Nonsmooth multigrid methods for problems in mechanics

Oliver Sander

joint work with Carsten Gräser (FU Berlin)

TU Dresden, 29. 2. 2021

Content

Kinematics

- Reference domain Ω
- Deformation function $x \mapsto \boldsymbol{\varphi}(x) = x + \mathbf{u}(x)$

Strains

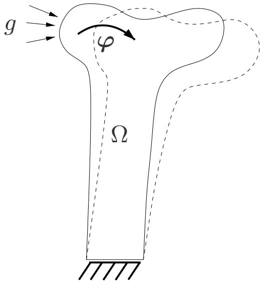
- Measure local change of shape
- Linear theory:

$$\varepsilon(\mathbf{u}) := \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^T).$$

3×3 tensor (matrix)

Stresses

- Field of symmetric 3×3 tensors $\boldsymbol{\sigma}$



Material laws

- In elastic materials: stress is function of strain ε (and nothing else)
- Linear theory: Hooke's law $\boldsymbol{\sigma}(\mathbf{u}) = C : \boldsymbol{\varepsilon}(\mathbf{u})$
- Fourth-order tensor C describes material (Hooke tensor)
- Simplest case: St. Venant–Kirchhoff material

$$\boldsymbol{\sigma} = \lambda \operatorname{tr}(\boldsymbol{\varepsilon}) I + 2\mu\boldsymbol{\varepsilon}.$$

- Lamé constants λ, μ

Boundary value problem

- Static equilibrium

$$-\operatorname{div} \sigma(\mathbf{u}) = f \qquad \text{in } \Omega.$$

- Boundary conditions

$$\begin{aligned} \mathbf{u} &= \mathbf{u}_D && \text{on } \Gamma_D \\ \boldsymbol{\sigma}(\mathbf{u})\mathbf{n} &= g && \text{on } \Gamma_N. \end{aligned}$$

Weak formulation

- Vector-valued Sobolev spaces $\mathbf{H}^1_0(\Omega) := H^1_0(\Omega, \mathbb{R}^3)$, $\mathbf{H}^1_D(\Omega)$
- The usual trick with Green's formula: find $\mathbf{u} \in \mathbf{H}^1_D(\Omega)$ such that

$$a(\mathbf{u}, \mathbf{v}) = l(\mathbf{v}) \qquad \forall \mathbf{v} \in \mathbf{H}^1_0(\Omega)$$

  with

$$a(\mathbf{v}, \mathbf{w}) := \int_\Omega \boldsymbol{\sigma}(\mathbf{v}) : \boldsymbol{\varepsilon}(\mathbf{w})\, dx \qquad l(\mathbf{v}) := \int_\Omega f\mathbf{v}\, dx + \int_{\Gamma_N} g\mathbf{v}\, dx$$

- Symmetric, continuous and $\mathbf{H}^1_0$-elliptic: there exists a unique solution (Lax–Milgram)

Finite elements

- Restrict to vector-valued finite element space $\mathbf{V}_{h,0} \subset \mathbf{H}^1_0$, $\mathbf{V}_{h,D} \subset \mathbf{H}^1_D$
- Vector-valued nodal basis ϕ^j_i

Linear system of equations

- Sparse, symmetric, positive definite linear system

$$Ax = b,$$

- A is $n \times n$ block matrix with 3×3 blocks

$$(A_{ij})_{kl} = \int_\Omega \boldsymbol{\sigma}(\phi_i^k) : \boldsymbol{\varepsilon}(\phi_j^l)\, dx$$
$$(b_i)_j = \int_\Omega f\phi_i^j\, dx + \int_{\Gamma_N} g\phi_i^j\, dx.$$

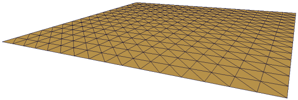
Solvers?

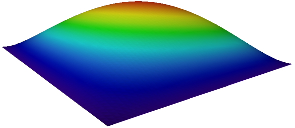
- Direct solvers: good, but need a lot of memory
- Iterative solvers: convergence rates degenerate with decreasing mesh size

Multigrid to the rescue

For illustration: the simpler Poisson problem

$$\begin{aligned} -\Delta u &= 1 \qquad \text{on } \Omega := (0,1)^2, \\ u &= 0 \qquad \text{on } \partial\Omega \end{aligned}$$





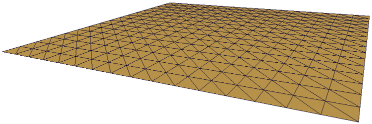
The Gauß–Seidel method

- Multigrid is based on the Gauß–Seidel method

Gauß–Seidel method

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}}\left(b_i - \sum_{j=1}^{i-1} A_{ij}x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij}x_j^{k}\right)$$

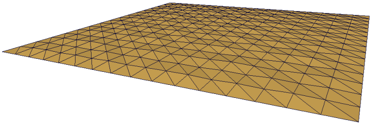


iterate 0

Gauß–Seidel method

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}}\left(b_i - \sum_{j=1}^{i-1} A_{ij}x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij}x_j^k\right)$$

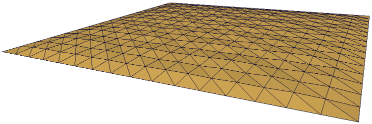


iterate 1

Gauß–Seidel method

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}}\left(b_i - \sum_{j=1}^{i-1} A_{ij}x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij}x_j^{k}\right)$$

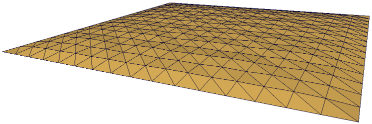


iterate 2

Gauß–Seidel method

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}} \left(b_i - \sum_{j=1}^{i-1} A_{ij} x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij} x_j^{k} \right)$$



