protoDUNE MC: Intro to LArSoft Simulations and MC Samples

Elizabeth Worcester (BNL)

protoDUNE Science Workshop

CERN, June 2016

Overview

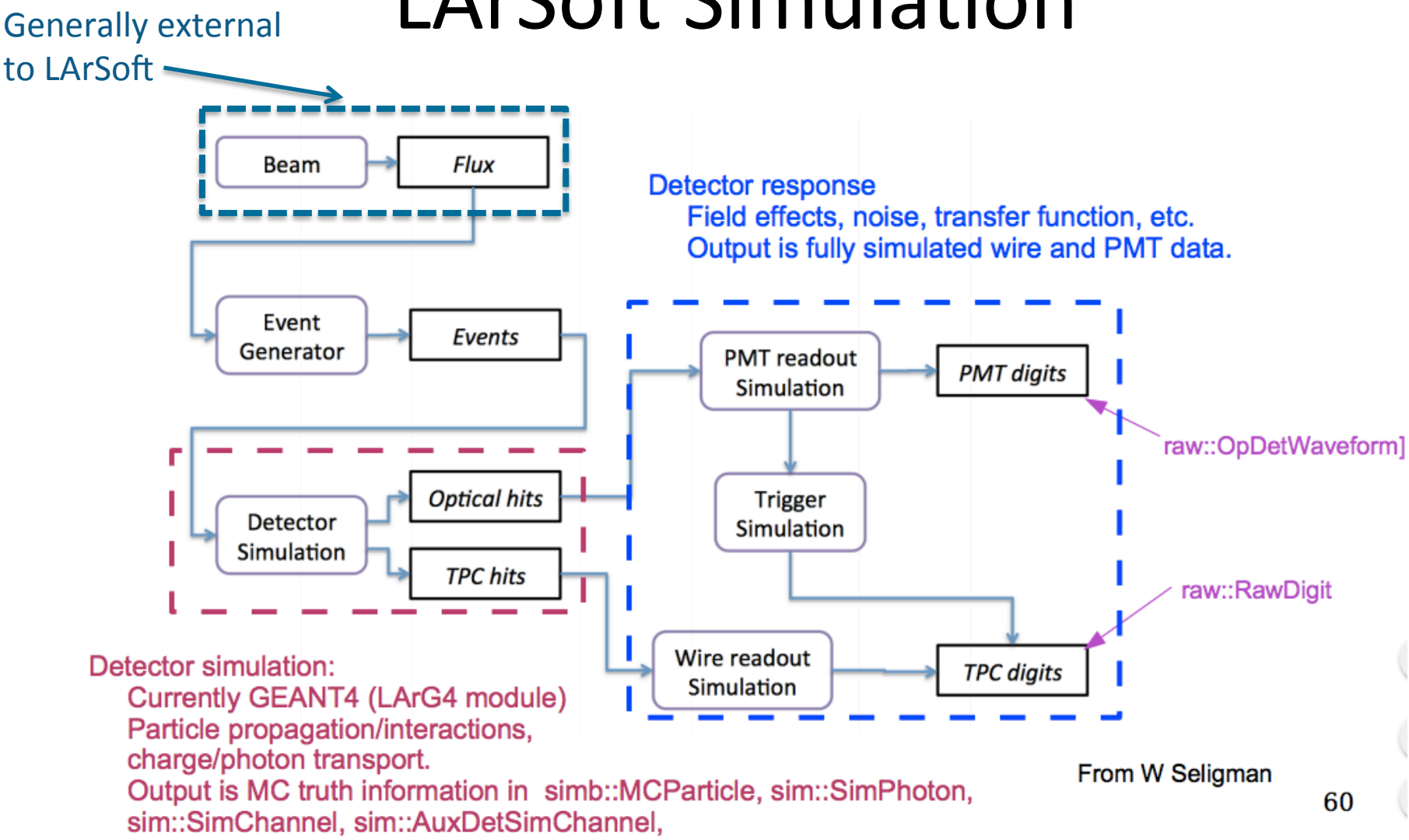
- The purpose of this talk is to give a general overview of how the LArSoft simulation is used for production of SP protoDUNE MC
  - My perspective is that of a user and developer of small modifications rather than that of a core developer
  - A lot of LArSoft documentation exists:
    - art/LArSoft course: indico.fnal.gov/conferenceDisplay.py?confId=9928
    - Redmine: cdcvs.fnal.gov/redmine/projects (search LArSoft, LArSim, etc)
  - A number of slides borrowed from Erica Snider and Diego Garcia-Gamez – thanks to them for their nice overview talks!
- Basics of LArSoft simulation
- protoDUNE geometry
- protoDUNE simulation of space charge effect
- Corsika simulation of cosmics in LArSoft
- Cosmic simulation in protoDUNE
- Existing and planned MC samples

