Graded meshes in optimal control of elliptic partial differential equations

Sergejs Rogovs

Universität der Bundeswehr München

Partial differential equations, optimal design and numerics

Benasque, 2015, Aug 23 – Sep 04

Joint work with Th. Apel, J. Pfefferer, A. Rösch and D. Sirch

Support by DFG is gratefully acknowledged.

1 Introduction

2 Elliptic boundary value problems

3 Neumann optimal control problems

4 Practical aspects of implementation

5 Summary

1 Introduction

2 Elliptic boundary value problems

3 Neumann optimal control problems

4 Practical aspects of implementation

5 Summary

Universität der Bundeswehr München

Model problem

Linear elliptic problems

Find weak solution y of

$$-\Delta y + y = f \quad \text{in } \Omega$$

which fulfills the boundary conditions

$$y = 0 \quad \text{on } \Gamma \quad \text{or} \quad \partial_n y = g \quad \text{on } \Gamma,$$

respectively.

Setting

- Ω is a polygonal domain with boundary Γ.
- Data f and g are as we need (smooth enough).

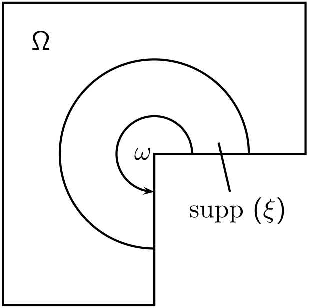
Problems in polygonal domains – corner singularities

The regularity of the solution y of

$$-\Delta y + y = f \text{ in } \Omega,$$
$$y = 0 \text{ on } \Gamma \quad \text{or} \quad \partial_n y = g \text{ on } \Gamma,$$

respectively, is limited by the largest interior angle ω in the domain, even if f and g are regular enough, e.g.,

- $y \in H^2(\Omega)$ for $\omega < \pi$
- $y \in W^{2,\infty}(\Omega)$ for $\omega < \pi/2$



One can write $y = y_r + y_s$, where y_r depends on the regularity of the right hand side and y_s contains terms like

$$\xi(r) r^{\lambda} \sin(\lambda\phi) \quad \text{or} \quad \xi(r) r^{\lambda} \cos(\lambda\phi),$$

respectively, with $\lambda = \pi/\omega$ and $\xi(r)$ is a smooth cut-off function.

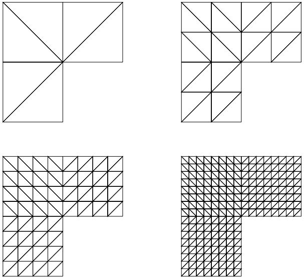
Finite element discretization on quasi-uniform meshes

Let $\mathcal{T}_h$ be a family of admissible triangulations with

$$h_T := diam T \sim h \quad \forall T \in \mathcal{T}_h,$$

where h denotes the mesh parameter. Furthermore, let

$$V_h = \{v_h \in C(\bar{\Omega}) : v_h|_T \in \mathcal{P}_1 \; \forall T \in \mathcal{T}_h\}.$$



Finite element discretizations

Find $y_h \in V_{h,0} := V_h \cap H_0^1(\Omega)$ such that

$$(\nabla y_h, \nabla v_h)_{L^2(\Omega)} + (y_h, v_h)_{L^2(\Omega)} = (f, v_h)_{L^2(\Omega)} \quad \forall v_h \in V_{h,0}.$$

Find $y_h \in V_h$ such that

$$(\nabla y_h, \nabla v_h)_{L^2(\Omega)} + (y_h, v_h)_{L^2(\Omega)} = (f, v_h)_{L^2(\Omega)} + (g, v_h)_{L^2(\Gamma)} \quad \forall v_h \in V_h.$$

1 Introduction

2 Elliptic boundary value problems

3 Neumann optimal control problems

4 Practical aspects of implementation

5 Summary

Universität der Bundeswehr München

The idea of mesh grading

The poor approximation property of the finite element method is due to the singular terms y_s in the solution, i.e.,

$$y_s = \xi(r) r^\lambda \sin(\lambda\phi) \quad \text{or} \quad y_s = \xi(r) r^\lambda \cos(\lambda\phi),$$

respectively, where $\lambda = \pi/\omega$ and $\xi(r)$ is a smooth cut-off function.

