Geometry Processing (601.458/658)

Misha Kazhdan

Outline

Recall

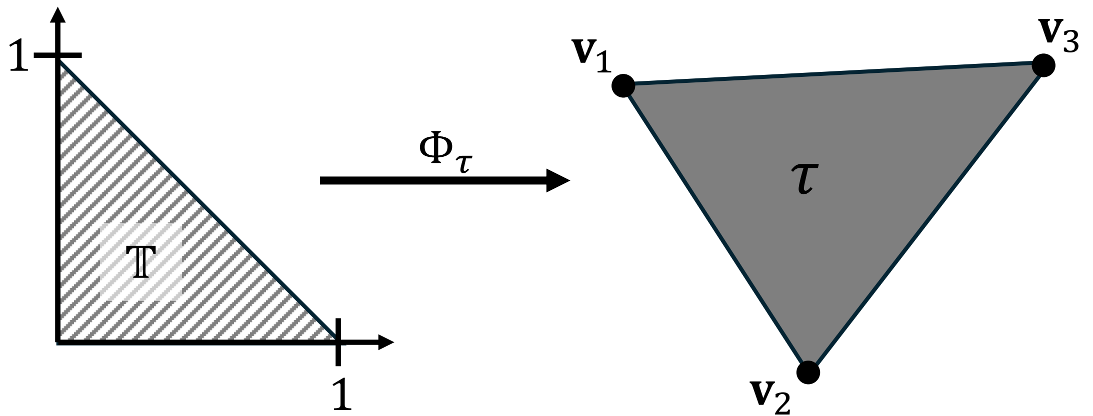
Triangle Meshes

Gradient Descent Revisited

Recall

Right Triangle Parameterization:

Given a triangle $\tau \subset \mathbb{E}^3$, we perform all our computation over the unit right triangle $\mathbb{T} \subset \mathbb{E}^2$, using the metric pulled back via the parameterization

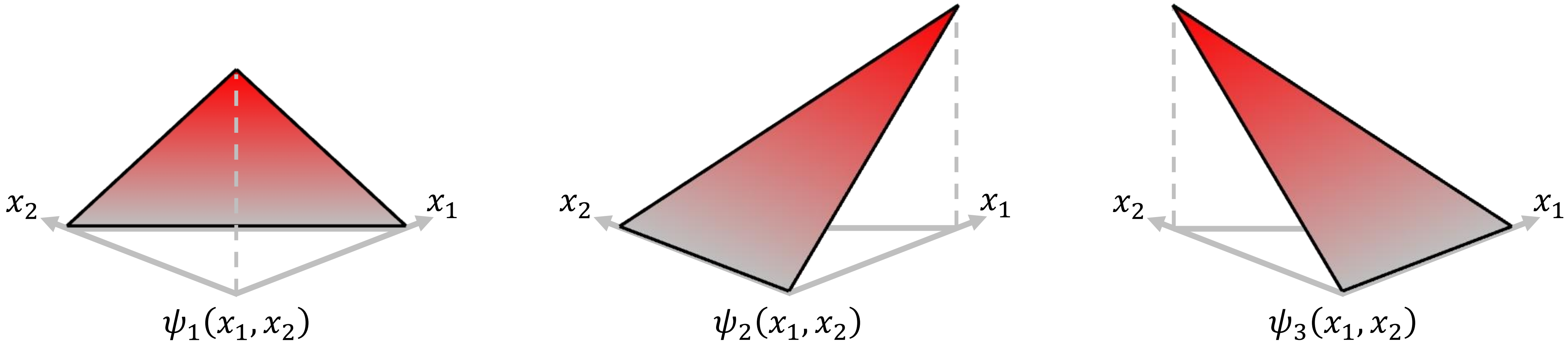


Recall

Hat Basis:

We discretize the space of functions on a triangle using the “hat” basis over the right triangle:

$$\psi_1(x_1, x_2) = 1 - x_1 - x_2$$
$$\psi_2(x_1, x_2) = x_1$$
$$\psi_3(x_1, x_2) = x_2$$



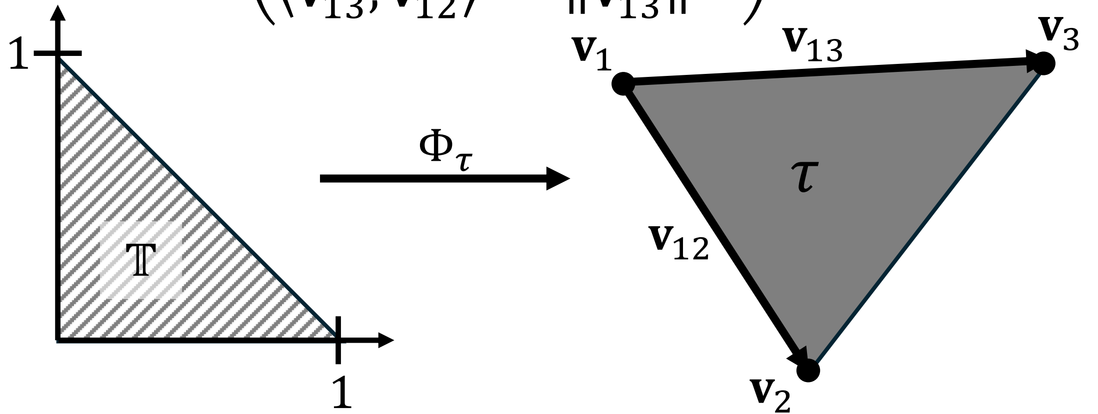
Recall

For $\mathbf{p} \in \mathbb{T}$, the differential $\mathbf{d}\Phi_\tau|_\mathbf{p} \in \mathbb{R}^{3\times 2}$ is the matrix whose columns are the differences:

$$\mathbf{d}\Phi_\tau\Big|_\mathbf{p} = (\mathbf{v}_{12} \quad \mathbf{v}_{13})$$

