INEXACT COARSE SOLVERS FOR ADAPTIVE PRECONDITIONERS USING BLOCK RECYCLED ITERATIVE METHODS

Pierre Jolivet, IRIT-CNRS

AN17, MS7
July 10

INTRODUCTION

Ubiquious components

- numerical robustness
- practical scalability

No silver bullet

- geometric grids
- spectral grids
- ...

COARSE GRID OPERATORS

Challenging issues

- MG: more levels means more communication
- DD, deflation: recursion is not always possible

- MG: more levels means more communication
- DD, deflation: recursion is not always possible

$\implies$ new hierarchies or inner solvers

- open-source
- `https://github.org/hpddm/hpddm`
- Python/C/C++/Fortran bindings
- efficient implementation of RAP for DDM
- block iterative methods and recycled Krylov solvers [Jolivet and Tournier 2016]

- any kind of M^{-1}/A functors, e.g., MatMatSolve/MatMatMult
- support left/right/variable preconditioning

Available methods

- BGMRES
- BGCRO-DR
- BCG
- BFBCG [Ji and Li 2016]

- any kind of M^{-1}/A functors, e.g., MatMatSolve/MatMatMult
- support left/right/variable preconditioning

Available methods

- BGMRES
- BGCRO-DR
- BCG
- BFBCG [Ji and Li 2016]

Enlarged Krylov methods

$$(i, j) \in [\![1; n]\!] \times [\![1; p]\!] \implies (B_p)_{i,j} = \begin{cases} b_i \text{ if } \left\lceil \frac{i}{p} \right\rceil (j-1) < i \leqslant \left\lceil \frac{i}{p} \right\rceil j \\ 0 \text{ otherwise} \end{cases}$$

WHY USE BLOCK ITERATIVE METHODS?

Advantages

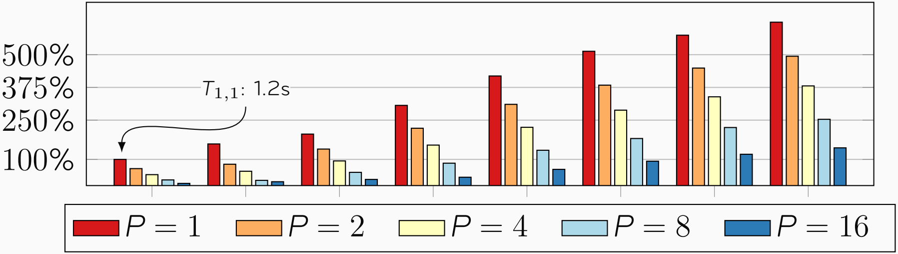
- larger Krylov subspace $\implies$ faster convergence
- higher arithmetic intensity
- fewer synchronizations with more data

WHY USE BLOCK ITERATIVE METHODS?

Advantages

- larger Krylov subspace $\implies$ faster convergence
- higher arithmetic intensity
- fewer synchronizations with more data

$$E_{P,p} = \frac{p \cdot T_{1,1}}{P \cdot T_{P,p}}$$



- any kind of M^{-1}/A functors, e.g., GAMG
- support left/right/variable preconditioning

Available method

- GCRO-DR [Parks et al. 2006]

