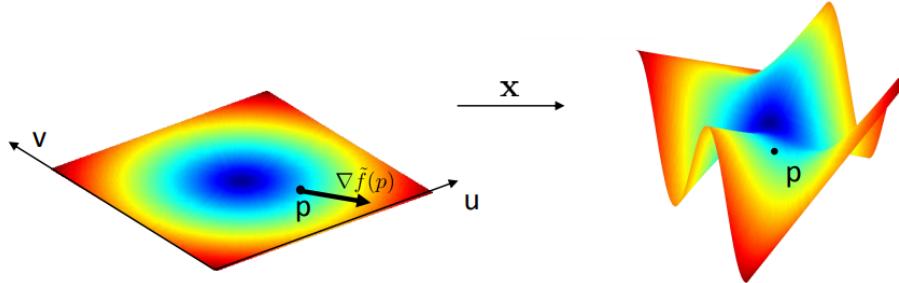


Figure 31: Changing the parametrization would change the gradient on S

13.3.1 Geometric meaning

Let us try to interpret the *geometric meaning* of the gradient:

- The vector that points in the *direction of steepest increase* of f
- Its length should measures the strength of increase e.g. in standard calculus, if the derivative is 10000 it means that there is a very steep increase of the function, if the derivative is close to zero means the function is almost flat. The largest the derivative the steepest the increase. In the general case the gradient is not just a number, as the derivative, but a vector that has a length
- We have a relationship with the *directional derivative* (Definition 11). We are specifying in which direction we want to look at the “increase over run” (i.e. the definition of standard derivative) of the function. In 1D we only have one choice of direction, in 2D and 3D we have infinite choices, we can choose whichever direction we like, and the direction is identified by a vector. So, the directional derivative is just the derivative computed along a given direction

Definition 11 (directional derivative). The directional derivative of a scalar function $f(p) = f(p_1, p_2, \dots, p_n)$ along the direction of the vector $\vec{v} = (v_1, \dots, v_n)$ at point p is the function $df_p(\vec{v})$ (also known as $\nabla_{\vec{v}}f(p)$):

$$\begin{aligned} df_p(\vec{v}) &= \lim_{h \rightarrow 0} \frac{f(p + h\vec{v}) - f(p)}{h} \\ &= \frac{d}{dh} f(p + h\vec{v})|_{h=0} \\ &= \dots \\ &= \langle \nabla f, \vec{v} \rangle \end{aligned} \tag{29}$$

where $p + h\vec{v}$ is just a small step from p in the direction of \vec{v} , and, (29) is a classical result from calculus, that says the directional derivative is just the projection of ∇f along the direction \vec{v} .

Remark 25. Do not think at point and vectors as matrices, that is just a representation problem. A point is a point, a vector is an arrow.

13.3.2 Representation theorem

The gradient of any differentiable function f can be *defined* in terms of the directional derivative as the *unique vector field* ∇f such that the following relationship holds:

$$\langle \nabla f, \vec{v} \rangle = df_p(\vec{v}) \quad \text{where } df_p(\vec{v}) : T_p(S) \rightarrow \mathbb{R} \tag{30}$$

This is an application of the *Riesz representation theorem*, that ensures ∇f exists and it's unique.

Remark 26. We are calling the gradient vector field, because at each point there is a vector, it is a collection of vectors on the domain.

May be necessary
to deepen the
'Riesz represen-
tation theorem'

13.3.3 Definition of the gradient

We are going to use the definition (30) for defining the gradient of a function on a surface. It is a good idea to use this definition since we have a well established notion of a direction on a surface, we talked about tangent directions (i.e. tangent vectors) all the time, so we know exactly what \vec{v} means on a surface: some vector on the tangent space.

Thus, we define the gradient $\nabla f(p) \in T_p(S)$ by means of the inner product:

$$I_p(\nabla f, \vec{v}) = \langle \nabla f, \vec{v} \rangle = df_p(\vec{v}) \quad \forall \vec{v} \in T_p(S) \quad (31)$$

where in (31) I_p refers to the generalized first fundamental form (Section 12.2.2).

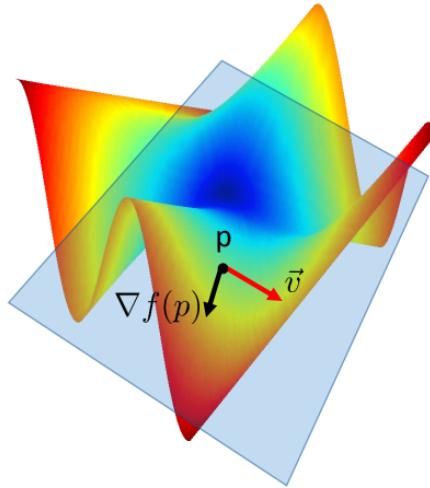


Figure 32: The tangent plane $T_p(S)$

Thus, if we are able to write down $df_p(\vec{v})$, we can *solve for* ∇f .

We are shifting the problem for defining a gradient to computing directional derivatives. It looks a similar problem, but it turns out that we can do this completely in the parameter space in \mathbb{R}^2 .

Remark 27. In standard calculus the gradient is perpendicular to the tangent of the level curve in that point. On surfaces, it is exactly the same (not proved).

Note that, there will be another curve in the parameter space correspondent to the curve level, and, there will be a vector in the parameter space with the same coefficients of the gradient. However these quantities pulled-back in the parameter space depend on the choice of the chart, so they are not anymore a curve level and a gradient, they aren't interesting.

13.4 The gradient in local coordinates

Since the gradient is a member of the tangent plane $T_p(S)$, we can express it in the local basis of $T_p(S)$ (Figure 33):

$$\nabla f(p) = f_1 \mathbf{x}_u + f_2 \mathbf{x}_v = D\mathbf{x} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = J_{\mathbf{x}} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}$$

Basically to obtain $\nabla f(p)$ we need to solve for f_1 and f_2 , the two coefficients of the gradient in this basis.

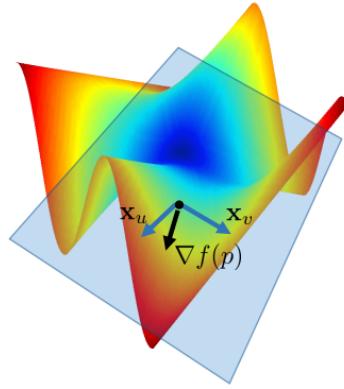
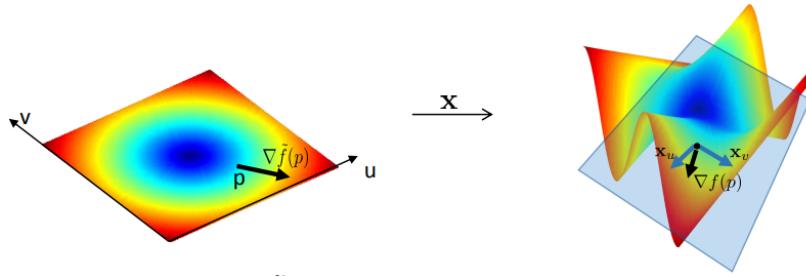


Figure 33: The gradient can be expressed in the local basis of $T_p(S)$

It turns out that the coefficients f_1 and f_2 can be obtained by considering the gradient of $\tilde{f} = f \circ \mathbf{x} : U \rightarrow \mathbb{R}$.



We will work on the parametric domain. We defined the function \tilde{f} which is basically the function f pulled back to \mathbb{R}^2 , the parameter domain, and for \tilde{f} we have some gradient. We are interested in the gradient on the surface.

Everything is based on this observation: the change of \tilde{f} in the direction of u and v should be the same as the change of f in the direction of \mathbf{x}_u and \mathbf{x}_v .

This is because if we bring u and v , the local basis in \mathbb{R}^2 , in the differential on the surface we get \mathbf{x}_u and \mathbf{x}_v . So the change of f along the direction of u should be the same as the change in the direction of \mathbf{x}_u . Thus, we can write:

$$\frac{\partial \tilde{f}}{\partial u} = \frac{\partial(f \circ x)}{\partial u} \quad (32)$$

$$= \frac{\partial f}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial u} \quad (33)$$

$$= df_p(\mathbf{x}_u) \quad (34)$$

where (32) is due to the definition of \tilde{f} , (33) is due to the chain rule (Definition 10) and (34) is due to the Riesz theorem.

We can see immediately from (34) that the directional derivative of f in the direction of \mathbf{x}_u is the partial derivative $\frac{\partial \tilde{f}}{\partial u}$, and same thing for v . Hence, we can compute the directional derivative df_p directly in parameter space.

Now, let $\vec{v} = v_1 \mathbf{x}_u + v_2 \mathbf{x}_v$ be a generic direction on the tangent plane and apply df_p on both sides:

$$df_p(\vec{v}) = v_1 df_p(\mathbf{x}_u) + v_2 df_p(\mathbf{x}_v) \quad (35)$$

$$= v_1 \frac{\partial \tilde{f}}{\partial u} + v_2 \frac{\partial \tilde{f}}{\partial v} \quad (36)$$

$$= \begin{pmatrix} \frac{\partial \tilde{f}}{\partial u} & \frac{\partial \tilde{f}}{\partial v} \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (37)$$

where (35) is due to the linearity of df_p (just like the usual derivative), (36) is due the previous result in (34), and (37) is due to the definition of the gradient in \mathbb{R}^2 , the parameter space, in equation (28).

This is not surprising, we know that the directional derivatives come from inner products of gradient with direction. The surprising thing, is that the gradient is computed in the parameter space, in \mathbb{R}^2 .

On the other hand (left side of (31)), given the definition of the generalized first fundamental form (Section 12.2.2) we can also write:

$$df_p(\vec{v}) = I_p(\nabla f, \vec{v}) = \begin{pmatrix} f_1 & f_2 \end{pmatrix} g \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (38)$$

where the unknown that we want to find are f_1 and f_2 , the coefficients of ∇f .

Thus, from (37) and (38), it must hold:

$$\begin{aligned}
(\nabla \tilde{f})^\top \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} &= (f_1 \ f_2) g \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \\
\implies (\nabla \tilde{f})^\top &= (f_1 \ f_2) g \\
\implies \nabla \tilde{f} &= ((f_1 \ f_2) g)^\top \\
\implies \nabla \tilde{f} &= g^\top \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \tag{39}
\end{aligned}$$

$$\begin{aligned}
\implies \nabla \tilde{f} &= g \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \tag{40} \\
\implies \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} &= g^{-1} \nabla \tilde{f}
\end{aligned}$$

where (39) is due to the fact that $(AB)^\top = B^\top A^\top$, and (40) is due to the fact that g is symmetric so $g^\top = g$

13.5 The gradient in global coordinates

The expression

$$\begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = g^{-1} \nabla \tilde{f}$$

is giving the coefficients of ∇f with respect to a basis in $T_p(S)$, hence it is said to be given in *local coordinates*.

Often we need a vector $\nabla f \in \mathbb{R}^3$, for example to be able to plot it. This vector is said to be in *global coordinates* and to compute it we just use f_1 and f_2 as coefficients of the basis vectors:

$$\nabla f = f_1 \mathbf{x}_u + f_2 \mathbf{x}_v = D\mathbf{x} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = D\mathbf{x} g^{-1} \nabla \tilde{f}$$

since the basis vectors are in \mathbb{R}^3 , we get a vector in \mathbb{R}^3 .

Note that \mathbf{x}_u and \mathbf{x}_v are expressed in the standard basis in \mathbb{R}^3 , meanwhile f_1 and f_2 are expressed in base $\{\mathbf{x}_u, \mathbf{x}_v\}$. We are just changing the base of f_1 and f_2 to the standard base in \mathbb{R}^3 .

13.6 Summary

Thus, according to our definition of the gradient, we have to find the unique ∇f such that:

$$\langle \nabla f, \vec{v} \rangle = df_p(\vec{v})$$

We can compute the directional derivative directly in U , as:

$$df_p(\vec{v}) = (\nabla \tilde{f})^\top \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad \tilde{f} = f \circ x$$

where f is defined on the surface S , and, \tilde{f} is defined on the parameter domain U .

We can thus write:

$$\langle \nabla f, \vec{v} \rangle = (\nabla \tilde{f})^\top \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (41)$$

Using the bilinear definition of the first fundamental form, we can also write

$$\langle \nabla f, \vec{v} \rangle = I_p(\nabla f, \vec{v}) = (f_1 \quad f_2) g \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

Together with (42) we have:

$$(\nabla \tilde{f})^\top \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = (f_1 \quad f_2) g \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (42)$$

and thus we finally obtain:

$$\begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = g^{-1} \nabla \tilde{f}$$

13.7 Gradient norm

In some application we need the *norm* of the gradient. Our final expression for the gradient is given by:

$$\nabla f = D\mathbf{x} g^{-1} \nabla \tilde{f} \quad (43)$$

Computing its (squared) norm is straightforward:

$$\|\nabla f\|^2 = \nabla f^\top \nabla f \quad (44)$$

$$= (D\mathbf{x} g^{-1} \nabla \tilde{f})^\top (D\mathbf{x} g^{-1} \nabla \tilde{f}) \quad (45)$$

$$= \nabla \tilde{f}^\top g^{-1} \underbrace{D\mathbf{x}^\top D\mathbf{x}}_g g^{-1} \nabla \tilde{f} \quad (46)$$

$$= \nabla \tilde{f}^\top g^{-1} \nabla \tilde{f} \quad (47)$$

where (44) is a classical way to compute the squared norm of a vector, (45) is due to the definition (43), (46) is due to the fact that $(ABC)^\top = C^\top B^\top A^\top$ and Theorem 4, and (47) is the classical $g^{-1}g = I$. Just a weighted inner product.

