Numerical integration and high-order finite element methods applied to time-harmonic Maxwell equations

M. Duruflé

INRIA Rocquencourt

ONERA Palaiseau

Bibliography and motivation

- Y. Maday, E. Ronquist, Spectral Methods
- N. Tordjman, mass lumping for wave equation (triangles/quadrilaterals)
- Cohen, Monk, mass lumping for Maxwell's equations (hexahedra)
- S. Fauqueux, mixed spectral elements for wave and elastic equations (hexahedra)
- S. Pernet, Discontinuous Galerkin methods for Maxwell's equations (hexahedra)

Introduction

- Apply techniques of “mass lumping” and “mixed formulation”, which are efficient in temporal domain
  - Application of these techniques to Helmholtz and time-harmonic Maxwell equations
  - Gain in storage and time, by using these techniques in frequential domain

Introduction

- Apply techniques of “mass lumping” and “mixed formulation”, which are efficient in temporal domain
  - Application of these techniques to Helmholtz and time-harmonic Maxwell equations
  - Gain in storage and time, by using these techniques in frequential domain
- Choose an efficient preconditioning technique to solve linear systems issued from these equations

Introduction

- Apply techniques of “mass lumping” and “mixed formulation”, which are efficient in temporal domain
  - Application of these techniques to Helmholtz and time-harmonic Maxwell equations
  - Gain in storage and time, by using these techniques in frequential domain

- Choose an efficient preconditioning technique to solve linear systems issued from these equations

- Apply the developped algorithms to evaluate accurately radar cross sections of electromagnetic targets

Outline

- Resolution of Helmholtz equation
  - Advantage to use high-order methods
  - Fast matrix-vector product on hexahedral elements
  - Comparison between tetrahedral elements and hexahedral elements

Outline

- Resolution of Helmholtz equation
  - Advantage to use high-order methods
  - Fast matrix-vector product on hexahedral elements
  - Comparison between tetrahedral elements and hexahedral elements
- Resolution of time-harmonic Maxwell equations
  - Spurious modes for Nedelec's second family
  - Discontinuous Galerkin method
  - Fast matrix-vector product for Nedelec's first family

Outline

- Resolution of Helmholtz equation
  - Advantage to use high-order methods
  - Fast matrix-vector product on hexahedral elements
  - Comparison between tetrahedral elements and hexahedral elements
- Resolution of time-harmonic Maxwell equations
  - Spurious modes for Nedelec's second family
  - Discontinuous Galerkin method
  - Fast matrix-vector product for Nedelec's first family
- Maxwell equations in axisymmetric domains
  - Model equations
  - Advantages of high-order methods

Outline

Resolution of Helmholtz equation

Resolution of time-harmonic Maxwell equations

Maxwell equations in axisymmetric domains

Softwares and libraries used

Mesh generators : Modulef, Gmsh

Softwares and libraries used

Mesh generators : Modulef, Gmsh

Visualization tools : Matlab, Medit

Softwares and libraries used

Mesh generators : Modulef, Gmsh

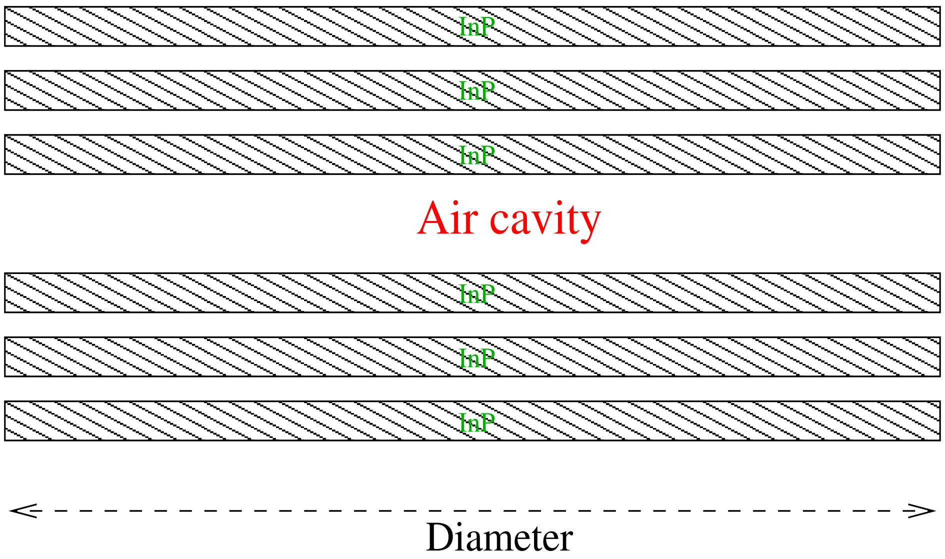
Visualization tools : Matlab, Medit

MUMPS : Fortran library to solve large sparse linear systems

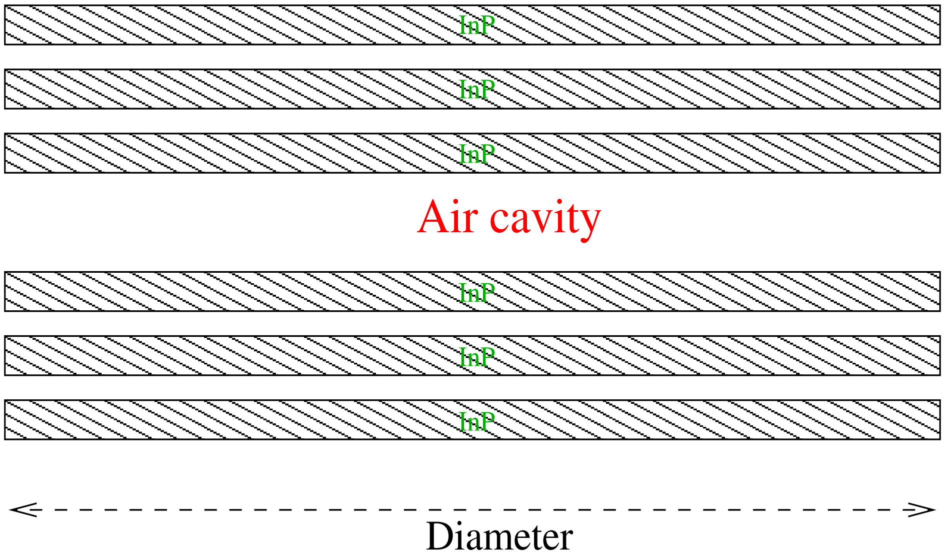
ARPACK : Fortran library to solve large sparse eigenvalue problems

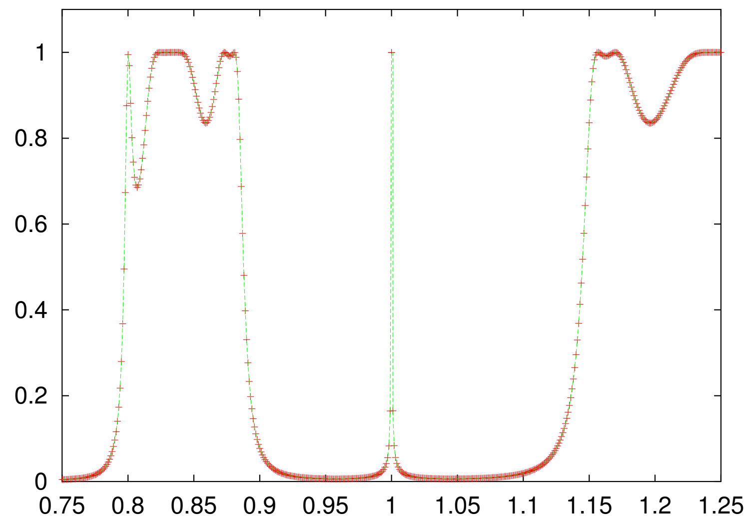
Seldon : C++ linear algebra library

A test case : an optical filter



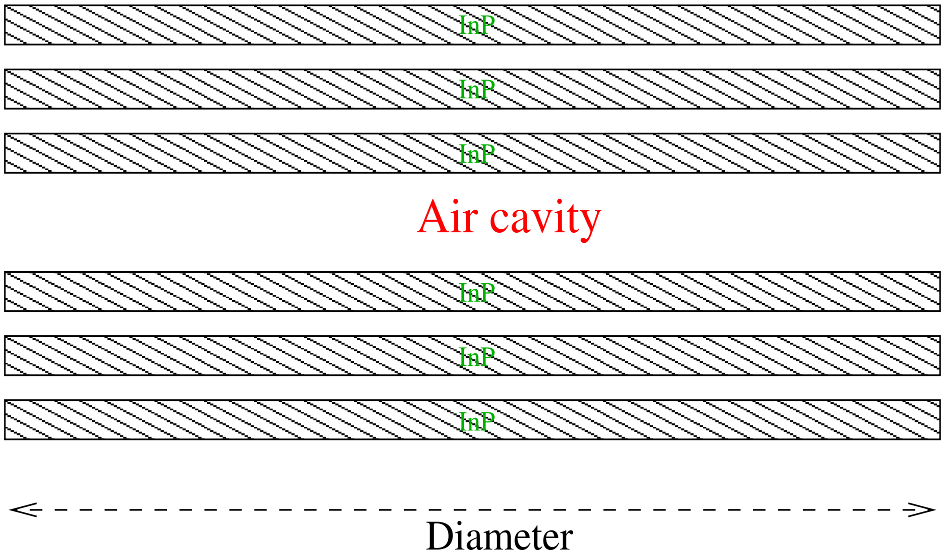
A test case : an optical filter

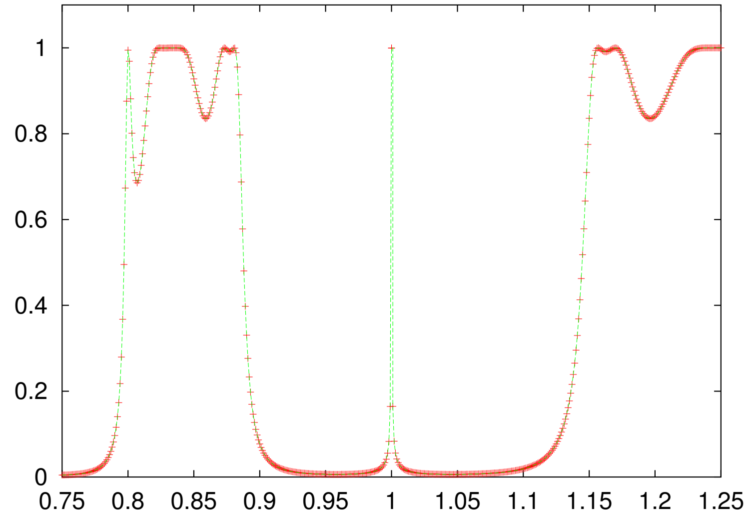




At right, transmission coefficient according to the frequency

A test case : an optical filter

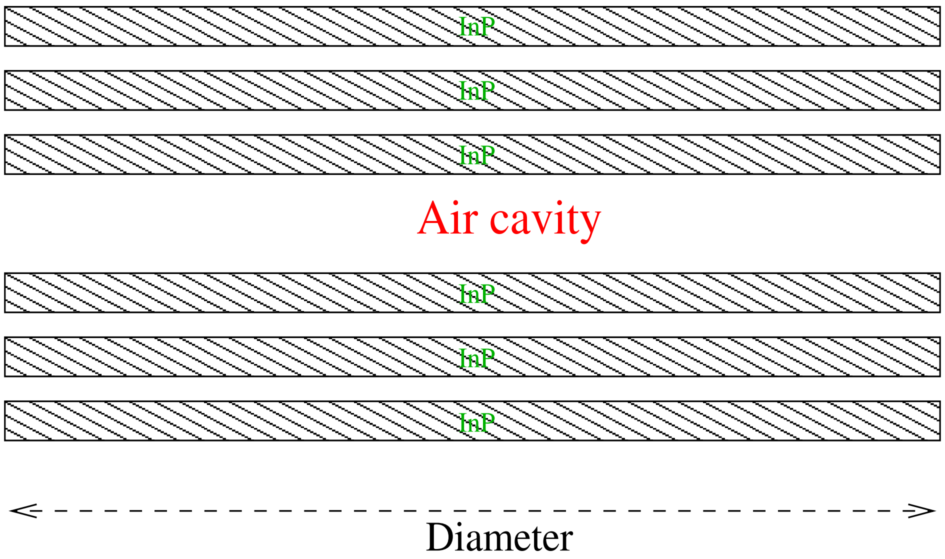


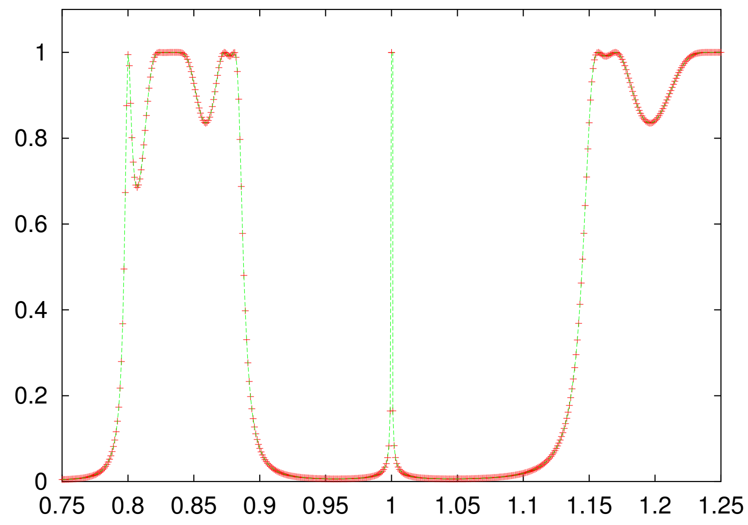


At right, transmission coefficient according to the frequency

- Frequency F = 1.0 is a resonant frequency of the device

A test case : an optical filter

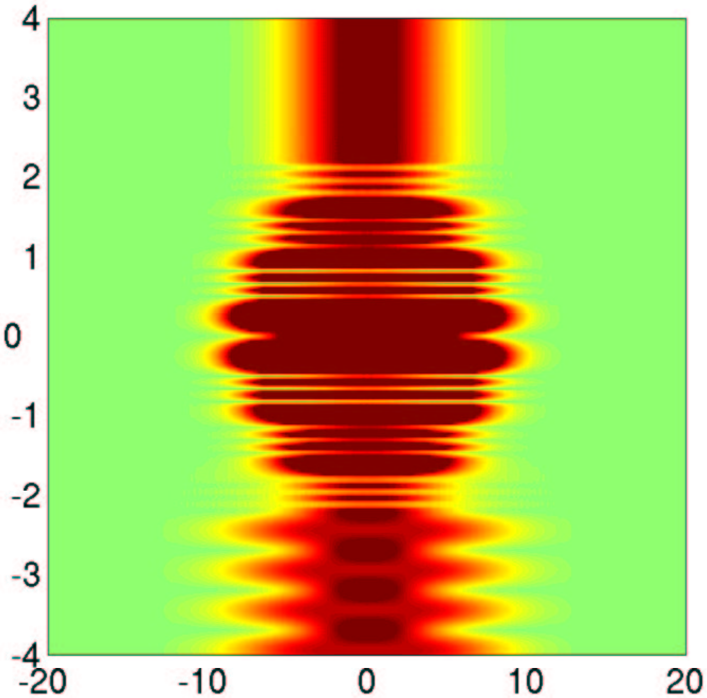




At right, transmission coefficient according to the frequency

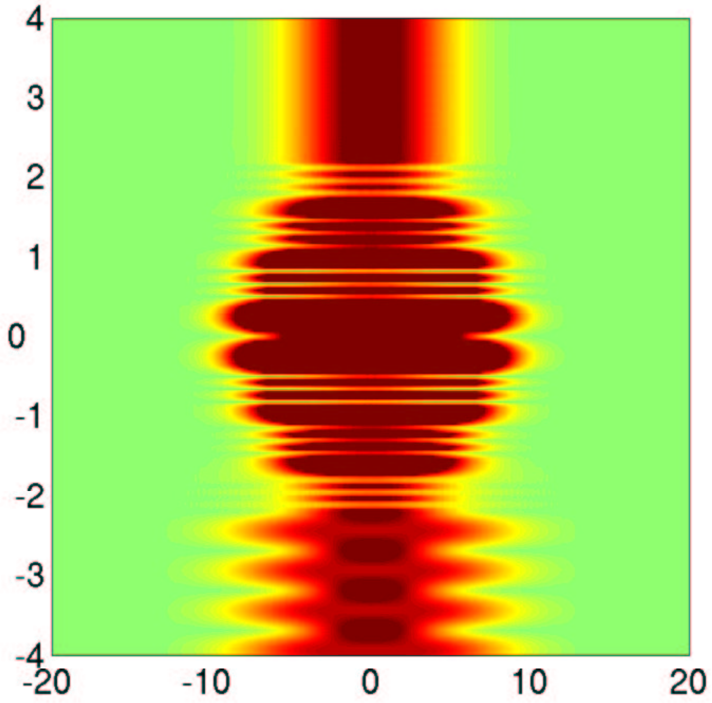
- Frequency F = 1.0 is a resonant frequency of the device
- Enlightment of the device by a gaussian beam.
- PML around the computational domain.

Advantage to use high order method



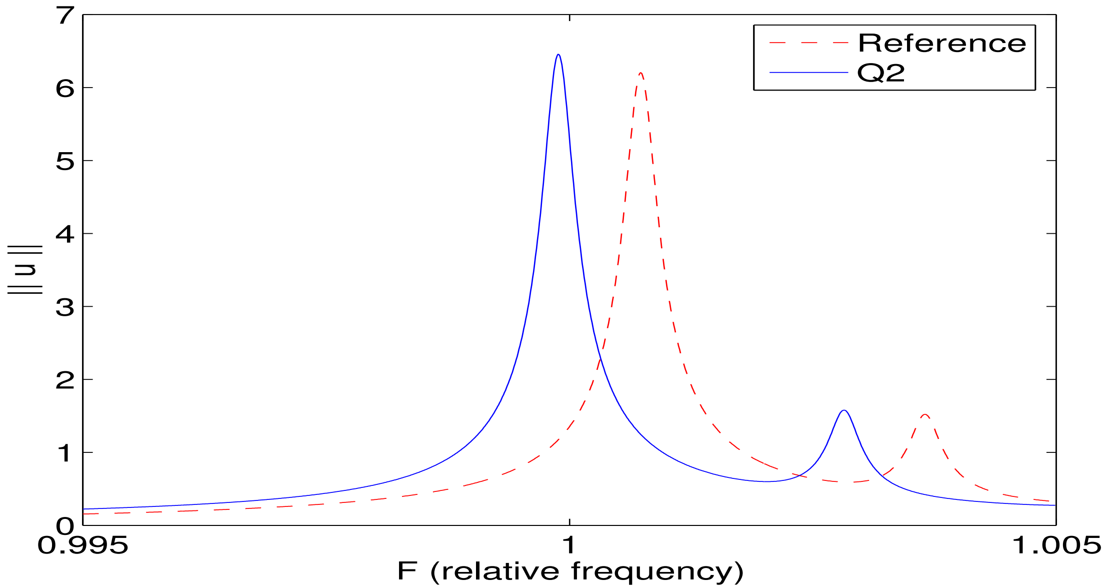
Numerical solution for $\mathbf{Q_5}$ with 10 points by wavelength

Advantage to use high order method



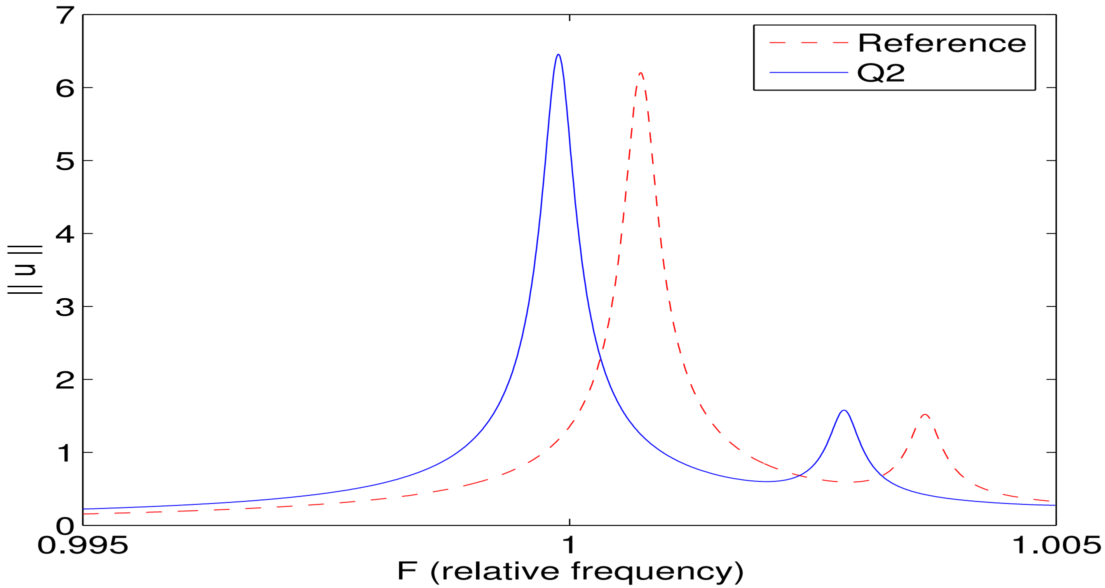
At right, numerical solution for Q_2 with 10 points by wave-length

Advantage to use high order method



Norm of the solution at the ouput, according to the frequency

Advantage to use high order method



Norm of the solution at the ouput, according to the frequency
Which order is optimal to reach an error less than 10% ?

Order	2	3	4	5	6	7
Nb dofs	453 000	69 800	52 000	33 200	47 700	42 200

Helmholtz equation

$$-\rho\,\omega^2\,u \;-\; \mathrm{div}(\mu\,\nabla u) \;=\; f \qquad \in \Omega$$

Helmholtz equation

$$-\rho\,\omega^2\,u - \mathrm{div}(\mu\,\nabla u) = f \quad \in \Omega$$

Use of finite element method leads to the following linear system :

$$(-\omega^2 D_h + K_h)\,U_h = F_h$$

Helmholtz equation

$$-\rho\,\omega^2\,u \;-\; \mathrm{div}(\mu\,\nabla u) \;=\; f \qquad \in \Omega$$

Use of finite element method leads to the following linear system :

$$(-\omega^2 D_h \;+\; K_h)\,U_h \;=\; F_h$$

Mass matrix $D_h \;=\; \displaystyle\int_\Omega \rho\,\varphi_i^{GL}\,\varphi_j^{GL}\,dx$

Stiffness matrix $K_h \;=\; \displaystyle\int_\Omega \mu\,\nabla\varphi_i^{GL}\cdot\nabla\varphi_j^{GL}\,dx$

Helmholtz equation

$$-\rho\,\omega^2\,u \;-\; \mathrm{div}(\mu\,\nabla u) \;=\; f \qquad \in \Omega$$

Use of finite element method leads to the following linear system :

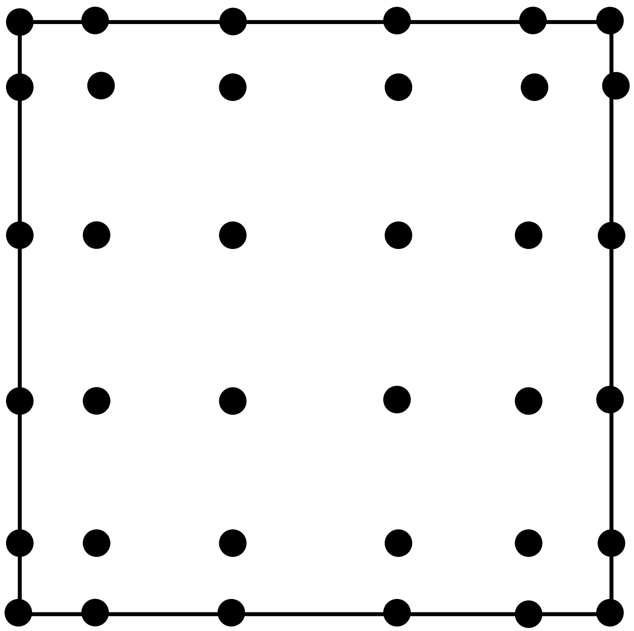
$$(-\omega^2 D_h \;+\; K_h)\,U_h \;=\; F_h$$

Mass matrix $D_h \;=\; \displaystyle\int_\Omega \rho\,\varphi_i^{GL}\,\varphi_j^{GL}\,dx$

Stiffness matrix $K_h \;=\; \displaystyle\int_\Omega \mu\,\nabla\varphi_i^{GL}\cdot\nabla\varphi_j^{GL}\,dx$

Our aim is to develop an efficient iterative solver for an high order of approximation r. We need then a fast matrix-vector product $(-\omega^2 D_h \;+\; K_h)\,U_h$

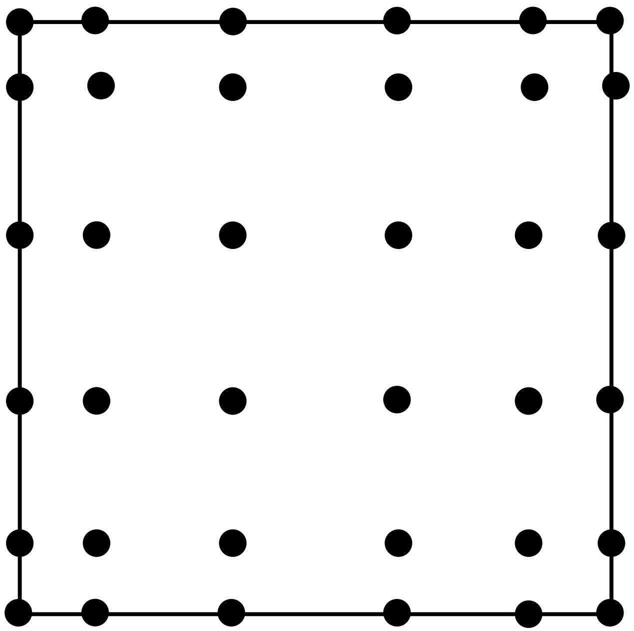
Use of Gauss-Lobatto points



Gauss-Lobatto points for Q_5

on the unit square $\hat{K}$

Use of Gauss-Lobatto points

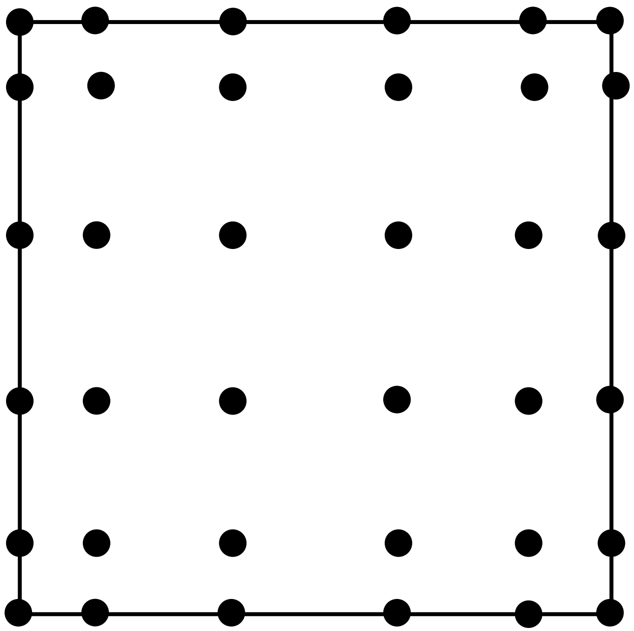


Gauss-Lobatto points for Q_5

on the unit square $\hat{K}$

Use of these points both for interpolation and numerical quadrature leads to a diagonal mass matrix D_h and a fast matrix-vector product for $K_h\, U_h$

