A New Mesh Deformation Method for A Posteriori r-Adaptivity in the Context of FEM Error Control

M. Grajewski

Matthias.Grajewski@math.uni-dortmund.de

Graduate School for Production Engineering and Logistics;

Institute for Applied Mathematics,

University of Dortmund

Outline

- motivation
- mathematical background
- basic deformation method
- numerical realisation and analysis
- r-adaptivity
- conclusion and discussion

Motivation

observation 1: Goal-oriented adaptivity and local mesh refinement is mandatory for accurate and efficient computation.

Motivation

observation 1: Goal-oriented adaptivity and local mesh refinement is mandatory for accurate and efficient computation.

observation 2: The MFlop/s rates in adaptive FEM calculations today do by far not reach the peak performance.

example: Sparse MV multiplication in **FEATFLOW** (F77-code):

Computer	#Unknowns	**CM**	**TL**	**STO**
	8,320	147	136	**116**
DEC 21264	33,280	125	105	**100**
(667 MHz)	133,120	81	71	**58**
‘EV67’	532,480	60	51	**21**
$\sim$ 1300 MFlop/s	2,129,920	58	47	**13**
	8,519,680	58	45	**10**

Motivation II

reasons for observation 2:

1. on modern computers: CPU speed $\gg$ memory speed

2. current FEM-codes use
   - indirect addressing : many (unaligned) memory accesses
   - global data structures : effective caching prevented

Motivation II

reasons for observation 2:

1. on modern computers: CPU speed $\gg$ memory speed
2. current FEM-codes use
   - indirect addressing : many (unaligned) memory accesses
   - global data structures : effective caching prevented

problem:

Elementwise unstructured grids **allow adaptivity** but **prevent direct adressing**;

tensor product grids **allow direct adressing** and (seem to) **prevent adaptivity**.

Motivation II

reasons for observation 2:

1. on modern computers: CPU speed $\gg$ memory speed
2. current FEM-codes use
   - indirect addressing : many (unaligned) memory accesses
   - global data structures : effective caching prevented

problem:

Elementwise unstructured grids **allow adaptivity** but **prevent direct adressing**;

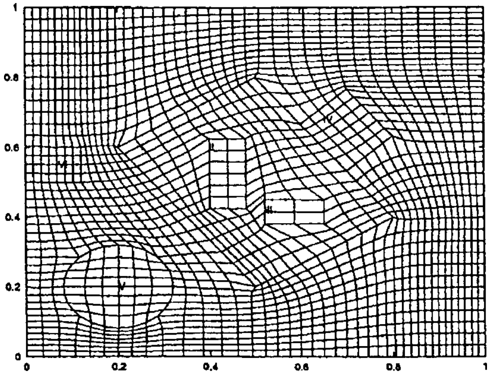
tensor product grids **allow direct adressing** and (seem to) **prevent adaptivity**.

synthesis: FEAST

(grid-related part of the) FEAST concept

definition of a generalized tensor product mesh: Every inner point has exactly 4 neighbours ⇒ linewise regular numbering

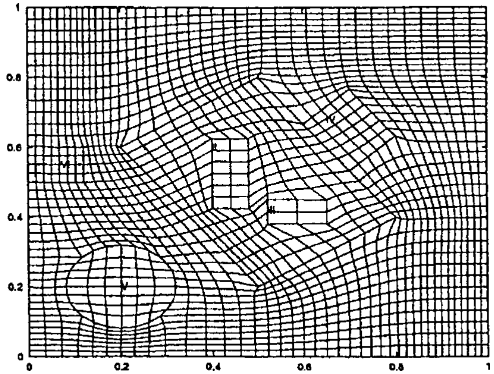
example:



(grid-related part of the) FEAST concept

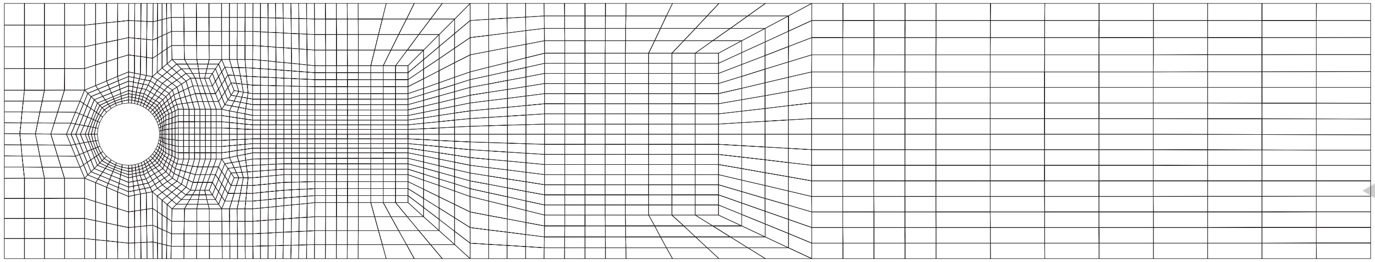
definition of a generalized tensor product mesh: Every inner point has exactly 4 neighbours $\Rightarrow$ linewise regular numbering

example:



The global grid consists of “many” local generalized tensor product meshes (“macros”).

example:



