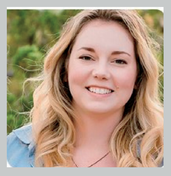
THE UNIVERSITY OF NEW MEXICO

SC23 DENVER NOV 12-17

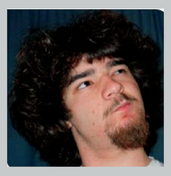
Ryan Scherbarth
Major: Computer Science and Mathetmatics

Maisy Dunlavy
Major: Computer Science and Linguistics

Alex Knigge
Major: Computer Science and Mathematics



Ellie Larence
Major: Computer Science and Biology



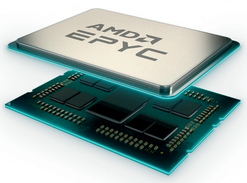
Carter Frost
Major: Computer Science and ASL

Aziz Raad
Major: Computer Science and Marketing

The University of New Mexico is a Hispanic-serving Tier 1 research institution located in Albuquerque, NM. Our school frequently collaborates with national labs and other research organizations on topics such as robotics and biocomputational modeling. We are so proud that the team we have chosen to compete this year represents the unique diversity of our school and every member brings unique skill sets. All of the team members are first-year participants, and over half our team is composed of members who are traditionally considered to be underrepresented in STEM fields.

HARDWARE

Three identical nodes:



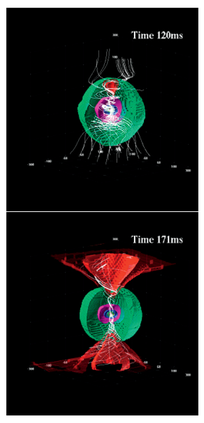
- 2x AMD EPYC 7413 (24c 180W) processors
- 16x64GB DDR4-3200 LR ECC memory
- 1x Micron 5300 Pro 480GB SATA Boot Drive
- 2x Kioxia CM6 7.68TB NVMe scratch drives
- 1x Rockport NC1225 300GB network card
- 4x NVIDIA A100 40GB PCIe accelerators
- Rockport SHFL networking switch
- Altus XE2214GT chassis

SOFTWARE

- Cluster Tool: Warewulf
- MPI: OpenMPI, MPICH
- Scheduler: Slurm
- OS: Rocky Linux 8.4
- GPU Software: CUDA
- Compiler: GCC and Intel oneAPI

THREE-DIMENSIONAL MAGNETO HYDRO DYNAMIC SIMULATION (3DMHD)

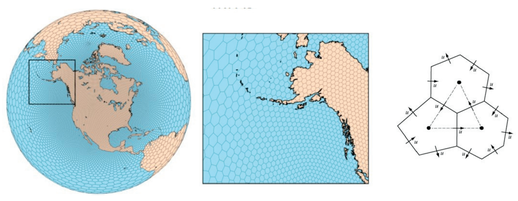
The Rast-3DMHD Simulation application uses complicated magnetohydrodynamic equations to study dense plumes in stratified layers. It was written by Dr. Rast's Laboratory for Atmopherice & Space Physics @ the University of Colorado Boulder.



Optimization Strategy

- Perform parameter sweep to find the optimal balance between processor spread and communication time
- Experiment with dynamically changing processor spread
- Use buffer arrays to reduce communication time
- Utilize CPUs for large data transferring
- Focus on efficient parallelization

ATMOSPHERIC MODEL FOR PREDICTION ACROSS SCALES (MPAS)



Atmospheric MPAS is an atmospheric modeling tool for climate, regional climate, and weather research. It uses centroidal Voronoi tessellations for the horizontal meshes (and was codeveloped by Los Alamos Labs in New Mexico!)

Optimization Strategy

- Use our pre-competition preliminary runs to find the combination of partitions and ranks on our nodes that will give us between 0.5 and 1 second timing integration steps
- Ensure we have a balanced mesh [all subdomains are close to equal]
- Use our experience with GPROF to adjust or replace any inefficient data structures

PREPARATION AND STRATEGY

During the competition, we plan to:

- Experiment with HPL and HPCG efficiency and run time with CPU's vs. GPU's on the cluster early on in the competition
- Have data backups so that we can get everything installed efficiently and start our runs as quickly as possible
- Have a working schedule of how long we will spend benchmarking each application
- Continuously submit our best runs as we go to avoid being derailed by unplanned issues that could prevent us from submitting a final run
- Pay close attention to our power limits
- In the case that we run into power contraints, look into throttling our GPUs [the last 20-30% of power sent to GPUs generally only yields around a 5% performance increase]

We have worked to all become very familiar with the benchmarks and applications so that we are able to shape our optimization strategies around a deep understanding of our hardware constraints. We included our preliminary data that we generated to find optimal values for HPL, and have worked similarly with HPCG.

We meet for 2.5 hours each week to prepare for the competition, as well as work hard individually outside of meetings to make sure we are prepared to win! To learn more about us, scan the QR code.

HPL

- Used MPI Implementation of HPL for our experiments on AMD EPYC cpu's
- Explored the calculated theoretical max exploration
- Given the memory limitations of our test machine, we found the optimal value for N to be 32768
- Performed parameter sweep on NB to find the optimal size (shown below)

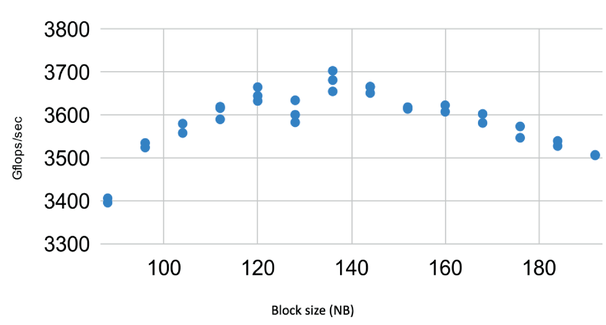


Figure: Graph of HPL Benchmark with different NB sizes to find the highest GFlops per second.

HPCG

- Performed automated parameter sweep for the size of the 3D array
- Experimented with different compilers and MPI libraries
- Utilized MPI/OpenMP/slurm cpu binding options
- Used vendor-optimized source code

MLPERF

- Benchmarking application for machine learning will benefit from NVIDIA A100 GPU offloading
- Use NVIDIA-optimized implementation to improve timing
- Collect power measurements from MaxQ
- Perform automated parameter sweep on CM script for tuning

FACULTY MENTOR/ADVISOR -
GEORGE MATTHEW FRICKE

LOGISTICS COORDINATORS -
NICHOLAS BACON & TROY REDFEARN

PENGUIN COMPUTING

CENTER FOR ADVANCED RESEARCH COMPUTING

Intersect360 RESEARCH