Lecture 3: Function Spaces I

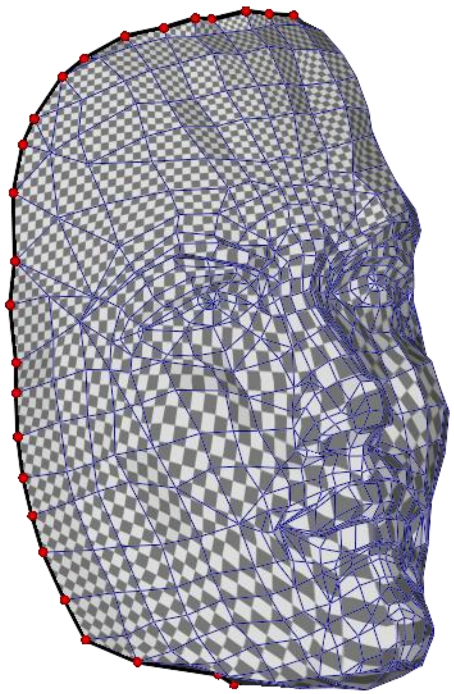
Finite Elements Modeling

Bruno Lévy

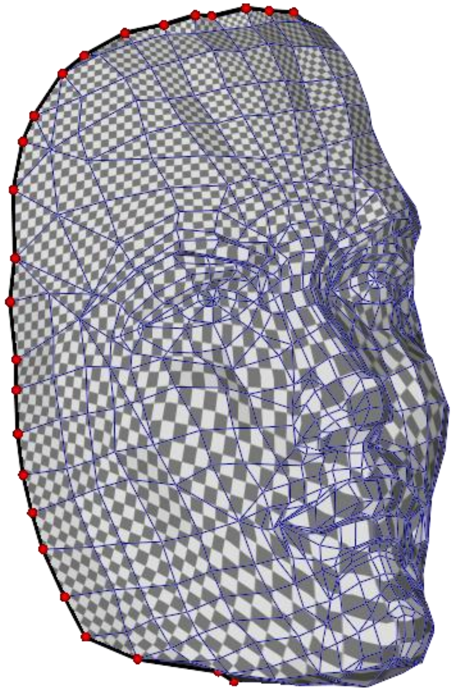
Overview

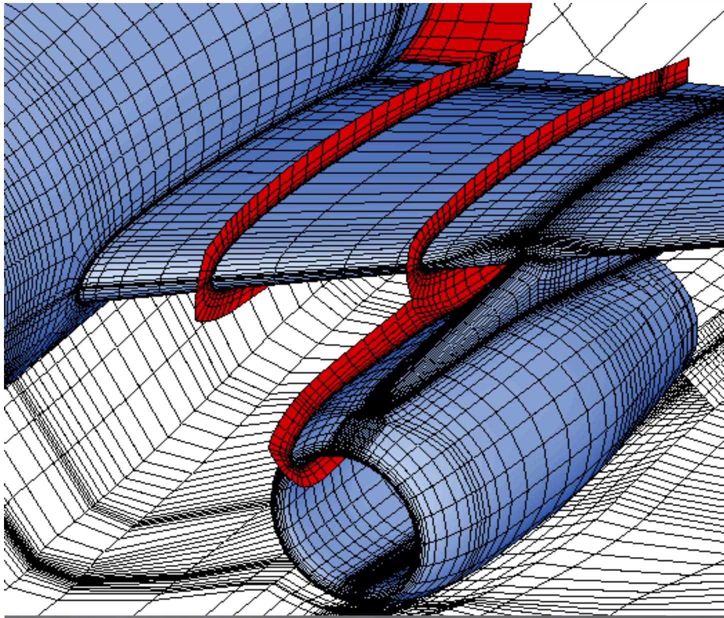
- 1. Motivations
- 2. Function Spaces
- 3. Discretizing a PDE
- 4. Example: Discretizing the Laplacian
- 5. Eigenfunctions → Spectral Mesh Processing(Richard)
- Epilogue – Continuous or Discrete ?

Computing with meshes

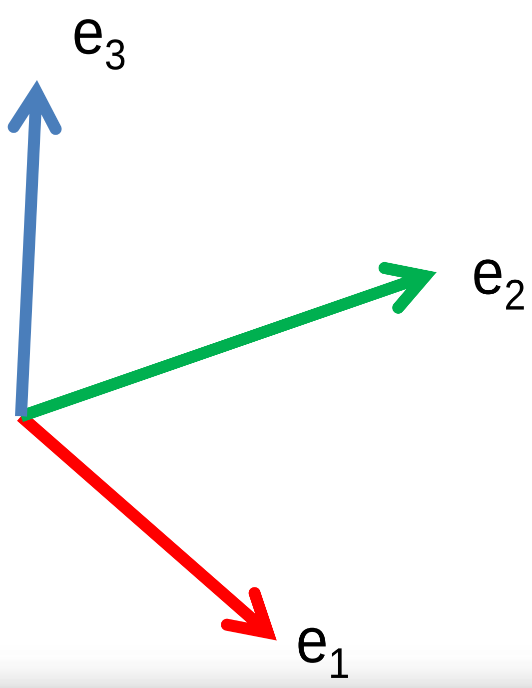


Computing with meshes



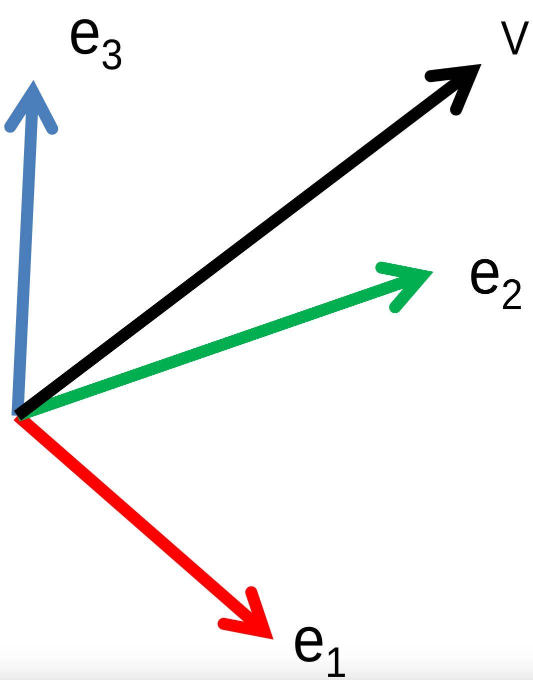


Vector spaces

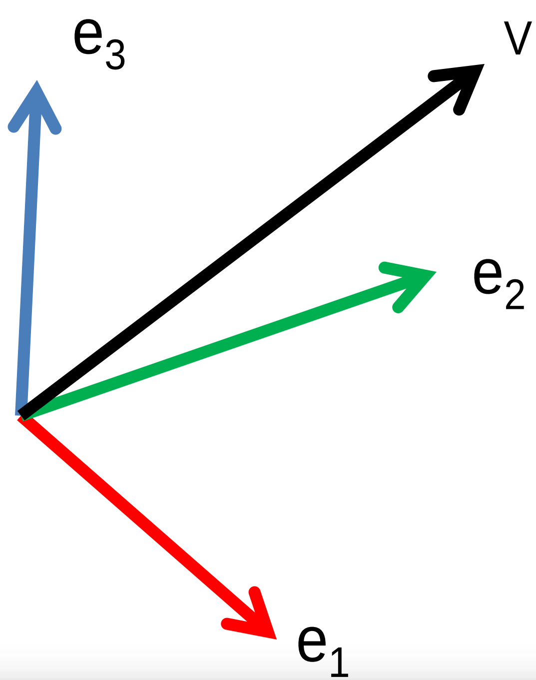


Vector spaces - coordinates

$$V = x\, e_1 + y\, e_2 + z\, e_3$$



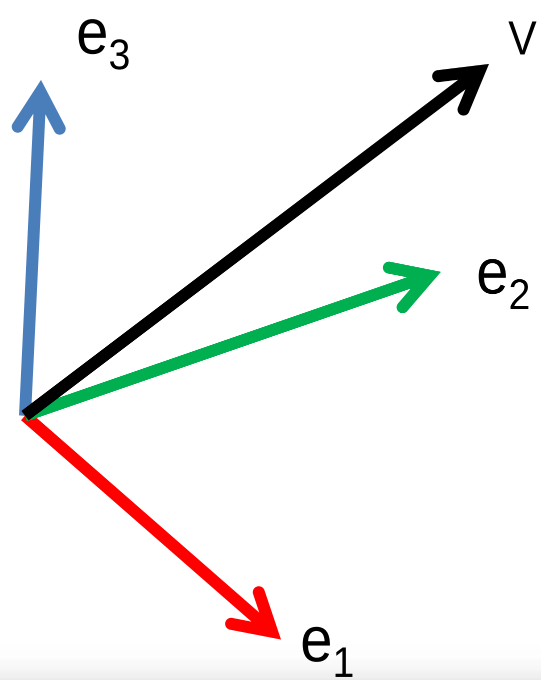
Vector spaces - coordinates



$$V = x\,e_1 + y\,e_2 + z\,e_3$$

$$x = V \,.\, e_1$$
$$y = V \,.\, e_2$$
$$z = V \,.\, e_3$$

Vector spaces - coordinates



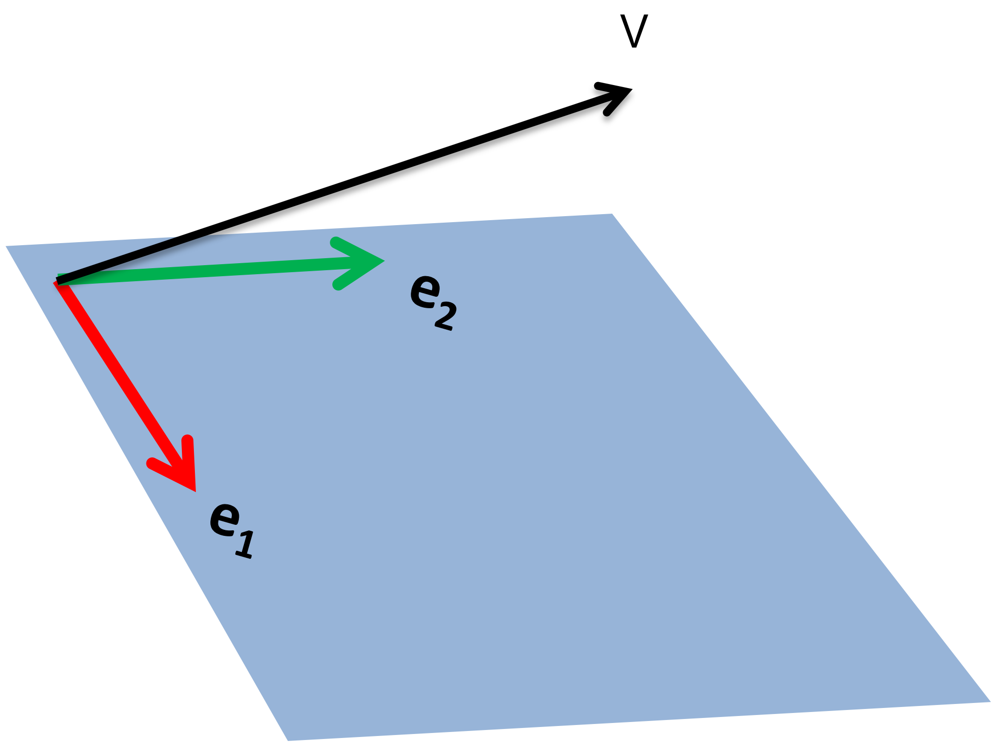
$$V = x\, e_1 + y\, e_2 + z\, e_3$$

$$x = V \,.\, e_1$$
$$y = V \,.\, e_2$$
$$z = V \,.\, e_3$$

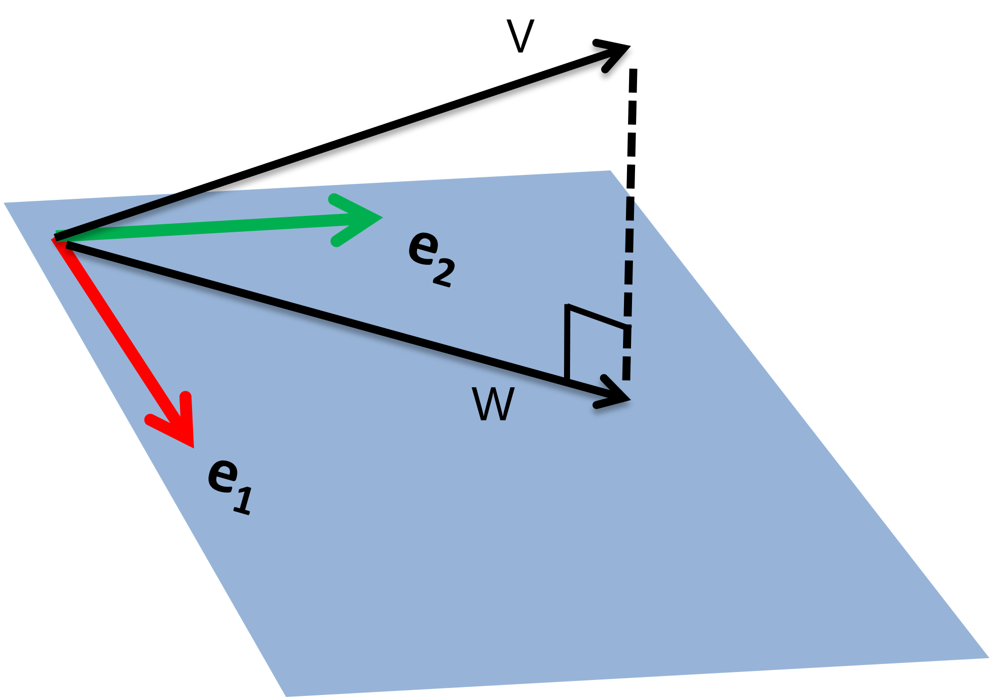
Dot product:

$$V \,.\, W = V_x W_x + V_y W_y + V_z W_z$$

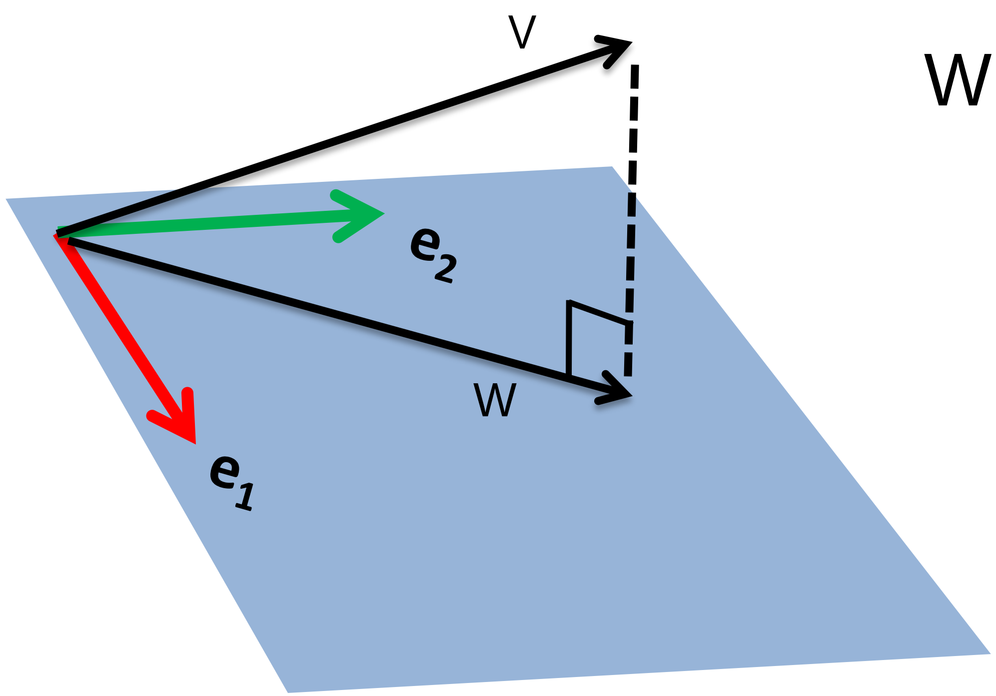
Vector spaces - projection



Vector spaces - projections



Vector spaces - projections



$$W = (V.e_1)e_1 + (V.e_2)e_2$$

Vector spaces – importance of the dot product

The dot product can compute

- Coordinates of a vector in a basis
- Projection of a vector onto a subspace
- Length of a vector = $\sqrt{v.v}$

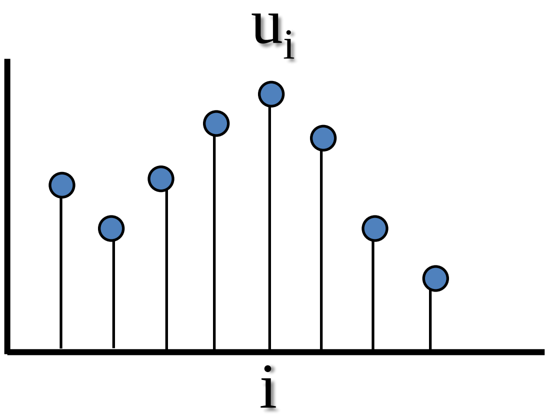
Vector spaces – importance of the dot product

<u>The dot product can compute</u>

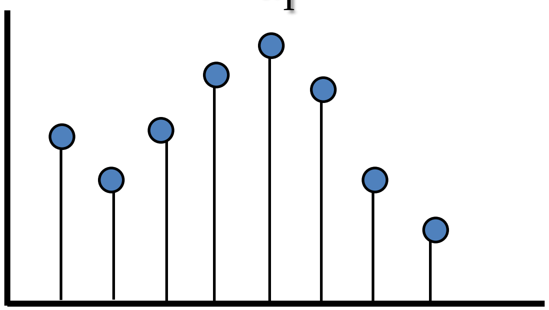
- Coordinates of a vector in a basis
- Projection of a vector onto a subspace
- Length of a vector = $\sqrt{v.v}$

→ Hilbert space

Dot product for vectors

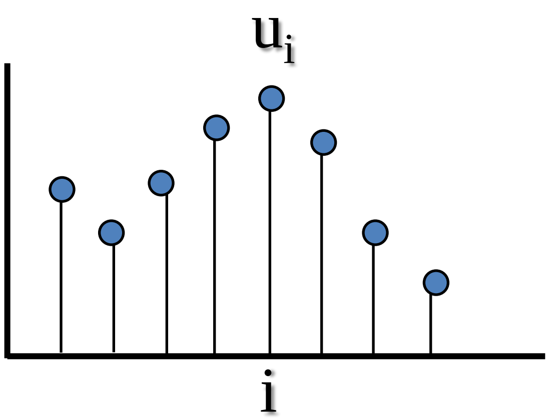


Dot product for vectors



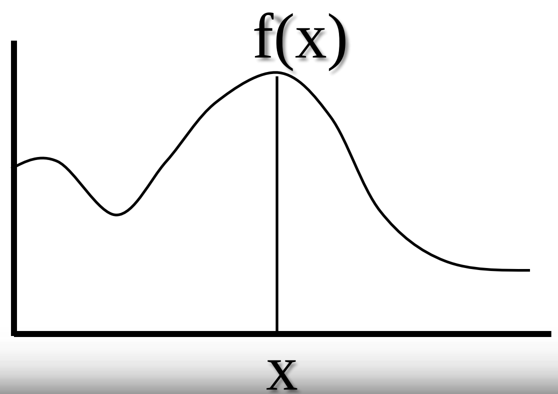
$$u \cdot v = \sum u_i v_i$$

Dot product for vectors



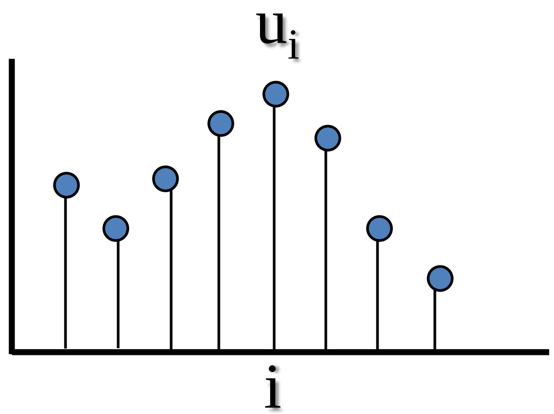
$$u \cdot v = \sum u_i v_i$$

Dot product for functions



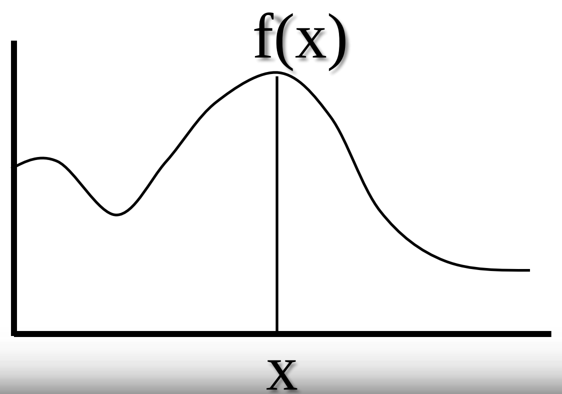
2. Function spaces

Dot product for vectors



$$u \cdot v = \sum u_i v_i$$

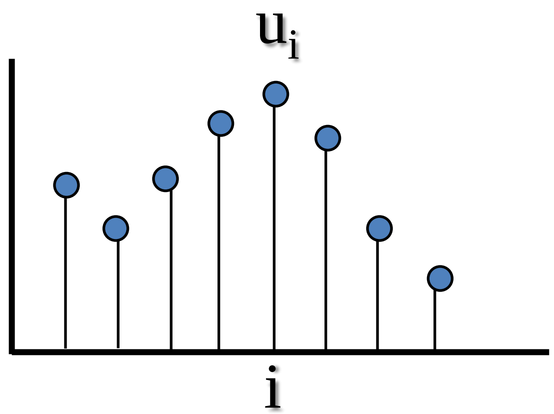
Dot product for functions



$$\langle f,g \rangle = \int f(t)g(t)dt$$

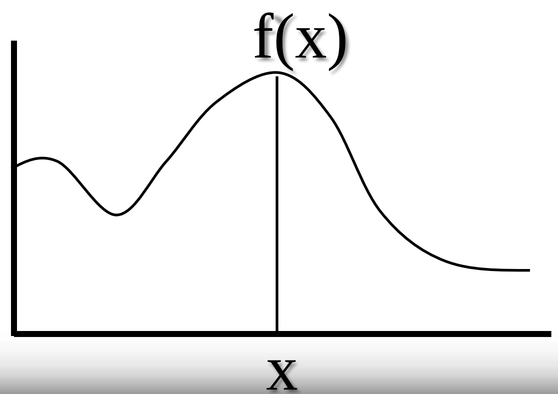
2. Function spaces

Dot product for vectors



$$u \cdot v = \sum u_i v_i$$

Dot product for functions



$$\langle f,g \rangle = \int f(t)g(t)dt$$

The inner product (dot product for functions)

Vector spaces – importance of the dot product

The inner (“dot”) product can compute

- Coordinates of a function in a function basis
- Projection of a function onto a subspace
- “Norm” of a function = $\sqrt{\langle f,f \rangle}$

2. Function spaces

Examples of function basis

The canonical polynomial basis (ϕi)

$\phi_0(x) = 1$

$\phi_1(x) = x$

$\phi_2(x) = x^2$

$\phi_3(x) = x^3$

$$f(x) = \Sigma\ \alpha_i\ \phi_i(x)$$

Examples of function basis

The Fourier basis (ϕi)

$\phi_0(x) = 1$

$\phi_{2k}(x) \quad = \sin(2k\pi x)$

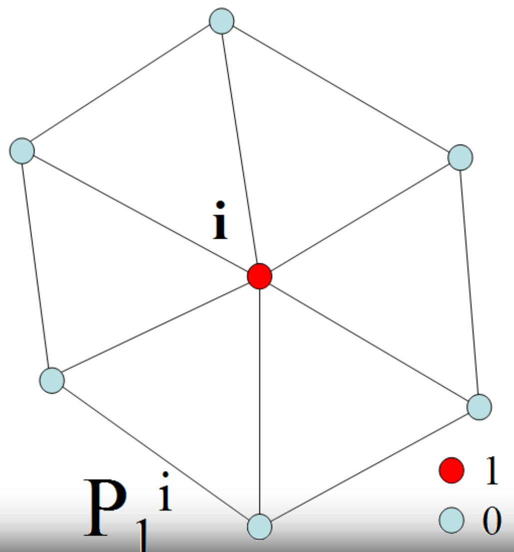
$\phi_{2k+1}(x) = \cos(2k\pi x)$

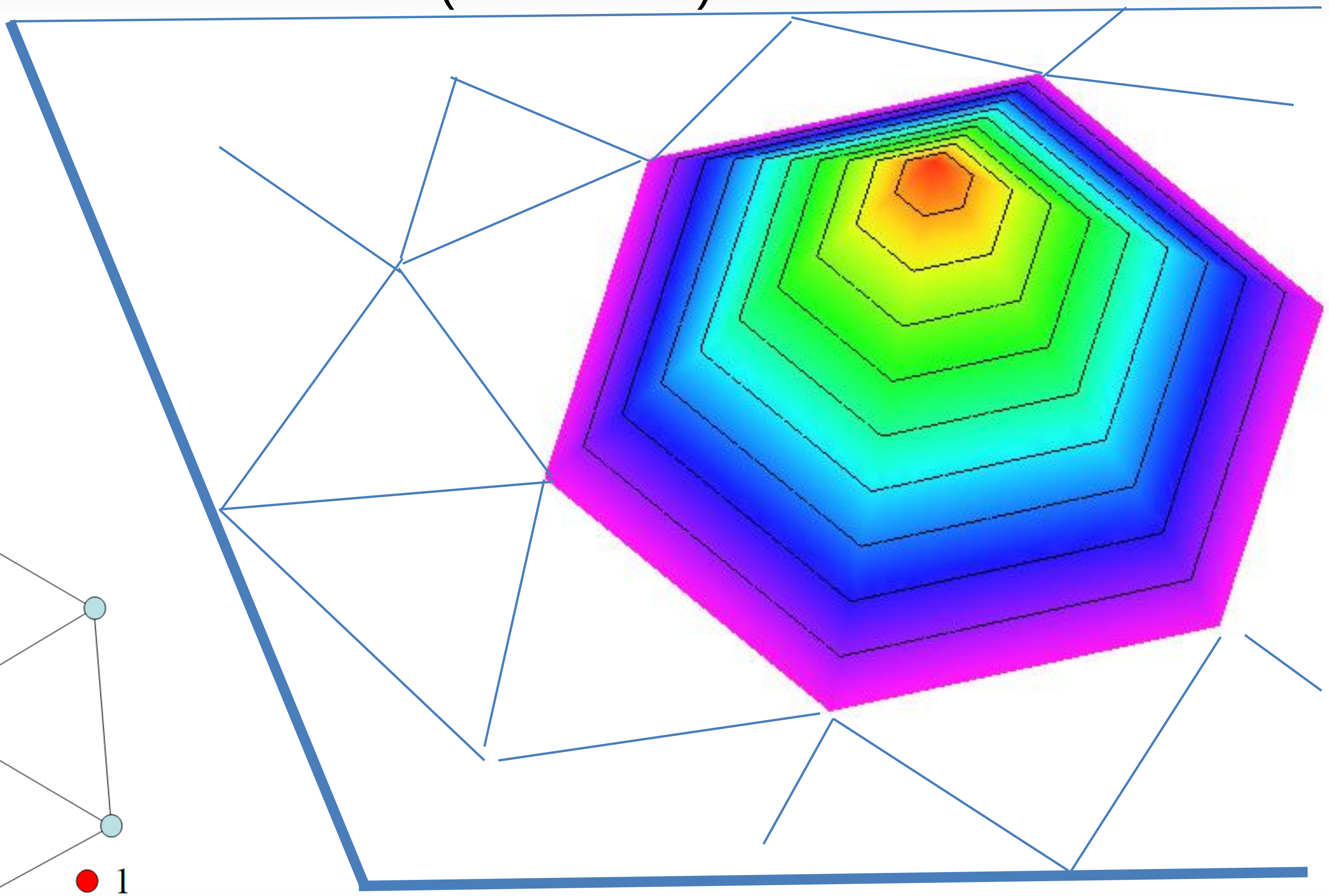
$$f(x) = \Sigma\, \alpha_i\, \phi_i(x)$$

