Petascale Strategies for FEM Simulations

Chr. Becker, S. Buijssen, D. Göddeke, M. Grajewski*, S. Turek, and H. Wobker

Technische Universität Dortmund

March 14th, 2008

Overview

technische universität dortmund

1. FEAST concepts
2. MFLOP/s-rate preserving adaptivity
3. Applications in structural mechanics

Overview

1. FEAST concepts
2. MFLOP/s-rate preserving adaptivity
3. Applications in structural mechanics

Petascale computing

technische universität dortmund

Petascale computing is the calculation of FEM simulations with 10^{15} unknowns in reasonable time.

FEAST overview

technische universität dortmund

FEAST Finite Element Analysis and Solution Tools

- under development at TU Dortmund in Stefan Turek's group
- http://www.feast.tu-dortmund.de

Core features

- separation of unstructured and structured data for optimised linear algebra components
- Finite Element discretisations (Q_1)
- parallel generalised domain decomposition multigrid solvers
- usage of GPUs as coprocessors
- grid adaptivity and error control
- scalar and vector-valued problems
- applications in CFD and CSM

FEAST overview

FEAST Finite Element Analysis and Solution Tools

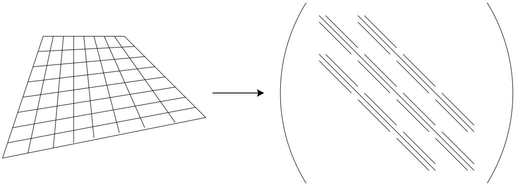
- under development at TU Dortmund in Stefan Turek's group
- http://www.feast.tu-dortmund.de

Core features

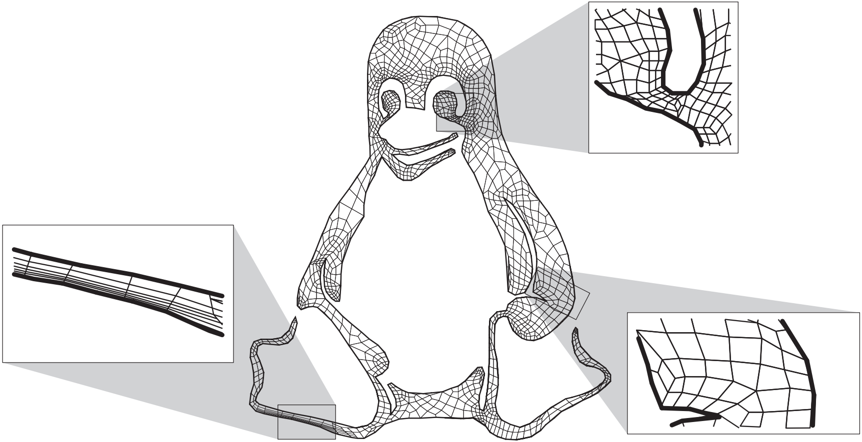
- separation of unstructured and structured data for optimised linear algebra components
- Finite Element discretisations (Q_1)
- parallel generalised domain decomposition multigrid solvers
- usage of GPUs as coprocessors
- grid adaptivity and error control
- scalar and vector-valued problems
- applications in CFD and CSM

local band matrices

- Poisson equation
- generalised tensor product mesh
- conforming bilinear Finite Elements Q_1
- matrix consists of 9 bands



FEAST grids



data structures

Cover domain by unstructured collection of subdomains

- resolve complex geometries, boundary layers in fluid dynamics, etc.

Refine each subdomain independently and discretise using FEs

- generalised tensorproduct fashion
- isotropic and anisotropic refinement combined with r/h/rh adaptivity

Performance

- clear separation of globally unstructured and locally structured parts
- nonzero pattern of local FE matrices known a priori
- exploit spatial and temporal locality for tuned LA building blocks (Sparse Banded BLAS)

Maximise computational efficiency per node.

