GArSoft and Integration Thoughts

Tom Junk, Leo Bellantoni, Thomas Campbell, Eldwan Brianne, Gavin Davies

DUNE ND Workshop

May 26, 2019

Getting Started with GArSoft

- GArSoft shares a lot of development environment and tools with LArSoft
  - git repository hosted on Fermilab's Redmine Server
  - mrb for building code (uses cmake)
  - *art* framework
- See the wiki at https://cdcvs.fnal.gov/redmine/projects/garsoft/wiki/Getting_Started_Developing_using_CVMFS_git_and_mrb

which starts you from a bare login and an empty directory and gets you generating, simulating, and reconstructing GENIE events in the MPD, and making a ROOT analsyis ntuple.

GArSoft Beginning to End

- Generators
  - Particle Gun
  - GENIE
  - CRY
  - Radiologicals (not yet tested)
- Simulation:
  - GEANT4 (uniform B assumed at the moment)
  - GArG4 module patterned on LArG4
  - Outputs energy deposits (recently added to LArSoft)
  - Also outputs channel waveforms: gar::raw::RawDigits, which are zero-suppressed. No noise. Small event sizes.

DUNE-Doc 13933

GArSoft TPC Simulation

- Drift model is a copy of LArSoft's
  - Nearest channel geometry – charge diverts over cracks and inward from edges. To do – deaden the cracks and edges to simulate cover electrodes.
  - No Lorentz angle or space charge yet. Uniform E and B
  - Longitudinal and Transverse Diffusion simulated numerically by sampling from Gaussian distributions
  - Drift velocity and diffusion input parameters from Magboltz
  - No induction field response on the pads – charge "collects"
  - This may overestimate response to tracks that point straight at pads
  - Less of an issue than with wires since tracks have to point along E rather than being in the plane containing E and a wire (a la MicroBooNE or DUNE FD)
  - Still a concern as low-energy electrons spiral along B (and E)

GArSoft Track Reco First Steps

- Hit finding
  - gar::rec::Hit
  - hits belong on one and only one pad. Necessary for BackTracker to work, as it makes this assumption (carried over from LArSoft)
  - zero-suppression threshold is a "hit finder".
  - Hit finder module refines hits – if a waveform drops below 20% of its peak it'll start a new hit.

- Hits → TPC Clusters (also done by ALICE)
  - gar::rec::TPCCluster
  - Nearby hits grouped together
  - Charge-weighted centroid and RMS calculated

Vector Hit Finding ("tracklets")

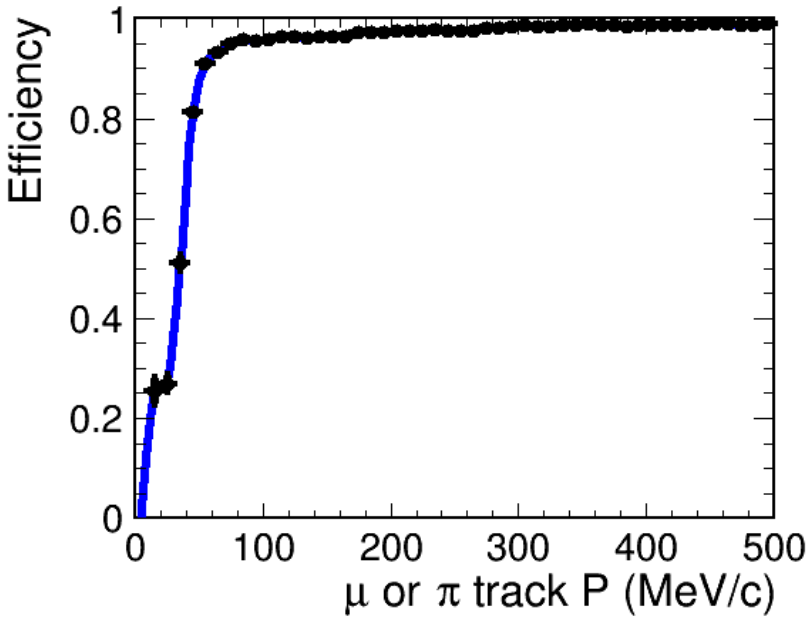
- gar::rec::VecHit
- Line segments fit in 3D
- Maximum length 20 cm (adjustable)
- TPCClusters added to line segment candidates if they are close to them
- Line segments started by close pairs of TPCClusters
- Two passes performed – remove highest chisquare hits and attempt to reassign them to other Vector Hits
- Vector hit contamination near the primary vertex is an issue ("hit stealing")
- Tendency to follow pad-row geometry in places (to fix)

Pattern Recognition & Fitting

- Pattern Recognition: grouping vector Hits into track candidates
  - Output is gar::rec::Track data products and associations with TPCClusters and VecHits
  - Matching in 2D (circles) and 3D (constant dip angle) of nearby vector hits
  - Currently has loose cuts so it's highly efficient, but it does break electron curlers up (cuts too tight), and it stitches together two legs of a conversion or a V
- Track fit: Kalman Filter
  - Output is a second set of gar::rec::Track data products and associations with TPCClusters
  - Fit is performed twice, once from either end of the track (vertex is not yet known)
  - Energy Loss and Scattering → track parameters are different on the two ends
- Tracking takes less than 5 secs of CPU/event

Tracking Performance: $\pi^{\pm}$ and $\mu^{\pm}$

Estimated using Leo's sample of ν_{μ} events with the optimized LBNF FHC spectrum



Charged pion and muon tracking efficiency

Electrons are similar, but including them produces a kink at 20 MeV (bigger than the one that's there).

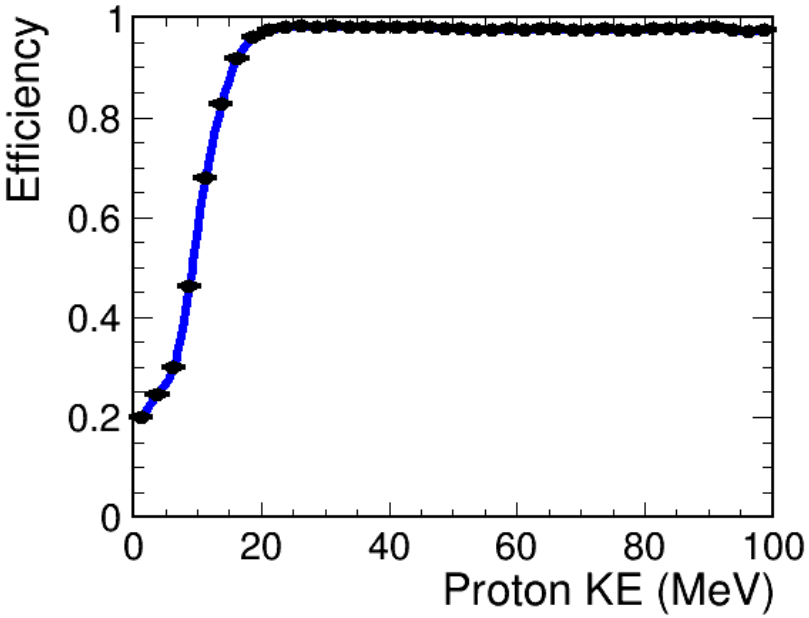
Low-energy electrons curl around – only partial efficiency for them

Low-energy pions and muons stop – have a track length cut of 20 TPC Clusters

Protons with P<150 MeV have very little KE and thus stop quickly – plot their efficiency vs. KE

Tracking Performance: Protons

Estimated using Leo's sample of ν_μ events with the optimized LBNF FHC spectrum



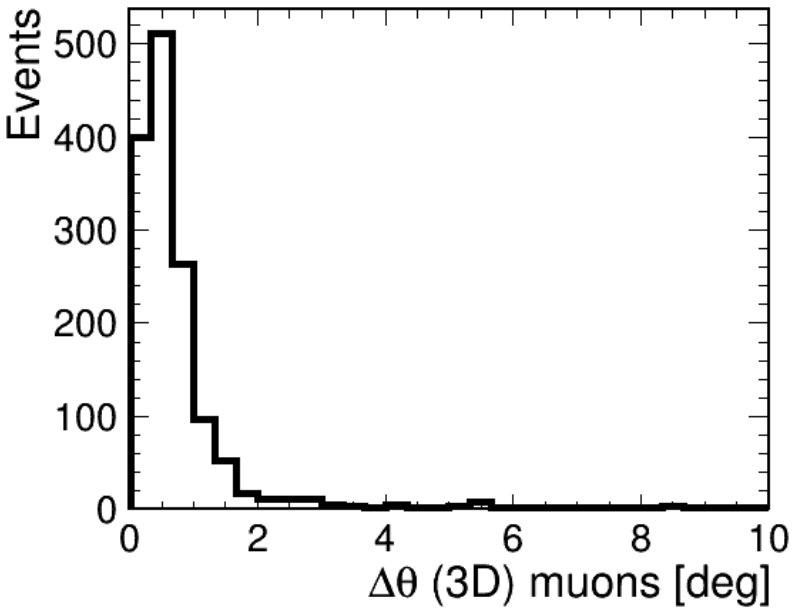
Very short track efficiency overestimated near a dense primary vertex due to combinatorics – fake matches.

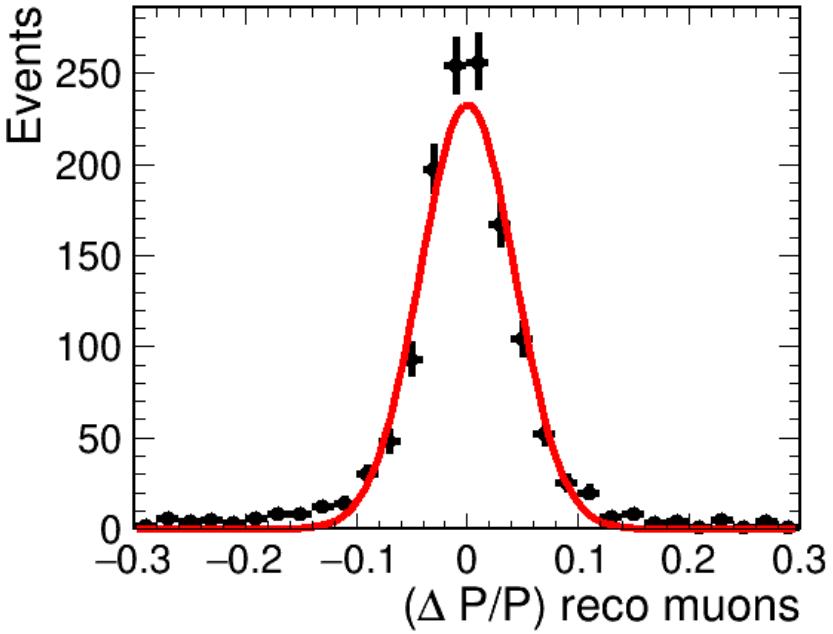
Efficiency should go to zero at KE=0.

