

Hybrid implementation of the Vegas Monte-Carlo algorithm

Wednesday, 10 September 2014 12:35 (30 minutes)

Multidimensional integration based on Monte-Carlo (MC) techniques are widely used in High Energy Physics (HEP) and numerous other computing domains. In HEP, they naturally arise from the multidimensional probability densities or from the likelihoods often present in the analysis.

Today HPC programming requires dealing with computing accelerators like GPGPU or 'many-core' processors, but also taking into account the development portability and the hardware heterogeneities with the use of open programming standard like OpenCL.

Among several MC possible algorithms (MISER, Markov Chain, etc.) the choice has been driven by the popularity and the efficiency of the method. The 'Vegas' algorithm is frequently used in the LHC analysis as it is accessible from ROOT environment and provides reasonably good performance.

The parallel implementation of Vegas for computing accelerators presents no major obstacle, however some technical difficulties occur when dealing with portability and heterogeneity mainly due to the lack of libraries and development tools (like performance analysis tools).

Combining MPI and OpenCL, we will present a scalable distributed implementation. Performance will be shown on different platforms (Nvidia K20, Intel Xeon Phi) but also on heterogeneous platform mixing CPUs, and different kind of computing accelerator cards.

The presented work is a canvas to integrate various multidimensional functions for different analysis processes. It is planned to integrate and exploit this implementation in the future CMS analysis processes.

Primary author: Mr GRASSEAU, Gilles (Laboratoire Leprince-Ringuet (IN2P3/CNRS))

Co-authors: Mr CHAMONT, David (Laboratoire Leprince-Ringuet (IN2P3/CNRS)); Mr LISNIAK, Stanislav (Laboratoire Leprince-Ringuet (IN2P3/CNRS))

Presenter: Mr GRASSEAU, Gilles (Laboratoire Leprince-Ringuet (IN2P3/CNRS))

Session Classification: GPU in Offline, Montecarlo and Analysis (1/3)