PCBDDC : dual-primal preconditioners in PETSc

Stefano Zampini

KAUST
Computer, Electrical and Mathematical Sciences & Engineering,
Extreme Computing Research Center
Saudi Arabia

Celebrating 20 years of PETSc, Argonne National Lab, June 16th, 2015

Outline

- Introduction
- Non-overlapping Domain Decomposition
- Balancing Domain Decomposition by Constraints and its current implementation (BDDC, PCBDDC) in 3.6
- Experimental interface to Finite Element Tearing and Interconnecting Dual Primal (FETI-DP)
- Numerical results with non-standard PDEs with high contrast coefficients.
- Future extensions.

More details in [S. Z. PCBDDC : a class of robust dual-primal methods in PETSc, submitted].

1. $\mathcal{L}u = f$ on Ω.
2. Find $u \in V$ s.t. $\mathbf{a}(u, v) =< f, v > \forall v \in V$.
3. Find $\mathbf{u_h} \in \widehat{\mathbf{W}}$ s.t. $\mathbf{a}(\mathbf{u_h}, \mathbf{v_h}) =< f, \mathbf{v_h} > \forall \mathbf{v_h} \in \widehat{\mathbf{W}}$.
4. $\{\widehat{A}\}_{ij} = \mathbf{a}(\phi_i, \phi_j), f_i =< f, \phi_i >, \widehat{\mathbf{W}} = \text{span}\{\phi_i\}$

- Mainly interested in very large (and sparse) linear systems
- Preconditioned Krylov solvers for $\widehat{A}\mathbf{u} = \mathbf{f}$
- Domain Decomposition approach with very many subdomains
- Goals:
  - Obtain convergence rates which are independent of the number of subdomains, and slowly deteriorates with the size of the subdomain problems
  - Accommodate arbitrary distributions of the coefficients of the PDE

[A. Toselli and O. B. Widlund, Domain Decomposition Methods: algorithm and theory, 2005]

Introduction: BDDC and FETI-DP pros and cons

- Simple customization
- Complex geometries
- High order discretizations
- Robust with respect to the coefficients of the PDE
  - slowly varying
  - piecewise constant
  - heterogeneous (SPD)
- Black-box (SPD)
- Highly scalable (multilevel extension)

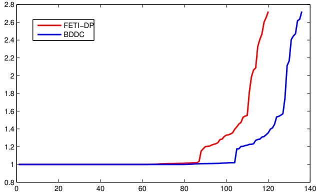
- Do not work with assembled matrices (still).
- Require factorizations and direct solves of subproblems (inexact solvers)
- Do not work in matrix-free contexts.
- High contrast problems for non-SPD systems

Introduction: BDDC and FETI-DP

The construction of both methods relies on the selection of *primal continuity constraints* and on the choice of an averaging procedure. For the same choice of the primal constraints and the averaging procedure, preconditioned spectra are the same. Tipically

$$\kappa_2 \sim (1 + \log(H/h))^2$$

with H maximum diameter of sudomains and h mesh size [Mandel, Dohrmann, Tezaur, Appl. Numer. Math. 54, 2005], [Li and Widlund, IJNME, 2008].



Eigenvalues comparison.

Introduction: some references (problems, incomplete)

- Contact problems [P. Avery, G. Rebel, M. Lesoinne, C. Farhat. CMAME 93, 2004.]
- Indefinite problems [J. Li, X. Tu. NLAA 16, 2009. C. Farhat, J. Li ANM 54, 2005.]
- Indefinite complex problems [C. Farhat, J. Li, P. Avery IJNME 63, 2005.]
- Electromagnetic problems [Y. J. Li, J. M. Jin IEEE Trans. Antennas Propag. 54, 2006.]
- Incompressible Stokes [J. Li, O. B. Widlund. SISC 44, 2006.]
- Linear elasticity [A. Klawonn, O. B. Widlund CPAM 59, 2006.]
- Stokes problem [H. H. Kim, C. O. Lee, E. H. Park SISC 47, 2010.]
- Stokes–Darcy coupling [J. Galvis, M. Sarkis. CAMCS 5, 2010.]
- Almost Incompressible Elasticity [L. Pavarino, O. B. Widlund, S. Z. SISC 32, 2010.]
- Nonlinear preconditioners [Klawonn, M. Lanser and O. Rheinbach SISC, 36, 2014]

Introduction: some references (discretizations, incomplete)

- Spectral Elements [L. F. Pavarino, CMAME 196, 2007.]
- Lowest order Nédélec elements [A. Toselli, IMAJNA 26, 2006], [C. Dohrmann , O. B.Widlund, CPAM 2015]
- Discontinuous Galerkin [M. Dryja, J. Galvis, M. Sarkis. J. Complexity 23, 2007]
- Mortar discretizations [H. H. Kim. SINUM 46, 2008, H. H. Kim, M. Dryja, O. B. Widlund. SINUM 47, 2009.]
- Reissner-Mindlin plates and Tu-Falk elements [J. H. Lee, SINUM, 2015.]
- Naghdi shells and MITC elements [L. Beirao da Veiga, C. Chinosi, C. Lovadina, L. F. Pavarino. Comp. Struct. 102, 2012.]
- IsoGeometric Analysis [L. Beirao da Veiga, L. Pavarino, S. Scacchi, O. B. Widlund, S.Z. SISC, 36, 2014.]
- Lowest order Raviart-Thomas elements [D.-S. Oh , C. Dohrmann , O. B. Widlund, TR, 2014.]

