Low energy muon neutrino reconstruction in MicroBooNE

Adrien Hourlier, on behalf of the MicroBooNE collaboration
Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA 02139, USA
E-mail: hourlier@mit.edu

Abstract. MicroBooNE is a Liquid Argon Time Projection Chamber (LArTPC) neutrino experiment on the Booster Neutrino Beamline at the Fermi National Accelerator Laboratory, with an 85-tonne active mass. One of MicroBooNE’s primary physics goals is to investigate the excess of electron neutrino events seen by MiniBooNE in the [200-600] MeV range. MicroBooNE will constrain the intrinsic electron neutrino component of the beam by measuring the muon neutrino spectrum. Several low-energy excess analyses are taking place in parallel, using independent reconstructions and selection schemes. This paper will focus on a low-energy excess analysis that makes use of deep learning algorithms applied to the high-resolution images provided by the MicroBooNE LArTPC. We present a novel 3D event reconstruction based on computer vision tools and a stochastic search algorithm, yielding a 2.2% energy resolution for 1µ1p muon neutrino interactions in the [200-1500] MeV range.

1. Introduction

The MicroBooNE experiment, currently taking data in the BNB neutrino beam at Fermilab since October 2015, 450 m downstream of the target, is a liquid argon time projection chamber (LArTPC) [1]. The detector cryostat has a total capacity of 170 tons of liquid argon, with an active region of (2.6×2.3×10.4) m³. The system comprises two major sub-detectors: a time projection chamber (TPC) for tracking, and a light collection system for trigger and reconstruction of the precise interaction time. The detector has been described in detail in Ref. [2].

Events which are contained within the detector, and which are consistent with the signature of one muon and one proton, are of interest to a number of physics studies on MicroBooNE. These are primarily measurements of exclusive cross sections, including charged-current quasi-elastic (CCQE). This group of events is also important for charged current π⁰ events (CCπ⁰) and the investigation of the MiniBooNE low energy excess [3] in MicroBooNE. While the signal for those events is hypothesized to consist of an electron and at least one proton, the normalization sample consists of a muon and at least one proton. In this case, the reconstruction is identifying candidate events that may be selected as “1µ1p” (one muon and one proton) events after particle ID. To obtain a large sample of contained 1µ1p events, the MicroBooNE collaboration has developed a specialized reconstruction package specialized for contained two-track events. The purpose of this reconstruction is to identify and reconstruct those events with two tracks emanating from a vertex that are both longer than a specified length; any number of shorter tracks may be attached to the same vertex.
Figure 1. A simulated $\nu_\mu$ event shown in the three wire planes that illustrates the features of interest for this reconstruction package. The muon neutrino MC truth energy is 483 MeV. A single proton (deposited energy 266 MeV) and muon (deposited energy 73 MeV) are produced. The image-like nature of the drift-time versus wire-plane plots is apparent, and explains why we use this language. Each two-dimensional bin on these plots is called a “pixel” and, in this case, contains the information on the ADC count. See text for explanation.

The physics analyses that will make use of this code package employ both the TPC and light collection subsystems. However the three-dimensional reconstruction code described here uses only the TPC information, and so we describe only this subsystem. In the TPC, electrons from the ionization tracks produced by charged particles in the interaction with liquid argon, due to a 273 V/cm electron field, drift towards three sense wire planes that provide the charge read-out. The signals from the three wire planes form three views, U (wires at +60 degrees from vertical), V (-60 degrees) and Y (0 degrees). The U and V planes detect signals via induction, while the Y view is the collection plane. The wire spacing in each plane is 0.3 cm. The wire waveforms are read out with a sampling time of 0.5 $\mu$s, and with a shaping time of the ASICs of 2 $\mu$s. This results in highly detailed event information that we exploit by treating the time versus wire hit plots from each of the three planes as images with pixels, as described below. This use of high resolution images allows the analysis chain to make use of deep learning algorithms. In this note, we present an algorithm-based reconstruction approach leveraging the output of a pre-processing performed using a deep neural network.

Our reconstruction package is discussed within the context of MicroBooNE analyses. However, the approaches are generic to LArTPC detectors that run in a $\sim$ 1 GeV neutrino beam. Future examples of such experiments are SBND and ICARUS, which will run in the same BNB neutrino line as MicroBooNE in the near future [4]. The approach is also appropriate to reconstructing atmospheric neutrino events in the DUNE far detector [5], although in this case, complications due to cosmic rays will be substantially reduced compared to the surface-based detectors on the BNB beam-line.

