Automatic hp-Adaptivity for Elliptic and Maxwell Problems

Leszek Demkowicz
Institute for Computational Engineering and Sciences (ICES)
The University of Texas at Austin
Austin, Texas 78712, U.S.A.
e-mail: leszek@ices.utexas.edu

Journées d'analyse fonctionnelle et numérique en l'honneur de
Michel Crouzeix, 2-3 June 2006

Outline of presentation

- Projection-based interpolation
- Algorithm underlying *hp*-adaptivity and hp-adaptivity for 3D acoustics scattering problems
- *hp*-adaptivity for Perfectly Matched Layers (PML)
- Goal-oriented *hp*-adaptivity with application to borehole electromagnetics simulations
- Conclusions and prospects

Projection-Based Interpolation

joint work with: A. Buffa, W. Cao, J. Gopalakrishnan, J. Schoeberl

H^1 Projection-Based Interpolation

Let K be a hexahedral element. Given a function $u \in H^{1.5+\epsilon}(K)$

H^1 Projection-Based Interpolation

Let K be a hexahedral element. Given a function $u \in H^{1.5+\epsilon}(K)$

- Construct the trilinear interpolant: $u_1(\boldsymbol{v}) = u(\boldsymbol{v}), \quad \forall \boldsymbol{v}$,

H^1 Projection-Based Interpolation

Let K be a hexahedral element. Given a function $u \in H^{1.5+\epsilon}(K)$

- Construct the trilinear interpolant: $u_1(\boldsymbol{v}) = u(\boldsymbol{v}), \quad \forall \boldsymbol{v}$,
- For each edge e, project difference $u - u_1$ onto the edge bubbles,

$$\|u - u_1 - u^e_{2,p}\|_{L^2(e)} \to \min$$

  Sum up the edge interpolants $u_{2,p} = \sum_e u^e_{2,p}$

H^1 Projection-Based Interpolation

Let K be a hexahedral element. Given a function $u \in H^{1.5+\epsilon}(K)$

- Construct the trilinear interpolant: $u_1(\boldsymbol{v}) = u(\boldsymbol{v}), \quad \forall \boldsymbol{v}$,
- For each edge e, project difference $u - u_1$ onto the edge bubbles,

$$\|u - u_1 - u^e_{2,p}\|_{L^2(e)} \to \min$$

  Sum up the edge interpolants $u_{2,p} = \sum_e u^e_{2,p}$

- For each face f, project difference $u - u_1 - u_{2,p}$ onto the face bubbles,

$$\|\boldsymbol{\nabla}_f(u - u_1 - u_{2,p} - u^f_{3,p})\|_{H^{-0.5}(f)} \to \min$$

  Sum up the face interpolants $u_{3,p} = \sum_f u^f_{3,p}$

H^1 Projection-Based Interpolation

Let K be a hexahedral element. Given a function $u \in H^{1.5+\epsilon}(K)$

- Construct the trilinear interpolant: $u_1(\boldsymbol{v}) = u(\boldsymbol{v}), \quad \forall \boldsymbol{v}$,
- For each edge e, project difference $u - u_1$ onto the edge bubbles,

$$\|u - u_1 - u^e_{2,p}\|_{L^2(e)} \to \min$$

  Sum up the edge interpolants $u_{2,p} = \sum_e u^e_{2,p}$
- For each face f, project difference $u - u_1 - u_{2,p}$ onto the face bubbles,

$$\|\boldsymbol{\nabla}_f(u - u_1 - u_{2,p} - u^f_{3,p})\|_{H^{-0.5}(f)} \to \min$$

  Sum up the face interpolants $u_{3,p} = \sum_f u^f_{3,p}$
- Project difference $u - u_1 - u_{2,p} - u_{3,p}$ onto the element bubbles,

$$\|\boldsymbol{\nabla}(u - u_1 - u_{2,p} - u_{3,p} - u_{4,p})\|_{L^2(K)} \to \min$$

H^1 Projection-Based Interpolation

Let K be a hexahedral element. Given a function $u \in H^{1.5+\epsilon}(K)$

- Construct the trilinear interpolant: $u_1(\boldsymbol{v}) = u(\boldsymbol{v}), \quad \forall \boldsymbol{v}$,
- For each edge e, project difference $u - u_1$ onto the edge bubbles,

$$\|u - u_1 - u^e_{2,p}\|_{L^2(e)} \rightarrow \min$$

  Sum up the edge interpolants $u_{2,p} = \sum_e u^e_{2,p}$
- For each face f, project difference $u - u_1 - u_{2,p}$ onto the face bubbles,

$$\|\boldsymbol{\nabla}_f (u - u_1 - u_{2,p} - u^f_{3,p})\|_{H^{-0.5}(f)} \rightarrow \min$$

  Sum up the face interpolants $u_{3,p} = \sum_f u^f_{3,p}$
- Project difference $u - u_1 - u_{2,p} - u_{3,p}$ onto the element bubbles,

$$\|\boldsymbol{\nabla}(u - u_1 - u_{2,p} - u_{3,p} - u_{4,p})\|_{L^2(K)} \rightarrow \min$$

- $\Pi^{grad} u = u_1 + u_{2,p} + u_{3,p} + u_{4,p}$.

H(curl) Projection-Based Interpolation

Given a function $\boldsymbol{E} \in \boldsymbol{H}^{0.5+\epsilon}(K) \cap \boldsymbol{H}^{\epsilon}(\mathbf{curl}, K)$

H(curl) Projection-Based Interpolation

Given a function $\boldsymbol{E} \in \boldsymbol{H}^{0.5+\epsilon}(K) \cap \boldsymbol{H}^{\epsilon}(\mathbf{curl}, K)$

- For each edge e,

H(curl) Projection-Based Interpolation

Given a function $\boldsymbol{E} \in \boldsymbol{H}^{0.5+\epsilon}(K) \cap \boldsymbol{H}^{\epsilon}(\mathbf{curl}, K)$

- For each edge e,
  - compute the edge average,

$$\int_e (E - E_0^e)_t = 0$$

H(curl) Projection-Based Interpolation

Given a function $\boldsymbol{E} \in \boldsymbol{H}^{0.5+\epsilon}(K) \cap \boldsymbol{H}^{\epsilon}(\mathbf{curl}, K)$

- For each edge e,
  - compute the edge average,

$$\int_e (E - E_0^e)_t = 0$$

  - compute the potential,

$$u(x) = \int_0^x (E - E_0^e)_t$$

H(curl) Projection-Based Interpolation

Given a function $\boldsymbol{E} \in \boldsymbol{H}^{0.5+\epsilon}(K) \cap \boldsymbol{H}^{\epsilon}(\mathbf{curl}, K)$

- For each edge e,
  - compute the edge average,

$$\int_e (E - E_0^e)_t = 0$$

  - compute the potential,

$$u(x) = \int_0^x (E - E_0^e)_t$$

  - project the potential on scalar bubbles,

$$\|u - u_p^e\|_{L^2(e)} \rightarrow \min$$

H(curl) Projection-Based Interpolation

Given a function $\boldsymbol{E} \in \boldsymbol{H}^{0.5+\epsilon}(K) \cap \boldsymbol{H}^{\epsilon}(\mathbf{curl}, K)$

- For each edge e,
  - compute the edge average,

$$\int_e (E - E_0^e)_t = 0$$

  - compute the potential,

$$u(x) = \int_0^x (E - E_0^e)_t$$

  - project the potential on scalar bubbles,

$$\|u - u_p^e\|_{L^2(e)} \rightarrow \min$$

  - differentiate the projection to complete the edge contribution,

$$\boldsymbol{E}_{1,p}^e = \frac{\partial u_p^e}{\partial t}$$

H(curl) Projection-Based Interpolation (cont.)

- Sum up the edge interpolants $\boldsymbol{E}_{1,p} = \sum_e \boldsymbol{E}_p^e$

H(curl) Projection-Based Interpolation (cont.)

- Sum up the edge interpolants $\boldsymbol{E}_{1,p} = \sum_e \boldsymbol{E}_p^e$
- For each face f, project difference $\boldsymbol{E} - \boldsymbol{E}_{e,1,p}$ onto the face bubbles,

$$\begin{cases} \|\mathrm{curl}_f(\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}^f)\|_{H^{-0.5}(f)} \to \min \\ (\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}^f, \boldsymbol{\nabla}_f \phi) = 0, \forall \text{ face bubble} \phi \end{cases}$$

Sum up the face interpolants $\boldsymbol{E}_{2,p} = \sum_f \boldsymbol{E}_{2,f}^e$

H(curl) Projection-Based Interpolation (cont.)

- Sum up the edge interpolants $\boldsymbol{E}_{1,p} = \sum_e \boldsymbol{E}_p^e$
- For each face f, project difference $\boldsymbol{E} - \boldsymbol{E}_{e,1,p}$ onto the face bubbles,

$$\begin{cases} \|\mathrm{curl}_f(\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}^f)\|_{H^{-0.5}(f)} \to \text{min} \\ (\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}^f, \boldsymbol{\nabla}_f \phi) = 0, \forall \text{ face bubble} \phi \end{cases}$$

Sum up the face interpolants $\boldsymbol{E}_{2,p} = \sum_f \boldsymbol{E}_{2,f}^e$

- Project difference $\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}$ onto the element bubbles,

$$\begin{cases} \|\boldsymbol{\nabla} \times (\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p} - \boldsymbol{E}_{3,p})\|_{L^2(K)} \to \text{min} \\ (\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p} - \boldsymbol{E}_{3,p}, \boldsymbol{\nabla} \phi)\|_{L^2(K)} \quad \forall \text{ element bubble } \phi \end{cases}$$

H(curl) Projection-Based Interpolation (cont.)

