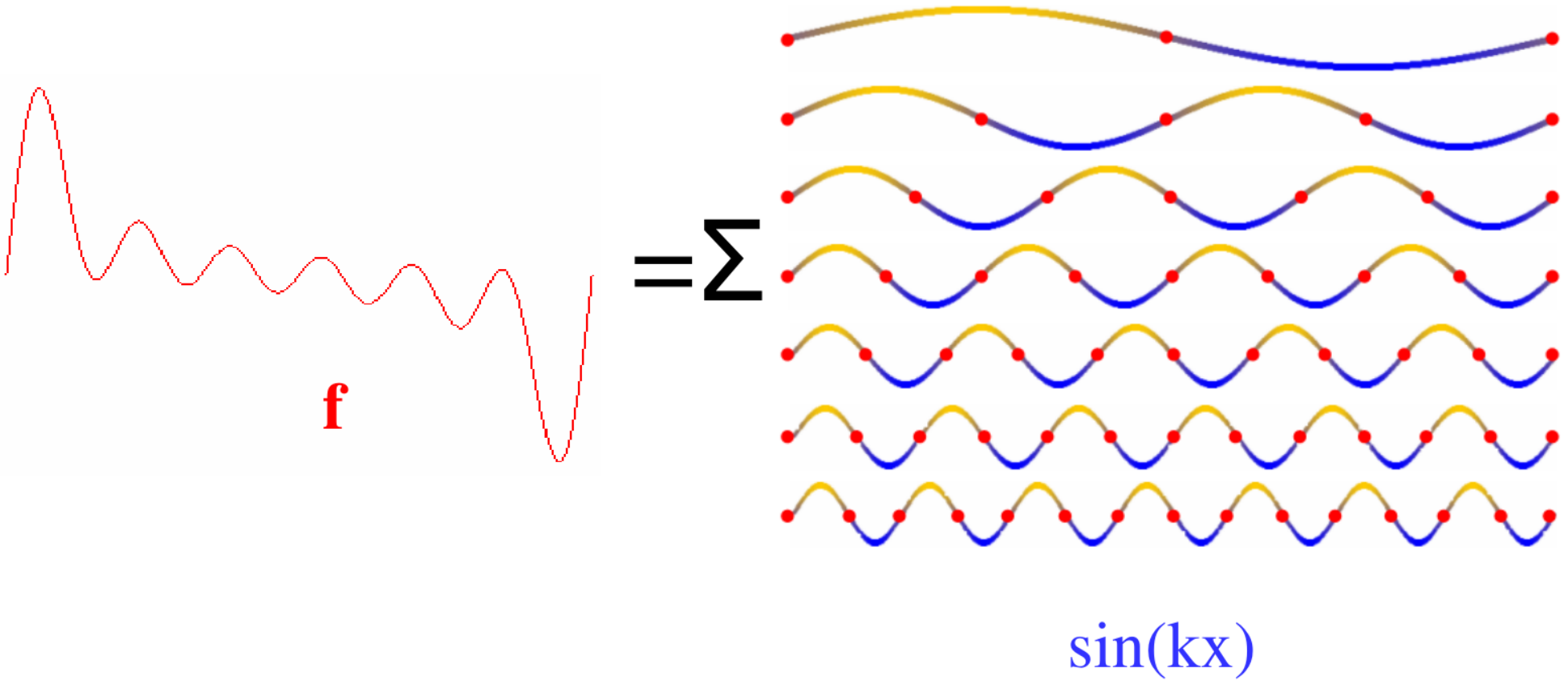
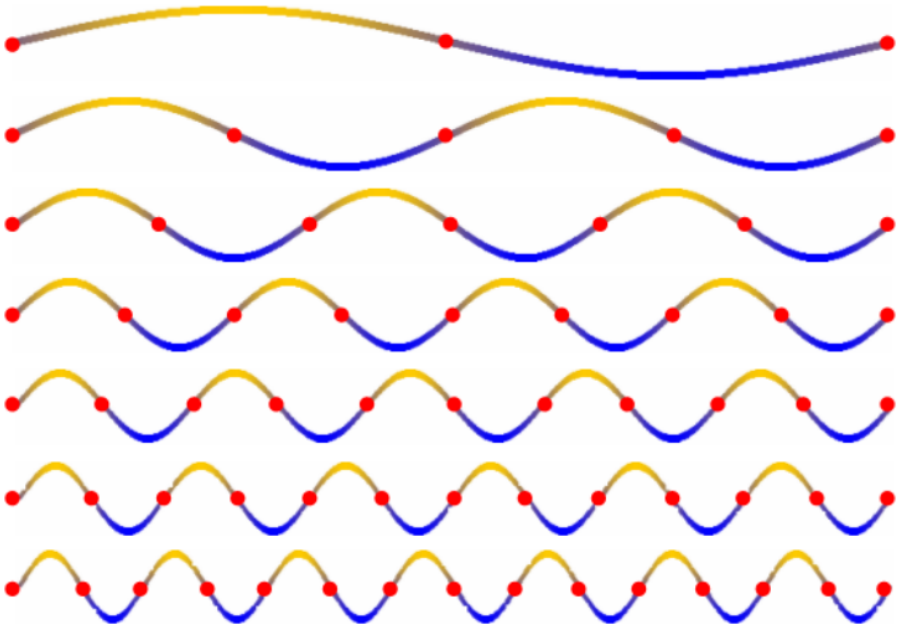
Spectral geometry on triangle meshes