We will present results based on simulated events at energies relevant to the MicroBooNE beam and using MicroBooNE’s simulation package. MicroBooNE uses GENIE [6] to simulate neutrino interactions. The simulated particles are fed to a GEANT4 simulation [7, 8, 9] of the detector. Frames of un-biased off-beam readouts, containing only cosmic rays, are overlaid on top of the simulated event displays containing the tracks from the neutrino interactions. This ensures an accurate representation of the cosmic ray background in our simulated neutrino events.
2. Data Pre-processing
An essential and difficult problem to solve in reconstruction is identification and removal of cosmic rays from the events. With no overburden and given the 2.3 millisecond readout window, MicroBooNE averages 12 cosmic rays per readout period. The high rate of cosmic rays, in conjunction with un-responsive wires can cause cosmic rays to look like neutrino events. To address this issue, prior to three-dimensional reconstruction, an algorithm is applied to tag pixels corresponding to cosmic rays. This code will be described in a future separate paper, and so is only briefly described here. Cosmic rays are identified by the boundary-crossings at the edges of the active region. Through-going cosmic rays will cross two boundaries, while stopping cosmic rays will cross only one boundary. Exiting muons from charged current neutrino events also cross one boundary, but this is not an issue, since the analyses for which this code is used employ only contained events. The cosmic ray tagging algorithm starts at the boundary and works inward, into the detector, labeling the consecutive charge. Once all charge that is connected to a boundary is identified, the remaining untagged charge clusters on each plane are included in a 3D-consistent volume called “contained regions of interest” (cROIs). These cROIs are then fed into the three-dimensional reconstruction code. Typically about 10 cROIs are found per event. Often tagged-cosmic charge will also appear within a cROI.

3. Using Images in the Reconstruction Package
This reconstruction package makes use of MicroBooNE data and Monte Carlo treated as “images.” By this, we mean that the TPC data are represented on a 2-dimensional plot, with wire number along the x axis and drift time along the y axis. The choice to analyze the detector in an image-format allows the use of widespread and very powerful computer vision tools such as “OpenCV” (Open source Computer Vision) [10], a widely used application for cutting-edge pattern recognition. Using images also allows for the implementation of deep learning algorithms at two points in the analysis. The first is the Semantic Segmentation Network (SSNet), which precedes this reconstruction package. The second is a convolution-neural-network-based particle identification, which follows this reconstruction package.

The reconstruction package will primarily make use of two kinds of images. The first is the “ADC-image,” which contains information on the charge in each pixel. The second is the “SSNet-image” which contains information on whether pixels representing connected chains of charge are track-like, shower-like or neither.

3.1. ADC-images
In the case of the ADC-image, the intensity of each “pixel” is determined by summing the amplitude of the noise-filtered, deconvolved signal [11] from 6 time ticks while keeping wires individually. This choice comes from the fact that, at 0.5 microseconds per tick, six time ticks is 3 microseconds. With the current drift velocity of the electrons of \( (0.11 \pm 0.01) \text{cm} \mu \text{s}^{-1} \), this corresponds to about 0.33 cm, which is similar to the detector’s 0.3 cm wire pitch. There is an image for each cROI in each of the wire-plane views.

Figure 1 shows an example \( \nu_\mu \) CCQE event of interest for this reconstruction package. The ADC-Images are made in each plane, as shown. This event has true neutrino energy of \( E_{\nu}^{\text{true}} = 483 \text{MeV} \), the kinematic energies of the emitted muon and proton are respectively 73 MeV and 266 MeV, which is typical of the kinematics that we aim to reconstruct with this package.

3.2. SSNet images
The cROIs are then fed into a deep-learning algorithm called Semantic Segmentation Network (SSNet) [13, 14, 15]. Typically, the SSNet will classify muons, charged-pion and protons as track-like, and electrons and photons as shower-like.
Figure 2. Two examples illustrating the SSNet-image pixel labeling. The left panel shows a $1e1p$ type event from a $\nu_e$ interaction. The pixels corresponding to the proton are here correctly classified as track by the SSNet. The pixels corresponding to the electron are here mostly classified as shower by the SSNet, except for a small portion mistakenly labeled as track. The right panel shows a $1\mu1p$ type event from a $\nu_\mu$ interaction. Here both the proton and muon tracks are correctly labeled as track pixels. The muon decays into a Michel electron, classified as shower pixels. Dark blue pixels correspond to empty pixels, without charge deposition.

