

BDDC Preconditioners for Matrix-Free Higher-Order Methods

Clark Dohrmann
Sandia National Laboratories

Stefano Zampini
KAUST

**2019 SIAM Conference on
Computational Science & Engineering
Spokane, Washington
February 25-29, 2019**

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of two U.S. Department of Energy organizations (Office of Science and the National Nuclear Security Administration) responsible for the planning and preparation of a capable exascale ecosystem, including software, applications, hardware, advanced system engineering, and early testbed platforms, in support of the nation's exascale computing imperative..

- **Background:**
 - Some Basics
 - A case for BDDC*
 - Related work

- **Approach:**
 - Precondition easier problem
 - Allow for inexact subdomain solvers

- **Numerical Results**
 - Sparse direct versus AMG** subdomain solvers
 - Structured & unstructured meshes

- **Recap & Future Work**

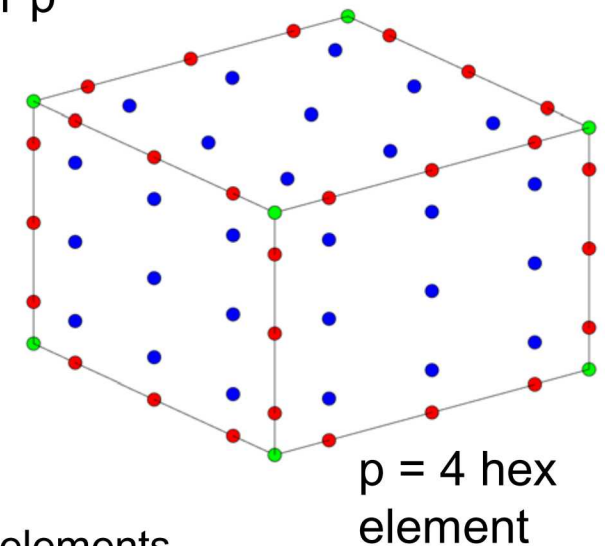
* Balancing Domain Decomposition by Constraints

** Algebraic MultiGrid

Background

■ Some Basics:

- Focus on 3D elements (degree p)
 - $O(p^3)$ degrees of freedom, $O(p^6)$ matrix entries
 - Matrix storage can be prohibitive for larger p
 - Interest in matrix-free approaches
- Basic approach:
 - $O(p^9)$ operations for element matrix
 - $O(p^6)$ for matrix-vector products
- Optimized approaches:
 - Sum factorization method (Orszag, 1980)
 - See refs for tensor product and simplicial elements
 - $O(p^6)$ operations for matrix (optimal)
 - $O(p^4)$ for matrix-vector products



Takeaway: Basic approach to calculate and store higher-order matrices can be prohibitive, but there are asymptotically more efficient approaches.

Background

- **A case for BDDC:**
 - Can isolate challenging computations to individual subdomains
 - Local work can be done independently of MPI communications
 - Coarse problem involves assembly of lower order elements
 - We know how to solve this problem
 - Existing theory available and promising numerical results

- **So what's the challenge?**
 - Problems at subdomain level involve large dense matrices, leading to memory & computational concerns

- **What to do?**
 - Develop BDDC preconditioner that avoids large dense matrices

Background

■ Related BDDC work*:

- Pavarino (2007):
 - Theory, 2D examples, exact solvers, single element subdomains
- Klawonn, Pavarino, Rheinbach (2008):
 - Theory, 2D focus, exact & inexact FETI-DP solvers
- Pavarino, Widlund, Zampini (2010)
 - Theory, 3D focus, almost incompressible elasticity, exact solvers
- Isogeometric analysis: BCPS13, BPSWZ14
- Bertoluzza, Pennacchio, Prada (2017)
 - Virtual element method, 2D only, theory for triangular elements