2. Function spaces

Meshes – The P1 Function Basis (linear FEM)

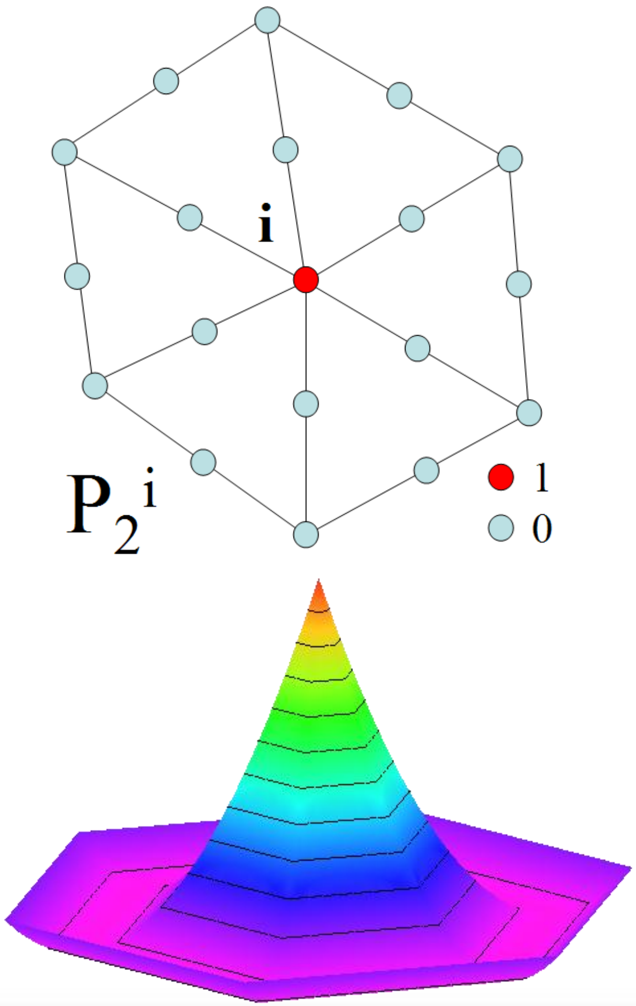
$\phi_i(p_i) = 1$
$\phi_i(p_j) = 0$

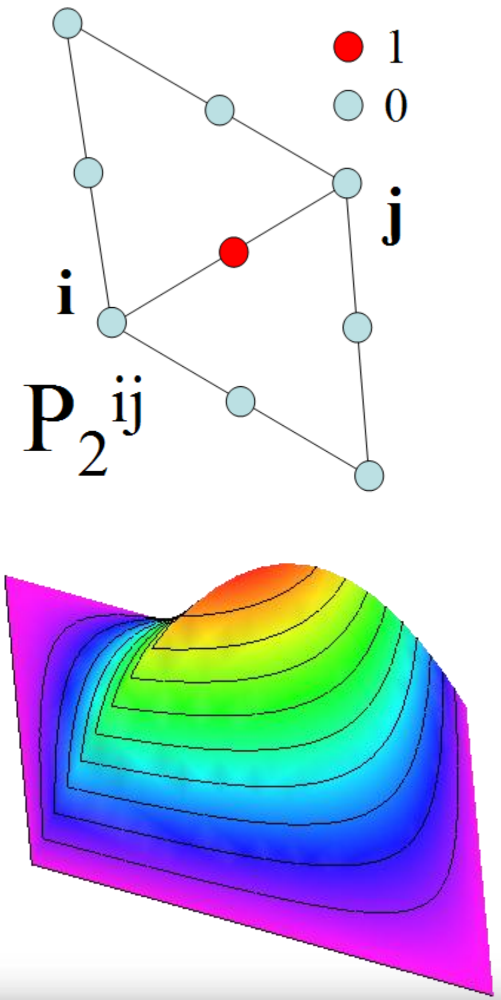




Courtesy of Bruno Vallet

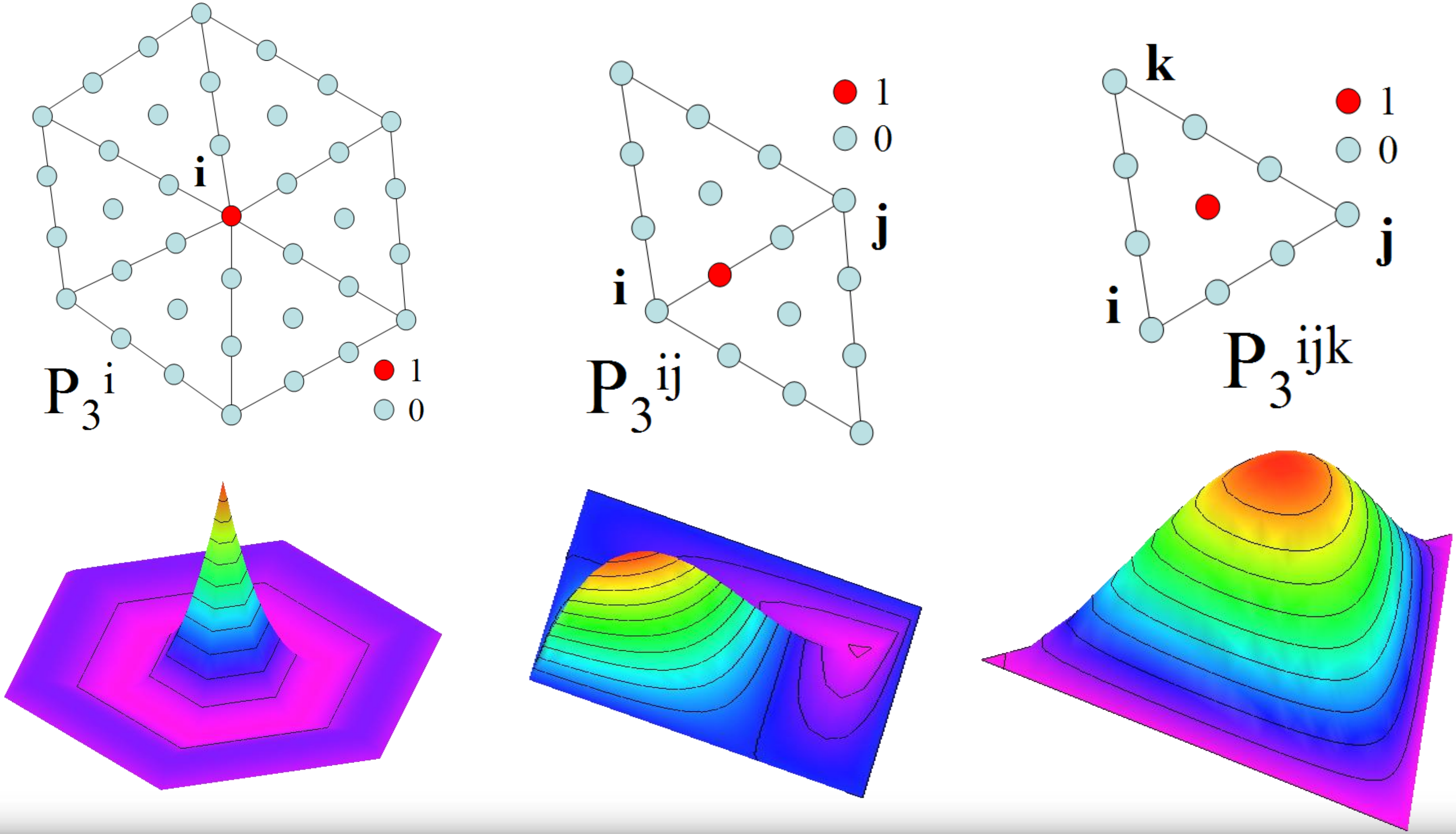
Meshes – The P2 Function Basis (quadratic FEM)





Courtesy of Bruno Vallet

Meshes – The P3 Function Basis (cubic FEM)



3. Discretizing a PDE

General form of a linear PDE:

$Lf = g$

General form of a linear PDE:

$Lf = g$

Linear operator

e.g., Δ

3. Discretizing a PDE

General form of a linear PDE:

$Lf = g$

Unknown function
(the one we want to compute)

3. Discretizing a PDE

General form of a linear PDE:

$Lf = g$

Known function
(right hand side)

3. Discretizing a PDE

General form of a linear PDE:

$Lf = g$

Weak form: $\forall h, \langle Lf, h \rangle = \langle g, h \rangle$

3. Discretizing a PDE

General form of a linear PDE:

$Lf = g$

Weak form: $\forall h, \langle Lf, h \rangle = \langle g, h \rangle$

Test function

3. Discretizing a PDE

General form of a linear PDE:

$Lf = g$

Weak form: $\forall h, \langle Lf, h \rangle = \langle g, h \rangle$

Test function

Intuitively, weak form: “Lf = g” projected onto all possible test functions

3. Discretizing a PDE

$$Lf = g$$

Discretization

Step 1: project PDE onto function basis (ϕi)

$\forall i, \langle Lf, \phi_i \rangle = \langle g, \phi_i \rangle \quad (1)$

3. Discretizing a PDE

$Lf = g$

Discretization

Step 1: project PDE onto function basis (ϕi)

$\forall i, \langle Lf, \phi_i \rangle = \langle g, \phi_i \rangle \quad (1)$

Step 2: express f in (ϕi): $f = \Sigma\, \alpha_j\, \phi_j$ and inject in (1)

$\forall i, \langle L\, \Sigma\, \alpha_j\, \phi_j, \phi_i \rangle = \langle g, \phi_i \rangle$

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i, \langle L \sum \alpha_j \phi_j, \phi_i \rangle = \langle g, \phi_i \rangle$$

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i, <L \sum \alpha_j \phi_j, \phi_i> = <g, \phi_i>$$

$\forall i, \sum \alpha_j <L \phi_j, \phi_i> = <g, \phi_i>$ (by linearlity of <.,.> and L)

3. Discretizing a PDE

$$Lf = g$$

Discretization

$\forall i,\ <L \Sigma\ \alpha_j\ \phi_j,\ \phi_i> = <g,\ \phi_i>$

$\forall i,\ \Sigma\ \alpha_j <L\ \phi_j,\ \phi_i> = <g,\ \phi_i>$ (by linearlity of <.,.> and L)

$\forall i,\ \Sigma\ \alpha_j <L\ \phi_j,\ \phi_i> = \Sigma\ \beta_j <\ \phi_j,\ \phi_i>$ (rhs $g = \Sigma\ \beta_j\ \phi_j$)

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i,\ \Sigma\, \alpha_j < L\, \phi_j, \phi_i > = \Sigma\, \beta_j < \phi_j, \phi_i >$$ Galerkin

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i,\ \Sigma\, \alpha_j \langle L\, \phi_j, \phi_i \rangle = \Sigma\, \beta_j \langle \phi_j, \phi_i \rangle$$ Galerkin

$$\begin{bmatrix} & & \\ & \langle L\phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & \langle \phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i,\ \Sigma\, \alpha_j \langle L\,\phi_j, \phi_i \rangle = \Sigma\, \beta_j \langle \phi_j, \phi_i \rangle \quad \text{Galerkin}$$

$$\begin{bmatrix} & & \\ & \langle L\phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & \langle \phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$

Discretization of
the operator L
(stiffness matrix)

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i,\ \sum \alpha_j \langle L\,\phi_j, \phi_i \rangle = \sum \beta_j \langle \phi_j, \phi_i \rangle$$ Galerkin

$$\begin{bmatrix} & & \\ & \langle L\phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & \langle \phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}$$

Discretization of the solution f (unknown vector)

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i, \ \Sigma \alpha_j \langle L \phi_j, \phi_i \rangle = \Sigma \beta_j \langle \phi_j, \phi_i \rangle$$ Galerkin

$$\begin{bmatrix} & & \\ & \langle L\phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & \langle \phi_i, \phi_j \rangle & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$

Discretization of the inner product (bilinear form)

3. Discretizing a PDE

$$Lf = g$$

Discretization

$$\forall i,\ \Sigma\, \alpha_j \langle L\,\phi_j, \phi_i\rangle = \Sigma\, \beta_j \langle \phi_j, \phi_i\rangle$$ Galerkin

$$\begin{bmatrix} \\ \langle L\phi_i, \phi_j\rangle \\ \\ \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} \\ \langle \phi_i, \phi_j\rangle \\ \\ \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$

Discretization of the rhs g (known vector)

4. Discretizing the Laplacian

$\Delta f = g$ (Poisson equation)

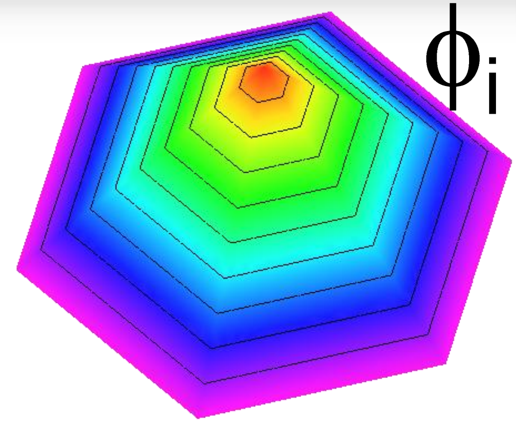
Discretization

$$\begin{bmatrix} \\ < \Delta\, \phi_i, \phi_j > \\ \\ \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} \\ <\phi_i, \phi_j> \\ \\ \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \\ \beta_n \end{bmatrix}$$

4. Discretizing the Laplacian

Discretization – P1

$$\begin{bmatrix} \\ < \Delta \phi_i, \phi_j > \\ \\ \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} \\ <\phi_i, \phi_j> \\ \\ \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$



