Technische Universität München

Resilience for Multigrid Software at the Extreme Scale

Markus Huber

joint work with: Björn Gmeiner, Lorenz John, Ulrich Rüde, Barbara Wohlmuth

huber@ma.tum.de

Technische Universität München, Germany

Januar 25-27, 2016

Gauss Centre for Supercomputing

SPPEXA Symposium 2016

Overview

- **Terraneo: An Exa-scale Mantle Convection Framework**
  - Model problem
  - Ultra scalability
- **Building a Fault Tolerant Multigrid Solver**
  - Challenges in exa-scale systems
  - Problem setting
  - Recovery strategies
  - Single fault scenarios
  - Multiple faults scenarios
- **Towards Geophyiscal Applications**

Terraneo

An Exa-scale Mantle Convection Framework

Stokes equations and equal order discretization

Let $\Omega \subset \mathbb{R}^3$ with $\Gamma = \partial\Omega$

$$\begin{aligned} -\nu\Delta \mathbf{u} + \nabla p &= \mathbf{f} && \text{in } \Omega, \\ \operatorname{div} \mathbf{u} &= 0 && \text{in } \Omega, \\ \mathbf{u} &= \mathbf{0} && \text{on } \Gamma. \end{aligned}$$

Equal order discretization $(\mathbf{P}_1\text{–}P_1)$ [Hughes 1986] [Brezzi, Douglas 1988]

Find $(\mathbf{u}_h, p_h) \in \mathbf{V}_h \times Q_h$ such that

$$\begin{aligned} a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) &= f(\mathbf{v}_h) && \forall\, \mathbf{v}_h \in \mathbf{V}_h, \\ b(\mathbf{u}_h, q_h) - c_h(q_h, p_h) &= g_h(q_h) && \forall\, q_h \in Q_h, \end{aligned}$$

with the level-dependent stabilization terms

$$c_h(q_h, p_h) = \sum_{T \in \mathcal{T}_h} \delta_T\, h_T^2\, \langle \nabla p_h, \nabla q_h \rangle_T \quad \text{and} \quad g_h(q_h) = -\sum_{T \in \mathcal{T}_h} \delta_T\, h_T^2\, \langle \mathbf{f}, \nabla q_h \rangle_T.$$

Numerical simulation to the extreme

- **Uzawa-type multigrid method** [Bank, Welfert, Yserentant 90], [Schöberl, Zulehner 03]
  Apply an inexact Uzawa smoother

$$\mathbf{u}_{k+1} = \mathbf{u}_k + \hat{A}^{-1}(\mathbf{f} - A\mathbf{u}_k - B^\top \mathbf{p}_k), \qquad \mathbf{p}_{k+1} = \mathbf{p}_k + \hat{S}^{-1}(B\mathbf{u}_{k+1} - C\mathbf{p}_k - \mathbf{g})$$

Remark: For convergence we need $\hat{A} \geq A$ and $\hat{S} \geq C + B\hat{A}^{-1}B^\top$

- **Sacalability on a current peta-scale system (JUQUEEN)**

COURTESY: FORSCHUNGSZENTRUM JÜLICH

Nodes	Threads	DoFs	iter	time
5	80	$2.7 \cdot 10^9$	10	617.28
40	640	$2.1 \cdot 10^{10}$	10	703.69
320	5 120	$1.2 \cdot 10^{11}$	10	741.86
2 560	40 960	$1.7 \cdot 10^{12}$	9	720.24
20 480	327 680	$1.1 \cdot 10^{13}$	9	776.09

Mountain Climbing and Faults

Resilience

- **Past:**

  Reliability of systems was a big concern for computing pioneers

  "The problem of building reliable systems out of unreliable components did preoccupy the first generation of computing system designers - see, e.g., Von Neumann, 1956, as first generation computers were very failure prone.", [Capello et al. 2009]

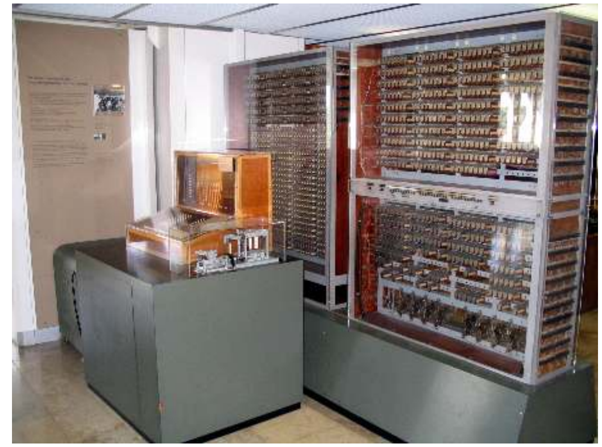
- **Present:** Built-in system level resilience

  Hardware failure is of minor relevance for numerical simulation

- **Future:** Huge number of components in exa-scale

  Algorithmic resilience will be of increasing importance for computational sciences [Dongarra et al. 2015]

  Storage of a vector of size $\mathcal{O}(10^{13})$: 73 TBytes.



