Spectral Geometry Processing

Misha Kazhdan

[Taubin, 1995] A Signal Processing Approach to Fair Surface Design
[Desbrun, *et al.*, 1999] Implicit Fairing of Arbitrary Meshes…
[Vallet and Levy, 2008] Spectral Geometry Processing with Manifold Harmonics
[Bhat *et al.*, 2008] Fourier Analysis of the 2D Screened Poisson Equation…
And much, much, much, more…

Outline

- Motivation
- Laplacian Spectrum
- Applications
- Conclusion

Motivation

Recall:

Given a signal, $f\colon [0,2\pi) \to \mathbb{R}$, we can write it out in terms of its *Fourier decomposition*:

$$f(\theta) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{f}}_k \frac{e^{ik\theta}}{\sqrt{2\pi}}$$

$\hat{\mathbf{f}}_k \in \mathbb{C}$ is the k-th *Fourier coefficients* of f.

Motivation

$$f(\theta) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{f}}_k \frac{e^{ik\theta}}{\sqrt{2\pi}}$$

<u>Frequency Decomposition</u>:

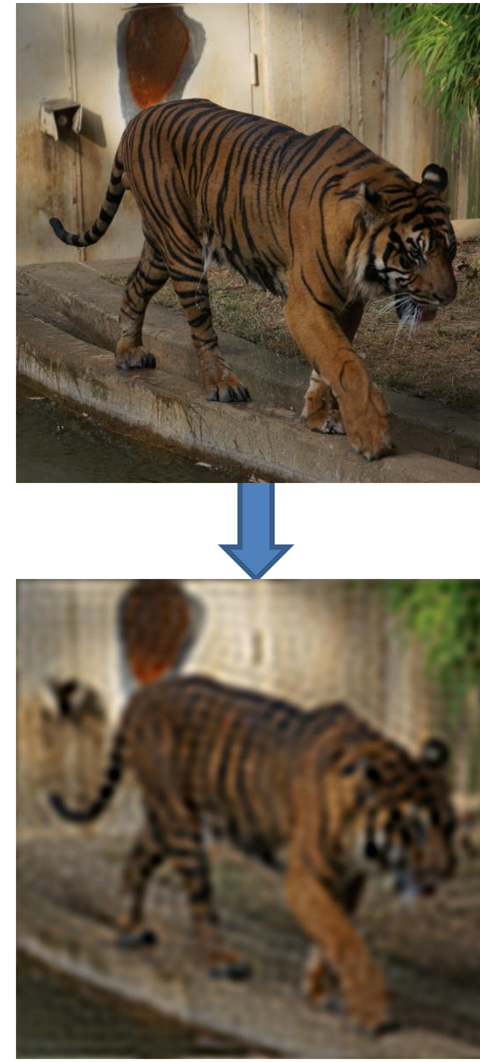
For smaller $N \in \mathbb{Z}$, the finite sum:

$$f^N(\theta) = \sum_{k=-N}^{N} \hat{\mathbf{f}}_k \frac{e^{ik\theta}}{\sqrt{2\pi}}$$

represents the lower frequency components of f.

Motivation

$$f(\theta) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{f}}_k \frac{e^{ik\theta}}{\sqrt{2\pi}}$$



<u>Filtering</u>:

By modulating the values of $\hat{\mathbf{f}}_k$ as a function of frequency, we can realize different signal filters:

$$\hat{\mathbf{f}}_k \leftarrow \begin{cases} \hat{\mathbf{f}}_k & \text{if } |k| < N \\ 0 & \text{otherwise} \end{cases}$$

Motivation

$$f(\theta) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{f}}_k \frac{e^{ik\theta}}{\sqrt{2\pi}}$$

<u>Filtering</u>:

By modulating the values of $\hat{\mathbf{f}}_k$ as a function of frequency, we can realize different signal filters:

$$\hat{\mathbf{f}}_k \leftarrow \hat{\mathbf{f}}_k e^{-k^2}$$

Motivation

$$f(\theta) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{f}}_k \frac{e^{ik\theta}}{\sqrt{2\pi}}$$

Filtering:

By modulating the values of $\hat{\mathbf{f}}_k$ as a function of frequency, we can realize different signal filters:

$$\hat{\mathbf{f}}_k \leftarrow \hat{\mathbf{f}}_k \left(2 - e^{-k^2}\right)$$

Motivation

<u>Goal</u>:

We would like to extend this type of processing to the context of signals defined on surfaces*:

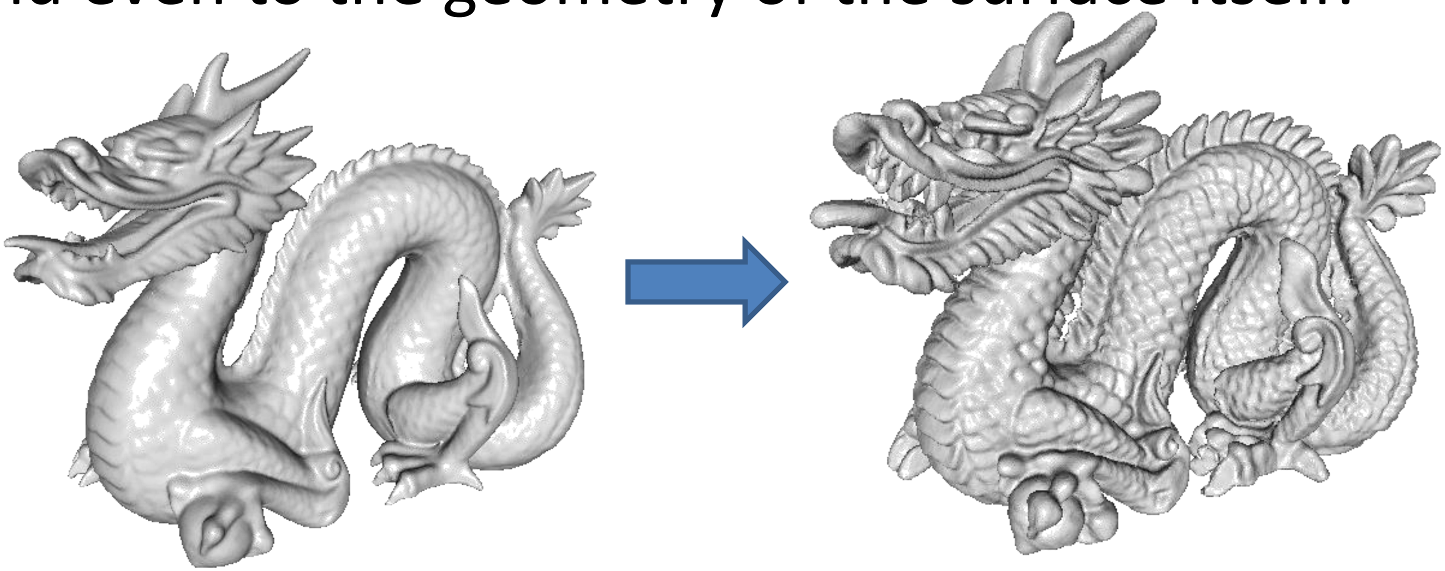
$$\hat{\mathbf{f}}_k \leftarrow \hat{\mathbf{f}}_k \left(2 - e^{-k^2}\right)$$

*For simplicity, we will assume all surfaces are w/o boundary.

Motivation

<u>Goal</u>:

We would like to extend this type of processing to the context of signals defined on surfaces* and even to the geometry of the surface itself:



$$\hat{\mathbf{f}}_k \leftarrow \hat{\mathbf{f}}_k \left(2 - e^{-k^2}\right)$$

*For simplicity, we will assume all surfaces are w/o boundary.

Motivation

$$f(\theta) = \sum_{k=-\infty}^{\infty} \hat{\mathbf{f}}_k \frac{e^{ik\theta}}{\sqrt{2\pi}}$$

<u>[WARNING]</u>:

– In Euclidean space we can use the FFT to obtain the Fourier decomposition efficiently.

– For signals on surfaces, this is more challenging.

Outline

- Motivation
- Laplacian Spectrum
  - Fourier ↔ Laplacian
  - FEM discretization
- Applications
- Conclusion

How do we obtain the Fourier decomposition?

