Preconditioning with BDDC and FDM for High Order Finite Elements with libCEED

Jeremy L Thompson[1]
Valeria Barra[1], Yohann Dudouit[2], Oana Marin[3], & Jed Brown[1]

1: University of Colorado Boulder
2: Lawrence Livermore National Laboratory
3: Argonne National Laboratory

jeremy.thompson@colorado.edu

Jan 15, 2020

libCEED Team

Developers: Jed Brown[1], Jeremy Thompson[1]
Thilina Rathnayake[4], Jean-Sylvain Camier[2], Tzanio Kolev[2],
Veselin Dobrev[2], Valeria Barra[1], Yohann Doudouit[2],
David Medina[5], Tim Warburton[6], & Oana Marin[3]

Grant: Exascale Computing Project (17-SC-20-SC)

1: University of Colorado, Boulder
2: Lawrence Livermore National Laboratory
3: Argonne National Laboratory
4: University of Illinois, Urbana-Champaign
5: OCCA
6: Virginia Polytechnic Institute and State University

Overview

High order matrix-free finite elements are less expensive than sparse matrices, with respect to both FLOPS and memory transfer

libCEED's finite element operator decomposition provides performance, portability, and opportunities for flexible preconditioning strategies

Fast Diagonalization Method compliments the libCEED decomposition and provides inexact subdomain solvers for BDDC or ASM preconditioning

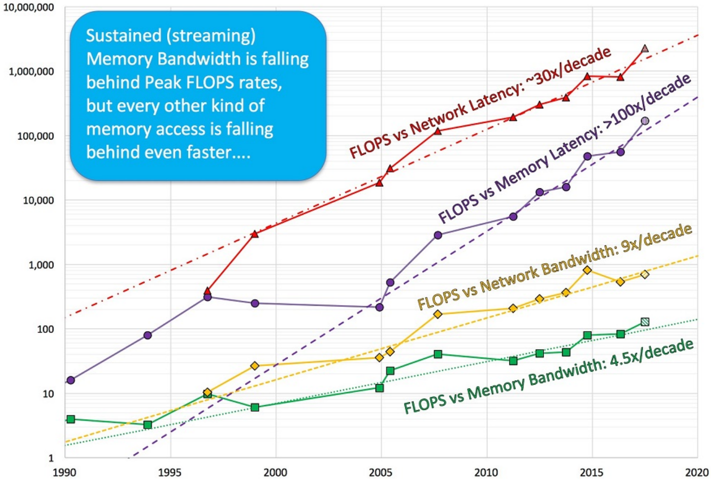
Overview

Center for Efficient Exascale Discretizations

DoE exascale co-design center

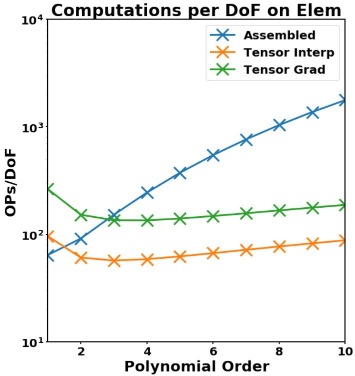
- Design discretization algorithms for exascale hardware that deliver significant performance gain over low order methods
- Collaborate with hardware vendors and software projects for exascale hardware and software stack
- Provide efficient and user-friendly unstructured PDE discretization component for exascale software ecosystem

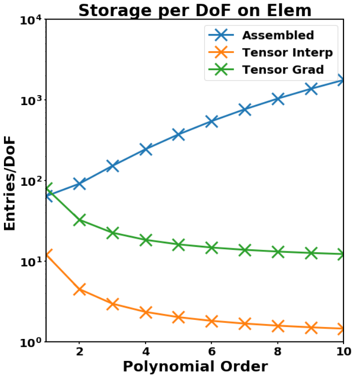
FLOPs vs Bandwidth



Growth of FLOPs outstriping bandwidth for decades, McCalpin SC16

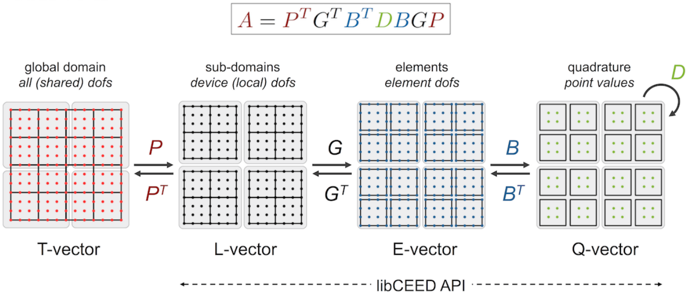
Tensor Product Elements





Matrix-free finite element formulations provide performance optimizations for hexahedral elements