Harmonics and spectral filtering



Harmonics and spectral filtering

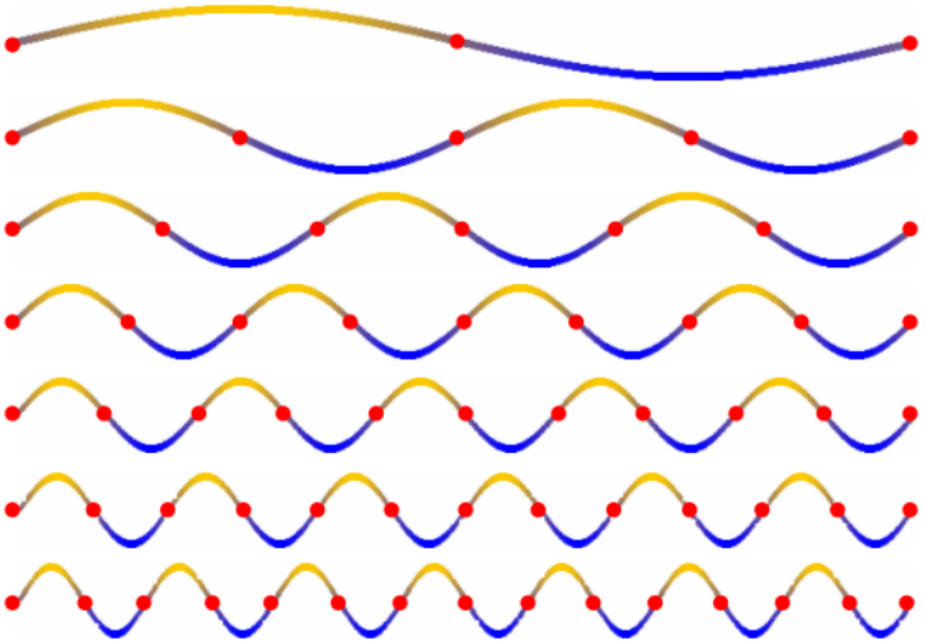


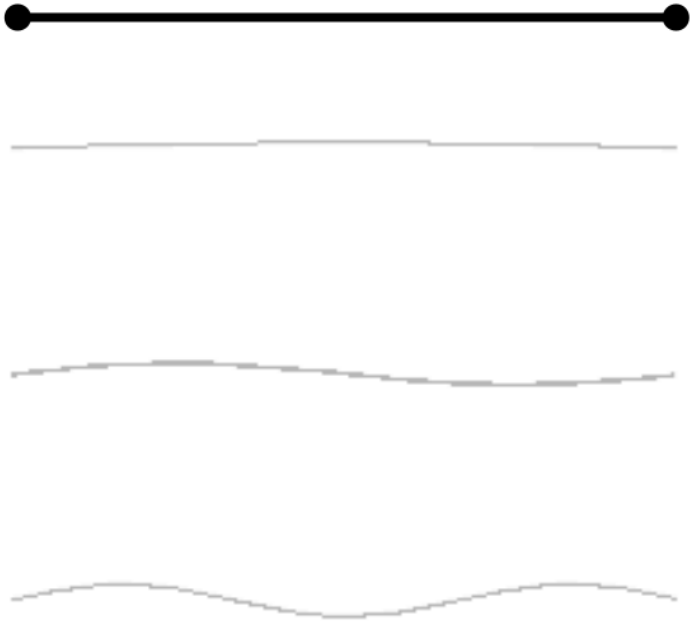
sin(kx)

on

?

