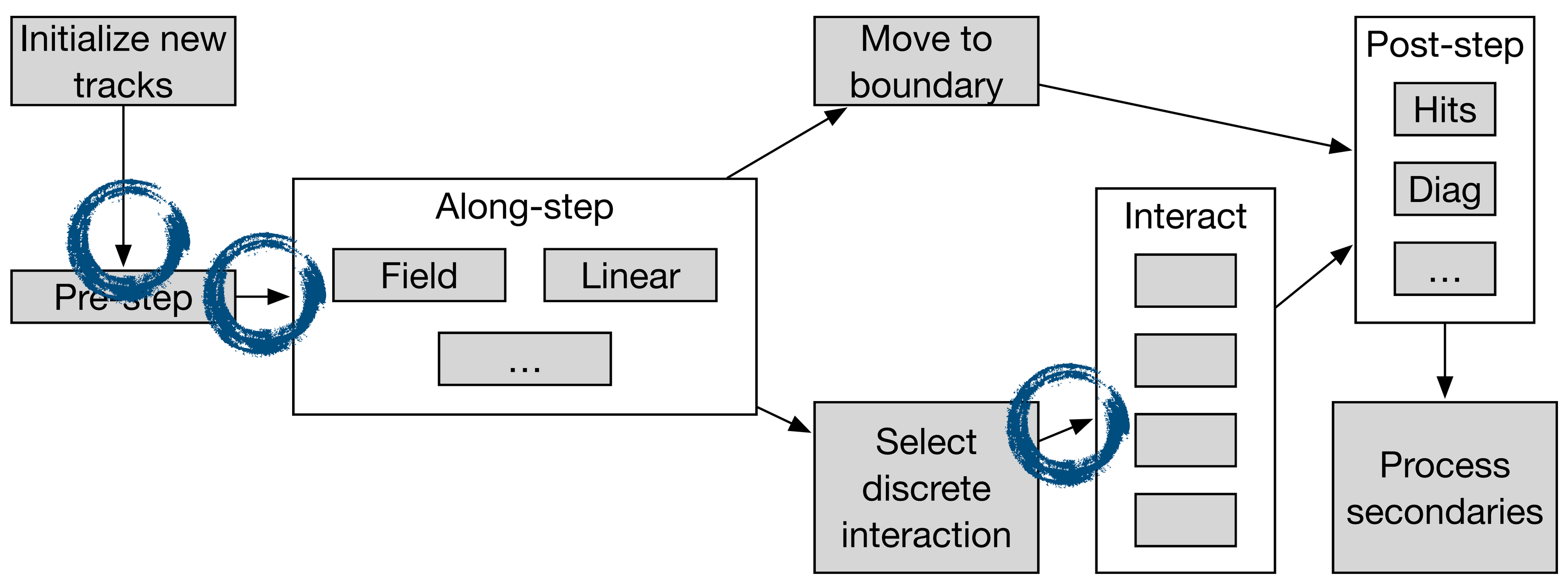
Celeritas: Evaluating Performance of HEP Detector Simulation on GPUs

