Automated Reconstruction, Signal Processing and Particle Identification in DUNE

Tingjun Yang[*][†]

Fermilab

E-mail: `tjyang@fnal.gov`

Liquid Argon TPC (LArTPC) technology is increasingly prevalent in large-scale detectors designed to observe neutrino scattering events induced by accelerators or by natural sources. LArTPCs consist of a very high fraction of active detector material with spatial resolutions on the order of a few millimeters. Three-dimensional interactions are imaged in multiple two-dimensional views by the process of projection onto planes of wires. The goal of automated reconstruction is to correctly classify each neutrino scattering event by the flavor of the incoming neutrino, to separate charged-current events from neutral-current and other backgrounds, and to measure the energies of the incoming neutrinos. Detection of neutrinos from supernova bursts and also searching for nucleon decay are important uses of automated reconstruction algorithms. Because the amount of spatial detail is high and the amount of scattering is high, this reconstruction presents many challenges that are now being investigated with sophisticated techniques. Signal processing, track and shower identification, particle identification, and event classification by a variety of innovative algorithms are reviewed.

[*]Speaker.

[†]On behalf of the DUNE Collaboration.

1. Introduction

Liquid argon time projection chambers (LArTPCs) provide a robust and elegant method for measuring the properties of neutrino interactions above a few tens of MeV by providing 3D event imaging with excellent spatial and energy resolution. Reconstruction of events in a LArTPC is challenging. A lot of effort has been devoted in this area over the past years. The adoption of the common framework LArSoft [1] by several LArTPC experiments helps people exchange ideas and test algorithms on real data. This paper reviews the latest reconstruction chain used by the Deep Underground Neutrino Experiment (DUNE).

2. Signal Processing

The raw data are in the format of ADC counts as a function of TPC ticks on each wire. The signal can have a unipolar shape if it is on a collection wire or a bipolar shape if it is on an induction wire. The first step in the reconstruction is to convert the raw signal from each wire to a standard shape such as a Gaussian shape. This is achieved by passing the raw data through a calibrated deconvolution algorithm. In real detectors, excess noises may exist and need to be properly dealt with through a dedicated noise filter [2, 3]. There is currently no such complication in the simulation.

The goal of the deconvolution is to "remove" the impact of field and electronics responses from the measured signal to recover the number of drifting ionization electrons. This technique has the advantages of being robust and fast and is an essential step in the overall drifted-charge profiling process.

3. Gaussian Hit Finder

GausHitFinder is a hit-finding algorithm that works starting from deconvolved signals on wires and defining areas above threshold known as "pulses". Once a pulse is found, a sum of *n* Gaussians is fit to the waveform where *n* is defined by the number of peaks initially identified within the pulse. Each Gaussian is defined as a hit.

4. Disambiguation

The induction wires are wrapped in the DUNE TPC design in order to put electronics on the ends of the anode plane assemblies (APAs) and minimize the size of the dead region between APAs. The consequence is that multiple induction wire segments will be read out by the same electronic channel. We need to determine which wire segment corresponds to the energy deposited by the particle in the TPC. This process is called disambiguation.

The current disambiguation algorithm was developed for the DUNE 35t prototype. It relies on the fact that the collection wires are not wrapped and the wire angles are slightly different for the two induction planes (44.3° vs 45.7°) so that any three wires from the three planes read out by the same three channels will never cross more than once. The disambiguation algorithm first loops over all collection hits. For each collection hit, it loops over all induction hits and looks for one hit on each of the two induction planes that are in time with the collection hit. Once the triplet of

hits are found, the algorithm checks all possible wire segments and looks for three wire segments that intersect. Once one and only one intersection is identified, the two induction wire segments are assigned to the two induction hits and the ambiguity is resolved. The same disambiguation algorithm also works for the far detector and protoDUNE.

5. Line Cluster

A cluster is a collection of hits on one wire plane that represents a single particle. The intent of the Line Cluster algorithm is to construct two-dimensional line-like clusters using local information. The algorithm was originally known as Cluster Crawler. The concept is to construct a short line-like "seed" cluster of proximate hits in an area of low hit density where hit proximity is a good indication that the hits are indeed associated with each other. Additional nearby hits are attached to the leading edge of the cluster if they are similar to the hits already attached to it. The conditions are that the impact parameter between a prospective hit and the cluster projection is similar to those previously added and the hit charge is similar as well. These conditions are moderated to include high charge hits that are produced by large dE/dx fluctuations and the rapid increase in dE/dx at the end of stopping tracks while rejecting large charge hits from δ-rays. Seed clusters are formed at one end of the hit collection so that crawling in only one direction is sufficient. Line Cluster uses hits as input and produces a new set of refined hits. More details on the Line Cluster algorithm can be found in Ref [4].

