OAK RIDGE National Laboratory

Early Experiences on the OLCF Frontier System with AthenaPK and Parthenon-Hydro

John K. Holmen, Oak Ridge National Laboratory

Philipp Grete, Hamburg Observatory, University of Hamburg

Verónica G. Vergara Larrea, Oak Ridge National Laboratory

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY

Motivation

- Nation's first exascale system, Frontier, is being prepared for production and end users
- Acceptance testing critical for ensuring functionality, performance, and usability
- Code selection is important
- Talk captures early experiences on Frontier with two selected codes
  - AthenaPK and Parthenon-Hydro

https://www.flickr.com/photos/olcf/52117623843/in/album-72177720299483343/

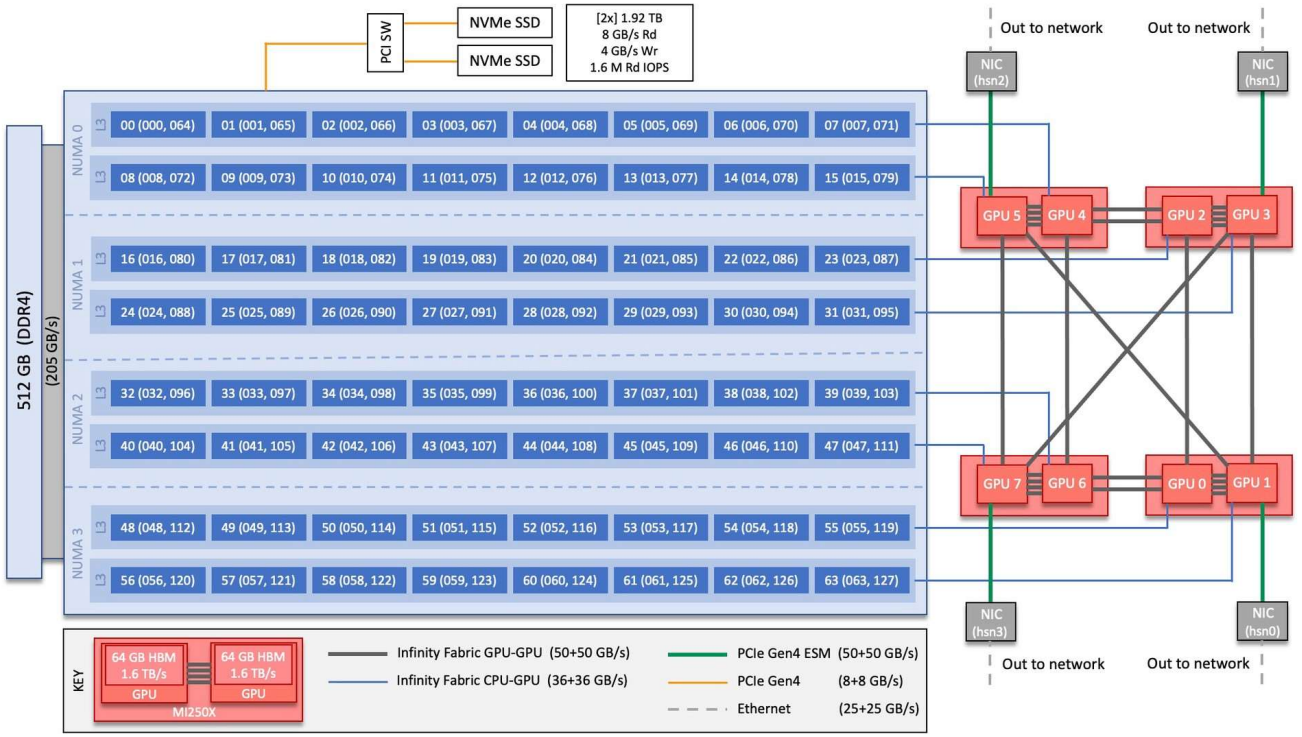
Frontier

- Frontier consists of 9,408 HPE Cray EX235a nodes
  - One 64-core AMD EPYC 7A53 CPU and four AMD MI250X GPUs per node
  - 512 GB of DDR4 memory and 512 GB of high-bandwidth memory per node
- Interconnected via HPE's Slingshot
- Lustre file system, Orion, with 679 PB usable namespace
- #1 on November's Top500 List

https://www.flickr.com/photos/olcf/52117623763/in/album-72177720299483343/

Frontier Node Configuration

- 8x L3 cache regions each spanning 8x cores
- 8x GCDs each mapped to an L3 cache region
- 4x NUMA domains per node
- 2x L3 cache regions per NUMA