LArSoft is…

- A project / collaboration
  - Provide an integrated, art-based, experiment-agnostic set of software tools for LAr neutrino experiments to perform simulation, reconstruction and analysis
    - The core LArSoft ("project") team maintains infrastructure, architecture, interfaces, coordination, code management and distribution
    - Experiment partners provide technical requirements, development effort and coordination, required experiment-specific plug-ins and configuration
- A body of code
  - Core LArSoft products
    - Experiment-agnostic data structures, algorithms, interfaces, etc
    - Lives in a set of repositories managed by the core LArSoft team
  - Experiment-specific components
    - Detector-specific geometry descriptions, electronics response functions, calibration functions, etc.
    - Live in repositories managed by the experiments

E. Snider

LArSoft Simulation



protoDUNE LArSoft Details

- protoDUNE code is part of dunetpc:
  - cdcvs.fnal.gov/redmine/projects/dunetpc
  - Wiki on the Redmine page has valuable technical details on how to setup/develop dunetpc code (somewhat Fermilab-centric)
- Most code is common with DUNE FD
- When protoDUNE-specific details are required, this will be set via services fcl files

```
protodune_services: {
 ExptGeoHelperInterface:       @local::dune_geometry_helper
 Geometry:                     @local::protodune_geo
 DetectorClocksService:        @local::dunefd_detectorclocks
 DetectorPropertiesService:    @local::dunefd_detproperties
 LArPropertiesService:         @local::dunefd_properties
 LArFFT:                       @local::dunefd_larfft
 DatabaseUtil:                 @local::dunefd_database
 BackTracker:                  @local::dunefd_backtracker
 SpaceCharge:                  @local::protodune_spacecharge
 LArSeedService:               @local::dune_seedservice
 SignalShapingServiceDUNE:     @local::dunefd_signalshapingservice
 ChannelStatusService:         @local::dunefd_channel_status
}

protodune_simulation_services: {
 LArFFT:                       @local::dunefd_larfft
 LArG4Parameters:              @local::protodune_largeantparameters
 ExptGeoHelperInterface:       @local::dune_geometry_helper
 Geometry:                     @local::protodune_geo
 DetectorClocksService:        @local::dunefd_detectorclocks
 DetectorPropertiesService:    @local::dunefd_detproperties
 LArPropertiesService:         @local::dunefd_properties
 DatabaseUtil:                 @local::dunefd_database
 LArVoxelCalculator:           @local::dunefd_larvoxelcalculator
 MagneticField:                @local::no_mag
 BackTracker:                  @local::dunefd_backtracker
 SpaceCharge:                  @local::protodune_spacecharge
 LArSeedService:               @local::dune_seedservice
 SignalShapingServiceDUNE:     @local::dunefd_signalshapingservice
 PhotonVisibilityService:      @local::protodune_photonvisibilityservice
 OpDetResponseInterface:       @local::dunefd_opdetresponse
 ChannelStatusService:         @local::dunefd_channel_status
```