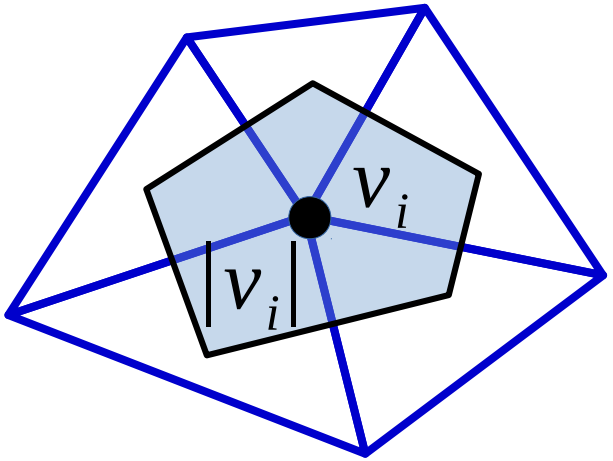
Harmonics and spectral filtering

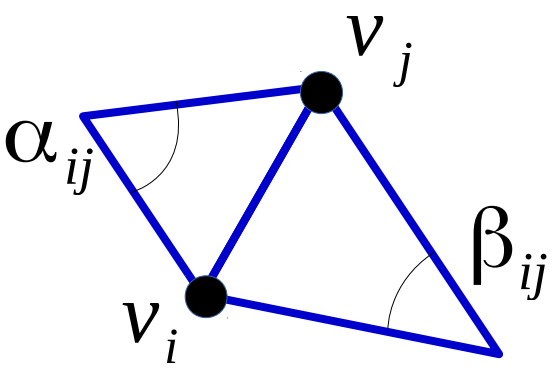




Strings harmonics = eigenvectors of unidimensional Laplacien

Reminder : Laplacian of scalar functions





$$\nabla^2 f(v_i) = \frac{1}{|v_i|} \sum_{e_{ij}} \frac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} (f(v_j) - f(v_i))$$

Note : You may see the version without 1/|v_i| here and there. Once again, the version without is the **integrated** operator (integrated over the area around vertex v_i), and the version with is the point-wise operator.

Spectral decomposition

Takes scalars defined on vertices, computes the Laplacian at each vertex

$$L \cdot \begin{bmatrix} f(0) \\ f(1) \\ . \\ . \\ f(|V|-1) \end{bmatrix} = \begin{bmatrix} \nabla^2 f(0) \\ \nabla^2 f(1) \\ . \\ . \\ \nabla^2 f(|V|-1) \end{bmatrix}$$

$$\in \mathbb{R}^{|V|\times 1} \qquad \in \mathbb{R}^{|V|\times 1}$$

$$\begin{cases} L(i,j) = \dfrac{1}{|v_i|} \dfrac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} \\ L(i,i) = -\sum_{e_{ij}} L(i,j) \end{cases}$$

Spectral decomposition

Takes scalars defined on vertices, computes the Laplacian at each vertex

$$L \cdot \begin{bmatrix} f(0) \\ f(1) \\ . \\ . \\ f(|V|-1) \end{bmatrix} = \begin{bmatrix} \nabla^2 f(0) \\ \nabla^2 f(1) \\ . \\ . \\ \nabla^2 f(|V|-1) \end{bmatrix}$$

$$\in \mathbb{R}^{|V|\times 1} \qquad \in \mathbb{R}^{|V|\times 1}$$

$$\begin{cases} L(i,j) = \dfrac{1}{|v_i|} \dfrac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} \\ L(i,i) = -\sum_{e_{ij}} L(i,j) \end{cases}$$

Eigenvectors of Laplacian shoud be eigenvectors of L

Spectral decomposition

Takes scalars defined on vertices, computes the Laplacian at each vertex

$$L \cdot \begin{bmatrix} f(0) \\ f(1) \\ \cdot \\ \cdot \\ f(|V|-1) \end{bmatrix} = \begin{bmatrix} \nabla^2 f(0) \\ \nabla^2 f(1) \\ \cdot \\ \cdot \\ \nabla^2 f(|V|-1) \end{bmatrix}$$

$$\in \mathbb{R}^{|V|\times 1} \qquad \in \mathbb{R}^{|V|\times 1}$$

$$\begin{cases} L(i,j) = \dfrac{1}{|v_i|} \dfrac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} \\ L(i,i) = -\sum_{e_{ij}} L(i,j) \end{cases}$$