MFLOP-rate preserving adaptivity I

goal: adapted grids with local generalized tensor product structure

approach 1: macro wise adaptive refinement

- hanging nodes on macro edges are allowed
- the level difference of 2 edge-adjacent macros must not exceed 1

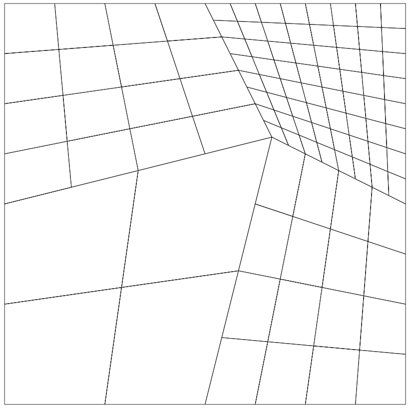
MFLOP-rate preserving adaptivity I

goal: adapted grids with local generalized tensor product structure

approach 1: macro wise adaptive refinement

- hanging nodes on macro edges are allowed
- the level difference of 2 edge-adjacent macros must not exceed 1

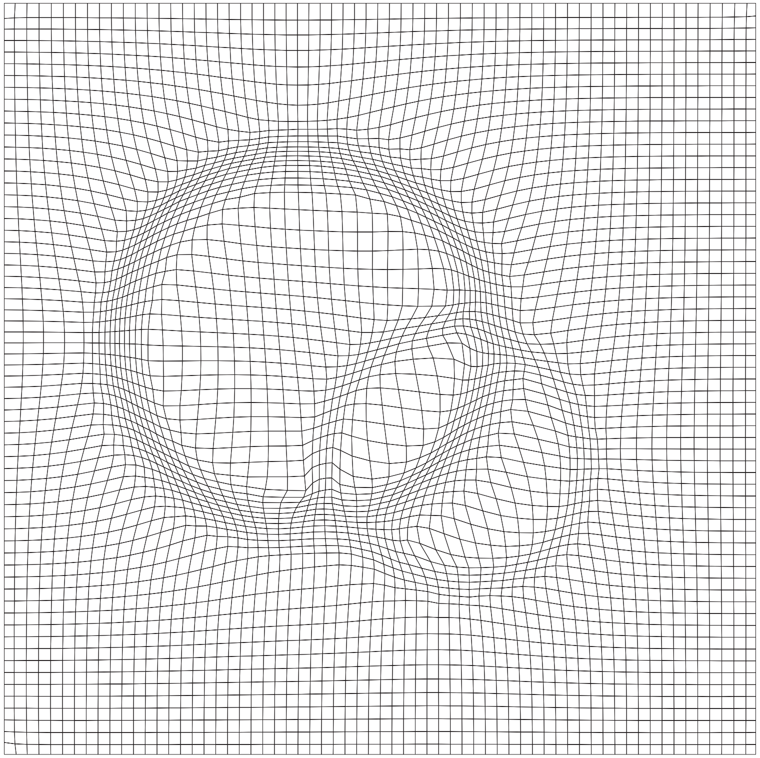
example



MFLOP-rate preserving adaptivity II

approach 2: grid deformation

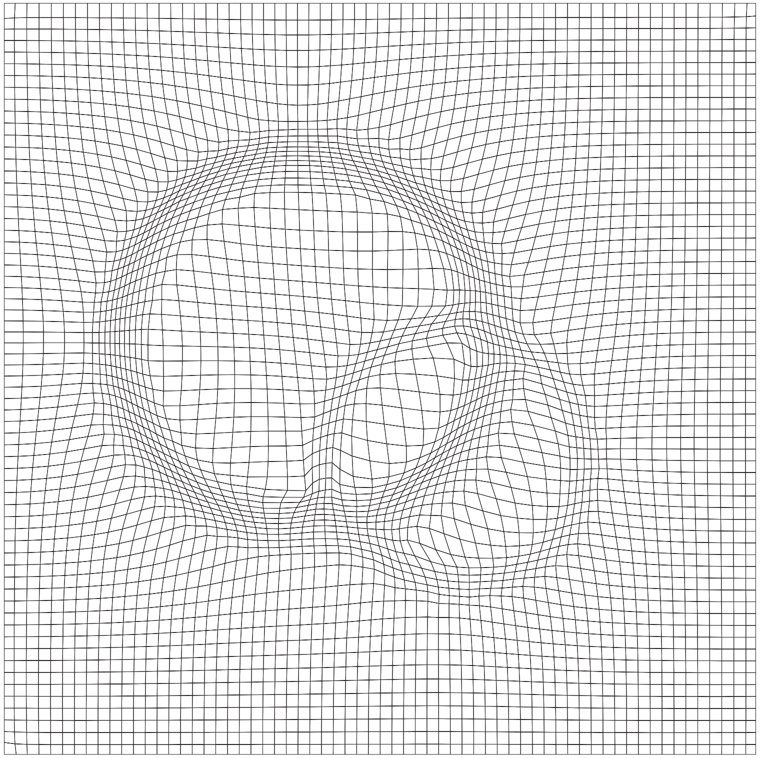
example



MFLOP-rate preserving adaptivity II

approach 2: grid deformation

example



Grid deformation preserves the (local) logical structure of the grid.