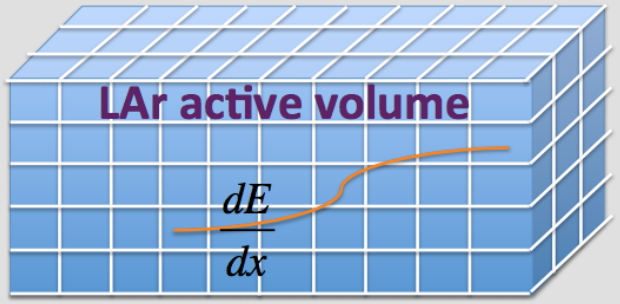
Event Generators

- Produces generator level truth information that is input to Geant4 simulation
- Many options:
  - Particle gun: used for protoDUNE MC “beam” events
  - GENIE: neutrino interaction generator
  - Cosmic ray event generators (CRY, CORSIKA):
    - For protoDUNE MC, we are using a pre-generated CORSIKA library (more on this later)
  - External files

```
#Set generator parameters
#Corresponds to beam window at center of left TPC

physics.producers.generator.PDG: [13]    # Particle ID
physics.producers.generator.PDist: 1   # Momentum distribution (0=uniform, 1=gaussian)
physics.producers.generator.P0: [1.0]    # Central value of momentum (GeV)
physics.producers.generator.SigmaP: [0.05]   # Width of momentum distribution (5%)

physics.producers.generator.PosDist: 0   # Position distribution (0=uniform, 1=gaussian)
physics.producers.generator.X0: [118.106]        # Starting position (cm)
physics.producers.generator.Y0: [395.649]
physics.producers.generator.Z0: [-196.113]
physics.producers.generator.SigmaX: [0.0]
physics.producers.generator.SigmaY: [0.0]
physics.producers.generator.SigmaZ: [0.0]

physics.producers.generator.AngleDist: 0         # Angle distribution (0=uniform, 1=gaussian)
physics.producers.generator.Theta0XZ: [-8.189] # Starting angles (degrees)
physics.producers.generator.Theta0YZ: [-11.229]
physics.producers.generator.SigmaThetaXZ: [0.]
physics.producers.generator.SigmaThetaYZ: [0.]
```

Geant4 Simulation

- ✓ Outsourced to Geant4: provides the LArSoft specific code to collect particles and energy depositions produced by Geant4 and to turn the latter into ionization electrons and scintillation photons.
- ✓ Particles are stepped one after another (oblivious to each-other's existence)
- ✓ The step length is calculated based on the Physics list applied (currently we're using: "QGSP_BERT")
- ✓ At each step, if there is an energy deposition (dE/dx ≠ 0), the charge gets “drifted” and the photons “propagated”



D. Garcia-Gamez

Number of Ionization Electrons

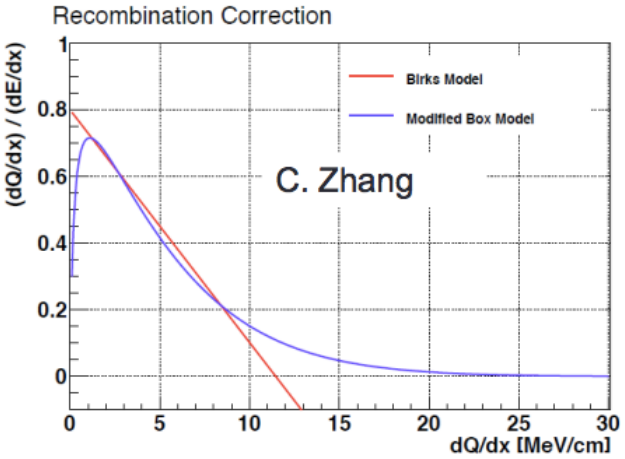
- Two options:
  - ISCalculationSeparate: Calculates ionization electrons and scintillation photons independently (DUNE/protoDUNE default)
  - NEST: "Noble Element Simulation Technique" calculates ionization electrons and scintillation photons together to capture correlations
- Number of electrons depends on energy deposit, calibration factor converting energy to electrons, and recombination factor
- Recombination factor calculated with Birks' model or modified Box model (DUNE/ protoDUNE default is modified Box)
  - ICARUS paper: NIM A523:275-286 (2004)
  - ArgoNeut paper: JINST 8 (2013) P08005
  - Depends on electric field: recombination factor will vary with position due to SCE (LArSoft implementation in progress: M. Mooney, ETW)

Birks model:

$$R = \frac{A}{1 + \left(k/\mathcal{E}\right)\frac{dE}{dx}}$$

Box model:

$$R_C = \frac{Q_{remaining}}{Q_{generated}} = \frac{\ln\left(B + \frac{C}{\xi} \times \frac{dE}{dx}\right)}{\frac{C}{\xi} \times \frac{dE}{dx}}$$