Theorem 4. *Given A nonsingular (i.e. it has an inverse), symmetric matrix:*

$$A^{-1} = (A^{-1})^\top$$

Proof.

$$\text{Since } I = I^\top \text{ and } AA^{-1} = I \implies AA^{-1} = (AA^{-1})^\top$$

$$\text{Since } (AB)^\top = B^\top A^\top \implies AA^{-1} = (A^{-1})^\top A^\top$$

$$\text{Since } AA^{-1} = A^{-1}A = I \implies A^{-1}A = (A^{-1})^\top A^\top$$

$$\text{Since } A = A^\top \implies A^{-1}A = (A^{-1})^\top A$$

$$\begin{aligned} &\implies A^{-1}A(A^{-1}) = (A^{-1})^\top A(A^{-1}) \\ &\implies A^{-1}I = (A^{-1})^\top I \implies A^{-1} = (A^{-1})^\top \end{aligned}$$

□

13.8 Example: The sphere

Consider the function $f : \mathbb{S}^2 \setminus \{n\} \rightarrow \mathbb{R}$ on a sphere that assigns to each point its distance to the north pole n (Figure 34):

$$f(p) = d_{\mathbb{S}^2}(n, p)$$

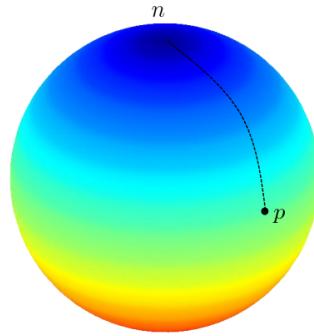


Figure 34: The distance of p from the north pole

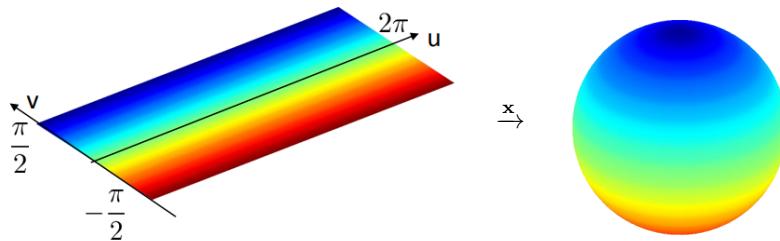
We want to compute its gradient, and, we expect to get a tangent arrow at p pointing down:

$$\nabla f = D\mathbf{x} g^{-1} \nabla \tilde{f}$$

We consider the parametrization:

$$\mathbf{x} : (0, 2\pi) \times (-\frac{\pi}{2}, \frac{\pi}{2}) \rightarrow \mathbb{R}^3 \quad \mathbf{x}(u, v) = \begin{pmatrix} \cos(u) \cos(v) \\ \sin(u) \cos(v) \\ \sin(v) \end{pmatrix}$$

Remark 28. Remember that the north pole $n \notin \mathbb{S}$, since we want to parametrize the sphere with a single chart for simplicity.



Given this particular parametrization, the function in the parameter space is the following:

$$\tilde{f}(u, v) = \frac{\pi}{2} - v$$

where the distance of a point p from the north pole changes from 0 to π , and only one axis is relevant.

13.8.1 Gradients in \mathbb{R}^2

Given the function in the parameter space, we can compute the gradient in the parameter space (Figure 35):

$$\begin{aligned}\tilde{f}(u, v) &= \frac{\pi}{2} - v \\ \nabla \tilde{f}(u, v) &= \begin{pmatrix} 0 \\ -1 \end{pmatrix}\end{aligned}$$

Remark 29. The gradient is constant and equal in every point only by chance, because the function is very simple.

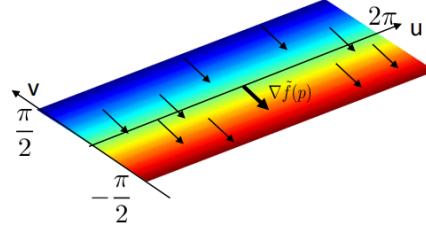


Figure 35: The gradient of \tilde{f} in the parameter space

13.8.2 Gradients on the surface

Now we want to compute the gradient of f on the surface at point $(-1, 0, 0)$:

$$\nabla f(-1, 0, 0) = ?$$

To compute this gradient, firstly we observe that the corresponding point on the parameter space is $(\pi, 0)$:

$$\mathbf{x}(\pi, 0) = \begin{pmatrix} \cos(\pi) \cos(0) \\ \sin(\pi) \cos(0) \\ \sin(0) \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 0 \end{pmatrix}$$

and note that the gradient in that point, as everywhere, is:

$$\nabla \tilde{f}(\pi, 0) = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

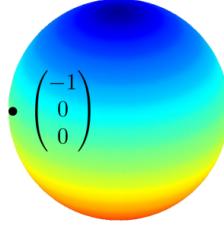


Figure 36: A point p on the surface

Now, we must compute the basis of the tangent plane $T_p(S)$ (i.e. the columns of the Jacobian of the chart, evaluated at p).

The Jacobian is:

$$\begin{aligned} D\mathbf{x}(u, v) &= \begin{pmatrix} \frac{d}{du} \cos(u) \cos(v) & \frac{d}{dv} \cos(u) \cos(v) \\ \frac{d}{du} \sin(u) \cos(v) & \frac{d}{dv} \sin(u) \cos(v) \\ \frac{d}{du} \sin(v) & \frac{d}{dv} \sin(v) \end{pmatrix} \\ &= \begin{pmatrix} \sin(u)(-\cos(v)) & -\cos(u) \sin(v) \\ \cos(u) \cos(v) & -\sin(u) \sin(v) \\ 0 & \cos(v) \end{pmatrix} \end{aligned}$$

evaluating at point p :

$$D\mathbf{x}(\pi, 0) = \begin{pmatrix} 0 & 0 \\ -1 & 0 \\ 0 & 1 \end{pmatrix}$$

Hence, we are able to compute g :

$$\begin{aligned} g_{\mathbf{x}}(\pi, 0) &= D\mathbf{x}(\pi, 0)^T D\mathbf{x}(\pi, 0) \\ &= \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ -1 & 0 \\ 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = (g_{\mathbf{x}}(\pi, 0))^{-1} \end{aligned} \tag{48}$$

where (48) is because the inverse of the identity is the identity.

Thus, we can compute the gradient:

$$\begin{aligned} \nabla f(-1, 0, 0) &= D\mathbf{x} g_{\mathbf{x}}^{-1} \nabla \tilde{f}(\pi, 0) \\ &= D\mathbf{x} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ -1 \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \end{aligned}$$

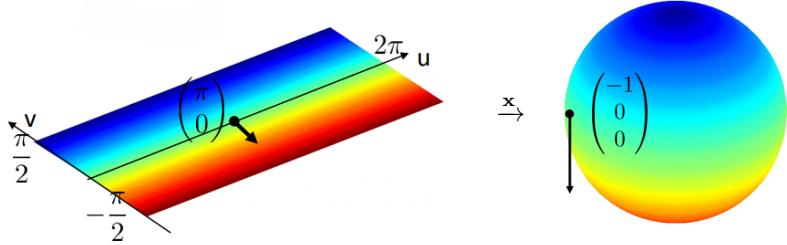


Figure 37: The gradient in the parameter space and on the surface

We can see that it is what we would intuitively expect in Figure 37. If we had chosen another parametrization for the sphere, we would have always got the same gradient. The gradient doesn't depend on the choice of the parametrization.

13.9 Discretization

Remember how we defined a chart for each triangle in the mesh (Figure 38):

$$\mathbf{x}_j(u, v) = x_{j,1} + u(x_{j,2} - x_{j,1}) + v(x_{j,3} - x_{j,1})$$

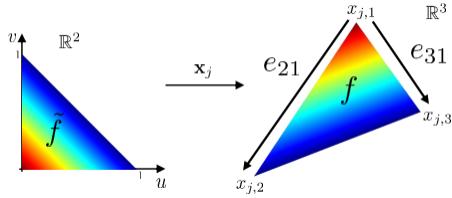


Figure 38: Chart for a generic triangle

and how we discretized the differential of a map and the metric tensor:

$$\begin{aligned} D\mathbf{x} &= (\mathbf{x}_u, \mathbf{x}_v) = (e_{21}, e_{31}) \\ g_j &= \begin{pmatrix} E_j & F_j \\ F_j & G_j \end{pmatrix} = \begin{pmatrix} \|e_{21}\|^2 & \langle e_{21}, e_{31} \rangle \\ \langle e_{21}, e_{31} \rangle & \|e_{31}\|^2 \end{pmatrix} \end{aligned}$$

and recall that we are assuming functions to be *linear* within each triangle.

13.9.1 The gradient

Since we assume the functions f to be linear within each triangle, the gradient ∇f is constant within each triangle. The gradient of a linear function is constant, e.g.

$$a \in \mathbb{R} \quad (ax)' = a$$

In order to compute the gradient $\nabla \tilde{f}$ we need the partial derivatives of \tilde{f} . Note that \tilde{f} is defined on the parametric domain, which is our usual reference triangle of side one. The gradient, the vector of partial derivatives, can be approximated by the finite differences at the vertices divided by the distance, which is one. So the partial derivatives are just the differences:

$$\begin{aligned}\frac{\partial \tilde{f}}{\partial u} &= f(x_{j,2}) - f(x_{j,1}) \\ \frac{\partial \tilde{f}}{\partial v} &= f(x_{j,3}) - f(x_{j,1})\end{aligned}$$

Then, the discrete gradient is given by:

$$\begin{aligned}\nabla f &= D\mathbf{x} g^{-1} \nabla \tilde{f} \\ &= (e_{21}, e_{31}) \begin{pmatrix} E_j & F_j \\ F_j & G_j \end{pmatrix}^{-1} \begin{pmatrix} f(x_{j,2}) - f(x_{j,1}) \\ f(x_{j,3}) - f(x_{j,1}) \end{pmatrix}\end{aligned}$$

also remember that the inverse of a 2×2 matrix is easy to compute:

$$\begin{pmatrix} E_j & F_j \\ F_j & G_j \end{pmatrix}^{-1} = \begin{pmatrix} G_j & -F_j \\ -F_j & E_j \end{pmatrix} \frac{1}{\det g_j} \quad (49)$$

13.9.2 Gradient norm

Given the gradient norm:

$$\|\nabla f\| = \sqrt{\nabla \tilde{f}^\top g^{-1} \nabla \tilde{f}}$$

We simply obtain:

$$\begin{aligned}\|\nabla f\| &= \sqrt{(f_u \ f_v) \begin{pmatrix} G_j & -F_j \\ -F_j & E_j \end{pmatrix} \begin{pmatrix} f_u \\ f_v \end{pmatrix}} \frac{1}{\sqrt{\det g_j}} \\ &= \sqrt{(f_u \ f_v) \begin{pmatrix} f_u G_j - f_v F_j \\ -f_u F_j + f_v E_j \end{pmatrix}} \frac{1}{\sqrt{\det g_j}} \\ &= \sqrt{\frac{f_u^2 G_j - 2 f_u f_v F_j + f_v^2 E_j}{\det g_j}}\end{aligned} \quad (50)$$

where in (50) we are computing g^{-1} using (49) and replacing the gradient in the parameter space with its vector representation.

13.9.3 Total variation

Definition 12 (total variation). The *total variation* of a function $f : S \rightarrow \mathbb{R}$ is:

$$TV_S(f) = \sum_j \int_{T_j} \|\nabla f(x)\| da$$

We know how to compute integrals of functions on triangulated meshes. We simply get, for each triangle:

$$\int_{T_j} \|\nabla f(x)\| \, da = \int_0^1 \int_0^{1-u} \sqrt{\frac{f_u^2 G_j - 2f_u f_v F_j + f_v^2 E_j}{\det g_j}} \sqrt{\det g_j} \, dv \, du \quad (51)$$

$$\begin{aligned} &= \int_0^1 \int_0^{1-u} \sqrt{f_u^2 G_j - 2f_u f_v F_j + f_v^2 E_j} \, dv \, du \\ &= \frac{1}{2} \sqrt{f_u^2 G_j - 2f_u f_v F_j + f_v^2 E_j} \end{aligned} \quad (52)$$

where in (51) we are applying the definition of integral of a function on surfaces (Section 12.8), and (52) is because:

$$\int_0^1 \left(\int_0^{1-x} dy \right) dx = \int_0^1 (1-x) \, dx = \frac{1}{2}$$

Remark 30. The gradient brings vertex based functions to triangle based vector fields. We have one gradient for each triangle.