Non-overlapping DD: matrix subassembling

- Non-overlapping Domain Decomposition

$$\overline{\Omega} = \bigcup_{i=1}^{N} \overline{\Omega}_i, \quad \Omega_i \cap \Omega_j = \emptyset.$$

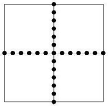
- $\widehat{A}$ never assembled explicitly; instead

$$\widehat{A} = R^T A R, \quad A = \text{diag}(A^{(i)})$$

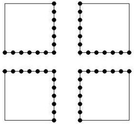
- Discrete analog of

$$\mathbf{a}(\cdot,\cdot) = \int_{\Omega} \cdot = \int_{\cup_{i=1}^{N}\Omega_i} \cdot = \sum_{i=1}^{N} \int_{\Omega_i} \cdot = \sum_{i=1}^{N} \mathbf{a}^{(i)}(\cdot,\cdot)$$

- Restriction operator $R : \widehat{\mathbf{W}} \rightarrow \mathbf{W}$



$\widehat{\mathbf{W}}$



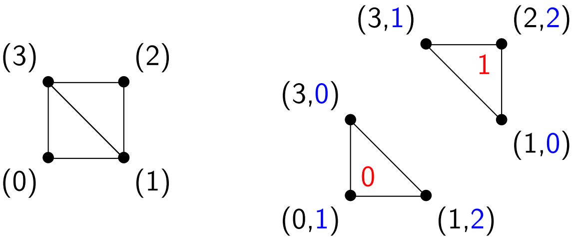
$\mathbf{W}$

Non-overlapping DD: subassembled matrices in PETSc

- Matrix type in PETSc is `MATIS`: most of the matrix operations are provided.
- `MatCreateIS(MPI_Comm comm, PetscInt bs, PetscInt m,PetscInt n, PetscInt M, PetscInt N, ISLocalToGlobalMapping map, Mat* A)`.
- One-to-one mapping between MPI processes and subdomains.
- Current implementation limited to square matrices (easily extensible).
- Handling local problems: `MatISSetLocalMat`, `MatISGetLocalMat`.
- Values can be inserted either with global or local numbering.
- Preallocation can be done via `MatISSetPreallocation`.
- Assembly can be performed by using `MatISGetMPIXAIJ`.

Non-overlapping DD: subassembled matrices in PETSc

How to construct a `MATIS` object (local numbering in blue)



- subdomain 0: map {3,0,1}.
- subdomain 1: map {1,3,2}.

Non-overlapping DD: Block factorizations

- Interface Γ among non-overlapping subdomains

$$\Gamma = \bigcup_{i \neq j} \partial\Omega_j \cap \partial\Omega_i$$

- Block factorization for $\widehat{A}$ based on the split $\widehat{\mathbf{W}} = \widehat{\mathbf{W}}_\Gamma \oplus \mathbf{W}_I$

$$\widehat{A}^{-1} = \begin{bmatrix} I_{II} & -A_{II}^{-1}\widehat{A}_{I\Gamma} \\ & I_{\Gamma\Gamma} \end{bmatrix} \begin{bmatrix} A_{II}^{-1} & \\ & S_\Gamma^{-1} \end{bmatrix} \begin{bmatrix} I_{II} & \\ -\widehat{A}_{\Gamma I}A_{II}^{-1} & I_{\Gamma\Gamma} \end{bmatrix}$$

with $S_\Gamma = \widehat{A}_{\Gamma\Gamma} - \widehat{A}_{\Gamma I}A_{II}^{-1}\widehat{A}_{I\Gamma}$.

- Block preconditioner

$$B^{-1} = \begin{bmatrix} I_{II} & -B_{II}^{-1}\widehat{A}_{I\Gamma} \\ & I_{\Gamma\Gamma} \end{bmatrix} \begin{bmatrix} B_{II}^{-1} & \\ & B_{\Gamma\Gamma}^{-1} \end{bmatrix} \begin{bmatrix} I_{II} & \\ -\widehat{A}_{\Gamma I}B_{II}^{-1} & I_{\Gamma\Gamma} \end{bmatrix}$$

- B_{II}^{-1} uncoupled Dirichlet solvers (static condensation), $B_{\Gamma\Gamma}^{-1}$ Schur complement preconditioner (BDDC).

