

Particle Tracking in the NA62 Experiment

Leonardo Plini,^{a,b,*} Gemma Tinti,^a Tommaso Spadaro,^b Indro Spinelli^c and Fabio Galasso^c

^a*National Institute of Nuclear Physics-National Laboratories of Frascati,
Via Enrico Fermi 54, Rome, Italy*

^b*Department of Computer, Control and Management Engineering, Sapienza University of Rome,
Via Ariosto 25, Rome, Italy*

^c*Department of Computer Science, Sapienza University of Rom,
Via Regina Elena 295, Rome, Italy*

*E-mail: leonardo.plini@lnf.infn.it, gmtinti@lnf.infn.it,
tommaso.spadaro@lnf.infn.it, indro.spinelli@uniroma1.it,
galasso@di.uniroma1.it*

The NA62 experiment at the Super Proton Synchrotron at CERN aims to measure the branching ratio of the ultra-rare channel $K^+ \rightarrow \pi^+ \nu \bar{\nu}$ with a predicted BR = $(8.6 \pm 0.42) \times 10^{-11}$. The GigaTracker, a four-layer silicon-pixel tracker, measures the momentum and direction of incoming hadron beam particles at a rate of 750 MHz. The experiment's success relies on effectively managing pile-up, temporal matching and vertex reconstruction between beam kaons and decay pions, a challenging task due to the current combinatorial track-building approach. This research integrates machine learning best practices in the current modeling strategies applied in the NA62 experiment. In this study, we extensively evaluate different deep-learning architectures for particle tracking, showing the most effective approaches. In particular, we designed and implemented three distinct strategies based on Multi-Layer Perceptron, Transformer and Graph Neural Network. All methods were evaluated using efficiency, purity and fake-track rate. The most promising ones were further assessed in an aggregation-free setup, where clustering based on ground truth information was removed to test their robustness in a real-world scenario.

*Third Italian Workshop on the Physics at High Intensity (WIFAI2024)
12-15 November 2024
Bologna, Italy*

*Speaker

1. Introduction

Machine-learning techniques, such as boosted decision trees, support vector machines and Multi-layer Perceptrons (MLPs), have been used in experimental particle physics since the 1980s, enabling significant methodological advancements [1, 2]. Their recent success in diverse high-energy physics (HEP) applications underscores their potential to enhance data-driven analyses and reconstruction tasks [2–4]. Moreover, the ever-increasing data volumes in HEP experiments necessitate efficient computational strategies, where ML-based methods capitalize on the parallel processing capabilities of GPUs to optimize reconstruction workflows.

Among these techniques, MLPs stand out as one of the earliest machine learning architectures, offering a simple solution for constrained particle tracking problems. More recently, Transformers, widely recognized as a state-of-the-art approach across various machine learning domains, have been explored for particle tracking in HEP, demonstrating promising capabilities [5, 6].

In particle physics, experimental data are often sparse in time and space, making them ill-suited for traditional representations such as regular grids (images) or ordered sequences. Unlike MLPs, Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs), which struggle to fully exploit the relational nature of particle interactions, Graph Neural Networks (GNNs) provide a more natural and effective framework for track reconstruction. In fact, Geometric Deep Learning generalizes machine learning techniques to non-Euclidean domains, including sets, manifolds, and graphs, allowing for a more structured representation of complex data [7]. Within this paradigm, GNNs, first introduced in [8], are designed to process graph-based data, where nodes represent entities and edges define their relationships. For comprehensive reviews, see [9–11].

The ability of GNNs to model relational structures has led to their application across a range of high-energy physics tasks, including particle tracking [12, 13], jet classification [14], clustering [13], vertexing [15] and pileup mitigation [16].

In this paper, we conduct a comprehensive comparative study of MLPs, Transformers and GNNs applied to particle tracking using NA62 Monte Carlo simulations of the hits left on the four Gigatracker Silicon pixel detectors [17, 18], evaluating their respective advantages and limitations. Additionally, we demonstrate the robustness of the best-performing models in an aggregation-free setup, where clustering based on ground truth information is removed.