The pulled-back inner-product, expressed in matrix form w.r.t. the cartesian basis $\{\partial_1|_\mathbf{p}, \partial_2|_\mathbf{p}\} \subset T_\mathbf{p}\mathbb{T}$ is:

$$\mathbf{g}_\tau = \begin{pmatrix} \|\mathbf{v}_{12}\|^2 & \langle \mathbf{v}_{12}, \mathbf{v}_{13} \rangle \\ \langle \mathbf{v}_{13}, \mathbf{v}_{12} \rangle & \|\mathbf{v}_{13}\|^2 \end{pmatrix}$$



Recall

$$\mathbf{g}_\tau = \begin{pmatrix} \|\mathbf{v}_{12}\|^2 & \langle \mathbf{v}_{12}, \mathbf{v}_{13} \rangle \\ \langle \mathbf{v}_{13}, \mathbf{v}_{12} \rangle & \|\mathbf{v}_{13}\|^2 \end{pmatrix}$$

Given functions:

$$f(x_1, x_2) = \mathbf{f}_1 \cdot \psi_1(x_1, x_2) + \mathbf{f}_2 \cdot \psi_2(x_1, x_2) + \mathbf{f}_3 \cdot \psi_3(x_1, x_2)$$
$$h(x_1, x_2) = \mathbf{h}_1 \cdot \psi_1(x_1, x_2) + \mathbf{h}_2 \cdot \psi_2(x_1, x_2) + \mathbf{h}_3 \cdot \psi_3(x_1, x_2)$$

the inner-product of f and h, w.r.t the inner-product pulled back from τ is:

$$\langle\langle f, h \rangle\rangle_\tau = \mathbf{f}^\top \cdot \mathbf{m}^\tau \cdot \mathbf{h}$$

Here $\mathbf{m}^\tau \in \mathbb{R}^{3\times3}$ is the matrix:

$$\mathbf{m}^\tau_{ij} = \sqrt{\det(\mathbf{g}_\tau)} \cdot \langle\langle \psi_i, \psi_j \rangle\rangle_{\mathbb{T}}$$

Recall

$$\mathbf{g}_\tau = \begin{pmatrix} \|\mathbf{v}_{12}\|^2 & \langle \mathbf{v}_{12}, \mathbf{v}_{13} \rangle \\ \langle \mathbf{v}_{13}, \mathbf{v}_{12} \rangle & \|\mathbf{v}_{13}\|^2 \end{pmatrix}$$

Given a function:

$$f(x_1, x_2) = \mathbf{f}_1 \cdot \psi_1(x_1, x_2) + \mathbf{f}_2 \cdot \psi_2(x_1, x_2) + \mathbf{f}_3 \cdot \psi_3(x_1, x_2)$$

the Dirichlet energy of f, w.r.t the inner-product pulled back from τ is:

$$\langle\langle df, df \rangle\rangle_\tau = \mathbf{f}^\top \cdot \mathbf{s}^\tau \cdot \mathbf{h}$$

Here $\mathbf{s}^\tau \in \mathbb{R}^{3\times 3}$ is the matrix:

$$\mathbf{s}_{ij}^\tau = \mathbf{s}_{ij}^\tau = \left\langle \sqrt{\det(\mathbf{g}_\tau)} \cdot \mathbf{g}_\tau^{-1}, \mathbf{d}^{ij} \right\rangle_F$$

And $\mathbf{d}^{ij} \in \mathbb{R}^{2\times 2}$ are the matrices:

$$\mathbf{d}^{ij} = \int_{\mathbb{T}} \mathbf{d}\psi_i \cdot \mathbf{d}\psi_i{}^\top \cdot \omega_\tau$$

Outline

Recall

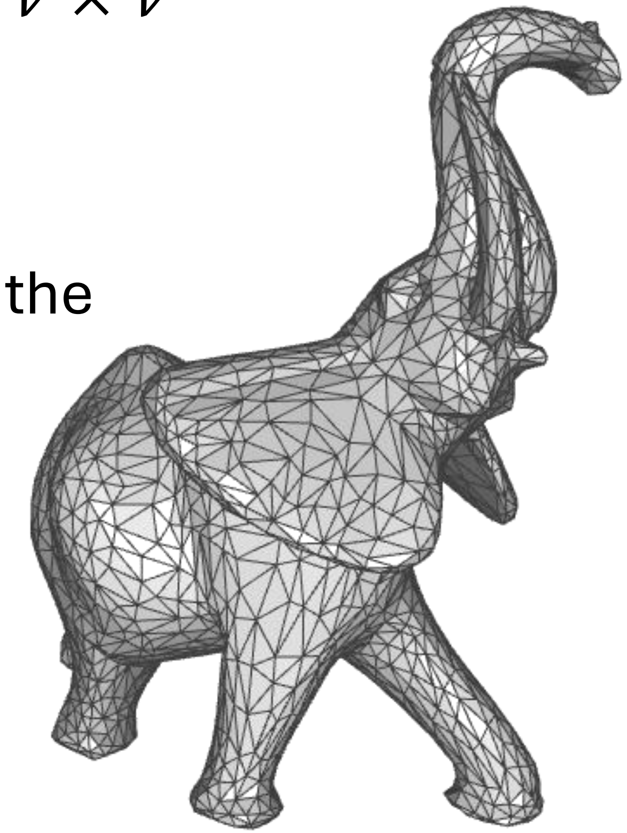
Triangle Meshes

Gradient Descent Revisited

Triangle Meshes

A triangle mesh, $\mathcal{M} = \{\mathcal{V}, \mathcal{T}\}$, is represented by:

- A *vertex set* $\mathcal{V}$
- A *triangle set* consisting of triplets of vertex indices $\mathcal{T} \subset \mathcal{V} \times \mathcal{V} \times \mathcal{V}$



Note:

- Re-ordering vertex indices within a triangle does not change the triangle's geometry.

