# Parallel Geometric-Algebraic Multigrid on Unstructured Forests of Octrees

Hari Sundar
George Biros, Carsten Burstedde, Johann Rudi
Omar Ghattas, Georg Stadler

Institute for Computational Engineering & Sciences
THE UNIVERSITY OF TEXAS AT AUSTIN

SC12 Salt Lake City, Utah

# Motivation

THE UNIVERSITY OF TEXAS AT AUSTIN

asymptotically optimal parallel solvers for elliptic PDEs

- variable coefficients
- adaptive discretizations
- arbitrary geometries

# Motivation

THE UNIVERSITY OF TEXAS AT AUSTIN

asymptotically optimal parallel solvers for elliptic PDEs

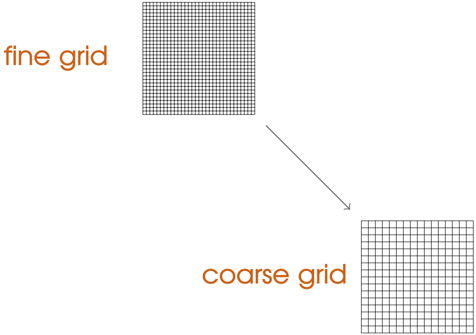
- variable coefficients
- adaptive discretizations
- arbitrary geometries

## Parallel Geometric Multigrid

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

# Multigrid

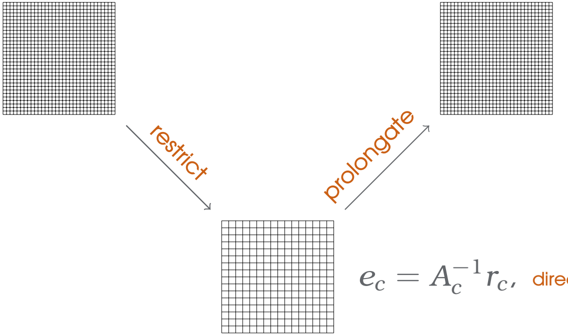
Solve $Au = f$ using two grids



# Multigrid

Solve $Au = f$ using two grids

Smooth ( $u, f$ )
$r = f - Au$



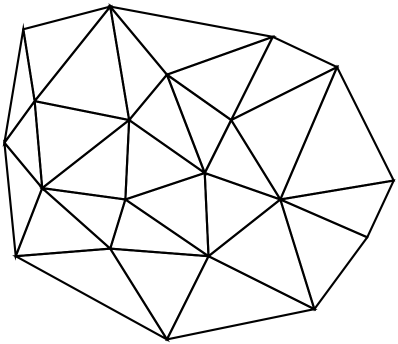

$u = u + e$, correct

$e_c = A_c^{-1} r_c$, direct solve

# Parallel Multigrid

## Challenges

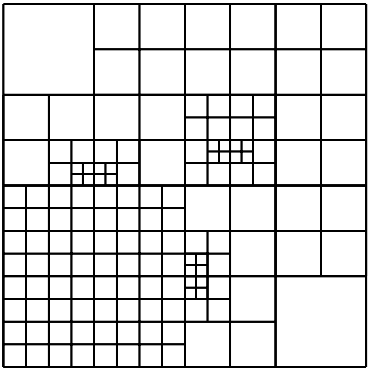
- Geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging



# Parallel Multigrid

## Challenges

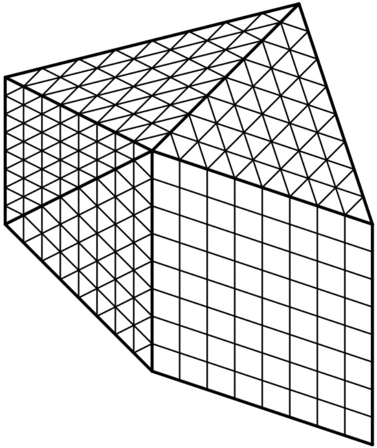
- Geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging
- Octrees
  - Dendro (Sampath et al., SC'08)
  - limited to cubic domains



