Geometric Multigrid for Higher-order Discretizations

Hari Sundar

Georg Stadler, Omar Ghattas, George Biros

Institute for Computational Engineering & Sciences

THE UNIVERSITY OF TEXAS AT AUSTIN

Copper Mountain Multigrid 2013

Motivation

Scalable solution of large-scale linear systems

asymptotically optimal parallel solvers for elliptic PDEs

Motivation

Scalable solution of large-scale linear systems

asymptotically optimal parallel solvers for elliptic PDEs

variable coefficients

adaptive discretizations

arbitrary geometries

high-order discretizations

massively parallel

Motivation

Scalable solution of large-scale linear systems

asymptotically optimal parallel solvers for elliptic PDEs

variable coefficients

adaptive discretizations

arbitrary geometries

high-order discretizations

massively parallel

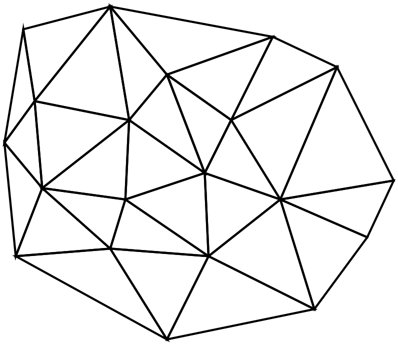
Parallel Geometric Multigrid

$$\mathcal{O}\left(\frac{N}{p} + \log N\right)$$ for elliptic PDEs with smooth coefficients

parallel multigrid

challenges

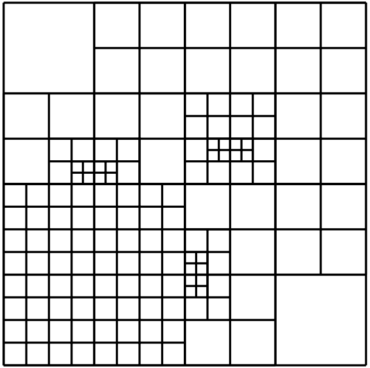
- geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging



parallel multigrid

challenges

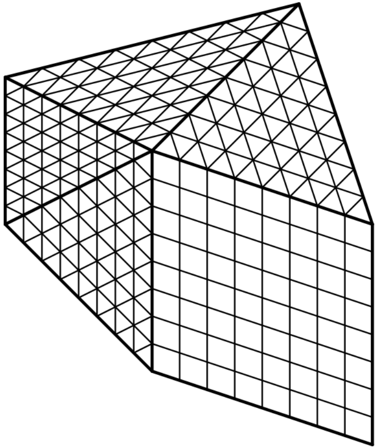
- geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging
- octrees
  - dendro (Sampath et al., SC'08)
  - limited to cubic domains



parallel multigrid

challenges

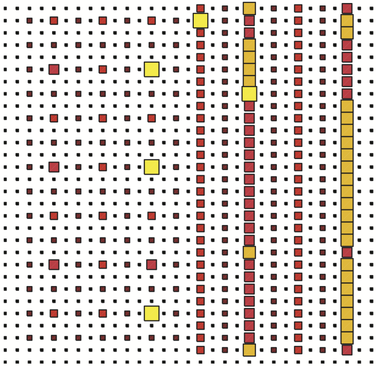
- geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging
- octrees
  - dendro (Sampath et al., SC'08)
  - limited to cubic domains
- two-tier meshes, macromesh + regular grid
  - HHG (Bergen et al., SC'05)
  - limited adaptivity



parallel multigrid

challenges

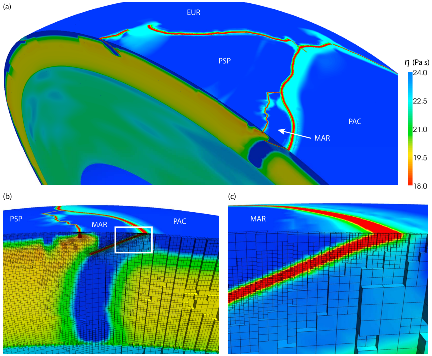
- geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging
- octrees
  - dendro (Sampath et al., SC'08)
  - limited to cubic domains
- two-tier meshes, macromesh + regular grid
  - HHG (Bergen et al., SC'05)
  - limited adaptivity
- algebraic multigrid
  - Adams et al., SC'04
  - Hypre(CHPC'10), `trilinos::ML`
  - graph based coarsening
  - need assembled matrix