- Sum up the edge interpolants $\boldsymbol{E}_{1,p} = \sum_e \boldsymbol{E}_p^e$
- For each face f, project difference $\boldsymbol{E} - \boldsymbol{E}_{e,1,p}$ onto the face bubbles,

$$\begin{cases} \|\mathrm{curl}_f(\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}^f)\|_{H^{-0.5}(f)} \to \min \\ (\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}^f, \boldsymbol{\nabla}_f \phi) = 0, \forall \text{ face bubble} \phi \end{cases}$$

Sum up the face interpolants $\boldsymbol{E}_{2,p} = \sum_f \boldsymbol{E}_{2,f}^e$

- Project difference $\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p}$ onto the element bubbles,

$$\begin{cases} \|\boldsymbol{\nabla} \times (\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p} - \boldsymbol{E}_{3,p})\|_{L^2(K)} \to \min \\ (\boldsymbol{E} - \boldsymbol{E}_{1,p} - \boldsymbol{E}_{2,p} - \boldsymbol{E}_{3,p}, \boldsymbol{\nabla}\phi)\|_{L^2(K)} \quad \forall \text{ element bubble } \phi \end{cases}$$

- $\Pi^{curl}\boldsymbol{E} = \boldsymbol{E}_{1,p} + \boldsymbol{E}_{2,p} + \boldsymbol{E}_{3,p}$.

$H(\text{div})$ Projection-Based Interpolation

Given a function $\boldsymbol{F} \in \boldsymbol{H}^{\epsilon}(K) \cap \boldsymbol{H}(\text{div}, K)$

$H(\mathrm{div})$ Projection-Based Interpolation

Given a function $\boldsymbol{F} \in \boldsymbol{H}^{\epsilon}(K) \cap \boldsymbol{H}(\mathrm{div}, K)$

- For each face f,

H(div) Projection-Based Interpolation

Given a function $\boldsymbol{F} \in \boldsymbol{H}^{\epsilon}(K) \cap \boldsymbol{H}(\text{div}, K)$

- For each face f,
  - compute the average

$$<(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n, 1>= 0$$

$H(\mathrm{div})$ Projection-Based Interpolation

Given a function $\boldsymbol{F} \in \boldsymbol{H}^{\epsilon}(K) \cap \boldsymbol{H}(\mathrm{div}, K)$

- For each face f,
  - compute the average

$$<(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n, 1 >= 0$$

  - project difference $(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n$ onto face bubbles,

$$\|(\boldsymbol{F} - \boldsymbol{F}_{f,0} - \boldsymbol{F}_{f,1,p})_n\|_{H^{-0.5}(f)} \to \min$$

$H(\text{div})$ Projection-Based Interpolation

Given a function $\boldsymbol{F} \in \boldsymbol{H}^{\epsilon}(K) \cap \boldsymbol{H}(\text{div}, K)$

- For each face f,
  - compute the average
$$<(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n, 1 >= 0$$
  - project difference $(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n$ onto face bubbles,
$$\|(\boldsymbol{F} - \boldsymbol{F}_{f,0} - \boldsymbol{F}_{f,1,p})_n\|_{H^{-0.5}(f)} \rightarrow \min$$
- sum up the face contributions: $\boldsymbol{F}_{1,p} = \sum_f (\boldsymbol{F}_{f,0} + \boldsymbol{F}_{f,1,p})$,

$H(\text{div})$ Projection-Based Interpolation

Given a function $\boldsymbol{F} \in \boldsymbol{H}^{\epsilon}(K) \cap \boldsymbol{H}(\text{div}, K)$

- For each face f,
  - compute the average

$$<(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n, 1 >= 0$$

  - project difference $(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n$ onto face bubbles,

$$\|(\boldsymbol{F} - \boldsymbol{F}_{f,0} - \boldsymbol{F}_{f,1,p})_n\|_{H^{-0.5}(f)} \rightarrow \text{min}$$

- sum up the face contributions: $\boldsymbol{F}_{1,p} = \sum_f (\boldsymbol{F}_{f,0} + \boldsymbol{F}_{f,1,p})$,
- Project difference $\boldsymbol{F} - \boldsymbol{F}_{1,p}$ onto the element bubbles,

$$\begin{cases} \|\boldsymbol{\nabla} \cdot (\boldsymbol{F} - \boldsymbol{F}_{1,p} - \boldsymbol{F}_{2,p})\|_{L^2(K)} \rightarrow \text{min} \\ (\boldsymbol{F} - \boldsymbol{F}_{1,p} - \boldsymbol{F}_{2,p}, \boldsymbol{\nabla} \times \boldsymbol{\phi})_{L^2(K)} = 0, \quad \forall \text{ element (vector bubble} \end{cases}$$

H(div) Projection-Based Interpolation

Given a function $\boldsymbol{F} \in \boldsymbol{H}^{\epsilon}(K) \cap \boldsymbol{H}(\text{div}, K)$

- For each face f,
  - compute the average

$$<(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n, 1 >= 0$$

  - project difference $(\boldsymbol{F} - \boldsymbol{F}_{f,0})_n$ onto face bubbles,

$$\|(\boldsymbol{F} - \boldsymbol{F}_{f,0} - \boldsymbol{F}_{f,1,p})_n\|_{H^{-0.5}(f)} \rightarrow \min$$

- sum up the face contributions: $\boldsymbol{F}_{1,p} = \sum_f (\boldsymbol{F}_{f,0} + \boldsymbol{F}_{f,1,p})$,
- Project difference $\boldsymbol{F} - \boldsymbol{F}_{1,p}$ onto the element bubbles,

$$\begin{cases} \|\boldsymbol{\nabla} \cdot (\boldsymbol{F} - \boldsymbol{F}_{1,p} - \boldsymbol{F}_{2,p})\|_{L^2(K)} \rightarrow \min \\ (\boldsymbol{F} - \boldsymbol{F}_{1,p} - \boldsymbol{F}_{2,p}, \boldsymbol{\nabla} \times \boldsymbol{\phi})_{L^2(K)} = 0, \quad \forall \text{ element (vector bubble} \end{cases}$$

- $\Pi^{div} \boldsymbol{F} = \boldsymbol{F}_{1,p} + \boldsymbol{F}_{2,p}$.

H(div) de Rham diagram

$$
\begin{array}{ccccccc}
H^r(\Omega) & \xrightarrow{\boldsymbol{\nabla}} & \boldsymbol{H}^{r-1}(\mathbf{curl},\Omega) & \xrightarrow{\boldsymbol{\nabla}\times} & \boldsymbol{H}^{r-1}(\mathsf{div},\Omega) & \xrightarrow{\boldsymbol{\nabla}\cdot} & H^{r-1}(\Omega) \\
\Big\downarrow \textstyle\prod^{grad} & & \Big\downarrow \textstyle\prod^{curl} & & \Big\downarrow \textstyle\prod^{div} & & \Big\downarrow P \\
W_p & \xrightarrow{\boldsymbol{\nabla}} & \boldsymbol{Q}_p & \xrightarrow{\boldsymbol{\nabla}\times} & \boldsymbol{V}_p & \xrightarrow{\boldsymbol{\nabla}\cdot} & Y_p
\end{array}
$$

Interpolation Error Etimates

- H^1-interpolation,

$$\|u - \Pi^{grad} u\|_{H^1(K)} \leq C \inf_{u_p} \|u - u_p\|_{H^1(K)}$$

$$+ C\epsilon^{-1.5} \left(\sum_f \inf_{u_p} \|u - u_p\|_{H^{0.5+\epsilon}(f)} + \sum_e \inf_{u_p} \|u - u_p\|_{H^{\epsilon}(e)} \right)$$

Interpolation Error Etimates

- H^1-interpolation,

$$
\begin{aligned}
\|u - \Pi^{grad} u\|_{H^1(K)} &\le C \inf_{u_p} \|u - u_p\|_{H^1(K)} \\
+ C\epsilon^{-1.5} &\left(\sum_f \inf_{u_p} \|u - u_p\|_{H^{0.5+\epsilon}(f)} + \sum_e \inf_{u_p} \|u - u_p\|_{H^{\epsilon}(e)} \right)
\end{aligned}
$$

- $$\le C(\ln p)^{1.5} p^{-(r-1)} \|u\|_{H^r(\Omega)}, \quad r > 1.5$$

Interpolation Error Etimates (cont.)

- $\boldsymbol{H}(\mathbf{curl})$-interpolation,

$$\begin{aligned}\|\boldsymbol{E}-\Pi^{curl}\boldsymbol{E}\|_{\boldsymbol{H}(\mathbf{curl},K)} \leq & C\inf_{\boldsymbol{E}_p}\|\boldsymbol{E}-\boldsymbol{E}_p\|_{\boldsymbol{H}(\mathbf{curl},K)} \\ & + C\epsilon^{-1.5}\Big(\sum_f \inf_{\boldsymbol{E}_p}\|(\boldsymbol{E}-\boldsymbol{E}_p)_t\|_{\boldsymbol{H}^{-0.5+\epsilon}(\mathrm{curl}_f,f)} \\ & + C\sum_e \inf_{\boldsymbol{E}_p}\|(\boldsymbol{E}-\boldsymbol{E}_p)_t\|_{H^{-1}(e)}\Big)\end{aligned}$$

Interpolation Error Etimates (cont.)

- $\boldsymbol{H}(\mathbf{curl})$-interpolation,

$$\begin{aligned}\|\boldsymbol{E} - \Pi^{curl}\boldsymbol{E}\|_{\boldsymbol{H}(\mathbf{curl},K)} \leq & C \inf_{\boldsymbol{E}_p} \|\boldsymbol{E} - \boldsymbol{E}_p\|_{\boldsymbol{H}(\mathbf{curl},K)} \\ & + C\epsilon^{-1.5}\Big(\sum_f \inf_{\boldsymbol{E}_p} \|(\boldsymbol{E} - \boldsymbol{E}_p)_t\|_{\boldsymbol{H}^{-0.5+\epsilon}(\mathrm{curl}_f, f)} \\ & + C\sum_e \inf_{\boldsymbol{E}_p} \|(\boldsymbol{E} - \boldsymbol{E}_p)_t\|_{H^{-1}(e)}\Big)\end{aligned}$$

- $$\leq C(\ln p)^{1.5} p^{-r} \|\boldsymbol{E}\|_{\boldsymbol{H}^r(\mathbf{curl},\Omega)}, \quad r > 0.5$$

Interpolation Error Etimates (cont.)

- $\boldsymbol{H}$(div)-interpolation,

$$\|\boldsymbol{F} - \Pi^{div}\boldsymbol{F}\|_{\boldsymbol{H}(\mathrm{div},K)} \leq C \inf_{\boldsymbol{F}_p} \|\boldsymbol{F} - \boldsymbol{F}_p\|_{\boldsymbol{H}(\mathrm{div},K)}$$
$$+ C\epsilon^{-1.5} \sum_f \inf_{\boldsymbol{F}_p} \|(\boldsymbol{F} - \boldsymbol{F}_p)_n\|_{\boldsymbol{H}^{-0.5+\epsilon}(f)}$$

Interpolation Error Etimates (cont.)

