Parallel 6B – R&D

J. Apostolakis

Thu 28/09

Print | PDF | Full screen | Detailed view | Filter

09:00	**AdePT progress report**	*Jonas Hahnfeld*	*Hokkaido University, Room B*	09:00 - 09:30
	Celeritas progress report	*Seth Johnson*	*Hokkaido University, Room B*	09:30 - 10:00
10:00	**Opticks progress report**	*Hans-Joachim Wenzel*	*Hokkaido University, Room B*	10:00 - 10:30

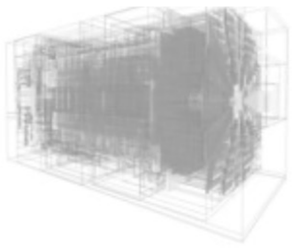
AdePT status report

Jonas Hahnfeld for the AdePT team
28.09.2023

28th Geant4 Collaboration Meeting

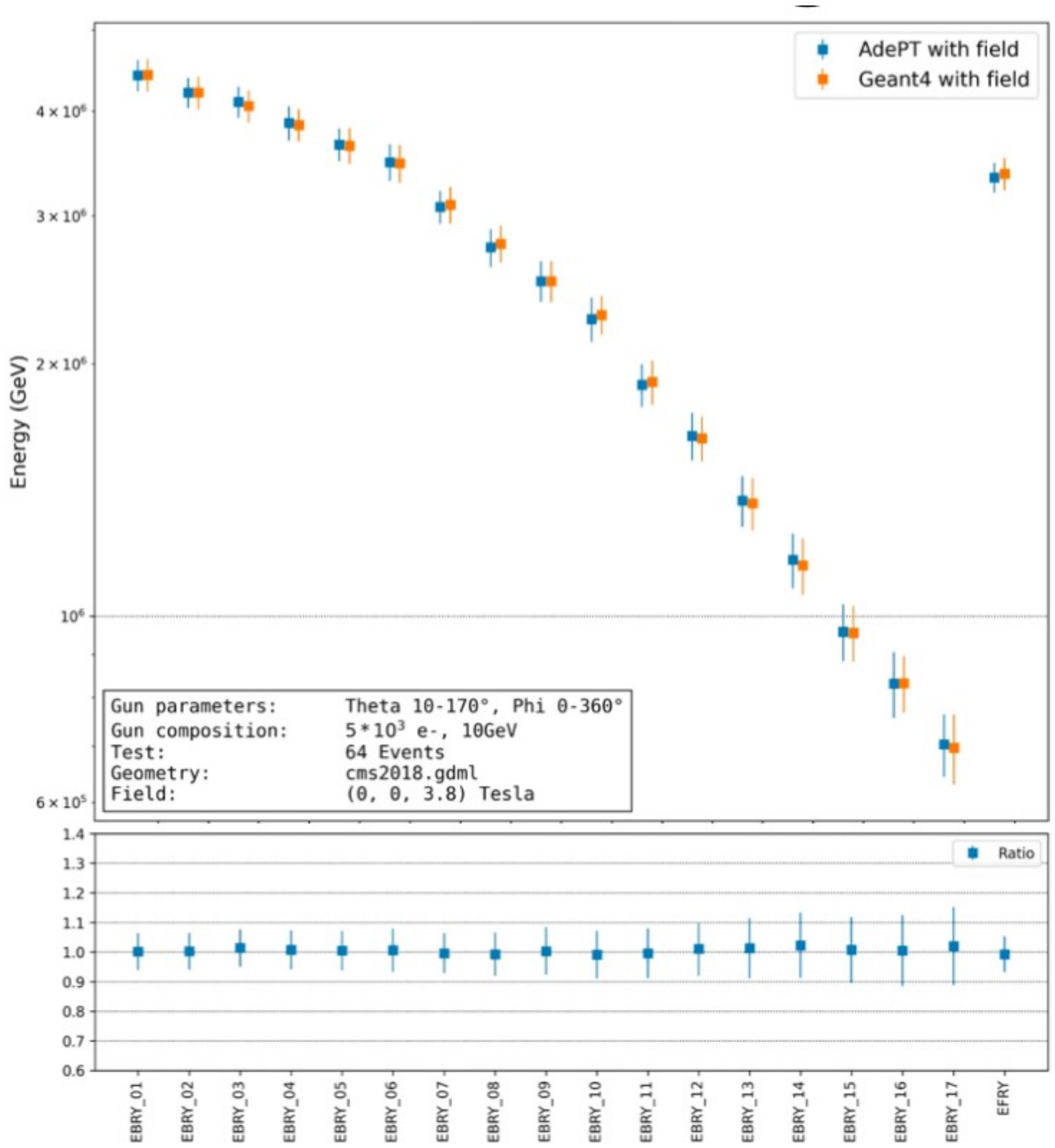
- Hybrid CPU-GPU workflow
  - off-loading tracks from Geant4 application
- On GPU step all tracks in parallel
- One kernel per particle type (e^-, e^+, γ)
- Validation against G4 & G4HepEM

