Efficient Schwarz preconditioning techniques for nonlinear problems

Alexander Heinlein[1]

28th International Conference on Domain Decomposition Methods (DD28), KAUST, Saudi Arabia, January 28 - February 1, 2024

[1]Delft University of Technology

Based on joint work with Axel Klawonn and Martin Lanser (Universität zu Köln) and Mauro Perego, Sivasankaran Rajamanickam, and Ichitaro Yamazaki (Sandia National Laboratories)

Linear & Nonlinear Preconditioning

Let us consider the nonlinear problem arising from the discretization of a partial differential equation

$$\boldsymbol{F}(\boldsymbol{u}) = 0.$$

We solve the problem using a **Newton-Krylov approach**, i.e., we solve a sequence of linearized problems using a Krylov subspace method:

$$D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\Delta\boldsymbol{u}^{(k+1)} = \boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right).$$

Linear preconditioning

In linear preconditioning, we **improve the convergence speed of the linear solver** by constructing a **linear operator** $\boldsymbol{M}^{-1}$ and solve linear systems

$$\boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\Delta\boldsymbol{u}^{(k+1)} = \boldsymbol{M}^{-1}\boldsymbol{F}(\boldsymbol{u}^{(k)}).$$

Goal:

- $\kappa\left(\boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\right) \approx 1.$

$\Rightarrow \boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right) \approx \boldsymbol{I}.$

Nonlinear preconditioning

In nonlinear preconditioning, we **improve the convergence speed of the nonlinear solver** by constructing a **nonlinear operator** G and solve the nonlinear system

$$(\boldsymbol{G} \circ \boldsymbol{F})(\boldsymbol{u}) = 0.$$

Goals:

- $\boldsymbol{G} \circ \boldsymbol{F}$ almost linear.
- Additionally: $\kappa\left(D(\boldsymbol{G} \circ \boldsymbol{F})(\boldsymbol{u})\right) \approx 1.$

Let us consider the nonlinear problem arising from the discretization of a partial differential equation

$$\boldsymbol{F}(\boldsymbol{u}) = 0.$$

We solve the problem using a **Newton-Krylov approach**, i.e., we solve a sequence of linearized problems using a Krylov subspace method:

$$D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\Delta\boldsymbol{u}^{(k+1)} = \boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right).$$

Linear preconditioning

In linear preconditioning, we **improve the convergence speed of the linear solver** by constructing a **linear operator** $\boldsymbol{M}^{-1}$ and solve linear systems

$$\boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\Delta\boldsymbol{u}^{(k+1)} = \boldsymbol{M}^{-1}\boldsymbol{F}(\boldsymbol{u}^{(k)}).$$

Goal:

- $\kappa\left(\boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\right) \approx 1.$

$\Rightarrow$ $\boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right) \approx \boldsymbol{I}.$

Nonlinear preconditioning

In nonlinear preconditioning, we **improve the convergence speed of the nonlinear solver** by constructing a **nonlinear operator** G and solve the nonlinear system

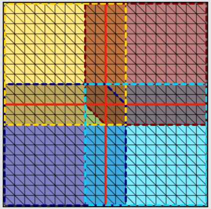
$$(\boldsymbol{G} \circ \boldsymbol{F})(\boldsymbol{u}) = 0.$$

Goals:

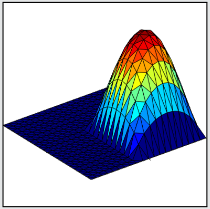
- $\boldsymbol{G} \circ \boldsymbol{F}$ almost linear.
- Additionally: $\kappa\left(D(\boldsymbol{G} \circ \boldsymbol{F})(\boldsymbol{u})\right) \approx 1.$

One-level Schwarz preconditioner

Overlap $\delta = 1h$



Solution of local problem



Based on an **overlapping domain decomposition**, we define a **one-level Schwarz operator**

$$M_{\text{OS-1}}^{-1} K = \sum_{i=1}^{N} R_i^\top K_i^{-1} R_i K,$$

where R_i and $R_i^\top$ are restriction and prolongation operators corresponding to Ω_i', and $K_i := R_i K R_i^\top$.

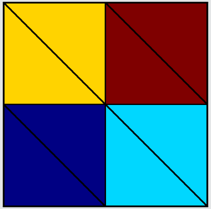
Condition number estimate:

$$\kappa \left(M_{\text{OS-1}}^{-1} K \right) \leq C \left(1 + \frac{1}{H\delta} \right)$$

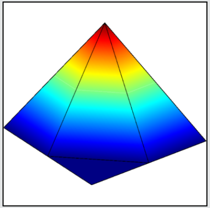
with subdomain size H and overlap width δ.

Lagrangian coarse space

Coarse triangulation



Coarse solution



The **two-level overlapping Schwarz operator** reads

$$M_{\text{OS-2}}^{-1} K = \underbrace{\Phi K_0^{-1} \Phi^\top K}_{\text{coarse level – global}} + \underbrace{\sum_{i=1}^{N} R_i^\top K_i^{-1} R_i K}_{\text{first level – local}},$$

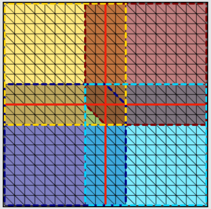
where Φ contains the coarse basis functions and $K_0 := \Phi^\top K \Phi$; cf., e.g., Toselli, Widlund (2005). The construction of a Lagrangian coarse basis requires a coarse triangulation.

Condition number estimate:

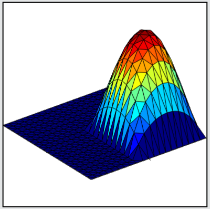
$$\kappa \left(M_{\text{OS-2}}^{-1} K \right) \leq C \left(1 + \frac{H}{\delta} \right)$$