Charge "Drift"

- ✓ `LArVoxelReadout::DriftIonizationElectrons`

  dE/dx → [recombination, lifetime correction (impurities)] → nElectrons (charge)

- ✓ This gets split into clusters (currently of 600, defined in Simulation/LArG4Parameters) that are teleported to the wire planes

- ✓ Their positions are modified: smear X position (drift time) by the longitudinal diffusion and the Y, Z positions by the transverse diffusion $\sigma_{L/T} = \sqrt{(2\, D_{L/T}\, X_{drift}/v_X)}$

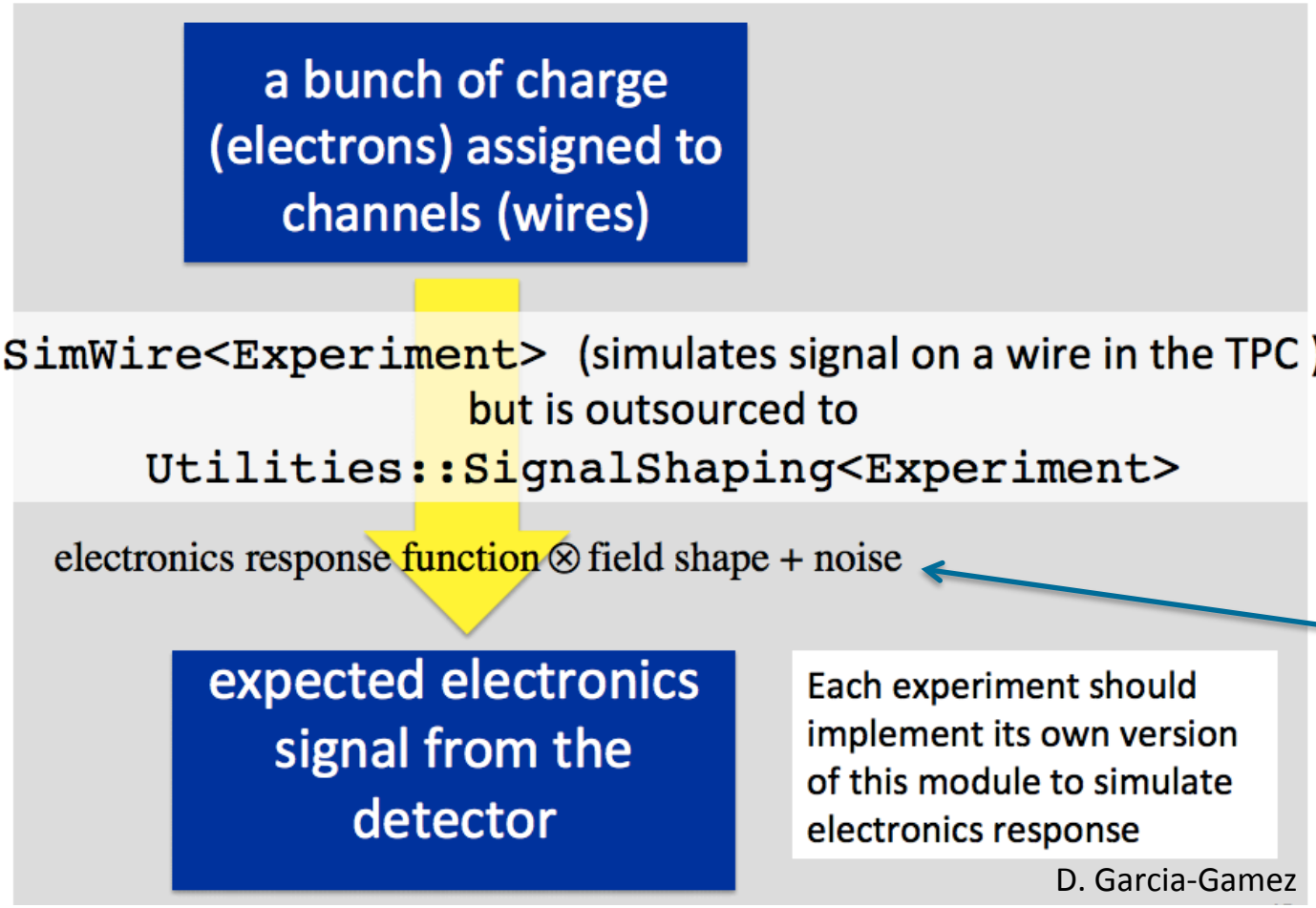
- ✓ This gets saved to `SimChannels (channel, IDEs)`, and more specifically IDE (trackID, numElectrons, energy, xyz) objects

