Spectral Geometry Processing with Manifold Harmonics

Bruno Vallet
Bruno Lévy

Introduction

Introduction

II. Harmonics

III. DEC formulation

IV. Filtering

V. Numerics

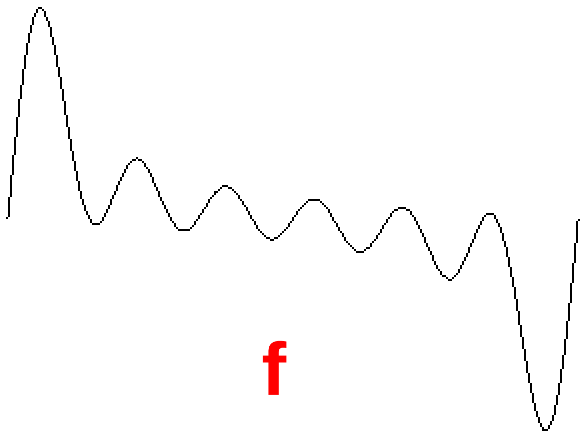
Results and conclusion

Introduction

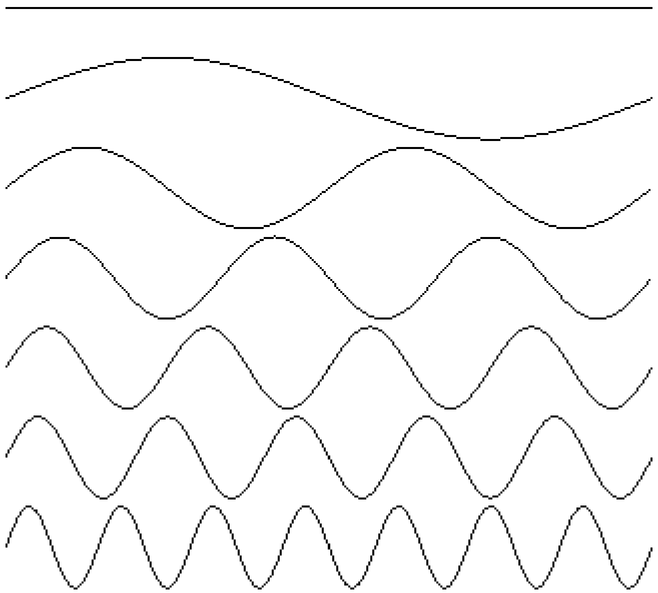
Extend to meshes:

- Fourier transform
- Spectral filtering

Fourier transform

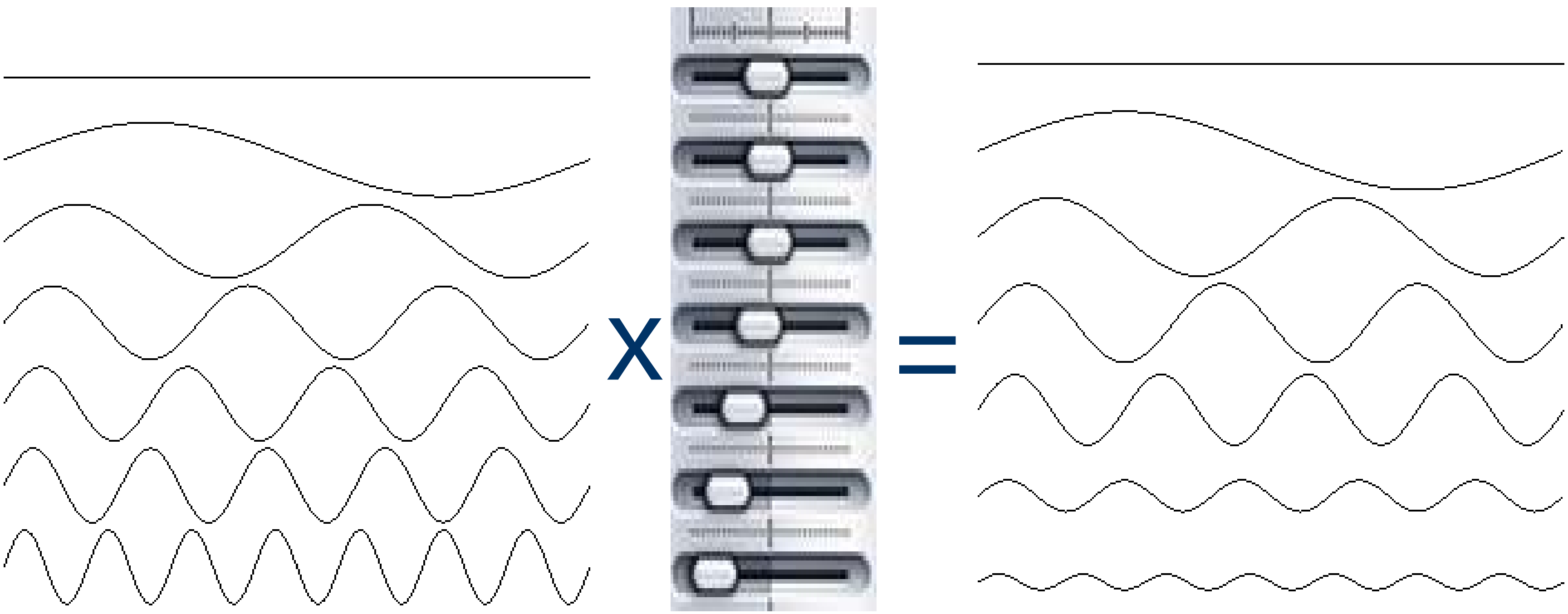


$= \sum$



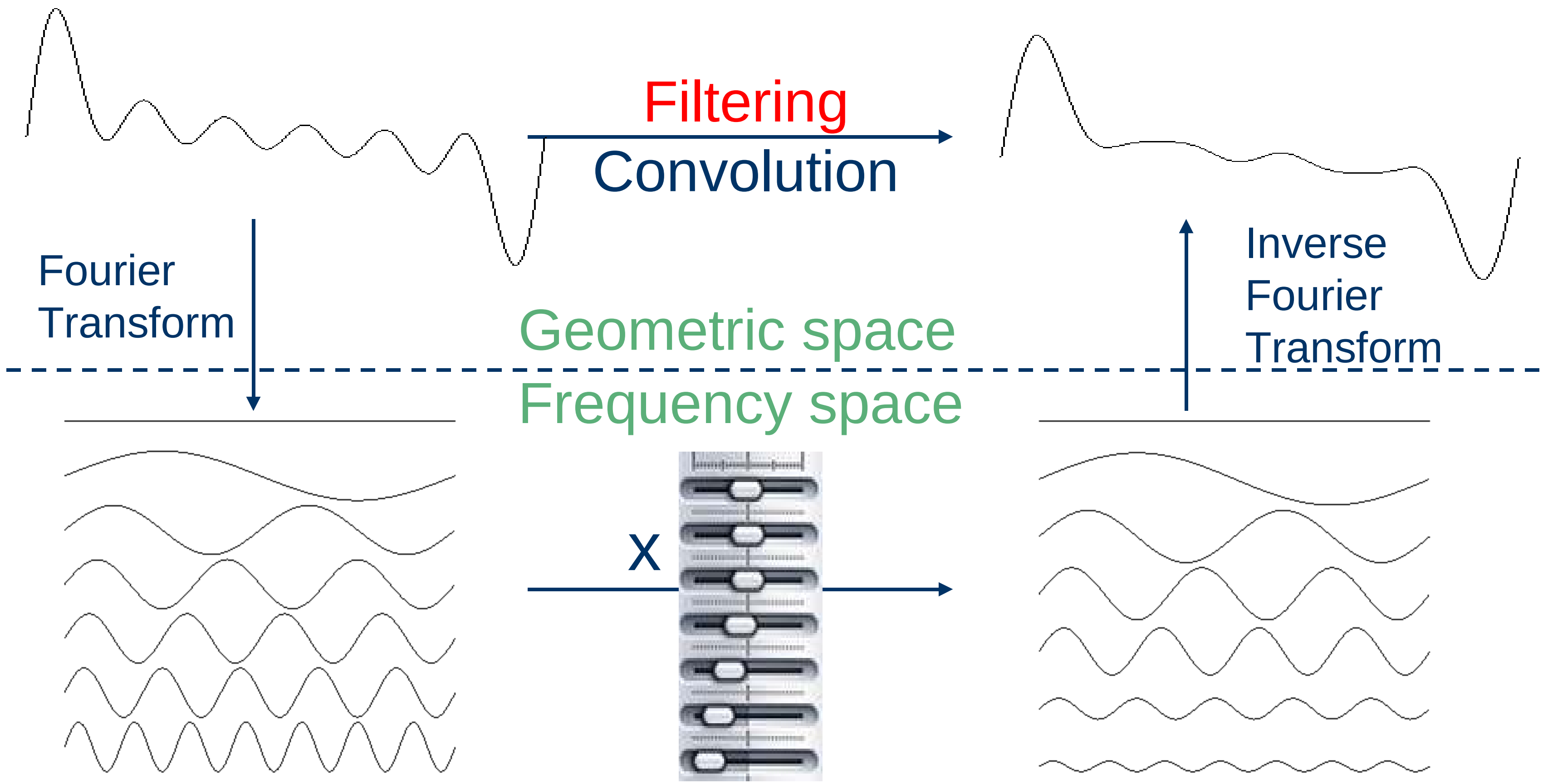
Introduction

Filtering



Introduction

Filtering



Filtering on a mesh

Filtering

[Taubin 95]

Geometric space

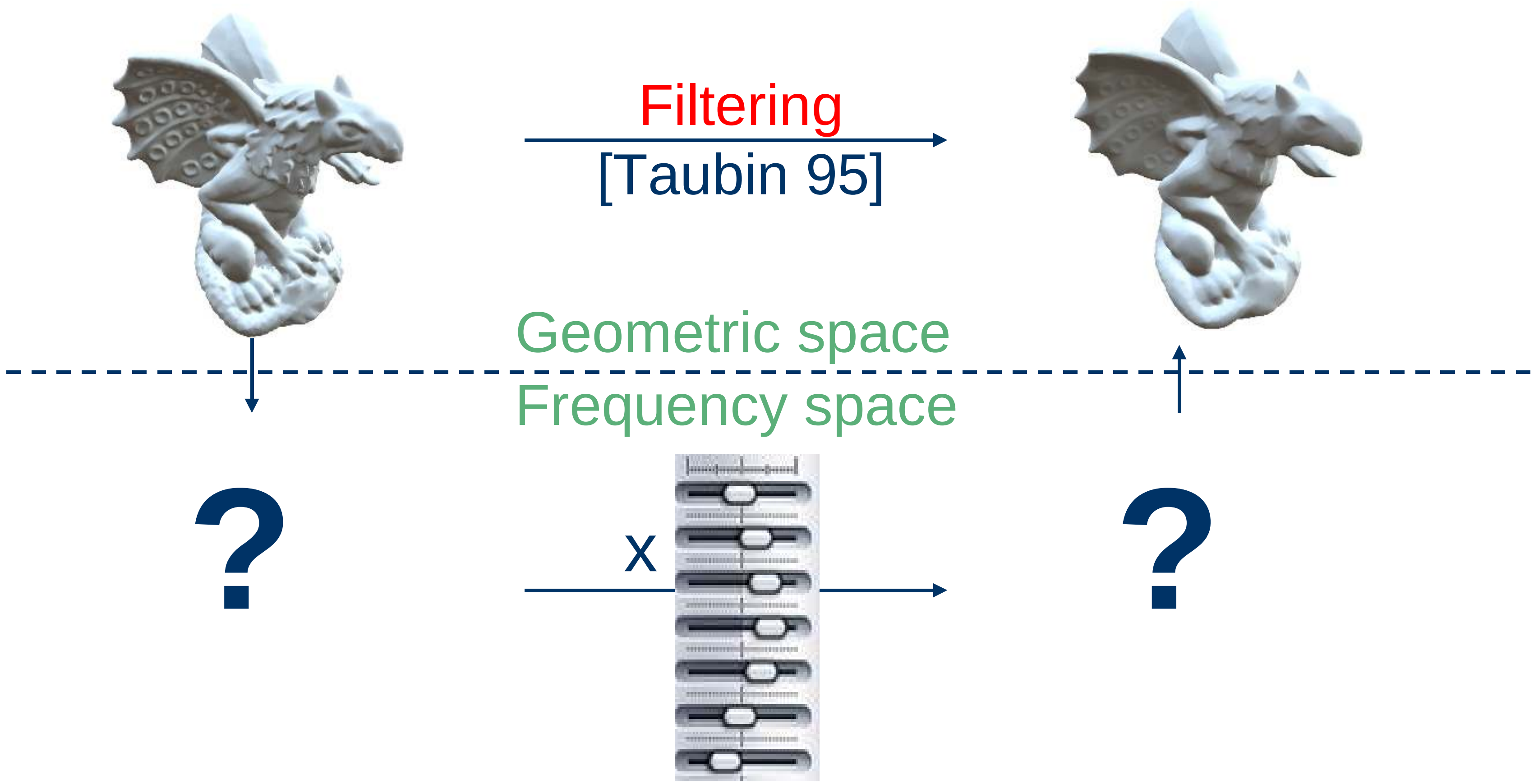
Frequency space

?