- $\boldsymbol{H}$(div)-interpolation,

$$\begin{aligned}\|\boldsymbol{F}-\Pi^{div}\boldsymbol{F}\|_{\boldsymbol{H}(\mathrm{div},K)} \leq & C\inf_{\boldsymbol{F}_p}\|\boldsymbol{F}-\boldsymbol{F}_p\|_{\boldsymbol{H}(\mathrm{div},K)}\\ & +C\epsilon^{-1.5}\sum_f \inf_{\boldsymbol{F}_p}\|(\boldsymbol{F}-\boldsymbol{F}_p)_n\|_{\boldsymbol{H}^{-0.5+\epsilon}(f)}\end{aligned}$$

-

$$\leq C(\ln p)^{1.5}p^{-r}\|u\|_{\boldsymbol{H}^r(\mathrm{div},\Omega)}, \quad r>0.5$$

Fully-Automatic hp-Adaptivity in Three Dimensions

joint work with: J. Kurtz

Introduction

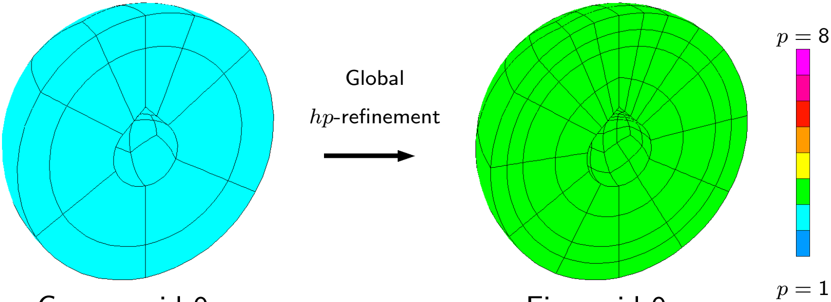
The hp-adaptive finite element method combines local reduction of the element size h with local elevation of the polynomial order of approximation p to achieve exponential convergence with respect to the number of global degrees of freedom N. An algorithm for automatically determining such refinements in an optimal way must address the following issues:

- Selection of optimal, possibly anisotropic, h-refinements for resolution of vertex and edge singularities and boundary and interior layers
- Selection of optimal, possibly anisotropic, distribution of p

These decisions require much more information than traditional h-adaptive methods.

A Two-Grid Paradigm

Our approach is to determine an optimal refinement strategy for a given *coarse grid* by examining the solution on a corresponding *fine grid* obtained by a global hp-refinement.

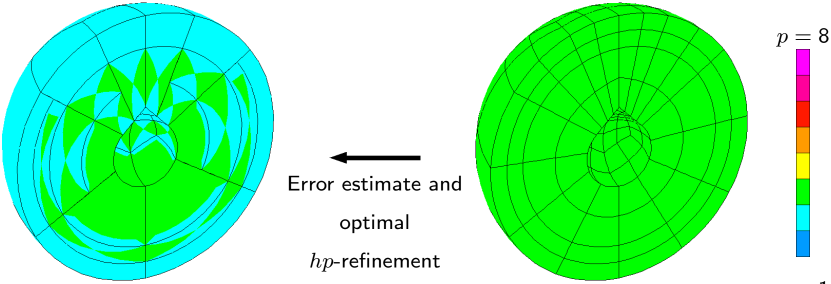


Coarse grid 0

Fine grid 0

A Two-Grid Paradigm

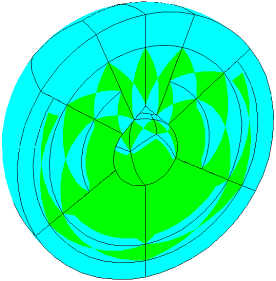
Our approach is to determine an optimal refinement strategy for a given *coarse grid* by examining the solution on a corresponding *fine grid* obtained by a global hp-refinement.



Coarse grid 1 Fine grid 0

A Two-Grid Paradigm

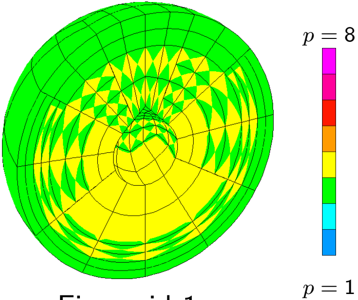
Our approach is to determine an optimal refinement strategy for a given *coarse grid* by examining the solution on a corresponding *fine grid* obtained by a global hp-refinement.



Coarse grid 1

Global

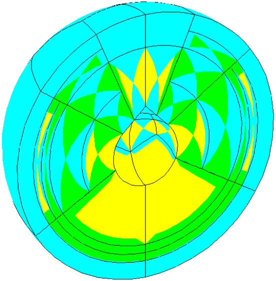
hp-refinement



Fine grid 1

A Two-Grid Paradigm

Our approach is to determine an optimal refinement strategy for a given *coarse grid* by examining the solution on a corresponding *fine grid* obtained by a global hp-refinement.

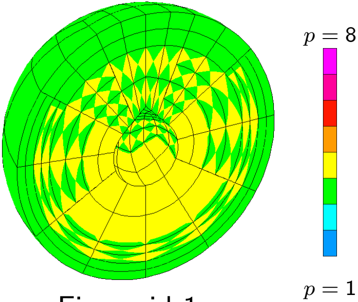


Coarse grid 2

Error estimate and

optimal

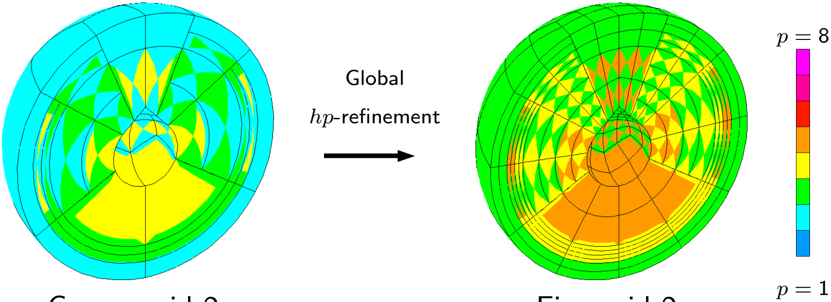
hp-refinement



Fine grid 1

A Two-Grid Paradigm

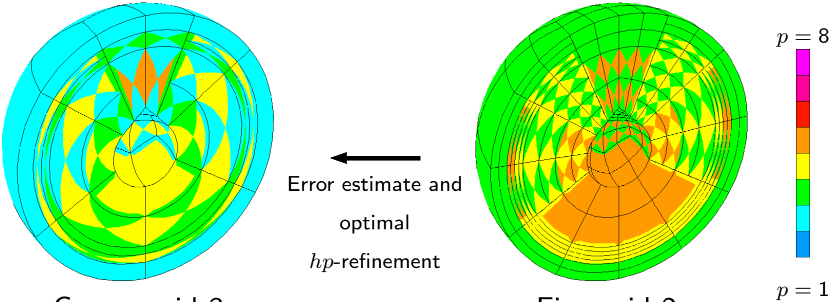
Our approach is to determine an optimal refinement strategy for a given *coarse grid* by examining the solution on a corresponding *fine grid* obtained by a global hp-refinement.



Coarse grid 2 Fine grid 2

A Two-Grid Paradigm

Our approach is to determine an optimal refinement strategy for a given *coarse grid* by examining the solution on a corresponding *fine grid* obtained by a global hp-refinement.

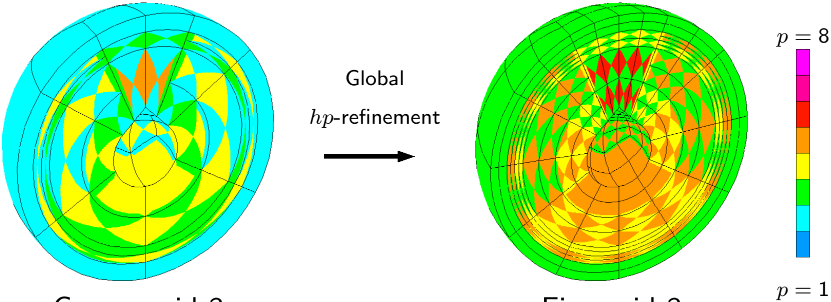


Coarse grid 3

Fine grid 2

A Two-Grid Paradigm

Our approach is to determine an optimal refinement strategy for a given *coarse grid* by examining the solution on a corresponding *fine grid* obtained by a global hp-refinement.



Coarse grid 3 Fine grid 3

The hp-Algorithm

Repeat until exhausted or error is below a prescribed tolerance:

- Solve on the coarse grid and write solution and grid to disk
- Break each element and enrich the order isotropically
- Solve on the resulting fine grid
- Compute error in the coarse grid
- Select refinements for the coarse grid by computing the projection-based interpolant of the fine grid solution onto the coarse grid and a dynamically-determined sequence of intermediate grids
- Possibly enrich optimal refinements to maintain 1-irregularity
- Read coarse grid from disk and perform optimal refinements

The energy-driven mesh optimization algorithm

- Find optimal hp-refinements of the current coarse grid hp yielding the next coarse grid hp^{next} such that $(u = u_{h/2,p+1})$,

$$\frac{\|u - \Pi_{hp} u\|_{H^1} - \|u - \Pi_{hp^{next}} u\|_{H^1}}{N_{hp^{next}} - N_{hp}} \rightarrow \max$$

The energy-driven mesh optimization algorithm

- Find optimal hp-refinements of the current coarse grid hp yielding the next coarse grid hp^{next} such that $(u = u_{h/2,p+1})$,

$$\frac{\|u - \Pi_{hp} u\|_{H^1} - \|u - \Pi_{hp^{next}} u\|_{H^1}}{N_{hp^{next}} - N_{hp}} \to \max$$

- The algorithm reflects the logic of the projection-based interpolation and consists of three steps:
  - Determining optimal refinement of edges
  - Determining optimal refinement of faces
  - Determining optimal refinement of element interiors

  Each of the steps sets up initial conditions for the next step, limiting the number of cases to be considered.