New surface model

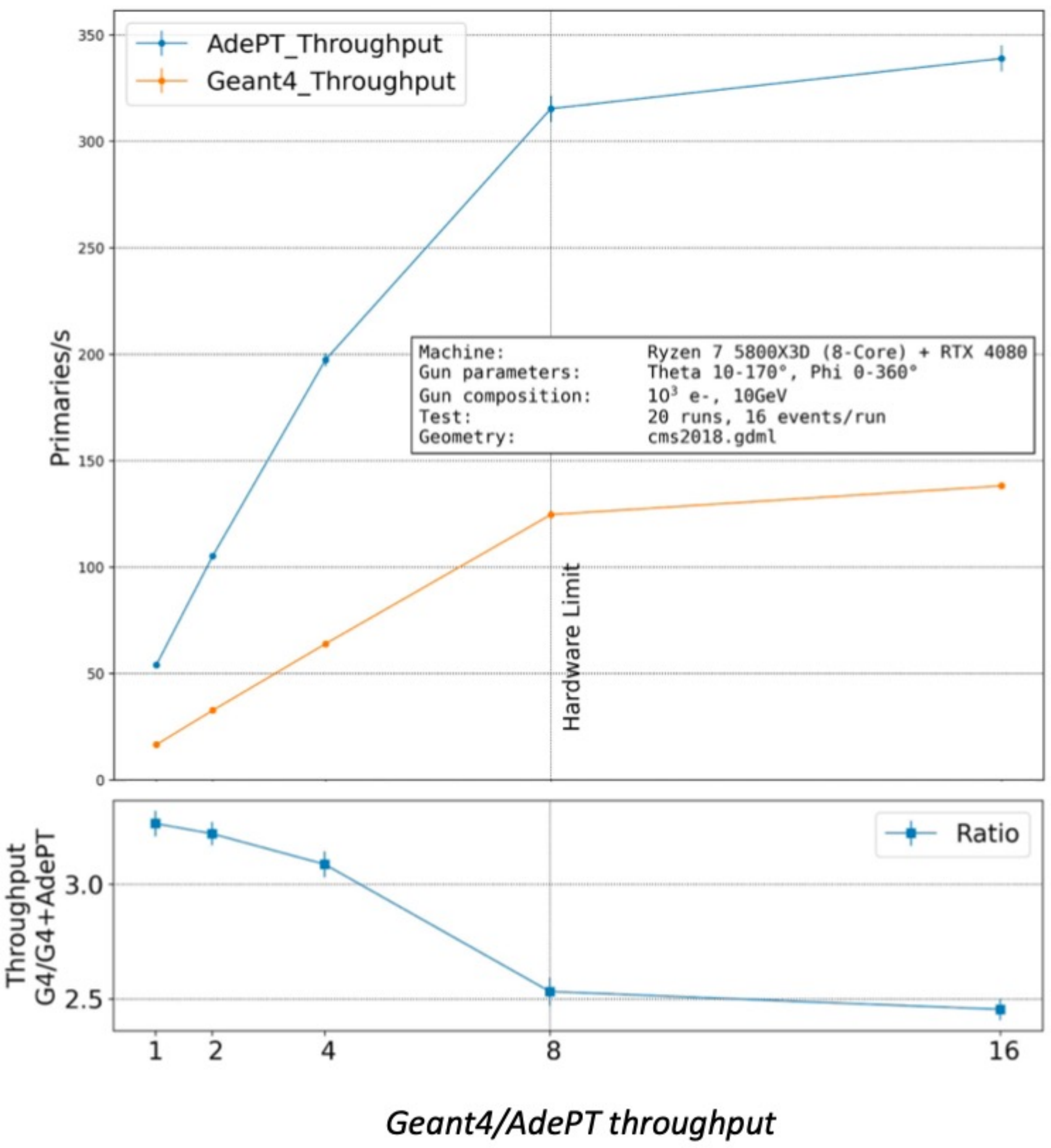


- Started >1 year ago a new surface model
- Header library implementation for GPU performance & portability
  - Eliminated recursions and virtual calls
  - Faster relocation, better scaling features and work balancing
  - Set of supported 3D solids increasing:
    - box, simple and general trapezoids, parallelepipede, tube, cone, polyhedron
    - Working on extruded, polycone

Validation



Benchmarking



Integration with Experiments

- To test AdePT, and identify challenges/issues
- Add complex setups
- Create realistic hit output & understand its perform. Impact

- ATLAS TileCal test beam
  - Using standalone G4 application
  - Total E per module done
  - Investigating detailed hit output
- LHCb Gaussino – calorimeter
  - Target same ‘Gauss’ hits
- CMSsw
  - G4HepEM on CPU – working in simple & ‘specialised tracking’ modes
  - Next-gen HGCal: defining info to record

Ongoing and summary

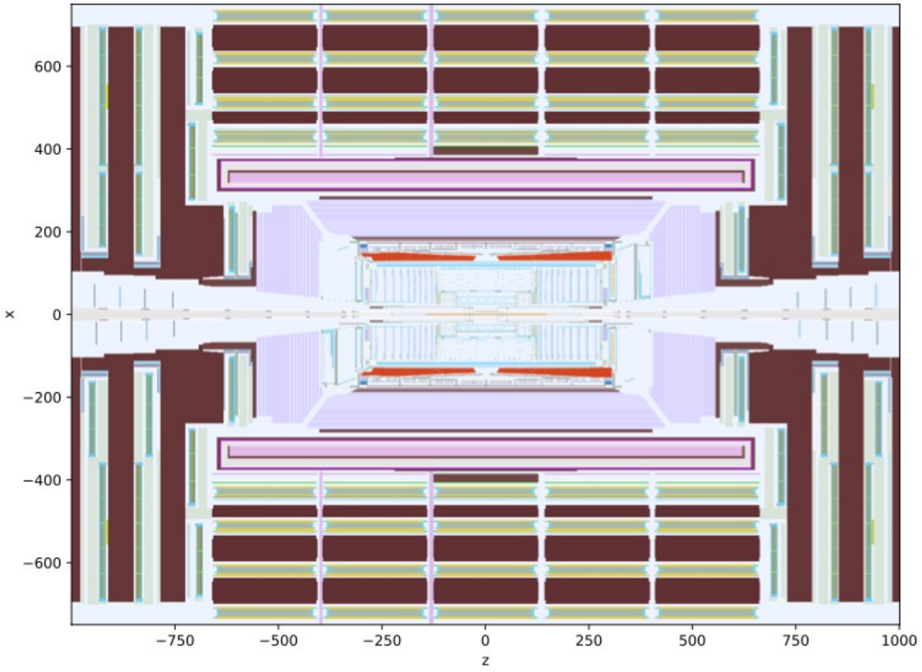
Current ongoing:

- New GPU-focused Geometry modeller (key for performance)
- Non-constant B field
  - validation & optimisation
- Integration with experiments

Outlook

- Focus on robustness, validation & benchmarking

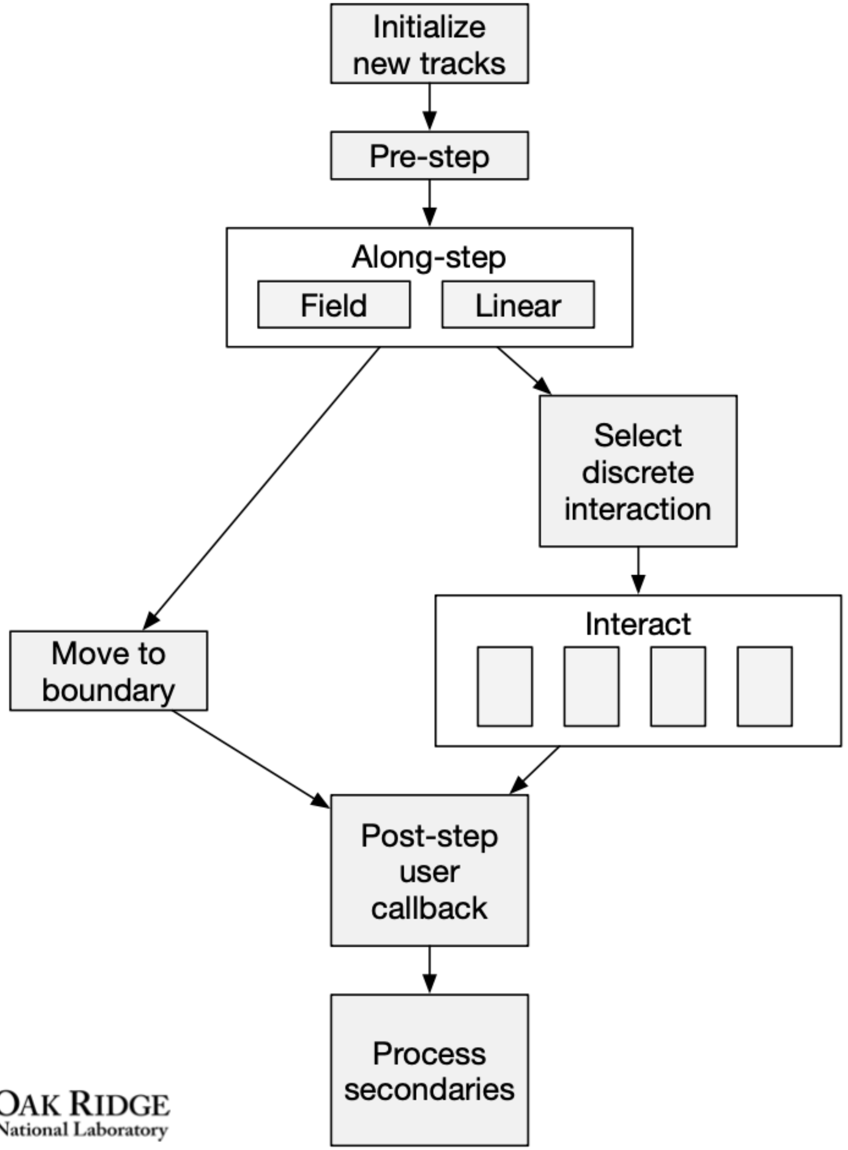
- GPU-focused HEP simulation
- Motivated by HL-LHC challenge & recent success in GPU MC (Exascale)
- Goal: LHC Run4

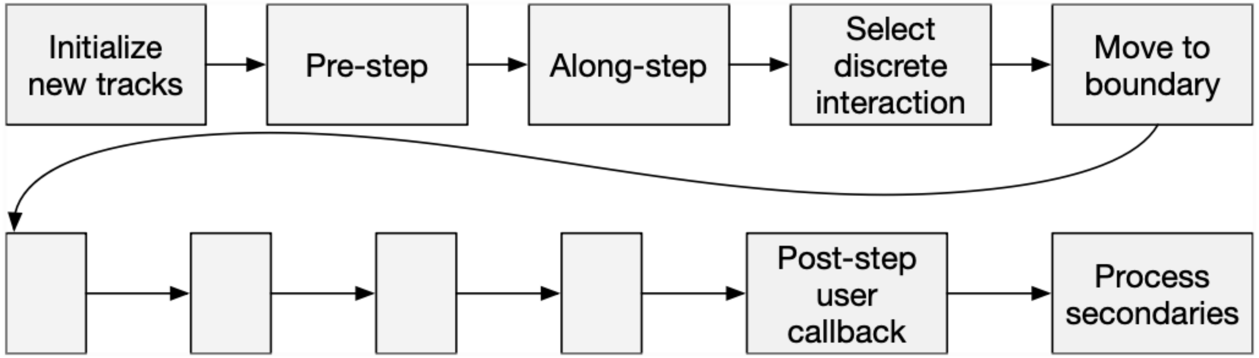


GPU-traced rasterization of CMS 2018

- Physics equivalent to G4EMStandardPhysics
- Full-feature Geometry via VecGeom 1.x (CPU, CUDA)
- Configurable at run-time: field, physics
- Can be run on HIP(AMD), Nvidia & CPU