x

?

Filtering on a mesh



Filtering on a mesh

Filtering

[Taubin 95]

Geometric space

Frequency space

?

[Karni00] mesh compression
[Zhang06] shape matching
[Dong06] quadrangulation

?

I Harmonics

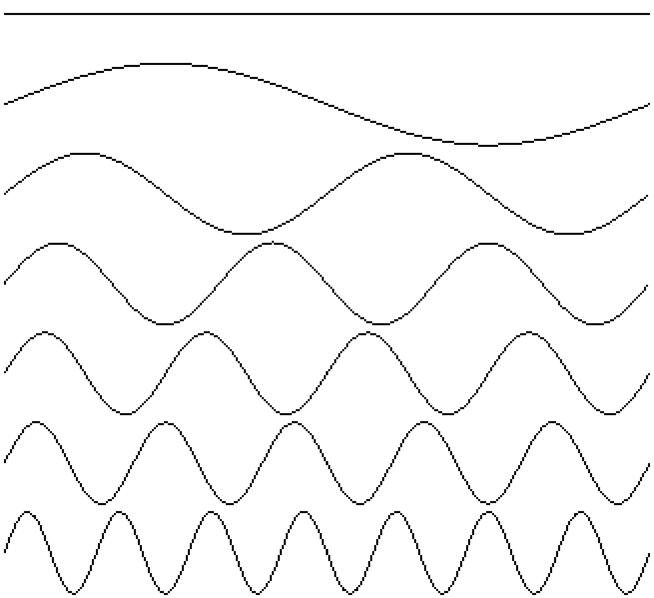
Introduction

- **Harmonics**
- DEC formulation
- Filtering
- Numerics

Results and conclusion

I Harmonics

Question

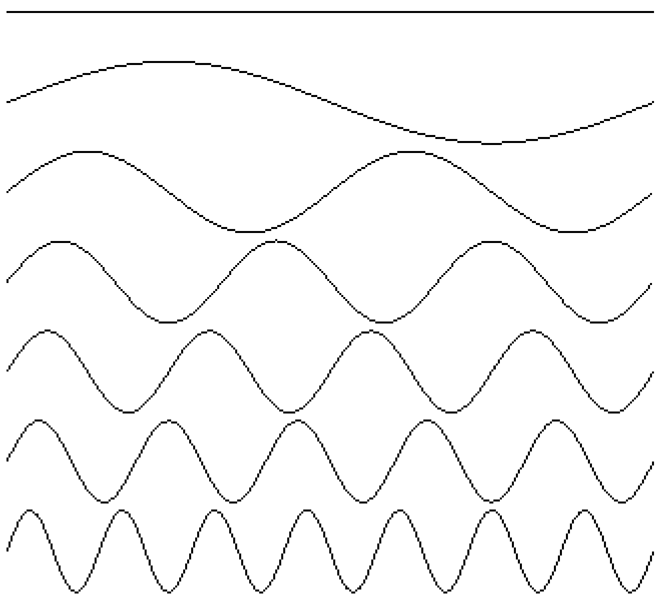


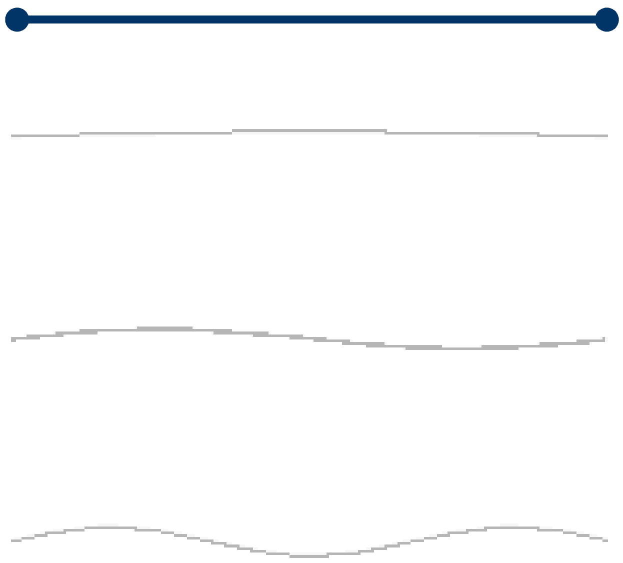
sin(kx)

on

?