For simplicity:

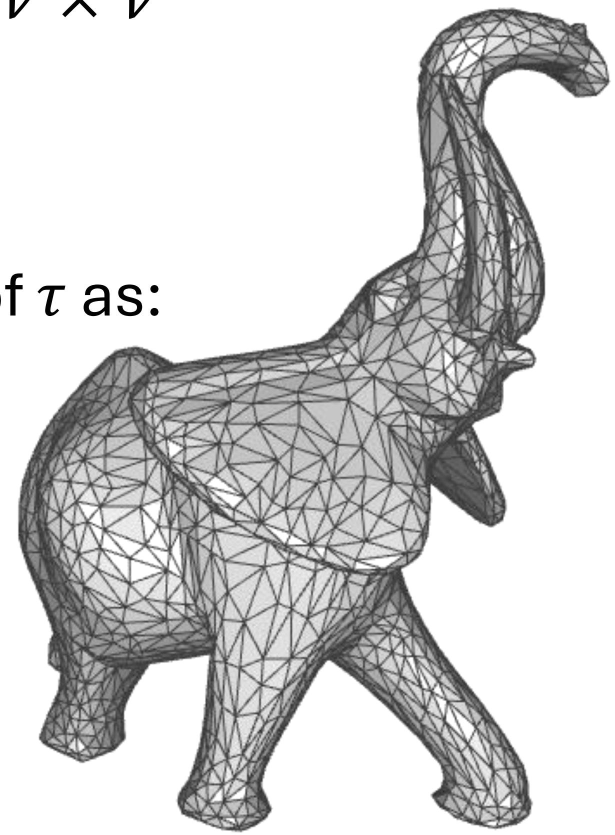
- We associate vertex indices with consecutive integers in the range $\{1, \ldots, |\mathcal{V}|\}$.

Triangle Meshes

A triangle mesh, $\mathcal{M} = \{\mathcal{V}, \mathcal{T}\}$, is represented by:

A *vertex set* $\mathcal{V}$

A *triangle set* consisting of triplets of vertex indices $\mathcal{T} \subset \mathcal{V} \times \mathcal{V} \times \mathcal{V}$



Notation:

For triangle $\tau \in \mathcal{T}$ and $1 \leq k \leq 3$, we denote the k-th vertex of τ as:

$$\tau[k] \in \mathcal{V}$$

We denote the set of triangles incident on vertex $v \in \mathcal{V}$ as:

$$\mathcal{T}(v) \subset \mathcal{T}$$

For $\tau \in \mathcal{T}(v)$, we denote by $\tau(v) \in \{1,2,3\}$ the index of the vertex within in the triangle equal to v:

$$\tau[\tau(v)] = v$$

Triangle Meshes

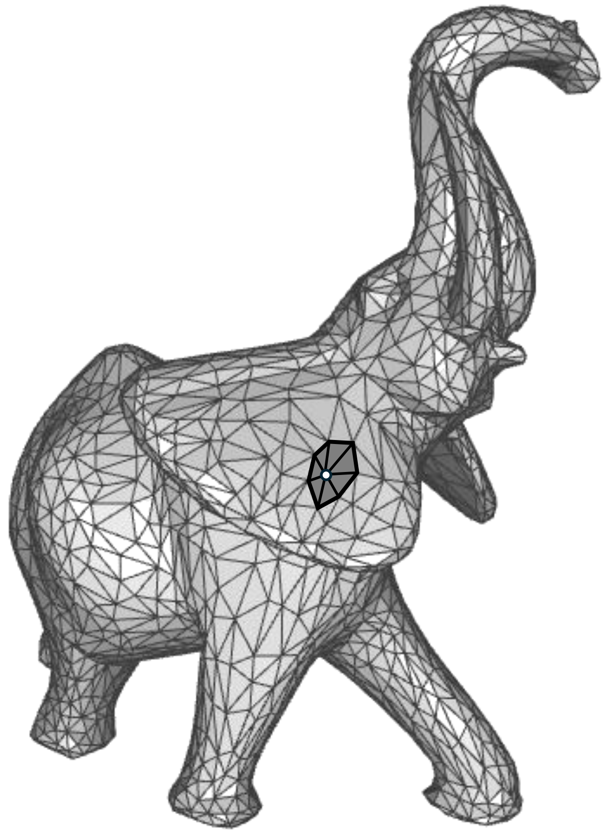
Given a triangle mesh, $\mathcal{M} = \{\mathcal{V}, \mathcal{T}\}$, we define a function space by assigning a piecewise-linear function each vertex $\{\phi_v : \mathcal{M} \to \mathbb{R}\}$.

The function ϕ_v:

Evaluates to one at vertex v

Evaluates to zero at every other vertex

Is linear within each triangle



$\Rightarrow$ The function ϕ_v is supported (i.e. non-zero) on the triangles incident on vertex v.

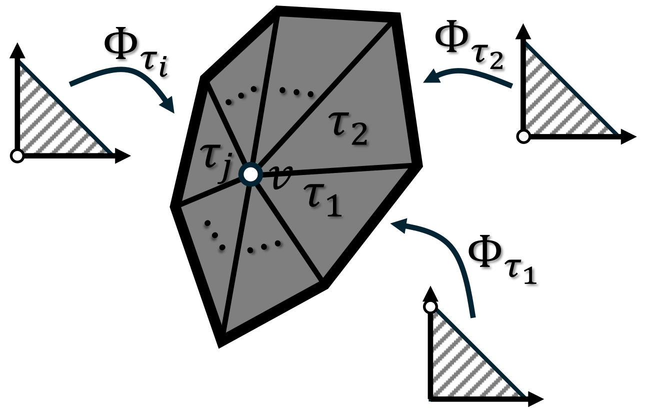
$\Rightarrow$ The functions $\{\phi_v\}$ form a partition of unity.