Work in Progress – Optimizations will improve this

Tracking Performance: Muon Angles and Momenta

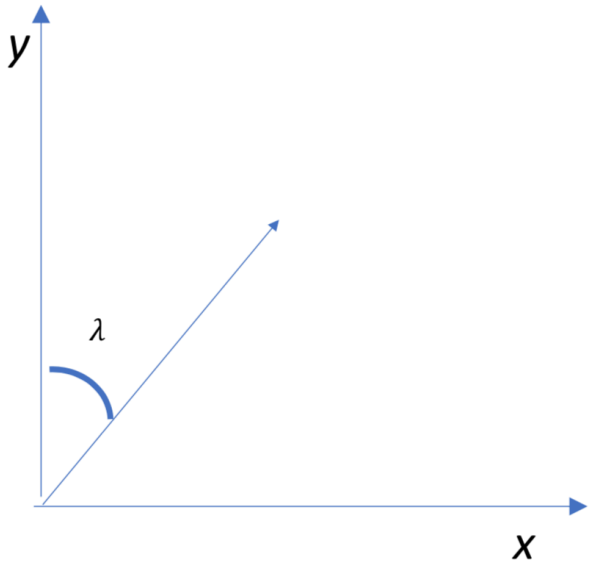
Work in Progress – Optimizations will improve these

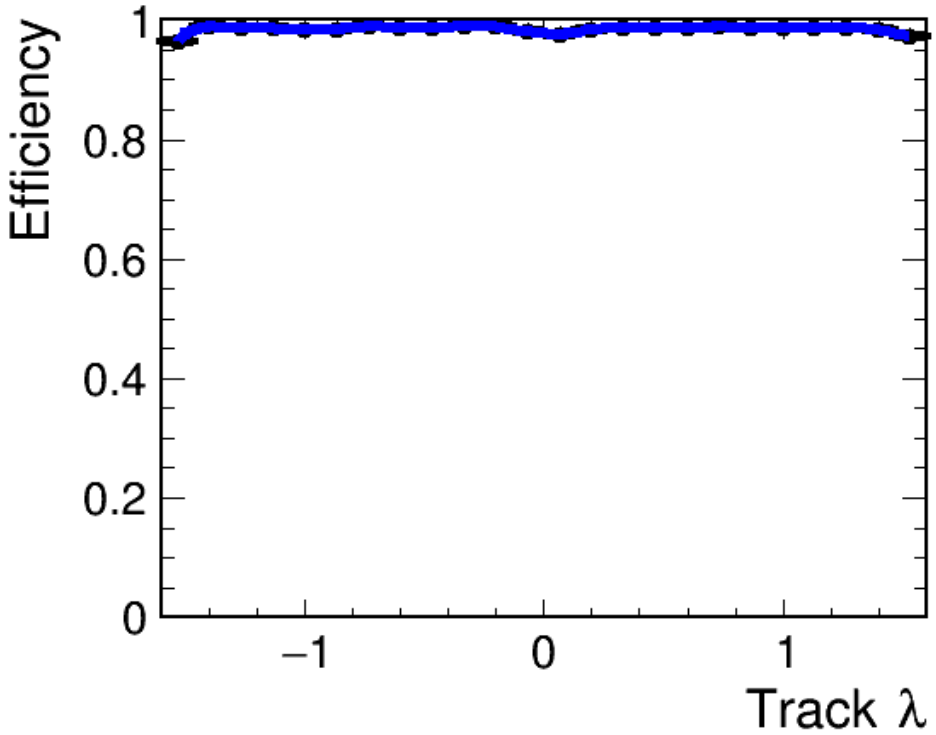




~1 Degree angular resolution, and ~4.2% momentum resolution

Tracking Performance: 4π Coverage

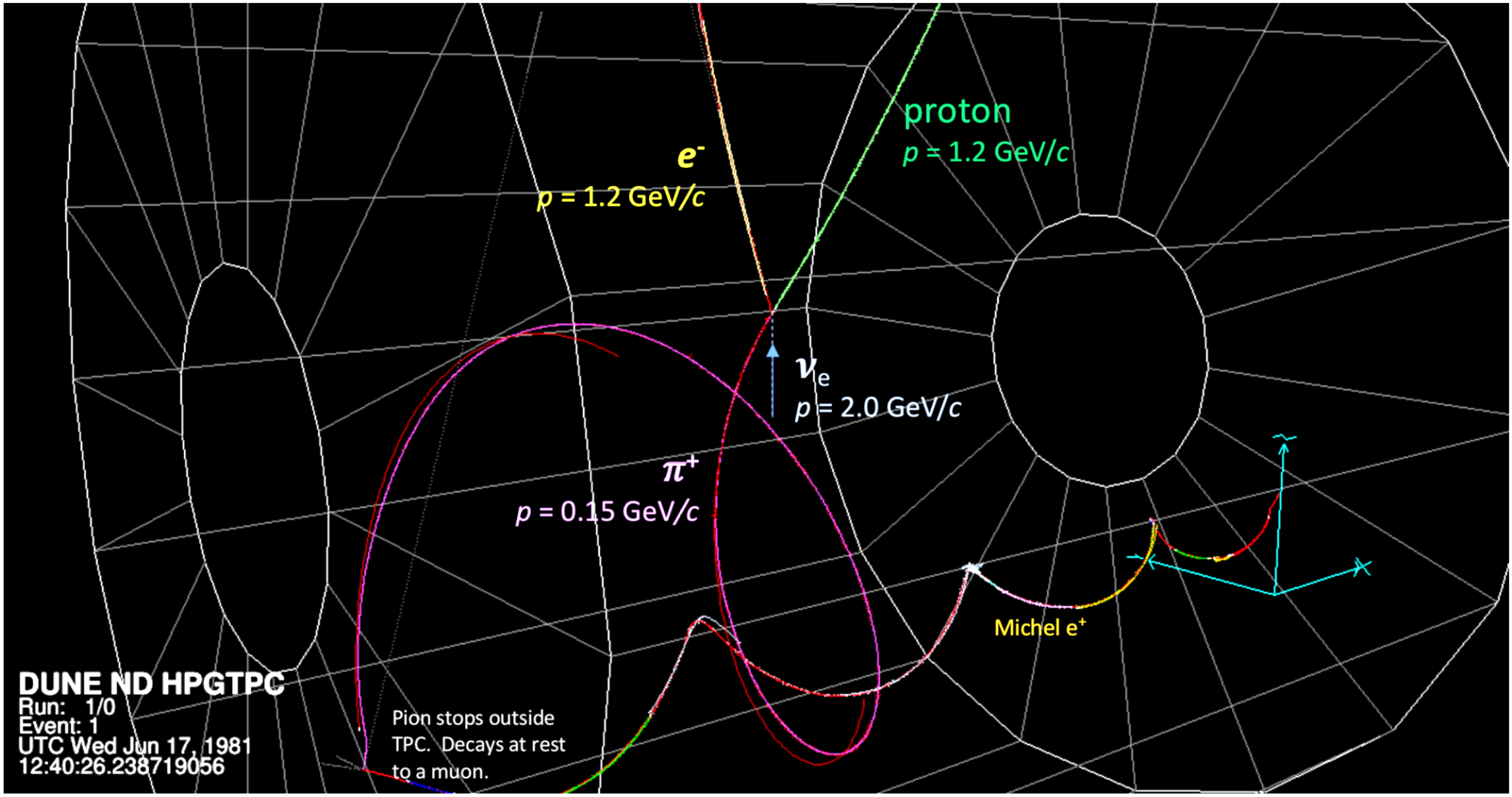




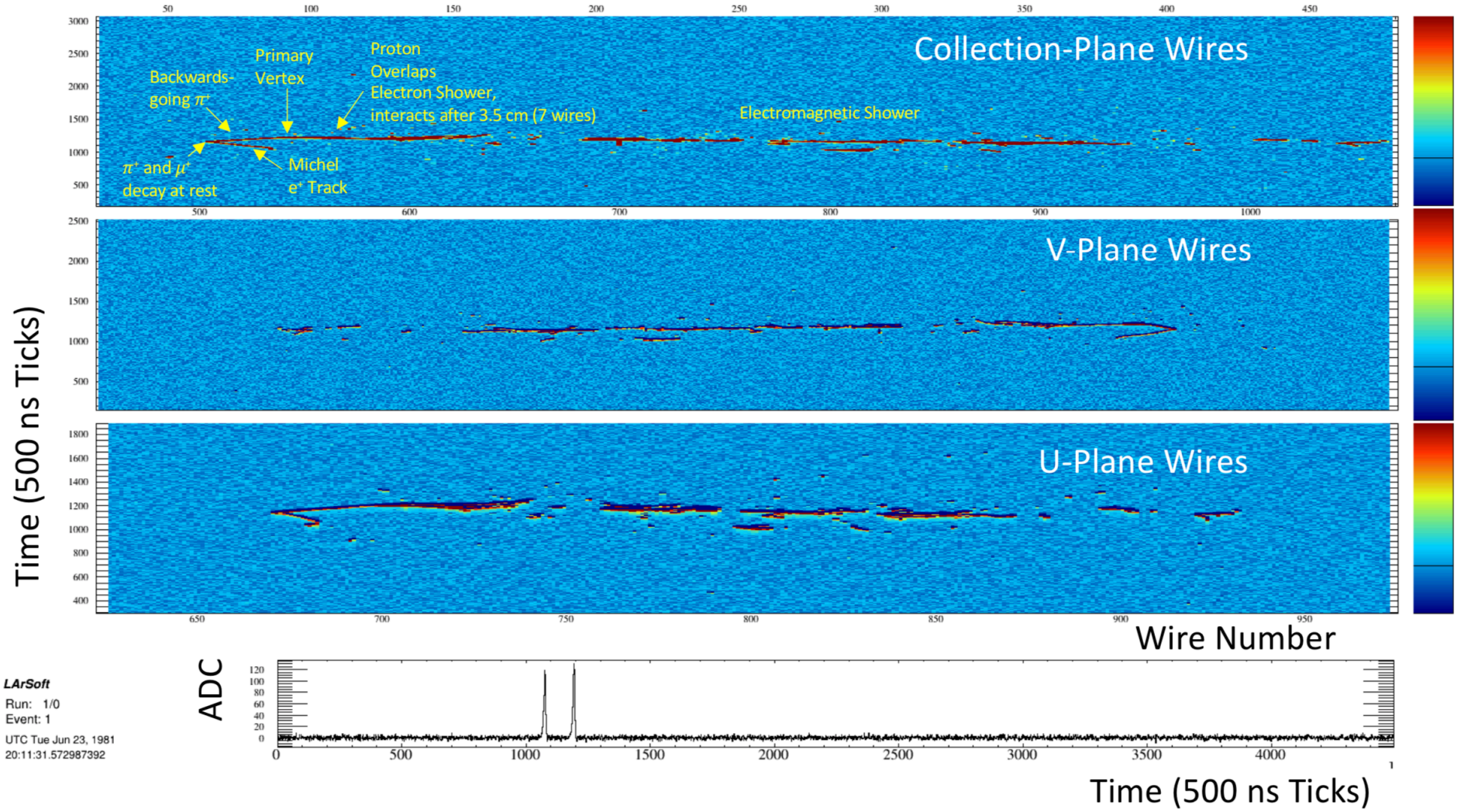
All tracks with momentum > 200 MeV/c (protons are inefficient for momenta below 150 MeV/c)

n.b. Charge modeling on the pads is naive – induced signals will be less for trains of charge arriving on the same pad over lengths of time

A Large Looper Found & Fit



The Same Event in LAr

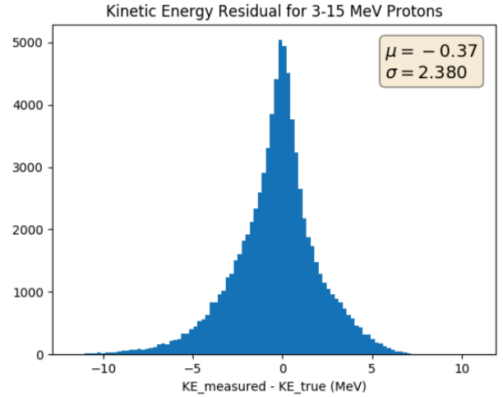


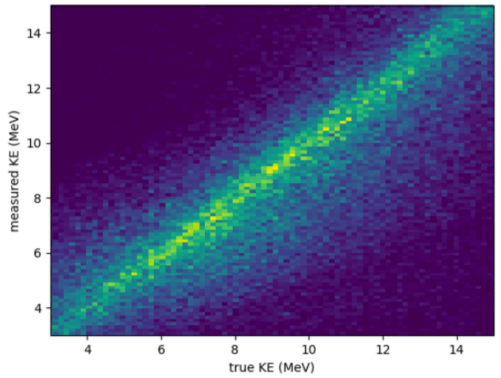
CVN selects it as a ν_e CC with Ereco = 1.914 GeV (2.0 was true)