Harmonics and vibrations

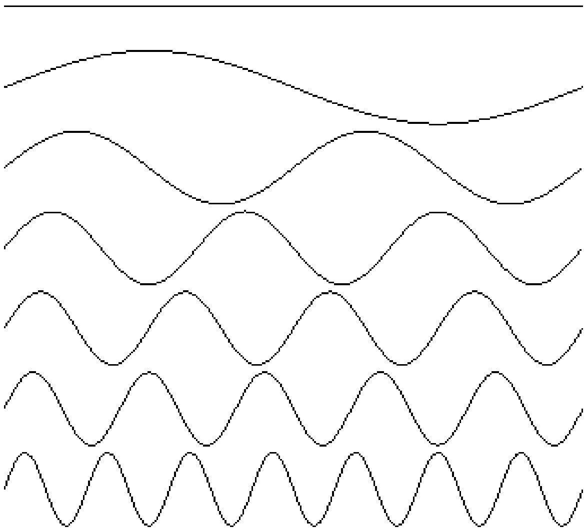




sin(kx) are the stationary vibrating modes = harmonics of a string

Manifold Harmonics

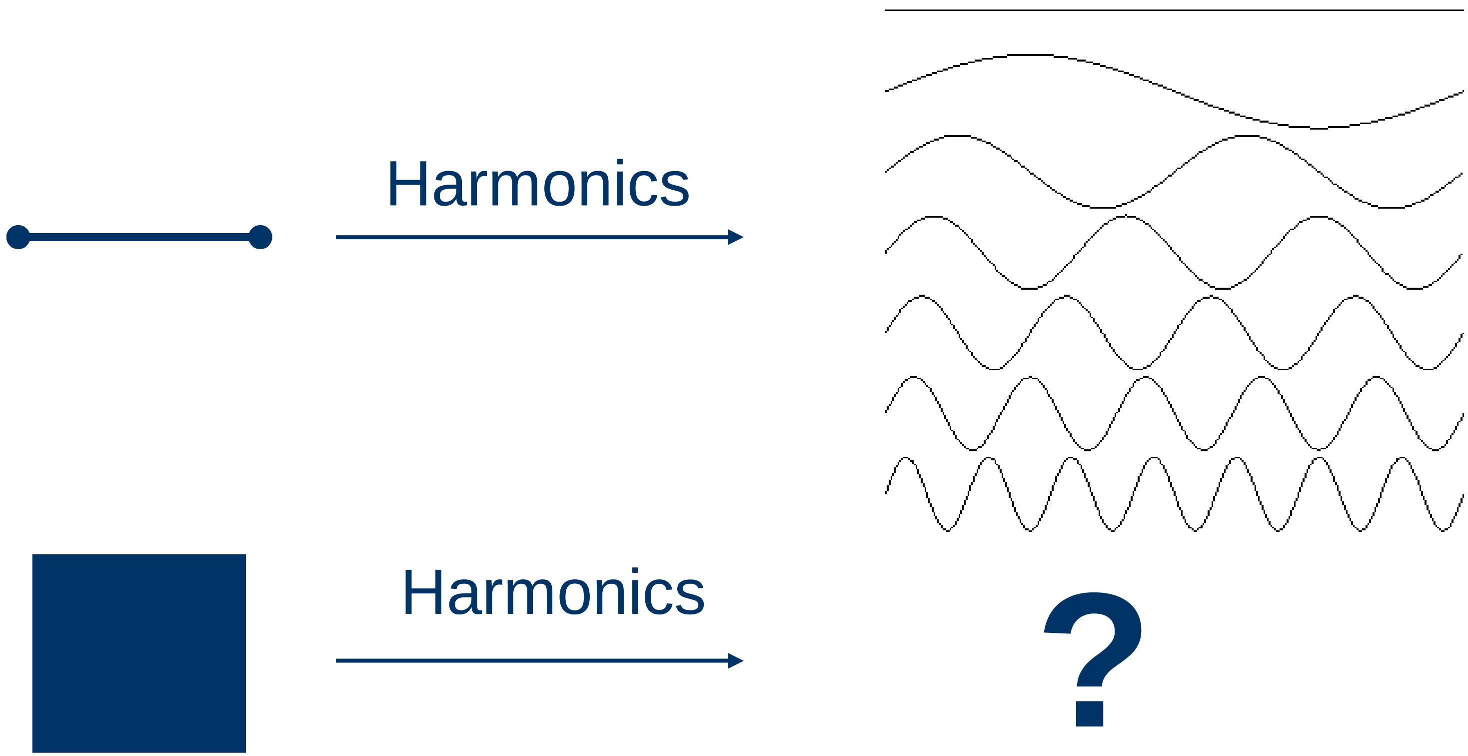
Harmonics →



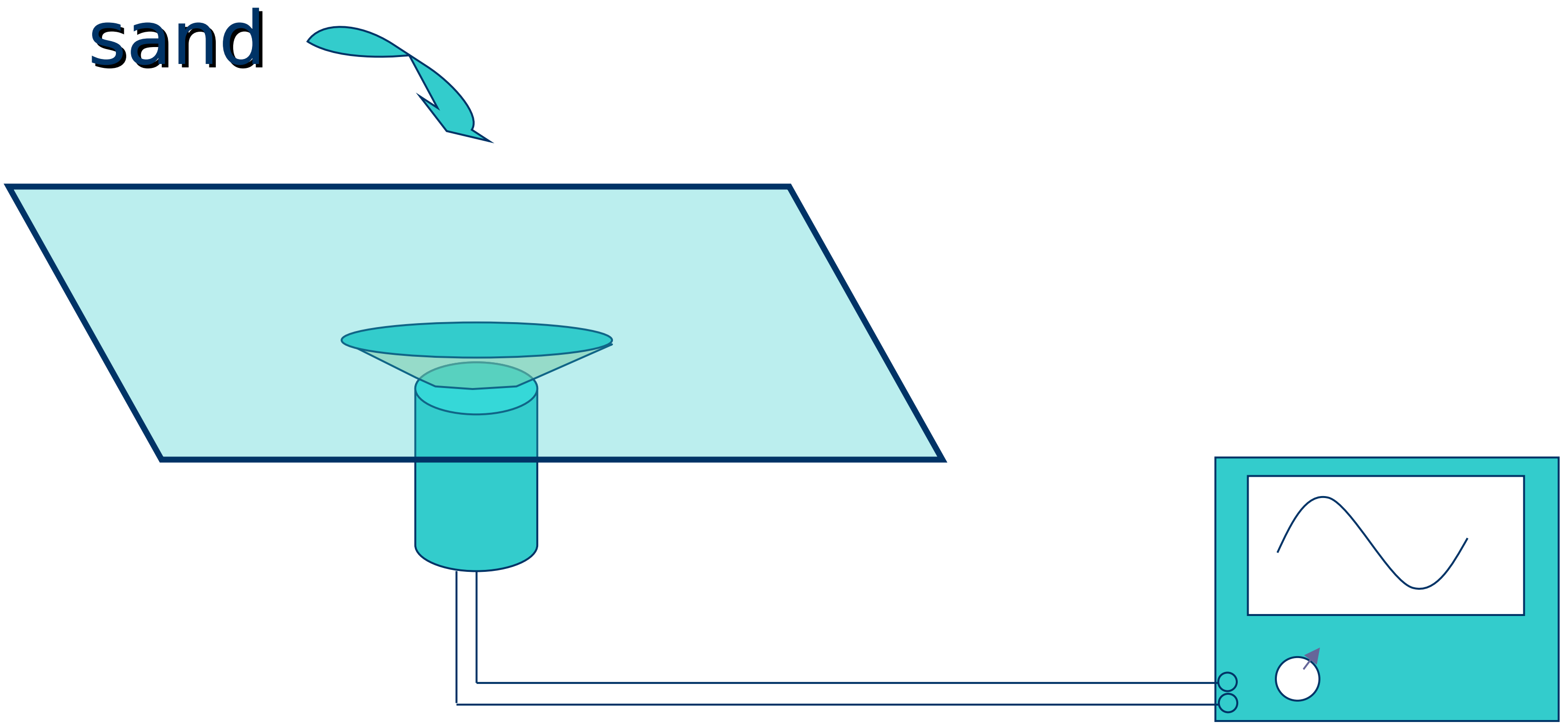
Harmonics →

?

Square Harmonics

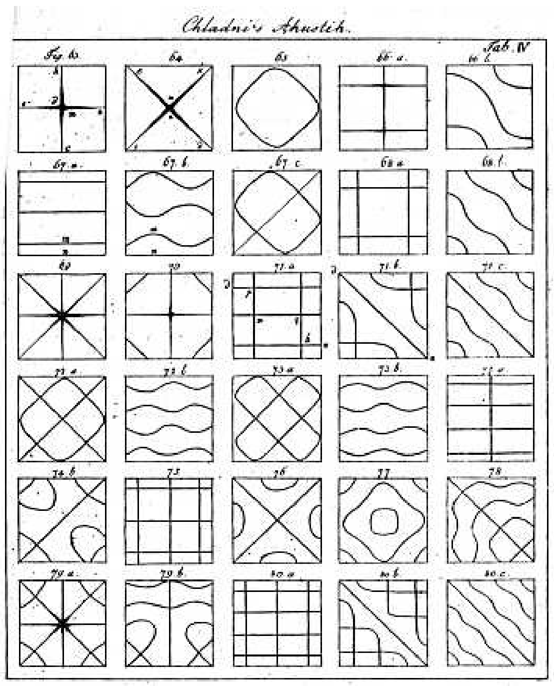


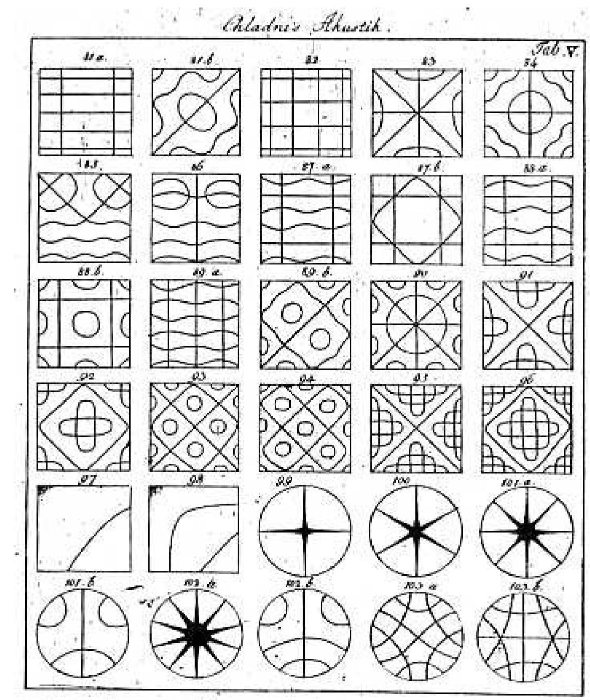
Chladni plates



Chladni plates

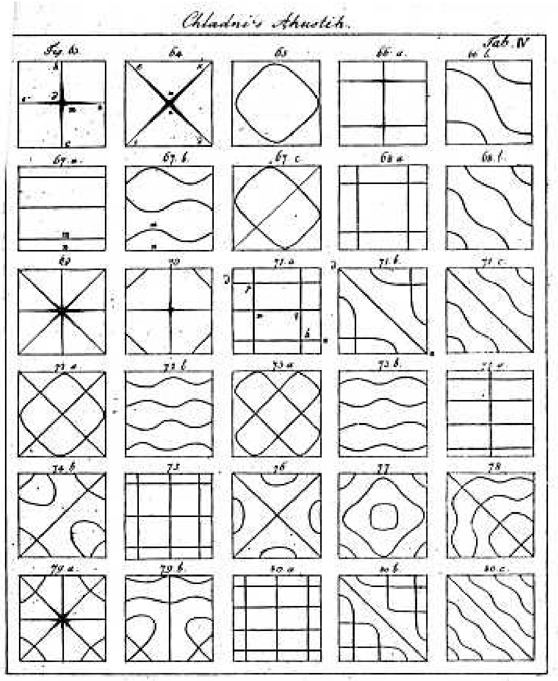
Chladni plates

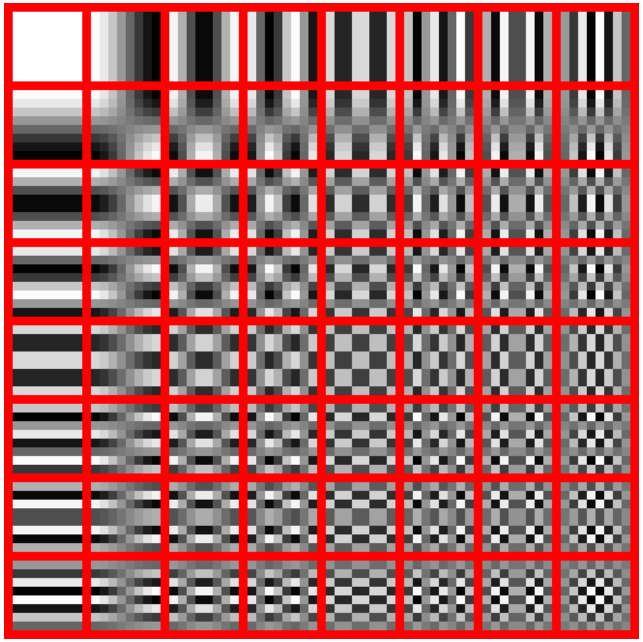




Discoveries concerning the theory of music, Chladni, 1787

Chladni plates and jpeg

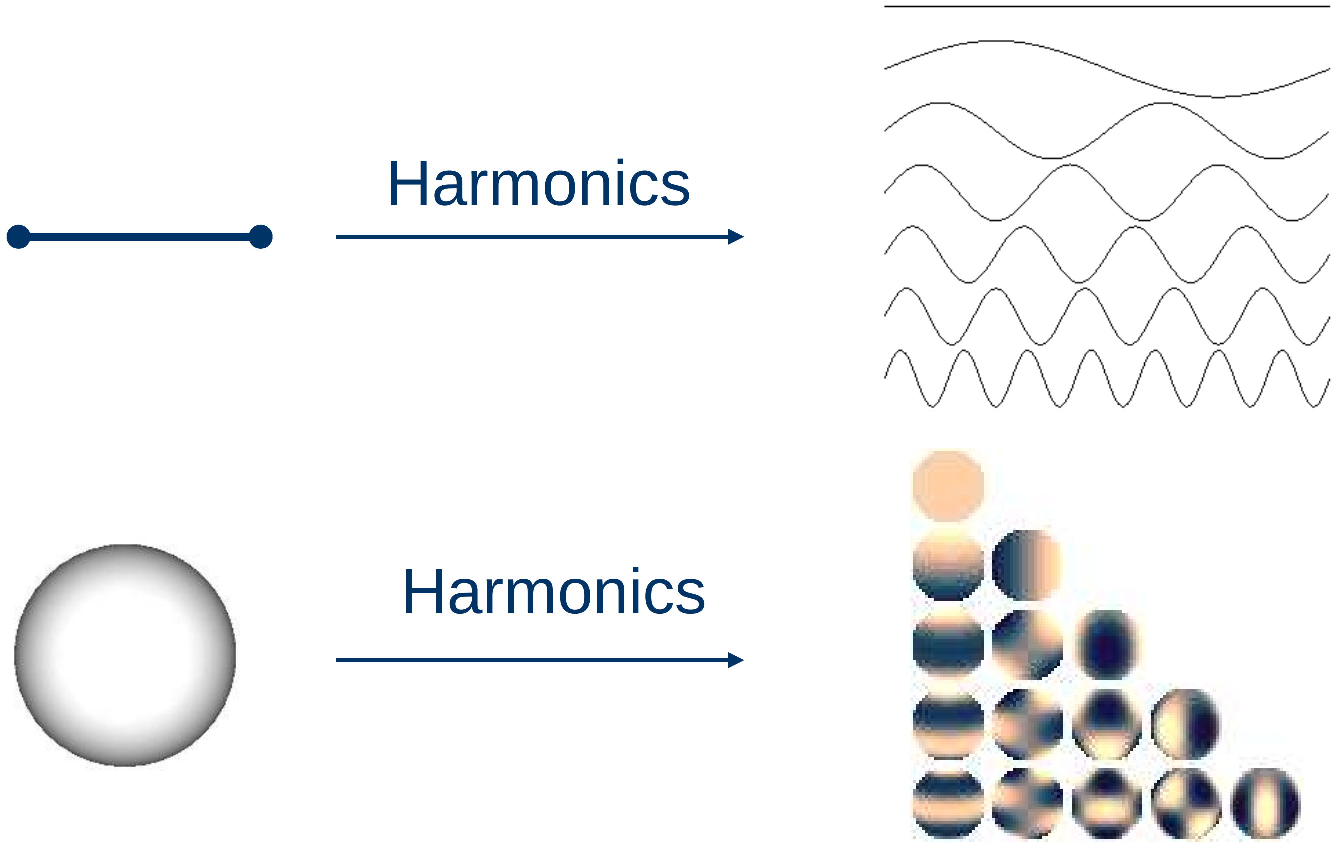




Chladni plates, 1787

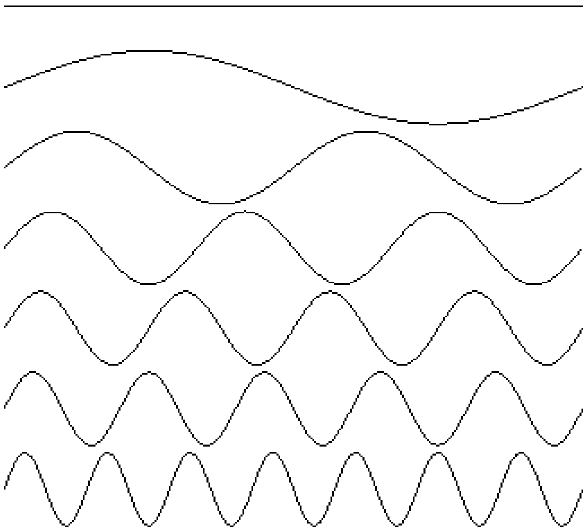
Discrete cosine transform (jpeg)