COURTESY: FORSCHUNGSZENTRUM JÜLICH

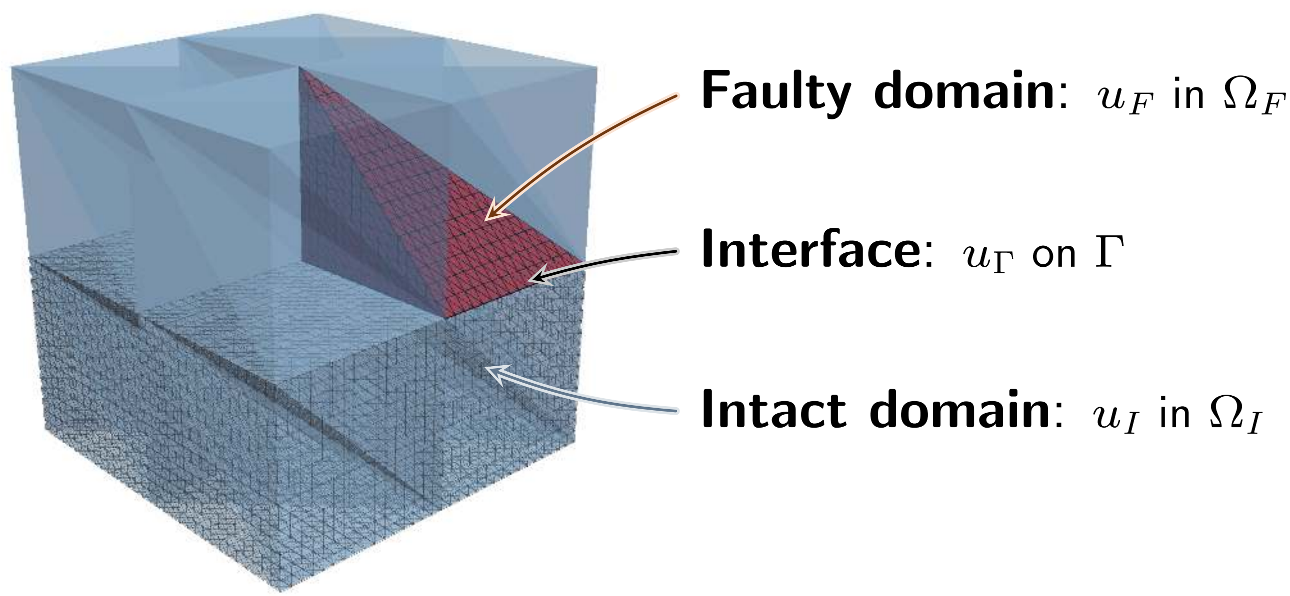
Problem setting and fault model

Model problem:

$$-\Delta u = f \qquad \text{in } \Omega, \qquad + \text{BC}$$

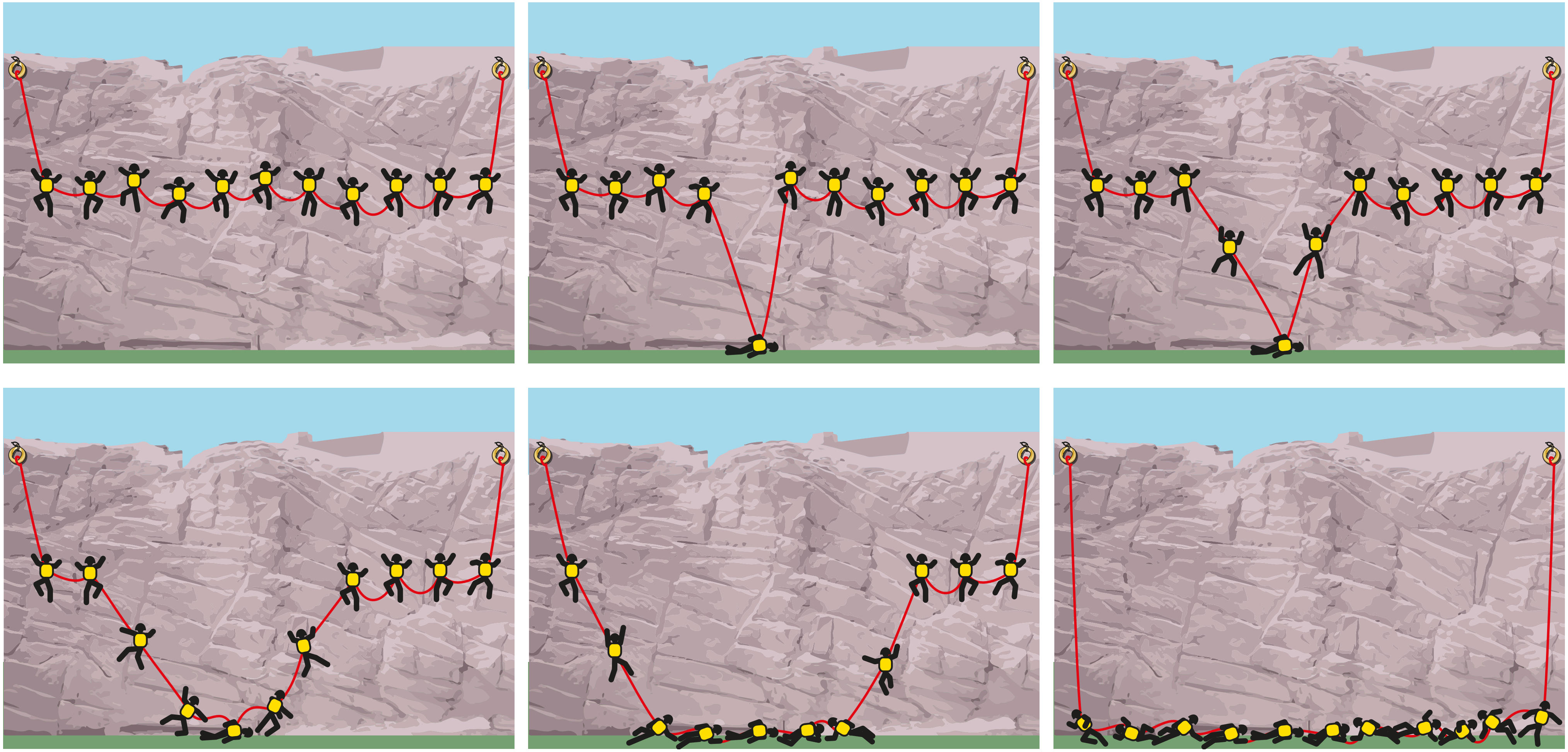
- Discretized by linear FE-method
- Solved by multigrid V-cycles with standard components in the HPC-framework Hierarchical Hybrid Grids [Bergen, Rüde et al. 2002, Gmeiner 2014]