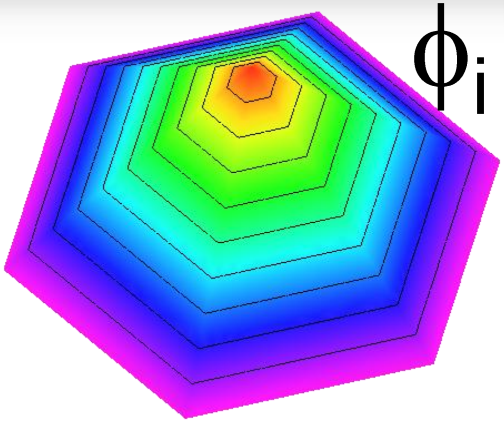
$$\int_{P \in t} \Phi_i \Phi_j dA = 2|t| \int_{\Phi_i=0}^{1} \int_{\Phi_j=0}^{1-\Phi_i} \Phi_i \Phi_j d\Phi_i d\Phi_j =$$

$$|t| \int_{\Phi_i=0}^{1} \Phi_i (1-\Phi_i)^2 d\Phi_i = |t| \left(\frac{1}{2} - \frac{2}{3} + \frac{1}{4} \right) = \frac{|t|}{12}$$

4. Discretizing the Laplacian

Discretization – P1

$$\begin{bmatrix} & & \\ & < \Delta \phi_i, \phi_j > & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & <\phi_i, \phi_j> & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$



ϕ_i

Coefficients of the mass matrix:

$<\phi_i, \phi_j> = (|t|+|t'|)/12$ if i and j share an edge

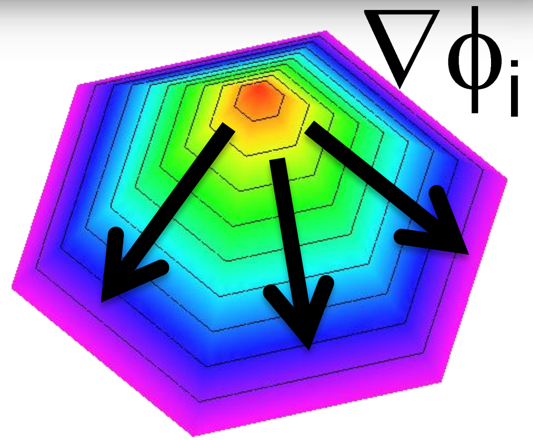
$$<\phi_i, \phi_i> = \Big(\sum_{t \text{ in } St(i)} |t|\Big) / 6$$

$<\phi_i, \phi_j> = 0$ otherwise

4. Discretizing the Laplacian

Discretization – P1

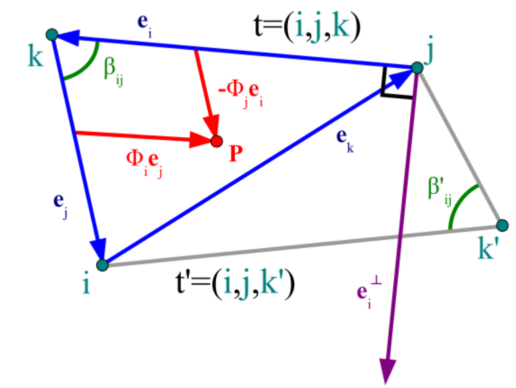
$$\begin{bmatrix} & & \\ & <\Delta \phi_i, \phi_j> & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & <\phi_i, \phi_j> & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}$$



$$<\Delta \phi_i, \phi_j> = -<\nabla \phi_i, \nabla \phi_j> \text{ (+ boundary term)}$$

Discretization – P1

$$\begin{bmatrix} & & \\ & <\Delta \phi_i, \phi_j> & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & <\phi_i, \phi_j> & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix}$$



$< \Delta \phi_i, \phi_j> = -<\nabla\phi_i,\nabla\phi_j>$ (+ boundary term)

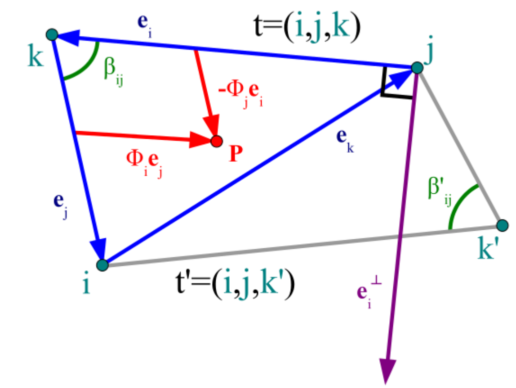
$$\int_t \nabla\Phi^i . \nabla\Phi^j dA = \frac{\mathbf{e}_i . \mathbf{e}_j}{4|t|} = \frac{||\mathbf{e}_i||.||\mathbf{e}_j|| \cos(\beta_{ij})}{2||\mathbf{e}_i||.||\mathbf{e}_j|| \sin(\beta_{ij})} = \frac{\cot(\beta_{ij})}{2}$$

Summing over the triangles gives …

4. Discretizing the Laplacian

Discretization – P1

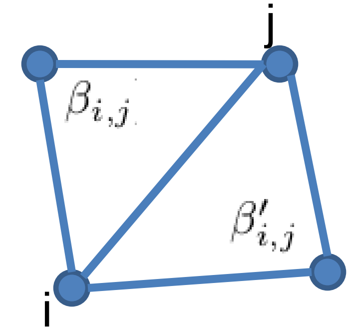
$$\begin{bmatrix} & & \\ & <\Delta \phi_i, \phi_j> & \\ & & \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} & & \\ & <\phi_i, \phi_j> & \\ & & \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$



$$\begin{aligned} Q_{i,j} &= \left(\text{cotan}(\beta_{i,j}) + \text{cotan}(\beta'_{i,j})\right)/2 \\ Q_{i,i} &= -\sum_j Q_{i,j} \end{aligned}$$

P1 FEM Laplacian - summary

$$\begin{bmatrix} \\ \langle \Delta \phi_i, \phi_j \rangle \\ \\ \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \\ \alpha_n \end{bmatrix} = \begin{bmatrix} \\ \langle \phi_i, \phi_j \rangle \\ \\ \end{bmatrix} \begin{bmatrix} \beta_1 \\ \\ \beta_n \end{bmatrix}$$



$$\begin{aligned} Q_{i,j} &= \left(\mathrm{cotan}(\beta_{i,j}) + \mathrm{cotan}(\beta'_{i,j})\right)/2 \\ Q_{i,i} &= -\textstyle\sum_j Q_{i,j} \end{aligned}$$

$$\begin{aligned} B_{i,j} &= \left(|t| + |t'|\right)/12 \\ B_{i,i} &= \left(\textstyle\sum_{t \in St(i)} |t|\right)/6 \end{aligned}$$

4. Discretizing the Laplacian

P1 FEM Laplacian

[Pinkall and Polthier 94]

[Meyer and Desbrun 99]

4. Discretizing the Laplacian

P1 FEM Laplacian

THE SOLUTION OF PARTIAL DIFFERENTIAL

EQUATIONS BY MEANS OF ELECTRICAL

NETWORKS

Thesis by

Richard H. MacNeal

California Institute of Technology

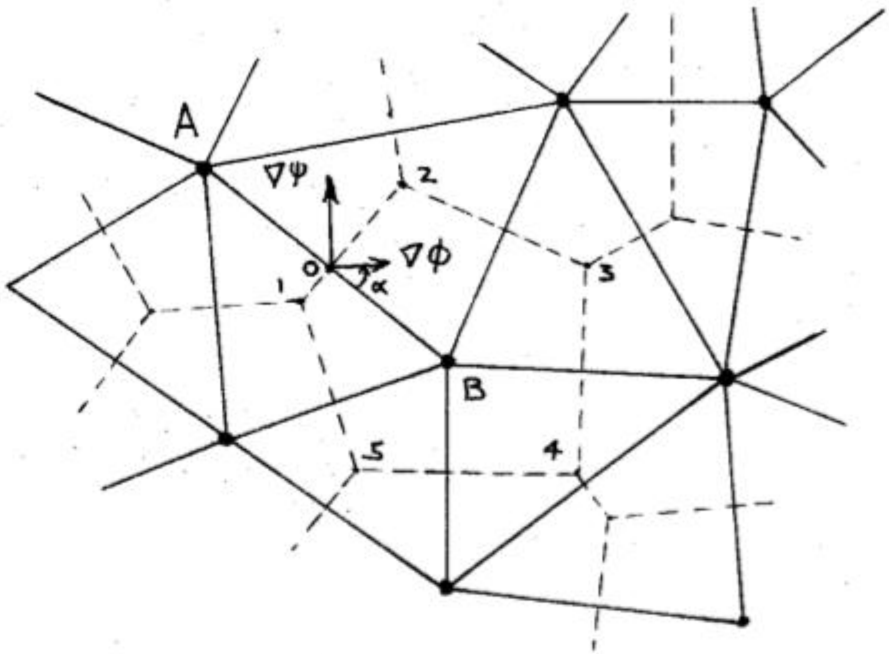
Pasadena, California

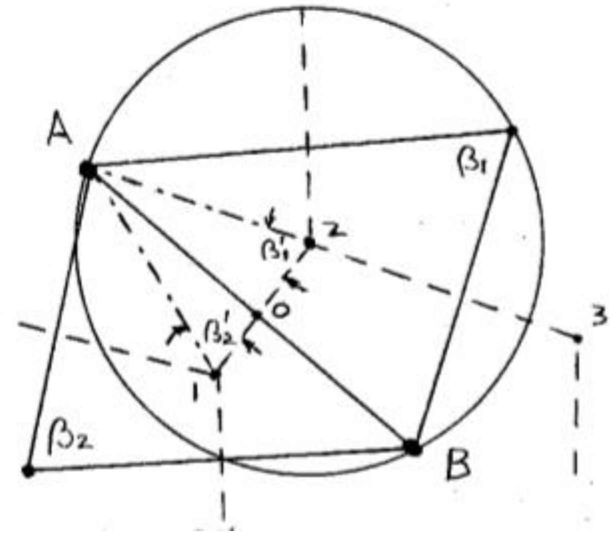
1949

Courtesy of Rhaleb Zayer

4. Discretizing the Laplacian

P1 FEM Laplacian [Mac Neal 1949]



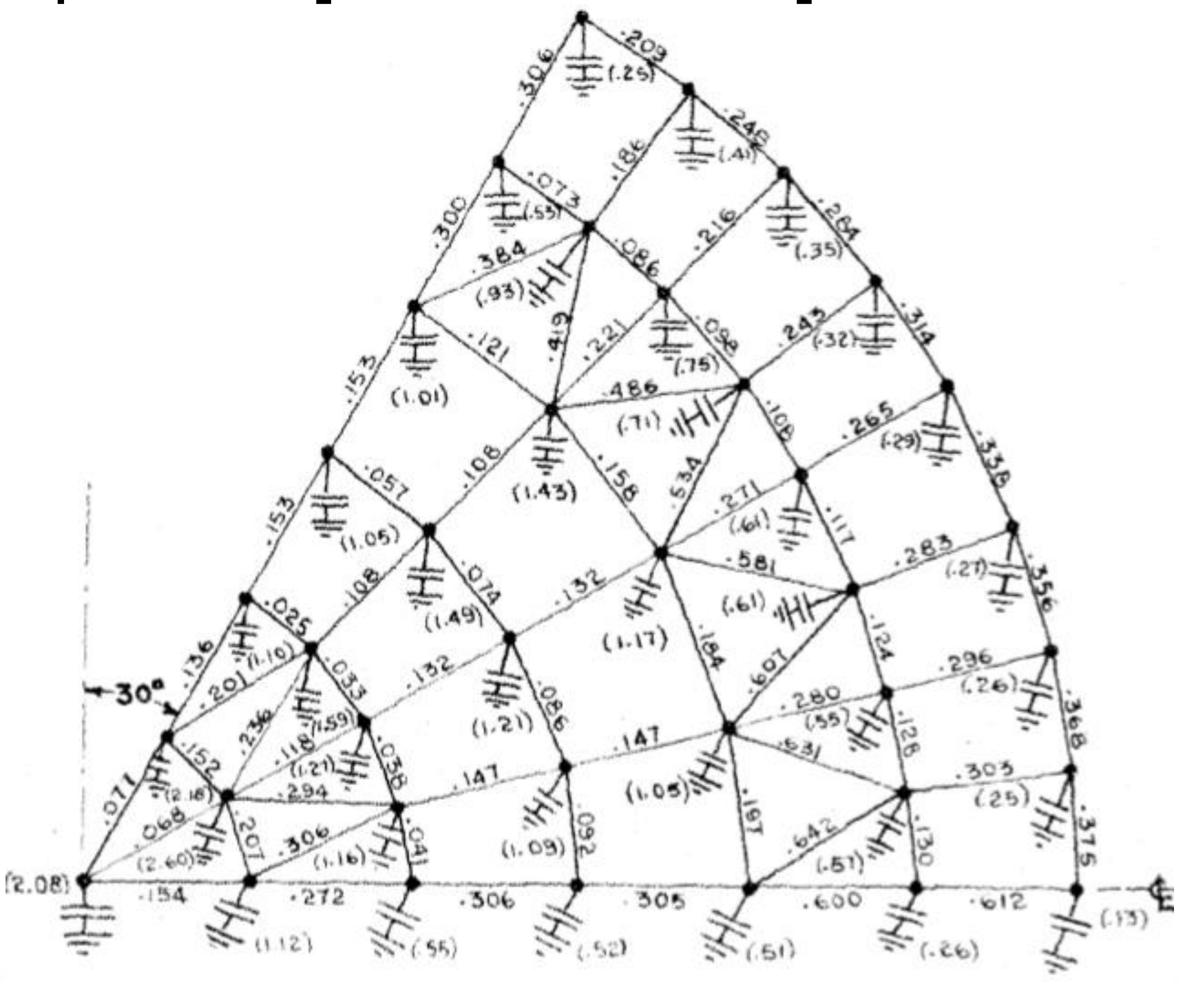


$$\frac{1}{4}\,\overline{AB}\;l_{12} = \frac{(\overline{AB})^2}{8}\,(\mathrm{ctn}\,\beta_1 + \mathrm{ctn}\,\beta_2)$$