Fourier $\leftrightarrow$ Laplacian

<u>Recall</u>:

In Euclidean space, the *Laplacian*, is the operator that takes a function and returns the sum of (unmixed) second partial derivatives:

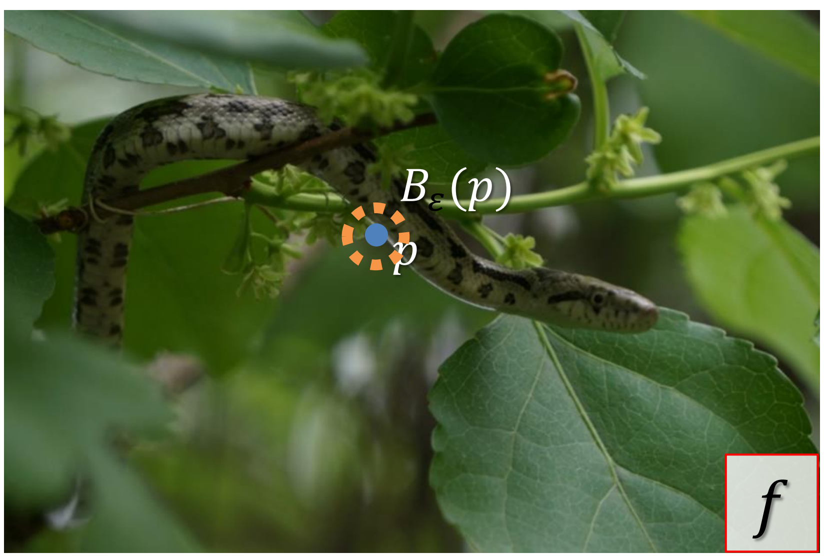
$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} + \cdots$$

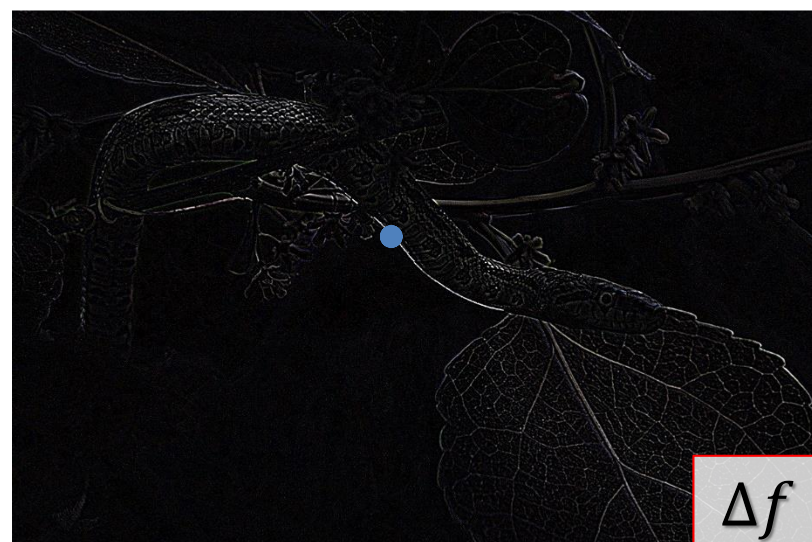
Fourier ↔ Laplacian

Informally:

The Laplacian gives the difference between the value at a point and the average in the vicinity:

$$\Delta f(p) \sim \lim_{\varepsilon \to 0} \frac{1}{\varepsilon^2}\left(\mathrm{Avg}_{B_\varepsilon(p)}(f) - f(p)\right)$$





Fourier ↔ Laplacian

<u>Note</u>:

The complex exponential $f(\theta)=\frac{e^{ik\theta}}{\sqrt{2\pi}}$ has Laplacian:

$$\frac{\partial^2}{\partial\theta^2}\left(\frac{e^{ik\theta}}{\sqrt{2\pi}}\right)$$

$$\Downarrow$$

$f(\theta)=\frac{e^{ik\theta}}{\sqrt{2\pi}}$ is an eigenfunction of the Laplacian with eigenvalue $-k^2$.

Fourier ↔ Laplacian

Note:

Similarly, $f(\theta,\phi) = \frac{e^{ik\theta}\cdot e^{il\phi}}{2\pi}$ has Laplacian:

$$\left(\frac{\partial^2}{\partial\theta^2} + \frac{\partial^2}{\partial\phi^2}\right)\left(\frac{e^{ik\theta}\cdot e^{il\phi}}{2\pi}\right) = -(k^2 + l^2)\cdot\left(\frac{e^{ik\theta}\cdot e^{il\phi}}{2\pi}\right)$$

$$\Downarrow$$

$f(\theta,\phi) = \frac{e^{ik\theta}\cdot e^{il\phi}}{2\pi}$ is an eigenfunction of the Laplacian with eigenvalue $-(k^2 + l^2)$.

Fourier ↔ Laplacian

<u>Approach</u>:

– Though we cannot compute the FFT for signals on general surfaces, we can define a Laplacian.

– To compute the Fourier decomposition of a signal, f, on a mesh we decompose f as the linear combination of eigenvectors of the Laplacian:

$$f(x) = \sum_{i=1}^{n} \hat{\mathbf{f}}_i \phi^i(x) \quad \text{with} \quad \Delta\phi^i = \lambda_i \phi^i.$$

This is called the *harmonic decomposition* of f.

Fourier $\leftrightarrow$ Laplacian

How do we know the eigenvectors of the Laplacian form a basis?

<u>Claims</u>:

1. The Laplacian is a symmetric operator.
2. The eigenvectors of a symmetric operator form an orthogonal basis (and have real eigenvalues).

Fourier ↔ Laplacian

Preliminaries:

- [Definition of the Laplacian]

$$\Delta f = \mathrm{div}(\nabla f)$$

- [Product Rule]

$$\mathrm{div}(f\vec{v}) = f\,\mathrm{div}(\vec{v}) + \langle \nabla f, \vec{v}\rangle$$

- [Inner Product on Functions]
  Given a surface $S \subset \mathbb{R}^3$:

$$\langle f, g\rangle_S = \int_S f(x)g(x)\,dx$$

- [Divergence Theorem*]

$$\int_S [\mathrm{div}(\vec{v})](p) = \int_{\partial S} \langle \vec{v}(s), \vec{n}(s)\rangle\,ds$$

Symmetry of The Laplacian

1. <u>The Laplacian is a symmetric operator</u>

Given a surface $S \subset \mathbb{R}^3$, we want to show that for any functions $f, g: S \to \mathbb{R}$ we have:

$$\langle \Delta f, g \rangle_S = \langle f, \Delta g \rangle_S$$

$$\Updownarrow$$

$$\int_S \Delta f \cdot g \, dx = \int_S f \cdot \Delta g \, dx$$

Symmetry of The Laplacian

<u>Proof</u>:

By the definition of the Laplacian:

$$\Delta f = \mathrm{div}(\nabla f)$$

$$\begin{aligned}
\langle \Delta f, g \rangle_S &= \int_S \Delta f \cdot g \, dx \\
&= \int_S \mathrm{div}(\nabla f) \cdot g \, dx \\
&= \int_S (\mathrm{div}(g \cdot \nabla f) - \langle \nabla f, \nabla g \rangle) \, dx \\
&= -\int_S \langle \nabla f, \nabla g \rangle \, dx
\end{aligned}$$

Symmetry of The Laplacian

Proof:

By the product rule:

$$\text{div}(f \cdot \vec{v}) = f \cdot \text{div}(\vec{v}) + \langle \nabla f, \vec{v} \rangle$$

$$\begin{aligned}\langle \Delta f, g \rangle_S &= \int_S \Delta f \cdot g \, dx \\ &= \int_S \text{div}(\nabla f) \cdot g \, dx \\ &= \int_S \left(\text{div}(g \cdot \nabla f) - \langle \nabla f, \nabla g \rangle\right) dx \\ &= -\int_S \langle \nabla f, \nabla g \rangle \, dx\end{aligned}$$

Symmetry of The Laplacian

Proof:

By the Divergence Theorem*:

$$\int_S [\mathrm{div}(\vec{v})](p) = \int_{\partial S} \langle \vec{v}(s), \vec{n}(s) \rangle \, ds = 0$$

$$\begin{aligned}
\langle \Delta f, g \rangle_S &= \int_S \Delta f \cdot g \, dx \\
&= \int_S \mathrm{div}(\nabla f) \cdot g \, dx \\
&= \int_S (\mathrm{div}(g \cdot \nabla f) - \langle \nabla f, \nabla g \rangle) \, dx \\
&= -\int_S \langle \nabla f, \nabla g \rangle \, dx
\end{aligned}$$

What happens in the discrete setting?

FEM Discretization

1. To enable computation, we restrict ourselves to a *finite-dimensional* space of functions, spanned by basis functions $\{B^i : S \to \mathbb{R}\}_{i=1}^{n}$.

