

PAPER

Wire-cell 3D pattern recognition techniques for neutrino event reconstruction in large LArTPCs: algorithm description and quantitative evaluation with MicroBooNE simulation

To cite this article: MicroBooNE Collaboration *et al* 2022 *JINST* **17** P01037

View the [article online](#) for updates and enhancements.

You may also like

- [A muon-track reconstruction exploiting stochastic losses for large-scale Cherenkov detectors](#)
on behalf of the IceCube collaboration, R. Abbasi, M. Ackermann et al.
- [Comparative evaluation of analogue front-end designs for the CMS Inner Tracker at the High Luminosity LHC](#)
The Tracker Group of the CMS Collaboration, W. Adam, T. Bergauer et al.
- [Measurement of the atmospheric muon rate with the MicroBooNE Liquid Argon TPC](#)
MicroBooNE, P. Abratenko, M. Alrashed et al.



IOP | ebooks™

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

Wire-cell 3D pattern recognition techniques for neutrino event reconstruction in large LArTPCs: algorithm description and quantitative evaluation with MicroBooNE simulation

MicroBooNE Collaboration

P. Abratenko,^{gg} R. An,ⁿ J. Anthony,^d L. Arellano,^r J. Asaadi,^{ff} A. Ashkenazi,^{dd} S. Balasubramanian,^k B. Baller,^k C. Barnes,^t G. Barr,^w V. Basque,^r L. Bathe-Peters,^m O. Benevides Rodrigues,^{cc} S. Berkman,^k A. Bhandari,^r A. Bhat,^{cc} M. Bishai,^b A. Blake,^p T. Bolton,^o J.Y. Book,^m L. Camilleri,ⁱ D. Caratelli,^k I. Caro Terrazas,^h R. Castillo Fernandez,^k F. Cavanna,^k G. Cerati,^k Y. Chen,^a D. Cianci,ⁱ J.M. Conrad,^s M. Convery,^z L. Cooper-Troendle,^{jj} J.I. Crespo-Anadón,^e M. Del Tutto,^k S.R. Dennis,^d P. Detje,^d A. Devitt,^p R. Diurba,^u R. Dorrill,ⁿ K. Duffy,^k S. Dytman,^x B. Eberly,^{bb} A. Ereditato,^a J.J. Evans,^r R. Fine,^q G.A. Fiorentini Aguirre,^{aa} R.S. Fitzpatrick,^t B.T. Fleming,^{jj} N. Foppiani,^m D. Franco,^{jj} A.P. Furmanski,^u D. Garcia-Gamez,^l S. Gardiner,^k G. Ge,ⁱ S. Gollapinni,^{ee,q} O. Goodwin,^r E. Gramellini,^k P. Green,^r H. Greenlee,^k W. Gu,^b R. Guenette,^m P. Guzowski,^r L. Hagaman,^{jj} O. Hen,^s C. Hilgenberg,^u G.A. Horton-Smith,^o A. Hourlier,^s R. Itay,^z C. James,^k X. Ji,^{b,*} L. Jiang,^{hh} J.H. Jo,^{jj} R.A. Johnson,^g Y.-J. Jwa,ⁱ D. Kalra,ⁱ N. Kamp,^s N. Kaneshige,^c G. Karagiorgi,ⁱ W. Ketchum,^k M. Kirby,^k T. Kobilarcik,^k I. Kreslo,^a R. LaZur,^h I. Lepetic,^y K. Li,^{jj} Y. Li,^b K. Lin,^q B.R. Littlejohn,ⁿ W.C. Louis,^q X. Luo,^c K. Manivannan,^{cc} C. Mariani,^{hh} D. Marsden,^r J. Marshall,ⁱⁱ D.A. Martinez Caicedo,^{aa} K. Mason,^{gg} A. Mastbaum,^y N. McConkey,^r V. Meddage,^o T. Mettler,^a K. Miller,^f J. Mills,^{gg} K. Mistry,^r A. Mogan,^{ee} T. Mohayai,^k J. Moon,^s M. Mooney,^h A.F. Moor,^d C.D. Moore,^k L. Mora Lepin,^r J. Mousseau,^t M. Murphy,^{hh} D. Naples,^x A. Navrer-Agasson,^r M. Nebot-Guinot,^j R.K. Neely,^o D.A. Newmark,^q J. Nowak,^p M. Nunes,^{cc} O. Palamara,^k V. Paolone,^x A. Papadopoulou,^s V. Papavassiliou,^v S.F. Pate,^v N. Patel,^p A. Paudel,^o Z. Pavlovic,^k E. Piasetzky,^{dd} I.D. Ponce-Pinto,^{jj} S. Prince,^m X. Qian,^b J.L. Raaf,^k V. Radeka,^b A. Rafique,^o M. Reggiani-Guzzo,^r L. Ren,^v L.C.J. Rice,^x L. Rochester,^z J. Rodriguez Rondon,^{aa} M. Rosenberg,^x M. Ross-Lonergan,ⁱ G. Scanavini,^{jj} D.W. Schmitz,^f A. Schukraft,^k W. Seligman,ⁱ M.H. Shaevitz,ⁱ R. Sharankova,^{gg} J. Shi,^d J. Sinclair,^a A. Smith,^d E.L. Snider,^k M. Soderberg,^{cc} S. Söldner-Rembold,^r P. Spentzouris,^k J. Spitz,^t M. Stancari,^k J. St. John,^k T. Strauss,^k K. Sutton,ⁱ S. Sword-Fehlberg,^v A.M. Szelc,^j N. Tagg,^w W. Tang,^{ee} K. Terao,^z C. Thorpe,^p D. Totani,^c M. Touns,^k Y.-T. Tsai,^z M.A. Uchida,^d T. Usher,^z W. Van De Pontseele,^{w,m} B. Viren,^b M. Weber,^a H. Wei,^b Z. Williams,^{ff}

S. Wolbers,^k T. Wongjirad,^{gg} M. Wospakrik,^k K. Wresilo,^d N. Wright,^s W. Wu,^k E. Yandel,^c T. Yang,^k G. Yarbrough,^{ee} L.E. Yates,^s H.W. Yu,^{b,*} G.P. Zeller,^k J. Zennaro^k and C. Zhang^b

^aUniversität Bern, Bern CH-3012, Switzerland

^bBrookhaven National Laboratory (BNL), Upton, NY, 11973, U.S.A.

^cUniversity of California, Santa Barbara, CA, 93106, U.S.A.

^dUniversity of Cambridge, Cambridge CB3 0HE, United Kingdom

^eCentro de Investigaciones Energéticas, Medioambientales y Tecnológicas (CIEMAT), Madrid E-28040, Spain

^fUniversity of Chicago, Chicago, IL, 60637, U.S.A.

^gUniversity of Cincinnati, Cincinnati, OH, 45221, U.S.A.

^hColorado State University, Fort Collins, CO, 80523, U.S.A.

ⁱColumbia University, New York, NY, 10027, U.S.A.

^jUniversity of Edinburgh, Edinburgh EH9 3FD, United Kingdom

^kFermi National Accelerator Laboratory (FNAL), Batavia, IL 60510, U.S.A.

^lUniversidad de Granada, E-18071, Granada, Spain

^mHarvard University, Cambridge, MA 02138, U.S.A.

ⁿIllinois Institute of Technology (IIT), Chicago, IL 60616, U.S.A.

^oKansas State University (KSU), Manhattan, KS, 66506, U.S.A.

^pLancaster University, Lancaster LA1 4YW, United Kingdom

^qLos Alamos National Laboratory (LANL), Los Alamos, NM, 87545, U.S.A.

^rThe University of Manchester, Manchester M13 9PL, United Kingdom

^sMassachusetts Institute of Technology (MIT), Cambridge, MA, 02139, U.S.A.

^tUniversity of Michigan, Ann Arbor, MI, 48109, U.S.A.

^uUniversity of Minnesota, Minneapolis, Mn, 55455, U.S.A.

^vNew Mexico State University (NMSU), Las Cruces, NM, 88003, U.S.A.

^wUniversity of Oxford, Oxford OX1 3RH, United Kingdom

^xUniversity of Pittsburgh, Pittsburgh, PA, 15260, U.S.A.

^yRutgers University, Piscataway, NJ, 08854, U.S.A., PA

^zSLAC National Accelerator Laboratory, Menlo Park, CA, 94025, U.S.A.

^{aa}South Dakota School of Mines and Technology (SDSMT), Rapid City, SD, 57701, U.S.A.

^{bb}University of Southern Maine, Portland, ME, 04104, U.S.A.

^{cc}Syracuse University, Syracuse, NY, 13244, U.S.A.

^{dd}Tel Aviv University, Tel Aviv, Israel, 69978

^{ee}University of Tennessee, Knoxville, TN, 37996, U.S.A.

^{ff}University of Texas, Arlington, TX, 76019, U.S.A.

^{gg}Tufts University, Medford, MA, 02155, U.S.A.

^{hh}Center for Neutrino Physics, Virginia Tech, Blacksburg, VA, 24061, U.S.A.

ⁱⁱUniversity of Warwick, Coventry CV4 7AL, United Kingdom

^{jj}Wright Laboratory, Department of Physics, Yale University, New Haven, CT, 06520, U.S.A.

*Corresponding author.

E-mail: microboone_info@fnal.gov

ABSTRACT: Wire-Cell is a 3D event reconstruction package for liquid argon time projection chambers. Through geometry, time, and drifted charge from multiple readout wire planes, 3D space points with associated charge are reconstructed prior to the pattern recognition stage. Pattern recognition techniques, including track trajectory and dQ/dx (ionization charge per unit length) fitting, 3D neutrino vertex fitting, track and shower separation, particle-level clustering, and particle identification are then applied on these 3D space points as well as the original 2D projection measurements. A deep neural network is developed to enhance the reconstruction of the neutrino interaction vertex. Compared to traditional algorithms, the deep neural network boosts the vertex efficiency by a relative 30% for charged-current ν_e interactions. This pattern recognition achieves 80–90% reconstruction efficiencies for primary leptons, after a 65.8% (72.9%) vertex efficiency for charged-current ν_e (ν_μ) interactions. Based on the resulting reconstructed particles and their kinematics, we also achieve 15-20% energy reconstruction resolutions for charged-current neutrino interactions.

KEYWORDS: Pattern recognition, cluster finding, calibration and fitting methods; Analysis and statistical methods

ARXIV EPRINT: [2110.13961](https://arxiv.org/abs/2110.13961)

Contents

1	Introduction	1
2	Initial iteration towards neutrino vertex identification	7
2.1	Determination of vertices and track segments	7
2.2	3D vertex fitting	9
2.3	Separation of track and shower topology	10
2.4	Neutrino vertex identification with traditional techniques	12
2.5	Neutrino vertex identification with a regressional segmentation network	15
2.6	Evaluation of the neutrino vertex identification performance	20
3	Particle flow tree reconstruction after neutrino vertex identification	23
3.1	Overall electromagnetic shower clustering	23
3.2	Neutral pion reconstruction	24
3.3	Reconstruction performance of primary leptons	25
4	Neutrino energy reconstruction	27
5	Summary	30

1 Introduction

The liquid argon time projection chamber (LArTPC) [1–4] is a three-dimensional tracking calorimeter widely used in neutrino physics [5–9]. Compared to water Cerenkov and liquid scintillator technologies, LArTPCs are highly effective in differentiating electrons and photons in neutrino interactions through identifying the gap between photon conversion vertex and neutrino interaction vertex and measuring the dE/dx (ionization energy loss per unit length) in the first few centimeters of electron or photon electromagnetic cascade (EM shower). This capability is advantageous for detecting ν_e charged-current interactions with high signal efficiencies and low background contamination. Utilizing this unique capability of LArTPCs, the MicroBooNE experiment [7] aims to understand the nature of the low-energy excess of ν_e -like events observed in the MiniBooNE experiment [10] and to measure neutrino-argon interaction cross sections in various inclusive and exclusive channels.

Figure 1 illustrates the MicroBooNE LArTPC, which utilizes three readout wire planes at the anode to obtain signals induced from drifting ionization electrons produced from neutrino interactions within the active volume of the detector. Drifting ionization electrons first pass by the two induction planes (the “U plane” and “V plane”) and are finally collected by the collection plane (the “Y plane”). Being one of the first large LArTPCs constructed, MicroBooNE also bears important responsibilities in the research and development of the LArTPC technology, which is

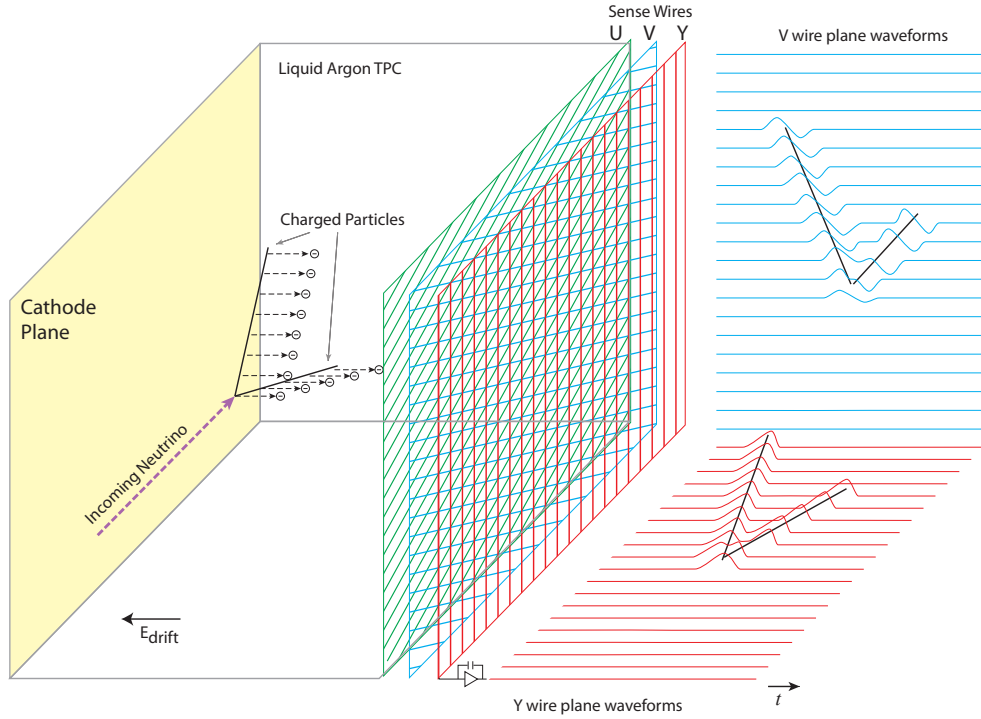


Figure 1. Illustration of signal formation in the MicroBooNE three-plane LArTPC, depicting the V plane (second induction plane) and Y plane (collection plane) TPC wire signals on the right of the image [7].

key to many future experiments [11, 12]. In particular, MicroBooNE pioneers the use of TPC cold electronics [13], which significantly reduce electronic noise [14]. Paired with enhanced TPC signal processing procedures [15, 16], the cold electronics lead to a much improved performance of the induction anode wire planes. Good matching of reconstructed ionization charge between the induction and collection wire planes is demonstrated for the first time [15, 16]. The significantly enhanced performance of the induction wire planes provides a solid foundation in applying the Wire-Cell (a software package [17, 18]) reconstruction to the MicroBooNE data.

Wire-Cell imaging [17, 18] is a 3D image reconstruction method developed by the MicroBooNE collaboration. The main idea of Wire-Cell imaging is to reconstruct 3D images in a tomographic way without involving heuristic topological assumptions (e.g. track or shower) prior to the pattern recognition stage. Using time and geometry information, Wire-Cell constructs the 3D space points given the observed 2D images (wire vs. time) [17]. Afterwards, Wire-Cell utilizes the charge-matching constraint and the sparsity condition to reduce ambiguities in 3D space points introduced by the wire readout. The charge-matching constraint requires that the same amount of reconstructed ionization charge should be seen by each (induction or collection) wire plane locally. The distribution of ionization electrons in space is expected to be sparse, typically occupying less than 10% of the local bounding volume that contains the activities, for any physical signals [18].

This sparsity condition is applied through the compressed sensing technique [19] via the coordinate descent algorithm [20]. The end results of Wire-Cell imaging are 3D space points with associated charges. Figure 2a shows an example of the Wire-Cell imaging result. We note that in dealing with imperfect detector performance in MicroBooNE (e.g. dead channels) [14], special algorithms, including two-wire-plane tiling and deghosting, are necessary to retain the quality of 3D images. Here, “tiling” is a technique that correlates 2D projections from multiple wire planes to obtain 3D information; and “deghosting” is a technique that removes ghost energy depositions caused by ambiguities in the 2D to 3D reconstruction [18].

