High-Order/hp-Adaptive Discontinuous Galerkin Finite Element Methods for Compressible Fluid Flows

Stefano Giani

School of Mathematical Sciences, University of Nottingham, UK

Funded by EPSRC & EU

ADIGMA

Joint work with

Paul Houston

www.AptoFEM.com

BAMC 2009

Outline

1. **Introduction**
   - Second–Order PDEs
2. **Adaptive algorithms**
   - Anisotropic h–Refinement
   - Anisotropic p–Refinement
3. **Numerical Experiments**
   - Compressible NS
   - 3D Experiments

Outline

1. **Introduction**
   - Second–Order PDEs
2. **Adaptive algorithms**
   - Anisotropic *h*–Refinement
   - Anisotropic *p*–Refinement
3. **Numerical Experiments**
   - Compressible NS
   - 3D Experiments

Abstract

Aims

1. Construction of *High–Order DGFEMs* for a class of second–order (quasilinear) PDEs;
2. Develop the *a posteriori* error analysis and adaptive mesh design of the DGFEM approximation of target functionals of the solution based on employing *anisotropic* h–/hp–refined meshes.

- Measurement Problem: Given a user–defined tolerance $\mathtt{TOL} > 0$, can we efficiently design $S_{h,p}$ such that

$$|J(\mathbf{u}) - J(\mathbf{u}_h)| \leq \mathtt{TOL}.$$

Fluid dynamics: drag and lift coefficients.
Other examples: point value, flux, mean value, etc.

- Applications
  - Compressible (aerodynamic) flows. ADIGMA

P. Houston (Nottingham), R. Hartmann (DLR, Braunschweig)

Measurement Problem for Compressible

- Measurement Problem: Given a user–defined tolerance $\mathtt{TOL} > 0$, can we efficiently design $S_{h,p}$ such that

$$|J(\mathbf{u}) - J(\mathbf{u}_h)| \leq \mathtt{TOL}.$$

Fluid dynamics: drag and lift coefficients.
Other examples: point value, flux, mean value, etc.

- Applications
  - Compressible (aerodynamic) flows. ADIGMA

P. Houston (Nottingham), R. Hartmann (DLR, Braunschweig)

- Measurement Problem: Given a user–defined tolerance $\mathtt{TOL} > 0$, can we efficiently design $S_{h,p}$ such that

$$|J(\mathbf{u}) - J(\mathbf{u}_h)| \leq \mathtt{TOL}.$$

Fluid dynamics: drag and lift coefficients.
Other examples: point value, flux, mean value, etc.

- Applications
  - Compressible (aerodynamic) flows. ADIGMA

P. Houston (Nottingham), R. Hartmann (DLR, Braunschweig)

Outline

1. **Introduction**
   - Second–Order PDEs
2. **Adaptive algorithms**
   - Anisotropic h–Refinement
   - Anisotropic p–Refinement
3. **Numerical Experiments**
   - Compressible NS
   - 3D Experiments

Model Problem

- Second–Order Quasilinear System:
  Given $\Omega \subset \mathbb{R}^n$ and $\mathbf{f} \in [L_2(\Omega)]^m$, find $\mathbf{u} : \Omega \to \mathbb{R}^m$, such that

$$\operatorname{div}(\mathcal{F}^c(\mathbf{u}) - \mathcal{F}^v(\mathbf{u}, \nabla\mathbf{u})) = \mathbf{f} \quad \text{in } \Omega.$$

- Writing

$$\mathcal{F}_i^v(\mathbf{u}, \nabla\mathbf{u}) = G_{ij}(\mathbf{u})\partial\mathbf{u}/\partial x_j, \qquad i = 1, \ldots, n,$$

  where $G_{ij}(\mathbf{u}) = \partial\mathcal{F}_i^v(\mathbf{u}, \nabla\mathbf{u})/\partial\mathbf{u}_{x_j}$, $i, j = 1, \ldots, n$, gives

$$\frac{\partial}{\partial x_i}\left(\mathcal{F}_i^c(\mathbf{u}) - G_{ij}(\mathbf{u})\frac{\partial\mathbf{u}}{\partial x_j}\right) = 0 \quad \text{in } \Omega.$$

