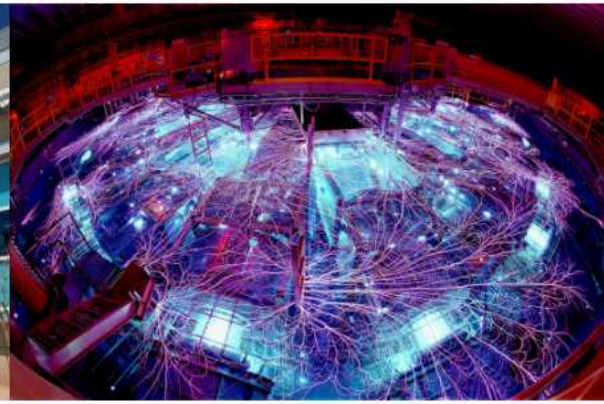
SAND2020-1438C





State of the Tpetra Linear Solver Stack

Chris Siefert, Karen Devine, Mark Hoemmen, Jonathan Hu and Brian Kelley

Special Thanks to: Luc Berger-Vergiat, Kevin Cox and Christian Glusa

2/3/20

Outline

- What is the Tpetra Solver Stack?
  - Matrix Assembly
  - Preconditioner Setup
  - Linear Solve
- Tpetra: Finite Element Assembly Model
- Tpetra/MueLu: Setup Scaling
- Belos/Tpetra/MueLu: Solve Scaling
- Conclusions

What is Tpetra?

Sandia National Laboratories

- Tpetra is Trilinos' sparse linear algebra package
  - Data structures: Matrices, vectors, etc.
  - Data migration / communication: Matrix & vector migration.
  - Computation: dense BLAS-1 and some of sparse BLAS operations.
- Tpetra is built on Kokkos
  - Performance portability for current & future platforms (Goal: DOE CTS/ATS systems current and future).
  - Current node level back-ends: Serial, OpenMP, Cuda.
- Tpetra uses vendor kernels when appropriate (through KokkosKernels).

What is the Tpetra Solver Stack?

Sandia National Laboratories

- Data structures: Tpetra.
- Krylov methods: Belos.
- Preconditioners: MueLu, Ifpack2.
- Direct Solvers: Amesos2.
- Graph Algorithms: Zoltan2.

For the purpose of this talk, we will consider multigrid (MueLu) preconditioned Krylov methods (Belos) using Tpetra.

Matrix Assembly Time Operations

- On-node assembly of matrix & rhs $\rightarrow$ Tpetra/KokkosKernels.
- Assembly of off-node contributions (e.g. finite elements) $\rightarrow$ Tpetra.

All of this is nicely abstracted by Tpetra.

MueLu Preconditioner Setup Time Operations

- Aggregation → MueLu.
- Matrix assembly → MueLu/Tpetra.
- Matrix-matrix multiplication → Tpetra/KokkosKernels.
- Repartitioning → Zoltan2.
- Smoother → Ifpack2.

Solve Time Operations

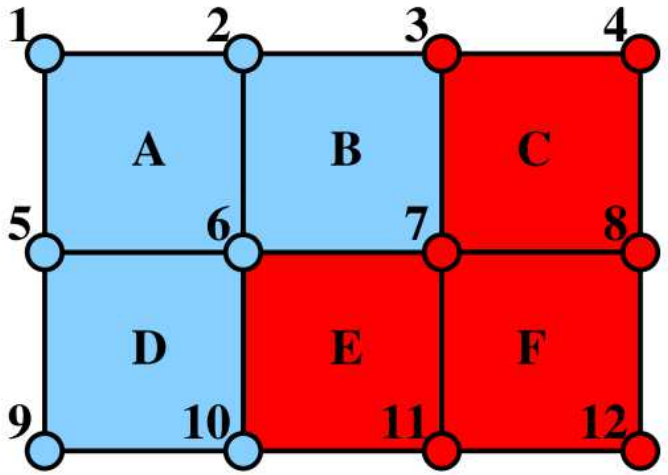
- Matrix-vector multiplication $\rightarrow$ Tpetra/KokkosKernels.
- Data migration $\rightarrow$ Tpetra.
- Smoothers (iterative methods) $\rightarrow$ Ifpack2.
- Coarse grid direct solver $\rightarrow$ Amesos2.

All of these have NGP-ready implementations...except the direct solve.

