Partitioning for Multigrid Solvers

Chris Siefert

Erik Boman, Karen Devine, Jonathan Hu and Ray Tuminaro

Sandia National Laboratories

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company,
for the United States Department of Energy's National Nuclear Security Administration
under contract DE-AC04-94AL85000.

Outline

- Introducing Trilinos/ML.
- Multigrid in four slides or less.
- Role of (re-)partitioning and Zoltan.
- Challenges for the future.

Trilinos (Part of TOPS-2)

Core

- Teuchos
- Zoltan
- Thyra
- RTOp
- Epetra
- EpetraExt
- ForTrilinos
- Isorropia

Discretizations

- Intrepid
- phdMesh
- Rythmos

Solvers

- AztecOO
- ML
- NOX
- Meros
- LOCA
- Ifpack
- Amesos
- Anasazi
- Belos
- Moocho

Methods

- Moertel
- Sacado

Trilinos (Part of TOPS-2)

Core

- Teuchos
- Zoltan
- Thyra
- RTOp
- Epetra
- EpetraExt
- ForTrilinos
- Isorropia

Discretizations

- Intrepid
- phdMesh
- Rythmos

Solvers

- AztecOO
- ML
- NOX
- Meros
- LOCA
- Ifpack
- Amesos
- Anasazi
- Belos
- Moocho

Methods

- Moertel
- Sacado

Sandia National Laboratories

ML Features

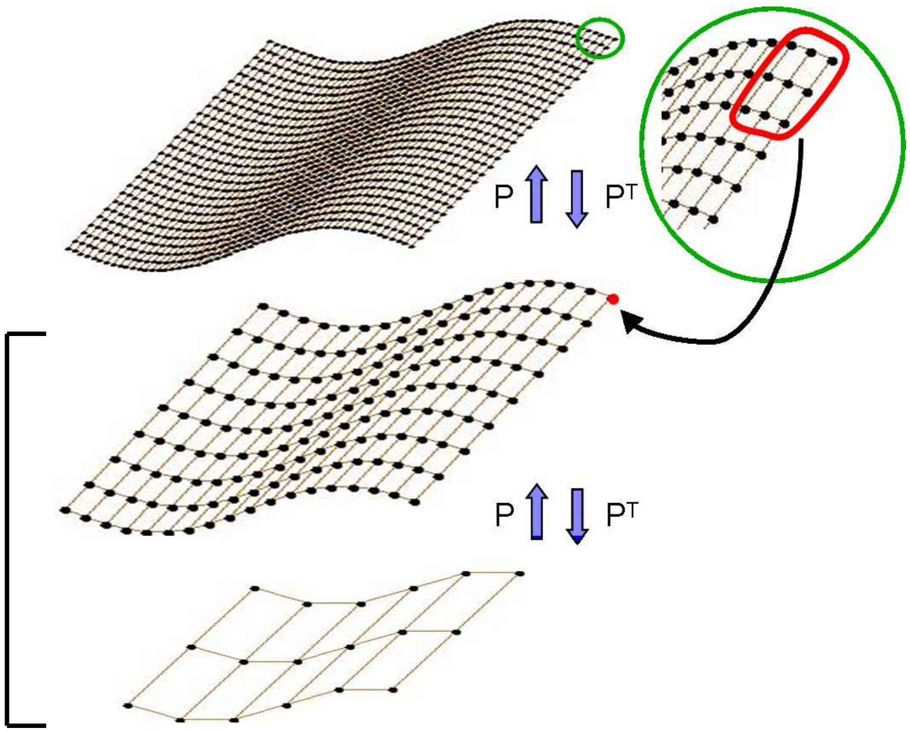
- ML provides scalable multilevel/multigrid preconditioners.
- Method types
  - Smoothed Aggregation (SA) - symmetric or nearly symmetric problems.
  - Non-symmetric SA - non-symmetric problems.
  - MatrixFree - matrix-free SA.
  - DD / DD-ML - domain decomposition.
  - Maxwell - Maxwell's equations.
  - RefMaxwell - new method for Maxwell's equations.

Multigrid in Four Slides or Less

- Goal: Solve problem with specified mesh.
- Idea #1: Use cheap smoother to reduce oscillatory error.
- Idea #2: Use coarse grid to approximate low energy error.

