Mesh Refinement for Quasilinear Two-Grid Discontinuous Galerkin Finite Element Methods with Polygonal Meshes

Scott Congreve

Fakultät für Mathematik, Universität Wien

IACM 19th International Conference on Finite Elements in Flow Problems

Overview

Standard Formulation

Nonlinear Problem

Given a semilinear form $\mathcal{N}(\cdot;\cdot,\cdot)$, find $u \in V$ such that

$$\mathcal{N}(u;u,v) = 0 \qquad \forall v \in V.$$

Standard Formulation

Nonlinear Problem

Given a semilinear form $\mathcal{N}(\cdot;\cdot,\cdot)$, find $u \in V$ such that

$$\mathcal{N}(u;u,v) = 0 \qquad \forall v \in V.$$

Define V_h be the FE space on the mesh, then:

(Standard) Discretization Method

Find $u_h \in V_h$ such that

$$\mathcal{N}_h(u_h;u_h,v_h) = 0 \qquad \forall v_h \in V_h.$$

Create a mesh which is 'coarser' than the original mesh and define V_H as the FE space on this mesh, then:

Two-Grid Discretization Method

Find $u_H \in V_H$ such that

$$\mathcal{N}_H(u_H; u_H, v_H) = 0 \qquad \forall v_H \in V_H,$$

Create a mesh which is 'coarser' than the original mesh and define V_H as the FE space on this mesh, then:

Two-Grid Discretization Method

Find $u_H \in V_H$ such that

$$\mathcal{N}_H(u_H; u_H, v_H) = 0 \qquad \forall v_H \in V_H,$$

find $u_{2G} \in V_h$ such that

$$\mathcal{N}_h(u_H; u_{2G}, v_h) = 0 \qquad \forall v_h \in V_h.$$

Xu 1992, 1994, 1996, Xu & Zhou 1999, Axelsson & Layton 1996, Dawson, Wheeler & Woodward 1998, Utnes 1997, Marion & Xu 1995, Wu & Allen 1999, Bi & Ginting 2007, 2011

Second-Order Quasilinear PDEs

universität wien

Quasilinear Problem

Given $\Omega \subset \mathbb{R}^d$, $d = 2, 3$ and $f \in L^2(\Omega)$, find u such that

$$-\nabla \cdot \{\mu(\boldsymbol{x}, |\nabla u|)\nabla u\} = f \qquad \text{in } \Omega,$$
$$u = 0 \qquad \text{on } \Gamma.$$

Assumption

1. $\mu \in C(\bar{\Omega} \times [0, \infty))$ *and*
2. *there exists positive constants* m_μ *and* M_μ *such that*

$$M_\mu(t - s) \leq \mu(\boldsymbol{x}, t)t - \mu(\boldsymbol{x}, s)s \leq M_\mu(t - s), \quad t \geq s \geq 0, \quad \boldsymbol{x} \in \bar{\Omega}.$$

- $\mathcal{T}_h$ is a mesh consisting of triangles/tetrahedrons elements κ of granularity h, which are an affine map of a reference element $\hat{\kappa}$; i.e., there exists an affine mapping $T_\kappa : \hat{\kappa} \to \kappa$ such that $\kappa = T_\kappa(\hat{\kappa})$.
- Define polynomial degree k_κ for all $\kappa \in \mathcal{T}_h$
- (Fine) *hp*-DG finite element space:

$$V_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{v \in L^2(\Omega) : v|_\kappa \circ T_\kappa \in \mathcal{P}_{k_\kappa}(\hat{\kappa}), \kappa \in \mathcal{T}_h\}.$$

- $\mathcal{F}_h = \mathcal{F}_h^B \cup \mathcal{F}_h^I$ denotes the set of all faces in the mesh $\mathcal{T}_h$.
- Trace operators

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

(Standard) Interior Penalty Method

Find $u_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$ such that

$$A_{hk}(u_{hk}; u_{hk}, v_{hk}) = F_{hk}(v_{hk})$$

for all $v_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

(Standard) Interior Penalty Method

Find $u_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$ such that

$$A_{hk}(u_{hk}; u_{hk}, v_{hk}) = F_{hk}(v_{hk})$$

for all $v_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

$$\begin{aligned} A_{hk}(\psi; u, v) = &\int_\Omega \mu(|\nabla_h \psi|) \nabla_h u \cdot \nabla_h v \, d\boldsymbol{x} + \int_{\mathcal{F}_h} \sigma_{h,k} [\![u]\!] \cdot [\![v]\!] \, ds \\ &- \int_{\mathcal{F}_h} \{\!\{\mu(|\nabla_h \psi|) \nabla_h u\}\!\} \cdot [\![v]\!] \, ds, \\ &+ \theta \int_{\mathcal{F}_h} \{\!\{\mu(h_F^{-1} |[\![\psi]\!]|) \nabla_h v\}\!\} \cdot [\![u]\!] \, ds, \\ F_{hk}(v) = &\int_\Omega f v \, d\boldsymbol{x}. \end{aligned}$$

(Standard) Interior Penalty Method

Find $u_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$ such that

$$A_{hk}(u_{hk}; u_{hk}, v_{hk}) = F_{hk}(v_{hk})$$

for all $v_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

Interior penalty parameter:

$$\sigma_{h,k} = \gamma \frac{k_F^2}{h_F},$$

where $k_F = \max(k_{\kappa_1}, k_{\kappa_2})$ and h_F is the diameter of the face.

(Standard) Interior Penalty Method

Find $u_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$ such that

$$A_{hk}(u_{hk}; u_{hk}, v_{hk}) = F_{hk}(v_{hk})$$

for all $v_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

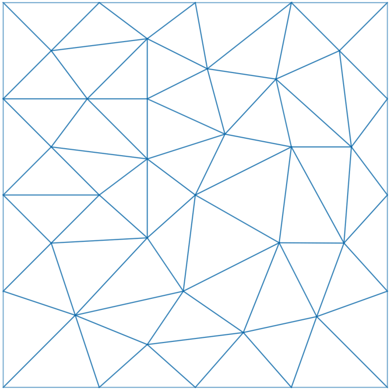
Interior penalty parameter:

$$\sigma_{h,k} = \gamma \frac{k_F^2}{h_F},$$

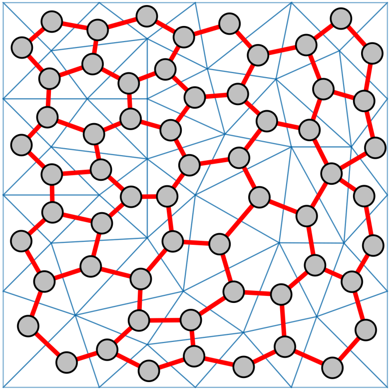
where $k_F = \max(k_{\kappa_1}, k_{\kappa_2})$ and h_F is the diameter of the face.
References:

Bustinza & Gatica 2004, Gatica, Gonzáles & Meddahi 2004, Houston, Robson & Suli 2005, Bustinza, Cockburn & Gatica 2005, Houston, Süli & Wihler 2007, Gudi, Nataraj & Pani 2008

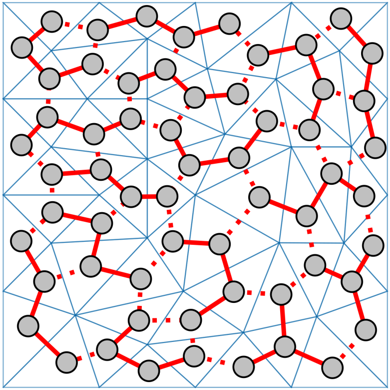
We construct a coarse mesh $\mathcal{T}_H$, consisting of general polygons/polyhedra κ_H by agglomerating elements in the fine mesh $\mathcal{T}_h$.



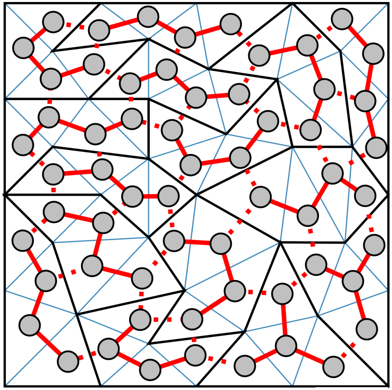
We construct a coarse mesh $\mathcal{T}_H$, consisting of general polygons/polyhedra κ_H by agglomerating elements in the fine mesh $\mathcal{T}_h$.