ScaRC solver

Contradictory properties

- numerical vs. computational efficiency
- weak and strong scalability vs. numerical scalability

Parallel multigrid

- strong recursive coupling optimal in serial codes
- usually relaxed to block-Jacobi due to high comm requirements
- degrades convergence rates in the presence of local anisotropies

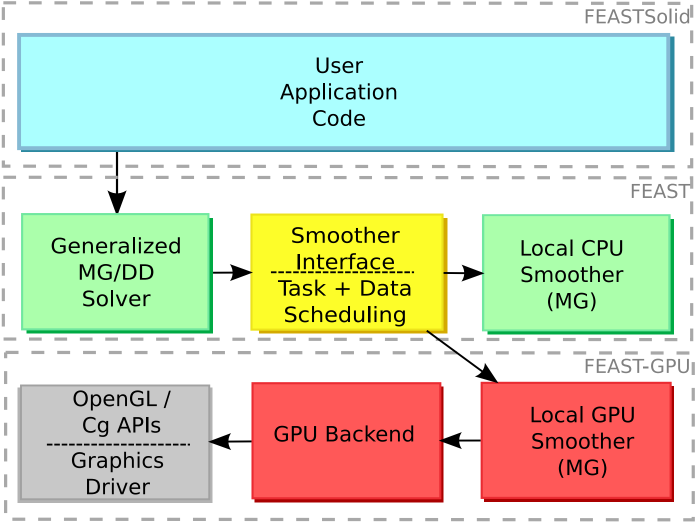
Generalised DD/MG approach (ScaRC)

- global MG, block-smoothed by local MGs (optimal asymptotic complexity)
- hide anisotropies locally
- good scalability by design
- global operations realised via special local BCs and syncronisation across subdomain boundaries (no overlap!)

Use iterative solvers featuring

- optimal asymptotic complexity
- robustness
- weak scalability

Overview of GPU integration



Petascale strategy III

Use specialised hardware when available.

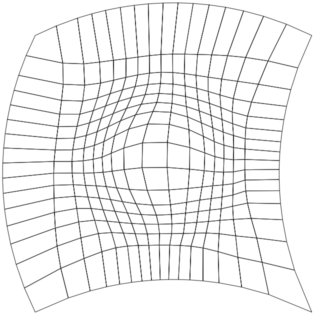
1. FEAST concepts
2. MFLOP/s-rate preserving adaptivity
3. Applications in structural mechanics

general idea

The performance of FEAST requires locally structured grids

elementwise h-adaptivity implies locally unstructured grids

$\Rightarrow$ relocate the grid points preserving the local tensor product structure



- domain Ω
- triangulation $\mathcal{T}$, quads T
- "monitor function" $0 < \varepsilon_f < f \in \mathcal{C}^1(\bar{\Omega})$: desired area distribution
- "weight function" $0 < \varepsilon_g < g \in \mathcal{C}^1(\bar{\Omega})$: current area distribution

goal: mapping $\Phi : \Omega \to \Omega$ with

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

and

$$\Phi : \partial\Omega \to \partial\Omega.$$

$$\mathcal{T}^d = \Phi(\mathcal{T}), \quad X := \Phi(x)$$

Deformation$(f, \mathcal{T})$

compute $\tilde{f} - \tilde{g}, \quad \tilde{f} := c/f, \tilde{g} = C/g, \int \tilde{f} \stackrel{!}{=} \int \tilde{g}$

solve $-\mathrm{div}(v(x)) = \tilde{f}(x) - \tilde{g}(x),\ x \in \Omega, \qquad v(x) \cdot \mathfrak{n} = 0,\ x \in \partial\Omega$

DO FORALL $x \in \mathcal{T}$

solve

$$\partial_t \varphi(x,t) = \frac{v(\varphi(x,t),t)}{t\tilde{f}(\varphi(x,t)) + (1-t)\tilde{g}(\varphi(x,t))}, \quad 0 \le t \le 1,\ \varphi(x,0) = x$$

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

ENDDO

END Deformation

realisation: $v := \nabla w \Rightarrow -\Delta w = \tilde{f} - \tilde{g}, \quad \partial_{\mathfrak{n}} w = 0$ on $\partial\Omega$

total amount:

1 Poisson problem + $2N$ decoupled IVPs

Deformation$(f, \mathcal{T})$

compute $\tilde{f} - \tilde{g}$, $\quad \tilde{f} := c/f, \tilde{g} = C/g, \int \tilde{f} \stackrel{!}{=} \int \tilde{g}$

solve $-\mathrm{div}(v(x)) = \tilde{f}(x) - \tilde{g}(x),\ x \in \Omega, \qquad v(x) \cdot \mathfrak{n} = 0,\ x \in \partial\Omega$

DO FORALL $x \in \mathcal{T}$

solve
$$\partial_t \varphi(x,t) = \frac{v(\varphi(x,t),t)}{t\tilde{f}(\varphi(x,t)) + (1-t)\tilde{g}(\varphi(x,t))}, \quad 0 \le t \le 1,\ \varphi(x,0) = x$$

