Exploring particle identification in ACTS using temporal information

Sanira Turganova
Supervisors: Paul Gessinger, Pierfrancesco Butti

Abstract
This report presents a comprehensive enhancement of the simulation and reconstruction pipeline within CERN's ACTS framework to allow exploring particle identification. The project implements substantial improvements to the ACTS workflow using the Open Data Detector (ODD) geometry. The key advancements include: (1) development of a randomized ParticleGun capable of generating mixed particle events ($\pi^{\pm}$, $K^{\pm}$, $p^{\pm}$) for more realistic simulation scenarios; (2) implementation of a unified ROOT-based output format that consolidates tracking parameters and hypothesis information into a single file, achieving a 9-fold reduction in output file count; and (3) introduction of a time-aware χ^2 selection algorithm that leverages detector timing information to study particle identification accuracy across 10000 simulated events.

Keywords
ACTS, Kalman Filter, ODD, 4D Tracking

1 Introduction

In high-energy physics experiments tracking the trajectories of charged particles through complex detector systems is a fundamental computational challenge. **A Common Tracking Software**, developed at **CERN**, provides a modular and experiment-independent toolkit for reconstructing particles trajectories with high efficiency and precision.

As the LHC transitions to its High-Luminosity phase, the demands on tracking software grow increasingly stringent requiring both algorithmic improvements and computational optimizations.

This study was conducted as part of the Summer Student Program, focusing on enhancing the ACTS's framework's capabilities. This report documents significant improvements to ACTS's simulation and reconstruction pipeline, specifically targeting three areas of development that were identified as limitations in current workflow.

The first area of improvement addresses particle generation. The single-particle type events were insufficient for this study, so randomized ParticleGun capable of generating events with multiple particle species ($\pi^{\pm}$, $K^{\pm}$, $p^{\pm}$), was implemented.

The second enhancement focuses on data storage optimization. Existing ACTS testing workflow typically produced multiple output files for different reconstruction hypotheses, creating organizational challenges and I/O inefficiencies. This work introduces a unified ROOT-based storage format that consolidates all track parameters, truth matching information, and particles' PDG codes into a single structured file. This enhancement not only simplifies data management but also improves processing efficiency for downstream analysis.

The third advancement involves reconstruction hypothesis selection. Leveraging the time resolution capabilities of the Open Data Detector (ODD), I have implemented a χ^2-based selection algorithm that incorporates timing information from detector layers. This approach demonstrates progressive improvement in particle identification accuracy as more detector layers contribute timing information.

2 Open Data Detector

The Open Data Detector (ODD) is a parametric detector model used as a development testbed for the ACTS. Designed with a focus on realism and computational performance, this parametric detector model combines essential features of modern silicon-based detectors while maintaining the simplicity required for rapid prototyping and testing.

The ODD implements a cylindrical geometry with multiple sensitive layers, carefully balancing spatial resolution and material budget to approximate real detector conditions [1].

Its performance characteristics, which are configurable, make it valuable for HL-LHC studies, as it can simulate spatial resolution comparable to actual tracking detectors while maintaining manageable computational requirements. Configurable time resolution also enables 4D tracking studies, where the time dimension can provide crucial information for particle identification.

The ODD geometry features a comprehensive silicon tracking system with a pixel-strip hybrid architecture. Detector components are arranged in cylindrical barrel layers and forward/backward endcap disks, implemented using the DD4hep geometry description toolkit for accurate material modeling. The ODD incorporates detailed support structures and cabling to realistically simulate material interactions in particle tracking scenarios.

For the purposes of this study, the ODD was configured with performance parameters typical of modern detectors. Spatial resolution was simulated by applying Gaussian smearing. A time resolution of $30ps$ was applied to all hits to enable 4D tracking studies.

As experiments push towards high luminosities – testbeds like the ODD become vital for developing the precise and computationally efficient tracking algorithms required for future detectors.

3 Kalman Filter

The Kalman filter is an optimal recurrent data processing algorithm widely used for state estimation of dynamic systems under uncertainty [3]. It was developed by Rudolf Kalman in 1960.

The Kalman filter allows you to estimate the state of a system using noisy measurements and a model of the system. It works in two stages:

1. Prediction – predicting the next state based on a model of the system
2. Update - refinement of the prediction using new measurements

The Kalman filter works with linear dynamic systems described by two equations:

– Equation of state (system model):

$$x_k = F_{k-1}x_{k-1} + \omega_{k-1} \tag{1}$$

where x_k – state vector at time k, F_{k-1} – transition matrix, ω_{k-1} – process noise. (random disturbance)

– Measurement equation:

$$m_k = H_k x_k + \epsilon_k \tag{2}$$

where m_k – measurements vector, H_k – measurements matrix, ϵ_k – measurements noise.

