

Track Reconstruction Using Cellular Automata for the High Luminosity LHC & AI-Based QA Monitoring for the CBM experiment

Sachin Gupta, Ruhr University Bochum

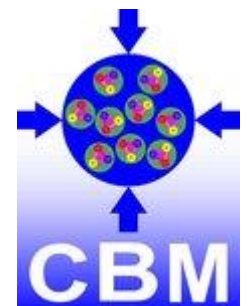
Sachin.Gupta@ruhr-uni-bochum.de, Workplace : GSI, Darmstadt.



UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

RUHR
UNIVERSITÄT
BOCHUM

RUB



Introduction

- Birthplace : Nainital, Uttarakhand (India).
- Education :
 - BSc Physics, St. Stephen's college (Delhi University) 2017-2020
 - Research assistant (Bangalore) : 2021
 - Heidelberg University: MSc physics, 2022-2024
 - DESY Summer student 2023
 - Internship RU Bochum (Jun 2025-Nov 2025)
 - PhD (Dec 2025-present)
- Interests : Musical Instruments (Flute, Guitar), Philosophy and Psychology.



Nainital



Delhi



Bangalore



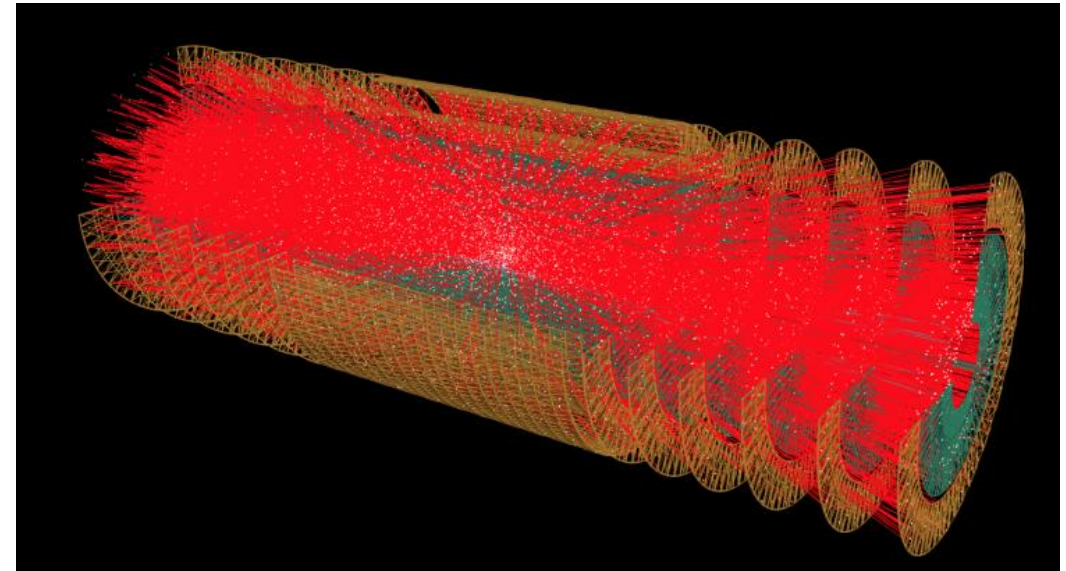
Heidelberg



RU Bochum 2

Motivation

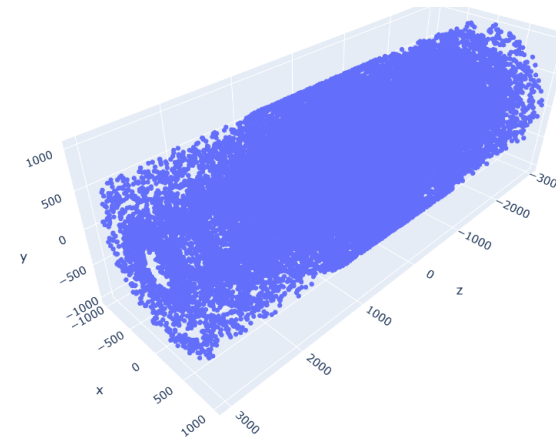
- CERN is currently preparing for HL-LHC phase that aims to achieve unprecedented luminosity ($7.5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$).
- The average number of proton-proton collisions per bunch crossing will be $\langle \mu \rangle = 200$ (pile-up).
- More collisions \rightarrow More charged particles are produced \rightarrow Tracking becomes challenging.
- State of the art algorithm for Tracking is Combinatorial Kalman Filter (Iterative, not easily parallelizable).
- This thesis evaluates the performance of parallelizable tracking algorithm based on the TrackML dataset.



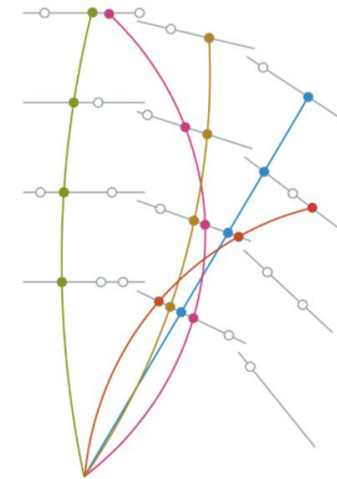
An Event Display of the TrackML Dataset

Track Reconstruction

- Getting the properties of charged particles (momentum, charge, vertex etc.) from the recorded detector measurements i.e. 3d space points or hits.
- Tracking involves two main stages:
 - **Track Finding** : Grouping hits into track candidates which potentially originate from a charged particle.
 - **Track Fitting** : Process of estimating a particle's parameters, covariance matrix and track quality χ^2 .

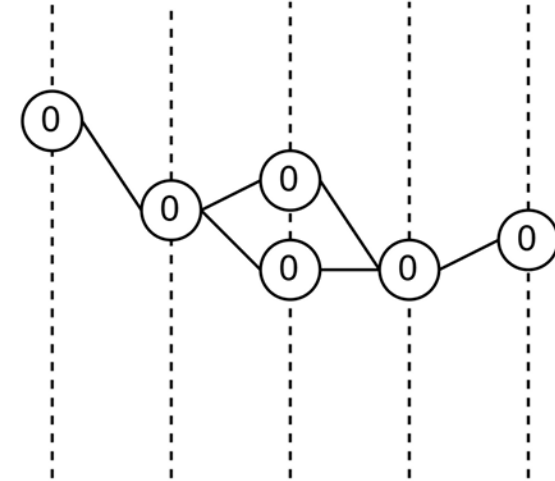


≈ 100,000 Hits
per event



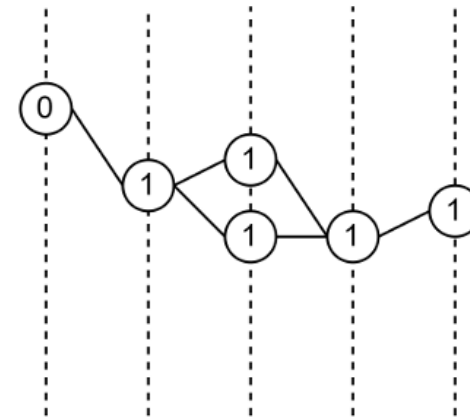
Track Finding with CA (CATS)

- **CATS** : Cellular automata for tracking in silicon for the HERA-B vertex detector (originally uses doublet as a cell).
- **Example** – 5 detector layers with some hit configuration.
- **Cells** – Each “hit” is described as a cell and initialized with a value of zero.
- **Neighbour** – “All hits in the previous layer” – (fully connected graph) with no neighbors within the layer

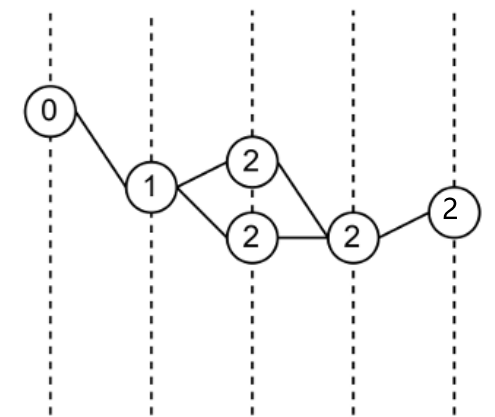


Rule and Evolution

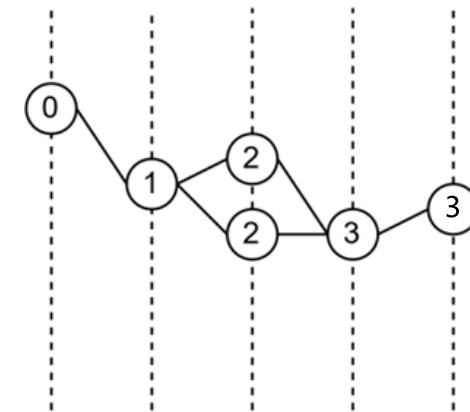
- Rule - At each iterative step :
 - Each cell looks at its left neighbour (if any).
 - Find the neighbour with maximum value.
 - New value = $\max(\text{neighbour values}) + 1$
 - The evolution is stopped when cells value do not change.



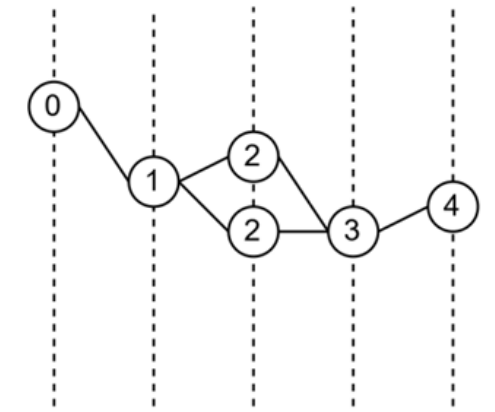
(a) Iteration = 1



(b) Iteration = 2



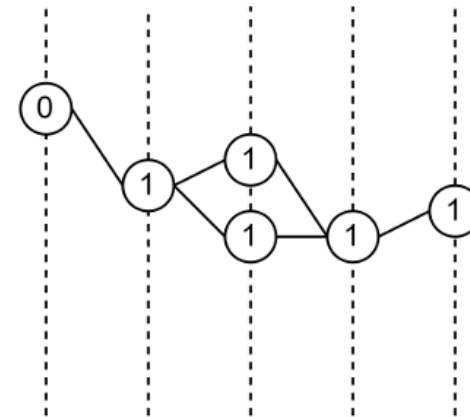
(c) Iteration = 3



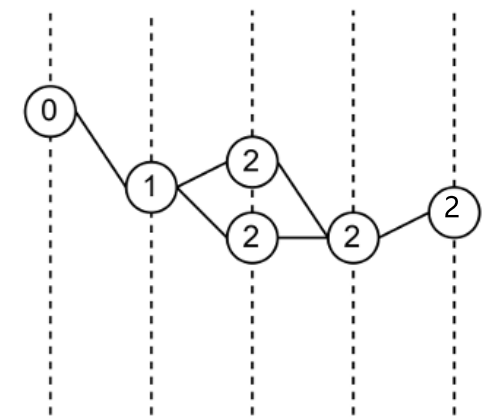
(d) Iteration = 4

Rule and Evolution

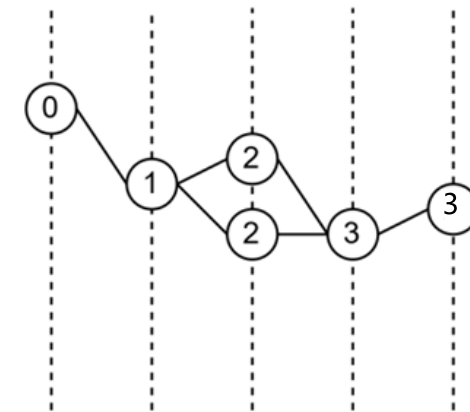
- Rule - At each iterative step :
 - Each cell looks at its left neighbour (if any).
 - Find the neighbour with maximum value.
 - New value = $\max(\text{neighbour values}) + 1$
 - The evolution is stopped when cells value do not change.
- Track candidates are formed from the **final configuration** (iteration = 4).
- From the cell with highest value, tracks are collected in order $4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0$.
- This example gives 2 possible tracks.
- Track candidates are further analysed by χ^2 value.



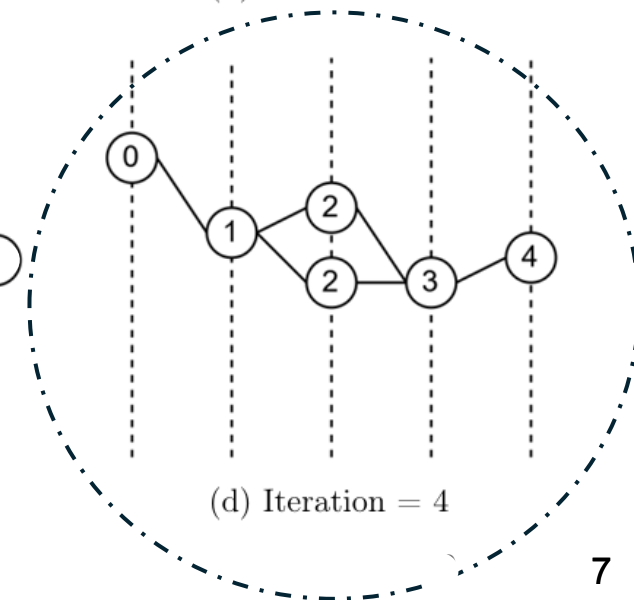
(a) Iteration = 1



(b) Iteration = 2



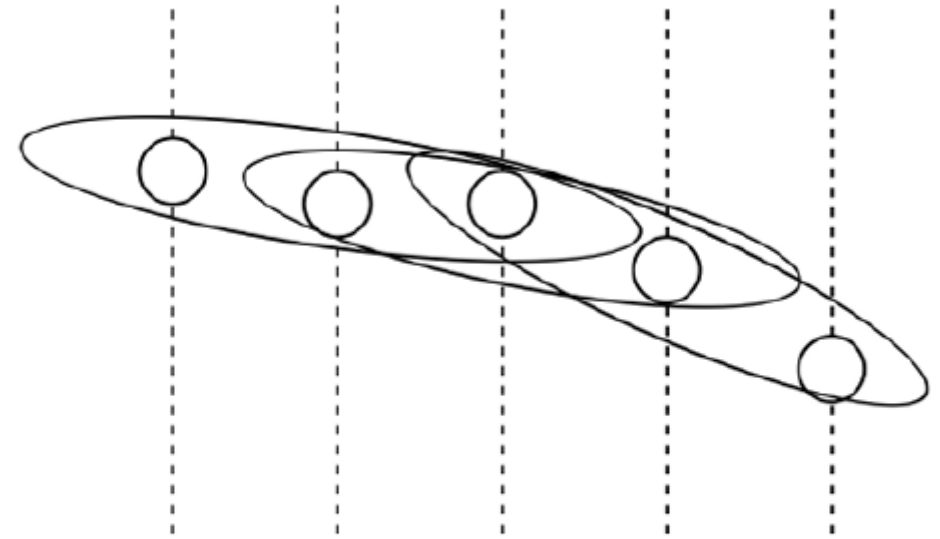
(c) Iteration = 3



(d) Iteration = 4

Triplet as a Cell

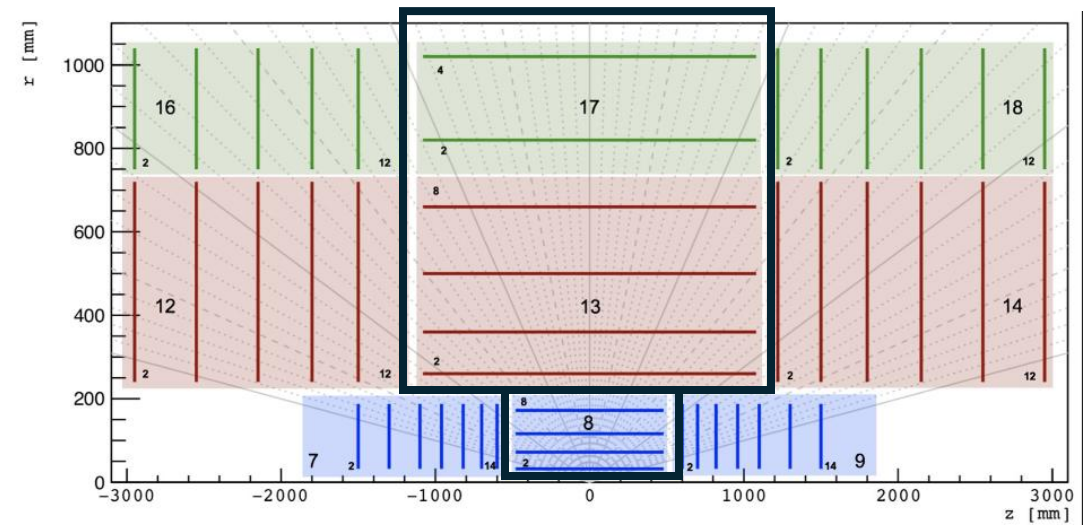
- Under pile-up conditions, the total number of hits per event is expected to be $\approx \mathcal{O}(10^5)$
- Treating hit as a cell will be computationally challenging and memory expensive.
- Triplet of hits can be used as a “cell” and track is formed by linking triplets.
- Early quality cut on triplets further reduces the combinatorics.



Track visualised as joining three triplets. Two consecutive triplets share two hits.

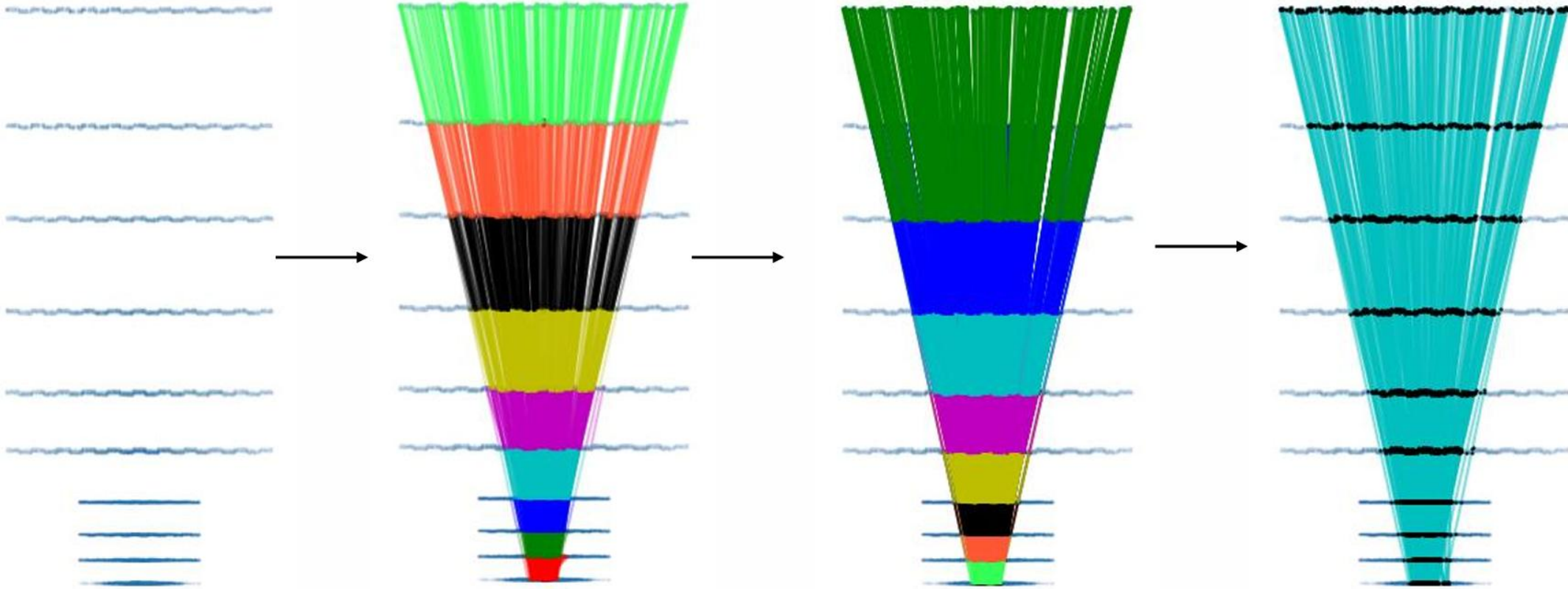
TrackML Dataset

- Dataset competition launched in 2018 on Kaggle to promote ML algorithms for tracking.
- Simulated data based on conditions expected for HL-LHC, with #hits $\approx (100k - 200k)$ per event
- MS thesis project :
 - CA is implemented in the barrel region of the TrackML detector.
 - Target signal : Particle's originating near the luminous region, $p_T \geq 1$ GeV, depositing hits in all barrel layers.



