



Application of Geometric Deep Learning for Tracking of Hyperons in a Straw Tube Detector

on behalf of PANDA Collaboration

Adeel Akram¹ · Xiangyang Ju² · Michael Papenbrock¹ · Jenny Taylor^{1,4} · Tobias Stockmanns³ · Karin Schönning¹

Received: 16 April 2025 / Accepted: 30 September 2025 / Published online: 21 October 2025
© The Author(s) 2025, corrected publication 2026

Abstract

We present track reconstruction algorithms based on deep learning, tailored to overcome specific central challenges in the field of hadron physics. Two approaches are used: (i) deep learning (DL) model known as fully-connected neural networks (FCNs), and (ii) a geometric deep learning (GDL) model known as graph neural networks (GNNs). The models have been implemented to reconstruct signals in the non-Euclidean detector geometry of the future antiproton experiment PANDA. In particular, the GDL model shows promising results for cases where other, more conventional track-finders fall short: (i) tracks from low-momentum particles that frequently occur in hadron physics experiments and (ii) tracks from long-lived particles such as hyperons, hence originating far from the beam-target interaction point. Benchmark studies using Monte Carlo simulated data from PANDA yield an average technical reconstruction efficiency of 92.6% for high-multiplicity muon events, and 97.1% for the Λ daughter particles in the reaction $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$. Furthermore, the technical tracking efficiency is found to be larger than 70% even for particles with transverse momenta p_T below 100 MeV/c. For the long-lived Λ hyperons, the track reconstruction efficiency is fairly independent of the distance between the beam-target interaction point and the Λ decay vertex. This underlines the potential of machine-learning-based tracking, also for experiments at low- and intermediate-beam energies.

Keywords Hyperons · Tracking · Edge classification · Graph neural networks

✉ Adeel Akram
adeel.akram@physics.uu.se

Xiangyang Ju
xju@lbl.gov

Michael Papenbrock
michael.papenbrock@physics.uu.se

Jenny Taylor
j.taylor@gsi.de

Tobias Stockmanns
t.stockmanns@fz-juelich.de

Karin Schönning
karin.schonning@physics.uu.se

¹ Department of Physics and Astronomy, Uppsala university, Lägerhyddsvägen 1, Uppsala 752 37, Sweden

² Physics Division, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley CA 94720, California, United States

³ Nuclear Physics Institute, Forschungszentrum Jülich, Wilhelm-Johnen-Straße, Jülich 52428, North Rhine-Westphalia, Germany

⁴ Present Address: GSI Helmholtzzentrum für Schwerionenforschung, Planckstr. 1, 64291 Darmstadt, Germany

Introduction

The goal of *hadron physics* is to understand how the strong interaction binds massless gluons and almost massless quarks into the massive hadrons that form our visible universe. The relevant interactions occur within the *confinement domain*, which corresponds to an energy range of ≈ 100 MeV up to a few tens of GeV. Particles produced in this energy region typically give rise to complicated detector signatures, including high-curvature tracks from low-momentum particles. Furthermore, the curvature of these tracks changes along the trajectory due to energy loss and other interactions with the material; hence, these trajectories cannot be described by simple geometries. Another challenge is the similarity between signatures of interest and background – a feature particularly prevalent at the low and intermediate energies characterising hadron physics. Constructing software algorithms that can handle these features is therefore challenging – but necessary to fully exploit the new and next generations of hadron physics facilities. A prominent example of the latter is the future PANDA experiment [1], where a beam of stored antiprotons and a large-acceptance, multi-purpose detector will open new avenues in exploring the strong interaction.

Hyperons are similar to the protons but also contain at least one heavy and unstable strange quark. Due to their weak, self-analysing decays, hyperons provide a precise diagnostic tool to test CP symmetry [2, 3] and electromagnetic structure [4, 5]. Hyperon spectroscopy provides crucial information about how composite systems emerge from strongly interacting quarks and gluons [6, 7], and interactions between hyperons, antihyperons and nuclei provide important pieces to the hyperon puzzle of neutron stars [8]. In PANDA, the antiproton-proton annihilations will enable the production of all known and predicted single-, double- and triple-strange hyperons (Y) in two-body reactions $\bar{p}p \rightarrow \bar{Y}Y$. This provides a clean, particle-antiparticle symmetric final state that is straightforward to parametrise – a significant advantage for spectroscopic partial wave analyses [9], investigations of interaction dynamics [10, 11] and CP symmetry tests [5]. Simulation studies have demonstrated that PANDA will be a *strangeness factory* already in its first operation phase [12].

Identifying hyperons has its unique challenges. Ground-state hyperons decay weakly, on a time-scale of 10^{-10} s. This implies that relativistic particles travel a distance of a few centimetres or even metres before decaying. Therefore, they will leave a track in the detector that starts a finite distance away from the beam-target interaction point (IP), i.e. a *displaced vertex*. As an example, the flight distance of the Λ hyperon is $c\tau = 7.89$ cm. Heavier hyperons, such as the multi-strange Ξ^0 , Ξ^- , Ξ^* , Ω^- and Ω^* , decay by

considerable fractions into states containing a Λ hyperon. Prominent decay channels of charm baryons, such as the Λ_c^+ , also contain the ground-state strange Λ or Σ hyperons. Hence, successful hyperon analyses rely on the ability to reconstruct tracks from displaced vertices and at the same time, to properly handle highly curved or even spiralling tracks from low-momentum particles [13]. However, most tracking algorithms are tailored to primary tracks, since the assumption that a track originates from the IP is a powerful constraint that reduces combinatorics and the background, as well as improves the momentum resolution. This is also the case for the standard PANDA track finder, used in most non-hyperon analyses up to now (see e.g. Refs. [14–16]). In recent years, several efforts have been made to develop algorithms that can handle secondary tracks from displaced vertices [17–19]. These are all based on classical approaches such as the Hough transformation, recursive annealing filters and Apollonius triplets. In particular, the algorithms explored by Aliche in Ref. [19] improve the reconstruction efficiency for secondary tracks up to 79%, to be compared with 45% for the standard primary track finder. Nevertheless, there is room for improvement and it is notable that Refs. [17–19] do not explicitly address low-momentum particles.

Machine-learning techniques are gaining importance in particle tracking, sparked by the *Tracking Machine Learning Challenge (TrackML)* within the high-energy community [20, 21]. In particular, Graph Neural Networks (GNN) have been found suitable for particle tracking in detectors with non-Euclidean geometries such as ATLAS [22, 23], WASA-FRS at FAIR [24], BESIII [25] and Belle II [26]. Since the central part of the PANDA detector has a non-Euclidean geometry, GNNs are natural candidates for facing the challenge of its track reconstruction. It is crucial to investigate how GNN-based solutions tackle the specific challenges of a low-to-intermediate energy experiment such as PANDA, particularly low-momentum particles with displaced vertices. In this paper, we present a detailed ML-based track reconstruction solution for a non-Euclidean PANDA detector system.

The paper is organised as follows. Sect. 2 briefly introduces the PANDA experiment at FAIR with an emphasis on the Straw Tube Tracker, Sect. 3 gives an overview of track reconstruction using machine learning, including an account of contemporary related work. Sect. 4 presents a detailed description of our methodology, from machine learning architecture designs to the performance evaluation metrics. Sect. 5 gives the final results for three different cases: (i) muon reconstruction using conventional deep learning, (ii) muon reconstruction using geometric deep learning, and (iii) hyperon reconstruction using geometric deep learning. Finally, in Sect. 6 we present the conclusions.

The PANDA Experiment at FAIR

The PANDA (anti-Proton ANnihilation at DArmstadt) experiment is currently under construction at FAIR (Facility for Anti-proton and Ion Research) [27, 28]. The anti-proton beam from the High Energy Storage Ring (HESR) will impinge on a fixed, internal hydrogen cluster jet target, a hydrogen or deuterium pellet target (for $\bar{p}p$ or $\bar{p}n$ reactions) or foils (for $\bar{p}A$ interactions). The interaction rate will be 1–20 MHz. A schematic layout of the planned detector is shown in Fig. 1.

The detector will consist of two parts: a Target Spectrometer (TS) employing a solenoid magnet for the detection of particles emitted centrally, and a Forward Spectrometer (FS) with a dipole magnet for forward-going particles. Together, the two spectrometers cover almost the full 4π solid angle. In this work, we focus on the TS that comprises a Micro Vertex Detector (MVD) surrounding the IP, followed by a Straw Tube Tracker (STT) for tracking. A Barrel Detector of Internally Reflected Cherenkov light (DIRC) and a Barrel Time-of-Flight (ToF) provide particle identification and an electromagnetic calorimeter will measure the energies of charged and neutral particles. The solenoid magnet will surround the EMC and its iron yoke will act as an absorber. Most particles that traverse

the full iron yoke will be muons, and they will be tagged in a dedicated muon system. The full PANDA detector is described in detail in Refs. [1, 12].

In this work, we focus on data collected by the STT, which will consist of 4224 single-channel straw tubes, distributed in 27 layers and six sectors in a hexagonal shape, as shown in Fig. 2 (left). The green-marked tubes (15 to 19 layers) will be arranged parallel to the beam axis to measure the hit position, whereas blue- and red-marked tubes (8 layers) are tilted or *skewed* with respect to the beam axis by a $\pm 2.9^\circ$ polar angle. The skewed straws enable reconstruction of the z -component of the tracks [17]. The STT detector will cover polar angles from $\theta = 22^\circ$ to $\theta = 140^\circ$. For a solenoid field of 2 T, particles produced at the IP with a transverse momentum p_T of at least 50 MeV will reach the innermost layer, while a minimum p_T of 100 MeV is required to traverse the full STT.

