FERMILAB-PUB-24-0232-CSAID (Accepted)
DOI: 10.1109/TNS.2024.3521357

Machine Learning-Based Extreme Data Reduction for Prompt Supernova Pointing at DUNE

Michael H.L.S. Wang, DUNE collaboration

***Abstract*—One of the goals of the Deep Underground Neutrino Experiment (DUNE) is to use the massive underground liquid argon time projection chamber (LArTPC) detectors at its far site for multimessenger astronomy (MMA), in the detection of neutrinos from core-collapse supernovae (SNe). Its current baseline trigger strategy detects activity in the detector that is consistent with supernova (SN) neutrinos and saves the raw data for further offline analysis but provides no prompt pointing information crucial for optical follow-ups by other observatories. This approach is based on the assumption that prompt pointing determination using raw data is computationaly prohibitive. In this paper, we demonstrate a proof-of-concept based on applying extreme data reduction on the buffered SN data in the DUNE data acquistion (DAQ) system's front-end computers using a machine learning (ML) workflow. This reduces the data by ∼5 orders of magnitude, allowing a full track reconstruction to be carried out quickly on a single server. The total time to perform the ML-based data reduction and the full track reconstruction is less than the time to transfer the SN data back to Fermilab or a High Performance Computing (HPC) center. This shows that prompt processing of raw SN data is possible and in fact trivial once the data has been reduced to reject radiological backgrounds, paving the way to a high-quality SN pointing trigger that is based on fully reconstructed data instead of trigger primitives (TPs).**

***Index Terms*—Data acquisition (DAQ), machine learning (ML), multimessenger astronomy (MMA), supernova (SN), trigger**

I. Introduction

THE primary scientific goals of the Deep Underground Neutrino Experiment's (DUNE) [1] long baseline physics program include the determination of the neutrino mass hierarchy, observation of charge-parity symmetry violation in the lepton sector, and the measurement of neutrino oscillation parameters. Beyond these goals, DUNE also intends to use its massive liquid argon time projection chambers (LArTPCs) as a neutrino observatory for studying solar neutrinos and neutrinos from core-collapse supernovae (SNe). In this paper, we focus on DUNE's ability to detect and trigger on the latter [2], [3]. We begin with a description of the current DUNE supernova (SN) trigger, including a discussion of its shortcomings. We then follow this up by proposing and describing in detail an approach that addresses these shortcomings. Results based on fully simulated samples are presented that demonstrate the effectiveness of this approach. Our main focus in this paper is the machine learning (ML) based data reduction workflow we have developed and to demonstrate a proof-of-principle and show how this workflow can be used to enable a high-quality SN pointing trigger for DUNE based on fully-reconstructed quantities. Despite some shortcomings, the baseline DUNE SN trigger meets the experiment's baseline scientific requirements. Our goal is to present ways to enhance its capabilities.

II. Baseline DUNE SN Trigger

The DUNE Far Detector (FD) will be located 1.5 km beneath the surface of the earth, at the Sanford Underground Research Facility (SURF). It consists of four 17-kton LArTPC detector modules having a total fiducial volume of ≥40 ktons. For simplicity, we assume all four detector modules are identical and based on the horizontal drift technology, each consisting of 150 anode plane assemblies (APAs) with 2,560 channels per APA, for a total of 384,000 channels per module or $\sim 1.5 \times 10^6$ channels for the entire FD [4]. Each APA is double sided with three active layers on each side, starting with an innermost collection layer (Z) followed by two induction layers (V and U). Ionization electrons produce bipolar signals as they drift through induction layers and unipolar signals when they reach the collection plane. Each Z layer has 480 signal channels, while each V and U pair share 800 signal channels. In the V and U planes, the wires wrap around to the other side, sometimes more than once, resulting in more wires than signal channels. DUNE will employ a streaming readout architecture, where all channels are digitized using 14-bit ADCs at a sampling rate of 1.953125 MHz, and read out continuously from the Warm Interface Boards (WIBs) [5] near the detector over Ethernet links, for an output rate of 1.05 TB/sec per module. The data for each module are received by 75 data acquisition (DAQ) readout units (RUs), which each serve two APAs and buffer the data in a software-defined 10-s latency circular buffer implemented in the RU's RAM. Each RU, which is based on a commercial off-the-shelf (COTS) multicore server, executes a trigger primitve (TP) generation algorithm that finds hits or signals in the raw digitized waveforms in each channel. A second algorithm then groups neighboring TPs close in space and time to find trigger candidates (TCs) representing clusters or tracks. A SN trigger is generated if a sufficient number of TCs are found within 10 seconds, consistent with neutrinos from a galactic core-collapse SN. This trigger causes all the data in the 10 second latency buffer to be persisted into non-volatile memory express (NVMe) solid state drives (SSDs) on the RUs. In addition, the subsequent ∼100 seconds worth of incoming data is also persisted into the same SSDs. This data will be buffered on these SSDs while they are transfered to the transient storage on

The original submission date of this manuscript was May 19, 2024. This is the first revision dated Oct. 29, 2024.

