Refactored Simulation with larg4

Wenqiang Gu (BNL), David Rivera (UPenn)

for the protoDUNE-SP simulation task force

Introduction

- The “larsim/LArG4” (legacy) module has been widely used in many larsoft based LArTPC experiments
- A recent refactorization, “larg4” module, has been proposed and applied in ProtoDUNE-SP
- In this talk, we will briefly review the refactored simulation flow as well as the recent update and performance
  - Improved infrastructure
  - Improved data-MC agreement for calorimetry measurement
  - Application in DUNE FD

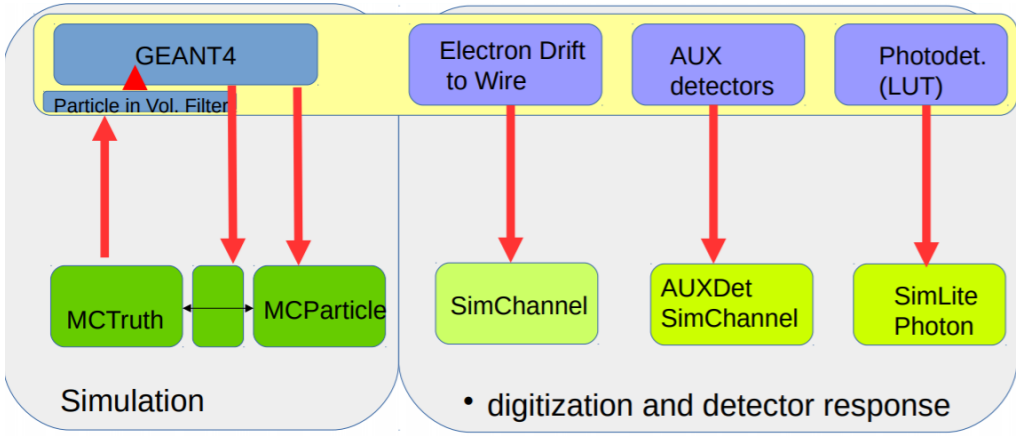
Refactored larg4

- larsim/LArG4, aka **legacy**
  - Based on nug4/G4Base
    - Heavy
  - All-in-1 simulation
    - Particle interaction, ionization, electron drift, optical simulation
  - ConfigurablePhysicsList.h
    - Needs to compile
  - Incomplete and obsolete physics list
  - TPC using SteppingAction
- **Refactored** larg4
  - Based on artg4tk
    - Pure: art + geant4
  - Only simulate particle interaction
    - Plugin modules for the other stages of simulation
  - Extensible physics from FHiCL
    - Neutron killer, step limiter, various EM physics options
  - More recent reference physics list
  - TPC using sensitive volume

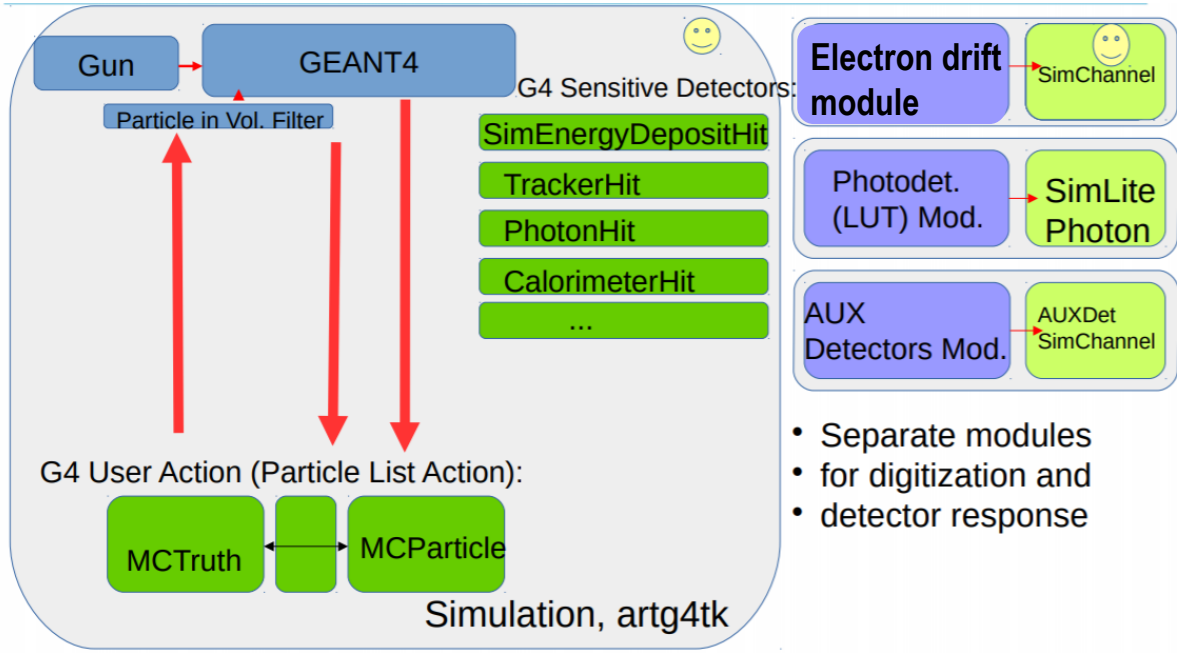
More about the refactored larg4:
[Hanz Wenzel] https://indico.fnal.gov/event/18681/session/6/contribution/61/material/slides/0.pdf

Refactored larg4

Legacy (larsim/LArG4)



Refactored (larg4)



More about the refactored larg4:
[Hanz Wenzel] https://indico.fnal.gov/event/18681/session/6/contribution/61/material/slides/0.pdf
[David Rivera] https://indico.fnal.gov/event/21037/contribution/2/material/slides/0.pdf

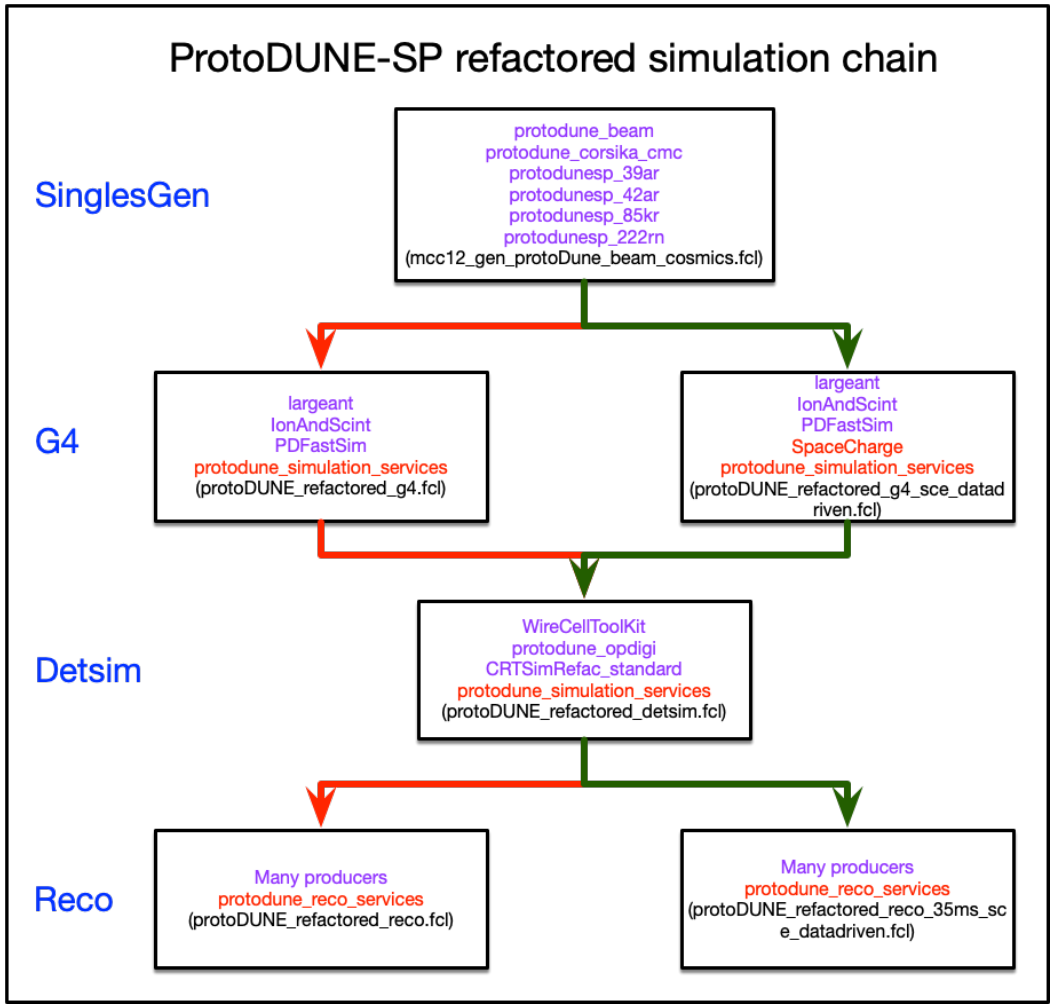
Extensible physics list

Using the Geant4 physics list factory :

- Access to all reference physics lists
- Can create a custom physics and register it to make it available
  - Access to the source code
- Can choose among the various options with fhicl parameters

```
Base G4VModularPhysicsLists in G4PhysListRegistry are:
 [  0]  "FTFP_BERT"
 [  1]  "FTFP_BERT_ATL"
 [  2]  "FTFP_BERT_HP"
 [  3]  "FTFP_BERT_TRV"
 [  4]  "FTFP_INCLXX"
 [  5]  "FTFP_INCLXX_HP"
 [  6]  "FTF_BIC"
 [  7]  "G4GenericPhysicsList"
 [  8]  "LBE"
 [  9]  "MyQGSP_BERT_HP"
 [ 10]  "NuBeam"
 [ 11]  "QBBC"
 [ 12]  "QGSP_BERT"
 [ 13]  "QGSP_BERT_HP"
 [ 14]  "QGSP_BIC"
 [ 15]  "QGSP_BIC_AllHP"
 [ 16]  "QGSP_BIC_HP"
 [ 17]  "QGSP_FTFP_BERT"
 [ 18]  "QGSP_INCLXX"
 [ 19]  "QGSP_INCLXX_HP"
 [ 20]  "QGS_BIC"
 [ 21]  "Shielding"
 [ 22]  "ShieldingLEND"
 [ 23]  "ShieldingM"
Replacement mappings in G4PhysListRegistry are:
           EMV =>    G4EmStandardPhysics_option1
           EMX =>    G4EmStandardPhysics_option2
           EMY =>    G4EmStandardPhysics_option3
           EMZ =>    G4EmStandardPhysics_option4
            GS =>          G4EmStandardPhysicsGS
           LIV =>           G4EmLivermorePhysics
    NEUTRONLIMIT =>           G4NeutronTrackingCut
       OPTICAL =>              G4OpticalPhysics
           PEN =>            G4EmPenelopePhysics
     STEPLIMIT =>           G4StepLimiterPhysics
           _GS =>          G4EmStandardPhysicsGS
Use these mapping to extend physics list; append with _EXT or +EXT
   to use ReplacePhysics( ) ("_") or RegisterPhysics( ) ("+").
```

ProtoDUNE-SP



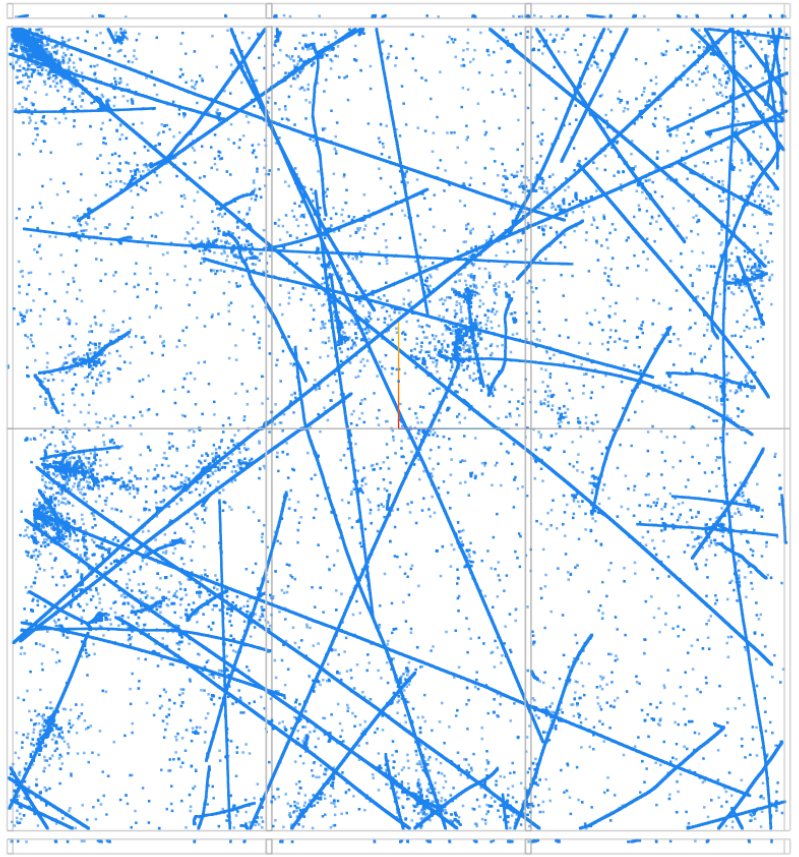
larg4 configuration in ProtoDUNE-SP

- Simultaneously simulate
  - Cosmic rays
  - Beam + beam background
  - Ar39, Ar42, Kr85, and Rn222
- No neutron tracking cut
- Keep EM shower daughters and trajectories

