Spectral geometry on triangle meshes
Harmonics and spectral filtering

\[ f = \sum \sin(kx) \]
Harmonics and spectral filtering on $\sin(kx)$
Harmonics and spectral filtering

Strings harmonics = eigenvectors of unidimensional Laplacian
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

\[
L \cdot \begin{bmatrix}
  f(0) \\
  f(1) \\
  \vdots \\
  f(|V| - 1)
\end{bmatrix} = \begin{bmatrix}
  \nabla^2 f(0) \\
  \nabla^2 f(1) \\
  \vdots \\
  \nabla^2 f(|V| - 1)
\end{bmatrix}
\in \mathbb{R}^{|V| \times 1} \quad \in \mathbb{R}^{|V| \times 1}
\]

\[
L(i, j) = \frac{1}{|v_i|} \left( \cot(\alpha_{ij}) + \cot(\beta_{ij}) \right)
\]
\[
L(i, i) = -\sum_{e_{ij}} L(i, j)
\]
Spectral decomposition

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

\[
L . \begin{bmatrix}
  f(0) \\
  f(1) \\
  \vdots \\
  f(|V|-1)
\end{bmatrix} = \begin{bmatrix}
  \nabla^2 f(0) \\
  \nabla^2 f(1) \\
  \vdots \\
  \nabla^2 f(|V|-1)
\end{bmatrix}
\in \mathbb{R}^{|V| \times 1}
\in \mathbb{R}^{|V| \times 1}
\]

Eigenvectors of Laplacian should be eigenvectors of L
Spectral decomposition

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

\[
L \cdot \begin{bmatrix}
  f(0) \\
  f(1) \\
  \vdots \\
  f(|V| - 1)
\end{bmatrix} = \begin{bmatrix}
  \nabla^2 f(0) \\
  \nabla^2 f(1) \\
  \vdots \\
  \nabla^2 f(|V| - 1)
\end{bmatrix}
\in \mathbb{R}^{|V| \times 1}
\]

\[
\in \mathbb{R}^{|V| \times 1}
\]

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

\[
\begin{align*}
L(i,j) &= \frac{1}{|v_i|} \cot(\alpha_{ij}) + \cot(\beta_{ij}) \\
L(i,i) &= -\sum_{e_{ij}} L(i,j)
\end{align*}
\]

Eigenvectors of \( L \) are not orthogonal
Spectral decomposition

\[ L = A^{-1} \cdot L_C \]

Point-wise Laplacian

\[ L(i, j) = \frac{1}{|v_i|} \frac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} \]

\[ L(i, i) = -\sum_{e_{ij}} L(i, j) \]

« Integrated » Laplacian:

\[ L_C(i, j) = \frac{\cot(\alpha_{ij}) + \cot(\beta_{ij})}{2} \]

\[ L_C(i, i) = -\sum_{e_{ij}} L_C(i, j) \]

Diagonal mass matrix:

\[ A(i, i) = |v_i| \]
Spectral decomposition

« General » eigenvectors of: \(- L_C \cdot \psi_i = \lambda_i A \cdot \psi_i\)

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

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

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

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

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

$\bar{\psi}_i^T \cdot \bar{\psi}_j = \psi_i^T \cdot \sqrt{A^T} \cdot \sqrt{A} \cdot \psi_j = \psi_i^T \cdot A \cdot \psi_j = \delta_i^j$

$\{ \bar{\psi}_i \}_{i}$ is a good choice for decomposition: It is an orthonormal basis. (choice seen in related works, not the most obvious, see next)
Decomposition

Given a function \( f \) on the vertices

\[
F_i := \bar{\psi}_i^T \cdot f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j) \quad \text{is its } i^\text{th} \text{ frequency.}
\]

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

\[
f = \sum_i F_i \bar{\psi}_i
\]

\[
\begin{bmatrix}
    f
\end{bmatrix} =
\begin{bmatrix}
    \bar{\psi}_0 & \bar{\psi}_1 & \cdots & \bar{\psi}_{n-1}
\end{bmatrix}
\begin{bmatrix}
    \bar{\psi}_0^T \\
    \bar{\psi}_1^T \\
    \vdots \\
    \bar{\psi}_{n-1}^T
\end{bmatrix}
\cdot
\begin{bmatrix}
    \bar{\psi}_0 \\
    \bar{\psi}_1 \\
    \vdots \\
    \bar{\psi}_{n-1}
\end{bmatrix}
\cdot
\begin{bmatrix}
    f
\end{bmatrix}
\]
Decomposition (probably more correct)

Given a function $f$ on the vertices

$$F_i := \langle \psi_i | f \rangle = \int_x \psi_i(x) f(x) \, dx = \psi_i^T . A . f$$

is its $i$th frequency.