Courtesy of Rhaleb Zayer

4. Discretizing the Laplacian

P1 FEM Laplacian [Mac Neal 1949]



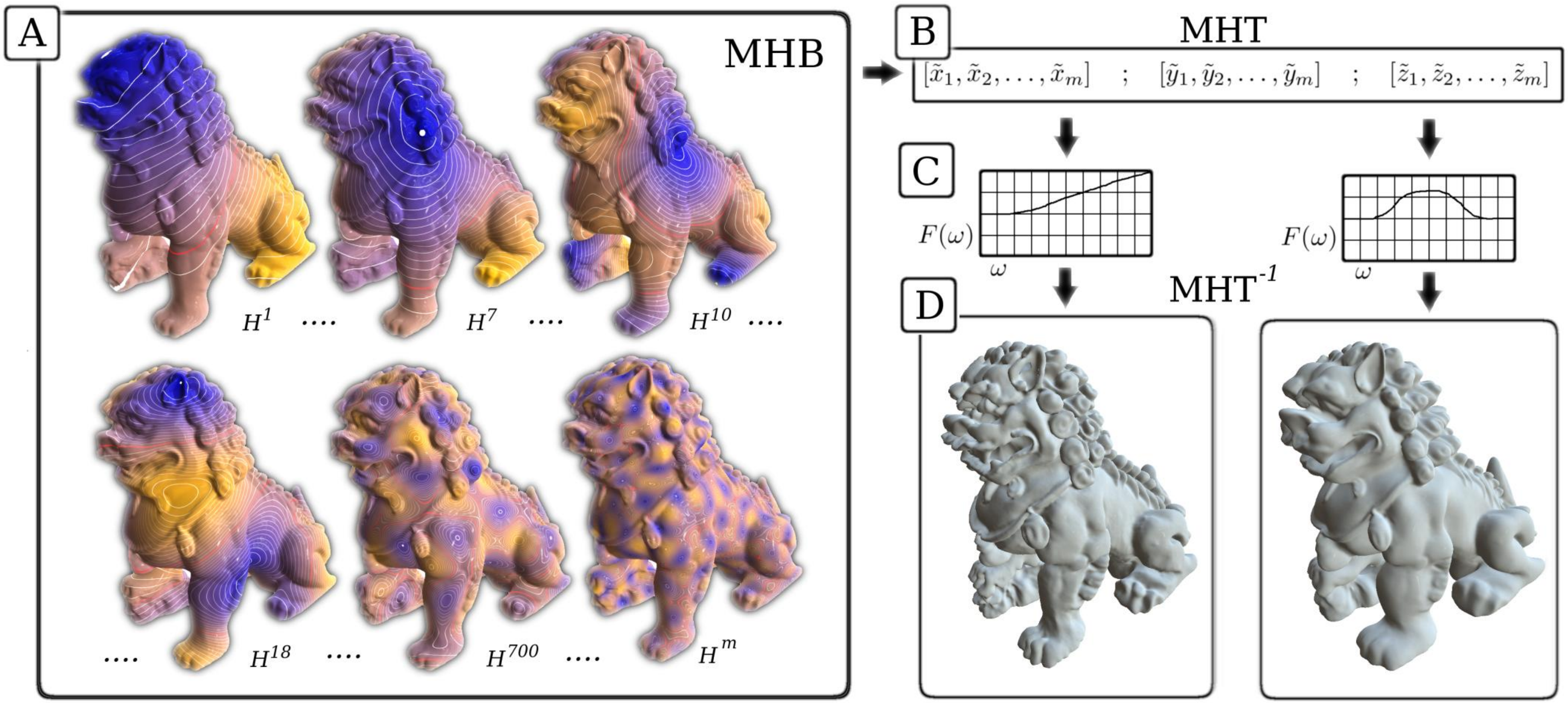
Courtesy of Rhaleb Zayer

5. Eigenfunctions

Operator equation: $Lf = \lambda f$

$Lf = \Delta f = \mathrm{div}(\mathrm{grad}\ f))$

Manifold Harmonics
[Vallet & L, 2007]



5. Eigenfunctions

- Operator equation: $Lf = \lambda f$
- Function basis (ϕi): $f = \Sigma \alpha_i \phi_i$
- Inner Product: $<f,g> = \int f(x)\, g(x)\, dx$
- $\forall i, <Lf, \phi_i> = \lambda <f, \phi_i>$

5. Eigenfunctions

Test function space (ϕi): $f = \Sigma\, \alpha_i\, \phi_i$ (P1,P2,P3)

$\forall i, <\Delta f, \phi_i> = \lambda <f, \phi_i>$

$<\Delta f, g> = -<\nabla f, \nabla g>$ (+ boundary term)

$Ax = \lambda Bx$

$a_{ij} = -<\nabla \phi_i, \nabla \phi_j>$; $b_{ij} = <\phi_i, \phi_j>$

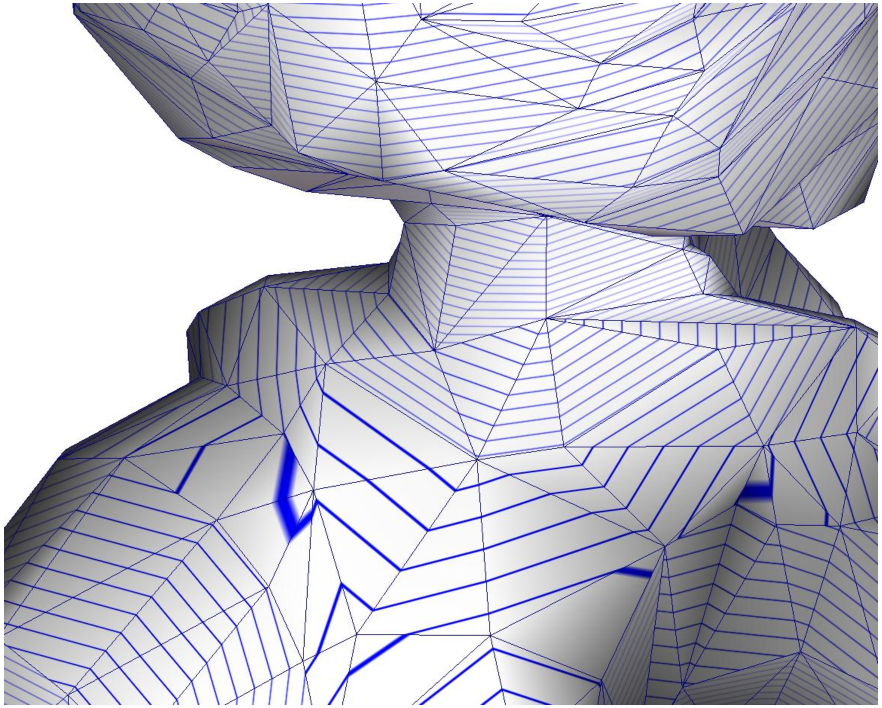
Generalized eigenvalue problem
(solved by ARPAK or MATLAB)

5. Eigenfunctions

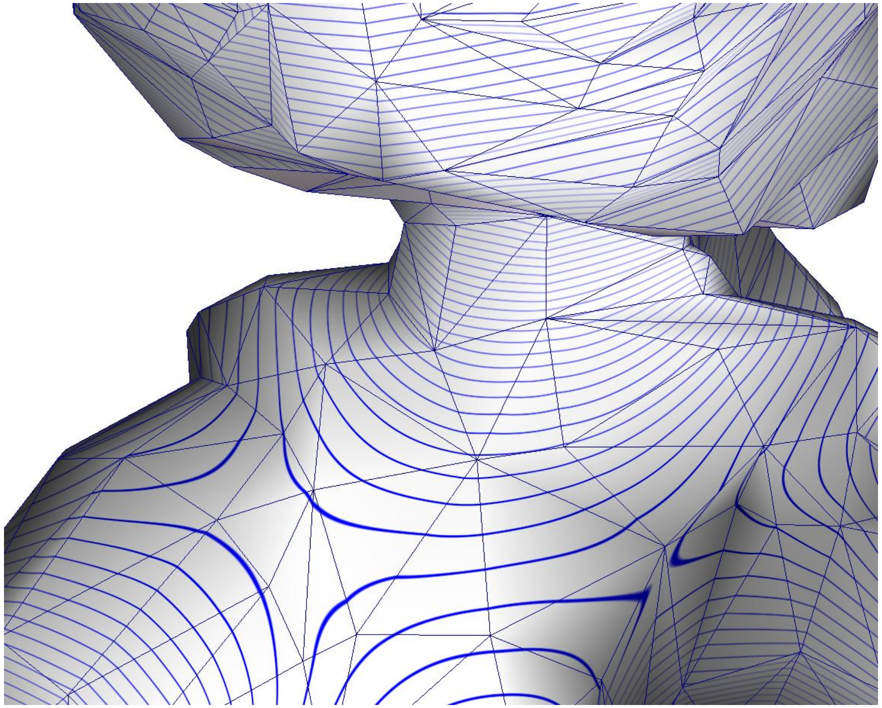
P1 function basis

P3 function basis

5. Eigenfunctions



P1 function basis

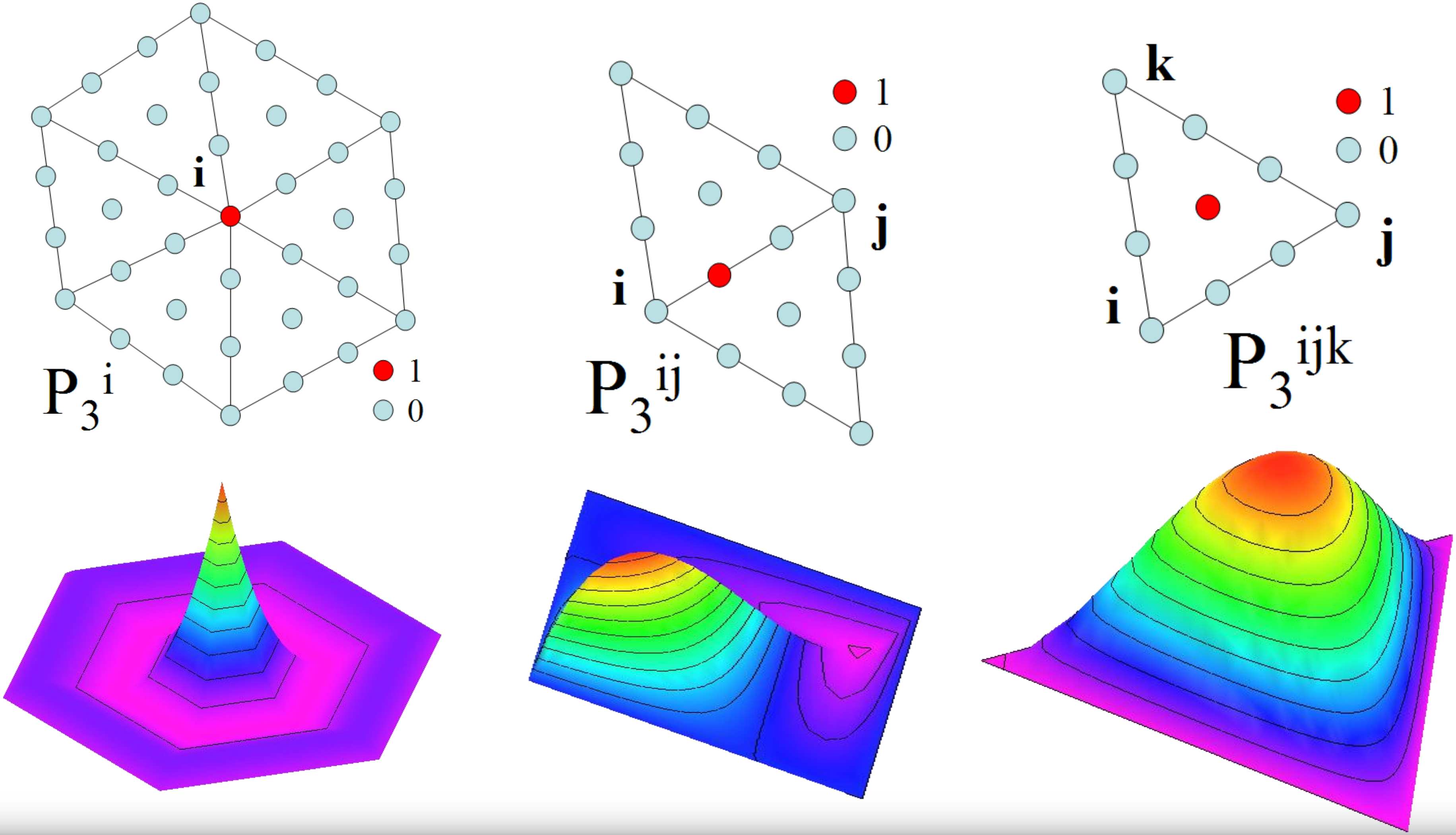


P3 function basis

5. Eigenfunctions

5. Eigenfunctions

Compute the terms $<\nabla\phi_i,\nabla\phi_j>$ and $<\phi_i,\phi_j>$ using the P3 function basis …



