CASTRO: A Massively Parallel Compressible Astrophysics Simulation Code

Ann Almgren[1], **Maria Barrios Sazo**[2], **John Bell**[1], **Alice Harpole**[2], **Max Katz**[3], **Jean Sexton**[1], **Donald Willcox**[1], **Weiqun Zhang**[1], and **Michael Zingale**[2, 4]

1 Center for Computational Sciences and Engineering, Lawrence Berkeley National Laboratory **2** Department of Physics and Astronomy, Stony Brook University **3** NVIDIA Corporation **4** Center for Computational Astrophysics, Flatiron Institute

Summary

Castro is a highly parallel, adaptive mesh, multiphysics simulation code for compressible astrophysical flows. It has been used to simulate different progenitor models of Type Ia supernovae, X-ray bursts, core-collapse and electron capture supernovae, and dynamics in exoplanets. Together, Castro, the low Mach number code MAESTROeX (Fan, Nonaka, Almgren, Harpole, & Zingale, 2019), and the cosmology code Nyx (Almgren, Bell, Lijewski, Lukić, & Van Andel, 2013) make up the AMReX-Astrophysics Suite of open-source, adaptive mesh, performance portable astrophysical simulation codes.

The core hydrodynamics solver in Castro (Almgren et al., 2010) is based on the directionally unsplit corner transport upwind method of Colella (1990) with piecewise parabolic reconstruction (Colella & Woodward, 1984). Modeling reactive flows in stellar environments is a core capability of Castro. Astrophysical reaction networks are stiff and require implicit integration techniques for accurate and stable solutions. In Castro, we have several modes of coupling the reactions to hydro. The simplest method is the traditional operator splitting approach, using Strang splitting to achieve second-order in time. However, when the reactions are energetic this coupling can break down, and we have two different implementations based on spectral deferred corrections (SDC), a method that aims to prevent the hydro and reactions from becoming decoupled. The simplified SDC method uses the CTU PPM hydro together with an iterative scheme to fully couple the reactions and hydro, still to second order (Zingale et al., 2019). Alternatively, we have implemented a traditional SDC method that couples hydro and reactions to both second and fourth-order in space and time (Zingale, Katz, et al., 2019) (at present, this method is single-level only). The Strang splitting and simplified SDC methods have a retry scheme, where a timestep will be rejected and retried at a smaller, subcycled timestep if the burning solve fails to meet its tolerance, negative densities are generated, or we violate one of the timestepping criteria.

In addition to reactive hydrodynamics, Castro includes full self-gravity with isolated boundary conditions and rotation, both implemented in an energy-conserving fashion, explicit thermal diffusion, and gray (Zhang, Howell, Almgren, Burrows, & Bell, 2011) and multigroup (Zhang et al., 2013) flux limited diffusion radiation hydrodynamics. A constrained transport MHD solver based on the CTU algorithm is also available, and can use the same physics source terms. Castro can use an arbitrary equation of state and reaction network, and these microphysics routines are provided by the StarKiller project (StarKiller Microphysics Development Team et al., 2020).

Castro is built on the AMReX (Zhang et al., 2019) adaptive mesh refinement (AMR) library and is largely written in C++ with a few Fortran compute kernels. AMR levels are advanced

at their own timestep (subcycling) and jumps by factors of 2 and 4 are supported between levels. We use MPI to distribute AMR grids across nodes and use logical tiling with OpenMP to divide a grid across threads for multi-core CPU machines (exposing coarse-grained parallelism) or CUDA to spread the work across GPU threads on GPU-based machines (fine-grained parallelism). All of the core physics can run on GPUs and has been shown to scale well to thousands of GPUs (Zingale, Almgren, et al., 2019) and hundreds of thousands of CPU cores (Zingale et al., 2018). For performance portability, we use the same source code for both CPUs and GPUs, and implement our parallel loops in an abstraction layer provided by AMReX. An abstract parallel for loop accepts as arguments a range of indices and the body of the loop to execute for a given index, and the AMReX backend dispatches the work appropriately (e.g., one zone per GPU thread). This strategy is similar to the way the Kokkos (Edwards, Trott, & Sunderland, 2014) and RAJA (Beckingsale et al., 2019) abstraction models provide performance portability in C++.