D. Garcia-Gamez

See previous slide for recombination details

Now includes position distortion from space charge effect (M. Mooney)

Detector Response



protoDUNE using nominal DUNE electronics response for now

Optical Simulation

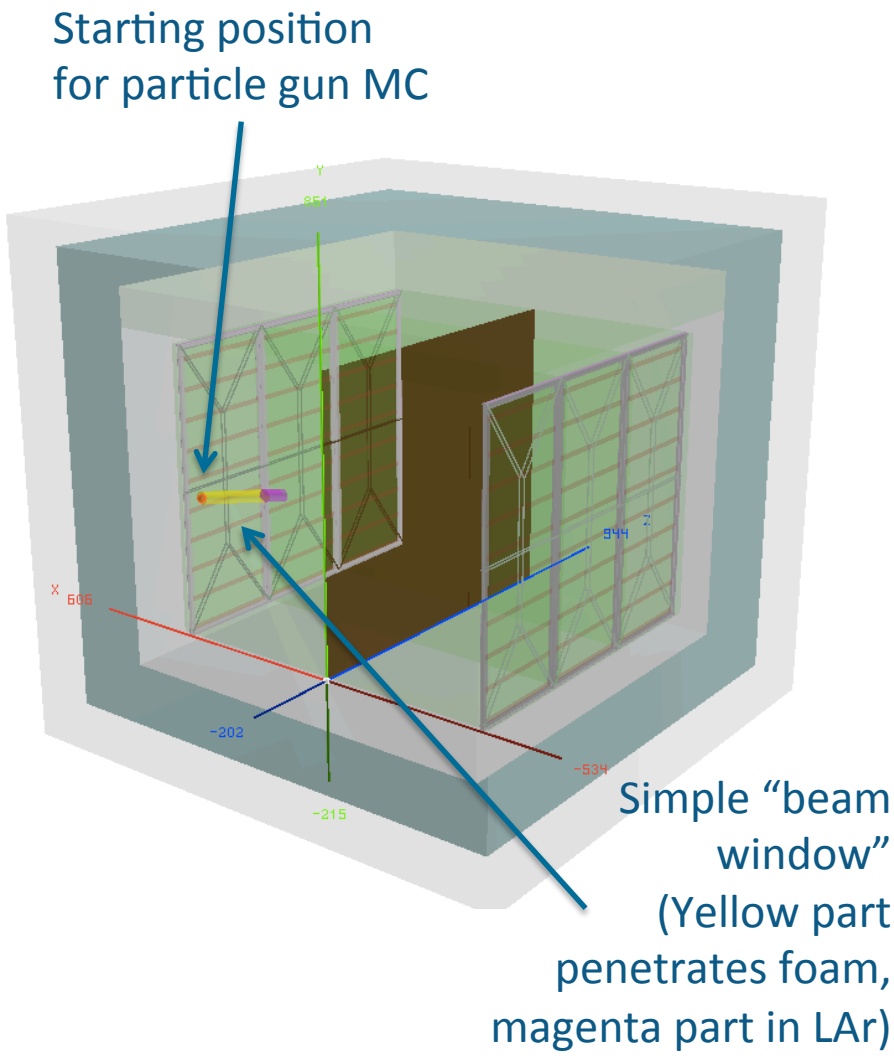
- ✓ Argon scintillates at about **24k photons/MeV** (with a 500 V/cm electric field)
- ✓ It is extremely CPU taxing to simulate each of them
- ✓ A ***"fast optical simulation"*** mode was developed (by B. Jones, MIT) using a fast lookup library (originally for MicroBooNE)
- ✓ **The above uses a "PhotonLibrary" and avoids individual photon propagation:**
  - Library of stored visibility* data to sample an expected detector response at some point in the volume
  - Only for isotropic source (not Cherenkov)
  - Much less CPU than FullOptical (standard geant4)
  - **But library needs to be pre-built (using the full mode optical simulation)!**