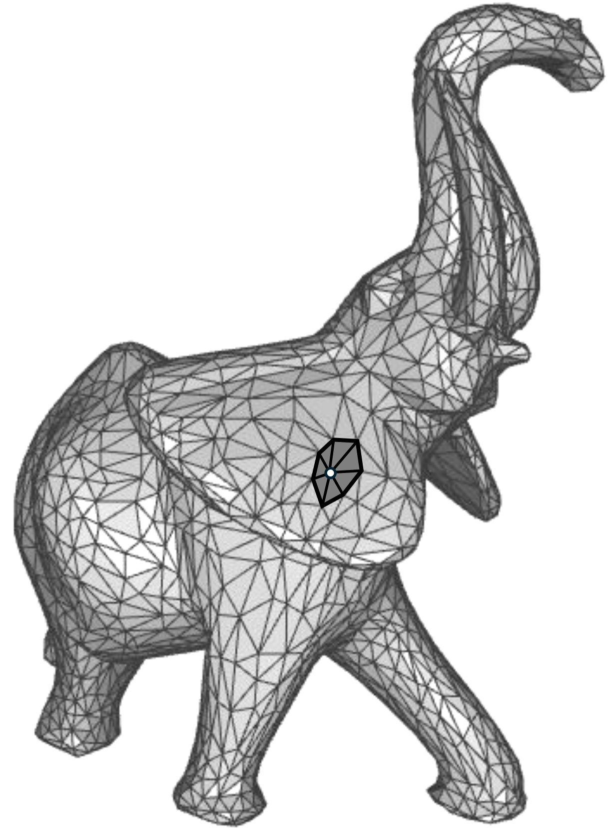
Triangle Meshes

Given a triangle mesh, $\mathcal{M} = \{\mathcal{V}, \mathcal{T}\}$, we define a function space by assigning a piecewise-linear function each vertex $\{\phi_v : \mathcal{M} \to \mathbb{R}\}$.

Property:

Given a vertex $v \in \mathcal{V}$, each of the incident triangles can be parameterized over the unit right triangle.





Triangle Meshes

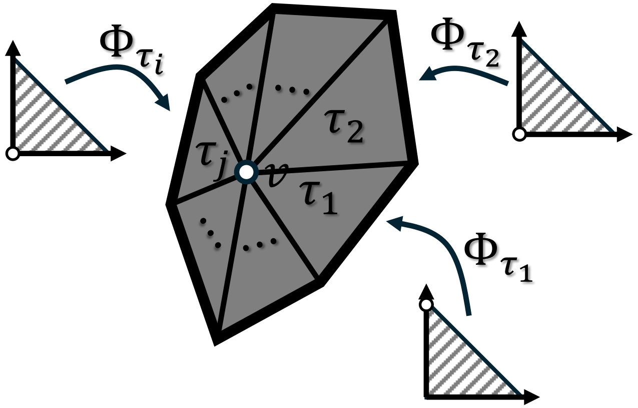
Given a triangle mesh, $\mathcal{M} = \{\mathcal{V}, \mathcal{T}\}$, we define a function space by assigning a piecewise-linear function each vertex $\{\phi_v: \mathcal{M} \to \mathbb{R}\}$.

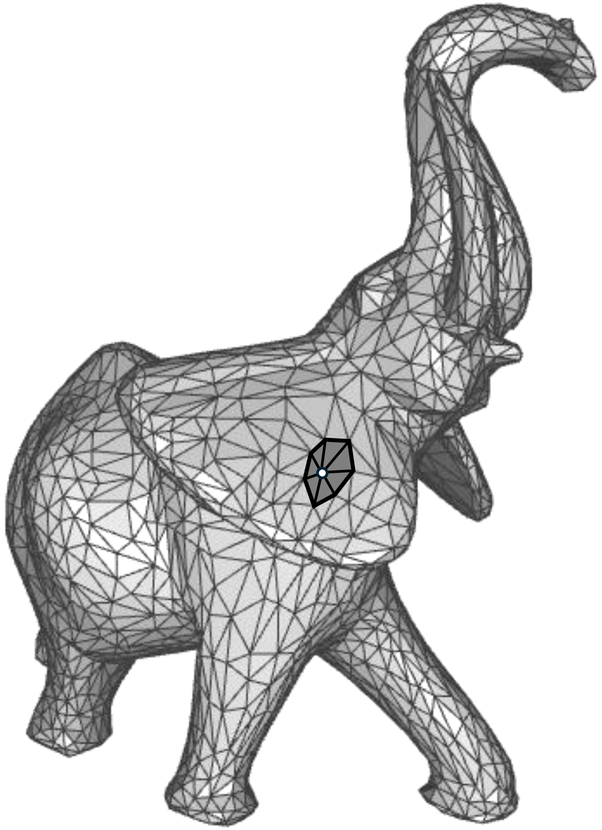
Property:

Given a vertex $v \in \mathcal{V}$, each of the incident triangles can be parameterized over the unit right triangle.

Restricted to an incident triangle $\tau \in \mathcal{T}(v)$, the function ϕ_v is a hat basis function on $\mathbb{T}$:

$$\phi_v\big(\Phi_\tau(\mathbf{p})\big) = \psi_{\tau(v)}(\mathbf{p})$$





Triangle Meshes

The collection of functions $\{\phi_v\}_{v\in\mathcal{V}}$ defines a $|\mathcal{V}|$-dimensional vector space V.

We would like to:

- Turn this into an inner-product space, defining a symmetric positive-definite bilinear form $M: V \to V^*$ such that for all $f, h \in V$:
$$[M(f)](h) = M(f, h) = \langle\langle f, h\rangle\rangle_{\mathcal{M}}$$

- Define a symmetric positive semi-definite bilinear form, giving the stiffness, $S: V \to V^*$, such that for all $f, h \in V$:
$$[S(f)](h) = \langle\langle df, dh\rangle\rangle_{\mathcal{M}}$$

Triangle Meshes

The collection of functions $\{\phi_v\}_{v\in\mathcal{V}}$ defines a $|\mathcal{V}|$-dimensional vector space V.