// protoDUNE_refactored_g4.fcl

```
1   MCTruthEventAction: {service_type: "MCTruthEventActionService"}
2   ParticleListAction: {service_type:  "ParticleListActionService"
3                        EnergyCut: 1e-5 # Kinetic Energy cut in [MeV]
4                        storeTrajectories: true
5                        keepGenTrajectories: ["generator"]
6                        keepEMShowerDaughters: true
7                        #NotStoredPhysics: ["muIon"] #if keepEmShowerDaughters is false
8                                                    #this will override the default li
9                                                    #of ignored processes
10                   }
11  }
```

Energy deposits: XZ view



https://www.phy.bnl.gov/wire-cell/bee/set/5633d85e-cefe-442f-8e84-3db7aa227d27/event/0/

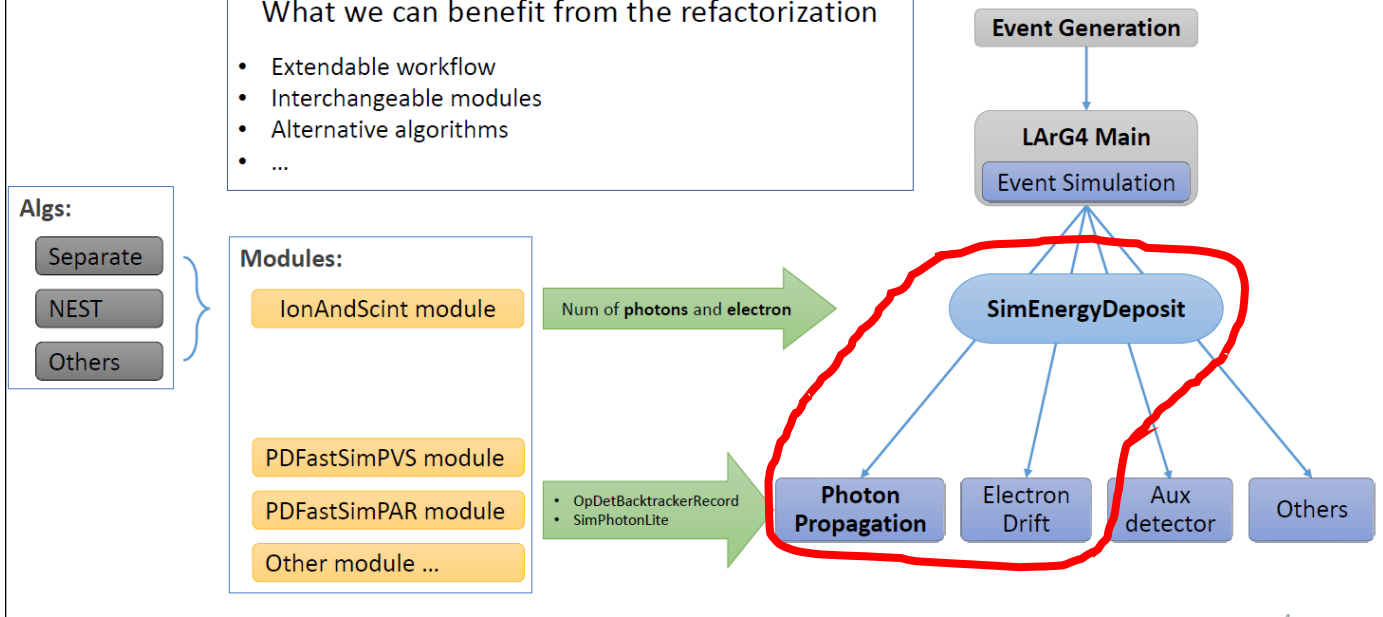
Optical simulation

Simulation workflow – Refactored model

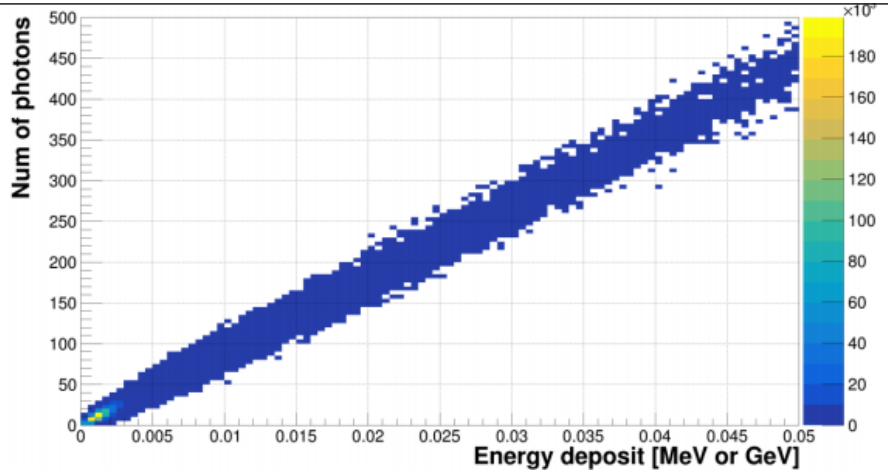
@Muve

What we can benefit from the refactorization

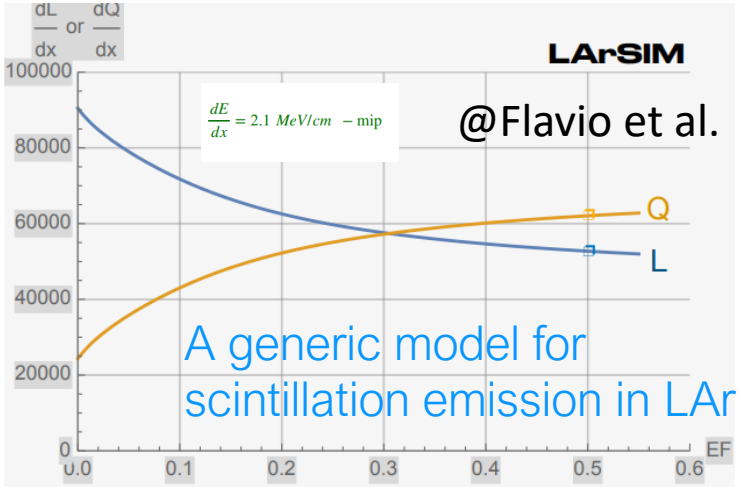
- Extendable workflow
- Interchangeable modules
- Alternative algorithms
- …



Optical photon information saved in **SimEnergyDeposit**



A "Separate" model – mimics the scintillation light prediction from the NEST model



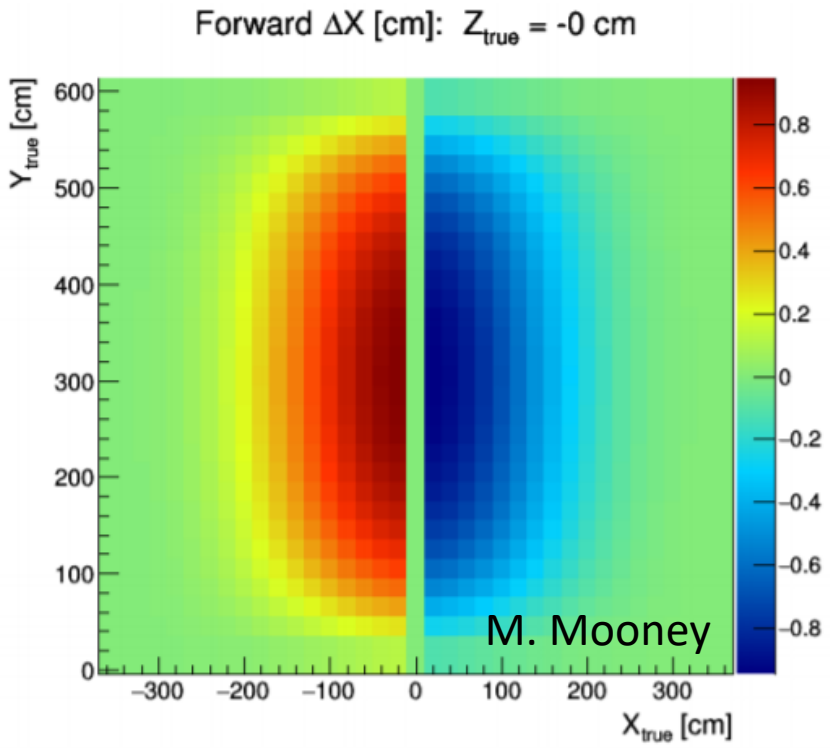
More readings:

[DUNE collaboration meeting, Sept 2019](#) Muve

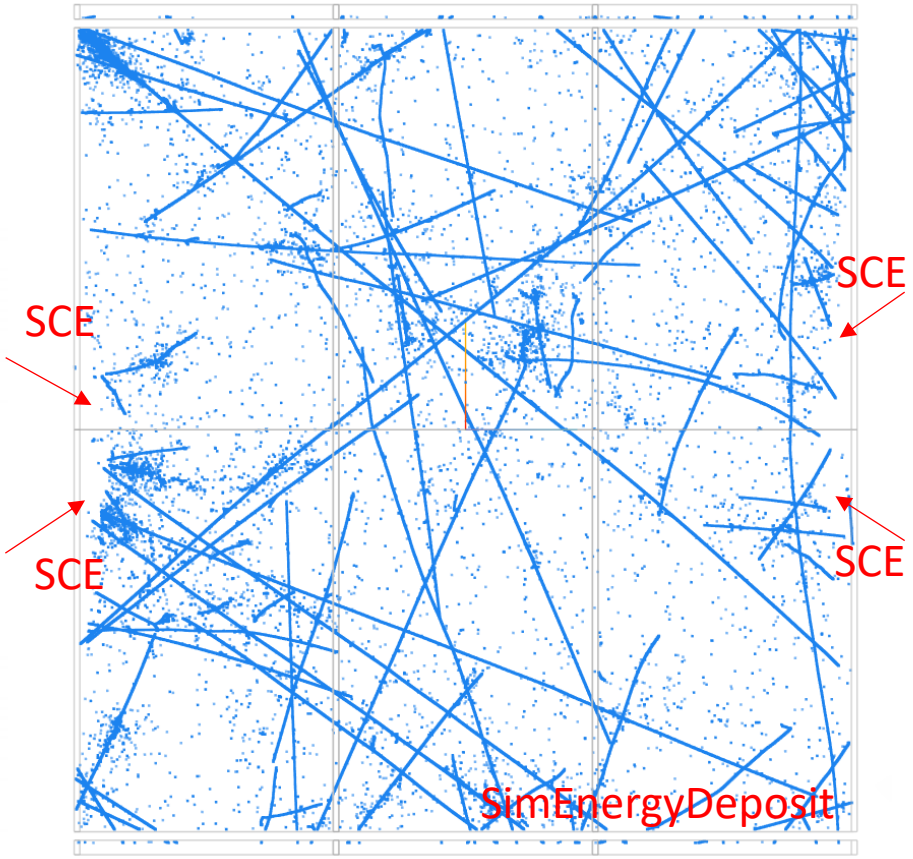
[ProtoDUNE Sim/Reco meeting](#) Flavio (under development)

