

ROBUST OVERLAPPING SCHWARZ METHODS AND THEIR APPLICATIONS

Pierre Jolivet — Sorbonne Université, CNRS, LIP6

29th International Conference on DDM, June 27, 2025

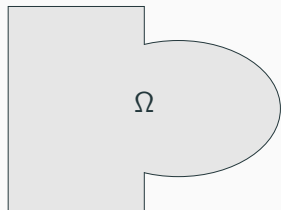
<https://joliv.et/DD29.pdf>

OVERVIEW

- journey since DD21
- emphasis on usability and interoperability
- some current challenges

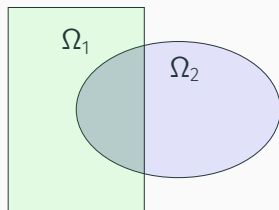
OVERLAPPING SCHWARZ METHODS

- global linear system $Ax = b \in \mathbb{K}^n$



OVERLAPPING SCHWARZ METHODS

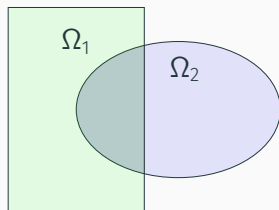
- global linear system $Ax = b \in \mathbb{K}^n$
- 2-way surjection of $\llbracket 1; n \rrbracket \rightsquigarrow$ restriction operators



OVERLAPPING SCHWARZ METHODS

- global linear system $Ax = b \in \mathbb{K}^n$
- 2-way surjection of $\llbracket 1; n \rrbracket \rightsquigarrow$ restriction operators

$$M_{\text{ASM}}^{-1} = \sum_{i=1}^{N=2} R_i^T (R_i A R_i^T)^{-1} R_i$$

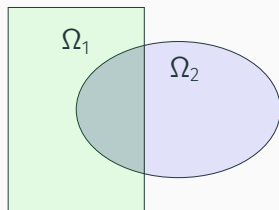


[Schwarz 1870]

OVERLAPPING SCHWARZ METHODS

- global linear system $Ax = b \in \mathbb{K}^n$
- 2-way surjection of $\llbracket 1; n \rrbracket \rightsquigarrow$ restriction operators

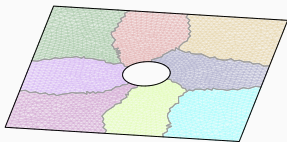
$$M_{\text{ASM}}^{-1} = \sum_{i=1}^{N=2} R_i^T (R_i A R_i^T)^{-1} R_i$$
$$M_{\text{RAS}}^{-1} = \sum_{i=1}^{N=2} \tilde{R}_i^T (R_i A R_i^T)^{-1} R_i$$



[Schwarz 1870] [Cai and Sarkis 1999]

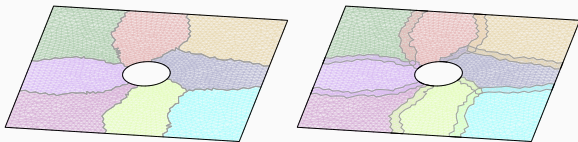
ONE-LEVEL PRECONDITIONERS

- geometric overlap



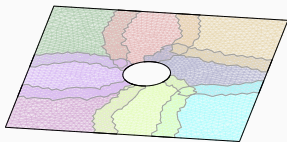
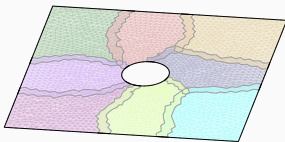
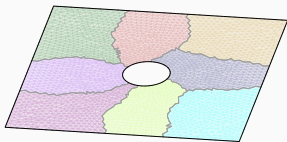
ONE-LEVEL PRECONDITIONERS

- geometric overlap



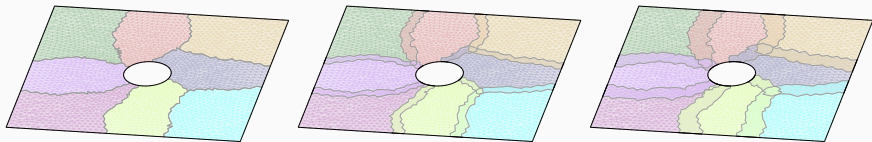
ONE-LEVEL PRECONDITIONERS

- geometric overlap

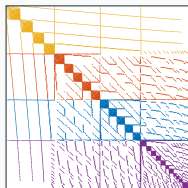


ONE-LEVEL PRECONDITIONERS

- geometric overlap

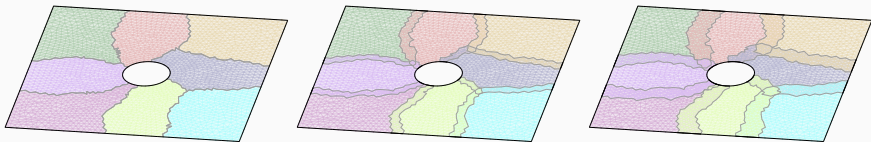


- algebraic overlap

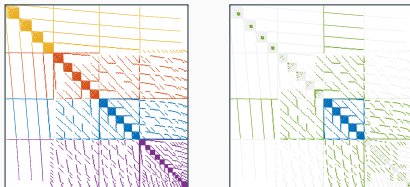


ONE-LEVEL PRECONDITIONERS

- geometric overlap

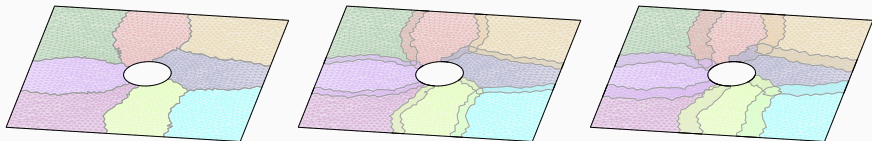


- algebraic overlap

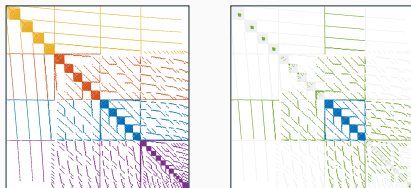


ONE-LEVEL PRECONDITIONERS

- geometric overlap



- algebraic overlap



→ no guaranteed scalability for large subdomain counts

COARSE CORRECTION

GenEO in a nutshell [Spillane et al. 2013]

- solve concurrently $N_i \Lambda_k = \lambda_k \tilde{R}_i A \tilde{R}_i^T \Lambda_k$

COARSE CORRECTION

GenEO in a nutshell [Spillane et al. 2013]

- solve concurrently $N_i \Lambda_k = \lambda_k \tilde{R}_i A \tilde{R}_i^T \Lambda_k$
- define locally $W_i = \begin{bmatrix} \Lambda_1 & \cdots & \Lambda_{\gamma_i} \end{bmatrix}$

COARSE CORRECTION

GenEO in a nutshell [Spillane et al. 2013]