# Parallel Multigrid

## Challenges

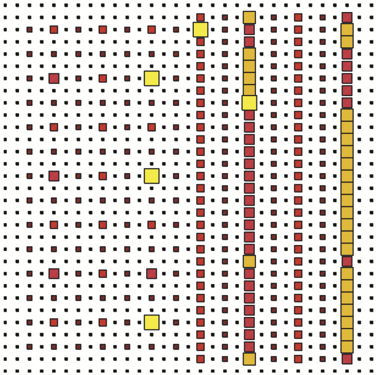
- Geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging
- Octrees
  - Dendro (Sampath et al., SC'08)
  - limited to cubic domains
- Two-tier meshes, macromesh + regular grid
  - HHG (Bergen et al., SC'05)
  - limited adaptivity



# Parallel Multigrid

## Challenges

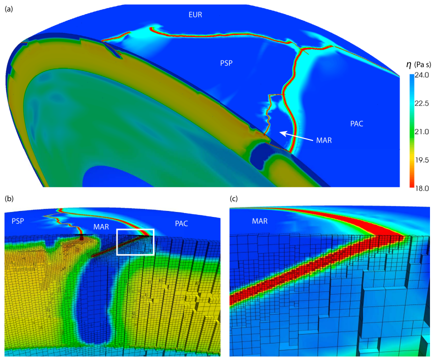
- Geometric multigrid for arbitrary meshes
  - graph based partitioning (ParMETIS SC'98, SC'00)
  - scalability is challenging
- Octrees
  - Dendro (Sampath et al., SC'08)
  - limited to cubic domains
- Two-tier meshes, macromesh + regular grid
  - HHG (Bergen et al., SC'05)
  - limited adaptivity
-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



# Parallel Multigrid

## Key Contributions

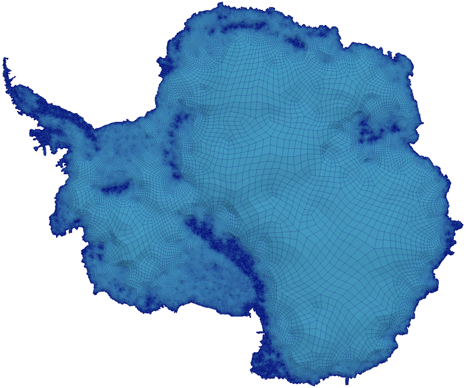
- GMG for complex geometries with adaptivity (macromesh + octrees)
- excellent strong and weak scalability
- low setup cost
- matrix-free implementation using non-blocking MPI calls
- $262K$ cores with single $\mathrm{MPI}$ process per core



# Parallel Multigrid

## Key Contributions

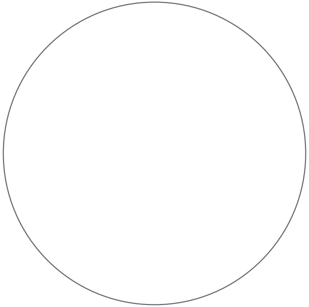
- GMG for complex geometries with adaptivity (macromesh + octrees)
- excellent strong and weak scalability
- low setup cost
- matrix-free implementation using non-blocking MPI calls
- $262K$ cores with single `MPI` process per core



# Two-tier Meshes

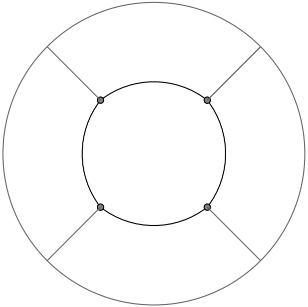


## Conforming macromesh of adaptive octrees



# Two-tier Meshes

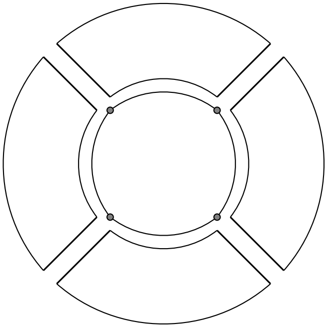
Conforming macromesh of adaptive octrees



