Evaluation of GPUs for the CMS track trigger: What is possible so far and where are we going?

H. Mohr, T. Dritschler, T. Schuh

Contact: timo.dritschler@kit.edu

CERN’s CMS detector produces unmanageably large amounts of raw data with each single collision event. With the upcoming high luminosity upgrade, the number of collision events will increase even further. With these changes, the amount of data transmitted from the first-level electronics is in the range of 100 terabits per second, with collisions every 25ns.

Analyzing and processing such huge amounts of data require efficient data reduction mechanisms. One option is to only trigger the readout and storage of collision data when certain events occur. The process of real-time reconstruction and analysis of particle trajectories is called track-triggering. Normally this requires expensive hardware components such as ASICs and FPGAs.

GPUs provide high parallel-computing capability with large amounts of memory and efficient floating-point operations. With the ever increasing performance of GPUs and interconnects, it might become feasible to use general purpose GPU computing as a cheaper, more flexible solution for track-triggering.

Requirements
- Data-transmission, processing and reply must take no longer than 6µs combined.
- The strict limitations on turnaround-time make pipelining difficult. This results in larger work-packages per GPU.
- Based on those data distribution limitations, each GPU would have to handle roughly 150Gb/s.

GPU RDMA:
- Conventional data transfer from Network or external devices requires many copy operations.
- RDMA (Remote Direct Memory Access) allows direct access to GPU memory from network or other external devices\(^1\)
- This decreases transfer-latency significantly!

Hough-Transformation
- Transform particle impact in a detector layers into a set of possible trajectory-parameters
- These trajectory-parameters are accumulated into a ‘hough-map’
- Each detected impact contributes one line to this map
- Intersections of multiple lines give a ‘track-candidate’ for further investigation
- Currently, we are able to process 500 hits per GPU in 13µs

In the future
- Moores Law predicts that chip performance doubles roughly every 18 months. Extrapolating our current 13µs based on this assumption, would put us in the range of ~3µs by the year 2020.
- Network speed of the InfiniBand interconnect is expected to reach 200Gb/s with its HDRx4 specification by the year 2018\(^3\)
- The upcoming PCIe Gen. 4 specification is expected to provide 250Gb/s at x16

Challenges
- GPU interconnects need to provide high enough throughput to transport all necessary data
- Network latency needs to be low enough to meet timing constraints
- Algorithm needs to be efficient and highly parallelizable

Requirements
- GPU interconnects need to provide high enough throughput to transport all necessary data
- Network latency needs to be low enough to meet timing constraints
- Algorithm needs to be efficient and highly parallelizable

Algorithm runtime
- 13µs
- 5µs
- 6µs
- 1µs

Interconnect Throughput
- 150Gb/s
- 100Gb/s
- 50Gb/s

In the future
- Moores Law predicts that chip performance doubles roughly every 18 months. Extrapolating our current 13µs based on this assumption, would put us in the range of ~3µs by the year 2020.
- Network speed of the InfiniBand interconnect is expected to reach 200Gb/s with its HDRx4 specification by the year 2018\(^3\)
- The upcoming PCIe Gen. 4 specification is expected to provide 250Gb/s at x16

Notes:
1) Technically, this data copy would also go through the CPU, but it would make the graphic too confusing and was therefore not visualized.
3) Based on the "InfiniBand Roadmap" of the InfiniBand Trade Association. (http://www.infinibandta.org)