Statement of Need

While there are a number of astrophysical hydrodynamics simulation codes, Castro offers a few unique features. The original motivation for developing Castro was to build a simulation code based on a modern, well-supported AMR library (BoxLib which evolved to AMReX), using unsplit integration techniques and targeting problems in nuclear astrophysics. The radiation solver was a key design consideration in the early development. The large developer community contributing to AMReX (representing a large number of application codes across various domains) results in Castro continually gaining optimizations for new architectures. As Castro evolved, we adopted a fully open development model (as does the Enzo (Bryan et al., 2014) code, for example). We pride ourselves in making all of the science problems available in the Castro git repository as we are developing them, and the infrastructure we use for running our problems and writing our science papers is publicly available in the AMReX-Astro organization. Other simulation codes, like Flash (Fryxell et al., 2000), also work with a general equation of state and reaction network, but Castro is unique in focusing on spectral deferred correction techniques for coupling the hydro and reactions. Finally, while some astrophysics codes have performance portable forks (like K-Athena (Grete, Glines, & O'Shea, 2021), which uses Kokkos), Castro's current design – which targets both CPUs and GPUs for all solvers – achieves performance portability as a core design principle, avoiding the need for a fractured development model.

Acknowledgments

References

Almgren, A. S., Beckner, V. E., Bell, J. B., Day, M. S., Howell, L. H., Joggerst, C. C., Lijewski, M. J., et al. (2010). CASTRO: A New Compressible Astrophysical Solver. I.

Hydrodynamics and Self-gravity. *Astrophysical Journal*, *715*(2), 1221–1238. doi:[10.1088/0004-637X/715/2/1221](https://doi.org/10.1088/0004-637X/715/2/1221)

Almgren, A. S., Bell, J. B., Lijewski, M. J., Lukić, Z., & Van Andel, E. (2013). Nyx: A Massively Parallel AMR Code for Computational Cosmology. *Astrophysical Journal*, *765*(1), 39. doi:[10.1088/0004-637X/765/1/39](https://doi.org/10.1088/0004-637X/765/1/39)

Beckingsale, D. A., Burmark, J., Hornung, R., Jones, H., Killian, W., Kunen, A. J., Pearce, O., et al. (2019). RAJA: Portable performance for large-scale scientific applications. In *2019 ieee/acm international workshop on performance, portability and productivity in hpc (p3hpc)* (pp. 71–81). doi:[10.1109/P3HPC49587.2019.00012](https://doi.org/10.1109/P3HPC49587.2019.00012)

Bryan, G. L., Norman, M. L., O'Shea, B. W., Abel, T., Wise, J. H., Turk, M. J., Reynolds, D. R., et al. (2014). ENZO: An Adaptive Mesh Refinement Code for Astrophysics. *Astrophysical Journal Supplement Series*, *211*(2), 19. doi:[10.1088/0067-0049/211/2/19](https://doi.org/10.1088/0067-0049/211/2/19)

Colella, P. (1990). Multidimensional upwind methods for hyperbolic conservation laws. *Journal of Computational Physics*, *87*, 171–200. doi:[10.1016/0021-9991(90)90233-Q](https://doi.org/10.1016/0021-9991(90)90233-Q)

Colella, P., & Woodward, P. R. (1984). The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations. *Journal of Computational Physics*, *54*, 174–201. doi:[10.1016/0021-9991(84)90143-8](https://doi.org/10.1016/0021-9991(84)90143-8)

Edwards, H. C., Trott, C. R., & Sunderland, D. (2014). Kokkos: Enabling manycore performance portability through polymorphic memory access patterns. *Journal of Parallel and Distributed Computing*, *74*(12), 3202–3216. doi:[10.1016/j.jpdc.2014.07.003](https://doi.org/10.1016/j.jpdc.2014.07.003)