■ Condition number estimate:

$$\text{cond}(M^{-1}A) \leq C(1 + \log(p^2 H/h))^2$$

Note: $H/h = 1$ for single-element subdomains

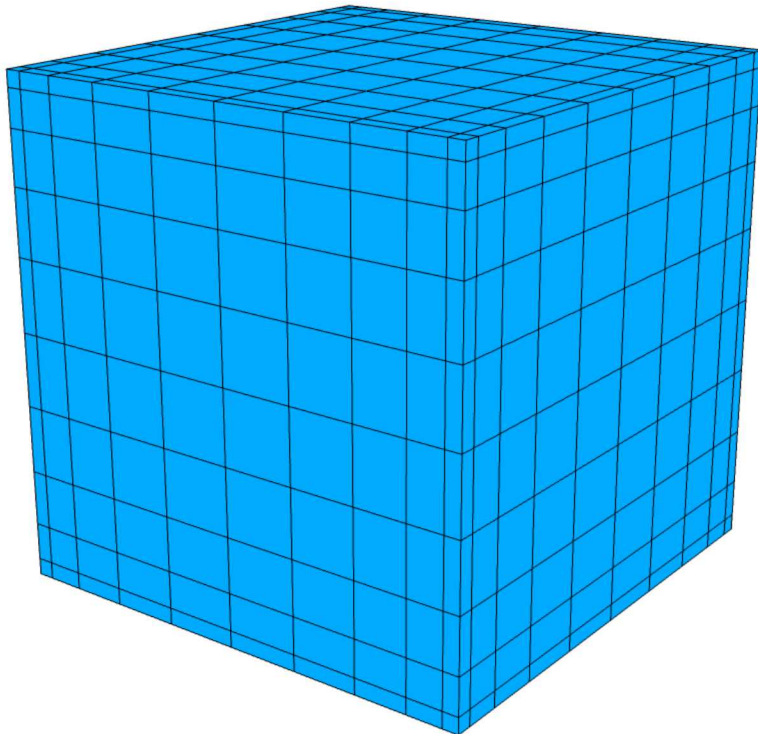
*see Chapter 7 of TW05 for a nice discussion of earlier non-BDDC work.

Previous work dealt directly with higher-order matrices.

Approach

- **How to avoid dense matrices?**

- Option 1: use old idea from Orszag (1980)
 - See Chapter 7 of TW05 for discussion and other applications



	K_p (dense)	K_h (sparse)
Memory	$O(p^6)$	$O(p^3)$
Factorization work	$O(p^9)$	$O(p^6)$
Factorization memory	$O(p^6)$	$O(p^{4/3})$
AMG initialization		$O(p^3)$
AMG memory		$O(p^3)$

Recall optimized matrix-vector products require $O(p^4)$ work

- Option 2: iteratively solve problems at subdomain level using matrix-free application of higher-order operator

Approach

- **How well does K_h approximate K_p ?**
 - Calculate min and max eigenvalues of $K_h x = \lambda K_p x$
 - Gauss-Lobatto integration for both K_p and K_h
 - Investigate effects of aspect ratio (pizza box elements)

	Aspect ratio 1			Aspect ratio 10			Aspect ratio 100		
p	λ_{min}	λ_{max}	ratio	λ_{min}	λ_{max}	ratio	λ_{min}	λ_{max}	ratio
2	0.42	2.25	5.3	0.42	2.25	5.3	0.42	2.25	5.3
3	0.49	2.39	4.9	0.49	2.68	5.5	0.49	2.75	5.6
4	0.50	2.51	5.0	0.50	2.77	5.6	0.50	2.98	6.0
5	0.50	2.59	5.2	0.50	2.84	5.7	0.50	3.10	6.2
6	0.49	2.64	5.4	0.49	2.89	5.9	0.49	3.16	6.4

Poisson equation

- K_h does good job approximating K_p
- weak dependence on aspect ratio
- no need for very accurate subdomain solvers.