Outline

- What is the Tpetra Solver Stack?
- Tpetra: Finite Element Assembly Model
  - What it Looks Like
  - CPU Performance
  - GPU Performance
- Tpetra/MueLu: Setup Scaling
- Belos/Tpetra/MueLu: Solve Scaling
- Conclusions

Two Basic Assembly Models



- Type A Assembly: Ghosted Elements
  - + Assembly is communication-free.
  - - Requires duplicate computation.
  - - Requires ghosting of element state.
- Type B Assembly: No Ghosted Elements
  - + No duplicated computation.
  - - Partially assembled matrix/vector must be communicated.

Weak Scaling: Intel Xeon Broadwell

Cores	Ghosted Elements (A) Local	Comm	Non-Ghosted Elements(B) Local	Comm
36	2.7	0.2	2.5	0.4
100	2.7	0.3	2.6	0.4
196	2.7	0.6	2.5	0.6
324	2.7	0.9	2.5	1.0
484	2.7	1.0	2.6	1.3
676	2.7	1.4	2.4	1.6
900	2.7	1.7	2.6	1.9

Note: 6.8k unknowns / core.

Weak Scaling: IBM Power9 + NVIDIA P100

Sandia National Laboratories

GPUs	Ghosted Elements (A) Local	Ghosted Elements (A) Comm	Non-Ghosted Elements (B) Local	Non-Ghosted Elements (B) Comm
1	3.6	0.0	3.6	0.0
4	3.3	0.2	3.3	0.7
9	3.5	0.2	3.5	0.8
16	3.5	0.4	3.3	0.8
25	3.4	0.4	3.4	0.8

Note: 1M unknowns / GPU.

Outline

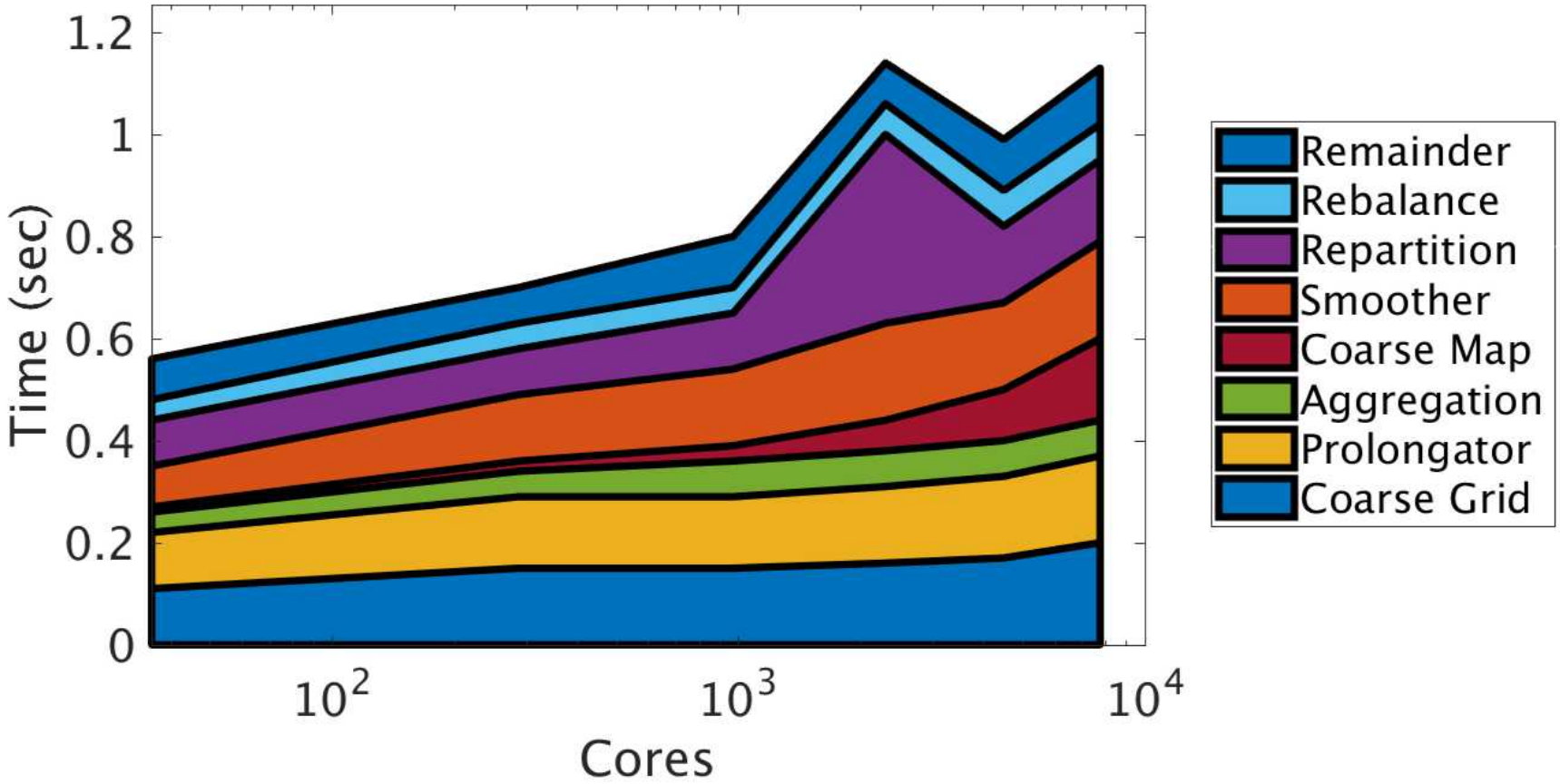
- What is the Tpetra Solver Stack?
- Tpetra: Finite Element Assembly Model
- Tpetra/MueLu: Setup Scaling
  - CPU Performance
  - GPU Performance
- Belos/Tpetra/MueLu: Solve Scaling
- Conclusions