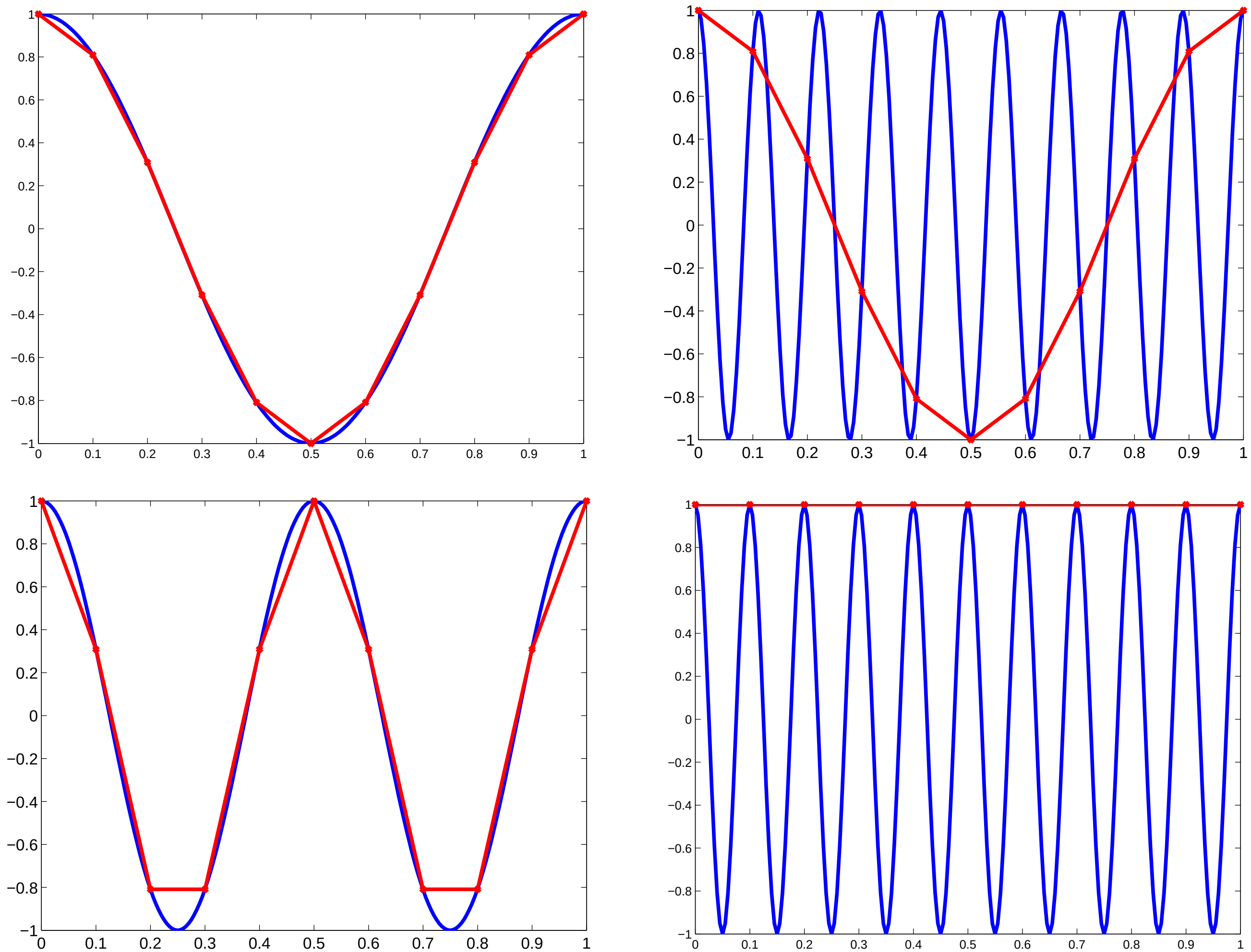
Provided by User

Computed by Multigrid



Low Energy Error

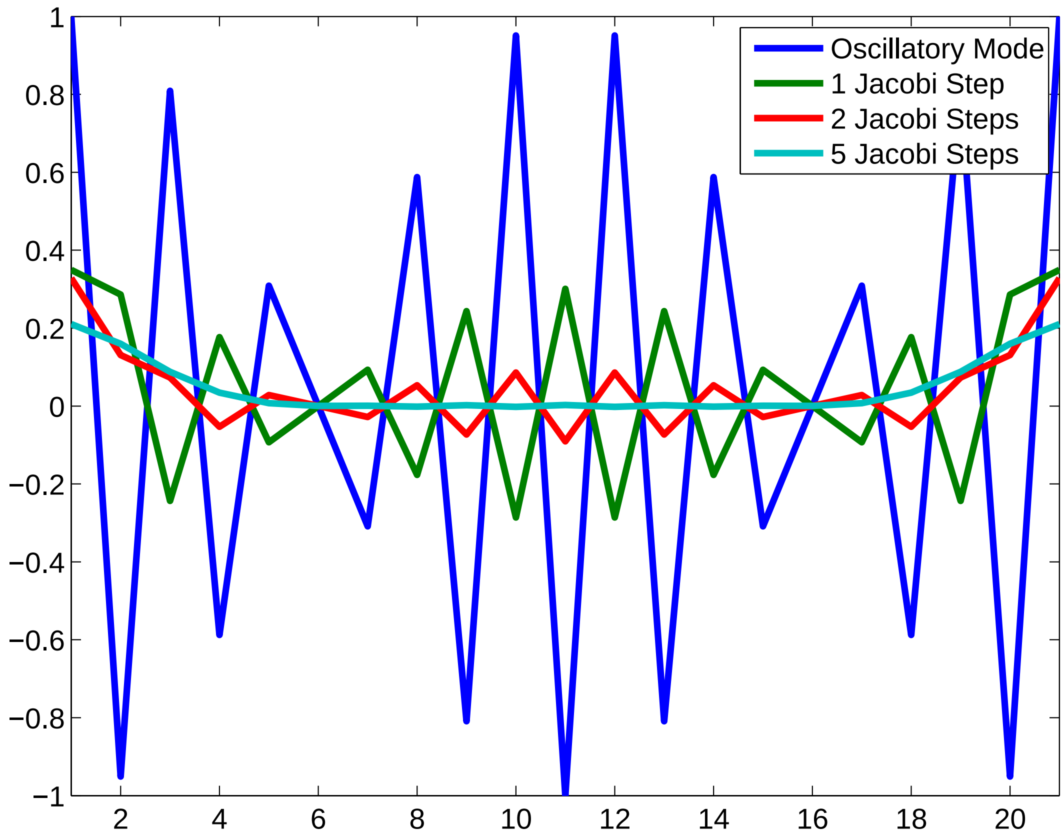
- Coarse grids approximate low energy error well.