Approach

■ Overview:

- Construct BDDC preconditioner M^{-1} for A_h
 - A_h obtained from assembly of K_h matrices
 - Each K_h matrix (or assembly thereof) can be treated as its own subdomain
- Use either sparse direct solver or AMG preconditioner for local Dirichlet and Neumann problems of BDDC
- Care needed for inexact solution of Neumann problems
 - Neumann problems used to construct coarse matrices
 - Inexact solvers should satisfy a null space property (see D07)
- Solve linear system $A_p x = b$ using a Krylov method with M^{-1} as the preconditioner

Numerical Results

▪ How well does M^{-1} precondition A_h ?

Unit cube, Poisson equation, 64 subdomains, DBC on one side, edge-based coarse space, solver tolerance 10^{-6} , sparse direct subdomain solvers

	Aspect ratio 1		Aspect ratio 10		Aspect ratio 100	
p	iters	cond#	iters	cond#	iters	cond#
2	7	1.4	31	28	72	2.7e3
3	10	1.8	39	32	164	3.1e3
4	10	2.3	42	36	245	3.3e3
5	12	2.8	43	38	318	3.5e3
6	14	3.3	45	41	387	3.6e3

- Meshes with high aspect ratio elements require special attention
 - Adaptive coarse spaces
 - some form of semi-coarsening

Numerical Results

- Putting it all together:

$$\kappa(M^{-1}A_p) \leq \kappa(M^{-1}A_h)\kappa(A_h^{-1}A_p)$$

p	sparse direct		AMG*	
	iterations	condition #	iterations	condition #
2	15	5.4	15	5.4
3	15	5.1	16	5.2
4	17	6.9	19	7.2
5	19	9.9	23	10.3
6	22	13.5	26	13.5

Poisson equation

Notes: Same problem but now BDDC preconditioner for higher-order problem (sparse direct or AMG subdomain & coarse solvers)

*Algebraic multigrid (AMG) V-cycle, single pre/post smoothing step (symmetric Gauss-Seidel), max coarse size 10, semi-coarsening based on geometry

Numerical Results

- Putting it all together:

$$\kappa(M^{-1}A_p) \leq \kappa(M^{-1}A_h)\kappa(A_h^{-1}A_p)$$

	LU		BDDC/direct		BDDC/AMG	
p	iters	cond#	iters	cond#	iters	cond#
2	25	9.2	25	9.2	26	9.2
3	26	10.0	28	10.3	28	10.0
4	29	13.2	31	13.2	31	13.0
5	32	16.1	35	16.5	36	17.0
6	35	20.7	41	19.9	40	20.6

elasticity
problem

Notes: 4x4x3 mesh, one face fixed, edge & vertex BDDC coarse space, PETSc BDDC implementation, native PETSc AMG preconditioner for subdomain solvers

Takeaway: AMG subdomain solvers can be quite effective

Numerical Results

- **Memory Requirements for element “factorizations”:**
 - **Operator complexity for sparse solver** = $\frac{nnz(K_h) + nnz(L_h)}{nnz(K_h)}$
 - **Operator complexity for AMG** = $\sum_{i=1}^{nlevels} nnz(K_{hi}) / nnz(K_{h1})$

Operator complexities for “equivalent” assembly of linear elements fixed on one side

	Poisson equation		Elasticity	
p	Pardiso	AMG	Pardiso	AMG
2	2.3	1.01	2.4	1.02
4	4.0	1.04	4.0	1.15
6	5.0	1.03	5.0	1.14
8	6.8	1.03	6.6	1.10
10	8.6	1.04	8.4	1.16

Takeaway: AMG is much more attractive from memory standpoint as a subdomain solver.