SCE simulation

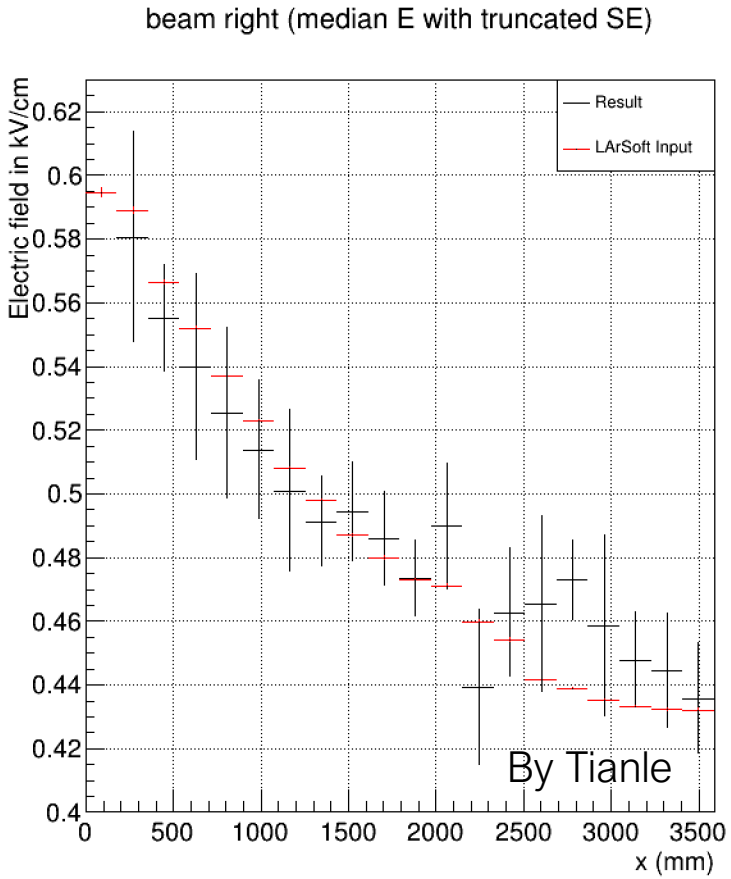
- SCE correction can be applied to the **SimEnergyDeposit** after recombination



SCE correction in the simulation



https://www.phy.bnl.gov/wire-cell/bee/set/5633d85e-cefe-442f-8e84-3db7aa227d27/event/0/

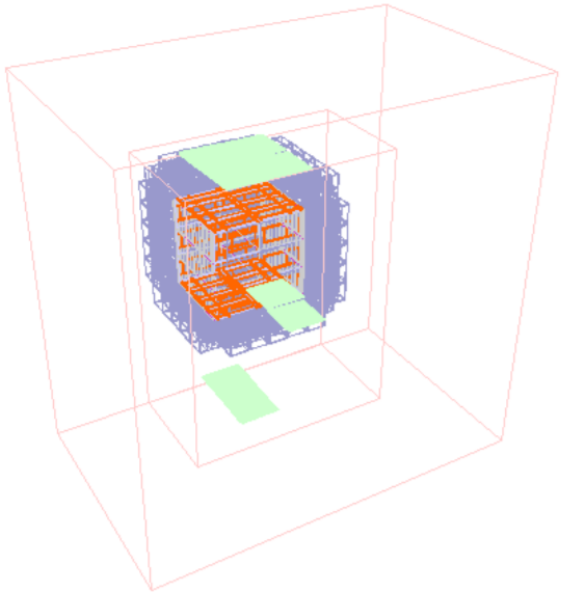


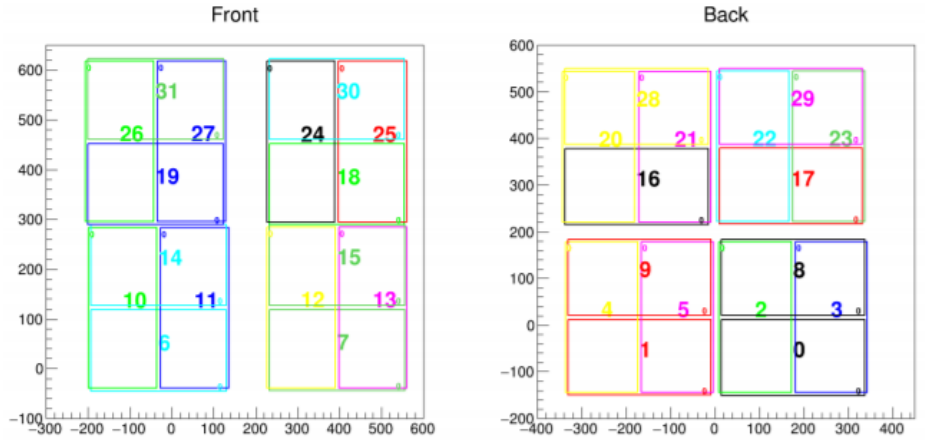
SCE validation: E-field / drift speed

Update in CRT simulation

@ Richie

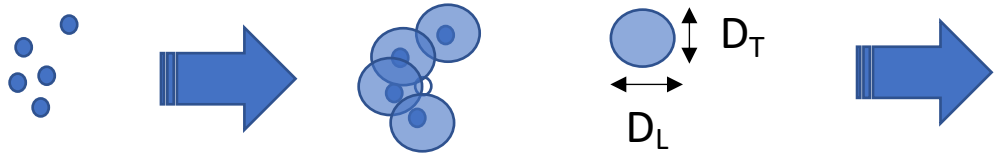
- Moved from **AuxDetSimChannel** to **AuxDetHit**
  - Needs to be clustered for getting entry and exit point
- Mapped GDML module to AuxDet module
  - Copy number in GDML greatly simplified channel mapping





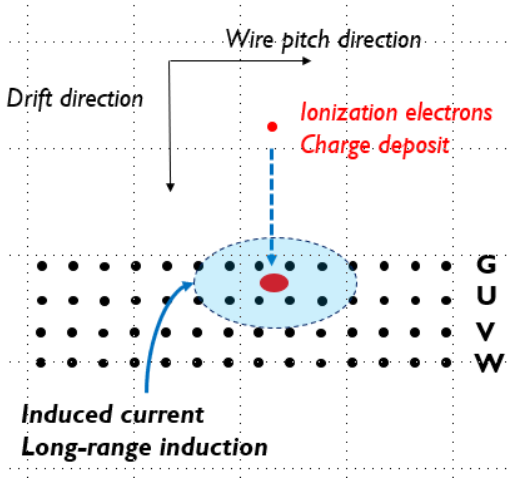
CRT Channel Map

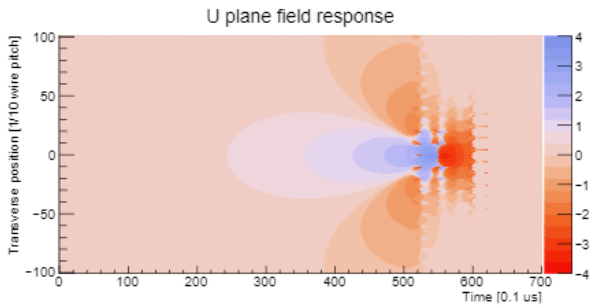
WireCell TPC simulation



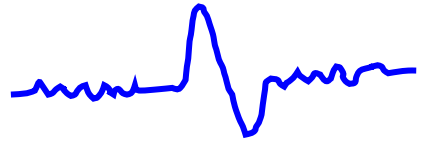
<SimEnegyDeposit>
* x, y, z, t, # of e

- Ionized electron absorption (lifetime in LAr)
- Gaussian random diffusion
  (longitudinal/transverse) σ^2= 2Dt
- Fluctuation in electron absorption





@Wenqiang Gu



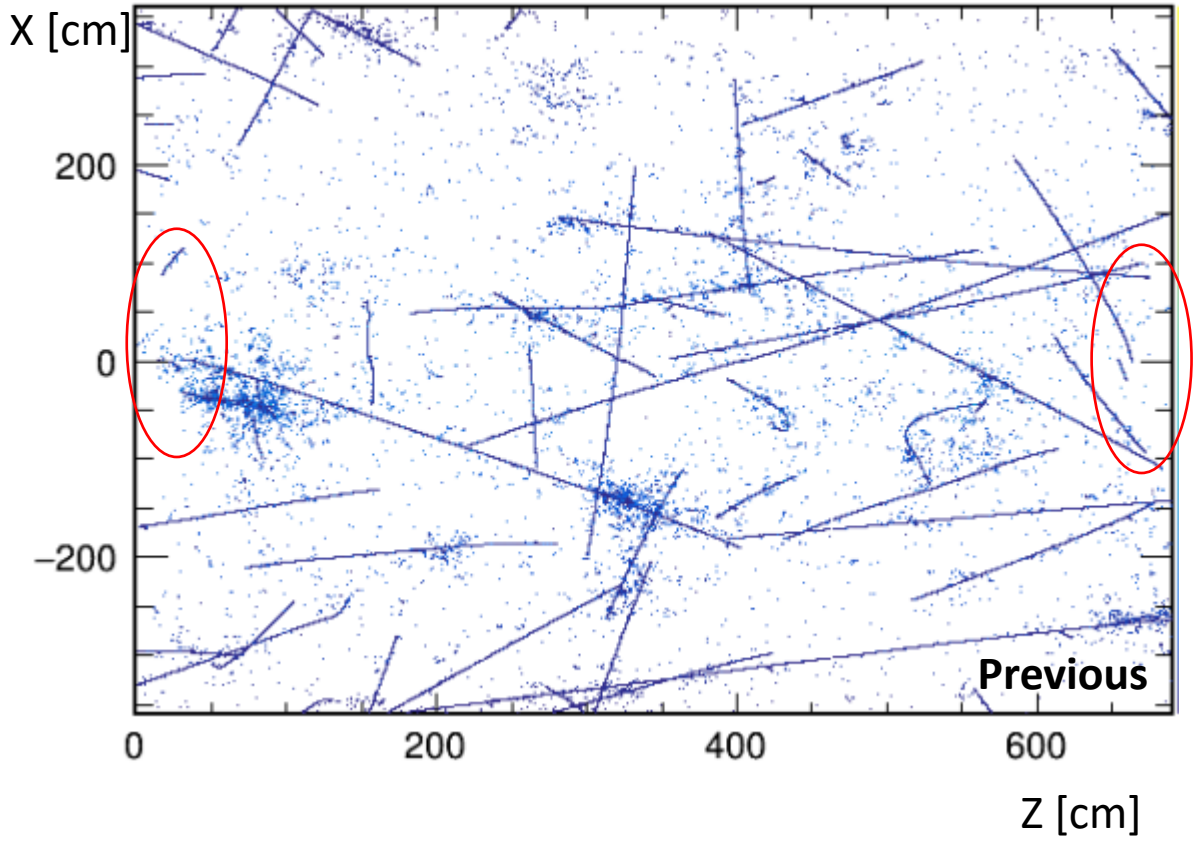
- Electronics response
- Preamp shaping
- AC coupling
- Noise
- Digitizer

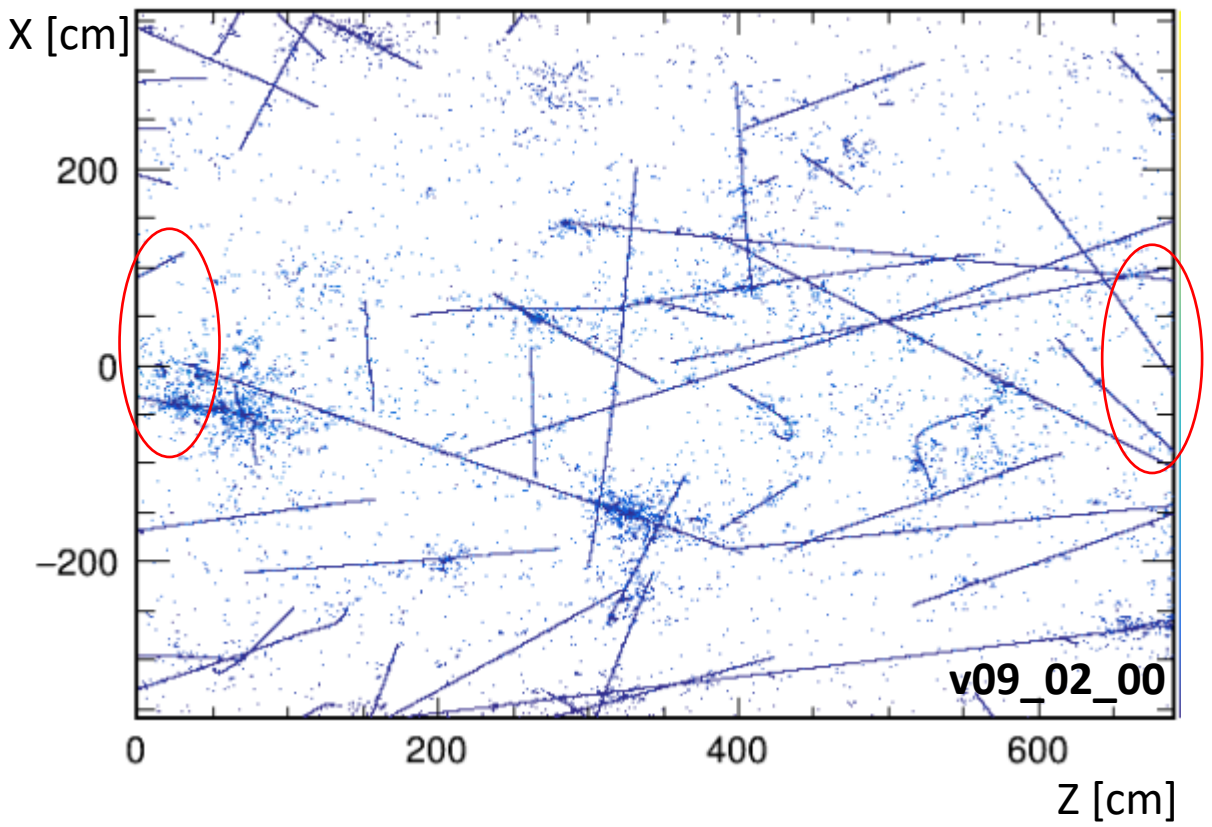
- Wire-Cell is a generic toolkit for LArTPC modeling and signal processing (https://lar.bnl.gov/wire-cell/)