The SSNet-images, which are also constructed for each cROI and for each wire plane view, are created by feeding the ADC-images to a SSNet. The SSNet identifies the pixels based on their surroundings into three categories:

- track pixels
- shower pixels
- background pixels

Two examples of SSNet outputs are shown in Fig. 2. The left panel shows a view of $1e1p$ $\nu_e$ interaction in the $Y$ plane, of a 600 MeV neutrino producing a 279 MeV proton and a 280 MeV electron. The proton track is correctly classified as containing only track-like pixels (in yellow), and the electron shower is mostly classified as shower pixels (in light blue). A small fraction of the shower pixels are mistakenly labeled as track-like. Background pixels, corresponding to pixels without charge deposition are shown in dark blue. The right panel shows a view in the $Y$ plane of a $1\mu1p$ $\nu_\mu$ interaction of a 936 MeV $\nu_\mu$ producing a 397 MeV proton and a 395 MeV muon, that decays into a Michel electron. The Michel electron is here classified as shower pixels, while the proton and muon tracks are correctly labeled as track-like.

In this note, we will focus on the $1\mu1p$ topology, therefore we will be looking for two-track vertices, neglecting the track-shower interface vertices.

4. 3D Vertex Finding

This reconstruction step finds the 3D vertex and then clusters pixels belonging to individual particles. In this algorithm, the pixels tagged as cosmic rays are removed from the images. This code makes use of a custom OpenCV package called Geo2D. This package has convenient tools for 2D geometrical analysis to supplement and extend OpenCV built in data types.
This step makes use of only the track-identified pixels (the track image) to reconstruct a 3D vertex. The algorithm searches for a coincident “vee” shape feature as shown in Figure 2, across the three wire planes which could indicate the presence of a $\mu\nu\nu$ interaction. The algorithm begins by identifying, per plane, a collection of vertex “seeds”. Vertex seeds are pixel locations in the image where a likely vertex may be present, for example at the location where two tracks meet at a kink point. The algorithm identifies vertex seeds by breaking down continuous sets of track clusters into smaller clusters which contain straight segments of charge.

First, a distinction between pixels in the low charge (LC) and high charge (HC) regime, materialized by the red and blue contours in Figure 3, left, is performed. The division between LC and HC ADC count is a constant threshold per plane, corresponding to 10% of the average pixel value for a proton track on each plane. Once the pixel ranges are separated, the algorithm finds groups of LC and HC pixels by applying the OpenCV contour finder.

Next the algorithm breaks down LC and HC clusters into linear sub-clusters. For example, the blue LC contour shown in Figure 3 has an obvious bend or “kink” in it. For each cluster, the algorithm computes the “convex hull” which the smallest convex polygon which bounds the original cluster. Figure 3, left image, shows an example of convex hull (purple polygon). The algorithm identifies the sides of the convex hull which are far away from their corresponding sides on the contour. The point on the contour that is farthest away from the corresponding hull side is called the “defect point”, and is a location where the cluster is potentially bending and changing direction. If the convex hull side is far enough away (5 pixels) from the defect point, the contour is then broken into two at this point. The right image shows the three clusters obtained after this stage, 1 HC cluster (red) and 2 LC clusters (green and blue). The algorithm then iteratively breaks down all clusters into linear segments until no defects point remain.

The collection of defect points are the first set of vertex seeds.

The second set of vertex seeds is found using a Principal Component Analysis (PCA) procedure which fits the clusters to a straight line hypothesis. A PCA is calculated for each broken cluster separately. The algorithm then computes the intersection of all possible PCA lines on the plane. If the lines intersect near a location on the image with charge, the point is saved and is added to the set of vertex seeds. Intersection points far from any charge are ignored. The right image in Figure 3 shows the three PCAs found in the event example.