In the MicroBooNE LArTPC, well-reconstructed 3D images form a good foundation for the neutrino interaction selection [21, 22], which is a challenging task for LArTPCs operating on the surface because of the numerous traversing cosmic-ray muons. A set of clustering techniques on the 3D space points [18] was developed to separate neutrino interactions from cosmic-ray muons. A many-to-many TPC-charge to PMT-light matching algorithm [18] is applied to match isolated TPC clusters to patterns of the PMT light signals, so that the neutrino interaction candidates in coincidence with the Booster Neutrino Beam (BNB) spill ($\sim 1.6 \mu\text{s}$ long) can be selected. Here, a “TPC cluster” is defined to be a set of space points that are connected in space, given that the LArTPC is a fully active tracking calorimeter. In addition, the links between the 3D space points and their 2D projections are also kept. To further reject cosmic muon backgrounds that occur in random coincidence with the BNB spill, a selection based on the TPC geometry is applied to reject cosmic-ray muons passing through the active TPC volume. The cosmic-ray muons that stop inside the TPC active volume can be determined by identifying the direction of the muon track. A 3D trajectory and dQ/dx (ionization charge loss per unit length) fitting procedure [22] was developed to identify Bragg peaks and reject those stopped muons. Note that the measurable dQ/dx is directly related to the previously mentioned dE/dx (energy loss per unit length). These tools are used to form a generic neutrino selection, which aims at rejecting cosmic-ray muon background while retaining high efficiencies for all kinds of neutrino interactions. This generic neutrino selection achieves 80% and 88% efficiencies for inclusive ν_μ charged-current ($\nu_\mu\text{CC}$) and ν_e charged-current ($\nu_e\text{CC}$) interactions with a 14.9% cosmic-ray background contamination in the selected events [21]. The generic neutrino selection serves as a good basis to further apply pattern recognitions and identify various neutrino flavors with high efficiencies and purities. Figure 2b shows an example of the Wire-Cell generic neutrino selection result. In addition to the removal of cosmic-ray activity, about 80% (90%) of the neutrino charged-current interactions have greater than 80% (70%) of their energy deposits reconstructed and included in the resulting 3D TPC clusters [18]. This provides a solid foundation to the subsequent 3D pattern recognition.

This paper summarizes the pattern recognition techniques developed and applied in Wire-Cell for the high-performance inclusive $\nu_e\text{CC}$ and $\nu_\mu\text{CC}$ event selections [23, 24]. Both traditional algorithms (non-machine-learning algorithms) and machine-learning algorithms are used in this pattern recognition procedure. Some of the basic tools — the track trajectory and dQ/dx fitting, for example — are improved versions of the techniques developed for generic neutrino detection [21, 22]. This fitting algorithm is expanded to fit multiple tracks and multiple vertices rather than fitting a single track. Figure 3 shows the overall flow of the Wire-Cell pattern recognition, which consists of two iterations. In the first iteration, we focus on neutrino vertex identification. An identified neutrino vertex location significantly reduces the possible solution space for other pattern recognition tasks.

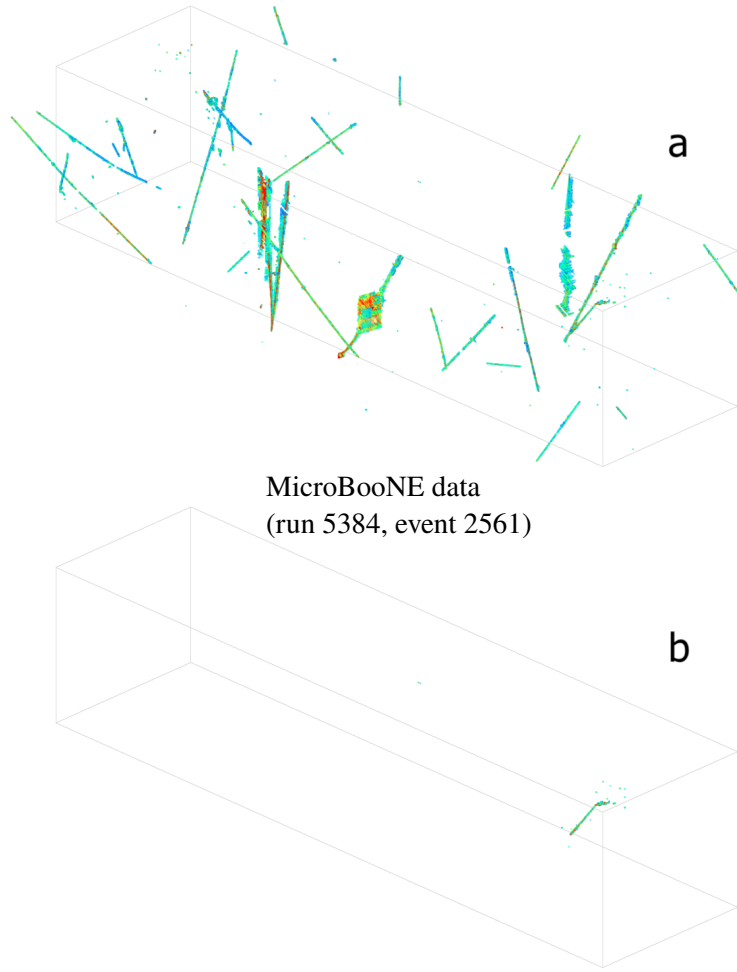


Figure 2. Example event displays of the Wire-Cell 3D imaging results before (panel a) and after (panel b) generic neutrino selection. Most TPC activities in panel a are identified as cosmic rays and removed in panel b. What remains are likely associated with a neutrino interaction and will be the inputs for the Wire-Cell pattern recognition algorithms described in this paper.

Here, solution space includes the origins and directions of particles as well as the mother-daughter relationships between them. In the second iteration, full particle flow trees are reconstructed based on the neutrino vertex locations and other intermediate objects reconstructed in the first iteration, e.g. track segments and initial showers. Here, a particle flow tree is a tree structure that contains the mother-daughter relationship for all particles in a set. At the beginning of the first iteration, vertices are defined by searching for kinks and splits in the reconstructed 3D images (section 2.1). With vertices determined, track segments between vertices are defined. A 3D vertex fitting technique (section 2.2) can then be used to refine the 3D vertex and connected track trajectories. Particle identification (PID) is subsequently performed on segments using dQ/dx and event topology information (section 2.3). Event topology information is primarily targeted toward electromagnetic (EM) shower identification (i.e. track/shower topology separation). Using the PID information, directions of particles can be determined in many cases. The neutrino interaction vertex is determined

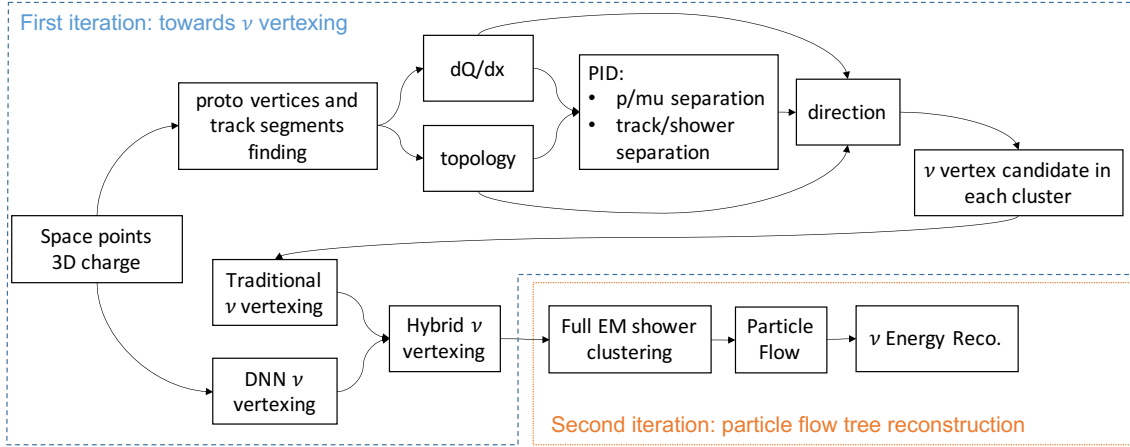


Figure 3. Overview of the Wire-Cell pattern recognition procedure. The blue dashed line indicates steps of the first iteration towards neutrino vertexing; the orange dotted line indicates steps towards particle flow tree reconstruction with a reconstructed neutrino vertex. Please note, intermediate objects reconstructed in the first iteration, e.g. track segments and initial showers, are also used in the second iteration, in addition to the reconstructed neutrino vertex. More details can be found in the text.

using the reconstructed particle directions and later improved with a deep neural network (DNN, section 2.5). With the neutrino interaction vertex reconstructed in the first iteration, EM showers can be fully clustered (section 3.1), and often contain several separated sub-clusters. Finally, a particle flow tree is reconstructed including the π^0 s reconstructed from EM showers (section 3.2).

Figure 4 shows an example of the pattern recognition at different stages from a neutrino interaction event in the MicroBooNE data. Colors in panels (d) and (e) represent the magnitude of the reconstructed dQ/dx , which is especially important for PID and determining the directions of particle trajectories. As we currently do not differentiate particles and anti-particles, in the legend, the charged particles can also refer to their antiparticles, e.g. e^- can refer to both e^- or e^+ , while the final state charged particles from neutrino mode interactions are mostly negatively charged leptons and positively charged pions.

The traditional algorithms in the Wire-Cell pattern recognition introduced in this article are developed using hundreds of MicroBooNE BNB data events, while the training of the DNN neutrino vertexing and the evaluation of the performance of this pattern recognition are performed using high-statistics MicroBooNE Monte-Carlo (MC) simulation events. E.g. in this paper, the DNN neutrino vertexing model is trained with 48000 selected ν_e charged-current events which leads to significant performance boost. The MicroBooNE MC simulates neutrino interactions using a flux model following ref. [25] and using the GENIE v3.0.6 neutrino event generator with a MicroBooNE tune [26, 27]. The particles produced in this neutrino interaction are processed through a realistic detector response model which takes the energy depositions from a Geant4 simulation [28], determines the ionization charge considering recombination [29, 30], drifts them through the TPC electric field including the space charge effect [31, 32], diffuses the charge [33], and calculates the resulting signal on the wires using a sophisticated electric field response [15]. This result is in good agreement between data and MC at a fundamental level [16]. Moreover, the MicroBooNE MC simulation used in this analysis adopts a scheme of overlaying the simulated neutrino interactions

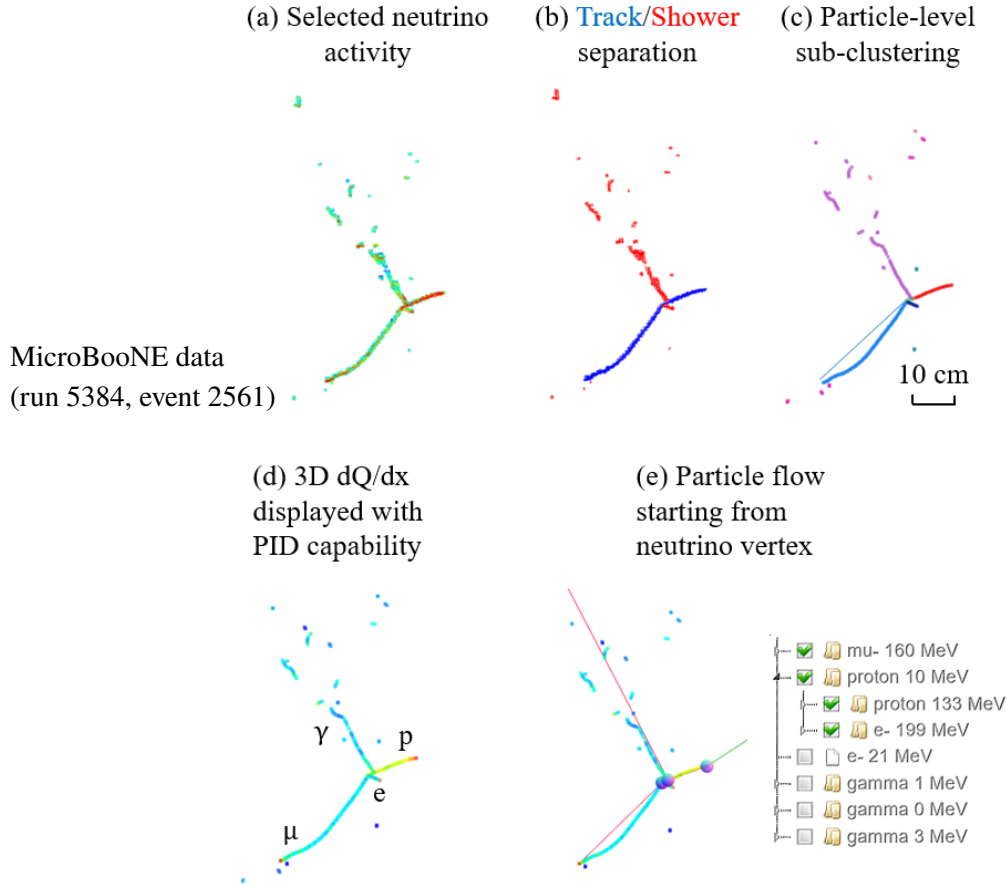


Figure 4. Displays of the Wire-Cell pattern recognition results at different stages. (a) In-beam candidate neutrino cluster selected by the generic neutrino selection. The color scale represents the reconstructed charge associated with each space point. (b) Identified tracks and EM showers, which are displayed in blue and red, respectively. (c) Identified particles (or track segments), which are displayed in different colors. (d) Fitted dQ/dx associated with each piece (~ 6 mm) along the trajectories. The blue, cyan, green, yellow, and red colors roughly correspond to $1/3$, 1 , 2 , 3 , and 4 times the dQ/dx of a minimum ionizing particle (MIP). The short track labeled as electron is likely to be a mis-reconstructed pion here. The EM shower (listed as electron in the particle flow) is labeled as a γ , given it is emitted by a proton track. It could be one of the two γ s from a produced π^0 . This γ is likely converted to electron-positron pair immediately. (e) Reconstructed particle flow starting from the primary neutrino interaction vertex. The folder icon indicates the corresponding particle has reconstructed daughter particle(s), while the file icon indicates the corresponding particle has no reconstructed daughter particle(s). Particles with check marks in the check-boxes are shown in panel (e) with rainbow circles marking the starting positions and line segments marking the starting and ending positions. A web based event display (BEE) for this example can be found here: <https://www.phy.bnl.gov/twister/bee/set/uboone/reco/2021-09/lee/event/1/>.

with the dedicated off-beam data that is randomly triggered when no neutrino beam spill is received, therefore the systematic uncertainties occurring in the simulation of readout noise, time-invariant detector response, and cosmic-ray backgrounds are avoided or largely reduced. Such MC simulation is referred to as overlay MC. This paper is organized as follows. Section 2 describes algorithms

of the first iteration towards the identification of the neutrino vertex. Follow up steps towards full particle flow tree reconstruction are introduced in section 3. Neutrino energy reconstruction is described in section 4. The conclusions are found in section 5.

2 Initial iteration towards neutrino vertex identification

A key step of the neutrino event reconstruction with LArTPCs is to identify the neutrino interaction vertex location. In this section, we describe two neutrino vertex identification algorithms. The first algorithm is a non-machine-learning one, utilizing particle direction information and event topology information. The second one is a machine-learning based algorithm, utilizing a sparse regression segmentation neural network. The final neutrino vertex position is determined by combining results from both algorithms. As described in section 1, to obtain the particle direction and event topology information, dedicated algorithms for track segment finding, dQ/dx fitting and track shower separation are performed. These algorithms are introduced in this section (sub-section 2.1, 2.2 and 2.3). Their results are not only useful for the neutrino vertex identification but also the full particle flow reconstruction which will be introduced in section 3.

2.1 Determination of vertices and track segments

In Wire-Cell, a *track segment* is defined as a subset of a TPC cluster including a set of 3D space points and associated 2D (projection) pixels with deposited charge information. Based on a track segment, the best-fit 3D trajectory points with about 6 mm spacing are obtained by minimizing a charge-weighted distance. That list of trajectory points is defined as a *track trajectory*. A key technique in determining vertices and track segments is the multi-track trajectory and dQ/dx fitting, which is an expansion of the single-track trajectory and dQ/dx fitting technique used in the generic neutrino selection [22]. Figure 5 illustrates the relationships between key concepts used in the multi-track trajectory and dQ/dx fitting algorithm.

The single-track fitting technique development was inspired by the projection matching algorithm [34]. First, an initial seed of the track trajectory is obtained by constructing a Steiner-tree graph [35] from the 3D points in the cluster and finding the shortest paths between extreme points. The Steiner-tree ensures that points associated with the largest charges are included in the initial seed. Then, the best-fit 3D trajectory points with about 6 mm spacing are obtained by minimizing a charge-weighted distance, which is constructed to compare the 2D measurements in time-versus-wire views from the three wire planes to the predictions given a 3D trajectory point. A numerical solver for large linear systems (BiCGSTAB [36]) is utilized to perform a minimization. With the trajectory determined, the dQ/dx associated with each trajectory point can be obtained by minimizing the squared difference between the reconstructed and predicted ionization charge. A parameterized model is used to predict the measured charge taking into account the diffusion of ionization electrons during transportation and the smearing of the charge distribution in the signal processing. This two-step procedure is adopted to avoid a nonlinear fitting process, ensuring the stability of the fit. Regularization on smoothness is included to further improve the dQ/dx fitting performance. More details about this single-track fitting technique can be found in ref. [22].