Non-overlapping DD: interface classes

Degrees of freedom on Γ partitioned into equivalence classes that play a central role in the design, analysis and programming of the methods; interface analysis done by using

- The set of sharing subdomains $\mathcal{N}_x$ (deduced from l2g map)
- An additional equivalence relation $\sim$ for dofs connectivity.

$$
2D \begin{cases} x, y \in E_k \iff |\mathcal{N}_x| = 2,\ \mathcal{N}_x \cong \mathcal{N}_y,\ x \sim y,\ x \notin \partial\Omega_N \\ x \in V_k \iff |\mathcal{N}_x| > 2 \text{ or } x \in \partial\Omega_N \end{cases}
$$

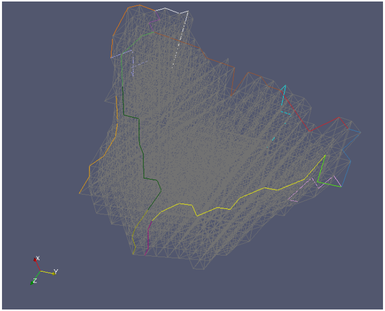
$$
3D \begin{cases} x, y \in F_k \iff |\mathcal{N}_x| = 2,\ \mathcal{N}_x \cong \mathcal{N}_y,\ x \sim y,\ x \notin \partial\Omega_N \\ x, y \in E_k \iff |\mathcal{N}_x| > 2,\ \mathcal{N}_x \cong \mathcal{N}_y,\ x \sim y \\ \qquad\qquad\qquad \text{or} \\ \qquad\qquad\qquad |\mathcal{N}_x| = 2,\ \mathcal{N}_x \cong \mathcal{N}_y,\ x \sim y,\ x \in \partial\Omega_N \\ x \in V_k \iff \nexists y \neq x \text{ s.t. } \mathcal{N}_x \cong \mathcal{N}_y,\ x \sim y \end{cases},
$$

[Klawonn and Rheinbach, CMAME 196, 2007]

Non-overlapping DD: interface classes

Available customizations (some of):

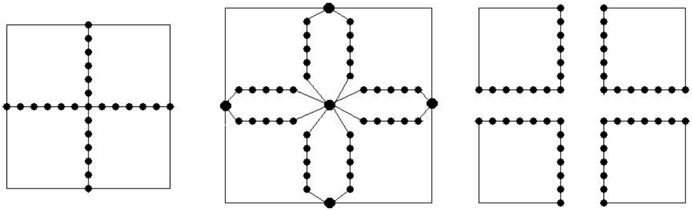
- `PCBDDCSetDofsSplitting`
- `PCBDDCSetDirichlet/NeumannBoundaries`
- `PCBDDCSetLocalAdjacencyGraph`



BDDC method: primal and dual spaces

- Idea: instead of $\widehat{S}_\Gamma$, invert $\widetilde{S}_\Gamma$, defined on partially assembled space

$$\widehat{\mathbf{W}} \subset \widetilde{\mathbf{W}} = \mathbf{W}_I \oplus \widetilde{\mathbf{W}}_\Gamma \subset \mathbf{W}, \quad \widetilde{\mathbf{W}}_\Gamma = \widehat{\mathbf{W}}_\Pi \oplus \mathbf{W}_\Delta$$



$\widehat{\mathbf{W}}$ $\widetilde{\mathbf{W}}$ $\mathbf{W}$

- Discontinuous functions on Γ except at primal dofs (Π).
- Primal vertices to prevent subdomains from floating.
- Additional primal dofs for edges and faces are usually needed to obtain quasi-optimal bounds and robustness with respect to rough coefficients distributions.

BDDC method: primal and dual spaces

- Edge/face primal dofs as constraints (e.g. averages)
- Implicit dofs using a constraint matrix C [C. Dohrmann SISC 25, 2003],

  c_{ik} = quadrature weight for i-th constraint and k-th dofs

- Explicit dofs with a change of basis T; the new Iteration matrix is obtained by projection [A. Klawonn, O. B. Widlund. CPAM 59, 2006]

$$T^T \widehat{A} T, \quad T\mathbf{u}_{\text{new}} = \mathbf{u}_{\text{old}}.$$

Primal space customizable using (many other command line opts)

- `MatSetNearNullSpace`; quadrature weights.
- `PCBDDCSetPrimalVerticesLocalIS`; user-defined additional primal vertices.
- `PCBDDCSetChangeOfBasisMat`; user-defined change of basis.

BDDC method: preconditioner application

BDDC preconditioner [C. Dohrmann, SISC, 2003], [C. Dohrmann, NLAA 2007], [J. Li and O. B. Widlund 2008]

$$M_\Gamma^{-1} = R_{D,\Gamma}^T \widetilde{S}_\Gamma^{-1} R_{D,\Gamma}, \quad \widetilde{S}_\Gamma^{-1} = P_C + P_L$$

- Local corrections (uncoupled Neumann problems)

$$P_L \, \mathbf{g}_\Gamma = \underset{C\mathbf{w}=0, \mathbf{w}\in\mathbf{W}}{\operatorname{argmin}} \; \mathbf{w}^T (A\mathbf{w} - \mathbf{g}).$$

  with $C = \operatorname{diag}(C^{(i)})$ defined on the unassembled space and $\mathbf{g}$ the extension by zero of $\mathbf{g}_\Gamma$