Remark 31. A curiosity: We have an hairy ball and we want to comb it in a smooth way, without any singular point. This is the same as asking for a function with a smooth gradient, without singular points.

Thanks to the *hairy ball theorem* we know that every smooth function on a sphere-like surface will have at least one singular point. Only the donuts admit smooth vector fields without singular points.

14 The Laplace-Beltrami operator

In this section we are going to cover the key tool in geometry processing, the Laplace-Beltrami operator. There are a lot of courses and tutorial online to deepen the subject (suggested *The Swiss Army Knife of Geometry Processing*). We have seen a simplified version of this tool, the Graph Laplacian.

14.1 Motivation

This is the motivation in the sense that we see the Laplacian appearing in a physical phenomena, that is heat diffusion. It turns out that the Laplacian will show up in a lot of physical phenomena (e.g. how waves propagate in a lake, how quantum particle are observed in a domain).

Heat diffusion

The function u at point x is giving the current amount of heat at that point x , it is a scalar function. Moreover the temperature on the table does not only depend only the position on the plane but even on the time. For a subset $U \subset \mathbb{R}^2$ the diffusion of heat is described by the *heat equation*:

$$\begin{aligned}\frac{\partial}{\partial t} u(x, t) &= \Delta u(x, t) \\ u(x, 0) &= u_0(x)\end{aligned}$$

where the *Laplacian* is defined as:

$$\Delta u(x, t) = -\operatorname{div}(\nabla u) = \frac{\partial^2 u(x, t)}{\partial x_1^2} + \frac{\partial^2 u(x, t)}{\partial x_2^2}$$

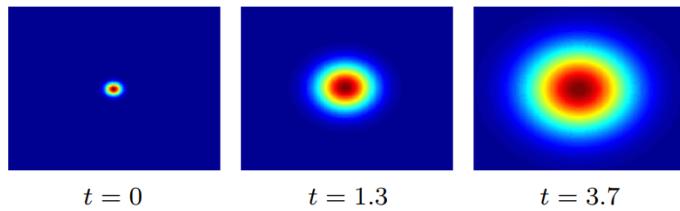


Figure 39: The diffusion of the heat, solution of heat equation, at different times

Heat diffusion on a surface

For a *surface* $\mathcal{X} \in \mathbb{R}^3$ the diffusion of heat is described by the *heat equation*:

$$\begin{aligned}\frac{\partial}{\partial t} u(x, t) &= \Delta u(x, t) \\ u(x, 0) &= u_0(x)\end{aligned}$$

where the *Laplacian* is defined in the same way, but on surfaces we are not able to talk about second order partial derivatives.

$$\Delta u(x, t) = -\operatorname{div}(\nabla u) = ?$$

14.2 Inner product on a manifold

We need to define how we can compute inner products on a manifold. Note that we can choose a different definition for the inner product, some people do, and everything would work anyway, it just needs to satisfy the properties of the inner product. However, this choice would change the properties of the quantities derived.

14.2.1 Scalar functions

Given two *scalar functions* $f, g : \mathcal{X} \rightarrow \mathbb{R}$, we define their inner product as:

$$\langle f, g \rangle_{\mathcal{X}} = \int_{\mathcal{X}} f(x)g(x) dx$$

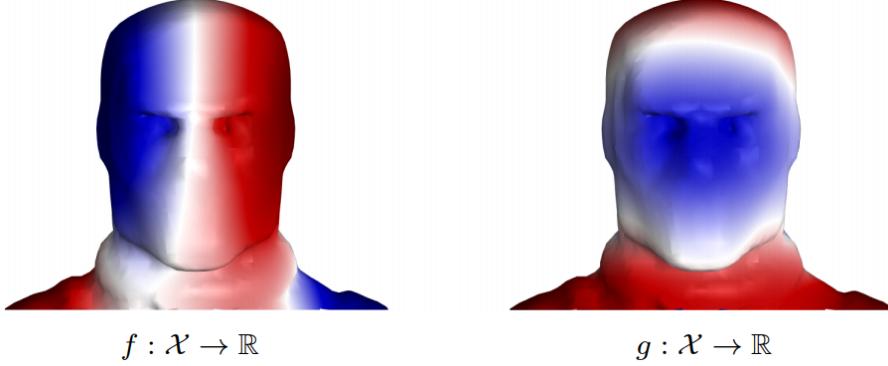


Figure 40: Two functions on the same manifold, if the functions are orthogonal the inner product will be zero

When discretized on a triangle mesh of n vertices, this boils down to:

$$\mathbf{f}^\top \text{diag}(a_i) \mathbf{g}$$

where a_i with $i = 1, \dots, n$ are the local *area elements* at each vertex.

This comes from how we discretized the integral of a function, i.e. using the area elements.

The i -th area element a_i , corresponding to vertex v_i , represents the area of the triangles adjacent to v_i , that is, *local* to the vertex itself. Clearly, only one third of each triangle is local to a vertex, since the other parts are local to the other vertices of the triangle. So, we can define the area element a_i for vertex v_i as

$$a_i = \frac{1}{3} \sum_{j: v_i \in T_j} A(T_j)$$

The key message is that if we want to compute inner products between scalar functions on a manifold, we need to take the areas into account.

14.2.2 Tangent vector fields

Given two *tangent vector fields* $F, G : \mathcal{X} \rightarrow T(\mathcal{X})$, we define their inner product as:

$$\langle F, G \rangle_{T\mathcal{X}} = \int_{\mathcal{X}} \langle F(x), G(x) \rangle_{T_x(\mathcal{X})} dx$$

Instead of a number per point, we have a vector per point.

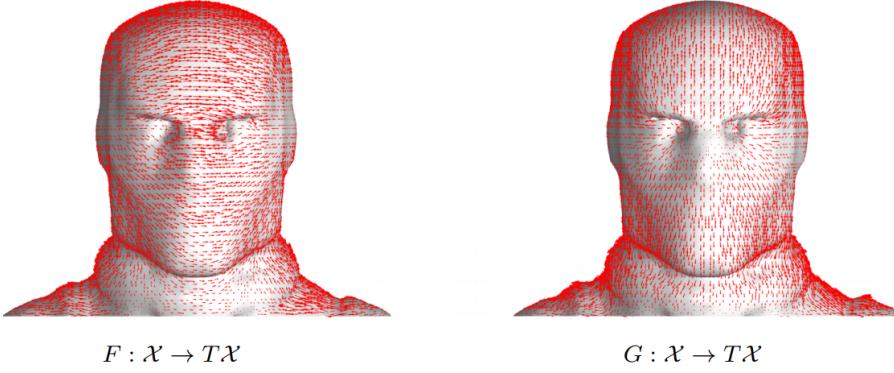


Figure 41: Two tangent vector fields on the same manifold

That is, given two vector fields F and G (Figure 41) computing the inner product means that at each point we take the two arrows, coming from the two vector fields, and we compute their inner products. Their inner products are computed using the first fundamental form.

Remark 32. In the inner product between scalar function we have integral of number times number, meanwhile, in the inner product between tangent vector fields we have integrals of a inner product between vectors.

Remark 33. Forgetting areas when computing inner products is the standard error that we commit when we do geometry processing, e.g. if we omit the areas the inner product between two orthogonal functions may not be zero.

14.3 Intuitions

These are some geometric Intuitions of the gradient, the divergence and the Laplacian.

Intuition 1 (Gradient). The gradient $\nabla f(x)$ is the *direction* of the steepest increase of f at x (Figure 43). Note that ∇f is a collection of arrows.

Intuition 2 (Divergence). The divergence $\text{div}(F(x))$ is the *scalar* density of an outward flux of F from an infinitesimal volume around x (Figure 44). It's a scalar function that gives a positive number if x behaves like a source, arrows flowing out of x , and, a negative number if x behaves like a sink. It could be zero (e.g. parallel vector field), some vector field can be divergence free

Intuition 3 (Laplacian). The Laplacian $\Delta f(x) = -\text{div}(\nabla f(x))$ is the *scalar* difference between $f(x)$ and the average of f on an infinitesimal sphere around x (Figure 45)

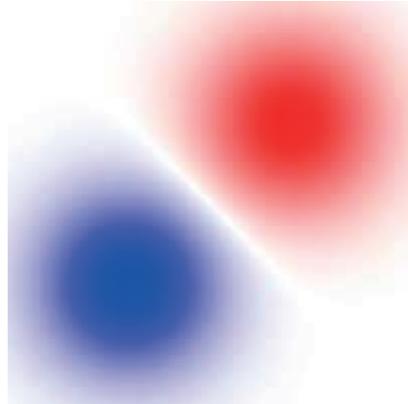


Figure 42: Smooth scalar field f

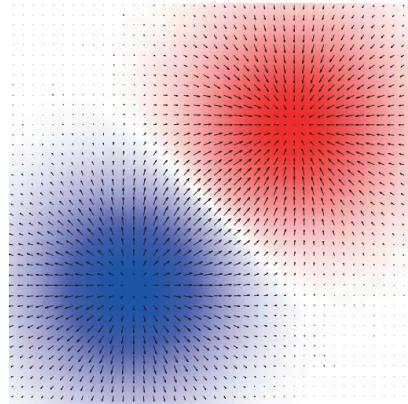


Figure 43: Smooth vector field F

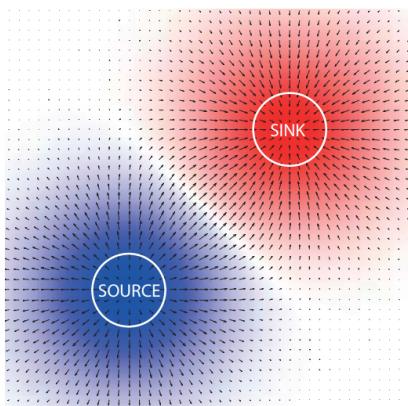


Figure 44: Source and sink in F

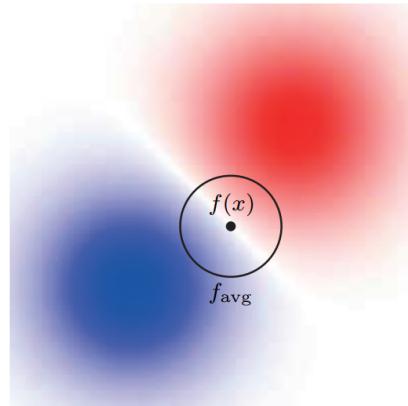


Figure 45: Average of f around x

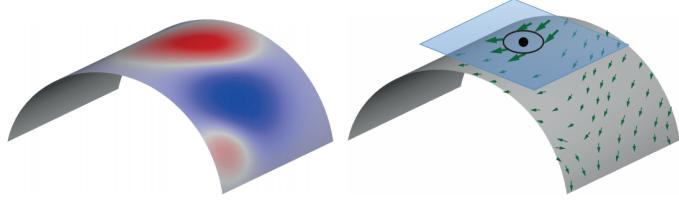


Figure 46: On surfaces we can still talk about sources and sinks, everything happens on tangent planes

14.4 Adjointness

Definition 13 (adjoint). Suppose $T \in \mathcal{L}(V, W)$, where $\mathcal{L}(V, W)$ is the set of all linear maps from V to W . The *adjoint* of T is the function $T^* : W \rightarrow V$ such that

$$\langle T v, w \rangle = \langle v, T^* w \rangle$$

for every $v \in V$ and every $w \in W$

The first definition for the Laplacian is:

Definition 14 (laplacian). The Laplacian is the *scalar* difference between $f(x)$ and the average of f on an infinitesimal sphere around x (Figure 45), and, it can be computed as the divergence of the gradient:

$$\Delta f(x) = -\operatorname{div}(\nabla f(x))$$

We have another definition for the Laplacian:

Definition 15 (laplacian). Gradient and divergence are (negative) *adjoint* operators

$$\langle F, \nabla f \rangle_{T(X)} = -\langle \operatorname{div} F, f \rangle_X$$

Remark 34. The first inner product is among *vector fields*, while the second is among *scalar functions*, but, they give the same number.

The reason why it's a definition is that we can actually define the divergence as being the adjoint of the gradient.

However, there is something missing in the definition. If our domain has a boundary, then there should be another term that takes care of what happens at the boundary. So, we are assuming that there is no boundary, we have close surfaces (e.g. sphere, dog with no holes).

Remembering the integration by parts, the formula for integrating product of functions, in the formula there was a definite integral on the boundary, this is the term that we are missing here. In a sense this formula is a generalization of integration by parts. We always used this formula but we didn't realize it because we worked with real valued functions in one dimension.

Remark 35. This formula is generalized by the *Green identity*, also known as divergence theorem. It's just integration by parts in many dimensions.

14.5 Finite elements method (FEM)

There are two options to discretize the Laplacian, either we try to discretize the divergence itself, and then we compose it with the gradient, or we just forget about divergence and gradient and we try to discretize the Laplacian directly.