Expectations: MueLu Preconditioner Setup

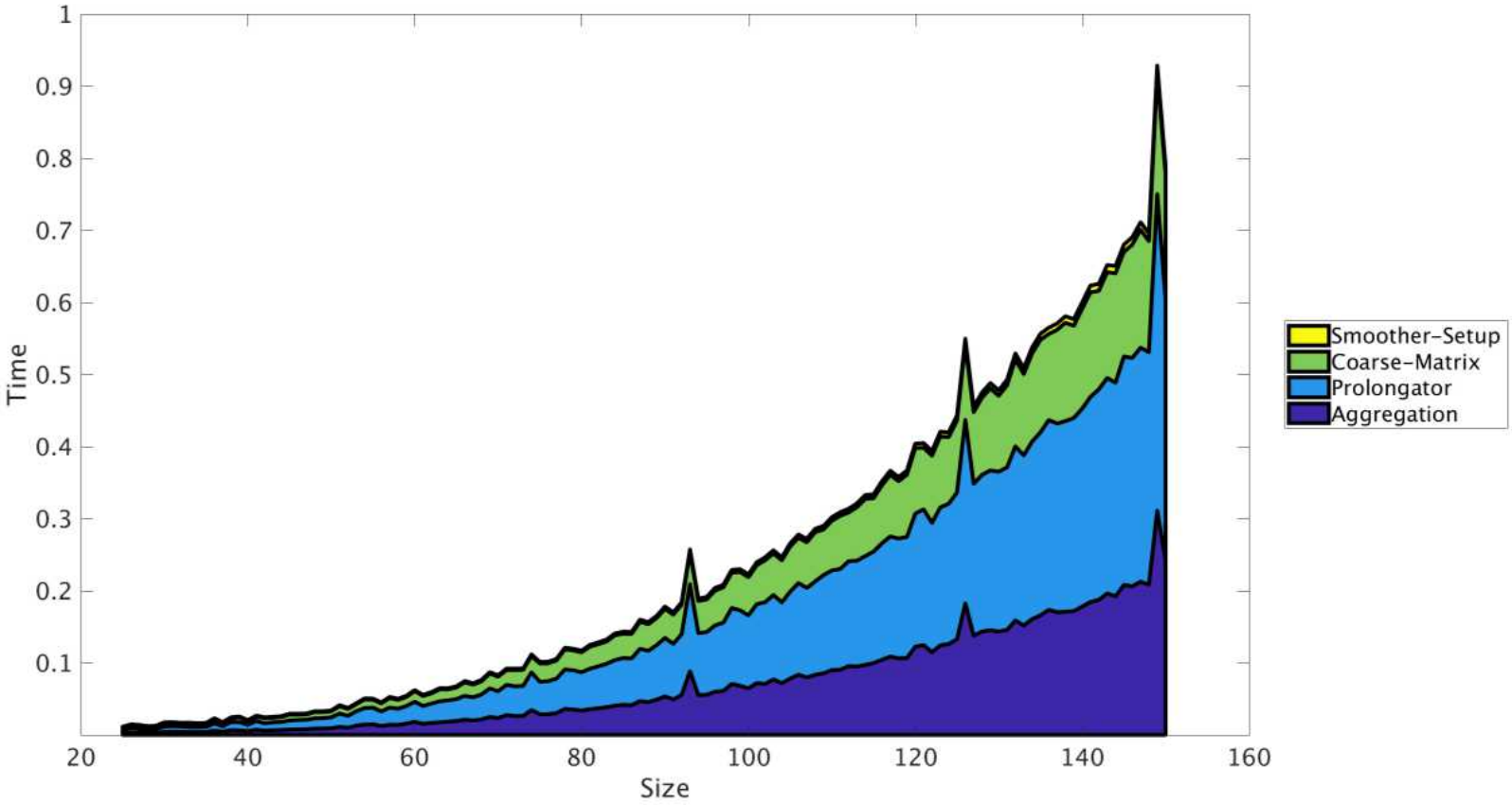
Sandia National Laboratories

- Aggregation: Relatively cheap.
- Matrix assembly: Cheap.
- Matrix-matrix multiplication: Expensive!
- Repartitioning: Cheap, if done far enough down the hierarchy.
- Smoother: Eigenvalue estimates can be expensive.