Use of Gauss-Lobatto points

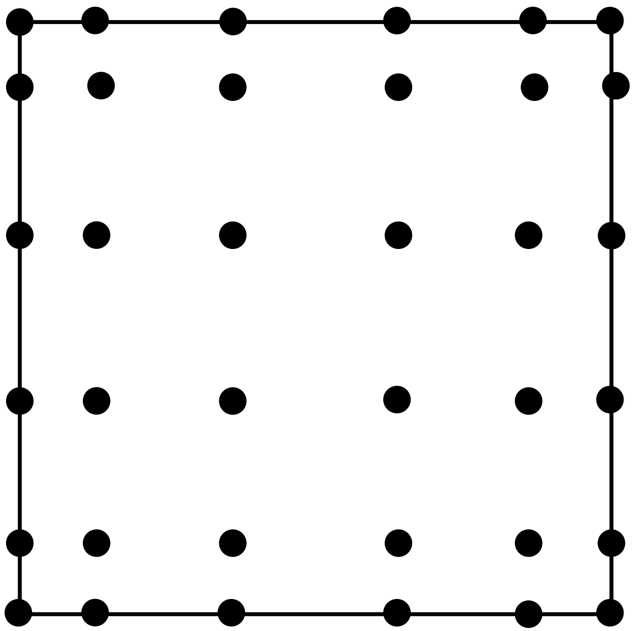


Gauss-Lobatto points for Q_5

on the unit square $\hat{K}$

Use of these points both for interpolation and numerical quadrature leads to a diagonal mass matrix D_h and a fast matrix-vector product for $K_h\,U_h$
See the thesis of S. Fauqueux, 2003

Use of Gauss-Lobatto points



Gauss-Lobatto points for Q_5

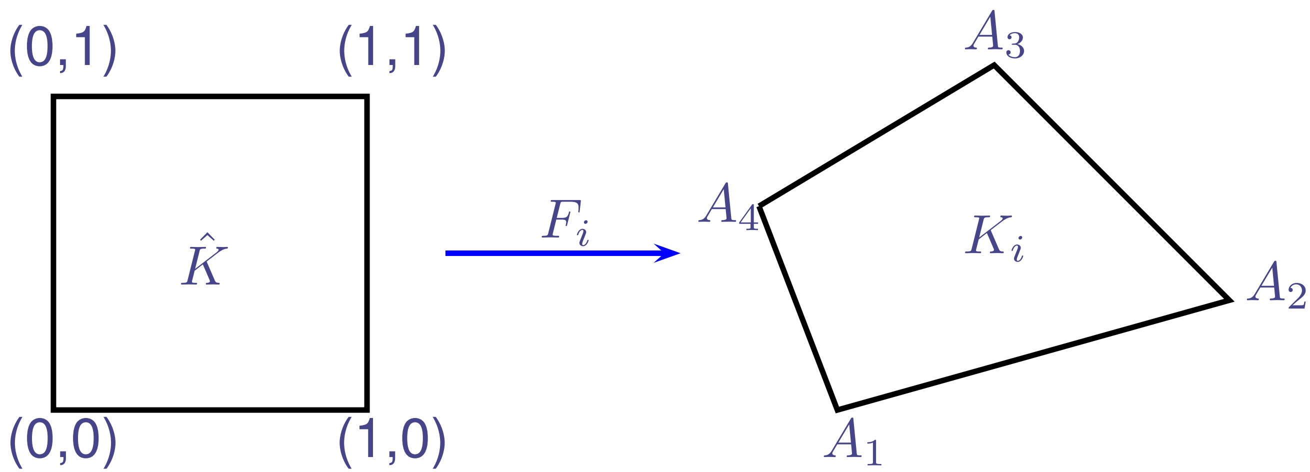
on the unit square $\hat{K}$

Use of these points both for interpolation and numerical quadrature leads to a diagonal mass matrix D_h and a fast matrix-vector product for $K_h U_h$

See the thesis of S. Fauqueux, 2003

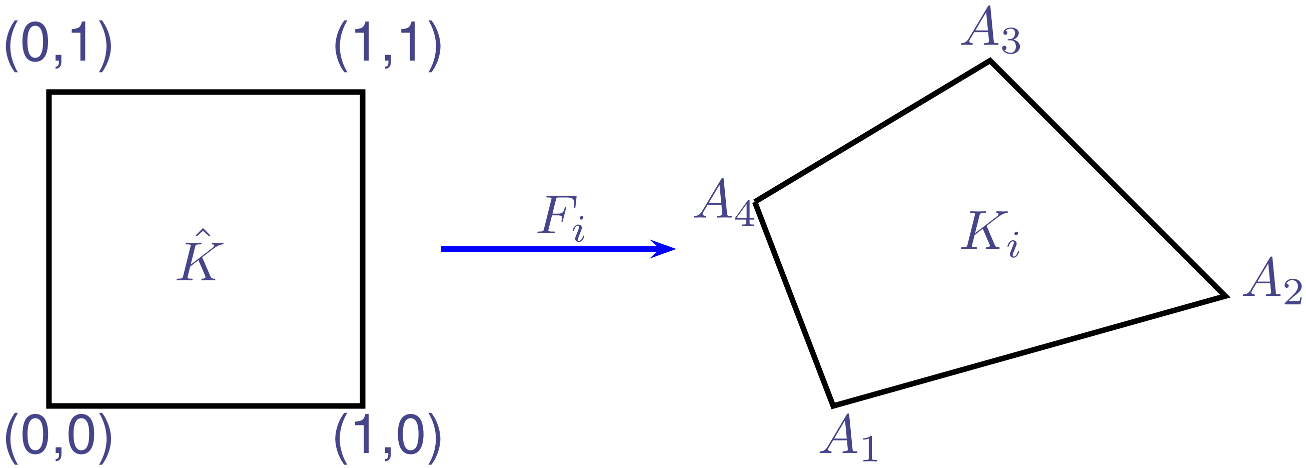
These points permit a fast matrix-vector product

Elementary matrices



The transformation F_i

Elementary matrices



The transformation F_i

$$(D_h)_{i,j} = \int_{\hat{K}} \rho J_i \, \hat{\varphi}_i^{GL} \, \hat{\varphi}_j^{GL} \, d\hat{x}$$

$$(K_h)_{i,j} = \int_{\hat{K}} \mu \, J_i \, DF_i^{-1} \, DF_i^{*-1} \, \hat{\nabla} \, \hat{\varphi}_i^{GL} \cdot \hat{\nabla} \hat{\varphi}_j^{GL} \, d\hat{x}$$

Elementary matrices

$$(D_h)_{i,j} = \int_{\hat{K}} \rho J_i \, \hat{\varphi}_i^{GL} \, \hat{\varphi}_j^{GL} \, d\hat{x}$$

$$(K_h)_{i,j} = \int_{\hat{K}} \mu \, J_i \, DF_i^{-1} \, DF_i^{*-1} \, \hat{\nabla} \, \hat{\varphi}_i^{GL} \cdot \hat{\nabla} \hat{\varphi}_j^{GL} \, d\hat{x}$$

- Use of quadrature formulas (ω_k^X, ξ_k^X) on the unit square
  - X can be equal to GL (Gauss-Lobatto quadrature)
  - X can be equal to G (Gauss quadrature)

Elementary matrices

$$(D_h)_{i,j} = \int_{\hat{K}} \rho J_i \, \hat{\varphi}_i^{GL} \, \hat{\varphi}_j^{GL} \, d\hat{x}$$

$$(K_h)_{i,j} = \int_{\hat{K}} \mu \, J_i \, DF_i^{-1} \, DF_i^{*-1} \, \hat{\nabla} \, \hat{\varphi}_i^{GL} \cdot \hat{\nabla} \hat{\varphi}_j^{GL} \, d\hat{x}$$

- Use of quadrature formulas (ω_k^X, ξ_k^X) on the unit square
- Diagonal matrix

$$(A_h)_{k,k} = \rho \, J_i(\xi_k^X) \, \omega_k^X$$

- Bloc-diagonal matrix

$$(B_h)_{k,k} = \mu \, J_i \, DF_i^{-1} \, DF_i^{*-1}(\xi_k^X) \, \omega_k^X$$

Fast matrix vector product with any points

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} = \hat{\varphi}_i^{GL}(\xi_j^X) \qquad \hat{R}_{i,j} = \hat{\nabla}\hat{\varphi}_i^X(\xi_j^X)$$

Fast matrix vector product with any points

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} = \hat{\varphi}_i^{GL}(\xi_j^X) \qquad \hat{R}_{i,j} = \hat{\nabla}\hat{\varphi}_i^X(\xi_j^X)$$

Thus, we have : $D_h = \hat{C}\, A_h \hat{C}^*$ $\qquad K_h = \hat{C}\hat{R}\, B_h \hat{R}^* \hat{C}^*$

Fast matrix vector product with any points

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} = \hat{\varphi}_i^{GL}(\xi_j^X) \qquad \hat{R}_{i,j} = \hat{\nabla}\hat{\varphi}_i^X(\xi_j^X)$$

Thus, we have : $D_h = \hat{C}\, A_h \hat{C}^* \qquad K_h = \hat{C}\hat{R}\, B_h \hat{R}^* \hat{C}^*$

r is the order of approximation

If $\hat{C}$ and $\hat{R}$ are stored as full matrices

- Complexity of $\hat{C}\, U$: $2\,(r+1)^6$ operations in 3-D
- Complexity of $\hat{R}\, U$: $6\,(r+1)^6$ operations in 3-D

Complexity of standard matrix vector product : $2\,(r+1)^6$ operations in 3-D

Fast matrix vector product with any points

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} = \hat{\varphi}_i^{GL}(\xi_j^X) \qquad \hat{R}_{i,j} = \hat{\nabla}\hat{\varphi}_i^X(\xi_j^X)$$

Thus, we have : $D_h = \hat{C}\, A_h \hat{C}^* \qquad K_h = \hat{C}\hat{R}\, B_h \hat{R}^* \hat{C}^*$

For hexahedral elements (tensorization), we have

- Complexity of $\hat{C}\, U$: $6\,(r+1)^4$ operations in 3-D
- Complexity of $\hat{R}\, U$: $6\,(r+1)^4$ operations in 3-D
- Complexity of $A_h\, U$ and $B_h\, V$: $16\,(r+1)^3$ operations in 3-D

Fast matrix vector product with any points

Let us introduce the two following matrices, independant of the geometry :

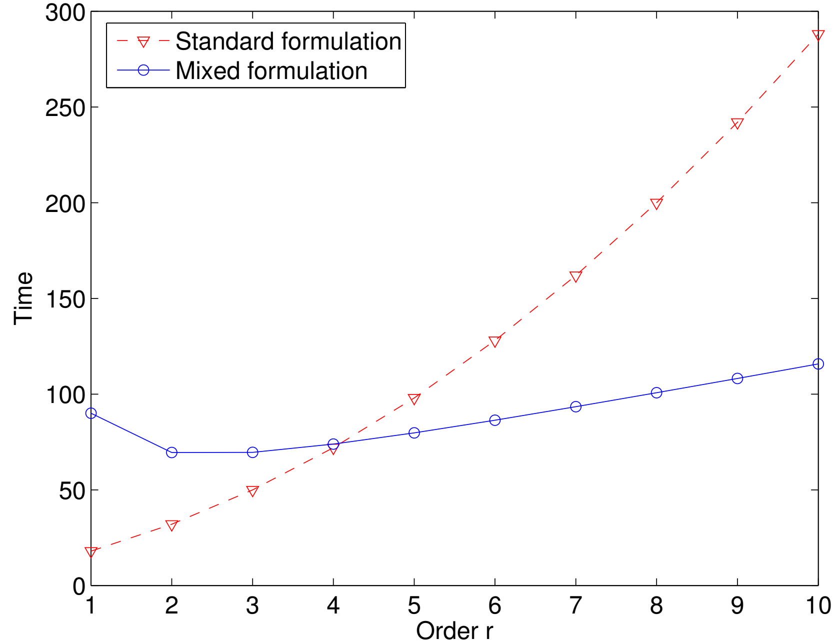
$$\hat{C}_{i,j} = \hat{\varphi}_i^{GL}(\xi_j^X) \qquad \hat{R}_{i,j} = \hat{\nabla}\hat{\varphi}_i^X(\xi_j^X)$$

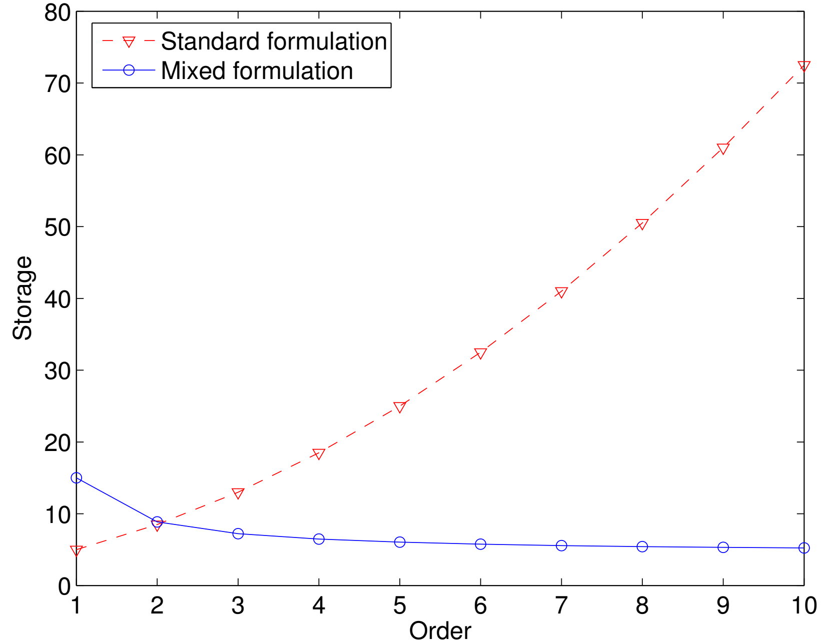
Thus, we have : $D_h = \hat{C}\, A_h \hat{C}^* \qquad K_h = \hat{C}\hat{R}\, B_h \hat{R}^* \hat{C}^*$

For hexahedral elements (tensorization), we have

- Complexity of $\hat{C}\, U$: $6\,(r+1)^4$ operations in 3-D
- Complexity of $\hat{R}\, U$: $6\,(r+1)^4$ operations in 3-D
- Complexity of $A_h\, U$ and $B_h\, V$: $16\,(r+1)^3$ operations in 3-D
- If we use Gauss-Lobatto points to integrate : $\hat{C} = I$
  In this case : “equivalence theorem” of S. Fauqueux
- Same storage for Gauss or GL points (A_h and B_h)
- MV product two times slower with Gauss integration

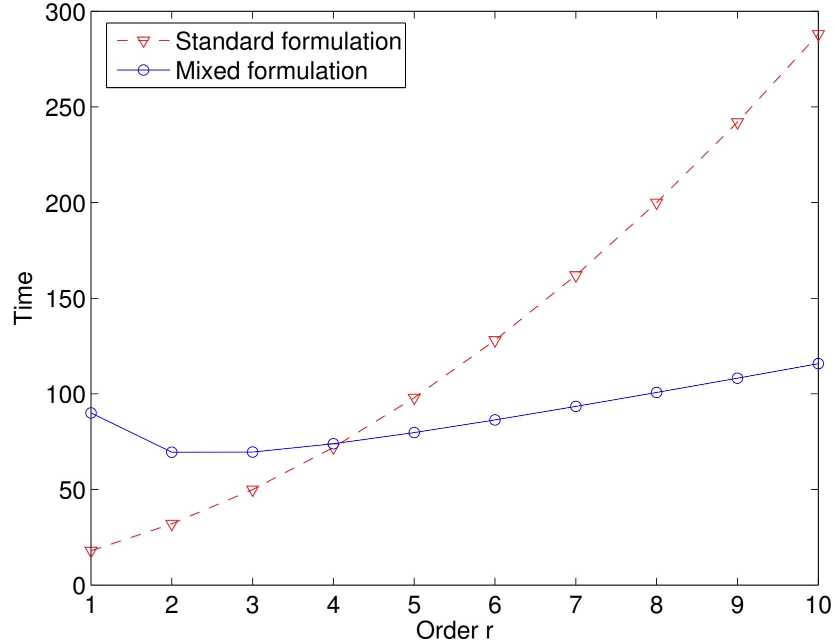
Matrix vector-product faster than standard

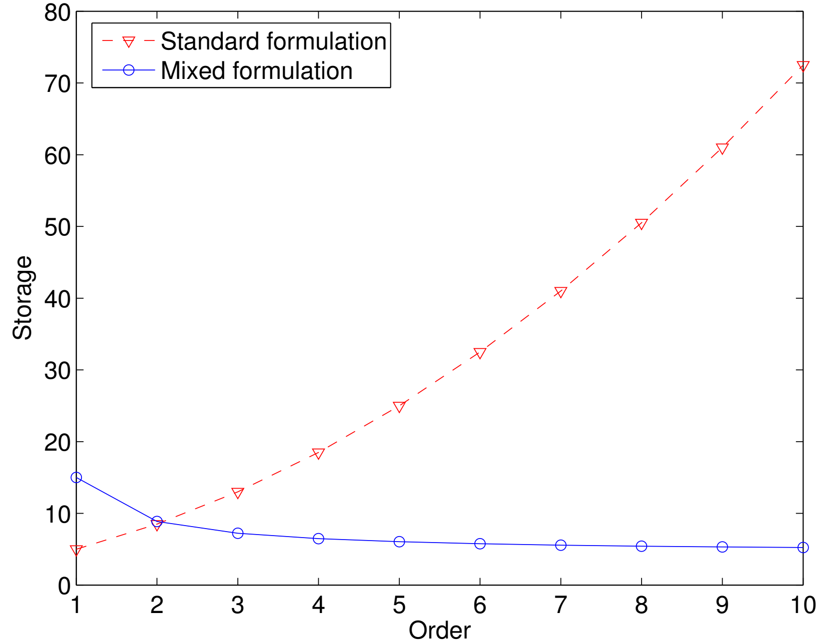




3-D comparison between the classical matrix-vector algorithm and the fast algorithm (mixed formulation), in 3-D. At left, time according to the order of approximation, at right storage.

Matrix vector-product faster than standard

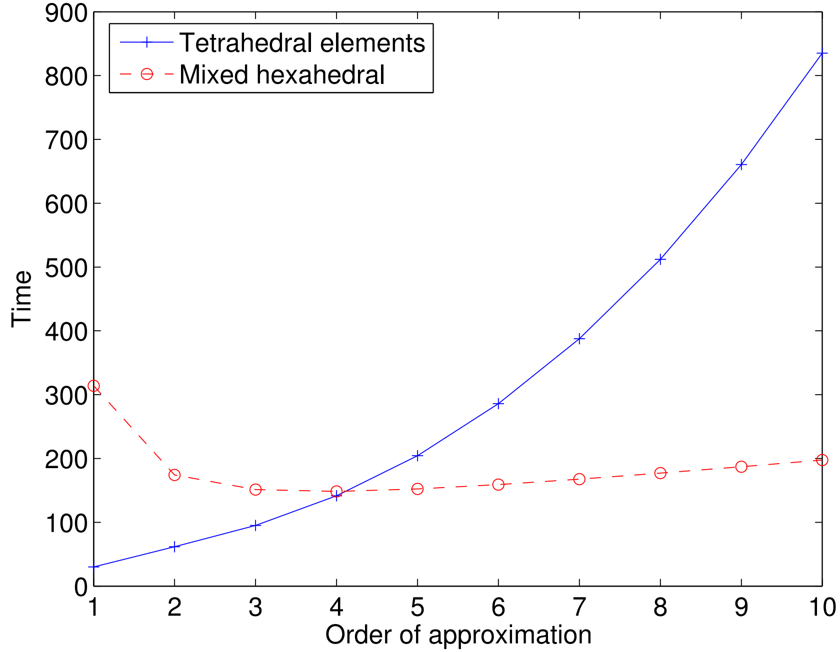


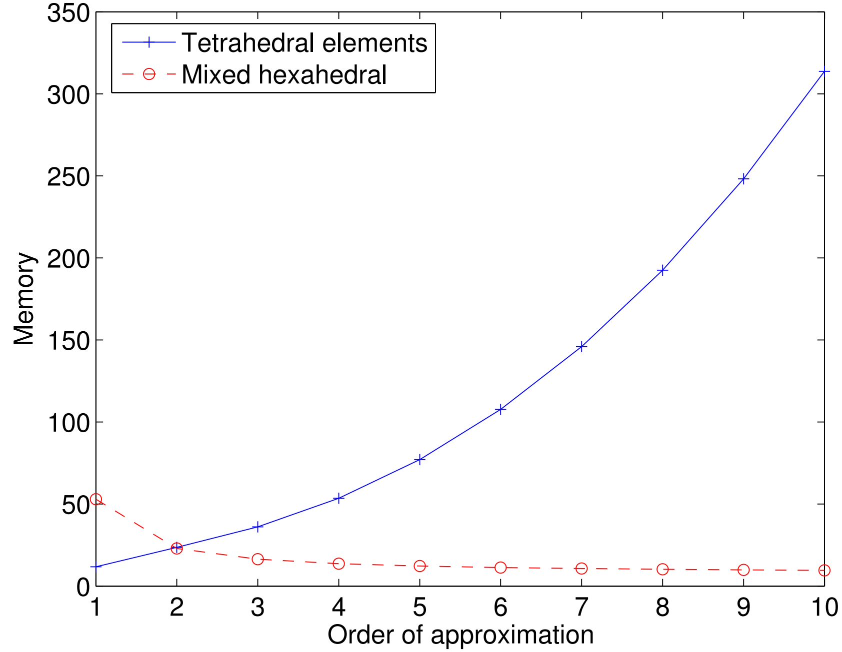


3-D comparison between the classical matrix-vector algorithm and the fast algorithm (mixed formulation), in 3-D. At left, time according to the order of approximation, at right storage.

Gain in time for $r \geq 4$, gain in storage for $r \geq 2$.

