# GNN application in drift chamber of BESIII/STCF

Xiaoqian Jia[1], **Xiaoshuai Qin**[1], Teng Li[1], Xiaoqian Hu[1], Qidong Zhou[1], Shuangbing Song[1], Xingtao Huang[1], Xueyao Zhang[1] , Yao Zhang[2], Ye Yuan[2]

**[1]Shandong University**

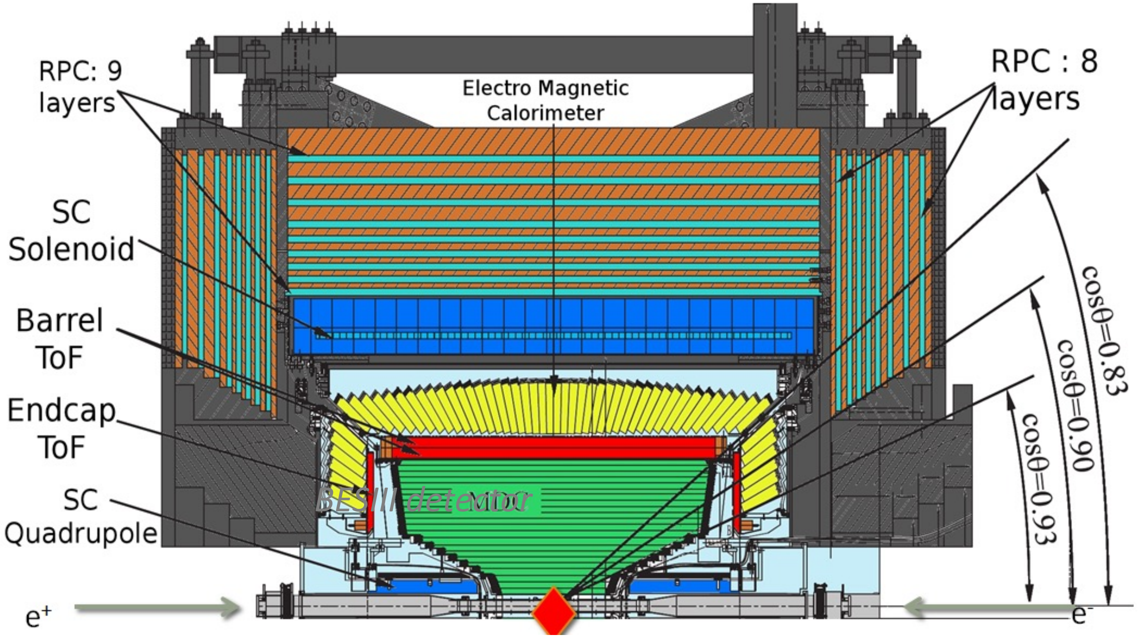
**[2]IHEP**

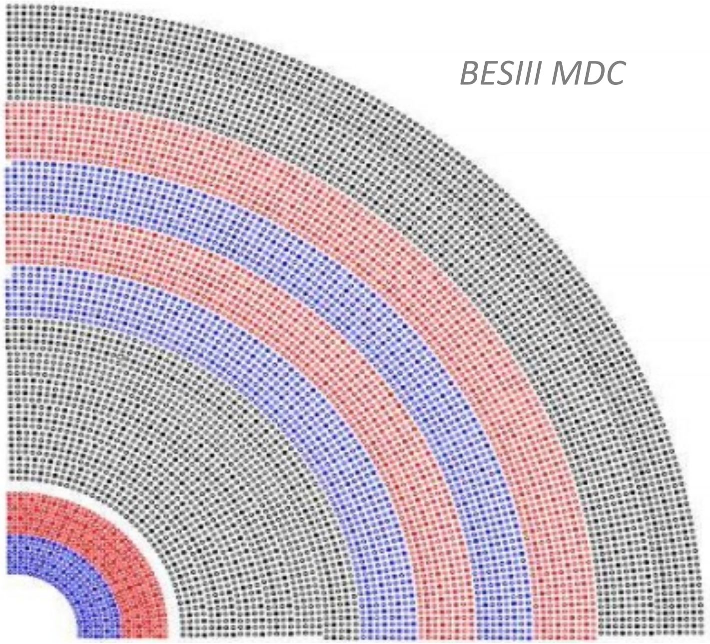
**2nd workshop on Tracking in Particle Physics Experiments**