Theoretical background I

- domain Ω
- triangulation $\mathbb{T}$ with quads T
- “monitor function” $0 < f \in \mathcal{C}^1(\bar{\Omega})$
- “weighting function” $0 < g \in \mathcal{C}^1(\bar{\Omega})$

goal: construct transformation $\Phi : \Omega \rightarrow \Omega$ with

$$g(x)|J\Phi(x)| = f(\Phi(x)) \quad \forall x \in \Omega$$

and

$$\Phi : \partial\Omega \rightarrow \partial\Omega.$$

Theoretical background II

then:

$$m(\Phi(T)) := \int_{\Phi(T)} 1 dx = \int_T |J\Phi(x)| dx$$

1×1 Gauss rule:

$$g(x_{\mathrm{mid}}) \frac{m(\Phi(T))}{m(T)} = f(\Phi(x_{\mathrm{mid}})) + \mathcal{O}(h^2).$$

Theoretical background II

then:

$$m(\Phi(T)) := \int_{\Phi(T)} 1 dx = \int_T |J\Phi(x)| dx$$

1×1 Gauss rule:

$$g(x_{\text{mid}}) \frac{m(\Phi(T))}{m(T)} = f(\Phi(x_{\text{mid}})) + \mathcal{O}(h^2).$$

For

$$g(x) = m(T) + \mathcal{O}(h), \quad x \in T,$$

we obtain

$$m(\Phi(T)) = f(\Phi(x_{\text{mid}})) + \mathcal{O}(h).$$

$\Rightarrow$ local cell size on deformed grid $\approx f$, on initial grid $\approx g$

Theoretical background III

Theorem (Moser):

Let $0 < k \in \mathbb{N}$, $\alpha > 0$. Let $\Omega \subset \mathbb{R}^n$ be a domain with $\mathcal{C}^{3+k,\alpha}$-smooth boundary. Suppose $f, g \in \mathcal{C}^{k,\alpha}(\bar{\Omega})$ with $\int_\Omega f = \int_\Omega g$. Then, there exists a $\mathcal{C}^{k+1}$-diffeomorphism Φ, which fulfills

$$g(x)|J\Phi(x)| = f(\Phi(x)) \quad \forall x \in \Omega$$

and

$$\Phi(x) = x \quad \forall x \in \partial\Omega.$$

Theoretical background III

Theorem (Moser):

Let $0 < k \in \mathbb{N}$, $\alpha > 0$. Let $\Omega \subset \mathbb{R}^n$ be a domain with $\mathcal{C}^{3+k,\alpha}$-smooth boundary. Suppose $f, g \in \mathcal{C}^{k,\alpha}(\bar{\Omega})$ with $\int_\Omega f = \int_\Omega g$. Then, there exists a $\mathcal{C}^{k+1}$-diffeomorphism Φ, which fulfills

$$g(x)|J\Phi(x)| = f(\Phi(x)) \quad \forall x \in \Omega$$

and

$$\Phi(x) = x \quad \forall x \in \partial\Omega.$$

advantage: mesh tangling impossible

Theoretical background III

Theorem (Moser):

Let $0 < k \in \mathbb{N}$, $\alpha > 0$. Let $\Omega \subset \mathbb{R}^n$ be a domain with $\mathcal{C}^{3+k,\alpha}$-smooth boundary. Suppose $f, g \in \mathcal{C}^{k,\alpha}(\bar{\Omega})$ with $\int_\Omega f = \int_\Omega g$. Then, there exists a $\mathcal{C}^{k+1}$-diffeomorphism Φ, which fulfills

$$g(x)|J\Phi(x)| = f(\Phi(x)) \quad \forall x \in \Omega$$

and

$$\Phi(x) = x \quad \forall x \in \partial\Omega.$$

advantage: mesh tangling impossible

problem: Φ is not unique

construction of Φ (based on Liao)

1. construct f (e.g. from error distribution) and area distribution g

construction of Φ (based on Liao)

1. construct f (e.g. from error distribution) and area distribution g
2. Scale f or g by

$$\int_\Omega \frac{1}{f(x)}dx = \int_\Omega \frac{1}{g(x)}dx, \quad \tilde{f} := 1/f, \tilde{g} := 1/g.$$

construction of Φ (based on Liao)

1. construct f (e.g. from error distribution) and area distribution g
2. Scale f or g by

$$\int_\Omega \frac{1}{f(x)}dx = \int_\Omega \frac{1}{g(x)}dx, \quad \tilde{f} := 1/f, \tilde{g} := 1/g.$$

3. compute vector field $v : \Omega \to \mathbb{R}^n$

$$\mathrm{div}(v(x)) = \tilde{f}(x) - \tilde{g}(x), x \in \Omega, \quad \text{and } v(x) \cdot \mathfrak{n} = 0, x \in \partial\Omega.$$

construction of Φ (based on Liao)

1. construct f (e.g. from error distribution) and area distribution g
2. Scale f or g by

$$\int_\Omega \frac{1}{f(x)} dx = \int_\Omega \frac{1}{g(x)} dx, \quad \tilde{f} := 1/f, \tilde{g} := 1/g.$$

3. compute vector field $v : \Omega \to \mathbb{R}^n$

$$\operatorname{div}(v(x)) = \tilde{f}(x) - \tilde{g}(x), x \in \Omega, \quad \text{and } v(x) \cdot \mathfrak{n} = 0, x \in \partial\Omega.$$