Basic idea according to Oganesyan and Rukhovets

Use a local transformation of coordinates via

$$r = \varrho^{1/\mu},$$

which transforms a neighborhood Ω_C of the critical corner to Ω'_C.

Essential properties

$$\partial_{\varrho\varrho} y_s \sim \partial_{\varrho\varrho} r^\lambda = \partial_{\varrho\varrho} \varrho^{\lambda/\mu}$$

$$\Rightarrow \; y_s \in H^2(\Omega'_C) \quad \Leftrightarrow \quad 2\,(\lambda/\mu - 2) + 1 > -1 \quad \Leftrightarrow \quad \mu < \lambda$$

Finite element error estimates on graded meshes (1)

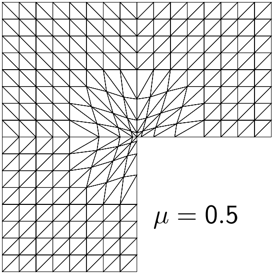
In computations the transformation of coordinates is not practicable.

Computational realization – graded meshes

We set the element size $h_T := diam T$ according to

$$h_T \sim \begin{cases} h^{1/\mu} & \text{for } r_T = 0 \\ h r_T^{1-\mu} & \text{for } R \geq r_T > 0 \,, \\ h & \text{for } r_T > R \end{cases}$$

where h is the global mesh parameter and $\mu \in (0,1]$ the grading parameter.



$\mu = 0.5$

FE-error estimate in $L^2(\Omega)$ and $H^1(\Omega)$ for both problems

The finite element error can be estimated by

$$\|y - y_h\|_{L^2(\Omega)} + h\|y - y_h\|_{H^1(\Omega)} \leq ch^2$$

on meshes introduced above with grading parameter $\mu < \lambda$. [Pfefferer 2014]

$\omega < \pi \Leftrightarrow \lambda > 1 \Rightarrow$ Mesh grading in non-convex domains only.

Finite element error estimates on graded meshes (2)

To get error estimates in $L^\infty(\Omega)$ of order close to two, we require $y \in W^{2,\infty}(\Omega)$.

Basic idea according to Oganesyan and Rukhovets

Use again the local transformation $r = \varrho^{1/\mu}$:

$$\partial_{\varrho\varrho} y_s \sim \partial_{\varrho\varrho} r^\lambda = \partial_{\varrho\varrho} \varrho^{\lambda/\mu}$$

$$\Rightarrow \;\; y_s \in W^{2,\infty}(\Omega'_C) \quad \Leftrightarrow \quad \lambda/\mu - 2 > 0 \quad \Leftrightarrow \quad \mu < \lambda/2$$

FE-error estimate in $L^\infty(\Omega)$ for Dirichlet problem

The finite element error can be estimated by

$$\|y - y_h\|_{L^\infty(\Omega)} \le ch^{2-\epsilon} \qquad \text{[Schatz/Wahlbin 1978]}$$

$$\|y - y_h\|_{L^\infty(\Omega)} \le ch^2 |\ln h|^{3/2} \qquad \text{[Sirch 2010]}$$

on graded meshes with grading parameter $\mu < \lambda/2$.

$\omega < \pi/2 \Leftrightarrow \lambda/2 > 1 \Rightarrow$ Mesh grading for domains with $\omega \ge \pi/2$.

Finite element error estimates on graded meshes (3)

FE-error estimate in $L^\infty(\Omega)$ for Neumann problem

The finite element error can be estimated by

$$\|y - y_h\|_{L^\infty(\Omega)} \le ch^2 |\ln h|^{3/2} \quad \text{[almost finished]}$$

on graded meshes with grading parameter $\mu < \lambda/2$.

Main difficulty

Friedrichs' inequality does not hold for Neumann problem

FE-error estimate in $L^\infty(\Omega)$ for Dirichlet pr. with new proof technique

The finite element error can be estimated by

$$\|y - y_h\|_{L^\infty(\Omega)} \le ch^2 |\ln h|$$

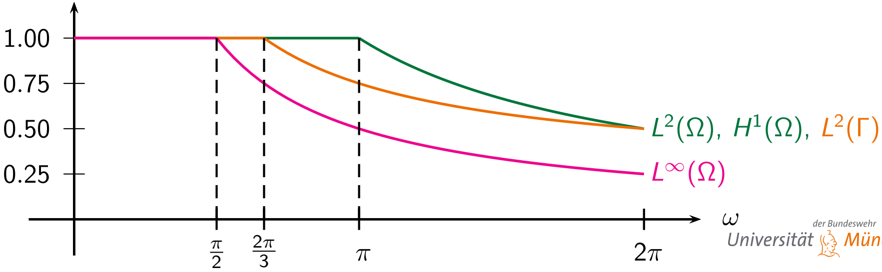
on graded meshes with grading parameter $\mu < \lambda/2$.

