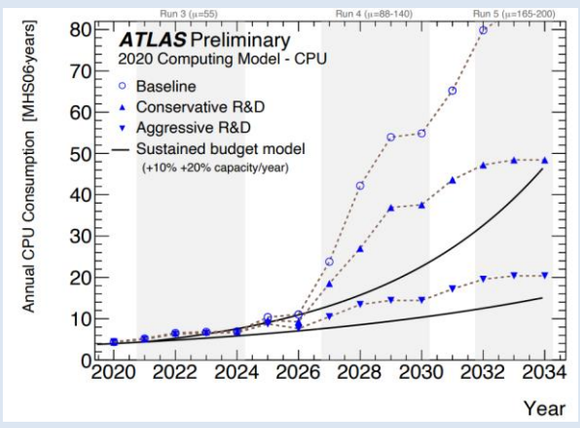
Accelerating Fast Calorimeter Simulation with CUDA in ATLAS

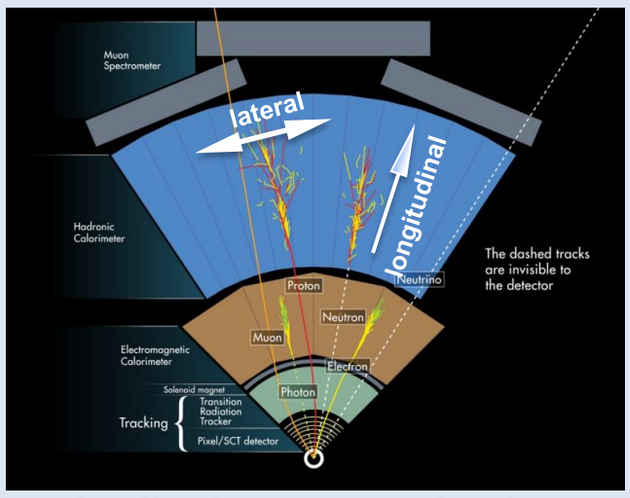
Introduction

- In ATLAS [1], very large samples of simulated events are needed
- The Simulation with Geant4 (G4) is very CPU intensive especially for the liquid argon calorimeter
- A Parameterization-based fast simulation (AtlFast2) [2] is developed to replace Geant4 simulation
- ATLAS employs GPUs to accelerate AtlFast2 further by parallelizing the simulation at the particle level

Parameterization-based fast calorimeter simulation

- Event simulation is crucial to the ATLAS physics program, and the sensitivity of many physics analyses is limited by the statistics of simulated events
- The rapid increase in luminosity of LHC requires larger number of simulated events
- ~90% of simulation time is devoted to the liquid argon calorimeter, thus a fast simulation was developed





In **AtlFast2**:

- Instead of simulating particle interactions, parameterize the detector response of single particles in the calorimeter using a simplified geometry
- Parametrize the single particle shower development in longitudinal (energy) and lateral (shape) directions

Lateral shape parameterization:

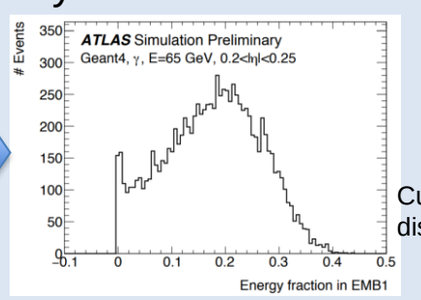
- Use the average shape for each particle type/eta/energy/layer/PCA bin
- Energy is deposited using N_{hits} of equal energy for photon/electron and weighted energy for pion

Longitudinal parameterization:

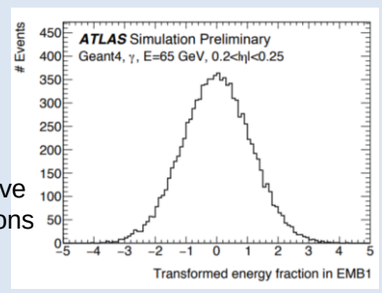
De-correlate the energies between layers

- Single particles are classified based on the depth
- Principal Component Analysis (PCA) is performed on each layer