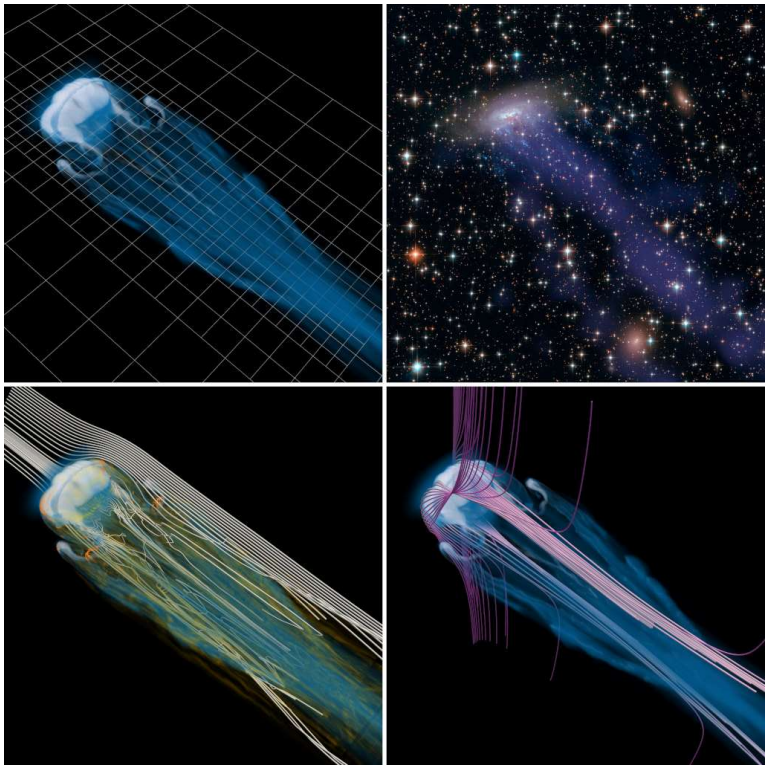


https://docs.olcf.ornl.gov/_images/Frontier_Node_Diagram.jpg

OAK RIDGE National Laboratory

AthenaPK

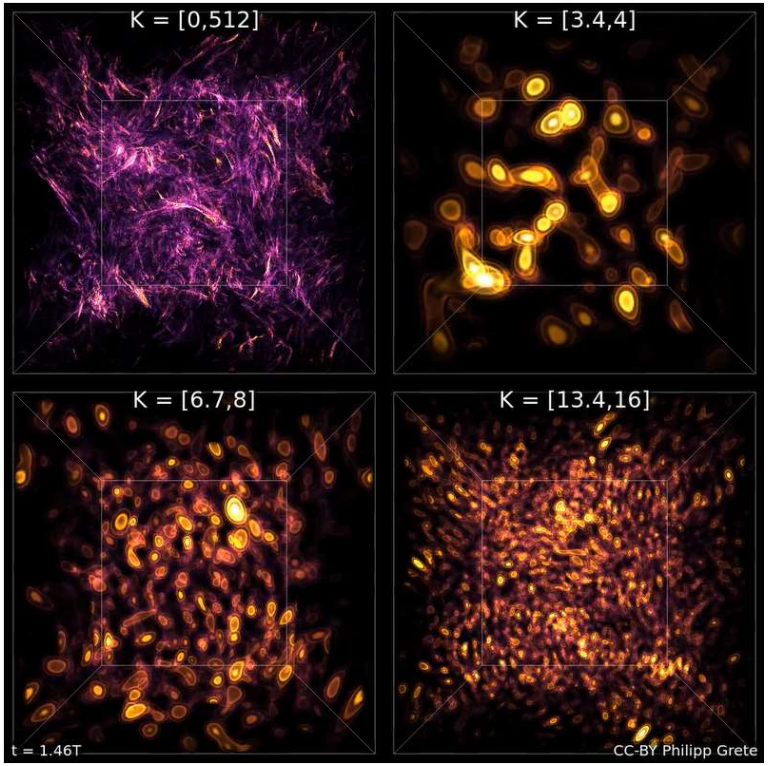
- AthenaPK is an astrophysical magnetohydronamics (MHD) code
  - https://arxiv.org/abs/2202.12309
- MHD is the study of the dynamics of electrically conducting fluids
  - e.g., plasmas, liquid metals
- AthenaPK has been used to simulate:
  - magnetized galaxy clusters
  - cloud crushing in galactic outflows
  - magnetohydrodynamic turbulence



