Real time data analysis using GPU for High energy physics experiments

V. Singhal[1],* P. P. Bhaduri[1], S. Chattopadhyay[1], and S. K. Aggarwal[2]
[1] *Variable Energy Cyclotron Centre, 1/AF Bidhan Nagar, Kolkata-700 064, INDIA and*
[2] *Department of Computer Science and Engineering, Indian Institute of Technology Kanpur, Kanpur - 208 016, INDIA*

Introduction and motivation

The use of the Graphical Processing Unit (GPU) for general purpose computing is becoming popular and now a days known as the General Purpose GPU (GPGPU). The GPU consists of a set of multiprocessors designed to obtain the best performance by parallel processing with graphics computing. In high energy physics (HEP) experiments, one has to analyse billions of similar events for obtaining physics results, which makes the HEP computing embarrassingly parallel for workload. We have the event selection process for Muon Chamber (MUCH) detector system at the Compressed Baryonic Matter (CBM) experiment is to find out J/ψ from μ+ and μ-. A high event rate (10^7 events/second) is a challenge and storing all these events by data acquisition systems may not be feasible. This scenario motivate us to use GPU or Compute Unified Device Architecture [CUDA] [1] in which we can execute thousands of parallel threads and reduce the data volume by storing only relevant events. In this paper we have modified the existing event selection algorithm [2] and implemented on NVIDIA Tesla C2075 GPU Card [3]. We show that using GPU Tesla C2075 card, desired goal of processing 10^7 events per second for event selection of MUCH system can be achieved.

Event Selection Algorithm and implementation in CUDA

In this work the algorithm selects only those events which are useful with respect to physics of muon chamber. As event rate is extremely high and J/ψ production is very low, therefore we have to store only those events which are likely to contain J/ψ. For achieving the goal we have modified the event selection algorithm [2] on TESLA C2075 Card such that it will serve the purpose of executing high event rate. The algorithm flowchart is shown in Fig. 1.

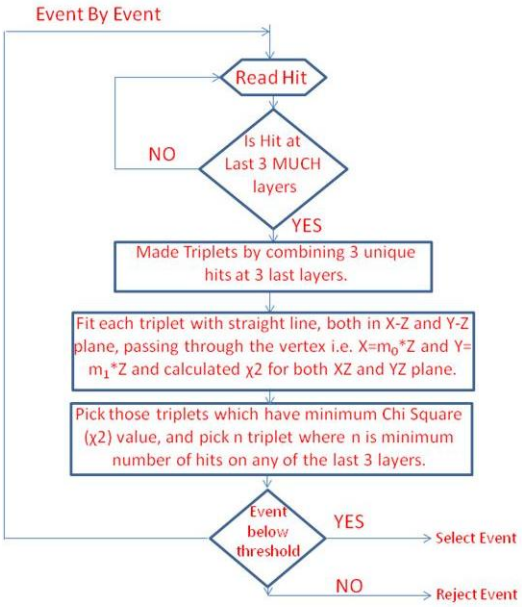

FIG. 1: Flowchart for Event Selection algorithm

First we have written GPU program using the X, Y, Z coordinates array of the hits, looped over 3 times for finding relevant triplet (Opt-1). Optimized by using coalesced memory access according to CUDA architecture (Opt-2). Optimized on the global reads which took significant execution time (Opt-3). Algorithm is implemented in 'C' language using CUDA API and then compiled with nvcc.

*Electronic address: vikas@vecc.gov.in

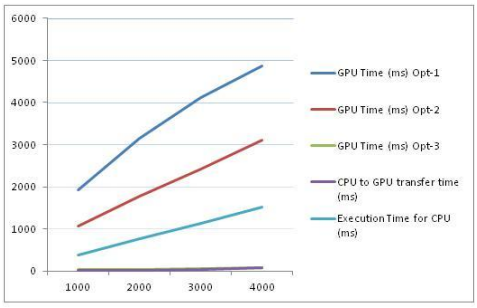

FIG. 2: Time comparisons between different optimization on GPU

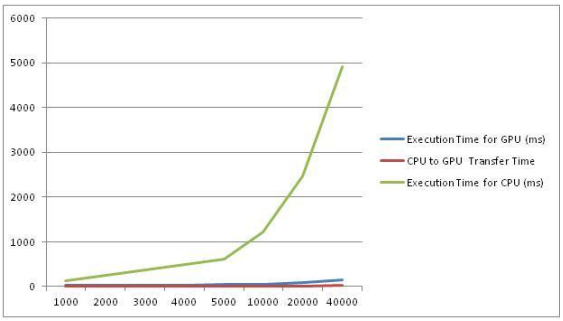

FIG. 3: GPU CPU execution time comparison

Fig. 2 shows execution time comparison between optimizations (Opt-1, Opt-2 and Opt-3) on GPU and CPU.

Final event selection program

After the optimization of GPU code, computation time reduced significantly but we analysed that CPU to GPU data transfer is contributing substantial time, therefore we analysed the entire process and reduced the data volume to transfer from CPU to GPU. Table I describes final result for event selection algorithm on GPU with showing negligible amount of CPU to GPU transfer time.

TABLE I: Final results for Event selection algorithm.

# Event	# block	# thread	GPU Time (ms)	CPU GPU Transfer Time	CPU Time (ms)	Speed Up
1000	32	32	30	1	120	4
2000	64	32	30	1	250	8
3000	64	64	30	10	370	12
4000	64	64	30	10	490	18
5000	128	64	40	10	610	15
10000	128	128	50	10	1230	25
20000	256	128	80	10	2470	31
40000	512	128	140	20	4920	35

As per Fig. 3 for processing 40k events, we achieved 35 times speedup on GPU compared to CPU. Also CPU to GPU data transfer time is negligible compared to computation time. With these results we show that by using one GPU Tesla C2075 card we can process 3* 10^5 events per second for event selection algorithm for MUCH.

Conclusion and Future Scope

In this paper we presented implementation of event selection algorithm for MUCH detector at CBM experiment based on GPU on the simulated data of 35A GeV Au+Au UrQMD central events as background and Pluto events as signal. The algorithm suppresses significant amount of background events without reducing the J/ψ reconstruction efficiency. Table I shows result that we can process 3*10^5 - 4*10^5 events/second using one GPU Card. Present hardware supports upto 4 GPU card on a single motherboard, so we can process more than 10^6 events/seconds which are very near to targeted event rate of 10^7 events/second. GPU/CPU ratio is a matter of research. In conclusion GPU can be beneficial for doing online computation or real time data analysis for HEP experiments. CBM is going to introduce time as fourth dimension along with X, Y, Z. Our next step is to develop and implement 4-dimensional event reconstruction algorithms on GPU.

References

[1] NVIDIA Cuda, www.nvidia.com/cuda
[2] DAE Symp. on Nucl. Phys. 55 (2010) page no. 640-641.
[3] www.nvidia.com/object/workstation-solutions-tesla.html