One-level Schwarz preconditioner

Overlap $\delta = 1h$

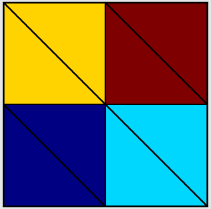


Solution of local problem

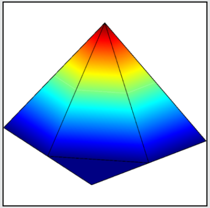


Lagrangian coarse space

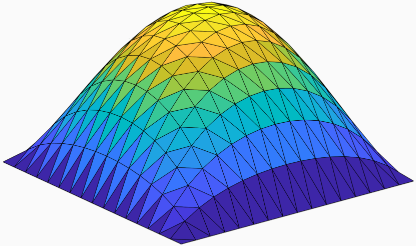
Coarse triangulation

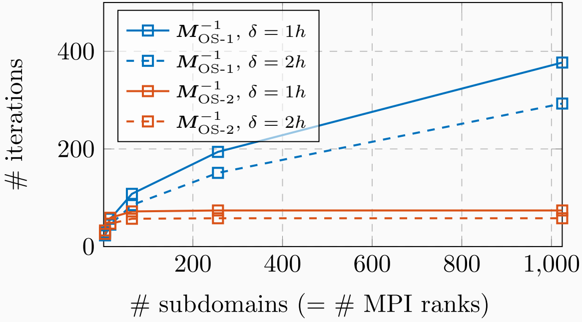


Coarse solution



Diffusion model problem in two dimensions,
$H/h = 100$





FROSch (Fast and Robust Overlapping Schwarz) Framework in Trilinos

Software

- Object-oriented C++ domain decomposition solver framework with MPI-based distributed memory parallelization
- Part of TRILINOS with support for both parallel linear algebra packages EPETRA and TPETRA
- Node-level parallelization and performance portability on CPU and GPU architectures through KOKKOS and KOKKOSKERNELS
- Accessible through unified TRILINOS solver interface STRATIMIKOS

Methodology

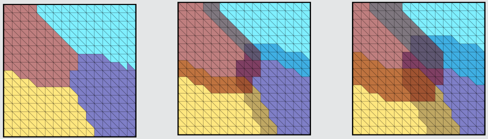
- Parallel scalable multi-level Schwarz domain decomposition preconditioners
- Algebraic construction based on the parallel distributed system matrix
- Extension-based coarse spaces

Team (active)

- Filipe Cumaru (TU Delft)
- Kyrill Ho (UCologne)
- Siva Rajamanickam (SNL)
- Oliver Rheinbach (TUBAF)
- Ichitaro Yamazaki (SNL)
- Alexander Heinlein (TU Delft)
- Axel Klawonn (UCologne)
- Friederike Röver (TUBAF)
- Lea Saßmannshausen (UCologne)

First level – Overlapping DD

In FROSch, the overlapping subdomains $\Omega'_1, \ldots, \Omega'_N$ are constructed by **recursively adding layers of elements** to the nonoverlapping subdomains; this can be performed based on the sparsity pattern of K.



Nonoverl. DD | Overlap $\delta = 1h$ | Overlap $\delta = 2h$

First level – Computation K_i

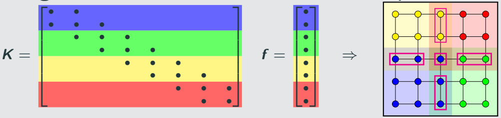
The overlapping matrices

$$\boldsymbol{K}_i = \boldsymbol{R}_i \boldsymbol{K} \boldsymbol{R}_i^\top$$

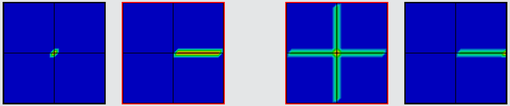
can easily be extracted from K since R_i is just a **global-to-local index mapping**.

Coarse level – Interface basis

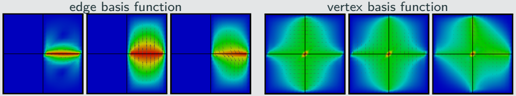
1. **Algebraic** identification of interface components:



2. Interface basis = partition of unity $\times$ null space



Coarse level – Extensions into interior



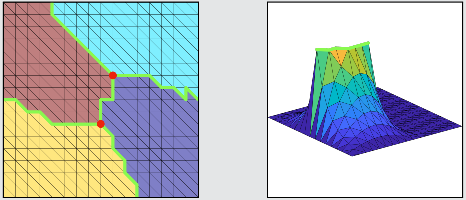
The values in the interior of the subdomains are computed via the **extension operator**:

$$\Phi = \begin{bmatrix} \Phi_I \\ \Phi_\Gamma \end{bmatrix} = \begin{bmatrix} -\boldsymbol{K}_{II}^{-1} \boldsymbol{K}_{I\Gamma}^T \Phi_\Gamma \\ \Phi_\Gamma \end{bmatrix}.$$

(For elliptic problems: **energy-minimizing extension**)

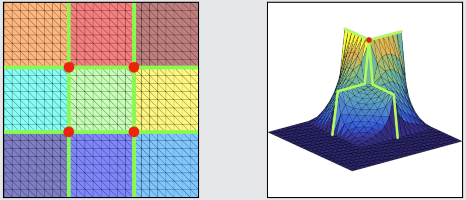
Examples of FROSch Coarse Spaces

GDSW (Generalized Dryja–Smith–Widlund)



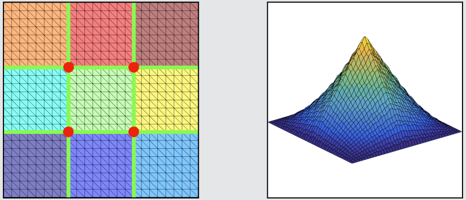
- Dohrmann, Klawonn, Widlund (2008)
- Dohrmann, Widlund (2009, 2010, 2012)

RGDSW (Reduced dimension GDSW)



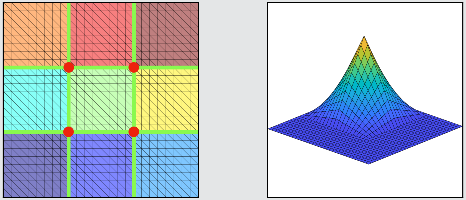
- Dohrmann, Widlund (2017)
- H., Klawonn, Knepper, Rheinbach, Widlund (2022)

MsFEM (Multiscale Finite Element Method)



- Hou (1997), Efendiev and Hou (2009)
- Buck, Iliev, and Andrä (2013)
- H., Klawonn, Knepper, Rheinbach (2018)

Q1 Lagrangian / piecewise bilinear

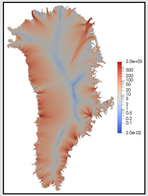


Piecewise linear interface partition of unity functions and a **structured domain decomposition**.