D. Garcia-Gamez

- protoDUNE optical library generated by A. Himmel
- Optical simulation included in nominal protoDUNE MC
- protoDUNE simulation of photons has not been studied yet

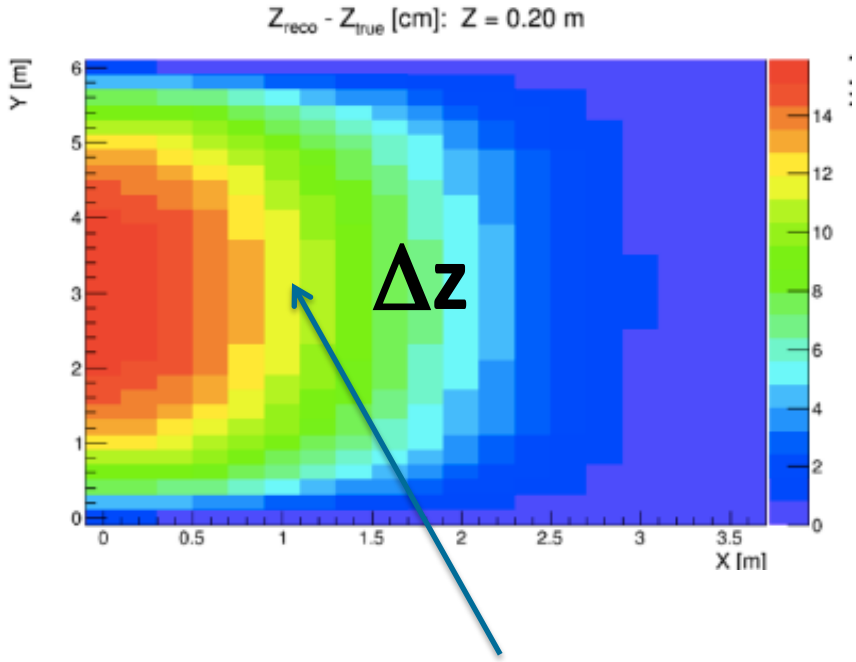
protoDUNE Geometry

- In gdml; includes wires and nowires version
- Developed by Martin Tzanov, starting from script to generate DUNE FD gdml written by Tyler Alion
- Active volume and wire planes are correctly implemented with sufficient detail
- Basic cryostat volume implemented based on Martin's reading of engineering drawings
- Includes basic implementation of field cage (needs work)
- Simple "beam window" implemented as nitrogen-filled steel pipe (needs work)
  - 25 cm outer diameter, 0.5 mm thick steel wall
  - Does not represent details of beam window design
  - Location somewhat arbitrary
  - Still some issues with particles interacting more than expected
- Work ongoing to correct material in outer volumes (foam, steel support) that affects cosmic simulations
- Thanks to Martin for his work so far…we should generate a wish list of desired improvements at this workshop