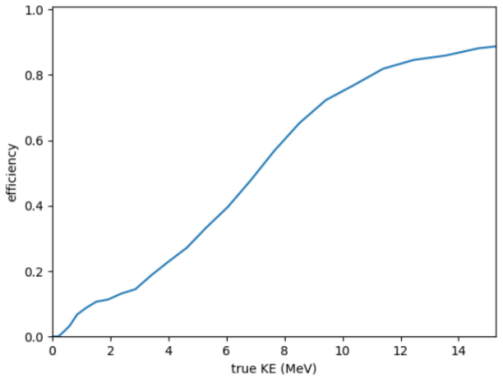
Finding Short Tracks Near the Primary Vertex

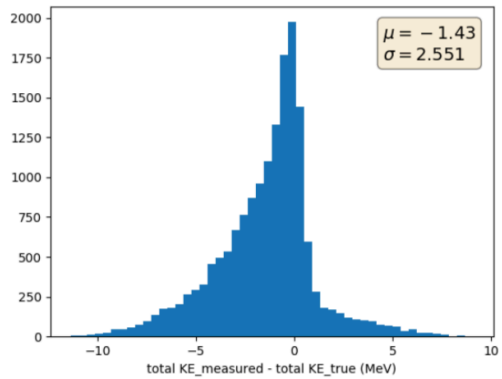
- Thomas Campbell's work
- RANSAC line finding + Neural Network for p/pi separation and energy estimation

Results





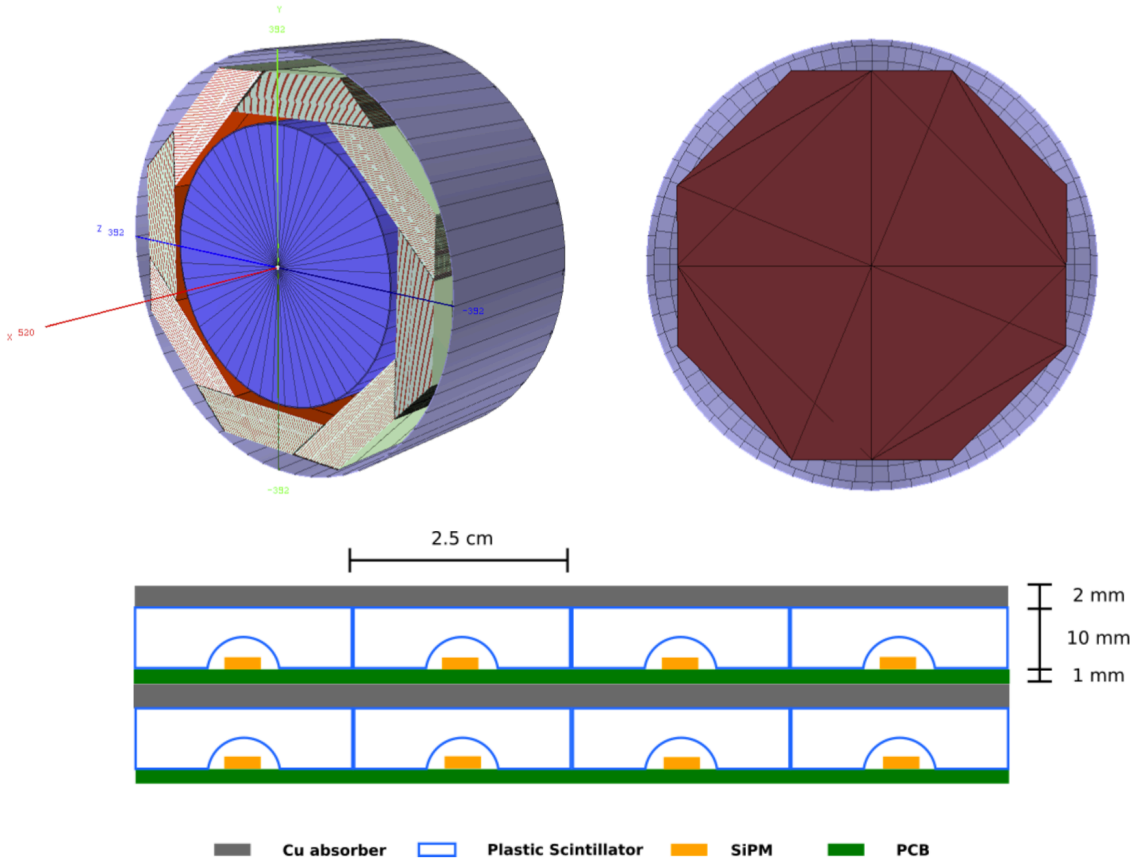




GArSoft ECAL Simulation

Eldwan Brianne

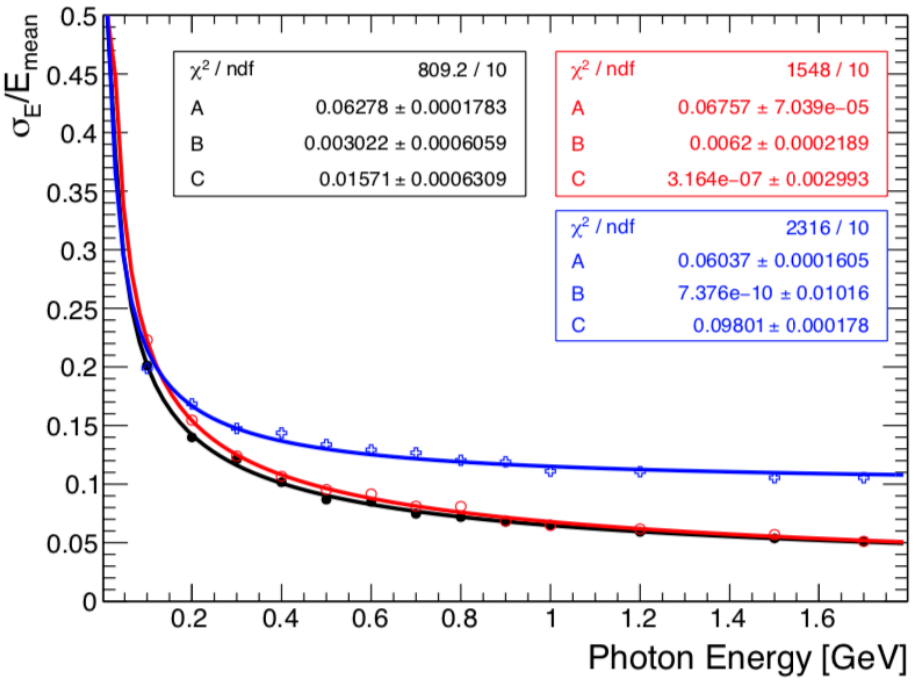
- Same GEANT4 step as the TPC
- Produces CaloEnergyDeposit data products



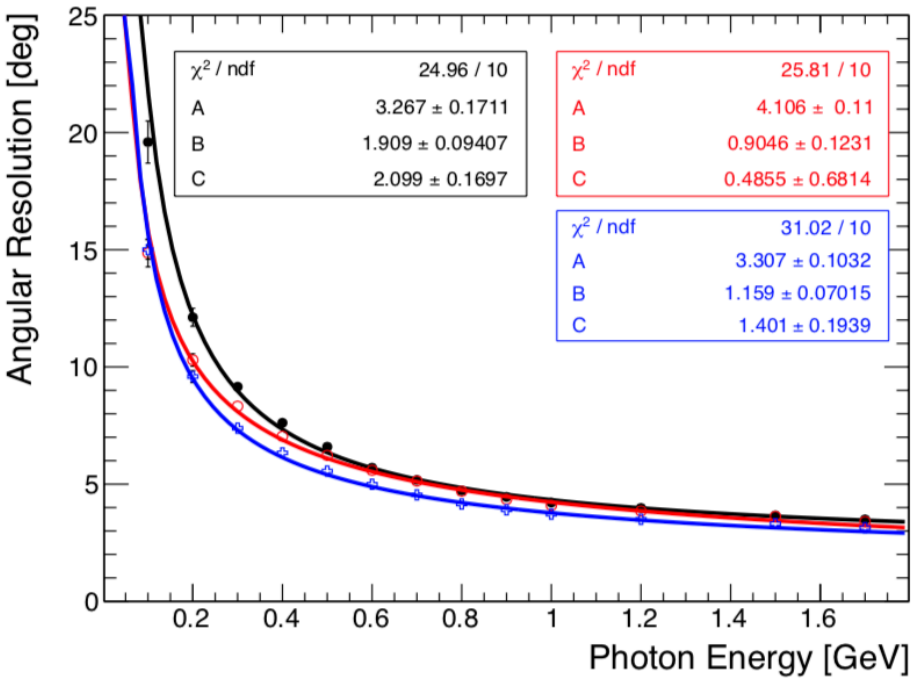
Eldwan Brianne

ECAL Performance

Energy Resolution



Angular Resolution



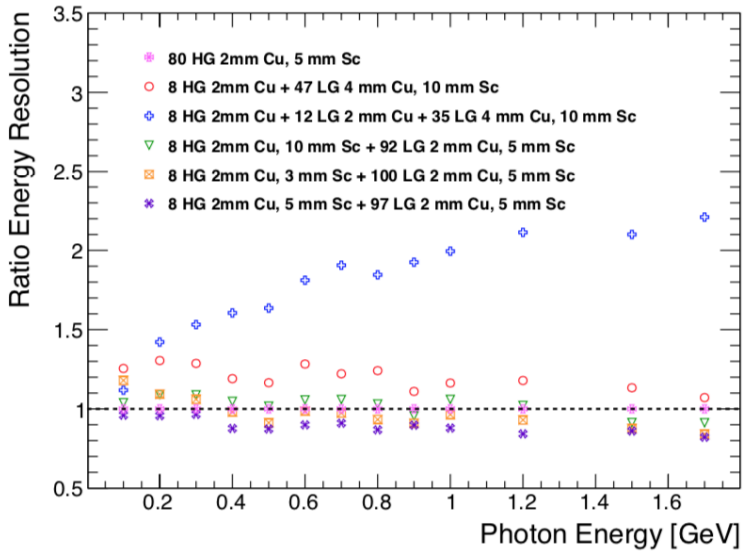
6%/Sqrt(E/GeV) is expected to be achievable

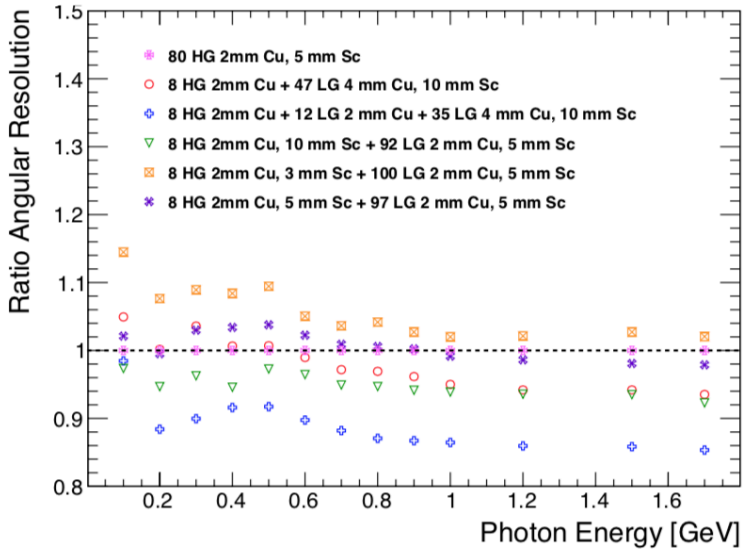
(3.85/Sqrt(E/GeV) ⊕ 2.12) degrees is expected to be achievable

Simulation studies.

