Computing Maxwell Eigenvalues in 3D using H^1 conforming FEM

Philipp Frauenfelder

Kersten Schmidt

Christoph Schwab

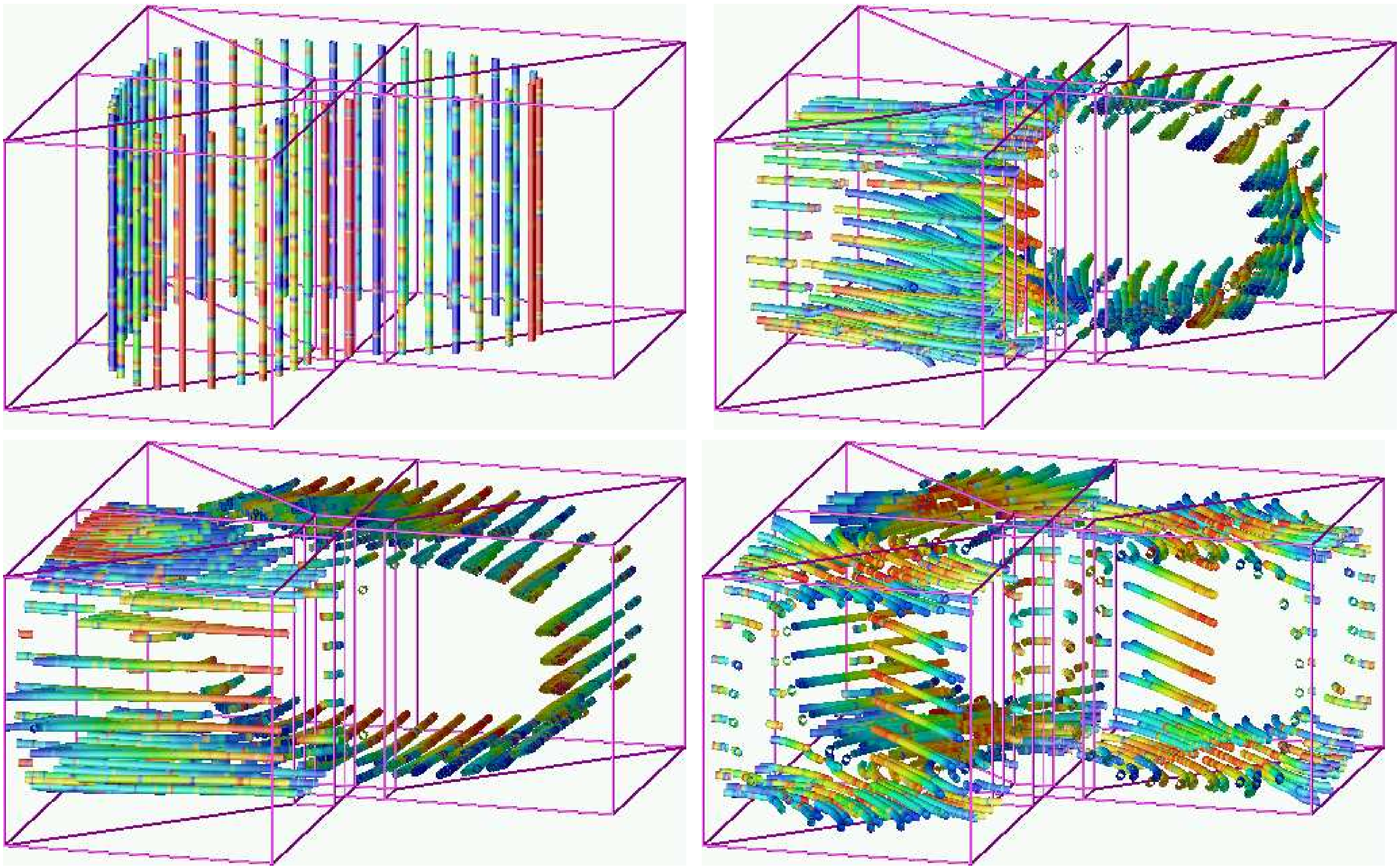
pfrauenf@math.ethz.ch

Seminar for Applied Mathematics

Federal Institute of Technology, ETH Zürich

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Motivation



Overview

- Introduction: FEM & Exponential Convergence
- Assembling
- Handling Hanging Nodes
- Finding Regular Supports
- Maxwell Eigenvalue Problems
- Perspectives

Our Software: Concepts

- Started by Christian Lage during his Ph.D. studies (1995).
- Used and improved by Frauenfelder, Matache, Schmidlin, Schmidt and several students.
- Concept Oriented Design using mathematical principles [1].
- Currently two parts: hp-FEM, BEM (wavelet and multipole methods).
- C++ class library for general elliptic PDEs.

[1] P. F. and Ch. Lage, "Concepts—An Object Oriented Software Package for Partial Differential Equations", *Mathematical Modelling and Numerical Analysis* 36 (5), pp. 937–951 (2002).

FEM Basics

Elliptic Boundary Value Problem in $\Omega \subset \mathbb{R}^n$ in variational form:
Find $u \in V$ such that:

$$a(u, v) = l(v) \qquad \forall v \in V,$$

where V is a FE space $a(.\,,.)$ a bilinear form and $l(.)$ a linear form.

FEM Basics

Elliptic Boundary Value Problem in $\Omega \subset \mathbb{R}^n$ in variational form:
Find $u \in V$ such that:

$$a(u,v) = l(v) \qquad \forall v \in V,$$

where V is a FE space $a(.\,,.)$ a bilinear form and $l(.)$ a linear form.

Example: $-\Delta u + u = f$ in Ω and $u = 0$ on $\partial\Omega$

$$\Longrightarrow a(u,v) = \int_\Omega \nabla u \cdot \nabla v \, dx + \int_\Omega uv \, dx,$$

$$l(v) = \int_\Omega fv \, dx$$

$$V = S^{1,\underline{p}}(\Omega, \mathcal{T}) \subset H^1(\Omega).$$

FEM Basics

Elliptic Boundary Value Problem in $\Omega \subset \mathbb{R}^n$ in variational form:
Find $u \in V$ such that:

$$a(u,v) = l(v) \qquad \forall v \in V,$$

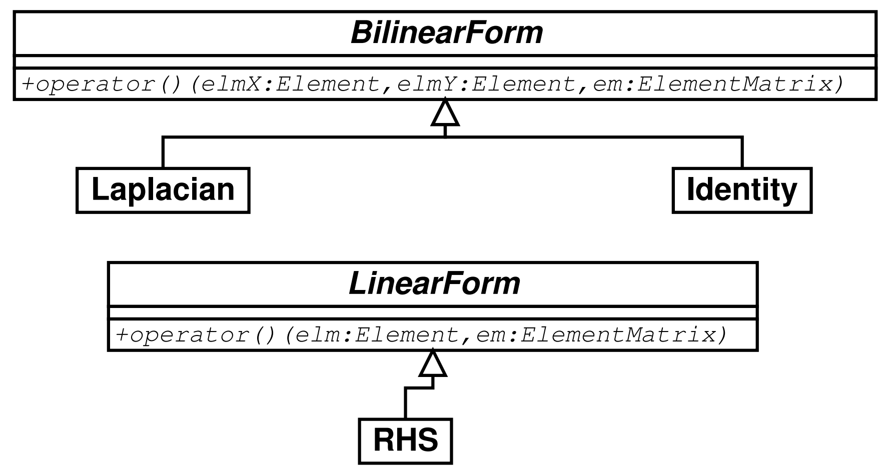
where V is a FE space $a(.\,,.)$ a bilinear form and $l(.)$ a linear form.

Example: $-\Delta u + u = f$ in Ω and $u = 0$ on $\partial\Omega$

$$\Longrightarrow a(u,v) = \int_\Omega \nabla u \cdot \nabla v \, dx + \int_\Omega uv \, dx,$$

$$l(v) = \int_\Omega fv \, dx$$

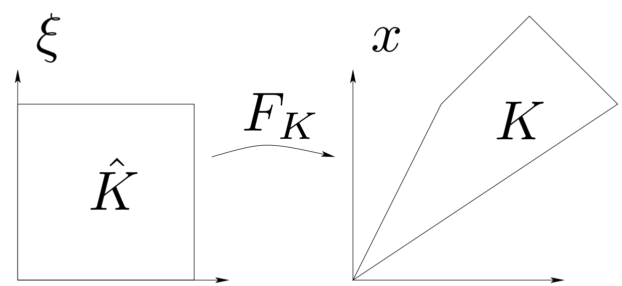
$$V = S^{1,\underline{p}}(\Omega, \mathcal{T}) \subset H^1(\Omega).$$

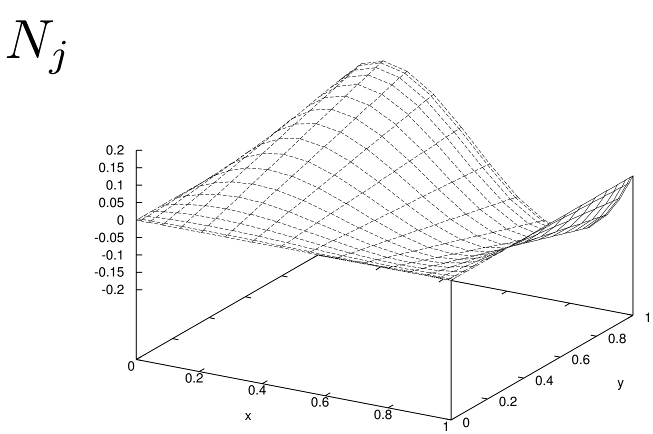


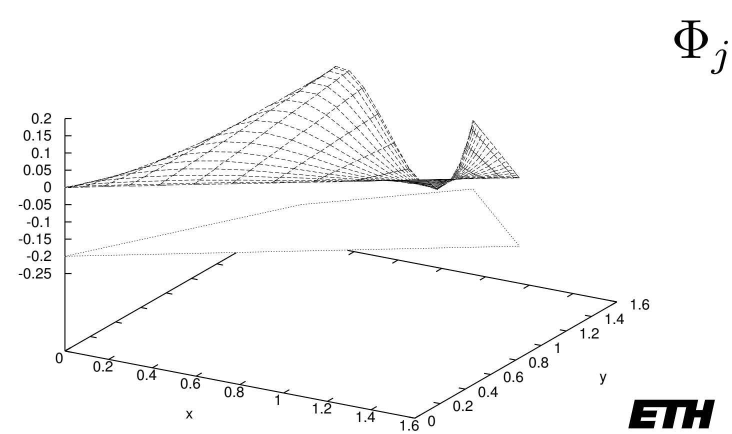
FE Space

- Basis $\{\Phi_i\}_{i=1}^N$ constructed from element shape functions ϕ_j^K on elements $K \in \mathcal{T}$.
- Reference element shape functions: N_j, element map: $F_K : \hat{K} \to K$

$$\Rightarrow \phi_j^K \circ F_K = N_j .$$

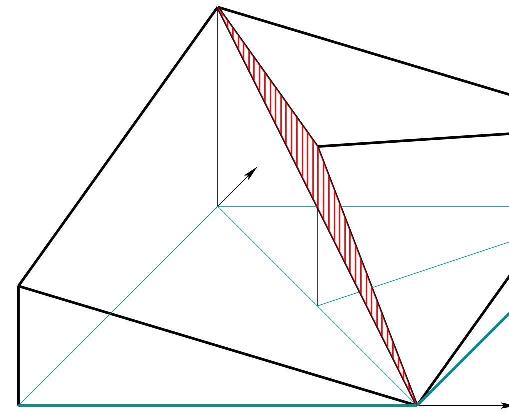


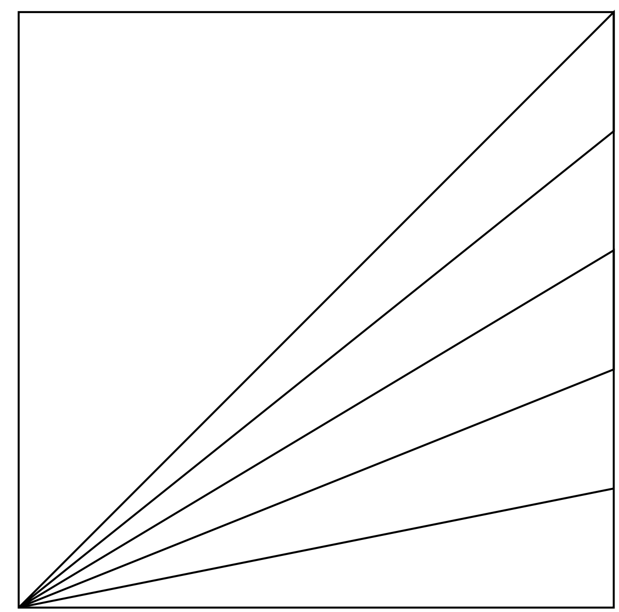




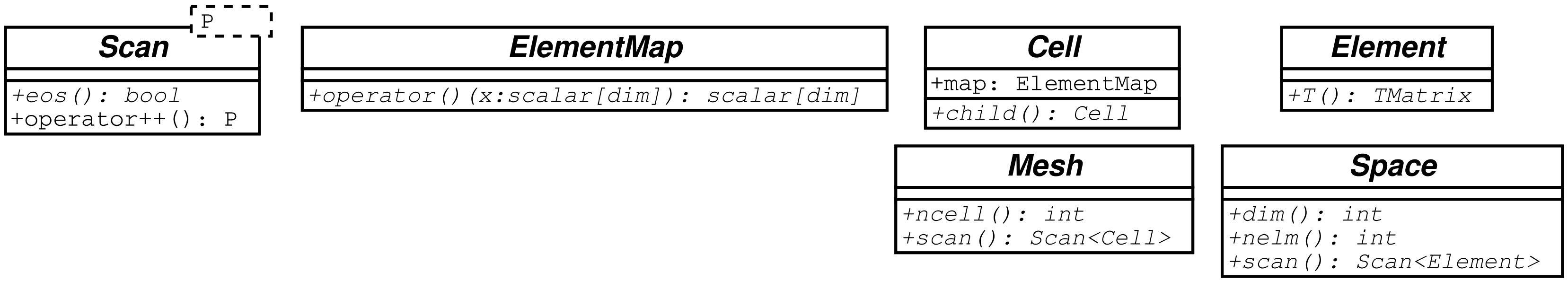
FE Meshes

- Need to be regular (no hanging nodes):
- Need to be shape-regular (no degenerate elements):





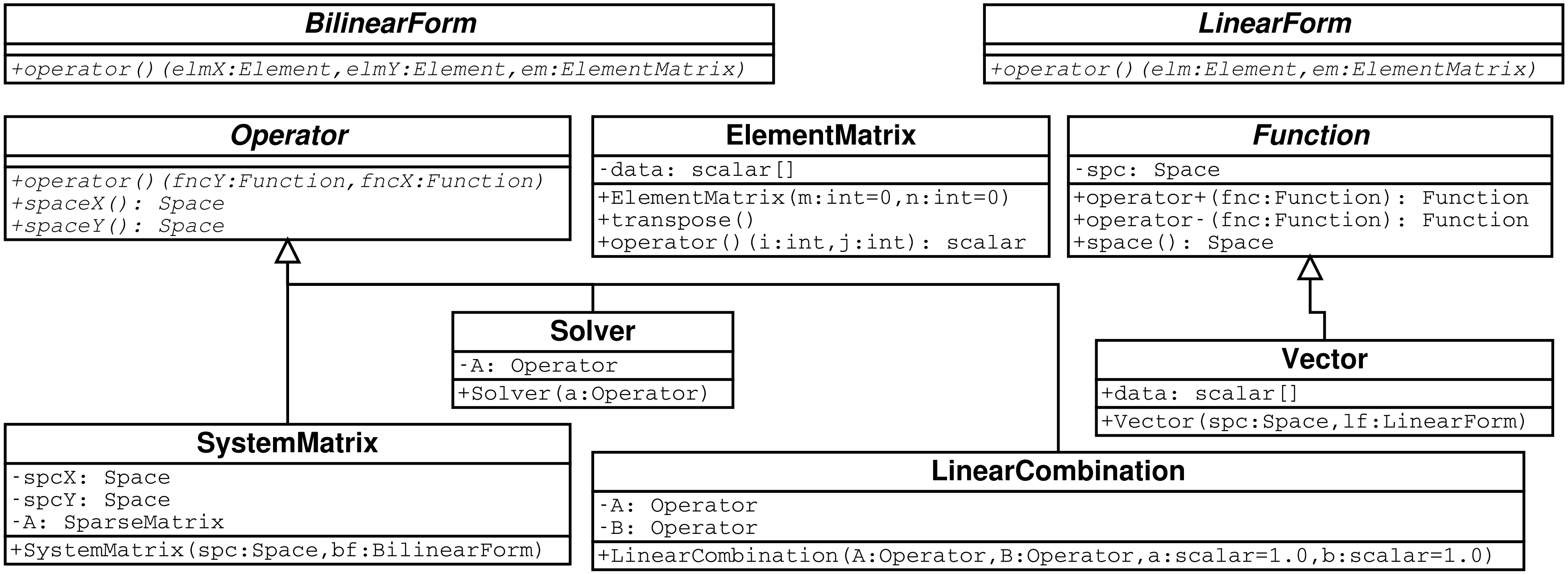
Classes for Mesh & Space



Ideas:

- Mesh consists of Cells, every Cell has an ElementMap
- Space consists of Elements, every Element has a Cell
- Scan used to loop over Elements of a Space or Cells of a Mesh

Classes for PDEs



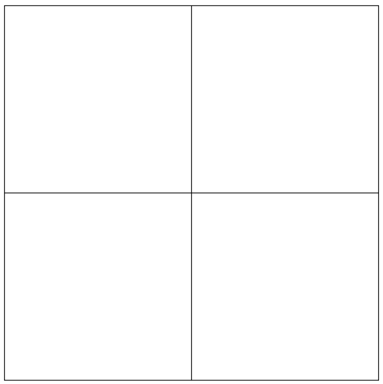
Variational formulation of an elliptic PDE on Ω:
Find $u \in V$ such that

$$a(u,v) = l(v) \qquad \forall v \in V$$

$$\Rightarrow (\boldsymbol{A} + \boldsymbol{M})\underline{u}_N = \underline{l}_N$$ solve for $\underline{u}_N$.

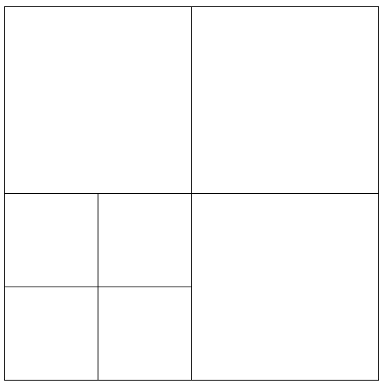
h *Refinements*

In general: elements with large error should be modified somehow
$\leadsto$ h refinement



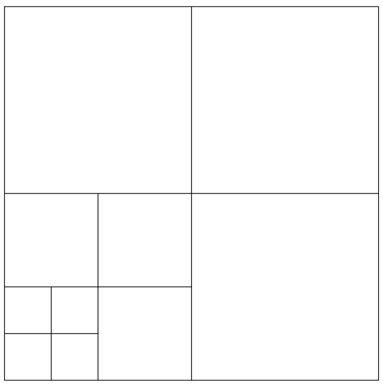
h Refinements

In general: elements with large error should be modified somehow
$\rightsquigarrow$ h refinement



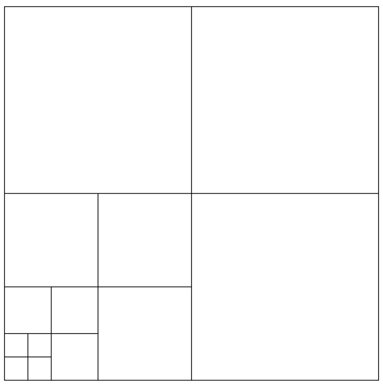
h ***Refinements***

In general: elements with large error should be modified somehow
$\rightsquigarrow$ h refinement



h ***Refinements***

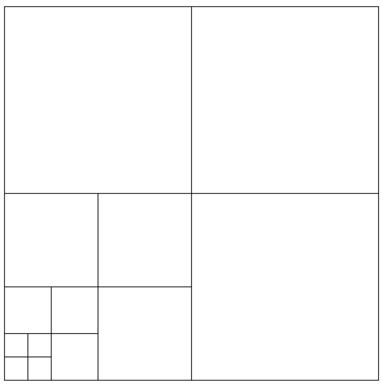
In general: elements with large error should be modified somehow
$\rightsquigarrow$ h refinement



1-irregular meshes!