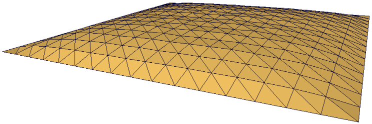
iterate 3

Gauß–Seidel method

Gauß–Seidel method

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}} \left(b_i - \sum_{j=1}^{i-1} A_{ij} x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij} x_j^k \right)$$



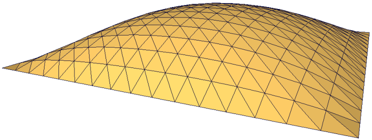
iterate 10

Gauß–Seidel method

Gauß–Seidel method

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}}\left(b_i - \sum_{j=1}^{i-1} A_{ij}x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij}x_j^k\right)$$

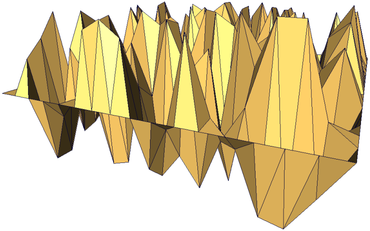


iterate 50

Works, but is painfully slow!

Important observation I:

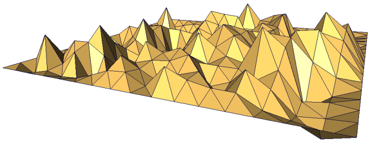
- Gauß–Seidel smoothes!



iterate x^0 (noise)

Important observation I:

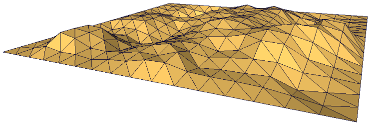
- Gauß–Seidel smoothes!



iterate x^1

Important observation I:

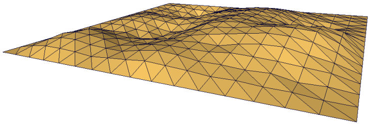
- Gauß–Seidel smoothes!



iterate x^2

Important observation I:

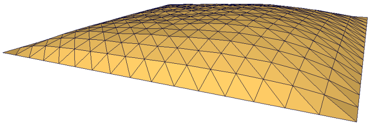
- Gauß–Seidel smoothes!



iterate x^3

Important observation I:

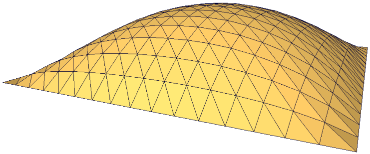
- Gauß–Seidel smoothes!



iterate x^{10}

Important observation I:

- Gauß–Seidel smoothes!



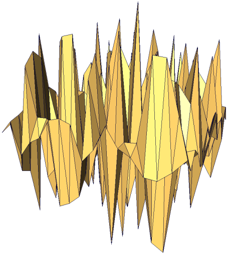
iterate x^{50}

Gauß–Seidel smoothes

- In fact, Gauß–Seidel smoothes the *residual* $r^k := b - Ax^k$
- Consider the error equation

$$A(x^k + e) = b \qquad \Rightarrow \qquad Ae = b - Ax^k.$$

- Residual is smooth after a few (e.g. 3) iterations.



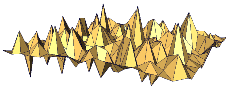
residual r^0 (with x^0 = noise)

Gauß–Seidel smoothes

- In fact, Gauß–Seidel smoothes the *residual* $r^k := b - Ax^k$
- Consider the error equation

$$A(x^k + e) = b \qquad \Rightarrow \qquad Ae = b - Ax^k.$$

- Residual is smooth after a few (e.g. 3) iterations.



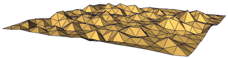
residual r^1

Gauß–Seidel smoothes

- In fact, Gauß–Seidel smoothes the *residual* $r^k := b - Ax^k$
- Consider the error equation

$$A(x^k + e) = b \qquad \Rightarrow \qquad Ae = b - Ax^k.$$

- Residual is smooth after a few (e.g. 3) iterations.



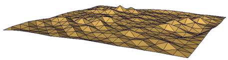
residual r^2

Gauß–Seidel smoothes

- In fact, Gauß–Seidel smoothes the *residual* $r^k := b - Ax^k$
- Consider the error equation

$$A(x^k + e) = b \qquad \Rightarrow \qquad Ae = b - Ax^k.$$

- Residual is smooth after a few (e.g. 3) iterations.



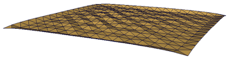
residual r^3

Gauß–Seidel smoothes

- In fact, Gauß–Seidel smoothes the *residual* $r^k := b - Ax^k$
- Consider the error equation

$$A(x^k + e) = b \qquad \Rightarrow \qquad Ae = b - Ax^k.$$

- Residual is smooth after a few (e.g. 3) iterations.



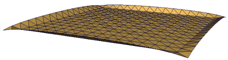
residual r^{10}

Gauß–Seidel smoothes

- In fact, Gauß–Seidel smoothes the *residual* $r^k := b - Ax^k$
- Consider the error equation

$$A(x^k + e) = b \qquad \Rightarrow \qquad Ae = b - Ax^k.$$

- Residual is smooth after a few (e.g. 3) iterations.