- Boundary conditions, for example,

$$\mathbf{u} = \mathbf{g}_{\mathrm{D}} \quad \text{on } \partial\Omega_{\mathrm{D}}, \qquad \mathcal{F}^v(\mathbf{u}, \nabla\mathbf{u}) \cdot \mathbf{n} = \mathbf{g}_{\mathrm{N}} \text{ on } \partial\Omega_{\mathrm{N}}.$$

$$
\begin{aligned}
\mathcal{N}(\mathbf{u}_h, \mathbf{v}) \quad &:= \quad -\int_\Omega \mathcal{F}^c(\mathbf{u}_h) : \nabla_h \mathbf{v} \mathrm{d}\mathbf{x} + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} \mathcal{H}(\mathbf{u}^{\mathrm{int}}, \mathbf{u}^{\mathrm{ext}}, \mathbf{n}_\kappa) \cdot \mathbf{v}^{\mathrm{int}} \mathrm{d}s \\
&\quad + \int_\Omega \mathcal{F}^v(\mathbf{u}_h, \nabla_h \mathbf{u}_h) : \nabla_h \mathbf{v} \mathrm{d}\mathbf{x} - \int_{\mathcal{F}_h} \{\!\{\mathcal{F}^v(\mathbf{u}_h, \nabla_h \mathbf{u}_h)\}\!\} : \underline{[\![\mathbf{v}]\!]} \mathrm{d}s \\
&\quad - \int_{\mathcal{F}_h} \{\!\{G^\top(\mathbf{u}_h) \nabla_h \mathbf{v}\}\!\} : \underline{[\![\mathbf{u}_h]\!]} \mathrm{d}s + \int_{\mathcal{F}_h} \underline{\delta}(\mathbf{u}_h) : \underline{[\![\mathbf{v}]\!]} \mathrm{d}s, \\
\ell(\mathbf{v}) \quad &:= \quad \sum_{\kappa \in \mathcal{T}_h} \int_\kappa \mathbf{f} \cdot \mathbf{v} \mathrm{d}\mathbf{x}.
\end{aligned}
$$

where

$\{\!\{\cdot\}\!\}$: Average Operator $\qquad$ $[\![\cdot]\!]$: Jump Operator

$$
\begin{aligned}
\mathcal{N}(\mathbf{u}_h, \mathbf{v}) &:= -\int_\Omega \mathcal{F}^c(\mathbf{u}_h) : \nabla_h \mathbf{v} \mathrm{d}\mathbf{x} + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} \mathcal{H}(\mathbf{u}^{\mathrm{int}}, \mathbf{u}^{\mathrm{ext}}, \mathbf{n}_\kappa) \cdot \mathbf{v}^{\mathrm{int}} \mathrm{d}s \\
&\quad + \int_\Omega \mathcal{F}^v(\mathbf{u}_h, \nabla_h \mathbf{u}_h) : \nabla_h \mathbf{v} \mathrm{d}\mathbf{x} - \int_{\mathcal{F}_h} \{\!\{\mathcal{F}^v(\mathbf{u}_h, \nabla_h \mathbf{u}_h)\}\!\} : \underline{[\![\mathbf{v}]\!]} \mathrm{d}s \\
&\quad - \int_{\mathcal{F}_h} \{\!\{G^\top(\mathbf{u}_h) \nabla_h \mathbf{v}\}\!\} : \underline{[\![\mathbf{u}_h]\!]} \mathrm{d}s + \int_{\mathcal{F}_h} \underline{\delta}(\mathbf{u}_h) : \underline{[\![\mathbf{v}]\!]} \mathrm{d}s, \\
\ell(\mathbf{v}) &:= \sum_{\kappa \in \mathcal{T}_h} \int_\kappa \mathbf{f} \cdot \mathbf{v} \mathrm{d}\mathbf{x}.
\end{aligned}
$$

DGFEM

Find $\mathbf{u}_h \in S_{h,\vec{\mathbf{p}}}$ such that

$$
\mathcal{N}(\mathbf{u}_h, \mathbf{v}_h) = \ell(\mathbf{v}_h) \qquad \forall \mathbf{v}_h \in S_{h,\vec{\mathbf{p}}}.
$$

Penalty Parameter

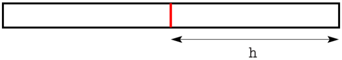
Given $C_{\mathrm{ip}} > 0$, we define

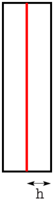
$$\underline{\delta}(\mathbf{u}_h)|_f = C_{\mathrm{ip}} \frac{\mathrm{p}_f^2}{\mathrm{h}_f} \underline{[\![\mathbf{u}_h]\!]} \quad \text{for } f \in \mathcal{F}_h.$$

Given $C_{\mathrm{ip}} > 0$, we define

$$\underline{\delta}(\mathbf{u}_h)|_f = C_{\mathrm{ip}} \frac{\mathrm{p}_f^2}{\mathrm{h}_f} \underline{[\![\mathbf{u}_h]\!]} \quad \text{for } f \in \mathcal{F}_h.$$

$$\mathrm{h}_f = \frac{\min\{\mathrm{vol}_d(\kappa_1), \mathrm{vol}_d(\kappa_2)\}}{\mathrm{vol}_{d-1}(f)}$$



