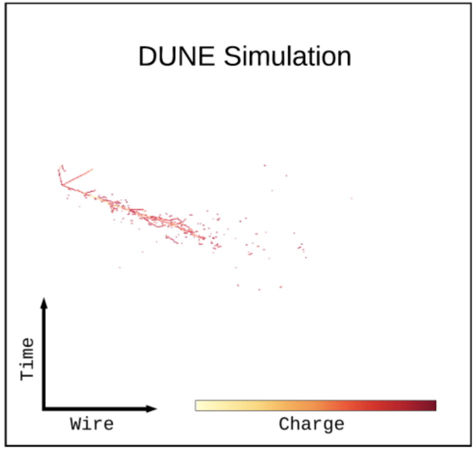
Introduction to LArTPC event reconstruction

Andy Chappell for the Pandora team

06/06/2024 DUNE-France Analysis Workshop

A brief aside on the CVN

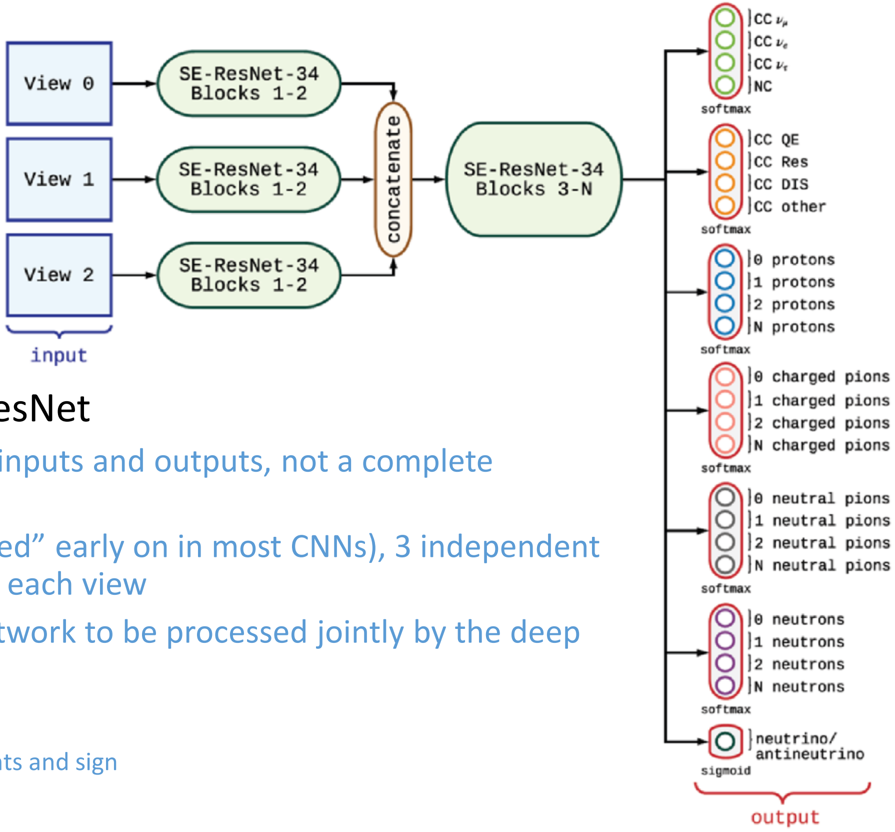
- DUNE has a Convolutional Visual Network for ν_e and ν_μ event selection
  - Uses 500x500 pixel input images of the reconstructed hits (wire number, time, charge) in each of the three readout planes
    - Time is down-sampled to correspond to a pixel size of approximately 5mm (i.e. similar to wire pitch) when accounting for drift velocity
    - For images that don't fit in the 500x500 pixel size, the image is cropped, such that there is an effort to retain the most upstream, relevant wire activity (attempts to veto upstream noise)
    - In time, a window centred on the mean time of the hits is formed and hits are binned in time on this basis
  - No particle reconstruction is applied to the inputs, and none is produced from the outputs