We would like to:

- Turn this into an inner-product space
- Define a symmetric positive semi-definite bilinear stiffness form

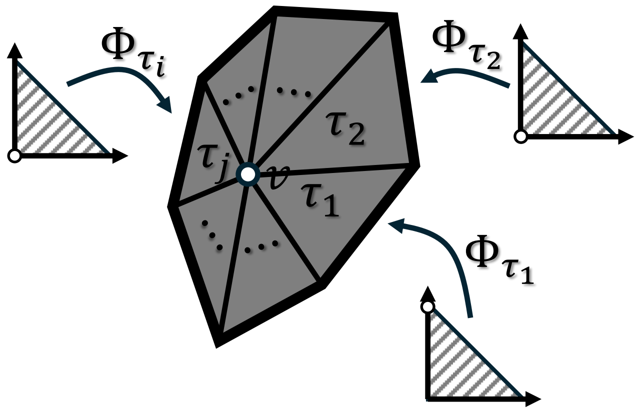
For both, we perform the integration one triangle at a time, reducing the problem to computing integrals, differentials, and inner-products over the unit right triangle.

We seek an expression of the bilinear forms as $\mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ matrices.

Mass Matrix

The mass matrix, $\mathbf{M} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$, is defined by integrating pairs of functions over the mesh:

$$\begin{aligned}
\mathbf{M}_{uv} &= \langle\langle \phi_u, \phi_v \rangle\rangle_{\mathcal{M}} \\
&= \int_{\mathcal{M}} \phi_u \cdot \phi_v \cdot \omega_{\mathcal{M}} \\
&= \sum_{\tau\in\mathcal{T}} \int_{\tau} \phi_u \cdot \phi_v \cdot \omega_{\mathcal{M}} \\
&= \sum_{\tau\in\mathcal{T}(u)\cap\mathcal{T}(v)} \int_{\tau} \phi_u \cdot \phi_v \cdot \omega_{\tau} \\
&= \sum_{\tau\in\mathcal{T}(u)\cap\mathcal{T}(v)} \int_{\mathbb{T}} \psi_{\tau(u)} \cdot \psi_{\tau(v)} \cdot \omega_{\tau} \\
&= \sum_{\tau\in\mathcal{T}(u)\cap\mathcal{T}(v)} \mathbf{m}^{\tau}_{\tau(u),\tau(v)}
\end{aligned}$$



Mass Matrix

The mass matrix, $\mathbf{M} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$, is defined by integrating pairs of functions over the mesh:

$$\mathbf{M}_{uv} = \langle\langle \phi_u, \phi_v \rangle\rangle_{\mathcal{M}}$$

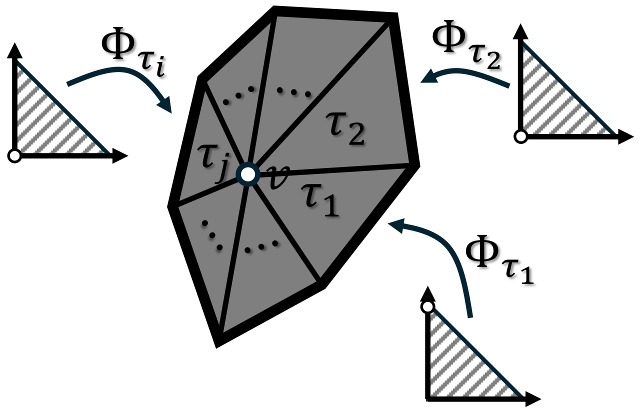
$$= \int \phi_u \cdot \phi_v \cdot \omega_{\mathcal{M}}$$

The coefficient $\mathbf{M}_{uv}$ will be non-zero only if u and v lie in the same triangle.

$$= \sum_{\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)} \int_\tau \phi_u \cdot \phi_v \cdot \omega_\tau$$

$$= \sum_{\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)} \int_{\mathbb{T}} \psi_{\tau(u)} \cdot \psi_{\tau(v)} \cdot \omega_\tau$$

$$= \sum_{\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)} \mathbf{m}^{\tau}_{\tau(u),\tau(v)}$$



Mass Matrix

Properties:

Let $f, h \in \text{Span}\left(\phi_1, \ldots, \phi_{|\mathcal{V}|}\right)$ be two piecewise linear functions on the mesh:

$$f = \mathbf{f}_1 \cdot \phi_1 + \cdots + \mathbf{f}_{|\mathcal{V}|} \cdot \phi_{|\mathcal{V}|}$$
$$h = \mathbf{h}_1 \cdot \phi_1 + \cdots + \mathbf{h}_{|\mathcal{V}|} \cdot \phi_{|\mathcal{V}|}$$

with $\mathbf{f}, \mathbf{h} \in \mathbb{R}^{|\mathcal{V}|}$.

The inner-product of f and h is the matrix/vector product:

$$\langle\langle f, h \rangle\rangle_{\mathcal{M}} = \mathbf{f}^\top \cdot \mathbf{M} \cdot \mathbf{h}$$

Stiffness Matrix

Similarly, the stiffness matrix, $\boldsymbol{S} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$, is defined by integrating the inner-product of differentials of pairs of functions over $\mathcal{M}$:

$$\mathbf{S}_{uv} = \sum_{\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)} \mathbf{s}^{\tau}_{\tau(u),\tau(v)}$$

Properties:

Let $f, h \in \mathrm{Span}\left(\phi_1, \ldots, \phi_{|\mathcal{V}|}\right)$ be two piecewise linear functions on the mesh:

$$f = \mathbf{f}_1 \cdot \phi_1 + \cdots + \mathbf{f}_{|\mathcal{V}|} \cdot \phi_{|\mathcal{V}|}$$
$$h = \mathbf{h}_1 \cdot \phi_1 + \cdots + \mathbf{h}_{|\mathcal{V}|} \cdot \phi_{|\mathcal{V}|}$$

with $\mathbf{f}, \mathbf{h} \in \mathbb{R}^{|\mathcal{V}|}$.

