Proceedings of two courses on

COMPUTATIONAL MECHANICS I (elastoplasticity)

May 10-14, 2010

and

COMPUTATIONAL MECHANICS II

December 1 - 3, 2010

held under the project

Educational and scientific collaboration CR-Iceland:
computational mechanics, geothermal energy and further applications
(supported by EEP/Norway funds).

at Institute of Geonics AS CR Ostrava

Project organizer: Prof. Dr. Radim Blaheta (Institute of Geonics AS CR Ostrava)
Assoc. Prof. Dr. Jan Valdman (University of Iceland)

Preface

These proceedings resulted from two courses - Computational Mechanics I and II, which were held at the Institute of Geonics AS CR with many participants from VSB - Technical University of Ostrava in May and December 2010.

The first course was focused on computational plasticity and lectured by Jan Valdman from the University of Iceland. The second course covers broader scope of topics from computational mechanics – damage mechanics, fluid dynamics, acoustics, contact problems, homogenization, solving problems of identification of material parameters, preconditioning for saddle point systems and aspects of aposteriori error computation, FEM, BEM, adaptiveness and parallel computing. The presented topics have lot of applications, among other covering geothermal energy and other geo engineering fields.

The courses Computational Mechanics I and II were supported by EEP/Norway funds via the project BG FTA 4th call - EČ 049-4V- Collaboration of CR and Island on education and science: Computational mechanics, geothermal energy and other applications.

This support and the work of all lecturers from Iceland, Sweden and the Czech Republic is highly appreciated.

Prof. Radim Blaheta
project coordinator

COMPUTATIONAL MECHANICS I (elastoplasticity)

May 10-14, 2010

activity: 5 lectures and 3 computer exercises designed for students, PhD students and researchers given by Jan Valdman (University of Iceland) on Mathematical modelling of elastoplasticity.

topics:

- modelling of elastoplasticity using rheological models
- introduction to theory of variational inequalities and their discretization with Finite Elelement method
- implementation details of a time-dependent two-dimentional problem in Matlab

COMPUTATIONAL MECHANICS II

December 1 - 3, 2010

activity: 13 lectures given by 11 speakers

participants and their talks (ordered according to schedule):

- D. Lukáš (VSB-TU) - Parallel BEM-based methods
- J. Valdman (U Iceland) - Aposteriori error estimates (3 talks)
- H. Palsson (U Iceland) - Computational fluid dynamics with OpenFOAM (2 talks)
- O. Axelsson (I Geonics) - Preconditioning for saddle point problems
- M. Neytcheva et al. (U Uppsala) - On an augmented Lagrangian-based preconditioning of Oseen type problems
- J. Kruis (CTU Prague) - Damage mechanics analysis with arc-length solver
- Z. Dostal (VSB-TU) - T-FETI based scalable algorithms for contact problems
- O. Vlach (VSB-TU) - Quasistatic contact problems
- T. Kozubek (VSB-TU) - Scalable algorithms for dynamics contact problems
- V. Vondrak (VSB-TU) - Efficient parallel contact shape optimization
- A. Markopoulos (VSB-TU) - MATSOL - implementation methods
- R. Blaheta et al. (I Geonics) - Computational (geo) micromechanics,
Identification of material parameters (2 talks)
- S. Sysala (I Geonics) - Elasto-plasticity: selected topics

Original posters to both conferences (in czech and english),

slides to COMPUTATIONAL MECHANICS I (elastoplasticity)

by Jan Valdman

and

some abstracts to COMPUTATIONAL MECHANICS II

follow.

Additional information can be found on web pages of Institute of Geonics AS CR Ostrava

http://www.ugn.cas.cz/?l=en&a=&p=events/2010/kvp/index.php

http://www.ugn.cas.cz/?l=en&a=&p=events/2010/wcm/index.php

Srdečně zveme všechny zájemce na týdenní kurz

VÝPOČETNÍ PLASTICITA

přednášející dr. JAN VALDMAN z University of Iceland, Reykjavík
Kurz je podporovaný fondy EHP/Norska a je určen
pro studenty, doktorandy a další zájemce.

Účelem kurzu je popis základních mechanických modelů v elastoplasticitě a jejich matematického modelování. Speciální důraz bude věnován numerickým metodám a jejich realizaci v Matlabu.

Témata **přednášek** zahrnují:

- modelování elastoplasticity pomocí reologických modelů
- úvod do teorie variačních nerovnic a jejich diskretizace pomocí metody konečných prvků (FEM)
- implementace řešiče časově závislé 2D úlohy v Matlabu

Cílem navazujícího **počítačového praktika** je:

- implementovat jednoduché úlohy lineární elasticity pomocí metody konečných prvků
- rozšířit je pro případ elastoplastického modelu včetně časové závislosti

Pro práci v počítačovém praktiku se předpokládá vlastní počítač. Vlastní MATLAB (v. 7 a výše) je vítán. K dispozici bude WiFi přístup k omezenému počtu licencí.

Program:

Zahájení **10. 5. 2010, 13:00**, J. Valdman, úvodní přednáška

Pokračování kurzu **11. - 14. 5. 2010** denně:

9:00-10:30	přednáška
10:30-11:00	diskuze, občerstvení
11:00-12:30	počítačové praktikum: tvorba a využití SW (MATLAB)

Přednášky i počítačové praktikum se budou konat v **konferenční místnosti Ústavu geoniky AV ČR**, Studentská 1768, areál VŠB TU Ostrava Poruba.

ÚGN AV ČR a KAM FEI VŠB TU Odborný garant: Prof. Radim Blaheta

You are invited to participate at the course

COMPUTATIONAL MECHANICS II

held under the project „Educational and scientific collaboration CR-Iceland: computational mechanics, geothermal energy and further applications“ (supported by EEP/Norway funds).

The course is a free continuation of the course Computational Mechanics I (plasticity) and is designed for students, PhD students and researchers.

Program

1.12. 2010 from 14.00

Opening
D. Lukáš (VŠB-TU) **Parallel BEM-based methods**
J. Valdman (U Iceland) **Aposteriori error estimates (1)**
H. Palsson (U Iceland) **Computational fluid dynamics with OpenFOAM(1)**

2.12. 2010 from 9.00

J. Valdman (U Iceland) **Aposteriori error estimates (2)**
H. Palsson (U Iceland) **Computational fluid dynamics with OpenFOAM(2)**
O. Axelsson (I Geonics) **Preconditioning for saddle point problems**
M. Neytcheva (U Uppsala) **Augmented Lagrangian preconditioning**

from 13.00

J. Kruis (CTU Prague) **Damage mechanics analysis with arc-length solver**
Z. Dostál (VŠB-TU) **T-FETI based scalable algorithms for contact problems**
O. Vlach (VŠB-TU) **Quasistatic contact problems**
T. Kozubek (VŠB-TU) **Scalable algorithms for dynamics contact problems**
V. Vondrák (VŠB-TU) **Efficient parallel contact shape optimization**
A. Markopoulos (VŠB-TU) **MATSOL - implementation methods**

3.12. 2010 from 9.00

J. Valdman (U Iceland) **Aposteriori error estimates (3)**
R. Blaheta et al. (I Geonics) **Computational (geo) micromechanics**
S. Sysala (I Geonics) **Elasto-plasticity: selected topics**
R. Blaheta et al. (I Geonics) **Identification of material parameters**

Lectures of the length of 30 or 45 minutes will be held in the **conference room of the Institute Geonics AS CR,** Studentská 1768, Ostrava Poruba.

Details: www.ugn.cas.cz. On behalf of the organizers: R. Blaheta, T. Kozubek.

COMPUTATIONAL MECHANICS I

Jan Valdman

- modelling of elastoplasticity using rheological models
- introduction to theory of variational inequalities and their discretization with Finite Elelement method
- implementation details of a time-dependent two-dimentional problem in Matlab

Mathematical modelling of elastoplasticity

Jan Valdman

School of Engineering and Natural Sciences
University of Iceland, Reykjavik
email: jan.valdman@gmail.com

Ostrava, May 10-14, 2010

Outline

Explaining papers to theory and numerics:

- Carsten Carstensen, Martin Brokate, Jan Valdman, A quasi-static boundary value problem in multi-surface elastoplasticity. I: Analysis. Math. Methods Appl. Sci. 27, No.14, 1697-1710 (2004), web link
- Carsten Carstensen, Martin Brokate, Jan Valdman, A quasi-static boundary value problem in multi-surface elastoplasticity. II: Numerical solution. Math. Methods Appl. Sci. 28, No.8, 881-901 (2005), web link
- Andreas Hofinger, Jan Valdman, Numerical solution of the two-yield elastoplastic minimization problem. Computing 81, No. 1, 35-52 (2007), web link
- Peter Gruber, Jan Valdman, Solution of one-time-step problems in elastoplasticity by a Slant Newton Method. SIAM J. Scientific Computing 31, No. 2, 1558-1580 (2009), web link.

Elastoplasticity solver can be downloaded at
http://www.mathworks.com/matlabcentral/fileexchange/authors/37756
as a package called 'Two-yield elastoplasticity solver'

Further papers on a posteriori error estimates in elastoplasticity:

- Antonio Orlando, Carsten Carstensen, Jan Valdman, A convergent adaptive finite element method for the primal problem of elastoplasticity. International Journal for Numerical Methods in Engineering 67, No. 13, 1851-1887 (2006), web link
- Sergey Repin, Jan Valdman, Functional a posteriori error estimates for problems with nonlinear boundary conditions. Journal of Numerical Mathematics 16, No. 1, 51-81 (2008), web link

The tensile test

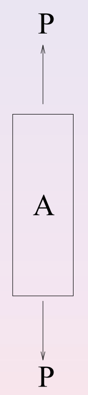


Figure: The tensile test: an increasing stress $\sigma = P/A$ is applied to the specimen.

The tensile test: stress-strain relation

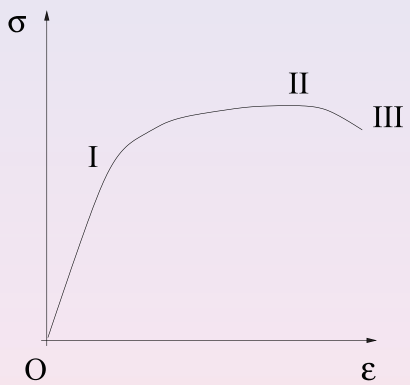
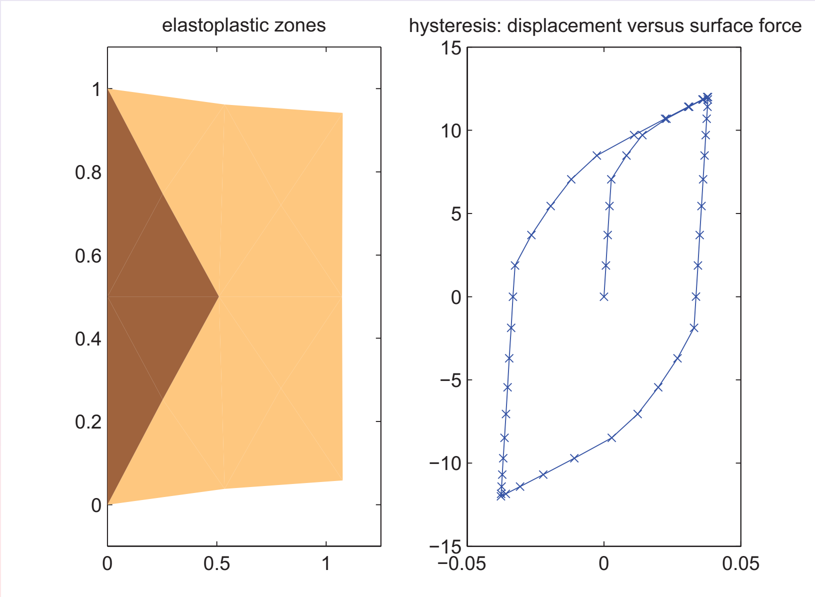


Figure: The tensile test: the resulting stress-strain relation.

- elasticity in the region $O - I$
- plasticity with hardening after the elastic limit (point I)
- softening after necking (point II) untill fractures occur (point III)

Time dependent 2D problem in Matlab



Rheological elements

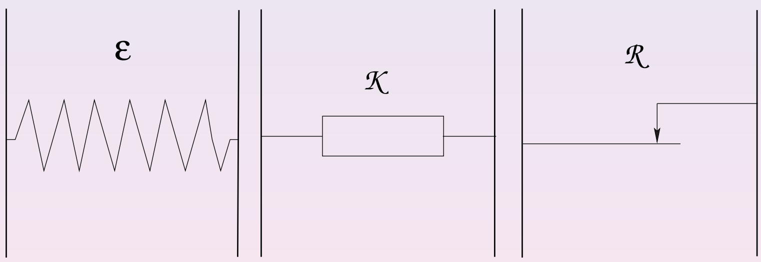


Figure: The elastic, kinematic and rigid-plastic element.

Rheological elements

Every element is characterized by its (internal) stress and strain tensors. We denote the stress by σ and the strain by ε.

The elastic element

$$\sigma = \mathbb{C}\varepsilon$$

The kinematic element

$$\sigma = \mathcal{H}\varepsilon,$$

where $\mathcal{H}$ is a positive definite matrix, for instance $\mathcal{H} = h\mathbb{I}$, where $h > 0$ is a hardening coefficient and $\mathbb{I}$ represents the identical matrix.

Rheological elements

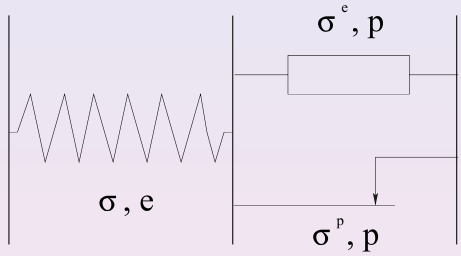
The rigid-plastic element

$$\begin{aligned} &\sigma \in Z \\ &\langle \dot{\varepsilon}, q - \sigma \rangle \leq 0 \qquad \text{for all } q \in Z \end{aligned}$$

with a convex set $Z \subset \mathbb{R}^{d \times d}_{sym}$.

Example: 1D

Kinematic hardening model



$$
\begin{aligned}
&\varepsilon = \mathrm{e} + p \\
&\sigma = \sigma^e + \sigma^p \\
&\sigma^e = \mathcal{H} p \\
&\sigma = \mathbb{C}\,\mathrm{e} \\
&\sigma^p \in Z \\
&\langle \dot{p}, q - \sigma^p \rangle \leq 0 \quad \text{for all } q \in Z.
\end{aligned}
$$

Hysteresis property of the kinematic hardening model

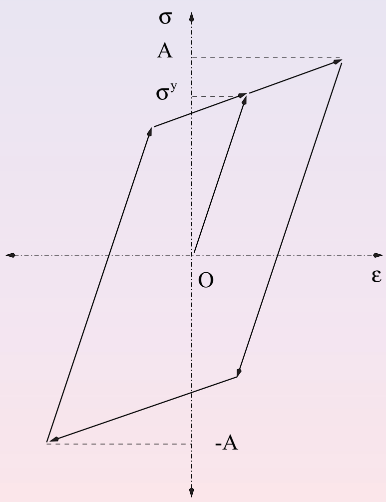


Figure: Stress-strain relation in case of linear kinematic hardening model and the cyclic stress $\sigma = A\sin(t)$.

Motivation for the multi-yield model

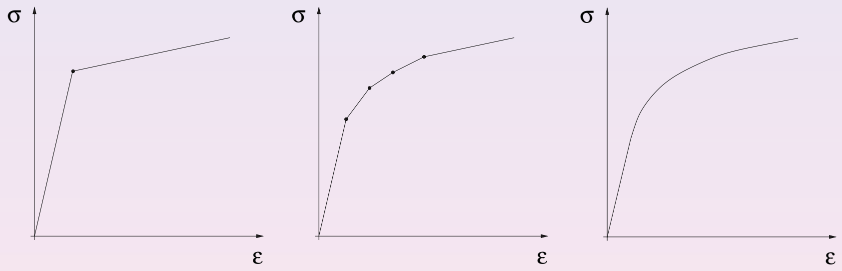
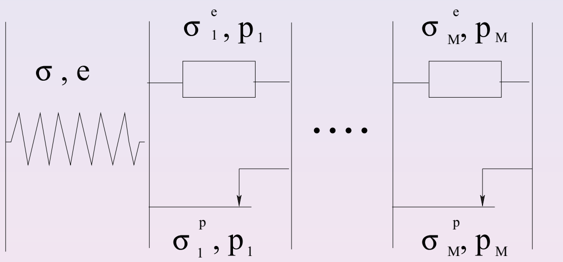


Figure: single-yield (left), multi-yield (middle) and realistic model (right) - stress-strain relation.

The M-yield hardening model



$$\varepsilon = \mathrm{e} + p, \quad p = \sum_{r=1}^{M} p_r,$$
$$\sigma = \sigma_r^e + \sigma_r^p \quad \text{for all } r = 1, \dots, M,$$
$$\sigma_r^p \in Z_r,$$
$$\langle \dot{p}_r, q_r - \sigma_r^p \rangle \leq 0 \quad \text{for all } q_r \in Z_r, r = 1, \dots, M,$$
$$\sigma = \mathbb{C}\,\mathrm{e},$$
$$\sigma_r^e = \mathcal{H}_r p_r, \quad r = 1, \dots, M.$$

Hysteresis property of the 2-yield hardening model

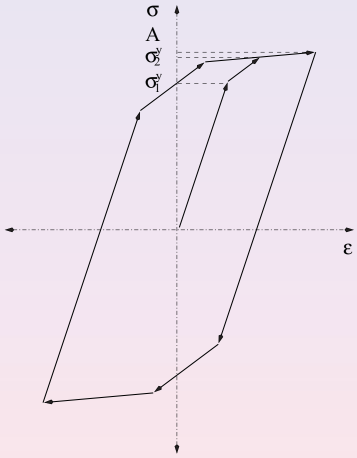


Figure: Stress-strain relation in case of two-yield model and cyclic stress $\sigma = A\sin(t)$.

Books on hysteresis

- Visintin, A., Differential models of hysteresis, Springer, 1994
- Brokate, M. and Sprekels, J., Hysteresis and Phase Transitions, Springer-Verlag New York, 1996
- Krejčí, P., Hysteresis, Convexity and Dissipation in Hyperbolic Equations, GAKUTO International Series, Mathematical Sciences and Applications, 1996

Yield criterion

von Mises criterion

$$Z = \{\sigma \in \mathbb{R}^{d\times d}_{sym} : ||\operatorname{dev}\sigma||_F \le \sigma^y\},$$

where $||\cdot||_F$ denotes the Frobenius matrix norm $||a||_F^2 = a:a = \sum_{i,j=1}^d a_{ij}^2$,
$\operatorname{dev}\sigma = \sigma - \frac{1}{d}\operatorname{tr}(\sigma)\mathbb{I}$ is the deviatoric operator (deviator),
$\operatorname{tr}\sigma = \sigma:\mathbb{I}$ is the trace operator.

Dissipation functional

Lemma

Let $(\dot p, \sigma^p) \in \mathbb{R}^{d\times d}_{sym} \times \mathbb{R}^{d\times d}_{sym}$. *Then*

$$\sigma^p \in Z, \quad \dot p : (\tau - \sigma^p) \le 0 \quad \text{for all } \tau \in Z \quad (*)$$

together with $\operatorname{tr}\dot p = 0$ *hold if and only if*

$$\sigma^p : (q - \dot p) \le \mathcal{D}(q) - \mathcal{D}(\dot p) \quad \forall q \in \mathbb{R}^{d\times d}_{sym}, \quad (**)$$

where $\mathcal{D} : \mathbb{R}^{d\times d}_{sym} \to \mathbb{R} \cup \{\infty\}$,

$$\mathcal{D}(q) = \begin{cases} \sigma^y ||q|| & \text{if } \operatorname{tr} q = 0, \\ +\infty & \text{otherwise.} \end{cases}$$

Proof: together only implication $(*) \Rightarrow (**)$.

Some convex analysis

Definition (indicator function)

For any set $Z \subset X$, the *indicator function* I_Z of Z is defined by

$$I_Z(x) = \begin{cases} 0 & \text{if } x \in Z, \\ +\infty & \text{if } x \notin Z. \end{cases} \tag{1}$$

Definition (subdifferential)

Let f be a convex function on X. For any $x \in X$ the *subdifferential* $\partial f(x)$ of x is the possibly empty subset of X^* defined by

$$\partial f(x) = \{x^* \in X^* : \langle x^*, y - x\rangle \leq f(y) - f(x) \quad \forall y \in X\}. \tag{2}$$

It means that

$$\dot{p} \in \partial I_Z(\sigma^p).$$

Some convex analysis

Definition (conjugate function)

For a function $f : X \to [-\infty, \infty]$ we define the *conjugate function* $f^* : X^* \to [-\infty, \infty]$ by

$$f^*(x^*) = \sup_{x \in X}(\langle x^*, x\rangle - f(x)). \tag{3}$$

Lemma

Let X be a Banach space, $f : X \to [-\infty, \infty]$ be a proper, convex, lower semicontinuous function. Then

$$x^* \in \partial f(x) \Leftrightarrow x \in \partial f^*(x^*). \tag{4}$$

Therefore,

$$\dot{p} \in \partial I_Z(\sigma^p) \Leftrightarrow \sigma^p \in \partial I_Z^*(\dot{p})$$

and

$$D(\cdot) := I_Z^*(\cdot).$$

Equilibrium and its weak formulation

The equilibrium between external and internal forces is given by

$$\operatorname{div} \sigma(x,t) + f(x,t) = 0, \quad x \in \Omega, \quad t \in (0,T). \tag{5}$$

With the assumption of small deformations

$$\varepsilon(v) = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right),$$

the variational formulation of (14) becomes (why?)

$$\int_\Omega \sigma : \varepsilon(v) \,\mathrm{d}x = \int_\Omega f \cdot v \,\mathrm{d}x + \int_{\Gamma_N} g \cdot v \,\mathrm{d}s, \tag{6}$$

valid for all $t \in [0,T]$ and all $v \in H^1_D(\Omega)$.

Weak formulation of rigid-plastic elements

We express constitutive laws

$$\sigma_r^p : (q_r - \dot{p}_r) \le \mathcal{D}_r(q_r) - \mathcal{D}_r(\dot{p}_r) \quad \forall\, q_r \in Q\,,\; r \in I\,, \tag{7}$$

where (note that we only consider arguments with zero trace here)

$$\mathcal{D}_r(q_r) = \sigma_r^y ||q_r||\,.$$

The integral form of (7) over Ω is given by

$$\int_\Omega \sigma_r^p : (q_r - \dot{p}_r) \,\mathrm{d}x \le \int_\Omega \mathcal{D}_r(q_r) \,\mathrm{d}x - \int_\Omega \mathcal{D}_r(\dot{p}_r) \,\mathrm{d}x. \tag{8}$$

Variational inequality

We sum the inequalities (8) over r
and subtract (6) in which we equivalently replace v by $v - \dot{u}$
to obtain

$$\int_\Omega \sigma : (\varepsilon(v) - \sum_{r\in I} q_r))\,\mathrm{d}x - \int_\Omega \sigma : (\varepsilon(\dot{u}) - \sum_{r\in I} \dot{p}_r)\,\mathrm{d}x + \sum_{r\in I} \int_\Omega \sigma_r^e : (q_r - \dot{p}_r)\,\mathrm{d}x$$
$$+ \sum_{r\in I} \int_\Omega \mathcal{D}_r(q_r)\,\mathrm{d}x - \sum_{r\in I} \int_\Omega \mathcal{D}_r(\dot{p}_r)\,\mathrm{d}x - \int_\Omega f \cdot (v - \dot{u})\,\mathrm{d}x - \int_{\Gamma_N} g \cdot (v - \dot{u})\,\mathrm{d}s \geq 0.$$

Next, we eliminate

$$\sigma = \mathbb{C}(\varepsilon(u) - p), \quad \sigma_r^e = \mathcal{H}_r p_r.$$

Variational inequality

We collect vectors of functions

$$w = (u, (p_r)_{r\in I}), \quad z = (v, (q_r)_{r\in I}).$$

to obtain

Problem (BVP of quasi-static multi-surface elastoplasticity)

For given $\ell \in H^1(0,T;\mathcal{H}^*)$ *with* $\ell(0) = 0$,
find $w \in H^1(0,T;\mathcal{H})$ *with* $w(0) = 0$, *such that*

$$a(w(t), z - \dot{w}(t)) + \psi(z) - \psi(\dot{w}(t)) \geq \langle \ell(t), z - \dot{w}(t) \rangle, \quad \text{for all } z \in \mathcal{H},$$

holds for almost all $t \in (0,T)$.

Variational inequality

A bilinear form $a(\cdot,\cdot)$, a linear functional $\ell(\cdot)$ and a nonlinear functional $\psi(\cdot)$ are defined as

$$a : \mathcal{H} \times \mathcal{H} \to \mathbb{R}, \quad a(w,z) = \int_\Omega \mathbb{C}(\varepsilon(u) - \sum_{r\in I} p_r) : (\varepsilon(v) - \sum_{r\in I} q_r)\, \mathrm{d}x + \\ + \sum_{r\in I} \int_\Omega \mathcal{H}_r p_r : q_r \, \mathrm{d}x,$$

$$\ell(t) : \mathcal{H} \to \mathbb{R}, \quad \langle \ell(t), z\rangle = \int_\Omega f(t)\cdot v\, \mathrm{d}x + \int_{\Gamma_N} g(t)\cdot v\, \mathrm{d}s,$$

$$\psi : \mathcal{H} \to \mathbb{R}, \quad \psi(z) = \sum_{r\in I} \int_\Omega \mathcal{D}_r(q_r)\, \mathrm{d}x.$$

and $\mathcal{H} = H^1_D(\Omega) \times \prod_{r\in I} Q$.

Rheological models Variational inequalities Existence FE Discretization

Further reading

- Glowinskii, R., Lions J. L. and Trémolières R., Numerical analysis of Variational Inequalities, North-Holland, Amsterdam, 1981
- Han, W. and Reddy, B., Plasticity: Mathematical Theory and Numerical Analysis, Springer-Verlag New York, 1999

Jan Valdman Computational elastoplasticity

Material assumptions

We pose the natural assumption that the elastic and hardening tensors are symmetric and positive definite,

$$\begin{aligned} &\xi : \mathbb{C}\lambda = \mathbb{C}\xi : \lambda \quad \text{for all } \xi, \lambda \in \mathbb{R}^{d\times d}, \\ &\xi : \mathcal{H}_r\lambda = \mathcal{H}_r\xi : \lambda \quad \text{for all } \xi, \lambda \in \mathbb{R}^{d\times d}, r = 1, \dots, M, \end{aligned} \tag{9}$$

and there exist constants $c, h_r > 0$ such that

$$\begin{aligned} &\mathbb{C}\xi : \xi \geq c||\xi||^2 \quad \text{for all } \xi \in \mathbb{R}^{d\times d}, \\ &\mathcal{H}_r\xi : \xi \geq h_r||\xi||^2 \quad \text{for all } \xi \in \mathbb{R}^{d\times d}, r = 1, \dots, M. \end{aligned} \tag{10}$$

Abstract theorem on solvability

Theorem

Assume that (9) and (10) hold, let $\ell \in H^1(0, T; \mathcal{H}^*)$ with $\ell(0) = 0$. Then there exists a unique solution $w \in H^1(0, T; \mathcal{H})$ of BVP of quasi-static multi-surface elastoplasticity.

based on

Abstract theorem on solvability

Theorem (Han, Reddy, 1999)

Let $\mathcal{H}$ be a Hilbert space, $a : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ be a bilinear form that is symmetric, bounded, and $\mathcal{H}$-elliptic; $\ell \in H^1(0, T; \mathcal{H}^*)$ with $\ell(0) = 0$; and $\psi : \mathcal{H} \to \mathbb{R}$ nonnegative, convex, positively homogeneous, and Lipschitz continuous. Then there exists a unique $w \in H^1(0, T; \mathcal{H})$ with $w(0) = 0$ which satisfies the variational inequality

$$a(w(t), z - \dot{w}(t)) + \psi(z) - \psi(\dot{w}(t)) \geq \langle \ell(t), z - \dot{w}(t) \rangle, \quad \text{for all } z \in \mathcal{H},$$

for almost all $t \in (0, T)$.

Remark on ellipticity

To prove that

$$a(w, z) = \int_\Omega \mathbb{C}(\varepsilon(u) - \sum_{r \in I} p_r) : (\varepsilon(v) - \sum_{r \in I} q_r) \, dx + \sum_{r \in I} \int_\Omega \mathcal{H}_r p_r : q_r \, dx,$$

is elliptic, the following partial result is important:

Problem

To determine the largest constant $k(M)$, $M \in \mathcal{N}$, such that

$$\left(x_0 - \sum_{r=1}^{M} x_r \right)^2 + \sum_{r=1}^{M} x_r^2 \geq k(M) \sum_{r=0}^{M} x_r^2 \tag{11}$$

holds for all $x_0, x_1, \ldots, x_M \in \mathbb{R}$.

Algebraic inequality

We refolmulate

$$\left(x_0 - \sum_{r=1}^{M} x_r\right)^2 + \sum_{r=1}^{M} x_r^2 = x^T A x, \tag{12}$$

where

$$A = D + a \otimes a\,, \quad D = \mathrm{diag}(0, 1, \ldots, 1)\,, \quad a = (1, -1, \ldots, -1)\,. \tag{13}$$

Thus, the optimal constant $k(M)$ is equal to the smallest eigenvalue of A!

Algebraic inequality

The analytical computation shows

$$k(M) = \lambda_{min} = 1 + \frac{M}{2} - \frac{1}{2}\sqrt{4M + M^2}$$

Properties:

$$\lim_{M\to\infty} k(M) = 0$$

and

$$\lim_{M\to\infty} Mk(M) = 1$$

Backward Euler scheme

In the first time step t_1, the time derivative $\dot{x}(t_1)$ is approximated by the backward Euler method as

$$\dot{X}^1 = \frac{X^1 - X^0}{k_1},$$

where $X^0 = 0$. The Hilbert space $\mathcal{H}$ is approximated by the conforming finite element (FEM) subspace

$$\mathcal{S} = \mathcal{S}^1_D(\mathcal{T}) \times \prod_{r \in I} \operatorname{dev}(\mathcal{S}^0(\mathcal{T})^{d\times d}_{\mathrm{sym}}),$$

which is a product space of $\mathcal{T}$- piecewise affine functions that are zero on Γ_D by

$$\mathcal{S}^1_D(\mathcal{T}) := \{v \in H^1_D(\Omega) : \forall T \in \mathcal{T}, v|_T \in \mathcal{P}_1(T)^d\}.$$

($\mathcal{P}_1(T)$ denotes the affine functions on T) and the space of $\mathcal{T}$- piecewise constant functions

$$\operatorname{dev}(\mathcal{S}^0(\mathcal{T})^{d\times d}_{\mathrm{sym}}) := \{a \in L^2(\Omega)^{d\times d} : \forall T \in \mathcal{T}, a|_T \in \operatorname{dev}\mathbb{R}^{d\times d}_{\mathrm{sym}}\}$$

Backward Euler scheme

The first time step problem

Find $X^1 = (U^1, (P^1_r)_{r\in I}) := (U^1, P^1) \in \mathcal{S}$ such that

$$\langle \ell(t_1), (Y - \frac{X^1 - X^0}{k_1})\rangle \le a(X^1, Y - \frac{X^1 - X^0}{k_1}) + \psi(Q) - \psi(\frac{P^1 - P^0}{k_1}).$$

holds for all $Y = (V, Q) = (V, (Q_r)_{r\in I}) \in \mathcal{S}$.

After introducing an incremental variable $X := (U, P) = X^1 - X^0$ and a linear functional $L(Y) = \langle \ell(t_1), Y\rangle - a(X^0, Y)$ we obtain a one-time step incremental problem

$$L(Y - X) \le a(X, Y - X) + \psi(Q) - \psi(P) \quad \text{for all } Y = (V, Q) \in \mathcal{S}.$$

Introducing the energy functional

Lemma (Equivalent Reformulations)

For each $X = (U, P) \in \mathcal{S}$ the following three conditions (a)-(c) are equivalent:

$$(a) \quad L(Y - X) \leq a(X, Y - X) + \psi(Q) - \psi(P) \quad \text{for all } Y = (V, Q) \in \mathcal{S}.$$

$$(b) \quad L(Y - X) = a(X, Y - X) \quad \text{for all } Y = (V, P) \in \mathcal{S} \quad \text{and}$$
$$L(Y - X) \leq a(X, Y - X) + \psi(Q) - \psi(P) \quad \text{for all } Y = (U, Q) \in \mathcal{S}.$$

$$(c) \quad \Phi(X) = \min_{Y \in S} \Phi(Y) \quad \text{with } \Phi(Y) = \frac{1}{2} a(Y, Y) + \psi(Q) - L(Y).$$

Abreviations

The following matrix notation allows for a brief formulation of the discrete problem. Let

$$P := \begin{pmatrix} P_1 \\ \vdots \\ P_M \end{pmatrix}, P^0 := \begin{pmatrix} P_1^0 \\ \vdots \\ P_M^0 \end{pmatrix}, Q := \begin{pmatrix} Q_1 \\ \vdots \\ Q_M \end{pmatrix}, \hat{\Sigma} := \begin{pmatrix} \mathbb{C}\varepsilon(U) \\ \vdots \\ \mathbb{C}\varepsilon(U) \end{pmatrix},$$

$$\hat{\Sigma}^0 := \begin{pmatrix} \mathbb{C}\varepsilon(U^0) \\ \vdots \\ \mathbb{C}\varepsilon(U^0) \end{pmatrix}, \hat{\mathbb{C}} := \begin{pmatrix} \mathbb{C} & \dots & \mathbb{C} \\ \vdots & & \vdots \\ \mathbb{C} & \dots & \mathbb{C} \end{pmatrix}, \quad \hat{\mathcal{H}} := \begin{pmatrix} \mathcal{H}_1 & \dots & 0 \\ \vdots & & \vdots \\ 0 & \dots & \mathcal{H}_M \end{pmatrix}.$$

Abreviations

Then there holds

$$\begin{aligned} -a(X, Y-X) &= \int_\Omega \left(\hat{\Sigma} - (\hat{\mathbb{C}} + \hat{\mathcal{H}})P\right) : (Q-P)\,\mathrm{d}x, \\ L(Y-X) &= \int_\Omega \left(\hat{\Sigma}^0 - (\hat{\mathbb{C}} + \hat{\mathcal{H}})P^0\right) : (Q-P)\,\mathrm{d}x, \\ \psi(Y) &= \int_\Omega |Q|_{\sigma^y}\,\mathrm{d}x. \end{aligned}$$

Since the plastic yield parameters $\sigma_1^y, \dots, \sigma_M^y$ are positive, the expansion

$$|(Q_1, \dots, Q_M)^T|_{\sigma^y} := \sigma_1^y|Q_1| + \cdots + \sigma_M^y|Q_M|$$

defines a norm in $\mathbb{R}^{Md\times d}$, where $|\cdot|$ denotes the Frobenius norm.

Coupled problem

Problem (Discrete problem)

Given $(U^0, P^0) \in \mathcal{S}$, seek $U^1 \in \mathcal{S}_D^1(\mathcal{T})$ such that for all $V \in \mathcal{S}_D^1(\mathcal{T})$,

$$\int_\Omega \mathbb{C}(\varepsilon(U^1) - \sum_{r=1}^M P_r^1) : \varepsilon(V)\,dx - \int_\Omega f(t)V\,dx - \int_{\Gamma_N} gV\,dx = 0. \quad (14)$$

Here $P = (P_1, \dots, P_M)^T = (P_1^1, \dots, P_M^1)^T - (P_1^0, \dots, P_M^0)^T$ satisfies

$$(\hat{A} - (\hat{\mathbb{C}} + \hat{\mathcal{H}})P) : (Q-P) \le |Q|_{\sigma^y} - |P|_{\sigma^y} \quad (15)$$

for all $Q = (Q_1, \dots, Q_M)^T$ with $Q_1, \dots, Q_M \in \mathrm{dev}(\mathcal{S}^0(\mathcal{T})^{d\times d}_{sym})$ and

$$\hat{A} := \hat{\Sigma}(U^1) + \hat{\Sigma}^0(U^0) - (\hat{\mathbb{C}} + \hat{\mathcal{H}})P^0.$$

Moreau regularization

Theorem (Moreau, 1965)

Let the function $\mathcal{F} : \mathcal{H} \times \mathcal{H} \to \overline{\mathbb{R}}$ *be defined*

$$\mathcal{F}(x, y) = \frac{1}{2} \|y - x\|^2_{\mathcal{H}} + \psi(x) \tag{16}$$

where ψ *is a convex, proper and lower semi continuous mapping of* $\mathcal{H}$ *into* $\overline{\mathbb{R}}$. *Then*

$$F(y) := \inf_{x \in \mathcal{H}} \mathcal{F}(x, y)$$

is well defined as a functional from $\mathcal{H}$ *into* $\mathbb{R}$ *and there exists a unique mapping* $\tilde{x} : \mathcal{H} \to \mathcal{H}$ *such, that*

$$F(y) = \mathcal{F}(\tilde{x}(y), y)$$

holds for all $y \in \mathcal{H}$. *Moreover,* F *is strictly convex and Fréchet differentiable with the derivative*

$$\mathcal{D}F(y) = \langle y - \tilde{x}(y) \, , \, \cdot \rangle_{\mathcal{H}} \in \mathcal{H}^* \qquad \forall \, y \in \mathcal{H} \, . \tag{17}$$

Moreau regularization

Theorem of Moreau implies for elastoplasticity

Theorem

There is a unique function

$$P = P(\varepsilon(U))$$

and the energy functional

$$\Phi(U) = \frac{1}{2} a(U, P(\varepsilon(U)); U, P(\varepsilon(U))) + \psi(P(\varepsilon(U))) - L(U)$$

is strictly convex and differentiable!

more details in

- Peter Gruber, Jan Valdman, Solution of one-time-step problems in elastoplasticity by a Slant Newton Method. SIAM J. Scientific Computing 31, No. 2, 1558-1580 (2009)

Analysis of single-yield model (M=1)

Localization to one element $T \in \mathcal{T}$:
One plastic strain

$$P \in \mathbb{R}^{2\times 2}_{\text{sym}}, \quad \operatorname{tr} P = 0,$$

the elastic matrix $\mathbb{C}$ with the (positive) Lamé coefficients μ and λ

$$\mathbb{C}P = 2\mu P + \lambda(\operatorname{tr}\mathcal{P})\mathbb{I} = 2\mu P,$$

the hardening matrix $\mathcal{H}$ with

$$\mathcal{H}P = hP,$$

the matrix norm

$$|P|_{\sigma^y} = \sigma^y |P|$$

and the matrix

$$A := \hat{A} := \mathbb{C}\varepsilon(U) + \mathbb{C}\varepsilon(U^0) - (\mathbb{C} + \mathcal{H})P^0.$$

Analysis of single-yield model (M=1)

Lemma (Alberty, Carstensen, Zarrabi, 1999)

Given $A \in \mathbb{R}^{d\times d}_{\text{sym}}$ and $\sigma^y > 0$. There exists exactly one $P \in \operatorname{dev}\mathbb{R}^{d\times d}_{\text{sym}}$ that satisfies

$$\{A - (\mathbb{C} + \mathcal{H})P\} : (Q - P) \leq \sigma^y\{|Q| - |P|\}$$

for all $Q \in \operatorname{dev}\mathbb{R}^{d\times d}_{\text{sym}}$. This P is characterized as the minimiser of

$$\frac{1}{2}(\mathbb{C} + \mathcal{H})Q : Q - Q : A + \sigma^y|Q| \tag{18}$$

(amongst trace-free symmetric $d \times d$-matrices) and is given by

$$P = \frac{(|\operatorname{dev} A| - \sigma^y)_+}{2\mu + h}\frac{\operatorname{dev} A}{|\operatorname{dev} A|}, \tag{19}$$

where $(\cdot)_+ := \max\{0, \cdot\}$ denotes the non-negative part.

Analysis of two-yield model (M=2)

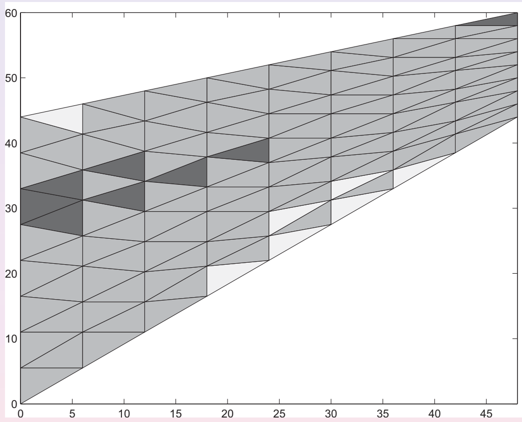


Figure: Cook's membrane problem in the first time step. The black colour shows elastic upgrade zones (where $P_1 = P_2 = 0$), brown and lighter gray colours shows the first plastic upgrade ($P_1 \neq 0, P_2 = 0$) and the both plastic upgrades ($P_1 \neq 0, P_2 \neq 0$) zones.

Analysis of two-yield model (M=2)

Two plastic strains P_1, P_2 coupled in a generalized plastic strain

$$P = (P_1, P_2)^T.$$

The generalized elasticity matrix and the generalized hardening matrices read

$$\hat{\mathbb{C}} := \begin{pmatrix} \mathbb{C} & \mathbb{C} \\ \mathbb{C} & \mathbb{C} \end{pmatrix} \quad \text{and} \quad \hat{\mathcal{H}} := \begin{pmatrix} \mathcal{H}_1 & 0 \\ 0 & \mathcal{H}_2 \end{pmatrix},$$

the generalized loading matrix reads

$$\hat{A} = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} = \begin{pmatrix} \mathbb{C}\varepsilon(U) \\ \mathbb{C}\varepsilon(U) \end{pmatrix} + \begin{pmatrix} \mathbb{C}\varepsilon(U^0) \\ \mathbb{C}\varepsilon(U^0) \end{pmatrix} - \begin{pmatrix} \mathbb{C} + \mathcal{H}_1 & \mathbb{C} \\ \mathbb{C} & \mathbb{C} + \mathcal{H}_2 \end{pmatrix} \begin{pmatrix} P_1^0 \\ P_2^0 \end{pmatrix}$$

and the matrix norm is defined by

$$|P|_{\sigma^y} = \sigma_1^y |P_1| + \sigma_2^y |P_2|.$$

Analysis of two-yield model (M=2)

Lemma

Given $\hat{A} = (A_1, A_2)^T, A_1, A_2 \in \mathbb{R}^{d\times d}_{\text{sym}}$, *there exists exactly one* $P = (P_1, P_2)^T, P_1, P_2 \in \operatorname{dev}\mathbb{R}^{d\times d}_{\text{sym}}$ *that satisfies*

$$(\hat{A} - (\hat{\mathbb{C}} + \hat{\mathcal{H}})P) : (Q - P) \le |Q|_{\sigma^y} - |P|_{\sigma^y} \tag{20}$$

for all $Q = (Q_1, Q_2)^T, Q_1, Q_2 \in \operatorname{dev}\mathbb{R}^{d\times d}_{\text{sym}}$. *This* P *is characterized as the minimiser of*

$$f(Q) = \frac{1}{2}(\hat{\mathbb{C}} + \hat{\mathcal{H}})Q : Q - Q : \hat{A} + |Q|_{\sigma^y} \tag{21}$$

(amongst trace-free symmetric $d \times d$ *matrices* Q_1, Q_2*).*

Exact minimizer?

Analysis of two-yield model (M=2)

We introduce the operator

$$\mathcal{F}(M, \sigma, h) := \frac{(|M| - \sigma)_+}{2\mu + h}\frac{M}{|M|}. \tag{22}$$

Algorithm (Iterative calculation of P_1, P_2)

Input μ, h_1, h_2, σ_1^y, σ_2^y, $\operatorname{dev} A_1$, $\operatorname{dev} A_2$ *and* $\text{tol} \ge 0$.

1. *Set* $i := 0$ *and set the initial approximation* $P_1^i = P_2^i = 0$.
2. *Update* P_2^i *via* $P_2^{i+1} = \mathcal{F}(\operatorname{dev} A_2 - 2\mu P_1^i, \sigma_2^y, h_2)$.
3. *Update* P_1^i *via* $P_1^{i+1} = \mathcal{F}(\operatorname{dev} A_1 - 2\mu P_2^{i+1}, \sigma_1^y, h_1)$.
4. *If the desired accuracy is reached, i. e., if*

   $$|P_1^{i+1} - P_1^i| + |P_2^{i+1} - P_2^i| \le \text{tol}\cdot(|P_1^{i+1}| + |P_1^i| + |P_2^{i+1}| + |P_2^i|)$$

   then output solution $(P_1, P_2) = (P_1^{i+1}, P_2^{i+1})$. *Otherwise, set* $i := i + 1$ *and go to step 2.*

Analysis of two-yield model (M=2)

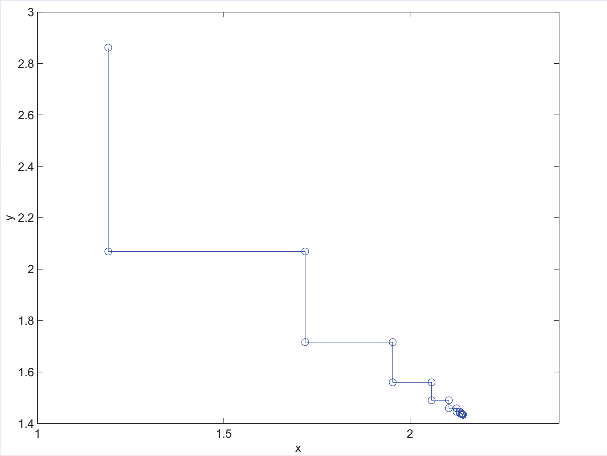


Figure: The approximations $P_1^i = (x^i, 0; 0, -x^i)$, $P_2^i = (y^i, 0; 0, -y^i)$, $i = 0, \ldots, 34$ computed by the iterative algorithm and displayed as the points (x^i, y^i) in the $x - y$ coordinate system.

Newton method

A nonlinear system of equations for $2N$ displacement unknowns $\mathbf{U^1} = (U_1^1, \ldots, U_{2N}^1)^T$:

$$\mathbf{F}_i(\mathbf{U^1}) = 0 \quad \text{for all} \quad i = 1, \ldots, 2N. \tag{23}$$

We use the Newton-Raphson method for the iterative solution of (23).

Algorithm (Newton-Raphson Method)

(a) Choose an initial approximation $\mathbf{U_0^1} \in \mathbb{R}^{\mathbf{2N}}$, *set* $k := 0$.
(b) Let $k := k + 1$, *solve* $\mathbf{U_k^1}$ *from*

$$D\mathbf{F}(\mathbf{U_{k-1}^1})(\mathbf{U_k^1} - \mathbf{U_{k-1}^1}) = -\mathbf{F}(\mathbf{U_{k-1}^1}).$$

(c) If $\mathbf{U_k^1} - \mathbf{U_{k-1}^1}$ *is sufficiently small then output* $\mathbf{U_k^1}$, *otherwise goto (b).*

Newton method

In order to incorporate the Dirichlet boundary conditions properly, the linear system in the step (b) is extended,

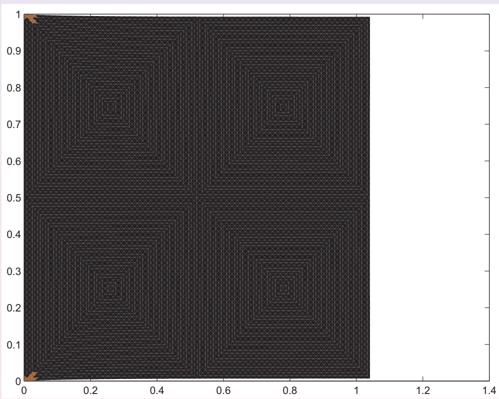
$$\begin{pmatrix} D\mathbf{F}(\mathbf{U^1_{k-1}}) & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} \mathbf{U^1_k} - \mathbf{U^1_{k-1}} \\ \lambda \end{pmatrix} = \begin{pmatrix} -\mathbf{F}(\mathbf{U^1_{k-1}}) \\ \mathbf{0} \end{pmatrix},$$

with some matrix B and the vector of Lagrange parameters λ. Here, $D\mathbf{F}(\mathbf{U^1_k}) \in \mathbb{R}^{2N\times 2N}$ represents a sparse tangential stiffness matrix

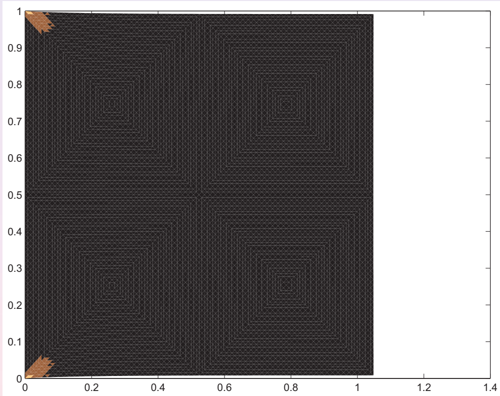
$$D\mathbf{F}(\mathbf{U})_{ij} \approx \frac{\mathbf{F}(U_1,\dots,U_j+\epsilon_j,\dots,U_{2N})_i - \mathbf{F}(U_1,\dots,U_j-\epsilon_j,\dots,U_{2N})_i}{2\epsilon_j}$$

approximated by a central difference scheme with small parameters $\epsilon_j > 0$, $j = 1,\dots,2N$.

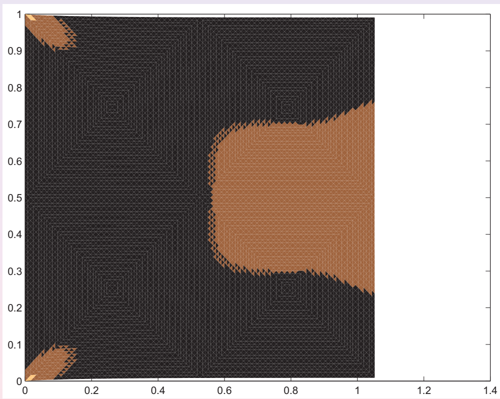
Matlab simulations: two-yield 2D beam model



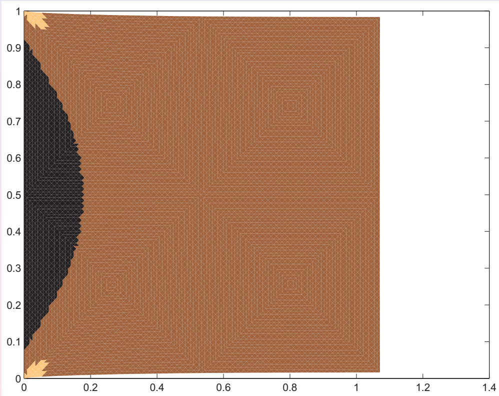
Matlab simulations: two-yield 2D beam model



Matlab simulations: two-yield 2D beam model

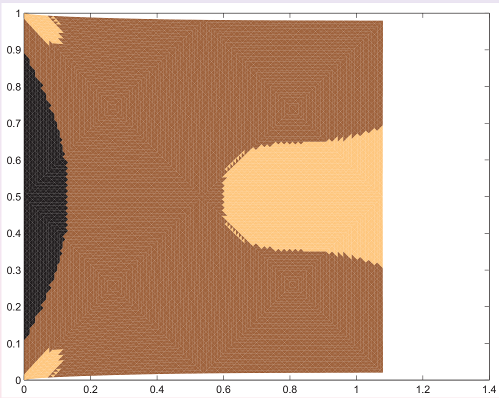


Matlab simulations: two-yield 2D beam model



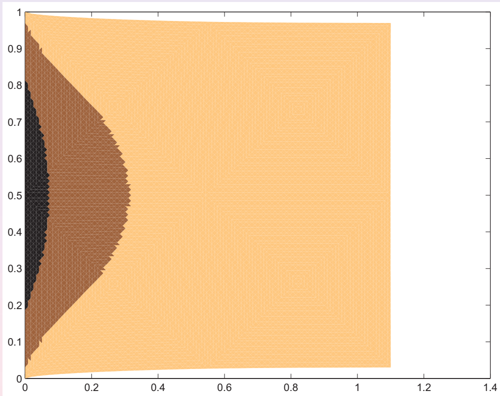
Rheological models Variational inequalities Existence **FE Discretization**

Matlab simulations: two-yield 2D beam model

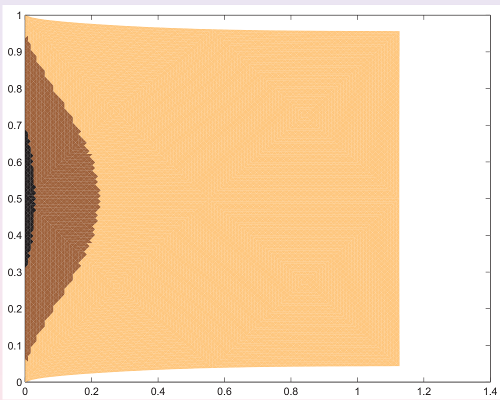


Jan Valdman Computational elastoplasticity

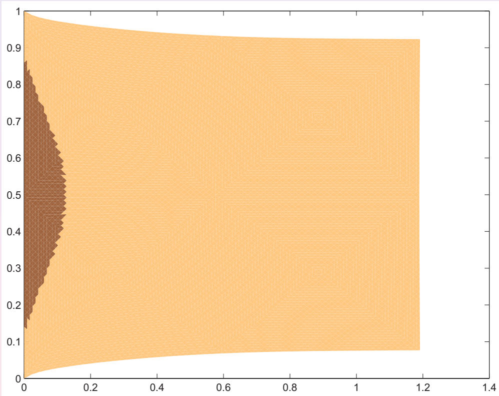
Matlab simulations: two-yield 2D beam model



Matlab simulations: two-yield 2D beam model



Matlab simulations: two-yield 2D beam model



Matlab simulations: single-yield model

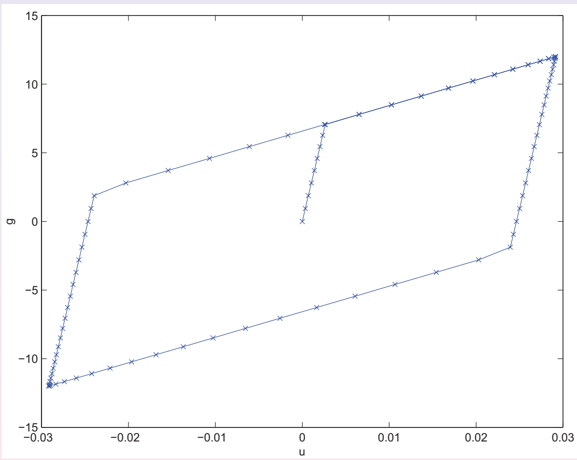


Figure: Displayed loading-deformation relation in terms of the uniform surface loading $g_x(t)$ versus the x-displacement of the point $(0, 1)$ for problem of the single-yield beam with 1D effects.

Matlab simulations: two-yield model

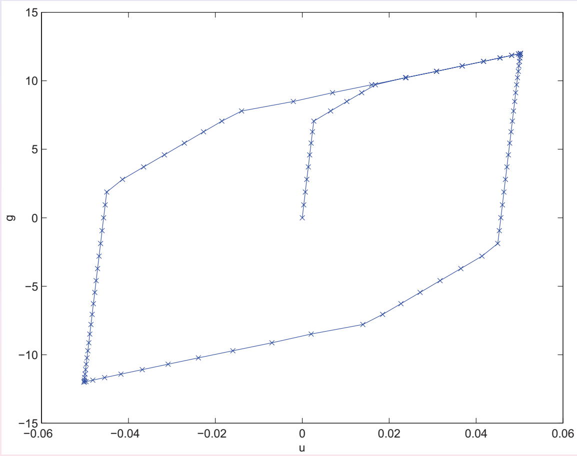


Figure: Displayed loading-deformation relation in terms of the uniform surface loading $g_x(t)$ versus the x-displacement of the point $(0, 1)$ for problem of the two-yield beam with 1D effects.

Papers on Matlab Implementation

- Jochen Alberty, Carsten Carstensen and Stefan A. Funken, Remarks around 50 lines of Matlab: short finite element implementation, Numerical Algorithms 20 (117), 117–137 (1999)
- Alberty, Carstensen, Funken, Klose, Matlab implementation of the finite element method in elasticity, Computing 69 (3), 239 – 263 (2002)
- Carstensen C., Klose R., Elastoviscoplastic Finite Element Analysis in 100 lines of Matlab, J. Numer. Math., 10 (3), 157–192 (2002)
- Rahman T., Valdman J., Fast MATLAB assembly of FEM stiffness- and mass matrices in 2D and 3D: nodal elements, Proceedings of conference PARA 2010 (submitted)

Computer exercises

- computation of triagulation areas
- uniform refinement in 2D
- generation of a stifness matrix
- generation of a right-hand side
- a posteriori computation of a plasticity strain from a given stress
- alternating directions iteration over equilibrium and plasticity inequality
- extension to time-dependent problems

Thank you for your attention!

COMPUTATIONAL MECHANICS I I

D. Lukáš (VŠB-TU) **Parallel BEM-based methods**
J. Valdman (U Iceland) **Aposteriori error estimates (3)**
H. Palsson (U Iceland) **Computational fluid dynamics with OpenFOAM(2)**
O. Axelsson (I Geonics) **Preconditioning for saddle point problems**
M. Neytcheva (U Uppsala) **Augmented Lagrangian preconditioning**
J. Kruis (CTU Prague) **Damage mechanics analysis with arc-length solver**
Z. Dostál (VŠB-TU) **T-FETI based scalable algorithms for contact problems**
O. Vlach (VŠB-TU) **Quasistatic contact problems**
T. Kozubek (VŠB-TU) **Scalable algorithms for dynamics contact problems**
V. Vondrák (VŠB-TU) **Efficient parallel contact shape optimization**
R. Blaheta et al. (I Geonics) **Computational (geo) micromechanics**
R. Blaheta et al. (I Geonics) **Identification of material parameters**
S. Sysala (I Geonics) **Elasto-plasticity: selected topics**

Parallel BEM–Based Methods

Workshop on Computational Mechanics II, ÚGN Ostrava, Dec. 1, 2010

D. Lukáš

Department of Applied Mathematics
VŠB–Technical University of Ostrava, Czech Rep.

email: dalibor.lukas@vsb.cz

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

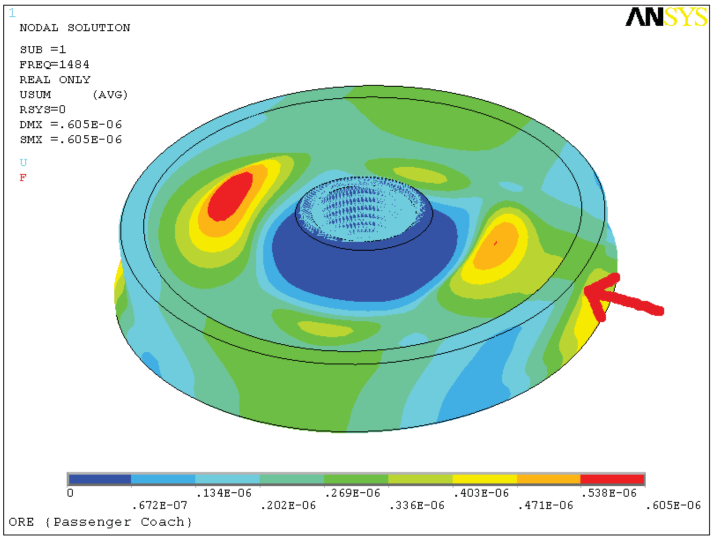
Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

Motivation: Acoustic scattering from a railway wheel

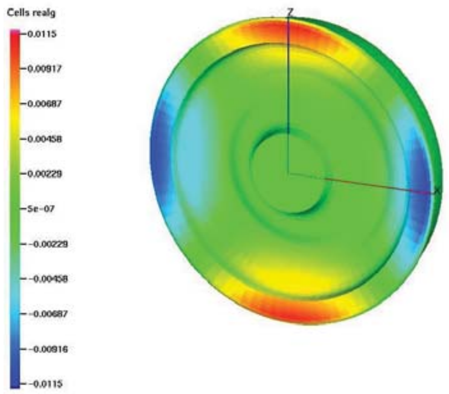
The goal: Acoustic noise elimination by profiling

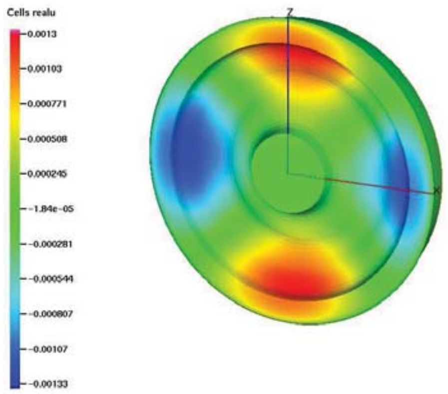


a joint work with Jan Szweda, Dep. of Mechanics, VŠB-TU Ostrava

Motivation: Acoustic scattering from a railway wheel

Numerical simulation on a single proc.: velocity, pressure at 341 Hz

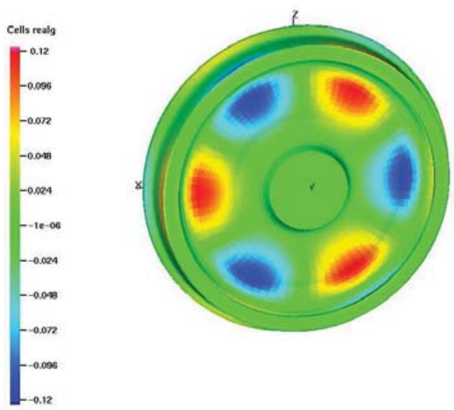


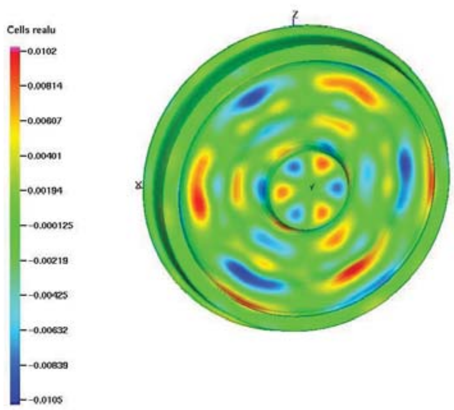


15112 triangles, 22668 nodes,
ACA-E assembling of $\mathbf{K}_\kappa$ (compr. to 12%) in 25 min, of $\mathbf{D}_\kappa$ (15%) in 40 min,
142 GMRES iters. in 223 s

Motivation: Acoustic scattering from a railway wheel

Numerical simulation on a single proc.: velocity, pressure at 2706 Hz





15112 triangles, 22668 nodes,
ACA-E assembling of $\mathbf{K}_\kappa$ (compr. to 12%) in 25 min, of $\mathbf{D}_\kappa$ (15%) in 50 min,
700 GMRES iters.

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

Boundary element method

Interior Dirichlet Laplace problem

$$\begin{aligned} -\triangle u(\mathbf{x}) &= 0, \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^3, \\ u(\mathbf{x}) &= g(\mathbf{x}), \ \ \mathbf{x} \in \Gamma := \partial\Omega. \end{aligned}$$

Boundary integral formulation

$\mathbf{x} \in \Omega : \quad u(\mathbf{x}) := \int_\Gamma w(\mathbf{y}) \frac{1}{4\pi|\mathbf{x}-\mathbf{y}|} dS(\mathbf{y}) =: (Vw)(\mathbf{x})$
Under some regularity assumptions, $(Vw)(\mathbf{x})$ is continuous along Γ, which leads us to the (less understood) collocation method: Find $w(\mathbf{y})$ such that

$$(Vw)(\mathbf{x}) = g(\mathbf{x}) \text{ on } \Gamma,$$

or to the well–posed Galerkin boundary integral method:

$$\text{Find } w(\mathbf{y}) \in W : \quad \langle Vw, z\rangle_\Gamma = \langle g, z\rangle_\Gamma \quad \forall z \in W,$$

where $\langle f, z\rangle_\Gamma := \int_\Gamma f(\mathbf{x}) z(\mathbf{x})\, dS(\mathbf{x})$ and $W := H^{-1/2}(\Gamma)$.

Boundary element method

Boundary element method (BEM)

Triangulate the boundary $\Gamma = \cup_{j=1}^{n} \overline{\gamma_j}$ and
approximate $H^{-1/2}(\Gamma)$ by piecewise constant base $\Psi_j(\mathbf{x})$ along the triangulation.
Find $w_h(\mathbf{x}) := \sum_{j=1}^{n} w_j \Psi_k(\mathbf{x})$:

$$\mathbf{A}\mathbf{w} = \mathbf{b},$$

where $a_{ij} := \int_{\gamma_i} \int_{\gamma_j} \frac{1}{4\pi|\mathbf{x}-\mathbf{y}|} \, dS(\mathbf{y}) \, dS(\mathbf{x})$, $b_i := \int_{\gamma_i} g(\mathbf{x}) \, dS(\mathbf{x})$. The approximate solution then reads

$$\mathbf{x} \in \overline{\Omega} : \; u_h(\mathbf{x}) = \sum_{j=1}^{n} w_j \int_{\gamma_j} \frac{1}{4\pi|\mathbf{x}-\mathbf{y}|} \, dS(\mathbf{y}).$$

Comparison to FEM

+ exterior problems: radiation conditions in the ansatz,
+ problem dimension reduced,

− expensive evaluation of singular integrals

− densely populated matrices.

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

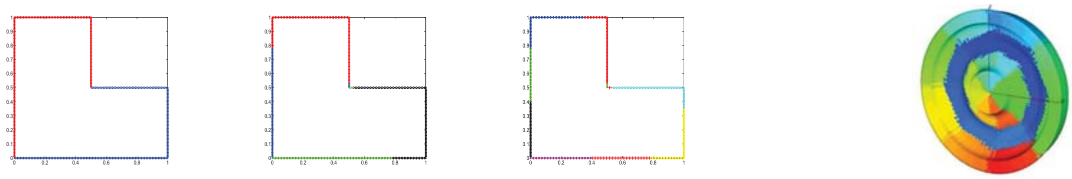
Acceleration by adaptive cross approximation

Cluster geometric bisection

$C := \{\gamma_1^C, \dots, \gamma_{n^C}^C\}$ ... cluster of elements from discretization $\{\gamma_1, \dots, \gamma_n\}$ of Γ,
$\mathbf{x}^C := \frac{1}{\sum_k |\gamma_k^C|} \sum_k |\gamma_k^C| \, \mathbf{x}_k^C$ ... cluster centroid, where $\mathbf{x}_k^C$ is the centroid of γ_k^C,
$\mathbf{C}^C := \sum_k |\gamma_k^C| \, (\mathbf{x}_k^C - \mathbf{x}^C) \cdot (\mathbf{x}_k^C - \mathbf{x}^C)^T$ ... cluster covariance matrix,
$\mathbf{n}^C$ ... a dominant eigenvector of $\mathbf{C}^C$.

The cluster is cutted into two subclusters by the plane $(\mathbf{x} - \mathbf{x}^C) \cdot \mathbf{n}^C = 0$ as follows:
$C_1 := \left\{ \gamma_k \in C \ : \ (\mathbf{x}_k^C - \mathbf{x}^C) \cdot \mathbf{n}^C \geq 0 \right\}$ ... first subcluster,
$C_2 := \left\{ \gamma_k \in C \ : \ (\mathbf{x}_k^C - \mathbf{x}^C) \cdot \mathbf{n}^C < 0 \right\}$ ... second subcluster.

METIS could be an alternative.



Acceleration by adaptive cross approximation

Admissible pairs of clusters (quadratic complexity)

$$\min\{\operatorname{diam} C_x, \operatorname{diam} C_y\} \leq \eta \operatorname{dist}(C_x, C_y), \quad \eta \in (0,1)$$

Stronger admissibility criterion (linear complexity)

$$\begin{aligned} &\min\{\operatorname{diam} C_x, \operatorname{diam} C_y\} \leq \\ &\qquad 2 \min\{\operatorname{rad} C_x, \operatorname{rad} C_y) \leq \eta \left(|\mathbf{x}^{C_x} - \mathbf{x}^{C_y}| - \operatorname{rad} C_x - \operatorname{rad} C_y \right) \\ &\qquad\qquad \leq \eta \operatorname{dist}(C_x, C_y), \end{aligned}$$

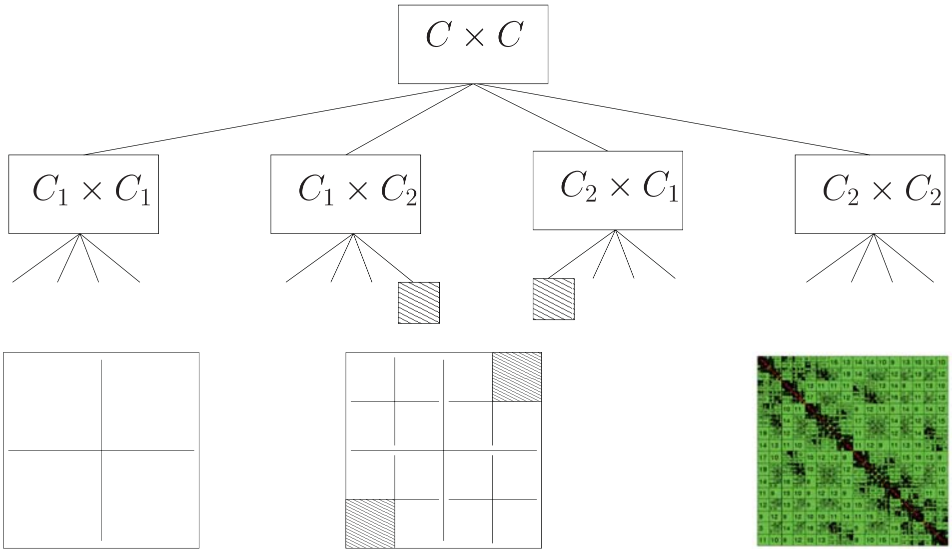
where $\operatorname{rad} C := \max_k |\mathbf{x}_k^C - \mathbf{x}^C|$.

Quad–tree of cluster pairs

$(\{\gamma_1, \dots, \gamma_m\}, \{\gamma_1, \dots, \gamma_m\})$ is the root.
Leaves (C, D) are either admissible or $\min\{n^C, n^D\} \leq n_{\min}$.
Nonleaves (C, D) has four sons (C_1, D_1), (C_1, D_2), (C_2, D_1), and (C_2, D_2).

Acceleration by adaptive cross approximation

Quad–tree of cluster pairs, $\mathcal{H}$–matrices



Nonadmissible blocks assembled as full, admissible approximated by low–rank matrices.

Acceleration by adaptive cross approximation

Compression by singular value decomposition (SVD)

$$\mathbf{A} = \sum_{i=1}^{r:=\operatorname{rank}\mathbf{A}} \sigma_i\,\mathbf{u}_i\,\mathbf{v}_i^T \approx \sum_{i=1}^{k} \sigma_i\,\mathbf{u}_i\,\mathbf{v}_i^T =: \mathbf{A}_k, \ \text{where } k < r,$$

$\sigma_1 \geq \sigma_2 \geq \ldots \sigma_r \geq 0$ ...singular values,
$(\mathbf{u}_1, \ldots, \mathbf{u}_r)$ ...an orthogonal system of left singular vectors,
$(\mathbf{v}_1, \ldots, \mathbf{v}_r)$ ...an orthogonal system of right singular vectors.
SVD gives the best approximation in the spectral (operator) norm:

$$\mathbf{A}_k = \arg \min_{\mathbf{M}:\operatorname{rank}\mathbf{M}=k} \|\mathbf{A} - \mathbf{M}\|.$$

The best compression, but worse than quadratic computational complexity $O(m\,n\,r)$.

Acceleration by adaptive cross approximation

Asymptotically smooth functions

Assume $(\mathbf{A})_{i,j} := f(\mathbf{x}_i, \mathbf{y}_j)$, where $\mathbf{x}_i \in C_x$, $\mathbf{y}_j \in C_y$, $C_x, C_y \subset \mathbb{R}^d$.
$f : C_x \times C_y \to \mathbb{R}$ is asymptotically smooth if

$$\exists c_1, c_2 > 0 \; \exists g \leq 0 \; \forall \alpha \in \mathbb{N}_0^d : \; |\partial_{\mathbf{x}}^\alpha f(\mathbf{x}, \mathbf{y})|, \left|\partial_{\mathbf{y}}^\alpha f(\mathbf{x}, \mathbf{y})\right| \leq c_1 p! (c_2)^p |\mathbf{x}-\mathbf{y}|^{g-p}, \quad p = |\alpha|.$$

Compression by Taylor expansion

Provided $\operatorname{diam} C_y \leq \operatorname{diam} C_x$, $dc_2\eta < 1$, choose $\mathbf{y}_0 \in C_y$ about which we expand f:

$$\mathbf{x} \in C_x, \; \mathbf{y} \in C_y \; : \; f(\mathbf{x}, \mathbf{y}) = \sum_{k=0}^{p-1} \frac{1}{k!} \left((\mathbf{y} - \mathbf{y}_0)\partial_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}_0)\right)^k + R_p(\mathbf{x}, \mathbf{y}),$$

where

$$\begin{aligned} |R_p(\mathbf{x}, \mathbf{y})| \; &= \; \frac{1}{p!} \left|(\mathbf{y} - \mathbf{y}_0)\partial_{\mathbf{y}} f(\mathbf{x}, \widetilde{\mathbf{y}})\right|^p \leq \frac{1}{p!} d^p |\mathbf{y} - \mathbf{y}_0|^p c_1 p! (c_2)^p |\mathbf{x} - \widetilde{\mathbf{y}}|^{g-p} \\ &\leq \; c_1 d^p c_2^p \frac{\operatorname{diam}^p C_y}{\operatorname{dist}^p(C_x, C_y)} \operatorname{dist}^g(C_x, C_y) \leq c_1 (dc_2\eta)^p \operatorname{dist}^g(C_x, C_y) \to 0 \text{ as } p \to \infty. \end{aligned}$$

Acceleration by adaptive cross approximation

Adaptive cross approximation (ACA)

$$\mathbf{P}_{C_x} \mathbf{A} \, \mathbf{P}_{C_y}^T =: \begin{pmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{22} \end{pmatrix} \approx \begin{pmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{21} \widetilde{\mathbf{A}}_{11}^{-1} \widetilde{\mathbf{A}}_{12} \end{pmatrix} = \begin{pmatrix} \widetilde{\mathbf{A}}_{11} \\ \widetilde{\mathbf{A}}_{21} \end{pmatrix} \left[\widetilde{\mathbf{A}}_{11}^{-1} \left(\widetilde{\mathbf{A}}_{11}, \widetilde{\mathbf{A}}_{12} \right) \right]$$
$$=: \; (\mathbf{u}_1, \ldots, \mathbf{u}_r) \, (\mathbf{v}_1, \ldots, \mathbf{v}_r)^T.$$

The rank $r := r(\varepsilon)$, where $\widetilde{\mathbf{A}}_{11} \in \mathbb{C}^{r \times r}$, is adaptively controlled by ε as follows:

$$\|\mathbf{u}_{k+1}\|_2 \, \|\mathbf{v}_{k+1}\|_2 \leq \frac{\varepsilon(1-\eta)}{1+\varepsilon} \|\mathbf{A}_k\|_F, \quad \text{where } \mathbf{A}_k := \sum_{m=1}^{k} \mathbf{u}_m \mathbf{v}_m^T$$

which implies, provided $\|\mathbf{R}_{k+1}\|_F \leq \eta \|\mathbf{R}_k\|_F$, that $\frac{\|\mathbf{R}_k\|_F}{\|\mathbf{A}\|_F} \leq \varepsilon$, where $\mathbf{R}_k := \mathbf{A} - \mathbf{A}_k$.
The pivots, stored in $\mathbf{P}_{C_x}$, $\mathbf{P}_{C_y}$, are chosen as to maximize $|\det \widetilde{\mathbf{A}}_{11}^k|$ with a wish to minimize $\|\mathbf{R}_k\| \equiv \|\widetilde{\mathbf{A}}_{22}^k - \widetilde{\mathbf{A}}_{21}^k \, (\widetilde{\mathbf{A}}_{11}^k)^{-1} \, \widetilde{\mathbf{A}}_{12}^k\|$.

Acceleration by adaptive cross approximation

ACA algorithm: an example ($\mathbf{R}_0 := \mathbf{A}$)

$$\mathbf{R}_0 = \begin{pmatrix} 0.431 & 0.354 & 0.582 & 0.417 \\ 0.491 & 0.396 & 0.674 & 0.449 \\ 0.446 & 0.358 & 0.583 & 0.413 \\ 0.380 & 0.328 & 0.557 & 0.372 \end{pmatrix} \xrightarrow[R=\{1\}]{i_1=1,\, j_1=3} \frac{1}{0.582} \begin{pmatrix} 0.582 \\ 0.674 \\ 0.583 \\ 0.557 \end{pmatrix} (0.431, 0.354, 0.582, 0.417)$$

$$\mathbf{R}_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -0.008 & -0.014 & 0 & -0.034 \\ 0.014 & 0.003 & 0 & -0.005 \\ -0.033 & -0.011 & 0 & -0.027 \end{pmatrix} \xrightarrow[R=\{1,2\}]{i_1=2,\, j_1=4} \frac{1}{-0.034} \begin{pmatrix} 0 \\ -0.034 \\ -0.005 \\ -0.027 \end{pmatrix} (-0.008, -0.014, 0, -0.0$$

$$\mathbf{R}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.015 & 0.005 & 0 & 0 \\ -0.026 & 0.0004 & 0 & 0 \end{pmatrix} \xrightarrow[R=\{1,2,4\}]{i_1=4,\, j_1=1} \frac{1}{-0.026} \begin{pmatrix} 0 \\ 0 \\ 0.015 \\ -0.026 \end{pmatrix} (-0.026, 0.0004, 0, 0)$$

The relative error decays as follows: $\|\mathbf{R}_k\|_2 / \|\mathbf{A}\|_2 = 0.030, 0.016, 0.003$ for $k = 1, 2, 3$

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

Parallel implementation

Master–slave model

N processes, one of which is master, each stores all the nodes and triangles.
Reset sets of adm./nonadm. indices: $\mathcal{A}_p := \emptyset$, $\mathcal{N}_p := \emptyset$ for $p = 1, 2, \ldots, N$.
Master sorts $\mathbf{A}_i^{\text{adm}} \in \mathbb{R}^{m_i^{\text{adm}} \times n_i^{\text{adm}}}$ s.t. $w_i^{\text{adm}} := m_i^{\text{adm}} + n_i^{\text{adm}}$
and distributes the indices to all processes so that $\mathcal{A}_k := \mathcal{A}_k \cup \{i\}$ with

$$k := \text{argmin}_{l=1,\ldots,N} \sum_{j \in \mathcal{A}_l} w_j^{\text{adm}}.$$

Master sorts $\mathbf{A}_i^{\text{non}} \in \mathbb{R}^{m_i^{\text{non}} \times n_i^{\text{non}}}$ w.r.t. weights $w_i^{\text{non}} := m_i^{\text{non}}\, n_i^{\text{non}}$
and distributes the indices to all processes so that $\mathcal{N}_k := \mathcal{N}_k \cup \{i\}$ with

$$k := \text{argmin}_{l=1,\ldots,N} \sum_{j \in \mathcal{N}_l} w_j^{\text{non}}.$$

Processes assemble in parallel all their admissible and nonadmissible blocks.

$$\mathbf{A}\,\mathbf{v} = \sum_{p=1}^{N} \left(\sum_{j \in \mathcal{A}_p} \mathbf{I}_j^{\text{adm}}\, \mathbf{U}_j^{\text{adm}}\, \mathbf{V}_j^{\text{adm}}\, \mathbf{v}_{J_j^{\text{adm}}} + \sum_{j \in \mathcal{N}_p} \mathbf{I}_j^{\text{non}}\, \mathbf{A}_j^{\text{non}}\, \mathbf{v}_{J_j^{\text{non}}} \right)$$

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

Numerical results

$u(\mathbf{x}) := g(\mathbf{x}) := x_1 + x_2 + x_3$, $\Omega := \{\mathbf{x} \ : \ |\mathbf{x}| \le 1\}$

n	err.	compr. of $\mathbf{A}$	scheduling+assembling times of $\mathbf{A}$ [s] $N := 2$	$N := 4$	$N := 8$	$N := 16$	$N := 32$	$N := 46$
40	3e-4	100%	0+0	0+0	0+0	0+0	0+0	0+0
160	2.3e-4	100%	0+0	0+0	0+0	0+0	0+0	0+0
640	9.5e-5	99%	0+4	0+2	0+1	0+0	0+1	0+0
2560	4.3e-5	65%	0+43	2+23	0+12	0+6	0+3	0+3
10240	2.1e-5	27%	2+282	1+143	1+72	1+35	1+19	0+13
40960	1.1e-5	10%	46+1572	24+792	15+399	17+201	20+102	21+72
163840	6.4e-6	3%	3041+8219	1457+4162	828+2084	492+1061	409+543	397+377

$$err. := \frac{\sqrt{\langle V(u-u_h), u-u_h\rangle_\Gamma}}{\sqrt{\langle Vu, u\rangle_\Gamma}}$$

Numerically as well as parallel scalable method: $CPU = O\left(\frac{n \log n}{N}\right)$,
but $Mem = O(N\, n \log n)$.

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

General setting

Particular solution approach

$$\begin{aligned} -\triangle u(\mathbf{x}) &= f(\mathbf{x}) \quad , \mathbf{x} \in \Omega, \\ u(\mathbf{x}) &= g(\mathbf{x}) \quad , \mathbf{x} \in \Gamma \end{aligned}$$

is replaced by

$$\begin{aligned} -\triangle u^{\mathrm{H}}(\mathbf{x}) &= 0 \quad & , \mathbf{x} \in \Omega, \\ u^{\mathrm{H}}(\mathbf{x}) &= g(\mathbf{x}) - u^{\mathrm{P}}(\mathbf{x}) \quad & , \mathbf{x} \in \Gamma, \end{aligned}$$

where $-\triangle u^{\mathbf{P}}(\mathbf{x}) = f(\mathbf{x})$, $u(\mathbf{x}) = u^{\mathrm{H}}(\mathbf{x}) + u^{\mathrm{P}}(\mathbf{x})$.

Interface problem: Piecewise homogeneous material $a_i > 0$

$$\begin{aligned} -\mathrm{div}(a_i \nabla u_i(\mathbf{x})) &= f_i(\mathbf{x}) \quad & , \mathbf{x} \in \Omega_i, \\ u_i(\mathbf{x}) &= g(\mathbf{x}) \quad & , \mathbf{x} \in \partial\Omega_i \cap \Gamma, \\ u_i(\mathbf{x}) - u_j(\mathbf{x}) &= 0 \quad & , \mathbf{x} \in \Gamma_{ij} := \partial\Omega_i \cap \partial\Omega_j \neq \emptyset, \\ a_i \frac{\partial u_i(\mathbf{x})}{\partial n} - a_j \frac{\partial u_j(\mathbf{x})}{\partial n} &= 0 \quad & , \mathbf{x} \in \Gamma_{ij} \end{aligned}$$

The transmission conditions are formulated by means of boundary integral equations.

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

DDM by Bramble, Pasciak, and Schatz '86

Nonoverlapping domain decomposition

$$\overline{\Omega} = \bigcup_{i=1}^{N} \overline{\Omega_i}, \quad \Omega_i \cap \Omega_j = \emptyset \text{ for } i \neq j$$

leads to the interface problem

$$\begin{aligned} -\mathrm{div}(a_i \nabla u_i(\mathbf{x})) &= f_i(\mathbf{x}) \text{ in } \Omega_i, & u_i(\mathbf{x}) &= g(\mathbf{x}) \text{ on } \Gamma_i \cap \Gamma, \\ u_i(\mathbf{x}) - u_j(\mathbf{x}) &= 0 \text{ on } \Gamma_{ij}, & a_i \partial_n u_i(\mathbf{x}) - a_j \partial_n u_j(\mathbf{x}) &= 0 \text{ on } \Gamma_{ij}. \end{aligned}$$

Preconditioner BPS I = particular solution + concept of corners

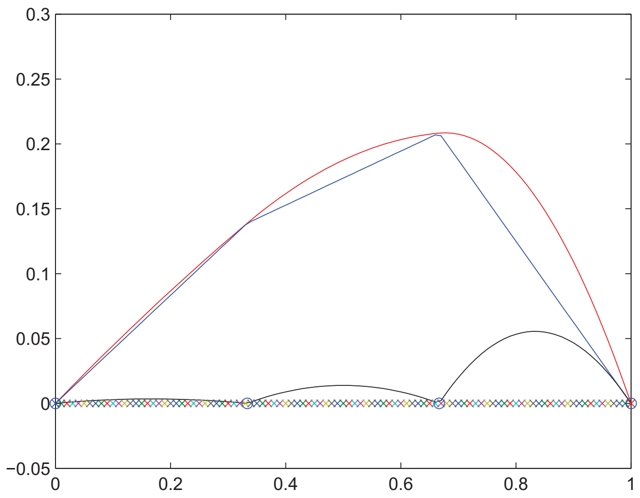
Step 1. Solve in parallel (by Multigrid–FEM): $-\triangle u_i^{\mathrm{P}} = f/a_i$ in Ω_i, $\quad u_i^{\mathrm{P}} = 0$ on Γ_i.
Step 2. Solve interface problem for piecewise harmonic $u^{\mathrm{H}}(\mathbf{x})$ (by parallel ACA–BEM):

$$a_i \partial_n u_i^{\mathrm{H}} - a_j \partial_n u_j^{\mathrm{H}} = -(a_i \partial_n u_i^{\mathrm{P}} - a_j \partial_n u_j^{\mathrm{P}}) \text{ on } \Gamma_{ij}, \quad u^{\mathrm{H}} = 0 \text{ on } \Gamma.$$

Step 3. Solve in parallel (by ACA–BEM): $-\triangle u_i^{\mathrm{H}} = 0$ in Ω_i, $u_i^{\mathrm{H}} = u^{\mathrm{H}}$ on Γ_i.
Preconditioner BPS I introduces corners to decompose the interface $\rightsquigarrow O(\log(H/h))$

DDM by Bramble, Pasciak, and Schatz '86

$u(\mathbf{x}) = u^{\mathrm{H}}(\mathbf{x}) + u^{\mathrm{P}}(\mathbf{x})$



where $\Omega_1 := (0, 1/3)$, $\Omega_2 := (1/3, 2/3)$, $\Omega_3 := (2/3, 1)$, $a_1 := f_3(\mathbf{x}) := 4$, $a_2 := f_2(\mathbf{x}) := 2$, $a_3 := f_1(\mathbf{x}) := 1$.

DDM by Bramble, Pasciak, and Schatz '86

Algebraic point of view of the FEM version

$$\mathbf{A}\mathbf{u} = \mathbf{b}$$

I_i ...interior DOFs of Ω_i, I_s ...DOFs along the skeleton

$$\begin{pmatrix} \mathbf{A}_{I_1,I_1} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{A}_{I_1,I_\mathrm{s}} \\ \mathbf{0} & \mathbf{A}_{I_2,I_2} & \dots & \mathbf{0} & \mathbf{A}_{I_2,I_\mathrm{s}} \\ \vdots & \vdots & \ddots & \vdots & \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{A}_{I_N,I_N} & \mathbf{A}_{I_N,I_\mathrm{s}} \\ \mathbf{A}_{I_\mathrm{s},I_1} & \mathbf{A}_{I_\mathrm{s},I_2} & \dots & \mathbf{A}_{I_\mathrm{s},I_N} & \mathbf{A}_{I_\mathrm{s},I_\mathrm{s}} \end{pmatrix} \begin{pmatrix} \mathbf{u}^\mathrm{H}_{I_1} + \mathbf{u}^\mathrm{P}_{I_1} \\ \mathbf{u}^\mathrm{H}_{I_2} + \mathbf{u}^\mathrm{P}_{I_2} \\ \vdots \\ \mathbf{u}^\mathrm{H}_{I_N} + \mathbf{u}^\mathrm{P}_{I_N} \\ \mathbf{u}^\mathrm{H}_{I_\mathrm{s}} \end{pmatrix} = \begin{pmatrix} \mathbf{b}_{I_1} \\ \mathbf{b}_{I_2} \\ \vdots \\ \mathbf{b}_{I_N} \\ \mathbf{b}_{I_\mathrm{s}} \end{pmatrix}$$

leads to

Step 1. $\mathbf{A}_{I_i,I_i}\mathbf{u}^\mathrm{P}_{I_i} = \mathbf{b}_{I_i}$
Step 2. $\mathbf{S}\mathbf{u}^\mathrm{H}_{I_\mathrm{s}} = \mathbf{c}$ with $\mathbf{S} := \mathbf{A}_{I_\mathrm{s},I_\mathrm{s}} - \sum_{i=1}^N \mathbf{A}_{I_\mathrm{s},I_i}\mathbf{A}^{-1}_{I_i,I_i}\mathbf{A}_{I_i,I_\mathrm{s}}$, $\mathbf{c} := -\sum_{i=1}^N \mathbf{A}_{I_\mathrm{s},I_i}\mathbf{u}^\mathrm{P}_{I_i}$
Step 3. $\mathbf{A}_{I_i,I_i}\mathbf{u}^\mathrm{H}_{I_i} = -\mathbf{A}_{I_i,I_\mathrm{s}}\mathbf{u}^\mathrm{H}_{I_\mathrm{s}}$

Parallel BEM–Based Methods

Outline

- Motivation: Acoustic scattering from a railway wheel
- Parallel fast boundary element method
  - Boundary element method
  - Acceleration by adaptive cross approximation
  - Parallel implementation
  - Numerical results
  - General setting
- DDM by Bramble, Pasciak, and Schatz
- BETI for contact problems

Boundary Element Tearing and Interconnecting (BETI)

Linear case: FETI [Farhat, Roux '91], BETI [Langer, Steinbach '03]

Nonoverlapping DDM with doubled DOFs along the interface, a variant of BPS I.