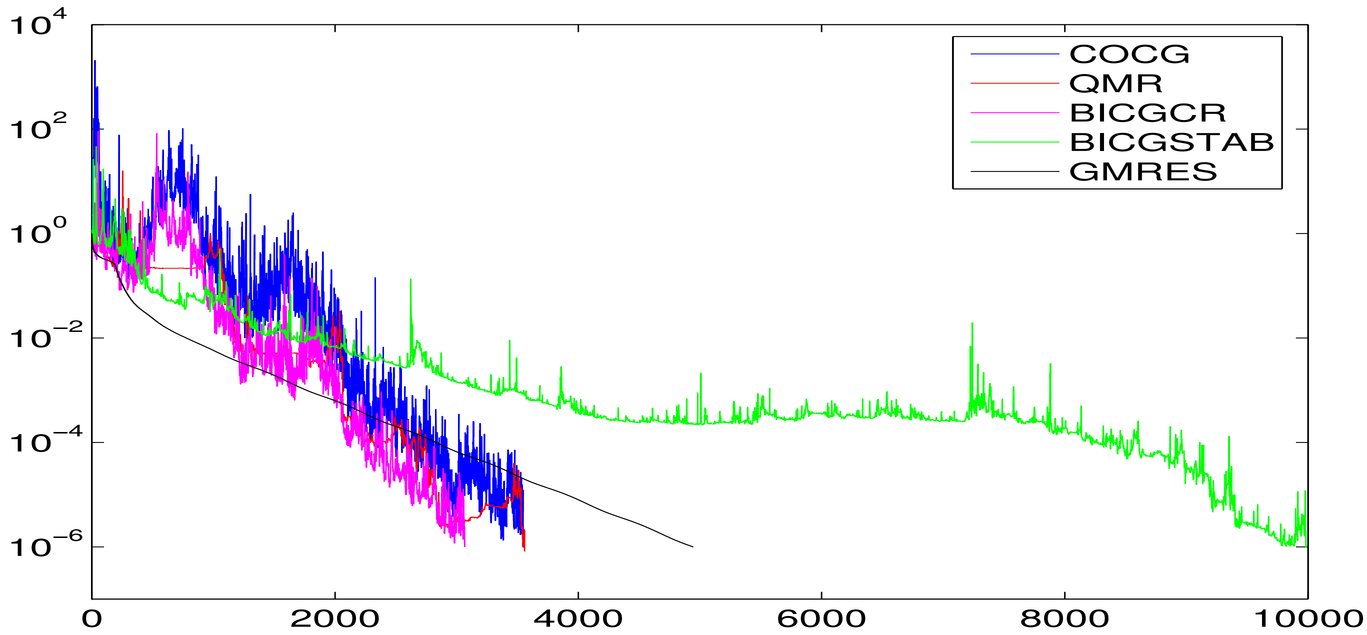
Matrix vector-product faster than standard





Comparison between hexahedral and tetrahedral elements, for time computation (at left) and storage (at right)

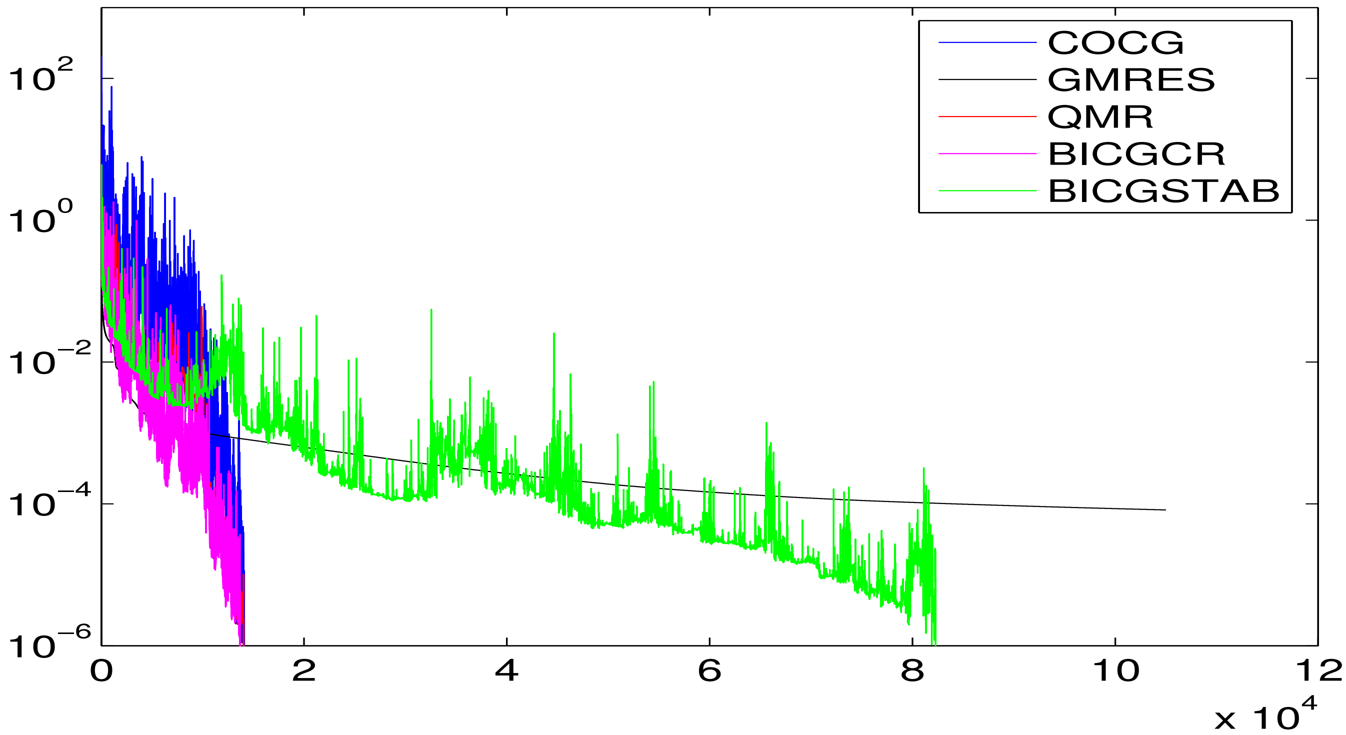
Iterative methods used



Evolution of the residual norm for the scattering of a perfectly conductor disc (Dirichlet condition).

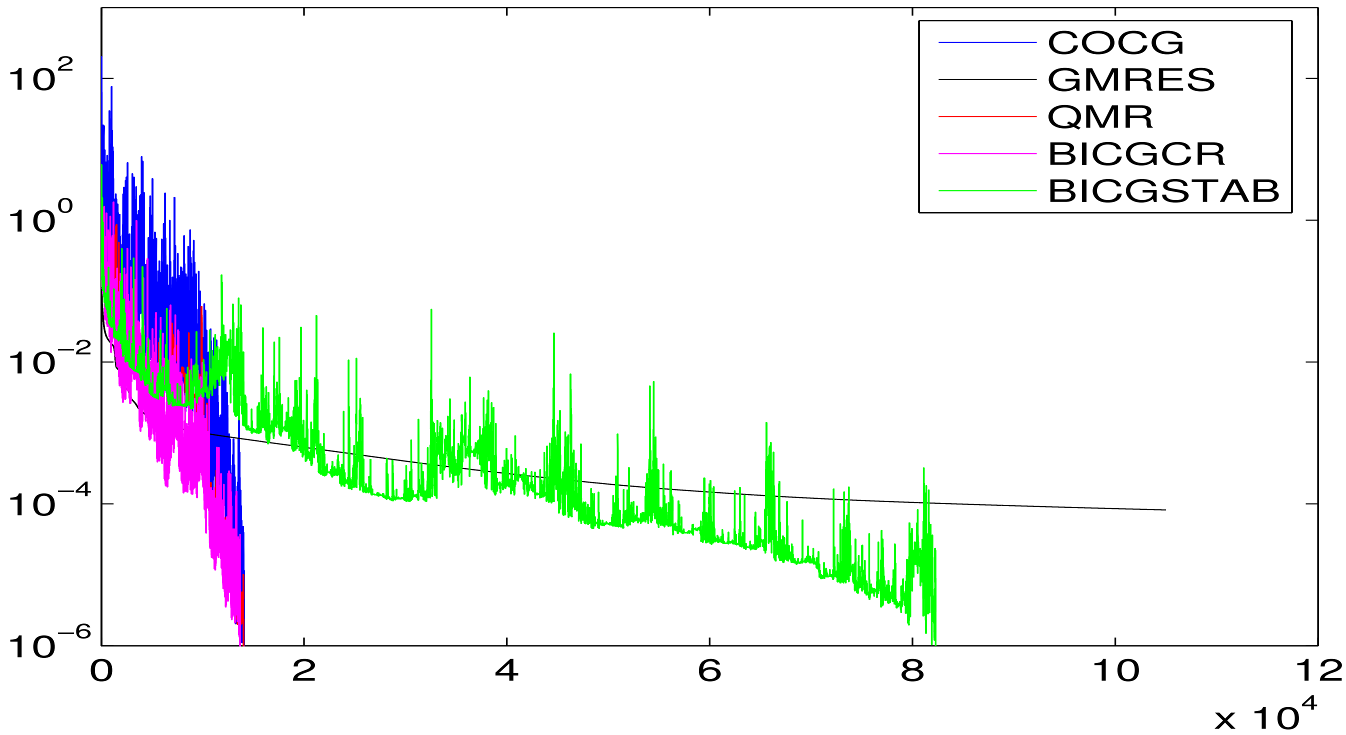
- GMRES, BICGSTAB and QMR for complex unsymmetric matrices
- COCG, BICGCR for complex symmetric matrices

Iterative methods used



Evolution of the residual norm for the scattering of a dielectric disc ($\rho = 4$).

Iterative methods used



- We choose to use BICGCR for all future experiments
- Need of preconditioning techniques to have less iterations

Preconditioning used

- Incomplete factorization with threshold on the damped Helmholtz equation :

$$-k^2(\alpha + i\beta)u - \Delta u = 0$$

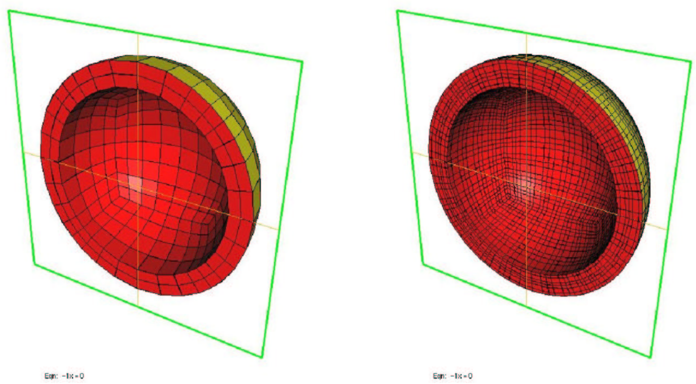
  - see Y. Saad, Iterative methods for sparse linear systems

Preconditioning used

- Incomplete factorization with threshold on the damped Helmholtz equation :

$$-k^2(\alpha + i\beta)u - \Delta u = 0$$

  - see Y. Saad, Iterative methods for sparse linear systems
  - We use a Q_1 subdivided mesh to compute matrix



At left, initial mesh Q_3, at right, subdivided mesh Q_1

Preconditioning used

- Incomplete factorization with threshold on the damped Helmholtz equation :

$$-k^2(\alpha + i\beta)u - \Delta u = 0$$

  - see Y. Saad, Iterative methods for sparse linear systems

- Multigrid method on the damped Helmholtz equation
  - see Y. A. Erlangga and al, Report of Delft University Technology, 2004

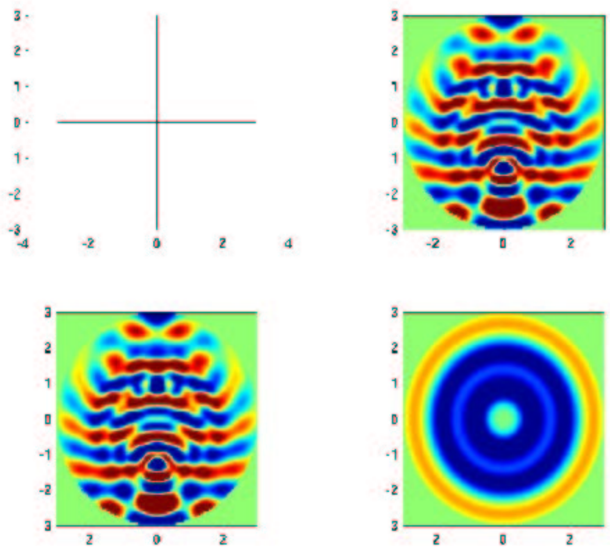
Preconditioning used

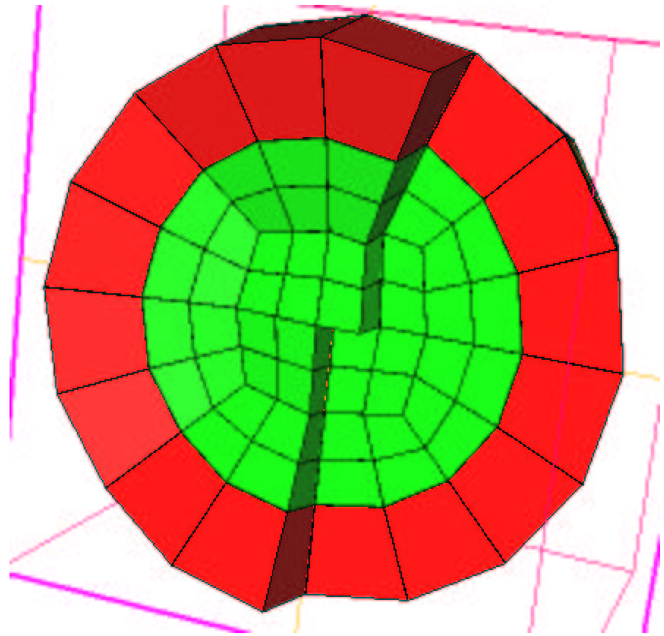
- Incomplete factorization with threshold on the damped Helmholtz equation :

$$-k^2(\alpha + i\beta)u - \Delta u = 0$$

  - see Y. Saad, Iterative methods for sparse linear systems
- Multigrid method on the damped Helmholtz equation
  - see Y. A. Erlangga and al, Report of Delft University Technology, 2004
- Without damping, both preconditioners **doesn't lead** to convergence.
- A good choice of parameter is $\alpha = 1, \ \beta = 0.5$

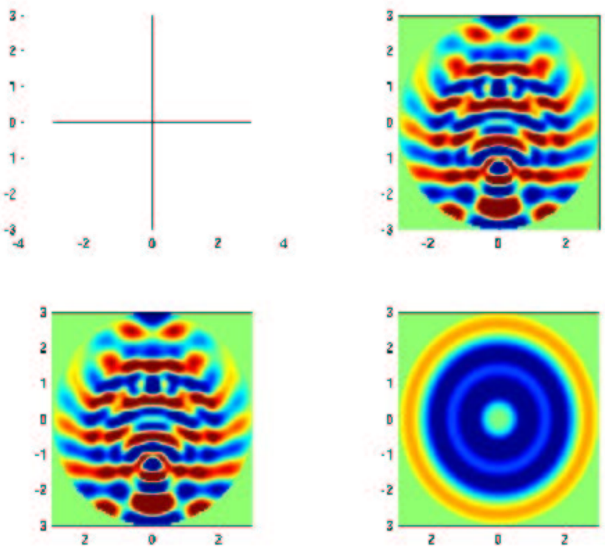
Scattering by a dielectric sphere

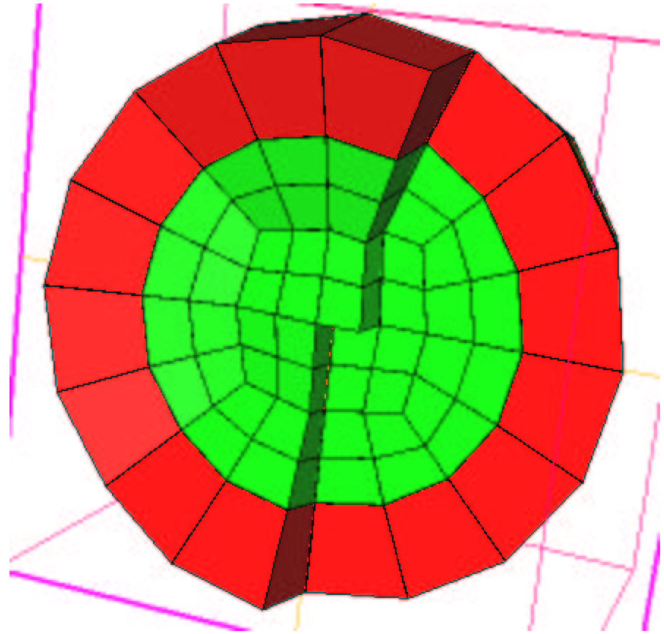




- Dielectric sphere of radius 2 and with $\rho = 4 \quad \omega = 2\pi$
- First order absorbing boundary condition on a sphere of radius 3

Scattering by a dielectric sphere

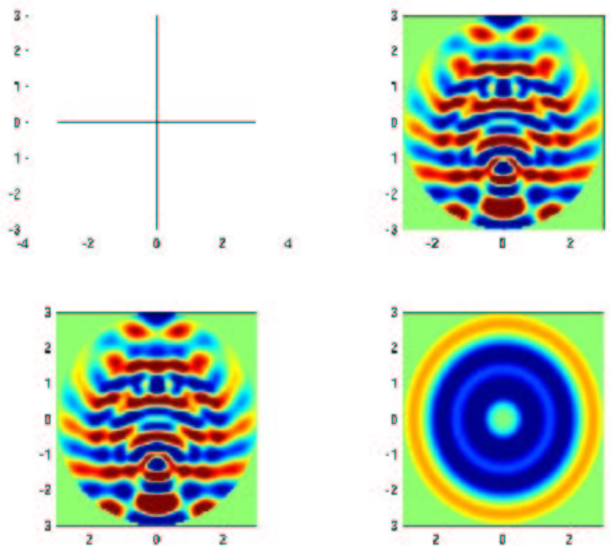


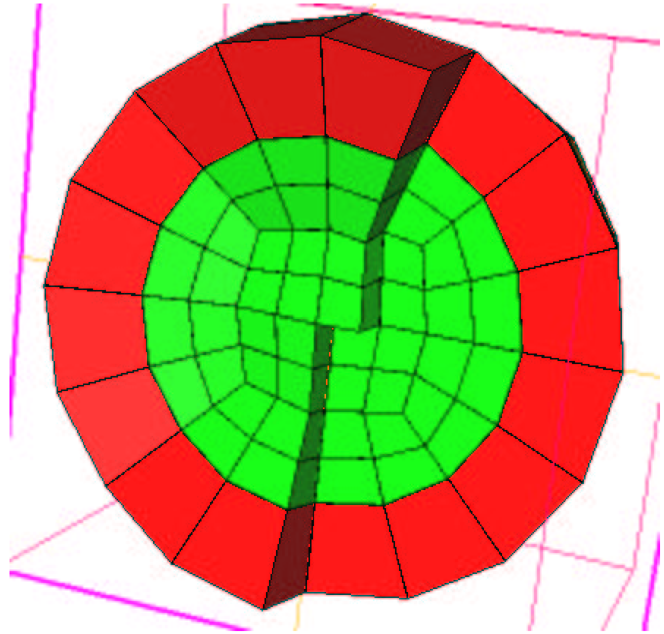


Number of dofs to reach less than 5 % L^2 error

Finite element	structured $\mathbf{Q}_2$	struct $\mathbf{Q}_4$	struct $\mathbf{Q}_6$	n.s. $\mathbf{Q}_4$	n.s. $\mathbf{P}_4$
Number of dofs	220 000	85 000	78 000	243 000	180 000

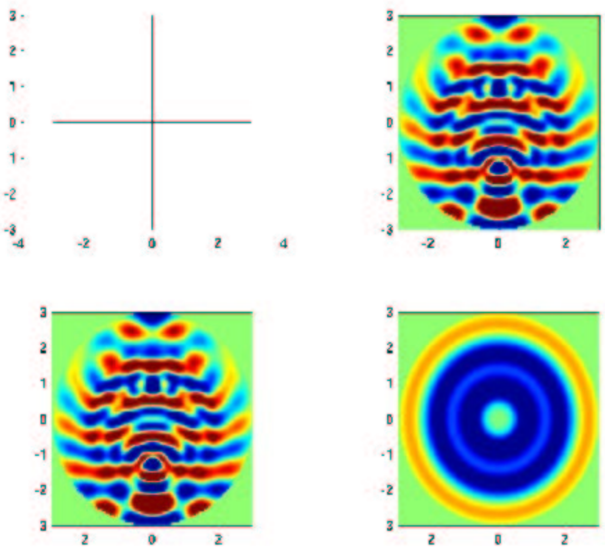
Scattering by a dielectric sphere

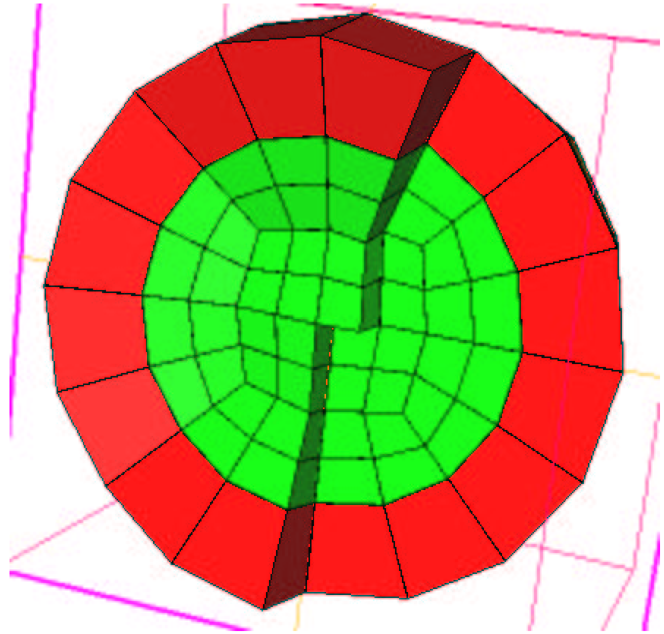




Finite element	structured $\mathbf{Q}_4$	non-structured $\mathbf{Q}_4$	non-structured $\mathbf{P}_4$
No preconditioning	708 s	5 795 s	1 597 s
ILUT(0.01)	91 s	534 s	363 s
Multigrid	185 s	729 s	695 s

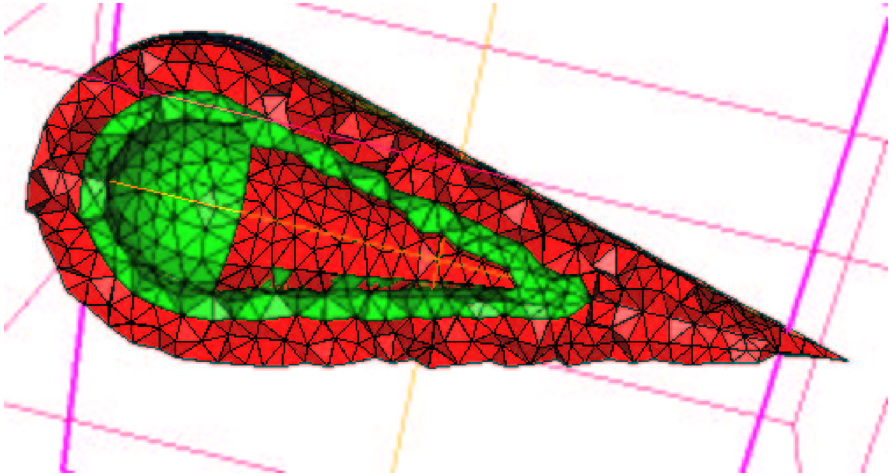
Scattering by a dielectric sphere

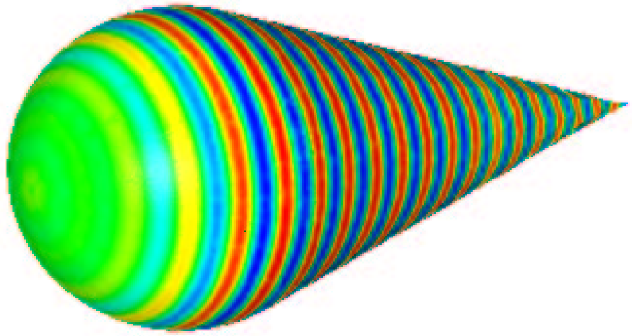




Finite element	structured $\mathbf{Q}_4$	non-structured $\mathbf{Q}_4$	non-structured $\mathbf{P}_4$
No preconditioning	34 Mo	99 Mo	136 Mo
ILUT(0.01)	137 Mo	420 Mo	507 Mo
Multigrid	50 Mo	143 Mo	327 Mo

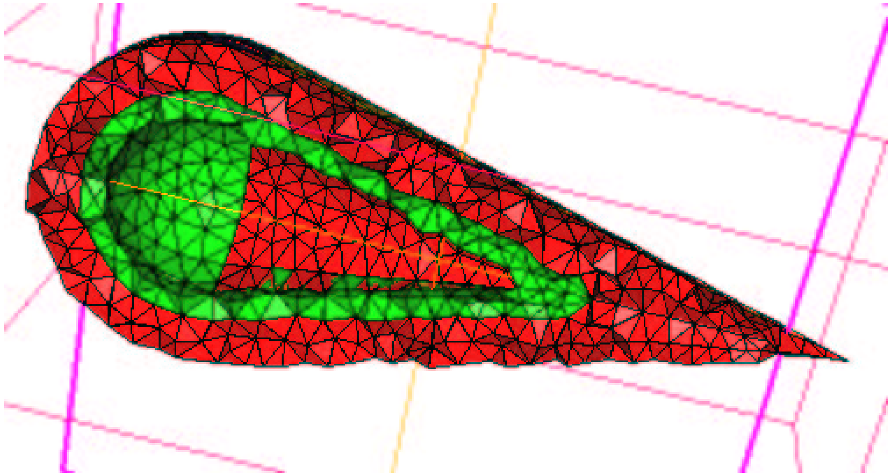
Scattering by a coated cone-sphere

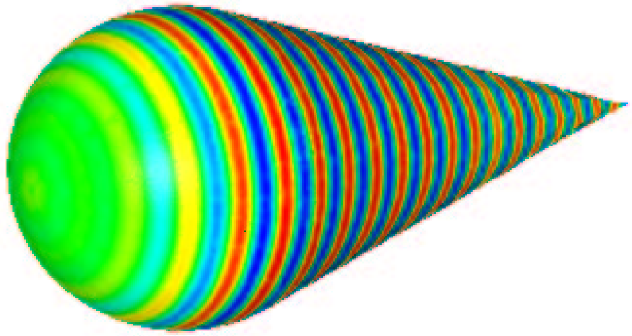




- Coated cone-sphere of radius 2 and length 12
- Dielectric layer of thickness 0.8 with
  $\rho = 3 + 0.5i \quad \mu = 0.5 - 0.5i$
- First order absorbing boundary condition on the outside boundary

Scattering by a coated cone-sphere

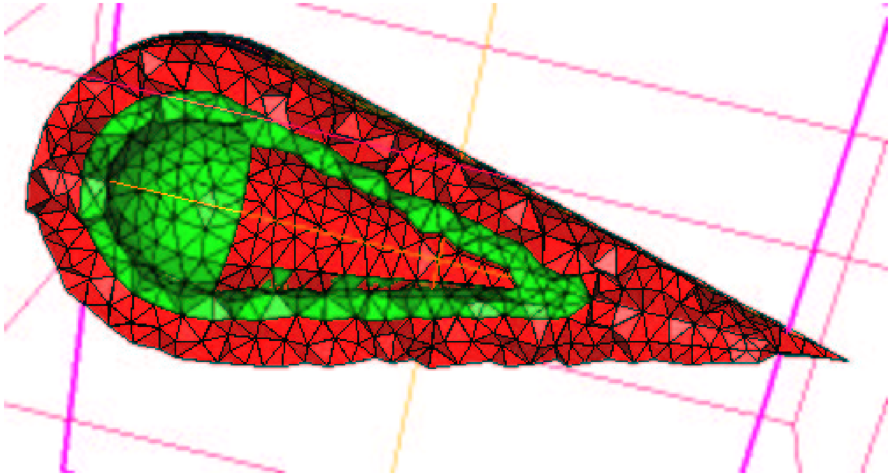


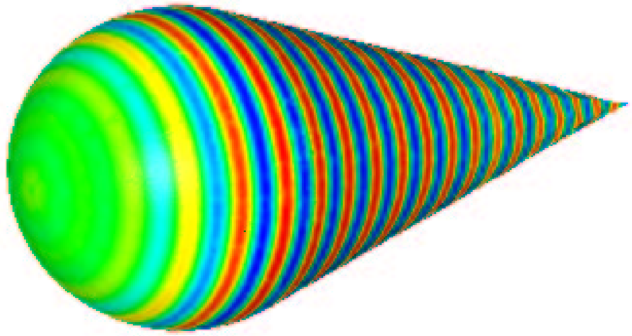


Number of dofs to reach less than 5 % L^2 error

Finite element	n.s. $\mathbf{Q_2}$	n.s. $\mathbf{Q_4}$	n.s. $\mathbf{P_2}$	n.s. $\mathbf{P_4}$
Number of dofs	494 000	3 838 000	178 000	166 000