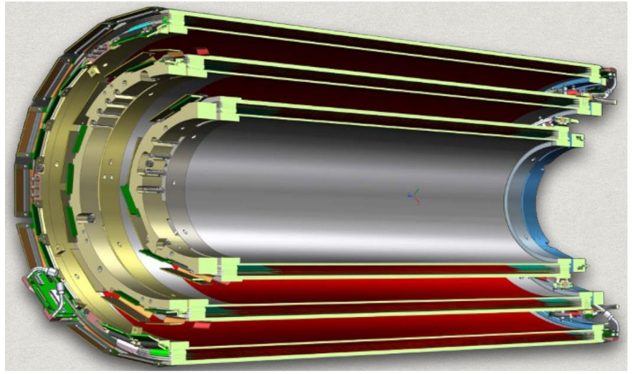
2025-07-22

# BESIII tracking system

- Beijing electron-positron collider (BEPCII)
  - Peak luminosity : $10^{33}\ cm^{-2}s^{-1}$
  - CMS: 2.0 - 4.95(5.6) GeV, $\tau$-charm region
- Main Drift Chamber (MDC) at BESIII
  - 43 sense wire layers
  - 5 axial wire super-layers, 6 stereo wire super-layers
  - $dE/dx$ resolution : 6%
  - Momentum resolution : 0.5%@1GeV/c



