Introduction

I propose a final project on benchmarking GPUs for compute workloads. The specific GPU families I wish to evaluate are the AMD Radeon HD 69xx series and the NVIDIA Fermi series. If time and resources permit, I would also like to evaluate their predecessors, the AMD Radeon HD 58xx and the NVIDIA GT 2xx series.

There are a few reasons why I want to benchmark these two families. One is that they are the latest commercially available consumer GPUs. The AMD Radeon HD 69xx series in particular only launched in December 2010, so there are few rigorous examinations of its compute performance. As NVIDIA Fermi is the gold standard in GPU compute support, it will be interesting to juxtapose the two GPU families in performance and will make this work unique.

Another reason is that the architecture of these two GPU families is very different. While Fermi relies entirely on thread-level parallelism (TLP) for extracting performance, the Radeon HD 69xx series also relies on instruction-level parallelism (ILP) with its VLIW4 architecture. Each SP-equivalent has 4 ALUs, and the job of scheduling instructions for these ALUs is handled by the compiler, and not by the hardware. With this architecture, the Radeon HD 6970 can advertise 1536 "stream processors," versus a GTX 580's 512. Fermi also has hardware-managed L1 and L2 caches, while the Radeon HD 69xx only has read-only L1 and L2 texture caches. I believe that with a strong OpenCL compiler and a workload with extremely high arithmetic intensity and ILP, a Radeon HD 69xx could outperform Fermi. Results of this competition would be very interesting.

A final reason is that with the 69xx series, AMD made a major architectural shift from VLIW5 to VLIW4, getting rid of the special function unit in their "SP"s in order to fit more "SM"s on die. Since VLIW performance is so dependent on the compiler, though, it will be interesting to see the effects of AMD's software tuning by comparing the 69xx series with its predecessor.

Prior work

- Danalis et al. created "The Scalable HeterOgeneous Computing (SHOC) Benchmark Suite" at Oak Ridge National Laboratory. They test both performance and power consumption for both NVIDIA (up to GT 2xx) and AMD (HD 58xx) GPUs and multicore CPUs using both OpenCL and CUDA benchmarks for a variety of workloads.
- Du et al. evaluate both CUDA and OpenCL performance for the NVIDIA Tesla C2050 and the AMD HD5870 for a matrix-multiply workload. They also analyze the code generation for both CUDA and OpenCL compilers.

Goals (in order of completion)

1. Choose and implement a GPGPU workload with high arithmetic intensity and potential ILP in OpenCL. Run an initial benchmark.
2. Tune a version for Fermi's architecture, and another version for the HD 69xx's architecture. Run another benchmark.
3. Choose and implement a GPGPU workload with lower arithmetic intensity in OpenCL. Run an initial benchmark.
4. Tune this new kernel for Fermi's architecture, and again for the HD 69xx's architecture. Run another benchmark. (**Baseline**)
5. Develop a configurable benchmark with a GUI frontend. The user can choose with a slider widget the level of arithmetic intensity the benchmark should have.
6. Tune (if necessary) for the HD58xx series for both benchmarks, and compare results with the HD69xx's results.
7. Develop a CUDA 3.2 implementation of both benchmarks and compare to previous results.
8. Develop a CUDA 4.0 RC implementation of both benchmarks and compare to previous results.