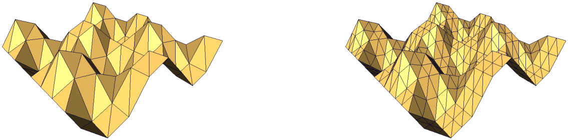
residual r^{50}

Observation II:

- r^k is smooth, use coarser grid to solve $Ae = r^k$

- Define prolongation operator

$$P : V_h^{\text{coarse}} \to V_h^{\text{fine}}$$



- Error equation for the coarse grid

$$\hat{A} e_c = \hat{r}^k \qquad \text{with} \qquad \hat{A} = P^T A P \qquad \text{and} \qquad \hat{r}^k = P^T r^k .$$

Let x^0 be an initial iterate

- Compute $\hat{A} = P^T A P$
- For $k = 1, 2, 3, \ldots$ do
  - **Smooth:** ν steps of Gauß–Seidel (usually $\nu = 3$) to obtain x_*^k
  - **Restrict:** compute $\hat{r}^k = P^T r^k = P^T(b - Ax_*^k)$
  - **Coarse correction:** solve $\hat{A} e_c = \hat{r}^k$
  - **Prolong and add:** $x^{k+1} = x^k + P e_c$

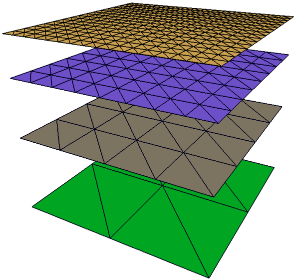
How do we solve $\hat{A}e_c = \hat{r}^k$?

Small $\hat{A}$

- Direct solver

Large $\hat{A}$

- Multigrid!
- We don't have to solve $\hat{A}e_c = \hat{r}^k$ exactly!
- Recursively do one multigrid iteration for $\hat{A}e_c = \hat{r}^k$



Multigrid is fast!

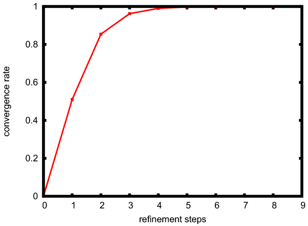
Theorem ([Hackbusch, Xu, Yserentant])

For a multigrid cycle with ν smoothing steps and l grid levels we have

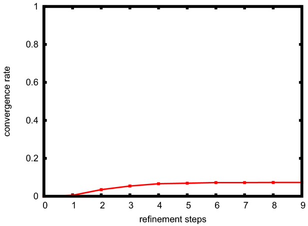
$$\|x^{k+1} - x\| \leq \rho_l \|x^k - x\|$$

$$\rho_l \leq \rho_\infty := \left(\frac{c}{c + 2\nu}\right)^{1/2}$$

with c a constant independent of l and ν.



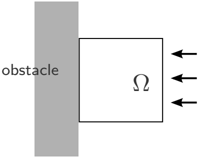
Gauß–Seidel



Multigrid

One-body contact problems

- Model contact with rigid obstacle



- Contact boundary Γ_C
- Define normal and tangential displacement

$$\mathbf{u}_n(x) := \mathbf{u}(x)\mathbf{n}(x) \qquad \text{and} \qquad \mathbf{u}_T(x) := \mathbf{u}(x) - \mathbf{u}_n \cdot \mathbf{n}(x).$$

($\mathbf{n}$: the unit outer normal to Ω)

- Normal stress $\boldsymbol{\sigma}\mathbf{n} \in \mathbb{R}^d$, and its normal component

$$\sigma_n := \mathbf{n}^T \boldsymbol{\sigma}\mathbf{n} \in \mathbb{R}.$$

Contact conditions

- Conditions for the normal displacement/stress

$$\mathbf{u}_n \leq 0, \qquad \sigma_n \geq 0, \qquad \mathbf{u}_n \sigma_n = 0 \qquad \text{on } \Gamma_C.$$

"complementarity conditions"

Tangential components

- Boundary conditions for $\mathbf{u}_T$, $\boldsymbol{\sigma}_T$
- Describe friction effects
- Simplest case: no friction

$$\boldsymbol{\sigma}_T = 0 \qquad \text{on } \Gamma_C.$$

Variational inequality of the first kind

- Find $\mathbf{u} \in \mathcal{K}$ such that

$$a(\mathbf{u}, \mathbf{v} - \mathbf{u}) \geq l(\mathbf{v} - \mathbf{u}) \qquad \forall \mathbf{v} \in \mathcal{K}$$

  with

$$\mathcal{K} := \{\mathbf{v} \in \mathbf{H}^1_D(\Omega) \mid \mathbf{v}_n \leq 0 \text{ a.e.}\}$$

- $\mathcal{K}$ closed and convex: There exists a unique solution

Finite elements

- Replace $\mathbf{H}^1_D$ by finite element subspace $V_{h,D}$.
- Find $\mathbf{u}_h \in \mathcal{K}_h$ such that

$$a(\mathbf{u}_h, \mathbf{v}_h - \mathbf{u}_h) \geq l(\mathbf{v}_h - \mathbf{u}_h) \qquad \forall \mathbf{v}_h \in \mathcal{K}_h$$

- $\mathcal{K}_h$: suitable approximation of the admissible set $\mathcal{K}$

Minimization problem

- The variational inequality is equivalent to minimizing

$$J(\mathbf{v}) = \frac{1}{2}a(\mathbf{v},\mathbf{v}) - l(\mathbf{v}) + \chi_{\mathcal{K}}(\mathbf{v})$$

  in $\mathbf{H}^1_D(\Omega)$
- $\chi_{\mathcal{K}}$ is the indicator functional

$$\chi_{\mathcal{K}}(\mathbf{v}) := \begin{cases} 0 & \text{if } \mathbf{v} \in \mathcal{K}, \\ \infty & \text{otherwise.} \end{cases}$$

- The functional J is strictly convex, coercive, and lower semicontinuous. Therefore it has a unique minimizer on $\mathbf{H}^1_D$.