1. Celeritas

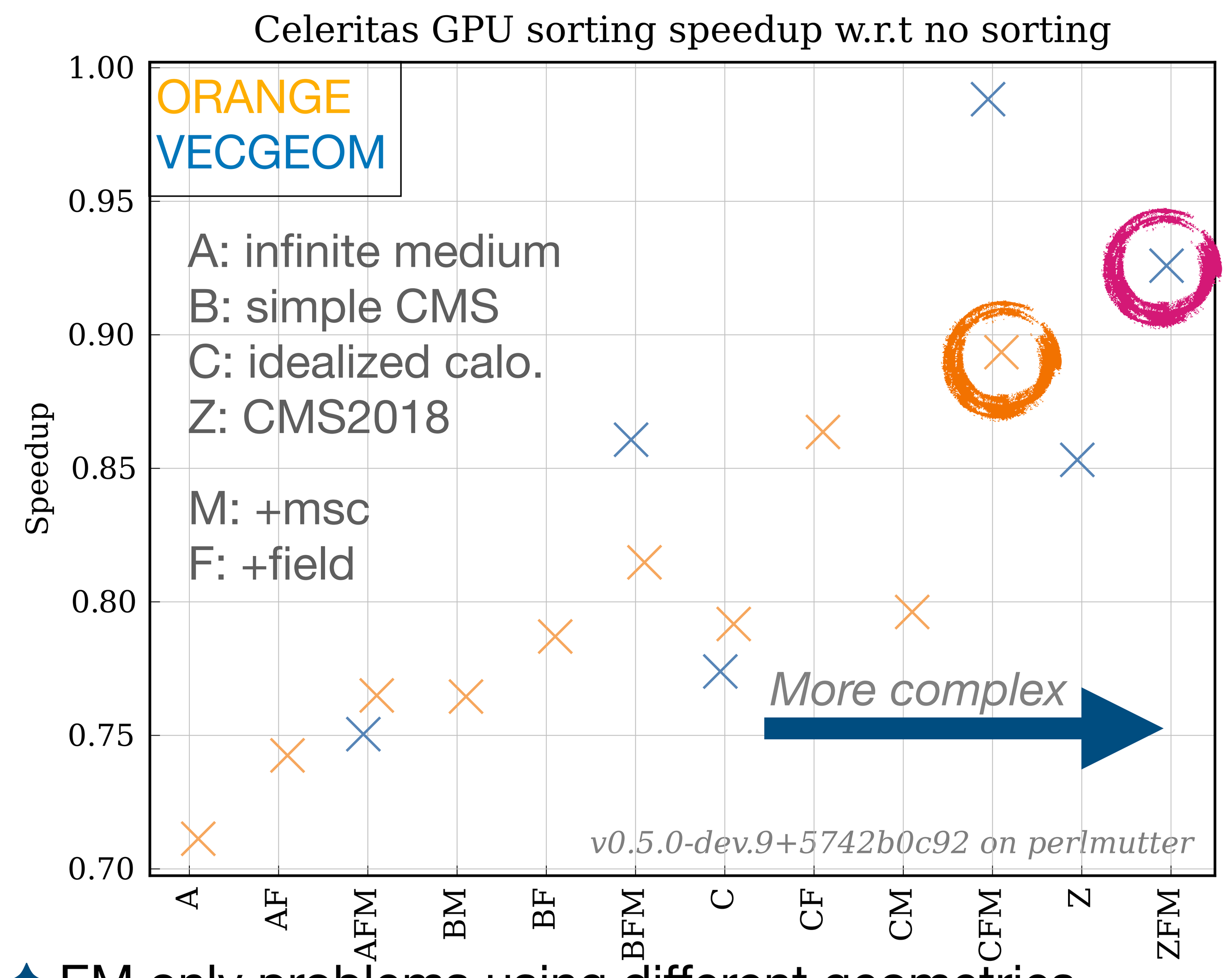
- ✦ Monte Carlo transport library for HEP simulations on GPUs
- ✦ Compatible with the Geant4 simulation toolkit
- ✦ Supports Vecgeom and ORANGE geometry engines
- ✦ Portable between CUDA and HIP
- ✦ Implements standard EM physics