Spherical Harmonics



Manifold Harmonics

Harmonics



Harmonics

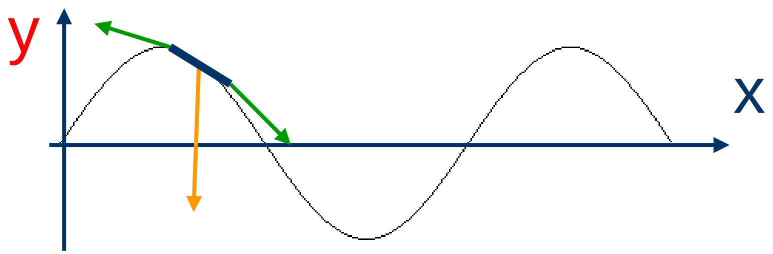
Stationary vibrating modes

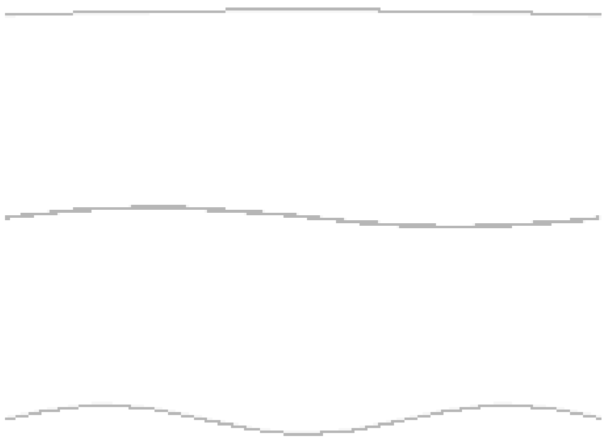
Harmonics and vibrations

- **Wave equation**:

T $\partial^2 y/\partial x^2 = \mu\ \partial^2 y/\partial t^2$

T: stiffness μ: mass





- **Stationary modes**:

$y(x,t) = y(x)\sin(\omega t)$

$\partial^2 y/\partial x^2 = -\mu\omega^2/T\ y$

eigenfunctions of $\partial^2/\partial x^2$

I Harmonics : recap

- Harmonics are **eigenfunctions** of $\partial^2/\partial x^2$
- On a mesh, $\partial^2/\partial x^2$ is the Laplacian Δ
- We need the **eigenfunctions** of Δ
- Let's use DEC

II DEC formulation

Introduction

- Harmonics
- **DEC formulation**
- Filtering
- Numerics

Results and conclusion

Discrete Exterior Calculus (DEC)

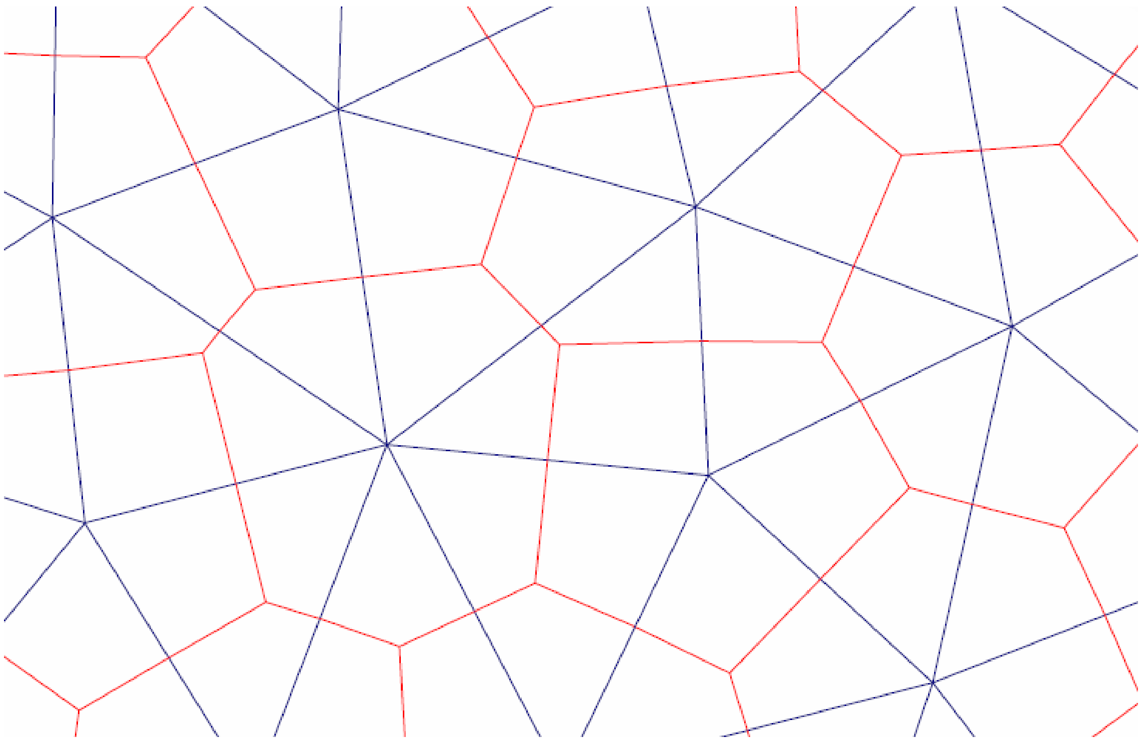
Discretize equations on a mesh
- Simple
- Rigorous

[Mercat], [Hirani], [Arnold], [Desbrun]