- Coarse correction (parallel sparse problem, subassembled)

$$P_C = \Psi A_c^{-1} \Phi^T, \quad A_c = \sum_{i=1}^{N} R_\Pi^{(i)T} \Phi^{(i)T} A^{(i)} \Psi^{(i)} R_\Pi^{(i)}$$

$$\Psi = \underset{C\mathbf{w}=I, \mathbf{w}\in\mathbf{W}}{\operatorname{argmin}} \; \mathbf{w}^T A \mathbf{w}, \quad \Phi = \underset{C\mathbf{w}=I, \mathbf{w}\in\mathbf{W}}{\operatorname{argmin}} \; \mathbf{w}^T A^T \mathbf{w}.$$

BDDC preconditioner: implementation details

Computation of subdomain corrections and primal basis Ψ (Φ similar)

$$\begin{pmatrix} A & C^T \\ C & 0 \end{pmatrix} \begin{pmatrix} \mathbf{w} \\ \boldsymbol{\mu} \end{pmatrix} = \begin{pmatrix} \mathbf{g} \\ \mathbf{h} \end{pmatrix}.$$

Split each block matrix in vertex (v) and remaining (r) nodes

$$A = \begin{pmatrix} A_{rr} & A_{rv} \\ A_{rv} & A_{vv} \end{pmatrix}, \quad C = \begin{pmatrix} C_r & 0 \\ 0 & I_v \end{pmatrix}, \quad A_{ab} = \text{diag}(A_{ab}^{(j)}).$$

Solution is [C. Dohrmann, NLAA 2007]

$$\boldsymbol{\mu}_c = \left(C_r A_{rr} C_r^T\right)^{-1} \left[C_r A_{rr}^{-1}(\mathbf{g}_r - A_{rv}\mathbf{h}_v) - \mathbf{h}_c\right]$$

$$\mathbf{w}_r = A_{rr}^{-1}\left(\mathbf{g}_r - A_{rv}\mathbf{h}_v - C_r^T \boldsymbol{\mu}_c\right)$$

$$\mathbf{w}_v = \mathbf{h}_v$$

$$\boldsymbol{\mu}_v = \mathbf{g}_v - A_{vr}\mathbf{w}_r - A_{vv}\mathbf{h}_v.$$

Computation of subdomain corrections ($\mathbf{h} = 0$, $\boldsymbol{\mu}_v$ not needed)

$$\boldsymbol{\mu}_c = (C_r A_{rr} C_r^T)^{-1} C_r A_{rr}^{-1} \mathbf{g}_r$$
$$\mathbf{w}_r = A_{rr}^{-1} \mathbf{g}_r + A_{rr}^{-1} C_r^T \boldsymbol{\mu}_c$$

i.e. subdomain solve + rank-n update.

Computation of primal basis ($\mathbf{g} = 0$, $\mathbf{h} = I$)

$$\widetilde{\Phi} = \begin{pmatrix} (A_{rr}^{-1} C_r^T (C_r A_{rr} C_r^T)^{-1} C_r \; - I) A_{rr}^{-1} A_{rv} & -(C_r A_{rr} C_r^T)^{-1} C_r \\ I_v & 0 \end{pmatrix}.$$

and coarse subdomain matrix as a by-product

$$\Phi^{(i)T} A^{(i)} \Psi^{(i)} = -\Phi^{(i)T} C^T \Lambda^{(i)} = -\Lambda^{(i)}$$

BDDC method: scaling operator

- $R_{D,\Gamma} = D\widetilde{R}_\Gamma$, $D = \text{diag}(D^{(j)})$, $\widetilde{R}_\Gamma : \widehat{\mathbf{W}}_\Gamma \to \widetilde{\mathbf{W}}_\Gamma$
- Restores continuity during Krylov iterations
- Accommodates jumps in the coefficients aligned with Γ
- Standard scaling: $D^{(j)}$ diagonal

$$\delta_j^\dagger(x) = \frac{\delta_j(x)}{\sum_{k\in\mathcal{N}_x} \delta_k(x)}, \quad x \in \mathbf{W}_\Delta$$

- Robust for scalar PDEs with some configurations of jumps across Γ [C. Pechstein, R. Scheicl, Numer. Math 111, 2008].
- Deluxe scaling: $D^{(j)}$ block diagonal with blocks

$$\Big(\sum_{k\in\mathcal{N}_F} S_F^{(k)}\Big)^{-1} S_F^{(j)}, \quad S_F^{(j)} = A_{FF}^{(j)} - A_{FI}^{(j)} A_{II}^{(j)-1} A_{IF}^{(j)}$$

with F an edge or a face of Γ, [C. Dohrmann, O. B. Widlund, DD21, 2011].

