CASTRO I: Hydro, Gravity, Scaling

Ann Almgren

Lawrence Berkeley National Laboratory

SciDAC Computational Astrophysics Consortium Meeting
May 19, 2010

Cast of Characters

CASTRO: Hydrodynamics and Self-Gravity,
A.S. Almgren, V.E. Beckner, J.B. Bell, M.S. Day,
L.H. Howell, C.C. Joggerst, M.J. Lijewski, A.J. Nonaka,
M. Singer, M. Zingale, 2010, ApJ, 709, 11-26.

Monday

- 9:35am – Louis Howell – Radiation Transport in CASTRO
- 4:15pm – Haitao Ma – CASTRO Models for SNe Ia

Tuesday

- 9:00am – Adam Burrows– Core Collapse using CASTRO
- 9:50am – Candace Joggerst – Mixing in Type II Supernovae
- 10:10am – Ken Chen – Pair-Instability Supernovae with CASTRO

What is CASTRO?

CASTRO is a massively parallel code that solves the

- multicomponent compressible hydrodynamic equations

with a

- general equation of state

and includes

- self-gravity,
- nuclear reactions,
- radiation.

CASTRO: The Hydrodynamics

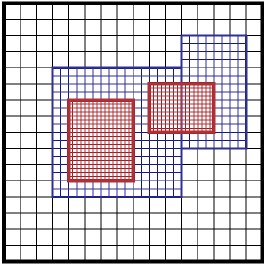
We integrate the compressible hydro equations with reactions, gravity, additional source terms:

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\nabla \cdot (\rho U) + S_{\mathrm{ext},\rho}, \\
\frac{\partial (\rho U)}{\partial t} &= -\nabla \cdot (\rho U U) - \nabla p + \rho g + S_{\mathrm{ext},\rho U}, \\
\frac{\partial (\rho E)}{\partial t} &= -\nabla \cdot (\rho U E + p U) + \rho H_{\mathrm{nuc}} + \rho U \cdot g + S_{\mathrm{ext},\rho E}, \\
\frac{\partial (\rho C_k^{\mathrm{adv}})}{\partial t} &= -\nabla \cdot (\rho U C_k^{\mathrm{adv}}) + S_{\mathrm{ext},\rho C_k^{\mathrm{adv}}}, \\
\frac{\partial (\rho C_k^{\mathrm{aux}})}{\partial t} &= -\nabla \cdot (\rho U C_k^{\mathrm{aux}}) + S_{\mathrm{ext},\rho C_k^{\mathrm{aux}}}.
\end{aligned}
$$

We also use dual energy formulation for e/E.

CASTRO: The Mesh

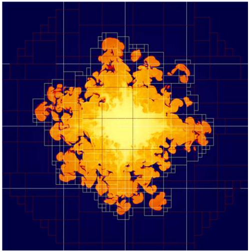
CASTRO uses an Eulerian grid with adaptive mesh refinement (AMR). Our approach to AMR uses a nested hierarchy of logically-rectangular grids with simultaneous refinement of the grids in both space and time.



Coordinate systems:

- 1D: spherical, cylindrical, Cartesian
- 2D: cylindrical, Cartesian
- 3D: Cartesian

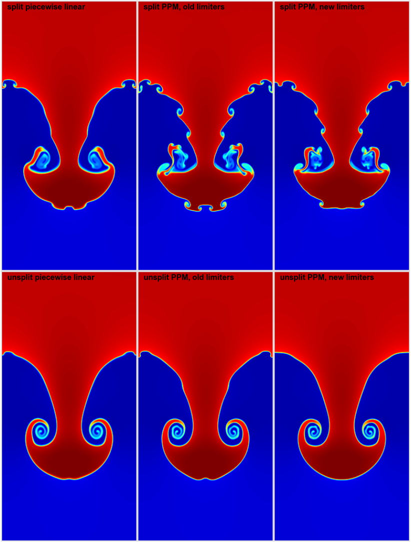
CASTRO: Comparison with Other Codes



Code	Subcycling?	Split / Unsplit	Refinement
CASTRO	Y	Unsplit	patch-based
FLASH	N	Split	patch-based
ENZO	Y	Split	patch-based
RAGE	N	Split	cell-by-cell

All four codes have structured grid AMR, but differ in how they construct and handle their patches

Why use an unsplit integrator?



CASTRO: Gravity

- constant gravity
- multilevel monopole approximation

$$g(r) = GM_{\mathrm{encl}}/r^2$$

- Poisson solve for gravitational potential – complete with multilevel synchronization

$$\nabla^2 \phi = 4\pi G \rho$$

CASTRO: Poisson Gravity with AMR

Because we subcycle in time, we must solve for ϕ on coarse and fine levels independently (unlike FLASH).