Stationary velocity problem

We use a **first-order (or Blatter-Pattyn) approximation of the Stokes equations**

$$\begin{cases} -\nabla \cdot (2\mu\, \dot{\epsilon}_1) &= -\rho_i\, |\boldsymbol{g}|\, \partial_x s, \\ -\nabla \cdot (2\mu\, \dot{\epsilon}_2) &= -\rho_i\, |\boldsymbol{g}|\, \partial_y s, \end{cases}$$



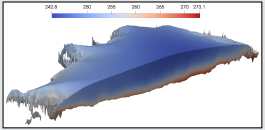
with ice density ρ_i, ice surface elevation s, gravity acceleration $\boldsymbol{g}$, and strain rates $\dot{\epsilon}_1$ and $\dot{\epsilon}_2$; cf. Blatter (1995) and Pattyn (2003).

Ice viscosity modeled by **Glen's law**: $\mu = \frac{1}{2} A(T)^{-\frac{1}{n}}\, \dot{\epsilon}_e^{\frac{1-n}{n}}$.

Stationary temperature problem

The **enthalpy equation** reads

$$\nabla \cdot \boldsymbol{q}(h) + \boldsymbol{u} \cdot \nabla h = 4\mu\, \epsilon_e^2$$

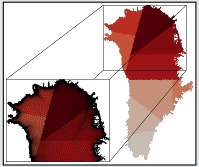


with the **enthalpy flux**

$$\boldsymbol{q}(h) = \begin{cases} \frac{k}{\rho_i c_i} \nabla h, & \text{for cold ice } (h \leq h_m), \\ \frac{k}{\rho_i c_i} \nabla h_m + \rho_w L \boldsymbol{j}(h), & \text{for temperate ice.} \end{cases}$$

Set of complex boundary conditions: Dirichlet, Neumann, Robin, and Stefan boundary and coupling conditions.

FROSch preconditioners



Construct **overlapping DD** from **nonoverlapping DD** based on the **surface mesh**.

For the coupled problem, we construct a **monolithic two-level (R)GDSW preconditioner** (Heinlein, Hochmuth, Klawonn (2019, 2020))

$$\mathcal{M}_{\text{GDSW}}^{-1} = \phi \mathcal{A}_0^{-1} \phi^\top + \sum_{i=1}^{N} \mathcal{R}_i^\top \mathcal{A}_i^{-1} \mathcal{R}_i,$$

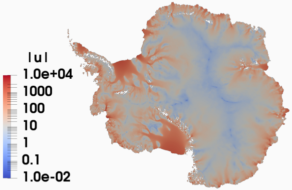
where the linearized system is of the form

$$\mathcal{A}x := \begin{bmatrix} A_u & C_{uT} \\ C_{Tu} & A_T \end{bmatrix} \begin{bmatrix} x_u \\ x_T \end{bmatrix} = \begin{bmatrix} \tilde{r}_u \\ \tilde{r}_T \end{bmatrix} =: r.$$

For single-physics problems, we employ a **standard (R)GDSW preconditioner**.

Antarctica Velocity Problem – Reuse Strategies (Strong Scaling)

We employ different **reuse strategies** to **reduce the setup costs** of the two-level preconditioner



$$M_{\mathrm{GDSW}}^{-1} = \boldsymbol{\Phi} \mathbf{K_0}^{-1} \boldsymbol{\Phi}^\top + \sum_{i=1}^{N} \mathbf{R_i}^\top K_i^{-1} \mathbf{R_i}.$$

reuse	restriction operators + symbolic fact. (1st level)			+ coarse basis + symbolic fact. (2nd level)			+ coarse matrix		
MPI ranks	avg. its (nl its)	avg. setup	avg. solve	avg. its (nl its)	avg. setup	avg. its solve	avg. its (nl its)	avg. setup	avg. solve
512	**41.9** (11)	25.10 s	**12.29 s**	42.6 (11)	14.99 s	12.50 s	46.7 (11)	**14.94 s**	13.81 s
1 024	**43.3** (11)	9.18 s	**5.85 s**	44.5 (11)	**5.65 s**	6.08 s	49.2 (11)	5.75 s	6.78 s
2 048	**41.4** (11)	4.15 s	**2.63 s**	42.7 (11)	3.11 s	2.79 s	47.7 (11)	**2.92 s**	3.10 s
4 096	**41.2** (11)	1.66 s	**1.49 s**	42.5 (11)	1.07 s	1.54 s	48.9 (11)	**0.95 s**	1.75 s
8 192	**40.2** (11)	1.26 s	**1.06 s**	42.0 (11)	1.20 s	1.16 s	50.1 (11)	**0.63 s**	1.35 s

Problem: Velocity **Mesh:** Antarctica 4 km hor. resolution 20 vert. layers **Size:** 35.3 m degrees of freedom (P1 FE) **Coarse space:** RGDSW

Cf. Heinlein, Perego, Rajamanickam (2022)

Greenland Coupled Problem – Coarse Spaces

Fully coupled extensions

Extensions for coarse basis

$$\phi = \begin{bmatrix} -\mathcal{A}_{II}^{-1}\mathcal{A}_{\Gamma I}^{T}\Phi_{\Gamma} \\ \Phi_{\Gamma} \end{bmatrix} = \begin{bmatrix} \phi_I \\ \phi_{\Gamma} \end{bmatrix}$$

using the coupled matrix $\mathcal{A}$.

Decoupled extensions

Extensions for coarse basis

$$\phi = \begin{bmatrix} -\tilde{\mathcal{A}}_{II}^{-1}\tilde{\mathcal{A}}_{\Gamma I}^{T}\Phi_{\Gamma} \\ \Phi_{\Gamma} \end{bmatrix} = \begin{bmatrix} \phi_I \\ \phi_{\Gamma} \end{bmatrix}$$

using the decoupled matrix

$$\tilde{\mathcal{A}} = \begin{bmatrix} A_u & 0 \\ 0 & A_T \end{bmatrix}.$$

fully coupled extensions							
		no reuse			reuse coarse basis		
MPI ranks	dim V_0	avg. its (nl its)	avg. setup	avg. solve	avg. its (nl its)	avg. setup	avg. solve
256	1 400	100.1 (27)	4.10 s	6.40 s	**18.5** (70)	**2.28 s**	**1.07 s**
512	2 852	129.1 (28)	1.88 s	4.20 s	**24.6** (38)	**1.04 s**	**0.70 s**
1 024	6 036	191.2 (65)	1.21 s	4.76 s	**34.2** (32)	**0.66 s**	**0.70 s**
2 048	12 368	237.4 (30)	0.96 s	4.06 s	**37.3** (30)	**0.60 s**	**0.58 s**

