Coprocessors and Accelerators

Philippe Canal

Annual Concurrency Meeting - FNAL - 2013

Challenges and

- How to mitigate latencies (due to data transfer)
  - use of pipe-lining, cuda-streams, etc.
- Criteria to decide which computation should be done on the GPU or on the CPU
- Data structures design
- Optimization strategies
- Performance gains and expectations, total cost of ownership
- Risk of vendor lock-in

Geant4 on GPU

- Gain 12 to 34 for Runga-Kutta,
- float faster but in the case of G4 degrade the precision too much.
- Implemented (part of) 3 EM Physics - no secondary production yet which is the next challenge.
- Importance of the memory layout and access.
- CUDA stream worked as expected - nicely to reduce memory upload/download latency.
- Change for CUDA leads to performance improvement in CPU code itself.

CMS on GPU

- OpenCL -> write once, run everywhere.
- More CPU include on board GPU that are idle if not explicitly used.
- Ported multiple scattering algorithm.
- With OpenCL needed 10,000 tracks to amortize data-transfer and scheduling.
- OpenCL on CPU as performant as regular code.
- OpenCL being behind feature-wise is compensated by the portability.
- OpenMP scaling for " for loop " parallelization.

NA62 on GPU

- Use CUDA stream but in real-time environment need to mitigate the problem of unknown latency.
- PCIe 2 to 3 does double the bandwidth.
- In 99.9% of the case latency stability is good.
- Ready for production.
- Wants to investigate different kind of synchronization mechanism.

Minuit on GPU

- Parallelizing the function evaluation within Minuit
- Uses the Thrust Library (so far the only backend).
- High speed up but require coding in Cuda for PDF.
- Lack of fine-grained profiling makes it hard to track down bottlenecks in execution.

Review

- CUDA: allow maximal performance.
  - Stream important to hide latency, are they available in OpenCL?
- OpenCL: allow cross platform development.
- GPU proven to be quite cost competitive.
  - As bandwidth increase capacity increase also, so latency stay an issue for the next generation.
- Change for CUDA leads to performance improvement in CPU code itself.