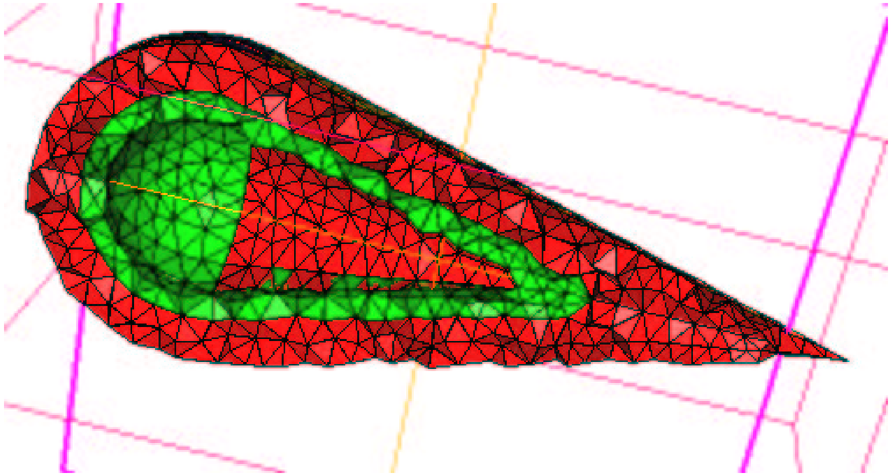
Scattering by a coated cone-sphere

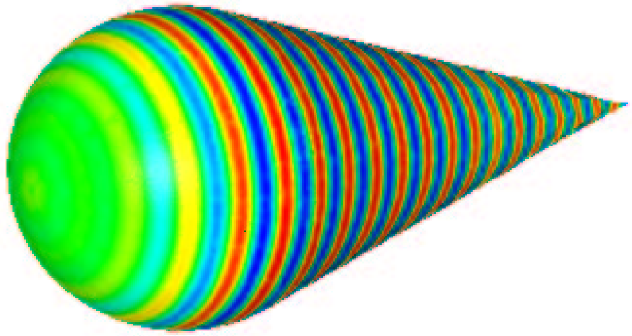




Finite element	n.s. $\mathbf{Q_2}$	n.s. $\mathbf{Q_4}$	n.s. $\mathbf{P_2}$	n.s. $\mathbf{P_4}$
No preconditioning	1 787 s	42 200 s	193 s	516 s
ILUT(0.01)	370 s	-	24 s	27 s
Multigrid	274 s	1 426 s	21 s	107 s

Scattering by a coated cone-sphere

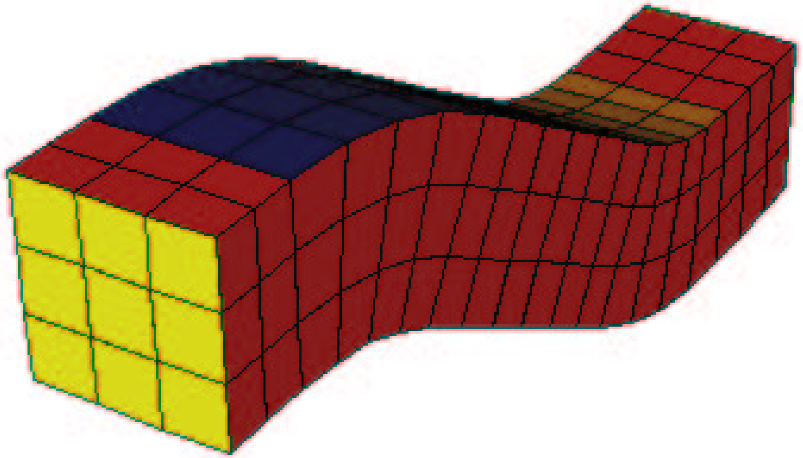
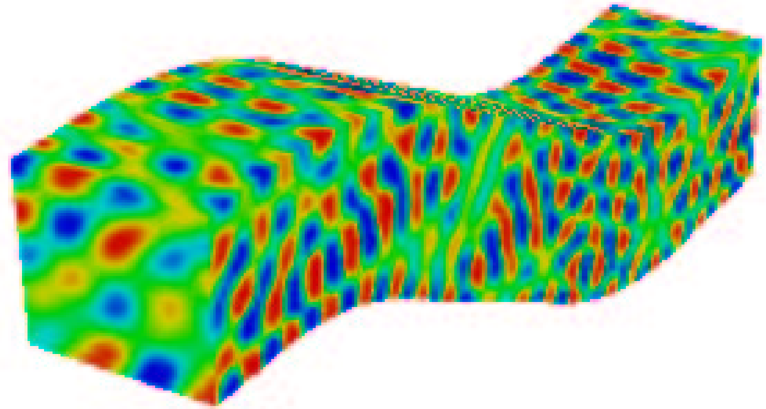




Finite element	n.s. $\mathbf{Q_2}$	n.s. $\mathbf{Q_4}$	n.s. $\mathbf{P_2}$	n.s. $\mathbf{P_4}$
No preconditioning	1 787 s	42 200 s	193 s	516 s
ILUT(0.01)	370 s	-	24 s	27 s
Multigrid	274 s	1 426 s	21 s	107 s

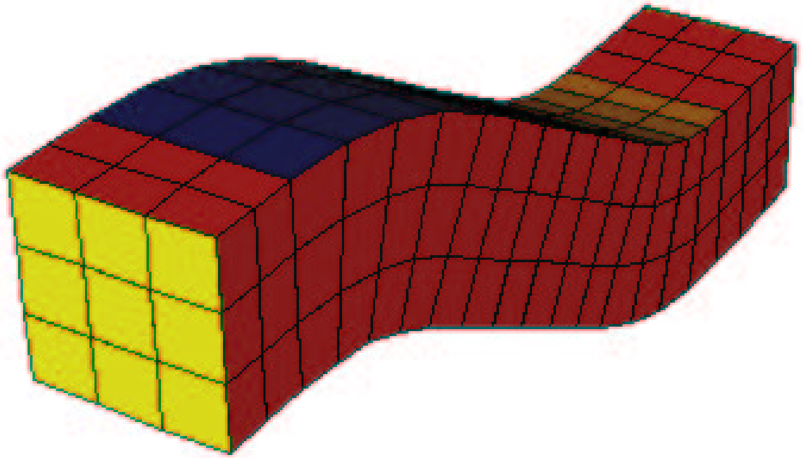
Finite element	n.s. $\mathbf{Q_2}$	n.s. $\mathbf{Q_4}$	n.s. $\mathbf{P_2}$	n.s. $\mathbf{P_4}$
No preconditioning	447 Mo	1 590 Mo	150 Mo	150 Mo
ILUT(0.01)	1 100 Mo	-	350 Mo	417 Mo
Multigrid	609 Mo	2 340 Mo	311 Mo	326 Mo

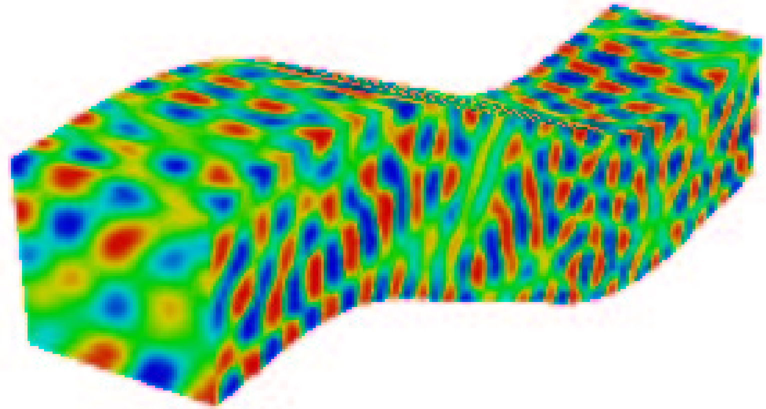
Scattering by a cobra cavity

- Cobra cavity of length 20, and depth 4
- First order absorbing boundary condition on the yellow face

Scattering by a cobra cavity

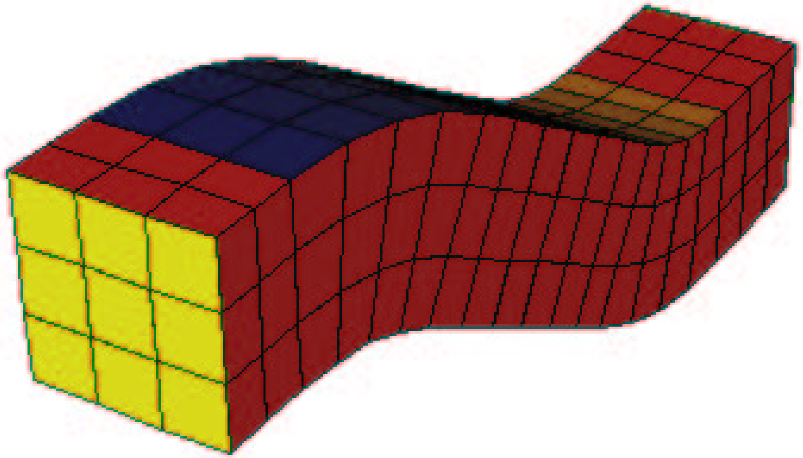


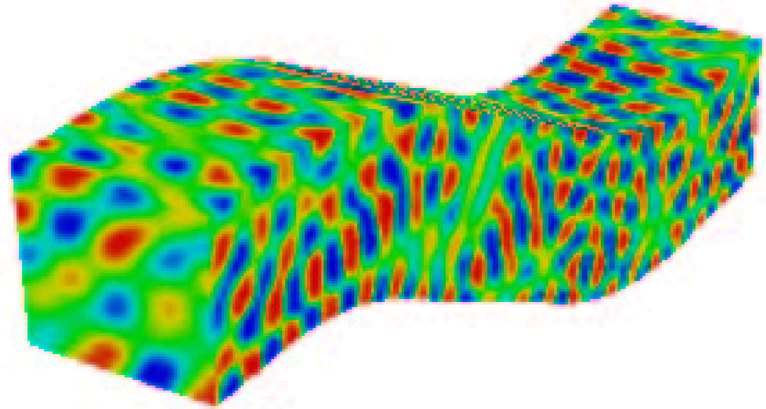


Number of dofs to reach less than 5 % L^2 error

Order	struct $\mathbf{Q_4}$	struct $\mathbf{Q_6}$	struct $\mathbf{Q_8}$	n.s. $\mathbf{Q_4}$	n.s. $\mathbf{Q_6}$	n.s. $\mathbf{P_4}$
Nb dofs	330 000	185 000	95 600	567,000	466 000	360 000

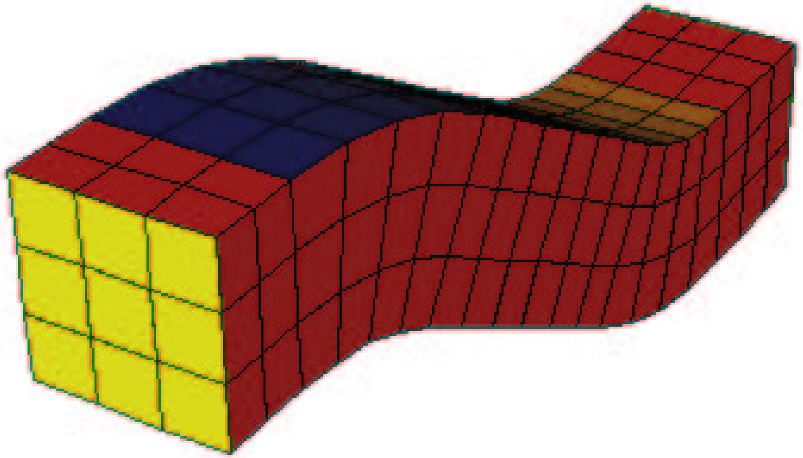
Scattering by a cobra cavity

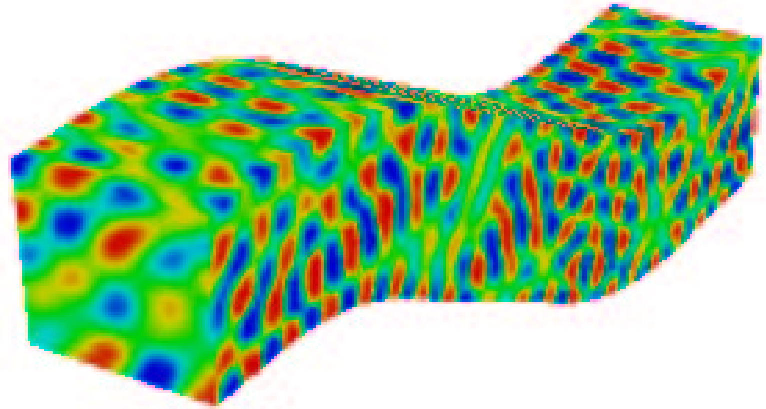




Finite element	structured $\mathbf{Q}_8$	non-structured $\mathbf{Q}_6$	non-structured $\mathbf{P}_4$
No preconditioning	9 860 s	NC	NC
ILUT(0.01)	1 021 s	13 766 s	8 036 s
Two-grid	1 082 s	6 821 s	14 016 s

Scattering by a cobra cavity





Finite element	structured $\mathbf{Q}_8$	non-structured $\mathbf{Q}_6$	non-structured $\mathbf{P}_4$
No preconditioning	9 860 s	NC	NC
ILUT(0.01)	1 021 s	13 766 s	8 036 s
Two-grid	1 082 s	6 821 s	14 016 s

Finite element	structured $\mathbf{Q}_8$	non-structured $\mathbf{Q}_6$	non-structured $\mathbf{P}_4$
No preconditioning	32 Mo	162 Mo	251 Mo
ILUT(0.01)	150 Mo	1 250 Mo	1 400 Mo
Two-grid	60 Mo	283 Mo	710 Mo

Outline

Resolution of Helmholtz equation

Resolution of time-harmonic Maxwell equations

Maxwell equations in axisymmetric domains

Nedelec's second family on quadrilaterals

Time-harmonic Maxwell's equations :

$$-\omega^2 \, \varepsilon \, \vec{E}(x) \; + \; \text{curl}\Big(\frac{1}{\mu(x)} \text{curl}(\vec{E}(x))\Big) \; = \; 0$$

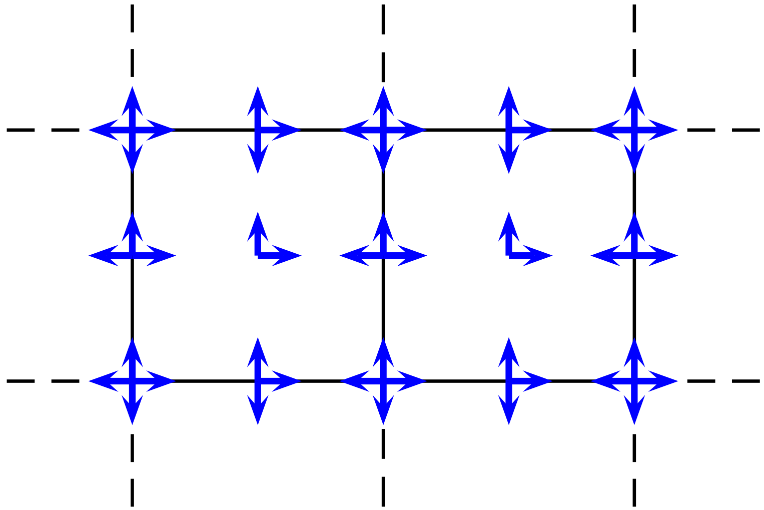
Space of approximation

$$V_h \; = \; \{ \, \vec{u} \in \; \text{H(curl,}\Omega\text{) such as } DF_i^* \, \vec{u} \circ F_i \; \in (Q_r)^2 \, \}$$

Nedelec's second family on quadrilaterals

Time-harmonic Maxwell's equations :

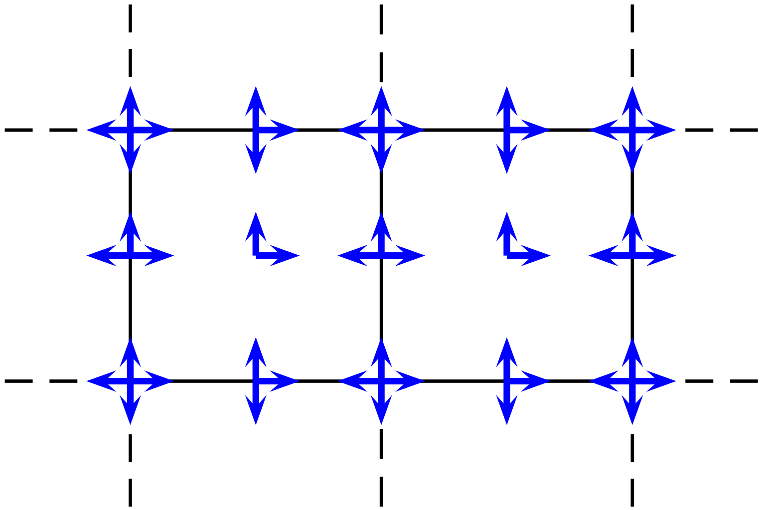
$$-\omega^2 \, \varepsilon \, \vec{E}(x) \; + \; \text{curl}\Big(\, \frac{1}{\mu(x)} \text{curl}(\vec{E}(x))\Big) \; = \; 0$$



Nedelec's second family on quadrilaterals

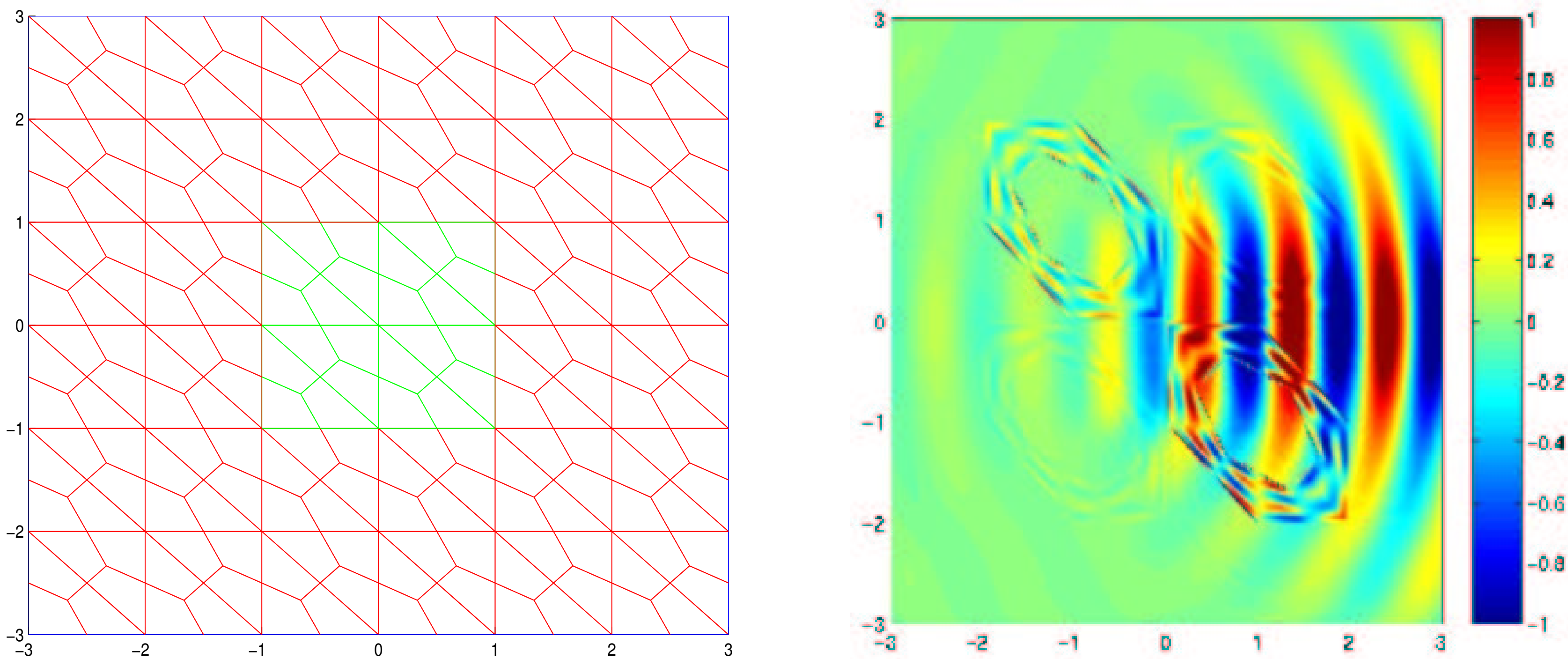
Time-harmonic Maxwell's equations :

$$-\omega^2 \, \varepsilon \, \vec{E}(x) \; + \; \text{curl}\Big(\frac{1}{\mu(x)} \text{curl}(\vec{E}(x))\Big) \; = \; 0$$



- Mass lumping and factorization of stiffness matrix
- Low-storage and fast matrix-vector product

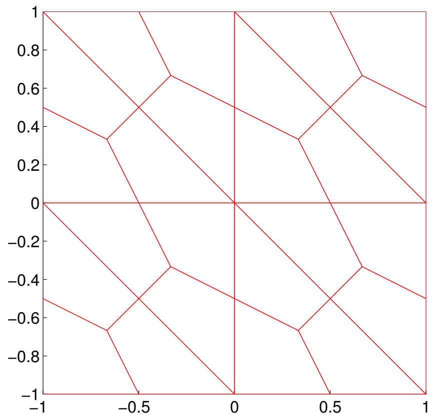
The unwanted oscillations



Scattering of a dielectric square. Left, mesh used for the simulations . Right, numerical solution with $\mathbf{Q}_5$ finite edge elements with mass-lumping.

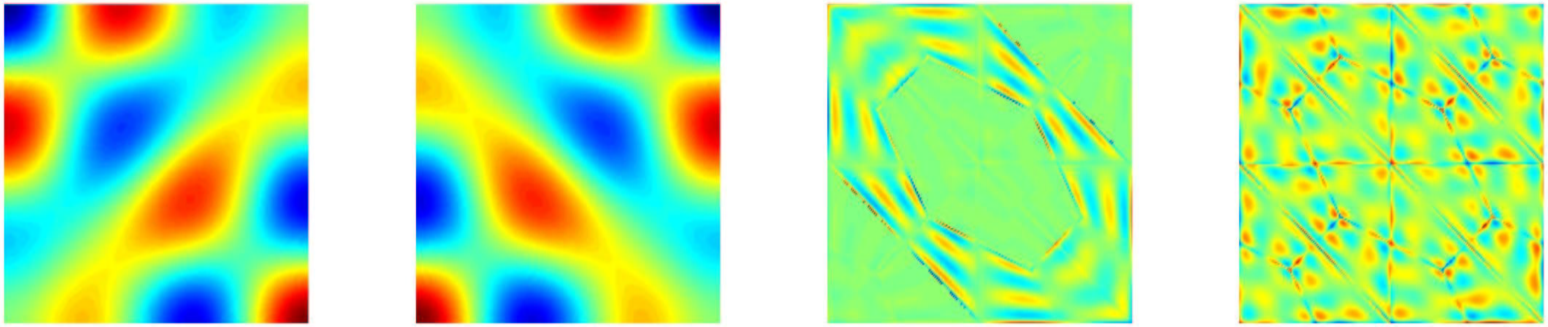
Eigenmodes with the second family

Mesh used for the simulations ($\mathbf{Q_5}$)



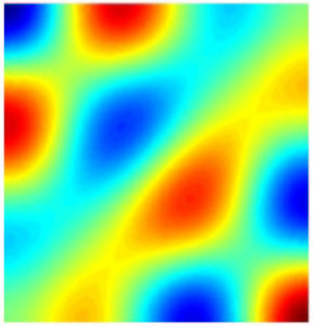
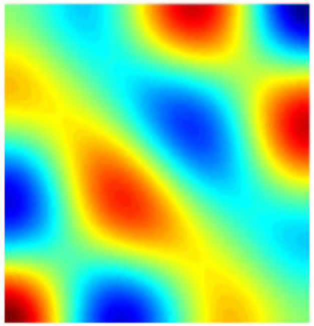
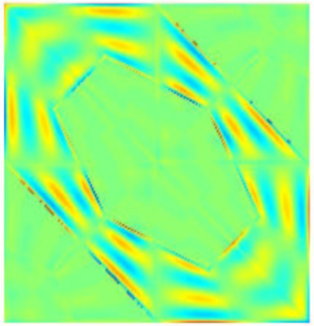
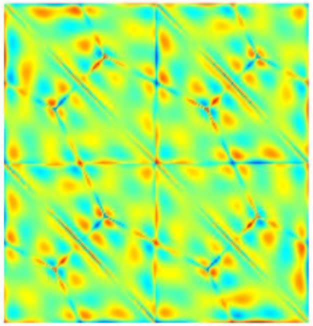
Eigenmodes with the second family

$\omega^2 = 32.08$ $\omega^2 = 32.08$ $\omega^2 = 37.54$ $\omega^2 = 37.95$

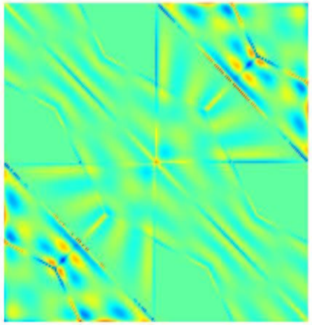
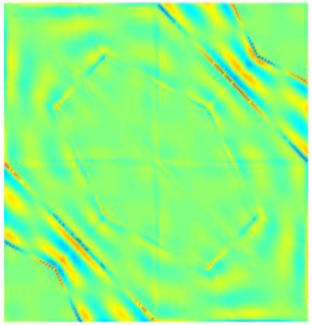
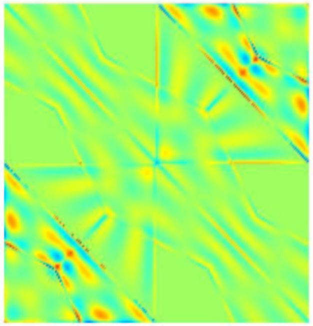
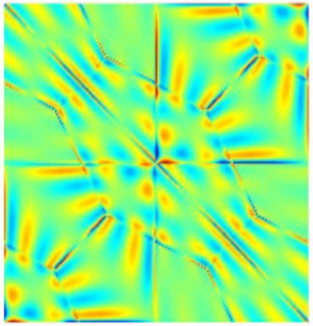