PERFORMANCE OF RECYCLED KRYLOV SOLVERS

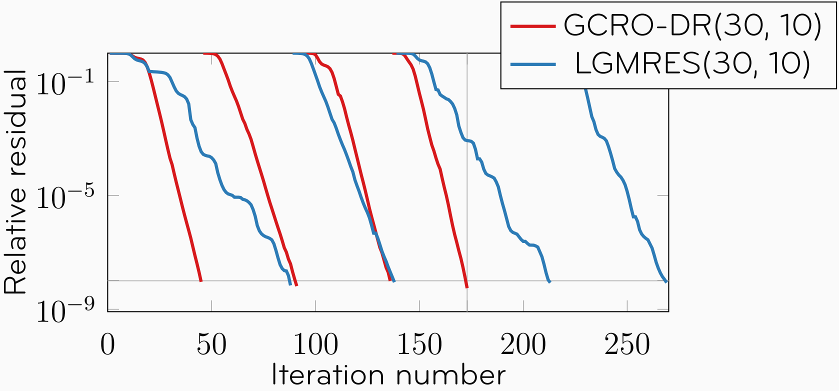
Comparison with Loose GMRES by [Baker et al. 2005]

```
mpirun -np 8000 ./ex56 -ne 399 -ksp_rtol 1e-8
  -ksp_type lgmres -ksp_pc_side right -pc_type gamg
  -ksp_lgmres_augment 10
```

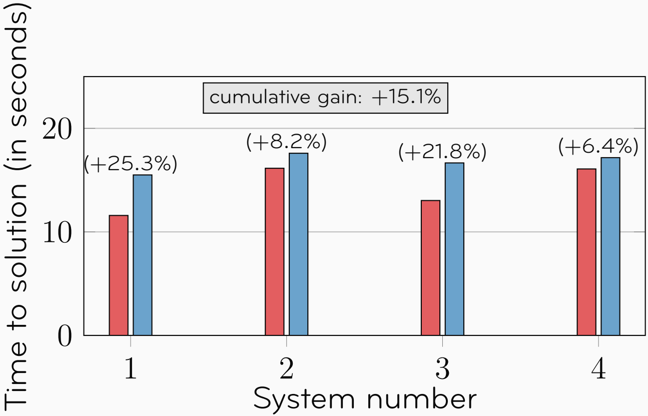
- small, moving inclusion (high contrast in E)
- assemble multiple linear systems/preconditioners

Comparison with Loose GMRES by [Baker et al. 2005]

```
mpirun -np 8000 ./ex56 -ne 399 -ksp_rtol 1e-8
  -ksp_type lgmres -ksp_pc_side right -pc_type gamg
  -ksp_lgmres_augment 10
```



PERFORMANCE OF RECYCLED KRYLOV SOLVERS



Maxwell's equation I

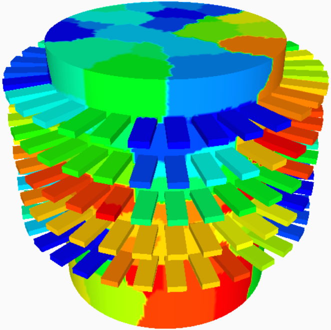
$$\nabla \times (\nabla \times \mathbf{E}) - \mu_0\Big(\omega^2\varepsilon + \mathrm{i}\omega\sigma\Big)\mathbf{E} = 0$$

$$\nabla \times (\nabla \times \mathbf{E}) - \mu_0 \left(\omega^2 \varepsilon + \mathrm{i}\omega\sigma \right) \mathbf{E} = 0$$

- domain decomposition preconditioner
- exact LDL^T local solvers

$$\mathcal{M}^{-1}_{\mathrm{ORAS}} = \sum_{i=1}^{N} R_i^T D_i B_i^{-1} R_i,$$

cf. [Gander 2006]



alternative	p	solve	# of it.	per RHS	eff.
GMRES	1				
GCRO-DR	1				

- $(m, k) = (50, 10)$ for solving 32 RHSs
- 2,048 subdomains and 2 threads per subdomain

alternative	p	solve	# of it.	per RHS	eff.
GMRES	1	3,078.4	20,068	627	–
GCRO-DR	1	1,836.9	10,701	334	1.7

- $(m, k) = (50, 10)$ for solving 32 RHSs
- 2,048 subdomains and 2 threads per subdomain

alternative	p	solve	# of it.	per RHS	eff.
GMRES	1	3,078.4	20,068	627	–
GCRO-DR	1	1,836.9	10,701	334	1.7
BGMRES	32	724.8	158	–	4.2

