

NEWS CENTER :: FEATURE STORIES

May 2008

Code Booster

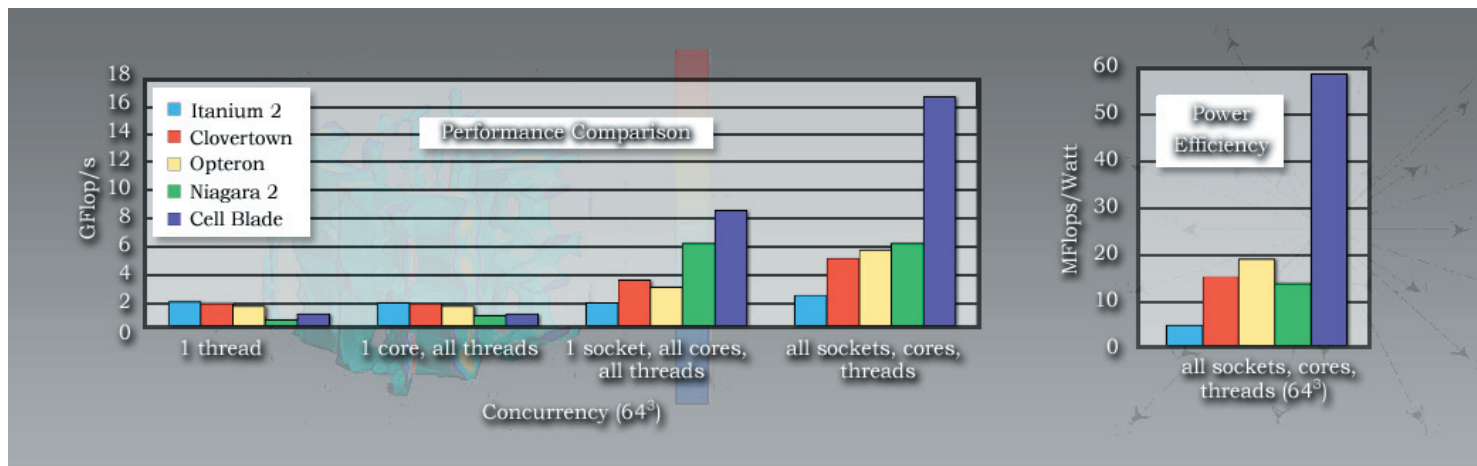
Award-winning research on code optimization explores multicore computing

Contact: Uclia Wang, uwang@lbl.gov

At the 2008 IEEE International Parallel and Distributed Processing Symposium (IPDPS) held in Miami, the award for Best Paper in the “applications” category went to a research paper exploring ways to make a popular scientific analysis code run smoothly on different types of multicore computers.

Samuel Williams, a researcher from Berkeley Lab’s Computational Research Division (CRD), was lead author of the award-winning paper, titled “Lattice Boltzmann simulation optimization on leading multicore platforms.” Williams and his collaborators chose to focus on lattice Boltzmann code as a way to explore a broader issue: how to make the best use of multicore supercomputers.

The multicore trend in supercomputing started only recently, and the computing industry is expected to add more cores per chip to boost performance in the future. Unfortunately the trend is taking flight without an equally concerted effort by software developers, says Williams. In the winning paper he writes, “The computing revolution towards massive on-chip parallelism is moving forward with relatively little concrete evidence on how to best to use these technologies for real applications.”



Using the lattice Boltzmann magnetohydrodynamics (LBMHD) problem as their testbed, the Berkeley Lab researchers compared runtime performance (left) and power efficiency (right) for a variety of multicore architectures.

Williams and his colleagues settled on the lattice Boltzmann code that is used to model turbulence in magnetohydrodynamics simulations, which play a key role in several areas of physics research, from star formation to magnetic fusion devices. Unfortunately the code, LBMHD (LB for lattice Boltzmann, MHD for magnetohydrodynamics), typically performs poorly on traditional multicore machines. In their paper, the researchers describe how they developed a code generator that could efficiently and productively optimize a lattice Boltzmann code to deliver better performance on a new breed of supercomputers built with multicore processors.

The optimization research performed by the authors resulted in a great improvement to LBMHD code performance—substantially higher than any published to date. The researchers also gained insight into building effective multicore applications, compilers, and other tools.

Jonathan Carter of NERSC, Lenny Olikier of CRD, John Shalf of NERSC, and Kathy Yelick of NERSC were coauthors of the paper, which was presented at the IPDPS in Miami in April, 2008, and won Best Paper Award in the application track. Yelick, who is NERSC Director, was a keynote speaker at the symposium.

continued

Oliker, Carter, and Shalf were authors of a paper that won the same award last year, “Scientific application performance on candidate petascale platforms,” coauthored by CRD researchers Andrew Canning, Costin Iancu, Michael Lijewski, Shoaib Kamil, Hongzhang Shan, and Erich Strohmaier. Stephane Ethier from the Princeton Plasma Physics Laboratory and Tom Goodale from Louisiana State University also contributed to the work.

In their recent work on LBMHD, the researchers determined how well the code runs on several processors used to build computers today: Intel’s quad-core Clovertown, Advanced Micro Devices’ dual-core Opteron X2, Sun Microsystems’ eight-core Niagara 2, and the eight-core STI Cell Blade (designed by Sony, Toshiba, and IBM). They also looked at Intel’s single-core Itanium 2, to compare its more complex single-core design with other, simpler multicores.

The researchers first looked at why the original LBMHD performs poorly on these multicore systems. Williams and his fellow researchers found that, contrary to conventional wisdom, memory-bus bandwidth didn’t present the biggest obstacle. Instead, lack of resources for mapping virtual memory pages, insufficient cache bandwidth, high memory latency, and/or poor functional unit scheduling did more to hamper the code’s performance.

The researchers created a code generator abstraction for LBMHD, in order to optimize it for different multicore architectures. The optimization efforts included loop restructuring, code reordering, software prefetching, and explicit “SIMDization” (single-instruction, multiple-data vectorization). The researchers characterized their effort as akin to the “autotuning methodology exemplified by libraries like ATLAS and OSKI,” means of automatically searching for the best codes to address specific problems.

Their results showed a wide range of performances on different processors and pointed to bottlenecks in the hardware that prevented the code from running well. The optimization efforts also resulted in a huge gain in performance—the speed of the optimized code ran up to 14 times faster than the original version. It also achieved sustained performance for this code higher than any published to date: on two of the processor architectures, over 50 percent of peak flops (floating point operations per second).

Compared with other processors, the Cell processor provided the highest raw performance and power efficiency for LBMHD. This processor’s design calls for a direct software control of the data movement between on-chip and main memory, resulting in the impressive performance. Overall, the researchers concluded, processor designs that focused on high throughput using sustainable memory bandwidth and a large number of simple cores perform better than processors with complex cores that emphasized sequential performance.

They also concluded that autotuning would be an important tool for ensuring that numerical simulation codes will perform well on future multicore computers.

Additional information

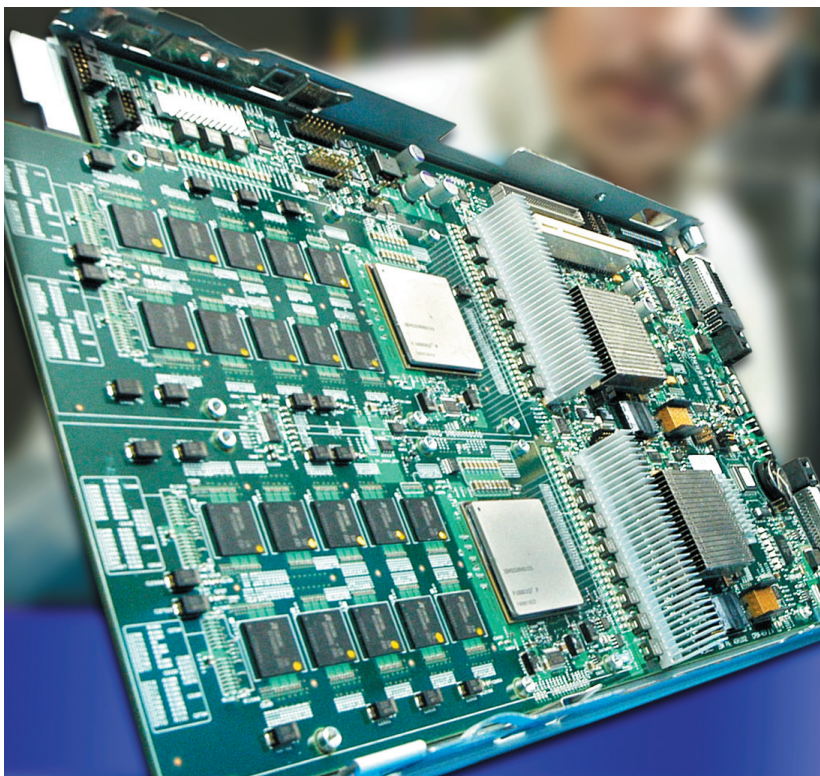
“Lattice Boltzmann simulation optimization on leading multicore platforms,” by Samuel Williams, Jonathan Carter, Leonid Oliker, John Shalf, and Katherine Yelick, was selected as Best Paper in the Application Track at IPDPS 2008. It may be found at http://crd.lbl.gov/~oliker/papers/ipdps08_final.pdf.

“Scientific application performance on candidate petascale platforms,” by Leonid Oliker, Andrew Canning, Jonathan Carter, Costin Iancu, Michael Lijewski, Shoaib Kamil, John Shalf, Hongzhang Shan, Erich Strohmaier, Stephane Ethier, and Tom Goodale, was winner of the Best Paper at IPDPS 2007. It may be found at <http://crd.lbl.gov/~oliker/papers/ipdps07.pdf>.

More about IPDPS, the IEEE International Parallel and Distributed Processing Symposium, is at <http://www.ipdps.org/>.

Visit Samuel Williams’s website at <http://www.cs.berkeley.edu/~samw/>.

For more about autotuning, see <http://cscads.rice.edu/workshops/july2007/autotune-slides-07/Yelick-Overview.pdf>.



Of the processors studied using the LBMHD problem, the Cell Blade yielded the highest raw performance and power efficiency.