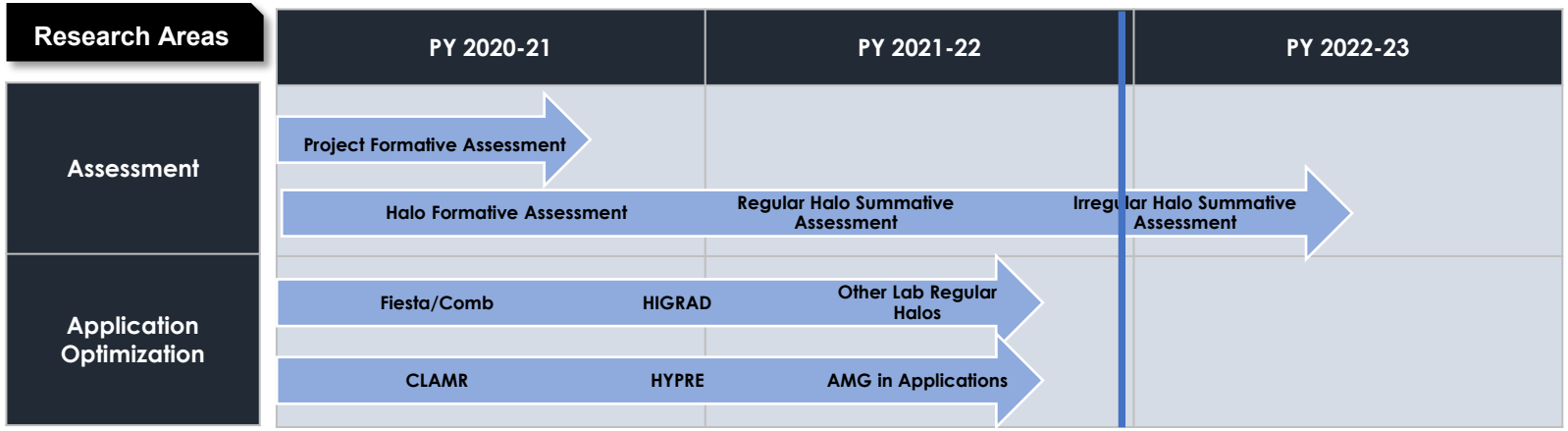
Assessment, Proxy Development, and Education

PSAAP-III 2022 Annual Review

Prof. Patrick Bridges

Assessment/Optimization Overview



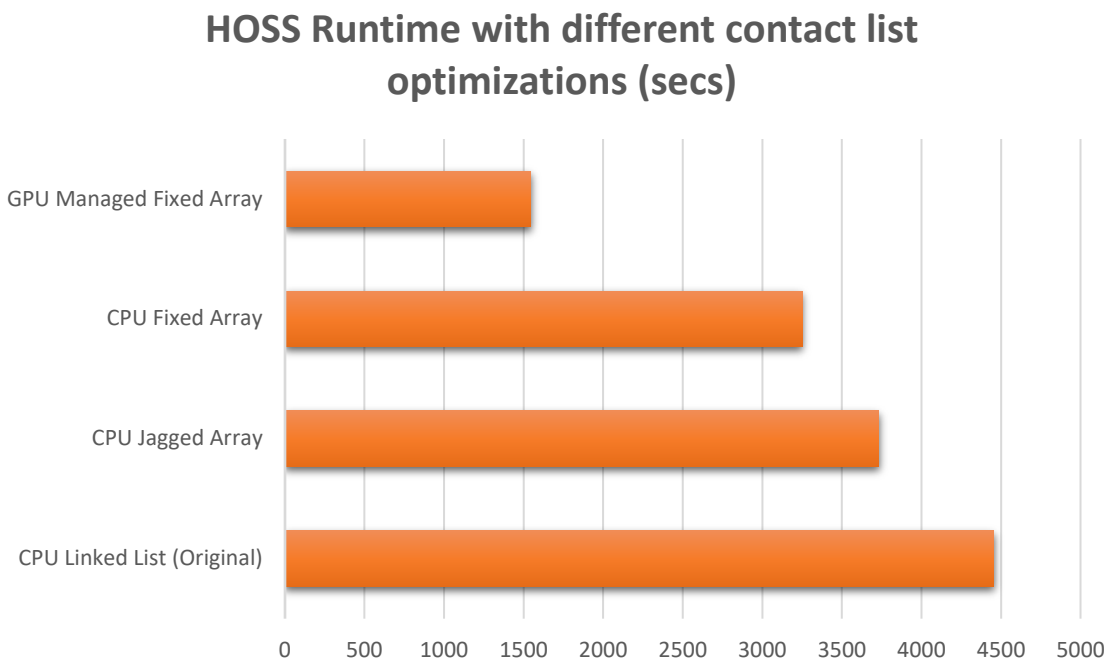
- Deep dive into GPU communication performance and abstraction development pushed back summative assessment and optimization
- Regular halo optimization and summative assessment progressing with Comb and HIGRAD
- AMG optimization done in HYPRE, planned for Trilinos.
- Application assessment of AMG optimizations planned
- Irregular assessment underway in HOSS, xRAGE
- Staff gaining expertise in HOSS and xRAGE to prepare for optimization