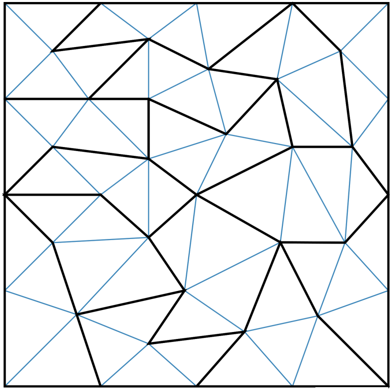
We construct a coarse mesh $\mathcal{T}_H$, consisting of general polygons/polyhedra κ_H by agglomerating elements in the fine mesh $\mathcal{T}_h$.



We construct a coarse mesh $\mathcal{T}_H$, consisting of general polygons/polyhedra κ_H by agglomerating elements in the fine mesh $\mathcal{T}_h$.



We construct a coarse mesh $\mathcal{T}_H$, consisting of general polygons/polyhedra κ_H by agglomerating elements in the fine mesh $\mathcal{T}_h$.



For example, METIS - Karypis & Kumar 1999

- Define $\mathcal{T}_h(\kappa_H) = \{\kappa \in \mathcal{T}_h : \kappa \subseteq \kappa_H\}$ for all $\kappa_H \in \mathcal{T}_H$.
- Define polynomial degree K_{κ_H}, for all $\kappa_h \in \mathcal{T}_H$, such that

$$K_{\kappa_H} \leq k_\kappa \text{ for all } \kappa \in \mathcal{T}_h(\kappa_H).$$

- (Coarse) *hp*-DG finite element space:

$$V_{HK}(\mathcal{T}_H, \boldsymbol{K}) = \{v \in L^2(\Omega) : v|_\kappa \in \mathcal{P}_{K_\kappa}(\kappa), \kappa \in \mathcal{T}_H\}.$$

Two-Grid *hp*-DGFEM

- Define $\mathcal{T}_h(\kappa_H) = \{\kappa \in \mathcal{T}_h : \kappa \subseteq \kappa_H\}$ for all $\kappa_H \in \mathcal{T}_H$.
- Define polynomial degree K_{κ_H}, for all $\kappa_h \in \mathcal{T}_H$, such that

$$K_{\kappa_H} \leq k_\kappa \text{ for all } \kappa \in \mathcal{T}_h(\kappa_H).$$

- (Coarse) *hp*-DG finite element space:

$$V_{HK}(\mathcal{T}_H, \boldsymbol{K}) = \{v \in L^2(\Omega) : v|_\kappa \in \mathcal{P}_{K_\kappa}(\kappa), \kappa \in \mathcal{T}_H\}.$$

- $V_{HK}(\mathcal{T}_H, \boldsymbol{K}) \subseteq V_{hk}(\mathcal{T}_h, \boldsymbol{k})$

- Define $\mathcal{T}_h(\kappa_H) = \{\kappa \in \mathcal{T}_h : \kappa \subseteq \kappa_H\}$ for all $\kappa_H \in \mathcal{T}_H$.
- Define polynomial degree K_{κ_H}, for all $\kappa_h \in \mathcal{T}_H$, such that

$$K_{\kappa_H} \leq k_\kappa \text{ for all } \kappa \in \mathcal{T}_h(\kappa_H).$$

- (Coarse) *hp*-DG finite element space:

$$V_{HK}(\mathcal{T}_H, \boldsymbol{K}) = \{v \in L^2(\Omega) : v|_\kappa \in \mathcal{P}_{K_\kappa}(\kappa), \kappa \in \mathcal{T}_H\}.$$

- $V_{HK}(\mathcal{T}_H, \boldsymbol{K}) \subseteq V_{hk}(\mathcal{T}_h, \boldsymbol{k})$
- We use a different *interior penalty parameter*:

$$\sigma_{h,k} = \gamma \max_{\kappa \in \{\kappa_1, \kappa_2\}} C_{\text{inf}}(k_\kappa, \kappa, F) k_\kappa^2 |F| |\kappa|^{-1},$$

where

$$C_{\text{inf}}(k, \kappa, F) = \min \left\{ \frac{|\kappa|}{\sup_{\kappa_\flat^F \subset \kappa} |\kappa_\flat^F|}, k^{2d} \right\}.$$

$\kappa_\flat^F$ represents a simplex sharing an edge with κ)

[Cangiani, Georgoulis, & Houston]

Two-Grid Approximation

1. Construct coarse and fine FE spaces $V_{HK}(\mathcal{T}_H, \boldsymbol{K})$ and $V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

Two-Grid Approximation

1. Construct coarse and fine FE spaces $V_{HK}(\mathcal{T}_H, \boldsymbol{K})$ and $V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.
2. Compute the coarse grid approximation $u_{HK} \in V_{HK}(\mathcal{T}_H, \boldsymbol{K})$ such that

$$A_{HK}(u_{HK}; u_{HK}, v_{HK}) = F_{HK}(v_{HK})$$

   for all $v_{HK} \in V_{HK}(\mathcal{T}_H, \boldsymbol{K})$.

Two-Grid Approximation

1. Construct coarse and fine FE spaces $V_{HK}(\mathcal{T}_H, \boldsymbol{K})$ and $V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.
2. Compute the coarse grid approximation $u_{HK} \in V_{HK}(\mathcal{T}_H, \boldsymbol{K})$ such that

$$A_{HK}(u_{HK}; u_{HK}, v_{HK}) = F_{HK}(v_{HK})$$

   for all $v_{HK} \in V_{HK}(\mathcal{T}_H, \boldsymbol{K})$.
3. Determine the fine grid approximation $u_{2G} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$ such that

$$A_{hk}(u_{HK}; u_{2G}, v_{hk}) = F_{hk}(v_{hk})$$

   for all $v_{hk} \in V_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

[C., Houston, & Wihler 2013]

Lemma (Standard Quasilinear DGFEM)

The following bound holds:

$$\|u - u_{hk}\|_{hk}^2 \leq C_1 \sum_{\kappa \in \mathcal{T}_h} \eta_\kappa^2 \qquad .$$

Here the local error indicators η_κ are defined, for all $\kappa \in \mathcal{T}_h$, as

$$\begin{aligned}\eta_\kappa^2 &= h_\kappa^2 k_\kappa^{-2} \|f + \nabla \cdot \{\mu(|\nabla u_{hk}|)\nabla u_{hk}\}\|_{L^2(\kappa)}^2 \\ &\quad + h_\kappa k_\kappa^{-1} \|[\![\mu(|\nabla u_{hk}|)\nabla u_{hk}]\!]\|_{L^2(\partial\kappa\setminus\Gamma)}^2 + \gamma^2 k_\kappa^3 h_\kappa^{-1} \|[\![u_{hk}]\!]\|_{L^2(\partial\kappa)}^2\end{aligned}$$

Proof.

See Houston, Süli & Wihler 2008. □

Lemma (Two-Grid Quasilinear Approximation)

The following bound holds:

$$\|u - u_{2G}\|_{hk}^2 \leq C_2 \sum_{\kappa \in \mathcal{T}_h} \left(\eta_\kappa^2 + \xi_\kappa^2 \right).$$

Here the local error indicators η_κ are defined, for all $\kappa \in \mathcal{T}_h$, as

$$\eta_\kappa^2 = h_\kappa^2 k_\kappa^{-2} \|f + \nabla \cdot \{\mu(|\nabla u_{HK}|)\nabla u_{2G}\}\|_{L^2(\kappa)}^2 + h_\kappa k_\kappa^{-1} \|[\![\mu(|\nabla u_{HK}|)\nabla u_{2G}]\!]\|_{L^2(\partial\kappa\setminus\Gamma)}^2 + \gamma^2 k_\kappa^3 h_\kappa^{-1} \|[\![u_{2G}]\!]\|_{L^2(\partial\kappa)}^2$$

and the local two-grid error indicators are defined, for all $\kappa \in \mathcal{T}_h$, as

$$\xi_\kappa^2 = \|(\mu(|\nabla u_{HK}|) - \mu(|\nabla u_{2G}|))\nabla u_{2G}\|_{L^2(\kappa)}^2.$$

Proof.

See C., Houston, & Wihler 2013. □

Two-Grid Adaptivity

1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.

Two-Grid Adaptivity

1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.
2. Compute the coarse grid approximation and two-grid solution.

