EIC-Opticks – Integration of the Opticks framework into Geant4 simulation of optical photons on GPU

Gabor Galgoczi, Dmitry Kalinkin, Kolja Kauder, Maxim Potekhin, Sakib Rahman, Dmitri Smirnov, Torre Wenaus

Nuclear and Particle Physics Software (NPPS) Group

BNL Physics Department

March 2026

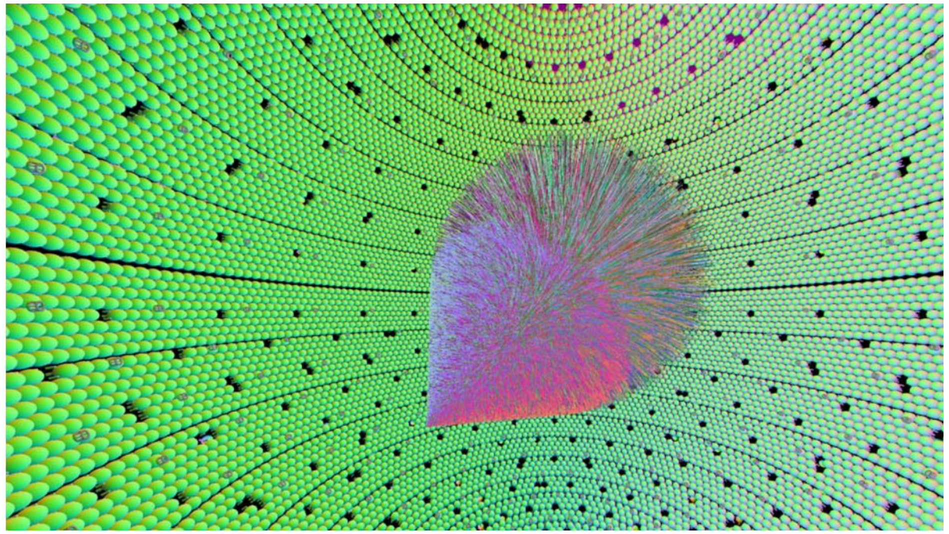
Simulation WG meeting

Outline

- Introduction to Opticks and EIC-Opticks
- Optical photons often dominate simulation time; GPU ray tracing (OptiX) can simulate O(10^7) photons/s
- What was added in EIC-Opticks (new features, dd4hep integration)
- Validation of Opticks and EIC-Opticks and performance quantification
- Potential usage in ePIC (and beyond)
- Limitations: NVIDIA only backend, limited geometry primitives
- Plans: Implement more physics and additional shapes (e.g. trapezoids), support G4Tasking and multi-GPU use

Opticks[1]

- Developed by Simon Blyth for the JUNO experiment
- Uses NVIDIA's OptiX[2] ray tracing engine to implement optical photon propagation on GPUs in Geant4 -> simulate 10^7 photons in 1 second
- Geometries:
  - Opticks can import 10 "primitives" (eg. box, sphere, torus…) either from GDML or Geant4 automatically



Photons simulated @ JUNO with Opticks[1] from 200 GeV muon (~10^7 photons)

[1] Opticks: GPU Optical Photon Simulation via NVIDIA OptiX EPJ Web of Conferences 295, 11014 (2024)

[2] https://developer.nvidia.com/rtx/ray-tracing/optix

Opticks

- Opticks is used by: JUNO, NEXT-CRAB0, LHCb RICH 1, VPG4
  - Initiatives in: LZ, MOLLER, AMoRE, nEXO, KM3NeT, DUNE experiments
  - Existing official G4 example using Opticks: CaTS
- Physical processes ported from G4 to GPU code:
  - Bulk absorption
  - Bulk reemission
  - Rayleigh scattering
  - Mie scattering
  - Surface processes (eg. absorption, diffuse, specular reflection…)
  - Photon wave effects: transfer matrix method
  - (Missing: wavelength shifting)