4. For every grid point x solve the IVP

$$\frac{\partial \varphi(x,t)}{\partial t} = \eta(\varphi(x,t),t), 0 < t < 1, \quad \varphi(x,0) = x \text{ mit}$$

$$\eta(y,s) := \frac{-v(y)}{s\tilde{f}(y) + (1-s)\tilde{g}(y)}, \quad y \in \Omega, s \in [0,1].$$

construction of Φ (based on Liao)

1. construct f (e.g. from error distribution) and area distribution g
2. Scale f or g by

$$\int_\Omega \frac{1}{f(x)}dx = \int_\Omega \frac{1}{g(x)}dx, \quad \tilde{f} := 1/f, \tilde{g} := 1/g.$$

3. compute vector field $v : \Omega \to \mathbb{R}^n$

$$\operatorname{div}(v(x)) = \tilde{f}(x) - \tilde{g}(x), x \in \Omega, \quad \text{and } v(x) \cdot \mathfrak{n} = 0, x \in \partial\Omega.$$

4. For every grid point x solve the IVP

$$\frac{\partial \varphi(x,t)}{\partial t} = \eta(\varphi(x,t),t), 0 < t < 1, \quad \varphi(x,0) = x \text{ mit}$$

$$\eta(y,s) := \frac{-v(y)}{s\tilde{f}(y) + (1-s)\tilde{g}(y)}, \quad y \in \Omega, s \in [0,1].$$

$$\boxed{\Phi(x) = \varphi(x,1)}$$

basic deformation method

Deformation(f, $GRID$)

compute $\tilde{f} - \tilde{g}$, $g = g(GRID)$

solve $(\nabla w_h, \nabla \varphi_h) = (\tilde{f} - \tilde{g}, \varphi_h) \quad \forall \varphi_h \in \mathcal{Q}_1(\mathbb{T})$

$v_h :=$ recovered_gradient(w_h)

DO FORALL $x \in GRID$

solve $\frac{\partial \varphi(x,t)}{\partial t} = \eta_h(\varphi(x,t),t), \quad 0 \leq t \leq 1, \quad \varphi(x,0) = x$

$\Phi(x) = \varphi(x,1)$

ENDDO

END Deformation

practical aspects I: IVP

solving the IVP (step 3):

- several methods tested (LMM, RK) : Runge-Kutta method of 3rd order performed best
- boundary projection necessary for boundary points

practical aspects I: IVP

solving the IVP (step 3):

- several methods tested (LMM, RK) : Runge-Kutta method of 3rd order performed best
- boundary projection necessary for boundary points

problem: solving IVP requires evaluating FEM functions in real coordinates: search the grid per grid point and per time step

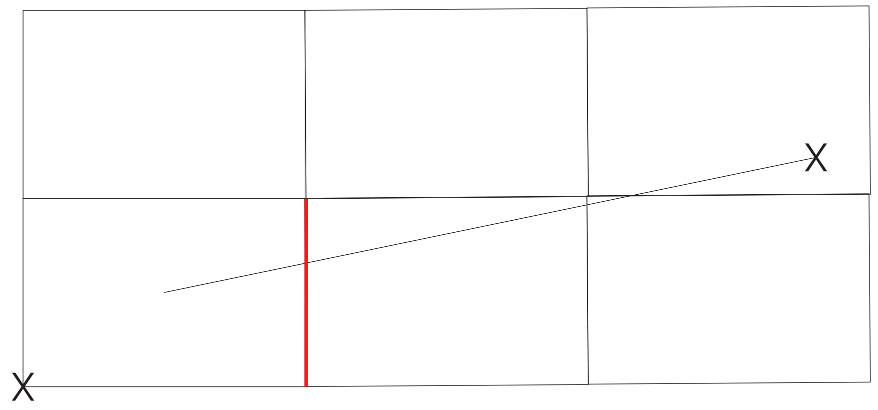
practical aspects I: IVP

solving the IVP (step 3):

- several methods tested (LMM, RK) : Runge-Kutta method of 3rd order performed best
- boundary projection necessary for boundary points

problem: solving IVP requires evaluating FEM functions in real coordinates: search the grid per grid point and per time step