More reading: DUNE collaboration meeting, May 2019

MC backtracker: X vs. Z

@Wenqiang Gu





- True position retrieved from MC backtracker is now without SCE distortion
- A nontrivial change to the current simulation flow, need to keep the “motherhood” of origin energy depos along the simulation chain (SCE, electron drift)

More reading: LArSoft Coordination Meeting

Performance in ProtoDUNE-SP

Longitudinal diffusion analysis

@ Ajib

- Deconvoluted charge width as function of drift distance on a single wire

Diffusion coefficient

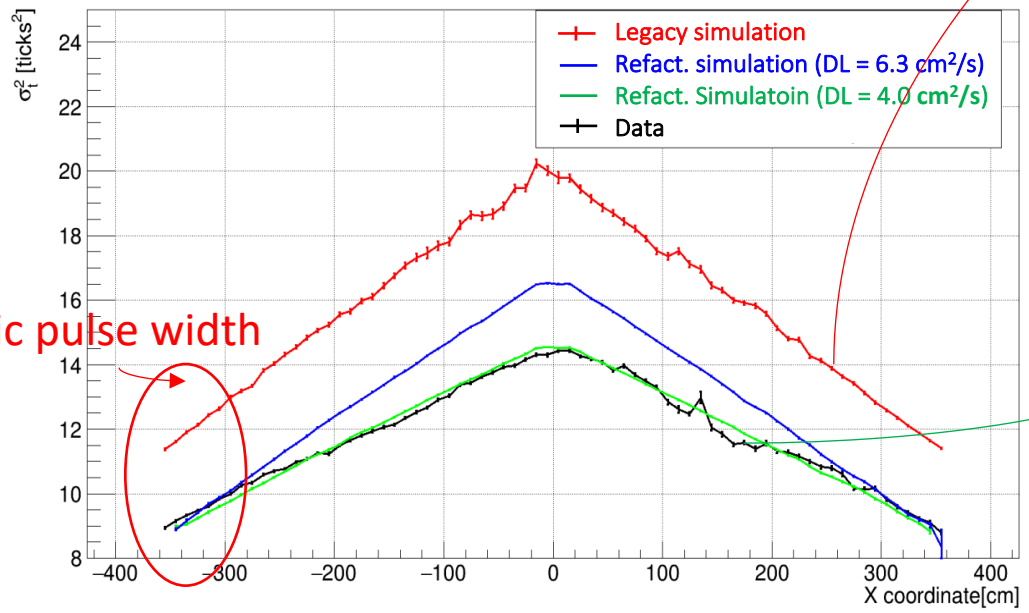
Total time width of pulse

Drift distance

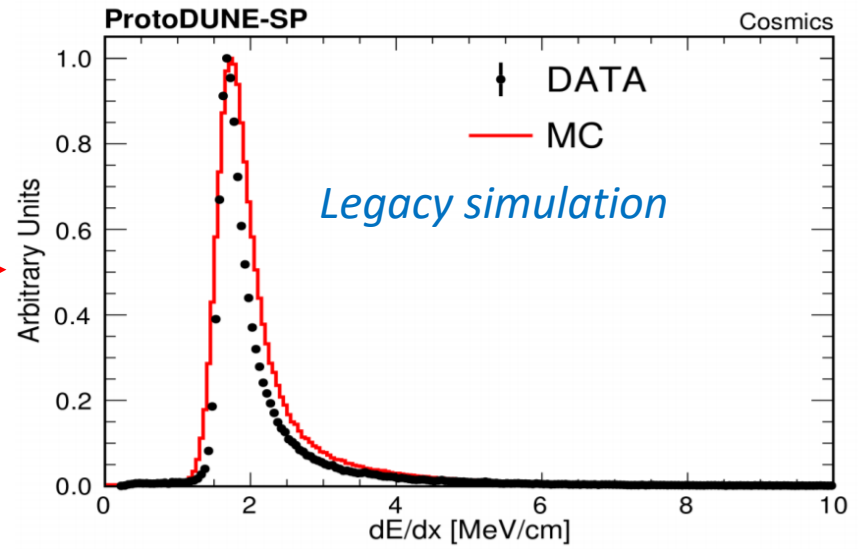
$$\sigma_t^2 = \left(\frac{2D_L}{v_d^3}\right) x + \sigma_0^2$$

