Optimizing GEANT for Monte Carlo simulation challenge for CMS Detector at the HL-LHC

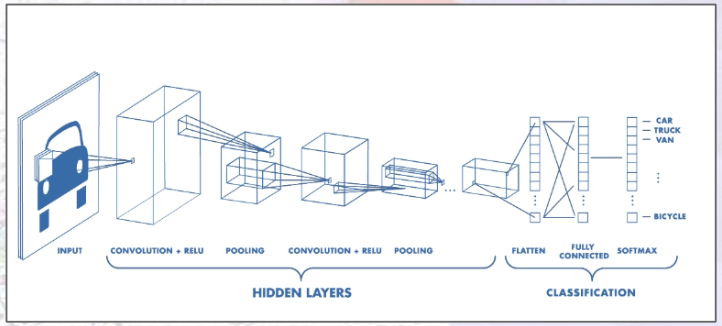
Harold D. Guerrero De La Cruz
on behalf of the CMS Collaboration
Physics Department
University of Puerto Rico - Mayaguez

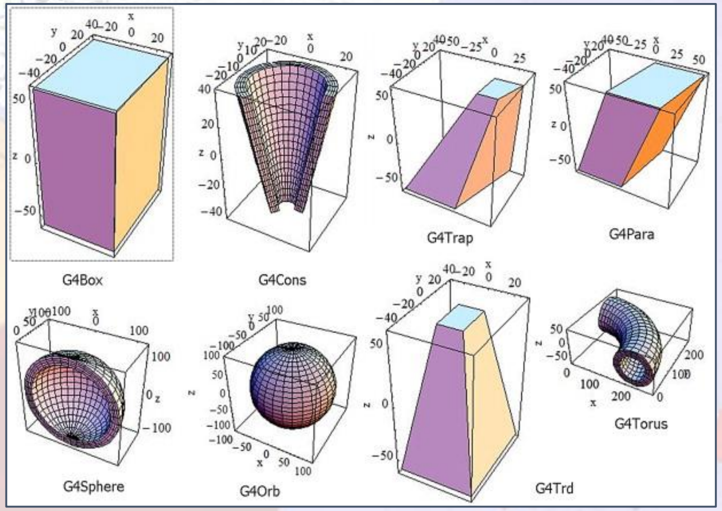
CMS Detector Simulation

- As we move forward we need to optimize our computing resources as we collect more data. One way is to optimize our simulation. Simulation employs models of various types of particle interacting with different materials and uses the Geant4 software

- Several efforts have lead to simulations about 4-5 times faster than the baseline (current detector)
  - Improvements due to multithreading and hyperthreading processes
  - New algorithms using fewer magnetic field evaluations
  - New library (VecGeom) for detector geometry that supports vectorization and new computing architectures
  - Note - Physics fidelity is crucial or else quoted speedups are meaningless

- Besides Full Simulation (Geant4 based) other simple fast simulation engines are used for studies like Fastsim and Delphes
- Machine Learning (ML) offers higher accuracy than “simple” fast simulations to produce faster results than Geant4 but may need large training datasets and training time
- Using ML we can optimize Geant4 parameters for precision vs speed
  - Run a fast, but less accurate, Geant4 simulation by changing parameters
  - Recover eliminated parameter values by using a regression-based ML approach with convolutional neural networks (CNNs)
  - Analyze if the recovered data offers better and more accurate information





Optimizing GEANT

General Objective

- Is to optimize the simulation step in the Geant4 model used.

Simulation Tool

- GEANT4 10.4.3-nmpfii2
- We control the amount of jobs to get a more accurate processing time.

Parameters studied

- **Russian Roulette/Probability**: algorithm randomly drops N-1 low-energy particles, adding their energy to the Nth particle that is kept. It has two parameters: the probability of dropping a particle, and the maximum energy limit for dropped particles. These two parameters can be specified for each particle type and each detector region.
- **Magnetic Fields:** these parameters control when a simpler algorithm (faster, but less accurate) is used. It has three parameters: one that decomposes this curved path into linear chord segments, another that adjusts the precision of the strings and the maximum energy threshold.
- **Production Cuts:** It is the maximum distance; that particles would deposit all their energy in less than the maximum distance (in current material) will not be able to create secondary particles

Time Variation in Parameters (1)

- The nominal time was calculated for various objects and than compared to changes in the value of the variables used.
- It was found that the different Energy Limit variables gave the greatest change in the time of simulation vs the nominal time.