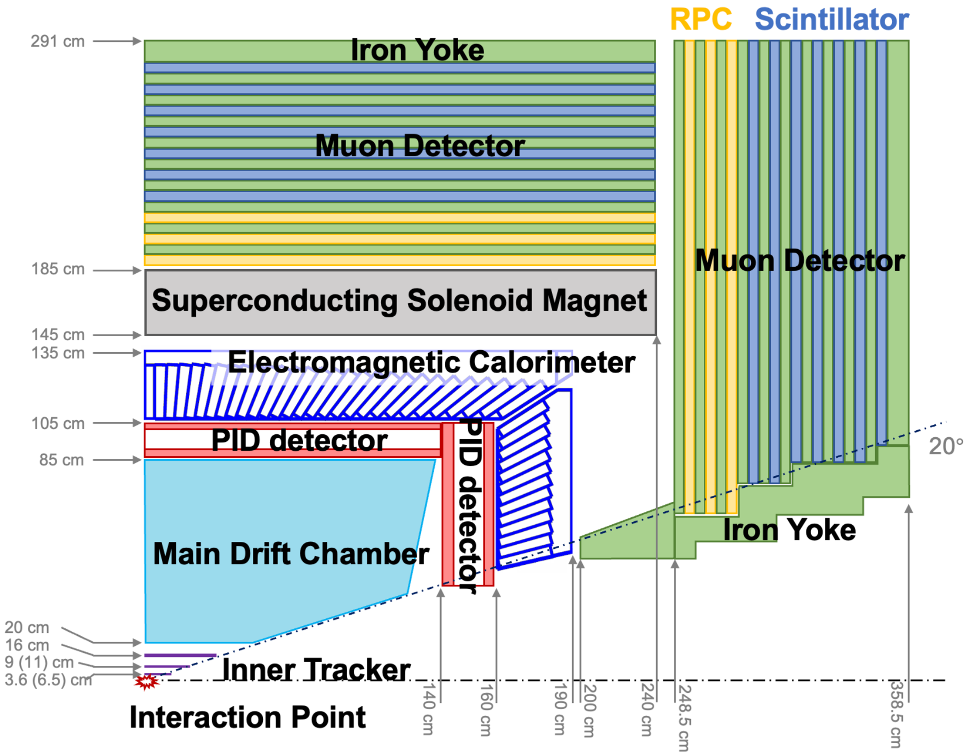



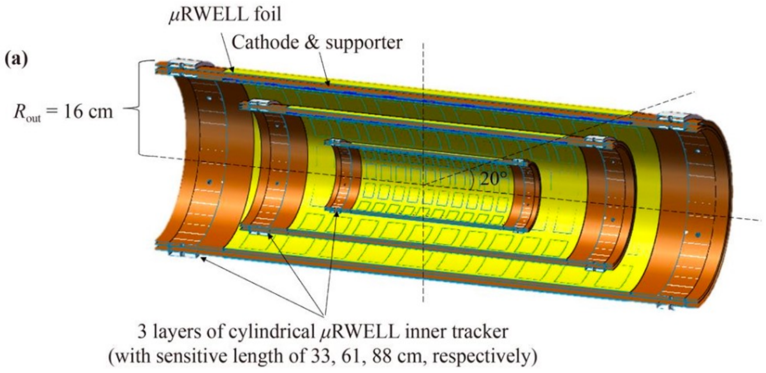



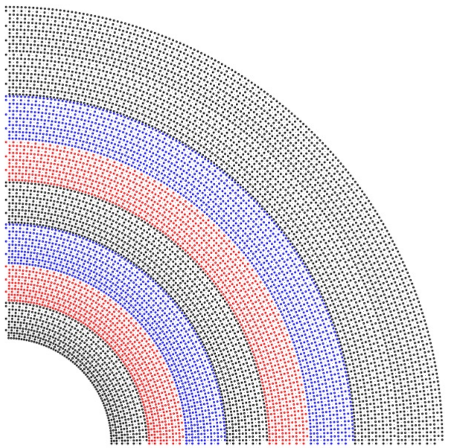
CGEM inner tracker

# STCF tracking system

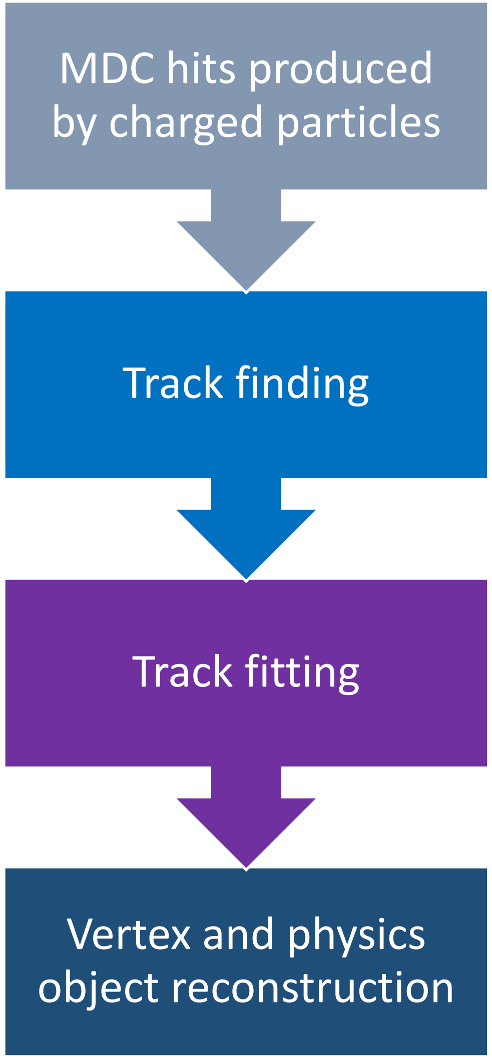
- Super Tau-Charm Facility (STCF)
  - High Luminosity: > $0.5 \times 10^{35}$ cm$^{-2}$ s$^{-1}$@4GeV
  - CMS: 2.0 - 7 GeV
- Main Drift Chamber (MDC) at STCF
  - 48 sense wire layers
  - 4 axial wire super-layers,4 stereo wire super-layers
  - $dE/dx$ resolution : ~6%
  - Momentum resolution : 0.5%@1GeV/c
- Inner tracker
  - ITK: $\mu$RWELL, CMOS MAPS