2. Related Works

Early particle tracking methods relied on deterministic curve fitting and interpolation, which were sensitive to noise. A major breakthrough came with the Kalman filter [19], which provided an optimal recursive estimation framework and was widely adopted in HEP.

Recent advancements have demonstrated the potential of Transformer-based architectures in particle tracking [5, 6], showcasing their ability to achieve remarkable efficiency in this domain. In particular, Van Stroud et al. [5] proposed an architecture that integrates a Transformer-based hit filtering network with a MaskFormer reconstruction model, enabling the simultaneous optimization of hit assignments and the estimation of charged particle properties.

The HEP.TRKX pilot project [20] marked a pivotal step in applying machine learning techniques to HEP tracking, laying the foundation for later efforts such as the Exa.TrkX project [21]. Similarly,

the GNN4ITkG pipeline [22, 23] leverages geometric deep learning models to enhance tracking performances, initially developed using the trackML dataset [24], a simulation inspired by LHC tracking detectors. This pipeline has demonstrated significant improvements in track-reconstruction efficiency and, since then, GNNs have gained increasing interest in the HEP community due to their ability to effectively represent particle-tracking data [15, 25]. Various tasks have been explored, including edge classification [12, 13], node regression and classification [16] and graph pooling [26]. Liu et al. [27] introduced a hierarchical GNN, a novel GNN variant for particle tracking that constructs high-level track representations while assigning space points to specific tracks. This approach allows for disconnected space points to be linked to the same trajectory and enables multiple tracks to share a common space point. DeZoort et al. [28] developed a physics-inspired interaction network GNN, specifically designed for tracking in pileup conditions, addressing challenges expected in the High-Luminosity Large Hadron Collider (HL-LHC). In our prior work [29], we demonstrated the effectiveness of GNNs for particle tracking in the NA62 experiment, achieving state-of-the-art results on Monte Carlo simulations.

3. Method

In this section, we will introduce the problem, then the data as well as the requirements of the experiment. Finally, we show how we applied the architectures to the NA62 tracking problem.

Given a set of events $E = \{e\}_i^N$ where each event consists of a variable number of particles k that leave one or more hits across the four stations of the GTK detectors, the objective is to reconstruct the correct tracks connecting these hits. To refine the input data, true simulation information is used to merge multiple hits originating from the same particle within the same detector. This is achieved by averaging their position and time, effectively eliminating the impact of clustering errors while isolating those inherent to the architecture. A track is considered valid if respects one of the following condition: hits belonging to GTK0, GTK2 and GTK3 or hits coming from GTK1, GTK2 and GTK3 or if it is composed of four hits, each from a different GTK station.

In the MLP-based approach, all possible valid hit combinations are evaluated and a score is assigned to each potential track. However, this method requires the combinatorial generation of all possible track alternatives for each true trajectory, leading to an overwhelming number of negative samples. This imbalance significantly affects the dataset, increasing the likelihood of false positives.

In contrast, both the Transformer-based and GNN-based approaches adopt a different problem formulation that results in better tracks. Instead of evaluating all possible track combinations, these models first process all hits using either a Transformer encoder or a GNN, and then a MLP is applied exclusively to admissible hit pairs to determine the likelihood of a valid connection. This strategy drastically reduces memory consumption and computational complexity, as it limits the number of candidate tracks requiring evaluation. Finally, a post-processing step is applied to ensure track consistency, selecting only the best entering and best exiting edge for each hit, thereby enforcing the constraint that a hit belongs to a single track, as described in [29]. To evaluate whether the Transformer and GNN can inherently handle the clustering phase by suppressing less probable connections, the aggregation constraint is removed. Standard techniques, such as batch normalization, were applied to ensure training stability. ¹

¹The models were implemented on an RTX 3060Ti Trio with 8 GB paired with a Ryzen 7 3700X CPU.