When a charged particle traverses the STT detector, it ionises the gas inside the tube, releasing electrons. The electrons then drift toward the centre of the tube, ionising more gas molecules on the way. All free electrons will be collected at an anode wire in the centre, resulting in a signal pulse referred to as a *hit*. The xy position of the hit is the position of the anode wire. The distance of the closest approach from the particle trajectory to the anode wire is the *isochrone radius*. Our analysis uses the xy position

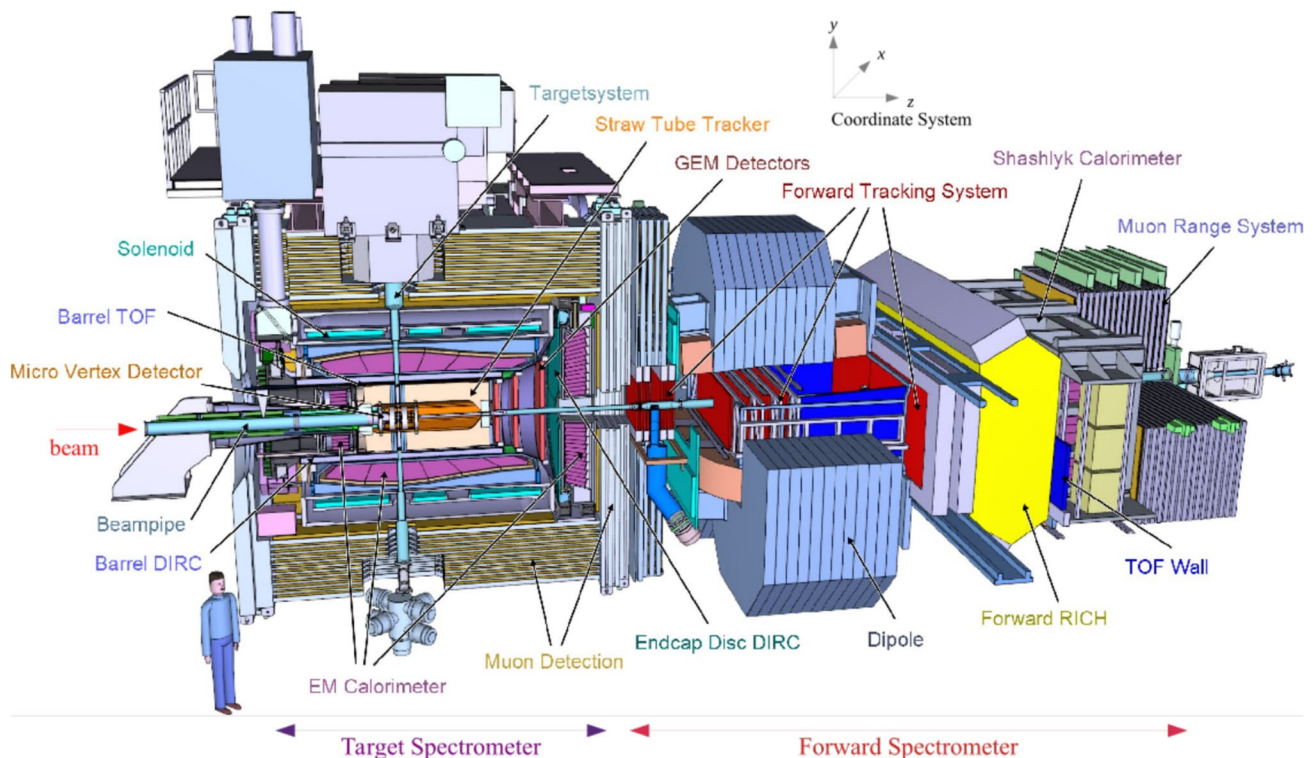


Fig. 1 Schematic of the PANDA Experiment

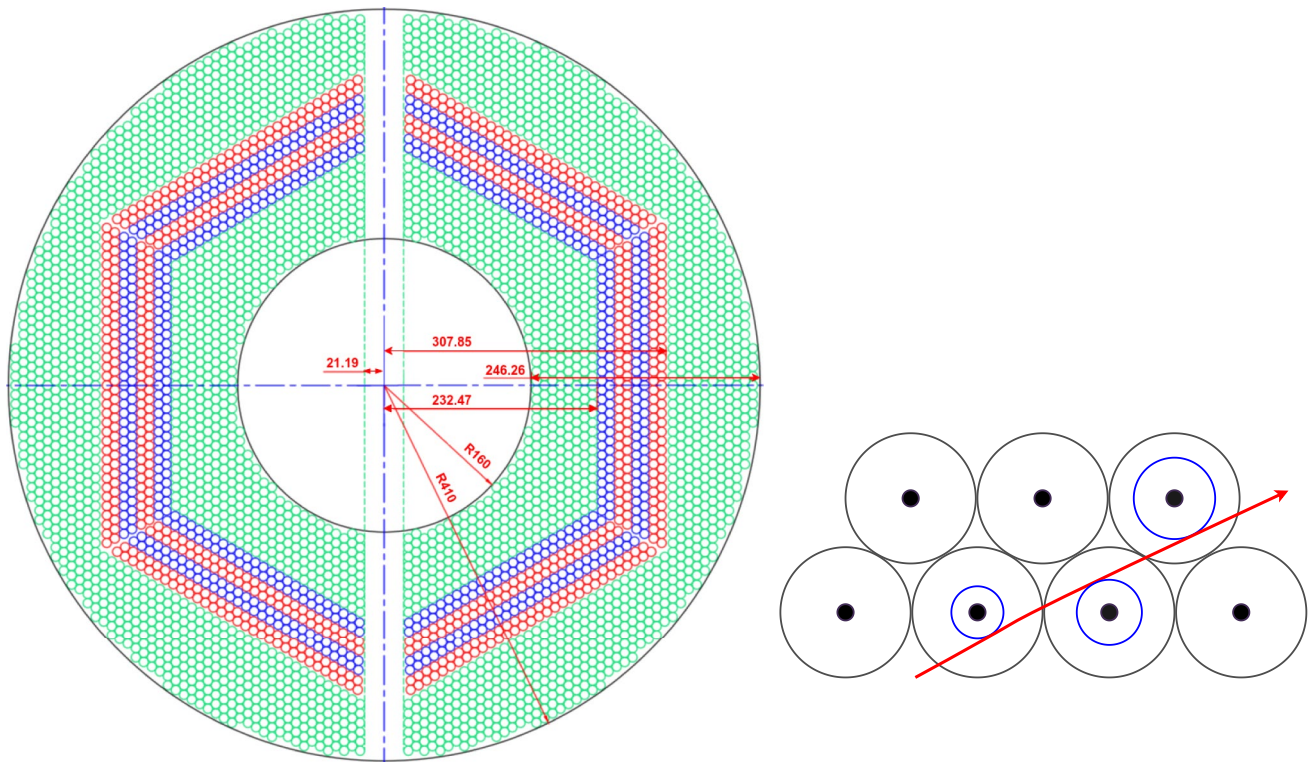


Fig. 2 Left: a cross-sectional view of the STT detector with distance to boundaries in *mm*. The green layers are parallel, whereas blue and red layers are skewed by $\pm 2.9^\circ$ to the *z*-axis. Right: zoom-in view of

straw tubes in black circles, the isochrone radius in blue circles, and a track in red

from the straight straws and the isochrone radii as input data.

PandaRoot Analysis Framework

We used the PandaRoot analysis framework [29] to produce the simulated data for the analysis. PandaRoot offers tools for event simulation, beginning with the production of Monte Carlo events and continuing with the propagation of particles through detector material, digitisation of signals, reconstruction and calibration, and physics analysis. PandaRoot, a detector-specific framework, is derived from the general-purpose FairRoot framework [30], which in turn is based on the ROOT framework [31]. FairRoot constitutes a base for other detector-specific frameworks within the FAIR software ecosystem and provides a wide range of basic classes that facilitate the customisation of each detector configuration. Furthermore, it provides an event display, database management, an input–output manager, a run manager, and the Virtual Monte Carlo (VMC) interface. The latter enables the selection of several simulation engines. In addition, it uses the task system of ROOT to combine and exchange different algorithms into a simulation chain.

Track Reconstruction with ML

Track reconstruction, a process that labels hits to reconstruct particle trajectories, is essential for all physics analyses performed in nuclear and particle physics. Various algorithms have been developed, most tailored to the experiment and the particles at focus. Factors such as particle multiplicity, point of origin, and momentum are crucial when choosing an algorithm. For instance, in high-energy physics (HEP), particles produced in a beam-target interaction have large transverse momenta p_T , resulting in mostly straight particle trajectories. In contrast, hadron physics experiments often produce particles with p_T as low as 100 MeV/c, leading to highly curved particle trajectories that may intersect multiple other particle trajectories in the detector. This factor alone makes track reconstruction a demanding task. Another factor is that long-lived particles, such as hyperons, decay centimetres or even metres from the beam-target interaction point. On the other hand, the track multiplicity is generally lower in hadron physics compared to HEP; it is unusual that one $\bar{p}p$ annihilation event within the energy region of PANDA gives rise to more than ten particles.

At the heart of every algorithm is the pattern recognition algorithm. Most *classical* algorithms [14–16] are combinatorial; they recursively try different hit combinations to find

particles, which makes these algorithms computationally expensive. In this work, we explore an ML-based solution for track reconstruction to address not only the computational challenges but also the aforementioned challenges of track reconstruction in hadron physics experiments.

Related Work

Pattern recognition using neural networks has seen significant advancements in recent years. Within the HEP.TrkX project [32], novel deep learning techniques were developed for track reconstruction to address the challenges of High-Luminosity LHC (HL-LHC). These solutions considered image-based techniques, such as image segmentation and image captioning, recurrent neural networks (RNNs) and convolutional neural networks (CNNs), applied to pseudo-data from simulations of a planar detector geometry [33]. However, these methods do not scale with realistic detectors of irregular geometries and data sparsity. Using the space-point representation of tracking data from a generic barrel detector, RNNs and GNNs were used for track reconstruction with great success [34, 35]. Building upon these developments, the Exa.TrkX project [36] which is the successor of HEP.TrkX, demonstrated the potential of GNNs to broader particle track and shower reconstruction [37], track-seeding and labelling [38] including full-detector analyses [39]. Additional applications of GNNs in particle physics can be found in Refs. [40–42]. Almost all these applications use the TrackML data simulated with a generic detector geometry. The application to the more realistic detector geometries is reported in Refs. [23, 43]. Beyond HL-LHC, the GNNs have been used in several other realistic detectors such as GEM detectors [44], straw-tube detectors [45–47], drift chambers [48–50], and LArTPCs [51, 52]. To keep track of the applications of GNNs in nuclear and particle physics, we refer readers to the *HEP ML Living Review* [53].

In the PANDA experiment, GNN models have been utilised for edge classification in track reconstruction within the Straw Tube Tracker (STT), building upon the methodology outlined in Ref. [41], with preliminary results presented in Ref. [46]. This paper takes a step further by exploring the application of various neural network architectures to data involving muons and hyperons with detailed studies presented in Ref. [47].