Full comparison

- A lot of models compared! (~12 models)
- **To take away**
  - Angular resolution dominated by front layers → granularity in the back layers does not matter much ➠ *strips can be used*
  - Thinner absorber with *small Molière radius* in the front is preferred for angular resolution
  - *Shower containment* is important for high energies ➠ more layers or thicker absorber in the back
  - Thicker scintillator in the front helps in the angular resolution
  - *Pressure vessel thickness* (see backup) needs to be kept below 1 X_0 to keep energy resolution below 6%/√E



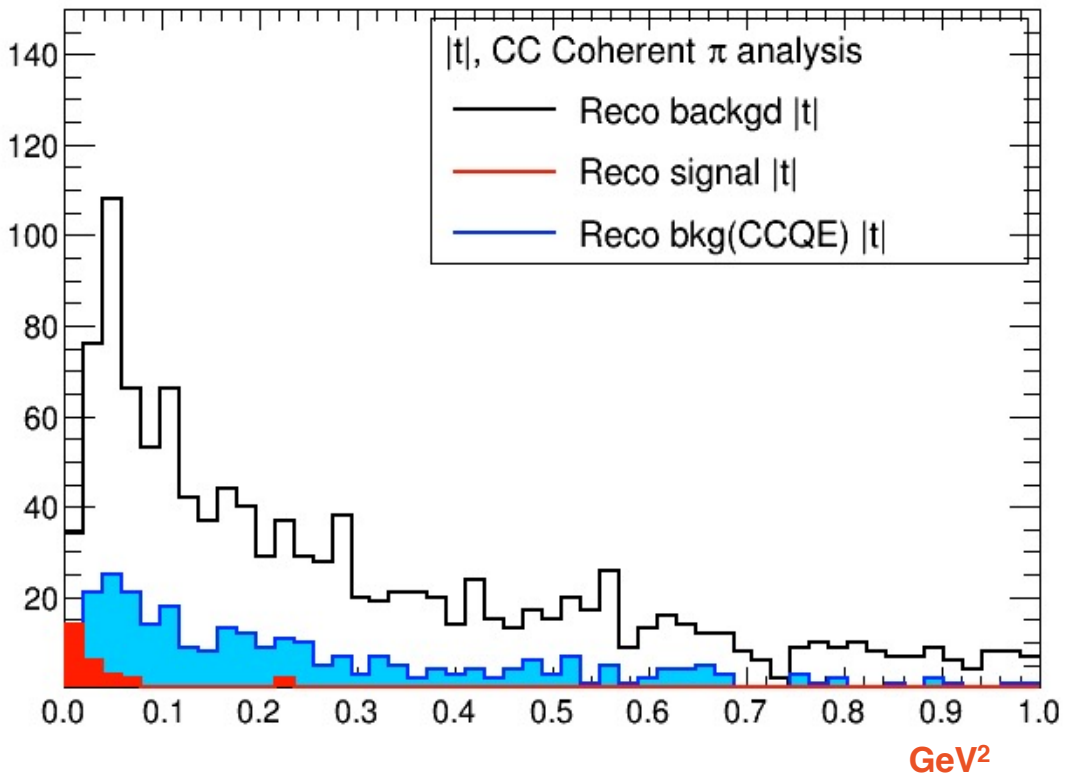


Eldwan Brianne

Coherent π^+ in garana

Using garana *in a physics analysis instigates debugging and developing it in ways that might not happen otherwise. Want to be able to rely heavily on* garana *for CDR*

- Rewritten track-for-vertex selection
- Implemented PID via dE/dx
- Got realistic flux implemented (almost) correctly, including ν interactions in ECAL
- General ana-code upgrades



WORK IN PROGRESS

Red is signal (not stacked)
Blue area is CCQE after dE/dx cut (~50% suppression)
White area is combo of 1-π^+ physics background and mistracking or misvertexing
Track quality cuts not yet optimized
That's only 2 days of running simulated – fitting shapes might be interesting

L. Bellantoni/FNAL

Fermilab DUNE

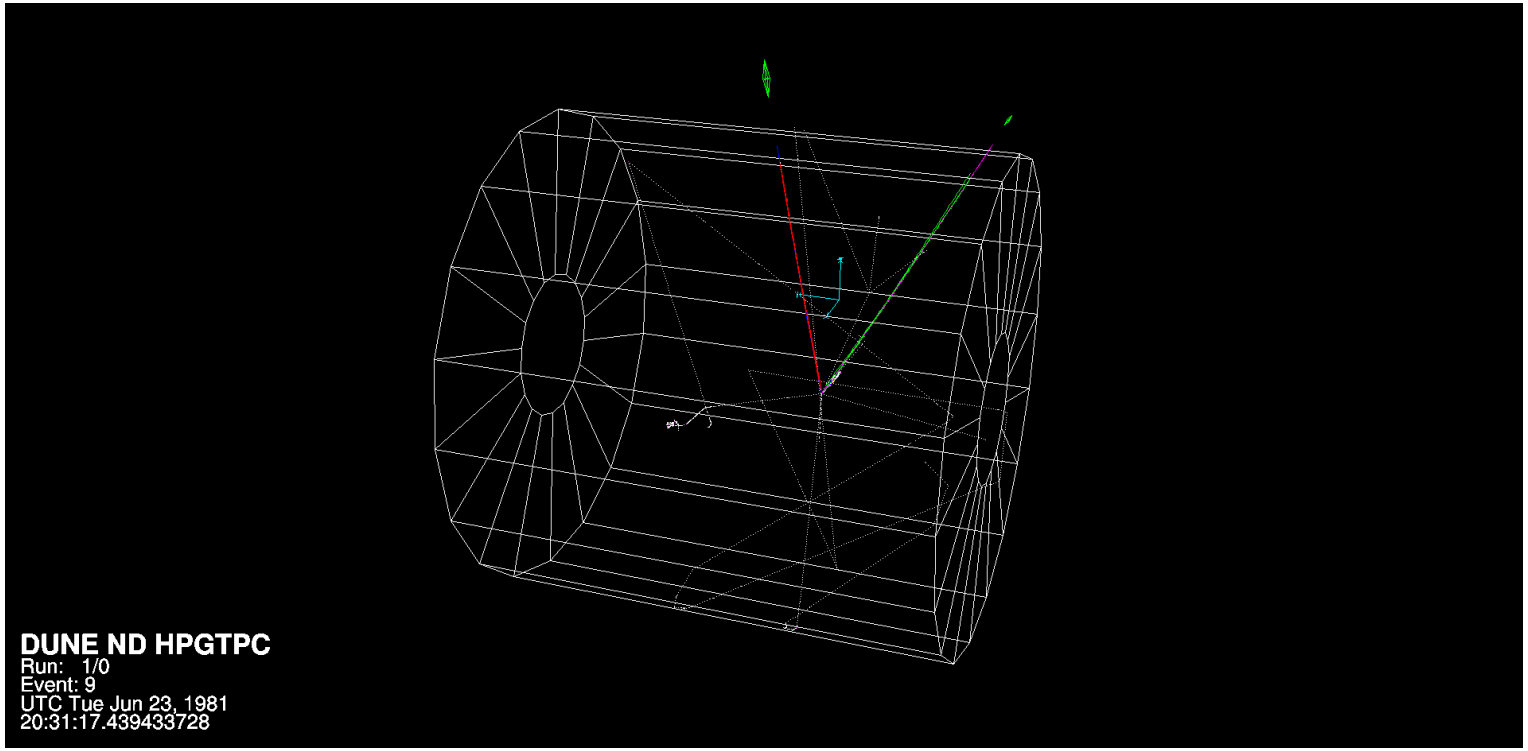
Some Tracking To-Do Items

- Incorporate realistic field response (take ALICE's)
- Deaden more channels in the gaps and edges (I filled up nearly all the space with channels but cover electrodes block them)
- Tune up the hit finder and TPCCluster finder with new field response
- Improve Vector Hit finding and pattern recognition
  - lower-momentum loopers
  - Photon conversions
- Tune Fitter
- Extrapolate to ECAL & evaluate performance
- Second-pass assignment of hits to tracks once vertex is found
- RANSAC and NN short track work near the primary vertex

Analysis To-Do Items

- Make plot of energy spectrum of photons entering TPC from the ECAL (from MC truth)
  - Rates and Energy spectra separately for conversions and Compton scatters
  - Conversion electron energy spectrum and asymmetry
- Estimate fake rate of photons → electrons for LBNC
  - Requires scanning of event displays for now
- Estimate pi-mu separation (vs energy)

ECAL Clusters in the EVD

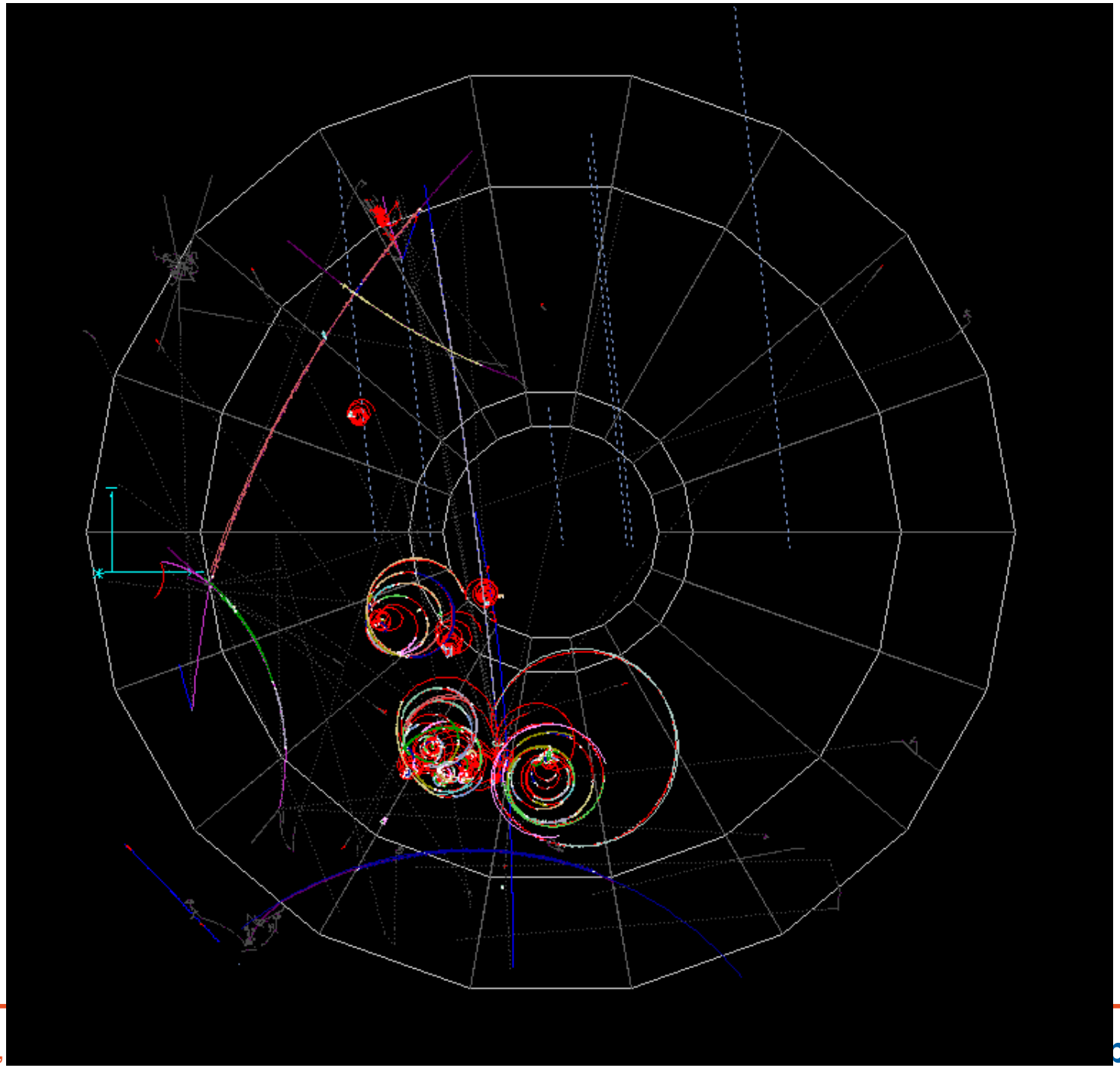


Octaheadron with length proportional to energy drawn. Cluster location at inside tip
Octahedron oriented along cluster major axis