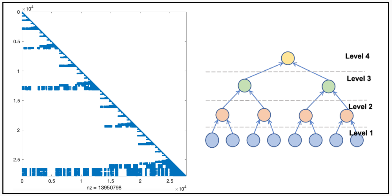
decoupled extensions							
		no reuse			reuse coarse basis		
MPI ranks	dim V_0	avg. its (nl its)	avg. setup	avg. solve	avg. its (nl its)	avg. setup	avg. solve
256	1 400	23.6 (29)	3.90 s	1.32 s	**21.5** (34)	**2.23 s**	**1.18 s**
512	2 852	27.5 (30)	1.83 s	0.78 s	**26.4** (33)	**1.13 s**	**0.78 s**
1 024	6 036	30.1 (29)	1.19 s	**0.60 s**	**28.6** (43)	**0.66 s**	0.61 s
2 048	12 368	36.4 (30)	0.69 s	0.56 s	**31.2** (50)	**0.57 s**	**0.55 s**

Problem: Coupled **Mesh:** Greenland 3-30 km hor. resolution 20 vert. layers **Size:** 7.5 m degrees of freedom (P1 FE) **Coarse space:** RGDSW

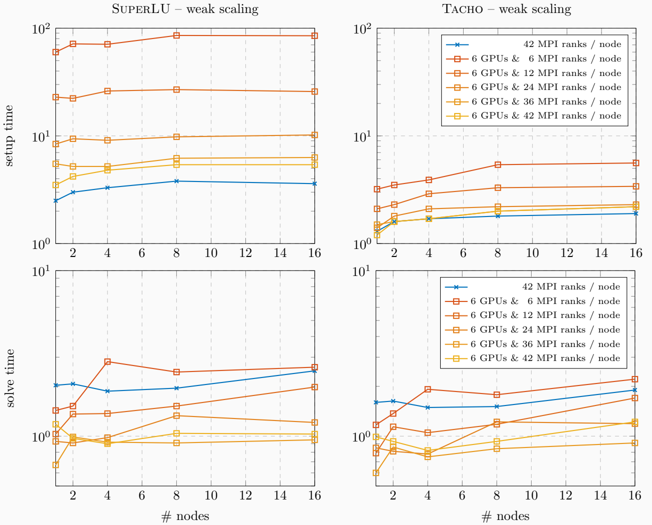
Parallel supernode-based triangular solver:

1. Factorization using a **sparse direct solver** typically leads to triangular matrices with **dense blocks** called **supernodes**
2. **Supernode-based level-set scheduling**, where **all leaf-supernodes within one level are solved in parallel** (batched kernels for hierarchical parallelism)

Cf. **Yamazaki, Rajamanickam, and Ellingwood (2020)**.



Three-Dimensional Linear Elasticity – Weak Scalability of FROSch



Computations on Summit (OLCF): 42 IBM Power9 CPU cores and 6 NVIDIA V100 GPUs per node.

Yamazaki, Heinlein, Rajamanickam (2023)

	ILU level	0	1	2	3
		setup			
CPU	No	**1.5**	**1.9**	**3.0**	**4.8**
CPU	ND	1.6	2.6	4.4	7.4
GPU	KK(No)	1.4	1.5	1.8	2.4
GPU	KK(ND)	1.7	2.0	2.9	5.2
GPU	Fast(No)	**1.5**	**1.6**	**2.1**	**3.2**
GPU	Fast(ND)	1.5	1.7	2.5	4.5
speedup		**1.0**×	**1.2**×	**1.4**×	**1.5**×
		solve			
CPU	No	**2.55 (158)**	**3.60 (112)**	**5.28 (99)**	**6.85 (88)**
CPU	ND	4.17 (227)	5.36 (134)	6.61 (105)	7.68 (88)
GPU	KK(No)	3.81 (158)	4.12 (112)	4.77 (99)	5.65 (88)
GPU	KK(ND)	2.89 (227)	4.27 (134)	5.57 (105)	6.36 (88)
GPU	Fast(No)	**1.14 (173)**	**1.11 (141)**	**1.26 (134)**	**1.43 (126)**
GPU	Fast(ND)	1.49 (227)	1.15 (137)	1.10 (109)	1.22 (100)
speedup		**2.2**×	**3.2**×	**4.3**×	**4.8**×

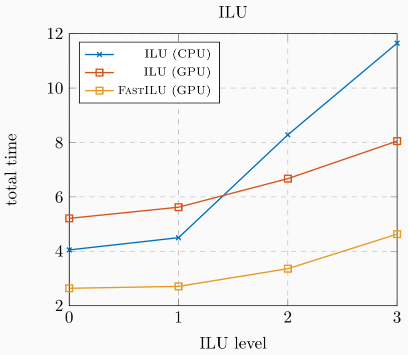
Computations on Summit (OLCF):
42 IBM Power9 CPU cores and 6 NVIDIA V100 GPUs per node.

Yamazaki, Heinlein, Rajamanickam (2023)

ILU variants

- KokkosKernels ILU (KK)
- FastILU (Fast); cf. **Chow, Patel (2015)** and **Boman, Patel, Chow, Rajamanickam (2016)**

No reordering (**No**) and nested dissection (**ND**)



Linear & Nonlinear Preconditioning

Let us consider the nonlinear problem arising from the discretization of a partial differential equation

$$\boldsymbol{F}(\boldsymbol{u}) = 0.$$

We solve the problem using a **Newton-Krylov approach**, i.e., we solve a sequence of linearized problems using a Krylov subspace method:

$$D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\Delta\boldsymbol{u}^{(k+1)} = \boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right).$$

Linear preconditioning

In linear preconditioning, we **improve the convergence speed of the linear solver** by constructing a **linear operator** $\boldsymbol{M}^{-1}$ and solve linear systems

$$\boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\Delta\boldsymbol{u}^{(k+1)} = \boldsymbol{M}^{-1}\boldsymbol{F}(\boldsymbol{u}^{(k)}).$$

Goal:

- $\kappa\left(\boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right)\right) \approx 1.$

$\Rightarrow \boldsymbol{M}^{-1}D\boldsymbol{F}\left(\boldsymbol{u}^{(k)}\right) \approx \boldsymbol{I}.$

Nonlinear preconditioning

In nonlinear preconditioning, we **improve the convergence speed of the nonlinear solver** by constructing a **nonlinear operator** G and solve the nonlinear system

$$(\boldsymbol{G} \circ \boldsymbol{F})(\boldsymbol{u}) = 0.$$

Goals:

- $\boldsymbol{G} \circ \boldsymbol{F}$ almost linear.
- Additionally: $\kappa\left(D(\boldsymbol{G} \circ \boldsymbol{F})(\boldsymbol{u})\right) \approx 1.$

Nonlinear Domain Decomposition Methods

Additive nonlinear left preconditioners (based on Schwarz methods)

ASPIN/ASPEN: Cai, Keyes (2002); Cai, Keyes, Marcinkowski (2002); Hwang, Cai (2005, 2007); Groß, Krause (2010, 2013), ...