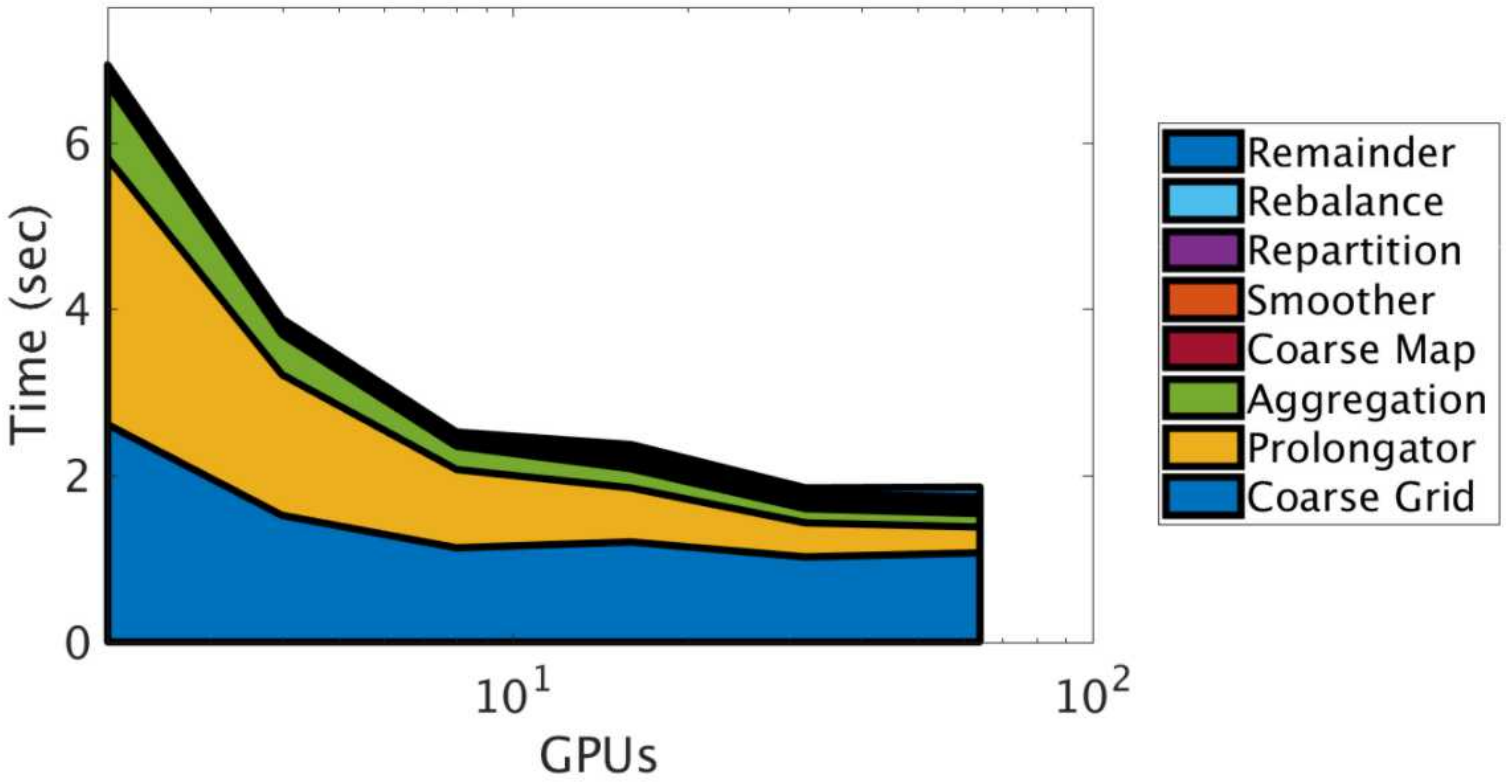
CPU Weak Scaling: Intel Xeon Broadwell



Single GPU: Intel Xeon Haswell + NVIDIA K40



GPU Strong Scaling: IBM Power9 + NVIDIA V100



Note: 15.6M unknowns.

Percent Time in Kokkos: IBM Power 9 + NVIDIA V100

Sandia National Laboratories

- Will look at on-node only, for first level.