Two-Grid Adaptivity

1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.
2. Compute the coarse grid approximation and two-grid solution.
3. Select elements for refinement based on η_κ and ξ_κ:
   3.1 Use $\eta_K + \xi_K$ to determine set $\mathfrak{R}(\mathcal{T}_h) \subseteq \mathcal{T}_h$ of elements to refine.

Two-Grid Adaptivity

1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.
2. Compute the coarse grid approximation and two-grid solution.
3. Select elements for refinement based on η_κ and ξ_κ:
   3.1 Use $\eta_K + \xi_K$ to determine set $\mathfrak{R}(\mathcal{T}_h) \subseteq \mathcal{T}_h$ of elements to refine.
   3.2 Choose fine or coarse mesh refinement. For all $\kappa \in \mathfrak{R}(\mathcal{T}_h)$
      - if $\lambda_F \xi_\kappa \leq \eta_\kappa$ refine the fine element κ, and
      - if $\lambda_C \eta_\kappa \leq \xi_\kappa$ refine the coarse element $\kappa_H \in \mathcal{T}_H$, where $\kappa \in \mathcal{T}_h(\kappa_H)$.

Two-Grid Adaptivity

1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.
2. Compute the coarse grid approximation and two-grid solution.
3. Select elements for refinement based on η_κ and ξ_κ:
   3.1 Use $\eta_K + \xi_K$ to determine set $\mathfrak{R}(\mathcal{T}_h) \subseteq \mathcal{T}_h$ of elements to refine.
   3.2 Choose fine or coarse mesh refinement. For all $\kappa \in \mathfrak{R}(\mathcal{T}_h)$
      - if $\lambda_F \xi_\kappa \leq \eta_\kappa$ refine the fine element κ, and
      - if $\lambda_C \eta_\kappa \leq \xi_\kappa$ refine the coarse element $\kappa_H \in \mathcal{T}_H$, where $\kappa \in \mathcal{T}_h(\kappa_H)$.
4. Perform *h*-/*hp*-mesh refinement of the fine space.

universität wien

Two-Grid Adaptivity

1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.
2. Compute the coarse grid approximation and two-grid solution.
3. Select elements for refinement based on η_κ and ξ_κ:
   - 3.1 Use $\eta_K + \xi_K$ to determine set $\mathfrak{R}(\mathcal{T}_h) \subseteq \mathcal{T}_h$ of elements to refine.
   - 3.2 Choose fine or coarse mesh refinement. For all $\kappa \in \mathfrak{R}(\mathcal{T}_h)$
     - if $\lambda_F \xi_\kappa \le \eta_\kappa$ refine the fine element κ, and
     - if $\lambda_C \eta_\kappa \le \xi_\kappa$ refine the coarse element $\kappa_H \in \mathcal{T}_H$, where $\kappa \in \mathcal{T}_h(\kappa_H)$.
4. Perform *h*-/*hp*-mesh refinement of the fine space.
5. Select *h*- or *p*-refinement for each coarse element to refine.

Two-Grid Adaptivity

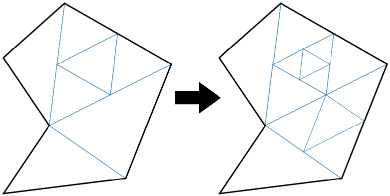
1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.
2. Compute the coarse grid approximation and two-grid solution.
3. Select elements for refinement based on η_κ and ξ_κ:
   3.1 Use $\eta_K + \xi_K$ to determine set $\mathfrak{R}(\mathcal{T}_h) \subseteq \mathcal{T}_h$ of elements to refine.
   3.2 Choose fine or coarse mesh refinement. For all $\kappa \in \mathfrak{R}(\mathcal{T}_h)$
      - if $\lambda_F \xi_\kappa \leq \eta_\kappa$ refine the fine element κ, and
      - if $\lambda_C \eta_\kappa \leq \xi_\kappa$ refine the coarse element $\kappa_H \in \mathcal{T}_H$, where $\kappa \in \mathcal{T}_h(\kappa_H)$.
4. Perform *h*-/*hp*-mesh refinement of the fine space.
5. Select *h*- or *p*-refinement for each coarse element to refine.
6. Perform *h*-/*hp*-refinement of the coarse space.

Two-Grid Adaptivity

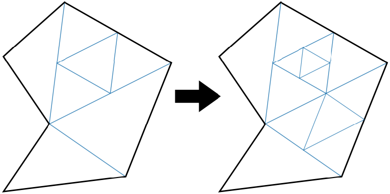
1. Construct initial coarse and fine FE spaces, with coarse mesh created by agglomerating the fine mesh.
2. Compute the coarse grid approximation and two-grid solution.
3. Select elements for refinement based on η_κ and ξ_κ:
   3.1 Use $\eta_K + \xi_K$ to determine set $\mathfrak{R}(\mathcal{T}_h) \subseteq \mathcal{T}_h$ of elements to refine.
   3.2 Choose fine or coarse mesh refinement. For all $\kappa \in \mathfrak{R}(\mathcal{T}_h)$
      - if $\lambda_F \xi_\kappa \leq \eta_\kappa$ refine the fine element κ, and
      - if $\lambda_C \eta_\kappa \leq \xi_\kappa$ refine the coarse element $\kappa_H \in \mathcal{T}_H$, where $\kappa \in \mathcal{T}_h(\kappa_H)$.
4. Perform *h*-/*hp*-mesh refinement of the fine space.
5. Select *h*- or *p*-refinement for each coarse element to refine.
6. Perform *h*-/*hp*-refinement of the coarse space.
7. Goto 2.

The constants λ_F and λ_C are steering parameters.

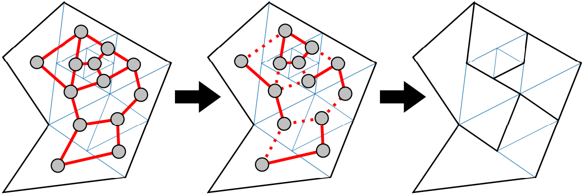
Fine Element Refine:



Fine Element Refine:



Coarse Element Refine — Partition patch of fine elements into 2^d elements



Similar to [Collis & Houston, 2016]

Coarse Element h-Refinement

Using a standard graph partition algorithm will attempt to create agglomerated elements with the same number of *child* fine elements, minimising the number of edge cuts.

However, we have information about the error for each fine element — can we distribute the agglomeration using this information?

Coarse Element h-Refinement

Using a standard graph partition algorithm will attempt to create agglomerated elements with the same number of *child* fine elements, minimising the number of edge cuts.

However, we have information about the error for each fine element — can we distribute the agglomeration using this information?

Possible to assign *weights* to each vertex and use a graph partitioning algorithm that balances these weights, rather than the number of elements. [Karypis & Kumar 1998]

Coarse Element h-Refinement

Using a standard graph partition algorithm will attempt to create agglomerated elements with the same number of *child* fine elements, minimising the number of edge cuts.

However, we have information about the error for each fine element — can we distribute the agglomeration using this information?

Possible to assign *weights* to each vertex and use a graph partitioning algorithm that balances these weights, rather than the number of elements. [Karypis & Kumar 1998]

Two options for weights:

- Two-Grid Error: ξ_κ
- Total Error: $\eta_\kappa + \xi_\kappa$

Using a standard graph partition algorithm will attempt to create agglomerated elements with the same number of *child* fine elements, minimising the number of edge cuts.

However, we have information about the error for each fine element — can we distribute the agglomeration using this information?