Adaptive selection of constraints

- Convergence properties of DD algorithms usually deteriorates when jumps in the coefficients of the PDE are not aligned with Γ
- Adaptive selection of constraints in dual-primal methods is a very active topic of research, different techniques (each based on some generalized eigenvalue problem, GEP) have been recently proposed for elliptic PDEs: [J. Mandel and B. Sousedík, DD XVI, 2007], [B. Sousedík, J. Šístek, and J. Mandel, Computing, 95, 2013], [A. Klawonn, P. Radtke and O. Rheinbach, SINUM 53, 2015, and PAMM, 2015], [H.H. Kim and E.T. Chung, SIAM J. Multiscale Model. Simul., 13, 2015], [C. Pechstein, C. R. Dohrmann, Seminar talk 2013]
- Similar techniques using GEP have been also studied for enriching the coarse space of Schwarz algorithms and FETI [J. Galvis and Y. Efendiev, Multiscale Models Simul. 8, 2010], [N. Spillane and D. J. Rixen, IJNME, 2013], [N. Spillane, V. Dolean, P. Hauret, F. Nataf, and D. J. Rixen, C.R. Math. Acad. Sci. Paris 2013], [N. Spillane, V. Dolean, P. Hauret, F. Nataf, C. Pechstein and R. Scheichl, Num. Math, 126, 2014]

Adaptive selection of constraints

PCBDDC implements the technique proposed by Pechstein and Dohrmann, by combining deluxe scaling with the following GEP on each subdomain face F

$$(S_F^{(i)} : S_F^{(j)})\phi = \lambda(\widetilde{S}_F^{(i)} : \widetilde{S}_F^{(j)})\phi$$

$$S_F^{(i)} : S_F^{(j)} = (S_F^{(i)-1} + S_F^{(j)-1})^{-1}$$
$$\widetilde{S}_F^{(k)} = S_{FF}^{(k)} - S_{FF'}^{(k)} S_{F'F'}^{(k)-1} S_{F'F}^{(k)}$$

If we include in the primal space all the vectors of the form $(S_F^{(i)} : S_F^{(j)})\phi_k$, with $\lambda_k > \lambda_m$, then the contribution of F to the maximum eigenvalue of the preconditioned operator will be less than λ_m times a constant.

Still an active topic of research for edge classes in 3D.

PCBDDC consider

$$(S_F^{(i)} : S_F^{(j)} : S_F^{(k)})\phi = \lambda(\widetilde{S}_F^{(i)} : \widetilde{S}_F^{(j)} : \widetilde{S}_F^{(k)})\phi$$

Adaptive selection of constraints: implementation details

- MUMPS is used to compute $S^{(i)}$; factorization of $A_{II}^{(i)}$ could be reused.
- $S_F^{(i)}$ explicitly inverted.
- Computation of $\widetilde{S}_F^{(i)-1}$: explicit inversion of $S^{(i)}$.
- Nearest neighbor communications to assemble each sum of Schur complements.
- LAPACK used to solve the GEPs

$$(\widetilde{S}_F^{(i)-1} + \widetilde{S}_F^{(j)-1})\phi = \lambda(S_F^{(i)-1} + S_F^{(j)-1})\phi$$
$$(\widetilde{S}_F^{(i)-1} + \widetilde{S}_F^{(j)-1} + \widetilde{S}_F^{(k)-1})\phi = \lambda(S_F^{(i)-1} + S_F^{(j)-1} + S_F^{(k)-1})\phi$$

- Selected by command line `-pc_bddc_use_deluxe_scaling`, `-pc_bddc_adaptive_threshold x`

Multilevel extension

Direct solution of coarse problem could become a bottleneck with many subdomains and/or constraints. A possible remedy consists in using Multilevel BDDC (or AMG if suitable, e.g. for elasticity).

- Coarse subdomain matrices as elements of a coarser discretization.
- Subdomains (coarse elements) can be aggregated into coarse subdomains.
- Exact solution of coarse problem replaced by the application of a coarse BDDC preconditioner.
- Local corrections and coarse problem at the coarser level can be overlapped.

[X. Tu, SISC 29, 2007], [J. Mandel, B. Sousedík and C. R. Dohrmann, Computing, 2008].

Approximate coarse solvers could degrade the convergence rates if not spectral equivalent.

FETI-DP method: constrained minimization

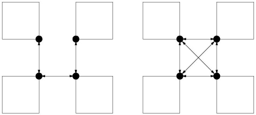
- Constrained minimization problem [C. Farhat *et. al* IJNME 50, 2001]

$$\underset{\mathbf{w}\in\widetilde{\mathbf{W}}}{\operatorname{argmin}} \left[\frac{1}{2}\mathbf{w}^T\widetilde{A}\mathbf{w} - \mathbf{w}^T\widetilde{\mathbf{f}}\right], \text{ s.t. } B\mathbf{w} = 0,$$

- Jump operator (sparse matrix with entries ± 1)

$$B = \left[B^{(1)}|\dots|B^{(N)}\right], \quad B\mathbf{w} = 0 \iff \mathbf{w} \in \widehat{\mathbf{W}}.$$

- row of B: continuity of dual dofs between two subdomains.
- n_x rows for each dof x
  - $n_x = |\mathcal{N}_x| - 1 \rightarrow$ non-redundant multipliers,
  - $n_x = |\mathcal{N}_x|(|\mathcal{N}_x| - 1)/2 \rightarrow$ fully-redundant multipliers.