6. Blurred Cluster

The Blurred Cluster reconstruction method aims to construct two-dimensional shower-like clusters from deposits left in the detector by showers. It specialises in shower reconstruction, especially in the separation of nearby showers in the reconstruction of, e.g., π^0 decay. The algorithm first applies a weighted 2D Gaussian smearing to the hit map in order to introduce "fake hits" and distribute the charge deposited in the detector more realistically. This proceeds by convolution of a Gaussian kernel, uniquely applied for each hit given information such as rough directionality of the showering particles and the width of the reconstructed hits in time in order to introduce the most accurate blurring possible. Clustering follows by grouping neighbouring hits within the blurred region before removing any artificial hits and forming output clusters from the remaining hit collections. BlurredCluster uses hits as input and the output clusters are in turn used as input to the EMShower algorithm (see Sec. 9). More details and detailed discussion are available in Ref [5].

7. Pandora

The Pandora Software Development Kit [6] was created to address the problem of identifying energy deposits from individual particles in fine-grained detectors. It promotes the idea of a multi-algorithm approach to solving pattern-recognition problems. In this approach, the input hits are considered by large numbers of decoupled pattern-recognition algorithms. Each algorithm targets a specific event topology and controls operations such as collecting Hits together in Clusters, merging or splitting Clusters, or collecting Clusters in order to build Particles. The output from the chain

of over 70 algorithms is a hierarchy of reconstructed 3D Particles, each with an identified particle type, vertex and direction.

8. Projection Matching Algorithm

The Projection Matching Algorithm (PMA) was primarily developed as a technique of 3D reconstruction of individual particle trajectories (trajectory fit) Ref [7]. PMA was designed to address a challenging issue of transformation from a set of independently reconstructed 2D projections of objects into a 3D representation. Reconstructed 3D objects are providing also a basic physics quantities like particle directions and dE/dx evolution along the trajectories. PMA uses as its input the output from 2D pattern recognition: clusters of hits. The algorithm attempts also to correct hit to cluster assignments using properties of 3D reconstructed objects.

The PMA underlying idea is to build and optimize objects in 3D space (formed as polygonal lines with iteratively increased number of segments) by minimizing the cost function calculated simultaneously in all available 2D projections. The cost function consists of 2D distance of hits to the optimized object 2D projections, penalty of tracks curvature, and 3D distance of various feature points to the optimized object.

Recent developments of PMA are relying on the same principle ideas and allow to build and optimize complex structures of 3D objects, i.e. multiple particle trajectories interconnected with interaction vertices. With such approach it is possible to employ in the vertex position reconstruction the local information from several tracks simultaneously, leading also to an improved fit of each individual trajectory.

9. EMShower

The EMShower reconstruction algorithm aims to find final 3D showers and all associated properties. It is intentionally high-level by design and relies heavily on previous reconstruction, specifically BlurredCluster (Sec. 6) and PMA (Sec. 8). The reconstruction proceeds in two general steps: first, the shower objects, including all associated hits in each of the views, are found; second, the properties of these showers, such as start point, direction, energy and initial dE/dx, are determined by multiple pattern recognition and calorimetric reconstruction algorithms.

The shower objects are created by simply matching the previously found, well-formed, shower-like clusters (provided by BlurredCluster) between the different views to form 3D objects with associated hits in each plane. This is achieved by associating hits between the 2D shower-like clusters and 3D tracks (provided by, e.g., PMA) in order to pull together clusters from different planes into one object. These shower hits are then analysed by various successive algorithms in order to find relevant properties before the output shower objects are constructed for later use.

Further details and detailed discussion can be found in Ref [8].

10. Calorimetric Energy Reconstruction and Particle Identification

As charged particles traverse a liquid argon volume, they deposit energy through ionization and scintillation. It is important to measure the energy deposition as it provides information on

particle energy and species. The algorithm for reconstructing the ionization energy in LArSoft is optimized for line-like tracks and is being extended to more complicated event topology such as showers. The algorithm takes all the hits associated with a reconstructed track. For each hit, the hit area or amplitude, in ADC counts, is converted to the charge in fC units, on the wire using an ADC to fC conversion factor that was determined by muons or teststand measurements. Two corrections are made to convert dQ/dx to dE/dx: lifetime correction and recombination correction based on the modified Box's model [9] or the Birks's model[10].

If the incident particle stops in the LArTPC active volume, the energy loss, dE/dx, as a function of the residual range (R), the path length to the end point of the track, is used as a powerful method for particle identification. One example calculates the quantity PIDA = $\langle A_i \rangle = \langle (dE/dx)_i R_i^{0.42} \rangle$ [9], which is defined to be the average of $A_i = (dE/dx)_i R_i^{0.42}$ over all track points where the residual range R_i is less than 30 cm. The particle species can be determined by making a selection on the PIDA value.