**Figure 3.** The convex hull (purple) is computed for the LC contour (blue). A defect is found on one of the convex hull edges. Left : A defect point (green) is found on the LC contour, more than 5 pixels away from the corresponding convex hull edge. The defect point indicates the location where the cluster is bending. Right : Once all clusters have been broken down in linear cluster, they are fitted by straight lines, the intersections of which are potential vertex seeds.
Figure 4. Vertex resolution for reconstructed $1\mu 1p$ events. The 3D distance between the Monte Carlo neutrino vertex and the reconstruction vertex is shown. 68% of the reconstructed vertices are found within $0.73^{+0.03}_{-0.02}$ cm of the true vertex position. The true neutrino vertex position is here corrected for displacements due to space-charge effects.

Each of these 2D points are considered a vertex seed and serves as a starting points for 3D vertex search.

The algorithm makes use of the fact that a correct vertex will appear near the same time tick in each view to reduce the seed sample to the time-coincident ones. The X position of these candidates can be determined by using the trigger time and the known drift speed to match the time tick to a X position. The Y and Z positions can then be determined by using wire coincidence between two or three planes.

4.1. Vertex Resolution

The quality of the track image vertex-finding can be assessed using MC by considering $\Delta R$, the distance between the simulated true neutrino vertex to the reconstructed vertex. The $\Delta R$ distribution is shown in Figure 4, unity normalized. Electric field inhomogeneities throughout the detector volume can cause distortion of tracks and displacement of the apparent vertex location. This effect is known as space-charge effect. To compensate for this, and in order to estimate the resolution of the vertex reconstruction, we correct for the space charge distortion introduced in the simulation before estimating $\Delta R$. 68% of well reconstructed events have their vertex found within $0.73 \pm 0.03$ cm of the true vertex position.

5. 3D Track Reconstruction Algorithm

5.1. 3D track finding

The reconstruction of 3D tracks is required to obtain the full kinematic reconstruction of the neutrino interaction. Track reconstruction is particularly sensitive to the quality of the image on a large scale, and to data-Monte Carlo differences. An interruption of the charge deposition along the track, due to dead or noisy wires, waveform deconvolution artifacts, etc., may lead to a wrong reconstructed length and ultimately to the reconstruction of an unphysical energy.

The reconstruction of a track finds a set of 3D points that belong to a given track by performing a stochastic search in the direct neighborhood of previously found 3D points, starting with the vertex position found in the previous section. Points are added to the track if they project on the three views either on non-zero pixels or on pixels corresponding to known unresponsive wires. At least two projections on non-zero pixels must be found. A regularization is then performed to find a minimal set of ordered 3D points that describe the track at the required spatial resolution. Finally, observables such as length, local and average charge deposition, and angles can be estimated.
5.2. Finding the other tracks
These operations are then iterated as long as a new track is found. To prevent the algorithm from finding the same track multiple times, the pixels corresponding to a found track are masked in the ADC image.

Once no new track is found, we iterate the process to the end points of the tracks already found. Indeed, the end points were selected as the point the furthest away from the vertex, but in some cases, if multiple scattering cases the track to curl up, the actual end of the track is not the furthest point. Starting at the end of a found track and looking for a missing portion of track helps reducing these cases. The two portions of the same tracks are then put together in a single new track.

5.3. Self-diagnostic
Once all the tracks associated with a vertex have been found, it is important to recognize and possibly reject cases where the reconstruction failed. This cross-check relies on a set of random points thrown on a spherical shell of radius 3 cm at the end point of each track. Only the forward going points with a solid angle of 65° are kept. For each track, the fraction of points that project on pixels corresponding to dead wires, empty pixels and pixels with charge deposited is evaluated, and a label is attributed to the end point on each plane. This stage is aimed at identifying and rejecting obvious reconstruction failures, and keep tracks that end at the end of the actual charge deposition, ensuring the completeness of the reconstruction.

6. Observable estimations and performance evaluation
In the rest of the paper, a well reconstructed vertex is a vertex that satisfies all these conditions: a vertex with exactly two tracks of more than 5 cm that both ends at the end of the charge deposition.

From the reconstructed 3D-path of each particle exiting a vertex, several key observables can be estimated. We will characterize the results using a $^{1}\mu^{1}p$ MC sample. In this sample, all events are generated with exactly one proton above 60 MeV and exactly one lepton above 35 MeV. A fiducial volume selection of 10 cm is applied on the vertex position, and a containment criterion applied to the lepton only. In addition to the neutrino interaction, a cosmic background from off-beam data is overlaid onto the images.