- $(m, k) = (50, 10)$ for solving 32 RHSs
- 2,048 subdomains and 2 threads per subdomain

alternative	p	solve	# of it.	per RHS	eff.
GMRES	1	3,078.4	20,068	627	–
GCRO-DR	1	1,836.9	10,701	334	1.7
BGMRES	32	724.8	158	–	4.2
BGCRO-DR	8	677.6	524	131	4.5

- $(m, k) = (50, 10)$ for solving 32 RHSs
- 2,048 subdomains and 2 threads per subdomain

Maxwell's equation II

alternative	p	solve	# of it.	per RHS	eff.
GMRES	1	3,078.4	20,068	627	–
GCRO-DR	1	1,836.9	10,701	334	1.7
BGMRES	32	724.8	158	–	4.2
BGCRO-DR	8	677.6	524	131	4.5
BGCRO-DR	32	992.3	127	–	3.1

- $(m, k) = (50, 10)$ for solving 32 RHSs
- 2,048 subdomains and 2 threads per subdomain

alternative	p	solve	# of it.	per RHS	eff.
GMRES	1	3,078.4	20,068	627	–
GCRO-DR	1	1,836.9	10,701	334	1.7
BGMRES	32	724.8	158	–	4.2
BGCRO-DR	8	677.6	524	131	4.5
BGCRO-DR	32	992.3	127	–	3.1

- $(m, k) = (50, 10)$ for solving 32 RHSs
- 2,048 subdomains and 2 threads per subdomain
- alternative #1 to #5 $\implies$ 158$\times$ fewer iterations

COARSE GRIDS FOR DDM

For overlapping Schwarz

- A, linear system
- R_i, restriction to subdomain i
- R_i^T, prolongation from subdomain i
- W_i, constraints/eigenvectors from subdomain i

RAP FOR DDM

For overlapping Schwarz

- A, linear system
- R_i, restriction to subdomain i
- R_i^T, prolongation from subdomain i
- W_i, constraints/eigenvectors from subdomain i

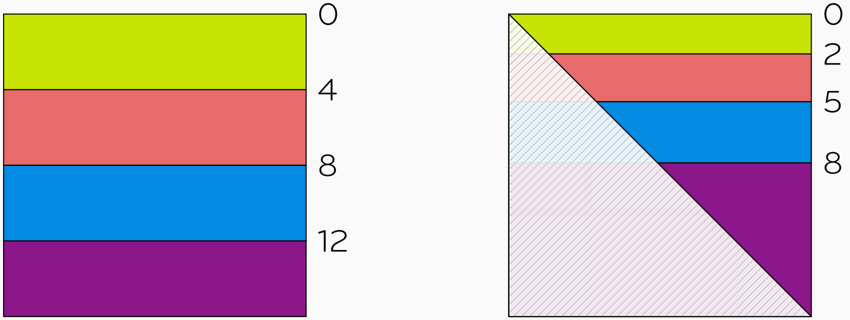
- $P = \begin{bmatrix} R_1^T W_1 & R_2^T W_2 & \cdots & R_N^T W_N \end{bmatrix}$
- $E = P^T A P$
- $E_{ij} = W_i^T R_i R_j^T A_{jj} W_j$

HOW TO HANDLE THE COARSE OPERATOR?

- direct solvers require an efficient redistribution
- multigrid methods may not converge

