Scalable GPU accelerated simulation of multiphase compressible flow

Anand Radhakrishnan, Henry Le Berre, and Spencer H. Bryngelson

GitHub →

Motivation

- Multiphase compressible flow problems ubiquitous
- CPU simulations slow due to large number of time steps
- New supercomputers rely on GPU computation
- GPUs necessary for marked speedup

Multi-Component Flow Code (MFC) [1]

- Diffuse interface model
- High-order accurate via WENO5
- Fortran90, MPI + GPUDirect
- Offload computation to GPUs via OpenACC

Objective

Achieve GPU speedup at scale to conduct large multiphase simulations

Challenges

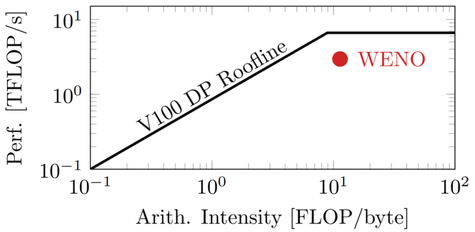
- Finite-volume algorithms usually memory bound, limiting speedup
- MPI communication time potentially significant for GPU case

Metaprogramming techniques

- User inputs passed as fixed constants using a Fortran preprocessor: Fypp
- Enables caching of fixed size private arrays available at compile time
- Elimination of conditional blocks improves kernel occupancy
- 8x- and 2x-speedup of most expensive kernels

Kernel Optimization

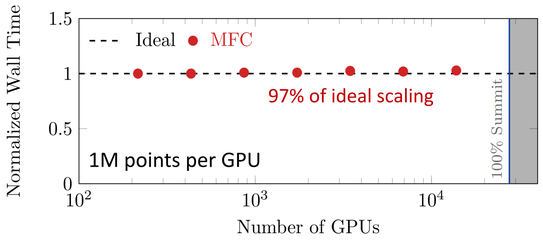
- WENO reconstruction kernel most expensive (40% of time step)
- Nested OpenACC loops collapsed for improved parallelization
- Smaller loops serialized to reduce kernel time
- **46%** of peak FLOPS and high arithmetic intensity (13 FLOPS/byte)



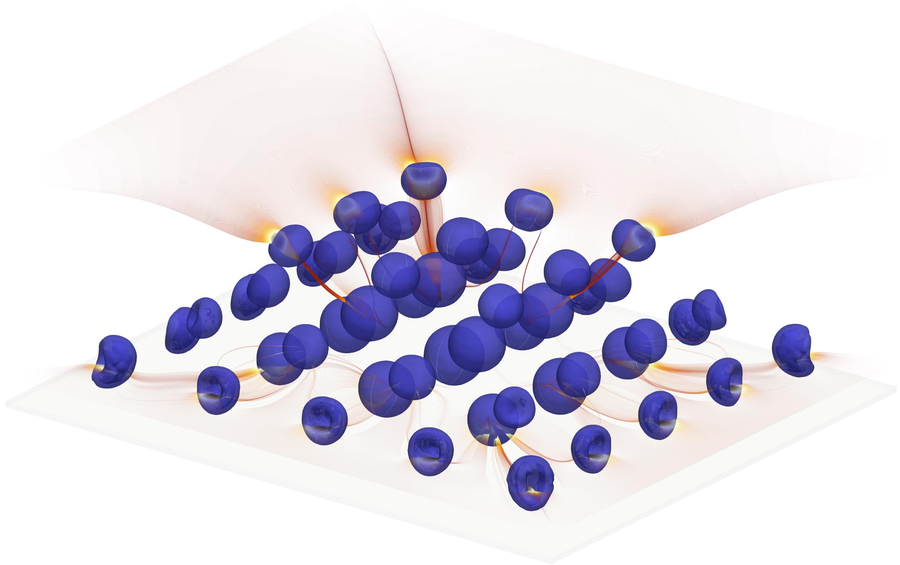
Achieved Speedup

- **300x** speedup on single NVIDIA A100 GPU over an Intel Xeon CPU core (8M grid point problem)
- GPUs **40x faster** than CPUs on Summit node

Weak Scaling



Multi-GPU Simulation

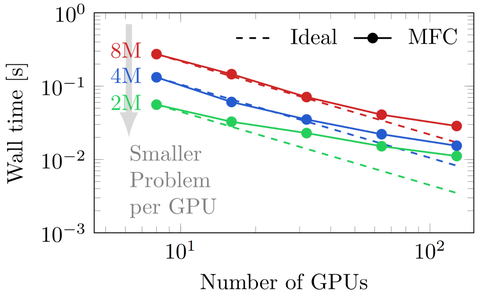


Streamlines of a collapsing bubble cloud near wall

- 600^3 points, 36 Summit nodes (1M pts per GPU)
- 2 hours of wall-clock time for 300K time steps

Strong Scaling

- 4x faster communication using CUDA aware MPI and NVIDIA GPUDirect
- 84% of ideal performance for 400^3 problem from 8 to 64 GPUs



Conclusion

- High compute intensity of expensive kernels enables large GPU speedups
  - **40x** on a Summit node
- Near ideal (**97**%) weak scaling
- Direct GPU–GPU communication enables good strong scaling despite fast kernels

Acknowledgement

We acknowledge use of OLCF Summit and Wombat under allocation CFD154, XSEDE under allocation TG-PHY210084, ONR Grant No. N00014-22-1-2519 (under Dr. Julie Young), and awards from the NVIDIA Academic Hardware Grants program.

[1] Bryngelson et al. (2021); mflowcode.github.io

School of Computational Science and Engineering, Georgia Institute of Technology