Step 1. $u^{\mathrm{P}}(\mathbf{x})$ is a FEM/BEM solution to the local Neumann problems:

$$-\triangle u_i^{\mathrm{P}} = f/a_i, \text{ in } \Omega_i, \quad \partial_n u_i = 0 \text{ on } \Gamma_i \quad \rightsquigarrow \quad u^{\mathrm{P}} = K^{+}(f/a_i)$$

Step 2. $u^{\mathrm{H}}(\mathbf{x})$ is represented by Lagrange multipliers $\lambda \in H^{-1/2}(\cup_i \Gamma_i)$:

$$\lambda \equiv \partial_n u^{\mathrm{H}},$$

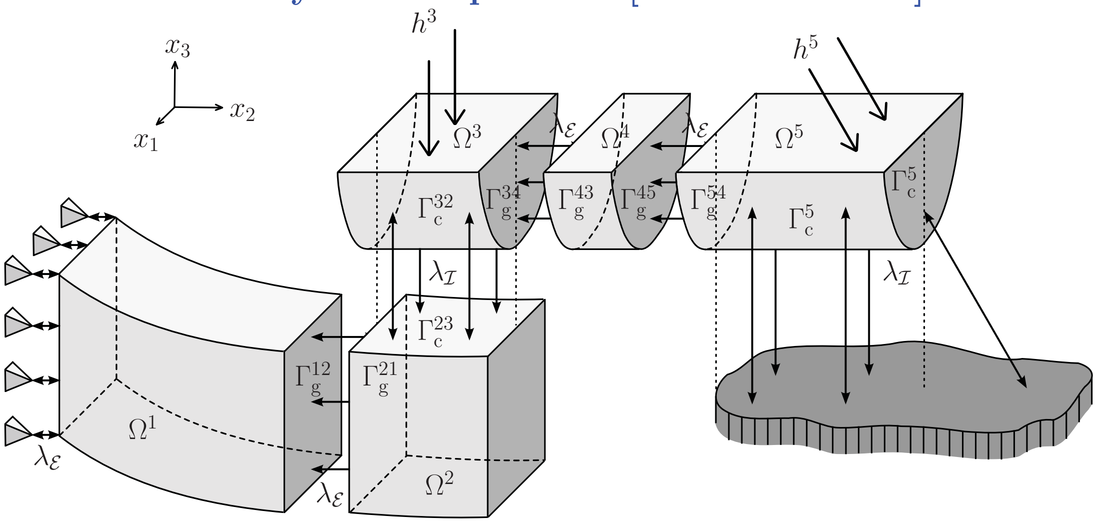
which leads to the Schur complement system with $S := BK^{+}B^{T}$.
Step 3. ...

Dirichlet preconditioner [Mandel, Tezaur '96]

S is preconditioned by $\widehat{S}^{-1} := (1/2)BK_0^{-1}(1/2)B^{T}$,
where K_0^{-1} solves the local Dirichlet problems, i.e. *Step 1* of BPS I.

BETI for contact problems

TBETI for multi–body contact problem [Sadowská et al.]



In the dual the linearized nonpenetration condition translates to a simple bound $\rightsquigarrow$
FETI/BETI ⚭ MPRGP [Dostál, Schöberl '05]

Parallel BEM–Based Methods

References

- Steinbach, Rjasanow, Springer, 2007.
- Bebendorf, Grzhibovskis, Math. Method. Appl. Sci. 29:1721–1747, 2006.
- Bebendorf, Springer, 2008.
- Bramble, Pasciak, Schatz, Math. Comp. 47:103–134, 1986.
- Farhat, Roux, Internat. J. Numer. Meths. Engrg. 32:1205–1227, 1991.
- Mandel, Tezaur, Numer. Math. 73:473–487, 1996.
- Langer, Steinbach, Computing 71:205–228, 2003.
- Dostál, Schöberl, Comput. Optim. Appl. 30:23–44, 2005.
- Sadowská, Bouchala, Kozubek, Markopoulos, Brzobohatý, . . .

Applications of functional a posteriori error estimates to some mechanical problems

Jan Valdman

School of Engineering and Natural Sciences
University of Iceland, Reykjavik
email: jan.valdman@gmail.com

Ostrava, December 1-3, 2010

Outline

Literature on functional a posteriori error estimates

Theory on functional a posteriori error estimates is explained in books:

Pekka Neittaanmäki and Sergey Repin, *Reliable methods for computer simulation, Error control and a posteriori estimates*, Elsevier, New York, 2004.

Sergey Repin, *A Posteriori Estimates for Partial Differential Equations* Radon Series on Computational and Applied Mathematics, de Gruyter, 2008

Linear problems Nonlin. BC Poroelaticity Porous media Elastoplasticity

Explaining papers to theory and numerics to this course:

1. Sergey Repin, Jan Valdman, Functional a posteriori error estimates for problems with nonlinear boundary conditions. Journal of Numerical Mathematics 16, No. 1, 51-81 (2008)
2. Jan Valdman, Minimization of Functional Majorant in A Posteriori Error Analysis based on H(div) Multigrid-Preconditioned CG Method. Advances in Numerical Analysis, vol. 2009, Article ID 164519 (2009)
3. Sergey Repin, Jan Valdman, Functional a posteriori error estimates for incremental models in elasto-plasticity. Cent. Eur. J. Math. 7, No. 3, 506-519 (2009)
4. Jan Martin Nordbotten, Talal Rahman, Sergey Repin, Jan Valdman, A posteriori error estimates for approximate solutions of Barenblatt-Biot poroelastic model. Computational Methods in Applied Mathematics 10, No. 3, 302-315 (2010)
5. P. Neittaanmäki, S. I. Repin and J. Valdman, Functional a posteriori error estimates for elasticity problems with nonlinear boundary conditions. (in preparation)

Jan Valdman Functional a posteriori error estimates

Aposteriori error estimates

Primal problem

$$\triangle u + f = 0 \quad \text{in } \Omega, \qquad u = 0 \quad \text{on } \partial\Omega$$

Let us assume that v is (numerical) approximation of u. Then it holds

Estimate of Runge

$$||\nabla(u-v)||_\Omega \leq ||\nabla v - y^*||_\Omega,$$

for $y^* \in H(\Omega, \mathrm{div})$ satisfying

$$\mathrm{div} y^* + f = 0 \quad \text{in } \Omega.$$

Aposteriori error estimates

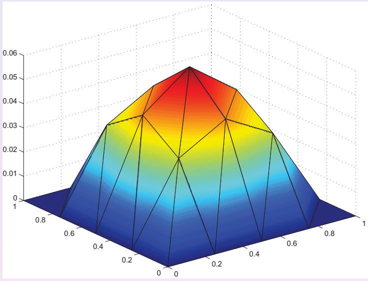
Estimate of Repin

$$||\nabla(u-v)||_\Omega \leq ||\nabla v - y^*||_\Omega + C_\Omega ||\mathrm{div} y^* + f||_\Omega$$

for $y^* \in H(\Omega, \mathrm{div})$. C_Ω is the constant in the Friedrichs' inequality

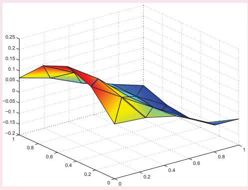
$$\|w\|_\Omega \leq C_\Omega \|\nabla w\|_\Omega \quad \forall w \in H^1_0(\Omega).$$

Example: $f(x,y) = 2x(1-x) + 2y(1-y)$

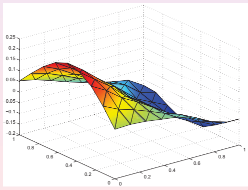


discrete solution v on coarse mesh compared to the exact solution
$u = x(1-x)y(1-y)$

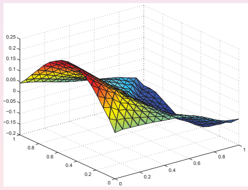
exact error^2 $=$ 1.62e-03



majorant $=$ 3.08e-03



majorant $=$ 2.56e-03



majorant $=$ 2.27e-03

Linear problems Nonlin. BC Poroelaticity Porous media Elastoplasticity

Majorant minimization problem

We have

$$||\nabla v - y^*|| + C_\Omega ||\mathrm{div} y^* + f|| \\ \leq [(1+\beta)||\nabla v - y^*||^2 + (1+\frac{1}{\beta})C_\Omega^2||\mathrm{div} y^* + f||^2]^{1/2}$$

for some $\beta > 0$. Therefore

Majorant minimization problem

Given $v \in H_0^1(\Omega)$ and $\beta > 0$, find the minimizer $y^* \in H(\Omega, \mathrm{div})$ of

$$\mathcal{M}(v, y^*, \beta) := (1+\beta)||\nabla v - y^*||^2 + (1+\frac{1}{\beta})C_\Omega^2||\mathrm{div} y^* + f||^2 \to \min$$

Jan Valdman Functional a posteriori error estimates

Majorant minimization

The minimization of the right hand side (majorant)

$$(1+\beta)||\nabla v - y^*||^2 + (1+\frac{1}{\beta})C_\Omega^2||\mathrm{div} y^* + f||^2 \to \min$$

leads to the linear system for the discrete solution y^*:

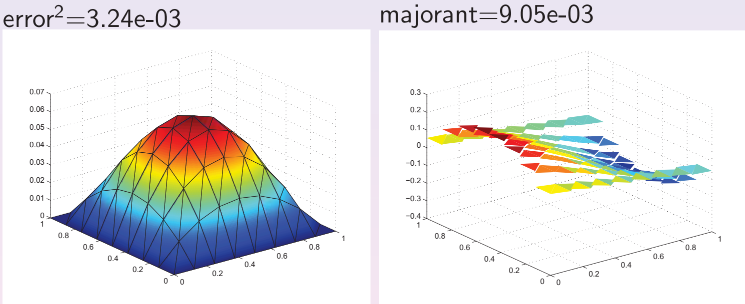
$$\left[(1+\beta)M + (1+\frac{1}{\beta})C_\Omega^2 DIVDIV\right] y^* = (1+\beta)l_1 - (1+\frac{1}{\beta})C_\Omega^2 l_2,$$

where matrices $M, DIVDIV$ represent the "mass" matrix and "divdiv" matrix defined by the equalities:

$$\int_\Omega uv\,dx = u^T Mv, \quad \int_\Omega \mathrm{div} u \ \mathrm{div} v\,dx = u^T DIVDIVv$$

$$(l_1)^T y^* = (\nabla v, y^*), \ (l_2)^T y^* = (f, \mathrm{div} y^*).$$

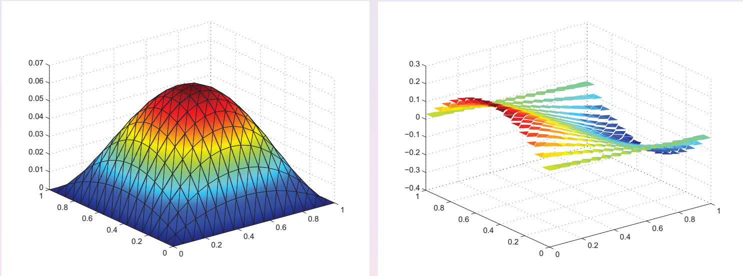
error^2=3.24e-03 majorant=9.05e-03



$\mathcal{T}_2 : I_{eff} = 1.67$

Figure: Discrete solution v (left) and y-component of the flux y (right).

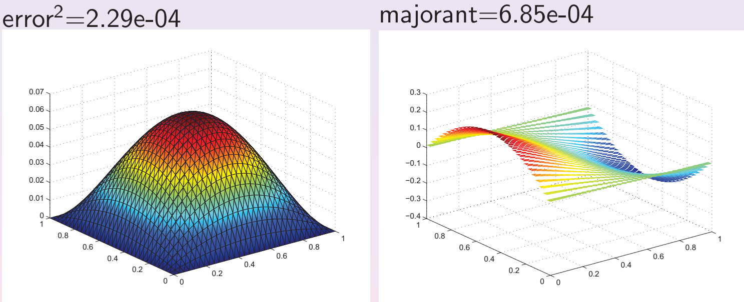
error^2=8.95e-04 majorant=2.63e-03



$\mathcal{T}_3 : I_{eff} = 1.71$

Figure: Discrete solution v (left) and y-component of the flux y (right).

error^2=2.29e-04 majorant=6.85e-04



$\mathcal{T}_4 : I_{eff} = 1.72$

Figure: Discrete solution v (left) and y-component of the flux y (right).

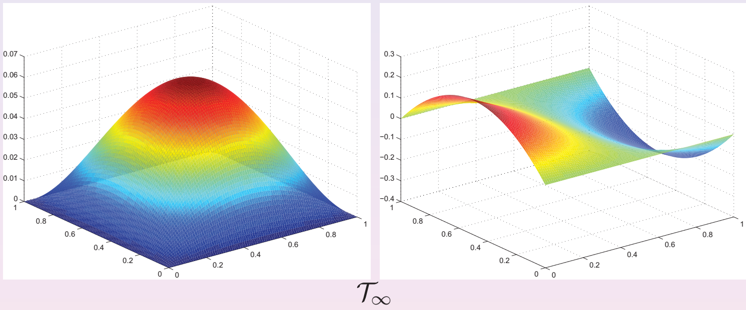


Figure: Exact solution v (left) and y-component of the exact flux y (right).

Computional efficiency for Raviart-Thomas (RT0) elements

System matrix: $(1+\beta)M + (1+\frac{1}{\beta})C_\Omega^2 DIVDIV$,
here $\beta = 1$ for all levels.

problem size	without preconditioner	multigrid preconditioner	time in seconds (without setup)
5	1	1	0.00
16	4	4	0.00
56	14	8	0.02
208	51	12	0.04
800	129	14	0.08
3136	264	15	0.24
12416	529	15	0.85
49408	1097	16	4.08
197120	2191	16	18.21
787456	4401	16	77.22

Table: Number of iterations of the CG method using no preconditioner or the multigrid (V cycles) preconditioner with the additive smoother of Arnold, Falk and Winther for 1 smothing step, tolerance=1e-8, Matlab!

Size of the discrete solution and of the discrete flux

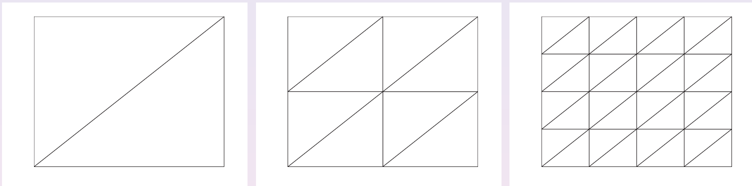


Figure: Refined meshes $\mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2$.

The discrete solution v is a piecewise linear nodal function (P1)
degrees of freedom on $\mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2$: 4, 9, 25

The discrete flux y is a lowest order Raviart Thomas function (RT0)
degrees of freedom on $\mathcal{T}_0, \mathcal{T}_1, \mathcal{T}_2$: 5, 16, 56

For fine triangulations it holds: number of edges $=$ number of nodes $\cdot$ 3

Papers

1. Jan Valdman, Minimization of Functional Majorant in A Posteriori Error Analysis based on H(div) Multigrid-Preconditioned CG Method. Advances in Numerical Analysis, vol. 2009, Article ID 164519 (2009)

Problem with nonlinear BC – Classical Formulation

Minimization problem

$$\int_\Omega \left(\frac{1}{2}|\nabla v|^2 - fv\right)\, dx + \mu \int_{\Gamma_1} |v|\, d\Gamma \to \min$$

among all $v \in U := \{v \in C^2(\Omega) \cap C^1(\Omega \cup \Gamma_1) \cap C^0(\Omega \cup \Gamma_0) : v|_{\Gamma_0} = 0\}$

Note that the variation leads to

Friction boundary condition

$$|u|\frac{\partial u}{\partial n} + \mu u = 0 \qquad \text{on } \Gamma_1$$

Problem with nonlinear bc – Classical Formulation

Friction boundary condition

$$|u|\frac{\partial u}{\partial n} + \mu u = 0 \qquad \text{on } \Gamma_1$$

Three parameter cases in our numerical examples:

1. $\mu \to +\infty$ -it implies the homogeneous Dirichlet boundary condition $u|_{\Gamma_1} = 0$.
2. $\mu = 0$ - it implies the homogeneous Neumann boundary condition $\frac{\partial u}{\partial n}|_{\Gamma_1} = 0$.
3. $\mu \in (0, +\infty)$ - this is a typical friction boundary condition.

Discrete solutions of the minimization problem

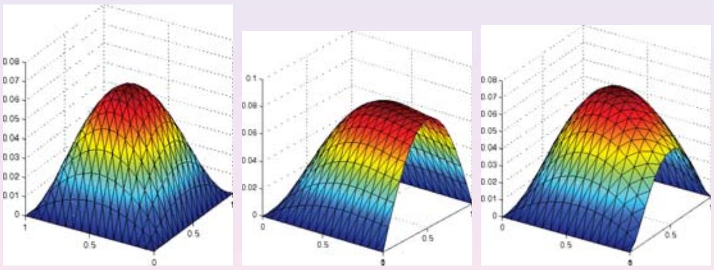


Figure: $\mu \to \infty$ (left), $\mu = 0$ (middle) and $\mu = 0.1$ (right).

Majorant estimate

Let u be an exact solution of the minimization problem and v its discrete approximation. Then it holds for all $\alpha, \beta > 0$

Estimate

$$\frac{1}{2}|||v-u|||_a^2 \leq (1+\beta)M_1(v,y^*) + \inf_{\xi^*} I_{\Gamma_1}(\gamma v, \delta_n y^*, \xi^*)$$
$$+\frac{1}{2}\left(1+\frac{1}{\beta}\right)(1+\alpha)C_\Omega^2 \mathbf{R}_\Omega^2(y^*)$$

for arbitrary y^* function from the (flux) test space

$$Q_{\Gamma_1}^* := \{y^* \in Y^* \mid \operatorname{div} y^* \in L_2(\Omega),\ \delta_n y^* \in L_2(\Gamma_1)\}\ .$$

Majorant estimate

Note that

$$M_1 = \frac{1}{2}||\nabla v - y^*||^2_{L^2(\Omega)}, \quad \mathbf{R}_\Omega(y^*) := ||\mathrm{div} y^* + f||_{L^2(\Omega)},$$

and using the *compound functional* the boundary term is defined as

$$I_{\Gamma_1}(\gamma v, \delta_n y^*, \xi^*) := \int_{\Gamma_1} \left(j(\gamma v) + j^*(\xi^*) - (\gamma v)\,\xi^* + \tfrac{\theta}{2}\,|\delta_n y^* + \xi^*|^2 \right) d\Gamma,$$

where

$$\theta := \left(1+\tfrac{1}{\beta}\right)\left(1+\tfrac{1}{\alpha}\right) C^2_{\Gamma_1}, \quad j(\xi) = \mu|\xi|, \quad j^*(\xi^*) = \begin{cases} 0, & \text{if } |\xi^*| \le \mu, \\ +\infty & \text{otherwise.} \end{cases}$$

Estimate of the boundary term $\inf_{\xi^*} I_{\Gamma_1}(\gamma v, \delta_n y^*, \xi^*)$

Summary:

$$\inf_{\xi^*} I_{\Gamma_1}(\gamma v, \delta_n y^*, \xi^*) \le \int_{\Gamma_1} \left(\mu|\gamma v| + \phi(\gamma v, \delta_n y^*, \mu)\right) d\Gamma,$$

where

$$\phi(\gamma v, \delta_n y^*, \mu) = \begin{cases} \frac{\theta}{2}(\delta_n y^* + \mu)^2 - \mu(\gamma v) & \text{if } \delta_n y^* < -\mu, \\ (-\delta_n y^*)(\gamma v) & \text{if } |\delta_n y^*| < \mu, \\ \frac{\theta}{2}(\delta_n y^* - \mu)^2 + \mu(\gamma v) & \text{if } \delta_n y^* > \mu. \end{cases}$$

Numerical results for $\mu = 0.1$

N	majorant	error^2/2	I_{eff}
25	2.9e-03	1.9e-03	1.22
81	9.0e-04	5.1e-04	1.33
289	2.7e-04	1.3e-04	1.44
1089	8.7e-05	3.3e-05	1.62
4225	2.8e-05	8.2e-06	1.87
16641	9.9e-06	1.9e-06	2.24
66049	3.9e-06	3.9e-07	3.17

Table: Majorant optimization on the same mesh.

Majorant optimized using an expensive nonlinear procedure
- can be improved!

Papers

S. Repin, J. Valdman, Functional A posteriori error estimates for problems with nonlinear boundary conditions, Journal of Numerical Mathematics 16 (2008), No. 1, 51-81.

Extension to elasticity with nonlinear boundary conditions

Friction boundary condition

Minimize the displacement v in the energy

$$\int_\Omega \left(\frac{1}{2}C\varepsilon(v):\varepsilon(v) - fv\right)\,dx + k_\tau \int_{\Gamma_1} |v_\tau|\,d\Gamma$$

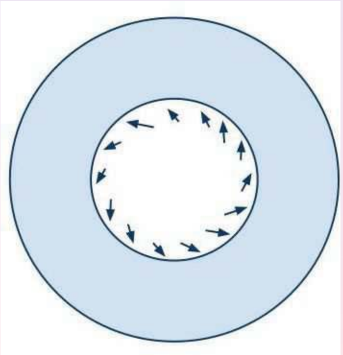
under the non-penetration condition

$$v_n = 0 \quad \text{on } \Gamma_1,$$

where $v = (v_\tau, v_n)$ is decomposed in the normal and tangential components on the boundary Γ_1.

Time dependent 2D symetric problem in Matlab

polar coordinates: $u = (u_r, u_\phi)$



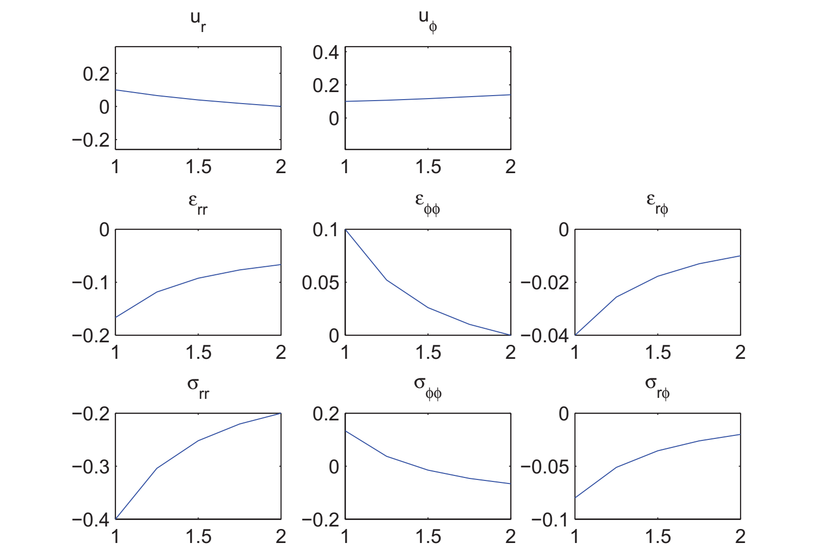
an inner radius $a = 1$,
an outer radius $b = 2$
friction parameter $k_\phi = 0.02$
Lamé parameters $\lambda = \mu = 1$

boundary conditons:

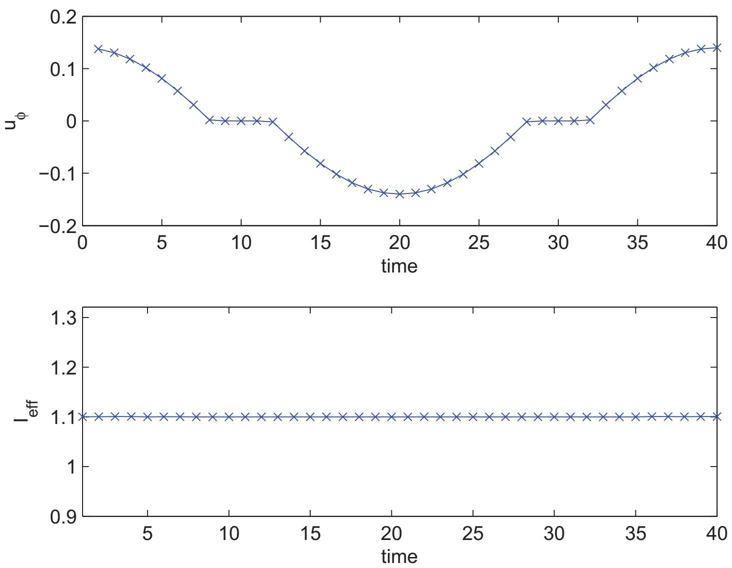
$$u_r(a) = u_\phi(a) = 0.1\cos(\frac{t\pi}{20})$$

discrete times: $t = 0, 1, 2, \ldots, 40$

Displacements, Strains, Stresses for t_0



Slip testing, index of eficiency



Papers

P. Neittaanmäki, S. I. Repin and J. Valdman, Functional a posteriori error estimates for elasticity problems with nonlinear boundary conditions. (in preparation)

Matlab solver can be downloaded at

http://www.mathworks.com/matlabcentral/fileexchange/authors/37756

Matematics model of the Barenblatt-Biot system

Barenblatt-Biot systems representing double diffusion in elastic porous media.

$$\begin{aligned} -\nabla \cdot (\mathbb{C}\varepsilon(u)) + \alpha_1 \nabla p_1 + \alpha_2 \nabla p_2 &= f(x,t) \\ c_1 \dot{p}_1 - \nabla \cdot (k_1 \nabla p_1) + \alpha_1 \nabla \cdot \dot{u} + \kappa(p_1 - p_2) &= h_1(x,t) \\ c_2 \dot{p}_2 - \nabla \cdot (k_2 \nabla p_2) + \alpha_2 \nabla \cdot \dot{u} + \kappa(p_2 - p_1) &= h_2(x,t) \end{aligned}$$

in which u is the displacement of the solid skeleton and p_1 and p_2 are the fluid pressures in the respective components.

Mathematical analysis of this model based on the theory of implicit evolution equations in Hilbert spaces is elaborated in

R. E. Showalter and B. Momken, *Single-phase flow in composite poroelastic media*, Math. Meth. Appl. Sci. 25 (2002), no. 2, 115–139.

Static model

Static case of the Barenblatt-Biot system

$$\begin{aligned} -\nabla \cdot (\mathbb{C}\varepsilon(u)) + \alpha_1 \nabla p_1 + \alpha_2 \nabla p_2 &= f(x) \\ -\nabla \cdot (k_1 \nabla p_1) + \kappa(p_1 - p_2) &= h_1(x) \\ -\nabla \cdot (k_2 \nabla p_2) + \kappa(p_2 - p_1) &= h_2(x) \end{aligned}$$

Combining a functional a posteriori error estimate for an elasticity problem

$$-\nabla \cdot (\mathbb{C}\varepsilon(u)) = f - \alpha_1 \nabla p_1 - \alpha_2 \nabla p_2 \tag{1}$$

and a functional a posteriori error estimate for a double-diffusion problem

$$-\nabla \cdot (k_1 \nabla p_1) + \kappa(p_1 - p_2) = h_1(x) \tag{2}$$
$$-\nabla \cdot (k_2 \nabla p_2) + \kappa(p_2 - p_1) = h_2(x) \tag{3}$$

which describes the flow of slightly compressible fluid in a general heterogeneous medium consisting of two components.

Problem (Variational formulation)

Assume that $(h_1, h_2) \in L^2(\Omega, \mathbb{R}^2)$. *Find* $\mathbf{p} = (p_1, p_2) \in H_0^1(\Omega, \mathbb{R}^2)$, *satisfying the system of variational equalities*

$$\int_\Omega k_1 \nabla p_1 \cdot \nabla q_1 + \int_\Omega \kappa (p_1 - p_2) q_1 \, dx = \int_\Omega (h_1(x) q_1 - k_1 \nabla \bar{p} \cdot \nabla q_1) \, dx$$
$$\int_\Omega k_2 \nabla p_2 \cdot \nabla q_2 + \int_\Omega \kappa (p_2 - p_1) q_2 \, dx = \int_\Omega (h_2(x) q_2 - k_2 \nabla \bar{p} \cdot \nabla q_2) \, dx$$

for all testing functions $\mathbf{q} = (q_1, q_2) \in H_0^1(\Omega, \mathbb{R}^2)$.

Dirichlet boundary conditions assumed for simplicity!

Problem (Abstract variational formulation)

Find $\mathbf{p} \in Q := H_0^1(\Omega, \mathbb{R}^2)$, *such that the equality*

$$a(\mathbf{p}, \mathbf{q}) = l(\mathbf{q})$$

holds for all $\mathbf{q} \in Q$. *The bilinear form* $a(\cdot, \cdot)$ *and the linear form* $l(\cdot)$ *are*

$$a(\mathbf{p}, \mathbf{q}) := \int_\Omega (\Lambda \mathbf{p} : (\mathbb{A} \Lambda \mathbf{q}) + \mathbf{p} \cdot \mathbb{B} \mathbf{q}) \, dx,$$
$$l(\mathbf{q}) := \int_\Omega (h \cdot \mathbf{q} - \mathbb{C} \Lambda \mathbf{q}) \, dx,$$

where $\Lambda \mathbf{q} := (\nabla q_1, \nabla q_2)$ *and* $\mathbb{A}$, $\mathbb{B}$ *and* $\mathbb{C}$ *are matrices formed by material dependant constants*

$$\mathbb{A} := \begin{pmatrix} k_1 & 0 \\ 0 & k_2 \end{pmatrix}, \quad \mathbb{B} := \begin{pmatrix} \kappa & -\kappa \\ -\kappa & \kappa \end{pmatrix}, \quad \mathbb{C} := \begin{pmatrix} k_1 \nabla \bar{p} & 0 \\ 0 & k_2 \nabla \bar{p} \end{pmatrix}$$

and h *is the right hand side vector* $h := \left(h_1 \; h_2 \right)^T$.

Problem (Equivalent minimization problem)

Find $\mathbf{p} \in Q = H_0^1(\Omega, \mathbb{R}^2)$ *satisfying*

$$F(\mathbf{p}) + G(\Lambda \mathbf{p}) = \inf_{\mathbf{q} \in Q} \{F(\mathbf{q}) + G(\Lambda \mathbf{q})\},$$

where

$$F : Q \to \mathbb{R}, \qquad F(\mathbf{q}) := \frac{1}{2} \int_\Omega \mathbf{q} \cdot \mathbb{B}\mathbf{q} \, dx - l(\mathbf{q}),$$

$$G : Y \to \mathbb{R}, \qquad G(\Lambda \mathbf{q}) := \frac{1}{2} \int_\Omega \Lambda \mathbf{q} : (\mathbb{A} \Lambda \mathbf{q}) \, dx.$$

Linear problems Nonlin. BC Poroelaticity Porous media Elastoplasticity

We need to find explicit forms of dual functionals

$$F^* : Q^* \to \mathbb{R}, \qquad F^*(\Lambda^* \mathbb{Y}^*) := \sup_{\mathbf{q} \in Q} \{\langle \Lambda^* \mathbb{Y}^*, \mathbf{q} \rangle - F(\mathbf{q})\},$$

$$G^* : Y^* \to \mathbb{R}, \qquad G^*(\mathbb{Y}^*) := \sup_{\Lambda \mathbf{q} \in Y} \{\langle\langle \mathbb{Y}^*, \Lambda \mathbf{q} \rangle\rangle - G(\Lambda \mathbf{q})\},$$

where $Y = Y^* := L^2(\Omega, \mathbb{R}^{2d})$, $\quad \Lambda^* \mathbb{Y}^* = (-\operatorname{div} y_1^*, -\operatorname{div} y_2^*)^T$
and construct the corresponding compound functionals

$$D_F : Q \times Q^* \to \mathbb{R}, \qquad D_F(\mathbf{q}, \Lambda^* \mathbb{Y}^*) := F(\mathbf{q}) + F^*(\Lambda^* \mathbb{Y}^*) - \langle \Lambda^* \mathbb{Y}^*, \mathbf{q} \rangle,$$

$$D_G : Y \times Y^* \to \mathbb{R}, \qquad D_G(\Lambda \mathbf{q}, \mathbb{Y}^*) := G(\Lambda \mathbf{q}) + G^*(\mathbb{Y}^*) - \langle\langle \mathbb{Y}^*, \Lambda \mathbf{q} \rangle\rangle.$$

By the the sum of D_F and D_G, we obtain the functional error majorant

$$M(\mathbf{q}, \mathbb{Y}^*) := D_F(\mathbf{q}, \Lambda^* \mathbb{Y}^*) + D_G(\Lambda \mathbf{q}, \mathbb{Y}^*), \tag{4}$$

which provides a guaranteed upper bound of the error:

$$\frac{1}{2} a(\mathbf{p} - \mathbf{q}, \mathbf{p} - \mathbf{q}) \leq M(\mathbf{q}, \mathbb{Y}^*) \quad \text{for all } \mathbb{Y}^* \in Y^*. \tag{5}$$

The majorant is fully computable and depends only on the approximation $\mathbf{q} \in Q$ and arbitrary variable $\mathbb{Y}^* \in Y^*$.

Jan Valdman Functional a posteriori error estimates

Lemma (dual functionals)

For $k_1, k_2 > 0$ and $\kappa > 0$, it holds

$$G^*(\mathbb{Y}^*) = \frac{1}{2}\int_\Omega \mathbb{A}^{-1}\mathbb{Y}^* : \mathbb{Y}^* \, dx,$$

$$F^*(\Lambda^*\mathbb{Y}^*) = \begin{cases} \frac{1}{4\kappa}\int_\Omega (\Lambda^*\mathbb{Y}^* + h)^2 \, dx & \text{if } \Lambda^* y_1^* + h_1 + \Lambda^* y_2^* + h_2 = 0, \\ +\infty & \text{otherwise.} \end{cases}$$

Note that the condition

$$\Lambda^*\mathbf{Y}_1^* + h_1 + \Lambda^*\mathbf{Y}_2^* + h_2 = 0$$

is weaker than two conditions

$$\Lambda^*\mathbf{Y}_1^* + h_1 = 0, \qquad \Lambda^*\mathbf{Y}_2^* + h_2 = 0,$$

which one would await from the general theory (COUPLING EFFECT!).

We obtain explicit expressions for the compound functionals

$$D_G(\Lambda\mathbf{q}, \mathbb{Y}^*) = \frac{1}{2}\int_\Omega \mathbb{A}(\Lambda\mathbf{q} - \mathbb{A}^{-1}\mathbb{Y}^*) : (\Lambda\mathbf{q} - \mathbb{A}^{-1}\mathbb{Y}^*) \, dx, \qquad (6)$$

$$D_F(\mathbf{q}, \Lambda^*\mathbb{Y}^*) = \begin{cases} \frac{1}{2}\int_\Omega \mathbb{B}\mathbf{q}\cdot\mathbf{q} \, dx + \frac{1}{4\kappa}\int_\Omega (\Lambda^*\mathbb{Y}^* + h)^2 \, dx \\ \qquad \text{if } \Lambda^*\mathbf{Y}_1^* + h_1 + \Lambda^*\mathbf{Y}_2^* + h_2 = 0, \\ +\infty \quad \text{otherwise.} \end{cases} \qquad (7)$$

and let us recall that

$$M(\mathbf{q}, \mathbb{Y}^*) := D_F(\mathbf{q}, \Lambda^*\mathbb{Y}^*) + D_G(\Lambda\mathbf{q}, \mathbb{Y}^*), \qquad (8)$$

provides a guaranteed upper bound of the error:

$$\frac{1}{2}a(\mathbf{p} - \mathbf{q}, \mathbf{p} - \mathbf{q}) \le M(\mathbf{q}, \mathbb{Y}^*) \quad \text{for all } \mathbb{Y}^* \in Y^*. \qquad (9)$$

Final estimate for the coupled poro-elastic system

It holds (**q** and v are known from computations)

$$a(\mathbf{p}-\mathbf{q},\mathbf{p}-\mathbf{q}) + \|\varepsilon(\mathbf{u}-\mathbf{v})\|^2_{\mathbb{L};\Omega}$$
$$\leq 2\widehat{C}\;\; M_{\beta_1,\beta_2}(\mathbf{q},\hat{\mathbb{Y}}^*) + (1+\beta_4+\beta_5)\left\|\varepsilon(\mathbf{v}) - \mathbb{L}^{-1}\tau\right\|^2_{\mathbb{L};\Omega} +$$
$$+\left(1+\frac{1}{\beta_4}+\beta_6\right) C^2 \left\|\operatorname{div}\tau + \mathcal{F} - \alpha_1\nabla q_1 - \alpha_2\nabla q_2\right\|^2_{\Omega},$$

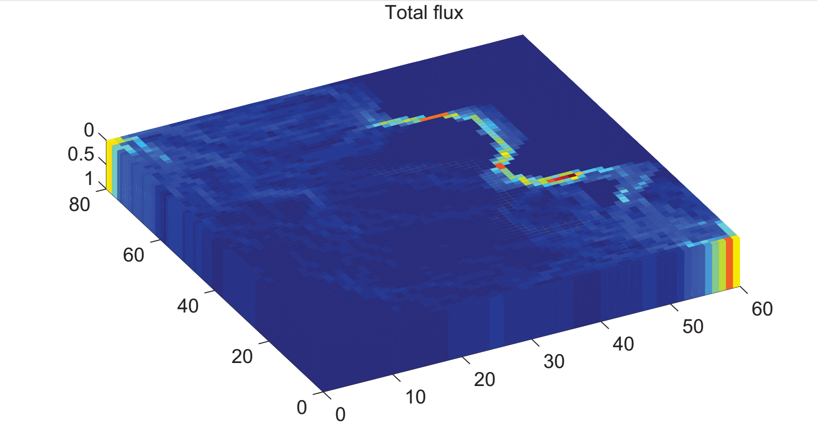
for all $\hat{\mathbb{Y}}^* \in Y^*_{div} := \{(\mathbf{Y}^*_1,\mathbf{Y}^*_2) \in Y^* : \Lambda^*\mathbf{Y}^*_1 + \Lambda^*\mathbf{Y}^*_2 \in L^2(\Omega)\}$,
for all $\tau \in Q$,
for all $\beta_1,\ldots,\beta_6 > 0$.

Here
$$\widehat{C} = 1 + C^2\left(1+\frac{1}{\beta_5}+\frac{1}{\beta_6}\right)\max\left\{\frac{1+\beta_3}{k_1},\frac{1+\beta_3}{k_2\beta_3}\right\},$$
where $C > 0$ satisfies Friedrichs' inequality
$$\|w\|_{L^2(\Omega)} \leq C\,\|\nabla w\|_{L^2(\Omega)}$$
valid for all $w \in H^1_0(\Omega)$.

Papers

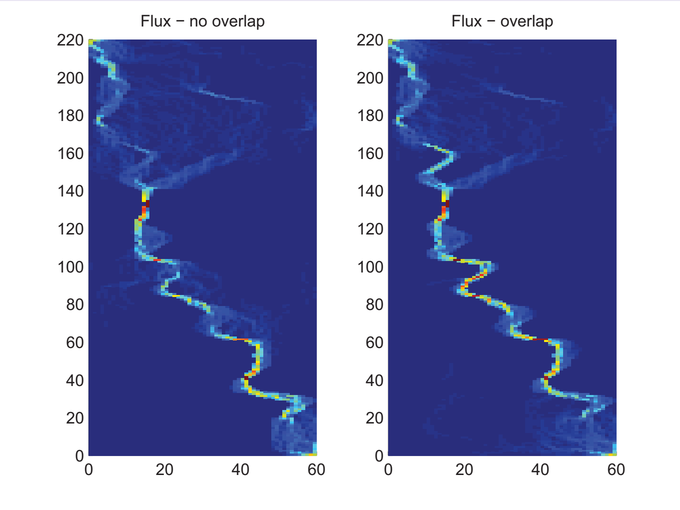
Jan Martin Nordbotten, Talal Rahman, Sergey Repin, Jan Valdman, A posteriori error estimates for approximate solutions of Barenblatt-Biot poroelastic model. Computational Methods in Applied Mathematics 10, No. 3, 302-315 (2010)

Reservoir simulator - project with SINTEF ICT



$$\begin{aligned} v &= -\lambda(\nabla p - \rho G) \\ \nabla \cdot v &= q \end{aligned}$$

Variational multiscale Method (VMS):



3D Matlab solver provided by SINTEF ICT

Similar results on mutiscale methods

MSc. thesis of Sergey Alyaev on
Adaptive Multiscale Methods Based on A Posteriori Error Estimates,
Bergen, June 2010

Basic estimate of the deviation from exact solution

For any $w \in H$ it holds

$$\frac{1}{2}|||u-v,p-q|||^2 \leq \mathcal{H}(v,q)-\mathcal{H}(u,p),$$

where $z=(u,p)$ is an exact elastoplastic solution
and $w=(v,q)$ is a discrete approximation.

where

$$|||u-v,p-q||| := \|\mathbb{C}(\varepsilon(u-v)-(p-q))\|^2_{\mathbb{C}^{-1}} + \sigma_y^2 H^2 \|q-p\|^2 .$$

Note, $H>0$ represents a hardening parameter (done for isotropic hardening model).

Perturbed problem

Original problem

$$\mathcal{H}(v,q) := \frac{1}{2}a(v,q;v,q) - l(v) + \int_\Omega \sigma_y |q|\,dx$$

Perturbed problem

$$\mathcal{H}_\lambda(v,q) := \frac{1}{2}a(v,q;v,q) - l(v) + \int_\Omega \sigma_y \lambda : q\,dx$$

where $\lambda \in \Lambda := \{\lambda \in L^\infty(\Omega, \mathbb{R}^{d\times d}) : |\lambda| \leq 1, tr(\lambda) = 0 \;\; a.\ e.\text{ in } \Omega\}$.

$$\sup_{\lambda\in\Lambda} \mathcal{H}_\lambda(v,q) = \mathcal{H}(v,q)$$

Lagrangian

Lagrangian

$$L_\lambda(v,q;\tau,\xi) := \int_\Omega \left(\tau : (\varepsilon(v) - q) - \frac{\mathbb{C}^{-1}\tau : \tau}{2} + \xi : q - \frac{|\xi|^2}{2\sigma_y^2 H^2} - fv\right) dx$$
$$+ \int_\Omega \sigma_y \lambda : q\,dx,$$

where $\tau \in Q := L^2(\Omega; \mathbb{R}^{d\times d}_{sym}), \xi \in Q_0 := \{q \in Q : tr(q) = 0 \;\; a.\ e.\text{ in } \Omega\}$.

$$\sup_{\tau\in Q, \xi\in Q_0} L_\lambda(v,q;\tau,\xi) = \mathcal{H}_\lambda(v,q)$$

First estimate

It holds for all $\lambda \in \Lambda$

$$\mathcal{H}(u,p) = \inf_{v,q} \mathcal{H}(v,q) \geq \inf_{v,q} \mathcal{H}_\lambda(v,q) \geq \inf_{v,q} L_\lambda(v,q;\tau,\xi)$$

which yields the estimate

$$\frac{1}{2}|||(u-v),(p-q)|||^2 \leq \mathcal{H}(v,q) - \inf_{v,q} L_\lambda(v,q;\tau,\xi)$$

How to compute $\inf_{v,q} L_\lambda(v,q;\tau,\xi)$?

Majorant estimate for equilibrated fields

$$\frac{1}{2}|||(u-v),(p-q)|||^2 \leq \inf_{(\tau,\xi)\in Q_{f_\lambda}} \mathcal{M}(v,q,\tau,\xi,\lambda),$$

where

$$\mathcal{M}(v,q,\tau,\xi,\lambda) = \frac{1}{2}\int_\Omega \mathbb{C}(\varepsilon(v) - q - \mathbb{C}^{-1}\tau) : (\varepsilon(v) - q - \mathbb{C}^{-1}\tau)\,dx$$
$$+\frac{1}{2}\int_\Omega \sigma_y^2 H^2 (q - \frac{1}{\sigma_y^2 H^2}\xi)^2\,dx + \int_\Omega (\sigma_y|q| - \sigma_y\lambda : q)\,dx$$

and

$$Q_{f_\lambda} := \{(\tau,\xi) \in Q \times Q_0 : \operatorname{div}\tau + f = 0, \tau^D = \xi + \sigma_y\lambda \;\; a.\,e.\text{ in } \Omega\}.$$

Structure of Functional Majorant

$\mathcal{M}(v, q, \tau, \xi, \lambda) = 0$ if and only if

$$\tau = \mathbb{C}(\varepsilon(v) - q), \quad (10)$$
$$\operatorname{div} \tau + f = 0, \quad (11)$$
$$\lambda : q = |q|, \qquad \lambda \in \Lambda, \quad (12)$$
$$\tau^D = \xi + \sigma_y \lambda, \quad (13)$$
$$\xi = \sigma_y^2 H^2 q. \quad (14)$$

These are conditions for the exact solution (u, p) of the elastoplastic minimization problem! The majorant naturally reflects properties of the original problem.

Majorant estimate for nonequilibrated fields

$$\frac{1}{2}|||(u - v), (p - q)|||^2 \leq \inf_{(\tau,\xi) \in Q_{f_\lambda}} \hat{\mathcal{M}}(v, q; \hat{\tau}, \lambda, \beta, \delta),$$

where

$$\hat{\mathcal{M}}(v, q; \hat{\tau}, \lambda, \beta, \delta) := \frac{1}{2}(1+\beta) \int_\Omega \mathbb{C}(\varepsilon(v) - q - \mathbb{C}^{-1}\hat{\tau}) : (\varepsilon(v) - q - \mathbb{C}^{-1}\hat{\tau}) \, dx$$
$$+ \frac{1}{2}(1+\delta) \int_\Omega \frac{1}{\sigma_y^2 H^2} (\hat{\tau}^D - \zeta)^2 dx + \int_\Omega (\sigma_y |q| - \sigma_y \lambda : q) \, dx$$
$$+ \frac{1}{2} \left[(1 + \frac{1}{\beta}) + \frac{c_2}{\sigma_y^2 H^2} (1 + \frac{1}{\delta}) \right] C^2 \, \|\operatorname{div} \hat{\tau} + f\|^2$$

and $\hat{\tau} \in Q_{\mathrm{div}} := \{\tau \in Q : \operatorname{div} \tau \in L^2(\Omega, \mathbb{R}^d)\}, \quad \zeta := \sigma_y^2 H^2 q + \sigma_y \lambda.$

Papers

Sergey Repin, Jan Valdman,
Functional a posteriori error estimates for incremental models in elasto-plasticity.
Cent. Eur. J. Math. 7, No. 3, 506-519 (2009)

Thank you for your attention!

Fast MATLAB assembly of FEM stiffness- and mass matrices in 2D and 3D: nodal elements

Talal Rahman, Jan Valdman

School of Engineering and Natural Sciences
University of Iceland, Reykjavik
email: janv@hi.is

ESCO 2010, Pilsen, 28.6. 2010

What will be vectorized?

A stifness matrix K and a mass matrix M defined as

$$K_{ij} = \int_\Omega \nabla\Phi_i \cdot \nabla\Phi_j \, dx, \quad M_{ij} = \int_\Omega \Phi_i\Phi_j \, dx,$$

where Ω is the domain of computation and ∇ denotes the gradient operator, Φ_i denote (nodal) shape functions.

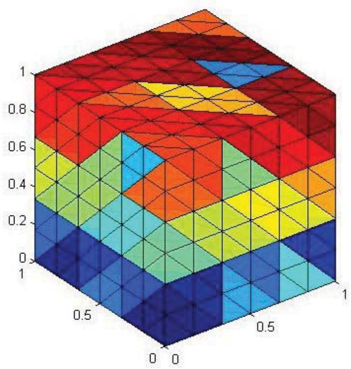


Figure: A triangulation of cube geometry.

FEM functions of interest

We are interested in *iso-parametric* shape functions Φ_i.

Then, the global-local element mapping reads (in 2D):

$$x = \sum_i \Phi_i(\xi, \eta) x_i, \quad y = \sum_i \Phi_i(\xi, \eta) y_i,$$

where (x, y) is a point on an element corresponding to the point (ξ, η) on the reference element, (x_i, y_i) are the global coordinates of the node corresponding to the shape function Φ_i.

in 3D:

$$x = \sum_i \Phi_i(\xi, \eta, \theta) x_i, \quad y = \sum_i \Phi_i(\xi, \eta, \theta) y_i, \quad z = \sum_i \Phi_i(\xi, \eta, \theta) z_i,$$

It holds also for higher order (quadratic, cubic, etc.) shape functions!

Examples of iso-parametric shape functions

Examples:

1. $\Phi_1 = 1 - \xi - \eta, \quad \Phi_2 = \xi, \quad \Phi_3 = \eta$
2. $\Phi_1 = 1 - \xi - \eta - \theta, \quad \Phi_2 = \xi, \quad \Phi_3 = \eta, \quad \Phi_4 = \theta$
3. $\Phi_1 = 1 - 3\xi - 3\eta + 4\xi\eta + 2\xi^2 + 2\eta^2, \quad \Phi_2 = -\xi + \xi^2, \quad \Phi_3 = -\eta + \eta^2,$
   $\Phi_4 = 4\xi\eta, \quad \Phi_5 = -4\xi\eta - 4\xi^2, \quad \Phi_6 = -4\xi\eta - 4\eta^2$

What needs to be vectorized?

For every element T and every integration point IP we need to vectorize:

in 2D -

$det \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix}$ storage: number of IP x number of T

$\begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix}^{-1}$ storage: 2 x 2 x number of IP x number of T

Vectorizations for higher order elements is not more difficult!

Concept of vectorization in Matlab - an array of matrices

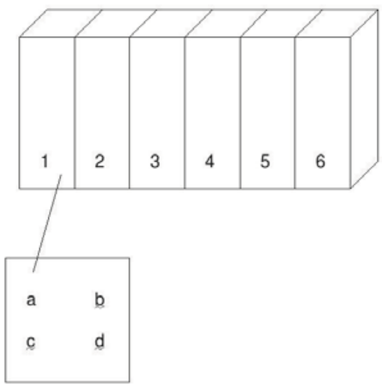


Figure: Computation of determinant (for linear elements in 2D).

Two ways of computing determinant

1. for all elements: compute determinant of 2 x 2 matrix using MATLAB command
$$det\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

2. construct long vectors
$$\begin{aligned} a &= (a_1, \dots, a_{\#T}), \\ b &= (b_1, \dots, b_{\#T}), \\ c &= (c_1, \dots, c_{\#T}), \\ d &= (d_1, \dots, d_{\#T}), \end{aligned}$$
and compute determinant using
$$a. * c - b. * d$$

Operations on array matrices

- smamt .... multiplication of scalar matrix and array of matrices
- aminv .... inversion (over first two indices) of an array of matrices
- amsm .... array of matrices times scalar matrix
- and many more located in our vectorization library (directory)

sm ..... scalar matrix, am .... array of matrices, t .... transpose

Example of Matlab code for computation of shape derivative:

```
for poi = 1:nop    %loop over all integration points
    %computation of Jacobian
    %inverse and determinant of Jacobian
    %computation of derivatives of shape functions

    tjac = smamt(dshape(:,:,poi),coord);

    [tjacinv,tjacdet] = aminv(tjac);

    dphi(:,:,poi,:)   = amsm(tjacinv,dshape(:,:,poi));
    ......
end
```

Our vectorization: basic features

- modularity
- reusability
- readability (still fast enough)

Matlab code web page

Demo located at Matlab Central.

Numerical performance in 2D: linear elements

mesh level	size of A	assembly of A time (sec)	assembly of M time (sec)
4	289	0.0047	0.0018
5	1089	0.0106	0.0053
6	4225	0.0308	0.0209
7	16641	0.1456	0.1021
8	66049	0.6662	0.4630
9	263169	2.835610	2.017507
10	1050625	11.991354	8.664730
11	4198401	50.309788	36.847517

Table: Times of assembly of a stiffness matrix A and a mass matrix M in 2D using $P1$ triangular elements.

Numerical performance in 2D: quadratic elements

mesh level	size of A	assembly of A time (sec)	assembly of M time (sec)
3	289	0.0064	0.0022
4	1089	0.0150	0.0058
5	4225	0.0471	0.0226
6	16641	0.2098	0.1045
7	66049	1.0146	0.4599
8	263169	4.4870	2.0471
9	1050625	18.2429	9.2360
10	4198401	78.0179	38.1942

Table: Times of assembly of a stiffness matrix A and a mass matrix M in 2D using $P2$ triangular elements.

Numerical performance in 3D: linear elements

mesh level	size of K and M	assembly of K time (sec)	assembly of M time (sec)
1	343	0.0661	0.0184
2	2197	0.1025	0.0462
3	15625	0.8524	0.4105
4	117649	7.0801	3.6764
5	912673	61.1436	33.4952

Table: Times of assembly of stiffness matrix A and mass matrix M in 3D using $P1$ tetrahedral elements.

Future extensions

1. rectangular elements
2. linear elasticity
3. Hdiv (Hcurl) problems

Thank you for your attention!

Computational fluid dynamics with OpenFOAM

Halldór Pálsson

University of Iceland

Outline

- Computational framework
- Mesh generation
- Solvers for various problems
- Numerical schemes
- Summary and conclusions

OpenFOAM

General description:

- OpenFOAM (Field Operation And Manipulation) is a general purpose tool set for solving partial differential equations.
- It is based on a huge collection of tailor made C++ classes and programs, including advanced solvers for complex flow problems.
- It is free of use and can be tweaked and modified according to the GNU General Public License.

It is *not*:

- A fully fledged software environment, keeping track of the problem from the point of geometrical definition to presenting results.

Ready to use software

OpenFOAM (current version is 1.7) is distributed with a large number of working programs. They can be divided into:

- Utilities for pre-processing, including mesh generation and manipulation.
- Solvers for various physical problems, flow, turbulence, heat transfer, solid mechanics, magnetohydrodynamics, e.t.c.
- Utilities for post-processing, such as calculation of derived values as well as integration and averaging of fields.

OpenFOAM has a built in support for parallel processing through the Message Passing Inteface.

The programming environment

- Based on C++, with all the advanced features included, such as
  - Object oriented programming
  - Object inheritance
  - Polymorphism and virtual objects
  - Templated classes and functions
- Divided into layers: Basic tools, containers, algorithms, solvers and utilities.
- Documented with Doxygen, giving an `html` interface for the whole software.

Using the programming environment

Strengths:

- Very customizable, everything can be modified and additions can be made
- Object oriented approach, resulting in a logical code structure
- Layers of complexity, e.g. specifying new PDE's without concerning parallel processing or numerical schemes

Weaknesses:

- Lack of proper documentation with examples
- The code is huge! Difficult for new users to familiarize
- The code is constantly being changed (improved hopefully), some designs are strange/peculiar

Using the utilities and solvers

Programs are divided into

Utilities for mesh generation and manipulation, data processing, etc.

Solvers for different problems, e.g. Laplace equation, flow, etc.

- Most programs are acompanied by files, for control. They are commonly called *dictionaries*
- Everything else is controlled by various files
- Files are stored in a well defined directory structure

Mesh generation

Basic mesh building blocks:

- Vertices, defining face corners
- Faces, which are generally polyhedra, but should be close to planar
- Cells, which consists of four or more polyhedra faces

Mesh properties:

- The mesh structure is stored in human readable data files
- Neighbour cells share a common face, but do not need to be joined at corners (as in FEM methods)
- Faces are either interior faces (between cells) or boundary faces

Mesh conversion

Meshes can be imported from other programs ans systems, using special utilities. Possible imports are:

- Ansys mesh file, by the utility `ansysToFoam`
- Gambit (Ansys Inc.), by `gambitToFoam`
- CFX (Ansys Inc.), by `cfx4ToFoam`
- STAR-CD, by `starToFoam`
- GMSH, by `gmshToFoam`
- Tetgen, by `tetgenToFoam`
- Netgen, by `netgenNeutralToFoam`

Meshing tools

Two tools are available as a part of OpenFOAM:

blockMesh A hexagonal mesher

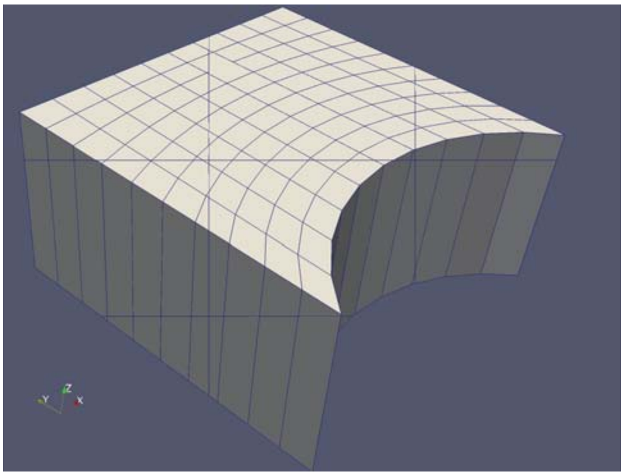
- Simple, but primitive
- Allows curved edges and faces
- Allows a linear mesh grading between faces

snappyHexMesh For complex geometries

- Base mesh is needed (typically hexagonal)
- Geometry must be specified as a closed STL (Stereolithography) surface
- Grading and mesh quality can be controlled in detail

The blockMesh program

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      blockMeshDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * //

convertToMeters 1.0;

vertices
(
  (-1 -1 0) (1 -1 0) (1 1 0) (-1 1 0)
  (-1 -1 1) (1 -1 1) (1 1 1) (-1 1 1)
);

blocks
(
  hex (0 1 2 3 4 5 6 7) (10 10 1) simpleGrading (1 1 1)
);

edges
(
  arc 0 1 (0 -0.5 0)
  arc 4 5 (0 -0.5 1)
);

patches
(
 patch top ( (2 3 7 6) )
 patch bottom ( (0 1 5 4) )
 patch left ( (0 3 7 4) )
 patch right ( (1 2 6 5) )
);

// ************************************************* //
```



Solvers for flow problems

- Incompressible flow
  - Transient (PISO) and steady (SIMPLE)
  - Porous flow
  - Shallow water equations
- Compressible flow
  - Transient (PISO) and steady (SIMPLE)
  - Porous flow
  - Sonic flow with high Mach numbers
- Multiphase flow
  - Two phases, liquid and gas
  - Multiple phases
  - Cavitation and phase change

Turbulence modeling

Three approaches are in general available:

- Reynolds averaged stress models (RAS or RANS)
  - About 15 common models are included ($k - \epsilon$, $k - \omega$ and others)
- Large eddy simulation (LES)
  - OpenFOAM was originally developed for LES
  - Many models (or filters) available (15-20).
- Direct numerical simulation (DNS)

Many of the solvers can use diffenrent turbulence approaches with a single switch. For an time dependent problem: Unsteady-RANS, LES, DNS, Laminar.

Other solvers

- Basic solvers for scalar transport, Laplace equation and Poisson equation
- Heat transfer, with and without flow effects (buyoant flow)
- Combustion and particle tracking
- Electrostatics and magnetohydrodynamics (MHD)
- Structural analysis and stresses

Different solvers exist for each case, including compressible effects and turbulence.

Numerical schemes

- The code is based on the finite volume method (FVM)
- Numerical schemes are adjusted in a single file
- Adjustments can be made while running solvers
- Schemes can be selected for each individual operator of a problem
  - Gradient, divergence and curl
  - Laplacian
  - Time derivative (first or second order)
  - Interpolation (used to calculate face values)

All schemes can be evaluated explicitly, or used to generate a linear system of equations

Numerical schemes: Advection

Advection of field ψ where ϕ is a surface flux

$$\nabla \cdot (\phi\psi) \quad \rightarrow \quad \text{div(phi,psi)}$$

A total of 51 schemes available, e.g.

- Gauss upwind
- QUICK (Quadratic)
- Cubic
- vanLeer
- MUSCL
- Various limited schemes

Numerical schemes: Laplacian

Laplacian of a field (vector or scalar) with a coefficient

$$\nabla \cdot (\Gamma \nabla \psi) \quad \rightarrow \quad \text{laplacian(gamma,psi)}$$

- Evaluation of gradient at faces (interpolation schemes)
- Gauss integration to evaluate the laplacian
- Limiters available for gradients
- Correction for non-orthogonal meshes

Gradients can also be evaluated with

$$\nabla \psi \quad \rightarrow \quad \text{grad(psi)}$$

Numerical schemes: Time integration

Two operators are available:

$$\frac{\partial \psi}{\partial t} \quad \rightarrow \quad \text{ddt(rho,psi)}$$
$$\frac{\partial^2 \psi}{\partial t^2} \quad \rightarrow \quad \text{d2dt2(rho,psi)}$$

Available schemes:

- Implicit Euler
- Crank Nicholson, central difference with weights
- Backward, second order difference using two time values to evaluate the third
- Steady-state, for steady problems

An example `fvSchemes` file

```
ddtSchemes
{
    default         Euler;
}

gradSchemes
{
    default         Gauss linear;
}

divSchemes
{
    default         none;
    div(phi,T)      Gauss limitedLinear 1;
    div(gflux,rhok)   Gauss limitedLinear 1;
}

laplacianSchemes
{
    default         none;
    laplacian((kappa|nu),p) Gauss linear corrected;
    laplacian((nu|Pr),T) Gauss linear corrected;
}

interpolationSchemes
{
    default         linear;
}
```


Solvers for linear systems

Three iterative solvers can be used:

- Preconditioned conjugate gradients (PCG), for symmetric positive definite systems.
- Bi-conjucate gradients (PBiCG), for unsymmetric systems
- Generalized algebraic multigrid (GAMG)

Possible preconditioners are:

- Incomplete Cholesky factorization
- Incomplete LU factorization (for unsymmetric systems)
- GAMG iterations

Solvers are specified in a `fvSolution` file

An example `fvSolution` file

```
solvers
{
    p
    {
        solver          GAMG;
        tolerance       1e-06;
        relTol          0.01;
        smoother        GaussSeidel;
        nPreSweeps      0;
        nPostSweeps     2;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator    faceAreaPair;
        mergeLevels     1;
    }

    U
    {
        solver          PBiCG;
        preconditioner  DILU;
        tolerance       1e-05;
        relTol          0.1;
    }

    k
    {
        solver          PBiCG;
        preconditioner  DILU;
        tolerance       1e-05;
        relTol          0.1;
    }

    epsilon
    {
        solver          PBiCG;
        preconditioner  DILU;
        tolerance       1e-05;
        relTol          0.1;
    }
}

SIMPLE
{
    nNonOrthogonalCorrectors 3;
}

relaxationFactors
{
    p               0.3;
    U               0.7;
    k               0.7;
    epsilon         0.1;
    R               0.7;
    nuTilda         0.7;
}
```


Parallel processing

- Domain decomposition is applied:
  - Simple decomposition based on coordinate directions
  - Metis decomposition algorithm
  - Manual decomposition, based on cell selection
- Uses the openMPI system (message passing interface)
- Processes run on either shared memory systems (multicore) or distributed systems
- Easy to set up and run on clusters with queuing systems

Decomposition is controlled by a single file, `decomposeParDict`.

Customization of solvers

Use available source code from other solvers

```
#include "readPISOControls.H"
#include "CourantNo.H"

// Pressure-velocity PISO corrector
{
    // Momentum predictor

    fvVectorMatrix UEqn
    (
        fvm::ddt(U)
      + fvm::div(phi, U)
      + turbulence->divDevReff(U)
    );

    UEqn.relax();

    if (momentumPredictor)
    {
        solve(UEqn == -fvc::grad(p));
    }
```

$$\frac{\partial \vec{U}}{\partial t} + \vec{U} \cdot \nabla \vec{U} + \nabla \cdot \bar{\tau} = -\nabla p$$

```
// Solution of the heat equation
volScalarField kappaEff
(
    "kappaEff",
    turbulence->nu()/Pr + turbulence->r
);
for (int nonOrth=0; nonOrth<=nNonOrthCc
{
    solve
    (
        fvm::ddt(T)
      + fvm::div(phi, T)
      - fvm::laplacian(kappaEff, T)
    );
}

runTime.write();
```

$$\frac{\partial T}{\partial t} + \vec{U} \nabla T - \nabla \cdot (\kappa_e \nabla T) = 0$$

Writing utilities

Possibilities: Boundary conditions, initial conditions, field calculations, others

```
if (fieldHeader.headerClassName() == "volScalarField")
{
    Info<< "    Reading volScalarField " << fieldName << endl;
    volScalarField field(fieldHeader, mesh);
    volVectorField grad(fvc::grad(field));

    scalar area = gSum(mesh.magSf().boundaryField()[patchi]);
    scalar sumField = 0;

    if (area > 0)
    {
        sumField = gSum
        (
            mesh.Sf().boundaryField()[patchi]
          & grad.boundaryField()[patchi]
        ) / area;
    }

    Info<< "    Average of flux of " << fieldName << " over patch "
        << patchName << '[' << patchi << ']' << " = "
        << sumField << endl;
}
```

Summary

The good things:

- A great number of solvers with many modeling options.
- Robust and fast algorithms, easily run in parallel.
- Fully customizable, for new developments or modifications.
- Solution procedures are in *batch mode* by default.
- No license fees!!!

The bad things:

- Software basis is huge and the design is rather complex.
- No decent application for generating complex meshes.
- Documentation is adequate in some areas, very poor in others.

Thank you for your attention

For more information visit: `http://www.openfoam.com`

Case studies in OpenFOAM

Halldór Pálsson

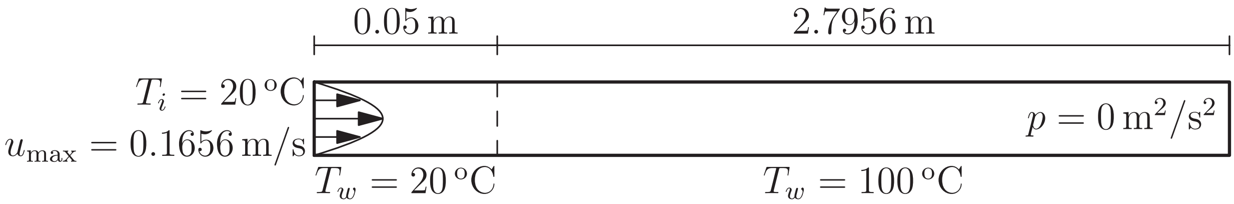
University of Iceland

Outline

- Laminar flow in a heated pipe
- Swirling flow for increased heat transfer
- Thermal transport in porous media
- Flow conditions around trawl doors
- Conclusions

Laminar flow in a heated pipe

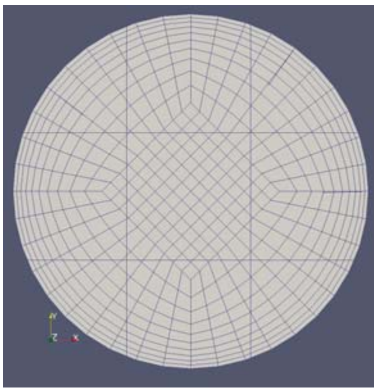
- Flow is laminar and steady along the whole pipe
- The wall has constant temperature at a given section, higher than the fluid temperature
- A well known problem with an analytical solution

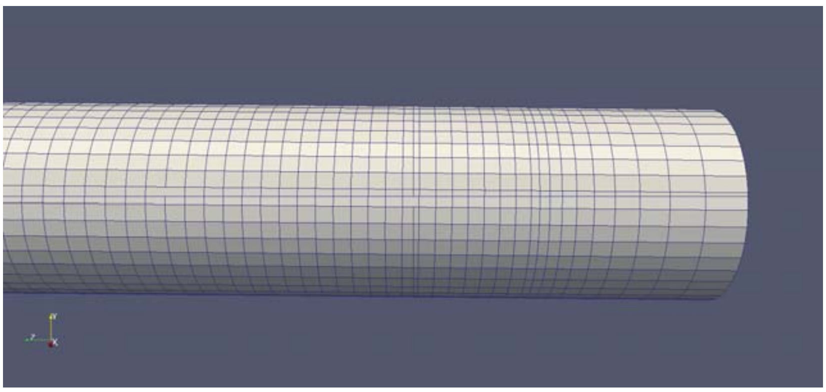


The pipe diameter is $0.05\,\mathrm{m}$, the Reynolds number 319.5 and the Peclet number 223.6.

Mesh generation

The mesh was generated with the `blockMesh` utility, with a total of 10 blocks, resulting initially in 105000 cells.





Note that the domain is very long in the z–direction, compared to the other two directions.

Boundary conditions

- No slip conditions at the pipe walls, and a constant wall temperature.
- Pressure is constant at the outlet.
- Temperature is given at the inlet, but a parabolic laminar velocity profile must be specified.

The profile is set by a custom made utility program `laminarPipeEntrance`, based on diameter and mean velocity.

Solvers used for computation

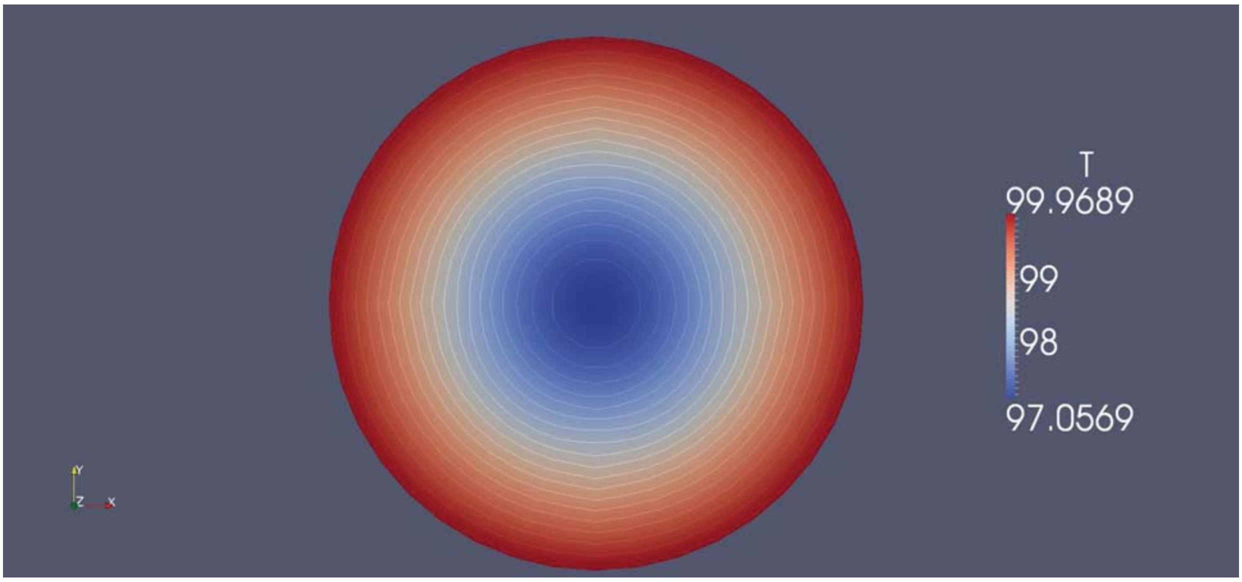
Two steps with different solvers are necessary to obtain a solution:

For flow The program `simpleFoam`, for steady incompressible flow with or without a RANS turbulence model

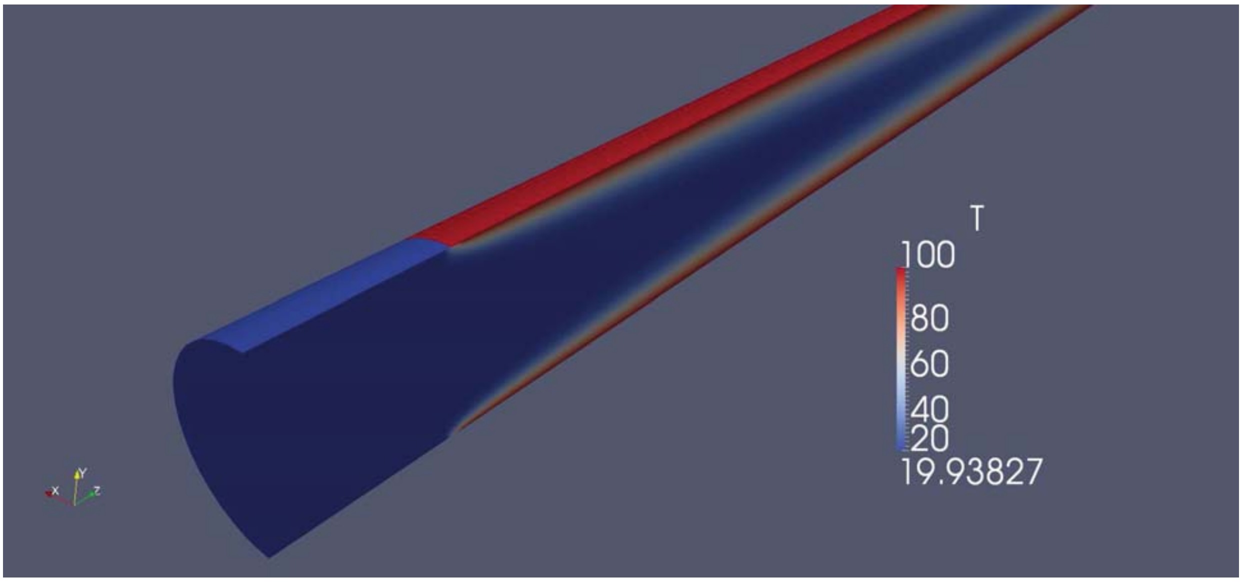
For heat The program `scalarTransportFoam`, for unsteady advective-diffusive heat transport with constant diffusion. The program is used in *steadyState* mode.

The length/diameter ratio is large (almost 56), so convergence of `simpleFoam` is slow.

Results, exit temperature profile



Results, inlet and heated wall



Note that the minimum temperature is lower than the physical minimum of $20\,^{\circ}\mathrm{C}$.

Velocity weighted average temperature

A useful numerical value for comparison is the average temperature, weighted by velocity:

$$\bar{T} = \frac{\int_A T\vec{u} \cdot d\vec{A}}{\int_A \vec{u} \cdot d\vec{A}}$$

- Computation can be performed by using surface slices of the geometry, notably the utility `sample` in OpenFOAM.
- A dictionary has to be constructed in the `system` directory: `sampleDict`
- The result is a triangulated surface *with field values given at the corner points*

Calculation for a single triangle

Each triangle is defined by closed loop of three vectors, $\vec{v}_1$, $\vec{v}_2$ and $\vec{v}_3$. Area is given by

$$\vec{A} = \frac{(\vec{v}_2 - \vec{v}_1) \times (\vec{v}_3 - \vec{v}_1)}{2}$$

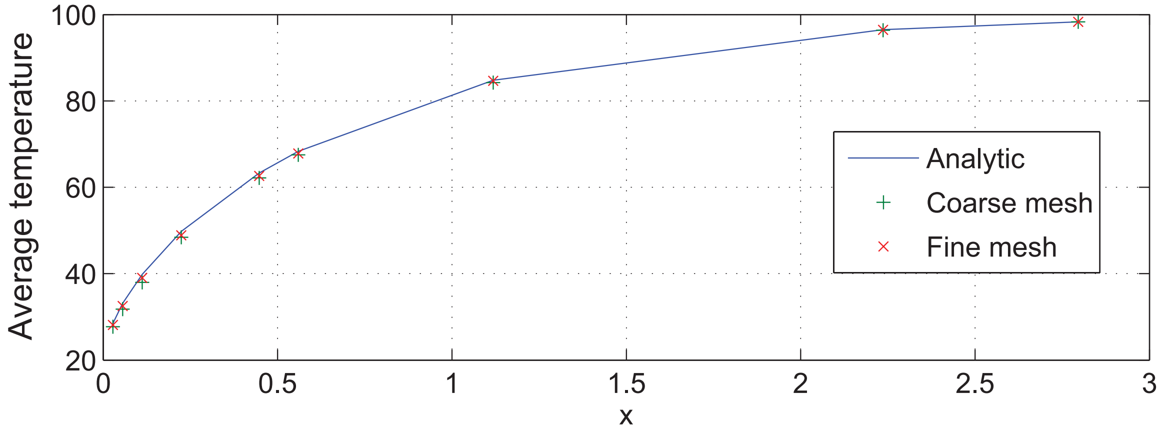
velocity flux by

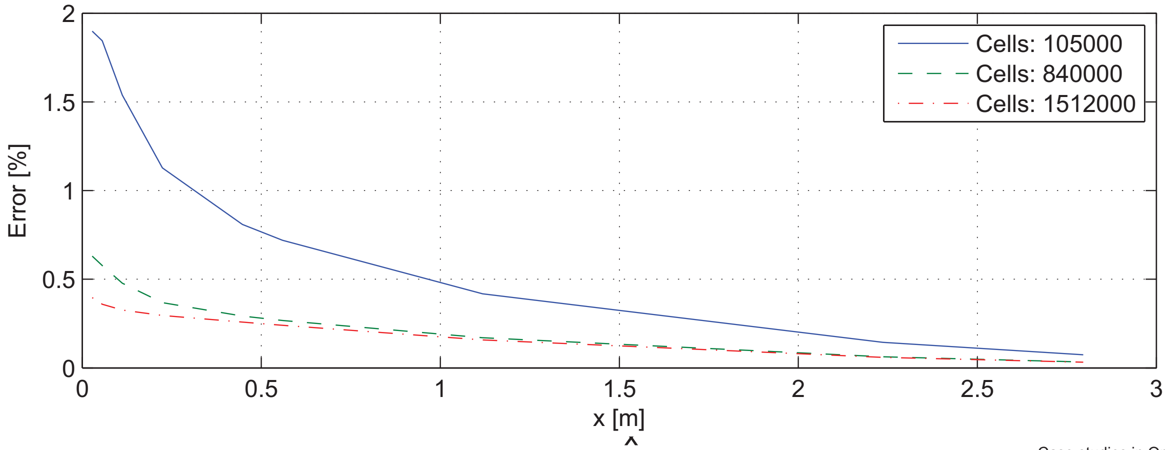
$$\hat{u} = \int_A \vec{u} \cdot d\vec{A} = \frac{(\vec{u}_1 + \vec{u}_2 + \vec{u}_3)}{3} \cdot \vec{A}$$

and temperature weighted flux by

$$\hat{T} = \int_A T\vec{u} \cdot d\vec{A} = \frac{1}{12} \begin{bmatrix} \vec{A} \cdot \vec{u}_1 & \vec{A} \cdot \vec{u}_2 & \vec{A} \cdot \vec{u}_3 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix}$$

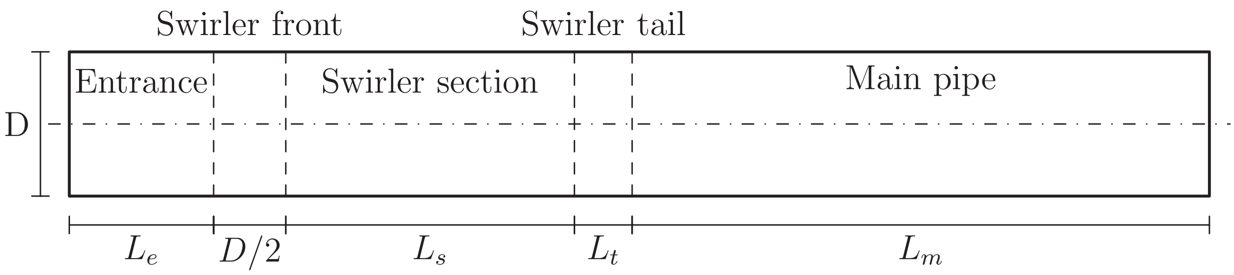
Comparison with theory





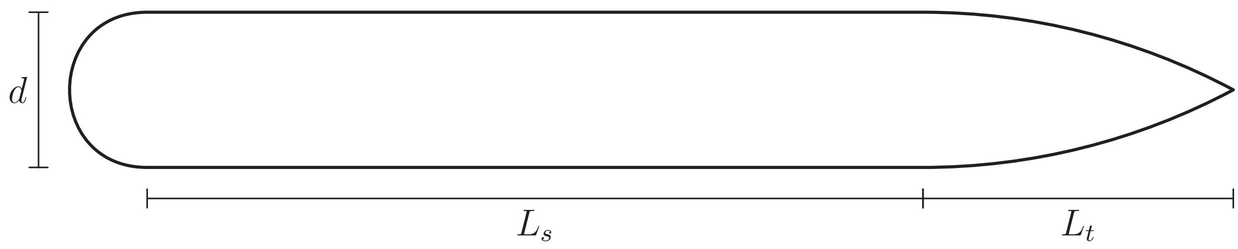
Induced swirling flow in a pipe

- A pipe is modeled, including a swirling device in one section
- The entrance is laminar, no heating takes place yet
- The flow model is steady state with realizable $k - \epsilon$ turbulence modeling.
- The model geometry is dictated by a set of parameters, specified in the dictionary file `swirlerDict`



Swirl device geometry

- The device consist of a narrow cylinder in the pipe center
- Thin fins or baffles connect the cylinder to the pipe, in the L_s section
- The fins are twisted along the axis, to direct flow into a swirling motion



Strategy for mesh generation

Three observations are used:

1. The pipe and the center piece are axi-symmetric in shape, so it is natural to use this symmetry to simplify the model generation.
2. The fins are distributed equally in the angular direction around the centerpiece, making it possible to identify their position in the axi-symmetric setup.
3. The fins are twisted along the length of the pipe, which can be performed in modeling terms with a coordinate transform, based on axial location.

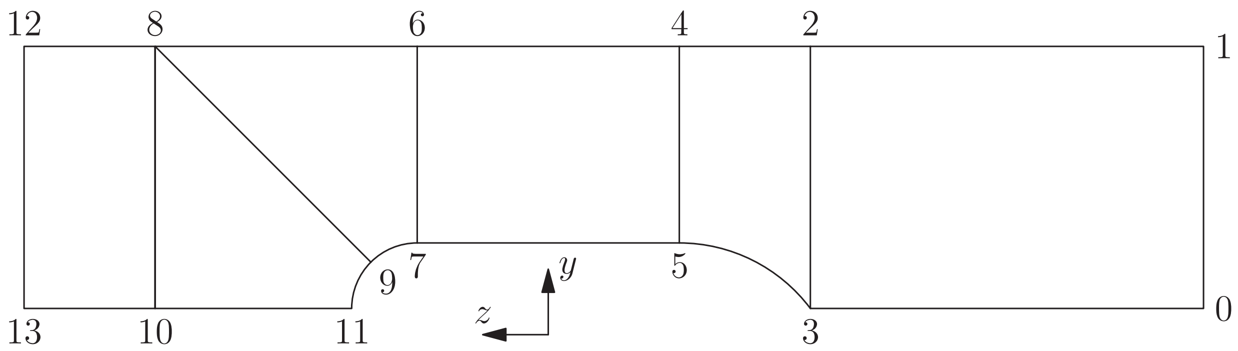
File structure in OpenFOAM

- A set of directories and files have to be generated
- The slanted ones are generated by the custom utility `swirlerMesh`
- Others are made/copied by the user

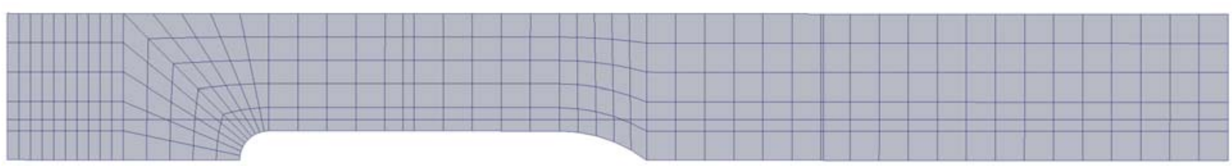
- **Case**
  - **planeCase**
    - **constant**
      - **polyMesh**
        - *blockMeshDict*
    - **system**
      - controlDict
      - fvSchemes
      - fvSolution
  - **constant**
    - swirlerDict
    - *extrudeProperties*
    - *setBatchFile*
  - **system**
    - controlDict
    - fvSchemes
    - fvSolution
    - createPatchDict

An axi-symmetrical cross section

The `blockMesh` utility is used to create a mesh, based on the following vertex and block definitions



Resulting in a planar mesh



Extrusion for generating a volume

- The utility `extrudeMesh` is used to rotate the plane around the z–axis
- Controlled by a dictionary `constant/extrudeProperties`, generated by `swirlerMesh`
- Specifications are:
  - Extrude type, *wedge*
  - Number of cells in extrusion direction
  - Rotation axis, position and direction
  - Rotation angle. If 360° then end planes are connected.

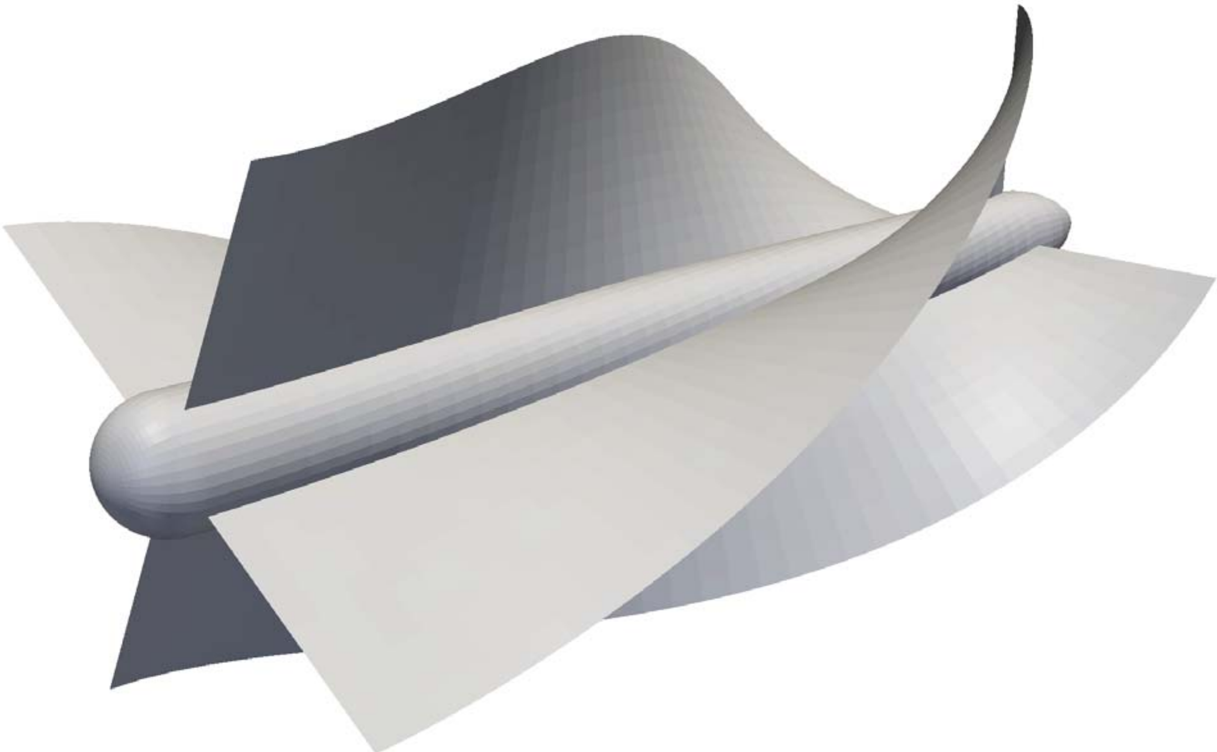
Specifying boundaries

Boundaries involve *inlet*, *outlet*, *pipe wall*, *centerpiece* and *fins*. The generation procedure is:

1. Divide boundary, using utility `autoPatch 45`. The parameter 45 is a *feature angle*
2. Use utility `setSet` to select internal faces that will become fins, based on `constant/setBatchFile`
3. Transform face sets into face zones, using utility `setsToZones`
4. Create fins with utility `createBaffles`. A face set and a name of a new boundary patch must be given
5. Redefine boundary patch names and clean up all boundary definitions, using utility `createPatch` which uses the dictionary `system/createPatchDict`

Twisting the mesh

The mesh is twisted along a predefined curve.



Mesh generation summary

In the whole, the following commands must be executed, e.g. as a script.