Summary

Table: Summary of mesh grading results for different norms

Norm	Grading parameter	Approximation rate	Critical angle
$\|y - y_h\|_{H^1(\Omega)}$	$\mu < \lambda$	h	π
$\|y - y_h\|_{L^2(\Omega)}$	$\mu < \lambda$	h^2	π
$\|y - y_h\|_{L^\infty(\Omega)}$	$\mu < \lambda/2$	$h^2 \lvert \ln h \rvert^{3/2}$	$\pi/2$
$\|y - y_h\|_{L^2(\Gamma)}$	$\mu < 1/4 + \lambda/2$	$h^2 \lvert \ln h \rvert^{3/2}$	$2\pi/3$

Figure: Mesh grading conditions for different norms depending on ω



1 Introduction

2 Elliptic boundary value problems

3 Neumann optimal control problems

4 Practical aspects of implementation

5 Summary

Universität der Bundeswehr München

Optimal control problems with Neumann boundary control

Model problem

$$\min \frac{1}{2}\|y - y_d\|^2_{L^2(\Omega)} + \frac{\nu}{2}\|u\|^2_{L^2(\Gamma)}$$
$$\text{s.t.} \quad -\Delta y + y = 0 \quad \text{in } \Omega$$
$$\partial_n y = u \quad \text{on } \Gamma$$
$$a \le u(x) \le b \quad \text{for a.a. } x \in \Gamma$$

Discrete problem

$$\min \frac{1}{2}\|y_h - y_d\|^2_{L^2(\Omega)} + \frac{\nu}{2}\|u_h\|^2_{L^2(\Gamma)}$$
$$\text{s.t.} \int_\Omega (\nabla y_h \cdot \nabla v_h + y_h v_h) = \int_\Gamma u_h v_h \; \forall v_h \in V_h$$
$$u_h \in U_h^{ad}$$

- Variational discretization: $U_h^{ad} = U_{ad} := \{u \in L^2(\Gamma) : a \le u \le b \text{ a.e. on } \Gamma\}$
- Postprocessing approach: $U_h^{ad} = U_{ad} \cap \{u_h \in L^\infty(\Gamma) : u_h|_E \in \mathcal{P}_0 \; \forall E \in \mathcal{E}_h\}$

Both problems admit a unique solution $(\bar{u}, \bar{y})$ and $(\bar{u}_h, \bar{y}_h)$.

Optimality condition

$$\bar{u} = \Pi_{[a,b]}(-\bar{p}_{|\Gamma}/\nu),$$
$$-\Delta\bar{p} + \bar{p} = \bar{y} - y_d \text{ in } \Omega$$
$$\partial_n \bar{p} = 0 \text{ on } \Gamma$$

Optimality condition

$$\text{VD: } \bar{u}_h = \Pi_{[a,b]}(-\bar{p}_{h|\Gamma}/\nu),$$
$$\text{PA: } \bar{u}_h^p = \Pi_{[a,b]}(-\bar{p}_{h|\Gamma}/\nu),$$
$$\int_\Omega (\nabla \bar{p}_h \cdot \nabla v_h + \bar{p}_h v_h) = \int_\Omega (\bar{y}_h - y_d) v_h \; \forall v_h \in V_h$$

Error estimates for Neumann boundary control problems

Error estimates for the variational discretization in L^2

The error estimates

$$\|\bar{u} - \bar{u}_h\|_{L^2(\Gamma)} + \|\bar{y} - \bar{y}_h\|_{L^2(\Omega)} + \|\bar{p} - \bar{p}_h\|_{L^2(\Omega)} \le ch^2 |\ln h|^{3/2}$$

are valid on graded meshes with grading parameter $\mu < 1/4 + \lambda/2$.