CA for the TrackML Dataset

Longitudinal (r-z) plane view



Barrel region hits

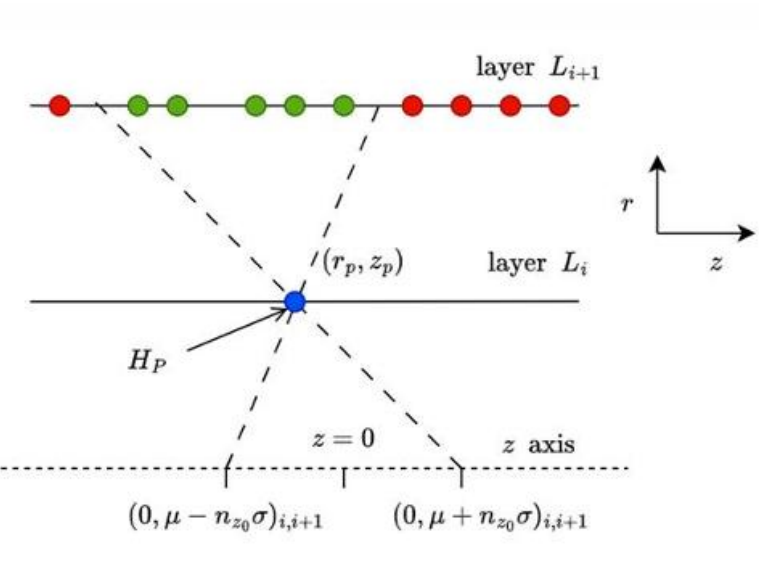
Doublet formation

Triplet Formation

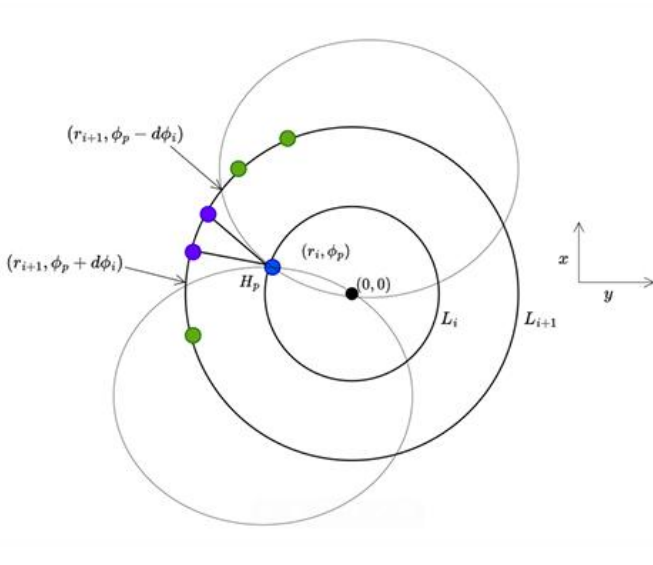
Tracks from CA+ ambiguity resolution

Cuts for Doublets and Triplets

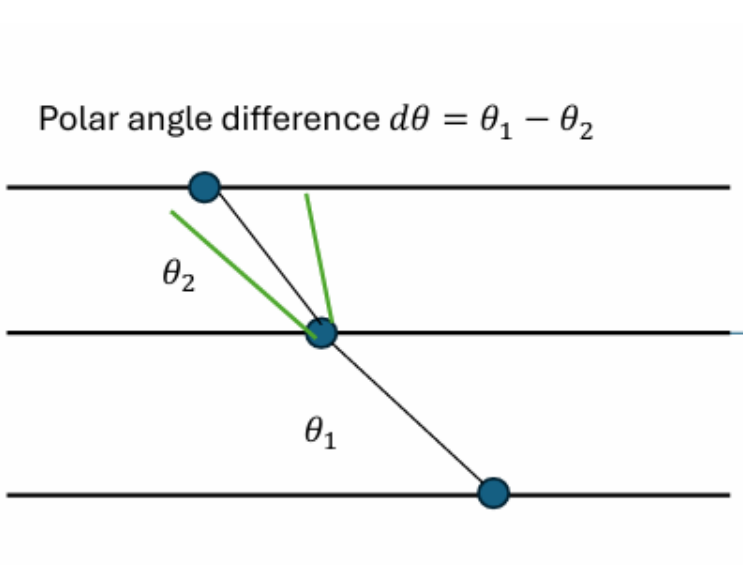
- Based on experimental informations :
 - Particle's originate along the beam axis.
 - Maximum bending due to the magnetic field.



Doublet cut : Particle's origin



Doublet cut : p_T cut



θ cut

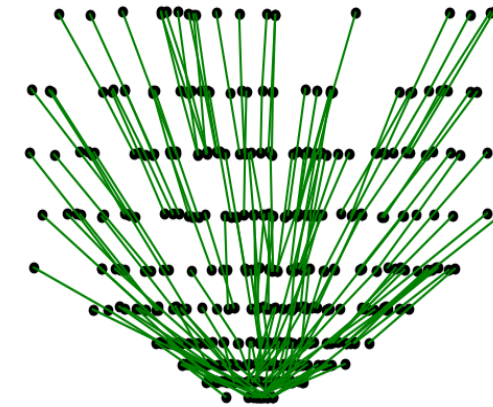
Performance metric

- Efficiency = $\frac{\# \text{reconstructed signal segments}}{\# \text{total signal segments}}$

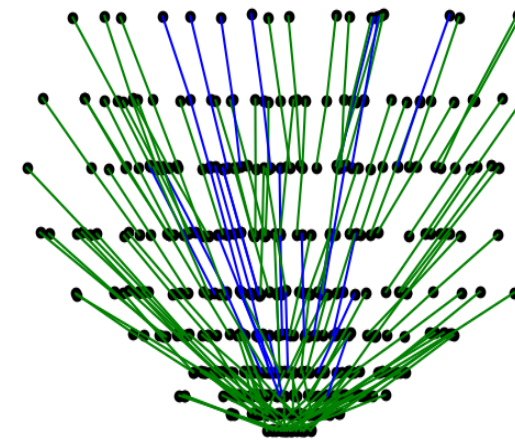
- $\frac{\# \text{Green (RS)}}{\# \text{Green (TS)}}$

- Purity = $\frac{\# \text{reconstructed signal segments}}{\# \text{total reconstructed segments}}$

- $\frac{\# \text{Green (RS)}}{\# \text{Green (RS)} + \# \text{Blue (RS)}}$



Truth Space



Reconstructed Space

Segments Building Results

- The cut values are selected based such that we get 100% reconstruction efficiency for muons at both the doublet and triplet levels.
- Triplet fit is implemented to the triplets obtained after preselection cuts and χ^2 cut is applied on triplets.

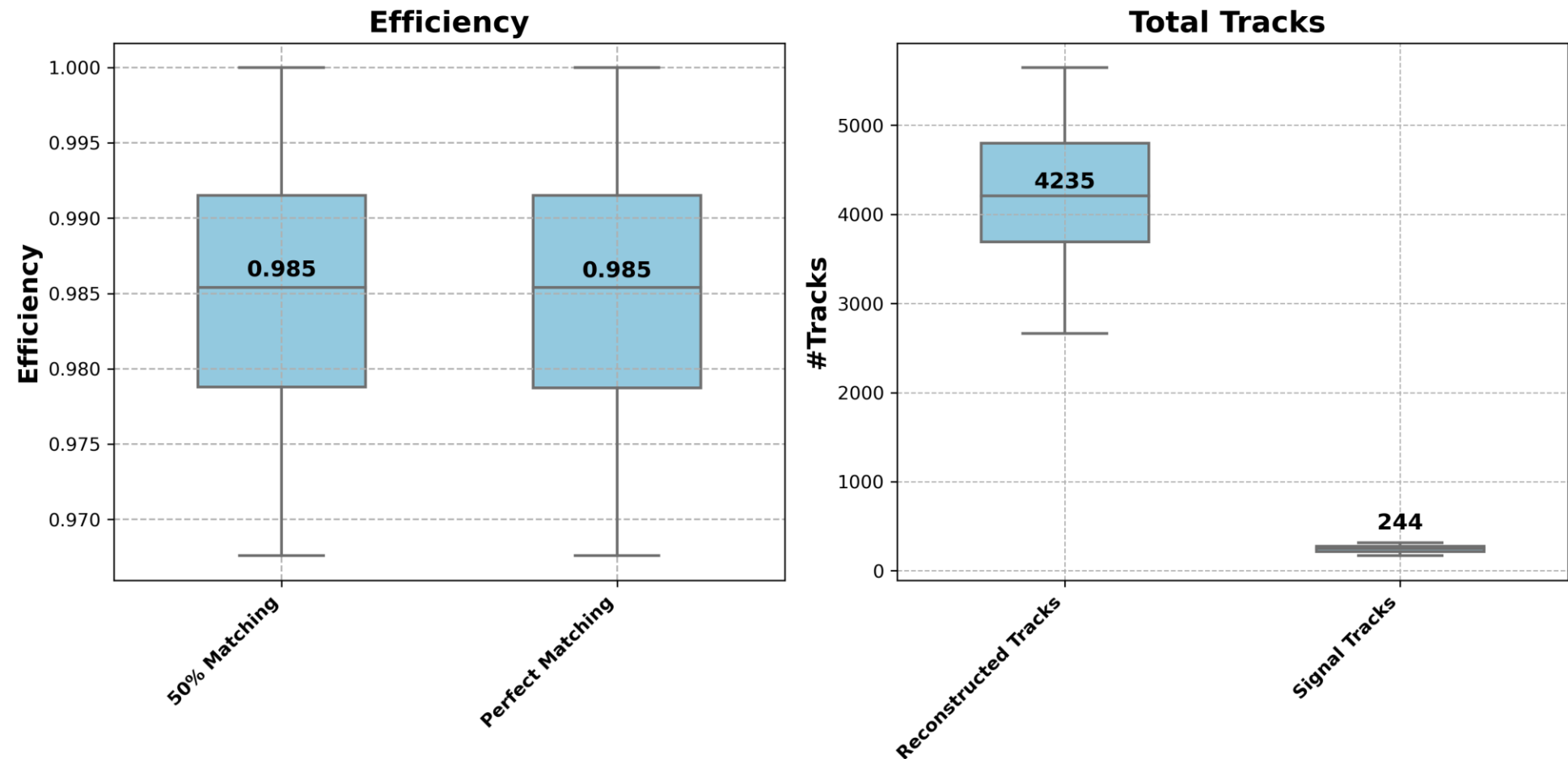
Segment	# Signal Segments	# Reconstructed Segments	Average Efficiency	Average Purity
Doublet	3,255	614,145	99.9%	0.5%
Triplet (before χ^2 cut)	3,526	153,953	99.8%	2.4%
Triplet (after χ^2 cut)	3,526	34,036	99.6%	10.5%

CA Performance Metric

- Triplets obtained after χ^2 cut is used to form track candidates with CA.
- The performance of CA is evaluated by **signal efficiency** and **signal purity**.
- Two matching schemes are used for evaluating performance, where a track candidate is only called successfully reconstructed signal if :
 - **50 % matching** : If within a track at least **50 % of hits** belong to signal track
 - **Perfect matching** : If within a track candidate **all hits** belong to a signal track

CA Performance

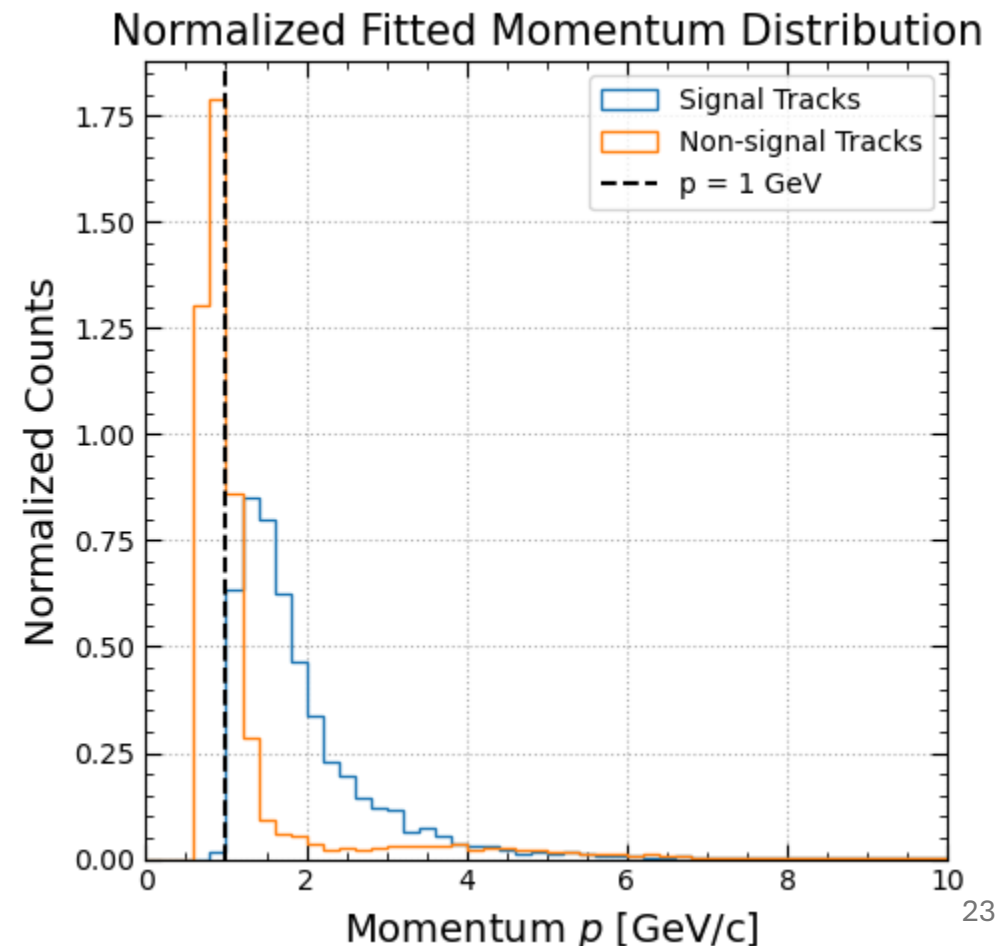
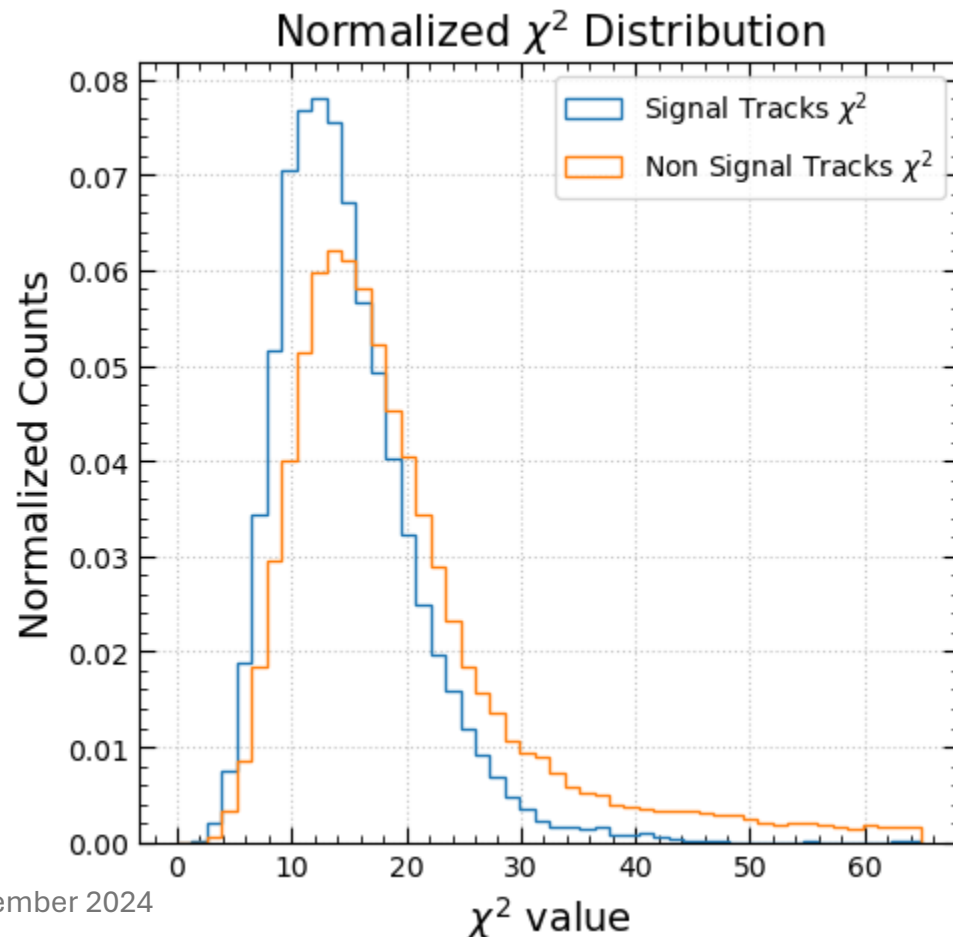
Performance are shown as a box plot for 90 events



Reducing Combinatorics
Without affecting efficiency
significantly

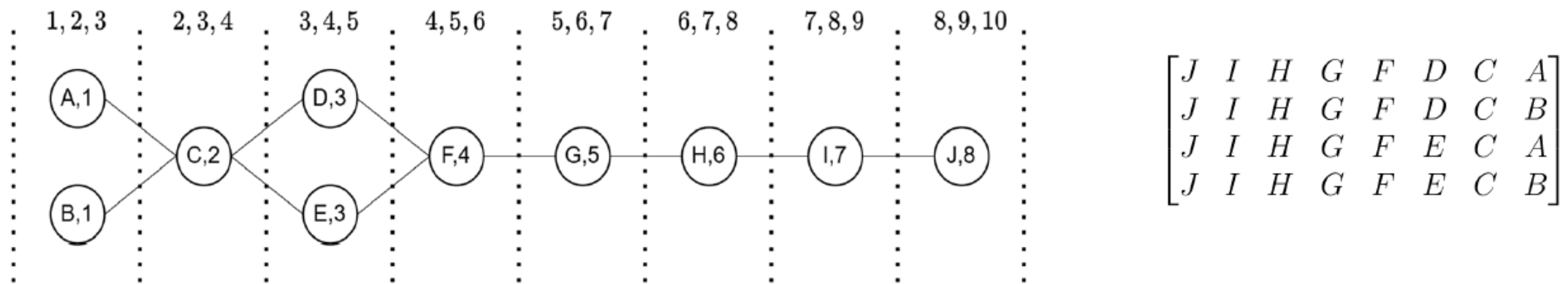
χ^2 and Momentum cuts

- Track fitting algorithm is applied to all track candidates generated by CA and χ^2 and momentum p are obtained.