Eigenmodes with the second family

$\omega^2 = 32.08$ $\omega^2 = 32.08$ $\omega^2 = 37.54$ $\omega^2 = 37.95$

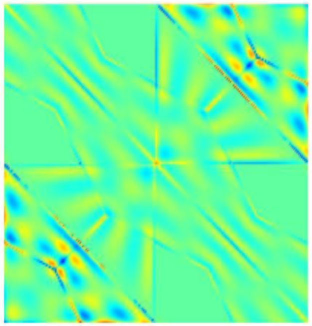
   

$\omega^2 = 37.98$ $\omega^2 = 38.00$ $\omega^2 = 38.03$ $\omega^2 = 38.03$

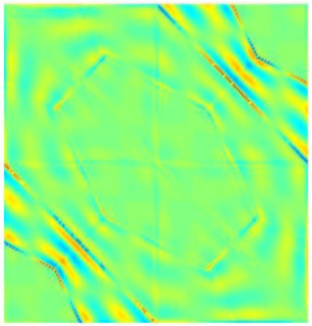
   

Eigenmodes with the second family

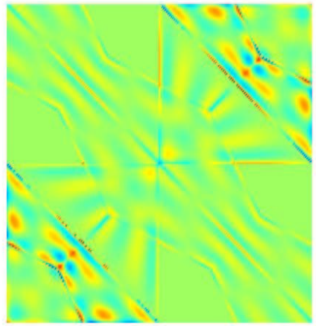
$\omega^2 = 37.98$



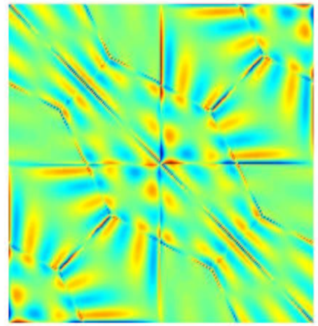
$\omega^2 = 38.00$



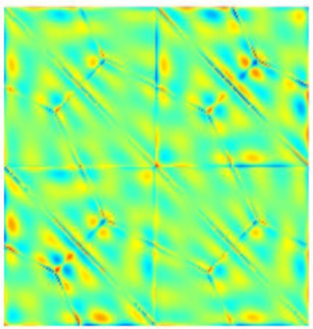
$\omega^2 = 38.03$



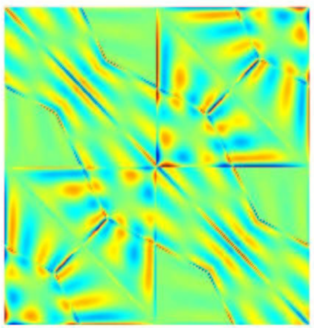
$\omega^2 = 38.03$



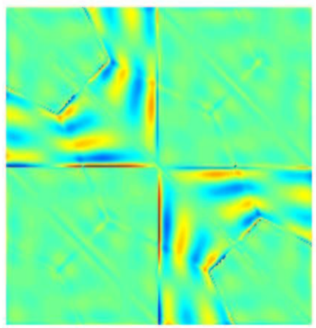
$\omega^2 = 38.04$



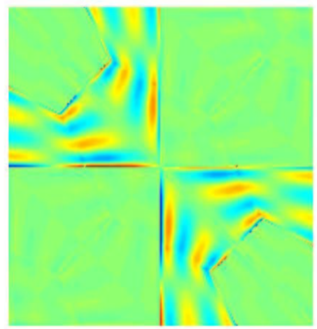
$\omega^2 = 38.05$



$\omega^2 = 38.07$

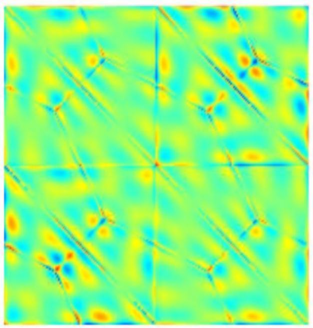
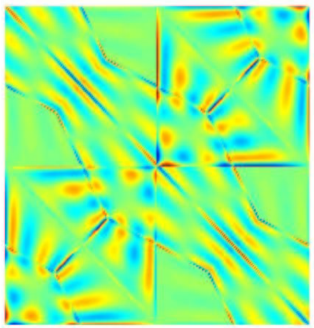
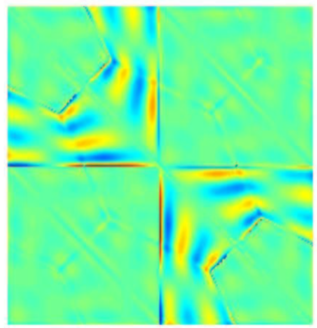
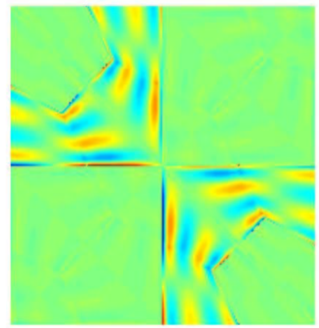


$\omega^2 = 38.20$

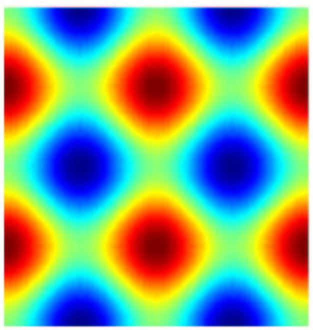
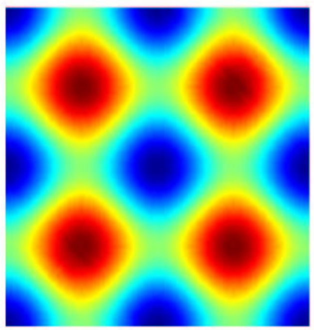
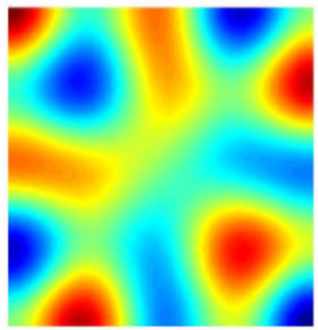
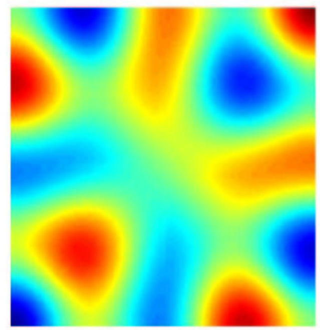


Eigenmodes with the second family

$\omega^2 = 38.04$ $\omega^2 = 38.05$ $\omega^2 = 38.07$ $\omega^2 = 38.20$

$\omega^2 = 39.48$ $\omega^2 = 39.48$ $\omega^2 = 41.95$ $\omega^2 = 41.95$

Discontinuous Galerkin method

$$
\begin{aligned}
-\omega \int_{K_i} \varepsilon \vec{E} \cdot \vec{\varphi} \; - \int_{K_i} H \, \nabla \times \vec{\varphi} \; - \int_{\partial K_i} \{H\} \, \vec{\varphi} \times \vec{\nu} &= 0 \\
-\omega \int_{K_i} \mu H \, \psi \; - \int_{K_i} \nabla \times \vec{E} \, \psi \; - \frac{1}{2} \int_{\partial K_i} [\vec{E}] \times \vec{\nu} \, \psi &= 0
\end{aligned}
$$

Let us notice that

$$
\begin{aligned}
\{H\} &= \tfrac{1}{2}(H_i + H_j) \\
[\,\vec{E}] &= (\vec{E}_i - \vec{E}_j)
\end{aligned} \tag{1}
$$

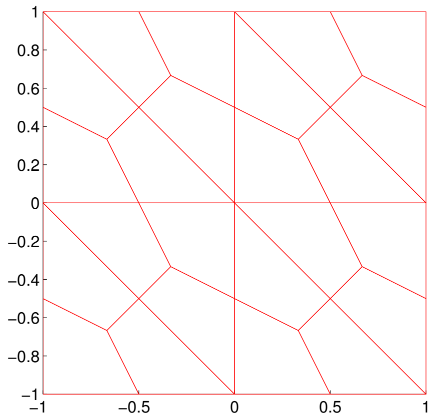
Discontinuous Galerkin method

$$
\begin{aligned}
-\omega \int_{K_i} \varepsilon \vec{E} \cdot \vec{\varphi} \;-\; \int_{K_i} H \, \nabla \times \vec{\varphi} \;-\; \int_{\partial K_i} \{H\} \, \vec{\varphi} \times \vec{\nu} &= 0 \\
-\omega \int_{K_i} \mu H \, \psi \;-\; \int_{K_i} \nabla \times \vec{E} \, \psi \;-\; \frac{1}{2} \int_{\partial K_i} [\vec{E}] \times \vec{\nu} \, \psi &= 0
\end{aligned}
$$

- Unknowns in $L^2 \Rightarrow$ Gauss points instead of GL points
- Mass lumping and fast matrix vector product
- Thesis of S. Pernet, in time-domain

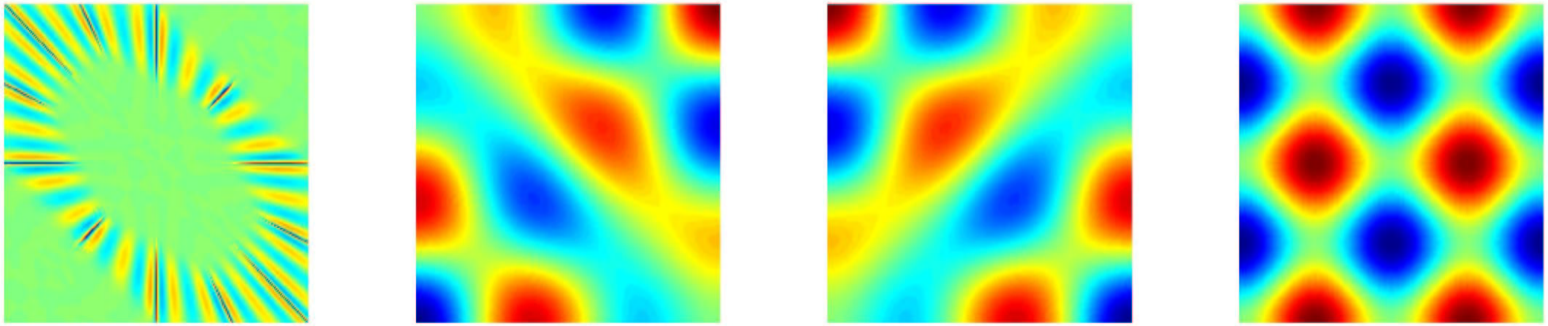
Eigenmodes in DG method (2-D)

Mesh used for the simulations ($\mathbf{Q_5}$)



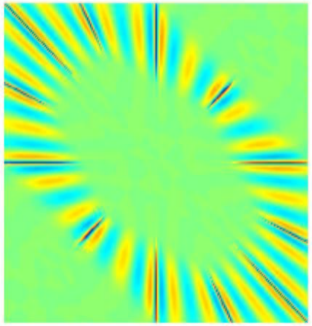
Eigenmodes in DG method (2-D)

$$\omega^2 = 26.92 \qquad \omega^2 = 32.08 \qquad \omega^2 = 32.08 \qquad \omega^2 = 39.48$$

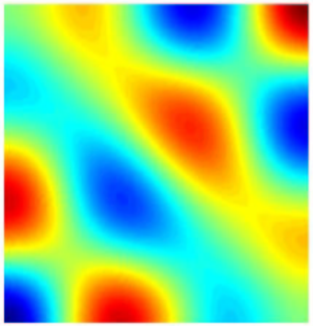


Eigenmodes in DG method (2-D)

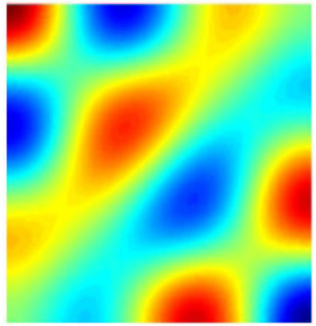
$\omega^2 = 26.92$



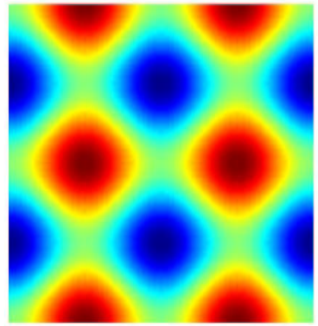
$\omega^2 = 32.08$



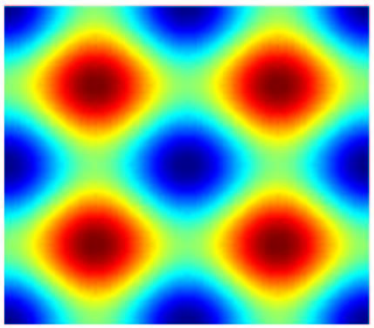
$\omega^2 = 32.08$



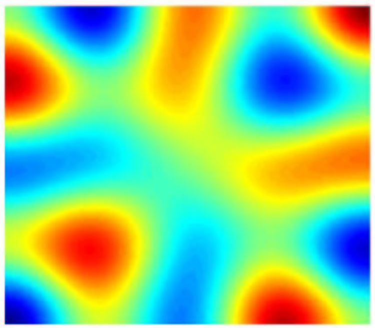
$\omega^2 = 39.48$



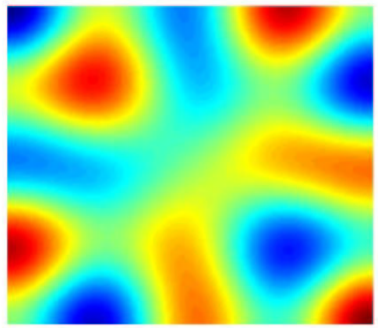
$\omega^2 = 39.48$



$\omega^2 = 41.95$

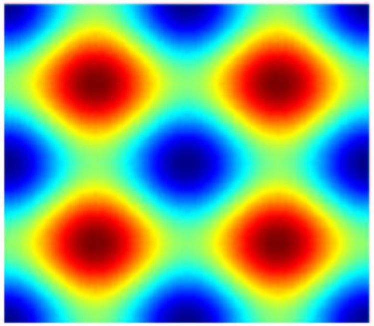


$\omega^2 = 41.95$

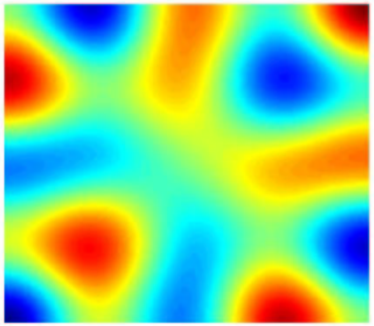


Eigenmodes in DG method (2-D)

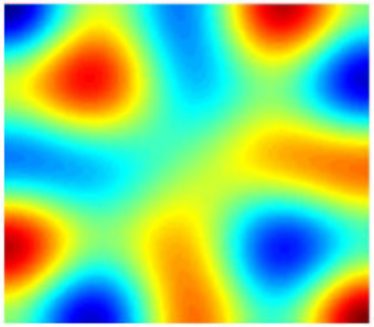
$\omega^2 = 39.48$



$\omega^2 = 41.95$



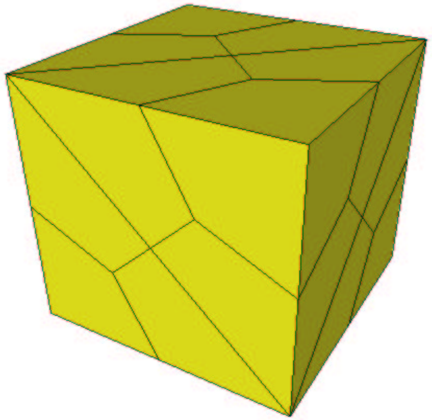
$\omega^2 = 41.95$



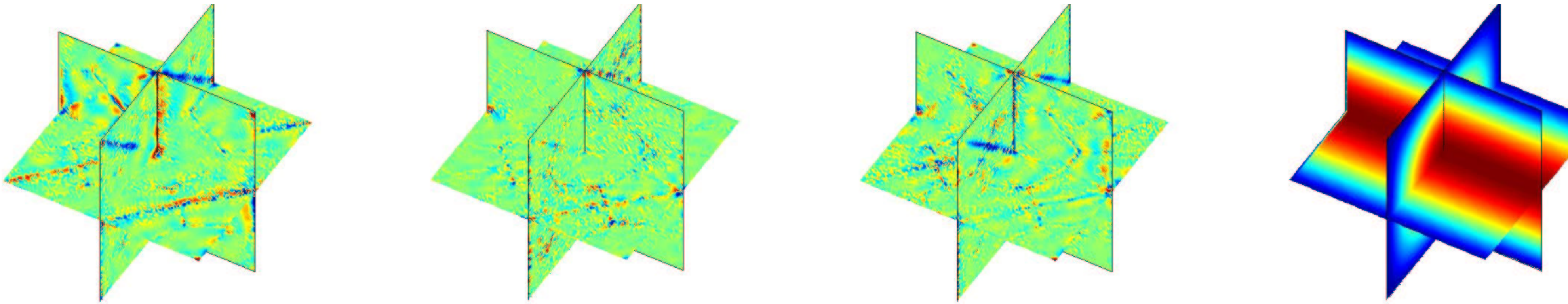
- “Constant” number of spurious modes on regular meshes
- “Decreasing” number of spurious on split meshes

Eigenmodes in DG method (3-D)

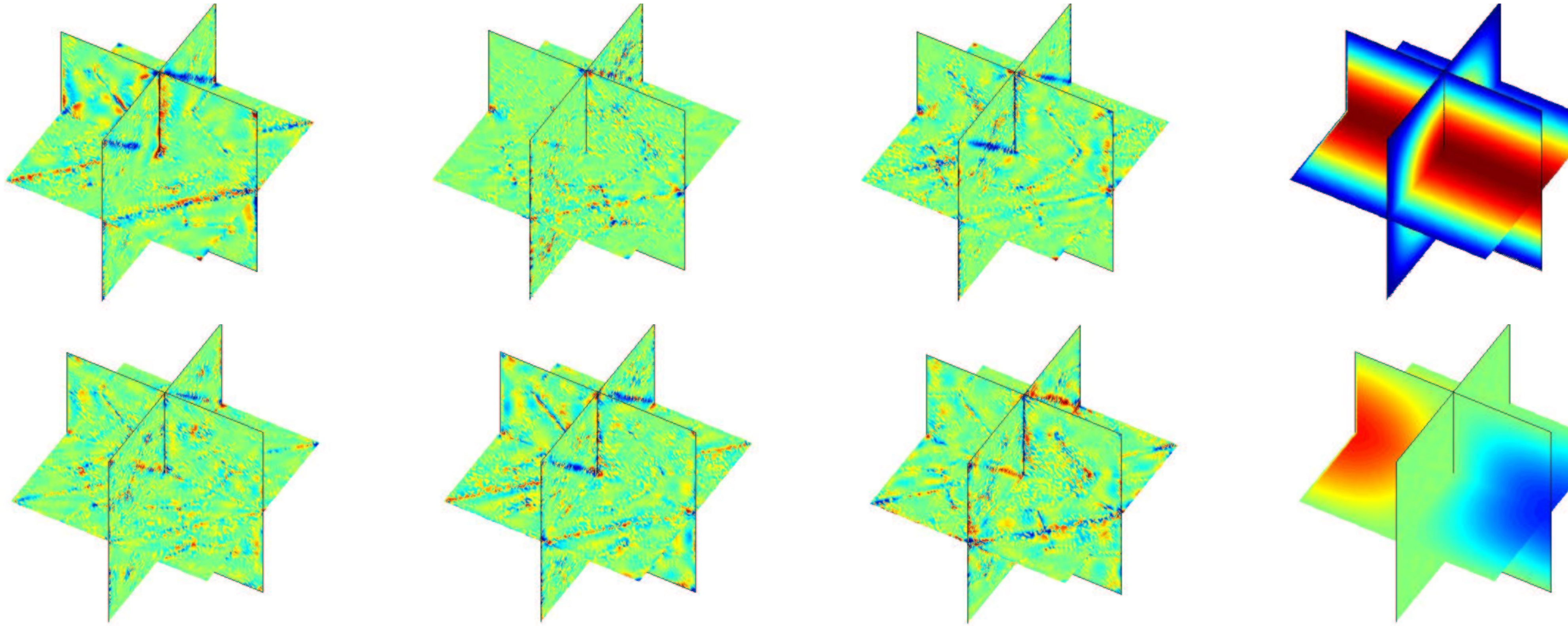
Mesh used for the simulations ($\mathbf{Q}_4$)



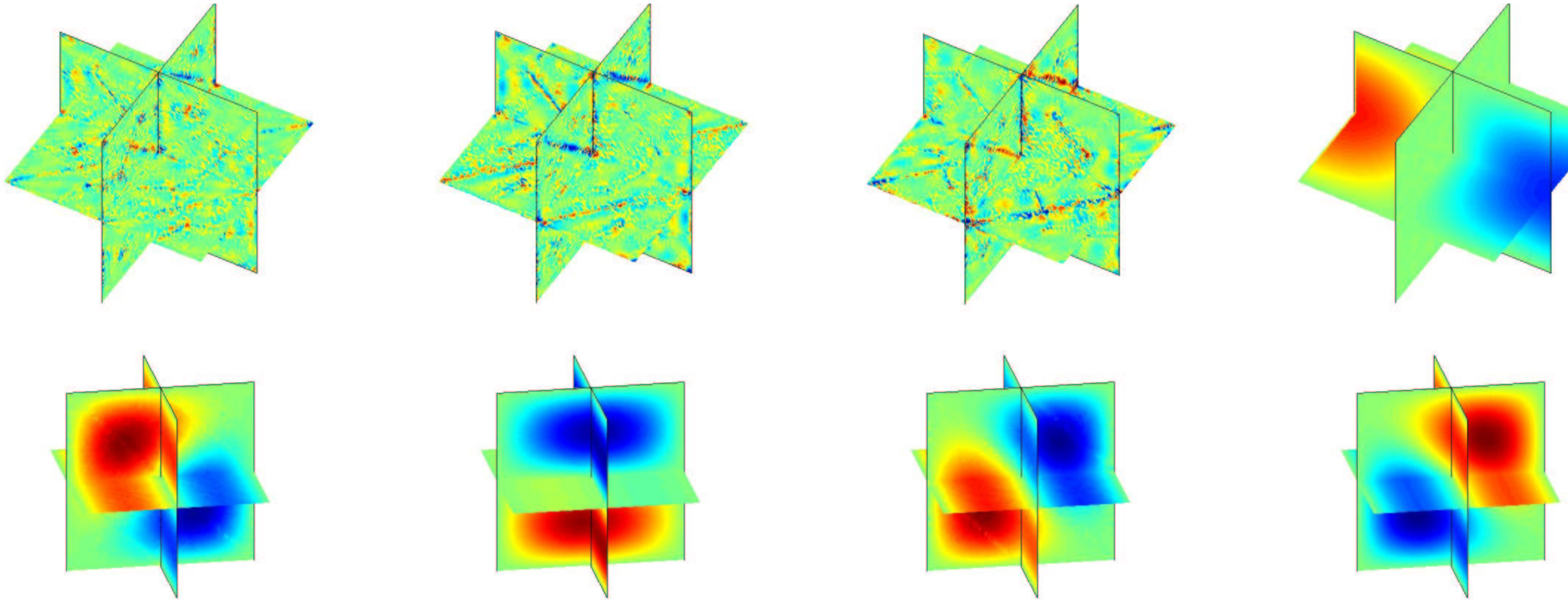
Eigenmodes in DG method (3-D)



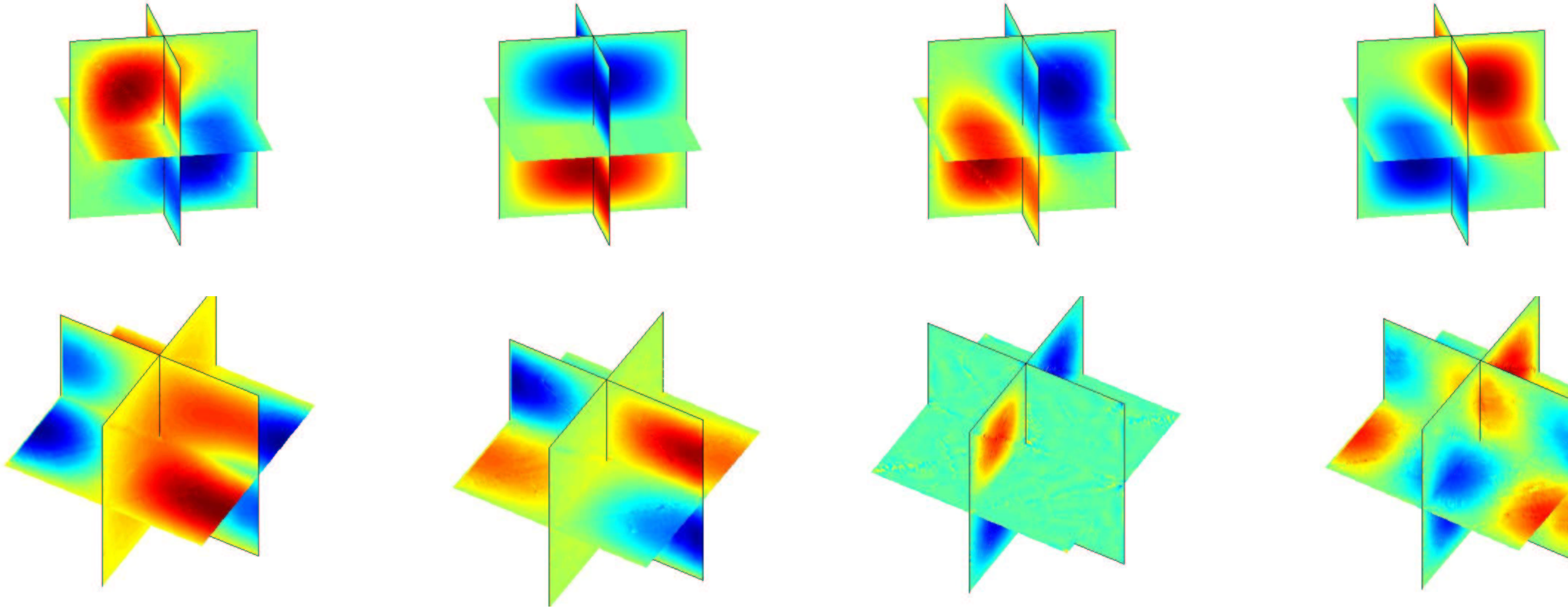
Eigenmodes in DG method (3-D)