FD HD Induction plane U image of a 2.2 GeV CC ν_e event

DUNE CVN paper

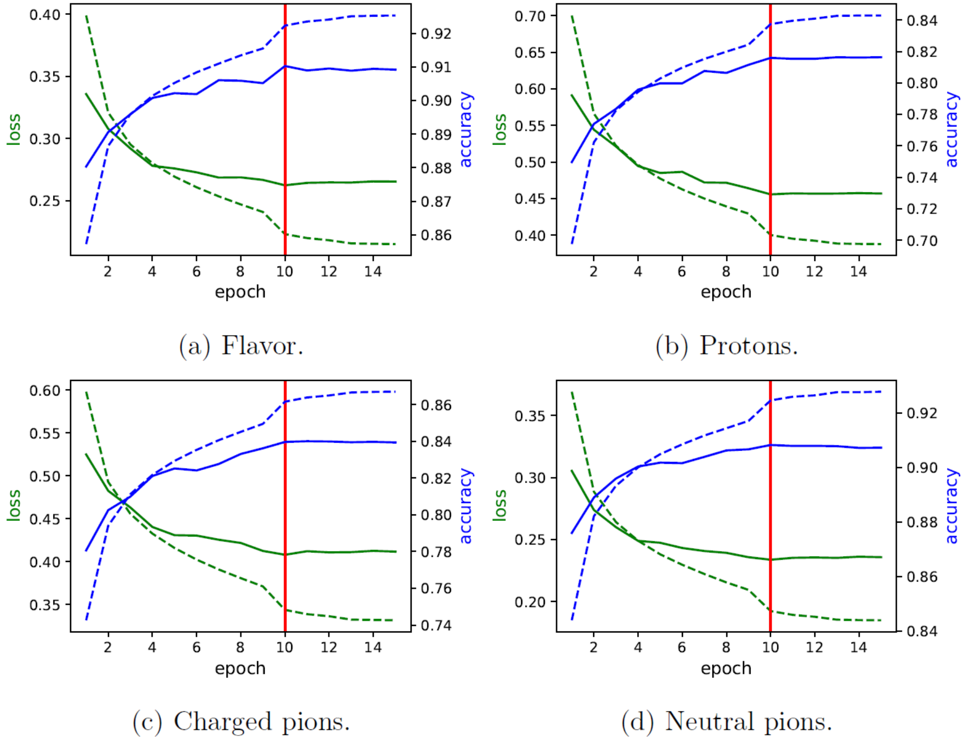
Key CVN architectural choices



- The DUNE CVN is based on a 34-layer SE-ResNet
  - ResNets seek to learn the residuals associating inputs and outputs, not a complete mapping from input to output
  - Rather than 3 RGB channels (which are “averaged” early on in most CNNs), 3 independent UVW branches, to allow parameter learning for each view
  - The 3 branches are concatenated late in the network to be processed jointly by the deep layers
  - 7 outputs for the CVN (Accessing in dunereco)
    - Flavor, interaction type, proton, neutron and pion counts and sign
    - Softmax probabilities for each category

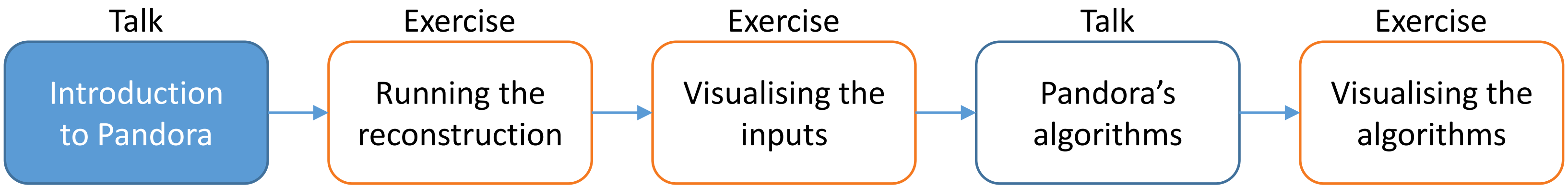
CVN performance

- The CVN has been retrained on newer productions since the paper, but this is an indicative guide to performance on 4 of the tasks
- Interaction type, neutron counts, and sign determination are not shown due to the respective challenges of model dependence, visibility and weak dependence on observables



Network performance in training (dashed) and validation (solid)

Reconstruction session

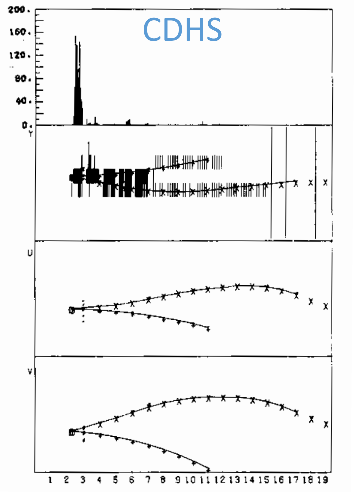


Credit: These slides are based on previous LArSoft workshop slides by John Marshall

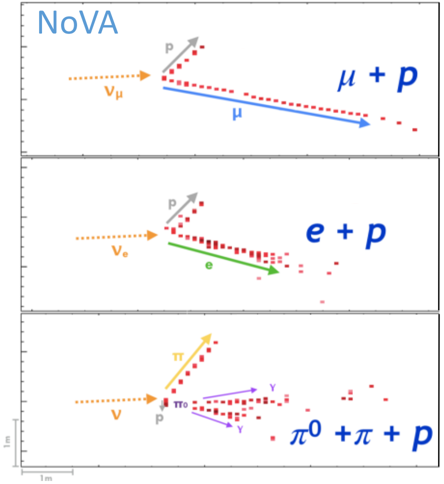
Key references: [Pandora ProtoDUNE paper]
[Pandora MicroBooNE paper]