https://pgrete.de/images/cloud_l.jpg

AthenaPK

- AthenaPK combines Athena++ with Parthenon and Kokkos
  - https://github.com/parthenon-hpc-lab/athenapk
- Parthenon is an adaptive mesh refinement framework
  - https://github.com/parthenon-hpc-lab/parthenon
- Kokkos is a performance portability layer
  - https://github.com/kokkos/kokkos



https://pgrete.de/dl/vids/Shell_Decomposition-Grete_et_al_2017_PoP.mp4

Parthenon

- Performance portable block-structured adaptive mesh refinement framework
  - Originates from Athena++
  - Performance portable via Kokkos
- Achieves performance by:
  - Device-first approach for data,
  - Packing of data across blocks to hide launch latency, and
  - Device-to-device communication via asynchronous, one-sided MPI
- Available on GitHub with contributions welcomed:
  - https://github.com/parthenon-hpc-lab/parthenon

Parthenon-Hydro

- Parthenon is a finite volume, compressible hydrodynamics sample implementation (i.e., miniapp) using Parthenon
- 1,400 lines of C++ showing use of Parthenon interfaces
  - Serves as an external integration and performance test
- Supports 1D, 2D, and 3D compressible hydrodynamics on uniform and (static and adaptive) multi-level meshes
- Available on GitHub with contributions welcomed:
  - https://github.com/parthenon-hpc-lab/parthenon-hydro

Kokkos

- C++ library enabling portable, thread-scalable code optimized for CPU, GPU, and MIC architectures
  - Back-ends to models such as CUDA, OpenMP, HIP, and SYCL
- Provides abstractions to control:
  - how/where kernels are executed,
  - where data is allocated, and
  - how data is mapped to memory
- Enables performance portability
  - Developers remain responsible for writing performant code

Large-Scale Simulation

- Codes target large-scale simulation across diverse systems
- Demonstrated scalability across leadership-class systems
  - e.g., Frontera, Frontier, Summit
- Scalability achieved in a portable manner
- Addressed challenges with performance portability layers
  - e.g., Kokkos

