EPJ Web of Conferences **337**, 01144 (2025) https://doi.org/10.1051/epjconf/202533701144
CHEP 2024

A Graph Neural Network Cosmic Muon Trigger for the Mu3e Experiment

David Karres[1,*], *André* Schöning[1], *Sebastian* Dittmeier[1], and *Tamasi* Kar[1] on behalf of the Mu3e collaboration

[1]Physikalisches Institut, Ruprecht-Karls-Universität Heidelberg,
Im Neuenheimer Feld 226, 69120 Heidelberg

Abstract. The Mu3e experiment at the Paul Scherrer Institute will be searching for the charged lepton flavor-violating decay $\mu^+ \to e^+e^-e^+$. To reach its ultimate sensitivity to branching ratios in the order of 10^{-16}, an excellent momentum resolution for the reconstructed electrons is required, which in turn necessitates precise detector alignment. To compensate for weak modes in the track-based alignment, which uses electrons and positrons from muon decays, the exploitation of cosmic ray muons is proposed.
The trajectories of cosmic ray muons are very different from the decays of stopped muons in the experiment and cannot be reconstructed using the same method in the online filter farm. For this reason and in view of their comparatively rare occurrence, a special cosmic muon trigger is being developed. A study on the application of graph neural networks to classify events and to identify cosmic muon tracks is presented.

1 Introduction

With the search for physics beyond the Standard Model (BSM) in mind, the Mu3e experiment was designed to search for the charged lepton flavor-violating (CLFV) decay $\mu^+ \to e^+e^-e^+$. This decay is only possible in the SM by considering neutrino mixing and thus heavily suppressed to branching fractions below 10^{-54}. Therefore, any observation of this decay would provide evidence for BSM physics.
Currently under construction at the Paul Scherrer Institute (PSI), the experiment aims to improve the sensitivity for the decay $\mu^+ \to e^+e^-e^+$ by up to four orders of magnitude with respect to the existing limit [1]. The first phase will use the πE5 muon beam at PSI, with an intensity of 10^8 μ/s to achieve an event sensitivity of 2×10^{-15}. With improvements to the detector and by exploiting the new high-intensity muon beam (HiMB, $>2 \times 10^9$ μ/s) the ultimate event sensitivity of 10^{-16} will be reached [2].
The largest challenge is the background suppression. The signal decay is characterized by three electrons being emitted from the same point in space, with an invariant mass equal to the muon mass and a vanishing momentum vector in the center of mass system. To suppress accidental background, a vertex resolution of $< 200\,\mu$m is needed. Additionally, suppressing background from radiative muon decays with internal conversion ($\mu^+ \to e^+e^-e^+\nu_e\bar{\nu}_\mu$) requires a momentum resolution of $< 0.5\,\mathrm{MeV/c}$ for mass reconstruction [3].

*e-mail: karres@physi.uni-heidelberg.de

As multiple Coulomb scattering dominates the spatial resolution of the Mu3e tracking detector, the detector is designed with a minimal material budget. Because of thermal expansion of the tracking modules, it is prone to misalignment and requires a software alignment system. Mu3e plans to perform a track-based alignment using tracks from Michel decays. Inherent weak modes, e.g. twisting or bending of the detector layers, will be resolved by exploiting high-momentum cosmic ray muons (see Figure 1 and [4]). They are particularly important to align the relative positions of the three tracking stations. However, since the topology of cosmic muons is totally different compared to vast amounts of $\mu^+ \rightarrow e^+\nu_e\bar{\nu}_\mu$ background decays, collecting these events requires a dedicated trigger system. Previously this has been studied using pattern recognition by exploiting associative memory chips and dedicated development of application-specific hardware [5]. This study aims to test the feasibility of using graph neural networks as a software-based alternative.
The following subsection provides a brief overview of the Mu3e detector and its data acquisition system. Additionally, the proposed software alignment system is introduced. Finally, the challenges and requirements of a cosmic muon trigger are discussed.