```
swirlerMesh
blockMesh -case planeCase
extrudeMesh
autoPatch 45 -overwrite
setSet -noVTK -batch constant/setBatchFile
setsToZones -noFlipMap
createBaffles baffles otherSide -overwrite
createPatch -overwrite
twistMesh
```

Boundary conditions

There are three types of boundary conditions used in the model:

- No-slip conditions at walls, using turbulent wall functions.
- Constant pressure at the outlet.
- Given laminar velocity profile at the inlet, computed with a custom utility `laminarSwirlerEntrance`

$$u_z(x,y) = -2\bar{u}\left(1 - \frac{2\sqrt{x^2+y^2}}{D}\right)$$

They are specified in the directory `0` as files `p`, `U`, `k` and `epsilon`

Results: Pressure

The whole pipe:

A cut through the center, around the fins:

Results: Velocity

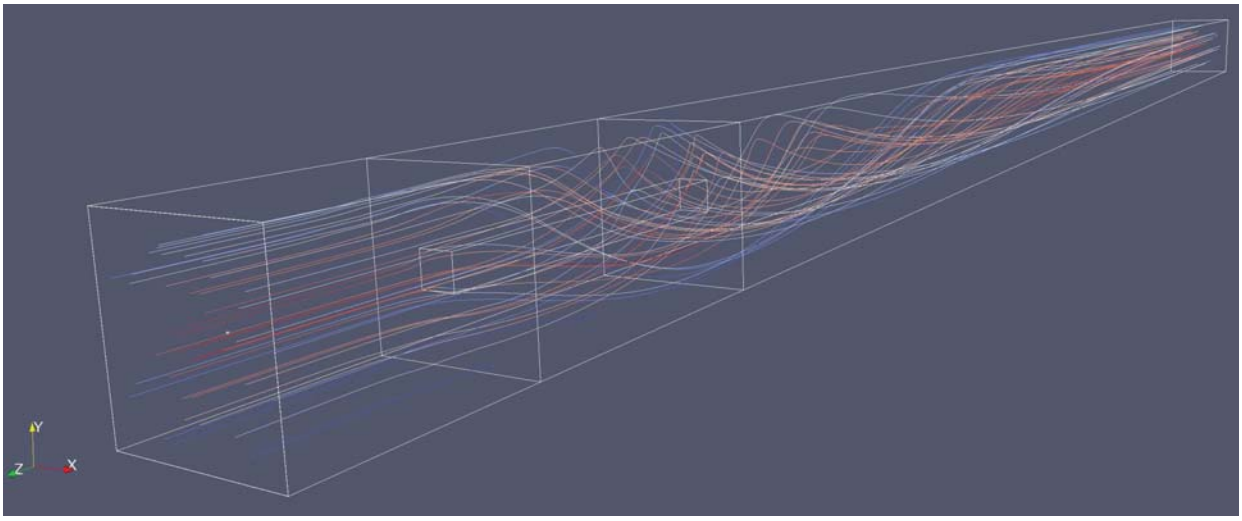
Velocity magnitude:

Velocity in x–direction, showing swirling:

Results: Wall shear stress

Shear stress at walls, computed with `wallShearStress`

Results: Streamlines



Heat flow in porous media

The Darcy equation

$$\vec{q} = -\frac{\bar{\kappa}}{\mu}\left(\nabla p + \rho\vec{g}\right)$$

The heat equation

$$\frac{\partial T}{\partial t} + \vec{q}\cdot\nabla T = \frac{k}{\rho_0 c}\nabla^2 T$$

The Boussinesq approximation

$$\rho = \rho_0\big(1 - \beta(T - T_0)\big)$$

Continuity of flow

$$\nabla\cdot\vec{q} = 0$$

Dimensionless variables

Dimensionless temperature is defined as

$$\theta = \frac{T - T_0}{T_1 - T_0}$$

the dimensionless pressure as

$$\phi = \frac{\rho_0 c \kappa}{\mu k} \left(p + \rho_0 g L z\right)$$

and the dimensionless time as

$$\tau = \left(\frac{k}{\rho_0 c L^2}\right) t$$

The Darcy-Lapwood system

By introduction the dimensionless field variables the heat equation becomes

$$\frac{\partial \theta}{\partial \tau} = \nabla \cdot \left(\left(\nabla \phi - \mathsf{Ra}\, \theta \vec{z}\right) \theta + \nabla \theta\right)$$

and the continuity requirement is then

$$\nabla \cdot \left(\nabla \phi - \mathsf{Ra}\, \theta \vec{z}\right) = 0$$

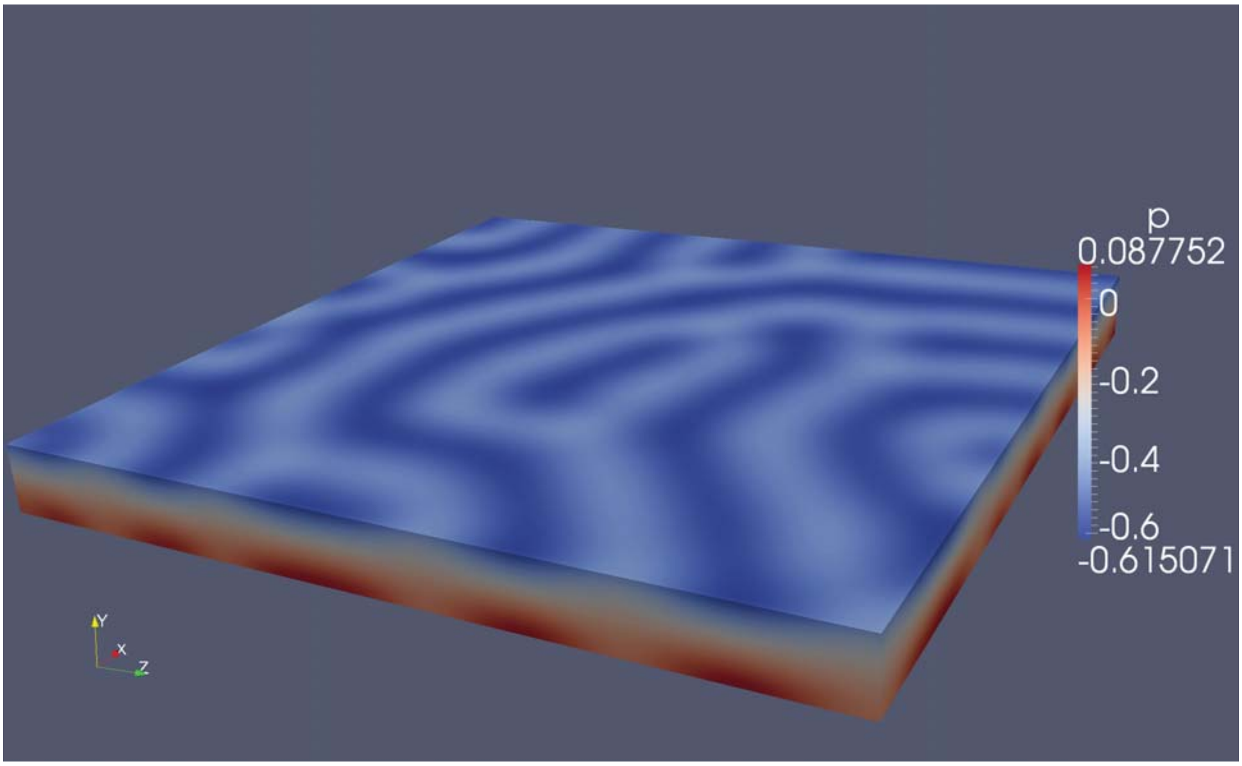
with the dimensionless porous Rayleigh number defined as

$$\mathsf{Ra} = \frac{\rho_0^2 c g \beta (T_1 - T_0) \kappa L}{\mu k}$$

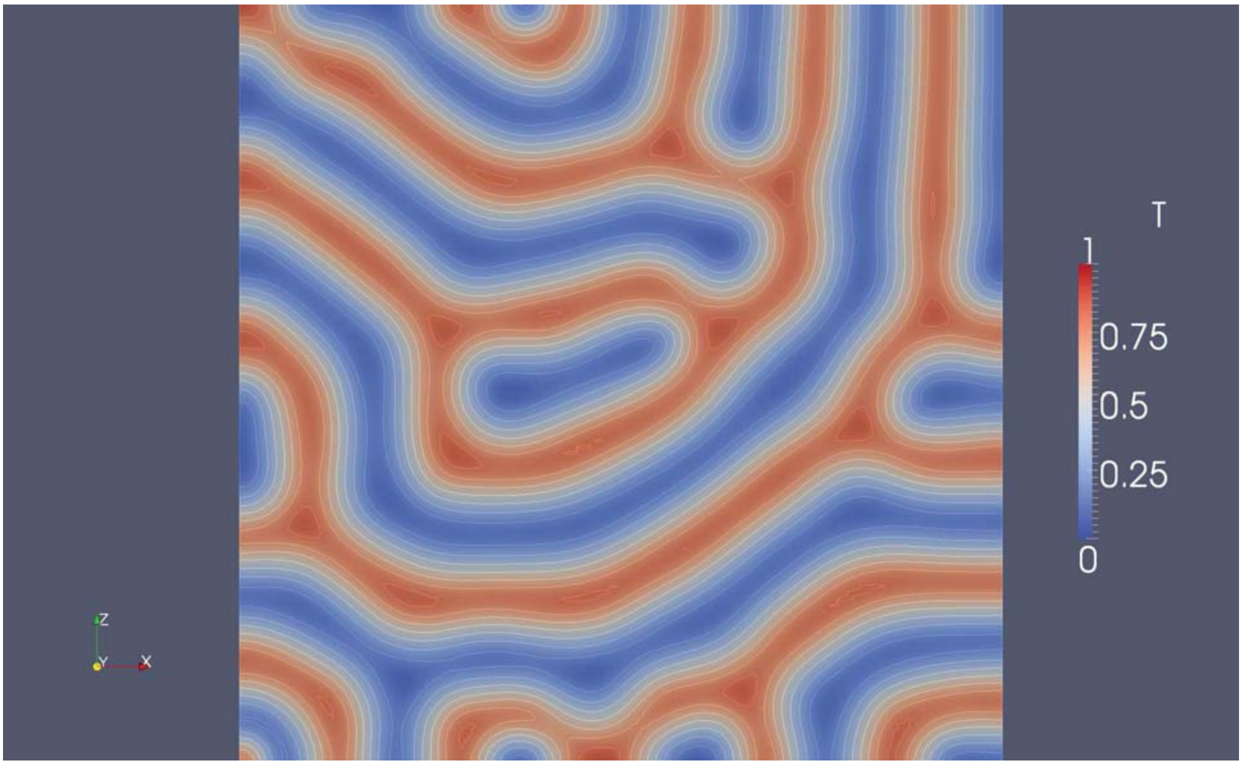
A customized solver code

```
while (runTime.loop())
{
    Info<< "Time = " << runTime.timeName() << nl << endl;

#   include "readPISOControls.H"
#   include "CourantNo.H"

    for (int nonOrth=0; nonOrth<=nNonOrthCorr; nonOrth++)
    {
        fvScalarMatrix pEqn
        (
            // Darcy equation for porous flow
            fvm::laplacian(kappa / nu, p) + kappa / nu * fvc::div(gflux, rhok)
        );

        // Set reference pressure and solve Darcy equation
        pEqn.setReference(pRefCell, pRefValue);
        pEqn.solve();

        // Update velocity field and flux
        U = -kappa / nu * (fvc::grad(p) + rhok * g);
        phi = fvc::interpolate(U) & mesh.Sf();

        solve
        (
            // Solve heat transport equation
            fvm::ddt(T) + fvm::div(phi, T) - fvm::laplacian(nu / Pr, T)
        );

        // Update kinematic density, based on Boussinesq approximation
        rhok = 1.0 - beta*(T - TRef);
    }

