Optimizing the Performance and Energy of LU Decomposition on a Heterogeneous Multicore System with Dynamic Power Scheduling

Brian Hunter, University of Wisconsin - Madison
Professor Rong Ge, Marquette University, Department of Mathematics, Statistics, and Computer Science

Introduction

Computer applications need to run with maximal performance and minimal energy consumption. This is a critical issue in high performance computing where speed is increased at the expense of consuming large quantities of energy. For continued progress in this field, it is essential to develop methods which will allow for an emphasis on fast computation while maintaining low energy usage.

LU decomposition is a computationally intense and common linear algebra function which factors a matrix into two triangular matrices.

Methods

- Utilize LU decomposition programs from the MAGMA and MORSE libraries.
- Measure performance of combined CPU and GPU computing.
- Observe the effects of DVFS (dynamic voltage frequency scaling) on CPUs and GPUs using metrics of energy consumed, power draw, and billions of operations per second (GFlops).
- Compare metrics of LU decomposition from CPU only and combined CPU and GPU.
- Use the collected data to form a dynamic scheduling approach which will use lower frequencies during times of low computation to save energy.

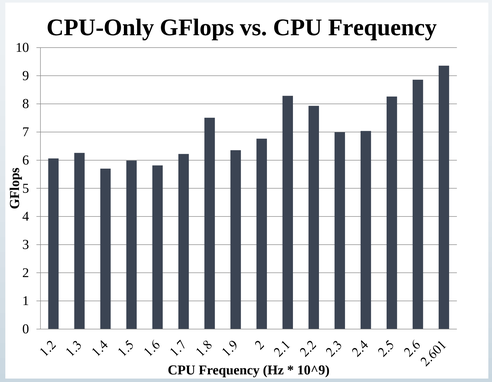
Platform

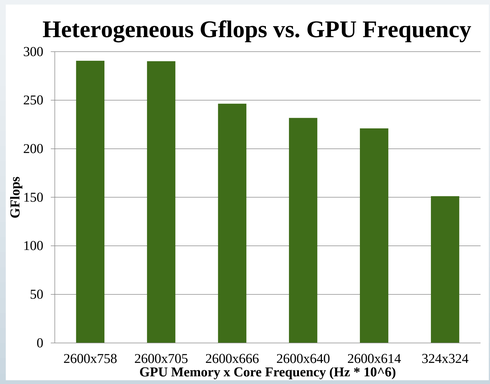
All research is done on a heterogeneous multicore system. The system consists of 16 Xeon SandyBridge E5-2670 cores, each of which has a default speed of 2.6 GHz, and 2 Tesla K20c GPUs which have 4799.6 MB of local memory, a default memory speed of 2.6 GHz, and a core speed of 705 MHz. All computing resources are capable of DVFS, which allowes them to change core speeds.

References

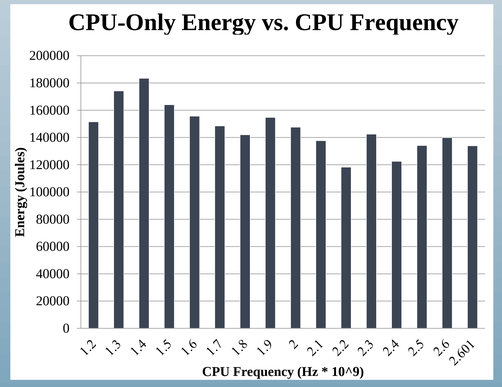
- [1] Kim, Hyesoon, Chi-Keung Luk, and Sunpyo Hong. *Qilin: Exploiting Parallelism on Heterogeneous Multiprocessors with Adaptive Mapping*. Tech. no. TR-2009- 001. Atlanta, GA: Georgia Institute of Technology, 2009.
- [2] Liu, Qiang, and Wayne Luk. "Heterogeneous Systems for Energy Efficient Scientific Computing." *International Conference on Reconfigurable Computing*. Hong Kong, China. N.p.: Springer, 2012. 64-75. *EPICS*. Web. 25 June 2013.
- [3] Simon, Horst. *Why We Need Exascale and Why We Won't Get There by 2020*. Proc. of Optical Interconnects Conference, Santa Fe, New Mexico, Berkeley, CA: Berkeley Lab, 2013. 1-27. Print.

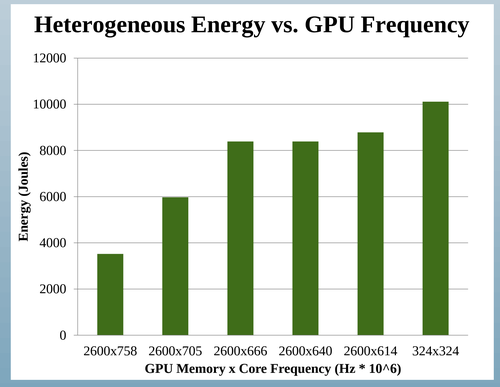
Data Collected



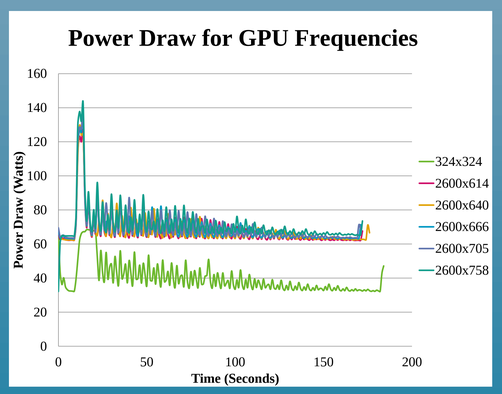


The heterogeneous approach is significantly faster than the CPU-only approach at all frequencies. Both obtained maximum performance at the highest setting.

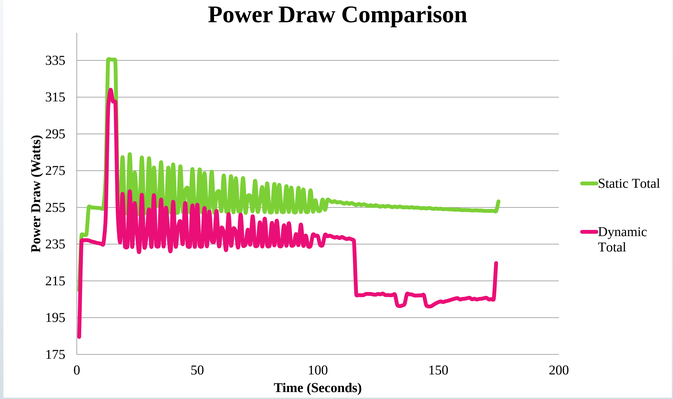




The heterogeneous approach was most energy efficient at the maximum frequency, 758 MHz, while the CPU was most efficient at 2.2 GHz.



The slower performance from 324 MHz comes from the first 2/3 of the runtime.



Findings

- Running the GPU at 758 MHz has better performance than 705, 666, 640, and 614 MHz, with almost identical power draw.
- LU decomposition has a long period at the end with a low computational demand.
- The computation time for the end period is nearly equal for all frequencies.
- A heterogeneous approach is the fastest and most efficient.

Results and Discussion

- Using a heterogeneous approach gives as much as 30x faster performance and uses as little as 1/15 of the energy as CPU-only.
- There is little penalty in terms of energy consumption to have the GPU at 758 MHz over 705, 666, 640, and 614 MHz.
- At times of low computation, a GPU speed of 324 MHz has comparable performance to 758 MHz with a lower power draw.
- The heterogeneous algorithm uses 1 of 16 cores.
- Incorporating these results into dynamic scheduling gives 11.3% less energy at no cost to performance.
- Using DVFS to scale down GPU frequency has a more significant effect on energy savings than scaling down the CPU cores.

Conclusion and Future Work

- Using a heterogeneous approach to computing offers a solution to obtain high performance while being energy efficient.
- Dynamic scheduling can be used to further manage energy consumption by utilizing DVFS to use low power frequencies during periods of low computation.
- The use of dynamic scheduling and DVFS needs to be implemented in other programs.
- DVFS only allows for a few ranges of frequencies. A larger range would allow for better utilization of computing resources.