Finite Elements

- Minimize

$$J(\mathbf{v}_h) = \frac{1}{2}a(\mathbf{v}_h,\mathbf{v}_h) - l(\mathbf{v}_h) + \chi_{\mathcal{K}_h}(\mathbf{v}_h),$$

  in FE space $\mathbf{V}_h$.

Algebraic minimization problem

Find a minimizer $x \in (\mathbb{R}^d)^n$ of

$$J(x) = \frac{1}{2} x^T A x + \chi_{\bar{K}}(x),$$

where

$$A \in (\mathbb{R}^{d \times d})^{n \times n}, \qquad (A_{ij})_{kl} = \int_\Omega \boldsymbol{\sigma}(\phi_{i,k}) : \varepsilon(\phi_{j,l}) \, dx$$

is the stiffness matrix of the linear elasticity problem.

Algebraic admissible set

In suitable coordinates the algebraic admissible set $\bar{K}$ has the form

$$\bar{K} = \prod_{i=1}^{dn} (-\infty, a_i), \qquad \text{with } a_i \in \mathbb{R} \cup \{\infty\} \text{ for all } 1 \leq i \leq dn$$

Multigrid revisited

- Let $A \in \mathbb{R}^{n \times n}$ be symmetric and positiv definite, and $b \in \mathbb{R}^n$.
- Let's look at the linear multigrid method again!

Minimization view

- Instead of thinking about how to solve

$$Ax = b$$

- Think about how to minimize

$$J(x) = \frac{1}{2} x^T A x - bx$$

- ... or even directly

$$J(\mathbf{v}_h) = \frac{1}{2} a(\mathbf{v}_h, \mathbf{v}_h) - l(\mathbf{v}_h)$$

Gauß–Seidel as we know it: One iteration of Gauß–Seidel is:

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}}\left(b_i - \sum_{j=1}^{i-1} A_{ij}x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij}x_j^k\right)$$

Equivalent minimization formulation:

- given an $x^k \in \mathbb{R}^n$

1. Set $w^0 = x^k$
2. For each line i of A do

$$w^i = \arg\min_{\alpha\in\mathbb{R}} J(w^{i-1} + \alpha e_i)$$

3. Set $x^{k+1} = w^n$

Gauß–Seidel as a minimization algorithm

Gauß–Seidel as we know it: One iteration of Gauß–Seidel is:

- given an $x^k \in \mathbb{R}^n$
- for each line i of A do

$$x_i^{k+1} = \frac{1}{A_{ii}}\left(b_i - \sum_{j=1}^{i-1} A_{ij}x_j^{k+1} - \sum_{j=1+1}^{n} A_{ij}x_j^k\right)$$

Gauß–Seidel for a contact problem:

- given an $x^k \in \mathbb{R}^n$

1. Set $w^0 = x^k$
2. For each line i of A do

$$w^i = \underset{w_i^{i-1}+\alpha \leq a_i}{\arg\min}\ J(w^{i-1} + \alpha e_i)$$

3. Set $x^{k+1} = w^n$

Nonlinear Gauß–Seidel

- Minimization view frees us from linearity assumption
- Even frees us from differentiability assumption!
- Let H be the set of all points where J is not differentiable.

Theorem ([Glowinski])

Let J be strictly convex, coercive, and lower semicontinuous.
Let the admissible set $\bar{K}$ be the tensor product of closed intervals.
Then the nonlinear Gauß–Seidel method converges for any initial iterate.

- $+$ globally convergent!
- $+$ solve contact problems without penalty parameters!
- - very slow

What about multigrid?

Ideas:

- Do nonlinear Gauß–Seidel on all grid levels
  $\longrightarrow$ Checking for admissibility is too expensive!
- Construction of admissible coarse grid spaces [Tai]:
  - A priori construction
- Monotone multigrid [Kornhuber]:

  Construct situation-dependent coarse grid obstacles
  - Provably convergent
  - Fast convergence
  - Challenging implementation

Gauß–Seidel for a contact problem:

- given an $x^k \in \mathbb{R}^n$

1. Set $w^0 = x^k$
2. For each line i of A do
$$w^i = \operatorname*{arg\,min}_{w_i^{i-1}+\alpha \leq a_i} J(w^{i-1} + \alpha e_i)$$
3. Set $x^{k+1} = w^n$

Generalization:
Replace minimization problem in 2. by
$$w^i = \operatorname*{arg\,min}_{\alpha \in \mathbb{R}} J(w^{i-1} + \alpha e_i) + \chi_{(-\infty, a_i - w_i^{i-1}]}(\alpha)$$
with χ again the indicator functional.

Let $J : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be the objective functional of a minimization problem.

We call J *block-separably nonsmooth* if it has the form

$$J(x) = J_0(x) + \sum_{i=1}^{M} \varphi_i(x_i),$$

where

- $J_0 : \mathbb{R}^n \to \mathbb{R}$ is coercive and continuously differentiable,
- there is a decomposition $\mathbb{R}^n$ with $\prod_{i=1}^{M} \mathbb{R}^{n_i}$, with $\sum_{i=1}^{M} n_i = n$,
- the functionals $\varphi_i : \mathbb{R}^{n_i} \to \mathbb{R} \cup \{\infty\}$, $i = 1, \dots, M$ are convex, proper, lower semi-continuous, and continuous on their domains.

Example: contact problem

$$J_0(x) = \frac{1}{2} x^T A x - b^T x, \qquad \varphi_i(x_i) = \chi_{(-\infty, a_i - w_i^{i-1}]}(x_i)$$

The algorithm:

1. Nonlinear presmoothing (Gauß–Seidel)
   - For each block i, solve a local minimization problem for

     $$J_i(v) := J_0(v) + \varphi_i(v)$$

     in the ith block.