RASPEN: Dolean, Gander, Kherijii, Kwok, Masson (2016)

MSPIN: Keyes, Liu (2015, 2016, 2021); Liu, Wei, Keyes (2017); Kopanicáková, Kothari, Krause (2023), ...

Multi-Level (R)ASPIN/(R)ASPEN: Heinlein, Lanser (2020); Heinlein, Lanser, Klawonn (2022, in prep.)

Nonlinear right preconditioners (based on either FETI or BDDC)

Nonlinear FETI-DP/BDDC: Klawonn, Lanser, Rheinbach (2012, 2013, 2014, 2015, 2016, 2018); Klawonn, Lanser, Rheinbach, Uran (2017, 2018); Klawonn Lanser, Uran (2021, 2023), ...

Nonlinear Elimination: Hwang, Lin, Cai (2010); Cai, Li (2011); Wang, Su, Cai (2015); Hwang, Su, Cai (2016); Gong, Cai (2018); Luo, Shiu, Chen, Cai (2019); Gong, Cai (2019); Liu, Hwang, Luo, Cai, Keyes (2022); ...

Nonlinear Neumann-Neumann: Bordeu, Boucard, Gosselet (2009)

Nonlinear FETI-1: Pebrel, Rey, Gosselet (2008); Negrello, Gosselet, Rey (2021)

Other DD work reversing linearization and decomposition: Ganis, Juntunen, Pencheva, Wheeler, Yotov (2014); Ganis, Kumar, Pencheva, Wheeler, Yotov (2014)

Early nonlinear DD work: Cai, Dryja (1994); Dryja, Hackbusch (1997)

ASPEN & ASPIN

Our approach is based on the nonlinear one-level Schwarz methods **ASPE/IN (Additive Schwarz Preconditioned Exact/Inexact Newton)** introduced in Cai and Keyes (2002). The nonlinear finite element problem

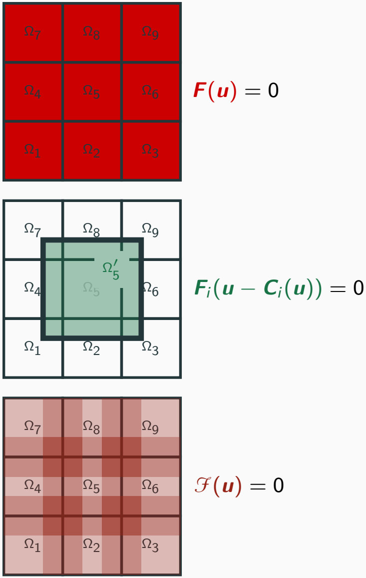
$$\boldsymbol{F}(\boldsymbol{u}) = 0 \quad \text{with } \boldsymbol{F} : V \to V$$

is reformulated as

$$\mathcal{F}(\boldsymbol{u}) = \boldsymbol{G}(\boldsymbol{F}(\boldsymbol{u})) = 0.$$

The **nonlinear left-preconditioner $\boldsymbol{G}$ is given implicitly** by **the local nonlinear problem** on each of the (overlapping) subdomains.

$$\boldsymbol{F}_i(\boldsymbol{u} - \underbrace{\boldsymbol{C}_i(\boldsymbol{u})}_{\text{local correction}}), \; i = 1, \ldots, N.$$



Nonlinear One-Level Schwarz Preconditioners

RASPEN

Local corrections $\boldsymbol{T}_i(\boldsymbol{u})$:

$$\boldsymbol{R}_i \boldsymbol{F}(\boldsymbol{u} - \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u})) = 0, \; i = 1, \ldots, N, \text{ with}$$

$$\text{restrictions} \quad \boldsymbol{R}_i \quad : V \to V_i,$$
$$\text{prolongations} \quad \boldsymbol{P}_i, \widetilde{\boldsymbol{P}}_i : V_i \to V.$$

Nonlinear RASPEN problem:

$$\mathcal{F}_{RA}(\boldsymbol{u}) := \sum\nolimits_{i=1}^{N} \widetilde{\boldsymbol{P}}_i \boldsymbol{T}_i(\boldsymbol{u}) = 0$$

We solve $\mathcal{F}_{RA}(\boldsymbol{u}) = 0$ using Newton's method with $\boldsymbol{u}_i = \boldsymbol{u} - \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u})$. The Jacobian writes

$$D\mathcal{F}_{RA}(\boldsymbol{u}) = \sum\nolimits_{i=1}^{N} \underbrace{\widetilde{\boldsymbol{P}}_i \left(\boldsymbol{R}_i D\boldsymbol{F}(\boldsymbol{u}_i) \boldsymbol{P}_i\right)^{-1} \boldsymbol{R}_i \; D\boldsymbol{F}(\boldsymbol{u}_i)}_{\substack{\text{local Schwarz operators} \\ \text{(preconditioned operators)}}}$$

- $\sum_{i=1}^{N} \widetilde{\boldsymbol{P}}_i \boldsymbol{R}_i = \boldsymbol{I}$
- **Reduced comm.** & (often) **better conv.**

Cf. Dolean et al. (2016).

Results for p-Laplacian model problem

p-Laplacian model problem

$$\begin{aligned} -\alpha \Delta_p u &= 1 \quad && \text{in } \Omega, \\ u &= 0 \quad && \text{on } \partial\Omega. \end{aligned}$$

with $\alpha \Delta_p u := \mathrm{div}(\alpha |\nabla u|^{p-2} \nabla u)$.

$p = 4$; $H/h = 16$; overlap $\delta = 1$					
		nonlin.		**lin.**
N	**solver**	**outer it.**	**inner it. (avg.)**	**GMRES it. (sum)**
9	NK-RAS	18	-	272
	RASPEN	5	25.2	89
25	NK-RAS	19	-	488
	RASPEN	6	28.3	172
49	NK-RAS	20	-	691
	RASPEN	6	27.3	232

$\Rightarrow$ **Improved nonlinear convergence**, but **no scalability** in the linear iterations.