Stepping loop on a GPU





Topological sort: a loop over kernels

Process large batches of tracks per kernel (10^3–10^6)

Celeritas v 0.3: Geant4 integration status

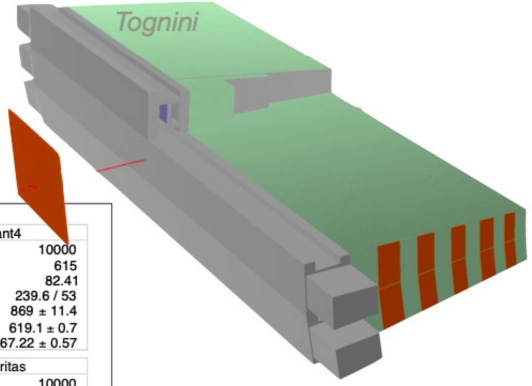
- **Imports** EM physics selection, cross sections, parameters
- **Converts** geometry to VecGeom model
- **Offloads** EM tracks from Geant4
- **Scores** hits to user "sensitive detectors"
  (Copies from GPU to CPU; reconstructs `G4Hit`, `G4Step`, `G4Track`; calls `Hit`)
- **Builds** against Geant4 10.5–11.1

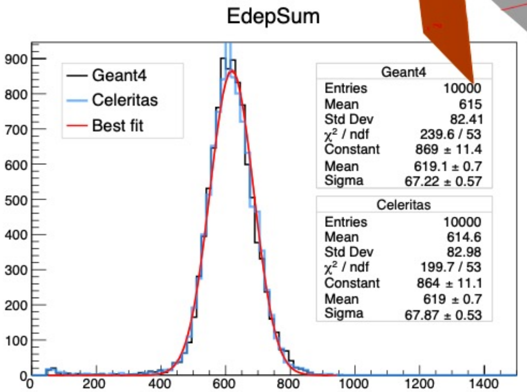
New development for quicker adoption.
Store information on x, p, DE and use on CPU to recreate G4Step/Track, then call SD hit class.

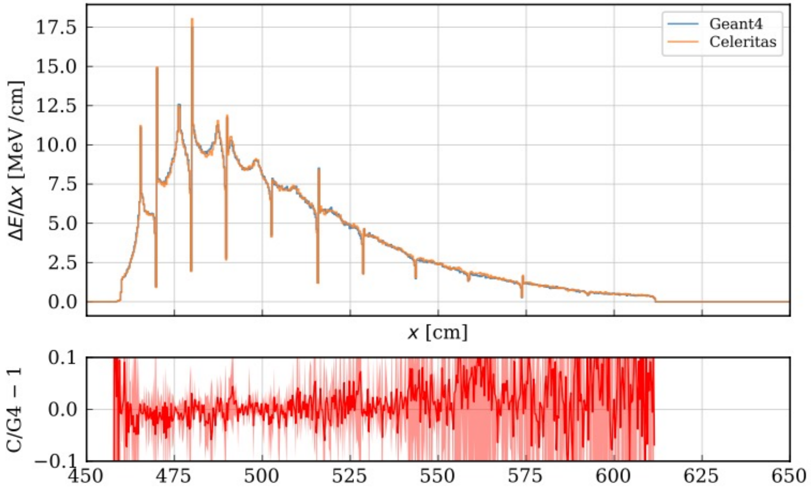
Two examples of Integration with Experiment code shown:
- ATLAS 'FullSimLight' standalone simplified simulation
- CMSsw framework integration

EM offloading with FullSimLight

- ATLAS FullSimLight: hadronic tile calorimeter module segment
  - 64 segments in full ATLAS, 1 in this test beam
  - 18 GeV π^+ beam, no field
  - FTFP_BERT (default) physics list
    (includes standard EM)
- **~100 lines of code to integrate**
  - Offload e^-, e^+, γ to Celeritas
  - Celeritas reconstructs hits and sends to user-defined `G4VSensitiveDetector`
- Excellent agreement in energy deposition