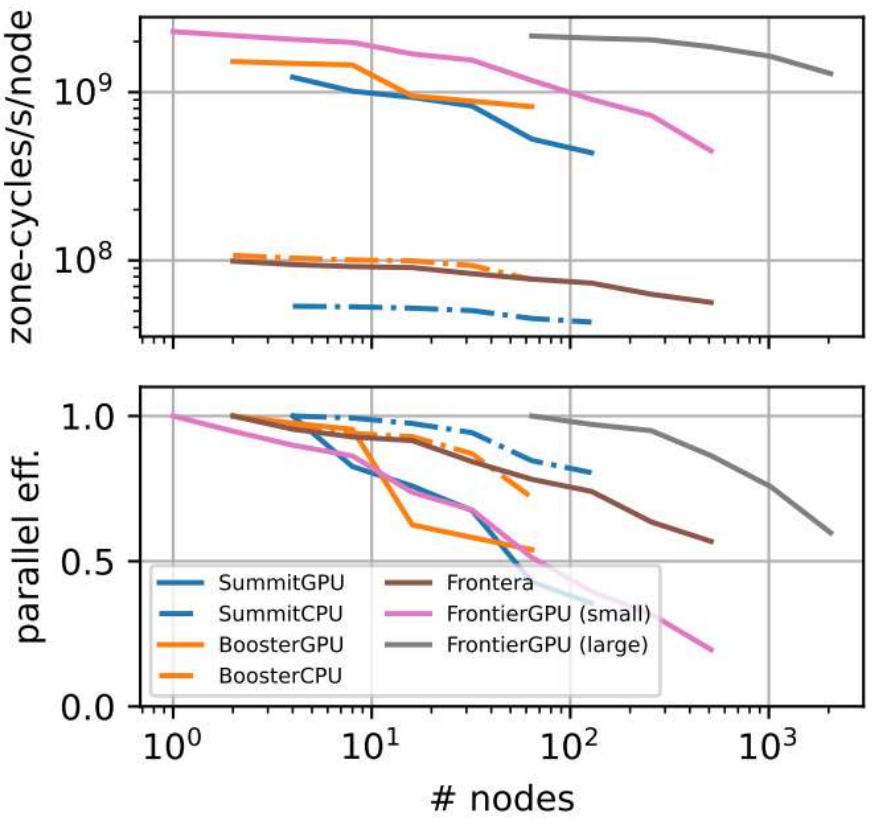


Figure 10. Strong scaling of PARTHENON-HYDRO on uniform grids on various supercomputers with raw performance in zone-cycles per second per node (top), parallel efficiency (bottom). On Summit GPUs (CPUs) the mesh size was fixed to

Grete, Philipp, et al. "Parthenon—a performance portable block-structured adaptive mesh refinement framework." *The International Journal of High Performance Computing Applications* (2022): 10943420221143775.