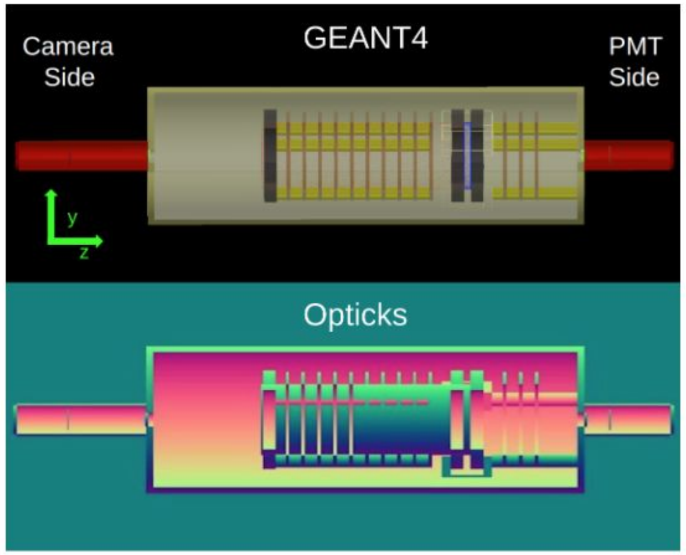
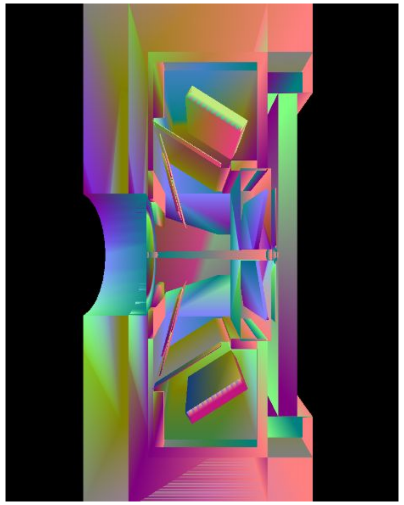


Fig. 3 Geometry rendering of `NEXT-CRAB-0` in `Opticks` and `Geant4`. The z direction points along the drift axis (with $z = 0$ closest to the electroluminescence region), while the x and y coordinates are perpendicular to this.

NEXT Collaboration. Eur. Phys. J. C 85, 910 (2025)



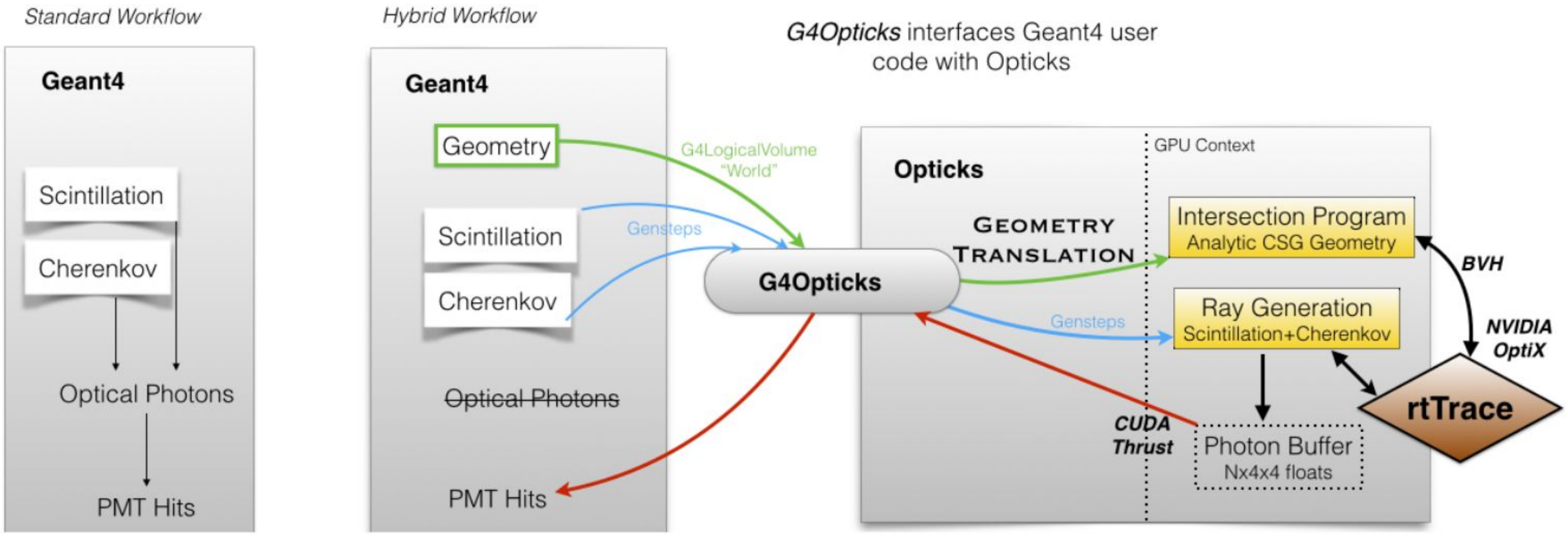
Yunlong Li et al.: GPU-based optical photon simulation for the LHCb RICH 1 Detector