2. Stepping Loop

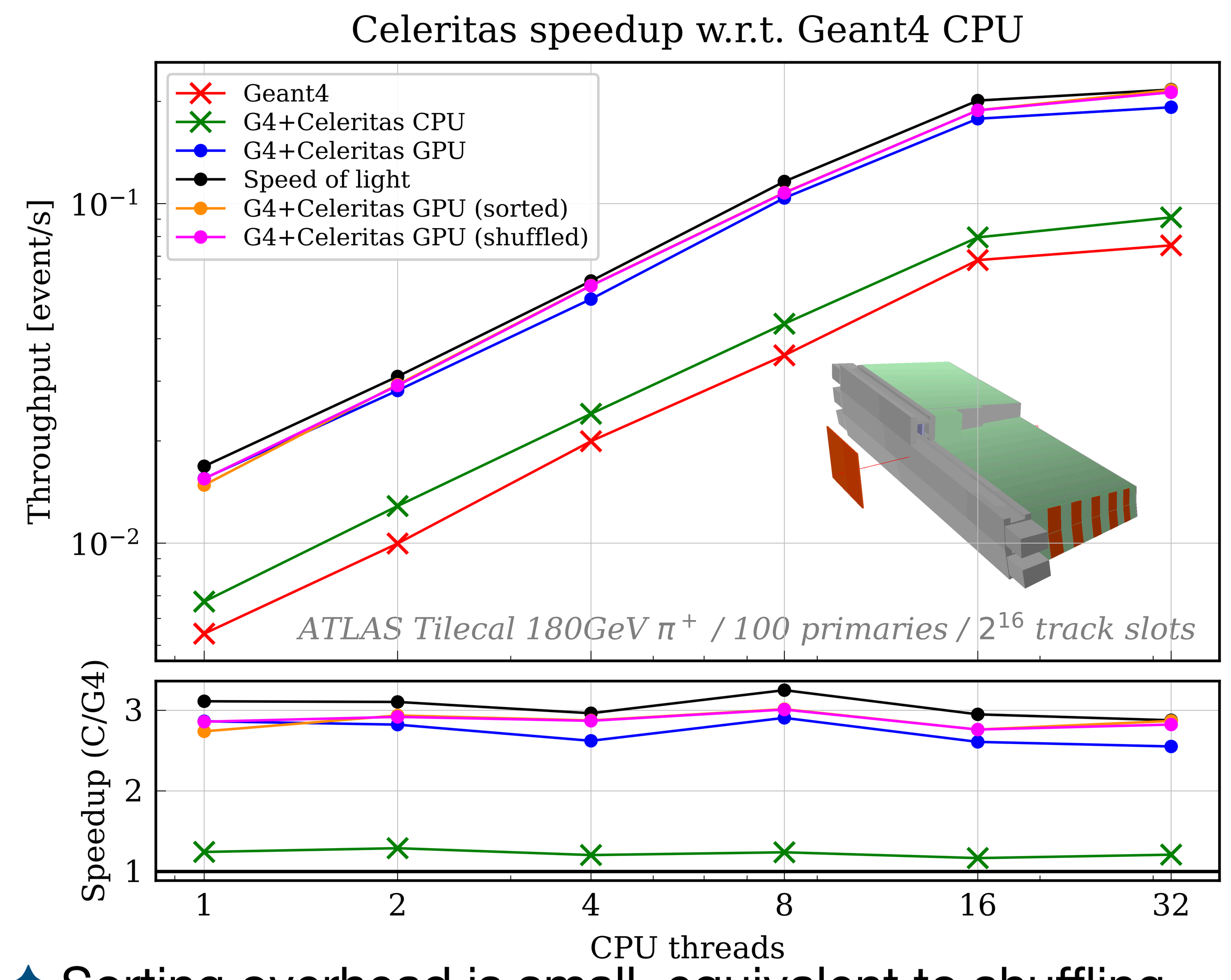


- ✦ Each simulation step executes a topologically sorted graph of actions (gray boxes)
- ✦ Sort actions can be inserted at different stages in the graph (blue circles) depending on the sorting key
- ✦ Actions are applied to all tracks offloaded to Celeritas
- ✦ Tracks to which an action doesn't apply are masked
- ✦ Sorting should reduce thread divergence caused by masking
- ✦ Only the track slot index is sorted, the track state is not moved

3. Benchmark Problems

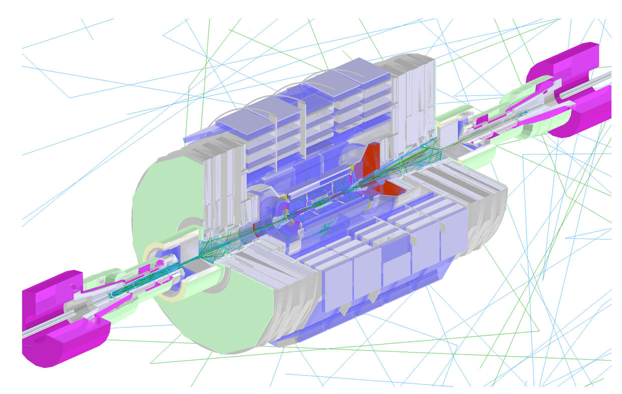


- ✦ EM only problems using different geometries
- ✦ Sort based on the along-step action assigned to the track
- ✦ Sorting results in a slowdown, more so for simple geometry

