AMReX

GPU for Science Day

July 3, 2019
AMReX is the (block-structured) AMR software framework being developed in the Co-Design Center. Originally designed for solution of time-dependent PDEs but is not constrained to PDEs.

Much of the algorithmic methodology embedded in AMReX was developed as part of the DOE Applied Mathematics Program.
AMR appears in multiple applications

Some AMReX ECP Projects
- WarpX: Accelerator design
- PeleLM: Combustion
- FLASH5: Astrophysics
- MFIX-Exa: Multiphase flow
AMReX: Using GPUs

- **New objects** to manage memory efficiently.

- Implement using **GPU streams** to maximize asynchronicity.

- Design for **portability** to CPU and other GPU systems.

- Adopt **finer-grain parallelism** techniques.
Overall Strategy: Put **floating point data** (mesh values, particle data) **on the accelerator** and leave it there. Move as little as possible throughout.

**CPU:** Few slower, generalized threads.

**GPU:** Many faster, specialized threads.

- Solution Control
- Communication
- Load Balancing
- I/O
- Particle Calculations
- Stencil Operations
- Linear Solvers
- And other highly parallelizable algorithms.

And other serial or race-condition prone calculations.
AMReX: Results

Comparing CPU/GPU systems with just CPUs.

- Cell-Centered Linear Solver: 2x
- Castro hydro: 10x at scale
- WarpX: 3x overall increase.

Simulation of Richtmyer–Meshkov instability performed on Summit.

256 x 128 x 128 cells, 1664 steps

Achieved a **30x speed-up** using 2 Volta V100s vs. 2 Power9s.
AMReX: Next Steps

- Improve MPI Communication.
- Port AMReX-based apps.
- Detailed Launch Algorithms.
- Prepare for other GPU architectures (AMD & Intel).
- GPU Roofline Analyses for applications.
- Low-Precision Implementations?
Thank You