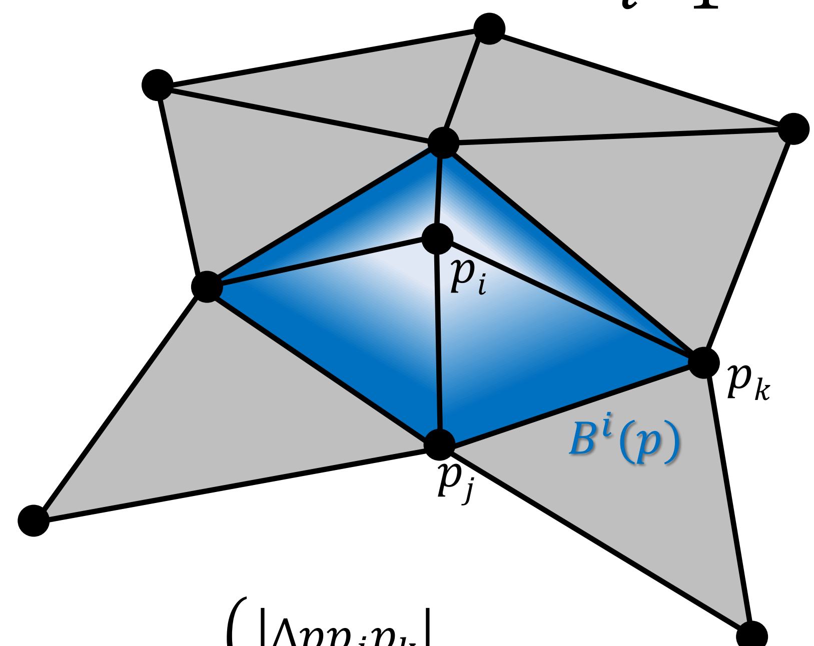
Often these are defined to be the "hat" functions centered at vertices.

– Piecewise linear

$\Rightarrow$ Gradients are constant within each triangle

– Interpolatory

$\Rightarrow B^i(p_j) = \delta_{ij}$



$$B^i(p) = \begin{cases} \dfrac{|\Delta p p_j p_k|}{|\Delta p_i p_j p_k|} & \text{if } p \in \Delta p_i p_j p_k \\ 0 & \text{otherwise} \end{cases}$$

FEM Discretization

1. To enable computation, we restrict ourselves to a *finite-dimensional* space of functions, spanned by basis functions $\{B^i : S \to \mathbb{R}\}_{i=1}^{n}$.

Having chosen a basis, we can think of a vector $\mathbf{f} \in \mathbb{R}^n$ as a "discrete" function:

$$\mathbf{f} \leftrightarrow f(p) = \sum_{i=1}^{n} \mathbf{f}_i B^i(p)$$

If we use the hat functions as a basis, then:

$$f(p_j) = \sum_{i=1}^{n} \mathbf{f}_i \, B^i(p_j)$$

FEM Discretization

1. To enable computation, we restrict ourselves to a *finite-dimensional* space of functions, spanned by basis functions $\left\{B^i: S \to \mathbb{R}\right\}_{i=1}^{n}$.

[WARNING]:

In general, given:

- $\mathcal{L}$: A continuous linear operator
- $\mathbf{f} \in \mathbb{R}^n \leftrightarrow f(p)$: A discrete function

The function $\mathcal{L}(f)$ will *not* be in the space of functions spanned by $\left\{B^i: S \to \mathbb{R}\right\}_{i=1}^{n}$.

FEM Discretization

1. To enable computation, we restrict ourselves to a *finite-dimensional* space of functions, spanned by basis functions $\left\{B^i : S \to \mathbb{R}\right\}_{i=1}^{n}$.
2. Given a continuous linear operator $\mathcal{L}$, we discretize the operator by *projecting*:

$$g = \mathcal{L}(f)$$
$$\Downarrow$$
$$\langle g, B^j \rangle_S = \langle \mathcal{L}(f), B^j \rangle_S \quad \forall j$$

FEM Discretization

$$\langle g, B^j \rangle_S = \langle \mathcal{L}(f), B^j \rangle_S \quad \forall j$$

Writing out the discrete functions:

$$g(p) = \sum_{i=1}^{n} \mathbf{g}_i B^i(p) \quad \text{and} \quad f(p) = \sum_{i=1}^{n} \mathbf{f}_i B^i(p)$$

$$\Downarrow$$

$$\sum_{i=1}^{n} \mathbf{g}_i \langle B^i, B^j \rangle_S = \sum_{i=1}^{n} \mathbf{f}_i \langle \mathcal{L}\left(B^i\right), B^j \rangle_S \quad \forall j$$

FEM Discretization

$$\sum_{i=1}^{n} \mathbf{g}_i \langle B^i, B^j \rangle_S = \sum_{i=1}^{n} \mathbf{f}_i \langle \mathcal{L}\left(B^i\right), B^j \rangle_S \quad \forall j$$

Setting $\mathbf{M}$ and $\mathbf{L}$ to be the matrices:

$$\mathbf{M}_{ij} = \langle B^i, B^j \rangle_S \quad \text{and} \quad \mathbf{L}_{ij} = \langle \mathcal{L}\left(B^i\right), B^j \rangle_S$$

$$\Downarrow$$

$$\sum_{i=1}^{n} \mathbf{M}_{ij} \mathbf{g}_i = \sum_{i=1}^{n} \mathbf{L}_{ij} \mathbf{f}_i \quad \forall j$$

$$\Downarrow$$

$$\mathbf{M}\mathbf{g} = \mathbf{L}\mathbf{f}$$

FEM Discretization

$$\mathbf{M}_{ij} = \langle B^i, B^j \rangle_S \quad \text{and} \quad \mathbf{L}_{ij} = \langle \mathcal{L}(B^i), B^j \rangle_S$$

Both the mass and stiffness matrices are symmetric and positive (semi)-definite.

When $\mathcal{L} = \Delta$, we have:

$$\mathbf{L}_{ij} = \langle \Delta B^i, B^j \rangle_S$$

Definition:

The matrix $\mathbf{M}$ is called the *mass matrix*.

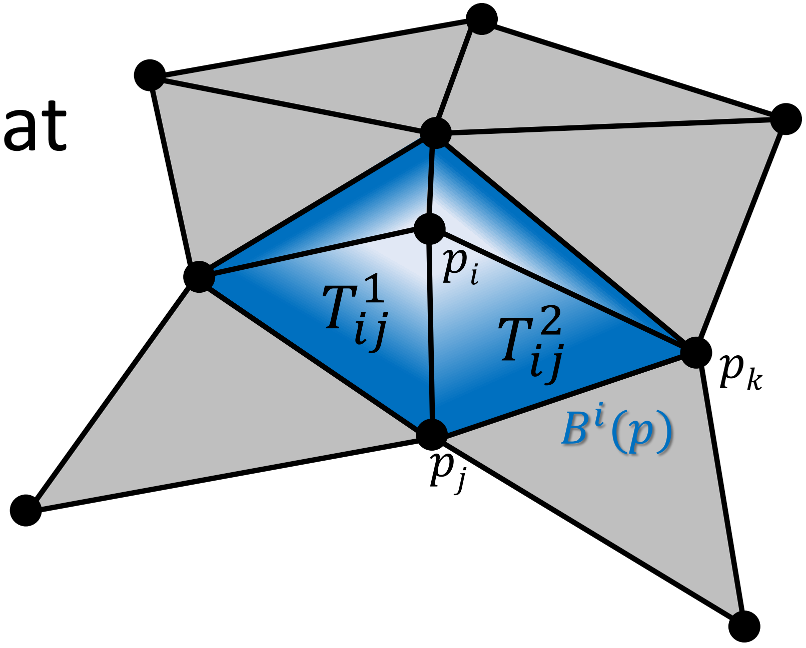
The matrix $\mathbf{S}$ is called the *stiffness matrix*.

FEM Discretization

$$\mathbf{M}_{ij} = \langle B^i, B^j \rangle_S \quad \text{and} \quad \mathbf{S}_{ij} = \langle \nabla B^i, \nabla B^j \rangle_S$$

Setting $\{B^i : S \to \mathbb{R}\}$ to the hat functions, the matrix $\mathbf{M}$ is:

$$\mathbf{M}_{ij} = \begin{cases} \dfrac{|T_{ij}^1| + |T_{ij}^2|}{12} & \text{if } j \in N(i) \\ \displaystyle\sum_{k \in N(i)} \mathbf{M}_{ik} & \text{if } i = j \end{cases}$$

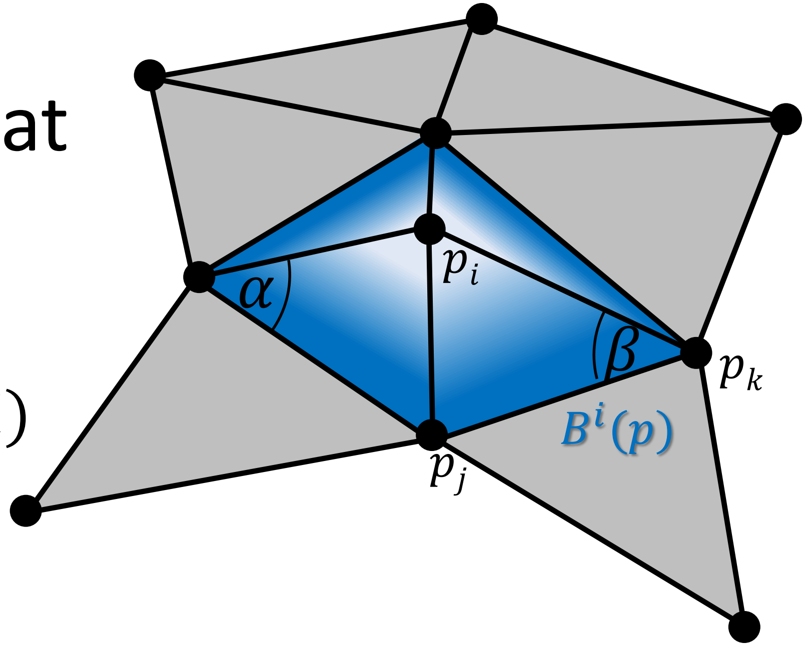


FEM Discretization

$$\mathbf{M}_{ij} = \langle B^i, B^j \rangle_S \quad \text{and} \quad \mathbf{S}_{ij} = \langle \nabla B^i, \nabla B^j \rangle_S$$