The energy-driven mesh optimization algorithm (cont.)

Once the optimal refinements have been determined, we

- Execute the requested h-refinements, enforcing the 1-irregularity of the mesh
- perform the optimal p-refinements

The goal-driven mesh optimization algorithm

- Primal problem:

$$\begin{cases} u \in V \\ b(u,v) = l(v), \quad \forall v \in V \end{cases}$$

The goal-driven mesh optimization algorithm

- Primal problem:

$$\begin{cases} u \in V \\ b(u,v) = l(v), \quad \forall v \in V \end{cases}$$

- Goal functional: $g(u), g \in V'$ and dual problem:

$$\begin{cases} v \in V \\ b(u,v) = g(v), \quad \forall u \in V \end{cases}$$

The goal-driven mesh optimization algorithm

- Primal problem:

$$\begin{cases} u \in V \\ b(u,v) = l(v), \quad \forall v \in V \end{cases}$$

- Goal functional: $g(u), g \in V'$ and dual problem:

$$\begin{cases} v \in V \\ b(u,v) = g(v), \quad \forall u \in V \end{cases}$$

- Error representation: $(u = u_{h/2,p+1}, v = v_{h/2,p+1})$

$$\begin{aligned} g(u - u_{hp}) &= b(u - u_{hp}, v - \Pi_{hp} v) \\ &\approx b(u - \Pi_{hp} u, v - \Pi_{hp} v) \end{aligned}$$

The goal-driven mesh optimization algorithm

- Primal problem:

$$\begin{cases} u \in V \\ b(u,v) = l(v), \quad \forall v \in V \end{cases}$$

- Goal functional: $g(u), g \in V'$ and dual problem:

$$\begin{cases} v \in V \\ b(u,v) = g(v), \quad \forall u \in V \end{cases}$$

- Error representation: $(u = u_{h/2,p+1}, v = v_{h/2,p+1})$

$$\begin{aligned} g(u - u_{hp}) \quad &= b(u - u_{hp}, v - \Pi_{hp} v) \\ &\approx b(u - \Pi_{hp} u, v - \Pi_{hp} v) \end{aligned}$$

- Error bound:

$$|g(u - u_{hp})| \quad \leq \sum_K |b_K(u - \Pi_{hp} u, v - \Pi_{hp} v)| =: J_{hp}$$

The goal-driven mesh optimization algorithm (cont.)

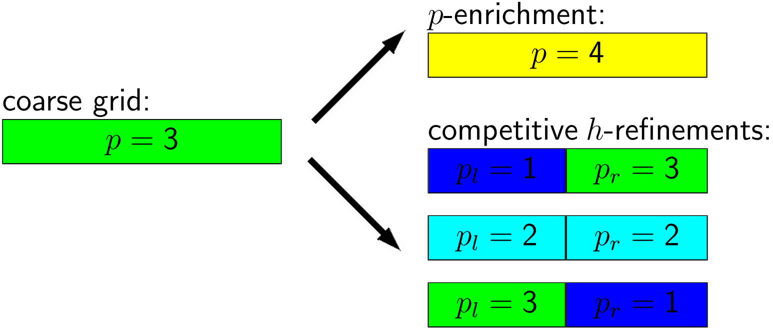
The goal-driven mesh optimization algorithm (cont.)

- Modified optimization problem:
  Find optimal hp-refinements of the current coarse grid hp yielding the next coarse grid hp^{next} such that,

$$\frac{J_{hp} - J_{hp^{next}}}{N_{hp^{next}} - N_{hp}} \rightarrow \text{max}$$

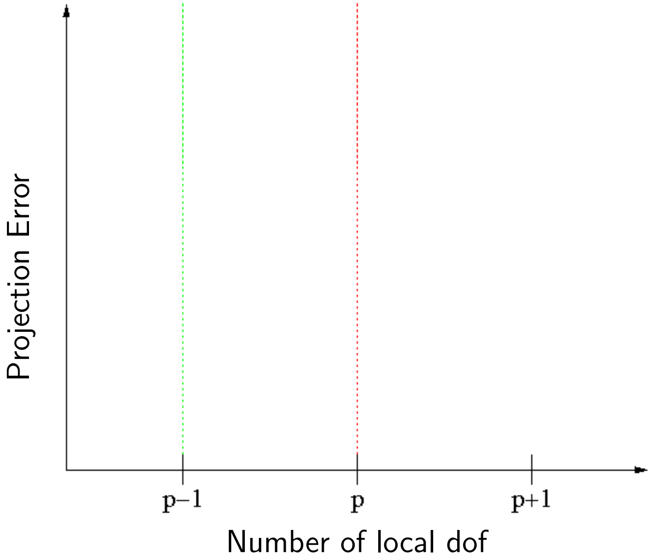
Edge Refinements: Local Competition

We stage a local competition to determine whether h-refinement or p-enrichment is optimal. Since p-enrichment adds a single degree of freedom, so do the competitive h-refinements:

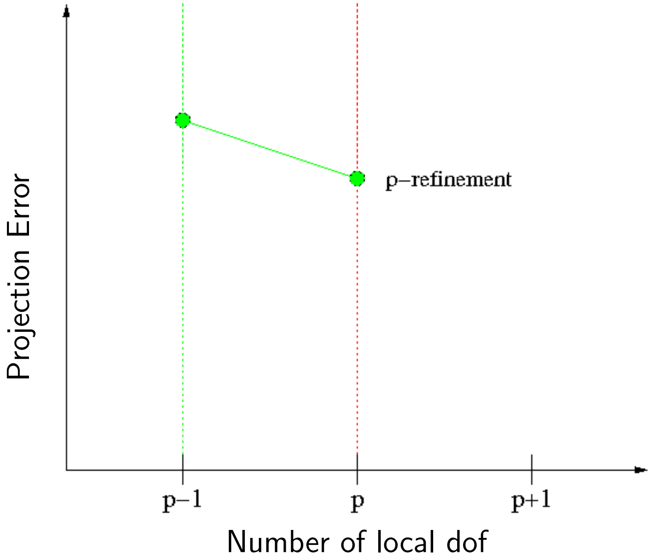


The projection-based interpolant of the fine grid solution is computed for each competitor and we select the one with smallest error.

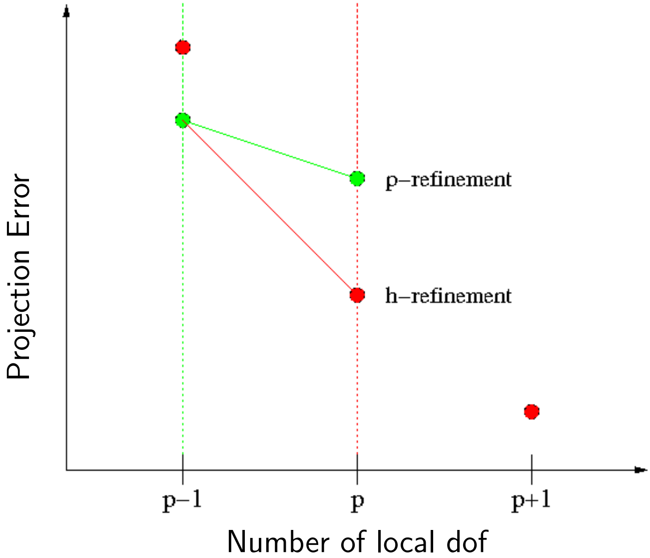
Edge Refinements: Global Competition



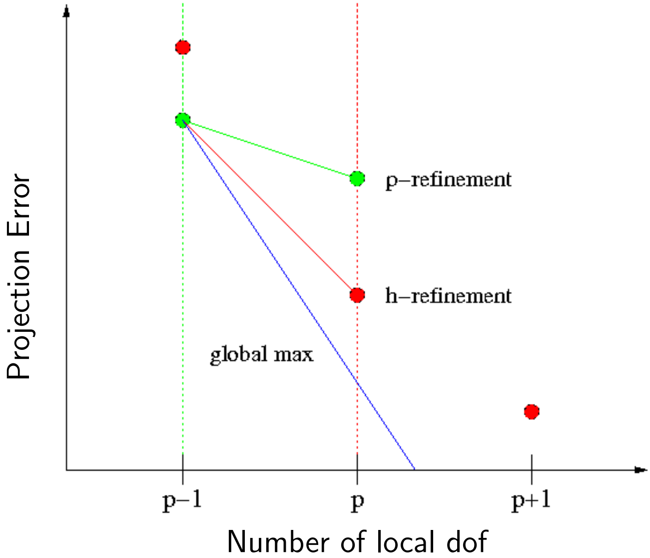
Edge Refinements: Global Competition



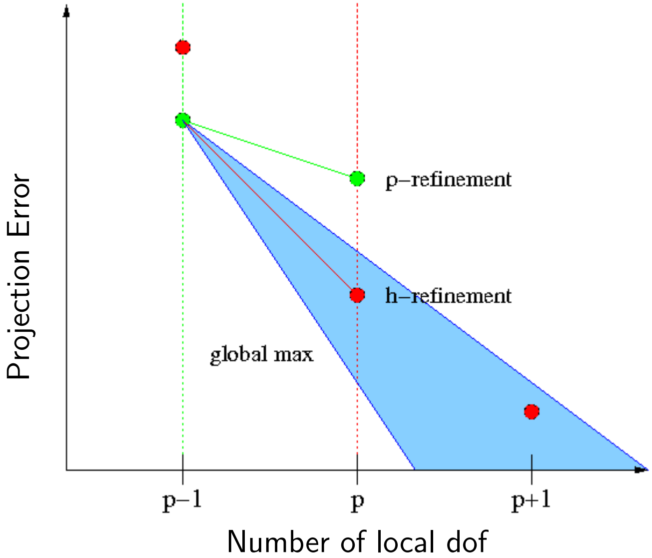
Edge Refinements: Global Competition



Edge Refinements: Global Competition



Edge Refinements: Global Competition



Faces and Interiors

- The same algorithm is then applied to element faces and finally interiors
- The conclusion of each stage provides the starting point for the next, i.e. optimal edge h-refinements and orders, in conjunction with the minimum rule, provide the starting point for face optimization
- Projections onto element interiors are expensive and required the development of a "telescoping solver" that computes a dynamically-determined sequence of nested projections by only updating the previously computed factorization
- The logic is identical for energy-driven and goal-oriented adaptivity: the only difference is in the evaluation of projection errors