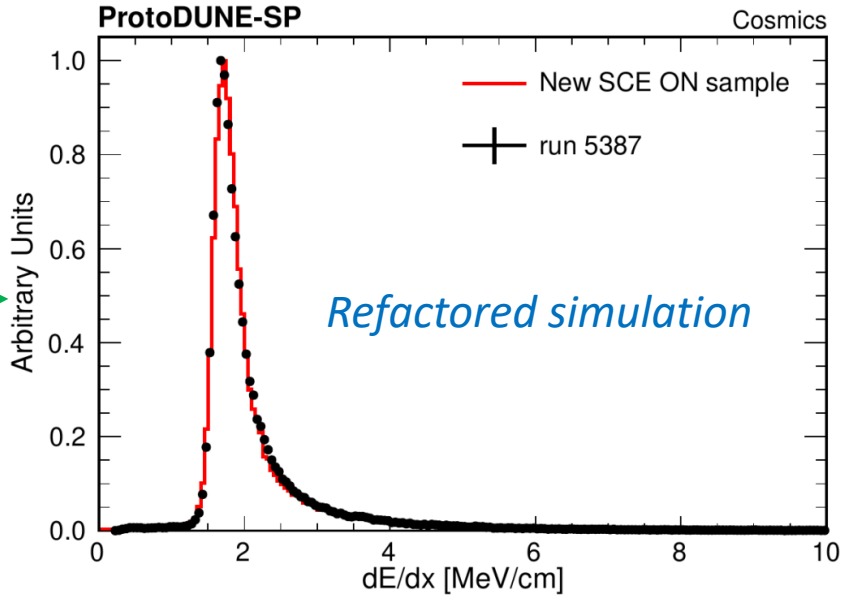
Drift velocity

Inherent pulse width

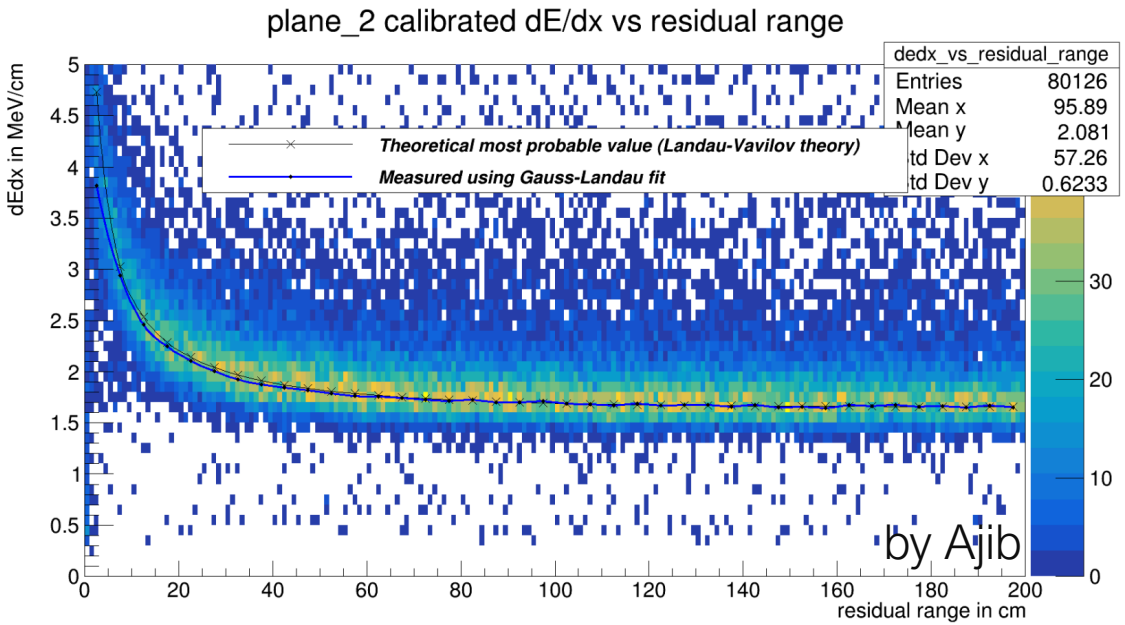


Intrinsic pulse width

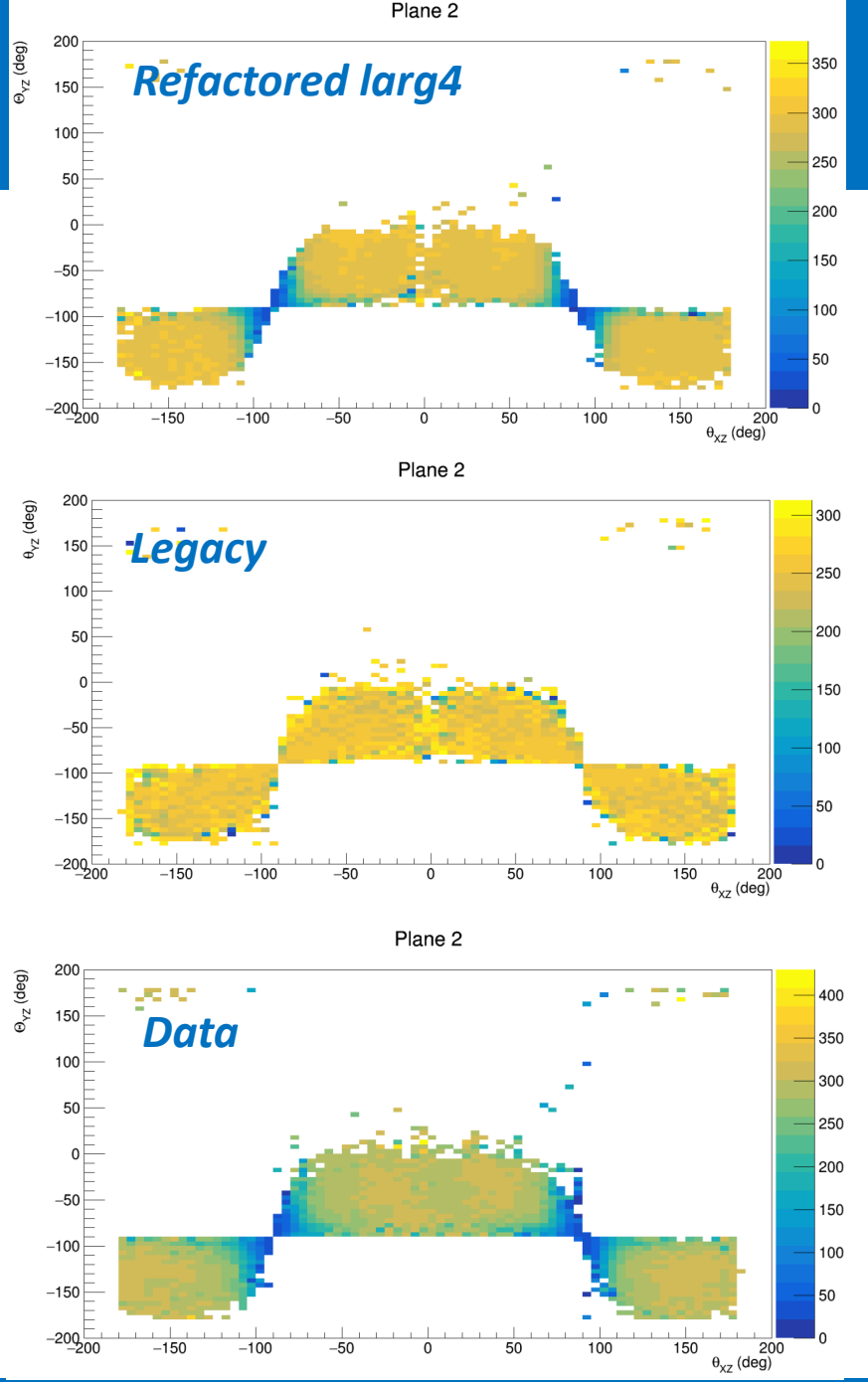




Calorimetry performance

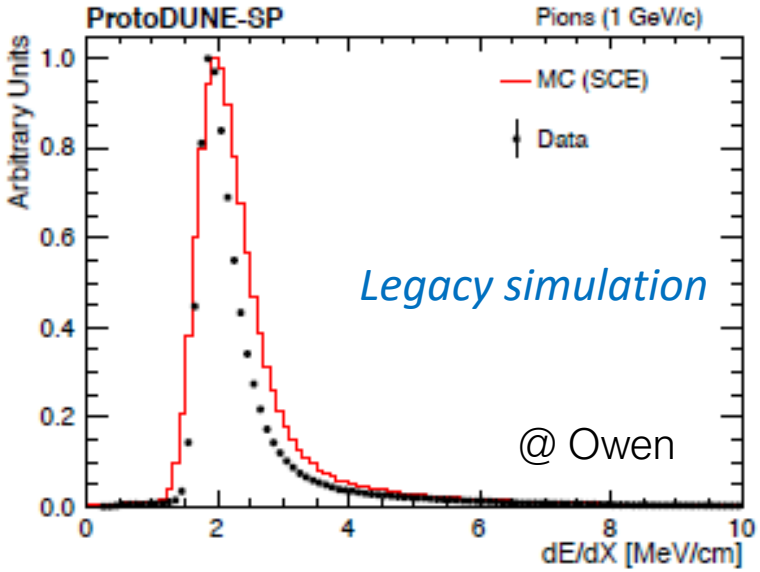


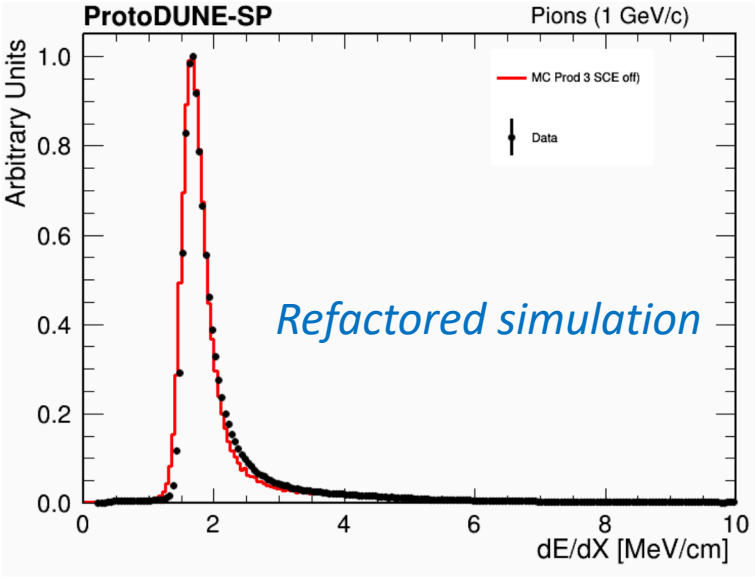
Refactored simulation for cosmics



- dE/dx looks fine for cosmic rays
- The θ_{yz} and θ_{xz} dependence agrees better with the data

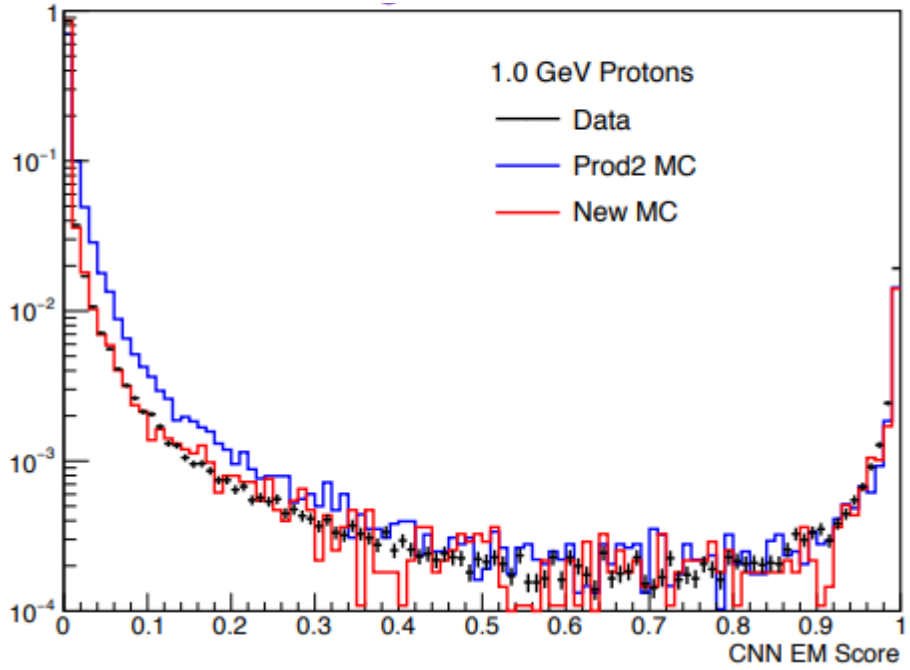
dE/dx: beam pion

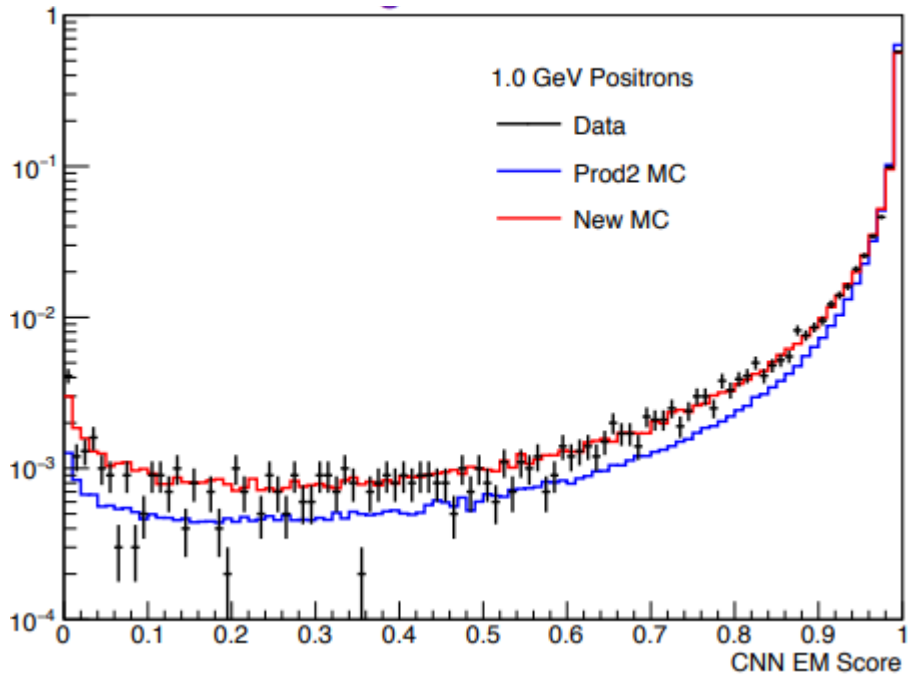




- Similarly, better agreement for pion and other particles

Improved CNN prediction





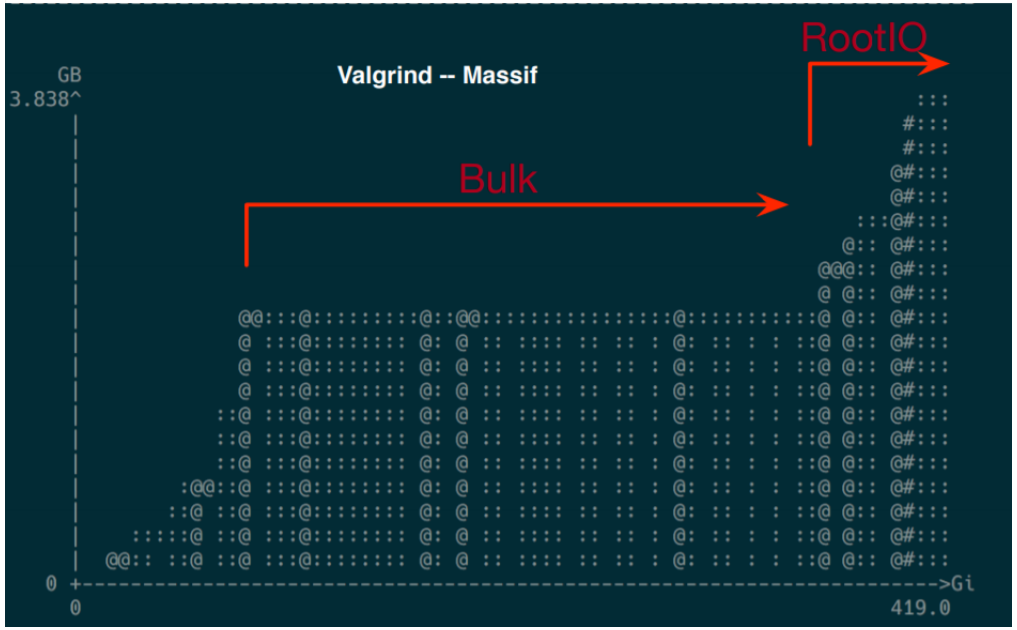
- Good performance of the hit-tagging CNN with the refactored larg4 framework ("New MC")
- "EM score" – provided by CNN, has a good agreement with protoDUNE data

More readings: ProtoDUNE Sim/Reco meeting, Leigh Whitehead

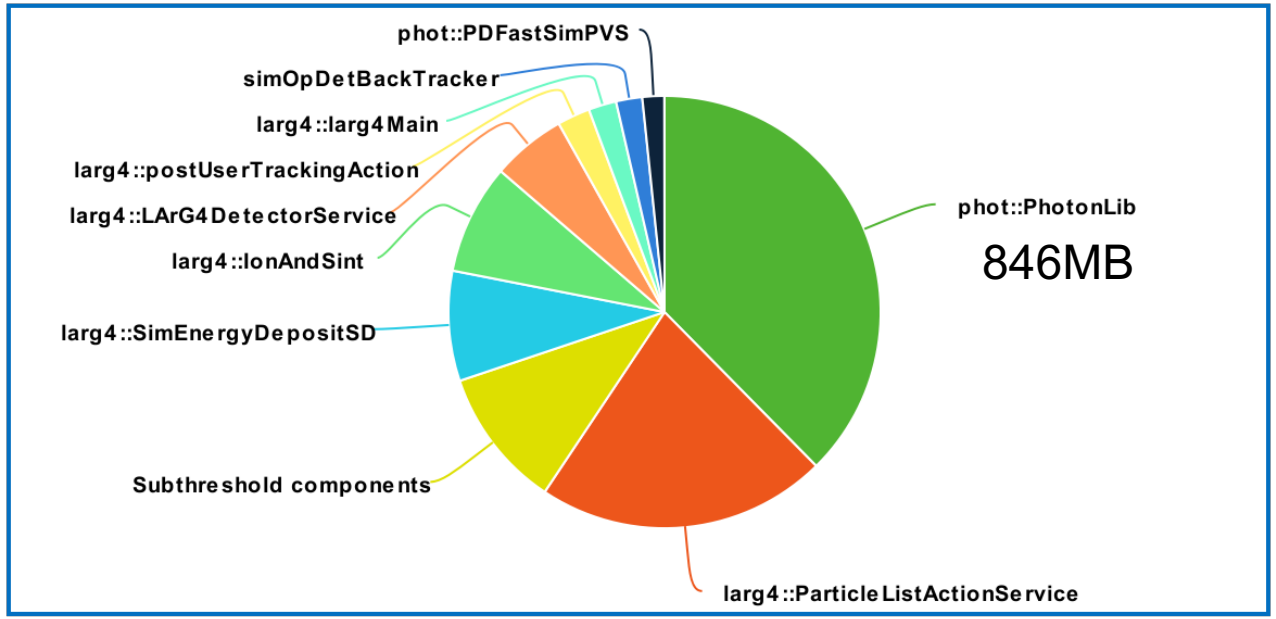
Going forward

Memory usage

David Rivera



Memory usage in larg4 bulk



- Possible ways to reduce memory
  - Revamp sim::MCParticle and sim::SimEnergyDeposit
  - Parameterized/CNN-based photon library