The noises are assumed to be *independent, Gaussian, and have zero mean*.

The algorithm is implemented using the following steps:

1. Prediction

$$x_k^{k-1} = F_{k-1}x_{k-1} \tag{3}$$

$$C_k^{k-1} = F_{k-1}C_{k-1}F_{k-1}^T + Q_{k-1} \tag{4}$$

2. Correction

$$K_k = C_k^{k-1}H_k^T(H_kC_k^{k-1}H_k^T + V_k)^{-1} \tag{5}$$

$$x_k = x_k^{k-1} + K_k(m_k - H_kx_k^{k-1}) \tag{6}$$

$$C_k = (I - K_kH_k)C_k^{k-1} \tag{7}$$

where K_k – Kalman gain matrix, C_k – covariance matrix of estimation errors, V_k – measurement noise covariance, Q_k – covariance of process noise.

Smoothing is used to estimate the state in the past. All measurements, including future ones, are used for this purpose. Smoothing is especially used in track detectors, where accuracy is important throughout the particle's trajectory. Smoothing is implemented in the following formulas:

$$x_k^n = x_k + A_k(x_{k+1}^n - x_{k+1}^k) \tag{8}$$

$$A_k = C_kF_k^T(C_{k+1}^k)^{-1} \tag{9}$$

The Kalman filter is the computational core of the track reconstruction procedure in ACTS, sequentially processing measurements to construct optimal estimates of trajectory parameters in accordance with the minimum variance criterion.

4 Track reconstruction within the ACTS framework

The process of reconstructing tracks of charged particles in the ACTS is typically structured as a multi-stage pipeline of sequential data processing. While the standard pipeline includes steps for clustering and pattern recognition, this study utilized a simplified, truth-informed workflow to focus on performance of track fitting and selection algorithms(see Figure 1).

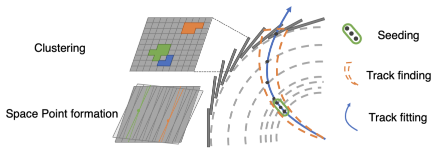


Figure 1: Track reconstruction pipeline [2]

The first step is modeling and digitizing the data. Particles are generated using ParticleGun, then their passage through the detector material and magnetic field is simulated, taking into account multiple scattering and ionization losses.

The second step is space point formation. Based on the smeared hit positions, three dimensional spatial points are created. This procedure provides some kind of building blocks for track reconstruction.

The third step is seednig. In the standard ACTS pipeline these points to find groups that lie on potential trajectories. However, for this specific study, this step was bypassed. Instead, a truth-based seeding approach was used, where initial track seeds were generated directly from the simulated particle information. This allowed us to isolate the performance of the later stages of the pipeline.

The fourth step is track finding. Using the seeds as a starting point, the algorithms search for all spatial points belonging to a given trajectory. Similarly, for this work, truth-informed track finding was employed to ensure correct associations, focusing the analysis on fitting and selection.

The final step is track fitting. This stage involves refining the track parameters using fitting algorithms, in particular the Kalman filter, which optimally takes into account measurement errors and physical effects like multiple scattering.

5 Results and discussion

5.1 Event Generator Upgrade

One of the first results is the randomization of the ParticleGun. Initially, the simulations used the standard ParticleGun generator, which generated events containing only one particle type per event set (e.g., 10,000 events containing only π^-). While useful for debugging and calibration, this approach fails to represent the mixed particle environments encountered in real experiments. To enable more comprehensive testing of reconstruction and identification algorithms under more varied conditions, a randomized ParticleGun generator was implemented.

The system randomly selects a particle type to generate for each event from a pre-defined pool. The particle pool and the probabilities of their generation were defined as follows:

- $\pi^{\pm}$: $1/3$
- $K^{\pm}$: $1/3$
- $p^{\pm}$: $1/3$

This mixture, while still simplified, provides more challenging and representative testbed than single-species events by requiring algorithms to handle the different signatures of particles. A branch was also added to the output ROOT tree to store the PDG codes of all simulated particles for truth matching (Figure 2).

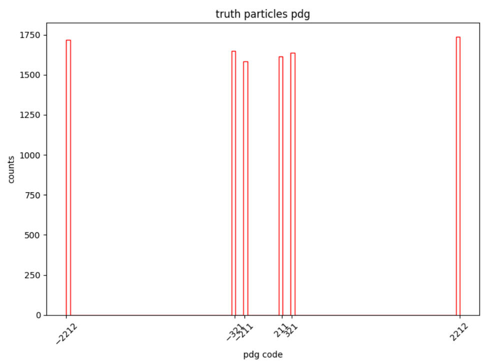


Figure 2: PDG codes of simulated particles

Here ± 2212 refers to proton and antiproton, ± 321 refers to postive and negative kaon, ± 211 refers to positive and negative pion.