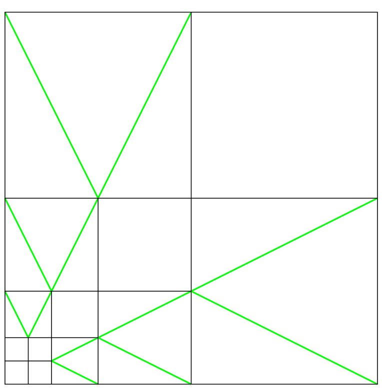
h Refinements

In general: elements with large error should be modified somehow
$\rightsquigarrow$ h refinement



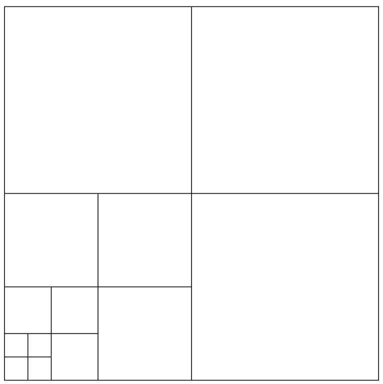
1-irregular meshes!

Remedy:



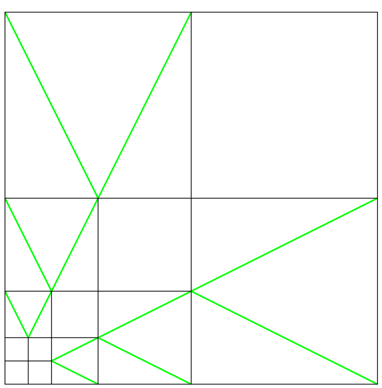
h Refinements

In general: elements with large error should be modified somehow
$\rightsquigarrow$ h refinement



1-irregular meshes!

Remedy:



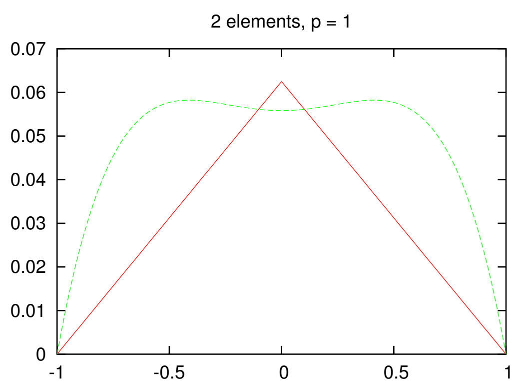
or

Handling the hanging nodes by applying constraints to them: A hanging node is not a real degree of freedom.

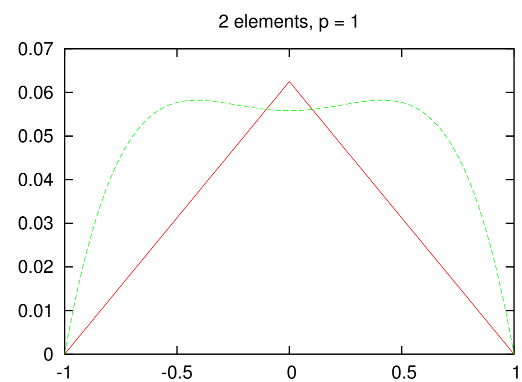
p *Enrichment*

Instead of refining an element: increasing polynomial degree p. Good for smooth functions.

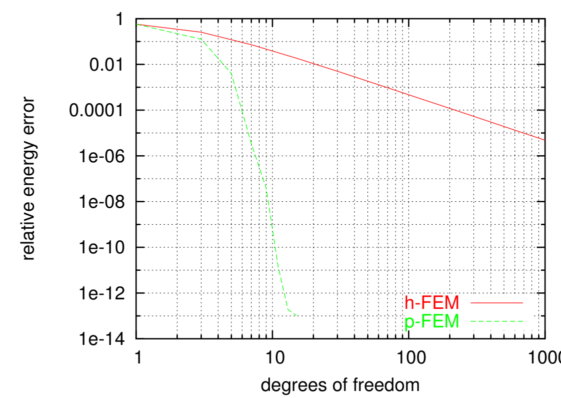
h FEM:



p FEM:



Convergence:



Solving the problem

$$-\Delta u + u = x^2 \text{ in } \Omega = (-1,1)$$

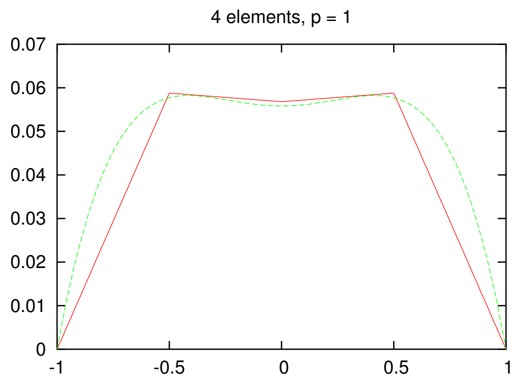
$$u = 0 \text{ on } \partial\Omega = \{-1,1\},$$

$$\Rightarrow u(x) = -3\frac{\cosh(x)}{\cosh(1)} + x^2 + 2.$$

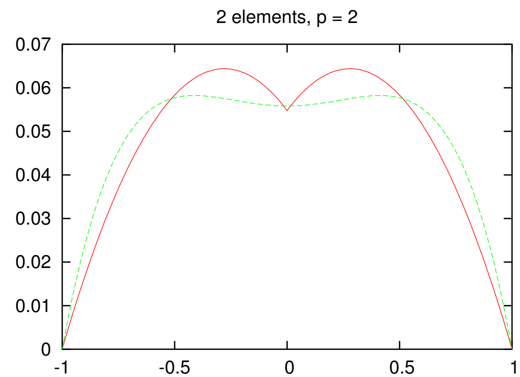
p Enrichment

Instead of refining an element: increasing polynomial degree p. Good for smooth functions.

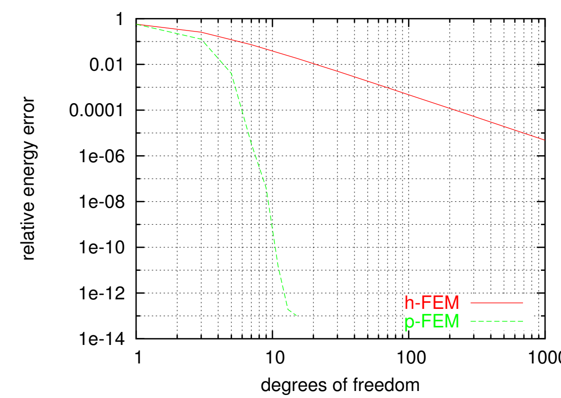
h FEM:



p FEM:



Convergence:



Solving the problem

$$-\Delta u + u = x^2 \text{ in } \Omega = (-1,1)$$

$$u = 0 \text{ on } \partial\Omega = \{-1,1\},$$

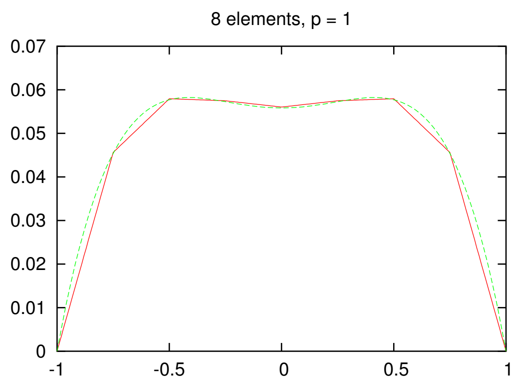
$$\Rightarrow u(x) = -3\frac{\cosh(x)}{\cosh(1)} + x^2 + 2.$$

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

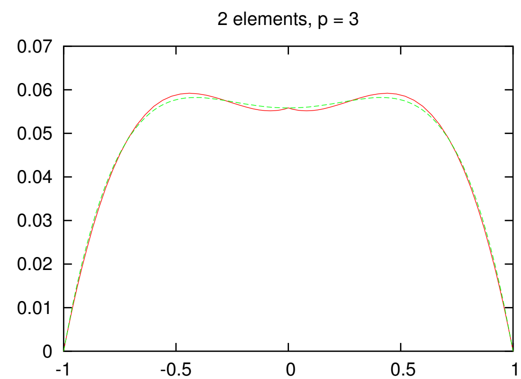
p Enrichment

Instead of refining an element: increasing polynomial degree p. Good for smooth functions.

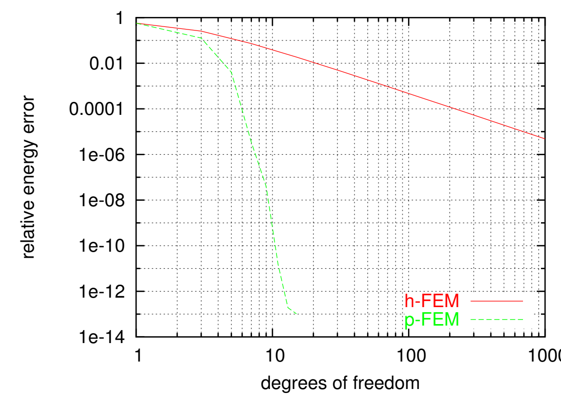
h FEM:



p FEM:



Convergence:



Solving the problem

$$-\Delta u + u = x^2 \text{ in } \Omega = (-1,1)$$

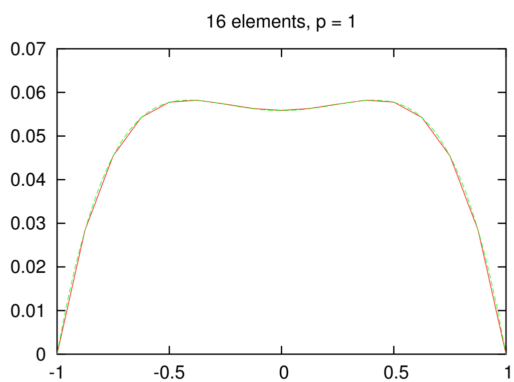
$$u = 0 \text{ on } \partial\Omega = \{-1,1\},$$

$$\Rightarrow u(x) = -3\frac{\cosh(x)}{\cosh(1)} + x^2 + 2.$$

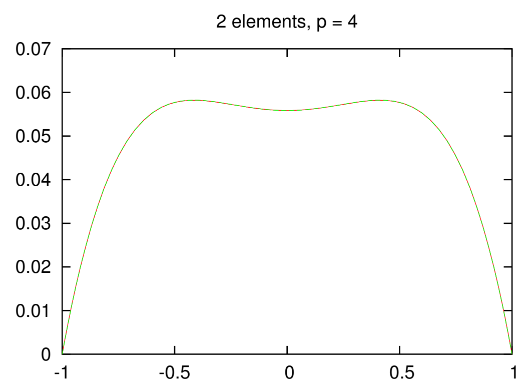
p Enrichment

Instead of refining an element: increasing polynomial degree p. Good for smooth functions.

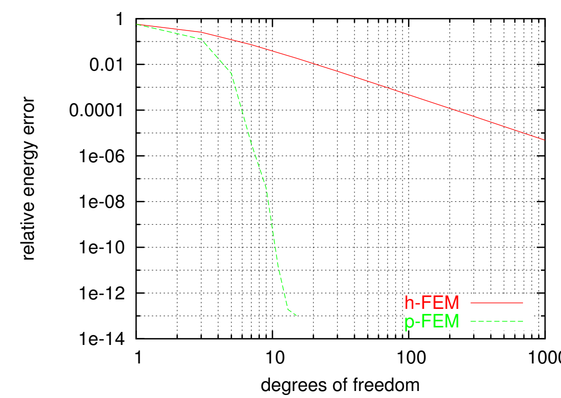
h FEM:



p FEM:



Convergence:



Solving the problem

$$-\Delta u + u = x^2 \text{ in } \Omega = (-1,1)$$

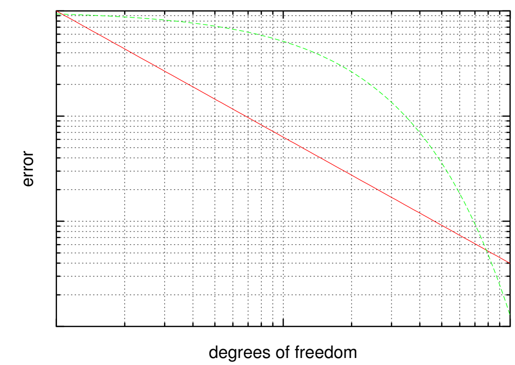
$$u = 0 \text{ on } \partial\Omega = \{-1,1\},$$

$$\Rightarrow u(x) = -3\frac{\cosh(x)}{\cosh(1)} + x^2 + 2.$$

Exponential Convergence

Theorem: Let $\Omega \subset \mathbb{R}^n$ $(n = 2, 3)$ a polyhedron, $V_N = S^{1,\underline{p}}(\Omega, \mathcal{T}) \ni u_N$ the FE space and $u \in H^k(\Omega)$, $k \geq 1$ the exact solution.

Then: $\|u - u_N\| \leq c \left(\frac{h}{p}\right)^{\min(p,k-1)} \|u\|_{H^k(\Omega)}$.



h FEM: $\|u - u_N\| \leq c_1 h^{c_2}$ algebraic convergence

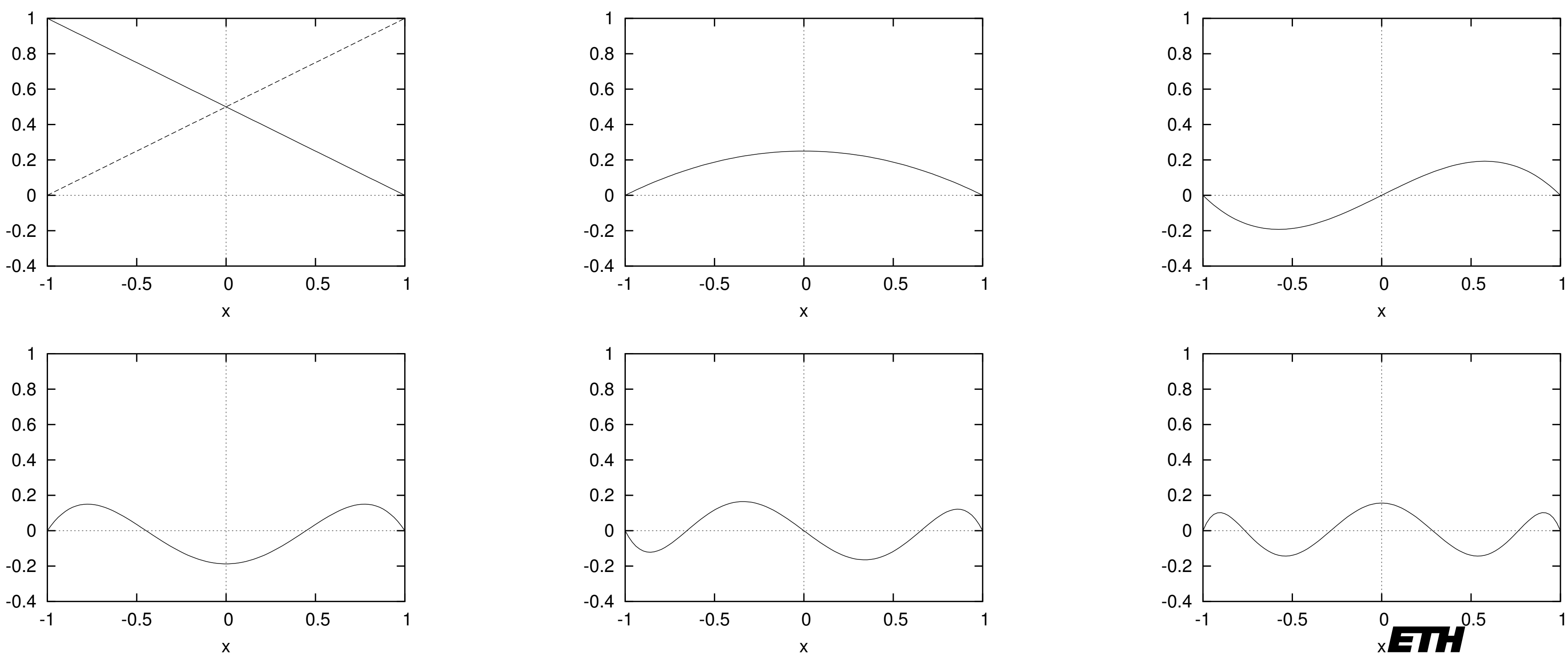
p FEM: $\|u - u_N\| \leq c \left(\frac{1}{p}\right)^p$ if $u \in \mathcal{C}^\infty$ exponential convergence

hp FEM: close to singularities: h FEM
in regular areas: p FEM
$\Rightarrow \|u - u_N\| \leq c \exp(-bN^\alpha)$ if $u \in B^2_\beta$ exponential convergence
$\alpha = 3$ in $\mathbb{R}^2$
$\alpha = 5$ in $\mathbb{R}^3$

Shape Functions in 1D

The reference element shape functions on $(-1, 1)$ of order p:

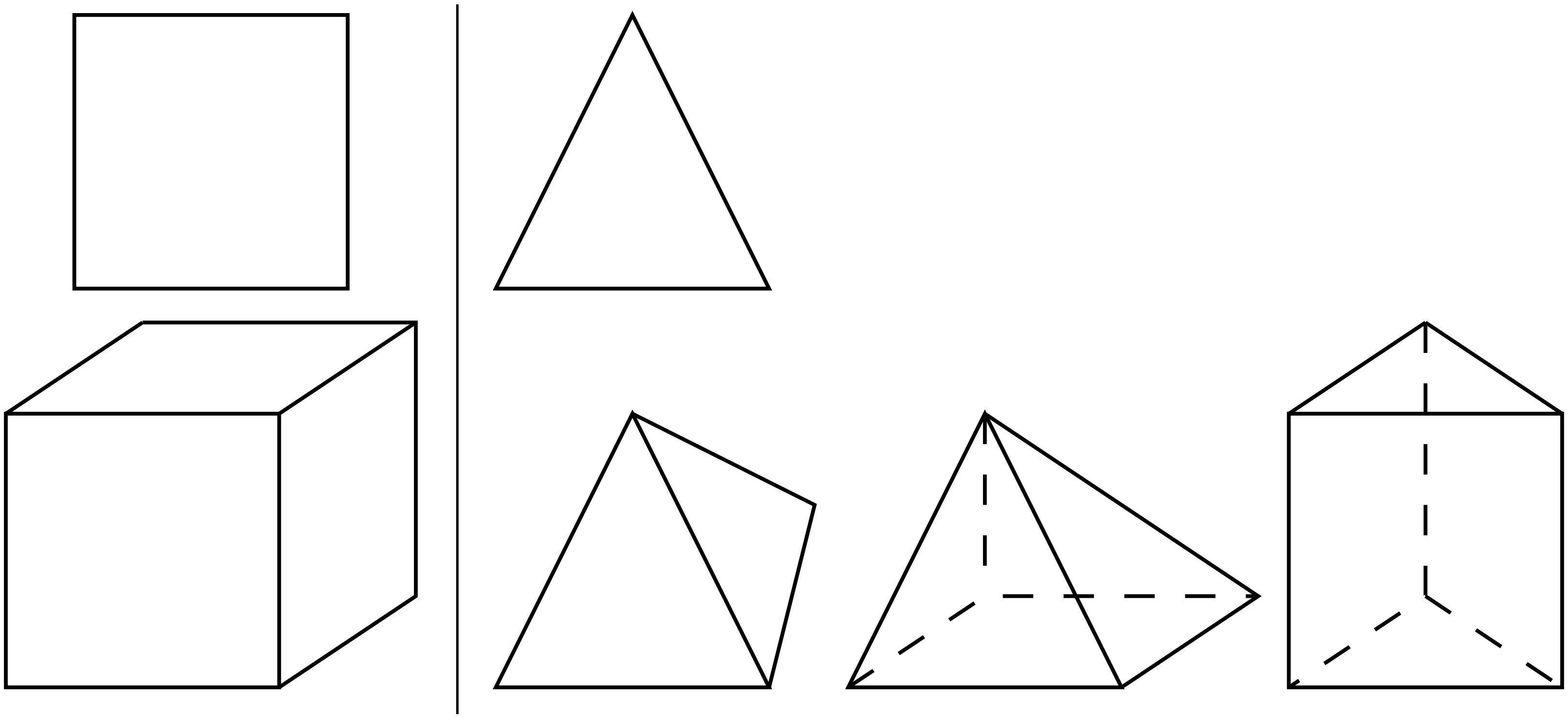
$$N_i(\xi) = \begin{cases} \frac{1-\xi}{2} & i = 0 \\ \frac{1-\xi}{2}\frac{1+\xi}{2} P_{i-1}^{1,1}(\xi) & 1 \le i \le p-1 \\ \frac{1+\xi}{2} & i = p \end{cases}$$