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

ENDDO

END Deformation

realisation: $v := \nabla w \Rightarrow -\Delta w = \tilde{f} - \tilde{g}, \quad \partial_{\mathfrak{n}} w = 0$ on $\partial\Omega$

total amount:

1 Poisson problem + $2N$ decoupled IVPs

deformation method

Deformation$(f, \mathcal{T})$

compute $\tilde{f} - \tilde{g}$, $\quad \tilde{f} := c/f, \tilde{g} = C/g, \int \tilde{f} \stackrel{!}{=} \int \tilde{g}$

solve $-\mathrm{div}(v(x)) = \tilde{f}(x) - \tilde{g}(x),\ x \in \Omega, \qquad v(x) \cdot \mathfrak{n} = 0,\ x \in \partial\Omega$

DO FORALL $x \in \mathcal{T}$

solve
$$\partial_t \varphi(x,t) = \frac{v(\varphi(x,t),t)}{t\tilde{f}(\varphi(x,t)) + (1-t)\tilde{g}(\varphi(x,t))}, \quad 0 \le t \le 1,\ \varphi(x,0) = x$$

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

ENDDO

END Deformation

realisation: $v := \nabla w \Rightarrow -\Delta w = \tilde{f} - \tilde{g}, \quad \partial_{\mathfrak{n}} w = 0$ on $\partial\Omega$

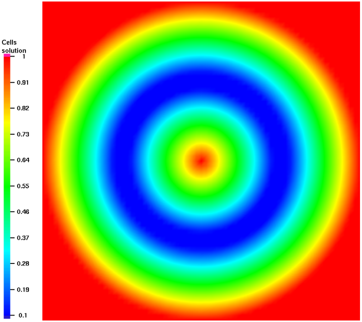
total amount:

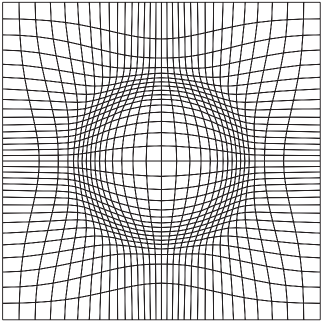
1 Poisson problem $+ 2N$ decoupled IVPs

test example

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

monitor function: $f(x) = \min\left(1, \max(\frac{|d-0.25|}{0.25}, \epsilon)\right)$,
$d := \sqrt{(x_1 - 0.5)^2 + (x_2 - 0.5)^2}$, $\epsilon = 1/10$





r-AFEM

$GRID_1 := GRID$

DO $i = 1, i_{\max}$

$u_i :=$ **SOLVE**$(f, g, GRID_i)$

$\eta_i :=$**ESTIMATE**(u_i, J)

IF $(\eta_i < TOL)$ EXIT LOOP

$f_{mon,i} :=$**MON**(η_i)

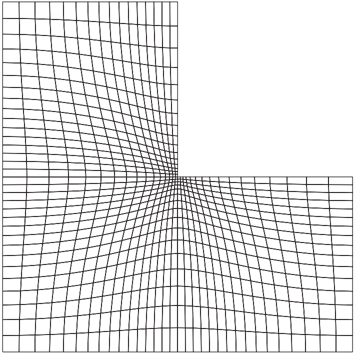
$GRID_{i+1} :=$ **DEFORM**$(f_{mon,i}, GRID_i)$
IF ($\exists$ non-convex elements) RETURN

END DO

$J(u_h) := J(u_i); \eta := \eta_i$

RETURN $J(u_h), \eta$

END r-AFEM

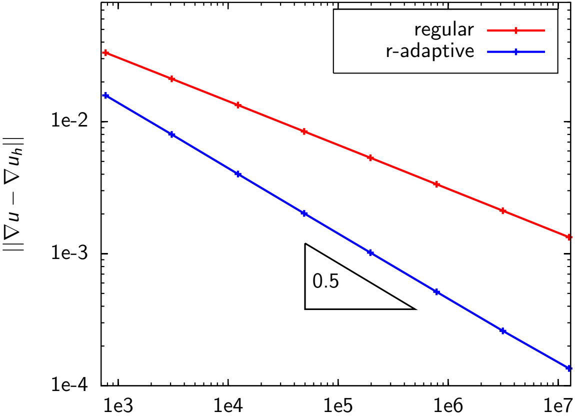


Poisson equation

$$\Omega = [-0.5, 0.5]^2 / [0, 0.5]^2$$

$$u(r, \varphi) = r^{2/3} \sin(2/3\varphi)$$

goal: gradient error



optimal convergence rates by *r*-adaptivity

details: M. Grajewski, A new fast and accurate grid deformation method for *r*-adaptivity in the context of high performance computing, PhD thesis, 2008

Raise discretisation efficiency by adaptivity without interfering with strategies I-III.