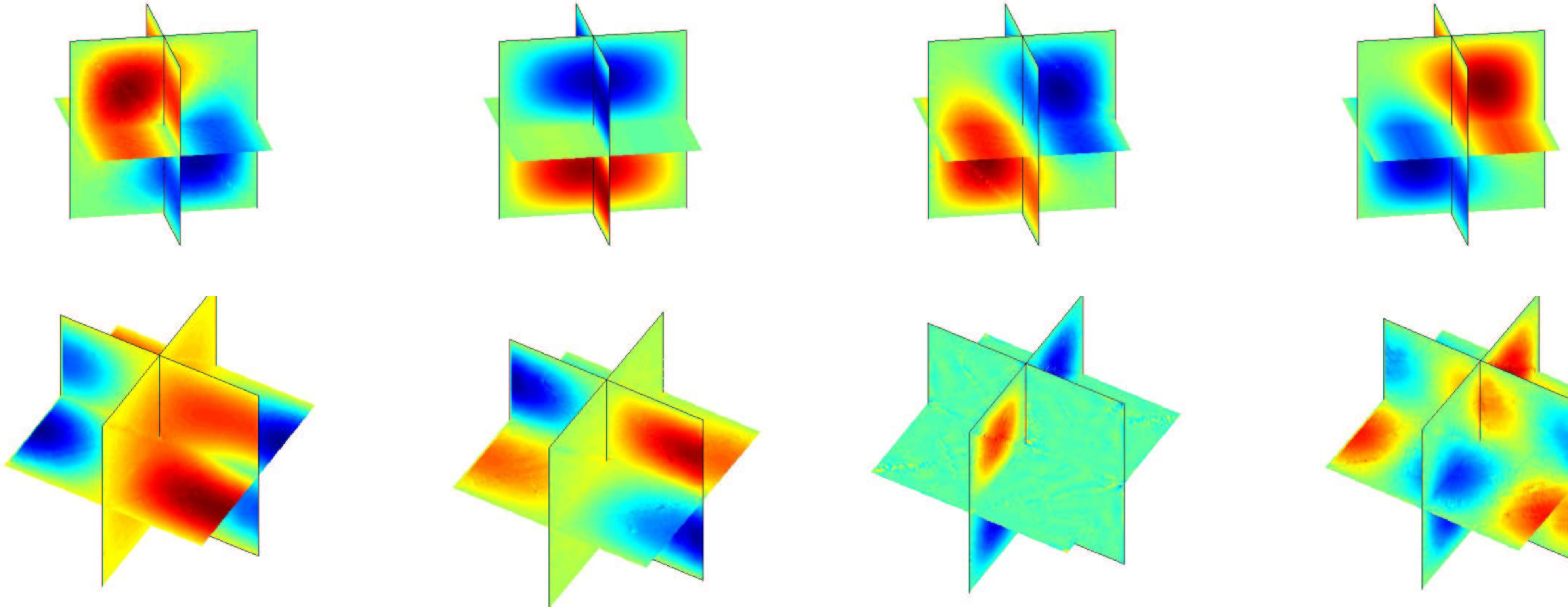
Eigenmodes in DG method (3-D)



Eigenmodes in DG method (3-D)



Eigenmodes in DG method (3-D)



- Increasing number of spurious modes

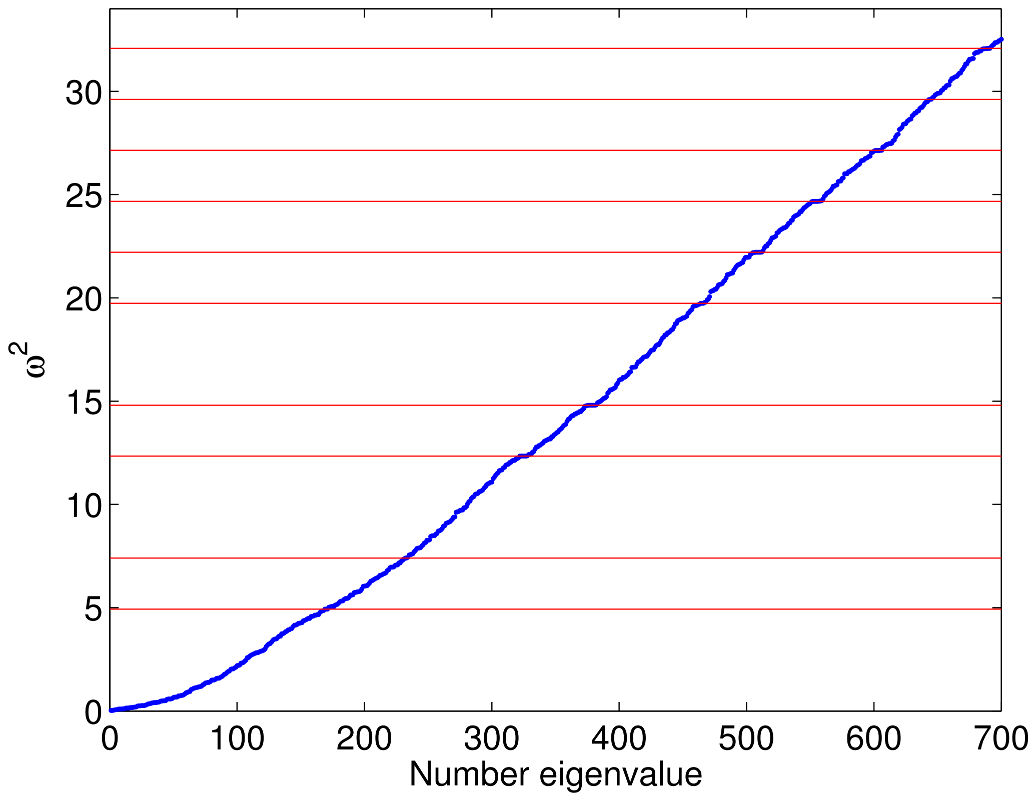
Penalization terms, eigenvalues

To the first equation in E, we add :

$$-i\omega\,\alpha \int_{\partial K_i} [\mathbf{E} \times \mathbf{n}] \cdot \boldsymbol{\varphi} \times \mathbf{n}\, dx$$

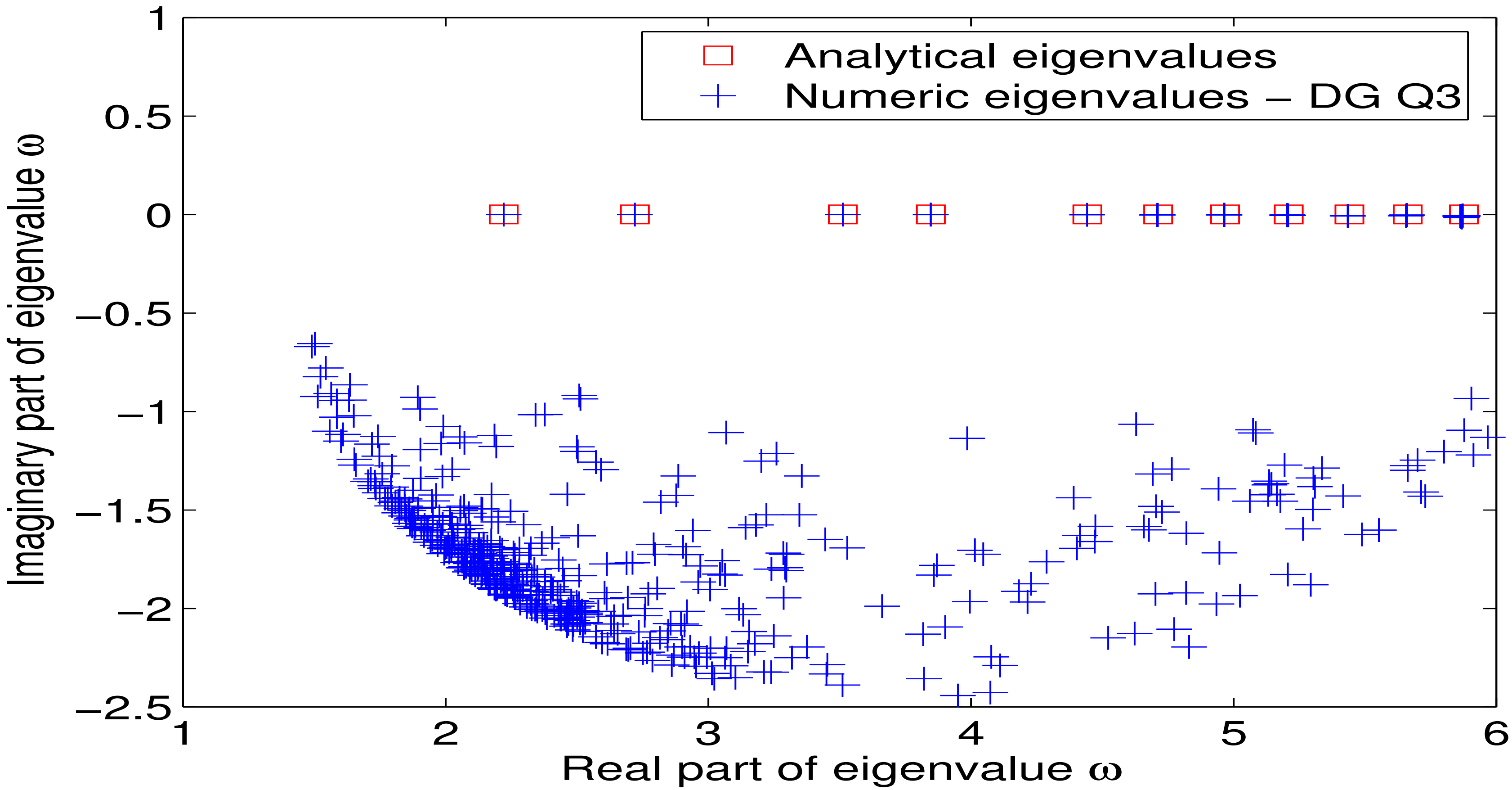
We take $\alpha = 0.5$

Penalization terms, eigenvalues



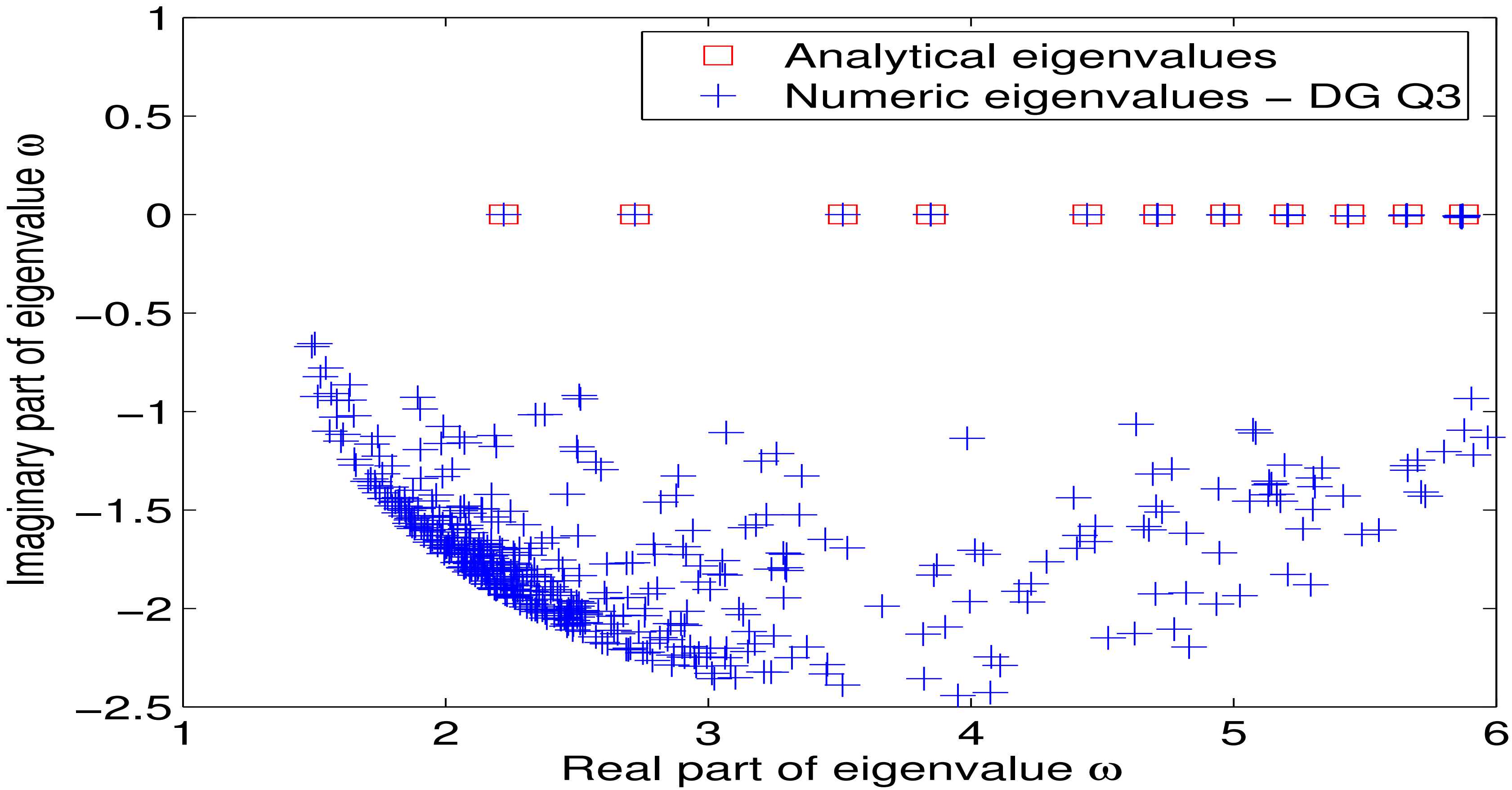
Eigenvalues, if no penalization is used $\alpha = 0$
Blue points are numeric eigenvalues, red lines analytic eigenvalues.

Penalization terms, eigenvalues



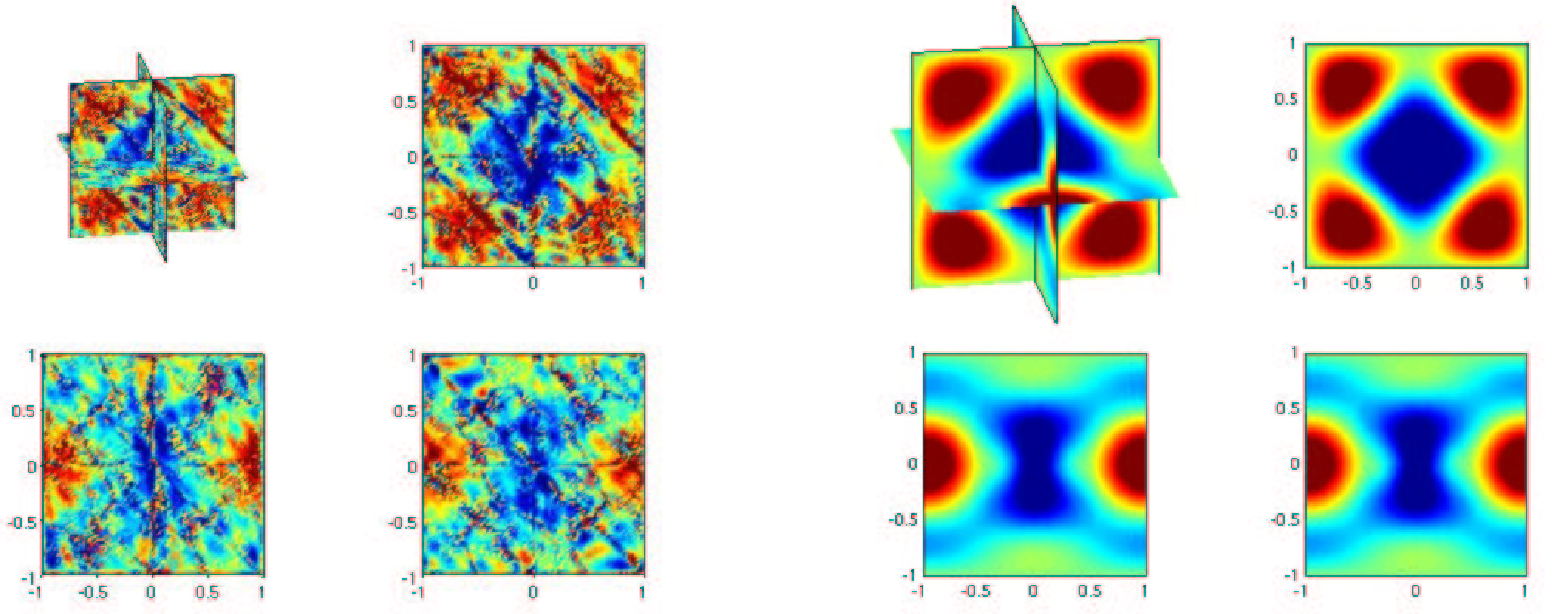
Eigenvalues if penalization is used $\alpha = 0.5$
Blue points are numeric eigenvalues, red squares analytic eigenvalues.

Penalization terms, eigenvalues



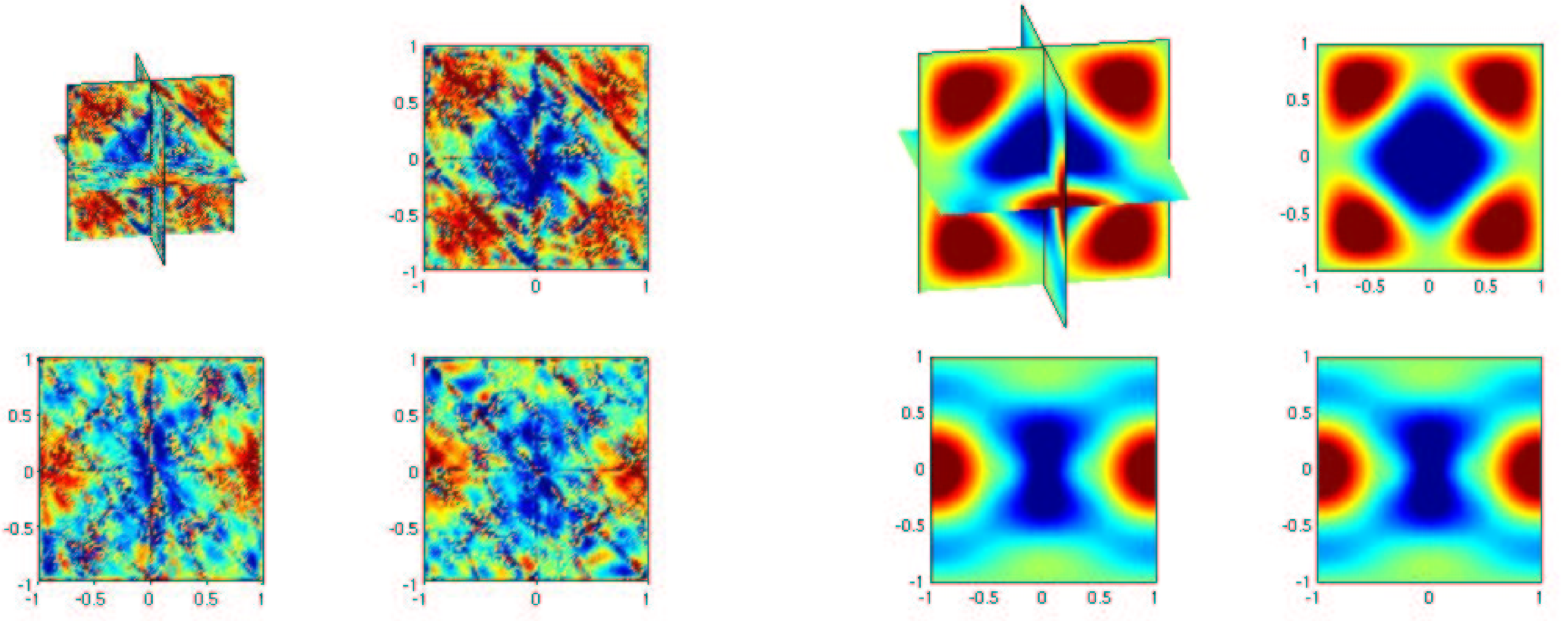
- Penalization terms reject ALL spurious modes in complex plane

Effects of penalization



At left, numerical solution with $\alpha = 0$, at right with $\alpha = 0.5$

Effects of penalization



- Fine solution on split meshes
- Negligible overcost in computational time

Nedelec's first family on hexahedra

Space of approximation

$$V_h = \{ \vec{u} \in \text{H(curl,}\Omega\text{)} \text{ so that } DF_i^t \, \vec{u} \circ F_i \in Q_{r-1,r,r} \times Q_{r,r-1,r} \times Q_{r,r,r-1} \}$$

Nedelec's first family on hexahedra

Space of approximation

$$V_h \;=\; \{\,\vec{u} \in \ \text{H(curl,}\Omega\text{)} \ \text{so that}\ DF_i^t\,\vec{u} \circ F_i \,\in Q_{r-1,r,r} \times Q_{r,r-1,r} \times Q_{r,r,r-1}\,\}$$

Basis functions

$$\hat{\vec{\varphi}}^1_{i,j,k}(\hat{x},\hat{y},\hat{z}) \;=\; \hat{\psi}^G_i(\hat{x})\,\hat{\psi}^{GL}_j(\hat{y})\,\hat{\psi}^{GL}_k(\hat{z})\,\vec{e_x} \quad 1 \le i \le r \quad 1 \le j,k \le r+1$$

$$\hat{\vec{\varphi}}^2_{j,i,k}(\hat{x},\hat{y},\hat{z}) \;=\; \hat{\psi}^{GL}_j(\hat{x})\,\hat{\psi}^G_i(\hat{y})\,\hat{\psi}^{GL}_k(\hat{z})\,\vec{e_y} \quad 1 \le i \le r \quad 1 \le j,k \le r+1$$

$$\hat{\vec{\varphi}}^3_{k,j,i}(\hat{x},\hat{y},\hat{z}) \;=\; \hat{\psi}^{GL}_k(\hat{x})\,\hat{\psi}^{GL}_j(\hat{y})\,\hat{\psi}^G_i(\hat{x})\,\vec{e_z} \quad 1 \le i \le r \quad 1 \le j,k \le r+1$$

Nedelec's first family on hexahedra

Space of approximation

$$V_h = \{ \vec{u} \in \text{H(curl,}\Omega\text{)} \text{ so that } DF_i^t \, \vec{u} \circ F_i \in Q_{r-1,r,r} \times Q_{r,r-1,r} \times Q_{r,r,r-1} \}$$

Basis functions

$$\vec{\hat{\varphi}}^1_{i,j,k}(\hat{x},\hat{y},\hat{z}) = \hat{\psi}_i^G(\hat{x})\,\hat{\psi}_j^{GL}(\hat{y})\,\hat{\psi}_k^{GL}(\hat{z})\,\vec{e_x} \quad 1 \le i \le r \quad 1 \le j,k \le r+1$$

$$\vec{\hat{\varphi}}^2_{j,i,k}(\hat{x},\hat{y},\hat{z}) = \hat{\psi}_j^{GL}(\hat{x})\,\hat{\psi}_i^{G}(\hat{y})\,\hat{\psi}_k^{GL}(\hat{z})\,\vec{e_y} \quad 1 \le i \le r \quad 1 \le j,k \le r+1$$

$$\vec{\hat{\varphi}}^3_{k,j,i}(\hat{x},\hat{y},\hat{z}) = \hat{\psi}_k^{GL}(\hat{x})\,\hat{\psi}_j^{GL}(\hat{y})\,\hat{\psi}_i^{G}(\hat{x})\,\vec{e_z} \quad 1 \le i \le r \quad 1 \le j,k \le r+1$$

ψ_i^G, ψ_i^{GL} lagragian functions linked respectively with Gauss points and Gauss-Lobatto points.