Parallel Multigrid

Key Contributions

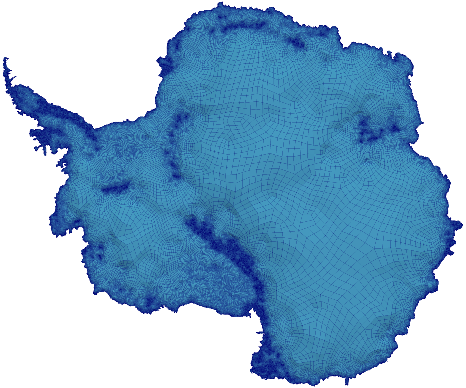
- GMG for complex geometries with adaptivity (macromesh + octrees)
- excellent strong and weak scalability
- low setup cost
- matrix-free implementation using non-blocking MPI calls
- $262K$ cores with single $\mathtt{MPI}$ process per core



Parallel Multigrid

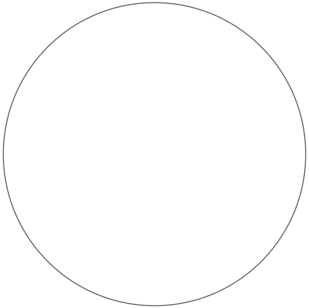
Key Contributions

- GMG for complex geometries with adaptivity (macromesh + octrees)
- excellent strong and weak scalability
- low setup cost
- matrix-free implementation using non-blocking MPI calls
- $262K$ cores with single MPI process per core



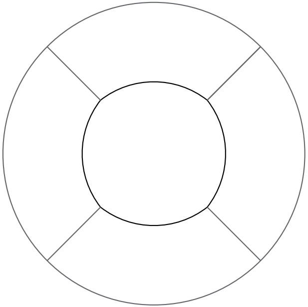
two-tier meshes

conforming macromesh of adaptive octrees



two-tier meshes

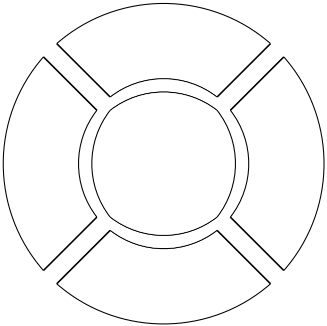
conforming macromesh of adaptive octrees



macromesh

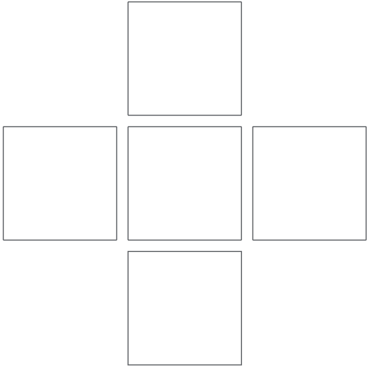
two-tier meshes

conforming macromesh of adaptive octrees



two-tier meshes

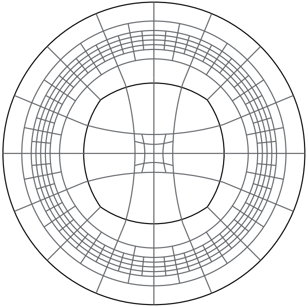
conforming macromesh of adaptive octrees



forest of octrees

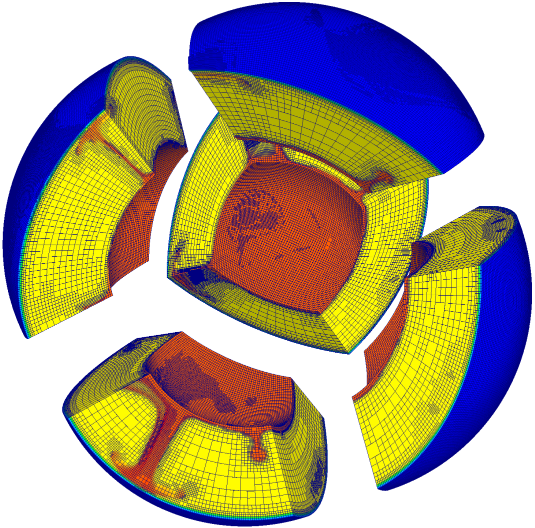
two-tier meshes

conforming macromesh of adaptive octrees



two-tier meshes

conforming macromesh of adaptive octrees

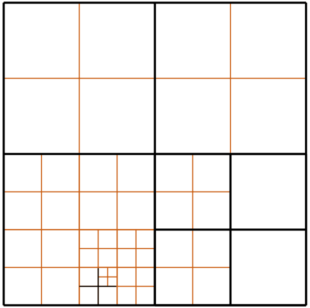


multigrid setup

coarsening

for octrees:

if all siblings exist, replace with parent

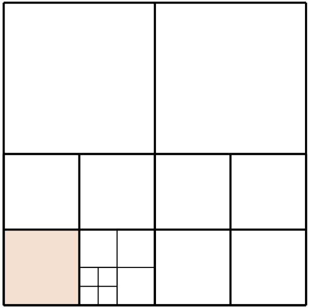


multigrid setup

coarsening

for octrees:

if all siblings exist, replace with parent

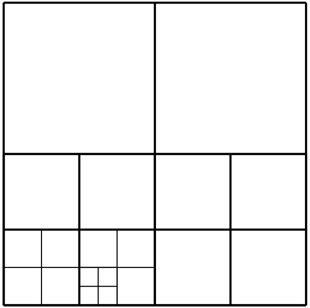


multigrid setup

coarsening

for octrees:

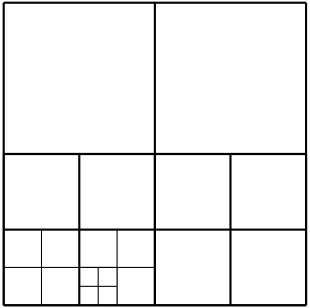
preserve 2:1 balance at all grids



multigrid setup

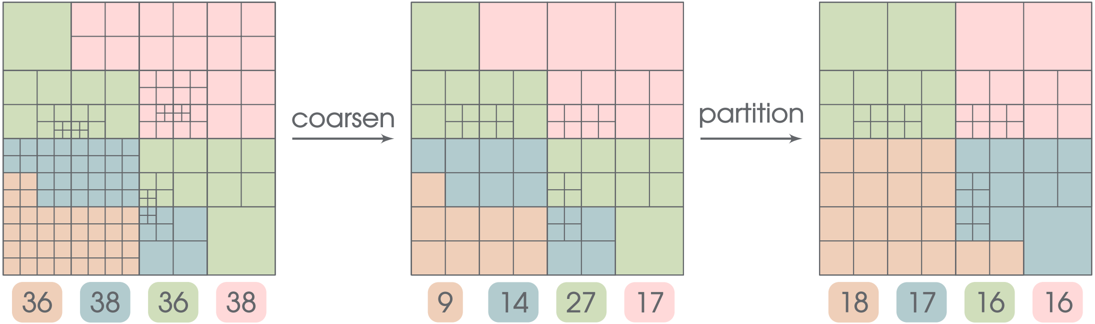
coarsening

for forests: cannot coarsen beyond first-tier macromesh



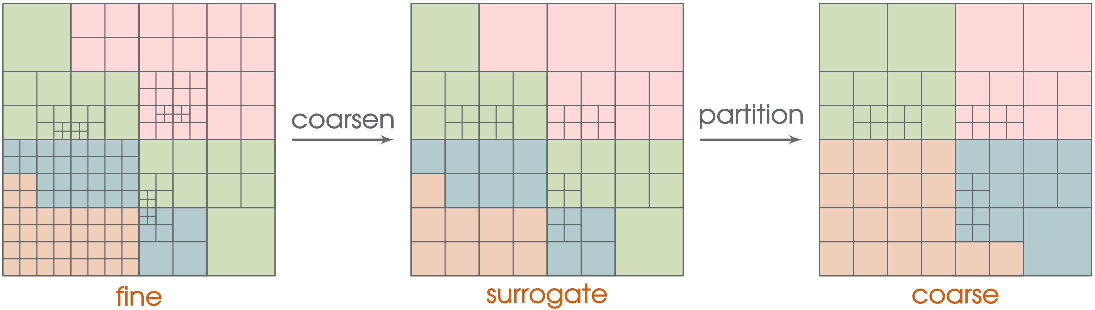
Multigrid Setup