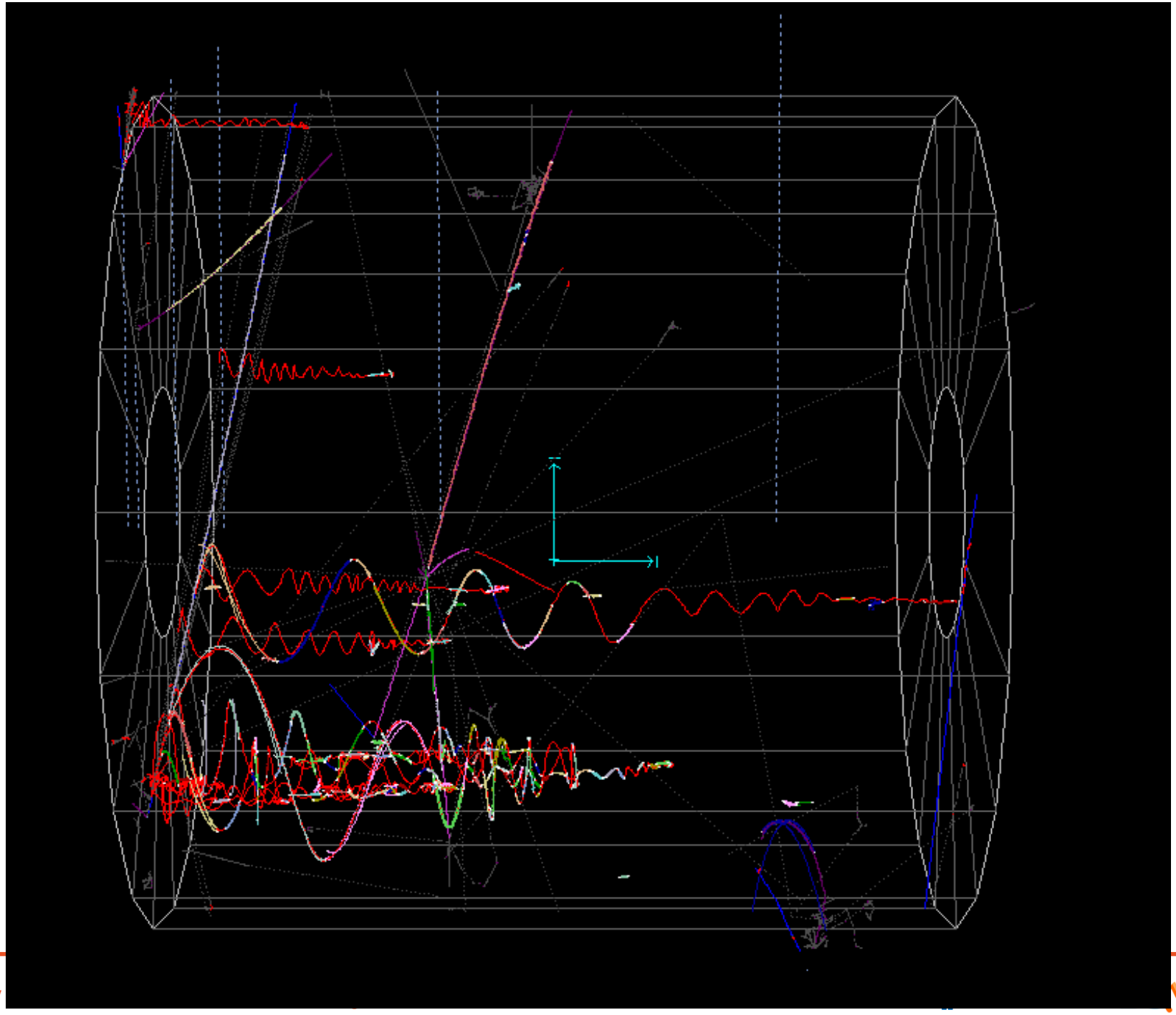
Red track is a muon (blue MC particle), Green track is a proton.

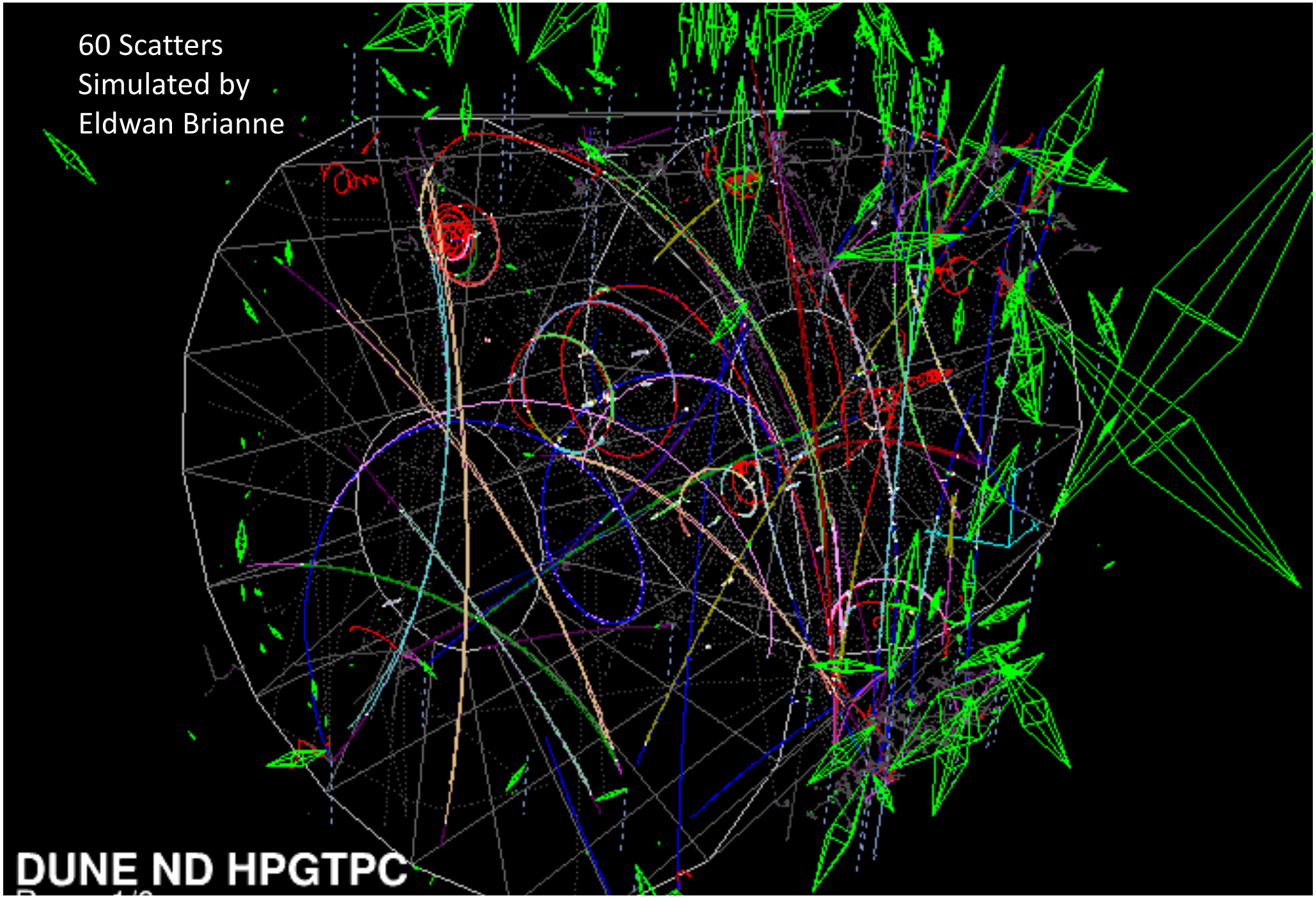
A 10-interaction event with conversions end view