See. G. Cohen, P. Monk, Gauss points mass lumping

Elementary matrices

Mass matrix :

$$(M_h)_{i,j} \;=\; \int_{\hat{K}} J_i \, DF_i^{-1} \, \varepsilon \, DF_i^{*-1} \, \hat{\varphi}_i \cdot \hat{\varphi}_k \, d\hat{x}$$

Stiffness matrix :

$$(K_h)_{i,j} \;=\; \int_{\hat{K}} \frac{1}{J_i} \, DF_i^t \, \mu^{-1} \, DF_i \, \hat{\nabla} \times \hat{\varphi}_i \cdot \hat{\nabla} \times \hat{\varphi}_k \, d\hat{x}$$

Elementary matrices

Mass matrix :

$$(M_h)_{i,j} \; = \; \int_{\hat{K}} J_i \, DF_i^{-1} \, \varepsilon \, DF_i^{*-1} \, \hat{\varphi}_i \cdot \hat{\varphi}_k \, d\hat{x}$$

Stiffness matrix :

$$(K_h)_{i,j} \; = \; \int_{\hat{K}} \frac{1}{J_i} \, DF_i^t \, \mu^{-1} \, DF_i \, \hat{\nabla} \times \hat{\varphi}_i \cdot \hat{\nabla} \times \hat{\varphi}_k \, d\hat{x}$$

- Use of Gauss-Lobatto quadrature (ω_k^{GL}, ξ_k^{GL})

Elementary matrices

Mass matrix :

$$(M_h)_{i,j} \;=\; \int_{\hat{K}} J_i\, DF_i^{-1}\, \varepsilon\, DF_i^{*-1}\, \hat{\varphi}_i \cdot \hat{\varphi}_k\, d\hat{x}$$

Stiffness matrix :

$$(K_h)_{i,j} \;=\; \int_{\hat{K}} \frac{1}{J_i}\, DF_i^t\, \mu^{-1}\, DF_i\, \hat{\nabla} \times \hat{\varphi}_i \cdot \hat{\nabla} \times \hat{\varphi}_k\, d\hat{x}$$

- Use of Gauss-Lobatto quadrature (ω_k^{GL}, ξ_k^{GL})
- Block-diagonal matrix

$$(A_h)_{k,k} = \left[J_i\, DF_i^{-1}\, \varepsilon\, DF_i^{*-1} \right] (\xi_k^{GL}) \omega_k^{GL}$$

- Block-diagonal matrix

$$(B_h)_{k,k} = \left[\frac{1}{J_i}\, DF_i^t\ \ \mu^{-1}\, DF_i \right] (\xi_k^{GL}) \omega_k^{GL}$$

Fast matrix vector product

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} = \hat{\varphi}_i(\xi_j^{GL}) \qquad \hat{R}_{i,j} = \hat{\nabla} \times \hat{\varphi}_i^{GL}(\xi_j^{GL})$$

Fast matrix vector product

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} = \hat{\varphi}_i(\xi_j^{GL}) \qquad \hat{R}_{i,j} = \hat{\nabla} \times \hat{\varphi}_i^{GL}(\xi_j^{GL})$$

Then, we have : $M_h = \hat{C}\, A_h \hat{C}^* \qquad K_h = \hat{C}\hat{R}\, B_h \hat{R}^* \hat{C}^*$

Fast matrix vector product

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} \; = \; \hat{\varphi}_i(\xi_j^{GL}) \qquad \hat{R}_{i,j} \; = \; \hat{\nabla} \times \hat{\varphi}_i^{GL}(\xi_j^{GL})$$

Then, we have : $M_h \; = \; \hat{C}\, A_h \hat{C}^* \qquad K_h \; = \; \hat{C}\hat{R}\, B_h \hat{R}^* \hat{C}^*$

- Complexity of $\hat{C}\, U$: $6\,(r+1)^4$ operations in 3-D
- Complexity of $\hat{R}\, U$: $12\,(r+1)^4$ operations in 3-D
- Complexity of $A_h\, U + B_h\, U$: $30\,(r+1)^3$ operations

Complexity of standard matrix vector product $18 r^3\,(r+1)^3$

Fast matrix vector product

Let us introduce the two following matrices, independant of the geometry :

$$\hat{C}_{i,j} = \hat{\varphi}_i(\xi_j^{GL}) \qquad \hat{R}_{i,j} = \hat{\nabla} \times \hat{\varphi}_i^{GL}(\xi_j^{GL})$$

Then, we have : $M_h = \hat{C}\, A_h \hat{C}^* \qquad K_h = \hat{C}\hat{R}\, B_h \hat{R}^* \hat{C}^*$

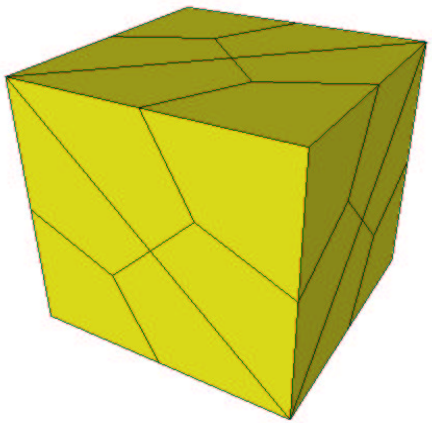
- Complexity of $\hat{C}\, U$: $6\,(r+1)^4$ operations in 3-D
- Complexity of $\hat{R}\, U$: $12\,(r+1)^4$ operations in 3-D
- Complexity of $A_h\, U + B_h\, U$: $30\,(r+1)^3$ operations

Complexity of standard matrix vector product $18r^3\,(r+1)^3$

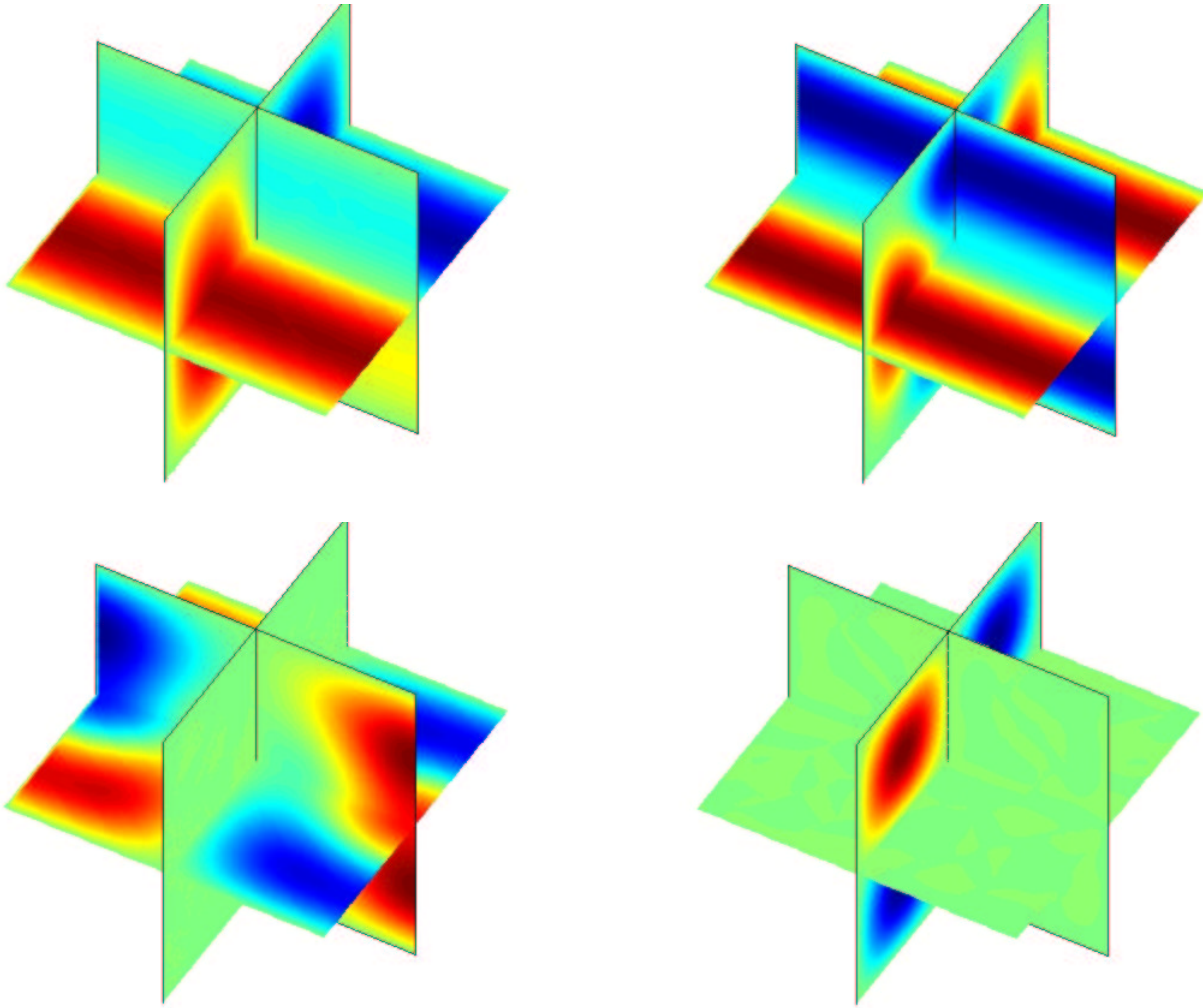
- Matrix-vector product 67% slower by using exact integration

Spurious free method

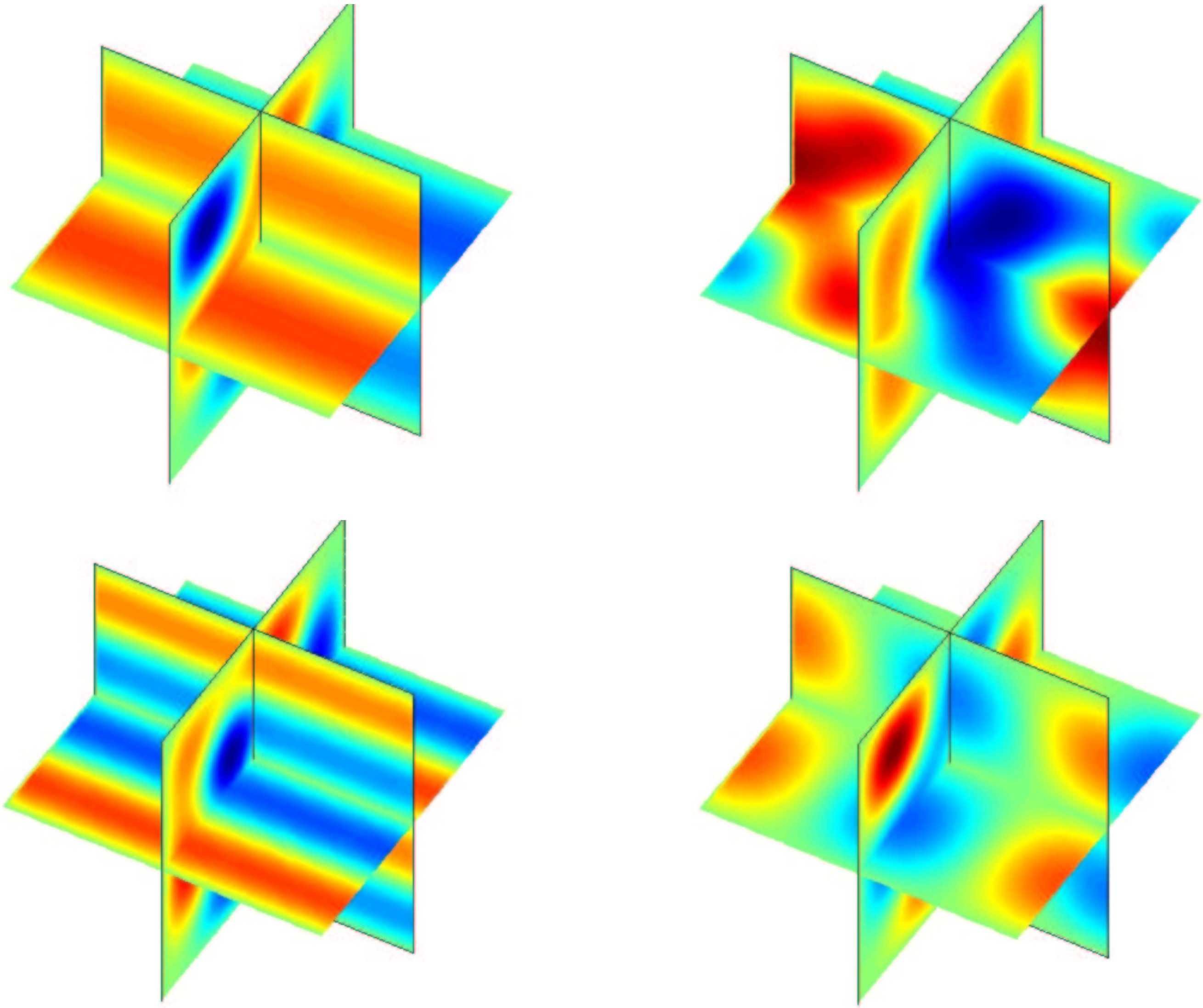
Mesh used for the simulations ($\mathbf{Q_5}$)



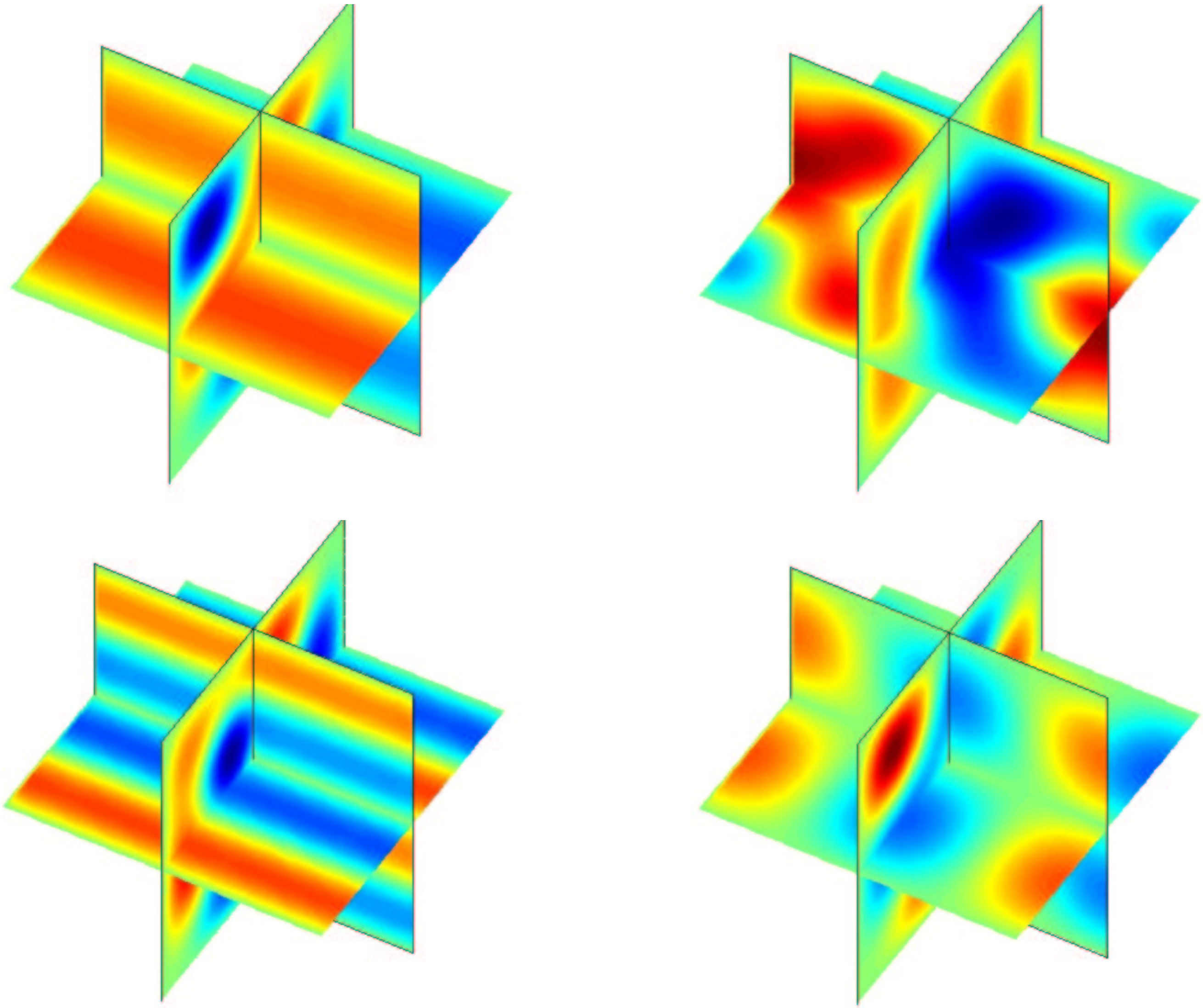
Spurious free method



Spurious free method



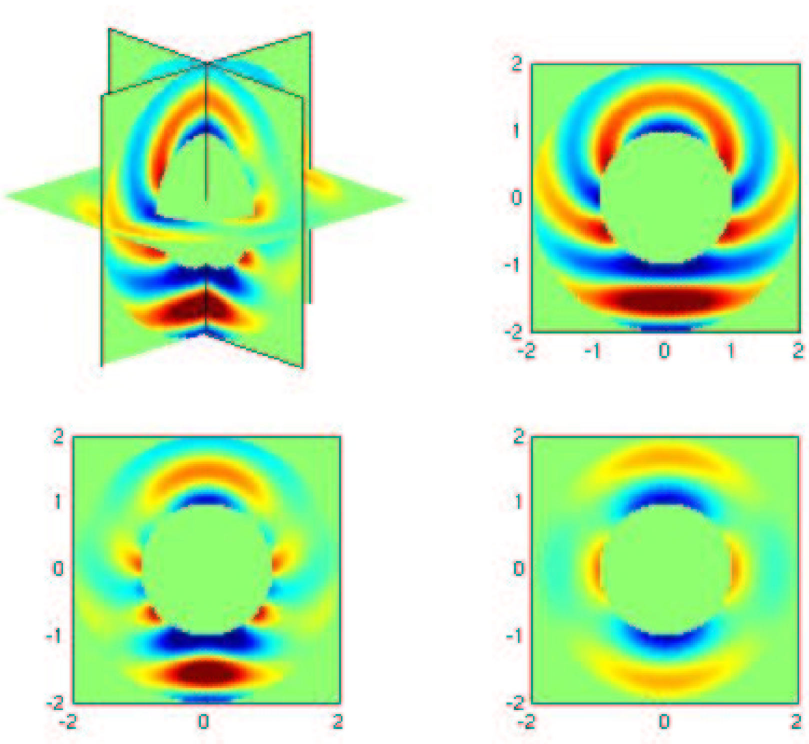
Spurious free method

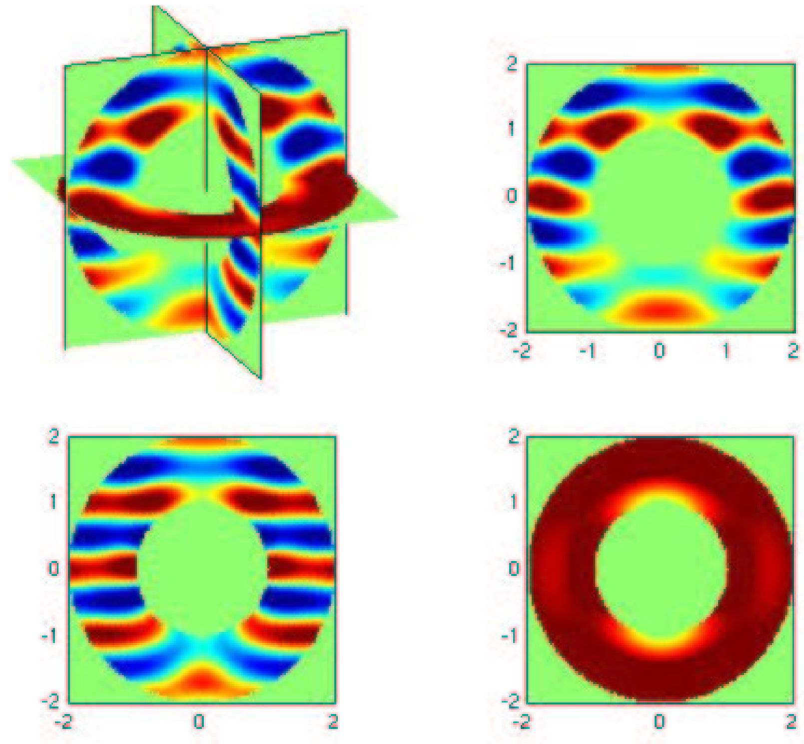


- Approximate integration leads to a spurious-free method

Convergence of the method

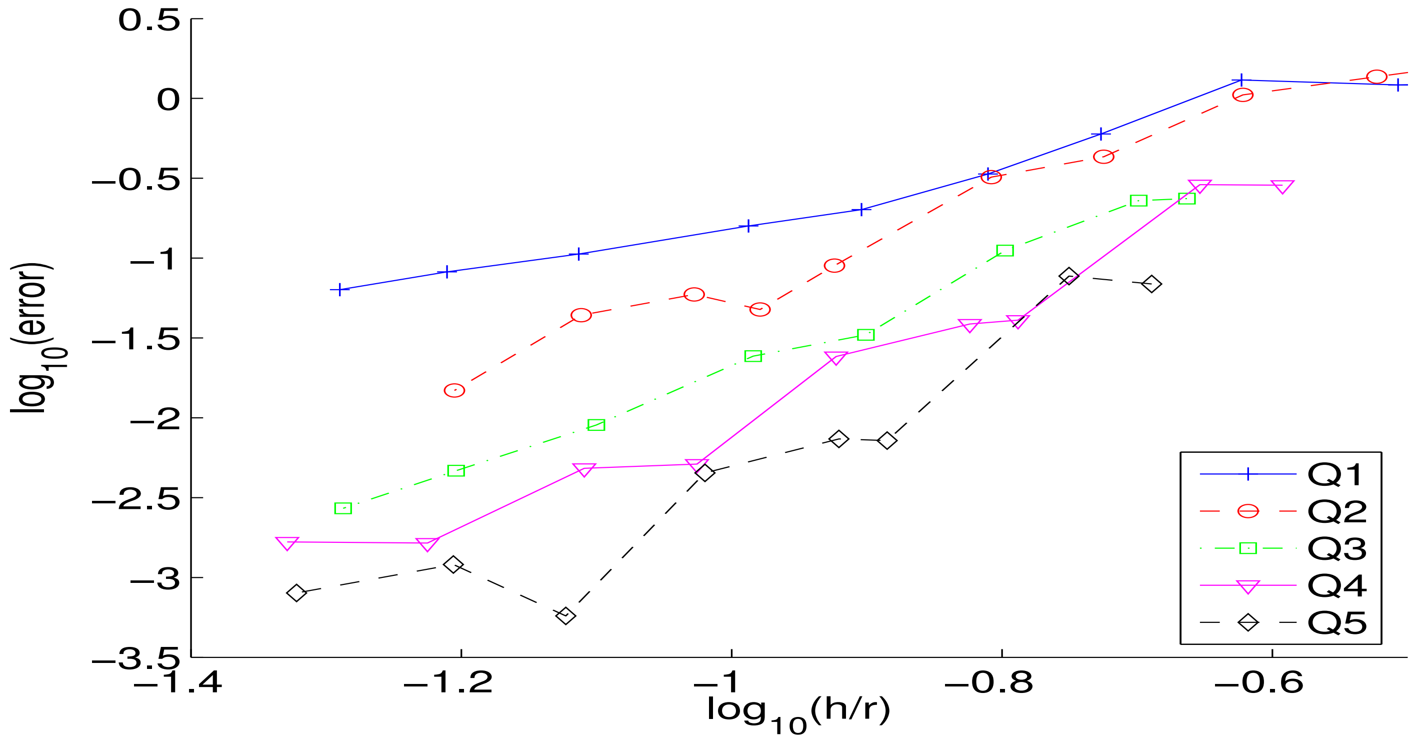
Scattering by a perfectly conductor sphere $E \times n = 0$





Convergence of the method

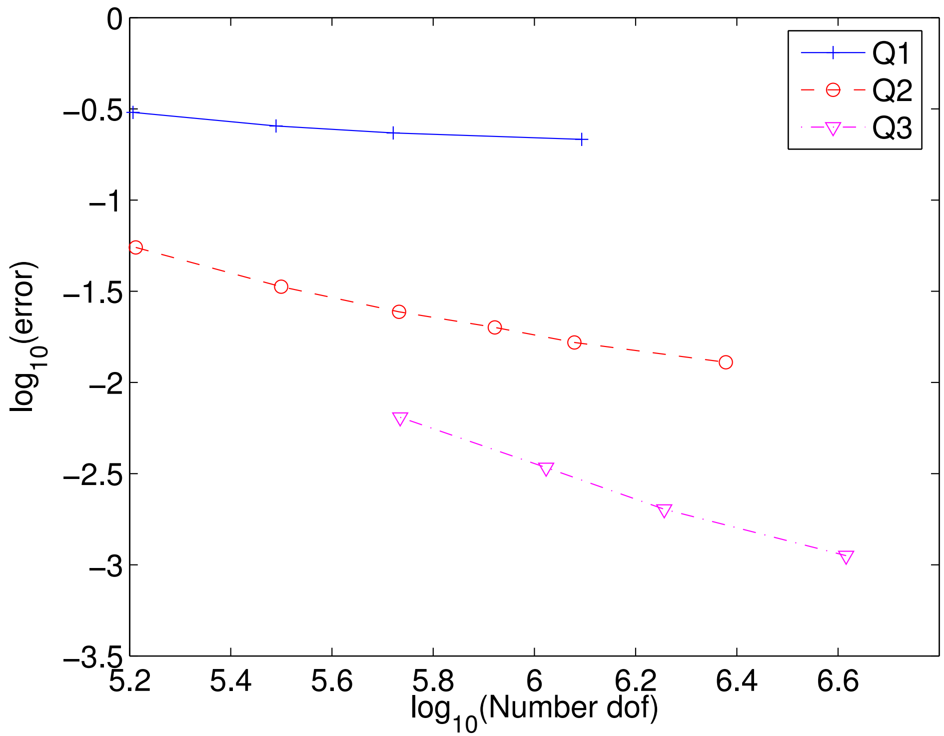
Convergence of Nedelec's first family on regular meshes



- Optimal convergence $O(h^r)$ in H(curl,Ω) norm

Convergence of the method

Convergence on tetrahedral meshes split in hexahedra



- Loss of one order, convergence $O(h^{r-1})$ in H(curl,Ω) norm

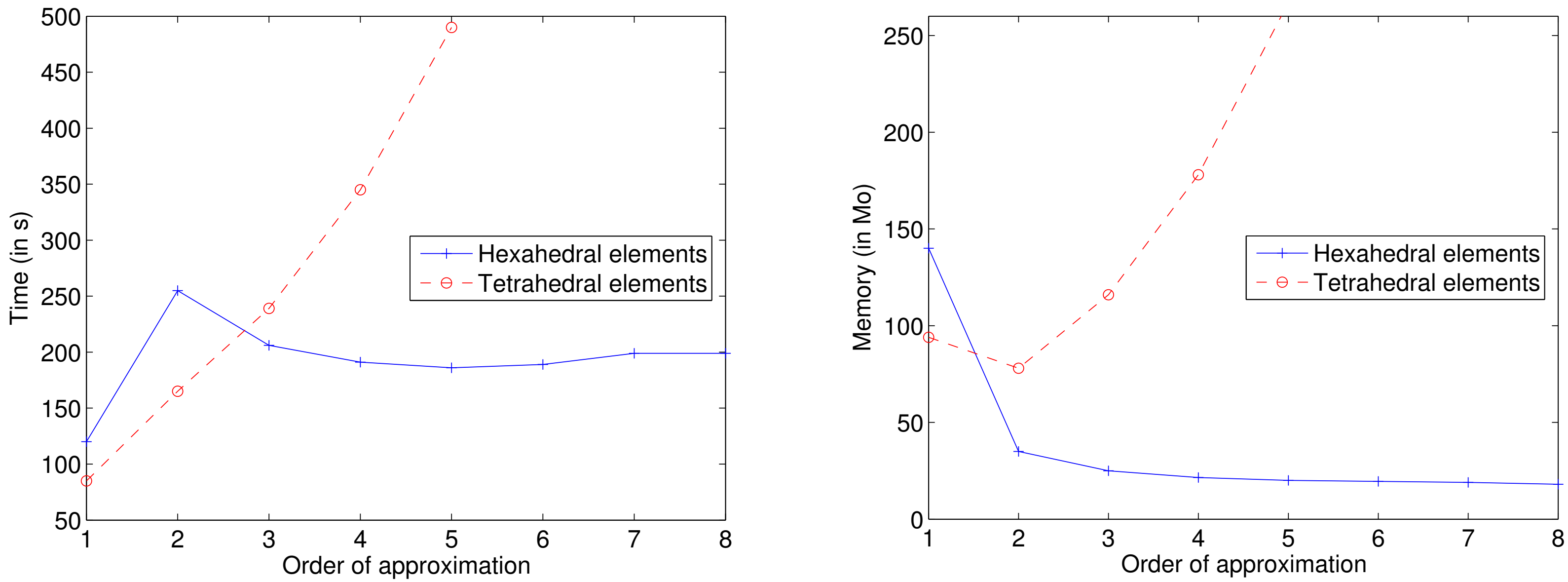
Is the matrix-vector product fast ?