Moving towards the new multi-track fitting algorithm, major improvements include:

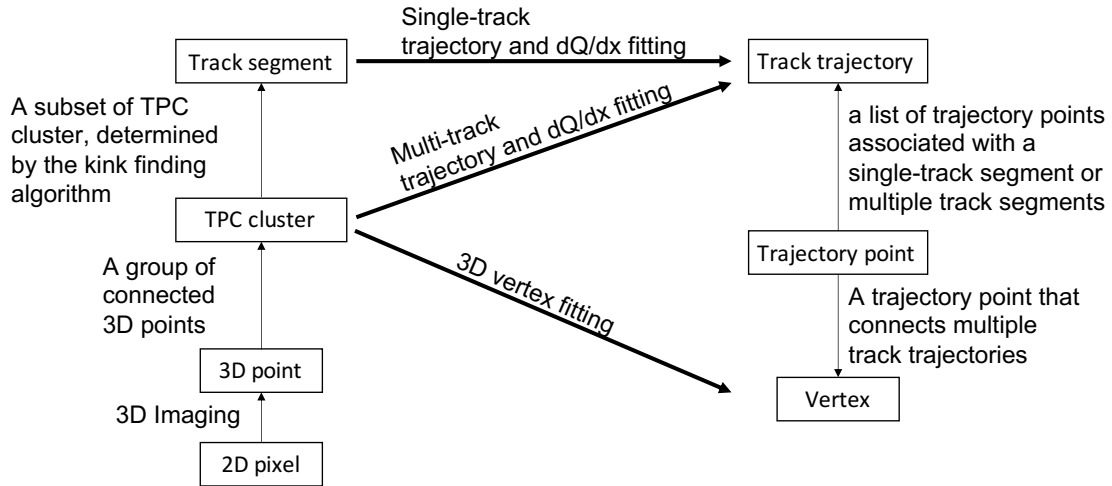


Figure 5. This diagram illustrates relationships between key concepts used in the multi-track trajectory and dQ/dx fitting algorithm. More details about the “3D Imaging”, the “Single-track trajectory and dQ/dx fitting” and the “3D vertex fitting” algorithms could be found in [18, 22] and section 2.2 respectively.

- The new algorithm handles simultaneous fitting of multiple track segments in 3D. In addition to the trajectory points that only belong to a single track segment, a new type of data product, vertex, is added. A vertex can be connected to one or more track segments.
- The distance between two adjacent trajectory points is regulated to be as close to 0.6 cm (twice the MicroBooNE TPC wire pitch) as possible, which makes the dQ/dx fitting easier. The more uniform step size also has advantages in particle identification (PID) given the distribution of dE/dx depending on the step size (dx).
- Grouping of the 2D pixels in the trajectory fitting is improved to better handle multiple track segments at once. The determination of the 2D pixel group for each track segment takes into account the associated 3D space points with 2D pixels. The 3D space points are much easier to cluster with the closest track trajectory, while the 2D pixels suffer more from ambiguities because they are projections.
- A dedicated fitting algorithm to determine the vertex position in 3D is introduced. See more detailed discussions in section 2.2.

An iterative approach is used to determine track segments and vertices in a TPC cluster, which allows one to systematically examine the TPC cluster given its various possible topologies. This approach is described below:

- Two extreme 3D space points (a pair of end points with the largest distance between them) are found on the TPC cluster, and a single track trajectory is fitted to connect the two extreme points.

- An algorithm is run to search for kinks¹ along the track trajectory from one end to the other. When found, the track trajectory is broken at the kink vertex into separate segments, then the track trajectory and dQ/dx are fitted again. Segmenting the track helps prevent track fitting from bypassing a kink by requiring the trajectory to pass through the vertex connecting the segments. This process is repeated until no more kinks are found.
- Track segments associated with existing track trajectories are removed from the TPC cluster and the remaining pieces are examined to search for new track segments. The newly found track segments are connected back to the existing track segments to seed new vertices. Track trajectory and dQ/dx fitting is repeated until no new track segments are found.
- Additional empirical algorithms are used to merge very close vertices to each other. For more details, please see ref. [37, [NeutrinoID_proto_vertex.h#L2231](#)].

The order of examinations (e.g. searching for kinks along a track or perform new track fitting) is fixed given a certain input so that the results of applying the aforementioned algorithms are reproducible. Figure 6 shows an example of the multi-track trajectory and dQ/dx fitting for the event shown in figure 4.

2.2 3D vertex fitting

In order to improve the quality of the multi-track trajectory and dQ/dx fitting (described in section 2.1), a dedicated algorithm is introduced to better determine the position of the vertex when multiple track segments are connected. Note, this vertex fitting algorithm is used to improve the precision of all vertices, both primary and secondary. The neutrino vertex (primary vertex) identification algorithms introduced later in section 2.4 and 2.5 are used to find the primary vertex.

Instead of operating on 2D projections, the vertex fitting procedure is performed on the 3D space points. First, we denote the position of the vertex to be determined as (x, y, z) . For the 3D space points associated with each track segment i , the points that are within a certain range (with a minimal distance of 1.5 cm and a maximal distance of 6 cm) are selected. Requiring a minimal and maximal distance reduces the bias in the fit, as points too close to the vertex can bias the fit due to incorrect associations with track segments and points too far from the vertex can bias the fit for tracks that change direction. A principle component analysis (PCA) is then performed:

- The center of the 3D points is (x_{i0}, y_{i0}, z_{i0}) . We further define $\vec{r}_i = (x_{i0} - x, y_{i0} - y, z_{i0} - z)$.
- The three eigenvectors of the PCA are denoted as \vec{v}_{i1} , \vec{v}_{i2} , and \vec{v}_{i3} , with the eigenvalues as λ_{i1} , λ_{i2} , and λ_{i3} , respectively. The distance from the vertex position to the main axis (first principal component) of the 3D points can thus be written as $(\vec{r}_i \cdot \vec{v}_{i2})$ and $(\vec{r}_i \cdot \vec{v}_{i3})$ for the second and third principal component directions respectively.

With these definitions, we can thus form a global test statistics to determine the vertex position:

$$T = \sum_i \left(\frac{\lambda_{i1}}{\lambda_{i2}} (\vec{r}_i \cdot \vec{v}_{i2})^2 + \frac{\lambda_{i1}}{\lambda_{i3}} (\vec{r}_i \cdot \vec{v}_{i3})^2 \right) + \lambda \cdot \left((x - x_{\text{org}})^2 + (y - y_{\text{org}})^2 + (z - z_{\text{org}})^2 \right), \quad (2.1)$$

¹Sharp turns with local angle $> 25^\circ$.

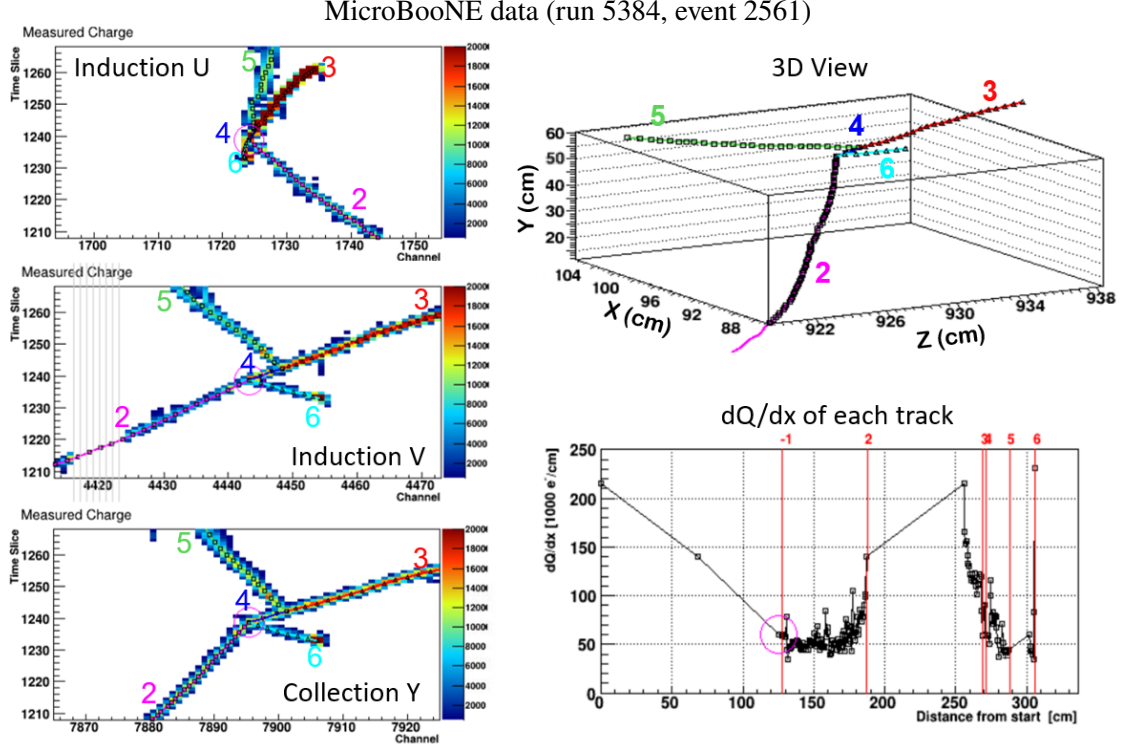


Figure 6. Example display of the results from multi-track trajectory and dQ/dx fitting. The three panels on the left show the reconstructed ionization charge on different planes (two induction planes and one collection plane) overlaid with the projection of the best-fit track trajectories. The track trajectories are color coded and labeled with numbers of the same color. The top right panel shows the 3D view of the best-fit track trajectories. The bottom right panel shows the best-fit dQ/dx for all track trajectories. dQ/dx distribution for each stopped track can be used to determine the direction of the track according to the Bragg peak. The color coding can be used to link the same trajectories in 2D and 3D views. Also using the color coded numbers, the trajectories can be further linked to the dQ/dx fitting in the bottom right panel. The gray lines in the induction V view represent non-functional channels. In the induction U view, short track number 6 is highly compacted because of the nature of 2D projection.

where the first term in the summation minimizes normalized distance between the vertex and the main axis of the 3D points from a track segment. The second term in eq. (2.1) minimizes the distance between the vertex position and its initial value $(x_{\text{org}}, y_{\text{org}}, z_{\text{org}})$ with a regularization strength λ . The addition of this term is to avoid unrealistic fitting results in certain situations (e.g. the track segments are all nearly parallel to each other). The minimization of eq. 2.1 can be solved with linear algebra, which ensures the stability of the fitting procedure. Figure 7 shows the best-determined track trajectories before and after the 3D vertex fitting. The improvement close to the vertex is most prominent in the middle row plots.

2.3 Separation of track and shower topology

We use three methods to differentiate tracks from electromagnetic (EM) showers at different energies. They are: i) multiple Coulomb scattering measurements to differentiate track-like electrons from other particles at low energy, ii) the presence of additional isolated clusters in close proximity to

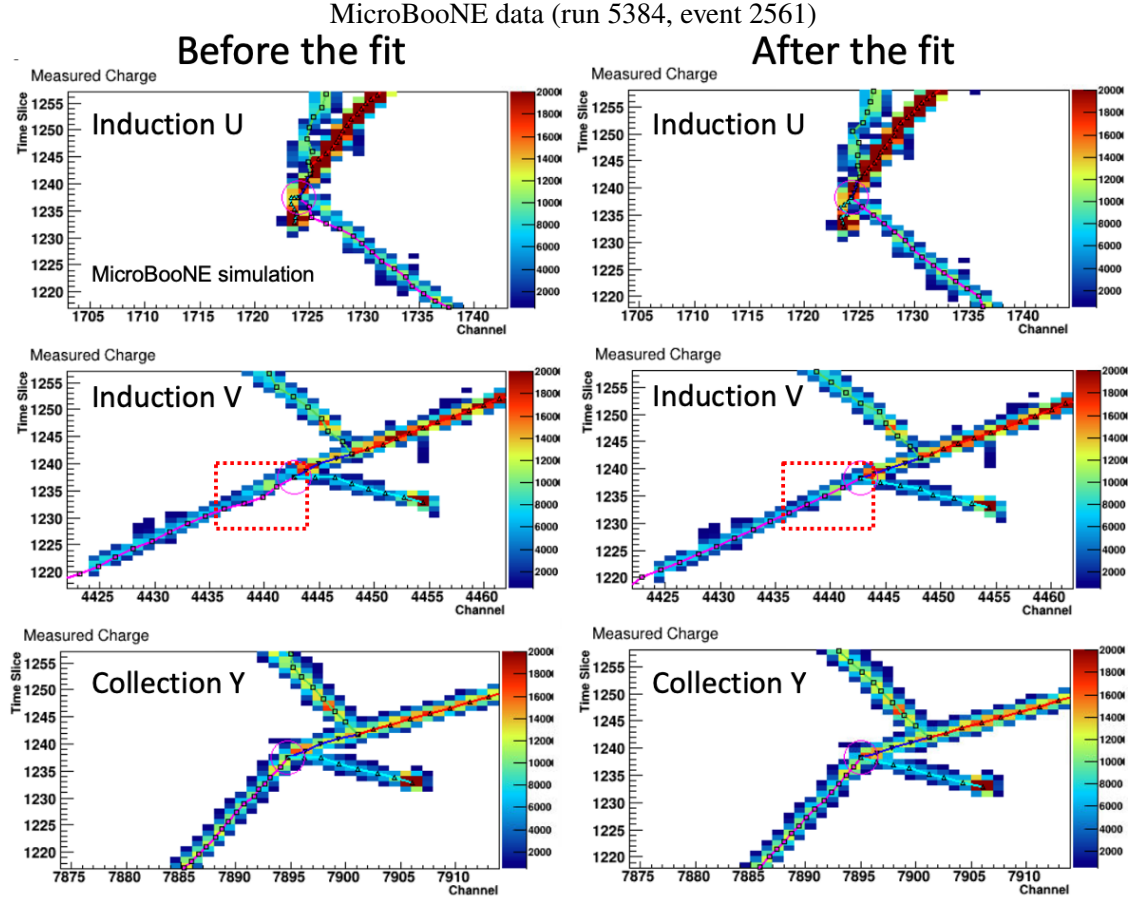


Figure 7. Best-determined track trajectories before (left) and after (right) the 3D vertex fitting are displayed as lines on the reconstructed ionization charge. The z -axis represents the amount of ionization charge. The improvement of trajectory fitting near the vertex region (highlighted with dashed red boxes) is most prominent in the middle row plots (induction V plane). The original bias in the fitted track trajectory is the result of the incorrect association of the 2D charge given that the initial track trajectory is determined from the shortest path algorithm. The accurate vertex position determination allows the shortest path algorithm to provide improved initial track trajectories to determine the charge association for the fitting algorithm.

track-like electrons, also at low energy, and iii) the EM shower width perpendicular to the trajectory direction for high energy electrons. We discuss each method in more detail below.

At low energies ($\mathcal{O}(10\text{ MeV})$), the bremsstrahlung effect becomes less important for electrons compared to the ionization effect. The low-energy electrons could behave like tracks instead of electromagnetic showers (i.e. multiple tracks including electron-positron pairs produced by the bremsstrahlung photons). Nevertheless, the combination of an electron's small mass and ample interactions with atomic electrons in liquid argon means low-energy electrons have more large-angle scattering than other particles. An optimal way to differentiate low-energy electrons from low-energy muons, pions, or protons in LArTPCs is to use the multiple Coulomb scattering (MCS) method. For electrons, the MCS-derived momentum assuming a muon mass [38] will be much lower than that reconstructed from the track residual range. We use a simplified algorithm to differentiate a wiggled track from a straight track. We divide each track segment into many sections with each section about

10 cm long. For each track section, we calculate the straight-line distance between the starting and ending points. Along its fitted track trajectory, the trajectory length in this track section is also calculated. For a wiggled track segment, the direct length after summing over all track sections is expected to be much smaller than the trajectory length after summing over all sections. For a straight track segment, on the other hand, the ratio of these two lengths is close to one.² Figure 8a shows such an example.

For low-energy electrons, in addition to the above mentioned length ratio, the presence of nearby isolated clusters (i.e. bremsstrahlung photons emitted during the development of EM shower) can also be used to identify an EM shower. Figure 8b shows an identified EM shower in which case two isolated clusters are associated with the wiggled track. The usage of isolated clusters for EM shower identification requires an alignment of the isolated clusters to the shower trunk (first few centimeters of the EM shower) direction. A good neutrino vertex reconstruction plays an important role in determining the shower trunk and its direction.