5.2 Unification of output data and expansion of the ROOT file structure

To manage the output data needed for this study, the output system has been completely redesigned. Instead of generating many separate files for different hypotheses or particle types, all data is now written to a single, unified ROOT file.

The previous approach was cumbersome and complicated subsequent analysis by increasing the number of files to handle and the risk of errors when synchronizing events across them. The new design simplifies the data management process. As described previously, each event contains one of three types of particles. For each event, the reconstruction algorithm (Kalman filter) is run three times – assuming that the track was left by a pion, a kaon, or a proton. This results in three separate sets of track parameters (hypotheses) per event. Crucially, the output for a single event is now organized within the ROOT tree such that the three hypothesis entries share the same event number. This allows for straightforward grouping and comparison of the different fitted tracks for the same simulated particle during the analysis. In addition to the PDG codes of the truth particles, a new branch was added containig the absolute values of the PDG codes for the hypotheses themselves (Figure 3).

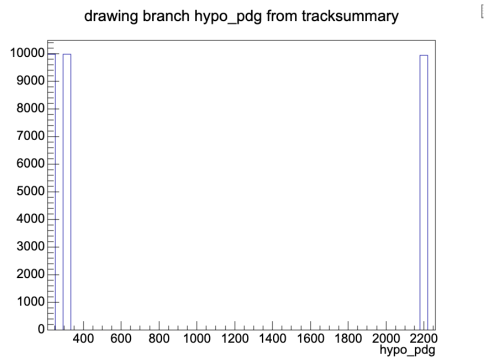
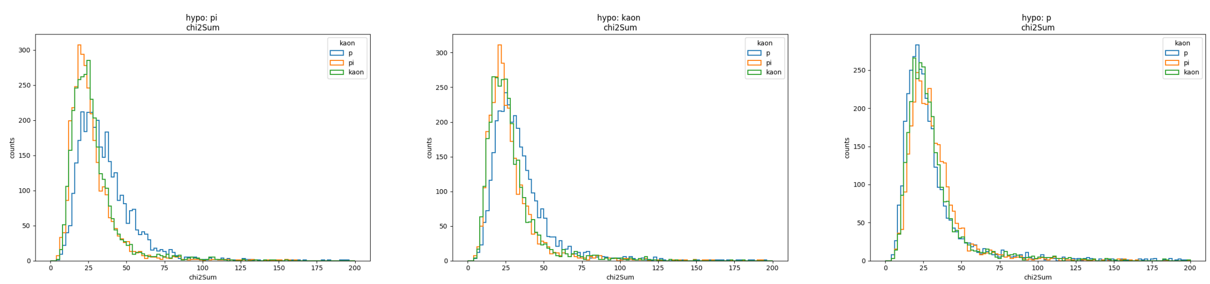


Figure 3: PDG codes of hypotheses

5.3 Algorithm for choosing the correct hypothesis

To identify particles, an algorithm was implemented that selects the most probable hypothesis for each event based on an analysis of the track quality. Since three sets of parameters are calculated for each track, a criterion is needed to determine which hypothesis is most likely. A standard method in tracking is to use the χ^2 match between the track and the detector measurements. The distributions of the χ^2 values for the three hypotheses are shown in the figures below.



(a) (b) (c)

Figure 4: χ^2 distributions under (a) π, (b) K, (c) p hypotheses

The algorithm goes through all events and for each track compares the χ^2 value for all three hypotheses. The hypothesis with the minimum χ^2 value is selected as the best. The PDG code of this hypothesis is then compared with the PDG code of the generated particle (truth) to determine whether the identification was correct.

For the most visual assessment of the algorithm's accuracy, the confusion matrix normalized to the true particle was constructed using the scikit-learn library (Figure 5).

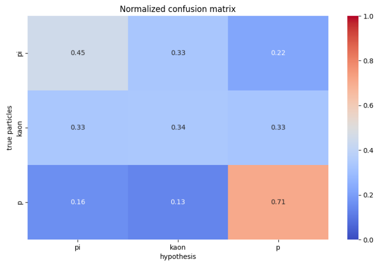


Figure 5: Confusion matrix of the particle type identification

5.4 Adding time information to the detector's substructures

As can be seen from the confusion matrix, the accuracy of the algorithm is quite low for kaons and pions. The correct identification rate was 50.29%. This is primarily due to the fact that the time information was added only to the pixels and χ^2 distributions overlap each other. However, the algorithm demonstrates a higher success rate for proton identification ($\sim 70\%$), a result of the larger mass difference that makes protons easier to distinguish even with limited information.