Courtesy of Bruno Vallet

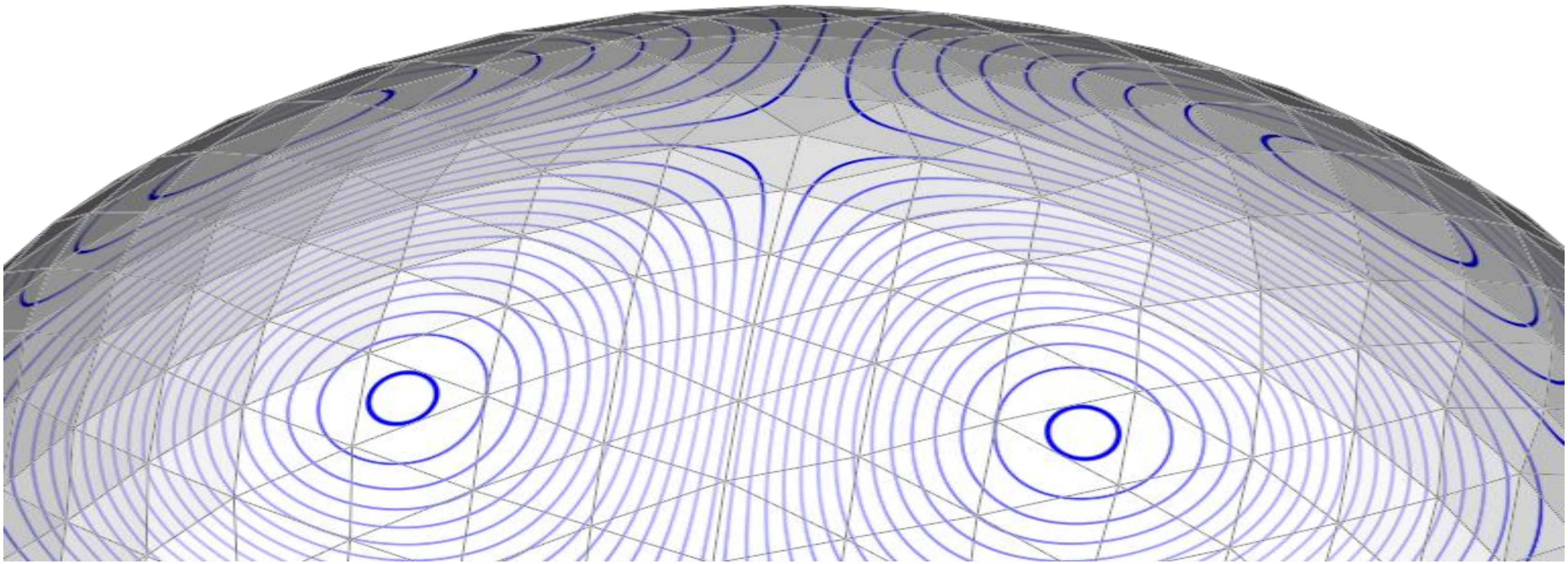
5. Eigenfunctions

Compute the terms $<\nabla\phi_i,\nabla\phi_j>$ and $<\phi_i,\phi_j>$ using the P3 function basis …

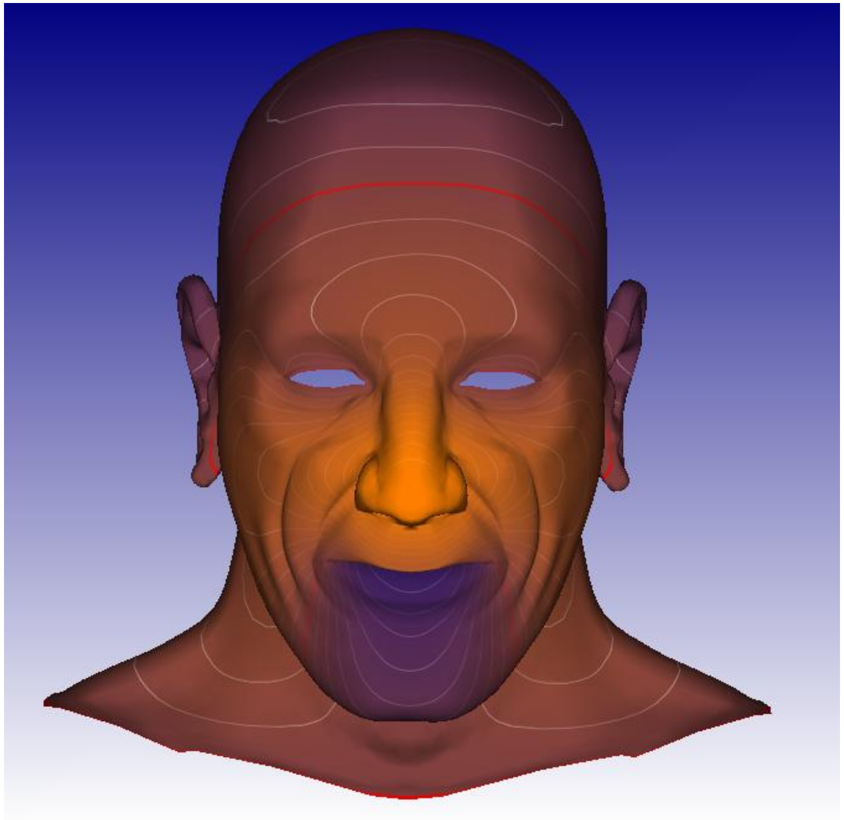
Derivations: see Bruno Vallet's Ph.D thesis

Implementation available in Graphite / ManifoldHarmonics plugin

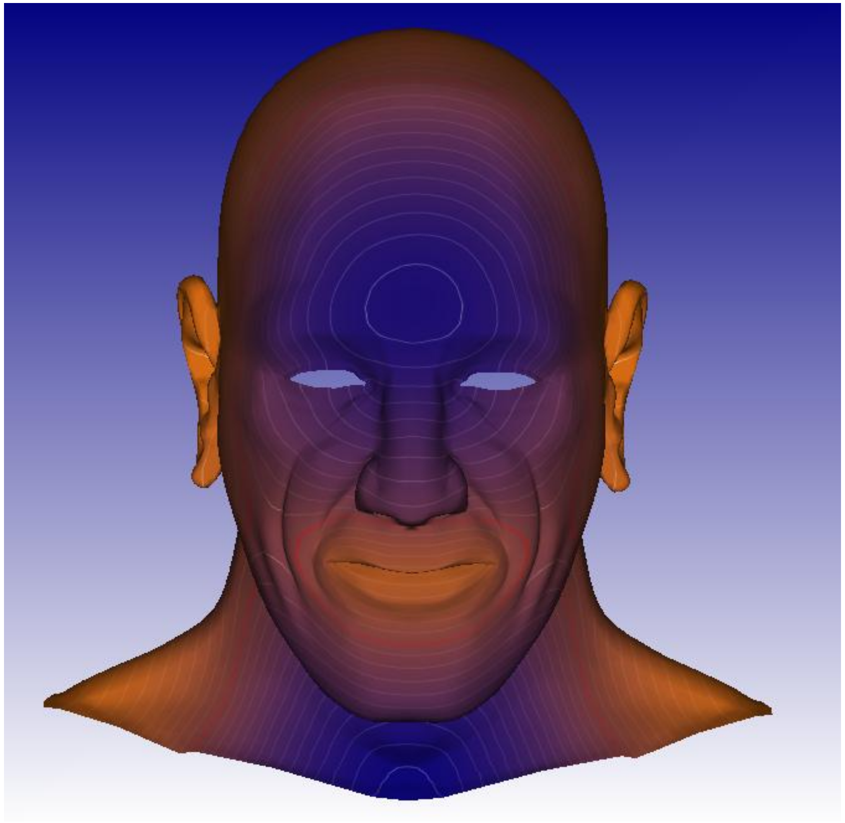
See http://alice.loria.fr/software



Boundary conditions

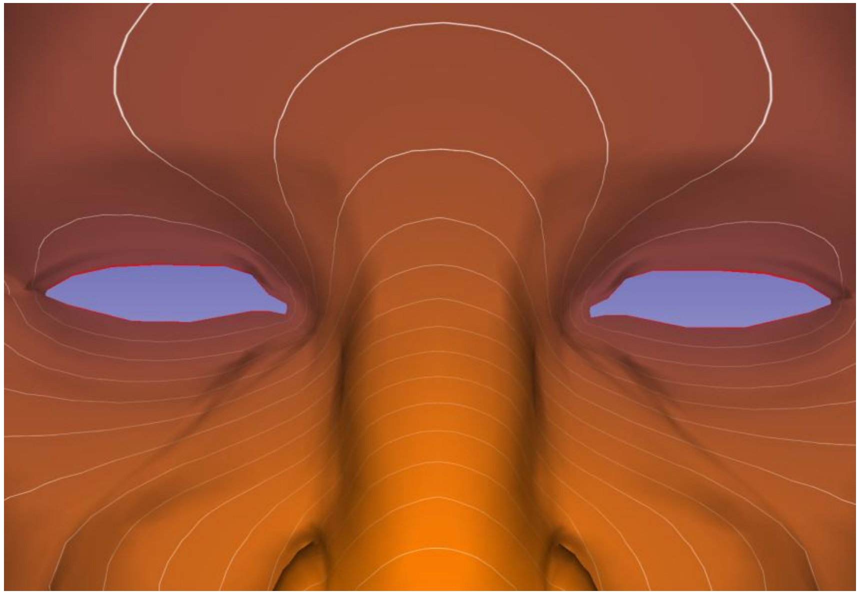


Dirichlet

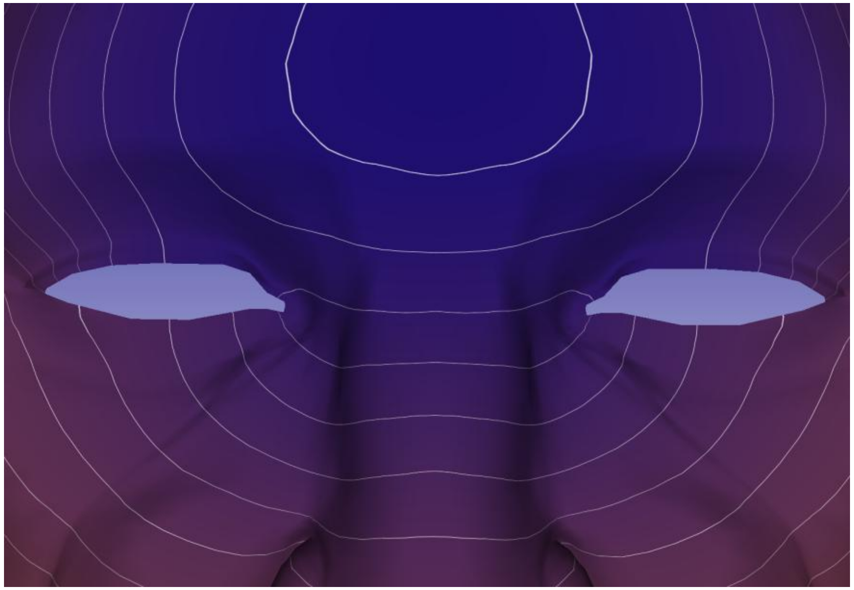


Neumann

Boundary conditions



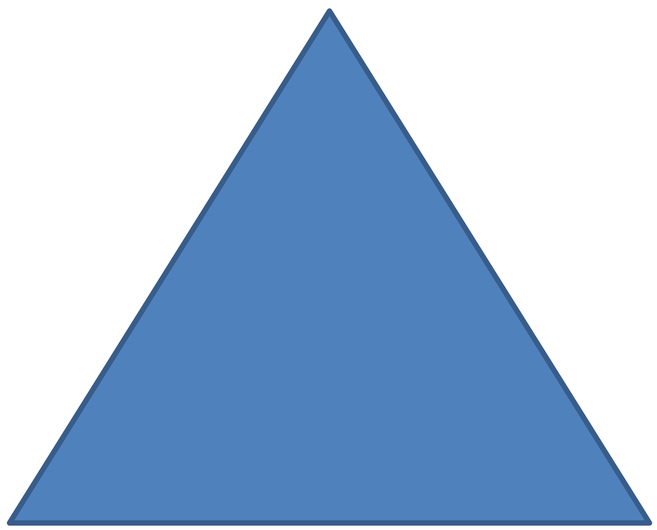
Dirichlet



Neumann

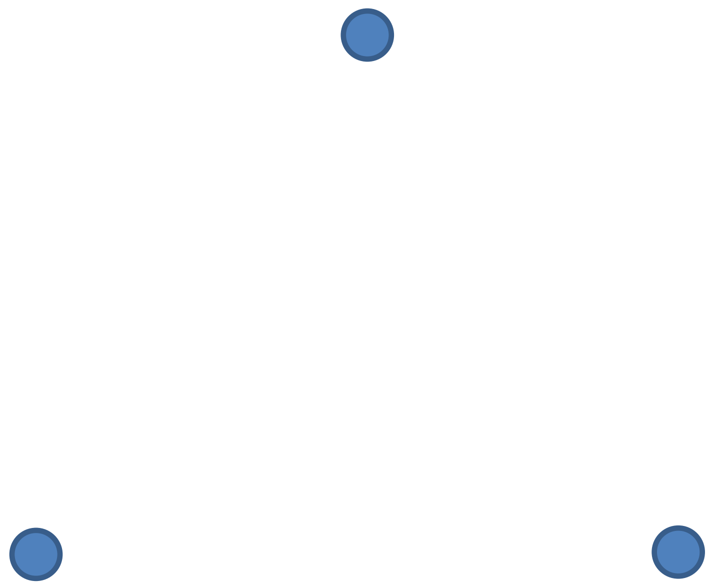
Epilogue

Discrete or continuous ?



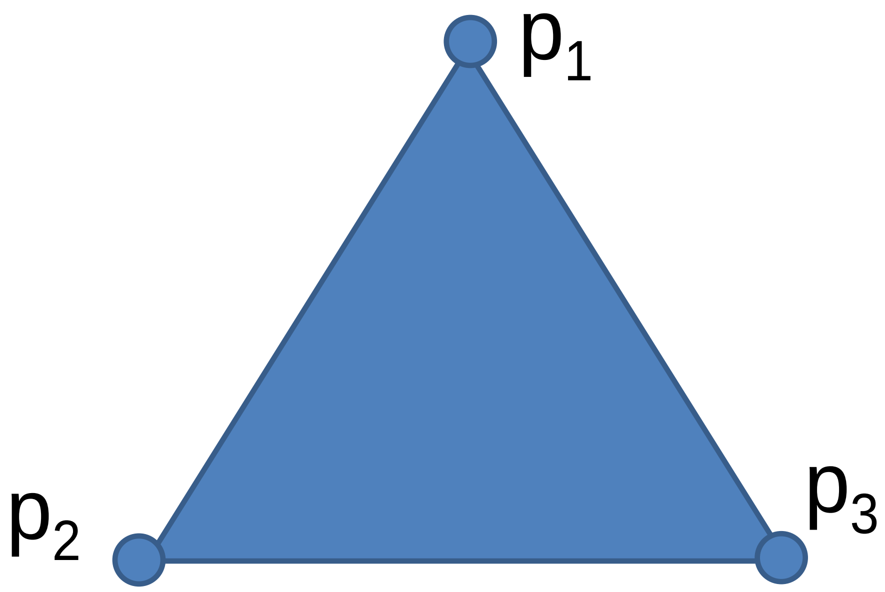
How many points in a triangle ?

