L15: Design Review and CUBLAS Paper Discussion

Administrative

- Bill Dally (Chief Scientist, NVIDIA and Stanford)
  - Monday, April 6, 11-12, WEB 3760
  - "Stream Programming: Parallel Processing Made Simple"
  - Arrive early
- Design Reviews, starting April 8 and 10
  - Volunteers for April 8
  - Volunteers for April 10
- Final Reports on projects
  - Poster session the week of April 27 with dry run the previous week
  - Also, submit written document and software
  - Invite your friends! I'll invite faculty, NVIDIA, graduate students, application owners, ..

Design Reviews

- Goal is to see a solid plan for each project and make sure projects are on track
  - Plan to evolve project so that results guaranteed
  - Show at least one thing is working
  - How work is being divided among team members
- Major suggestions from proposals
  - Project complexity – break it down into smaller chunks with evolutionary strategy
  - Add references – what has been done before? Known algorithm? GPU implementation?
  - In some cases, claim no communication but it seems needed to me

Design Reviews

- Oral, 10-minute Q&A session
  - Each team member presents one part
  - Team should identify "lead" to present plan
- Three major parts:
  - I. Overview
  - Define computation and high-level mapping to GPU
  - II. Project Plan
  - The pieces and who is doing what.
  - What is done so far? (Make sure something is working by the design review)
  - III. Related Work
  - Prior sequential or parallel algorithms/implementations
  - Prior GPU implementations (or similar computations)
- Submit slides and written document revising proposal that covers these and cleans up anything missing from proposal.

Publishing your projects?

- I would like to see a few projects from this class be published, perhaps in workshops
  - I am willing to help with writing and positioning
- Publishing the work may require additional effort beyond course requirements or timetable of semester
  - So not appropriate for everyone, and certainly not part of your grade in course
- Let's look at some examples (also consider for related work)

Places to look for examples

- NVIDIA CUDA Zone
  - Huge list of research projects using CUDA with speedups ranging from 1.3x to 420x
  - Many of your projects are related to projects listed there
  - http://www.nvidia.com/cuda
- GPGPU
  - http://www.gpgpu.org
  - Links to workshops, research groups, and news from industry
- Some recent workshops
  - SIAM CSE'09: Scientific Computing on Emerging Many-Core Architectures, http://people.maths.ox.ac.uk/~gilesm/SIAM_CSE/index.html
  - WORKSHOP on GPU Supercomputing 2009, National Taiwan University, http://cqse.ntu.edu.tw/cqse/gpu2009.html
  - Workshop on General-Purpose Computation on Graphics Processing Units, http://www.ece.neu.edu/groups/nucar/GPGPU/

Places to look for examples, cont.

- Upcoming calls
  - PPAM (Parallel Processing and Applied Mathematics): due 4/10, also in Poland…
  - Symposium on Application Accelerators in High Performance Computing (SAAHPC'09), http://www.saahpc.org/, 2-3 page abstracts due 4/20
  - Probably, some new calls over the summer
  - Also, application workshops and conferences

Today's Lecture

- Presenting "Benchmarking GPUs to Tune Dense Linear Algebra", Vasily Volkov and James W. Demmel, Proceedings of SC08, November, 2008.

Winner of SC08 Best Paper Award.

A MUST READ FOR THIS CLASS!!!

Paper: (in ACM Digital Library)

http://portal.acm.org/citation.cfm?id=1413402

Slides:

http://www.eecs.berkeley.edu/~volkov/volkov08-sc08talk.pdf

Paper Highlights

- Use short vectors, maximize usage of registers, limit usage of shared memory
- Global synchronization across blocks using atomic operations, made efficient (I'll probe further on this)
- Discovered a number of performance limitations and architectural features
  - There's a TLB. Who knew!
- Exceeds performance of CUBLAS 1.0 by 60% and runs at close to peak of hardware
- Uses decuda to figure out what is happening in code generation.
  - A third party disassembler of GPU binaries based on reverse engineering of ISA

A Few Details not in SC08 Presentation

- Latencies
  - Launch overhead of 3-7 micro-seconds (asynchronous) or 10-14 micro-seconds (synchronous)
- Effective memory bandwidth
  - Time = 11micro-seconds (o/h) + #bytes/3.3GB/s
- Talks about L1 and L2 cache (texture cache) and TLB
- Measurements derived via microbenchmarking
  - L1:
    - 20-way set associative L1s, with 5KB, 8 of them
    - Latency of 280 cycles for a hit (designed for increased bw rather than minimizing latency)
  - L2:
    - 24-way set associative L2s, with 32KB, 6 of them
  - TLB:
    - 16-entry, fully associative TLB