REDISTRIBUTION IN HPDDM

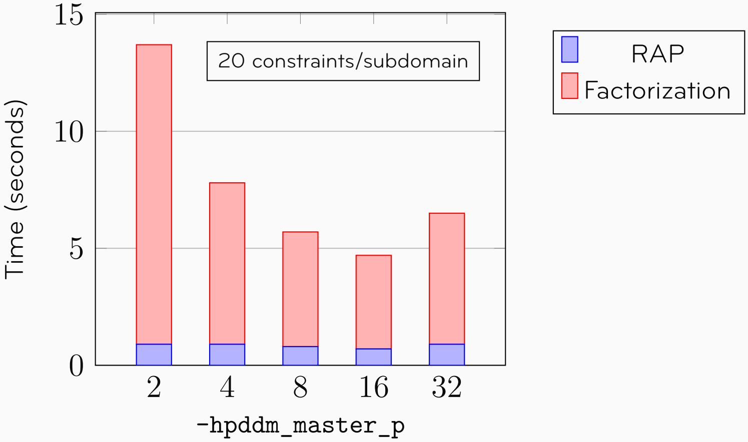
16 subdomains, `-hpddm_master_p 4` [Jolivet, Hecht, et al. 2013]



$\Longrightarrow$ scalability of direct solvers quickly attained

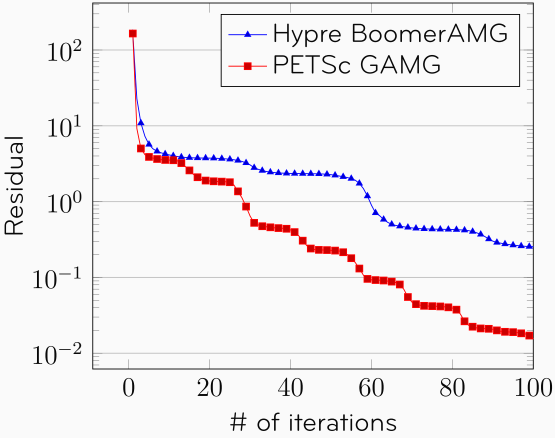
$\Longrightarrow$ scalability of direct solvers quickly attained

3D Stokes, $\mathbb{TH}_2$ FE, 145M d.o.f., 4,096 subdomains



$\Longrightarrow$ poor convergence with MG

3D Stokes, $\mathbb{TH}_2$ FE, 145M d.o.f., 4,096 subdomains

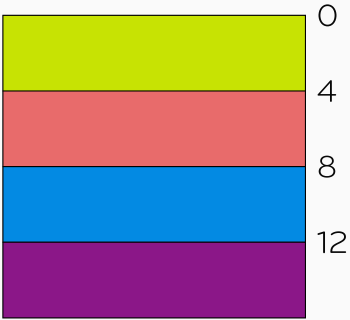


NEW TOOLS FOR COARSE OPERATORS

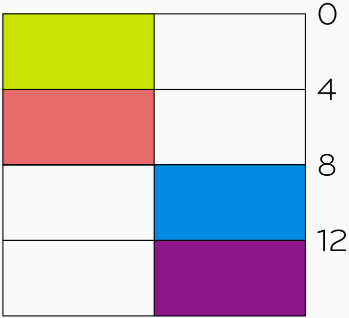
Inexact coarse grid solver

- mixed-precision
- block Jacobi with “fat” aggregates
- enlarge Krylov subspace + recycling (one solve for each outer iteration)