Element Types in 2D and 3D

2D:

3D:



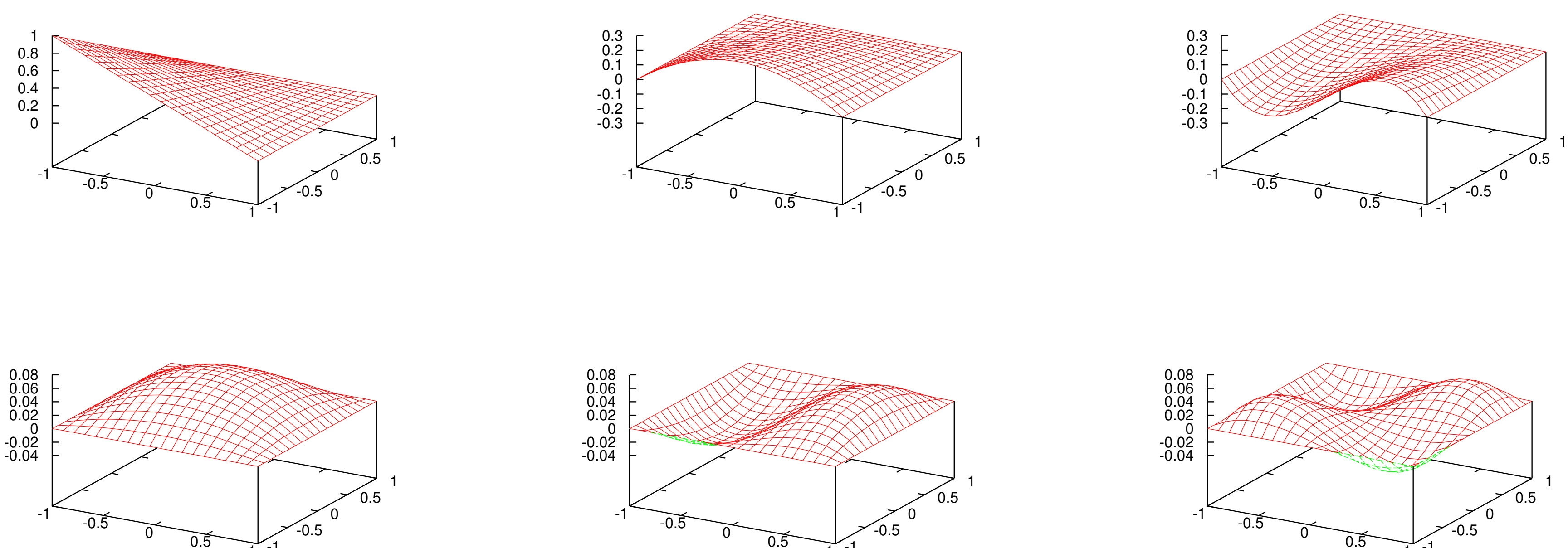
Quads and Hexahedra: fully anisotropic, variable degree p.

Shape Functions in 2D

The reference element shape functions on $(-1,1)^2$ of order (p,q) are tensor product functions of 1D shape functions:

$$N_{i,j} = N_i \otimes N_j$$
$$N_{i,j}(\xi,\eta) = N_i(\xi)\cdot N_j(\eta)$$

$$N_i(\xi) = \begin{cases} \frac{1-\xi}{2} & i = 0 \\ \frac{1-\xi}{2}\frac{1+\xi}{2}P_{i-1}^{1,1}(\xi) & 1 \le i \le p-1 \\ \frac{1+\xi}{2} & i = p \end{cases}$$



Overview

- Introduction: FEM & Exponential Convergence
- Assembling
- Handling Hanging Nodes
- Finding Regular Supports
- Maxwell Eigenvalue Problems
- Perspectives

T Matrix

Definition 1 (T Matrix). *Element shape functions* $\{\phi_j^K\}_{j=1}^{m_K}$ *on element* K*, global basis functions* $\{\Phi_i\}_{i=1}^{N}$*.*
The T matrix $\boldsymbol{T}_K \in \mathbb{R}^{m_K \times N}$ *of element* K *is implicitly defined by*

$$\Phi_i|_K = \sum_{j=1}^{m_K} [\boldsymbol{T}_K]_{ji}\, \phi_j^K$$

as vectors:

$$\underline{\Phi}|_K = \boldsymbol{T}_K^\top \underline{\phi}^K .$$

Assembly using T Matrices

Stiffness matrix: $A_{ij} = a(\phi_i, \phi_j)$, load vector: $l_i = l(\phi_i)$.
Assembling:

$$\underline{l} = l(\underline{\Phi}) = l\Big(\sum_{\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} \underline{\phi}^{\tilde{K}}\Big) = \sum_{\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} l(\underline{\phi}^{\tilde{K}}) = \sum_{\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} \underline{l}_{\tilde{K}}$$

Assembly using T Matrices

Stiffness matrix: $A_{ij} = a(\phi_i, \phi_j)$, load vector: $l_i = l(\phi_i)$.
Assembling:

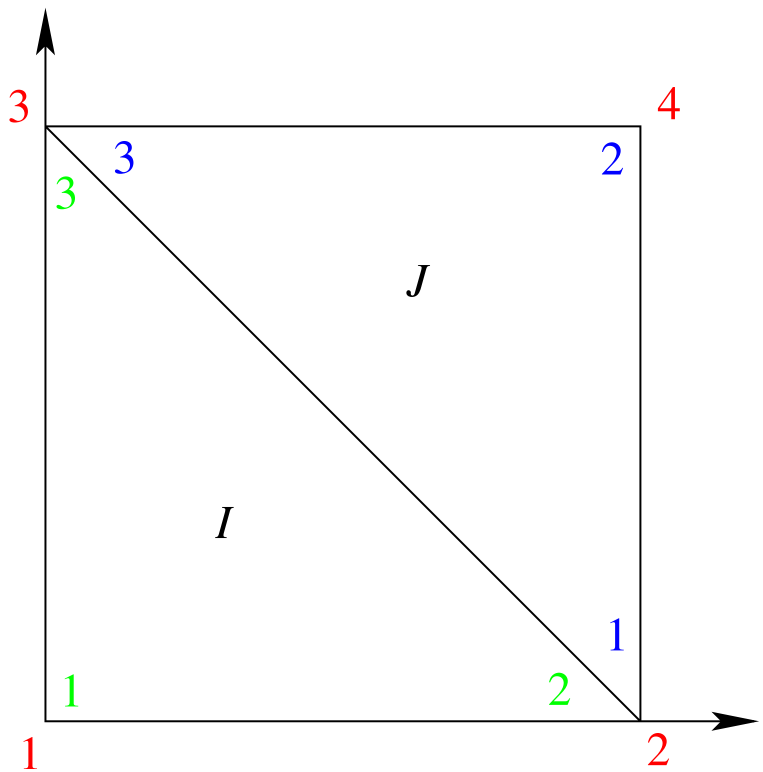
$$\underline{l} = l(\underline{\Phi}) = l\Big(\sum_{\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} \underline{\phi}^{\tilde{K}}\Big) = \sum_{\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} l(\underline{\phi}^{\tilde{K}}) = \sum_{\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} \underline{l}_{\tilde{K}}$$

$$\boldsymbol{A} = a(\underline{\Phi}, \underline{\Phi}) = \sum_{K,\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} a(\underline{\phi}^{K}, \underline{\phi}^{\tilde{K}}) \boldsymbol{T}_K = \sum_{K,\tilde{K}} \boldsymbol{T}_{\tilde{K}}^{\top} \boldsymbol{A}_{\tilde{K}K} \boldsymbol{T}_K$$

Note: $\boldsymbol{A}_{\tilde{K}K} = 0$ in standard FEM for $\tilde{K} \neq K$.

Example 1: Regular Mesh

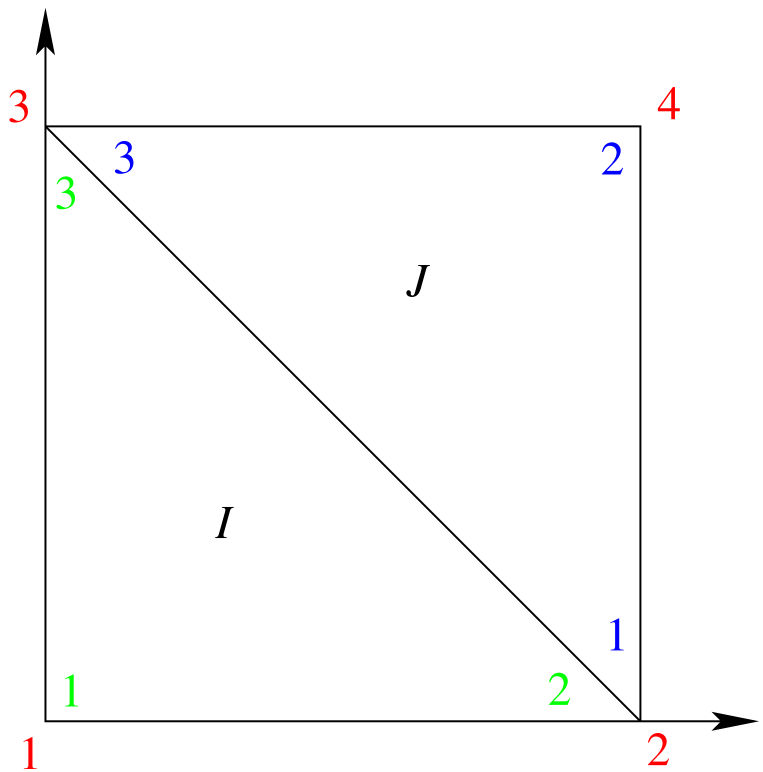
Two elements with three local shape functions each and four global basis functions.



$$\boldsymbol{T}_I = \begin{pmatrix} & 1 & 2 & 3 & 4 \\ 1 & 1 & 0 & 0 & 0 \\ 2 & 0 & 1 & 0 & 0 \\ 3 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Example 1: Regular Mesh

Two elements with three local shape functions each and four global basis functions.

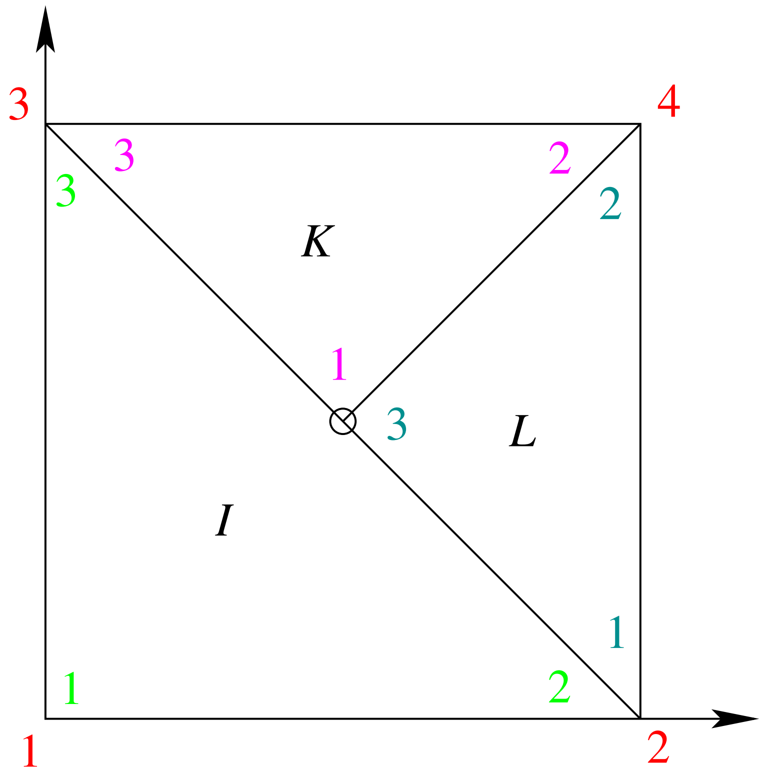


$$\boldsymbol{T}_I = \begin{pmatrix} & 1 & 2 & 3 & 4 \\ 1 & 1 & 0 & 0 & 0 \\ 2 & 0 & 1 & 0 & 0 \\ 3 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\boldsymbol{T}_J = \begin{pmatrix} & 1 & 2 & 3 & 4 \\ 1 & 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 1 \\ 3 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Example 2: Irregular Mesh

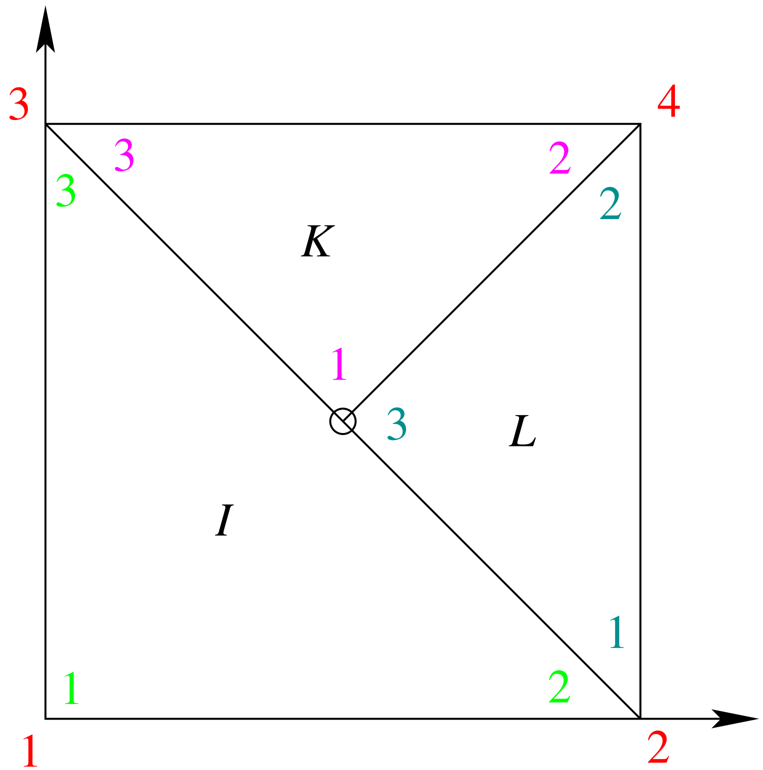
Three elements with three local shape functions each and four global basis functions. The hanging node is marked with $\circ$.



$$\boldsymbol{T}_L = \begin{pmatrix} & 1 & 2 & 3 & 4 \\ 1 & 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 1 \\ 3 & 0 & 1/2 & 1/2 & 0 \end{pmatrix}$$

Example 2: Irregular Mesh

Three elements with three local shape functions each and four global basis functions. The hanging node is marked with $\circ$.



$$\boldsymbol{T}_L = \begin{pmatrix} & 1 & 2 & 3 & 4 \\ 1 & 0 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 1 \\ 3 & 0 & 1/2 & 1/2 & 0 \end{pmatrix}$$

$$\boldsymbol{T}_K = \begin{pmatrix} & 1 & 2 & 3 & 4 \\ 1 & 0 & 1/2 & 1/2 & 0 \\ 2 & 0 & 0 & 0 & 1 \\ 3 & 0 & 0 & 1 & 0 \end{pmatrix}$$

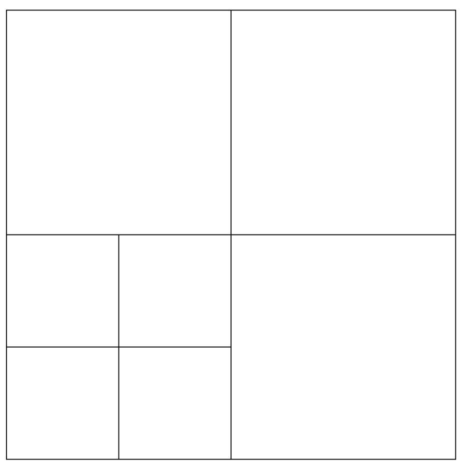
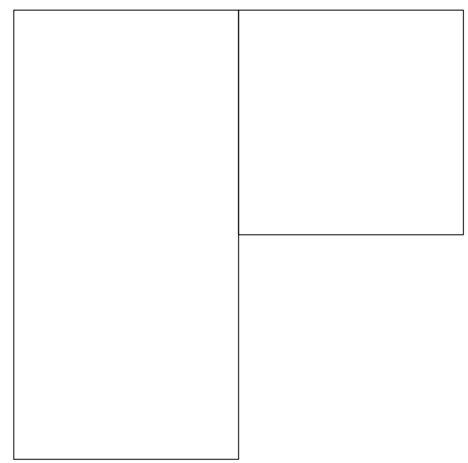
$\Rightarrow$ continuous basis functions.

Generation of T Matrices

- **Regular Mesh:** Counting and assigning indices with respect to topological entities such as vertices, edges and faces. Explained in detail later.

Generation of T Matrices

- **Regular Mesh:** Counting and assigning indices with respect to topological entities such as vertices, edges and faces. Explained in detail later.
- **Irregular Mesh:** Irregularity due to a refinement of an initially regular mesh.

OK: , not OK: 

Explanation follows.

T Matrices for Irregular Meshes

Irregularity due to a refinement of an initially regular mesh.

Mesh	$\mathcal{M}$	refine	$\mathcal{M}'$
Basis fcts.	$B = B_{\text{repl}} \cup B_{\text{keep}}$	$\longrightarrow$	$B' = B_{\text{ins}} \cup B_{\text{keep}}$

T Matrices for Irregular Meshes

Irregularity due to a refinement of an initially regular mesh.

Mesh	$\mathcal{M}$	refine	$\mathcal{M}'$
Basis fcts.	$B = B_{\text{repl}} \cup B_{\text{keep}}$	$\longrightarrow$	$B' = B_{\text{ins}} \cup B_{\text{keep}}$

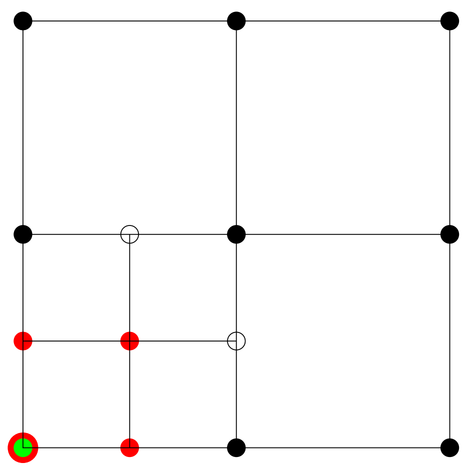
B_{repl}: basis fcts. which can be solely described by elements of $\mathcal{M}' \backslash \mathcal{M}$

B_{ins}: basis fcts. generated by regular parts of $\mathcal{M}' \backslash \mathcal{M}$

T Matrices for Irregular Meshes

Irregularity due to a refinement of an initially regular mesh.

