Table of Contents

SimHits, SimTracks, and SimVertices in FastSim

SimTrack and SimVertex collections

The SimTrack and SimVertex collections summarise the trajectories of particles through the detector.

One part of the SimTrack and SimVertex collection comes from the particles provided by the generator. For each generator particle, the origin vertex is stored as a SimVertex, and the properties of the particle at the origin vertex are stored as SimTrack.

- These SimVertices and SimTracks are produced in the function `FBaseSimEvent::fill(const std::vector& simTracks,const std::vector& simVertices)` (link to code☑)
- which is caled inside `FamosManager::reconstruct` (link to code☑)

The remaining part of the SimTrack and SimVertex collections represent decays of particles and interactions between particles and detector material. The following interactions lead to creation of SimVertices and SimTracks:

- **bremsstrahlung:** the interaction vertex is stored as SimVertex, the photons are stored as SimTracks, the new state of the electron is **not** stored as SimTrack see MaterialEffects::interact☑ and BremsstrahlungSimulator::compute☑

- **photon conversion:** (pair production): the interaction vertex is stored as SimVertex, the electron positron pair is stored as two Simtracks see MaterialEffects::interact☑ and PairProductionSimulator::compute☑

- **nuclear interactions:** two kinds of nuclear interactions are stored, elastic and inelastic ones
  - ♦ see MaterialEffects::interact☑ and NuclearInteractionSimulator::compute☑
  - ♦ elastic nuclear interactions: only if the angle between the intial and the final state particle is large enough, the interaction vertex is stored as SimVertex and the final state particle as SimTrack (link to code☑)
  - ♦ inelastic nuclear interactions: interaction vertex is stored as SimVertex final state particles are stored as SimTracks
  - ♦ if the particle is stopped, the end vertex is stored

- **decays:** the decay vertex is stored as SimVertex and the decay products as SimTracks see PythiaDecays::particleDaughters☑

Filtering: only those particles for which KineParticleFilter::isOKForMe☑ is true (tested at the origin vertex) are considered for simulation and may be stored as SimTracks.

SimHits in tracker

* objects of class `PSimHit` (link to code⧉)

Useful variables

* `int PSimHit::trackId()` returns index of `SimTrack` in `SimTrackCollection` that produced the `SimHit` * `int PSimHit::particleType()` returns the Pdg id number of the `SimTrack` that produced the `SimHit` * `float PSimHit::energyLoss()` returns the energy lost by the `SimTrack` through ionisation in the silicon module at which the hit resides. This variable is used, during digitization, to determine the strength of the signal in the silicon sensors.

-- LukasVanelderen - 2015-04-21

This topic: CMSPublic > SWGuideFastSimSimHitsAndTracks
Topic revision: r4 - 2015-04-27 - LukasVanelderen