    runTime.write();
}
```

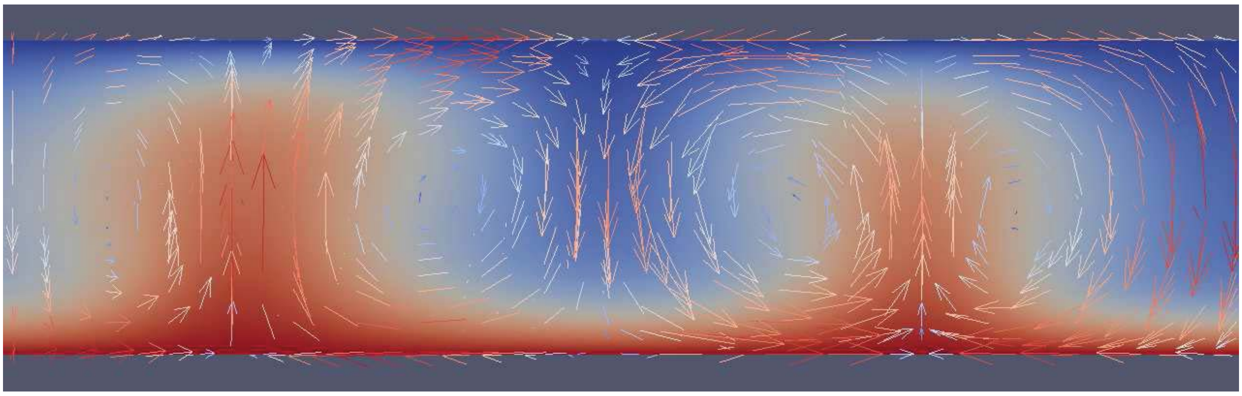

Results for Ra $= 100$, pressure



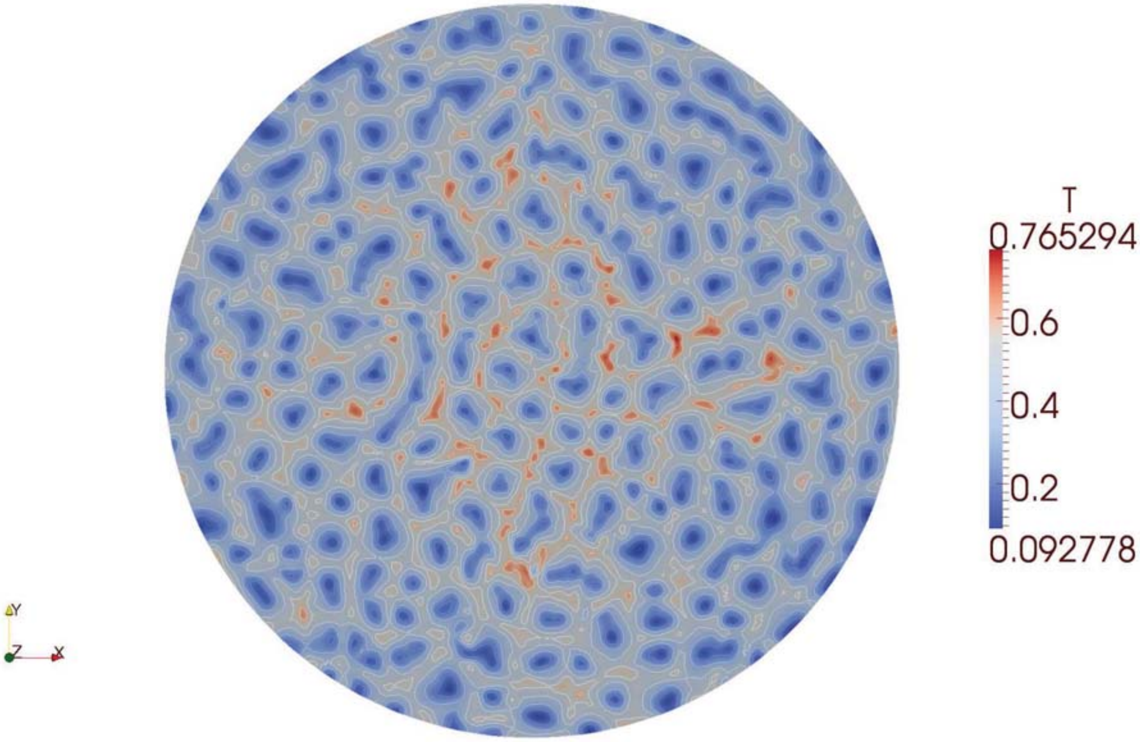
Results for Ra $= 100$, temperature



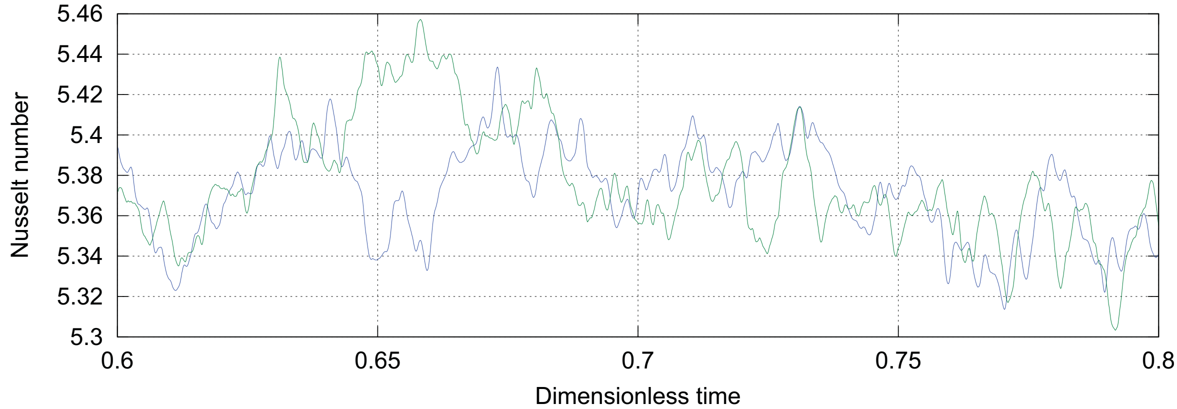
Results for Ra $= 100$, continued

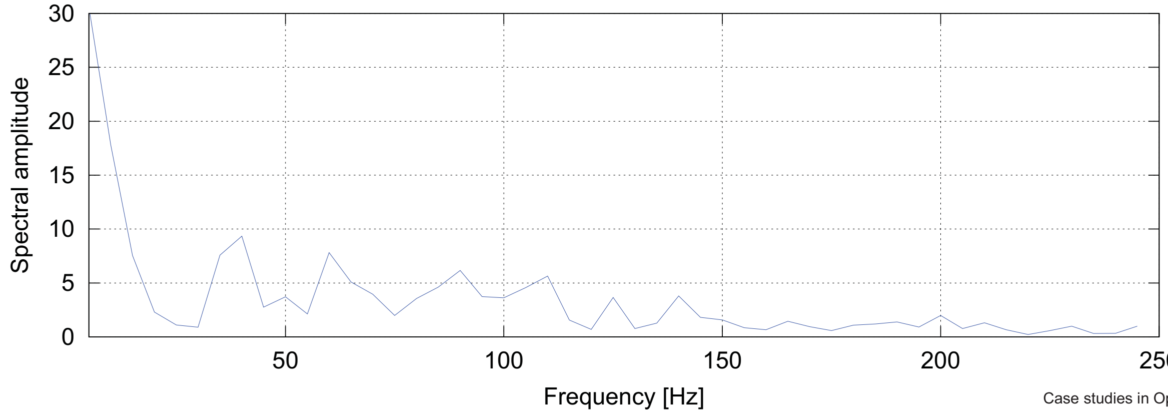


Results for Ra $= 500$, temperature



Dimensionless flux from top and bottom





Conclusion

- Both simple and complex geometries can be meshed, using the tools included
- When needed, customized programs can be written, but that can be quite demanding because of the complexity of the programming environment
- Standard flow problems, as well as customized problems can be solved with relative ease, using parallel processing when appropriate
- Visualization of results is performed in `paraview`, but further calculations can be performed with builtin libraries or custom programs
- Results compared to theory are generally accurate, but some care must be taken when calculating derivatives of fields

On preconditioning of matrices in two-by-two block form

Owe Axelsson

Institute of Geonics, AS CR, Ostrava, The Czech Republic

Department of Information Technology, Uppsala University, Sweden

Institute of Geonics, Ostrava, December 1-3, 2010

Institute of Geonics
Ostrava

This talk is based on the following three papers:

- Owe Axelsson: Preconditioners for regularized saddle point matrices; in preparation.
- Owe Axelsson, Maya Neytcheva: A general approach to analyze preconditioners for two-by-two block matrices; submitted to NLA.
- Owe Axelsson, Radim Blaheta: Preconditioning of matrices partitioned in 2×2 block form: eigenvalue estimates and Schwarz DD for mixed FEM, Numer. Linear Algebra Appl. 17(2010), pp. 787–810.

Plan of the talk:

- Introduction: Some examples of practical problems where saddle point matrices arise
- Preconditioners for saddle point problems and regularized saddle point problems
- A general approach to construct and control the accuracy of preconditioners for matrices in two-by-two block form
- Some numerical illustrations

Introduction

Matrices in 2×2 block form

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

arise naturally in several applications.

Examples:

- Flows in porous media, modelled by Darcy's equations,

$$\begin{aligned} K^{-1}\mathbf{u} + \nabla p &= 0 \quad \text{in } \Omega \\ \nabla \cdot \mathbf{u} &= f \quad \text{in } \Omega, \end{aligned}$$

  where $\mathbf{u} \cdot \mathbf{n} = q_1$ on $\partial\Omega_D$, $p = q_2$ on $\partial\Omega_N$.
- Mixed FEM for elliptic problems
- The elasticity equations for nearly incompressible materials can be formulated in a variational form (due to Hermann) as

$$\begin{aligned} \tfrac{1}{\nu} a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) &= \tfrac{1}{2\nu\mu} \int_\Gamma (\mathbf{g}, \mathbf{v}) d\Omega \quad \forall \mathbf{v} \in \mathbf{V}, \\ b(\mathbf{u}, q) - (1 - 2\nu)(p, q) &= 0 \quad \forall q \in \mathbf{H}. \end{aligned}$$

  Here $a(\mathbf{u}, \mathbf{v}) = \sum_{i,j} \int_\Omega \epsilon_{i,j}(\mathbf{u}) \epsilon_{i,j}(\mathbf{v}) d\Omega$ is a H^1-elliptic form (Korn's inequality) with strains $\epsilon_{i,j}$ and displacements $\mathbf{u}$, and $b(\mathbf{u}, p) = \int_\Omega div(\mathbf{u}) p d\Omega$, where p denotes the pressure function. The material parameter $\nu (\nu < 1/2)$, called Poisson ratio, becomes $\nu = 1/2$ for incompressible materials.

Saddle point matrices

The above problems can be formulated as constrained optimization problems and lead to saddle point matrices in the form

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & 0 \end{bmatrix} = \begin{bmatrix} M & B^T \\ B & 0 \end{bmatrix}$$

or some regularized form thereof.

Nonsymmetric saddle point problems

Nonsymmetric saddle point problems arise, e.g., in the numerical solution of steady Navier–Stokes equations which are often solved via a sequence of linearized problems, referred to as the Oseen problem:

$$\begin{aligned} -\nu\Delta\mathbf{u} + (\mathbf{w}\cdot\triangledown)\mathbf{u} + \triangledown p &= \mathbf{f} \quad \text{in } \Omega \\ \triangledown\cdot\mathbf{u} &= 0 \quad \text{in } \Omega \\ \mathbf{u} &= \mathbf{g} \quad \text{on } \partial\Omega \end{aligned}$$

Here $\nu > 0$ is the kinematic viscosity coefficient, Δ is the Laplace operator and $\mathbf{w}$ is the current approximation of the velocity vector $\mathbf{u}$; $\mathbf{f} : \Omega \to \Re^d$ is the given force field and $\mathbf{g} : \partial\Omega \to \Re^d$ the given boundary data.

The problem is to find the velocity field $\mathbf{u} : \Omega \to \Re^d$ and pressure variable $p : \Omega \to \Re$.

The FEM discretization leads here to a saddle point matrix with a non–selfadjoint operator M .

Preconditioners and the CBS constant

Block diagonal preconditioners are easy to handle but in general not sufficiently accurate. For instance, for an spd matrix for the generalized eigenvalue problem,

$$\lambda \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

it holds $$1-\gamma \le \lambda \le 1+\gamma,$$

where $\gamma = \rho(A_{11}^{-1}A_{12}A_{22}^{-1}A_{21})$, the so-called CBS constant, which can be seen to measure the relative strength of the off-diagonal blocks.
Here $\gamma < 1$ but in many ill-conditioned problems γ can take values very close to unit value which leads to large condition numbers $(1+\gamma)/(1-\gamma)$.

Similarly, for the Schur complement matrix $S_2 = A_{22} - A_{21}A_{11}^{-1}A_{12}$, it holds

$$(1-\gamma^2)A_{22} \le S_2 \le A_{22} \quad \text{(inequalities in a positive semidefinite sense).}$$

The above relations are of particular interest when a finite element mesh is separated in 'coarse' and 'fine' mesh nodes. Then, with the use of hierarchical basis functions, A_{22} becomes the coarse mesh matrix.

Preconditioners and the CBS constant, cont.

More accurate approximations can be constructed by use of an approximate block matrix factorization in the form

$$P = \begin{bmatrix} I_1 & 0 \\ A_{21}C_{11} & I_2 \end{bmatrix} \begin{bmatrix} \widetilde{A}_{11} & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I_1 & B_{11}A_{12} \\ 0 & I_2 \end{bmatrix}.$$

Here B_{11}, C_{11} are some sparse approximations of A_{11}^{-1} (possibly one of them is a zero block) and $\widetilde{A}_{11}^{-1}$ denotes an approximation, often only implicitly defined by use of inner iterations to solve the arising systems with matrix A_{11}. Finally, S denotes a nonsingular approximation of S_2.

To measure the accuracy of such approximations a more generally applicable measure (σ) than the CBS constant will be used. This measure can be controlled by the choice of the approximations B_{11}, C_{11}.

In the talk we present first some comprehensible results for clustering of eigenvalues for preconditioning of saddle point matrices and discuss then shortly some new results for the general matrix factorization method.

Efficient preconditioning for saddle point matrices

Given: a possibly nonsymmetric, real valued saddle point matrix in the form

$$\mathcal{M} = \begin{bmatrix} M & B^T \\ C & 0 \end{bmatrix},$$

where M has order $n \times n$ and B, C have orders $m \times n$, $m \leq n$.

Next:

- We give assumptions to enable to state that $\mathcal{M}$ is nonsingular
- If they do not hold, we consider a regularization of $\mathcal{M}$ to make the regularized matrix nonsingular.

For both cases we present efficient preconditioners for which a strong clustering of the eigenvalues of the preconditioned matrix take place.

Preconditioning for a nonsingular saddle point matrix

The construction of the preconditioner is based on the following assumption.

Let $\mathcal{R}(A)$ denote the range of an operator (A).

Assumption 1

(i) There exists a nonsingular matrix W (possibly $W = I$) such that $\widetilde{M} = M + B^T W^{-1} C$ is nonsingular. Note that this implies in particular that $\mathcal{N}(C) \cap \mathcal{N}(M) = \emptyset, \mathcal{N}(B) \cap \mathcal{N}(M^T) = \emptyset$.

(ii) The intersection of $\mathcal{N}(C)$ with the one-to-one transformation of $\mathcal{R}(B^T)$ with $\widetilde{M}^{-1}$, denoted $\widetilde{\mathcal{R}}(B^T)$, includes only the trivial vector, that is, $\mathcal{N}(C) \cap \widetilde{\mathcal{R}}(B^T) = \emptyset$.

(iii) Matrix B has full rank. Note that if $C = B$, then this condition implies $\mathcal{N}(B) \cap \mathcal{R}(B^T) = \emptyset$, i.e. condition (ii) holds.

Preconditioning for a nonsingular saddle point matrix, cont.

Lemma 1 *Under Assumption 1 it follows that $\mathcal{M}$ is nonsingular.*

Proof The homogeneous system

$$\begin{bmatrix} M & B^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{1}$$

implies that $Mx + B^T y = 0$ and $Cx = 0$. Then $B^T W^{-1} Cx = 0$ so $(M + B^T W^{-1} C)x + B^T y = \widetilde{M}x + B^T y = 0$, or $x = -\widetilde{M}^{-1} B^T y$. But then x belongs to a one-to-one transformation of $\mathcal{R}(B^T)$ and, since $x \in \mathcal{N}(C)$, by Assumption 1, it follows that $x = 0$. Then $B^T y = 0$ so, <u>since B has full rank</u>, it follows that $y = 0$. Hence the homogeneous system (1) has only the trivial solution. ■

Preconditioning for a nonsingular saddle point matrix, cont.

For the nonsingular matrix $\mathcal{M}$ it turns out that the preconditioner $\begin{bmatrix} M & \alpha B^T \\ 0 & -W \end{bmatrix}, \alpha = 2,$ can be very efficient in clustering the eigenvalues of the preconditioned matrix.

The corresponding generalized eigenvalue problem,

$$\lambda \begin{bmatrix} M & 2B^T \\ 0 & -W \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} M & B^T \\ C & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2}$$

implies $\lambda y = -W^{-1}Cx$. Since both matrices in (2) are nonsingular, it follows that $\lambda \neq 0$. Hence $y = -\lambda^{-1}W^{-1}Cx$ and from the first equation in (2) it follows that $(\lambda - 1)Mx = (-\lambda + 2 - \frac{1}{\lambda})B^TW^{-1}Cx$, or

$$\lambda(\lambda - 1)Mx = -(\lambda - 1)^2 B^TW^{-1}Cx. \tag{3}$$

It follows that if $x \in \mathcal{N}(C)$ or $x \in \mathcal{N}(B)$, then $\lambda = 1$.

Preconditioning for a nonsingular saddle point matrix, cont.

Likewise, if $x \in \mathcal{N}(M)$ (i.e., then $x \notin \mathcal{N}(C)$), then $\lambda = 1$. The dimension of $\mathcal{N}(C)$ is at least $n - m$. Assume that the dimension of $\mathcal{N}(M)$ equals ν (it is seen from assumptions made that $\nu < m$). Then the multiplicity of eigenvalues $\lambda = 1$ equals $n - m + \nu$.

If $x \notin \mathcal{N}(M) \cup \mathcal{N}(B) \cup \mathcal{N}(C)$, then $\lambda \neq 1$ and it follows from (3) that $\lambda Mx = -(\lambda - 1)B^TW^{-1}Cx$, or

$$\lambda \widetilde{M}x = B^TW^{-1}Cx, \; x \neq 0, \tag{4}$$

where we recall that $\widetilde{M} = M + B^TW^{-1}C$.

We will use a matrix W involving a parameter r in the form $\frac{1}{r}W$, where $r > 0$ and W is nonsingular. Then (4) takes the form

$$\lambda(M + rB^TW^{-1}C)x = rB^TW^{-1}Cx, \; x \neq 0. \tag{5}$$

The above implies that for eigenvalues $\lambda \neq 1$, it holds $\lambda = \lambda_r \to 1$ as $r \to \infty$.

This shows that all eigenvalues cluster at the unit value as $r \to \infty$.

Preconditioning for a nonsingular saddle point matrix, cont.

We collect the results in a theorem.

Theorem 1 *Let Assumption 1 hold. Then the preconditioned matrix*

$$\begin{bmatrix} M & 2B^T \\ 0 & -W_r \end{bmatrix}^{-1} \begin{bmatrix} M & B^T \\ C & 0 \end{bmatrix}, \text{ where } W_r = \frac{1}{r}W,$$

has eigenvalues $\lambda = 1$ of multiplicity at least $n - m + \nu$, where ν $(\nu \leq m)$ is the dimension of $\mathcal{N}(M)$. The remaining eigenvalues satisfy $\lambda = \lambda_r \to 1$ as $r \to \infty$.

If $C = B$ and $M = M^T$, then

$$\frac{1}{1 + \mu_1/r} \leq \lambda \leq \frac{1}{1 - \mu_0/r},$$

where μ_0, μ_1 are the extreme eigenvalues of

$$Mx = \mu B^T W^{-1} Bx, \quad x \notin \mathcal{N}(B).$$

Preconditioning for a regularized saddle point matrix

The matrix $M_r = M + rB^T W^{-1} C$ can be said to be a regularized form of M. The corresponding regularized saddle point matrix takes the form

$$\widetilde{\mathcal{M}}_r = \begin{bmatrix} M_r & B^T \\ C & 0 \end{bmatrix}.$$

Note that the regularized and unregularized systems,

$$\widetilde{\mathcal{M}}_r \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} \quad \text{and} \quad \mathcal{M} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}$$

have the same (unique if $\mathcal{M}$ is nonsingular) solution. However, depending on the choice of W, applying actions of $\widetilde{\mathcal{M}}_r$ can be costly.

Preconditioning for a regularized saddle point matrix, cont.

Instead we shall only use such a regularized form, where M is replaced by M_{r_0}, in the preconditioner to $\mathcal{M}$. Here $M_{r_0} = M + r_0 B^T W^{-1} C$, and $r_0 < r$, where r is another method parameter.
In addition, the above regularization does not handle the case where $\mathcal{M}$ is singular, due to a rank deficient matrix B. We replace then the zero (2,2) block in the matrix, and in its preconditioner with $-W_r = -\frac{1}{r}W$.

This means that the matrix $\mathcal{M}$ is perturbed with the matrix $\begin{bmatrix} 0 & 0 \\ 0 & -W_r \end{bmatrix}$. We assume that r takes sufficiently large values so that the corresponding perturbation of the solution is negligible or, otherwise, we can use some steps of a defect-correction method to correct for this perturbation.

From the corresponding block matrix factorization,

$$\mathcal{M}_r = \begin{bmatrix} M & B^T \\ C & -W_r \end{bmatrix} = \begin{bmatrix} M_r & B^T \\ 0 & -\frac{1}{r}W \end{bmatrix} \begin{bmatrix} I_1 & 0 \\ -rW^{-1}C & I_2 \end{bmatrix},$$

it follows that $\mathcal{M}_r$ is nonsingular if and only if $M_r = M + rB^T W^{-1} C$ is nonsingular.

Preconditioning for a regularized saddle point matrix, cont.

We make now the following assumption.

Assumption 2 *The matrix $M_r = M + rB^T W^{-1} C$ is nonsingular for all $r \geq r_0$ for some $r_0 > 0$.*

The preconditioner to $\mathcal{M}_r$ will be taken as $\begin{bmatrix} M_{r_0} & B^T \\ 0 & -W_r \end{bmatrix}$, where $W_r = \frac{1}{r}W$ and the corresponding generalized eigenvalue problem takes the form

$$\lambda \begin{bmatrix} M_{r_0} & B^T \\ 0 & -W_r \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} M & B^T \\ C & -W_r \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

or

$$(\lambda - 1) \begin{bmatrix} M_{r_0} & B^T \\ 0 & -W_r \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -r_0 B^T W^{-1} C x \\ C x \end{bmatrix}.$$

It follows that $\lambda = 1$ for any eigenvector $\begin{bmatrix} x \\ y \end{bmatrix}$ in the form $x \in \mathcal{N}(C), y \in \Re^m$.

Preconditioning for a regularized saddle point matrix, cont.

We collect the results in the next theorem.

Theorem 2 *Assume that $M + rB^TW^{-1}C$ is nonsingular for all $r \geq r_0, r_0 > 0$. Then the preconditioned matrix*

$$\begin{bmatrix} M_{r_0} & B^T \\ 0 & -W_r \end{bmatrix}^{-1} \begin{bmatrix} M & B^T \\ C & -W_r \end{bmatrix}$$

has eigenvalues $\lambda = 1$ of multiplicity at least $n - m$. As $r_0 \to \infty$, the remaining eigenvalues cluster about the point r/r_0.
($W_r = 1/rW$))

Two-by-two block matrices

We present now a general algebraic approach to construct, analyze and control the accuracy of preconditioners for matrices in two-by-two block form.
This includes symmetric and nonsymmetric matrices, as well as indefinite matrices.
The general form of a matrix in two-by-two block form is:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

We assume here that both
- A_{11}
- $S_2(A) = A_{22} - A_{21}A_{11}^{-1}A_{12}$ (the Schur complement)

are nonsingular, which implies that A is also nonsingular.

General form: approximate block matrix factorization

A general form of an approximate block factorization of the matrix A takes the form

$$P = \begin{bmatrix} I_1 & 0 \\ A_{21}C_{11} & I_2 \end{bmatrix} \begin{bmatrix} \widetilde{A}_{11} & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I_1 & B_{11}A_{12} \\ 0 & I_2 \end{bmatrix}.$$

- B_{11} and C_{11} are approximations of A_{11}^{-1} (possibly zero matrices but normally sparse and given on explicit form)
- $\widetilde{A}_{11}^{-1}$ denotes some approximation of A_{11}^{-1}, often only implicitly defined via inner iterations
- S is a nonsingular approximation of S_2.

When we use inner iterations, the preconditioner at each outer iteration step becomes variable. In such a case, the outer iteration method must, in general, be some form of a generalized conjugate gradient method, such as GCG, GMRES or the modified Least Squares GMRES method.

$$P = \begin{bmatrix} I_1 & 0 \\ A_{21}C_{11} & I_2 \end{bmatrix} \begin{bmatrix} \widetilde{A}_{11} & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I_1 & B_{11}A_{12} \\ 0 & I_2 \end{bmatrix}$$

Further, we consider the special but important case when only one of B_{11} and C_{11} equals zero. In this case, P is block-triangular and if, say, $B_{11} = 0$, then P takes the form

$$P = \begin{bmatrix} \widetilde{A}_{11} & 0 \\ A_{21} & S \end{bmatrix}.$$

This corresponds to the choice $C_{11} = \widetilde{A}_{11}^{-1}$ and $B_{11} = 0$ above. Note that the computational expense of this preconditioner is essentially the same as for the block-diagonal preconditioner but, as we shall see, it can be much more efficient.

The rate of convergence of preconditioned conjugate gradient methods depends on the distribution of eigenvalues of $P^{-1}A$, which we now estimate.
In the analysis, we introduce a scalar

$$\sigma$$

which plays a role similar to the CBS constant γ,or rather to

$$\gamma^2/(1-\gamma^2)$$

but is of relevance not only for symmetric and positive definite matrices. As we shall see, $\sigma = 1$ for indefinite problems on saddle point form, but for other types of problems it is important that C_{11} (and/or B_{11}) is a sufficiently accurate preconditioner to limit the upper bound of σ to a viable value.

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}; P = \begin{bmatrix} I_1 & 0 \\ A_{21}C_{11} & I_2 \end{bmatrix} \begin{bmatrix} \widetilde{A}_{11} & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I_1 & B_{11}A_{12} \\ 0 & I_2 \end{bmatrix}$$

Proposition 1 *Let A, C be defined as above. Then $P^{-1}A$ is similarly equivalent to (i.e. the eigenvalues of $P^{-1}A$ equal those of) the matrix*

$$\left(\begin{bmatrix} I_1 & 0 \\ 0 & I_2 \end{bmatrix} + \begin{bmatrix} (\widetilde{A}_{11}^{-1}A_{11} - I_1) & 0 \\ 0 & (S^{-1}S_2 - I_2) \end{bmatrix} \right) \times$$
$$\left(\begin{bmatrix} I_1 & 0 \\ 0 & I_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ \widetilde{A}_{21} & 0 \end{bmatrix} \right) \left(\begin{bmatrix} I_1 & 0 \\ 0 & I_2 \end{bmatrix} + \begin{bmatrix} 0 & \widetilde{A}_{12} \\ 0 & 0 \end{bmatrix} \right)$$

where $\widetilde{A}_{12} = (I_1 - B_{11}A_{11})A_{11}^{-1}A_{12}$, $\widetilde{A}_{21} = S_2^{-1}A_{21}(I_1 - C_{11}A_{11})$.

Proof Use the similarity transformation $\begin{bmatrix} I_1 & B_{11}A_{11} \\ 0 & I_2 \end{bmatrix} P^{-1}A \begin{bmatrix} I_1 & -B_{11}A_{11} \\ 0 & I_2 \end{bmatrix}$. ■

The limit case: $\widetilde{A}_{11}^{-1} = A_{11}^{-1}$, $S = S_2$

Note that in this case we do not need to assume that $I + G$ is diagonalizable.

Proposition 2 *Consider P with $\widetilde{A}_{11} = A_{11}$ and $S = S_2$. Then, for the generalized eigenvalue problem $A\mathbf{x} = \lambda P\mathbf{x}$ there is a multiple eigenvalue $\lambda = 1$ for eigenvectors $x = [x_1, x_2]^T$, where*

$$x_1 \in ker(A_{21}(I_1 - C_{11}A_{11})),\ x_2 \in ker((I_1 - B_{11}A_{11})A_{11}^{-1}A_{12})\,.$$

The remaining eigenvalues equal $\lambda = 1 + \frac{1}{2}\zeta(1 \pm \sqrt{1+4/\zeta})$ for ζ being any nonzero eigenvalue of the matrix product $\widetilde{A}_{12}\widetilde{A}_{21}$, where $\widetilde{A}_{12} = (I_1 - B_{11}A_{11})A_{11}^{-1}A_{12}$ and $\widetilde{A}_{21} = S_2^{-1}A_{21}(I_1 - C_{11}A_{11})$.

The limit case: $\widetilde{A}_{11}^{-1} = A_{11}^{-1}$, $S = S_2$

Proposition 2 (cont.)

If $C_{11} = B_{11}$, A is symmetric and S_2 is positive definite, then the eigenvalues ζ are real and positive, and we obtain

$$\lambda_{\max} = 1 + \frac{1}{2}\sigma(1 + \sqrt{1+4/\sigma})\,,\ \lambda_{\min} = \frac{\sqrt{1+4/\sigma}-1}{\sqrt{1+4/\sigma}+1},$$

where $\sigma = \rho(\widetilde{A}_{12}\widetilde{A}_{21})$ and $0 < \zeta \leq \sigma$. In the general case of nonsymmetric matrices, letting $\sigma = \|\widetilde{A}_{12}\widetilde{A}_{21}\|$, then for the absolute values of the eigenvalues it holds that $\lambda_{\min} \leq |\ \lambda\ | \leq \lambda_{\max}$.

The value of σ can be large if B_{11} and C_{11} are less accurate approximations of A_{11}^{-1}.

When A_{22} is nonsingular, a more visible connection between the measures $\boxed{\gamma = \rho(A_{11}^{-1}A_{12}A_{22}^{-1}A_{21})}$ and $\boxed{\sigma = \rho((I_1 - B_{11}A_{11})A_{11}^{-1}A_{12}S_2^{-1}A_{21}(I_1 - C_{11}A_{11}))}$ can be established. Using the Sherman-Morrison-Woodbury formula for the matrix product $Q = A_{11}^{-1}A_{12}S_2^{-1}A_{21}$, which is a factor in σ, it holds

$$
\begin{aligned}
Q &= A_{11}^{-1}A_{12}\left(A_{22} - A_{21}A_{11}^{-1}A_{12}\right)^{-1}A_{21} \\
&= A_{11}^{-1}A_{12}\left[A_{22}^{-1} + A_{22}^{-1}A_{21}\left(A_{11} - A_{12}A_{22}^{-1}A_{21}\right)^{-1}A_{12}A_{22}^{-1}\right]A_{21} \\
&= A_{11}^{-1}A_{12}A_{22}^{-1}A_{21} + A_{11}^{-1}A_{12}A_{22}^{-1}A_{21}(I_1 - A_{11}^{-1}A_{12}A_{22}^{-1}A_{21})^{-1}A_{11}^{-1}A_{12}A_{22}^{-1}A_{21} \\
&= \Gamma + \Gamma(I_1 - \Gamma)^{-1}\Gamma \\
&= \Gamma(I_1 - \Gamma)^{-1},
\end{aligned}
$$

where $\Gamma = A_{11}^{-1}A_{12}A_{22}^{-1}A_{21}$.

Proposition 3 *Assume that A is symmetric and positive definite, $C_{11} = B_{11}$, $\widetilde{A}_{11} = A_{11}$ and $S = S_2$. Let $\sigma = \rho(\widetilde{A}_{12}\widetilde{A}_{21})$, where $\widetilde{A}_{12}$, $\widetilde{A}_{21}$ are defined in Proposition 1. Then*

$$\sigma \leq \frac{\gamma^2}{1-\gamma^2}\|I_1 - B_{11}A_{11}\|^2. \tag{6}$$

Block-diagonal preconditioners for saddle point matrices

The block-diagonal preconditioner $P = \begin{bmatrix} A_{11} & 0 \\ 0 & S_2 \end{bmatrix}$, however, is directly applicable for saddle point problems, where $A_{22} = 0$. In this case $S_2 = -A_{21}A_{11}^{-1}A_{12}$ and

$$\sigma = \mp\rho(A_{11}^{-1}A_{12}S^{-1}A_{21}) = \begin{cases} -1 & \text{if} \quad S = S_2 \\ +1 & \text{if} \quad S = -S_2 \end{cases}$$

This implies the next proposition.

Proposition 4 *Let* $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & 0 \end{bmatrix}$ *be symmetric, where* A_{11} *and* $S_2 = -A_{21}A_{11}^{-1}A_{12}$ *are nonsingular (i.e.,* A_{21} *has full rank). Then the preconditioned matrix* $P^{-1}A$*, where* $P = \begin{bmatrix} A_{11} & 0 \\ 0 & S \end{bmatrix}$*, has eigenvalues* $\lambda = \begin{cases} 1 \pm \sqrt{5}, & \text{if} \quad S = -S_2 \\ 1 \pm i\sqrt{3}, & \text{if} \quad S = S_2 \end{cases}$
There are only three eigenvalues of $P^{-1}A$*, the unit value plus either of the two eigenvalue pairs given above.*

Block-triangular preconditioners

Clearly, the optimal balance to get the smallest total computational cost, including the costs for the inner iterations and for the Schur complement matrix, is problem dependent, i.e., must be analyzed for each separate type of problem.

The most efficient form of the preconditioner is in general of block-tridiagonal form. If we let $C_{11} = \widetilde{A}_{11}^{-1}$ and, for simplicity, let $B_{11} = 0$ then it takes the form

$$P = \begin{bmatrix} \widetilde{A}_{11}^{-1} & 0 \\ A_{21} & S_2 \end{bmatrix}$$

and

$$\sigma = \|A_{11}^{-1}A_{12}S_2^{-1}A_{21}(I_1 - \widetilde{A}_{11}^{-1}A_{11})\|.$$

Here we can control the value of σ, and make it arbitrarily small, by making a sufficient number of inner iterations in solving the arising systems with matrix A_{11}.

Approximations of Schur complement matrices

For ill–conditioned problems, for example when the LBB – stability condition is violated, one can use some form of regularization of the problem. This is similar to the use of a penalized or augmented Lagrangian method for general constrained optimization problems. For instance, if the given saddle point matrix has the form $\mathcal{A} = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix}$,

then the matrix for the regularized problem (which actually has the same solution) has the form

$$\mathcal{A}_r = \begin{bmatrix} A + rB^T B & B^T \\ B & 0 \end{bmatrix} ,$$

where r can be a large penalization parameter.

As has been shown in an earlier works, here the Schur complement matrix approaches the value $\frac{1}{r} I_2$, where I_2 is an identity matrix, so there is no need to precondition the Schur complements in the outer iteration method. In the ideal case, using a block-diagonal preconditioner, there will only be three to four outer iterations for large values of the parameter r.

How to solve with the modified pivot block?

However, here the difficulty is left to the solution of the pivot block matrix.
This can be handled by some form of a domain decomposition and Schwarz alternating iteration method, for instance.

The major intention of this presentation is to introduce the eigenvalue analysis and the handling of various arising matrices, using methods such as Schwarz alternating iteration method will therefore not be taken up further here.

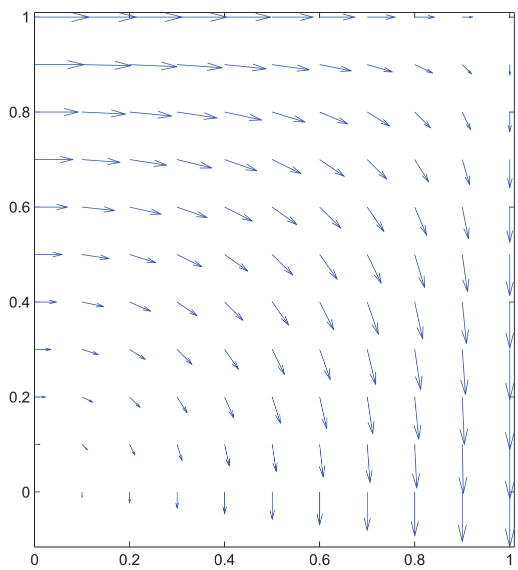
Numerical illustrations

Problem 1 (Convection-diffusion problem) *Find u satisfying the equation*

$$\begin{aligned} -\varepsilon\Delta u + (\mathbf{b}\cdot\nabla)u &= f(x,y) \text{ in } \Omega \\ u(x,y) = g(x,y) \text{ on } \Gamma_D, \frac{\partial u}{\partial n} &= 0 \text{ on } \Gamma_N, \end{aligned}$$

where $\Omega = [0,1]^2$, $\Gamma_D \cup \Gamma_N = \partial\Omega$ and $\Gamma_D \cap \Gamma_N = \emptyset$ and $0 < \varepsilon \leq 1$. We choose $\mathbf{b} = \begin{bmatrix} 2y(1-x^2) \\ -2x(1-y^2) \end{bmatrix}$, which represents a quarter of a vortex flow, centered at the origin, visualized in the following figure. The boundary conditions are of inhomogeneous Dirichlet type on $x = 0$, $x = 1$, $y = 1$ and of homogeneous Neumann type on $y = 0$. The problem is discretized using piece-wise linear conforming finite elements. (Within this setting we do not consider very strongly convection dominated problems.)

Numerical illustrations: test problem 1



The vector $\mathbf{b}$ in Problem 1

Numerical illustrations: test problem 2

Problem 2 (Moving interface problem) *Simulation of a moving interface with a constant speed by using the Cahn-Hilliard equation, written in the form of a coupled system of two partial differential equations:*

$$\begin{aligned} \eta - \Psi'(C) + \alpha \Delta C = 0, \qquad & x \in \Omega, \qquad t > 0 \\ -\beta \Delta \eta + \frac{\partial C}{\partial t} + (\mathbf{b} \cdot \nabla) C = 0, \qquad & x \in \Omega, \qquad t > 0 \\ \frac{\partial C}{\partial \mathbf{n}} = 0, \quad \frac{\partial \eta}{\partial \mathbf{n}} = 0, \qquad & x \in \partial\Omega, \qquad t > 0, \\ C(x,0) = C_0(x) \qquad & x \in \Omega. \end{aligned} \tag{7}$$

Here the unknown function C (the concentration) is a continuous scalar variable that describes the diffusive interface profile. It has constant value in each phase, ± 1, and changes rapidly but in a continuous manner from one to the other in an interface strip of certain thickness. The function Ψ is a double-well function with two minima at ± 1, corresponding to the two stable phases. The variable $\eta = \Psi'(C) - \alpha \Delta C$ is the so-called chemical potential. α, β are constant and positive problem parameters.

Numerical illustrations: test problem 2

For the particular test problem we use $\Psi(C) = \frac{1}{4}(C+1)^2(C-1)^2$. The domain of definition is $\Omega = [-1,1] \times [0,1])$ and the initial position of the front is at $x = 0$. The velocity vector $\mathbf{b} = [1, 0]$, i.e., the front is moving to the right with time.

The problem is discretized in time using a backward Euler implicit time-stepping method and in space - by linear FEM for both variables on a triangular grid. As is seen from (7), the problem is nonlinear and within each time step we use Newton's method to solve it.

Here we are interested primarily in the solution of the corresponding Jacobian matrix equation. The linear system to be solved during each nonlinear iteration has the following form.

$$\begin{bmatrix} M & -J - \alpha K \\ \beta \Delta t_k K & M + \Delta t_k W \end{bmatrix} \begin{bmatrix} \eta \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{f} \end{bmatrix} \tag{8}$$

where J is the part of the Jacobian, which corresponds to the nonlinear term $\Psi'(C)$. Here, K, M and W are the stiffness, mass and convection matrices, respectively.

There are several problem parameters involved in (7). For the numerical experiments we have used $\beta = 1/Pe$, where $Pe = 300$ is the Peclet number, α is the square of the so-called Cahn number, chosen in this case as 0.1.

Problem 1: we impose a 2×2 block structure by considering two consecutive (nested) refinements of the computational domain.

We consider some given mesh, referred to as 'coarse' and perform one regular refinement (in this case, into four congruent triangles), to obtain a 'fine' mesh. In this way, the degrees of freedom on the fine mesh is split into two non-intersecting classes, imposing the desired block two-by-two structure of the matrix on the fine level. We note, that the finite elements on the coarse mesh can be seen as macro-elements on the fine mesh and, by the same ordering, the corresponding macro-element stiffness matrix is also of block two-by-two form.

All numerical experiments are performed in Matlab. The chosen size of the test matrices is not very large since the theoretical bounds, to be illustrated, involve matrix functions, which are costly to compute exactly, such as the value of γ computed as $\rho(A_{11}^{-1}A_{12}A_{22}^{-1}A_{21})$, and the value of σ, computed as $\rho(\widetilde{A}_{12}\widetilde{A}_{21})$ or $\|\widetilde{A}_{12}\widetilde{A}_{21}\|$.

We test the effect of three different approximations of A_{11}^{-1}, namely, $C_{11}^{(1)}$, $C_{11}^{(2)}$ and $C_{11}^{(3)}$. The first two are computed as $C_{11}^{(i)} = (L^{(i)}U^{(i)})^{-1}$, where $L^{(i)}U^{(i)}$, $i = 1, 2$ is an incomplete factorization of A_{11} with a drop tolerance $\tau_1 = 0.01$, $\tau_2 = 0.001$ and $\tau_3 = 0.0001$. The LU-factors are computed using the Matlab function `luinc(`A_{11}, τ_i`)` or `cholinc` for spd matrices.

The matrix $C_{11}^{(3)}$ is constructed only in the setting of Problem 1 as a sparse approximate inverse of A_{11} as follows. First we compute an element-by-element approximation of A_{11}^{-1} as

$$C_{11}^{(3,1)} = \sum_{\ell=1}^{M} \left(A_{11,E}^{(\ell)}\right)^{-1}.$$

Clearly, $C_{11}^{(3,1)}$ is sparse and cheap to obtain. In order to improve the quality of $C_{11}^{(3,1)}$ as an approximation of A_{11}^{-1}, we compute a sparse additive correction, $C_{11}^{(3,2)}$, to it, such that the following Frobenius norm is minimized

$$\|I_1 - (C_{11}^{(3,1)} + C_{11}^{(3,2)})A_{11}\|_{A_{11}^{-1}} \tag{9}$$

Then we let $C_{11}^{(3)} = C_{11}^{(3,1)} + C_{11}^{(3,2)}$. Here, $C_{11}^{(3,1)}$ has the sparsity pattern of A_{11} and $C_{11}^{(3,2)}$ has the sparsity pattern of the error matrix $I_1 - C_{11}^{(3,1)}A_{11}$.

The matrix B_{11} is chosen to be either zero or equal to C_{11}, as indicated in the tables.

Size	γ	σ	δ	$eig(P^{-1}A)$		$cond(P^{-1}A)$	$\|I_1 - C_{11}A_{11}\|$
				λ_{min}	λ_{max}		
				$C_{11}^{(3)}$			
1089	0.9969	4.3717	4.6552	0.1610	6.211	38.573	0.1714
4225	0.9992	18.956	20.003	0.0478	20.91	437.45	0.1745
				$C_{11}^{(1)} = chol(A_{11}, 0.01)$			
1089	0.9969	0.1353	0.2639	0.6937	1.442	2.0784	0.0408
4225	0.9992	0.6183	1.1652	0.4642	2.154	4.6402	0.0421
				$C_{11}^{(2)} = chol(A_{11}, 0.001)$			
1089	0.9969	0.0023	0.0049	0.9530	1.049	1.1010	0.0056
4225	0.9992	0.0106	0.0218	0.9024	1.108	1.2282	0.0058
				$C_{11}^{(2)} = chol(A_{11}, 0.0001)$			
1089	0.9969	1.64e-5	5.21e-5	0.9960	1.0041	1.008	0.0006
4225	0.9992	7.82e-5	0.00023	0.9912	1.0089	1.018	0.0006

Problem 1, $\mathbf{b} = \mathbf{0}$, $\varepsilon = 1$: $\widetilde{A}_{11} = A_{11}$, $S = S_2$, $B_{11} = C_{11}$

$\delta = \frac{\gamma^2}{1-\gamma^2}\|I - B_{11}A_{11}\|\|I - C_{11}A_{11}\|$, $\gamma^2 = \rho(A_{11}^{-1}A_{12}A_{22}^{-1}A_{21})$

Size	γ	σ	δ	$eig(P^{-1}A)$ λ_{min}^{est}	λ_{min}	λ_{max}	λ_{max}^{est}	$\lVert I_1 - C_{11}A_1$
				$C_{11}^{(3)}, B_{11} = 0$				
1089	0.9944	20.517	27.872	0.0445	0.0445	22.472	22.472	0.1570
4225	0.9987	88.397	119.39	0.0111	0.0111	90.386	90.386	0.1601
				$C_{11}^{(1)} = chol(A_{11}, 0.01), B_{11} = 0$				
1089	0.9944	3.8784	7.233	0.1753	0.1753	5.7031	5.7031	0.04075
4225	0.9987	17.357	31.41	0.0518	0.0518	19.305	19.305	0.04210
				$C_{11}^{(2)} = chol(A_{11}, 0.001), B_{11} = 0$				
1089	0.9944	0.4263	0.9326	0.5263	0.5263	1.8999	1.8999	0.00525
4225	0.9987	1.9173	4.0879	0.2745	0.2745	3.6428	3.6428	0.00548
				$C_{11}^{(2)} = chol(A_{11}, 0.0001), B_{11} = 0$				
1089	0.9944	0.0399	0.1078	0.8193	0.8193	1.2206	1.2206	0.00061
4225	0.9987	1.9173	4.0879	0.2745	0.2745	3.6428	3.6428	0.00548
				$C_{11}^{(2)} = chol(A_{11}, 0.0001), B_{11} = C_{11}$				
1089	0.9944	2.07e-5	6.548e-5	0.9955	0.9955	1.0046	1.0046	0.00061
4225	0.9987	8.53e-5	0.0003	0.9908	0.9908	1.0093	1.0093	0.00060

Problem 1, $\mathbf{b} = [1, 0]$, $\varepsilon = 1$: $\widetilde{A}_{11} = A_{11}$, $S = S_2$,
$\delta = \frac{\gamma^2}{1-\gamma^2} \|I - B_{11}A_{11}\| \|I - C_{11}A_{11}\|$, $\gamma^2 = \rho(A_{11}^{-1}A_{12}A_{22}^{-1}A_{21})$

Next, we apply the two-level preconditioner P in a multilevel setting. We test with the matrices from Problem 1, where on each level C_{11} is computed as $C_{11}^{(3)}$ and $B_{11} = C_{11}$. The multilevel construction requires an approximation of the Schur complement. For this test we have used the element-by-element technique, where on each level local Schur complements are computed exactly and summed up in a FEM manner. In other words, we compute

$$S_E^{(\ell} = A_{22,E}^{(\ell)} - A_{22,E}^{(\ell)} (A_{11,E}^{(\ell)})^{-1} A_{12,E}^{(\ell)} \quad \text{and let} \quad S = \sum_{\ell} S_E^{(\ell)}.$$

The so-obtained matrices $S_E^{(\ell}$ play the role of the element matrices on the coarser levels so that the construction can be repeated recursively. We see, that on the coarser levels the value of σ decreases as well as the condition number of the preconditioned system $P^{-1}A$. We also see that the eigenvalue bounds are quite tight.

size$(A/A_{11}/S)$	γ	σ	$eig(P^{-1}A)$ λ_{min}	λ_{max}	$cond(P^{-1}A)$	$\|I_1 - C_{11}A_{11}\|$
			$\varepsilon = 1$			
			5 levels of refinement			
1089/800/289	0.9944	3.8919	0.1749	5.717	32.684	0.15702
289/208/ 81	0.9725	0.57897	0.4754	2.1036	4.4251	0.13464
81/ 56/ 25	0.8779	0.071477	0.7660	1.3055	1.7042	0.10544
25/ 16/ 9	0.5480	0.002499	0.9513	1.0513	1.1051	0.05545
9/ 5/ 4	0.0468	0	1	1	1	0.00568
			6 levels of refinement			
4225/3136/1089	0.9987	17.170	0.0523	19.118	365.48	0.16005
1089/ 800/289	0.9934	2.8380	0.2164	4.6216	21.359	0.14355
289/ 208/81	0.9703	0.5008	0.4997	2.0011	4.0043	0.12842
81/ 56/25	0.8729	0.0644	0.7765	1.2879	1.6587	0.10128
25/ 16/9	0.5395	0.0023	0.9536	1.0486	1.0996	0.05306
9/ 5/4	0.0529	0	1	1	1	0.00506

Problem 1: $\widetilde{A}_{11} = A_{11}$, $S = S_2$, $C_{11}^{(3)}$, $B_{11} = C_{11}$

size$(A/A_{11}/S)$	γ	σ	$eig(P^{-1}A)$ λ_{min}	λ_{max}	$cond(P^{-1}A)$	$\|I_1 - C_{11}A_{11}\|$
			$\varepsilon = 0.01$			
			5 levels of refinement			
1089/800/289	0.9966	2.5871	0.22948	4.3576	18.9886	0.17364
289/208/ 81	0.9814	0.023764	0.85727	1.1665	1.3607	0.062409
81/ 56/ 25	0.8945	0.000125	0.98887	1.0113	1.0226	0.009115
25/ 16/ 9	0.5862	8.05e-10	0.99997	1	1.0001	1.7279e-5
9/ 5/ 4	3.81e-8	0	1	1	1	0
			6 levels of refinement			
4225/3136/1089	0.9986	3.7184	0.1806	5.5378	30.668	0.1452
1089/ 800/289	0.9958	0.2972	0.5835	1.7137	2.9367	0.0909
289/ 208/81	0.9785	0.0096	0.9065	1.1031	1.2169	0.0282
81/ 56/25	0.8996	2.63e-5	0.9949	1.0051	1.0103	0.0010
25/ 16/9	0.6278	2.60e-12	1	1	1	1.06e-6
9/ 5/4	0	0	1	1	1	0

Problem 1: $\widetilde{A}_{11} = A_{11}$, $S = S_2$, $C_{11}^{(3)}$, $B_{11} = C_{11}$

Problem 2:

The next table contains results for matrices arising from Problem 2. It is well known that a high quality approximation of the mass matrix is its diagonal and therefore we set $\widetilde{A}_{11} = B_{11} = C_{11} = (diag(A_{11}))^{-1}$ and $S = A_{22} - A_{21}(diag(A_{11}))^{-1}A_{12}$.

We include the iteration counts to solve one system with the Jacobian matrix with a block-factorized preconditioner and with a block-triangular preconditioner (the case when $B_{11} = 0$). Systems with $\widetilde{A}_{11}$ and with S are solved by a direct method. In this example the system matrix is not symmetric and not positive definite in general. We see, that the value of γ is larger than 1 and nearly doubles when we refine the mesh once, while σ is less than 1 and its increase is much less pronounced and even stabilizes. We include also the numerically estimated two-norm of the error matrices $\|I_1 - C_{11}A_{11}\|$ and $\|I_1 - S^{-1}S_2\|$, which illustrate the effect on approximation of the inverse of A_{11} (the mass matrix in this case) by the inverse of its diagonal.

		$B_{11} = C_{11}$					$B_{11} = 0$
Size	γ	σ	δ	$\rho(I_1 - C_{11}A_{11})$	$\rho(I_1 - S^{-1}S_2)$	it	it
$C_{11} = diag(A_{11})^{-1}$							
578	0.7069	0.371	0.99869	1	0.9961	12	22
2178	0.7071	0.555	0.99968	1	0.9991	13	22
8450	-	-	-	-	-	14	23
33282	-	-	-	-	-	14	23
132098	-	-	-	-	-	14	23
$C_{11} = C_{11}^{(3)}$							
578	0.7069	0.0444	0.171	0.4133	0.2646	8	14
2178	0.7071	0.0453	0.179	0.4232	0.2697	8	14
8450	-	-	-	-	-	8	14
33282	-	-	-	-	-	8	14
132098	-	-	-	-	-	8	14

Problem 2: Iteration counts for the block-factorized and the block upper-triangular preconditioners; values of γ, δ and σ computed for the small-sized tests; $\Delta t = h^2$

Conclusions

- A general form of approximate block factorizations for matrices on two-by-two block form has been presented.
- A new parameter (σ) to measure the quality of the corresponding preconditioner has been introduced. This replaces the previously commonly used parameter, the CBS constant γ. The latter is fixed for the given matrix, i.e., does not depend on the preconditioner and, furthermore, is applicable only for symmetric positive definite problems.
- A problem with the parameter σ is that it can not be computed locally, as the parameter γ can. However, its upper bound involves $\gamma^2/(1-\gamma^2)$, which can be computed locally, and the additional factors in σ can be approximated by its values on local subdomains.
- By involving inner iterations, one can get arbitrarily accurate preconditioners or, at least in the limit, of a form leading to just two or three conjugate gradients iterations.
- For matrices on saddle point form, one can use a form of regularization which implies that the Schur complement approaches a multiple of the identity matrix, i.e., there is then no need to devise some other approximations of the Schur complement matrix.
- Several of the results have been illustrated by numerical tests.

Thank you for your attention!

On an augmented Lagrangian-based preconditioning of Oseen type problems

Maya Neytcheva

Division of Scientific Computing

Department of Information Technology

Uppsala University, Sweden

Joint work with He Xin and Stefano Serra

Institute of Geonics
Ostrava

Large scale problems in Statistics

Plan of the talk:

- Saddle point matrices, preconditioning, Oseen's problem
- The augmented Lagrangian technique - two approaches
- Do we, indeed, avoid the need to approximate a Schur complement matrix?
- On the approximation of FEM mass matrices
- Numerical experiments

Motivation: multiphase flow

Processes, modelled by the (convective) Cahn-Hilliard equ.:

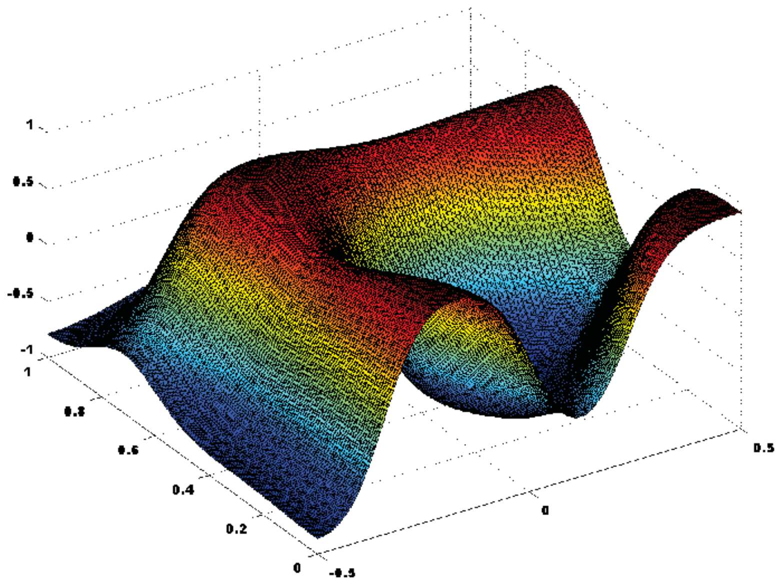
$$\frac{\partial C}{\partial t} + (\mathbf{u} \cdot \nabla)C = \nabla \cdot [\kappa(C)\nabla \left(\Psi'(C) - \epsilon^2 \Delta C\right)].$$

Above, $\mathbf{u}$ is the velocity vector, obtained as a solution of the time-dependent Navier-Stokes (N-S) equation:

$$\rho\frac{\partial \mathbf{u}}{\partial t} + (\rho\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \nabla \cdot [\mu(\nabla)\mathbf{u} + \nabla \mathbf{u}^T] - \eta\nabla C + F.$$

Here p is the pressure, ρ and $\mu(\mathbf{x})$ are the density and viscosity, correspondingly, and F is the force term. The term $\eta\nabla C$, where $\eta = \Psi'(C) - \epsilon^2\Delta C$, gives the coupling with C-H and represents the surface tension force in a potential form.

Motivation: multiphase flow



Oseen's problem:

$$\begin{aligned} -\nu\Delta \mathbf{u} + (\mathbf{w}\cdot\nabla)\mathbf{u} + \nabla p = & \quad \mathbf{f} \quad \text{in } \Omega, \\ \operatorname{div}\mathbf{u} = & \quad 0 \quad \text{in } \Omega, \end{aligned}$$

subject to suitable boundary conditions on $\partial\Omega$.

$\nu > 0$	-	the kinematic viscosity coefficient, assumed to be constant.
Δ	-	the Laplace operator in $\mathbb{R}^d$
∇	-	the gradient operator
div	-	the divergence operator.

The first equation - conservation of momentum
the second equation - incompressibility condition

The (wind) vector $\mathbf{w}$ is an approximation of the velocity at a previous iteration step, so it is updated at every nonlinear iteration.

The discrete linear system:

$$\begin{bmatrix} A & B^T \\ B & O \end{bmatrix} \begin{bmatrix} \mathbf{u}_h \\ \mathbf{p}_h \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix} \quad \text{or} \quad \mathcal{A}\mathbf{x} = \mathbf{b},$$

where $\mathbf{u}_h$ is the discrete velocities vector, $\mathbf{p}_h$ is the discrete pressure vector and $\mathbf{x}^T = [\mathbf{u}_h^T \;\; \mathbf{p}_h^T]$.
The matrix A is the discretized convection-diffusion operator (nonsymmetric).
The matrix B is the (negative) divergence matrix and B^T is the gradient matrix.

How to precondition $\begin{bmatrix} A & B^T \\ B & O \end{bmatrix}$?

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} I_1 & 0 \\ A_{21}A_{11}^{-1} & I_2 \end{bmatrix} \begin{bmatrix} A_{11} & 0 \\ 0 & S_2 \end{bmatrix} \begin{bmatrix} I_1 & A_{11}^{-1}A_{12} \\ 0 & I_2 \end{bmatrix}$$

$S_2 = A_{22} - A_{21}A_{11}^{-1}A_{12}$

$$\mathcal{M}_F = \begin{bmatrix} \widetilde{A} & O \\ B & S \end{bmatrix} \begin{bmatrix} I_1 & \widetilde{A}^{-1}B^T \\ O & I_2 \end{bmatrix}$$

$$\mathcal{M}_D = \begin{bmatrix} \widetilde{A} & O \\ 0 & S \end{bmatrix}, \qquad \mathcal{M}_L = \begin{bmatrix} \widetilde{A} & O \\ B & S \end{bmatrix}, \qquad \mathcal{M}_U = \begin{bmatrix} \widetilde{A} & B^T \\ 0 & S \end{bmatrix}$$

How to approximate the Schur complement $BA^{-1}B^T$?!

(1) Pressure mass matrix M_p

(2) The pressure convection-diffusion (PCD) preconditioner $S_{PCD}^{-1} = \widetilde{M_p}^{-1} A_p L_p^{-1}$,

(3) The BFBt preconditioner $S_{BFBt}^{-1} = (B\widehat{M}_u^{-1}B^T)^{-1} B\widehat{M}_u^{-1} A \widehat{M}_u^{-1} B^T (B\widehat{M}_u^{-1}B^T)^{-1}$,

(4) Element-by-element Schur complement

$$S_{EBE} = \sum_e S_{2,e} = \sum_e A_{22,e} - A_{21,e}A_{11,e}^{-1}A_{12,e}$$

How to avoid approximating the Schur complement?

Augmented Lagrangian technique:
Approach 1: Consider a regularized linear system (*not consistent* with the original one),

$$\begin{bmatrix} A & B^T \\ B & -\frac{1}{r}W \end{bmatrix} \begin{bmatrix} \mathbf{u}_h \\ \mathbf{p}_h \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix} \qquad \text{or} \qquad \overline{\mathcal{A}}\mathbf{x} = \mathbf{b},$$

for some large enough scalar parameter r and some spd matrix W. Systems of this form are obtained via various stabilization techniques.

$\mathcal{A} = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix}, \overline{\mathcal{A}} = \begin{bmatrix} A & B^T \\ B & -\frac{1}{r}W \end{bmatrix}$

$$\overline{\mathcal{A}} = \begin{bmatrix} I_1 & -rB^TW^{-1} \\ 0 & I_2 \end{bmatrix} \begin{bmatrix} A + rB^TW^{-1}B & 0 \\ 0 & -\frac{1}{r}W \end{bmatrix} \begin{bmatrix} I_1 & 0 \\ -rW^{-1}B & I_2 \end{bmatrix}$$

$$\mathcal{M}_L = \begin{bmatrix} A + rB^TW^{-1}B & 0 \\ B & -\frac{1}{r}W \end{bmatrix} \quad \text{or} \quad \mathcal{M}_U = \begin{bmatrix} A + rB^TW^{-1}B & B^T \\ 0 & -\frac{1}{r}W \end{bmatrix}$$

- good candidates to precondition the matrix $\overline{\mathcal{A}}$ and, for large values of r, for $\mathcal{A}$ itself. » b

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix}$$

Approach 2: Transform the original system into an *equivalent* one:

$$\begin{bmatrix} A + rB^TW^{-1}B & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_h \\ \mathbf{p}_h \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{f}} \\ \mathbf{g} \end{bmatrix}$$

where $\widehat{\mathbf{f}} = \mathbf{f} + rB^TW^{-1}B\,\mathbf{g}$.

NOTE: the transformation holds for any value of r, including $r = 1$ or $r \ll 1$, and there is more freedom in choosing the (nonsingular) matrix W.

$$\widetilde{\mathcal{A}} = \begin{bmatrix} A + rB^TW^{-1}B & B^T \\ B & 0 \end{bmatrix}, \ \widetilde{\mathcal{A}}\mathbf{v} = \lambda \mathcal{M}_L \mathbf{v}$$

$$\mathcal{M}_L^{-1}\widetilde{\mathcal{A}} = \begin{bmatrix} I & (A + rB^TW^{-1}B)^{-1}B^T \\ 0 & rW^{-1}B(A + rB^TW^{-1}B)^{-1}B^T \end{bmatrix}.$$

Apply Sherman-Morrison-Woodbury's formula to $(A + rB^TW^{-1}B)^{-1}$:

$$rW^{-1}B(A + rB^TW^{-1}B)^{-1}B^T = rQ - rQ(I + rQ)^{-1}rQ,$$

where $Q = W^{-1}BA^{-1}B^T$. » f

$$\mathcal{M}_L^{-1}\widetilde{\mathcal{A}} = \begin{bmatrix} I & (A + rB^TW^{-1}B)^{-1}B^T \\ 0 & rW^{-1}B(A + rB^TW^{-1}B)^{-1}B^T \end{bmatrix}.$$

Lemma 1 *Let the matrices $\widetilde{\mathcal{A}}$ and $\mathcal{M}_L$ are defined above, and let μ be any eigenvalue of $BA^{-1}B^T\mathbf{w} = \mu W\mathbf{w}$. Let δ be an eigenvalue of the matrix $\widetilde{Q} \equiv rQ - rQ(I + rQ)^{-1}rQ$, where $Q = W^{-1}BA^{-1}B^T$. Then the following holds:*
The matrices Q and $\widetilde{Q}$ have the same eigenvectors and the eigenvalues of $\widetilde{Q}$ are equal to

$$\delta = \frac{r\mu}{1 + r\mu} = \frac{1}{1 + \frac{1}{r\mu}}.$$

When $r \to \infty$ all nonzero eigenvalues of the eigenproblem $\widetilde{\mathcal{A}}\mathbf{v} = \lambda\mathcal{M}_L\mathbf{v}$ converge to 1.

Consider now the modified pivot block in $\begin{bmatrix} A + rB^TW^{-1}B & B^T \\ B & 0 \end{bmatrix}$:

How to solve systems with $\widehat{A} = A + rB^TW^{-1}B$?

$$A + rB^TW^{-1}B = (I + rB^TW^{-1}BA^{-1})A$$

UPPSALA UNIVERSITET

We notice, that

$Q = W^{-1}BA^{-1}B^T$ and $\mathcal{Q} = B^TW^{-1}BA^{-1}$
have the same spectra.

UPPSALA UNIVERSITET

We notice, that

$Q = W^{-1}BA^{-1}B^T$ and $\mathcal{Q} = B^TW^{-1}BA^{-1}$
have the same spectra.

$$Q = W^{-1}BA^{-1}B^T \qquad \mathcal{Q} = B^TW^{-1}BA^{-1}$$

UPPSALA UNIVERSITET

We notice, that

$Q = W^{-1}BA^{-1}B^T$ and $\mathcal{Q} = B^T W^{-1} B A^{-1}$
have the same spectra.

$Q = W^{-1}BA^{-1}B^T$ $\mathcal{Q} = B^T W^{-1} B A^{-1}$

Lemma 4 *Let X and Y be two general matrices, $X \in \mathbb{C}^{n\times m}$ and $Y \in \mathbb{C}^{m\times n}$. Then the spectra of the products XY and YX are identical up to some additional zero eigenvalues of multiplicity $(max(n,m) - min(n,m))$.*

»skip proof

UPPSALA UNIVERSITET

XY and YX have the same spectra

Proof:

(a) Let X and Y be square, i.e., $n = m$.

(a1) Let X be nonsingular. Then XY is spectrally equivalent to YX because of the trivial equality $XY = X(YX)X^{-1}$.

XY and YX have the same spectra

(a2) Let both X and Y be singular. We consider the Schur decomposition of X, $X = UTU^*$, where U is unitary and T is upper triangular with diagonal elements equal to the eigenvalues of X. Clearly, at least one diagonal entry in T is zero. Let $\varepsilon > 0$ and construct T_ε as following

$$(T_\varepsilon)_{ij} = \begin{cases} T_{ij} & \text{for } i \neq j; \\ T_{ii} & \text{if } T_{ii} \neq 0; \\ \varepsilon & \text{if } T_{ii} = 0. \end{cases}$$

Let $X_\varepsilon = UT_\varepsilon U^*$. Clearly X_ε is nonsingular and $\|X - X_\varepsilon\| < \varepsilon$.
Due to (a1), there holds that the spectrum of $X_\varepsilon Y$ coincides with that of YX_ε as well as their characteristic polynomials, i.e., $P(X_\varepsilon Y) = P(YX_\varepsilon)$. Using the fact that the matrices are continuous functions of their entries and so are the coefficients of their characteristic polynomials, letting $\varepsilon \to 0$ we obtain that

$$P(XY) = P(YX).$$

XY and YX have the same spectra

(b) Let now $n \neq m$, and for convenience assume that $n > m$. Augmenting X and Y to square matrices by adding $(n-m)$ columns, respectively rows,

$$\widetilde{X} = \begin{bmatrix} X & 0 \end{bmatrix}, \quad \widetilde{Y} = \begin{bmatrix} Y \\ 0 \end{bmatrix},$$

$$\widetilde{X}\widetilde{Y} = XY,$$

$$\widetilde{Y}\widetilde{X} = \begin{bmatrix} Y \\ 0 \end{bmatrix} \begin{bmatrix} X & 0 \end{bmatrix} = \begin{bmatrix} YX & 0 \\ 0 & 0 \end{bmatrix}.$$

By (i2) we know that the spectrum of $\widetilde{X}\widetilde{Y}$ is equal to $\widetilde{Y}\widetilde{X}$, which in its turn coincides with the spectrum of YX augmented with $(n-m)$ zero eigenvalues.

UPPSALA UNIVERSITET

Two condition number indicators:

$$\kappa\left(A^{-1}(A+rB^TW^{-1}B)\right)=|1+r\mu| \qquad \kappa\left(\mathcal{M}_L^{-1}\widetilde{\mathcal{A}}\right)=|1+1/(r\mu)|$$

Try to balance the two constraints, for instance, by minimizing the quantity

$$\left(|1+r\mu|^p+|1+1/r\mu|^p\right)^{1/p},$$