Mesh	$\mathcal{M}$	refine	$\mathcal{M}'$
Basis fcts.	$B = B_{\text{repl}} \cup B_{\text{keep}}$	$\longrightarrow$	$B' = B_{\text{ins}} \cup B_{\text{keep}}$



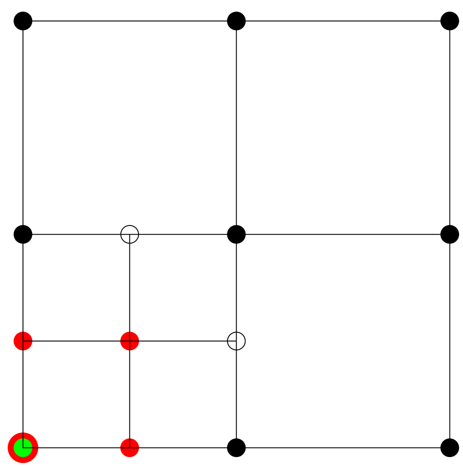
B_{repl}: basis fcts. which can be solely described by elements of $\mathcal{M}'\backslash\mathcal{M}$

B_{ins}: basis fcts. generated by regular parts of $\mathcal{M}'\backslash\mathcal{M}$

T Matrices for Irregular Meshes

Irregularity due to a refinement of an initially regular mesh.

Mesh	$\mathcal{M}$	refine	$\mathcal{M}'$
Basis fcts.	$B = B_{\mathrm{repl}} \cup B_{\mathrm{keep}}$	$\longrightarrow$	$B' = B_{\mathrm{ins}} \cup B_{\mathrm{keep}}$



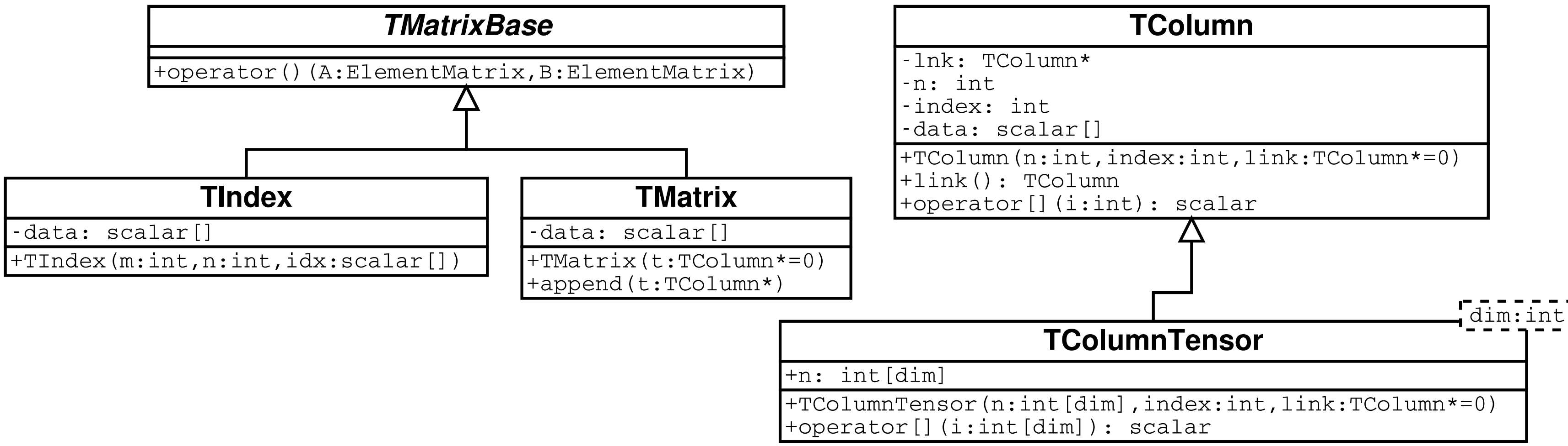
B_{repl}: basis fcts. which can be solely described by elements of $\mathcal{M}'\backslash\mathcal{M}$

B_{ins}: basis fcts. generated by regular parts of $\mathcal{M}'\backslash\mathcal{M}$

Every element of B has a column in the T matrix. Generation is

- easy for B_{ins} (like regular mesh),
- simple for B_{keep}: modify column from $\mathcal{M}$ by S matrix.

Classes for T Matrices



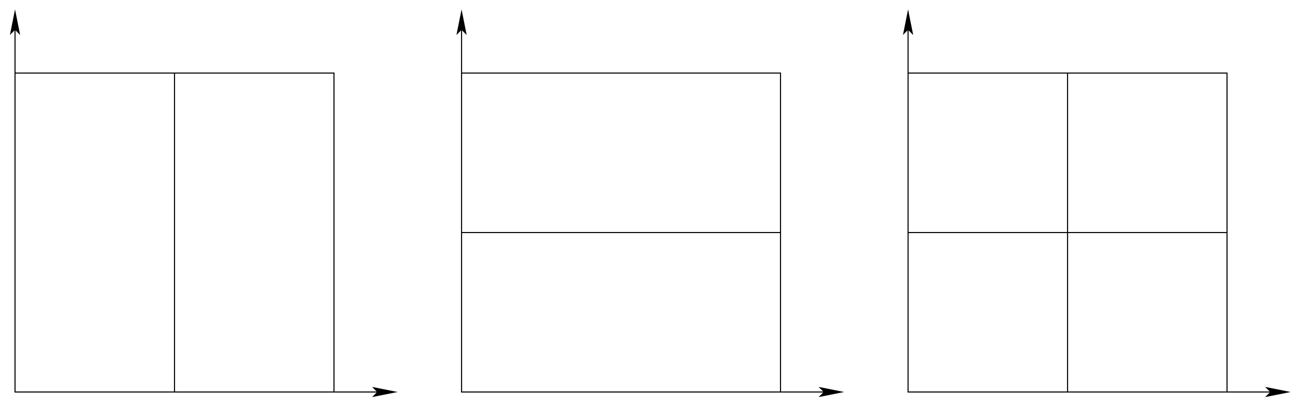
- Different implementations for a T Matrix: TIndex and TMatrix
- TColumn represents a column of a T matrix: coefficients of a global degree of freedom in a particular element
- TColumnTensor: different interface to the data of TColumn with multi-indices

Overview

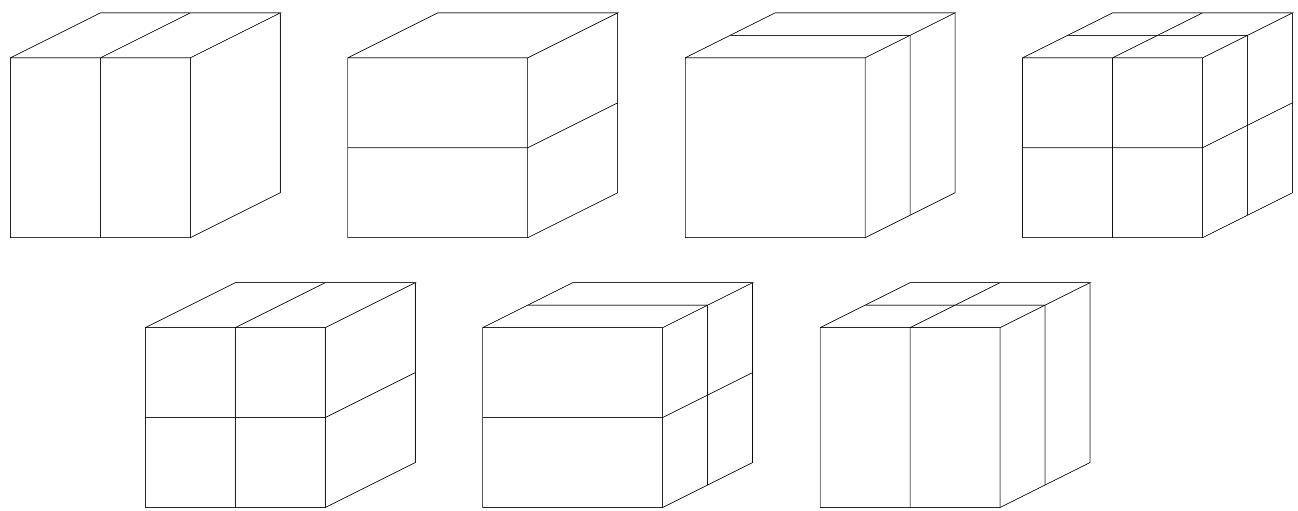
- Introduction: FEM & Exponential Convergence
- Assembling
- Handling Hanging Nodes
- Finding Regular Supports
- Maxwell Eigenvalue Problems
- Perspectives

Subdivisions

Subdivisions of a quadrilateral in 2D:



Subdivisions of a hexahedron in 3D:



S Matrix

Definition 2 (S Matrix). *Let $K' \subset K$ be the result of a refinement of element K. The S matrix $\boldsymbol{S}_{K'K} \in \mathbb{R}^{m_{K'} \times m_K}$ is defined by*

$$\phi_j^K\Big|_{K'} = \sum_{l=1}^{m_{K'}} [\boldsymbol{S}_{K'K}]_{lj}\, \phi_l^{K'}$$

as vectors:

$$\underline{\phi}^K\Big|_{K'} = \boldsymbol{S}_{K'K}^{\top} \underline{\phi}^{K'}$$

$\phi_j^K\big|_{K'}$ is represented as a linear combination of the shape functions $\left\{\phi_l^{K'}\right\}_{l=1}^{m_{K'}}$ of K'.

Application of S Matrix

Proposition 1. *Let $K' \subset K$ be the result of a refinement of an element K. Then, the T matrix of K' can be computed as*

$$\boldsymbol{T}_{K'} = \boldsymbol{S}_{K'K}\boldsymbol{T}_K^{\mathrm{keep}} + \boldsymbol{T}_{K'}^{\mathrm{ins}}$$

where $\boldsymbol{T}_K^{\mathrm{keep}}$ denotes the T matrix of element K (with columns not related to functions in B_{keep} set to zero) and $\boldsymbol{T}_{K'}^{\mathrm{ins}}$ the T matrix for functions in B_{ins} with respect to K'.

Proposition 2. *Let $\hat{K}' \subset \hat{K}$ be the result of a refinement of the reference element $\hat{K}$ with $H : \hat{K} \to \hat{K}'$ the subdivision map. The element maps are*

$$F_K : \hat{K} \to K \text{ and } F_{K'} : \hat{K} \to K'$$

and $F_{K'} \circ H^{-1} = F_K$ holds. Then, $\boldsymbol{S}_{\hat{K}'\hat{K}} = \boldsymbol{S}_{K'K}$.

S Matrix in Dimension $d = 1$

Subdividing $\hat{J} = (0,1)$ in $\hat{J}' = (0, 1/2)$ and $\hat{J}^\star = (1/2, 1)$ with the reference element shape functions

$$N_j(\xi) = \begin{cases} 1-\xi & j = 1 \\ \xi & j = 2 \\ \xi(1-\xi)P^{1,1}_{j-3}(2\xi - 1) & j = 3, \dots, J \end{cases}$$

yields (solving a linear system) for $J = 4$:

$$\boldsymbol{S}_{\hat{J}',\hat{J}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1/2 & 1/2 & 1/4 & 0 \\ 0 & 0 & 1/4 & -3/4 \\ 0 & 0 & 0 & 1/8 \end{pmatrix} \text{ and } \boldsymbol{S}_{\hat{J}^\star,\hat{J}} = \begin{pmatrix} 1/2 & 1/2 & 1/4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/4 & 3/4 \\ 0 & 0 & 0 & 1/8 \end{pmatrix}.$$

Hierarchic shape functions $\Rightarrow$ hierarchic S matrices.

S Matrices: Tensor Product in 2D – I

- $d > 1$ with hexahedral meshes $\Rightarrow$ S matrices are built from tensor products of 1D S matrices.

S Matrices: Tensor Product in 2D – I

- $d > 1$ with hexahedral meshes $\Rightarrow$ S matrices are built from tensor products of 1D S matrices.
- In 2D: $N_{i,j} = N_i \otimes N_j$, the four bilinear shape functions are:

$$N_{1,2}(\underline{\xi}) = N_1(\xi_1) \cdot N_2(\xi_2) \qquad N_{2,2}(\underline{\xi}) = N_2(\xi_1) \cdot N_2(\xi_2)$$

$$N_{1,1}(\underline{\xi}) = N_1(\xi_1) \cdot N_1(\xi_2) \qquad N_{2,1}(\underline{\xi}) = N_2(\xi_1) \cdot N_1(\xi_2)$$

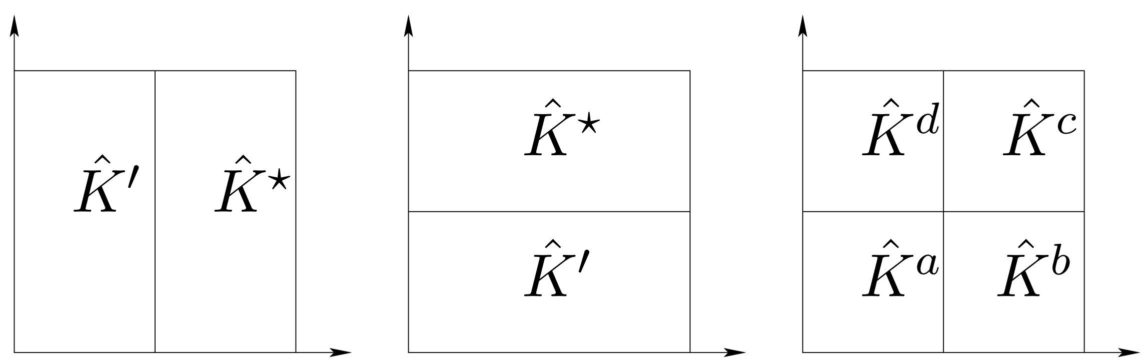
S Matrices: Tensor Product in 2D – I

- $d > 1$ with hexahedral meshes $\Rightarrow$ S matrices are built from tensor products of 1D S matrices.
- In 2D: $N_{i,j} = N_i \otimes N_j$, the four bilinear shape functions are:

$$N_{1,2}(\underline{\xi}) = N_1(\xi_1) \cdot N_2(\xi_2) \qquad N_{2,2}(\underline{\xi}) = N_2(\xi_1) \cdot N_2(\xi_2)$$

$$N_{1,1}(\underline{\xi}) = N_1(\xi_1) \cdot N_1(\xi_2) \qquad N_{2,1}(\underline{\xi}) = N_2(\xi_1) \cdot N_1(\xi_2)$$

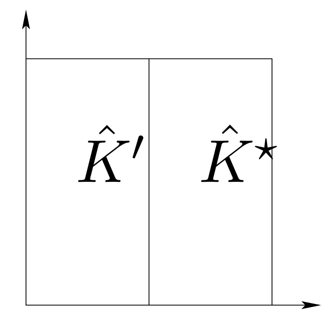
- Consider the subdivisions:



S Matrices: Tensor Product in 2D – II

Subdivision map of left variant: $H : \hat{K} \to \hat{K}', \underline{\xi} \mapsto \begin{pmatrix} \xi_1/2 \\ \xi_2 \end{pmatrix}$. S matrix $\boldsymbol{S}_{\hat{K}'\hat{K}}$ is defined by:

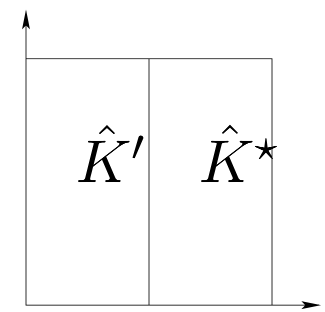
$$N_{i,j}|_{\hat{K}'} = \sum_{k,l} \left[\boldsymbol{S}_{\hat{K}'\hat{K}}\right]_{(k,l),(i,j)} N_{k,l} \circ H^{-1}.$$



S Matrices: Tensor Product in 2D – II

Subdivision map of left variant: $H : \hat{K} \to \hat{K}', \underline{\xi} \mapsto \begin{pmatrix} \xi_1/2 \\ \xi_2 \end{pmatrix}$. S matrix $\boldsymbol{S}_{\hat{K}'\hat{K}}$ is defined by:

$$N_{i,j}|_{\hat{K}'} = \sum_{k,l} \left[\boldsymbol{S}_{\hat{K}'\hat{K}}\right]_{(k,l),(i,j)} N_{k,l} \circ H^{-1}.$$



Tensor product shape functions:

$$(N_i \otimes N_j)|_{\hat{K}'} = \sum_{k,l} \left[\boldsymbol{S}_{\hat{K}'\hat{K}}\right]_{(k,l),(i,j)} (N_k \otimes N_l) \circ H^{-1}. \quad (1)$$

S Matrices: Tensor Product in 2D – III

S matrices for 1D reference element shape fcts. used in (1):

$$N_i|_{\hat{\jmath}'} = \sum_m [\boldsymbol{S}_{\hat{\jmath}',\hat{\jmath}}]_{mi} N_m \circ G^{-1}$$ for the ξ_1 part and

$$N_j = \sum_n [\boldsymbol{E}]_{nj} N_n$$ for the ξ_2 part,

where $G : \xi \mapsto \xi/2$.

S Matrices: Tensor Product in 2D – III

S matrices for 1D reference element shape fcts. used in (1):

$$N_i|_{\hat{j}'} = \sum_m \left[\boldsymbol{S}_{\hat{j}',\hat{j}}\right]_{mi} N_m \circ G^{-1}$$ for the ξ_1 part and

$$N_j = \sum_n [\boldsymbol{E}]_{nj} N_n$$ for the ξ_2 part,

where $G : \xi \mapsto \xi/2$. Plugging into the left hand side of (1) yields:

$$\begin{aligned}(N_i \otimes N_j)|_{\hat{K}'} = N_i|_{\hat{j}'} \otimes N_j &= \sum_{m,n} \left(\left[\boldsymbol{S}_{\hat{j}',\hat{j}}\right]_{mi} N_m \circ G^{-1}\right) \otimes \left([\boldsymbol{E}]_{nj} N_n\right) \\ &= \sum_{m,n} \left[\boldsymbol{S}_{\hat{j}',\hat{j}}\right]_{mi} \cdot [\boldsymbol{E}]_{nj} N_m \circ G^{-1} \otimes N_n .\end{aligned}$$

S Matrices: Tensor Product in 2D – IV

Comparing with the right hand side of (1):

$$\sum_{m,n} \left[\boldsymbol{S}_{\hat{J}',\hat{J}}\right]_{mi} \cdot [\boldsymbol{E}]_{nj}\, N_m \circ G^{-1} \otimes N_n$$

$$= \sum_{k,l} \left[\boldsymbol{S}_{\hat{K}'\hat{K}}\right]_{(k,l),(i,j)} N_k \circ G^{-1} \otimes N_l.$$

S Matrices: Tensor Product in 2D – IV

Comparing with the right hand side of (1):

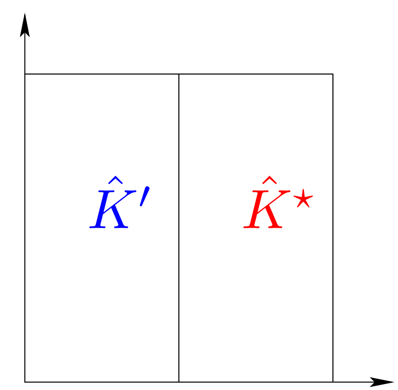
$$\sum_{m,n} [\boldsymbol{S}_{\hat{J}'\hat{J}}]_{mi} \cdot [\boldsymbol{E}]_{nj} \, N_m \circ G^{-1} \otimes N_n$$

$$= \sum_{k,l} [\boldsymbol{S}_{\hat{K}'\hat{K}}]_{(k,l),(i,j)} \, N_k \circ G^{-1} \otimes N_l .$$

Therefore for the vertical subdivision:

$\boldsymbol{S}_{\hat{K}'\hat{K}} = \boldsymbol{S}_{\hat{J}'\hat{J}} \otimes \boldsymbol{E}$ for the left quad $\hat{K}'$,

$\boldsymbol{S}_{\hat{K}^\star\hat{K}} = \boldsymbol{S}_{\hat{J}^\star\hat{J}} \otimes \boldsymbol{E}$ for the right quad $\hat{K}^\star$.