The algorithm:

1. Nonlinear presmoothing (Gauß–Seidel)
   - For each block i, solve a local minimization problem for

     $$J_i(v) := J_0(v) + \varphi_i(v)$$

     in the ith block.
2. Truncated linearization
   - Freeze all variables where the φ_i are not differentiable.
   - Linearize everywhere else

The algorithm:

1. Nonlinear presmoothing (Gauß–Seidel)
   - For each block i, solve a local minimization problem for
   $$J_i(v) := J_0(v) + \varphi_i(v)$$
   in the ith block.
2. Truncated linearization
   - Freeze all variables where the φ_i are not differentiable.
   - Linearize everywhere else
3. Linear correction
   - E.g., one linear multigrid step,
   - or solve lineared problem exactly.

The algorithm:

1. Nonlinear presmoothing (Gauß–Seidel)
   - For each block i, solve a local minimization problem for

     $$J_i(v) := J_0(v) + \varphi_i(v)$$

     in the ith block.
2. Truncated linearization
   - Freeze all variables where the φ_i are not differentiable.
   - Linearize everywhere else
3. Linear correction
   - E.g., one linear multigrid step,
   - or solve lineared problem exactly.
4. Projection onto admissible set
   - Simply in the ℓ^2-sense / block-wise

The algorithm:

1. Nonlinear presmoothing (Gauß–Seidel)
   - For each block i, solve a local minimization problem for
     $$J_i(v) := J_0(v) + \varphi_i(v)$$
     in the ith block.
2. Truncated linearization
   - Freeze all variables where the φ_i are not differentiable.
   - Linearize everywhere else
3. Linear correction
   - E.g., one linear multigrid step,
   - or solve lineared problem exactly.
4. Projection onto admissible set
   - Simply in the ℓ^2-sense / block-wise
5. Line search
   - 1d nonsmooth minimization problem: use bisection

Introduce (inexact) local minimization operators

$$\mathcal{M}_i(\cdot) = \underset{v \in (\cdot) + e_i \otimes \mathbb{R}^{n_i}}{\arg\min} J(v)$$

Theorem (Gräser, S. 2017)

Let $v^0 \in \operatorname{dom} J$ and assume that the inexact solution local solution operators $\mathcal{M}_i$ satisfy:

- *Monotonicity: $J(\mathcal{M}_i(w)) \leq J(w)$ for all $w \in \operatorname{dom} J$.*
- *Continuity: $J \circ \mathcal{M}_i$ is continuous.*
- *Stability: $J(\mathcal{M}_i(w)) < J(w)$ if $J(w)$ is not minimal in the i-th block.*

Then the iterates produced by the TNNMG method converge to a stationary point of J.

Introduce (inexact) local minimization operators

$$\mathcal{M}_i(\cdot) = \operatorname*{arg\,min}_{v \in (\cdot) + e_i \otimes \mathbb{R}^{n_i}} J(v)$$

Theorem (Gräser, S. 2017)

Let $v^0 \in \operatorname{dom} J$ and assume that the inexact solution local solution operators $\mathcal{M}_i$ satisfy:

- *Monotonicity: $J(\mathcal{M}_i(w)) \leq J(w)$ for all $w \in \operatorname{dom} J$.*
- *Continuity: $J \circ \mathcal{M}_i$ is continuous.*
- *Stability: $J(\mathcal{M}_i(w)) < J(w)$ if $J(w)$ is not minimal in the i-th block.*

Then the iterates produced by the TNNMG method converge to a stationary point of J.

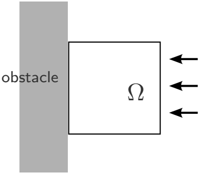
Corollary (Gräser, S. 2017)

If J is strictly convex and coercive, then the TNNMG method converges to the unique minimizer of J.

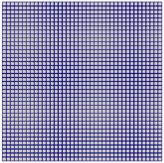
Geometry

- Domain: $\Omega = [0,1]^2$
- Obstacle: negative half space

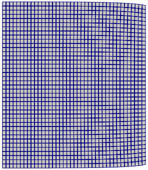
$$H = \{x \in \mathbb{R}^d \; : \; x_0 < 0\}.$$



setting

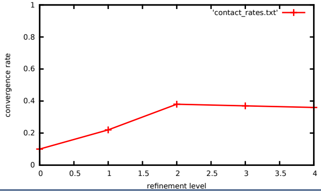


grid



deformation

Convergence rates



Boundary conditions for the normal component

$$\mathbf{u}_n \leq 0, \qquad \sigma_n \leq 0, \qquad \mathbf{u}_n \sigma_n = 0.$$

Boundary conditions for the tangential components

- Previously: no friction

$$\boldsymbol{\sigma}_T = 0 \qquad \text{on } \Gamma_C.$$

- Tresca friction model

$$\|\boldsymbol{\sigma}_T\| \leq \mu \bar{\sigma}_N.$$

$$\mathbf{u}_T = \lambda \boldsymbol{\sigma}_T, \qquad \text{with } \begin{cases} \lambda = 0 & \text{if } \|\boldsymbol{\sigma}_T\| < \mu \bar{\sigma}_N, \\ \lambda \geq 0 & \text{if } \|\boldsymbol{\sigma}_T\| = \mu \bar{\sigma}_N. \end{cases}$$

- $\mu \in \mathbb{R}$, $\mu > 0$ is called the *coefficient of friction*.
- $\bar{\sigma}_N > 0$ fixed parameter (approximation to the normal stress)
- Simple stick–slip friction