macromesh

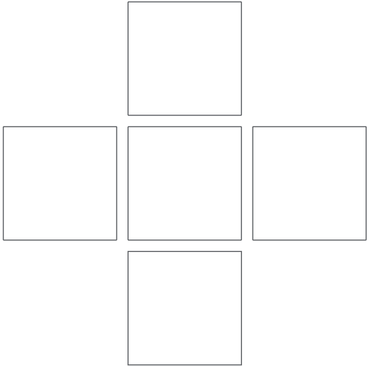
# Two-tier Meshes

## Conforming macromesh of adaptive octrees



# Two-tier Meshes

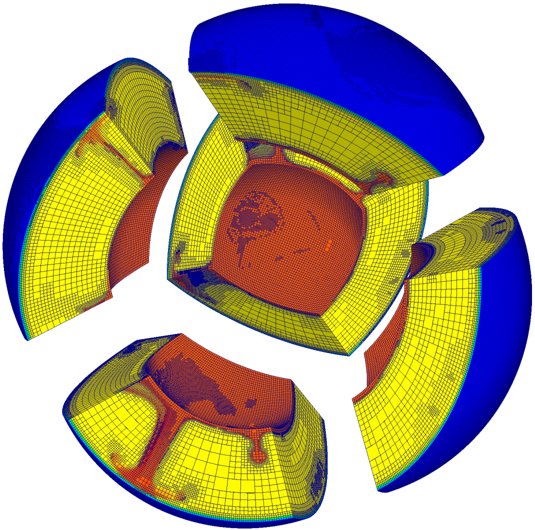
## Conforming macromesh of adaptive octrees



forest of octrees

# Two-tier Meshes

Conforming macromesh of adaptive octrees



# Parallel Geometric Multigrid on Forests

## Overall algorithm

$$-\text{div}(\mu(x)\nabla u(x)) = f(x), \qquad Au = f.$$

**Input:** fine mesh (forest), $\mu(x)$, $f(x)$

**Output:** $u(x)$

# Parallel Geometric Multigrid on Forests



## Overall algorithm

$$-\text{div}(\mu(x)\nabla u(x)) = f(x), \qquad Au = f.$$

**Input:** fine mesh (forest), $\mu(x)$, $f(x)$
**Output:** $u(x)$

**setup : build multigrid hierarchy**

**for** $i \leftarrow 1$ : number of GMG levels
surrogate $\leftarrow$ **coarsen** (fine)
coarse $\leftarrow$ **partition** (surrogate) -- for load-balance
fine $\leftarrow$ coarse

# Parallel Geometric Multigrid on Forests

## Overall algorithm

$$-\text{div}(\mu(x)\nabla u(x)) = f(x), \qquad Au = f.$$

**Input:** fine mesh (forest), $\mu(x)$, $f(x)$

**Output:** $u(x)$

**solve : iterate till convergence,**

$u \leftarrow$ v-cycle (grid, $u$, $A$, $f$)

$u \leftarrow$ **smooth** ($u$, $A$, $f$)

$r \leftarrow f - Au$

$r_c \leftarrow Rr$ ( **restriction** )

$e_c \leftarrow$ v-cycle (grid.coarse, $e_c$, $A$, $r_c$)

$e \leftarrow Pe_c$ ( **prolongation** )

$u \leftarrow u + e$

$u \leftarrow$ **smooth** ($u$, $A$, $f$)