$f$ can be recovered from its frequencies (inverse transform): $f = \sum_i F_i \psi_i$

$\left[ \begin{array}{c} f \\ \psi \end{array} \right] = \psi^T . A . \left[ \begin{array}{c} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{n-1} \end{array} \right]$

Dot product with Basis

$\left[ \begin{array}{c} \psi_0^T \\ \psi_1^T \\ \vdots \\ \psi_{n-1}^T \end{array} \right] . A . \left[ \begin{array}{c} f \end{array} \right]$
Filtering

Given a function \( f \) on the vertices

\[
F_i := \bar{\psi}_i^T \cdot f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)
\]

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

$$F_i := \bar{\psi}_i^T \cdot f = \sum_{v_j} \bar{\psi}_i(v_j) f(v_j)$$

is its $i$\textsuperscript{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

$$F_i := \overline{\psi}_i^T \cdot f = \sum_{v_j} \overline{\psi}_i(v_j) f(v_j)$$

is its $i$\textsuperscript{th} frequency.

Filter $h$ can be applied on the frequencies:

$$h \circ f = \sum_i h(F_i) \overline{\psi}_i$$
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 $F_i := \overline{\psi}_i^T \cdot f = \sum_{v_j} \overline{\psi}_i(v_j) f(v_j)$ is its $i^{th}$ frequency.

Filter $h$ can be applied on the frequencies: $h \circ f = \sum_i h(F_i) \overline{\psi}_i$
Quad meshing
Shape retrieval
Heat diffusion

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

\[
\begin{aligned}
\frac{\partial}{\partial t} u(x, t) &= \nabla^2_x u(x, t) \\
\left. u \right|_{t=0} &= u_0(x)
\end{aligned}
\]

\[
\alpha_i(0) = \int_x \psi_i(x) u_0(x) \, dx
\]

and

\[
\alpha_i(t) = \alpha_i(0) \exp(-\lambda_i t)
\]

Closed-form solution

\[
\rightarrow u(k, t) = \int_x \sum_i \psi_i(x) \psi_i(k) \exp(-\lambda_i t) u_0(x)
\]

\( \alpha_i(0) \) decomposes solution on basis

\( \sum_i \dot{\alpha}_i(t) \psi_i = \sum_i -\lambda_i \alpha_i(t) \psi_i \)
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
Heat kernel signature
Geodesics in Heat

Link with:

- Spectral properties
- Physics (heat)

Different from front-propagation approaches

Geodesics in Heat

\[ \partial_t u = \nabla^2 u \]
\[ (u_t - u_0)/t = \nabla^2 u_t \]
\[ u_t - t \nabla^2 u_t = u_0 \]
\[ (id - t \nabla^2) u_t = u_0 \]

Linear system

Algorithm 1 The Heat Method

I. Integrate the heat flow \( \dot{u} = \Delta u \) for some fixed time \( t \).

Algorithm 1 The Heat Method

I. Integrate the heat flow $\dot{u} = \Delta u$ for some fixed time $t$.
II. Evaluate the vector field $X = -\nabla u / |\nabla u|$.

Geodesics in Heat

**Algorithm 1** The Heat Method

I. Integrate the heat flow $\dot{u} = \Delta u$ for some fixed time $t$.

II. Evaluate the vector field $X = -\nabla u / |\nabla u|$.  

III. Solve the Poisson equation $\Delta \phi = \nabla \cdot X$.  

Geodesics in Heat

Step I \( (id-t \nabla^2) u_t = u_0 \)

**Linear system:**
- can be prefactored indep of \( u_0 \)

Step II \( \vec{X} = - \nabla u_t / \| \nabla u_t \| \)

**straightforward**

Step III \( \nabla^2 \phi = \nabla \cdot \vec{X} \)

**Linear system:**
- can be prefactored indep of \( u_0 \)

Laplacian operators have been studied for general polygonal meshes and pointsets:

Geodesics in Heat: value of $t$?

[source]

$t_{\text{small}}$

(provides smoother approx of geodesics)

$t_{\text{big}}$

?