Node crash in the MG:



No fault recovery strategy within a MG

From almost on the top



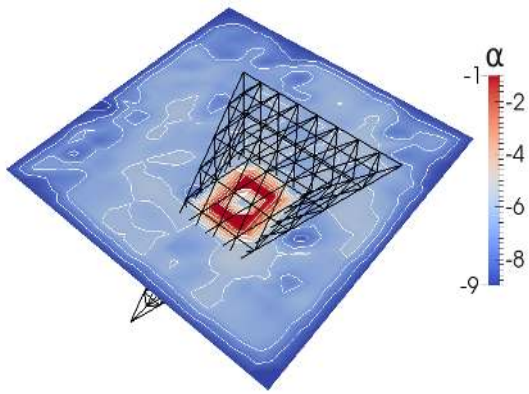
back to the checkpoint level

Comparison of a local recovery strategies

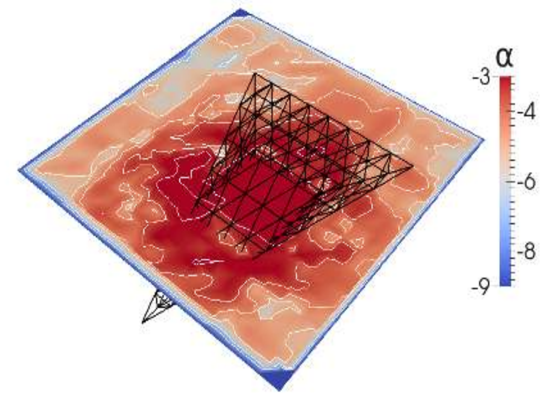
6th iteration

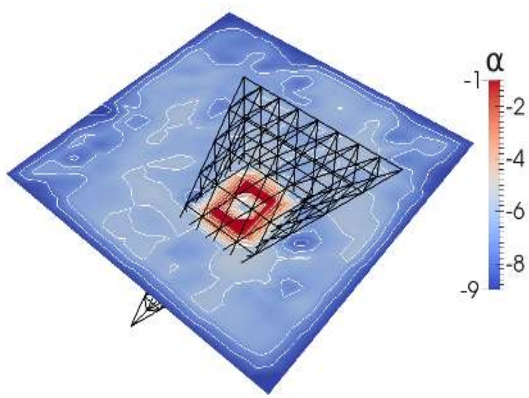
Fault

7th iteration

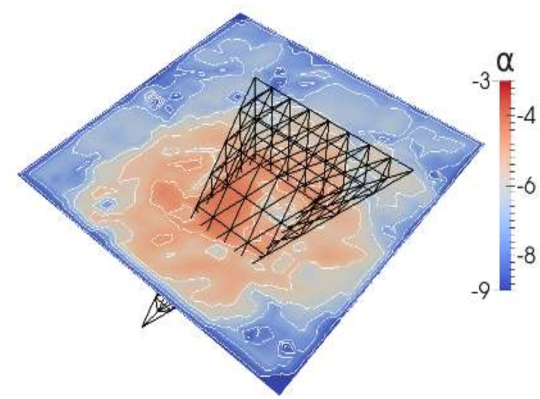


no recovery





local recovery
one F-cycle



$\alpha = \log(||\text{Residual}||)$

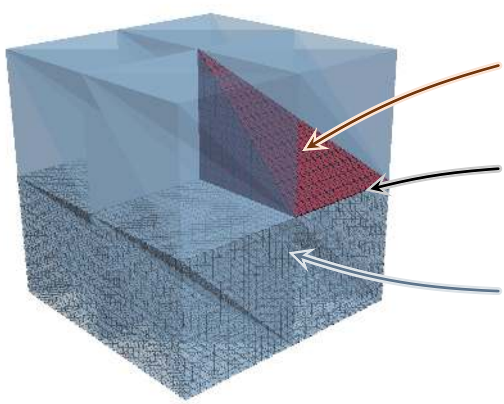
Local recovery strategy

In case of a fault

- Fix interface values u_I on Γ
- Recover faulty values u_F by solving

$$-\Delta u_F = f \qquad \text{in } \Omega_F \qquad \text{with } u_I \text{ Dirichlet BC.}$$

Possiblility for local recovery: smoother, cg-iterations, multigrid cycles, direct solver...