# Parallel Geometric Multigrid on Forests

## Overall algorithm

$$-\text{div}(\mu(x)\nabla u(x)) = f(x), \qquad Au = f.$$

**Input:** fine mesh (forest), $\mu(x)$, $f(x)$
**Output:** $u(x)$

**solve : iterate till convergence,**

$u \leftarrow$ v-cycle (grid, $u$, $A$, $f$)

| | | | |
|---|---|---|---|
| $u$ | $\leftarrow$ | $\omega-$jacobi ($u$, $A$, $f$) | $\mathcal{O}(N/p)$ |
| $r$ | $\leftarrow$ | $f - Au$ | $\mathcal{O}(N/p)$ |
| $r_c$ | $\leftarrow$ | $Rr$ ( **restriction** ) | |
| $e_c$ | $\leftarrow$ | v-cycle (grid.coarse, $e_c$, $A$, $r_c$) | |
| $e$ | $\leftarrow$ | $Pe_c$ ( **prolongation** ) | |
| $u$ | $\leftarrow$ | $u + e$ | $\mathcal{O}(N/p)$ |
| $u$ | $\leftarrow$ | $\omega-$jacobi ($u$, $A$, $f$) | |

# Parallel Geometric Multigrid on Forests

Overall algorithm

$$-\text{div}(\mu(x)\nabla u(x)) = f(x), \qquad Au = f.$$

**Input:** fine mesh (forest), $\mu(x)$, $f(x)$
**Output:** $u(x)$

**setup : build multigrid hierarchy**

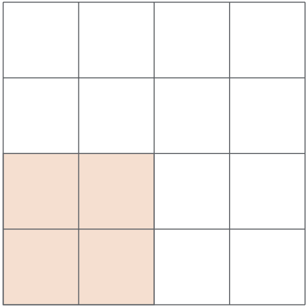
**for** $i \leftarrow 1$ : number of GMG levels
    surrogate $\leftarrow$ **coarsen** (fine)
    coarse $\leftarrow$ **partition** (surrogate) -- for load-balance
    fine $\leftarrow$ coarse

# Multigrid Setup

## Coarsening

for regular grids:

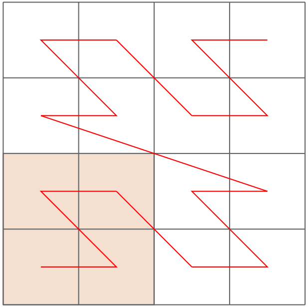
replace $2^d$ siblings with parent



# Multigrid Setup

## Coarsening

for regular grids:
replace $2^d$ siblings with parent



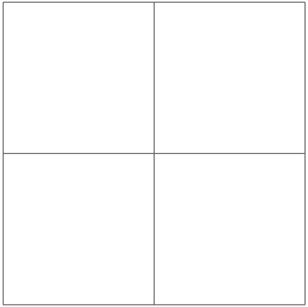
# Multigrid Setup

## Coarsening

THE UNIVERSITY OF TEXAS AT AUSTIN

for regular grids:

replace $2^d$ siblings with parent

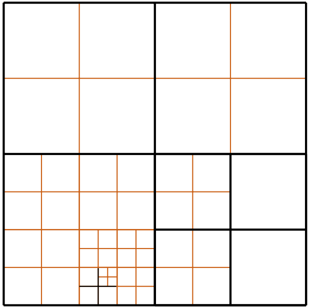


# Multigrid Setup

## Coarsening

for octrees:

if all siblings exist, replace with parent

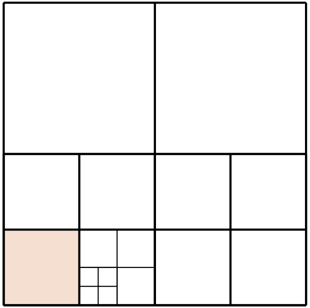


# Multigrid Setup

## Coarsening

for octrees:

if all siblings exist, replace with parent

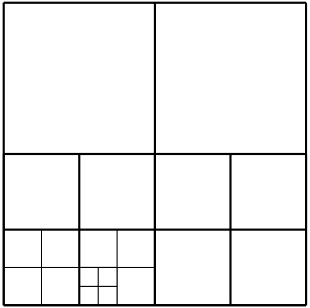


# Multigrid Setup

## Coarsening

for octrees:

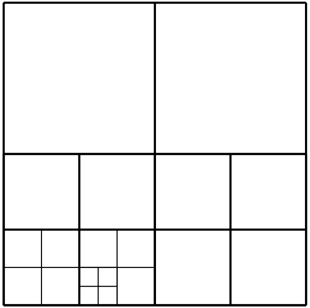
preserve 2:1 balance at all grids



# Multigrid Setup

## Coarsening

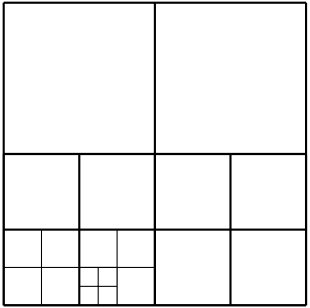
for forests: cannot coarsen beyond first-tier macromesh



# Multigrid Setup

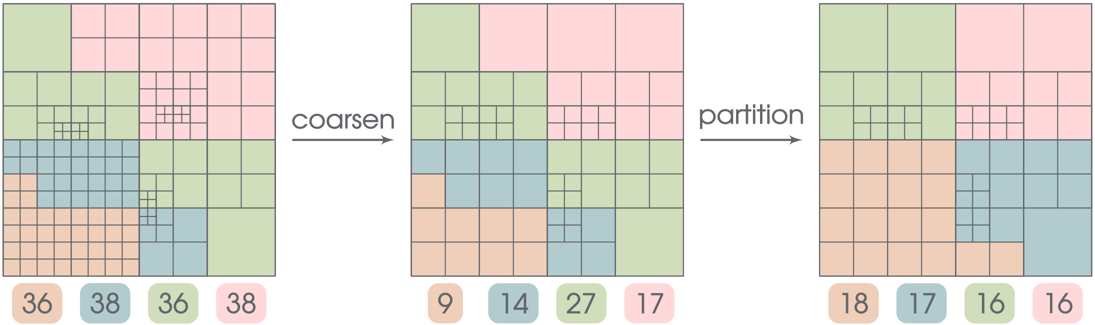
## Coarsening

Complexity: $\mathcal{O}(N/p)$



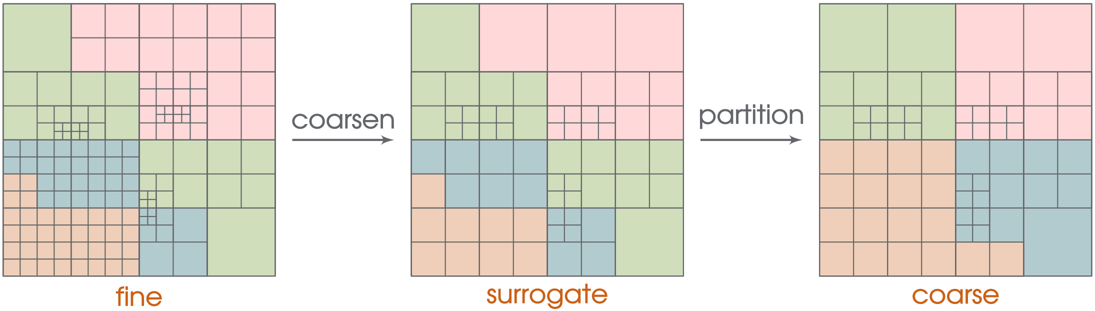
# Multigrid Setup

## Partitioning & load balancing



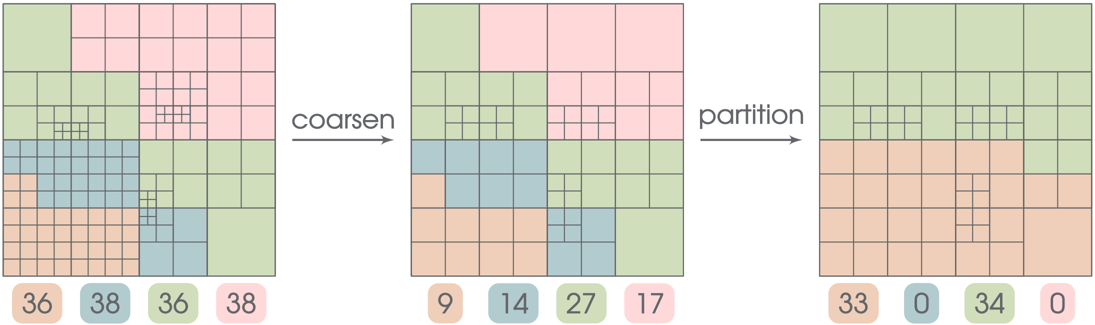
# Multigrid Setup

## Partitioning & load balancing



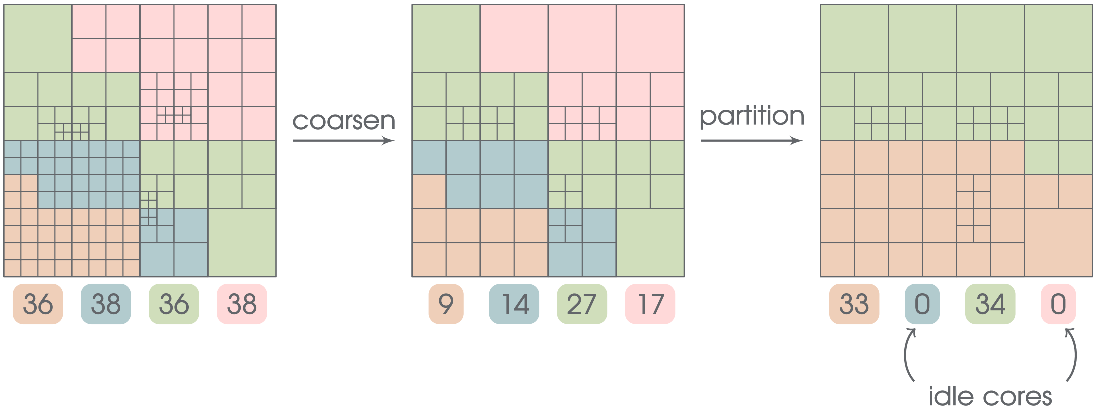
# Multigrid Setup

## partitioning & load balancing



# Multigrid Setup

partitioning & load balancing



# Multigrid Solve

Inter-grid transfer operators

## prolongation (coarse to fine)

- preserve every coarse-grid vector on the fine-grid,

$$Pv = v \qquad \forall v \in V_c \in V_f.$$

- matrix entries: coarse grid shape functions evaluated at the fine grid points,