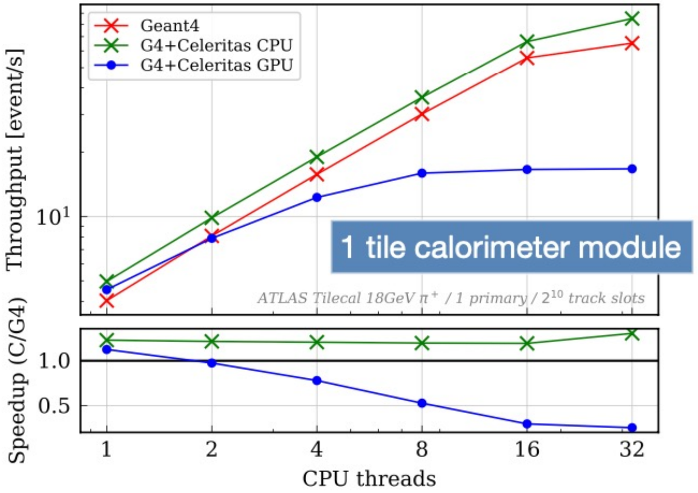
yz-integrated energy deposition

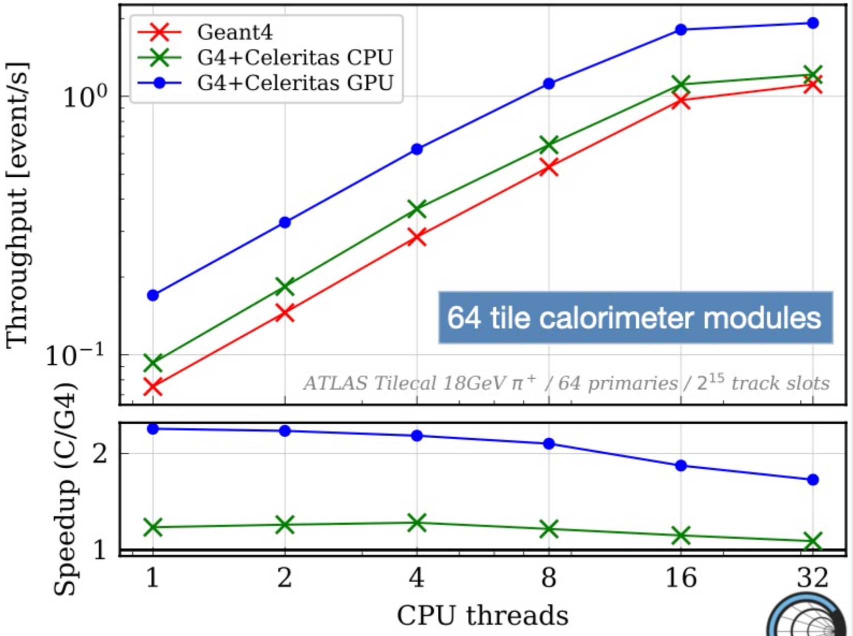
Oak Ridge National Laboratory

Test problem: Lachnit, Pezzotti; FSL integration: Morgan

Offload performance results

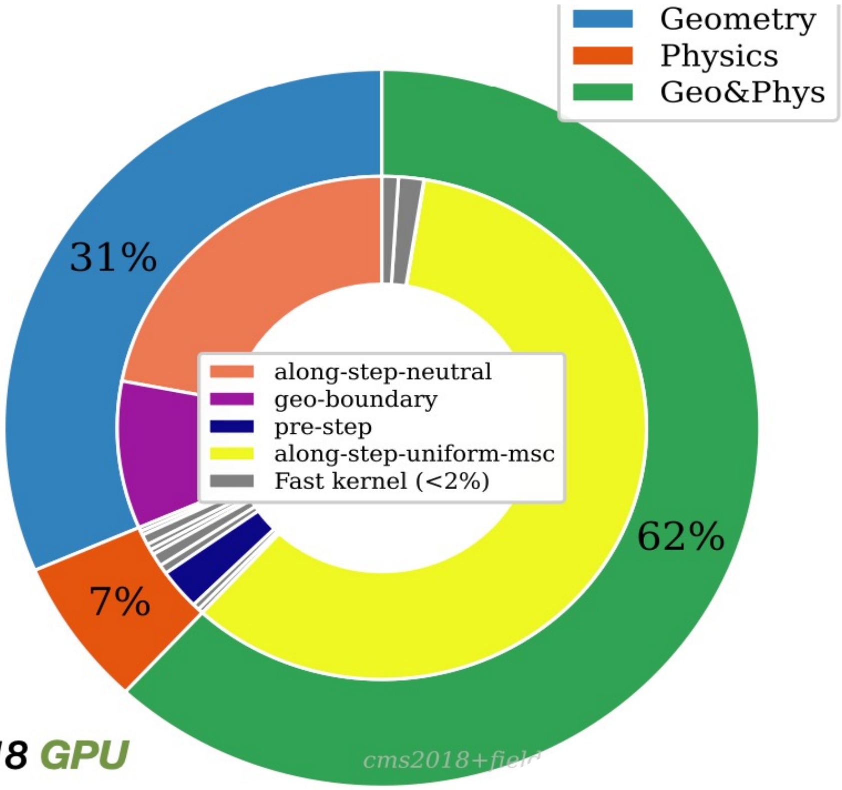
- 1/4 of a Perlmutter (NERSC) GPU node
  16 cores of AMD EPYC, 1 Nvidia A100
- Time **includes** startup overhead, Geant4 hadronic physics, track reconstruction, and SD callback *(2048 π^+ in all cases)*
- GPU speedup: **1.7–1.9×** at full occupancy
  Using all CPU cores with a single GPU
- CPU-only speedup: still **1.1–1.3×**!
- LHC-scale energy per event (i.e., all 64 modules) is needed for GPU efficiency
- One fast GPU can be shared effectively by full multithreaded Geant4





Ongoing

- Integration
- Verification & validation
- Optimisation & geometry
  - 90% time in geometry



