Multi-Threaded Algorithms for GPGPUs in the ATLAS High Level Trigger

Patricia Conde Muínó on behalf of the ATLAS Collaboration
Higher luminosity $\rightarrow$ larger pile-up $\rightarrow$ larger volume of data

Advanced algorithms needed to provide the same rejection

Run 3

- 2.4 MB/25 ns
- 240 GB/s
- 29 GB/s
- 2.4 GB/s
General Purpose GPUs for triggering

- HLT farm size is limited, mainly by power and cooling
- CPU time increases with pile-up
  Dominated by combinatorial nature of the tracking algorithms
- GPGPUs: provide massive parallelization potential

**ATLAS GPGPU prototype**

- Evaluate the use of GPGPUs at trigger level
  Figure of merit: processed events/s/cost

**Algorithms:**
  Calorimeter, tracking, muon and jet reconstruction

**Hardware: NVidia Tesla K80**
  12 GB RAM
  13 multi-processor
  2496 CUDA Cores per chip
  824 MHz GPU, 2505 MHz memory clock
Trigger GPGPU demonstrator architecture

Client-server architecture:

- **Client:**
  - One HLT processing unit per core
  - Athena offline & online framework
  - Provides data
  - Executes chains of algorithms
  - Provides monitoring services
  - Rejects/accepts the events

- **Server:**
  - Independent from Athena
  - Accelerator resource management
  - Serve many Athena processes
  - Can exploit several technologies
  - Pre-allocate memory for data
  - Store global/constant data
GPU Inner Detector algorithms

➢ Spacepoints (SP) arranged in wedge-shaped slices
➢ For each SP in middle layer
  Search for inner/outer SP in same and neighbouring slices
  Impose z-limits from beam spot
➢ GPU kernels
  Calculate all doublets with middle-SPs
  Store doublets in global SoA storage
  Form triplets
  Apply kinematic and quality cuts
GPGPU ID Algorithm Performance

➢ GPGPU algorithm provides same efficiency and resolution as CPU one (very different GPU and CPU algorithms)

Link to GPU trigger public plots
GPGPU ID Algorithm Timing Performance

- Total algorithm execution time
  Measured in tt-bar events (very busy) with 46 interactions/bunch crossing
  Reduced by a factor ~5
- Very small data transfer overhead

![Diagram showing relative contributions of different processes]

- Athena 72%
  - CPU track seeding 27.7%
  - Time per event 1.6 s

- Athena 92.4%
  - GPU track seeding 7.6%
  - Time per event 1.2 s

- Doublet making 1.7%
- Triplet making 2.6%
- Counting 2.2%
- Data conversion and transfer 0.6%
GPGPU Calorimeter Clustering Algorithm

- TopoCluster reconstruction on CPU (~8% of total time)
  Group cells in 3-dimensions according to their signal/noise ratio

- TAC: Topo-Automaton Clustering
  Use a cellular automaton for the GPU (maximize parallelism)
  Propagate tag on a grid of elements (cell pair)
  Cells get the largest tag on each iteration
  Process all cells pairs until no tag changes

Legend:
- Seed ($S/N>4$)
- Growing ($S/N>2$)
- Terminal ($S/N>0$)
- Not enough S/N
- Not evaluated
GPGPU Calorimeter Clustering Performance

- Energy difference <5% for most clusters
- Cluster growing time reduction factor:
  
  | Sample   | Pile-up | Reduction factor |
  |----------|---------|------------------|
  | tt-bar   | 138     | 2                |
  | tt-bar   | 46      | 2                |
  | di-jets  | 40      | 1.3              |

- Potential larger gain with parallelization of next clustering steps (splitting)

Calorimeter clustering on CPU

- Time per event 1.02 s
- CPU clustering 84 ms, 8.2%
- Athena 91.6%

Calorimeter clustering on GPU

- Time per event 1.06 s
- GPU clustering 44 ms, 4.3%
- Athena 95.6%
- Growing 1.8%
- IPC 0.7%

Data conversion 1%
- Conversion (CPU to GPU) 0.2%
- Data Transfer 0.1%
- Classification 0.05%
- Growing 1.8%
- Tagging

GPU

Conversion (GPU to CPU) 1.0%
- IPC 0.7%
- Data transfer 0.1%

Sample

Chep'16, San Francisco, 10-14 Oct 16
Muon reconstruction algorithms

- Uses Hough Transform (HT) to convert track finding to maxima finding
  - Straight HT is used for xy-plane, curved HT in rz-plane
- GPU algorithm
  - Filter hits and fill Hough-space matrices
  - Sort maxima above certain threshold
  - Associate hits in spectrometer
- 3D segments constructed by combining 2D segments in CPU
Gain in throughput: 20-40% depending on number of processes running

1 GPU can serve efficiently up to 14 processes

Gains will increase when more code is offloaded (ex. Bytestream conversion, track following, cluster splitting, …)

Jet reconstruction algorithms already implemented on GPU - performance measurements underway
Summary and conclusions

➢ The LHC Upgrade will impose stringent requirements on the ATLAS trigger system
   Need advanced algorithms, capable of higher rejection with same efficiency
➢ ATLAS is studying the use of GPGPUs for triggering
   Require re-implementation of the algorithms to maximize parallelization
➢ First evaluation of calorimeter and tracking reconstruction
   Achieved the same physics performance in tracking & cluster reconstruction
   Total execution time reduced by a maximum of
   A factor of 5 for tracking
   A factor of 2 for cluster formation
   Lesson: data structures suitable for CPU & GPU would reduce overheads
➢ Gain in number of processed events/s:
   Between 20-40%, depending on number of processes accessing the GPU
   Larger gain expected when more code is offloaded to the GPU
Acknowledgements

➢ OE, FCT-Portugal, CERN/FIS-NUC/0005/2015
Backup