Partitioning & load balancing



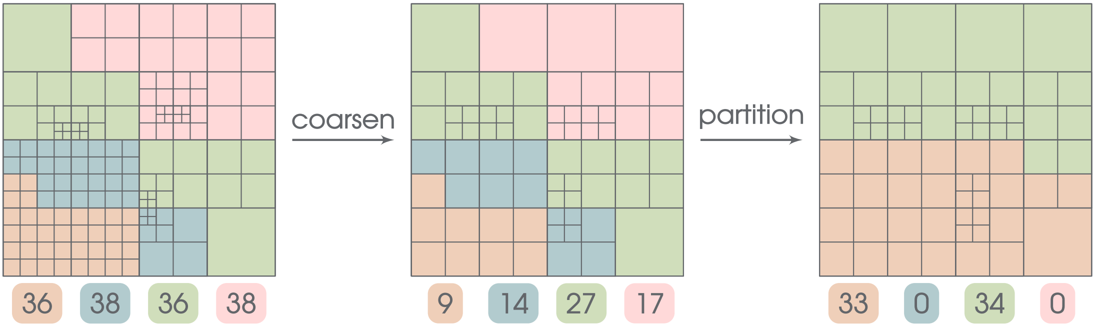
Multigrid Setup

Partitioning & load balancing



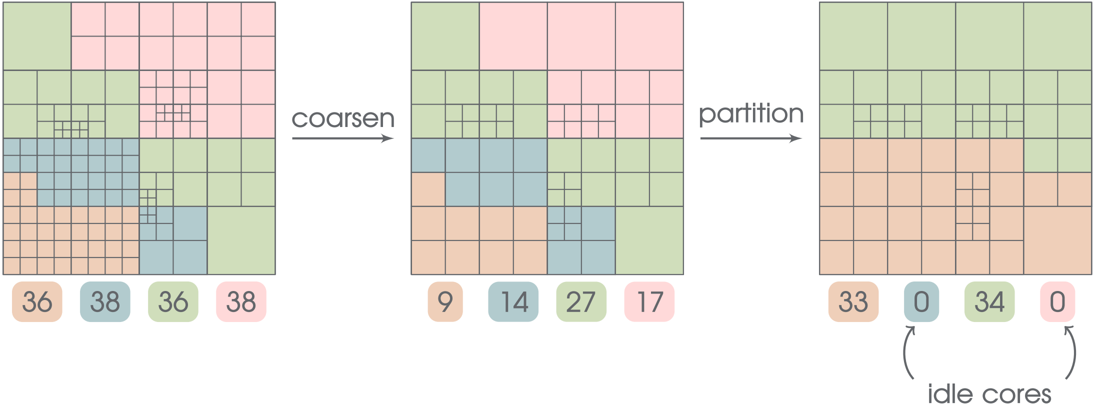
Multigrid Setup

partitioning & load balancing



Multigrid Setup

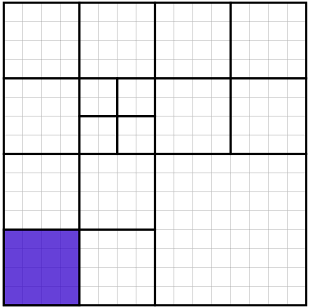
partitioning & load balancing

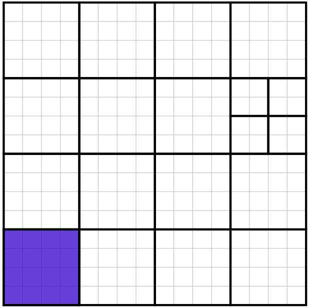


Inter-grid transfer operators

Simultaneous traversal over coarse and fine meshes

- matrix-free implementation
- P_{ij} - coarse grid shape function (ϕ_j) evaluated at the fine grid point (p_i)