Neutrino detectors

New Frontiers in High-Energy Physics, pp227-261

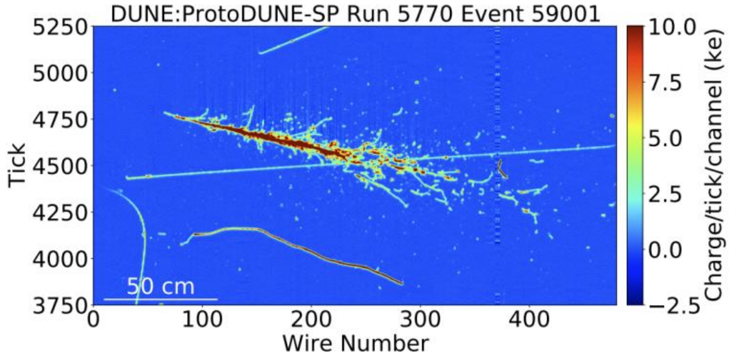


DOI: 10.5281/zenodo.1286758



B. Abi et al 2020 JINST 15 P12004

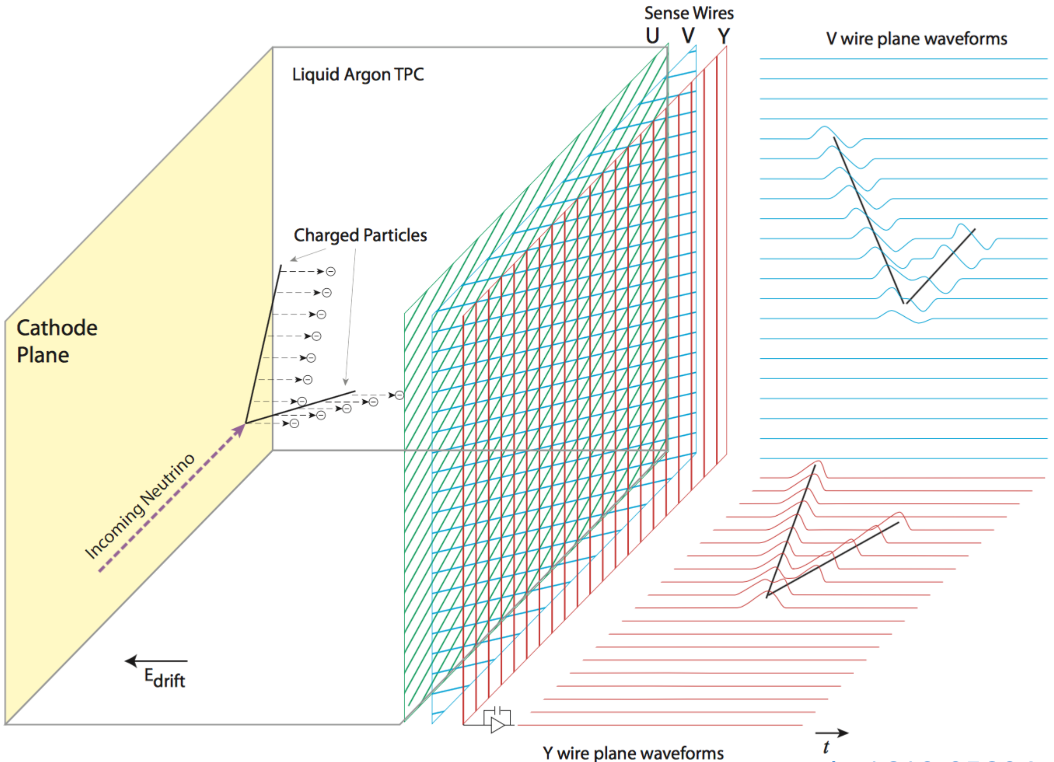
ProtoDUNE-SP



- Evolving detector technologies, with a general trend towards imaging neutrino interactions
  - Emphasis on identifying and characterizing individual visible particles
- Physics sensitivity now depends critically on both hardware and software
  - Need a sophisticated event reconstruction to harness information in the images
- Aim to reconstruct hierarchy of particles of identified types, with measured four-momenta
  - “Particle flow” reconstruction

LArTPC detectors

- Charged particles deposit ionization trails in liquid argon
- Ionization electrons drift in an applied electric field
- Electrons are detected by a series of readout planes (wire planes in this example)
- LArTPC detectors:
  - Past: ICARUS, ArgoNeuT, ProtoDUNE-SP/DP
  - Current: MicroBooNE, LArIAT, ICARUS@SBN
  - Coming soon: SBND, ProtoDUNE-HD, ProtoDUNE-VD