FETI-DP method: linear system

- Associated saddle point formulation

$$\begin{bmatrix} \widetilde{A} & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \widetilde{\mathbf{f}} \\ \mathbf{0} \end{bmatrix}, \quad \boldsymbol{\lambda} \in \Lambda = \text{Range}(B).$$

- FETI-DP linear system for multipliers

$$F\boldsymbol{\lambda} = \mathbf{d}, \quad F = B\widetilde{A}^{-1}B^T, \quad \mathbf{d} = B\widetilde{A}^{-1}\widetilde{\mathbf{f}}.$$

- Dirichlet preconditioner

$$M^{-1} = B_D \overline{R}_\Delta^T (A_{\Delta\Delta} - A_{I\Delta}^T A_{II}^{-1} A_{I\Delta}) \overline{R}_\Delta B_D^T.$$

- Standard solution can be recovered as

$$\mathbf{w} = \widetilde{A}^{-1}\left(\widetilde{\mathbf{f}} - B^T\boldsymbol{\lambda}\right), \quad \mathbf{w} \in \widehat{\mathbf{W}}.$$

FETI-DP method: cont'd

- Non-redundant case [A.. Klawonn, O. B. Widlund, CPAME 54, 1999]

$$B_D = D^{-1}B^T(BD^{-1}B^T)^{-1}B.$$

- Fully-redundant case [J. Mandel, C. Dohrmann, R. Tezaur, Appl. Numer. Math. 54, 2005]

$$B_D = \left[D_\Lambda^{(1)}B^{(1)}|\dots|D_\Lambda^{(N)}B^{(N)}\right].$$

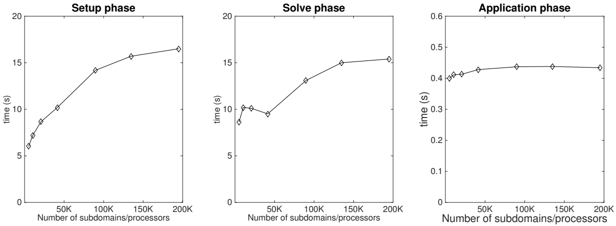
with $D_\Lambda^{(j)}$ diagonal matrix with entries $d_{ii}^{(j)} = \delta_k^\dagger(x)$, where i is the local index of the multiplier imposing continuity between jth and kth subdomain for dof x.

- `PCBDDCCreateFETIDPOperators(pcbddc,&F,&M)`
- `PCBDDCMatFETIDPGetRhs(F,dofs_rhs,fetidp_rhs)`
- `PCBDDCMatFETIDPGetSolution(F,fetidp_sol,dofs_sol)`

Numerical results: experimental setting

- Running on Cray XC40 Shaheen at KAUST: 6192 dual 16-core Haswell processors clocked at 2.3 Ghz and equipped with 128GB of DRAM per node, for a total of 198,144 cores (rank in Top500 will be announced on July 23).
- PETSc 3.6 compiled with GNU 4.9.2, using `-O3` and with support for AVX instructions.
- Intel MKL version 11.2.2 for linear algebra kernels.
- $\Omega = [0, 1]^3$.
- Random right-hand sides, null initial guesses, PCG with rtol 1.e-8

Numerical results: inexact multilevel BDDC at extreme scale



Weak scaling test for the inexact multilevel BDDC applied to the Poisson problem with constant coefficients (hexahedra). AMG based local solvers and multilevel BDDC (coarsening ratio 768, no adaptivity at all). Computational times in seconds for the setup of the preconditioner (left), the PCG (central) and the application of the preconditioner (right) are plotted as a function of the number of processors. 70K dofs/subdomain, 12.5B dofs with 195K cores. Efficiency at 195K cores: 92%.

Numerical results: H(div)

- Bilinear form

$$\mathbf{a}(\mathbf{u},\mathbf{v}) = \int_\Omega \alpha \ \mathrm{div}\ \mathbf{u}\ \mathrm{div}\ \mathbf{v} + \beta\ \mathbf{u}\cdot\mathbf{v}\ dx, \quad \alpha \geq 0, \beta > 0.$$

- $\widehat{\mathbf{W}}$ = lowest order Raviart-Thomas elements on tetrahedra.
- dofs: normal component of $\mathbf{u}$ on elements' faces.
- Dirichlet boundary conditions on $\partial\Omega$.
- Implemented using FENICS: http://fenicsproject.org/
- Joint work with C. Dohrmann, D.-S. Oh, O. B. Widlund

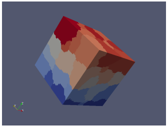
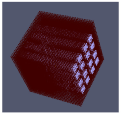
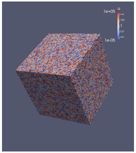
Numerical results: H(curl)

- Bilinear form (Eddy formulation of Maxwell's equations)

$$\mathbf{a}(\mathbf{u},\mathbf{v}) = \int_\Omega \alpha \, \nabla \times \mathbf{u} \cdot \nabla \times \mathbf{v} + \beta \, \mathbf{u} \cdot \mathbf{v} \, dx, \quad \alpha \geq 0, \beta > 0.$$