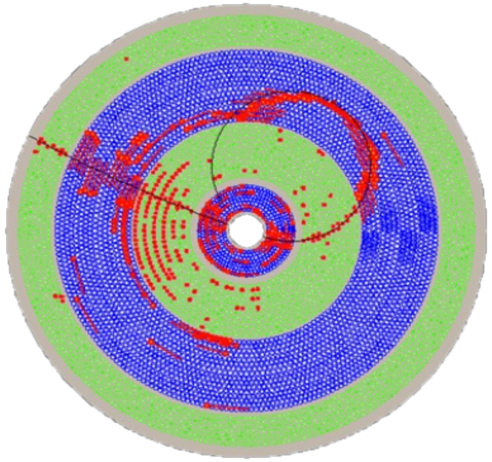



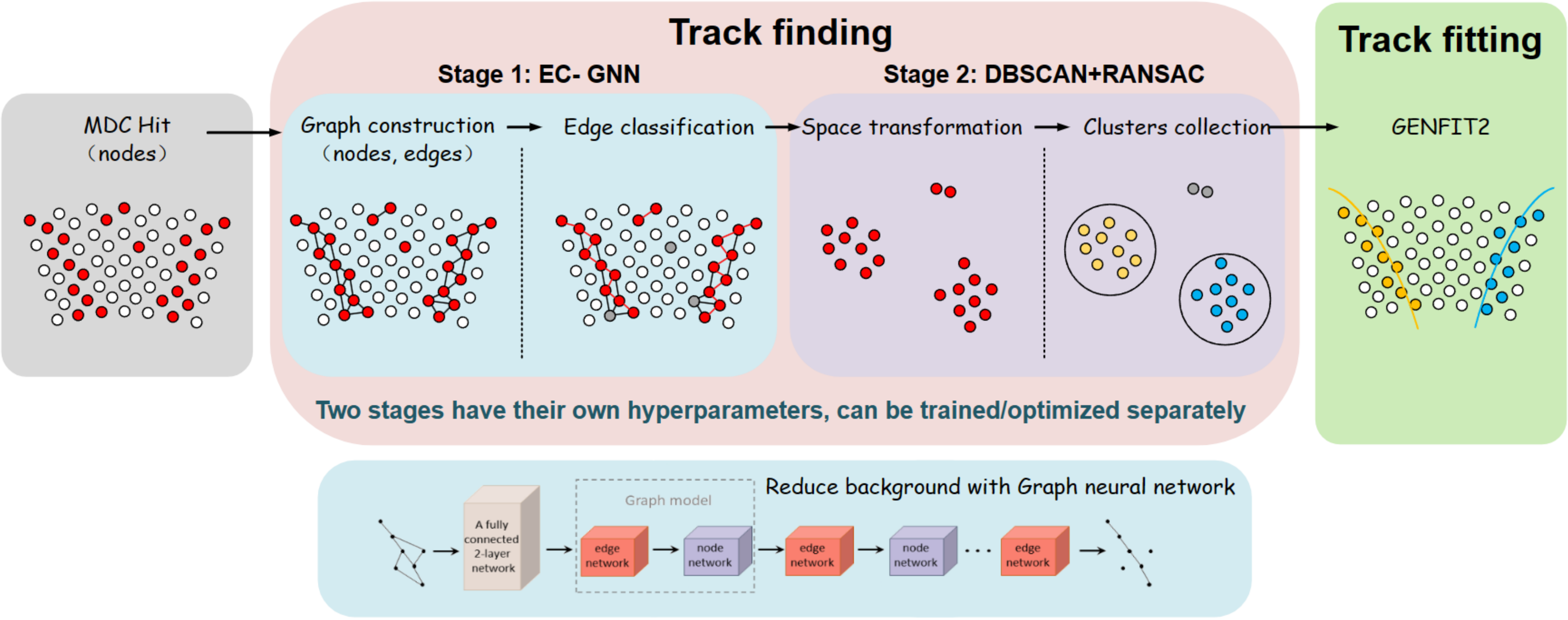
# Traditional tracking in drift chamber




- Build candidate tracks and perform hits assignment
  - Global approach : Hough Transform (HOUGH)
  - Local approach :
    - Template Matching  (PAT)
    - Track Segment Finding (TSF)
    - Combinatorial Kalman Filter (CKF)
- Estimate the track parameters
  - Global fit : Least Square Method, Runge-Kutta Method
  - Recursive fit : Kalman filter

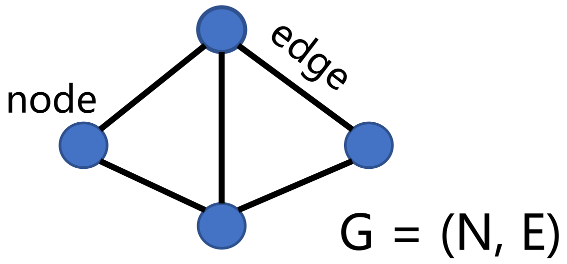


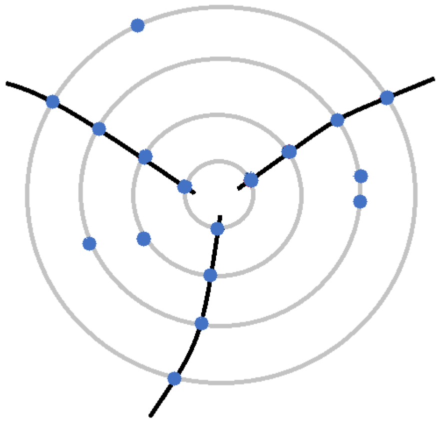
# Methodology: GNN based tracking pipeline



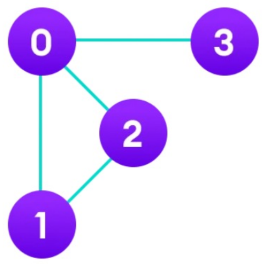
# Graph representation

- A type of neural network that are specifically designed to operate on graph-structured data
- Graph elements: nodes, edges
- From graph  to track
  - nodes → hits
  - edges → track segments



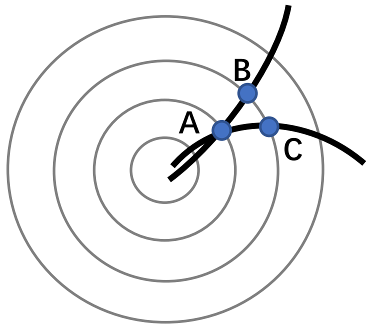
- The storage structure of graphs
  - Adjacency matrix √
  - Adjacency table
  - Orthogonal list
  - Adjacency multiple table
  - Edge set array
  - ……




| | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 2 | 1 | 1 | 0 | 0 |
| 3 | 1 | 0 | 0 | 0 |