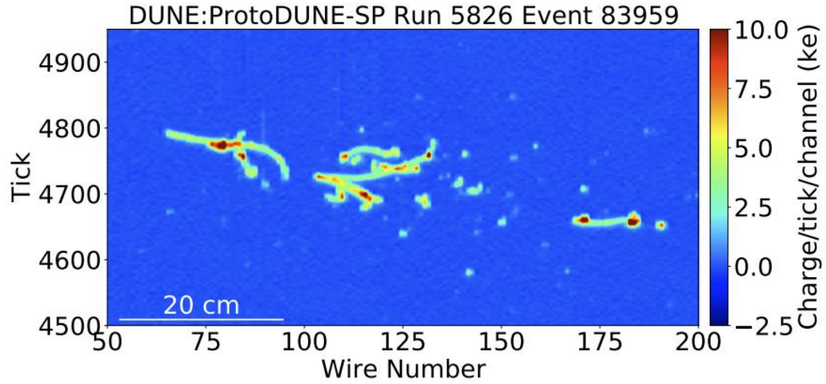


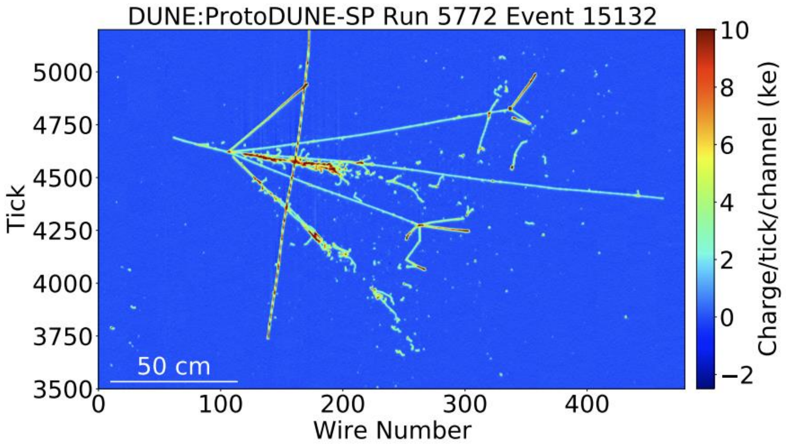
arxiv:1612.05824

LArTPC detectors

Link to LArTPC Animation

LArTPC detectors



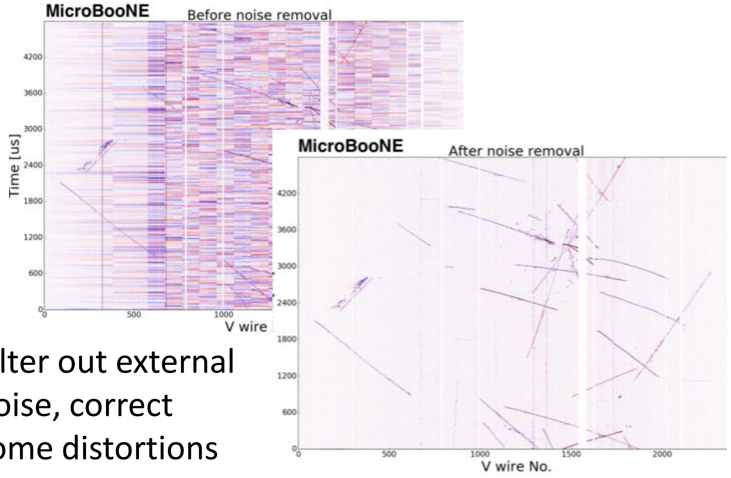


The conversion of raw LArTPC images into analysis-level quantities:

- Low-level steps:
  - Noise filtering
  - Signal processing
- Pattern recognition:
  - The bit you do by eye!
  - Turn images into sparse 2D hits
  - Assign 2D hits to clusters
  - Match features between planes
  - Output a hierarchy of 3D particles
- High-level characterisation:
  - Particle identification
  - Neutrino flavour and interaction type
  - Neutrino energy, etc…

Event reconstruction – low-level (pre-Pandora)

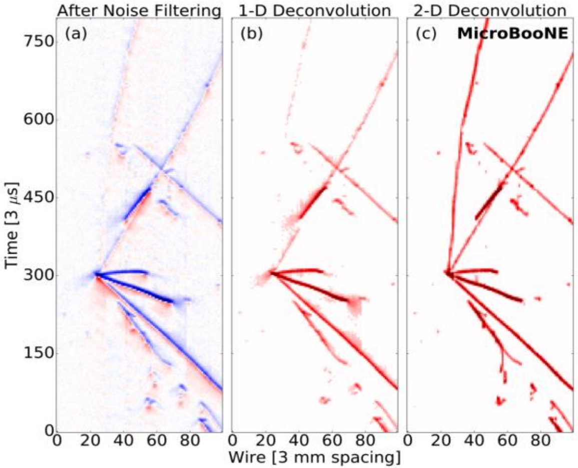
Noise filtering



Filter out external noise, correct some distortions

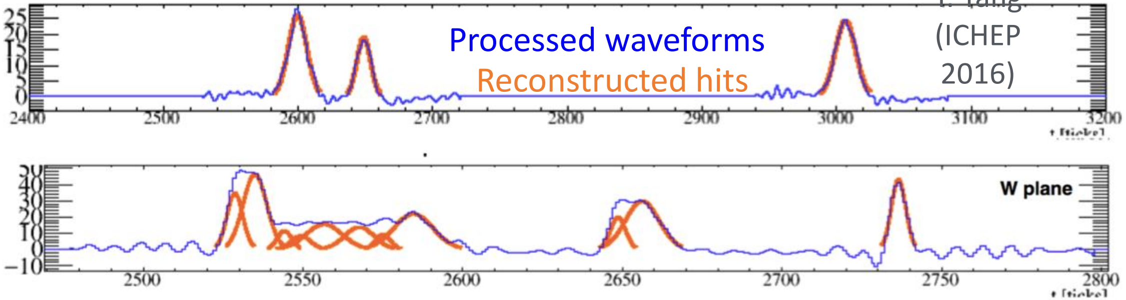
Signal processing

Convert digitized TPC waveform to number of ionization electrons passing through a wire plane at a given time (via deconvolution)