Acoustic Scattering from a Rigid Obstacle Ω_{int}

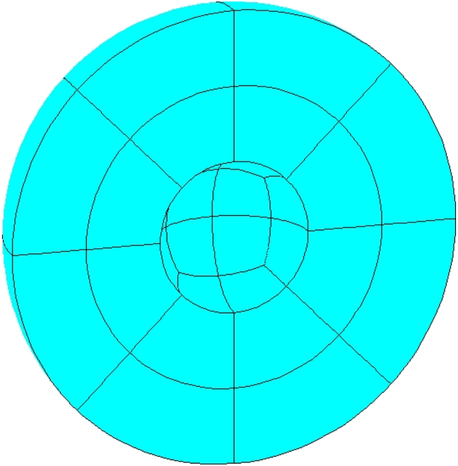
We solve the Helmholtz equation with real wave number k for the scattered pressure u

$$
\begin{aligned}
-\Delta u - k^2 u &= 0 \text{ in } \mathbb{R}^3 \setminus \Omega_{int} \\
\frac{\partial u}{\partial n} &= -\frac{\partial u^{inc}}{\partial n} \text{ on } \Gamma = \partial\Omega_{int} \\
\frac{\partial u}{\partial r} + iku &= o(1/r) \text{ as } r \to \infty
\end{aligned}
$$

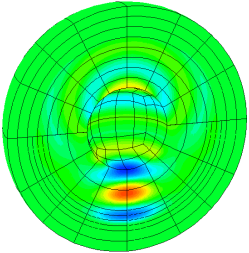
The unbounded exterior domain is truncated via either infinite elements (IE) or a perfectly matched layer (PML).

Scattering from a Sphere of Radius λ

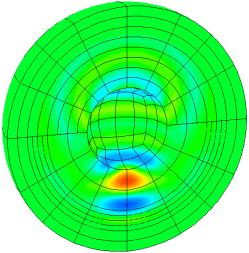
Initial mesh with PML ($p = 2$)



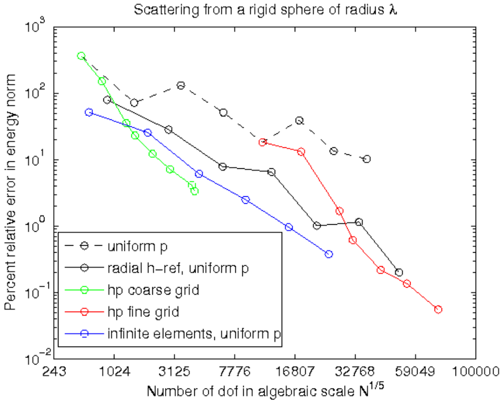
$Re(u)$



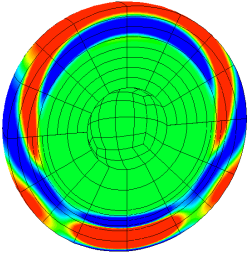
$Im(u)$



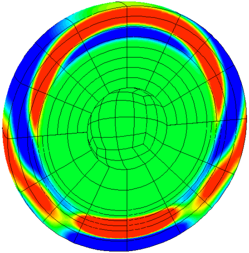
Scattering from a Sphere of Radius λ



$Re(u - u^{ex})$

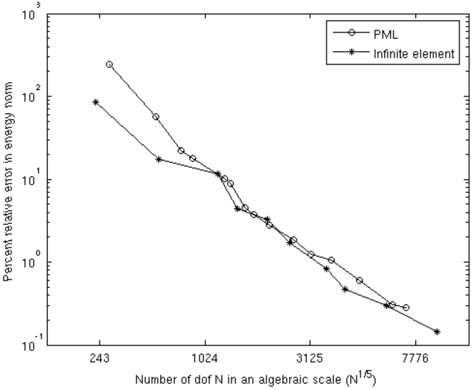


$Im(u - u^{ex})$

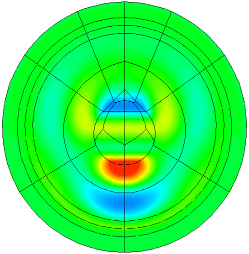


hp-adaptivity resolves the PML without any "parameter tuning"

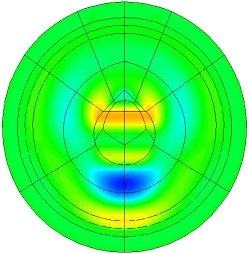
Scattering from a Cone-sphere of diameter λ



$Re(u)$

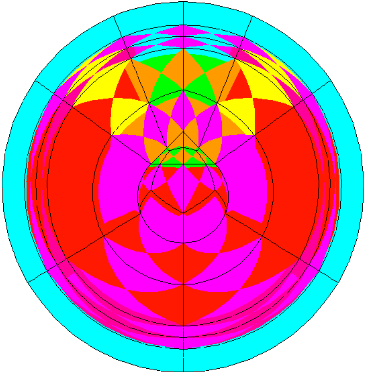


$Im(u)$

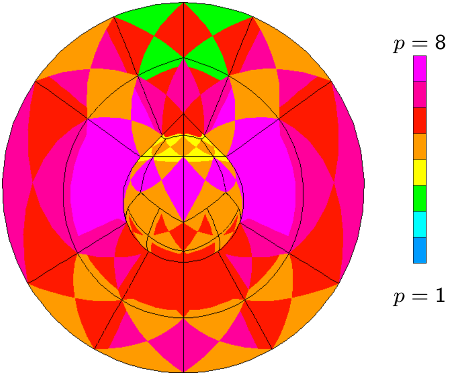


Scattering from a Cone-sphere of diameter λ

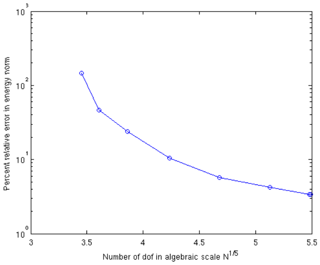
Perfectly matched layer

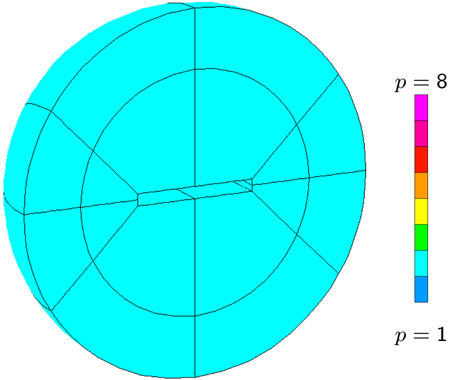


Infinite elements (not shown)

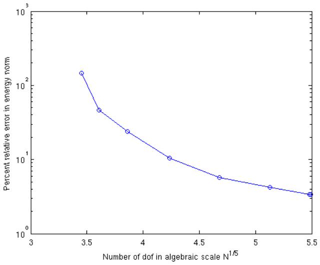


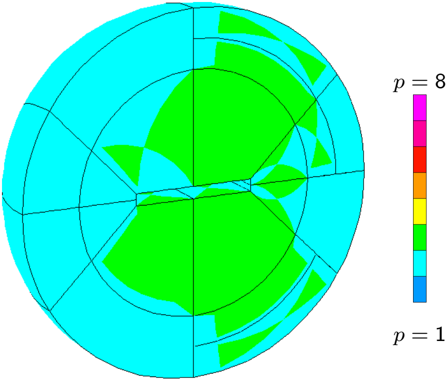
Scattering from a $\lambda \times \lambda \times \lambda/10$ plate



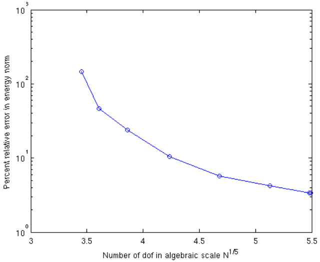


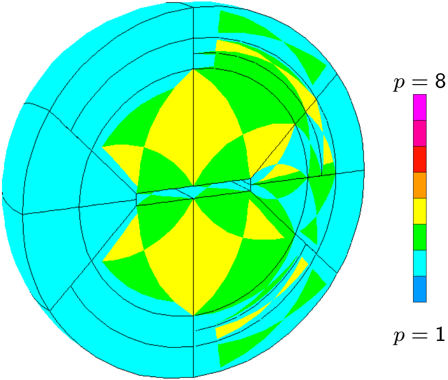
Scattering from a $\lambda \times \lambda \times \lambda/10$ plate



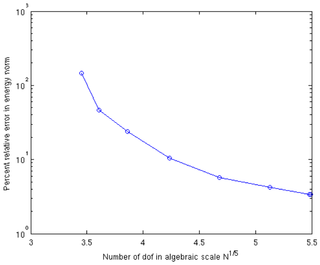


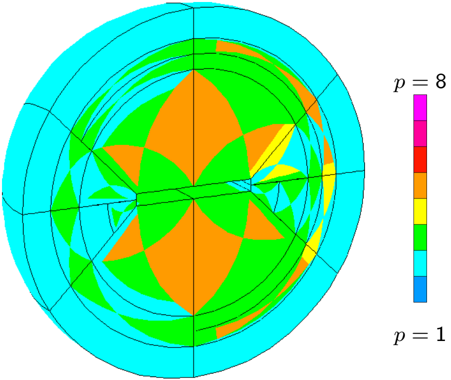
Scattering from a $\lambda \times \lambda \times \lambda/10$ plate



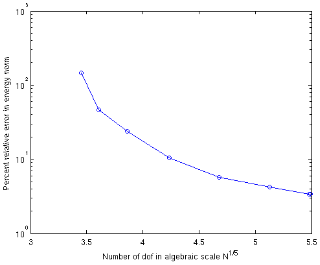


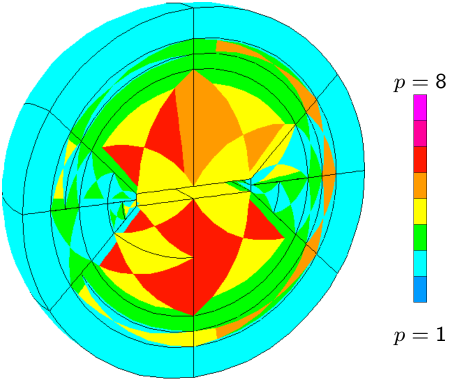
Scattering from a $\lambda \times \lambda \times \lambda/10$ plate