16 subdomains, `-hpddm_master_p 4`



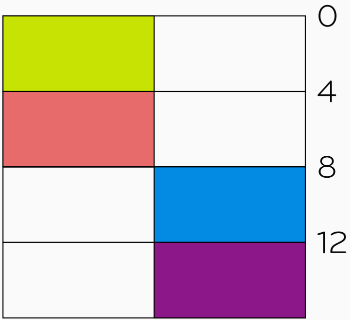
16 subdomains, `-hpddm_master_p 4`



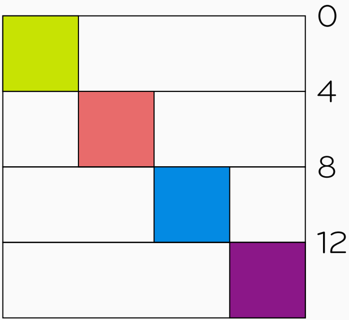
`-hpddm_master_aggregate_size 2`

Block Jacobi with "fat" aggregates

16 subdomains, `-hpddm_master_p 4`

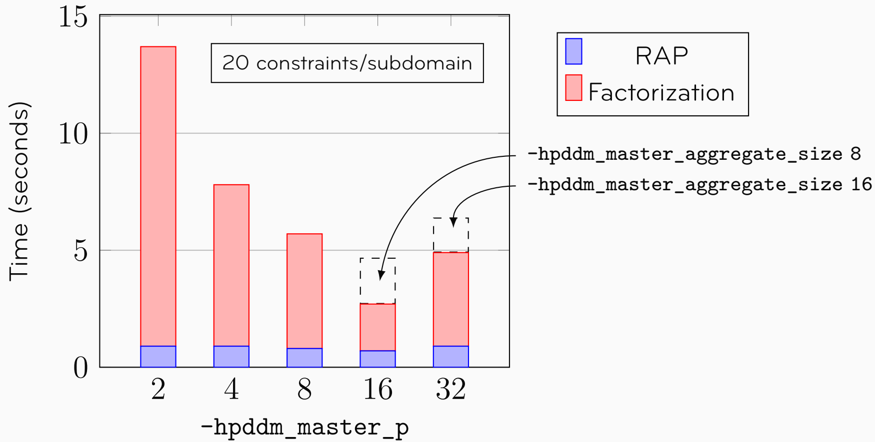


`-hpddm_master_aggregate_size 2`

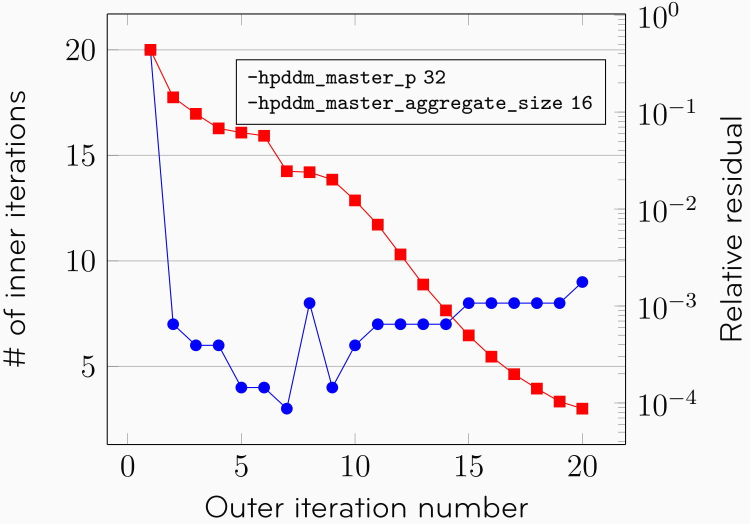


`-hpddm_master_aggregate_size 1`

3D Stokes, $\mathbb{TH}_2$ FE, 145M d.o.f., 4,096 subdomains



3D Stokes, $\mathbb{TH}_2$ FE, 145M d.o.f., 4,096 subdomains



Composing options

- coarse enlarged Krylov subspace
- recycled inner solver

FINAL WORDS

Summary:

- various strategies to improve coarse grid solvers
- applicable to most hierarchies

Summary:

- various strategies to improve coarse grid solvers
- applicable to most hierarchies

Thank you!

Baker, Allison H., Elizabeth R. Jessup, and Thomas Manteuffel (2005). "A Technique for Accelerating the Convergence of Restarted GMRES". In: *SIAM Journal on Matrix Analysis and Applications* 26.4, pp. 962–984.

Gander, Martin J. (2006). "Optimized Schwarz Methods". In: *SIAM Journal on Numerical Analysis* 44.2, pp. 699–731.

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 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.
- Parks, Michael L., Eric de Sturler, Greg Mackey, Duane D. Johnson, and Spandan Maiti (2006). "Recycling Krylov Subspaces for Sequences of Linear Systems". In: *SIAM Journal on Scientific Computing* 28.5, pp. 1651–1674.