- solve concurrently $N_i \Lambda_k = \lambda_k \tilde{R}_i A \tilde{R}_i^T \Lambda_k$
- define locally $W_i = \begin{bmatrix} \Lambda_1 & \cdots & \Lambda_{\gamma_i} \end{bmatrix}$
- use $Z = \begin{bmatrix} \tilde{R}_1^T W_1 & \cdots & \tilde{R}_N^T W_N \end{bmatrix}$ to compute $A_C = Z^T A Z$

COARSE CORRECTION

GenEO in a nutshell [Spillane et al. 2013]

- solve concurrently $N_i \Lambda_k = \lambda_k \tilde{R}_i A \tilde{R}_i^T \Lambda_k$
- define locally $W_i = \begin{bmatrix} \Lambda_1 & \cdots & \Lambda_{\gamma_i} \end{bmatrix}$
- use $Z = \begin{bmatrix} \tilde{R}_1^T W_1 & \cdots & \tilde{R}_N^T W_N \end{bmatrix}$ to compute $A_C = Z^T A Z$
- coarse correction $M_{\text{additive}}^{-1} = Z A_C^{-1} Z^T + M_{\text{ASM}}^{-1}$

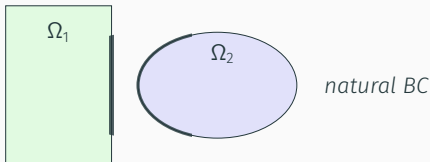
COARSE CORRECTION

GenEO in a nutshell [Spillane et al. 2013]

- solve concurrently $N_i \Lambda_k = \lambda_k \tilde{R}_i A \tilde{R}_i^T \Lambda_k$
- define locally $W_i = \begin{bmatrix} \Lambda_1 & \cdots & \Lambda_{\gamma_i} \end{bmatrix}$
- use $Z = \begin{bmatrix} \tilde{R}_1^T W_1 & \cdots & \tilde{R}_N^T W_N \end{bmatrix}$ to compute $A_C = Z^T A Z$
- coarse correction $M_{\text{additive}}^{-1} = Z A_C^{-1} Z^T + M_{\text{ASM}}^{-1}$

How to choose N_i ?

- unassembled matrices



COARSE CORRECTION

GenEO in a nutshell [Spillane et al. 2013]

- solve concurrently $N_i \Lambda_k = \lambda_k \tilde{R}_i A \tilde{R}_i^T \Lambda_k$
- define locally $W_i = \begin{bmatrix} \Lambda_1 & \cdots & \Lambda_{\gamma_i} \end{bmatrix}$
- use $Z = \begin{bmatrix} \tilde{R}_1^T W_1 & \cdots & \tilde{R}_N^T W_N \end{bmatrix}$ to compute $A_C = Z^T A Z$
- coarse correction $M_{\text{additive}}^{-1} = Z A_C^{-1} Z^T + M_{\text{ASM}}^{-1}$

How to choose N_i ?

- unassembled matrices
- local SPSP splitting [Al Daas and Grigori 2019]

COARSE CORRECTION

GenEO in a nutshell [Spillane et al. 2013]

- solve concurrently $N_i \Lambda_k = \lambda_k \tilde{R}_i A \tilde{R}_i^T \Lambda_k$
- define locally $W_i = \begin{bmatrix} \Lambda_1 & \cdots & \Lambda_{\gamma_i} \end{bmatrix}$
- use $Z = \begin{bmatrix} \tilde{R}_1^T W_1 & \cdots & \tilde{R}_N^T W_N \end{bmatrix}$ to compute $A_C = Z^T A Z$
- coarse correction $M_{\text{additive}}^{-1} = Z A_C^{-1} Z^T + M_{\text{ASM}}^{-1}$

How to choose γ_i and $\{\Lambda_j\}_{j=1,\dots,\gamma_i}$?

- aggressive coarsening
- capturing the “difficulties” of A
- at a moderate cost

- implementation of the required machinery for GenEO
- initially interfaced in FreeFEM [Jolivet et al. 2013]
- applicability of the library for:
 - dealing with systems with multiple right-hand sides [Jolivet and Tournier 2016] (also B. Martin@MS01)
 - optimized Schwarz methods [Haferssas et al. 2017]
 - preconditioning hierarchical matrices [Marchand et al. 2020] (also P. Marchand@MS14)

INTEGRATION IN PETSc [JOLIVET ET AL. 2021]

Application Codes

Higher-Level Libraries and Frameworks



TS
Time Steppers

Pseudo-Transient, Runge-Kutta, IMEX, SSP...
Local and Global Error Estimators
Adaptive Timestepping
Event Handling
Sensitivity via Adjoints
...

 **TAO**
Optimization Solvers

PDE-Constrained
Adjoint-Based
Derivative-Free

Levenberg-Marquardt
Newton's Method
Interior Point Methods
...

SNES
Nonlinear Solvers

Newton LineSearch Successive Substitution
Newton Trust Region Nonlinear CG
BFGS (Quasi-Newton) Active Set VI
Nonlinear Gauss-Seidel
...

KSP
Linear Solvers

CG, GMRES, BiCGStab, FGMRES, LSQR...
Pipelined Krylov Methods
Hierarchical Krylov Methods
Block and Batched Krylov Methods
...

 **DM**
Domain Management

 **DMDA**
Regular Grids
 **DMstag**
Staggered Grids

 **DMPlex**
Unstructured Meshes
 **DMNetwork**
Networks

 **DMForest**
Forest-of-trees AMR
 **DMSwarm**
Particles

PC
Preconditioners

(Incomplete) LU, (Incomplete) Cholesky, QR
Overlapping Schwarz, BDDC, FETI-DP
Fieldsplit (Block Preconditioners)
PCMG (Geometric Multigrid), PCPATCH
GAMG (Algebraic Multigrid), HYPRE
...

Vec
Vectors

IS
Index Sets

Mat
Linear Operators

AIJ (Compressed Sparse Row)
BAIJ (Blocked)
SBAIJ (Symmetric)
Dense, Hierarchical, Shell (MF)
GPU-specific
...

PetscSF
Parallel Communication

Communication and Computational Kernels

MPI

BLAS/LAPACK

Kokkos

CUDA

HIP

ScaLAPACK

SLEPc
Eigensolvers

WHY PETSc?