Track Trimming

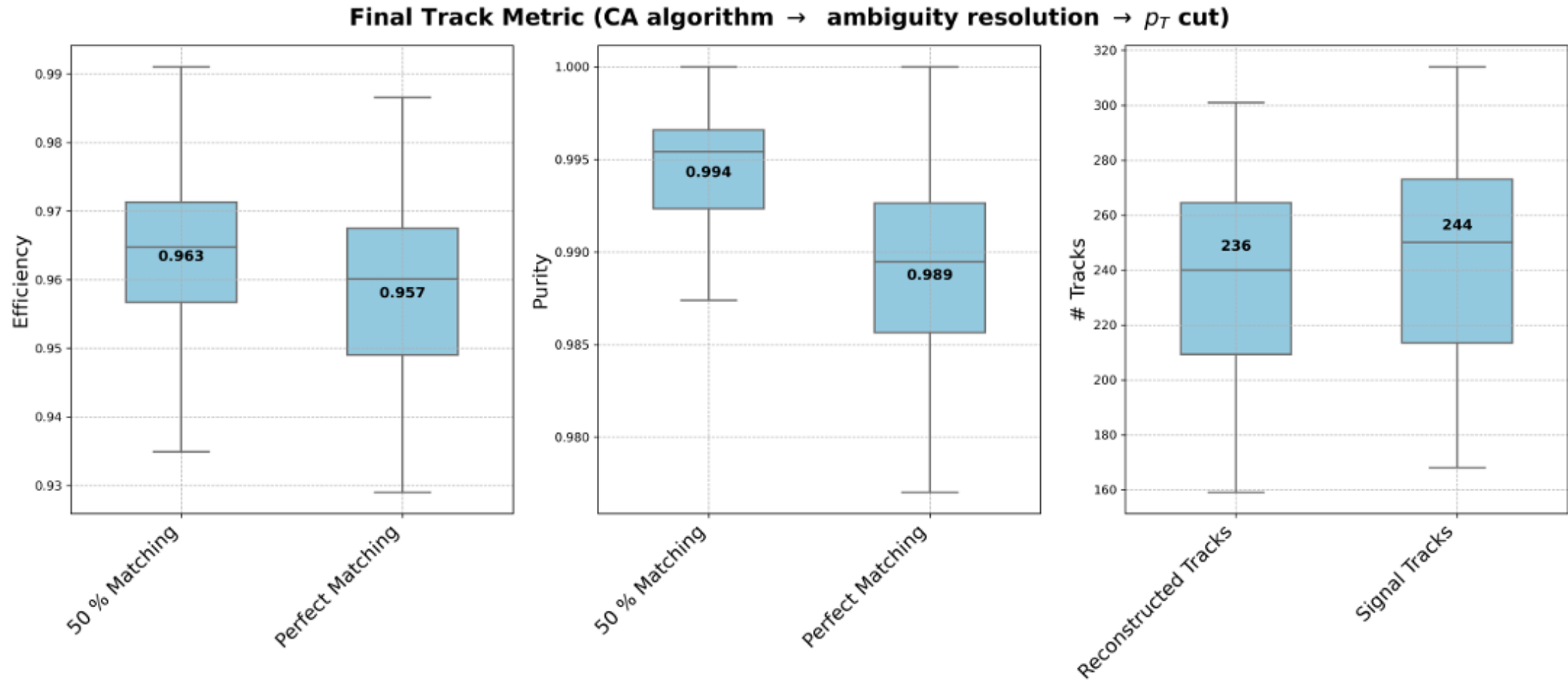
- Forming tracks with CA is an optimization problem that has no free ends that give rise to tree-like structure from both end points.



- Trimming**: selecting **one track** from a tree-like structure that has a minimum χ^2 .

Final Results

- Dealing with “track sharing hits” and applying p_T cut on the tracks.



Summary and Outlook

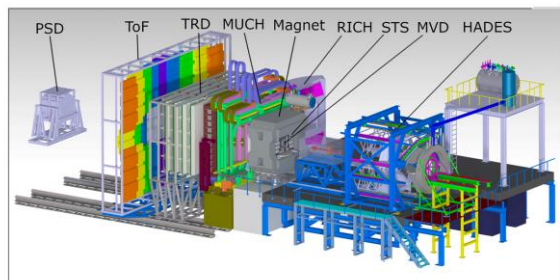
- CA was applied in the barrel region of the TrackML dataset to specifically reconstruct a longest track chain in the barrel region.
- The CA algorithm itself gives an impressive 98.5% perfect matching eff.
- The Ambiguity resolution reduces the number of track candidates significantly and improves the avg. perfect purity to 99 % by reducing eff. about ~ 2.5 %.
- The CA can also be studied for entire tracking detectors by forming track candidates with varying number of hits.
- Triplet formation, triplet fitting , CA and track fitting algorithm can be accelerated with GPUs.

PhD Project

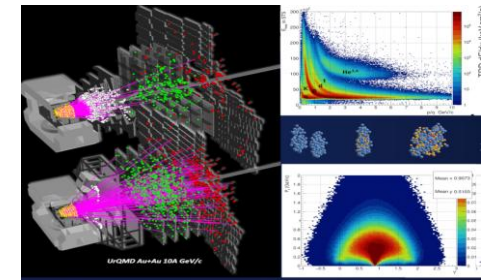
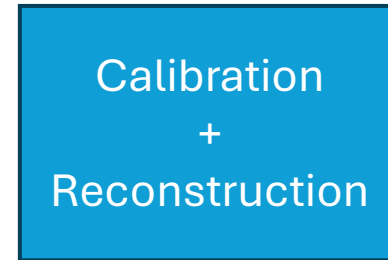
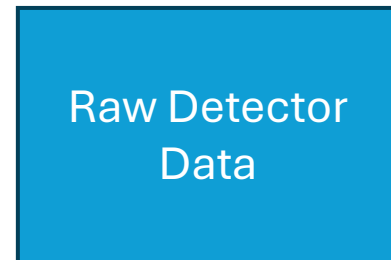
AI-Based QA Monitoring for the CBM experiment

Data Quality Monitoring (DQM)

- Experimental nuclear and particle physics data are costly to obtain.



Experiment

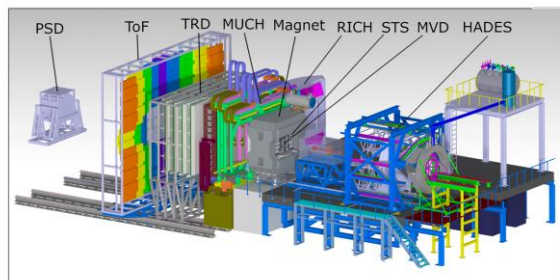


Physics Analysis

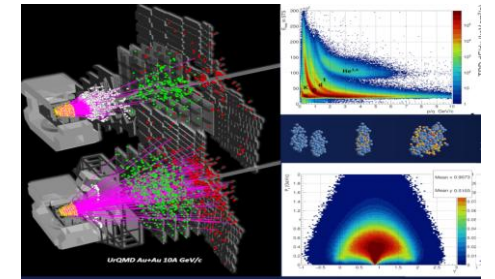
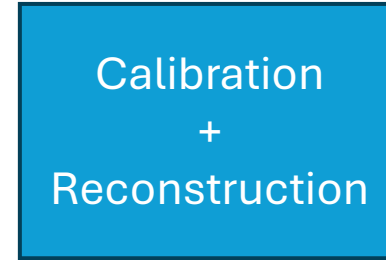
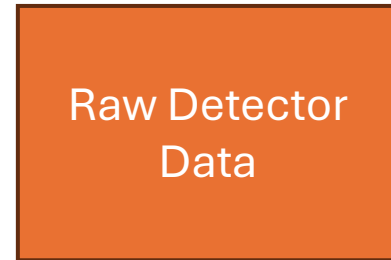
- DQM ensures **safe operation** of sub detectors and **data integrity**.
- During the run, hundreds of **QA plots (histograms)** are monitored by the shift crew.
- **Inconsistent task** due to human reaction time, different level of expertise during data taking.
- **Goal** : To automate QA plots monitoring for CBM with computer vision models.

Data Quality Monitoring (DQM)

- Experimental nuclear and particle physics data are costly to obtain.



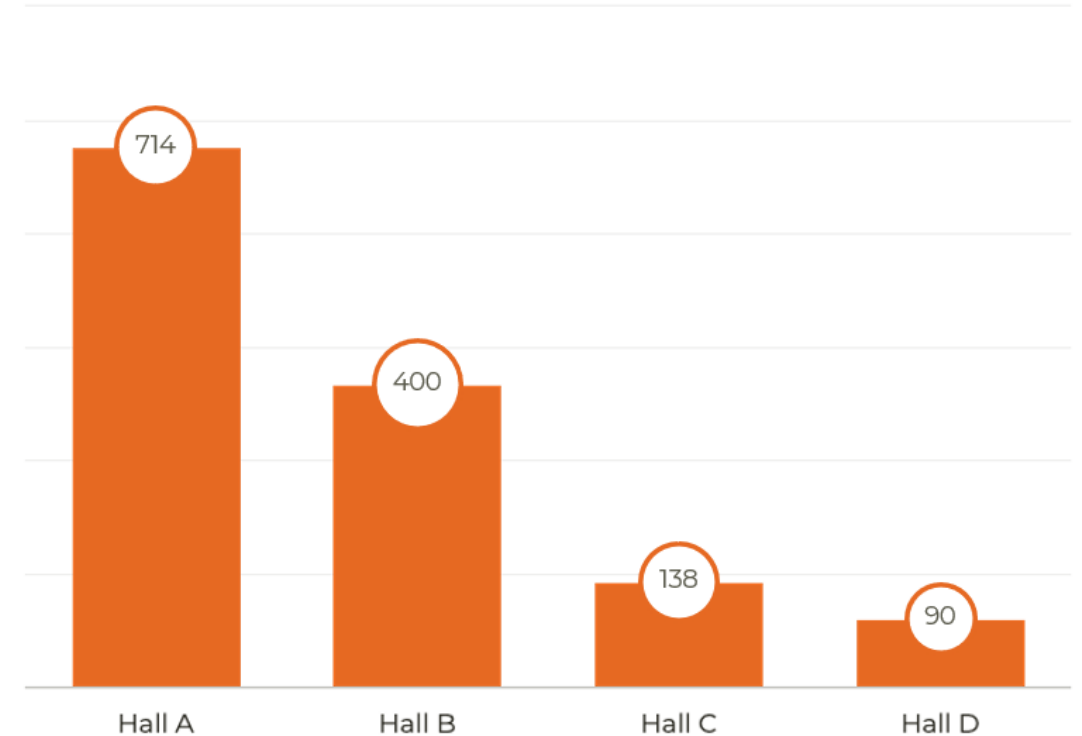
Experiment



Physics Analysis

- DQM ensures **safe operation** of sub detectors and **data integrity**.
- During the run, hundreds of **QA plots (histograms)** are monitored by the shift crew.
- **Inconsistent task** due to human reaction time, different level of expertise during data taking.
- **Goal** : To automate QA plots monitoring for CBM with computer vision models.

- Hydra was first developed at Jefferson Lab in 2019 and is now operational across all experimental halls.
- It is a computer vision based experimental agnostic deployable system (docker,MySQL) that performs near real time image classification.
- Main features :
 - Training and deploying of multiple computer vision models.
 - Labelling images via web browser.
 - Online monitoring and prediction.



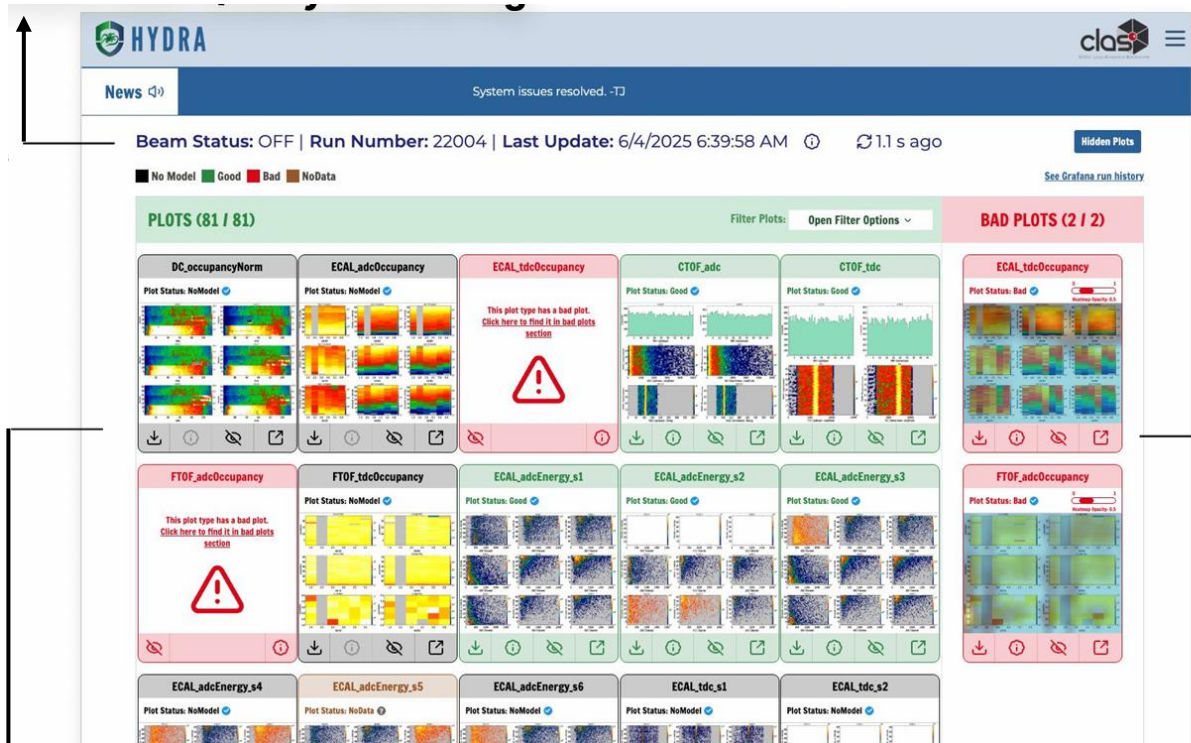
Approximate **number of individual histograms per experiment per run**, monitored by the shift crew for each experimental hall at JLab.

Hydra in Operation

- Different models are trained for each subdetectors and their corresponding plots.
- The alarm is rung as soon as any bad image encounters during the run.

Status

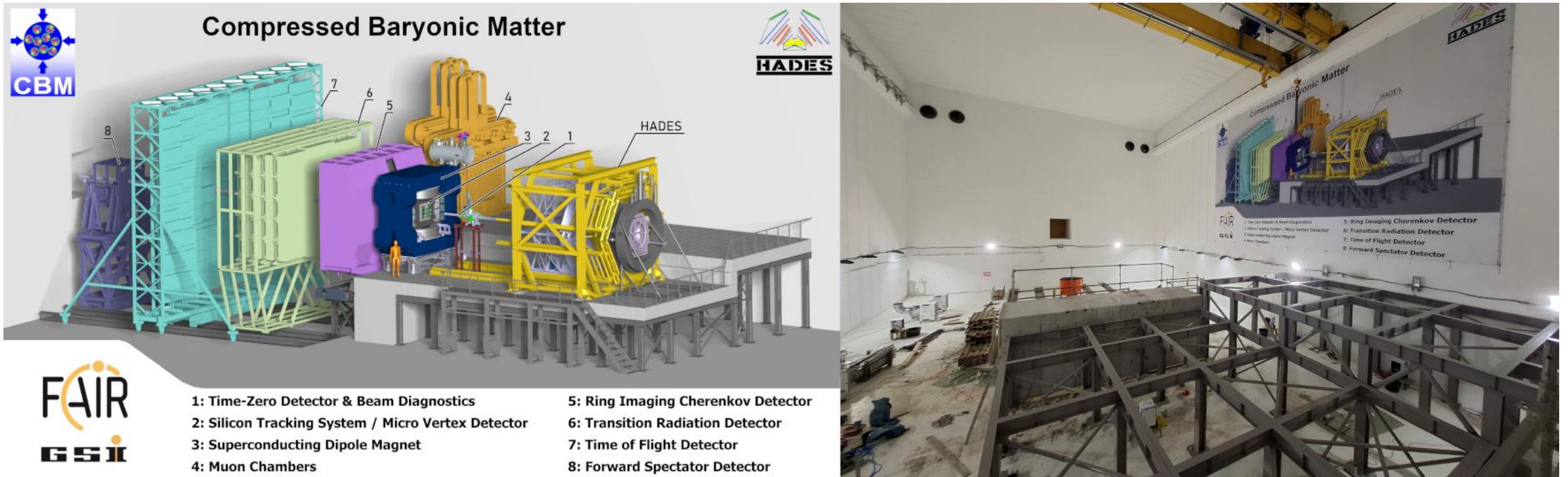
Image Gallery



Reserved for bad plots

<https://indico.jlab.org/event/459/contributions/11393/>

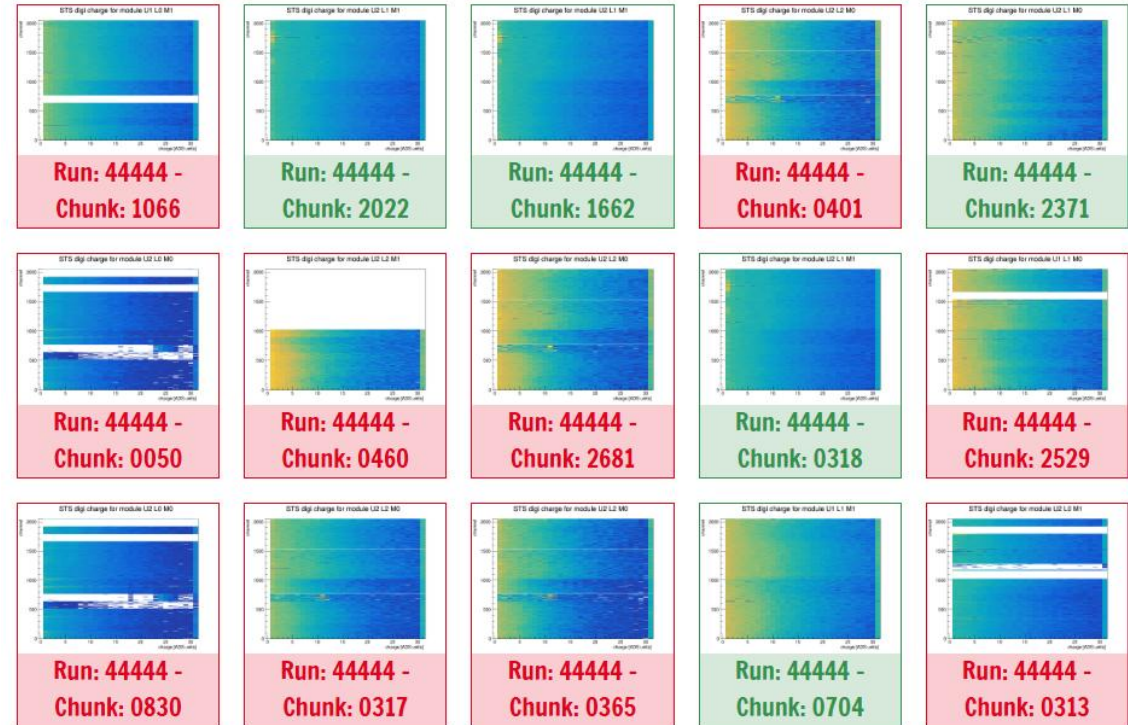
HYDRA for CBM



- Fixed-target heavy-ion experiment at FAIR (GSI Darmstadt) to explore QCD phase diagram in the region of high net-baryon density.
- Aim of my project : Implementing HYDRA for CBM and expanding its capabilities with unsupervised learning.

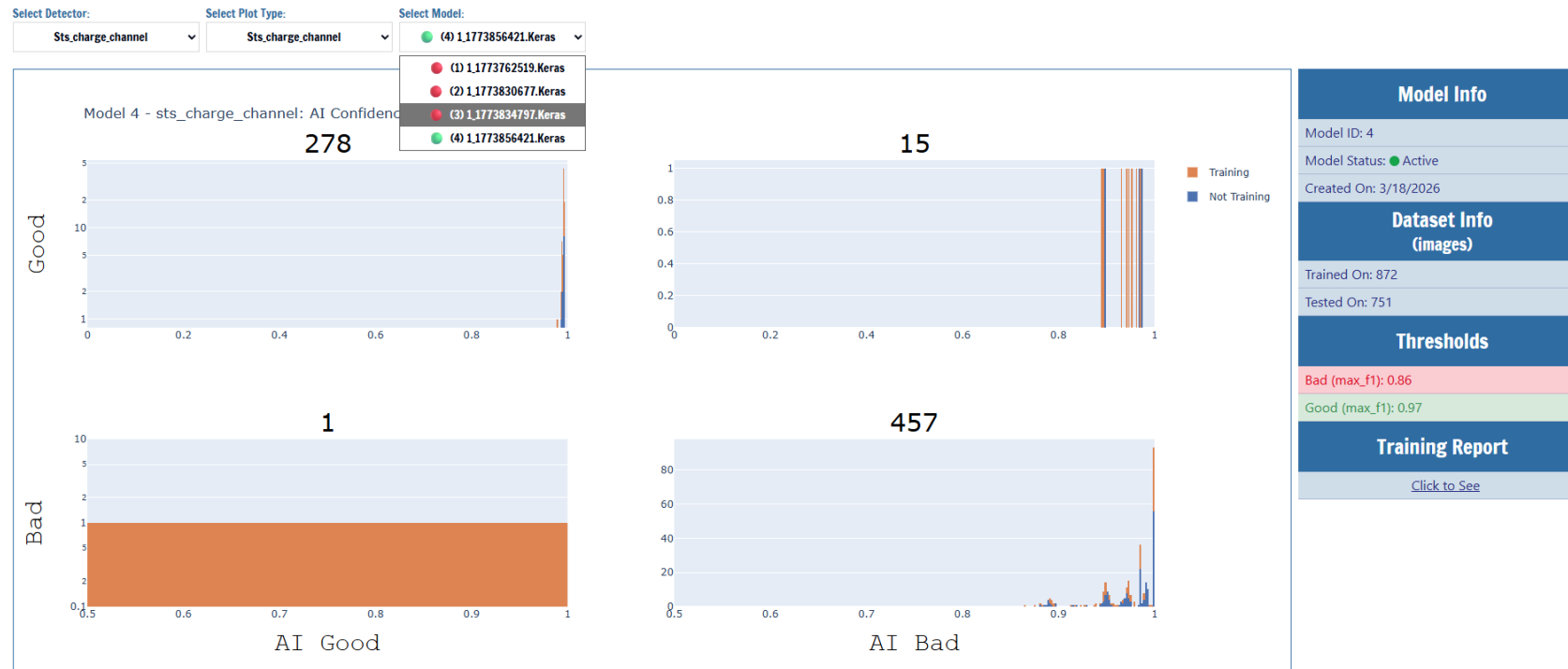
First Stage : Labelling plots

- Analysing QA plots for the STS detector (mCBM run 2024) with HYDRA.
- Images are labeled on the web page and used as a training dataset.
- Labelling task will be distributed among the sub detector experts.