$$P(i,j) = \phi_j^c(f_i).$$

## restriction (fine to coarse)

transpose of prolongation

- matrix-free implementation
- performed between fine and surrogate meshes
- *no intergrid element searches or look-up tables are needed*
- single simultaneous traversal over both meshes

# Multigrid Solve

Inter-grid transfer operators

## prolongation (coarse to fine)

- preserve every coarse-grid vector on the fine-grid,

$$Pv = v \qquad \forall v \in V_c \in V_f.$$

- matrix entries: coarse grid shape functions evaluated at the fine grid points,

$$P(i,j) = \phi_j^c(f_i).$$

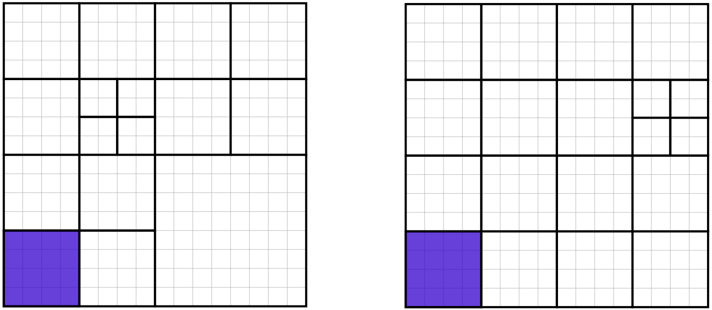
## restriction (fine to coarse)

transpose of prolongation

- matrix-free implementation
- performed between fine and surrogate meshes
- *no intergrid element searches or look-up tables are needed*
- single simultaneous traversal over both meshes -- $\mathcal{O}(N/p)$