Methodology

In this work, we aim to perform pattern recognition using concepts of *nodes*, *edges*, *graphs*, and deep neural networks. It is natural to consider particle trajectories in a detector as graphs, where the graph nodes represent the detector hits and graph edges represent the possibility of two detector hits

coming from the same particle. An edge is labelled as true if the two linked hits are from the same particle and false otherwise. The core idea is to build a graph from the detector hits that includes all true edges and as few fake edges as possible. Then, the graph can be classified by FCNs or GNNs. One can perform classification on graphs in three different ways: (a) node classification where the hits are classified as either signal or noise, (b) edge classification where a link between two hits is classified as either true or false, and (c) graph classification where a full event is classified either signal or noise. For track reconstruction, an edge classification is a suitable option where all edges in a graph are labelled with *edge scores*. The labelled graph is then passed to a clustering algorithm to group hits as track candidates.

First, we consider two representations of the data from the PANDA STT detector: Euclidean and non-Euclidean. They differ in whether a classification model is itself geometrical or non-geometrical. Second, we use two different reactions to produce events: muon pairs and hadrons from a $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ reaction at a beam momentum of 1.642 GeV/c. The latter corresponds to an excess energy of ~ 73 MeV with respect to the $\bar{\Lambda}\Lambda$ production threshold and has been studied in detail before in the context of PANDA with ideal tracking [11]. Our strategy is to first compare both representations using hit data produced by the muons, and then use the best-performing approach to reconstruct hadrons from $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ reaction. Here, tracks from the final state particles originate from the Λ and $\bar{\Lambda}$ decay vertices, typically located several centimetres away from the beam-target interaction point. Hence, this reaction provides a benchmark for evaluating the performance of machine learning algorithms in reconstructing tracks from displaced decay vertices.

The following sections detail the three major steps in our study: data generation and acquisition, the application of deep learning, and the evaluation of reconstructed trajectories. The relevant code used for this work is available at [54].

Data Generation

The muon data sample consists of five $\mu^\pm$ pairs in the momentum range from 100 MeV/c to 1.5 GeV/c, where the muons are produced with a particle gun, isotropically distributed in the STT acceptance. The $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ reaction is simulated with the generator EvtGen [55]. GEANT4 handles the particle transport through the detector material [56]. For both muons and hyperons, 10^5 events are generated for training, which is further distributed into 90% for training, 5% for validation and 5% for testing, respectively. For inference, a separate prediction sample containing $2 \cdot 10^3$ events for muons and $3 \cdot 10^3$ events for hyperons is used to avoid any bias.

Finally, we preprocess the events by creating a hit feature vector for each hit using their positions (r, ϕ, z) and respective isochrone radii and defining true edges between hits with the Monte Carlo truth information.

Deep Learning Pipeline

The deep learning pipeline contains several stages: (1) Graph Construction, (2) Edge Classification, and (3) Graph Segmentation as shown in Fig. 3.

In the graph construction stage, we use a *layerwise* heuristic method to build edges by connecting nodes in adjacent layers, starting from the innermost layer to the outermost layer of the STT. If a node is missing in one layer, the edge is created with the next available layer. We also restrict graph construction in the adjacent sectors of the STT. Since the detector occupancy from hyperons, i.e. the $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ reaction, is much smaller than the muons, we remove the adjacent sector constraint for hyperons. This stage produces an edge list where each edge is labelled as true or false depending on whether it belongs to a particle or not. Furthermore, all edges resulting from the noise, if any, are labelled as false.

In the edge classification stage, we train a deep learning model to classify edges as produced in the previous stage. This stage has two modes: Euclidean and non-Euclidean. In the Euclidean mode, each edge is fed separately to an FCN for classification. The relational information between hits beyond one-hop connections is inherently absent in this mode. In the non-Euclidean mode, the full event presented as a graph is fed to a GNN for classification. The idea is that the GNN can better capture the topological features of data.

Our FCN model has six fully connected layers with hidden dimensions of $[128, 128, 1024, 1024, 128, 1]$. We apply the `relu()` activation function in hidden layers and the `sigmoid()` activation function in the final layer for binary classification. In addition, layer norm is applied to the final layer. These hyperparameters are chosen after tuning the FCN model.

Similarly, the GNN model is the Interaction Graph Neural Network (IGNN) [57] formulated under a message-passing framework [58]. The IGNN consists of three modules: (i) encoder module, (ii) graph module, and (iii) decoder or output module. The encoder module consists of an edge network and a node network, and its task is to encode input node features to a vector of hidden features and to create edge features from neighbouring nodes. In the graph module, aggregated neighbouring edge features are passed to the node network, and the neighbouring node features are passed to the edge network. This is a message-passing step where information is exchanged between nodes and edges. This step is repeated eight times ($N = 8$), where each step is a hidden layer of the graph module. In addition, residual connections $\{H_i\}_{i=0}^N$ are formed by concatenating the input and the output of each graph module. The final output is then passed to the decoder module, which performs binary classification using the binary-cross-entropy loss function. As a result, each edge is assigned an edge score between 0 and 1. A schematic diagram of IGNN used in this work is shown in Fig. 4:

Networks inside IGNN modules are of FCN-type. Each network has a three-layer architecture with nodes of $[128, 128, 128]$ with a ReLU activation function on all layers. The following hyperparameters are used during training:

- Binary Cross-Entropy (BCE) loss
- AmsGrad Optimizer: $\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999$, weight_decay=0.01
- Learning rate: $\alpha = 0.001$
- Message-passing steps or layers: 8, Edge aggregation function: `sum_max()`

The hyperparameters of IGNN are chosen exactly as used in Ref. [41]. We set the batch size to 128 for the FCN and 1 for the IGNN. The networks are trained for 50 epochs as the generalisation gap between the training and the validation errors does not change significantly. Finally, each edge in the graph is labelled with a probability score, later called the edge score.

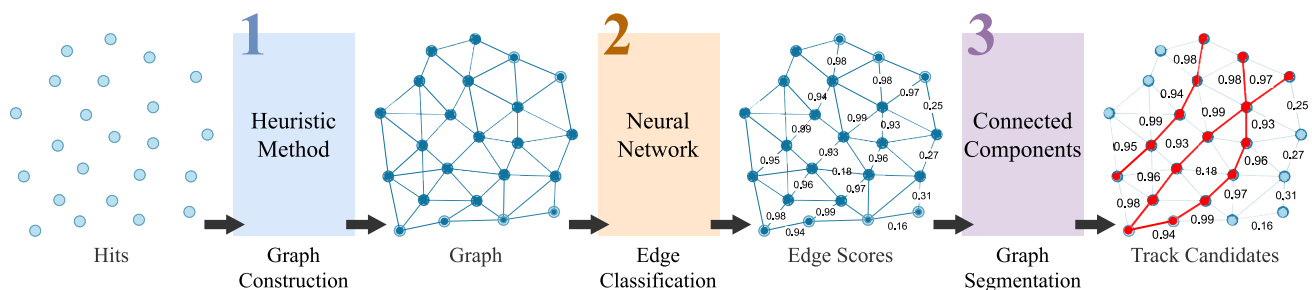


Fig. 3 Sketch of a deep learning pipeline showing graph construction, edge classification and graph segmentation stages, image adapted from the Exa.TrkX [41]

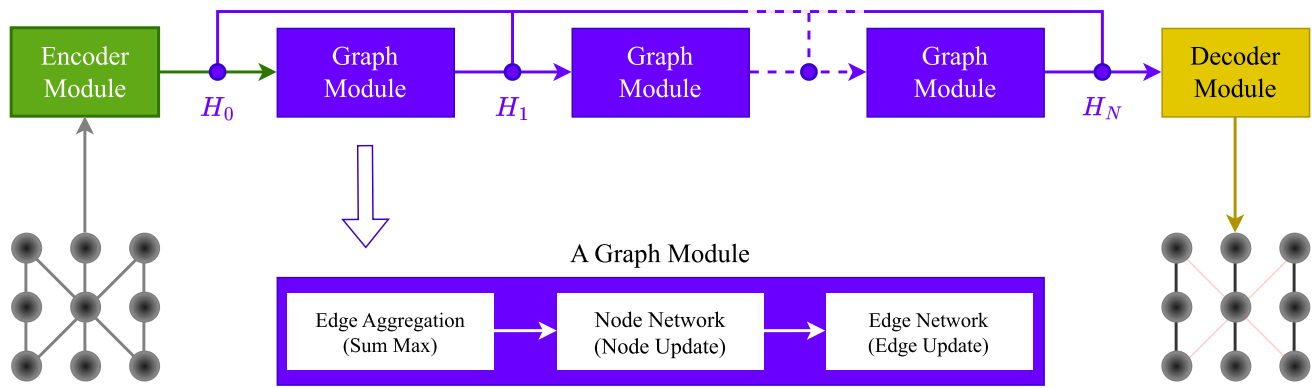


Fig. 4 Schematic of an IGNN with an encoder module, graph modules illustrating message passing, and a decoder module. The output classifies edges marked as red (low score) and black (high score)

In the graph segmentation stage, we use the density-based spatial clustering of applications with noise (DBSCAN) algorithm [59] to find connected components of the labelled graph. We use the graph score as the distance metric between two nodes. The distance metric (ϵ_{db}) defines the maximum distance between two nodes to be clustered together. The value of ϵ_{db} is scanned to find an optimal value where the efficiency of graph segmentation is high.

Track Evaluation

Track evaluation ensures that the reconstructed tracks accurately represent true particle trajectories. One method to assess the quality of the track reconstruction algorithm is by calculating the overall tracking efficiency, referred to here as *physics efficiency* and *track purity*. Physics efficiency indicates how effectively the tracking algorithm can identify all particle tracks from the detector signals, and depends both on the performance of the algorithm and the efficiency of the detector. Track purity refers to the algorithm's ability to distinguish true particle tracks from wrongly reconstructed tracks or other types of backgrounds. To evaluate the performance of the tracking algorithm itself, independently of the detector efficiency, a conditional tracking efficiency is defined, i.e. the *technical efficiency*. When this is evaluated, a minimum number of hits is required below which the algorithm cannot be expected to reconstruct a track. The tracking metrics will be defined using an evaluation scheme that closely aligns with the ATLAS community [60]:

- $N_{\text{particles}}(\text{selected})$ is the number of generated particles in the detector acceptance, which will be referred to as *particles*.
- $N_{\text{particles}}(\text{selected, matched})$ is the number of particles matched to at least one reconstructed track.
- $N_{\text{particles}}(\text{selected, reconstructable})$ is the number of generated particles that leave at least seven or more hits

(N_r) in the detector, they will be referred to as the *reconstructable particles*.