Let $\bar{u}_h^p = \Pi_{[a,b]}(-\bar{p}_{h|\Gamma}/\nu)$ and K the union of all elements $E \in \mathcal{E}_h$, where the optimal control $\bar{u}$ has kinks with the control constraints.

Error estimates for the postprocessing approach in L^2

The error estimates

$$\|\bar{u} - \bar{u}_h^p\|_{L^2(\Gamma)} + \|\bar{y} - \bar{y}_h\|_{L^2(\Omega)} + \|\bar{p} - \bar{p}_h\|_{L^2(\Omega)} \le ch^2 |\ln h|^{3/2}$$

are valid on graded meshes with grading parameter $\mu < 1/4 + \lambda/2$ if $|K| \le ch$.

Essential ingredients for error estimates

Finite element error estimates in $L^2(\Omega)$ and $L^2(\Gamma)$.

Error estimates for Neumann boundary control problems

Error estimates for the variational discretization in L^∞

The error estimates

$$\|\bar{u} - \bar{u}_h\|_{L^\infty(\Gamma)} + \|\bar{y} - \bar{y}_h\|_{L^\infty(\Omega)} + \|\bar{p} - \bar{p}_h\|_{L^\infty(\Omega)} \le ch^2|\ln h|^{3/2}$$

are valid on graded meshes with grading parameter $\mu < \lambda/2$.

Let $\bar{u}_h^p = \Pi_{[a,b]}(-\bar{p}_h/\nu)$ and K the union of all elements $E \in \mathcal{E}_h$, where the optimal control $\bar{u}$ has kinks with the control constraints.

Error estimates for the postprocessing approach in L^∞

The error estimates

$$\|\bar{u} - \bar{u}_h^p\|_{L^\infty(\Gamma)} + \|\bar{y} - \bar{y}_h\|_{L^\infty(\Omega)} + \|\bar{p} - \bar{p}_h\|_{L^\infty(\Omega)} \le ch^2|\ln h|^{3/2}$$

are valid on graded meshes with grading parameter $\mu < \lambda/2$ if $|K| \le ch$.

Essential ingredients for error estimates

Finite element error estimates in $L^\infty(\Omega)$.

1 Introduction

2 Elliptic boundary value problems

3 Neumann optimal control problems

4 Practical aspects of implementation

5 Summary

Universität der Bundeswehr München

Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

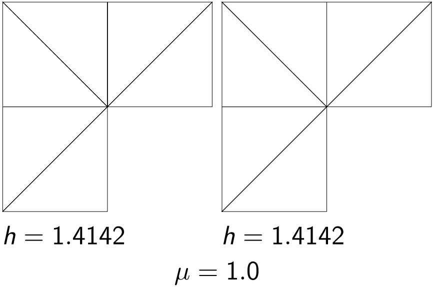
for all $X^{(i)} \in \Omega \cap S_R$.

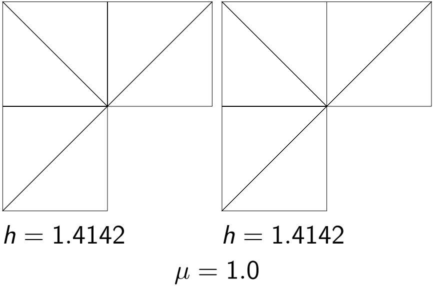
Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.





Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

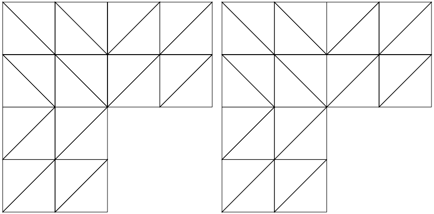
for all $X^{(i)} \in \Omega \cap S_R$.

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

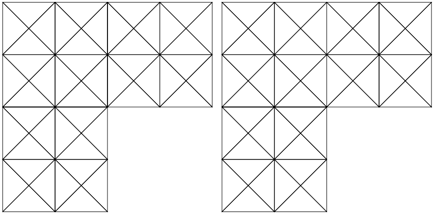
$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.7071$ $h = 0.7071$

$\mu = 1.0$



$h = 0.5000$ $h = 0.5000$

$\mu = 1.0$

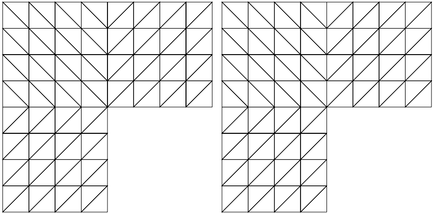
Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

for all $X^{(i)} \in \Omega \cap S_R$.