Sandia National Laboratories

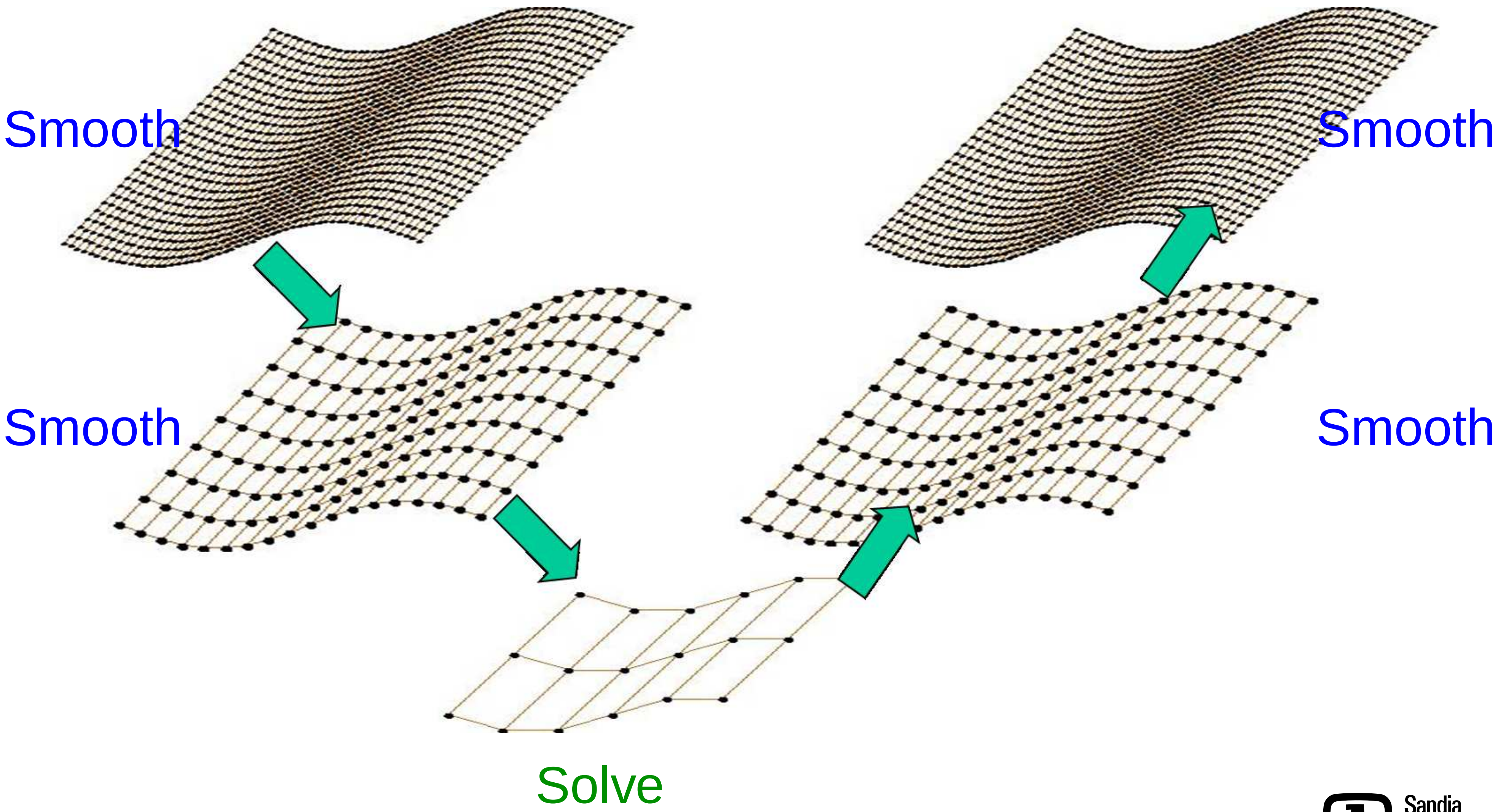
Multigrid & Oscillatory Error

- Question: How to resolve oscillatory modes?
- Answer: Cheap Smoother (Jacobi, Gauss-Seidel, etc.)



Sandia National Laboratories

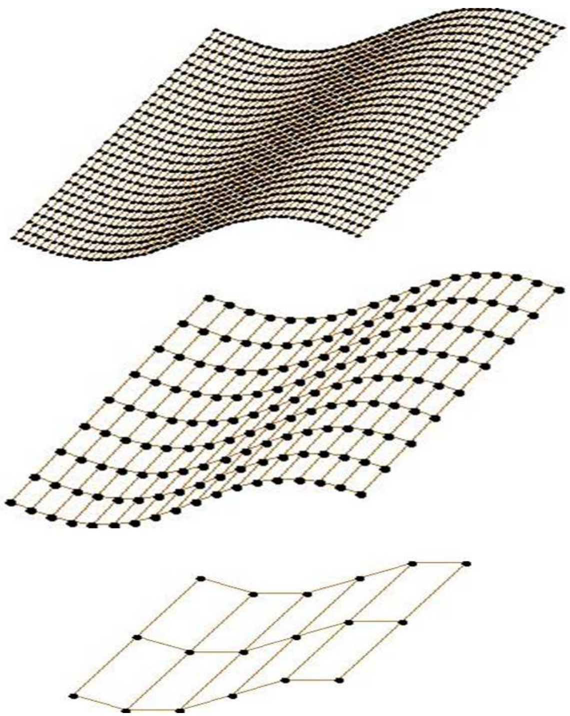
Multigrid by Picture



Repartitioning

- Problem: Coarser grids have less work → worse computation to communication ratio.