remedy : raytracing search

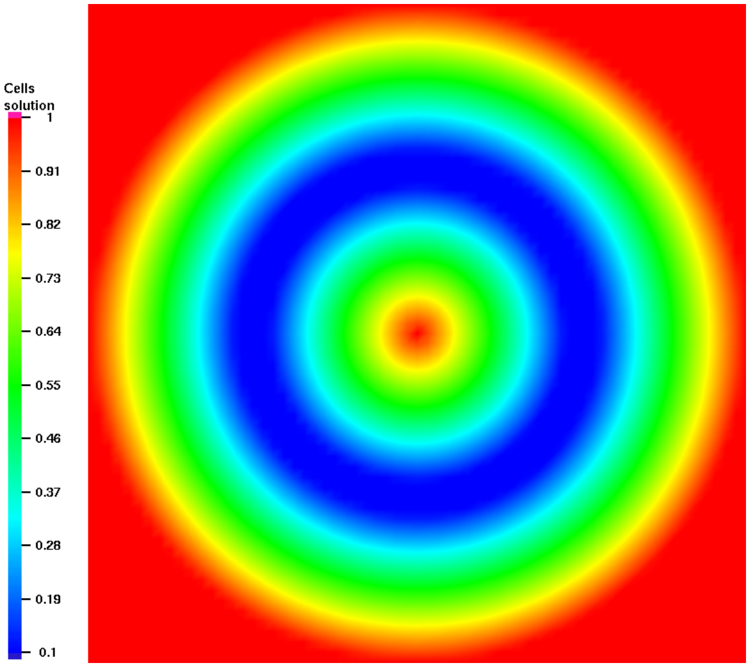


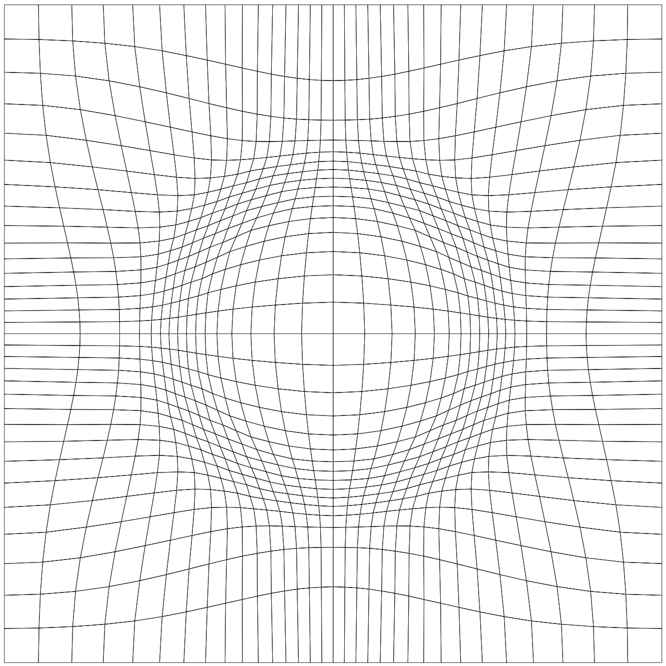
test problem I: circle in a square

domain : $\Omega = [0,1]^2$

monitor function: $f(x) = \min\left(1, \max(\frac{|d-0.25|}{0.25}, \epsilon)\right)$,

$d := \sqrt{(x_1 - 0.5)^2 + (x_2 - 0.5)^2}$, $\epsilon = 1/10$





practical aspects II: search time

N	raytracing	brute force
256	$9.36 \cdot 10^{-3}$	$5.65 \cdot 10^{-2}$
1024	$3.96 \cdot 10^{-2}$	$8.39 \cdot 10^{0}$
4096	$1.70 \cdot 10^{-1}$	$1.30 \cdot 10^{2}$
16384	$8.22 \cdot 10^{-1}$	$2.05 \cdot 10^{3}$
64536	$4.28 \cdot 10^{0}$	$3.25 \cdot 10^{4}$
262144	$2.42 \cdot 10^{1}$	-

Total search time needed by the the different search methods on, 10 Runge-Kutta steps

practical aspects III: accuracy

Definition:

$$q(x) := \frac{f(x)}{area(x)}$$

$$Q := ||1 - q||_{l^2}$$

practical aspects III: accuracy

Definition:

$$q(x) := \frac{f(x)}{area(x)}$$

$$Q := ||1 - q||_{l^2}$$

IteratedDeformation(f, $GRID$, $NCORR$, TOL)

DO i = 1, $NCORR$

$GRID$ = **Deformation(f, $GRID$)**

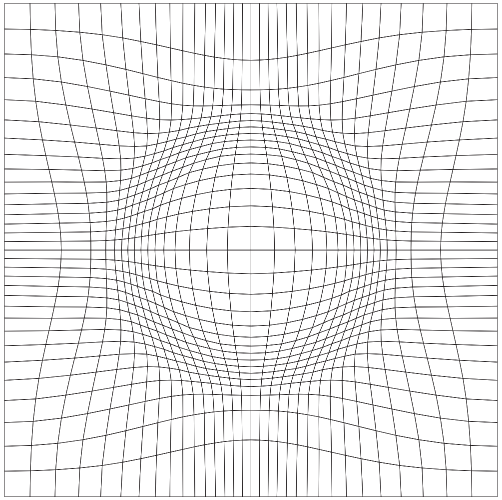
Q = Q(GRID)