Possible to assign *weights* to each vertex and use a graph partitioning algorithm that balances these weights, rather than the number of elements. [Karypis & Kumar 1998]

Two options for weights:

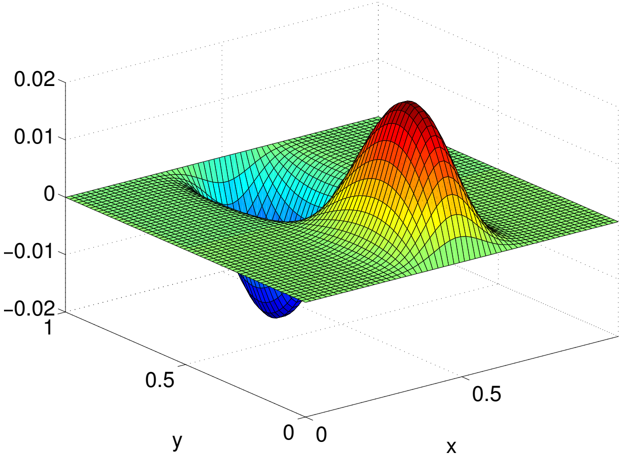
- Two-Grid Error: ξ_κ
- Total Error: $\eta_\kappa + \xi_\kappa$

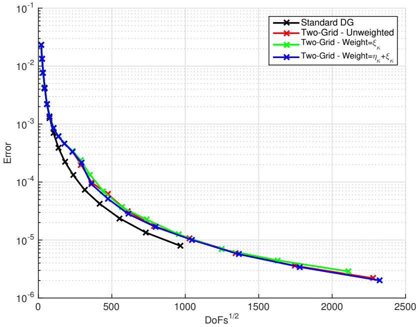
The coarse element refinement uses the fine elements *after* refinement; therefore, we divide the weight from a fine element marked for refinement equally between the new fine elements.

Quasilinear PDE: Smooth Solution

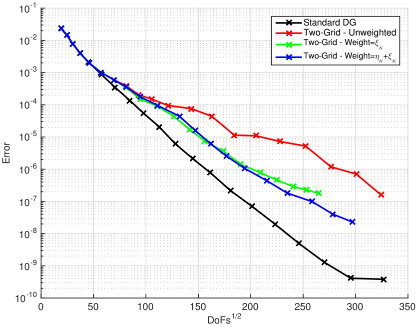
We let $\Omega = (0,1)^2, \mu(\boldsymbol{x}, |\nabla u|) = 2 + \frac{1}{1+|\nabla u|^2}$ and select f so that

$$u(x,y) = x(1-x)y(1-y)(1-2y)\mathrm{e}^{-20(2x-1)^2}.$$



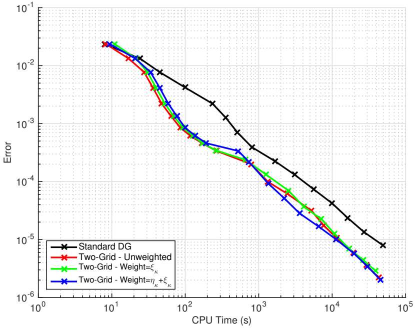


Error vs. #DoFs
(*h*-refinement)

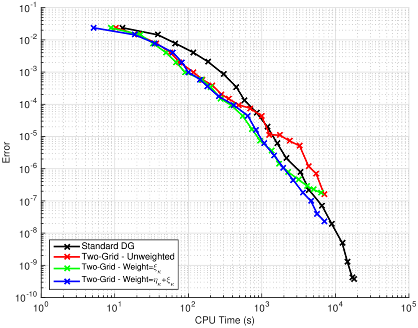


Error vs. #DoFs
(*hp*-refinement)

Quasilinear PDE: Smooth Solution

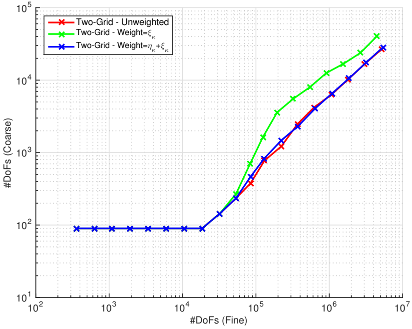


Error vs. Computation Time (*h*-refinement)

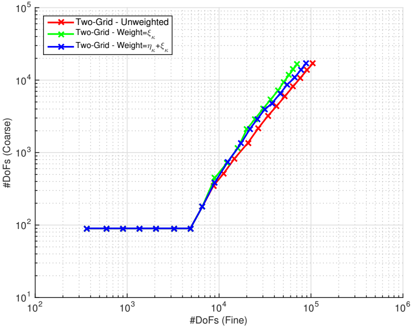


Error vs. Computation Time (*hp*-refinement)

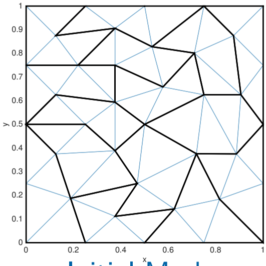
Quasilinear PDE: Smooth Solution



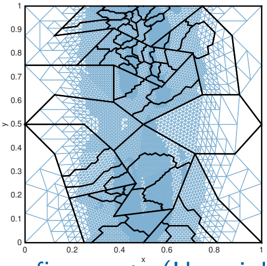
#DoFs: Fine vs. Coarse
(*h*-refinement)



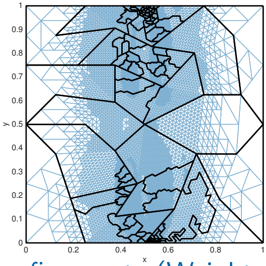
#DoFs: Fine vs. Coarse
(*hp*-refinement)



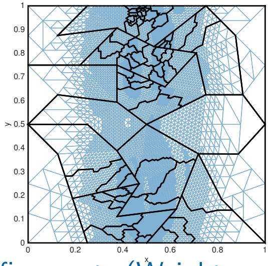
Initial Mesh



10 h-refinements (Unweighted)

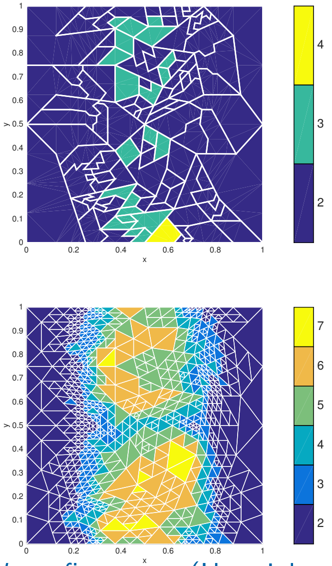


10 h-refinements (Weight $= \xi_\kappa$)

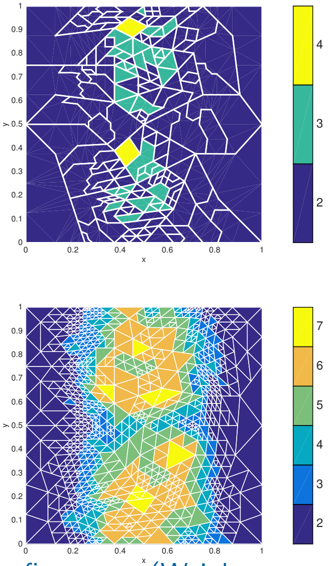


10 h-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

Quasilinear PDE: Smooth Solution



10 *hp*-refinements (Unweighted)

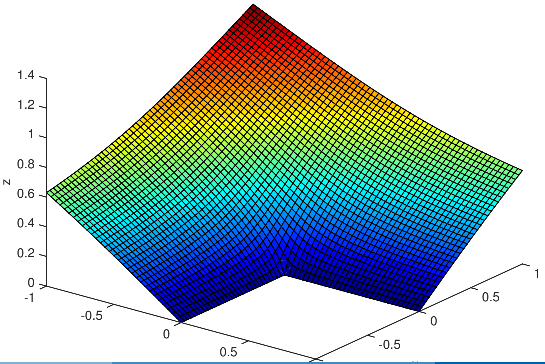


10 *hp*-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

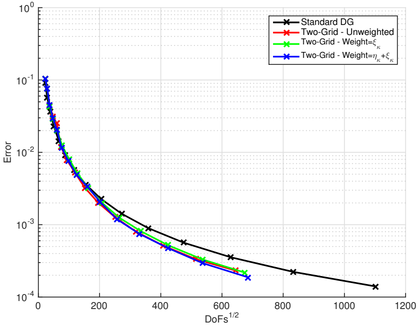
We let $\Omega = (-1,1)^2 \setminus [0,1) \times (-1,0]$, $\mu(\boldsymbol{x}, |\nabla u|) = 1 + \mathrm{e}^{-|\nabla u|^2}$ and select f so that

$$u(r,\phi) = r^{2/3} \sin\left(\frac{2}{3}\varphi\right).$$

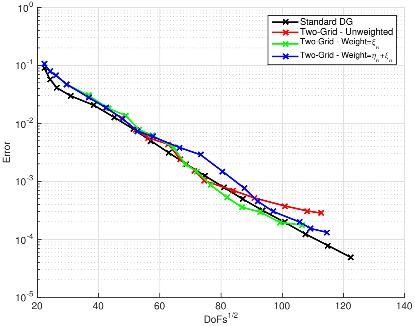
Note that u in analytic in $\overline{\Omega} \setminus \{\mathbf{0}\}$, but ∇u is singular at the origin.