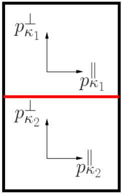


Penalty Parameter

Given $C_{\mathrm{ip}} > 0$, we define

$$\underline{\delta}(\mathbf{u}_h)|_f = C_{\mathrm{ip}} \frac{\mathrm{p}_f^2}{\mathrm{h}_f} \underline{[\![\mathbf{u}_h]\!]} \quad \text{for } f \in \mathcal{F}_h.$$

$$\mathrm{p}_f = \max\{p_{\kappa_1}^{\perp}, p_{\kappa_2}^{\perp}\}$$



Optimal order IPDGFEM

Three key ingredients:

1. Adjoint consistent imposition of the boundary terms present in $\mathcal{N}(\cdot,\cdot)$.

   Lu & Darmofal 2006, Hartmann 2007

2. Adjoint consistent reformulation of the target functional $J(\cdot)$.

   Harriman, Gavaghan, & Süli 2004, Hartmann 2007

3. Definition of the interior penalty terms.
   - Standard SIPG scheme

$$\underline{\delta}(\mathbf{u}_h) \equiv \underline{\delta}^{\mathrm{STSIPG}}(\mathbf{u}_h) = \ C_{\mathrm{ip}} \frac{\mathrm{p}_f^2}{\mathrm{h}_f} \underline{[\![\mathbf{u}_h]\!]}.$$

   - Modified SIPG scheme

$$\underline{\delta}(\mathbf{u}_h) \equiv \underline{\delta}^{\mathrm{SIPG}}(\mathbf{u}_h) = \ C_{\mathrm{ip}} \frac{\mathrm{p}_f^2}{\mathrm{h}_f} \{\!\{ G(\mathbf{u}_h) \}\!\} \underline{[\![\mathbf{u}_h]\!]}.$$

Outline

1. Introduction
   - Second–Order PDEs
2. **Adaptive algorithms**
   - Anisotropic *h*–Refinement
   - Anisotropic *p*–Refinement
3. Numerical Experiments
   - Compressible NS
   - 3D Experiments

Adaptivity

- Goal:

$$|J(u) - J(u_h)| \le \sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa(u_h)| \le \texttt{Tol}.$$

- Automatic refinement algorithm:

  0. `Start with initial (coarse) grid` $\mathcal{T}_h^{(j=0)}$.
  1. `Compute the numerical solution` $u_h^{(j)}$ `on` $\mathcal{T}_h^{(j)}$.
  2. `Compute the local error indicators` η_κ.
  3. `If` $\sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa| \le \texttt{Tol} \rightarrow$ `stop. Otherwise, .`
  4. $j = j + 1$, `and go to step (1).`

R. Verfürth
A Review of a Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques,
B.G. Teubner, Stuttgart, 1996.

Adaptivity

- Goal:

$$|J(u) - J(u_h)| \leq \sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa(u_h)| \leq \mathtt{Tol}.$$

- Automatic refinement algorithm:

0. Start with initial (coarse) grid $\mathcal{T}_h^{(j=0)}$.
1. Compute the numerical solution $u_h^{(j)}$ on $\mathcal{T}_h^{(j)}$.
2. Compute the local error indicators η_κ.
3. If $\sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa| \leq \mathtt{Tol} \rightarrow$ stop. Otherwise, adapt $S_{h,\vec{\mathbf{p}}}$.
4. $j = j + 1$, and go to step (1).

R. Verfürth
A Review of a Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques,
B.G. Teubner, Stuttgart, 1996.

- Goal:

$$|J(u) - J(u_h)| \leq \sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa(u_h)| \leq \mathtt{Tol}.$$

- Automatic refinement algorithm:

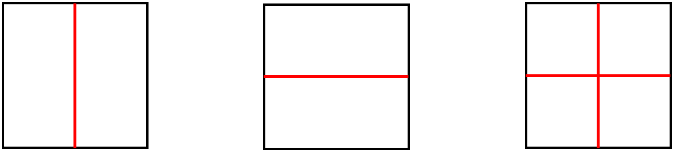
  0. `Start with initial (coarse) grid` $\mathcal{T}_h^{(j=0)}$.
  1. `Compute the numerical solution` $u_h^{(j)}$ `on` $\mathcal{T}_h^{(j)}$.
  2. `Compute the local error indicators` η_κ.
  3. `If` $\sum_{\kappa \in \mathcal{T}_h} |\eta_\kappa| \leq \mathtt{Tol} \rightarrow$ `stop. Otherwise, adapt` $S_{h,\vec{\mathbf{p}}}$.
  4. $j = j + 1$, `and go to step (1).`

R. Verfürth
A Review of a Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques,
B.G. Teubner, Stuttgart, 1996.

Outline

Adaptive Algorithm (h–Adaptivity)

- Elements marked for refinement/derefinement using the fixed–fraction strategy.
- Element refinements.