For high-energy electrons (see figure 8c), the resulting EM showers tend to extend further in the projections perpendicular to the momentum direction because a large number of secondary electrons and positrons are produced. The measurement of the perpendicular spread of associated 3D space points (reconstructed in the 3D imaging step) along the EM shower direction can be used to identify an EM shower. The evolution of the width along the EM shower direction (e.g. a gradual increase of the width) can also be used to determine the starting point of the EM shower, which can be further used to determine the direction of the EM shower using the stem of the EM shower (track-like segment at the beginning of an EM shower). The direction of the EM shower is a crucial input for neutrino vertex reconstruction. For any projective readout, one expects ambiguities in the transverse plane (parallel to the readout plane). The measurement of widths must take this into account. We define the direction of the shower to be \vec{r}_{mom} and the direction of drifting electrons (perpendicular to the transverse plane) to be \vec{r}_{drift} . The measurement of the widths is performed along the direction of $\vec{r}_{\text{mom}} \times (\vec{r}_{\text{mom}} \times \vec{r}_{\text{drift}})$, which is perpendicular to both the momentum direction \vec{r}_{mom} and the vector representing the ambiguities $\vec{r}_{\text{mom}} \times \vec{r}_{\text{drift}}$.

2.4 Neutrino vertex identification with traditional techniques

The reconstructed particle direction provides important information for the neutrino interaction vertex identification. In the primary neutrino interaction vertex, the particles entering into the vertex would be the neutrino (invisible to the detector) and the argon nuclei (also invisible to the detector). Therefore, all tracks and EM showers would travel outward from the primary neutrino interaction vertex. The situation is different for a secondary interaction vertex in the main interaction cluster containing the true neutrino interaction vertex. In this case, the particles entering into the vertex would be a charged particle (e.g. produced by the primary neutrino interaction) and an argon nuclei (invisible to the detector). The rest of tracks and EM showers would travel outward from the secondary vertex. Given each interaction always involves an argon nuclei, the chance to have two visible particle tracks entering into a vertex is negligible. Therefore, the determination of the track and EM shower directions is crucial. As described in section 2.3, the gradual increase of the EM shower's width can be used to determine its direction. For tracks, the identification of a Bragg peak can be used to determine the direction. Figure 9 shows the dQ/dx distribution as a function

²Typically, a value of 0.9 or greater is required empirically.

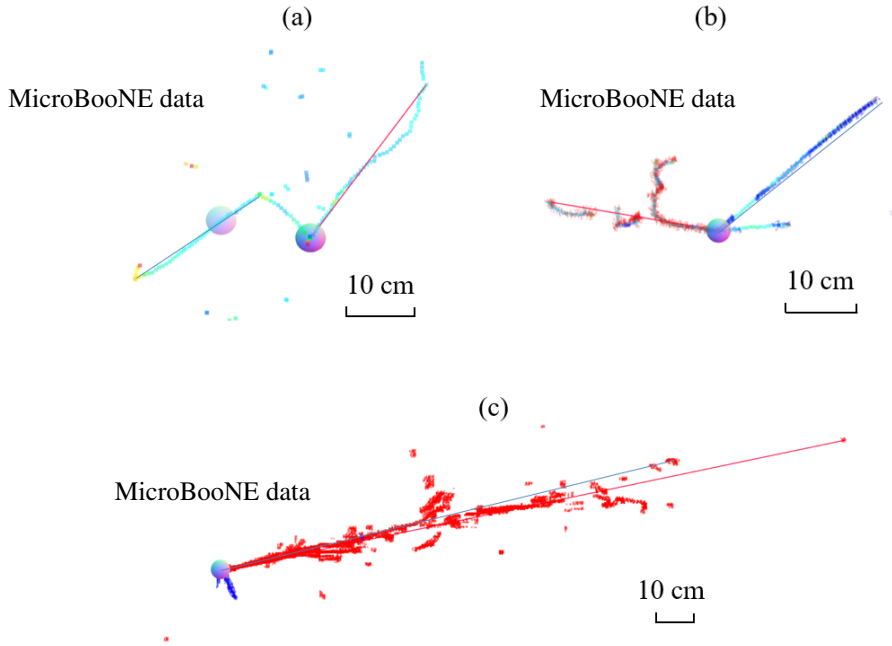


Figure 8. Example of various EM showers. (a) A very wiggled track. Color represents reconstructed charge along the fitted track trajectories. The blue (red) color indicates low (high) ionization charge. (b) A scattered track with two isolated showers aligned with a straight line. Shower-like objects are indicated by red color overlay; track-like objects are indicated by blue color overlay. (c) A high-energy EM shower. Shower-like objects are indicated by red color overlay; track-like objects are indicated by blue color overlay. Rainbow circles indicate particle starting positions in all three panels.

of the residual range (distance to the end point of a track). The dQ/dx rise can be used to mark the end point of a track. In addition, two clearly separated bands, indicated by black and red lines, correspond to protons and muons/pions. And this can be used to separate protons from muons/pions.

In addition to the track and EM shower's directional information, the following information is taken into consideration in determining the neutrino vertex (primary vertex):

- The position of the vertex. For example, the neutrino vertex in an interaction cluster tends to be more upstream than any other activity near the neutrino beam direction. On the other hand, a vertex at the detector boundary is less likely to be the neutrino vertex.
- The number of directly connected tracks and EM showers. Statistically, more tracks and showers are connected to the neutrino vertex than secondary vertices.
- A set of connectivity rules for the activities (tracks or showers) near the vertex. For example, there should not be two tracks going into the same vertex. There should not be one EM shower going into a vertex and a track coming out of the same vertex. In addition, the situation that one track comes in and another track goes backward with a large angle is not preferred.
- Some vertex candidates can be excluded by the local event topology. For example, vertices inside an EM shower and vertices from a delta-ray emerging from a straight muon track can be safely excluded. For details, please see ref. [37, [NeutrinoID_track_shower.h#L1356](#)].

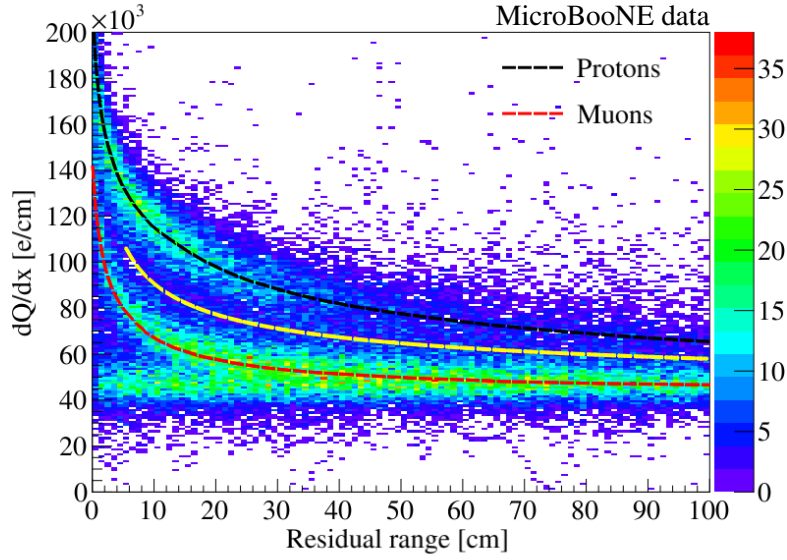


Figure 9. dQ/dx vs residual range from MicroBooNE data. Two clearly separated bands, indicated by black and red lines, represent protons and muons. The measurement of dQ/dx near the end point of the track can be used to determine the track direction and particle identification. The band around $dQ/dx \sim 50\text{ k e/cm}$ extending to zero residual range represents the muons exiting the detector. The black and red curves are the predictions of most probable dQ/dx values for protons and muons, respectively, considering the recombination effect as reported in ref. [39]. The yellow curve indicates the empirical cut to separate protons and muons/pions.

In practice, each piece of information is summarized into a score. The vertex candidate with the highest score is determined to be the primary neutrino vertex. Given this method is surpassed by the Deep-learning neutrino vertex determination (discussed in section 2.5), the technical details of the score evaluation are skipped.

Using the strategies above, one neutrino vertex candidate can be identified in each TPC cluster. Given there could be multiple TPC clusters in each neutrino event, another algorithm is used to select the primary interaction vertex (neutrino vertex) from all vertex candidates, since the primary neutrino vertex may be outside the main TPC cluster (the TPC cluster most closely matched to the observed light pattern). Therefore, we utilize the position of each candidate vertex and the size and direction of its associated TPC cluster to select primary vertices among candidates in all TPC clusters. Figure 10a shows an example case where the neutrino vertex is inside the main TPC cluster consisting of multiple tracks. Figure 10b shows an example case where the neutrino vertex is located on a short proton track with two separated EM showers (reconstructed to be from a neutral pion decay) pointing back toward it.

The performance of the primary neutrino vertex identification with these traditional pattern recognition techniques has limitations. Among $\nu_\mu\text{CC}$ events (ν_μ charged-current interactions), about 70% of neutrino vertices are correctly determined, where a reconstructed vertex position is defined to be correct if it lies within 1 cm of the true neutrino vertex position. Among $\nu_e\text{CC}$ events (ν_e charged-current interactions), about 50% of neutrino interaction vertices are correctly identified. We describe further improvements of the neutrino vertex identification using deep learning techniques in section 2.5.

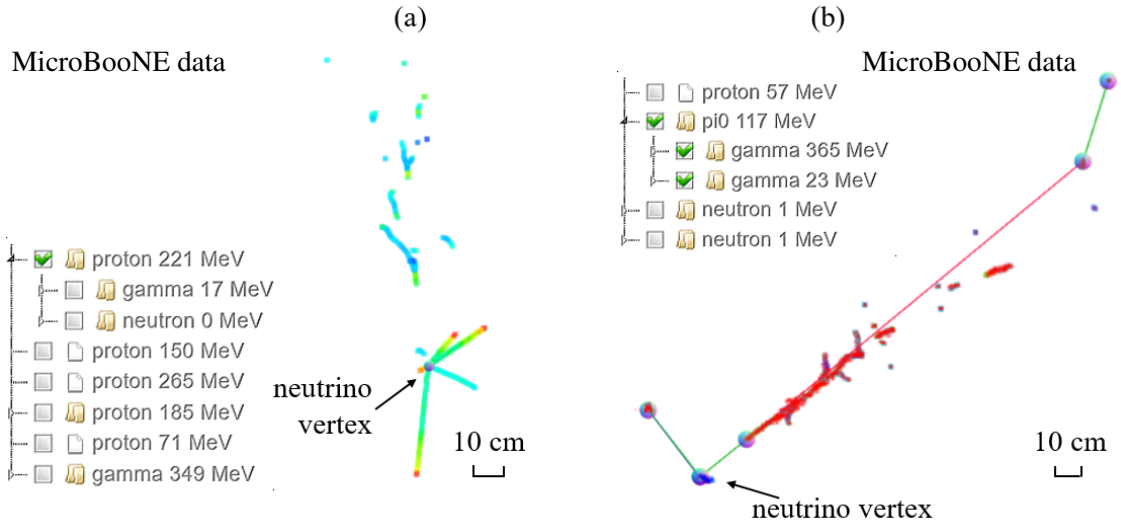


Figure 10. Illustration of the primary neutrino vertex determination. (a) the neutrino vertex is determined to be inside the main TPC cluster consisting of multiple tracks. (b) the neutrino vertex is determined to be at a short proton where two separated EM showers point back toward. This is a candidate $\text{NC}\pi^0$ production event. The reconstructed particle flow information is also displayed (Refer to the caption of figure 4 for more details).

2.5 Neutrino vertex identification with a regressional segmentation network

While traditional pattern recognition techniques provide reasonable performance in identifying the primary neutrino interaction vertex (section 2.4), an improved performance is desired to enhance the selection efficiency of ν_e and ν_μ charged-current interactions. We use a Deep Neural Network (DNN) on the reconstructed 3D space points and vertex candidates to further enhance the primary neutrino interaction vertex identification performance.

This neutrino vertex identification problem can be considered as a special case of an object detection task in computer vision. Object detection usually takes 3 steps: region proposal, object type identification, and object segmentation [40]. In the vertex identification case, we are targeting only one type of object: the neutrino vertex and the region of the object, so we focus on the region proposal step. We use a customized regressional segmentation network to predict the distance map of each voxel (a 3D pixel) to the neutrino vertex, which improves the training efficiency compared to only predicting whether a voxel is a vertex or not. We use SparseConvNet [41, 42], which makes the model training and inferencing (the process of using a trained DNN model to make predictions against previously unseen data) much faster and more scalable. Similar techniques were used in other MicroBooNE analyses [43]. The inferencing utilities are available in the Wire-Cell software package [44].

The key part of our regressional segmentation network is a Sparse U-Net [45] which extracts a feature vector for each pixel or voxel. Figure 11 shows details of a level 2 Sparse U-Net. One can add more levels by iteratively inserting more “Concat-Join” blocks. We use level 5 in this paper. More implementation details can be found in [46]. The SparseConvNet works on sparsified 2D or 3D images, which consist of a list of pixels (2D) or voxels (3D). The Wire-Cell reconstructed 3D points are placed into voxels, which is a cube with 0.5 cm in each of three dimensions. If multiple

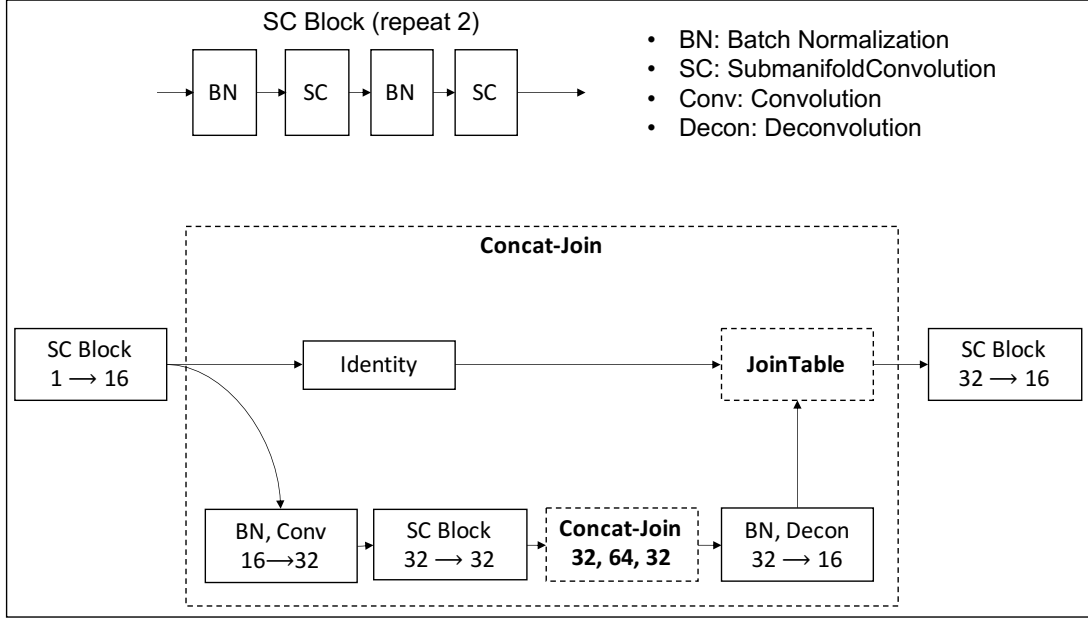


Figure 11. Illustration of the Sparse U-Net network structure from SparseConvNet [41, 42]. The bottom diagram shows a level 2 Sparse U-Net.

3D points fall into the same voxel, the values are averaged. The SparseConvNet takes two tensors as input. One is a coordinate tensor (integer type) with position information for each voxel. The other one is a feature tensor (float type) with the charge information. The output of the SparseConvNet is also a list of features for each voxel, for now we extract only one feature which we call *confidence value* related to the distance between a point and the truth vertex. The truth label for this *confidence value* is calculated before voxelization, and the equation used is:

$$\text{Conf}_{\text{truth}} = \exp\left(-\frac{\|\vec{x} - \vec{v}_{\text{truth}}\|^2}{2\sigma^2}\right), \quad (2.2)$$

where $\text{Conf}_{\text{truth}}$ is the truth label of the *confidence value*; \vec{x} and \vec{v}_{truth} are the reconstructed 3D charge and truth vertex positions; σ is a regularization parameter (1 cm is used). *Confidence values* from multiple voxels (pixels) form a “Confidence Map” [47]. Figure 12 shows an example of the input (charge) and label before voxelization.