PROBLEM ! It is not symmetric : L(i,j) not = L(j,i)

→ **Eigenvectors of L are not orthogonal**

Spectral decomposition

$$L = A^{-1} . L_C$$

Point-wise Laplacian

$$\begin{cases} L(i,j) = \dfrac{1}{|v_i|} \dfrac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} \\ L(i,i) = -\sum_{e_{ij}} L(i,j) \end{cases}$$

« Integrated » Laplacian :

$$\begin{cases} L_C(i,j) = \dfrac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} \\ L_C(i,i) = -\sum_{e_{ij}} L_C(i,j) \end{cases}$$

Diagonal mass matrix :

$$A(i,i) = |v_i|$$

Spectral decomposition

« General » eigenvectors of : $-L_C . \psi_i = \lambda_i A . \psi_i$

Low freq → High freq

Pseudo-orthogonality : $\psi_i^T . A . \psi_j = \delta_i^j$ (instead of $\psi_i^T . \psi_j = \delta_i^j$)

C++ : arpack++ (used for the examples made here) , Eigen3 with Spectra

An orthogonal basis

« General » eigenvectors of : $-L_C . \psi_i = \lambda_i A . \psi_i$

Pseudo-orthogonality : $\psi_i^T . A . \psi_j = \delta_i^j$

$$\bar{\psi}_i := \sqrt{A} . \psi_i$$

$$\bar{\psi}_i^T . \bar{\psi}_j = \psi_i^T . \sqrt{A}^T . \sqrt{A} . \psi_j = \psi_i^T . A . \psi_j = \delta_j^i$$

$\{\bar{\psi}_i\}_i$ is a good choice for decomposition : It is an orthonormal basis.

Decomposition

Given a function f on the vertices

$F_i := \bar{\psi}_i^T . f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)$ is its i^th frequency.

f can be recovered from its frequencies (inverse transform) :

$f = \sum_i F_i \bar{\psi}_i$

$$\begin{bmatrix} f \end{bmatrix} = \underbrace{\begin{bmatrix} \bar{\psi}_0 \, \bar{\psi}_1 .. \bar{\psi}_{n-1} \end{bmatrix}}_{B^{-1}} \cdot \underbrace{\begin{bmatrix} \bar{\psi}_0^T \\ \bar{\psi}_1^T \\ . \\ . \\ \bar{\psi}_{n-1}^T \end{bmatrix}}_{B} \cdot \begin{bmatrix} f \end{bmatrix}$$

Filtering

Given a function f on the vertices

$F_i := \bar{\psi}_i^T . f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)$ is its i^th frequency.

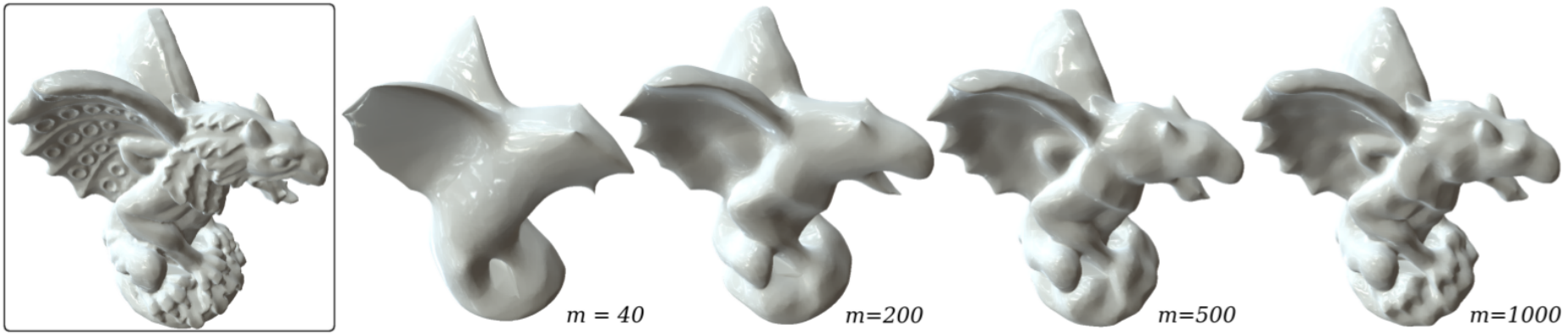
Filter h can be applied on the frequencies: $h \circ f = \sum_i h(F_i) \bar{\psi}_i$

Filtering

Given a function f on the vertices

$F_i := \bar{\psi}_i^T . f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)$ is its i^th frequency.