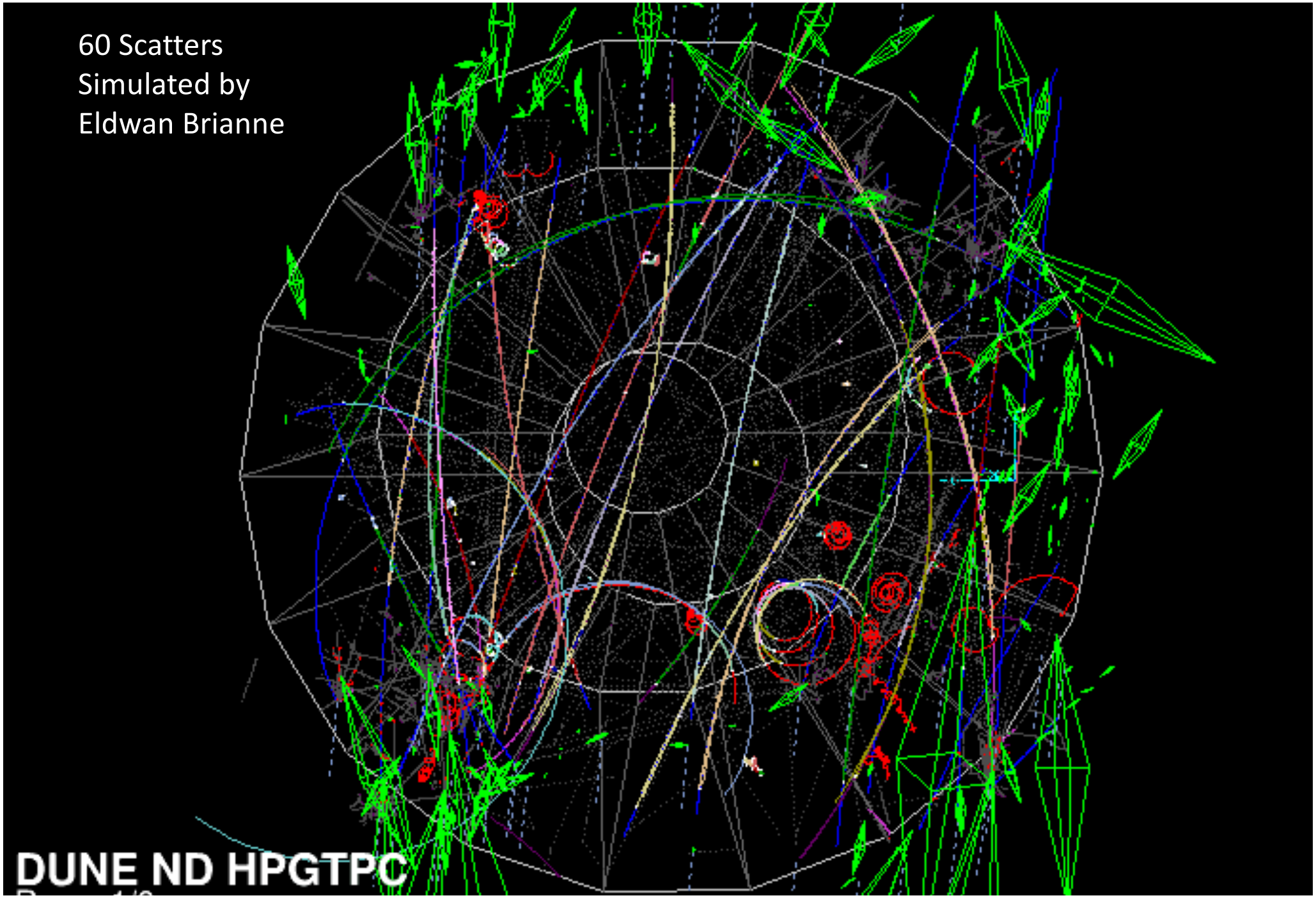
No ecal in this display



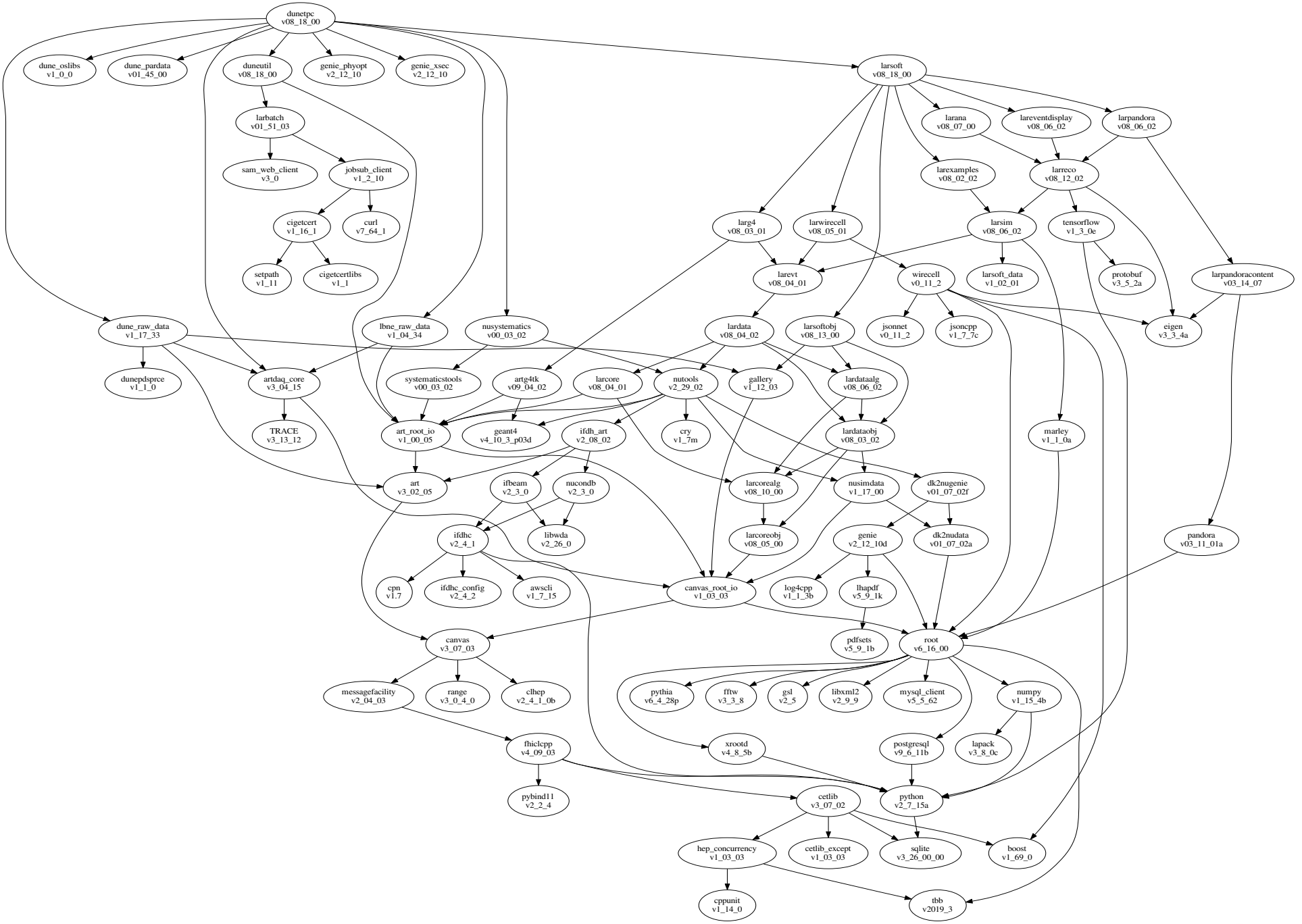
A 10-interaction event with conversions side view



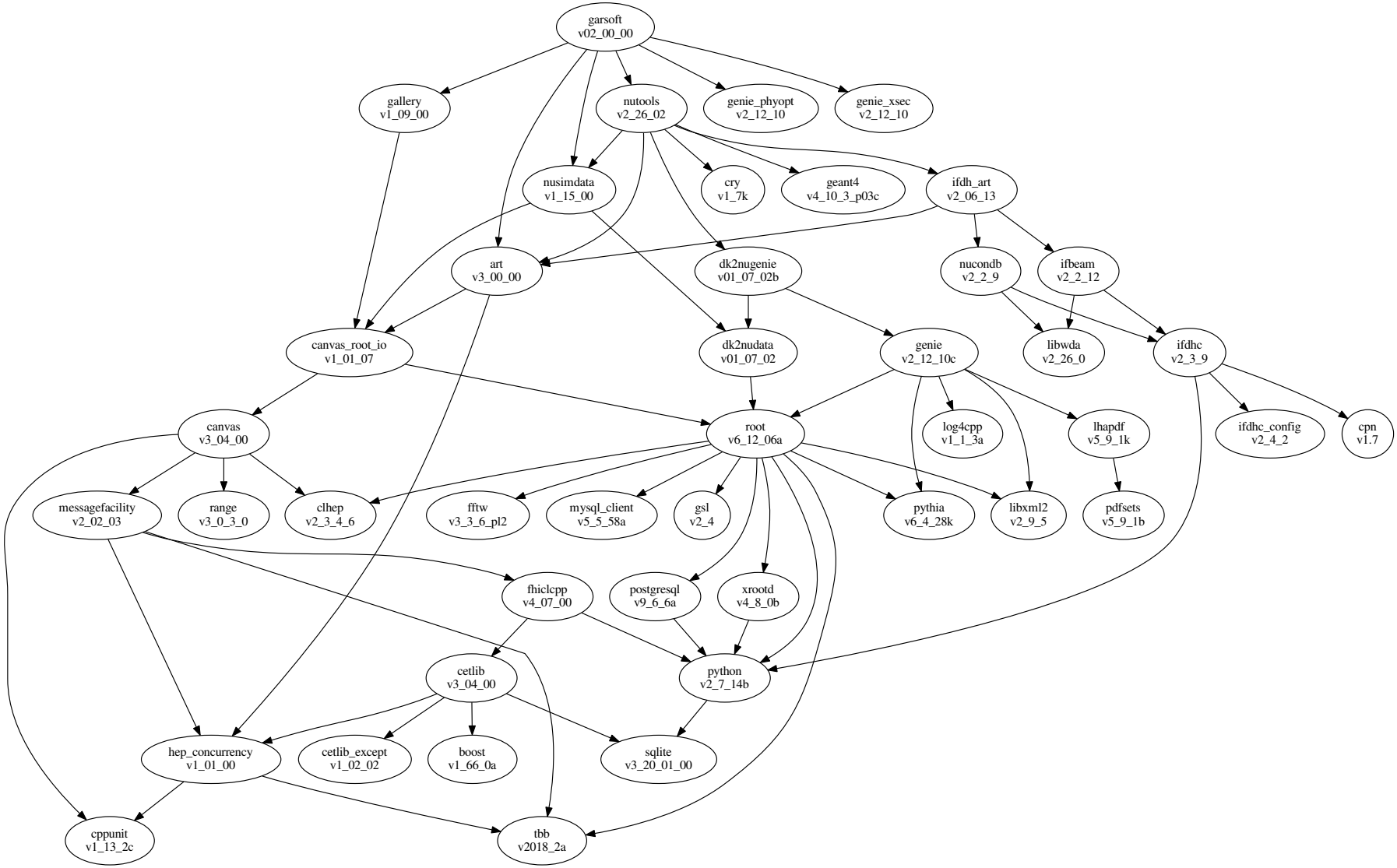

60 Scatters
Simulated by
Eldwan Brianne
DUNE ND HPGTPC


60 Scatters
Simulated by
Eldwan Brianne
DUNE ND HPGTPC

The dunetpc Dependency Diagram



The GArSoft Dependency Diagram



Integration Thoughts

- Running GArSoft and LArSoft modules in the same job "should" be possible
  - Both are based on the *art* framework
  - *art* loads modules dynamically based on FHiCL configuration
  - Data products for GArSoft have names in the gar namespace. e.g. gar::raw::RawDigit, so as not to collide or be confused with raw::RawDigit in LArSoft
- But there is some work to do to keep it all together
  - Dependency trees have to match. Must use same version of *art* for example. "A tree with two trunks"
  - GArSoft is updated to *art* V3. LArSoft has followed a few point releases since then but they involve few breaking changes.

Integration: Easy Issues First

Running detector-specific simulation and reconstruction are all independent pieces – modules work on independent data.