Michael H.L.S. Wang is with Fermi National Accelerator Laboratory, Batavia, IL 60510 USA (e-mail: mwang@fnal.gov).

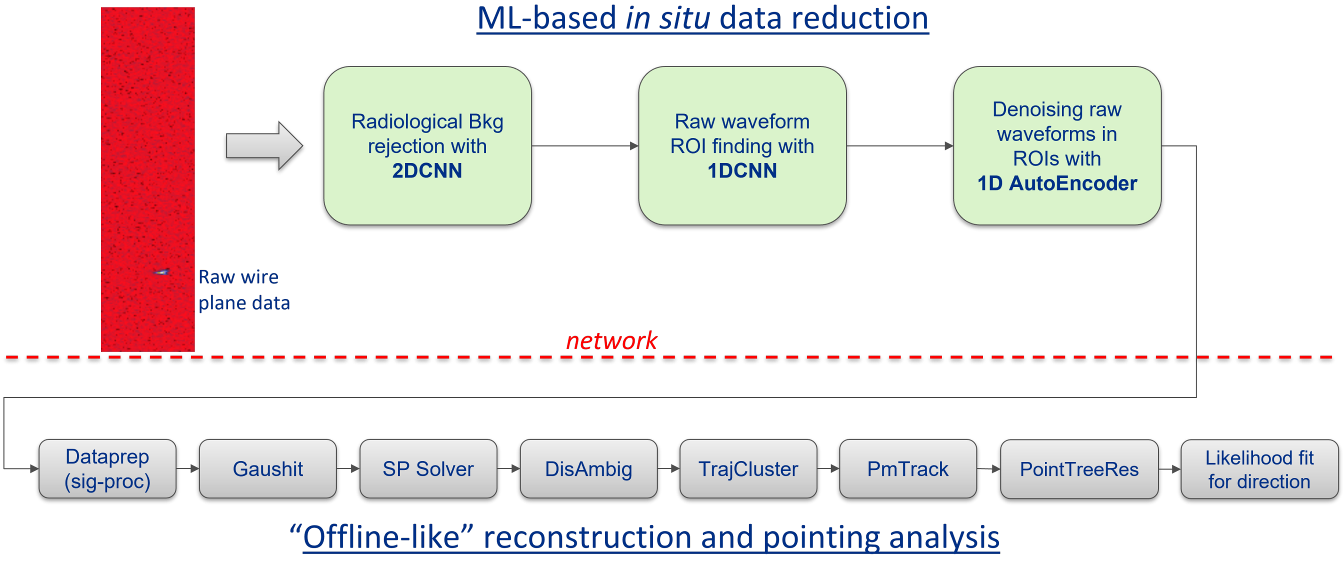

ML-based *in situ* data reduction
Radiological Bkg rejection with 2DCNN
Raw waveform ROI finding with 1DCNN
Denoising raw waveforms in ROIs with 1D AutoEncoder
Raw wire plane data
network
Dataprep (sig-proc)
Gaushit
SP Solver
DisAmbig
TrajCluster
PmTrack
PointTreeRes
Likelihood fit for direction
"Offline-like" reconstruction and pointing analysis

Fig. 1. The two stage strategy described in the text for processing the SN data is shown in the figure above. The upper half shows the ML-based data reduction pipeline used to reduce the raw data by 5 orders of magnitude. The reduced dataset is then transferred over the network to a server that executes the second stage shown in the lower half consisting of a full track reconstruction pipeline followed by the pointing determination.

the surface and then to Fermilab, where additional processing will be performed to determine quantities such as the position of the SN in the sky.