Setting $\{B^i : S \to \mathbb{R}\}$ to the hat functions, the matrix $\mathbf{L}$ is the cotangent-Laplacian:

$$\mathbf{S}_{ij} = \begin{cases} -(\cot\alpha + \cot\beta) & \text{if } j \in N(i) \\ -\sum_{k \in N(i)} \mathbf{S}_{ik} & \text{if } i = j \end{cases}$$

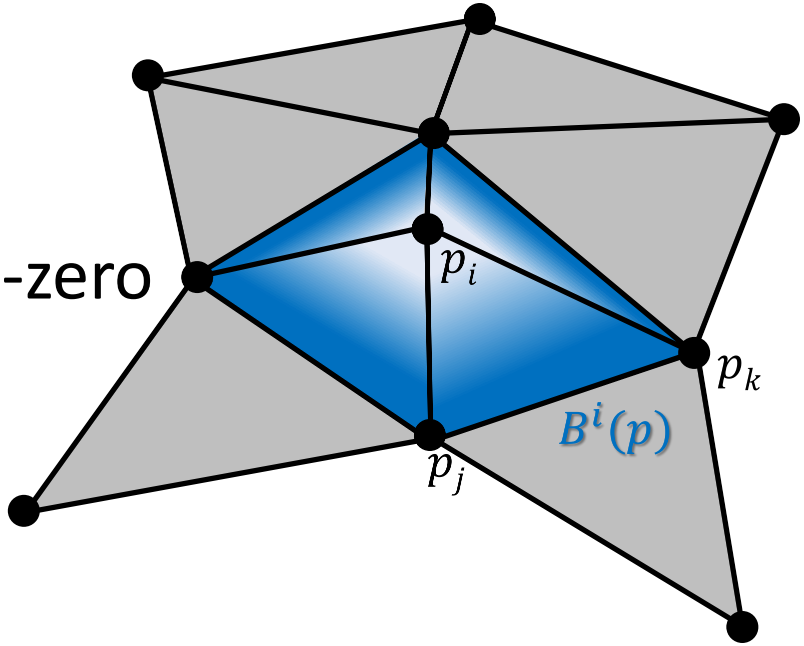


FEM Discretization

$$\mathbf{M}_{ij} = \begin{cases} \dfrac{\left|T_{ij}^1\right| + \left|T_{ij}^2\right|}{12} & \boxed{\text{if } j \in N(i)} \\ \displaystyle\sum_{k \in N(i)} \mathbf{M}_{ik} & \text{if } i = j \end{cases} \quad \text{and} \quad \mathbf{S}_{ij} = \begin{cases} -(\cot\alpha + \cot\beta) & \boxed{\text{if } j \in N(i)} \\ -\displaystyle\sum_{k \in N(i)} \mathbf{S}_{ik} & \text{if } i = j \end{cases}$$

Observations:

– [Sparsity]
Entry (i, j) can only be non-zero if vertex i and vertex j are neighbors in the mesh.



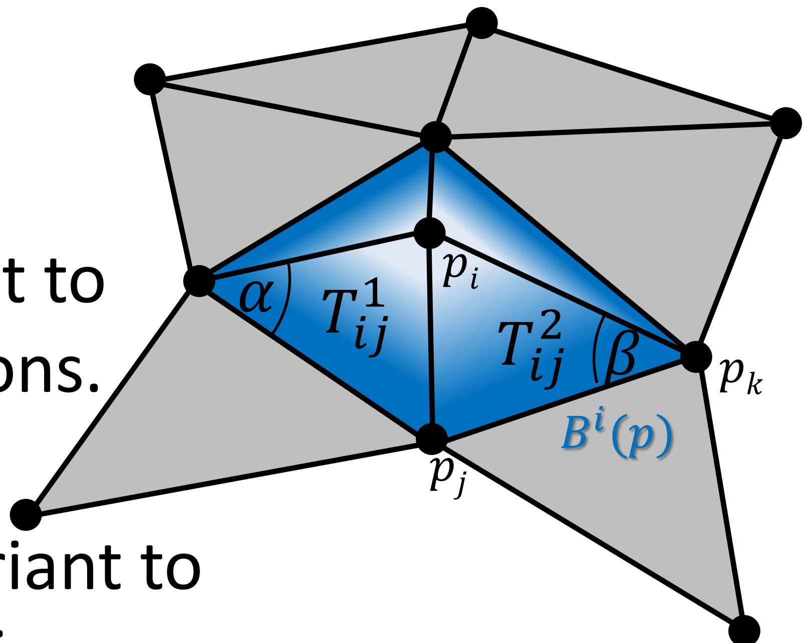
FEM Discretization

$$\mathbf{M}_{ij} = \begin{cases} \frac{|T_{ij}^1| + |T_{ij}^2|}{12} & \text{if } j \in N(i) \\ \sum_{k \in N(i)} \mathbf{M}_{ik} & \text{if } i = j \end{cases} \quad \text{and} \quad \mathbf{S}_{ij} = \begin{cases} -(\cot\alpha + \cot\beta) & \text{if } j \in N(i) \\ -\sum_{k \in N(i)} \mathbf{S}_{ik} & \text{if } i = j \end{cases}$$

<u>Observations</u>:

– [Authalicity]
  The mass matrix is invariant to area-preserving deformations.

– [Conformality]
  The stiffness matrix is invariant to angle-preserving deformations.



FEM Discretization

[WARNING]:

Given a discrete function $\mathbf{f} \leftrightarrow f(p)$, the vector:

$$\mathbf{g} = -\mathbf{S}\mathbf{f}$$

does not correspond to the Laplacian of $f(p)$.

The coefficients of the Laplacian of $f(p)$ satisfy:

$$\mathbf{M}\mathbf{g} = -\mathbf{S}\mathbf{f}$$

$$\Downarrow$$

$$\mathbf{g} = -\mathbf{M}^{-1}\mathbf{S}\mathbf{f}$$

FEM Discretization

<u>Laplacian Spectrum</u>:

In the continuous setting, the spectrum of the Laplacian, $\left\{\left(\phi^i: S \to \mathbb{R}, -\lambda_i \in \mathbb{R}^{\geq 0}\right)\right\}$, satisfies:

$$\Delta\phi^i = -\lambda_i\phi^i$$

And the $\{\phi^i\}$ form an orthonormal basis:

$$\langle\phi^i, \phi^j\rangle_s = \int_S \phi^i(p) \cdot \phi^j(p)\, dp = \delta_{ij}$$

The Spectrum of the Laplacian

<u>Interpreting the Eigenvalues</u>:

If ϕ is a (unit-norm) eigenfunction of the Laplacian, with eigenvalue λ:

$$\Delta\phi = -\lambda\phi$$
$$\Downarrow$$
$$\langle\Delta\phi, \phi\rangle_S = -\lambda\langle\phi, \phi\rangle_S$$
$$\Downarrow$$
$$-\langle\nabla\phi, \nabla\phi\rangle_S = -\|\nabla\phi\|_S^2 = -\lambda$$

$\Rightarrow$ The eigenvalue λ is a measure of how much ϕ changes, i.e. the frequency of ϕ.

FEM Discretization

<u>Laplacian Spectrum</u>:

In the discrete setting, the spectrum of the Laplacian, $\left\{\left(\boldsymbol{\phi}^i \in \mathbb{R}^n, \lambda_i \in \mathbb{R}^{\geq 0}\right)\right\}$, satisfies:

$$\mathbf{S}\boldsymbol{\phi}^i = \lambda_i \mathbf{M}\boldsymbol{\phi}^i$$

And the $\left\{\boldsymbol{\phi}^i\right\}$ form an orthonormal basis:

$$\langle \boldsymbol{\phi}^i, \boldsymbol{\phi}^j \rangle_s = \left(\boldsymbol{\phi}^i\right)^T \mathbf{M}\left(\boldsymbol{\phi}^j\right) = \delta_{ij}$$

Finding the $\left\{\left(\boldsymbol{\phi}^i, \lambda_i\right)\right\}$ is called the *generalized eigenvalue problem.*

How do we compute the spectral decomposition?

