Ares – Simulating Type Ia Supernovae on Heterogeneous HPC Architectures

LANDON DYKEN, University of Illinois Chicago, USA
ALEXANDER HOLAS, Heidelberg Institute for Theoretical Studies, Germany
MARK IVAN UGALINO, University of Maryland College Park, USA
MD NAGEEB BIN ZAMAN, Louisiana State University, USA

ACM Reference Format:
Landon Dyken, Alexander Holas, Mark Ivan Ugalino, and Md Nageeb Bin Zaman. 2023. Ares – Simulating Type Ia Supernovae on Heterogeneous HPC Architectures (LA-UR-23-29154) . 1, 1 (September 2023), 2 pages.

Type Ia Supernovae are highly luminous thermonuclear explosions of white dwarfs which serve as standardizable distance markers for investigating the accelerating expansion of our Universe. Most existing supernovae simulation codes are only designed to run on homogeneous CPU-only systems and do not take advantage of the increasing shift towards heterogeneous architectures in HPC. To address this, we present Ares, the first performance portable massively-parallel code for simulating thermonuclear burn fronts. By creating multi-physics modules using the Kokkos and Parthenon frameworks, we are able to scale supernovae simulations to distributed HPC clusters operating on any of CUDA, HIP, SYCL, HPX, OpenMP and serial backends. We evaluate our application by conducting weak and strong scaling studies on both CPU and GPU clusters, showing the efficiency of our method for a diverse set of targets.

1 INTRODUCTION

Simulating the thermonuclear explosion of a white dwarf is an inherently multi-scale problem. The spatial dimensions of the problem range from the diameter of the white dwarf at about 4-10 thousand kilometers to the nuclear burning region at about 1 cm thickness. This poses a huge challenge in performing computational simulations of these systems in a uniform mesh. To resolve relevant physical mechanisms at every scale possible, we need Adaptive Mesh Refinement (AMR), which enables us to selectively refine regions in our computational domain. However, most of these AMR simulations are designed to run only on homogeneous CPU-only architectures. There do exist efforts to design these codes for GPUs, but they are vendor-specific. To address these issues, we create ARES, the first performance portable hydrodynamics code for thermonuclear explosions based on the block-structured AMR framework Parthenon [1]. To exploit on-node data parallelism, Parthenon internally uses Kokkos [4] to target various device architectures which enables us to resolve scales that are out of reach for current state-of-the-art codes.

2 METHODOLOGY

As a starting point, we use the open-source hydrodynamics code Parthenon-Hydro [1] which illustrates the capabilities of the Parthenon framework in handling AMR for hydrodynamics simulations. To simulate a type Ia explosion,we build

Authors' addresses: Landon Dyken, ldyke@uic.edu, University of Illinois Chicago, USA; Alexander Holas, alexander.holas@h-its.org, Heidelberg Institute for Theoretical Studies, Germany; Mark Ivan Ugalino, ugalino@umd.edu, University of Maryland College Park, USA; Md Nageeb Bin Zaman, mzama14@lsu.edu, Louisiana State University, USA.

© 2023 Association for Computing Machinery.
Manuscript submitted to ACM

upon Parthenon-Hydro by adding a nuclear statistical equilibrium (NSE) solver to accurately solve for energy released through thermonuclear burning based on [2], a monopole gravity solver for self-gravity, and a stellar equation of state (EOS) which supports conditions in the degenerate gas interior of white dwarfs based on [3].

3 DISCUSSION

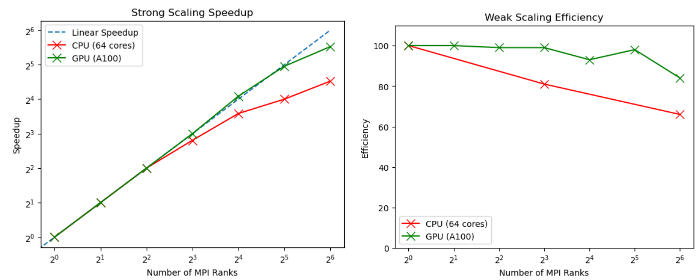

Fig. 1. Strong and weak scaling studies for a gas sphere collapse problem using Ares on the LANL Chicoma system for varying number of MPI ranks.

We conduct weak and strong scaling on CPU and GPU nodes to understand the scalability of our application on the LANL Chicoma system. We choose a gas sphere collapse problem to conduct our scaling studies, and evaluate performance after 1000 cycles. Scaling studies were done by varying the number of MPI ranks from 1 to 64, with each rank using either OpenMP-enabled Rome 64-core AMD EPYC 7H12 cores or 1 NVIDIA A100 Tensor Core GPU. We find that the strong scaling shows perfect speedup up to 32 ranks on GPUs. On average, runs using one MPI rank per GPU scale up to 50% faster than those that use one MPI rank per 64 OpenMP threads. The weak scaling plot outlines that the CPU runs maintain up to 66% efficiency at 64 MPI ranks with OpenMP. The weak scaling results for GPU runs show an efficiency of 84% using 64 GPUs on 16 nodes.

4 CONCLUSION

By utilizing the Kokkos and Parthenon frameworks, we were able to show the possibility for astrophysics simulation codes to utilize heterogeneous HPC architectures. We present verified physics modules for gravity, nuclear network, and equation of state necessary for stellar applications within these frameworks and show that these can take advantage of modern supercomputing resources. We hope this exploratory work can encourage the adoption of new programming standards that will ensure that recently developed codes maintain their competitiveness as HPC architectures become more diverse.

REFERENCES

[1] Philipp Grete, Joshua C Dolence, Jonah M Miller, Joshua Brown, Ben Ryan, Andrew Gaspar, Forrest Glines, Sriram Swaminarayan, Jonas Lippuner, Clell J Solomon, et al. 2022. Parthenon—a performance portable block-structured adaptive mesh refinement framework. *The International Journal of High Performance Computing Applications* (2022), 10943420221143775.

[2] I R Seitenzahl, D M Townsley, F Peng, and J W Truran. 2009. Nuclear statistical equilibrium for type Ia supernova simulations. *Atomic Data and Nuclear Data Tables* 95, 1 (1 2009). https://doi.org/10.1016/j.adt.2008.08.001

[3] F. X. Timmes and F. Douglas Swesty. 2000. The Accuracy, Consistency, and Speed of an Electron-Positron Equation of State Based on Table Interpolation of the Helmholtz Free Energy. *The Astrophysical Journal Supplement Series* 126, 2 (feb 2000), 501. https://doi.org/10.1086/313304

[4] Christian R Trott, Damien Lebrun-Grandié, Daniel Arndt, Jan Ciesko, Vinh Dang, Nathan Ellingwood, Rahulkumar Gayatri, Evan Harvey, Daisy S Hollman, Dan Ibanez, et al. 2021. Kokkos 3: Programming model extensions for the exascale era. *IEEE Transactions on Parallel and Distributed Systems* 33, 4 (2021), 805–817.