The first 10 seconds of data has enough information for SN pointing determination. The total size of this data for all four FD modules is ~ 48 GB, and will require more than an hour to transfer over the 100 Gbpslinks connecting SURF and Fermilab. In the worst case, it could take much longer, since DUNE's requirement is that all the data needs to be transferred within 24 hours. The time between the arrival of neutrinos and photons from a core collapse SN is approximately equal to the shock propagation time, which ranges from 1 minute to several days, depending on the type of the progenitor [6]. This time window represents the time available for DUNE to determine the direction of the SN and send out notifications over an alert network, and for other observatories to respond by performing optical follow-ups. Unfortunately, the network transfer times between SURF and Fermilab alone would rule out optical follow-ups for most SNs, except for cases when the progenitors are red supergiants.

III. ML-based Solution for SN Pointing at SURF

The major shortcoming of the baseline DUNE SN trigger is that it provides no pointing information prior to transferring the SN data back to Fermilab. This is a consequence of the sheer amount of data produced by continuously read out LArTPC detectors, which can be very challenging to manage and analyze without some form of compression [7] or intelligent data reduction. In order to address this shortcoming, we propose a strategy that consists of two major stages which are depicted in Figure 1. The first stage involves a ML-based data reduction workflow, implemented as early and as close to the buffered SN data on the SSDs as possible. This stage will be performed on the RUs, using co-processors like FPGAs or GPUs with direct access to the data on the SSDs, saving time by minimizing host CPU intervention and eliminating redundant copies to and from host memory. The purpose of this stage is to reduce the data to such a degree that it can be transferred quickly across the network to a single server that performs the next stage. This second stage involves a pipeline consisting of track reconstruction followed by SN pointing determination. The first stage will be executed on the 75 RUs in parallel for the 150 APAs in each LArTPC detector module. The reduced data from all 75 RUs of a detector module will then be received over Ethernet by the server that executes the reconstruction and pointing analysis pipeline in the second stage. In the discussion that follows, we focus only on the first 10 seconds of the triggered SN data stored in the SSDs, since this information is sufficient for SN pointing determination. This immediately provides a factor of 10 reduction in the data to 12 TBs per module.

A. ML-based Data Reduction

As shown in the upper half of Figure 1, the ML-based data reduction workflow implemented on the RUs consists of three separate steps. In the first step, a two-dimensional convolutional neural network (2DCNN) is used on the raw wire plane data to identify 2D regions-of-interest (ROIs), in the form of frames that likely contain SN neutrino interactions. The architecture of this 2DCNN is shown in Figure 2. The objective of this step is to reject majority of the raw data which contain only radiological background. This is done by subdividing the raw wire plane data into smaller, equally spaced, and overlapping subframes spanning the 10 seconds of data. These subframes serve as input images that are fed to the 2DCNN, which performs inference on each separately.

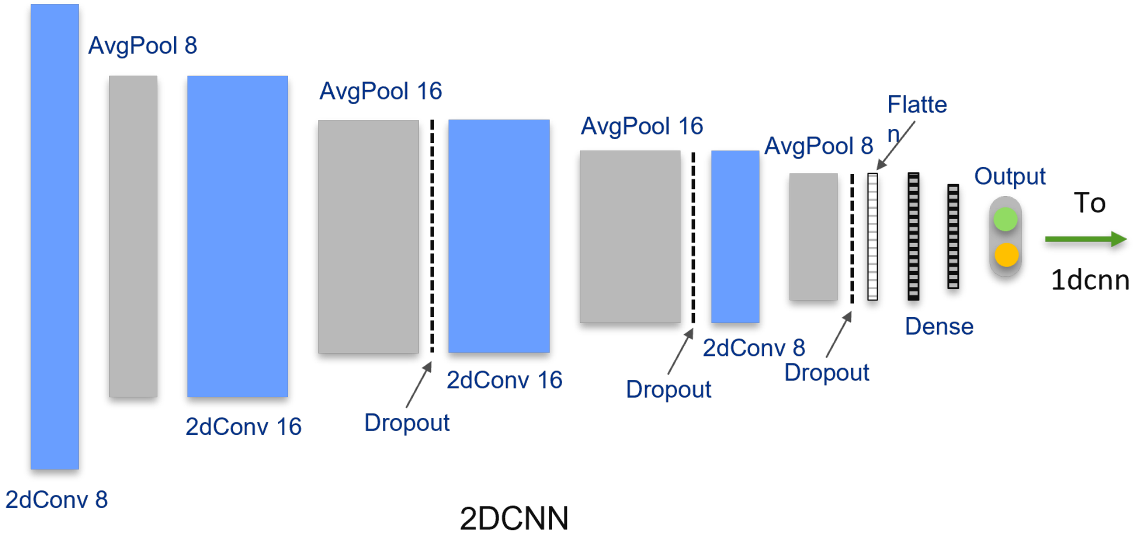


Fig. 2. The figure above shows the architecture of the 2DCNN used in the first step of the ML-based data reduction workflow to reject radiological backgrounds. The architectures for the 1DCNN and 1DAE are not shown in this paper but can be found in references [8], [9], and [10].