- Metric: % time in Kokkos from kp-space-time-stack profiler.

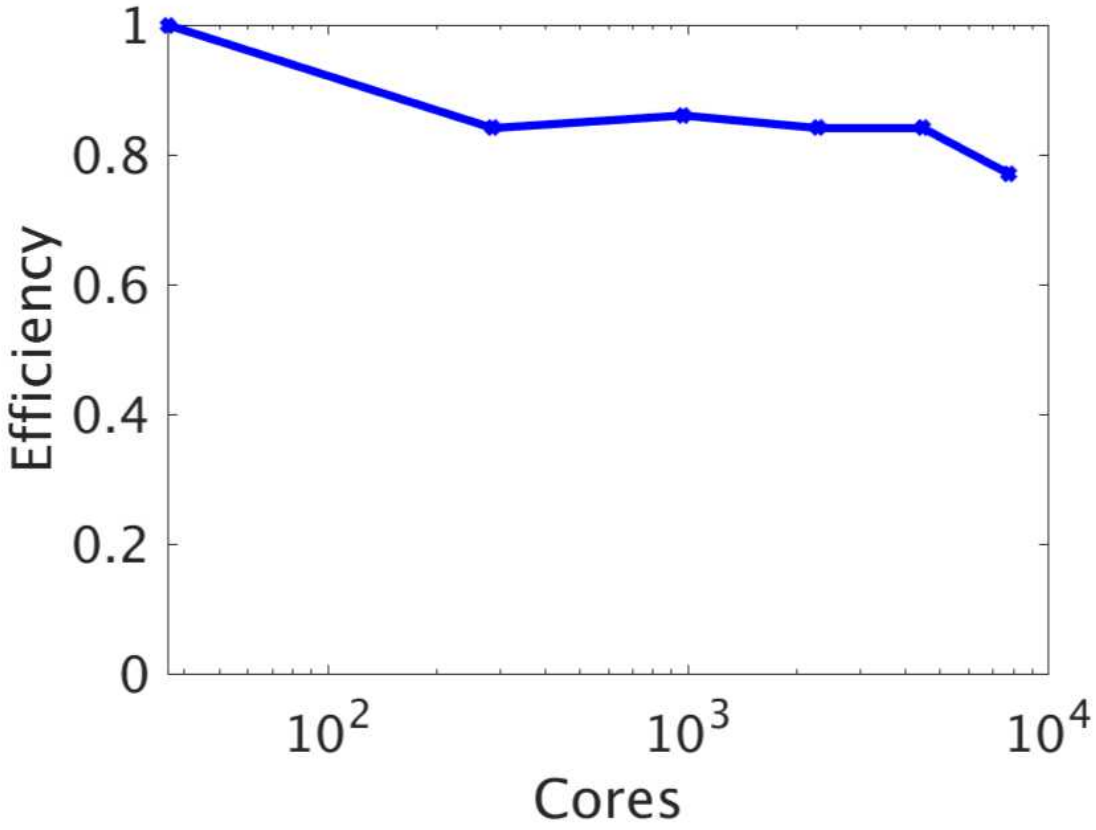
Algorithm	% Time in Kokkos
Total Setup	96.4
Aggregation	98.6
Prolongator (Tentative/Smooth)	98.1 / 96.7
Coarse Grid (AP, R(AP))	98.6 / 94.0
Smoother	68.1
Total Solve	94.7
Matvec	99.5
Precond Apply	94.0

Summary: Both solve and setup are occuring almost entirely on the GPU.