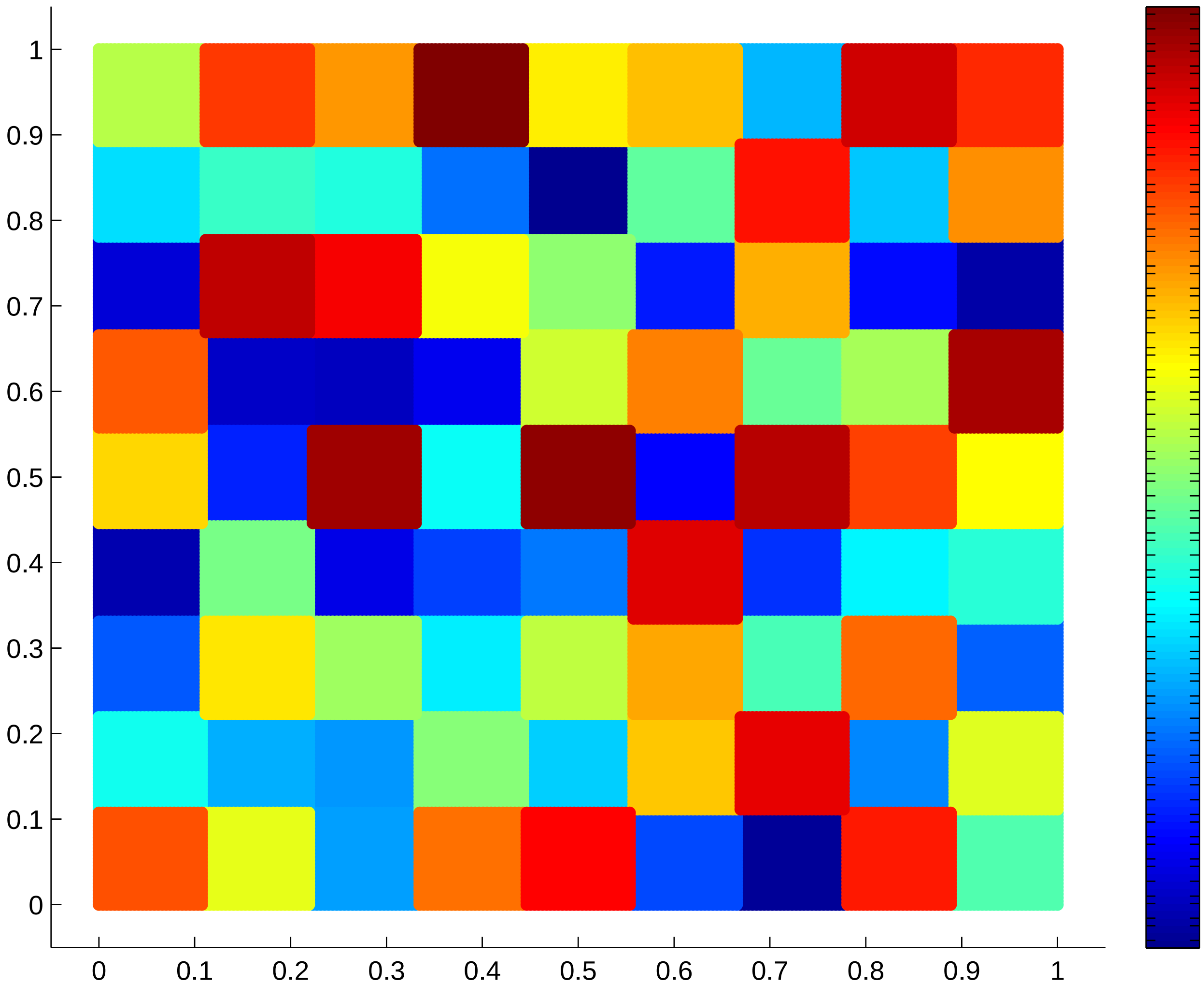
Lots of Work



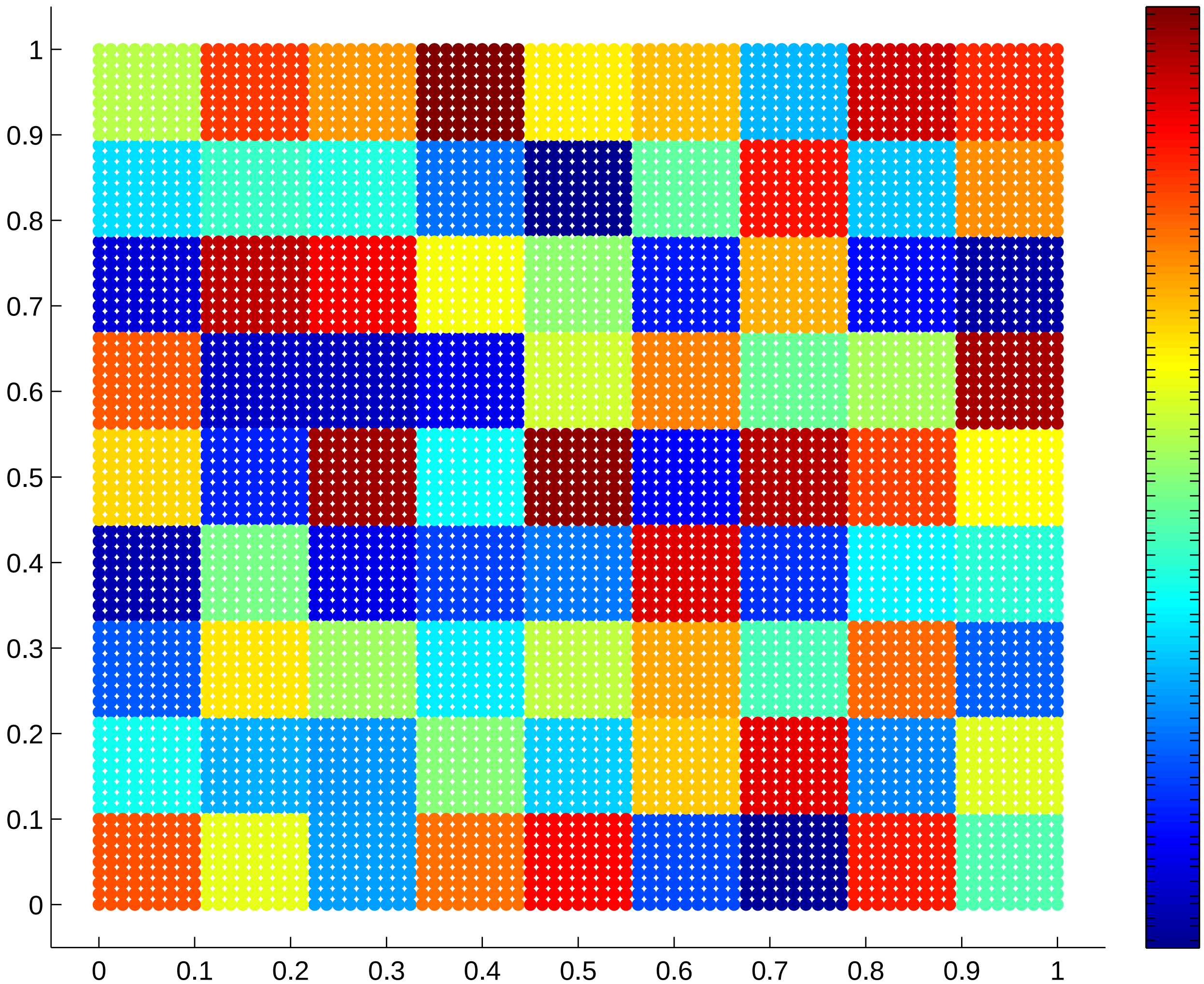
Not Enough Work

- Key idea: Leave processors idle if there's not enough work.