- easier comparison with the state-of-the-art
- large user-base (and potential HPDDM users)
- leverage PETSc and SLEPc infrastructures
- Stefano Zampini was very pressing (at DD24 and SIAM CSE19)

SELECTED OVERVIEW OF THE API

- PCHPDDMSetAuxiliaryMat()
 - LHS of local GEVP (Neumann matrices)

SELECTED OVERVIEW OF THE API

- PCHPDDMSetAuxiliaryMat()
 - LHS of local GEVP (Neumann matrices)
- PCHPDDMSetRHSMat()
 - RHS of local GEVP (computed automatically)
 - different coarse spaces for Helmholtz [Bootland et al. 2021]

SELECTED OVERVIEW OF THE API

- PCHPDDMSetAuxiliaryMat()
 - LHS of local GEVP (Neumann matrices)
- PCHPDDMSetRHSMat()
 - RHS of local GEVP (computed automatically)
 - different coarse spaces for Helmholtz [Bootland et al. 2021]
- PCHPDDMSetDeflationMat()
 - supply the local matrices W_i directly (no GEVP)

SELECTED OVERVIEW OF THE API

- PCHPDDMSetAuxiliaryMat()
 - LHS of local GEVP (Neumann matrices)
- PCHPDDMSetRHSMat()
 - RHS of local GEVP (computed automatically)
 - different coarse spaces for Helmholtz [Bootland et al. 2021]
- PCHPDDMSetDeflationMat()
 - supply the local matrices W_i directly (no GEVP)
- PCHPDDMSetCoarseCorrectionType(), $Q = ZA_C^{-1}Z^T$

$$M_{\text{additive}}^{-1} = Q + M_{\star}^{-1}$$

$$M_{\text{balanced}}^{-1} = Q + (I - AQ)^T M_{\star}^{-1} (I - AQ)$$

$$M_{\text{deflated}}^{-1} = Q + M_{\star}^{-1} (I - AQ)$$

BACK TO GENEO

- one does not know γ_i a priori
- needed by most generalized eigensolver
- threshold criterion τ instead for low eigenmodes

BACK TO GENEO

- one does not know γ_i a priori
- needed by most generalized eigensolver
- threshold criterion τ instead for low eigenmodes

Inertia

- compute the inertia of $N_i - \tau \tilde{R}_i A \tilde{R}_i^T$
- precisely γ_i
- bound on the condition number of $M^{-1}A$

BACK TO GENEO

- one does not know γ_i a priori
- needed by most generalized eigensolver
- threshold criterion τ instead for low eigenmodes

Inertia

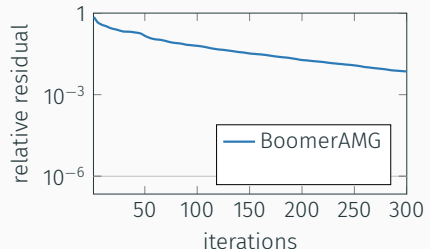
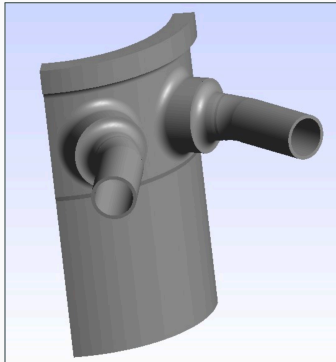
- compute the inertia of $N_i - \tau \tilde{R}_i A \tilde{R}_i^T$
- precisely γ_i
- bound on the condition number of $M^{-1}A$

Custom stopping criterion

- provide an upper-bound for γ_i
- SLEPc will automatically stop iterating when $\lambda_i > \tau$

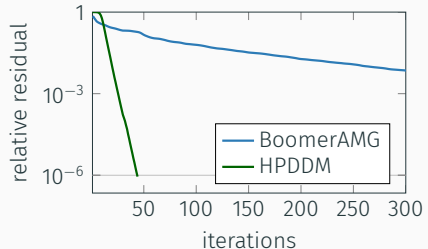
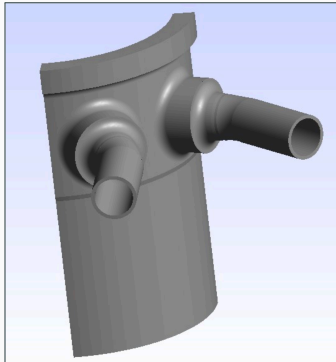
APPLICATION TO SOLID MECHANICS...

- thermo-elasticity at EDF
- code_aster discretization - “legacy” code (N. Tardieu@MSHI)
 - with ghost elements
- 68M d.o.f., 1.2k MPI processes



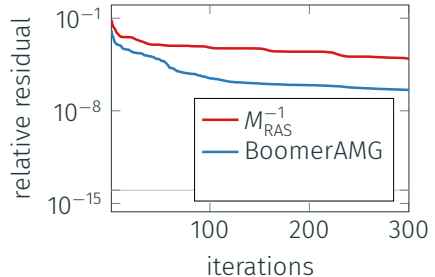
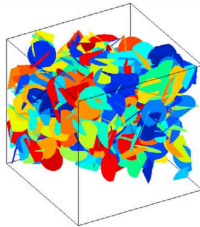
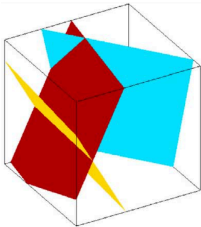
APPLICATION TO SOLID MECHANICS...

- thermo-elasticity at EDF
- code_aster discretization - “legacy” code (N. Tardieu@MSHI)
 - with ghost elements
- 68M d.o.f., 1.2k MPI processes
- 1.0007 GC and 1.002 OC (1 min setup, 20 sec solve)



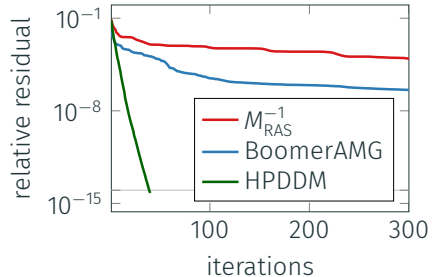
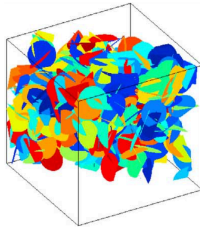
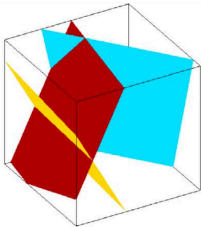
...AND TO FLOW IN POROUS MEDIA [JOLIVET ET AL. 2025]

- lowest-order Raviart–Thomas for Darcy’s law
- discrete fracture matrix model from Inria SERENA
- 140M d.o.f., 4k MPI processes



...AND TO FLOW IN POROUS MEDIA [JOLIVET ET AL. 2025]

- lowest-order Raviart–Thomas for Darcy’s law
- discrete fracture matrix model from Inria SERENA
- 140M d.o.f., 4k MPI processes
- 1.002 GC and 1.2 OC (80 sec setup, 20 sec solve)



ALGEBRAIC PRECONDITIONING

- unassembled local matrices and ghost elements

ALGEBRAIC PRECONDITIONING

- unassembled local matrices and ghost elements
- sparse least-squares

ALGEBRAIC PRECONDITIONING

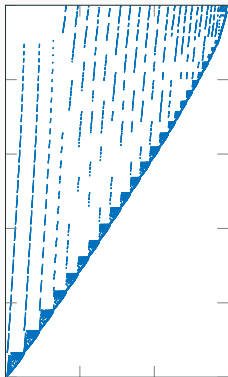
- unassembled local matrices and ghost elements
- sparse least-squares
 - LSQR [Paige and Saunders 1982]
 - normal equations $\rightarrow$ preconditioner for $A^T A$