libCEED Operator Decomposition



$$A_L = G^T B^T D B G$$

- G - CeedElemRestriction, local gather/scatter
- B - CeedBasis, provides basis operations such as interp and grad
- D - CeedQFunction, representation of PDE at quadrature points
- A_L - CeedOperator, aggregation of Ceed objects for local action of operator

Laplacian Example

Solving the 2D Poisson problem: $-\Delta u = f$
Weak Form: $\int \nabla v \nabla u = \int vf$

- General libCEED Operator
  $A_L = G^T B^T D B G$

- Laplacian Operator
  $A_L = G^T \nabla B^T D \nabla B G$

where D is block diagonal by quadrature point:

$$D_i = J_{geo}^{-1} \left(w_i \det J_{geo} \right) J_{geo}^{-T} \text{ and } J_{geo} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial s} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial s} \end{bmatrix}$$

x, y physical coords; r, s reference coords

Helmholtz Example

Solving the 2D Inhomogeneous Helmholtz problem: $-\left(\Delta + k^2\right) u = f$
Weak Form: $\int \left(\nabla v \nabla u - k^2 vu\right) = \int vf$

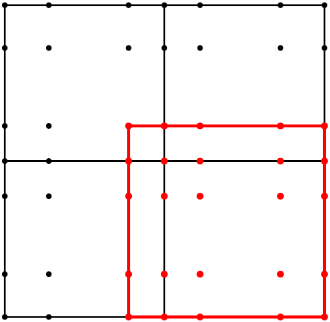
- General libCEED Operator
  $A_L = G^T B^T D B G$

- Helmholtz Operator
  $$A_L = G^T \begin{bmatrix} B \\ \nabla B \end{bmatrix}^T D \begin{bmatrix} B \\ \nabla B \end{bmatrix} G$$
  where D is block diagonal by quadrature point:
  $$D_i = \left(w_i \det J_{geo}\right) \begin{bmatrix} -k^2 & \\ & J_{geo}^{-1} J_{geo}^{-T} \end{bmatrix} \text{ and } J_{geo} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial s} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial s} \end{bmatrix}$$
  x, y physical coords; r, s reference coords

Domain Decomposition Preconditioning

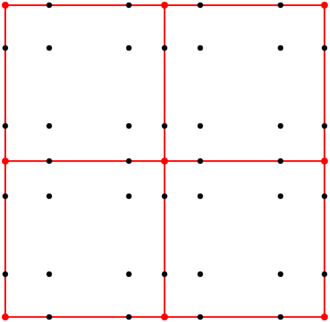
- Preconditioning essential for iterative solvers, especially with high order elements
- P-Multigrid preconditioning offers $O(N)$ elliptic PDE solve
  - Requires careful communication implementation for parallel performance
- Additive Schwartz with high order element subdomains
  - Element halo can be large and complicated in unstructured meshes
- BDDC/FETI eliminate the halo, needs unassembled operator
  - Requires subdomain continuity conditions to converge

Additive Schwartz



- Tradeoff with overlap in convergence and computations/bandwidth
- See Nek5000 pressure solves, work by Fischer, Miller, and Tufo

BDDC



- Global coarse solve gives boundary conditions for subdomain solves
- See BDDC preconditioner in PETSc, work by Zampini

Inexact Subdomain Solves

- Li and Widlund 2007 demonstrated BDDC with inexact subdomain solves using multigrid
- Fast Diagonalization provides inverses of separable operators
  - Used in Nek5000 with Additive Schwartz subdomain solves
- Fast Diagonalization can provide inexact subdomain solves for BDDC
  - Open question: Which non-separable operators can use FDM based inexact subdomain solves?

