Fermilab **ENERGY** Office of Science



Optical Photon Mediator for Vectorization/GPU/FastSim

Soon Yung Jun Geant4 Collaboration Meeting August 26-30, 2018, Lund, Sweden

Not a proposal, but a discussion



Optical Photon Processes

- Optical photons are produced copiously when charged particles go through scintillation material
 - N γ /(MeV energy loss) where N γ > O(10⁴)
 - Scintillation, Cherenkov radiation, ...
- Full optical photon simulation is prohibitively expensive on CPUs for a large scintillation-based detector: example

CPU[sec]/Event for 1GeV with a 1mx1mx1m LAr Fiducial Volume:

Particle	optical photons OFF	optical photons ON	ON/OFF ratio
e-	3.25 ± 0.02	3302.40 ± 8.36	1016 ± 6
mu-	1.11 ± 0.01	930.54 ± 6.07	838 ± 9
pi-	1.11 ± 0.01	1385.10 ± 9.73	1250 ± 12
proton	1.03 ± 0.01	1418.10 ± 9.92	1370 ± 14

• A good candidate to acceleration

🔁 Fermilab

Optical Photon Simulation: Different Approaches

- Optical photons simulation on GPU
 - Blyth S 2017 Opticks : GPU Optical Photon Simulation for Particle Physics using NVIDIA® OptiX[™] J. Phys.: Conf. Series 898 042001)
 - Interfaces and CUDA kernels
- Fast Simulation
 - Liquid Ar Time Projection Chamber (LAr-TPC) experiments: use lookup tables to roughly estimate the arrival time in LArSoft/Dune-Far detector simulation
- Vectorization: SIMD kernels (processing N photons together)
- Require a single entry point (G4VFastSimulation), but repeating essentially Scintillaton::PostStep up to the point looping over the number of photons produced

🛠 Fermilah

Preemptive Vectorization Bypath

• Call the SIMD/GPU kernel or FastSim at the right point



- Minimize memory transaction before vectorization/launching cuda kernel
 - Kernel A = Generate N γ + Kernel B (optical processes)
 - Avoid stacking secondary tracks (store) and load again
 - Size of tracks * N γ + huge memory transaction

🛟 Fermilab

Optical Photon Mediator and Related Topics

- Delaying optical photon generation to the right stage in the full simulation chain requires
 - Additional interfaces in Geant4 kernel/process
 - Introduction of a mediator
- Optical photon mediator
 - Position (vertex), time,
 - Number of photons to generate, integrated scintillation energy
- Related topics
 - Transportation: Unified tracking mechanism for exotic particles (Session 3B, Makoto)
 - Biasing (Session 4A, Marc)
 - Smart tracking
 - Geant4 task model and GPU (Session 7A, Jonathan)
 - GeantV (Session 7A, Andrei)

🌫 Fermilah

Discussion

- Can we cleanly introduce this type of object (i.e, an envisaged optical photon mediator as an example, or its variants) in the standard Geant4 stepping/tracking?
 - Avoid extra introduction of interfaces or specialization
 - Beneficial for both vectorization and GPU tasks (i.e, increase overall FLOPS/(memory transactions) or minimizing data move)