The dataset used for training the network consists of 48k ν_e CC events simulated through the MicroBooNE simulation chain and Wire-Cell reconstruction. The dataset for validating the network performance consists of 4k ν_e CC events. The dataset for testing the network performance consists of 4k ν_e CC and 4k ν_μ CC events. The total dataset size is chosen based on the available official MicroBooNE simulation production sample size. Then we try to maximize the training set size while keeping enough for validation and testing. The choice of focusing on the ν_e CC event topology is motivated by the need to improve ν_e CC event selection, which is much more challenging than the ν_μ CC event selection. Nevertheless, the final testing results of the network show an improvement over the performance of the traditional algorithm (described in section 2.4) in both ν_e CC and ν_μ CC events.

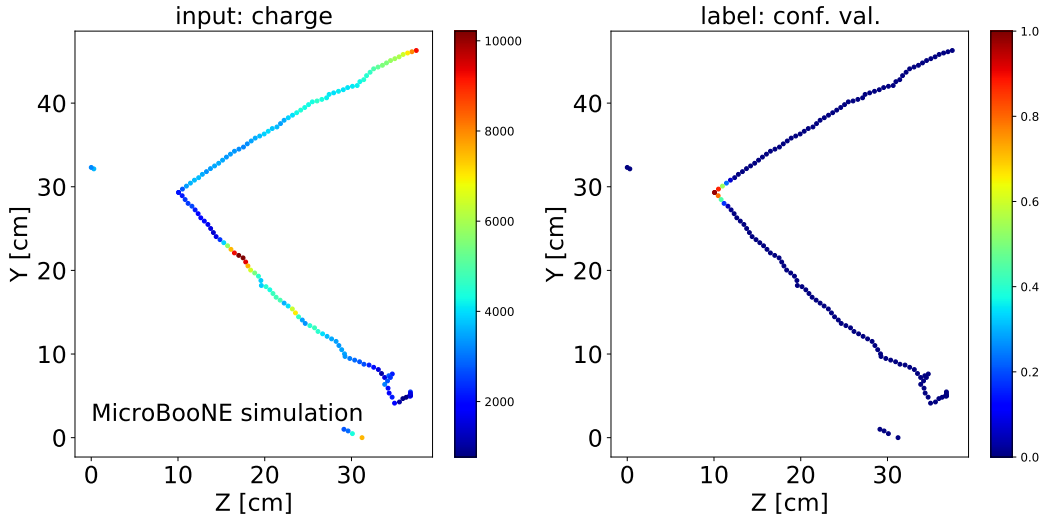


Figure 12. Example of input (charge) and label (confidence value) images before voxelization (showing a 2D projection). The color represents reconstructed charge in the left panel and confidence value in the right panel. From the right panel, we can see the confidence value becomes larger when getting closer to the vertex as defined in eq. 2.2.

Machine learning model parameters are estimated with the help of loss functions [48]. The loss function we used is:

$$\mathcal{L} = \frac{1}{N_{\text{voxel}}} \sum_{N_{\text{voxel}}} \|\text{Conf}_{\text{pred}} - \text{Conf}_{\text{truth}}\|^2. \quad (2.3)$$

The optimizer used is Adam [49]; the learning rate decays after each epoch as $l_0 \cdot \exp(-n \cdot l_d)$, where l_0 is 10^{-5} and l_d is 0.05 and n is the epoch number. Figure 13 shows the training progress, epoch average loss, and hit rate as a function of epoch number for the ν_e CC training and validation samples. “Hit rate” here is defined as the probability that the best predicted voxel is within 1 cm of the truth voxel (the voxel closest to the truth vertex). From figure 13, we can see that: i) the solution phase space is not fully covered by the training sample according to the gaps between the train and validation curves (we also observed the gaps getting smaller when increasing training samples); ii) there are signs of over-training; and iii) validation performance stopped improving after epoch 24. The results shown in this paper are based on the model at epoch 24.

To accommodate existing Wire-Cell reconstruction/pattern recognition algorithms, we select the neutrino vertex from the candidates provided by the traditional pattern recognition algorithms. We call this vertex the hybrid vertex. Figure 14 shows an event display to demonstrate finding the hybrid vertex based on a DNN vertex. The decision proceeds as follows:

- Find the voxel with the highest *confidence value*, which we call the “DNN Vertex”.
- Convert from voxel index back to continuous space coordinate.
- From the list of traditional vertex candidates, find the closest one to this DNN vertex in the coordinate space. (As described in section 2.4, the traditional neutrino vertex is also selected from these traditional candidates.)

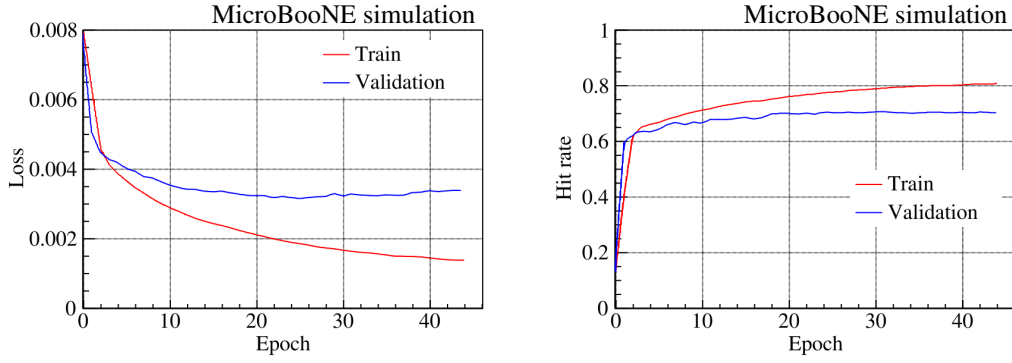


Figure 13. Left: epoch average loss as a function of epoch number for training (red) and validation (blue) samples; right: probability that the best predicted voxel is within 1 cm of the best truth voxel (voxel closest to the truth vertex).

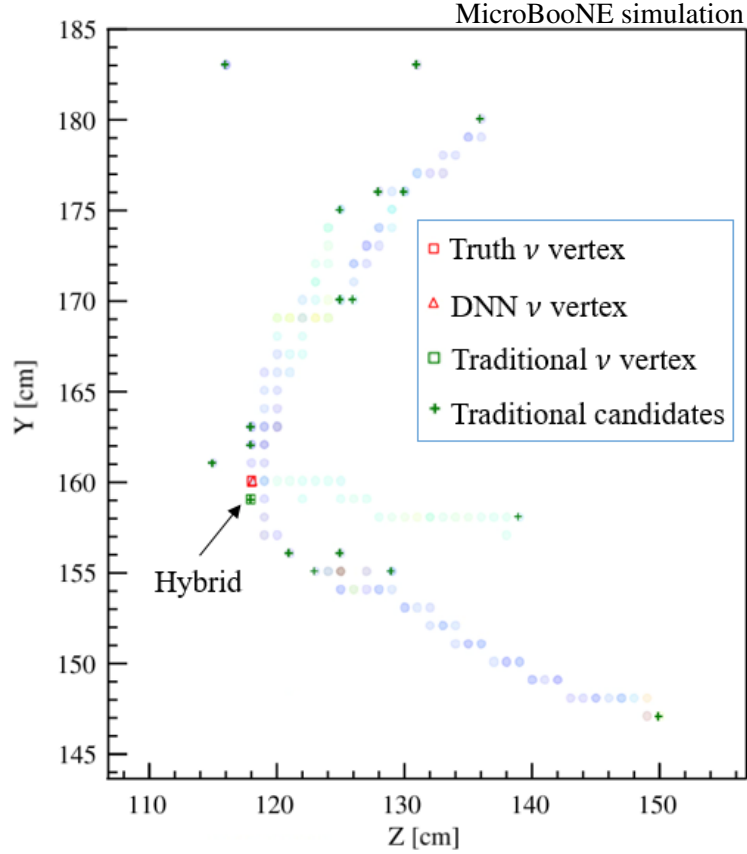


Figure 14. An event display to demonstrate finding the hybrid vertex based on a DNN vertex. Each filled circle represents a voxelized 3D charge projected on the Y-Z plane. In this example, the hybrid vertex (indicated by the black arrow) is different from the DNN one, which is better in this case.

- Check if the distance between this traditional candidate and the DNN vertex is smaller than a threshold, currently 2 cm. Note this “2 cm” threshold used in the DNN-traditional vertex matching should not be confused with the “1 cm” reco-truth matching threshold used in the vertex finding algorithm performance evaluations.
- If the previous distance check is good (< 2 cm) use this traditional candidate as the “hybrid vertex”; if not, use the “reconstructed traditional neutrino vertex” (section 2.4) as the “hybrid vertex”.

Figure 15 shows the performance of the trained model on the ν_e CC test samples. Compared to the performance of traditional pattern recognition algorithm, the hybrid algorithm leads to a relative 30% improvement in identifying the neutrino vertex. At low distance cut (~ 0.5 cm) between the truth and the reconstructed neutrino vertex, the traditional algorithm outperforms the DNN algorithm, because of the finite voxel size (0.5 cm in each dimension) and the smearing of the reconstructed 3D space points. Figure 16 shows the performance of the current best model on the ν_μ CC test samples. The hybrid algorithm leads to a relative 10% improvement in identifying the neutrino vertex, which is achieved on the ν_μ CC samples not used in the training. Furthermore, while the performance of the hybrid algorithm is slightly worse than that of DNN for ν_e CC events, it is slightly better for ν_μ CC. This is because the performance of traditional algorithm for ν_e CC is worse than that for ν_μ CC. As mentioned before, the hybrid vertex is used as the main result in the current workflow to accommodate existing Wire-Cell reconstruction/pattern recognition algorithms. In the future, it is likely we will revise the downstream reconstructions directly based on the DNN vertexing results then we do not need to use the “hybrid” one.

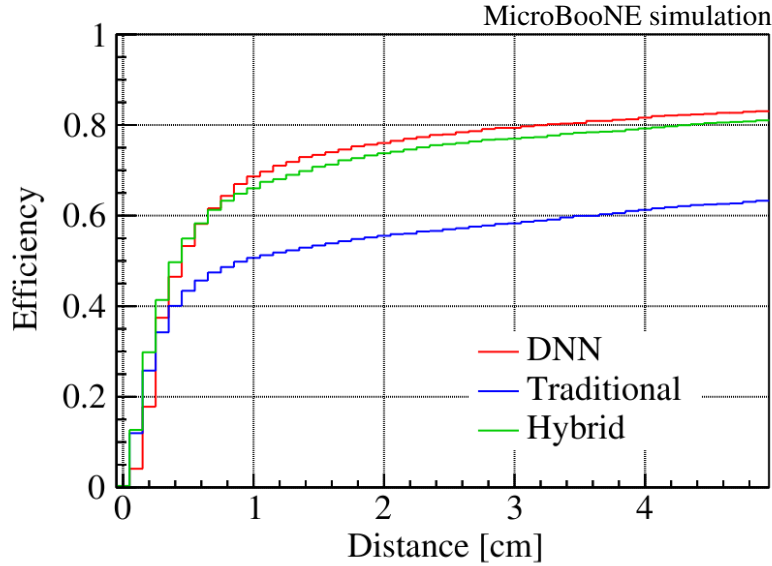


Figure 15. ν_e CC neutrino vertex identification efficiency as a function of the maximum distance between the reconstructed and the truth, i.e., the numerator of the efficiency is defined as the number of events with a neutrino vertex reconstructed within X cm of the truth, where X is the horizontal coordinate of the plot. Three algorithms are compared: Traditional, DNN and Hybrid.

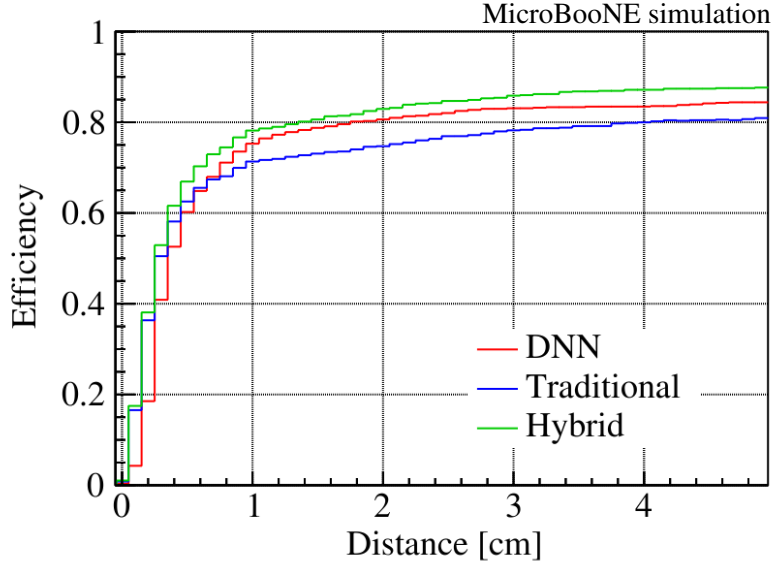


Figure 16. ν_μ CC neutrino vertex identification efficiency as a function of the maximum distance between the reconstructed and the truth, i.e., the numerator of the efficiency is defined as the number of events with a neutrino vertex reconstructed within X cm of the truth, where X is the horizontal coordinate of the plot. Three algorithms are compared: Traditional, DNN and Hybrid.

2.6 Evaluation of the neutrino vertex identification performance

The efficiency of the neutrino vertex identification plays a crucial role in ν_μ CC and ν_e CC event selections. Therefore, we performed evaluations of the neutrino, primary muon, and primary EM shower vertex reconstruction. *Pandora* [50] is another state-of-the-art traditional pattern recognition technique³ that is also used in MicroBooNE. We compare the performance of the Wire-Cell vertex identification with Pandora as a benchmark.

Two types of overlay MC samples are used to evaluate the neutrino vertexing performance. BNB ν overlay MC, which simulates neutrino interactions based on the BNB neutrino flux of which 99.5% are ν_μ 's, is used to evaluate the performance for ν_μ CC and NC events. BNB intrinsic ν_e overlay MC, which is a ν_e -enriched MC sample only simulating the electron neutrino interactions based on the $\nu_e/\bar{\nu}_e$ component in the BNB neutrino flux, is used to evaluate the performance for ν_e CC events.

The left panels of figure 17, 18, and 19 show distributions of the distance between the reconstructed neutrino vertex and true neutrino vertex for charged-current events in BNB ν overlay, neutral-current events in BNB ν overlay, and charged-current events in BNB intrinsic ν_e overlay samples, respectively. Percentages of events with a neutrino vertex reconstructed within 1 cm of the truth for these three samples are 70.3%, 49.3% and 65.2%; within 5 cm are 84.8%, 63.5% and 80.0%, respectively. In the right panels of these figures we calculate the percentage of events with the reco-truth vertex distance less than 1 cm.

³Some new developments of Pandora reconstruction start to involve deep-learning algorithms, but such developments were not included in this comparison and are yet to be used in the MicroBooNE experiment.

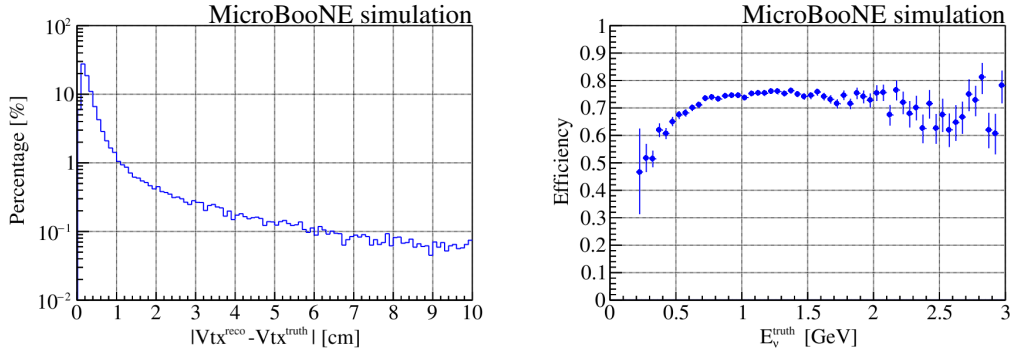


Figure 17. Neutrino vertex reconstruction of the ν overlay sample, charged-current interaction, reconstructed with Wire-Cell. Left: distance between the reconstructed and truth neutrino vertex; right: efficiency for events with vertex reconstruction position within 1 cm of the truth, as a function of truth neutrino energy. The error bars represent the statistical uncertainties.