The underlying principle for using timing in particle identification (PID) is that a particle's time of flight between detector layers provides crucial information about its velocity. When combined with the momentum measured from the trajectory's curvature in the magnetic field, this velocity measurement helps determine the particle's mass, which is the key discriminating factor.

To evaluate how the availability of timing informationimpacts performance, a series of simulations were conducted. In these tests, time information with a fixed resolution was progressively added to different combinations of the ODD subsystems, which include pixels, long strips, and short strips positioned at increased radii(Figure 6). This approach allowed for a direct analysis of how the classification performance improved as more timing hits were incorporated along the particle's trajectory.

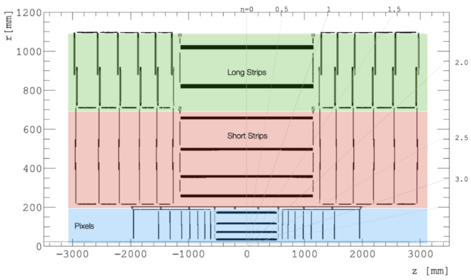
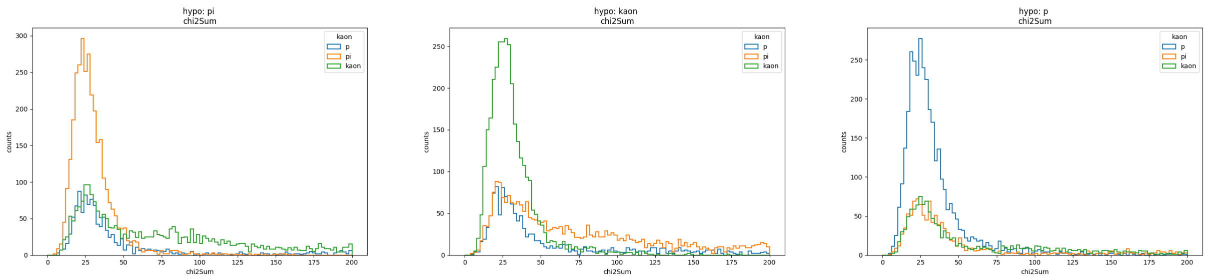


Figure 6: Schematic view of the Open Data Detector

5.4.1 *Extending time information to Pixels and Long Strips*

To improve particle identification, the configuration was enhanced by instrumenting the long strips with timing capabilities, in addition to the pixels. This created a baseline for measuring the time of flight between two subsystems separated by a significant radius.

The resulting χ^2 distributions below show better separation between the hypotheses. The correct identification rate increased to 80.76%, which clearly shows the importance of **measuring the time interval**. The confusion matrix (Figure 8) also shows a significant improvement in the diagonal elements.



(a) (b) (c)

Figure 7: χ^2 distributions under (a) π, (b) K, (c) p hypotheses with time information in Pixels and Long Strips

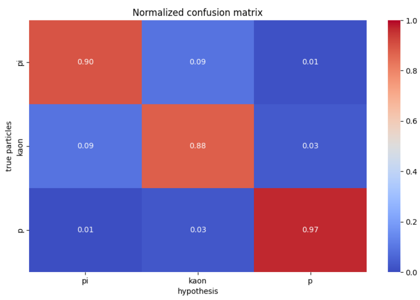


Figure 8: Confusion matrix for time information added to Pixels and Long Strips

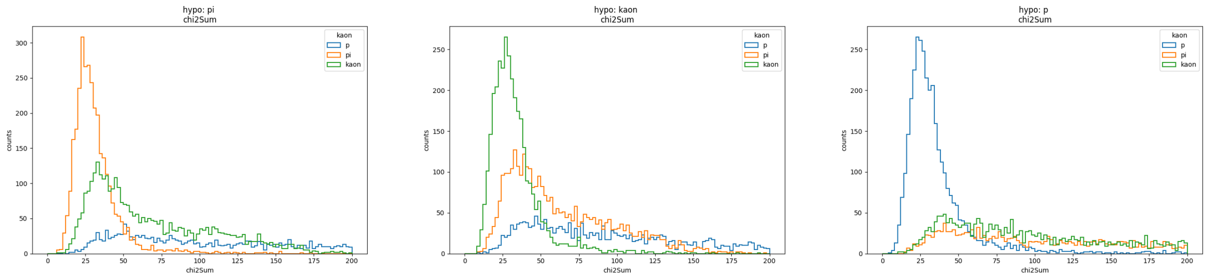
5.4.2 Extending time information to Pixels and Short Strips

At this configuration, timing information was activated for the short strip detector, alongside the already instrumented pixels. The χ^2 distributions show even better separation between the hypotheses compared to the previous configuration.