ALGEBRAIC PRECONDITIONING

- unassembled local matrices and ghost elements
- sparse least-squares
 - LSQR [Paige and Saunders 1982]
 - normal equations $\rightarrow$ preconditioner for $A^T A$
- general sparse problems

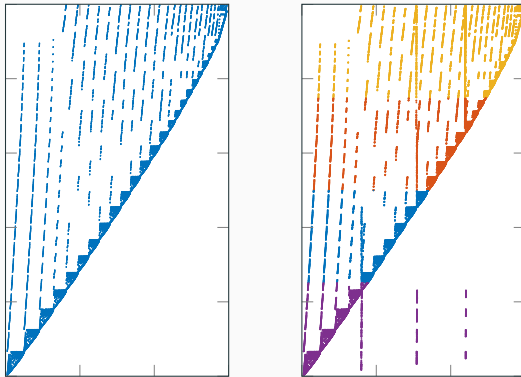
ACADEMIC EXAMPLE

mk13-b5 from the SSMC



ACADEMIC EXAMPLE

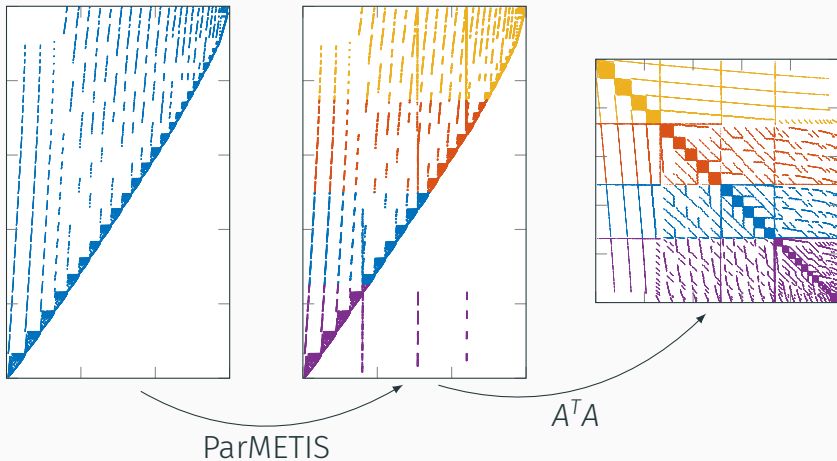
mk13-b5 from the SSMC on 4 processes



ParMETIS

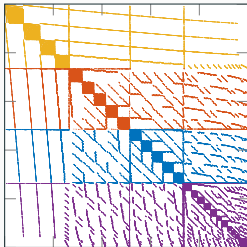
ACADEMIC EXAMPLE

mk13-b5 from the SSMC on 4 processes



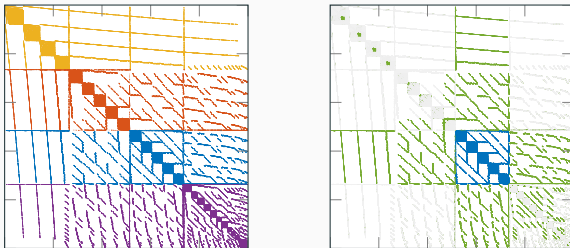
PRECONDITIONING $A^T A$

- AMG
- (incomplete) Cholesky



PRECONDITIONING $A^T A$

- AMG
- (incomplete) Cholesky
- domain decomposition



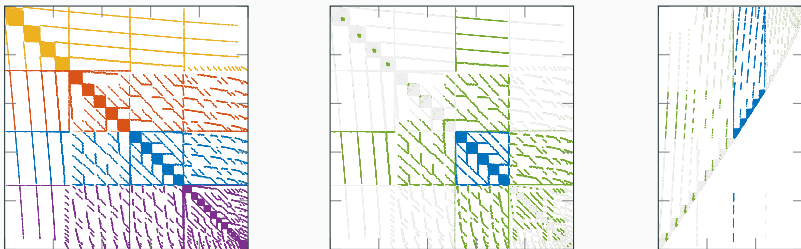
subdomain #3 + one overlap layer

$$M_{\text{RAS}}^{-1} = \sum_{i=1}^{N=4} \tilde{R}_i^T (R_i A^T A R_i^T)^{-1} R_i$$

Cholesky decompositions

PRECONDITIONING $A^T A$

- AMG
- (incomplete) Cholesky
- domain decomposition

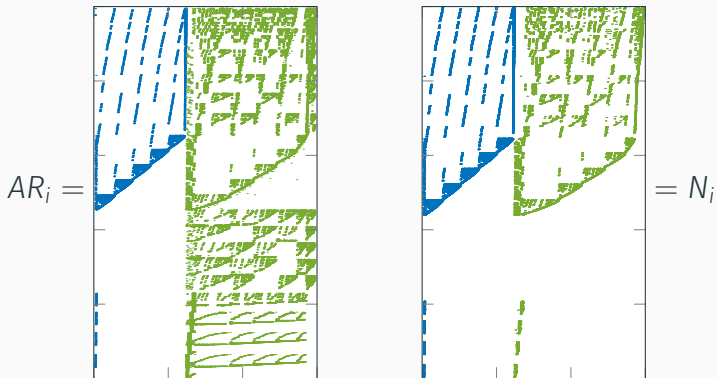


subdomain #3 + one overlap layer

$$M_{\text{RAS}}^{-1} = \sum_{i=1}^{N=4} \tilde{R}_i^T (R_i A^T A R_i^T)^{-1} R_i$$

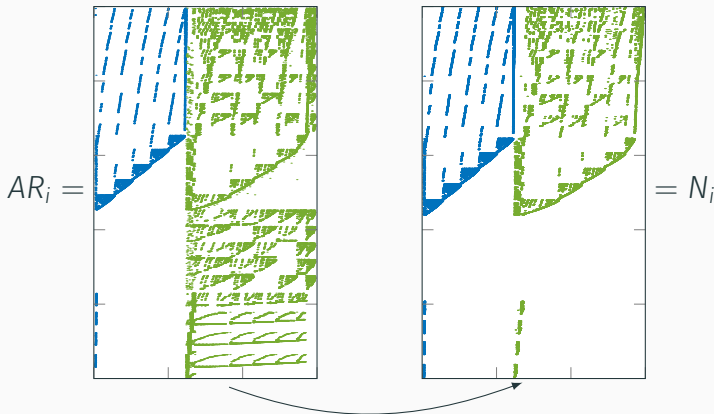
QR decompositions

SPSD SPLITTING



zero rows on the overlap that are zero in the subdomain

SPSD SPLITTING



zero rows **on the overlap** that are zero **in the subdomain**

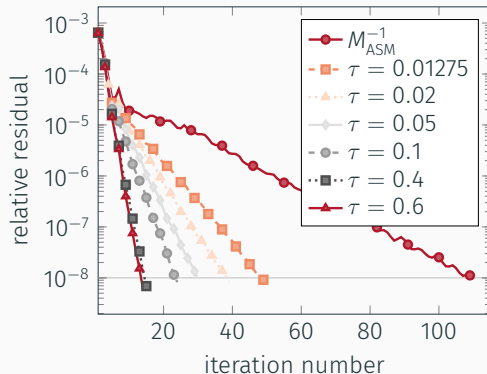
- solve concurrently $N_i^T N_i \Lambda_k = \lambda_k \tilde{R}_i A^T \tilde{A} \tilde{R}_i^T \Lambda_k$
- stable decomposition [Al Daas et al. 2022]