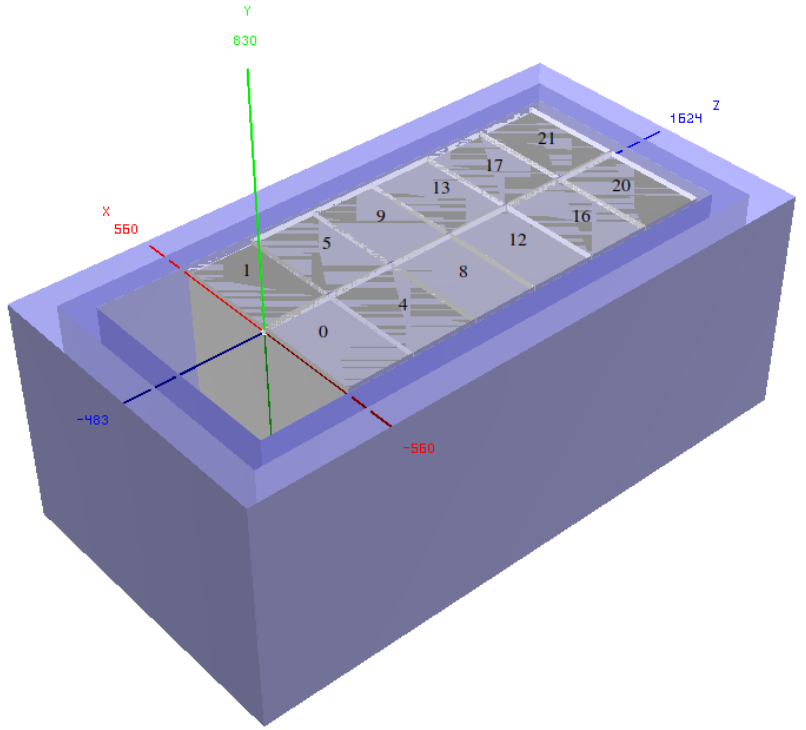
More about memory usage:
[T. Yang] https://indico.fnal.gov/event/24380/contributions/185204/attachments/127942/154552/pdunesimmemory.pdf

Simulation: DUNE FD 1x2x6 module

@Bishu

- The refactored larg4 recently has been tested on the DUNE FD 1x2x6 module
  - Geometry, Gean4, TPC simulation and reconstruction

```
dune
  Geometry
    gdml
      dune10kt_v4_refactored_1x2x6.gdml
      dune10kt_v4_refactored_1x2x6_nowires.gdml
fcl
  dunefd
    detsim
      opdet_multidetsim_refactored_dune10kt_1x2x6.fcl
    g4
      standard_g4_refactored_dune10kt.fcl
      supernova_g4_refactored_dune10kt_1x2x6.fcl
    gen
      supernova
        prodmarley_nue_spectrum_radiological_refactored_dune10kt_1x2x6_vtxbox.fcl
    mergeana
      opdet_multiana_refactored_dune10kt_1x2x6.fcl
    reco
      opdet_multireco_refactored_dune10kt_1x2x6.fcl
```



More readings:

[Photon Detector WG meeting]

[DUNE Geometries]

Summary

- Switching from legacy (larsim/LArG4) to the refactored larg4 framework is recommended
  - Easier to maintain
  - More recent, accurate and extensible physics list
  - Separates physics simulation and detector response
- The full simulation chain is well defined with the refactored larg4, ionization simulation and the subdetector simulations
  - The application in DUNE FD is under development
- The refactored framework shows better agreement with protoDUNE-SP measurement
  - TPC calorimetry measurement: dE/dx
  - Intrinsic pulse width in the diffusion analysis
  - CNN Hit-tagging

Thank you!

- For more details on the performance of ProtoDUNE-SP see our recently published arxiv paper !
  - First results on ProtoDUNE-SP liquid argon time projection chamber performance from a beam test at the CERN Neutrino Platform

Backup Slides

Supplementary

- Slack channel #pdsp-sim-tasks
- Simulation task force wiki: https://wiki.dunescience.org/wiki/ProtoDUNE-SP_Simulation_Task_Force

Extensible physics list

- Extensible physics list factory

Reference physics lists + EM options + user cuts (neutron tracking cuts, stepLimiter, etc.)

Configurable (fhicl) options in physics list service

```
19 artg4tk::PhysicsListService::PhysicsListService(fhicl::ParameterSet const & p, art::ActivityRegistry &) :
20   PhysicsListName_( p.get<std::string>("PhysicsListName","FTFP_BERT")),
21   DumpList_( p.get<bool>("DumpList",false)),
22   enableNeutronLimit_(p.get<bool>("enableNeutronLimit",true)),
23   NeutronTimeLimit_(p.get<double>("NeutronTimeLimit",10.*microsecond)),
24   NeutronKinELimit_(p.get<double>("NeutronKinELimit",0.0)),
25   enableStepLimit_(p.get<bool>("enableStepLimit",true)),
26   enableOptical_(p.get<bool>("enableOptical",true)),
27   enableCerenkov_( p.get<bool>("enableCerenkov",false)),
28   CerenkovStackPhotons_( p.get<bool>("CerenkovStackPhotons",false)),
29   CerenkovMaxNumPhotons_(p.get<int>(" CerenkovMaxNumPhotons",100)),
30   CerenkovMaxBetaChange_(p.get<double>("CerenkovMaxBetaChange",10.0)),
31   CerenkovTrackSecondariesFirst_( p.get<bool>("CerenkovTrackSecondariesFirst",false)),
32   enableScintillation_( p.get<bool>("enableScintillation",true)),
33   ScintillationStackPhotons_( p.get<bool>("ScintillationStackPhotons",false)),
34   ScintillationByParticleType_( p.get<bool>("ScintillationByParticleType",true)),
35   ScintillationTrackInfo_( p.get<bool>("ScintillationTrackInfo",false)),
36   ScintillationTrackSecondariesFirst_( p.get<bool>("ScintillationTrackSecondariesFirst",false)),
37   enableAbsorption_( p.get<bool>("enableAbsorption",false)),
38   enableRayleigh_( p.get<bool>("enableRayleigh",false)),
39   enableMieHG_( p.get<bool>("enableMieHG",false)),
40   enableBoundary_( p.get<bool>("enableBoundary",false)),
41   enableWLS_( p.get<bool>("enableWLS",false)),
42   BoundaryInvokeSD_( p.get<bool>("BoundaryInvokeSD",false)),
43   verbositylevel_( p.get<int>("Verbosity",0)),
44   WLSProfile_( p.get<std::string>("WLSProfile","delta"))
```

```
Base G4VModularPhysicsLists in G4PhysListRegistry are:
 [  0]  "FTFP_BERT"
 [  1]  "FTFP_BERT_ATL"
 [  2]  "FTFP_BERT_HP"
 [  3]  "FTFP_BERT_TRV"
 [  4]  "FTFP_INCLXX"
 [  5]  "FTFP_INCLXX_HP"
 [  6]  "FTF_BIC"
 [  7]  "G4GenericPhysicsList"
 [  8]  "LBE"
 [  9]  "NuBeam"
 [ 10]  "QBBC"
 [ 11]  "QGSP_BERT"
 [ 12]  "QGSP_BERT_HP"
 [ 13]  "QGSP_BIC"
 [ 14]  "QGSP_BIC_AllHP"
 [ 15]  "QGSP_BIC_HP"
 [ 16]  "QGSP_FTFP_BERT"
 [ 17]  "QGSP_INCLXX"
 [ 18]  "QGSP_INCLXX_HP"
 [ 19]  "QGS_BIC"
 [ 20]  "Shielding"
 [ 21]  "ShieldingLEND"
 [ 22]  "ShieldingM"
Replacement mappings in G4PhysListRegistry are:
          EMV =>    G4EmStandardPhysics_option1
          EMX =>    G4EmStandardPhysics_option2
          EMY =>    G4EmStandardPhysics_option3
          EMZ =>    G4EmStandardPhysics_option4
           GS =>          G4EmStandardPhysicsGS
          LIV =>           G4EmLivermorePhysics
    NEUTRONLIMIT =>          G4NeutronTrackingCut
       OPTICAL =>             G4OpticalPhysics
          PEN =>           G4EmPenelopePhysics
     STEPLIMIT =>          G4StepLimiterPhysics
          _GS =>          G4EmStandardPhysicsGS
Use these mapping to extend physics list; append with _EXT or +EXT
   to use ReplacePhysics( ) ("_") or RegisterPhysics( ) ("+").
 Name of Physics list: QGSP_BERT_HP+OPTICAL+STEPLIMIT
G4PhysListRegistry::GetModularPhysicsList <QGSP_BERT_HP+OPTICAL+STEPLIMIT>,
as "QGSP_BERT_HP" with extensions "+OPTICAL+STEPLIMIT"
<<< Geant4 Physics List simulation engine: QGSP_BERT_HP 3.0
```

Customized Bertini cascade model

@ David Rivera

- Pion interaction in the nucleus described by the Bertini cascade model
- Customized Bertini model can provide the particle types from intra-nuclear cascade
  - Helpful for pion quasi-elastic scattering measurement
  - Subprocess measurement

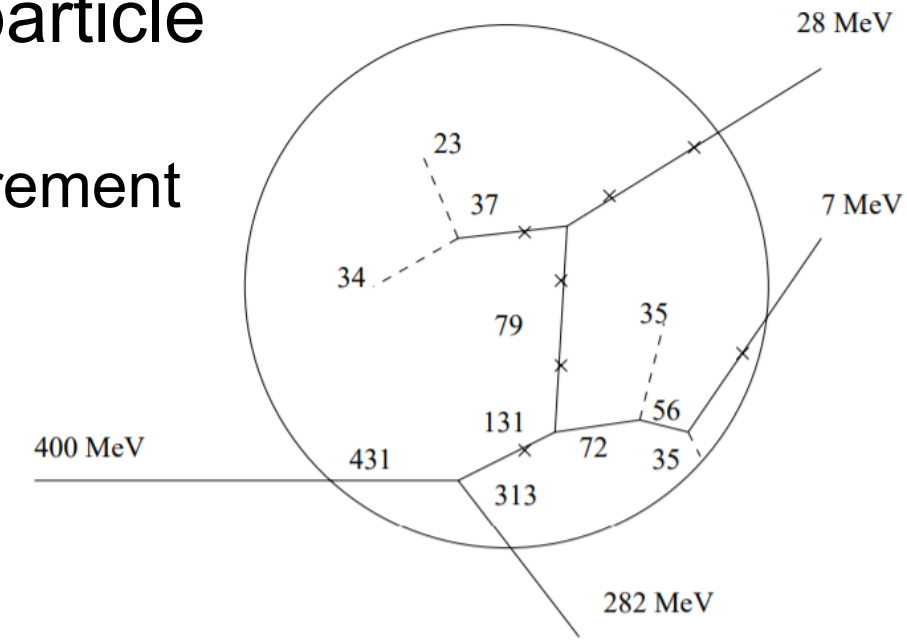


Figure 1: Schematic presentation of the intra-nuclear cascade. A hadron with 400 MeV energy is forming an INC history. Crosses present the Pauli exclusion principle in action. (The picture is a reproduction from original work of Bertini [4].)