Tresca friction functional

$$j_T(\mathbf{v}) = \int_{\Gamma_C} \mu \bar{\sigma}_N \|\mathbf{v}_T\| \, da, \qquad \forall \mathbf{v} \in \mathbf{H}^1_D.$$

Weak formulation

Variational inequality of the second type: Find $\mathbf{u} \in \mathcal{K} \subset \mathbf{H}^1_D(\Omega)$ such that

$$a(\mathbf{u}, \mathbf{v} - \mathbf{u}) + j_T(\mathbf{v}) - j_T(\mathbf{u}) \geq l(\mathbf{v} - \mathbf{u}) \qquad \forall \mathbf{v} \in \mathcal{K}$$

Minimization formulation

- Find minimizer in $\mathbf{H}^1_D$ of

$$J(\mathbf{v}) = \frac{1}{2} a(\mathbf{v}, \mathbf{v}) - l(\mathbf{v}) + \chi_K(\mathbf{v}) + j_T(\mathbf{v})$$

- Functional is strictly convex, coercive, and lower semicontinuous
- There exists a unique minimizer

Naive approach:

- Find minimizers in $\mathbf{V}_{h,D}$ of

$$J(\mathbf{v}_h) = \frac{1}{2}a(\mathbf{v}_h, \mathbf{v}_h) + \chi_{K_h}(\mathbf{v}_h) + j_T(\mathbf{v}_h),$$

- integral for the nonsmooth term j_T over each element involves the degrees of freedom of all corners of that element
- Structure of the nondifferentiable points not suitable for TNNMG

Modified approach:

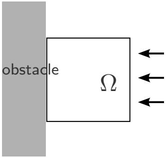
- Find minimizers in $\mathbf{V}_{h,D}$ of

$$J(\mathbf{v}_h) = \frac{1}{2}a(\mathbf{v}_h, \mathbf{v}_h) + \chi_{K_h}(\mathbf{v}_h) + \sum_{i=1}^{n} \mu\bar{\sigma}_N \|(\mathbf{v}(p_i))_T\| \int_{\Gamma_C} |\lambda_i|\, dx.$$

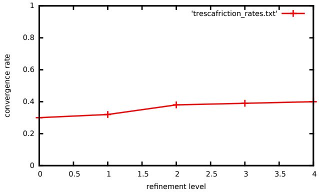
- Interpretation: lumped quadrature rule
- First-order FE: functional still strictly convex, coercive, l.s.c.

Behavior:

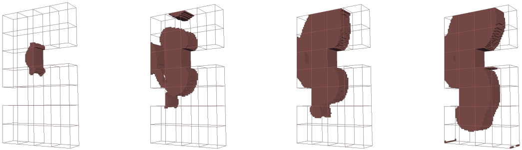
- Functional is strictly convex, coercive, and lower semicontinuous.
- Set of nondifferentiable points has correct block-separable structure
- The TNNMG method converges for all initial iterates
- Convergence rate independent of the grid resolution

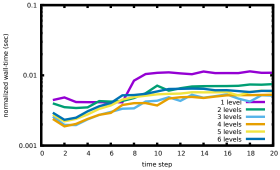


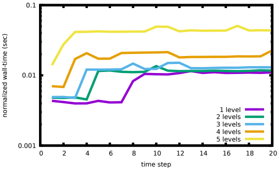
setting



- Primal formulation: displacement u, plastic strain p
- Von Mises and Tresca yield laws (and others)
- Kinematic and isotropic hardening







Normalized wall-time per time step: TNNMG (left) vs. predictor–corrector (right)

- Low-hardening case still problematic

Phase-field model of brittle fracture: [Miehe, Welschinger, Hofacker, 2010]

- Unknowns: displacement $\mathbf{u} : \Omega \to \mathbb{R}^d$, fracture phase field $d : \Omega \to [0,1]$

Variational model:

- Rate potential

$$\Pi(\dot{\mathbf{u}}, \dot{d}) := \dot{E}(\dot{\mathbf{u}}, \dot{d}) + D(d, \dot{d}) - P_{\text{ext}}(\dot{\mathbf{u}}).$$

Dissipation potential

- Regularized crack surface density: $\gamma(d) = \frac{1}{2l}(d^2 + l^2 \|\nabla d\|^2)$
- Dissipation potential

$$D(\dot{d}; d) = \int_{\mathcal{B}} g_c \dot{\gamma}(d, \nabla d) + \chi_{[0,\infty)}(\dot{d})\, dV$$

- $\chi_{[0,\infty)}$ = indicator functional of $[0, \infty)$

Elastic energy degraded in tension:

- Elastic bulk energy density

$$\psi_0(\mathbf{u}) = \frac{\lambda}{2}(\operatorname{tr}\boldsymbol{\varepsilon}(\mathbf{u}))^2 + \mu\operatorname{tr}(\boldsymbol{\varepsilon}(\mathbf{u}))^2$$

- Split into tensile/compressive parts $\psi_0^{\pm}$.
- Strain-based split

$$\psi_0^{\pm}(\boldsymbol{\varepsilon}) := \frac{\lambda}{2}\Big\langle\sum_{i=1}^{d}\mathrm{Eig}(\boldsymbol{\varepsilon})_i\Big\rangle_{\pm}^{2} + \mu\Big[\sum_{i=1}^{d}\langle\mathrm{Eig}(\boldsymbol{\varepsilon})_i\rangle_{\pm}^{2}\Big].$$

- Degraded elastic energy

$$E(\mathbf{u},d) = \int_\Omega ((1-d^2)+k)\psi_0^{+}(\mathbf{u}) + \psi_0^{-}(\mathbf{u})\,dx$$