Nonlinear Two-Level Schwarz Preconditioners

Two-level (R)ASPEN

Local/coarse corrections $T_i(u)$:

$$R_i F(u - P_i T_i(u)) = 0, \; i = 0, 1, \ldots, N, \; \text{with}$$

$$\text{restrictions} \quad R_i : V \to V_i,$$

$$\text{prolongations} \quad P_i : V_i \to V.$$

Nonlinear two-level ASPEN problem:

$$\mathcal{F}_A(u) := P_0 T_0(u) + \sum_{i=1}^{N} P_i T_i(u) = 0$$

We solve $\mathcal{F}_A(u) = 0$ using Newton's method with $u_i = u - P_i T_i(u)$. The Jacobian writes

$$D\mathcal{F}_{RA}(u) = \overbrace{P_0 \left(R_0 DF(u_0) P_0\right)^{-1} R_0 \; DF(u_0)}^{\text{coarse Schwarz operator}} + \sum_{i=1}^{N} \underbrace{P_i \left(R_i DF(u_i) P_i\right)^{-1} R_i \; DF(u_i)}_{\text{local Schwarz operators}}$$

Cf. Heinlein & Lanser (2020).

Results for p-Laplacian model problem

1-lvl	One-level RASPEN
2-lvl A	Two-level RASPEN with **additively coupled coarse level**
2-lvl M	Two-level RASPEN with **multiplicatively coupled coarse level**

$p = 4$; $H/h = 16$; overlap $\delta = 1$

N	RASPEN solver	nonlin. outer it.	nonlin. inner it. (avg.)	nonlin. coarse it.	lin. GMRES it. (sum)
9	1-lvl	5	25.2	-	89
	2-lvl A	6	33.4	27	93
	2-lvl M	4	17.1	29	52
49	1-lvl	6	27.3	-	232
	2-lvl A	6	29.2	28	137
	2-lvl M	4	12.6	29	80

$\Rightarrow$ **Improved nonlinear convergence and scalability.**

Linear Schwarz operators

$$
\begin{aligned}
&\text{(Additive)} && \boldsymbol{P}_A = \sum_{i=0}^{N} \boldsymbol{P}_i \\
&\text{(Multiplicative 1)} && \boldsymbol{P}_{M1} = \boldsymbol{I} - \left(\boldsymbol{I} - \sum_{i=1}^{N} \boldsymbol{P}_i\right)(\boldsymbol{I} - \boldsymbol{P}_0) \\
&\text{(Multiplicative 2)} && \boldsymbol{P}_{M2} = \boldsymbol{I} - (\boldsymbol{I} - \boldsymbol{P}_0)\left(\boldsymbol{I} - \sum_{i=1}^{N} \boldsymbol{P}_i\right) \\
&\text{(Multiplicative symmetric)} && \boldsymbol{P}_{M} = \boldsymbol{I} - (\boldsymbol{I} - \boldsymbol{P}_0)\left(\boldsymbol{I} - \sum_{i=1}^{N} \boldsymbol{P}_i\right)(\boldsymbol{I} - \boldsymbol{P}_0)
\end{aligned}
$$

Nonlinear Schwarz operators

$$
\begin{aligned}
&\text{(Additive)} && \mathcal{F}_A(\boldsymbol{u}) = \sum_{i=0}^{N} \boldsymbol{R}_i^\top \boldsymbol{T}_i(\boldsymbol{u}) \\
&\text{(Multiplicative 1)} && \mathcal{F}_{M1}(\boldsymbol{u}) = \sum_{i=1}^{N} \boldsymbol{R}_i^\top \boldsymbol{T}_i(\boldsymbol{u} - \boldsymbol{P}_0 \boldsymbol{T}_0(\boldsymbol{u})) + \boldsymbol{P}_0 \boldsymbol{T}_0(\boldsymbol{u}) \\
&\text{(Multiplicative 2)} && \mathcal{F}_{M2}(\boldsymbol{u}) = \sum_{i=1}^{N} \boldsymbol{R}_i^\top \boldsymbol{T}_i(\boldsymbol{u}) + \boldsymbol{P}_0 \boldsymbol{T}_0\left(\boldsymbol{u} - \sum_{i=1}^{N} \boldsymbol{R}_i^\top \boldsymbol{T}_i(\boldsymbol{u})\right) \\
&\text{(Multiplicative symmetric)} && \mathcal{F}_{M}(\boldsymbol{u}) = \boldsymbol{P}_0 \boldsymbol{T}_0(\boldsymbol{w}) + \sum_{i=1}^{N} \boldsymbol{R}_i^\top \boldsymbol{T}_i(\boldsymbol{v}) + \boldsymbol{P}_0 \boldsymbol{T}_0(\boldsymbol{u})
\end{aligned}
$$

The **tangent matrices** of **nonlinear Schwarz operators** are **corresponding linear Schwarz operators**!

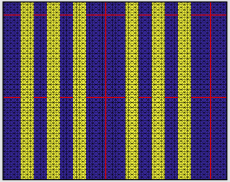
Problem configuration (Heinlein, Klawonn, Lanser (2022))

p-Laplacian problem with $p = 4$ and a **binary coefficient** α:

find u such that

$$
\begin{aligned}
-\alpha\Delta_p u &= 1 && \text{in } \Omega,\\
u &= 0 && \text{on } \partial\Omega.
\end{aligned}
$$

Domain decomposition into 6×6 subdomains with $H/h = 32$ and overlap $1h$.



yellow: $\alpha = 10^3$ **blue:** $\alpha = 1$

size cp	method	coarse space	outer it.	local it. (avg.)	coarse it.	GMRES it. (sum)
no globalization						
145	H1-RASPEN	AGDSW	5	27.0	35	77
25	H1-RASPEN	MsFEM-D	>20	-	-	-
25	H1-RASPEN	MsFEM-E	>20	-	-	-
145	NK-RAS	AGDSW	>20	-	-	-
inexact Newton backtracking (INB); cf. Eisenstat and Walker (1994)						
145	H1-RASPEN	AGDSW	5	24.8	21	77
25	H1-RASPEN	MsFEM-D	**15**	**75.8**	**62**	**645**
25	H1-RASPEN	MsFEM-E	**18**	**83.9**	**75**	**852**
145	NK-RAS	AGDSW	**13**	-	-	207