Filter h can be applied on the frequencies: $h \circ f = \sum_i h(F_i) \bar{\psi}_i$

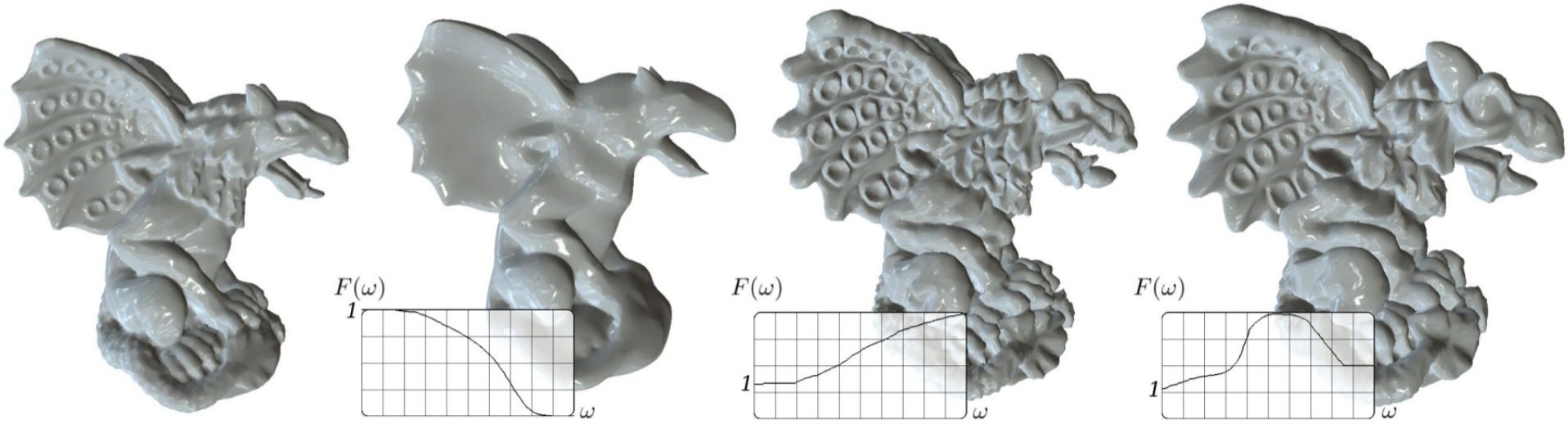


Filtering

Given a function f on the vertices

$F_i := \bar{\psi}_i^T . f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)$ is its i^th frequency.

Filter h can be applied on the frequencies: $h \circ f = \sum_i h(F_i) \bar{\psi}_i$



Filtering

Given a function f on the vertices

$F_i := \bar{\psi}_i^T . f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)$ is its i^th frequency.