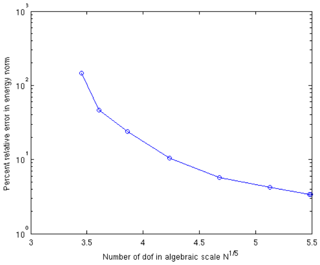


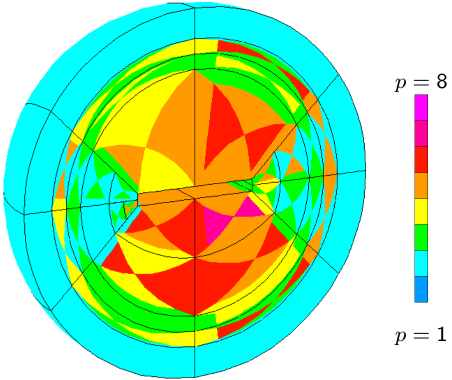
Scattering from a $\lambda \times \lambda \times \lambda/10$ plate



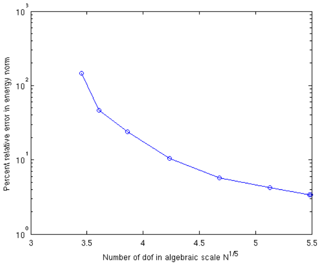


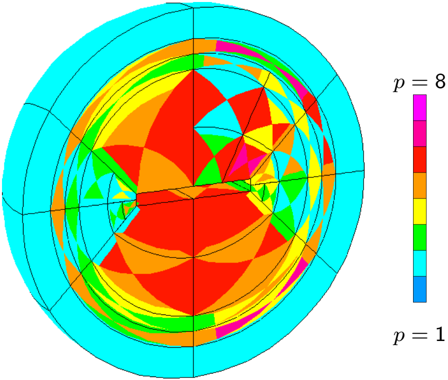
Scattering from a $\lambda \times \lambda \times \lambda/10$ plate



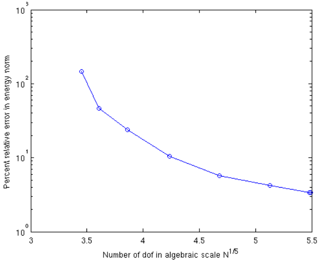


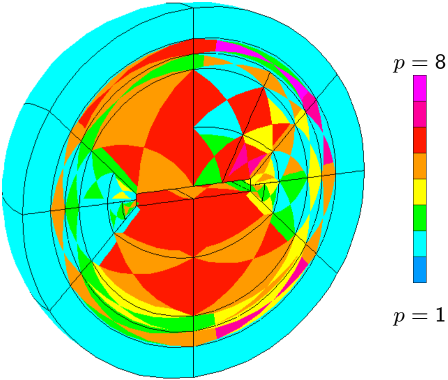
Scattering from a $\lambda \times \lambda \times \lambda/10$ plate





Scattering from a $\lambda \times \lambda \times \lambda/10$ plate





Improving the performance of Perfectly Matched Layers (PML) by means of hp-adaptivity

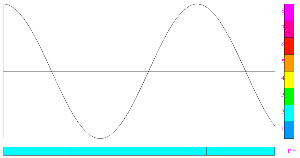
joint work with: C. Michler, J. Kurtz and D. Pardo

Introduction

- Most wave-propagation problems are posed on infinite domains
- Truncation of the computational domain is customarily achieved by the *Perfectly Matched Layer (PML)*
- PML uses analytic continuation of a function into the complex plane (*"complex coordinate stretching"*)
- Satisfactory performance of the PML typically involves tedious parameter tuning
- We show that such parameter tuning becomes obsolete when using *hp*-adaptive finite elements

Motivation

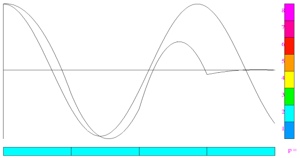
- PML must damp out the wave before it reaches the boundary



1D wave on unbounded domain ..

Motivation

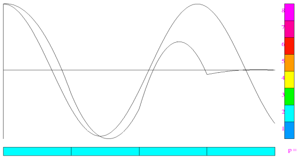
- PML must damp out the wave before it reaches the boundary
- This induces strong gradients (*"boundary layers"*) in the PML
- Such boundary layers are difficult to resolve with conventional methods



1D wave on unbounded domain ..
.. and numerical solution

Motivation

- PML must damp out the wave before it reaches the boundary
- This induces strong gradients (*"boundary layers"*) in the PML
- Such boundary layers are difficult to resolve with conventional methods

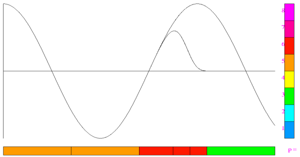


1D wave on unbounded domain .. .. and numerical solution

- Not resolving these boundary layers can significantly degrade the accuracy of the solution in the domain of interest

Motivation

- PML must damp out the wave before it reaches the boundary
- This induces strong gradients (*"boundary layers"*) in the PML
- Such boundary layers are difficult to resolve with conventional methods



1D wave on unbounded domain ..
.. and *hp-refined* numerical solution

- Not resolving these boundary layers can significantly degrade the accuracy of the solution in the domain of interest
- Remedy: Use *hp*-adaptivity to resolve such PML-induced boundary layers

Formulation for 2D acoustics

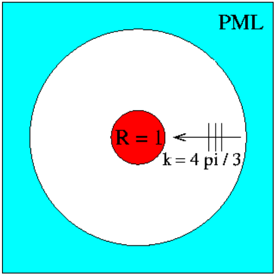
2D Helmholtz equation in polar coord. (stretching $r \to z$):

$$\begin{cases} p \in \tilde{p}_D + V \\ \displaystyle\int_\Omega \left(\frac{z}{z'r} \frac{\partial p}{\partial r} \frac{\partial q}{\partial r} + \frac{z'}{rz} \frac{\partial p}{\partial \theta} \frac{\partial q}{\partial \theta} - \left(\frac{\omega}{c}\right)^2 \frac{z'z}{r} pq \right) r dr d\theta = 0 \quad \forall q \in V \end{cases}$$

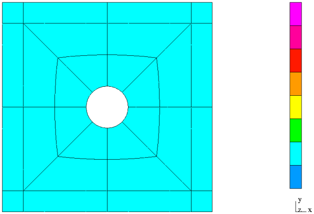
2D Helmholtz

Scattering of a plane wave on a unit cylinder:

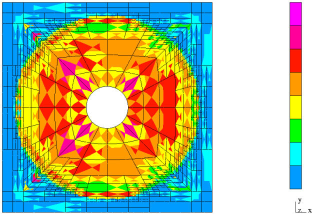
- PML truncated with homogeneous Dirichlet BC
- Analytic solution available



2D Helmholtz: cylindrical PML



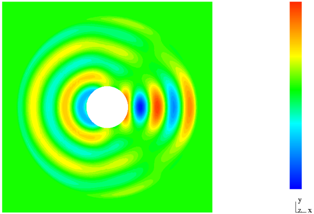
Initial mesh



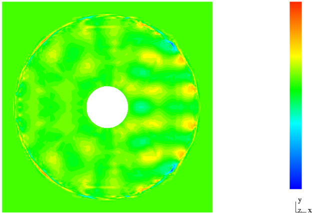
Final mesh (1% error)

redPML can be chosen independently of mesh geometry !

2D Helmholtz: cylindrical PML



Real part of solution



Real part of error function

2D elasticity: cartesian PML

Variational formulation with complex stretching $x_i \to z_i$

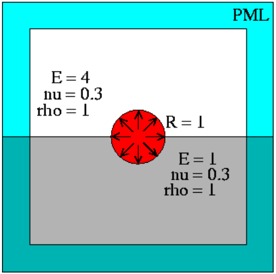
$$\begin{cases} \boldsymbol{u} \in \bar{\boldsymbol{V}}, \\ \displaystyle\int_\Omega E_{ijkl} \frac{z'}{z'_j z'_l} u_{k,l} v_{i,j} \, d\boldsymbol{x} - \omega^2 \int_\Omega \rho z' u_i v_i \, d\boldsymbol{x} \\ \displaystyle = \int_{\Gamma_N} g_i v_i \, dS, \quad \forall \boldsymbol{v} \in \bar{\boldsymbol{V}} \quad \text{with} \quad z' := z'_1 z'_2 z'_3 \end{cases}$$

- "Modified" elasticity tensor (*non-symmetric*) and density
- Problem is complex symmetric
- Note: formulation also valid for layered media with interfaces aligned with the Cartesian axes

2D elasticity: cartesian PML

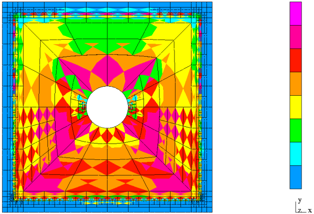
Wave propagation in an elastic layered medium:

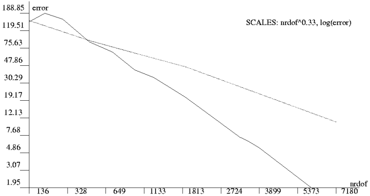
- PML truncated with homogeneous Dirichlet BC
- BC: traction of unit pressure along circular hole
- nondim. wavenumber $k = \pi$



2D elasticity: cartesian PML

Wave propagation in an elastic layered medium





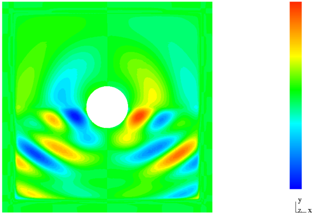
hp mesh (2% relative error)