IF (Q $<$ TOL) EXIT

ENDDO

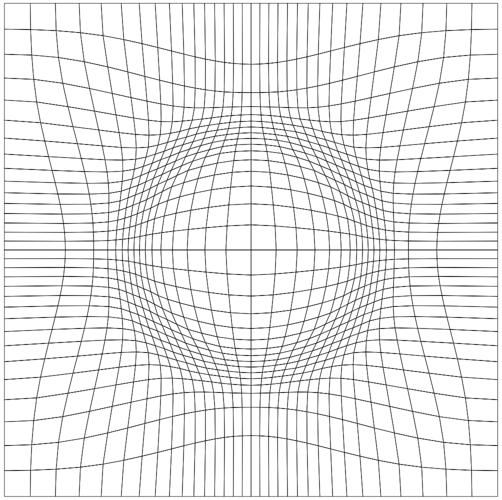
END IteratedDeformation

test problem I (circle in a square) revisited



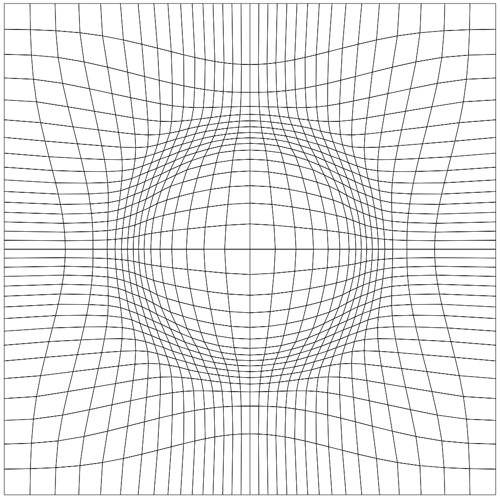
no correction

$Q = 1.15 \cdot 10^{-1}$



1 corr. step

$Q = 6.80 \cdot 10^{-2}$



2 corr. steps

$Q = 6.10 \cdot 10^{-2}$

practical aspects III: robustness

problem: numerical errors (v_h, IVP)
$\Rightarrow$ mesh tangling possible in practical computations

Definition:

$$f_s(x) := sf(x) + (1-s)g(x), \quad s \in [0,1]$$

practical aspects III: robustness

problem: numerical errors (v_h, IVP)
$\Rightarrow$ mesh tangling possible in practical computations

Definition:

$$f_s(x) := sf(x) + (1-s)g(x), \quad s \in [0,1]$$

EnhancedDeformation(f,$GRID$,$NCORR$,$NADAP$,TOL,S)

DO i = 1, NADAP

$GRID$ = **IteratedDeformation(**$f_{S(i)}$**,** $GRID$**,**
$NCORR(i)$**,** $TOL(i)$**)**