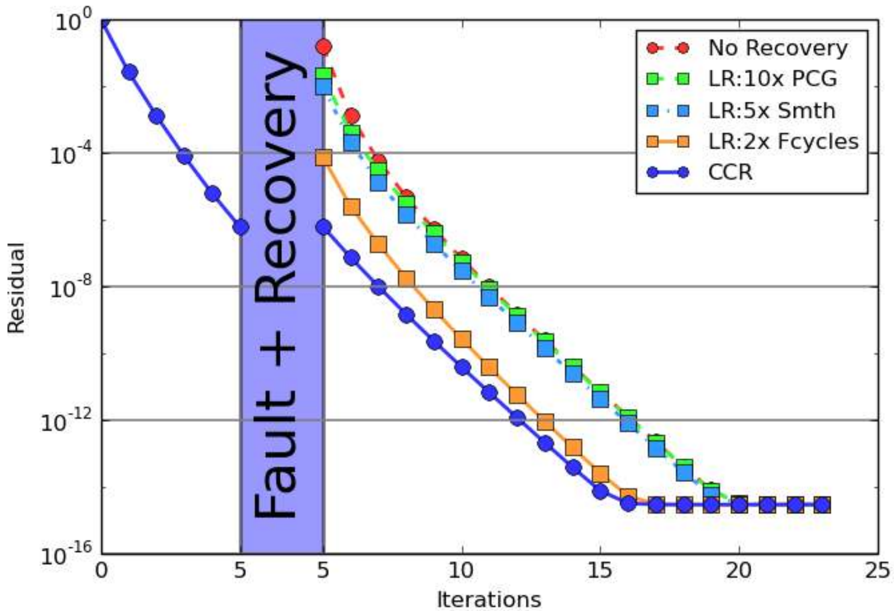
Faulty domain: u_F in Ω_F

Interface: u_Γ on Γ

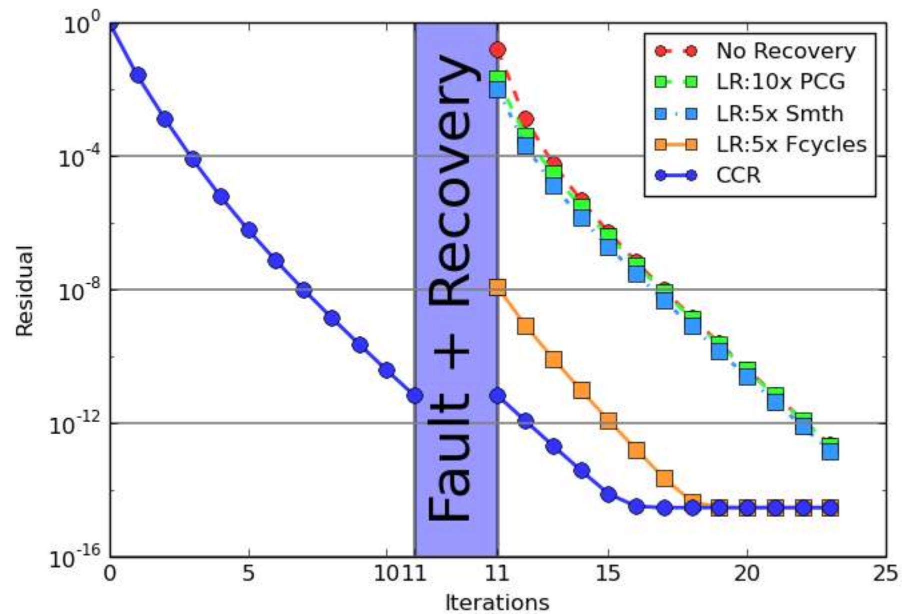
Intact domain: u_I in Ω_I

Numerical results

Fault and local recovery ...



... after 5th iteration
with a perfect superman.



... after 11th iteration
with a perfect superman.

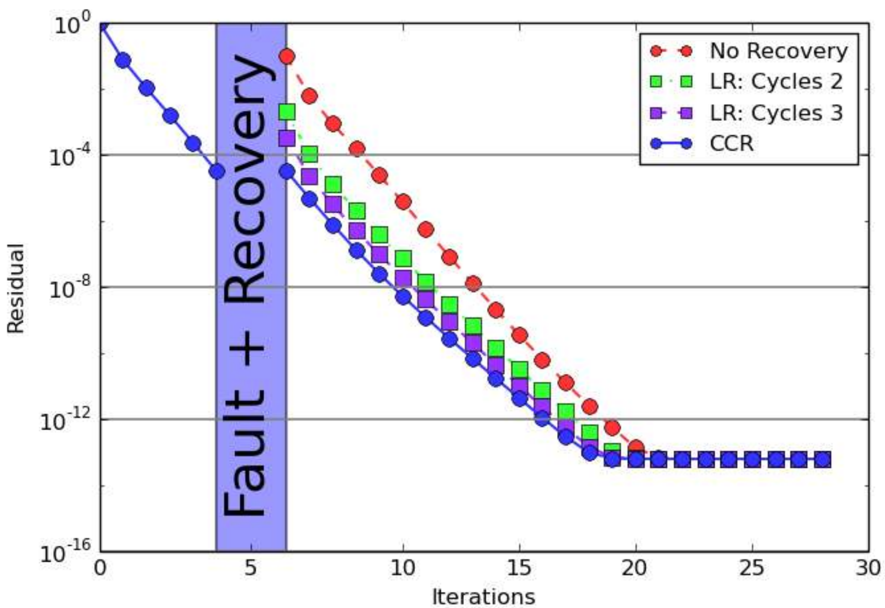
Only MG cycles are efficient.

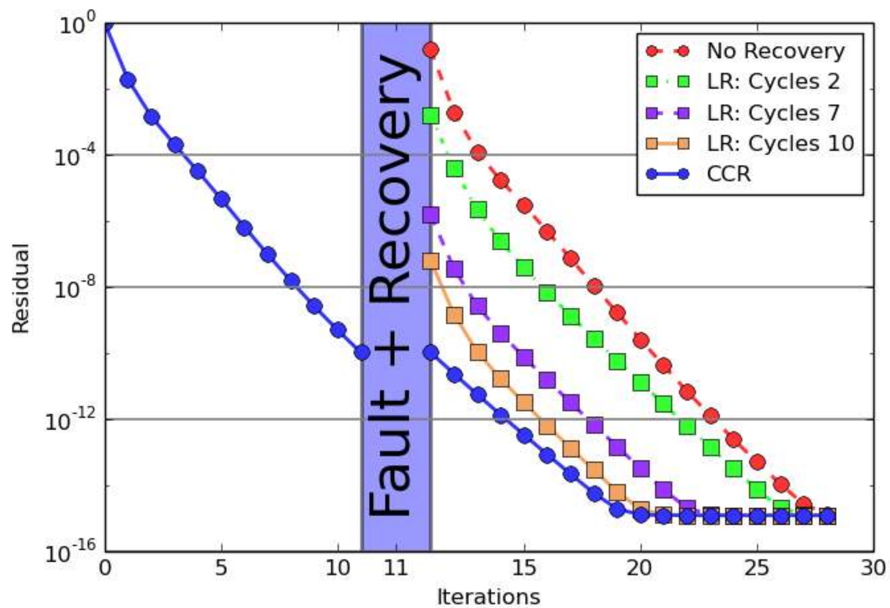
Fault for the Stokes system

Algorithmic strategy:

- Fault in a multigrid algorithm with Uzawa-type smoother
- Freeze velocity and pressure data at the interface
- Locally re-calculated the lost values by superman power

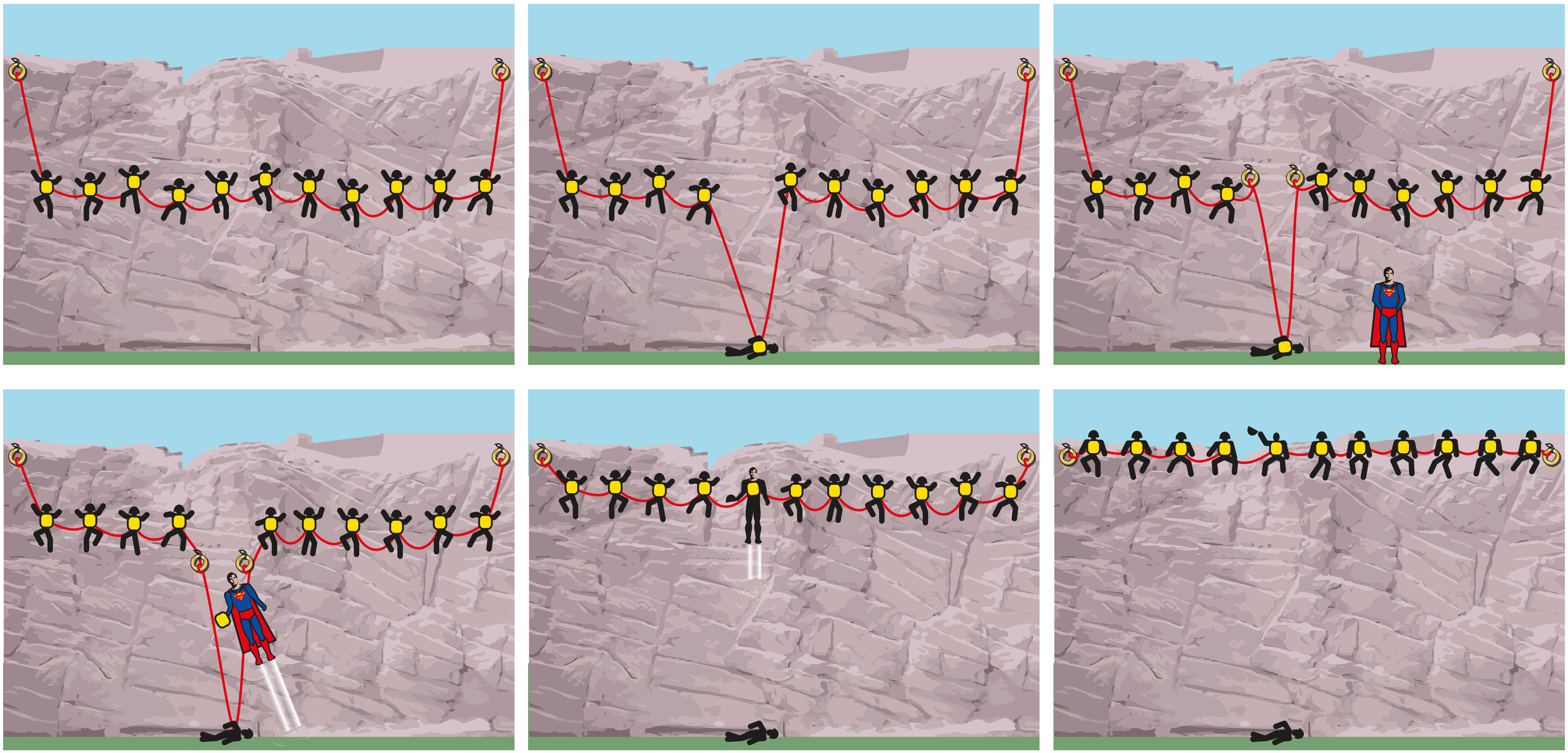
Fault after 5th (left) and 11th (right) iteration step





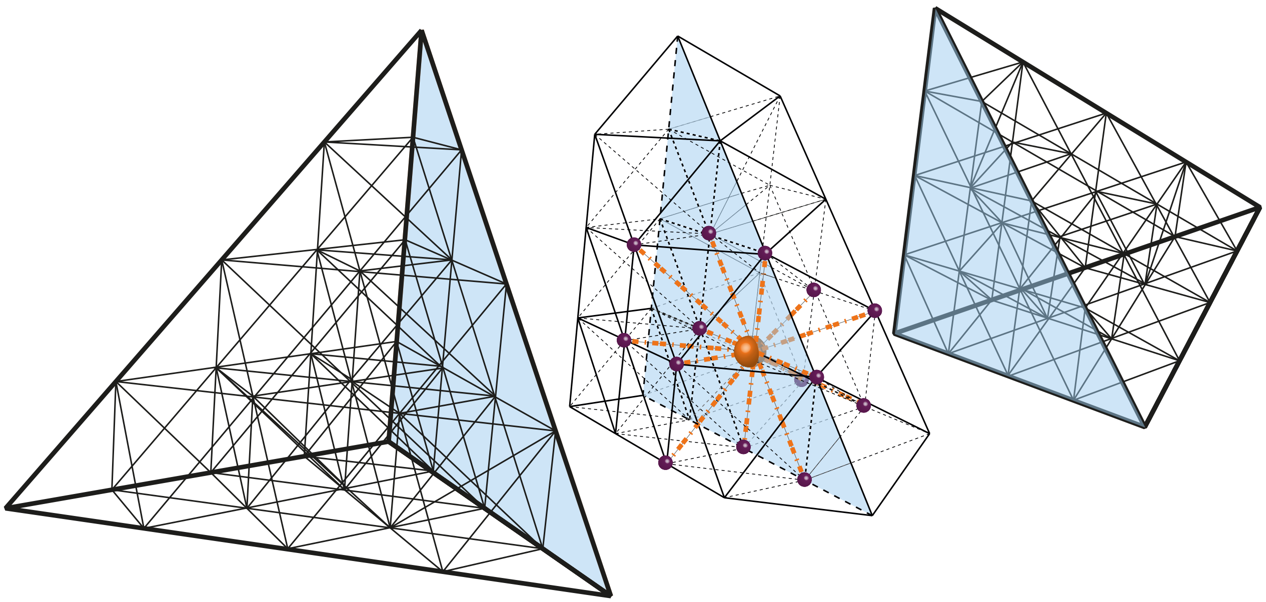
Optimal fault recovery strategy within a MG