Irregular Communication Assessment

- Developed general strategy for characterizing irregular communication behaviors
  - Estimate distribution of # neighbors, # blocks, block size, block stride in an application
  - Distributions capture per-node variance in communication size and dynamic changes (e.g. AMR).
  - Starting with gaussian distribution assumption, will need to capture more complex distributions
  - Created benchmark to reproduce irregular communication patterns based on this information Characterization of CLAMR, HOSS, and xRAGE behavior in progress
- Considering other codes (e.g. MiniAero, LULESH, SPARC, SAMRAI);
- Requires instrumenting codes to examine data structure layouts
- Hard to automate, so must balance instrumentation effort vs. gained information
- More detail in Carson Woods afternoon presentation

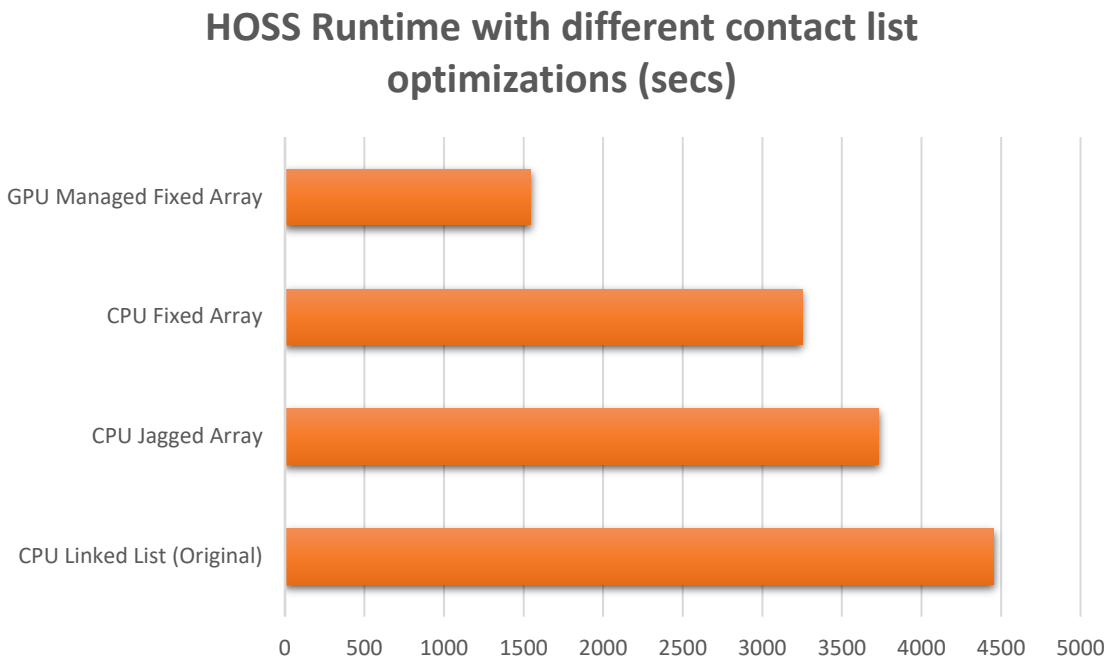
Preparing for HOSS Optimization

- Postdoc Ryan Marshall ported HOSS to OpenACC to gaining expertise with HOSS prior to communication optimization
- Converted HOSS contact list to use ragged or sparse arrays instead of linked lists
- Conversion to arrays also will help MPI performance