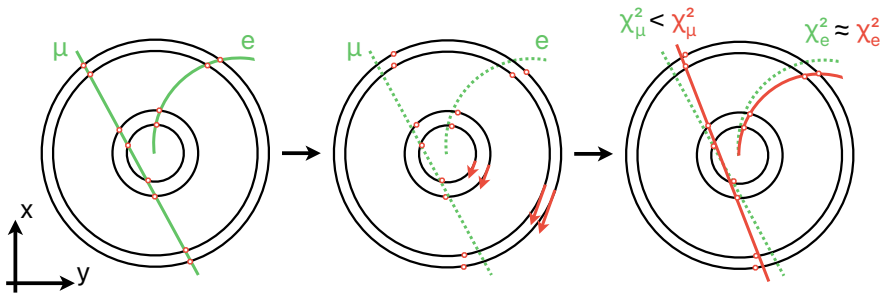

Figure 1: Sketch of the central Mu3e tracking detector in the transversal plane. Left: Fitted tracks (green) of a cosmic ray muon and Michel electron in the aligned case. Middle: A curling weak mode is introduced to the setup. Right: New tracks (red) are fitted in the misaligned case. The newly fitted muon track produces a larger χ^2 than the one in the aligned case, while both electron tracks produce similar fit qualities. Adapted from [6].

1.1 Mu3e Detector Geometry and DAQ

The Mu3e detector is divided into three stations along the beam line. The central station contains the muon-stopping target, made from hollow Mylar double-cones, two double layers of silicon pixel sensors and an intermediate layer of scintillating fibers, providing precise timing information. The acceptance of the tracking detector is extended up- and downstream by so-called recurl stations, which also include a scintillating tile detector for providing a second precise time measurement. The detector is operated in a 1 T solenoidal field and is cooled with gaseous helium.
Data from the tracking and timing detectors is read out by front-end FPGA boards. It is then distributed to the filter farm of 12 GPUs via switching boards. There, an online track and vertex reconstruction of 64 ns time frames is executed. At this stage, only short, outgoing tracks (four hits in the central barrel) are reconstructed. After an online event selection, the data stream is reduced from about 100 Gbit s^{-1} to around 100 MB s^{-1}, which can be permanently stored on mass storage devices.

1.2 Software Alignment Approaches at Mu3e

The detector is subject to misalignment due to several effects: thermal expansion, magnetic field effects and the gaseous helium flow for cooling. The two alignment systems envisioned to mitigate this comprise a camera tracking system and track-based alignment. The camera system consists of three high-resolution cameras on an outer cage, tracking visual markers mounted on the outside of the outermost detector layer. With this information, the position in space can be measured in real time to provide a reference layer for the second alignment method.
As the name suggests, track-based alignment makes use of particle tracks to obtain alignment parameters. This is done by minimizing the following χ^2 function with respect to the track parameters $\mathbf{q}_j$ and the alignment parameters $\mathbf{p}$:

$$\chi^2(\mathbf{q}_j, \mathbf{p}) = \sum_{j}^{\text{tracks}} \sum_{i}^{\text{hits}} \left(\frac{m_{ij} - f(\mathbf{q}_j, \mathbf{p})}{\sigma_{ij}} \right)^2 , \tag{1}$$

where m_{ij} corresponds to a hit measurement with uncertainty σ_{ij} and f to the fit function.

1.3 Challenges and Requirements for a Cosmic Muon Trigger

The online filter farm only reconstructs tracks originating from the target to perform an event selection. Only events containing three tracks, originating from a common vertex are selected. Consequently, cosmic muons are not accounted for, hence requiring a separate online trigger. The main challenge in developing such a cosmic muon trigger is achieving sufficient background suppression. While the muon stopping rate is $O(10^8 - 10^9)$ Hz (for phase I and II), a cosmic muon incident rate of approximately 10 Hz is expected. Assuming readout frames of 64 ns, this equates to about 1 in 10^6 frames containing a cosmic muon track. In order to to cope with the limited processing power of the Mu3e filter farm and to minimize additional computing needs , it is estimated that a cosmic muon trigger should suppress the number of frames at least by a factor 1000, while maintaining a trigger efficiency of over 90%.
Another challenge is the low number of hits cosmic muons produce, compared to Michel electrons and positrons. Because of that, the track multiplicity of signal and background frames is similar (see Figure 2).