- $N_{\text{particles}}(\text{selected, reconstructable, matched})$ is the number of reconstructable particles that are matched to at least one reconstructed track.
- $N_{\text{tracks}}(\text{selected})$ is the number of reconstructed tracks containing at least five or more hits (N_r), which will be referred to as *reconstructed tracks*.
- $N_{\text{tracks}}(\text{selected, matched})$ is the number of reconstructed tracks that are matched to a particle.

A particle is considered matched to a reconstructed track if more than (i) 50% of the hits in the reconstructed track belong to the same true particle, (ii) 50% of the hits in the matched true particle are found in the reconstructed tracks. This is known as *two-way matching* scheme. Furthermore, the reconstructable particles are the selected particles that also have at least seven hits in the detector before performing the track reconstruction.

The physics efficiency (ϵ_{phys}) is the fraction of particles that match at least one reconstructed track:

$$\epsilon_{\text{phys}} = \frac{N_{\text{particles}}(\text{selected, matched})}{N_{\text{particles}}(\text{selected})} \quad (1)$$

The technical efficiency (ϵ_{tech}) is the fraction of reconstructable particles that match at least one reconstructed track:

$$\epsilon_{\text{tech}} = \frac{N_{\text{particles}}(\text{selected, reconstructable, matched})}{N_{\text{particles}}(\text{selected, reconstructable})} \quad (2)$$

Finally, the track purity (ρ) is defined as the fraction of reconstructed tracks that match a selected particle:

$$\rho = \frac{N_{\text{tracks}}(\text{selected, matched})}{N_{\text{tracks}}(\text{selected})} \quad (3)$$

In addition, the *fake rate* ($\equiv 1 - \rho$) or *ghost rate* is defined as the fraction of reconstructed tracks not matching any particle tracks. In contrast, the *clone rate* is the rate at which a particle is matched to more than one reconstructed track.

In addition to the requirement that the reconstructable track has at least 7 STT hits (i.e. $N_t \geq 7$), reconstructed tracks require at least 5 STT hits (i.e. $N_r \geq 5$) and a matching fraction (MF) greater than 50%.

Results

For a fine-grained understanding of the performance, we investigate how the track efficiencies depend on variables such as transverse momentum p_T , and the radial distance d_0 between the beam-target interaction point and the decay vertex:

$$d_0 = \sqrt{v_x^2 + v_y^2}. \quad (4)$$

where v_x and v_y denote the positions of the decay vertex of particles along the x and y axes, respectively.

Muon Reconstruction with GDL

To reconstruct muons, two different approaches are adopted: the Euclidean, using FCN and the non-Euclidean, using IGNN. We investigate the performance of edge labelling and graph segmentation stages, leading to the evaluation of both approaches.

To examine the edge labelling, different evaluation metrics are used. The model output gives the classification probabilities, referred to as edge scores, for each edge in the

graph. For edge predictions, an optimal threshold on the edge score is required. Fig. 5 shows the model outputs of FCN and IGNN where edge scores of true (blue) and false (orange) edges are shown without applying a threshold on the model outputs.

IGNN gives better separation power between true and false edges compared to the FCN for a particular threshold value. To quantitatively evaluate model performance, we used the Receiver Operating Characteristic Curve (ROC) and measured Area Under the Curve (AUC). The ROC curve is constructed using the edge classification efficiency ($\epsilon_E \equiv \text{TPR}$) and edge classification purity ($\rho_E \equiv 1 - \text{FPR}$) for various thresholds, where TPR is the true positive rate and FPR is the false positive rate from the confusion matrix. The ROC curves along with the AUCs for both models are shown in Fig. 6.

A high value of AUC represents high model performance and vice versa, thus prompting a reasonable model training period. Since the ROC curve is constructed at varying threshold values on the model output, one needs to find an optimal threshold value or edge score cut (s). For this purpose, the ϵ_E and ρ_E are plotted as a function of edge score cut as shown in Fig. 7 for FCN (left) as well as IGNN (right).

The higher values of s give high edge purity but low edge efficiency, and vice versa; hence, there is a trade-off in choosing a particular value of s . For example, choosing $s = 0.5$ gives $\epsilon_E \sim 96\%$ and $\rho_E \sim 97\%$ for the FCN model, whereas $\epsilon_E = 99.2\%$ and $\rho_E = 99.0\%$ for the IGNN. Alternatively, we can examine the signal efficiency (ϵ_{sig}) vs background rejection factor (BRF) at various values of edge score cut. We define signal efficiency as the TPR ($\epsilon_{\text{sig}} \equiv \text{TPR}$) or recall, and misidentification rate as the FPR from the ROC curve. The BRF is defined as the

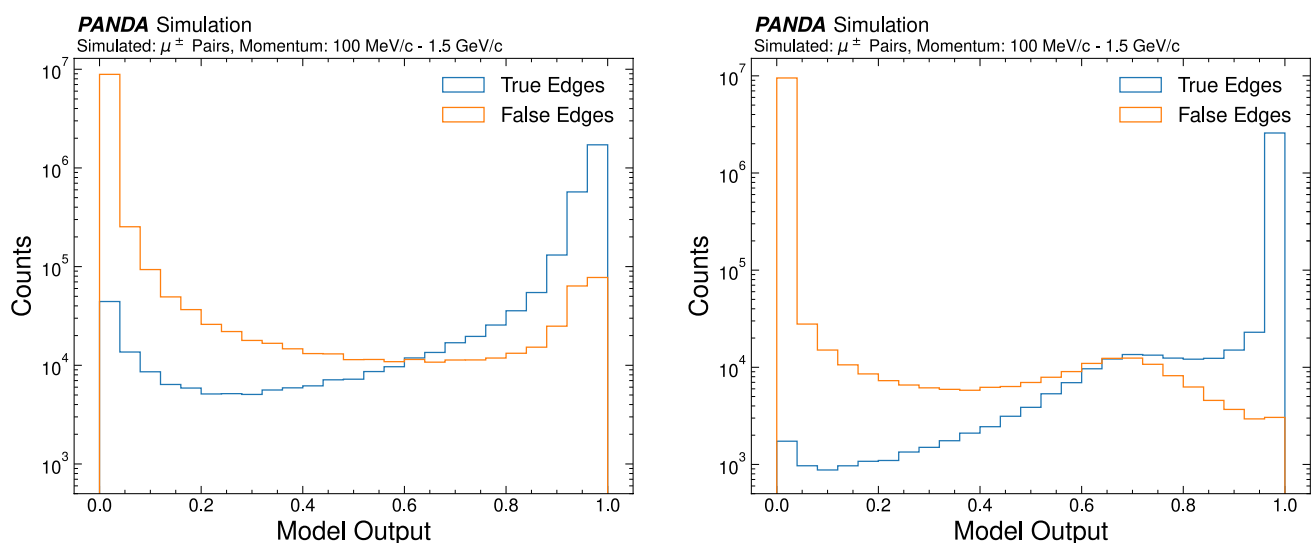


Fig. 5 The output of a classification model: FCN (left) and IGNN (right)

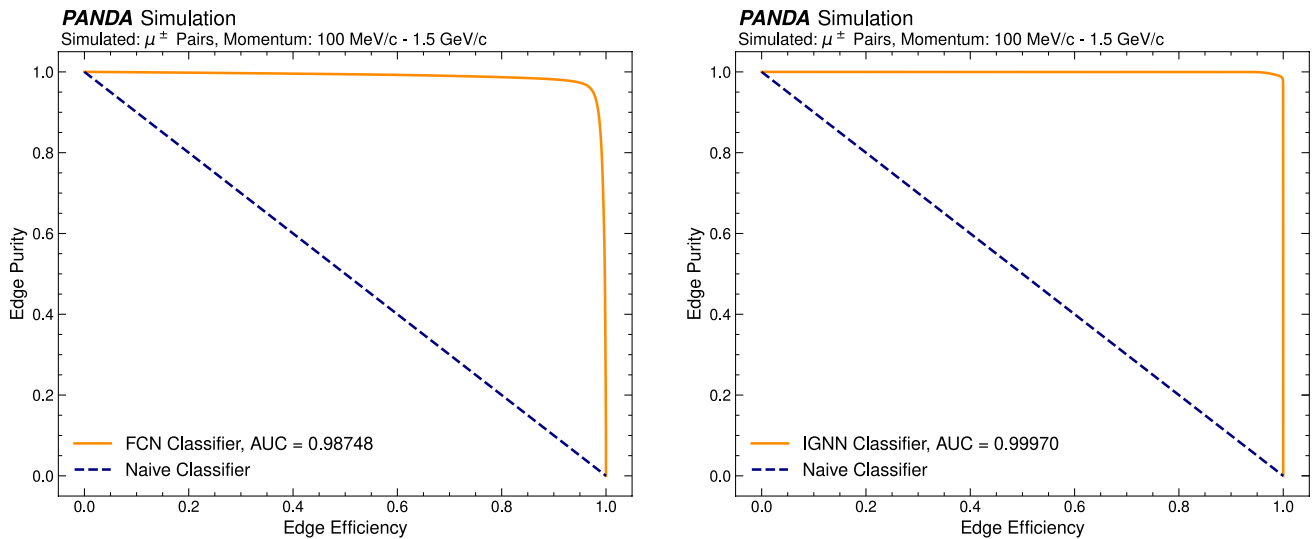


Fig. 6 ROC curve with AUC of 0.98748 for FCN (left) and 0.99970 for IGNN (right)