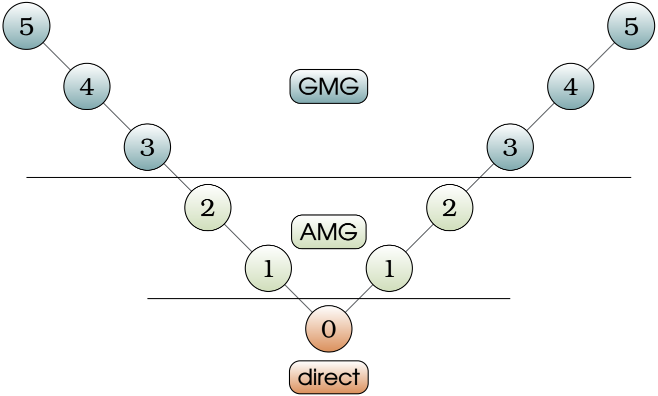




Multigrid Solve

Coarse grid solver

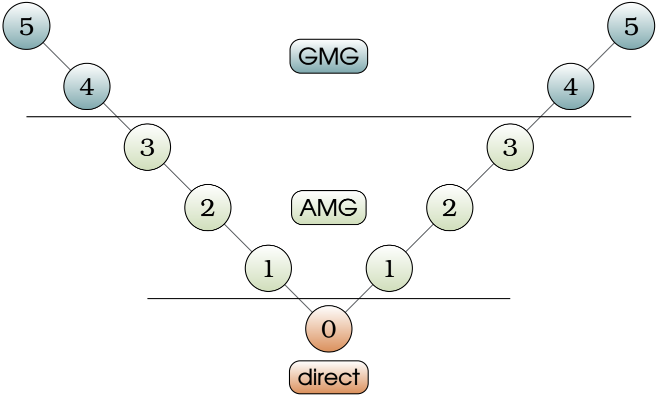
- smoothed aggregation algebraic multigrid (`trilinos::ML`)
- GMG-AMG approach matches our two-tier geometric decomposition of the domain
- AMG is used for small problem sizes on small process counts



Multigrid Solve

Coarse grid solver

- smoothed aggregation algebraic multigrid (`trilinos::ML`)
- GMG-AMG approach matches our two-tier geometric decomposition of the domain
- AMG is used for small problem sizes on small process counts



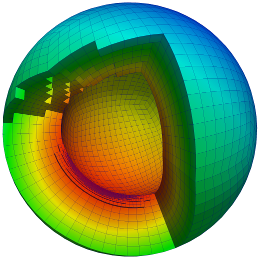
Results

Test problem

$$-\mathrm{div}(\mu(x)\nabla u(x)) = f(x) \qquad \forall x \in \Omega, \qquad u(x) = 0 \text{ on } \partial\Omega.$$

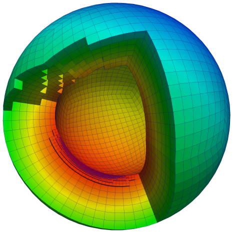
$$\mu(x) = 1 + 10^6\left(e^{-(x-x_1)^2/2\sigma_1^2} + e^{-(x-x_2)^2/2\sigma_2^2}\right)$$

- 3D Poisson problem
- Dirichlet boundary conditions
- isotropic spatially varying coefficient
- forest of 24 Octrees



Results

Strong scaling



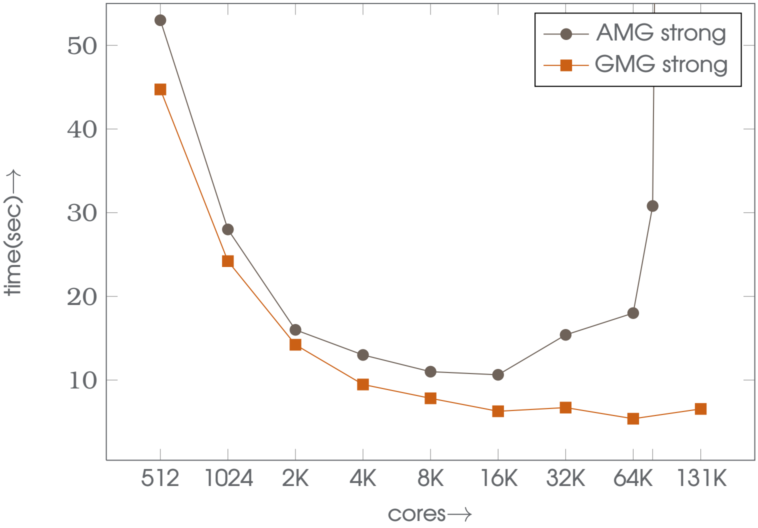
$124M$ elements

5 GMG levels

AMG* for Coarse solve

1 MPI process per core

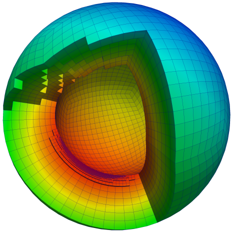
Jaguar $XK6$



*smoothed aggregation (ML)