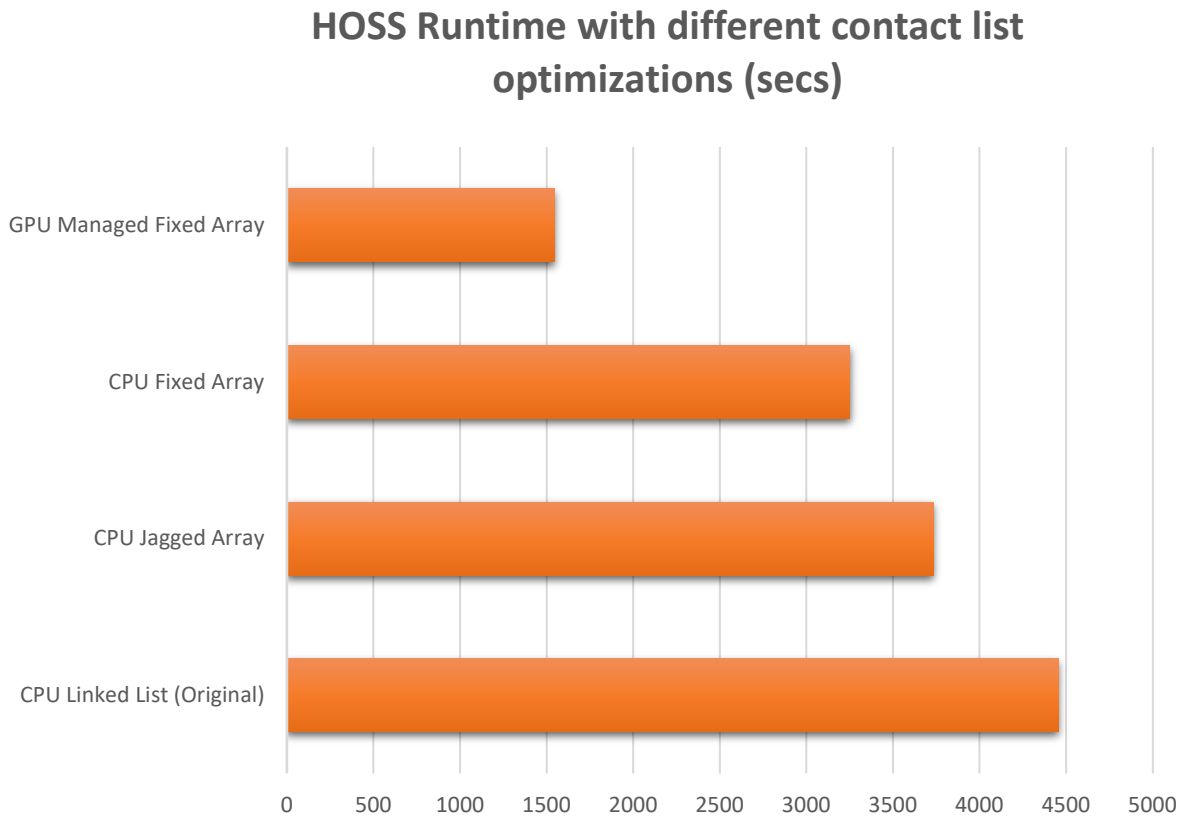
Preparing for HOSS Optimization

- Postdoc Ryan Marshall ported HOSS to OpenACC to gaining expertise with HOSS prior to communication optimization
- Converted HOSS contact list to use jagged and/or sparse arrays instead of linked lists
  - Necessary for GPU port
  - Also helps with MPI port
- 20% CPU, 3x GPU speedup



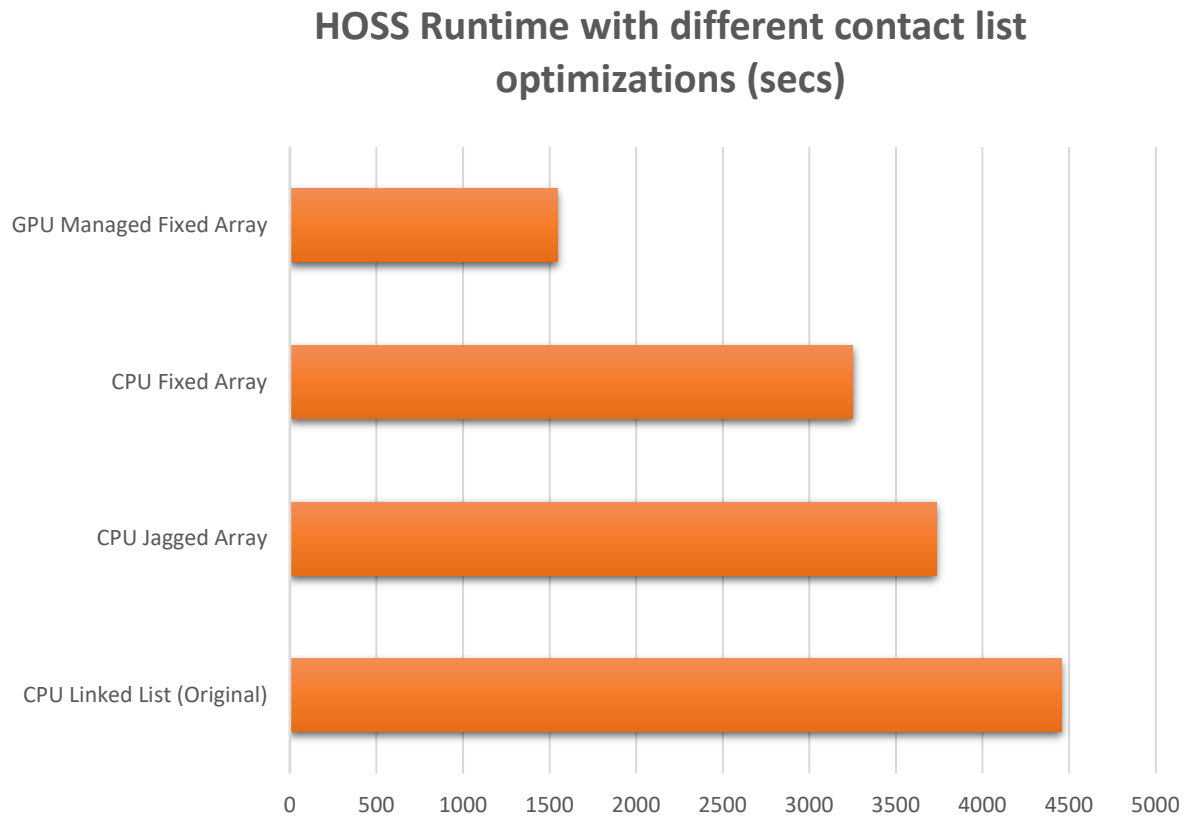
HOSS Optimization Results

- Test setup
  - Runs on LANL Darwin testbed
  - CPU: Intel Haswell E5-2697
  - GPU: NVIDIA RTX A5000 or Tesla K80 GPUs
  - “fracture model” tests (2D) with 10000 iterations
- 28% CPU performance improvement
- 3x CPU->GPU speedup