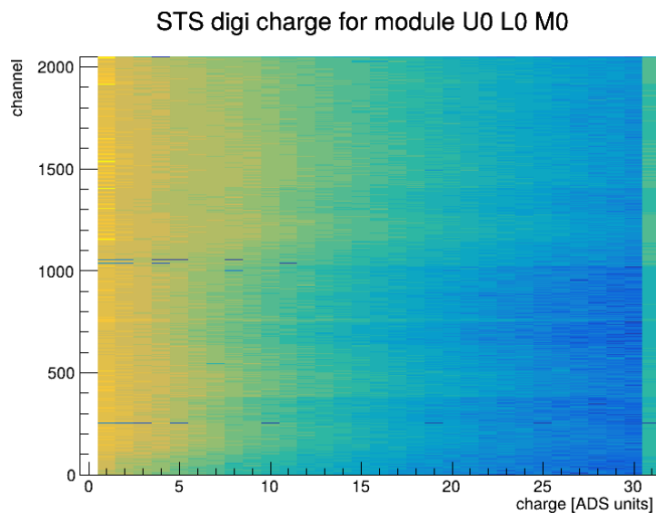
Second Stage : Training and Deployment

- Once training dataset is prepared we can train computer vision model.
- Multiple models can also be trained with different configuration and best model for real-time deployment can be selected.

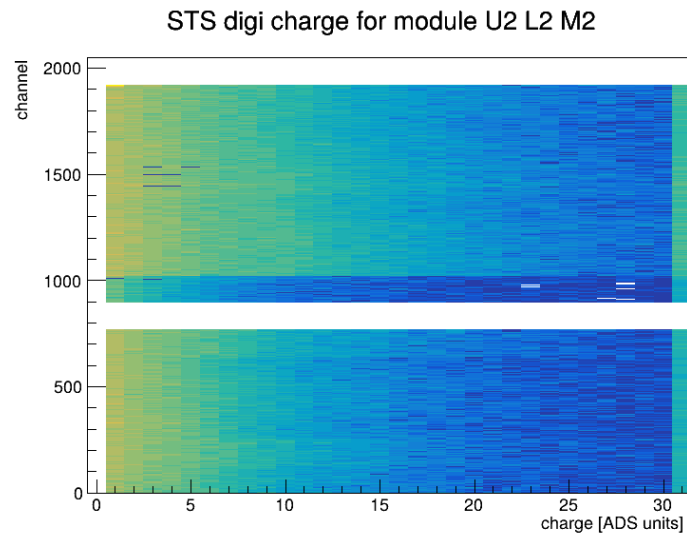


Interpretability with GradCAM

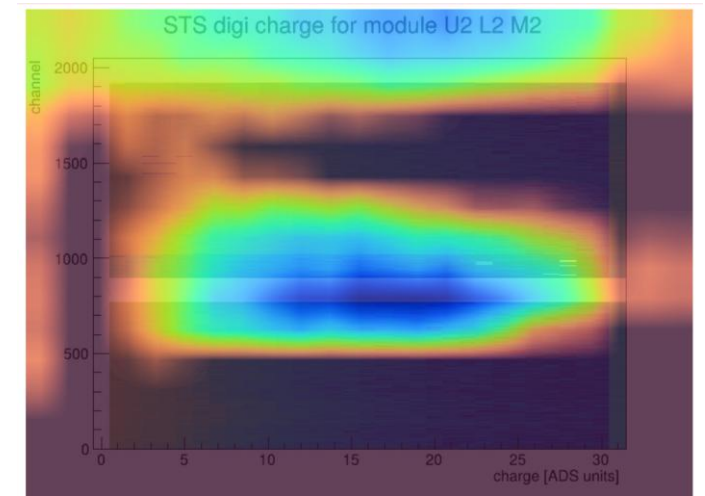
- HYDRA also provide visual explanations with Gradient-weighted Class Activation Maps (GradCAM)
- Understanding the reason behind the prediction of a model.



Recognized Good Image



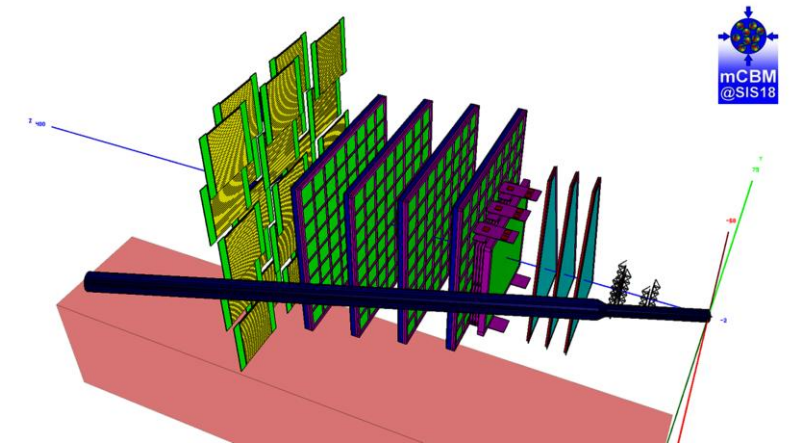
Recognized Bad Image



Bad Image with GradCAM

Progress so Far

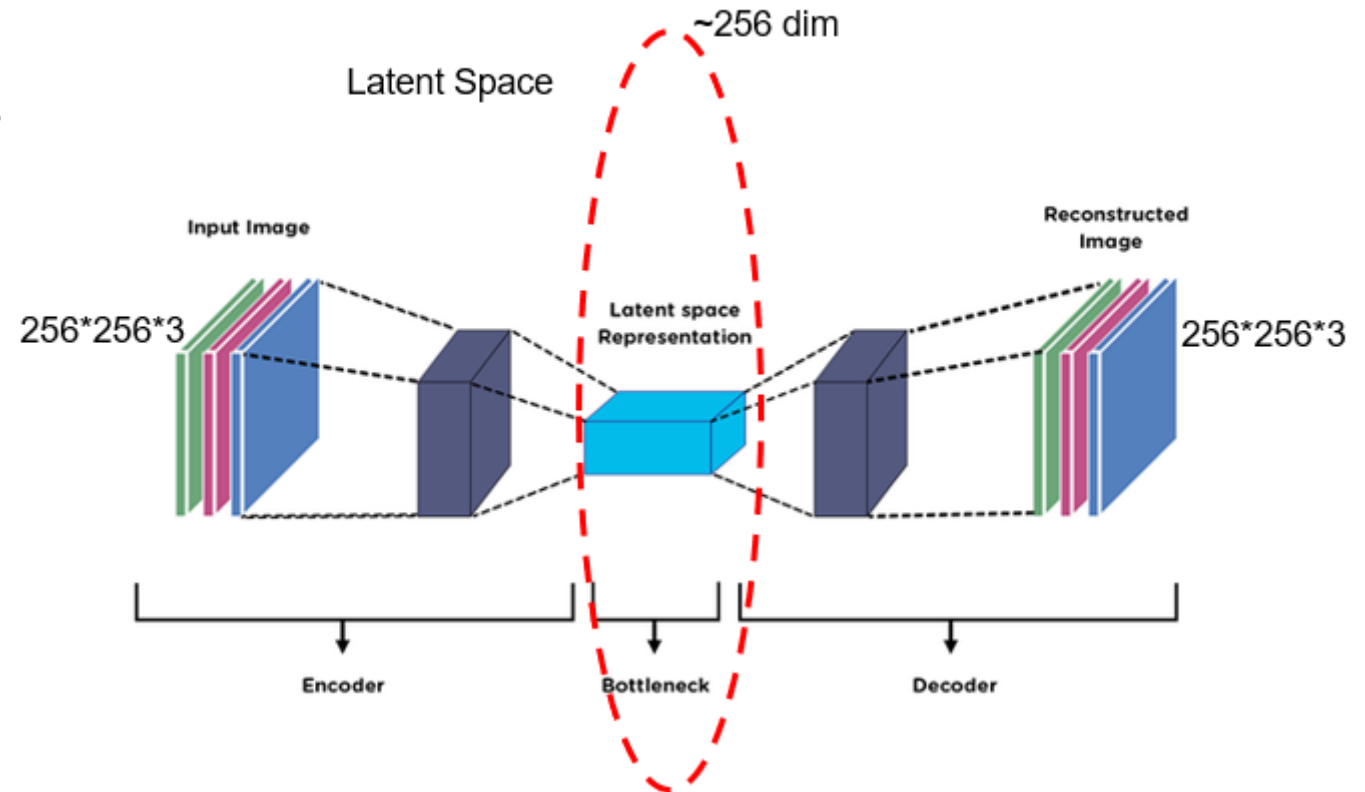
- Understanding the HYDRA system and inherent database structure (Podman, MySQL, Tensorflow)
- Deployment at GSI computing node.
- For STS (2d charge distributions, mCBM Run 2024) :
 - Labelling plots via web browser.
 - Training computer vision model.
 - Deploying model and monitoring plots in real time.



mCBM experiment

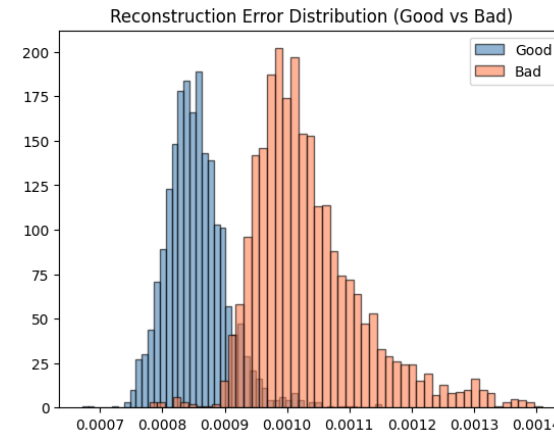
Autoencoders with Hydra

- Hydra currently supports the training and deployment of only Supervised ML models.
- Exploration of Autoencoders (Unsupervised Models) for two cases :
 - Anomaly detection.
 - Clustering images.

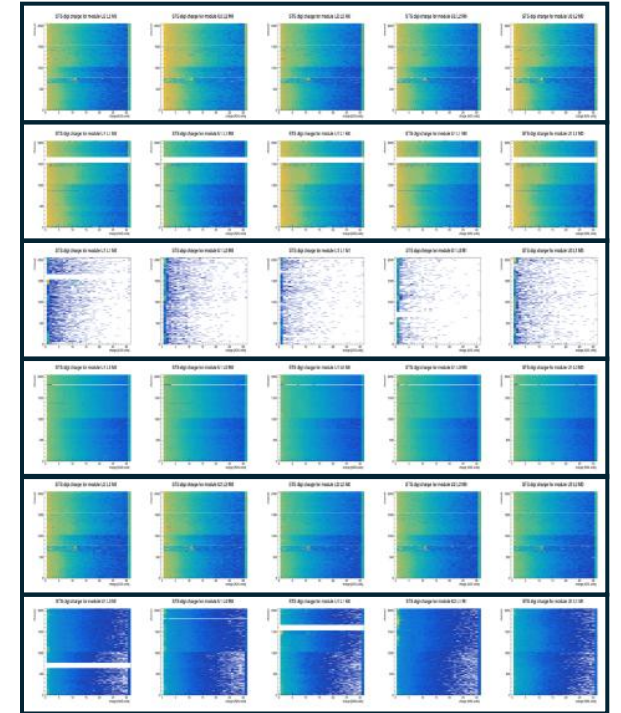


Current Status and Future Prospects

- Autoencoders performance for STS QA plots for clustering and anomaly detection : Looks promising !
- Exploring various architectures and attempting to add autoencoders functionality within HYDRA.
- Comparison between supervised and unsupervised ML models.
- Adding other subdetectors QA plots to hydra (mCBM run & simulations) and performance analysis.



Anomaly detection with Autoencoders



STS Images clusters by Autoencoders

The Science of Deduction



The ideal reasoner would, when he had once been shown a single fact in all its bearings, deduce from it not only all the chain of events which led up to it but also all the results which would follow from it.

The Five Orange Pips, The Adventure of Sherlock Holmes

Thank you

References

- I. Abt et al. “CATS: a cellular automaton for tracking in silicon for the HERA-B vertex detector”. In: Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 489.1-3 (Aug. 2002)
- Andre Schöning. A General Track Fit based on Triplets. 2024. arXiv: 2406.05240 [physics.ins-det]. url: <https://arxiv.org/abs/2406.05240>
- Valentina Akishina. “Four-dimensional event reconstruction in the CBM experiment”. PhD thesis. Frankfurt am Main, Germany: Goethe University Frankfurt, 2019.
- TrackML dataset : [The TrackML high-energy physics tracking challenge on Kaggle](#)

SLIDES from : CHEP , ABHI

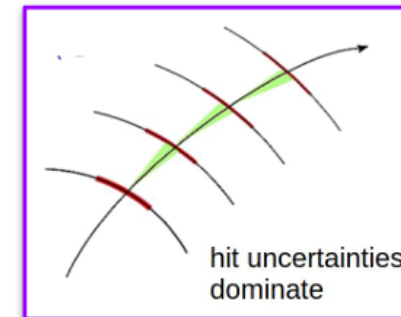
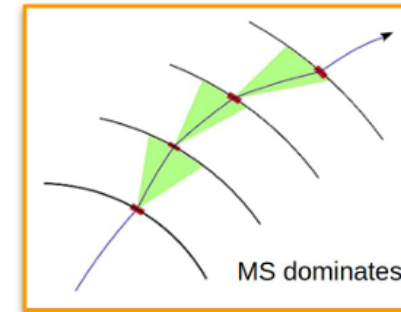
General Triplet Track Fit

- Generalized from multiple Coulomb scattering (MS)-only triplet fit [1] to include hit uncertainties [2].
- Fit quality for a track:

$$\chi^2(p, \vec{\delta}) = \sum_{\text{scatt. } k=0}^{n_{\text{scatt.}}-1} \left(\frac{\Delta\Phi_{\text{MS},k}^2}{\sigma_{\Phi_{\text{MS},k}}^2} + \frac{\Delta\Theta_{\text{MS},k}^2}{\sigma_{\Theta_{\text{MS},k}}^2} \right) + \sum_{\text{hit } j=0}^{n_{\text{hit}}-1} \vec{\delta}_j^\top V_j^{-1} \vec{\delta}_j$$

$$\vec{\delta}_j = \vec{r}_{\text{fit},j} - \vec{r}_{\text{meas},j}$$

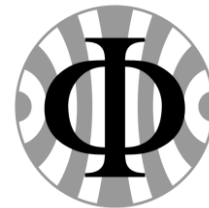
- The fit of the total momentum p & hit position shifts $\vec{\delta}_j$ is done in two steps:
local processing of hit triplets & a global track fit.



Track Reconstruction Using Cellular Automata for the High Luminosity LHC

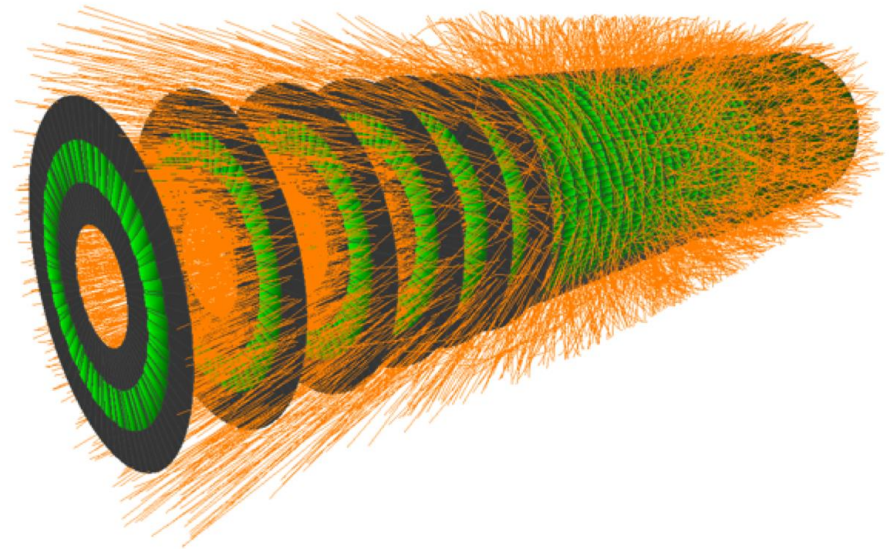


UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386



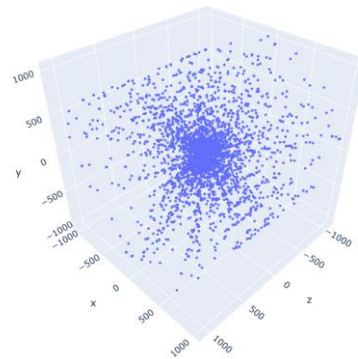
Motivation

- CERN is currently preparing for HL-LHC phase that aims to achieve unprecedented luminosity (5 to 7.5 times the current one).
- The average number of proton-proton collision per bunch crossing will be $\langle\mu\rangle = 200$.
- More collisions – more data recorded
- Track reconstruction for the central silicon-based tracker (ATLAS, CMS) becomes challenging in the high pile up environment.
- This thesis project explored the offline performance of parallelizable algorithm for track reconstruction.



Track Reconstruction

- Getting the properties of charged particles (momentum, charge, vertex etc.) from the recorded detector measurements i.e. 3d space points or hits.



- Performed in two stages :
 - **Track finding** : From an event, grouping hits into subsets (track candidates) which potentially originate from a particle.
 - **Track fitting** : Process of estimating a particle's parameters, covariance matrix, and track quality (χ^2) by taking theoretical and measurement uncertainty into account.
 - Fitting algorithm will be used at two stages : triplet level and track level.

What is Track Finding ?

September 2023



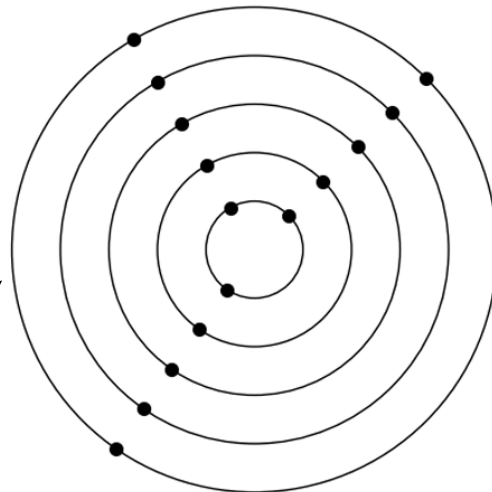
January 2024



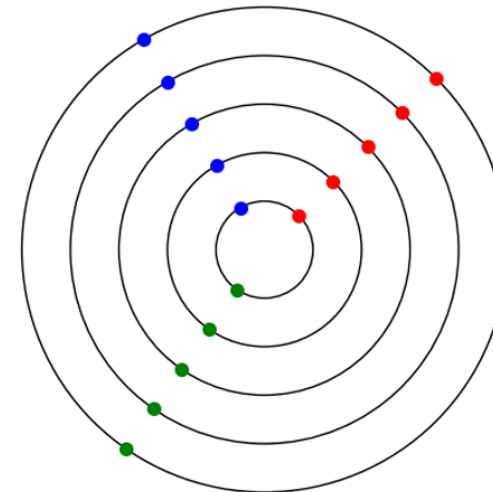
NEUENHEIMER FELD 227

Track finding : Grouping leaves/hits

- From the pile of leaves, form sets such that each set is associated with a branch symbolizing particle.
- Track finding employs pattern recognition algorithm in order to group hits into track candidate.



