Celeritas—a nascent GPU detector simulation code

T.M. Evans *[1], S.R. Johnson[1], V.D. Elvira[2], P. Calafiura[3], Ph. Canal[2], K.L. Genser[2], S.Y. Jun[2], G. Lima[2], A. Lund[4], V.R. Pascuzzi[3], and S.C. Tognini[1]

[1]Oak Ridge National Laboratory
[2]Fermi National Accelerator Laboratory
[3]Lawrence Berkeley National Laboratory
[4]Argonne National Laboratory

August 31, 2020

Objective

Upgrades to the Large Hadron Collider (LHC) and its detectors (including CMS, ALICE, and ATLAS) demand a commensurate increase in modeling and simulation capacity that is out of reach of traditional software but can be alleviated through the use of advanced high performance computing (HPC) hardware that use GPUs for improved performance at low power consumption. Our objective is to provide the needed modeling capacity using a new application *Celeritas* that performs fast and accurate Monte Carlo (MC) particle transport simulations on GPUs. Celeritas is designed to complement, not replace, Geant4 and ultimately satisfy the detector response requirements as defined in *A Roadmap for HEP Software and Computing R&D for the 2020s* [1] using the advanced architectures that will form the backbone of HPC over the next decade.

Background

The current state-of-the-art MC application for simulating the passage of particles through matter is Geant4, a toolkit that encompasses the simulation of all interactions described by the Standard Model of Particle Physics: electromagnetic (EM), weak, and strong interactions [2]. A drawback to Geant4 for LHC modeling is the very long run times required to perform simulations. One pathway to address this limitation is to utilize advanced accelerator (GPU) architectures that are now available at DOE leadership computing facilities on systems such as Summit at the Oak Ridge Leadership Computing Facility (OLCF). These architectures are becoming increasingly available on institutional-scale computing clusters as well. However, porting and optimizing MC algorithms to work efficiently on GPU architectures is a difficult task, particularly in CPU-only codes such as Geant4 that rely on extensive runtime polymorphism, random walks and branching. GPU coding environments are highly sensitive to memory access patterns, device occupancy, and thread divergence. Furthermore, some common C++ language idioms that are heavily used in Geant4 transport and physics routines, including inheritance and dynamic memory allocation, cannot be used in an efficient manner in device code.

Recent work in the *ExaSMR: Coupled Monte Carlo Neutronics and Fluid Flow Simulation of Small Modular Reactors* project within the DOE Exascale Computing Project (ECP) [3] has demonstrated significant performance for neutron MC transport on GPU architectures [4]. These architectural advancements are implemented in the Oak Ridge National Laboratory (ORNL) MC application Shift [5]. Shift has achieved greater than 60× speed up per-node on Summit versus running on CPU cores alone. However, there are

*Corresponding author, evanstm@ornl.gov

several distinctions between that work and the necessary capabilities required for particle physics detector modeling. The reactor and nuclear technology applications that algorithm targets are not characterized by large showers of secondary particles, and because the particles are neutral, there are no EM field interactions.

Approach

Whereas Geant4 is a fully general toolkit capable of addressing many modeling and simulation transport problems, Celeritas is a highly specialized *application* focused on the most computationally intensive calculations currently performed in experimental detector simulations: time- and energy-dependent detector response. Celeritas is designed to complement, not replace, the simulation capabilities and workflows currently using Geant4.

The current work focuses on the development of a transport code for EM interactions of photons and charged leptons. This particular choice as a first step towards full Standard Model MC transport is due to simplicity and usefulness. Strong interactions present high-stakes challenges to implement, and weak interactions do not just encompass a vast list of particles with different branching ratios but also entail following the produced hadrons that depend on strong interaction processes in order to achieve a direct comparison with Geant4. Conversely, EM interactions can be limited to ionization energy loss, bremsstrahlung, pair production, Cherenkov radiation, and photonuclear interactions. As for usefulness, a proof-of-concept should provide a clear form of comparison between Celeritas and Geant4. In this scenario, leptons such as electrons and muons constitute perfect probes, as these particles are widely used for detector calibration purposes [6, 7]. Finally, since processing EM interactions takes the most runtime in a simulation due to the cardinality of particles, accelerating EM physics provides the greatest opportunity for immediate performance gains.