From almost on the top



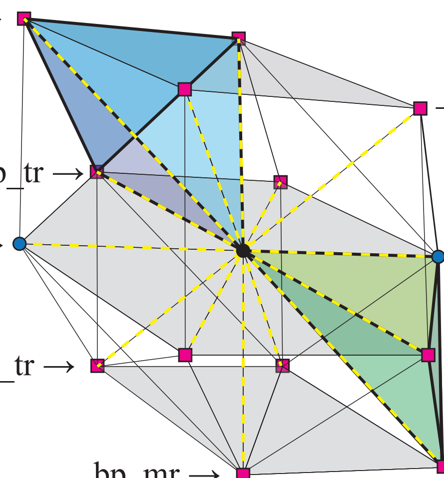
to the top without delay

Data structure for the recovery



Ghost layer primitives

Stencil and **sub-stencil** structure



Global recovery strategies based on tearing concepts

Basic idea: **coupling** via halos on **lower primitives**

- **Dirichlet** (faulty)– **Dirichlet**(healthy) strategy (DD)

$$\begin{pmatrix} A_{II} & A_{I\Gamma_I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathsf{Id} & -\mathsf{Id} & \mathbf{0} & \mathbf{0} \\ A_{\Gamma I} & \mathbf{0} & A_{\Gamma\Gamma} & \mathbf{0} & A_{\Gamma F} \\ \mathbf{0} & \mathbf{0} & -\mathsf{Id} & \mathsf{Id} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & A_{F\Gamma_F} & A_{FF} \end{pmatrix} \begin{pmatrix} u_I \\ u_{\Gamma_I} \\ u_\Gamma \\ u_{\Gamma_F} \\ u_F \end{pmatrix}$$

- **Dirichlet** (faulty)– **Neumann** (healthy) strategy (DN)

$$\begin{pmatrix} A_{II} & A_{I\Gamma_I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ A_{\Gamma_I I} & A_{\Gamma_I\Gamma_I} & \mathsf{Id} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathsf{Id} & \mathbf{0} & -\mathsf{Id} & \mathbf{0} & \mathbf{0} \\ A_{\Gamma I} & \mathbf{0} & \mathbf{0} & A_{\Gamma\Gamma} & \mathbf{0} & A_{\Gamma F} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathsf{Id} & \mathsf{Id} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & A_{F\Gamma_F} & A_{FF} \end{pmatrix} \begin{pmatrix} u_I \\ u_{\Gamma_I} \\ \lambda_{\Gamma_I} \\ u_\Gamma \\ u_{\Gamma_F} \\ u_F \end{pmatrix}$$

Dirichlet-Dirichlet Recovery Strategy

Dirichlet boundary condition on healthy domain

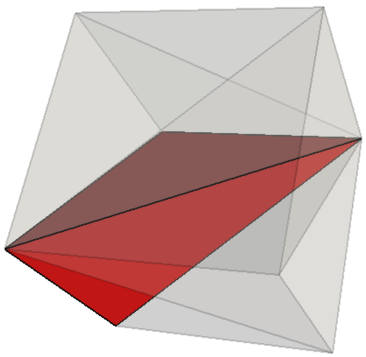
Dirichlet boundary condition on faulty domain

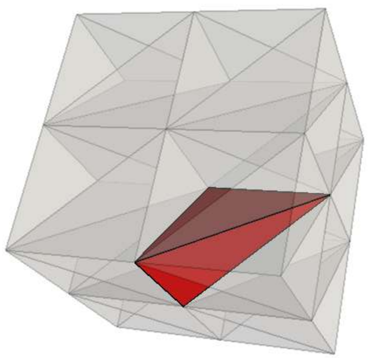
Alg. 1 Dirichlet-Dirichlet recovery

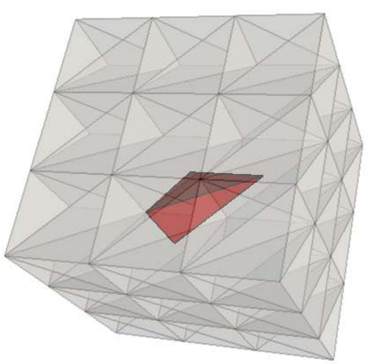
1: Solve $A\underline{u} = \underline{f}$ by multigrid cycles.
2: **if** Fault has occurred **then**
3: **STOP** solving.
4: Recover boundary data $\underline{u}_{\Gamma_F}$ from line 4
5: Initialize $\underline{u}_F$ with zero
6: **In parallel do:**
7: a) Use n_F MG cycles accelerated by η_s to approximate line 5:
8: $A_{FF}\underline{u}_F = \underline{f}_F - A_{F\Gamma_F}\underline{u}_{\Gamma_F}$
9: b) Use n_I MG cycles to approximate line 1
10: $A_{II}\underline{u}_I = \underline{f}_I - A_{I\Gamma_I}\underline{u}_{\Gamma_I}$
11: **RETURN** to line 1 with new values $\underline{u}_I$ in Ω_I and $\underline{u}_F$ in Ω_F.
12: **end if**