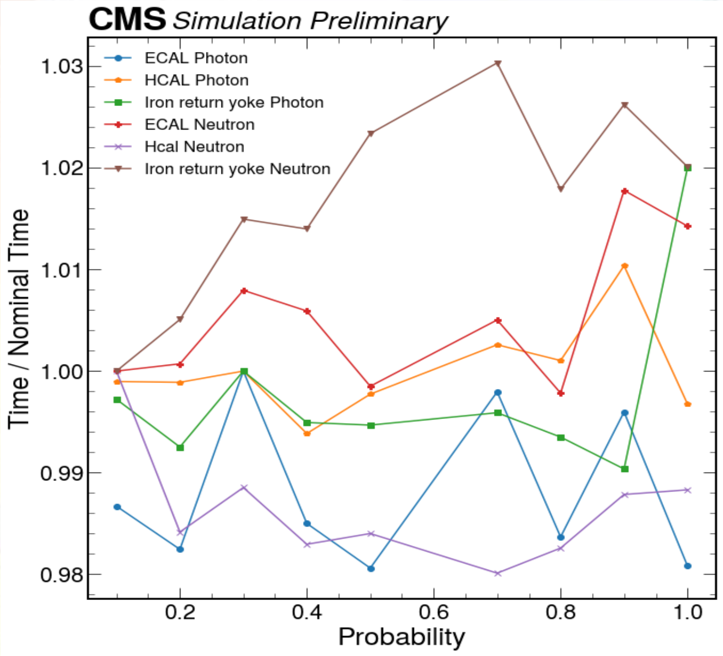


fig 1. Time variation in probability parameters

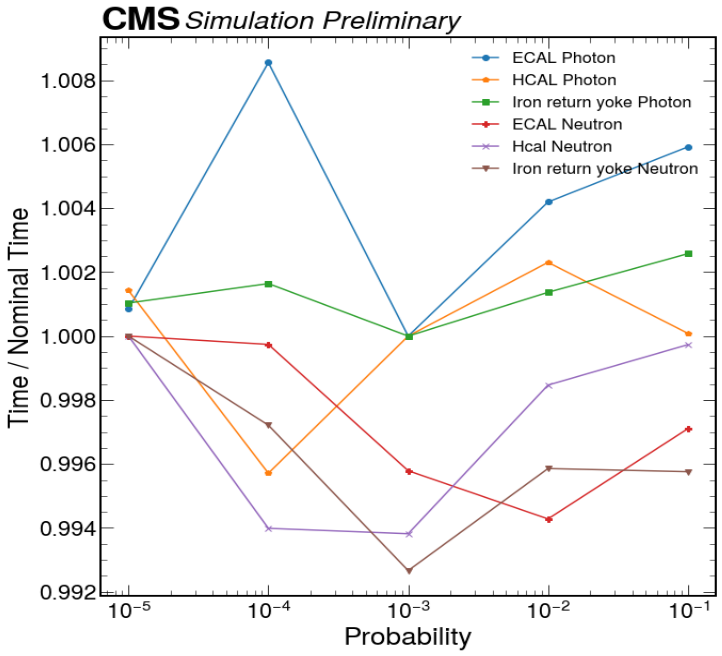


fig 2. Time variation in probability parameters with logarithmic scale on the x-axis