protoDUNE SCE Simulation

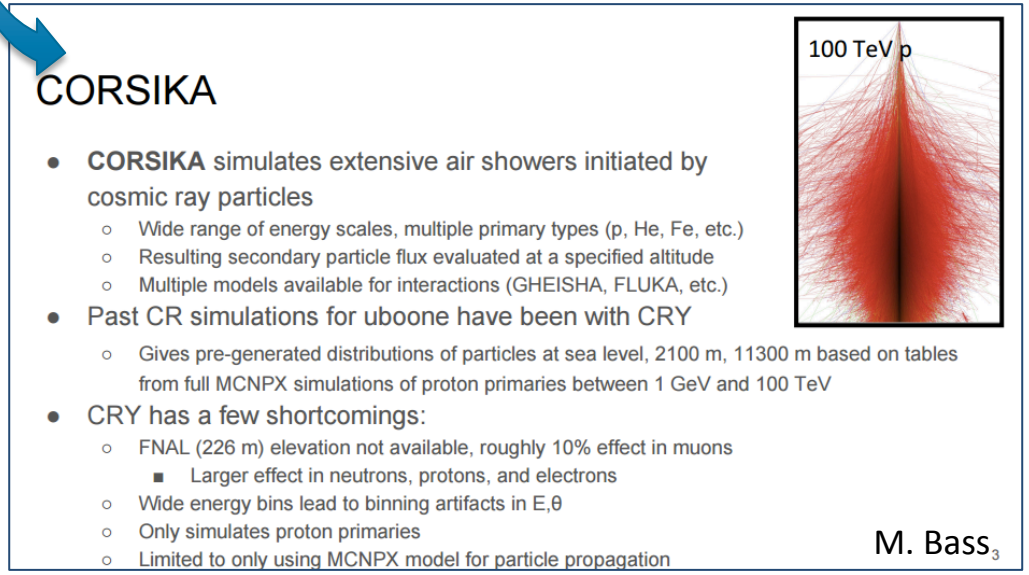
- Not going to repeat details here -- see talk by M. Mooney at this workshop
- Position distortions based on calculated variation in E-field implemented in LArSoft (LArG4/ LArVoxelReadout)
- Adjustment to E-field used to calculate recombination factor in process of being implemented in LArSoft (LArG4/ IonizationAndScintillation)



Typical location of particle gun particles in current geometry

CORSIKA in LArSoft

- CORSIKA cosmic ray simulation implemented in LArSoft by Matt Bass
  - Originally for uBooNE
  - See talk at LArSoft coordination meeting: indico.fnal.gov/conferenceDisplay.py?confId=10893
- CorsikaGen module in LArSoft samples from pre-generated library of CORSIKA samples
- Data files containing pre-generated samples produced on per-experiment basis – thanks to Matt for generating this library for protoDUNE



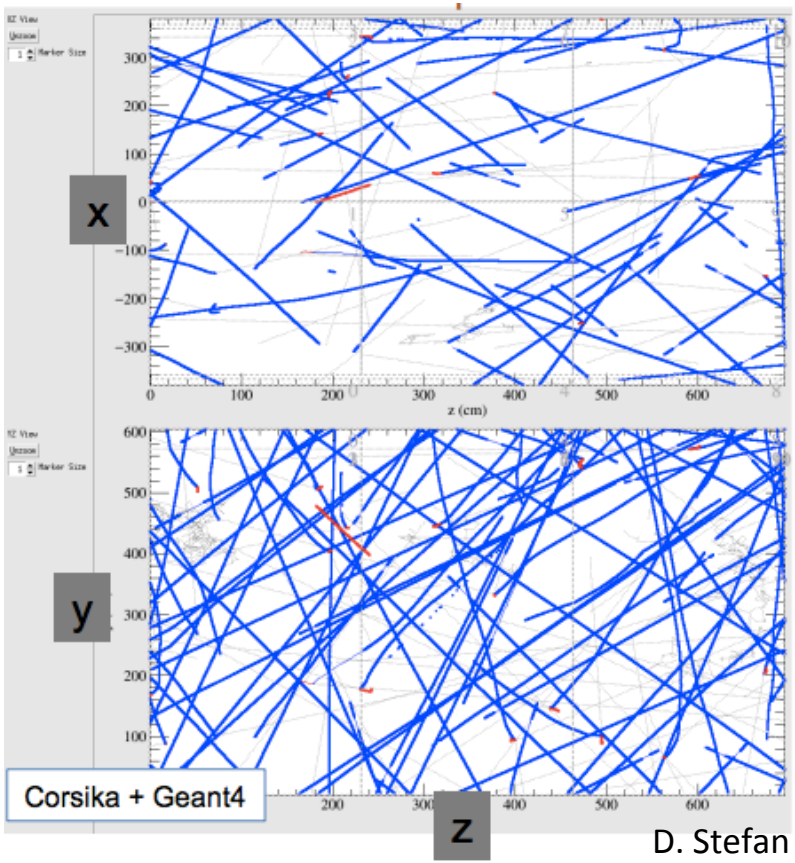
Quick check of muon rates:

- uBooNE: ~160 Hz/m^2
- protoDUNE: ~170 Hz/m^2
- Expected difference based on altitude ~6%
- This exercise uncovered issues with material in outer layers of DUNE/protoDUNE geometry – fix in progress.