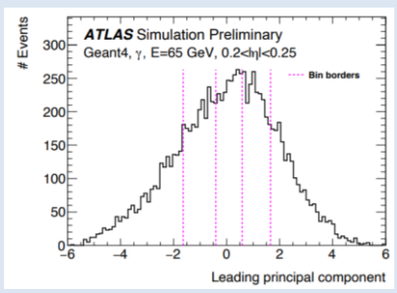
G4 simulated particles

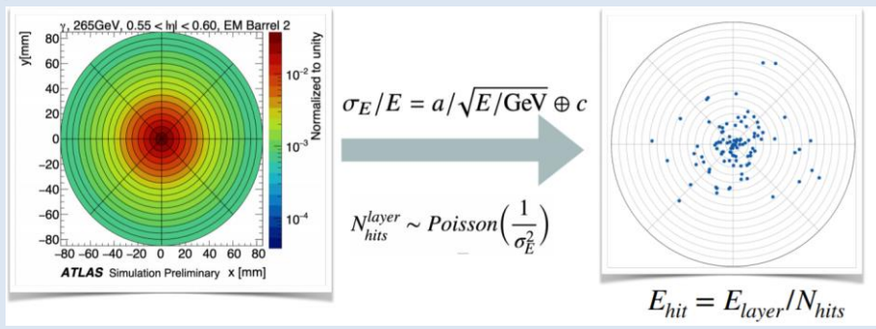


Cumulative distributions



PCA

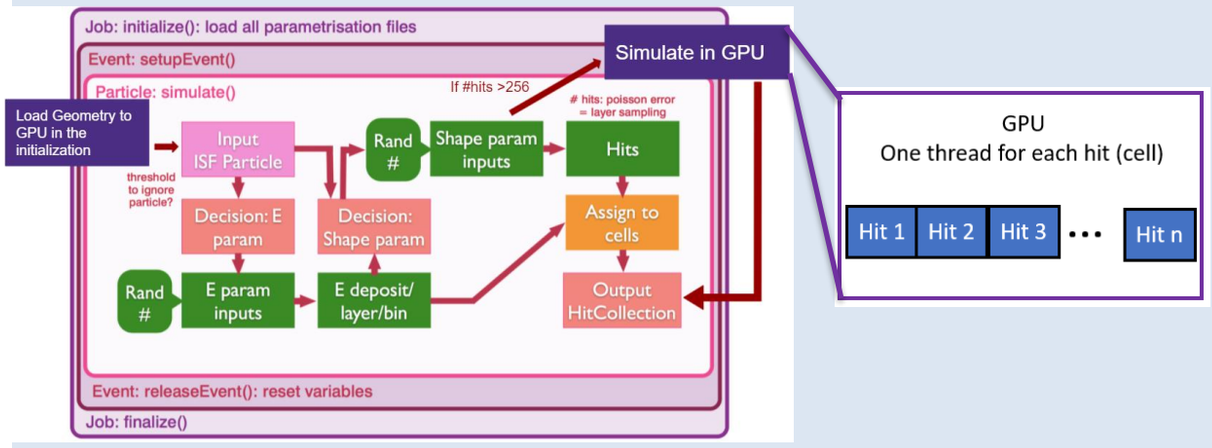




- **Good modeling for all reconstructed observables compared to G4**
- **A factor of O(500) speed up for calorimeter**

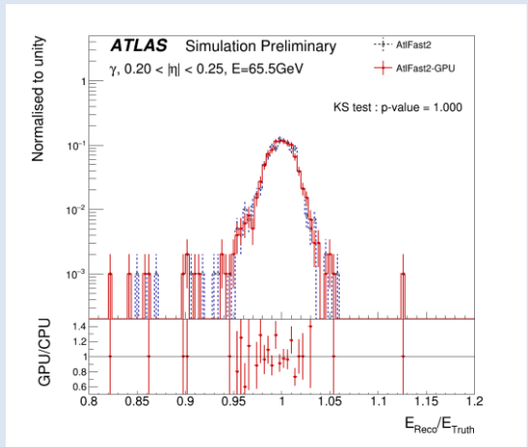
Accelerate with GPU

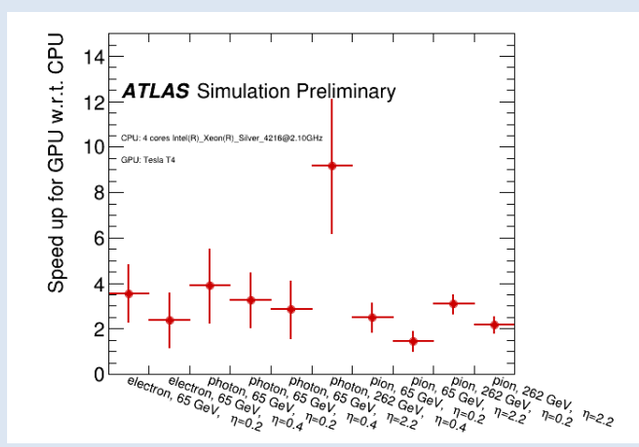
- Calorimeter has massive inherent parallelism
  - Many independent cells
- A previous study [3] has observed very good speed up (up to a factor of 60 for the event loop and 4.6 for the total job) when porting the simulation to GPU
- Here is the reintegration and modernization of the CUDA code to Athena, the ATLAS offline software framework [4]



- The event data model is rewritten for geometry/identifier/histograms/functions with CUDA
- The random numbers are generated on GPU with cuRAND
  - 3 per hit, thousands of hits per particle
- Geometry is transferred to GPU once per job
- 3 kernels for the simulations of hits
  - Memory initialization, simulation, reduction
  - Launch latency limited

Preliminary Validation and Performance with GPU





Validated with simulation of single particle process

- Good consistency in the total energy response w.r.t. the truth energy (left) is observed for fast simulation on CPU and GPU.

Performance is studied with several simulations for different particles without pile-up

- In general, a speed up of O(3) for the simulation of one particle in calorimeter on GPU (right) (errorbar means the standard deviation from a series of tests)
- The speed up is more significant for the particles with more associated hits

References

- 1, ATLAS Collaboration, JINST 3 (2008) S09003.
- 2, ATLAS Collaboration, ATL-SOFT-PUB-2018-002, (2018).
- 3, Z. Dong, et al., arXiv:2103.14737(2021)
- 4, ATLAS Collaboration, (2019), Athena (22.0.1), Zenodo 10.5281

Ke Li (University of Washington)
on behalf of ATLAS Collaboration

ACAT 2021 poster session