We want to *discretize* the Laplace-Beltrami operator directly. It means that we want to compute the Laplacian on our triangle meshes, this is all we are interested in, we are jumping right into discretizing the operator.

We will follow a new idea, called the *finite elements method (FEM)*:

1. Consider the point-wise equation (or *strong equality*):

$$\Delta f(x) = g(x) \quad \forall x$$

We are interested in computing g , and then directly “solve for” Δ

2. To do so, we give up on having the equality satisfied at all points. We instead look at the *weak formulation*, i.e. we are just interested at looking at the inner product between some test function:

$$\langle \Delta f, h_j \rangle_{\mathcal{X}} = \langle g, h_j \rangle_{\mathcal{X}}$$

where $h_j : \mathcal{X} \rightarrow \mathbb{R}$ are some *test functions*

We will be happy if this equality holds, for any choice of h . It is a new idea, a new notion of equality, it is not point wise anymore. It’s like a weaker version of the equality, because the inner product is taking an integral between two functions. If something happens, if h has discontinuities, we don’t realize it, and, it does not change the values of the integral (we can take integral of functions not differentiable).

So we are allowing some discontinuities, some irregularities, and, since we are working with triangle meshes we are full of edges and singular points. All the vertices are singularities, so we don’t really want to be accurate at all points. We want to allow some irregularities

Thus, there is a conceptual difference between the traditional notion of equality “this must hold at all points”. Let’s say we have a sampled domain, and we want the equality to hold at all samples of the domain. If this holds for all samples then this holds for the entire domain. Now instead we ask for the equality to hold for some selection of functions, and then we’ll say then it holds for the entire space spanned by these functions. It is the same kind of reasoning, but instead of looking at points we look at functions

3. Then we obtain explicit expressions for the left and right sides, from which we explicitly compute a *matrix representation* for the Laplacian Δ

14.5.1 Hat basis

We are going to use the hat basis as *test functions*. It's not surprising because we expect the equality to hold for all test functions, and then we claim it holds for the entire space spanned by these functions. So it's natural that as our test functions we take a basis for the functions we are interested in, all functions on meshes.

We already have a basis for all the functions on meshes, the hat basis (Figure 47). We can approximate any scalar function by the linear combinations of the hat functions h_j , and this will give us a piece-wise linear approximation. Since for triangle meshes we are assuming the functions to be linear within each triangle, we are always working with piece-wise linear functions.

Thus we can approximate any function on the triangular mesh in this way:

$$f(x) \approx \sum_{i=1}^n f(v_i) h_i(x)$$

where h_i are *hat basis functions* defined at each vertex:

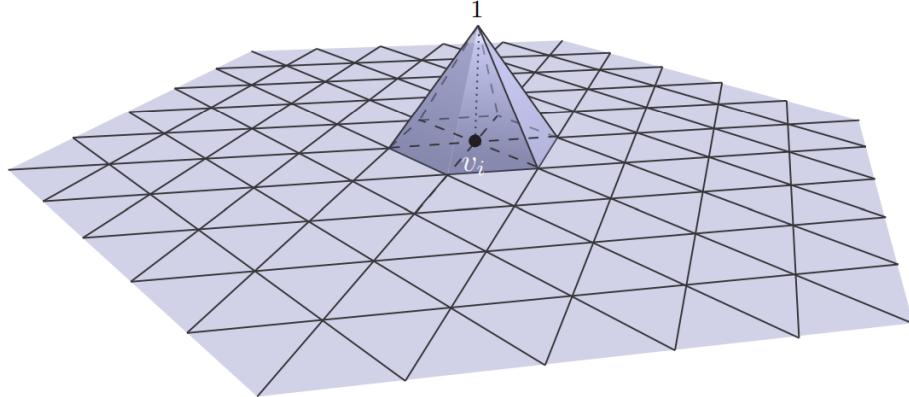


Figure 47: The vector h_j of the hat basis is 1 on v_j , 0 on neighbors vertices and decreases linearly from 1 to 0

14.5.2 Discretization

We use the hat basis as test functions in the equation:

$$\langle \Delta f, h_j \rangle_{\mathcal{X}} = \langle g, h_j \rangle_{\mathcal{X}}$$

Stiffness matrix

We can rewrite the left side as:

$$\langle \Delta f, h_j \rangle_{\mathcal{X}} = -\langle \operatorname{div} \nabla f, h_j \rangle_{\mathcal{X}} \quad (53)$$

$$= \langle \nabla f, \nabla h_j \rangle_{T(\mathcal{X})} \quad (54)$$

$$= \langle \nabla \sum_i f(v_i) h_i(x), \nabla h_j \rangle_{T(\mathcal{X})} \quad (55)$$

$$= \sum_i f(v_i) \underbrace{\langle \nabla h_i, \nabla h_j \rangle_{T(\mathcal{X})}}_{s_{ij}=s_{ji}} \quad (56)$$

where (53) is the definition of the Laplacian operator (Definition 14), (54) is because gradient and divergence are negative adjoint operators, in (55) we are expressing f as a linear combinations of hat functions, in (56) we are applying the linearity of the inner product since $f(v_i)$ is a number, so we can bring it out. *Remark 36.* We choose to define the Laplacian with the minus so that we get rid of the minus of the *negative* adjointness.

We define the matrix $\mathbf{S} = (s_{ij}) \in \mathbb{R}^{n \times n}$ as the symmetric *stiffness matrix*. The stiffness matrix allows us to have:

$$\langle \Delta f, h_j \rangle_{\mathcal{X}} = (\mathbf{S}\mathbf{f})_j$$

Note that we still don't know how to compute the inner products s_{ij} .

Curiosity. The name stiffness matrix comes from the first application of finite elements in history, it was in structural mechanics in the rigidity of buildings.

Mass matrix

We can rewrite the right side as:

$$\langle g, h_j \rangle_{\mathcal{X}} = \langle \sum_i g(v_i) h_i(x), h_j \rangle_{\mathcal{X}} \quad (57)$$

$$= \sum_i g(v_i) \underbrace{\langle h_i, h_j \rangle_{\mathcal{X}}}_{a_{ij}=a_{ji}} \quad (58)$$

where in (57) we are discretizing g using the hat basis, in (58) we are applying the linearity of the inner product, bringing out the sum of numbers $g(v_i)$.

We define $\mathbf{A} = (a_{ij}) \in \mathbb{R}^{n \times n}$ as the symmetric *mass matrix*, so we have:

$$\langle g, h_j \rangle_{\mathcal{X}} = (\mathbf{A}\mathbf{g})_j$$

Remark 37. The inner products on a manifold are defined in Section 14.2

14.5.3 Discrete Laplace-Beltrami operator

So, starting from:

$$\langle \Delta f, h_j \rangle_{\mathcal{X}} = \langle g, h_j \rangle_{\mathcal{X}}$$

We get to:

$$\begin{aligned} (\mathbf{S}\mathbf{f})_j &= (\mathbf{A}\mathbf{g})_j \quad \text{for } j = 1, \dots, n \\ \mathbf{S}\mathbf{f} &= \mathbf{A}\mathbf{g} \\ \mathbf{A}^{-1}\mathbf{S}\mathbf{f} &= \mathbf{g} \end{aligned}$$

it can be proven that the way we are computing \mathbf{A} yields an invertible matrix.

Our *discrete Laplacian* is thus given by:

$$\Delta f = g \quad \text{and} \quad \mathbf{A}^{-1}\mathbf{S}\mathbf{f} = \mathbf{g} \quad \implies \quad \Delta = \mathbf{A}^{-1}\mathbf{S}$$

Remark 38. We will indicate the Laplacian with Δ or with \mathbf{L} .

What remains to see is how to compute the elements that appear in the stiffness matrix \mathbf{S} and in the mass matrix \mathbf{A} , that are inner products.

Properties

Some interesting *properties* of this discretization:

- The matrices involved are *sparse* and *symmetric*. The fact that are symmetric was clear since we are looking at inner products
- They have the same structure (zero pattern) as the vertex-to-vertex *adjacency* matrix. There is some analogy with the graph Laplacian. Moreover, everything that we said for the adjacency matrix works even here for the Laplacian, e.g. we can take k -powers of the Laplacian, changing the geometric intuition to “difference between f and the average in a k -neighborhood”
- The stiffness \mathbf{S} is *positive semi-definite* (Definition 16)
 - They can be computed easily and *efficiently* for any given mesh

Definition 16 (positive semi-definite). A symmetric $n \times n$ real matrix \mathbf{M} is said to be *positive semi-definite* if the scalar $z^T \mathbf{M} z$ is positive or zero for every non-zero column vector z of n real numbers.

14.5.4 Mass integral

Now we will try to compute the mass matrix.

Off-diagonal of \mathbf{A}

Assume $i \neq j$:

$$a_{ij} = \langle \mathbf{h}_i, \mathbf{h}_j \rangle_{\mathcal{X}} \quad (59)$$

$$= \sum_l \int_{T_l} \mathbf{h}_i(\mathbf{x}) \mathbf{h}_j(\mathbf{x}) \, d\mathbf{x} \quad (60)$$

$$= \int_{T_{jii'}} \mathbf{h}_i(\mathbf{x}) \mathbf{h}_j(\mathbf{x}) \, d\mathbf{x} + \int_{T_{ji''i}} \mathbf{h}_i(\mathbf{x}) \mathbf{h}_j(\mathbf{x}) \, d\mathbf{x} \quad (61)$$

where

- (59) is by definition of the mass matrix
- First of all, we defined the integral within each triangle, so the integral of a functions on a triangle mesh is the sum of the integrals on each triangle. Same thing for the inner product in (60), the inner product is just an integral of product of functions (Section 14.2) so it is the sum of integrals of each triangle. We are covering \mathcal{X} using all triangles $T_l \in \mathcal{X}$
- in (61) we are exploiting the observation that we don't really need to compute the integral on all triangles of the mesh. The hat functions \mathbf{h}_i and \mathbf{h}_j have only very local support, so their product will be zero in most triangles, thus most of the integrals in the sum will be zero.

Let's try to understand where are the triangles of interest, consider the Figure 48 where we have the hat functions \mathbf{h}_i and \mathbf{h}_j . Notice that we must consider every pair of hat functions that overlap in at least one point, otherwise their product would be zero. If they touch each other only in one vertex (they're at distance 2) their product is still zero. The only hat functions that overlap are the immediate neighbors, i.e. the corresponding vertices are connected.

Given two neighbors hat functions \mathbf{h}_i and \mathbf{h}_j there are only two triangles where these hat functions overlap (Figure 48), where their product isn't zero.

Thus, we can simplify the sum over all the triangles to only those two triangles.

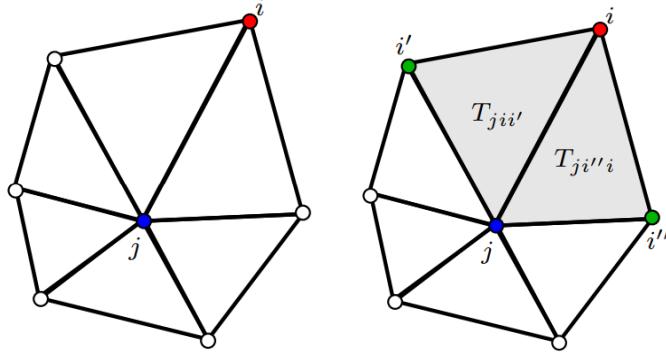


Figure 48: Given v_i and v_j connected, h_i and h_j overlap in only two triangles

We are going to compute one of these integral, since they are equal, on the generic triangle T_l .

So, remembering how we discretized the integral of a function in (24), the integral can be written in *parameter space* as:

$$\int_{T_j} \mathbf{h}_i(\mathbf{x}) \mathbf{h}_j(\mathbf{x}) \, dx \implies 2A(T_l) \int_0^1 \int_0^{1-u} \mathbf{v} \mathbf{u} \, dv \, du = 2A(T_l) \frac{1}{24} = \frac{1}{12} A(T_l)$$

where $2A(T_l)$ comes from $\sqrt{\det g_j}$, inside the definition of area element dx (previously da), and the intervals of the integrals are just covering the canonical triangle. In order to bring in parameter space this integral we need to choose a chart, a chart that will yield a nice expression.