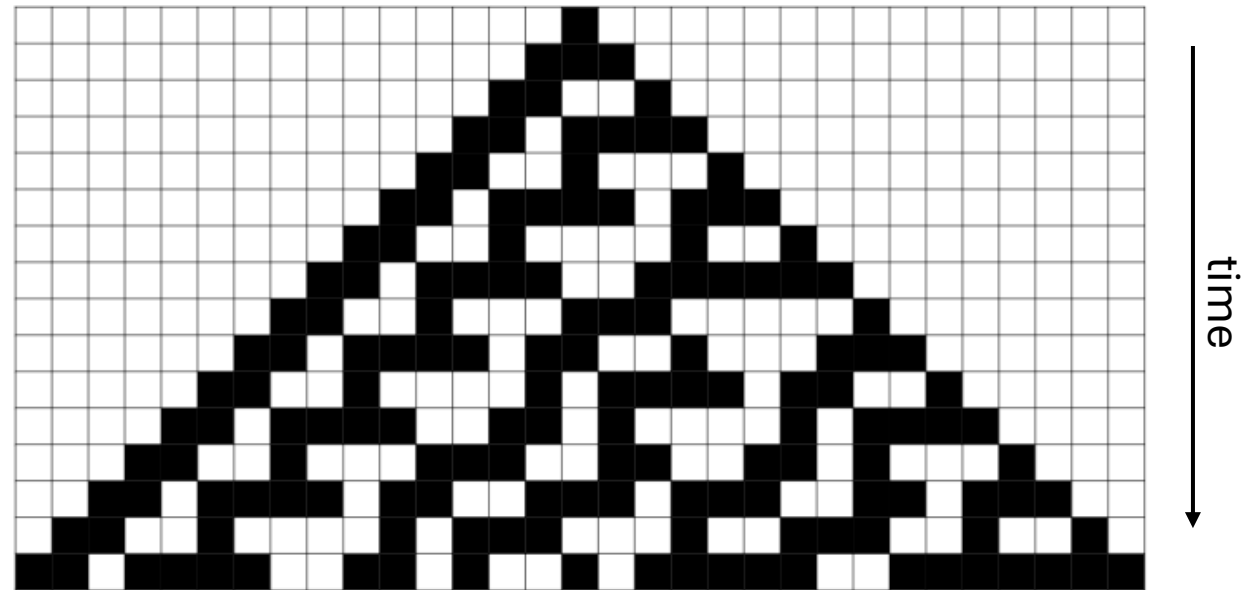
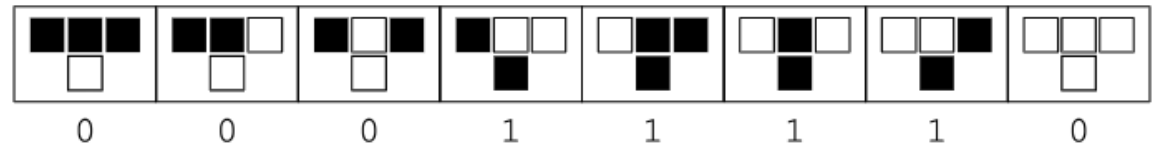
Hits before the track finding step.



Hits after the track finding step.

CA: Overview

- Dynamical system where space and time are both discretized.
- Each spatial site is called a “Cell” that can take integer or binary values.
- The evolution of each cell depends on the values of its neighboring cells and predefined rule.
- In one time step, all cells are updated simultaneously.



Example : 1D system (Rule 30)

Takeaway

- CA are local systems and gives full freedom in deciding :
 - Information to be embedded on a cell (universal).
 - Neighbour formation.
 - Local law for cell evolution.
- Since all cells in a grid update their values simultaneously, they are well suited for GPUs implementation.

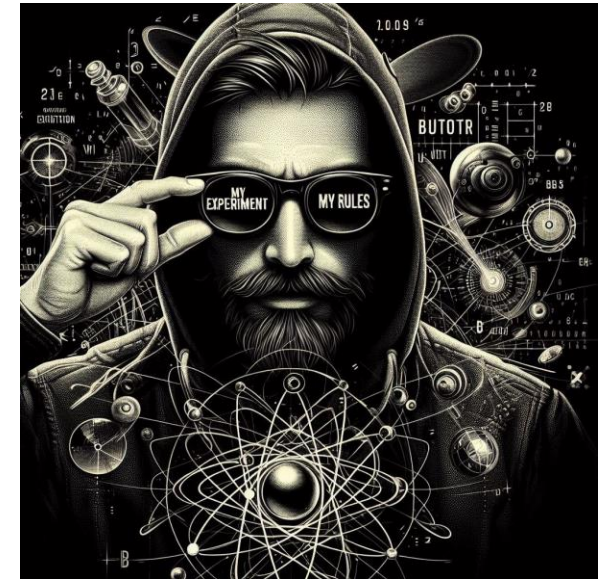
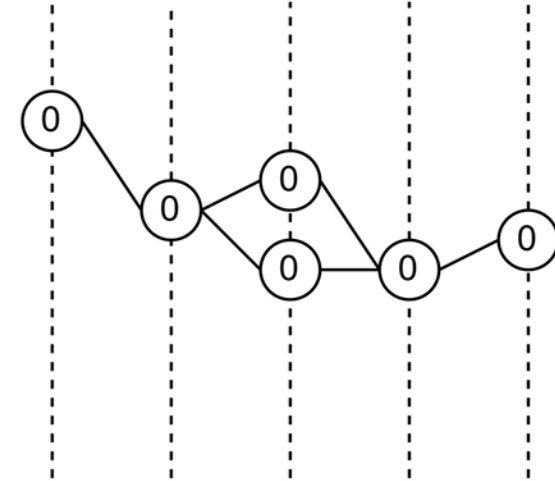


Image generated by MS copilot

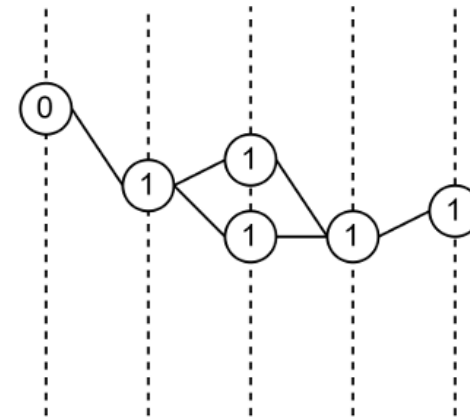
Track Finding with CA (CATS)

- **CATS** : Cellular automata for tracking in silicon for the HERA-B vertex detector (originally uses doublet as a cell).
- **Example** – 5 detector layers with some hit configuration.
- **Cells** – Each “hit” is described as a cell and initialized with a value of zero.
- **Neighbour** – “All hits in the previous layer” – (fully connected graph) with no neighbors within the layer

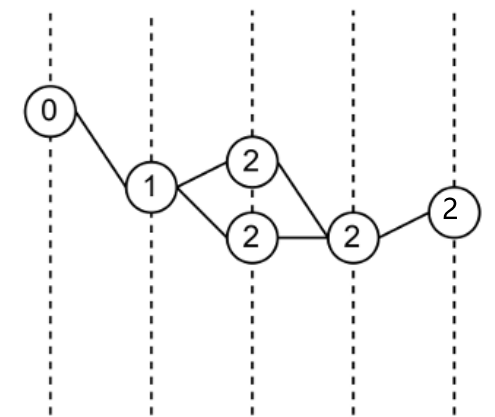


Rule and Evolution

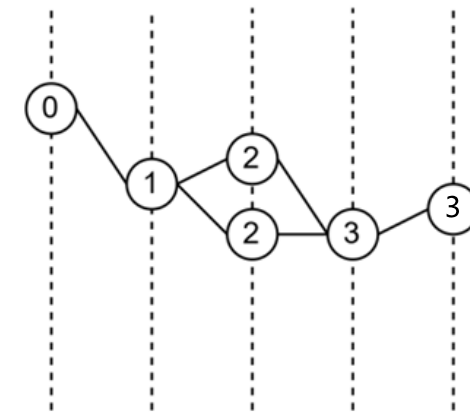
- Rule - At each iterative step :
 - Each cell looks at its left neighbour (if any).
 - Find the neighbour with maximum value.
 - New value = $\max(\text{neighbour values}) + 1$
 - The evolution is stopped when cells value do not change.



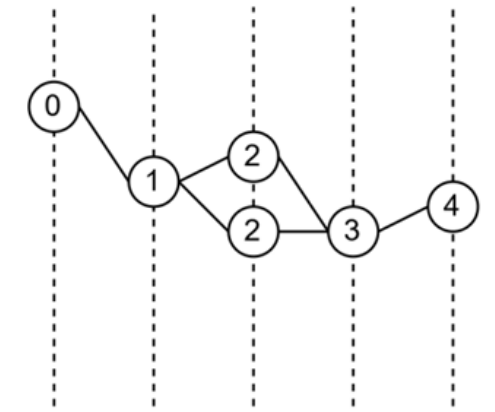
(a) Iteration = 1



(b) Iteration = 2



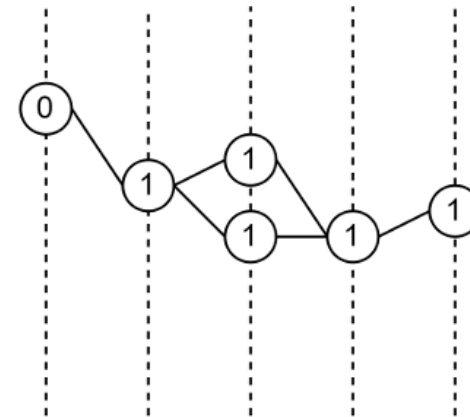
(c) Iteration = 3



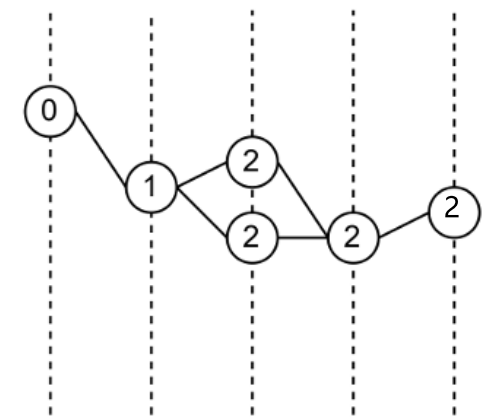
(d) Iteration = 4

Rule and Evolution

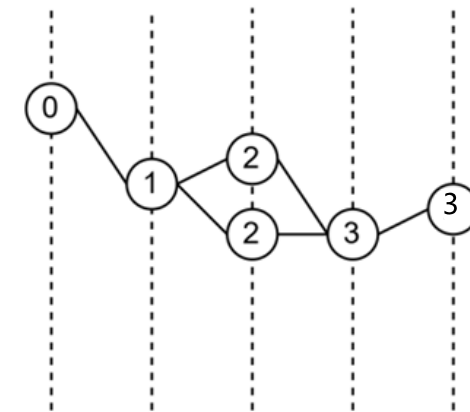
- Rule - At each iterative step :
 - Each cell looks at its left neighbour (if any).
 - Find the neighbour with maximum value.
 - New value = $\max(\text{neighbour values}) + 1$
 - The evolution is stopped when cells value do not change.
- Track candidates are formed from the **final configuration** (iteration = 4).
- From the cell with highest value, tracks are collected in order $4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0$.
- This example gives 2 possible tracks.
- Track candidates are further analysed by χ^2 value.



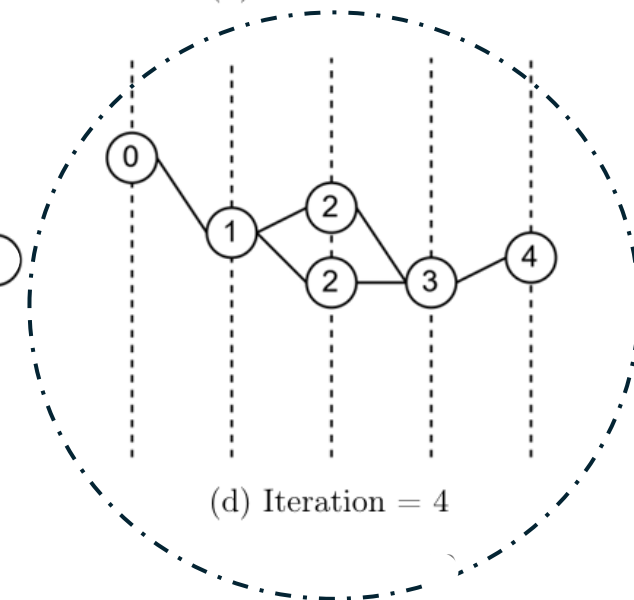
(a) Iteration = 1



(b) Iteration = 2



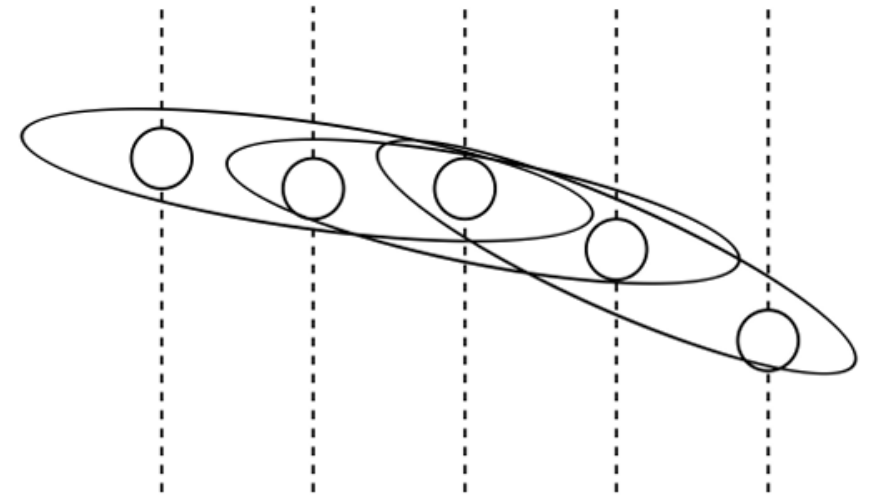
(c) Iteration = 3



(d) Iteration = 4

Triplet as a Cell

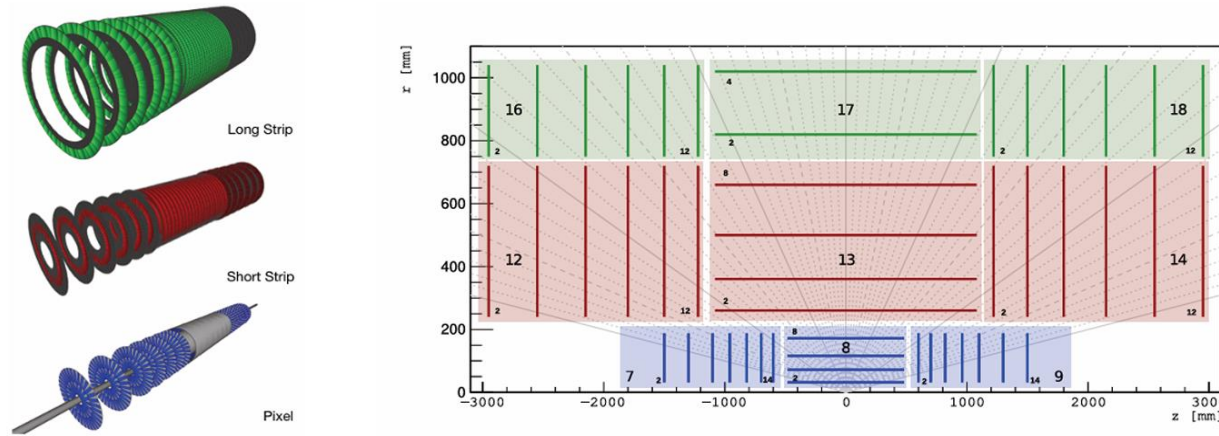
- For pile up conditions, the total number of hits per event is expected to be $\mathcal{O}(10^5)$
- Treating hit as a cell will be computationally challenging and memory expensive.
- Higher level information can be embedded in a cell by utilizing CA flexibility.
- Triplet of hits can be used a cell and track is formed by linking triplets.
- An early quality cut can also be applied by performing triplet fit and calculating χ^2 .



Track visualised as joining three triplets. Two consecutive triplets share two hits.

TrackML Dataset

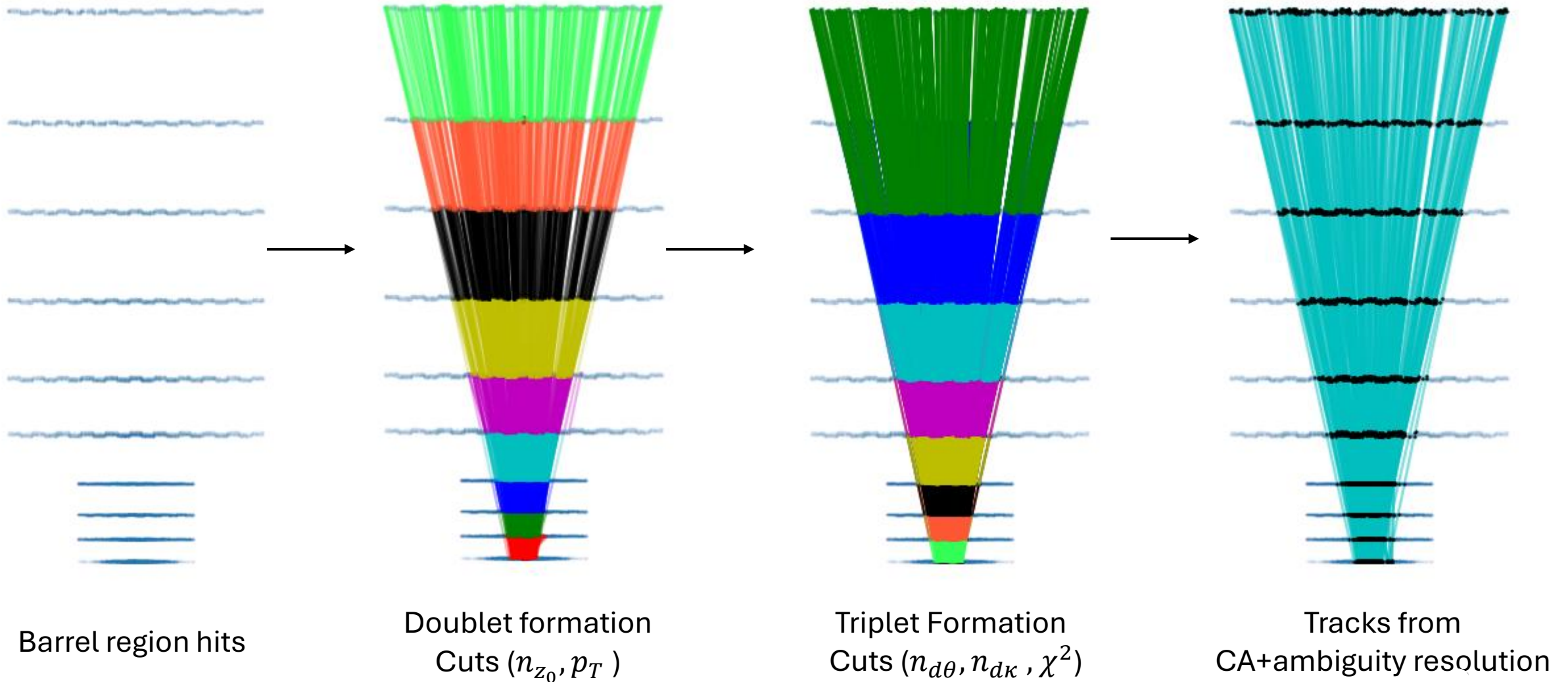
- TrackML dataset was launched a tracking machine learning challenge on the Kaggle platform. The dataset contains the collision data that mimics conditions for HL-LHC.