Subframes identified by the 2DCNN as likely containing SN interactions are then sent to the second step. In this step, a one-dimensional convolutional neural network (1DCNN) [8] is applied to each individual wire in the subframe to identify only those wires that likely contain a signal in the raw waveform. Wires identified as having signals are then sent to the third step, which uses a one-dimensional autoencoder (1DAE) [9], [10] to denoise the raw wire waveforms. These denoised waveforms on individual channels represent the reduced dataset that is then transferred over the network to the server that executes the second stage of processing described in the following section. An example, using fully simulated data, illustrating the results at each step in this workflow, is shown in Figure 3.

B. Track Reconstruction and Pointing Determination Pipeline

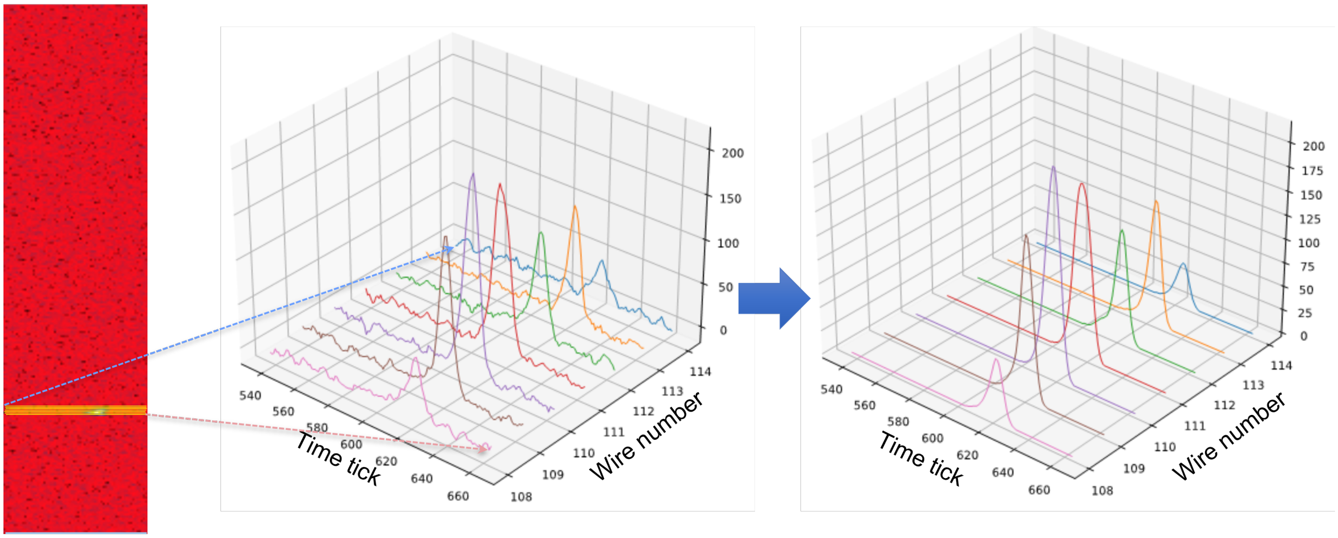
A diagram depicting the workflow in the second stage of processing is shown in the lower half of Figure 1. It involves a typical track reconstruction pipeline employed in LArTPC-based neutrino experiments, followed by analysis to determine the SN direction. The track reconstruction begins with a signal processing step that performs a fast Fourier transform (FFT) based deconvolution on the denoised waveforms from the previous stage, to restore the original waveform, free from shaping effects due to the field response and electronics. This is followed by a step that fits a Gaussian function to the deconvolved waveform, to extract hit parameters like peak position and the area, which is necessary for estimating the deposited charge [11]. After this, 3D space points are determined from the 2D hits in each of the three wire plane views, followed by a step that removes hit ambiguities due to different wire segments that share the same channel. Next, clusters and tracks are reconstructed. Finally, a likelihood fit is performed to extract the direction of the SN.