We'll use the following chart in Figure 49. This chart $\mathbf{x}_{jii'}$ maps:

$$(0, 0) \rightarrow i' \quad (0, 1) \rightarrow i \quad (1, 0) \rightarrow j$$

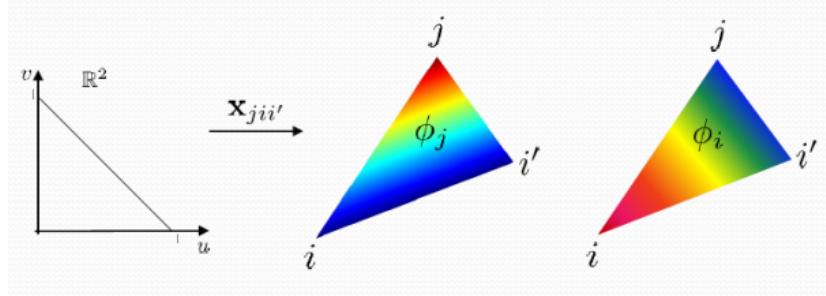


Figure 49: The chart from the canonical triangle to a generic one

Each function in the reference triangle is given by:

$$f(0,0)(1-u-v) + f(0,1)u + f(1,0)v$$

Thus, it's easy to see how $u \in [0, 1]$ and $v \in [0, 1 - u]$ yield a linear interpolation of the function. Remembering that the hat functions decrease linearly from 1 to 0, it's straightforward that those hat functions have the same exact behaviour of u and v (and, of course, they have the same effect on functions).

Remark 39. We can't discretize this integral as a simple inner product as we already did, since we don't know how to represent the hat vectors in a discrete manner. In which basis should we represent them? We'd have to choose a base, and, define the possible inner products between the vectors of that base. This is exactly what we are doing for the hat basis (not so sure).

Diagonal of A

Now assume $i = j$:

$$a_{jj} = \langle h_j, h_j \rangle \quad (62)$$

$$= \sum_l \int_{T_l} h_j(x) h_j(x) dx \quad (63)$$

$$= \sum_{i \in \mathcal{N}(j)} \int_{T_{jii'}} h_j(x) h_j(x) dx \quad (64)$$

where (62) and (63) have the same motivation as before, to simplify (64) we must understand which are the triangles involved in the integral. The integral is non zero exactly where h_j is supported, exactly where h_j is non zero. Thus, it is non zero at the neighboring triangles, so we have just a summation over neighboring triangles of v_j (Figure 50).

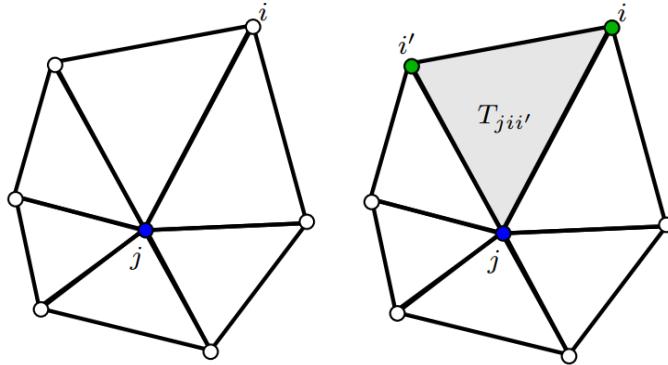


Figure 50: The triangles neighbors of v_j are the only ones where h_j is not zero

Like before, each integral in the summation in parameter space looks like:

$$\int_{T_j} h_j(x) h_j(x) dx \implies 2A(T_l) \int_0^1 \int_0^{1-u} u^2 dv du = \frac{1}{6} A(T_l)$$

14.5.5 Stiffness integral

The stiffness matrix is more complicated to compute. In the stiffness integral we need to compute inner products between gradient of hats.

Like before, we don't need to consider all the triangles. We need to consider triangles where the hats overlap, the same triangles as before (Figure 51). Thus, the integrals are non-zero in the same positions as the mass matrix.

We'll not show the derivation, just trust the formula in the next paragraph. e.g. in the off-diagonal of \mathbf{S} we have:

$$\begin{aligned} s_{ij} &= \langle \nabla h_i, \nabla h_j \rangle_{T(\mathcal{X})} \\ &= \sum_l \int_{T_l} \langle \nabla h_i(x), \nabla h_j(x) \rangle dx \\ &= \dots \\ &= -\frac{1}{2} (\cot \alpha_{ij} + \cot \beta_{ij}) \end{aligned}$$

Notice that the formula involve internal *angles* (Figure 51), and in particular the cotangent of these angles (the cotangent is just the cosine over the sine).

For this reason when we compute the finite element discretization of the Laplacian, this is also called the “cotangent Laplacian”, because cotangents show up when we discretize the stiffness matrix.

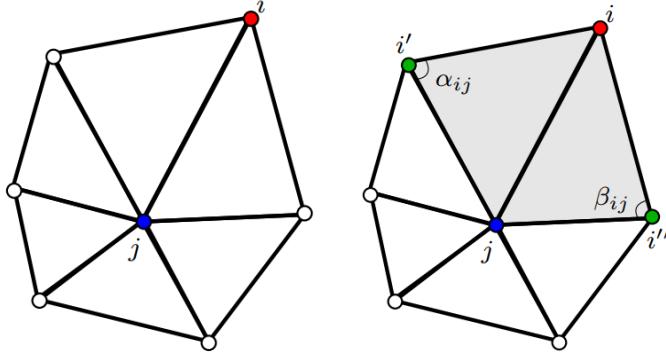


Figure 51: The triangles involved are the same as in the mass matrix

14.5.6 Stiffness and mass matrices

The discrete (FEM) Laplace-Beltrami operator is the $n \times n$ matrix:

$$\mathbf{L} = \mathbf{A}^{-1} \mathbf{S}$$

where

$$s_{ij} = \begin{cases} -\frac{1}{2}(\cot \alpha_{ij} + \cot \beta_{ij}) & \text{if } e_{ij} \in E \\ -\sum_{k \neq i} s_{ik} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

$$a_{ij} = \begin{cases} \frac{1}{12}(A(T_{jii'}) + A(T_{ji'i})) & \text{if } e_{ij} \in E \\ \frac{1}{6} \sum_{k \in \mathcal{N}(i)} A(T_k) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

This is the correct discretization for the Laplacian. There is a much more popular discretization, which exactly the same expect that the mass matrix is much simpler.

A *lumped mass* matrix is obtained summing up each row of the mass matrix (taking the sum is called lumping) and put them on the diagonal of a new matrix:

$$\tilde{a}_{ij} = \begin{cases} \sum_{k=1}^n a_{ik} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

$$= \begin{cases} \frac{1}{3} \sum_{k \in \mathcal{N}(i)} A(T_k) & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

It's interesting that the sum of each row in the mass matrix gives $\frac{1}{3} \sum_{k \in \mathcal{N}(i)} A(T_k)$, i.e. the area elements at each vertex, that we already discussed.

Note that the inverse of a sparse matrix is not necessarily sparse. This is in general a dense matrix, and we don't want to compute the dense inverse of a $n \times n$ matrix. However, the inverse of a diagonal matrix is a diagonal matrix with one over the numbers along the diagonal. It's very convenient, this is why the lumped mass matrix it's used.

Remark 40. In the stiffness matrix the sum along each row is zero, because on the diagonal we have the sum of the remaining elements in that row with a negative sign, thus, the overall sum is zero. This was true even in the Graph Laplacian.

Remark 41. Looking at the formulas is evident that:

- They are symmetric
- The structure of the two matrices is the same as the adjacency matrix of mesh, but there is something more, since now we have a diagonal that wasn't present on the adjacency matrix

Remark 42. Why should we use the FEM Laplacian and not just the Graph Laplacian? The answer is areas. I can take one mesh, take huge triangles, and compute the graph Laplacian. If I just shrink the triangles keeping the same connectivity, the graph Laplacian won't change. This is not good, since we are dealing with surfaces, so we need to take care of lengths, areas and so on. The

graph Laplacian is not the right tool to use on surfaces, it ignores areas, it is scale invariant.

15 Spectral geometry

In this section we'll finally talk about spectral geometry. This is a very generic and fascinating topic.

Spectral geometry is all about studying spectral properties of the Laplacian or other operators, spectral meaning eigenvalues or eigenvectors.

We'll start with studying some general properties of the Laplace-Beltrami operator, and then we'll see what they mean from a spectral prospective.

15.1 Self-adjointness of Δ

The Laplace-Beltrami operator Δ is *self-adjoint*.

$$\langle F, \nabla g \rangle_{T(\mathcal{X})} = -\langle \operatorname{div} F, g \rangle_{\mathcal{X}} \quad (65)$$

$$\langle \nabla f, \nabla g \rangle_{T(\mathcal{X})} = -\langle \operatorname{div} \nabla f, g \rangle_{\mathcal{X}} \quad (66)$$

$$\langle \nabla f, \nabla g \rangle_{T(\mathcal{X})} = \langle \Delta f, g \rangle_{\mathcal{X}} \quad (67)$$

$$\langle \nabla f, \nabla g \rangle_{T(\mathcal{X})} = \langle \nabla g, \nabla f \rangle_{T(\mathcal{X})} \quad (68)$$

$$\langle \nabla g, \nabla f \rangle_{T(\mathcal{X})} = -\langle \operatorname{div} \nabla g, f \rangle_{\mathcal{X}} \quad (69)$$

$$\langle \nabla g, \nabla f \rangle_{T(\mathcal{X})} = \langle \Delta g, f \rangle_{\mathcal{X}} \quad (70)$$

$$\langle \nabla g, \nabla f \rangle_{T(\mathcal{X})} = \langle f, \Delta g \rangle_{\mathcal{X}} \quad (71)$$

So we get the self adjointness of Δ :

$$\langle \Delta f, g \rangle_{\mathcal{X}} = \langle f, \Delta g \rangle_{\mathcal{X}}$$

where (65) is because the gradient and the divergence are adjoint operators, in (66) we pick a specific choice for the tangent vector field F , let's say F is the gradient of some scalar function $F = \nabla f$, in (67) we are using the Definition 14 of the Laplacian $\Delta f = -\operatorname{div} \nabla f$, in (68) we are exploiting the symmetry of the inner product, in (69) and (70) we are using again the adjointness of gradient and divergence and the definition of laplacian, in (71) we use symmetry again.

Basically we can move the Laplacian on the left argument or the right argument of a inner product. This property is known as the self adjointness of the operators.

So we found out that the Laplace-Beltrami operator is self-adjoint, it is interesting because self-adjoint operators have a lot of interesting properties that we are going to exploit.

15.2 Laplacian eigenvalues

Consider the (in general, *complex*) eigenvalue equation:

$$\Delta \phi = \lambda \phi$$

where λ is a scalar, the eigenvalue, and ϕ is a function, the eigenfunction. The reason why we can do this is because Δ is a linear operator, meaning that it admits a matrix representation.

However we don't know anything about the eigenvalues being reals or complex, we can't claim anything yet, it could be that the eigenvalues have an imaginary part.

For the eigenfunction/eigenvalue pair (ϕ, λ) , we can write this equation:

$$\lambda \langle \phi, \phi \rangle_{\mathcal{X}} = \langle \lambda \phi, \phi \rangle_{\mathcal{X}} \quad (72)$$

$$= \langle \Delta \phi, \phi \rangle_{\mathcal{X}} \quad (73)$$

$$= \langle \phi, \Delta \phi \rangle_{\mathcal{X}} \quad (74)$$

$$= \overline{\langle \Delta \phi, \phi \rangle_{\mathcal{X}}} \quad (75)$$

$$= \overline{\langle \lambda \phi, \phi \rangle_{\mathcal{X}}} \quad (76)$$

$$= \overline{\lambda} \langle \phi, \phi \rangle_{\mathcal{X}} \quad (77)$$

where (72) is due to the linearity of the inner product, (73) is because we are considering a eigenfunction/eigenvalue pair, in (74) we are exploiting the self adjointness of the Laplace-Beltrami operator, in (75) we are applying the conjugate symmetry of the inner product, (76) is because we are considering a eigenfunction/eigenvalue pair, (77) is due to the linearity of the inner product and because for a *holomorphic* function φ it holds $\varphi(\bar{z}) = \overline{\varphi(z)}$, and, the inner product is holomorphic.

Hence, we have $\overline{\lambda} = \lambda$ so the eigenvalues of Δ are *real*. Actually, every operator that is self-adjoint has real eigenvalues.

The fact that we have real eigenvalues is really nice, because as we will see the eigenvalues of the Laplacian are a very interesting tool.

Remark 43. Inner products in general have the property of conjugate symmetry. If dealing with real numbers, the symmetry and the conjugate symmetry are equal.

15.3 Laplacian eigenfunctions

Consider again the eigenvalue equation

$$\Delta \phi = \lambda \phi$$

and two distinct eigenfunctions ϕ_i, ϕ_j with eigenvalues $\lambda_i \neq \lambda_j$. We have:

$$\langle \phi_i, \Delta \phi_j \rangle_{\mathcal{X}} = \langle \Delta \phi_i, \phi_j \rangle_{\mathcal{X}} \quad (78)$$

$$\lambda_j \langle \phi_i, \phi_j \rangle_{\mathcal{X}} = \lambda_i \langle \phi_i, \phi_j \rangle_{\mathcal{X}} \quad (79)$$

where (78) is because Δ is a self adjoint operator, (79) is due to the linearity on the first slot and the symmetry of the inner product on real numbers.

This can only be true if

$$\langle \phi_i, \phi_j \rangle_{\mathcal{X}} = 0$$

Therefore, the eigenfunction ϕ of Δ are *orthogonal*.