Opticks video of Daya Bay and JUNO

- NVIDIA Optix was designed for visualization
- Can visualize detectors too!
- Example:
  https://www.youtube.com/watch?v=CBpOha4RzIs

Architecture

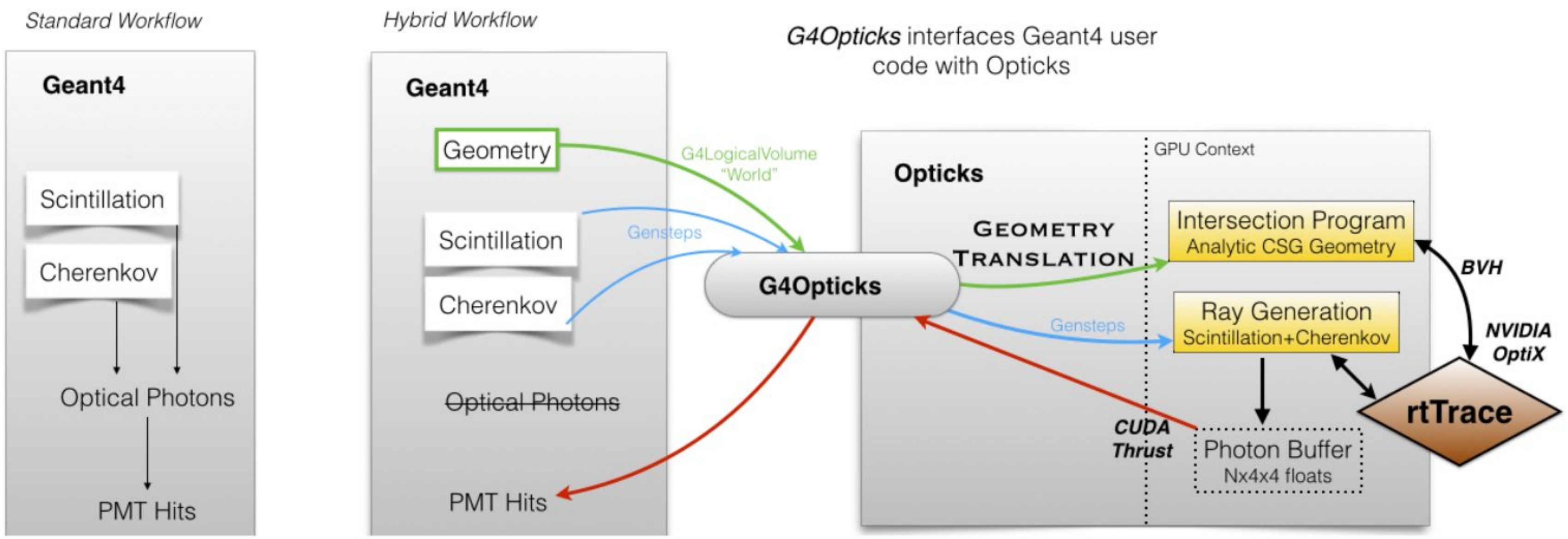
- EM, Hadronic done in Geant4 with CPU
  - For Cherenkov/scintillation, Geant4 calls G4Opticks API instead of creating photons
- G4Opticks records production distributions (position, direction, polarization, Nγ, event ID, …)
- EIC-Opticks uses these records to generate and propagate photons on the GPU with NVIDIA OptiX
- Hits are sent back through G4opticks and the user can do whatever they want -> DD4Hep, ROOT…
- Geant4 and EIC-Opticks share the same geometry and optical properties → same physics, much faster optical part



Simon Blyth, EPJ Web of Conferences 245, 11003 (2020)

