Midterm Project Option 2: Optimize Computational Fluid Dynamics Code Running on GPU

Date Due: November 15, 2013 – 23:59 PM
Intermediate Report Due: October 31– 23:59 PM

NOTE: This project can continue and become a Final Project.

In this project, a Computational Fluid Dynamics (CFD) code, Chrono::Fluid, will be provided to you. In turn, you will need to run a performance analysis on the code to pinpoint the bottlenecks and improve its performance.

In Chrono::Fluid the dynamics of the fluid is simulated in a Lagrangian fashion using Smoothed Particle Hydrodynamics (SPH). The basic idea of SPH is to discretize the equations of motion using moving markers. Each marker has a domain of influence in which its properties are disseminated. This is a sphere in 3D or a circle in 2D. To calculate a fluid property or its derivative, each marker needs to access the neighbor markers to account for their influence. For instance, in Figure 1, marker ‘a’ needs to access the properties of markers ‘b’, ‘d’, ‘e’, ‘f’, and ‘g’. All the code is implemented using thrust and several CUDA kernel function. A schematic diagram of the simulation algorithm is provided in Figure 2.

You will have to:

• Profile the code to pinpoint the computation and memory access bottlenecks
• Optimize the memory usage, e.g. by relying on shared memory, improving global memory accesses, reducing data communication, etc.
• Run the provided benchmarks to demonstrate that the code produces the same results after the changes you made to it
• Profile the code after you make changes to understand what change was responsible for what speedup (positive or negative, they are both interesting)
• Report the overall speedup for the provided benchmark
• Run the scaling test, i.e. understand how the run time increase with the domain size and/or number of SPH markers

Remarks:

a) A speedup (compared to original version) of 2X or more guarantees an automatic ‘A’ in 759
b) Replacing thrust vectors with user-managed arrays should be avoided unless it leads to compelling speedups
c) The lab student who put together the code will be available to meet with you as many time as necessary to explain the code in its gory details
d) Details of the SPH formalism will be provided in the form of a technical report
Figure 2: Flow chart of the simulation algorithm