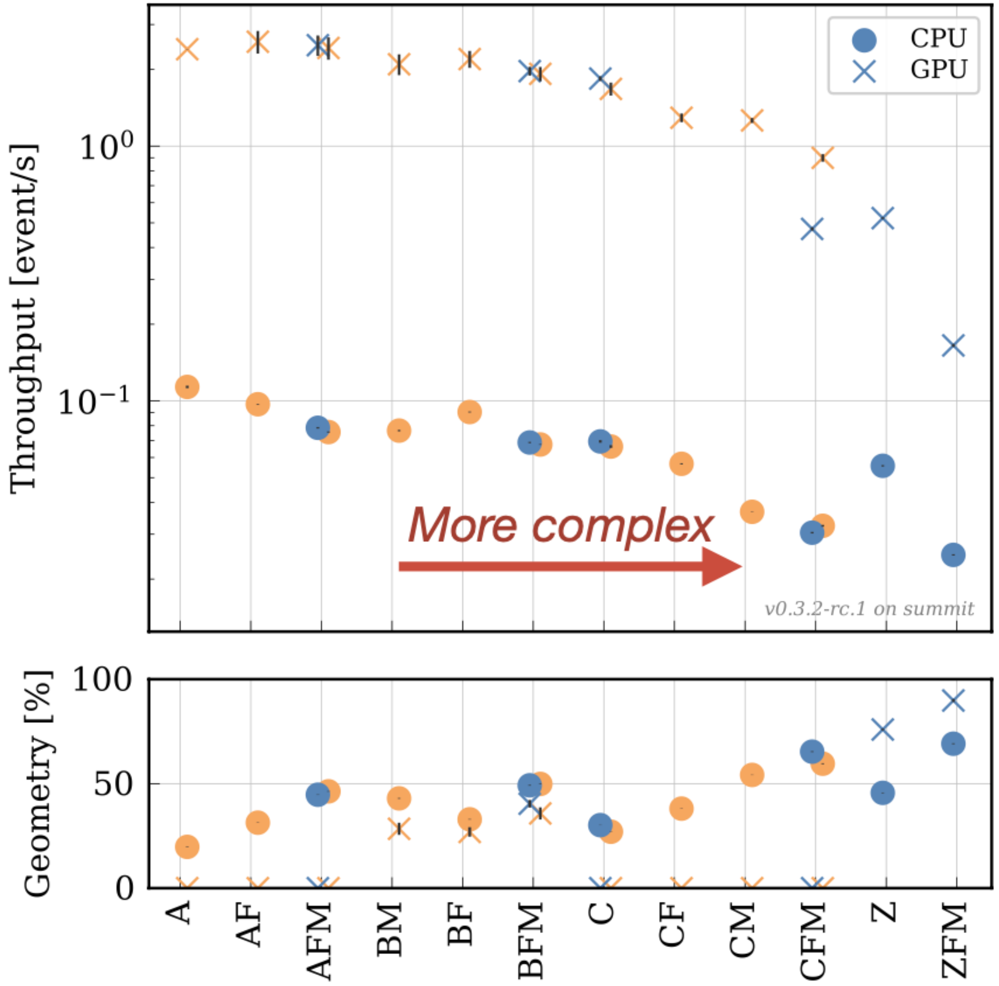
CMS2018 GPU

Summary

- Straightforward integration into existing Geant4 apps
- Demonstrated performance gains by offloading EM to Celeritas
  - Comparisons with 1 GPU, multicore CPU, against pure Geant4
  - Calorimeter test beam net improvement: **10–30% faster** on CPU, **1.8–2.2×** on GPU (Nvidia A100)
  - CMS Run 3 configuration standalone simulation speedup: **12–87% faster** on GPU (Nvidia V100)
- Anticipated performance even higher
  - Standalone EM problems: **~7–34× faster** (Celeritas CPU vs GPU) on Summit (Nvidia V100) (49–238× GPU/CPU core equivalence)
  - ORANGE vs current VecGeom for TestEM3: **85% faster**