6.1. Local ionization
For each 3D point, the local ionization for a given plane is estimated by integrating the values of non-zero pixels in a 2 pixel radius around the projection of the 3D point on that plane. The values measured on the three planes can then be used individually, or summed across planes. A scale factor $3/N$ is applied where $N$ is the number of planes on which a non-zero value was found. This scale factor allows to correct for a possible plane in which the 3D point projects onto an un-responsive region.

The average pixel intensity reconstructed for $^{1}\mu^{1}p$ simulated $\nu_{\mu}$ events in MicroBooNE is shown in Figure 5. At this stage, no particle identification has been performed, the blue and red populations have been separated by identifying the muon as the track with the lowest average ionization within the pair of reconstructed tracks (blue distribution) and identifying the proton as the track with the highest average ionization (red distribution). All vertices with two reconstructed tracks will have tracks identified as muon or proton with that method. A more definitive particle identification will be performed later on in the analysis chain.
Figure 5. Average pixel intensity along each reconstructed track. The red and blue distributions represent respectively the tracks with the highest and lowest average ionization in a given vertex.

Figure 6. Relative difference between the energy reconstructed for the interaction \(E_{\nu}^{\text{range}}\) and the true length-based energy from the simulation \(E_{\nu}^{\text{true}}\). A fit by a Gaussian function allows to obtain an estimation of the global fractional resolution of 2.2 \(\pm\) 0.1%.

6.2. Energy Estimation

The length of each track is the sum of distances between two consecutive points. From the length of a track, a kinetic energy can be obtained, assuming a given particle identification, based on the stopping power of muons and protons in liquid argon \[16\] \[17\]. As no particle identification has been performed yet, energies for both hypotheses are estimated for all tracks. It is left to the analyzers, downstream, to decide which one to use based on more rigorous particle identification.

For the sake of this discussion, and evaluating the tracker’s performances, the attribution of proton or muon identification is performed as previously, by using the average ionization and calling muon the particle with the lowest average ionization and proton the one with the highest.

A range-based estimation of the neutrino energy \(E_{\nu}^{\text{range}}\) can be achieved assuming a simple 1\(\mu\)p CCQE (charge current, quasi elastic) interaction by summing the kinetic energies of the reconstructed muon and proton, accounting for mass difference in final and initial states, and using 40 \(\pm\) 10MeV of effective nuclear binding energy \[18\]:

\[
E_{\nu}^{\text{range}} \sim KE_p + KE_\mu + m_\mu + m_p - m_n + B
\]

In the rest of this discussion, we will use true \(E_{\nu}^{\text{range}}\) as reference to the true visible energy as it is the variable we can approach best in a truly perfect reconstruction.

Figure 6 shows the relative error made in reconstructing the full energy of the neutrino. The distribution is fitted with a Gaussian around its central peak to evaluate the global energy resolution. The peak of the distribution shows a bias in reconstructed energy of \(\sim 0.5\%\), with a resolution of 2.2 \(\pm\) 0.1\%.

In the rest of this discussion, we will use true \(E_{\nu}^{\text{range}}\) as reference to the true visible energy as it is the variable we can approach best in a truly perfect reconstruction.

Figure 7 shows the projections of reconstructed tracks for each plane overlaid on top of the corresponding ADC images. The straight vertical light blue lines correspond to the un-responsive wires. The reconstructed energies are respectively 626.8 MeV for the muon track (red dots) and 220.6 MeV for the proton.
Figure 7. Example of a reconstructed MC event of a 974.8 MeV simulated $\mu p$ neutrino event producing a 602.3 MeV muon and a 225.9 MeV proton. The reconstructed tracks are overlaid on top of the ADC image. The event is reconstructed as a 626.8 MeV muon (red) and a 220.6 MeV proton (black), for a reconstructed $E_{\nu}^{\text{vis}}$ of 993.4 MeV.

track (black dots). The reconstructed length-based energy is 993.4 MeV and constitutes an error of +2% from the true visible energy.

7. Conclusions
We have presented a reconstruction method for three-dimensional event reconstruction of two-track events in LArTPCs. We have discussed the algorithms within the context of reconstruction of events in the MicroBooNE detector. This reconstruction uses computer vision and clustering tools to find 3D-consistent vertices, and a 3D stochastic best neighbor search to reconstruct tracks emerging from these vertices. Because the future experiments of the Fermilab SBN program have similar LArTPC design and run in the same BNB neutrino beam-line, the code is easily adaptable for SBN use. The off-beam DUNE program, which will reconstruct atmospheric neutrinos, will also find aspects of the code to be applicable. The code that can be used to perform this reconstruction can be found publicly on GITHUB [12].
Acknowledgments
This material is based upon work 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 of the United Kingdom; and The Royal Society (United Kingdom). Additional support for the laser calibration system and cosmic ray tagger was provided by the Albert Einstein Center for Fundamental Physics. Fermilab is operated by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the United States Department of Energy.