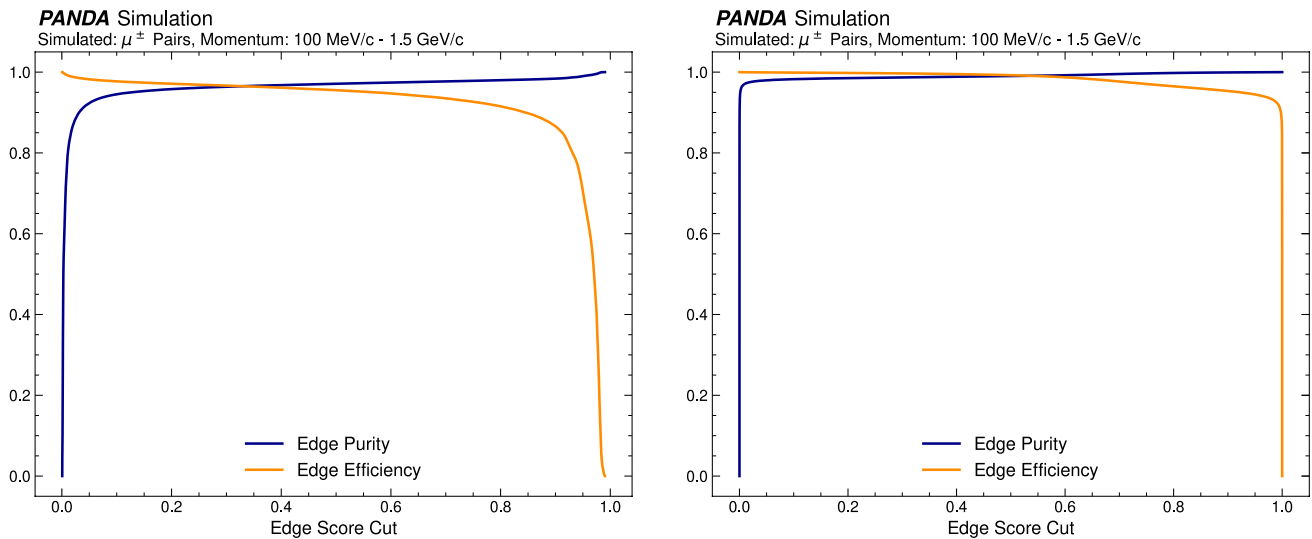


Fig. 7 The edge classification efficiency ϵ_E and edge classification purity ρ_E as a function of s for FCN (left) and IGNN (right)

inverse of the misidentification rate ($\text{BRF} \equiv 1/\text{FPR}$). Figure 8 shows the signal efficiency as a function of the BRF for various values of edge score cut for FCN (left) and IGNN (right) models.

The orange curve shows how the signal efficiency and the BRF depend on s , and the black dot represents the edge score cut value of 0.5. With this cut value, we get $\epsilon_{\text{sig}} = 95.5\%$ and $\text{BRF} = 34.8$ for FCN, and the ϵ_E is 99.2% and the BRF of 101.4 for IGNN. Increasing the cut value to 0.7 yields $\epsilon_{\text{sig}} = 93.5\%$ and $\text{BRF} = 43.4$ for FCN and $\epsilon_{\text{sig}} = 97.7\%$ and $\text{BRF} = 213.5$ for IGNN. Hence, the high BRF comes at a cost of a reduced value of ϵ_{sig} . Therefore, we chose $s = 0.5$ for further analysis.

After edge labelling, we look into the graph segmentation using the DBSCAN algorithm. We utilise a prediction dataset comprising $2 \cdot 10^4$ events for this purpose. The DBSCAN extracts the connected components (Euclidean case) and weakly connected components (non-Euclidean case) of the graphs. This algorithm requires an optimal value of the distance metric (ϵ_{db}) to cluster nodes together. For this purpose, we used different values of ϵ_{db} to find connected components and then performed track evaluation. Figure 9 shows a scan of ϵ_{db} against different tracking metrics for FCN (left) and IGNN (right), where the optimal values of ϵ_{db} are shown as magenta lines with values of 0.20 and 0.25 for FCN and IGNN, respectively.

Finally, we extract the track candidates using the optimal values of ϵ_{db} and track evaluation criteria as discussed in Sect. 4.3. These tracking metrics are summarised in Table 1 for FCN and IGNN models as follows:

In the FCN case, we note that the efficiencies are fairly small from a physics perspective; for example, for events with four tracks, a tracking efficiency of $\approx 77\%$ will result in a three-fold reduction of the total efficiency. Hence, there is room for improvement. One major issue for FCN is to handle

the huge class imbalance with a ratio of true to false edges of 1:4. IGNN can handle such imbalances by aggregating neighbourhood relations through message-passing, which results in an increase from 77.2% to 92.6% in the tracking efficiencies, an almost 20% increase in efficiencies. Furthermore, it reduces the ghost rate to be almost negligible. The clone rate is also reduced, but is still high.

To better understand the algorithm's performance, we investigate how the tracking efficiency depends on the

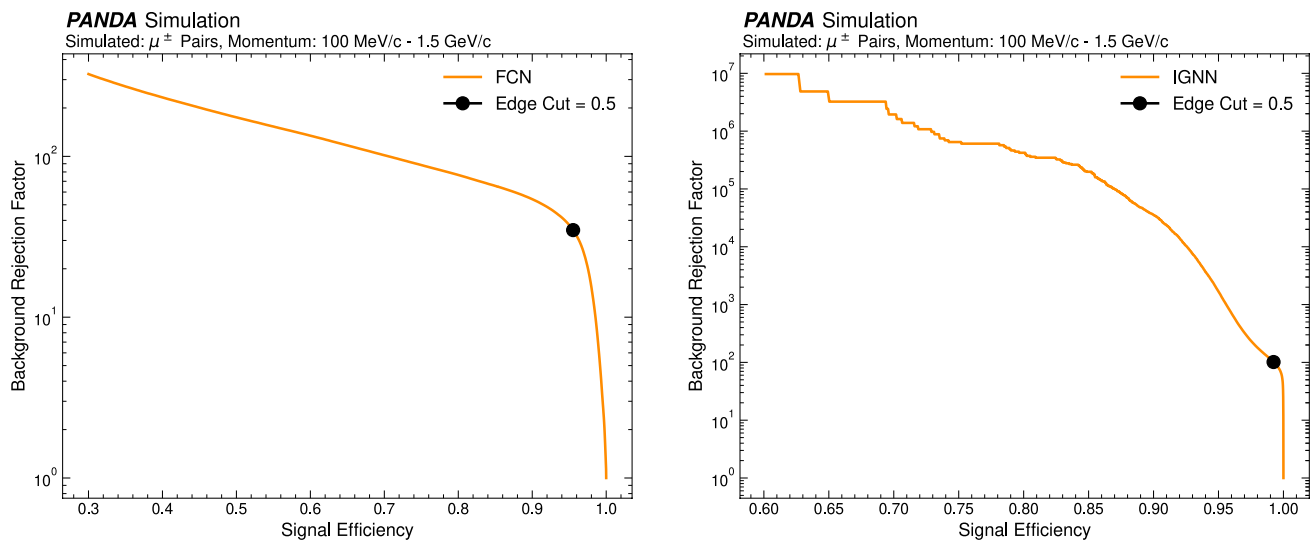


Fig. 8 The ϵ_{sig} as a function of BRF for various values of s for FCN (left) and IGNN (right) models, the dot represents $s = 0.5$

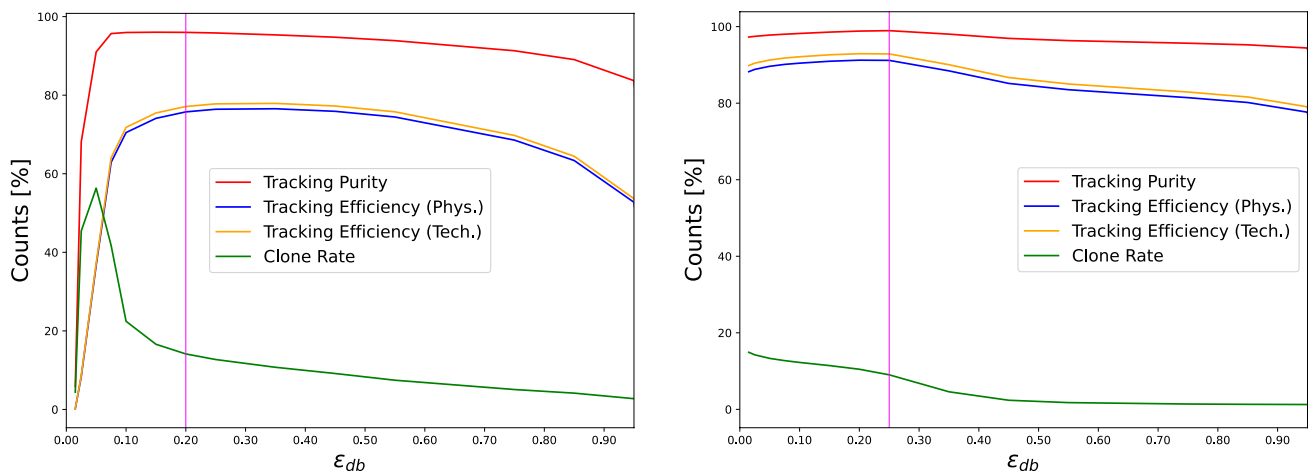


Fig. 9 The scan of ϵ_{db} for DBSCAN algorithm for FCN (left) and IGNN (right). The magenta lines show the selected values of ϵ_{db}

Table 1 Tracking efficiencies, ghost rate, and clone rate for muons

	N_t	N_r	MF [%]	$\epsilon_{phys.}$ [%]	$\epsilon_{tech.}$ [%]	Ghost Rate [%]	Clone Rate [%]
FCN	7	5	> 50	76.3 ± 0.3	77.2 ± 0.3	3.6 ± 0.3	17.2 ± 0.1
IGNN	7	5	> 50	91.0 ± 0.3	92.6 ± 0.3	1.3 ± 0.3	11.5 ± 0.1

transverse momentum (p_T) of muons. Figure 10 shows the number of particles (selected, selected and matched, reconstructable and reconstructable and matched) for FCN (left panel) and IGNN (right panel) as a function of p_T . In Fig. 11, we show the corresponding track efficiencies. We conclude that the main loss of tracks occurs at low p_T , especially below $p_T = 0.25$ GeV/c. Particles with such low p_T have trajectories with so large curvature that they make a turn before traversing the full STT detector. Hence, they are trapped inside the detector, with trajectories spiralling in the magnetic field, lose energy in interactions with the detector material and the gas, and potentially also intersect the trajectories of other particles. This is in contrast to high p_T particles, which have rather straight trajectories with fewer intersections, resulting in higher efficiencies for this class of

particles. The improved performance of the IGNN compared to the FCN for low- p_T tracks is striking, as seen in Fig. 11.