Hit finding

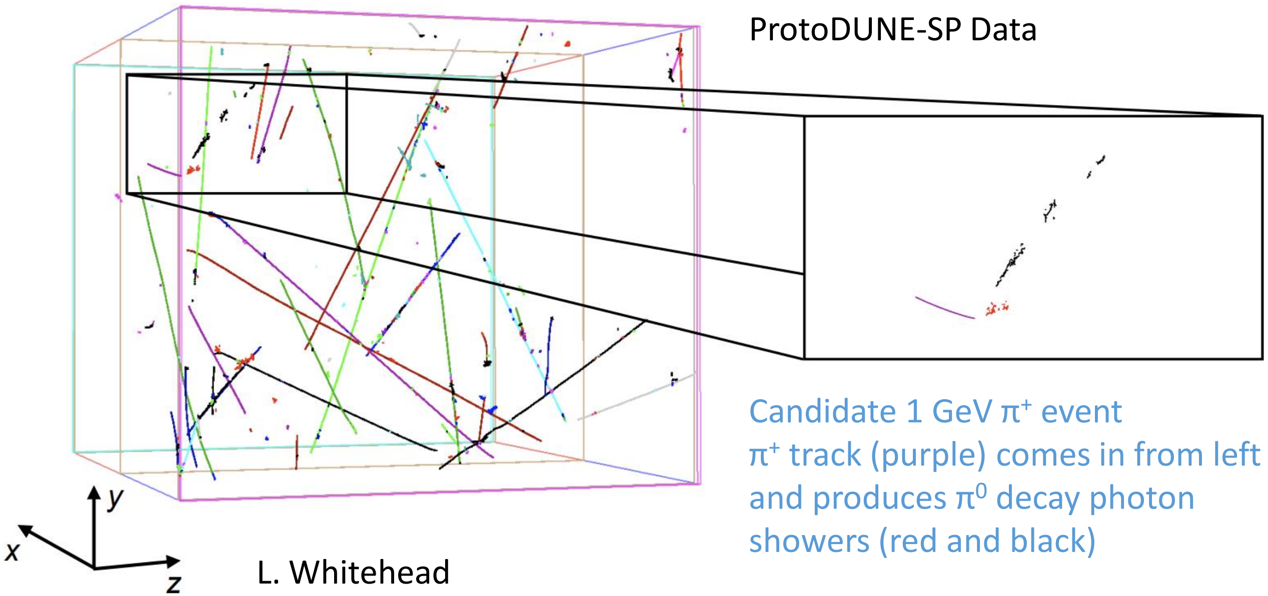
Fit clean waveform with N Gaussians, where N is number of peaks in a pulse. Each Gaussian represents a hit.



T. Yang (ICHEP 2016)

Event reconstruction – pattern recognition (Pandora)

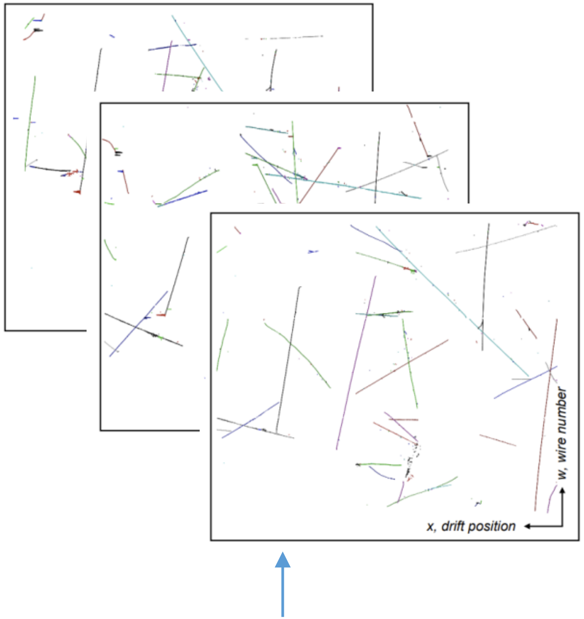
- Main aims of the pattern recognition step are to:
  - Produce 3D reconstructed particles, based on inputs of 3 x 2D images
  - Reconstruct the hierarchy of particles resulting from an interaction.