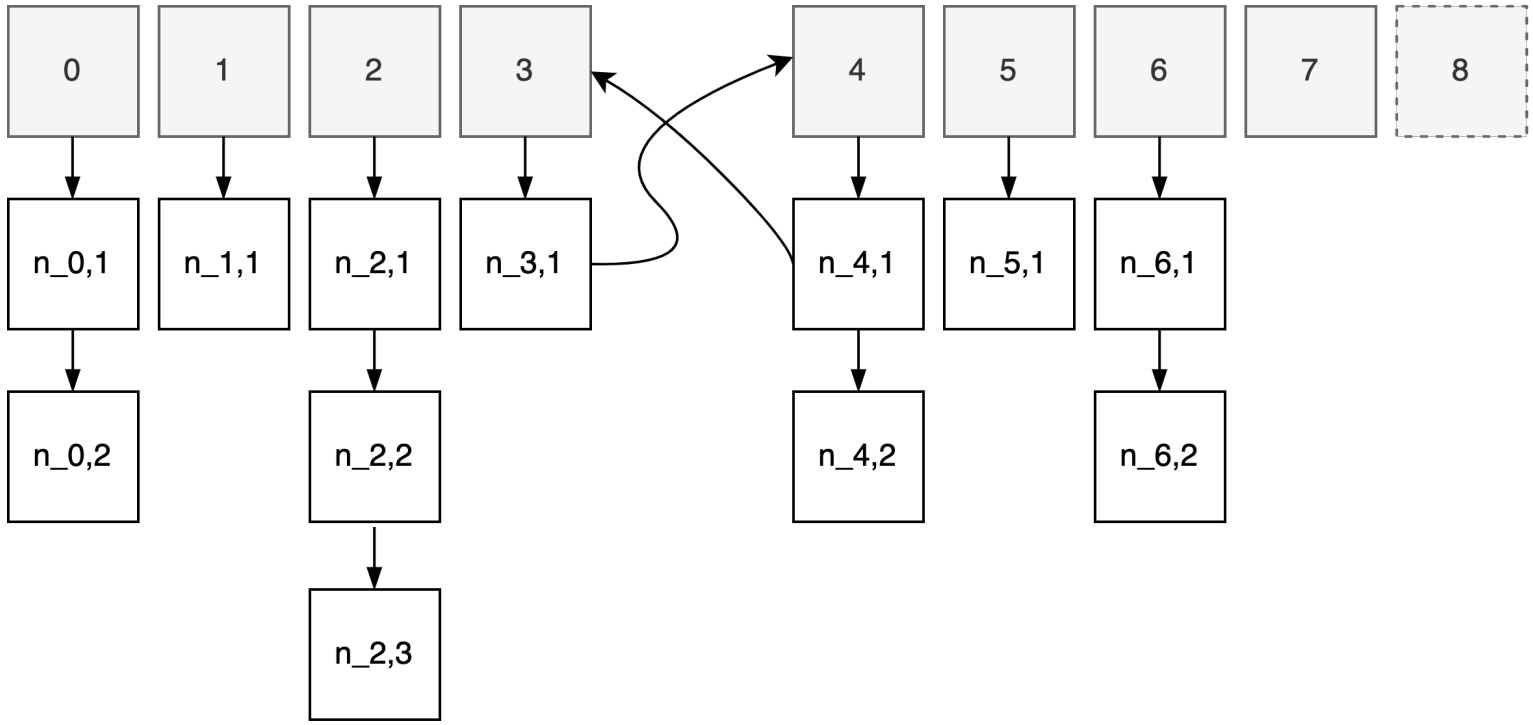
HOSS Optimization Results

- Test setup
  - Runs on LANL Darwin testbed
  - CPU: Intel Haswell E5-2697
  - GPU: NVIDIA RTX A5000 or Tesla K80 GPUs
  - “fracture model” tests (2D) with 10000 iterations
- 28% CPU performance improvement
- 3x CPU->GPU speedup
- Runs on LANL Cray systems planned