The inner-product of the differentials of f and h is:

$$\langle\langle df, dh \rangle\rangle_{\mathcal{M}} = \mathbf{f}^{\top} \cdot \boldsymbol{S} \cdot \mathbf{h}$$

Stiffness Matrix

Similarly, the stiffness matrix, $\boldsymbol{S} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$, is defined by integrating the inner-product of differentials of pairs of functions over $\mathcal{M}$:

$$\mathbf{S}_{uv} = \sum_{\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)} \mathbf{s}^{\tau}_{\tau(u),\tau(v)}$$

Properties:

The coefficient $\boldsymbol{S}_{uv}$ will be non-zero only if u and v lie in the same triangle.

Let $f, h \in S$ … s on the mesh:

$$f = \mathbf{f}_1 \cdot \phi_1 + \cdots + \mathbf{f}_{|\mathcal{V}|} \cdot \phi_{|\mathcal{V}|}$$
$$h = \mathbf{h}_1 \cdot \phi_1 + \cdots + \mathbf{h}_{|\mathcal{V}|} \cdot \phi_{|\mathcal{V}|}$$

with $\mathbf{f}, \mathbf{h} \in \mathbb{R}^{|\mathcal{V}|}$.

The inner-product of the differentials of f and h is:

$$\langle\langle df, dh \rangle\rangle_{\mathcal{M}} = \mathbf{f}^{\top} \cdot \mathbf{S} \cdot \mathbf{h}$$

Finite Element Assembly

$$\mathbf{M}_{uv} = \sum_{\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)} \mathbf{m}^{\tau}_{\tau(u),\tau(jv)} \quad \text{and} \quad \mathbf{S}_{uv} = \sum_{\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)} \mathbf{s}^{\tau}_{\tau(u),\tau(v)}$$

Naively we would implement this as:

For all $u, v \in \mathcal{V}$:
    For all $\tau \in \mathcal{T}(u) \cap \mathcal{T}(v)$
        Compute $\mathbf{m}^{\tau}$ and $\mathbf{s}^{\tau}$
        $\mathbf{M}_{uv} \leftarrow \mathbf{M}_{uv} + \mathbf{m}^{\tau}_{\tau(u),\tau(v)}$
        $\mathbf{S}_{uv} \leftarrow \mathbf{S}_{uv} + \mathbf{s}^{\tau}_{\tau(u),\tau(v)}$

- ✗ Quadratic in the number of vertices
- ✗ Compute the same triangle's mass and stiffness matrices multiple times

Finite Element Assembly

$$\mathbf{M}_{uv} = \sum_{\tau\in\mathcal{T}(u)\cap\mathcal{T}(v)} \mathbf{m}^{\tau}_{\tau(u),\tau(jv)} \quad \text{and} \quad \mathbf{S}_{uv} = \sum_{\tau\in\mathcal{T}(u)\cap\mathcal{T}(v)} \mathbf{s}^{\tau}_{\tau(u),\tau(v)}$$

In practice, we use *finite-element assembly*:

For all $\tau \in \mathcal{T}$:
    Compute $\mathbf{m}^{\tau}$ and $\mathbf{s}^{\tau}$
    For all $1 \le i, j \le 3$:
        $\mathbf{M}_{\tau[i],\tau[j]} \leftarrow \mathbf{M}_{\tau[i],\tau[j]} + \mathbf{m}^{\tau}_{ij}$
        $\mathbf{S}_{\tau[i],\tau[j]} \leftarrow \mathbf{S}_{\tau[i],\tau[j]} + \mathbf{s}^{\tau}_{ij}$

✓ Linear in the number of non-zero entries

✓ Compute a triangle's mass and stiffness matrices once

Supported by `Eigen`'s `SparseMatrix< double >::setFromTriplets` member function

Algebraic Interpretation

The hat basis defines a $|\mathcal{V}|$-dimensional space of functions:

$$V = \mathrm{Span}\left(\phi_1, \dots, \phi_{|\mathcal{V}|}\right)$$

Mass matrix:

Letting M be the symmetric, positive-definite bilinear map:

$$\begin{aligned} M: V \times V &\to \mathbb{R} \\ (f, h) &\mapsto \langle\langle f, h \rangle\rangle_{\mathcal{M}} \end{aligned}$$

this makes V into an inner-product space $\{V, M: V \to V^*\}$.

$\Rightarrow$ The mass matrix $\mathbf{M} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ gives an expression for the bilinear form $M: V \to V^*$ w.r.t. the bases $\left\{\phi_1, \dots, \phi_{|\mathcal{V}|}\right\}$ and $\left\{\phi_1^*, \dots, \phi_{|\mathcal{V}|}^*\right\}$.

Algebraic Interpretation

The hat basis defines a $|\mathcal{V}|$-dimensional space of functions:

$$V = \text{Span}\left(\phi_1, \dots, \phi_{|\mathcal{V}|}\right)$$

Stiffness matrix:

Similarly, we have a symmetric, positive semi-definite bilinear map:

$$\begin{aligned} S: V \times V &\to \mathbb{R} \\ (f, h) &\mapsto \langle\langle df, dh \rangle\rangle_{\mathcal{M}} \end{aligned}$$

$\Rightarrow$ The stiffness matrix $\mathbf{S} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ gives an expression for the bilinear form $S: V \to V^*$ w.r.t. the bases $\{\phi_1, \dots, \phi_{|\mathcal{V}|}\}$ and $\{\phi_1^*, \dots, \phi_{|\mathcal{V}|}^*\}$.