- channel response
- data output from sim job and readin in reco job
- noise filtering
- deconvolution
- TPC clusters and hit-finding
- tracking
- shower reco
- calorimetry

Integration: Harder Issues

- Unified GEANT4 simulation
  - Current modules: LArG4 and GArG4. Consume MCTruth data products, make sim::SimChannel and energy deposits
  - particles produced in LAr -> GAr -- one can imagine running LArG4 first and then piping particles that come out of the LAr as MCTruth for GArG4, which gets run second.
  - Particles produced in GAr traveling back into LAr?
    - Either need to iterate this, or run a unified GEANT4 step
- Unifying the GEANT4 step means having a single geometry description GDML file (or files), and calling GEANT4 once to follow particles back and forth.
- Hans Wenzel's new Energy Deposits in LArG4 look a lot like Brian Rebel's solution in GArG4.
- Data products have different names but that's okay

Integration: Event Display

- The three-detector ND Complex will have particles exiting one detector and possibly going into the other two.
- Visualizing the events will be useful in developing (traditional) reconstruction and track-matching algorithms
- Currently we are working independently

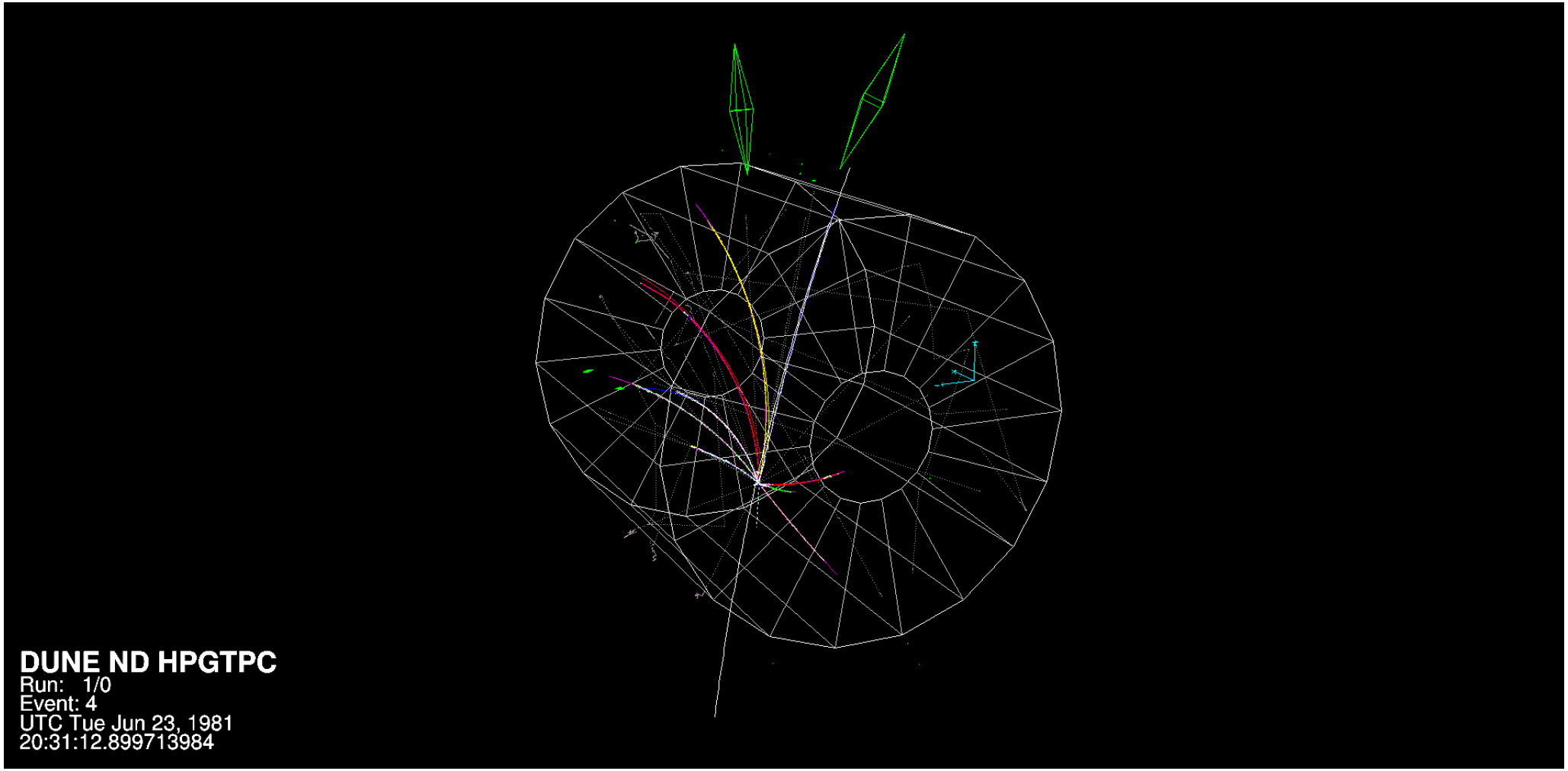
Integration with 3DST-S

- Less understood on the MPD software side how the 3DST-S would fit in.
- GEANT4 step needs to be unified with ArgonCube and MPD for reasons explained before
- Off-axis positions are interesting – five combined geometry descriptions may be necessary, since 3DST-S does not move off axis.
- 3DST-S has gas TPC components. May want to re-use GArSoft algorithms, as they are intended to be homogeneous and isotropic. GArSoft assumes pixel readout however.

Definition of an "Event"

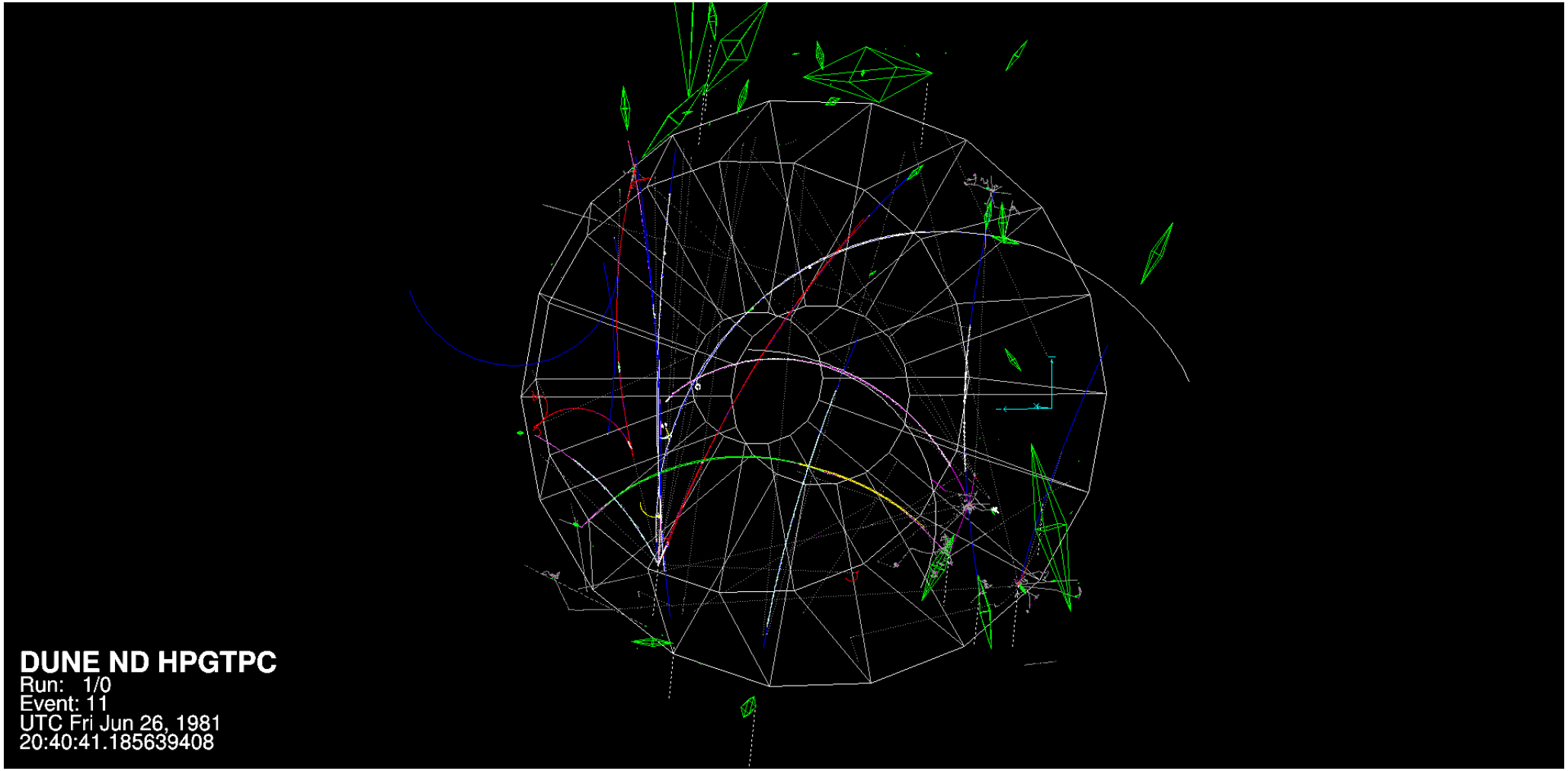
- *art* handles events as the smallest bit of independent data
- We associate these with triggered detector readouts.
- The entire complex will want to share a single trigger
  - beam spill signal from LBNF
  - Random triggers for background constraints

ECAL Clusters in the EVD



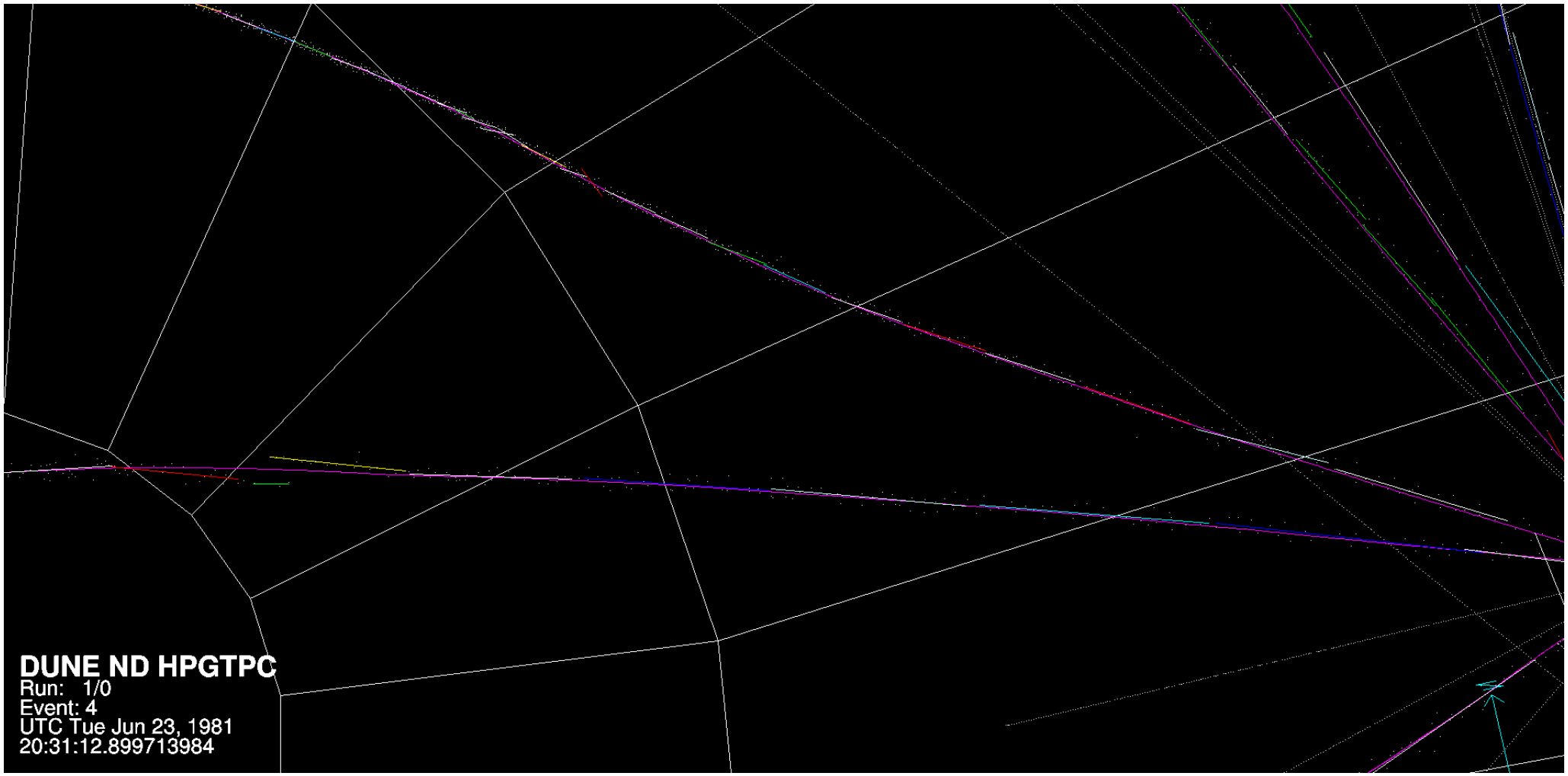
An GENIE event. ECAL Clusters shown in green.