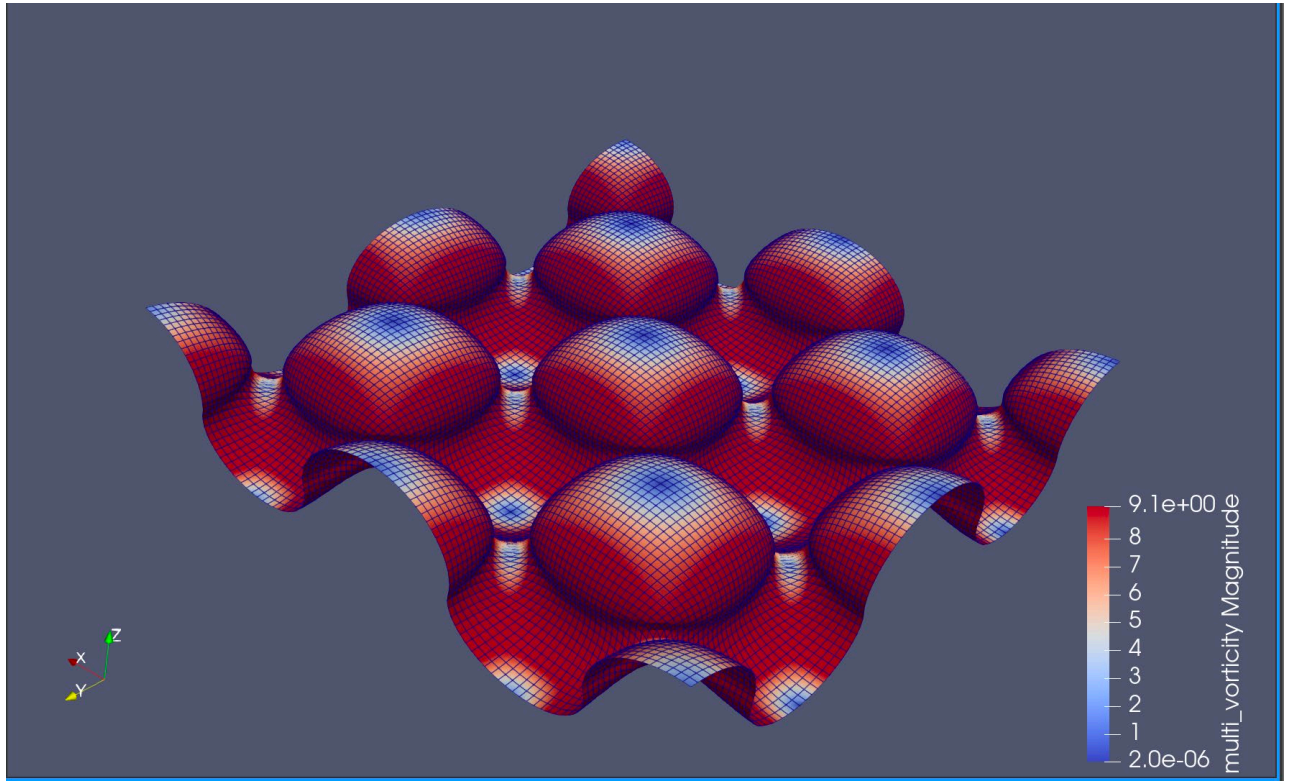
HOSS Irregular Communication

- MPI communication occurs when a cell ID in a cell's neighbor list refers to a cell in another process
- Pretty standard irregular pack/send/recv/unpack
- Communication overhead for 2 processes running fracture model is 25%
- Newer GPUs will increase this overhead substantially