Caution

We are now working with two types of inner-product spaces.

$\{T_{\mathbf{p}}\mathbb{T}, g_{\mathbf{p}}: T_{\mathbf{p}}\mathbb{T} \to T_{\mathbf{p}}^*\mathbb{T}\}$:

The space of tangent vectors at the point $\mathbf{p} \in \mathbb{T}$, with inner-product $g_{\mathbf{p}}$ defined by pulling back the Euclidean inner-product using the differential of the parametrization $\Phi_\tau: \mathbb{T} \to \tau$.

$\{V, M: V \to V^*\}$:

The space of piecewise-linear functions on the mesh, with inner-product M defined by integrating functions over the mesh.

Outline

Recall

Triangle Meshes

Gradient Descent Revisited

Gradient Descent Revisited

Recall:

Given an inner-product space $\{V, B: V \to V^*\}$ and given a **symmetric** positive semi-definite bilinear form $S: V \to V^*$, we can define an energy:

$$Q_S(v) = \frac{1}{2} \cdot S(v, v) \equiv \frac{1}{2} \cdot [S(v)](v)$$

$Q_S(v) = \frac{1}{2} \cdot [S(v)](v)$

Gradient Descent Revisited

Given $\delta \in V$, the derivative of the energy at v along direction δ is:

$$\begin{aligned}
dQ_S\Big|_v(\delta) = \frac{\partial Q_s}{\partial \delta}\Big|_v &= \lim_{\varepsilon \to 0} \frac{Q_S(v + \varepsilon \cdot \delta) - Q_S(v)}{\varepsilon} \\
&= \lim_{\varepsilon \to 0} \frac{\frac{1}{2} \cdot [S(v + \varepsilon \cdot \delta)](v + \varepsilon \cdot \delta) - \frac{1}{2} \cdot [S(v)](v)}{\varepsilon} \\
&= \frac{1}{2} \cdot \lim_{\varepsilon \to 0} \frac{[S(v)](v) + \varepsilon \cdot [S(v)](\delta) + \varepsilon \cdot [S(\delta)](v) + \varepsilon^2 \cdot [S(\delta)](\delta) - [S(v)](v)}{\varepsilon} \\
&= \frac{1}{2} \cdot \big([S(v)](\delta) + [S(\delta)](v)\big) \\
&= [S(v)](\delta)
\end{aligned}$$

⇒ The differential of Q_S at v is $S(v) \in V^*$

⇒ This is **not** the gradient.

⇒ To get the gradient we need to map the dual vector to a primal one:

$$\nabla_B Q_S\Big|_v \equiv B^{-1}\big(S(v)\big)$$

$$Q_S(v) = \frac{1}{2} \cdot [S(v)](v)$$
$$\nabla_B Q_S \Big|_v = B^{-1}\big(S(v)\big)$$

Gradient Descent Revisited

Problem Statement:

Given an initial vector $v_0 \in V$, we evolve v_0 to reduce the energy defined by the symmetric, positive semi-definite bilinear form S.

We do this by solving for the time varying function $v\colon \mathbb{R}^{\geq 0} \to V$ such that:

$$\frac{\partial v}{\partial t} = -B^{-1}\big(S(v)\big)$$

and $v(t) = v_0$.

$$\frac{\partial v}{\partial t} = -B^{-1}\big(S(v)\big)$$

Gradient Descent Revisited

Time-stepping as before, we get:

Explicit:

$$\frac{v^{t+\varepsilon} - v^t}{\varepsilon} = -B^{-1}\big(S(v^t)\big)$$

$$\Downarrow$$

$$v^{t+\varepsilon} - v^t = -\varepsilon \cdot B^{-1}\big(S(v^t)\big)$$

$$\Downarrow$$

$$v^{t+\varepsilon} = v^t - \varepsilon \cdot B^{-1}\big(S(v^t)\big)$$

$$\Downarrow$$

$$v^{t+\varepsilon} = B^{-1}(B(v^t))v^t - \varepsilon \cdot B^{-1}\big(S(v^t)\big)$$

$$\Downarrow$$

$$v^{t+\varepsilon} = B^{-1}\big((B - \varepsilon \cdot S)(v^t)\big)$$

$$\frac{\partial v}{\partial t} = -B^{-1}\big(S(v)\big)$$

Gradient Descent Revisited

Time-stepping as before, we get:

Implicit:

$$\frac{v^{t+\varepsilon} - v^t}{\varepsilon} = -B^{-1}\big(S(v^{\boxed{t+\varepsilon}})\big)$$
$$\Downarrow$$
$$v^{t+\varepsilon} - v^t = -\varepsilon \cdot B^{-1}\big(S(v^{t+\varepsilon})\big)$$
$$\Downarrow$$
$$v^{t+\varepsilon} + \varepsilon \cdot B^{-1}\big(S(v^{t+\varepsilon})\big) = v^t$$
$$\Downarrow$$
$$B(v^{t+\varepsilon}) + \varepsilon \cdot S(v^{t+\varepsilon}) = B(v^t)$$
$$\Downarrow$$
$$(B + \varepsilon \cdot S)(v^{t+\varepsilon}) = B(v^t)$$
$$\Downarrow$$
$$v^{t+\varepsilon} = (B + \varepsilon \cdot S)^{-1}\big(B(v^t)\big)$$

$$\frac{\partial v}{\partial t} = -B^{-1}\big(S(v)\big)$$

Gradient Descent Revisited

Time-stepping as before, we get:

Explicit:

$$v^{t+\varepsilon} = B^{-1}\big((B - \varepsilon \cdot S)(v^t)\big)$$

Implicit:

$$v^{t+\varepsilon} = (B + \varepsilon \cdot S)^{-1}\big(B(v^t)\big)$$

Expressing the gradient descent w.r.t. a basis $\{v_1, \ldots, v_n\}$ gives:

Explicit:

$$\mathbf{v}^{t+\varepsilon} = \mathbf{B}^{-1}\big((\mathbf{B} - \varepsilon \cdot \mathbf{S})(\mathbf{v}^t)\big)$$

Implicit:

$$\mathbf{v}^{t+\varepsilon} = (\mathbf{B} + \varepsilon \cdot \mathbf{S})^{-1}\big(\mathbf{B}(\mathbf{v}^t)\big)$$

$$\frac{\partial v}{\partial t} = -B^{-1}\big(S(v)\big)$$

Gradient Descent Revisited

Now both the implicit and explicit expressions for the PDE require solving a sparse symmetric positive-definite system.