Hyperon Reconstruction with GDL

We have performed simulations for training and testing with the reaction $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ at a beam momentum of $\bar{p}_{beam} = 1.64$ GeV/c. Since Λ hyperons are neutral, tracking information is obtained from their charged daughters, i.e. protons and pions. The $\bar{p}p \rightarrow \bar{\Lambda}\Lambda$ reaction has been rigorously studied by the PS185 experiment at LEAR, CERN [61, 62], in particular at this beam momentum [63]. It has been found that in the CMS system of the reaction, the $\bar{\Lambda}$ antihyperon is emitted predominantly in the forward direction while the Λ hyperons go backwards. This

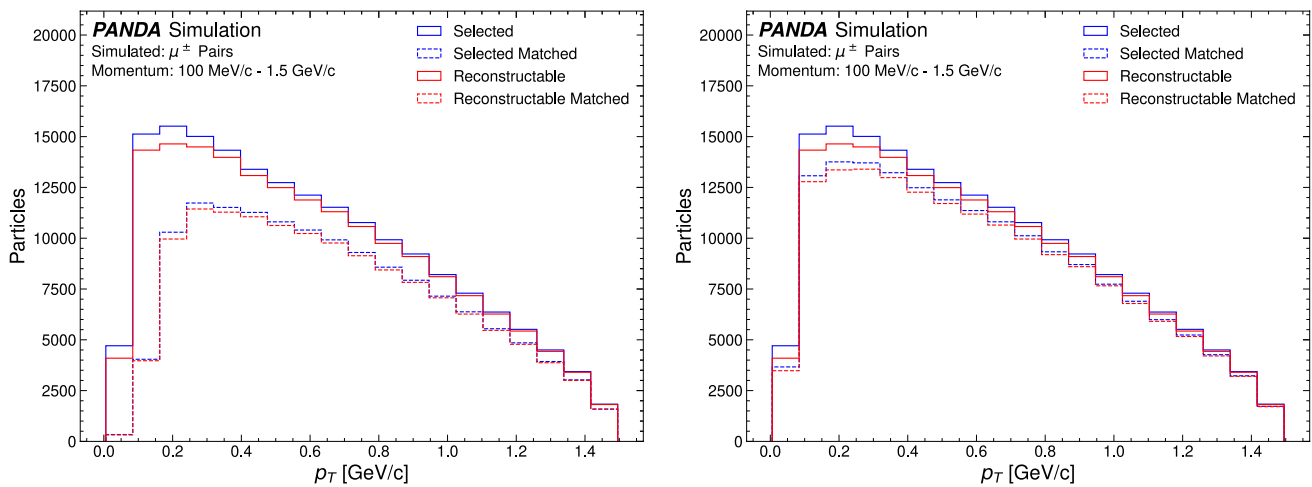


Fig. 10 The number of selected, selected and matched, reconstructable, reconstructable and matched particles as a function of p_T for FCN (left) and IGNN (right)

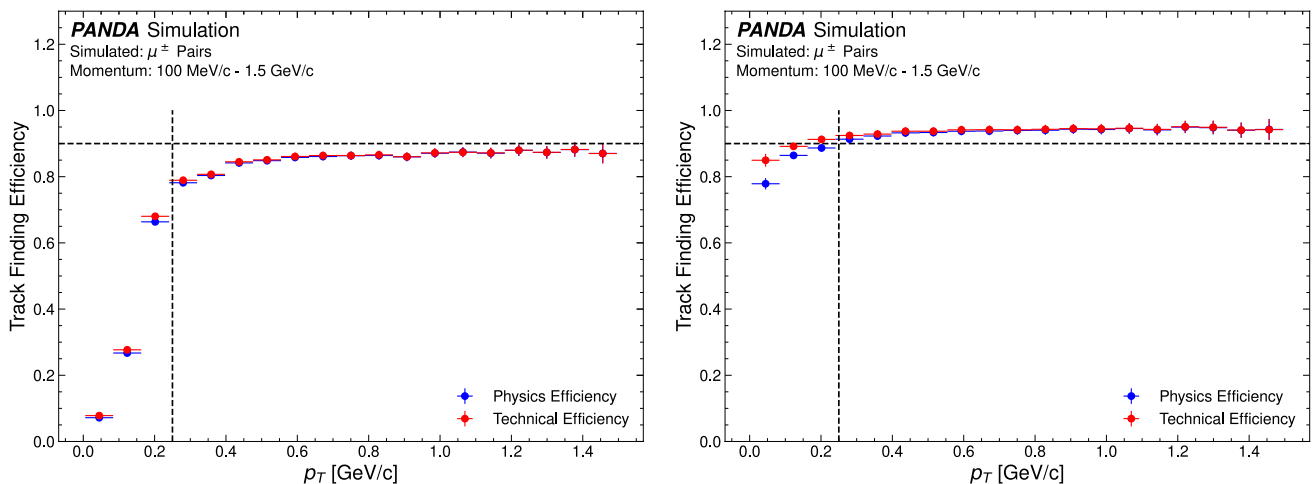


Fig. 11 Tracking efficiencies as a function of p_T for FCN (left) and IGNN (right) with reference lines at $p_T = 0.25$ GeV/c (vertical) and $\epsilon = 90\%$ efficiency (horizontal)

means that in the lab system of PANDA, the fast $\bar{\Lambda}$ antihyperon goes into the acceptance of the Forward Spectrometer while the Λ hyperons are slow and decay inside the Target Spectrometer. Hence, the daughters of the Λ give rise to hits in the STT and can be reconstructed with our algorithm. Of special interest is the daughter pions from Λ decays, since they often have very low momenta (see left panel of Fig. 12). In the decay, antiproton and protons ($\bar{p}, p$) take the larger share of the momentum, while only a small fraction goes to the pions (π^+, π^-). These pions are challenging to reconstruct due to the high curvature of their trajectories and their high probability of intersecting with the trajectories of other particles. Furthermore, due to the relatively long lifetime of the Λ hyperons, they are expected to decay far from the beam-target interaction point (see right panel of Fig. 12). This makes the $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ reaction an important benchmark for track reconstruction algorithms with PANDA.

For the hyperon reconstruction, we applied the same GDL pipeline as in the muon case, except with one small difference: in the graph construction stage, the heuristic method for building nodes and edges was not restricted to adjacent sectors. Since the $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ reaction contains fewer particles per event compared to the $5\mu^+\mu^-$ case, and since we expect many pions to be emitted at extremely low p_T [13], removing this condition increases the amount of data in each event. After edge labelling, we tested $2 \cdot 10^3$ events during inference. In the graph segmentation stage, we used the DBSCAN method with $\epsilon_{db} = 0.15$ after rescanning this parameter, similar to Fig. 9, and a minimum number of samples to be two to

find the connected components from the test events. Using the same track evaluation criteria as in previous cases, the tracking efficiencies, ghost rate and clone rate are obtained as in Table 2.

The physics tracking efficiency $\epsilon_{phys.}$ is about 90%. However, the technical efficiency $\epsilon_{tech.}$ is significantly higher, and the ghost rate and clone rate are significantly lower compared to the muon case. This performance gain is understood as each event has fewer particles than the muon case and there are fewer track intersections.

Similar to the muon case, we analyse the tracking efficiencies as a function of p_T as shown in Fig. 13: the number of particles (left) and tracking efficiencies (right).

In a large fraction of the events, there are particles with momenta as low as $p_T < 0.25$ GeV/c, which are captured in the magnetic field of the PANDA solenoid and therefore remain inside the detector. These particles are primarily pions and form an enhancement at low p_T in the left panel of Fig. 13. Protons generally have much larger momenta, and the different kinematics of protons and pions manifest in different lab polar angles. As a result, the track lengths will differ and thus the reconstruction probability. In the right panel of Fig. 13, we see that the physical track efficiency $\epsilon_{phys.}$ has a structure where low-momentum pions and high-momentum protons are relatively well reconstructed. At the same time, there is a dip in the efficiency in the intermediate region. However, the technical efficiency has no such structure, which leads to the conclusion that the intermediate p_T region has a high content of non-reconstructable tracks.

Next, we investigate tracking efficiency as a function of the radial position of decay vertices (d_0), as shown in Fig. 14.

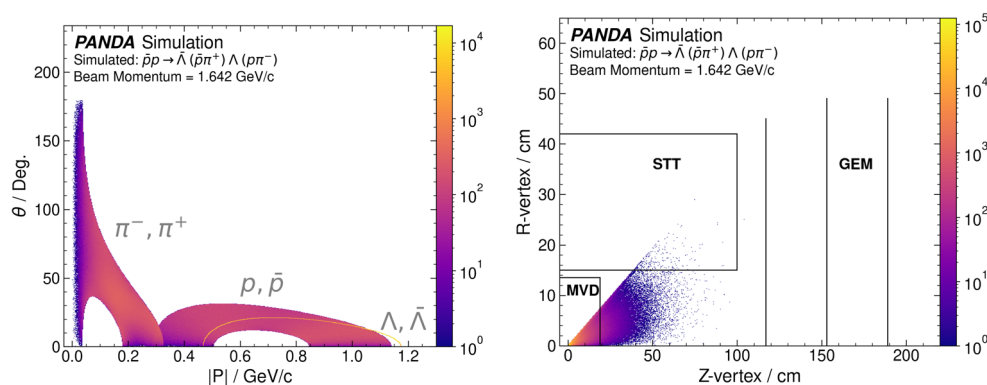


Fig. 12 Kinematic properties of $\bar{p}p \rightarrow \bar{\Lambda}\Lambda \rightarrow \bar{p}\pi^+p\pi^-$ reaction: (left) MC $|p|$ vs θ distribution, (right) MC decay vertex distribution of Λ and $\bar{\Lambda}$

Table 2 Tracking efficiencies, ghost rate, and clone rate for hyperons

N_t	N_r	MF [%]	$\epsilon_{phys.}$ [%]	$\epsilon_{tech.}$ [%]	Ghost Rate [%]	Clone Rate [%]
7	5	> 50	89.6 ± 0.6	97.1 ± 0.6	0.5 ± 0.6	4.9 ± 0.1