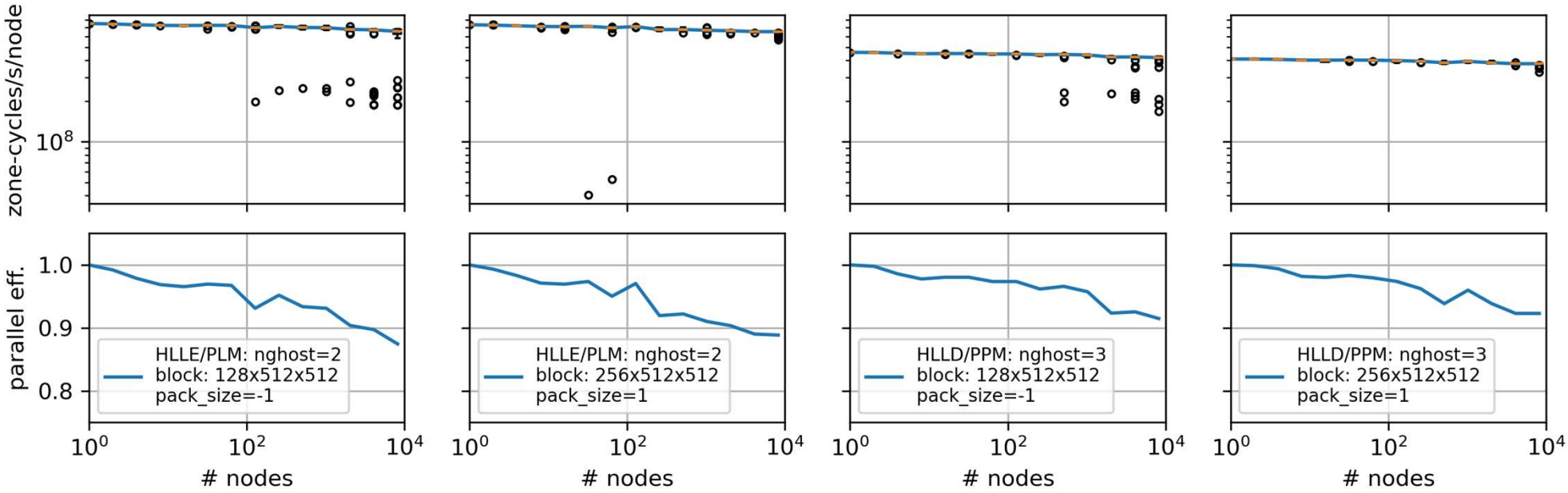
Experiments

- Modeled a 3-dimensional linear wave problem
  - Implemented in both AthenaPK and Parthenon-Hydro
- Per-cell work constant across domain
  - Easy distribution across devices
  - Predictable wallclock times
- Experiments vary domain decomposition approaches
  - Manages the size and amount of MPI messages

Domain Decomposition

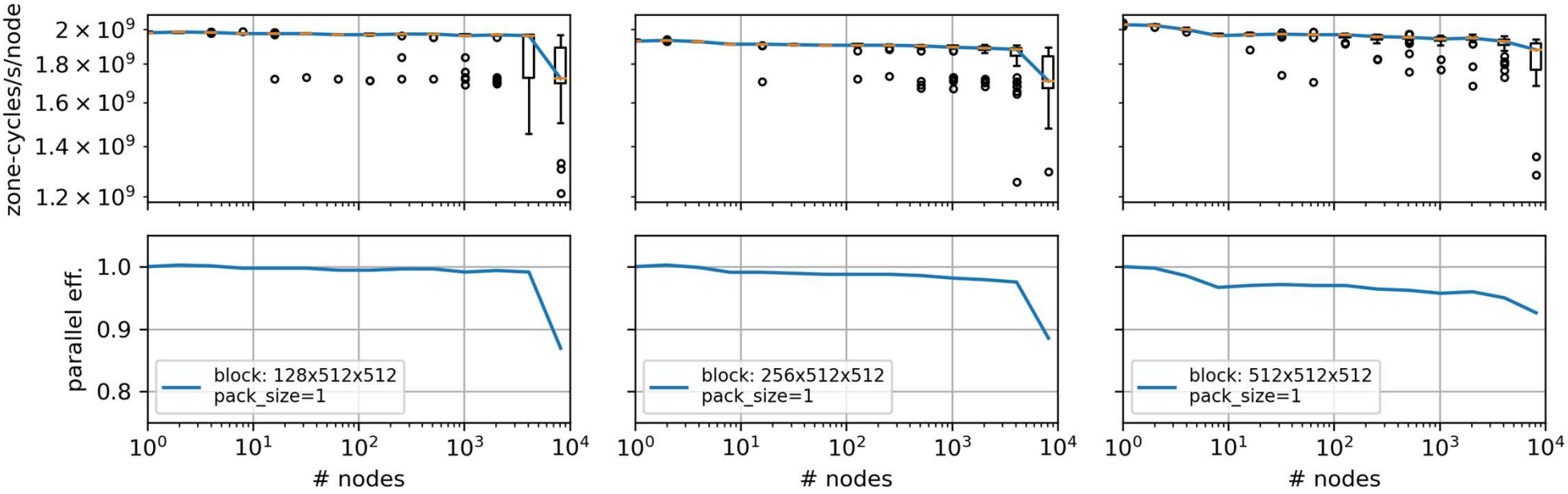
- Domains are decomposed into blocks of cells
- Blocks can be logically packed using a pack_size parameter
  - Increases work size within kernels and reduces launch latency
- Explored various blocks sizes, numbers of blocks per GCD, numbers of blocks per pack, and numbers of packs per GCD
- Block size impacts the number and size of MPI messages sent
  - Smaller block sizes -> more, yet smaller, messages
  - Larger block sizes -> fewer, yet larger, messages
- Packs allow data to be sent while other buffers are being filled