Getting the Dominant Eigenvector

Assume matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ is diagonalizable, with (unit-norm) spectrum $\{(\boldsymbol{\phi}^i, \lambda_i)\}$.

Given $\mathbf{v} \in \mathbb{R}^{\boldsymbol{n}}$, we have the decomposition:

$$\mathbf{v} = \sum_{i=1}^{n} \mathbf{v}_i \boldsymbol{\phi}^i$$

$$\Rightarrow \qquad \mathbf{A}\mathbf{v} = \sum_{i=1}^{n} \mathbf{v}_i \lambda_i \boldsymbol{\phi}^i$$

$$\Rightarrow \qquad \mathbf{A}^k\mathbf{v} = \sum_{i=1}^{n} \mathbf{v}_i \lambda_i^k \boldsymbol{\phi}^i$$

Getting the Dominant Eigenvector

$$\mathbf{A}^k\mathbf{v} = \sum_{i=1}^{n} \mathbf{v}_i \lambda_i^k \boldsymbol{\phi}^i$$

Without loss of too much generality, assume λ_n is the largest eigenvalue, $|\lambda_i/\lambda_n| < 1$ for $i \neq n$.

$$\mathbf{A}^k\mathbf{v} = \lambda_n^k \sum_{i=1}^{n} \mathbf{v}_i \left(\frac{\lambda_i}{\lambda_n}\right)^k \boldsymbol{\phi}^i$$

Then $(\lambda_i/\lambda_n)^k \to 0$ as $k \to \infty$, for $i \neq n$.

$$\Rightarrow \qquad \frac{\mathbf{A}^k\mathbf{v}}{|\mathbf{A}^k\mathbf{v}|} \to \boldsymbol{\phi}^n \quad \text{as} \quad k \to \infty$$

Getting the Dominant Eigenvector

ArnoldiDominant($\mathbf{A} \in \mathbb{R}^{n\times n}$)

1. $\mathbf{v} \leftarrow$ RandomVector( )
2. while(…)
3. $\quad\quad \mathbf{v} \leftarrow \mathbf{Av}$
4. $\quad\quad \mathbf{v} \leftarrow \mathbf{v}/|\mathbf{v}|$
5. $\lambda \leftarrow \langle \mathbf{Av}, \mathbf{v} \rangle$
6. return ($\mathbf{v}$, λ)

Getting the Sub-Dominant Eigenvector

If the matrix $\mathbf{A}$ is symmetric, the eigenvectors will be orthogonal:

<u>ArnoldiSubDominant</u>($\mathbf{A} \in \mathbb{R}^{n \times n}$)

1. $(\mathbf{v}^0, \lambda_0) \leftarrow$ ArnoldiDominant($\mathbf{A}$)
2. $\mathbf{v}^1 \leftarrow$ RandomVector( )
3. while(…)
4. $\quad \mathbf{v}^1 \leftarrow \mathbf{A}\mathbf{v}^1$
5. $\quad \mathbf{v}^1 \leftarrow \mathbf{v}^1 - \langle \mathbf{v}^1, \mathbf{v}^0 \rangle \mathbf{v}^0$
6. $\quad \mathbf{v}^1 \leftarrow \mathbf{v}^1 / |\mathbf{v}^1|$
7. $\lambda_1 \leftarrow \langle \mathbf{A}\mathbf{v}^1, \mathbf{v}^1 \rangle$
8. return ($\mathbf{v}^1$, λ_1)

Getting the Sub-Dominant Eigenvector

If the matrix $\mathbf{A}$ is symmetric, the eigenvectors will be orthogonal:

<u>ArnoldiSubDominant</u>($\mathbf{A} \in \mathbb{R}^{n \times n}$)

1. $(\mathbf{v}^0, \lambda_0) \leftarrow$ ArnoldiDominant($\mathbf{A}$)
2. $\mathbf{v}^1 \leftarrow$ RandomVector( )
3. while(…)
4. $\quad\quad \mathbf{v}^1 \leftarrow \mathbf{A}\mathbf{v}^1$
5. $\quad\quad \mathbf{v}^1 \leftarrow \mathbf{v}^1 - \langle \mathbf{v}^1, \mathbf{v}^0 \rangle \mathbf{v}^0$

A similar approach can be applied to:

- Solving the generalized eigenvalue problem
- Finding the eigenvectors with smallest eigenvalues
- Finding the eigenvectors with eigenvalues closest to λ

Outline

- Motivation
- Laplacian Spectrum
- Applications
  - Signal/Geometry Filtering
  - Partial Differential Equations
  - Complexity and Approximation
- Conclusion

Signal/Geometry Filtering

<u>HarmonicDecomposition</u>($S \subset \mathbb{R}^3$, $\mathbf{f} \in \mathbb{R}^n$)

1. $(\mathbf{M}, \mathbf{S}) \leftarrow$ MassAndStiffness (S)
2. $\{(\boldsymbol{\phi}^i, \lambda_i)\}_{i=1}^n \leftarrow$ GeneralizedEigen($\mathbf{M}$, $\mathbf{S}$)
3. For each $i \in [1, n]$:
4. $\quad\hat{\mathbf{f}}_i \leftarrow \langle \mathbf{f}, \boldsymbol{\phi}^i \rangle_S$

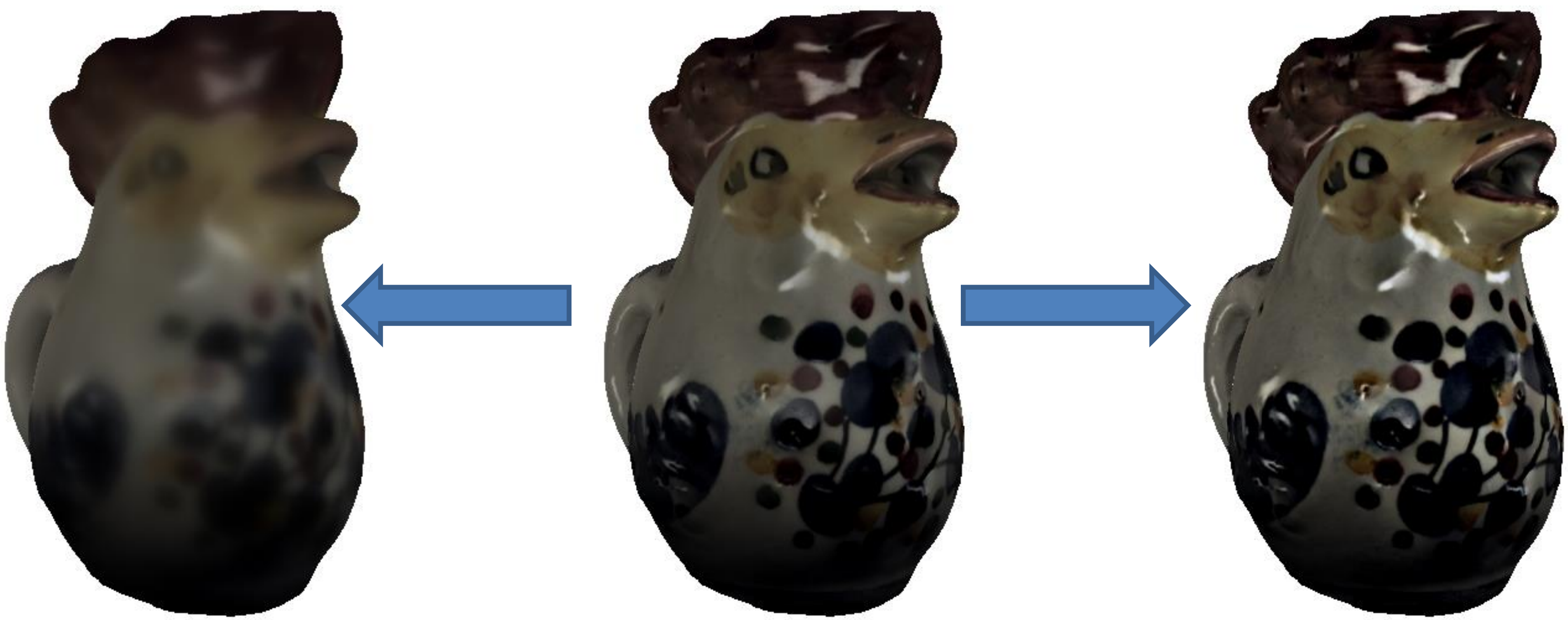
$\mathbf{f}^T \mathbf{M} \boldsymbol{\phi}^i$

<u>Process</u>($F: \mathbb{R} \rightarrow \mathbb{R}$)

1. $\mathbf{g} \leftarrow 0$
2. For each $i \in [1, n]$:
3. $\quad\hat{\mathbf{g}}_i \leftarrow \hat{\mathbf{f}}_i F(\lambda_i)$
4. $\quad\mathbf{g} \leftarrow \mathbf{g} + \hat{\mathbf{g}}_i \boldsymbol{\phi}^i$
5. return $\mathbf{g}$