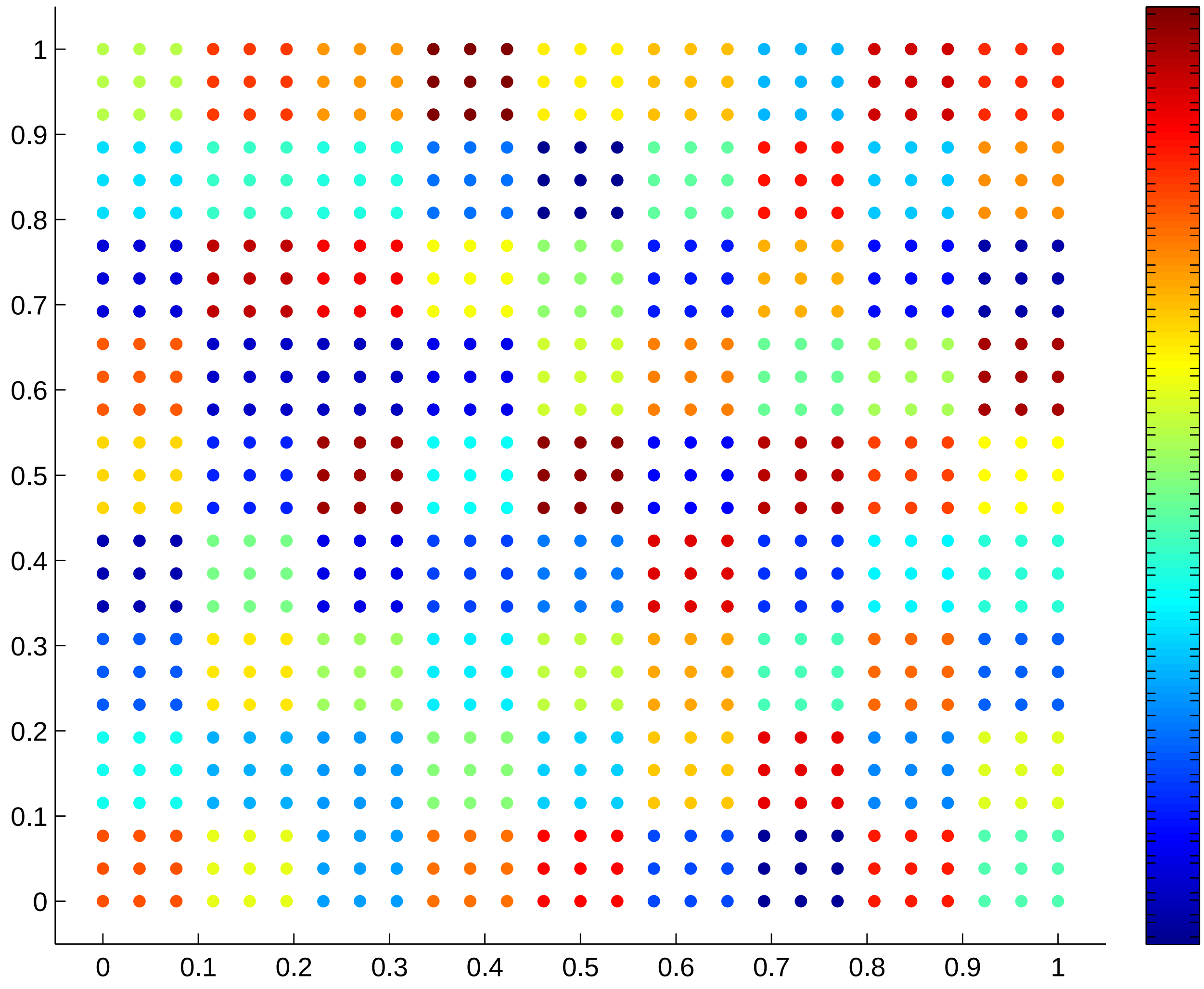
Coarsening Example (243x243)



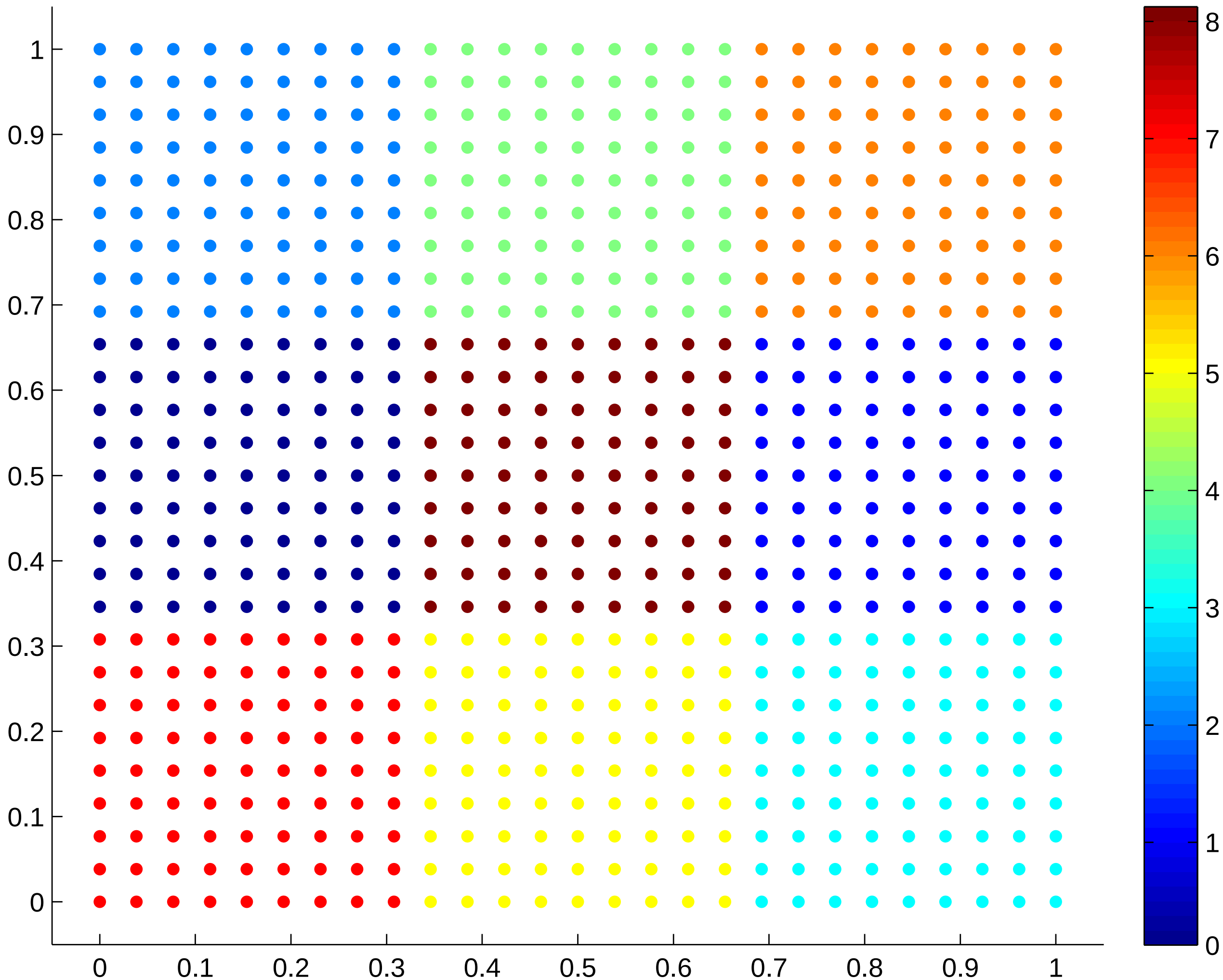
Coarsening Example (81x81)



Coarsening Example (27x27)



Repartitioning (27x27)



Sample Multigrid Hierarchy

Level	Procs	Rows	Rows / Proc
Fine(1)	5,832	158,047,740	27,100
2	5,832	6,097,646	1,045
3	5,832	365,520	62
4	5,832	8,486	1.5
5	5,832	327	.05
Coarse(6)	1	15	15

- $\approx$ 27-to-1 coarsening rate.