Fan, D., Nonaka, A., Almgren, A. S., Harpole, A., & Zingale, M. (2019). MAESTROeX: A massively parallel low mach number astrophysical solver. *Astrophysical Journal*, *887*(2), 212. doi:[10.3847/1538-4357/ab4f75](https://doi.org/10.3847/1538-4357/ab4f75)

Fryxell, B., Olson, K., Ricker, P., Timmes, F. X., Zingale, M., Lamb, D. Q., MacNeice, P., et al. (2000). FLASH: An Adaptive Mesh Hydrodynamics Code for Modeling Astrophysical Thermonuclear Flashes. *Astrophysical Journal Supplement Series*, *131*(1), 273–334. doi:[10.1086/317361](https://doi.org/10.1086/317361)

Grete, P., Glines, F. W., & O'Shea, B. W. (2021). K-Athena: a performance portable structured grid finite volume magnetohydrodynamics code. *IEEE Transactions on Parallel and Distributed Systems*, *32*(1), 85–97. doi:[10.1109/tpds.2020.3010016](https://doi.org/10.1109/tpds.2020.3010016)

StarKiller Microphysics Development Team, Bishop, A., Fields, C. E., Harpole, A., Jacobs, A. M., Katz, M., Li, X., et al. (2020). *Starkiller-astro/microphysics: Microphysics 20.02*. Zenodo. doi:[10.5281/zenodo.3633773](https://doi.org/10.5281/zenodo.3633773)

Zhang, W., Almgren, A., Beckner, V., Bell, J., Blaschke, J., Chan, C., Day, M., et al. (2019). AMReX: A framework for block-structured adaptive mesh refinement. *Journal of Open Source Software*, *4*(37), 1370. doi:[10.21105/joss.01370](https://doi.org/10.21105/joss.01370)

Zhang, W., Howell, L., Almgren, A., Burrows, A., & Bell, J. (2011). CASTRO: A New Compressible Astrophysical Solver. II. Gray Radiation Hydrodynamics. *Astrophysical Journal Supplement Series*, *196*(2), 20. doi:[10.1088/0067-0049/196/2/20](https://doi.org/10.1088/0067-0049/196/2/20)

Zhang, W., Howell, L., Almgren, A., Burrows, A., Dolence, J., & Bell, J. (2013). CASTRO: A New Compressible Astrophysical Solver. III. Multigroup Radiation Hydrodynamics. *Astrophysical Journal Supplement Series*, *204*(1), 7. doi:[10.1088/0067-0049/204/1/7](https://doi.org/10.1088/0067-0049/204/1/7)

Zingale, M., Almgren, A. S., Barrios Sazo, M., Bell, J. B., Eiden, K., Harpole, A., Katz, M. P., et al. (2019). The Castro AMR Simulation Code: Current and Future Developments. *Journal of Physics: Conference Series*, *1623*, 012021. doi:[10.1088/1742-6596/1623/1/012021](https://doi.org/10.1088/1742-6596/1623/1/012021)

Zingale, M., Almgren, A. S., Sazo, M. G. B., Beckner, V. E., Bell, J. B., Friesen, B., Jacobs, A. M., et al. (2018). Meeting the challenges of modeling astrophysical thermonuclear

explosions: Castro, maestro, and the AMReX astrophysics suite. *Journal of Physics: Conference Series*, *1031*, 012024. doi:10.1088/1742-6596/1031/1/012024

Zingale, M., Eiden, K., Cavecchi, Y., Harpole, A., Bell, J. B., Chang, M., Hawke, I., et al. (2019). Toward resolved simulations of burning fronts in thermonuclear x-ray bursts. *Journal of Physics: Conference Series*, *1225*, 012005. doi:10.1088/1742-6596/1225/1/012005

Zingale, M., Katz, M. P., Bell, J. B., Minion, M. L., Nonaka, A. J., & Zhang, W. (2019). Improved Coupling of Hydrodynamics and Nuclear Reactions via Spectral Deferred Corrections. *Astrophysical Journal*, *886*(2), 105. doi:10.3847/1538-4357/ab4e1d