Fast Diagonalization Method

- Consider 2D Helmholtz problem on a reference element

$$\text{Let } A = \nabla B^T w \nabla B,\ M = B^T w B$$

$$L = A_{2D} - k^2 M_{2D} = A \otimes A - k^2 M \otimes M$$

- Diagonalize 1D element Laplacian and mass matrix

$$P^T A P = \Lambda,\ P^T M P = I$$

- Build element inverse

$$L^{-1} = P \otimes P \left[\Lambda \otimes I + I \otimes \Lambda - k^2 I \otimes I \right]^{-1} P^T \otimes P^T$$

Inverse of Separable Operators

- In general

$$L = cA_{2D} + kM_{2D} = c\left(A_x \otimes A_y\right) + k\left(M_x \otimes M_y\right)$$

- Diagonalize 1D element Laplacians and mass matrices

$$P^T A_x P = \chi \Lambda,\ P^T M_x P = \chi I$$
$$P^T A_y P = \Lambda,\ P^T M_y P = I$$

- Build element inverse

$$L^{-1} = P \otimes P \left[c\left(\chi \Lambda \otimes I\right) + c\left(\chi I \otimes \Lambda\right) + k\left(\chi I \otimes I\right)\right]^{-1} P^T \otimes P^T$$

Nonlinear Coefficients

- Generalized inhomogeneous Helmholtz equation

$$-\left(f_1\left(x\right)\Delta u + k^2 f_0\left(x\right)u\right) = f$$

- Discretized generalized inhomogeneous Helmholtz operator

$$A_e = \begin{bmatrix} B \\ \nabla B \end{bmatrix}^T D \begin{bmatrix} B \\ \nabla B \end{bmatrix}$$

$$\text{where } D_i = \left(w_i \det J_{geo}\right)\begin{bmatrix} -k^2 f_0\left(x\right) & \\ & J_{geo}^{-1} f_1\left(x\right) J_{geo}^{-T} \end{bmatrix}$$

- Approximate element inverse

$$A_e^{-1} \approx P \otimes P \left(\Lambda_g\right)^{-1} P^T \otimes P^T$$
$$\text{where } \Lambda_g = diag\left(P^T \otimes P^T \left(A_e\right) P \otimes P\right)$$

Separable Approximate Inverses

- Weak form of general second order PDE

$$\int \left(\nabla v f_1 \left(\nabla u, u, x \right) + v f_0 \left(\nabla u, u, x \right) \right) = \int v f$$

- Discretized generalized inhomogeneous Helmholtz operator

$$A_e = \begin{bmatrix} B \\ \nabla B \end{bmatrix}^T D \begin{bmatrix} B \\ \nabla B \end{bmatrix}$$

$$\text{where } D_i = \begin{bmatrix} I & \\ & J_{geo}^{-1} \end{bmatrix} \left(w_i \det J_{geo} \right) \begin{bmatrix} f_{0,0} & f_{0,1} \\ f_{1,0} & f_{1,1} \end{bmatrix} \begin{bmatrix} I & \\ & J_{geo}^{-T} \end{bmatrix}$$

- Approximate element inverse

$$A_e^{-1} \approx ???$$

Future Work

- Further performance enhancements (GPU and CPU)
- Improved mixed mesh and operator composition support
- Expanded non-linear and multi-physics examples
- Preconditioning based on libCEED operator decomposition
- Algorithmic differentiation of user quadrature functions
- We invite contributors and friendly users

Questions?

Advisors : Jed Brown[1] & Daniel Appelö[1]

Collaborators: Valeria Barra[1], Oana Marin[2], Tzanio Kolev[3], Jean-Sylvain Camier[3], Veselin Dobrev[3], Yohann Doudouit[3], Tim Warburton[4], David Medina[5], & Thilina Rathnayake[6]

Grant: Exascale Computing Project (17-SC-20-SC)

1: University of Colorado, Boulder
2: Argonne National Laboratory
3: Lawrence Livermore National Laboratory
4: Virginia Polytechnic Institute and State University
5: OCCA
6: University of Illinois, Urbana-Champaign

Preconditioning with BDDC and FDM for High Order Finite Elements with libCEED

Jeremy L Thompson[1]
Valeria Barra[1], Yohann Dudouit[2], Oana Marin[3], & Jed Brown[1]

1: University of Colorado Boulder
2: Lawrence Livermore National Laboratory
3: Argonne National Laboratory

jeremy.thompson@colorado.edu

Jan 15, 2020