11. WireCell

WireCell [11], which adopts a very different approach from the other algorithms mentioned here, is a new reconstruction method under development. Instead of directly doing pattern recognition on each of the 2D views (drift time vs. wire number), the first step of the WireCell reconstruction is to perform 3D imaging with time, geometry, and charge information. The definition of "Hit" is based on signal strength after charge extraction in a 2 μs time slice. The usage of time information means that hits from different wire planes at different time slices cannot be associated. The usage of geometry information means that hits from wires that are not crossing each other cannot be associated together. The usage of charge information means that the hits from different wire plane with different signal strength are unlikely to be associated together. The usage of the charge information is quite unique for LArTPCs, as each of the wire plane in principle detects the same amount of ionization electrons. The advantage of the WireCell approach is that it utilizes the full TPC information. The strong requirement of the time/geometry/charge information provides a natural way to suppress electronic noise at the cost of being more sensitive to the hit inefficiency. Since the track and shower hypotheses are not used, the 3D imaging works for any event topology. Once the 3D images are reconstructed, 3D pattern recognition is needed to identify the content inside the image.

12. Optical Reconstruction

The first step of the DUNE optical reconstruction is reading individual waveforms from the simulated photon detector electronics and finding optical hits – regions of the waveforms containing pulses. The optical hit contains the optical channel that the hit was found on, time corresponding to the hit, its width, area, amplitude, and number of photoelectrons.

The current DUNE optical-hit-finder algorithm searches for regions of the waveform exceeding a certain threshold over pedestal, checking whether that region is wider than a preset time width corresponding to a typical optical pulse width, and, if it is, calculating the optical-hit parameters for the region (including parts of the waveform around it that have ADC values greater than 1) and

recording it as an optical hit. The number of photoelectrons is calculated by dividing the area of the hit by the area of a single-photoelectron pulse.

After optical hits are reconstructed, they are grouped into higher-level objects called optical flashes. The optical flash contains the time and time width of the flash, its approximate coordinates in the plane perpendicular to the electric field (and spatial widths along those axes), its location and size in the wire planes, the distribution of photoelectrons across all photon detectors, and the total number of photoelectrons in the flash, among other parameters.

The flash-finding algorithm searches for an increase in photon-detector activity (the number of photoelectrons) in time using information from optical hits on all photon detectors. When a collection of hits with the total number of photoelectrons greater than or equal to 2 is found, the algorithm begins creating an optical flash. It starts with the largest hit and adds hits from the found hit collection that lie closer than one half of the sum of the widths of the flash under construction and the hit being added to it. The flash is stored after no more hits can be added to it and if it has more than 2 photoelectrons.

The algorithm also estimates spatial parameters of the optical flash by calculating the photoelectron-weighted mean and root mean square of the locations of the contributing optical hits (defined as centers of photon detectors where those hits were detected) contained in the flash.

13. Conclusions

The reconstruction of events in LArTPC is being actively developed and encouraging progress has been made. A fully automated reconstruction chain is in place in DUNE to reconstruct tracks, showers, calorimetric and optical information for various physics analyses.

References

[1] http://larsoft.org/

[2] "Noise Characterization and Filtering in the MicroBooNE TPC", MicroBooNE NOTE 1016-PUB.

[3] "A Method to Extract the Charge Distribution Arriving at the TPC Wire Planes in MicroBooNE", MicroBooNE NOTE 1017-PUB.

[4] https://cdcvs.fnal.gov/redmine/documents/727

[5] http://larsoft.org/single-record/?pdb=110

[6] J. S. Marshall and M. A. Thomson, Eur. Phys. J. C **75**, no. 9, 439 (2015) doi:10.1140/epjc/s10052-015-3659-3 [arXiv:1506.05348 [physics.data-an]].

[7] M. Antonello *et al.*, Adv. High Energy Phys. **2013**, 260820 (2013) doi:10.1155/2013/260820 [arXiv:1210.5089 [physics.ins-det]].

[8] http://larsoft.org/single-record/?pdb=113

[9] ArgoNeuT Collaboration, "A study of electron recombination using highly ionizing particles in the ArgoNeuT Liquid Argon TPC," JINST **8**, P08005 (2013).

[10] ICARUS Collaboration, "Study of electron recombination in liquid argon with the ICARUS TPC", Nucl. Instrum. Meth. Phys. Res. A 523 (2004), 275.

[11] http://www.phy.bnl.gov/wire-cell/