- $\widehat{\mathbf{W}}$ lowest order Nédélec elements on tetrahedra.
- dofs: tangential component of $\mathbf{u}$ on elements' edges.
- Dirichlet boundary conditions on $\partial\Omega$.
- Special change of basis as in [A. Toselli, IMAJNA 26, 2006], [C. R. Dohrmann, O. B. Widllund, CAPM, 2015].
- Implemented using FENICS: http://fenicsproject.org/
- Joint work with O. B. Widlund

Numerical results: experimental setting

- H(div) tests: 670K tetrahedra.
- H(curl) tests: 200K tetrahedra.
- 40 irregular subdomains (left).
- Channel test A: $\begin{cases} \alpha = \beta = 1.e4, & \text{channels} \\ \alpha = \beta = 1, & \text{elsewhere} \end{cases}$
- Channel test B:

$$\text{odd} \begin{cases} \alpha = 1.e4, \ \beta = 1, & \text{c} \\ \alpha = 1, \ \beta = 1.e4, & \text{e} \end{cases}, \quad \text{even} \begin{cases} \alpha = 1, \ \beta = 1.e4, & \text{c} \\ \alpha = 1.e4, \ \beta = 1, & \text{e} \end{cases}$$

- Random test: α and β element-wise randomly chosen.
- Eigenvalue threshold: 10.

H(div), channel test A

β \ α	0	1	4	9	16
0	6.05/17 (174)	6.05/12 (174)	6.05/12 (174)	6.05/12 (174)	6.06/12 (174)
1	9.55/21 (175)	9.67/18 (175)	9.02/17 (176)	8.08/16 (177)	9.79/17 (176)
4	9.40/23 (178)	9.41/19 (178)	9.47/20 (178)	9.56/18 (179)	9.43/18 (179)
9	9.98/24 (177)	10.03/20 (178)	10.14/19 (178)	10.01/20 (178)	9.86/19 (179)
16	10.41/24 (178)	10.48/20 (178)	10.52/19 (179)	10.60/21 (180)	10.47/20 (178)

Condition number, number of iterations, and total number of adaptive constraints computed (in parenthesis) are shown for different number of channels.

H(div), channel test B

β \ α	0	1	4	9	16
0	6.05/17 (174)	6.97/15 (176)	6.91/15 (176)	6.91/15 (176)	6.89/15 (176)
1	6.98/15 (219)	7.02/14 (220)	6.65/15 (220)	8.68/16 (218)	8.21/15 (219)
4	8.55/18 (276)	8.54/17 (277)	8.94/17 (280)	8.64/17 (277)	8.81/18 (276)
9	10.19/18 (319)	9.98/18 (319)	10.00/18 (318)	10.30/17 (323)	9.93/18 (318)
16	8.28/17 (373)	7.98/17 (375)	8.17/17 (376)	9.04/18 (373)	8.30/17 (376)

Condition number, number of iterations, and total number of adaptive constraints computed (in parenthesis) are shown for different number of channels.

H(div), random coefficients

β \ α	0	2	4	6	8
0	6.05/17 (174)	5.85/16 (174)	5.13/14 (174)	4.03/11 (174)	- - -
2	10.18/22 (213)	10.02/22 (212)	9.70/20 (199)	10.22/18 (178)	8.64/15 (174)
4	9.75/22 (1171)	9.73/22 (1165)	9.64/20 (1139)	11.45/19 (1060)	12.53/18 (902)
6	8.58/21 (1903)	8.55/21 (1900)	9.26/20 (1864)	9.45/19 (1819)	9.05/17 (1658)
8	9.53/22 (2291)	9.62/21 (2285)	9.90/20 (2267)	10.28/19 (2201)	9.09/17 (2067)

α and β element-wise randomly chosen in $[10^{-p}, 10^{p}]$. Condition number, number of iterations, and total number of adaptive constraints computed (in parenthesis) are shown for different values of the contrast (i.e. $2p$).

H(curl), channel test A

β \ α	0	1	4	9	16
0	3.26/14 (954)	3.83/15 (954)	3.49/15 (954)	4.63/16 (957)	4.29/16 (954)
1	4.40/17 (973)	3.86/16 (954)	3.67/16 (955)	6.92/19 (965)	8.00/19 (956)
4	4.48/17 (976)	5.54/20 (961)	3.78/15 (955)	11.2/23 (979)	8.67/20 (967)
9	7.36/22 (979)	6.81/21 (974)	5.50/20 (967)	7.62/20 (968)	8.33/22 (986)
16	6.25/19 (1024)	7.78/20 (992)	5.41/18 (987)	7.30/22 (1032)	8.17/20 (978)

Condition number, number of iterations, and total number of adaptive constraints computed (in parenthesis) are shown for different number of channels.