2 Track Reconstruction with Graph Neural Networks

Developments of recent years have identified graph neural networks (GNNs) as a promising solution in the context of charged particle track reconstruction [7]. GNNs generally operate on graph-structured data, i.e. a collection of nodes connected by edges. Hits are represented as nodes and the edges between nodes denote whether two nodes are successive hits of a particle's track, i.e. a track segment. The GNN assigns an edge score between 0 and 1 to every edge of the graph. High scores indicate a high probability of that edge being a track segment. The edges are classified as either true or false, depending on a score cut threshold. Tracks are found by disconnecting the graph and only frames containing a connected component with four or more nodes are triggered.

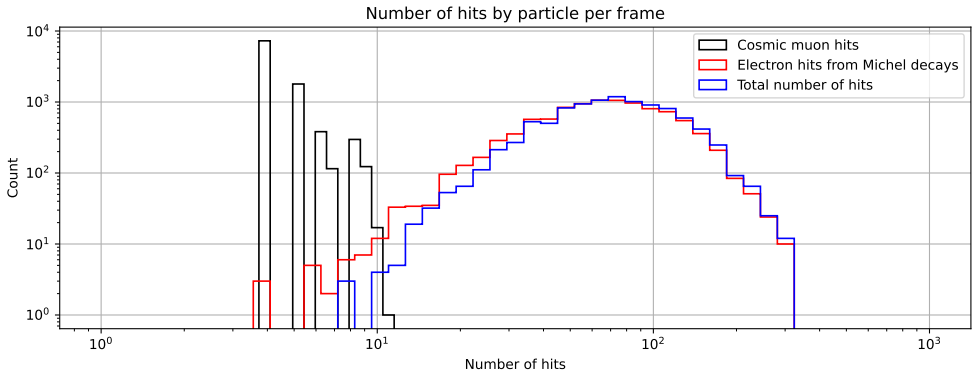

Figure 2: Histograms of the number of hits per particle per frame, evaluated from 10^4 frames containing cosmic ray muons. The black histogram corresponds to cosmic muons, the red histogram to electrons and positrons from Michel decays and the blue one is the sum of both histograms. Adapted from [8].

2.1 Track Finding Pipeline

The track finding pipeline is based on the Acorn framework [9] and works in three main steps.

1. Graph construction: Tracker hits are represented as fully connected graphs without self-loops.
2. Edge labeling: An edge-classifying GNN is inferred on the graphs to produce edge scores for every edge.
3. Track reconstruction: Edges are cut if their assigned score is below a threshold. A connected components algorithm is run to find track candidates with four or more hits.

In comparison to many other experiments, the frames at Mu3e are significantly smaller, with only up to $O(10^2)$ hits per frame. Because of that, fully connected graphs without self-loops can be used.

2.2 Edge Labeling with Graph Neural Networks

The edge labeling is also done in three steps:

1. Encoders: Embed node and edge features into a high-dimensional latent space (see Figure 3).
2. Interaction network: Iteratively perform message passing to propagate information across the graph (see Figure 4).
3. Edge decoder: Transform the latent space representation of the graph into a single edge score.