IV. Performance Tests and Results

In this section, the results presented on algorithm performance were all determined with fully simulated events. SN neutrino interactions in liquid argon considered in these studies included electron-neutrino charged-current absorption interactions (ν_eCC) and neutrino-electron elastic scattering (eES), both of which were generated using the MARLEY event generator [12], with an input energy spectrum according to the Gava-Kneller-Volpe-McLaughlin (GVKM) model [13]. Radiological backgrounds were generated primarily using BxDecay0 [14]. Particle passage through the detector volume was simulated using the GEANT4 simulation toolkit [15], [16], followed by a simulation of detector electronics response.

A. Data Reduction Performance of ML-based Workflow

As mentioned previously, for the purpose of pointing determination, we will limit ourselves to the first 10 seconds of data, which correspond to ~12 TB per LArTPC module. Using the fully simulated events, our ML-based data reduction workflow identifies an average of 124 ROIs for ν_eCC interactions, and 80 ROIs for eES interactions. The average number of ADC samples in an ROI is 218 for ν_eCC interactions, and 212 for eES interactions. With a 14-bit ADC, the average data size is 47,306 bytes for a ν_eCC interaction, and 29,680 bytes for an eES interaction. From the GVKM model, we estimate there will be 3,300 ν_eCC interactions and 326 eES interactions in all four LArTPC detector modules, over a period of 10 seconds, for a galactic core-collapse SN. To get a rough idea of the size of the reduced data sample, we assume that the 2DCNN rejects 100% of all radiological backgrounds and that all neutrino interactions are retained. This is not an unreasonable assumption since, based on simulated samples, the 2DCNN rejects $\gtrsim$99%. This results in a reduced data size of 158 MB for the full detector, which represents a data reduction of five orders of magnitude from the initial size of 48 TB.



(a) 2DCNN

(b) 1DCNN

(c) 1DAE

Fig. 3. (a) Raw data subframe identified by the 2DCNN as likely containing an SN neutrino interaction. The yellow horizontal lines in this subframe are the seven wires that the 1DCNN identified as likely having signals. (b) Raw waveforms containing these signals are shown. (c) These raw waveforms are processed using the 1DAE, resulting in the denoised waveforms shown.

B. Track Reconstruction Performance on the Reduced Dataset

The reduced dataset, resulting from running the ML-based workflow described above on the fully simulated SN samples, is then processed through the track reconstruction pipeline shown in the lower half of Figure 1. This pipeline was executed within the liquid argon software (LArSoft) framework [17] in single-threaded operation, using a single CPU core of an Intel Xeon E-2176M processor, with access to 32 GB of DDR4-2666 RAM. The time it took to run this pipeline, from the FFT deconvolution up to the track finding step, was 61 ms for ν_eCC interactions, and 26 ms for eES interactions. For the sake of simplicity, we assume, once again, that the ML-based data reduction rejects all radiological backgrounds and retains all neutrino interactions. Using these per-interaction execution times, and the numbers of ν_eCC and eES interactions produced in the full detector within 10 seconds from the GVKM model, we estimate it will take 3.5 minutes on a single CPU core, to execute the track reconstruction pipeline on the reduced data set from the full detector. Running this pipeline with multiple threads on a multi-core server can easily reduce this execution time to less than a minute. Peak RAM or resident set size usage reported by LArSoft was 1.9 GB (1.7 GB) for the ν_eCC (eES) interaction job. Peak virtual memory usage was 3.3 GB for both cases.

TABLE I
HIT FINDING EFFICIENCY

Reconstruction	Primary track hits			Daughter track hits		
chain	U	V	Z	U	V	Z
ML-reduced	0.69	0.71	0.66	0.16	0.18	0.073
Std full dataset	0.68	0.67	0.62	0.14	0.17	0.062

The table above shows the hit finding efficiency after the Gaushit finder stage for the ML-reduced reconstruction and the full dataset reconstruction for the induction (U, V) and collection (Z) planes. Uncertainties for all planes are $\lesssim 0.0003$.