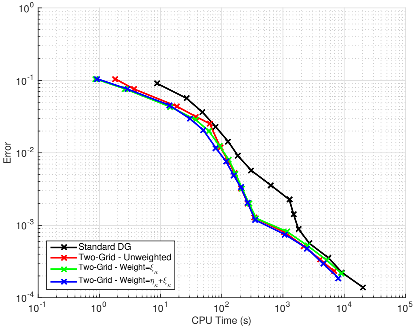
Quasilinear PDE: Singular Solution



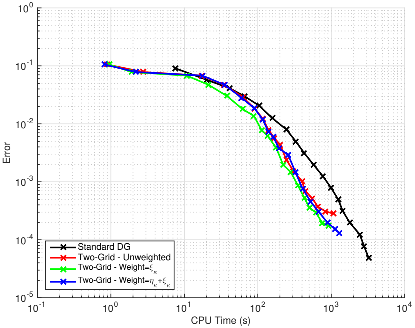
Error vs. #DoFs
(h-refinement)



Error vs. #DoFs
(hp-refinement)

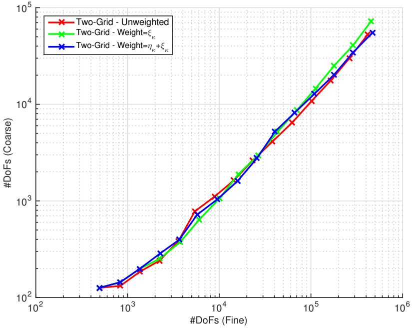


Error vs. Computation Time
(*h*-refinement)

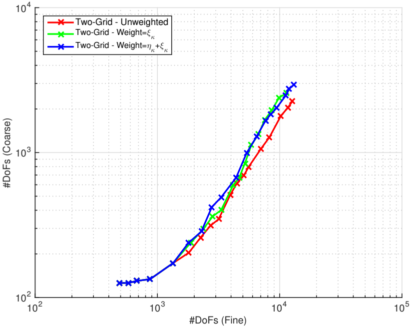


Error vs. Computation Time
(*hp*-refinement)

Quasilinear PDE: Singular Solution

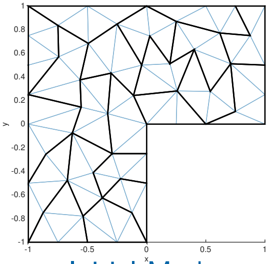


#DoFs: Fine vs. Coarse
(*h*-refinement)

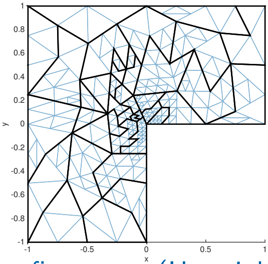


#DoFs: Fine vs. Coarse
(*hp*-refinement)

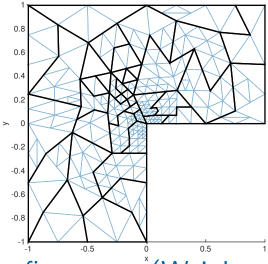
Quasilinear PDE: Singular Solution



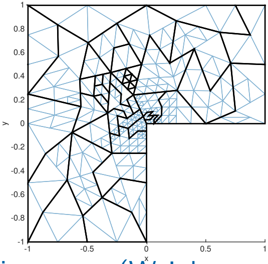
Initial Mesh



3 h-refinements (Unweighted)

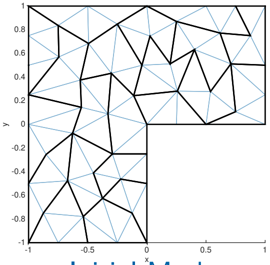


3 h-refinements (Weight $= \xi_\kappa$)

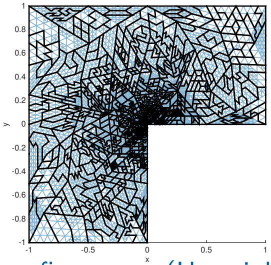


3 h-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

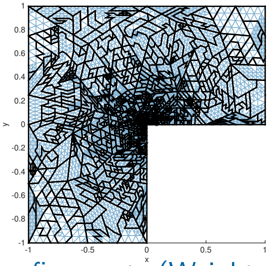
Quasilinear PDE: Singular Solution



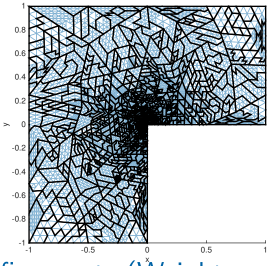
Initial Mesh



10 h-refinements (Unweighted)

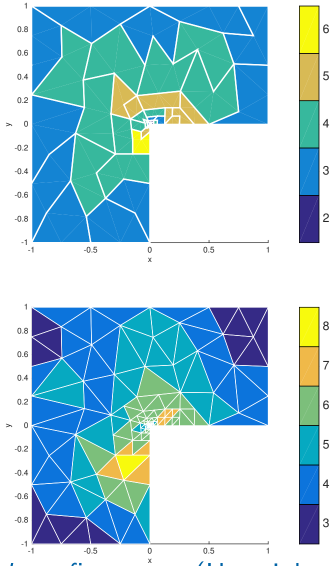


10 h-refinements (Weight $= \xi_\kappa$)

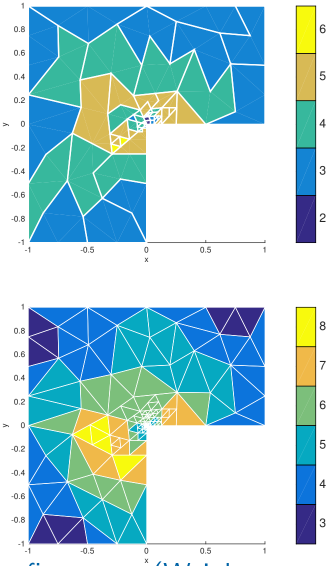


10 h-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

Quasilinear PDE: Singular Solution



10 *hp*-refinements (Unweighted)



10 *hp*-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

Non-Newtonian Fluid Problem

Given $\Omega \subset \mathbb{R}^d, d = 2, 3$ and $\boldsymbol{f} \in [L^2(\Omega)]^d$, find $(\boldsymbol{u}, p)$ such that

$$\begin{aligned} -\nabla \cdot \{\mu(\boldsymbol{x}, |\underline{e}(\boldsymbol{u})|)\underline{e}(\boldsymbol{u})\} + \nabla p &= \boldsymbol{f} && \text{in } \Omega, \\ \nabla \cdot \boldsymbol{u} &= 0 && \text{in } \Omega, \\ \boldsymbol{u} &= \boldsymbol{0} && \text{on } \Gamma, \end{aligned}$$

where $\underline{e}(\boldsymbol{u})$ is the *symmetric $d \times d$ strain tensor* defined by

$$e_{ij}(\boldsymbol{u}) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right).$$

Assumption

$\mu \in C(\bar{\Omega} \times [0, \infty))$ *and there exists positive constants* m_μ *and* M_μ *s.t.*

$$M_\mu(t - s) \leq \mu(\boldsymbol{x}, t)t - \mu(\boldsymbol{x}, s)s \leq M_\mu(t - s), \quad t \geq s \geq 0, \quad \boldsymbol{x} \in \bar{\Omega}.$$

hp-DGFEM

- Fine *hp*-DG finite element spaces:

$$\boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{\boldsymbol{v} \in L^2(\Omega)^d : \boldsymbol{v}|_{\kappa_H} \circ T_\kappa \in \left[\mathcal{P}_{k_\kappa}(\hat{\kappa})\right]^d, \kappa \in \mathcal{T}_h\},$$
$$Q_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{q \in L^2_0(\Omega) : q|_{\kappa_H} \circ T_\kappa \in \mathcal{P}_{k_\kappa - 1}(\hat{\kappa}), \kappa \in \mathcal{T}_h\}.$$