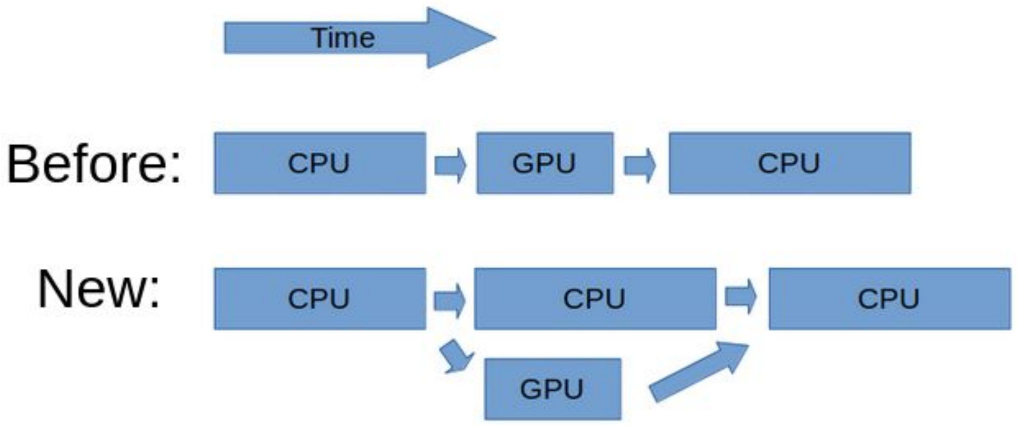
API/integration snippet

- Standalone example: esi-g4ox: https://github.com/BNLNPPS/esi-g4ox/tree/PerformanceAnalysis
- In **G4DetectorConstruction** call **G4CXOpticks::SetWorld()** transforms geometry into CSG (Constructive Solid Geometry) and then BVH (Bounding Volume Hierarchy) for NVIDIA
- In **G4SteppingAction** call **U4::CollectGenstep_G4Cerenkov_modified()** and pass Cerenkov distribution parameters (similar for scintillation)
- In G4RunAction or G4EventAction call
  - **G4CXOpticks->simulate()** runs the GPU ray tracing
  - **SEvt::Get_EGPU()->getHit()** returns the hits from GPU



EIC-Opticks

- EIC-Opticks[1]: fork of Opticks
- Modifications:
  - Opticks can only simulate one G4Event in a GPU simulation
    - Very large overhead when simulating less than 1M photons in a GPU call
  - EIC-Opticks extended usability to cases with much fewer photons (as in most EIC applications)
    - Added G4Event ID to GPU simulated photons
  - Asynchronous CPU, GPU launches



[1] github.com/BNLNPPS/eic-opticks

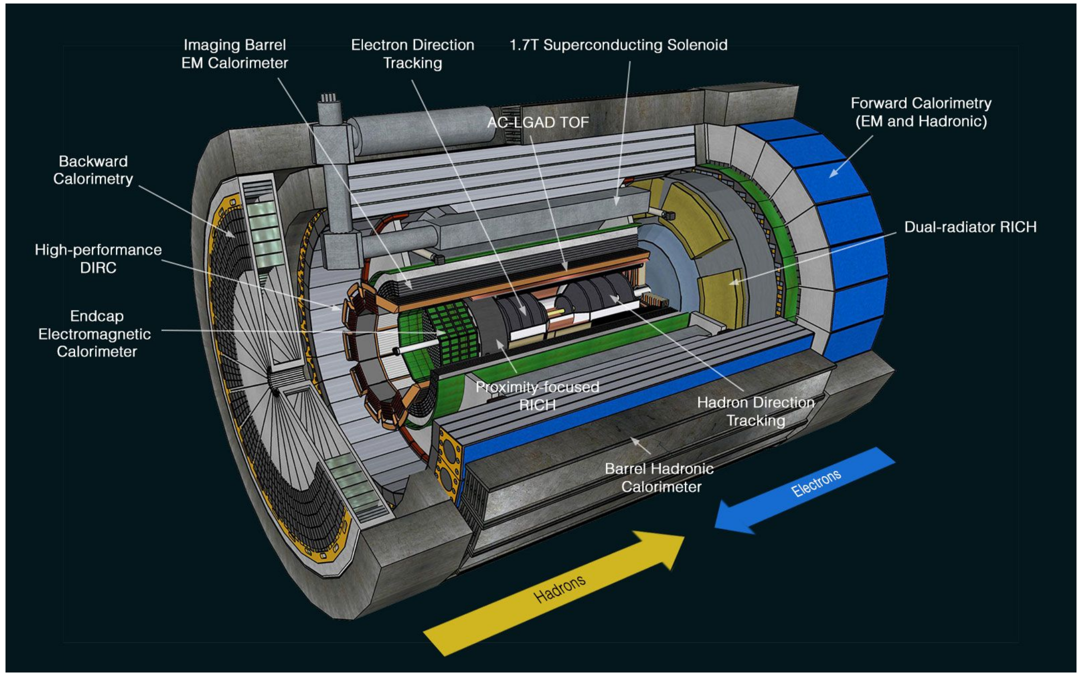
EIC-Opticks

- EIC-Opticks[1]: fork of Opticks
- Modifications:
  - Opticks can only simulate one G4Event in a GPU simulation
    - Very large overhead when simulating less than 1M photons in a GPU call
  - EIC-Opticks extended usability to cases with much fewer photons (as in most EIC applications)
    - Added G4Event ID to GPU simulated photons
  - Asyncronous CPU, GPU launches
  - Fixed some bugs in Opticks, eg. some cones shapes were not working
  - Created a spack package
  - Tested NERSC Perlmutter
  - Installed in eic_cude container
  - Integrating eic-opticks now into dd4hep:
    - Soon can be used through dd4hep in eic_cuda containers
- Besides EIC there is interest from medical physics community:
  - Harvard scientist testing it now
  - We are integrating eic-opticks into opengate sim. framework