Speedup: Comparison of configurations

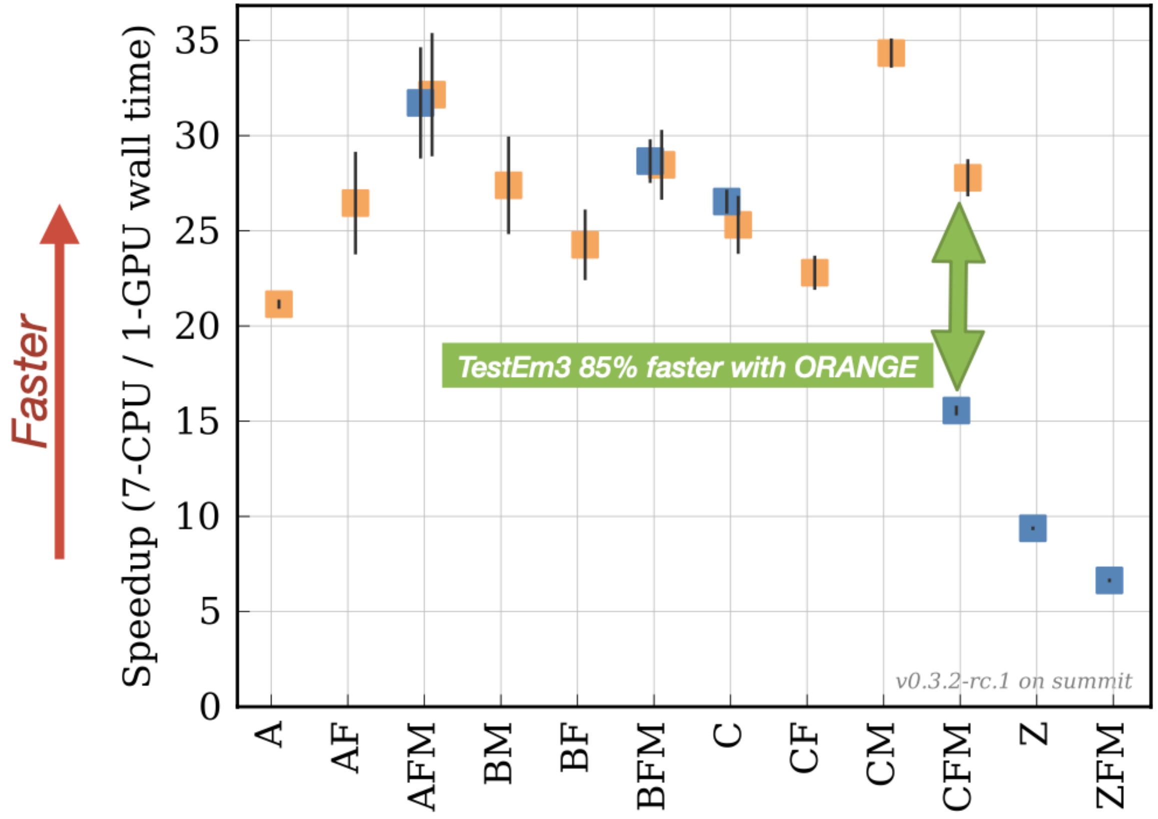
EM-only GPU performance



ORANGE
VecGeom

Problem definition	
A	TestEm15
B	Simple CMS
C	TestEm3
Z	CMS 2018

Modifier	
F	+field (1T)
M	+msc (Urban)



Multiply speedup by 7× for CPU:GPU equivalence

CaTS: Integration of Geant4 and Opticks

Hans Wenzel

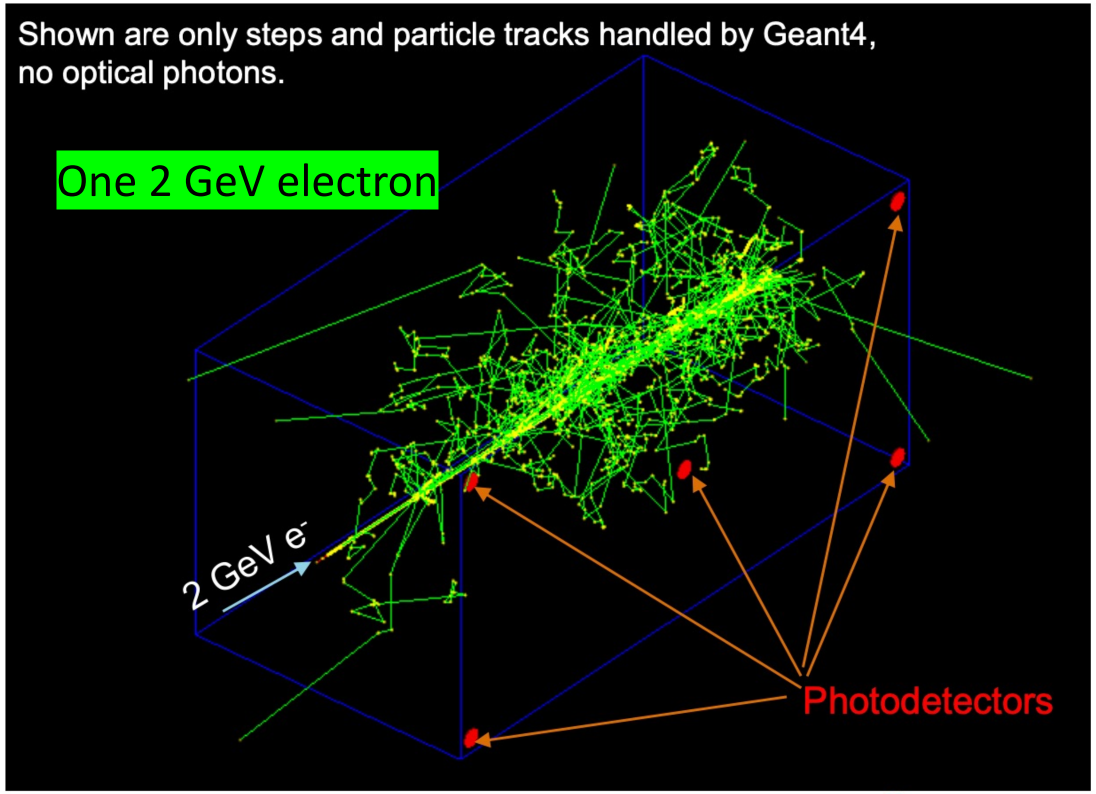
Soon Yung Jun
Krzysztof Genser

Outline

- Motivation:
  - The computational challenge for TPCs based on liquid Argon (LArTPCs).
  - Simulation of optical photons: an ideal application to be ported to GPU's.
- Opticks.
- CaTS is an advanced example Geant4 application.
  - CaTS workflow.
  - Performance.
- Plans.