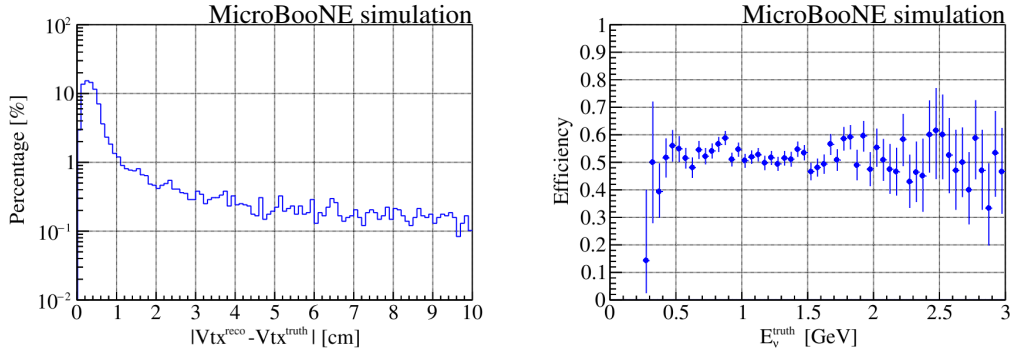


Figure 18. Neutrino vertex reconstruction of the ν overlay sample, neutral-current interaction, reconstructed with Wire-Cell. Left: distance between the reconstructed and truth neutrino vertex; right: efficiency for events with vertex reconstruction position within 1 cm of the truth, as a function of truth neutrino energy. The error bars represent the statistical uncertainties.

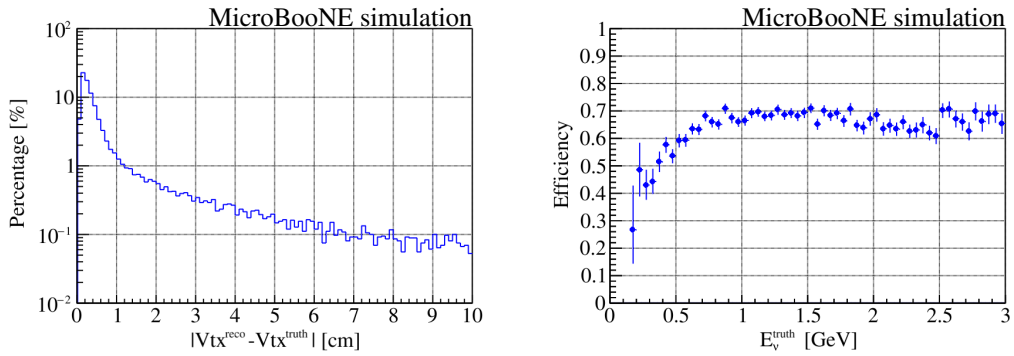


Figure 19. Neutrino vertex reconstruction of the intrinsic ν_e overlay sample, charged-current interaction, reconstructed with Wire-Cell. Left: distance between the reconstructed and truth neutrino vertex; right: efficiency for events with vertex reconstruction position within 1 cm of the truth, as a function of truth neutrino energy. The error bars represent the statistical uncertainties.

Table 1. The performance with the distance between reconstructed neutrino vertex and true neutrino vertex less than 1 cm. The performance are estimated with a BNB ν overlay sample, while charged-current ν_e event efficiency are estimated with a BNB intrinsic ν_e overlay sample.

	Pandora reconstruction	Wire-Cell Traditional	Wire-Cell DNN Hybrid
Charged-current ν_μ	70.3%	67.1%	72.9%
Neutral-current	49.3%	50.6%	52.1%
Charged-current ν_e	65.2%	50.1%	65.8%

Table 1 summarizes the performance comparison between Wire-Cell and Pandora [50] reconstructions. In order to make a fair performance comparison focusing on reconstruction algorithms, the same input events, with cosmic-rays cleaned up and unresponsive detector areas recovered using the upstream Wire-Cell imaging and clustering algorithms [18], are feed into both Wire-Cell and Pandora reconstruction workflows. Except for the inputs, the Pandora configuration is the same as used in ref. [51]. For charged-current events in the BNB ν overlay sample, Pandora reconstruction results in 70.3% good reconstructed vertices (within 1 cm of the true vertex), and Wire-Cell input plus Wire-Cell reconstruction leads to 67.1% (without DNN Vertexing) and 72.9% (with DNN Vertexing) good reconstructed vertices. For the neutral-current case, the Pandora reconstruction has 49.3% efficiency while the Wire-Cell reconstruction has 50.6% (without DNN Vertexing) and 52.1% (with DNN Vertexing). Neutrino vertex identification is generally harder for neutral-current interactions due to the absence of primary charged particles connecting to the neutrino interaction vertex and more complicated event topology. When the BNB intrinsic ν_e sample is used, Pandora has 65.2% efficiency while Wire-Cell has 50.1% (without DNN Vertexing) and 65.8% (with DNN Vertexing) efficiency. The DNN Vertexing model is trained on a ν_e charged-current sample and boosts the Wire-Cell vertexing performance by about 30% for samples of the same type while it does not degrade the performance on samples of other types. After using the DNN Vertexing, the Wire-Cell vertexing performance is comparable with Pandora reconstruction for all samples tested.

As noted earlier, the BNB intrinsic ν_e events can be used for the shower vertex reconstruction study as there are only ν_e CC interactions in these events, which are supposed to have an EM shower attached to each neutrino vertex. Figure 20 is a 2D distribution of distance between the reconstructed primary shower vertex and the true neutrino vertex vs the distance between the reconstructed neutrino vertex and the true neutrino vertex. For the Pandora reconstruction, the percentage of events that have both the reconstructed primary shower vertex and reconstructed neutrino vertex within 1 cm of the true neutrino vertex is 30.8%, while the Wire-Cell reconstruction achieves 62.0%. This is because that the hybrid Wire-Cell neutrino vertex is selected from a list of candidates including the shower vertices. So for ν_e CC interactions, when the neutrino vertex is identified correctly, there is a high chance that it is exactly the same position of a shower vertex. While the Pandora reconstruction takes a different approach that the shower vertices are reconstructed more independently from the neutrino vertex, so that the analyzers would have additional information to use. The better aligned neutrino and shower vertex should enhance the capability to differentiate primary electrons from photons and provide better a better capability in identifying the gap between the photon EM shower vertex and the neutrino vertex. Furthermore, compared to the 65.8% efficiency in correctly identifying neutrino

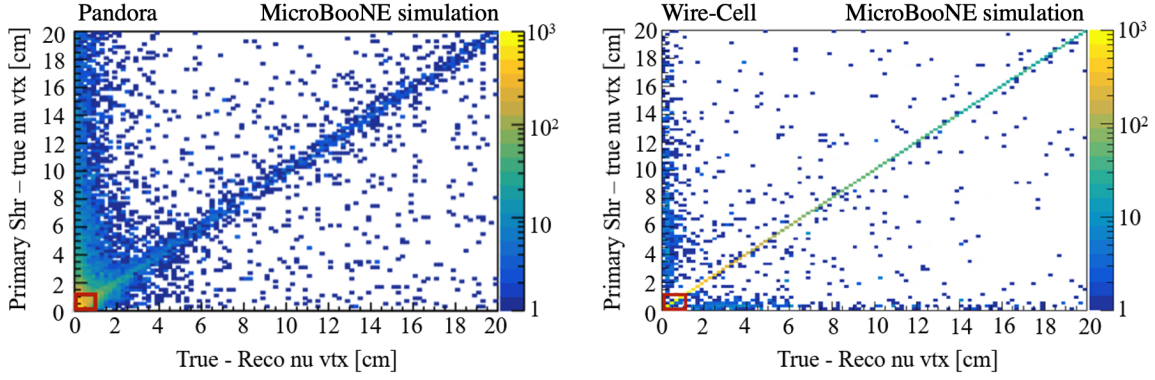


Figure 20. 2D distribution of reconstructed primary shower vertex-true neutrino vertex difference vs. reconstructed neutrino vertex-true neutrino vertex of BNB intrinsic ν_e sample, with reconstructed with Pandora (left) and with Wire-Cell (right). The two 2D histograms contain the same number of events. For Wire-Cell, the distribution is more concentrated along the diagonal line, which shows the reconstructed neutrino vertices and EM shower vertices are better aligned. Red boxes in the lower left corner indicates events with the position difference less than 1 cm. Note that the z-axis is in log scale.

vertex, the lower efficiency number when requiring both a reconstructed primary shower vertex and a reconstructed neutrino vertex within 1 cm of the true neutrino vertex essentially gives a measure of the reconstruction efficiency for the primary EM shower.

In this section, we describe various pattern recognition techniques towards neutrino vertex identification. 65.8% and 72.9% efficiencies have been achieved in the Wire-Cell reconstruction for charged-current ν_e and ν_μ interactions respectively for the MicroBooNE experiment.

3 Particle flow tree reconstruction after neutrino vertex identification

In the first iteration of the Wire-Cell pattern recognition (section 2), we identified the neutrino vertex as well as track segments and EM shower segments. In this section, we describe the reconstruction of complete EM showers and the particle flow tree including π^0 reconstruction.

3.1 Overall electromagnetic shower clustering

As explained in section 1 and in ref. [22], “a TPC cluster”, which consists of TPC activities that are connected, is used to organize (or divide) 3D and 2D data. Within a TPC cluster, track segments are identified as outlined in section 2.1. Such a data structure is not sufficient to describe electromagnetic (EM) showers, which typically consist of many separated TPC clusters. Therefore, once the primary neutrino vertex is identified (section 2.4), a dedicated EM shower clustering algorithm is applied. The clustering algorithm considers directional and distance information. First, the EM shower candidates within the TPC cluster that contains the primary neutrino vertex are examined. Second, along the direction of each EM shower candidate, the separated pieces within a certain distance (80 cm) and a certain angle (15 degrees) are clustered together. Finally, isolated EM shower candidates are examined according to their distances to the primary neutrino vertex and the same EM shower clustering algorithm is performed.

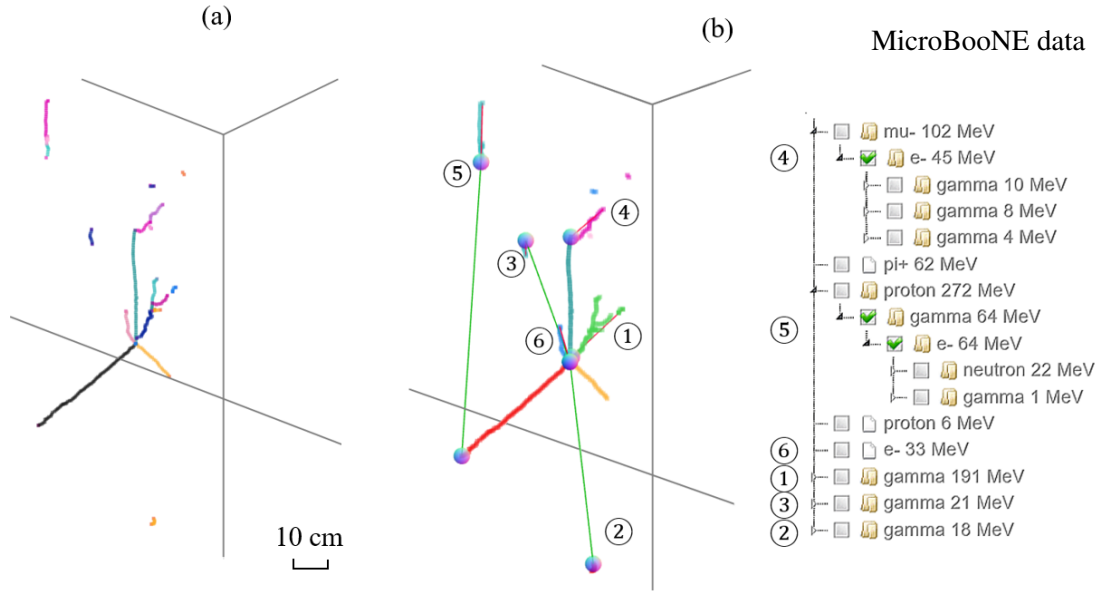


Figure 21. Illustration of EM shower clustering. (a) shows the identified track segments with different colors in different TPC clusters. (b) shows the EM showers after clustering. The green lines illustrate the associations. The red lines connect the start and end points of each EM shower. The numbers of EM showers are also displayed with the particle flow information (Refer to the caption of figure 4 for more details).

Figure 21 shows an example of the EM clustering algorithm. Here, the first, second, third, and fifth EM showers are isolated EM showers. The fourth EM shower is a Michel electron at the end of a stopped muon. The sixth EM shower is connected to the primary neutrino vertex. It is expected to be the result of a low-energy stopped charged pion, which is invisible because of the detector spatial resolution.

3.2 Neutral pion reconstruction

With EM showers defined, one can reconstruct neutral pions which decay to two photons with a $\sim 99\%$ branching ratio. The reconstruction of neutral pions is useful in validating the energy scale for the EM shower energy reconstruction and is crucial for reconstructing the neutrino energy and in selecting events with neutral pions in both charged-current and neutral-current channels, which can be used to constrain backgrounds in single EM shower searches, for example the ν_e CC selection or a single photon selection.

For a π^0 produced in a neutral current interaction (or outside the TPC active volume), there may not be any other TPC information to identify the π^0 vertex. In these cases, the reconstructed primary neutrino vertex is erroneously connected to an EM shower. A dedicated algorithm is therefore implemented to find a π^0 with a displaced vertex. For each EM shower, a PCA is performed to find the EM shower's primary axis (i.e. its direction). Given any two EM showers with calculated primary axes, we find two points, one on each of them. These points are the closest to the opposite primary axis. The midpoint of the pair is then identified and labeled as the displaced vertex. Once the displaced vertex is determined, the direction of each EM shower is redefined with respect to that vertex for the invariant mass calculation. Figure 22a shows such an example.

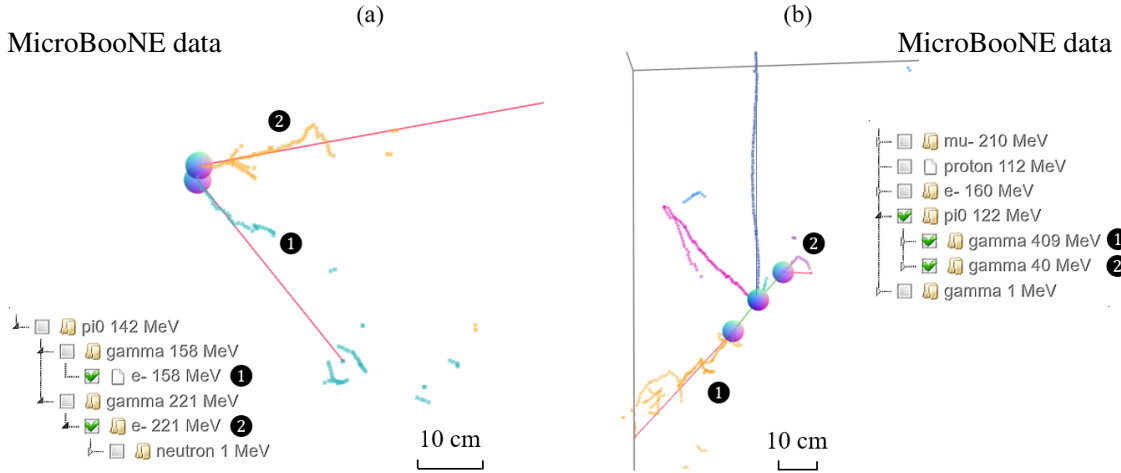


Figure 22. Examples of reconstructed neutral pions. (a) a neutral pion is reconstructed at an invariant mass of $142 \text{ MeV}/c^2$ without any TPC activity at its generation location. (b) a neutral pion is reconstructed at an invariant mass of $122 \text{ MeV}/c^2$ from the primary neutrino vertex. Separate photons are labeled with the corresponding numbers in the particle flow diagram. The color coding in both (a) and (b) are to separate different reconstructed particles similar as used in panel (c) of figure 4. The reconstructed particle flow information is also displayed (Refer to the caption of figure 4 for more details).

For each vertex with two or more associated EM showers, a π^0 is reconstructed with two separate strategies. The first is to find the pair of photons (above a certain energy threshold) that reconstruct as a π^0 with an invariant mass closest to the π^0 mass. The second is to find the pair of photons that have the highest total energy. Both strategies are implemented and applied to all events with two or more photons reconstructed from the same vertex. The first strategy is used to construct the particle flow and is generally good for rejecting backgrounds of $\nu_e \text{CC}$ events, as incorrectly-paired photons would generally result in an incorrect invariant π^0 mass. The second strategy is used to select π^0 events, which is useful in calibrating the electromagnetic energy reconstruction, given that it has no bias in the invariant mass reconstruction. Figure 22 shows two examples of π^0 reconstructions. In figure 22b the π^0 vertex is also associated with other activities. Figure 23 shows the reconstructed π^0 mass in simulation for the fully contained $\nu_\mu \text{CC}$ π^0 selection. Here, the fully contained events are defined to be events with the reconstructed TPC activity fully contained within the fiducial volume (3 cm inside the effective TPC boundary [22], which is the corrected boundary that takes a space charge effect [31, 52] into account). Compared to the earlier MicroBooNE results [53], the resolution of the reconstructed π^0 mass is improved by about 15%.