- Fine *hp*-DG finite element spaces:

$$\boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{\boldsymbol{v} \in L^2(\Omega)^d : \boldsymbol{v}|_{\kappa_H} \circ T_\kappa \in \left[\mathcal{P}_{k_\kappa}(\hat{\kappa})\right]^d, \kappa \in \mathcal{T}_h\},$$
$$Q_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{q \in L^2_0(\Omega) : q|_{\kappa_H} \circ T_\kappa \in \mathcal{P}_{k_\kappa - 1}(\hat{\kappa}), \kappa \in \mathcal{T}_h\}.$$

- Coarse *hp*-DG finite element spaces:

$$\boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{\boldsymbol{v} \in L^2(\Omega)^d : \boldsymbol{v}|_{\kappa} \in \left[\mathcal{P}_{K_\kappa}(\kappa)\right]^d, \kappa \in \mathcal{T}_H\},$$
$$Q_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{q \in L^2_0(\Omega) : q|_{\kappa} \in \mathcal{P}_{K_\kappa - 1}(\kappa), \kappa \in \mathcal{T}_H\}.$$

- Fine *hp*-DG finite element spaces:

$$\boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{\boldsymbol{v} \in L^2(\Omega)^d : \boldsymbol{v}|_{\kappa_H} \circ T_\kappa \in \left[\mathcal{P}_{k_\kappa}(\hat{\kappa})\right]^d, \kappa \in \mathcal{T}_h\},$$
$$Q_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{q \in L_0^2(\Omega) : q|_{\kappa_H} \circ T_\kappa \in \mathcal{P}_{k_\kappa - 1}(\hat{\kappa}), \kappa \in \mathcal{T}_h\}.$$

- Coarse *hp*-DG finite element spaces:

$$\boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{\boldsymbol{v} \in L^2(\Omega)^d : \boldsymbol{v}|_{\kappa} \in \left[\mathcal{P}_{K_\kappa}(\kappa)\right]^d, \kappa \in \mathcal{T}_H\},$$
$$Q_{hk}(\mathcal{T}_h, \boldsymbol{k}) = \{q \in L_0^2(\Omega) : q|_{\kappa} \in \mathcal{P}_{K_\kappa - 1}(\kappa), \kappa \in \mathcal{T}_H\}.$$

- Jump operator: $\underline{[\![\boldsymbol{v}]\!]} = \boldsymbol{v}^+ \otimes \boldsymbol{n}^+ + \boldsymbol{v}^- \otimes \boldsymbol{n}^-$

(Standard) Interior Penalty Method

Find $(\boldsymbol{u}_{hk}, p_{hk}) \in \boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) \times Q_{hk}(\mathcal{T}_h, \boldsymbol{k})$ such that

$$\begin{aligned} A_{hk}(\boldsymbol{u}_{hk}; \boldsymbol{u}_{hk}, \boldsymbol{v}_{hk}) + B_{hk}(\boldsymbol{v}_{hk}, p_{hk}) &= F_{hk}(\boldsymbol{v}_{hk}) \\ -B_{hk}(\boldsymbol{u}_{hk}, q_{hk}) &= 0 \end{aligned}$$

for all $(\boldsymbol{v}_{hk}, q_{hk}) \in \boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) \times Q_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

$$A_{hk}(\psi; \boldsymbol{u}, \boldsymbol{v}) = \int_\Omega \mu(|\underline{e}_h(\psi)|)\underline{e}_h(\boldsymbol{u}) : \underline{e}_h(\boldsymbol{v})\, d\boldsymbol{x}$$
$$- \int_{\mathcal{F}_h} \{\!\{\mu(|\underline{e}(\psi)|)\underline{e}(\boldsymbol{u})\}\!\} : \underline{[\![\boldsymbol{v}]\!]}\, ds$$
$$+ \theta \int_{\mathcal{F}_h} \{\!\{\mu(h_F^{-1}|\underline{[\![\psi]\!]}|)\underline{e}(\boldsymbol{v})\}\!\} : \underline{[\![\boldsymbol{u}]\!]}\, ds$$
$$+ \int_{\mathcal{F}_h} \sigma_{h,k}\underline{[\![\boldsymbol{u}]\!]} : \underline{[\![\boldsymbol{v}]\!]}\, ds,$$
$$B_{hk}(\boldsymbol{v}, q) = -\sum_{\kappa\in\mathcal{T}_h} \int_\kappa q\nabla\cdot\boldsymbol{v}\, d\boldsymbol{x} + \sum_{F\in\mathcal{F}_h} \int_F \{\!\{q\}\!\}[\![\boldsymbol{v}]\!]\, ds,$$
$$F_{hk}(v) = \sum_{\kappa\in\mathcal{T}_h} \int_\kappa \boldsymbol{f}\cdot\boldsymbol{v}\, d\boldsymbol{x}.$$

universität wien

Two-Grid Approximation

1. Construct $\boldsymbol{V}_{HK}(\mathcal{T}_H, \boldsymbol{K})$, $Q_{HK}(\mathcal{T}_H, \boldsymbol{K})$, $\boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k})$, and $Q_{hk}(\mathcal{T}_h, \boldsymbol{k})$.
2. Compute $(\boldsymbol{u}_{hk}, p_{HK}) \in \boldsymbol{V}_{HK}(\mathcal{T}_H, \boldsymbol{K}) \times Q_{HK}(\mathcal{T}_H, \boldsymbol{K})$ such that

$$\begin{aligned} A_{HK}(\boldsymbol{u}_{HK}; \boldsymbol{u}_{HK}, v_{HK}) + B_{HK}(\boldsymbol{v}_{HK}, p_{HK}) &= F_{HK}(\boldsymbol{v}_{HK}), \\ -B_{HK}(\boldsymbol{u}_{HK}, q_{HK}) &= 0 \end{aligned}$$

for all $(\boldsymbol{v}_{HK}, q_{HK}) \in \boldsymbol{V}_{HK}(\mathcal{T}_H, \boldsymbol{K}) \times Q_{HK}(\mathcal{T}_H, \boldsymbol{K})$.

Two-Grid Approximation

1. Construct $\boldsymbol{V}_{HK}(\mathcal{T}_H, \boldsymbol{K})$, $Q_{HK}(\mathcal{T}_H, \boldsymbol{K})$, $\boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k})$, and $Q_{hk}(\mathcal{T}_h, \boldsymbol{k})$.
2. Compute $(\boldsymbol{u}_{hk}, p_{HK}) \in \boldsymbol{V}_{HK}(\mathcal{T}_H, \boldsymbol{K}) \times Q_{HK}(\mathcal{T}_H, \boldsymbol{K})$ such that

$$\begin{aligned} A_{HK}(\boldsymbol{u}_{HK}; \boldsymbol{u}_{HK}, v_{HK}) + B_{HK}(\boldsymbol{v}_{HK}, p_{HK}) &= F_{HK}(\boldsymbol{v}_{HK}), \\ -B_{HK}(\boldsymbol{u}_{HK}, q_{HK}) &= 0 \end{aligned}$$

   for all $(\boldsymbol{v}_{HK}, q_{HK}) \in \boldsymbol{V}_{HK}(\mathcal{T}_H, \boldsymbol{K}) \times Q_{HK}(\mathcal{T}_H, \boldsymbol{K})$.
3. Determine $(\boldsymbol{u}_{2G}, p_{2G}) \in \boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) \times Q_{hk}(\mathcal{T}_h, \boldsymbol{k})$ such that

$$\begin{aligned} A_{hk}(u_{HK}; \boldsymbol{u}_{2G}, \boldsymbol{v}_{hk}) + B_{hk}(\boldsymbol{v}_{hk}, p_{2G}) &= F_{hk}(v_{hk}), \\ -B_{hk}(\boldsymbol{u}_{2G}, q_{hk}) &= 0 \end{aligned}$$

   for all $(\boldsymbol{v}_{hk}, q_{hk}) \in \boldsymbol{V}_{hk}(\mathcal{T}_h, \boldsymbol{k}) \times Q_{hk}(\mathcal{T}_h, \boldsymbol{k})$.