$$\begin{pmatrix} A_{II} & A_{I\Gamma_I} & 0 & 0 & 0 \\ 0 & \mathrm{Id} & -\mathrm{Id} & 0 & 0 \\ A_{\Gamma I} & 0 & A_{\Gamma\Gamma} & 0 & A_{\Gamma F} \\ 0 & 0 & -\mathrm{Id} & \mathrm{Id} & 0 \\ 0 & 0 & 0 & A_{F\Gamma_F} & A_{FF} \end{pmatrix} \begin{pmatrix} u_I \\ u_{\Gamma_I} \\ u_\Gamma \\ u_{\Gamma_F} \\ u_F \end{pmatrix} \tag{1}$$

Cycle advantage factor κ







Define $\kappa := \frac{k_{\mathsf{R}}-k}{k_{\mathsf{F}}} \in [0,1]$, k, k_{R} required number of iterations

n_I number of MG cycles on the healthy subdomain

$\eta_{\mathsf{s}} n_I$ number of MG cycles on the faulty subdomain

Fault at $k_{\mathsf{F}} = 5$ and speedup $\eta_{\mathsf{s}} = 2$

	17% loss		2% loss		0.6% loss	
n_I	DD	DN	DD	DN	DD	DN
0	0.80	0.80	0.80	0.80	0.80	0.80
1	**0.20**	**0.00**	**0.20**	**0.20**	**0.20**	**0.00**
2	**0.20**	**0.00**	**0.20**	**0.00**	**0.00**	**0.00**
3	0.40	0.40	0.40	**0.20**	**0.20**	**0.00**
4	0.60	0.60	0.60	0.40	0.40	**0.20**

Fault at $k_{\mathsf{F}} = 11$ and speedup $\eta_{\mathsf{s}} = 5$

n_I	DD	DN	DD	DN	DD	DN
0	0.82	0.82	0.82	0.82	0.91	0.91
1	0.36	0.36	0.27	0.27	0.27	0.27
2	**0.09**	**0.00**	**0.09**	**0.00**	**0.00**	**0.00**
3	0.18	**0.09**	0.27	**0.09**	**0.09**	**0.09**
4	0.27	0.18	0.36	0.18	0.18	0.18

Parallel setup: $0.6\% - 0.00047\%$ information loss

Adaptivly steering: $n_F = \eta_s n_I + \Delta n_F$

- **DD** and **DN** strategies: one failure at $k_\mathsf{F} = 7$ with $n_I = 3$

Size	No Rec	$\eta_\mathsf{s} = 1$	2	4	8	$\eta_\mathsf{s} = 1$	2	4	8
$4.5 \cdot 10^8$	13.73 (21)	9.14	**-0.01**	**0.00**	**-0.00**	11.47	2.30	**0.02**	**0.04**
$2.1 \cdot 10^9$	11.69 (20)	9.31	**0.04**	**0.08**	**0.11**	9.35	2.41	**0.11**	**0.14**
$1.2 \cdot 10^{10}$	12.49 (20)	7.42	**-0.01**	**-0.02**	**-0.00**	9.96	2.54	**0.06**	**0.06**
$8.2 \cdot 10^{10}$	11.16 (19)	5.54	**0.08**	**0.07**	**0.04**	8.36	**0.11**	**0.15**	**0.17**
$6.0 \cdot 10^{11}$	13.59 (19)	3.47	**0.13**	**0.19**	**0.13**	**0.13**	**0.24**	**0.29**	**0.26**

- **DN strategy:** two consequtive failures at $k_\mathsf{F} = 5$ and $k_\mathsf{F} = 9$ with $\eta_\mathsf{s} = 4$

Size	No Rec	(1,2)	(1,3)	(2,2)	(2,3)
$4.5 \cdot 10^8$	18.35 (23)	**0.02**	**0.03**	**0.03**	**0.04**
$2.1 \cdot 10^9$	16.33 (22)	**0.05**	**0.06**	**0.06**	**0.06**
$1.2 \cdot 10^{10}$	17.43 (22)	**0.07**	**0.08**	**0.09**	**0.08**
$8.2 \cdot 10^{10}$	16.69 (21)	**0.16**	**0.17**	**0.16**	**0.17**
$6.0 \cdot 10^{11}$	20.64 (21)	**0.30**	**0.33**	**0.36**	**0.36**

Global recovery can be fully compensate fault wrt time-to-solution.

Towards Geophysics

Application to geophysics

Stokes system with mixed bc's in a spherical shell $\Omega = \{\mathbf{x} \in \mathbb{R}^3 : 0.55 < \|\mathbf{x}\|_2 < 1\}$,

$$\begin{aligned} -\mathrm{div}\,(2\nu(\tau,\mathbf{x})D(\mathbf{u})) + \nabla p &= \mathbf{f} && \text{in } \Omega, \\ \mathrm{div}\,\mathbf{u} &= 0 && \text{in } \Omega, \\ \mathbf{u} &= \mathbf{g} && \text{on } \Gamma_{\mathsf{D}}, \\ D(\mathbf{u})\mathbf{n}\cdot\mathbf{t} &= 0 && \text{on } \Gamma_{\mathsf{FS}}, \\ \mathbf{u}\cdot\mathbf{n} &= 0 && \text{on } \Gamma_{\mathsf{FS}}, \end{aligned}$$

with $D(\mathbf{u}) = 1/2(\nabla\mathbf{u} + \nabla\mathbf{u}^\top)$.