Signal Filtering

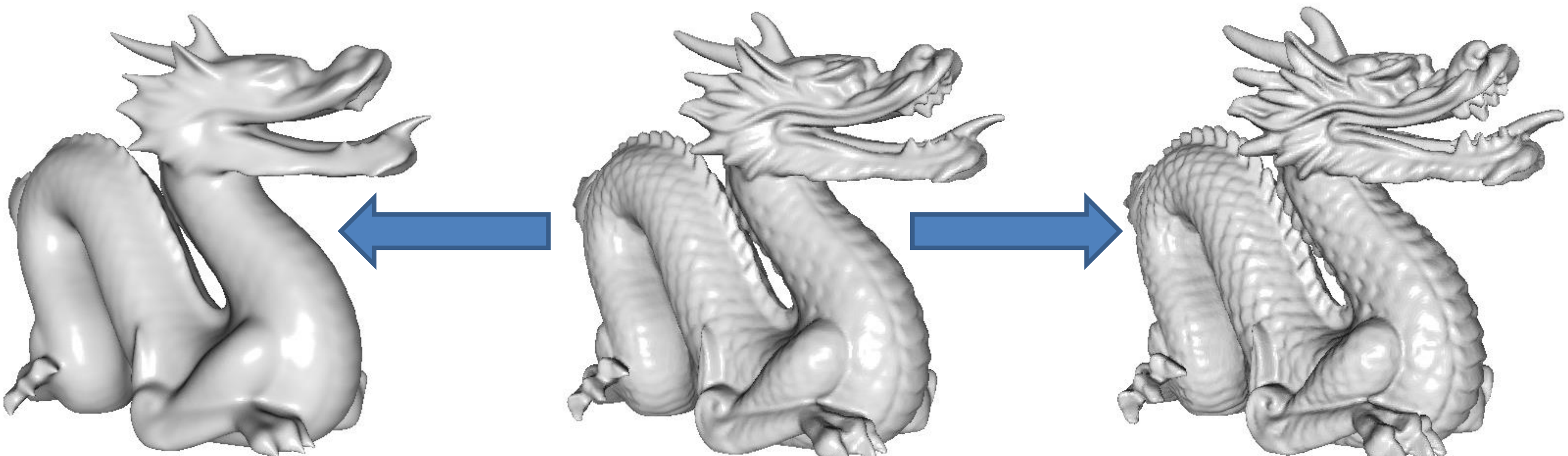
Given a color at each vertex, we can modulate the frequency coefficients of each channel to smooth/sharpen the colors.



$F(\lambda) = e^{-\lambda}$ Input $F(\lambda) = \left(2 - e^{-\lambda}\right)$

Geometry Filtering

Using the position of the vertices as the signal, we can modulate the frequency coefficients of each coordinate to smooth/sharpen the shape.



$F(\lambda) = e^{-\lambda}$ Input $F(\lambda) = \left(2 - e^{-\lambda}\right)$

Partial Differential Equations

Recall:

The Laplacian of a function at a point $p \in S$ is the difference between the value at p and the average value of its neighbors.

Heat Diffusion

Newton's Law of Cooling:

The rate of heat loss of a body is proportional to the difference in temperatures between the body and its surroundings.

Translating this into the PDE, if $h(p,t)$ is the heat at position $p \in S$ at time t, then:

$$\frac{\partial h}{\partial t} = \eta \Delta h$$

Heat Diffusion

Newton's Law of Cooling:

The rate of heat loss of a body is proportional to the difference in temperatures between the body and its surroundings.

Goal:

Given an initial heat distribution $h^0: S \to \mathbb{R}$, find the solution to the PDE:

$$\frac{\partial h}{\partial t} = \eta \Delta h$$

such that:

$$h(p, 0) = h^0(p)$$

Heat Diffusion

$$\frac{\partial h}{\partial t} = \eta \Delta h$$

Note:

Let $\left\{\left(\phi^i, -\lambda_i\right)\right\}_{i=1}^{n}$ be the Laplacian spectrum.

$\Rightarrow$ The functions:

$$\mathcal{H}_i(p,t) = e^{-\eta\lambda_i t}\phi^i(p)$$

are solutions to the PDE.

$\Rightarrow$ Any linear sum of the $\mathcal{H}_i(p,t)$ is a solution.

Heat Diffusion

$$\frac{\partial h}{\partial t} = \eta \Delta h$$

Compute the harmonic decomposition of h^0:

$$h^0(p) = \sum_{i=1}^{n} \hat{\mathbf{h}}_i^0 \phi^i(p)$$

Then consider the function:

$$h(p,t) = \sum_{i=0}^{n} \hat{\mathbf{h}}_i^0 \mathcal{H}_i(p,t) = \sum_{i=0}^{n} \hat{\mathbf{h}}_i^0 e^{-\eta\lambda_i t} \phi^i(p)$$

– It is a solution to the heat equation.

– It satisfies $h(p,0) = h^0(p)$.

Heat Diffusion (Colors)

<u>HarmonicDecomposition</u>($S \subset \mathbb{R}^3$, $h^0 : S \rightarrow \mathbb{R}$)

1. …

<u>Process</u>($t \in [0, \infty)$)

1. $\mathbf{g} \leftarrow 0$
2. For each $i \in [1, n]$:
3. $\quad \hat{\mathbf{g}}_i \leftarrow \hat{\mathbf{h}}_i^0 e^{-\eta \lambda_i t}$
4. $\quad \mathbf{g} \leftarrow \mathbf{g} + \hat{\mathbf{g}}_i \boldsymbol{\phi}^i$
5. return $\mathbf{g}$

Heat Diffusion (Geometry)

HarmonicDecomposition($S \subset \mathbb{R}^3$, $h^0: S \to \mathbb{R}^3$)

1. …

Process($t \in [0, \infty)$)

1. $\mathbf{g} \leftarrow 0$
2. For each $i \in [1, n]$:
3. $\quad \hat{\mathbf{g}}_i \leftarrow \hat{\mathbf{h}}_i^0 e^{-\eta\lambda_i t}$
4. $\quad \mathbf{g} \leftarrow \mathbf{g} + \hat{\mathbf{g}}_i \boldsymbol{\phi}^i$
5. return $\mathbf{g}$

Heat Diffusion (Geometry)

[WARNING]:

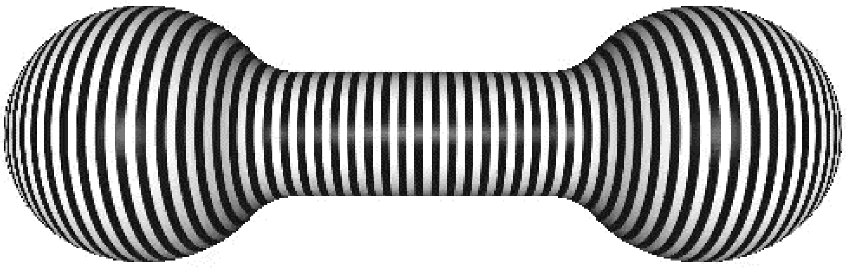
1. As the geometry diffuses, the areas and angles of the triangles change.

   ⇒ The mass and stiffness matrices change.

   ⇒ The harmonic decomposition changes.

   If we take this into account, we get a non-linear PDE called *mean curvature flow*.

2. Mean curvature flow can create singularities.



Wave Equation

The acceleration of a wave's height is proportional to the difference in height of the surrounding.

Translating this into the PDE, if $h(p,t)$ is the height at position $p \in S$ at time t, then:

$$\frac{\partial^2 h}{\partial t^2} = \eta \Delta h$$

Wave Equation

The acceleration of a wave's height is proportional to the difference in height of the surrounding.

Goal:

Given an initial height distribution $h^0 : S \to \mathbb{R}$, find the solution to the PDE:

$$\frac{\partial^2 h}{\partial t^2} = \eta \Delta h$$

such that:

$$h(p,0) = h^0(p) \qquad \text{and} \qquad \frac{\partial h}{\partial t}(p,0) = 0$$

Wave Equation

$$\frac{\partial^2 h}{\partial t^2} = \eta \Delta h$$

Note:

If $\left\{\left(\phi^i, -\lambda_i\right)\right\}_{i=1}^{n}$ are the eigenfunctions/values of the Laplacian, then:

$$\mathcal{H}_i^c(p,t) = \cos\left(\sqrt{\eta\lambda_i}t\right)\phi^i(p)$$

$$\mathcal{H}_i^s(p,t) = \sin\left(\sqrt{\eta\lambda_i}t\right)\phi^i(p)$$

are solutions to the PDE.

$\Rightarrow$ Any linear sum of the $\mathcal{H}_i^c(p,t)$ and $\mathcal{H}_i^s(p,t)$ is a solution to the PDE.