An MPD event with 10 Interactions



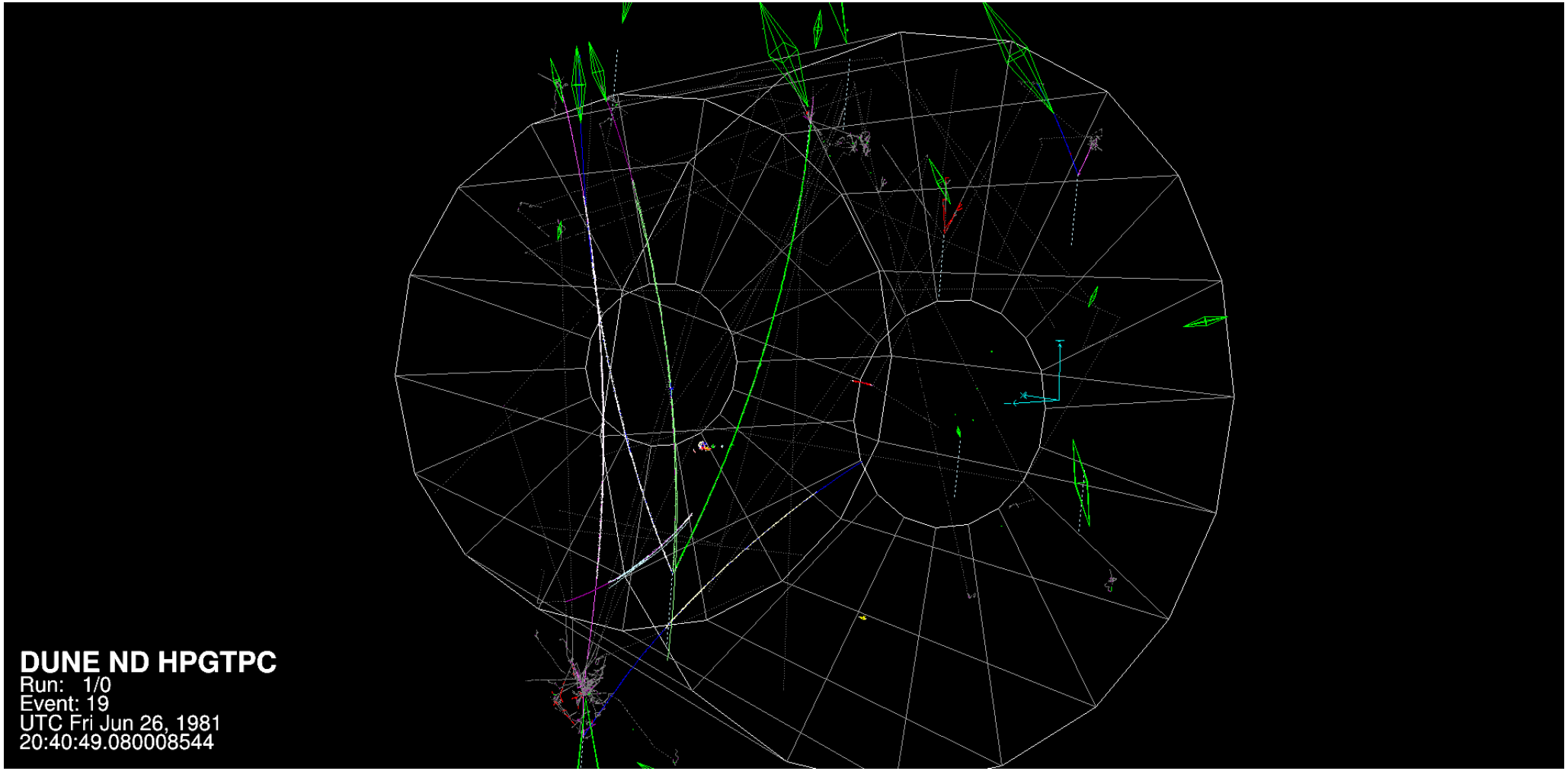
About twice the pileup we expect on average. But well within the distribution.

Vector Hits in the Event Display



Many thanks to Gavin Davies!

Another 10-interaction Event



Extras

Recent Work

- New Tracking Note: DUNE-Doc 13933. Authors: T. Junk, E. Brianne, T. Campbell, L. Bellantoni
- Describes (briefly)
  - Software structure
  - Event generators
  - Detector simulation
  - Hit finding, hit clustering, pattern recognition, fitting
  - Track parameters and data products
  - Tracking performance
- ECAL note to be produced separately

Tracking To-Do List from January 29, 2019

- Figure out how to keep the pattern recognition from stitching tracks across the primary vertex. Partially addressed, but still a problem.
- Look at split tracks – getting better, but some tracks still split especially near the primary vertex
- Use the backtracker to make track completeness and purity metrics. Expect completeness to be more of an issue, perhaps purity for very short stubs at the vertex.
- Address curlers.
- Address low-momentum tracks near the vertex: Got more of these, needs more work.
- Vertex finding measurement uncertainties
- Event display updates – vector hits, pickability, truth labels, etc.
- Extrapolate to ECAL
- Try with more complex events (pileup with particles coming from ECAL)
- Look at performance of hit clustering algorithm – efficiency and resolution T. Campbell has started this
- Get the CRY and Radiological generators working
- Add realistic electronics response and noise
- Explore Machine-Learning Techniques
- Write technote

Addressing Lower-Momentum Particles

- Pattern recognition sometimes correctly clustered vector hits into long, looping tracks, but the fit gave crazy results.

- Rewrote the hit-sorting algorithm. Now start at one extreme end and "walk" along the track

- Still some work to do to improve tight curlers, scattering and delta-rays

Recent Work

- Tracking Module split into three pieces
  - Vector Hit Finder
  - Pattern Recognition (VH's → tracks)
  - Track fitter
- Kept pre-clustered TPC Hits as "hits" and renamed the TPC Clusters from "hits" to "TPCClusters".
  - Reason: The BackTracker wanted hits associated with DAQ channels and TPCClusters weren't. Should make BackTracker work better but this hasn't been tested yet.
- Vector-hit finder now does a second pass, reassigning hits
- CRY generator now works. Makes just one muon at a time
- GENIE Fluxes more realistic
- Drift velocity increase due to quenching gas (pure GAr is very slow, but even 10% methane speeds up the drift velocity by a lot)
- Analysis Tree updates by Leo

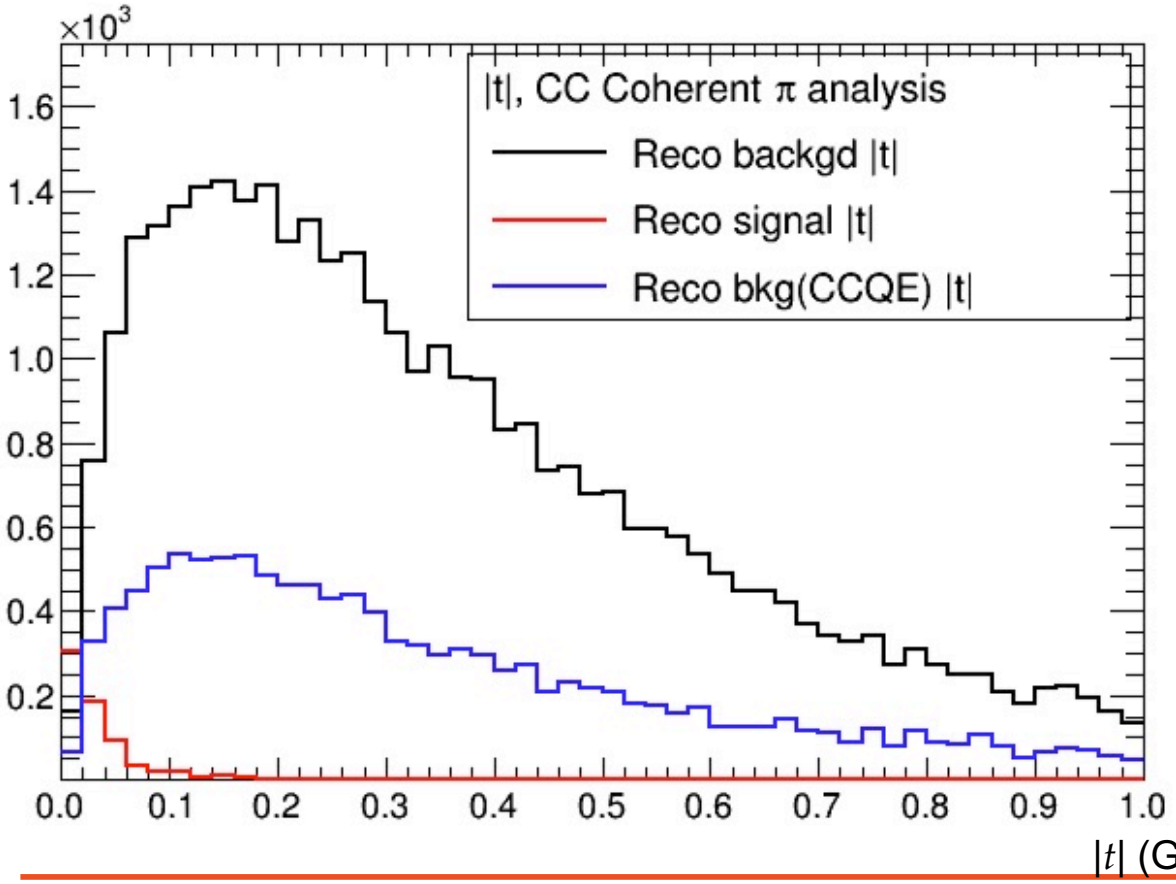
To-Do List Status

- Figure out how to keep the pattern recognition from stitching tracks across the primary vertex. Partially addressed, but still a problem. Still to do
- Look at split tracks – getting better, but some tracks still split especially near the primary vertex
  Still to do
- Use the backtracker to make track completeness and purity metrics. Expect completeness to be more of an issue, perhaps purity for very short stubs at the vertex. Resurrecting pre-clustered hits makes this possible.
- Address curlers. Addressed. Need to work on lower-momentum ones
- Address low-momentum tracks near the vertex: Got more of these, needs more work.
  See Thomas Campbell's talk
- Vertex finding measurement uncertainties To Do
- Event display updates – vector hits, pickability, truth labels, etc. VH's done, ECAL added, MC Particles listed now to terminal. Eve-based event display also has many improvements.
- Extrapolate to ECAL Preliminary method written – more work to do. Eldwan is working on this too.
- Try with more complex events (pileup with particles coming from ECAL) Done
- Look at performance of hit clustering algorithm – efficiency and resolution To Do
- Get the CRY and Radiological generators working CRY Works
- Add realistic electronics response and noise. Met with experts. Problem better understood.
- Explore Machine-Learning Techniques See Thomas Campbell's Talk
- Write technote Done. Needs to be kept up.

A slide from January @ CERN

$$|t| \equiv \left|\left(p_\nu - p_\mu - p_\pi\right)^2\right| \approx \left(\sum_{i=\mu,\pi} E_i - \vec{p_i} \cdot \widehat{p_\nu}\right)^2 - \left|\sum_{i=\mu,\pi} (\vec{p_i})_\perp\right|^2$$

$|t|$ = 0 for true coherent events in perfect detector



No ECAL ν
(Pure physics bkg)

No CCQE suppression via dE/dx

20 ton-days of data

L. Bellantoni/FNAL

Fermilab DUNE