# Multigrid Solve

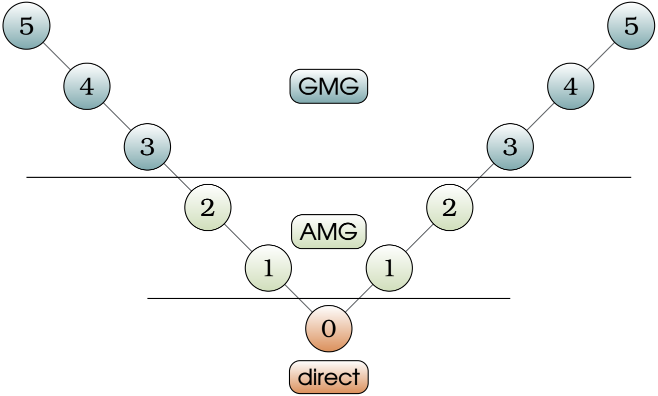
Simultaneous traversal over coarse and fine meshes



# Multigrid Solve

## Coarse grid solver

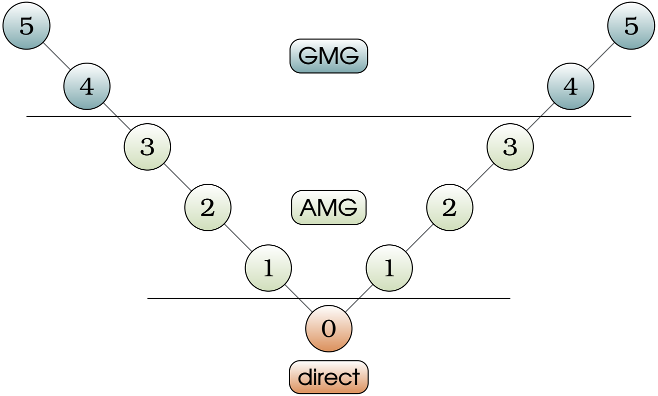
- smoothed aggregation algebraic multigrid (`trilinos::ML`)
- GMG-AMG approach matches our two-tier geometric decomposition of the domain
- AMG is used for small problem sizes on small process counts



# 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



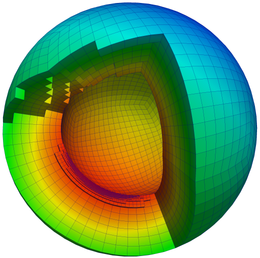
# Results

## Test problem

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

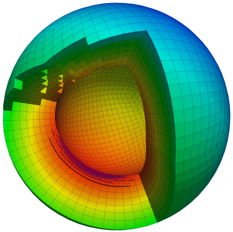
$$\mu(x) = 10^6(1 + e^{-(x-x_1)^2/2\sigma_1^2} + e^{-(x-x_2)^2/2\sigma_2^2})$$