A direct computation shows that, independently of p, $r\mu$ should be $O(1)$ and since we want r to be also $O(1)$, we conclude that $\mu=O(1)$, i.e., W should approximate well the negative Schur complement of the original system, $BA^{-1}B^T$. More specifically, the minimum is attained for real positive μ and for $p=1$ the global minimum over the positive complex half plane is attained when $r\mu=1$.

UPPSALA UNIVERSITET

Outcome:

The above reasonings, although appearing to be trivial, show that the attempts to approximate the modified pivot block $A+rB^TW^{-1}B$ with matrices or computational procedures, which perform reasonably well for small values of r, ultimately destroy the outer convergence rate.
Thus, the weight of the product $B^TW^{-1}B$ is significant and cannot be neglected:

- either find a good approximation of the Schur complement, or
- solve the modified block very accurately.

UPPSALA UNIVERSITET

The influence of r on the outer and the inner convergence rates:

Size	$r=\nu$	$r=1$	$r=1/\nu$	$r=1000$
		$\nu=1$		
578	15(10)	15(10)	15(10)	2(42)
2178	15(10)	15(10)	15(10)	2(51)
8450	15(11)	15(11)	15(11)	2(53)
		$\nu=0.1$		
578	28(3)	16(11)	4(38)	2(54)
2178	29(3)	15(13)	4(44)	2(66)
8450	26(3)	14(14)	4(45)	2(71)
		$\nu=0.01$		
578	58(3)	49(11)	5(65)	3(76)
2178	83(3)	47(14)	4(115)	2(135)
8450	106(3)	45(18)	4(154)	2(179)

Outer(inner) iterations as functions of ν and r

«< conclude

UPPSALA UNIVERSITET

Mass matrices

The finite element matrices

$$M = \sum_{k=1}^{n_E} R_k^T M_k R_k.$$

Here R_k are the standard Boolean matrices which prescribe the local-to-global correspondence of the degrees of freedom. The matrix M is symmetric, positive definite.

For some special FEM discretizations M is diagonal. (Example: nonconforming FEM)

For the conforming FEM discretization M is a sparse matrix.

Tasks:
- How to approximate M^{-1}?
- How to measure how good the approximation is?

Known results:

I. Fried, Bounds on the spectral and maximum norms of the finite element stiffness, flexibility and mass matrices. *International Journal of Solids and Structures*, 9 (1973), 1013–1034.

A.J. Wathen, Realistic eigenvalue bounds for the Galerkin mass matrix, *IMA Journal of Numerical Analysis* 7 (1987), 449-457.

I. Fried and M. Coleman, Improvable bounds on the largest eigenvalue of a completely positive finite element flexibility matrix *Journal of Sound and Vibration*, (283) 2005, pp. 487-494

UPPSALA UNIVERSITET

An element-by-element representations

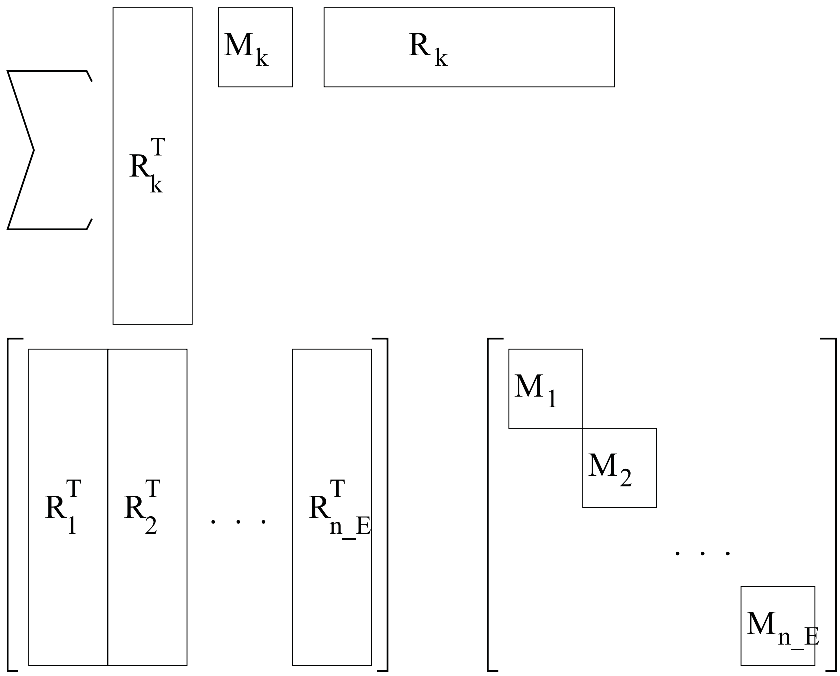
$$M = \sum_{k=1}^{n_E} R_k^T M_k R_k.$$

$$M = R^T diag(M_k) R,$$

where $R^T = [R_1^T, \cdots, R_{n_E}^T]$.

UPPSALA UNIVERSITET

$M = \sum_{k=1}^{n_E} R_k^T M_k R_k$, $M = R^T diag(M_k) R$



Estimating the eigenvalues of M via the eigenvalues of the element stiffness matrices M_k

I. Fried:

$$\min_{k=1,\cdots,n_E}(\lambda_{min}(M_k)) \leq \lambda(M) \leq p_{max} \max_{k=1,\cdots,n_E}(\lambda_{max}(M_k))$$

where p_{max} is the degree of the graph of the mesh (the maximum number of elements around any nodal points).

How to approximate M^{-1}

Denote:

D_M	the (pointwise) diagonal of M
D_M^{-1}	$inv(D_M)$
L_M	the lumped mass matrix
$\widehat{M}^{-1}$	the element-by-element sparse approximate inverse of M^{-1}: $\widehat{M}^{-1} = \sum_{k=1}^{n_E} R_k^T M_k^{-1} R_k$
$D_{\widehat{M}}^{-1}$	the diagonal of $\widehat{M}^{-1}$

$\varkappa(D_M^{-1}M)$

Recall: $M = R^T diag(M_k)R$.
Let λ be an eigenvalue of $D_M^{-1}M$.
Using Rayleigh quotient:

$$\min_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}^T M\mathbf{x}}{\mathbf{x}^T D_M\mathbf{x}} \leq \lambda \leq \max_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}^T M\mathbf{x}}{\mathbf{x}^T D_M\mathbf{x}}$$

$$\min_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}^T L^T diag(M_k)L\mathbf{x}}{\mathbf{x}^T L^T diag(D_{M_k})L\mathbf{x}} \leq \lambda \leq \max_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{x}^T L^T diag(M_k)L\mathbf{x}}{\mathbf{x}^T L^T diag(D_{M_k})L\mathbf{x}}$$

$$\min_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{y}^T diag(M_k)\mathbf{y}}{\mathbf{y}^T diag(D_{M_k})\mathbf{y}} \leq \lambda \leq \max_{\mathbf{x}\neq\mathbf{0}} \frac{\mathbf{y}^T diag(M_k)\mathbf{y}}{\mathbf{y}^T diag(D_{M_k})\mathbf{y}}$$

$$\min_{e} \lambda_{min}(D_k^{-1}M_k) \leq \lambda \leq \max_{e} \lambda_{min}(D_k^{-1}M_k)$$

Some particular element matrices

Element	λ_{min}	λ_{max}	$\varkappa(D_M^{-1}M)$
Arbitrary linear triangles	1/2	2	4
Bilinear rectangles	1/4	9/4	9
Arbitrary linear tetrahedra	1/2	5/2	5
Rectangular trilinear bricks	1/8	27/8	27

Straightforward concequence: θ-method

For parabolic problems:

$$(M + \theta \Delta t K)U^{n+1} = (M - (1-\theta)\Delta t K)U^n$$

$$M + \theta \Delta t K = L^T \left(diag(M_e + \theta \Delta t K_e) \right) L$$

Can we do better?

Consider the element-by-element idea:

$$\begin{aligned} M_k &= \frac{\alpha_k}{12} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}, & \lambda(M_k) &= \frac{\alpha_k}{12}[1,1,4] \\ D_{M_k^{-1}} M_k &= \frac{3}{4} \begin{bmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{bmatrix}, & \lambda(D_{M_k^{-1}} M_k) &= \frac{3}{4}[1,1,4] \end{aligned}$$

Thus, $\varkappa(D_{\widehat{M}^{-1}} M) \leq 4$, $\varkappa(\widehat{M}^{-1} D^{-1}_{\widehat{M}^{-1}}) \leq 4$ and $\varkappa(\widehat{M}^{-1} M) \leq 16$.

UPPSALA UNIVERSITET

Preconditioners, involving mass matrices:

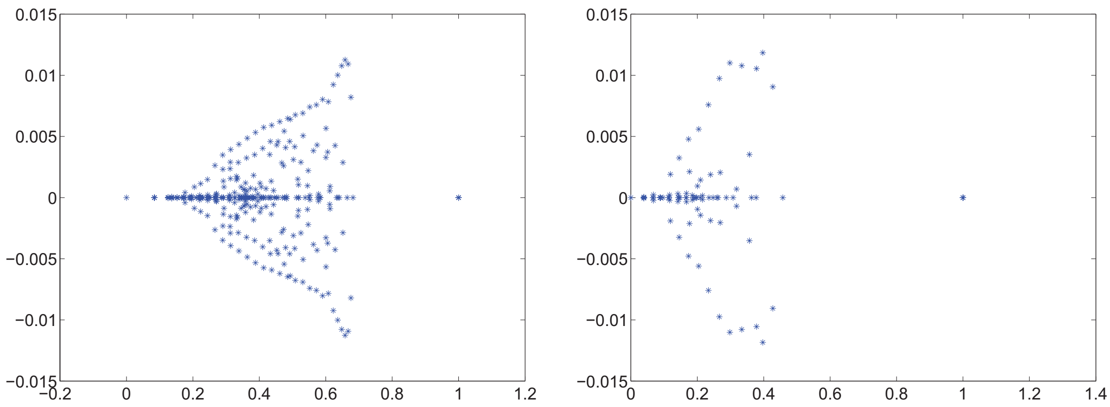
Stationary Navier-Stokes:

$$\widetilde{A} = \begin{bmatrix} A + \gamma B^T M^{-1} B & B^T \\ B & 0 \end{bmatrix}$$

$$\mathcal{M} = \begin{bmatrix} A + \gamma B^T M^{-1} B & 0 \\ B & -\frac{1}{\gamma} M \end{bmatrix}$$

UPPSALA UNIVERSITET

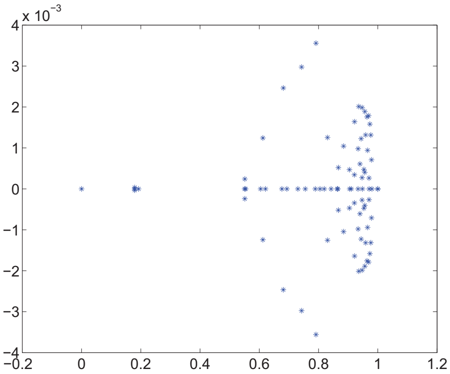
$$\begin{bmatrix} A + \gamma B^T M^{-1} B & 0 \\ B & -\frac{1}{\gamma} M \end{bmatrix}^{-1} \begin{bmatrix} A + \gamma B^T M^{-1} B & B^T \\ B & 0 \end{bmatrix}$$



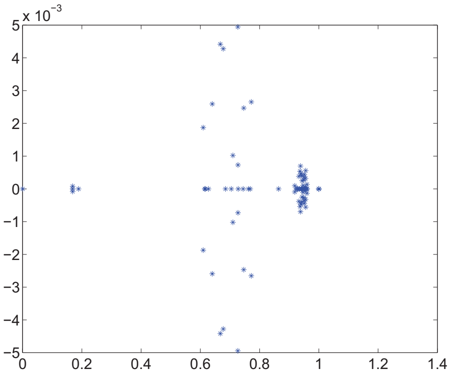
(a) D_M^{-1} (b) L_M^{-1}

The spectrum of $\mathcal{M}_L^{-1}\widetilde{\mathcal{A}}$,$\gamma = 1$, $\nu = 1$

$$\begin{bmatrix} A+\gamma B^T M^{-1} B & 0 \\ B & -\frac{1}{\gamma} M \end{bmatrix}^{-1} \begin{bmatrix} A+\gamma B^T M^{-1} B & B^T \\ B & 0 \end{bmatrix}$$



(c) $D_{\widehat{M}^{-1}}$

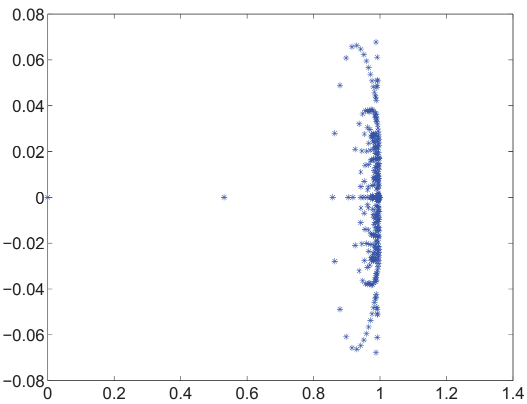


(d) $\widehat{M}^{-1}$

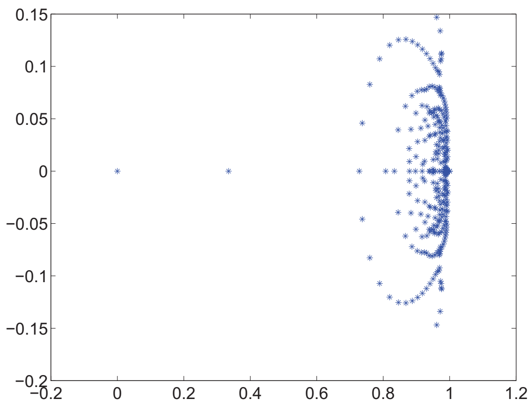
The spectrum of $\mathcal{M}_L^{-1}\widetilde{\mathcal{A}}$,$\gamma = 1$, $\nu = 1$

UPPSALA UNIVERSITET

$$\begin{bmatrix} A+\gamma B^T M^{-1} B & 0 \\ B & -\frac{1}{\gamma} M \end{bmatrix}^{-1} \begin{bmatrix} A+\gamma B^T M^{-1} B & B^T \\ B & 0 \end{bmatrix}$$



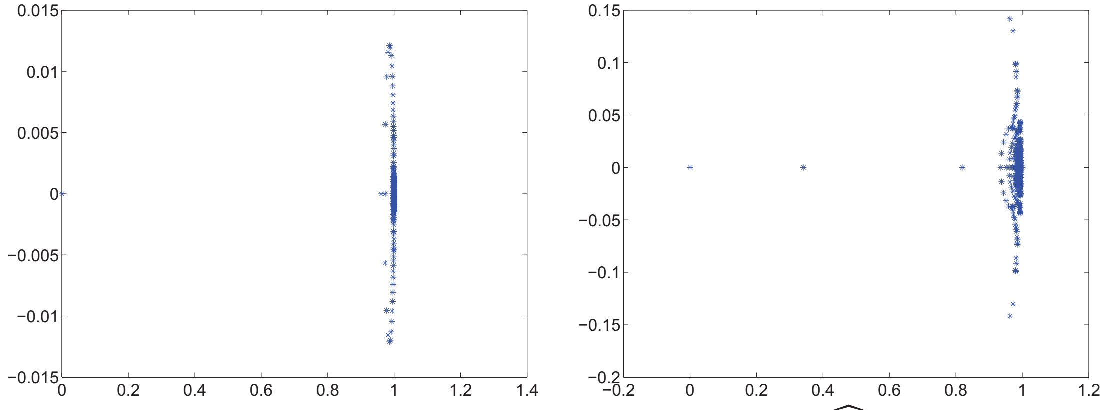
(e) D_M^{-1}



(f) L_M^{-1}

The spectrum of $\mathcal{M}_L^{-1}\widetilde{\mathcal{A}}$, $\gamma = 1$, $\nu = 1/320$

$$\begin{bmatrix} A+\gamma B^T M^{-1} B & 0 \\ B & -\frac{1}{\gamma} M \end{bmatrix}^{-1} \begin{bmatrix} A+\gamma B^T M^{-1} B & B^T \\ B & 0 \end{bmatrix}$$



(g) $D_{\widehat{M}^{-1}}$ (h) $\widehat{M}^{-1}$

The spectrum of $\mathcal{M}_L^{-1}\widetilde{\mathcal{A}}$, $\gamma=1$, $\nu=1/320$

Cahn-Hilliard equation:

$$\frac{\partial C}{\partial t} + (\mathbf{u}\cdot\nabla)C = \nabla\cdot[\kappa(C)\nabla(\Psi'(C) - \epsilon^2\Delta C)]$$

$$\begin{aligned} \eta - \Psi'(C) + \epsilon^2\Delta C &= 0, \qquad (\mathbf{x},t)\in\Omega_T \equiv \Omega\times(0,T) \\ \nabla\cdot[\kappa(C)\nabla\eta] - \frac{\partial C}{\partial t} - (\mathbf{u}\cdot\nabla)C &= 0, \qquad (\mathbf{x},t)\in\Omega_T \end{aligned}$$

$$\begin{bmatrix} \theta M & -\theta J(\mathbf{c}) - \theta\epsilon^2 K \\ \theta\kappa\Delta t_k K & M + \theta\Delta t_k W \end{bmatrix}.$$

UPPSALA UNIVERSITET

What about

$$\|I - \widehat{M}^{-1}M\|_F$$

or

$$\|I - D_M^{-1}M\|_F$$

UPPSALA UNIVERSITET

To conclude:

- In the context of the augmented Lagrangian technique, we still need a reasonably good approximation of the Schur complement matrix $BA^{-1}B^T$.
- It remains an open problem how to solve systems of the type $A + rB^TW^{-1}B$.
- There are very good quality diagonal approximations of the inverse of the mass matrix.
  $\|I - \widehat{M}^{-1}M\|$ can be further improved using Frobenius norm minimization techniques.

Thank you for your attention!

Czech Technical University in Prague
Faculty of Civil Engineering
Department of Mechanics

Damage Analysis and Arc-Length Methods

Jaroslav Kruis

Outline

- Damage
  - uniaxial case
  - mesh-adjusted softening modulus
  - multiaxial case
- Non-linear equations
- Arc-legth method
  - spherical and cylindrical method
  - linearized method
  - original method

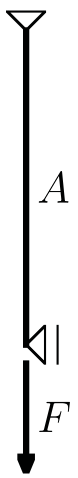
Damage

The theory of damage describes the evolution between the virgin state and macroscopic crack initiation.

The phenomenon of damage represents surface discontinuities in the form of microcracks, or volume discontinuities in the form of cavities.

CTU Damage, Arc-length Method J. Kruis

Uniaxial Case



A	area of cross section of bar
$\tilde{A}$	area of undamaged part of cross section
A_d	area of damaged part of cross section
$\sigma = \dfrac{F}{A}$	apparent stress
$\tilde{\sigma} = \dfrac{F}{\tilde{A}}$	effective stress
ε	strain

1. 12. – 3. 12. 2010 Institute of Geonics AS CR Ostrava

areas of cross section

$$A = \tilde{A} + A_d$$

damage parameter

$$\omega = \frac{A_d}{A} \qquad \omega \in \langle 0; 1 \rangle$$

$$F = \tilde{\sigma}\tilde{A} = \sigma A \quad \Rightarrow \quad \sigma = \frac{A - A_d}{A}\tilde{\sigma}$$

effective stress

$$\tilde{\sigma} = \frac{1}{1-\omega}\sigma$$

Principles of Equivalence

Principle of Strain Equivalence–The Effective Stress Concept

The strain associated with a damaged state under the applied stress σ is equivalent to the strain associated with the undamaged state under the effective stress $\tilde{\sigma}$.

$$\left.\begin{array}{l} \tilde{\sigma} = E\tilde{\varepsilon} = E\varepsilon \\ \sigma = \tilde{E}\varepsilon \end{array}\right\} \frac{\tilde{\sigma}}{E} = \frac{\sigma}{\tilde{E}} \quad \Rightarrow \quad \frac{\tilde{E}}{E} = 1 - \omega$$

Principle of Stress Equivalence–The Effective Strain Concept

The stress associated with a damaged state under the applied strain ε is equivalent to the stress associated with the undamaged state under the effective strain $\tilde{\varepsilon}$.

$$\left.\begin{aligned} \tilde{\varepsilon} &= \frac{\tilde{\sigma}}{E} = \frac{\sigma}{E} \\ \varepsilon &= \frac{\sigma}{\tilde{E}} \end{aligned}\right\} \tilde{\varepsilon}E = \tilde{E}\varepsilon \quad \Rightarrow \quad \frac{E}{\tilde{E}} = \frac{1}{1-\omega}$$

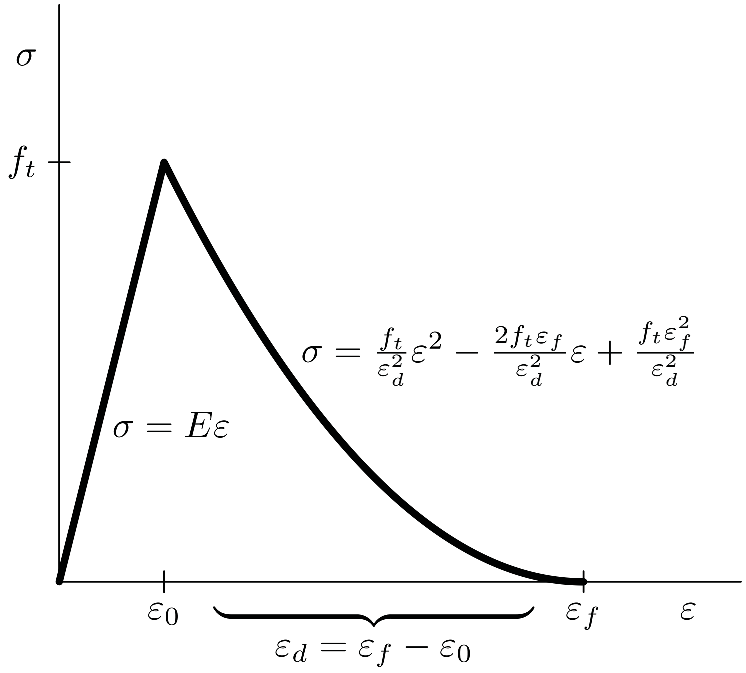
CTU Damage, Arc-length Method J. Kruis

Principle of Elastic Energy Equivalence

$$\left.\begin{aligned} \tilde{\sigma} &= E\tilde{\varepsilon} \qquad & \tilde{W} &= \frac{1}{2}\tilde{\sigma}\tilde{\varepsilon} \\ \sigma &= \tilde{E}\varepsilon \qquad & W &= \frac{1}{2}\sigma\varepsilon \end{aligned}\right\} \tilde{\sigma}\tilde{\varepsilon} = \sigma\varepsilon \quad \Rightarrow \quad \sqrt{\frac{\tilde{E}}{E}} = 1-\omega$$

1. 12. – 3. 12. 2010 Institute of Geonics AS CR Ostrava

Stress–Strain Diagram



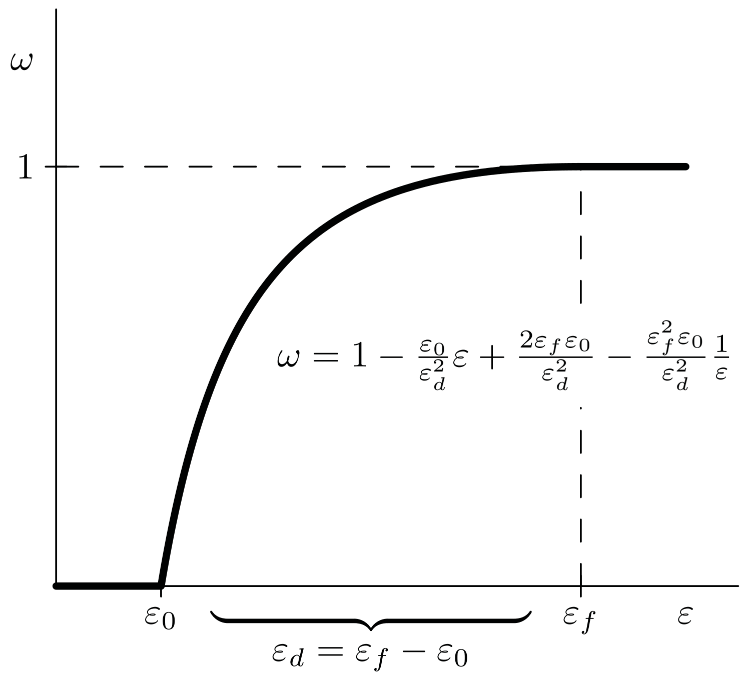
$$\sigma = (1-\omega)E\varepsilon = (1-g(\varepsilon))E\varepsilon$$

$$g(\varepsilon) = 1 - \frac{\sigma}{E\varepsilon}$$

stress-strain diagram with linear and quadratic functions

$$\varepsilon \in \langle 0; \varepsilon_0 \rangle \qquad \sigma = E\varepsilon$$
$$\varepsilon \in \langle \varepsilon_0; \varepsilon_f \rangle \qquad \sigma = \frac{f_t}{\varepsilon_d^2}\varepsilon^2 - \frac{2f_t\varepsilon_f}{\varepsilon_d^2}\varepsilon + \frac{f_t\varepsilon_f^2}{\varepsilon_d^2}$$

$$\omega = 1 - \frac{\varepsilon_0}{\varepsilon_d^2}\varepsilon + \frac{2\varepsilon_f\varepsilon_0}{\varepsilon_d^2} - \frac{\varepsilon_f^2\varepsilon_0}{\varepsilon_d^2}\frac{1}{\varepsilon}$$



loading function - specifies the elastic domain

$$f(\varepsilon,\kappa) = \varepsilon - \kappa \leq 0$$

κ is the largest strain ever reached in the history

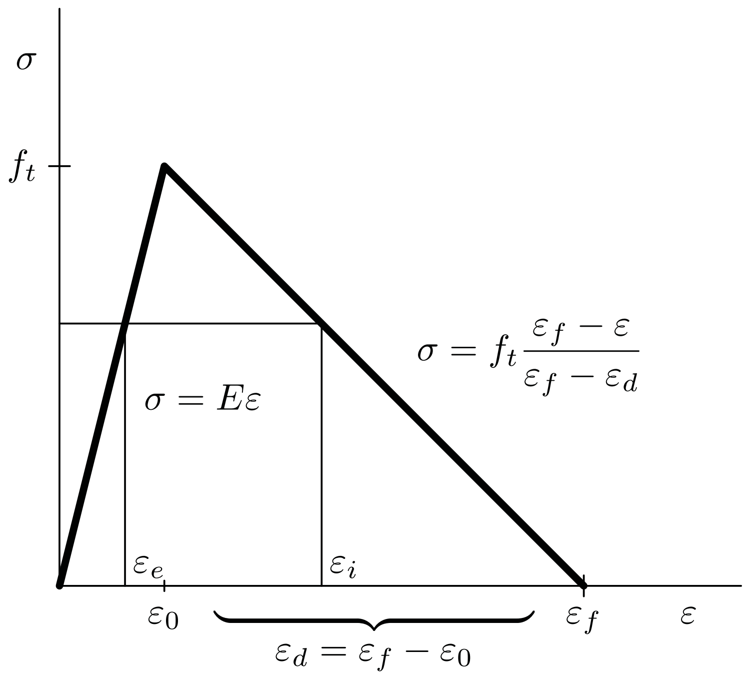
$$f(\varepsilon,\kappa) \leq 0, \quad \dot{\kappa} \geq 0, \quad \dot{\kappa} f(\varepsilon,\kappa) = 0$$

$f(\varepsilon,\kappa) < 0, \dot{\kappa} = 0$ elastic state

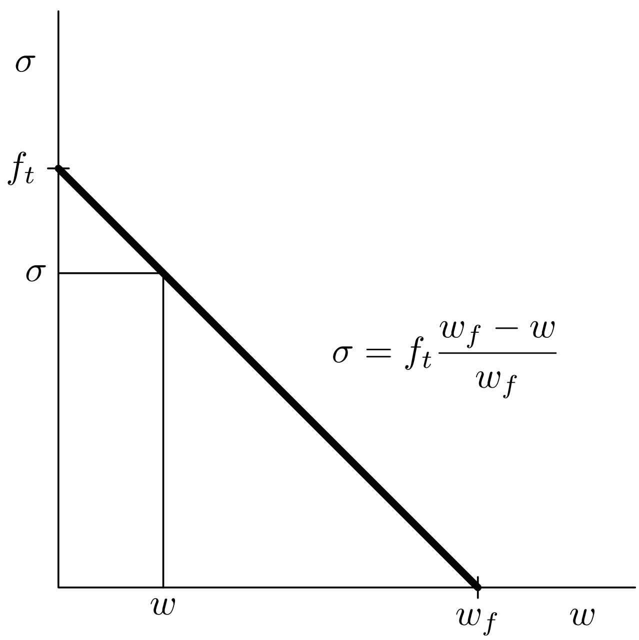
$f(\varepsilon,\kappa) = 0, \dot{\kappa} = 0$ neutral loading, damage does not grow

$f(\varepsilon,\kappa) = 0, \dot{\kappa} > 0$ damage grows

Mesh-Adjusted Softening Modulus



Traction–Separation Law



elongation of the bar

$$u = h\varepsilon_i + (l-h)\varepsilon_e = \varepsilon_f h + \sigma\left(\frac{l-h}{E} + \frac{\varepsilon_0 - \varepsilon_f}{f_t}h\right)$$

the traction-separation law

$$\sigma = f(w) = f_t\frac{w_f - w}{w_f}$$

where w is the crack opening and σ is the stress normal to the crack

cracking strain

$$\sigma = (1-\omega)E\varepsilon = E(\varepsilon - \omega\varepsilon) = E(\varepsilon - \varepsilon_c) = E\varepsilon_e$$

$$\varepsilon_c = \omega\varepsilon$$

the crack opening

$$w = h\varepsilon_c = h\omega\varepsilon$$

$$(1-\omega)E\varepsilon = f_t\left(1 - \frac{h\omega\varepsilon}{w_f}\right)$$

$$w_f = h\varepsilon_f$$

$$\omega = 1 - \frac{\varepsilon_0}{\varepsilon}\frac{w_f - \varepsilon h}{w_f - \varepsilon_0 h} = 1 - \frac{\varepsilon_0}{\varepsilon}\frac{\varepsilon_f - \varepsilon}{\varepsilon_f - \varepsilon_0}$$

exponential traction–separation law

$$(1-\omega)E\varepsilon = f_t e^{-\frac{w}{w_f}} = f_t e^{-\frac{h\omega\varepsilon}{w_f}}$$

numerical solution

Multiaxial Case

strain components

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

strain tensor

$$\begin{pmatrix} \varepsilon_{xx} & \varepsilon_{xy} & \varepsilon_{xz} \\ \varepsilon_{yx} & \varepsilon_{yy} & \varepsilon_{yz} \\ \varepsilon_{zx} & \varepsilon_{zy} & \varepsilon_{zz} \end{pmatrix}$$

strain vector

$$\boldsymbol{\varepsilon}^T = (\varepsilon_{xx}, \varepsilon_{yy}, \varepsilon_{zz}, 2\varepsilon_{yz}, 2\varepsilon_{zx}, 2\varepsilon_{xy})$$

loading function

$$f(\boldsymbol{\varepsilon}, \kappa) = \hat{\varepsilon}(\boldsymbol{\varepsilon}) - \kappa \leq 0$$

$\hat{\varepsilon}(\boldsymbol{\varepsilon})$ is the equivalent strain

evolution law

$$\omega = \omega(\kappa)$$

Equivalent Strains

strain norm

$$\hat{\varepsilon} = \sqrt{\varepsilon_{xx}^2 + \varepsilon_{yy}^2 + \varepsilon_{zz}^2 + 2(\varepsilon_{yz}^2 + \varepsilon_{zx}^2 + \varepsilon_{xy}^2)}$$

$$\hat{\varepsilon} = \frac{1}{\sqrt{1 - 2\nu^2}} \sqrt{\varepsilon_{xx}^2 + \varepsilon_{yy}^2 + \varepsilon_{zz}^2 + 2(\varepsilon_{yz}^2 + \varepsilon_{zx}^2 + \varepsilon_{xy}^2)}$$

energy norm

$$\hat{\varepsilon} = \sqrt{\frac{\boldsymbol{\varepsilon}^T \boldsymbol{D} \boldsymbol{\varepsilon}}{E}}$$

Mazar's norm

$$\hat{\varepsilon} = \sqrt{\langle \varepsilon_\alpha \rangle \langle \varepsilon_\alpha \rangle} \qquad \varepsilon_\alpha \text{ are the principal strains}$$

isotropic damage

$$\tilde{\boldsymbol{\sigma}} = \frac{1}{1-\omega}\boldsymbol{\sigma}$$

asymmetric effective stress

$$\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}(\boldsymbol{I} - \boldsymbol{\Omega})^{-1}$$

symmetric part of the asymmetric effective stress

$$\tilde{\boldsymbol{\sigma}} = \frac{1}{2}\left(\boldsymbol{\sigma}(\boldsymbol{I} - \boldsymbol{\Omega})^{-1} + (\boldsymbol{I} - \boldsymbol{\Omega})^{-1}\boldsymbol{\sigma}\right)$$

$$\tilde{\boldsymbol{\sigma}} = (\boldsymbol{I} - \boldsymbol{\Omega})^{-\frac{1}{2}}\boldsymbol{\sigma}(\boldsymbol{I} - \boldsymbol{\Omega})^{-\frac{1}{2}}$$

Non-linear System of Algebraic Equations

body occupies a domain	$\Omega \in R^3$ with a boundary Γ
the boundary is split	$\Gamma = \Gamma_u \cup \Gamma_t \qquad (\Gamma_u \cap \Gamma_t = \emptyset)$
the displacement field	$\boldsymbol{u} : \Omega \to R^3$
the strain field	$\boldsymbol{\varepsilon} : \Omega \to R^6$
the stress field	$\boldsymbol{\sigma} : \Omega \to R^6$
the body forces	$\boldsymbol{b} : \Omega \to R^3$
the surface traction	$\boldsymbol{t} : \Gamma_t \to R^3$

equilibrium condition

$$\forall \boldsymbol{x} \in \Omega : \quad \boldsymbol{\partial \sigma} + \boldsymbol{b} = \boldsymbol{0}$$

boundary conditions

$$\begin{aligned} \forall \boldsymbol{x} \in \Gamma_u \quad &: \quad \boldsymbol{u} = \boldsymbol{0} \\ \forall \boldsymbol{x} \in \Gamma_t \quad &: \quad \boldsymbol{\sigma n} = \boldsymbol{t} \end{aligned}$$

strain–displacement relationship

$$\boldsymbol{\varepsilon} = \boldsymbol{\partial}^T \boldsymbol{u}$$

constitutive law

$$\boldsymbol{\sigma} \;\; = \;\; \boldsymbol{\sigma}(\boldsymbol{\varepsilon}) = \boldsymbol{\sigma}(\boldsymbol{u})$$

$$\int_\Omega \boldsymbol{\varphi}^T \boldsymbol{\partial \sigma} \mathrm{d}\Omega + \int_\Omega \boldsymbol{\varphi}^T \boldsymbol{b} \mathrm{d}\Omega = \boldsymbol{0}$$

$$-\int_\Omega \boldsymbol{\sigma}^T \boldsymbol{\partial \varphi} \mathrm{d}\Omega + \int_{\Gamma_t} \boldsymbol{\varphi}^T \boldsymbol{\sigma n} \mathrm{d}\Gamma + \int_\Omega \boldsymbol{\varphi}^T \boldsymbol{b} \mathrm{d}\Omega = \boldsymbol{0}$$

finite element discretization

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{N d} \\ \boldsymbol{\varphi} &= \boldsymbol{N p} \\ \boldsymbol{B} &= \boldsymbol{\partial N} \end{aligned}$$

$$-\boldsymbol{p}^T \int_\Omega \boldsymbol{B}^T \boldsymbol{\sigma} \mathrm{d}\Omega + \boldsymbol{p}^T \int_{\Gamma_t} \boldsymbol{N}^T \boldsymbol{t} \mathrm{d}\Gamma + \boldsymbol{p}^T \int_\Omega \boldsymbol{N}^T \boldsymbol{b} \mathrm{d}\Omega = \boldsymbol{0}$$

notation

$$\begin{aligned} \boldsymbol{f}^{int} &= \int_\Omega \boldsymbol{B}^T \boldsymbol{\sigma}(\boldsymbol{d}) \mathrm{d}\Omega \\ \boldsymbol{f}^{ext} &= \int_{\Gamma_t} \boldsymbol{N}^T \boldsymbol{t} \mathrm{d}\Gamma + \int_\Omega \boldsymbol{N}^T \boldsymbol{b} \mathrm{d}\Omega \end{aligned}$$

equilibrium condition

$$\boxed{\boldsymbol{f}_{int}(\boldsymbol{d}) = \boldsymbol{f}_{ext}}$$

CTU Damage, Arc-length Method J. Kruis

Arc-length Method

equilibrium condition

$$\boldsymbol{f}_{int}(\boldsymbol{d}) = \boldsymbol{f}_c + \lambda \boldsymbol{f}_p$$

residual – the vector of unbalanced forces

$$\boldsymbol{r}(\boldsymbol{d}, \lambda) = \boldsymbol{f}_c + \lambda \boldsymbol{f}_p - \boldsymbol{f}_{int}(\boldsymbol{d})$$

equilibrium

$$\boldsymbol{r}(\boldsymbol{d}, \lambda) = \boldsymbol{0}$$

assumption

the vector $\boldsymbol{d}_i$ and the parameter λ_i are known and $\boldsymbol{r}(\boldsymbol{d}_i, \lambda_i) = \boldsymbol{0}$

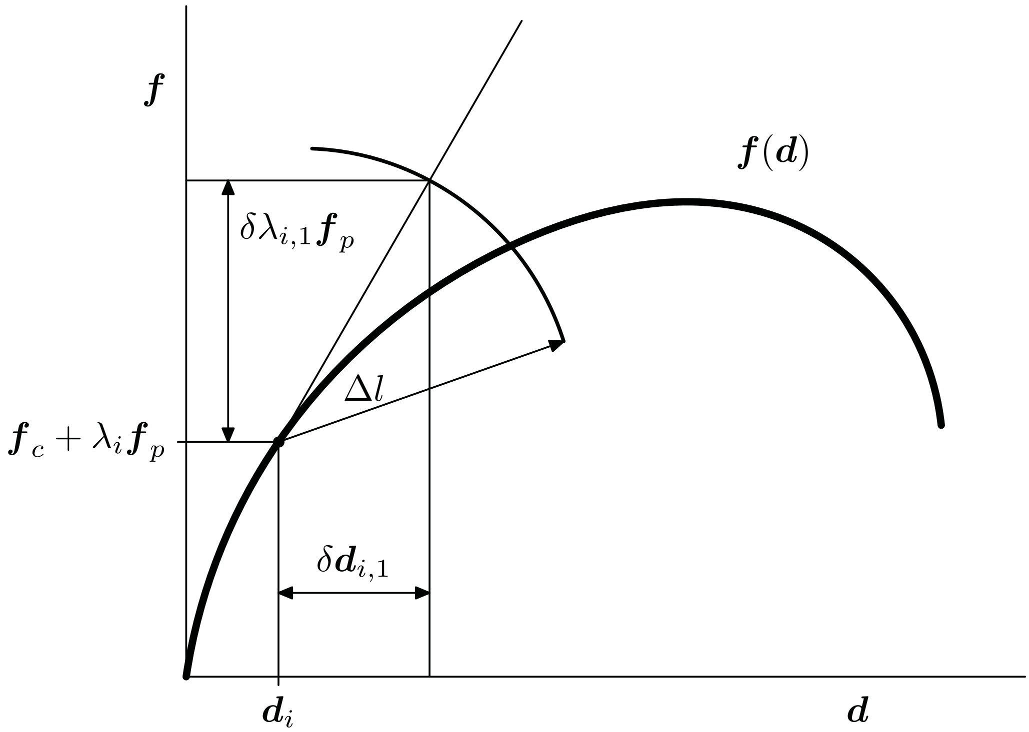
1. 12. – 3. 12. 2010 Institute of Geonics AS CR Ostrava

$$\boldsymbol{r}(\boldsymbol{d}_{i+1}, \lambda_{i+1}) = \boldsymbol{r}(\boldsymbol{d}_i, \lambda_i) + \frac{\partial \boldsymbol{r}(\boldsymbol{d}_i, \lambda_i)}{\partial \boldsymbol{d}} \delta \boldsymbol{d}_i + \frac{\partial \boldsymbol{r}(\boldsymbol{d}_i, \lambda_i)}{\partial \lambda} \delta \lambda_i$$

$$\frac{\partial \boldsymbol{r}(\boldsymbol{d}_i, \lambda_i)}{\partial \boldsymbol{d}} = -\boldsymbol{K}_i$$

$$\frac{\partial \boldsymbol{r}(\boldsymbol{d}_i, \lambda_i)}{\partial \lambda} = \boldsymbol{f}_p$$

$$\boldsymbol{r}(\boldsymbol{d}_{i+1}, \lambda_{i+1}) = -\boldsymbol{K}_{i,0} \delta \boldsymbol{d}_{i,1} + \boldsymbol{f}_p \delta \lambda_{i,1} = \boldsymbol{0}$$



assumption

$$\exists \boldsymbol{v}_{i,1} : \quad \delta \boldsymbol{d}_{i,1} = \delta\lambda_{i,1} \boldsymbol{v}_{i,1}$$

$$\delta\lambda_{i,1} \boldsymbol{K}_{i,0} \boldsymbol{v}_{i,1} = \delta\lambda_{i,1} \boldsymbol{f}_p \quad \Rightarrow \quad \boldsymbol{v}_{i,1} = \boldsymbol{K}_{i,0}^{-1} \boldsymbol{f}_p$$

the length of arc

$$(\delta \boldsymbol{d}_{i,1})^T \delta \boldsymbol{d}_{i,1} + \psi^2 (\delta\lambda_{i,1})^2 \boldsymbol{f}_p^T \boldsymbol{f}_p = (\Delta l)^2$$

$$(\delta\lambda_{i,1})^2 \boldsymbol{v}_{i,1}^T \boldsymbol{v}_{i,1} + \psi^2 (\delta\lambda_{i,1})^2 \boldsymbol{f}_p^T \boldsymbol{f}_p = (\Delta l)^2$$

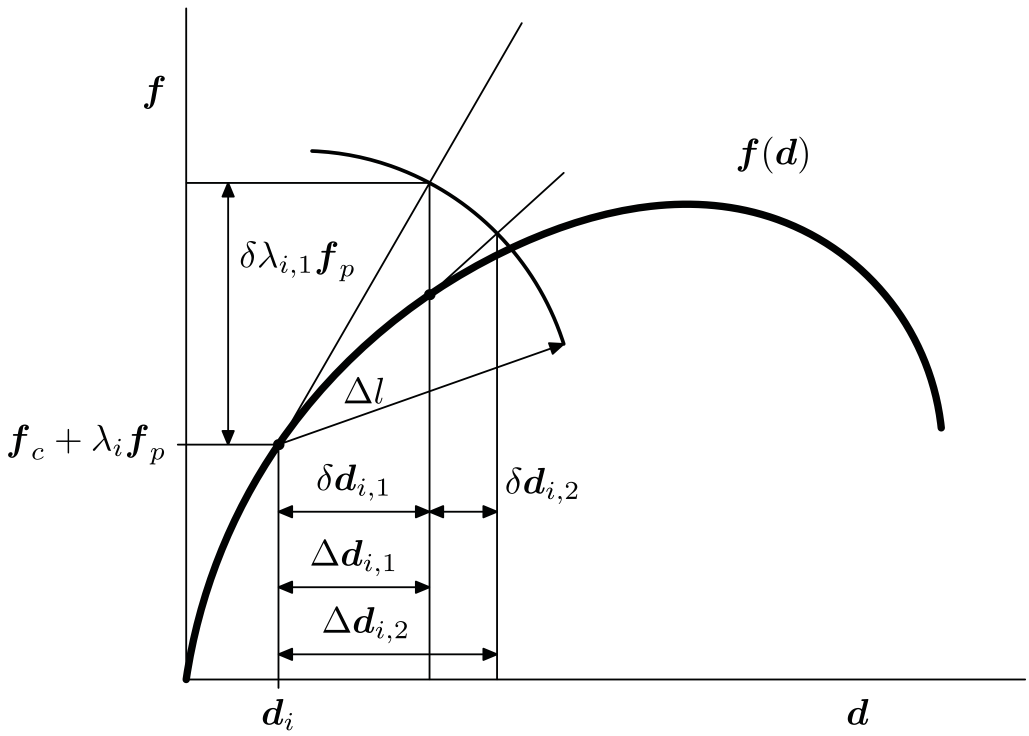
$$\delta\lambda_{i,1} = \pm \frac{\Delta l}{\sqrt{\boldsymbol{v}_{i,1}^T \boldsymbol{v}_{i,1} + \psi^2 \boldsymbol{f}_p^T \boldsymbol{f}_p}}$$

$$\boldsymbol{r}(\boldsymbol{d}_i + \delta \boldsymbol{d}_{i,1}, \lambda_i + \delta\lambda_{i,1}) = \boldsymbol{f}_c + (\lambda_i + \delta\lambda_{i,1}) \boldsymbol{f}_p - \boldsymbol{f}_{int}(\boldsymbol{d}_i + \delta \boldsymbol{d}_{i,1}) \neq \boldsymbol{0}$$

$$\boldsymbol{r}_{i,1} = \boldsymbol{r}(\boldsymbol{d}_i + \delta \boldsymbol{d}_{i,1}, \lambda_i + \delta\lambda_{i,1})$$

$$\boldsymbol{r}(\boldsymbol{d}_{i+1}, \lambda_{i+1}) = \boldsymbol{r}_{i,1} - \boldsymbol{K}_{i,1} \delta \boldsymbol{d}_{i,2} + \boldsymbol{f}_p \delta\lambda_{i,2} = \boldsymbol{0}$$

$$\begin{aligned} \boldsymbol{r}(\boldsymbol{d}_{i+1}, \lambda_{i+1}) &= \boldsymbol{f}_c + (\lambda_i + \delta\lambda_{i,1}) \boldsymbol{f}_p - \boldsymbol{f}_{int}(\boldsymbol{d}_i + \delta \boldsymbol{d}_{i,1}) - \\ &- \boldsymbol{K}_{i,1} \delta \boldsymbol{d}_{i,2} + \boldsymbol{f}_p \delta\lambda_{i,2} = \boldsymbol{0} \end{aligned}$$



$$\begin{aligned}
\Delta\boldsymbol{d}_{i,j} &= \Delta\boldsymbol{d}_{i,j-1} + \delta\boldsymbol{d}_{i,j} \qquad (\Delta\boldsymbol{d}_{i,1} = \delta\boldsymbol{d}_{i,1}) \\
\Delta\lambda_{i,j} &= \Delta\lambda_{i,j-1} + \delta\lambda_{i,j} \qquad (\Delta\lambda_{i,1} = \delta\lambda_{i,1})
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{r}(\boldsymbol{d}_{i+1}, \lambda_{i+1}) &= \boldsymbol{f}_c + (\lambda_i + \delta\lambda_{i,1})\boldsymbol{f}_p - \boldsymbol{f}_{int}(\boldsymbol{d}_i + \delta\boldsymbol{d}_{i,1}) - \\
&- \boldsymbol{K}_{i,1}\delta\boldsymbol{d}_{i,2} + \boldsymbol{f}_p\delta\lambda_{i,2} = \boldsymbol{0}
\end{aligned}$$

$$\boldsymbol{K}_{i,1}\delta\boldsymbol{d}_{i,2} = \boldsymbol{f}_c + (\lambda_i + \Delta\lambda_{i,1})\boldsymbol{f}_p - \boldsymbol{f}_{int}(\boldsymbol{d}_i + \Delta\boldsymbol{d}_{i,1}) + \boldsymbol{f}_p\delta\lambda_{i,2}$$

auxiliary systems of equations

$$\boldsymbol{K}_{i,1}\boldsymbol{u}_{i,2} = \boldsymbol{f}_c + (\lambda_i + \Delta\lambda_{i,1})\boldsymbol{f}_p - \boldsymbol{f}_{int}(\boldsymbol{d}_i + \Delta\boldsymbol{d}_{i,1})$$

$$\boldsymbol{K}_{i,1}\boldsymbol{v}_{i,2} = \boldsymbol{f}_p$$

$$\delta \boldsymbol{d}_{i,2} = \boldsymbol{u}_{i,2} + \delta\lambda_{i,2}\boldsymbol{v}_{i,2}$$

the length of arc

$$\|\Delta \boldsymbol{d}_{i,2} + \boldsymbol{u}_{i,2} + \delta\lambda_{i,2}\boldsymbol{v}_{i,2}\|^2 + \psi^2\|\Delta\lambda_{i,1}\boldsymbol{f}_p + \delta\lambda_{i,2}\boldsymbol{f}_p\|^2 = (\Delta l)^2$$

$$\begin{aligned}
a_1 &= \boldsymbol{v}_{i,2}^T\boldsymbol{v}_{i,2} + \psi^2\boldsymbol{f}_p^T\boldsymbol{f}_p \\
a_2 &= 2\boldsymbol{v}_{i,2}^T(\Delta\boldsymbol{d}_{i,1} + \boldsymbol{u}_{i,2}) + 2\Delta\lambda_{i,1}\psi^2\boldsymbol{f}_p^T\boldsymbol{f}_p \\
a_3 &= (\Delta\boldsymbol{d}_{i,1} + \boldsymbol{u}_{i,2})^T(\Delta\boldsymbol{d}_{i,1} + \boldsymbol{u}_{i,2}) + (\Delta\lambda_{i,1})^2\psi^2\boldsymbol{f}_p^T\boldsymbol{f}_p - (\Delta l)^2
\end{aligned}$$

the length of arc

$$a_1(\delta\lambda_{i,2})^2 + a_2(\delta\lambda_{i,2}) + a_3 = 0$$

CTU Damage, Arc-length Method J. Kruis

Algorithm of the Arc-length Method

$\lambda_0 = 0, \boldsymbol{d}_0 = \boldsymbol{0}$

For $i = 0, 1, 2, \dots$

- $\Delta\lambda_{i,0} = 0, \Delta\boldsymbol{d}_{i,0} = \boldsymbol{0}, \boldsymbol{r}_{i,0} = \boldsymbol{0}$
- For $j = 0, 1, 2, \dots$
- $\boldsymbol{u}_{i,j+1} = \boldsymbol{K}_{i,j}^{-1}\boldsymbol{r}_{i,j}$
- $\boldsymbol{v}_{i,j+1} = \boldsymbol{K}_{i,j}^{-1}\boldsymbol{f}_p$
- $a_1 = \boldsymbol{v}_{i,j+1}^T\boldsymbol{v}_{i,j+1} + \psi^2\boldsymbol{f}_p^T\boldsymbol{f}_p$
- $a_2 = 2\boldsymbol{v}_{i,j+1}^T(\Delta\boldsymbol{d}_{i,j} + \boldsymbol{u}_{i,j+1}) + 2\Delta\lambda_{i,j}\psi^2\boldsymbol{f}_p^T\boldsymbol{f}_p$
- $a_3 = \|\Delta\boldsymbol{d}_{i,j} + \boldsymbol{u}_{i,j+1}\|^2 + (\Delta\lambda_{i,j})^2\psi^2\boldsymbol{f}_p^T\boldsymbol{f}_p - (\Delta l)^2$
- $a_1(\delta\lambda_{i,j+1})^2 + a_2(\delta\lambda_{i,j+1}) + a_3 = 0 \;\Rightarrow\; \delta\lambda_{i,j+1}$
- $\delta\boldsymbol{d}_{i,j+1} = \boldsymbol{u}_{i,j+1} + \delta\lambda_{i,j+1}\boldsymbol{v}_{i,j+1}$
- $\Delta\boldsymbol{d}_{i,j+1} = \Delta\boldsymbol{d}_{i,j} + \delta\boldsymbol{d}_{i,j+1}$
- $\Delta\lambda_{i,j+1} = \Delta\lambda_{i,j} + \delta\lambda_{i,j+1}$
- $\boldsymbol{r}_{i,j+1} = \boldsymbol{f}_c + (\lambda_i + \Delta\lambda_{i,j})\boldsymbol{f}_p - \boldsymbol{f}_{int}(\boldsymbol{d}_i + \Delta\boldsymbol{d}_{i,j})$
- if $\|\boldsymbol{r}_{i,j+1}\| < \varepsilon$, stop

$\lambda_{i+1} = \lambda_i + \Delta\lambda_i$

$\boldsymbol{d}_{i+1} = \boldsymbol{d}_i + \Delta\boldsymbol{d}_i$

Selection of the Roots $\delta\lambda_{i,j+1}$

$$\begin{aligned}\Delta \boldsymbol{d}_{i,j+1} &= \Delta \boldsymbol{d}_{i,j} + \delta \boldsymbol{d}_{i,j+1}\\ \Delta\lambda_{i,j+1} &= \Delta\lambda_{i,j} + \delta\lambda_{i,j+1}\end{aligned}$$

$$\cos\theta = \frac{\Delta \boldsymbol{d}_{i,j+1}^T \Delta \boldsymbol{d}_{i,j}}{(\Delta l)^2} \rightarrow \max$$

$$\cos\theta = \frac{1}{(\Delta l)^2} \Delta \boldsymbol{d}_{i,j}^T (\Delta \boldsymbol{d}_{i,j} + \boldsymbol{u}_{i,j+1} + \delta\lambda_{i,j+1} \boldsymbol{v}_{i,j+1})$$

$$\begin{aligned}a_4 &= \Delta \boldsymbol{d}_{i,j}^T (\Delta \boldsymbol{d}_{i,j} + \boldsymbol{u}_{i,j+1})\\ a_5 &= \Delta \boldsymbol{d}_{i,j}^T \boldsymbol{v}_{i,j+1}\end{aligned}$$

$$\cos\theta = \frac{a_4 + \delta\lambda_{i,j+1} a_5}{(\Delta l)^2}$$

Linearized Arc-length Method

the length of arc

$$l(\Delta \boldsymbol{d}, \Delta\lambda) = \Delta \boldsymbol{d}^T \Delta \boldsymbol{d} + \psi^2 (\Delta\lambda)^2 \boldsymbol{f}_p^T \boldsymbol{f}_p - (\Delta l)^2 = 0$$

$$\frac{\partial l(\Delta \boldsymbol{d}, \Delta\lambda)}{\partial \Delta \boldsymbol{d}} = 2\Delta \boldsymbol{d}^T$$

$$\frac{\partial l(\Delta \boldsymbol{d}, \Delta\lambda)}{\partial \Delta \lambda} = 2\psi^2 \Delta\lambda \boldsymbol{f}_p^T \boldsymbol{f}_p$$

$$l_{i,j} = l(\Delta \boldsymbol{d}_{i,j}, \Delta\lambda_{i,j})$$

$$l_{i,j+1} = l_{i,j} + 2\Delta \boldsymbol{d}_{i,j}^T \delta \boldsymbol{d}_{i,j+1} + 2\psi^2 \Delta\lambda_{i,j} \delta\lambda_{i,j+1} \boldsymbol{f}_p^T \boldsymbol{f}_p = 0$$

$$\begin{aligned} l_{i,j+1} &= l_{i,j} + 2\Delta \boldsymbol{d}_{i,j}^T (\boldsymbol{u}_{i,j+1} + \delta\lambda_{i,j+1} \boldsymbol{v}_{i,j+1}) + \\ &+ 2\psi^2 \Delta\lambda_{i,j} \delta\lambda_{i,j+1} \boldsymbol{f}_p^T \boldsymbol{f}_p = 0 \end{aligned}$$

$$\delta\lambda_{i,j+1} = \frac{-\frac{1}{2} l_{i,j} - \Delta \boldsymbol{d}_{i,j}^T \boldsymbol{u}_{i,j+1}}{\Delta \boldsymbol{d}_{i,j}^T \boldsymbol{v}_{i,j+1} + \psi^2 \Delta\lambda_{i,j} \boldsymbol{f}_p^T \boldsymbol{f}_p}$$

the length of arc

$$l(\Delta\boldsymbol{d},\Delta\lambda) = \Delta\boldsymbol{d}^T\Delta\boldsymbol{d} + \psi^2(\Delta\lambda)^2\boldsymbol{f}_p^T\boldsymbol{f}_p - (\Delta l)^2 = 0$$

$$\begin{aligned} l_{i,j+1} &= l_{i,j} + 2\Delta\boldsymbol{d}_{i,j}^T(\boldsymbol{u}_{i,j+1} + \delta\lambda_{i,j+1}\boldsymbol{v}_{i,j+1}) + \\ &+ 2\psi^2\Delta\lambda_{i,j}\delta\lambda_{i,j+1}\boldsymbol{f}_p^T\boldsymbol{f}_p = 0 \end{aligned}$$

force residual

$$\boldsymbol{r}_{i,j+1} = \boldsymbol{r}_{i,j} + \frac{\partial\boldsymbol{r}_{i,j}}{\partial\boldsymbol{d}}\delta\boldsymbol{d}_{i,j+1} + \frac{\partial\boldsymbol{r}_{i,j}}{\partial\lambda}\delta\lambda_{i,j+1}$$

$$\boldsymbol{r}_{i,j+1} = \boldsymbol{r}_{i,j} - \boldsymbol{K}_{i,j}\delta\boldsymbol{d}_{i,j+1} + \boldsymbol{f}_p\delta\lambda_{i,j+1} = \boldsymbol{0}$$

$$\begin{pmatrix} \boldsymbol{K}_{i,j} & -\boldsymbol{f}_p \\ \Delta\boldsymbol{d}_{i,j}^T & 2\psi^2\Delta\lambda_{i,j}\boldsymbol{f}_p^T\boldsymbol{f}_p \end{pmatrix} \begin{pmatrix} \delta\boldsymbol{d}_{i,j+1} \\ \delta\lambda_{i,j+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{r}_{i,j} \\ -l_{i,j} \end{pmatrix}$$

Conclusions

- isotropic, orthotropic, anisotropic damage model
- mesh adjusted softening modulus
- spherical, cylindrical, linearized arc-length methods
- efficient implementation

Optimal QP algorithms and scalable algorithms for nonlinear problems of mechanics

Zdeněk Dostál
and colleagues

3.12.2010
CM-II
Ostrava

With

Tomáš Kozubek
Alex Markopoulos
Tomáš Brzobohatý
Marie Sadovská
Vít Vondrák
Jiří Bouchala
Petr Vodstrčil
…

Outline

1. Motivation, optimal algorithms
2. TFETI/TBETI for contact problems as application of duality
3. Linear and subsymmetric separable QPQC (SMALSE)
4. Optimal algorithms for bound constrained QP (MPEGP)
5. Application to the contact problems

Scalable algorithms

An algorithm is *numericaly scalable* if

the cost of the solution ~ number of unknowns

(for unconstrained QP problems multigrid – Fedorenko 60s, FETI Farhat and Roux 90s)

An algorithm enjoys *parallel scalability* if

the time of the solution ~ 1/number of processors

(for unconstrained QP problems Farhat and Roux FETI 1991)

Challenges

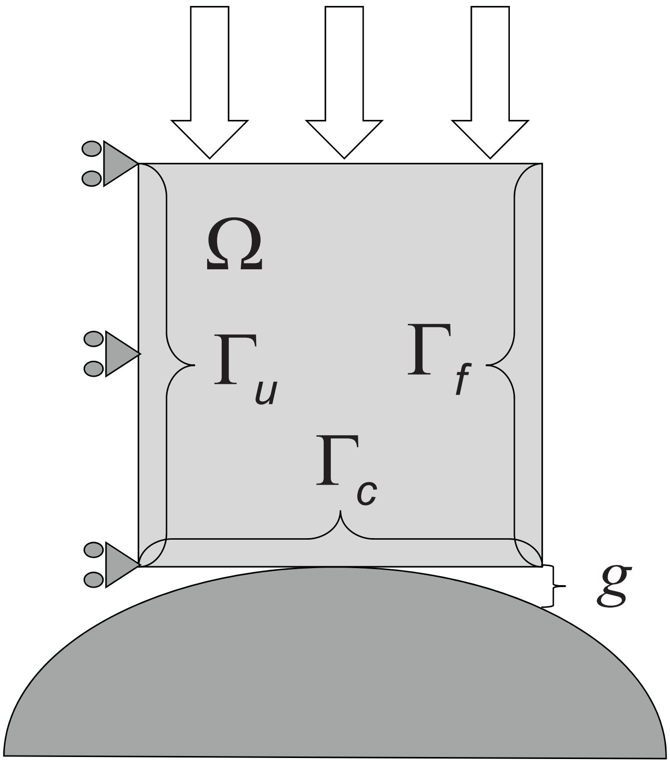
- Identify the active constraints for free
- Get rate of convergence independent of conditioning of constraints
- Use only preconditioners that preserve bound constraints
- Get an initial approximation which is near the solution, i.e.

$$\left\|\widehat{\mathbf{u}} - \mathbf{u}^0\right\| \le C\left\|\mathbf{f}\right\|, \ \mathbf{u}^0 \text{ feasible}$$

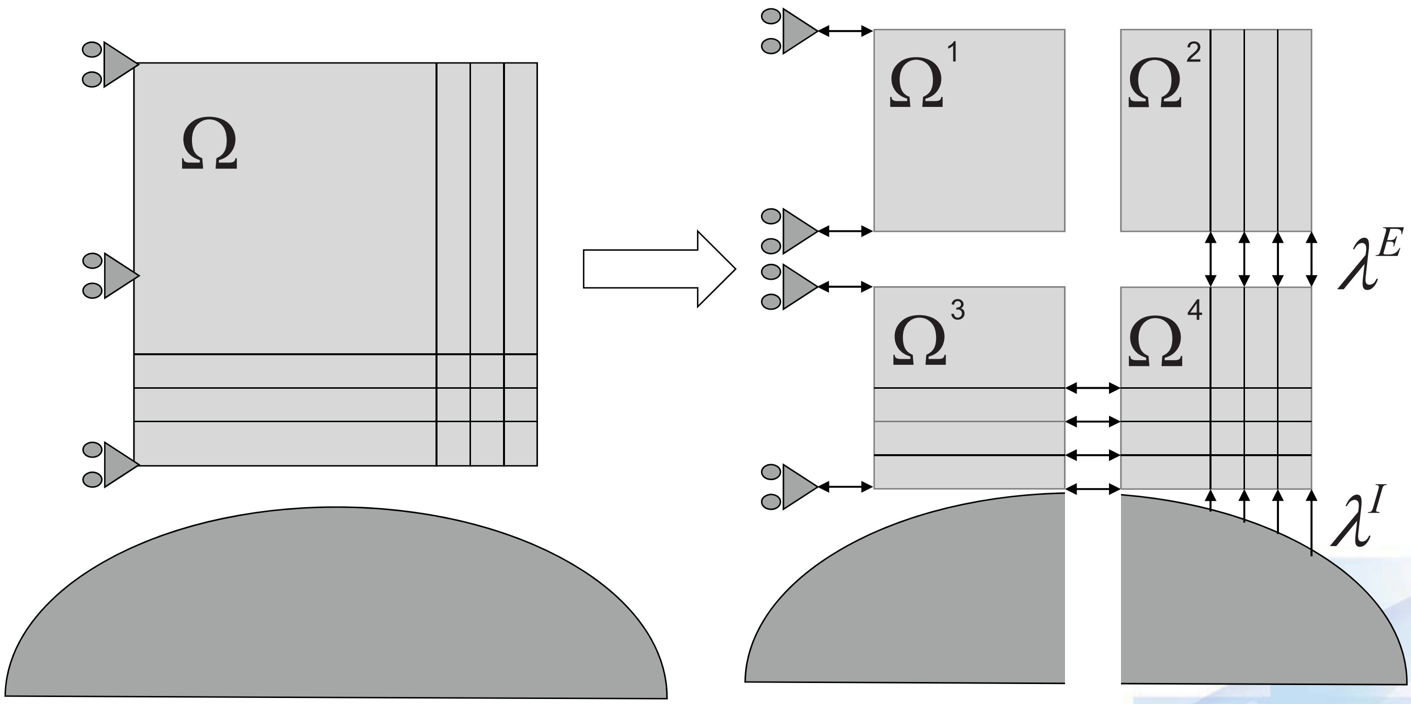
Key observation

- In dual, there is a well defined subspace with the solution that can be used as a coarse grid

Contact problem

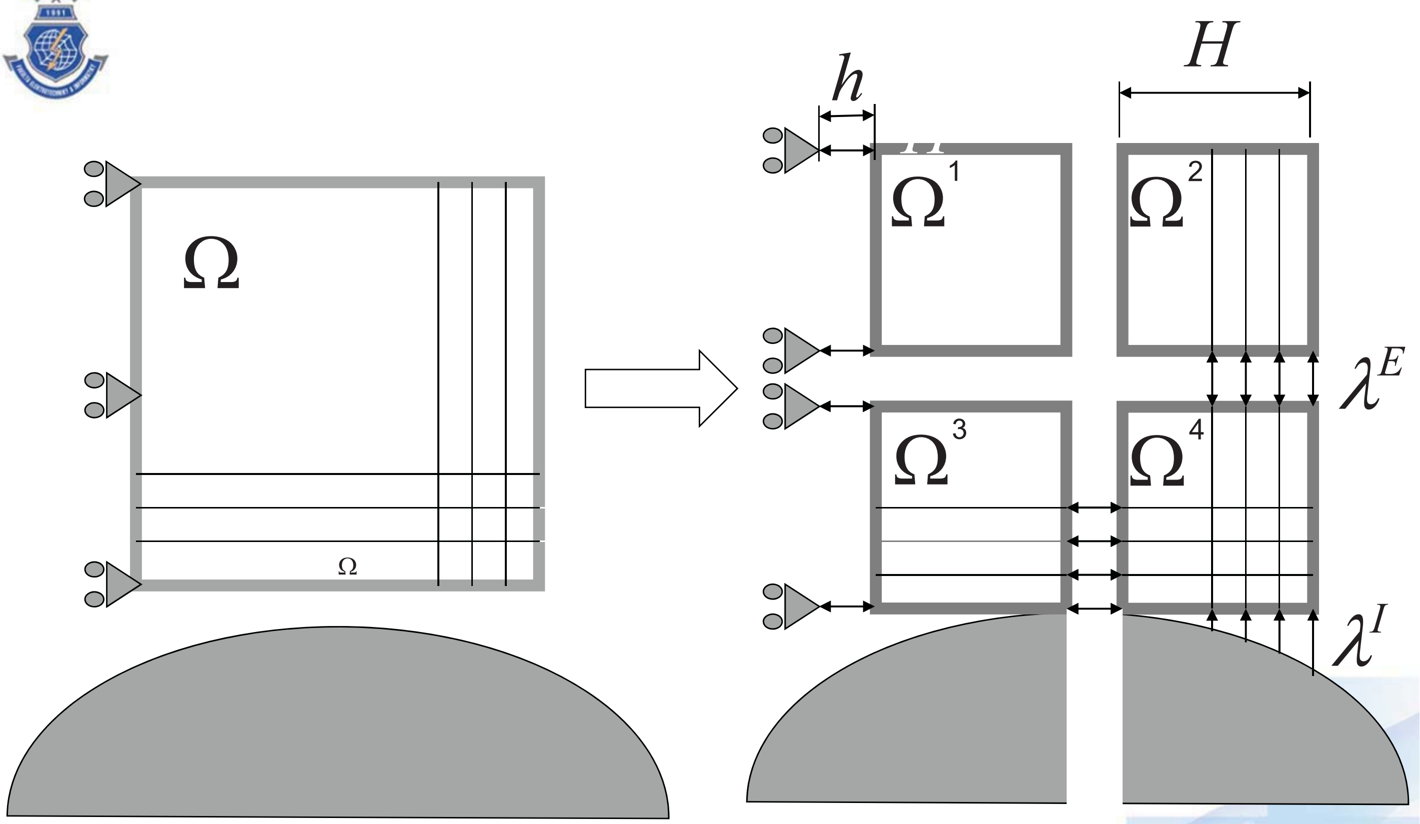


TFETI (AF FETI) domain decomposition



Z.D., Horák, Kučera CNME 2006, Of Thesis 2006

TBETI (AF BETI) domain decomposition



Linear problems Langer and Steinbach Computing 2003
Variational inequalities Bouchala, Z.D., Sadowská Computing 2008, 2009

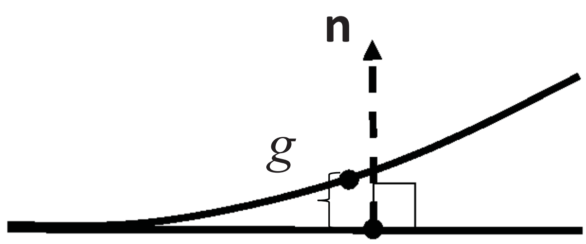
Stiffness matrices TFETI

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}^1 & & \\ & \ddots & \\ & & \mathbf{K}^s \end{bmatrix} \quad \text{(positive semidefinite)} \qquad \mathbf{f} = \begin{bmatrix} \mathbf{f}^1 \\ \vdots \\ \mathbf{f}^s \end{bmatrix}$$

$$\operatorname{Ker} \mathbf{K}^j = \begin{bmatrix} y_i & 1 & 0 \\ -x_i & 0 & 1 \end{bmatrix} \quad \text{(in 2D)}$$

$$\operatorname{Ker} \mathbf{K}^j = \begin{bmatrix} 0 & -z_i & y_i & 1 & 0 & 0 \\ z_i & 0 & -x_i & 0 & 1 & 0 \\ -y_i & x_i & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(in 3D)}$$

Discretized non-penetration



$$\mathbf{B}^I = \left[\cdots (\mathbf{n}^i)^T \cdots (\mathbf{n}^k)^T \cdots\right], \quad \mathbf{n}^i = -\mathbf{n}^k$$

$$\Rightarrow \quad \text{non-penetration}: \ \mathbf{B}^I\mathbf{u} \le \mathbf{g}^I$$

Discretized frictionless problem

$$J(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T\mathbf{K}\mathbf{u} - \mathbf{f}^T\mathbf{u} \qquad \text{(convex)}$$

$$\text{gluing}: \ \mathbf{B}^E\mathbf{u} = \mathbf{o}$$

$$\text{non-penetration}: \ \mathbf{B}^I\mathbf{u} \le \mathbf{g}$$

$$\mathcal{K}_h = \left\{\mathbf{u}: \ \mathbf{B}^E\mathbf{u} = \mathbf{o} \quad \text{and} \quad \mathbf{B}^I\mathbf{u} \le \mathbf{g}\right\}$$

$$\boxed{(\mathrm{P}_h) \quad \text{Find} \qquad \min J_h(\mathbf{u}) \qquad \text{s.t.} \quad \mathbf{u} \in \mathcal{K}_h}$$

Dual formulation

Convexity of $L(\,.\,,\lambda)$ and gradient argument :

$\nabla L(\,.\,,\lambda) = \mathbf{Ku} - \mathbf{f} + \mathbf{B}^T\lambda \;= \mathbf{o}$

$\mathbf{R}$ full rank matrix, $\mathrm{Im}\,\mathbf{R} = \mathrm{Ker}\,\mathbf{K}$

Solvable for $\mathbf{f} - \mathbf{B}^T\lambda \in \mathrm{Im}\,\mathbf{K} \;\Leftrightarrow\; \mathbf{R}^T\left(\mathbf{f} - \mathbf{B}^T\lambda\right) = \mathbf{o}$

$\mathbf{K}^+$ generalized inverse $\mathbf{KK}^+\mathbf{K} = \mathbf{K}$

$$\boxed{\begin{array}{ll} (\mathrm{D}_h)\ \text{Find} & \min\ \lambda^T\mathbf{BK}^+\mathbf{B}\lambda - \lambda^T\left(\mathbf{BK}^+\mathbf{f} - \mathbf{c}\right) \\ & \text{s.t.}\quad \lambda^I \geq \mathbf{o},\quad \mathbf{R}^T\left(\mathbf{f} - \mathbf{B}^T\lambda\right) = \mathbf{o} \end{array}}$$

FETI notation and homogenization

Notation :

$\mathbf{F} = \mathbf{BK}^+\mathbf{B}^T$ $\qquad\qquad$ $\mathbf{G} = \mathbf{R}^T\mathbf{B}^T$

$\hat{\mathbf{d}} = \mathbf{BK}^+\mathbf{f} - \mathbf{c}$ $\qquad\qquad$ $\mathbf{e} = \mathbf{R}^T\mathbf{f}$

$$\boxed{\begin{array}{l} \frac{1}{2}\lambda^T\mathbf{F}\lambda - \lambda^T\hat{\mathbf{d}} \rightarrow \min \\ \text{s.t.}\quad \lambda^I \geq \mathbf{o} \quad \text{and} \quad \mathbf{G}\lambda = \mathbf{e} \end{array}}$$

Homogenization :

$\mathbf{G}\overline{\lambda} = \mathbf{e}$ $\qquad\qquad$ $\lambda = \mu + \overline{\lambda}$

$\mathbf{G}\lambda = \mathbf{e} \quad \Leftrightarrow \quad \mathbf{G}\mu = \mathbf{o}$

$\lambda^I \geq \mathbf{o} \quad \Leftrightarrow \quad \mu^I \geq -\overline{\lambda}^I$

$$\boxed{\begin{array}{ll} (\mathrm{FETI}_h) & \frac{1}{2}\lambda^T\mathbf{F}\lambda - \lambda^T\mathbf{d} \rightarrow \min \\ & \text{s.t.}\quad \lambda^I \geq -\overline{\lambda}^I \quad \text{and} \quad \mathbf{G}\lambda = \mathbf{o} \end{array}}$$

More on homogenization

How to choose $\bar{\lambda}$ so that we are able to find $\lambda^0 \geq \bar{\lambda}$ so that

$\|\lambda^0 - \hat{\lambda}\| \leq C\|\mathbf{d}\|$, C independent of h and H ?

(i) Lemma : If the problem is coercive, then there is $\bar{\lambda}$ such that $\lambda = \mathbf{o}$ satisfies $\lambda^I \geq -\bar{\lambda}^I$

(ii) Use $\bar{\lambda}$ which solves $\min \frac{1}{2}\|\lambda\|^2$ s.t. $\lambda^I \geq \mathbf{o}$ and $\mathbf{G}\lambda = \mathbf{e}$

In our experiments $\bar{\lambda} = \mathbf{G}^T(\mathbf{GG})^{-1}\mathbf{e}$

Natural coarse grid projectors

$$\mathbf{Q} = \mathbf{G}^T\left(\mathbf{GG}^T\right)^{-1}\mathbf{G} \qquad \mathbf{P} = \mathbf{I} - \mathbf{Q}$$

$$\operatorname{Im}\mathbf{Q} = \operatorname{Im}\mathbf{G}^T \qquad \operatorname{Im}\mathbf{P} = \operatorname{Ker}\mathbf{G}$$

$$(\text{FETI-NCG}_h) \quad \frac{1}{2}\lambda^T\mathbf{PFP}\lambda - \lambda^T\mathbf{Pd} + \frac{\rho}{2}\lambda^T\mathbf{Q}\lambda \quad \to \quad \min$$

$$\text{s.t. } \lambda^I \geq -\bar{\lambda}^I \quad \text{and} \quad \mathbf{G}\lambda = \mathbf{o}$$

$\rho \approx \|\mathbf{F}\|$

Optimal estimates

Theorem : Let there be positive constants B_1, B_2 such that for any discretization parameter h and H

$$B_1 \leq \lambda_{\min}\left(\mathbf{B}\mathbf{B}^T\right) \leq \lambda_{\min}\left(\mathbf{B}\mathbf{B}^T\right) \leq B_2$$

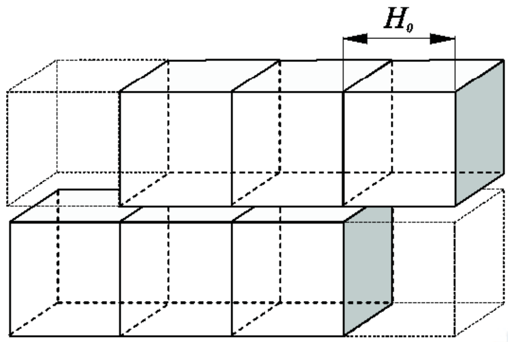
Let the elements and subdomains have regular shape and size.

Then

$$C_1 \leq \lambda_{\min}(\mathbf{F} \,|\, \mathrm{Im}\mathbf{F})$$

$$\|\mathbf{F}\| \leq C_2 \frac{H}{h}$$

$$\kappa(\mathbf{F} \,|\, \mathrm{Im}\mathbf{F}) \leq C_2 \frac{H}{h}$$



ETI: Farhat, Mandel, Roux 1994

ETI: Bouchala, Z.D., Sadowská 2009, based on Langer and Steinbach 2003

Discretized Tresca problem

$$J_h(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T\mathbf{K}\mathbf{u} - \mathbf{f}^T\mathbf{u} + j_h(\mathbf{u}) \quad \text{(convex)}$$

$$j_h(\mathbf{u}) = \sum_{i=1}^{m} \psi_i \|\mathbf{T}_i\mathbf{u}\| \quad \text{(non - differentiable)}$$

gluing : $\mathbf{B}^E\mathbf{u} = \mathbf{o}$

non - penetration : $\mathbf{B}^I\mathbf{u} \leq \mathbf{g}$

$$\mathcal{K}_h = \left\{\mathbf{u} : \ \mathbf{B}^E\mathbf{u} = \mathbf{o} \quad \text{and} \quad \mathbf{B}^I\mathbf{u} \leq \mathbf{g}\right\}$$

$$\boxed{(\mathrm{P}_h) \quad \text{Find} \quad \min J_h(\mathbf{u}) \quad \text{s.t.} \quad \mathbf{u} \in \mathcal{K}_h}$$

Dissipative terms in 2D and 3D

2D :

$$j_h(\mathbf{u}) = \sum_{i=1}^{m} \psi_i \|\mathbf{T}_i \mathbf{u}\| = \sum_{i=1}^{m} \max_{|\tau_i| \le \psi_i} \tau_i \mathbf{T}_i \mathbf{u}$$

3D :

$$j_h(\mathbf{u}) = \sum_{i=1}^{m} \psi_i \|\mathbf{T}_i \mathbf{u}\| = \sum_{i=1}^{m} \max_{\|\tau_i \le \psi_i\|} \tau_i \mathbf{T}_i \mathbf{u}$$

Mixed formulation (3D)

$$L_h(\mathbf{u}, \lambda) = \frac{1}{2}\mathbf{u}^T \mathbf{K} \mathbf{u} - \mathbf{u}^T \mathbf{f} + \lambda_N^T (\mathbf{B}^I \mathbf{u} - \mathbf{c}) + \lambda_E^T \mathbf{B}^E \mathbf{u} + \sum_{i=1}^{m} \tau_i^T \mathbf{T}_i \mathbf{u}$$

$$= J_h(\mathbf{u}) + \lambda^T \mathbf{B} \mathbf{u}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}^N \\ \mathbf{T} \\ \mathbf{B}^E \end{bmatrix}, \qquad \lambda = \begin{bmatrix} \lambda_N \\ \tau \\ \lambda_E \end{bmatrix}, \quad \Lambda = \Lambda(\psi) = \{\lambda : \lambda_N \ge \mathbf{o} \quad \text{and} \quad \|\tau_i\| \le \psi_i\}$$

$$\boxed{(\mathrm{M}_h) \quad \text{Find} \quad \min_{\mathbf{u}} \max_{\lambda \in \Lambda} L_h(\mathbf{u}, \lambda) = \max_{\lambda \in \Lambda} \min_{\mathbf{u}} L_h(\mathbf{u}, \lambda)}$$

2 sets of variables

differentiable problem

Haslinger, Kučera, Z.D., JCAM 2004

Duality and natural coarse grid projectors

$$\mathbf{Q} = \mathbf{G}^T\left(\mathbf{G}\mathbf{G}^T\right)^{-1}\mathbf{G} \qquad \mathbf{P} = \mathbf{I} - \mathbf{Q}$$

$$\operatorname{Im}\mathbf{Q} = \operatorname{Im}\mathbf{G}^T \qquad \operatorname{Im}\mathbf{P} = \operatorname{Ker}\mathbf{G}$$

$$\boxed{\begin{array}{lll} (\text{TFETI-NCG}_h) & \frac{1}{2}\lambda^T\mathbf{PFP}\lambda - \lambda^T\mathbf{Pd} \quad \rightarrow & \min \\ & \text{s.t. } \lambda \in \Lambda \quad \text{and} & \mathbf{G}\lambda = \mathbf{o} \end{array}}$$

Bound and equality constrained problems

For $i \in \mathcal{T}$ let

$$f_i(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}_i\mathbf{x} - \mathbf{b}_i^T\mathbf{x}$$

$$\Omega_i = \{\mathbf{x} : \mathbf{x} \geq \mathbf{c}_i \text{ and } \mathbf{B}_i\mathbf{x} = \mathbf{o}\}, \quad \|\mathbf{B}_i\| \leq C_0$$

$$\mathbf{A}_i = \mathbf{A}_i^T$$

$$C_1\|\mathbf{x}\|^2 \leq \mathbf{x}^T\mathbf{A}_i\mathbf{x}_i \leq C_2\|\mathbf{x}\|^2, \quad \mathbf{o} \in \Omega_i$$

$$(\text{QPBE}_i) \qquad \text{Find} \qquad \min_{\Omega i} f_i(\mathbf{x})$$

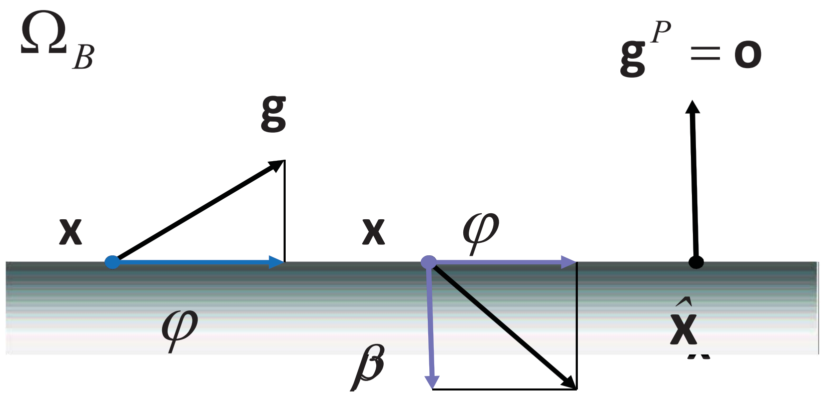
Challenge: Find an approximate solution in O(1) iterations!!!

Augmented Lagrangian and gradient

$$L(\mathbf{x},\mu,\rho)=f(\mathbf{x})+\mu^T\mathbf{Bx}+\frac{1}{2}\|\mathbf{Bx}\|^2$$

$$\mathbf{g}(\mathbf{x},\mu,\rho)=\nabla_{\mathbf{x}}L(\mathbf{x},\mu,\rho)$$

$$\mathbf{g}^P=\mathbf{g}^P(\mathbf{x},\mu,\rho)=\varphi(\mathbf{x},\mu,\rho)+\beta(\mathbf{x},\mu,\rho)$$



$$\mathbf{g}^P=\varphi+\beta$$

Algorithm SMALBE-M

Step 0 $\quad \beta<1,\ \rho>0,\ M_0>0,\ \eta>0,\ \mu^0$

$\{$Approximate solution of bound constrained problem$\}$

Step 1 $\quad$ Find $\mathbf{x}^k$ such that $\boxed{\left\|\mathbf{g}^P(\mathbf{x}^k,\mu^k,\rho)\right\|\le\min\{M_k\|\mathbf{Bx}^k\|,\eta\}}$

$\{$Test$\}$

Step 2 $\quad$ if $\left\|\mathbf{g}^P(\mathbf{x}^k,\mu^k,\rho)\right\|$ and $\|\mathbf{Bx}^k\|$ are small then $\mathbf{x}^k$ is solution

$\{$Update Lagrange multipliers$\}$

Step 3 $\quad \mu^{k+1}=\mu^k+\rho\mathbf{Bx}^k$

$\{$Update $M_k\}$

Step 4 $\quad$ If $L(\mathbf{x}^k,\mu^k,\rho)\le L(\mathbf{x}^{k-1},\mu^{k-1},\rho)+\frac{\rho}{2}\|\mathbf{Bx}^k\|$

then $\quad M_{k+1}=\beta M_k$

else $\quad M_{k+1}=M_k$

Step 5 $\quad k=k+1$ and return to *Step* 1

Basic relations for SMALBE

Theorem: Let $\mathbf{x}^k, \mu^k, \rho$ be generated by SMALBE-M. Then

(i) If $\rho \geq M_k^2 / \lambda_{\min}(\mathbf{A})$ then

$$\boxed{L(\mathbf{x}^k, \mu^k, \rho) \geq L(\mathbf{x}^{k-1}, \mu^{k-1}, \rho) + \frac{\rho}{2}\|\mathbf{B}\mathbf{x}^k\|^2}$$

(ii) There is C such that

$$\boxed{\sum_{k=1}^{\infty} \frac{\rho}{2}\|\mathbf{B}\mathbf{x}^k\|^2 \leq C}$$

Z.D. SINUM 2005, Computing 2006

Optimality of SMALBE

Corollary: Let $\mathbf{x}^k, \mu^k, M_k$ be generated by SMALBE-M, $\varepsilon > 0$.

Then:

(i) $$\boxed{M_k^2 \geq \min\{M_0^2, \rho\,\lambda_{\min}(\mathbf{A})\}}$$

(ii) SMALBE-M generates $\mathbf{x}^k$ that satisfies

$$\boxed{\|\mathbf{g}_i^P(\mathbf{x}^k)\| \leq \varepsilon\|\mathbf{b}_i\| \quad \text{and} \quad \|\mathbf{B}_i\mathbf{x}^k\| \leq \varepsilon\|\mathbf{b}_i\|}$$

at O(1) outer iterations

Z.D. SINUM 2005, Z.D. Computing 2006, Z.D. book 2009

Bound constrained problems

For $i \in \mathcal{T}$ let

$$f_i(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}_i\mathbf{x} - \mathbf{b}_i^T\mathbf{x}$$

$$\Omega_i = \{\mathbf{x} : \mathbf{x} \geq \mathbf{c}_i\}, \quad \|\mathbf{c}_i^+\| \leq C_0$$

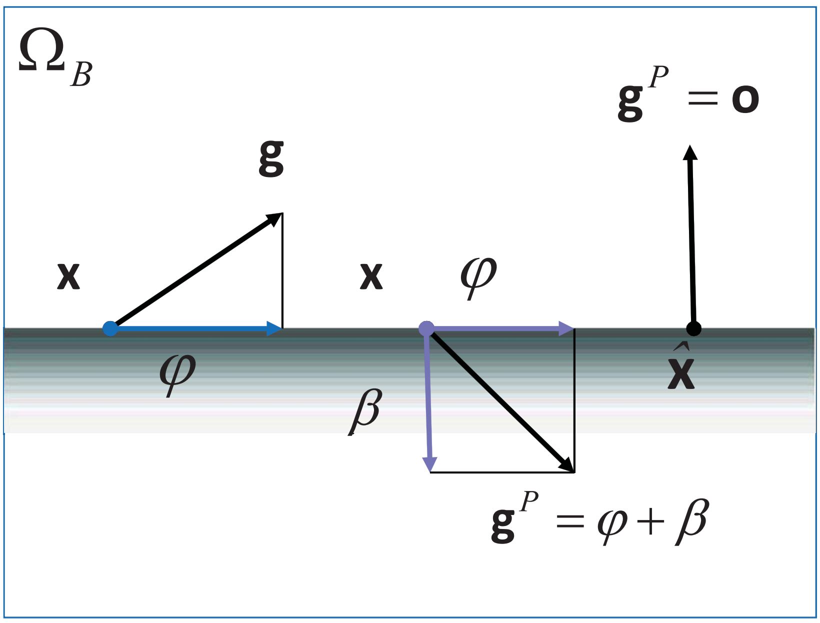
$$\mathbf{A}_i = \mathbf{A}_i^T$$

$$C_1\|\mathbf{x}\|^2 \leq \mathbf{x}^T\mathbf{A}_i\mathbf{x}_i \leq C_2\|\mathbf{x}\|^2,$$

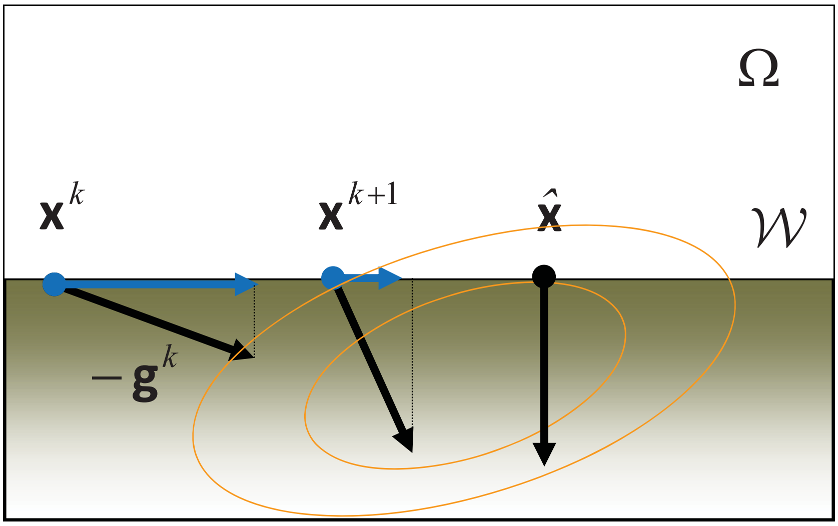
$$(\mathrm{QPB}_i) \qquad \text{Find} \qquad \min_{\Omega_i} f_i(\mathbf{x})$$

Challenge : Find an approximate solution in O(1) iterations!!!

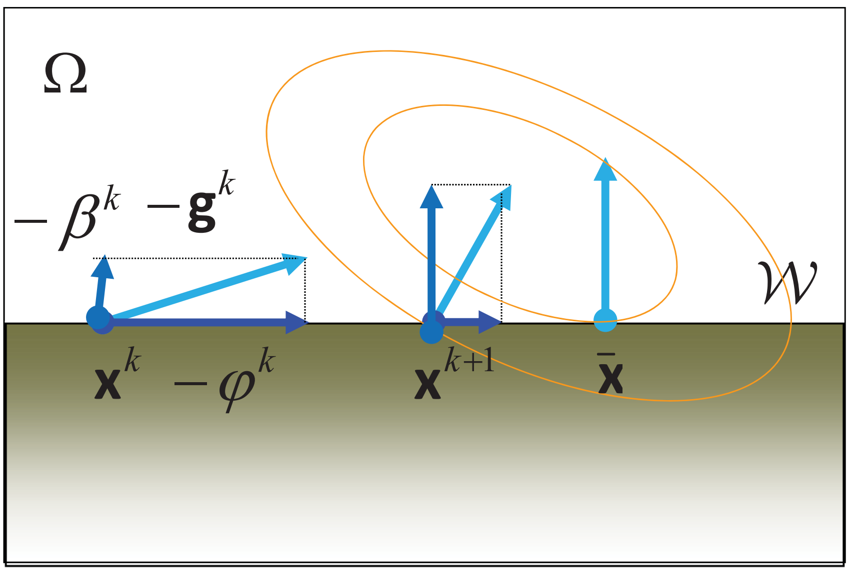
Bound constraints: Splitting of the gradient and KKT



Active set strategy in the face with the solution



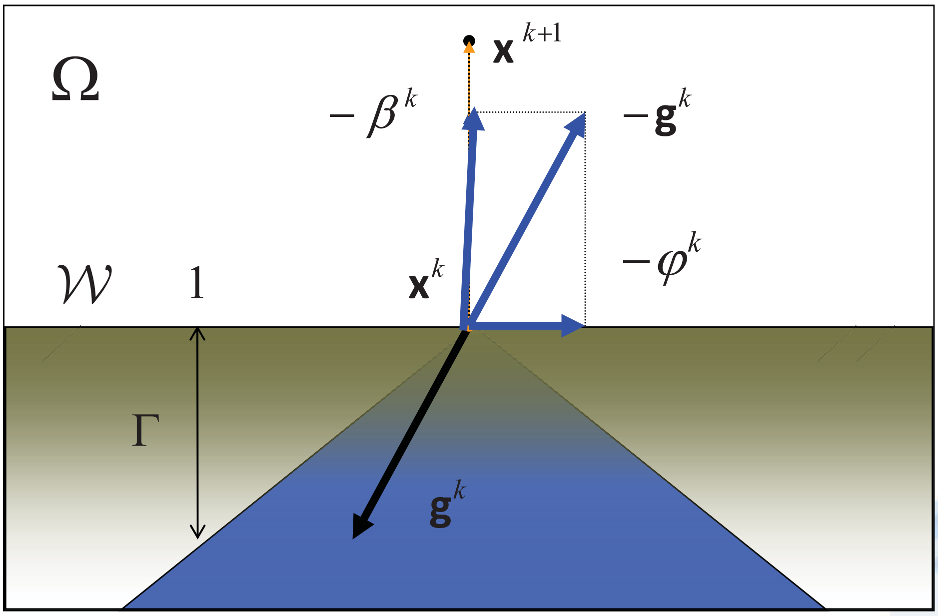
When to leave the face?



Proportioning

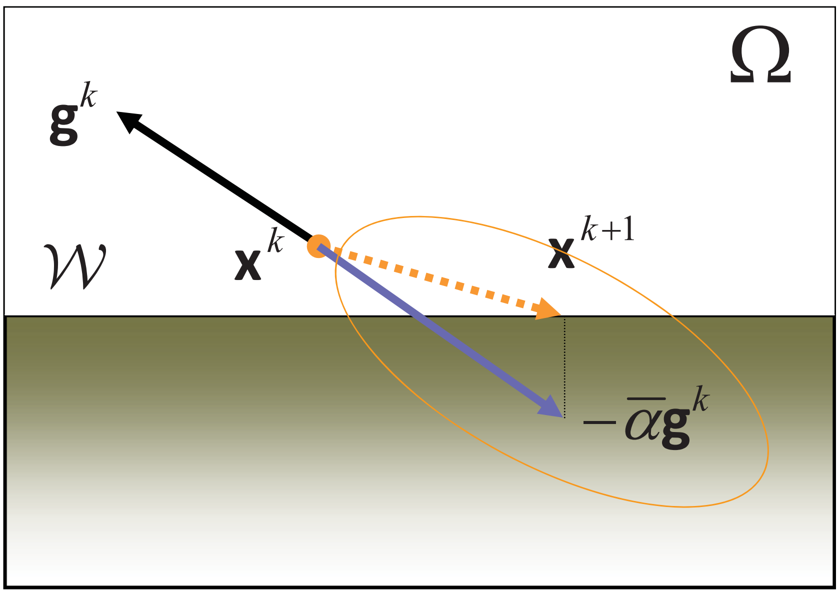
x not proportional: $\boxed{\Gamma^2 \widetilde{\varphi}^T(x)\varphi(x) \le \|\beta(x)\|^2}$

Reduction of the active set for non-proportional iterations



How to expand the face? Reduced gradient projection with superrelaxation

$$\mathbf{x}^{k+1} = P_\Omega\left(\mathbf{x}^k - \overline{\alpha}\varphi^k\right), \quad \overline{\alpha} \in \left(0, 2\|\mathbf{A}\|^{-1}\right)$$



Effect of reduced gradient projection

$$\mathbf{x}^{k+1} = P_{\Omega_S}\left(\mathbf{x}^k - \overline{\alpha}\varphi^k\right),\ \text{proportional}$$

$$\alpha \in \left(0, \|A\|^{-1}\right]\ \text{Schöberl, 1998:}$$

$$\boxed{f\left(\mathbf{x}^{k+1}\right) - f(\hat{\mathbf{x}}) \le \left(1 - \frac{\alpha\lambda_{\min}}{2+2\Gamma^2}\right)\left(f\left(\mathbf{x}^k\right) - f(\hat{\mathbf{x}})\right)}$$

Superrelaxation,

$$\hat{\alpha} = \min\left\{\alpha, 2\|A\|^{-1} - \alpha\right\},\ \alpha \in \left(0, \|A\|^{-1}\right],\ \text{Z.D., 2008:}$$

$$\boxed{f\left(\mathbf{x}^{k+1}\right) - f(\hat{\mathbf{x}}) \le \left(1 - \frac{\hat{\alpha}\lambda_{\min}}{2+2\Gamma^2}\right)\left(f\left(\mathbf{x}^k\right) - f(\hat{\mathbf{x}})\right)}$$

Conjugate gradient step

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_{\mathrm{cg}}\mathbf{p}_j, \quad \mathbf{p}_0 = \varphi_i\left(\mathbf{x}^k\right)$$

If $\left\|\varphi(\mathbf{x}^k)\right\| \ge \Gamma\left\|\beta(\mathbf{x}^k)\right\|, \quad \Gamma > 0$

then

$$\boxed{f\left(\mathbf{x}^{k+1}\right) - f(\hat{\mathbf{x}}) \le \left(1 - \frac{\hat{\alpha}\lambda_{\min}}{2+2\Gamma^2}\right)\left(f\left(\mathbf{x}^k\right) - f(\hat{\mathbf{x}})\right)}$$

Algorithm MPRGP

$\{\text{Initialization}\}$

Step 0 $\quad \mathbf{x}^0 \geq \mathbf{c}, \Gamma > 0, \overline{\alpha} \in (0, 2\|\mathbf{A}\|^{-1}]$

$\{\text{Proportioning}\}$

Step 1 $\quad$ If

$$\boxed{\Gamma^2 \widetilde{\varphi}\left(\mathbf{x}^k\right)^T \varphi\left(\mathbf{x}^k\right) < \left\|\beta\left(\mathbf{x}^k\right)\right\|^2}$$

then define $\mathbf{x}^{k+1}$ by minimization in the direction $-\beta\left(\mathbf{x}^k\right)$

$\{\text{Conjugate gradient step}\}$

Step 2 $\quad$ if $\mathbf{x}^k$ is proportional, then generate $\mathbf{x}^{k+1}$ by trial CG step

$\{\text{Projection}\}$

Step 3 $\quad$ If $\mathbf{x}^{k+1} \geq \mathbf{c}$, then use it,

else $\mathbf{x}^{k+1} = \left(\mathbf{x}^k - \overline{\alpha}\varphi\left(\mathbf{x}^k\right)\right)^+$

Z.D., Schöberl COA 2005, book 2009

Rate of convergence of MPRGP

Theorem: Let $\mathbf{x}^k$ be generated with $\Gamma > 0$, $\hat{\Gamma} = \max\{\Gamma, \Gamma^{-1}\}$, $\overline{\alpha} \in \left(0, 2\|\mathbf{A}\|^{-1}\right)$

Then:

(i) The R-linear rate of convergence in the energy norm is given by

$$\boxed{\left\|\mathbf{x}^k - \hat{\mathbf{x}}\right\|_{\mathbf{A}} \leq \eta^k \left(f\left(\mathbf{x}^0\right) - f(\hat{\mathbf{x}})\right)} \quad \text{with} \quad \eta = 1 - \frac{\overline{\alpha}\lambda_{\min}}{2 + 2\hat{\Gamma}^2}$$

(ii) The R-linear rate of convergence of the projected gradient is given by

$$\boxed{\left\|\mathbf{g}^P\left(\mathbf{x}^k\right)\right\| \leq 2a\eta^k \left(f\left(\mathbf{x}^0\right) - f(\hat{\mathbf{x}})\right)} \quad \text{with} \quad a = \frac{\overline{\alpha}^{-1}\lambda_{\min}^{-1}}{\eta(1-\eta)}$$

Z.D., Schoeberl COA 2005, Z.D. book 2009

Optimality of MPRGP

Theorem :

Let $\Gamma > 0,\ 0 < \alpha < 2C_2^{-1},\ \bar{\mathbf{x}}_i$ solution of (QPB_i)

For $i \in \mathcal{T}$ let $\{\mathbf{x}_i^k\}$ be generated by MPRGP

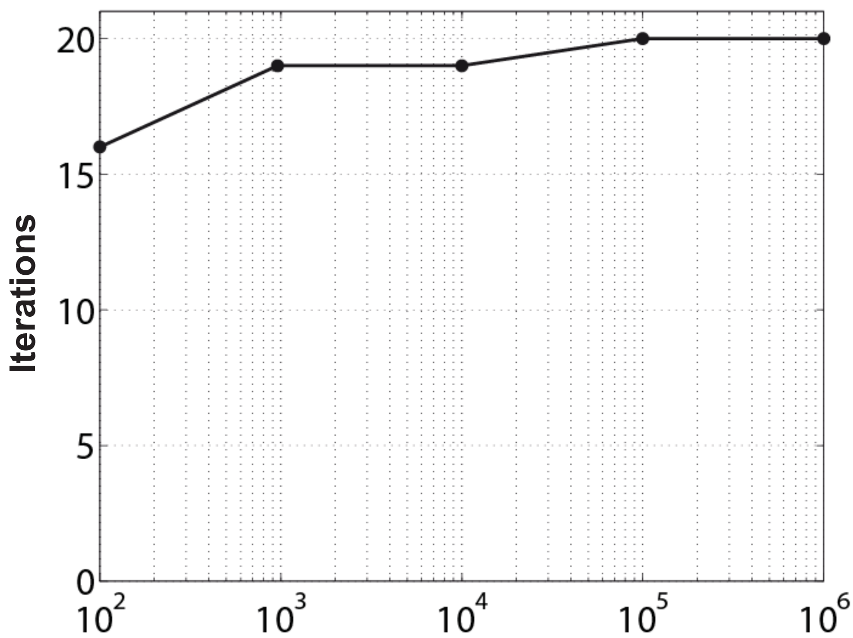
with α and Γ. Then $\mathbf{x}_i^k$ that satisfy

$$\boxed{\left\|\mathbf{x}_i^k - \bar{\mathbf{x}}_i\right\| \le \varepsilon \left\|\mathbf{b}_i\right\| \text{ and } \left\|\mathbf{g}^P\left(\mathbf{x}_i^k\right)\right\| \le \varepsilon \left\|\mathbf{b}_i\right\|}$$

can be find in O(1) iterations

Optimality for bound constrained problems

String system on Winkler support, bound constraints, cond=5



Optimality results SMABE/MPRGP

Theorem : Let $\hat{\mathbf{x}}_i$ be the solution of (QPBE$_\mathrm{i}$) generated with $\Gamma > 0$,
$\overline{\chi} \in \left(0, 2C_2^{-1}\right)$, $\rho > 0$, $M > 0$. Let $\varepsilon > 0$.
Then $\mathbf{x}_i^k$ that satisfies

$$\boxed{\left\|\mathbf{x}_i^k - \hat{\mathbf{x}}_i\right\| \le \varepsilon\|\mathbf{b}\| \quad \text{and} \quad \left\|\mathbf{g}^P\left(\mathbf{x}_i^k\right)\right\| \le \varepsilon\|\mathbf{b}\|}$$

s found at

$$\boxed{\mathrm{O}(1) \text{ matrix - vector multiplications}}$$

For bound constrained problems Z.D. Computing 2006, Z.D. book 2009,
application to scalar problems TFETI Z.D., Horák SINUM 2007
application to scalar problems TBETI Bouchal, Z.D., Sadowská Computing 2008
Generalization to separable subsymmetric constraints Z.D., Kozubek 2010

Optimality of TFETI with SMALBE/MPRGP

Theorem : Let $\hat{\mathbf{x}}_h$ be the solution of (TFETI - NCG$_h$) and $\varepsilon > 0$.

Then $\mathbf{x}_h^k$ that satisfies

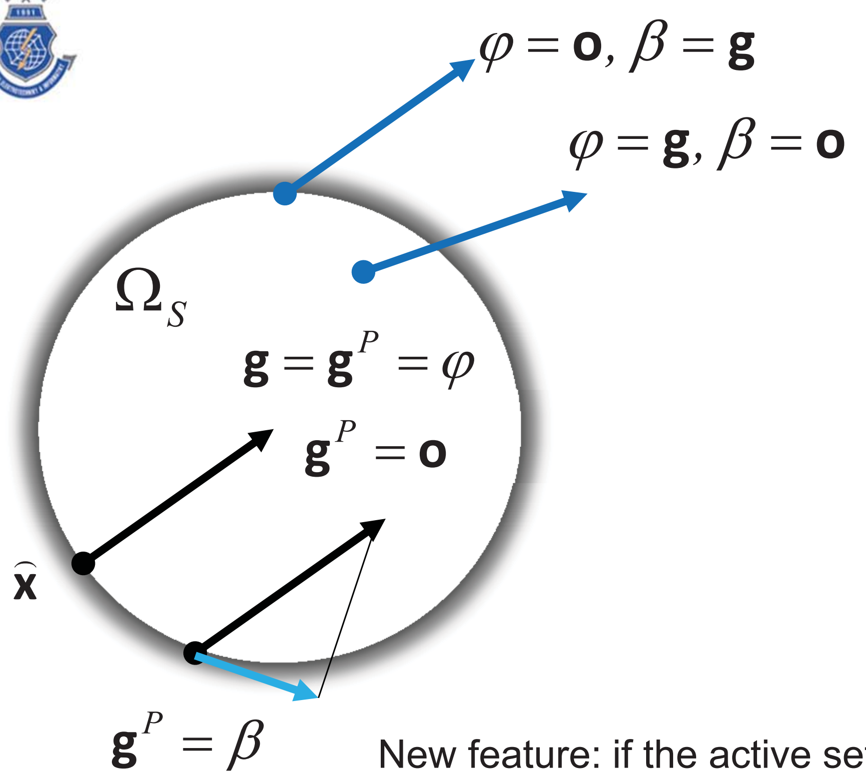
$$\boxed{\left\|\mathbf{B}_h\mathbf{x}_h^k\right\| \le \varepsilon\left\|\mathbf{b}_h\right\| \quad \text{and} \quad \left\|\mathbf{g}^P\left(\mathbf{x}_h^k\right)\right\| \le \varepsilon\left\|\mathbf{b}_h\right\|}$$

is found at

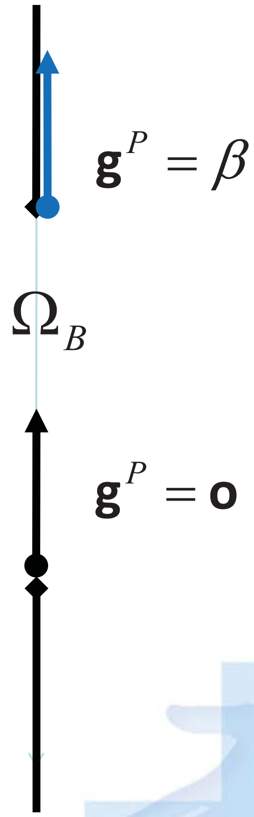
$$\boxed{\mathrm{O}(1) \text{ matrix - vector multiplications}}$$

Scalability no friction FETI Z.D., Kozubek, Brzobohatý, Markopoulos 2009
Scalability no friction BETI Bouchala, Z.D., Sadowská 2008
Scalability 2D friction Z.D., Kozubek, Brzobohatý, Markopoulos, Horyl 2010
Scalability 3D friction Z.D., Kozubek, Brzobohatý, Markopoulos, Horyl 2010

Separable spheric and elliptic constraints



New feature: if the active set
Is known, the components of
the solution are not



Effect of gradient projection

$$\mathbf{x}^{k+1} = P_{\Omega_S}\left(\mathbf{x}^k - \overline{\alpha}\varphi^k\right),\ \text{proportional}$$

$$\alpha \in \left(0, \|A\|^{-1}\right]\ \text{Schöberl, 1998}:$$

$$\boxed{f\left(\mathbf{x}^{k+1}\right) - f(\hat{\mathbf{x}}) \le \left(1 - \frac{1}{2}\alpha\lambda_{\min}\right)\left(f\left(\mathbf{x}^k\right) - f(\hat{\mathbf{x}})\right)}$$

Superrelaxation,

$$\hat{\alpha} = \min\left\{\alpha, 2\|A\|^{-1} - \alpha\right\},\ \alpha \in \left(0, \|A\|^{-1}\right]\ \text{Z.D., Kozubek 2010}:$$

$$\boxed{f\left(\mathbf{x}^{k+1}\right) - f(\hat{\mathbf{x}}) \le \left(1 - \frac{1}{2}\hat{\alpha}\lambda_{\min}\right)\left(f\left(\mathbf{x}^k\right) - f(\hat{\mathbf{x}})\right)}$$

Algorithm MPGP

{Initialization}

Step 0 $\mathbf{x}^0 \geq \mathbf{c}, \Gamma > 0, \overline{\alpha} \in (0, 2\|\mathbf{A}\|^{-1}]$

{Proportioning}

Step 1 If

$$\boxed{\Gamma\left\|\varphi\left(\mathbf{x}^k\right)\right\| < \left\|\beta\left(\mathbf{x}^k\right)\right\|^2}$$

then define $\mathbf{x}^{k+1}$ by gradient projection

{Conjugate gradient step}

Step 2 if $\mathbf{x}^k$ is proportional, then generate $\mathbf{x}^{k+1}$ by trial CG step

{Projection}

Step 3 If $\mathbf{x}^{k+1} \geq \mathbf{c}$, then use it,

else $\mathbf{x}^{k+1} = P_\Omega\left(\mathbf{x}^k - \overline{\alpha}\varphi\left(\mathbf{x}^k\right)\right)$

Z.D., Kozubek 2010

Rate of convergence of MPGP

Theorem: Let $\mathbf{x}^k$ be generated with $\Gamma > 0,\ \overline{\alpha} \in \left(0, 2\|\mathbf{A}\|^{-1}\right)$

Then:

(i) The R-linear rate of convergence in the energy norm is given by

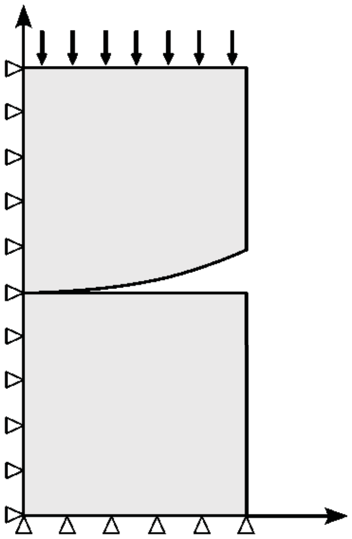
$$\boxed{\left\|\mathbf{x}^k - \hat{\mathbf{x}}\right\|_{\mathbf{A}} \leq \eta^k\left(f\left(\mathbf{x}^0\right) - f\left(\hat{\mathbf{x}}\right)\right)} \quad \text{with} \quad \eta = 1 - \frac{\overline{\alpha}\lambda_{\min}}{2 + 2\Gamma^2}$$

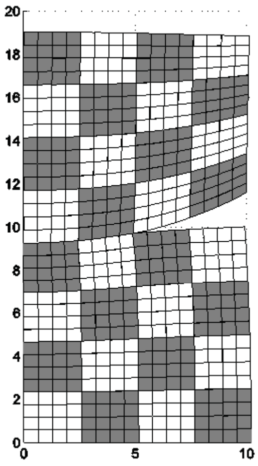
(ii) The R-linear rate of convergence of the projected gradient is given by

$$\boxed{\left\|\mathbf{g}^P\left(\mathbf{x}^k\right)\right\| \leq 2a\eta^k\left(f\left(\mathbf{x}^0\right) - f\left(\hat{\mathbf{x}}\right)\right)} \quad \text{with} \quad a = \frac{\overline{\alpha}^{-1}\lambda_{\min}^{-1}}{\eta(1-\eta)}$$

Z.D., Kozubek 20102005, Z.D. book 2009

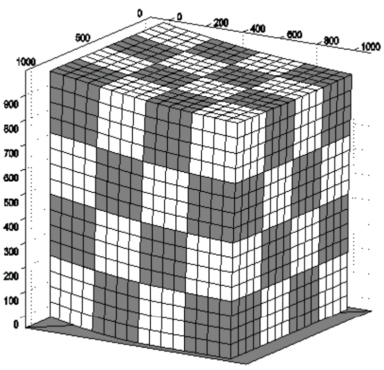
Scalability TFETI – Hertz 2D

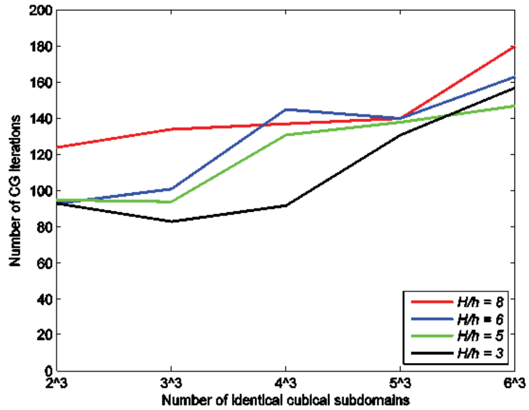




Primal dimension	Dual dimension	Subdomais	Null space	Matrix-vector	Time (sec)
40000	6000	2	6	45	10
640000	11200	32	96	88	78
10240000	198400	512	1536	134	1300

Scalability TBETI – no friction 3D



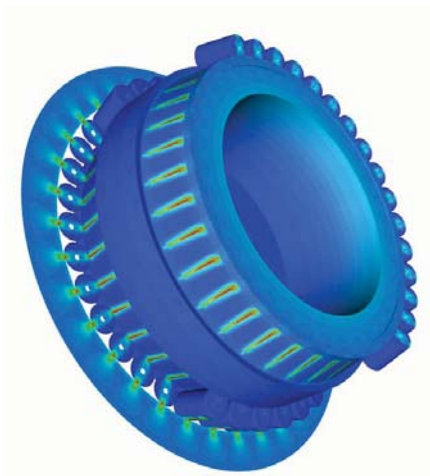


Primal dimension	Dual dimension	Subdomais	Null space	Matrix-vector
11712	5023	8	48	130
93696	43441	64	192	137
316224	63275	396	1068	133

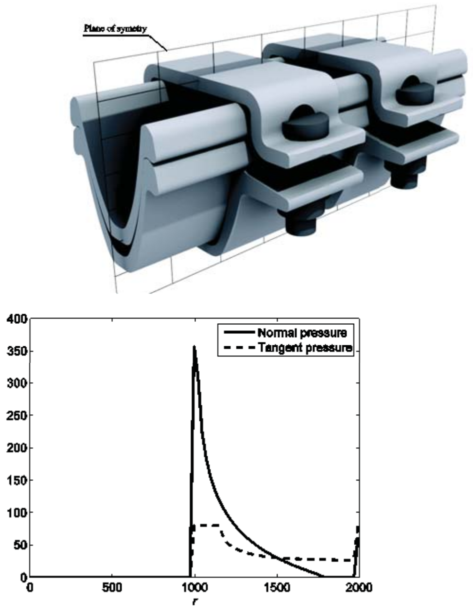
Roler bearing of wind generator

Decomposition and solution of roler bearing of wind generator

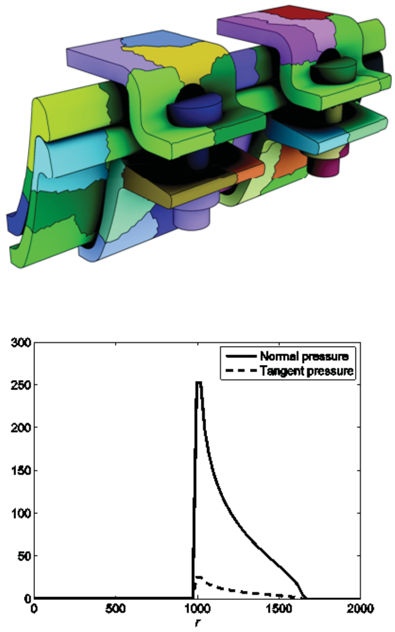
Bodies	**73**
Subdomains	700
Primal variables	2,73 M
Dual variables	459,8 k
Iterations	4270



Applications: yielding clamped connection