The node features, i.e. cylindrical coordinates (r, φ, z), are first embedded into a higher dimensional latent space using a multilayer perceptron (MLP). The embedded feature vectors of two connected nodes are concatenated and passed to the edge encoder MLP, producing a

latent space representation of the edge feature vector.
The interaction network comprises two MLPs for processing embedded node and edge features respectively. The node block MLP receives a sum of in- and outgoing edge feature vectors concatenated with the node feature embeddings to update every node embedding (see Figure 5). Next, the edge embeddings are updated by passing a concatenation of the current edge embedding and corresponding updated node embeddings to the edge block MLP (see Figure 6). This procedure is called a message passing step, i.e. the updating of node and edge embeddings, and can be repeated N times to propagate information across the graph.
In the final edge labeling step, a concatenation of updated node and edge embeddings is passed to the edge decoder MLP. A final sigmoid activation then outputs an edge score between 0 and 1.

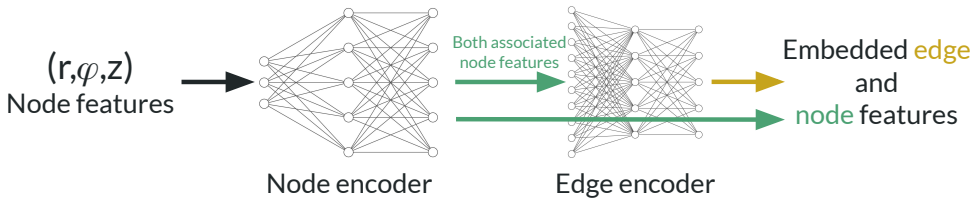

Figure 3: Schematic representation of the two encoder MLPs. The node features are embedded into latent space representations to generate embedded edge features. The network size is not to scale and can be taken from Table 1.

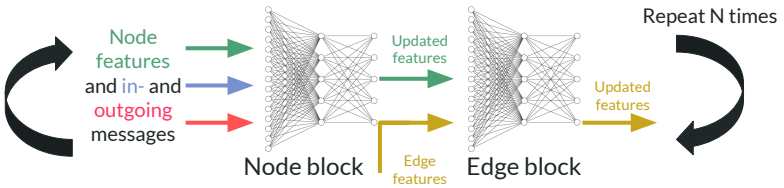

Figure 4: Schematic representation of the interaction network. Embedded node features and summed embedded edge features, i.e. messages, are passed to the node and edge blocks to update them via message passing (see Figures 5, 6). A message passing step can be repeated N times (in this study: N=4) to propagate information along the graph. The network size is not to scale and can be taken from Table 1.

2.3 Network Architecture and Training

The GNN is made up of multiple MLPs that serve different purposes. The dimensionality of node and edge feature embeddings is chosen to be equal and is shown in Table 1. A hyperparameter optimization was used to minimize the number of network parameters to keep the network as computationally inexpensive as possible. Because no substantial improvements in edgewise efficiency or purity are found by increasing the latent space dimensionality, 24 dimensions are chosen for the following performance analysis. Based on the same optimization study, the number of message passing steps is set to 4 and the number of hidden layers for both node and edge MLPs is set to 2. In addition, every layer employs layer normalization

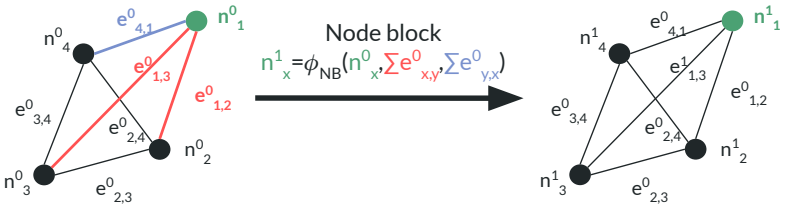

Figure 5: Schematic of a node update. The feature vector of node i is denoted by n_i^k, where k corresponds to the current message passing step. The feature vector of an edge connecting nodes i and j is denoted by $e_{i,j}^k$. The updated node feature vector n_i^{k+1} is produced by passing the current node feature vector and a sum of in- and outgoing edge feature vectors to the node block ϕ_{NB}.