ENDDO

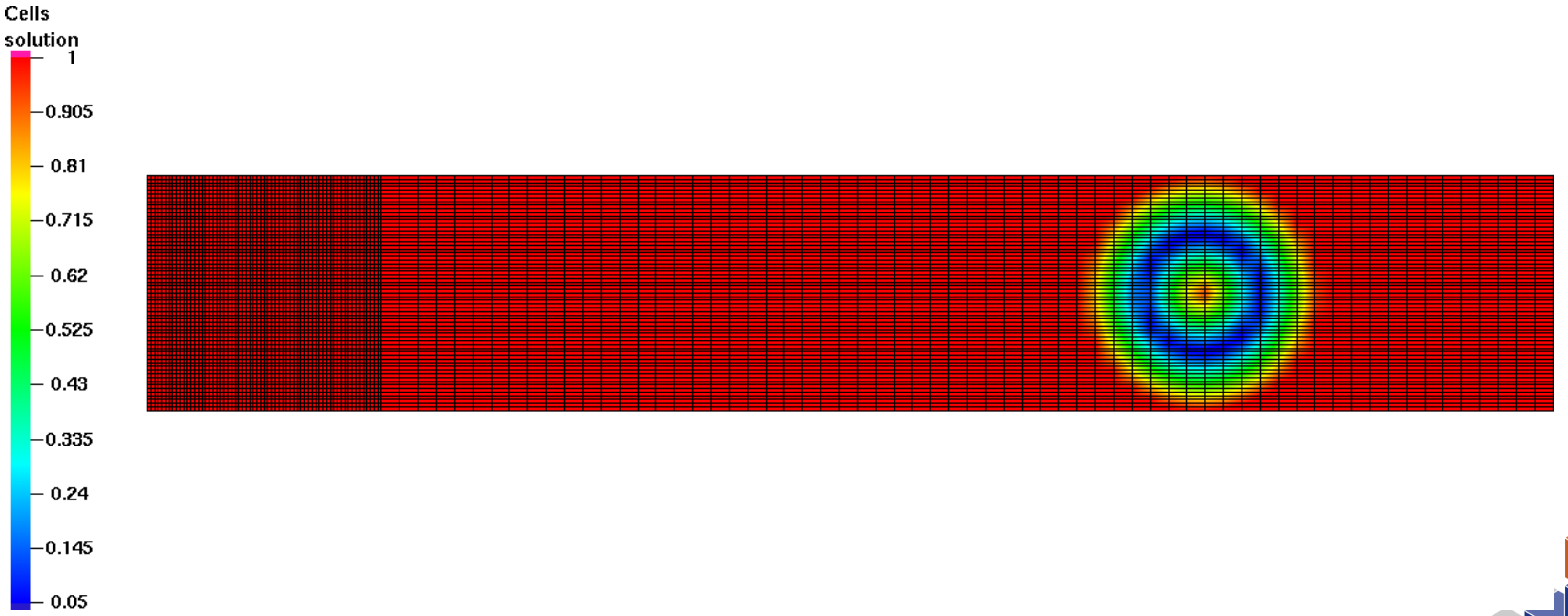
END EnhancedDeformation

test problem II: circle in a channel

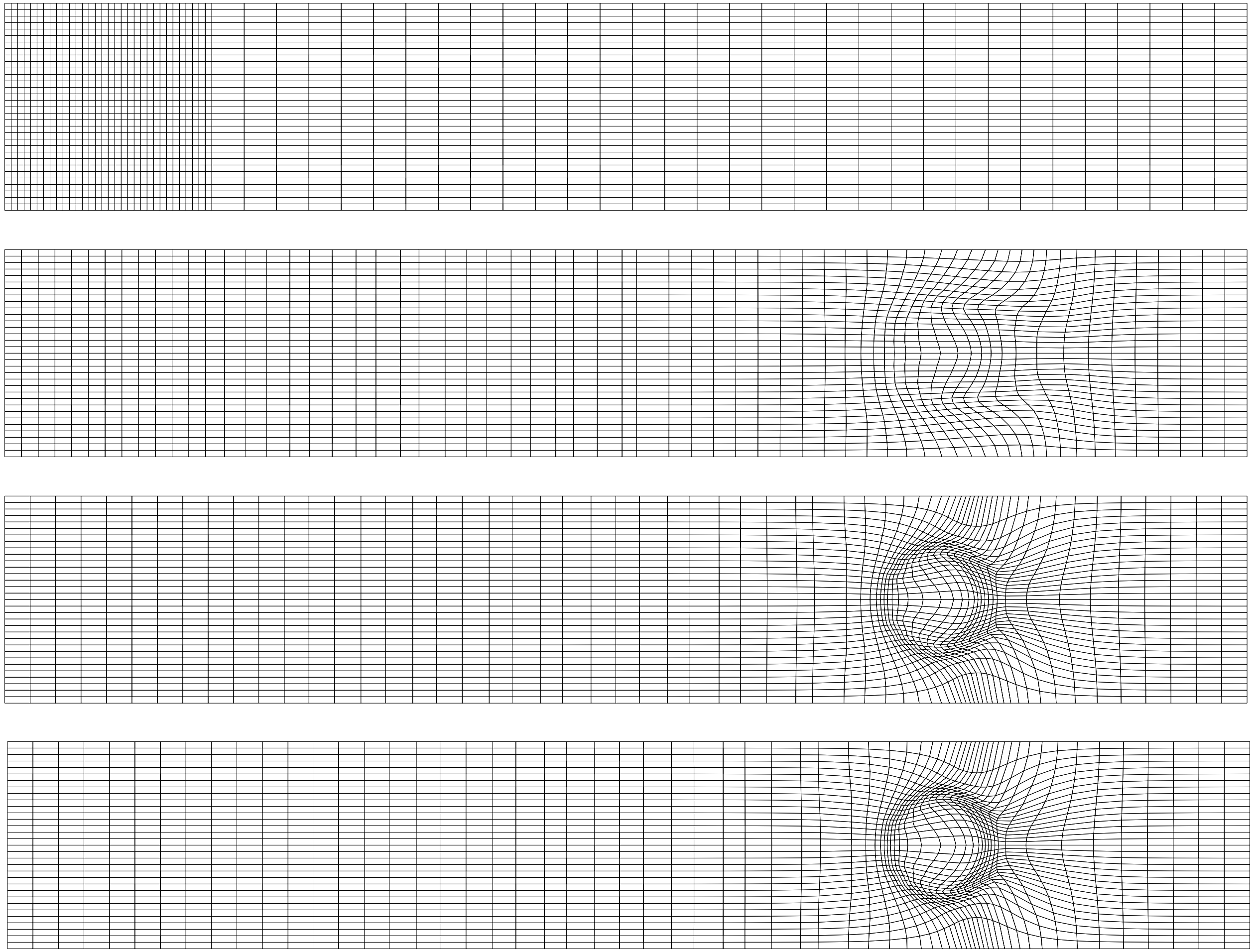
domain : $\Omega = [0,1] \times [0.6]$

monitor function: $f(x) = \min\left(1, \max(\frac{|d-0.25|}{0.25}, \epsilon)\right),$