In LHC detector response modeling applications, the desired outputs are time- and particle-dependent energy depositions in user-identified cells (sensitive detectors) correlated to each generating event (e.g., proton-proton collision). The basic neutral particle GPU transport algorithm developed within ExaSMR must be extended to treat continuous processes such as multiple scattering, generation of massive secondary particle showers, as well as tracking in the EM field. To ensure that progress can be made towards a fully featured application that meets these transport requirements, we must address these potential bottlenecks in a systematic way early in the GPU transport algorithm development.

As part of the GeantV and ECP Geant exascale pilot projects, the navigation (geometry tracking) capabilities of Geant4 have been reengineered and improved into a stand-alone library called *VecGeom* [8]. Since VecGeom is written to support both C++/CPU and CUDA/GPU targets and is able to read `GDML` geometry definitions, it provides a production geometry capability for transport on GPUs.

Outlook

We have presented Celeritas, an application for demonstrating MC EM particle physics simulations on GPUs. It leverages architectural techniques and algorithms from the ECP project *ExaSMR* and models from the Geant4 detector simulation toolkit. The experiences gained through the ECP *ExaSMR* project provide a proof-of-principle that MC transport applications can realize significant performance improvements on GPUs. Charged-particle and EM physics transport simulations require additional algorithmic changes due to the presence of large secondary particle showers and modified transport paths. We are addressing these challenges within Celeritas and have confidence that these complexities can be solved in a manner that does not impact overall computational performance.

While the current focus on Celeritas is on EM physics, we intend to expand Celeritas capabilities to include weak and strong physics. Celeritas is planned to complement Geant4 by providing a pathway for the acceleration of a subset of charged particle transport modeling use cases on HPC architectures. The long term goal is to re-incorporate these methodologies back into the Geant framework for broader application use.

References

[1] The HEP Software Foundation, J. Albrecht, A. A. Alves, *et al.*, "A Roadmap for HEP Software and Computing R&D for the 2020s," en, *Computing and Software for Big Science*, vol. 3, no. 1, p. 7, Dec. 2019, ISSN: 2510-2036, 2510-2044. DOI: 10.1007/s41781-018-0018-8. [Online]. Available: http://link.springer.com/10.1007/s41781-018-0018-8 (visited on 10/01/2019).

[2] S. Agostinelli, J. Allison, K. Amako, *et al.*, "Geant4—a simulation toolkit," en, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 506, no. 3, pp. 250–303, Jul. 2003, ISSN: 01689002. DOI: 10.1016/S0168-9002(03)01368-8. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0168900203013688 (visited on 03/23/2018).

[3] *The Exascale Computing Project, Addressing a national imperative: Application Development Update*, Oak Ridge, TN, Sep. 2019. [Online]. Available: www.exascaleproject.org.

[4] S. P. Hamilton and T. M. Evans, "Continuous-energy Monte Carlo neutron transport on GPUs in the Shift code," en, *Annals of Nuclear Energy*, vol. 128, pp. 236–247, Jun. 2019, ISSN: 03064549. DOI: 10.1016/j.anucene.2019.01.012. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0306454919300167 (visited on 01/17/2019).

[5] T. M. Pandya, S. R. Johnson, T. M. Evans, *et al.*, "Implementation, capabilities, and benchmarking of Shift, a massively parallel Monte Carlo radiation transport code," en, *Journal of Computational Physics*, vol. 308, pp. 239–272, Mar. 2016, ISSN: 00219991. DOI: 10.1016/j.jcp.2015.12.037. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0021999115008566 (visited on 06/19/2017).

[6] The ATLAS Collaboration, G. Aad, B. Abbott, *et al.*, "The ATLAS collaboration," *Electron and photon energy calibration with the ATLAS detector using LHC Run 1 data*, vol. 74, p. 3071, 2014.

[7] The ATLAS Collaboration, "Resolution of the ATLAS muon spectrometer monitored drift tubes in LHC run 2," *Journal of Instrumentation*, vol. 14, no. 09, P09011–P09011, Sep. 2019, ISSN: 1748-0221. DOI: 10.1088/1748-0221/14/09/p09011. [Online]. Available: http://dx.doi.org/10.1088/1748-0221/14/09/P09011.

[8] J. Apostolakis, M. Bandieramonte, G. Bitzes, *et al.*, "Towards a high performance geometry library for particle-detector simulations," en, *Journal of Physics: Conference Series*, vol. 608, p. 012 023, May 2015, ISSN: 1742-6588, 1742-6596. DOI: 10.1088/1742-6596/608/1/012023. [Online]. Available: http://stacks.iop.org/1742-6596/608/i=1/a=012023?key=crossref.bdd3f6ed96e42e9beb181cc658a48d4f (visited on 12/06/2019).