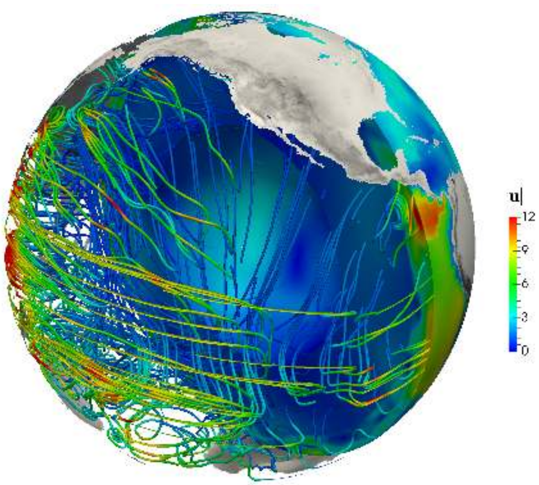
- Right-hand side with scaled temperature τ [Zhong, Gurnis et. al. 2008]

$$\mathbf{f} = \mathsf{Ra}\,\tau\,\frac{\mathbf{x}}{\|\mathbf{x}\|_2},$$

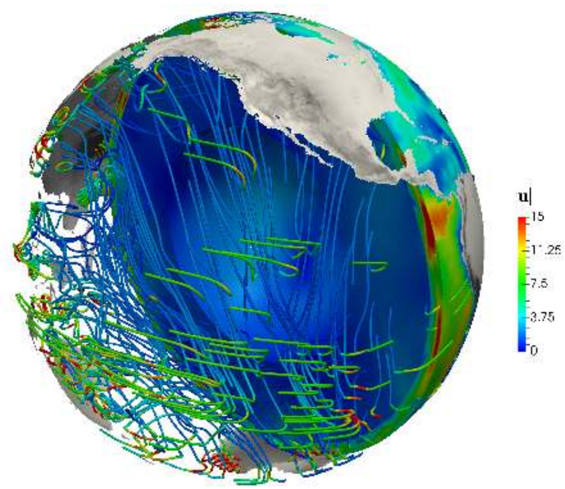
- Dirichlet datum $\mathbf{g}$, plate velocity data (from the open source software GPlates) [Williams, Müller, Landgrebe, Whittaker 2012],

Geophysical Application

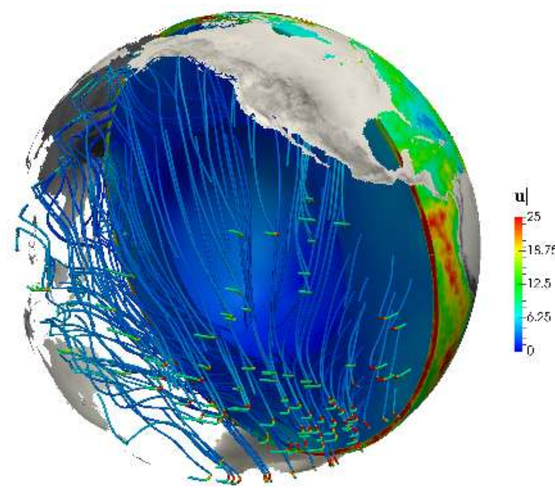
Influence of the asthenossphere thickness



660km



410km



200km

Viscosity model with $d_a \in \{200/6371, 410/6371, 660/6371\}$ according to [Davies et al. 2012]

$$\nu(\mathbf{x}) = \exp(4.61 \, \tfrac{1-\|x\|_2}{1-r_{\text{inner}}} - 2.99\,\tau) \begin{cases} 1/10\, 6.371^3 \, d_{\text{a}}^3 & \text{for } \|\mathbf{x}\|_2 > 1 - d_{\text{a}}, \\ 1 & \text{else.} \end{cases}$$

Conclusion

- Solving problems up to 10^{13} unknowns
- Fault tolerant multigrid method
- Local recovery strategies
- Recovery in the Stokes system
- Asynchronous accelerated global recovery strategies
- Geophysical application

Outlook

- Statistical evaluation of faults and their recovery
- Advanced recoupling strategies
- Implementation in a fault-tolerant MPI environment (ULFM)
- Combination/Comparison of ABFT with check-pointing

References

[1] E. Agullo, L. Giraud, A. Guermouche, J. Roman, and M. Zounon. Towards resilient parallel linear Krylov solvers: recover-restart strategies. Research Report RR-8324, July 2013.

[2] B. Bergen, T. Gradl, U. Rüde, and F. Hülsemann. A massively parallel multigrid method for finite elements. Computing in Science and Engineering, 8(6):56–62, 2006.

[3] F. Cappello, A. Geist, S. Kale, B. Kramer, and M. Snir. Toward exascale resilience: 2014 update. Supercomput. Front. Innov., 1:1–28, 2014.

[4] J. Dongarra, T. Herault, and Y. Robert. *Fault-Tolerance Techniques for High-Performance Computing*. Springer International Publisher, Springer International Publisher. Switzerland, 2015.

[5] B. Gmeiner, M. Huber, L. John, U. Rüde, and B. Wohlmuth. A quantitative performance analysis for Stokes solvers at the extreme scale, submitted, arXiv:1511.02134.

[6] D. Göddeke, M. Altenbernd, and D. Ribbrock. Fault-tolerant finite-element multigrid algorithms with hierarchically compressed asynchronous checkpointing. Parallel Comput., 49:117 – 135, 2015.

[7] M. Huber, B. Gmeiner, U. Rüde, and B. Wohlmuth. Resilience for Exascale Enabled Multigrid Methods. arXiv:1501.07400

[8] J. Langou, Z. Chen, G. Bosilca, and J. Dongarra. Recovery patterns for iterative methods in a parallel unstable environment. SIAM J. Sci. Comput., 30(1):102–116, 2008.

[9] J. Schöberl and W. Zulehner. On Schwarz-type smoothers for saddle point problems. Numer. Math., 95(2):377–399, 2003.