Time-st…

Explic…

$$v^{t+\varepsilon} = B^{-1}\big((B - \varepsilon \cdot S)(v^t)\big)$$

Implicit:

$$v^{t+\varepsilon} = (B + \varepsilon \cdot S)^{-1}\big(B(v^t)\big)$$

Expressing the gradient descent w.r.t. a basis $\{v_1, \ldots, v_n\}$ gives:

Explicit:

$$\mathbf{v}^{t+\varepsilon} = \mathbf{B}^{-1}\big((\mathbf{B} - \varepsilon \cdot \mathbf{S})(\mathbf{v}^t)\big)$$

Implicit:

$$\mathbf{v}^{t+\varepsilon} = (\mathbf{B} + \varepsilon \cdot \mathbf{S})^{-1}\big(\mathbf{B}(\mathbf{v}^t)\big)$$

Gradient Descent Revisited

Time-st

Expli

Now both the implicit and explicit expressions for the PDE require solving a sparse symmetric positive-definite system.

In the context of geometry processing:

- The vector space is $V = \text{Span}(\phi_1, \dots, \phi_{|\mathcal{V}|})$
- The inner-product on V is defined by the mass
- The symmetric positive semi-definite bilinear form is the stiffness

Ex

m

Using mass *matrix lumping* the matrix $\mathbf{M}$ can be approximated by a diagonal matrix, which is trivial to solve (so the explicit temporal discretization is still easier to time-step).

Explicit:

$$\mathbf{v}^{t+\varepsilon} = \mathbf{M}^{-1}\left((\mathbf{M} - \varepsilon \cdot \mathbf{S})(\mathbf{v}^t)\right)$$

Implicit:

$$\mathbf{v}^{t+\varepsilon} = (\mathbf{M} + \varepsilon \cdot \mathbf{S})^{-1}\left(\mathbf{M}(\mathbf{v}^t)\right)$$

Gradient Descent Revisited

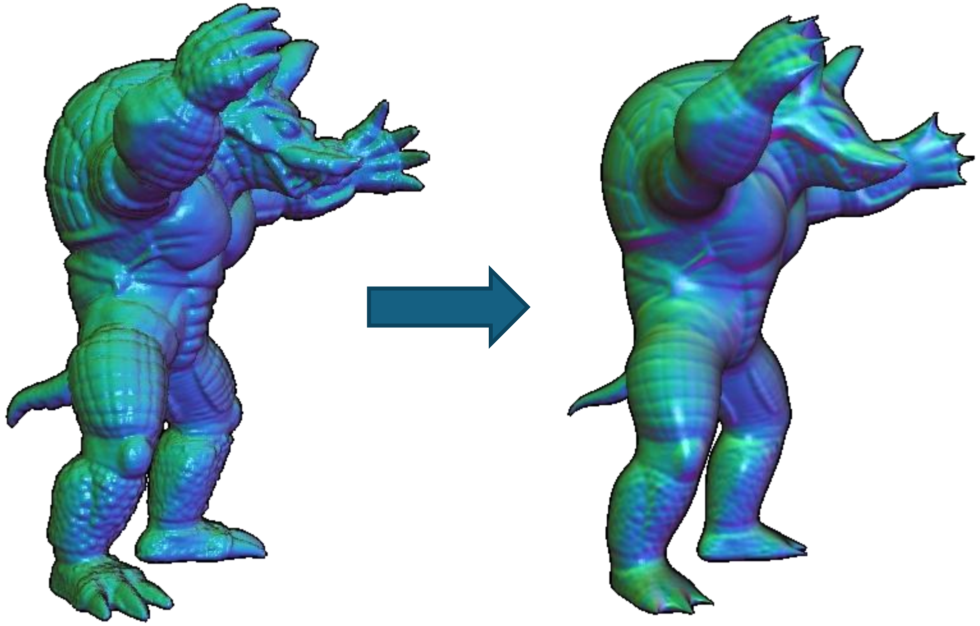
$$\mathbf{v}^{t+\varepsilon} = \mathbf{M}^{-1}\big((\mathbf{M} - \varepsilon \cdot \mathbf{S})(\mathbf{v}^t)\big)$$
$$\mathbf{v}^{t+\varepsilon} = (\mathbf{M} + \varepsilon \cdot \mathbf{S})^{-1}\big(\mathbf{M}(\mathbf{v}^t)\big)$$

Time-stepping the PDE we progressively smooth signals on the surface.

We can apply this to the x-, y-, and z-coordinates of the embedding, to smooth the geometry.

However, when we do this, the mass and stiffness matrices ($\mathbf{M}$ and $\mathbf{S}$) will change.

This results in a non-linear PDE called *mean curvature flow*.



Gradient Descent Revisited

Unless you start with a "nice" shape the flow will form *neck-pinch singularities*.

Time-stepping the PDE we progressively smooth signals on the surface.

We can apply this to the x-, y-, and z-coordinates of the embedding, to smooth the geometry.

However, when we do this, the mass and stiffness matrices ($\mathbf{M}$ and $\mathbf{S}$) will change.

This results in a non-linear PDE called *mean curvature flow*.