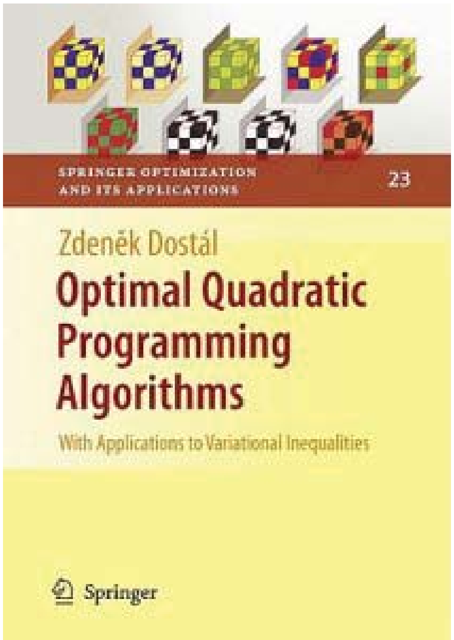


Tresca



Coulomb

1592853 primal, 216604 dual, 250 subdomains,
1922 Hessian multiplications, 5100 sec/24 CPU



Conclusions

- Total FETI/BETI is powerful tool for the solution of contact problem
- Natural coarse grid is a unique way how to get coarse grid to the contact interface
- Well conditioned convex QP and QPQC problems can be solved with optimal complexity
- Theory covers 2D and 3D frictionless contact problems and contact problems with a given (Tresca) friction
- MatSol (Kozubek et al.) is a great tool for the solution of contact problems
- MatSol often outperforms commercial solvers by orders

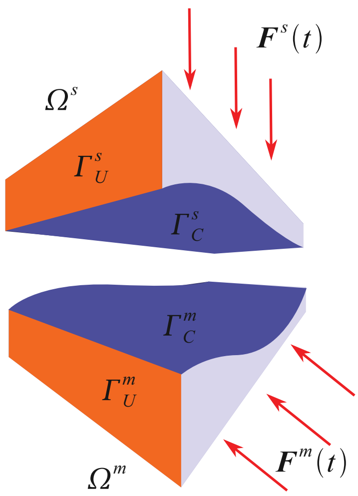
T-FETI domain decomposition method for quasistatic contact problems with Coulomb friction

O. Vlach

KAM VŠB-TU, Ostrava
oldrich.vlach2@vsb.cz

COMPUTATIONAL MECHANICS II 2.12.2010

Geometry of the problem



$$\Omega = \Omega^m \cup \Omega^s \qquad \partial\Omega^p = \overline{\Gamma}_U^p \cup \overline{\Gamma}_F^p \cup \overline{\Gamma}_C^p \quad p \in \{m, s\}$$

Classical formulation

$(\forall p \in \{m, s\})$

equilibrium equations:

$$\frac{\partial \sigma_{ij}(\mathbf{u}^p)}{\partial x_j} + F_i^p = 0 \quad \text{in } \Omega^p \times (0, T)$$

linear Hooke's law, small deformations :

$$\sigma_{ij}(\mathbf{u}^p) = c_{ijkl}\varepsilon_{kl}(\mathbf{u}^p) \qquad \varepsilon_{kl}(\mathbf{u}^p) = \frac{1}{2}\left(\frac{\partial u_k^p}{\partial x_l} + \frac{\partial u_l^p}{\partial x_k}\right)$$

classical boundary conditions:

$$\mathbf{u}^p = \mathbf{0} \quad \text{on } \Gamma_U^p \times (0, T)$$
$$\mathbf{T}(\mathbf{u}^p) := \sigma(\mathbf{u}^p)\mathbf{n}^p = \mathbf{P}^p \quad \text{on } \Gamma_F^p \times (0, T)$$

initial condition:

$$\mathbf{u}^p(0) = \mathbf{u}_0^p \quad \text{in } \Omega^p$$

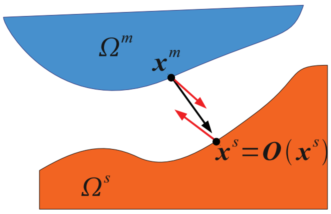
Classical formulation

contact mapping:

$$\mathbf{O}: \; \Gamma_C^m \to \Gamma_C^s \qquad \mathbf{n}^m \parallel (\mathbf{x}^m - \mathbf{O}(\mathbf{x}^m))$$

action - reaction:

$$\big(\sigma(\mathbf{u}^m(\mathbf{x}^m)) - \sigma(\mathbf{u}^s(\mathbf{O}(\mathbf{x}^m)))\big) \cdot \mathbf{n}^m = \mathbf{0} \text{ on } \Gamma_C^m \times (0, T)$$



Classical formulation

notation:

$$[u_n](\mathbf{x}^m) := \left(\mathbf{u}^m(\mathbf{x}^m) - \mathbf{u}^s(\mathbf{O}(\mathbf{x}^m))\right) \cdot \mathbf{n}^m \qquad T_n := \sigma_{ij} n_i^m n_j^m$$

$$[\mathbf{u}_t](\mathbf{x}^m) := \left(\mathbf{u}^m(\mathbf{x}^m) - \mathbf{u}^s(\mathbf{O}(\mathbf{x}^m))\right) - [u_n]\mathbf{n}^m \qquad \mathbf{T}_t := \mathbf{T} - T_n \mathbf{n}^m$$

$$c(\mathbf{x}^m) := \left(\mathbf{O}(\mathbf{x}^m) - \mathbf{x}^m\right) \qquad \ldots \text{ gap}$$

unilateral conditions:

$$[u_n] \le c, \;\; T_n \le 0, \;\; ([u_n] - c)T_n = 0 \quad \text{on } \Gamma_C^m \times (0, T_0);$$

Coulomb's law of friction:

$$[\dot{\mathbf{u}}_t](\mathbf{x}^m) = \mathbf{0} \Rightarrow ||\mathbf{T}_t(\mathbf{x}^m)|| \le -\mathcal{F} T_n(\mathbf{x}^m)$$

$$[\dot{\mathbf{u}}_t](\mathbf{x}^m) \ne \mathbf{0} \Rightarrow \mathbf{T}_t(\mathbf{x}^m) = \mathcal{F} T_n(\mathbf{x}^m) \frac{[\dot{\mathbf{u}}_t](\mathbf{x}^m)}{||[\dot{\mathbf{u}}_t](\mathbf{x}^m)||}, \qquad \mathbf{x}^m \in \Gamma_C^m \times (0, T_0)$$

Weak formulation

Notation

$$V = \prod_{p \in \{m,s\}} \{v^p \in H^1(\Omega^p) \,|\, v^p = 0 \text{ on } \Gamma_U^p\}, \qquad \mathbb{V} = V^3$$

$$\mathbb{K} = \{\mathbf{v} \in \mathbb{V} \,|\, [v_n] \le c \text{ a.e. on } \Gamma_C^m\}$$

$$H^{1/2}(\Gamma_C^m) = V_{|\Gamma_C^m} \quad (\textit{trace space on } \Gamma_C^m \textit{ of functions from } V)$$

$$H^{-1/2}(\Gamma_C^m) = (H^{1/2}(\Gamma_C^m))' \; (\textit{the dual of } H^{1/2}(\Gamma_C^m))$$

$$H_-^{-1/2}(\Gamma_C^m) \ldots \; (\textit{cone of non–positive elements of } H^{-1/2}(\Gamma_C^m))$$

$$\langle \; , \; \rangle \ldots \; \textit{duality pairing between } H^{-1/2}(\Gamma_C^m) \textit{ and } H^{1/2}(\Gamma_C^m)$$

Weak formulation

Notation

$\mathbf{u}, \mathbf{v} \in \mathbb{V}$, $\lambda_n \in H^{-1/2}(\Gamma_C^m)$

$$a(\mathbf{u}, \mathbf{v}) := \sum_{p \in \{m,s\}} \int_{\Omega^p} \sigma_{ij}(\mathbf{u}^p)\varepsilon_{ij}(\mathbf{v}^p)\, dx$$

$$j(\lambda_n, \mathbf{v}) := -\langle \mathcal{F}\lambda_n, ||[\dot{\mathbf{v}}_t]||\rangle$$

$$L(t)(\mathbf{v}) := \sum_{p \in \{m,s\}} \int_{\Omega^p} \mathbf{F}(t) \cdot \mathbf{v}\, dx + \int_{\Gamma_F^p} \mathbf{P}(t) \cdot \mathbf{v}\, ds$$

where

$$\mathbf{F} \in W^{1,2}(0, T, \prod_{p \in \{m,s\}} (L^2(\Omega^p))^3)$$

$$\mathbf{P} \in W^{1,2}(0, T, \prod_{p \in \{m,s\}} (L^2(\Gamma_F^p))^3)$$

Weak formulation

$$\left.\begin{array}{l} \text{Find } \mathbf{u} \in W^{1,2}(0, T, \mathbb{V}),\ \lambda_n \in W^{1,2}(0, T, H^{-1/2}(\Gamma_C^m)) : \\ \mathbf{u}(t) \in \mathbb{K} \quad \text{for a.a. } t \in (0, T),\ \mathbf{u}(0) = \mathbf{u}_0 \text{ in } \Omega \\ a(\mathbf{u}(t), \mathbf{v} - \dot{\mathbf{u}}(t)) + j(\lambda_n(t), \mathbf{v}) - j(\lambda_n(t), \dot{\mathbf{u}}(t)) \geq L(t)(\mathbf{v} - \dot{\mathbf{u}}(t)) \\ \qquad + \langle \lambda_n(t), [v_n - \dot{u}_n(t)]\rangle \quad \forall \mathbf{v} \in \mathbb{V} \text{ and for a.a } t \in (0, T) \\ \langle \lambda_n(t), z_n - ([u_n(t)] - c)\rangle \geq 0 \quad \forall \mathbf{z} \in \mathbb{K} \end{array}\right\} (\mathcal{P})$$

where $\mathbf{u}_0 \in \mathbb{K}$ is such that

$a(\mathbf{u}_0, \mathbf{v} - \mathbf{u}_0) + j(\lambda_{n0}, \mathbf{v} - \mathbf{u}_0) \geq L(0)(\mathbf{v} - \mathbf{u}_0)\ \forall \mathbf{v} \in \mathbb{K}$, $\lambda_0 = T_n(\mathbf{u}_0)_{|\Gamma_C^m}$.

It holds:

$$\lambda_n = T_n(\mathbf{u})_{|\Gamma_C^m}$$

Theorem ([Rocca R. and Coccu M. 01])

If supp $\mathcal{F} \subset \Gamma_c$ and $\mathcal{F}$ is sufficiently small, then $(\mathcal{P})$ has at least one solution.

Time discretization

$\Delta t = T/n \ldots$ time step,

- $t_i = i\Delta t, \quad u^i := u(t_i)$
- $\dot{\mathbf{u}}^{i+1} \approx \frac{\Delta^{i+1}\mathbf{u}}{\Delta t}$, where $\Delta^{i+1}\mathbf{u} := \alpha \mathbf{u}^{i+1} + \beta \mathbf{u}^i + \gamma \mathbf{u}^{i-1}$,
  $\alpha = \frac{3}{2}, \beta = -2, \gamma = \frac{1}{2} \qquad (\alpha = 1, \ \beta = -1, \ \gamma = 0)$
- set $\mathbf{w} := \mathbf{u}^i + \Delta t \mathbf{v} \in \mathbb{V}$
- for simplicity we write $\mathbf{u}$ instead $\mathbf{u}^{i+1}$ and $\mathbf{v}$ instead $\mathbf{u}^i$
- skip index i

Time discretization

At each time step we obtain the following implicit VI:

$$\left.\begin{array}{l} \textit{Find } (\mathbf{u}, \lambda_n) \in \mathbb{V} \times H_-^{-1/2}(\Gamma_C^m) \textit{ such, that} \\ a(\mathbf{u}, \mathbf{w} - \mathbf{u}) + j(\lambda_n, \mathbf{w} - \mathbf{v}) - j(\lambda_n, \mathbf{u} - \mathbf{v}) \geq \\ \qquad L(\mathbf{w} - \mathbf{u}) + \langle \lambda_n, [w_n - u_n] \rangle \quad \forall \mathbf{w} \in \mathbb{V} \\ \langle \mu_n - \lambda_n, [u_n] - c \rangle \geq 0 \qquad \forall \mu_n \in H_-^{-1/2}(\Gamma_C^m) \ . \end{array}\right\} \qquad (\mathcal{Q})$$

$(\mathcal{Q})$ is nothing else than the static contact problem with the following Coulomb friction law:

$$||\mathbf{T}_t(\mathbf{x})|| \leq -\mathcal{F} T_n(\mathbf{x}),$$

$$[\mathbf{u}_t](\mathbf{x}) \neq [\mathbf{v}_t](\mathbf{x}) \quad \Rightarrow \quad \mathbf{T}_t(\mathbf{x}) = \mathcal{F} T_n(\mathbf{x}) \frac{[\mathbf{u}_t(\mathbf{x})] - [\mathbf{v}_t(\mathbf{x})]}{||[\mathbf{u}_t(\mathbf{x})] - [\mathbf{v}_t(\mathbf{x})]||}$$

$$\mathbf{x} \in \Gamma_C^m \times (0, T).$$

Fixed-point formulation of $(\mathcal{Q})$

Let $g \in H_-^{-1/2}(\Gamma_C^m)$, $\mathbf{v} \in \mathbb{K}$ be given and define the auxiliary problem:

$$\left.\begin{aligned} &\textit{Find } \mathbf{u} := \mathbf{u}(g) \in \mathbb{V},\ \lambda_n := \lambda_n(g) \in H_-^{-1/2}(\Gamma_C^m) \textit{ such that} \\ &a(\mathbf{u}, \mathbf{w}-\mathbf{u}) + j(g, \mathbf{w}-\mathbf{v}) - j(g, \mathbf{u}-\mathbf{v}) \geq \\ &\qquad\qquad L(\mathbf{w}-\mathbf{u}) + \langle \lambda_n, [w_n - u_n] \rangle \qquad \forall \mathbf{w} \in \mathbb{V} \\ &\langle \mu_n - \lambda_n, [u_n] - c \rangle \geq 0 \qquad \forall \mu_n \in H_-^{-1/2}(\Gamma_C^m)\ . \end{aligned}\right\} \quad (\mathcal{Q}(g))$$

Define the mapping $\Phi : H_-^{-1/2}(\Gamma_C^m) \mapsto H_-^{-1/2}(\Gamma_C^m)$ by:

$$\Phi(g) = \lambda_n\,, \qquad g \in H_-^{-1/2}(\Gamma_C^m)\ .$$

$(\mathbf{u}, \lambda_n)$ solves $(\mathcal{Q})$ iff λ_n is a fixed-point of Φ:

$$\Phi(\lambda_n) = \lambda_n\ .$$

Solution strategy: the method of successive approximations

Let $\lambda_n^{(0)} \in H_-^{-1/2}(\Gamma_C^m)$ be given, $k := 1$;

if $\lambda_n^{(k)} \in H_-^{-1/2}(\Gamma_C^m)$, $k \geq 1$ is known,

solve $(\mathcal{Q}(\lambda_n^{(k)}))$ and set $\lambda_n^{(k+1)} := \lambda_n$,

where $(\mathbf{u}, \lambda_n)$ is a solution of $(\mathcal{Q}(\lambda_n^{(k)}))$;

$k := k + 1$;

repeat until stopping criterion

Mixed formulation of $(\mathcal{Q}(g))$

Let

$$\Lambda_n = H_-^{-1/2}(\Gamma_C^m)$$
$$\Lambda_t(g) = \{\boldsymbol{\mu}_t \in (L^2(\Gamma_C^m))^2 | \, ||\boldsymbol{\mu}_t|| \le -\mathcal{F}g \text{ a.e. on } \Gamma_C^m\} \, , \quad g \in L_-^2(\Gamma_C^m)$$

Mixed formulation of $(\mathcal{Q}(g))$ reads as follows:

$$\left.\begin{array}{ll} \textit{Find } (\mathbf{u}, \lambda_n, \boldsymbol{\lambda}_t) \in \mathbb{V} \times \Lambda_n \times \Lambda_t(g) \textit{ such that} & \\ & \\ a(\mathbf{u},\mathbf{w}) = L(\mathbf{w}) + \langle \lambda_n, [w_n]\rangle + \langle \boldsymbol{\lambda}_t, [\mathbf{w}_t]\rangle & \forall \mathbf{w} \in \mathbb{V} \\ \langle \mu_n - \lambda_n, [u_n]\rangle \ge \langle \mu_n - \lambda_n, c\rangle & \forall \mu_n \in \Lambda_n \\ \langle \boldsymbol{\mu}_t - \boldsymbol{\lambda}_t, [\mathbf{u}_t]\rangle \ge \langle \boldsymbol{\mu}_t - \boldsymbol{\lambda}_t, [\mathbf{v}_t]\rangle & \forall \boldsymbol{\mu}_t \in \Lambda_t(g)\ . \end{array}\right\} \quad (\mathcal{M}(g))$$

It holds:

- $\mathbf{u} \in \mathbb{K}$ solves $(\mathcal{Q}(g))$;
- $\lambda_n = T_n(\mathbf{u})_{|\Gamma_C^m}$, $\boldsymbol{\lambda}_t = \mathbf{T}_t(\mathbf{u})_{|\Gamma_C^m}$ on Γ_C^m .

T-FETI domain decomposition method

$\overline{\Omega} = \bigcup_{i=1}^{s} \overline{\Omega}_i; \qquad \Omega_i \cap \Omega_j = \emptyset, \quad i \ne j;$ we skip indices m, n for now

$\Gamma_{ij} := \partial\Omega_i \cap \partial\Omega_j$ ... a common part of Ω_i and Ω_j, $i \ne j$, if $meas_2 \Gamma_{ij} > 0$

$\Gamma_{uk} := \partial\Omega_k \cap \Gamma_u$ ... a common part of Ω_k and Γ_u if $meas_2 \Gamma_{uk} > 0$,

$\Gamma_{cl} := \partial\Omega_l \cap \Gamma_c$ ... a common part of Ω_l and Γ_c if $meas_2 \Gamma_{cl} > 0$,

Let

$$\mathcal{I} := \{(i,j) \,|\, meas_2 \Gamma_{ij} > 0\, , \ i < j\}\ , \qquad \mathcal{D} := \{k \,|\, meas_2 \Gamma_{uk} > 0\}$$
$$\mathcal{C} := \{l \,|\, meas_2 \Gamma_{cl} > 0\}$$

$$\mathbb{W} := \{\mathbf{v} \in (L^2(\Omega))^3 \,|\, \mathbf{v}_{|\Omega_i} \in (H^1(\Omega_i))^3\ ,\ i = 1, \dots, s\} = \prod_{i=1}^{s} (H^1(\Omega_i))^3$$

$$\mathbf{v} \in \mathbb{V} \Leftrightarrow \left\{ \begin{array}{ll} \mathbf{v} \in \mathbb{W} & \\ [\mathbf{v}]_{ij} := (\mathbf{v}_i - \mathbf{v}_j)_{|\Gamma_{ij}} = \mathbf{0} & \forall (i,j) \in \mathcal{I} \\ \mathbf{v}_{|\Gamma_{uk}} = \mathbf{0} & \forall\ k \in \mathcal{D} \end{array} \right. \left.\vphantom{\begin{array}{l} a \\ b \end{array}}\right\} \text{realized by Lagrange mult.}$$

T-FETI domain decomposition method

Define the trace spaces

$$Y_{ij} := (H^{1/2}(\Gamma_{ij}))^3 = (H^1(\Omega_i))^3_{|\Gamma_{ij}} = (H^1(\Omega_j))^3_{|\Gamma_{ij}}\,, \quad (i,j) \in \mathcal{I}\,;$$
$$Y_k := (H^{1/2}(\Gamma_{uk}))^3 = (H^1(\Omega_k))^3_{|\Gamma_{uk}}\,,\ k \in \mathcal{D}$$

Then

$$\mathbf{v} \in \mathbb{V} \Leftrightarrow \begin{cases} \mathbf{v} \in \mathbb{W} & & \\ \langle \boldsymbol{\mu}_{ij}, [\mathbf{v}]_{ij} \rangle = \mathbf{0} & \forall \boldsymbol{\mu}_{ij} \in (Y_{ij})' & \forall (i,j) \in \mathcal{I} \\ \langle \boldsymbol{\mu}_k, \mathbf{v}_{|\Gamma_{uk}} \rangle = \mathbf{0} & \forall \boldsymbol{\mu}_k \in (Y_k)' & \forall k \in \mathcal{D}\,, \end{cases}$$

or denoting

$$\Lambda_\Gamma := \prod_{(i,j)\in\mathcal{I}} (Y_{ij})'\,, \qquad \Lambda_d := \prod_{k\in\mathcal{D}} (Y_k)'$$

$$\mathbf{v} \in \mathbb{V} \Leftrightarrow \begin{cases} \mathbf{v} \in \mathbb{W} & \\ \langle \boldsymbol{\mu}_\Gamma, [\mathbf{v}] \rangle := \sum_{(i,j)\in\mathcal{I}} \langle \boldsymbol{\mu}_{ij}, [\mathbf{v}]_{ij} \rangle = \mathbf{0} & \forall \boldsymbol{\mu}_\Gamma \in \Lambda_\Gamma \\ \langle \boldsymbol{\mu}_d, \mathbf{v} \rangle := \sum_{k\in\mathcal{D}} \langle \boldsymbol{\mu}_k, \mathbf{v}_{|\Gamma_{uk}} \rangle = \mathbf{0} & \forall \boldsymbol{\mu}_d \in \Lambda_d \end{cases}$$

T-FETI domain decomposition method

T-FETI formulation of $(\mathcal{M}(g))$:

$$\left.\begin{array}{ll} \textit{Find } (\mathbf{u}, \lambda_n, \boldsymbol{\lambda}_t, \boldsymbol{\lambda}_\Gamma, \boldsymbol{\lambda}_d) \in \mathbb{W} \times \widetilde{\Lambda}_n \times \Lambda_t(g) \times & \Lambda_\Gamma \times \Lambda_d \textit{ s.t.} \\ \sum_{i=1}^{s} a_i(\mathbf{u}, \mathbf{w}) = \sum_{i=1}^{s} L_i(\mathbf{w}) + \langle \lambda_n, [w_n] \rangle + & \\ \qquad \langle \boldsymbol{\lambda}_t, [\mathbf{w}_t] \rangle + \langle \boldsymbol{\lambda}_\Gamma, [\mathbf{w}]_\Gamma \rangle + \langle \boldsymbol{\lambda}_d, \mathbf{w}_d \rangle & \forall \mathbf{w} \in \mathbb{W} \\ \langle \mu_n - \lambda_n, [u_n] \rangle \geq 0 & \forall \mu_n \in \widetilde{\Lambda}_n \\ \langle \boldsymbol{\mu}_t - \boldsymbol{\lambda}_t, [\mathbf{u}_t] \rangle \geq \langle \boldsymbol{\mu}_t - \boldsymbol{\lambda}_t, [\mathbf{v}_t] \rangle & \forall \boldsymbol{\mu}_t \in \Lambda_t(g) \\ \langle \boldsymbol{\mu}_\Gamma, [\mathbf{u}]_\Gamma \rangle = \mathbf{0} & \forall \boldsymbol{\mu}_\Gamma \in \Lambda_\Gamma \\ \langle \boldsymbol{\mu}_d, \mathbf{u}_d \rangle = \mathbf{0} & \forall \boldsymbol{\mu}_d \in \Lambda_d\,. \end{array}\right\} \quad (\mathcal{TM}(g))$$

where $\widetilde{\Lambda}_n := \prod_{l\in\mathcal{C}} H^{-1/2}(\Gamma_{cl})$.

Discretization

Let Ω be a polyhedral domain and $\mathcal{T}_i$ be a partition of $\overline{\Omega}_i$ into tetrahedrons, $i = 1, \ldots, s$, such that $\mathcal{T}_{i|_{\Gamma_{ij}}} = \mathcal{T}_{j|_{\Gamma_{ij}}}$ for all $(i,j) \in \mathcal{I}$.

$$\mathbb{X}_i = \{v \in (C(\overline{\Omega}_i))^3 \mid v_{|_T} \in (P_1(T))^3 \quad \forall T \in \mathcal{T}_i\} , \qquad \mathbb{X} = \prod_{i=1}^{s} \mathbb{X}_i$$

We use algebraic Lagrange multipliers.
Denote

$$\begin{array}{lll} \{x_q^{ij}\}_{q=1}^{d_{ij}} & \ldots \text{ nodes of } \mathcal{T}_i \text{ on } \overline{\Gamma}_{ij} , & (i,j) \in \mathcal{I} ; \\ \{y_q^{k}\}_{q=1}^{d_{k}} & \ldots \text{ nodes of } \mathcal{T}_k \text{ on } \overline{\Gamma}_{uk} , & k \in \mathcal{D} ; \\ \{z_q^{l}\}_{q=1}^{d_{l}} & \ldots \text{ nodes of } \mathcal{T}_l \text{ on } \overline{\Gamma}_{cl} \setminus \overline{\Gamma}_u , & l \in \mathcal{I}^c ; \end{array}$$

Discretization

$$\Lambda_n = \{\mu_n \mid \mu_n = \sum_{l \in \mathcal{I}^c} \sum_{q=1}^{d_l} \mu_q^l \delta_{z_q^l} \, , \ \mu_q^l \le 0\}$$

$$[\mu_n, v_n] := \sum_{l \in \mathcal{I}^c} \sum_{q=1}^{d_l} \mu_q^l v_n(z_q^l) \qquad \forall v \in \mathbb{X} ;$$

$$[\mu_n, v_n] \ge 0 \quad \forall \mu_n \in \Lambda_n \quad \Leftrightarrow \quad v_n(z_q^l) \le 0 \quad \forall l \in \mathcal{I}^c , \quad q = 1, \ldots, d_l$$

$$\Lambda_t(\mathbf{g}) = \{\mu_t \mid \mu_t = \sum_{l \in \mathcal{I}^c} \sum_{q=1}^{d_l} \mu_q^l \delta_{z_q^l} \, , \ \mu_q^l \in \mathbb{R}^2 , \quad ||\mu_q^l|| \le (\mathcal{F} g_q^l)(z_q^l)\}$$

where $\delta_{z_q^l}$ is the Dirac function at z_q^l.

$$[\mu_t, v_t] := \sum_{l \in \mathcal{I}^c} \sum_{q=1}^{d_l} \sum_{k=1}^{2} \mu_{qk}^l v_{tk}(z_q^l) \ , \qquad v_t = (v_{t1}, v_{t2})$$

Discretization

$$\mathbb{\Lambda}_\Gamma = \{\mu_\Gamma \mid \mu_\Gamma = \sum_{(i,j)\in\mathcal{I}} \sum_{q=1}^{d_{ij}} \mu_q^{ij} \delta_{x_q^{ij}}, \ \mu_q^{ij} \in \mathbb{R}^3\}$$

$$\mathbb{\Lambda}_d = \{\mu_d \mid \mu_d = \sum_{k\in\mathcal{D}} \sum_{q=1}^{d_k} \mu_q^k \delta_{y_q^k}, \ \mu_q^k \in \mathbb{R}^3\}$$

$$[\mu_d, v] := \sum_{k\in\mathcal{D}} \sum_{q=1}^{d_k} \sum_{l=1}^{3} \mu_{ql}^k v_l(y_q^k)$$

$$[\mu_d, v] = 0 \quad \forall \mu_d \in \mathbb{\Lambda}_d \quad \Leftrightarrow \quad v_l(y_q^k) = 0 \quad \forall k \in \mathcal{D}, \ q = 1, ..., d_k, \ l = 1, 2, 3$$

$$\mathbb{W} \sim \mathbb{X}; \quad \widetilde{\Lambda}_n \sim \mathbb{\Lambda}_n; \quad \Lambda_t(g) \sim \mathbb{\Lambda}_t(\mathbf{g})$$
$$\Lambda_\Gamma \sim \mathbb{\Lambda}_\Gamma; \quad \Lambda_d \sim \mathbb{\Lambda}_d$$

Algebraic formulation of the mixed problem

$$\left.\begin{array}{l} \text{Find } (\mathbf{u}, \boldsymbol{\lambda}) \in \mathbb{R}^p \times \widetilde{\boldsymbol{\Lambda}}(\mathbf{g}) \text{ such that} \\ \mathbf{K}\mathbf{u} = \mathbf{f} + \mathbf{B}^\top \boldsymbol{\lambda} \\ (\boldsymbol{\mu} - \boldsymbol{\lambda})^\top \mathbf{B}\mathbf{u} \geq (\boldsymbol{\mu} - \boldsymbol{\lambda})^\top \mathbf{z} \qquad \forall \boldsymbol{\mu} \in \widetilde{\boldsymbol{\Lambda}}(\mathbf{g}) \end{array}\right\} \qquad (\mathbf{M}(\mathbf{g}))$$

where

$$\widetilde{\boldsymbol{\Lambda}}(\mathbf{g}) = \mathbb{R}_-^{m_n} \times \boldsymbol{\Lambda}_t(\mathbf{g}) \times \mathbb{R}^{m_\Gamma + m_d}$$
$$\boldsymbol{\Lambda}_t(\mathbf{g}) = \{(\boldsymbol{\mu}_a^\top, \boldsymbol{\mu}_b^\top)^\top \in \mathbb{R}^{2m_n} \mid ||(\mu_{ai}, \mu_{bi})|| \leq \mathcal{F}_i g_i \ \forall i = 1, \dots, m_n\}$$
$$\boldsymbol{\mu} = (\boldsymbol{\mu}_n^\top, \boldsymbol{\mu}_t^\top, \boldsymbol{\mu}_\Gamma^\top, \boldsymbol{\mu}_d^\top)^\top$$
$$\mathbf{z} = (\mathbf{c}^\top, (\mathbf{T}\mathbf{v})^\top, \mathbf{0}^\top, \mathbf{0}^\top)^\top$$
$$\mathbf{B} = (\mathbf{N}^\top, \mathbf{T}, \mathbf{B}_\Gamma^\top, \mathbf{B}_d^\top)^\top$$

Remark

Note that $\mathbf{K}$ is block–diagonal where all diagonal blocks are singular.

Dual algebraic formulation

We eliminate $\mathbf{u}$ from $(\mathbf{M}(\mathbf{g}))_1$:

$$\mathbf{u} = \mathbf{K}^{\dagger}(\mathbf{f} + \mathbf{B}^{\top}\boldsymbol{\lambda}) + \mathbf{R}\boldsymbol{\alpha} \ ,$$

where $\mathbf{K}^{\dagger}$ is a generalized inverse of $\mathbf{K}$, columns of $\mathbf{R} \in \mathbb{R}^{p \times l}$ span *ker* $\mathbf{K}$ and $\boldsymbol{\alpha} \in \mathbb{R}^{l}$.

Dual algebraic formulation

$$\left.\begin{array}{ll} \textit{Find } \boldsymbol{\lambda} \in \boldsymbol{\Lambda}(\mathbf{g}) \textit{ such that} & \\ \mathcal{S}(\boldsymbol{\lambda}) \leq \mathcal{S}(\boldsymbol{\mu}) & \forall \boldsymbol{\mu} \in \boldsymbol{\Lambda}(\mathbf{g}) \end{array}\right\}$$

where

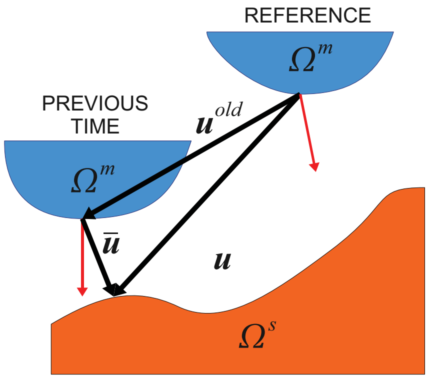
$$\begin{aligned} \mathcal{S}(\boldsymbol{\mu}) &= \frac{1}{2}\boldsymbol{\mu}^{\top}\mathbf{F}\boldsymbol{\mu} - \boldsymbol{\mu}^{\top}\mathbf{h} \\ \boldsymbol{\Lambda}(\mathbf{g}) &= \{\boldsymbol{\mu} \in \widetilde{\boldsymbol{\Lambda}}(\mathbf{g}) \,|\, \mathbf{G}\boldsymbol{\mu} = \mathbf{e}\} \\ \mathbf{F} &= \mathbf{B}\mathbf{K}^{\dagger}\mathbf{B}^{\top} \ , \qquad \mathbf{h} = \mathbf{z} - \mathbf{B}\mathbf{K}^{\dagger}\mathbf{f} \\ \mathbf{G} &= \mathbf{R}^{\top}\mathbf{B}^{\top} \ , \qquad \mathbf{e} = -\mathbf{R}^{\top}\mathbf{f} \end{aligned}$$

Remark

This quadratic programming problem with separable simple and quadratic constraints is solved by the algorithm MPGP.

Correction of contact mapping

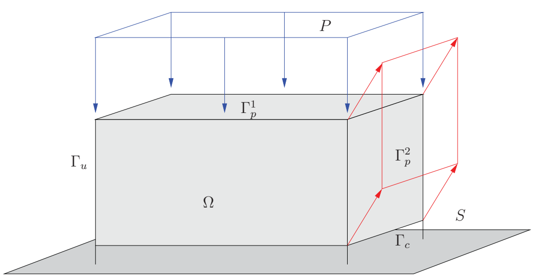
$$\overline{\mathbf{N}}\overline{\mathbf{u}} \leq \overline{c} \wedge \mathbf{u} = \overline{\mathbf{u}} + \mathbf{u}^{old} \;\Rightarrow\; \overline{\mathbf{N}}\mathbf{u} \leq \overline{\mathbf{c}} + \overline{\mathbf{N}}\overline{\mathbf{u}}$$



Model example

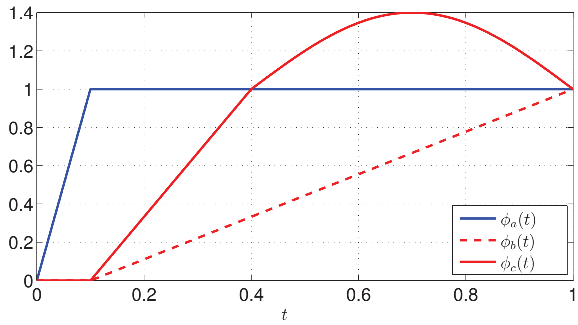
$\Omega = (0,3) \times (0,1) \times (0,1)$ $[m]$.
Young's modulus $E = 2.119e5$ $[Pa]$,
Poisson's ratio $\sigma = 0.277$
50 timesteps

$$\begin{aligned} P(t) &= \phi_a(t)(\quad 0\,,\ 0\,,\ 3)1e1 && \text{on } \Gamma_P^1 \\ P(t) &= \phi_x(t)(\ 0.9\,,\ 0\,,\ 1.5)1e1 && \text{on } \Gamma_P^2\,,\ x \in \{b, c\}. \end{aligned}$$

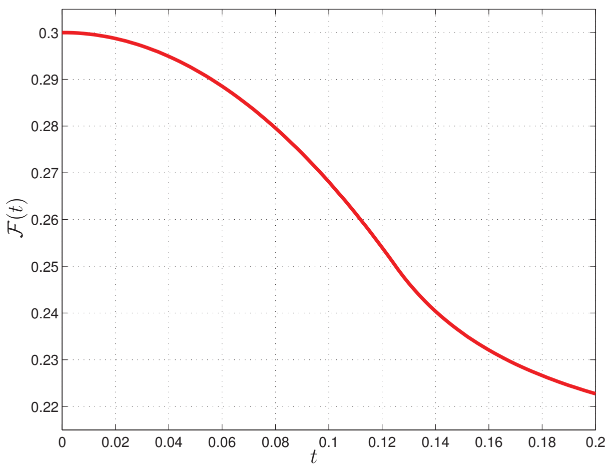


Loading history (characterized by $\phi_x : [0,1] \to \mathbb{R}^1$)

$$\phi_x(t) := \begin{cases} \phi_b(t) \ \ldots & \textit{monotone loading} \\ \phi_c(t) \ \ldots & \textit{nonmonotone loading} \end{cases}$$



Coefficient of friction $\mathcal{F}(||\dot{\mathbf{u}}_t||)$

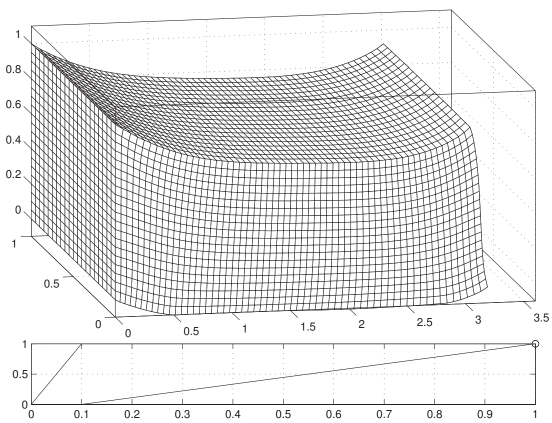


Dependence on h

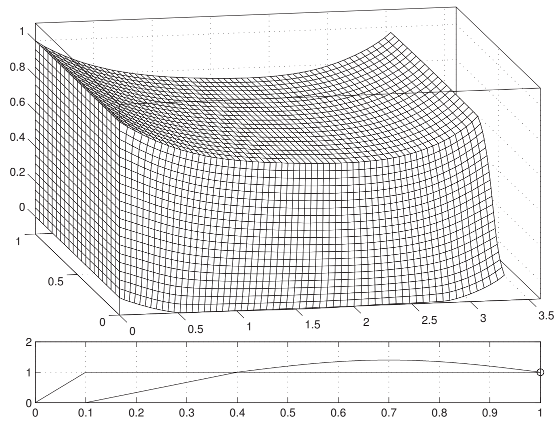
n_s, n_p, n_d ... number of subdomains, primal and dual variables
it ... number of fixed point iterations (sum for all time steps)
n_m ... dual matrix multiplication
time [s] ... CPU time

n	n_s	n_p	n_d	it	n_m	time [s]
1	3	1944	594	254/234	24903/30353	2.8e2/3.4e2
2	24	15552	5652	257/237	32554/38562	2.5e3/3.4e3
4	192	124416	48816	256/238	43013/54067	3.0e4/3.5e4
5	648	419904	168804	257/239	53631/69620	1.1e5/1.5e5

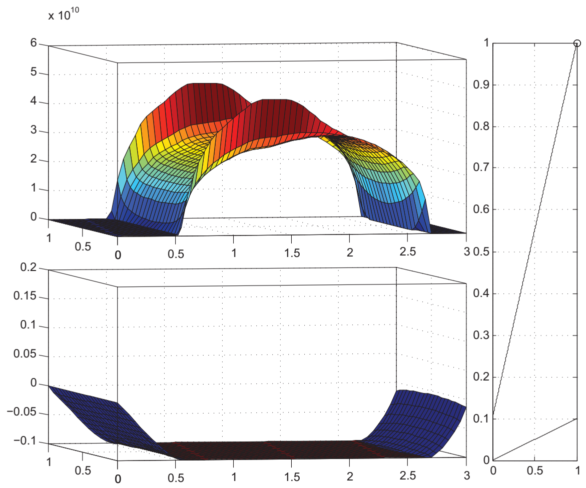
Deformation



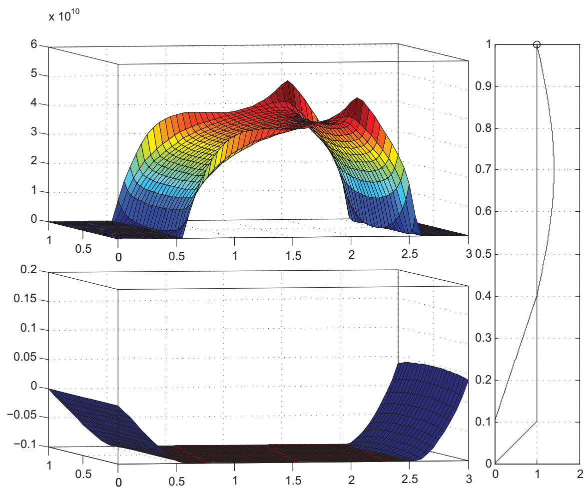
Deformation



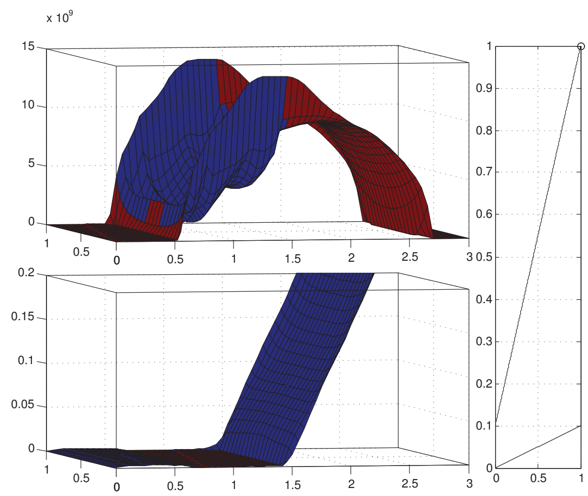
Normal contact stress $-T_n$ and displacement $-u_n$



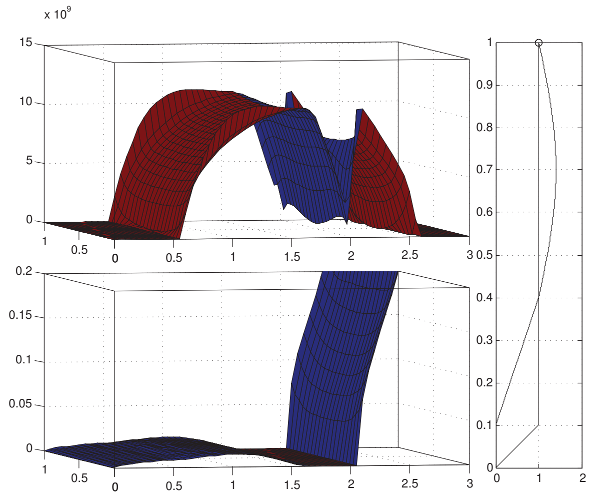
Normal contact stress $-T_n$ and displacement $-u_n$



Norms of tangential stress $||T_t||$ and velocity $||\dot{u}_t||$



Norms of tangential stress $||T_t||$ and velocity $||\dot{u}_t||$

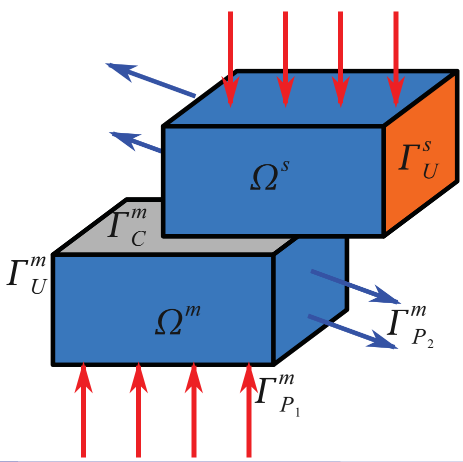


Model example

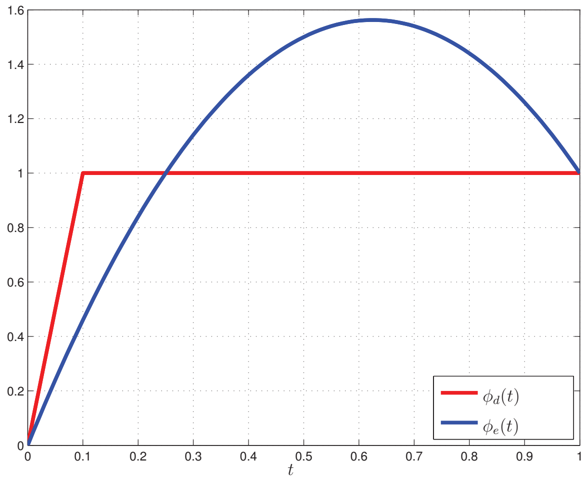
$\Omega^m = (0,2) \times (-0.05, 1.05) \times (0,1)$ $\Omega^s = (\frac{2}{3}, \frac{8}{3}) \times (0,1) \times (1.1, 2.1)$.
$E^m = E^s = 5e4, \qquad \sigma^m = 0.277, \quad \sigma^m = 0.35$

$$\left.\begin{aligned} P(t) &= \phi_d(t)(0,\ 0,\ 2)1e3 && \text{on } \Gamma^m_{P_1} \\ P(t) &= \phi_e(t)(3,\ 0,\ -1)1e3 && \text{on } \Gamma^m_{P_2} \end{aligned}\right\}$$

and symmetric on Γ
10 timesteps



Loading history

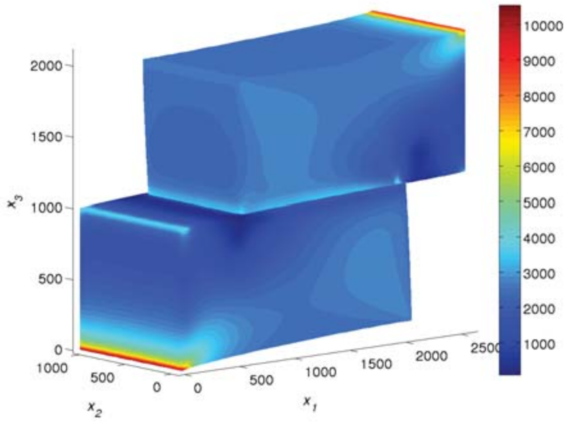


Dependence on h

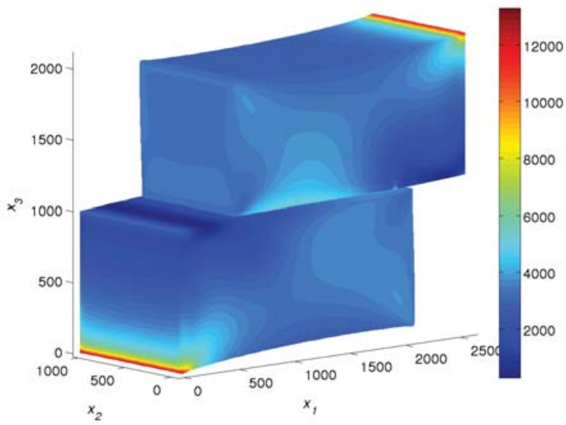
n_s, n_p, n_d ... number of subdomains, primal and dual variables
it ... number of fixed point iterations (sum for all time steps)
n_m ... dual matrix multiplication
time [s] ... CPU time

n	n_s	n_p	n_d	it	n_m	time [s]
1	4	4116	798	119	1851	3.5e1
2	32	32928	9336	138	4353	3.2e2
3	108	111132	34809	163	5930	1.2e3
4	256	263424	86301	195	7440	3.1e3

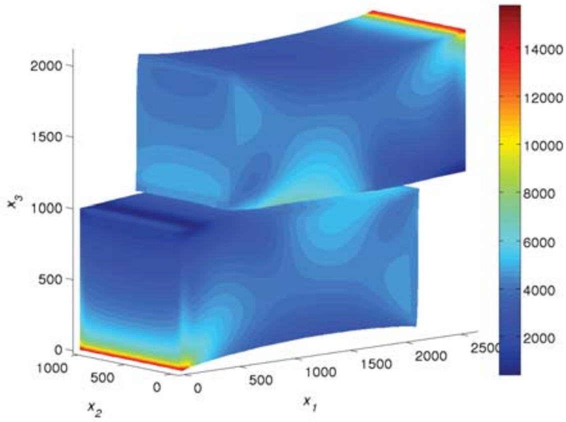
Deformation



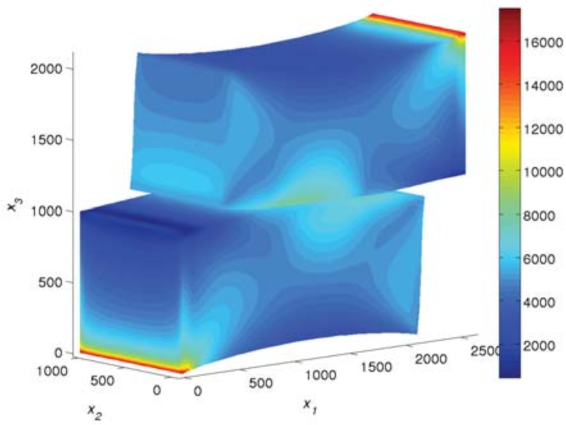
Deformation



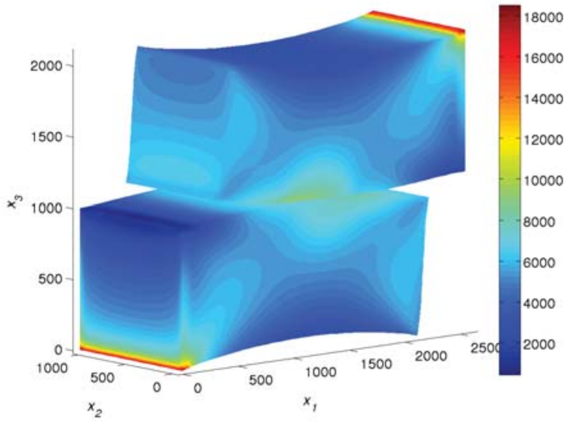
Deformation



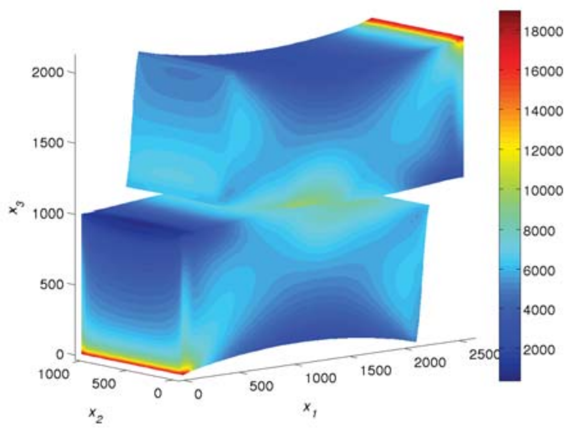
Deformation



Deformation

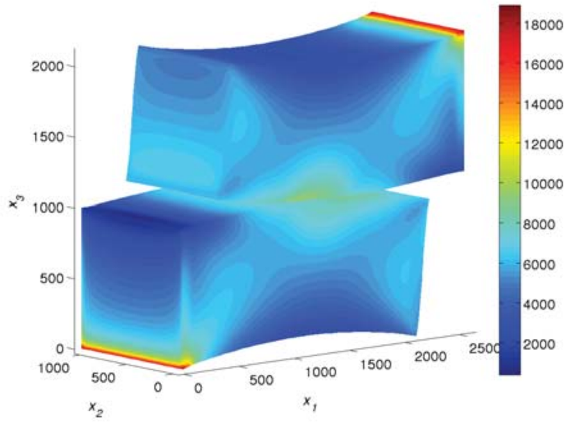


Deformation

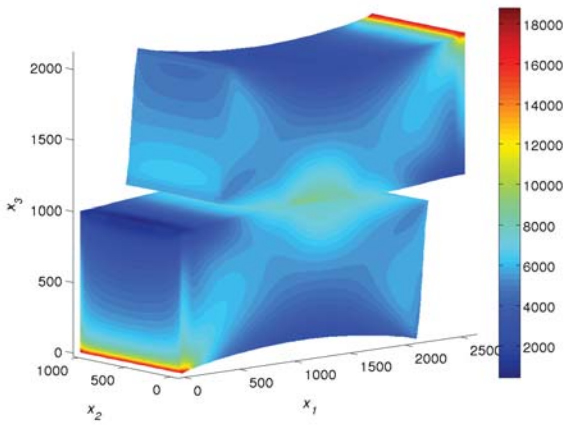


Vlach (KAM VŠB) TFETI + 3D quasistatic CM II, 2.12 24 / 25

Deformation

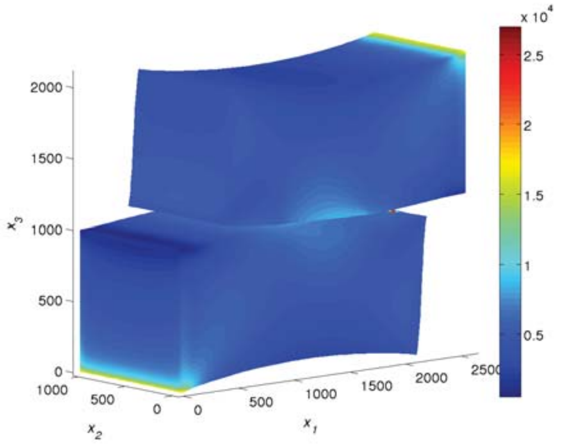


Deformation

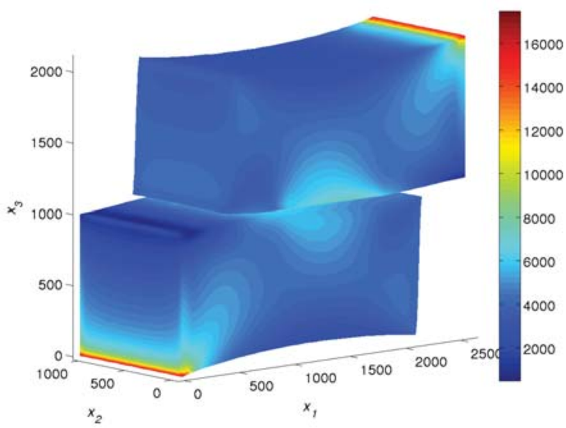


Vlach (KAM VŠB) TFETI + 3D quasistatic CM II, 2.12 24 / 25

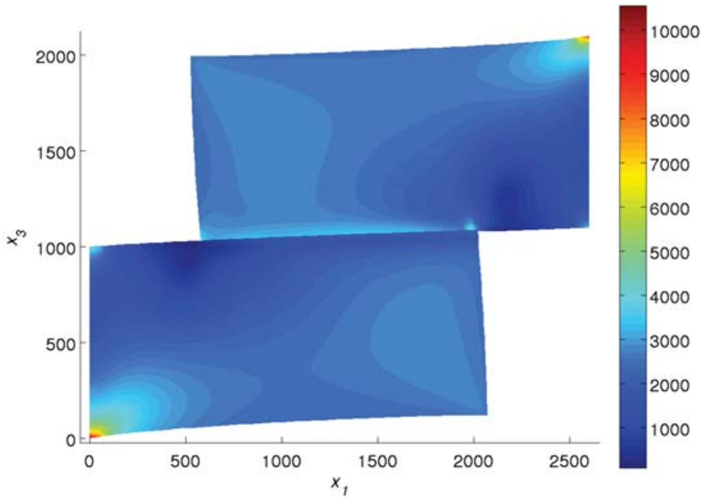
Deformation



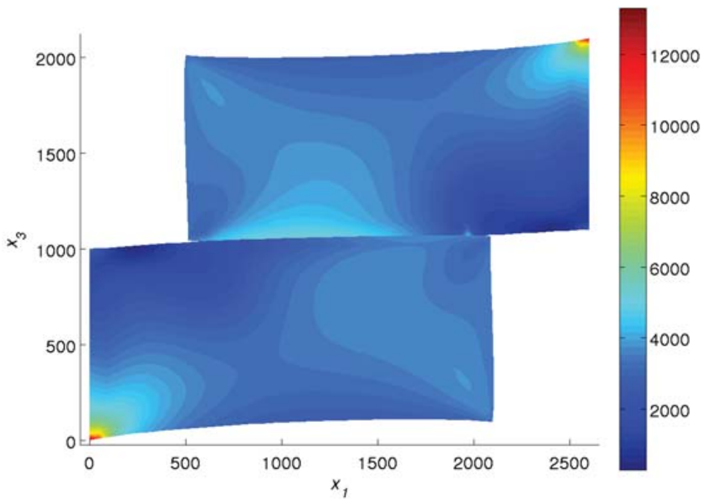
Deformation



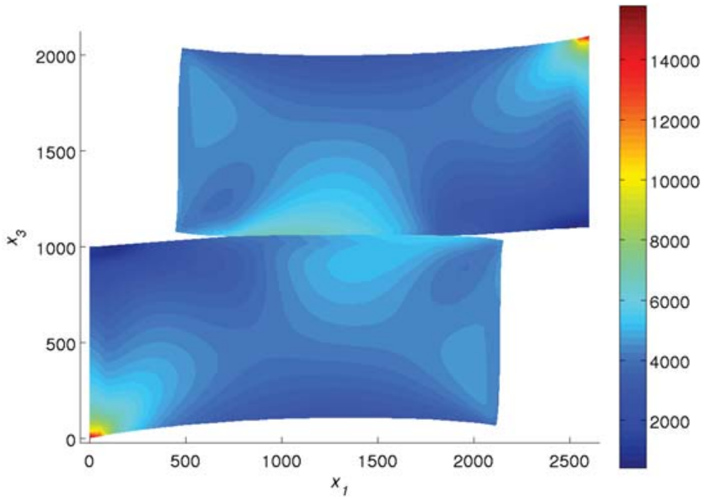
Deformation



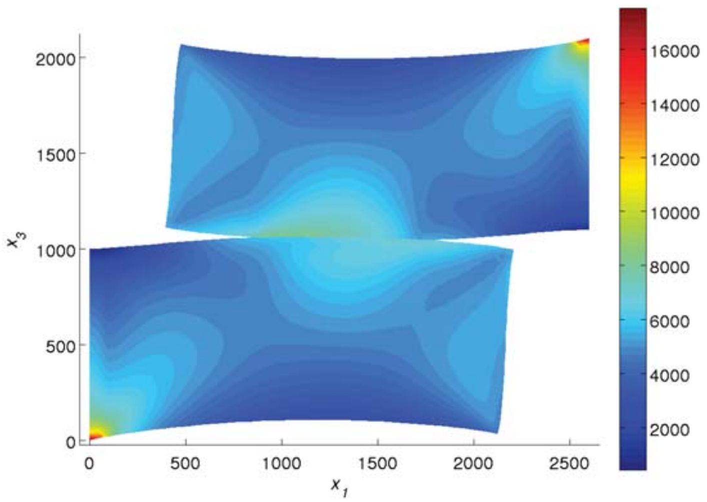
Deformation



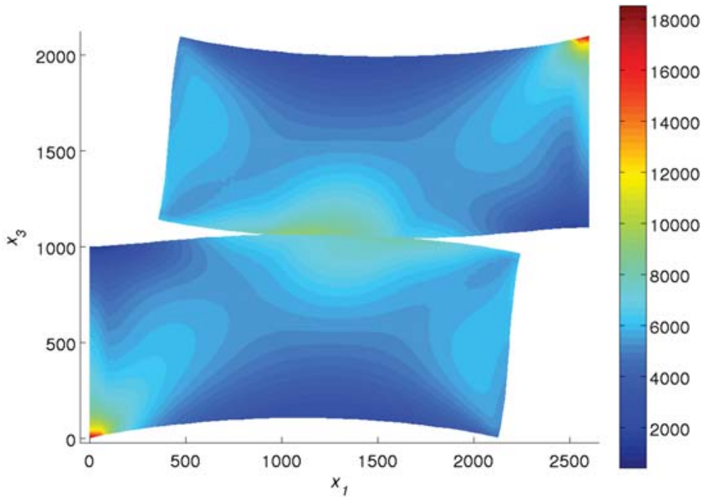
Deformation



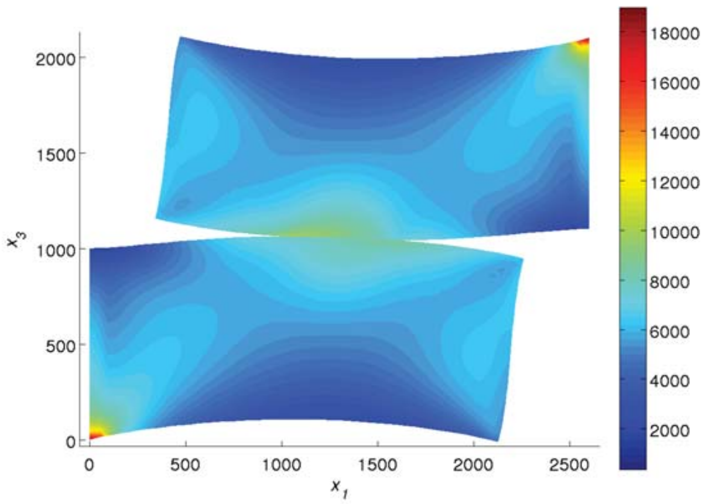
Deformation



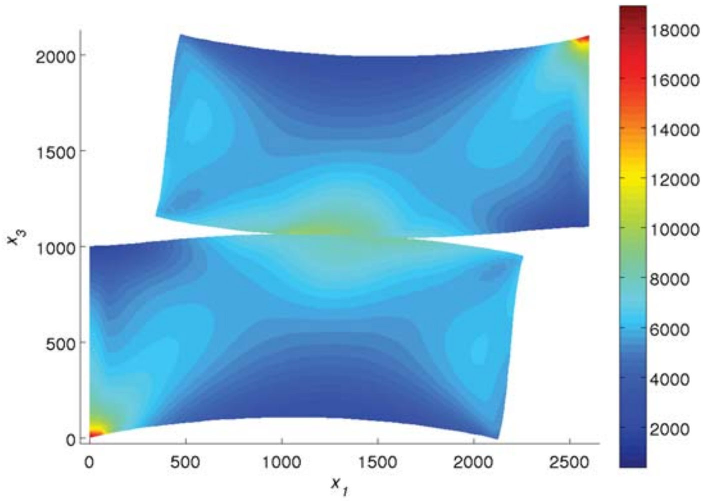
Deformation



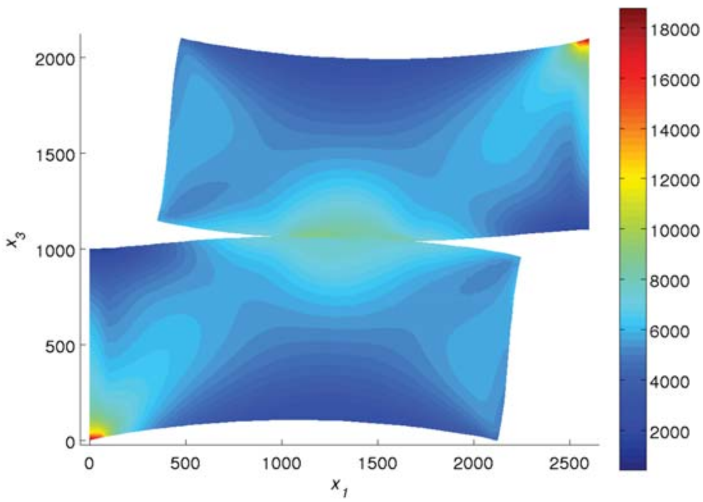
Deformation



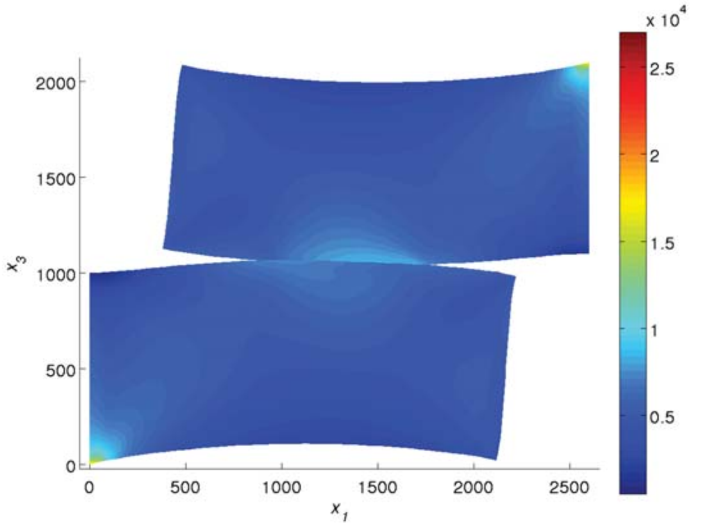
Deformation



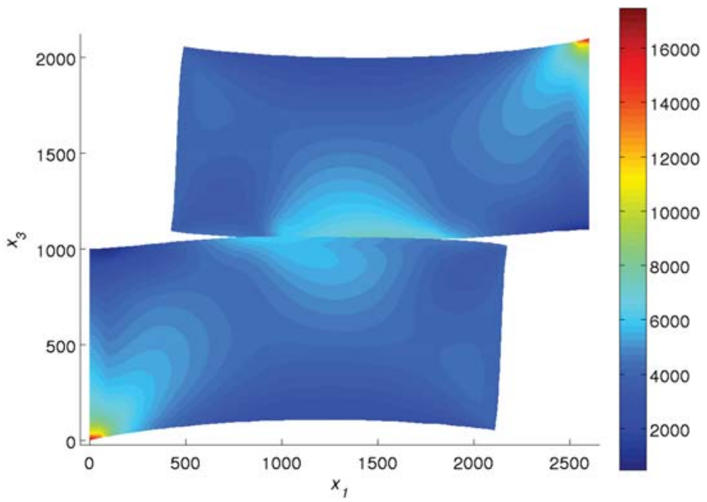
Deformation



Deformation



Deformation



Numerically and Parallel Scalable FETI Based Algorithms for Contact Problems of Mechanics and Their Powerful Ingredients

T. Kozubek,
Z. Dostál, T. Brzobohatý, A. Markopoulos,
R. Kučera, V. Vondrák, M. Sadowská

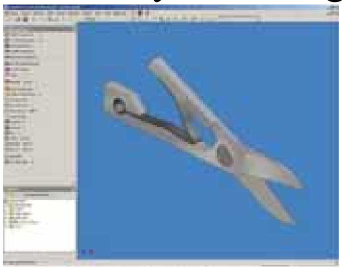
Department of Applied Mathematics
VSB-Technical University of Ostrava
Czech Republic

Outline

1. MatSol – Matlab Library with scalable solvers based on FETI(BETI)
2. Optimal TFETI(TBETI) based algorithm for contact problems
3. Main ingredients of TFETI(TBETI)
    1. Parallel implementation
    2. Pseudoinverse stabilization
    3. Fixing rigid body motions
    4. Domain decomposition correction
    5. Contact direction correction
4. Benchmarks
5. Dynamics

Problem specification and mesh gen.
(problem type, material parameters, initial and boundary conditions, …)

Geometry modeling



Importer
STL, IGIS, VRML, …

ANSYS
COMSOL MULTIPHYSICS
ANSA

Exporter
CDB format

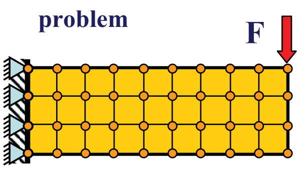
CAD System Integration
- Autodesk Inventor
- CATIA
- Pro/ENGINEER
- Solid Edge
- SolidWorks
- Unigraphics

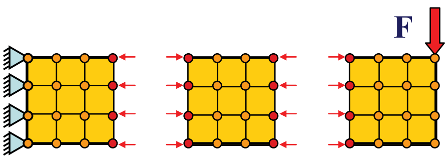
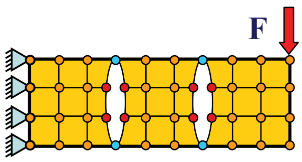
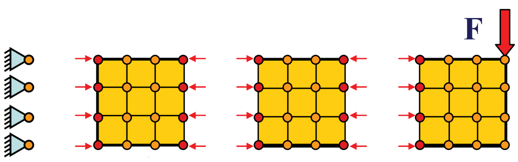
MatSol

Library with scalable solvers based on FETI and BETI domain decomposition methods

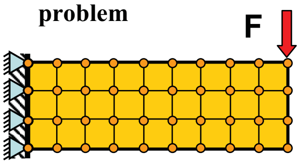
T. Kozubek, T. Brzobohatý, A. Markopoulos
Z. Dostál, V. Vondrák, R. Kučera, M. Sadowská

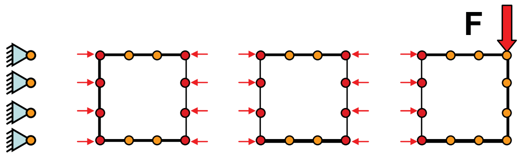
FETI approaches



1. FETI		**subdomains are fixed or free but with different defects**
2. FETI-DP		**FETI-DP (partial splitting, nonsingular)**
3. TFETI		**all subdomains are free with the same defect**

problem



1. FETI		subdomains are fixed or free but with different defects
2. FETI-DP		FETI-DP (partial splitting, nonsingular)
3. TBETI		all subdomains are free with the same defect

Total FETI – primal formulation

$$\min \tfrac{1}{2} u^T K u - u^T f$$ energy functional

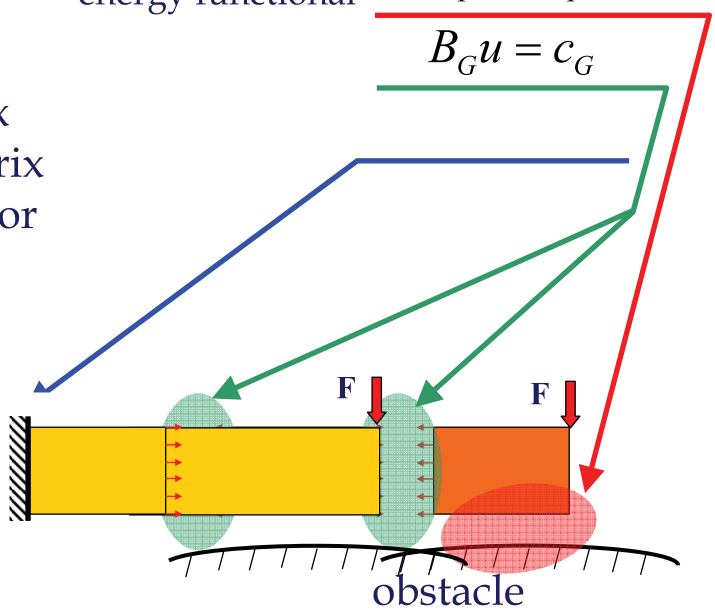
$B_I u \le c_I$

$B_G u = c_G$

u displacement
K stiffness matrix
B constraint matrix
c constraint vector

Total FETI

Dirichlet b.c. are enforced by LM, $B_B u = c_B$



Total FETI – dual formulation

$$\min \tfrac{1}{2}\lambda^T F\lambda - \lambda^T \hat{d} \quad \text{subject to} \quad \lambda_I \geq o,\ G\lambda = e$$ a

$$F = BK^+B^T$$
$$G = R^T B^T$$
$$\hat{d} = BK^+ f - c$$
$$e = R^T f$$ b

$$B = \begin{bmatrix} B_I \\ B_G \\ B_B \end{bmatrix} = \begin{bmatrix} B_I \\ B_E \end{bmatrix},\ c = \begin{bmatrix} c_I \\ c_G \\ c_B \end{bmatrix} = \begin{bmatrix} c_I \\ c_E \end{bmatrix},\quad \lambda = \begin{bmatrix} \lambda_I \\ \lambda_E \end{bmatrix}$$ c

R *basis of the kernel of K, rigid body motions*
α *amplitudes* d

Relation between primal and dual variables: $u = K^+\left(f - B^T\lambda\right) + R\alpha$ e

Homogenization of equality constraints

$$\min \tfrac{1}{2}\lambda^T F\lambda - \lambda^T \hat{d} \quad \text{subject to} \quad \lambda_I \geq o,\ G\lambda = e$$

$$G\hat{\lambda} = e,\ \ \hat{\lambda} = G^T(GG^T)^{-1}e$$
$$G\lambda = e,\ \lambda = \mu + \hat{\lambda} \Leftrightarrow G\mu = o,$$
$$\lambda_I \geq o \ \Leftrightarrow\ \mu_I \geq -\hat{\lambda}_I$$

$$\min \tfrac{1}{2}\lambda^T F\lambda - \lambda^T d \quad \text{subject to} \quad \lambda_I \geq -\hat{\lambda}_I,\ G\lambda = o$$

$$\textit{with}\ \ d = \hat{d} - F\hat{\lambda}.$$

Applying natural coarse grid preconditioner

Natural coarse grid projectors:

$$Q = G^T (GG^T)^{-1} G \qquad P = I - Q$$

$$\operatorname{Im} Q = \operatorname{Im} G^T \qquad \operatorname{Im} P = Ker\ G$$

$$\min \tfrac{1}{2} \lambda^T PFP\lambda - \lambda^T Pd \quad \text{subject to } \lambda_I \geq -\hat{\lambda}_I, G\lambda = o$$

MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

Optimal estimates

Theorem: The following bounds for PFP hold:

$$\lambda_{\min}(PFP \mid \operatorname{Im} P) \geq C_1$$

$$\lambda_{\max}(PFP \mid \operatorname{Im} P) = \|PFP\| \leq C_2 \frac{H}{h}$$

$$\kappa(PFP \mid \operatorname{Im} P) \leq C_3 \frac{H}{h}$$

with constants C_1, C_2, C_3 independent on both H and h

Proof in C. Farhat, J. Mandel and F.-X. Roux CMAME 1994

BETI J.Bouchala, Z.Dostal, M.Sadowska, Computing 2009

Optimality and scalability

Theorem:

The solutions of the discretized model problem with

$$H / h \leq C$$

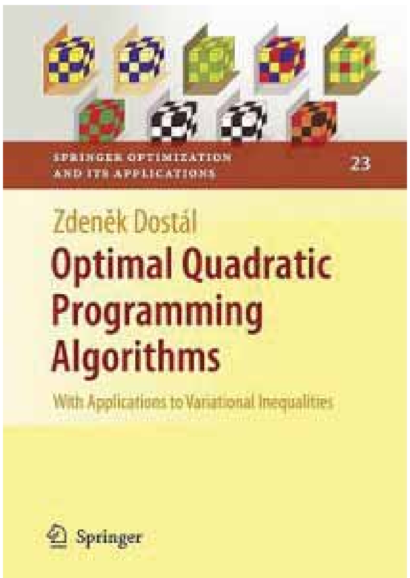
and a given relative precision may be obtained by SMALBE/MPRGP at

$O(1)$ **matrix / vector multiplications**.

Z. Dostal, D. Horak, SINUM 2007 (scalar case)
Z. Dostal, T. Kozubek, V. Vondrak, A. Markopoulos, T. Brzobohaty, *IJNME* **2009**
J. Bouchala, Z. Dostal, M. Sadowska, *Computing* **2008, 2009**

MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

Many of this and much much more ….



Z. Dostál: Theoretically supported scalable algorithms for contact problems,
next presentation, at 14.00.

Primal formulation with Tresca friction

$$\min \tfrac{1}{2} u^T K u - u^T f + \sum_{i=1}^{m_c} \Psi_i \left\| T_i u \right\| \quad s.t.\ B_I u \le c_I, B_E u = c_E$$

$T_i u$ ... jump of the (master and slave) displacement at projected λ_i to the tengent vector in 2D and to the tangential plane in 3D

Ψ_i ... associated slip bound

Removing non-differentiability

$$\sum_{i=1}^{m_c} \Psi_i \left\| T_i u \right\| = \sum_{i=1}^{m_c} \max_{\|\tau_i\| \le \Psi_i} \tau_i^T T_i u$$

Dual formulation with Tresca friction

$$\min \tfrac{1}{2} \lambda^T F \lambda - \lambda^T \hat{d} \quad \text{s.t.}\ \lambda_I \ge o,\ \|\tau_i\| \le \Psi_i\ \forall i,\ G\lambda = e$$ a

$$F = BK^+B^T$$
$$G = R^T B^T$$
$$\hat{d} = BK^+ f - c$$
$$e = R^T f$$
b

$$\lambda = \begin{bmatrix} \lambda_I \\ \tau \\ \lambda_E \end{bmatrix} \qquad B = \begin{bmatrix} B_I \\ T \\ B_E \end{bmatrix} \qquad c = \begin{bmatrix} c_I \\ o \\ c_E \end{bmatrix}$$
c

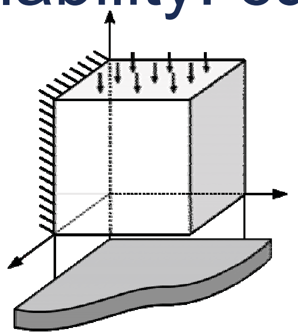
R *basis of the kernel of K, rigid body motions*
α *amplitudes*
d

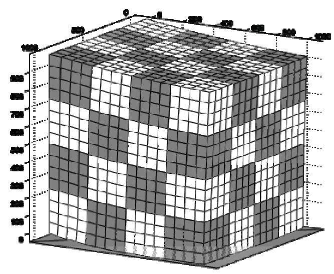
Relation between primal and dual variables: $u = K^+ \left(f - B^T \lambda \right) + R\alpha$ e

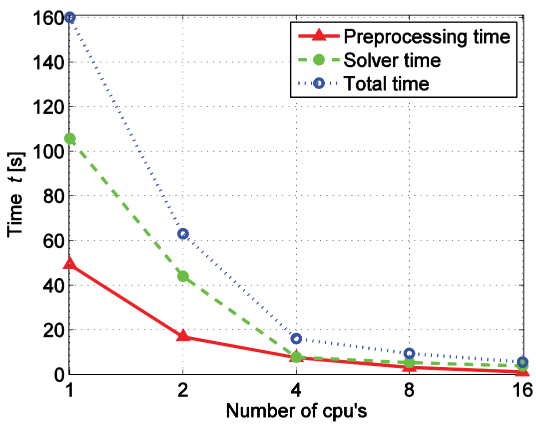
Remark. The Tresca friction is a simple friction law which violates some natural physical principles, but it can be used to define a mapping whose fixed point is a solution to the problem with the Coulomb friction.

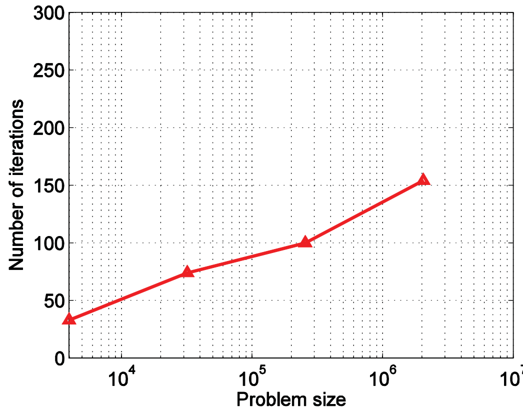
- For more details: *Hlaváček I, Haslinger J, Nečas J, Lovíšek J. Solution of Variational Inequalities in Mechanics, Springer Verlag, Berlin, 1988.*

Scalability: cube over the obstacle

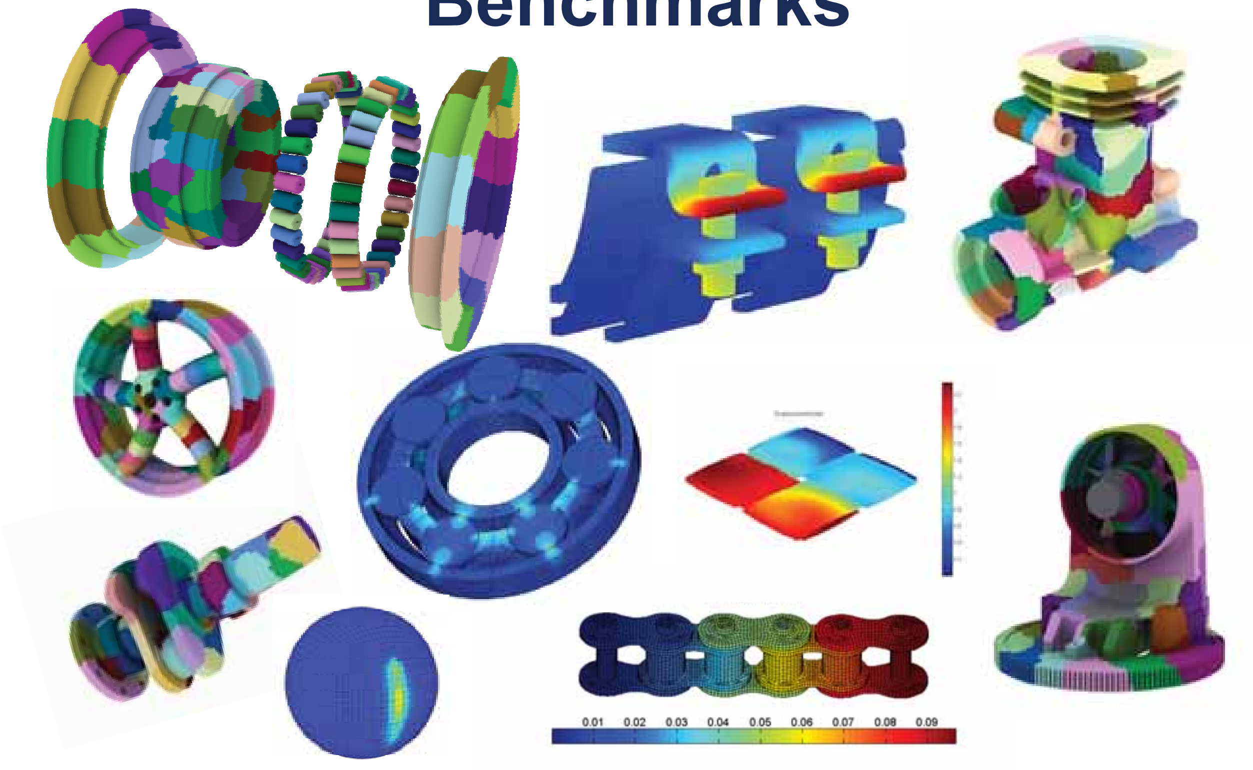




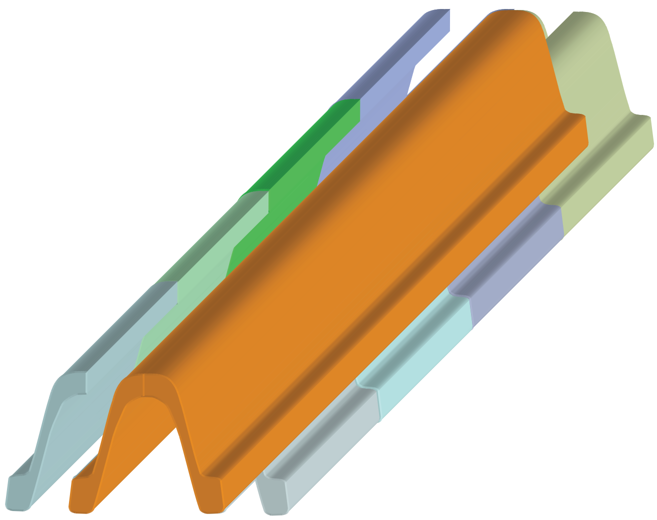




Benchmarks



1. ingredient: parallel implementation



Block diagonal structure of the stiffness matrix => suitable for parallel implementation

$$K = \begin{pmatrix} K^1 & O & \vdots & O \\ O & K^2 & \vdots & O \\ \cdots & \cdots & \ddots & \cdots \\ O & O & \vdots & K^n \end{pmatrix}$$

Coercive and semicoercive problems may be solved!

Parallel programming

Typical use cases

- Parallel for-Loops
  - Many iterations loop
  - Long iterations loop
- Offloading Work
- Large Data Sets

Cluster HP model BLc7000 (c-class).

Configuration of the computational node
2x dual core CPU AMD Opteron 2210 HE
8GB ECC DDR2 667MHz RAM

IT4Innovations, Centrum excelence, Ostrava, Czech Republic, cluster with more than 30,000 processors. http://www.it4i.eu

IT4Innovations
Centrum excelence

MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

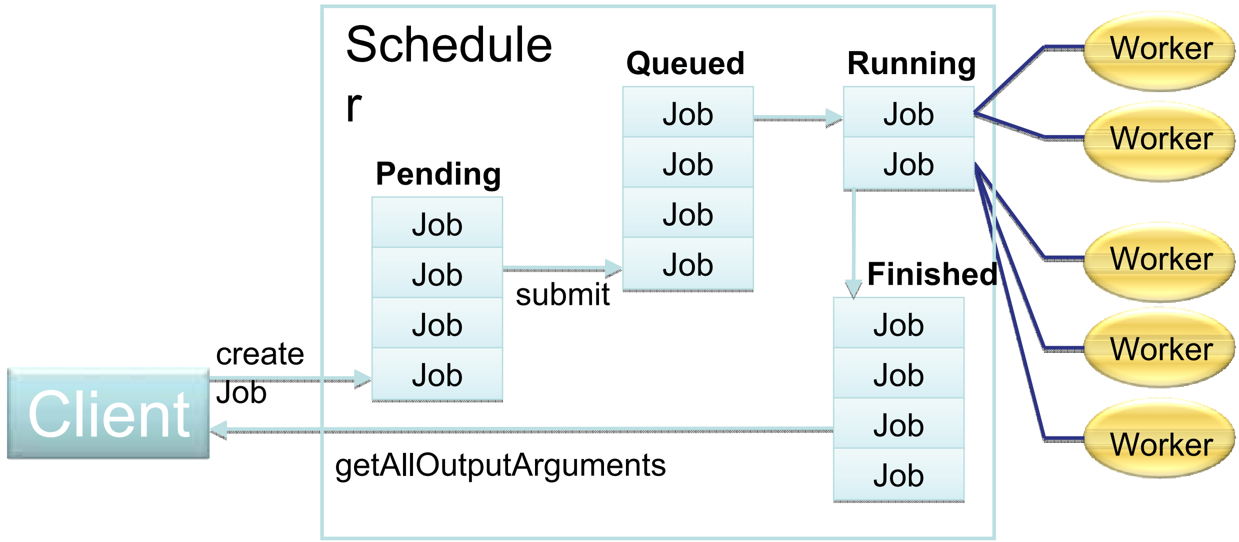
Matlab Distributed Computing Server

Client Parallel Computing Toolbox → Scheduler or Job manager →

- **Worker** Distributed Computing Server
- **Worker** Distributed Computing Server
- **Worker** Distributed Computing Server

- Local job manager
- MathWorks job manager
- Third-party schedulers (Windows CCS, PBS Pro, LSF)

Life cycle of a job



MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

Distributed computations

1. Find scheduler or job manager
   ```
   >>sched=findResource('scheduler',
   'type','local');
   ```
2. Create a job
   ```
   >>job=createJob(sched);
   ```
3. Create tasks and associate them with the job
   ```
   >>createTask(job,@rand,1,{3,3});
   >>createTask(job,@eye,1,{4});
   >>createTask(job,@ones,1,{{4},{3}});
   ```
4. Send job to the front
   ```
   >>submit(job);
   ```
5. Wait until job finishes
   ```
   >>waitForState(job);
   ```
6. Gather results
   ```
   >>results=getAllOutputArguments(job)
   ```
7. Destroy job
   ```
   >>destroy(job);
   ```

Parallel computations

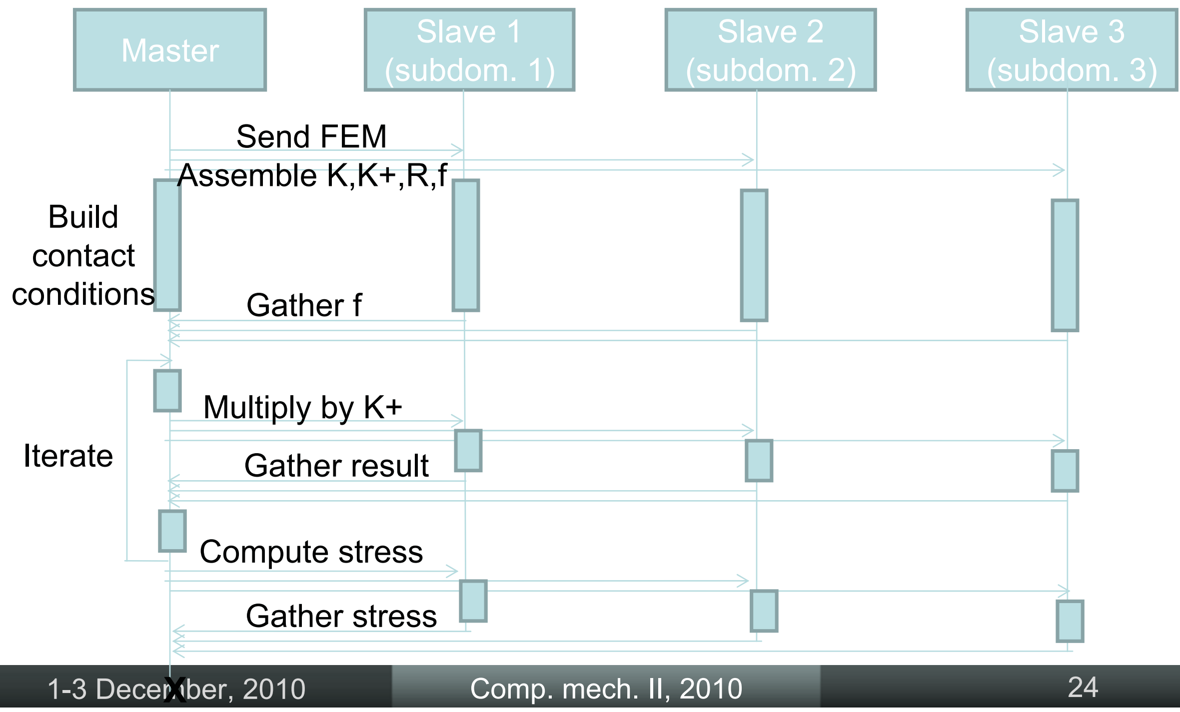
1. Find scheduler or job manager
   `>>sched=findResource('scheduler',… 'type','local');`
2. Create a parallel job and sets the numer of tasks
   `>>pjob=createParallelJob(sched);`
   `set(pjob,'MaximumNumberOfWorkers,4);`
   `set(pjob,'MinimumNumberOfWorkers,4);`
3. Create parallel tasks
   `>>createTask(pjob,@MyParTask,1,{});`
4. Send job to the front
   `>>submit(pjob);`
5. Wait until job finishes
   `>>waitForState(pjob);`
6. Gather results
   `>>results=getAllOutputArguments(pjob)`
7. Destroy job
   `>>destroy(pjob);`

MPT functions: numlabs, labindex, labBarrierlabBroadcast, labProbe, labReceive, labSend, labSendReceive

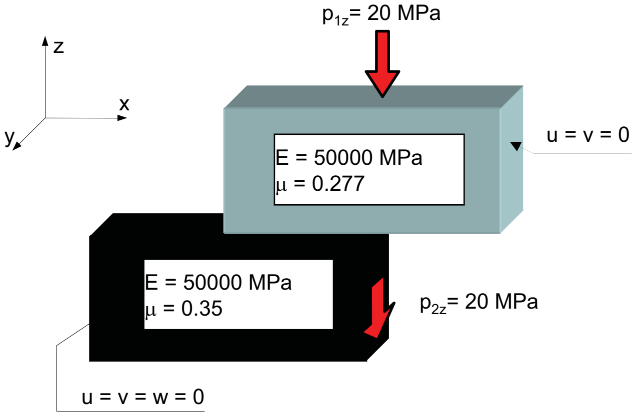
MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

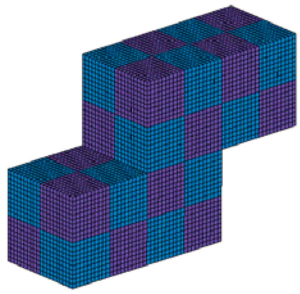
Parallel computation scenario

Send FEM models to workers and parallelize assembling of K, K+, R, fv, stresses, searching contact pairs, multiplication procedures.

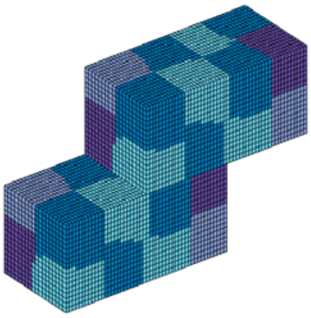


semi-coercive problem

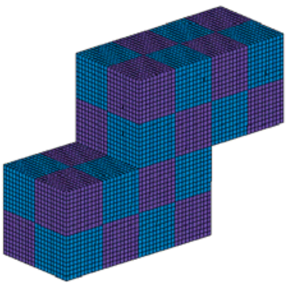




uniform decomposition



METIS decomposition

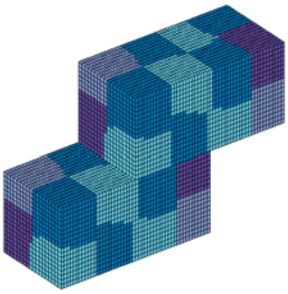


Numerical scalability

Tresca friction (f = 80)

uniform domain decomposition

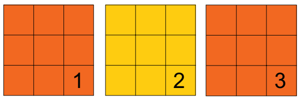
Number of subdomains	4	64	1,024
Number of CPUs	4	24	24
Primal variables	15,972	255,552	4,088,832
Dual variables	1,694	50,634	942,954
Hessian multiplications	76	216	325
Total time (s)	13.81	141.8	4,330.01

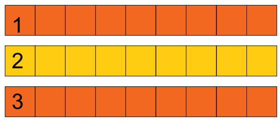


Numerical scalability

Tresca friction (f = 80)
METIS domain decomposition

Number of subdomains	4	64	1024
Number of CPUs	4	24	24
Primal variables	15,972	259,902	4,125,570
Dual variables	1,694	54,984	1,024,898
Hessian multiplications	76	324	1,289
Total time (s)	13.16	206.21	16,865.10

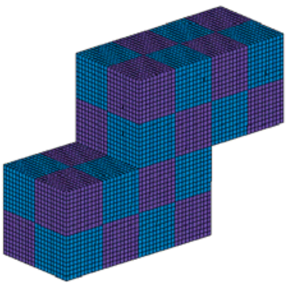




κ = condition number

$\kappa_1 < \kappa_2$

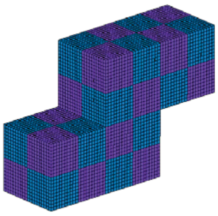
MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution



Numerical scalability

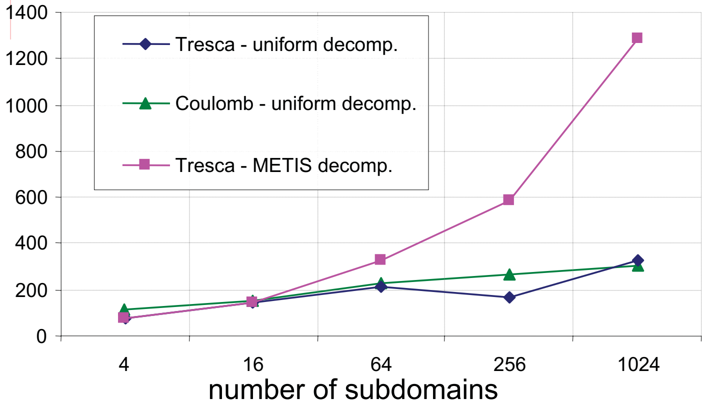
Coulomb friction (f = 0.1)
uniform domain decomposition

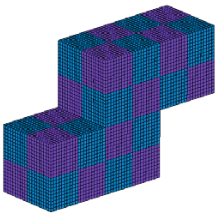
Number of subdomains	4	64	1,024
Number of CPUs	4	24	24
Primal variables	15,972	255,552	4,088,832
Dual variables	1,694	50,634	942,954
Hessian multiplications	115	226	301
Total time (s)	19.34	143.02	4,216.34



Numerical scalability

matrix vector multiplication

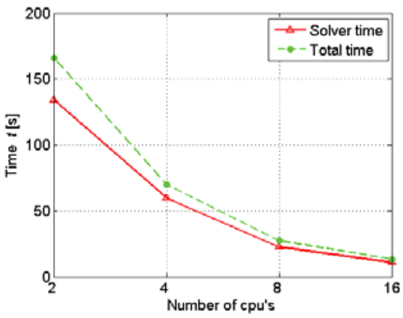




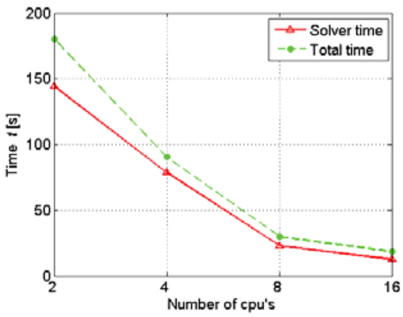
Parallel scalability

fixed number of elements 11 664

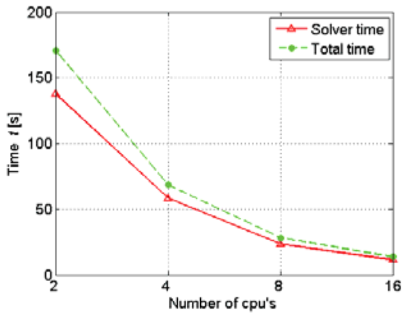
Tresca friction (f = 80)
uniform domain
decomposition



Tresca friction (f = 80)
METIS domain
decomposition



Coulomb friction (f = 0.1)
uniform domain
decomposition



2. ingredient: stable pseudoinverse computation

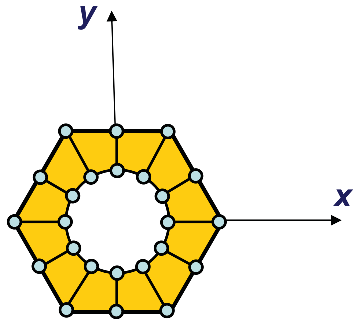
Rigid body motions in 2D elastic structure

- in *x*-direction
- in *y*-direction
- rotation

$$\boxed{Ku = f, \qquad f \in \operatorname{Im} K}$$

$$u = K^{\dagger} f + R\alpha, \quad R^T f = o$$

2D



screw female

$$KK^{\dagger}K = K$$

$$R^T = \begin{bmatrix} 1 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 1 & \cdots & 1 \\ -y_1 & x_1 & -y_2 & x_2 & \cdots & x_n \end{bmatrix}$$

Stable pseudoinverse computation

$$x = K^{\dagger} y \Leftrightarrow Kx = y$$

1. Singular Choleski factorization

$$L_{i,i} < \varepsilon, \quad \varepsilon = ?$$

universal ε doesn't exist

2. Choleski factorization combined with SVD

if a suspected 'zero' pivot appears

use the SVD decomposition proposed by C. Farhat, M. Géradin.

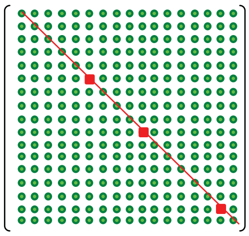
Active choice of the nonsingular and SVD part using fixing nodes.

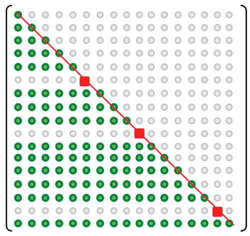
Proposed by Z. Dostal, T. Kozubek, P. Kovar, A. Markopoulos, T. Brzobohatý, Cholesky-SVD decomposition with fixing nodes to stable computation of a generalized inverse of the stiffness matrix of a floating structure, IJNME 2010

Stable pseudoinverse computation

1. Singular Choleski factorization

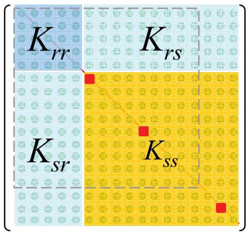
number of zero pivots = defect(K)

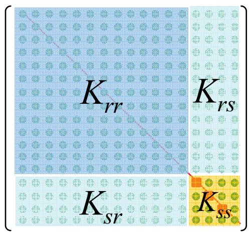
$K =$ 

$L =$ 

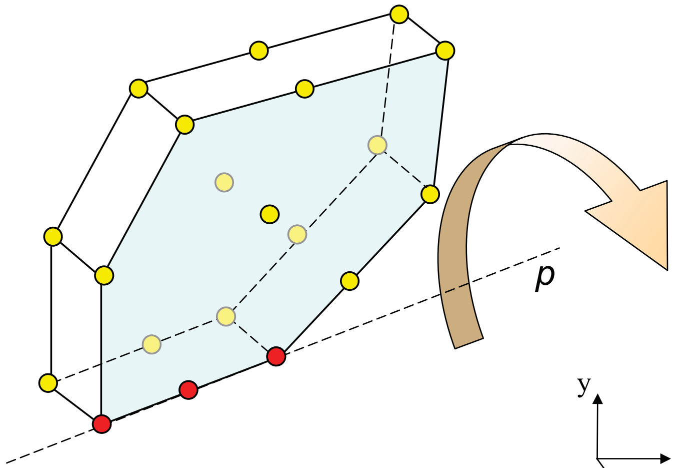
2. Choleski factorization combined with SVD

Active choice of the SVD part

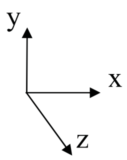
$K =$ 



Stable pseudoinverse computation

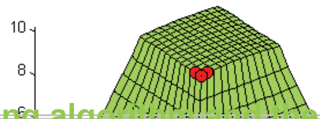


$$PKP^T = \begin{pmatrix} K_{rr} & K_{rs} \\ K_{sr} & K_{ss} \end{pmatrix}$$



P ... reordering into singular and regular parts and sparse reordering

$$K_{rr}u = \vec{1}$$

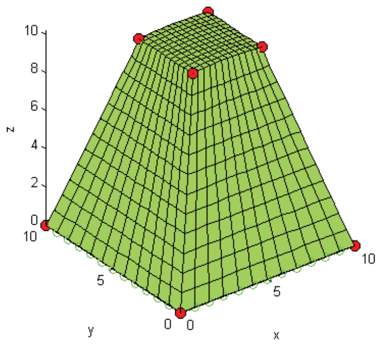


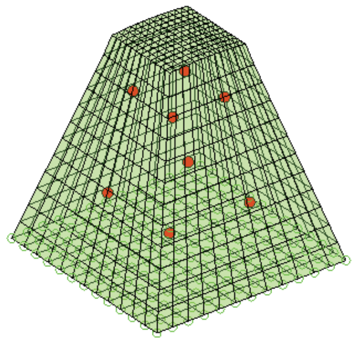
Fixing nodes are chosen using a graph partitioning algorithm and the Perron vector

Z. Dostal, T. Kozubek, P. Kovar, A. Markopoulos, T. Brzobohatý, Cholesky-SVD decomposition with fixing nodes to stable computation of a generalized inverse of the stiffness matrix of a floating structure, submitted to IJNME 2010

Cond=1e8, n.iter=708

Cond=7.5e7, n.iter = 635





Cond=13359, n.iter = 306

Cond=2643, n.iter = 198

Algorithm for choosing *M* uniformly distributed fixing nodes.

Given mesh and $M > 0$.

1.Split the mesh into M submeshes using the mesh partitioning algorithm.
2.Verify whether the resulting submeshes are connected. If not a graph postprocessing may be used to get connected submeshes.
3.Take a node lying near the center of each submesh.

Graph preliminaries

Adjacency matrix *D* – entry *dij* is equal to 1 if the corresponding nodes *i* and *j* are adjacent in the mesh, and zero otherwise.

Walk of length *k* – a sequence of distinct nodes of the given mesh $(v_1, v_2, \ldots, v_k)$ such that all the edges (v_i, v_{i+1}) are present in the mesh for all *i* = 1,2, … ,*k*-1. In other words $d_{v_i, v_{i+1}} = 1 \ \forall i = 1, \ldots, k-1.$

MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

Lemma. Let *D* be the adjacency matrix of a given mesh and let

$$B = D^k.$$

Then each entry *bij* of *B* gives the number of distinct (*i*,*j*)-walks of length *k* in the mesh.

Corollary. Let *D* be the adjacency matrix of a given mesh and

$$e = [e_i],\ e_i = 1,\ \text{i=1,...,n}.$$

Then the number *w*(*i*,*k*) of distinct walks of length *k* starting at node *i* is given by

$$w(i,k) = [D^k e]_i.$$

Remark 1. Since the mesh is approximately regular, we expect that more walks originate from the nodes that are near a center of the mesh rather than from vertices that are far from it.

Remark 2. The node with index i which satisfies

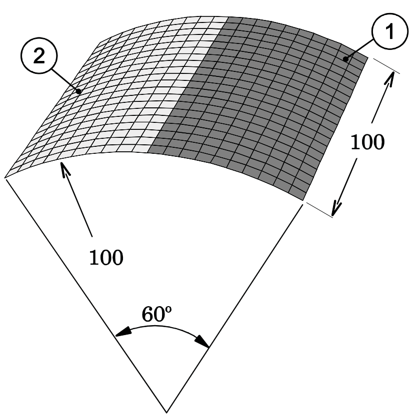
$$w(i,k) \geq w(j,k)\ \forall j$$

for sufficiently large k is in a sense near to the center of the subdomain associated with the submesh.

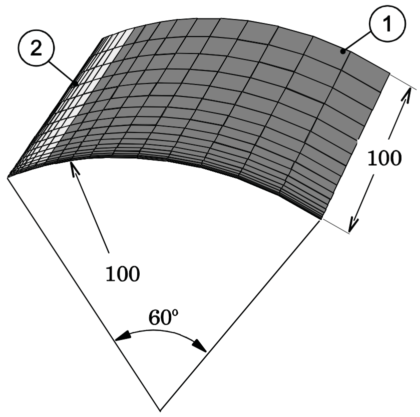
Remark 3. Notice that the vector $p = \lim_{k\to\infty} \left\| D^k e \right\|^{-1} D^k e$ is a unique nonnegative eigenvector which corresponds to the largest eigenvalue of D (known as the Perron vector of D).

MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

Variant A



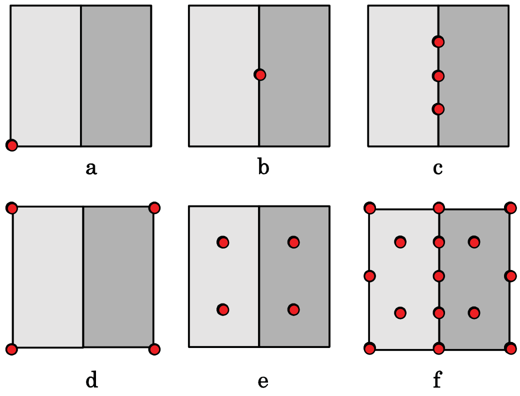
Variant B



material A: E = 2.1e5 MPa, μ = 0.3, t = 1 mm

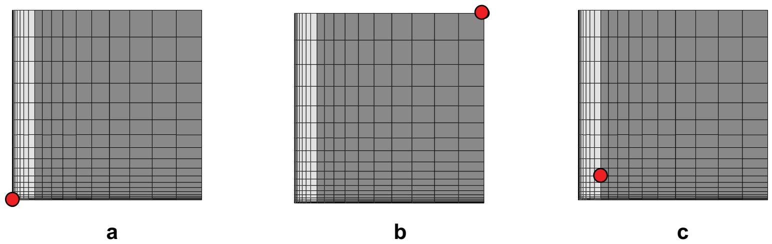
material B: E = 2.1e3 MPa, m = 0.3, t = 1 mm

Variant A



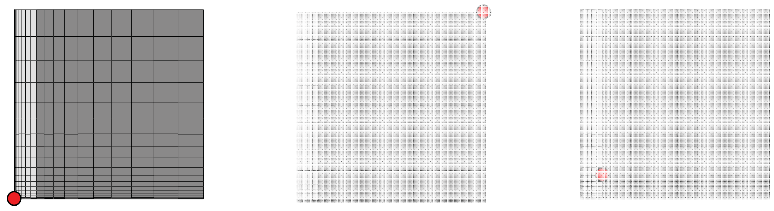
	a	b	c	d	e	f
cond(A)	2.27E+07	2.27E+07	2.27E+07	2.27E+07	2.27E+07	2.27E+07
cond(A_{JJ})	2.79E+11	3.97E+08	3.03E+07	7.52E+06	6.49E+06	9.72E+05
cond(A^{+})	2.79E+11	3.97E+08	3.03E+07	6.60E+07	2.29E+07	2.35E+07

Variant B



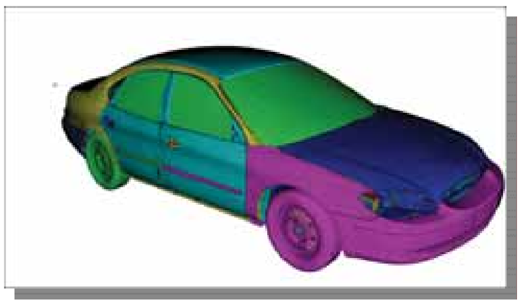
	a	b	c
cond(A)	3.44E+012	3.44E+12	3.44E+12
cond(A_{JJ})	1.79E+18	3.44E+12	3.44E+12
cond(A^{+})	2.62E+18	3.44E+12	3.44E+12

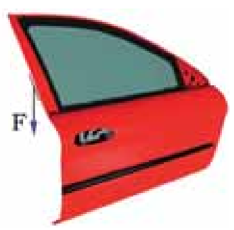
Variant B

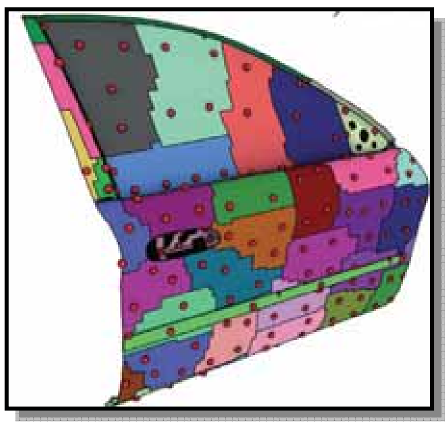


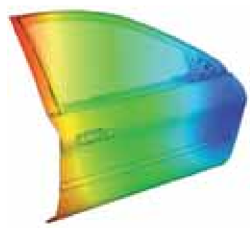
a b c

Matlabs function Chol applied on the matrix A_{jj} of the order 2640 finished with zero pivot at the line 2636. The results indicate that our method can be especially useful for large shells with irregular discretization.





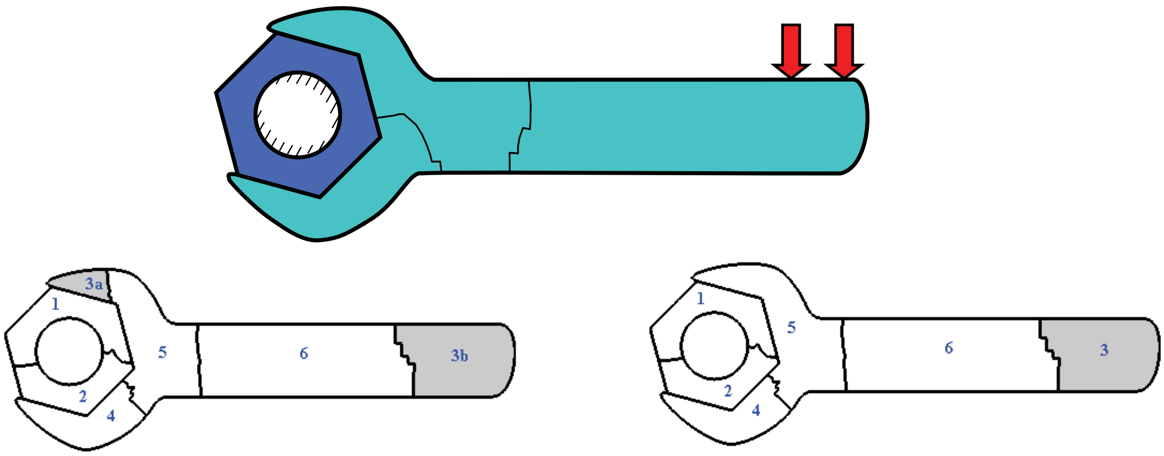




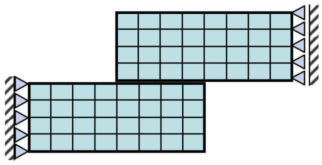
Moore-Penrose pseudoinverse, K is SPS

3. ingredient: domain decomposition correction

Spanner with a screw female

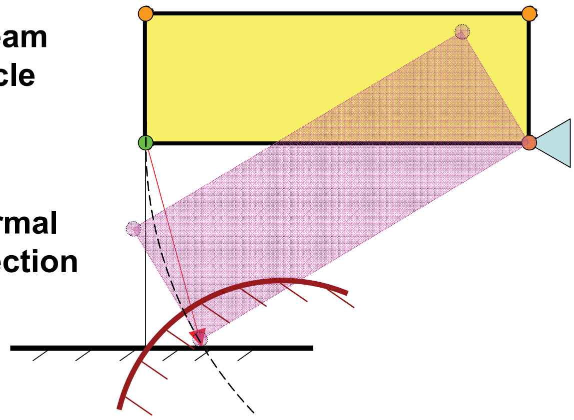


4. ingredient: contact direction correction

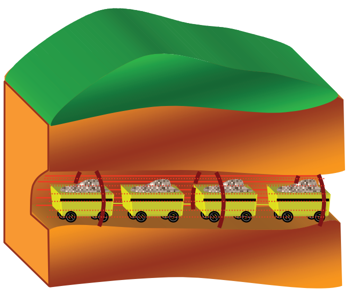


contact
between beam
and obstacle

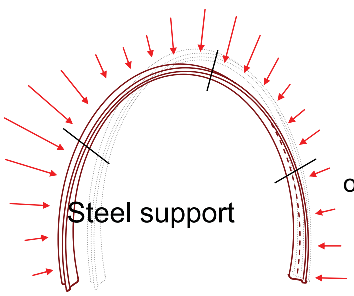
normal
dirrection

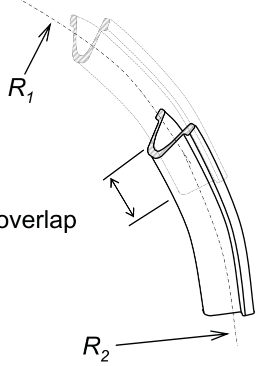


Mining industry: clamp joint of the steel arc support



Pressure arising from the surrounding massif





Coulomb friction

f = 0.1

E = 2.1e5 MPa,

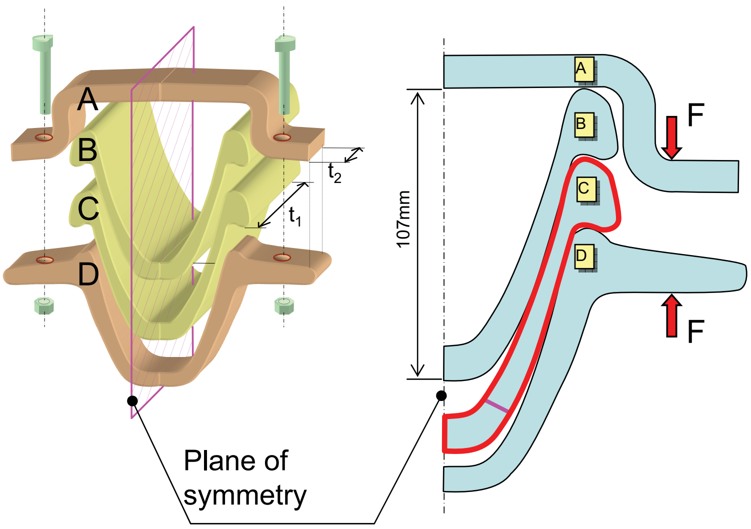
μ = 0.3

$t_A = t_D = 10$ mm

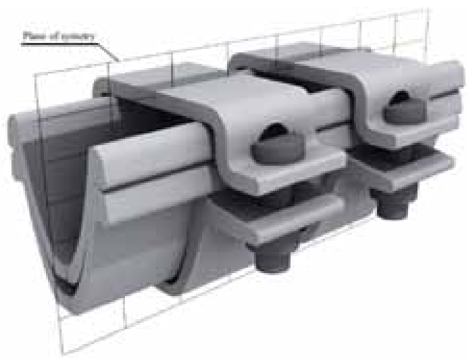
$t_B = t_C = 40$ mm

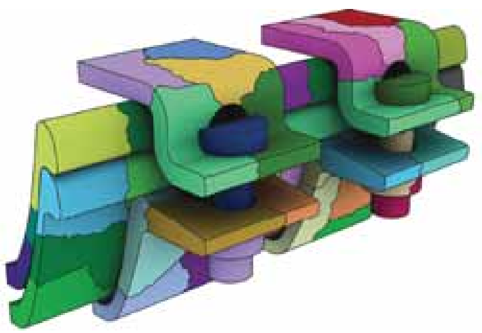
primal variables 65562

dual variables 3112



MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution





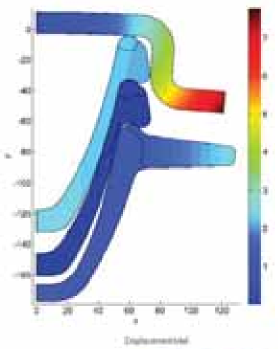
8 bodies
7 floating

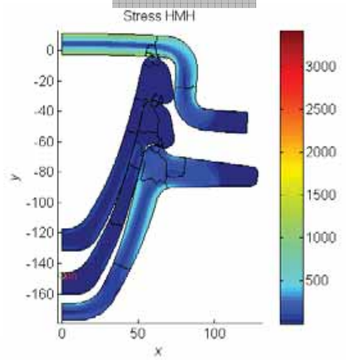
Mining industry: clamp joint of the steel arc support

Displacements

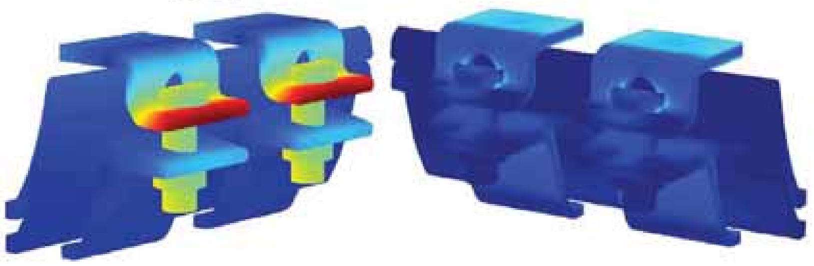
HMH stress

2D

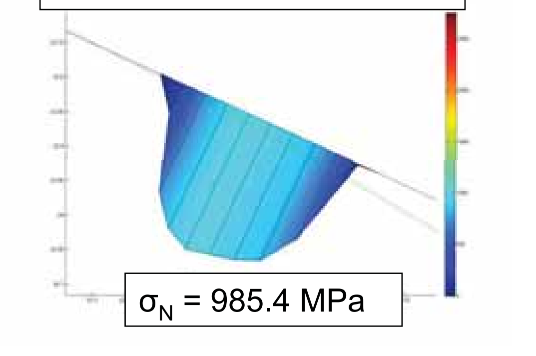




3D

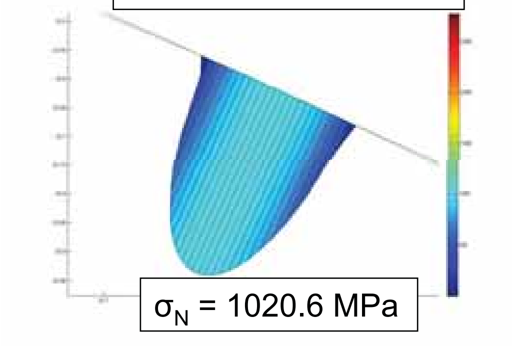


without direction corrections

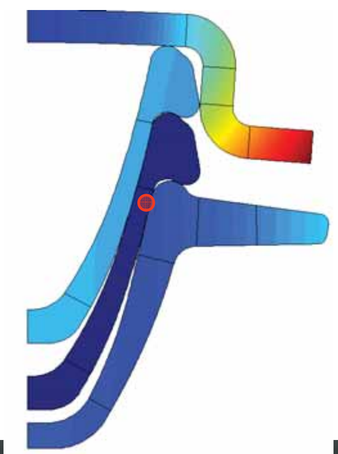


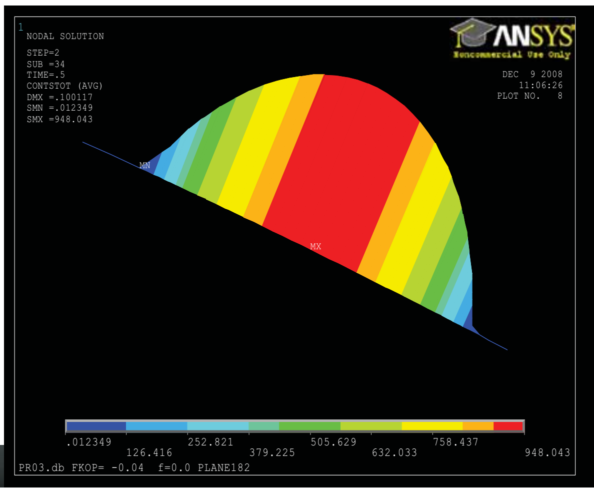
σ_N = 985.4 MPa

3 direction corrections



σ_N = 1020.6 MPa





PARALLEL ANSYS dynamic solution: 27 hours

PARALLEL ANSYS static solution: 1 hour

MATSOL static with direction corrections: 15 mins
seq. version 7 mins

Comparing TBETI and TFETI

250 subdomains f = 0.1

Solution time	2.41 hr.
Total time	2.55 hr.
matrix-vector multiplications	2 126
primal variables	1 592 853
dual variables	261 553
max(abs(u))	1.303 mm
max(totalU)	1.548 mm

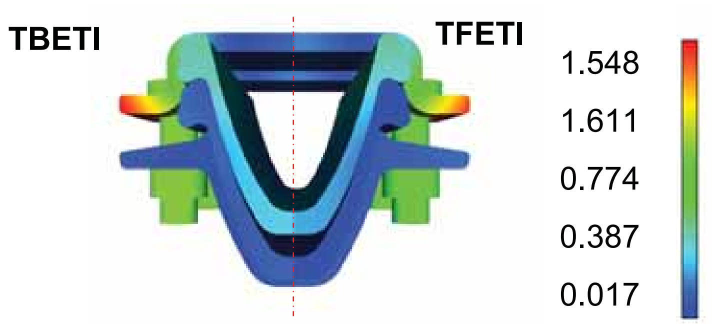
FETI

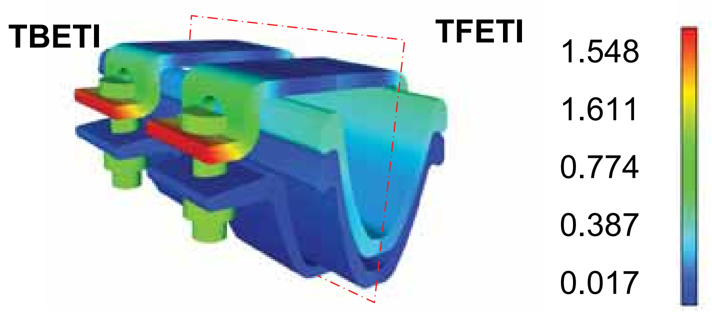
Solution time	2.54 hr.
Total time	3.33 hr.
matrix-vector multiplications	1 882
primal variables	713 751
dual variables	261 553
max(abs(u))	1.378 mm
max(totalU)	1.616 mm

BETI

Comparing TBETI and TFETI

Total displacements [mm]

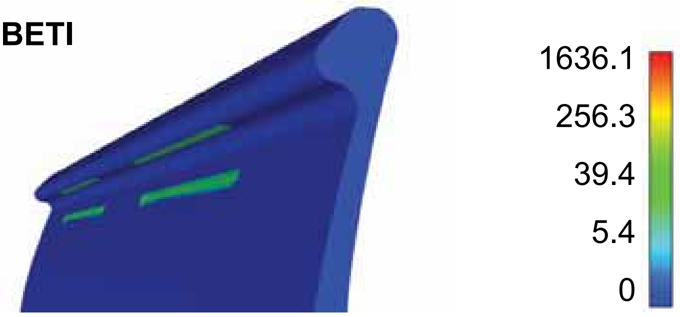




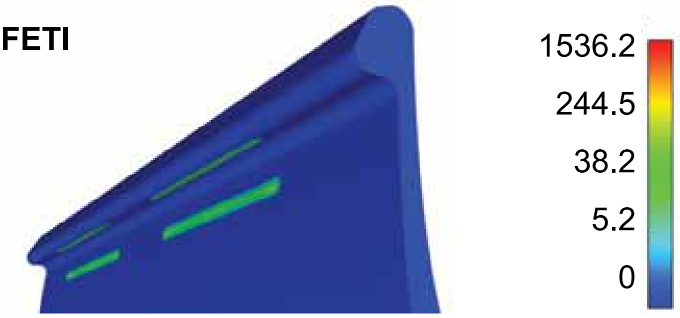
Comparing TBETI and TFETI

Normal contact stress [MPa]

TBETI



TFETI

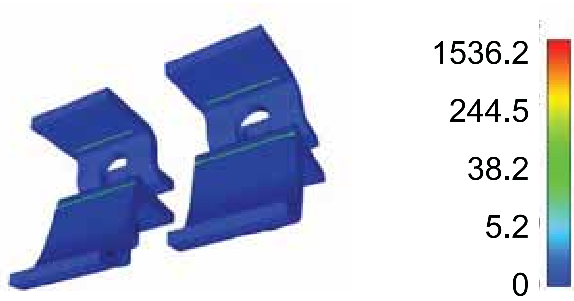


Comparing TBETI and TFETI

Normal contact stress [MPa]

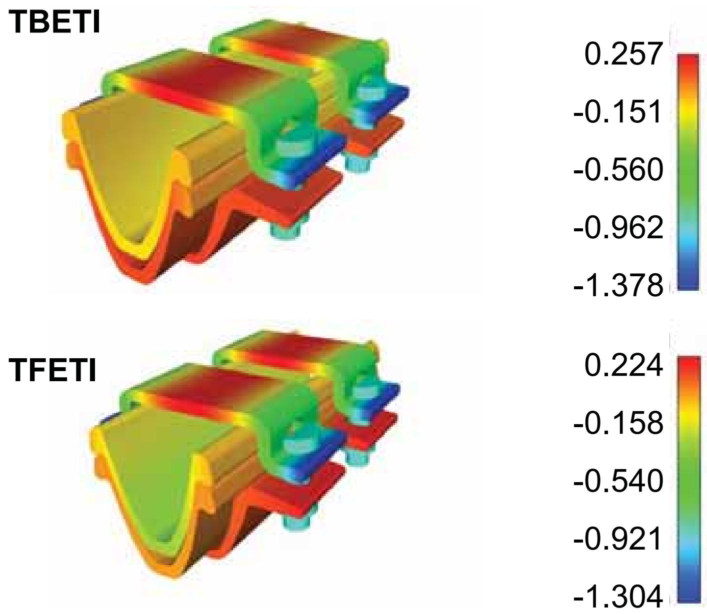
TBETI

TFETI



Comparing TBETI and TFETI

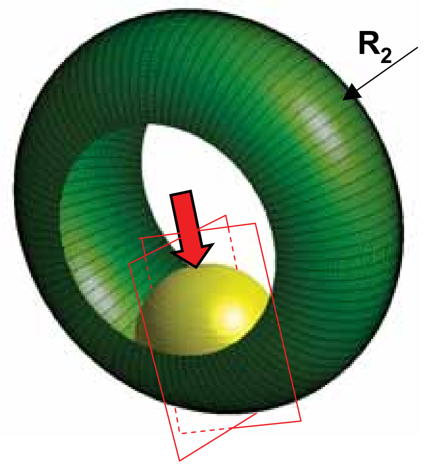
Displacement u_y
[mm]

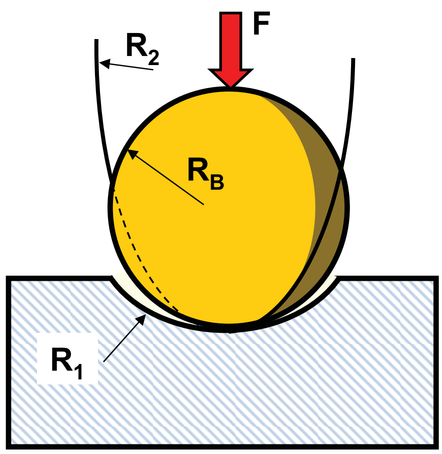


Comparing TBETI and TFETI

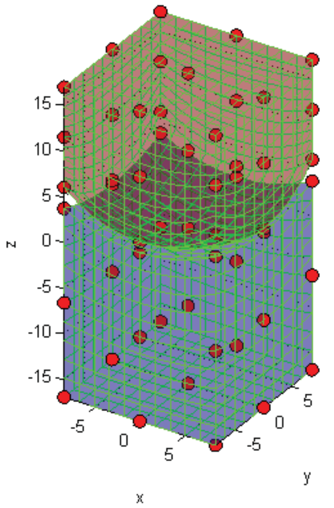
R_1 = 15 mm
R_2 = 17.5 mm
R_3 = 25 mm

$E_1 = E_2$ = 2.1e5 MPa
$\mu_1 = \mu_2$ = 0.3
F = 5 kN





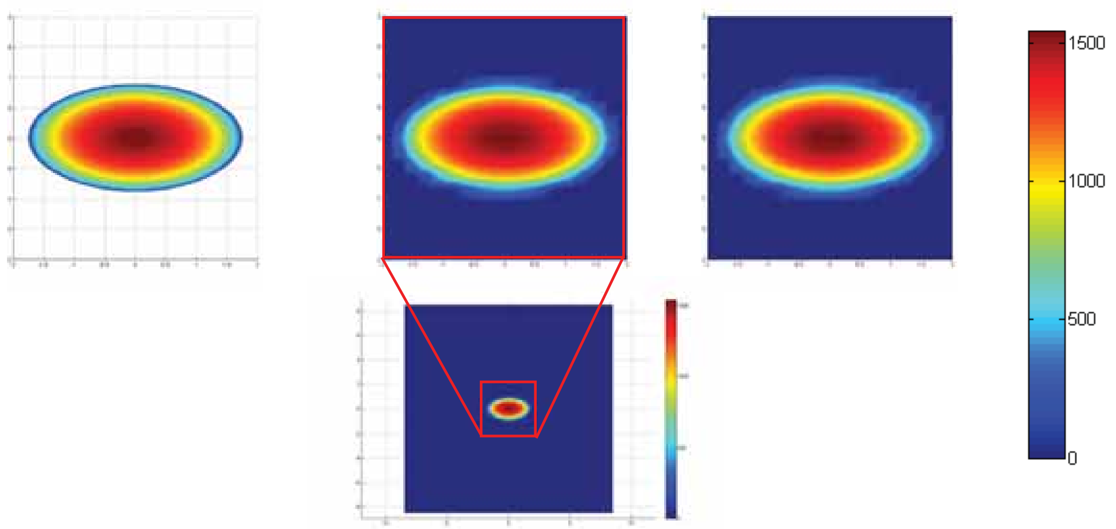
Comparing TBETI and TFETI



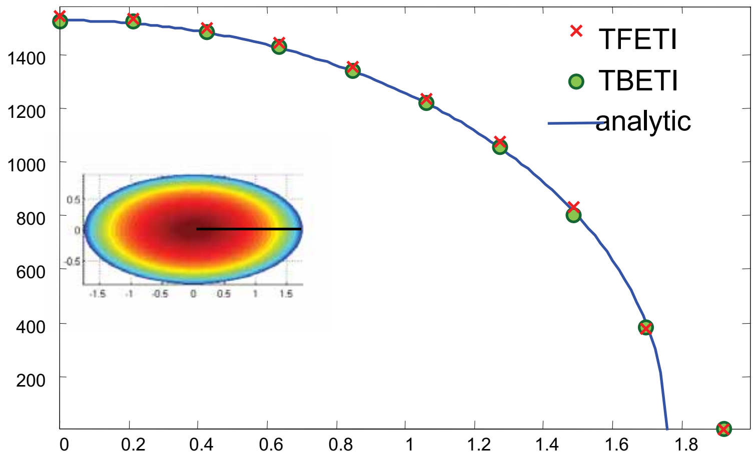
Method	TFETI	TBETI
Solution time	1.82 hr.	1.51 hr.
Total time	2.17 hr.	3.06 hr.
subdomains	1 024	1 024
matrix-vector multiplications	593	667
primal variables	4 088 832	1 849 344
dual variables	926 435	926 435

Comparing TBETI and TFETI

analytical solution **TFETI** **TBETI**

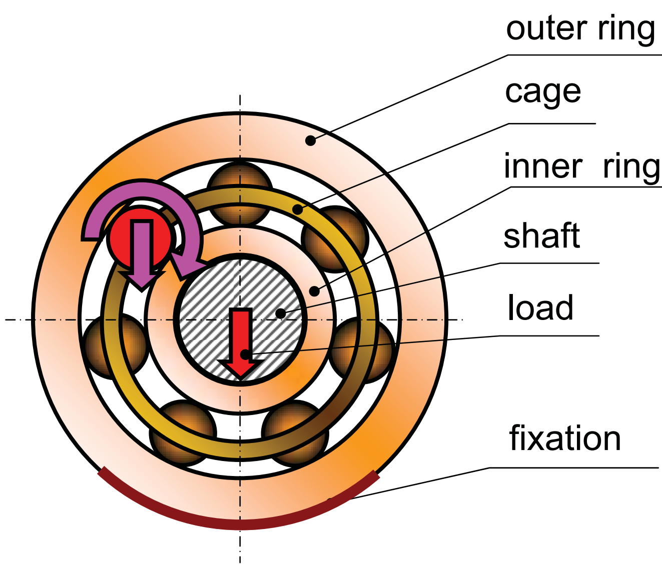


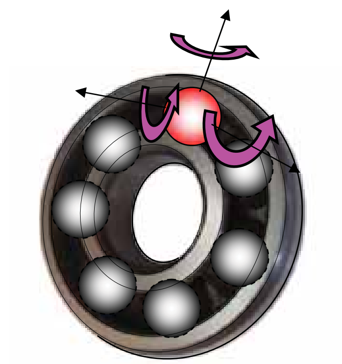
Comparing TBETI and TFETI



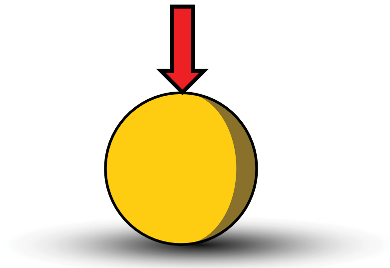
max contact stress
TFETI relative error = 1.1759 %
TBETI relative error = 0.6324 %

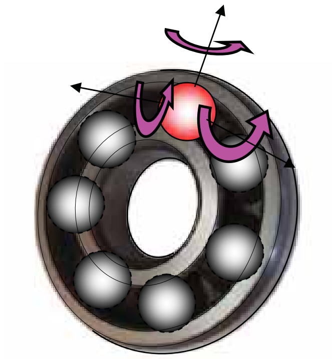
5. ingredient: fixing rigid body motions





Fixing rigid body motions

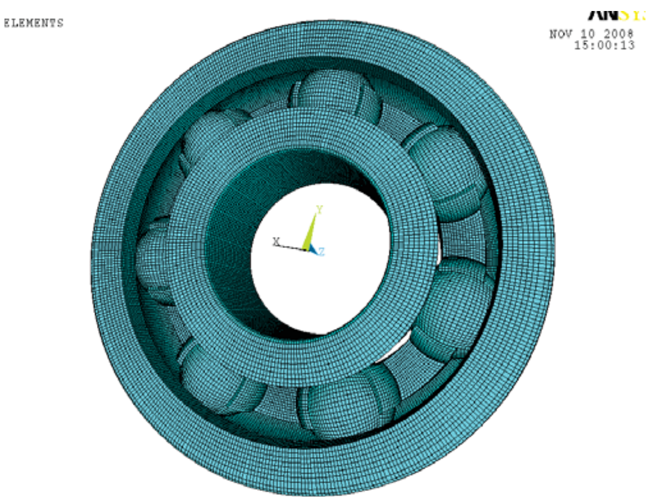




$$\min \tfrac{1}{2} u^T K u - u^T f$$
$$B_I u \le c_I$$
$$B_E u = o$$
$$B_B u = c_B$$

additional condition
$$B_r u = o$$

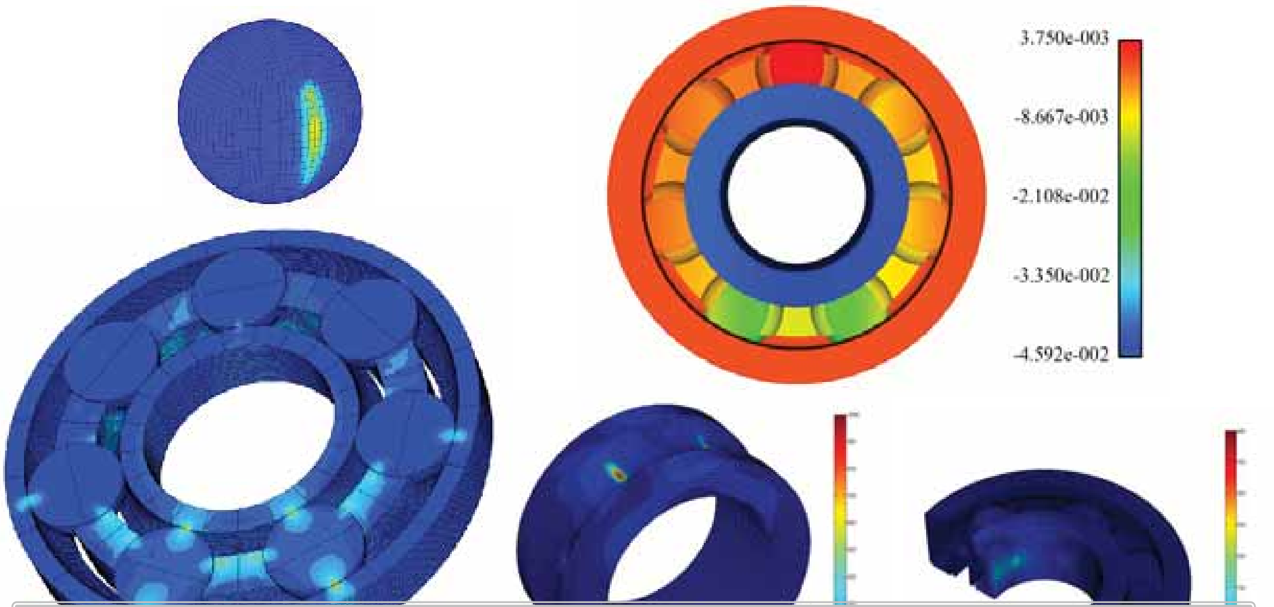
$$R^T = \begin{array}{c}\left[\begin{array}{cccccccc} 1 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & \cdots & 1 \\ -y_1 & x_1 & 0 & -y_2 & x_2 & 0 & \cdots & 0 \\ -z_1 & 0 & x_1 & -z_2 & 0 & x_2 & \cdots & x_n \\ 0 & -z_1 & y_1 & 0 & -z_2 & y_2 & \cdots & y_n \end{array}\right]\end{array} \begin{array}{l} \text{x} \\ \text{y} \\ \text{z} \\ \text{xy} \\ \text{xz} \\ \text{yz} \end{array}$$



- 1 688 000 primal variables
- 408 000 dual variables
- 10 bodies, 9 floating
- 700 subdomains
- 2364 matrix-vector multiplications

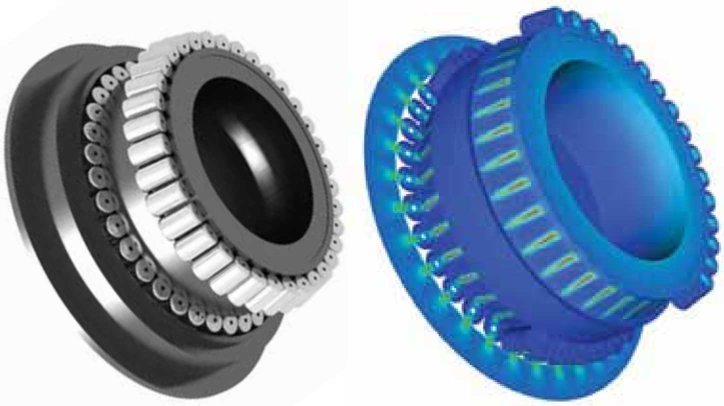
	Parallel
Solution time	1.75 hr
Total time	1.83 hr

Car industry: ball bearing



Scalable TFETI algorithm for the solution of multibody contact problems of elasticity (IJNME 2009), Z. Dostál, T. Kozubek, V. Vondrák, T. Brzobohatý , A. Markopoulos

Roller bearing of wind generator



Statistics	
Bodies	73
Subdomains	**700**
Primal variables	2,73 M
Dual variables	459,8 k
Iterations	4270

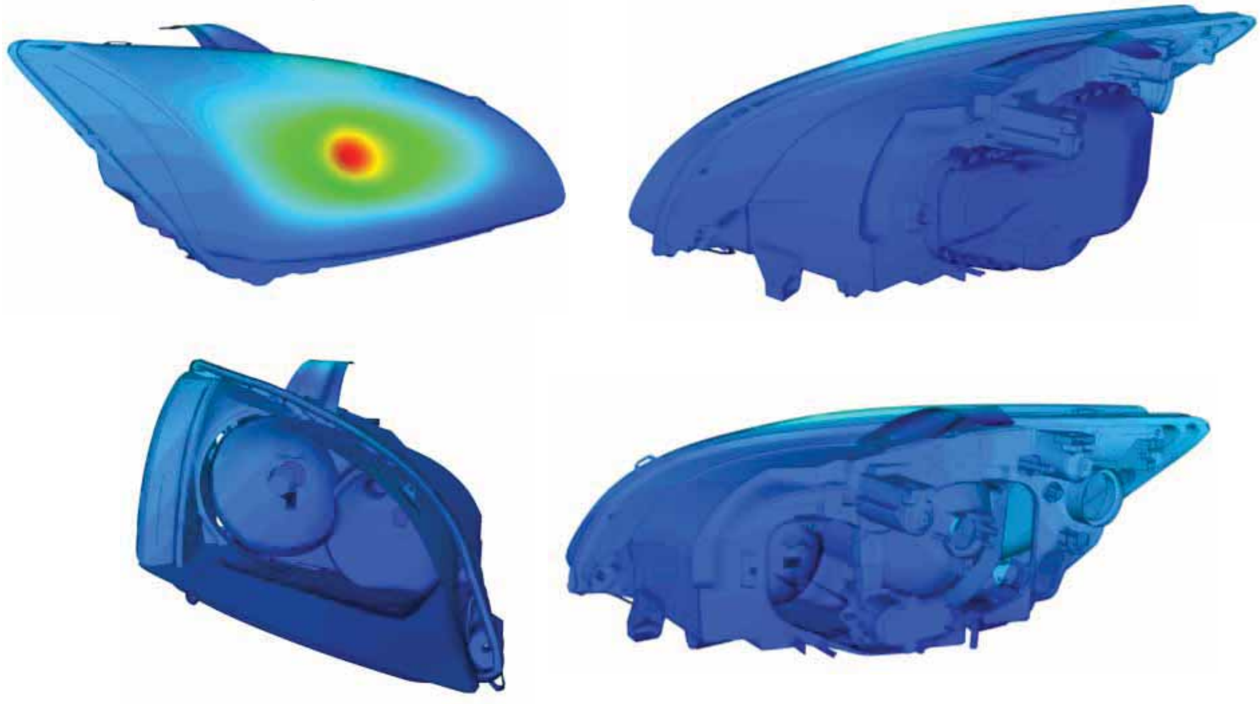
Visteon Headlight

12 bodies

Visteon Headlight

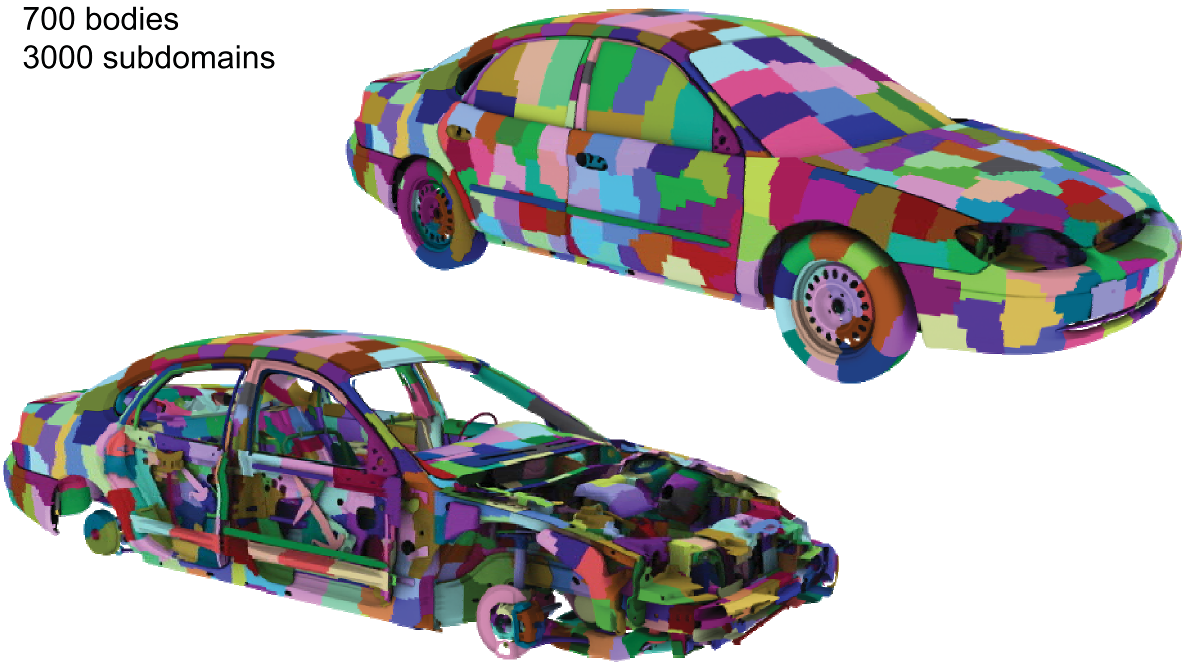
64 subdomains

Visteon Headlight - Total Displacement



MatSol
Department of Applied Mathematics
VSB-Technical University, Ostrava, CZ
Multibody Contact | Domain Decomposition | Parallel Solution

700 bodies
3000 subdomains



Extension to dynamics

Contact stabilized Newmark scheme combined with MPGRP algorithm

R. Krause, M. Walloth, *A Time Discretization Scheme Based on Rothe's Method for Dynamical Contact Problems with Friction* -Computer Methods in Applied Mechanics and Engineering, Vol. 199, pages 1-19 (December 2009). - Available as INS Preprint No. 0802.

begin

$$\left[\text{for } \tau \in \langle 0, T\rangle,\ 0 = \tau_0 < \tau_1 \ldots < \tau_n = T, \tau_i = i\Delta\tau\right]$$

Step 1. { *Solution of predictor displacement*}

$$\min\left[\frac{1}{2}\left(u_{\tau+\Delta\tau}^{pred}\right)^T M\left(u_{\tau+\Delta\tau}^{pred}\right) - \left(Mu_\tau + \Delta\tau M \dot{u}_\tau\right)^T u_{\tau+\Delta\tau}^{pred}\right]$$

$$\text{subject to } B_I u_{\tau+\Delta\tau}^{pred} \le c_I, \text{ and } B_E u_{\tau+\Delta\tau}^{pred} = c_E$$

Step 2. { *Solution of contact stabilized displacement*}

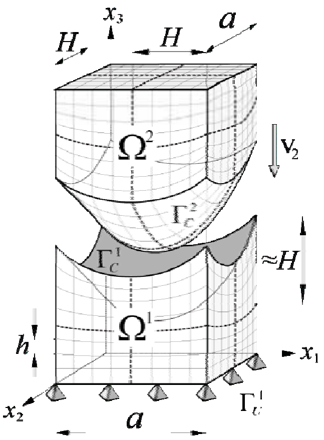
$$\min\left[\frac{1}{2}\left(u_{\tau+\Delta\tau}\right)^T K\left(u_{\tau+\Delta\tau}\right) - \left(\frac{2}{\Delta\tau^2} Mu_{\tau+\Delta\tau}^{pred} - \frac{1}{2}Au_\tau + \frac{1}{2}\left(f_{\tau+\Delta\tau} + f_\tau\right)\right)^T u_{\tau+\Delta\tau}\right]$$

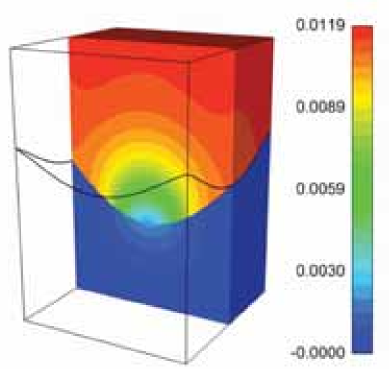
$$\text{subject to } B_I u_{\tau+\Delta\tau} \le c_I \text{ and } B_E u_{\tau+\Delta\tau} = c_E$$

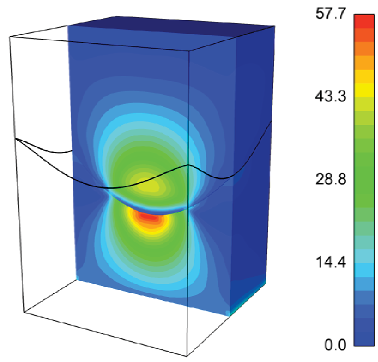
Step 3. { *Solution of velocity*}

$$\dot{u}_{\tau+\Delta\tau} = \dot{u}_\tau + \frac{2}{\Delta\tau}\left(u_{\tau+\Delta\tau} - u_{\tau+\Delta\tau}^{pred}\right)$$

end



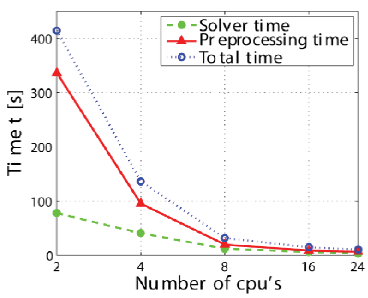


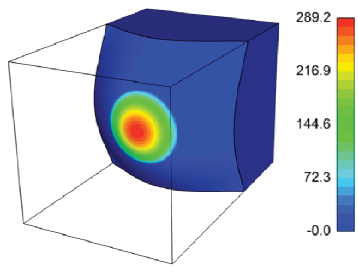


2 739 198 primal

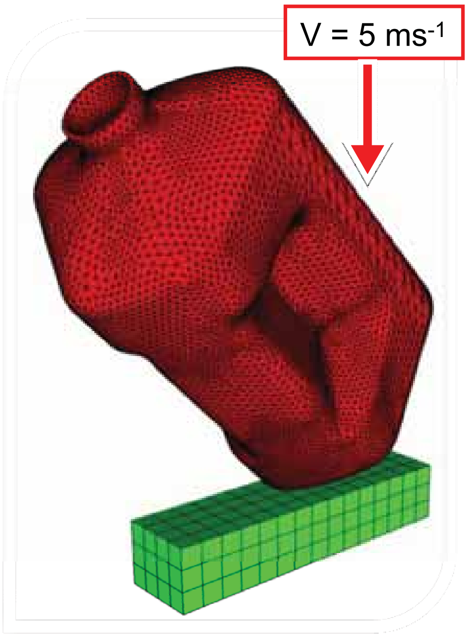
596 771 dual

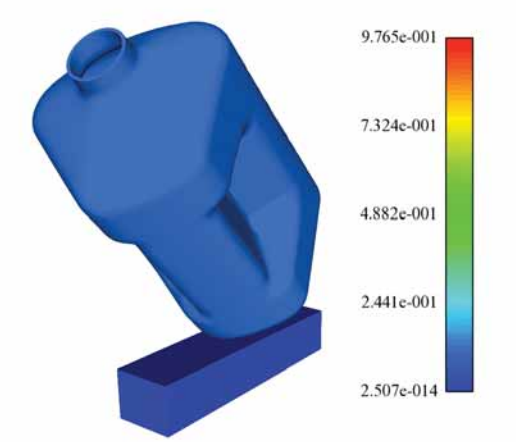
Solver time of one step 50s



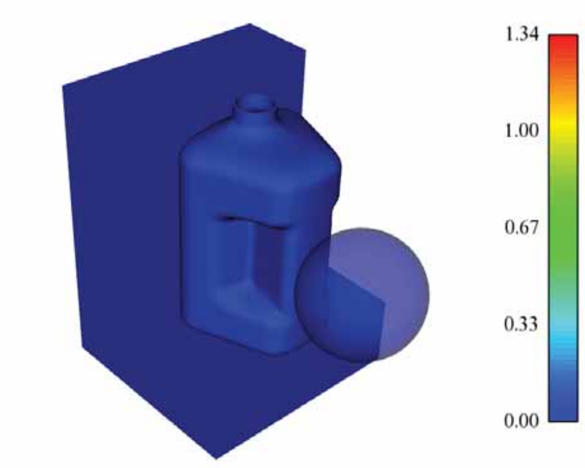


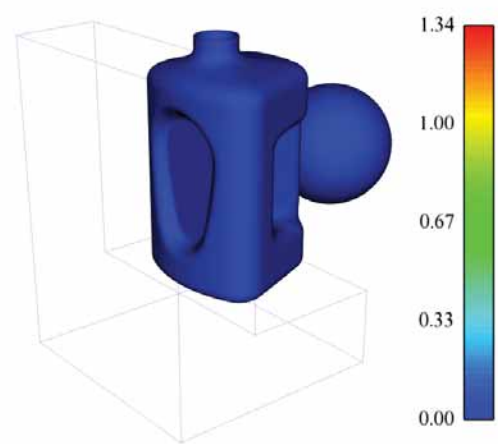
Extension to dynamics





Extension to dynamics





1. Brzobohaty, T., Dostal, Z., Kovar, P., Kozubek, T., Markopoulos, A.: Cholesky–SVD decomposition with fixing nodes to stable evaluation of a generalized inverse of the stiffness matrix of a floating structure. Accepted for publishing in IJNME, 2010.
2. Dostal, Z., Kozubek, T., Vondrak, V., Brzobohaty, T., Markopoulos, A.: Scalable TFETI algorithm for the solution of multibody contact problems of elasticity. Int J Numer Meth Eng 82,No. 11, p. 1384-1405 (2010),
3. Dostal, Z., Kozubek, T., Horyl, P., Brzobohaty, T., Markopoulos, A.: Scalable TFETI algorithm for two dimensional multibody contact problems with friction. J Comput Appl Math. 2010, 235(2) (2010), 403-418.
4. Dostal, Z., Kozubek, T., Markopoulos, A., Brzobohaty, T., Vondrak, V., Horyl, P.: Theoretically supported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction. Accepted for publishing in CMAME 2010.
5. Dostal, Z., Kozubek, T., Markopoulos, A., Brzobohaty, T., Vlach, O. Scalable TFETI with preconditioning by conjugate projector for transient contact problems of elasticity, in preparing 2010.

Thank you for your attention

Efficient parallel contact shape optimization

V. Vondrák, T. Kozubek, A. Markopoulos, M. Sadowská and Z. Dostál

Dept. of Applied Math., VŠB-TU Ostrava

Czech Republic

Outline

- *Contact shape optimization problem*
  - Parallel sensitivity analysis
  - 3D Hertz optimization problem
  - Numerical results
- *Total FETI & BETI domain decomposition*
  - Parallel solution of state problem
  - BETI & FETI methods
  - Sensitivity analysis
- *Conclusions and future work*

Contact shape optimization

$$\begin{cases} \min \Im(\alpha, u(\alpha)) \\ \alpha \in U_{ad} \end{cases}$$

$\Im(\alpha, u(\alpha))$... objective function

$u(\alpha)$ solves contact problem

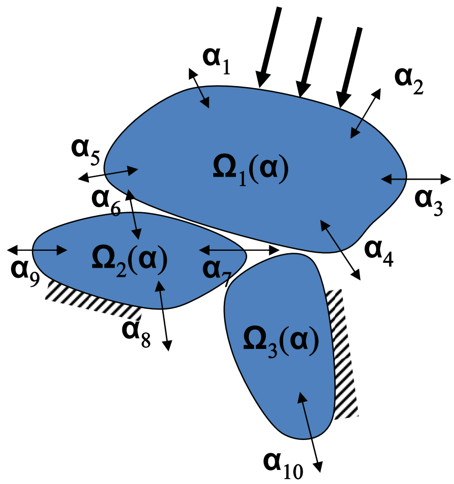
Variational inequality

$$a_\alpha(u, v-u) \geq b_\alpha(v-u), \quad \forall v \in C_\alpha$$

$$\downarrow$$

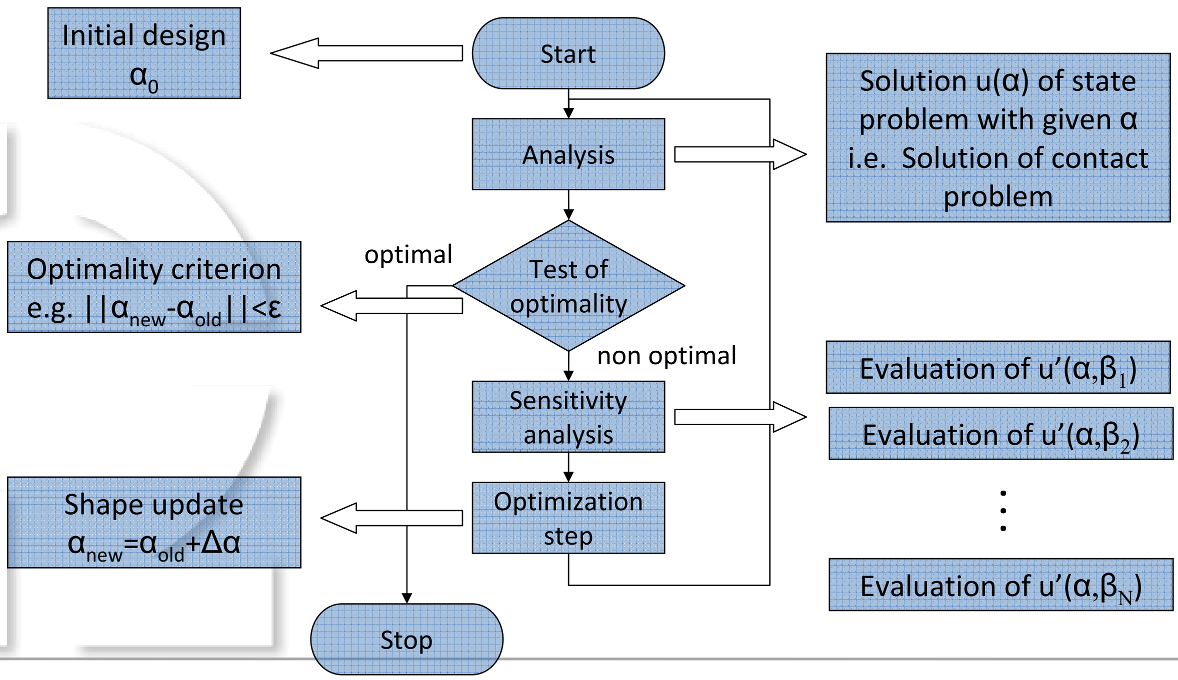
$$\min \tfrac{1}{2} u^T K(\alpha) u - u^T f(\alpha),$$

s.t. $B(\alpha) u \leq c(\alpha)$



Department
of Applied Mathematics
December 2, 2010

General shape optimization scheme



Solution using MatSol

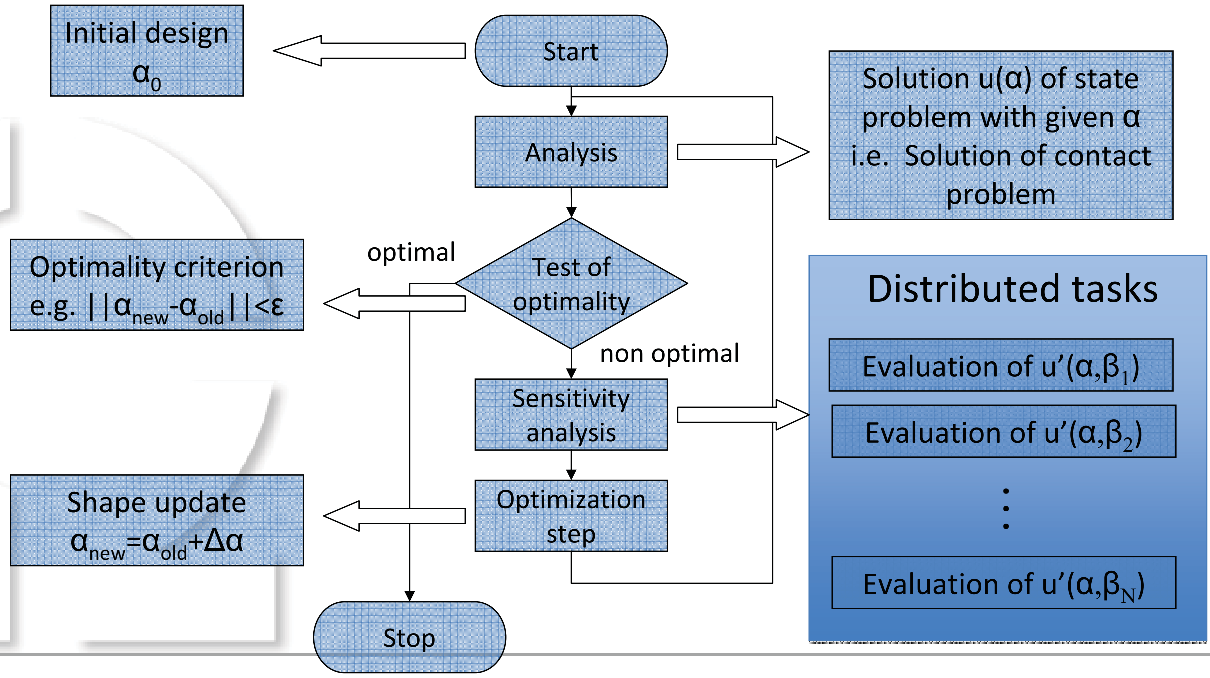
function fem=problem(alpha)
Settings of
fem.geom …Geometry def.
fem.mesh …Mesh informations
fem.opt
- *fem.opt.dv …Design variables*
- *fem.opt.lb … Lower bounds*
- *fem.opt.ub … Upper bounds*
- *fem.opt.Aeq…Equality constr.*
- *fem.opt.beq*
- *fem.opt.Aiq …Inequality constr.*
- *fem.opt.biq*