Comparison between standard formulation and discrete factorization

Order	1	2	3	4	5
Time, standard formulation	55s	127s	224s	380s	631
Time, discrete factorization	244s	128s	106s	97s	96s
Storage, standard formulation	18 Mo	50 Mo	105 Mo	187 Mo	308 Mo
Storage, discrete factorization	23 Mo	9.9 Mo	6.9 Mo	5.7 Mo	5.0 Mo

Is the matrix-vector product fast ?

Comparison between tetrahedral and hexahedral elements



At left, time computation for a thousand iterations of COCG

At right, storage for mesh and matrices

Comparison DG method vs first family

- Both methods are spectrally correct
- Both methods have a fast MV product

Comparison DG method vs first family

- Both methods are spectrally correct
- Both methods have a fast MV product

Comparison DG method vs first family

- Both methods are spectrally correct
- Both methods have a fast MV product

Preconditioning used

- Incomplete factorization with threshold on the damped Maxwell equation :

$$-k^2(\alpha + i\beta)\varepsilon E - \nabla \times (\frac{1}{\mu}\nabla \times E) = 0$$

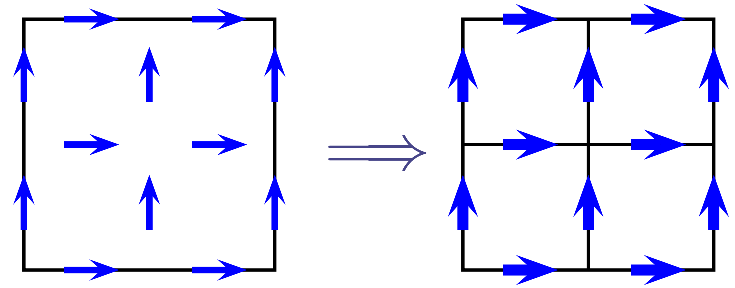
  - ILUT threshold ≥ 0.05 in order to have a low storage

Preconditioning used

- Incomplete factorization with threshold on the damped Maxwell equation :

$$-k^2(\alpha + i\beta)\varepsilon\, E \;-\; \nabla \times (\frac{1}{\mu}\nabla \times E) \;=\; 0$$

  - ILUT threshold ≥ 0.05 in order to have a low storage
  - Use of a Q_1 subdivided mesh to compute matrix



Preconditioning used

- Incomplete factorization with threshold on the damped Maxwell equation :

$$-k^2(\alpha + i\beta)\varepsilon E - \nabla \times (\frac{1}{\mu}\nabla \times E) = 0$$

- Multigrid method on the damped Maxwell equation
  - Use of the Q_1 mesh to do the multigrid iteration

Preconditioning used

- Incomplete factorization with threshold on the damped Maxwell equation :

$$-k^2(\alpha + i\beta)\varepsilon E - \nabla \times (\frac{1}{\mu}\nabla \times E) = 0$$

- Multigrid method on the damped Maxwell equation
  - Use of the Q_1 mesh to do the multigrid iteration
- Without damping, both preconditioners **doesn't lead** to convergence.
- A good choice of parameter is $\alpha = 0.7,\ \beta = 0.35$

Transparent condition

Silver-Muller condition is a first-order ABC :

$$E \times n + n \times H \times n = 0$$

Transparent condition

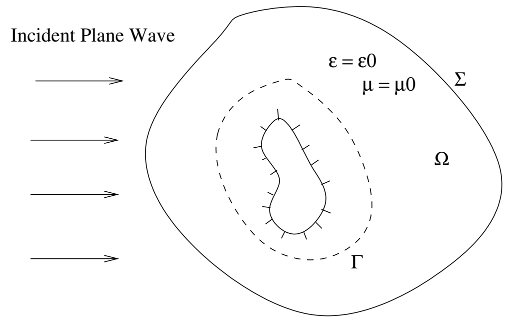
Silver-Muller condition is a first-order ABC :

$$E \times n + n \times H \times n = 0$$

- Use of a transparent condition based on integral representation formulas :

$$E^{pot}(x) = \int_\Gamma ik\left(G(x,y) + \frac{1}{k^2}\nabla_y \nabla_y G(x,y)\right)(n \times H)(y)\, dy + \int_\Gamma (n \times E)(y) \times \nabla_y G(x,y)\, d$$

new boundary condition $E \times n + n \times H \times n = E^{pot} \times n + n \times H^{pot} \times n$



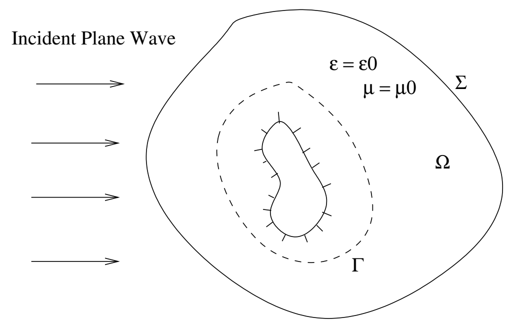
- **Needs of a virtual boundary Γ**
- **GMRES iterations to solve linear system**

Transparent condition

Silver-Muller condition is a first-order ABC :

$$\color{red}{E} \times n + n \times H \times n = 0$$

- Use of a transparent condition based on integral representation formulas :



- Needs of a virtual boundary Γ
- GMRES iterations to solve linear system
- C. Hazard, M. Lenoir, On the solution of time-harmonic scattering problems for Maxwell's equations

Radar cross section

Computation of far field of the electromagnetic objects by the formula

$$\sigma(\mathbf{u}) \;=\; \frac{k^2}{4\,\pi} \int_{\Sigma} e^{ik\mathbf{u}\cdot\mathbf{OM}} \left[\,\mathbf{u} \times (\mathbf{n} \times \mathbf{H}) \;+\; (u \otimes u \;-\; I)(\mathbf{E} \times \mathbf{n})\right] dM$$

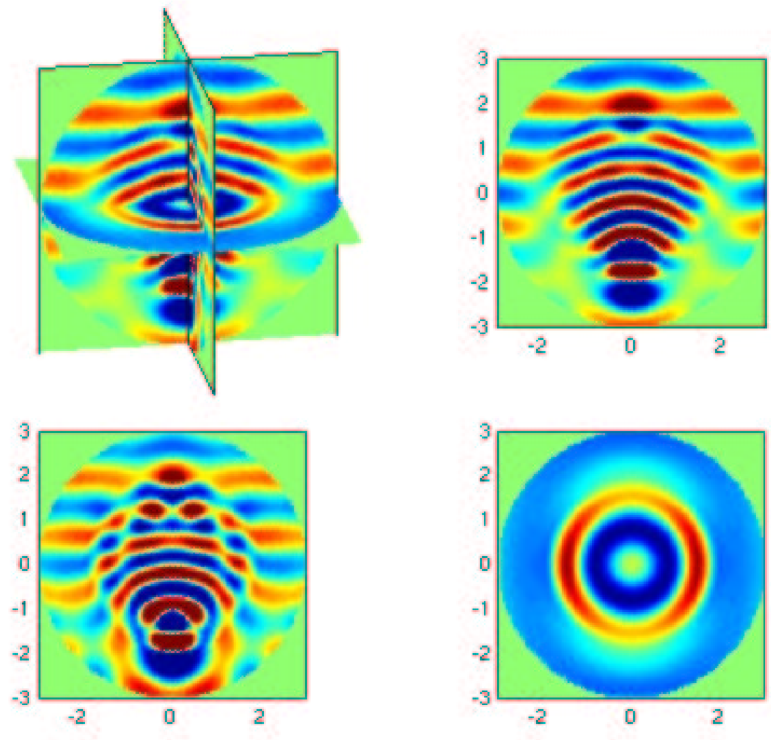
Radar cross section

Computation of far field of the electromagnetic objects by the formula

$$\sigma(\mathbf{u}) \; = \; \frac{k^2}{4\,\pi} \int_{\Sigma} e^{ik\mathbf{u}\cdot\mathbf{OM}} \left[\mathbf{u} \times (\mathbf{n} \times \mathbf{H}) \; + \; (u \otimes u \; - \; I)(\mathbf{E} \times \mathbf{n}) \right] dM$$

- Bistatic RCS : the vector of observation $\mathbf{u}$ varies
- Monostatic RCS : the wave vector $\mathbf{k}$ varies and $\mathbf{u} \; = \; \mathbf{k}$

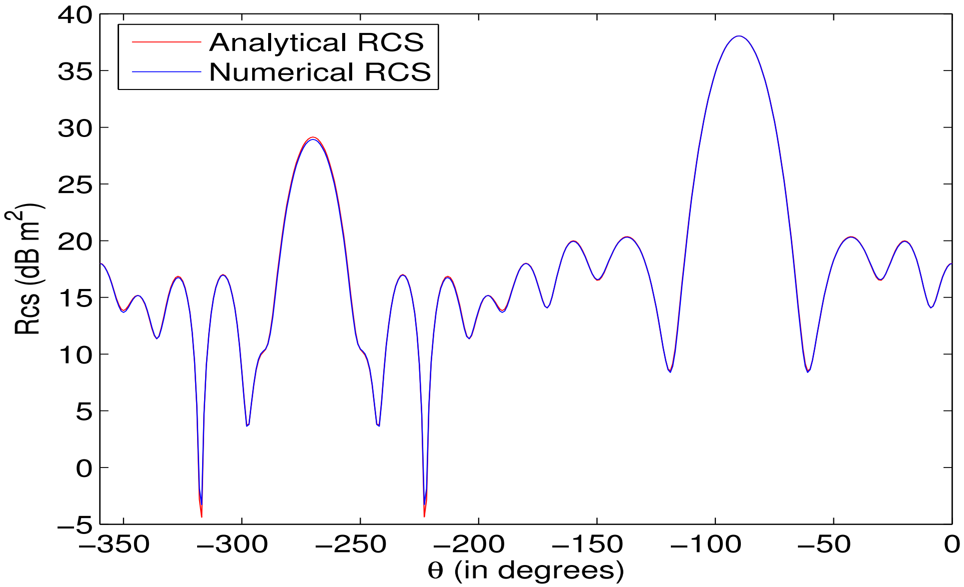
Scattering by a dielectric sphere



- Sphere of radius 2 with $\varepsilon = 3.5$ $\mu = 1$
- Outside boundary on a sphere of radius 3.

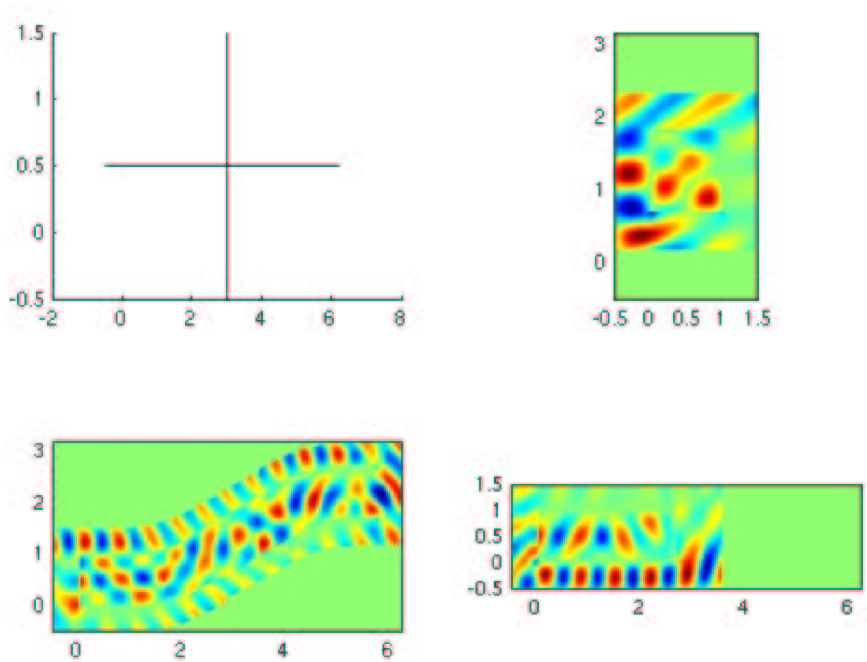
Scattering by a dielectric sphere

How many dofs/time to reach an error less than 0.5 dB



Finite Element	Q_2	Q_4	Q_6	Q_8
Nb dofs	940 000	88 000	230 000	88 000
No preconditioning	19 486 s	894 s	4 401 s	1 484 s
ILUT(0.05)	-	189 s	1 035 s	307 s
Two-grid	4 4344 s	488 s	1 095 s	952 s

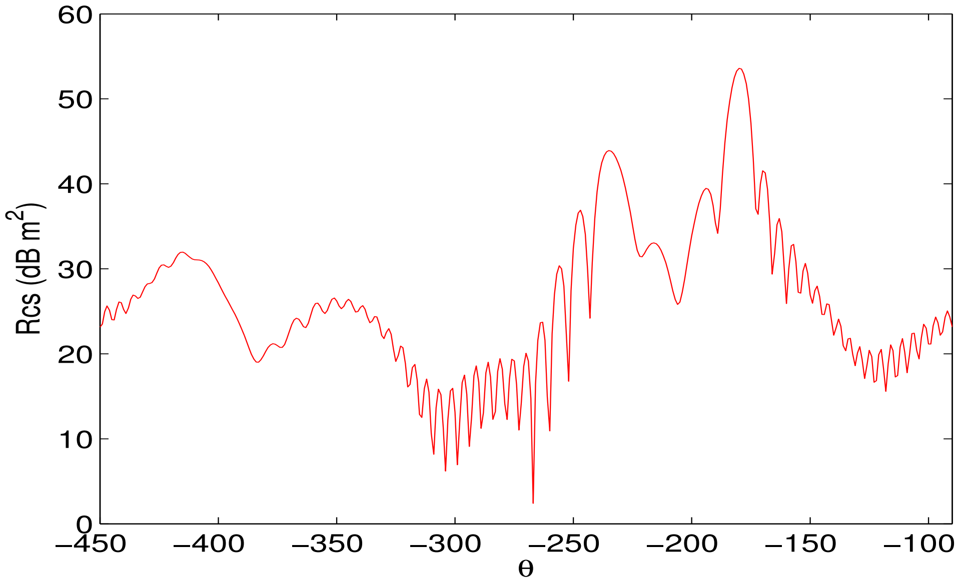
Scattering by a cobra cavity



- Cobra cavity of length 10, and depth 2
- Outside boundary at a distance of 1

Scattering by a cobra cavity

How many dofs/time to reach an error less than 0.5 dB



Finite Element	Q_4	Q_6
Nb dofs	412 000	187 000
No preconditioning	14 039 s	12 096 s
ILUT(0.05)	2 247 s	846 s
Two-grid	9 294 s	10 500 s

Outline

Resolution of Helmholtz equation

Resolution of time-harmonic Maxwell equations

Maxwell equations in axisymmetric domains

Maxwell equations in axisymmetric domains

Polar coordinates (r, θ, z) and associated vectors $(\hat{\mathbf{r}}, \hat{\boldsymbol{\theta}}, \hat{\mathbf{z}})$
Fourier decomposition of electric and magnetic field :

$$\mathbf{E} = \sum_{m=-\infty}^{+\infty} \left| \begin{array}{l} E_{r,m} \\ E_{\theta,m} \\ E_{z,m} \end{array} \right. e^{-im\theta} \qquad \mathbf{H} = \sum_{m=-\infty}^{+\infty} \left| \begin{array}{l} H_{r,m} \\ H_{\theta,m} \\ H_{z,m} \end{array} \right. e^{-im\theta}$$

Four unknowns : $\mathbf{E} = (E_r, E_z), \; E_\theta, \; \mathbf{H}, \; H_\theta$

Maxwell equations in axisymmetric domains

Polar coordinates (r, θ, z) and associated vectors $(\hat{\mathbf{r}}, \hat{\boldsymbol{\theta}}, \hat{\mathbf{z}})$
Four unknowns : $\mathbf{E} = (E_r, E_z),\ E_\theta,\ \mathbf{H},\ H_\theta$
Independant equations for each mode m

$$
\begin{aligned}
-\omega^2 \varepsilon \mathbf{E} + \frac{m}{r} \tilde{\mathbf{H}} - \frac{1}{r} \mathrm{rot}(r H_\theta) &= 0 \\
\mu \mathbf{H} + \frac{m}{r} \tilde{\mathbf{E}} - \frac{1}{r} \mathrm{rot}(r E_\theta) &= 0 \\
-\omega^2 \varepsilon E_\theta + \mathrm{rot}\, \mathbf{H} &= 0 \\
\mu H_\theta + \mathrm{rot}\, \mathbf{E} &= 0
\end{aligned}
$$

Discretization

- Mixed formulation with $\mathbf{E}, E_\theta \in$ H(curl,Ω) $\times\ H^1(\Omega)$
  $\mathbf{H}, H_\theta \in (L^2(\Omega))^3$
- High-order edge elements and nodal elements

Discretization

- Mixed formulation with $\mathbf{E}$, $E_\theta \in$ H(curl,Ω) $\times H^1(\Omega)$
  $\mathbf{H}$, $H_\theta \in (L^2(\Omega))^3$
- High-order edge elements and nodal elements
- Coupling with an integral equation at the boundary
- Additional unknown $J = n \times H$, is discretized in $H_1(\Gamma)$
- Numerical integration of singularities (Duffy, polar...)

Discretization

- Mixed formulation with $\mathbf{E}, E_\theta \in$ H(curl,Ω) $\times H^1(\Omega)$
  $\mathbf{H}, H_\theta \in (L^2(\Omega))^3$
- High-order edge elements and nodal elements
- Coupling with an integral equation at the boundary
- Additional unknown $J = n \times H$, is discretized in $H_1(\Gamma)$
- Numerical integration of singularities (Duffy, polar...)
- CFIE (Combined Field Integral Equation) formulation is used to avoid resonant problems
- Curved elements to have a good approximation of geometry

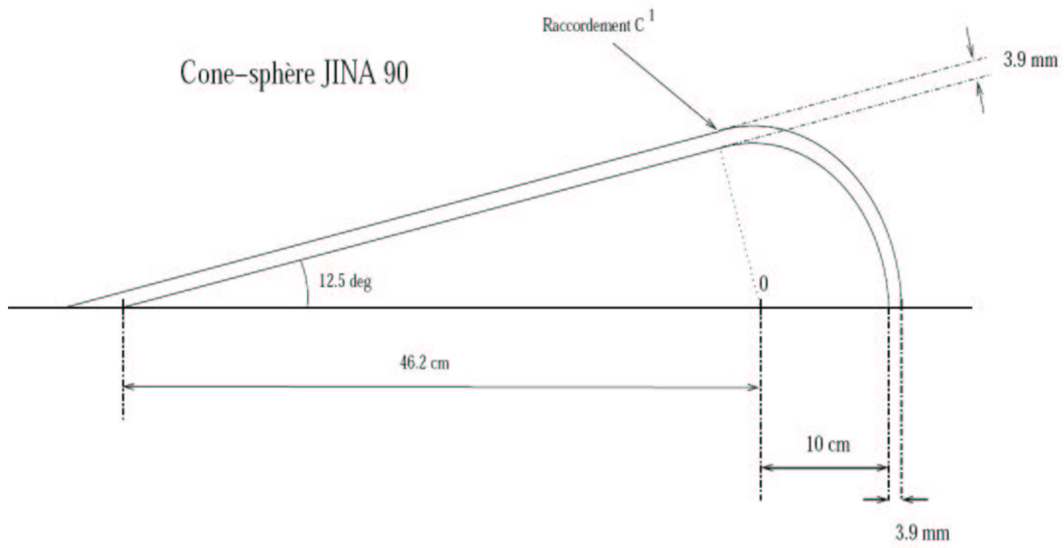
Perfectly conductor case

Only integral equations are used

Perfectly conductor case

Only integral equations are used

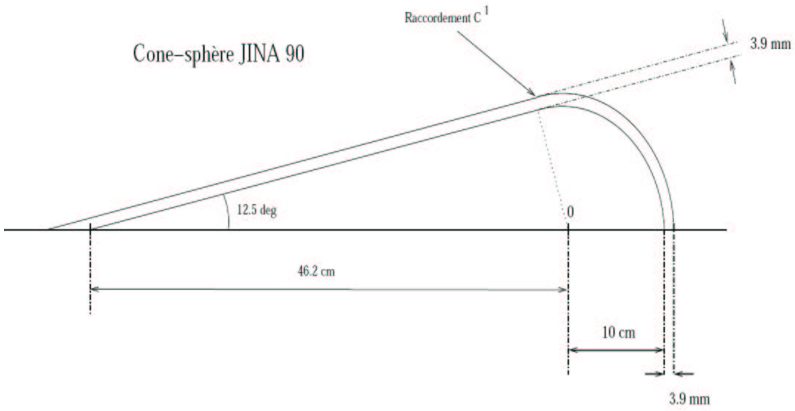
Cone-sphere case :



Perfectly conductor case

Only integral equations are used

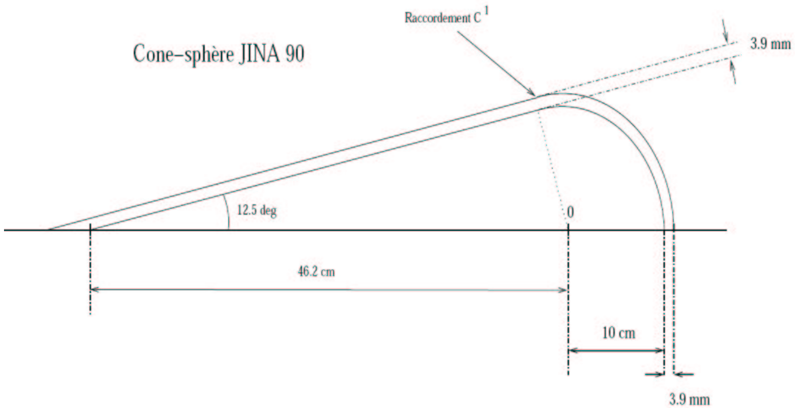
Monostatic RCS, polarization HH :



Perfectly conductor case

Only integral equations are used

Monostatic RCS, polarization HH :



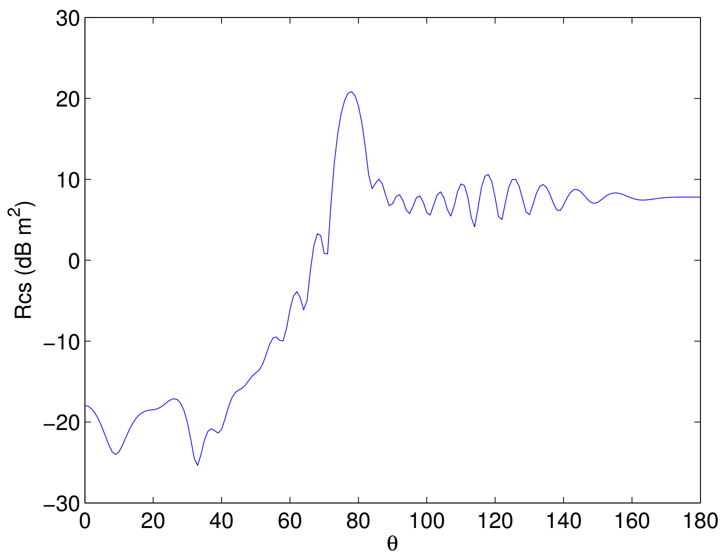
Number of dofs to reach error less than 0.5 dB

Order	Q_1	Q_2	Q_3	Q_4	Q_5
Dofs	120	66	62	74	72

Dielectric case

Cone-sphere case, with dielectric material :

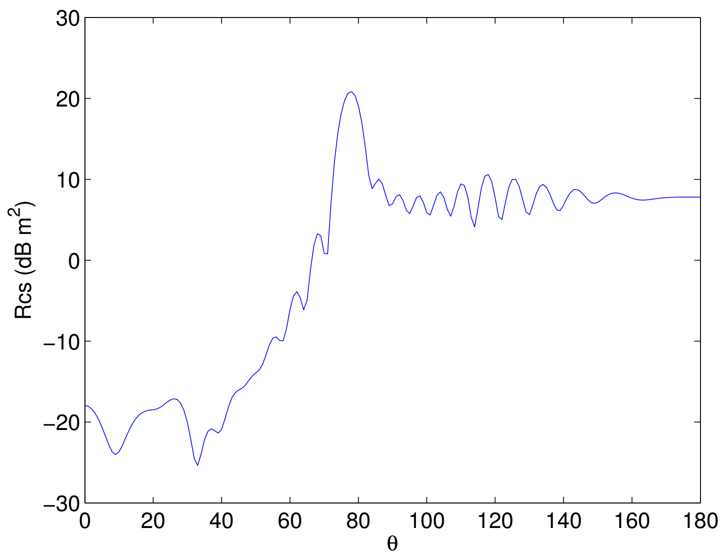
$$\varepsilon = 15 + 1.8i \qquad \mu = 1.7 + 1.7i$$



Dielectric case

Cone-sphere case, with dielectric material :

$$\varepsilon = 15 + 1.8i \qquad \mu = 1.7 + 1.7i$$



Number of dofs (for the unknown J) to reach error $\leq$ 0.5 dB :

Order	Q_1	Q_2	Q_3	Q_4	Q_5
Nb dofs	634	258	182	170	162

Prospects

- Improvement of preconditioning techniques for 3-D Maxwell's equations
- Coupling of integral equations and finite element in 3-D
  - Integration of singularities is more difficult than in axisymmetric case
  - Non-conformity of surfacic mesh and volumic mesh, in order to use different orders of approximation
- Coupling of 3-D solver and axisymmetric solver
- Coupling of Discontinuous Galerkin method with Nedelec's first family