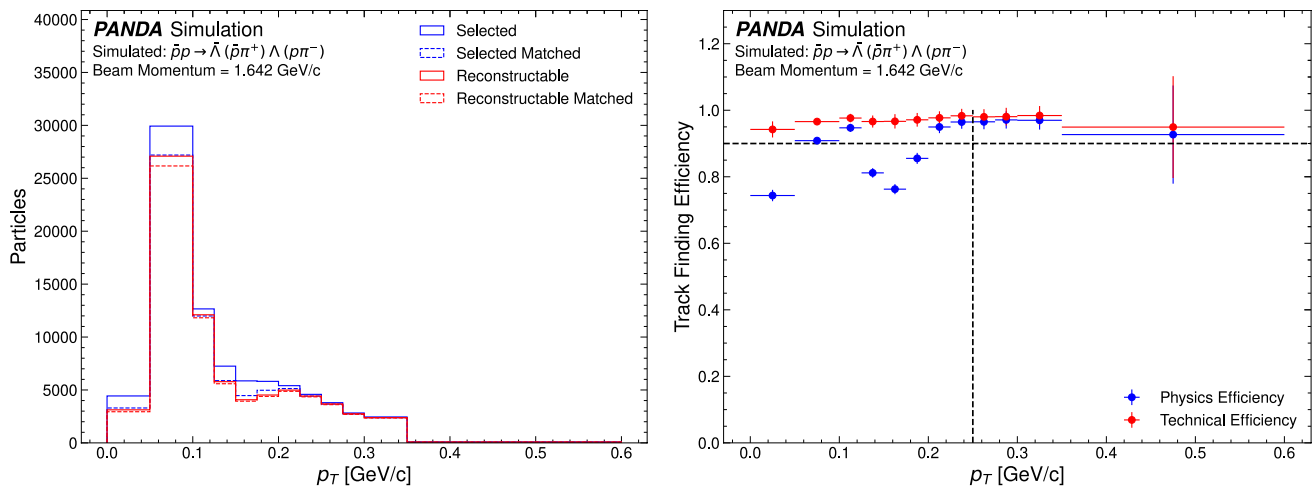


Fig. 13 Number of particles (left), and tracking efficiencies (right) as a function of p_T with reference lines at $p_T = 0.25$ GeV/c (vertical) and $\epsilon = 90\%$ efficiency (horizontal)

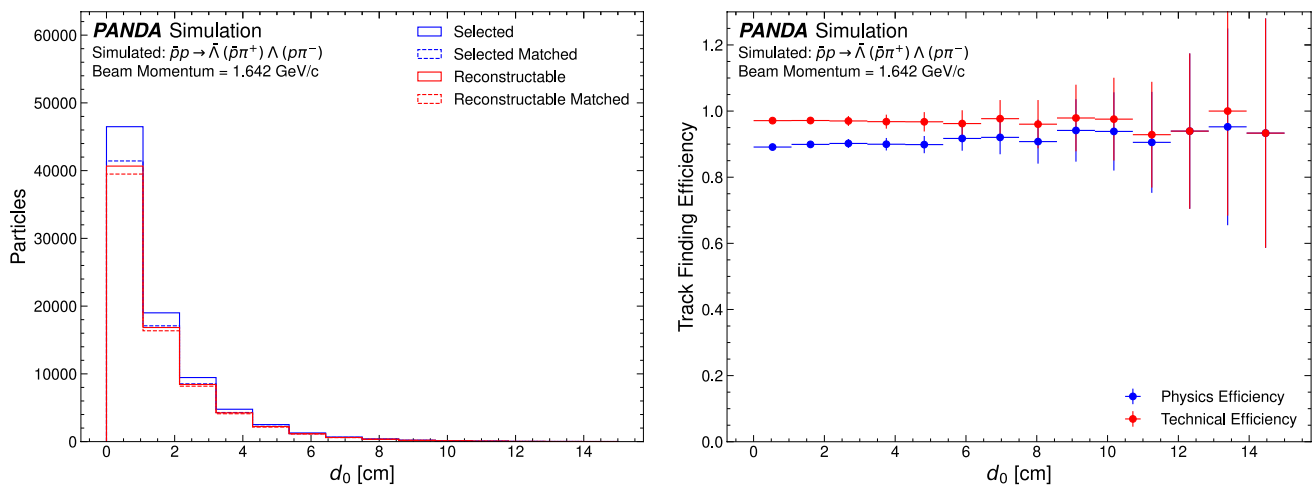


Fig. 14 Number of particles (left), and tracking efficiencies (right) as a function of d_0

Most particles (protons, pions) are generated close to the interaction point, however, a considerable fraction is generated up to 14 cm from the beam-target interaction point. From Fig. 14, we conclude that our algorithm also performs well for these kinds of tracks: the physical and technical track efficiencies are above 90% for both pions and protons over the full d_0 range. Moreover, the technical efficiency is about 97%. This is an important finding as most heavier hyperons (Ξ , Ω , etc.) decay at $d_0 < 15$ cm [13].

Conclusion

In this work, we have successfully applied machine learning to reconstruct particle trajectories in a hadron physics experiment. Our work shows the first use of GNN-based

track reconstruction in the straw tube detector with non-Euclidean geometry. It is found that GDL models give promising results, giving overall tracking efficiency $\geq 90\%$. It can reconstruct pions with p_T as low as ~ 0.05 GeV/c and protons with p_T as low as ~ 0.1 GeV/c. Further studies show that this method also works well for reconstructing particles with secondary decay vertices up to at least $d_0 = 14$ cm away from the IP in the radial direction. Beyond $d_0 = 14$ cm, our simulated data contains no decaying Λ hyperons. This is an important result as heavier hyperons, such as Ξ^- and Ω^- , are expected to decay through intermediate Λ hyperons with the Λ decay vertices mostly occurring less than 15 cm from the IP. These results are promising for the hyperon reconstruction at PANDA and demonstrate the virtues of GNNs for the specific challenges of particle tracking in hadron physics experiments.

Acknowledgements The authors would like to thank Nikolai in der Wiesche for productive discussions. This project has received funding from the Knut and Alice Wallenberg Foundation and the Swedish Research Council (Sweden).

Author contributions A.A. is the main analyst of this work who developed the ML code used for this work from an open-source project, performed physics simulation and analysis to produce results and plots, and compiled them in this manuscript. X.J. provided assistance related to ML code and contributed to editing. T.S. provided assistance related to the PandaRoot framework. J.T. provided feedback on the analysis and also contributed to the editing. M.P. supervised the analysis and contributed to the editing. K.S. supervised the project, provided funding, and contributed to the drafting and editing of this manuscript. All authors have reviewed the manuscript.

Funding Open access funding provided by Uppsala University.

Data availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no Conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Lutz MFM, et al (2009) Physics Performance Report for PANDA: Strong Interaction Studies with Antiprotons. [arXiv:0903.3905](https://arxiv.org/abs/0903.3905) [hep-ex]
2. Ablikim M et al (2019) Polarization and Entanglement in Baryon-Antibaryon Pair Production in Electron-Positron Annihilation. *Nature Phys* 15:631–634. <https://doi.org/10.1038/s41567-019-0494-8>. [arXiv:1808.08917](https://arxiv.org/abs/1808.08917) [hep-ex]
3. Ablikim M et al (2022) Probing cp symmetry and weak phases with entangled double-strange baryons. *Nature* 606(7912):64–69. <https://doi.org/10.1038/s41586-022-04624-1>. [arXiv:2105.11155](https://arxiv.org/abs/2105.11155) [hep-ex]
4. Ablikim M et al (2019) Complete measurement of the λ electromagnetic form factors. *Phys Rev Lett* 123(12):122003. <https://doi.org/10.1103/PhysRevLett.123.122003>. [arXiv:1903.09421](https://arxiv.org/abs/1903.09421) [hep-ex]
5. Schönning K et al (2023) Production and decay of polarized hyperon-antihyperon pairs*. *Chin Phys C* 47(5):052002. <https://doi.org/10.1088/1674-1137/acc790>. [arXiv:2302.13071](https://arxiv.org/abs/2302.13071) [hep-ph]
6. Crede V, Roberts W (2013) Progress towards understanding baryon resonances. *Rept Prog Phys* 76:076301. <https://doi.org/10.1088/0034-4885/76/7/076301>. [arXiv:1302.7299](https://arxiv.org/abs/1302.7299) [nucl-ex]
7. Thiel A et al (2022) Light Baryon Spectroscopy. *Prog Part Nucl Phys* 125:103949. <https://doi.org/10.1016/j.pnpnp.2022.103949>. [arXiv:2202.05055](https://arxiv.org/abs/2202.05055) [nucl-ex]
8. Tolos L, Fabbietti L (2020) Strangeness in Nuclei and Neutron Stars. *Prog Part Nucl Phys* 112:103770. <https://doi.org/10.1016/j.pnpnp.2020.103770>. [arXiv:2002.09223](https://arxiv.org/abs/2002.09223) [nucl-ex]
9. Barucca G et al (2021) Study of excited ξ baryons with the $\overline{p}$ anda detector. *Eur Phys J A* 57(4):149. <https://doi.org/10.1140/epja/s10050-021-00444-5>. [arXiv:2012.01776](https://arxiv.org/abs/2012.01776) [hep-ex]
10. Singh B (2016) Study of doubly strange systems using stored antiprotons. *Nucl Phys A* 954:323–340. <https://doi.org/10.1016/j.nuclphysa.2016.05.014>
11. Barucca G et al (2021) The potential of λ and ξ^- studies with panda at fair. *Eur Phys J A* 57(4):154. <https://doi.org/10.1140/epja/s10050-021-00386-y>. [arXiv:2009.11582](https://arxiv.org/abs/2009.11582) [hep-ex]
12. Barucca G et al (2021) Panda phase one. *Eur Phys J A* 57(6):184. <https://doi.org/10.1140/epja/s10050-021-00475-y>. [arXiv:2101.11877](https://arxiv.org/abs/2101.11877) [hep-ex]
13. Abazov V, et al (2023) Hyperon signatures in the PANDA experiment at FAIR. [arXiv:2304.11977](https://arxiv.org/abs/2304.11977) [nucl-ex]
14. Singh B et al (2016) Feasibility studies of time-like proton electromagnetic form factors at $\overline{p}$ anda at fair. *Eur Phys J A* 52(10):325. <https://doi.org/10.1140/epja/i2016-16325-5>. [arXiv:1606.01118](https://arxiv.org/abs/1606.01118) [hep-ex]
15. Barucca G et al (2021) Feasibility studies for the measurement of time-like proton electromagnetic form factors from $\overline{p}p \rightarrow \mu^+\mu^-$ at $\overline{p}$ anda at fair. *Eur Phys J A* 57(1):30. <https://doi.org/10.1140/epja/s10050-020-00333-3>. [arXiv:2006.16363](https://arxiv.org/abs/2006.16363) [hep-ex]
16. Barucca G et al (2019) Precision resonance energy scans with the panda experiment at fair: sensitivity study for width and line-shape measurements of the $\chi(3872)$. *Eur Phys J A* 55(3):42. <https://doi.org/10.1140/epja/i2019-12718-2>. [arXiv:1812.05132](https://arxiv.org/abs/1812.05132) [hep-ex]
17. Ikegami-Andersson W et al (2021) A Generalized Approach to Longitudinal Momentum Determination in Cylindrical Straw Tube Detectors. *Comput Softw Big Sci* 5(1):18. <https://doi.org/10.1007/s41781-021-00064-0>. [arXiv:2012.06442](https://arxiv.org/abs/2012.06442) [physics.ins-det]
18. Taylor J et al (2024) 4D Track Reconstruction on Free-Streaming Data at PANDA at FAIR. *Comput Softw Big Sci*. <https://doi.org/10.1007/s41781-021-00064-0>
19. Aliche A (2023) Development of fast track finding algorithms for densely packed straw tube trackers and its application to (Ξ^-) (1820) hyperon reconstruction for the PANDA experiment. PhD thesis, Ruhr-Universität Bochum. <https://doi.org/10.13154/294-10449>
20. Amrouche S et al (2019). The Tracking Machine Learning Challenge Accuracy Phase. https://doi.org/10.1007/978-3-030-29135-8_9
21. Amrouche S et al (2023) The Tracking Machine Learning Challenge: Throughput Phase. *Comput Softw Big Sci* 7(1):1. <https://doi.org/10.1007/s41781-023-00094-w>. [arXiv:2105.01160](https://arxiv.org/abs/2105.01160) [cs.LG]
22. Ju X et al (2021) Performance of a Geometric Deep Learning Pipeline for HL-LHC Particle Tracking. *Eur Phys J C* 81(10):876. <https://doi.org/10.1140/epjc/s10052-021-09675-8>. [arXiv:2103.06995](https://arxiv.org/abs/2103.06995) [physics.data-an]
23. Caillou S et al (2022). ATLAS ITk Track Reconstruction with a GNN-based pipeline. <https://doi.org/10.5281/zenodo.8119762>
24. Ekawa H et al (2023) Development of machine learning analyses with graph neural network for the wasa-frs experiment. *Eur Phys J A* 59(5):103. <https://doi.org/10.1140/epja/s10050-023-01016-5>
25. Jia X et al (2024) BESIII Track Reconstruction Algorithm based on Machine Learning. *EPJ Web Conf* 295:09006. <https://doi.org/10.1051/epjconf/202429509006>
26. Reuter L, et al (2024) End-to-End Multi-Track Reconstruction using Graph Neural Networks at Belle II [arXiv:2411.13596](https://arxiv.org/abs/2411.13596) [physics.ins-det]

