NEW HYBRID CPU-GPU SOLVER FOR CFD-DEM SIMULATION OF FLUIDIZED BED

R. Zarghami, School of Chemical Engineering, College of Engineering, University of Tehran, Iran
rzarghami@ut.ac.ir

H. R. Norouzi, N. Mostoufi, School of Chemical Engineering, College of Engineering, University of Tehran, Iran

Modeling is an alternative to experiment to explore more in multiphase flows. Various modeling approaches have been developed and used from 1D models in the macro-scale to multidimensional models in the micro-scale. Well-known modeling approaches for fluidization systems are TFM and CFD-DEM, both have found many practical applications in fluidization systems. The TFM considers both fluid and particulate phase as interpenetrating continua. In contrast, the CFD-DEM considers the fluid as a continuous fluid in the meso-scale (cell-scale) and the solid as discrete particles in the micro-scale. The translational and rotational motions of individual particles are described by applying Newton’s and Euler’s second laws of motion, respectively.

Since the first introduction of CFD-DEM technique by Tsuji et al. (1) and Hoomans et al. (2), different aspects of this modeling approach have been being enhanced and developed. Nowadays, this modeling approach has found many applications in different engineering fields specially fluidization (3). One of the main limitations of this modeling approach is its high computational demand which makes the parallelization necessary in order to model larger systems with more details. A CFD-DEM code comprises of three computational parts: CFD, DEM and coupling. The CFD part can be efficiently parallelized using space decomposition method on the distributed-memory platform like MPI, while the DEM part, due to its low granularity, is better to be parallelized using loop-level parallelization on the shared-memory platform like CUDA.

We used a combination of both platforms to speed-up the CFD-DEM code. Figure 1 shows the data transfer between different parts of the code and the platforms that are used for their implementation. As it can be seen, the CFD and coupling parts are parallelized using MPI and executed on multiple CPUs and the DEM part is parallelized using CUDA and executed on a GPU. To solve the Navier-Stokes equations, we used the open-source CFD package, OpenFOAM®, while the code for coupling and DEM calculations were developed internally. The main goal of this programming style was to benefit from the maximum computational power of CPU and GPU in a single PC equipped with a CUDA-capable GPU. This code was successfully utilized for a fluidization system containing 100,000 spherical particles with the mean size of 2200 micrometers and density of 1500 kg/m3. Particles were placed in a cylinder with inner diameter of 0.14 m and height of 1 m. Number of fluid cells in the simulation was 7,400. The superficial gas velocity was 2 m/s. The code was executed on one CPU core of an Intel® core™-i7 processor (3.6 GHz) and an NVIDIA GeForce® GTX 660 Ti GPU. The simulation was continued for 1 second and the execution time was about 1.5 hr. Snapshots of this simulation are shown in Figure 2. These snapshots show the contour of gas velocity field and particles which are colored based on their velocity. This code is in its very first stages of developments and needs optimizations in both coupling and DEM parts to gain more execution speed.

References
Figure 1: (a) schematic of the main parts of a CFD-DEM code and the iteration loop, (b) different parts of the new solver implemented on CPU and GPU and the data transfer among them.

Figure 2: Results of a test run with new hybrid solver.