Coarse spaces

- **MsFEM-D:** MsFEM with linear edge trace
- **MsFEM-E:** MsFEM with energy-minimizing edge trace
- **AGDSW:** adaptive GDSW

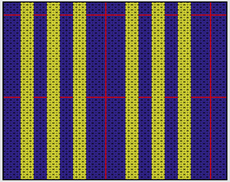
(Extensions using the first linearization)

Problem configuration (Heinlein, Klawonn, Lanser (2022))

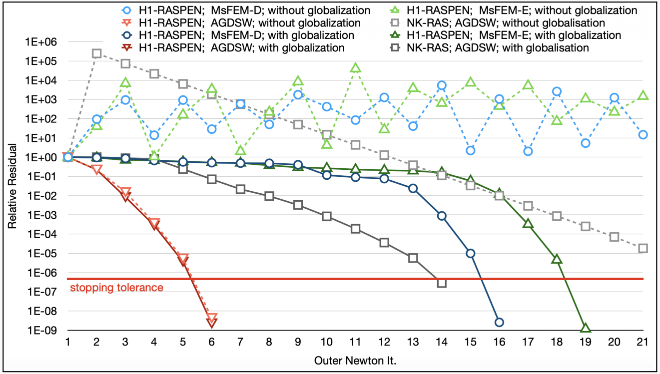
p-Laplacian problem with $p = 4$ and a **binary coefficient** α:
find u such that

$$\begin{aligned} -\alpha\Delta_p u &= 1 && \text{in } \Omega, \\ u &= 0 && \text{on } \partial\Omega. \end{aligned}$$

Domain decomposition into 6×6 subdomains with $H/h = 32$ and overlap $1h$.



yellow: $\alpha = 10^3$ **blue:** $\alpha = 1$

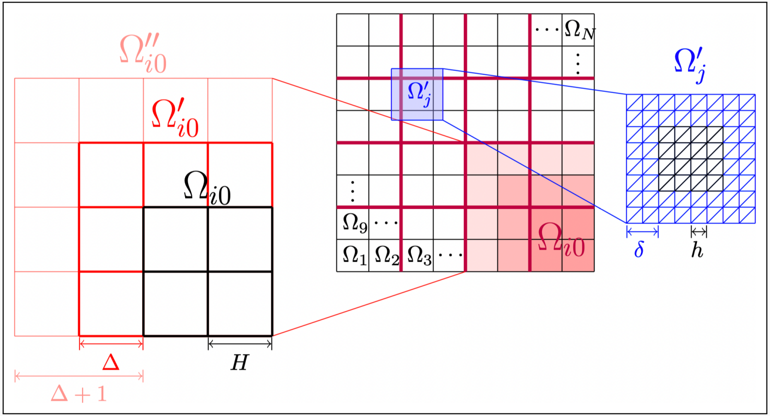


Coarse spaces

- **MsFEM-D:** MsFEM with linear edge trace
- **MsFEM-E:** MsFEM with energy-minimizing edge trace
- **AGDSW:** adaptive GDSW

(Extensions using the first linearization)

Three-Level Nonlinear Schwarz Operators



Domain decomposition hierarchy

Introduction of a domain decomposition into **subregions** consisting of several subdomains.

Recursive construction

Construction of **nonlinear Schwarz operator** on subregion level for the **nonlinear coarse problem**.

$\rightarrow$ Can be extended to an **arbitrary number of levels**

$\rightarrow$ **Improved parallelism** due to **reduction of the global nonlinear coarse problem**

Nonlinear Three-Level Schwarz Preconditioners

Two-level (R)ASPEN

Local/coarse corrections $T_i(\boldsymbol{u})$:

$$\boldsymbol{R}_i \boldsymbol{F}(\boldsymbol{u} - \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u})) = 0, \;\; i = 0, 1, ..., N, \;\; \text{with}$$

$$\text{restrictions } \boldsymbol{R}_i : V \to V_i,$$

$$\text{prolongations } \boldsymbol{P}_i : V_i \to V.$$

Nonlinear two-level ASPEN problem:

$$\mathcal{F}_A(\boldsymbol{u}) := \boldsymbol{P}_0 \boldsymbol{T}_0(\boldsymbol{u}) + \sum\nolimits_{i=1}^{N} \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u}) = 0$$

We solve $\mathcal{F}_A(\boldsymbol{u}) = 0$ using Newton's method with $\boldsymbol{u}_i = \boldsymbol{u} - \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u})$. The Jacobian writes

$$D\mathcal{F}_{RA}(\boldsymbol{u}) = \overbrace{P_0 \left(\boldsymbol{R}_0 D\boldsymbol{F}(\boldsymbol{u}_0) \boldsymbol{P}_0 \right)^{-1} \boldsymbol{R}_0 \;\; D\boldsymbol{F}(\boldsymbol{u}_0)}^{\text{coarse Schwarz operator}}$$

$$+ \sum_{i=1}^{N} \underbrace{P_i \left(\boldsymbol{R}_i D\boldsymbol{F}(\boldsymbol{u}_i) \boldsymbol{P}_i \right)^{-1} \boldsymbol{R}_i \;\; D\boldsymbol{F}(\boldsymbol{u}_i)}_{\text{local Schwarz operators}}$$

Cf. Heinlein & Lanser (2020).

Three-level (R)ASPEN

Corrections $\boldsymbol{T}_i(\boldsymbol{u})$ and $\boldsymbol{T}_{j0}(\boldsymbol{u})$:

$$\boldsymbol{R}_i \boldsymbol{F}(\boldsymbol{u} - \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u})) = 0, \;\; i = 1, ..., N,$$

$$\boldsymbol{R}_{j0} \boldsymbol{F}(\boldsymbol{u} - \boldsymbol{P}_{j0} \boldsymbol{T}_{j0}(\boldsymbol{u})) = 0, \;\; j = 0, ..., M, \;\; \text{with}$$

$$\text{restrictions } \boldsymbol{R}_i : V \to V_i, \boldsymbol{R}_{j0} : V \to V_{j0},$$

$$\text{prolongations } \boldsymbol{P}_i : V_i \to V, \boldsymbol{P}_{j0} : V_{j0} \to V.$$

Nonlinear three-level ASPEN problem:

$$\mathcal{F}_{AA}(\boldsymbol{u}) := \sum\nolimits_{j=0}^{M} \boldsymbol{P}_{j0} \boldsymbol{T}_{j0}(\boldsymbol{u}) + \sum\nolimits_{i=1}^{N} \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u}) = 0$$

We solve $\mathcal{F}_A(\boldsymbol{u}) = 0$ using Newton's method with $\boldsymbol{u}_i = \boldsymbol{u} - \boldsymbol{P}_i \boldsymbol{T}_i(\boldsymbol{u})$.
Again, the **Jacobian is of the form of a linear three-level Schwarz preconditioner**; we skip the full expression due to its length.