ProtoDUNE-SP Data

L. Whitehead

Candidate 1 GeV π^+ event
π^+ track (purple) comes in from left and produces π^0 decay photon showers (red and black)



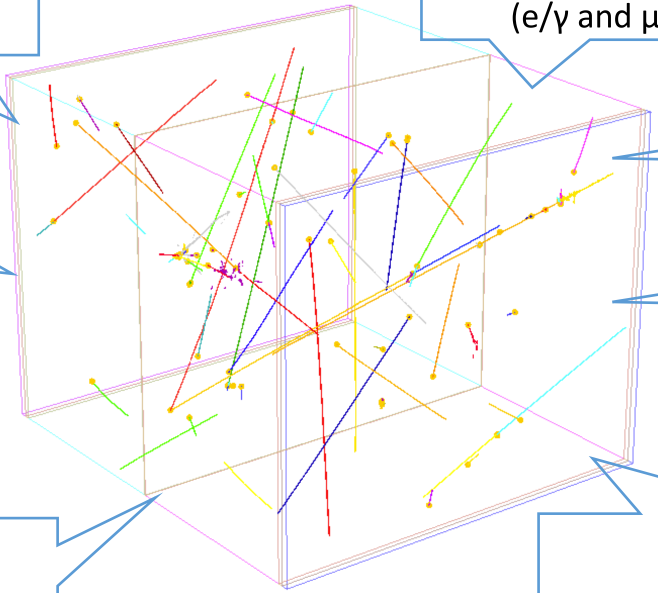
Output from hit finding

Event reconstruction – high-level characterization

Calorimetry
dE/dx from ionization

Particle ID
PID from dE/dx
Start/End for separation
(e/γ and μ/K/p)

Neutrino Interactions
Flavor ID
Background rejection
Energy estimation



π^0 Mass Reconstruction

Michel Electron Studies

ProtoDUNE-SP Simulation

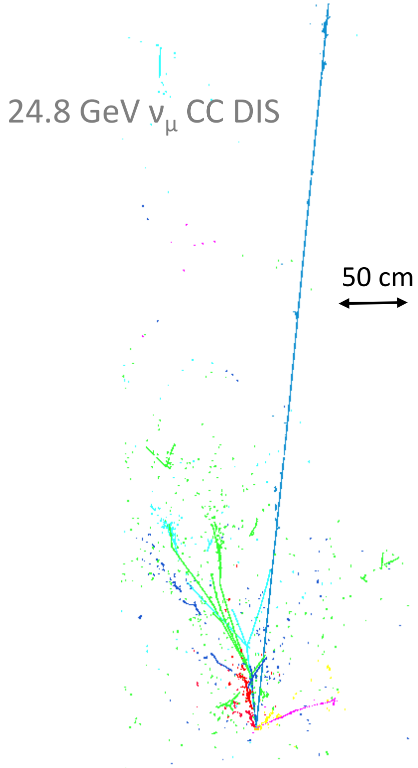
Energy Estimation
Muons
EM showers
Hadronic activity

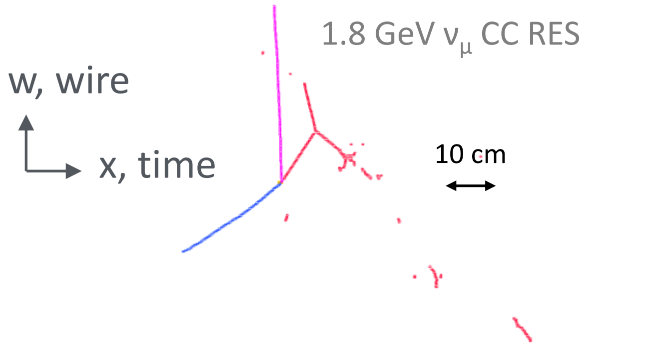
Track Fitting
Trajectory determination
Momentum from multiple Coulomb scattering for exiting tracks

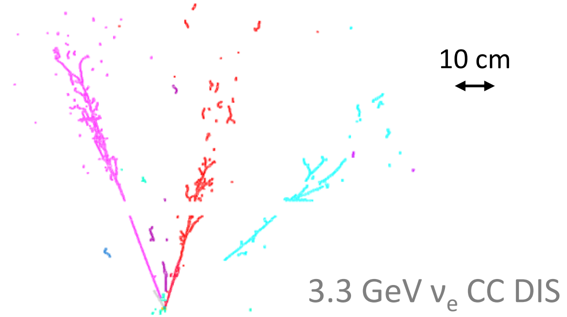
Slide: S. Green

Focus on pattern recognition

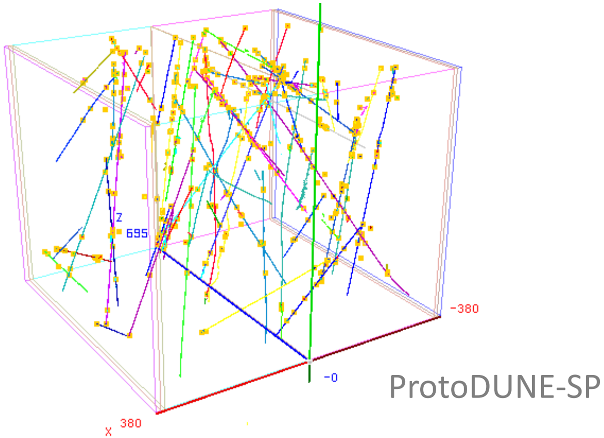
It is a significant challenge to develop automated, algorithmic LArTPC pattern recognition