[C., & Houston 2014]

Lemma (Standard Non-Newtonian Fluid DGFEM)

The following bound holds:

$$\|(\boldsymbol{u}-\boldsymbol{u}_{hk}, p - p_{hk})\|^2_{DG} \leq C_3 \sum_{\kappa\in\mathcal{T}_h} \eta_\kappa^2 \qquad .$$

Here the local error indicators η_κ *are defined, for all* $\kappa \in \mathcal{T}_h$, *as*

$$\eta_\kappa^2 = h_\kappa^2 k_\kappa^{-2} \|\boldsymbol{f} + \nabla\cdot\{\mu(|\underline{e}(\boldsymbol{u}_{hk})|)\underline{e}(\boldsymbol{u}_{hk})\} - \nabla p_{hk}\|^2_{L^2(\kappa)} + \|\nabla\cdot\boldsymbol{u}_{hk}\|^2_{L^2(\kappa)}$$
$$+ h_\kappa k_\kappa^{-1} \|[\![p_{hk}]\!] - [\![\mu(|\underline{e}(\boldsymbol{u}_{hk})|)\underline{e}(\boldsymbol{u}_{hk})]\!]\|^2_{L^2(\partial\kappa\setminus\Gamma)} + \gamma^2 k_\kappa^3 h_\kappa^{-1} \|\underline{[\![\boldsymbol{u}_{hk}]\!]}\|^2_{L^2(\partial\kappa)}$$

Proof.

See C., Houston, Süli & Wihler 2013. □

Lemma (Two-Grid Non-Newtonian Fluid Approximation)

The following bound holds:

$$\|(\boldsymbol{u} - \boldsymbol{u}_{2G}, p - p_{2G})\|_{DG}^2 \leq C_4 \sum_{\kappa\in\mathcal{T}_h} \left(\eta_\kappa^2 + \xi_\kappa^2\right).$$

Here the local error indicators η_κ are defined, for all $\kappa \in \mathcal{T}_h$, as

$$\eta_\kappa^2 = h_\kappa^2 k_\kappa^{-2}\|\boldsymbol{f} + \nabla\cdot\{\mu(|\underline{e}(\boldsymbol{u}_{HK})|)\underline{e}(\boldsymbol{u}_{2G})\} - \nabla p_{2G}\|_{L^2(\kappa)}^2 + \|\nabla\cdot\boldsymbol{u}_{2G}\|_{L^2(\kappa)}^2$$
$$+ h_\kappa k_\kappa^{-1}\|[\![p_{2G}]\!] - [\![\mu(|\underline{e}(\boldsymbol{u}_{HK})|)\underline{e}(\boldsymbol{u}_{2G})]\!]\|_{L^2(\partial\kappa\setminus\Gamma)}^2 + \gamma^2 k_\kappa^3 h_\kappa^{-1}\|\underline{[\![\boldsymbol{u}_{2G}]\!]}\|_{L^2(\partial\kappa)}^2$$

and the local two-grid error indicators are defined, for all $\kappa \in \mathcal{T}_h$, as

$$\xi_\kappa^2 = \|(\mu(|\underline{e}(\boldsymbol{u}_{HK})|) - \mu(|\underline{e}(\boldsymbol{u}_{2G})|))\nabla\boldsymbol{u}_{2G}\|_{L^2(\kappa)}^2.$$

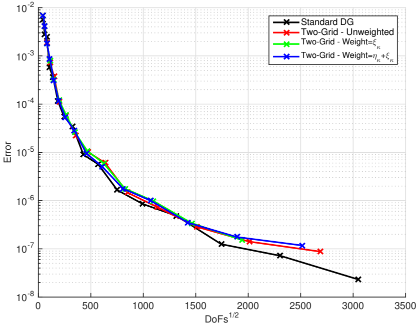
Proof.

See C., & Houston 2014. □

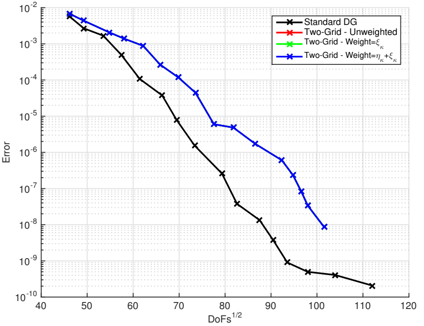
We let $\Omega = (-1,1)^2 \setminus [0,1) \times (-1,0], \mu(\boldsymbol{x}, |\underline{e}(u)|) = 2 + \frac{1}{1+|\underline{e}(\boldsymbol{u})|^2}$ and select f so that

$$\boldsymbol{u}(x,y) = \begin{pmatrix} -\mathrm{e}^x(y\cos y + \sin y) \\ e^x y \sin y \end{pmatrix}$$

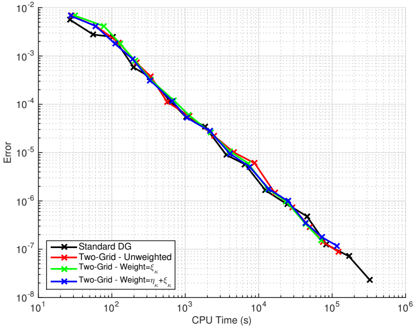
$$p(x,y) = 2\mathrm{e}^x \sin y - \frac{2}{3}(1-e)(cos1 - 1).$$



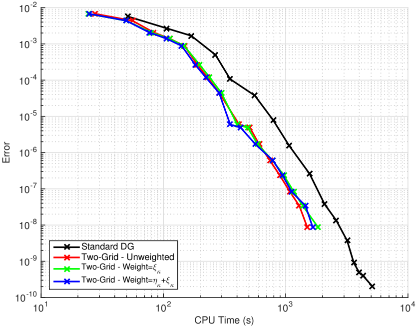
Error vs. #DoFs
(h-refinement)



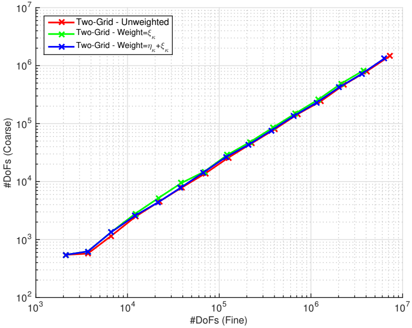
Error vs. #DoFs
(hp-refinement)



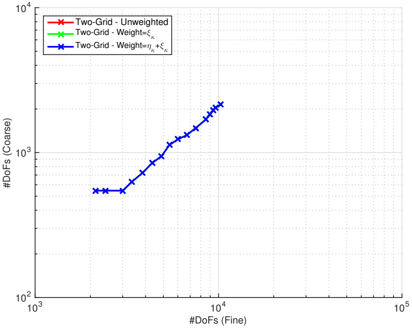
Error vs. Computation Time
(*h*-refinement)



Error vs. Computation Time
(*hp*-refinement)

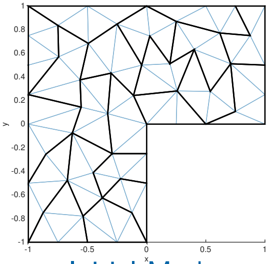


#DoFs: Fine vs. Coarse
(*h*-refinement)

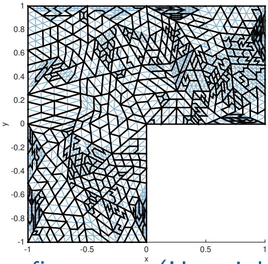


#DoFs: Fine vs. Coarse
(*hp*-refinement)

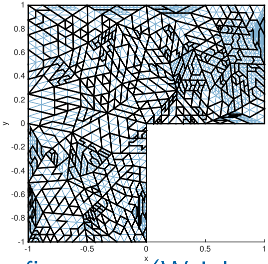
Non-Newtonian Fluid Flow: Smooth Solution



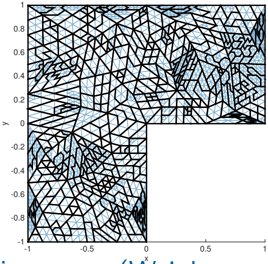
Initial Mesh



6 h-refinements (Unweighted)

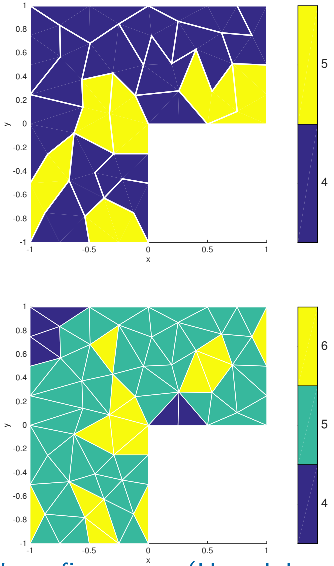


6 h-refinements (Weight $= \xi_\kappa$)