27. Gutbrod H (2006) FAIR Baseline Technical Report - Volume 1 Executive Summary, p. 92. GSI, Darmstadt. Downloadfile ohne Cover, Druckversion beinhaltet Band 2-6 auf CD [Volumes 1,2,3a,3b,4,5,6 liegen auch als separate Files im GSI Institutional Repository]. <https://repository.gsi.de/record/54062>
28. Gutbrod H (2006) (ed.): FAIR Baseline Technical Report, Volume 2 Accelerator and Scientific Infrastructure, p. 738. GSI, Darmstadt. Downloadfile ohne Cover. <https://repository.gsi.de/record/54068>
29. Spataro S (2011) Panda collaboration): the pandaroot framework for simulation, reconstruction and analysis. J Phys Conf Ser 331(3):032031. <https://doi.org/10.1088/1742-6596/331/3/032031>
30. Al-Turany M et al (2012) The FairRoot framework. J Phys Conf Ser 396:022001. <https://doi.org/10.1088/1742-6596/396/2/022001>
31. Brun R, Rademakers F (1997) ROOT: An object oriented data analysis framework. Nucl Instrum Meth A 389:81–86. [https://doi.org/10.1016/S0168-9002\(97\)00048-X](https://doi.org/10.1016/S0168-9002(97)00048-X)
32. HEP.TrkX Collaboration: HEP Advanced Tracking Algorithms with Cross-cutting Applications (Project HEP.TrkX). <https://heptrkx.github.io/>
33. Farrell S et al (2017) The HEP.TrkX Project: Deep Neural Networks for HL-LHC Online and Offline Tracking. EPJ Web Conf 150:00003. <https://doi.org/10.1051/epjconf/20171500003>
34. Tsaris A et al (2018) The HEP.TrkX Project: Deep Learning for Particle Tracking. J Phys Conf Ser 1085(4):042023. <https://doi.org/10.1088/1742-6596/1085/4/042023>
35. Farrell S, et al (2018) Novel Deep Learning Methods for Track Reconstruction. <https://arxiv.org/abs/1810.06111>
36. Exa.TrkX Collaboration: HEP Advanced Tracking Algorithms at the Exascale (Project Exa.TrkX). <https://exatrnx.github.io/>
37. Ju X, et al (2020) Graph Neural Networks for Particle Reconstruction in High Energy Physics Detectors. <https://arxiv.org/abs/2003.11603>
38. Choma N, et al (2020) Track Seeding and Labelling with Embedded-space Graph Neural Networks. <https://arxiv.org/abs/2007.00149>
39. Biscarat C et al (2021) Towards a Realistic Track Reconstruction Algorithm Based on Graph Neural Networks for the HL-LHC. EPJ Web Conf 251:03047. <https://doi.org/10.1051/epjconf/202125103047>
40. Pata J, et al (2021) MLPF: Efficient Machine-learned Particle-flow Reconstruction using Graph Neural Networks Eur Phys J C <https://doi.org/10.1140/epjc/s10052-021-09158-w>
41. Ju X et al (2021) Performance of a geometric deep learning pipeline for hl-lhc particle tracking. Eur Phys J C 81(10):876–14. <https://doi.org/10.1140/epjc/s10052-021-09675-8>
42. DeZoort G, et al (2021) Charged Particle Tracking via Edge-Classifying Interaction Networks. Computing and Software for Big Science 5(1) <https://doi.org/10.1007/s41781-021-00073-z>
43. Caillou S et al (2024) Physics Performance of the ATLAS GNN4ITk Track Reconstruction Chain. EPJ Web Conf 295:03030. <https://doi.org/10.1051/epjconf/202429503030>
44. Baranov D, et al (2019) Graph Neural Network Application to the Particle Track Reconstruction for Data from the GEM Detector. AIP Conf. Proc. 2163(1) <https://doi.org/10.1063/1.5130100>
45. Esmail WAM (2022) Deep learning for track finding and the reconstruction of excited hyperons in proton induced reactions. doctoralthesis, Ruhr-Universität Bochum, Universitätsbibliothek. <https://doi.org/10.13154/294-8563>
46. Akram A, Ju X (2022) Track Reconstruction using Geometric Deep Learning in the Straw Tube Tracker (STT) at the PANDA Experiment. <https://arxiv.org/abs/2208.12178>
47. Akram A (2023) Towards Realistic Hyperon Reconstruction in PANDA: From Tracking with Machine Learning to Interactions with Residual Gas. PhD thesis, Uppsala U
48. Ekawa H et al (2023) Development of machine learning analyses with graph neural network for the wasa-frs experiment. Eur Phys J A 59(5):103. <https://doi.org/10.1140/epja/s10050-023-01016-5>
49. Jia X et al (2024) BESIII Track Reconstruction Algorithm based on Machine Learning. EPJ Web of Conf 295:09006. <https://doi.org/10.1051/epjconf/202429509006>
50. Reuter L, et al (2024) End-to-End Multi-Track Reconstruction using Graph Neural Networks at Belle II [arXiv:2411.13596](https://arxiv.org/abs/2411.13596) [physics.ins-det]
51. Hewes J et al (2021) Graph Neural Network for Object Reconstruction in Liquid Argon Time Projection Chambers. EPJ Web Conf 251:03054. <https://doi.org/10.1051/epjconf/202125103054>
52. Aurisano A, et al (2024) NuGraph2: A Graph Neural Network for Neutrino Physics Event Reconstruction. <https://arxiv.org/abs/2403.11872>
53. HEP ML Community: A Living Review of Machine Learning for Particle Physics. <https://iml-wg.github.io/HEPML-LivingReview/>
54. Akram A (2025) Code for the paper “application of geometric deep learning for tracking of hyperons in a straw tube detector”. Zenodo. <https://doi.org/10.5281/zenodo.15024201>
55. Ryd A, et al (2005) EvtGen: A Monte Carlo Generator for B-Physics
56. Agostinelli S et al (2003) GEANT4-a simulation toolkit. Nucl Instrum Meth A 506:250–303. [https://doi.org/10.1016/S0168-9002\(03\)01368-8](https://doi.org/10.1016/S0168-9002(03)01368-8)
57. Battaglia PW, et al (2016) Interaction Networks for Learning about Objects, Relations and Physics. <https://arxiv.org/abs/1612.00222>
58. Battaglia PW, et al (2018) Relational Inductive Biases, Deep Learning, and Graph Networks. <https://arxiv.org/abs/1806.01261>
59. Ester M et al (1996) A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Kdd 96:226–231
60. Committee CGTL (2017) Technical Design Report for the ATLAS Inner Tracker Pixel Detector. Technical report, CERN, Geneva . <https://doi.org/10.17181/CERN.FOZZ.ZP3Q> . <https://cds.cern.ch/record/2285585>
61. Johansson T et al (1999) Results from PS185. Nucl Phys A 655:173–178. [https://doi.org/10.1016/S0375-9474\(99\)00198-0](https://doi.org/10.1016/S0375-9474(99)00198-0)
62. Barnes PD et al (2000) High statistics measurements of the anti-p p → anti-Lambda Lambda and anti-p p → Lambda Sigma0 + c.c. reactions at threshold. Phys Rev C 62:055203. <https://doi.org/10.1103/PhysRevC.62.055203>
63. Paschke KD et al (2006) Experimental determination of the complete spin structure for anti-p p → anti-Lambda Lambda at p(anti-p) = 1.637-GeV/c. Phys Rev C 74:015206. <https://doi.org/10.1103/PhysRevC.74.015206>. [arXiv:nucl-ex/0605025](https://arxiv.org/abs/nucl-ex/0605025)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.