Wave Equation

$$\frac{\partial^2 h}{\partial t^2} = \eta \Delta h$$

Compute the harmonic decomposition of h^0:

$$h^0(p) = \sum_{i=1}^{n} \hat{\mathbf{h}}_i^0 \phi^i(p)$$

Then consider the function:

$$h(p,t) = \sum_{i=0}^{n} \hat{\mathbf{h}}_i^0 \mathcal{H}_i^c(p,t) = \sum_{i=0}^{n} \hat{\mathbf{h}}_i^0 \cos\left(\sqrt{\eta\lambda_i}\,t\right) \phi^i(p)$$

– It is a solution to the wave equation.

– It satisfies $h(p,0) = h^0(p)$.

– It satisfies $\frac{\partial h}{\partial t}(p,0) = 0$.

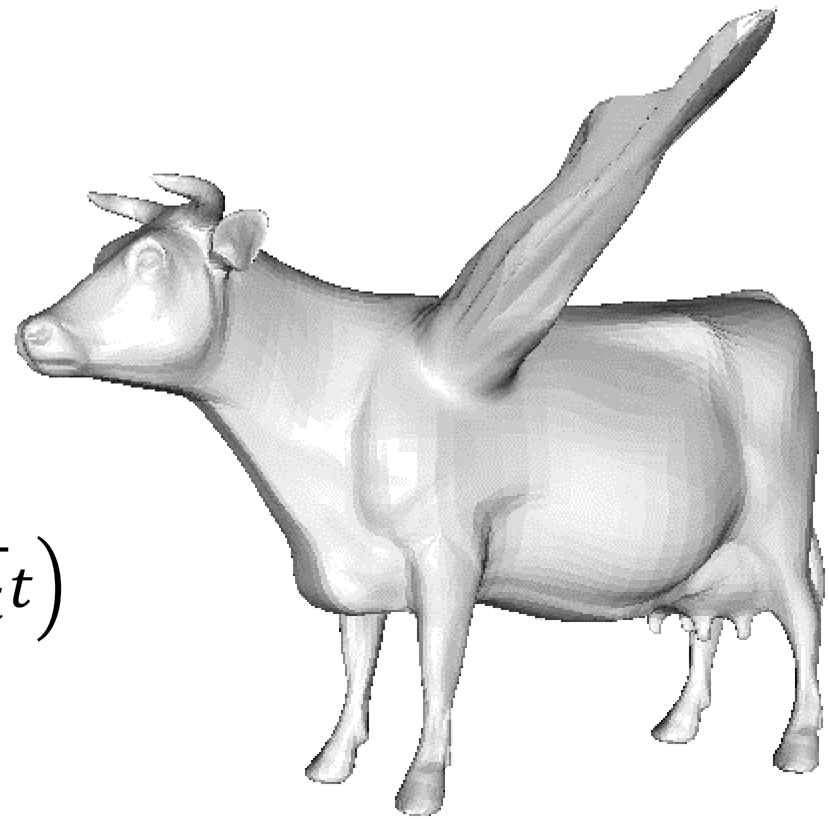
Wave Equation

HarmonicDecomposition($S \subset \mathbb{R}^3$, $h^0: S \to \mathbb{R}$)

1. …

Process($t \in [0, \infty)$)

1. $\mathbf{g} \leftarrow 0$
2. For each $i \in [1, n]$:
3. $\quad \hat{\mathbf{g}}_i \leftarrow \hat{\mathbf{h}}_i^0 \cos\left(\sqrt{\eta\lambda_i}t\right)$
4. $\quad \mathbf{g} \leftarrow \mathbf{g} + \hat{\mathbf{g}}_i \boldsymbol{\phi}^i$
5. return **g**



How practical is it to use the spectral decomposition?

Complexity

Challenge:

If we have a mesh with n vertices we get n generalized eigenvectors.

✘ $O(n^2)$ storage / $O(> n^2)$ computation.

Approximate:

Sometimes a low-frequency solution will do.

✓ $O(kn)$ storage

Sometimes a numerically inaccurate solution will do.

✓ $O(n)$ storage / $O(?)$ computation

Approximate Spectral Processing

Preliminaries:

Let $\mathbf{M}$ be the mass matrix, $\mathbf{S}$ the stiffness matrix, and $\{(\boldsymbol{\phi}^i, \lambda_i)\}$ the spectrum, we have:

$$\mathbf{M}\boldsymbol{\phi}^i = \mathbf{M}\boldsymbol{\phi}^i \qquad \mathbf{S}\boldsymbol{\phi}^i = \lambda_i \mathbf{M}\boldsymbol{\phi}^i$$

Taking α times the 1st equation plus β times the 2nd:

$$(\alpha\mathbf{M} + \beta\mathbf{S})\boldsymbol{\phi}^i = (\alpha + \beta\lambda_i)\mathbf{M}\boldsymbol{\phi}^i$$

Multiplying by $(\alpha\mathbf{M} + \beta\mathbf{L})^{-1}$:

$$\boldsymbol{\phi}^i = (\alpha + \beta\lambda_i)\big((\alpha\mathbf{M} + \beta\mathbf{S})^{-1} \circ \mathbf{M}\big)\boldsymbol{\phi}^i$$

$$\frac{1}{(\alpha + \beta\lambda_i)}\boldsymbol{\phi}^i = \big((\alpha\mathbf{M} + \beta\mathbf{S})^{-1} \circ \mathbf{M}\big)\boldsymbol{\phi}^i$$

Approximate Spectral Processing

Preliminaries:

$$(\gamma + \delta\lambda_i)\mathbf{M}\boldsymbol{\phi}^i = (\gamma\mathbf{M} + \delta\mathbf{S})\boldsymbol{\phi}^i$$

$$\frac{1}{(\alpha + \beta\lambda_i)}\boldsymbol{\phi}^i = \left((\alpha\mathbf{M} + \beta\mathbf{S})^{-1} \circ \mathbf{M}\right)\boldsymbol{\phi}^i$$

Combining these, we get:

$$\begin{aligned}\left((\alpha\mathbf{M} + \beta\mathbf{S})^{-1} \circ (\gamma\mathbf{M} + \delta\mathbf{S})\right)\boldsymbol{\phi}^i \\ &= (\gamma + \delta\lambda_i)\left((\alpha\mathbf{M} + \beta\mathbf{S})^{-1} \circ \mathbf{M}\right)\boldsymbol{\phi}^i \\ &= \frac{\gamma + \delta\lambda_i}{\alpha + \beta\lambda_i}\boldsymbol{\phi}^i\end{aligned}$$

Approximate Spectral Processing

<u>Example (Signal Smoothing)</u>:

The goal is to obtain a smoothed signal:

$$\hat{\mathbf{f}}_i \leftarrow \hat{\mathbf{f}}_i F(\lambda_i)$$

We can relax the condition that $F(\lambda) = e^{-\lambda}$ and use a different filter $F: \mathbb{R} \rightarrow \mathbb{R}$.

The new filter should:

– preserve the low frequencies

– decay at higher frequencies

Approximate Spectral Processing

$$\big((\alpha\mathbf{M}+\beta\mathbf{S})^{-1}\circ(\gamma\mathbf{M}+\delta\mathbf{S})\big)\boldsymbol{\phi}^i=\left(\frac{\gamma+\delta\lambda_i}{\alpha+\beta\lambda_i}\right)\boldsymbol{\phi}^i$$

Smoothing Example ($\alpha=1,\gamma=1,\delta=0$):

Consider the solution to the linear system:

$$\mathbf{g}=\big((\mathbf{M}+\beta\mathbf{S})^{-1}\circ\mathbf{M}\big)\mathbf{f}$$

Taking the spectral decomposition of $\mathbf{f}$:

$$\mathbf{g}=\sum\hat{\mathbf{f}}_i\big((\mathbf{M}+\beta\mathbf{S})^{-1}\circ\mathbf{M}\big)\boldsymbol{\phi}^i$$

$$=\sum\hat{\mathbf{f}}_i\,\frac{1}{1+\beta\lambda_i}\,\boldsymbol{\phi}^i$$

Approximate Spectral Processing

$$\left((\alpha\mathbf{M}+\beta\mathbf{S})^{-1}\circ(\gamma\mathbf{M}+\delta\mathbf{S})\right)\boldsymbol{\phi}^i=\left(\frac{\gamma+\delta\lambda_i}{\alpha+\beta\lambda_i}\right)\boldsymbol{\phi}^i$$

<u>Smoothing Example ($\alpha=1,\gamma=1,\delta=0$)</u>:

Consider the solution to the linear system:

$$\mathbf{g}=\left((\mathbf{M}+\beta\mathbf{S})^{-1}\circ\mathbf{M}\right)\mathbf{f}$$

Solving this linear system is equivalent to filtering with:

$$F(\lambda)=\frac{1}{1+\beta\lambda}$$

with β the rate of decay of higher frequencies.