To see if the ML-based data reduction step has any effect on the quality of the results from the track reconstruction pipeline, we also run the entire simulated sample, without data reduction, through the same pipeline, and use that as a reference for comparison. The two reconstruction pipelines are identical, except that we use a simpler 1D deconvolution in the signal processing step for the reduced dataset, instead of the more sophisticated 2D deconvolution, which is the default in the offline analysis workflow used on the entire data sample. The basic idea in this step is that the measured signal is a convolution of a response function, characterizing the shaping effects of the field and readout electronics, with the true signal. This true signal can be extracted by performing a Fourier transform into frequency space where the response term can be removed due to simple factorization, followed by an inverse Fourier transform back into temporal space. The difference between 1D and 2D deconvolution is that the latter considers contributions from signals in the neighboring wires. The results of this comparison are shown in Table I for the hit finding efficiency after the Gaushit finder [11] step in the reconstruction pipeline. This is determined by first using Monte Carlo truth information to identify all raw waveform hits or signal regions that are matched to the energy depositions of all the charged tracks associated with the neutrino interaction. The efficiency is the fraction of these hits found by the Gaushit finder. A comparison of the reconstructed energy of the scattered electron in the eES interactions is shown in Figure 4. In both cases, we see that applying the ML-based data reduction step prior to the track reconstruction pipeline has absolutely no negative effects. The more computationally intensive 2D deconvolution does not

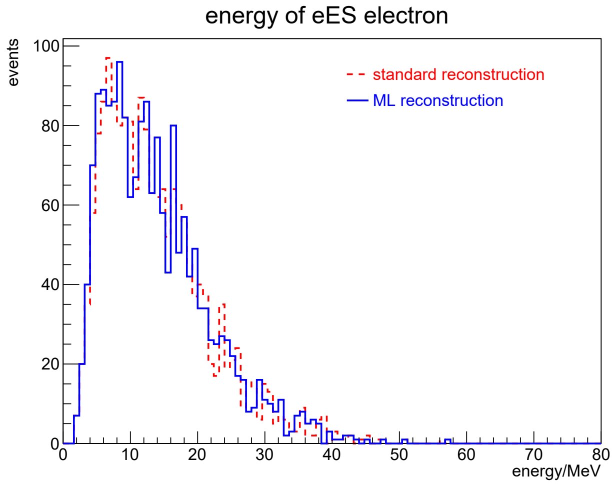


Fig. 4. Comparison of the energy distributions of the scattered electron in the eES interactions between the ML-reduced reconstruction and the standard full dataset reconstruction described in the text.

appear to improve results, likely because we are dealing with isolated low energy depostions where neighboring wire effects are minimal.

C. Processing Time for the ML-based Data Reduction Pipeline

While the results above show that the ML-based data reduction is effective in picking out SN neutrino interactions from the background, and speeds up the track reconstruction considerably, without any degradation in reconstruction quality, all processing, including the data reduction, must be performed within time constraints, in order to complete pointing determination and send out alerts early enough to permit optical follow-ups. Ultimately, the ML-based data reduction should be implemented on devices that achieve fast inference with low power consumption, such as FPGAs. However, as the main purpose of this work is to demonstrate a proof-of-principle, we implemented the ML data reduction algorithms on a GPU, to get a ballpark figure for what can realistically be achieved in terms of total processing times. We performed our timing estimates using half of an 80 GB Nvidia A100 GPU. Referring to Section III-A, since we use 200 tick-wide subframes that overlap by 50 ticks, we need to perform 130,208 inferences with the 2DCNN to cover 10 seconds of data.

We feed raw wire plane images that are 200 time ticks wide as input to the 2DCNN described above. The time to perform an inference on a single 480×200 (1148×200) collection (induction) plane raw data subframe with the A100 is 524 μs (785 μs). The total inference time on all 3 wire plane views is therefore $2 \times 130,208 \times (.000524 + 2 \times .000785) = 545.31$ seconds $=$ 9.1 minutes, where the first factor of 2 is due to the two TPCs associated with each double-sided APA. There is an additional 4.8 minutes, mainly due to the task of constructing the input images for the 2DCNN. The 1DCNN and 1D denoising autoencoder contribute very little to the total execution time, since they are only used to perform inference very rarely, after the 2DCNN has rejected most of the data. The total time to perform the ML-based data reduction is therefore approximately 14 minutes, assuming this can be done in parallel for all APAs in the full detector.

If we add the 3.5 minutes for the track reconstruction pipeline, the total is less than 20 minutes. In other words, we can perform a full track reconstruction on the SN sample in less than the data transfer time back to Fermilab alone, of 65 minutes for 10 seconds of data over 100 Gbps links. The likelihood fit to determine the SN direction takes ~700 ms to perform and including it contributes little to the total time.

While it does not contribute to the total execution time of the ML-based pipeline, the training time for the machine learning models is an important consideration since it has an impact on how easily models can be deployed or retrained. Fortunately, none of the models used in this pipeline are difficult to train. Training the 2DCNN, which is the most complex model, requires less than an hour for the collection plane and a little over a couple of hours for the induction planes.

