The triad of extreme computing— fast algorithms, open software and heterogeneous systems.

GPU Technology Conference, Sept. 20–23, 2010; San Jose Convention Center, San Jose, California

L. A. Barba
Dept. of Mechanical Engineering
Boston University

August 7, 2010

Abstract

In the prominent book by Trefethen and Bau, "Numerical Linear Algebra" (SIAM), we read that *"... the fundamental law of computer science [is]: the faster the computer, the greater the importance of speed of algorithms."*

The rebirth of hardware acceleration in recent years, especially with GPUs, has posed the challenge of rethinking the core algorithms in our scientific applications. Facing this challenge, the first wave of successful GPU accelerations has been crowded with highly-parallel methods that adapted well to the hardware. Perhaps the paradigmatic example is molecular dynamics and other N-body simulations, where the embarrassingly parallel problem of calculating all-pairwise interactions can exploit the fine-grained parallelism of the hardware exceedingly well. But the easy-pickings are now running out. The truly challenging applications usually involve intricate methods that require "going back to the algorithmic drawing board" (to quote an Intel blog) for a successful implementation on the new hardware. And to realize scientific discoveries of extreme scale, not just the hardware, but the algorithms and methods need to be fast. To develop new versions of the most effective fast algorithms, such that our science can most benefit, an ideal environment is created by the open software model, where heroic efforts can at least be shared, and accumulate.

One area of application where we see at work the triad of extreme computing—fast algorithms, open software, and heterogeneous computing—is electrostatics of macromolecules in solution. Here, one popular model is the continuum approximation, or implicit-solvent model. A vast acceleration is possible in this application via fast algorithms for long-range interactions, in particular the fast multipole method, FMM. The open-source library PetFMM provides an efficient and scalable parallel version of the FMM, with GPU capability recently added for the main kernels. This new, GPU-enabled version of PetFMM is being applied to the computation of macromolecular electrostatics, and demonstrating that the combined speedup of fast algorithms, efficient parallel software, and hardware acceleration will enable unprecedented speed in this field. We aim this work at enabling routine calculations of all charge interactions in molecules in seconds on commodity hardware. In the open-source model, a GPU-enabled PetFMM library can allow computational chemists to benefit fully from these advances.

Acknowledgements. The work in Barba's group is currently done in collaboration with Dr Jaydeep Bardhan, Dr Tsuyoshi Hamada, Dr Matthew Knepley, and Dr Rio Yokota, and with the hard work of graduate students Christopher Cooper, Felipe A Cruz, Anush Krishnan, and Simon K Layton.