Error vs. # DOFs for *hp*-adaptive and uniform *h*-refinement

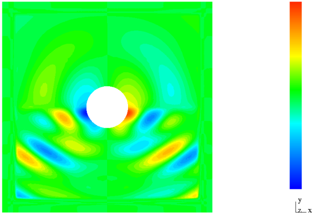
2D elasticity: cartesian PML

Wave propagation in an elastic layered medium

Horizontal displacement



Real part of solution



Imaginary part of solution

2D electromagnetics: cartesian PML

Variational formulation of Maxwell's equations *(for 3D setting)*

$$\begin{cases} \boldsymbol{E} \in \boldsymbol{W} \\ \displaystyle\int_\Omega \left((-\omega^2\epsilon + i\omega\sigma) \sum_i \frac{z'}{(z_i')^2} E_i F_i - \frac{z'}{\mu} \sum_k (\sum_{m,n} \epsilon_{kmn} \frac{1}{z_n' z_m'} E_{n,m})(\sum_{i,j} \epsilon_{kij} \frac{1}{z_i' z_j'} F_{i,j}) \right) d\boldsymbol{x} \\ \displaystyle = -i\omega \int_\Omega \frac{z'}{z_i'} J_i^{\mathsf{imp}} F_i \, d\boldsymbol{x} + i\omega \int_{\Gamma_N} J_{S,i}^{\mathsf{imp}} F_i \, dS, \quad \forall \boldsymbol{F} \in \boldsymbol{W} \end{cases}$$

with $E_i := z_i' E_i$ and $F_i := z_i' F_i$

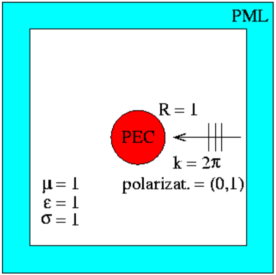
Weighted energy space

$$\boldsymbol{W} = \left\{ E_i \; : \; \frac{|z'|^{\frac{1}{2}}}{|z_i'|} E_i, \; \frac{|z_i'|}{|z'|^{\frac{1}{2}}} (\boldsymbol{\nabla} \times \boldsymbol{E})_i \in L^2(\Omega), \quad \boldsymbol{n} \times \boldsymbol{E} = \boldsymbol{0} \text{ on } \Gamma_D \right\}$$

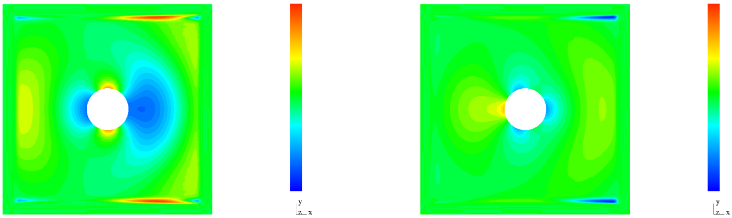
2D Maxwell: cartesian PML

Scattering of a plane wave on a PEC cylinder:

- BC: PEC (perfect electric conductor) cylinder
- PML truncated with homogeneous Dirichlet BC
- Analytic solution available



2D Maxwell: cartesian PML



Real part *(left)* and imaginary part *(right)* of 2^{nd} comp. of solution

- Solve for modified variables $z_1' E_1$, $z_2' E_2$
- In contrast to Helmholtz, for Maxwell the solution may exhibit an initial growth in the PML

Conclusions: *hp*-adaptivity for PML

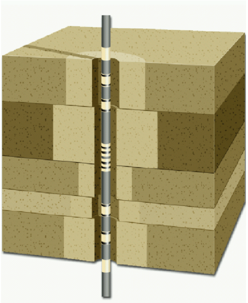
- PML induces *"boundary layers"*
- Resolution of such boundary layers with conventional methods is difficult / costly
- Not resolving such boundary layers can significantly degrade the accuracy of the solution in the domain of interest
- *hp*-adaptivity is capable of resolving such boundary layers in an efficient and automatic way
- Hence, *hp*-adaptivity is ideally suited for PML ... and, conversely, PML is ideally suited for *hp*-adaptive FEM

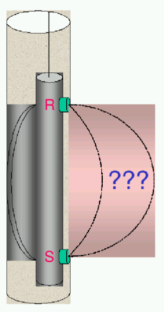
Application of goal-oriented *hp*-adaptivity to EM borehole simulations

joint work with: D. Pardo and C. Torres-Verdin

Resistivity Logging Instruments

Main Objective: To Solve an Inverse Problem





A software for solving the DIRECT problem is essential in order to solve the INVERSE problem

Time-Harmonic Maxwell's Equations

$\nabla \times \mathbf{H} = (\bar{\bar{\sigma}} + j\omega\bar{\bar{\epsilon}})\mathbf{E} + \mathbf{J}^{imp}$	Ampere's law
$\nabla \times \mathbf{E} = -j\omega\bar{\bar{\mu}}\mathbf{H} - \mathbf{M}^{imp}$	Faraday's law
$\nabla \cdot (\bar{\bar{\epsilon}}\mathbf{E}) = \rho$	Gauss' law of Electricity
$\nabla \cdot (\bar{\bar{\mu}}\mathbf{H}) = 0$	Gauss' law of Magnetism

3D E-variational formulation: Find $\mathbf{E} \in \mathbf{E}_D + H_D(\mathbf{curl};\Omega)$ such that:

$$\int_\Omega (\bar{\bar{\mu}}^{-1}\boldsymbol{\nabla} \times \mathbf{E}) \cdot (\boldsymbol{\nabla} \times \bar{\mathbf{F}})\, dV - \int_\Omega (\bar{\bar{k}}^2\mathbf{E}) \cdot \bar{\mathbf{F}}\, dV = -j\omega \int_\Omega \mathbf{J}^{imp} \cdot \bar{\mathbf{F}}\, dV$$

$$+j\omega \int_{\Gamma_N} \mathbf{J}^{imp}_{\Gamma_N} \cdot \bar{\mathbf{F}}_t dS - \int_\Omega (\bar{\bar{\mu}}^{-1}\mathbf{M}^{imp}) \cdot (\boldsymbol{\nabla} \times \bar{\mathbf{F}}) dV \quad \forall \mathbf{F} \in H_D(\mathbf{curl};\Omega)$$

2D Variational Formulation (Axi-symm. Problems)

$\mathbf{E}_\phi$ -Variational Formulation (Azimuthal): Find $E_\phi \in E_{\phi,D} + \tilde{H}^1_D(\Omega)$ s.t. :

$$\int_\Omega (\bar{\bar{\mu}}^{-1}_{\rho,z} \nabla \times \mathbf{E}_\phi)\cdot(\nabla \times \bar{\mathbf{F}}_\phi) dV - \int_\Omega (\bar{\bar{k}}^2_\phi \mathbf{E}_\phi)\cdot \bar{\mathbf{F}}_\phi dV = -j\omega \int_\Omega J^{imp}_\phi \bar{F}_\phi dV$$

$$+ j\omega \int_{\Gamma_N} J^{imp}_{\phi,\Gamma_N} \bar{F}_\phi \, dS - \int_\Omega (\bar{\bar{\mu}}^{-1}_{\rho,z} \mathbf{M}^{imp}_{\rho,z}) \cdot \bar{\mathbf{F}}_\phi \, dV \qquad \forall\, F_\phi \in \tilde{H}^1_D(\Omega)$$

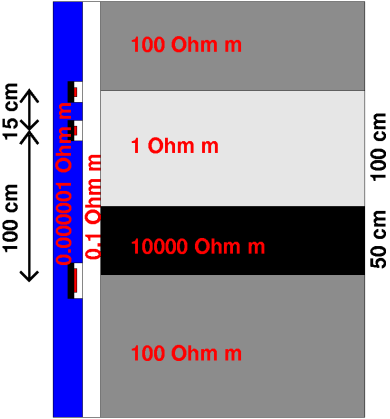
$\mathbf{E}_{\rho,z}$ -Variational Formulation (Meridian):
Find $(E_\rho, E_z) \in \mathbf{E}_D + \tilde{H}_D(\mathbf{curl};\Omega)$ such that:

$$\int_\Omega (\bar{\bar{\mu}}^{-1}_\phi \nabla \times \mathbf{E}_{\rho,z}) \cdot (\nabla \times \bar{\mathbf{F}}_{\rho,z})\, dV - \int_\Omega (\bar{\bar{k}}^2_{\rho,z} \mathbf{E}_{\rho,z}) \cdot \bar{\mathbf{F}}_{\rho,z}\, dV =$$

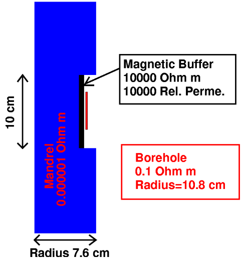
$$- j\omega \int_\Omega J^{imp}_\rho \bar{F}_\rho + J^{imp}_z \bar{F}_z \, dV + j\omega \int_{\Gamma_N} J^{imp}_{\rho,\Gamma_N} \bar{F}_\rho + J^{imp}_{z,\Gamma_N} \bar{F}_z \, dS$$

$$- \int_\Omega (\bar{\bar{\mu}}^{-1}_\phi \mathbf{M}^{imp}_\phi) \cdot \bar{\mathbf{F}}_{\rho,z}\, dV \qquad \forall\, (F_\rho, F_z) \in \tilde{H}_D(\mathbf{curl};\Omega)$$

2D hp-FEM: Induction Instruments



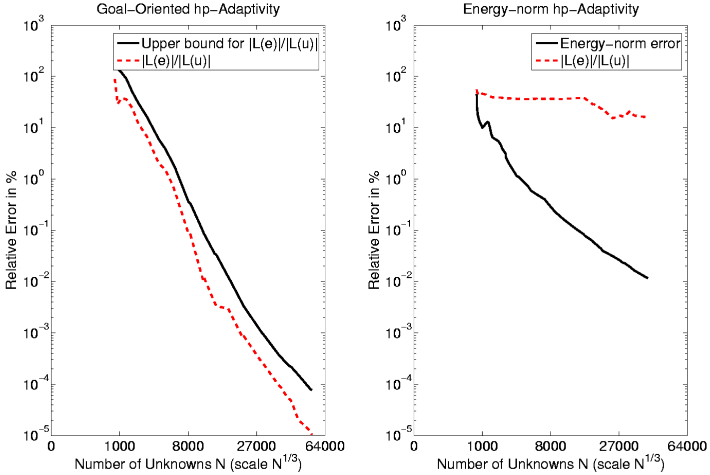
Description of Antennas



Goal: To Study the Effect of Invasion, Anistotropy, and Magnetic Permeability.