$\mathcal{E}_1 \equiv \sum_{\kappa \in \mathcal{T}_{h,1}} |\eta_\kappa^{\text{new}}|$ $\mathcal{E}_2 \equiv \sum_{\kappa \in \mathcal{T}_{h,2}} |\eta_\kappa^{\text{new}}|$ $\mathcal{E}_3 \equiv \sum_{\kappa \in \mathcal{T}_{h,3}} |\eta_\kappa^{\text{new}}|$

- Solve local primal and dual problems on elemental patches.
- Boundary data extracted from global primal and dual solutions.

Adaptive Algorithm (h–Adaptivity)

Algorithm 1

Select optimal refinement

$$\max_{i=1,2,3}(|\eta_\kappa^{\text{old}}| - \mathcal{E}_i)/(\#\text{dofs}(\mathcal{T}_{h,i}) - \#\text{dofs}(\mathcal{T}_{h,\kappa})).$$

Algorithm 2

- Prescribe an h–anisotropy parameter $\theta_h > 1$.
- When

$$\frac{\max_{i=1,2}(\mathcal{E}_i)}{\min_{i=1,2}(\mathcal{E}_i)} > \theta_h,$$

  perform refinement in direction with minimal $\mathcal{E}_i$, $i = 1, 2$.
- else perform isotropic h–refinement.

Adaptive Algorithm (h–Adaptivity)

Algorithm 1

Select optimal refinement

$$\max_{i=1,2,3}(|\eta_\kappa^{\text{old}}| - \mathcal{E}_i)/(\#\text{dofs}(\mathcal{T}_{h,i}) - \#\text{dofs}(\mathcal{T}_{h,\kappa})).$$

Algorithm 2

- Prescribe an h–anisotropy parameter $\theta_h > 1$.
- When

$$\frac{\max_{i=1,2}(\mathcal{E}_i)}{\min_{i=1,2}(\mathcal{E}_i)} > \theta_h,$$

  perform refinement in direction with minimal $\mathcal{E}_i$, $i = 1, 2$.
- else perform isotropic h–refinement.

Outline

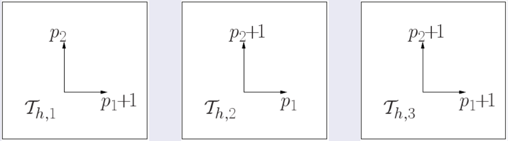
1 Introduction
- Second–Order PDEs

2 **Adaptive algorithms**
- Anisotropic *h*–Refinement
- Anisotropic *p*–Refinement

3 Numerical Experiments
- Compressible NS
- 3D Experiments

Anisotropic p–Refinement

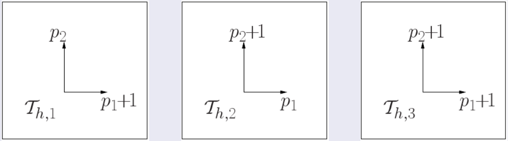
Local Problems



$\mathcal{E}_1 \equiv |\eta_\kappa^{\mathrm{new}}|$ $\quad$ $\mathcal{E}_2 \equiv |\eta_\kappa^{\mathrm{new}}|$ $\quad$ $\mathcal{E}_3 \equiv |\eta_\kappa^{\mathrm{new}}|$

Anisotropic p–Refinement

Local Problems



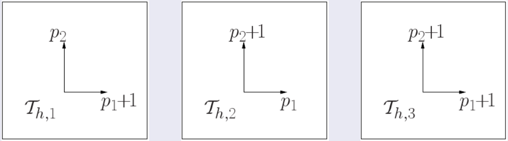
$\mathcal{E}_1 \equiv |\eta_\kappa^{\text{new}}|$ $\quad$ $\mathcal{E}_2 \equiv |\eta_\kappa^{\text{new}}|$ $\quad$ $\mathcal{E}_3 \equiv |\eta_\kappa^{\text{new}}|$

Algorithm 1

Select optimal refinement

$$\max_{i=1,2,3} (|\eta_\kappa^{\text{old}}| - \mathcal{E}_i)/(\#\text{dofs}(\mathcal{T}_{h,i}) - \#\text{dofs}(\mathcal{T}_{h,\kappa})).$$

Local Problems



$\mathcal{E}_1 \equiv |\eta_\kappa^{\text{new}}|$ $\quad$ $\mathcal{E}_2 \equiv |\eta_\kappa^{\text{new}}|$ $\quad$ $\mathcal{E}_3 \equiv |\eta_\kappa^{\text{new}}|$