$h = 0.3536$ $\quad$ $h = 0.3536$

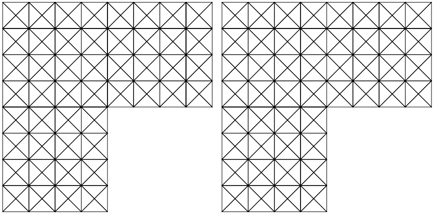
$\mu = 1.0$

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.2500$ $\quad$ $h = 0.2500$

$\mu = 1.0$

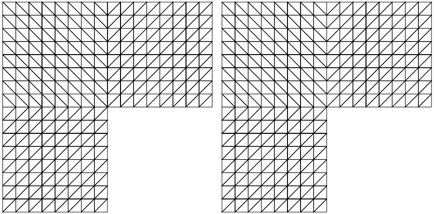
Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X_{new}^{(i)} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

for all $X^{(i)} \in \Omega \cap S_R$.



$h = 0.1768$ $h = 0.1768$

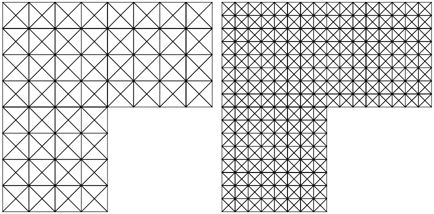
$\mu = 1.0$

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.2500$ $h = 0.1250$

$\mu = 1.0$

Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \ \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X_{new}^{(i)} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

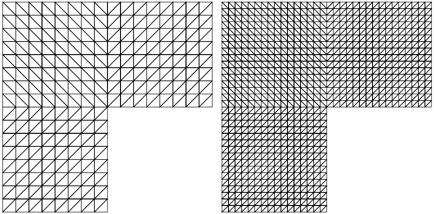
for all $X^{(i)} \in \Omega \cap S_R$.

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

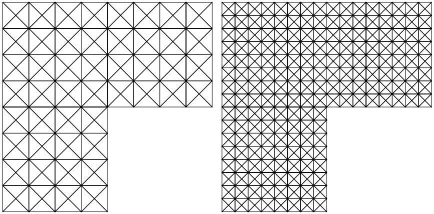
$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.1768$ $h = 0.0884$

$\mu = 1.0$



$h = 0.2500$ $h = 0.1250$

$\mu = 1.0$

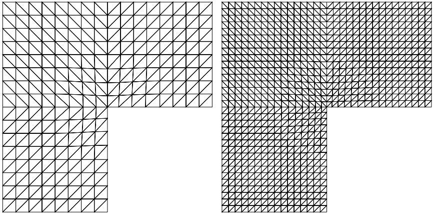
Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

for all $X^{(i)} \in \Omega \cap S_R$.



$h = 0.1768$ $h = 0.0884$

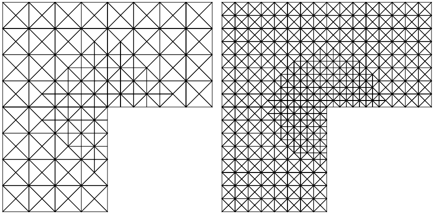
$\mu = 0.9$

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.2500$ $h = 0.1250$

$\mu = 0.9$

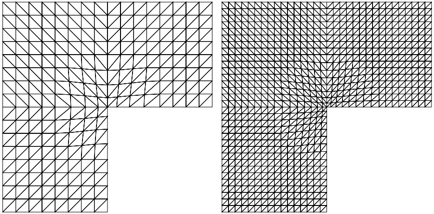
Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

for all $X^{(i)} \in \Omega \cap S_R$.



$h = 0.1768$ $\qquad$ $h = 0.0884$

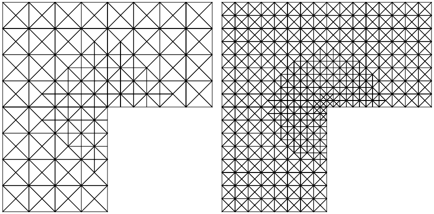
$\mu = 0.8$

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.2500$ $\qquad$ $h = 0.1250$

$\mu = 0.8$

Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

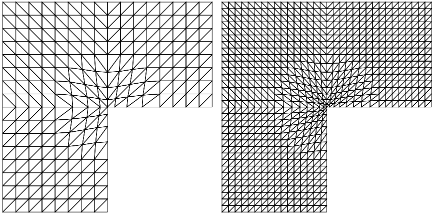
for all $X^{(i)} \in \Omega \cap S_R$.