Proxy Development

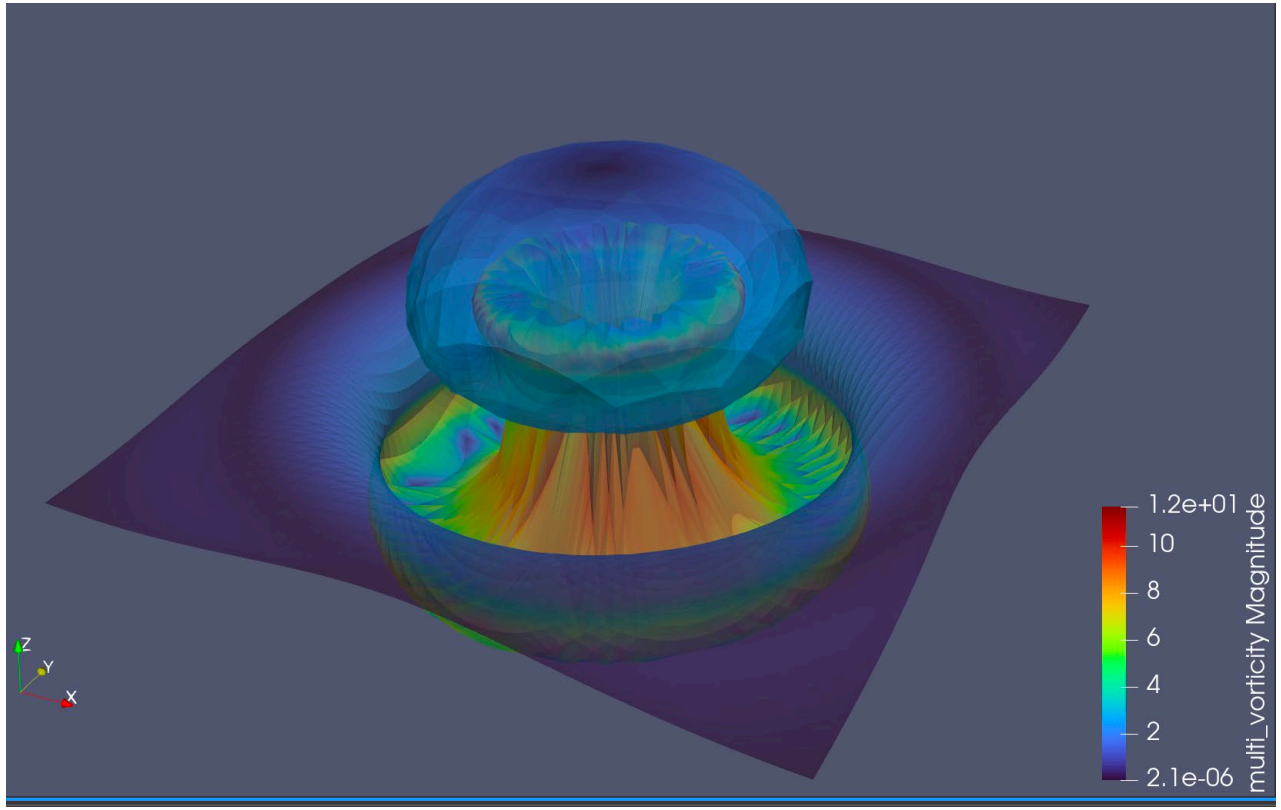
- Beatnik - full MPI/Kokkos fluid interface model proxy
  - Currently focuses on the lower order implementation
  - Good benchmark for assessing FFT optimizations
  - FFT interface optimizations folded into mainline ECP-Copa Cabana git
  - Using to assess impact of locality-aware alltoallv from MPI Advance
- Now available from github: https://github.com/CUP-ECS/beatnik



Low-order simulation of rocket rig Raleigh-Taylor instability

Proxy Development

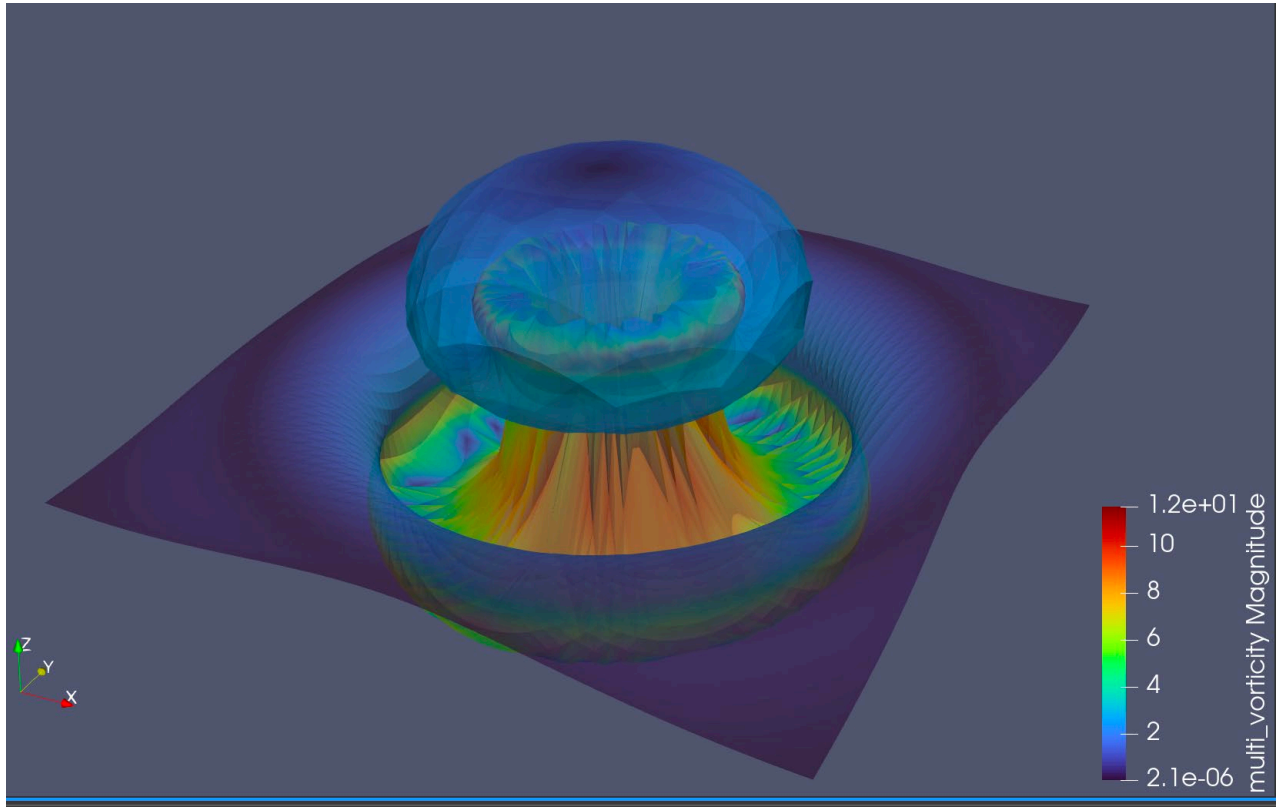
- Beatnik - full MPI/Kokkos fluid interface model proxy
  - **High-order model currently brute forces far-field calculation**
  - **Next step is to bin points point and implement cut-off based approach**
  - **Good benchmark for assessing global sort performance**
- Now available from github: https://github.com/CUP-ECS/beatnik