Results

Weak scaling

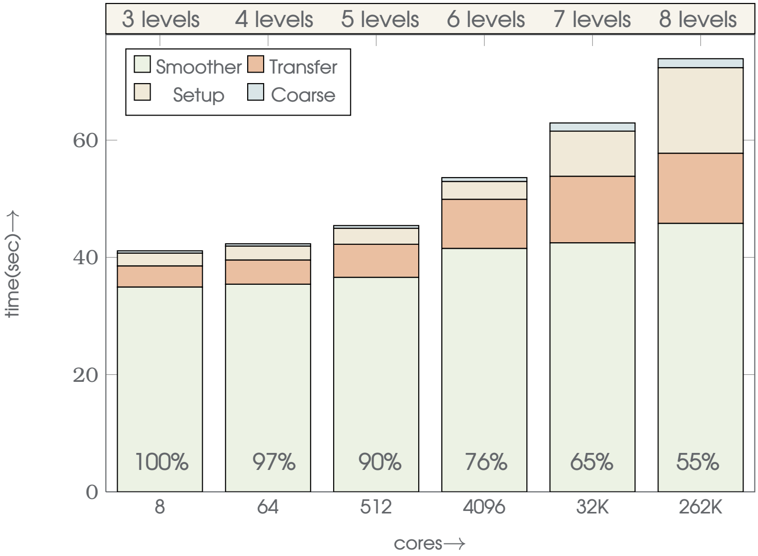


$215K$ elements per process

AMG for Coarse solve

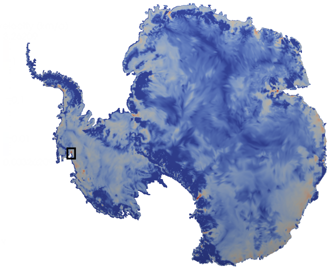
1 $\mathtt{MPI}$ process per core

Jaguar $XK6$



Results

Weak scaling : Antarctica mesh



$45K$ Octrees

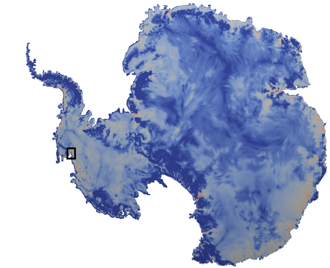
$400K$ elements per process

constant coefficient Poisson

Cores	64	512	4096	32768	262144
Setup	2.97	2.64	3.1	3.76	8.6
Smoother	**289.7**	**301.5**	**336.3**	**391.3**	**409.1**
Transfer	7.45	8.47	11.5	11.35	15.88
Coarse Setup	1.85	2.13	0.82	1.27	1.63
Coarse Solve	24.3	30.8	18.47	30.1	26.01
Total Time	326.3	345.5	370.2	437.8	461.2

Results

Weak scaling : Antarctica mesh



$45K$ Octrees

$400K$ elements per process

constant coefficient Poisson

Cores	64	512	4096	32768	262144
Setup	2.97	2.64	3.1	3.76	8.6
Smoother	**289.7**	**301.5**	**336.3**	**391.3**	**409.1**
Transfer	7.45	8.47	11.5	11.35	15.88
Coarse Setup	1.85	2.13	0.82	1.27	1.63
Coarse Solve	24.3	30.8	18.47	30.1	26.01
Total Time	326.3	345.5	370.2	437.8	461.2

100 Billion unknowns on $262K$ cores while sustaining 272 TFlops/sec.

Motivation

Scalable solution of large-scale linear systems

asymptotically optimal parallel solvers for elliptic PDEs

variable coefficients

adaptive discretizations

arbitrary geometries

high-order discretizations

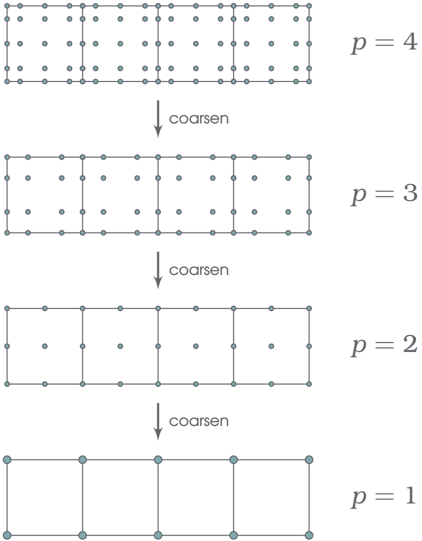
massively parallel

How to support high-order discretizations ?