Eigenfunctions span an entire vector space of dimension one, which means that the scale of the eigenfunction doesn't matter, because we can always multiply by a scalar and we obtain an element in that space. So, whenever we are given eigenfunctions we can rescale them to have norm one, so we are not ruining the eigenfunctions, and then we obtain an *orthonormal* set of functions. Thus, eigenfunctions can be rescaled to be orthonormal:

$$\langle \phi_i, \phi_j \rangle_{\mathcal{X}} = \delta_{ij} \quad \text{where } \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

Note that in general is true for every self-adjoint operator, because the only property that we use is that Δ is self-adjoints

Remark 44. $\langle \phi_i, \phi_i \rangle_{\mathcal{X}} = 1 \implies \|\phi_i\|^2 = \|\phi_i\| = 1$ thanks to Theorem 3.

Remark 45. Any self-adjoint operator has real eigenvalues and orthogonal eigenvectors.

Curiosity. Any solver for eigenvectors and eigenvalues, if the input is a symmetric matrix, will give orthonormal vectors. It is an arbitrary choice, but it is a convention to take them to have norm one.

15.4 Spectral theorem

It is known as spectral theorem on Hilbert spaces, where Hilbert space is a vector space of functions with a norm.

Remember that the Laplacian $\Delta : \mathcal{F}(\mathcal{X}) \rightarrow \mathcal{F}(\mathcal{X})$ operates on a *vector space* of functions $\mathcal{F}(\mathcal{X})$, i.e. it takes a function on the shape and gives a function on the shape, and, we know that these functional spaces are vector spaces.

Theorem 5. *The eigenfunctions $\{\phi_i\}$ of Δ form an orthonormal basis of $\mathcal{F}(\mathcal{X})$.*

We have just proven that the eigenfunctions an orthonormal set. We can imagine that they form a basis, of course they span something, the interesting thing is that they span exactly the space of functions on the shape $\mathcal{F}(\mathcal{X})$.

This is not exactly correct, it is all possible squared integrable functions, but for our purposes we can think to all possible functions.

We have seen some possible choices for basis on the function space on shapes (e.g. hat basis, voronoi basis). Now we have another choice, the eigenfunctions of the Laplacian it's a new basis.

It means that if we are given some function on the shape $f \in \mathcal{F}(\mathcal{X})$, it can be written as a linear combination of the Laplacian eigenfunctions $\{\phi_i\}$ (Figure 52).

So we have these coefficients of the linear combination, they can be computed simply by $c_j = \langle f, \phi_j \rangle$ since the eigenfunctions are orthonormal (second recap on algebra). We are just projecting f into the basis function and it gives its coefficient c_j .

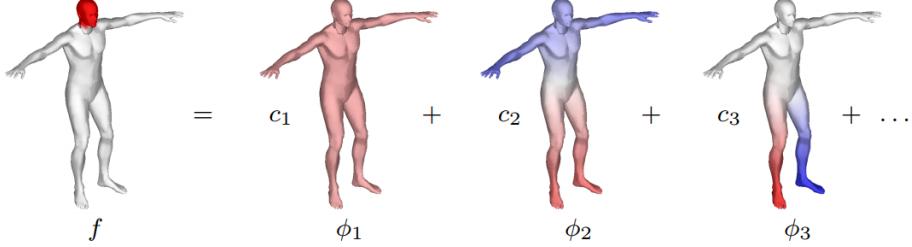


Figure 52: The function f as linear combination of vectors in the basis ϕ_i

15.5 Discretization

In the *discrete* setting we have:

$$\langle f, g \rangle_{\mathcal{X}} \implies \mathbf{f}^T \mathbf{A} \mathbf{g} \quad (80)$$

$$\langle f, \Delta g \rangle_{\mathcal{X}} \implies \mathbf{f}^T \mathbf{S} \mathbf{g} \quad (81)$$

$$\langle \phi_i, \phi_j \rangle_{\mathcal{X}} = \delta_{ij} \implies \Phi^T \mathbf{A} \Phi = \mathbf{I} \quad (82)$$

$$c_i = \langle \phi_i, f \rangle_{\mathcal{X}} \implies \mathbf{c} = \Phi^T \mathbf{A} \mathbf{f}$$

where:

- in (80) we use something that we saw in Section 14.2, the inner product between functions $\langle f, g \rangle_{\mathcal{X}}$ on a manifold is discretized as $\mathbf{f}^T \mathbf{A} \mathbf{g}$, where A is the diagonal matrix with the area element on the diagonal, i.e. the mass matrix. If it was on a Euclidean space, it would have been $\langle f, g \rangle \implies \mathbf{f}^T \mathbf{g}$
- in (81) we have an inner product, so it must be $\mathbf{f}^T \mathbf{A}$ times the discrete version of Δg . We saw that we can discretize the Laplace-Beltrami operator as $\Delta = \mathbf{A}^{-1} \mathbf{S}$ (Section 14.5.3), so we get that:

$$\langle f, \Delta g \rangle_{\mathcal{X}} \implies \mathbf{f}^T \mathbf{A} (\mathbf{A}^{-1} \mathbf{S} \mathbf{g}) = \mathbf{f}^T \mathbf{S} \mathbf{g}$$

- in (82) we are organizing all the $\{\phi\}$ in a matrix, where each column is a ϕ . Remember that the inner products give one only if $i = j$ since the eigenfunctions are orthogonal.
- The i -th coefficient in the Laplacian eigenbasis is given by the inner product $c_i = \langle \phi_i, f \rangle_{\mathcal{X}}$ since $\{\phi\}$ are orthonormal (second recap on algebra). In order to compute all coefficient in the laplacian eigenbasis, we arrange all the eigenfunctions in the columns of a matrix Φ and discretize the inner product as usual.

Remark 46. We must always use \mathbf{A} , otherwise we are not treating the surface as a surface, but we would operate in the standard basis in \mathbb{R}^3 .

We consider the eigenvalue problem, we are looking for the eigenfunction and eigenvalue matrices:

$$\mathbf{A}^{-1}\mathbf{S}\Phi = \Phi\Lambda$$

where $\mathbf{A}^{-1}\mathbf{S} = \Delta$, and Λ is diagonal with the eigenvalues on the diagonal. This is equivalent to do k equations, one per eigenfunction.

However, this formulation is not numerical stable, so we will solve the, equivalent, *generalized eigenvalue problem*:

$$\mathbf{S}\Phi = \mathbf{A}\Phi\Lambda$$

Remark 47. \mathbf{S} and \mathbf{A} are known.

15.6 Integral of eigenfunctions

It is easy to see that the following holds for any $f : \mathcal{X} \rightarrow \mathbb{R}$:

$$\int_{\mathcal{X}} \Delta f = \int_{\mathcal{X}} f_1 \Delta f \tag{83}$$

$$= \langle f_1, \Delta f \rangle_{\mathcal{X}} \tag{84}$$

$$= \langle \nabla f_1, \nabla f \rangle_{T(\mathcal{X})} \tag{85}$$

$$= \int_{\mathcal{X}} \langle \nabla f_1, \nabla f \rangle_{T_x(\mathcal{X})} \tag{86}$$

$$= 0 \tag{87}$$

where in (83) we are multiplying the Laplacian by a function that gives always 1, in (84) we are considering the integral as a inner product (Section 14.2), in (85) we are exploiting the equivalence derived in (71): $\langle f_1, \Delta f \rangle_{\mathcal{X}} = \langle \nabla f_1, \nabla f \rangle_{T(\mathcal{X})}$ in (86) we considering the inner product as an integral (Section 14.2), (87) is because the gradient of a constant function is zero, the inner product is zero since it's (conjugate) linear and finally the integral of zero is zero.

In particular, let $\Delta\phi = \lambda\phi$. Then, we can write:

$$\int_{\mathcal{X}} \phi = \frac{1}{\lambda} \int_{\mathcal{X}} \Delta\phi \tag{88}$$

$$= 0 \tag{89}$$

where (88) is because we are considering a eigenfunction/eigenvalue pair, (89) is due to (87).

In other words, the integral of the Laplacian of any function is zero. All Laplacian eigenfunctions integrate to zero, which means that Laplacian eigenfunctions have *zero average*.

15.7 Dirichlet energy

The Dirichlet energy of a function $f : \mathcal{X} \rightarrow \mathbb{R}$ measures its *smoothness*:

$$\varepsilon(f) = \int_{\mathcal{X}} \|\nabla f(x)\|^2 dx$$

Remark 48. We are summing up the squared length of each vector in the vector field. Each vector will have a large magnitude if the function changes abruptly and it will be small if the function is smooth.

We'll consider the Dirichlet energy for some Laplacian eigenfunction ϕ with eigenvalue λ :

$$\int_{\mathcal{X}} \|\nabla \phi\|^2 = \int_{\mathcal{X}} \langle \nabla \phi, \nabla \phi \rangle_{T_x(\mathcal{X})} \quad (90)$$

$$= \langle \nabla \phi, \nabla \phi \rangle_{T(\mathcal{X})} \quad (91)$$

$$= \langle \phi, \Delta \phi \rangle_{\mathcal{X}} \quad (92)$$

$$= \int_{\mathcal{X}} \phi \Delta \phi \quad (93)$$

$$= \lambda \int_{\mathcal{X}} \phi \phi = \lambda \quad (94)$$

where (90) is due to Theorem 3, (91) is due to the definition of inner products on manifolds, (92) holds again thanks to (71), (93) is again due to the definition of inner products on manifolds, in (94) we are using the definition of eigenfunction $\Delta \phi = \lambda \phi$ and the fact that the $\{\phi\}$ are orthonormal and thus have norm one, in the manifold sense.

So, *eigenvalues are the Dirichlet energies of eigenfunctions*.

This is a nice characterization, you never give a geometric meaning the an eigenvalue. An eigenvalue is just a number, but here, it seems to measure how smooth the eigenfunction is. It is the Dirichlet energy of the eigenfunction.

Note that exists other operators, different from the Laplacian, that don't have a precise notion of eigenvalue, because maybe they are not even linear operators. But, we can always come up with some notion of generalized eigenvalue, because we can take the operator, apply it to a function, and we measure something like the Dirichlet energy. So we can come up with an analogy of the Dirichlet energy for other operators as well, and whatever number we get out of it, we can claim that it's going to be my generalized notion eigenvalue for that value. People in mathematics do this.

There is always a relationship of this form, the eigenvalue is some integral of something, some energy, applied to the eigenfunction.

Further, we can see that $\lambda = 0$ only when $\nabla \phi = 0$, that is, when ϕ is *constant*. So, if there is a constant eigenfunction then its eigenvalue is zero. Now, we

always have a constant eigenfunction, because the Laplacian of any constant function f is zero (the laplacian is the divergence of the gradient of f , and the gradient of a constant function is 0), and thus, we can write:

$$\Delta f = 0 = 0f \quad \forall f \text{ constant}$$

which means that any constant function f is an eigenfunction of the Laplacian and their eigenvalue is always 0.

15.8 Discrete spectrum

We'll call the sequence of eigenvalues *spectrum*. The eigenvalues of the Laplacian are *countable* and are canonically ordered non-decreasingly:

$$0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \dots \rightarrow \infty$$

It turns out that it is doable, because the eigenvalues of the Laplacian form a discrete sequence, they are not dense.

There is an equal sign at the beginning, it is saying that there is only one zero eigenvalue. The moltiplicity of eigenvalue 0 identifies the number of connected components of the object (not proved), usually we have one connected component so we have only one eigenvalue equal to zero.

Remark 49. In the continuous world we have infinite eigenvalues, countably many. In practice we deal with discrete shapes with n vertices, and so we have n by n matrices thus at most n eigenvalues.

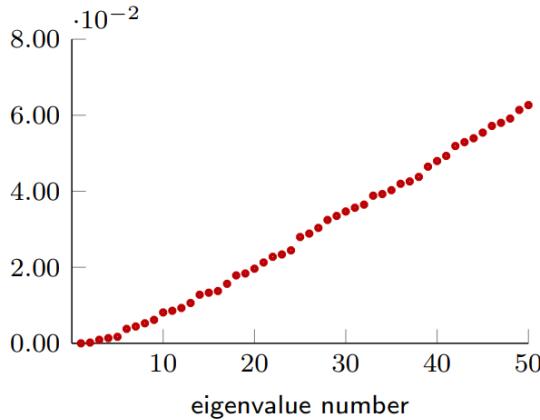


Figure 53: The eigenvalues canonical ordered increase linearly

The Laplacian eigenvalues follow *Weyl's asymptotic law*:

$$\lambda_i \approx \frac{\pi}{\int_X da} i \quad \text{for } i \rightarrow \infty$$

That is, they grow *linearly* (Figure 53) with growth rate (i.e. the slope of the line) inversely proportional to *surface area* of the manifold (e.g. smaller manifolds have a steeper spectrum).

Note that this law holds for any manifold on any dimensions (not only 2-dimensional), what changes is the power that you give to i .