Although the distance is less than in the long strip configuration, the identification accuracy was 89.96%. This is explained by the fact that these two detector subsystems are the innermost. The track in this section has the least uncertainty, since it has not yet accumulated significant errors. The confusion matrix (Figure 10) shows further improvement in identification accuracy.

5.4.3 Extending time information to all of the ODD subsystems

The final stage assumes the presence of time information on all detector subsystems. The χ^2 distributions show the best separation, and the confusion matrix demonstrates the maximum identification accuracy. The highest identification accuracy of 91.41% was achieved using time information across the entire track.



(a) (b) (c)

Figure 9: χ^2 distributions under (a) π, (b) K, (c) p hypotheses with time information in Pixels and Short Strips

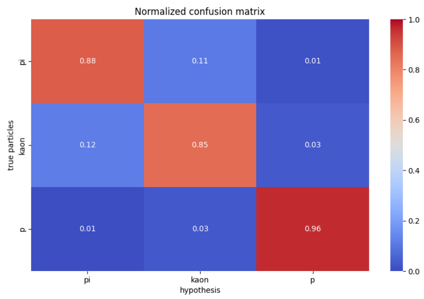
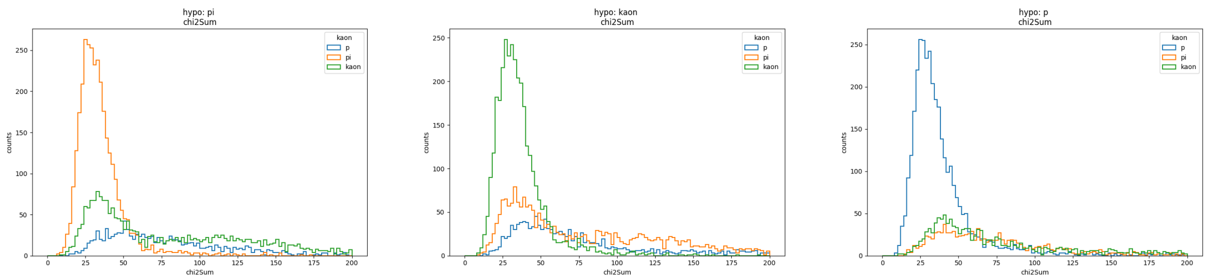


Figure 10: Confusion matrix for time information added to Pixels and Short Strips



(a) (b) (c)

Figure 11: χ^2 distributions under (a) π, (b) K, (c) p hypotheses with time information in all detector elements

6 Conclusion and Summary

The work carried out during the Summer Student Program served as a first exploration of particle identification techniques within the ACTS framework, establishing a new testbed for future studies. The project implemented a randomized ParticleGun generating a mixture of particles. The output data format was optimized – a single file storing the entire set of track parameters for each hypothesis.

The most significant results were obtained in the study of the influence of time information on the accuracy of identification. The step-by-step addition of time information to various combinations of the ODD subsystems showed that the presence of time measurements in only one component is insufficient for reliable separation of hypotheses. The maximum correct identification rate (91.41%) was achieved using time information from all detector elements.

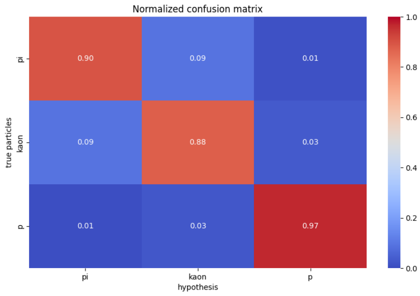


Figure 12: Confusion matrix for time information added to all subsystems

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Paul Gessinger, Pierfrancesco Butti, and CERN, for the incredible opportunity to work under their guidance at CERN. Your support and guidance have taught me so much and have been essential to my personal and professional development. I truly appreciate everything you have done.

Bibliography

[1] M. Frank; et al. Dd4hep: A detector description toolkit for high energy physics experiments. *Journal of Physics: Conference Series, Volume 513, Issue 2*, 2014.

[2] Paul Gessinger; et al. The acts project: track reconstruction software for hl-lhc and beyond. *EPJ Web of Conferences 245*, 2020.

[3] R. Frühwirth. Application of kalman filtering to trak and vertex fitting. *Nuclear Instruments and Methods in Physics Research*, 1987.