NUMERICAL RESULTS ON 256 PROCESSES

Identifier	M_{balanced}^{-1}	BoomerAMG	GAMG
mesh_deform	13		35
EternityII_E	43		63
lp_stocfor3	34		513
deltaX	23		784
sc205-2r	54		195
stormg2-125	42		
Rucci1	21	118	364
image_interp	11	40	203
mk13-b5	19	11	
pds-100	18	16	35
fome21	20	16	20
sgpf5y6	224		163
Hardesty2	30	88	404
Delor338K	10		
watson_2	15		64
LargeRegFile	41	19	
cont11_l	30	53	723

ADAPTIVE PRECONDITIONING

watson_2 from the SSMC on 256 processes



τ	n_C	Iterations
0.01275	2,400	49
0.02	2,683	39
0.05	3,049	30
0.1	3,337	24
0.4	8,979	15
0.6	30,246	14

GENERAL SPARSE PROBLEMS [AL DAAS ET AL. 2025]

- δ (resp. 1) layers of overlap (resp. oversampling)

$$R_i^T A R_i = A_{ii} = \begin{pmatrix} A_{l_i, l_i} & A_{l_i, \Gamma_{i, 1:\delta}} & 0_{l_i, |\Gamma_{i, \delta+1}|} \\ A_{\Gamma_{i, 1:\delta}, l_i} & A_{\Gamma_{i, 1:\delta}, \Gamma_{i, 1:\delta}} & A_{\Gamma_{i, 1:\delta}, \Gamma_{i, \delta+1}} \\ 0_{|\Gamma_{i, \delta+1}|, l_i} & A_{\Gamma_{i, \delta+1}, \Gamma_{i, 1:\delta}} & A_{\Gamma_{i, \delta+1}, \Gamma_{i, \delta+1}} \end{pmatrix}$$

GENERAL SPARSE PROBLEMS [AL DAAS ET AL. 2025]

- δ (resp. 1) layers of overlap (resp. oversampling)

$$R_i^T A R_i = A_{ii} = \begin{pmatrix} A_{l_i, l_i} & A_{l_i, \Gamma_{i,1:\delta}} & 0_{l_i, |\Gamma_{i,\delta+1}|} \\ A_{\Gamma_{i,1:\delta}, l_i} & A_{\Gamma_{i,1:\delta}, \Gamma_{i,1:\delta}} & A_{\Gamma_{i,1:\delta}, \Gamma_{i,\delta+1}} \\ 0_{|\Gamma_{i,\delta+1}|, l_i} & A_{\Gamma_{i,\delta+1}, \Gamma_{i,1:\delta}} & A_{\Gamma_{i,\delta+1}, \Gamma_{i,\delta+1}} \end{pmatrix}$$

- local harmonic projection

$$\Pi_i = \begin{pmatrix} -A_{\Omega_{i,1:\delta}, \Omega_{i,1:\delta}}^{-1} A_{\Omega_{i,1:\delta}, \Gamma_{i,\delta+1}} \\ I_{|\Gamma_{i,\delta+1}|} \end{pmatrix} \begin{pmatrix} 0_{|\Gamma_{i,\delta+1}|, |\Omega_{i,1:\delta}|} & I_{|\Gamma_{i,\delta+1}|} \end{pmatrix}$$

GENERAL SPARSE PROBLEMS [AL DAAS ET AL. 2025]

- δ (resp. 1) layers of overlap (resp. oversampling)

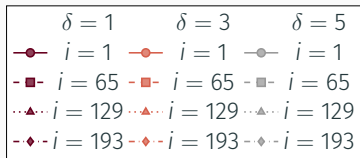
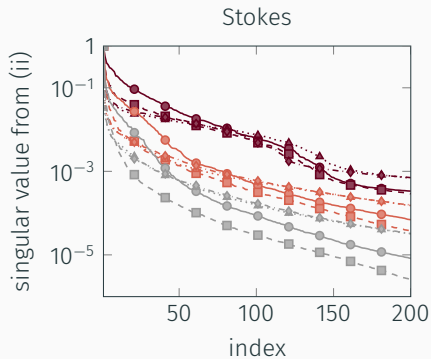
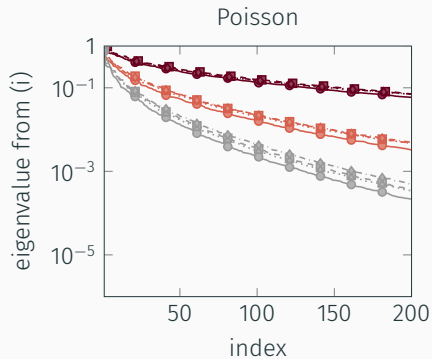
$$R_i^T A R_i = A_{ii} = \begin{pmatrix} A_{l_i, l_i} & A_{l_i, \Gamma_{i,1:\delta}} & 0_{l_i, |\Gamma_{i,\delta+1}|} \\ A_{\Gamma_{i,1:\delta}, l_i} & A_{\Gamma_{i,1:\delta}, \Gamma_{i,1:\delta}} & A_{\Gamma_{i,1:\delta}, \Gamma_{i,\delta+1}} \\ 0_{|\Gamma_{i,\delta+1}|, l_i} & A_{\Gamma_{i,\delta+1}, \Gamma_{i,1:\delta}} & A_{\Gamma_{i,\delta+1}, \Gamma_{i,\delta+1}} \end{pmatrix}$$

- local harmonic projection

$$\Pi_i = \begin{pmatrix} -A_{\Omega_{i,1:\delta}, \Omega_{i,1:\delta}}^{-1} A_{\Omega_{i,1:\delta}, \Gamma_{i,\delta+1}} \\ l_{|\Gamma_{i,\delta+1}|} \end{pmatrix} \begin{pmatrix} 0_{|\Gamma_{i,\delta+1}|, |\Omega_{i,1:\delta}|} & l_{|\Gamma_{i,\delta+1}|} \end{pmatrix}$$

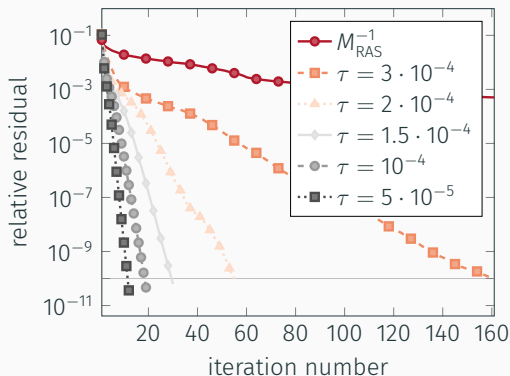
- for SPD A : eigenvalue problem $\tilde{\Pi}_i^T A_{ii} \tilde{\Pi}_i w = \lambda A_{ii} w$ (i)
- otherwise: singular value problem $\tilde{\Pi}_i^T \tilde{\Pi}_i u = \sigma^2 u$ (ii)