15.9 Laplacian eigenvalues: Properties

The Laplacian spectrum is particularly useful in shape analysis:

- The spectrum is *isometry invariant*. The Laplacian is isometry invariant (i.e. if we isometrically deform the shape its Laplace stays the same). Which means the eigenspaces of the Laplacian stays the same, so also the eigenvalues stay the same. Thus eigenvalues are invariant to isometries: we can use the spectrum as a *signature* to represent the shapes in a isometric invariant way (e.g. useful for shape retrieval, shape clustering).
- The spectrum depends *continuously* on the Riemannian metric. The Riemannian metric, the tensor g , determines the Laplacian and changing the tensor a little bit (i.e. deforming the shape non isometrically a little bit) changes the Laplacian eigenvalues just a little bit.
- The spectrum is easy and *efficient* to compute
- The spectrum is *informative* as it encodes metric and topological information of the surface (e.g. surface area looking at the slope of the spectrum, presence of symmetries, topology, etc.). Can I reconstruct the shape given the spectrum? This problem is also called “hearing the shape of the drum”, since if the shape is identifying a drum, the frequency of the sound produced are the eigenvalues of the laplacian. One “cannot hear the shape of the drum” (Figure 54). However, these examples are very rare and exotic, can one *in practice* hear the shape of the drum? Yes, we just did it :)

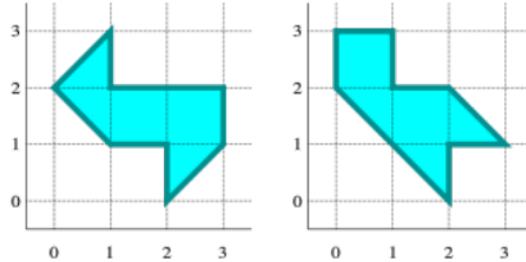


Figure 54: Two non-isometric shapes with the same spectrum

15.10 Laplacian eigenfunctions

In Fourier analysis we have some signal in one dimension, a real valued signal e.g. the sound. We can decompose this signal into a linear combination of sines and cosines, this is what Fourier is saying.

It turns out that the Fourier basis function, these sines and cosines, are the eigenfunctions of the 1D Euclidean Laplacian (Figure 55).

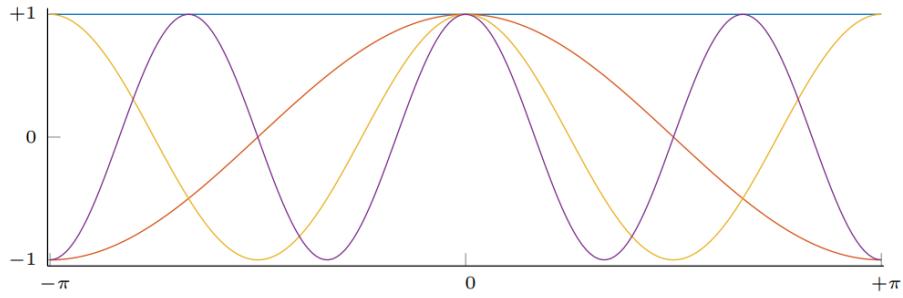


Figure 55: First four eigenfunctions of the 1D Laplacian, they are sine and cosine.

Basically, now we have the analogy for Fourier analysis for functions defined on a manifold instead of on the real line (Figure 56). They are the eigenfunction of the manifold Laplacian, the analogy of the Fourier basis, but for surfaces. This kind of explains, or at least gives some intuition, of this oscillatory behaviour in the eigenfunctions.

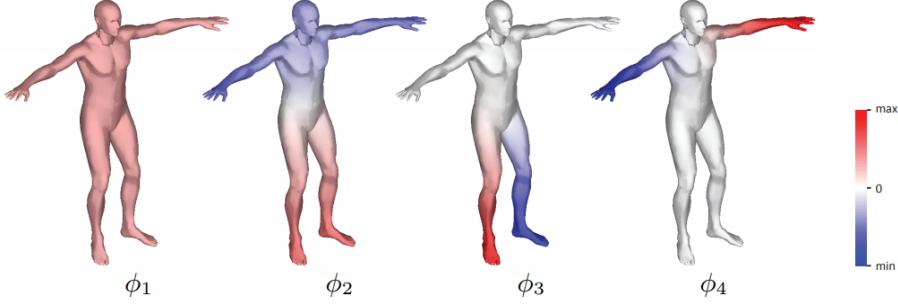


Figure 56: First four eigenfunctions of the manifold Laplacian

The analogy holds even for the Graph Laplacian, a Laplacian that ignores the areas (it does not care about triangles and faces). In Figure 57 we have a function vertex-based being represented in the graph Laplacian basis, we can see the sines and cosines on a graph.

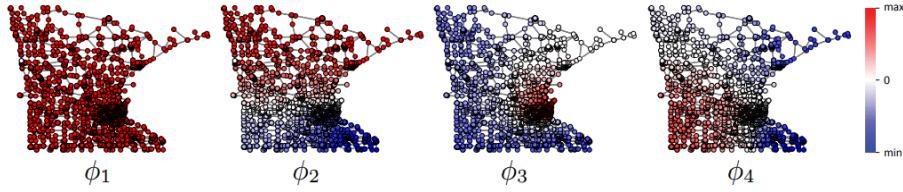


Figure 57: First four eigenfunctions of the graph Laplacian

This analogy is very useful and very nice because with this intuition we can basically do signal processing on functions defined on a surface, e.g. we can do low pass filtering.

15.10.1 Fourier analysis

Euclidean space

This is the classical Fourier analysis, where a function $f : [-\pi, \pi] \rightarrow \mathbb{R}$ can be written as a *Fourier series*, a linear combination of the Fourier basis (sines and cosines):

$$f(x) = \sum_{k \geq 0} \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x') e^{-ikx'} dx' e^{ikx}$$

The Fourier basis is equal to the 1D Laplacian eigenfunctions, defined as:

$$\frac{d^2}{dx^2} e^{ikx} = k^2 e^{ikx} \quad (95)$$

that is, the second derivative. The equivalence in (95) makes clear that the exponential e^{ikx} is an eigenfunction of the operator, with eigenvalue k^2 .

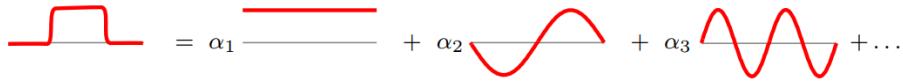


Figure 58: Fourier basis = 1D Laplacian eigenfunctions

Non-Euclidean space

For manifolds we have the same thing, we can write a function defined on a manifold $f : \mathcal{X} \rightarrow \mathbb{R}$ as a *Fourier series*:

$$f(x) = \sum_{k \geq 1} \int_{\mathcal{X}} f(x') \phi_k(x') dx' \phi_k(x)$$

The Fourier basis is equal to the 2D Laplacian eigenfunctions:

$$\Delta \phi_k(x) = \lambda_k \phi_k(x)$$

May be necessary
to improve this
section (Analogy
Fourier - Lapla-
cian)

So, for manifold we have exactly the same relationship, exactly the same formula.

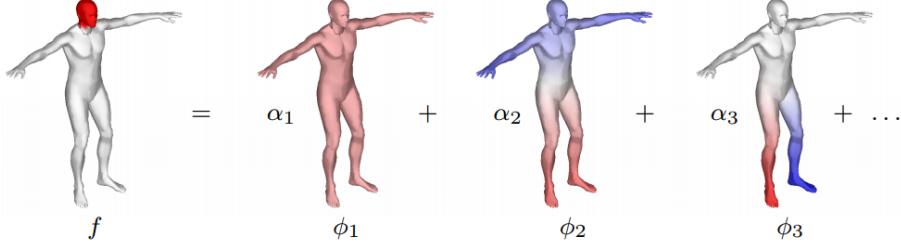


Figure 59: Fourier basis = 2D Laplacian eigenfunctions

Example

In the Euclidean space, we can take a function f , express it as a linear combination of the Fourier basis but then cut this combination to the first k coefficients. We forget about the high frequencies, we're just taking the k low frequencies.

Then, with these first k we can reconstruct the signal. We get the smoothed signal, this is known as *low-pass filtering*, smoothing or blurring operator of the signal.

We can do the same thing with functions on manifolds. We express the function f with the first k Laplacian eigenfunctions, and we get a smooth version of this function. This is the key tool of spectral geometry.

Remark 50. The eigenvalues are invariant to isometries. Also the eigenfunctions are invariant to isometries.

Remark 51. It can be proven that the Laplace-Beltrami eigenbasis is the optimal basis for smooth functions. There isn't a better basis that we can choose to represent a smooth function. This is known as the L_2 property of the Fourier basis, in classical signal processing in Euclidean spaces.

Remark 52. Consider two isometric shapes \mathcal{X} and \mathcal{Y} with Laplacian eigenfunctions $\{\phi_i\}$ and $\{\psi_j\}$ respectively spanning the functional spaces $\mathcal{F}(\mathcal{X})$ and $\mathcal{F}(\mathcal{Y})$. Let $T : \mathcal{F}(\mathcal{X}) \rightarrow \mathcal{F}(\mathcal{Y})$ be the ground-truth map. The matrix representation of T in the Laplacian Eigenbasis looks like a $k \times k$ diagonal matrix, with ones and minus ones on the diagonal. The minus comes from the fact that the eigenvalues and eigenfunctions are isometry invariant, e.g. reflections of the shapes have the same eigenfunctions and eigenvalues.

16 Heat diffusion

The heat diffusion is a physical phenomenon describing the diffusion of heat on some surface.

16.1 Heat equation

The heat diffusion is governed by the *heat equation*

$$\begin{aligned}\frac{\partial}{\partial t} u(x, t) &= -\Delta u(x, t) \\ u(x, 0) &= u_0(x)\end{aligned}\tag{96}$$

where function u describes the *heat distribution* at point x after time t (Figure 60), that is, the amount of heat that is present at point x in time t . Note that the initial distribution $u_0(x)$ is given, and can be any function on the surface.
Remark 53. The minus sign in (96) is just a convention.

The function u is a function of space and time, regarding space we are using a single symbol x but it could be a point in \mathbb{R} , \mathbb{R}^3 or on a surface.

In Figure 60 we can see that the heat distribution follows a typical isotropic behaviour, diffusing equally in all directions. The same behaviour will be observed even on surfaces, however we will ignore all the material properties of the surface (e.g. our surfaces do not have any material, what changes is that instead of the Laplacian we get a slightly different operator).

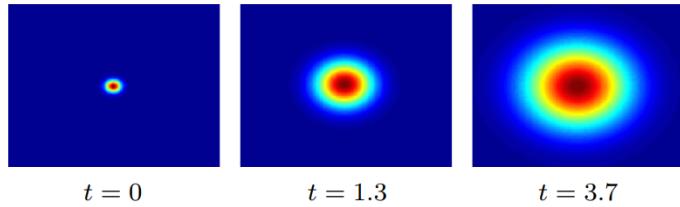


Figure 60: The heat distribution at different time t

We will assume that our surfaces do not have any boundary, otherwise we have to talk about some discrete differential geometry.

16.2 Heat kernel

A solution, that is finding the function u , to the *heat equation* is obtained by computing this integral:

$$u(x, t) = \int_{\mathcal{X}} k_t(x, y) u_0(y) \, dy\tag{97}$$

Note that this integral is an integral on our domain, and it involves the initial condition $u_0(x)$ that is given by us, and it involves another function k_t that depend on the time parameter t .

The function $k_t : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is called the *heat kernel* of \mathcal{X} , and we are going to provide an explicit expression for the heat kernel. Once we have an expression for the heat kernel, we can apply (97) and obtained the solution for the heat equation.

Note that the heat kernel $k_t(x, y)$:

- intuitively describes the amount of *heat transferred* from point x to point y in time t
- is a *symmetric function*, the amount of heat transferred from x to y and vice-versa is the same
- must satisfy the *kernel* properties, that we won't discuss
- is a *property of the manifold \mathcal{X}* and does not depend on the initial distribution $u_0(x)$, the only thing that matters for computing the heat kernel is the geometry itself. So, heat diffusion is completely determined by the geometry of the domain.

Remark 54. Heat kernel and heat diffusion was studied by Einstein, basically he gave a relationship between heat diffusion and random walks.

16.3 Dirac initialization

Definition 17 (dirac delta). Let \mathcal{X} be a mesh, given an arbitrary point $z \in \mathcal{X}$, the *Dirac delta* functions at z is a function δ_z that is non-zero only at z , and, it satisfies the sampling property:

$$\int_{\mathcal{X}} f(x) \delta_z(x) dx = f(z)$$

This sampling property is even defining the Dirac delta function δ_z .

Assume the initial distribution is a *Dirac delta* at point $z \in \mathcal{X}$. That is, there is a high amount of heat in z and zero elsewhere.

With this initialization, the solution to heat equation is simply

$$u(x, t) = \int_{\mathcal{X}} k_t(x, y) \delta_z(y) dy = k_t(x, z)$$

due to the sampling property of the Dirac distribution. The δ_z is operating on the y , that is sampling the y at z , so the solution of the integral is given by $k_t(x, z)$. Note that z is *fixed*, where we decided to put the initial distribution of heat, so here $k_t(x, z)$ is a function of x .