References
[1] B. Fleming, “The MicroBooNE Technical Design Report,” 2012.
[2] R. Acciarri et al., “Design and construction of the microboone detector,” Journal of Instrumentation, vol. 12, no. 02, p. P02007, 2017.
[3] A. A. Aguilar-Arevalo et al., “Improved search for $\nu_\mu \to \nu_e$ oscillations in the miniboone experiment,” Phys. Rev. Lett., vol. 110, p. 161801, Apr 2013.
[4] R. Acciarri, C. Adams, R. An, C. Andreopoulos, A. Ankowski, M. Antonello, J. Asaadi, W. Badgett, L. Bagby, B. Baiubusinov, et al., “A proposal for a three detector short-baseline neutrino oscillation program in the fermilab booster neutrino beam,” arXiv preprint arXiv:1503.01520, 2015.
[5] R. Acciarri, M. Acero, M. Adamowski, C. Adams, P. Adamsen, S. Adhikari, Z. Ahmad, C. Albright, T. Alion, E. Amador, et al., “Long-baseline neutrino facility (lbnf) and deep underground neutrino experiment (dune) conceptual design report, volume 4 the dune detectors at lbnf,” arXiv preprint arXiv:1601.02984, 2016.
[6] C. Andreopoulos et al., “The GENIE Neutrino Monte Carlo Generator,” Nucl. Instrum. Meth., vol. A614, pp. 87–104, 2010.
[7] S. Agostinelli, J. Allison, K. a. Amako, J. Apostolakis, H. Araujo, P. Arce, M. Asai, D. Axen, S. Banerjee, G. . Barrand, et al., “Geant4—a simulation toolkit,” Nuclear instruments and methods in physics research section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 506, no. 3, pp. 250–303, 2003.
[8] J. Allison, K. Amako, J. Apostolakis, H. Araujo, P. A. Dubois, M. Asai, G. Barrand, R. Capra, S. Chauvie, R. Chytracek, et al., “Geant4 developments and applications,” IEEE Transactions on nuclear science, vol. 53, no. 1, pp. 270–278, 2006.
[9] J. Allison, K. Amako, J. Apostolakis, P. Arce, M. Asai, T. Aso, E. Bagli, A. Bagulya, S. Banerjee, G. Barrand, et al., “Recent developments in geant4.” Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, vol. 835, pp. 186–225, 2016.
[10] http://opencv.org/.
[11] R. Acciarri et al., “Noise characterization and filtering in the microboone liquid argon tpc,” Journal of Instrumentation, vol. 12, no. 08, p. P08003, 2017.
[12] https://github.com/LArbys/dllee_unified.
[13] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical image computing and computer-assisted intervention, pp. 234–241, Springer, 2015.
[14] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778, 2016.
[15] C. Adams, M. Alrashed, R. An, J. Anthony, J. Assadi, A. Ashkenazi, M. Auger, S. Balasubramanian, B. Baller, C. Barnes, et al., “A deep neural network for pixel-level electromagnetic particle identification in the microboone liquid argon time projection chamber.” arXiv preprint arXiv:1808.07269, 2018.
[16] http://pdg.lbl.gov/2017/AtomicNuclearProperties/HTML/liquid_argon.html.
[17] https://physics.nist.gov/PhysRefData/Star/Text/PSTAR.html.
[18] M. Anghinolfi, M. Ripani, R. Cenni, P. Corvisiero, A. Longhi, L. Mazzaschi, V. Mokeev, G. Ricco, M. Taibuti, A. Teglia, A. Zucchiatti, N. Bianchi, A. Fantoni, V. Muccifora, P. LeviSandri, V. Lucherini, E. Polli, A. R. Reolon, P. Rossi, and S. Simula, “Inclusive electron scattering from an oxygen and argon jet target,” Journal of Physics G: Nuclear and Particle Physics, vol. 21, no. 3, p. L9, 1995.