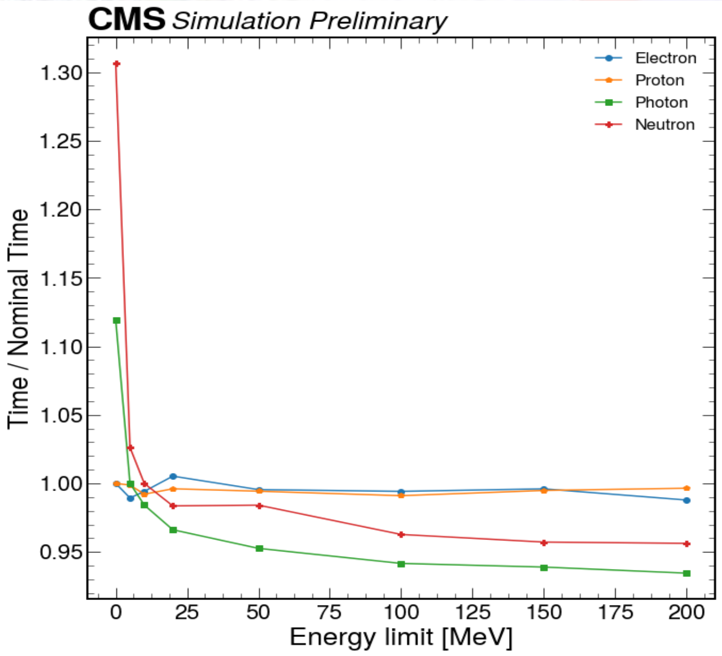


fig 3. Time variation in energy parameters

Time Variation in Parameters (2)

- The nominal time was calculated for various objects and than compared to changes in the value of the variables used.
- It was found that the Production cut gave the greatest change in the time of simulation vs the nominal time.