Discrete or continuous ?



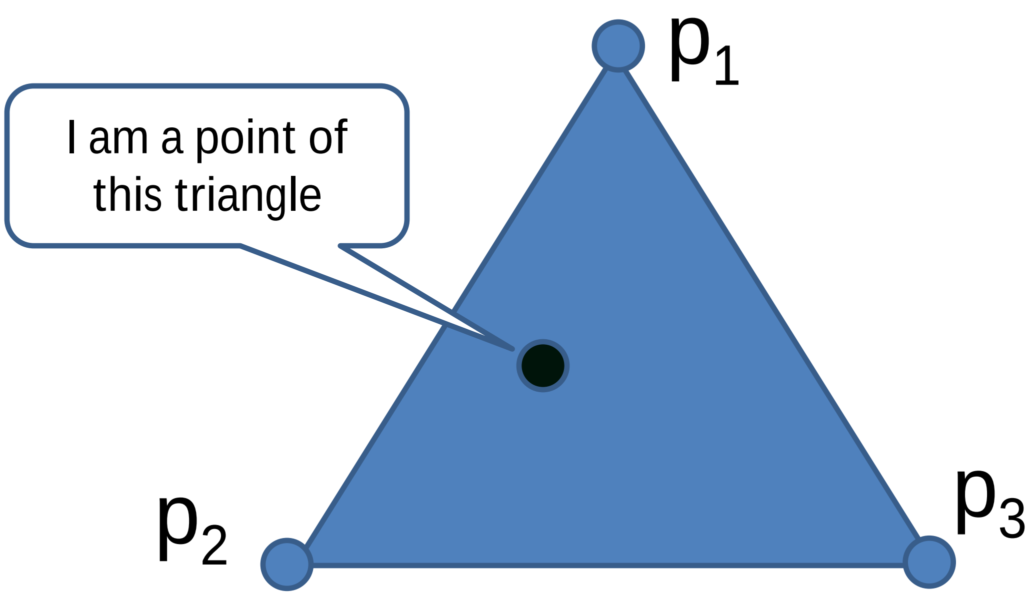
Is this a triangle ?

Discrete or continuous ?



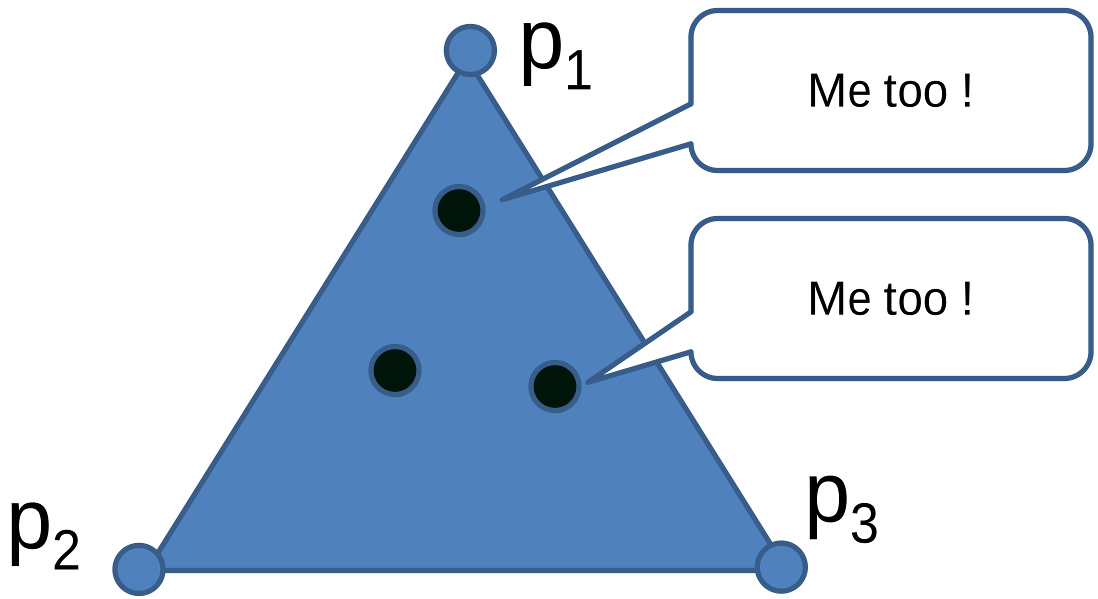
$$t = \{ \lambda_1 p_1 + \lambda_2 p_2 + \lambda_3 p_3,\ 0 < \lambda_1, \lambda_2, \lambda_3,\ \lambda_1+\lambda_2+\lambda_3=1\}$$

Discrete or continuous ?



$$t = \{ \lambda_1 p_1 + \lambda_2 p_2 + \lambda_3 p_3, 0 < \lambda_1, \lambda_2, \lambda_3, \lambda_1+\lambda_2+\lambda_3=1\}$$

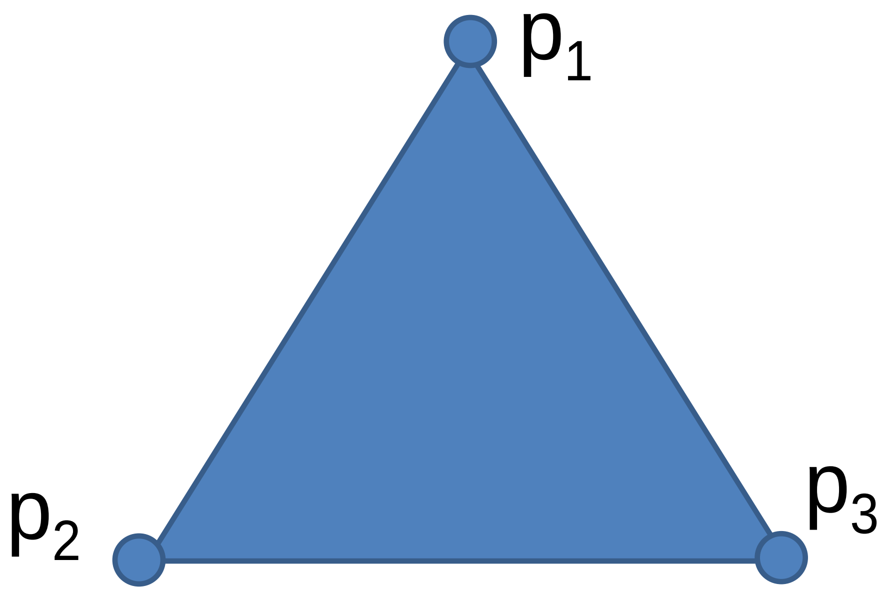
Discrete or continuous ?



$$t = \{ \lambda_1 p_1 + \lambda_2 p_2 + \lambda_3 p_3, 0 < \lambda_1, \lambda_2, \lambda_3, \lambda_1+\lambda_2+\lambda_3=1\}$$

Epilogue

A mesh surface is a continuous object encoded by a discrete set of parameters

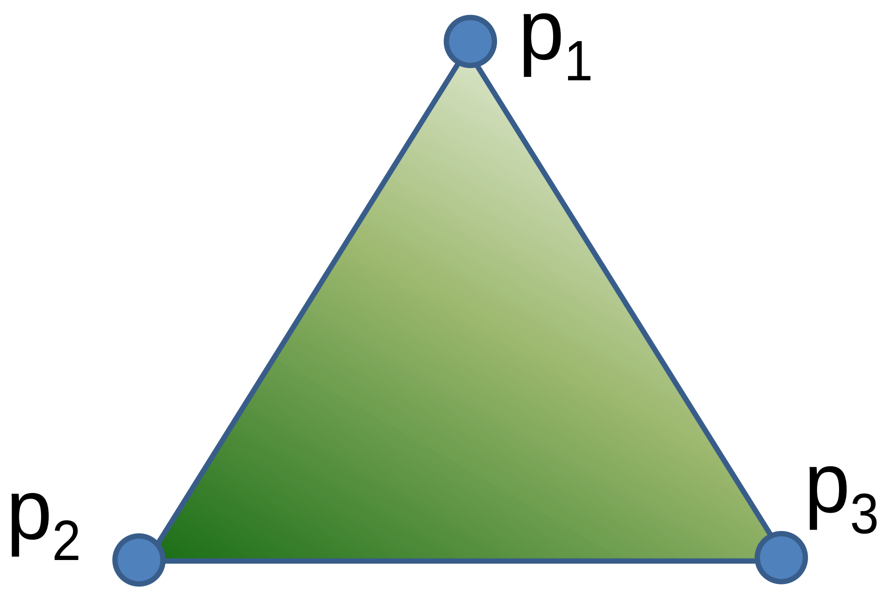


$$t = \{ \lambda_1 p_1 + \lambda_2 p_2 + \lambda_3 p_3, 0 < \lambda_1, \lambda_2, \lambda_3, \lambda_1+\lambda_2+\lambda_3=1\}$$

Epilogue

SIGGRAPH ASIA 2011 HONG KONG

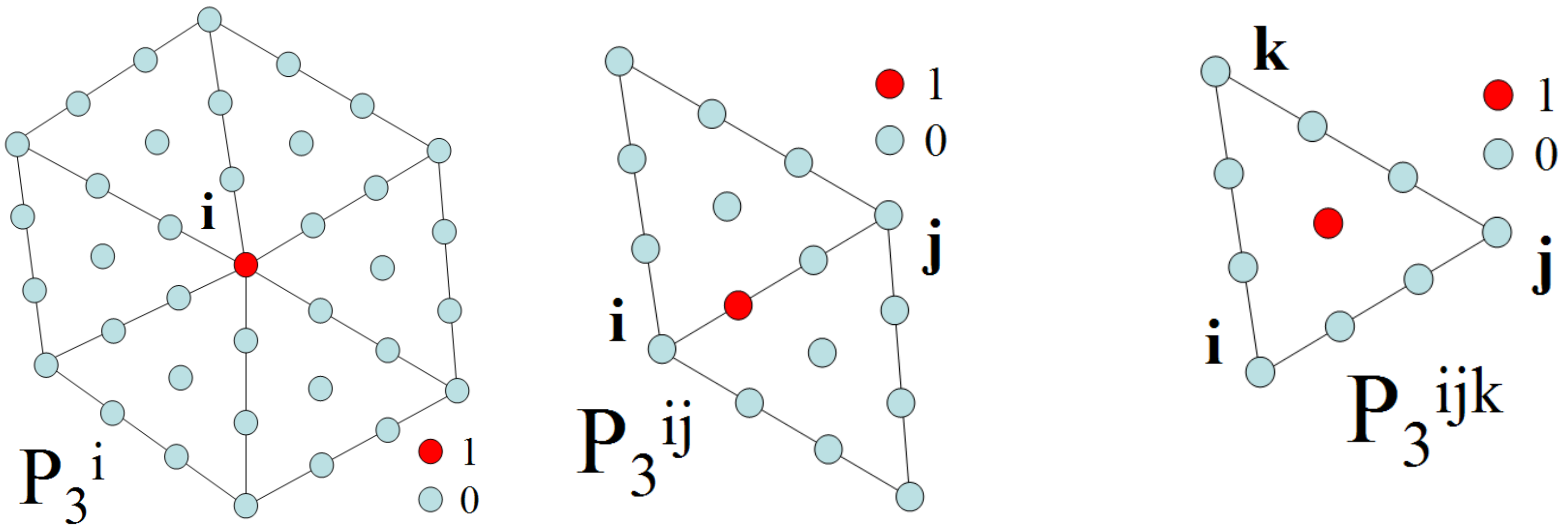
We can compute integrals over the triangle



$$t = \{ \lambda_1 p_1 + \lambda_2 p_2 + \lambda_3 p_3,\ 0 < \lambda_1, \lambda_2, \lambda_3,\ \lambda_1+\lambda_2+\lambda_3=1\}$$

Epilogue

A Pn function is also a continuous object encoded by a discrete set of parameters



Epilogue

Discretization of parameters (geo. or func.)

$$\neq$$

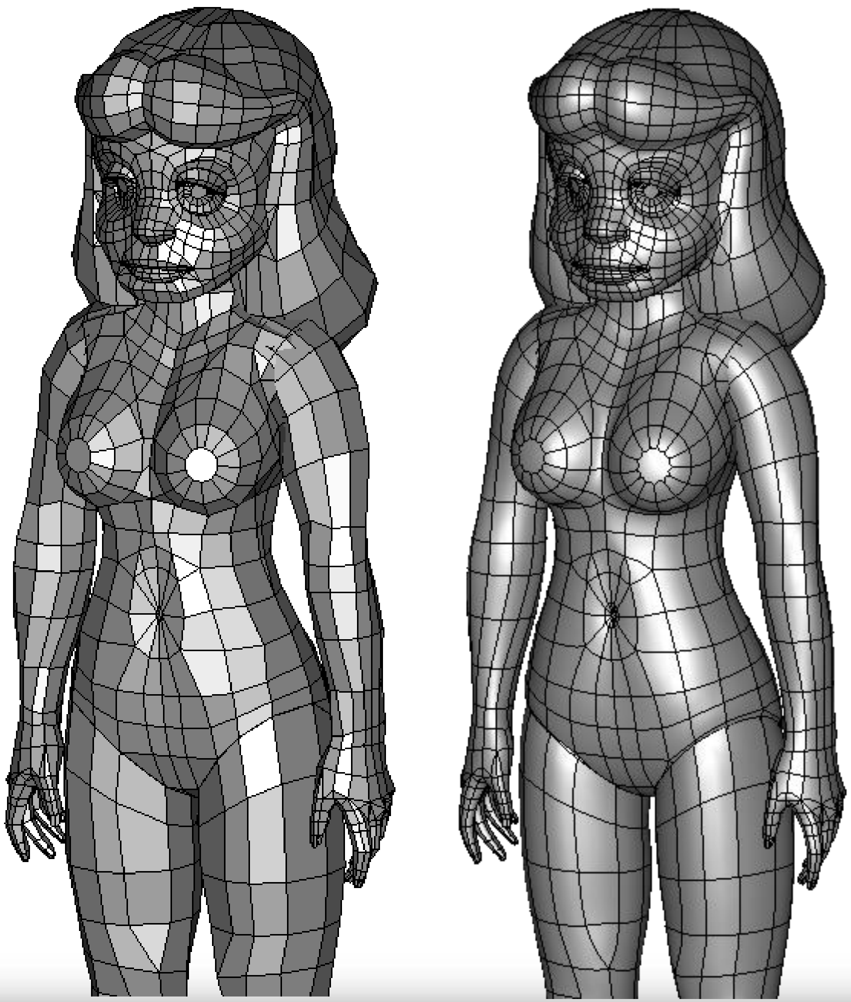
Discretization of geometry

What's the point ?