- CA is implemented in the **barrel region** of the TrackML detector.
- Target signal** : Particle's originating near the luminous region, $p_T \geq 1$ GeV and depositing hits in all barrel layers.

CA for the TrackML Dataset

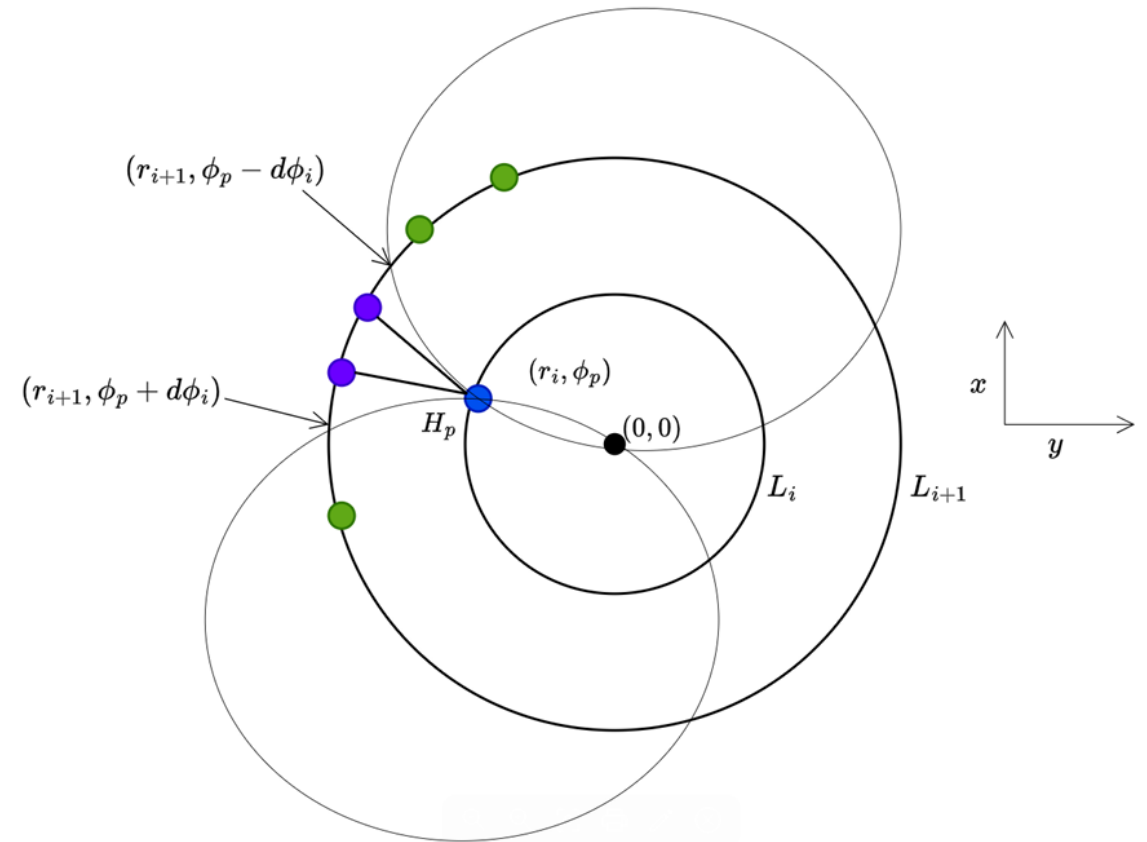
Longitudinal (r-z) plane view



Doublet cut : p_T

- Based on beam-line constraint. Particle's originate near origin in the transverse plane
- Gives the angular selection region in the transverse plane.
- The two circles are that passes through hit H_p and origin are obtained from :

$$p_{Tcut}[\text{GeV}] = 0.3B[\text{T}]r[\text{m}]$$

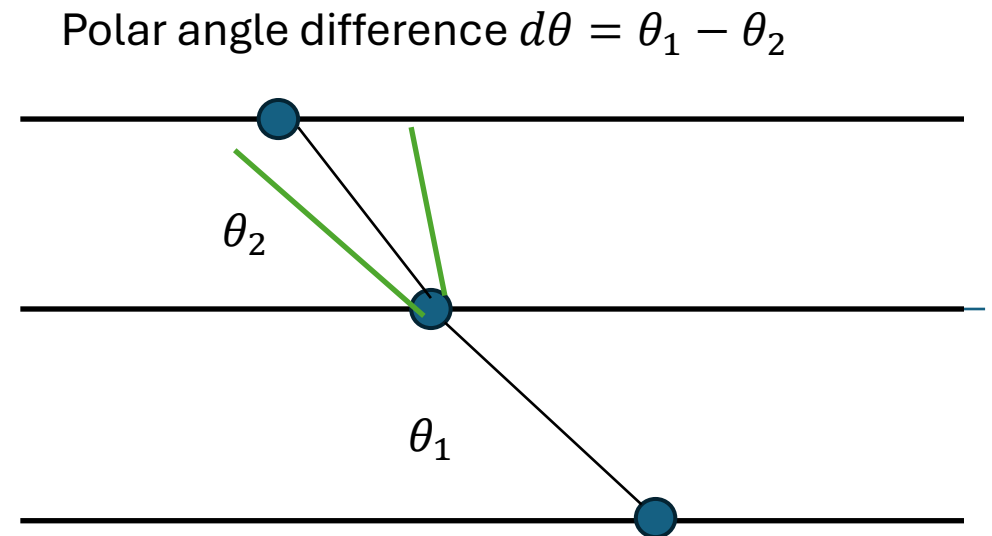


Triplet cut : $n_{d\theta}$

- Based on multiple scattering.
- The distribution for $d\theta$ is calculated from **signal triplets** and the layer specific selection windows are obtained.
- Two doublet must satisfy :

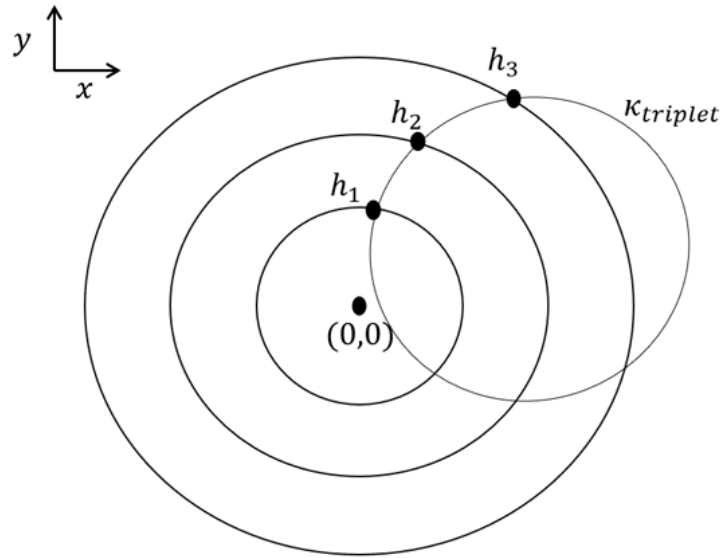
$$d\theta < [\mu - n_{d\theta}\sigma, \mu + n_{d\theta}\sigma]_{i+1}$$

- Pair of doublets that satisfy this cut are processed to next cut.

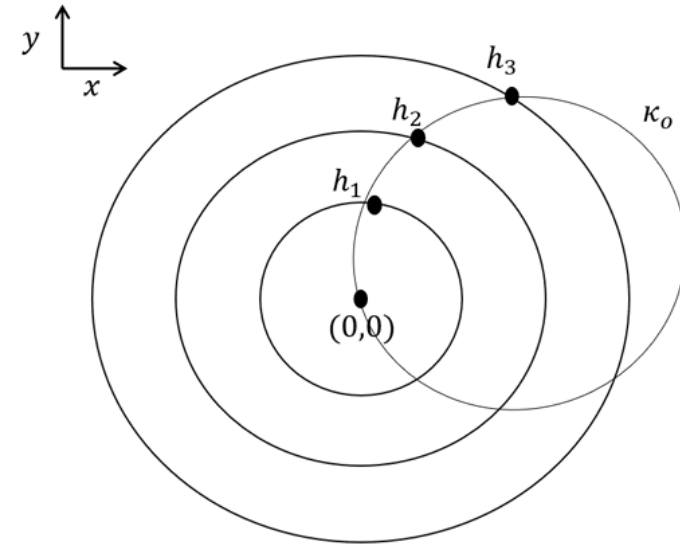


Triplet cut : $n_{d\kappa}$

- Based on beamline constraints and uses magnetic field information.



(a) Calculation of $\kappa_{triplet}$ from the three hits h_1, h_2, h_3



(b) Calculation of κ_O from the three hits h_1, h_2, h_3 and the origin

- The cut is based on curvature difference and applied similarly to the previous cuts.

$$d\kappa = \kappa_O - \kappa_{triplet} < [\mu - n_{d\kappa}\sigma, \mu - n_{d\kappa}\sigma]_{i+1}$$

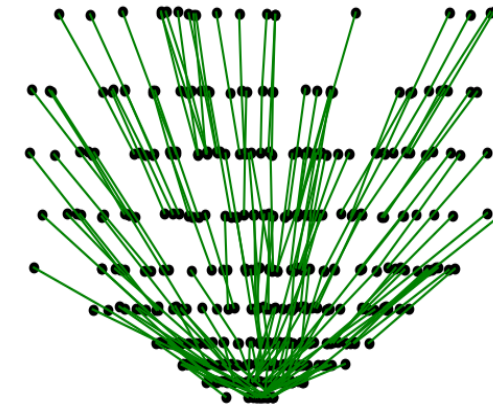
Performance metric

- Efficiency = $\frac{\# \text{reconstructed signal segments}}{\# \text{total signal segments}}$

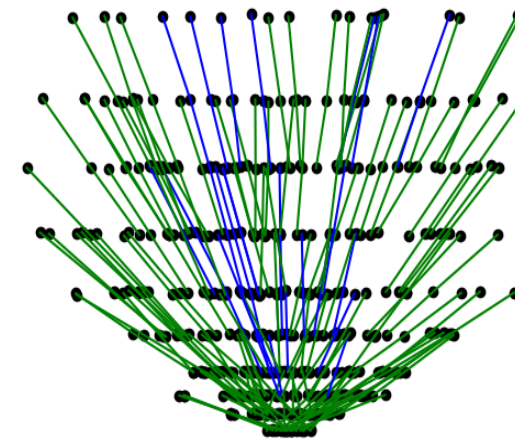
- $\frac{\# \text{Green (RS)}}{\# \text{Green (TS)}}$

- Purity = $\frac{\# \text{reconstructed signal segments}}{\# \text{total reconstructed segments}}$

- $\frac{\# \text{Green (RS)}}{\# \text{Green (RS)} + \# \text{Blue (RS)}}$



Truth Space



Reconstructed Space

Triplet Building Results

- The following cuts are selected based on their impact on **muon signal tracks segments**, achieving 100% reconstruction efficiency at both the doublet and triplet levels.
- The cuts are chosen with differential analysis for doublets and triplets separately.
- Triplet fit is implemented to triplets obtained after preselection cuts and χ^2 cut is applied.

Segment	Preselection Cut	Value
Doublet	n_{z0}	4
Doublet	p_{Tcut}	0.6 GeV
Triplet	$n_{d\theta}$	3
Triplet	$n_{d\kappa}$	6

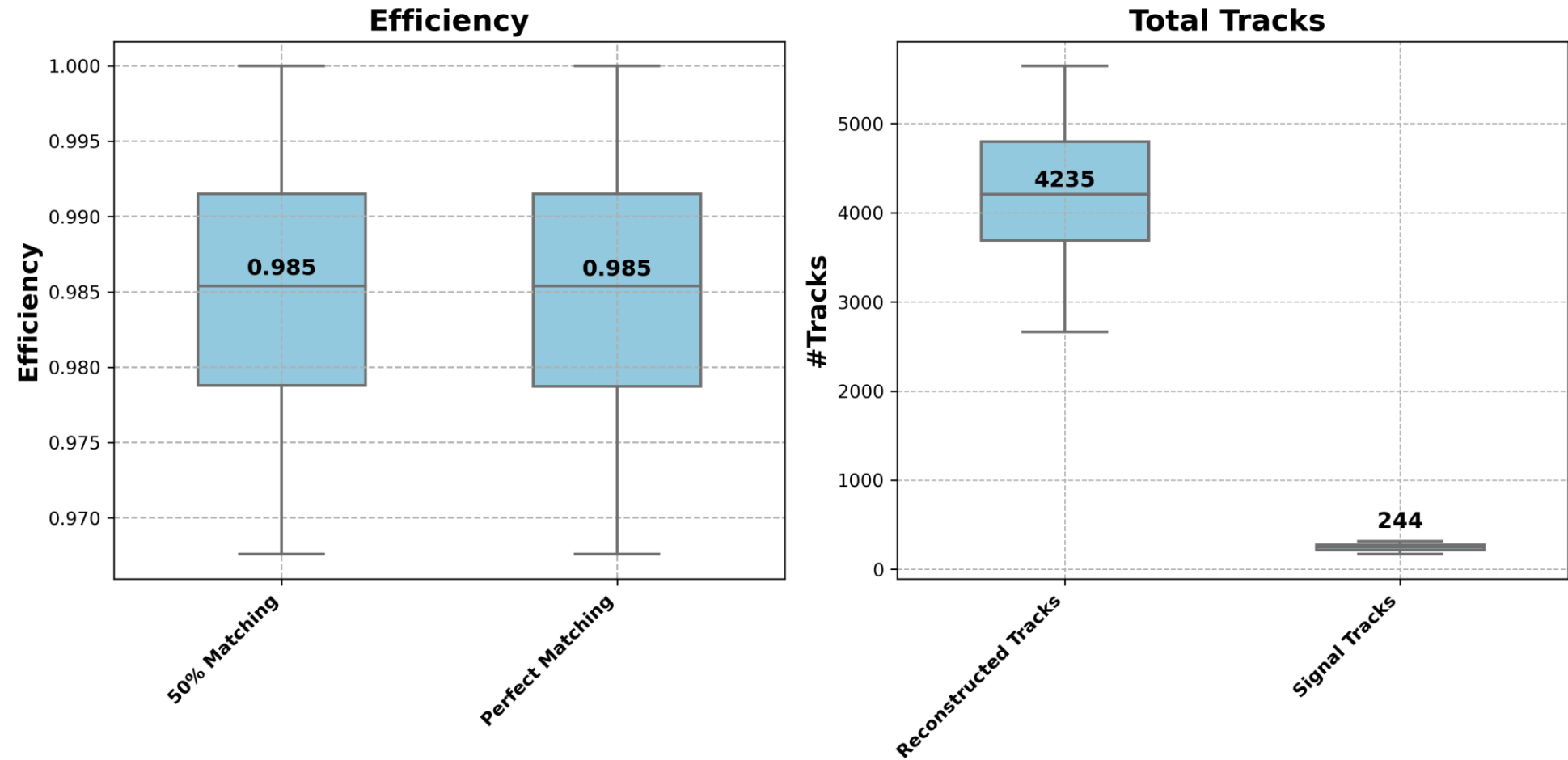
Segment	# Signal Segments	# Reconstructed Segments	Average Efficiency	Average Purity
Doublet	3,255	614,145	99.9%	0.5%
Triplet (before χ^2 cut)	3,526	153,953	99.8%	2.4%
Triplet (after χ^2 cut)	3,526	34,036	99.6%	10.5%

CA Metric

- Triplets obtained after χ^2 cut is used to form track candidates with CA.
- The performance of CA is also evaluated by **signal efficiency** and **signal purity**.
- The purity plots are only shown when track candidates do not share any hit among them.
- Two matching schemes are used for evaluating performance where a track is only called **successfully reconstructed signal** if :
 - **50 % matching** : If within a track at least 50 % of hits belong to signal track
 - **Perfect matching** : If within a track candidate all hits belong to a signal track.

CA Performance

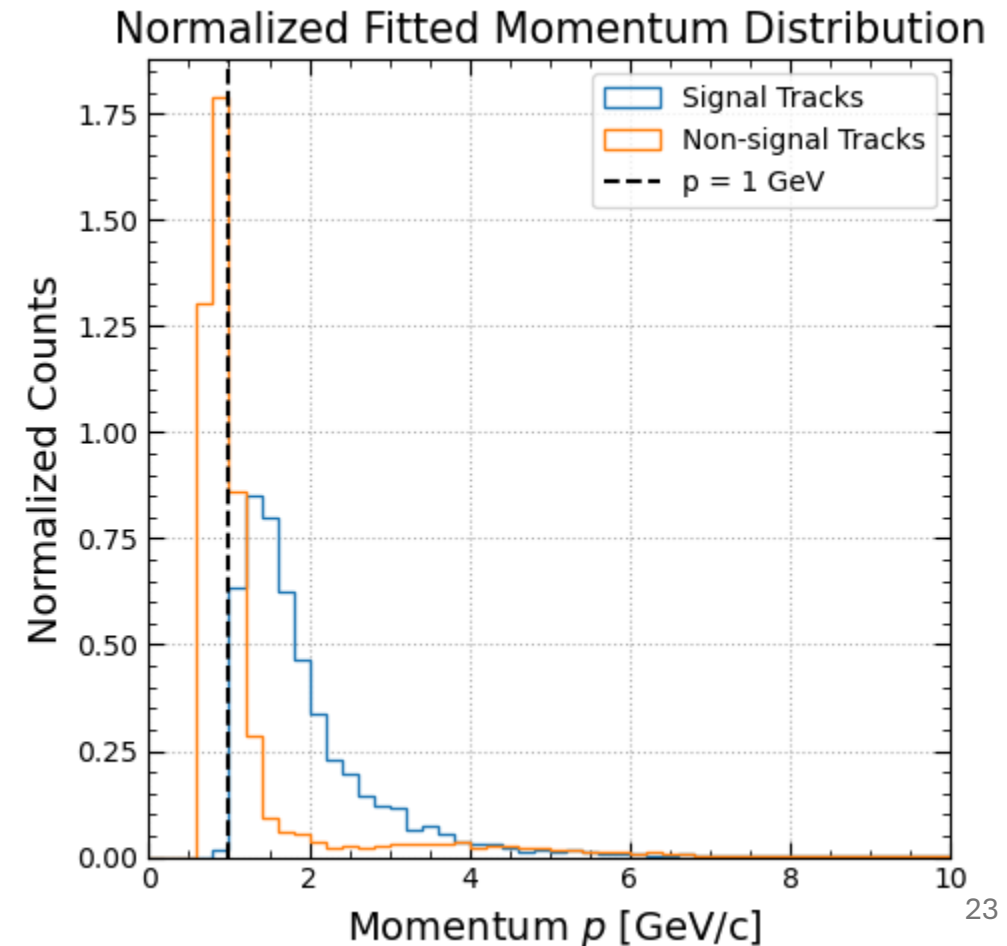
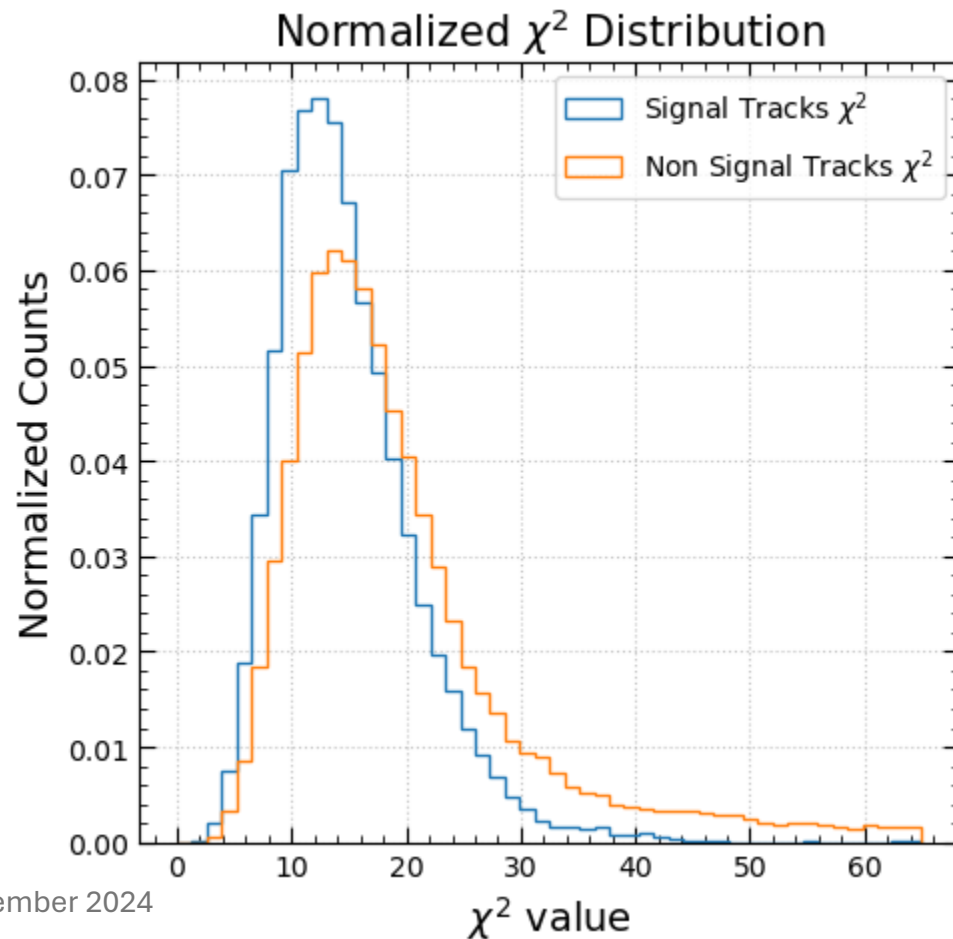
Performance are shown as a box plot for 90 events