V. Future and Other Related Work

The approach we describe above can enable DUNE to perform a SN pointing determination on site at SURF, with the same pointing resolution as a full offline analysis done after transferring the data to Fermilab, but in significantly less time, $\mathcal{O}(10)$ minutes instead of hours. This is because it executes essentially the same offline reconstruction and analysis pipeline described in [3] on the full 10-second raw data, after drastically reducing this data in situ. However, this is by no means the only way to achieve prompt SN pointing in DUNE. For example, the trigger primitives, which provide summarized information describing signal regions in the raw waveforms in terms of parameters like *start time*, *time over threshold*, and *area*, can be used to form crude clusters or tracks for pointing determination. While this will not provide results as precise as the method we propose, it may be the fastest way to approximate SN direction. For this reason we are also looking into such approaches to allow DUNE to send out multimessenger alerts as early as possible and to follow them up with more refined results as they become available.

Power for the DAQ computing needs of each FD module is provided by a 150 kVA transformer. It is estimated that the DAQ hardware for one module will consume $\approx$ 70% of this capacity, based on a configuration consisting of 38 switches and 85 servers for readout, 8 spine/leaf switches, 44 control/intelligent platform management interface (IPMI) management switches, and 50 servers for storage and trigger/data selection. This figure is likely underestimated because it is only based on DC power draw without including power supply inefficiencies, and because uninterruptible power supply (UPS) and lighting needs have not been taken into account. Nevertheless, it is clear that the power budget for DAQ computing at SURF is quite limited. For this reason, FPGAs offer a good solution for the ML-based data reduction workflow, because they can provide fast inference times with relatively low power consumption ($\lesssim$ 100W). However, newer GPU architectures are beginning to achieve similar energy efficiencies while maintaining high performance, and supporting features that provide direct paths to their memory from network interface cards or storage devices (e.g. Nvidia's GPUDirect [18]). We plan to investigate the use of this GPU technology for our application, since GPUs offer significant advantages over FPGAs in terms of ease of programming and maintenance.

Finally, a valid concern with ML-based approaches like the one we propose is that models trained on simulated data may not work well with the real data to which they will be applied. This could be due to poor simulations or assumptions made in the simulations that do not reflect real data or changes in detector running conditions. For instance, incorrect assumptions in the modeling of the SN signal and the radiological background, or an over-simplified electronics noise simulation can be detrimental to our ML-based SN data reduction pipeline. To address this, we plan to look into domain adaptation techniques [19] that aim to align the source domain (training samples) better with the target domain (real data).

VI. Conclusion

The current baseline DUNE SN trigger can alert other observatories that a SN neutrino burst has occurred, but, beyond that, it provides no information about the direction of the source of the burst. Unfortunately, this missing piece of information is crucial for allowing optical follow-ups to be performed in a timely fashion. The prevailing assumption is that SN processing requires significant computational resources, requiring the SN data to be sent back to Fermilab or an HPC center for additional processing and analysis, before a careful determination of the SN direction can be made. In this paper, we demonstrated that the key to SN processing is to reduce the raw data by rejecting as much of the background and retaining as much of the signal as possible. We showed that, after reducing the SN data by five orders of magnitude through a ML-based workflow, the task of performing a full track reconstruction, producing results as good as those from a full offline reconstruction, becomes trivial, taking little time with minimal CPU resources. By benchmarking on a readily available GPU, we also showed that total time to reduce the raw data and run the full track reconstruction was less than the time to transfer the data back to Fermilab. These results dispell previously held assumptions about computing requirements for SN processing at DUNE, paving the way for prompt, high-quality SN pointing to be completed on-site at SURF, so that alerts can be sent out early enough to make optical follow-ups by other observatories possible. Furthermore, performing a high-quality SN pointing early in the chain at SURF will also reduce the number of fake SNs, reducing network utilization and storage requirements. With these goals in mind, we are now implementing the ML-based data reduction workflow on FPGA hardware, in order to achieve lower inference times and lower power consumption. At the same time we will also explore newer generation GPUs that achieve similar levels of performance as the Nvidia A100 used in our timing estimates, but which consume significantly less power.

Acknowledgment

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics. The DUNE Collaborators are Maira Khan, Jovan Mitrevski, Ben Hawks, Tom Junk, Tingjun Yang, Jennifer Ngadiuba, Pengfei Ding, Kate Scholberg, Janina Hakenmueller, Van Tha Bik Lian, Georgia Karagiorgi, Judicael Claire, Guanqun Ge, Akshay Malige, Tejin Cai, Amanda Weinstein, and Avik Ghosh.