$$=\sum\hat{\mathbf{f}}_i\frac{1}{1+\beta\lambda_i}\boldsymbol{\phi}^i$$

Approximate Spectral Processing

$$\left((\alpha\mathbf{M}+\beta\mathbf{S})^{-1}\circ(\gamma\mathbf{M}+\delta\mathbf{S})\right)\boldsymbol{\phi}^i=\left(\frac{\gamma+\delta\lambda_i}{\alpha+\beta\lambda_i}\right)\boldsymbol{\phi}^i$$

Sharpening Example ($\alpha=1,\gamma=1,\delta=\beta\sigma$):

Consider the solution to the linear system:

$$\mathbf{g}=\left((\mathbf{M}+\beta\mathbf{L})^{-1}\circ(\mathbf{M}+\beta\sigma\mathbf{L})\right)\mathbf{f}$$

Taking the spectral decomposition of $\mathbf{f}$:

$$\mathbf{g}=\sum\hat{\mathbf{f}}_i\frac{1+\sigma\beta\lambda_i}{1+\beta\lambda_i}\boldsymbol{\phi}^i$$

Approximate Spectral Processing

$$\left((\alpha\mathbf{M}+\beta\mathbf{S})^{-1}\circ(\gamma\mathbf{M}+\delta\mathbf{S})\right)\boldsymbol{\phi}^i=\left(\frac{\gamma+\delta\lambda_i}{\alpha+\beta\lambda_i}\right)\boldsymbol{\phi}^i$$

<u>Sharpening Example ($\alpha=1,\gamma=1,\delta=\beta\sigma$)</u>:

Consider the solution to the linear system:

$$\mathbf{g}=\left((\mathbf{M}+\beta\mathbf{L})^{-1}\circ(\mathbf{M}+\beta\sigma\mathbf{L})\right)\mathbf{f}$$

$$\Rightarrow\qquad F(\lambda)=\frac{1+\sigma\beta\lambda}{1+\beta\lambda}$$

This filter satisfies:

- $\lim_{\lambda\to 0} F(\lambda)=1$: Low-frequencies preserved
- $\lim_{\lambda\to\infty} F(\lambda)=\sigma$: High frequencies scaled by σ.

Approximate Spectral Processing

$$\left((\alpha\mathbf{M}+\beta\mathbf{S})^{-1}\circ(\gamma\mathbf{M}+\delta\mathbf{S})\right)\boldsymbol{\phi}^i=\left(\frac{\gamma+\delta\lambda_i}{\alpha+\beta\lambda_i}\right)\boldsymbol{\phi}^i$$

Sharpening Example ($\alpha=1,\gamma=1,\delta=\beta\sigma$):

Consider the solution to the linear system:

$$\mathbf{g}=\left((\mathbf{M}+\beta\mathbf{L})^{-1}\circ(\mathbf{M}+\beta\sigma\mathbf{L})\right)\mathbf{f}$$

$$\Rightarrow\qquad F(\lambda)=\frac{1+\sigma\beta\lambda}{1+\beta\lambda}$$

Signal smoothing is a special instance, with $\sigma=0$.

– $\lim_{\lambda\to 0}F(\lambda)=1$: Low-frequencies preserved

– $\lim_{\lambda\to\infty}F(\lambda)=\sigma$: High frequencies scaled by σ.

Approximate Spectral Processing

Process($S \subset \mathbb{R}^3$, $\mathbf{f} \in \mathbb{R}^n$, $\sigma \in \mathbb{R}$, $\beta \in \mathbb{R}$)

1. $(\mathbf{M}, \mathbf{L}) \leftarrow$ MassAndStiffness (S)
2. $\mathbf{g} \leftarrow (\mathbf{M} + \beta\sigma\mathbf{L})\mathbf{f}$
3. $\mathbf{A} \leftarrow (\mathbf{M} + \beta\mathbf{L})$
4. return Solve($\mathbf{A}$, $\mathbf{g}$)

By approximating, we replace the computational complexity of storing/computing the spectral decomposition with the complexity of solving a sparse linear system.

Heat Diffusion (Revisited)

$$\frac{\partial h}{\partial t} = \Delta h \quad \text{s.t.} \quad h(p,0) = h^0(p)$$

Discretization (Temporal):

Letting $h^t\colon S \to \mathbb{R}$ be the solution at time t, we can (temporally) discretize the PDE in two ways:

Explicit	Implicit
$\frac{h^{t+\varepsilon} - h^t}{\varepsilon} \approx \Delta h^t$	$\frac{h^{t+\varepsilon} - h^t}{\varepsilon} \approx \Delta h^{t+\varepsilon}$
$\Downarrow$	$\Downarrow$
$h^{t+\varepsilon} = h^t + \varepsilon \Delta h^t$	$(1 - \varepsilon\Delta) h^{t+\varepsilon} = h^t$

Heat Diffusion (Revisited)

Explicit

$$h^{t+\varepsilon} = h^t + \varepsilon\Delta h^t$$

Implicit

$$(1 - \varepsilon\Delta)h^{t+\varepsilon} = h^t$$

Discretization (Spatial):

Projecting onto the discrete function basis gives:

Explicit:

$$\Downarrow$$

$$\mathbf{M}\mathbf{h}^{t+\varepsilon} = \mathbf{M}\mathbf{h}^t - \varepsilon\mathbf{S}\mathbf{h}^t$$

$$\Downarrow$$

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

Implicit:

$$\Downarrow$$

$$(\mathbf{M} + \varepsilon\mathbf{S})\mathbf{h}^{t+\varepsilon} = \mathbf{M}\mathbf{h}^t$$

$$\Downarrow$$

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

Heat Diffusion (Revisited)

Explicit

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

$$\alpha = 1, \beta = 0, \gamma = 1, \delta = -\varepsilon$$

$$\Downarrow$$

$$\hat{\mathbf{h}}_i^{t+\varepsilon} = (1 - \varepsilon\lambda_i)\hat{\mathbf{h}}_i^t$$

Implicit

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

$$\alpha = 1, \beta = \varepsilon, \gamma = 1, \delta = 0$$

$$\Downarrow$$

$$\hat{\mathbf{h}}_i^{t+\varepsilon} = \frac{1}{1 + \varepsilon\lambda_i}\hat{\mathbf{h}}_i^t$$

Discretization:

Both give an inaccurate answer when a large time-step, ε, is used. But…

$$\mathbf{g} = \left((\alpha\mathbf{M} + \beta\mathbf{S})^{-1} \circ (\gamma\mathbf{M} + \delta\mathbf{S})\right)\mathbf{f}$$

$$\Rightarrow \qquad F(\lambda) = \frac{\gamma + \delta\lambda}{\alpha + \beta\lambda}$$

Heat Diffusion (Revisited)

Explicit

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

$$\Downarrow$$

$$\hat{\mathbf{h}}_i^{t+\varepsilon} = (1 - \varepsilon\lambda_i)\hat{\mathbf{h}}_i^{t}$$

Implicit

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

$$\Downarrow$$

$$\hat{\mathbf{h}}_i^{t+\varepsilon} = \frac{1}{1 + \varepsilon\lambda_i}\hat{\mathbf{h}}_i^{t}$$

Discretization:

Both filters preserve low frequencies:

$$\lim_{\lambda\to 0}(1 - \varepsilon\lambda) = 1 \qquad \Bigg| \qquad \lim_{\lambda\to 0}\left(\frac{1}{1 + \varepsilon\lambda}\right) = 1$$

But at high frequencies (and large time-steps):

$$\lim_{\lambda\to \infty}(1 - \varepsilon\lambda) = -\infty \qquad \Bigg| \qquad \lim_{\lambda\to 0}\left(\frac{1}{1 + \varepsilon\lambda}\right) = 0$$

Heat Diffusion (Revisited)

Explicit	Implicit

Though neither approximation gives an accurate answer at large times-steps, implicit integration is guaranteed to be (*unconditionally*) *stable.*

D

B

A similar approach can be used to approximate the solution to the wave equation without a harmonic decomposition.

$$\lim_{\lambda \to 0} (1 - \varepsilon\lambda) = 1 \qquad \lim_{\lambda \to 0} \left(\frac{1}{1 + \varepsilon\lambda} \right) = 1$$

But at high frequencies (and large time-steps):

$$\lim_{\lambda \to \infty} (1 - \varepsilon\lambda) = -\infty \qquad \lim_{\lambda \to 0} \left(\frac{1}{1 + \varepsilon\lambda} \right) = 0$$

Outline

- Motivation
- Laplacian Spectrum
- Applications
- Conclusion

Conclusion

Though there is no Fourier Transform for general surfaces, we can use the spectrum of the Laplacian to get a frequency decomposition.

This enables:

- Filtering of signals
- Solving PDEs

by modulating the frequency coefficients.

Conclusion

Though computing a full spectral decomposition is not space/time efficient, we can often:

- Use the lower frequencies.
- Design linear operators whose solution has the desired frequency modulation.

Using the theory of spectral decomposition:

- We can design stable simulations, without explicitly computing the decomposition.