Weak-Scaling Studies: AthenaPK



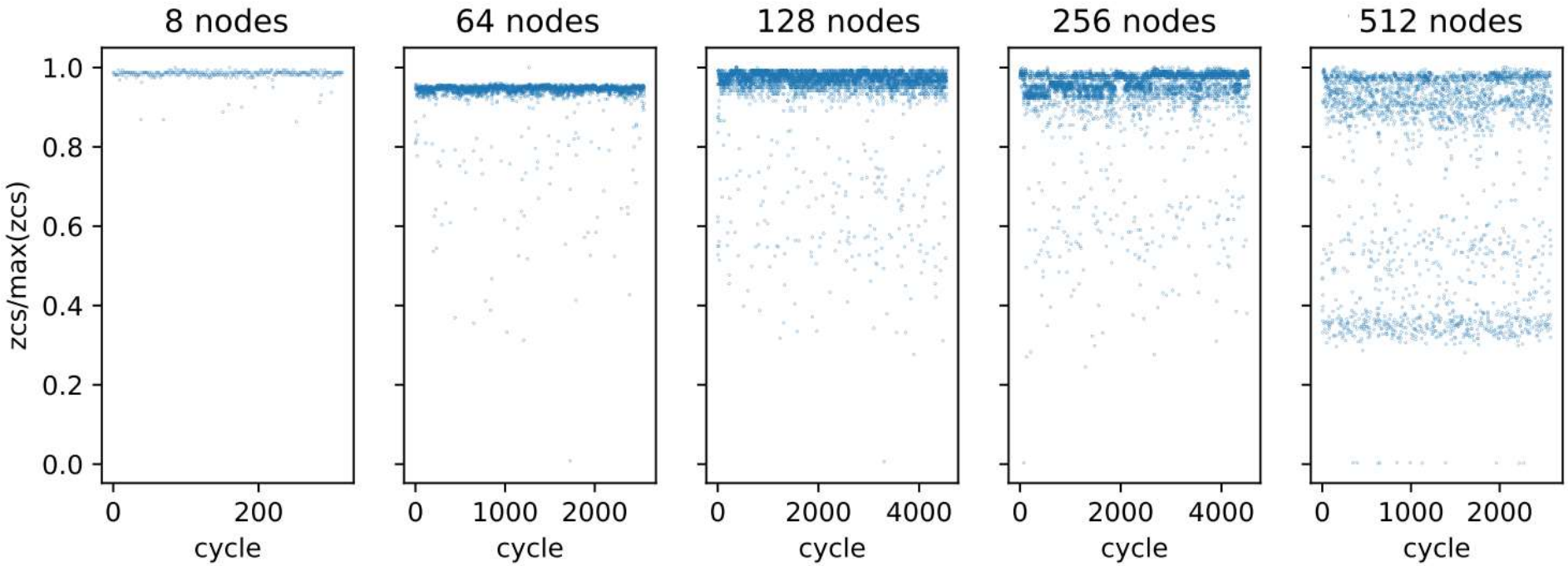
- >90% weak-scaling efficiency achieved to 9,216 Frontier nodes