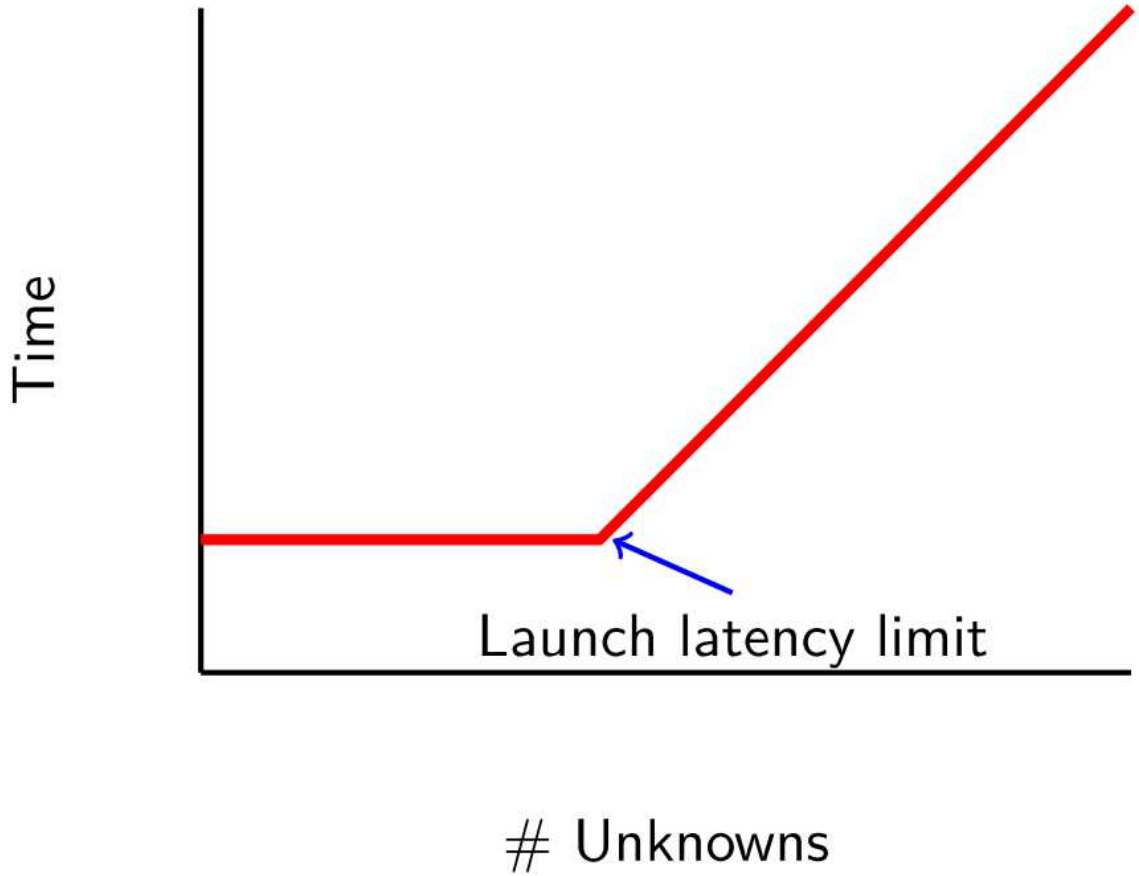
Outline

- What is the Tpetra Solver Stack?
- Tpetra: Finite Element Assembly Model
- Tpetra/MueLu: Setup Scaling
- Belos/Tpetra/MueLu: Solve Scaling
  - CPU Performance
  - GPU Performance
- Conclusions