```
function
    alphaopt=optimize(objective)
…
opt=feval(problem)
…
alphaopt = fmincon(objective,
    opt.dv, opt.Aiq, opt.biq,
    opt.Aeq, opt.biq, opt.lb, opt.ub,
    optimoptions)
```

Function [f,g]=objective(alpha)
Solves state problem and sensitivity analysis

How to speed-up optimization process?

General shape optimization scheme



Department
of Applied Mathematics
December 2, 2010

Finite difference sensitivity analysis

$$\frac{\partial u(\alpha)}{\partial \alpha_i} \approx \frac{u(\alpha + he_i) - u(\alpha)}{h}$$

where

$u(\alpha + he_i)$ solves

$$\min \tfrac{1}{2} u^T K(\alpha + he_i) u - u^T f(\alpha + he_i)$$

subject to $B(\alpha + he_i) u \le c(\alpha + he_i)$

and $e_i = (0,\dots,0,\underset{i}{1},0,\dots,0),\ i = 1,\dots,m$

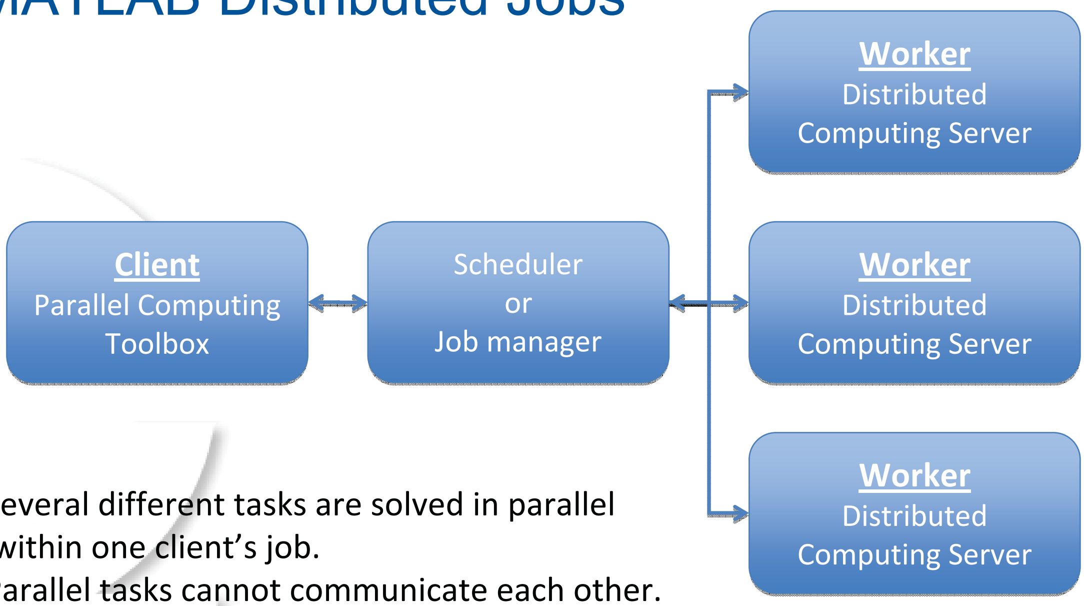
- *Advantage*
  - Simple implementation
- *Disadvantages*
  - m+1 assemblies of stiffness matrix
  - m+1 solution of contact problem (m+1 decompositions of K)
  - numerically unstable

Parallel MATLAB and MatSol

- *Parallel MATLAB – what does it mean?*
  - MATLAB® Distributed Computing Server™
  - Parallel Computing Toolbox™
  - Job managers or schedulers (local job manager, Mathworks job manager, Windows CCS, LSF, PBS Pro, …)
  - http://www.mathworks.com/products/distriben/
- *MatSol uses Parallel Computing Toolbox*
  - Parallel implementation of algorithms
  - Parallel preprocessing and postprocessing
- *Star-P: alternative solution of Parallel MATLAB*
  - http://www.interactivesupercomputing.com/

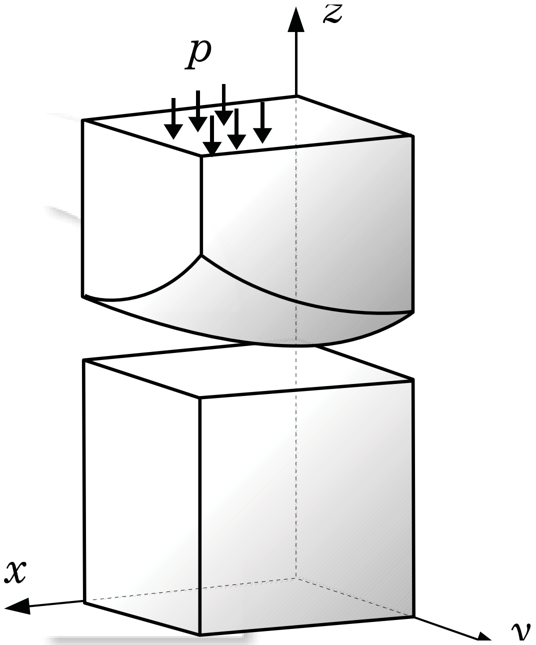
Department
of Applied Mathematics
December 2, 2010

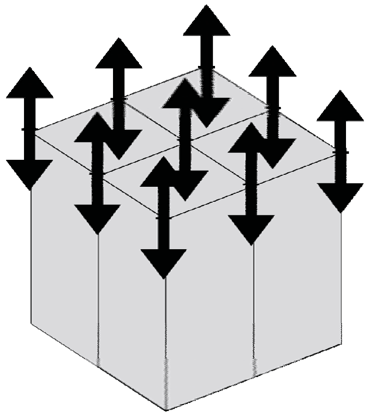
MATLAB Distributed Jobs



Several different tasks are solved in parallel within one client's job.
Parallel tasks cannot communicate each other.

3D Hertz optimization problem



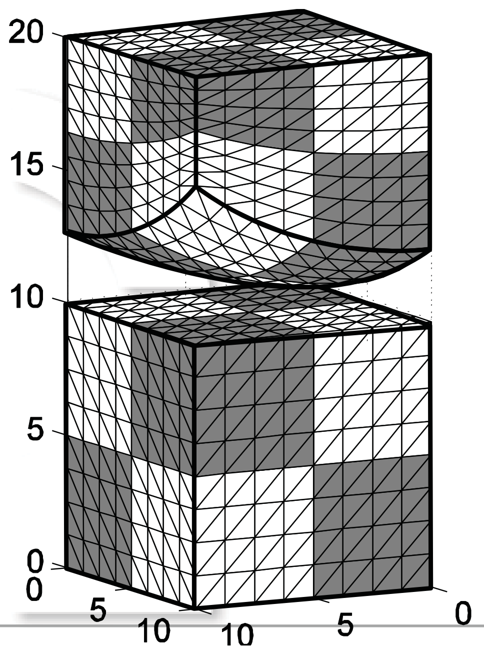


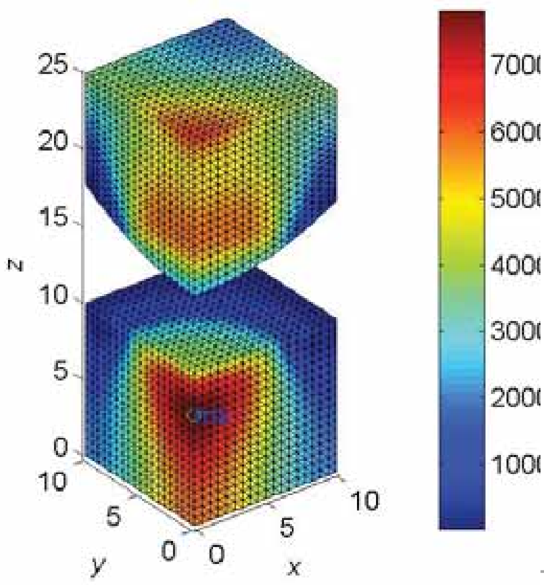
- min ½u^TKu-u^Tf
- 9 and 16 design variables with box constraints
- Vol(Ω)=Vol($Ω_0$)

Department
of Applied Mathematics

December 2, 2010

3D Hertz problem – initial design



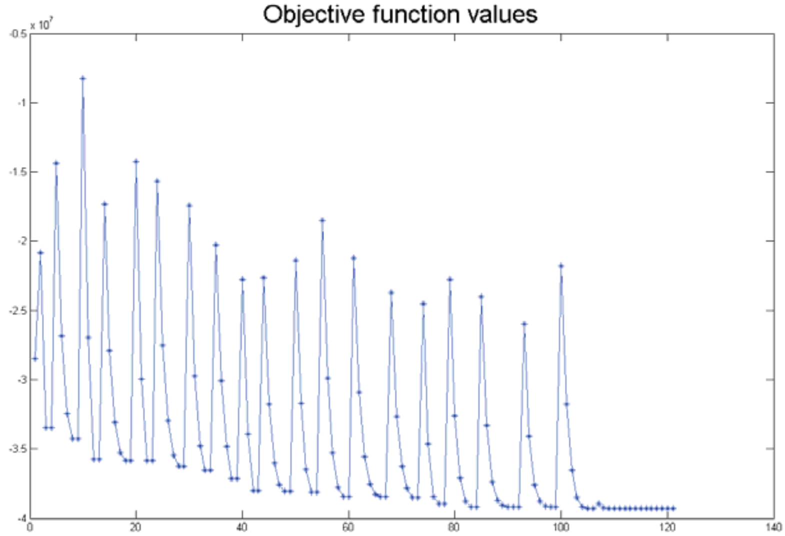


Department
of Applied Mathematics

December 2, 2010

3D Hertz problem – optimized

Cubic spline function with 3x3 nodes = 9 design variables

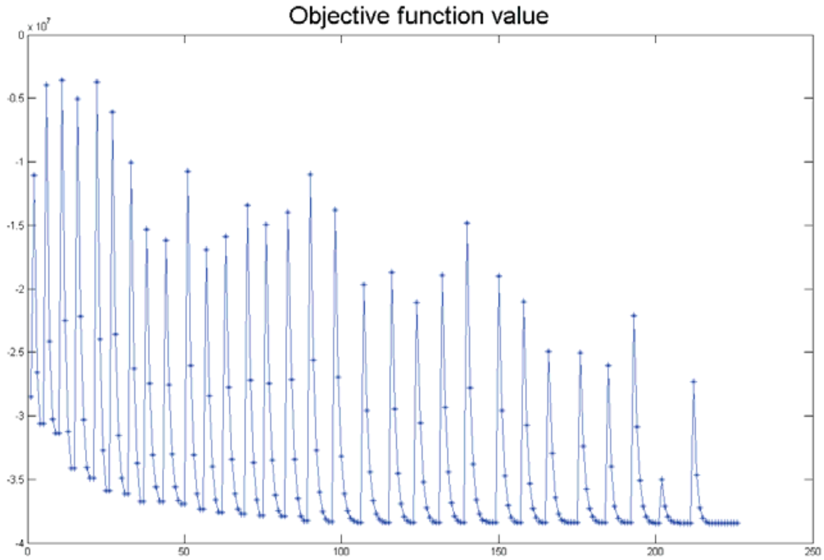


Department
of Applied Mathematics

December 2, 2010

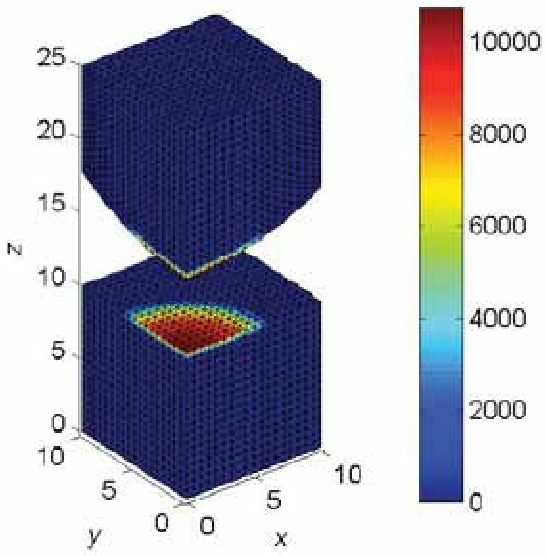
3D Hertz problem – optimized

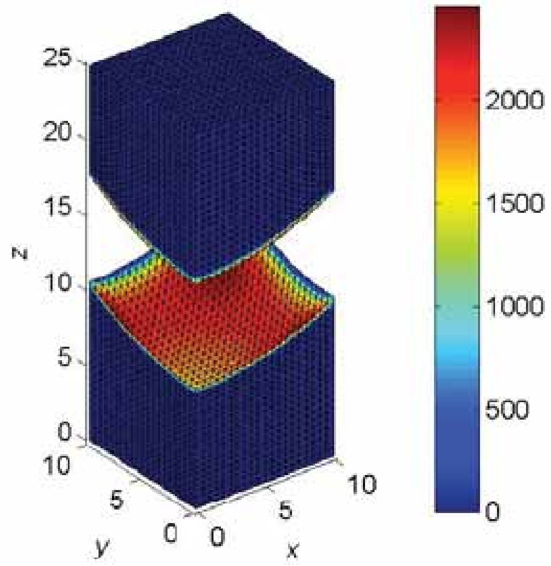
Cubic spline function with 4x4 nodes = 16 design variables



3D Hertz problem – contact pressure

Initial design *Optimized design*



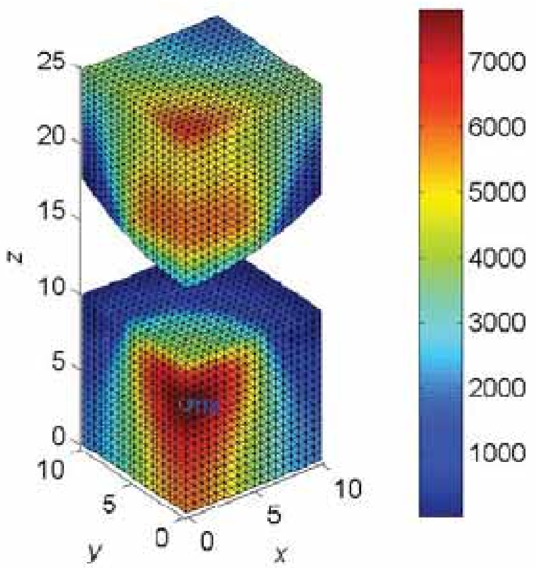


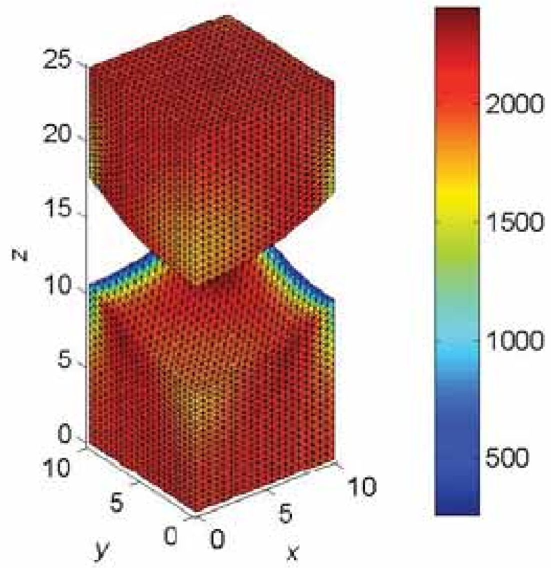
Department
of Applied Mathematics

December 2, 2010

3D Hertz problem – von Mises stress

Initial design *Optimized design*





Parallel speed-up

	State problem	Sensitivity analysis		
Úloha		Sequential	Parallel	Speed-up
2D Hertz 5DV	0.4s	2s	5s	0.4x
2D Hertz 10DV	0.4s	4s	6s	0.67x
3D Hertz 9DV	15s	135s	30s	4.5x
3D Hertz 16DV	15s	240s	40s	6x

Cluster COMSIO
HP Blade server, 18x AMD Opteron Dual Core 1.8GHz,
16x4GB +2x6GB, total 76GB
Infiniband interconnect
MATLAB Distributed Computing Server – 24 licences

Number of parallel processes is limited by number of design variables!

☹

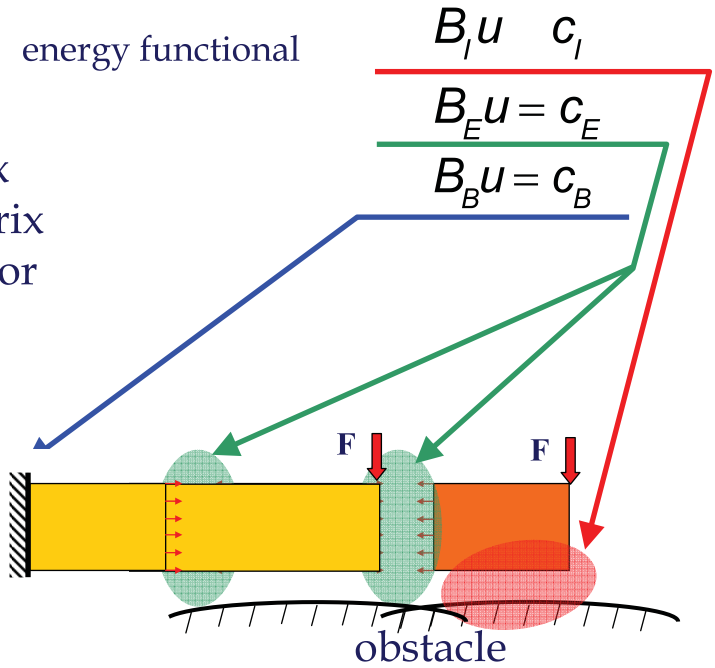
Total FETI – primal formulation

$\min \frac{1}{2} u^T K u - u^T f$ energy functional

u	displacement
K	stiffness matrix
B	constraint matrix
c	constraint vector

Total FETI

Dirichlet b.c. are enforced by Lagrange multipliers



Total FETI solution of state problem

$$\min \tfrac{1}{2} \lambda^T F(\alpha)\lambda - \lambda^T d(\alpha) \text{ subject to } \lambda_I \geq o,\ E(\alpha)\lambda = g(\alpha)$$

$$F(\alpha) = B(\alpha) K^+(\alpha) B(\alpha)^T$$
$$d(\alpha) = B(\alpha) K^+(\alpha) f(\alpha) - c(\alpha)$$
$$E(\alpha) = R(\alpha)^T B(\alpha)^T$$
$$g(\alpha) = R(\alpha)^T f(\alpha)$$

$$K^+(\alpha) = diag\left(K_1^+(\alpha), \ldots, K_N^+(\alpha)\right)$$

$$\lambda = \begin{bmatrix} \lambda_B \\ \lambda_E \\ \lambda_I \end{bmatrix},\ B = \begin{bmatrix} B_B \\ B_E \\ B_I \end{bmatrix},\ c = \begin{bmatrix} o \\ o \\ c_I \end{bmatrix}$$

$$\text{span}\{R_{*,i}(\alpha)\} = \text{null } K(\alpha)$$

$R(\alpha)$ is a-priori known!

Reconstruction formula

$$u(\alpha) = K^+(\alpha)\left(f(\alpha) - B(\alpha)^T \lambda(\alpha)\right) + R(\alpha)\xi$$

with appropriate vector $\xi \in {}^{6N}$

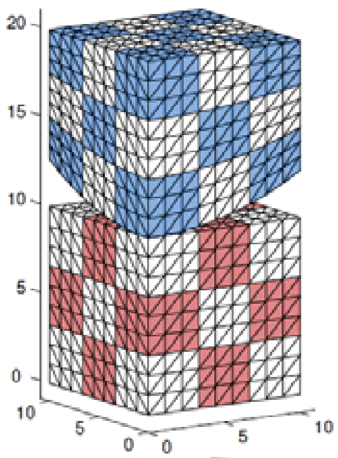
Changing the shape of the bodies, they have to be remeshed in each design optimization step!

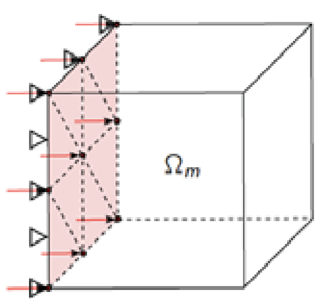
☹

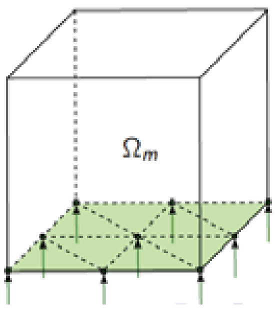
Department
of Applied Mathematics

December 2, 2010

Total BETI solution of state problem







$$\min J(u) \qquad \text{subject to}$$
$$B_I u \quad c_I, B_E u = 0, B_B u = 0$$

$$J(u) = \tfrac{1}{2} u^T S(\alpha) u - u^T r(\alpha)$$
$$S(\alpha) = diag\left(S_1(\alpha), \dots, S_N(\alpha)\right)$$

$$S_i(\alpha) = D_i + \left(\tfrac{1}{2} M_i + K_i\right)^T V_i^{-1} \left(\tfrac{1}{2} M_i + K_i\right)$$
$$r_i(\alpha) = M_i^T V_i^{-1} N_i^0$$

Total BETI solution of state problem

$$\min \tfrac{1}{2}\lambda^T F(\alpha)\lambda - \lambda^T d(\alpha) \text{ subject to } \lambda_I \geq o,\ E(\alpha)\lambda = g(\alpha)$$

$$F(\alpha) = B(\alpha)S^+(\alpha)B(\alpha)^T$$
$$d(\alpha) = B(\alpha)S^+(\alpha)r(\alpha) - c(\alpha)$$
$$E(\alpha) = R(\alpha)^T B(\alpha)^T$$
$$g(\alpha) = R(\alpha)^T r(\alpha)$$

$$S^+(\alpha) = diag\left(S_1^+(\alpha),\ldots,S_N^+(\alpha)\right)$$

$$\lambda = \begin{bmatrix} \lambda_B \\ \lambda_E \\ \lambda_I \end{bmatrix},\ B = \begin{bmatrix} B_B \\ B_E \\ B_I \end{bmatrix},\ c = \begin{bmatrix} o \\ o \\ c_I \end{bmatrix}$$

$$\text{span}\{R_{*,i}(\alpha)\} = \text{null } S(\alpha)$$

$R(\alpha)$ is a-priori known!

Reconstruction formula

$$u(\alpha) = S^+(\alpha)\left(r(\alpha) - B(\alpha)^T\lambda(\alpha)\right) + R(\alpha)\xi$$

with appropriate vector $\xi \in {}^{6N}$

Department
of Applied Mathematics
December 2, 2010

Finite difference sensitivity analysis

$$\frac{\partial u(\alpha)}{\partial \alpha_i} \approx \frac{u(\alpha + he_i) - u(\alpha)}{h}$$

where

$u(\alpha + he_i)$ solves

$$\min \tfrac{1}{2}u^T K(\alpha + he_i)u - u^T f(\alpha + he_i)$$

subject to $B(\alpha + he_i)u \leq c(\alpha + he_i)$

and $e_i = (0,\ldots,0,\underset{i}{1},0,\ldots,0),\ i = 1,\ldots,m$

- *Advantage*
  - Simple implementation
- *Disadvantages*
  - m+1 assemblies of stiffness matrix
  - m+1 solution of contact problem (m+1 decompositions of K)
  - m+1 constructions of R
  - numerically unstable
- *Does "semi-analytical" method exist?*

Semi-analytical sensitivity analysis

$I_C = \{i : B_{i,*}(\alpha)u(\alpha) = c_i(\alpha)\}$ ... indices of nodal variables in contact

$I_S = \{i : i \in I_C \wedge \lambda_i(\alpha) > 0\}$ ... indices of nodal variables in strong contact

$I_W = \{i : i \in I_C \wedge \lambda_i(\alpha) = 0\}$ ... indices of nodal variables in weak contact

$$u'(\alpha,\beta) = \lim_{h\to 0} \tfrac{1}{h}\big(u(\alpha + h\beta) - u(\alpha)\big)$$

solves

$$\min \tfrac{1}{2} z^T K(\alpha) z - z^T \bar{f}(\alpha,\beta)$$

$$\text{s.t. } B_W(\alpha) z \le c_W(\alpha,\beta),\ B_S(\alpha) z = c_S(\alpha,\beta)$$

$$\bar{f}(\alpha,\beta) = f'(\alpha,\beta) - K'(\alpha,\beta)u(\alpha) + B'^T(\alpha,\beta)\lambda(\alpha)$$

$$B_S(\alpha) = [B_i(\alpha)]_{i\in I_S},\ c_S(\alpha,\beta) = \left[f_i'(\alpha,\beta) - B_{i,*}'(\alpha,\beta)u(\alpha)\right]_{i\in I_S}$$

$$B_W(\alpha) = [B_i(\alpha)]_{i\in I_W},\ c_W(\alpha,\beta) = \left[f_i'(\alpha,\beta) - B_{i,*}'(\alpha,\beta)u(\alpha)\right]_{i\in I_W}$$

Only one assembly and one decomposition of stiffness matrix for all $\beta = e_i$, $i=1,\ldots,m$

Department
of Applied Mathematics

December 2, 2010

BETI & FETI based sensitivity analysis

$$\min \tfrac{1}{2}\mu^T \bar{F}(\alpha)\mu - \mu^T \bar{d}(\alpha,\beta) \quad \text{s. t.} \quad \mu_w \ge o,\ \bar{E}(\alpha)\mu = \bar{g}(\alpha,\beta)$$

where

$$\bar{F}(\alpha) = \bar{B}(\alpha)K^+(\alpha)\bar{B}^T(\alpha),\ \bar{d}(\alpha,\beta) = \bar{B}(\alpha)K^+(\alpha)\bar{f}(\alpha,\beta) - \bar{c}(\alpha,\beta),$$

$$\bar{E}(\alpha) = R^T(\alpha)\bar{B}^T(\alpha),\ \bar{g}(\alpha,\beta) = R^T(\alpha)\bar{f}(\alpha,\beta)$$

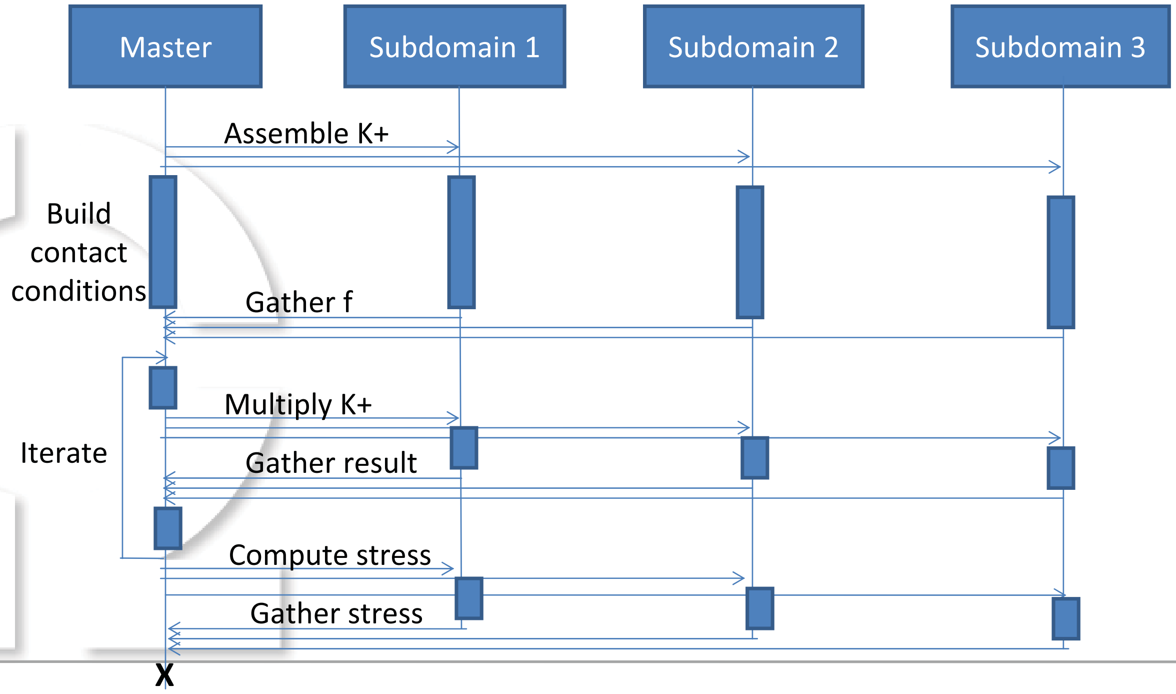
$$\bar{B}(\alpha) = \begin{bmatrix} B_W(\alpha) \\ B_S(\alpha) \end{bmatrix},\quad \bar{c}(\alpha,\beta) = \begin{bmatrix} c_W(\alpha,\beta) \\ c_S(\alpha,\beta) \end{bmatrix},\quad \mu = \begin{bmatrix} \mu_W \\ \mu_S \end{bmatrix},\quad z(\alpha,\beta) = u'(\alpha,\beta)$$

$$u'(\alpha,\beta) = K^+(\alpha)\big(\bar{f}(\alpha,\beta) - \bar{B}(\alpha)^T \mu(\alpha,\beta)\big) + R(\alpha)\zeta$$

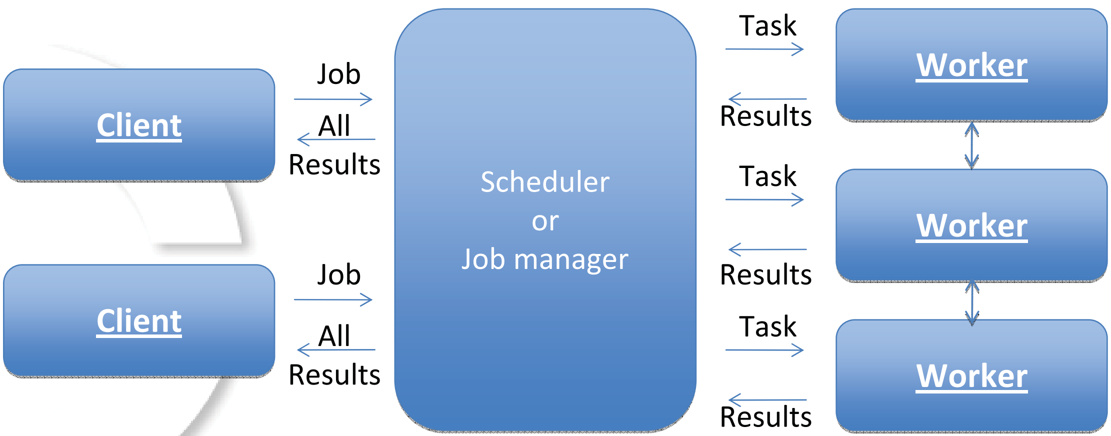
with such appropriate ζ

K^+ and R is known from state problem analysis, I_w is empty in reality!

Scenario of parallel solution



MATLAB Parallel Jobs



Only one task is solved in parallel within one client’s job.
Parallel tasks can communicate each other.

Total BETI vs. Total FETI-sequential code

	State problem			Sensitivity analysis		
Prim/Dual	Pre-Post	Sol	Total	Pre-Post	Sol	Total
10368/6480	48s 16s	10s 8s	58s 24s	768s 256s	160s 128s	928s 384s
48000/20256	470s 44s	30s 30s	500s 74s	7520s 704s	480s 480s	8000s 1184s
279936/70848	7680s 350s	500s 430s	8180s 780s	122880s 5600s	8000s 6880s	130880s 12480s

10368/6480	3x	1.25x	2.4x	3x	1.25x	2.4x
48000/20256	10.7x	1x	6.8x	10.7x	1x	6.8x
279936/70848	22x	1.2x	10.5x	22x	1.2x	10.5x

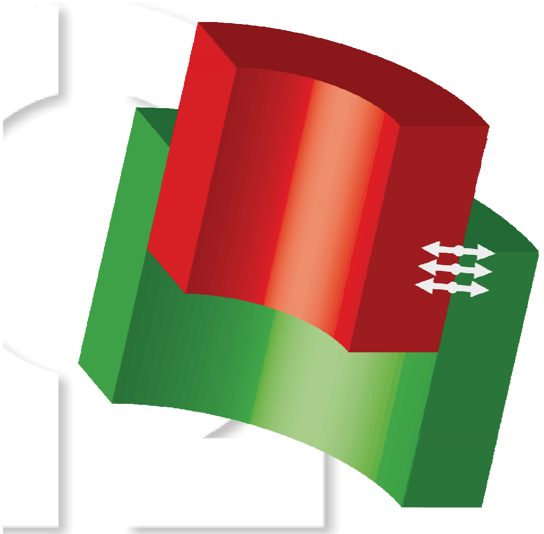
Total BETI vs. Total FETI-parallel code

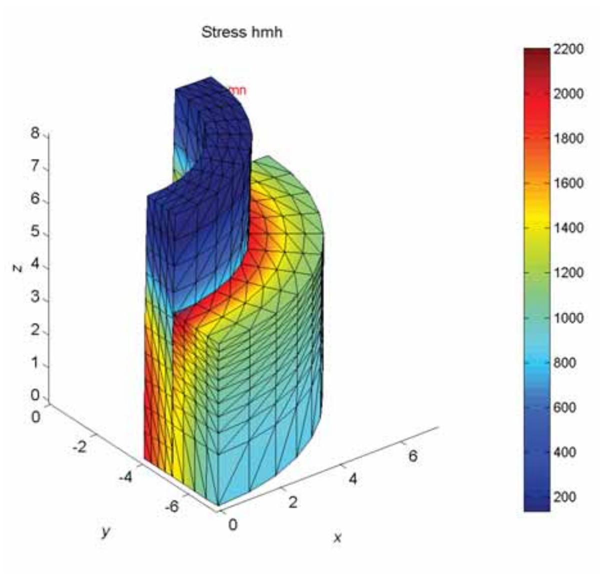
	State problem			Sensitivity analysis		
Prim/Dual	Pre-Post	Sol	Total	Pre-Post	Sol	Total
10368/3108	34s 24s	2s 2s	36s 26s	243s 151s	13s 14s	256s 165s
48000/9021	152s 29s	6s 8s	158s 37s	1370s 258s	50s 66s	1420s 324s
279936/30036	3500s 143s	49s 150s	3549s 393s	31500s 1959s	441s 972s	31941s 2931s

10368/3108	1.4x	1x	1.4x	1.6x	0.9x	1.6x
48000/9021	5.2x	0.75x	4.3x	5.3x	0.75x	4.4x
279936/70848	24x	0.33x	9x	16x	0.45x	11x

Two Cylinders

- min $\frac{1}{2}u^TKu-u^Tf$
- 3 design variables box constrained

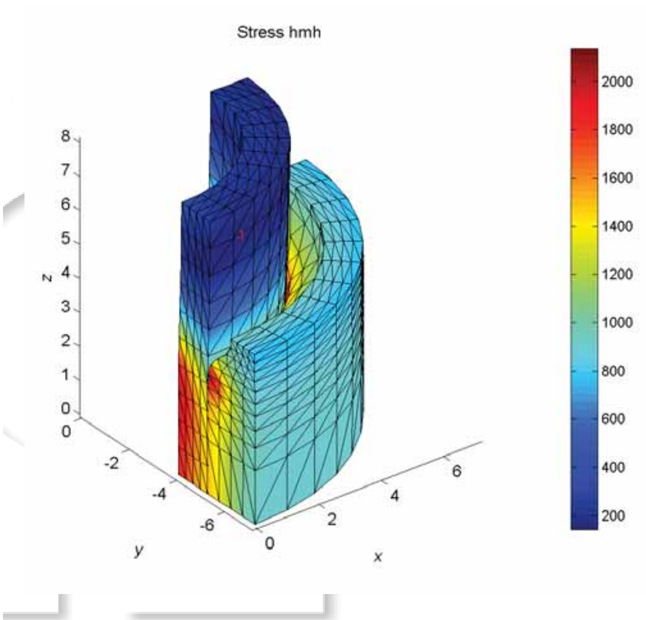


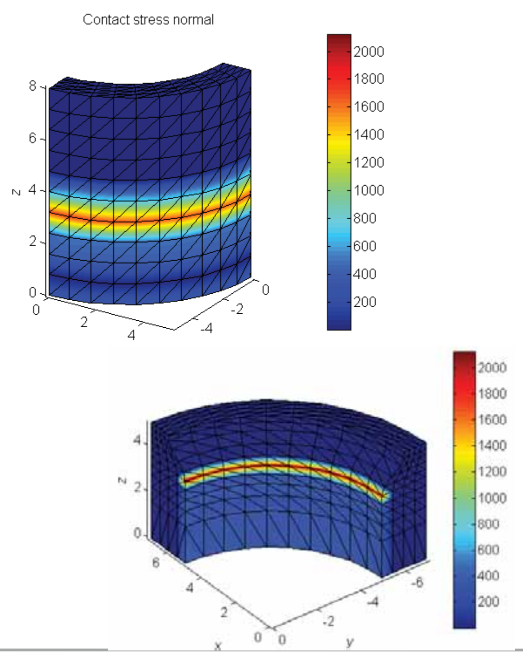


Department
of Applied Mathematics

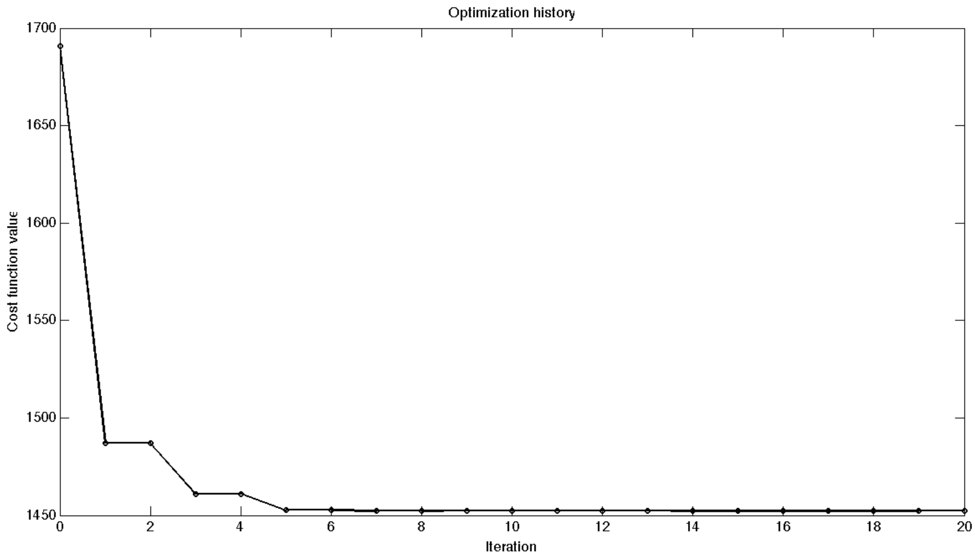
December 2, 2010

Two Cylinders - optimization





Two Cylinders - history



Conclusions and future works

- *Efficient solution on parallel computers*
  - MATLAB distributed computing server
  - Scalable Total FETI algorithm
  - Efficient sensitivity analysis
- *BETI vs. FETI*
  - BETI do not need remeshing of interior of bodies
  - Matrices resulting form BETI are full
  - Assembling of BETI operator is very time consuming
- *Future works*
  - Fast sparse aproximation for BETI
  - Massive parallel solution – both sensitivity and domain decomposition parallelism

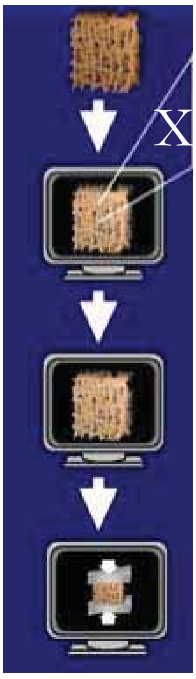
Thank you

HTTP://WWW.AM.VSB.CZ/MATSOL

Computational (geo) micromechanics

R. Blaheta, P. Byczanski,
R. Kohut, V. Sokol et al.
Computational Mechanics II, Ostrava

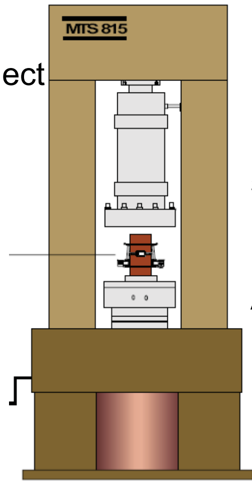
Context



- Project GACR: Multiscale modelling and X-Ray tomography in geotechnics
- Experience
  - X-ray CT
  - Laboratory preparation and testing of geocomposites
  - FE modelling
- Aims – interconnection of the knowledge, better understanding of behaviour of geocomposites, development of homogenization techniques, FE software and solvers

U Kumamoto
CTU Prague
new ICT project

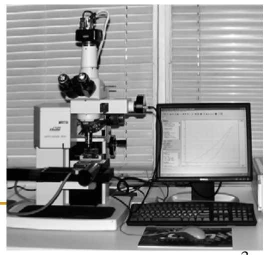
Zwick 1494
(600 kN)
new ICT project



- load
- ultra-sound
- AE
- perme Ability
- …

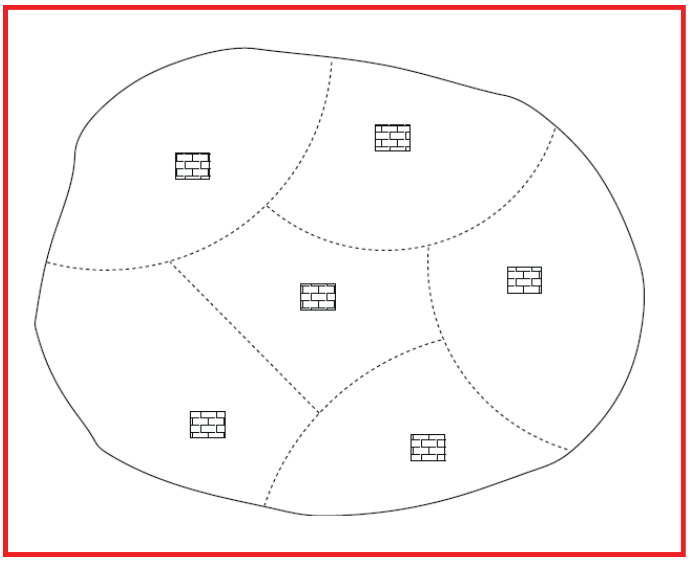
IGN - Hubert
32-core SMP system
128GB RAM
new IT4 project

Micro Hardness Tester
CSM Instruments

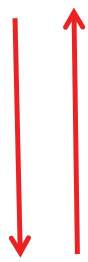


Multiscale Heterogeneity

- Large difference in characteristic dimensions
- Impossible to cover by unique model
- Requirement for special approaches



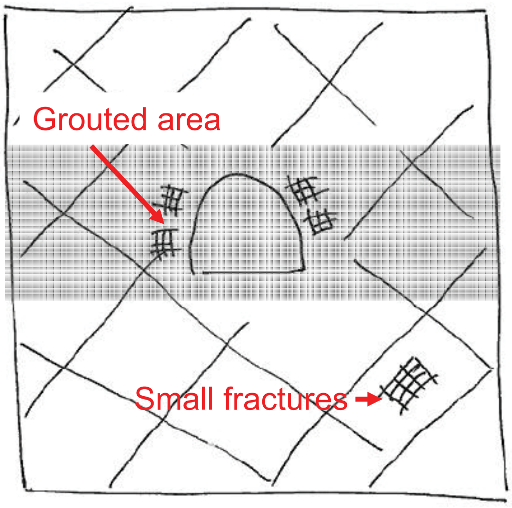
MACROSCALE



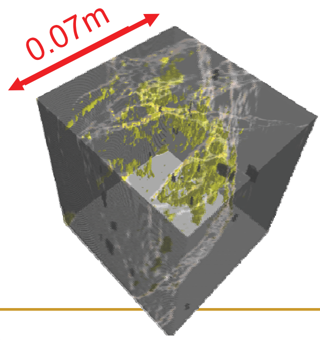
up-
scaling

MICROSCALE

Grouting and geocomposites

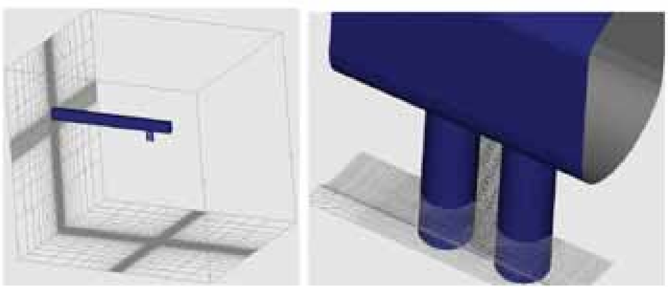


- Macroscale 50-100m
- Global model = rock blocks and main fractures
- Microscale 0.001m – small fractures, grouted fractures



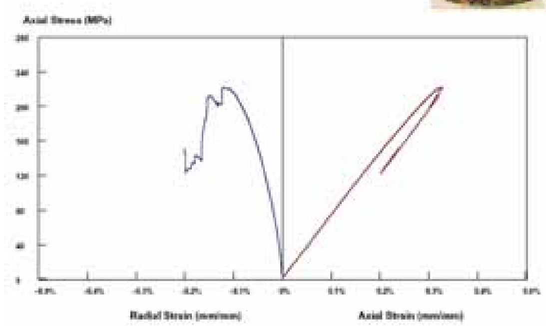
Microscale properties influence the macroscale behaviour

Decovalex -APSE damage of granite rocks

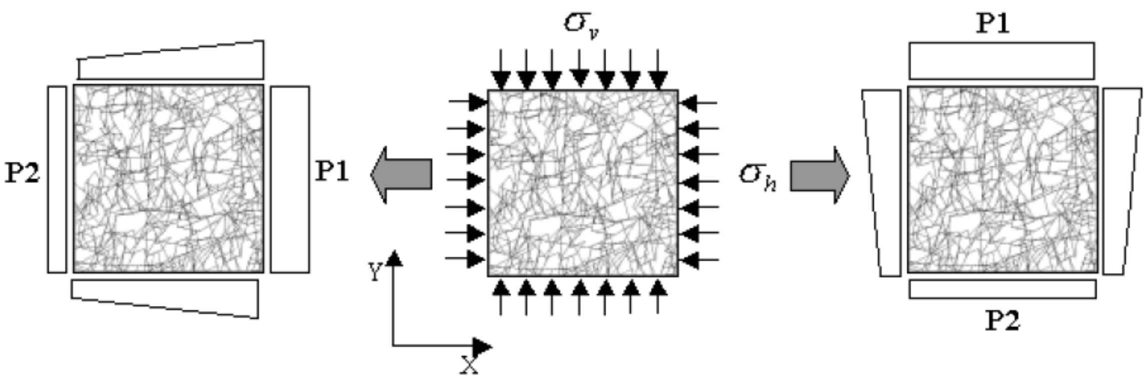


Damage of granite is also included by macrostress due to continuum damage model

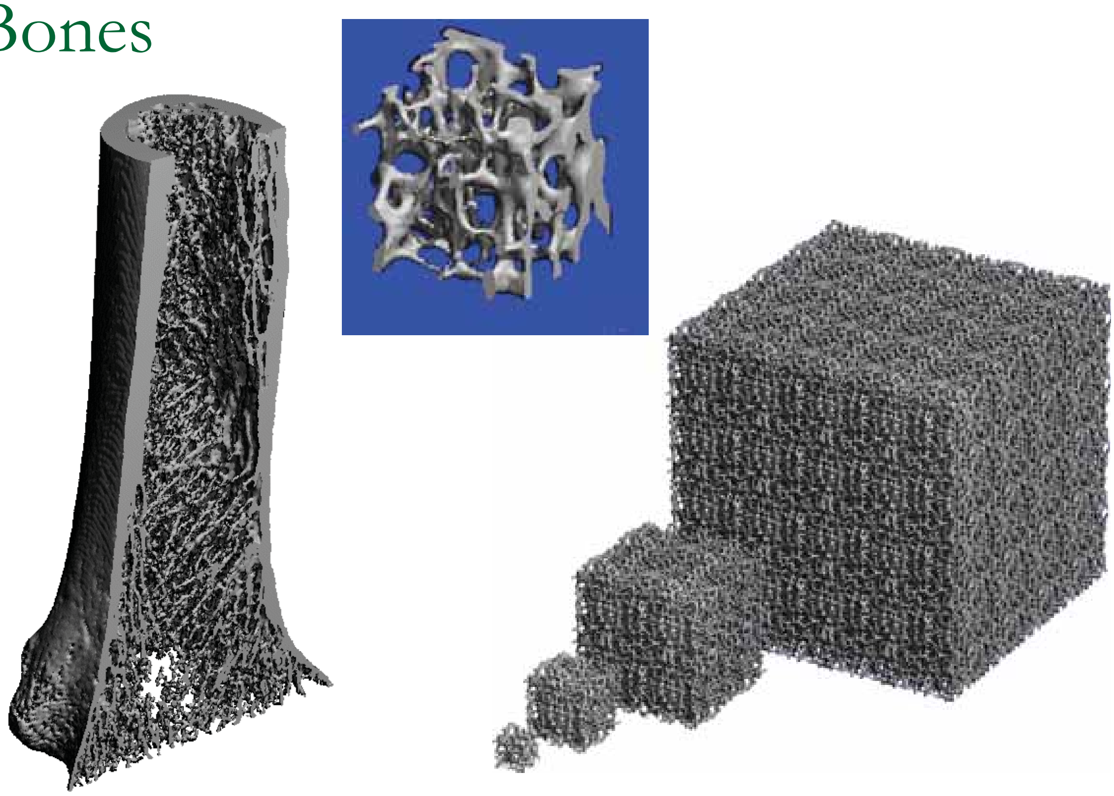
- Macro stress influences damage in microscale
- Damage influences flow in macroscale



Decovalex – flow in fractured rocks



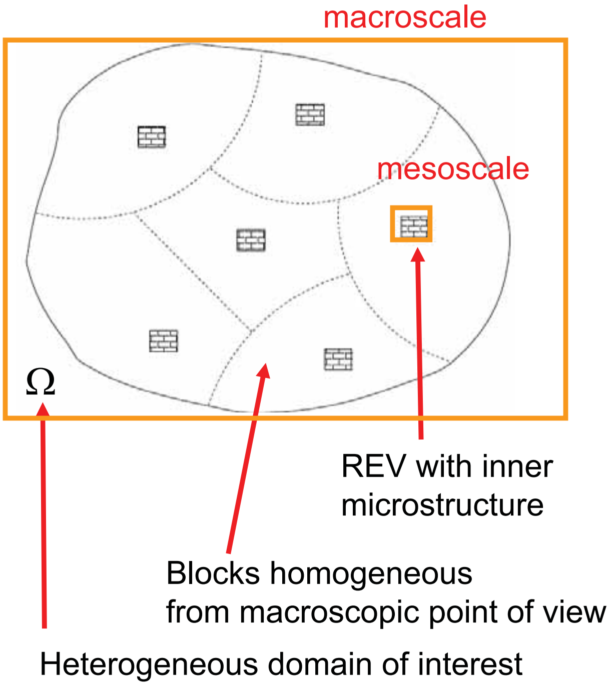
Bones



Multiscale - scale separation

- Macro – meso - micro scale
- REV representative volume element
- Upscaling and homogenization
- Special case: periodic and layered structures

Heterogeneity & multiscale



$$\varepsilon(x) = \overline{\varepsilon}(x) + \widetilde{\varepsilon}(x)$$

$$\sigma(x) = \overline{\sigma}(x) + \widetilde{\sigma}(x)$$

Microscale variables =
macroscopic fields
+ local microscale fluctuations

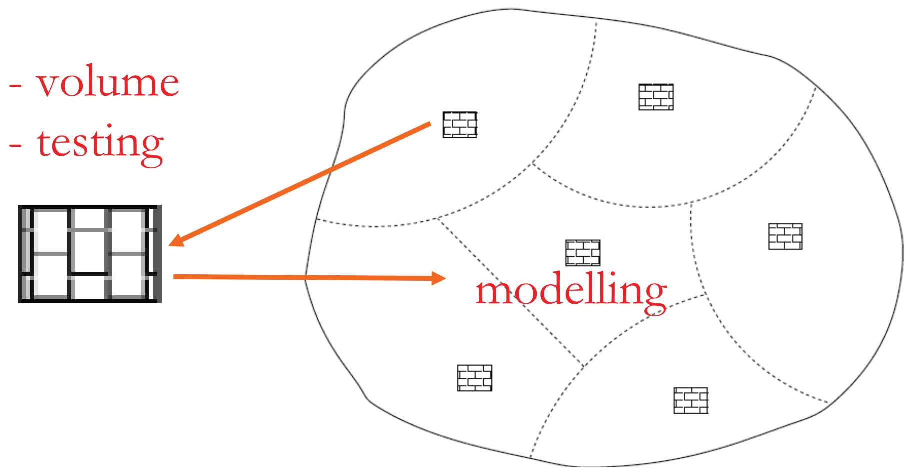
$$\overline{\varepsilon} = \frac{1}{V}\int_V \varepsilon(x)\,dV =$$

$$= \frac{1}{2V}\int_S (u \otimes n + n \otimes u)\,dS$$

$$\overline{\sigma} = \frac{1}{V}\int_V \sigma(x)\,dV =$$

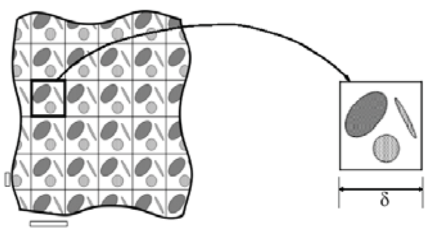
$$= \frac{1}{V}\int_S (t(x) \otimes x)\,dS$$

Upscaling



RVE – representative volume element V

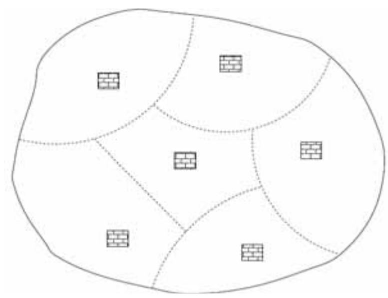
- Definition I: RVE is a cell that sufficiently accurate represents the overall macroscale properties of interest (increasing the size does not change calculated material parameters (Size effect)
- Definition II: statistically homogeneous



Ω_0 a unit periodic cell for periodic microstructure

Upscaling scheme

Ω



scale separation

macroscale

$$L_h u_h = L_h(\bar{k}) u_h = f_h \quad in \;\; \Omega + BC$$

$$L_h u_h = -div\,\sigma,\; \sigma = \bar{k}\varepsilon,\; \varepsilon = \varepsilon(u_h)$$

mesoscale - microscale

$$\tilde{L}_h u_h = \tilde{L}_h(k) u_h = 0 \quad in \;\; V + BC$$

$$\tilde{L}_h \tilde{u}_h = -div\,\sigma,\; \sigma = k\varepsilon,\; \varepsilon = \varepsilon(\tilde{u}_h)$$

$$\bar{\sigma} = \langle \sigma \rangle = |V|^{-1} \int_V \sigma(x)\,dx$$

$$\bar{k}: \quad \bar{\sigma} = \bar{k}\bar{\varepsilon} \quad \text{or} \quad \overline{\sigma \cdot \varepsilon} = \bar{k}\bar{\varepsilon} \cdot \bar{\varepsilon}$$

Homogenization $\tilde{L}_h u_h = \tilde{L}_h(k) u_h = 0 \;\; in \;\; V + \boxed{BC}$

- ε_0-KUBC $\qquad u(x) = \varepsilon_0 \cdot x \quad \text{on } \partial\Omega$
- σ_0-SUBC $\qquad t = \sigma \cdot n = \sigma_0 \cdot n \quad \text{on } \partial\Omega$
- periodic $\qquad u(x) = \varepsilon_0 \cdot x + v(x) \text{ on } \partial\Omega,\;\; v(x) \;\text{ periodic}$
- mixed $\qquad (u(x) - \varepsilon_0 \cdot x)(\sigma(x) \cdot n(x) - \sigma_0 \cdot n(x)) = 0 \;\; \text{on } \partial\Omega$

$$\langle \varepsilon \rangle = \varepsilon_0, \quad \langle \sigma \rangle = \sigma_0$$

$$\langle \sigma \rangle = C_\varepsilon^{app} \varepsilon_0 \quad \Rightarrow \quad C_\varepsilon^{app}$$

$$\langle \varepsilon \rangle = S_\sigma^{app} \sigma_0 \quad \Rightarrow \quad C_\sigma^{app} = \left(S_\sigma^{app}\right)^{-1}$$

Influence of BC and test volume size

- The homogenized coefficients are unique only for well separated scales and REV size of Test Volume
- Otherwise there is a dependence on BC
- And there is a size effect regarding size of SD – partition of sample and averaging of the test results for subsamples gives
- Equivalence of mechanical and energetic definition as well as size criterion – Hill condition

$$C_\sigma^{app} \le C^{app} \le C_\varepsilon^{app}$$

cf. Reuss lower – Voight upper bounds

$$\left\langle S_{\sigma,k}^{app} \right\rangle^{-1} \le C_\sigma^{app} \le C_\varepsilon^{app} \le \left\langle C_{\varepsilon,k}^{app} \right\rangle$$

difference and SD size

$$\langle \varepsilon : \sigma \rangle = \langle \varepsilon \rangle : \langle \sigma \rangle$$

1D homogenization

$1D$ *periodic microstructure problem*

$$-[k(x)u'(x)]' = 0 \ in \ \Omega$$

Reference cell $(0,1)$

$$k(x) = \begin{cases} k_1 \ in \ (0,\frac{1}{2}) \\ k_2 \ in \ (\frac{1}{2},1) \end{cases}$$

$$BC: \ \sigma \cdot n = \sigma_0 \cdot n \ on \ \partial\Omega$$

$$\bar{\sigma} = \sigma_0 = \sigma$$

$$\bar{\varepsilon} = \frac{1}{2}\frac{\sigma_0}{k_1} + \frac{1}{2}\frac{\sigma_0}{k_2}$$

$$\frac{\bar{\varepsilon}}{\bar{\sigma}} = \frac{1}{\bar{k}} = \frac{1}{2}\left(\frac{1}{k_1} + \frac{1}{k_2}\right)$$

$$BC: \ u = \varepsilon_0 \cdot x + p \ on \ \partial\Omega$$

$$\bar{\varepsilon} = \varepsilon_0, \ \ piecewise \ constant \ \varepsilon_a, \varepsilon_b$$

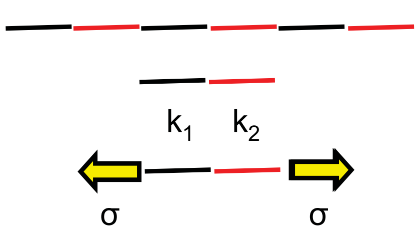
$$\bar{\varepsilon} = \tfrac{1}{2}\varepsilon_a + \tfrac{1}{2}\varepsilon_b$$

$$\bar{\sigma} = \sigma \ \ constant = k_1\varepsilon_a = k_2\varepsilon_b$$

$$\Rightarrow (1 + \tfrac{k_1}{k_2})\varepsilon_a = 2\bar{\varepsilon}$$

$$\Rightarrow \bar{\sigma} = 2k_1(1 + \tfrac{k_1}{k_2})^{-1}\bar{\varepsilon}$$

Homogenization – 1D periodic medium



1D peiodic medium
Basic cell, RVE
Material coefficients
Application of load, flux => constant flux

$$\bar{\varepsilon} = \frac{1}{|V|}\int_V \varepsilon \, dV = \frac{1}{2}\left(\frac{\sigma}{k_1} + \frac{\sigma}{k_2}\right) = \frac{\bar{\sigma}}{\bar{k}}$$

$$\bar{k} = \frac{2k_1k_2}{k_1 + k_2}$$

Application of different BC:
here preserve constant flux
=> give the same result k¯

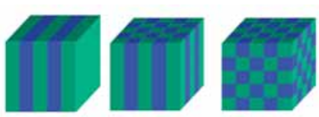
NOTES
- The same result for any test volume being a multiple of basic cell (the same proportion of material components)
- Big difference from arithmetic average

Asymptotic theory for periodic media

$$\frac{\partial}{\partial x}\left(A\left(\frac{x}{\epsilon}\right)\frac{\partial u}{\partial x}\right) = 0$$

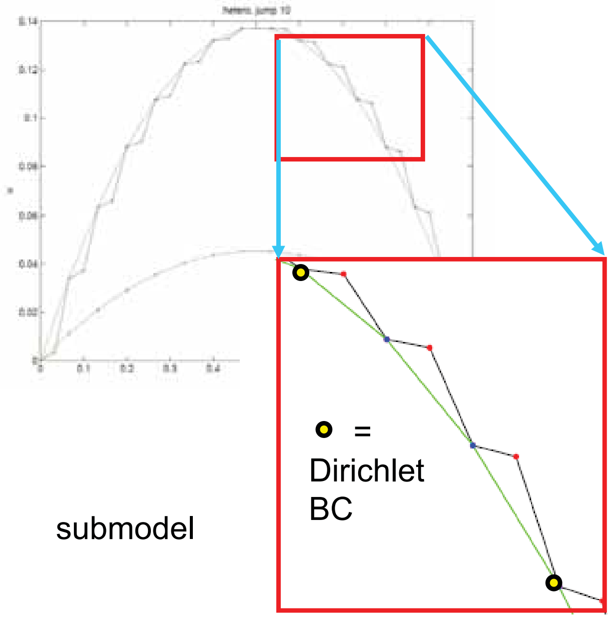
$$u_\varepsilon \to u^*$$

$$\frac{\partial}{\partial x}\left(A^*\frac{\partial u}{\partial x}\right) = 0$$



- PDE with rapidly oscillating periodic coefficients, ε=l/L, A_ε=A(x/ε)
- u_ε → u* – boundedness in H^1, weak convergence
- u* is solution of a boundary value problem with constant coefficients A*
- A* is constant, known as effective property
- local correctors

Local correctors

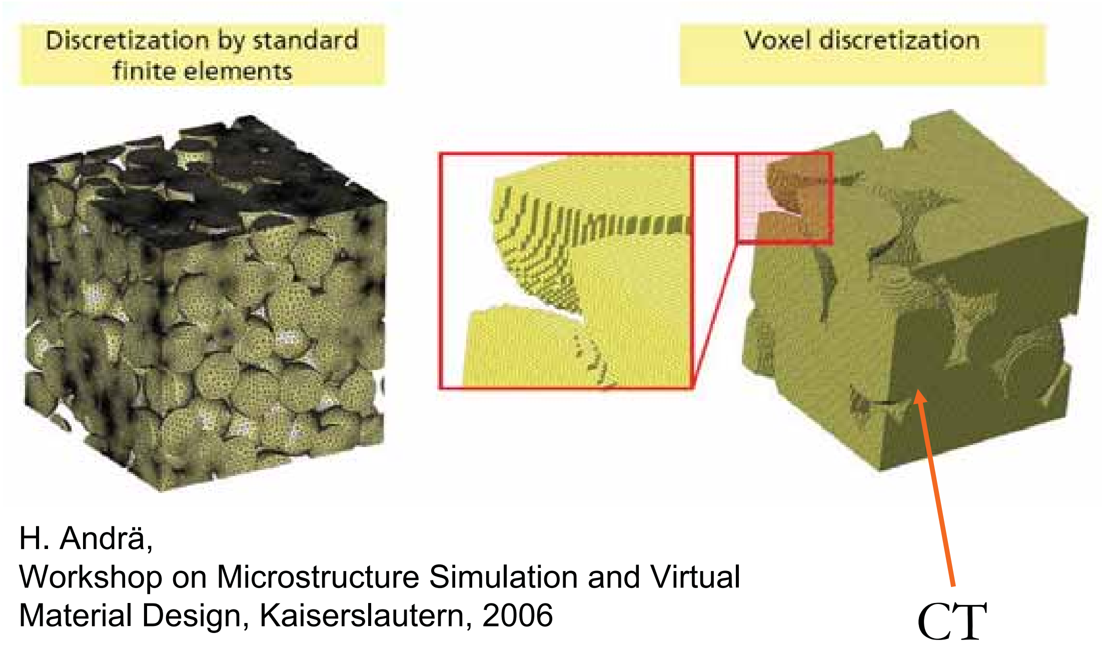


- Local – global approaches

FEM analysis

- Standard FEM on macroscale
- Standard FEM on mesoscale
  - BC, pure Neumann
  - Aligned and nonaligned (voxel) grids
- Accuracy and heterogeneity
- Heterogeneity and efficient solvers

FEM – voxel/aligned grids



H. Andrä,
Workshop on Microstructure Simulation and Virtual Material Design, Kaiserslautern, 2006

Solvers

- Large scale problems
- Jumps in coefficients

Parallel solver – P. Arbenz, ETH

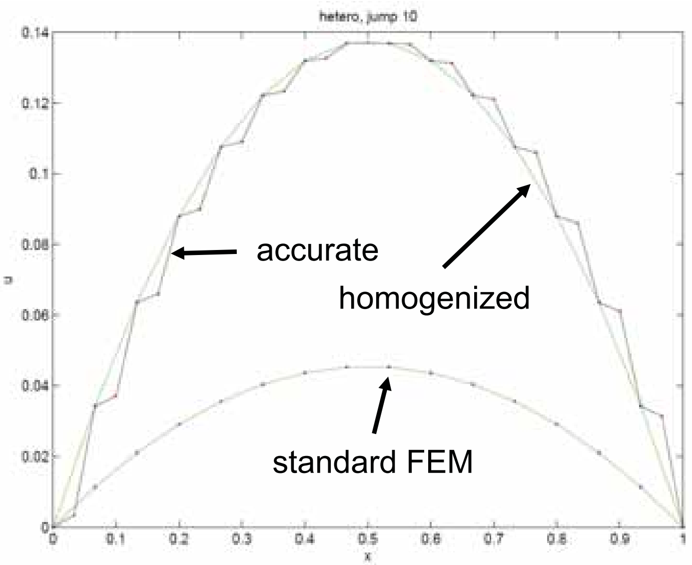
- PCG with smoothed aggregation (SA) multilevel preconditioning

$$AB^{-1}\mathbf{y} = \mathbf{b}, \qquad \mathbf{y} = B\mathbf{x}.$$

Solving with B means applying one multigrid V-cycle.

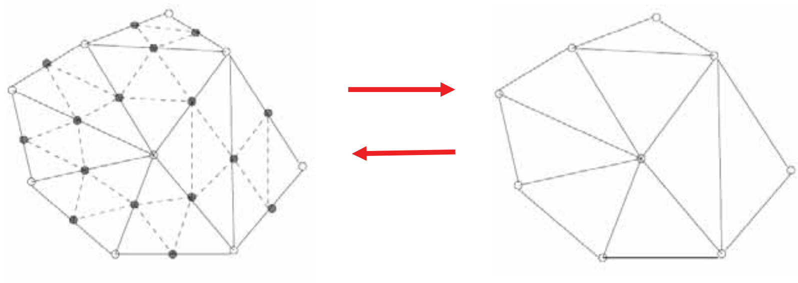
• P. Arbenz, G.H. van Lenthe, U. Mennel, R. Müller, and M. Sala: "A Scalable Multi-level Preconditioner for Matrix-Free µ-Finite Element Analysis of Human Bone Structures". Internat. J. Numer. Methods Engrg. 73 (7): 927-947 (2008), doi:10.1002/nme.2101.

FEM on macro and mesoscale



- Figure: -(ku')'=1 in (0, 1) k є {1;10}, periodic ε=1/30 hom. Dirichlet BC
  - accurate = P1 solution, h=1/30
  - standard FEM = P1 solution, h=1/15
  - homogenized = P1 solution, h=1/15, homogen. coeff. (constant)

Efficient multilevel solvers



- Multigrid, AMG, AMLI, DD methods
- Fine and (auxiliary) coarse grids/problems
- Intergrid transfers
- Coarse grid elements homo/heterogeneous
- Analogy with multiscale

Multigrid – auxiliary macroscale

Two scales

- Two Grid method
- initial $u_h, r_h = b - Au_h$
- while $\|r_h\| > \varepsilon$
  - Smoothing $u_h \leftarrow u_h + \omega D^{-1}(b - Au_h)$
  - Coarse grid correction $u_h \leftarrow u_h + \boxed{R^T A_H^{-1} R}(b - Au_h)$
  - Smoothing $u_h \leftarrow u_h + \omega D^{-1}(b - Au_h)$
- end

Multigrid = system with A_H is solved iteratively by TG using still coarser level(s)

Efficient multigrid

-(ku')'=1 in (0, 1), k є {1;10}, periodic ε=1/30
hom. Dirichlet BC

$$A_H = \text{coarse grid, homogenized coeff.}$$

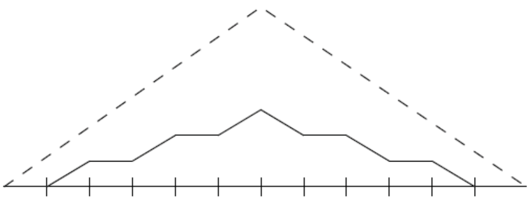
$$R = \begin{pmatrix} x & & & & & & \\ & 1/2 & 1 & 1/2 & & & \\ & & & 1/2 & 1 & 1/2 & \\ & & & & & & \ddots \end{pmatrix} + \begin{pmatrix} 0 & & & & & & \\ & c & 0 & -c & & & \\ & & & c & 0 & -c & \\ & & & & & & \ddots \end{pmatrix}$$

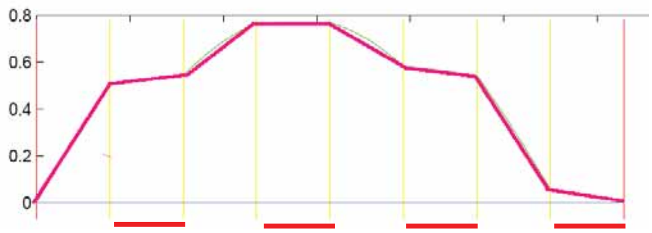
$$c = H(x_{i+1/2}) = \int_{x_i}^{x_{i+1/2}} \left(\frac{\bar{k}}{k(x)} - 1 \right) dx \qquad A_H = R A_h R^T$$

Aggregations

$$R = \begin{pmatrix} x & & & & & \\ & 1 & 1 & & & \\ & & & 1 & 1 & \\ & & & & & \ddots \end{pmatrix}$$

Aggregations – jumping coefficients





- we shall investigate aggregations within the two-level Schwarz framework
- show that if elements in aggregations are at least as stiff like elements in the surrounding THEN the two-level Schwarz is robust w.r.t. coefficient jumps

Stiff aggregations

-div(k grad(u))=f

$$V_h = \text{span}\,\{\phi_1, \dots, \phi_n\}\,, \text{ FE space and nodal basis functions}$$

$$V_0 = \text{span}\{\psi_1, \dots, \psi_N\}, \text{ aggregations } N < n$$

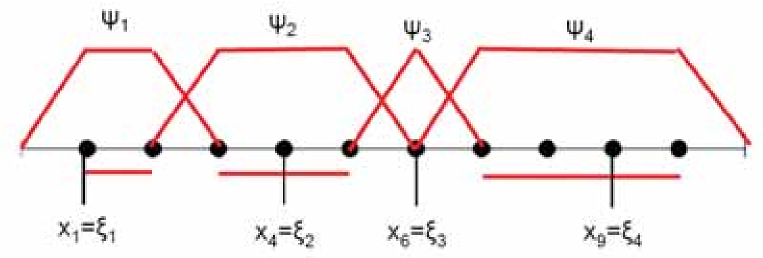
$$\psi_i = \sum \varphi_{ij}\phi_j\,,\ \varphi_{ij} \in \{0,1\}\,,\ \sum_j \varphi_{ij} \le m_a \quad \forall_j \exists! \, i : \varphi_{ij} = 1$$

$$\mathcal{T}_h = \mathcal{T}_h^a \cup \mathcal{T}_h^b \qquad \begin{array}{l} T \in \mathcal{T}_h^a \Leftrightarrow \text{ex. } i : T \subset \text{interior } (\text{supp}\{\psi_i\}) \\ T \in \mathcal{T}_h^b \Leftrightarrow T \in \mathcal{T}_h \setminus \mathcal{T}_h^a\,. \end{array}$$

$$T \in \mathcal{T}_h^b \Rightarrow S_T = \{S \in \mathcal{T}_h^a, \exists i : T, S \subset \overline{\text{supp}\{\psi_i\}}\}$$

$$\text{STIFF AGGREGATIONS } \forall T \in \mathcal{T}_h^b \ \forall S \in S_T \quad k_T \le k_S$$

Stiff aggregations

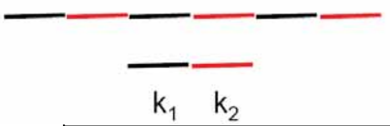


$$(i)\ \ Qv = v_0\,,\ \ v_0 = \sum_1^N \alpha_i \psi_i\,,\ \ \alpha_i = v(\xi_i)$$

$$(ii)\ \ \ldots,\ \alpha_i = \mid \mathrm{supp}\{\psi_i\} \mid^{-1} \int_{\mathrm{supp}\{\psi_i\}} v(x)dx$$

- Stability constant K_0 does not depends on h, k_max / k_min
- Constant K_1 depends on the domain decomposition (K_1=3 for layered decomposition)
- Efficient and robust two-level method

Multilevel methods – model 1D problem



k_1/k_2 =	1/1e0	1/1e1	1/1e2	1/1e3	1/1e4
standard TG method	7	46	451	-	-
TGH=TG+homogenized coarse problem	7	27	-	-	-
TGHC=TGH+corrected transfer	7	7	7	7	7
AG=TG+aggregation transfer	23	10	8	8	8

Numbers of iterations - solving the system from FE discretization of a 1D model problem with periodic microstructure, period size 1/15, discretization h=1/30 (microstructure is fully resolved on fine grid), coarse system from discretization with H=1/15 and the variants exploiting homogenization ideas.

Schwarz – exact solvers

Stability estimate : $v = \sum_k v_k \,, \quad v_k = \Pi_h(\Theta_k v)\,.$

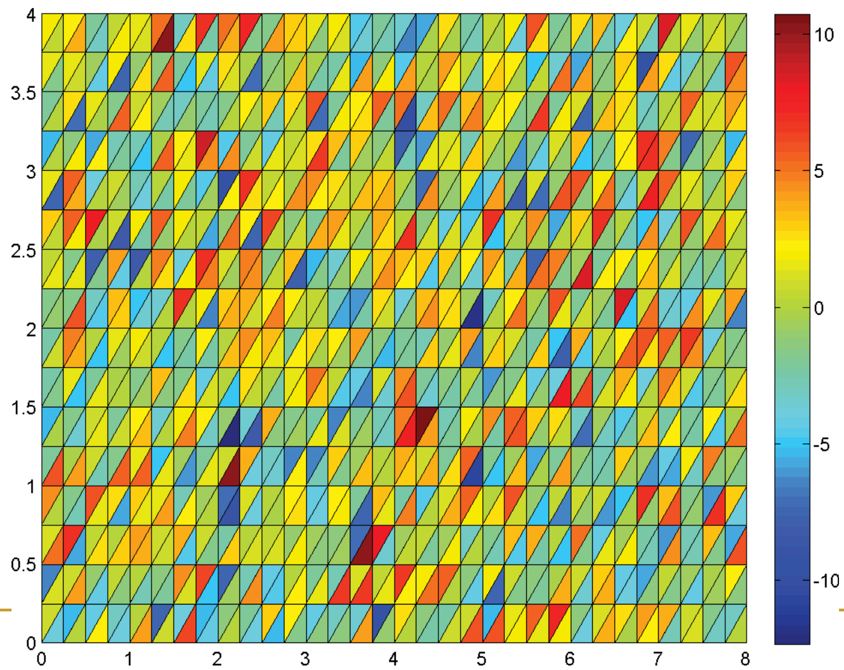
$$\sum \| v_k \|_A^2 = \sum \int_{\Omega_k} k \left\| \nabla \left(\Pi_h(\theta_k v) \right) \right\|^2 dx$$

$$\leq c \sum \int_T k \|\theta_k\|_\infty^2 \|\nabla v\|^2 dx + c \sum \int_T \underline{k \|\nabla \theta_k\|_\infty^2} \|v\|^2 dx$$

$$\leq K_0 \int_\Omega k \|\nabla v\|^2 dx$$

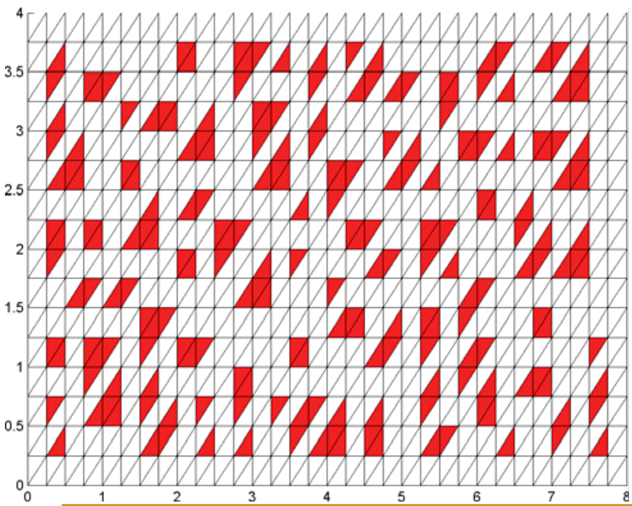
- Heterogeneity outside overlap
- Inside overlap
- Two-level methods

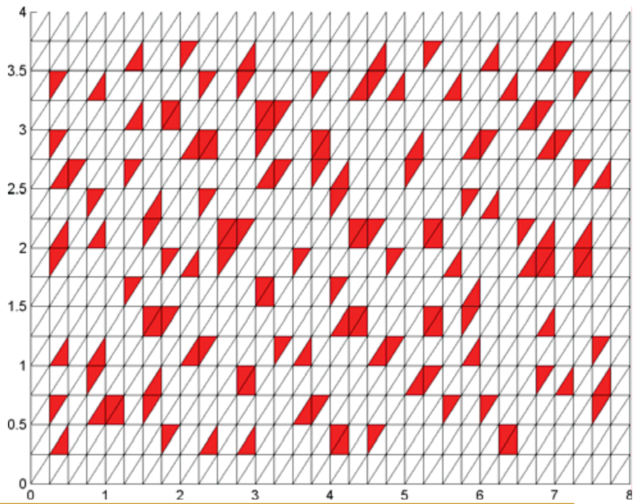
Heterogeneity



Aggregations

- thresholding to p=0.3
- thresholding to p=0.7





Numerical results

- number of iterations (32385 DOFs)

$\frac{k_{max}}{k_{min}}$	CG	$p = 0.1$	$p = 0.3$	$p = 0.5$	$p = 0.7$	$p = 0.9$
10^0	365	14	14	14	14	14
10^2	695	11	11	10	9	7
10^4	1538	11	10	9	8	6
10^6	4046	11	10	9	8	6
10^8	10273	13	11	9	8	6
10^{10}	25735	14	12	10	8	6

(b) $H/h = 8,\ H = 1/16$

- size coarse spaces

h	$\frac{k_{max}}{k_{min}} = 1$	$\frac{k_{max}}{k_{min}} > 1$				
		$p = 0.1$	$p = 0.3$	$p = 0.5$	$p = 0.7$	$p = 0.9$
$1/32$	719	855	927	1059	1231	1666
$1/64$	2463	2988	3377	3871	4613	6301
$1/128$	9023	11361	12716	14795	18098	24757

Homogenized parameters of geocomposite

BC	3aBC		1aBC	
loading	**E(MPa)**	**ν**	**E(MPa)**	**ν**
direction x	2367.82	0.2861	2318.21	0.3060
direction y	1947.20	0.3054	2018.21	0.2676
direction z	2369.74	0.2860	2319.60	0.3043

Reuss bound : E_R = 1837.71 MPa, ν_R = **0.2565**
Voight bound: E_V = 2387.57 MPa, ν_V = **0.3141**

Reuss and Voight bounds for E and **ν** are determined from Reuss and Voight bounds for the effective material tensor C_{eff}

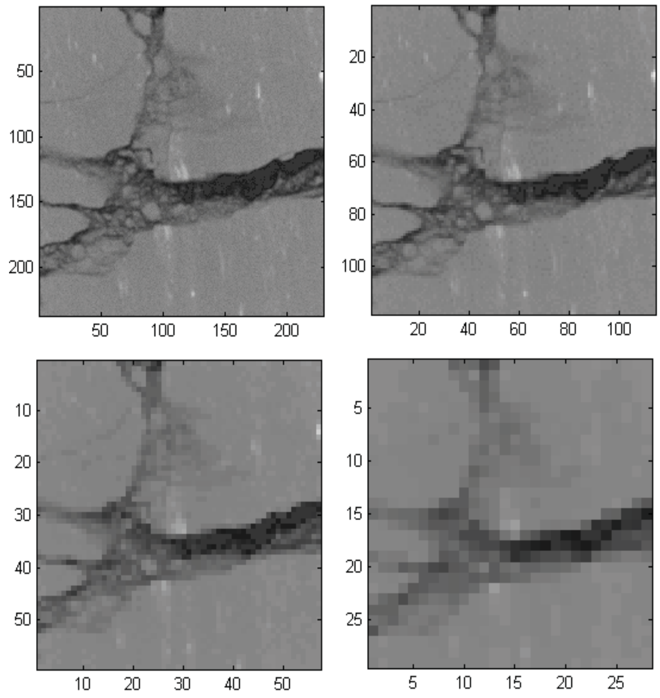
Geocomposite seems to be softer in direction y

Sensitivity of macro response to changes in local material properties (+10 %)

	3aBC(z)		1aBC(z)	
	ΔE_{hom} [%]	**$\Delta\nu_{hom}$ [%]**	**ΔE_{hom} [%]**	**$\Delta\nu_{hom}$[%]**
E_1^+	+0.01	+0.07	+0.04	+0.00
ν_1^+	-0.00	+0.00	-0.00	+0.00
E_2^+	+0.14	+0.28	+0.22	+0.00
ν_2^+	-0.01	+0.07	-0.01	+0.23
E_3^+	+0.99	+0.17	+1.04	+0.03
ν_3^+	+0.01	+0.80	-0.03	+0.82
E_4^+	**+8.81**	-0.59	**+8.60**	+0.10
ν_4^+	+0.77	**+9.06**	+0.08	**+9.00**
E,ν initial	2369.74	0.2860	2319.60	0.3043

Sensitivity to changes in coal parameters

Sensitivity to selected voxel grid



- grid 1: 230 × 238 × 37 = 2 025 380 nodes (6 076 140 degrees of freedom),
- grid 2: 115 × 119 × 37 = 506 345 nodes (1 519 035 degrees of freedom),
- grid 3: 58 × 60 × 37 = 128 760 nodes (386 280 degrees of freedom),
- grid 4; 29 × 30 × 37 = 32 190 nodes (96 570 degrees of freedom).

Sensitivity of macro response to voxel grid density

Material	E_i [MPa]	ν_i [--]	grid 1	grid2	grid 3	grid 4
0 void	0.01	0.001	0.735	0.705	0.600	0.387
1 PUR 1	200	0.100	1.386	1.108	0.855	0.667
2 PUR 2	500	0.175	5.543	5.466	4.870	3.640
3 PUR 3	2100	0.250	9.670	10.226	11.545	13.379
4 coal	2600	0.320	82.667	82.495	82.131	81.927

The elasticity parameters of the individual materials and the volume fractions of materials

Sensitivity of macro response to voxel grid density

	E [MPa]	Change in [%] of the homogenized E		
type of BC	**grid 1**	**grid 2**	**grid 4**	**grid 4**
3aBC(x)	2368	1	2	3
3aBC(y)	1947	2	5	12
3aBC(z)	2370	0	1	2
1aBC(x)	2318	1	2	4
1aBC(y)	2018 ?	1	4	10
1aBC(z)	2319 ?	0	1	3
averaged values	2223	1	2	6
Voight bound	2383	0	1	2

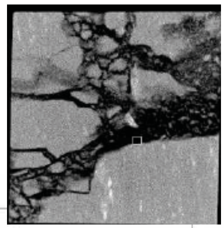
The sensitivity of the homogenized elasticity modulus to the grid density

I. The computation of elastic behaviour of geocomposites is
   - Possible
   - Stable
   - Verified with laboratory experiments

II. Strength and nonlinear behaviour

III. Some challenges

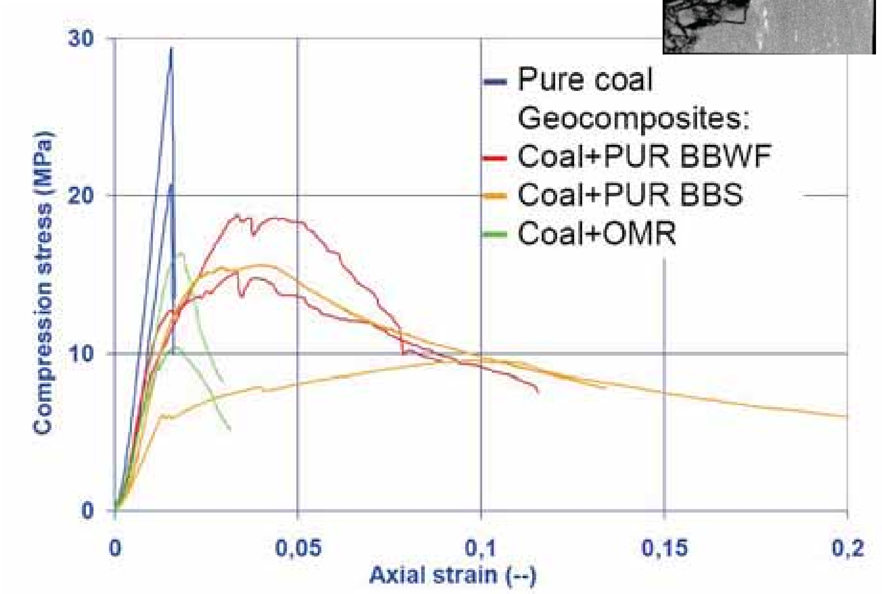
Nonlinear geocomposite



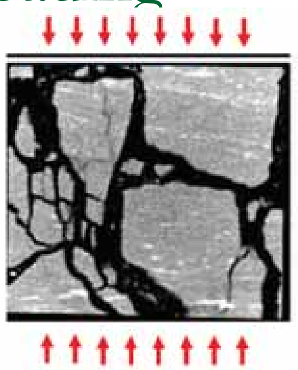
Coal – PE composite

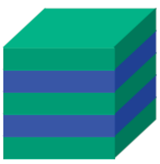
brittle

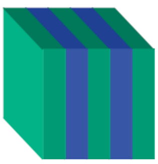
quasi-brittle/plastic behaviour



Inelastic behaviour in uniaxial/triaxial loading



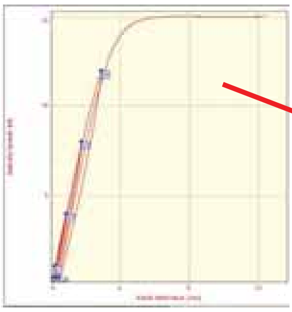


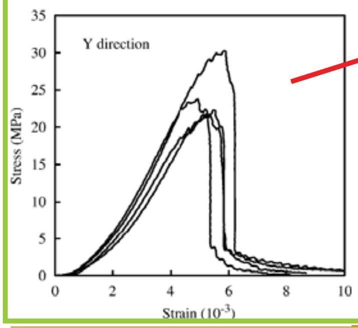


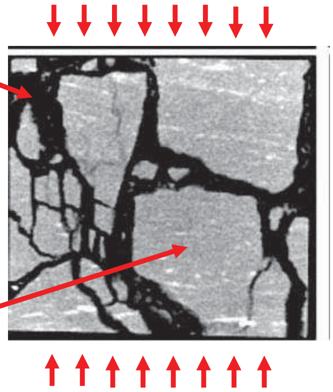
- Finding the ultimate load
  - Anisotropic strength
  - Depends heavily on local stress singularities
    - Nonlocal approach
  - Need for knowledge of local material properties
  - continuum/discrete models
  - Inverse analysis
- Finding the full strain-stress curve

Homogenization of inelastic features

INPUT:

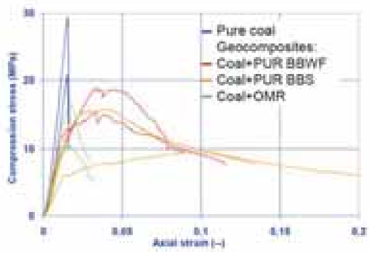




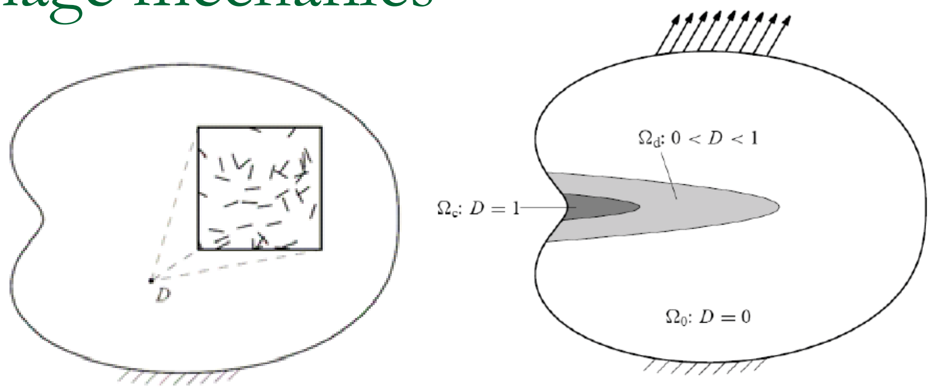


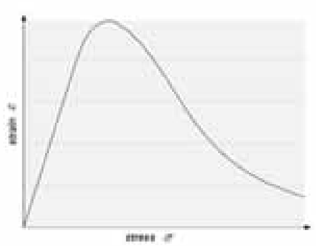
Uniaxial compression test on a digital specimen

OUTPUT:
Complete stress-strain curve for characterization of REV behaviour and modelling on macroscale



Damage mechanics





$$(a)\quad \varepsilon,\ \kappa,\ \sigma$$

$$(b)\quad f(\varepsilon,\kappa)=\bar{\varepsilon}-\kappa\le 0$$

$$(c)\quad \dot{\kappa}\ge 0,\quad \dot{\kappa}\, f(\varepsilon,\kappa)=0$$

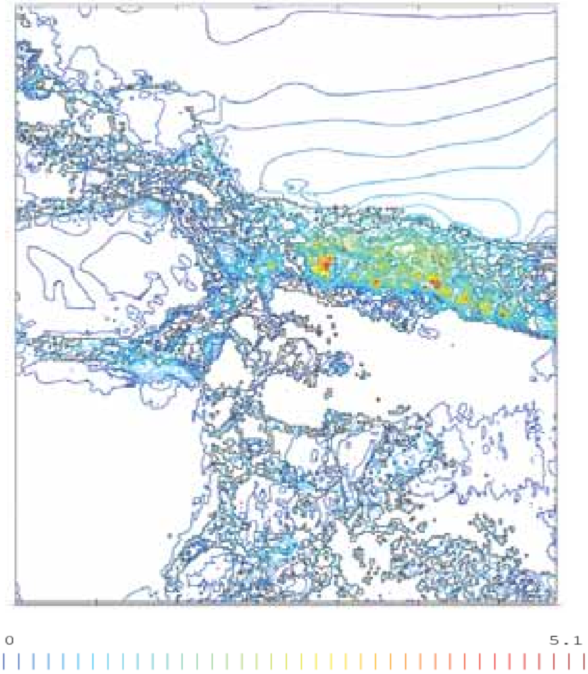
$$(d)\quad \omega=g(\kappa),\ g'\ge 0$$

$$(e)\quad \sigma=(1-\omega)E\varepsilon$$

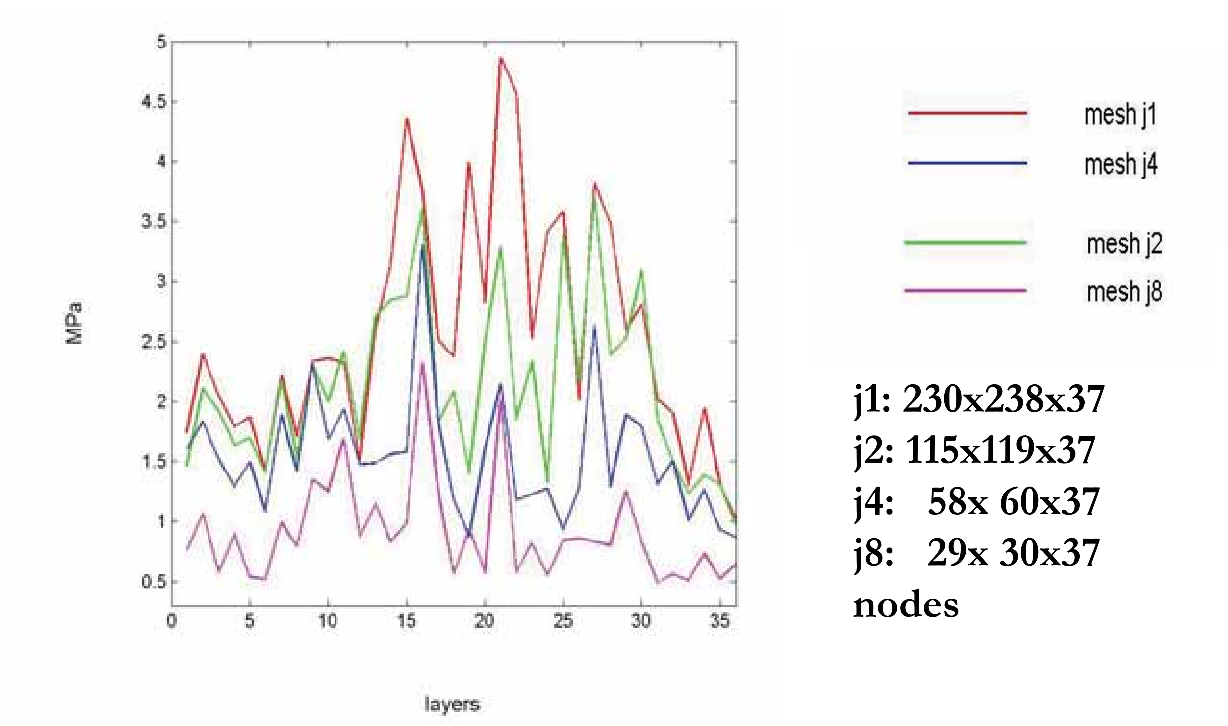
κ is internal damage parameter, f damage function, (b) admissi bility condition, (c) complementarity cond., (d) damage law, (e) secant type Hooke law

Local failure

Stress analysis in in coal geocomposite sample under compressive loading

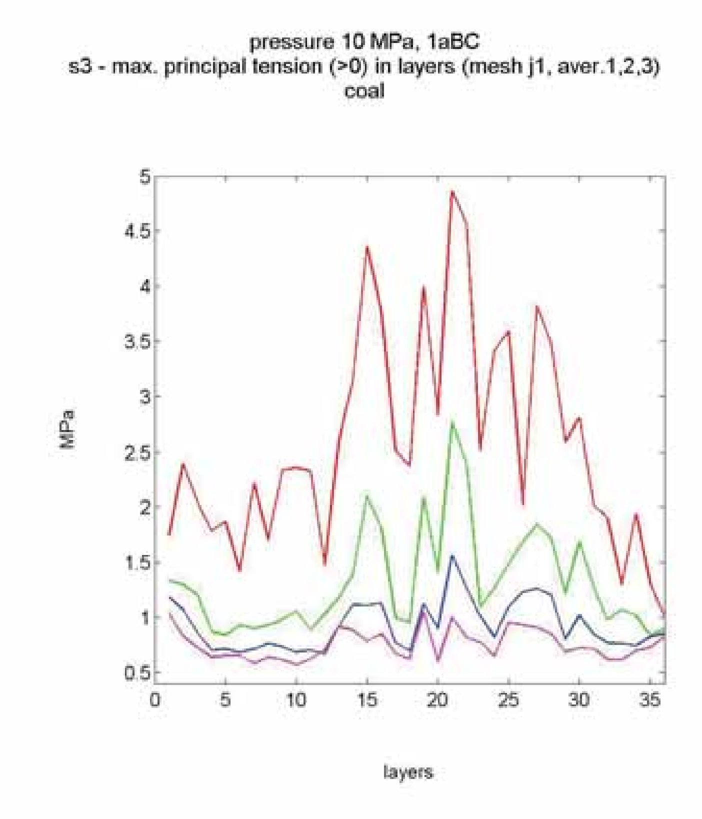


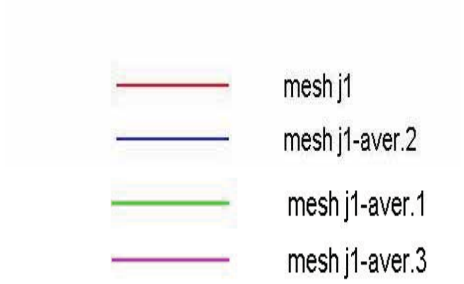
Max tension in layers – loading 10 MPa



j1: 230x238x37
j2: 115x119x37
j4: 58x 60x37
j8: 29x 30x37
nodes

Nonlocal stress





aver1 – 1 layer neighborhood (27 hexahedra)
aver2 – 2 layer neighborhood (125 hexahedra)
aver3 – 3 layer neighborhood (343 hexahedra)

$$\Lambda_M(\sigma(y), y) = \sqrt{[(\sigma_1(y) - \sigma_2(y))^2 + (\sigma_2(y) - \sigma_3(y))^2 + (\sigma_3(y) - \sigma_1(y))^2]/(2\sigma_c^2(y))}$$

$$\underline{\Lambda}(\sigma, y) = \Lambda\left(\tilde{\sigma}(y), y\right)$$

$$\tilde{\sigma}_{ij}(y) = \int_\Omega \varphi(x,y)\sigma_{ij}(x)dx$$

$$\varphi(x,y) = \begin{cases} \frac{3}{4\pi d^3} & |x-y| < d \\ 0 & |x-y| \geq d \end{cases}$$

local strength condition for micro-stresses

$$\sup_{y\in\Omega} \Lambda\left(\sigma(y), y\right) < 1.$$

(point) non-local strength condition

$$\Lambda^{\odot}(\sigma; y) := \Lambda\left(\sigma^{\odot}(y), y\right)$$

$$\Lambda^{\odot}(\sigma; y) < 1$$

non-local strength condition for the whole body

$$\sup_{y\in\Omega} \Lambda\left(\sigma^{\odot}, y\right) < 1$$

S.E. Mikhailov

- local strength condition for micro-stresses

$$\sup_{y\in\Omega} \Lambda\left(\sigma(y), y\right) < 1.$$

- (point) non-local strength condition

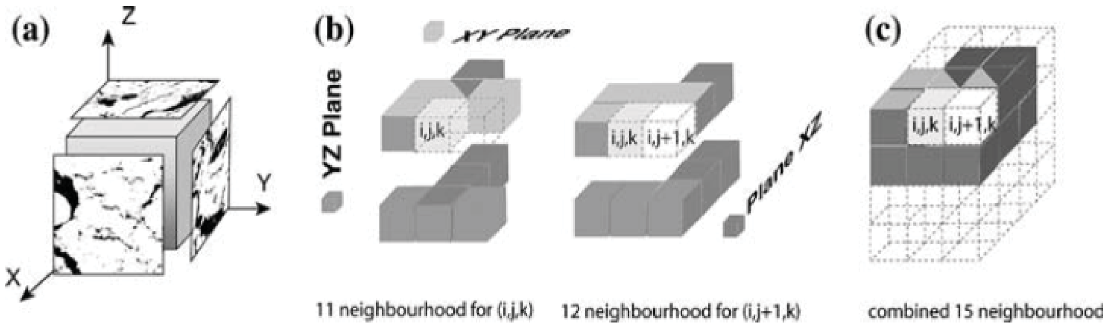
$$\Lambda^{\odot}(\sigma; y) < 1$$

$$\Lambda^{\odot}(\sigma; y) := \underline{\Lambda}(\sigma, y) = \Lambda\left(\tilde{\sigma}(y), y\right), \quad \tilde{\sigma}_{ij}(y) = \int_{\Omega} \varphi_{ijkl}(x, y)\sigma_{kl}(x)dx$$

- non-local strength condition for the whole body

$$\sup_{y\in\Omega} \Lambda\left(\sigma^{\odot}, y\right) < 1$$

Stochastic: Paper 3D Stochastic Modelling of Heterogeneous Porous Media 2006



- Determination of probabilities of neighbourhoodness of given components
- Frontal type stochastic generation
- Monte Carlo, Markov chain
- Checking global properties – mercury porosimetry
- Data for Liberec-type granite

Thank you for your attention !

Identification of material parameters / Calibration of models

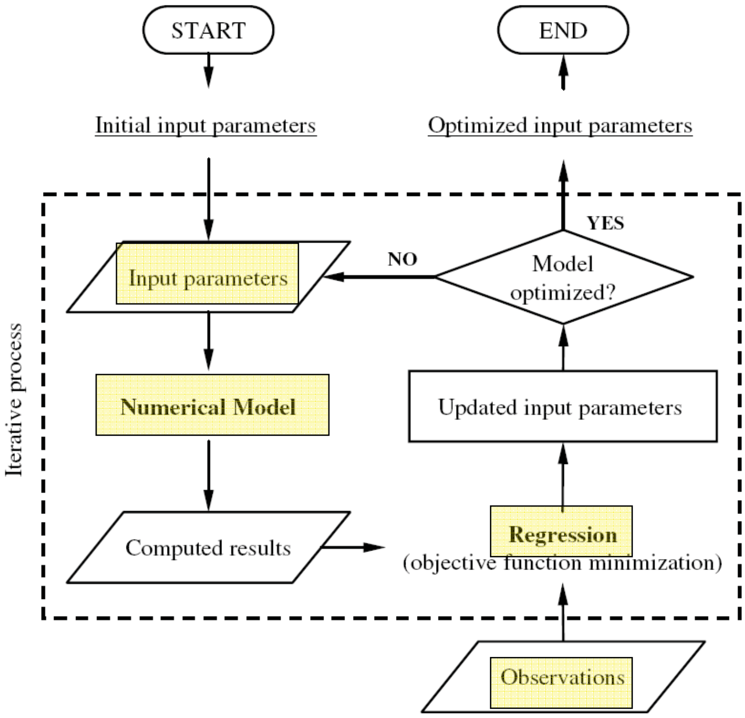
R. Blaheta, R. Hrtus, R. Kohut,
O. Jakl, J. Starý et al.
Computational Mechanics II, Ostrava

Outline

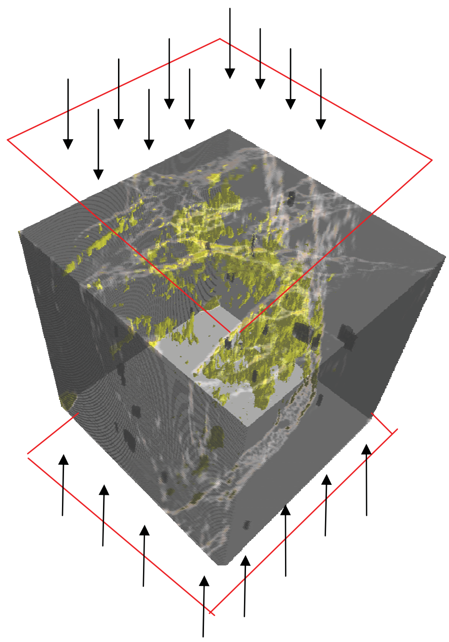
- Inverse problems
  - Identification of material parameters or calibration of the model - tasks requiring optimization of the difference between computed and measured data
  - Inverse problem with apriori given (heterogeneous) distribution of material but unknown material properties
  - Solution scheme and parallelization
- Geo-application
- Discrete model - Least-squares formulation
- NM, GA and gradient methods
- Numerical experiments with NM and GA
- Conclusions

Parameter identification

- use for identification of problem parameters
- use for calibration of the model
- numerical model = FEM solution of time depend./indep. heat conduction
- regression = least square minimization



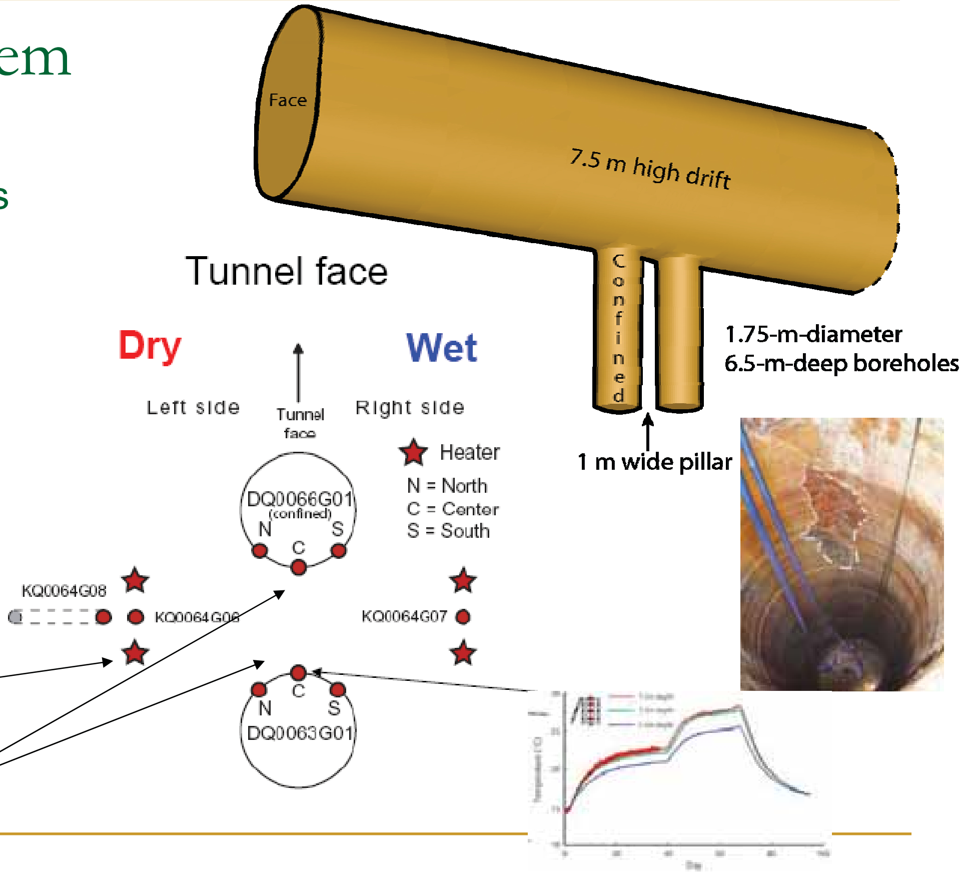
Identification of parameters



- Geocomposites
- CT scan of sample
- Materials – CT values
- Identification of local material properties
- Identification with use of several loading cases

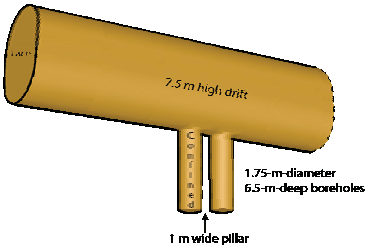
APSE problem

I. Initial stress
II. Elastic stress due to construction phases
III. Heating phase
IV. Damage, spalling, EDZ evolution
…



Heaters
Sensors in diff. depths

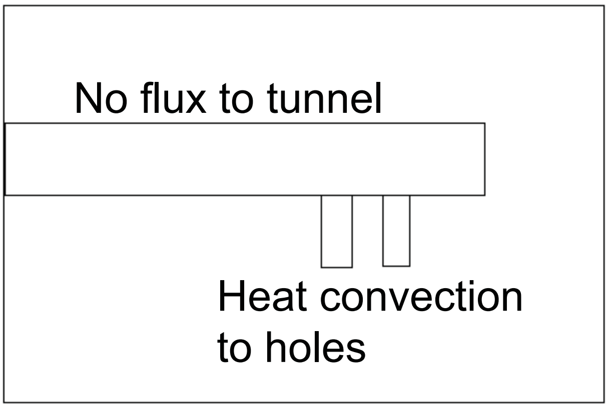
Calibration



- Due to an underground water flow in a vicinity of the pillar it is difficult to use simple heat conduction model with heat capacity and conductivity from lab tests
- Therefore, we consider heat capacity and conductivity and heat convection parameters as unknown and try to fit the measured temperatures
- Change of heat conductivity a capacity induce change in heat flow in the model.
- Correctness of the calibrated model. The change of parameters provides model, which meets the observation data and seems to provides reasonable results (to justify physical correctness is generally difficult)

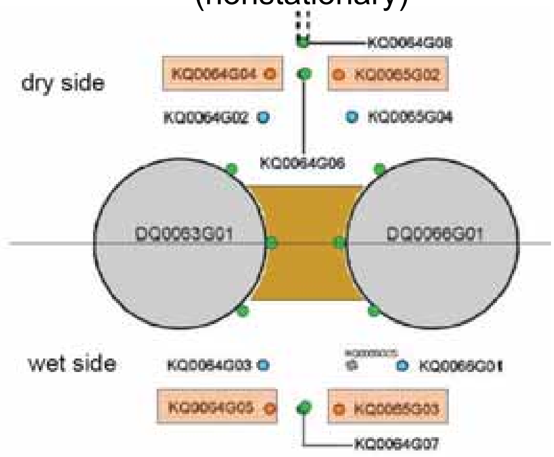
The calibrated model

Boundary conditions:



Temperature 14.5 °C at
the whole outer boundary

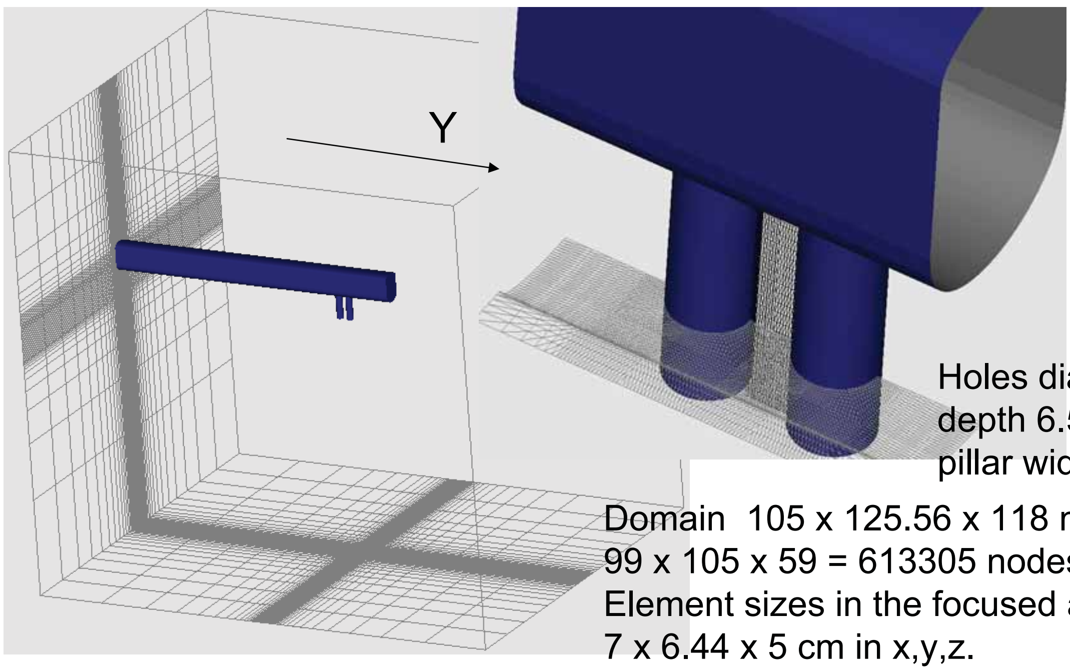
Linear heaters
(nonstationary)



Linear heaters

Parameters: k1,k2,k3
c1,c2,c3
h1,h2,h3

Discretization/computation with GEM sw



Holes diam. 1.75m,
depth 6.5m
pillar width 1.03 m

Domain 105 x 125.56 x 118 m
99 x 105 x 59 = 613305 nodes
Element sizes in the focused area
7 x 6.44 x 5 cm in x,y,z.
Time stepping: 580 time steps

Calibration via parameter identification

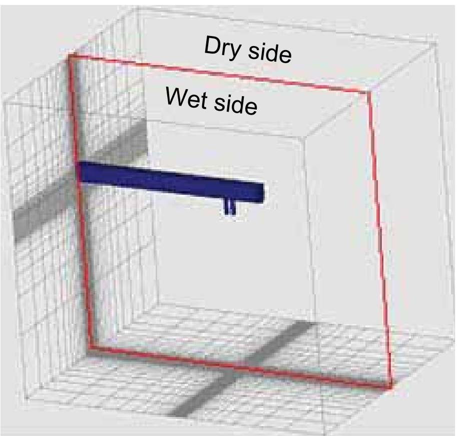
$$FEM : (k_1, k_2, c_1, c_2, \ldots) \rightarrow u_{FEM}$$

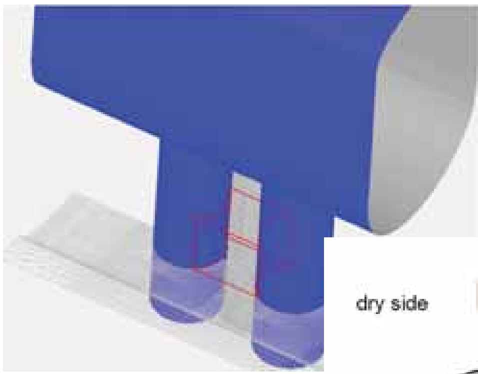
$$F(k_1, k_2, c_1, c_2, \ldots) = \sum_{ij} \left[u_{FEM}(x_i, t_j) - u_{ij} \right]^2$$

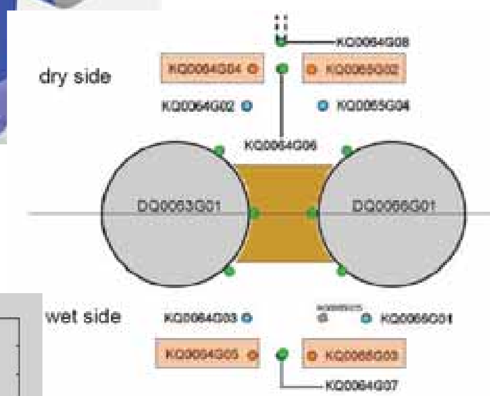
$$(k_1, k_2, c_1, c_2, \ldots) = \arg\min F(k_1, k_2, c_1, c_2, \ldots)$$

We use the discrete heat conduction model. Calibration of discretized model. Mesh dependence of optimal parameters ? Objective function includes the model evaluation. Another point of view: model create a constraint for the aim function.
Properties? Weighting, regularization in the objective function.

Parameters







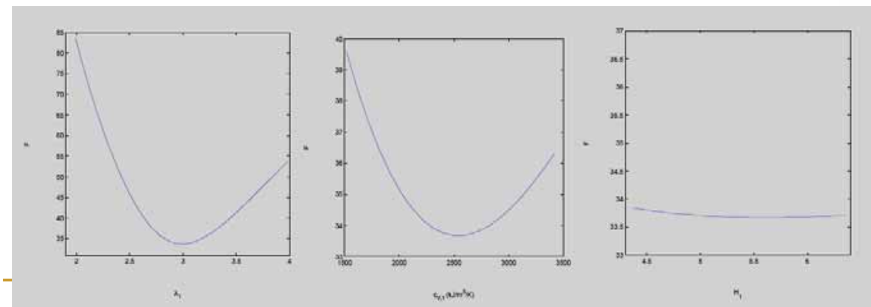
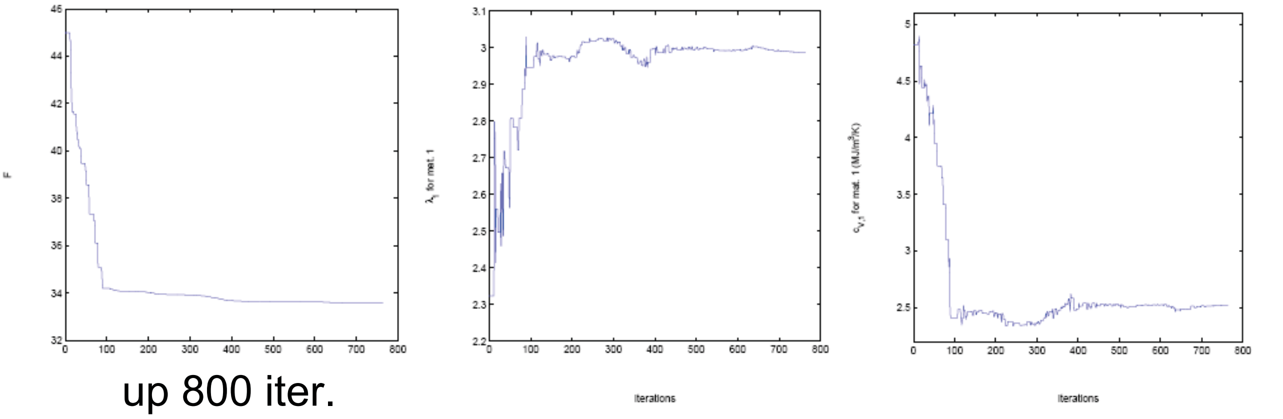


Figure 3: The dependence of the cost functional F on λ_1 (left), c_1 (center) and H_1 (right).

Convergence of MN for 9 parameters



up 800 iter.

Figure 2: The convergence of - the cost functional F (left), parameter λ_1 (center) and c_1 (right).

- Stop criterion $\Delta F, \Delta p < 0.001$

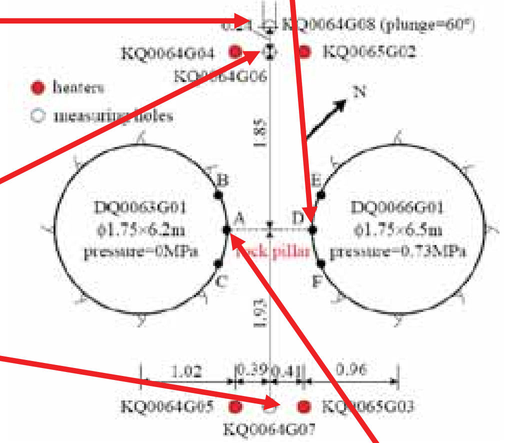
Accuracy

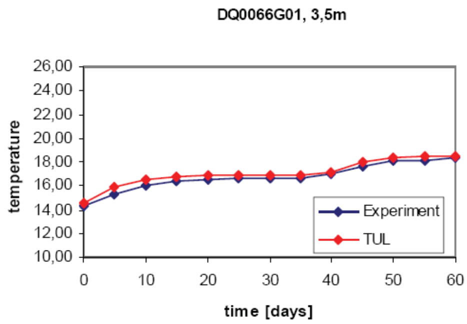
DQ0066G01		
1.5	3.5	5.5
0.24	0.33	0.91

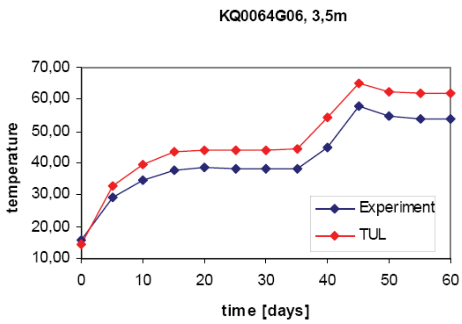
KQ0064G08	
2.75	5.5
0.23	0.12

KQ0064G06		
1.5	3.5	5.5
2.96	6.67	4.52

KQ0064G07		
1.5	3.5	5.5
2.59	3.02	1.78







DQ0063G01		
1.5	3.5	5.5
0.68	0.63	0.62

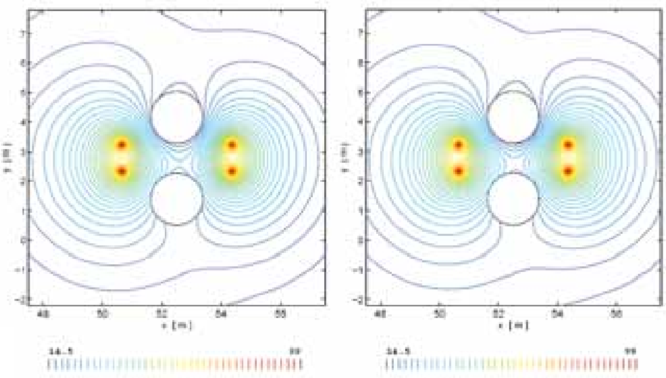
Time averages

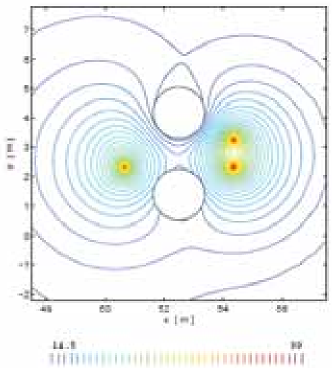
$$\varepsilon_T = \left(\frac{1}{12} \sum_{j=1}^{12} \left[u_{FEM}(x_i, t_j) - u_{ij} \right]^2 \right)^{1/2}$$

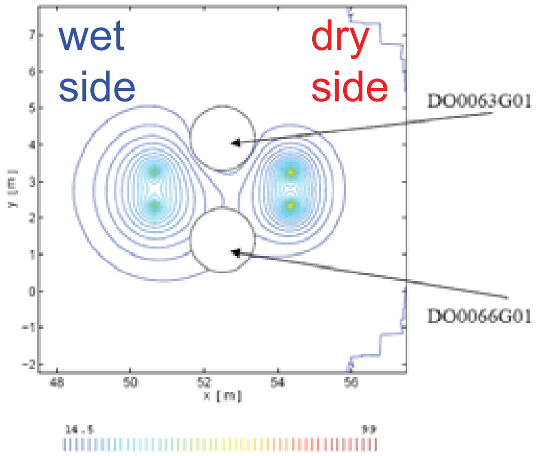
Temperature fields

Right: After 60 days of heating, depths 1.5, 3.5 and 5.5m

Below: After 5 days, depth 3.5m







Decrease of computational work: continuation in discretization parameter Δt

9 *parameters*,
$FEM\,(\Delta t = 0.0005,$
580 steps)
$stop\ if\ \Delta F \le 10^{-3}$
realistic initial gues
540 *iterations of* MN
790 *calls of* FEM
F = 33.6283
time = $56h52m04s$

FEM	560	*time steps*
# p	*# it*	*time [s]*
2	1372	418
4	1398	262
8	1476	181

A
$FEM\,(\Delta t = 0.01,\ 28\ \text{steps})$
$stop\ if\ \Delta F \le 10^{-1}$
realistic initial guess
167 *iterations of* MN
249 *calls of* FEM
F = 36.0679
time = $8h40m58s$

B
$FEM\,(\Delta t = 0.001,\ 280\ \text{steps})$
$stop\ if\ \Delta F \le 10^{-2}$
guess from A
161 *iterations of* MN
220 *calls of* FEM
F = 33.6357
time = $14h19m08s$

C
$FEM\,(\Delta t = 0.0005,\ 560\ \text{steps})$
$stop\ if\ \Delta F \le 10^{-3}$
gues from B
36 *iterations of* MN
62 *calls of* FEM
F = 33.6323
time = $4h27m47s$

Convergence from both realistic and non realistic initial guess

Twice shorter time,

Twice less NM iterations

Gradient methods - nonlinear least squares

$$F(k_1, k_2, c_1, c_2, \ldots) = F(p) = R(p)^T R(p)$$

$$R : R^N \to R^M \text{ , } N \text{ parameters, } M \text{ measurements, } (N < M)$$

p>0, unconstrained after an exponential transformation

- R *is differentiable*, $R(p) \approx M_c(p) = R(p_c) + J(p_c)(p - p_c)$

  Gauss – Newton method, Levenberg – Marquardt

  $$p^c \to p^+ = \arg\min M_c(p)^T M_c(p) \ \ (\text{linear } LS)$$

  $$p^+ = p^c - [J(p^c)^T J(p^c) + \mu_c I]^{-1} J(p^c)^T R(p^c), \ GN \Rightarrow \mu_c = 0$$

- Convergence dependent on nonlinearity of R and residuum $R(p^*)$

Nelder-Mead (simplex) method

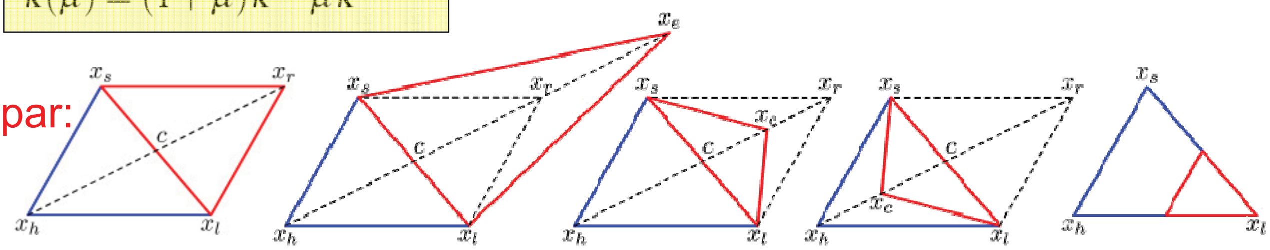
$$\kappa^{(k,1)}, \ldots, \kappa^{(k,p+1)}$$

$$F(\kappa^{(k,1)}) \le F(\kappa^{(k,2)}) \le \ldots \le F(\kappa^{(k,p+1)})$$

$$\kappa(\mu) = (1+\mu)\bar{\kappa} - \mu \kappa^{(k,p+1)}$$

Initial simplex around initial guess x_0, generally in R^N => New simplex Adaptation to local landscape, decreasing values at vertices

2par:



Reflect the point with highest F | Expand | Contract | Shrink

Stop – small decrease of functional & small size of simplex
Easier when the aim is decrease of F not exactly finding min F
- typically only 1-2 function evaluations at each step, while many other direct search methods use #param. or even more function evaluations.

Genetic algorithms

GA with $N = N_{GA}$ individuals

(1) generate N random vectors $\kappa^{(i)} \in \mathscr{K}$, $i = 1, \ldots, N$

(2) for given generation, evaluate (in parallel) $F_i = F(\kappa^{(i)})$, if F_i is not known yet,

(3) select τN parameter vectors $\kappa^{(i)}$ with smallest values F_i; so called parents. Then create $(1-\tau)N$ new vectors (childrens) by crossing randomly selected parents,

(4) create a new generation by taking the selected parents and created childrens with mutating some of them,

(5) evaluate stopping test and GOTO (2) if results are still not satisfactory.

K=Π<ai ,bi>

For vectors x,y

Crossing

$z_i = x_i + \alpha_i(y_i - x_i)$,

α_i randomly selected

Mutation

$z_i = x_i \pm \Delta_i 2^{-k\alpha}$,

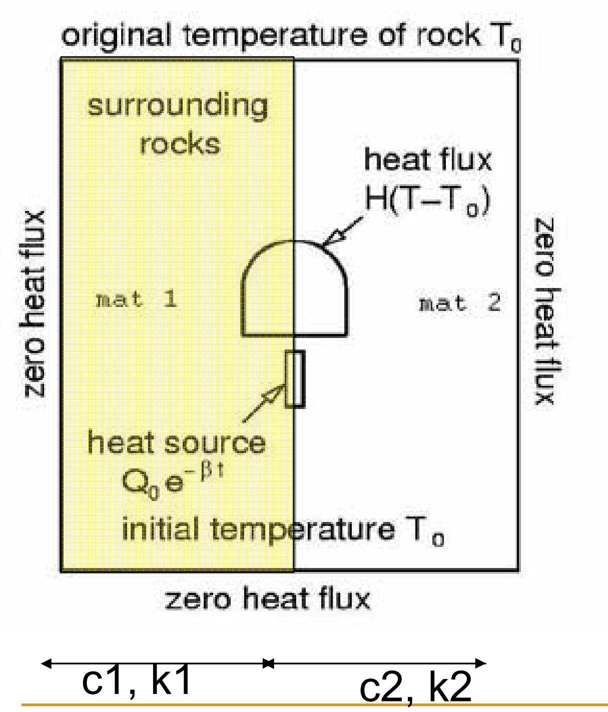
α randomly selected, mutat. range Δ_i

NM vs. GA, problem with 8 parameters

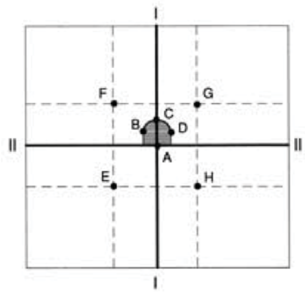
- N – size of generation = 20