Sample Multigrid Hierarchy

Level	Procs	Rows	Rows / Proc
Fine(1)	5,832	158,047,740	27,100
2	5,832	6,097,646	1,045
3	365	365,520	1,001
4	8	8,486	1,060
5	1	327	327
Coarse(6)	1	15	15

- $\approx$ 27-to-1 coarsening rate.
- Substantial decrease in # procs between levels 2-3 and 3-4 (repartitioning).

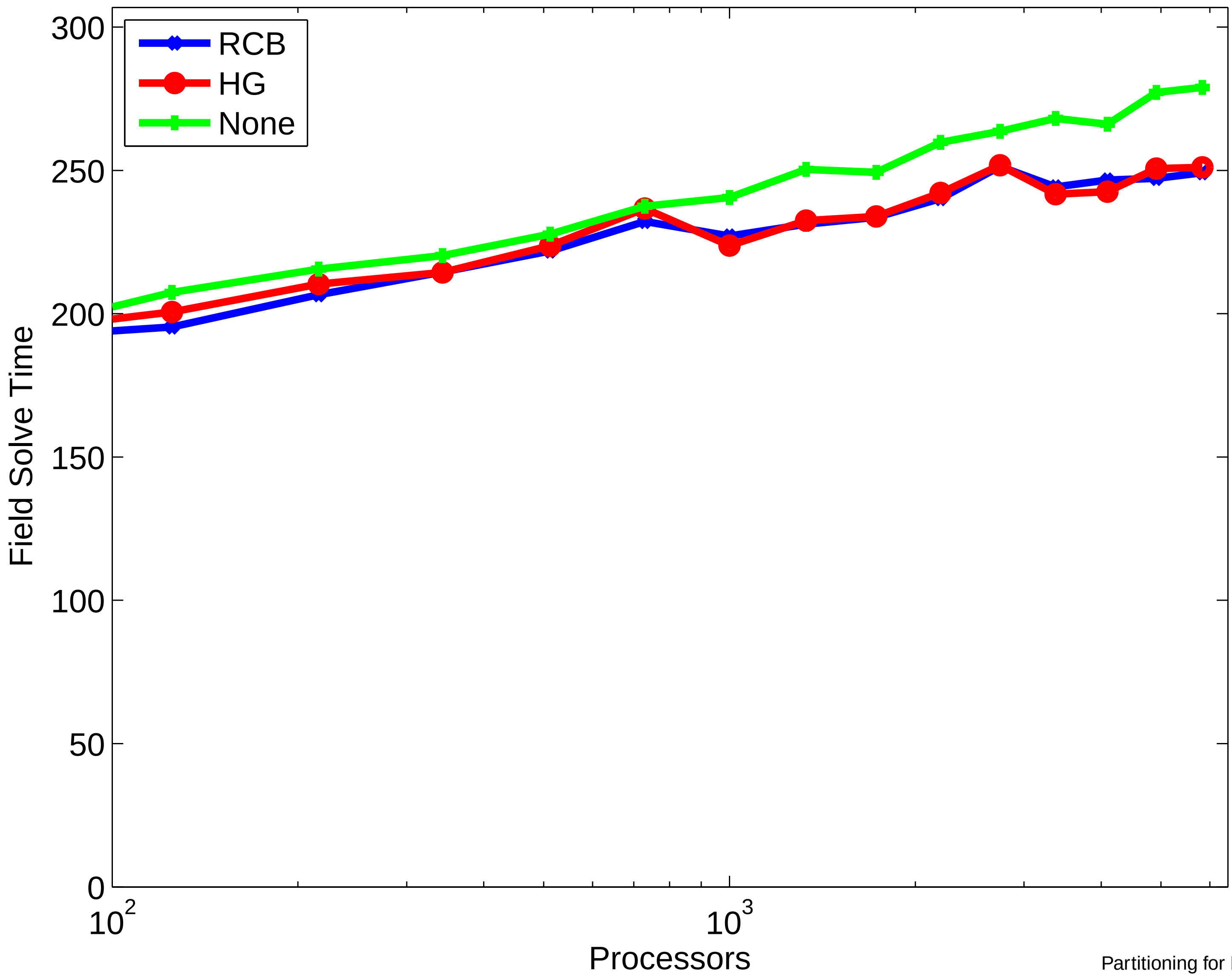
Sandia National Laboratories

Repartitioning w/ Zoltan

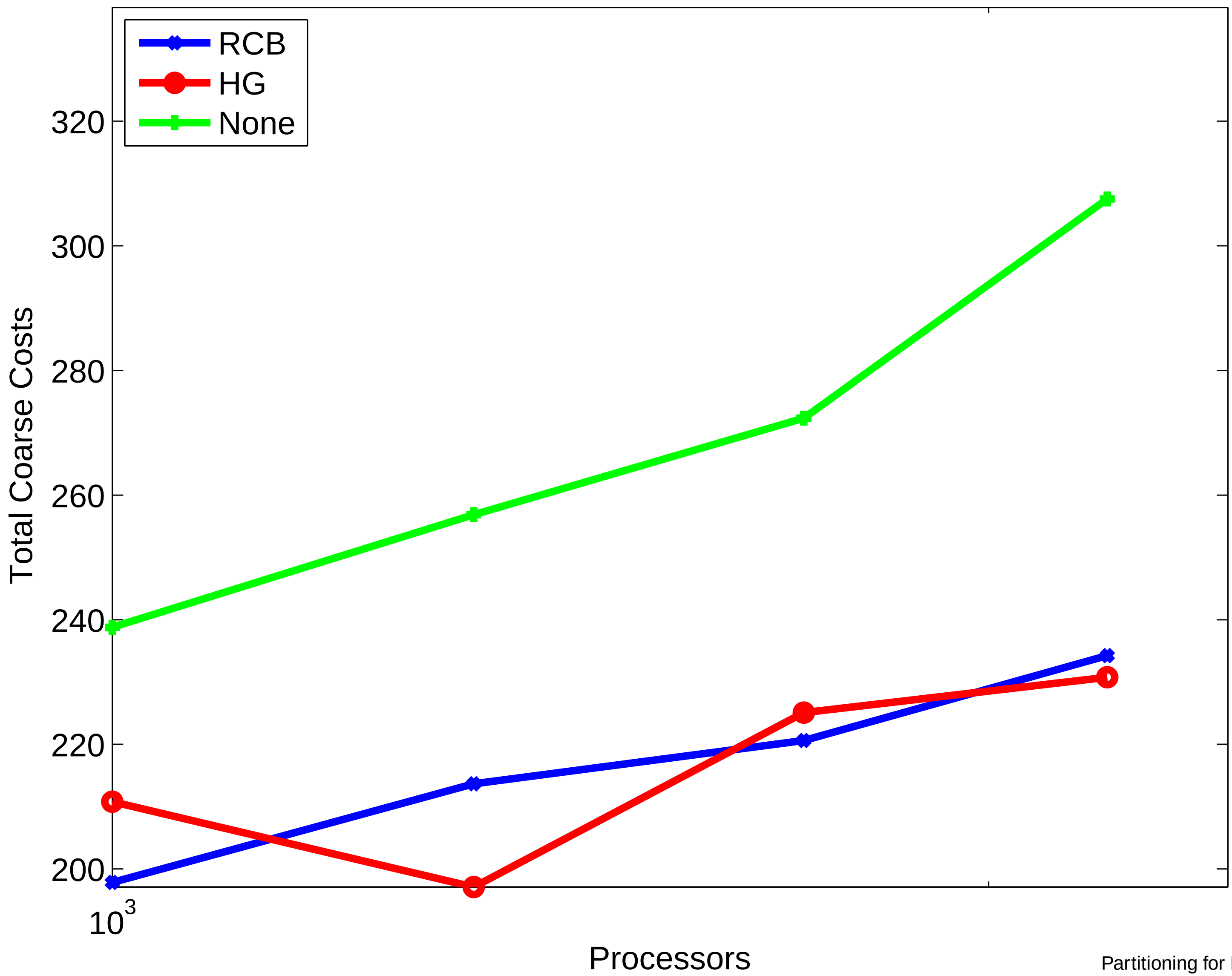
