Yulin Chen

3rd year PhD student from Combustion Modeling Lab (CML) in Mechanical Engineering in UC Berkeley.

Research interest:

Computational fluid dynamics (CFD) with detailed chemistry as well as reduction of large chemical kinetic mechanism for computationally efficient simulation of complex multidimensional, reactive flows and other engineering systems.

Goal in this class:

Implement CPU and GPU-based parallel schemes to accelerate CFD calculation with detailed chemistry for turbulence-chemistry interaction studies.

Application problem with parallel computing:

The study by Le et. al. [1] describes the implementation of a numerical solver for simulating chemically reacting flow in the Graphic Processing Unit (GPU). The reactive-flow solver is the first to evaluate both the chemical kinetics and fluid transport terms on the GPU. As with most other approaches, the operator splitting is used to decouple and independently solve the chemical kinetics and fluid transport terms. The stiff chemical kinetics terms are solved by a first-order implicit/backward Euler method, using a direct Gaussian elimination to solve the resulting linear system of equations. To implement the algorithms on the GPU, the thread-to-cell approach is applied.

In the chemical kinetics kernel, shared memory is used to store the system of variables as well as the Jacobian matrix (necessary for the Euler time integration), but could only store two rows of the matrix at a time due to the limited size of shared memory. Unfortunately, this causes excessive memory transfer between the global and shared memories, negatively impacting performance. The use of coalesced global memory alone offered better performance.

Compared against an equivalent CPU version executed on a single processor core, their combined GPU solver performed up to 40 times faster by using a reaction mechanism of methane with 36 species and 308 reactions, on a grid with over $10^4$ cells.

CUDA programming model is used for the GPU computing.

However, the relatively low accuracy of the first-order chemistry solver should be noted.

References