- ½ of parents are selected, ½ new off is generated
- Coincidence F(GA) = 34.381 < F(NM)
- In 24 generations – lucky guess
- Method is sensitive to parameter bounds
- Heat conductivity λ [W/(m·K)], volume heat capacity c [MJ/(m3·K)], heat conduction coefficient H [W/(m2 ·K)].

	λ_1	c_1	λ_2	c_2	λ_3	c_3	H_1	H_2
NelderMead	2.984	2.640	4.605	1.504	5.478	1.830	5.524	8.284
GA.	3.008	2.559	4.460	1.811	8.635	2.890	5.662	7.346

A model problem and testing



- Identification of 4 material parameters



- Measured=computed temperatures in 8 points and 8 times (0.2, 0.5, 1.0, 4.0, 20.0, 30.0, 50.0,100.0) years

Comparison of methods

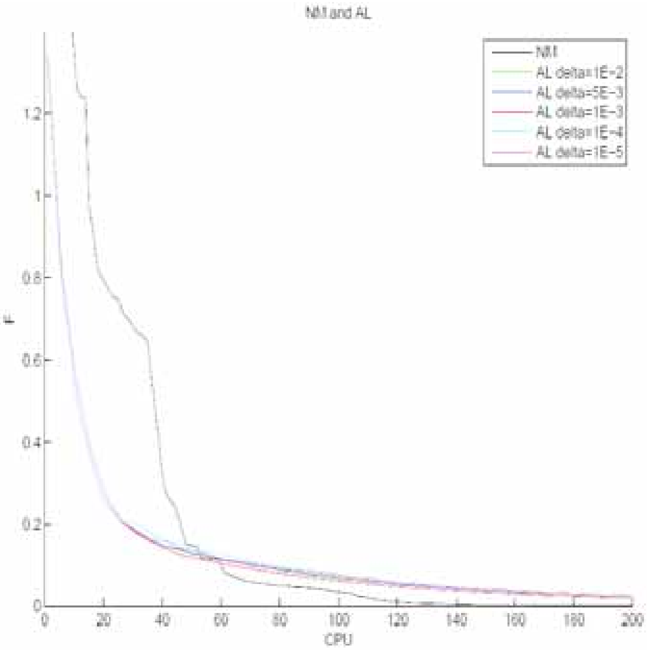
20 member populations

algorithm	nmbr of solutions	λ_1	c_1	λ_2	c_2	F
NM	150	2.9996	2.4046	5.9994	4.8041	0.0025
DR	740=200 +27*20	3.0653	2.4672	5.9338	4.7471	0.0404
EIR	1220=200 +51*20	3.0158	2.4209	5.9809	4.7868	0.0119
ELR	890=200 +34*20	3.0828	2.4628	5.9308	4.7629	0.0506
exact	-	3.0000	2.4030	6.0000	4.8060	0

Stopping rule:
NM : the difference $F(par_{worst}) - F(par_{best})$ in the simplex < 0.001
DR,EIR,ELR: the difference $F(par_{worst}) - F(par_{best})$ in the family < 0.001

Steepest descent with apriori line search

Input: p^0, ϵ, Δ
Output: p^r
for $(k = 0, 1, 2, \ldots)$ **do**
  $g^k \leftarrow app\_grad\left(F, x^k, \Delta\right);$
  $d_i^k \leftarrow \frac{g_i^k}{\|g^k\|};$
  $t^k \leftarrow LINESEARCH;$
  $p^{k+1} \leftarrow p^k + t_k d^k;$
  if $\left|F\left(p^{k+1}\right) - F\left(p^k\right)\right| < \epsilon$ **then**
    $BREAK;$
  end
end
$r \leftarrow k;$
return p^r;



$$\min_{x \in \mathbb{R}^n} F(x).$$

We solve above equation iteratively- each iteration has a form

$$x^{k+1} = x^k + t_k\, d^k, \quad d^k = -G^k\, \nabla F\left(x^k\right),$$

stepsize $t^k = \rho^{j_k}$, such that j_k is the smallest nonnegative integer j satisfying the following inequality for constants $\sigma \in (0,1)$ and $\rho \in (0,1)$

$$F\left(x^k + \rho^j\, d^k\right) - F\left(x^k\right) \leq \sigma \rho^j \nabla F\left(x^k\right)^T d^k$$

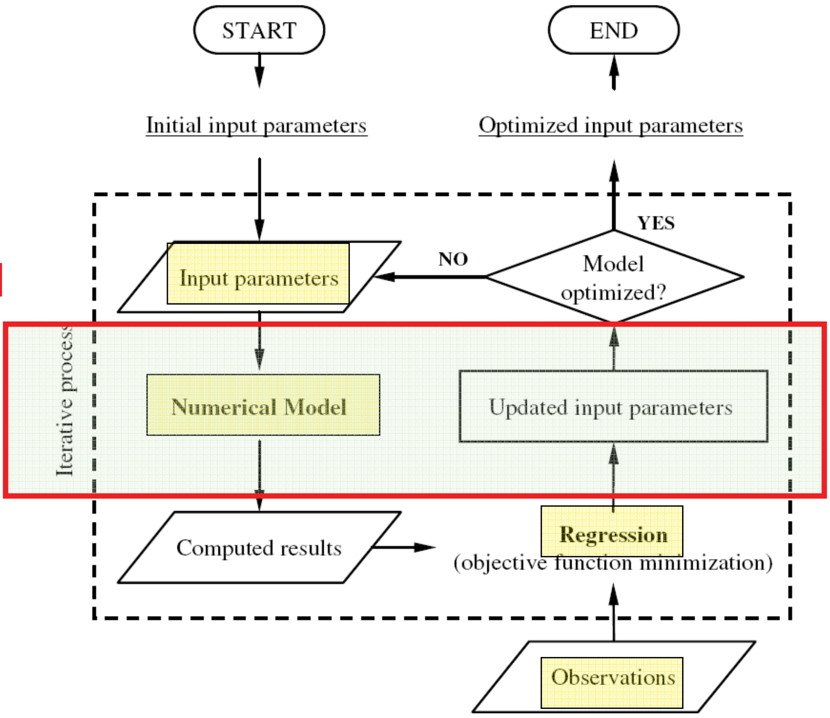
and using assumption that there exist constants $\lambda_* > 0$ and $\lambda^* > 0$ such that

$$\lambda_* \|d\|^2 \leq d^T\, G^k\, d \leq \lambda^* \|d\|^2,$$

then if $\left(G^k = I\right)$ $\inf_{x \in \mathbb{R}^n} \{F(x)\} = \lim_{k \to \infty} F\left(x^k\right)$

Parallel processing

- Application of parallel computing
  - within numerical model = FEM solution of
  - within an optimization method for regression

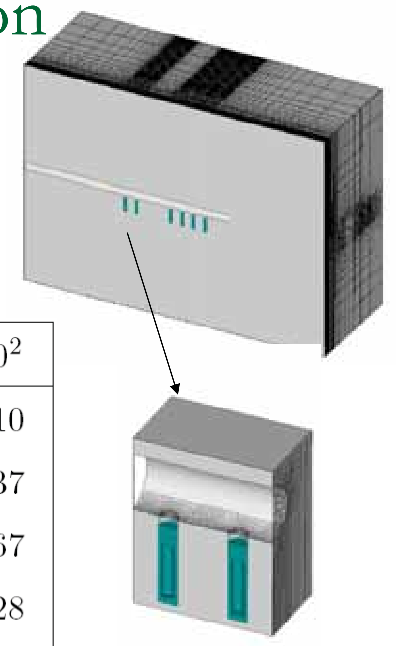


Optimization method and parallelization

- Gradient methods
  - //: computation of gradient
  - //: starting from different initial values to avoid local minima
- Nelder-Mead (simplex) direct method
  - //: starting from different initial values to avoid local minima
- Genetic algorithms
  - //: easy, parallel evaluation of new generation
- Combinations

Parallelization in model evaluation

- FEM + backward Euler
- one – level Schwarz method for systems $M+\Delta tA$, ILU on subdomains



$>$**2.5 mil. DOF**

#P \ Δt	10^{-4}	10^{-3}	10^{-2}	10^{-1}	10^{0}	10^{1}	10^{2}
1	12	12	17	27	39	61	110
4	14	14	17	25	40	68	137
8	16	18	22	25	40	84	167
16	16	18	22	25	41	99	228
24	16	18	22	26	42	97	262

Table: #iterations (Time step size x #subdomains)

APSE problem

$FEM \quad 560 \quad time \quad steps$

$\# p$	$\# it$	$time\ [s]$	Efficiency
2	1372	418	
4	1398	262	x
8	1476	181	x

BG Blue Gene

Continuous (not discretized) formulation

$discrete\ measurements \Rightarrow function\ \ z = z(x,t)$

$\min\ \ LS\ functionals$

$$1)\ J(k) = \tfrac{1}{2}\int_{T-\sigma}^{T} dt \int_{\Omega} k\left|u(k,x,t) - z(x,t)\right|^2 dx + \alpha \int_{\Omega} \left|\nabla k\right|^2 dx$$

$subjected\ to: u(k,x,t)\ is\ solution\ of\ the\ given\ parabolic\ problem$

$with\ heat\ conduction\ coefficient\ k$

$$2)\ J(k,v) = \tfrac{1}{2}\int_{T-\sigma}^{T} \left\| \tfrac{\partial}{\partial t}(v(t)-z(t)) - \nabla\cdot(k\nabla(v(t)-z(t))\right\|_{L2}^2 dx + \beta\left|k\right|_{H2}^2$$

$subjected\ to: e(k,v) = 0,\ where\ e\ is\ the\ solution\ of\ "error\ equation"$

$$\tfrac{\partial}{\partial t}e + \Delta e = \tfrac{\partial}{\partial t}v + \nabla(k\nabla v) - f\ \ in\ \Omega\times(0,T\rangle$$

$$e(x,0) = v(x,0) - u_0(x) + BC$$

Summary

- We discus identification/calibration of model with application in geoengineering
- Discuss numerical realization of identification using discrete parabolic model
- NM robust, reasonably efficient
- GA less robust, efficient on parallel platforms, incomplete convergence
- Combination, multiple run
- Continuation approach recommended
- Choice of parameters – stable and unstable minima
- Other applications

Conclusions

- We saw an application problem(s) for identification/calibration
- Discuss numerical realization of identification using discrete parabolic model
  - Choice of optimization method
  - Experience with Nelder – Mead algorithm
  - Discussion of other optimization algorithms
  - Choice of parameters – stable and unstable minima
  - Computational expense, parallel computing, continuation approach
- Use of continuous (not discretized) model

Conclusions

- Adaptation of the thermal model to real situation
  - Calibration parameters
  - Accuracy
- Inverse analysis
  - Nelder – Mead algorithm
  - Gauss-Newton
  - Genetics algorithms
- Modifications of the price functional
  - Weights $\sum w_{ij} |u_{FEM}-u_{ij}|$
  - Relative differences $\sum w_{ij} |u_{FEM}-u_{ij}| / |u_{ij}|$
  - Stresses $\sum w_{ij} |u_{FEM}-u_{ij}| / |u_{ij}| - w|\sigma_{ij}|$

NM robust, reasonably efficient
GA less robust, efficient on parallel platforms, incomplete convergence
Combination, multiple run
Continuation approach recommended
Choice of parameters
– stable and unstable minima

Thank you for your attention
and comments

Semismoothness and other properties of elastoplastic operator

Stanislav Sysala

Department of Applied Mathematics and Computer Science
Institute of Geonics AS CR, Ostrava

email: sysala@ugn.cas.cz

Motivation

Nonsmooth Newton method

System of non-linear equations:

$$F : \mathbb{R}^n \to \mathbb{R}^n : \quad F(\boldsymbol{u}) = \boldsymbol{f}$$

- F is generally non-differentiable and implicit function
- F is locally Lipschitz function

Nonsmooth Newton iterates:

$$\boldsymbol{u_{j+1}} := \boldsymbol{u_j} + \boldsymbol{V_j^{-1}}(\boldsymbol{f} - F(\boldsymbol{u_j})), \quad \boldsymbol{V_j} \in \partial F(\boldsymbol{u_j})$$

Local convergence assumptions (Kummer 1988; Qi, J. Sun 1991)

- F is locally Lipschitz function,
- F is semismooth at $\boldsymbol{u}$,
- all $\boldsymbol{V} \in \partial F(\boldsymbol{u})$ are non-singular.

Other useful property:

- F has a potential (generalized derivatives are symmetric)
- F is strictly monotone (generalized derivatives are positively definite)

- Semismoothness, Lipschitz continuity, monotonicity and potentiality will be investigated for the elastoplastic operator.

- Non-singularity of the generalized Jacobians depends generally on the load increment.

Contents

1. Semismooth functions.
2. Generalized projected mapping onto a convex set.
3. Elastoplastic constitutive problem.
4. One-time-step elastoplastic problem.
5. Classical isotropic yield functions.
6. Example of elastoplastic operator.
7. Conclusion.

1 Semismooth functions

Clarke's generalized derivative

X, Y - finite dimensional spaces, $F: X \to Y$ - locally Lipschitz function

$\mathcal{D}_F$ - set of points in X where F is Fréchet differentiable

$DF(x)$, $x \in \mathcal{D}_F$, - Fréchet derivative of F at x

Generalized derivative (Clarke 1983):

$$\partial F(x) := conv \left\{ \lim_{x_i \to x,\, x_i \in \mathcal{D}_F} DF(x_i) \right\}$$

Example: $F(x) := \max\{0, x\}$, $\lim\limits_{x \to 0^+} DF(x) = 1$, $\lim\limits_{x \to 0^-} DF(x) = 0$,

$$\partial F(0) = conv\{0, 1\} = [0, 1]$$

Definition of semismooth function (Qi, J. Sun 1993)

$F: X \to Y$ - locally Lipschitz continuous function is

semismooth at $x \in X$ if

(i) F is directionally differentiable at x,

(ii) for any $\triangle x \in X$, $\triangle x \to 0$, and $F^o \in \partial F(x + \triangle x)$,

$$F(x + \triangle x) - F(x) - F^o \triangle x = o(\|\triangle x\|).$$

strongly semismooth at $x \in X$ if

(ii)* for any $\triangle x \in X$, $\triangle x \to 0$, and $F^o \in \partial F(x + \triangle x)$,

$$F(x + \triangle x) - F(x) - F^o \triangle x = O(\|\triangle x\|^2).$$

Examples and properties of semismooth functions

- $C^1(\mathcal{O})$-functions are semismooth on $\mathcal{O} \subset X$.
- $C^{1,1}(\mathcal{O})$-functions are strongly semismooth on $\mathcal{O} \subset X$.
- Max-function is strongly semismooth.
- Scalar product, sum, compositions of (strongly) semismooth functions are (strongly) semismooth.
- If F is Lipschitz continuous on X and (strongly) semismooth a.e. on X then F is (strongly) semismooth on X.
- Let $F: \ X \to X$ be Lipschitz continuous and strictly monotone on X. Then F is (strongly) semismooth on X if and only if F^{-1} is (strongly) semismooth on X (Gowda 2004, Meng, D. Sun, Zhao 2005).

Implicit function theorem for semismooth functions

Let $\mathcal{I}: \ Y \times X \to X$ be a locally Lipschitz function in a neighborhood of $(\bar{y}, \bar{x})$, which solve $\mathcal{I}(\bar{y}, \bar{x}) = 0$. Let

$$[\partial_y \mathcal{I}(y,x), \partial_x \mathcal{I}(y,x)] := \partial \mathcal{I}(y,x).$$

If $\partial_x \mathcal{I}(\bar{y}, \bar{x})$ is of maximal rank, i.e.

$$\mathcal{I}_x^o \triangle x = 0, \ \mathcal{I}_x^o \in \partial_x \mathcal{I}(\bar{y}, \bar{x}) \quad \Longrightarrow \quad \triangle x = 0,$$

then there exist an open neighborhood $\mathcal{O}_{\bar{y}}$ of $\bar{y}$ and a function $F : \mathcal{O}_{\bar{y}} \to X$ such that F is locally Lipschitz continuous in $\mathcal{O}_{\bar{y}}$, $F(\bar{y}) = \bar{x}$ and for every y in $\mathcal{O}_{\bar{y}}$

$$\mathcal{I}(y, F(y)) = 0.$$

Moreover, if $\mathcal{I}$ is (strongly) semismooth at $(\bar{y}, \bar{x})$, then F is (strongly) semismooth at $\bar{y}$. (Clarke 1983, D. Sun 2001)

2 Generalized projective mapping onto a convex set

Assumptions

W is finite dimensional space, $G: W \to W$ fulfils:

(G1) G is Lipschitz continuous on W, i.e $\exists L > 0:$

$$\|G(\Theta_1) - G(\Theta_2)\| \leq L\|\Theta_1 - \Theta_2\| \quad \forall \Theta_1, \Theta_2 \in W,$$

(G2) G is strictly monotone on W, i.e. $\exists \alpha > 0$

$$(G(\Theta_1) - G(\Theta_2), \Theta_1 - \Theta_2) \geq \alpha\|\Theta_1 - \Theta_2\|^2 \quad \forall \Theta_1, \Theta_2 \in W,$$

(G3) G is (strongly) semismooth on W

(G4) G is potential on W, i.e $\exists \Psi_G: W \to \mathbb{R}$:

$$D\Psi_G(\Theta) = G(\Theta) \quad \forall \Theta \in W.$$

Definition and basic properties of the projection

K - closed, convex and non-empty set in W

$\Pi_K: W \to K$, $\Sigma := \Pi_K(P)$:

$$(P - G(\Sigma), \Theta - \Sigma) \leq 0 \quad \forall \Theta \in K$$

Let (G1), (G2) hold. Then

- Π_K is a single-valued mapping onto K,
- $\Pi_K(G(\Pi_K(P))) = \Pi_K(P)$,
- $\Pi_K(P) = G^{-1}(P)$ if and only if $G^{-1}(P) \in K$,
- $(\Pi_K(P_1) - \Pi_K(P_2), P_1 - P_2) \geq \alpha\|\Pi_K(P_1) - \Pi_K(P_2)\|^2 \quad \forall P_1, P_2 \in W$,
- $\|\Pi_K(P_1) - \Pi_K(P_2)\| \leq \frac{1}{\alpha}\|P_1 - P_2\| \quad \forall P_1, P_2 \in W$.

Potential function Π_K

Let (G1), (G2), (G4) hold and Ψ_G is a potential to G. Let

$$\Psi_\Pi(P) := (P, \Pi_K(P)) - \Psi_G\left(\Pi_K(P)\right), \quad P \in W.$$

Then

$$D\Psi_\Pi(P) = \Pi_K(P) \quad \forall P \in W.$$

Proof idea: using of a proximal mapping (Moreaux 1965):
f - convex, proper, lower semicontinuous function on W:

$$prox_f(P) = \text{arg} \min_{\Theta \in \mathcal{W}} \left\{ \frac{1}{2} \|P - \Theta\|^2 + f(\Theta) \right\}$$

The mapping $P \mapsto \Pi_K(\alpha P)$ is a proximal mapping with

$$f(\Theta) := \frac{1}{\alpha} \Psi_G(\Theta) - \frac{1}{2} \|\Theta\|^2 + I_K(\Theta)$$

Specification of the set K

$K := \{\Theta \in W : \ \Phi(\Theta) \leq 0\}$, $\Phi : \ W \to \mathbb{R}$ fulfils

(Φ1) Φ is convex on W,

(Φ2) $\Phi(0) < 0$,

(Φ3) Φ is a.e. differentiable on W.

(Φ4) Φ is a.e. differentiable on ∂K.

(Φ5) If $\Theta \in \mathcal{D}_\Phi$, then $\exists\ \mathcal{O}_\Theta$ such that $\Phi \in C^1(\mathcal{O}_\Theta)$ and $D\Phi$ is (strongly) semismooth on $\mathcal{O}_\Theta$.

(Φ6) If $\Theta \in \partial K$ and $\Theta \in \mathcal{D}_\Phi$, then $D\Phi(\Theta) \neq 0$.

Semismoothness of Π_K

Theorem 1 *Let G fulfil (G1)-(G3) and Φ fulfil $(\Phi 1)$-$(\Phi 6)$. Then Π_K is (strong) semismooth on W.*

Idea of the proof:

1. W - space partition:

$$
\begin{aligned}
M_1 &:= \{P \in W : \; \Sigma = \Pi_K(P) = G^{-1}(P) \in int(K)\}, \\
M_2 &:= \{P \in W : \; \Sigma = \Pi_K(P) \in \partial K, \; P \in int(N_{K,\Pi}(\Sigma))\}, \\
& \quad\; N_{K,\Pi}(\Sigma) := \{P \in W : \; \Pi_K(P) = \Sigma\}, \\
M_3 &:= \{P \in W : \; \Sigma = \Pi_K(P) \in \partial K, \; \Sigma \in \mathcal{D}_\Phi\}.
\end{aligned}
$$

It is only sufficient to prove the semismoothness on M_1, M_2 a M_3 since the measure of $W \setminus (M_1 \cup M_2 \cup M_3)$ in W vanishes.

Idea of the proof - continuation

2. Semismoothness on $M_1 = \{P \in W : \; \Sigma = \Pi_K(P) = G^{-1}(P) \in int(K)\}$:
   $\Sigma = \Pi_K(P) = G^{-1}(P)$, G^{-1} is (strongly) semismooth, M_1 is open.
3. Semismoothness on $M_2 = \{P : \Sigma = \Pi_K(P) \in \partial K, P \in int(N_{K,\Pi}(\Sigma))\}$:
   Π_K is constant in a neighborhood of $P \in M_2$.
4. Semismoothness on $M_3 = \{P \in W : \; \Sigma = \Pi_K(P) \in \partial K, \; \Sigma \in \mathcal{D}_\Phi\}$:
   Let $\bar{P} \in M_3$ and $\bar{\Sigma} := \Pi_K(\bar{P})$. KKT conditions:

$$G(\bar{\Sigma}) - \bar{P} + \bar{\gamma} D\Phi(\bar{\Sigma}) = 0,$$

$$\Phi(\bar{\Sigma}) \leq 0, \; \bar{\gamma} \geq 0, \; \bar{\gamma}\Phi(\bar{\Sigma}) = 0, \; i = 1, \dots, m.$$

The second conditions can be equivalently rewritten:

$$\Phi(\bar{\Sigma}) + \max\{0, -\Phi(\bar{\Sigma}) - \bar{\gamma}\} = 0.$$

Idea of the proof - continuation

Let $\mathcal{I}: \; W \times W \times \mathbb{R} \to W \times \mathbb{R}$,

$$\mathcal{I}(P;\Sigma,\gamma) = \begin{pmatrix} G(\Sigma) - P + \gamma D\Phi(\Sigma) \\ \Phi(\Sigma) + \max\{0, -\Phi(\Sigma) - \gamma\} \end{pmatrix}.$$

Then

$$\mathcal{I}(\bar{P};\bar{\Sigma},\bar{\gamma}) = 0.$$

$\mathcal{I}$ is (strongly) semismooth in a neighborhood of $(\bar{P};\bar{\Sigma},\bar{\gamma})$ and thus it is sufficient to prove

$$\mathcal{I}^o \begin{pmatrix} \delta\Sigma \\ \delta\gamma \end{pmatrix} = 0, \quad \mathcal{I}^o \in \partial_{(\Sigma,\gamma)}\mathcal{I}(\bar{P};\bar{\Sigma},\bar{\gamma}) \quad \Longrightarrow \quad \begin{pmatrix} \delta\Sigma \\ \delta\gamma \end{pmatrix} = 0.$$

A similar implication is proved in (Meng, D. Sun, Zhao 2005).

Derivative of Π_K in a vector representation

Let G be differentiable, Φ be twice differentiable, $W = \mathbb{R}^n$ and

$$\begin{aligned} M_1 &:= \{\boldsymbol{P} \in \mathbb{R}^n : \; \boldsymbol{\Sigma} = \Pi_K(\boldsymbol{P}) \in int(K)\}, \\ M_2 &:= \{\boldsymbol{P} \in \mathbb{R}^n : \; \boldsymbol{\Sigma} = \Pi_K(\boldsymbol{P}) \in \partial K, \; \boldsymbol{P} \in int(N_{K,\Pi}(\boldsymbol{\Sigma}))\}, \\ M_4 &:= \{\boldsymbol{P} \in \mathbb{R}^n : \; \boldsymbol{\Sigma} = \Pi_K(\boldsymbol{P}) \in \partial K, \; \boldsymbol{\Sigma} \neq G^{-1}(\boldsymbol{P}), \; \boldsymbol{\Sigma} \in D_\Phi\}. \end{aligned}$$

Then

$$D\Pi_K(\boldsymbol{P}) = \boldsymbol{C}(\boldsymbol{\Sigma},\gamma) = \begin{cases} (DG(\boldsymbol{\Sigma}))^{-1}, & \boldsymbol{P} \in M_1, \\ \boldsymbol{0}, & \boldsymbol{P} \in M_2, \\ \boldsymbol{A}^{-1} + \boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{A}^{-1}\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{A}^{-1}, & \boldsymbol{P} \in M_4, \end{cases}$$

with

$$\boldsymbol{A} = DG(\boldsymbol{\Sigma}) + \gamma D^2\Phi(\boldsymbol{\Sigma}), \quad \boldsymbol{B} = D\Phi(\boldsymbol{\Sigma}), \quad \boldsymbol{\Sigma} = \Pi_K(\boldsymbol{P}).$$

3 Elasto-plastic constitutive problem

Notation and assumptions

1. Additive decomposition of the strain tensor: $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon^e} + \boldsymbol{\varepsilon^p}$.
2. Linear elastic law: $\boldsymbol{\sigma} = \boldsymbol{D^e}\boldsymbol{\varepsilon}^e$, $D^e_{ijkl} = D^e_{ijlk} = D^e_{klij}$.
3. Simple combination of kinematic and non-linear isotropic hardening:

$$\boldsymbol{\beta} = a\boldsymbol{X}, \quad \kappa = H(\bar{\varepsilon}^p), \quad a > 0, \quad H: \ \mathbb{R}^+ \rightarrow \mathbb{R}^+$$

   $\boldsymbol{X}, \bar{\varepsilon}^p$ - kinematic (tensor) and isotropic (scalar) hardening variables,
   $\boldsymbol{\beta}, \kappa$ - corresponding hardening thermodynamical forces,
   H fulfils (G1)-(G4) and $H(0) = 0$ and can be extended on $\mathbb{R}$.
4. Generalized stress and strain: $\Sigma := (\boldsymbol{\sigma}, \boldsymbol{\beta}, \kappa)$ and $G(\Sigma) = (\boldsymbol{\varepsilon^e}, \boldsymbol{X}, \bar{\varepsilon}^p)$,
   $G: W \rightarrow W$, $W := S \times S \times \mathbb{R}$, $G(\Theta) := ((\boldsymbol{D^e})^{-1}\boldsymbol{\Theta_\sigma}, a^{-1}\boldsymbol{\Theta_\beta}, H^{-1}(\Theta_\kappa))$

Notation and assumptions - continuation

5. Yield function: $\Phi: W \rightarrow \mathbb{R}$, $\Phi(\boldsymbol{\sigma}, \boldsymbol{\beta}, \kappa) := \varphi(\boldsymbol{\sigma} - \boldsymbol{\beta}) - (\sigma_{y_0} + \kappa)$,
   $\sigma_{y_0} > 0$ ... initial yield stress or shear yield stress or cohesion or...
   $\varphi: \ S \rightarrow \mathbb{R}$, $S = \mathbb{R}^{3\times 3}_{sym}$, Φ fulfils (Φ_1)-(Φ_6).
6. Principle of maximum plastic dissipation:

$$\text{find} \ \ \Sigma := (\boldsymbol{\sigma}, \boldsymbol{\beta}, \kappa) \in K: \quad \Upsilon^p(\dot{P}; \Sigma) \geq \Upsilon^p(\dot{P}; \Theta) \quad \forall \Theta \in K$$

   $K := \{\Theta \in W: \ \Phi(\Theta) \leq 0\}$ ... set of admissible generalized stresses
   $\Upsilon^p(\dot{P}; \Theta) := \langle \boldsymbol{\dot{\varepsilon}^p}, \boldsymbol{\Theta_\sigma}\rangle + \langle -\boldsymbol{\dot{X}}, \boldsymbol{\Theta_\beta}\rangle + (-\dot{\bar{\varepsilon}}^p)\Theta_\kappa$ ... dissipation functional,
   $\dot{P} := (\boldsymbol{\dot{\varepsilon}^p}, -\boldsymbol{\dot{X}}, -\dot{\bar{\varepsilon}}^p) = (\boldsymbol{\dot{\varepsilon}} - \boldsymbol{\dot{\varepsilon}^e}, -\boldsymbol{\dot{X}}, -\dot{\bar{\varepsilon}}^p)$ ... plastic strain rate
   Equivalent formulation:

$$\text{find} \ \ \Sigma \in K: \ \ (\dot{P}, \Theta - \Sigma) \leq 0 \quad \forall \Theta \in K \ \ \text{or} \ \ \dot{P} \in N_K(\Sigma).$$

Elasto-plastic constitutive initial value problem

Given:

- the history of the strain tensor $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(t)$, $t \in [0, t_{\max}]$,
- the initial values

$$\boldsymbol{\varepsilon}(0) = 0, \quad \boldsymbol{\varepsilon}^{\boldsymbol{e}}(0) = 0, \quad \bar{\varepsilon}^p(0) = 0, \quad \boldsymbol{X}(0) = 0.$$

Find:

- the generalized stress $\Sigma(t) = (\boldsymbol{\sigma}(t), \boldsymbol{\beta}(t), \kappa(t)) \in K$:

$$(\dot{P}, \Theta - \Sigma) \leq 0 \quad \forall \Theta \in K, \quad \dot{P} = (\dot{\boldsymbol{\varepsilon}} - \dot{\boldsymbol{\varepsilon}}^{\boldsymbol{e}}, -\dot{\boldsymbol{X}}, -\dot{\bar{\varepsilon}}^p),$$

- the generalized strain

$$(\boldsymbol{\varepsilon}^{\boldsymbol{e}}(t), \boldsymbol{X}(t), \bar{\varepsilon}^p(t)) = G(\Sigma(t)).$$

Time discretization of constitutive elasto-plastic problem

Time discretization: $0 = t_0 < t_1 < \ldots < t_k < \ldots < t_N = t_{\max}$.

Implicit Euler method:

$$\dot{P}_k \equiv \dot{P}(t_k) \approx \frac{P(t_k) - P(t_{k-1})}{\triangle t_k} = \frac{P_k^t - G(\Sigma_k)}{\triangle t_k},$$

with

$$\Sigma_k \equiv \Sigma(t_k) = (\boldsymbol{\sigma}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}}, \kappa_k).$$

and a trial generalized strain

$$P_k^t = (\boldsymbol{\varepsilon}^{\boldsymbol{e}}_{\boldsymbol{k-1}} + \triangle\boldsymbol{\varepsilon}_{\boldsymbol{k}}, \boldsymbol{X}_{\boldsymbol{k-1}}, \bar{\varepsilon}^p_{k-1}), \quad \triangle\boldsymbol{\varepsilon}_{\boldsymbol{k}} = \boldsymbol{\varepsilon}_{\boldsymbol{k}} - \boldsymbol{\varepsilon}_{\boldsymbol{k-1}},$$

One-time-step elasto-plastic constitutive problem

Given:

- $P_{k-1} = (\boldsymbol{\varepsilon}^{\boldsymbol{e}}_{\boldsymbol{k-1}}, \boldsymbol{X}_{\boldsymbol{k-1}}, \bar{\varepsilon}^p_{k-1})$ such that $\Sigma_{k-1} = G^{-1}(P_{k-1}) \in K$
- $\triangle\boldsymbol{\varepsilon}_{\boldsymbol{k}} = \boldsymbol{\varepsilon}_{\boldsymbol{k}} - \boldsymbol{\varepsilon}_{\boldsymbol{k-1}}$

Find: $\Sigma_k = (\boldsymbol{\sigma}_{\boldsymbol{k}}, \boldsymbol{\beta}_{\boldsymbol{k}}, \kappa_k) \in K$:

$$(P_k^t - G(\Sigma_k), \Theta - \Sigma_k) \leq 0 \quad \forall \Theta \in K, \quad P_k^t = (\boldsymbol{\varepsilon}^{\boldsymbol{e}}_{\boldsymbol{k-1}} + \triangle\boldsymbol{\varepsilon}_{\boldsymbol{k}}, \boldsymbol{X}_{\boldsymbol{k-1}}, \bar{\varepsilon}^p_{k-1}),$$

or

$$\Sigma_k = \Pi_K(P_k^t),$$

Compute:

$$P_k = (\boldsymbol{\varepsilon}^{\boldsymbol{e}}_{\boldsymbol{k}}, \boldsymbol{X}_{\boldsymbol{k}}, \bar{\varepsilon}^p_k) = G(\Sigma_k).$$

Correctness of the extended function H

$H: \mathbb{R}^+ \to \mathbb{R}^+$ fulfils (G1)-(G4) and $H(0) = 0$, $\kappa = H(\bar{\varepsilon}^p)$

- possible extension to the whole $\mathbb{R}$ such that (G1)-(G4) hold
- it is necessary to verify that $\kappa_k \geq 0$, where $\Sigma_k = \Pi_K(P_k^t)$
- it holds

$$0 = \kappa_0 \leq \kappa_1 \leq \ldots \leq \kappa_k \leq \ldots \leq \kappa_N$$

- this is in accordance with mechanical assumptions

4 One-time-step elastoplastic problem

$\Omega \subset \mathbb{R}^3$ - investigated domain with partially fixed boundary

$V \subset [H^1(\Omega)]^3$ - space of kinematically admissible displacement

$f \in V^*$ - the load (a combination of surface and volume loads)

Find: $\boldsymbol{u_k} \in V$, $\boldsymbol{\sigma_k}, \boldsymbol{\beta_k}, \kappa_k$, $\boldsymbol{\varepsilon^e_k}, \boldsymbol{X_k}, \bar{\varepsilon}^p_k$:

$$\int_\Omega \langle \boldsymbol{\sigma_k}, \boldsymbol{\varepsilon}(\boldsymbol{v}) \rangle \boldsymbol{dx} = f_k(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in V,$$

$$\begin{aligned}
(\boldsymbol{\sigma_k}, \boldsymbol{\beta_k}, \kappa_k) &= \Pi_K(\boldsymbol{\varepsilon^e_{k-1}} + \boldsymbol{\varepsilon}(\triangle \boldsymbol{u_k}), \boldsymbol{X_{k-1}}, \bar{\varepsilon}^p_{k-1}) \quad \text{a.e. in } \Omega, \\
(\boldsymbol{\varepsilon^e_k}, \boldsymbol{X_k}, \bar{\varepsilon}^p_k) &= G(\boldsymbol{\sigma_k}, \boldsymbol{\beta_k}, \kappa_k) \quad \text{a.e. in } \Omega,
\end{aligned}$$

where $\triangle \boldsymbol{u_k} = \boldsymbol{u_k} - \boldsymbol{u_{k-1}}$.

Reformulation of the one-time-step problem

Let

$$\begin{aligned}
(\Pi_{K,\sigma}(P), \Pi_{K,\beta}(P), \Pi_{K,\kappa}(P)) &:= \Pi_K(P), \quad P \in W, \\
T_k : \; S \to S, \quad T_k(\boldsymbol{\eta}) &:= \Pi_{K,\sigma}(\boldsymbol{\varepsilon^e_{k-1}} + \boldsymbol{\eta}, \boldsymbol{X_{k-1}}, \bar{\varepsilon}^p_{k-1}) - \boldsymbol{\sigma_{k-1}}
\end{aligned}$$

Then

$$\triangle \boldsymbol{\sigma_k} := \boldsymbol{\sigma_k} - \boldsymbol{\sigma_{k-1}} = T_k(\triangle \boldsymbol{\varepsilon_k}).$$

Find: $\boldsymbol{u_k} \in V$, $\boldsymbol{\sigma_k}, \boldsymbol{\beta_k}, \kappa_k$, $\boldsymbol{\varepsilon^e_k}, \boldsymbol{X_k}, \bar{\varepsilon}^p_k$:

$$\int_\Omega \langle T_k(\boldsymbol{\varepsilon}(\triangle \boldsymbol{u_k})), \boldsymbol{\varepsilon}(\boldsymbol{v}) \rangle \boldsymbol{dx} = \triangle f_k(\boldsymbol{v}) \quad \forall \boldsymbol{v} \in V,$$

$$\begin{aligned}
(\boldsymbol{\sigma_k}, \boldsymbol{\beta_k}, \kappa_k) &= \Pi_K(\boldsymbol{\varepsilon^e_{k-1}} + \boldsymbol{\varepsilon}(\triangle \boldsymbol{u_k}), \boldsymbol{X_{k-1}}, \bar{\varepsilon}^p_{k-1}) \quad \text{a.e. in } \Omega, \\
(\boldsymbol{\varepsilon^e_k}, \boldsymbol{X_k}, \bar{\varepsilon}^p_k) &= G(\boldsymbol{\sigma_k}, \boldsymbol{\beta_k}, \kappa_k) \quad \text{a.e. in } \Omega,
\end{aligned}$$

Properties of $T_k(\boldsymbol{\eta}) = \Pi_{K,\sigma}(\boldsymbol{\varepsilon^e_{k-1}} + \boldsymbol{\eta}, \boldsymbol{\beta_{k-1}}, \kappa_{k-1}) - \boldsymbol{\sigma_{k-1}}$

Let

- the constitutive mapping G fulfils (G1)-(G4),
- the yield function Φ fulfils $(\Phi1) - (\Phi6)$,

Then

- $\Psi_{T_k}(\boldsymbol{\eta}) = \langle \boldsymbol{\eta}, T_k(\boldsymbol{\eta}) + \boldsymbol{\sigma_{k-1}} \rangle - \frac{1}{2}\langle (\boldsymbol{D^e})^{-1} T_k(\boldsymbol{\eta}), T_k(\boldsymbol{\eta}) \rangle$, $D\Psi_{T_k} = T_k$,
- T_k is Lipschitz continuous and monotone on S,
- T_k is (strongly) semismooth on S,
- the generalized derivative of T_k are symmetric and positively semidefinite.

Derivative of the elastoplastic operator T_k

Let

$$\begin{aligned} P_\eta &:= (\boldsymbol{\varepsilon^e_{k-1}} + \boldsymbol{\eta}, \boldsymbol{X_{k-1}}, \bar{\varepsilon}^p_{k-1}) \in \mathcal{D}_{\Pi_K}, \\ \Pi_K(P_\eta) &:= (\Pi_{K,\sigma}(P_\eta), \Pi_{K,\beta}(P_\eta), \Pi_{K,\kappa}(P_\eta)), \\ T_k(\boldsymbol{\eta}) &:= \Pi_{K,\sigma}(P_\eta) - \boldsymbol{\sigma_{k-1}}. \end{aligned}$$

Then

$$DT_k(\boldsymbol{\eta}) = \frac{\partial \Pi_{K,\sigma}(P_\eta)}{\partial \boldsymbol{\sigma}}.$$

If $G^{-1}(P_\eta) = (\boldsymbol{\sigma_{k-1}} + \boldsymbol{D}^e\boldsymbol{\eta}, \boldsymbol{\beta_{k-1}}, \kappa_{k-1}) \in int(K)$ (elastic region), then

$$T_k(\boldsymbol{\eta}) = \boldsymbol{D}^e\boldsymbol{\eta}, \quad DT_k(\boldsymbol{\eta}) = \boldsymbol{D}^e.$$

FEM discretization of elastoplastic problem

τ_h - triangulation of Ω with linear simplex elements

V_h - approximation of V_h (piecewise linear functions)

Non-linear operator: $F_{k,h} : V_h \to V_h$

$$(F_{k,h}(\boldsymbol{v}_h), \boldsymbol{w}_h)_h := \sum_{\mathcal{K}\in\tau_h} |\mathcal{K}| \langle T_k(\boldsymbol{\varepsilon}(\boldsymbol{v}_h)|_{\mathcal{K}}), \boldsymbol{\varepsilon}(\boldsymbol{w}_h)|_{\mathcal{K}}\rangle \boldsymbol{dx} \quad \forall \boldsymbol{v}_h, \boldsymbol{w}_h \in V_h.$$

Vector representation: $\boldsymbol{v}_h \in V_h \mapsto \mathbf{v} \in \mathbb{R}^{\mathbf{n}}$, $\boldsymbol{f}_h \in V_h \mapsto \mathbf{f} \in \mathbb{R}^{\mathbf{n}}$,

$$\mathcal{F}_k: \ \mathbb{R}^n \to \mathbb{R}^n, \quad \mathcal{F}_k(\mathbf{v}) = F_{k,h}(\boldsymbol{v}_h) \quad \forall \boldsymbol{v}_h \in V_h.$$

One-time step elastoplastic problem:

$$\text{find } \triangle\mathbf{u_k} \in \mathbb{R}^{\mathbf{n}}: \quad \mathcal{F}_{\mathbf{k}}(\triangle\mathbf{u_k}) = \triangle\mathbf{f_k}.$$

$\mathcal{F}_k$ - similar properties as T_k, semismoothness depends on h.

5 Classical isotropic yield functions

Scalar isotropic functions

Definition: $S := \mathbb{R}^{3\times3}_{sym}$,

$$\varphi: \ S \to \mathbb{R}, \quad \varphi(\boldsymbol{\sigma}) = \varphi(\boldsymbol{Q}\boldsymbol{\sigma}\boldsymbol{Q}^T) \quad \forall \boldsymbol{Q} \in Orth$$

Eigenvalues representation: $\boldsymbol{\sigma} = \sum_{i=1}^{3} \sigma_i \boldsymbol{e_i} \otimes \boldsymbol{e_i}$

$$\varphi(\boldsymbol{\sigma}) = \hat{\varphi}(\sigma_1, \sigma_2, \sigma_3) = \hat{\varphi}(\sigma_2, \sigma_1, \sigma_3) = \hat{\varphi}(\sigma_1, \sigma_3, \sigma_2) \quad \forall \boldsymbol{\sigma} \in S,$$

First derivative (Ogden 1984):

$$\frac{\partial \varphi}{\partial \boldsymbol{\sigma}} = \sum_{i=1}^{3} \frac{\partial \hat{\varphi}}{\partial \sigma_i} \boldsymbol{e_i} \otimes \boldsymbol{e_i}$$

Second derivative of isotropic function (Carlson, Hoger 1986)

Let $\boldsymbol{\sigma} \in S$, $\sigma_1 \neq \sigma_2 \neq \sigma_3 \neq \sigma_1$. Then the eigenprojections of $\boldsymbol{\sigma}$ fulfil

$$\boldsymbol{E_i} = \prod_{j=1, j\neq i}^{3} \frac{1}{(\sigma_i - \sigma_j)} (\boldsymbol{\sigma} - \sigma_i \boldsymbol{I}) = e_i \otimes e_i, \quad i = 1, 2, 3.$$

$$\begin{aligned} D^2\varphi(\boldsymbol{\sigma}) = \sum_{i,j=1}^{3} \frac{\partial \hat{\varphi}^2}{\partial \sigma_i \partial \sigma_j} \boldsymbol{E_i} \otimes \boldsymbol{E_j} + \sum_{a=1}^{3} \frac{\partial \hat{\varphi}/\partial \sigma_a}{(\sigma_a - \sigma_b)(\sigma_a - \sigma_c)} \Bigg\{ \frac{\partial \boldsymbol{\sigma}^2}{\partial \boldsymbol{\sigma}} - \\ -(\sigma_b + \sigma_c) I_\sigma - [(\sigma_a - \sigma_b) + (\sigma_a - \sigma_c)] \boldsymbol{E_a} \otimes \boldsymbol{E_a} - \\ - (\sigma_b - \sigma_c)(\boldsymbol{E_b} \otimes \boldsymbol{E_b} - \boldsymbol{E_c} \otimes \boldsymbol{E_c}) \Bigg\}, \end{aligned}$$

where $a \neq b \neq c \neq a$ and

$(\partial \boldsymbol{\sigma}^2 / \partial \boldsymbol{\sigma}) \boldsymbol{T} = \boldsymbol{\sigma} \boldsymbol{T} + \boldsymbol{T} \boldsymbol{\sigma}$, $\boldsymbol{I_\sigma} \boldsymbol{T} = \boldsymbol{T}$, $(\boldsymbol{E_i} \otimes \boldsymbol{E_j}) \boldsymbol{T} = \langle \boldsymbol{E_j}, \boldsymbol{T} \rangle \boldsymbol{E_i} \quad \forall \boldsymbol{T} \in S$

The second derivative of scalar-valued isotropic functions (Carlson, Hoger 1986)

- Isotropic function φ is twice continuously differentiable in a neighborhood of $\boldsymbol{\sigma} \in S$, $\sigma_1 \neq \sigma_2 \neq \sigma_3 \neq \sigma_1$ if $\hat{\varphi}$ is twice continuously differentiable in a neighborhood of $(\sigma_1, \sigma_2, \sigma_3)$.

- The same implication holds for semismoothness of $D\varphi$ a neighborhood of $\boldsymbol{\sigma} \in S$, $\sigma_1 \neq \sigma_2 \neq \sigma_3 \neq \sigma_1$

Assumptions on isotropic function

$$\Phi : W \to \mathbb{R}, \quad \Phi(\boldsymbol{\sigma}, \boldsymbol{\beta}, \kappa) := \varphi(\boldsymbol{\sigma} - \boldsymbol{\beta}) - (\sigma_{y_0} + \kappa)$$

$\varphi : S \to \mathbb{R}$, φ is isotropic, $\varphi(\mathbf{0}) = 0$ and

($\varphi 1$) φ is convex on S,

($\varphi 2$) φ is a.e. differentiable on S.

($\varphi 3$) φ is a.e. differentiable on the set

$$K_c := \{\boldsymbol{\sigma} \in S : \ \varphi(\boldsymbol{\sigma}) = c\}, \quad c > 0.$$

($\varphi 4$) If $\boldsymbol{\sigma} \in \mathcal{D}_\varphi$, then $\exists O_{\boldsymbol{\sigma}}$, $\varphi \in C^2(\mathcal{O}_{\boldsymbol{\sigma}})$.

($\varphi 5$) If $\boldsymbol{\sigma} \in K_c$ and $\boldsymbol{\sigma} \in \mathcal{D}_\varphi$, then $D\varphi(\boldsymbol{\sigma}) \neq \mathbf{0}$.

Von Mises yield function

$$\Phi(\boldsymbol{\sigma}, \boldsymbol{\beta}, \kappa) = \varphi(\boldsymbol{\sigma} - \boldsymbol{\beta}) - (\sigma_{y_0} + \kappa), \ \ \varphi(\boldsymbol{\sigma}) := \sqrt{\frac{3}{2}} \|dev(\boldsymbol{\sigma})\|_F,$$

$$p(\boldsymbol{\sigma}) := \frac{1}{3} tr(\boldsymbol{\sigma}) = \frac{1}{3}(\sigma_{11} + \sigma_{22} + \sigma_{33}), \quad dev(\boldsymbol{\sigma}) = \boldsymbol{\sigma} - p(\boldsymbol{\sigma})\boldsymbol{I}.$$

- φ is isotropic and convex on S, $\varphi(\mathbf{0}) = 0$
- φ is not differentiable only for $\boldsymbol{\sigma} = p(\boldsymbol{\sigma})\boldsymbol{I}$, otherwise is C^∞ smooth
- φ is differentiable on $K_c = \{\boldsymbol{\sigma} : \ \varphi(\boldsymbol{\sigma}) = c\}$, $c > 0$

$$D\varphi(\boldsymbol{\sigma}) = \sqrt{\frac{3}{2}} \hat{\boldsymbol{n}}(\boldsymbol{\sigma}), \quad D^2\varphi(\boldsymbol{\sigma}) = \frac{\boldsymbol{I}_{dev} - \hat{\boldsymbol{n}}(\boldsymbol{\sigma}) \otimes \hat{\boldsymbol{n}}(\boldsymbol{\sigma})}{\|dev(\boldsymbol{\sigma})\|_F},$$

$$\hat{\boldsymbol{n}}(\boldsymbol{\sigma}) = \frac{dev(\boldsymbol{\sigma})}{\|dev(\boldsymbol{\sigma})\|_F}, \quad \boldsymbol{I}_{dev}\boldsymbol{\sigma} = dev(\boldsymbol{\sigma}).$$

Drucker-Prager yield function

$$\Phi(\boldsymbol{\sigma},\boldsymbol{\beta},\kappa)=\varphi(\boldsymbol{\sigma}-\boldsymbol{\beta})-\xi(c_0+\kappa),\ \ \varphi(\boldsymbol{\sigma}):=\sqrt{\frac{1}{2}}\|dev(\boldsymbol{\sigma})\|_F+\eta p(\boldsymbol{\sigma}),$$

$$p(\boldsymbol{\sigma}):=\frac{1}{3}tr(\boldsymbol{\sigma})=\frac{1}{3}(\sigma_{11}+\sigma_{22}+\sigma_{33}),\quad dev(\boldsymbol{\sigma})=\boldsymbol{\sigma}-p(\boldsymbol{\sigma})\boldsymbol{I},$$

$c_0>0$ - initial cohesion, $\eta,\xi>0$ - given parameters

- φ is isotropic and convex on S, $\varphi(\mathbf{0})=0$
- φ is not differentiable only for $\boldsymbol{\sigma}=p(\boldsymbol{\sigma})\boldsymbol{I}$, otherwise is C^∞ smooth
- φ is not differentiable on $K_c=\{\boldsymbol{\sigma}:\ \varphi(\boldsymbol{\sigma})=c\}$ only at $\boldsymbol{\sigma}=(\xi c/\eta)\boldsymbol{I}$

$$D\varphi(\boldsymbol{\sigma})=\sqrt{\frac{3}{2}}\hat{\boldsymbol{n}}(\boldsymbol{\sigma})+\frac{\eta}{3}\boldsymbol{I},\quad D^2\varphi(\boldsymbol{\sigma})=\frac{\boldsymbol{I}_{dev}-\hat{\boldsymbol{n}}(\boldsymbol{\sigma})\otimes\hat{\boldsymbol{n}}(\boldsymbol{\sigma})}{\|dev(\boldsymbol{\sigma})\|_F},$$

Tresca yield function

$$\Phi(\boldsymbol{\sigma},\boldsymbol{\beta},\kappa)=\varphi(\boldsymbol{\sigma}-\boldsymbol{\beta})-(\sigma_{y_0}+\kappa),\ \ \varphi(\boldsymbol{\sigma})=\hat{\varphi}(\sigma_1,\sigma_2,\sigma_3):=\sigma_{\max}-\sigma_{\min},$$

- convexity of φ follows from

$$\sigma_{\max}=\sup_{\|\boldsymbol{\nu}\|=1}(\boldsymbol{\sigma\nu},\boldsymbol{\nu}),\quad \sigma_{\min}=\inf_{\|\boldsymbol{\nu}\|=1}(\boldsymbol{\sigma\nu},\boldsymbol{\nu})$$

- $\hat{\varphi}$ is C^∞ smooth at $(s_1,s_2,s_3)\in\mathbb{R}^3$, $s_i\neq s_j$, $i\neq j$
- hence φ is C^∞ smooth at $\boldsymbol{\sigma}\in\mathbb{S}$, $\sigma_i\neq\sigma_j$, $i\neq j$
- $\hat{\varphi}$ is a.e. differentiable on $\hat{K}_c=\{(s_1,s_2,s_3):\ \hat{\varphi}(s_1,s_2,s_3)=c\}$, $c>0$
- hence φ is a.e. differentiable on $K_c=\{\boldsymbol{\sigma}:\ \varphi(\boldsymbol{\sigma})=c\}$, $c>0$

Derivative of Tresca yield function

Let $s_1 > s_2 > s_3$. Then $\hat{\varphi}(s_1, s_2, s_3) = s_1 - s_3$ and

$$D_1\hat{\varphi} = 1, \quad D_2\hat{\varphi} = 0, \quad D_3\hat{\varphi} = -1, \quad D^2\hat{\varphi} = 0.$$

Then

$$D\phi(\boldsymbol{\sigma}) = \boldsymbol{e_1} \otimes \boldsymbol{e_1} - \boldsymbol{e_3} \otimes \boldsymbol{e_3} = \boldsymbol{E_1} - \boldsymbol{E_3}, \quad \sigma_1 > \sigma_2 > \sigma_3,$$

$$D^2\varphi(\boldsymbol{\sigma}) = \frac{1}{(\sigma_1 - \sigma_2)(\sigma_1 - \sigma_3)} \left\{ \frac{\partial \boldsymbol{\sigma}^2}{\partial \boldsymbol{\sigma}} - (\sigma_2 + \sigma_3)\boldsymbol{I_\sigma} - \right.$$

$$\left. -[(\sigma_1 - \sigma_2) + (\sigma_1 - \sigma_3)]\boldsymbol{E_1} \otimes \boldsymbol{E_1} - (\sigma_2 - \sigma_3)(\boldsymbol{E_2} \otimes \boldsymbol{E_2} - \boldsymbol{E_3} \otimes \boldsymbol{E_3}) \right\}$$

$$- \frac{1}{(\sigma_3 - \sigma_1)(\sigma_3 - \sigma_1)} \left\{ \frac{\partial \boldsymbol{\sigma}^2}{\partial \boldsymbol{\sigma}} - (\sigma_1 + \sigma_2)\boldsymbol{I_\sigma} - \right.$$

$$\left. -[(\sigma_3 - \sigma_1) + (\sigma_3 - \sigma_2)]\boldsymbol{E_3} \otimes \boldsymbol{E_3} - (\sigma_1 - \sigma_2)(\boldsymbol{E_1} \otimes \boldsymbol{E_1} - \boldsymbol{E_2} \otimes \boldsymbol{E_2}) \right\}$$

Mohr-Coulomb yield function

$$\Phi(\boldsymbol{\sigma}, \boldsymbol{\beta}, \kappa) = \varphi(\boldsymbol{\sigma} - \boldsymbol{\beta}) - 2(c_0 + \kappa)\cos\psi,$$

$$\begin{aligned} \varphi(\boldsymbol{\sigma}) = \hat{\varphi}(\sigma_1, \sigma_2, \sigma_3) \quad &:= \quad (\sigma_{\max} - \sigma_{\min}) + (\sigma_{\max} + \sigma_{\min})\sin\psi \\ &= \quad (1 + \sin\psi)\sigma_{\max} - (1 - \sin\psi)\sigma_{\min} \end{aligned}$$

$c_0 > 0$ - initial cohesion
$\psi \in [0, \pi/2)$ - frictional angle

- similar properties and derivatives as Tresca yield function

6 Example of elastoplastic operator

Von Mises yield criterion with nonlinear isotropic hardening

$$\Phi(\boldsymbol{\sigma}, \boldsymbol{\kappa}) = \sqrt{\frac{3}{2}}\|dev(\boldsymbol{\sigma})\|_F - (\sigma_{y_0} + \kappa),\ \boldsymbol{D^e}\boldsymbol{\varepsilon^e} = \lambda tr(\boldsymbol{\varepsilon^e})\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon^e}, \kappa = H(\bar{\varepsilon}^p)$$

$$T_k(\boldsymbol{\eta}) = \boldsymbol{D^e}\boldsymbol{\eta} - 2\mu\sqrt{\frac{3}{2}}\gamma^+\hat{\boldsymbol{n}}(\boldsymbol{\sigma}(\boldsymbol{\eta})),$$

$$\sqrt{\frac{3}{2}}\|dev(\boldsymbol{\sigma}(\boldsymbol{\eta}))\|_F - 3\mu\gamma - (\sigma_{y_0} + H(\bar{\varepsilon}^p_{k-1} + \gamma)) = 0 \quad \text{if } \Phi(\boldsymbol{\sigma}(\boldsymbol{\eta}), \kappa_{k-1}) > 0,$$

$$\hat{\boldsymbol{n}}(\boldsymbol{\sigma}) = \frac{dev(\boldsymbol{\sigma})}{\|dev(\boldsymbol{\sigma})\|_F}, \quad \boldsymbol{\sigma}(\boldsymbol{\eta}) = \boldsymbol{\sigma_{k-1}} + \boldsymbol{D^e}\boldsymbol{\eta}$$

Potential and derivative of the von Mises operator

$$\begin{aligned}\Psi_{T_k}(\boldsymbol{\eta}) &= \langle \boldsymbol{\eta}, T_k(\boldsymbol{\eta}) + \boldsymbol{\sigma_{k-1}} \rangle - \frac{1}{2}\langle (\boldsymbol{D^e})^{-1} T_k(\boldsymbol{\eta}), T_k(\boldsymbol{\eta}) \rangle = \\ &= \frac{1}{2}\langle (\boldsymbol{D^e})^{-1}\boldsymbol{\sigma}(\boldsymbol{\eta}), \boldsymbol{\sigma}(\boldsymbol{\eta}) \rangle - 3\mu^2(\gamma^+)^2.\end{aligned}$$

$$DT_k(\boldsymbol{\eta}) = \begin{cases} \boldsymbol{D^e}, & \Phi(\boldsymbol{\sigma}(\boldsymbol{\eta}), \kappa_{k-1}) < 0, \\ \boldsymbol{D^e} - 2\mu(\bar{\beta}\boldsymbol{I}_{dev} + \bar{\gamma}\hat{\boldsymbol{n}} \otimes \hat{\boldsymbol{n}}), & \Phi(\boldsymbol{\sigma}(\boldsymbol{\eta}), \kappa_{k-1}) > 0, \end{cases}$$

$$\bar{\beta}(\boldsymbol{\eta}) = \sqrt{\frac{3}{2}}\gamma\|dev(\boldsymbol{\sigma}(\boldsymbol{\eta}))\|_F^{-1}, \ \ \bar{\gamma}(\boldsymbol{\eta}) = \frac{3\mu}{3\mu + H'(\bar{\varepsilon}^p_{k-1} + \gamma)} - \bar{\beta}(\boldsymbol{\eta})$$

Properties of T_k and DT_k ensuring quadratic convergence of the Newton method was investigated in (Blaheta, 1997).

7 Conclusion

- Elastoplastic problem was formulated by the generalized projective mapping.
- Such a formulation is suitable for investigation of the operator properties like semismoothness, differentiability or potentiality.
- It is possible to consider some generalization like non-linear elastic law, perfect plasticity, other hardening law or non-associative plasticity.
- Positive definiteness of the tangential operator for plasticity with hardening has not been investigated within this framework yet.

Thank you for your attention.