H(curl), channel test B

β \ α	0	1	4	9	16
0	3.26/14 (954)	26.8/33 (961)	27.5/40 (955)	25.5/33 (989)	19.6/29 (991)
1	3.74/15 (956)	26.1/33 (963)	27.6/41 (956)	25.4/34 (988)	19.6/30 (991)
4	3.59/15 (957)	24.7/33 (962)	27.4/40 (959)	26.1/32 (992)	19.7/29 (996)
9	4.75/17 (955)	33.1/35 (968)	28.4/42 (960)	26.6/33 (1003)	19.9/30 (1009)
16	5.22/17 (959)	25.7/32 (974)	26.2/41 (964)	26.0/33 (1014)	21.7/30 (1015)

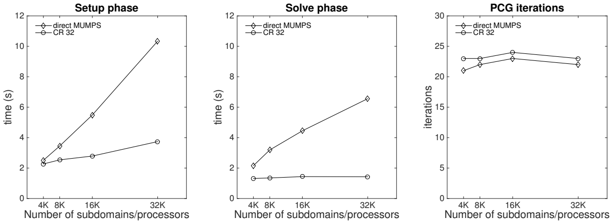
Condition number, number of iterations, and total number of adaptive constraints computed (in parenthesis) are shown for different number of channels.

H(curl), random test

β \ α	0	2	4	6	8
0	3.26/14 (954)	5.31/19 (977)	5.48/19 (1484)	10.14/20 (1867)	13.15/20 (2101)
2	3.25/14 (954)	5.33/19 (978)	7.26/19 (1486)	10.15/20 (1866)	13.12/20 (2100)
4	3.20/14 (954)	5.36/18 (986)	6.27/19 (1491)	10.17/20 (1865)	12.83/20 (2107)
6	3.91/15 (958)	5.26/18 (1030)	7.70/19 (1525)	10.11/19 (1882)	11.42/21 (2112)
8	5.37/18 (1015)	6.46/20 (1161)	8.68/20 (1616)	10.68/22 (1931)	21.17/23 (2158)

α and β element-wise randomly chosen in $[10^{-p}, 10^{p}]$. Condition number, number of iterations, and total number of adaptive constraints computed (in parenthesis) are shown for different values of the contrast (i.e. $2p$).

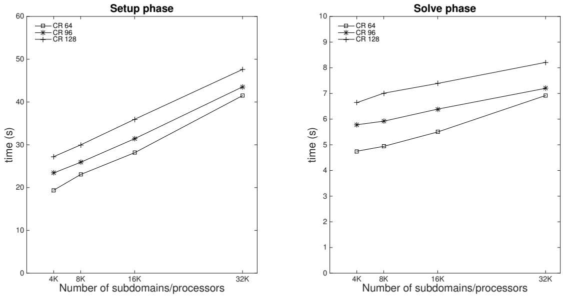
Numerical results: adaptive multilevel BDDC



Weak scaling test, H(div) problem with constant material coefficients. Standard BDDC with direct coarse solver (direct MUMPS) is compared against adaptive multilevel BDDC with coarsening ratio 32 (CR32) (coarse threshold equal to 2). Computational times in seconds for the setup of the preconditioner (left) and for the PCG (right) are shown as a function of the number of processors. 40K dofs/subdomain, 1.1B dofs with 32K cores.

Efficiency at 32K cores: 99%.

Numerical results: adaptive multilevel BDDC



Weak scaling test, H(div) problem with random material coefficients in $[10^{-2}, 10^{2}]$. Adaptive multilevel BDDC with coarsening ratios 64 (CR64), 96 (CR96), and 128 (CR128). (coarse threshold equal to 2). Computational times in seconds for the setup of the preconditioner (left) and for the PCG (right) are shown as a function of the number of processors. 80K dofs/subdomain, 2.1B dofs with 32K cores.

What's next...

- `MatSolveSparse` for speed up setup.
- Further optimizations in multilevel extension.
- Support for `MATIS` from `DMPlex`.
- Support from FENICS (currently under development).
- Mesh partitioning with balanced interfaces.
- Adaptive BDDC for porous media flows and electromagnetic inversion.
- Adaptivity for more general linear systems.
- Design of FETI-DP classes (deluxe scaling and adaptivity support).
- Extensions of Schur complement support to other factorization packages (PARDISO, PASTIX).
- GPU acceleration: Cholesky inversion, primal basis computation, rank-n update.

That's all folks

Thank you for your attention.

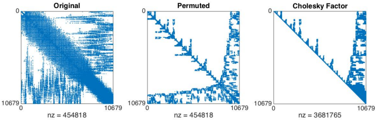
Numerical results: details on GEP

procs	S		S^{-1}		GEP	
	min	max	min	max	min	max
8192	1.02	2.23	0.25	4.82	0.05	0.52
16384	0.98	2.32	0.23	5.28	0.08	0.49
32768	0.94	2.30	0.24	5.57	0.06	0.68

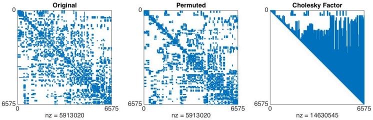
Computational times for the main phases of the adaptive selection of constraints at the finest level in the weak scaling test for H(div) with random material coefficients in $[10^{-2}, 10^{2}]$. Minimum and maximum times in seconds are reported for the explicit computation of the Schur complement (S), its explicit inversion (S^{-1}), and the solution of all the generalized eigenvalue problems (GEP).

Numerical results: details of coarse matrices at coarser level

H(div), const, CR32, 32K cores:



H(div), rand, CR128, 4K cores:



Poisson, const, CR768, 41K cores: