



Contribution ID: 300

Type: **Poster presentation**

Parallelization of particle transport with INTEL TBB

Monday, 14 October 2013 15:00 (45 minutes)

One of the current problems in HEP computing is the development of particle propagation algorithms capable of efficient work at parallel architectures. An interesting approach in this direction has been recently introduced by the GEANT5 group at CERN [1]. Our report will be devoted to realization of similar functionality using Intel Threading Building Blocks (TBB) library.

In the prototype implemented by the GEANT5 group the independent part of job to be processed by a single computing node is transport of a group of particles located in the same volume. Such a group of particles is called a basket. Processing of a single basket by a single core allows to exploit most efficiently the principle of data locality. The implementation is done using traditional pthreads library. This tool requires manual handling of jobs' mapping onto the threads and hence a special manager-thread code has been written. This manager always occupies one thread and is a potential bottleneck causing low scalability of the code.

Intel TBB is a library of parallel programming templates which is in some sense an extension of STL. Along with standard thread-programming features like locks, mutexes and thread-safe concurrent containers, Intel TBB states a new standard in concurrent programming - task-based programming. The idea of task-based programming is to allow the developer to write tasks and to send them to the task scheduler. The last will automatically solve the core load balancing problem. Intel TBB is considered as a candidate for being a standard tool in various domains of HEP computations.

We are going to present at the CHEP2013 conference an adaptation of the basket oriented track propagation algorithm to the possibilities and immanent logic of TBB. Our prototype is based on an interplay between two types of tasks: PropagationTask which contains job taking care of particle transport within a basket, the result of a PropagationTask is a collection of tracks located on the boundaries of the volume; and DispatcherTask into which the job distributing one or more collections of tracks between several new baskets is packed.

The basic implementation starts a PropagationTask for each basket produced by the DispatcherTask and a DispatcherTask for a group of track collections produced by the PropagationTasks. A number of dispatching policies is under investigation. Intel TBB concurrent queues are used for data exchange between the tasks.

Performance and scalability tests for the basic algorithm with different dispatching policies will be presented. Results will be compared with original GEANT 5 implementation. Possible improvements and further development towards compatibility with vectorized basket processing will be discussed.

The authors acknowledge stimulating and instructive discussions with Federico Carminati and Andrei Gheata (CERN).

[1] Rethinking particle transport in the many-core era towards GEANT 5
Apostolakis, John; Brun, Rene; Carminati, Federico; Gheata, Andrei
J. Phys.: Conf. Ser. 396 (2012) 022014

Summary

Primary authors: OVCHARENKO, Egor (ITEP Institute for Theoretical and Experimental Physics (RU)); BELOGUROV, Sergey (ITEP Institute for Theoretical and Experimental Physics (RU))

Presenter: OVCHARENKO, Egor (ITEP Institute for Theoretical and Experimental Physics (RU))

Session Classification: Poster presentations

Track Classification: Software Engineering, Parallelism & Multi-Core