Cosmic Simulation in protoDUNE

```
protodune_corsika_cmc:                @local::standard_CORSIKAGen_CMC
protodune_corsika_cmc.SampleTime:     7.25e-3 #0.2 ms (g4 rise time) + 2.25 ms (1 full drift window) + 4.8 ms (readout) @ -70 kv, 114 cm/ms
protodune_corsika_cmc.TimeOffset:     -4.05e-3 #4.8 ms readout should start at -1.6 ms to match data
protodune_corsika_cmc.BufferBox:      [ -250.0,250.0,-250.0,250.0,-250.0,250.0 ]
protodune_corsika_cmc.ProjectToHeight:  865  #height to which particles are projected in cm
protodune_corsika_cmc.ShowerInputFiles: [
            "/pnfs/dune/persistent/users/mibass/corsika/sqShowers/DAT2*.db",
            "/pnfs/dune/persistent/users/mibass/corsika/sqShowers/DAT3*.db",
            "/pnfs/dune/persistent/users/mibass/corsika/sqShowers/DAT4*.db",
            "/pnfs/dune/persistent/users/mibass/corsika/sqShowers/DAT5*.db",
            "/pnfs/dune/persistent/users/mibass/corsika/sqShowers/DAT6*.db"
        ]
```

- SampleTime and TimeOffset must be adjusted for protoDUNE readout window
- ProjectToHeight: Showers start here (just above the top of the cryostat volume in existing geometry because we do not have realistic simulation of building, etc)
- ShowerInputFiles: Files in dCache persistent
- Only private (MB/ETW) implementation so far; will commit fcl files to dunetpc when ready for more general use



D. Stefan

Existing MC Samples

- MCC6 samples:
  - No SCE
  - Particle gun injected at front face of "beam window"
  - Both mono-energetic and $\Delta p/p$-5%
  - Positrons (1 GeV)
  - Kaons (1 GeV)
  - Protons (1 GeV)
  - Pions (200 MeV)
  - Muons (200 MeV)
  - Samples chosen based on needs of reconstruction effort (Robert/ Dorota) and to for first exercise of MCC machinery with protoDUNE
- Privately generated samples:
  - Mono-energetic MCC samples with SCE position effect
  - Additional particle momenta using mono-energetic MCC setup w/ and w/o SCE position effect (BNL summer students J. Larkin and A. Depoian)
    - Positrons (300 MeV, 500 MeV, 1 GeV, 5 GeV)
    - Protons (700 GeV, 1 GeV, 2 GeV, 3 GeV)
    - Pions (300 MeV, 500 MeV, 1 GeV, 5 GeV)
    - Muons (500 MeV, 1 GeV, 5 GeV)
  - Small (100 event) test samples of cosmic ray events

All samples stored on FNAL DUNE disks (/dune/data or /pnfs/dune/scratch). Some samples have been copied to CERN for local access during workshop. Please contact Elizabeth and Dorota for locations. Additional samples easily available upon request.

Planned MC Samples (Near Future)

- More “official” cosmic sample
  - Once details of readout window and geometry updates in place
- Cosmic overlays with particle gun (“beam”) events
  - LArSoft has capability of using multiple generators in the event generator step, so overlay of particle gun and CORSIKA events can be done nearly trivially
  - Cosmic generation very “expensive” (compare hundreds or thousands of particles in the TPC volume for cosmic simulation to handful of particles from single particle gun event) so will produce samples with cosmic overlays for requested samples only
  - Makes sense to wait on this until reconstruction of test cosmic samples more fully understood
- Samples with space charge effect implemented in recombination factor
  - Once LArSoft implementation complete
- Goal to have all of the above plus geometry improvements in place for next MCC challenge

Summary

- Work on protoDUNE-specific simulation began fall 2015
  - LArSoft protoDUNE services defined
  - Geometry with many details implemented
    - Improvements ongoing
  - Space charge effect simulation
    - Position distortion implemented
    - Recombination effect in progress
  - Cosmic simulation
    - CorsikaGen implemented
    - Overlays coming soon
  - Continued addition of features and improvements to existing features will be driven by analysis needs
- Many MC samples exist – please start to analyze them!
- Additional MC samples available on request – please ask!