2D hp-FEM: Induction Instruments

First. Vert. Diff. E_ϕ (solenoid). Position: 0.475m



Goal-Oriented vs. Energy-norm hp-Adaptivity

Problem with Mandrel at 2 Mhz.

Continuous Elements (Goal-Oriented Adaptivity)

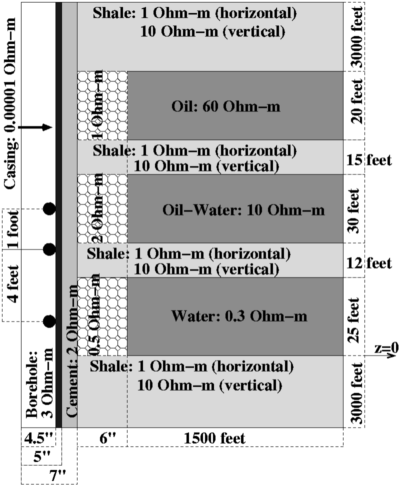
Quantity of Interest	Real Part	Imag Part
COARSE GRID	-0.1629862203E-01	-0.4016944732E-02
FINE GRID	-0.1629862347E-01	-0.4016944223E-02

Continuous Elements (Energy-norm Adaptivity)

Quantity of Interest	Real Part	Imag Part
0.01% ENERGY ERROR	-0.1382759158E-01	-0.2989492851E-02

It is critical to use GOAL-ORIENTED adaptivity.

2D hp-FEM: Through-casing Instruments



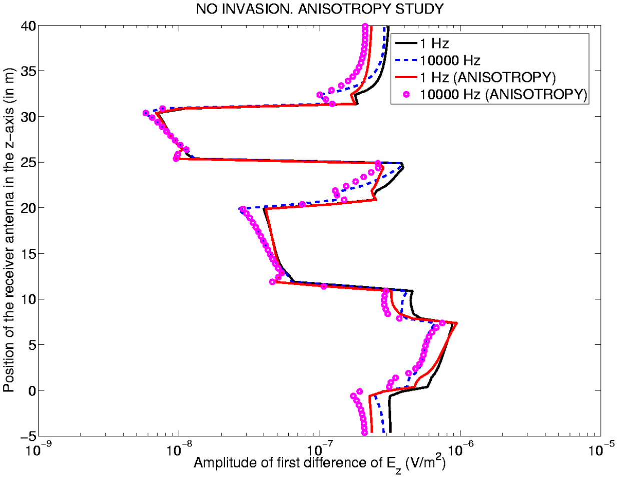
Axisymmetric 3D problem.

Seven different materials with high contrast on resistivity.

Through casing resistivity instrument.

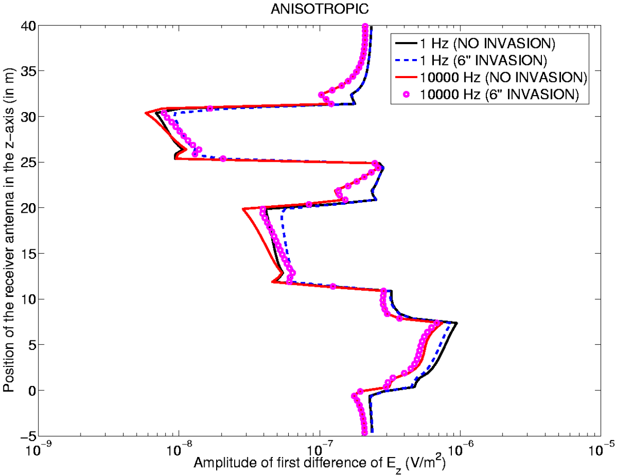
Objective: Study the effect of invasion and anisotropy THROUGH CASING.

2D hp-FEM: Through-casing Instruments



Study of anisotropy and frequency effects requires high accuracy simulations

2D hp-FEM: Through-casing Instruments



Variations due to invasion are below 20%.

2D hp-FEM: Perfectly Matched Layer (PML)

The PML is composed of the following anisotropic materials:

$$\begin{array}{l} \bar{\bar{\boldsymbol{\sigma}}}_{PML} = \bar{\bar{\boldsymbol{\Lambda}}}\bar{\bar{\boldsymbol{\sigma}}} \\ \bar{\bar{\boldsymbol{\epsilon}}}_{PML} = \bar{\bar{\boldsymbol{\Lambda}}}\bar{\bar{\boldsymbol{\epsilon}}} \\ \bar{\bar{\boldsymbol{\mu}}}_{PML} = \bar{\bar{\boldsymbol{\Lambda}}}\bar{\bar{\boldsymbol{\mu}}} \end{array} \; ; \; \bar{\bar{\boldsymbol{\Lambda}}} = \begin{bmatrix} \frac{\tilde{\rho}}{\rho}\frac{s_z}{s_\rho} & 0 & 0 \\ 0 & \frac{\rho}{\tilde{\rho}} s_z s_\rho & 0 \\ 0 & 0 & \frac{\tilde{\rho}}{\rho}\frac{s_\rho}{s_z} \end{bmatrix} ; \; \tilde{\rho} = \int_0^\rho s_\rho(\rho')d\rho'$$

s_ρ, s_ϕ, and s_z are the stretching coordinate functions. We define:

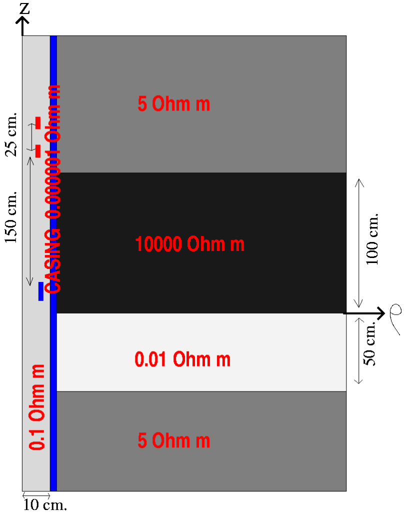
$$s_\rho = s_\phi = s_z = 1 + \phi - j\phi$$

We consider 3 different PML's by defining 3 different functions $\phi(x)$:

$$\phi(x) = \begin{cases} \phi_1(x) = \left[2\left(\frac{x - x_0}{x_1 - x_0}\right)\right]^{17} & \text{PML 1,} \\ \phi_2(x) = 20000 \left(\frac{x - x_0}{x_1 - x_0}\right) & \text{PML 2,} \\ \phi_3(x) = 10000 & \text{PML 3.} \end{cases} \qquad x \in (x_0, x_1)$$

Within the PML, both propagating and evanescent waves become arbitrarily fast evanescent waves.

2D hp-FEM: Perfectly Matched Layer (PML)



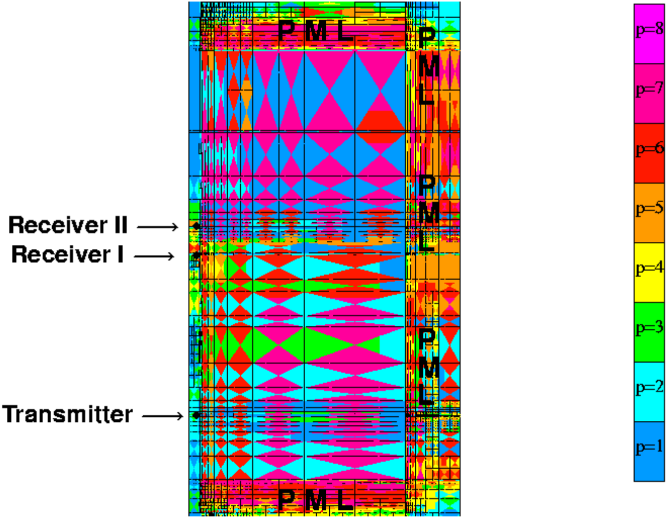
Axisymmetric 3D problem.

Six different materials.

Through casing resistivity instrument.

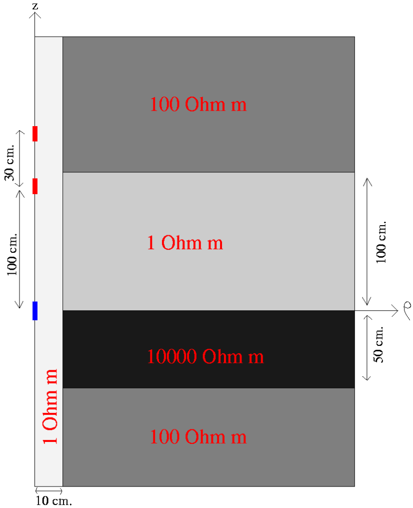
2D hp-FEM: Perfectly Matched Layer (PML)

Final hp-Grid with a 0.5 m thick PML



3D hp-FEM: Numerical Results

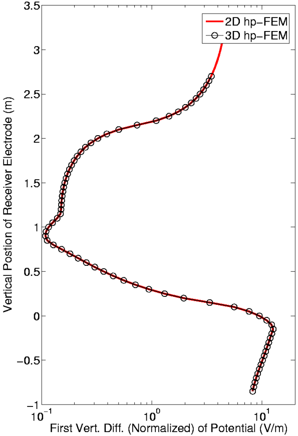
Axisymmetric Model Problem

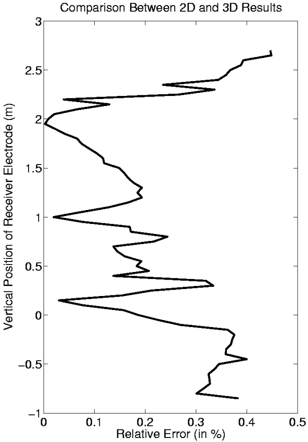


- Borehole and four materials on the formation.
- Size of computational domain: $100m \times 100m$.
- Size of electrode: $0.05m \times 0.05m$.
- Objective: Compute First Vertical Difference of Potential.

3D hp-FEM: Numerical Results

Axisymmetric Model Problem





Conclusions and prospects

- Energy-/goal-oriented *hp*-adaptivity is a versatile and powerful tool for solving challenging engineering problems
- Precise representation of geometry is crucial
- Solution of challenging problems necessitates the use of problem-specific energy-norms
- Two-grid paradigm motivates the use of two-grid solvers; research needed for indefinite problems and anisotropic meshes