More about Bertini model in Geant4:
https://www.slac.stanford.edu/econf/C0303241/proc/papers/MOMT008.PDF

Overview of Wire-Cell simulation

$$\textbf{ADC Wavform} = (\boldsymbol{Depo} \circledast \boldsymbol{Drifter} \circledast \boldsymbol{Ductor} + \boldsymbol{Noise}) \times \boldsymbol{Digitizer}$$

One charge depo
(x, y, z, t0, # of electrons)

Data-driven input + analytic method

- ✓Field response (pre-calculated 2D Garfield calculation)
- ✓Pre-amplifier electronic response (gain, shaping)
- ✓Additional response (AC coupling)

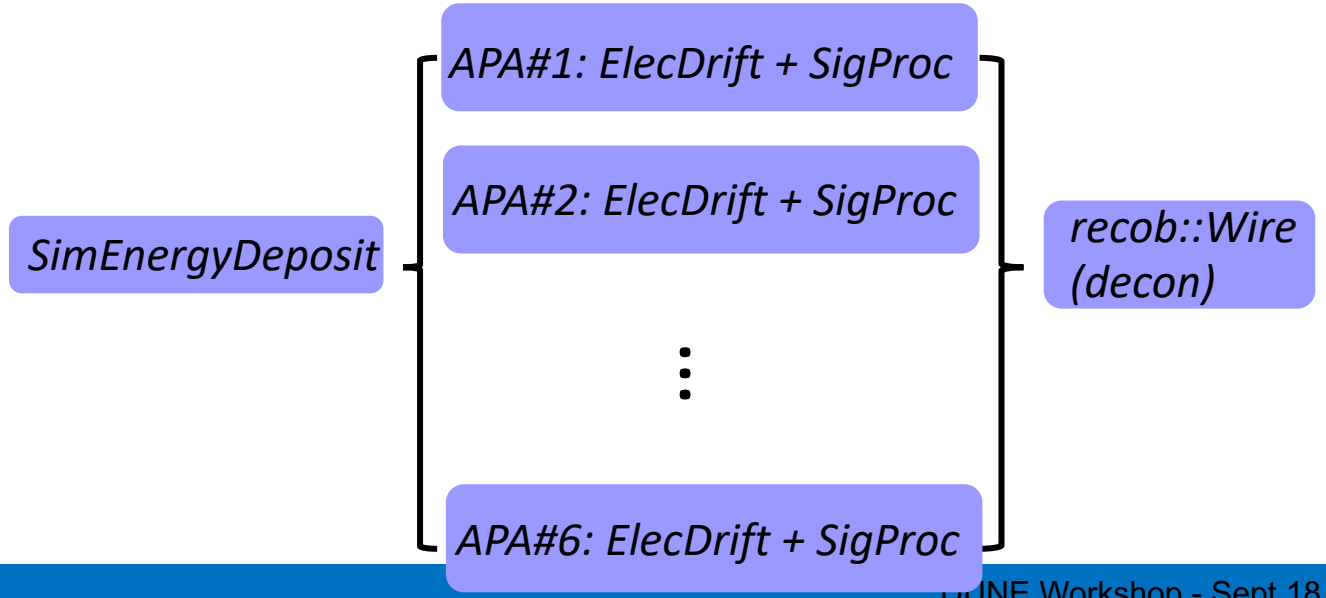
Kernel:

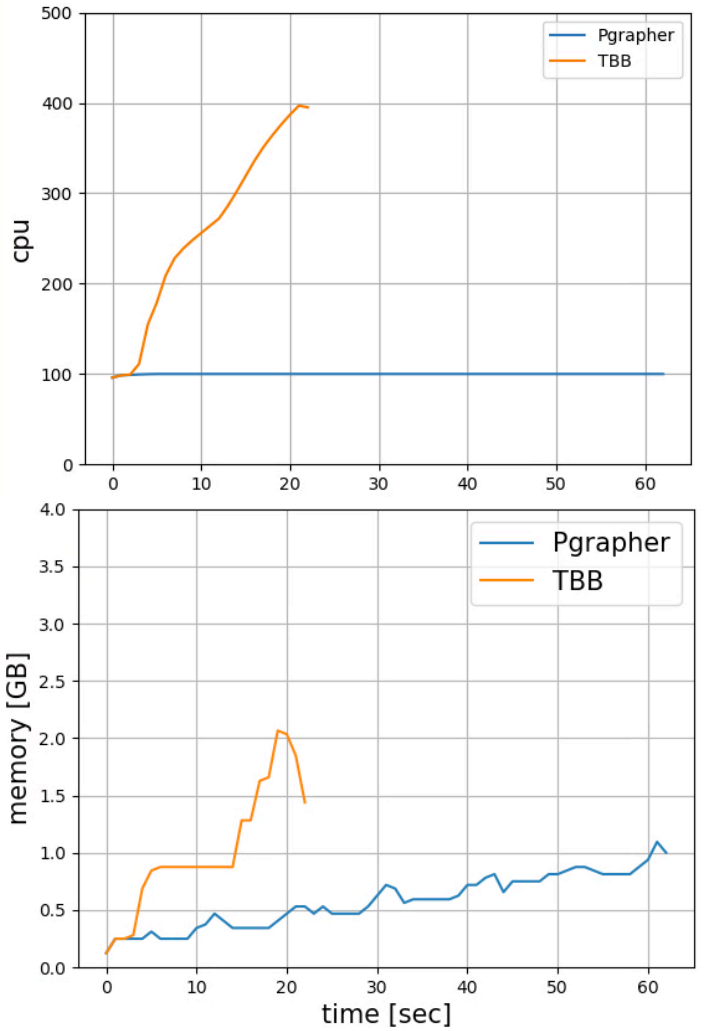
$$[Gaus(t) \cdot Gaus(x)] \otimes field(x,t) \otimes Preamp\ (t) \otimes RC(t)$$

- ✓ Ionized electron absorption (electron lifetime in LAr)
- ✓ Gaussian diffusion (longitudinal / transverse)
- ✓ Fluctuation (for each gird of the discretized 2D Gaussian cloud)

Multithreading in WireCell Simulation

- Multithreading engine based on Intel Thread Building Block (TBB)
- 3 times faster than single thread mode (“pgrapher”)
- Uses maximum 4×CPU and 2×memory
  - memory sharing is working
  - need tests for longer jobs

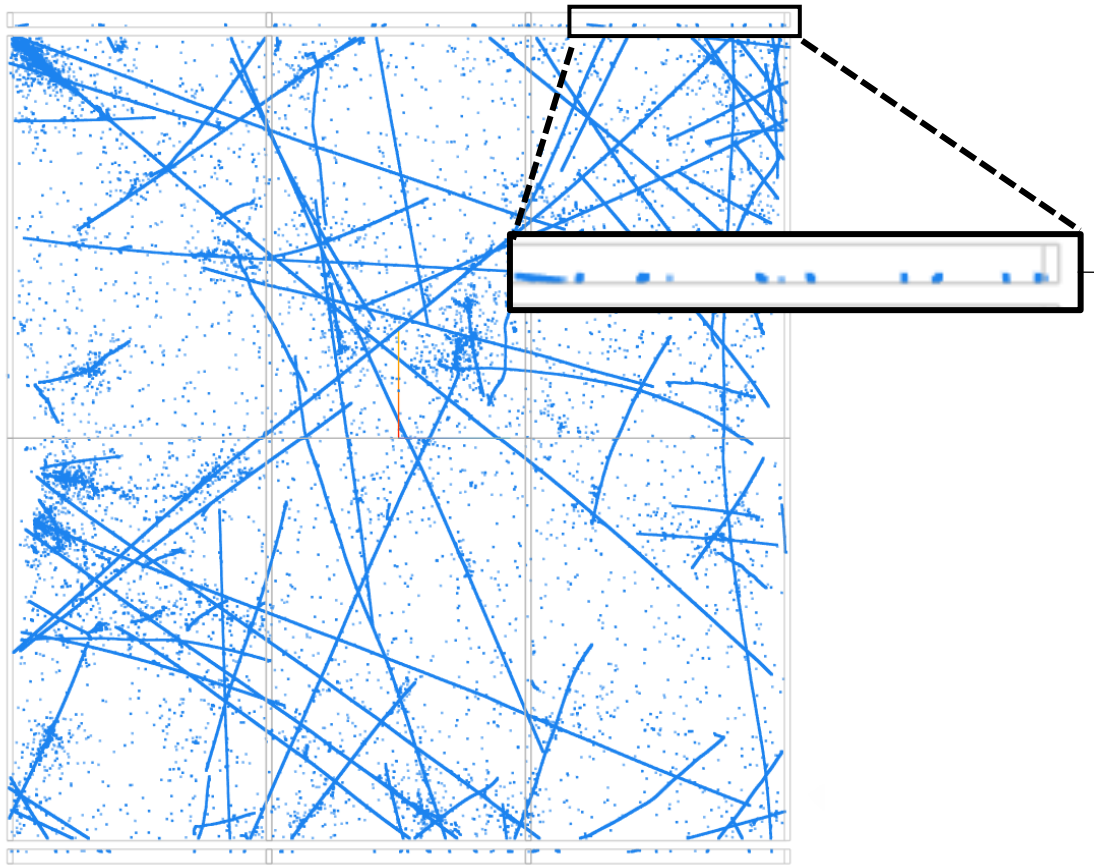


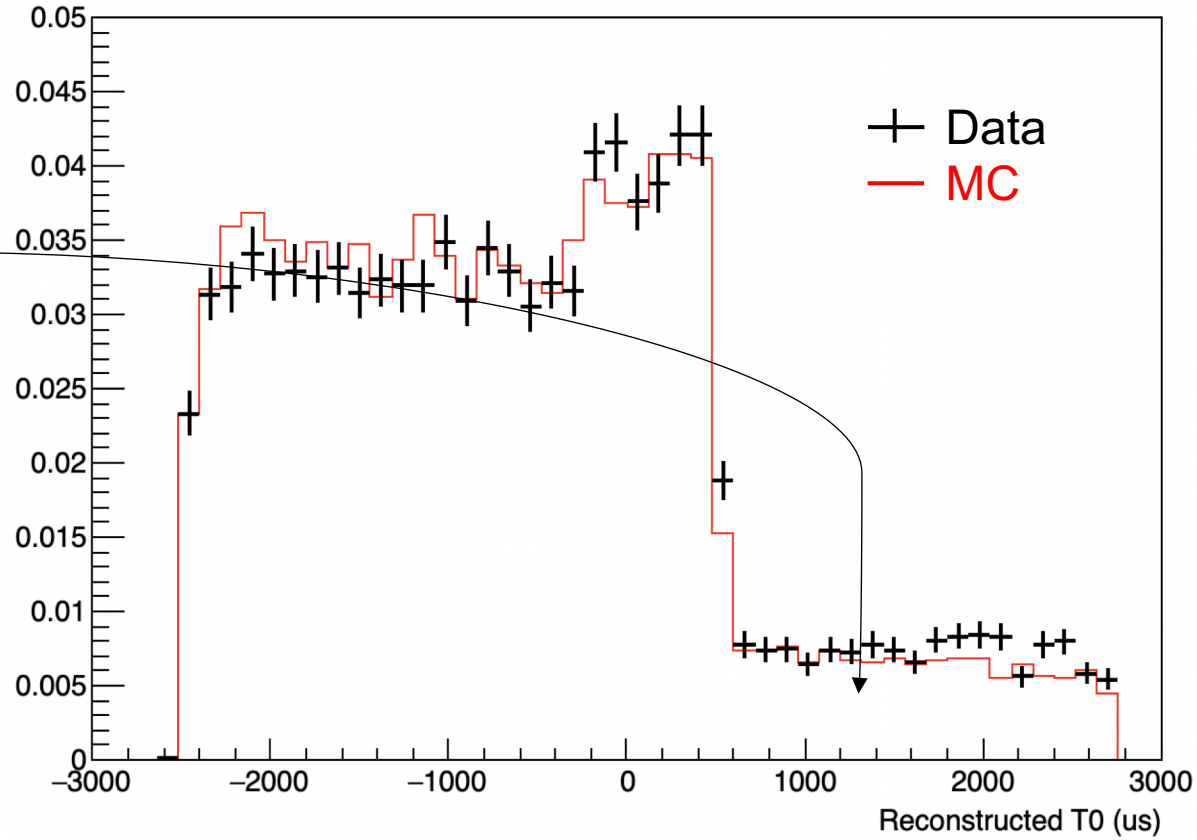


CPU and memory profile for 1-track

Cryostat-side simulation

- Simulation is also done at the cryostat side
- T0 stitched at cathode: good agreement for the cryostat-side

