Exceptional service in the national interest

Sandia National Laboratories

SAND2019-14629PE

December, 2019

Trilinos linear solvers group update

Luc Berger-Vergiat, Jonathan Hu,
Ichitaro Yamazaki, Siva Rajamanickam
Center for Computing Research

Sandia National Laboratories
Albuquerque, New Mexico USA
SAND 2019-XXXXX

Reminder

Here is the title of the milestone:

Compare linear-system solver and preconditioner stacks with emphasis on GPU performance and propose phase-2 NGP solver-development pathway

The spirit of the milestone is to compare Hypre and Trilinos for linear solvers on CPU/GPU, with emphasis on strong scaling, on a whole turbine model.

MueLu's setup refactor

Sandia National Laboratories

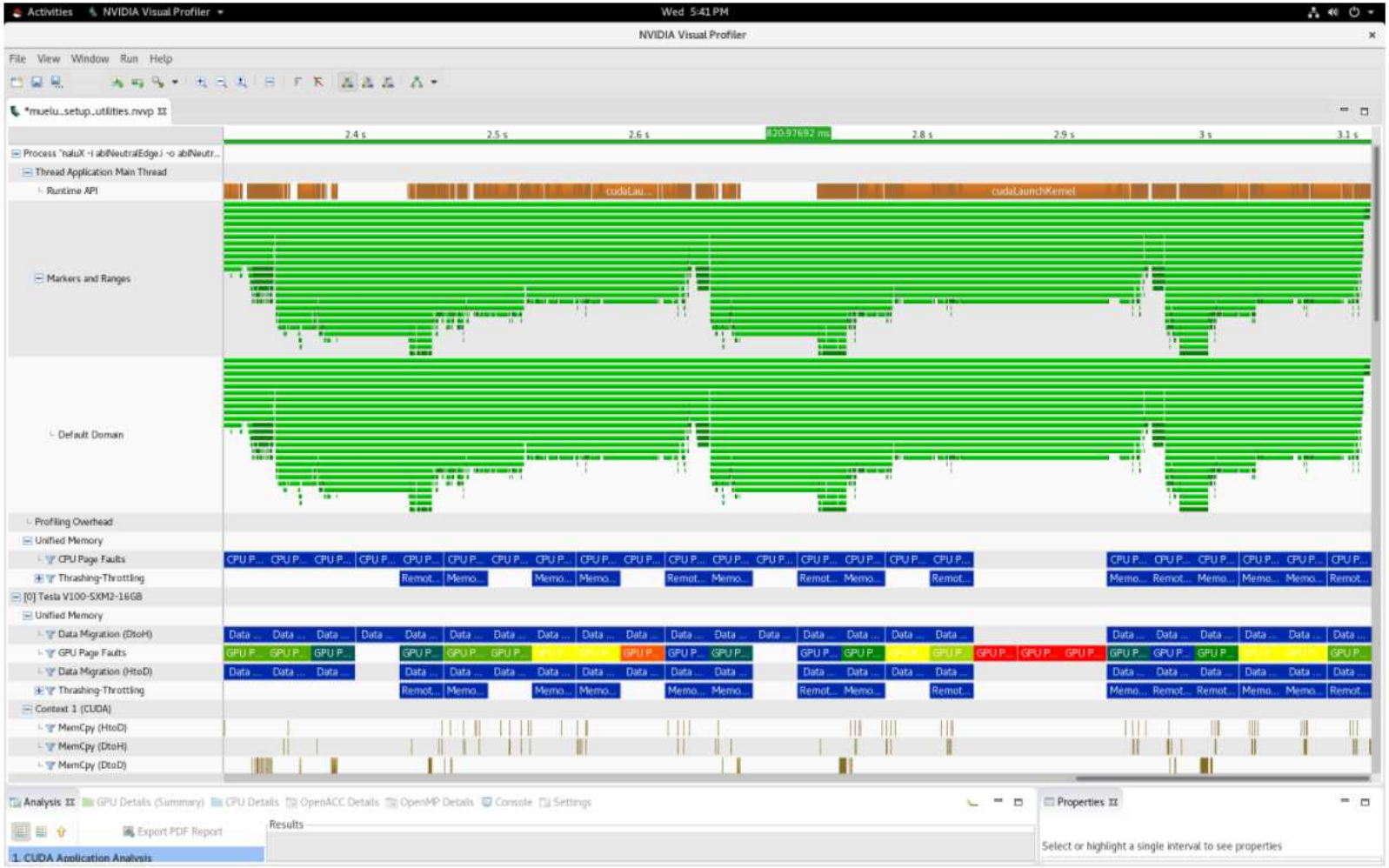
Multigrid's setup main components:

1. Smoother setup: Jacobi, Chebyshev and Gauss-Seidel all on device
2. Aggregation: recent Kokkos refactor, 90% on device, still need small clean-up
3. Prolongator: refactored, performance tuning
4. RAP: Matrix-Matrix product for leverages Kokkos-Kernels' SpGEMM
5. Rebalancing: currently on host
6. Coarsest level direct solve: LU factorization on host (this happens at first iteration, not setup)

Setup profiling

Trilinos version: commit 8314882 on Tue Nov 26 2019

Nalu-Wind version: commit b6e285b on Tue Nov 26 2019



ablNeutralEdge on Summit

Sandia National Laboratories

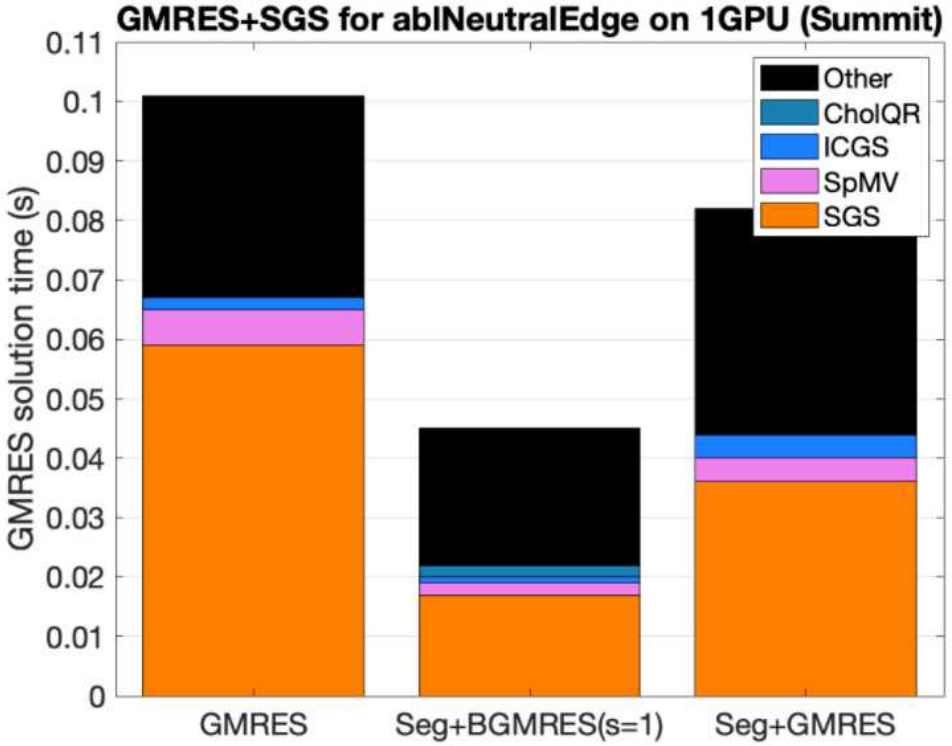
Goal, as stated by Shreyas:

Demonstrate GPU capabilities in Nalu-Wind

From linear solvers perspective:

- collaborative effort between Shreyas, Alan and Luc
- multiple bug discovered and fixed
- test runs correctly on multiple nodes of Summit with multiple GPUs

Block GMRES for segregated momentum solve



- Stand-alone tester (not within Nalu, yet) on Summit
- Kokkos-kernels supports (ECP CLOVER/Kokkos Kernels project) for block ops, i.e. with multiple RHSs:
  - tall-skinny GEMM (Seher Acer) for orthogonalization
  - multi-thread SGS (Brian Kelley)
  - SpMM

Single CrsGraph setup

Sandia National Laboratories

Goal: Create single matrix graph that will be shared between different linear system setups
What's done now: For each linear system, new matrix graph is created
Matrix graph:

- defines matrix sparsity pattern
- creates row, column, domain, range maps
- constructs Import and Export objects for interprocess comm.

Status of refactor

- TpetraLinearSystem and TpetraSegregatedLinearSystem call new CrsGraph class
- All tests pass under RHEL7 gcc 7.3
- Next step: move CrsGraph from LinearSystem into Realm

Plan for Q2

Sandia National Laboratories

- Luc
  1. work on refactoring TpetraSegregateLinearSystem for GPU (with Jonathan?)
  2. further profiling of ablNeutralEdge
  3. start code optimization in MueLu/Kokkos-Kernels for strong scaling on GPU
- Ichi
  1. scaling test of block GMRES within Nalu-wind
- Jonathan
  1. Profile and optimize linear solver setup/apply on GPUs.
- Siva
  1. Integrate and profile cluster GS
  2. Profile d2 graph coloring for aggregation (Luc)

Some requests

Sandia National Laboratories

- Define "Whole turbine model" that needs to run for Q2 milestone
- Add nightly test from Sandia machines
- Need better GPU regression coverage