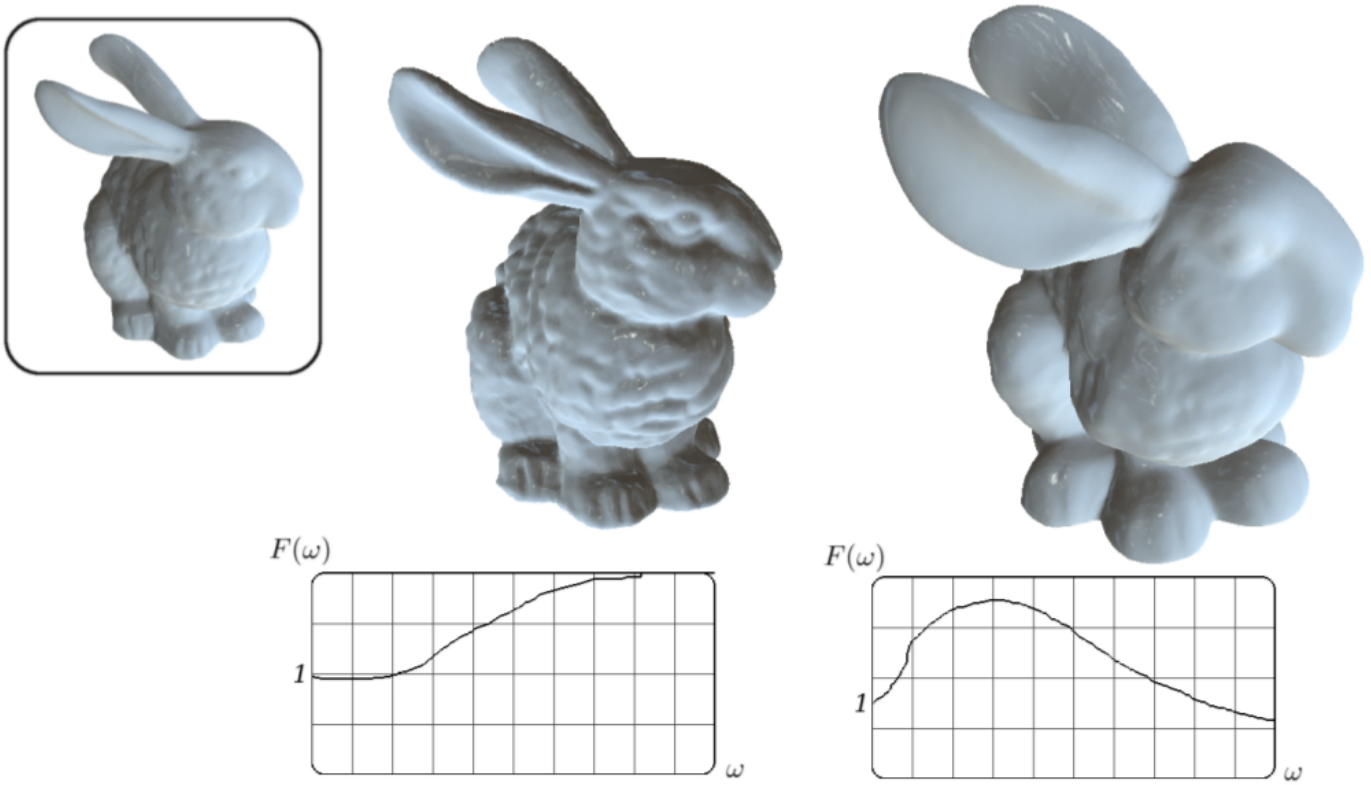
Filter h can be applied on the frequencies: $h \circ f = \sum_i h(F_i) \bar{\psi}_i$