DECAY OF EIGENVALUES/SINGULAR VALUES



APPLICATION TO LINEARIZED R13 EQUATIONS

- discretization in FEniCS [Theisen and Torrilhon 2021]
- monolithic algebraic preconditioning
- 16M d.o.f., 1024 MPI processes



τ	n_C	Iterations
$3 \cdot 10^{-4}$	319,082	161
$2 \cdot 10^{-4}$	354,934	55
$1.5 \cdot 10^{-4}$	381,646	30
$1 \cdot 10^{-4}$	420,191	19
$5 \cdot 10^{-5}$	493,631	12

- `-pc_fieldsplit_schur_fact_type full`

$$A = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix}$$

- `-pc_fieldsplit_schur_fact_type full`

$$A = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix}$$

$$A^{-1} = U^{-1}D^{-1}L^{-1} = \begin{bmatrix} I & -A_{00}^{-1}A_{01} \\ 0 & I \end{bmatrix} \begin{bmatrix} A_{00}^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -A_{10}A_{00}^{-1} & I \end{bmatrix}$$

SADDLE-POINT SYSTEMS [NATAF AND TOURNIER 2022]

- `-pc_fieldsplit_schur_fact_type full`

$$A = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix}$$

$$A^{-1} = U^{-1}D^{-1}L^{-1} = \begin{bmatrix} I & -A_{00}^{-1}A_{01} \\ 0 & I \end{bmatrix} \begin{bmatrix} A_{00}^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -A_{10}A_{00}^{-1} & I \end{bmatrix}$$

- setup HPDDM on the top-left block: $A_{00}^{-1} \approx M_{00,\text{additive}}^{-1}$

SADDLE-POINT SYSTEMS [NATAF AND TOURNIER 2022]

- `-pc_fieldsplit_schur_fact_type full`

$$A = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix}$$

$$A^{-1} = U^{-1}D^{-1}L^{-1} = \begin{bmatrix} I & -A_{00}^{-1}A_{01} \\ 0 & I \end{bmatrix} \begin{bmatrix} A_{00}^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -A_{10}A_{00}^{-1} & I \end{bmatrix}$$

- setup HPDDM on the top-left block: $A_{00}^{-1} \approx M_{00,\text{additive}}^{-1}$
- approximate $S \approx \tilde{S} = A_{11} - A_{10}M_{00,\text{additive}}^{-1}A_{01}$

SADDLE-POINT SYSTEMS [NATAF AND TOURNIER 2022]

- `-pc_fieldsplit_schur_fact_type full`

$$A = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix}$$

$$A^{-1} = U^{-1}D^{-1}L^{-1} = \begin{bmatrix} I & -A_{00}^{-1}A_{01} \\ 0 & I \end{bmatrix} \begin{bmatrix} A_{00}^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -A_{10}A_{00}^{-1} & I \end{bmatrix}$$

- setup HPDDM on the top-left block: $A_{00}^{-1} \approx M_{00,\text{additive}}^{-1}$
- approximate $S \approx \tilde{S} = A_{11} - A_{10}M_{00,\text{additive}}^{-1}A_{01}$
- write $\tilde{S} = S_{11} - A_{10}Q_{00}A_{01}$ and build $M_{11,\text{balanced}}^{-1}$ for S_{11}

SADDLE-POINT SYSTEMS [NATAF AND TOURNIER 2022]

- `-pc_fieldsplit_schur_fact_type full`

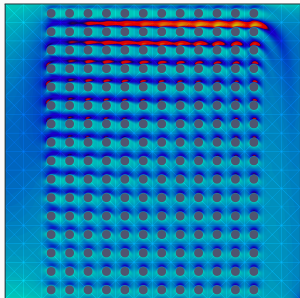
$$A = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix}$$

$$A^{-1} = U^{-1}D^{-1}L^{-1} = \begin{bmatrix} I & -A_{00}^{-1}A_{01} \\ 0 & I \end{bmatrix} \begin{bmatrix} A_{00}^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -A_{10}A_{00}^{-1} & I \end{bmatrix}$$

- setup HPDDM on the top-left block: $A_{00}^{-1} \approx M_{00,\text{additive}}^{-1}$
- approximate $S \approx \tilde{S} = A_{11} - A_{10}M_{00,\text{additive}}^{-1}A_{01}$
- write $\tilde{S} = S_{11} - A_{10}Q_{00}A_{01}$ and build $M_{11,\text{balanced}}^{-1}$ for S_{11}
- solve inner systems with $I - M_{11,\text{balanced}}^{-1}A_{10}Q_{00}A_{01}$

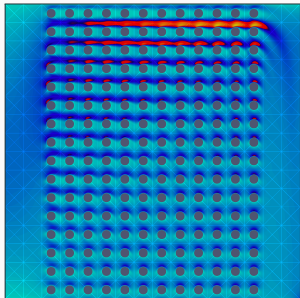
APPLICATION TO FLUID DYNAMICS

- Oseen equations: $-\rho \mathbf{V} \cdot \nabla \mathbf{u} = -\nabla p + \mu \Delta \mathbf{u}$
 $-\nabla \cdot \mathbf{u} = 0$
- solution in a perforated domain from CEA
- 10^{-10} outer tolerance, 100M d.o.f., 1k MPI processes



APPLICATION TO FLUID DYNAMICS

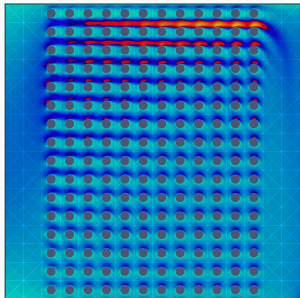
- Oseen equations: $-\rho \mathbf{V} \cdot \nabla \mathbf{u} = -\nabla p + \mu \Delta \mathbf{u}$
 $-\nabla \cdot \mathbf{u} = 0$
- solution in a perforated domain from CEA
- 10^{-10} outer tolerance, 100M d.o.f., 1k MPI processes



$\mu = 0.1$	outer solves	A_{00} solves	S solves
GenEO	5	50	8
harmonic	5	22	7

APPLICATION TO FLUID DYNAMICS

- Oseen equations: $-\rho \mathbf{V} \cdot \nabla \mathbf{u} = -\nabla p + \mu \Delta \mathbf{u}$
 $-\nabla \cdot \mathbf{u} = 0$
- solution in a perforated domain from CEA
- 10^{-10} outer tolerance, 100M d.o.f., 1k MPI processes