high-order discretizations

challenges

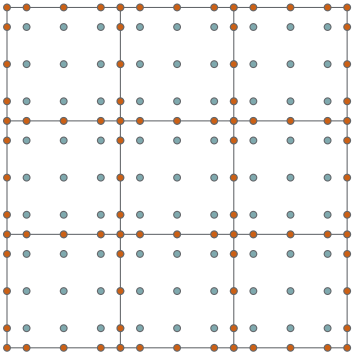
- p-multigrid
  - coarse grid based on lower-order polynomials
  - Use GMG once $p = 1$ has been reached



high-order discretizations

challenges

- p-multigrid
  - coarse grid based on lower-order polynomials
  - Use GMG once $p = 1$ has been reached
- Schwarz-based methods
  - expensive local solves (deformed elements)
  - parallel scalability of coarse grid solve



Local Solve

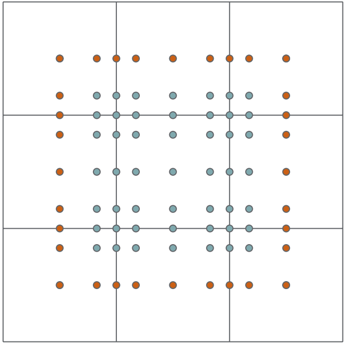
Dirichlet boundary node

unknowns

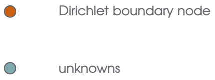
high-order discretizations

challenges

- p-multigrid
  - coarse grid based on lower-order polynomials
  - Use GMG once $p = 1$ has been reached
- Schwarz-based methods
  - expensive local solves (deformed elements)
  - parallel scalability of coarse grid solve



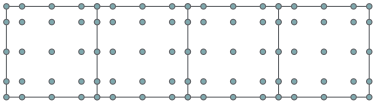
Schwarz Solve



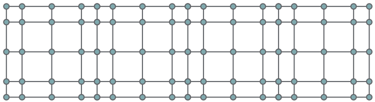
high-order discretizations

challenges

- p-multigrid
  - coarse grid based on lower-order polynomials
  - Use GMG once $p = 1$ has been reached
- Schwarz-based methods
  - expensive local solves (deformed elements)
  - parallel scalability of coarse grid solve
- precondition using lower-order operator
  - Spectrally equivalent
  - lower convergence at higher orders



$p = 4$

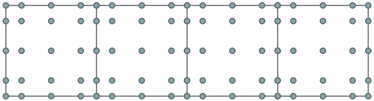


$p = 1$ collocated with $p = 4$

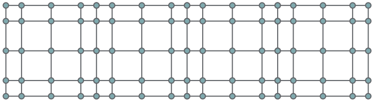
high-order discretizations

challenges

- p-multigrid
  - coarse grid based on lower-order polynomials
  - Use GMG once $p = 1$ has been reached
- Schwarz-based methods
  - expensive local solves (deformed elements)
  - parallel scalability of coarse grid solve
- precondition using lower-order operator
  - Spectrally equivalent
  - lower convergence at higher orders
- h-multigrid using high-order discretizations
  - Use AMG (Heys *et al.*)



$p = 4$



$p = 1$ collocated with $p = 4$

high-order discretizations

precondition using lower-order operator

Number of pCG iterations to converge to a relative tolerance of 10^{-8}

order	8×8	32×32	128×128	$8 \times 8 \times 8$
1*	4	4	4	5
2	14	14	14	25
3	21	21	21	45
4	30	30	30	79
5	43	43	43	144
6	64	65	65	260
7	98	99	99	465
8	142	146	148	840

high-order discretizations

h-multigrid using high-order discretizations

- Coarsening, inter-grid operators & coarse-grid solvers are straightforward

high-order discretizations

h-multigrid using high-order discretizations

THE UNIVERSITY OF TEXAS AT AUSTIN

- Coarsening, inter-grid operators & coarse-grid solvers are straightforward
- Smoothers
  - Jacobi
  - chebyshev
  - Symmetric SOR

high-order discretizations

h-multigrid using high-order discretizations

- Coarsening, inter-grid operators & coarse-grid solvers are straightforward
- Smoothers
  - Jacobi
  - chebyshev
  - Symmetric SOR
- MG as a solver or with PCG