Based on k-forms

II DEC formulation

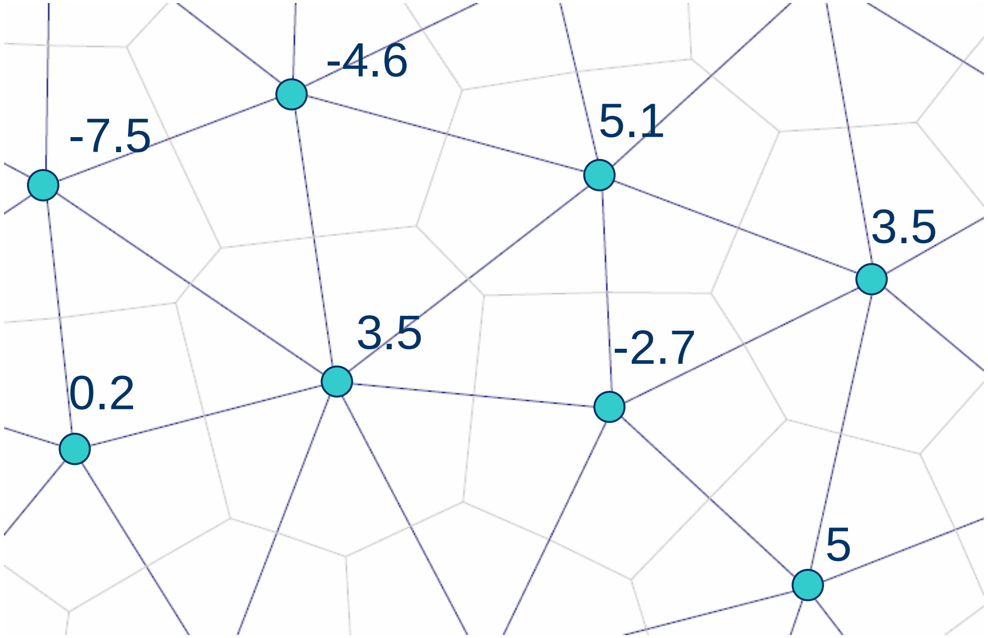
k-forms



mesh
dual mesh

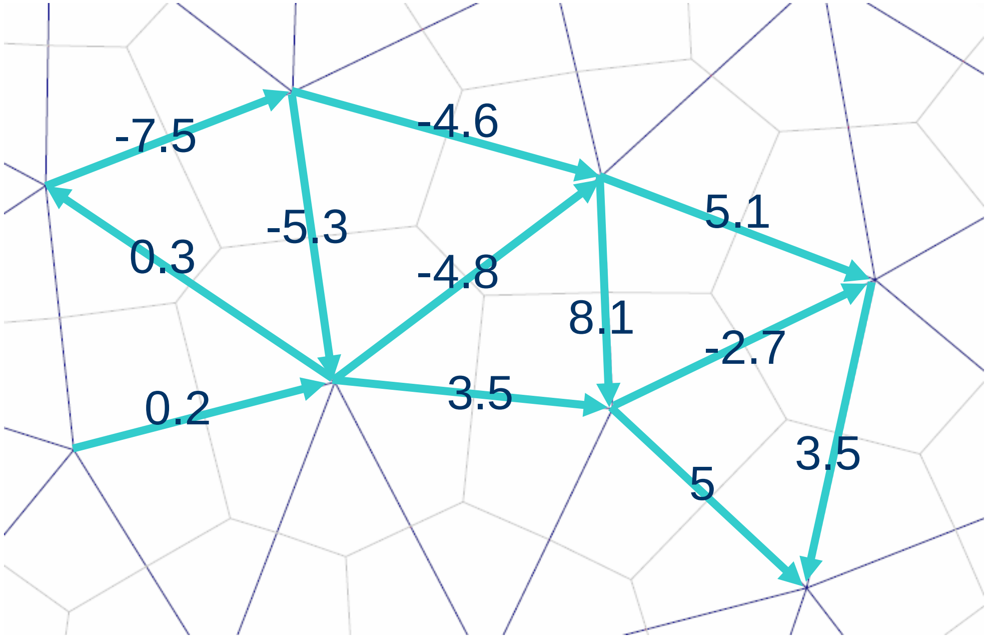
II DEC formulation

0-forms



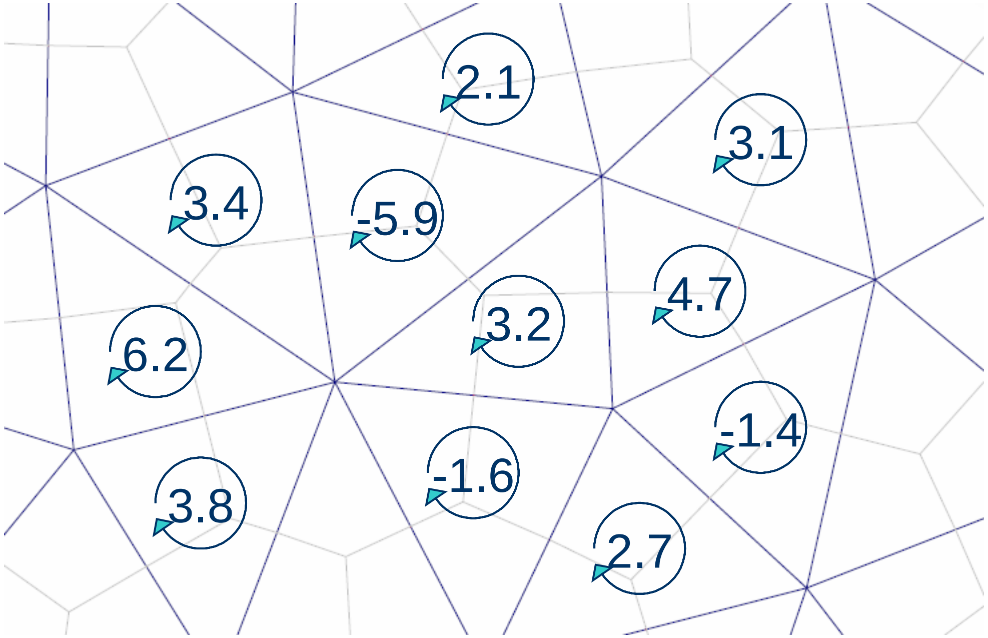
II DEC formulation

1-forms



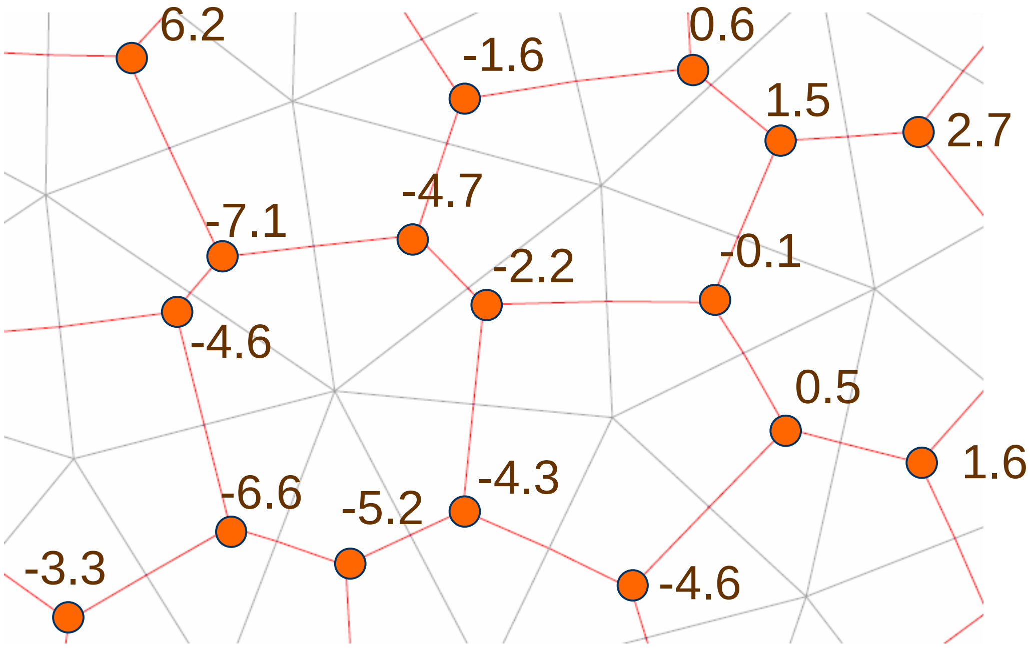
II DEC formulation

2-forms



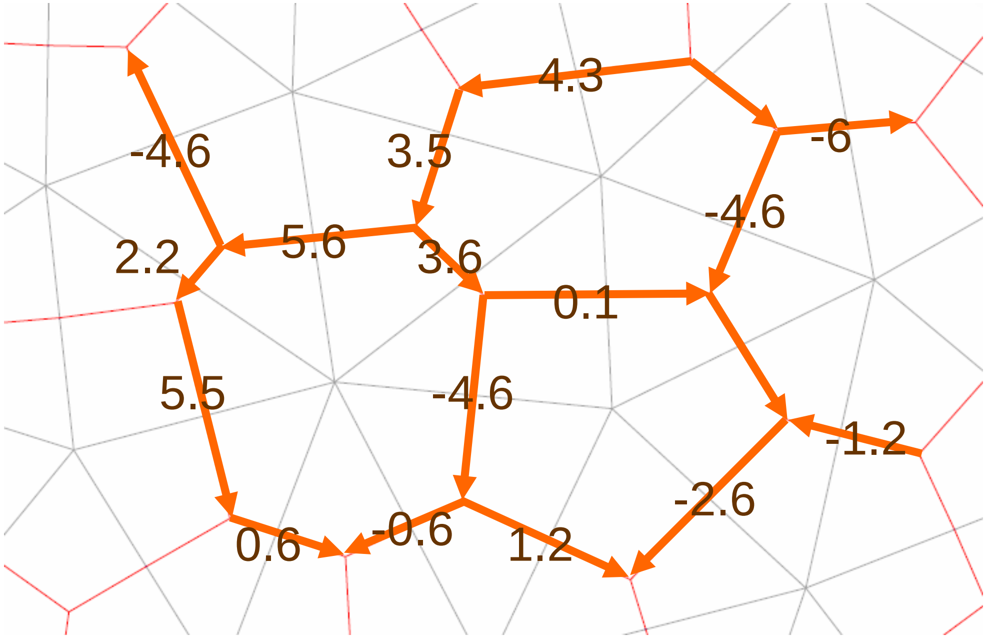
II DEC formulation

dual 0-forms



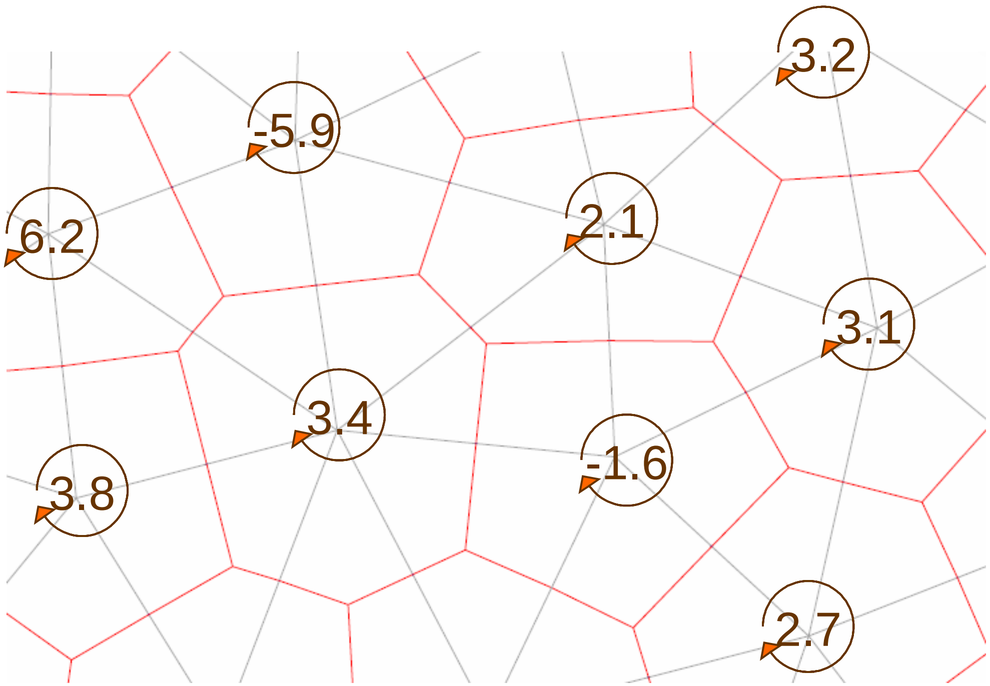
II DEC formulation

dual 1-forms



II DEC formulation

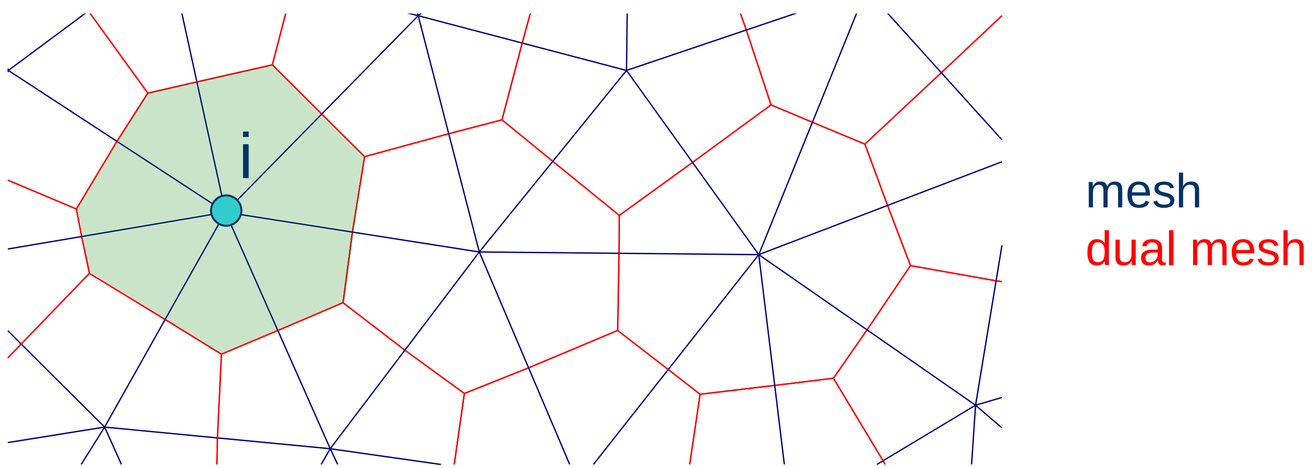
dual 2-forms



II DEC formulation

Hodge star $*_0$

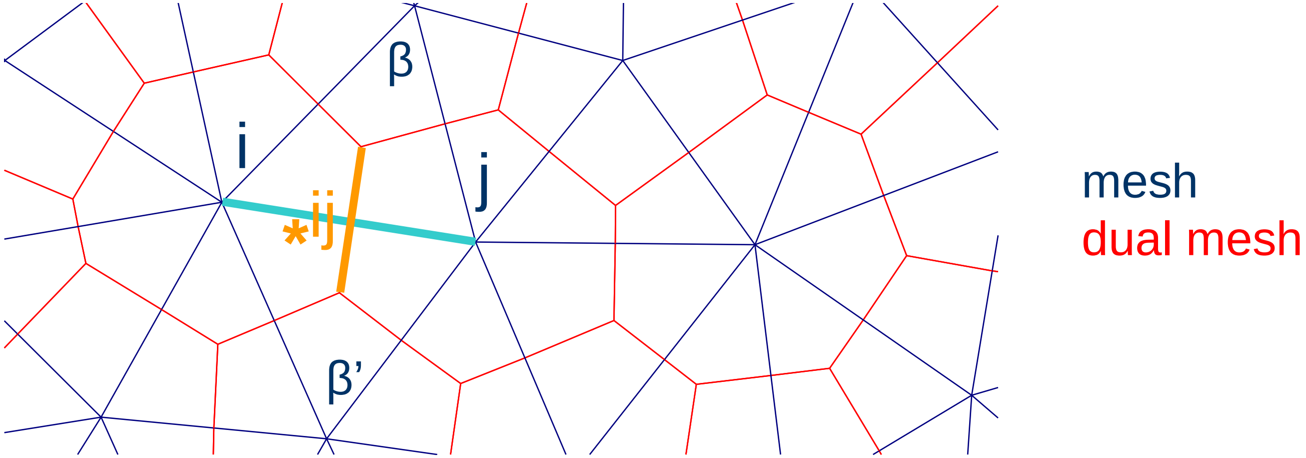
from	to	term
0-forms	dual 2-forms	$\lvert *i \rvert$