- Idea: Use Zoltan to repartition matrices.
- Algorithm
  1. Detect when problem is too small or unbalanced.
  2. Determine “ideal” # of processors.
  3. Repartition using Zoltan.
  4. Build new transfer operator.
- Obvious: Improves load balancing & computation to communication ratio.
- Less Obvious: Takes stress off aggregation method (not subject of this talk).

Sandia National Laboratories

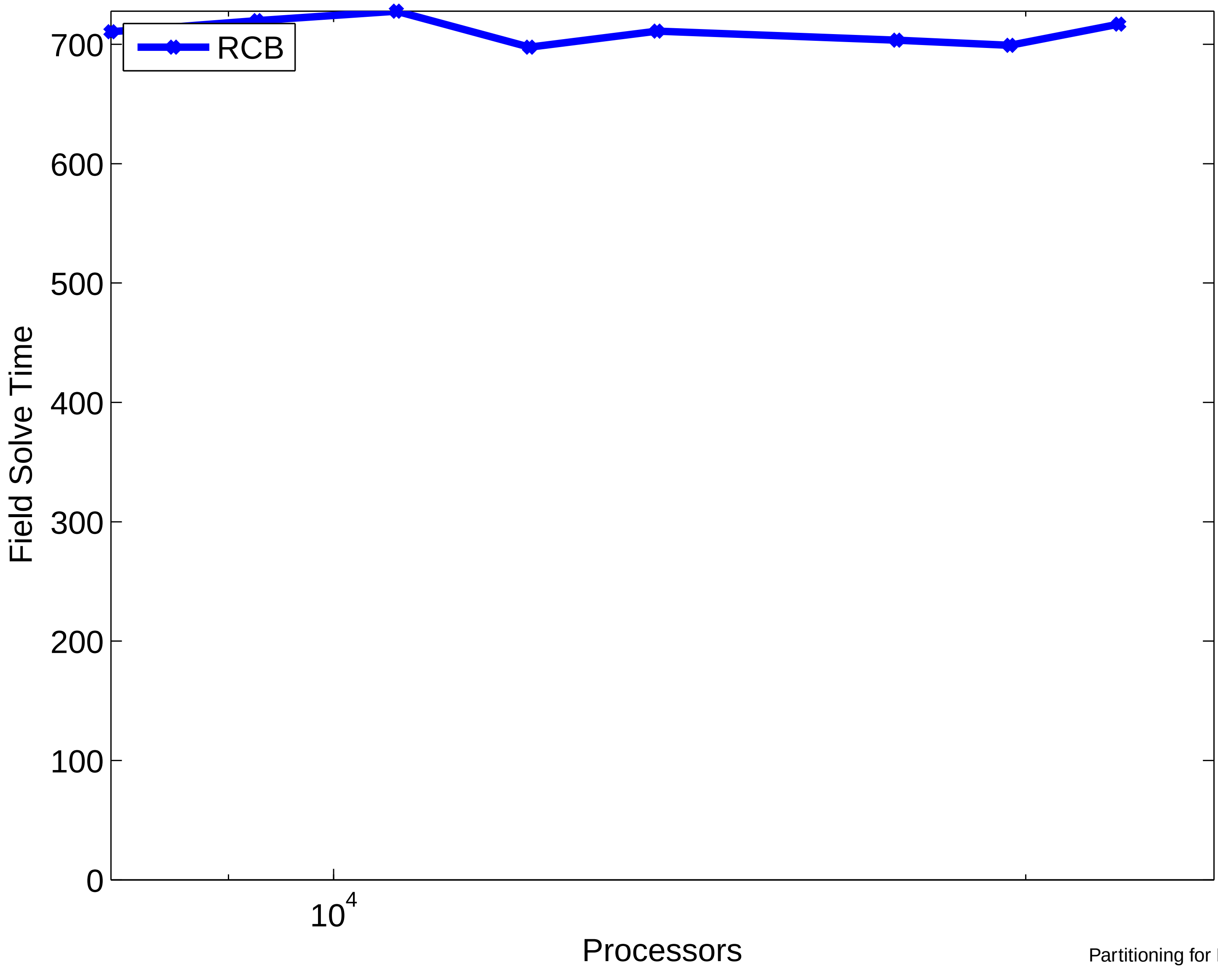
Benefits of Repartitioning (1)