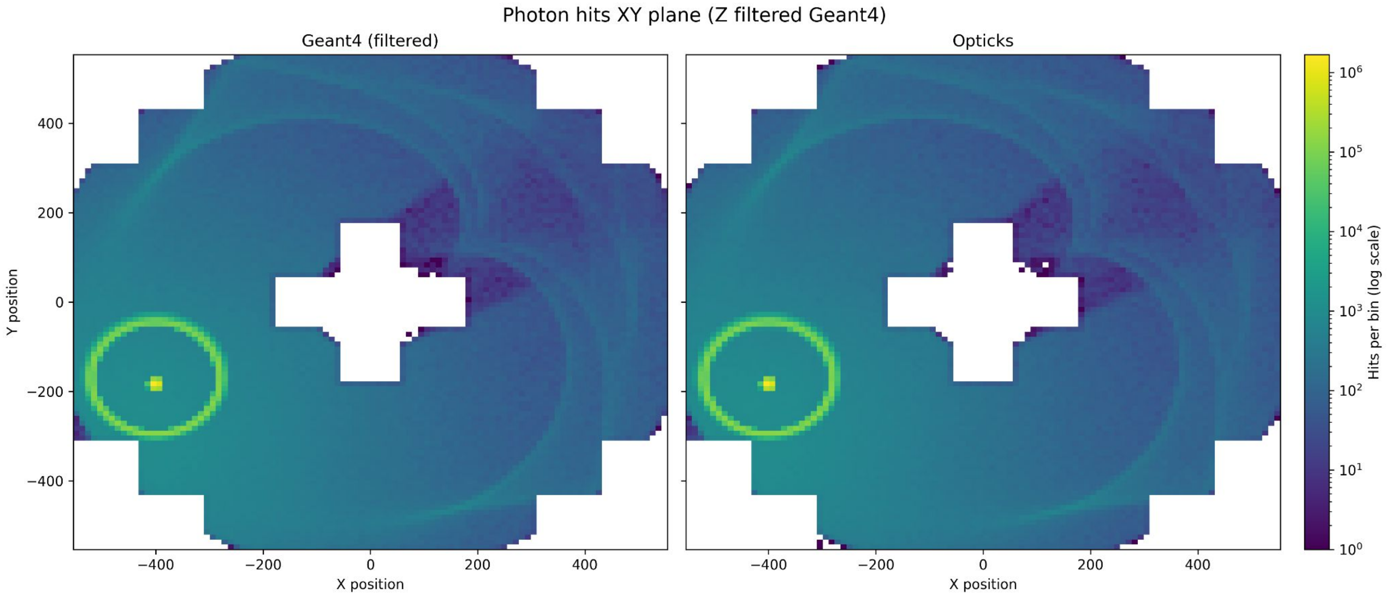
[1] github.com/BNLNPPS/eic-opticks

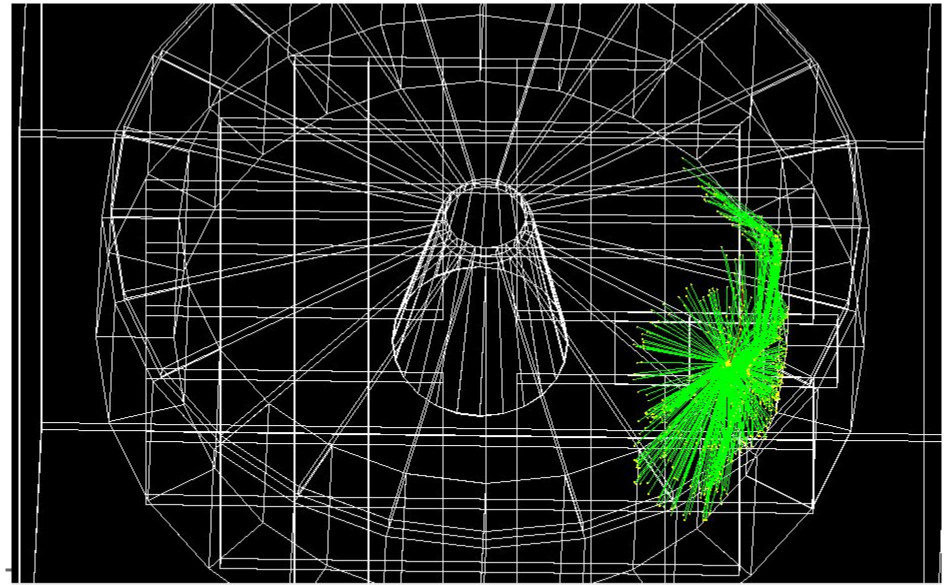
Validation of EIC-Opticks vs G4 w. pfRICH geometry