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

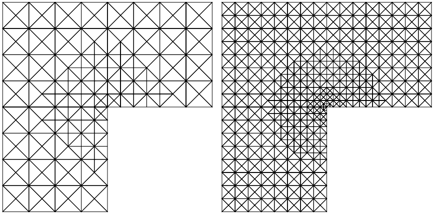
$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.1768$ $h = 0.0884$

$\mu = 0.7$



$h = 0.2500$ $h = 0.1250$

$\mu = 0.7$

Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

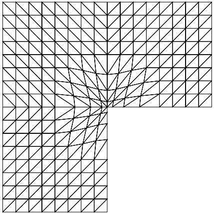
for all $X^{(i)} \in \Omega \cap S_R$.

Local refinement

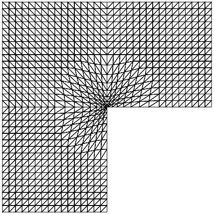
Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.

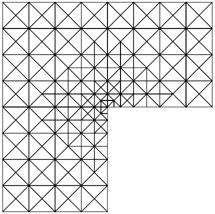


$h = 0.1768$

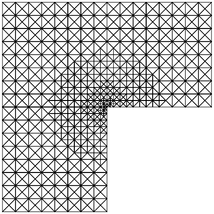


$h = 0.0884$

$\mu = 0.6$



$h = 0.2500$



$h = 0.1250$

$\mu = 0.6$

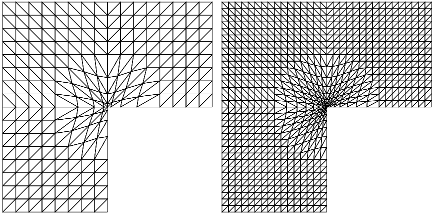
Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

for all $X^{(i)} \in \Omega \cap S_R$.



$h = 0.1768$ $h = 0.0884$

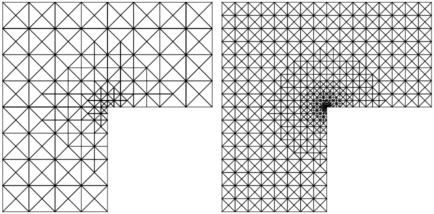
$\mu = 0.5$

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.2500$ $h = 0.1250$

$\mu = 0.5$

Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

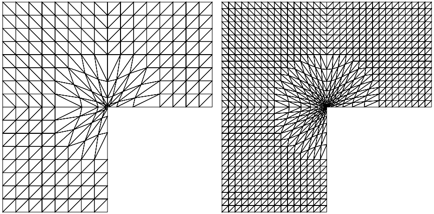
for all $X^{(i)} \in \Omega \cap S_R$.

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

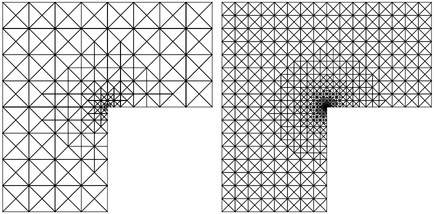
$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.1768$ $\quad$ $h = 0.0884$

$\mu = 0.4$



$h = 0.2500$ $\quad$ $h = 0.1250$

$\mu = 0.4$

Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

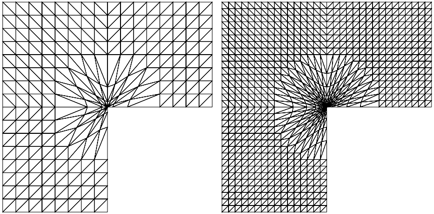
for all $X^{(i)} \in \Omega \cap S_R$.