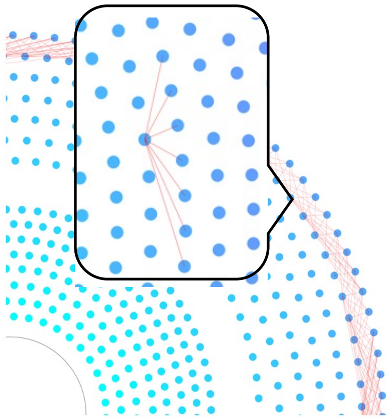
# Graph construction at BESIII

*To reduce the number of fake edges during graph construction*




Pattern Map based on MC simulation at BESIII

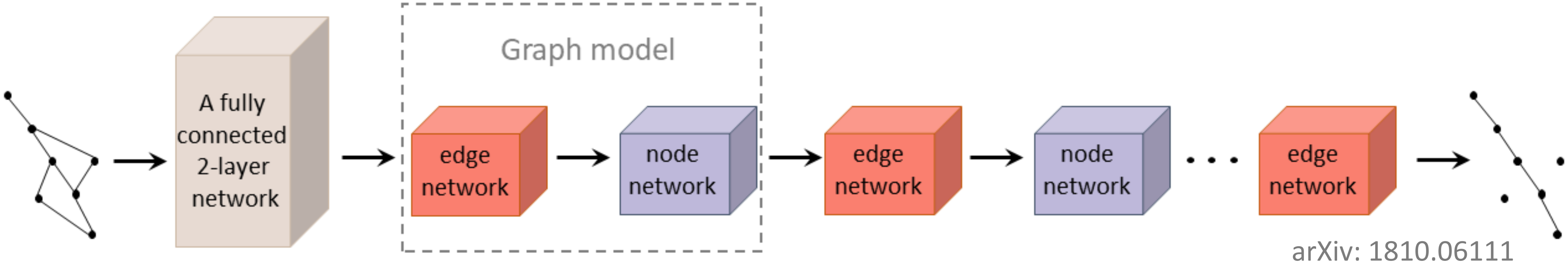
- Definition of valid neighbors
  - Hits on the same layer
    - Two adjacent sense wires on the left and right
  - Hits on the next layer
    - The collection of sense wires that could potentially represent two successive hits on a track
  - To reduce the size of the graphs, the Pattern Map is further reduced based on a probability cut (>1%)
- Edge assignment based on Pattern Map
  - Hit with its neighbors on the same layer and next layer
  - Hit with its neighbors' neighbors on one layer apart
  - Edge label: two hits of this edge belongs to same track or not.
- Graph representation
  - Node features (raw time, position coordinates r, φ of the sense wires), adjacency matrices, edge labels



*A wire on layer13 and its neighbors on layer14*

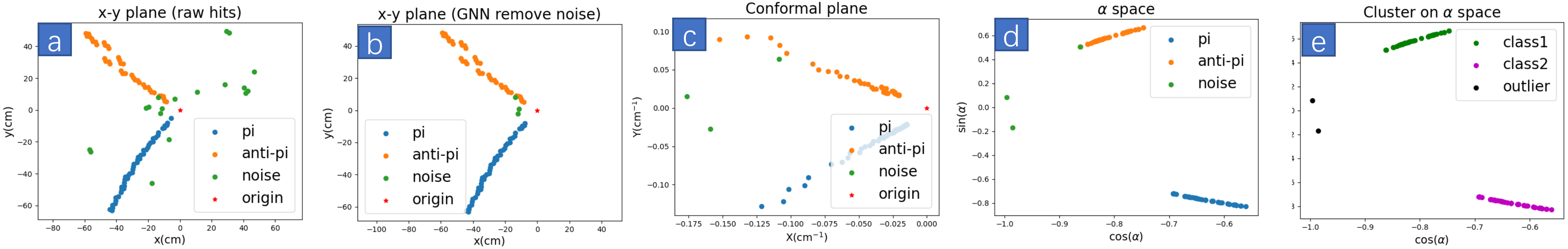
# GNN edge classifier based on PyTorch

- Input network
  - Node features embedded in latent space
- Graph model
  - Edge network computes weights for edges using the features of the start and end nodes
  - Node network computes new node features using the edge weight aggregated features s of the connected nodes and the nodes' current features
  - MLPs
  - 8 graph iterations
  - Strengthen important connections and weaken useless or spurious ones