High-order simulation of rocket rig Raleigh-Taylor instability

Proxy Development

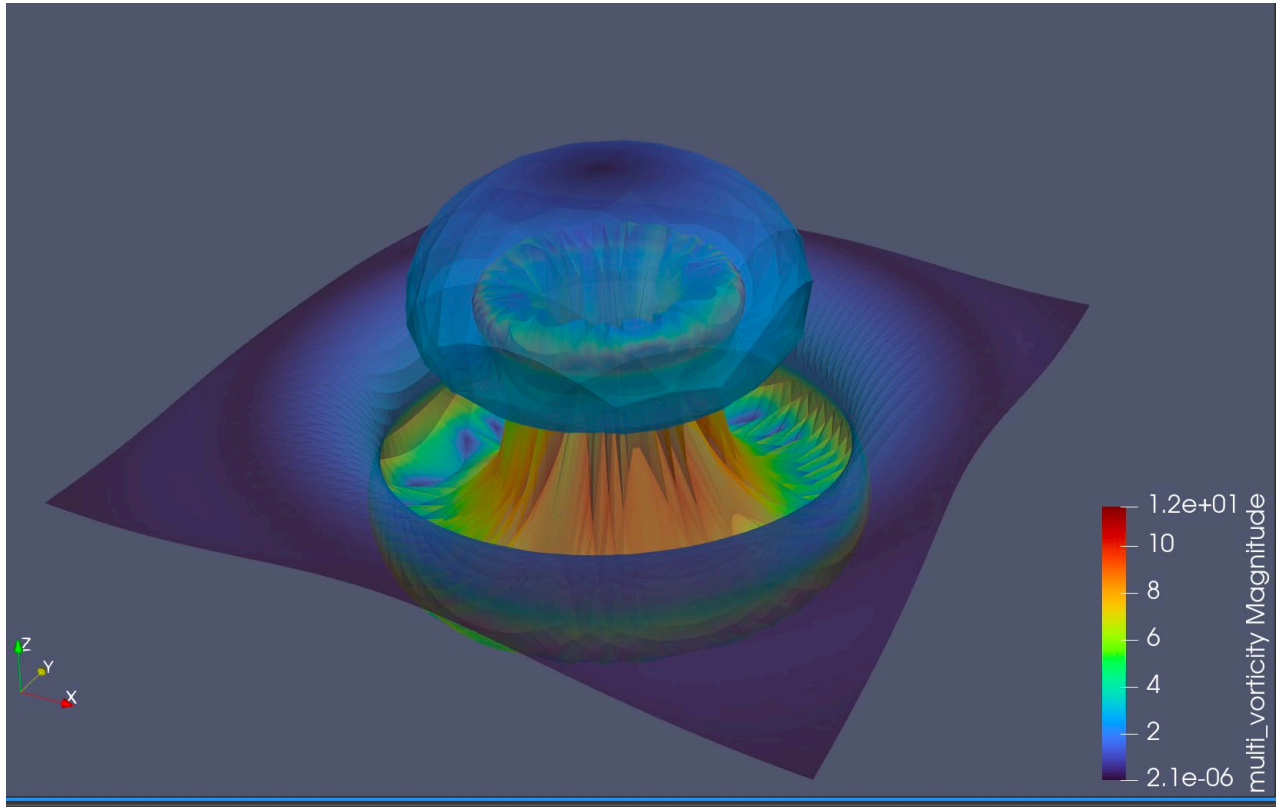
- Beatnik - full MPI/Kokkos fluid interface model proxy
  - **Need to supplement with space-filling curve spatial mesh decomposition**
  - **Fast multi-pole solver ideal for far field force calculation**
  - **Can use PVFMM as CPU-based far field solver, no open GPU FMM solvers**
- Now available from github: https://github.com/CUP-ECS/beatnik