References

[1] B. Abi *et al.*, DUNE Collaboration, "Volume I. Introduction to DUNE," *JINST*, vol. 15, no. 08, p. T08008, Aug. 2020.

[2] ——, "Supernova neutrino burst detection with the Deep Underground Neutrino Experiment," *EPJ C*, vol. 81, no. 5, pp. 1–26, May 2021.

[3] ——, "Supernova pointing capabilities of dune," [Online]. Available: https://arxiv.org/abs/2407.10339. Accessed on: Oct. 29, 2024., 2024.

[4] ——, "Volume IV. The DUNE far detector single-phase technology," *JINST*, vol. 15, no. 08, p. T08010, Aug. 2020.

[5] A. Abed Abud *et al.*, DUNE Collaboration, "The DUNE Far Detector Vertical Drift Technology. Technical Design Report," *JINST*, vol. 19, no. 08, p. T08004, Aug. 2024.

[6] M. D. Kistler, W. C. Haxton, and H. Yuksel, "Tomography of massive stars from core collapse to supernova shock breakout," *Astrophys. J.*, vol. 778, no. 1, p. 81, Nov. 2013.

[7] P. Abratenko *et al.*, MicroBooNE Collaboration, "The continuous readout stream of the microboone liquid argon time projection chamber for detection of supernova burst neutrinos," *JINST*, vol. 16, no. 02, p. P02008, Feb. 2021.

[8] L. Uboldi *et al.*, "Extracting low energy signals from raw lartpc waveforms using deep learning techniques – a proof of concept," *Nucl. Instrum. Methods Phys. Res. A*, vol. 1028, p. 166371, 2022.

[9] J. Mitrevski, "Low energy lartpc signal detection using anomaly detection," Fast Machine Learning for Science Workshop, 2023. [Online]. Available: https://indico.cern.ch/event/1283970/contributions/5550632/. Accessed on: Oct. 29, 2024., pp. 1–8.

[10] V. Lian *et al.*, "Denoising autoencoder for raw, wireplane, waveforms in dune's lartpc detector," NSF HDR Postbaccalaureate Workshop 2023. [Online]. Available: https://indico.cern.ch/event/1253923/contributions/5443551/. Accessed on: Oct. 29, 2024., pp. 1–30.

[11] M. H. L. S. Wang, G. Cerati, and B. Norris, "Optimizing the LArSoft GaussHitFinder Module," Fermilab, Batavia, IL, Tech. Rep. FERMILAB-TM-2731-SCD, 2020. [Online]. Available: https://lss.fnal.gov/archive/test-tm/2000/fermilab-tm-2731-scd.pdf. Accessed on: Oct. 29, 2024.

[12] S. Gardiner, "Simulating low-energy neutrino interactions with marley," *Comput. Phys. Commun.*, vol. 269, p. 108123, 2021.

[13] J. Gava, J. Kneller, C. Volpe, and G. C. McLaughlin, "Dynamical collective calculation of supernova neutrino signals," *Phys. Rev. Lett.*, vol. 103, p. 071101, Aug. 2009.

[14] F. Mauger and V. Tretyak, "BxDecay0 - C++ port of the legacy Decay0 FORTRAN library," [Software]. Available: https://github.com/BxCppDev/bxdecay0. Accessed on: Oct. 29, 2024.

[15] S. Agostinelli *et al.*, "Geant4–a simulation toolkit," *Nucl. Instrum. Methods Phys. Res. A*, vol. 506, no. 3, pp. 250–303, 2003.

[16] J. Allison *et al.*, "Recent developments in Geant4," *Nucl. Instrum. Methods Phys. Res. A*, vol. 835, pp. 186–225, 2016.

[17] E. Snider and G. Petrillo, "Larsoft: toolkit for simulation, reconstruction and analysis of liquid argon tpc neutrino detectors," *J. Phys. Conf. Ser.*, vol. 898, no. 4, p. 042057, Oct. 2017.

[18] Nvidia Corp., "NVIDIA GPUDirect," [Online]. Available: https://developer.nvidia.com/gpudirect. Accessed on: Oct. 29, 2024.

[19] A. Ćiprijanović *et al.*, "Domain adaptation techniques for improved cross-domain study of galaxy mergers," [Online]. Available: https://arxiv.org/abs/2011.03591. Accessed on: Oct. 29, 2024.