Cf. Heinlein, Klawonn, and Lanser (in preparation).

Results for p-Laplacian model problem

	additive		multiplicative 1	
	2-level	**3-level**	**2-level**	**3-level**
linear	M_A^{-1}	M_{AA}^{-1}	M_M^{-1}	M_{MM}^{-1}
nonlinear	$\mathcal{F}_A$	$\mathcal{F}_{AA}$	$\mathcal{F}_M$	$\mathcal{F}_{MM}$

$p = 4$; 16^2 subd.; 4^2 subr.; $H/h = 8$; overlap $\delta = 1$

method	nonlin.				lin.
	outer it.	**subd. it. (avg./min/max)**	**subr. it. (avg./min/max)**	**coarse it.**	**GMRES it. (sum)**
$\mathcal{F}_A$	6	23/16/46	-	36	90
$\mathcal{F}_M$	5	11/10/25	-	34	60
M_A^{-1}	24	-	-	-	381
M_M^{-1}	24	-	-	-	335
$\mathcal{F}_{AA}$	6	25/17/47	25/20/39	35	108
$\mathcal{F}_{MM}$	7	17/15/40	19/16/33	34	92
$\mathcal{F}_{MM}^{(BFGS)}$	6	15/13/35	15/9/45	31	79
M_{AA}^{-1}	24	-	-	-	396
M_{MM}^{-1}	24	-	-	-	338

Three-level (R)ASPEN

Corrections $T_i(u)$ and $T_{j0}(u)$:

$$R_i F(u - P_i T_i(u)) = 0, \; i = 1, \ldots, N,$$

$$R_{j0} F(u - P_{j0} T_{j0}(u)) = 0, \; j = 0, \ldots, M, \; \text{with}$$

$$\text{restrictions } R_i : V \to V_i, R_{j0} : V \to V_{j0},$$

$$\text{prolongations } P_i : V_i \to V, P_{j0} : V_{j0} \to V.$$

Nonlinear three-level ASPEN problem:

$$\mathcal{F}_{AA}(u) := \sum_{j=0}^{M} P_{j0} T_{j0}(u) + \sum_{i=1}^{N} P_i T_i(u) = 0$$

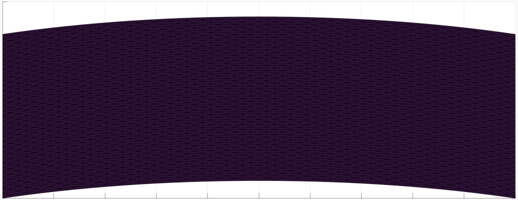
We solve $\mathcal{F}_A(u) = 0$ using Newton's method with $u_i = u - P_i T_i(u)$.

Again, the **Jacobian is of the form of a linear three-level Schwarz preconditioner**; we skip the full expression due to its length.

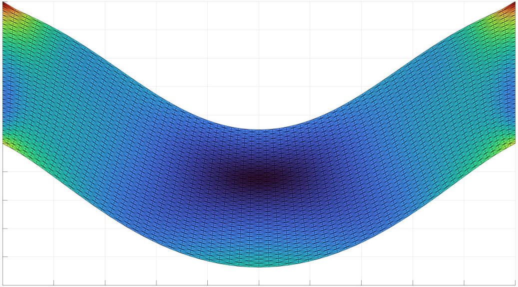
Cf. Heinlein, Klawonn, and Lanser (in preparation).

Results for Neo–Hookean Hyperelasticity

Undeformed geometry



Deformed geometry



Cf. **Heinlein, Klawonn, and Lanser (in preparation)**.

Homogeneous Neo-Hooke

outer its (GMRES its) with increasing load

$p = 4$; $H/h = 8$; overlap $\delta = 1$

16×16 subdomains; 4×4 subregions

method	**load** 2	5	10	12.5	15
$\mathcal{F}_A$	3 (102)	4 (146)	4 (154)	4 (156)	5 (203)
$\mathcal{F}_M$	3 (90)	3 (91)	3 (96)	3 (98)	3 (103)
$\boldsymbol{M}_A^{-1}$	4 (119)	5 (155)	6 (198)	7 (250)	fail
$\boldsymbol{M}_M^{-1}$	4 (103)	5 (135)	6 (166)	7 (214)	fail
$\mathcal{F}_{AA}$	4 (155)	4 (160)	5 (220)	5 (224)	5 (228)
$\mathcal{F}_{MM}$	3 (96)	4 (136)	4 (134)	4 (136)	4 (140)
$\mathcal{F}_{MM}^{(BFGS)}$	3 (96)	4 (136)	4 (134)	4 (136)	4 (140)
$\boldsymbol{M}_{AA}^{-1}$	4 (147)	5 (188)	6 (236)	7 (293)	fail
$\boldsymbol{M}_{MM}^{-1}$	4 (117)	5 (150)	6 (188)	7 (236)	fail

coarse space includes lin. rotations

Summary

- FROSch is based on the **Schwarz framework** and **energy-minimizing coarse spaces**, which provide **numerical scalability** using **only algebraic information** for a **variety of applications** including **nonlinear multi-physics problems**
- For nonlinear problems,
  - **the reuse of components of the preconditioner** and
  - **the speedup of the solver phase (e.g., using GPUs)**

  can significantly help to improve the solver performance.
- **Nonlinear Schwarz preconditioners** offer further improvements by **reducing the number of global Newton iterations**.
- **Scalability** of nonlinear Schwarz methods can be **improved** by **adding one or multiple coarse levels**.

Acknowledgements

- **Financial support:** DFG (KL2094/3-1, RH122/4-1), DFG SPP 2311 project number 465228106
- **Computing resources:** Summit (OLCF), Cori (NERSC), magnitUDE (UDE), Piz Daint (CSCS), Fritz (FAU)

Thank you for your attention!

→ Plenary talk by Martin Lanser on Monday, 8.45 am: *Robust nonlinear two-level domain decomposition methods*