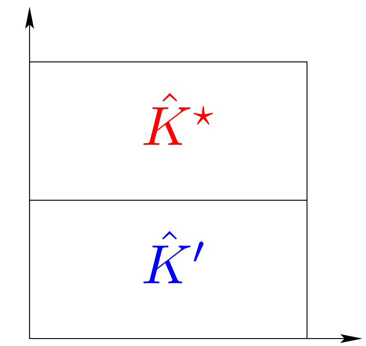


S Matrices: Tensor Product in 2D – V

Horizontal subdivision:

$\boldsymbol{S}_{\hat{K}'\hat{K}} = \boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}'\hat{J}}$ for the bottom quad $\hat{K}'$,

$\boldsymbol{S}_{\hat{K}^\star\hat{K}} = \boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}^\star\hat{J}}$ for the top quad $\hat{K}^\star$.

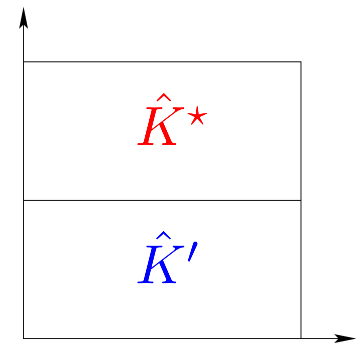


S Matrices: Tensor Product in 2D – V

Horizontal subdivision:

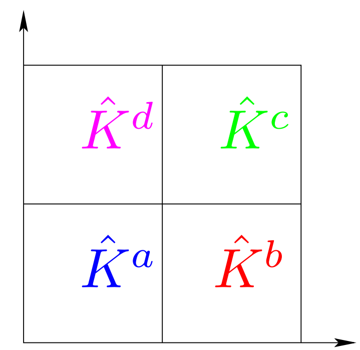
$$\boldsymbol{S}_{\hat{K}'\hat{K}} = \boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}'\hat{J}} \qquad \text{for the bottom quad } \hat{K}',$$

$$\boldsymbol{S}_{\hat{K}^\star\hat{K}} = \boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}^\star\hat{J}} \qquad \text{for the top quad } \hat{K}^\star.$$



Subdivision into four quads:

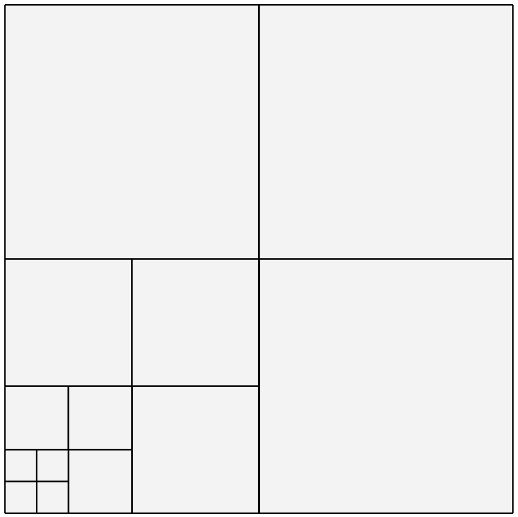
- subdivide $\hat{K}$ horizontally into two children
- subdivide upper and lower
  child vertically into $\hat{K}^d$ and $\hat{K}^c$ and $\hat{K}^a$ and $\hat{K}^b$ resp.



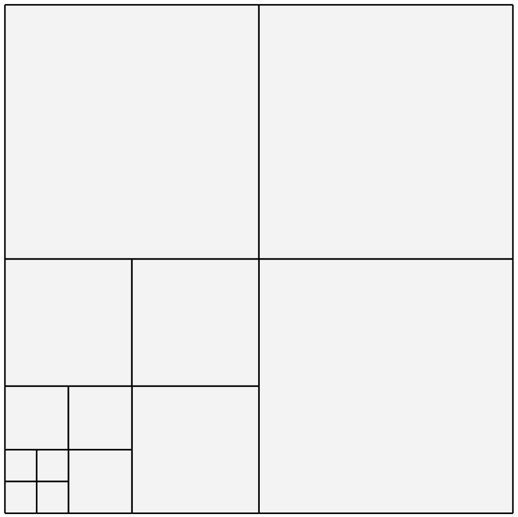
$$\boldsymbol{S}_{\hat{K}^d\hat{K}} = \left(\boldsymbol{S}_{\hat{J}'\hat{J}} \otimes \boldsymbol{E}\right) \cdot \left(\boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}^\star\hat{J}}\right) \qquad \boldsymbol{S}_{\hat{K}^c\hat{K}} = \left(\boldsymbol{S}_{\hat{J}^\star\hat{J}} \otimes \boldsymbol{E}\right) \cdot \left(\boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}^\star\hat{J}}\right)$$

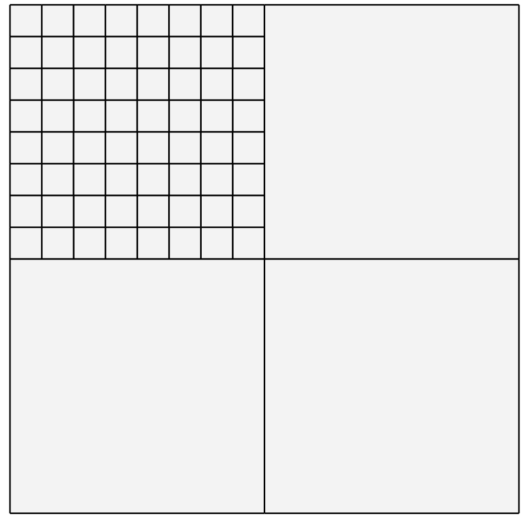
$$\boldsymbol{S}_{\hat{K}^a\hat{K}} = \left(\boldsymbol{S}_{\hat{J}'\hat{J}} \otimes \boldsymbol{E}\right) \cdot \left(\boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}'\hat{J}}\right) \qquad \boldsymbol{S}_{\hat{K}^b\hat{K}} = \left(\boldsymbol{S}_{\hat{J}^\star\hat{J}} \otimes \boldsymbol{E}\right) \cdot \left(\boldsymbol{E} \otimes \boldsymbol{S}_{\hat{J}'\hat{J}}\right)$$

Meshes

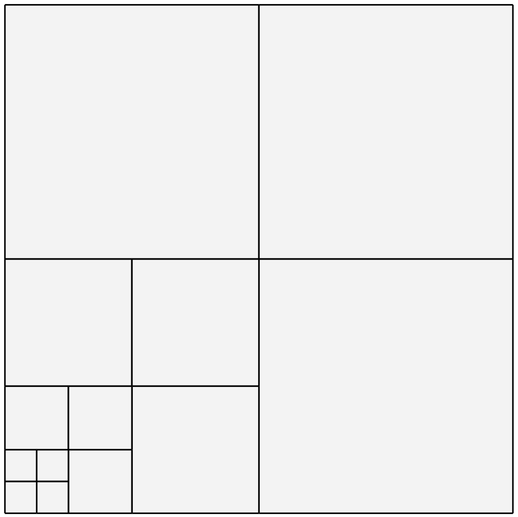


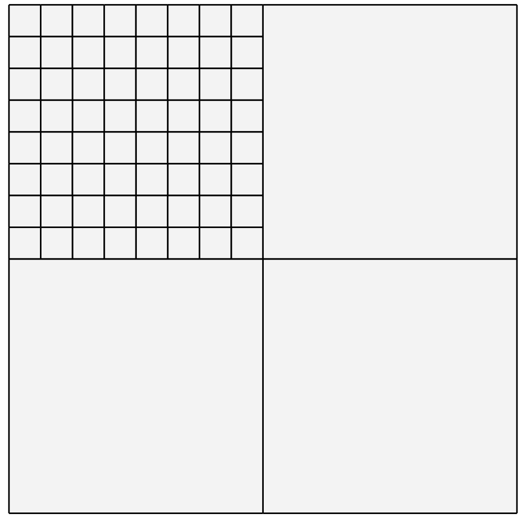
Meshes

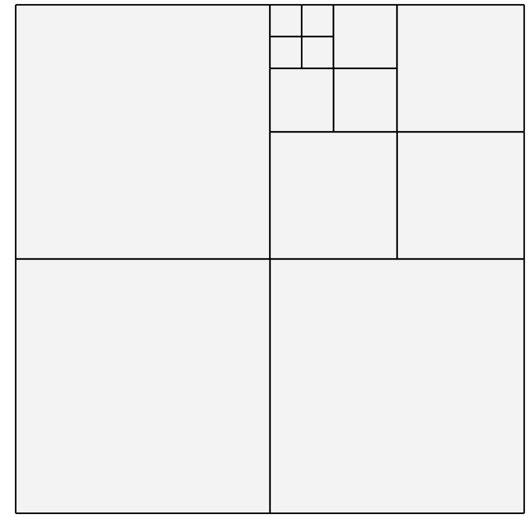




Meshes





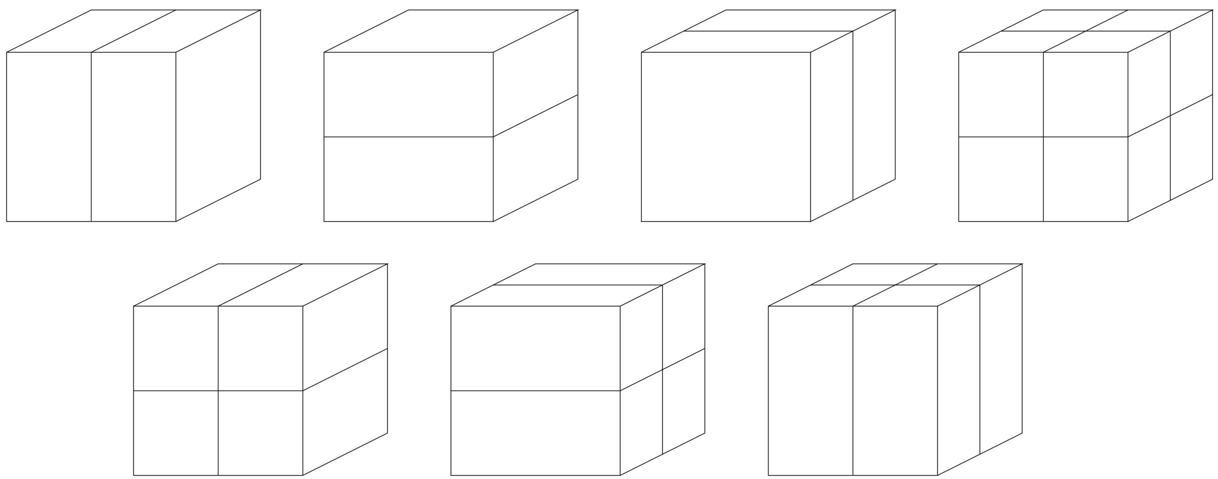


S Matrices: Tensor-Product in 3D

Same idea as in 2D, just of this form:

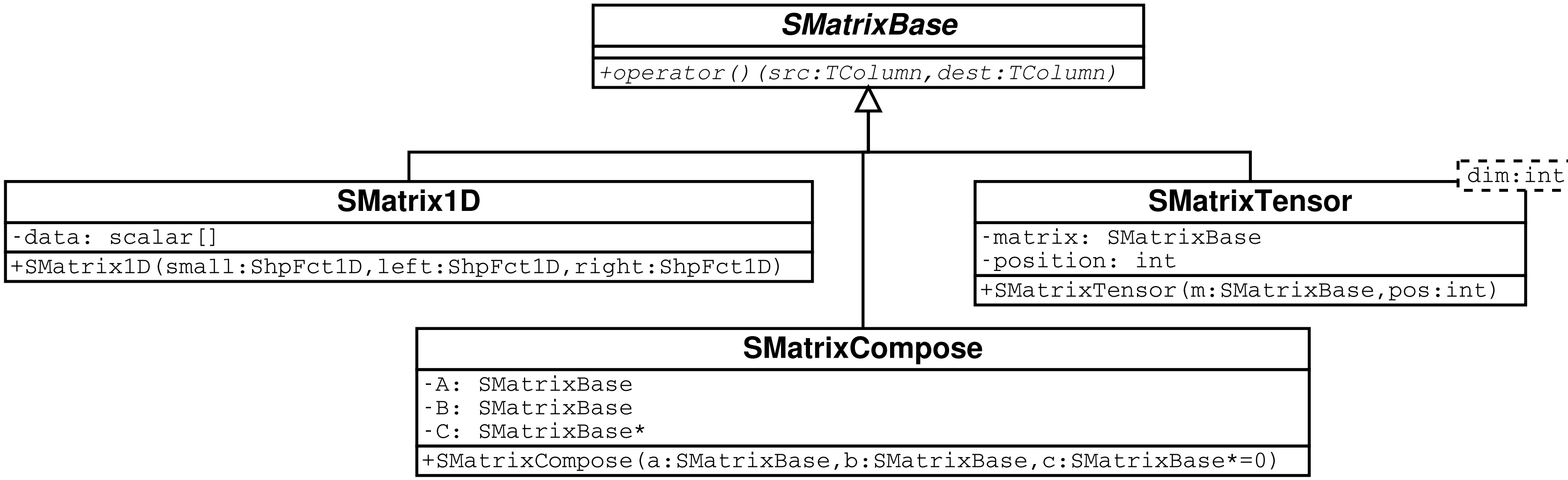
$$S_{\hat{K}'\hat{K}} = \prod (A \otimes B \otimes C)$$

in each of the factors, one of A, B or C is an 1D S matrix.
Depending on the factors, 7 subdivisions are possible:



Concepts: arbitrary number and combination of these 7 subdivisions in 3D.

Classes for S Matrices



- SMatrix1D contains the coefficients, all other classes make use of that
- SMatrixCompose implements $\cdot$
- SMatrixTensor implements $\otimes$

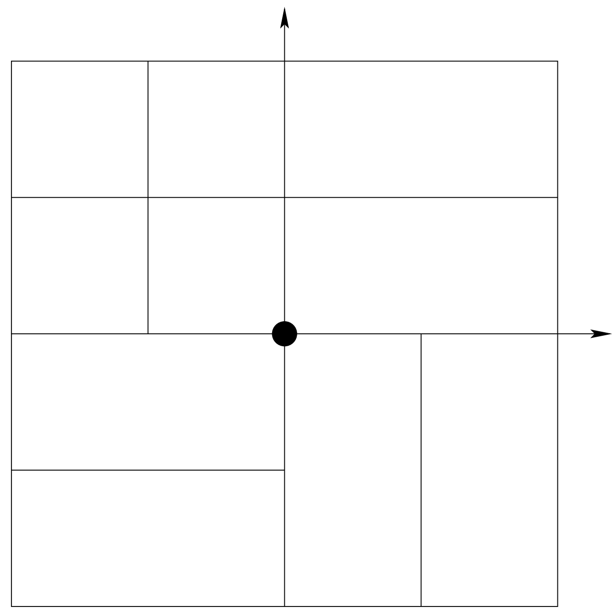
ETH Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Overview

- Introduction: FEM & Exponential Convergence
- Assembling
- Handling Hanging Nodes
- Finding Regular Supports
- Maxwell Eigenvalue Problems
- Perspectives

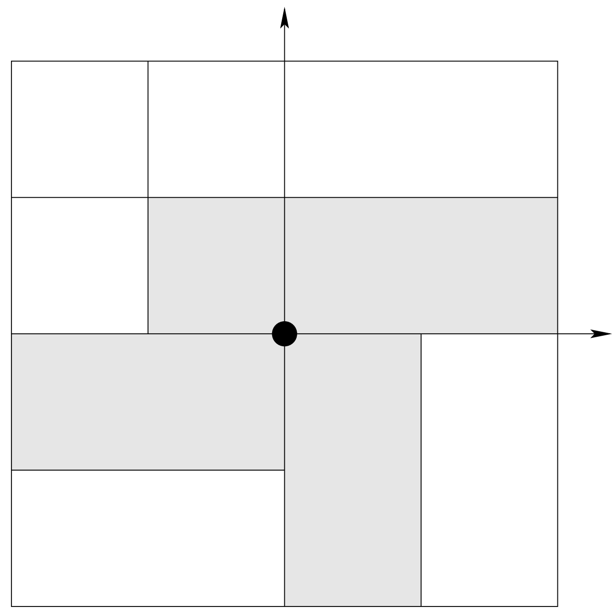
Anisotropic and Conforming

Main point: find the cells (either coarse or fine) which belong to the support of a certain basis function.



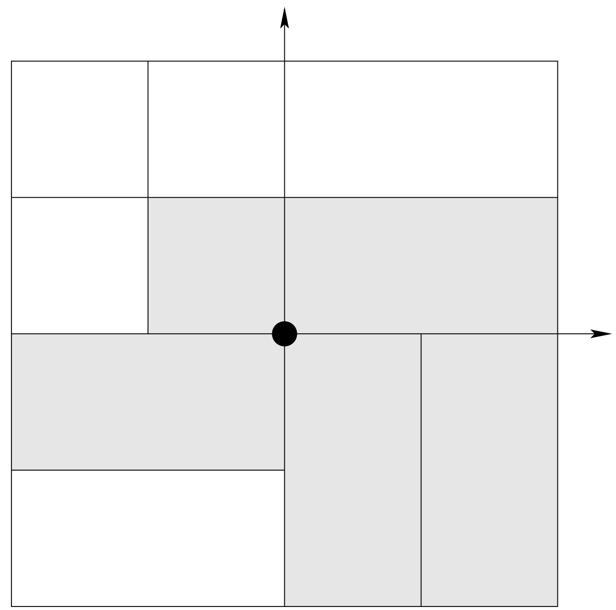
Anisotropic and Conforming

Main point: find the cells (either coarse or fine) which belong to the support of a certain basis function.



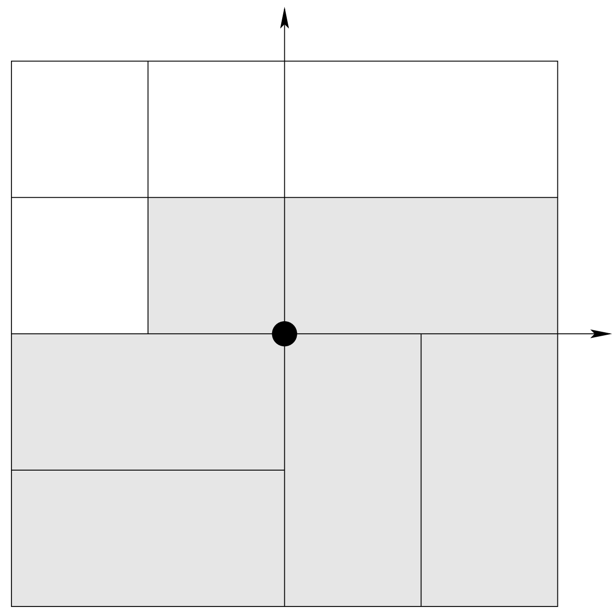
Anisotropic and Conforming

Main point: find the cells (either coarse or fine) which belong to the support of a certain basis function.



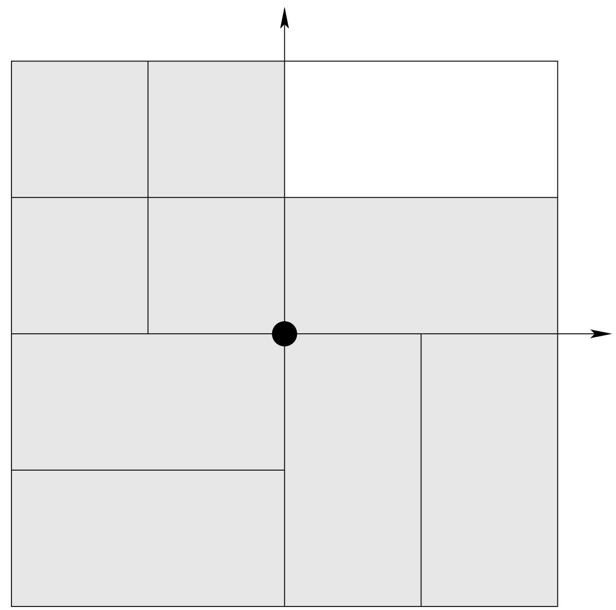
Anisotropic and Conforming

Main point: find the cells (either coarse or fine) which belong to the support of a certain basis function.