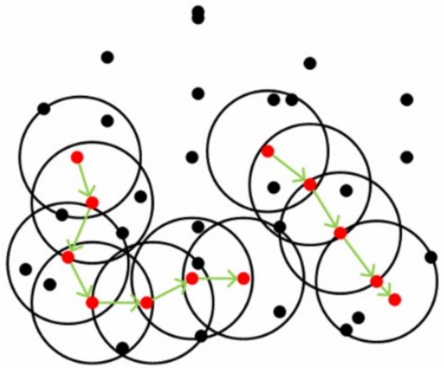

arXiv: 1810.06111

# Clustering based on DBSCAN



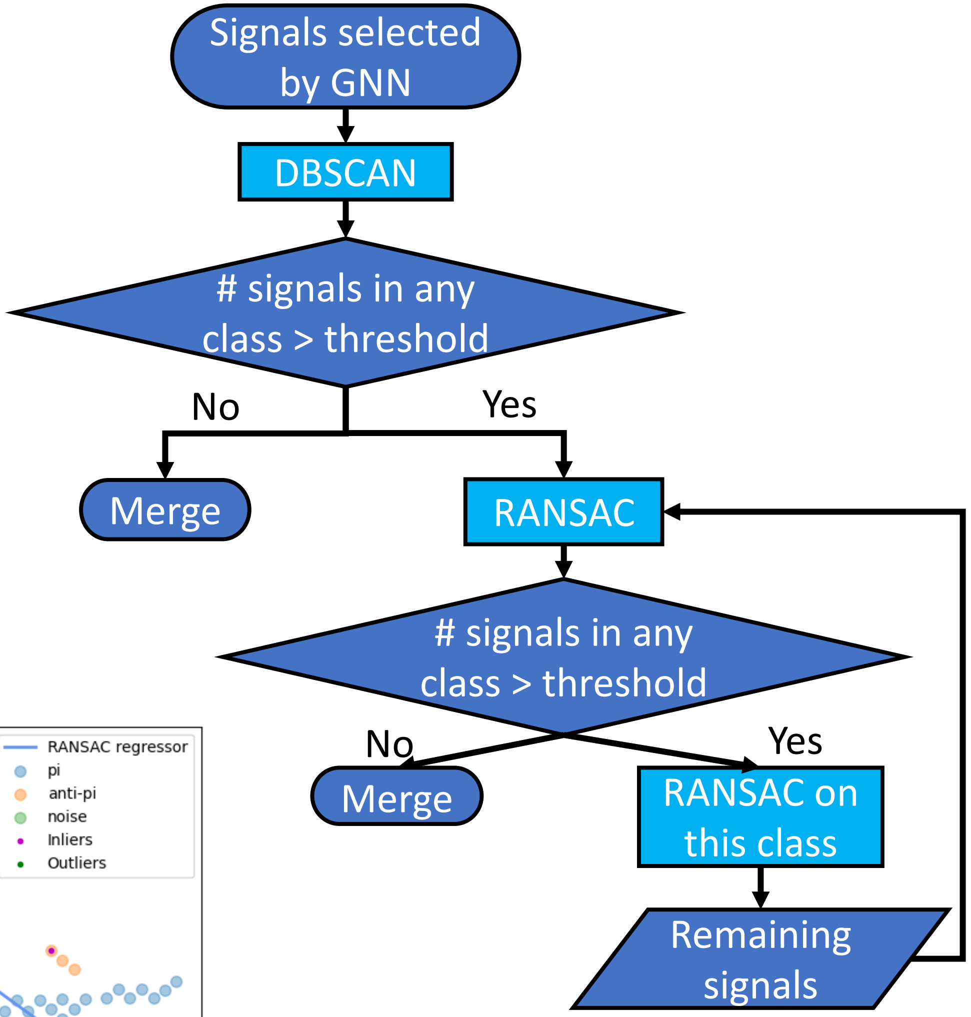

- Transform to Conformal plane
  - $X = \frac{2x}{x^2+y^2}$ $Y = \frac{2y}{X^2+y^2}$
  - Circle passing the origin
  - transform into a straight line
- Transform to 'α' parameter plane
  - Hits connected in the X-Y plane in a straight line
  - α as the angle between the straight line and X axis
  - The parameter space as cosα and sinα

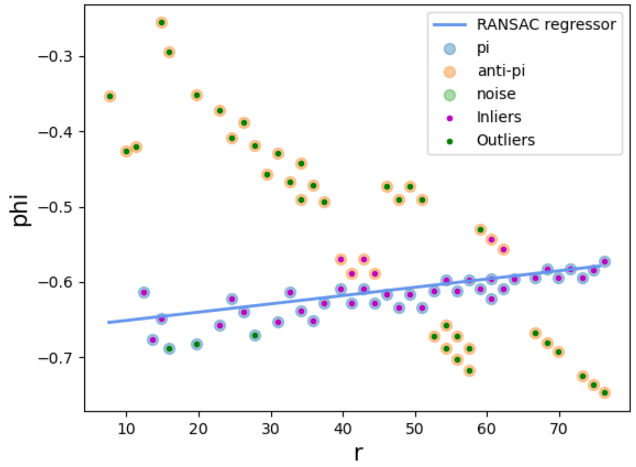
- DBSCAN clustering in 'α'parameter plane
  - Density-Based Spatial Clustering of Application with Noise
  - Hits in a cluster are considered to be in the same track