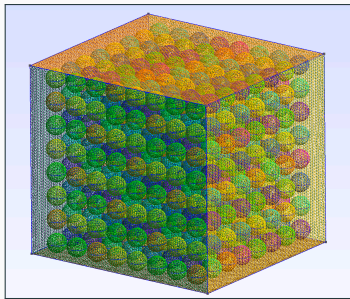
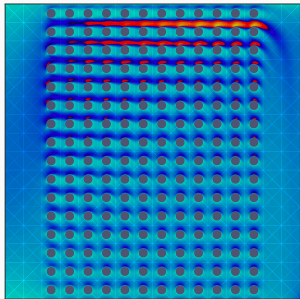


$\mu = 0.1$	outer solves	A_{00} solves	S solves
GenEO	5	50	8
harmonic	5	22	7

5×10^{-4}	outer solves	A_{00} solves	S solves
GenEO	†	†	†
harmonic	6	14	6

APPLICATION TO FLUID DYNAMICS

- Oseen equations: $-\rho \mathbf{V} \cdot \nabla \mathbf{u} = -\nabla p + \mu \Delta \mathbf{u}$
 $-\nabla \cdot \mathbf{u} = 0$
- solution in a perforated domain from CEA
- 10^{-10} outer tolerance, 100M d.o.f., 1k MPI processes



SOME CURRENT CHALLENGES...

- other types of saddle-point systems
 - different discretization

SOME CURRENT CHALLENGES...

- other types of saddle-point systems
 - different discretization
 - contact mechanics

SOME CURRENT CHALLENGES...

- other types of saddle-point systems
 - different discretization
 - contact mechanics
 - casting S as a least-square problem

$$S \approx A_{10} \text{diag}(A_{00})^{-\frac{1}{2}} \text{diag}(A_{00})^{-\frac{1}{2}} A_{01}$$

SOME CURRENT CHALLENGES...

- other types of saddle-point systems
 - different discretization
 - contact mechanics
 - casting S as a least-square problem

$$S \approx A_{10} \text{diag}(A_{00})^{-\frac{1}{2}} \text{diag}(A_{00})^{-\frac{1}{2}} A_{01}$$

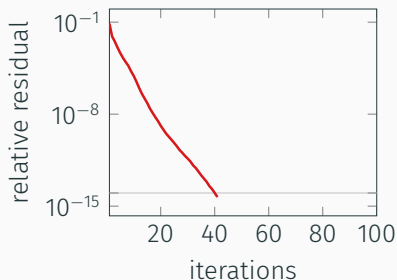
- matrix-free (C. Guillet@MS16)

...LOWER-PRECISION FACTORIZATIONS...

- impact of mixed-precision (F. Nataf@MS21)
 1. subdomain solvers
 2. coarse solver

...LOWER-PRECISION FACTORIZATIONS...

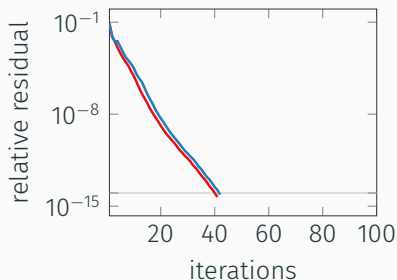
- impact of mixed-precision (F. Nataf@MS21)
 1. subdomain solvers
 2. coarse solver
- DFM model from Inria SERENA



— 1. double, 2. double

...LOWER-PRECISION FACTORIZATIONS...

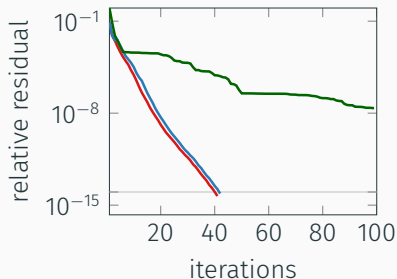
- impact of mixed-precision (F. Nataf@MS21)
 1. subdomain solvers
 2. coarse solver
- DFM model from Inria SERENA



— 1. double, 2. double — 1. double, 2. float

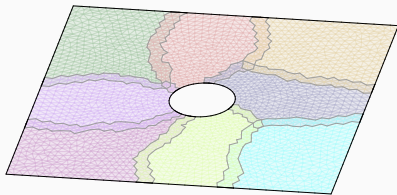
...LOWER-PRECISION FACTORIZATIONS...

- impact of mixed-precision (F. Nataf@MS21)
 1. subdomain solvers
 2. coarse solver
- DFM model from Inria SERENA

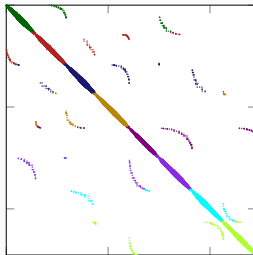


— 1. double, 2. double — 1. double, 2. float — 1. float, 2. float

...MORE EFFICIENT MULTILEVEL

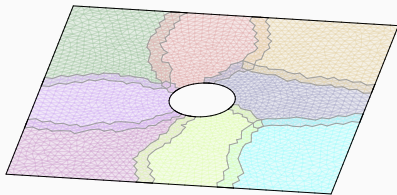


fine-level decomposition

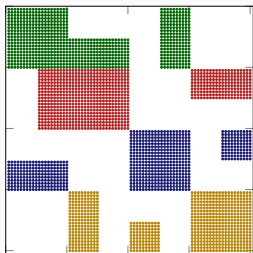


$A^{(1)}$

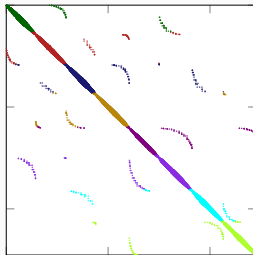
...MORE EFFICIENT MULTILEVEL



fine-level decomposition

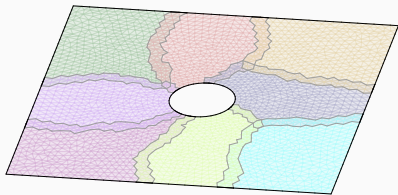


$$A^{(2)} = Z^{(1)T} A^{(1)} Z^{(1)}$$

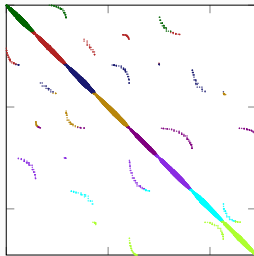


$A^{(1)}$

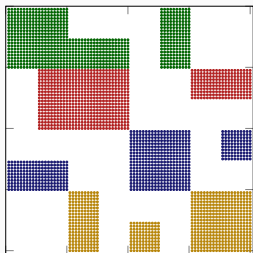
...MORE EFFICIENT MULTILEVEL



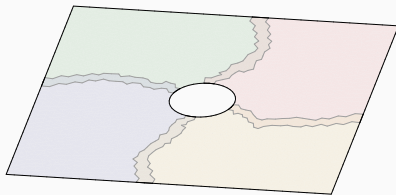
fine-level decomposition



$A^{(1)}$

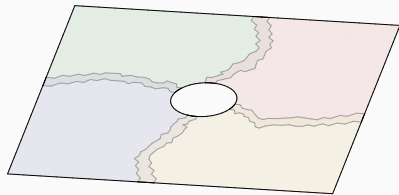


$$A^{(2)} = Z^{(1)T} A^{(1)} Z^{(1)}$$

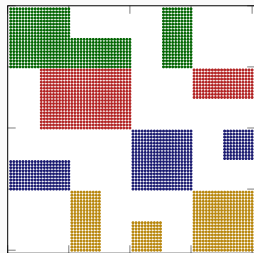


second-level rank aggregation

...MORE EFFICIENT MULTILEVEL

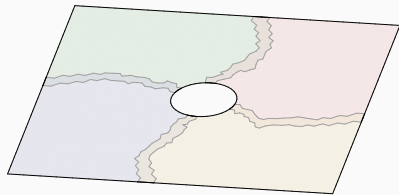


$$N_i^{(2)} = \sum_{j \in \text{aggregate } \#i} W_j^{(1)T} N_j^{(1)} W_j^{(1)}$$