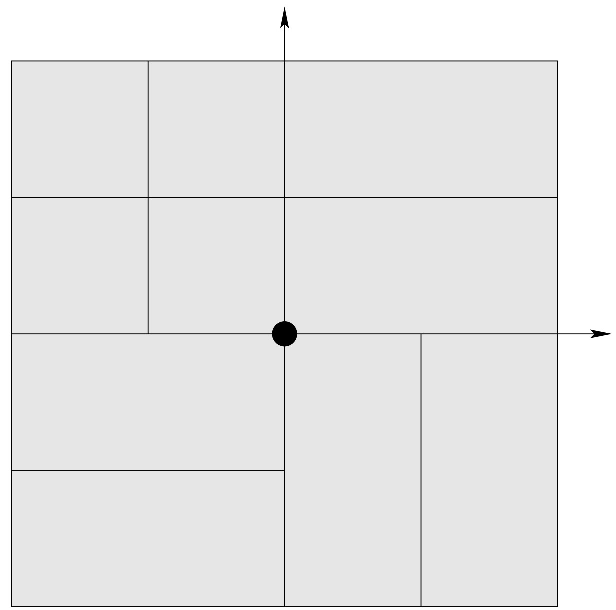
Anisotropic and Conforming

Main point: find the cells (either coarse or fine) which belong to the support of a certain basis function.



Anisotropic and Conforming

Main point: find the cells (either coarse or fine) which belong to the support of a certain basis function.

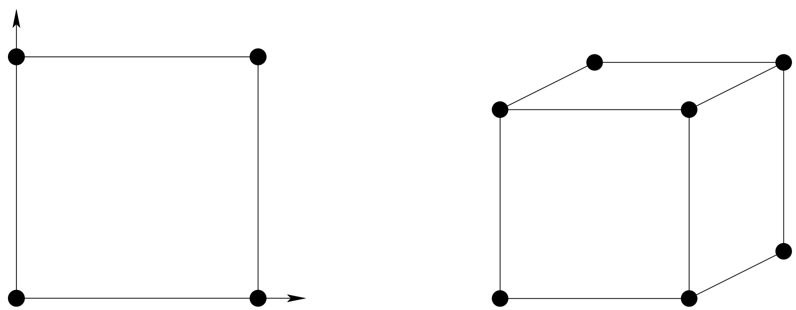


Condition for Continuity

- In order to have continuous global basis functions Φ_i, the unisolvent sets on the interfaces in the support of Φ_i have to match.

Condition for Continuity

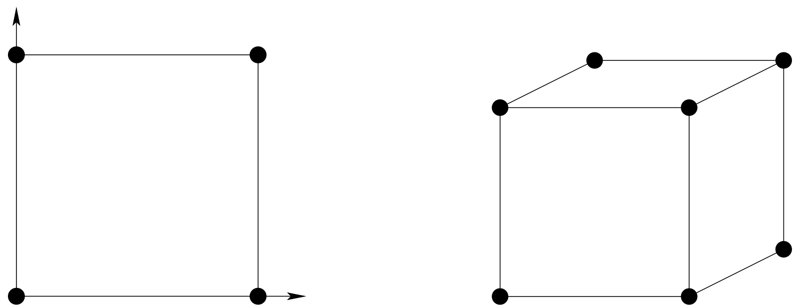
- In order to have continuous global basis functions Φ_i, the unisolvent sets on the interfaces in the support of Φ_i have to match.
- Unisolvent set for $\mathcal{Q}_1$ in a quad / hex are the corners:



$\Rightarrow$ matching edges which coincide in a vertex is sufficient

Condition for Continuity

- In order to have continuous global basis functions Φ_i, the unisolvent sets on the interfaces in the support of Φ_i have to match.
- Unisolvent set for $\mathcal{Q}_1$ in a quad / hex are the corners:



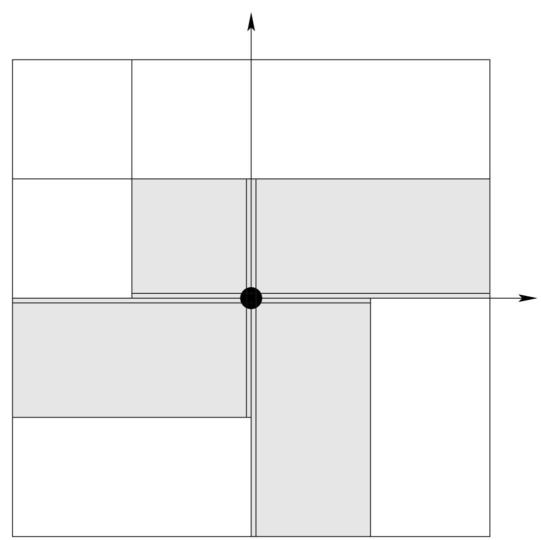
$\Rightarrow$ matching edges which coincide in a vertex is sufficient

- Unisolvent set for $\mathcal{Q}_p$ are
  - additional $p-1$ points on every edge
  - additional $(p-1)^2$ points on every face
  - additional $(p-1)^3$ points in the interior

$\Rightarrow$ matching faces which coincide in an edge is sufficient for continuous edge modes

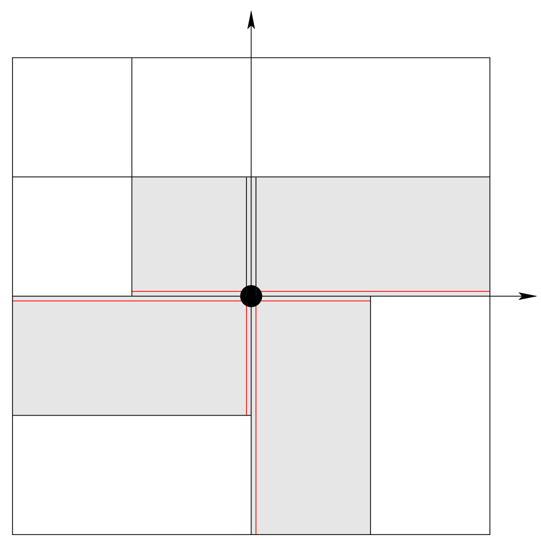
Algorithm for Continuity (Vertex)

- In every cell of the finest mesh, register all edges and cells in their vertices.



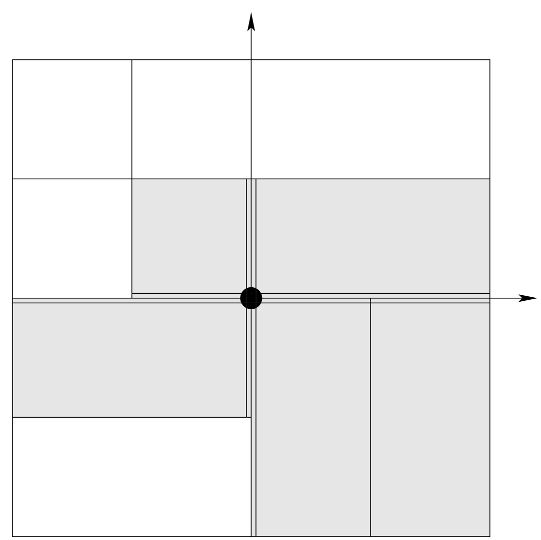
Algorithm for Continuity (Vertex)

- In every cell of the finest mesh, register all edges and cells in their vertices.
- For every vertex, while something changed in the last loop:
  - Check if some of the edges of the vertex have a relationship (ancestor / descendant).

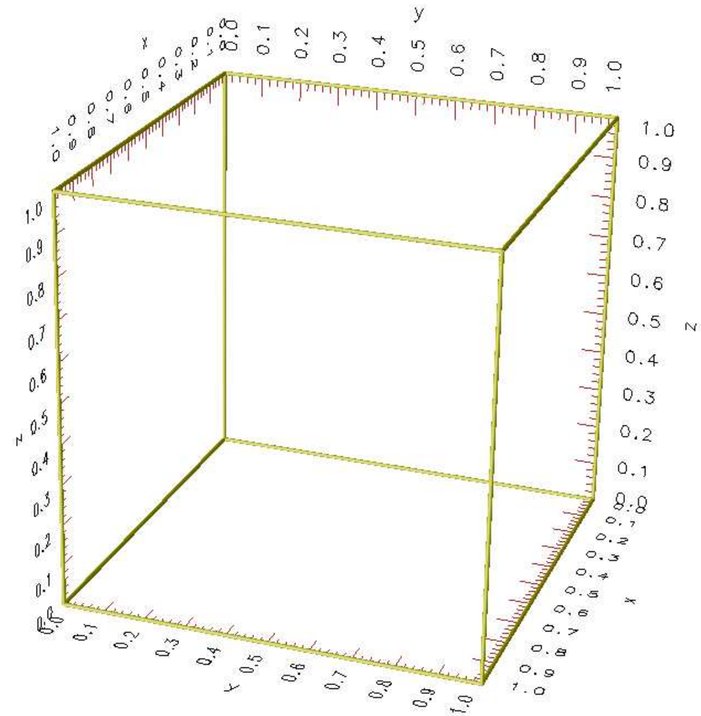


Algorithm for Continuity (Vertex)

- In every cell of the finest mesh, register all edges and cells in their vertices.
- For every vertex, while something changed in the last loop:
  - Check if some of the edges of the vertex have a relationship (ancestor / descendant).
  - If two edges are related, exchange the smaller cell in the list of the vertex by the cell matching the larger cell.
  - Delete the list of edges and rebuild it from the list of cells.