Reducing Combinatorics
Without affecting efficiency
significantly

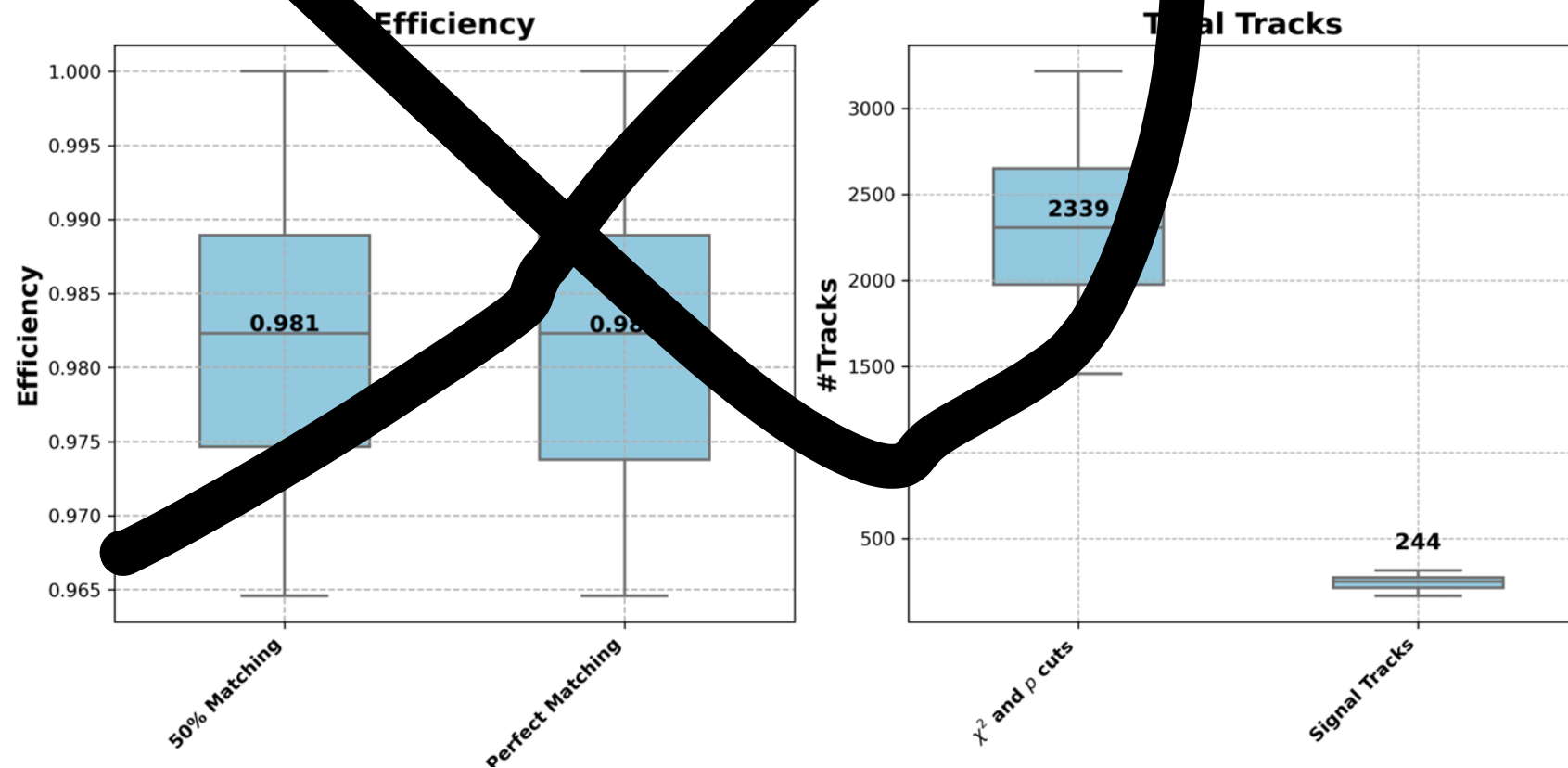
χ^2 and Momentum cuts

- Track fitting algorithm is applied to all track candidates generated by CA and χ^2 and momentum p are obtained.



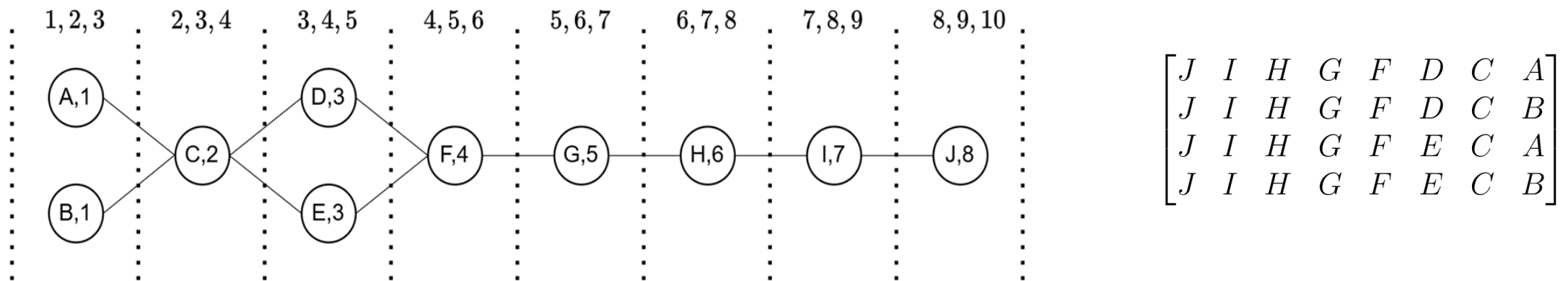
CA metric : after cuts

$\chi^2 < 45$
$p \geq 1 \text{ GeV}/c$



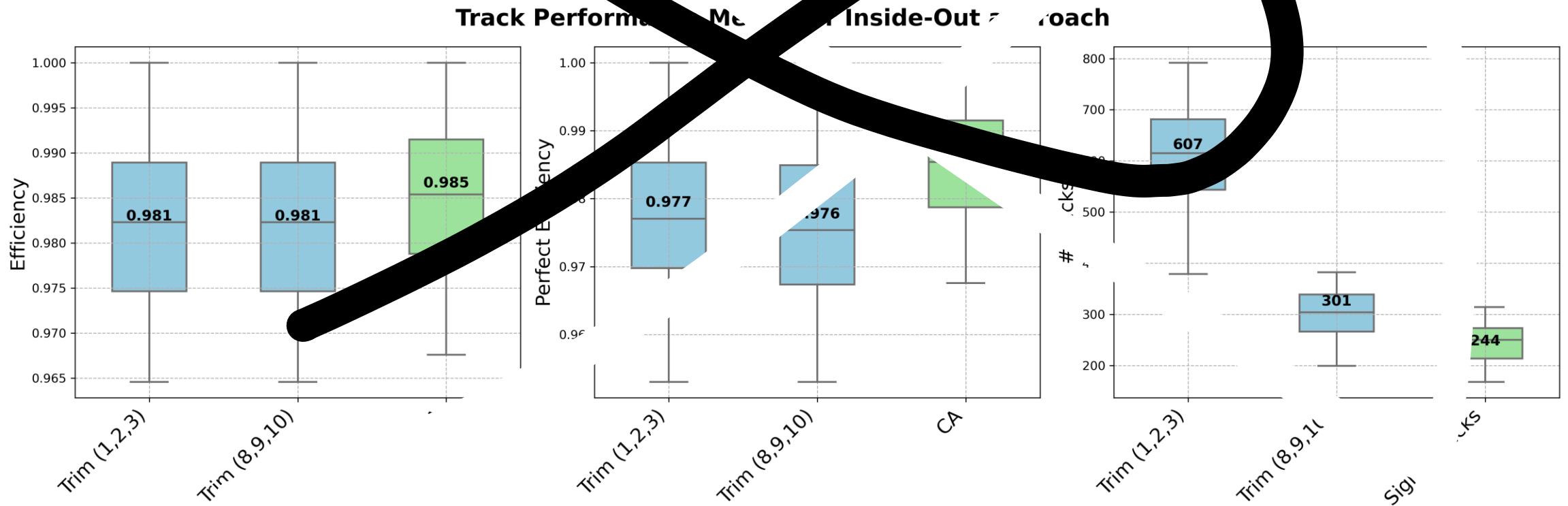
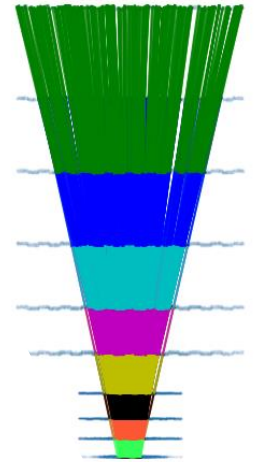
Track Trimming

- Forming tracks with CA is an optimization problem that has no free ends that give rise to tree-like structure from both end points.



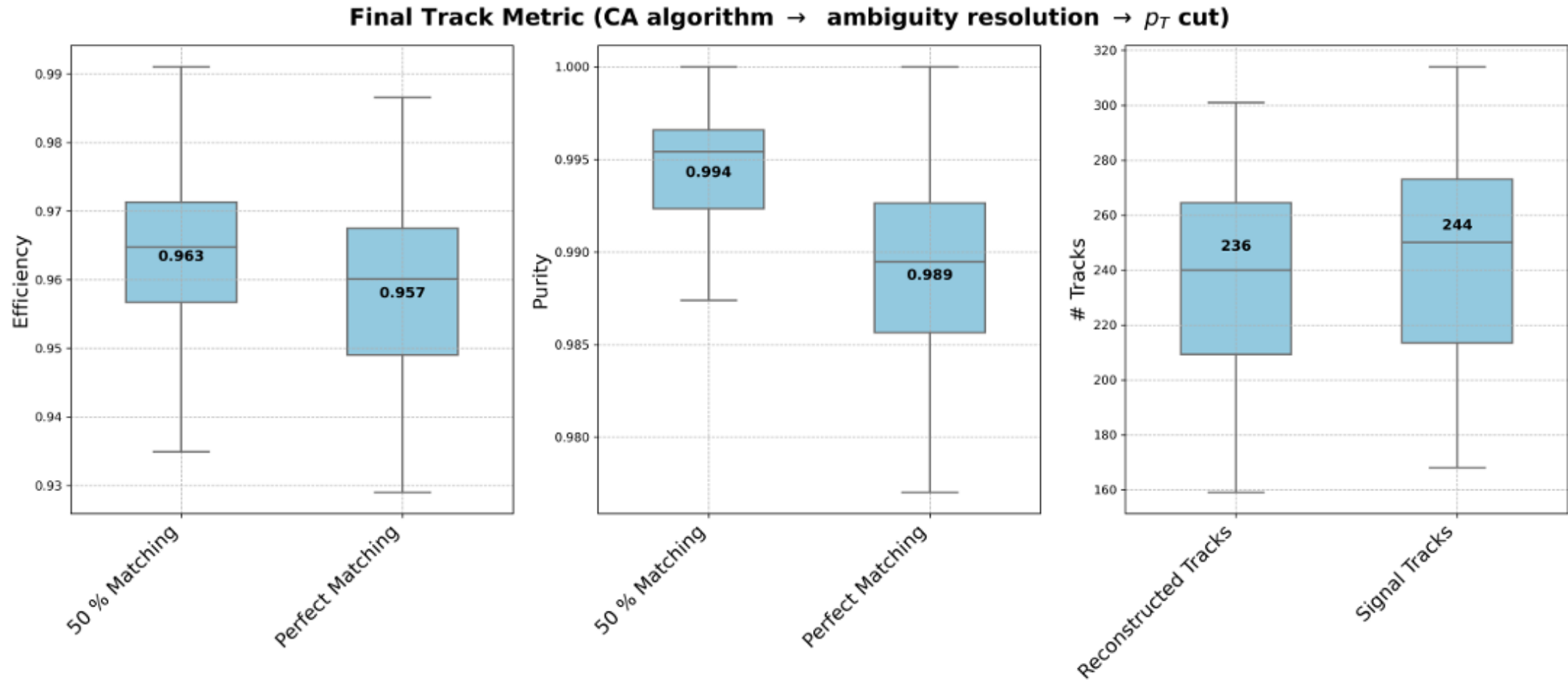
- Trimming** : selecting **one track** from a tree like structure that has a minimum χ^2 . This is performed all possible tree structure from both ends sequentially.
- After trimming from both ends, no track candidates have any triplet common at the end points

In-side out Approach



Final Results

- Dealing with “track sharing hits” and applying p_T cut on the tracks.



CA Algorithm + Ambiguity Resolution

- Reducing combinatorics from CA algorithm.

Stage	Mean Efficiency (%)		Mean Purity (%)		Reconstructed Tracks
	50% Matching	Perfect Matching	50% Matching	Perfect Matching	
CA algorithm	98.5	98.5	-	-	4235
χ^2 and p cut	98.1	98.1	-	-	2339
Trim(1,2,3)	98.7	97.7	-	-	607
Trim(8,9,10)	97.1	97.6	-	-	301
Track Sharing hits	98.1	97.6	79.5	79.1	300
p_T cut = 1 GeV	96.3	95.7	99.4	98.9	236

Summary and Outlook

- CA was applied in the barrel region of the TrackML dataset to specifically reconstruct a longest track chain in the barrel region.
- The CA algorithm itself gives an impressive 98.5 % avg. perfect matching eff.
- The Ambiguity resolution reduces the number of track candidates significantly and improves the avg. perfect purity to 99 % by reducing eff. About ~ 2.5 %. Can be improved !
- The CA can also be studied for entire tracking detectors with forming track candidates with varying number of hits.
- Triplet formation, triplet fitting , CA and track fitting algorithm are both can be accelerated with parallel computing hardware.



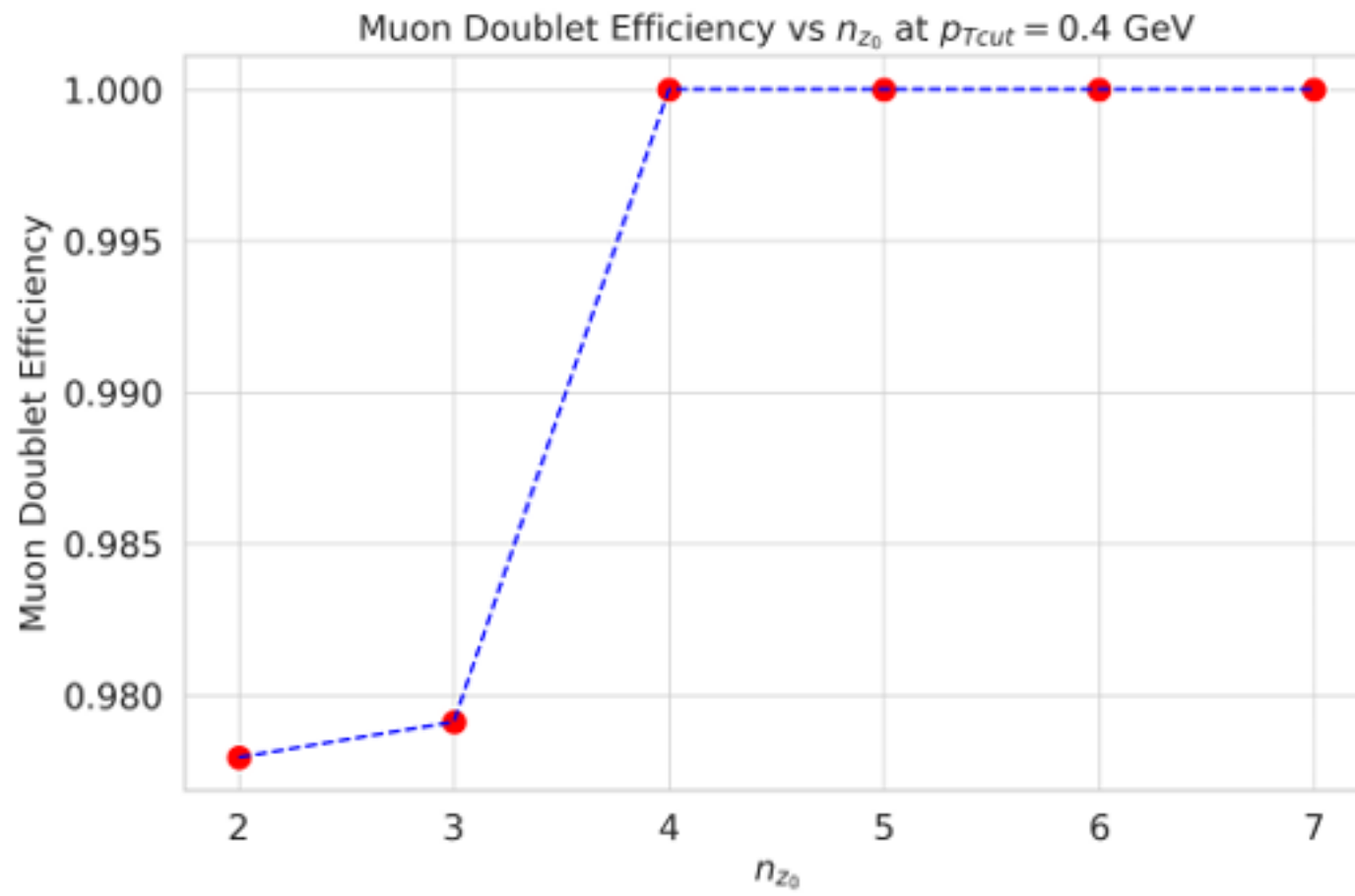
The ideal reasoner would, when he had once been shown a single fact in all its bearings, deduce from it not only all the chain of events which led up to it but also all the results which would follow from it.