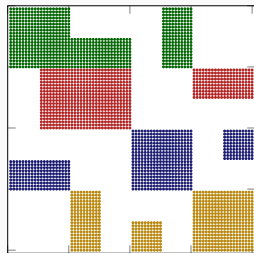


algebraic $\tilde{R}_i^{(2)} A^{(2)} \tilde{R}_i^{(2)T}$

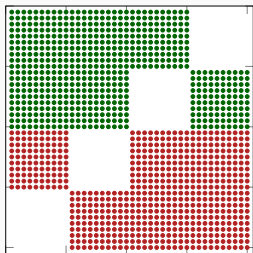
...MORE EFFICIENT MULTILEVEL



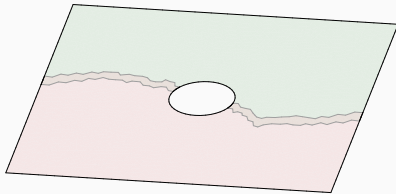
$$N_i^{(2)} = \sum_{j \in \text{aggregate } \#i} W_j^{(1)T} N_j^{(1)} W_j^{(1)}$$



$$\text{algebraic } \tilde{R}_i^{(2)} A^{(2)} \tilde{R}_i^{(2)T}$$



$$A^{(3)} = Z^{(2)T} A^{(2)} Z^{(2)}$$



FINAL WORDS

- crucial link between discretization and preconditioning

FINAL WORDS

- crucial link between discretization and preconditioning
- but not always possible

FINAL WORDS

- crucial link between discretization and preconditioning
- but not always possible
- leverage the flexibility of algebraic backbone

FINAL WORDS

- crucial link between discretization and preconditioning
- but not always possible
- leverage the flexibility of algebraic backbone
- our goal is not to overwhelm you with choices

FINAL WORDS

- crucial link between discretization and preconditioning
- but not always possible
- leverage the flexibility of algebraic backbone
- our goal is not to overwhelm you with choices

Thank you!

REFERENCES I



Al Daas, Hussam and Laura Grigori (2019). **“A class of efficient locally constructed preconditioners based on coarse spaces”**. In: *SIAM Journal on Matrix Analysis and Applications* 40.1, pp. 66–91.



Al Daas, Hussam, Pierre Jolivet, Frédéric Nataf, and Pierre-Henri Tournier (2025). **“A robust two-level Schwarz preconditioner for sparse matrices”**. In: *SIAM Journal on Scientific Computing*, accepted for publication.



Al Daas, Hussam, Pierre Jolivet, and Jennifer A. Scott (2022). **“A robust algebraic domain decomposition preconditioner for sparse normal equations”**. In: *SIAM Journal on Scientific Computing* 44.3, A1047–A1068. URL: <https://github.com/prj-/aldaas2021robust>.



Bootland, Niall, Victorita Dolean, Pierre Jolivet, and Pierre-Henri Tournier (2021). **“A comparison of coarse spaces for Helmholtz problems in the high-frequency regime”**. In: *Computers & Mathematics with Applications* 98, pp. 239–253.



Cai, Xiao-Chuan and Marcus Sarkis (1999). **“A Restricted Additive Schwarz Preconditioner for General Sparse Linear Systems”**. In: *SIAM Journal on Scientific Computing* 21.2, pp. 792–797.

REFERENCES II



Haferssas, Ryadh, Pierre Jolivet, and Frédéric Nataf (2017). **“An additive Schwarz method type theory for Lions’s algorithm and a symmetrized optimized restricted additive Schwarz method”**. In: *SIAM Journal on Scientific Computing* 39.4, A1345–A1365.



Jolivet, Pierre, Frédéric Hecht, Frédéric Nataf, and Christophe Prud’homme (2013). **“Scalable Domain Decomposition Preconditioners for Heterogeneous Elliptic Problems”**. In: *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, SC13*. ACM.



Jolivet, Pierre, Michel Kern, Frédéric Nataf, Géraldine Pichot, and Daniel Zegarra Vasquez (2025). **“Domain decomposition preconditioners for efficient parallel simulations of single-phase flow in three-dimensional fractured porous media with a very large number of fractures”**. In: working paper or preprint. URL: <https://inria.hal.science/hal-05029676>.

REFERENCES III



Jolivet, Pierre, Jose E. Roman, and Stefano Zampini (2021). **“KSPHPDDM and PCHPDDM: extending PETSc with advanced Krylov methods and robust multilevel overlapping Schwarz preconditioners”**. In: *Computers & Mathematics with Applications* 84, pp. 277–295. URL: <https://github.com/prj-/jolivet2020petsc>.



Jolivet, Pierre and Pierre-Henri Tournier (2016). **“Block iterative methods and recycling for improved scalability of linear solvers”**. In: *Proceedings of the 2016 International Conference for High Performance Computing, Networking, Storage and Analysis*. SC16. IEEE.



Marchand, Pierre, Xavier Claeys, Pierre Jolivet, Frédéric Nataf, and Pierre-Henri Tournier (2020). **“Two-level preconditioning for h -version boundary element approximation of hypersingular operator with GenEO”**. In: *Numerische Mathematik* 146, pp. 597–628.



Nataf, Frédéric and Pierre-Henri Tournier (2022). **“Recent advances in domain decomposition methods for large-scale saddle point problems”**. In: *Comptes Rendus. Mécanique*.

REFERENCES IV



Paige, Christopher C. and Michael A. Saunders (1982). “**LSQR: an algorithm for sparse linear equations and sparse least squares**”. In: *ACM Transactions on Mathematical Software* 8.1, pp. 43–71.



Schwarz, Hermann (1870). “**Über einen Grenzübergang durch alternierendes Verfahren**”. In: *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich* 15, pp. 272–286.



Spillane, Nicole, Victorita Dolean, Patrice Hauret, Frédéric Nataf, Clemens Pechstein, and Robert Scheichl (2013). “**Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps**”. In: *Numerische Mathematik* 126.4, pp. 741–770.



Theisen, Lambert and Manuel Torrilhon (2021). “**fenicsR13: A Tensorial Mixed Finite Element Solver for the Linear R13 Equations Using the FEniCS Computing Platform**”. In: *ACM Transaction on Mathematical Software* 47.2.