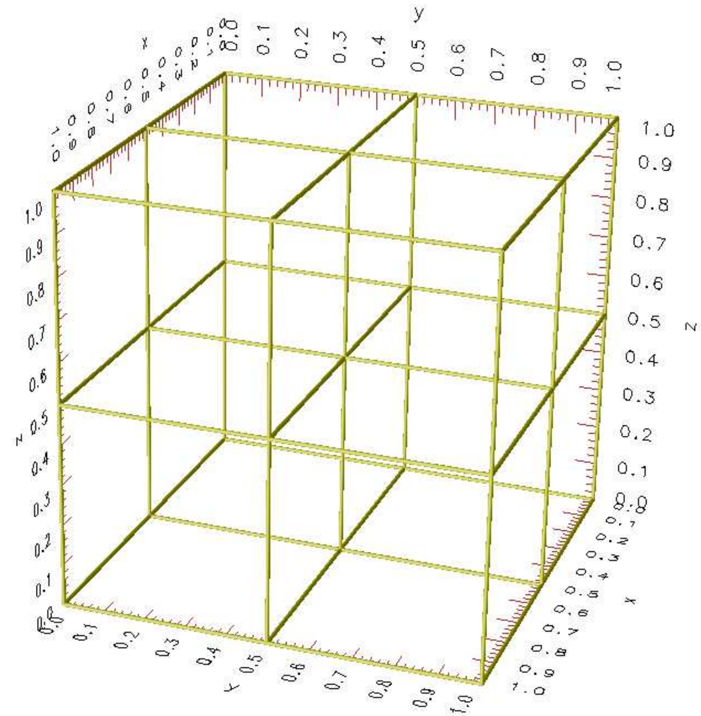


Sample Basis Functions

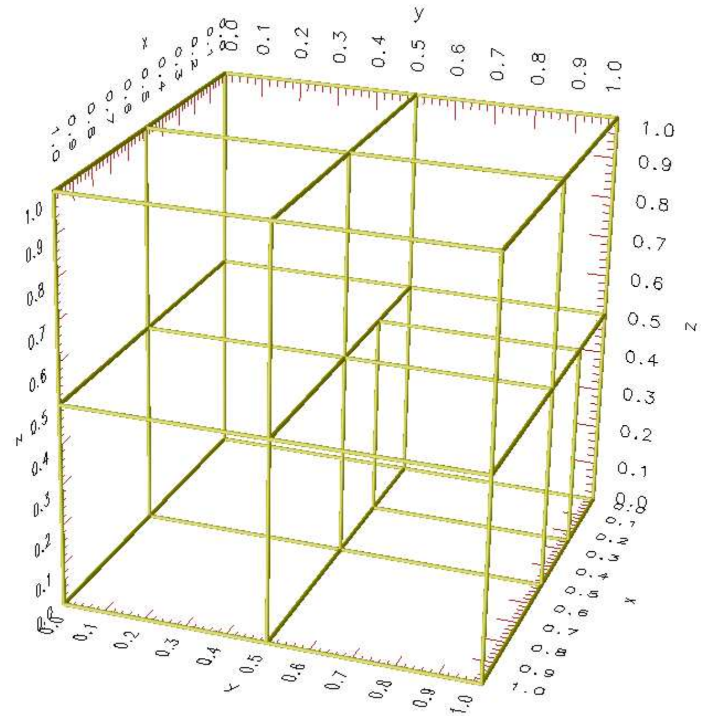


Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

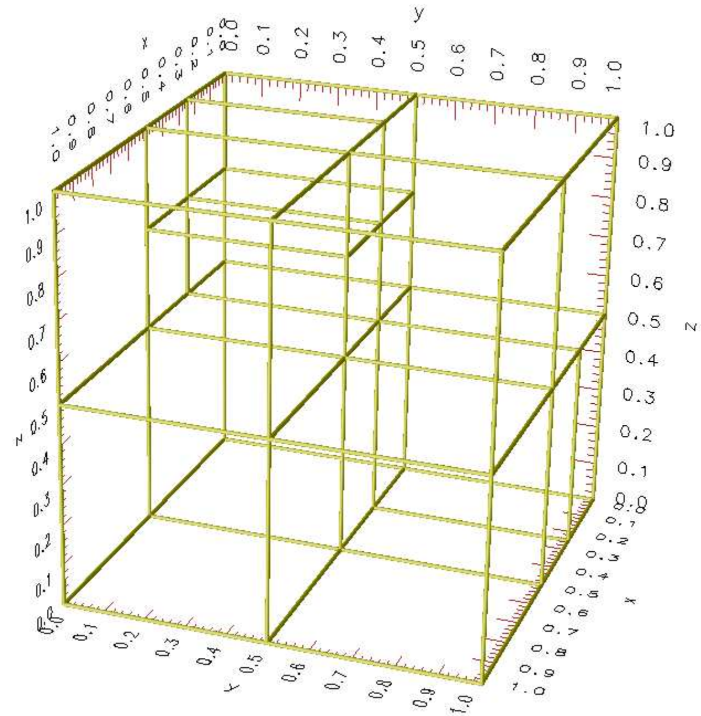
Sample Basis Functions



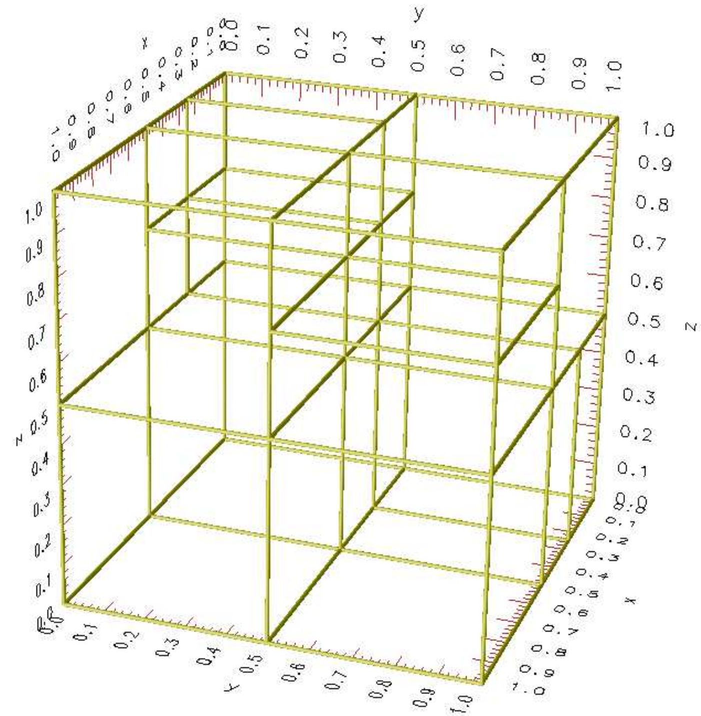
Sample Basis Functions

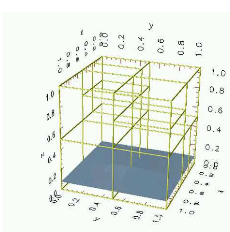


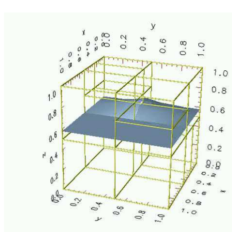
Sample Basis Functions

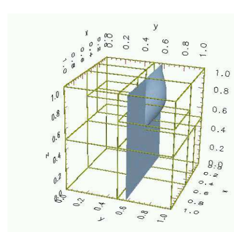


Sample Basis Functions

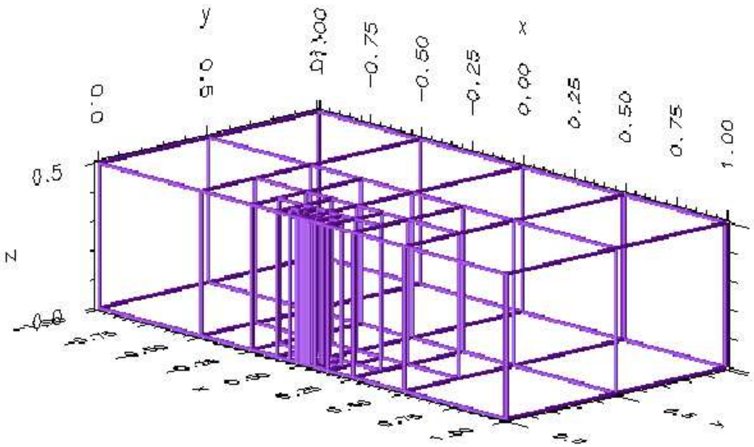


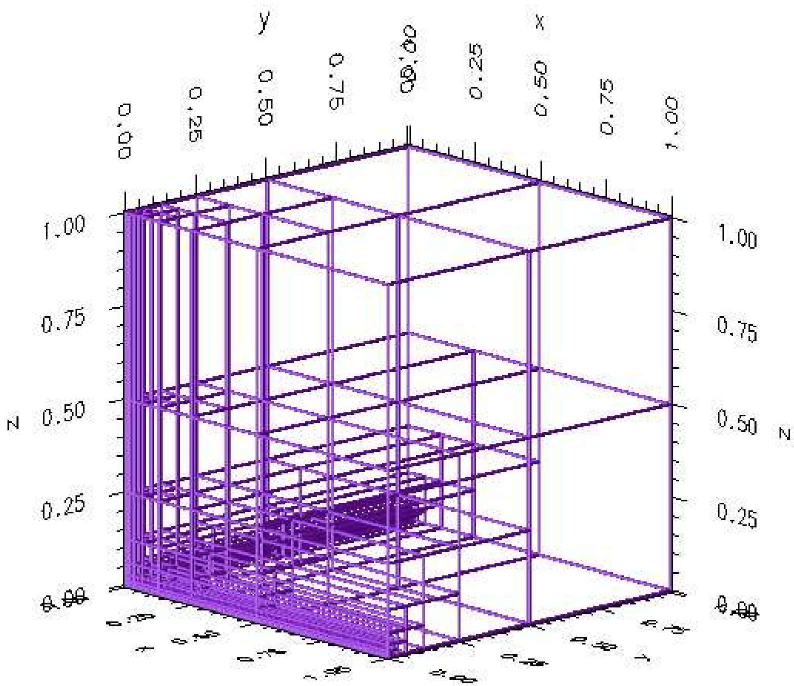


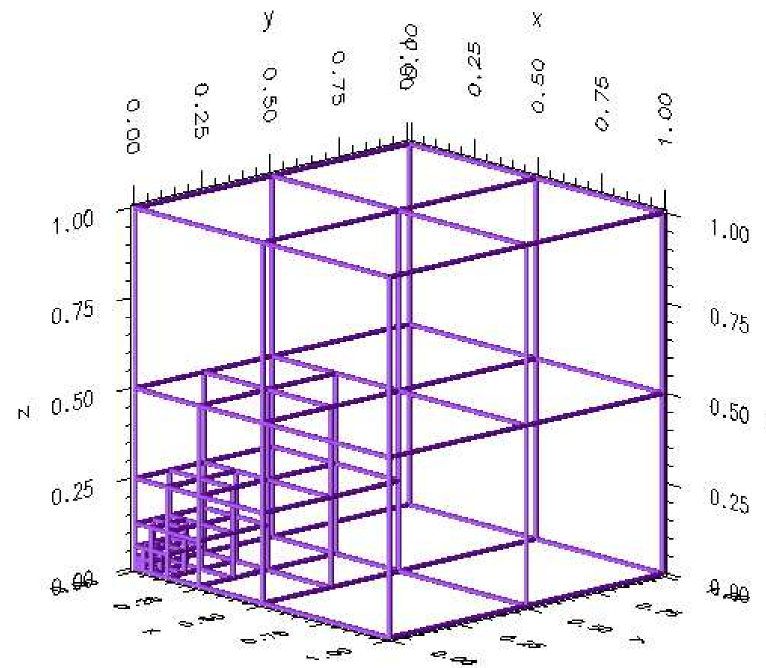




Sample hp ***Meshes***

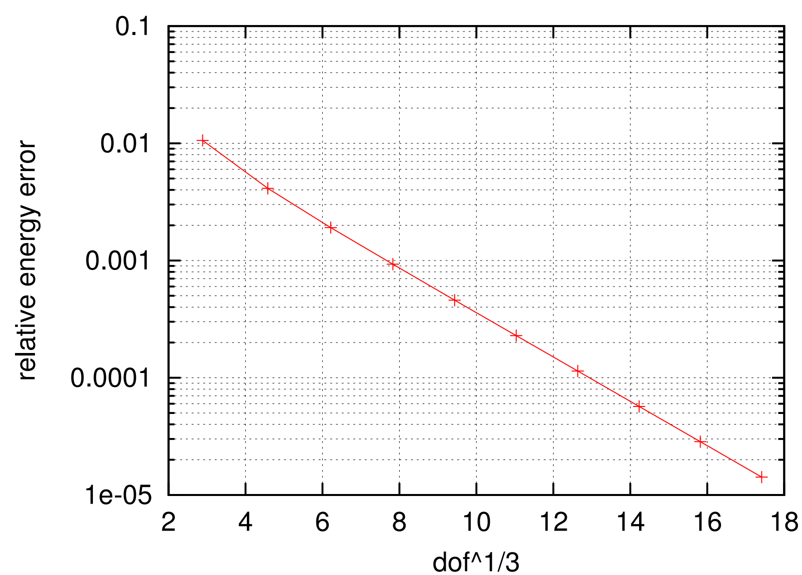


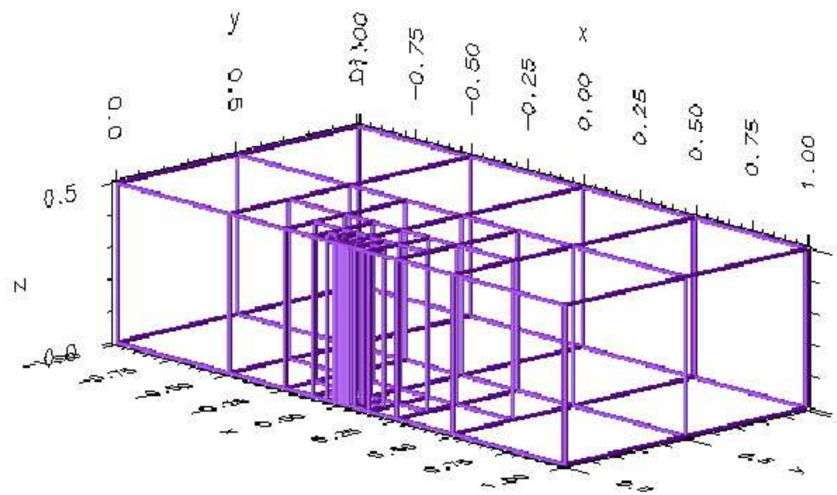




Scalar Computations: Pseudo 3D

Edge type singularity.





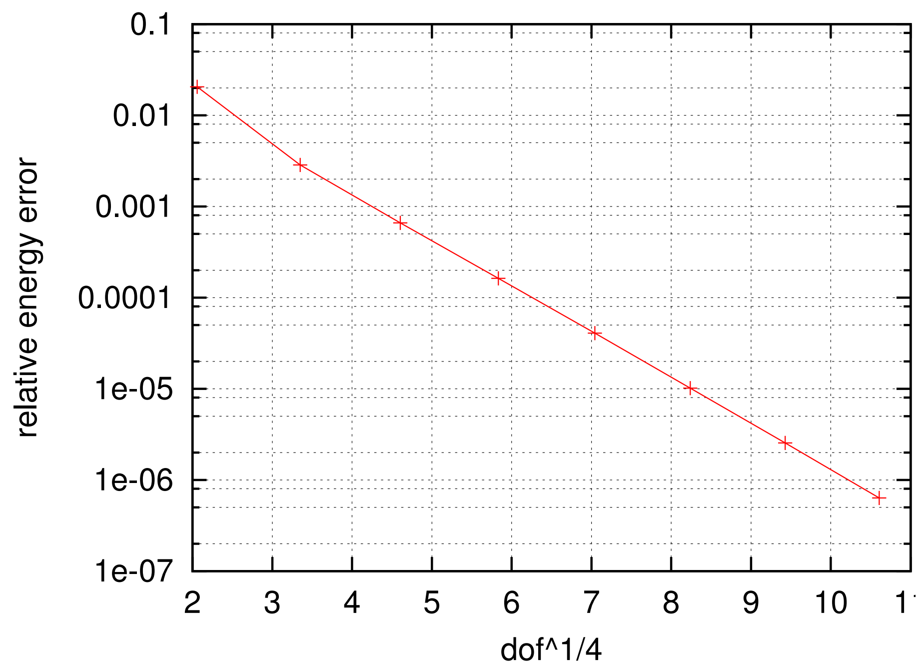
$$-\Delta u + u = f \text{ in } \Omega = (-1,1) \times (0,1) \times (0, 1/2)$$

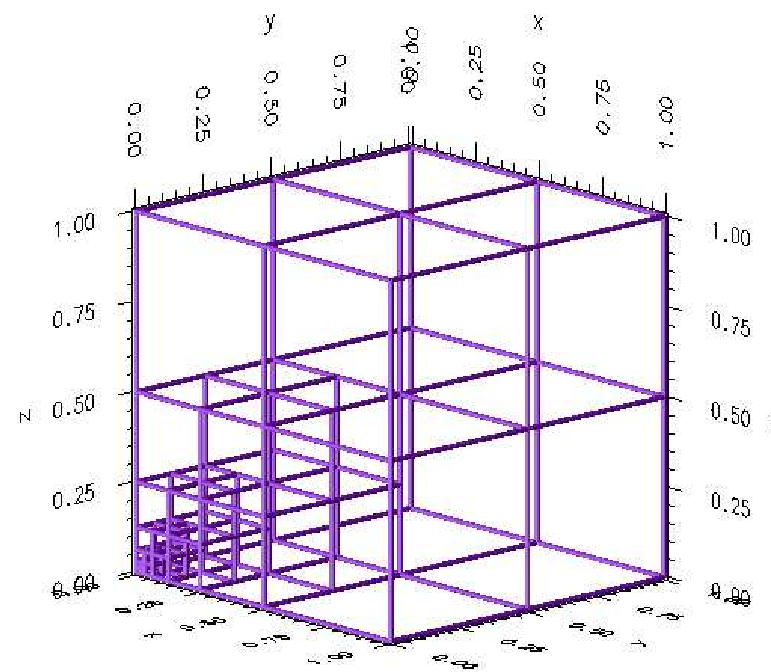
$$u(r,\phi,z) = \sqrt{r}\sin(\phi/2)z(1-z) \quad \text{in } \Omega$$

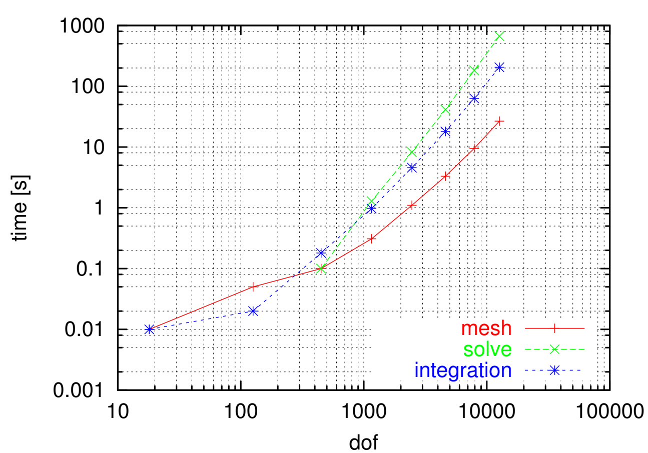
$$u = 0 \quad \text{on } \{z=0\} \subset \partial\Omega$$

$$\text{and on } \{y=0\} \cap \{x \geq 0\} \subset \partial\Omega$$

Scalar Computations: Vertex Singularity







Vertex type singularity.

$$-\Delta u + u = f \text{ in } \Omega = (0,1)^3$$

$$u(r,\theta,\phi) = \sqrt{r}\sin\theta\sin\phi \qquad \text{in } \Omega$$

$$u = 0 \qquad \text{on } \{y=0\} \subset \partial\Omega$$

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Overview

- Introduction: FEM & Exponential Convergence
- Assembling
- Handling Hanging Nodes
- Finding Regular Supports
- Maxwell Eigenvalue Problems
- Perspectives

Eigenvalue Problems

Source problem:

Find u with $-\Delta u + cu = f$ in Ω and $u = 0$ on $\partial\Omega$.

A variational form ($\cdot v$, $\int$, integration by parts):

Find $u \in V$ such that

$$\int_\Omega \nabla u \cdot \nabla v \, dx + c \int_\Omega uv \, dx = \int_\Omega fv \, dx \qquad \forall v \in V$$

$$\Rightarrow \quad (\boldsymbol{A} + c\boldsymbol{M})\underline{u}_N = \underline{l}_N$$ solve for $\underline{u}_N \in V_N$.

Eigenvalue Problems

Source problem:
Find u with $-\Delta u + cu = f$ in Ω and $u = 0$ on $\partial\Omega$.
A variational form ($\cdot v$, $\int$, integration by parts):
Find $u \in V$ such that

$$\int_\Omega \nabla u \cdot \nabla v \, dx + c \int_\Omega uv \, dx = \int_\Omega fv \, dx \qquad \forall v \in V$$

$$\Rightarrow \quad (\boldsymbol{A} + c\boldsymbol{M})\underline{u}_N = \underline{l}_N \qquad \text{solve for } \underline{u}_N \in V_N.$$

Eigenvalue Problem:
Find an Eigenpair (λ, u) with $-\Delta u = \lambda u$ in Ω and $u = 0$ on $\partial\Omega$.
Find $(\lambda, u) \in \mathbb{R} \times V$ such that

$$\int_\Omega \nabla u \cdot \nabla v \, dx = \lambda \int_\Omega uv \, dx \qquad \forall v \in V$$

$$\Rightarrow \quad \boldsymbol{A}\underline{u}_N = \lambda_N \boldsymbol{M}\underline{u}_N \qquad \text{solve for } (\lambda_N, \underline{u}_N) \in \mathbb{R} \times V_N.$$

Convergence of Eigenvalues

Eigenvectors:

$$\|w_m - w_{m,N}\|_{H^1} \leq C \sup_{v \in W_m} \inf_{v_N \in V_N} \|v - v_N\|_{H^1}$$

Simple Eigenvalues:

$$|\lambda_m - \lambda_{m,N}| \leq C \sup_{v \in W_m} \inf_{v_N \in V_N} \|v - v_N\|_{H^1}^2$$

Eigenvalues converge twice as fast as Eigenvectors.
Eigenvectors converge quasi-optimally.

For $\|v - v_N\|_{H^1}$, exponential convergence is possible.

Maxwell Equations

$$-\partial_t \underline{D} + \operatorname{curl} \underline{H} = \sigma \underline{E} + \underline{j} \qquad \text{Ampère's Law} \qquad (2)$$

$$\partial_t \underline{B} + \operatorname{curl} \underline{E} = 0 \qquad \text{Farraday's Law} \qquad (3)$$

$\underline{j}$: current density, $\underline{H}, \underline{E}$: magnetic & electric field, $\underline{B}, \underline{D}$: magnetic & electric induction.
Constitutive laws (in general: $\boldsymbol{\varepsilon}, \boldsymbol{\mu} \in \mathbb{R}^{3\times 3}$ later assumed to be scalars):
$\underline{D} = \boldsymbol{\varepsilon}\underline{E}$ and $\underline{B} = \boldsymbol{\mu}\underline{H}$ applied to (2) & (3):

$$-\partial_t \varepsilon \underline{E} + \operatorname{curl} \underline{H} = \sigma \underline{E} + \underline{j} \qquad \partial_t \mu \underline{H} + \operatorname{curl} \underline{E} = 0$$

Consider time harmonic solutions, Ansatz:

$$\underline{E}(\underline{x}, t) = \operatorname{Re}\big(\underline{E}(\underline{x}) e^{i\omega t}\big) \qquad \underline{H}(\underline{x}, t) = \operatorname{Re}\big(\underline{H}(\underline{x}) e^{i\omega t}\big)$$

ie., combinations of $\sin$ and $\cos$.

Towards a Variational Form

Time harmonic Maxwell equations:

$$-i\omega\varepsilon\underline{E} + \operatorname{curl}\underline{H} = \sigma\underline{E} + \underline{j} \qquad i\omega\mu\underline{H} + \operatorname{curl}\underline{E} = 0$$

$$\operatorname{curl}(\varepsilon^{-1}\operatorname{curl}\underline{H}) - \omega^2\tilde{\mu}\underline{H} = \operatorname{curl}(\varepsilon^{-1}\underline{j}) \qquad \operatorname{curl}(\mu^{-1}\operatorname{curl}\underline{E}) - \omega^2\tilde{\varepsilon}\underline{E} = -i\omega\underline{j}$$

with perfect conductor boundary conditions ($\sigma \to \infty$):

$$\mu\underline{H}\cdot\underline{n} = 0 \qquad \underline{E}\wedge\underline{n} = 0$$

Spaces for these equations:

$$H(\operatorname{div},\Omega) := \left\{\underline{u} \in L^2(\Omega)^3 : \operatorname{div}\underline{u} \in L^2(\Omega)\right\}$$

$$H(\operatorname{curl},\Omega) := \left\{\underline{u} \in L^2(\Omega)^3 : \operatorname{curl}\underline{u} \in L^2(\Omega)\right\}$$

Variational Electric Source Problem

Consider

$$\mathrm{curl}(\mu^{-1}\,\mathrm{curl}\,\underline{E}) - \omega^2\tilde{\varepsilon}\underline{E} = -i\omega\underline{j} \text{ in } \Omega \text{ and } \underline{E}\wedge\underline{n} = 0 \text{ on } \partial\Omega.$$

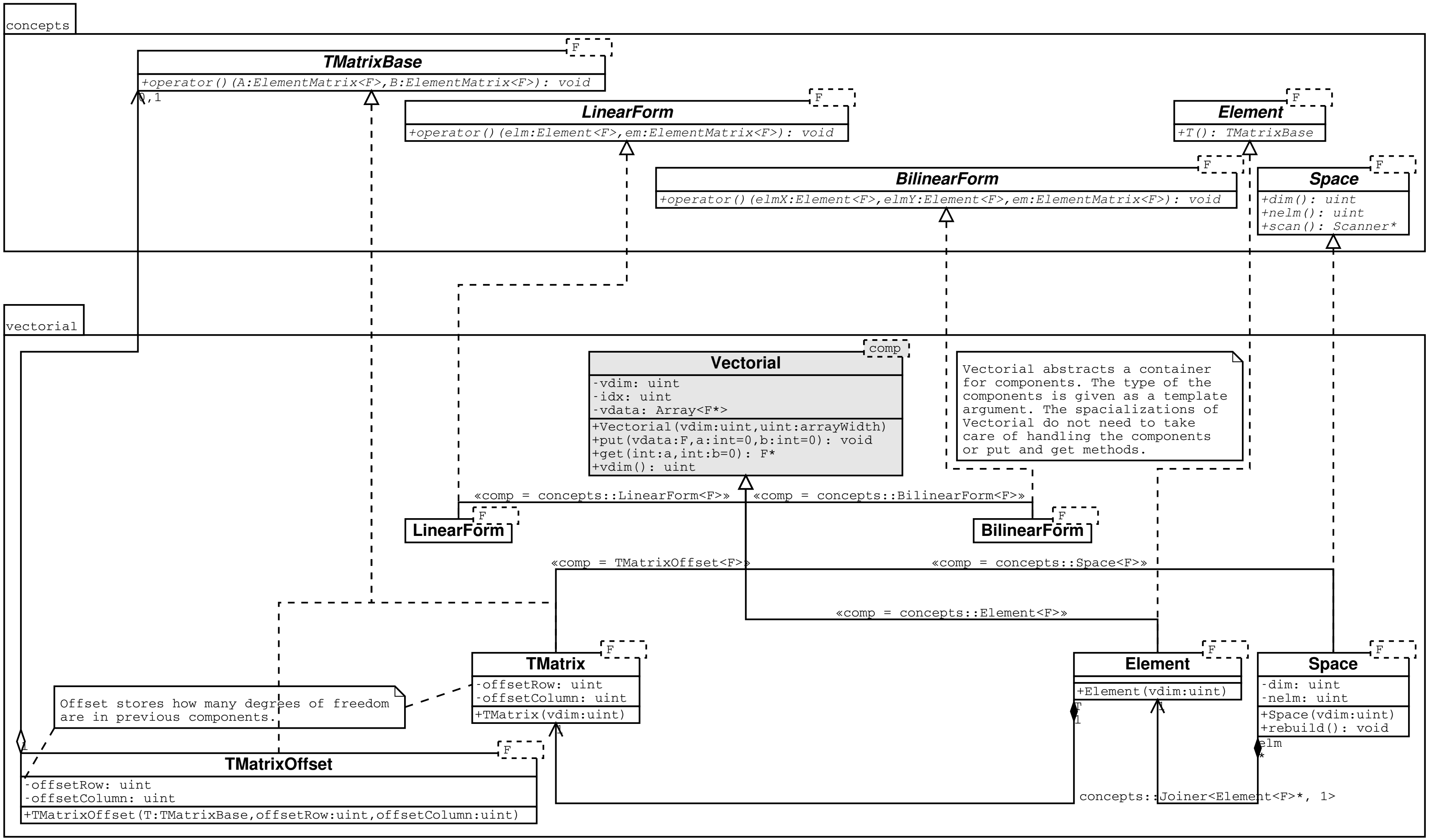
in variational form:
Find $\underline{E} \in H_0(\mathrm{curl},\Omega)$ with $\mathrm{div}\,\varepsilon\underline{E} = 0$ such that

$$\int_\Omega \mu^{-1}\,\mathrm{curl}\,\underline{E}\cdot\mathrm{curl}\,\underline{F} - \omega^2\int_\Omega \varepsilon\underline{E}\cdot\underline{F} = \int_\Omega i\omega\underline{j}\cdot\underline{F} \quad \forall\underline{F}\in H_0(\mathrm{curl},\Omega)$$

Constraint $\mathrm{div}\,\varepsilon\underline{E} = 0$ makes discretisation difficult (Nédelec elements).
Better introduce $X_n := \{\underline{E}\in H_0(\mathrm{curl},\Omega) : \mathrm{div}\,\underline{E}\in L^2(\Omega)\}$ and the variational form
Find $\underline{E}\in X_n$ such that

$$\int_\Omega \mu^{-1}\,\mathrm{curl}\,\underline{E}\cdot\mathrm{curl}\,\underline{F} + \int_\Omega \mathrm{div}\,\underline{E}\,\mathrm{div}\,\underline{F} - \omega^2\int_\Omega \varepsilon\underline{E}\cdot\underline{F} = \int_\Omega i\omega\underline{j}\cdot\underline{F} \quad \forall\underline{F}\in X_n$$

Classes for Vector Valued Problems



Electric Eigenvalue Problem

Find $\omega > 0$ sucht that $\exists \underline{E} \in X_n \setminus \{0\}$ with

$$\int_\Omega \mu^{-1} \operatorname{curl} \underline{E} \cdot \operatorname{curl} \underline{F} + \int_\Omega \operatorname{div} \underline{E} \operatorname{div} \underline{F} = \omega^2 \int_\Omega \varepsilon \underline{E} \cdot \underline{F} \quad \forall \underline{F} \in X_n$$

$H_n := \{\underline{u} \in H^1(\Omega)^3 : \underline{u} \wedge \underline{n} = 0 \text{ on } \partial\Omega\}$

- X_n is curl and div conforming, hence continuous across interfaces $\Rightarrow H_n = X_n$
- H_n is easy to discretise and implement: Cartesian product of scalar discretisation $S^{1,\underline{p}}(\Omega, \mathcal{T})$ of $H^1(\Omega)$
- Converges to wrong solutions if Ω has reentrant corners:
  - $H_n \neq X_n$
  - $\operatorname{codim}_{X_n} H_n = \infty$
  - H_n closed in X_n ie., sequences in H_n have their limits in H_n.