Overview

technische universität
dortmund

1. FEAST concepts
2. MFLOP/s-rate preserving adaptivity
3. Applications in structural mechanics

technische universität dortmund

Computational Solid Mechanics Application FEASTsolid

Fundamental model problem:

- elastic, compressible material
- small deformations, static loading process
- Hooke's material law

Lamé equation

$$\begin{aligned} -2\mu \mathrm{div} \varepsilon(u) - \lambda \nabla(\mathrm{div} u) &= f, \quad x \in \Omega \\ u &= g, \quad x \in \Gamma_D \\ \sigma(u) \cdot n &= t, \quad x \in \Gamma_N \end{aligned}$$

Separate displacement ordering

$$-\begin{pmatrix} (2\mu + \lambda)\partial_{xx} + \mu\partial_{yy} & (\mu + \lambda)\partial_{xy} \\ (\mu + \lambda)\partial_{xy} & \mu\partial_{xx} + (2\mu + \lambda)\partial_{yy} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}$$

Discretisation of reordered Lamé equation

block-structured system

$$\begin{pmatrix} K_{11} & K_{12} \\ k_{21} & K_{22} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}$$

Coupling of K_{11}, K_{21}, K_{22}

- block Gauß-Seidel smoothing of global multigrid solver
- $K_{11} u_1 = f_1$ and $K_{22} u_2 = f_2$ correspond to scalar elliptic equations $\Rightarrow$ ScaRC

Solver specialisation

- global BiCGStab (vector-valued) preconditioned by
- global multigrid (vector-valued) block-GS-smoothed by
- local multigrids (scalar) per subdomain

Test goals

Accuracy

- analytic reference solution
- global anisotropies to worsen condition numbers

Scalability

- here: only weak scalability

Speedup

- exemplarily for some test scenarios

Test case: Cantilever beam

Global anisotropy and ratio of fixed Dirichlet and free Neumann BCs proportional to # of processors

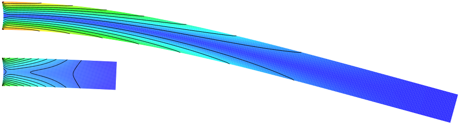
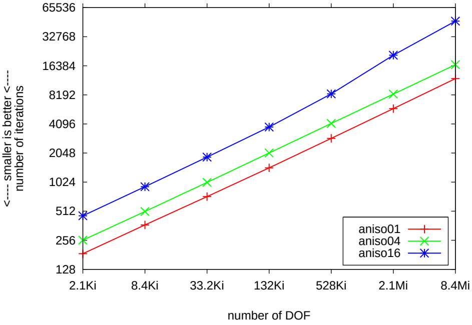


Illustration of ill-conditioning

- plain Conjugate gradient solver
- anisotropies of 1:1, 1:4 and 1:16
- plot: Number of iterations for increasing problem size, logscale
- Aniso16 does not even exhibit doubling of iterations any more



GPU-CPU comparison

aniso04 refinement L	iterations CPU	GPU	volume CPU	GPU	y-displacement CPU	GPU
8	4	4	1.6087641E-3	1.6087641E-3	-2.8083499E-3	-2.8083499E-3
9	4	4	1.6087641E-3	1.6087641E-3	-2.8083628E-3	-2.8083628E-3
10	4.5	4.5	1.6087641E-3	1.6087641E-3	-2.8083667E-3	-2.8083667E-3
aniso16						
8	6	6	6.7176398E-3	6.7176398E-3	-6.6216232E-2	-6.6216232E-2
9	**6**	**5.5**	6.7176427E-3	6.7176427E-3	-6.621655**1**E-2	-6.621655**2**E-2
10	5.5	5.5	6.7176516E-3	6.7176516E-3	-6.621750**1**E-2	-6.621750**2**E-2