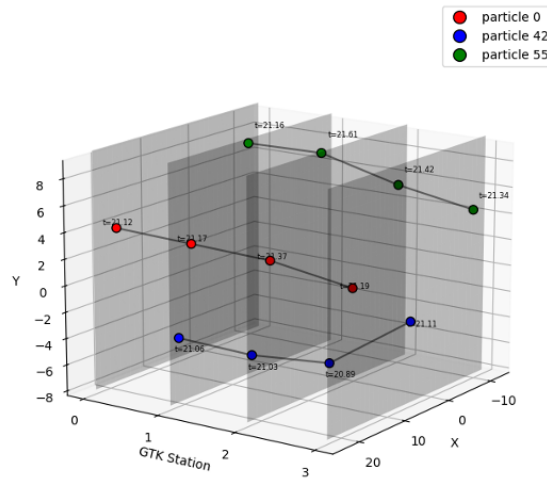


Figure 1: Graph Neural Networks are able to predict the tracks by formulating the problem as a binary classification of the edge of the graph

4. Results

The dataset is generated from a Monte Carlo simulation of the NA62 experiment which is able to reproduce the hit rate and spatial occupancy of the beam particles. To assess the model’s ability to accurately reconstruct tracks, we evaluate its efficiency and its purity. The efficiency is measured as the fraction of correctly predicted tracks relative to the total number of true tracks, while purity is defined as the fraction of correctly predicted tracks relative to the total number of predicted tracks. In machine learning terms, efficiency corresponds to recall, whereas purity aligns with precision. Whenever possible, models are designed to maintain an efficiency of at least 94%. Additionally, the fake-track rate is introduced to quantify errors, calculated as the ratio of incorrectly predicted tracks to the total number of true tracks.

Model	Efficiency	Purity	Fake-Track Rate
MLP	92.3%	70.06%	39.45%
Transformer	95.94%	98.62%	1.33%
Graph Neural Network	94.78%	99.78%	0.21%

Table 1: Comparison of different models in terms of Efficiency, Purity and Fake-Track Rate.

Table 1 highlights the superiority of Transformers and GNNs over MLPs, particularly in terms of purity and fake-track rate. These architectures benefit from a global event perspective, effectively integrating information from all hits. However, no definitive best model emerges between the two, and further studies using different Monte Carlo samples are required to determine the most suitable architecture for particle tracking in NA62.

Table 2 demonstrates that both architectures can inherently handle clustering without explicit constraints. Moreover, the increased number of available hits leads to a higher number of true positives, enhancing efficiency while maintaining competitive levels of purity and fake-track rate.

Model	Efficiency	Purity	Fake-Track Rate
Transformer	95.62%	99.03%	0.94%
Graph Neural Network	97.01 %	99.39 %	0.59 %

Table 2: Performance comparison of Graph Neural Network and Transformer models without clustering

5. Conclusion

This study demonstrates the effectiveness of GNNs and Transformers in reconstructing particle tracks from simulated NA62 data using maximum-score edge selection. Several alternative architectures were evaluated to quantitatively assess their strengths and limitations.

Future research will focus on testing these models on real experimental data, evaluating their robustness to domain shifts and generalization to different datasets. Additionally, it would be valuable to quantify the impact of imperfect clustering algorithms and explore whether these architectures can simultaneously perform hit clustering and track reconstruction.

Overall, the ability of these models to achieve high purity and low fake-track rates while maintaining strong efficiency highlights their potential for particle-tracking applications.

Future research will involve testing these models on real experimental data, with a focus on their robustness and generalization. We also plan to quantify the impact of imperfect clustering algorithms and explore whether these architectures can integrate hit clustering and track reconstruction within a single end-to-end framework.

Acknowledgments

This work was carried out while Leonardo Plini was enrolled in the Italian National Doctorate on Artificial Intelligence run by Sapienza University of Rome. The present work was completed thanks to the dedication of the whole NA62 Collaboration in developing the full simulation and later in providing results from the off-line data processing.

References

- [1] B. Denby. Neural networks and cellular automata in experimental high energy physics. *Computer Physics Communications*, 49(3):429–448, 1988.
- [2] Alexander Radovic, Mike Williams, David Rousseau, Michael Kagan, Daniele Bonacorsi, Alexander Himmel, Adam Aurisano, Kazuhiro Terao, and Taritree Wongjirad. Machine learning at the energy and intensity frontiers of particle physics. *Nature*, 560(7716):41–48, August 2018.
- [3] Bourilkov D. Machine and deep learning applications in particle physics. *International Journal of Modern Physics A*, 34(35):1930019, December 2019.
- [4] M. Feickert and B. Nachman. A living review of machine learning for particle physics, 2021.
- [5] Samuel Van Stroud, Philippa Duckett, Max Hart, Nikita Pond, Sébastien Rettie, Gabriel Facini, and Tim Scanlon. Transformers for charged particle track reconstruction in high energy physics. *arXiv preprint arXiv:2411.07149*, 2024.