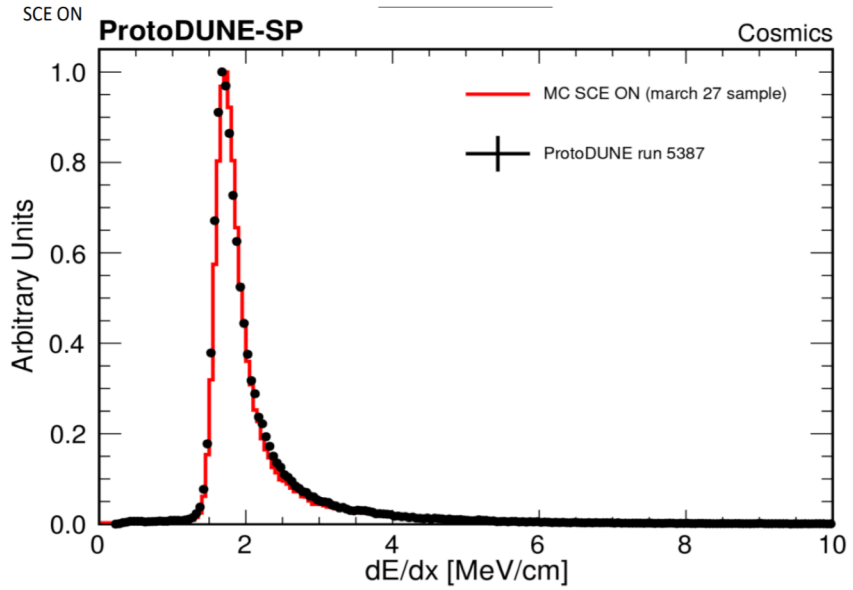


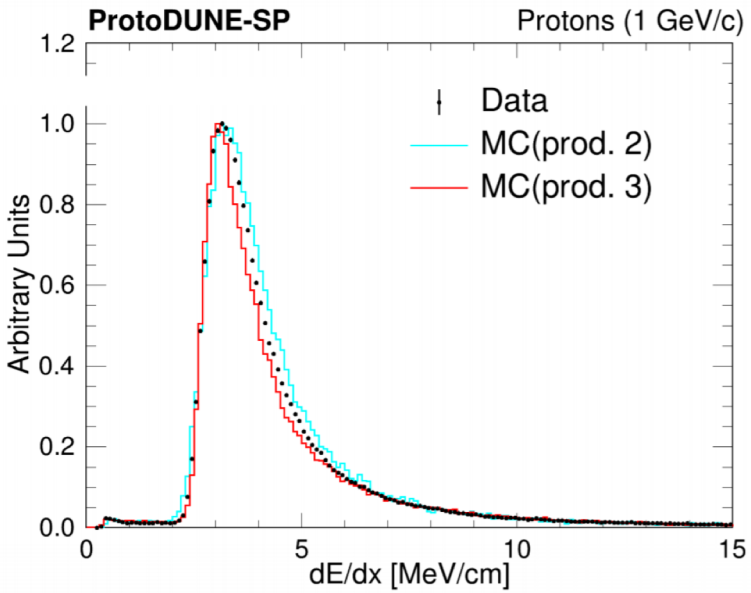


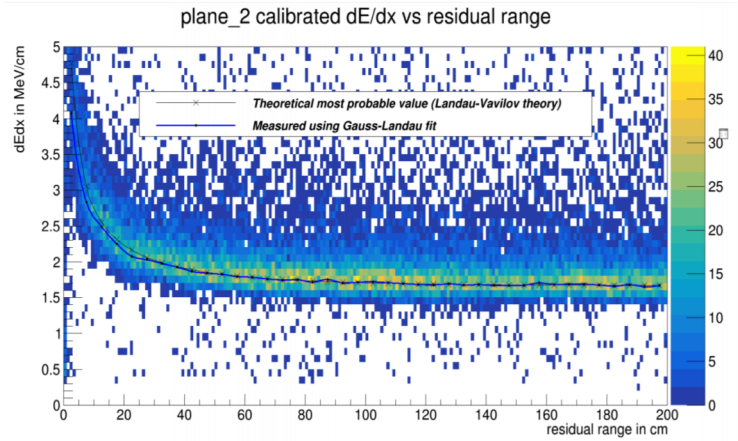
https://www.phy.bnl.gov/wire-cell/bee/set/5633d85e-cefe-442f-8e84-3db7aa227d27/event/0/

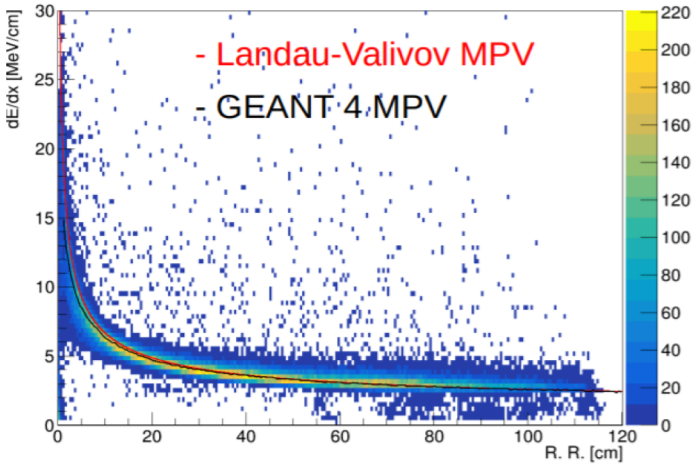
https://indico.fnal.gov/event/23920/contribution/0/material/slides/1.pdf
https://indico.fnal.gov/event/23882/contribution/0/material/slides/0.pdf

Prod 3 sample validation by Ajib & Heng-Ye



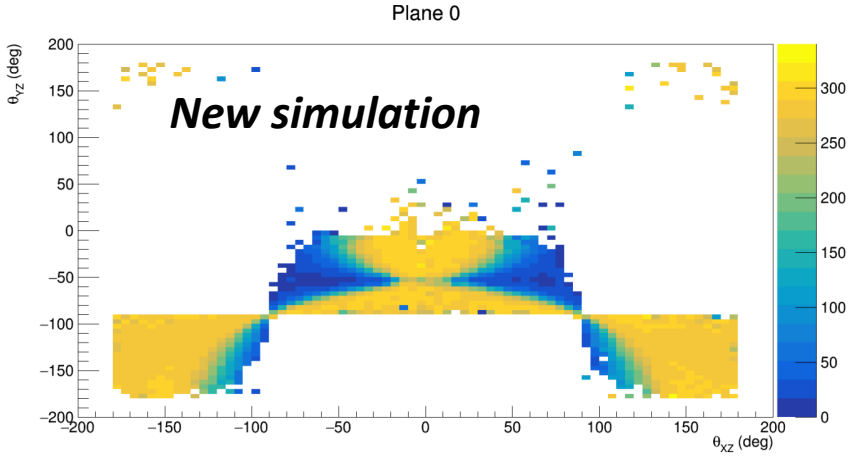


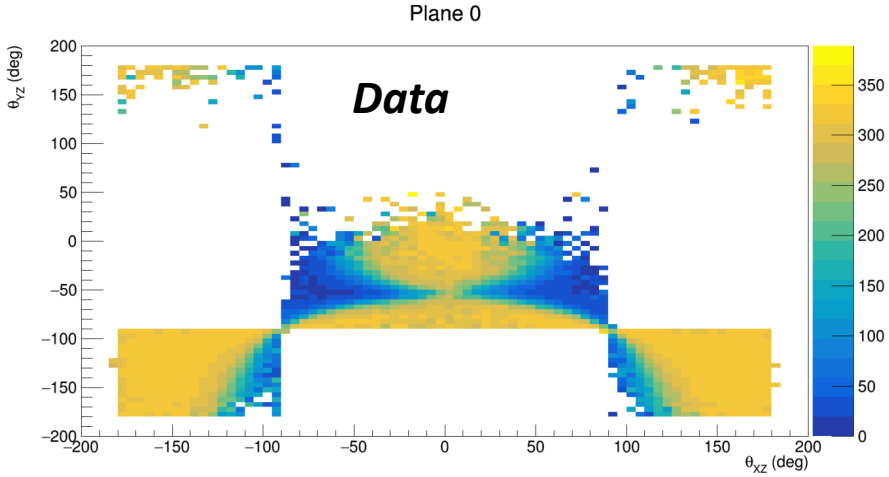


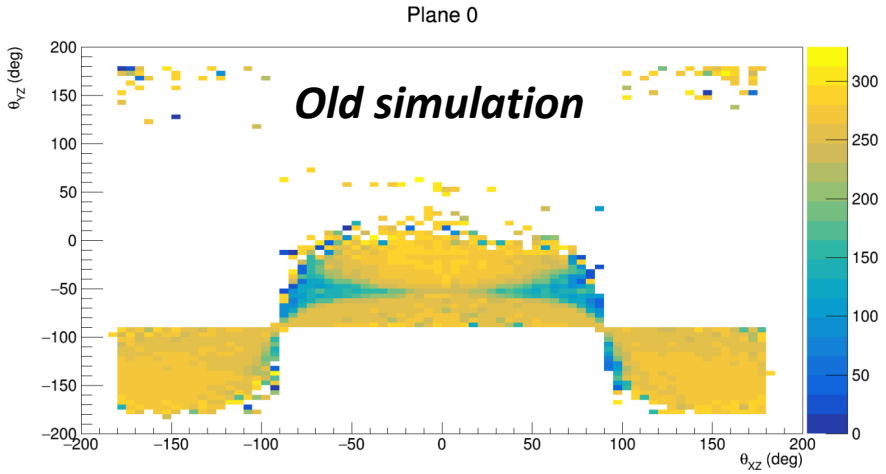


Induction plane (SCE OFF comparison)

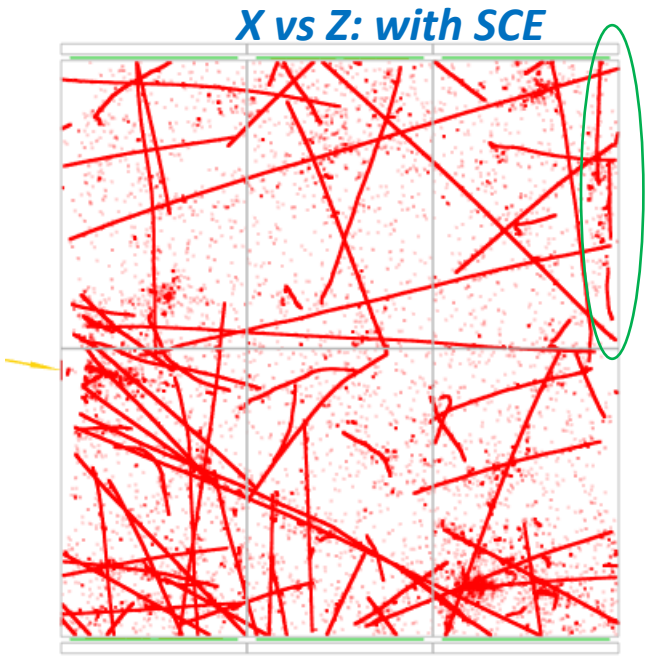
By Ajib



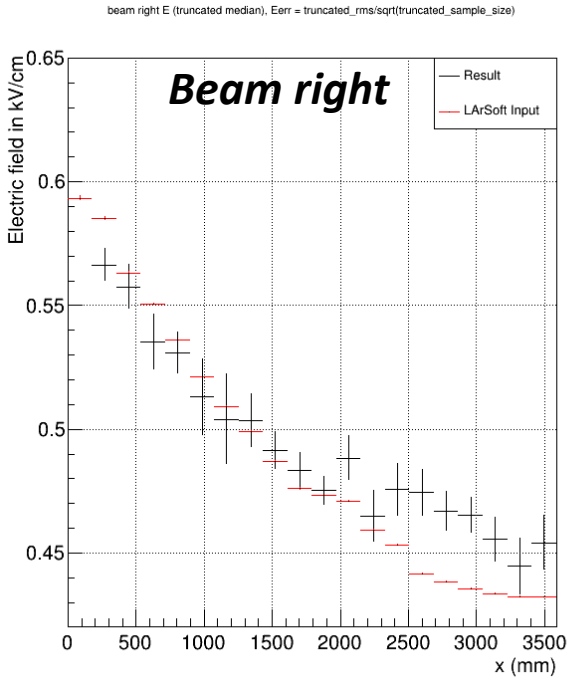


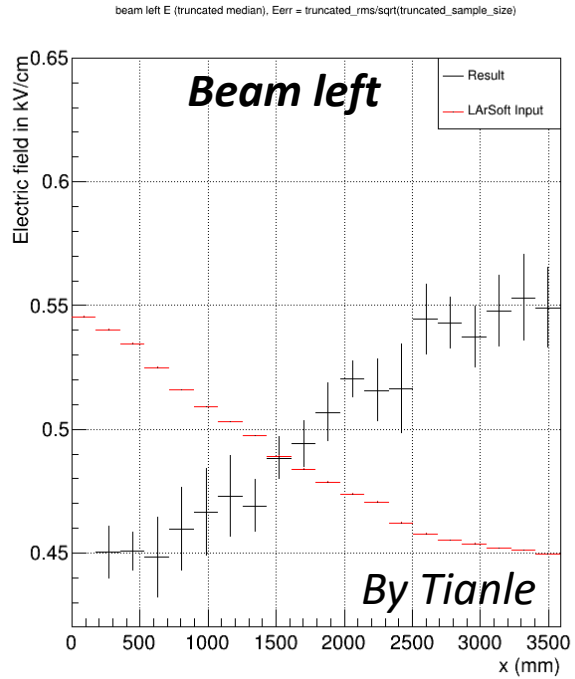


SCE simulation: sign flipping issue in beam left



https://www.phy.bnl.gov/wire-cell/bee/set/98db20da-49a3-478b-8c60-0b5d06be7946/event/0/





- Beam left looks weird
- Sign flipping issue in X axis

```
432  if (fRepresentationType=="Voxelized_TH3"){
433    if (point.X() > 0.) {
434            thePosOffsets = GetOffsetsVoxel(point, SCEhistograms.at(0), SCEhistograms.at(1), SCEhistograms.at(2));
435            thePosOffsets[0] = thePosOffsets[0];
436    } else {
437            thePosOffsets = GetOffsetsVoxel(point, SCEhistograms.at(6), SCEhistograms.at(7), SCEhistograms.at(8));
438            thePosOffsets[0] = -1.0*thePosOffsets[0];
439    }
```

Corrected result