- Validated EIC-Opticks vs pure Geant4 for a simplified pfRICH geometry
  - Shooting 50k electrons to the same place
  - 50k 5 MeV e$^-$ yields:
    - Geant4 hits : 13’840’638 ± 3720
    - Opticks hits : 13’840’566 ± 3720
- Opticks was validated by JUNO, Crab, LHCb



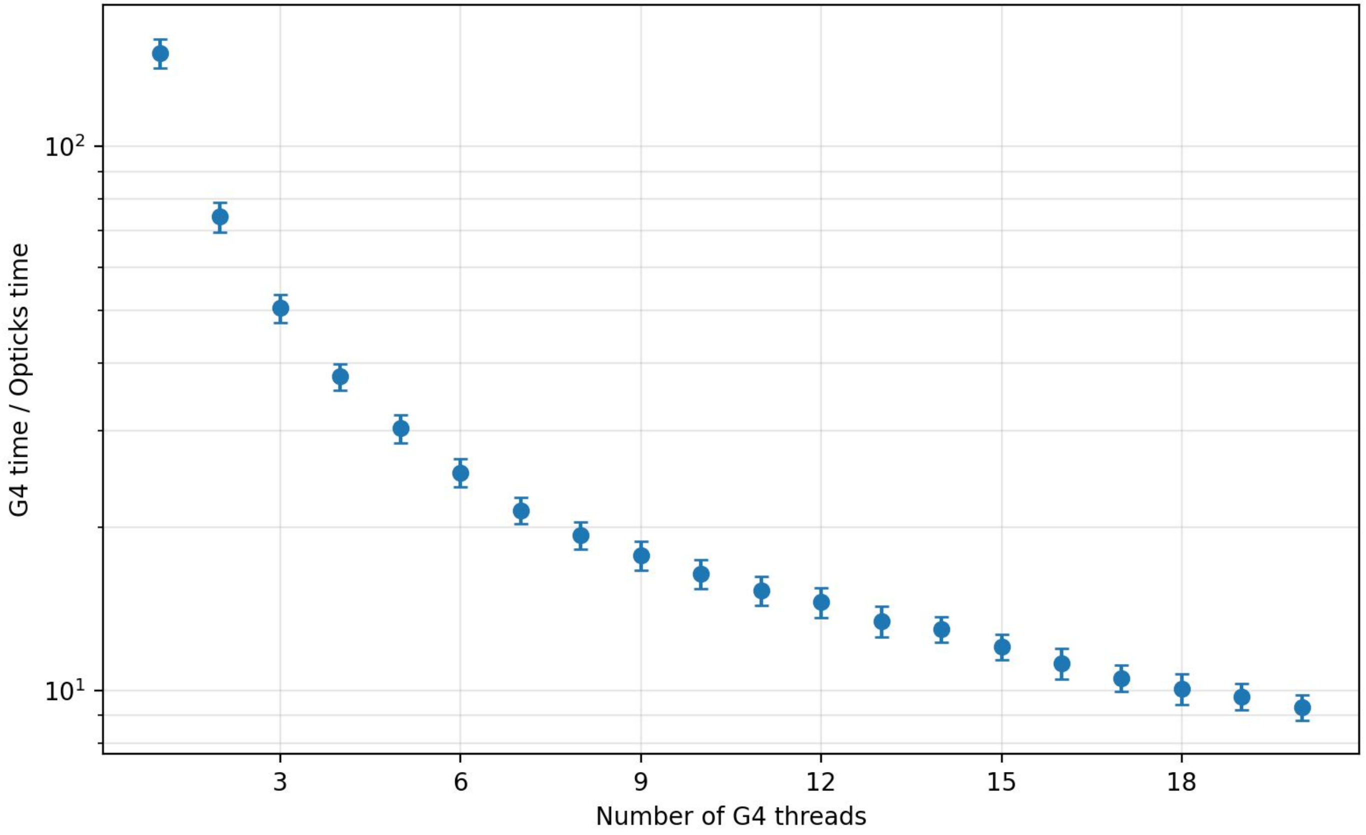
Imaging Barrel EM Calorimeter
Electron Direction Tracking
1.7T Superconducting Solenoid
AC-LGAD TOF
Forward Calorimetry (EM and Hadronic)
Backward Calorimetry
Dual-radiator RICH
High-performance DIRC
Endcap Electromagnetic Calorimeter
Proximity-focused RICH
Hadron Direction Tracking
Barrel Hadronic Calorimeter
Electrons
Hadrons