Weak-Scaling Studies: Parthenon-Hydro



- >90% weak-scaling efficiency achieved to 9,216 Frontier nodes

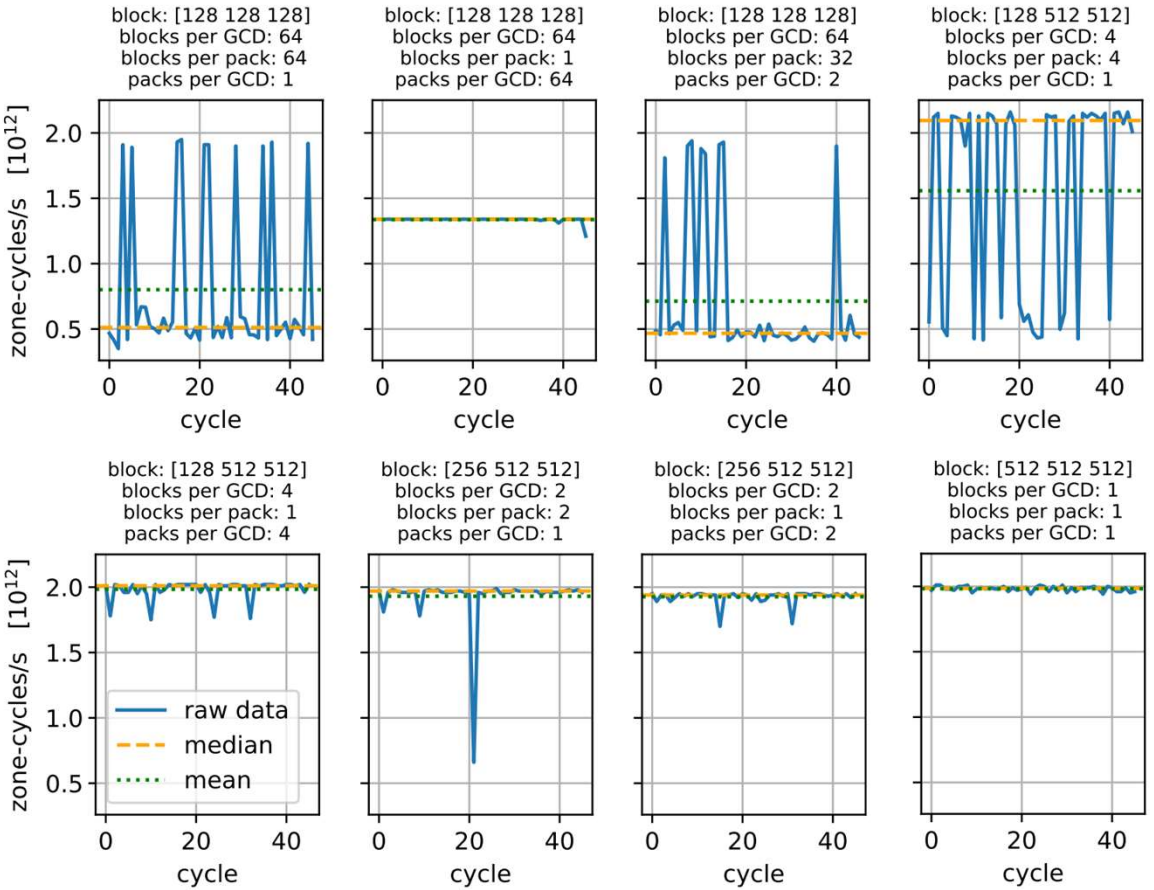
Performance Variability



- Initial experiments had per-cycle slowdowns as large as 80%

Performance Variability

- Experimented with:
  - block sizes,
  - blocks per GCD, and
  - blocks per pack
- In general:
  - 1 block per pack results in the most consistent performance
  - 4 blocks per pack or more results in greater variation



System Testing: Code Selection

- User codes helpful for stressing interesting third-party library configurations and "real-world" system use
- AthenaPK and Parthenon-Hydro added for various reasons
  - Successful performance portable demonstrations of leadership class system use
  - Effective use of MPI+Kokkos at large-scale
  - Easy to build, run, and scale
  - Well documented
  - Friendly and supportive developer community