Filtering

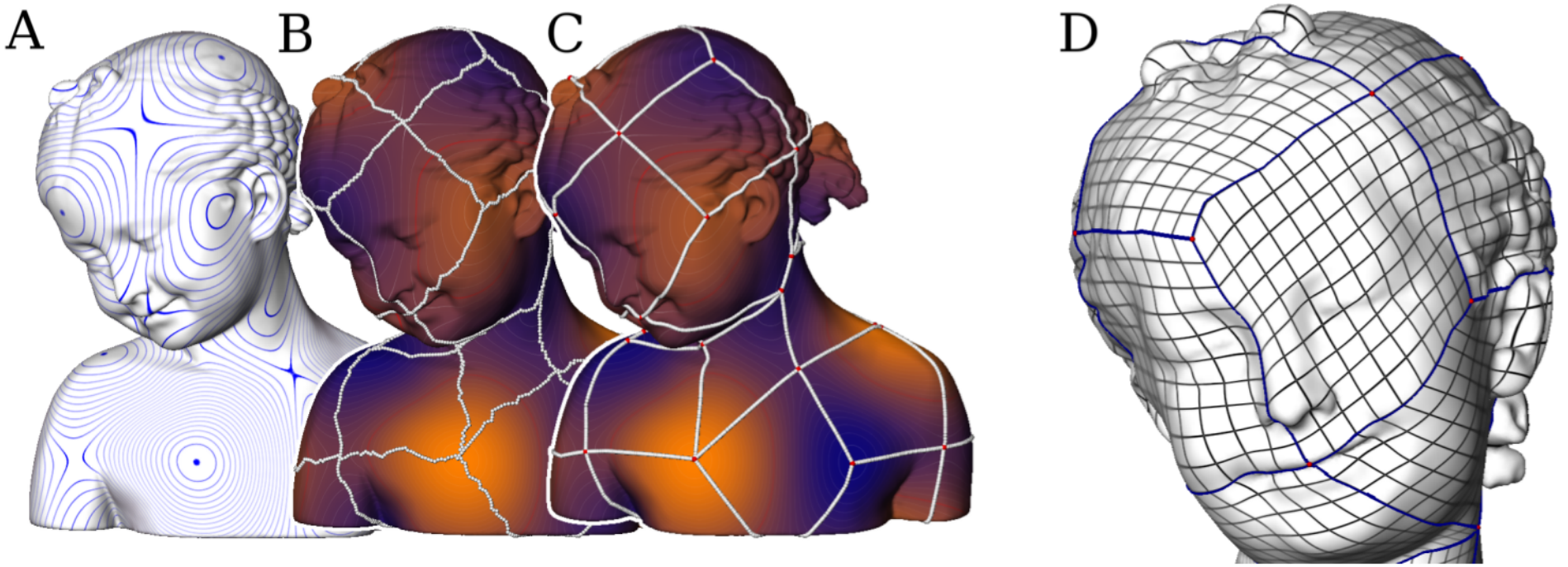
Given a function f on the vertices

$F_i := \bar{\psi}_i^T . f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)$ is its i^th frequency.

Filter h can be applied on the frequencies: $h \circ f = \sum_i h(F_i) \bar{\psi}_i$



Quad meshing



Shape retrieval

Heat diffusion

$\{\psi_i\}_i$ is a good basis for heat diffusion :

$$\begin{cases} \partial_t u(x,t) = \nabla_x^2 u(x,t) \\ u(x,0) = u_0(x) \end{cases}$$

$$u(x,t) = \sum_i \alpha_i(t)\psi_i(x) \quad \text{(decompose solution on basis)}$$

$$\partial_t u(x,t) = \nabla_x^2 u(x,t) \longrightarrow \sum_i \dot{\alpha}_i(t)\psi_i = \sum_i -\lambda_i \alpha_i(t)\psi_i$$

$$\dot{\alpha}_i(t) + \lambda_i \alpha_i(t) = 0 \longrightarrow \alpha_i(t) = \alpha_i(0)\exp(-\lambda_i t)$$

and $\alpha_i(0) = \int_x \psi_i(x) u_0(x)\, dx$

$$\longrightarrow u(k,t) = \int_x \sum_i \psi_i(x)\psi_i(k)\exp(-\lambda_i t) u_0(x) \quad \text{(in the continuous setting)}$$