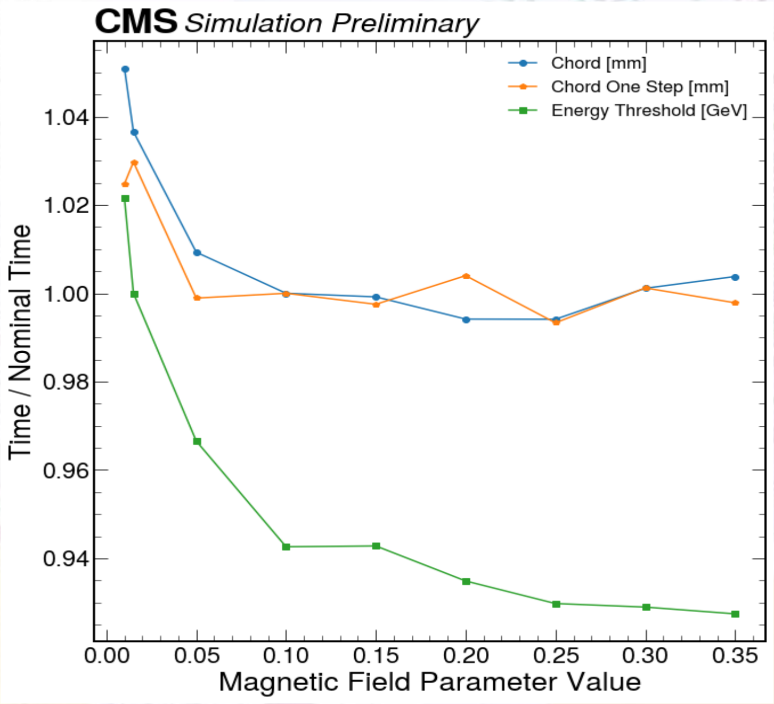


fig 4. Time variation in magnetic parameters

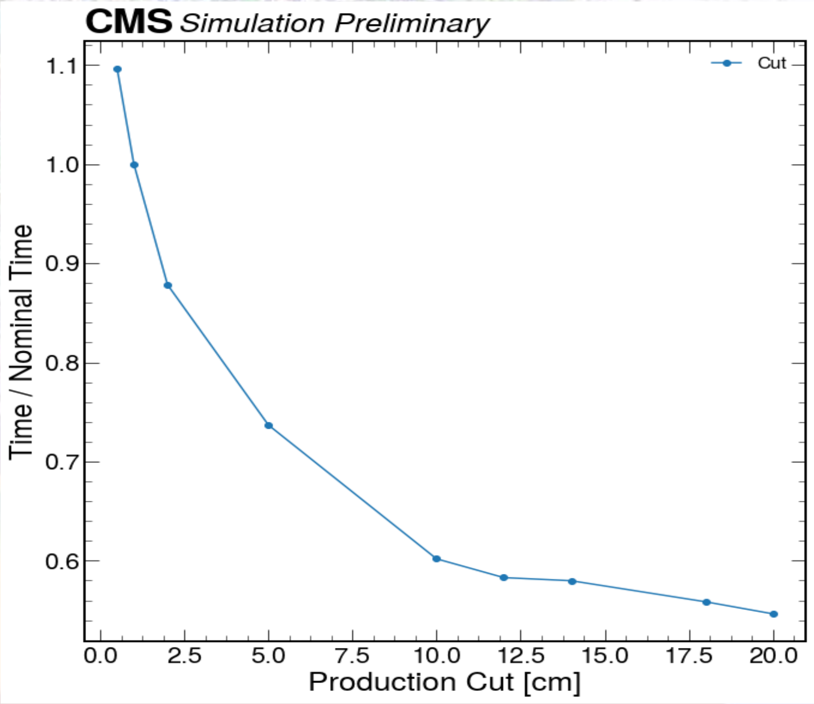


fig 5. Time variation in cut parameters

Conclusions

- The most promising parameters are *Photon, Neutron, Energy Threshold* and *Production cut*
- Moving towards combining parameters like the *Photon* and *Neutron* to see what the output time and simulation is
- Checking the change in accuracy as well
- Looking into the aspects of automated learning using CNNs
- Thanks Scarlet Norberg (UPRM), Kevin Pedro (FNAL)

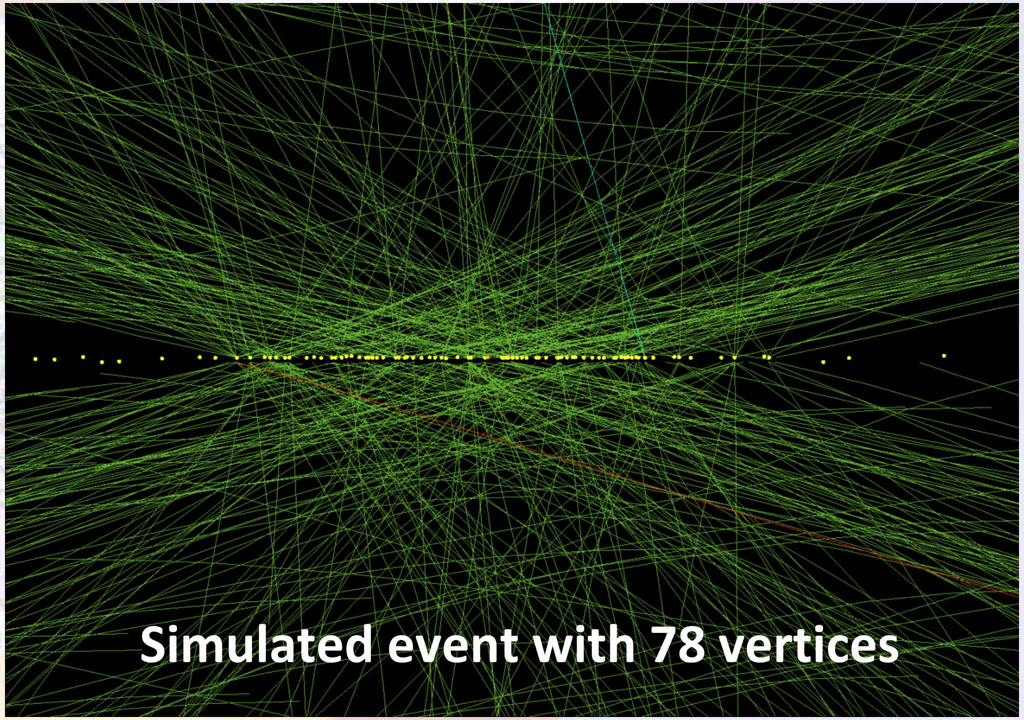
Thank You !!

Backup

CMS Detector at the HL-LHC

Experimental Challenges for CMS Phase-2 Upgrade for the HL-LHC

- Luminosity - 7.5×10^{34} cm^{-2} s^{-1}, Pileup - 200, L1 Trigger Rate - 750 kHz
- Upgraded Sub-detectors - e.g. HGCal (6 million channels), Inner Tracker (2 billion channels)
- Increase in geometry channels 2.1 million to 21.9 million elements
- Increases in channel counts and geometry elements translates into an increase in CPU usage for the simulation:
  - must provide more events and more accuracy, with a more complicated geometry using a smaller fraction of the CPU budget
  - serious efforts to improve the speed and efficiency of the detector simulation


Simulated event with 78 vertices