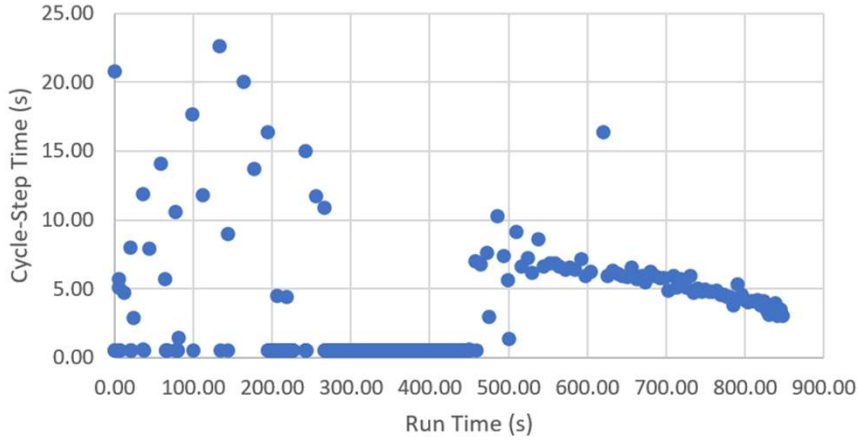
System Testing: Problem Design

- Two types of Frontier test: large-scale and "every node"
- Large-scale tend to target 1/8, 1/4, 1/2, and full system use
  - Other node counts explored as issues are identified
- "Every node" based on single-node tests
  - Launch a group of individual single-node jobs across the system, e.g.,:
    - Launching 9,408 single-node AthenaPK tests across Frontier in a single job
  - Helpful for isolating "bad" nodes and node failures
  - Invaluable for ability to pinpoint problematic nodes

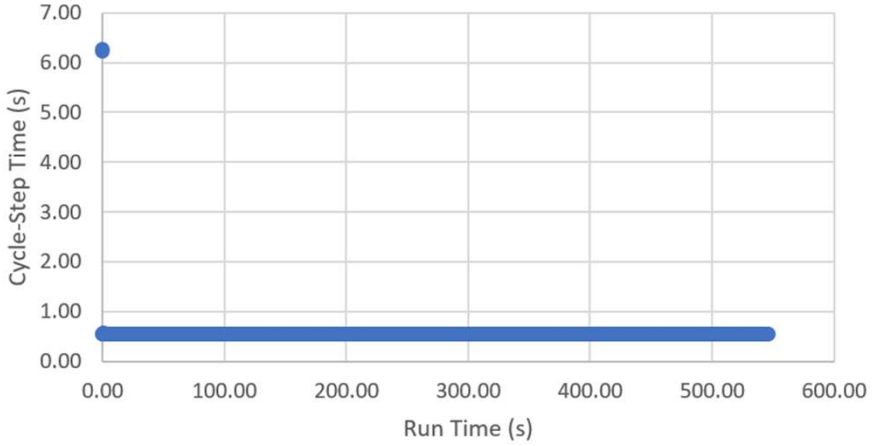
System Testing: Early Challenges

- Early "every node" tests had significant per-cycle slowdowns
- Early tests output per-node data to the same directory
- During runtime, rank 0 checks once per cycle for the existence of a file in the run directory
  - Non-issue for large-scale runs
- Individual directories for per-node data resolved slowdowns

Before



After



Conclusions

- Early experiences have been smooth
  - Issues easily worked around
  - Good performance achievable
- MPI+Kokkos scales well across Frontier
  - >90% weak-scaling efficiency achieved across 9,216 nodes
- Successful collaboration has been beneficial to both parties
  - Helped OLCF identify problematic nodes
  - Helped Hamburg Observatory demonstrate capabilities

Future Work

- At the OLCF:
  - Continue to extend Frontier's test coverage
    - Seeking similar opportunities to collaborate with users
  - Specifically, extending coverage of performance portability layers
    - e.g., Kokkos, OCCA, RAJA, SYCL/DPC++
- At the Hamburg Observatory:
  - INCITE runs on Frontier simulating magnetized plasma jets from active galactic nuclei
    - More detailed evaluation of the ordering of filling buffers and sending messages
  - Development efforts to decouple the global block packing from a communication related pack

Innovative and Novel Computational Impact on Theory and Experiment (INCITE) Program for 2024

- Seeking proposals for high-impact, computationally intensive research campaigns in a broad array of science, engineering and computer science domains.
- Proposals are due June 16, 2023.
- INCITE Informational Webinars are scheduled for April 25 & May 2, 2023.
- Early career track continues in 2023.
- For more information, visit http://www.doeleadershipcomputing.org/

U.S. DEPARTMENT OF ENERGY | Office of Science

U.S. DEPARTMENT OF ENERGY INCITE LEADERSHIP COMPUTING

Argonne NATIONAL LABORATORY

OAK RIDGE National Laboratory

Questions?

This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101030214.

holmenjk@ornl.gov

pgrete@hs.uni-hamburg.de

vergaravg@ornl.gov

ORNL is managed by UT-Battelle, LLC for the US Department of Energy

U.S. DEPARTMENT OF ENERGY