If just solving the Poisson equation, $\nabla^2 \phi = 4\pi G \rho$:

- Solve $L^c \phi^c = 4\pi G \rho^c$ on coarse level
- Solve $L^f \phi^f = 4\pi G \rho^f$ on fine level with Dirichlet boundary conditions for ϕ^f
- Synchronize by solving

$$L^{c-f}(\delta\phi) = 4\pi G(\delta\rho) - (\nabla \cdot \delta F_\phi)|^c$$

where δF_ϕ captures the mismatch in $\frac{\partial \phi}{\partial n}$ at the coarse-fine interface.

CASTRO: EOS and Reaction Networks

EOS and reaction networks are included in a modular way (identical routines for MAESTRO and CASTRO):

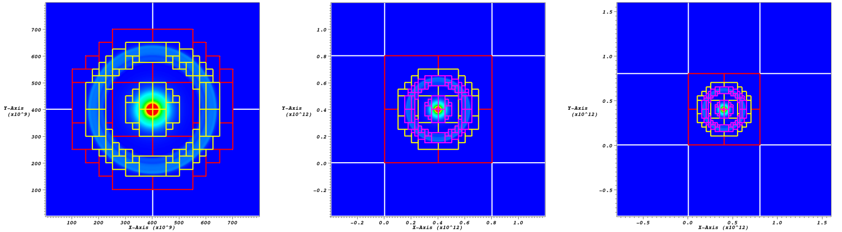
Examples of EOS routines include:

- gamma law gas
- Helmholtz EOS
- Lattimer-Swesty
- user-defined...

Examples of reacion networks include:

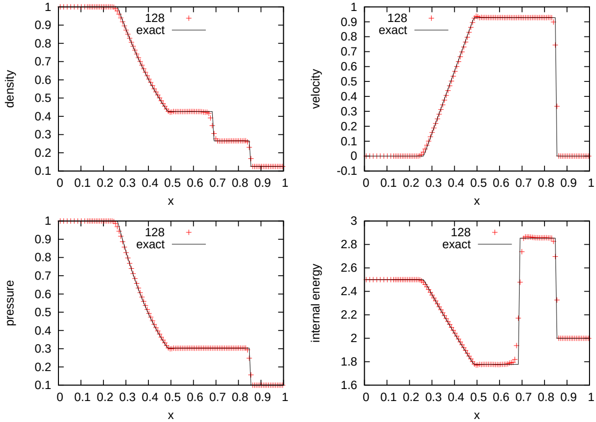
- C12 + C12
- 3-alpha burner
- simple alpha chain + NSE
- user-defined...

There is an option on restart that allows you to grow the domain and coarsen the base grid:



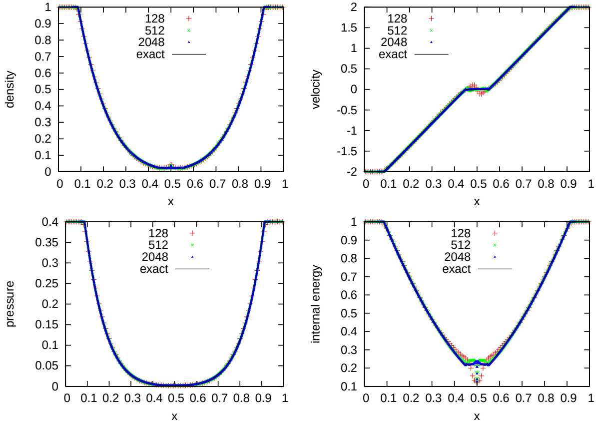
Original domain Doubled Tripled

CASTRO: Validation



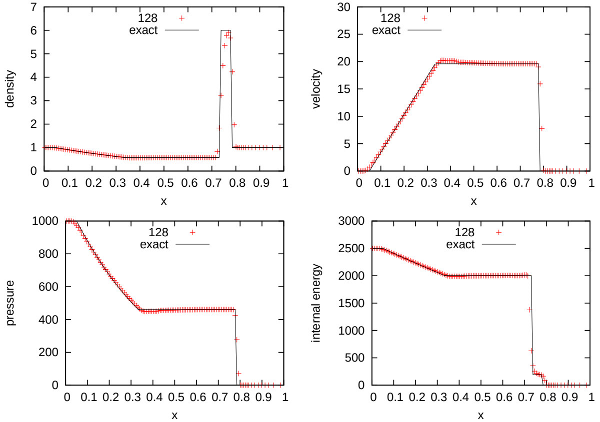
Adaptive CASTRO solution vs. analytic solution for Sod's problem run in 1D at an effective resolution of 128 cells.

CASTRO: Validation



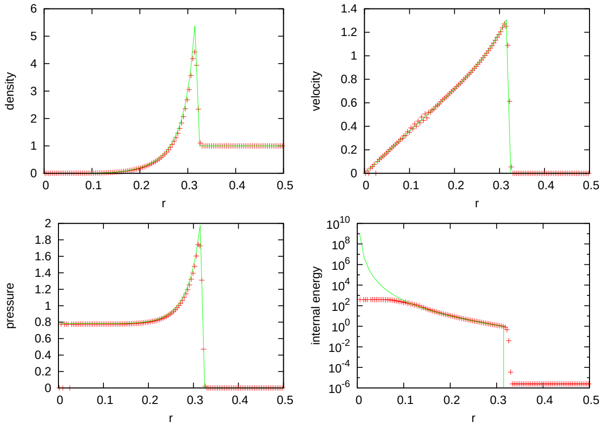
Adaptive CASTRO solution vs. analytic solution for the double rarefaction problem run in 1D at an effective resolution of 128, 512 and 2048 cells.