$\longrightarrow$ Closed-form solution

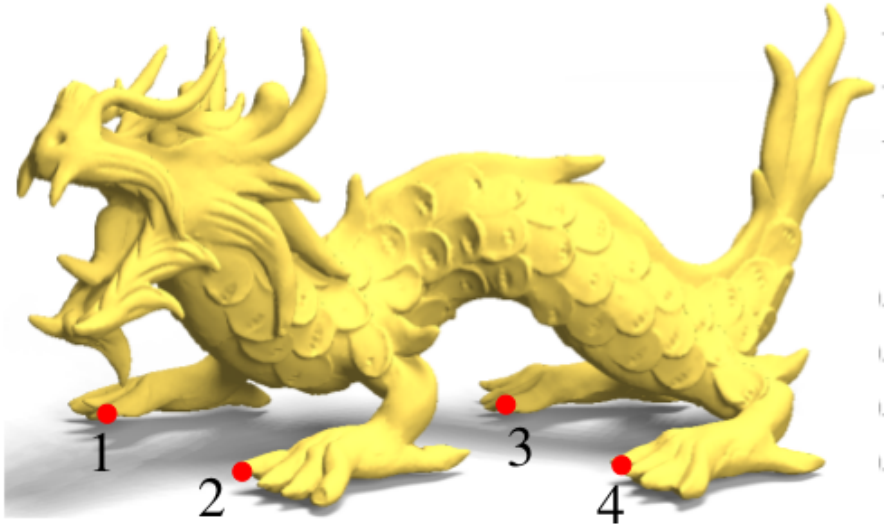
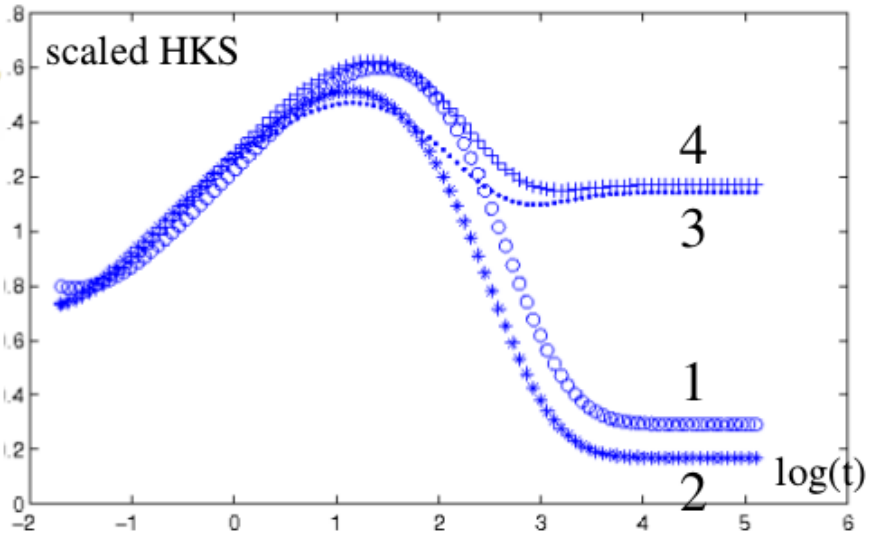
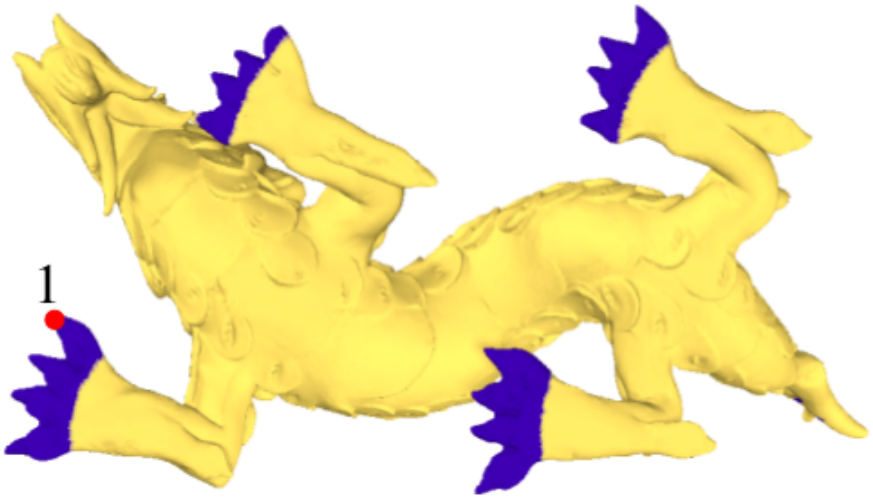
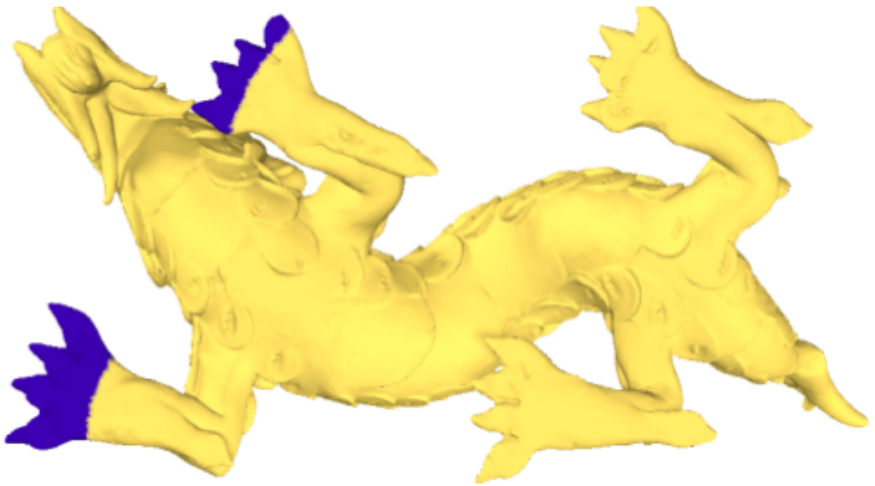
Heat diffusion

$h_t(j,k) = \sum_i \psi_i(j)\,\psi_i(k)\exp(-\lambda_i t)$: heat kernel at (j,k)

$\{h_t(j,j)\}_t$: multi-scale signature of vertex j

t = 1

t = 100

t = 10 000

Heat kernel signature