Algorithm 2

- Prescribe a p–anisotropy parameter $\theta_p > 1$
- When

$$\frac{\max_{i=1,2}(\mathcal{E}_i/(\#\text{dofs}(\mathcal{T}_{h,i}) - \#\text{dofs}(\mathcal{T}_{h,\kappa})))}{\min_{i=1,2}(\mathcal{E}_i/(\#\text{dofs}(\mathcal{T}_{h,i}) - \#\text{dofs}(\mathcal{T}_{h,\kappa})))} > \theta_p,$$

  enrich in polynomial in the direction with minimal $\mathcal{E}_i$, $i = 1, 2$.
- else perform isotropic p–refinement.

Adaptive Algorithm (*hp*–Adaptivity)

- Elements marked for refinement/derefinement using the fixed–fraction strategy.
- Regularity estimation via truncated Legendre series expansions.
  Houston, Senior & Süli 2003, Houston & Süli 2005, Eibner & Melenk 2005.
- If both u and z are deemed to be non-smooth, apply anisotropic h–refinement.
- Else, perform anisotropic p–refinement

Outline

1. **Introduction**
   - Second–Order PDEs
2. **Adaptive algorithms**
   - Anisotropic h–Refinement
   - Anisotropic p–Refinement
3. **Numerical Experiments**
   - Compressible NS
   - 3D Experiments

Outline

NACA0012 Airfoil (Comp. NS)

$\mathrm{Ma} = 0.5$, $\mathrm{Re} = 5000$, $\alpha = 2^\circ$ and adiabatic wall condition.

Drag coefficients:

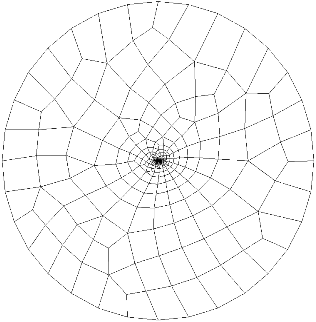
$$J_{c_{\mathrm{dp}}}(\mathbf{u}) = \frac{2}{l\bar{\rho}|\bar{\mathbf{v}}|^2} \int_S p\,(\mathbf{n} \cdot \psi_d)\mathrm{d}s, \qquad J_{c_{\mathrm{df}}}(\mathbf{u}) = \frac{2}{l\bar{\rho}|\bar{\mathbf{v}}|^2} \int_S (\boldsymbol{\tau}\,\mathbf{n}) \cdot \psi_d \mathrm{d}s,$$

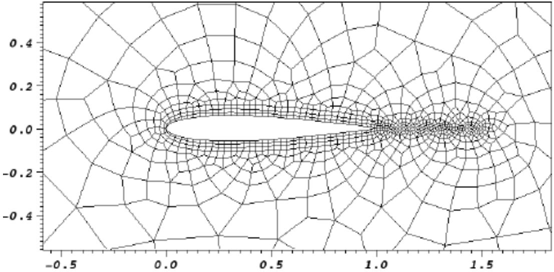
where

$$\psi_d = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

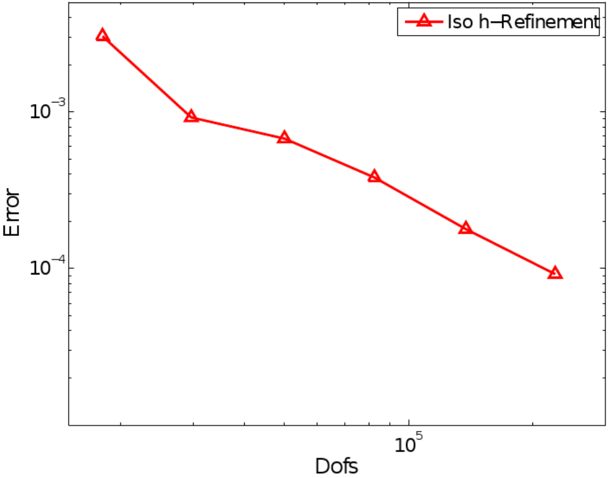
$$\boxed{J_{c_{\mathrm{d}}}(\mathbf{u}) \approx 0.056084.}$$