CASTRO: Validation



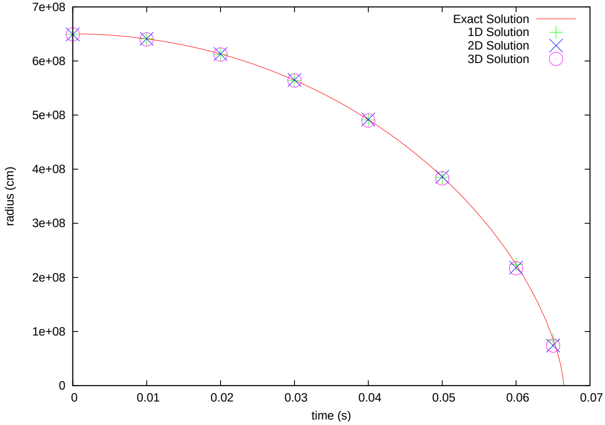
Adaptive CASTRO solution vs. analytic solution for the strong shock problem run in 1D at an effective resolution of 128 cells.

CASTRO: Validation



CASTRO solution at $t = 0.1$s for the cylindrical Sedov blast wave problem run in 2D Cartesian coordinates. This was run with a base grid of $\Delta x = 0.03125$ cm and 3 levels of factor 2 refinement.

CASTRO: Validation



Radius vs. time for the homologous dust collapse problem in 1D, 2D and 3D simulations as compared to the exact solution.

CASTRO: Software Design

Written in combination of C++ and Fortran90

Parallelization achieved by

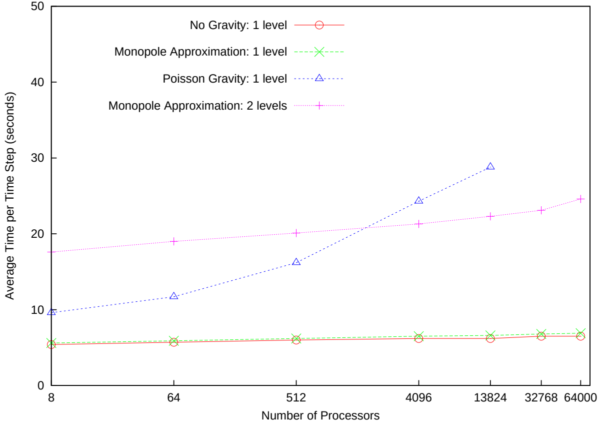
- pure MPI approach – distribute grids to processors and use MPI to communicate between processors
- hybrid approach – distribute grids to nodes, use MPI to communicate between nodes, and use OpenMP to allow multiple processors on a node to work on the same grid

Special attention paid to parallel I/O for checkpoint/restart and plotfiles (CASTRO I/O matches top performance of N5 IOR benchmark on franklin, roughly 13GB/s in idealized problem, roughly 5GB/s sustained)

Visualization: BoxLib format supported by amrvis and Visit (see VACET talk)

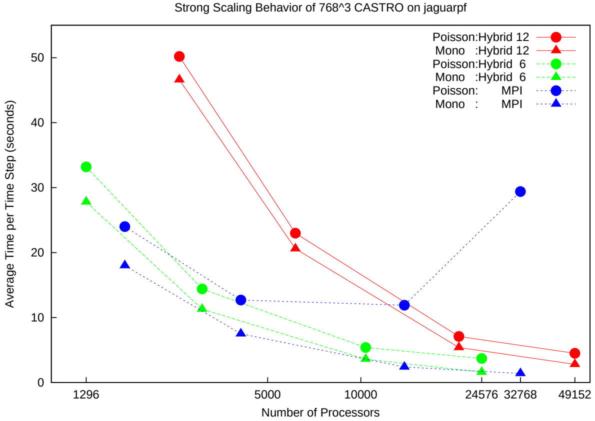
CASTRO: Parallel Scaling

In paper I, we showed weak scaling:



CASTRO: Parallel Scaling

More recent strong scaling tests:



CASTRO: Looking Forward

- See code talks by Howell (radiation), Nonaka (MAESTRO).
- See science talks by Ma, Burrows, Joggerst, Chen
- Subgrid scale turbulence models
- Flame models – level sets, thickened flames...
- Lagrangian tracer particles
- Extend to $O(10^5)$ processors
- GPUs???
- Public release as a community code

If you are interested in using CASTRO, please talk to us.

CASTRO: Future Reference

See our web page:

ccse.lbl.gov/Research/CASTRO

- Paper I: **CASTRO: Hydrodynamics and Self-Gravity**, 2010, ApJ, 709, 11-26.
- User Guide (100+ pages)