II DEC formulation

Hodge star $*_1$

from	to	term
1-forms	dual 1-forms	$|{*ij}|/|ij| = \cot(\beta)+\cot(\beta')$

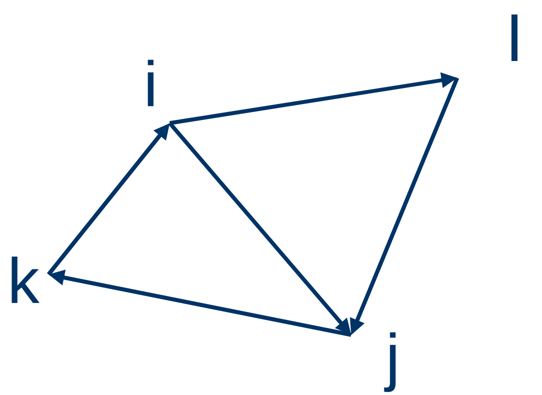


II DEC formulation

Exterior derivative d

from	to	term
0-forms	1-forms	$df\,(ij) = f_i - f_j$

Oriented connectivity of the mesh:



d	i	j	k	l	f
ij	-1	+1	0	0	f_i
jk	0	-1	+1	0	f_j
ki	+1	0	-1	0	f_k
il	-1	0	0	+1	f_l
lj	0	+1	0	-1	

DEC Laplacian

In DEC the Laplacian is $*_0^{-1} \, d^T \, *_1 \, d$

0-form (function) **f**

II DEC formulation

DEC Laplacian

In DEC the Laplacian is $*_0^{-1} \, d^T \, *_1 \, d$

0-form (function) **f** → (d) → 1-form (gradient) d**f**

$(\mathbf{f}_j - \mathbf{f}_i)$

DEC Laplacian

In DEC the Laplacian is $*_0^{-1} d^T *_1 d$

0-form (function) **f**

$\xrightarrow{d}$

1-form (gradient) d**f**

$\downarrow *_1$

$(\cot(\beta)+\cot(\beta'))(f_j - f_i)$

dual 1-form (cogradient) $*$d**f**

DEC Laplacian

In DEC the Laplacian is $*_0^{-1}\, d^T\, *_1\, d$

0-form (function) **f**

$\xrightarrow{\ d\ }$

1-form (gradient) d**f**

$\downarrow *_1$

$\Sigma(\cot(\beta)+\cot(\beta'))(\mathbf{f}_j-\mathbf{f}_i)$

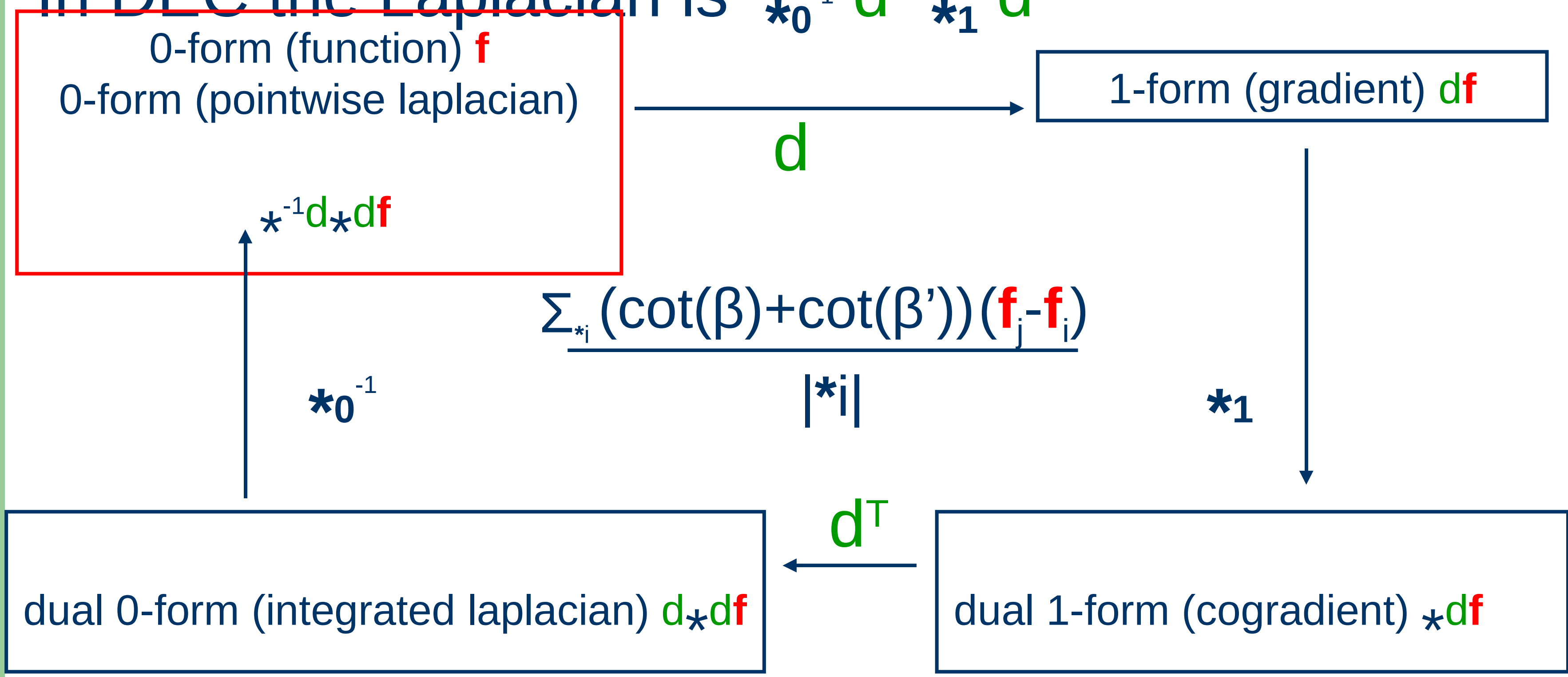
dual 1-form (cogradient) $*$d**f**

$\xleftarrow{\ d^T\ }$

dual 0-form (integrated laplacian) d$*$d**f**

DEC Laplacian

In DEC the Laplacian is $*_0^{-1}\, d^T\, *_1\, d$



Manifold Harmonics Basis (MHB)

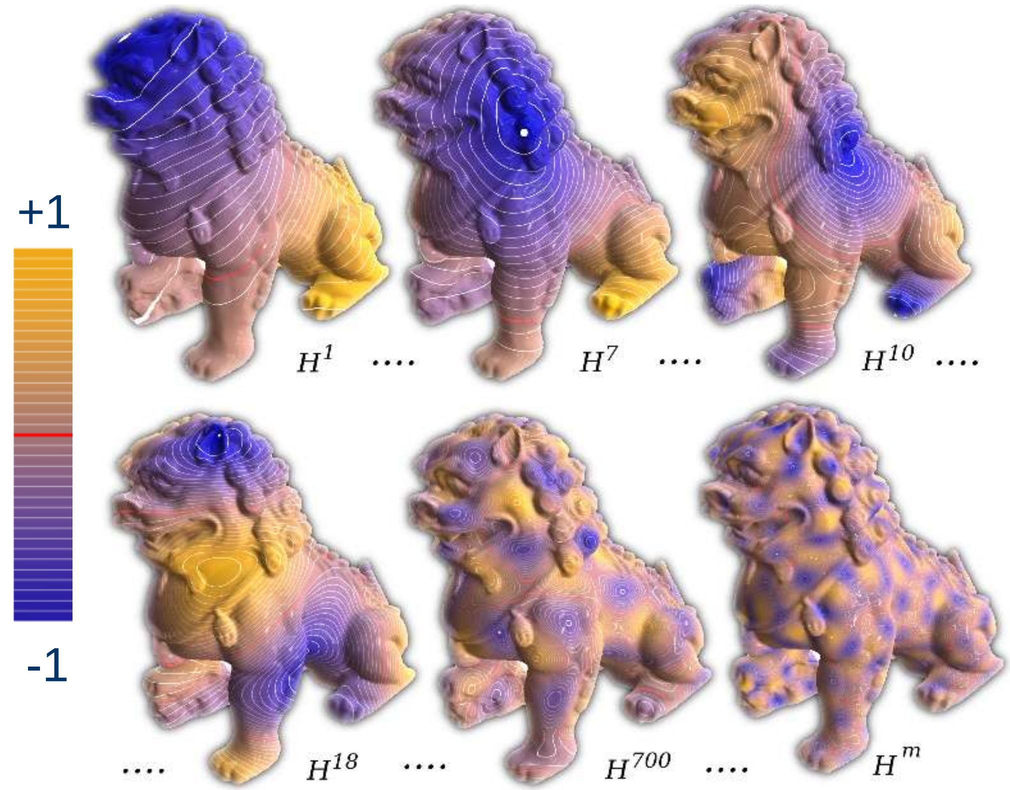
Eigenfunctions of

operator Δ

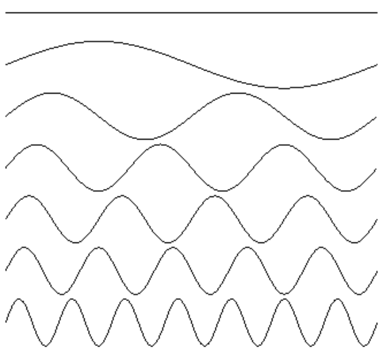
↓ **DEC**

Eigenvectors of

matrix $*_0^{-1} d^T *_1 d$



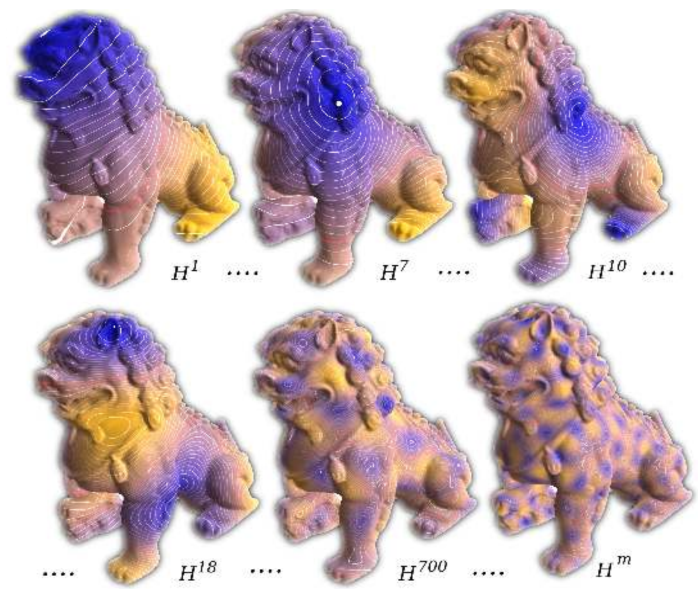
II DEC formulation : recap



sin(kx)