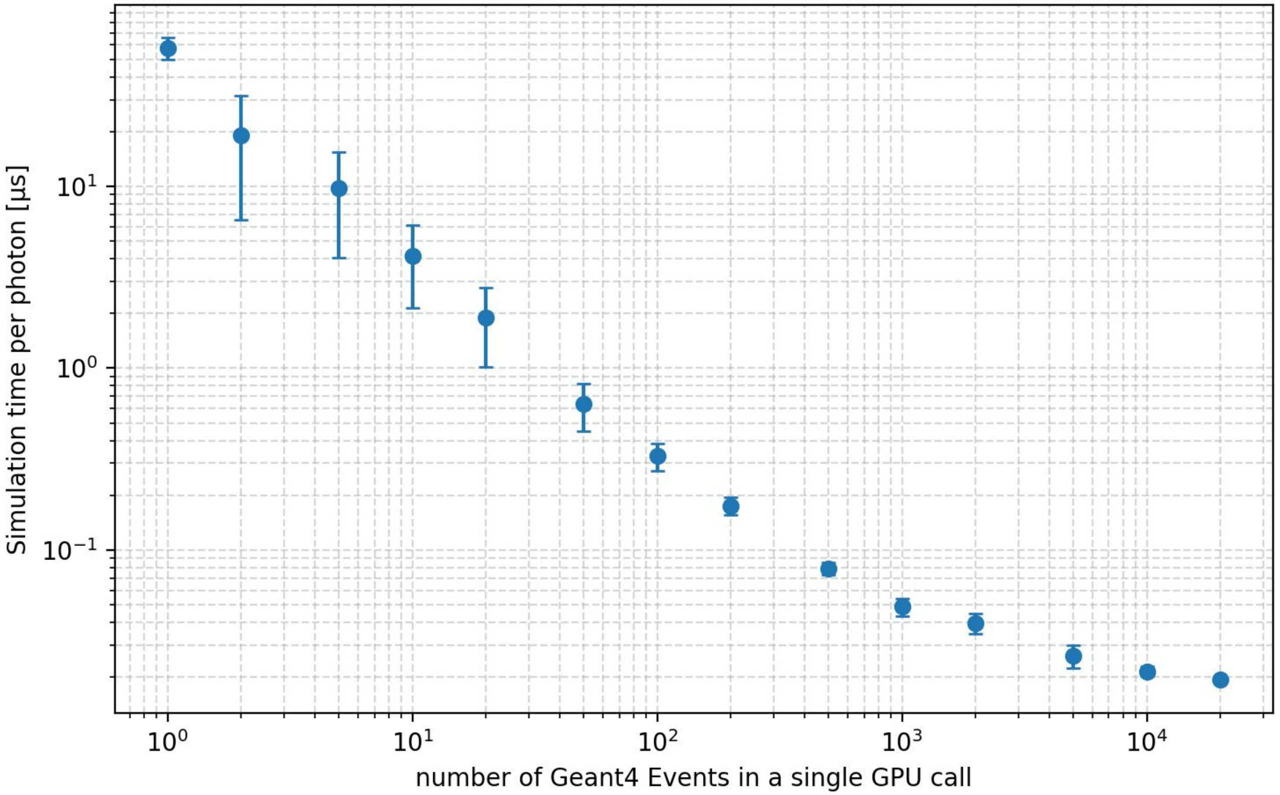
Performance

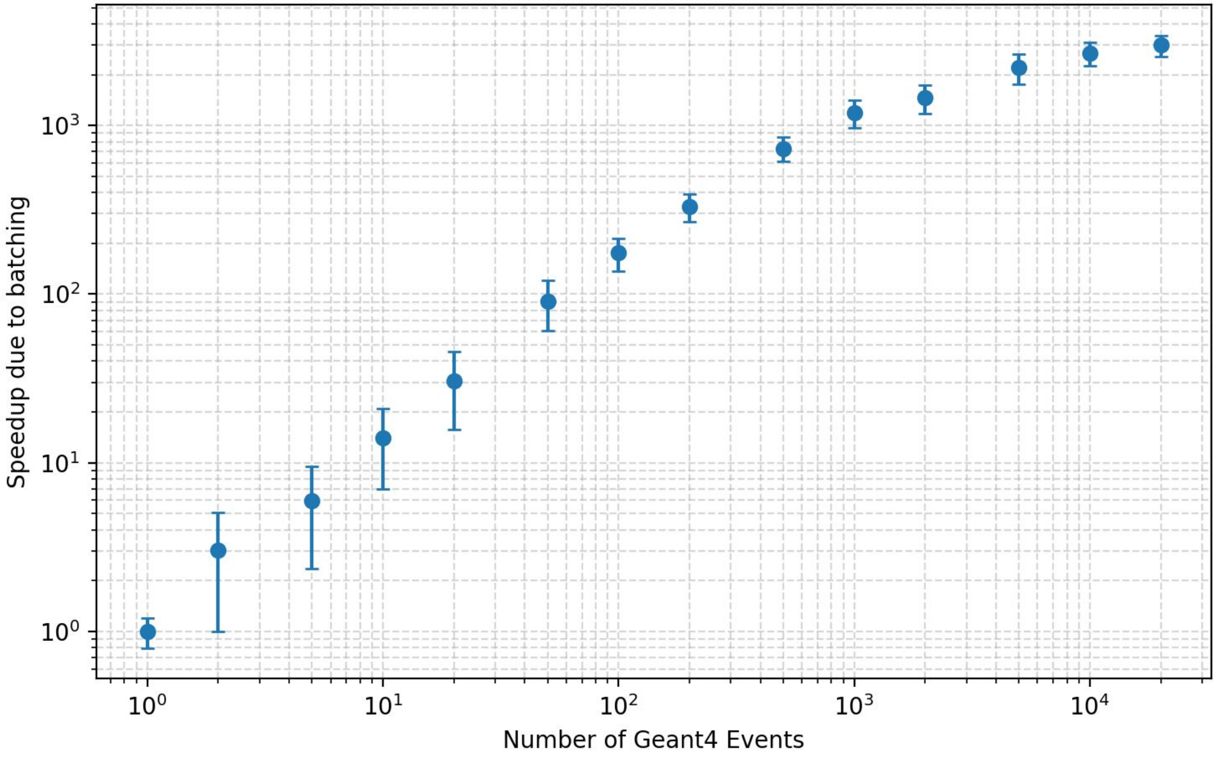
- Ran on our GPU server
- NVIDIA GeForce RTX 4090
- Intel(R) Xeon(R) w7-3445
- Time measurement for EIC-Opticks included:
  - Data movement from CPU -> GPU
  - Ray tracing
  - Copying results from GPU -> CPU
- 50k 5 MeV/c e^- → ~ 50M photons
  - EIC-Opticks vs Single thread G4 CPU
    148±9x speedup
  - 20 threads MT G4 CPU vs EIC-Opticks
    9.3±0.5x speedup



Event batching in EIC-Opticks

- We introduced event batching in EIC-Opticks
- Overhead, to start GPU simulation, run and return results
  - For low energy events too high
- In the case of pfRICH batching 50k event speeds up GPU simulation time 3000x

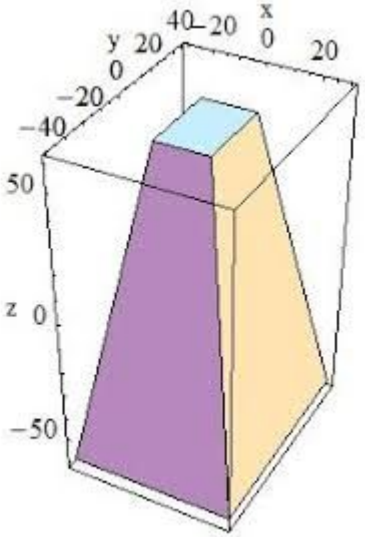




Brookhaven National Laboratory

Development plans

- Implement wavelength shifting process in the GPU code
- Implement a “scheduler” that would balance CPU, GPU node launches on clusters
- Implement support for trapezoid volumes into EIC-Opticks
- Investigate how to add support for triangulated volumes



Summary

- Optical photons can dominate Geant4 simulation time, limiting detailed and large-scale studies
- EIC-Opticks: OptiX-based GPU propagation integrated with Geant4, with event batching and reflection / ray-offset tuning
- GPU results agree with standard Geant4
- Large end-to-end speedups vs. CPU Geant4, including data transfer
- NVIDIA/OptiX-only, limited geometry primitives, and incomplete physics (e.g. WLS)
- Plan: Extend physics and geometry support, improve tasking/multi-GPU use
- In few weeks everybody can test eic-opticks in eic_cuda container, no need to install anything (except nvidia drivers..)

Thank you for your attention!