Epilogue

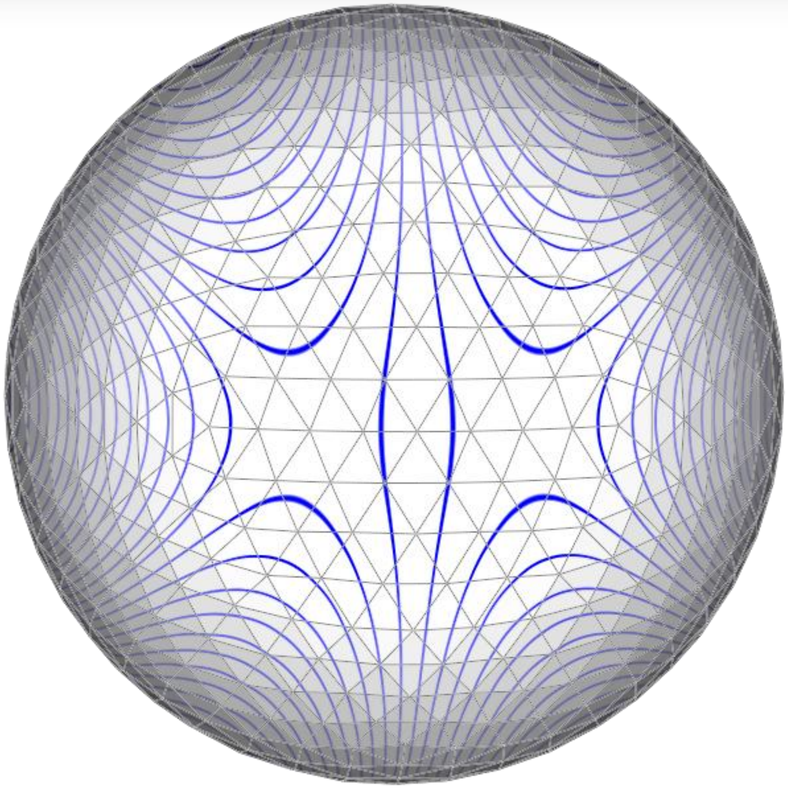
What's the point ?

(1) Smooth is beautiful



What's the point ?

(1) Smooth is beautiful



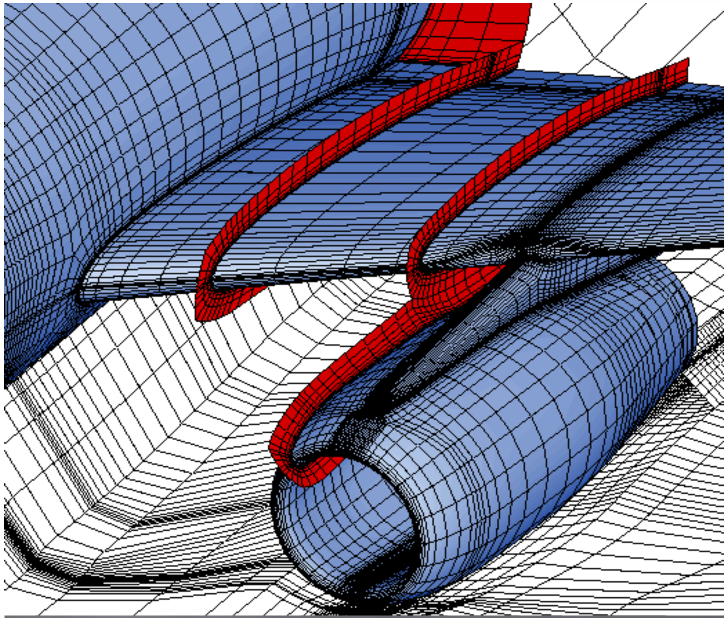
Epilogue

What's the point ?

(1) Smooth is beautiful

(2) FEM framework is general

From bunnies to spaceships …

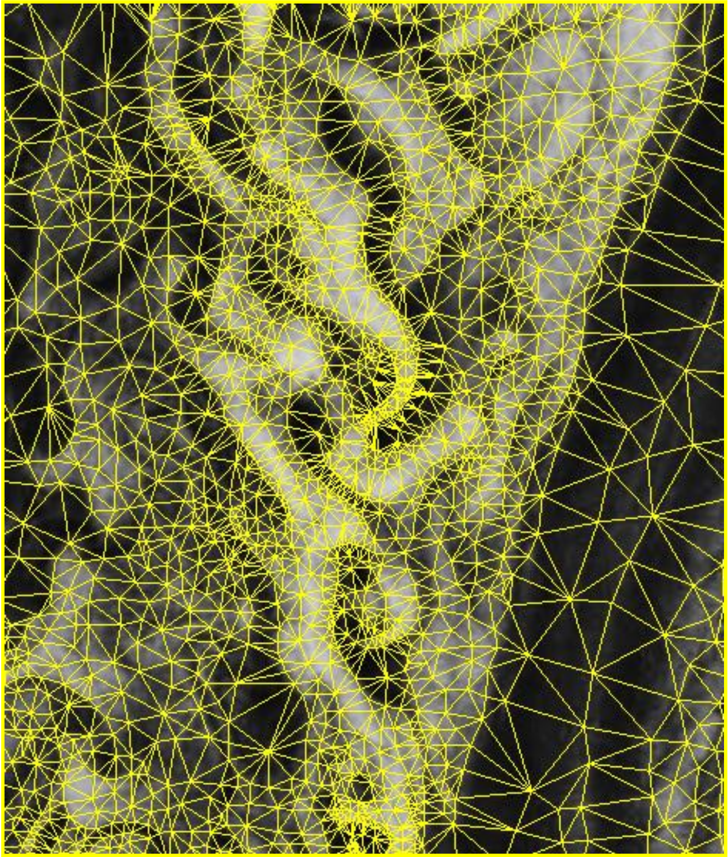


Epilogue

What's the point ?

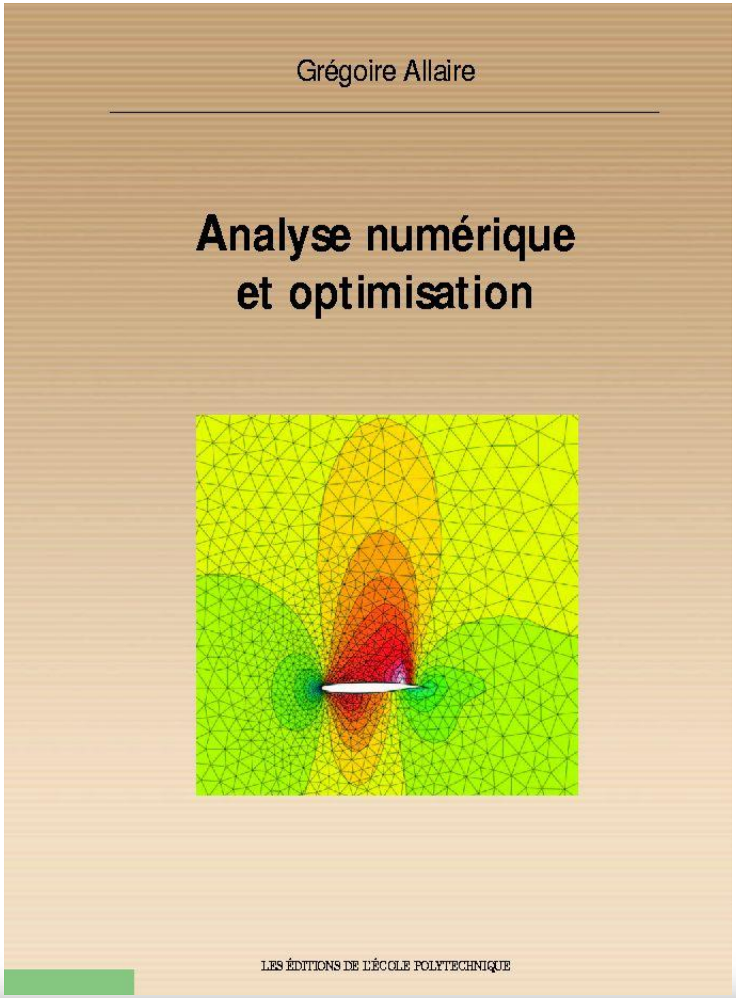
(1) Smooth is beautiful

(2) FEM framework is general



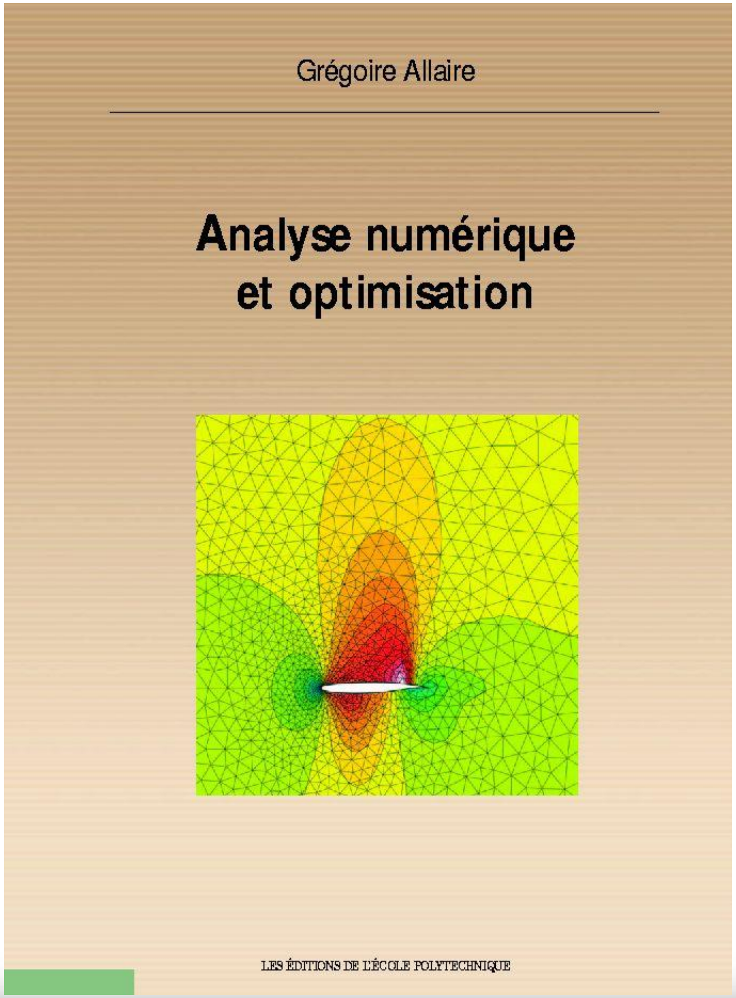
(3) P1 does not always suffice (Bi-Laplacian…)

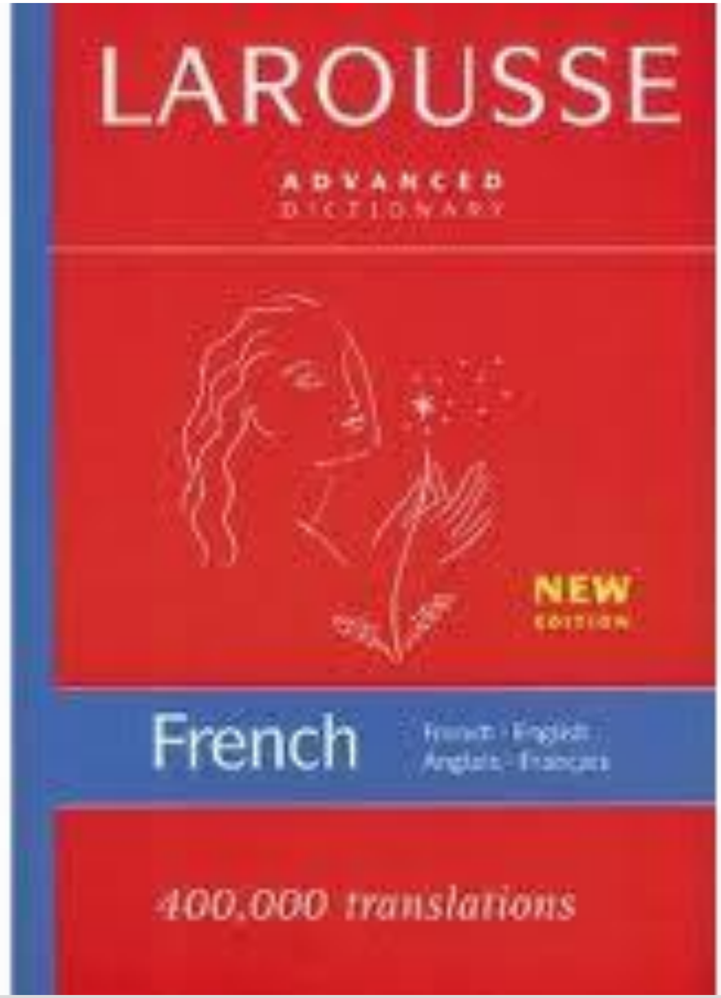
Further reading



Theory, validy, convergence
Sobolev spaces
Lax-Milgram
Other function bases
Other inner products
...

Further reading





Further reading