Benefits of Repartitioning (2)



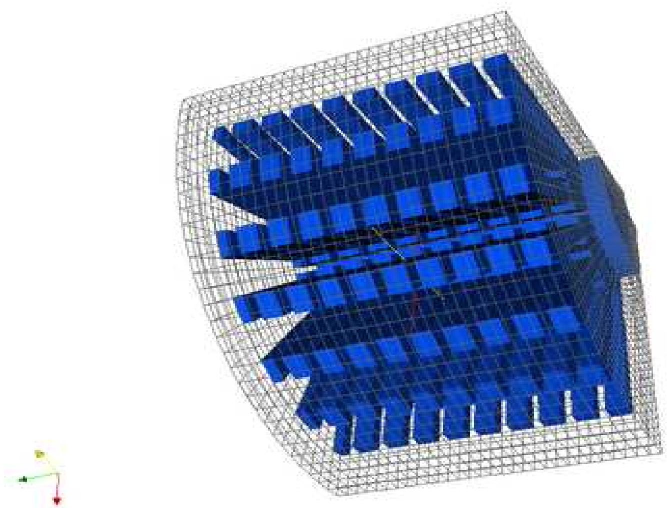
. . . and it works at 21k cores!



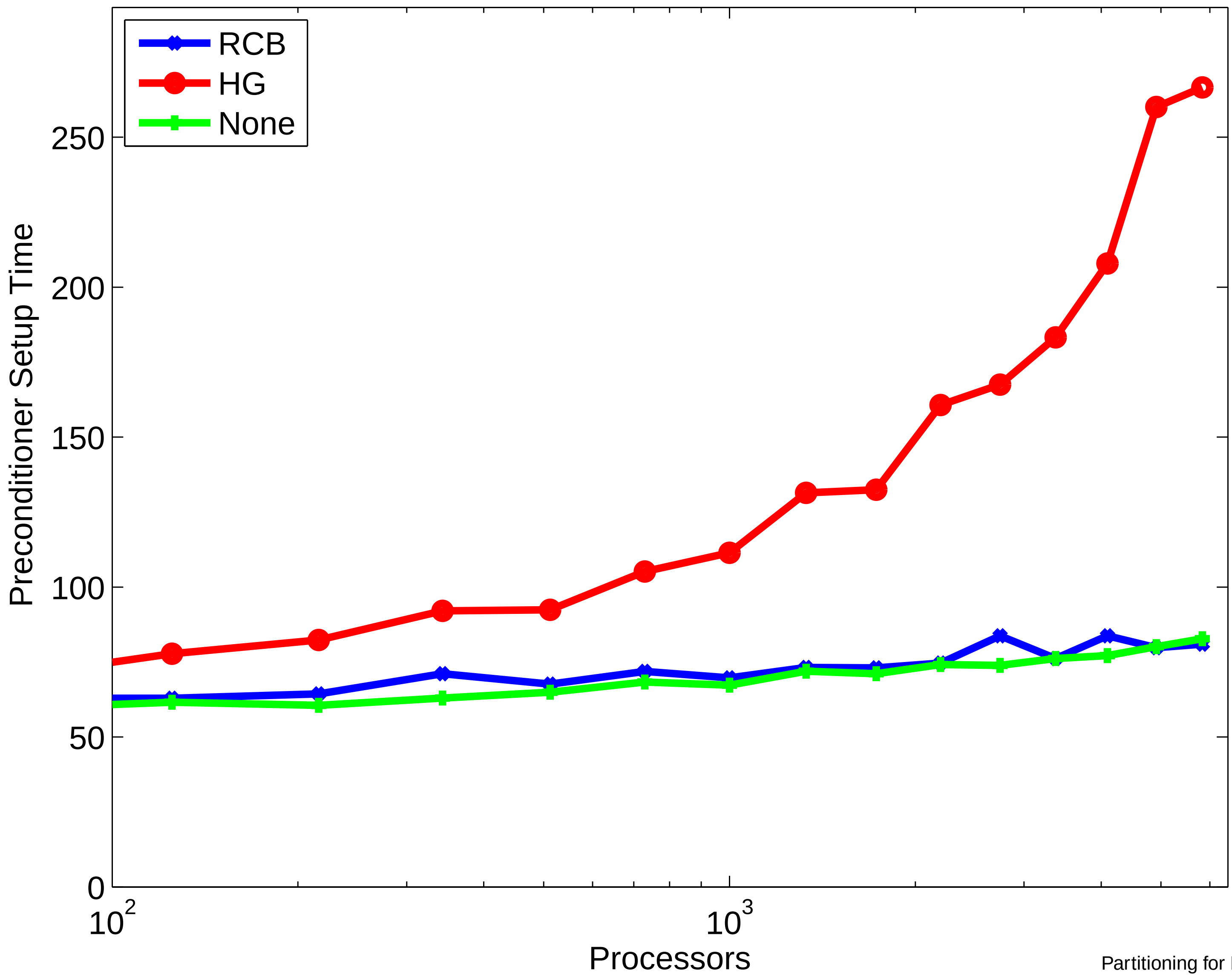
Sandia National Laboratories

Challenges of (Re-)partitioning

- Large change in # processors used
  $\Rightarrow$ "diffusive" methods of limited utility (semi-coarsening?)
- Message limits can be severe (given change in procs) $\Rightarrow$ Communication-efficient algorithms.
- Is RCB good enough for complicated geometries?
- Scalability of (re-)partitioning.



Challenges of (Re-)partitioning



Challenges of (Re-)partitioning

- Large change in # processors used
  $\Rightarrow$ "diffusive" methods of limited utility (semi-coarsening?)
- Message limits can be severe (given change in procs) $\Rightarrow$ Communication-efficient algorithms.
- Is RCB good enough for complicated geometries?
- Scalability of (re-)partitioning.