Numerical Results

- **Solution Times:**

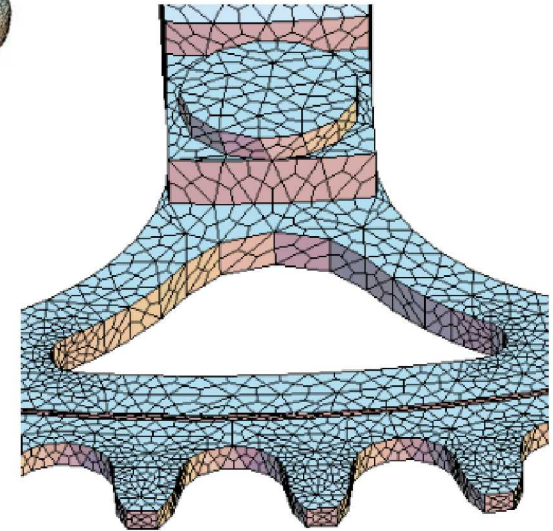
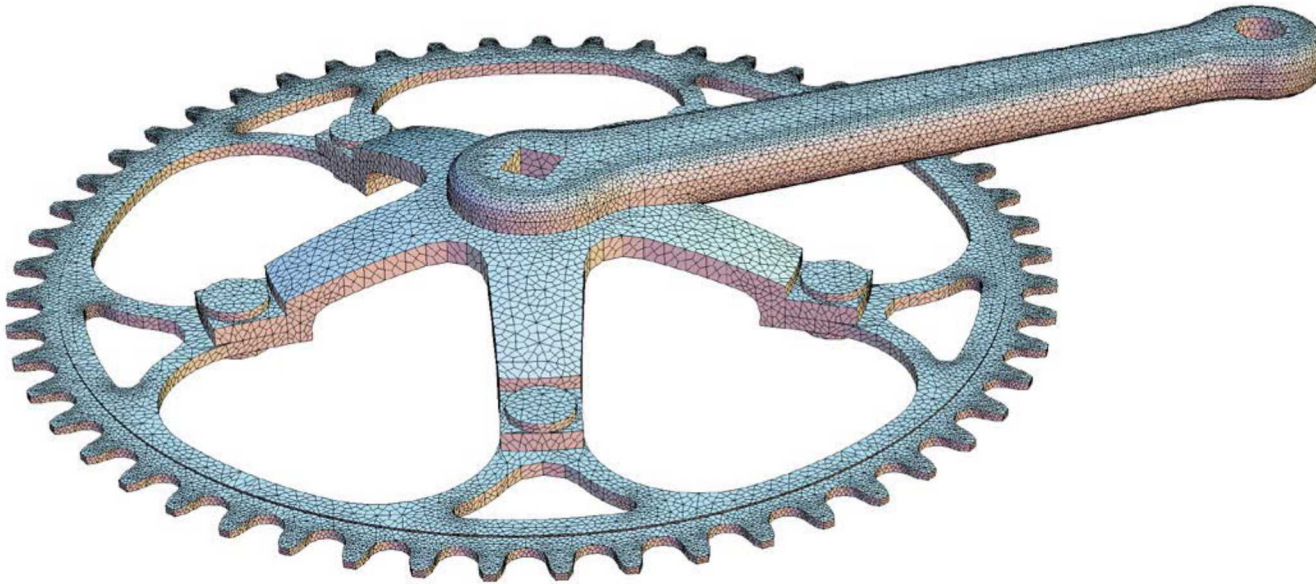
Speedups for AMG versus Intel/Pardiso sparse direct subdomain solvers

p	Poisson equation		Elasticity	
	initialization	solve	initialization	solve
2	1.5	1.7	0.89	0.68
4	1.3	0.71	0.52	0.50
6	1.2	0.64	0.51	0.58
8	1.4	0.65	0.64	0.68

Notes: Unit cube, Poisson equation, 64 subdomains, DBC on one side, solver tolerance 10^{-6} , edge-based BDDC coarse space, sparse direct (Pardiso) or AMG subdomain solvers, Gauss-Legendre quadrature for elasticity, best times from three runs.