3.3 Reconstruction performance of primary leptons

The reconstruction quality of the primary leptons (muons and electron EM showers) is essential for both neutrino event flavor tagging and neutrino energy reconstruction. Therefore, we evaluate the reconstruction efficiency and angular resolution. The first step of the evaluation is to define truth-reco matching. We select events with a good reconstructed neutrino vertex (within 1 cm of the truth vertex), to decouple the evaluation of single particle reconstruction

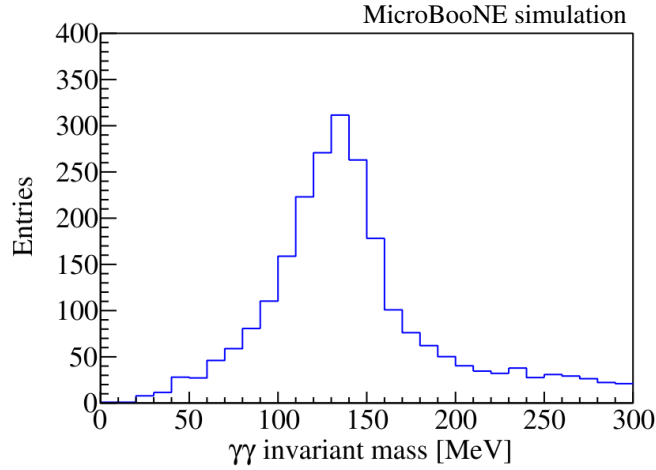


Figure 23. Reconstructed $\gamma\gamma$ invariant mass ($\sqrt{2E_{\gamma 1}E_{\gamma 2} \cdot (1 - \cos \theta_{\gamma\gamma})}$) on simulated fully contained charged-current $\nu_{\mu}\pi^0$ events. The tail at the high reconstructed mass is the result of incorrect association of γ candidates.

and neutrino vertex reconstruction. Then within the selected events, using the simulation truth information, we find all the leading (with largest energy) primary (from neutrino interaction vertex) muons or electrons originating inside a fiducial volume (20 cm from the active volume boundary) as the *truth target*. The fiducial volume requirement ensures that the particles deposit enough visible information for reconstruction in the detector. With the particle flow information reconstructed from the aforementioned pattern recognition algorithms, we find all the leading primary muons or electrons. If those muons or electrons have a reconstructed vertex within 1 cm of the truth counterpart, we count this as a truth-reco matching in the following evaluations in this subsection.

Figure 24 shows the reconstruction efficiencies as a function of the truth particle energy for leading primary muons and electrons. These efficiencies are related to the performances of algorithms introduced in this section as well as algorithms from section 2, i.e., the track segment finding, the trajectory and dQ/dx fitting, the track-shower separation, the PID and the neutrino vertex identification. The denominators are the number of the truth targets aforementioned and the numerators are the number of matched reconstructed counterpart. The muon reconstruction efficiency rises to its plateau at about 85% to 95% above 300 MeV. The electron reconstruction efficiency plateaus at about 90% and slightly decreases above 1.5 GeV. The slight reduction in efficiency at high energy is the result of incorrect reconstruction for isochronous event topologies (close to parallel to the wire planes), where a large amount of ionization electrons arrive at the anode plane at the same time creating ambiguities in the projective wire-plane readout.

Figure 25 shows the resolution of the reconstructed angle as a function of the truth particle energy for leading primary muons and electrons. Here the θ and ϕ refer to polar and azimuthal angles defined with respect to the neutrino beam direction. The truth-reco matching is defined as before. The angular resolutions for both muon and electron reconstruction are at the 0.1 rad (5.7 degrees) level.

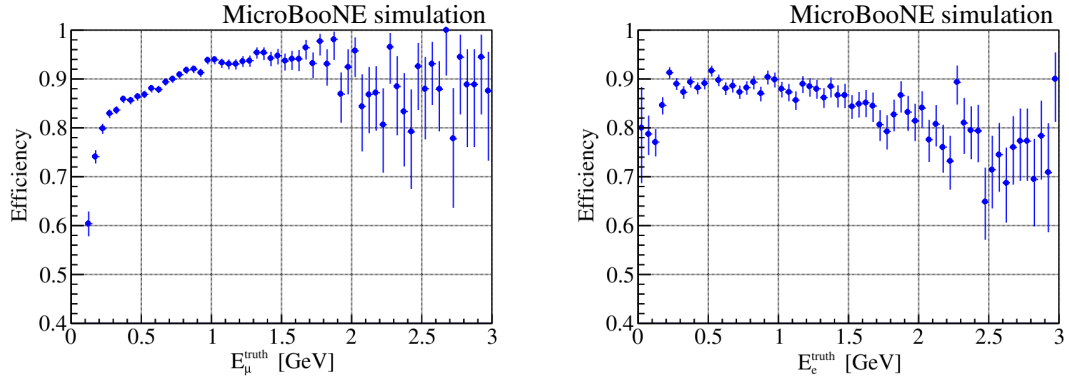


Figure 24. Single particle reconstruction efficiency for the leading primary muon (left) and electron (right). The x-axis is the truth particle energy.

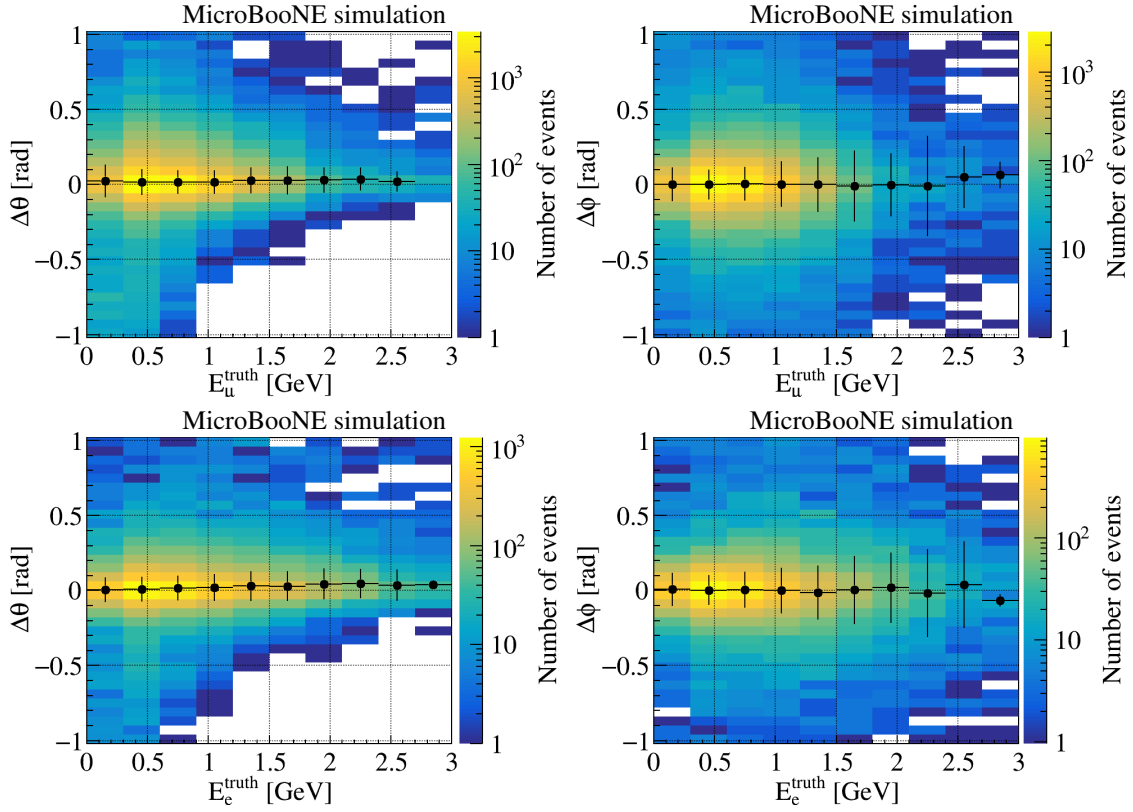


Figure 25. Difference between reconstructed and truth track angle as a function of truth particle energy for leading primary muons (top) and electrons (bottom). The left panels are for polar angles and the right panels are for azimuthal angles. Slice-wise mean and resolution values from Gaussian fittings are overlaid. The slight bias in the polar angle is because of its non-negative nature in its reconstruction.

4 Neutrino energy reconstruction

With a particle flow tree reconstructed from the neutrino vertex, the neutrino energy can be calculated. There are, in general, three methods to compute the deposited energy of a charged track or shower:

- Range: the travel range is used to calculate the energy of track-like particles if they stop inside the detector. The NIST PSTAR database [54] is used to derive the relation between the range and the kinetic energy of different particles with different masses.
- Summation of dE/dx : the (visible) kinetic energy is estimated by summing up dE/dx for each piece (~ 6 mm) of the track. The energy loss per unit length dE/dx is converted from the ionization charge per unit length dQ/dx considering the recombination effect when the ionization electrons are produced. Note that an *effective* recombination model that takes into account the overall normalization difference between MicroBooNE data and simulation charge signals [39] is built by tuning the parameters of the modified box model from ArgoNeuT [55]. This effective recombination model improves the agreement with MicroBooNE data; however, it still underestimates the energy by about 10%.
- Overall charge-energy scaling: while the previous two methods are appropriate for track-like objects, they are not suitable for EM showers because of the difficulties in deriving dQ/dx or range. After multiplying 23.6 eV per ionization pair [56, 57], the energy of EM showers is estimated by applying another overall scaling factor 2.50 ($= 1/0.4$). This factor is derived from a simulation that includes the bias in the reconstructed charge [18] and the average recombination factor (≈ 0.5) of an EM shower in argon. For ν_e CC events, the energy reconstruction of the primary electromagnetic shower gives a 12% energy resolution with about 2% bias, indicating a good performance of charge clustering for EM showers (described in section 3.1).

The three methods are combined in the neutrino energy estimation. For a stopped track longer than 4 cm, range method is used. For short stopped tracks (< 4 cm), the summation of dE/dx method is used, given the range measurement is expected to have a larger uncertainty since track trajectories consist of points separated by 0.6 cm. For a long muon, where delta rays become visible (multiple track segments are part of a reconstructed muon track), we again use the summation of dE/dx method, given the track length of delta rays may bias the range calculation of the long muon. For EM showers, the charge scaling method is used, since track trajectories are not well defined in this case. The neutrino energy is estimated by summing up the kinetic energy of all particles in the reconstructed particle flow tree. For each muon or charged pion or electron, its mass is added to the energy reconstruction. In addition, an average binding energy of 8.6 MeV per nucleon in which case an argon-40 nucleus is completely disassembled is added for each identified proton in the reconstructed particle flow. These protons may be produced at the primary neutrino interaction or secondary interactions (e.g. produced by a neutron).

Figures 26 and 27 summarize the neutrino energy reconstruction performance for fully contained ν_e CC and ν_μ CC interactions. The top left panels show the reconstructed visible energy and the top right panels show the reconstructed neutrino energy. We can see that the reconstructed neutrino energy is closer to the diagonal line, meaning that the bias is largely corrected from the reconstructed visible energy. The bottom panels show that the reconstructed neutrino energy resolution is 10% to 15% for ν_e CC events and 15-20% for ν_μ CC events. At a truth neutrino energy of 800 MeV, the reconstructed neutrino energy resolution is 15% for ν_e CC and 20% for ν_μ CC. The bias of the reconstructed neutrino energy relative to the true neutrino energy is 7% and 10% on average for BNB

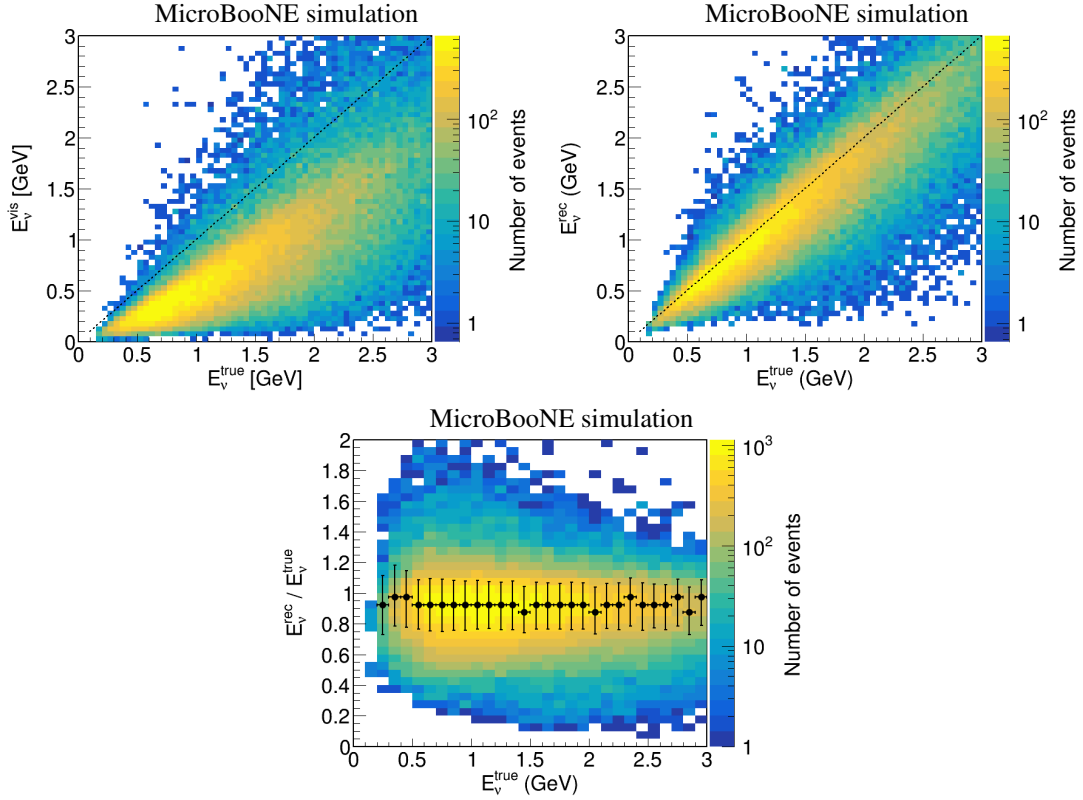


Figure 26. Energy reconstruction for selected fully contained ν_e CC candidates: reconstructed visible energy (top left), reconstructed neutrino energy (top right), and ratio of reconstructed neutrino energy to truth (bottom). In the bottom panel, the peak values and the corresponding resolutions (asymmetric) for each true energy bin are plotted as well.

ν_e CC and ν_μ CC interactions, respectively. The bias in the high energy region, e.g. true neutrino energy greater than 1 GeV, is obvious for the selected ν_μ CC sample. This is because the high energy ν_μ CC interactions after the selection, in particular with the full containment requirement, are more likely 1) to have high energy transfer (difference of incoming neutrino energy and outgoing lepton energy) which results in a sizable amount of invisible energy carried away by final state hadrons especially neutrons or other neutral particles; and 2) to have high energy muons (long muons with significant delta-rays), as discussed above, the reconstructed energies of which have a relatively large bias because of the usage of an imperfect calorimetric reconstruction method instead of the range method.

In real data analysis [23, 24], the various methods used to reconstruct neutrino energy are validated. The summation of dE/dx method is validated by comparing the dQ/dx vs. residual range for the stopped muons and protons between data and MC. The energy reconstruction of EM showers is validated by comparing the reconstructed π^0 mass between data and MC. The reconstructed hadronic energy has also been validated by comparing the lepton-kinematics-constrained MC prediction and real data measurement, in which case the systematic effect, in particular from the neutrino-argon interaction modeling, is largely reduced.