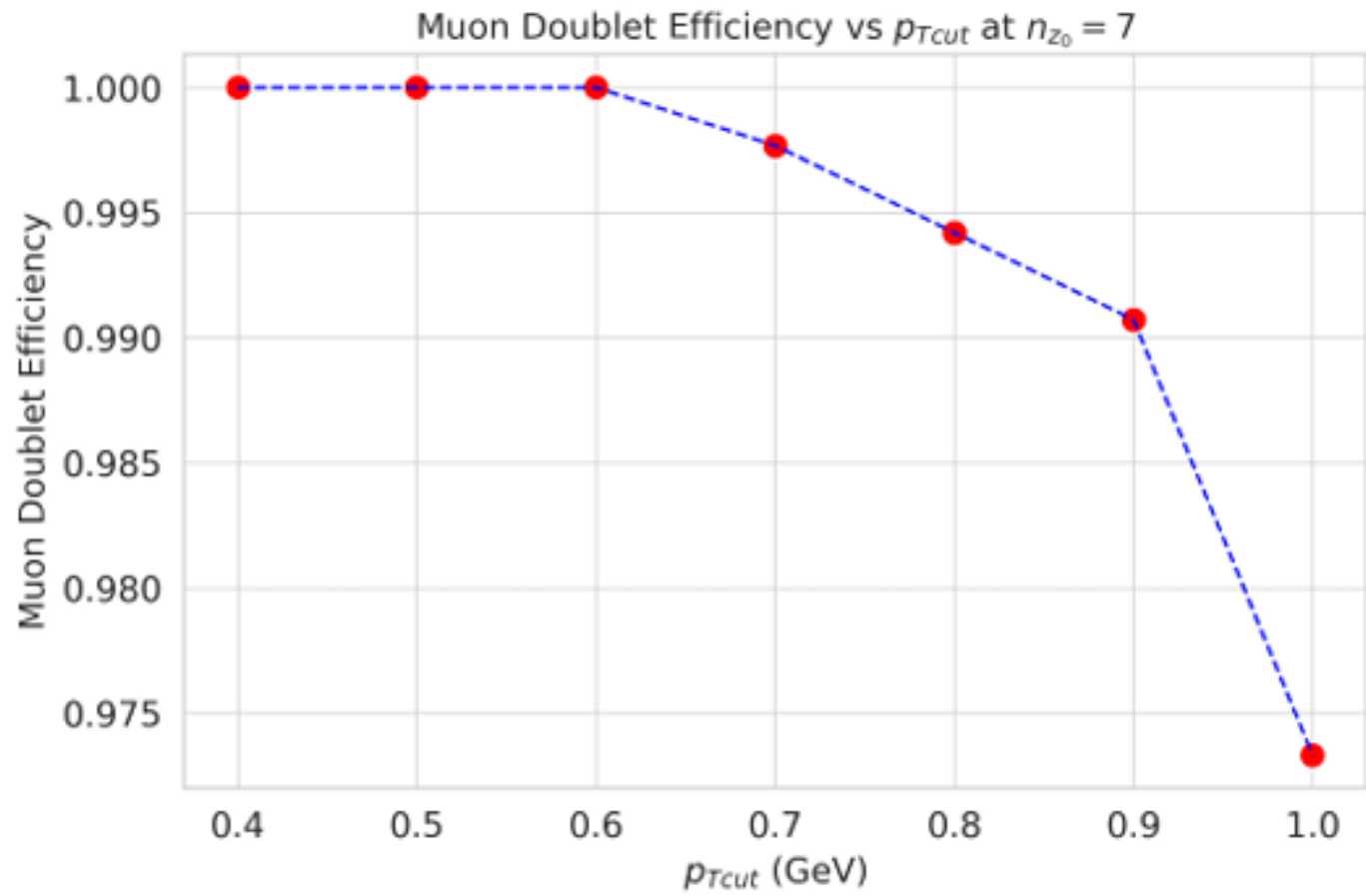
The Five Orange Pips, The Adventure of Sherlock Holmes

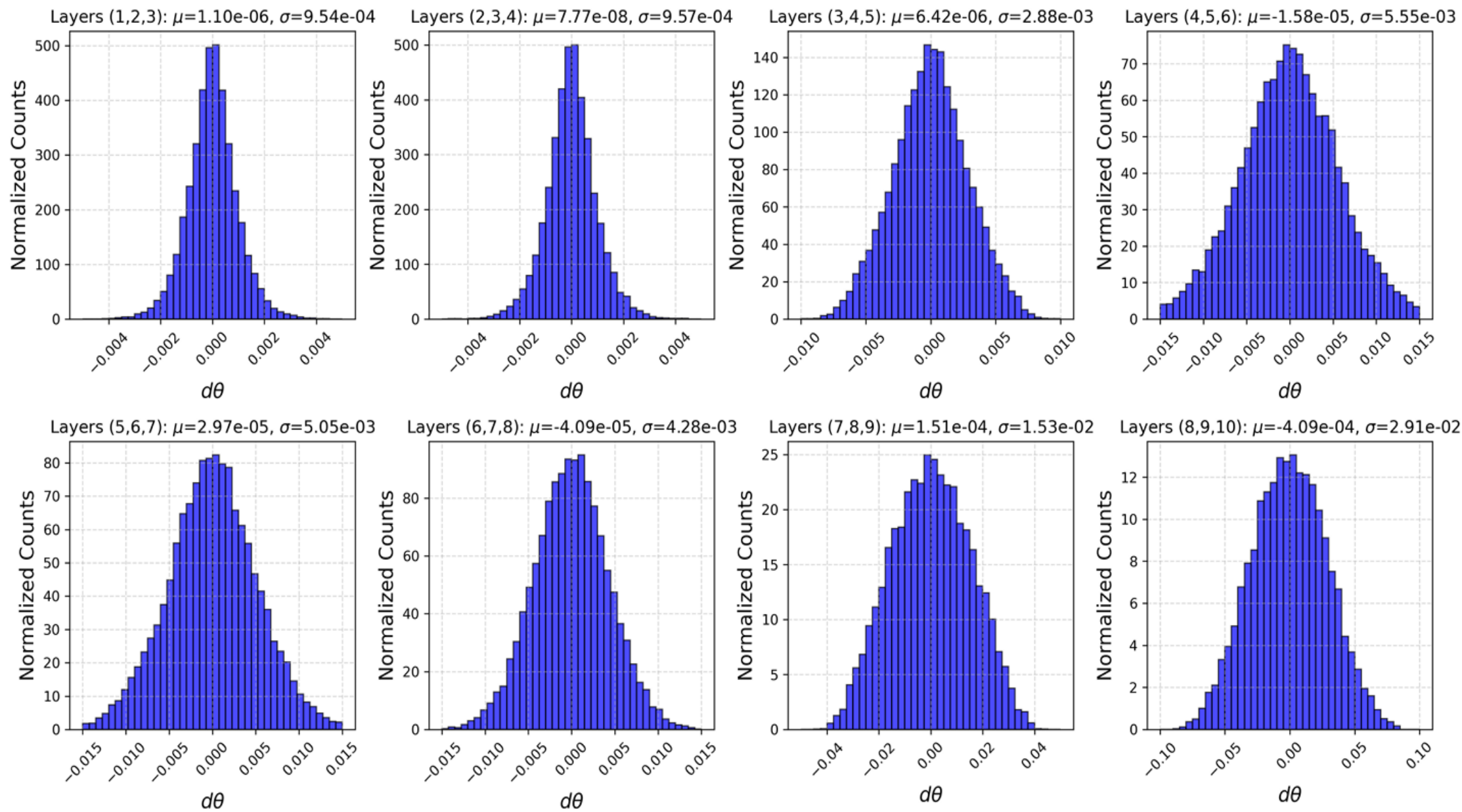
BACKUP

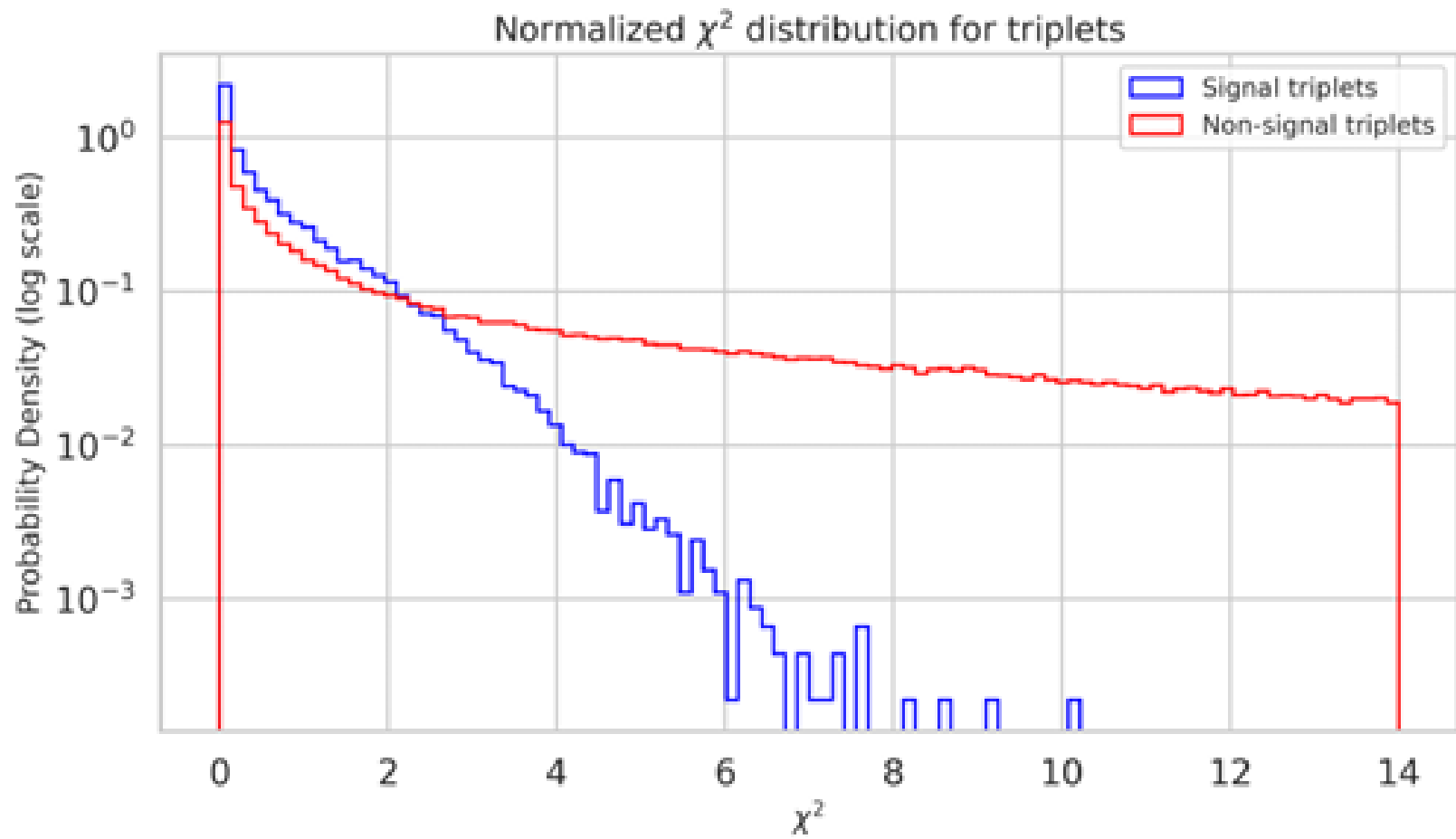
References

- I. Abt et al. “CATS: a cellular automaton for tracking in silicon for the HERA-B vertex detector”. In: Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 489.1-3 (Aug. 2002)
- Andre Schöning. A General Track Fit based on Triplets. 2024. arXiv: 2406.05240 [physics.ins-det]. url: <https://arxiv.org/abs/2406.05240>
- Valentina Akishina. “Four-dimensional event reconstruction in the CBM experiment”. PhD thesis. Frankfurt am Main, Germany: Goethe University Frankfurt, 2019.
- TrackML dataset : [The TrackML high-energy physics tracking challenge on Kaggle](#)

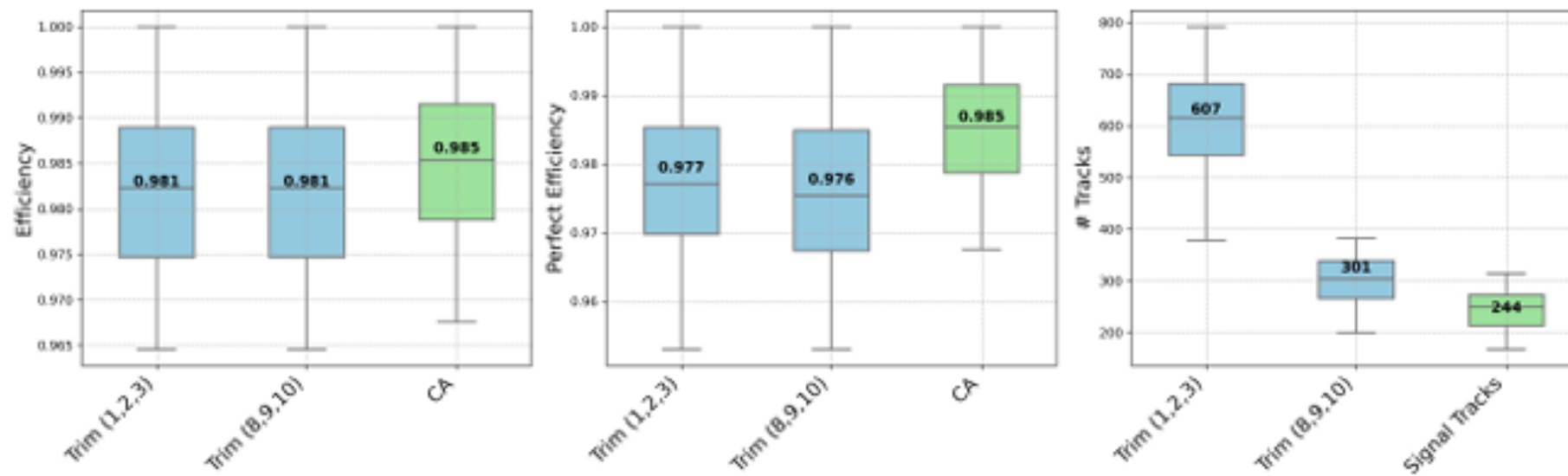




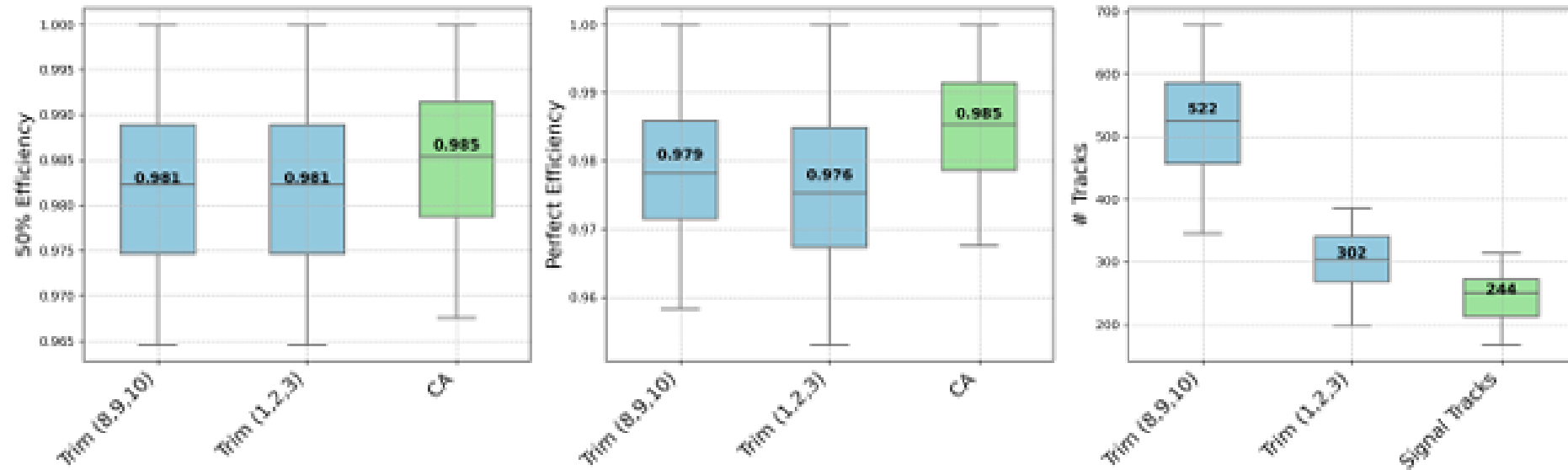


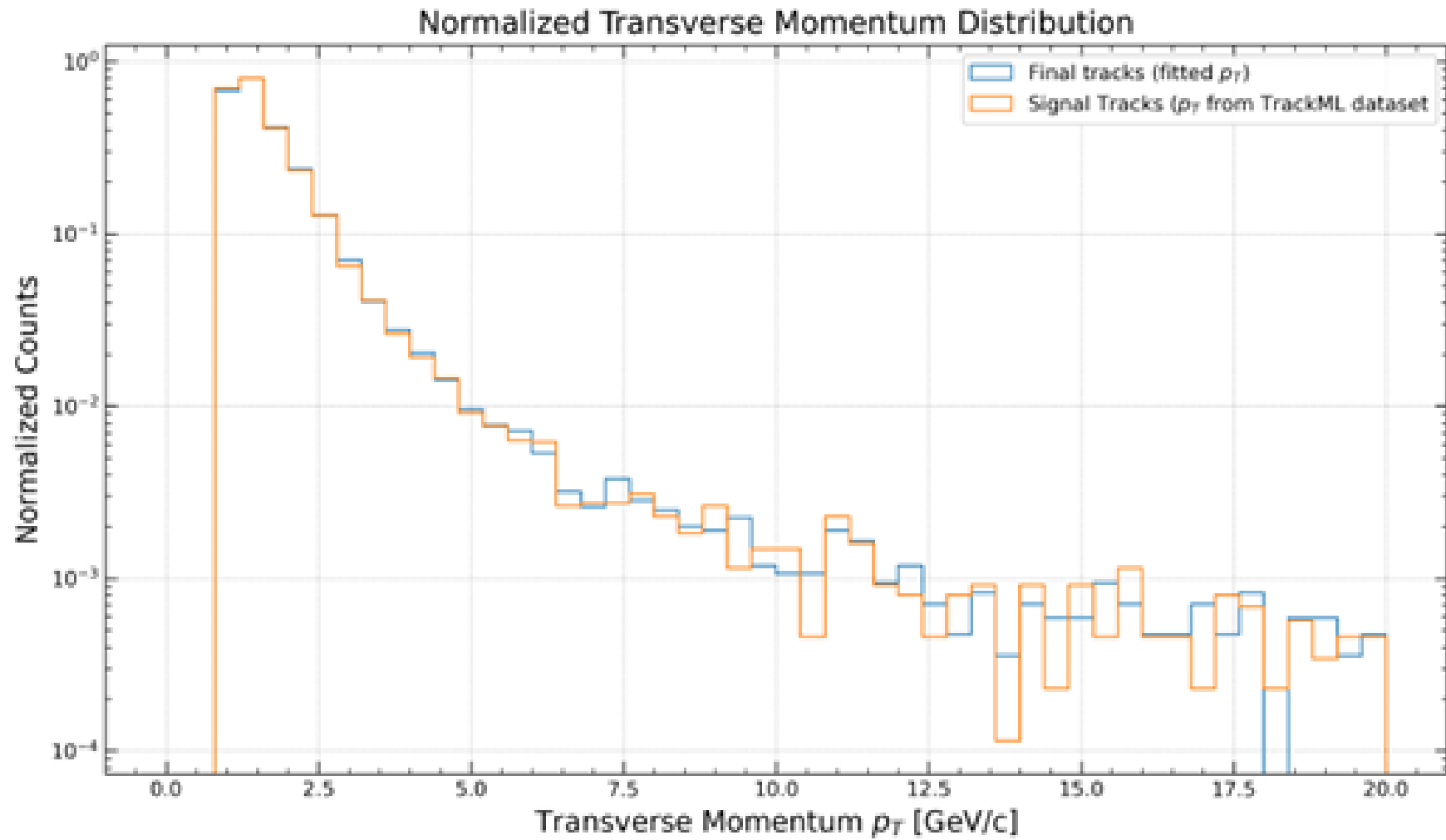


Track Performance Metric for Inside-Out approach



Track Performance Metric for Outside-in approach





SLIDES from : CHEP , ABHI

General Triplet Track Fit

- Generalized from multiple Coulomb scattering (MS)-only triplet fit [1] to include hit uncertainties [2].
- Fit quality for a track:

$$\chi^2(p, \vec{\delta}) = \sum_{\text{scatt. } k=0}^{n_{\text{scatt.}}-1} \left(\frac{\Delta\Phi_{\text{MS},k}^2}{\sigma_{\Phi_{\text{MS},k}}^2} + \frac{\Delta\Theta_{\text{MS},k}^2}{\sigma_{\Theta_{\text{MS},k}}^2} \right) + \sum_{\text{hit } j=0}^{n_{\text{hit}}-1} \vec{\delta}_j^\top V_j^{-1} \vec{\delta}_j$$

$$\vec{\delta}_j = \vec{r}_{\text{fit},j} - \vec{r}_{\text{meas},j}$$

- The fit of the total momentum p & hit position shifts $\vec{\delta}_j$ is done in two steps:
local processing of hit triplets & a global track fit.