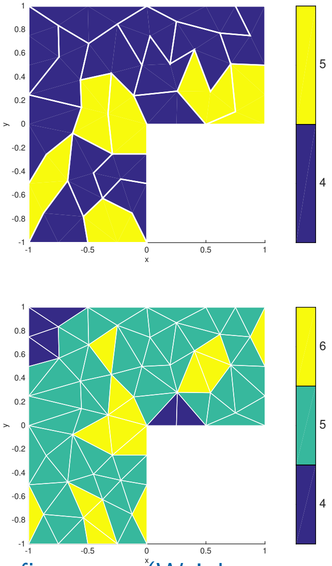


6 h-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

Non-Newtonian Fluid Flow: Smooth Solution



6 *hp*-refinements (Unweighted)



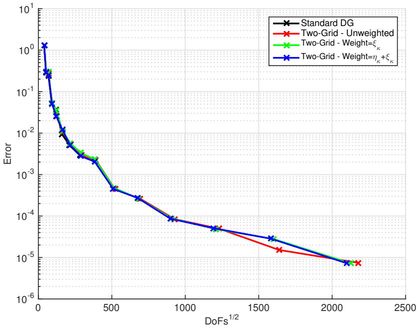
6 *hp*-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

We let $\Omega = (0,1)^2 \setminus [0,1) \times (-1,0]$, with Carreau Law nonlinearity $\mu(\boldsymbol{x}, |\underline{e}(u)|) = k_\infty + (k_0 - k_\infty)(1 + \lambda |\underline{e}(\boldsymbol{u})|^2)^{(\vartheta-2)/2}$, and select f so that

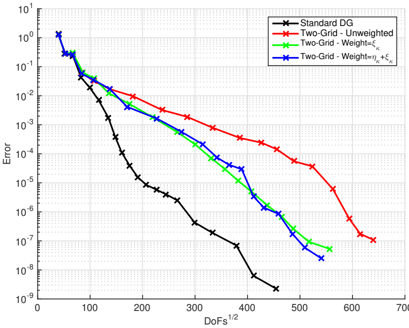
$$\boldsymbol{u}(x,y) = \begin{pmatrix} \left(1 - \cos\left(2\frac{\pi(e^{\vartheta x}-1)}{e^{\vartheta}-1}\right)\right) \sin(2\pi y) \\ -\vartheta e^{\vartheta x} \sin\left(2\frac{\pi(e^{\vartheta x}-1)}{e^{\vartheta}-1}\right) \frac{1-\cos(2\pi y)}{e^{\vartheta}-1} \end{pmatrix}$$

$$p(x,y) = 2\pi\vartheta e^{\vartheta x} \sin\left(2\frac{\pi(e^{\vartheta x}-1)}{e^{\vartheta}-1}\right) \frac{\sin(2\pi y)}{e^{\vartheta}-1}.$$

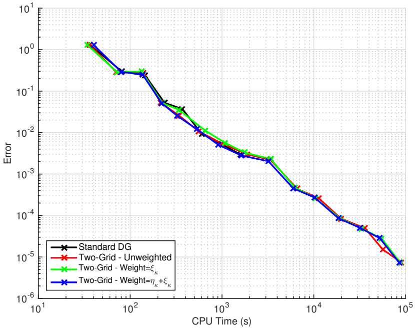
We select $k_\infty = 1$, $k_0 = 2$, $\lambda = 1$, and $\vartheta = 1.2$.



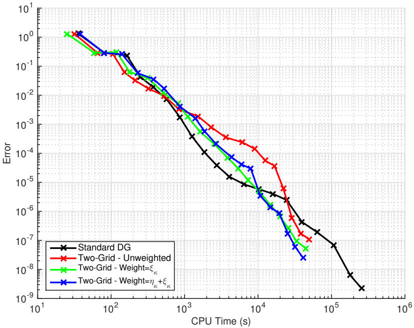
Error vs. #DoFs
(h-refinement)



Error vs. #DoFs
(hp-refinement)

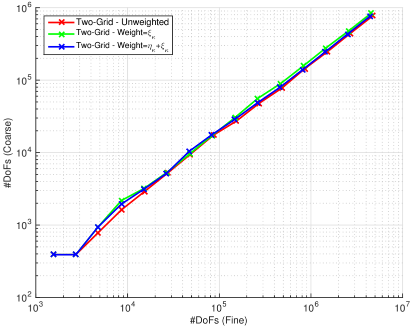


Error vs. Computation Time (*h*-refinement)

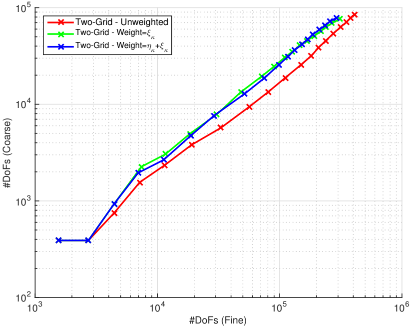


Error vs. Computation Time (*hp*-refinement)

Non-Newtonian Fluid Flow: Cavity Problem

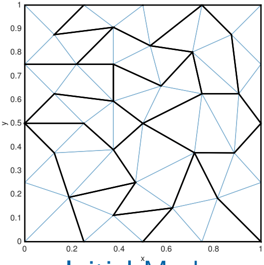


#DoFs: Fine vs. Coarse
(*h*-refinement)

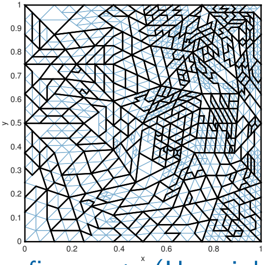


#DoFs: Fine vs. Coarse
(*hp*-refinement)

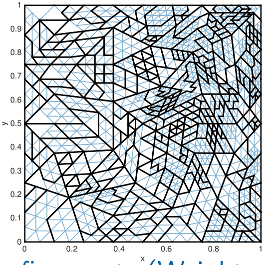
Non-Newtonian Fluid Flow: Cavity Problem



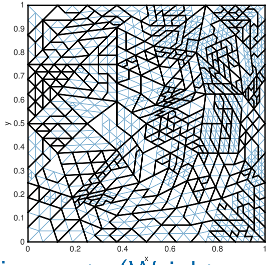
Initial Mesh



6 h-refinements (Unweighted)



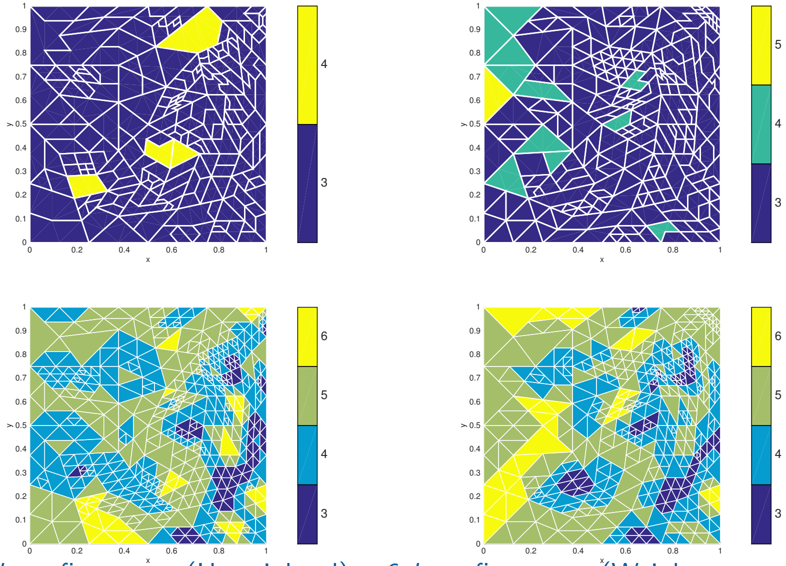
6 h-refinements (Weight $= \xi_\kappa$)



6 h-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

Non-Newtonian Fluid Flow: Cavity Problem

universität wien



6 *hp*-refinements (Unweighted)

6 *hp*-refinements (Weight $= \eta_\kappa + \xi_\kappa$)

Summary:

- Two-Grid DG *a posteriori* error estimates still hold for agglomerated coarse mesh of polygons and fine mesh of simplices.
- We can adaptively refine the coarse mesh based on the error estimates.

Future Aims:

- *A priori* error bounds.
- Extend to general nonlinearities.