Weighted Regularization

Find the frequencies $\omega > 0$ such that $\exists \underline{E} \in H_n \setminus \{0\}$ with

$$\int_\Omega \operatorname{curl} \underline{E} \cdot \operatorname{curl} \underline{F} + s\langle \underline{E}, \underline{F} \rangle_Y = \omega^2 \int_\Omega \underline{E} \cdot \underline{F} \qquad \forall \underline{F} \in H_n$$

$$\langle \underline{E}, \underline{F} \rangle_Y = \int_\Omega \rho(\underline{x}) \operatorname{div} \underline{E} \operatorname{div} \underline{F}$$

Properly chosen weight $\rho(\underline{x})$ and $s \in \mathbb{R}_+$.

Weighted Regularization

Find the frequencies $\omega > 0$ such that $\exists \underline{E} \in H_n \setminus \{0\}$ with

$$\int_\Omega \operatorname{curl} \underline{E} \cdot \operatorname{curl} \underline{F} + s\langle \underline{E}, \underline{F} \rangle_Y = \omega^2 \int_\Omega \underline{E} \cdot \underline{F} \qquad \forall \underline{F} \in H_n$$

$$\langle \underline{E}, \underline{F} \rangle_Y = \int_\Omega \rho(\underline{x}) \operatorname{div} \underline{E} \operatorname{div} \underline{F}$$

Properly chosen weight $\rho(\underline{x})$ and $s \in \mathbb{R}_+$.

Idea: use spaces

$$X_n[Y] := \{\underline{u} \in H_0(\operatorname{curl}, \Omega) : \operatorname{div} \underline{u} \in Y\} \supset H_n \text{ dense}$$

and the solutions of Maxwell equations $\in X_n[Y]$.

[2] Martin Costabel and Monique Dauge, “Weighted regularization of Maxwell equations in polyhedral domains”, *Numer. Math.* 93 (2), pp. 239–277 (2002).

Chosing the Weight and s

$$s\langle \underline{E}, \underline{F} \rangle_Y = s \int_\Omega \rho(\underline{x}) \operatorname{div} \underline{E} \operatorname{div} \underline{F}$$

2D:
$\rho(\underline{x}) = r^\alpha$ where r is the distance to a reentrant corner and $\alpha \in [0, 2]$ depending on the angle of the reentrant corner: $\alpha \in (2 - 2^\pi/\omega_c, 2]$

s scales the $\langle . , . \rangle_Y$ form. Spurious Eigenvalues get scaled too, real Eigenvalues not. Sensible range: $(0, 30)$. $s = 0$ gives a large kernel since $\operatorname{div} \underline{E} = 0$ is not enforced at all.
$\alpha = 2$ is the limiting case, nice to implement since r^2 is polynomial.

Chosing the Weight and s

$$s\langle \underline{E}, \underline{F}\rangle_Y = s \int_\Omega \rho(\underline{x}) \operatorname{div} \underline{E} \operatorname{div} \underline{F}$$

2D:
$\rho(\underline{x}) = r^\alpha$ where r is the distance to a reentrant corner and $\alpha \in [0, 2]$ depending on the angle of the reentrant corner: $\alpha \in (2 - 2^\pi/\omega_c, 2]$

3D:

$$\rho(\underline{x}) = \operatorname{dist}(\underline{x}, \mathcal{C} \cup \mathcal{E})^\alpha$$

where $\alpha \in [0, 2]$ (depending on angle of edge and cone of corner).

s scales the $\langle .\,, .\rangle_Y$ form. Spurious Eigenvalues get scaled too, real Eigenvalues not. Sensible range: $(0, 30)$. $s = 0$ gives a large kernel since $\operatorname{div} \underline{E} = 0$ is not enforced at all.
$\alpha = 2$ is the limiting case, nice to implement since r^2 is polynomial.

Convergence of Eigenvalues

As before:
Eigenvectors:

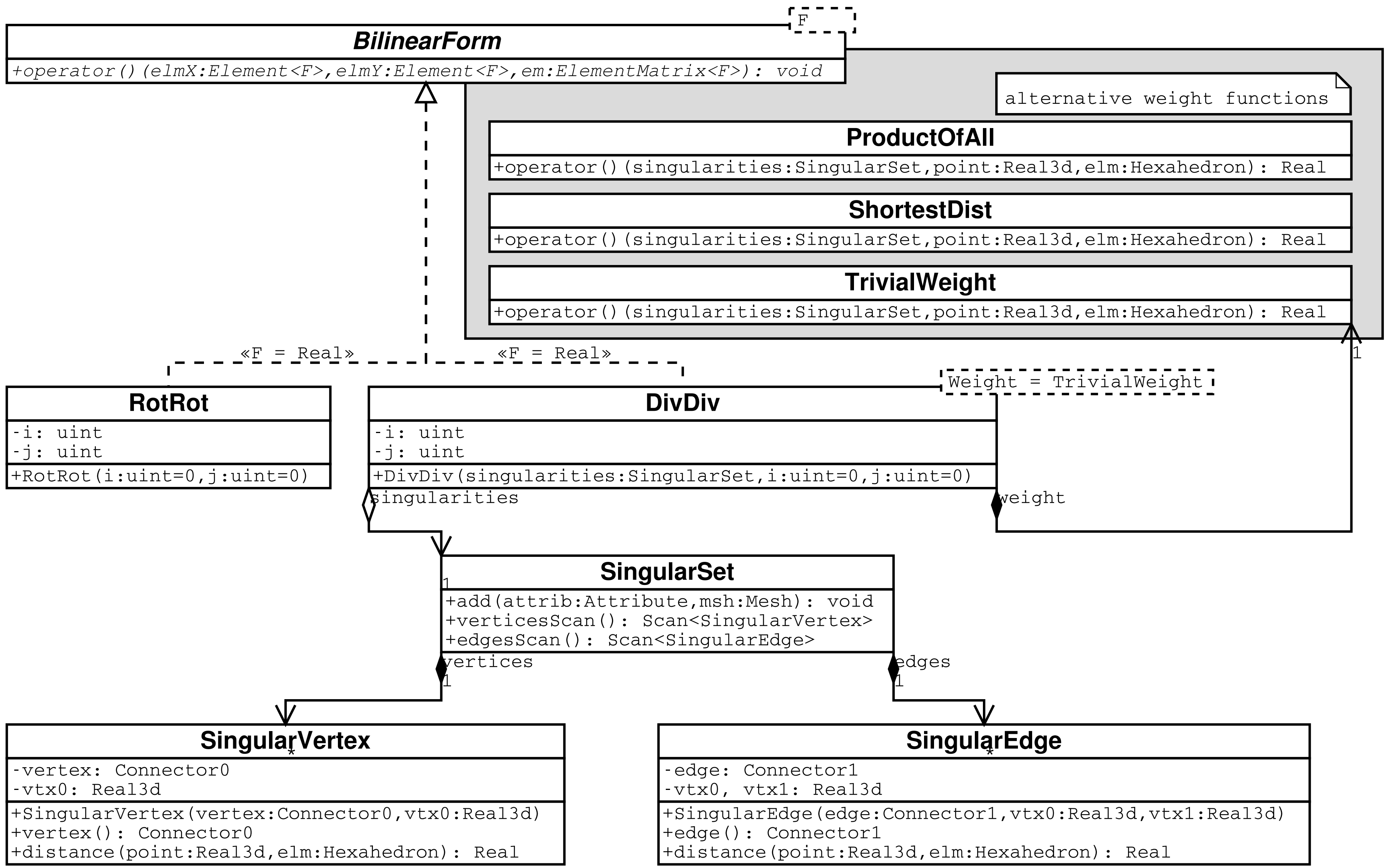
$$\|E_m - E_{m,N}\|_{X_n} \leq C \sup_{F \in W_m} \inf_{F_N \in V_N} \|F - F_N\|_{X_n}$$

Simple Eigenvalues:

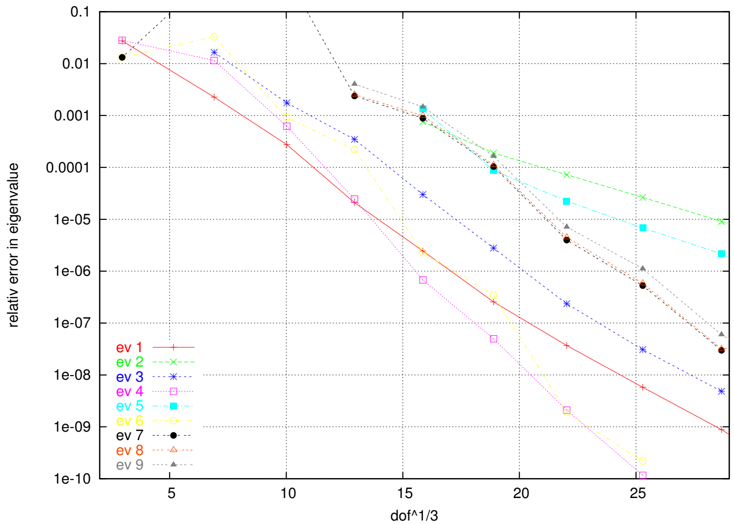
$$|\lambda_m - \lambda_{m,N}| \leq C \sup_{F \in W_m} \inf_{F_N \in V_N} \|F - F_N\|^2_{X_n}$$

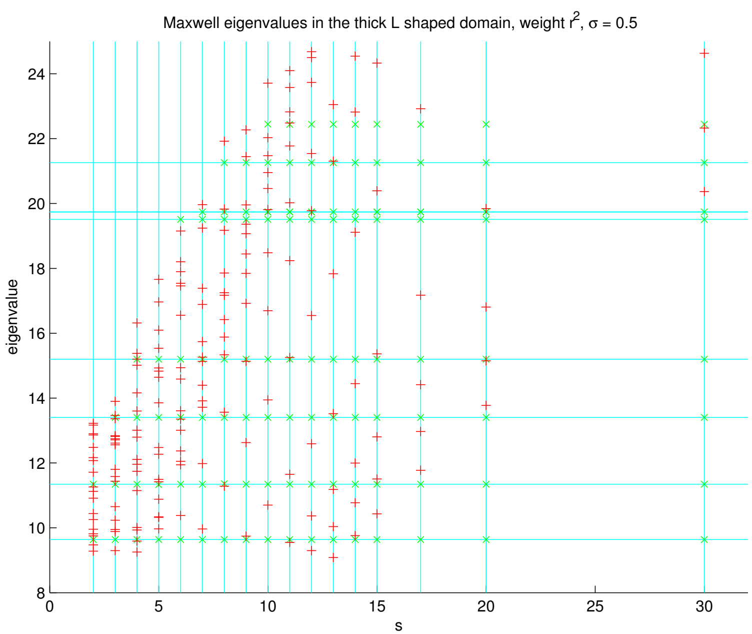
For $\|F - F_N\|_{X_n}$, exponential convergence possible:
$\mathbb{R}^2$: Proof by Costabel, Dauge, Schwab
$\mathbb{R}^3$: experimental evidence, proof in preparation

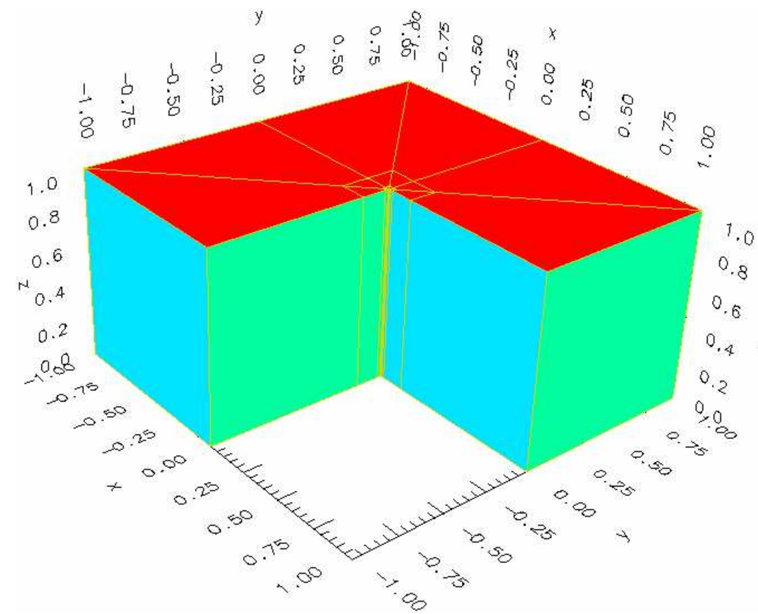
Classes for Maxwell Discretisation



EVP in the Thick L Shaped Domain



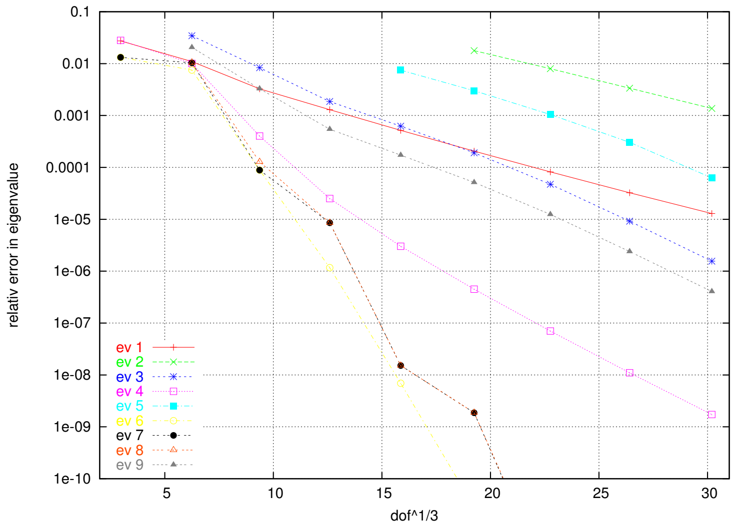


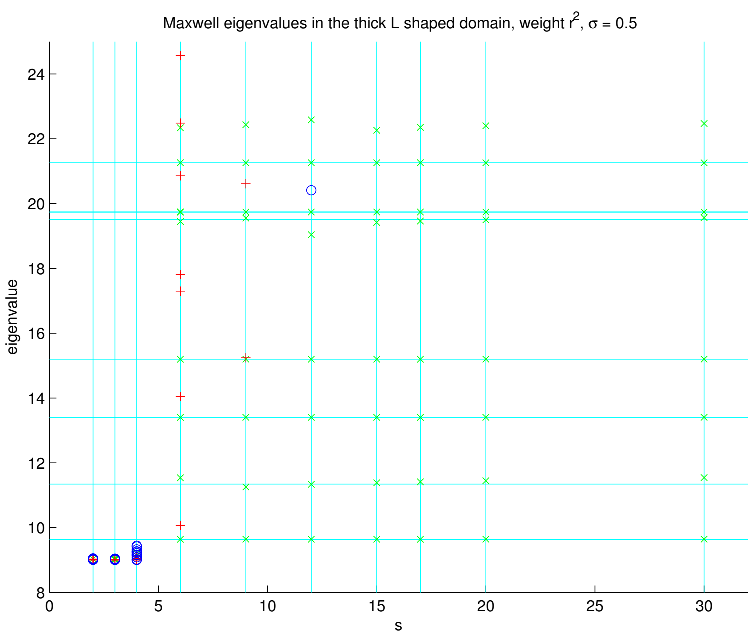


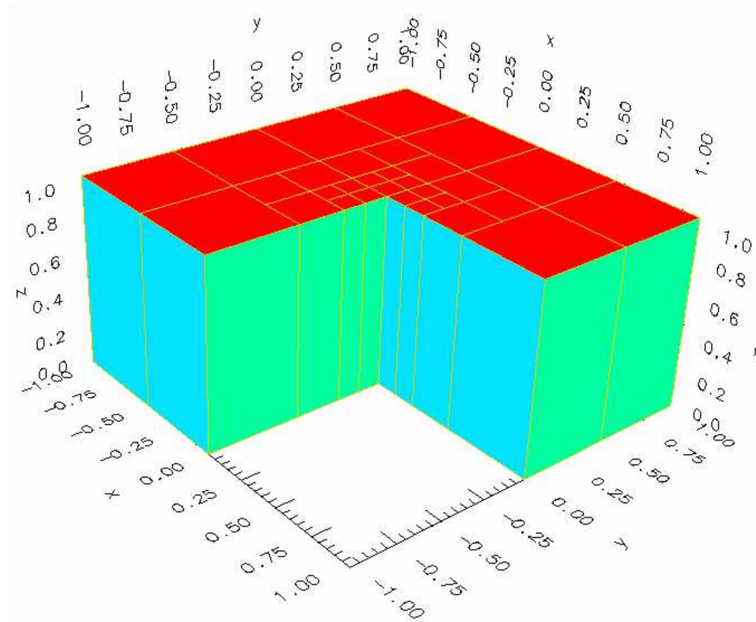
$$\sigma = 0.15$$

$$\alpha = 2$$

EVP in the Thick L Shaped Domain







$$\sigma = 0.5$$

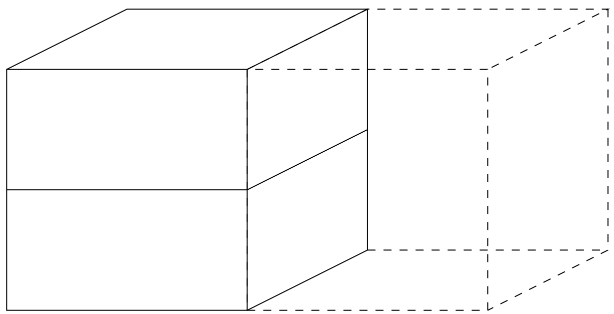
$$\alpha = 2$$

7054 dof

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Perspectives

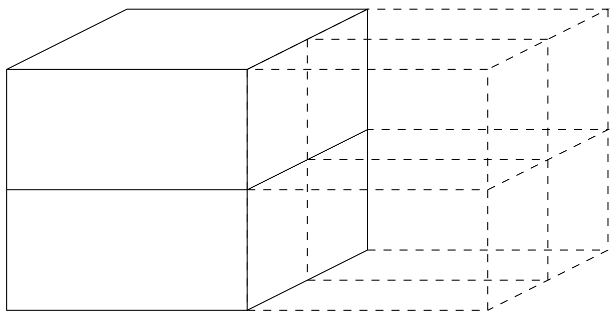
- Maxwell EVP in the Fichera corner
- Maxwell source problems
- A posteriori error estimation, anisotropic regularity estimation
- Improved mesh handling



- Iterative multilevel domain decompositioning solvers:
  Toselli (Zürich), Schöberl (Linz)

Perspectives

- Maxwell EVP in the Fichera corner
- Maxwell source problems
- A posteriori error estimation, anisotropic regularity estimation
- Improved mesh handling



- Iterative multilevel domain decompositioning solvers:
  Toselli (Zürich), Schöberl (Linz)