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

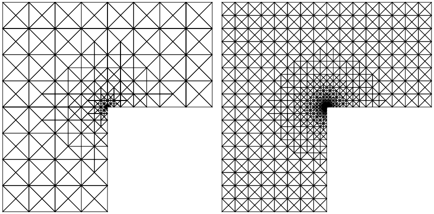
$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.1768$ $\quad$ $h = 0.0884$

$\mu = 0.3$



$h = 0.2500$ $\quad$ $h = 0.1250$

$\mu = 0.3$

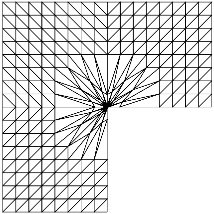
Generation of graded meshes

Transformation of nodes

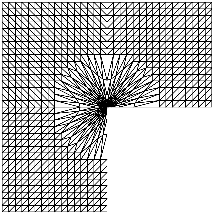
Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

for all $X^{(i)} \in \Omega \cap S_R$.



$h = 0.1768$



$h = 0.0884$

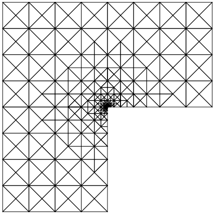
$\mu = 0.2$

Local refinement

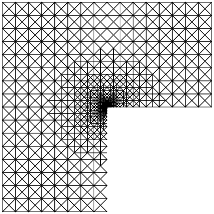
Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.2500$



$h = 0.1250$

$\mu = 0.2$

Generation of graded meshes

Transformation of nodes

Refine uniformly a coarse start mesh until $h_T \sim h \; \forall T \in \mathcal{T}_h$ with desired mesh size h. Afterwards, transform the nodes $X^{(i)}$ according to

$$X^{(i)}_{new} = X^{(i)} \left(\frac{r(X^{(i)})}{R} \right)^{1/\mu - 1}$$

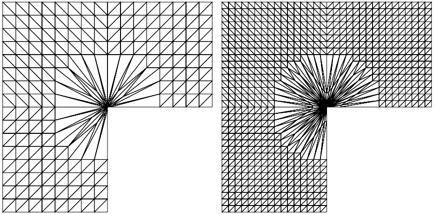
for all $X^{(i)} \in \Omega \cap S_R$.

Local refinement

Initialize refinement algorithm with coarse start mesh. Afterwards, mark every element $T \in \mathcal{T}_h$ for refinement which satisfies

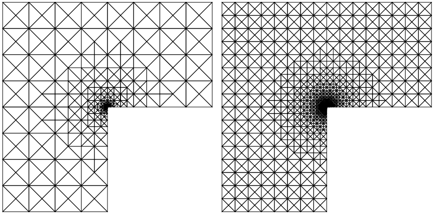
$$h_T > h \quad \text{or} \quad h_T > h \left(\frac{r_{T,C}}{R} \right)^{1-\mu}$$

until desired mesh size h is reached.



$h = 0.1768$ $\quad$ $h = 0.0884$

$\mu = 0.1$



$h = 0.2500$ $\quad$ $h = 0.1250$

$\mu = 0.1$

1 Introduction

2 Elliptic boundary value problems

3 Neumann optimal control problems

4 Practical aspects of implementation

5 Summary

Summary

- Introduction to corner singularities
- Introduction to mesh grading
- FE-error estimate with graded meshes in different norms
  - $\|y - y_h\|_{L^2(\Omega)} + h\|y - y_h\|_{H^1(\Omega)} \leq ch^2$ for $\mu < \lambda$
  - $\|y - y_h\|_{L^\infty(\Omega)} \leq ch^2|\ln h|^{3/2}$ for $\mu < \lambda/2$
  - $\|y - y_h\|_{L^2(\Gamma)} \leq ch^2|\ln h|^{3/2}$ for $\mu < 1/4 + \lambda/2$
- Optimal control problems with Neumann boundary control
  - Variational approach
    $\|\bar{u} - \bar{u}_h\|_{L^q(\Gamma)} + \|\bar{y} - \bar{y}_h\|_{L^q(\Omega)} + \|\bar{p} - \bar{p}_h\|_{L^q(\Omega)} \leq ch^2|\ln h|^{3/2}, \quad q = 2, \infty$
  - Postprocessing approach
    $\|\bar{u} - \bar{u}_h^p\|_{L^q(\Gamma)} + \|\bar{y} - \bar{y}_h\|_{L^q(\Omega)} + \|\bar{p} - \bar{p}_h\|_{L^q(\Omega)} \leq ch^2|\ln h|^{3/2}, \quad q = 2, \infty$
- Generation of graded meshes