$d := \sqrt{(x_1 - 4.5)^2 + (x_2 - 0.5)^2}, \epsilon = 1/10$



test problem II: circle in a channel



$s = 0$

$s = \sqrt{1/2}$

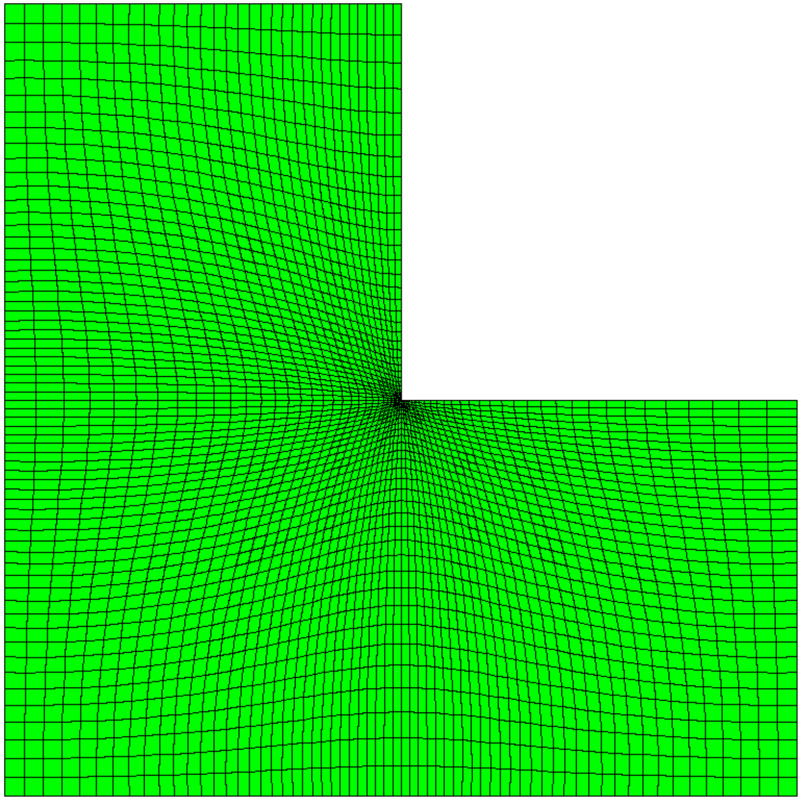
$s = 1$
$Q = 8.07 \cdot 10^{-2}$

$s = 1$
$Q = 3.98 \cdot 10^{-2}$

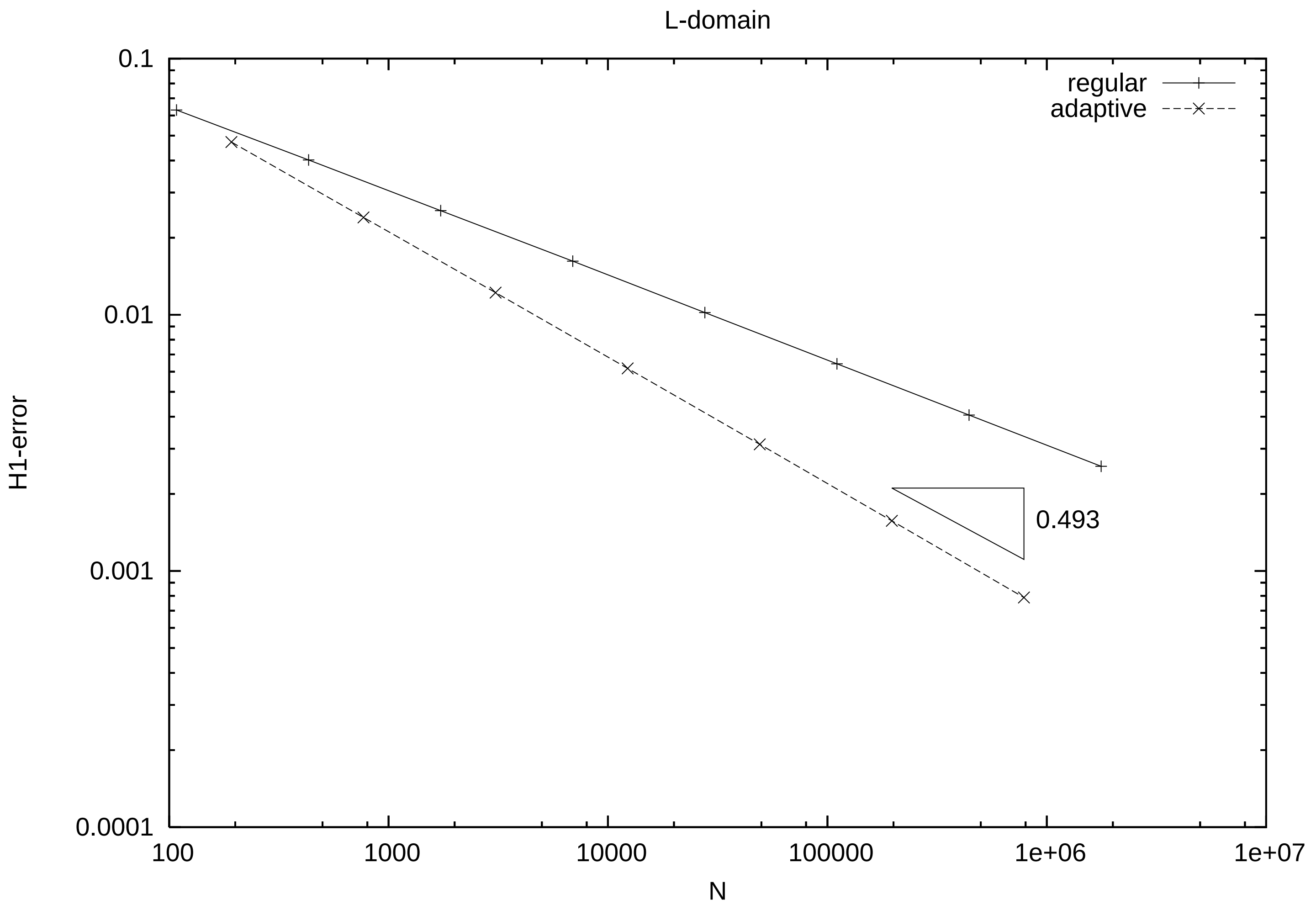
test problem III: L-domain

generic monitor function:

$$f(x) = \min(1, \max(|x|, h * c_0))$$



test problem III: L-domain



gradient error

Conclusion and discussion

- traditional adaption concepts: poor MFlop/s-rates
- FEAST-concept:
  - macro-wise hanging nodes
  - grid deformation
- Enhanced Deformation method
- numerical realisation and analysis
- r-adaptivity on L-domain

Thank you for your attention!