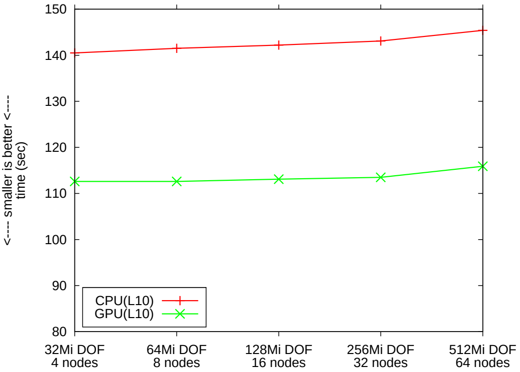
Same solution for GPU and CPU

- volume of deformed body
- displacement of reference point at tip of the beam
- same number of iterations until convergence

technische universität dortmund

Good scalability

original and accelerated CSM solver Infiniband, Xeon EM64T, 3.4GHz, outdated Quadro 1400 GPU

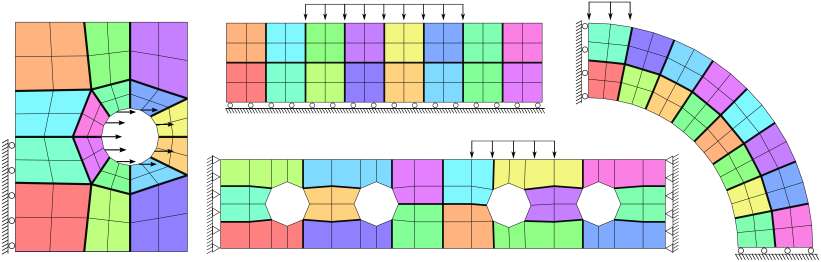


More results

Poisson problem for 1.3 billion unknowns in less than 50 seconds on 160 outdated GPUs (Quadro 1400)

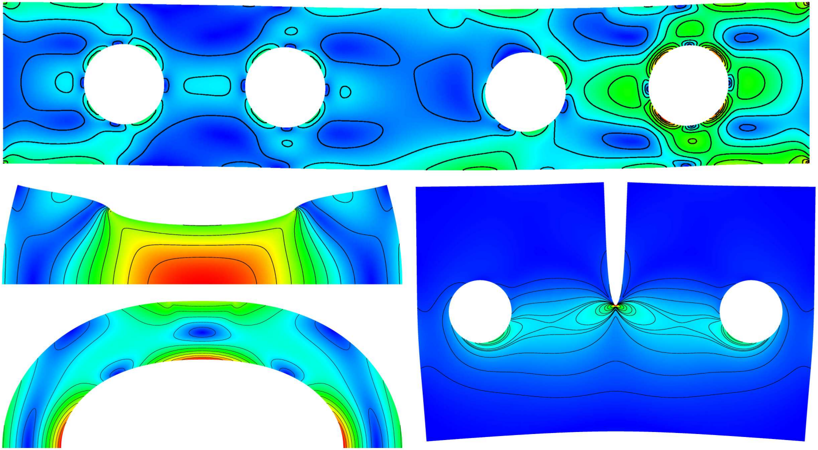
Paper: Göddeke et al., Exploring weak scalability for FEM calculations on a GPU-enhanced cluster, Parallel Computing 33(10-11), 685-699, 2007

Test configurations

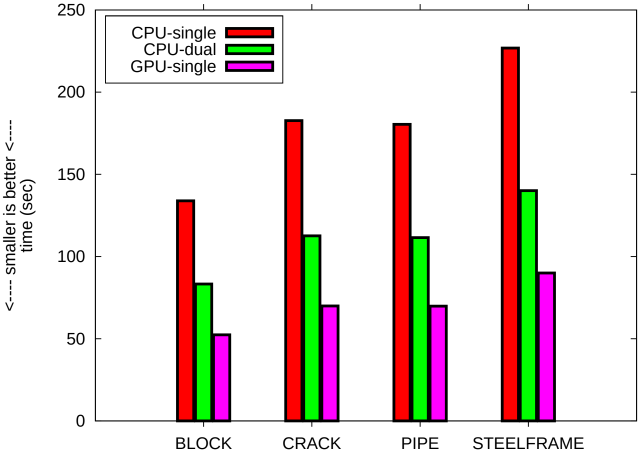


Test system with 16 nodes: dualcore Santa Rosa Opteron CPU, Quadro 5600 GPU, Infiniband

van Mises stresses



speed ups


CPU-single
CPU-dual
GPU-single
<---- smaller is better <----
time (sec)
0
50
100
150
200
250
BLOCK
CRACK
PIPE
STEELFRAME

Summary

- Finite Element code
- local band matrices
- generalised DD/MG approach (ScaRC)
- GPU integration

- MFLOP/s preserving adaptivity
- grid deformation

- model problem: Lamé equation
- numerical tests
- accuracy
- scalability
- speedup

- www.feast.tu-dortmund.de

Thank you for your attention!