high-order discretizations

h-multigrid using high-order discretizations

Number of CG iterations/v-cycles to converge to a relative tolerance of 10^{-8}

order	Multigrid			MG pCG		
	Jacobi(3)	Chebyshev(3)	SSOR(2)	Jacobi(3)	Chebyshev(3)	SSOR(2)
1	5	6	4	4	4	4
2	6	10	4	5	6	4
3	8	24	5	6	11	4
4	-	54	8	-	16	6
5	-	178	13	-	28	8
6	-	-	24	-	51	12
7	-	-	70	-	105	19
8	-	-	270	-	204	36

high-order discretizations

h-multigrid using high-order discretizations

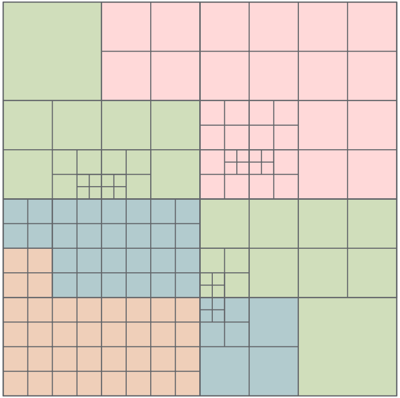
Number of CG iterations/v-cycles to converge to a relative tolerance of 10^{-8}

order	Multigrid			MG pCG			pCG
	Jacobi(3)	Chebyshev(3)	SSOR(2)	Jacobi(3)	Chebyshev(3)	SSOR(2)	linear
1	5	6	4	4	4	4	4
2	6	10	4	5	6	4	14
3	8	24	5	6	11	4	21
4	-	54	8	-	16	6	30
5	-	178	13	-	28	8	43
6	-	-	24	-	51	12	65
7	-	-	70	-	105	19	99
8	-	-	270	-	204	36	148

high-order discretizations

parallelizing SSOR - Octree Coloring

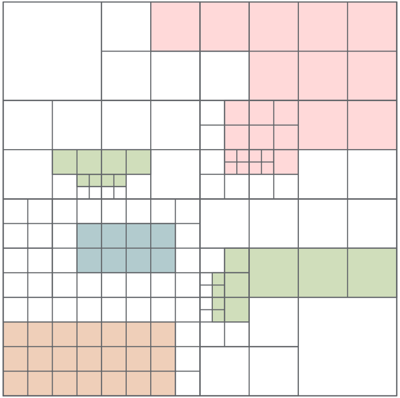
- block parallel SOR
- need coloring to define ordering
- Only boundary nodes need to be colored
- Coloring for balanced octree boundaries
  - Similar to Quadtree coloring
- Luby, LDF, SDL
- SDL $\longrightarrow$ 5 balanced colors



high-order discretizations

parallelizing SSOR - Octree Coloring

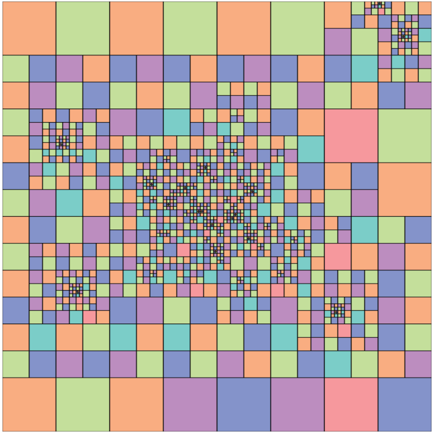
- block parallel SOR
- need coloring to define ordering
- Only boundary nodes need to be colored
- Coloring for balanced octree boundaries
  - Similar to Quadtree coloring
- Luby, LDF, SDL
- SDL $\rightarrow$ 5 balanced colors



high-order discretizations

parallelizing SSOR - Octree Coloring

- block parallel SOR
- need coloring to define ordering
- Only boundary nodes need to be colored
- Coloring for balanced octree boundaries
  - Similar to Quadtree coloring
- Luby, LDF, SDL
- SDL $\rightarrow$ 5 balanced colors



Summary

THE UNIVERSITY OF TEXAS AT AUSTIN

- parallel, matrix-free multigrid method on geometry-conforming unstructured forests of octrees
- pCG using SSOR as a smoother capable of supporting high-order discretizations
- demonstrated strong scalability from 512 to $131K$ cores
- demonstrated weak scalability up to $262K$ cores using one MPI process per core
- largest solve was on a mesh with $45K$ octrees with 100 billion unknowns on $262K$ cores sustaining 272 TFlops/s
- Ongoing work to parallelize support for high-order discretizations

Thank you !