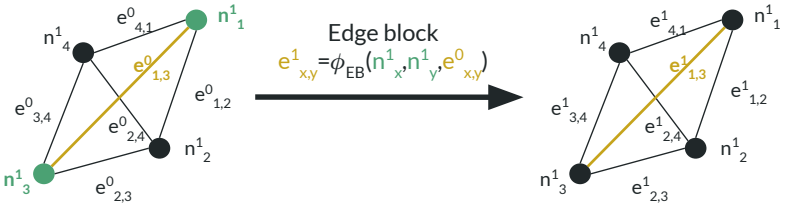

Figure 6: Schematic of an edge update. The updated edge feature vector $e_{i,j}^{k+1}$ is produced by passing the current edge feature vector and both node feature vectors to the edge block ϕ_{EB}. See also description of Figure 5.

Table 1: GNN architecture chosen for this study.

MLP	Input size	Output size	Total number of layers
Node encoder	3	24	2
Edge encoder	48	24	2
Node block	72	24	2
Edge block	72	24	2
Edge decoder	72	1	3

and is activated using the rectified linear unit (ReLU) activation function.

The network training hinges on minimizing a weighted binary cross entropy loss

$$\text{Loss}(y_{pred}) = -\frac{1}{N_{edges}} \sum_{i=1}^{N_{edges}} \text{loss}_i, \tag{2}$$

with the edgewise loss, defined for:

$$\text{True muon edges: } \text{loss}_i = 3 \log(y_{pred,i}) \tag{3}$$

$$\text{Non-target true edges and fake edges: } \text{loss}_i = \log(1 - y_{pred,i}), \tag{4}$$

where the network's prediction for every edge is denoted by $y_{pred,i}$. It is important to note, that true muon edges are weighted by a factor of 3, compared to fake and non-target true

edges, i.e. electron edges. This is done to reward the network more for correctly labeling muon edges and thus shift the focus of the training accordingly.

2.4 Simulation and Training Data

Network training and trigger evaluation are carried out using simulated data from the Mu3e simulation package [2], which, in turn, is based on Geant4 [10]. Two different kinds of frame are considered: background frames only containing hits from Michel electrons and positrons, as well as the signal frames in the scope of the cosmic trigger, i.e. Michel hits overlayed with one cosmic muon track. Two GNNs of the same architecture (see Table 1) are trained: the first one purely on signal frames and the second one on a mixture of 50% signal and 50% background frames. By inferring the network on both samples independently, the trigger efficiency and background acceptance are estimated.

3 Performance Evaluation

This study aims to investigate the feasibility of a GNN-based cosmic muon trigger for the Mu3e experiment. The most meaningful evaluation metrics are the reconstruction efficiency and background acceptance.
The cosmic muon trigger would trigger any frame where the GNN is able to reconstruct at least one cosmic muon track, i.e. at least four nodes connected by at least three edges. For the evaluation, a true track is a track, where every hit stems from a muon and a fake track is any track with at least one non-muon hit.
The reconstruction efficiency ϵ is given as the ratio of the number of (correct) triggers to the number of signal frames on which the GNN is inferred:

$$\epsilon = \frac{n_{True}^{Reco}}{n_{True}}. \tag{5}$$

The background acceptance α is defined as the ratio of the number of false triggers to the number of background frames on which the GNN is inferred:

$$\alpha = \frac{n_{Fake}^{Reco}}{n_{Fake}}. \tag{6}$$

Figure 7 shows the reconstruction efficiency at multiple score cuts between 0.5 and 0.9999 versus the background acceptance.
While both networks reach the goal of a background acceptance of $< 10^{-3}$, the GNN trained on the mixed data set outperforms the other by almost two orders of magnitude in background acceptance. It is important to note that the mixed network is statistically limited, because the network produces only one fake track in 1.6×10^4 frames. In conclusion, the mixed model achieves a reconstruction efficiency of 95.2% with a background acceptance of $6.2 \cdot 10^{-5}$.