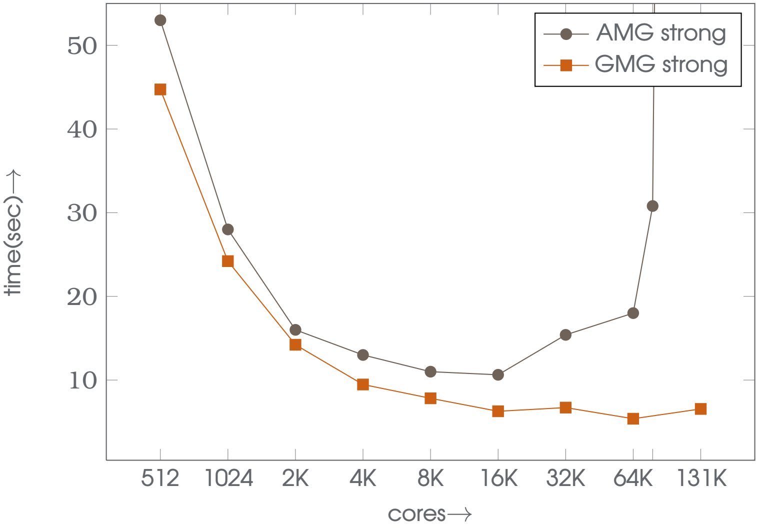
- 3D Poisson problem
- Dirichlet boundary conditions
- isotropic spatially varying coefficient
- forest of 24 Octrees



# Results

## Strong scaling



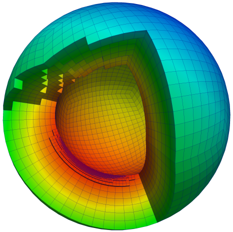



- $124M$ elements
- 5 GMG levels
- AMG* for Coarse solve
- 1 MPI process per core
- Jaguar $XK6$

* smoothed aggregation (ML)

# Results

## Weak scaling

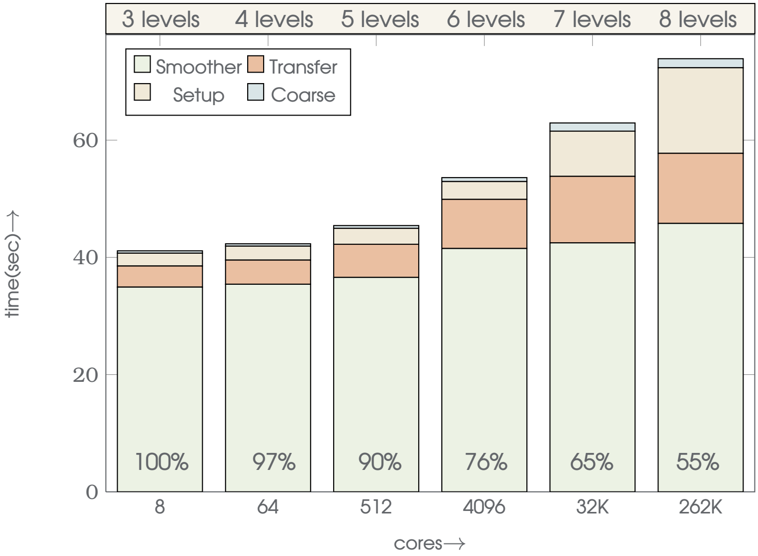


$215K$ elements per process

AMG for Coarse solve

1 MPI process per core

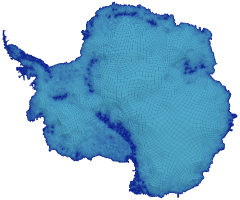
Jaguar $XK6$



# Results



## Weak scaling : Antarctica mesh



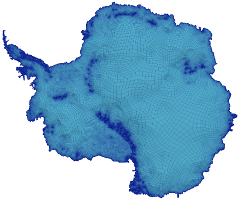
$45K$ Octrees
$400K$ elements per process
constant coefficient Poisson

| 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

THE UNIVERSITY OF TEXAS AT AUSTIN

## 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.

# Limitations

- limitations of the macromesh
  - limited to unstructured hexahedral meshes
  - scalability of coarse solver
- anisotropy
  - parallel plane and line smoothers
  - harder to identify in octrees
- jumping coefficients
  - coefficient aware inter-grid operators
- extend to higher-order discretizations

# Summary

THE UNIVERSITY OF TEXAS AT AUSTIN

- parallel, matrix-free multigrid method on geometry-conforming unstructured forests of octrees
- v-cycle implementation uses only non-blocking point-to-point communications
- 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

Thank you !