- [6] Sascha Caron, Nadezhda Dobreva, Antonio Ferrer Sánchez, José D Martín-Guerrero, Uraz Odyurt, Roberto Ruiz de Austri Bazan, Zef Wolfs, and Yue Zhao. Trackformers: In search of transformer-based particle tracking for the high-luminosity lhc era. *arXiv preprint arXiv:2407.07179*, 2024.
- [7] M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, July 2017.
- [8] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, Jan 2009.
- [9] Z. Zhang, P. Cui, and W. Zhu. Deep learning on graphs: A survey, 2020.
- [10] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, January 2021.
- [11] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun. Graph neural networks: A review of methods and applications, 2021.
- [12] S. Farrell et al. Novel deep learning methods for track reconstruction, 2018.
- [13] X. Ju and et al. Graph neural networks for particle reconstruction in high energy physics detectors, 2020.
- [14] I. Henrion, J. Brehmer, J. Bruna, K. Cho, K. Cranmer, G. Louppe, and G. Rochette. Neural message passing for jet physics. 2017.
- [15] J. Shlomi, S. Ganguly, E. Gross, K. Cranmer, Y. Lipman, H. Serviansky, H. Maron, and N. Segol. Secondary vertex finding in jets with neural networks. *The European Physical Journal C*, 81(6), June 2021.
- [16] J. A. Martinez, O. Cerri, M. Pierini, M. Spiropulu, and J. Vlimant. Pileup mitigation at the large hadron collider with graph neural networks, 2019.
- [17] E. Cortina Gil. et al. The beam and detector of the na62 experiment at cern. *Journal of Instrumentation*, 12(05):P05025, may 2017.
- [18] Rinella et al. The na62 gigatracker: a low mass high intensity beam 4d tracker with 65 ps time resolution on tracks. *Journal of Instrumentation*, 14(07):P07010–P07010, July 2019.
- [19] A new approach to linear filtering and prediction problems. 2002.
- [20] A. Tsaris and et al. The hep.trkx project: Deep learning for particle tracking. *Journal of Physics: Conference Series*, 1085(4):042023, sep 2018.
- [21] Exa.TrkX Collaboration. Exa.trkx. <https://exatrnx.github.io/>.
- [22] S. Caillou, P. Calafiura, S. Farrell, X. Ju, D. Murnane, C. Rougier, J. Stark, and A. Vallier. ATLAS ITk Track Reconstruction with a GNN-based pipeline. Technical report, CERN, Geneva, 2022.
- [23] X. et al. Ju. Performance of a geometric deep learning pipeline for hl-lhc particle tracking. *The European Physical Journal C*, 81(10), October 2021.
- [24] M. Kiehn and et al. The trackml high-energy physics tracking challenge on kaggle. *EPJ Web of Conferences*, 214:06037, 01 2019.
- [25] S. Thais and et al. Graph neural networks in particle physics: Implementations, innovations, and challenges, 2022.
- [26] V. Mikuni and F. Canelli. ABCNet: an attention-based method for particle tagging. *The European Physical Journal Plus*, 135(6):463, 2020.

- [27] R. Liu, P. Calafiura, S. Farrell, X. Ju, D. Murnane, and T. Pham. Hierarchical graph neural networks for particle track reconstruction, 2023.
- [28] G. DeZoort, S. Thais, J. Duarte, V. Razavimaleki, M. Atkinson, I. Ojalvo, M. Neubauer, and P. Elmer. Charged particle tracking via edge-classifying interaction networks. *Computing and Software for Big Science*, 5(1), November 2021.
- [29] Leonardo Plini, Gemma Tinti, Tommaso Spadaro, and Fabio Galasso. Graph neural networks for particle tracking in na62 experiment. *Il Nuovo Cimento C*, -, 2025. Open Access.