on

=



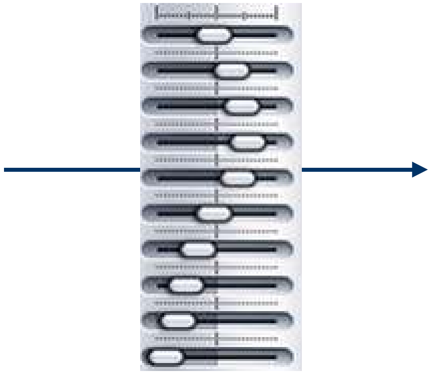
III Filtering

Introduction

- Harmonics
- DEC formulation
- **Filtering**
- Numerics

Results and conclusion

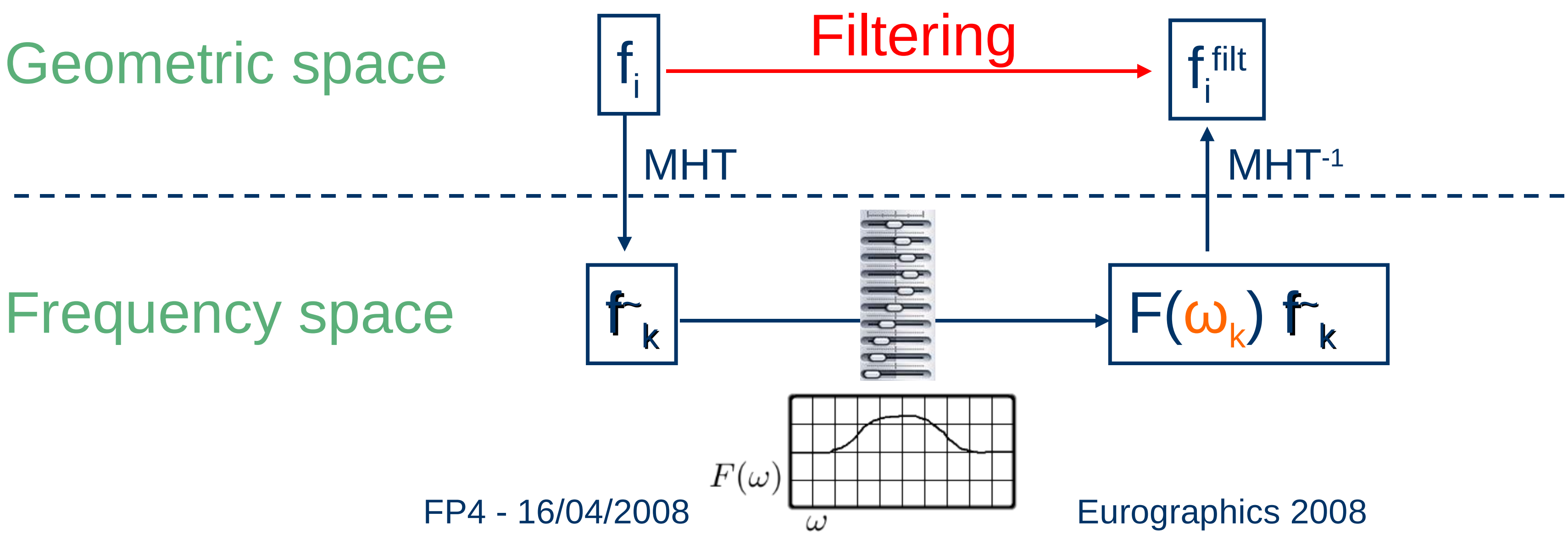
III Filtering



?

Spectral Filtering

- The Manifold Harmonics H^k come with an eigenvalue λ_k
- The $\lambda_k = \omega_k^2$ is a squared spatial frequency
- A filter is a transfer function $F(\omega)$

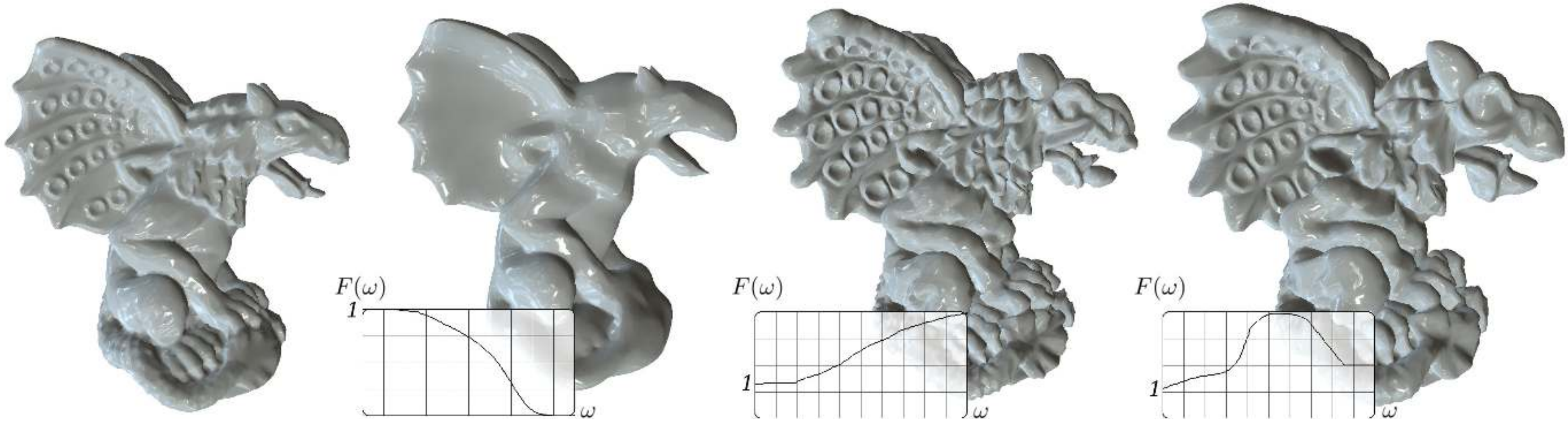


Color Filtering

Take **f** =(r,g,b)

Geometry Filtering

Take **f** =(x,y,z)



IV Numerics

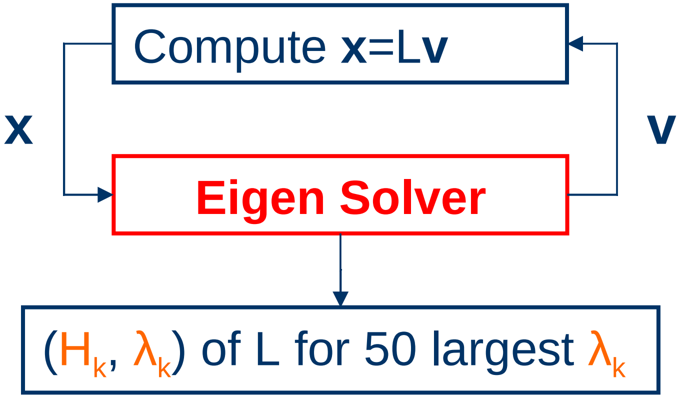
Introduction

- Harmonics
- DEC formulation
- Filtering
- **Numerics**

Results and conclusion

Eigenvalues

- Compute the eigenpairs (H_k, λ_k) of $L = \star_0^{-1} d^T \star_1 d$
- Solver returns eigenvectors of **highest** eigenvalue



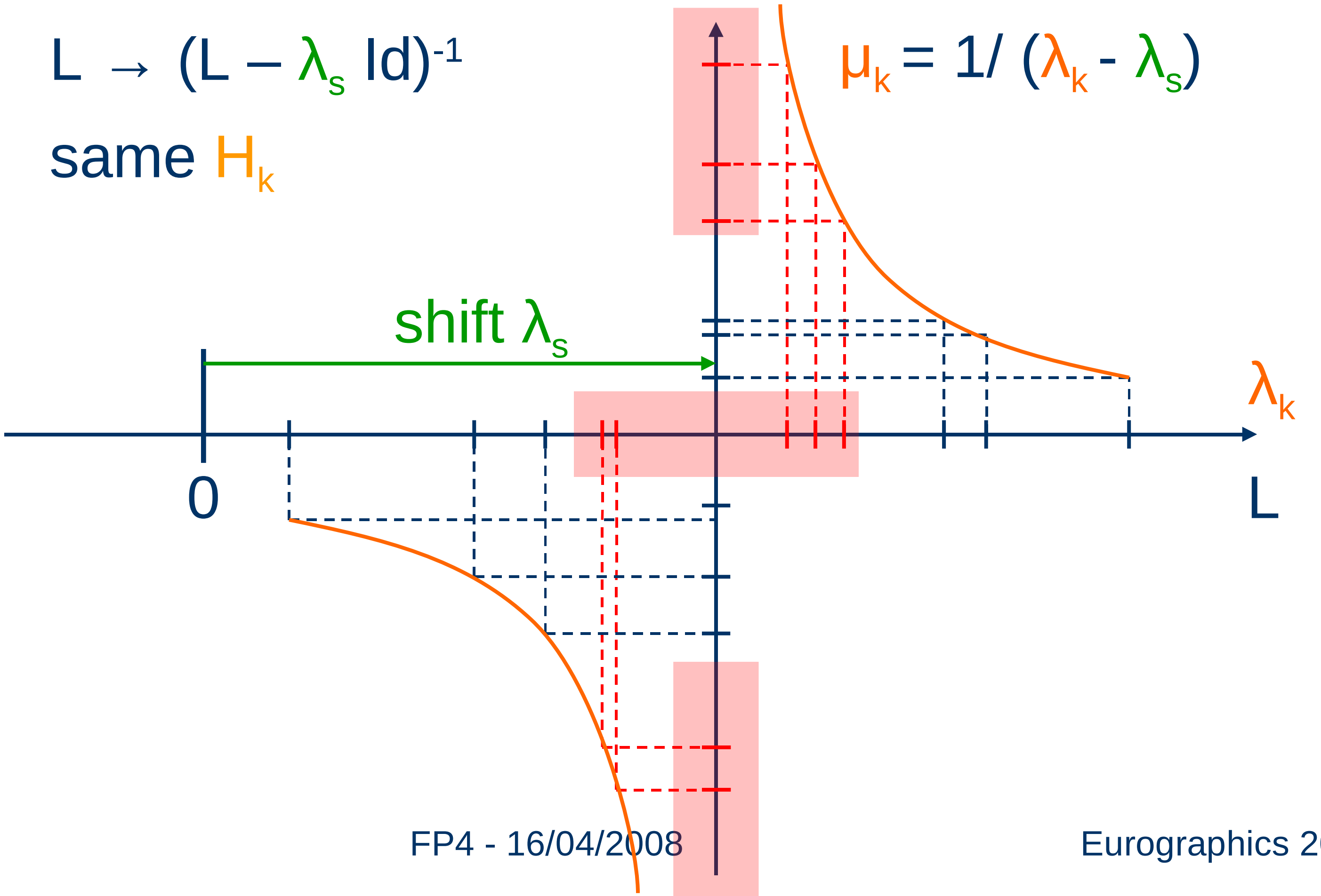
Problem:

•We want smallest λ_k

•We want more than 50

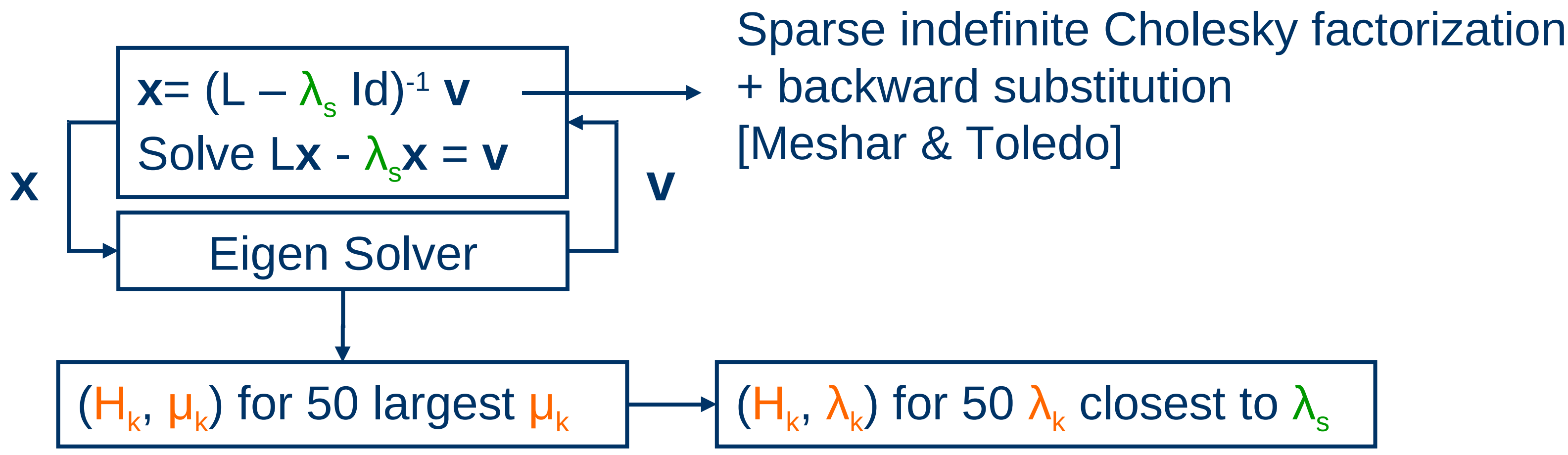
Shift Invert

- $L \rightarrow (L - \lambda_s \, Id)^{-1}$
- same H_k



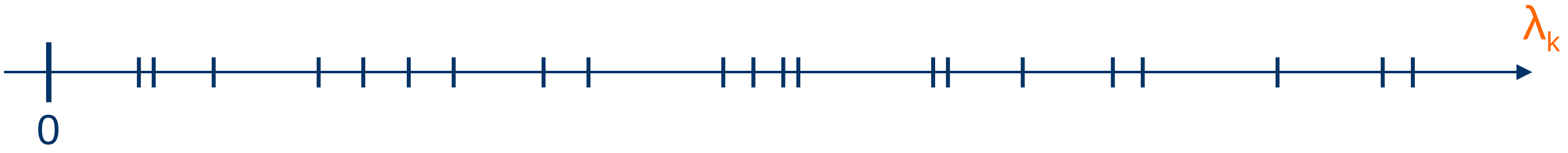
Eigen solver

Compute a **band** of eigenpairs (H^k, λ^k) around λ_s



Band by band algorithm

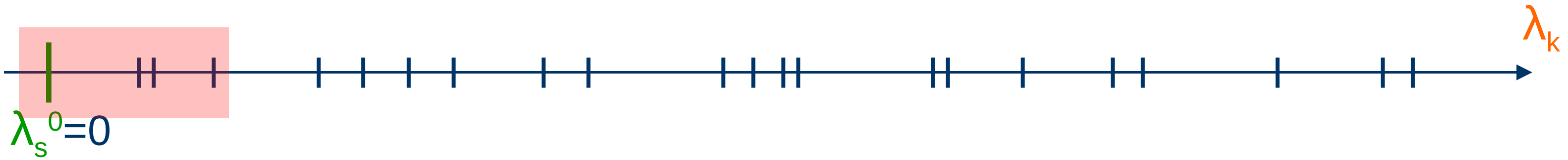
Compute the eigenpairs (h^k,λ^k) of L **band by band**



IV Numerics

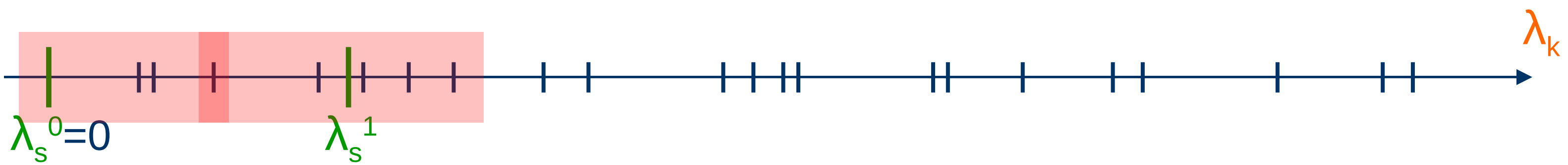
Band by band algorithm

Compute the eigenpairs (h^k,λ^k) of L **band by band**



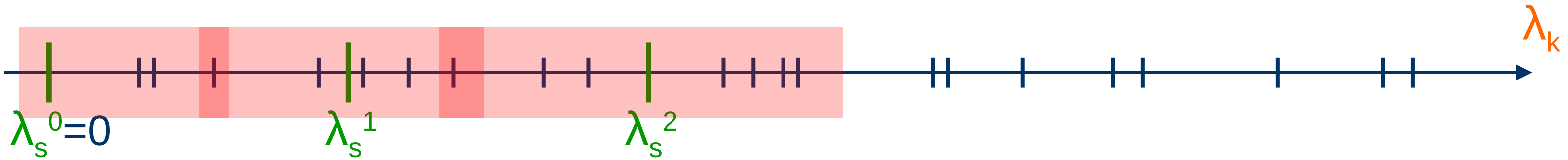
Band by band algorithm

Compute the eigenpairs (h^k,λ^k) of L **band by band**



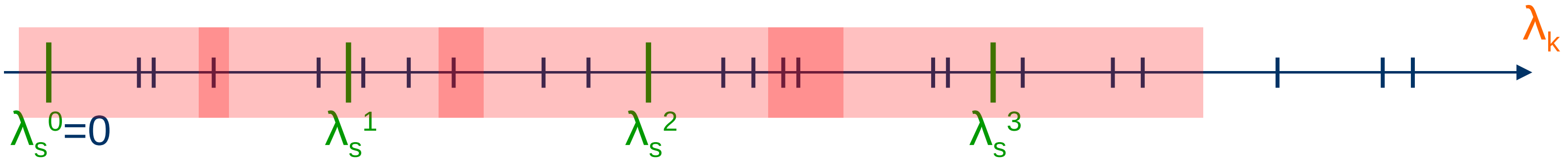
Band by band algorithm

Compute the eigenpairs (h^k,λ^k) of L **band by band**



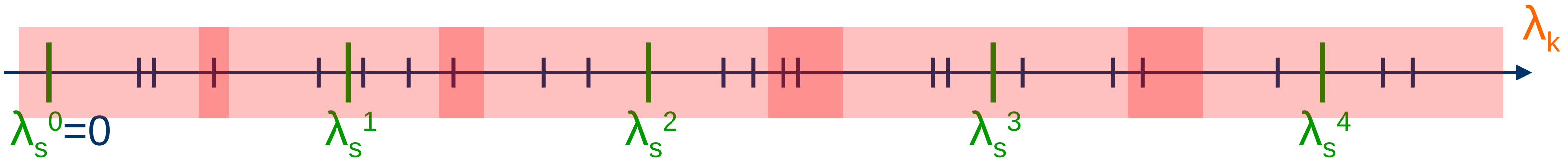
Band by band algorithm

Compute the eigenpairs (h^k,λ^k) of L **band by band**



Band by band algorithm

Compute the eigenpairs (h^k,λ^k) of L **band by band**



Results and conclusion

Introduction

II. Harmonics

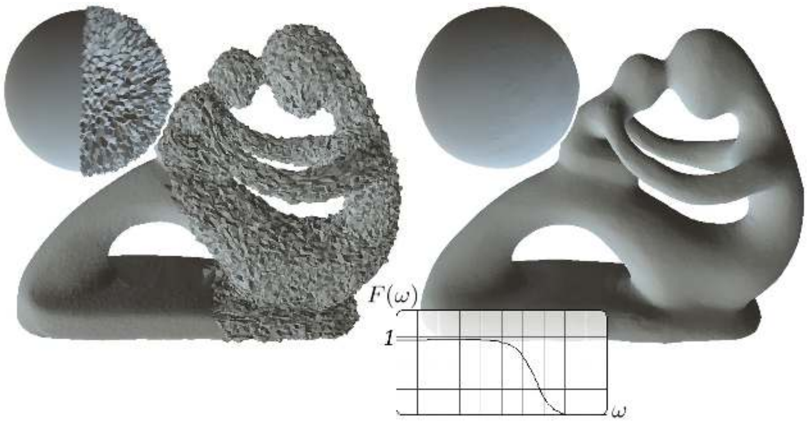
III. DEC formulation

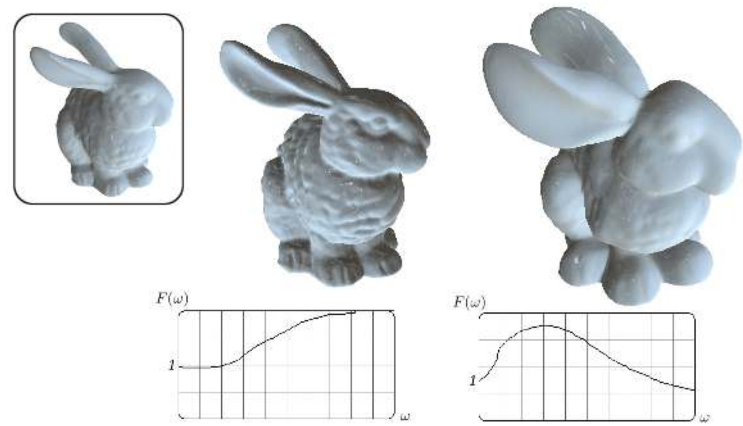
IV. Filtering

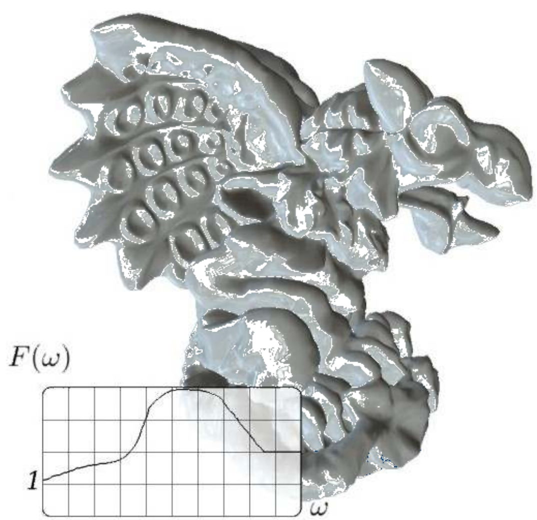
V. Numerics

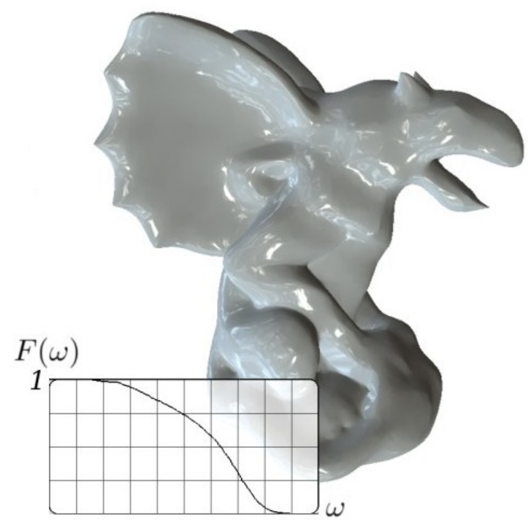
Results and conclusion

Results









Conclusion

We make explicit Fourier Analysis and Filtering tractable

Time to compute MHB ~ Time to compute a filter

(5 minutes for 300k vertices)

Time to update filter ~ real time

Acknowledgements

- **Ramsay Dyer** for personal communication
- **Sivan Toledo** for the sparse indefinite Cholesky factorization code

Questions ?