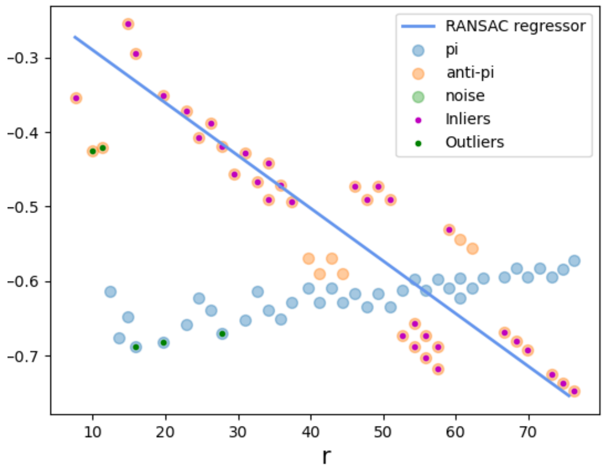


# Clustering salvage algorithm RANSAC

- Random sample consensus (RANCAS)
  - Estimate a mathematical model from the data that contains outliers
  - Its good robustness to noise and outliers
  - Model can be specified
- RANCAS is triggered by the events when DBSCAN fails
  - Polar coordinate space
  - linear model
  - Inliers → a track , outliers → other tracks
  - Stop condition: outliers < threshold

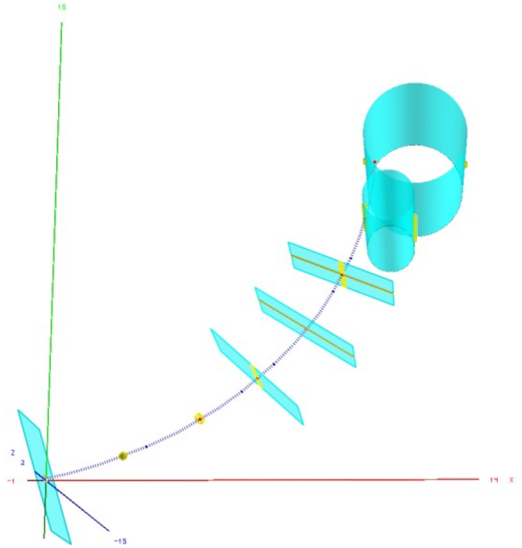


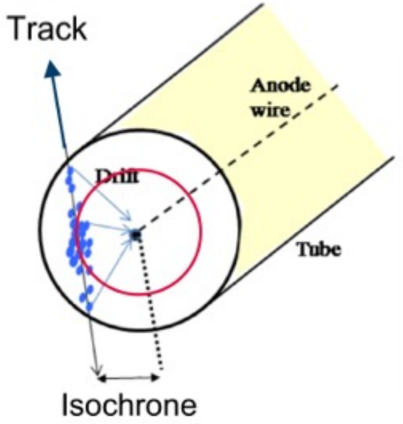



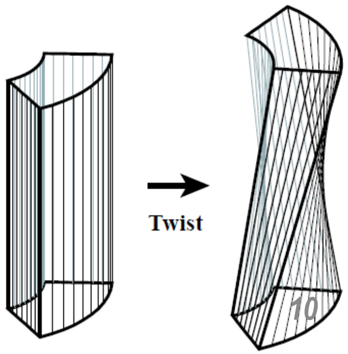
# Track fitting

- Genfit2
  - A Generic Track-Fitting Toolkit
  - Experiment-independent framework
  - PANDA, Belle II, FOPI and other experiments
  - Deterministic annealing filter (DAF) to resolving the left-right ambiguities of wire measurements
- Configuration: Detector geometry and materials; TGeoManager
- Input :
  - Signal wire position, initial values of position and momentum
  - particle hypothesis for $e, \mu, \pi, k, p$
- Fitting procedure:
  - Start 1st try: drift distance roughly estimated from TDC、ADC of sense wires
  - Iteration to update information of drift distance, left-right assignment, hit position on z direction and entrancing angle in the cell et al.

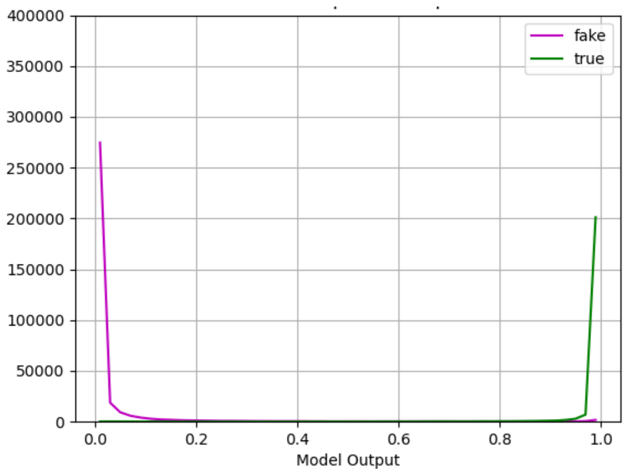




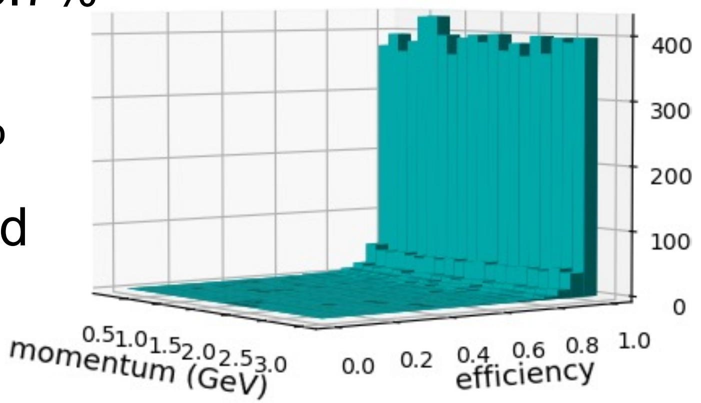

$$t_{\text{drift}} = t_{\text{TDC}} - t_{\text{EST}} - t_{\text{flight}} - t_{\text{wp}} - t_{\text{elec}}$$