CPU Weak Scaling: Intel Xeon Broadwell

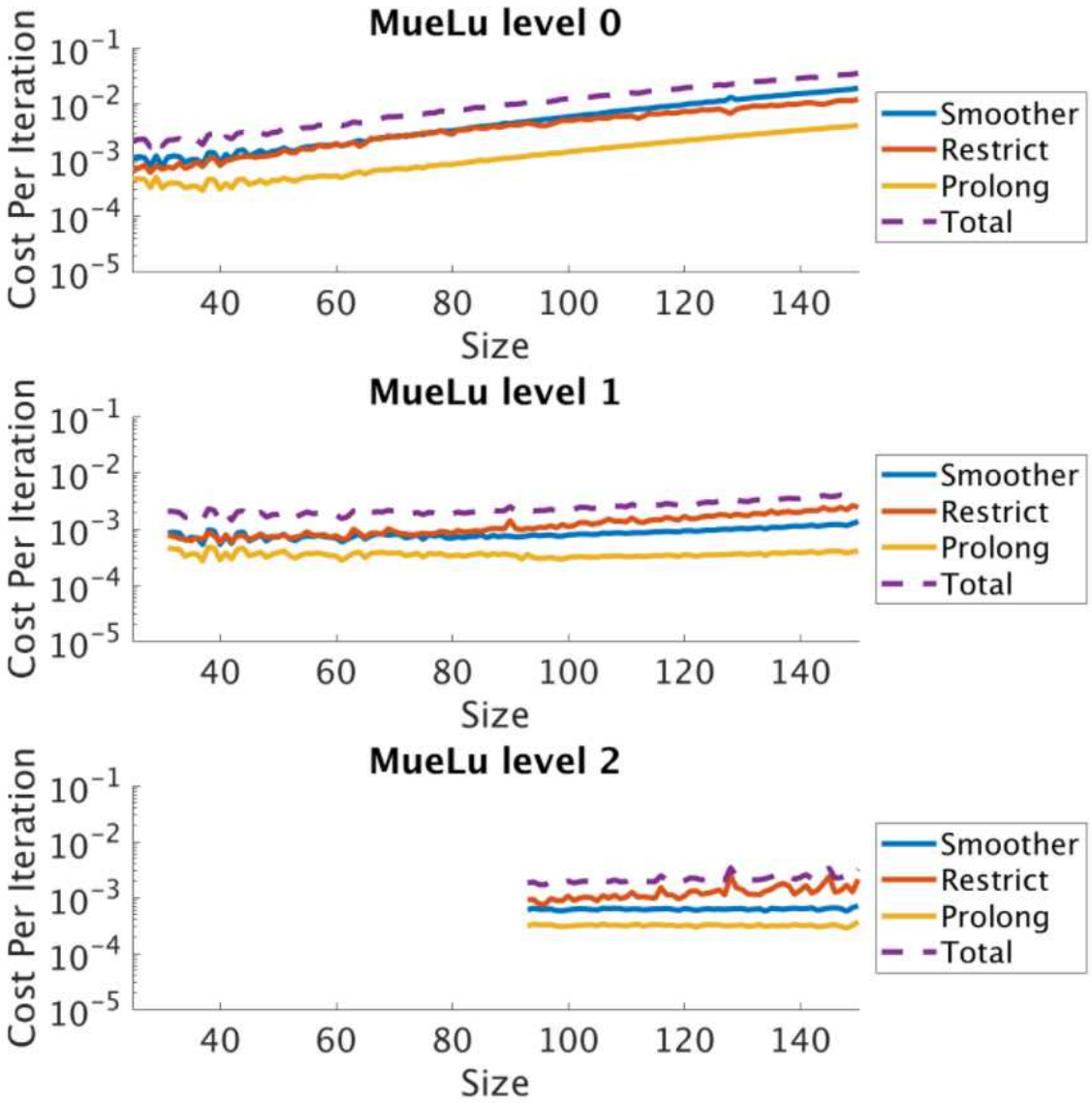


Expected Single GPU Performance

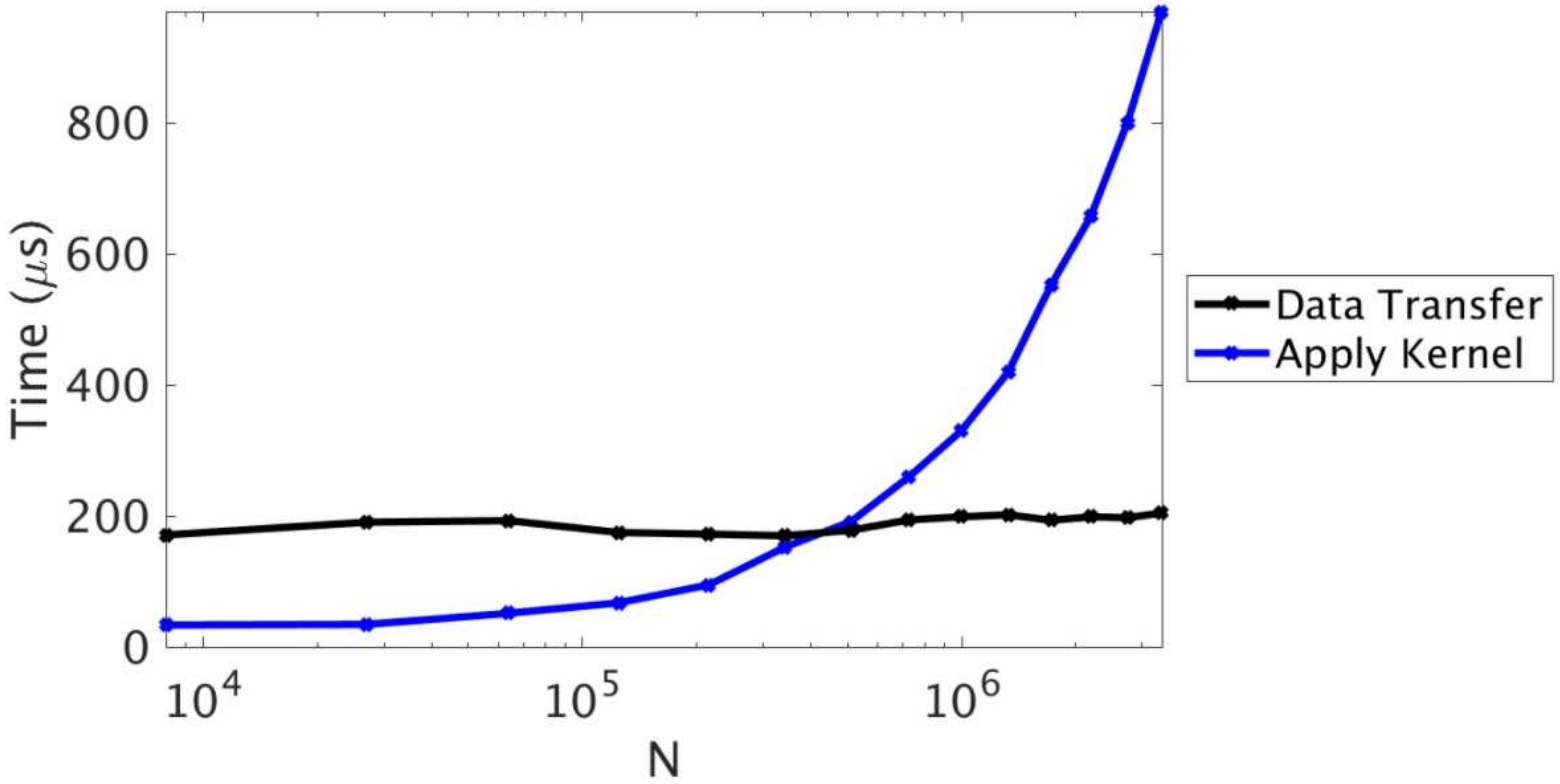


Below a certain amount of work, GPU costs are about constant.

Single GPU: Intel Xeon Haswell + NVIDIA K40



Two GPU Matvec: IBM Power9 + NVIDIA V100



Note: Launch latency (inc profiling) $\sim 31\mu$s, max MPI cost (OSU microbenchmark) 20μs. Crossover @ 220k rows / GPU.

Outline

- What is the Tpetra Solver Stack?
- Tpetra: Finite Element Assembly Model
- Tpetra/MueLu: Setup Scaling
- Belos/Tpetra/MueLu: Solve Scaling
- Conclusions

Conclusions

- Tpetra stack provides scalable solution of linear systems on modern CPU and GPU architectures.
- Preconditioner setup performance is decent.
- Linear solve performance is very good on both CPU and GPU architectures (except for sparse direct solve on GPU).
- Visit us at: trilinos.org or github.com/trilinos/Trilinos.