NACA0012 Airfoil – Unstructured

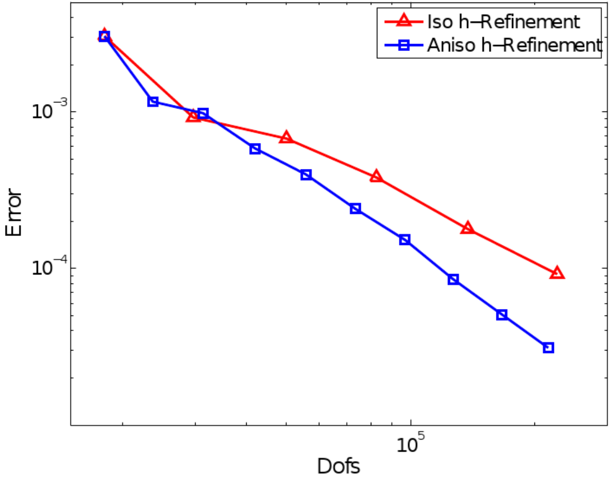




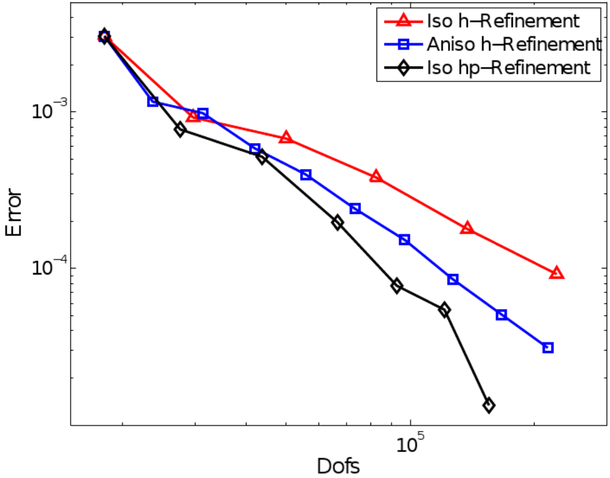
NACA0012 Airfoil – Unstructured



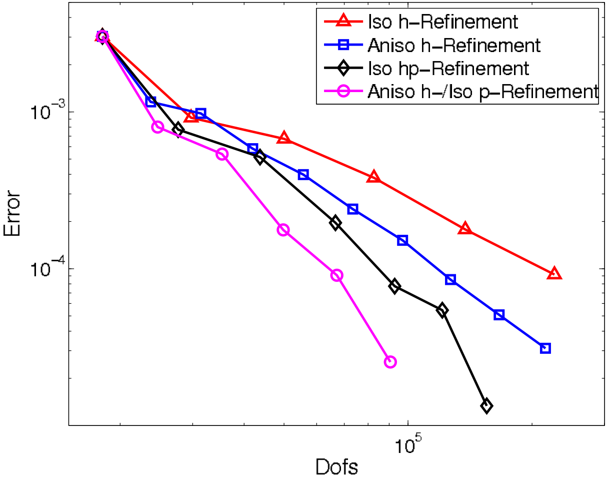
NACA0012 Airfoil – Unstructured



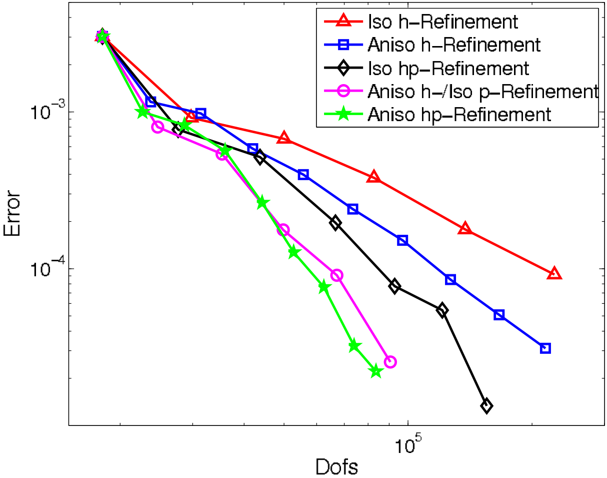
NACA0012 Airfoil – Unstructured



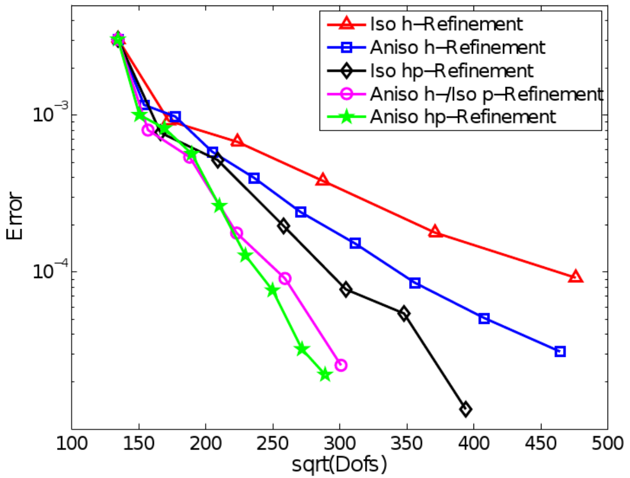
NACA0012 Airfoil – Unstructured



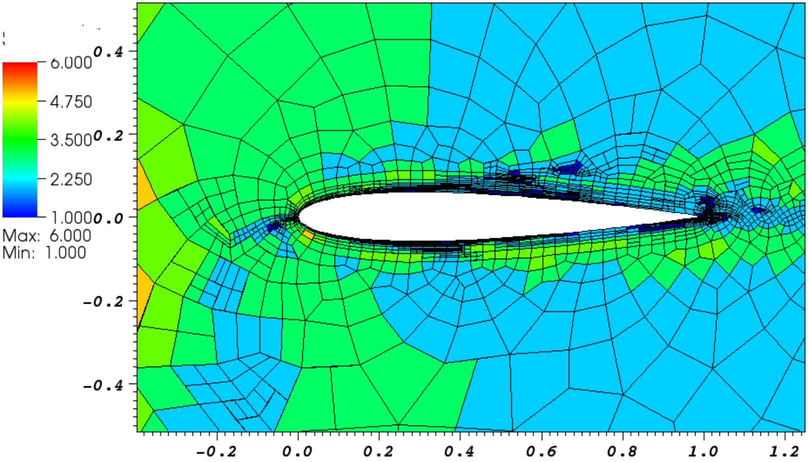
NACA0012 Airfoil – Unstructured



NACA0012 Airfoil – Unstructured

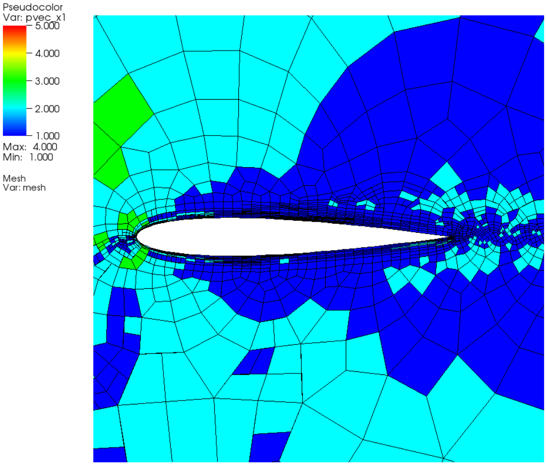