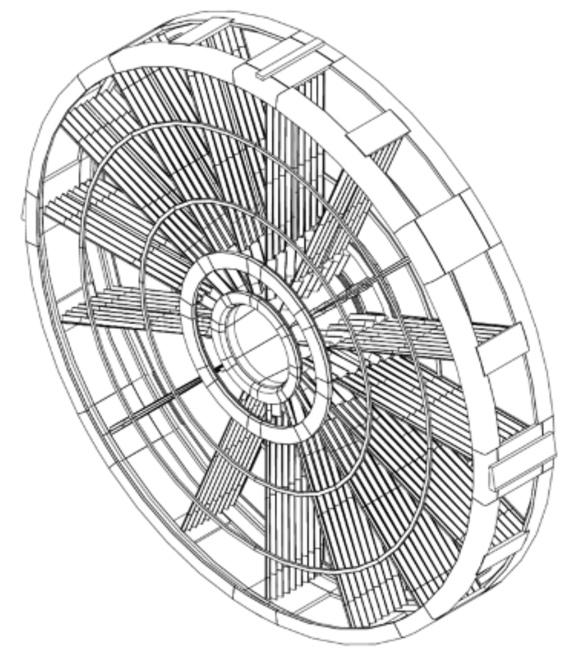


- ✦ Sorting overhead is small, equivalent to shuffling
- ✦ "Speed-of-light" kills all tracks offloaded to Celeritas
- ✦ Dominated by Geant4 hadronic physics

CMS2018 geometry



ATLAS EMEC geometry



4. Along-step Kernel Profiling on A100

Problem		# instr. [M instr.]	# Branch instr. [M instr.]	Active threads/ warp	Cycles/instr.	GMem R+W [MB]	Kernel time [ms]
ATLAS EMEC	Baseline	262.27	18.63	1.69	7.55	11.29	7.19
	Sort	160.58 (-38%)	12.08 (-35%)	2.75 (+63%)	7.85 (+4%)	8.69 (-23%)	5.5 (-23%)
Full CMS 2018 (VecGeom)	Baseline	458.72	27.82	2.41	5.81	152.58	13.51
	Sort	384.7 (-16%)	22.97 (-17%)	2.8 (+16%)	6.31 (+8%)	358.24 (+134%)	9.46 (-30%)
TestEM3 (ORANGE)	Baseline	164.55	18.91	6.41	9.61	187.26	2.48
	Sort	110.33 (-33%)	12.71 (-33%)	9.44 (+47%)	18.80 (+96%)	832.15 (+344%)	2.98 (+20%)

- ✦ The impact of sorting tracks by applicable along-step kernel on execution time is problem-dependent
- ✦ Many tracks per kernel and a complex geometry are required to make track sorting worthwhile

Authors

Julien Esseiva *Lawrence Berkeley National Laboratory*
Seth R. Johnson *Oak Ridge National Laboratory*
Soon Yung Jun *Fermi National Accelerator Laboratory*
Amanda Lund *Argonne National Laboratory*
Benjamin Morgan *University of Warwick*

U.S. DEPARTMENT OF ENERGY

Acknowledgments

This material is based upon work supported by the U.S.~Department of Energy, Office of Science, Office of Advanced Scientific Computing Research and Office of High Energy Physics, Scientific Discovery through Advanced Computing (SciDAC) program. - Work for this paper was supported by Oak Ridge National Laboratory (ORNL), which is managed and operated by UT--Battelle, LLC, for the U.S.~Department of Energy (DOE) under Contract No.~DE-AC05-00OR22725 and by Fermi National Accelerator Laboratory, managed and operated by Fermi Research Alliance, LLC under Contract No.~DE-AC02-07CH11359 with the U.S.~Department of Energy. - This research was supported by the Exascale Computing Project (ECP), project number 17-SC-20-SC. The ECP is a collaborative effort of two DOE organizations, the Office of Science and the National Nuclear Security Administration, that are responsible for the planning and preparation of a capable exascale ecosystem---including software, applications, hardware, advanced system engineering, and early testbed platforms---to support the nation's exascale computing imperative. - This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S.~Department of Energy under Contract No.~DE-AC05-00OR22725. - This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S.~Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No.~DE-AC02-05CH11231 using NERSC award HEP-ERCAP-0023868.