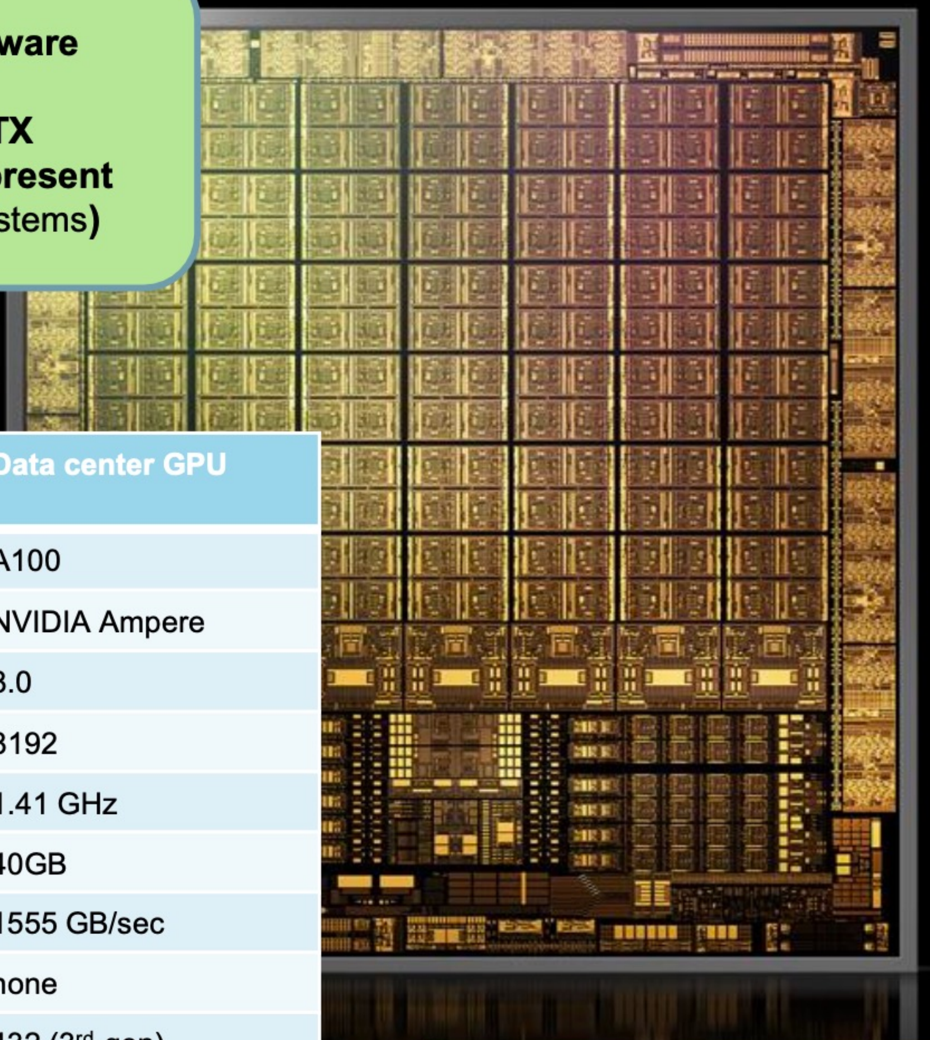
Computational Challenge of liquid Argon TPCs

- Photon yield 50,000 photons per MeV of ΔE
- Full simulation in Geant4 needs **minutes / event**
- Without simulating optical photons 0.034 sec/event
- So production simulations currently use **tables & parameterisations** for detector response
- Opportunity to use GPU to offload optical photon simulation (well suited)



Parallelism for Optical Photons

Opticks will only run on:
NVIDIA® hardware and NVIDIA® software
Software: NVIDIA® CUDA, OptiX
OptiX 6: allows to select/deselect RTX
OptiX 7: RTX cores are used when present
(RTX is not usually available on HPC systems)

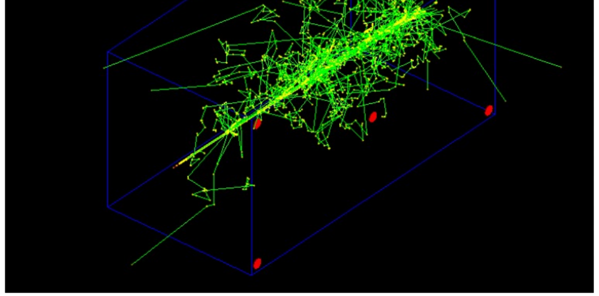


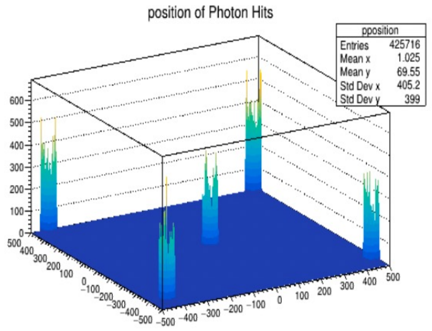
	Graphics card	Data center GPU
	GeForce RTX 3090	A100
architecture	NVIDIA Ampere	NVIDIA Ampere
Compute capability	8.6	8.0
CUDA cores	10,496	8192
Boost Clock	1,7 GHz	1.41 GHz
Memory	24 GB	40GB
Memory bandwidth	936 GB/sec	1555 GB/sec
RT cores	82 (2nd-gen)	none
Tensor cores	382 (3rd-gen)	432 (3rd-gen)
Shared Memory size	64kB	up to 164 kB

RT core: based on bounding volume hierarchy (BVH), a commonly used acceleration structure in ray tracing, ray-triangle intersection.

Performance: “Legacy Opticks”

Hardware:	
CPU	Intel® Core i9-10900k@ 3.7 GHz, 10 CPU cores
GPU	NVIDIA GeForce RTX 3090 @ 1.7 GHz, 10496 cores
Software:	
Geant4: 11.0, Opticks based on OptiX® 6	





Number of CPU threads	Single threaded. Geant4 [sec/evt]	Opticks [sec/evt]	Gain/speed up
1	330	1.8	189x

→It becomes feasible to run full optical simulation event by event! But comparison is to single threaded Geant4→ somehow unfair! Single geant4 threat can saturate the GPU and doesn’t allow the use of multiple CPU cores.

Fermilab

Ongoing/plans

Nvidia Optix 7 – large change
Opticks reengineered to make it

- more modular, easier to test
- changing API

Revising CaTS

Plans

- Adjust to the new Opticks API → in progress. At the moment we keep compatibility with the legacy Opticks → propably drop this at some point.
- Once it’s working:
  - Benchmark the performance compared to legacy Opticks.
  - Compare Opticks with multithreaded Geant4 on multicore machines for fair comparison.
  - Profiling using nvprof.
  - Physics validation.
- Update the Geant4 advanced example CaTS:

Also to note

- Internal Geant4 assessment of R&D – 13-14 December 2023