NACA0012 Airfoil – Unstructured



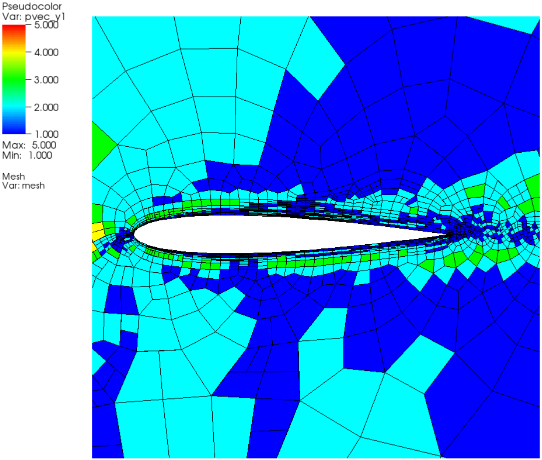
hp–mesh distribution after 6 adaptive (anisotropic *h*–/isotropic *p*–) refinements, with 2835 elements and 118520 degrees of freedom

NACA0012 Airfoil – Unstructured



hp/p_x–mesh distribution after 6 adaptive (anisotropic h–/anisotropic p–) refinements

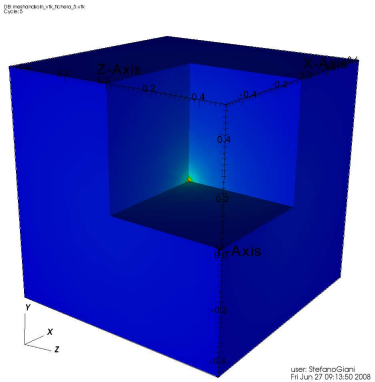
NACA0012 Airfoil – Unstructured



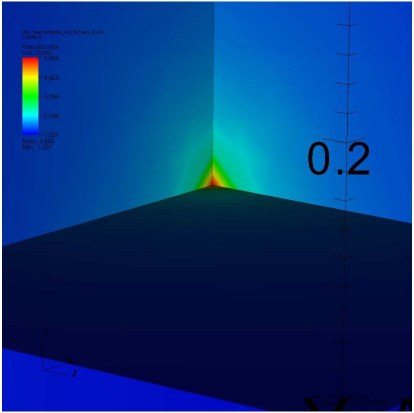
hp/p_y–mesh distribution after 6 adaptive (anisotropic h–/anisotropic p–) refinements