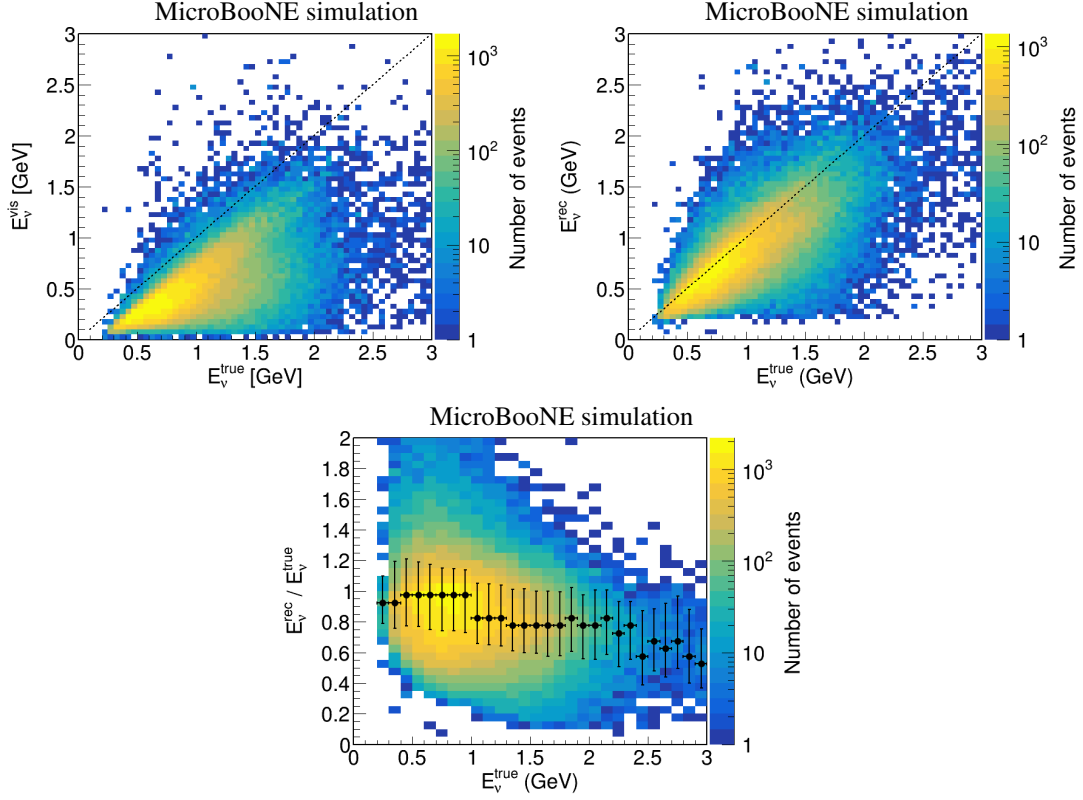


Figure 27. Energy reconstruction for selected fully contained ν_μ CC candidates: reconstructed visible energy (top left), reconstructed neutrino energy (top right), and ratio of reconstructed neutrino energy to truth (bottom). In the bottom panel, the peak values and the corresponding resolutions (asymmetric) for each true energy bin are plotted as well.

5 Summary

To summarize, we describe a set of pattern recognition algorithms to reconstruct neutrino events based on the Wire-Cell reconstruction paradigm. These algorithms perform well on the tasks of neutrino vertex identification, single particle reconstruction, and neutrino energy reconstruction, which are essential to achieve high-performance inclusive ν_e CC and ν_μ CC event selections [23, 24]. Evaluated using the current state-of-the-art MicroBooNE MC simulation, we achieve 65.8% and 72.9% neutrino vertex identification efficiencies for ν_e and ν_μ charged-current interactions, respectively. These neutrino vertex identification efficiencies are comparable to those achieved with the Pandora reconstruction paradigm [50]. In addition, 80–90% single particle reconstruction efficiencies for primary leptons and 15–20% reconstructed neutrino energy resolutions are achieved for these charged-current neutrino interactions. The achievements reported in this paper build a solid foundation for downstream physics analyses, such as the search for new physics with electron neutrinos [23] and measurement of charged-current muon neutrino interaction cross section on Argon [24]. Regarding the performance comparison between the Wire-Cell reconstruction and other reconstruction paradigms in MicroBooNE, more information can be found in ref. [58], which summarizes the searches for an excess of electron neutrino interactions in MicroBooNE

using multiple final state topologies and multiple event reconstruction paradigms. In addition to the search in the more demanding inclusive ν_e CC channel using Wire-Cell [23], searches in exclusive quasi-elastic ν_e CC channel [59] and semi-inclusive ν_e CC channel without pions in the final state [51] are performed with the Deep-Learning based reconstruction [43, 60, 61] and the Pandora reconstruction [50], respectively. The current Wire-Cell pattern recognition relies on many traditional (or non-Deep-Learning) algorithms, in order to mitigate the risk of amplifying any discrepancies between simulation and real data. As the simulation for LArTPCs becomes more mature, we are expecting to adopt more Deep-Learning based algorithms in the Wire-Cell reconstruction paradigm to further enhance the overall performance.

Acknowledgments

This document was prepared by the MicroBooNE collaboration using the resources of the Fermi National Accelerator Laboratory (Fermilab), a U.S. Department of Energy, Office of Science, HEP User Facility. Fermilab is managed by Fermi Research Alliance, LLC (FRA), acting under Contract No. DE-AC02-07CH11359. MicroBooNE is supported by the following: the U.S. Department of Energy, Office of Science, Offices of High Energy Physics and Nuclear Physics; the U.S. National Science Foundation; the Swiss National Science Foundation; the Science and Technology Facilities Council (STFC), part of the United Kingdom Research and Innovation; the Royal Society (United Kingdom); and The European Union’s Horizon 2020 Marie Skłodowska-Curie Actions. Additional support for the laser calibration system and cosmic ray tagger was provided by the Albert Einstein Center for Fundamental Physics, Bern, Switzerland. We also acknowledge the contributions of technical and scientific staff to the design, construction, and operation of the MicroBooNE detector as well as the contributions of past collaborators to the development of MicroBooNE analyses, without whom this work would not have been possible.

References

- [1] C. Rubbia, *The liquid argon time projection chamber: A new concept for neutrino detector*, [CERN-EP-INT-77-08](#), [CERN-EP-77-08](#), CERN, Geneva (1977).
- [2] H.H. Chen, P.E. Condon, B.C. Barish and F.J. Sciulli, *A Neutrino detector sensitive to rare processes. I. A Study of neutrino electron reactions*, FERMILAB-PROPOSAL-0496 (1976).
- [3] W.J. Willis and V. Radeka, *Liquid Argon Ionization Chambers as Total Absorption Detectors*, [Nucl. Instrum. Meth.](#) **120** (1974) 221.
- [4] D.R. Nygren, *The Time Projection Chamber: A New 4 pi Detector for Charged Particles*, [eConf C 740805](#) (1974) 58.
- [5] ICARUS collaboration, *Design, construction and tests of the ICARUS T600 detector*, [Nucl. Instrum. Meth. A](#) **527** (2004) 329.
- [6] ARGO-NEUT collaboration, *First Measurements of Inclusive Muon Neutrino Charged Current Differential Cross Sections on Argon*, [Phys. Rev. Lett.](#) **108** (2012) 161802 [[arXiv:1111.0103](#)].
- [7] MicroBooNE collaboration, *Design and Construction of the MicroBooNE Detector*, [2017 JINST](#) **12** P02017 [[arXiv:1612.05824](#)].

- [8] LArIAT collaboration, *LArIAT: Liquid Argon In A Testbeam*, [arXiv:1406.5560](#).
- [9] CAPTAIN collaboration, *The CAPTAIN Detector and Physics Program*, in proceedings of in *Community Summer Study 2013: Snowmass on the Mississippi*, Minneapolis, MN, U.S.A., 29 July–6 August 2013 [[arXiv:1309.1740](#)].
- [10] MINIBOOONE collaboration, *A Combined $\nu_\mu \rightarrow \nu_e$ and $\bar{\nu}_\mu \rightarrow \bar{\nu}_e$ Oscillation Analysis of the MiniBooNE Excesses*, 2012 [[arXiv:1207.4809](#)].
- [11] MICROBOONE, LAr1-ND, ICARUS-WA104 collaboration, *A Proposal for a Three Detector Short-Baseline Neutrino Oscillation Program in the Fermilab Booster Neutrino Beam*, [arXiv:1503.01520](#).
- [12] DUNE collaboration, *Deep Underground Neutrino Experiment (DUNE), Far Detector Technical Design Report, Volume I Introduction to DUNE*, 2020 *JINST* **15** T08008 [[arXiv:2002.02967](#)].
- [13] V. Radeka et al., *Cold electronics for ‘Giant’ Liquid Argon Time Projection Chambers*, *J. Phys. Conf. Ser.* **308** (2011) 012021.
- [14] MICROBOONE collaboration, *Noise Characterization and Filtering in the MicroBooNE Liquid Argon TPC*, 2017 *JINST* **12** P08003 [[arXiv:1705.07341](#)].
- [15] MICROBOONE collaboration, *Ionization electron signal processing in single phase LArTPCs. Part I. Algorithm Description and quantitative evaluation with MicroBooNE simulation*, 2018 *JINST* **13** P07006 [[arXiv:1802.08709](#)].
- [16] MICROBOONE collaboration, *Ionization electron signal processing in single phase LArTPCs. Part II. Data/simulation comparison and performance in MicroBooNE*, 2018 *JINST* **13** P07007 [[arXiv:1804.02583](#)].
- [17] X. Qian, C. Zhang, B. Viren and M. Diwan, *Three-dimensional Imaging for Large LArTPCs*, 2018 *JINST* **13** P05032 [[arXiv:1803.04850](#)].
- [18] MICROBOONE collaboration, *Neutrino event selection in the MicroBooNE liquid argon time projection chamber using Wire-Cell 3D imaging, clustering, and charge-light matching*, 2021 *JINST* **16** P06043 [[arXiv:2011.01375](#)].
- [19] E.J. Candès, J.K. Romberg and T. Tao, *Stable signal recovery from incomplete and inaccurate measurements*, *Commun. Pure Appl. Math.* **59** (2006) 1207 [[arXiv:math/0503066](#)].
- [20] J. Friedman, T. Hastie and R. Tibshirani, *Regularization paths for generalized linear models via coordinate descent*, *J. Stat. Softw.* **33** (2010) 1.
- [21] MICROBOONE collaboration, *High-performance Generic Neutrino Detection in a LArTPC near the Earth’s Surface with the MicroBooNE Detector*, [arXiv:2012.07928](#).
- [22] MICROBOONE collaboration, *Cosmic Ray Background Rejection with Wire-Cell LArTPC Event Reconstruction in the MicroBooNE Detector*, *Phys. Rev. Applied* **15** (2021) 064071 [[arXiv:2101.05076](#)].
- [23] MICROBOONE collaboration, *Search for an anomalous excess of inclusive charged-current ν_e interactions in the MicroBooNE experiment using Wire-Cell reconstruction*, [arXiv:2110.13978](#).
- [24] MICROBOONE collaboration, *First Measurement of Energy-dependent Inclusive Muon Neutrino Charged-Current Cross Sections on Argon with the MicroBooNE Detector*, [arXiv:2110.14023](#).
- [25] MINIBOOONE collaboration, *The Neutrino Flux prediction at MiniBooNE*, *Phys. Rev. D* **79** (2009) 072002 [[arXiv:0806.1449](#)].
- [26] GENIE collaboration, *Recent highlights from GENIE v3*, [arXiv:2106.09381](#).

- [27] MicroBooNE collaboration, *New Theory-driven GENIE Tune for MicroBooNE*, [arXiv:2110.14028](#).
- [28] GEANT4 collaboration, *GEANT4—a simulation toolkit*, *Nucl. Instrum. Meth. A* **506** (2003) 250.
- [29] M. Jaskolski and M. Wojcik, *Electron Recombination in Ionized Liquid Argon: A Computational Approach Based on Realistic Models of Electron Transport and Reactions*, *J. Phys. Chem. A* **115** (2011) 4317.
- [30] ARGO-NeUT collaboration, *A Study of Electron Recombination Using Highly Ionizing Particles in the ArgoNeuT Liquid Argon TPC*, *2013 JINST* **8** P08005 [[arXiv:1306.1712](#)].
- [31] MicroBooNE collaboration, *A method to determine the electric field of liquid argon time projection chambers using a UV laser system and its application in MicroBooNE*, *2020 JINST* **15** P07010 [[arXiv:1910.01430](#)].
- [32] MicroBooNE collaboration, *Measurement of space charge effects in the MicroBooNE LArTPC using cosmic muons*, *2020 JINST* **15** P12037 [[arXiv:2008.09765](#)].
- [33] Y. Li et al., *Measurement of Longitudinal Electron Diffusion in Liquid Argon*, *Nucl. Instrum. Meth. A* **816** (2016) 160 [[arXiv:1508.07059](#)].
- [34] M. Antonello et al., *Precise 3D track reconstruction algorithm for the ICARUS T600 liquid argon time projection chamber detector*, *Adv. High Energy Phys.* **2013** (2013) 260820 [[arXiv:1210.5089](#)].
- [35] Steiner tree greedy algorithm, <http://paal.mimuw.edu.pl/docs/index.html>, 2019.
- [36] Biconjugate gradient stabilized method (BiCGSTAB), https://eigen.tuxfamily.org/dox/classEigen_1_1BiCGSTAB.html, 2019.
- [37] Intensity Frontier at Brookhaven National Lab., *Wire Cell PID*, <https://github.com/BNLIF/wire-cell-pid/>.
- [38] MicroBooNE collaboration, *Determination of muon momentum in the MicroBooNE LArTPC using an improved model of multiple Coulomb scattering*, *2017 JINST* **12** P10010 [[arXiv:1703.06187](#)].
- [39] MicroBooNE collaboration, *Calibration of the charge and energy loss per unit length of the MicroBooNE liquid argon time projection chamber using muons and protons*, *2020 JINST* **15** P03022 [[arXiv:1907.11736](#)].
- [40] J. Brownlee, *A Gentle Introduction to Object Recognition With Deep Learning*, <https://machinelearningmastery.com/object-recognition-with-deep-learning/>.
- [41] B. Graham, M. Engelcke and L. van der Maaten, *3d semantic segmentation with submanifold sparse convolutional networks*, [arXiv:1711.10275](#), 2018.
- [42] B. Graham and L. van der Maaten, *Submanifold sparse convolutional networks*, [arXiv:1706.01307](#), 2017.
- [43] MicroBooNE collaboration, *Semantic segmentation with a sparse convolutional neural network for event reconstruction in MicroBooNE*, *Phys. Rev. D* **103** (2021) 052012 [[arXiv:2012.08513](#)].
- [44] Intensity Frontier at Brookhaven National Lab., *Wire-Cell Prototype*, <https://github.com/BNLIF/wire-cell>.
- [45] O. Ronneberger, P. Fischer and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, [arXiv:1505.04597](#), 2015.
- [46] H. Yu, *Wire-Cell Deep Learning Vertexing for MicroBooNE*, <https://github.com/HaiwangYu/uboone-dl-vtx>.

- [47] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei and Y. Sheikh, *Openpose: Realtime multi-person 2d pose estimation using part affinity fields*, [arXiv:1812.08008](#), 2018
- [48] S. Raschka and V. Mirjalili, *Python Machine Learning*, third edition, Packt (2019).
- [49] D.P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, [arXiv:1412.6980](#).
- [50] MICROBooNE collaboration, *The Pandora multi-algorithm approach to automated pattern recognition of cosmic-ray muon and neutrino events in the MicroBooNE detector*, *Eur. Phys. J. C* **78** (2018) 82 [[arXiv:1708.03135](#)].
- [51] MICROBooNE collaboration, *Search for an anomalous excess of charged-current ν_e interactions without pions in the final state with the MicroBooNE experiment*, [arXiv:2110.14065](#).
- [52] M. Mooney, *The MicroBooNE Experiment and the Impact of Space Charge Effects*, in *Meeting of the APS Division of Particles and Fields*, 11, 2015 [[arXiv:1511.01563](#)].
- [53] MICROBooNE collaboration, *Reconstruction and Measurement of $O(100)$ MeV Energy Electromagnetic Activity from $\pi^0 \rightarrow \gamma\gamma$ Decays in the MicroBooNE LArTPC*, *2020 JINST* **15** P02007 [[arXiv:1910.02166](#)].
- [54] PSTAR at NIST, <https://physics.nist.gov/PhysRefData/Star/Text/PSTAR.html>.
- [55] ARGO-NeUT collaboration, *A Study of Electron Recombination Using Highly Ionizing Particles in the ArgoNeUT Liquid Argon TPC*, *2013 JINST* **8** P08005 [[arXiv:1306.1712](#)].
- [56] E. Shibamura, A. Hitachi, T. Doke, T. Takahashi, S. Kubota and M. Miyajima, *Drift velocities of electrons, saturation characteristics of ionization and W-values for conversion electrons in liquid argon, liquid argon-gas mixtures and liquid xenon*, *Nucl. Instrum. Meth.* **131** (1975) 249.
- [57] M. Miyajima, T. Takahashi, S. Konno, T. Hamada, S. Kubota, H. Shibamura et al., *Average energy expended per ion pair in liquid argon*, *Phys. Rev. A* **9** (1974) 1438.
- [58] MICROBooNE collaboration, *Search for an Excess of Electron Neutrino Interactions in MicroBooNE Using Multiple Final State Topologies*, [arXiv:2110.14054](#).
- [59] MICROBooNE collaboration, *Search for an anomalous excess of charged-current quasi-elastic ν_e interactions with the MicroBooNE experiment using Deep-Learning-based reconstruction*, [arXiv:2110.14080](#).
- [60] MICROBooNE collaboration, *Deep neural network for pixel-level electromagnetic particle identification in the MicroBooNE liquid argon time projection chamber*, *Phys. Rev. D* **99** (2019) 092001 [[arXiv:1808.07269](#)].
- [61] MICROBooNE collaboration, *Convolutional neural network for multiple particle identification in the MicroBooNE liquid argon time projection chamber*, *Phys. Rev. D* **103** (2021) 092003 [[arXiv:2010.08653](#)].