Takeaway: AMG not always faster as a subdomain solver for range of p considered

Unstructured Meshes



			Edge/Corner coarse space		Adaptive coarse space	
p	subs	ndof	iters	ncdof	iters	ncdof
1	32	712,623	26	699	26	1021
2	256	5,067,309	72	8706	52	9564
3	864	16,545,081	112	31,191	60	39,354
4	2048	38,626,995	223	68,133	70	114,886

elasticity
problem

■ Some Challenges:

- Required use of more expensive “deluxe scaling” to get reasonable number of iterations using standard coarse spaces
- Even with deluxe scaling, fast growth of condition numbers was observed as polynomial degree increased
- Adaptive coarse spaces can be used to reduce iterations and condition numbers (e.g. 70 versus 223 iterations for $p = 4$ elasticity model with $\approx 39\text{M}$ unknowns)

■ Some Questions:

- Is poor approximation of K_p by K_h a problem?
 - Iterations/cond# were just 26/12.9 for $p = 1$ elasticity model
- Is it better to iteratively “solve” local subdomain problems using matrix-free higher-order operator?
- How to improve convergence rate of AMG subdomain solvers?
- Precondition tet element model rather than hex “equivalent”?

Recap & Future Work

- **BDDC viable option for preconditioning matrix-free higher-order methods**
 - Challenging computations isolated to individual elements or subdomains
 - Problem looks like standard one after first coarsening
 - Inexact subdomain solvers (e.g. AMG) save memory
- **Possible Future Work:**
 - Better understand challenges for unstructured meshes
 - Explore options to directly use matrix-free higher-order operator when solving local subdomain problems
 - Explore options to deal with high aspect ratio elements
 - Explore BDDC preconditioners for $H(\text{div})/H(\text{curl})$ problems

References

- Beirao da Veiga, Cho, D., Pavarino, L.F., and Scacchi, S., “BDDC preconditioners for isogeometric analysis”, *Math. Models Methods Appl. Sci.*, Vol. 23, pp. 1099-1142, 2013.
- Beirao da Veiga, L., Pavarino, L.F., Scacchi, S., Widlund, O.B., and Zampini, S., “Isogeometric BDDC Preconditioners with Deluxe Scaling”, *SIAM J. Sci. Comput.*, Vol 36, pp. A1118-A1139, 2014.
- Bertoluzza, S., Pennacchio, M., and Prada, D., “BDDC and FETI-DP for the virtual element method”, *Calcolo*, Vol. 54, pp. 1565-1593, 2017.
- Dohrmann, C.R., “An approximate BDDC preconditioner”, *Numer. Linear Algebra Appl.*, Vol. 14, pp 149-168, 2007.
- Klawonn, A., Pavarino, L.F., and Rheinbach, O., “Spectral element FETI-DP and BDDC preconditioners with multi-element subdomains”, *Comput. Methods Appl. Mech. Engrg.*, Vol. 198, pp. 511-523, 2008.
- Orszag, S.A., “Spectral methods for problems in complex geometries”, *J. Comput. Phys.*, Vol. 37, pp. 70-92, 1980.
- Pavarino, L.F., “BDDC and FETI-DP preconditioners for spectral element discretization”, *Comput. Methods Appl. Mech. Engrg.*, Vol. 196, pp. 1380-1388, 2007.

References

- Pavarino, L.F., Widlund, O.B., and Zampini, S., “BDDC preconditioners for spectral element discretizations of almost incompressible elasticity in three dimensions”, *SIAM J. Sci. Comput.*, Vol. 32, pp. 3604-3626, 2010.
- Toselli, A. and Widlund, O., Domain Decomposition Methods – Algorithms and Theory, *Springer Series in Computational Mathematics*, Vol. 34, 2005.

- Optimized approaches for element matrices and matrix-vector products:
 - Melenk, J.M., Gerdes, K., and Schwab, C., “Fully discrete hp-finite elements: fast quadrature”, *Comput. Methods Appl. Mech. Engrg.*, Vol. 190, pp. 4339-4364, 2001.
 - Ainsworth, M., Andriamaro, G., and Davydov, O., “Bernstein-Bezier finite elements of arbitrary order and optimal assembly procedures”, *SIAM J. Sci. Comput.*, Vol. 33, pp. 3087-3109, 2011.
 - Ainsworth, M., Davydov, O., and Schumaker, L.L., “Bernstein-Bezier finite elements on tetrahedral-hexahedral-pyramidal partitions”, *Comput. Methods Appl. Mech. Engrg.*, Vol. 304, pp. 140-170, 2016.