Outline

1. **Introduction**
   - Second–Order PDEs
2. **Adaptive algorithms**
   - Anisotropic h–Refinement
   - Anisotropic p–Refinement
3. **Numerical Experiments**
   - Compressible NS
   - 3D Experiments

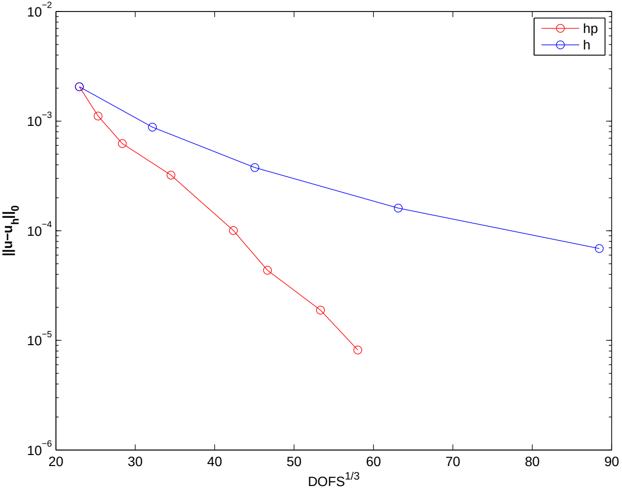


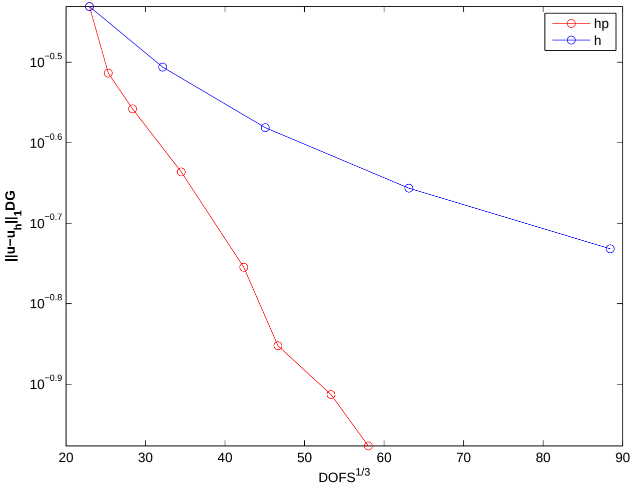
$$\mathbf{u} = |\mathbf{x}|^{-1/4}\,, \qquad \begin{cases} -\Delta \mathbf{u} &= f, \qquad \text{in } \Omega\,, \\ \mathbf{u} &= \mathbf{u}_D, \quad \text{on } \partial\Omega\,. \end{cases}$$

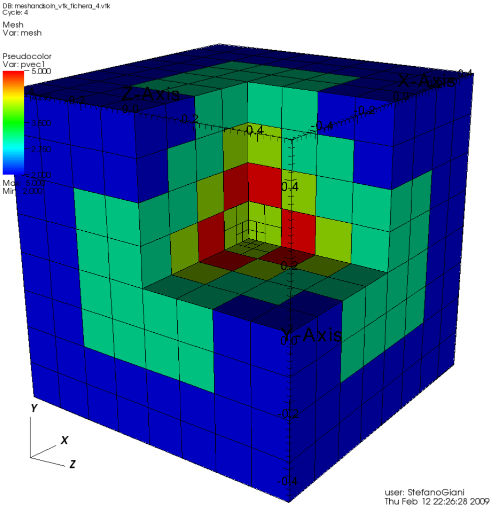


$$\mathbf{u} = |\mathbf{x}|^{-1/4}, \quad \begin{cases} -\Delta \mathbf{u} &= f, \quad \text{in } \Omega\,, \\ \mathbf{u} &= \mathbf{u}_D, \quad \text{on } \partial\Omega\,. \end{cases}$$

Fichera Corner - Isotropic *hp*–Refinement




hp
h
$10^{-0.5}$
$10^{-0.6}$
$10^{-0.7}$
$10^{-0.8}$
$10^{-0.9}$
$\|u-u_h\|_{1}DG$
20
30
40
50
60
70
80
90
$DOFS^{1/3}$


DB: meshandsoln_vtk_fichera_4.vtk
Cycle: 4
Mesh
Var: mesh
Pseudocolor
Var: pvec1
5.000
3.500
2.750
2.000
Max: 5.000
Min: 2.000
Z-Axis
X-Axis
Y-Axis
Y
X
Z
user: StefanoGiani
Thu Feb 12 22:26:28 2009