Time discretization

- Integrate rate potential over time interval $[t_k, t_{k+1})$
- Next iterate $(\mathbf{u}^{k+1}, d^{k+1})$ minimizes

$$\Pi_k^\tau(\mathbf{u}, d) = \int_\Omega \big((1-d)^2 + k\big)\psi_0^+(\mathbf{u}) + \psi_0^-(\mathbf{u}) + g_c\gamma(d, \nabla d) + \chi_{[d^k,1]}(d)\, dV$$

The indicator functional $\chi_{[d^k,1]}(d)$

- No regularization
- No $\mathcal{H}$-field

Discretization

Time discretization

- Integrate rate potential over time interval $[t_k, t_{k+1})$
- Next iterate $(\mathbf{u}^{k+1}, d^{k+1})$ minimizes

$$\Pi_k^\tau(\mathbf{u}, d) = \int_\Omega \big((1-d)^2 + k\big)\psi_0^+(\mathbf{u}) + \psi_0^-(\mathbf{u}) + g_c\gamma(d, \nabla d) + \chi_{[d^k,1]}(d)\, dV$$

The indicator functional $\chi_{[d^k,1]}(d)$

- No regularization
- No $\mathcal{H}$-field

Space discretization

- First-order Lagrange finite elements

Algebraic increment problem

- Nonsmooth, biconvex, coercive, lower-semicontinuous

Block-separable form

Algebraic increment energy

$$\Pi_k^{\text{alg}}(\mathbf{u}, d) = \int_\Omega \big((1-d)^2 + k\big)\psi_0^+(\mathbf{u}) + \psi_0^-(\mathbf{u}) + g_c\gamma(d) + \chi_{[d^k,1]}(d)\,dV$$

is of the form

$$\Pi_k^{\text{alg}}(\mathbf{u}, d) = \underbrace{\sum_{i=1}^{\#\text{ elements}} \gamma_i(B_i v) + \langle Av, v\rangle}_{=:\Pi_0(v)} + \sum_{i=1}^{\#\text{ vertices}} \underbrace{\chi_{[d_i^k,1]}(d_i)}_{=:\varphi_i(d_i)}$$

with

$$B_i v = (\boldsymbol{\varepsilon}(\mathbf{u}), d)\big|_{\text{element } i}$$
$$\gamma_i(\boldsymbol{\varepsilon}(\mathbf{u}), d) = \big((1-d)^2 + k\big)\psi_0^+(\boldsymbol{\varepsilon}(\mathbf{u})) + \psi_0^-(\boldsymbol{\varepsilon}(\mathbf{u}))$$

- Π_0 is coercive, continuously differentiable, and biconvex.
- Indicator functionals $\chi_{[d_i^k,1]}(\cdot)$ are convex, proper, and lower semicontinuous.

Abstract problem setting

Nonsmooth bi-convex minimization problem:

$$u^* \in \mathbb{R}^n : \qquad J(u^*) \leq J(v) \qquad \forall v \in \mathbb{R}^n$$

Block structure:

- m blocks of sizes $n_1, \dots, n_m$

$$\mathbb{R}^n = \bigotimes_{i=1}^{m} \mathbb{R}^{n_i}$$

- Canonical restriction operators $R_i : \mathbb{R}^n \to \mathbb{R}^{n_i}$

Block-separable form:

$$J(v) = J_0(v) + \sum_{i=1}^{n} \varphi_i(R_i v)$$

- Coercive, continuously differentiable functional $J_0 : \mathbb{R}^n \to \mathbb{R}$
- Convex, proper, lower semi-continuous functionals $\varphi_i : \mathbb{R}^{n_i} \to \mathbb{R} \cup \{\infty\}$
- The TNNMG algorithm converges against a solution!

Truncated Nonsmooth Newton Multigrid (TNNMG)

The algorithm:

1. Nonlinear presmoothing (Gauß–Seidel)
   - For each block i, solve a local minimization problem for
     $$J_i(v) := J_0(v) + \varphi_i(v)$$
     in the ith block.
2. Truncated linearization
   - Freeze all variables where the φ_i are not differentiable.
   - Linearize everywhere else
3. Linear correction
   - E.g., one linear multigrid step,
   - or solve lineared problem exactly.
4. Projection onto admissible set
   - Simply in the ℓ^2-sense / block-wise
5. Line search
   - 1d nonsmooth minimization problem: use bisection

Smoother:

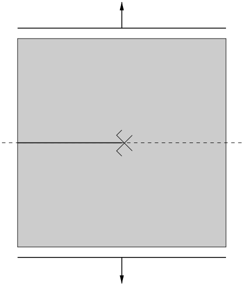
For each vertex:

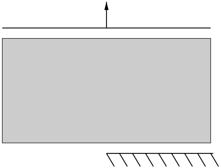
1. Solve displacement minimization problem for this vertex only
    - Only 2 or 3 variables
    - Strictly convex, coercive, C^1
    - Second derivatives exist in a generalized sense
    - Nonsmooth Newton method
2. Solve phase-field problem for this vertex only
    - Scalar coercive strictly convex minimization problem
    - Quadratic $+$ indicator functional

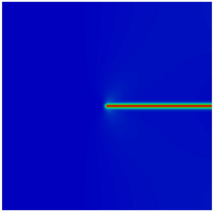
Inexact smoothers:

- Solve local problems inexactly
- Can save lots of wall-time

Numerical example: Notched square







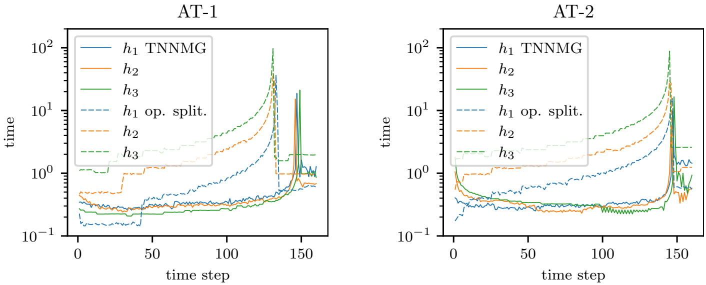
Problem

- Pre-fractured 2d square
- Loaded under pure tension

Solvers

- TNNMG vs. Operator Split with $\mathcal{H}$-field
- Operator split alternates between displacement and damage problem

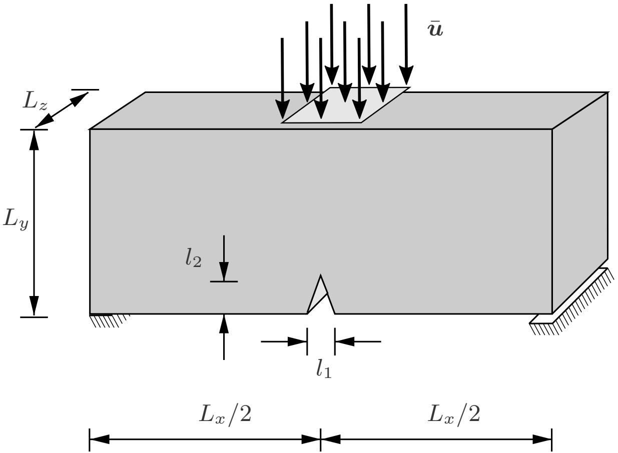
Numerical example: Notched square



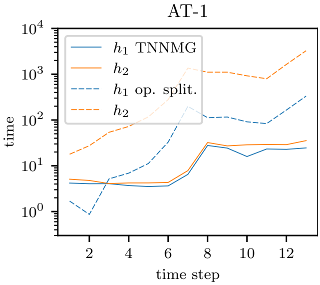
Wall-time per degree of freedom per time step

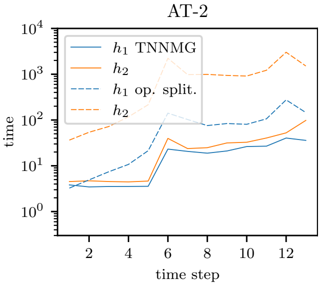
	AT-1			AT-2		
	TNNMGEX	TNNMGPRE	OS	TNNMGEX	TNNMGPRE	OS
h_1	86.85	70.92	155.32	76.82	66.75	186.58
h_2	73.61	63.83	353.63	61.63	54.91	387.06
h_3	74.55	54.98	672.26	80.25	63.27	760.48

Numerical example: Three-Point Bending



Numerical example: Three-Point Bending





Wall-time per degree of freedom per time step

	AT-1			AT-2		
	TNNMGEX	TNNMGPRE	OS	TNNMGEX	TNNMGPRE	OS
h_1	168.47	94.18	1158.98	231.08	112.79	1058.21
h_2	216.32	120.18	10762.81	366.24	195.85	12309.42

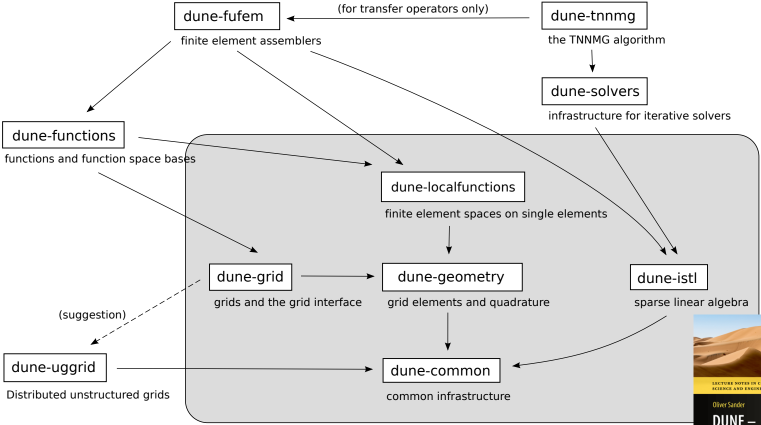
TNNMG implementation challenges:

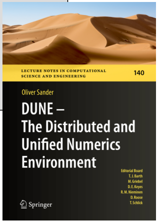
- Nonlinear smoother
- Gradients and tangent matrices
- Line search
- Direct sparse solver for linear correction problems
- For non-quadratic smooth parts: Caching of shape function values

Multigrid steps:

- Hierarchy of finite element grids
- Assembly of prolongation/restriction operators

Support in DUNE





Further Reading

C. Gräser, R. Kornhuber.
Multigrid Methods for Obstacle Problems.
Journal of Computational Mathematics, 27(1):1–44, 2009.

C. Gräser, O. Sander.
Truncated Nonsmooth Newton Multigrid Methods for Block-Separable Minimization Problems.
IMA Journal of Numerical Analysis, 39(1):454–481, 2018.

O. Sander, P. Jaap.
Solving primal plasticity increment problems in the time of a single predictor-corrector iteration.
Computational Mechanics, 65:663–685, 2020.

C. Gräser, D. Kienle, O. Sander.
Truncated Nonsmooth Newton Multigrid for phase-field brittle-fracture problems.
arXiv, 2007.12290, 2020. (complete article on request)

O. Sander.
DUNE — The Distributed and Unified Numerics Environment
Springer, 2020.