4 Summary and Conclusion

The presented study investigates the feasibility of using graph neural networks in the context of a cosmic muon trigger for the Mu3e experiment and makes a first step towards optimizing the performance of this approach. Although the pure track reconstruction efficiency of more than 95% at background acceptances of around 6×10^{-5} satisfies the requirements, the

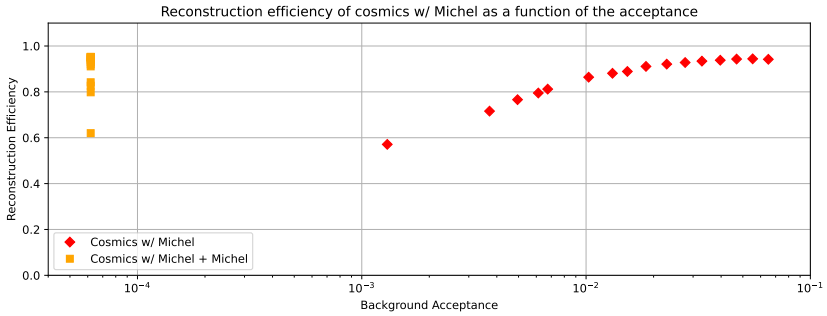

Figure 7: Reconstruction efficiency versus the background acceptance for multiple score cuts $\in [0.5, 0.999]$ of the GNN purely trained on signal frames (red) and on a mix of 50% signal and 50% background frames (orange). In total, 10^3 signal frames and 1.6×10^4 background frames are considered. Adapted from [8].

network must be tested on a larger sample to reduce the statistical limitations. A massive performance boost is observed by including background events in the network's training. With the given performance, a reconstruction rate of under 1000 frames/s for full cosmic muon reconstruction would be needed. An online trigger deployment on either GPU or FPGA is targeted.
The next step toward a GNN-based cosmic muon trigger is a throughput study and, if needed, further optimizations, e.g. the network's architecture.

References

[1] U. Bellgardt et al. (SINDRUM), Search for the Decay $\mu^+ \to e^+e^-e^+$, *Nucl. Phys. B* 299, 1–6 (1988), doi.org/10.1016/0550-3213(88)90462-2.

[2] K. Arndt et. al. (Mu3e), Technical design of the phase I Mu3e experiment, *Nucl. Instrum. Meth. A* (2021), doi.org/10.1016/j.nima.2021.165679.

[3] D. vom Bruch (Mu3e), Track and Vertex Reconstruction on GPUs for the Mu3e Experiment, *GPU Computing in High-Energy Physics* 71-76 (2015), doi.org/10.3204/DESY-PROC-2014-05/13.

[4] U. Hartenstein, Track Based Alignment for the Mu3e Pixel Detector, *PhD thesis, Mainz University* (2019), doi.org/10.25358/openscience-4837.

[5] K. Neureither, Towards an Online Reconstruction of Cosmic Muons for Mu3e using Hardware-Based Pattern Recognition, *Bachelor Thesis, Heidelberg University* (2020), physi.uni-heidelberg.de/Publications/BachelorThesis_KonstantinNeureither.pdf.

[6] D. Fritz, Track Based Detector Alignment using the General Triplet Track Fit, *Master Thesis, Heidelberg University* (2025).

[7] G. DeZoort et al., Charged Particle Tracking via Edge-Classifying Interaction Networks, *Comput Softw Big Sci 5* 26 (2021), doi.org/10.1007/s41781-021-00073-z.

[8] D. Karres, Feasibility Study of a Graph-Neural-Network-Based Cosmic Muon Trigger for the Mu3e Experiment, *Bachelor Thesis, Heidelberg University* (2024), physi.uni-heidelberg.de/Publications/bachelor_karres_corrected.pdf

[9] M.J. Atkinson et al., ACORN, github.com/GNN4ITkTeam/CommonFramework.

[10] S. Agostinelli et at., Geant4 - A Simulation Toolkit, *Nucl. Instrum. Meth. A* 506 250-303 (2003), doi.org/10.1016/S0168-9002(03)01368-8.