Members

Jasmine Brewer, Samuel Elliott, Matthew Gross, Aniq Shahid, & Li-yin Young

Optimizing Hydrodynamic Simulations of Quantum Fluids

CSCI 4576/5576 Project Checkpoint, Fall 2014

**Introduction**

This project is focused on optimizing a set of hydrodynamic simulations based on the lattice Boltzmann method. We have focused on the 2-dimensional code and have made progress in the areas of single-core optimization, OpenMP, MPI, and data IO. In the following report we briefly discuss our approach and implementation specifics, show performance results, and discuss our outlook and goals for the remainder of the project.

**Progress**

The single-core optimized and OpenMP codes were developed in parallel and are therefore both independent and based on the original code. The MPI code is based on the single-core optimized implementation. We have written a script to check that the data output of all codes discussed herein matches that of the original code to machine precision.

1. **Profiling:** We profiled the original code and found that the execution time was dominated by the time taken to run the Lattice Boltzmann equilibration and streaming step, and we have since focused our optimization efforts on these routines. We were surprised to find, on the other hand, that writing the data to text files took a relatively small percentage of the execution time.
2. **Single-Core Optimization:** We have implemented and benchmarked an optimized single-core version of the code. Primary optimizations included changing 3- and 4-dimensional C++ vectors in the original code to 3- and 4-dimensional arrays and implementing SIMD vectorization on several important loops. We also cleaned and commented the code and made smaller data structure improvements (for example, using bool instead of int types for Boolean values). The scaling performance of the original and single-core optimized codes and the speedup of the single-core optimized code are shown below.



We also developed an optimization which employed 1-dimensional arrays to represent the 3-dimensional and 4-dimensional vectors. For simplicity we reverted to 3-dimensional and 4-dimensional arrays, but we consider 1-dimensional arrays as a possibility for future improvement.

1. **OpenMP:** We have developed and benchmarked an OpenMP implementation of the original code. The speedup and efficiency of our benchmark problem sizes as a function of the number of processors are shown below.



We achieve speedup relative to the single-core implementation for 2 threads and all problem sizes. However, we see our performance deteriorate as we increase the number of threads. We intend to continue to improve the OpenMP implementation through the next phase of the project with the goal of efficiently using all twelve cores on a compute node. Once this is achieved we will integrate the OpenMP implementation into the MPI code.

1. **MPI:** We have begun to develop an MPI version of the single-core optimized code, and so far have succeeded in developing and testing a multi-threaded version of our initialization function using MPI. We have also developed a conceptual framework for how we will implement the next-neighbor lattice Boltzmann streaming with domain decomposition in MPI. In the next phase of the project we will implement these ideas to develop a full MPI version of the code. We will then integrate OpenMP to develop a hybrid MPI/OpenMP code.
2. **HDF5:** We have developed a routine to write the output data to HDF5 files in a serial code. During the next phase of the project, we will integrate this routine into the single-core optimized code and benchmark its affect. We will then develop and test a routine to write HDF5 files in parallel in the MPI implementation.

**Outlook**

As discussed in the previous section, we intend to complete the MPI implementation in the next stage of the project and improve and expand upon the OpenMP and HDF5 implementations so that they can be used reliably and efficiently in the MPI code. We will then extend the hybrid OpenMP/MPI code to take advantage of the architecture on the Phi coprocessors and will perform architecture-based performance comparisons between the Intel Phi and Xeon. Finally, we hope to extend the methods we have used to improve performance in 2 dimensions to optimize our simulations in 3 dimensions. We have included a list of references which we have consulted so far in the course of our project and which we anticipate being useful for the tasks remaining to be completed.

**References**

1. Shiyi, Chen and Gary D. Doolen. “Lattice Boltzmann method for fluid flows.” *Annual Review of Fluid Mechanics*. 1998, 30.
2. Mattila, K, et. al. “Comparison of implementations of the lattice-Boltzmann method.” *Computers and Mathematics with Applications*. Vol. 55 Issue 7, April 2008, 1514-1524.
3. Latt, Jonas. “Hydrodynamic Limit of the Lattice Boltzmann Equations.” *Thesis, University of Geneva*. 2007.
4. Succi, Sauro. “The Lattice Boltzmann Equation for Fluid Dynamics and Beyond.” *Oxford University Press .* 2001.
5. Vidal, David et. al. “A parallel workload balance and memory efficient lattice-Boltzmann algorithm with single unit BGK relaxation time for laminar Newtonian flows.” *Computers and Fluids.* Vol. 39 No. 8, 1411-1423, 2010.
6. Argentini R., Bakker A.F, Lowe C.P. “Efficiently using memory in lattice Boltzmann simulations.” *Future Generation Computer Systems* 2004; 20(6):973-980.
7. Vidal, David et.al. “On improving the performance of large parallel lattice Boltzmann flow simulations in heterogeneous porous media.” *Computers and Fluids*. Vol. 39 No. 2, 324-337, 2010.
8. Wellein, G. et.al. “On the single processor performance of simple lattice Boltzmann kernels.” *Computers and Fluids* 2006; 35:910-919.
9. Satofuka, N. and Nishioka T. “Parallelization of lattice Boltzmann method for incompressible flow computations”. *Computational Mechanics* 1999; 23:164-71.
10. Kandhai D., et.al. “Lattice-Boltzmann hydrodynamics on parallel systems.” *Computer Physics Communications* 1998; 111:14-26.
11. Wang J. et.al. “Domain-decomposition method for parallel lattice Boltzmann simulation of incompressible flow in porous media. *Physical Review E* 2005; 72:016706-11.
12. Schulz M. et.al. “Parallelization strategies and efficiency of CFD computations in complex geometries using lattice Boltzmann methods on high performance computers.” *High performance scientific and engineering computing.* Berlin: Springer Verlag: 2002. 115-122.