Thus, the solution of the heat equation whenever we choose the Dirac delta as initialization distribution at some point, is given by the heat kernel. It gives us a way to compute the heat kernel.

16.4 Heat kernel: Properties

The heat kernel has some interesting properties, that we will not prove:

- In Euclidean space \mathbb{R}^n there is an explicit expression for the heat kernel.
It is given by

$$k_t(x, y) = \frac{1}{(\sqrt{4\pi t})^n} \exp\left(-\frac{\|x - y\|^2}{4t}\right)$$

so, up to some global scaling, the heat kernel is basically a Gaussian in space. It is a bump, a symmetric bump living on the Euclidean space. So it is going to be isotropic and from the formula it is clear that the variance of this bump, so the spread of this bump, directly depends on the diffusion time. Another interesting thing, is that the heat kernel directly depends on the distance in the Euclidean domain, so if I have estimate for the heat kernel (e.g. some sensors measuring temperature) then I can solve for distance between points. Since I don't know how to measure distance on surfaces in an easy way, it would be cool to be able to compute the heat diffusion in an easy way and then solve for geodesic distances

- *Distance* can be recovered by Varadhan's result:

$$d_{\mathcal{X}}^2(x, y) = -\lim_{t \rightarrow 0} 4t \log(k_t(x, y))$$

It states that the relationship between heat diffusion and distance holds for any manifold.

- If $T : \mathcal{X} \rightarrow \mathcal{Y}$ is an *isometry*, then:

$$k_t^{\mathcal{X}}(x, y) = k_t^{\mathcal{Y}}(T(x), T(y)) \quad (98)$$

Given two isometric shapes, heat kernel at corresponding point are the same. This is an if and only if, so if there exist a mapping T such that (98) is true for all pairs of points, then T is an isometry and the two shapes are isometric. This means in practice that heat diffusion is invariant to isometries. We can exploit this we can use if we are trying to solve some matching problem, for example we can compute heat diffusion at some points, compare the heat values and then based on this we can decide whether some point is matching some other point or not

16.5 Heat kernel: Spectral decomposition

There are many methods to compute the heat kernel, we are going to compute its decomposition in the Laplacian eigenbasis.

This is the heat equation that we want to solve

$$\begin{aligned} \frac{\partial}{\partial t} u(x, t) &= -\Delta u(x, t) \\ u(x, 0) &= u_0(x) \end{aligned}$$

We are going to write the solution u , which is unknown. It is a function on our geometric domain, the surface, so it can be expressed in some basis for functions on surfaces, for example in the Laplacian eigenbasis $\{\phi_i\}$.

So we can write:

$$u(x, t) \approx \sum_{i=0}^k c_i(t) \phi_i(x) \quad (99)$$

where it is an approximation since we are considering only the first k eigenfunctions, truncating the summation performing a low pass filtering.

We write the coefficients of the linear combination of the basis $c_i(t)$ as function of t , because we are considering this particular scalar function $u(x)$ at time t (hence we write $u(x, t)$). So for a different diffusion time t we will have different linear combination coefficients.

Thus, the right-hand side becomes:

$$\Delta u(x, t) = \Delta \sum_{i=0}^k c_i(t) \phi_i(x) \quad (100)$$

$$= \sum_{i=0}^k c_i(t) \Delta \phi_i(x) \quad (101)$$

$$= \sum_{i=0}^k c_i(t) \lambda_i \phi_i(x) \quad (102)$$

where (100) is due to (99), (101) is due to the linearity of the Laplacian and it is going to be applied to the functions ϕ_i , in (102) we are applying the definition of eigenfunctions $\Delta \phi = \lambda \phi$.

So we can substitute it in (96):

$$\frac{\partial}{\partial t} u(x, t) = - \sum_{i=0}^k c_i(t) \lambda_i \phi_i(x) \quad (103)$$

$$\frac{\partial}{\partial t} \sum_{i=0}^k c_i(t) \phi_i(x) = - \sum_{i=0}^k c_i(t) \lambda_i \phi_i(x) \quad (104)$$

$$\sum_{i=0}^k \frac{\partial}{\partial t} c_i(t) \phi_i(x) = - \sum_{i=0}^k c_i(t) \lambda_i \phi_i(x) \quad (105)$$

$$\frac{\partial}{\partial t} c_i(t) = -c_i(t) \lambda_i \quad (106)$$

$$c_i(t) = d_i e^{-\lambda_i t} \quad (107)$$

where

- in (103), (104) and (105) we are substituting (99) and exploiting the linearity of the derivative

- (106) is the only way in which the equality (105) can be true. Note that it is not always true since I could achieve the same sum choosing different coefficients in general, it is so since the ϕ are a basis hence linearly independent, so the only way is to have the same coefficients.
- (107) is the solution to the differential equation. We are saying that $f(t)' = -f(t)a$ has as solution $f(t) = de^{-at}$, to check it we just need to plug-in the solution $(de^{-at})' = d(e^{-at})' = -ade^{-at} = -af(t)$, so it works.

We arrive at

$$u(x, t) \approx \sum_{i=0}^k d_i e^{-\lambda_i t} \phi_i(x) \quad (108)$$

where in (108) we are just substituting (107) into (99).

We have a nice expression for the heat equations, it involves purely the eigenfunctions and eigenvalues of the Laplacian. We know exactly how to compute them, we just need to find an expression for the coefficients d_i .

We use the given *initial condition* to solve for d_i , depending on the initial distribution we will have a different solution for d_i , we are going to choose the Dirac delta.

For $t = 0$, it must be:

$$u(x, 0) \approx \sum_{i=0}^k d_i e^{-\lambda_i 0} \phi_i(x) \quad (109)$$

$$u(x, 0) \approx \sum_{i=0}^k d_i \phi_i(x) \quad (110)$$

$$u_0(x) \approx \sum_{i=0}^k d_i \phi_i(x) \quad (111)$$

where in (109) we are substituting t with 0, in (110) we use the classic property $a^0 = 1$, in (111) we are just changing the notation, since the distribution at $t = 0$ is known, given as $u_0(x)$.

We are expressing the initial condition as a linear combination of the Laplacian eigenfunctions (111), which is a orthonormal basis. So we can solve for the combination coefficients by projecting $u_0(x)$ on the Laplacian eigenbasis.

With the Dirac delta as initial condition $u_0(x) = \delta_y(x)$ we get:

$$\delta_y(x) \approx \sum_{i=0}^k d_i \phi_i(x) \implies d_i = \langle \phi_i, \delta_y \rangle \chi \quad (112)$$

note that the inner product in (112) is an integral:

$$\langle \phi_i, \delta_y \rangle_{\mathcal{X}} = \int_{\mathcal{X}} \phi_i(x) \delta_y(x) \quad (113)$$

$$= \phi_i(y) \quad (114)$$

where (113) is due to the definition of inner products on manifolds, (114) is due to the sampling property of δ_y .

Therefore, the solution of the heat equation if $u_0(x) = \delta_y(x)$, is simply given by

$$\begin{aligned} u(x, t) &\approx \sum_{i=0}^k e^{-\lambda_i t} \phi_i(x) \phi_i(y) \\ k_t(x, t) &\approx \sum_{i=0}^k e^{-\lambda_i t} \phi_i(x) \phi_i(y) \end{aligned} \quad (115)$$

We choose the Dirac delta as initialization because we found that when we take this particular choice, then the solution of the heat equation is the heat kernel. Now we have an explicit expression for the heat kernel on a manifold, and everything only depends on the manifold itself.

Remark 55. The minus on the eigenvalues λ comes from the convention we took, otherwise it would be weird to have a positive exponential. Since it would seem like it is a diverging sum, but the point is that these are the eigenvalue of the eigenvalue of the Laplacian, and we defined them as always positive. Another possibility is to define the Laplacian with negative eigenvalues, and you do it by not putting the minus in the divergence of gradient definition of the Laplacian.

In *matrix notation*, the heat kernel (115) can be written as a $n \times n$ matrix

$$\mathbf{K}_t = \Phi \operatorname{diag}(e^{-\lambda_i t}) \Phi^\top$$

where each column of Φ is an eigenfunction of the Laplacian, so it is a $n \times k$ matrix, and, in the middle there is a $k \times k$ diagonal matrix.

Note that it is the heat kernel at time t , the moment we change the time t we have a different matrix hence a different kernel.

Remark 56. The matrix formulation of the kernel seems like the eigendecomposition of another matrix, it has some implications that we will not cover.

16.6 Heat kernel signature

We stated that:

If $T : \mathcal{X} \rightarrow \mathcal{Y}$ is an *isometry*, then

$$k_t^{\mathcal{X}}(x, y) = k_t^{\mathcal{Y}}(T(x), T(y))$$

and the vice-versa also holds.

We use this property to define a *local descriptor* based on the heat kernel. The amount of heat remaining in a given point after time t , this amount of heat is given by the diagonal of the heat kernel, which is the same as looking for the heat kernel from x to x :

$$k_t(x, x) = \sum_{i=0}^k e^{-\lambda_i t} \phi_i(x)^2$$

The *heat kernel signature* is then defined as the local descriptor at different times:

$$\text{hks}(x) = (k_{t_1}(x, x), \dots, k_{t_T}(x, x)) \in \mathbb{R}^T$$

If \mathcal{X} and \mathcal{Y} are isometric, corresponding points $(x, y) \in \mathcal{X} \times \mathcal{Y}$ are expected to have similar signatures.

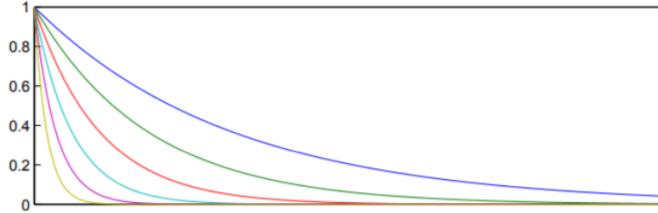


Figure 61: The amount of heat remaining in the point while the time increases

16.7 Diffusion distance

The heat kernel doesn't define a *distance* function. It can be easily seen since $k_t(x, x)$ isn't zero.

$$k_t(x, t) \approx \sum_{i=0}^k e^{-\lambda_i t} \phi_i(x) \phi_i(y)$$

A family of *diffusion distances* can be defined by taking the L_2 distance between

two heat kernels:

$$\begin{aligned} d_t^2(x, y) &= \|k_t(x, \cdot) - k_t(\cdot, y)\|^2 \\ &= \int_{\mathcal{X}} (k_t(x, z) - k_t(z, y))^2 dz \end{aligned}$$

where the integral comes from the definition of inner product on manifolds.

Note that:

- The definition above satisfies all properties of a *metric*
- Diffusion time $t \geq 0$ plays the role of a *scale* parameter
- Interpretation: If two points x and y are close, there is a large *probability of transition* from x to y and vice-versa. This is because heat diffusion can be seen as a probability density function of a random walker on a surface.

$$d_t^2(x, y) = \|k_t(x, \cdot) - k_t(\cdot, y)\|^2 \quad (116)$$

$$= \int_{\mathcal{X}} (k_t(x, z) - k_t(z, y))^2 dz \quad (117)$$

$$= \int_{\mathcal{X}} k_t(x, z)^2 + k_t(z, y)^2 - 2k_t(x, z)k_t(z, y) dz \quad (118)$$

$$= \int_{\mathcal{X}} k_t(x, z)k_t(z, x) + k_t(y, z)k_t(z, y) - 2k_t(x, z)k_t(z, y) dz \quad (119)$$

$$= k_{2t}(x, x) + k_{2t}(y, y) - 2k_{2t}(x, y) \quad (120)$$

where (116) and (117) is the definition, (118) is because it is a square of binomial, (119) is because the heat kernel is symmetric, in (120) the intuition is that if I go from x to z in time t and then I go from z to x in time t , it's like going from x to x in $2t$.

Remark 57. This thing can be generalized, if you replace the heat kernel with some other kernel.

We know how to express the heat kernel in terms of eigenfunctions and eigenvalues, we can use this knowledge to compute all the kernels in (116) and compose them obtaining the simpler expression (121):

$$\begin{aligned} d_t^2(x, y) &= \|k_t(x, \cdot) - k_t(\cdot, y)\|^2 \\ &= \dots \\ &= \sum_{i=0}^k e^{-2\lambda_i t} (\phi_i(x) - \phi_i(y))^2 \end{aligned} \quad (121)$$

Notes

■ May be necessary to deepen the ‘first fundamental theorem of calculus’	31
■ May be necessary to deepen the notational trick that brings dt on the right side	31
■ May be necessary to deepen the definition and notation of ‘surface integral’	32
■ May be necessary to prove that the norm of the cross product is the area of the parallelogram	32
■ May be necessary to add a paragr ‘area element of vertices’	38
■ May be necessary to deepen the ‘Riesz representation theorem’	42
■ May be necessary to improve this section (Analogy Fourier - Laplacian)	78