High-order simulation of rocket rig Raleigh-Taylor instability

Proxy Development

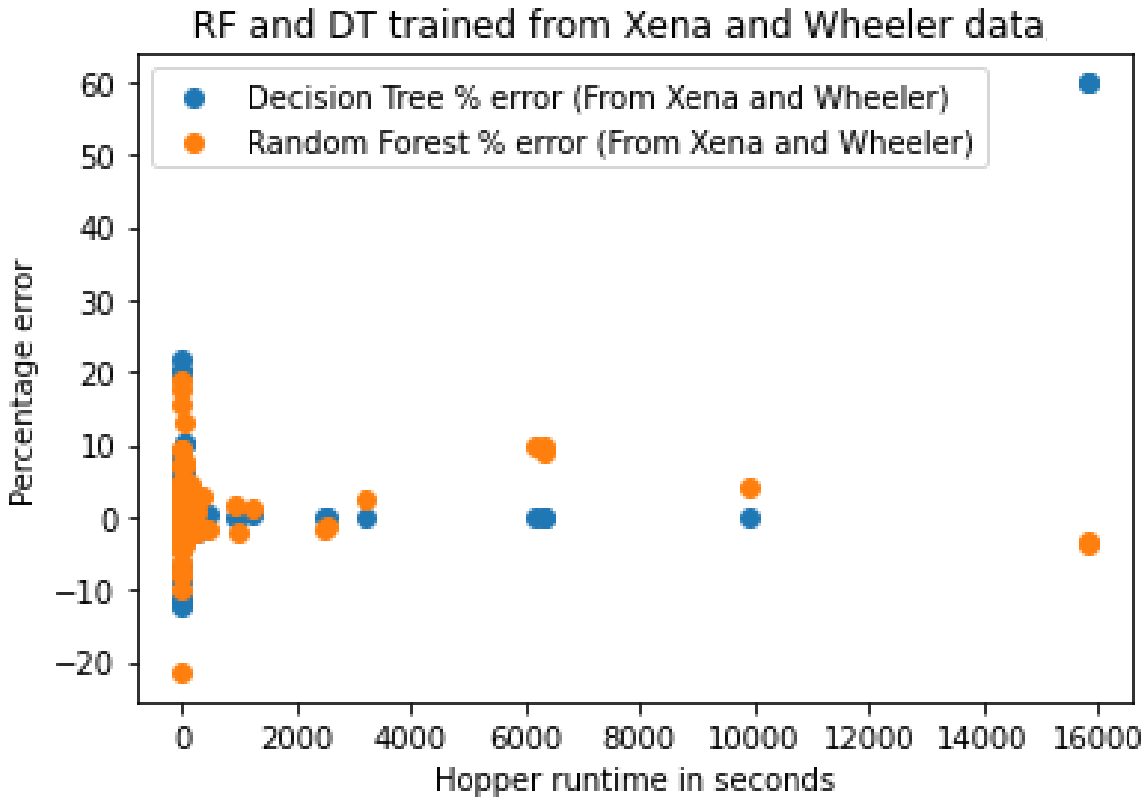
- Beatnik - full MPI/Kokkos fluid interface model proxy
  - **Would like to couple with both production app and appropriate open benchmark**
  - **Would provide good coupled code example**
  - **Need help identifying appropriate production codes, open benchmark, and coupling approach**
- Now available from github: https://github.com/CUP-ECS/beatnik



High-order simulation of rocket rig Raleigh-Taylor instability

Performance Assessment/Prediction

- Issue: How to predict impact of communication abstraction changes on application performance
- Approach: Communication rooflines plus machine learning
  - Characterize the performance of communication primitives on different systems using roofline models
  - Train machine learning models to predict application runtime based on changes in compute and communication rooflines
  - Can predict runtimes of simole halo codes when changing between GPU and network architectures
  - Working to expand rooflines to more complex benchmarks (HPCG, Beatnik, etc.) and communication abstractions
- Student research lead UNM student Akhil Asalundgutti
  - Student funded by NSF award OAC-1807583
  - NSF funded goal to model impact of HPC workload co-location to inform schedulers such as Flux.



Education Efforts

- Importance of education increasing due to need to train students
- Focused meeting at July in-person hackathon on developing list of materials/topics for MPI implementation course
- Phase 1: 5000 foot overview of important MPI concepts
  - Message Passing Semantics
  - Network Hardware Basics
  - Messaging Performance Basics
  - High-level messaging protocol issues – (e.g. long/short protocols, eager/rendezvous)
  - Simple Communication Performance Models

Education Efforts (2)

- Importance of education increasing due to need to train students
- Focused meeting at July in-person hackathon on developing list of materials/topics for MPI implementation course
- Phase 2: Dig into the details
  - Implement simple messaging primitives using UCX
  - Simple projects on two-sided communication, one-sided communication, collective communications
  - Introduction to GPU and accelerator computing issues
  - Introduction to protocol-level optimization/tuning (e.g. collective algorithms)
- Provide tutorials at conferences

Education Implementation Plan

- Partitioned communication tutorial provided using MPIAdvance implementation at EuroMPI 2022
- 1-2 colloquium seminars per semester lead by faculty leads to brief key issues not captured in example books and papers
- UNM students working in fall 2022 seminar to identify and summarize key references, gaps in information available
  - MPICH and OpenMPI overview papers/references
  - Historic optimization/characterization papers
  - Sandia personnel (Dosanjh, Levy) already helping, plan to engage more broadly
- Plan to offer prototype class (formally or informally) in spring 2023
- New UTC research professor (Prof. David Walker from Cardiff) joining project to assist with multiple issues, but primarily education.