- Complex, diverse topologies
- Long exposures due to long drift times (up to a few milliseconds)
- Significant cosmic-ray muon background in surface-based detectors



Pandora multi-algorithm approach

- Single clustering approach is unlikely to work for such complex topologies:
    - Mix of track-like and shower-like clusters
- Pandora uses a multi-algorithm approach:
    - Build up events gradually
    - Each step is incremental - aim not to make mistakes (undoing mistakes is hard…)
    - Deploy more sophisticated algorithms as picture of event develops
    - Build physics and detector knowledge into algorithms
    - “Algorithms” can also include machine learning techniques

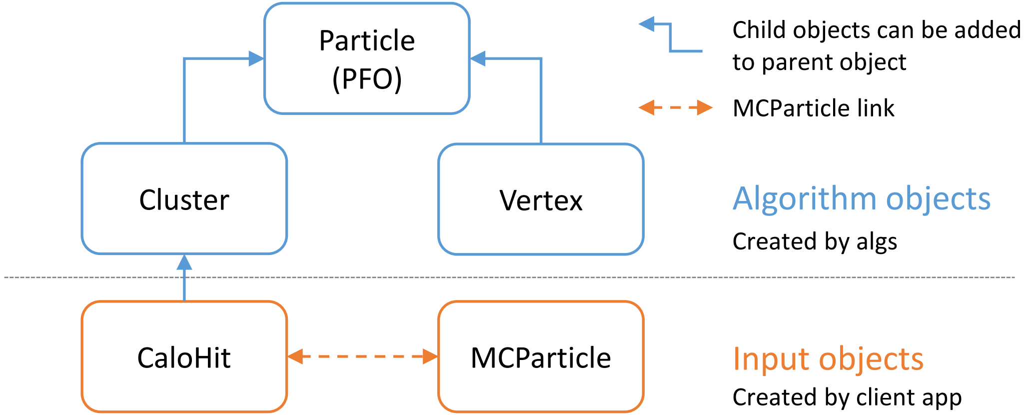
Pandora multi-algorithm approach

- Algorithms contain high-level logic and concentrate on the important bits
  - Physics and pattern recognition ideas
- Pandora software development kit (SDK) supports algorithms
  - Functions to access objects
  - Make new objects
  - Modify existing objects, etc.

Algorithm 1 Cluster creation pseudocode. The logic determining when to create new Clusters and when to extend existing Clusters will vary between algorithms.

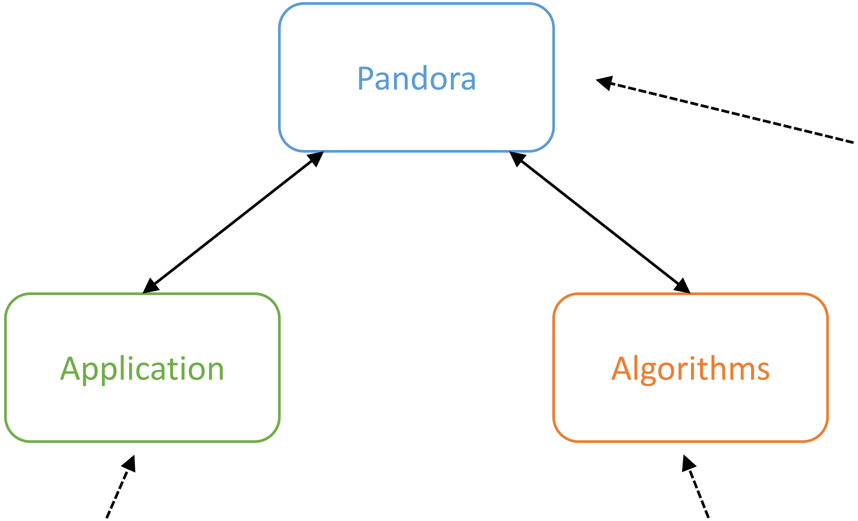
```
procedure CLUSTER CREATION
    Create temporary Cluster list
    Get current CaloHit list
    for all CaloHits do
        if CaloHit available then
            for all newly-created Clusters do
                Find best host Cluster
            if Suitable host Cluster found then
                Add CaloHit to host Cluster
            else
                Add CaloHit to a new Cluster
    Save new Clusters in a named list
```

Example algorithm structure



Event Data Model

Pandora in LArSoft



pandora (framework, visualization)

Re-usable libraries to support multi-alg approach

Hosted on PandoraPFA GitHub
pre-built as external package by LArSoft

larpandora

Producer module, provides translation LArSoft⟷Pandora

Hosted via LArSoft GitHub, built with mrb

larpandoracontent

100+ algorithms and tools that control the pattern recognition

Hosted via LArSoft and PandoraPFA GitHub, can build with mrb

Pandora application: larpandora

- The controlling application should be simple: create Pandora instance(s), register algorithms, provide a Pandora Settings XML file and handle event input/output:
  - In reality, often tends to get complicated (vindicating decision to separate algorithms from steps needed to access and control the inputs and outputs…)

```
/**
 *  @brief  ILArPandora class
 */
class ILArPandora : public art::EDProducer
```

Implement as an art::EDProducer

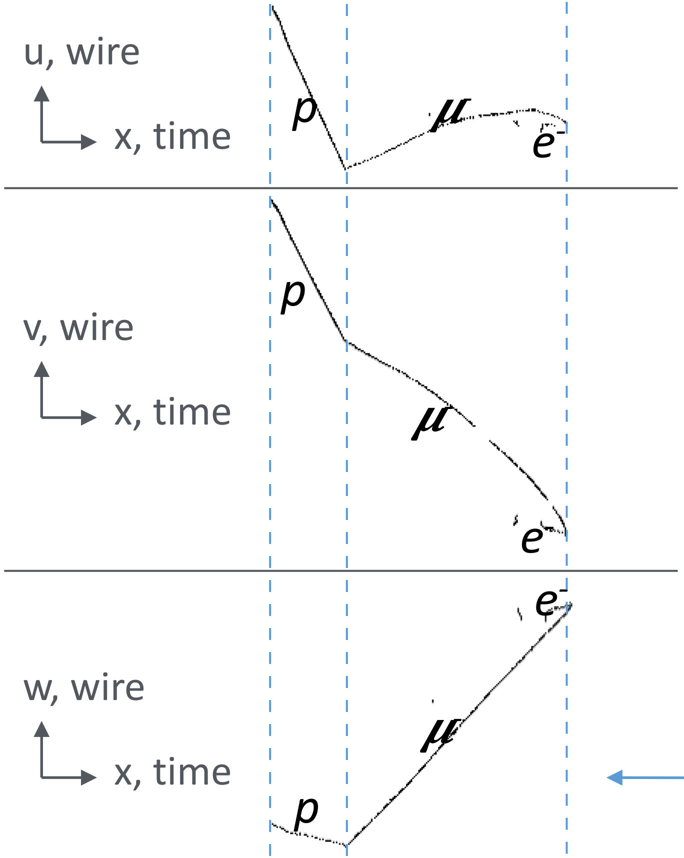
Relevant callbacks

```
void beginJob();
void produce(art::Event &evt);
```

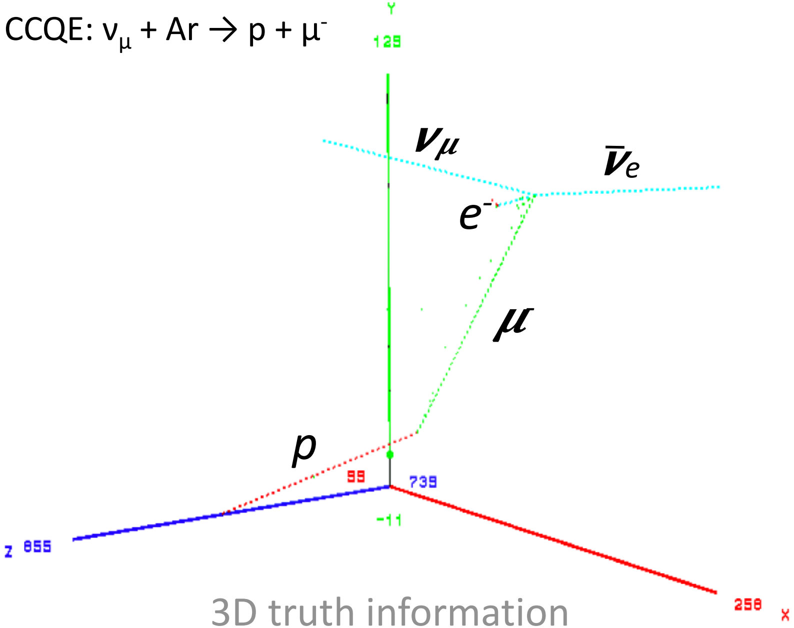
Algorithm Pseudocode description of a client application for LAr TPC event reconstruction in a single drift volume

```
procedure MAIN
    Create a Pandora instance
    Register Algorithms and Plugins
    Ask Pandora to parse XML settings file
    for all Events do
        Create CaloHit instances
        Create MCParticle instances
        Specify MCParticle-CaloHit relationships
        Ask Pandora to process the event
        Get output PFOs and write to file
        Reset Pandora before next event
```

Inputs to Pandora



CCQE: ν_{μ} + Ar → p + μ^{-}



3D truth information

← Three 2D representations with common x coordinate, derived from drift time

LArSoft workflows

- In this session we'll be working from pre-prepared reco1 files
  - You'll see a little more detail on workflows in the next exercise
  - Reco1 files have gone through μ+p particle gun, G4 propagation, detector simulation and (the actual reco1 part) hit finding/disambiguation
- Where does Pandora run?
  - Pandora typically runs as part of the reco2 workflow step
  - It is followed by various high-level reconstruction steps that use its outputs
  - Typically, these steps will produce reco2 output files on which analyzers will run
  - We won't do that here, we're just going to look at how Pandora operates

We will now try running the larpandora module in LArSoft