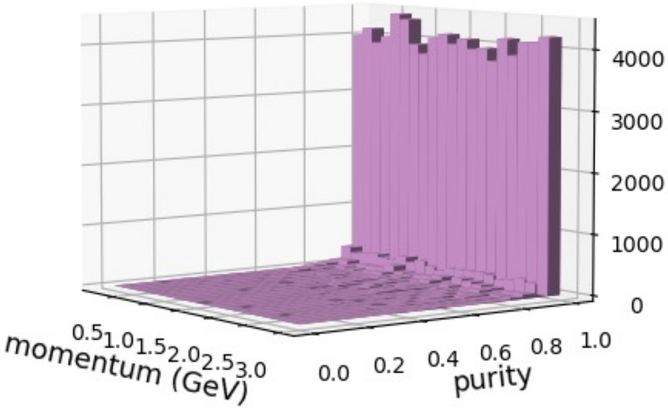


# Performance of filtering noise at BESIII



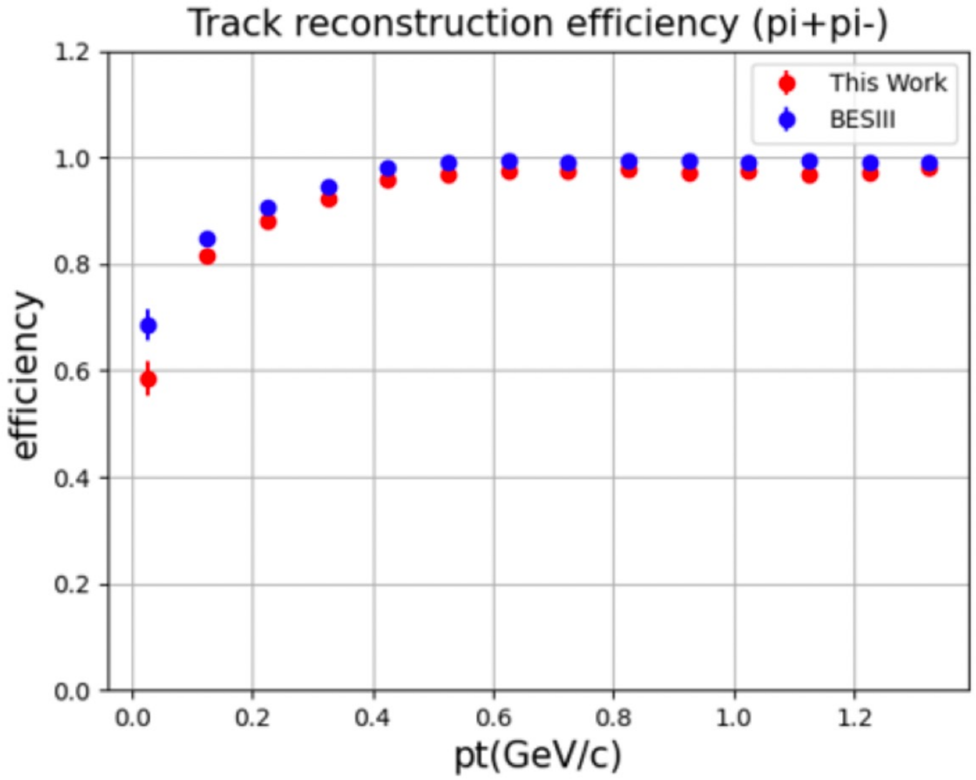

- Datase
  - Single-particle ($e^{\pm}, \mu^{\pm}, \pi^{\pm}, K^{\pm}, p/\bar{p}$) MC sample
  - 0.2 GeV/c < $p$ < 3.0 GeV/c
  - Mixed with BESIII random trigger data as background (~45% hits)
  - Train: Validation: Test = 4: 1: 1
- Hit selection performance
  - The preliminary results show that GNN provides high efficiency and purity of hits selection
  - Hit selection Efficiency : $\frac{N_{signal}^{predicted}}{N_{signal}^{real}}$ 98.7%
  - Hit selection Purity : $\frac{N_{signal}^{predicted}}{N_{all}^{predicted}}$ 96.5%
  - Efficiency and purity can be balanced by adjusting the model parameter

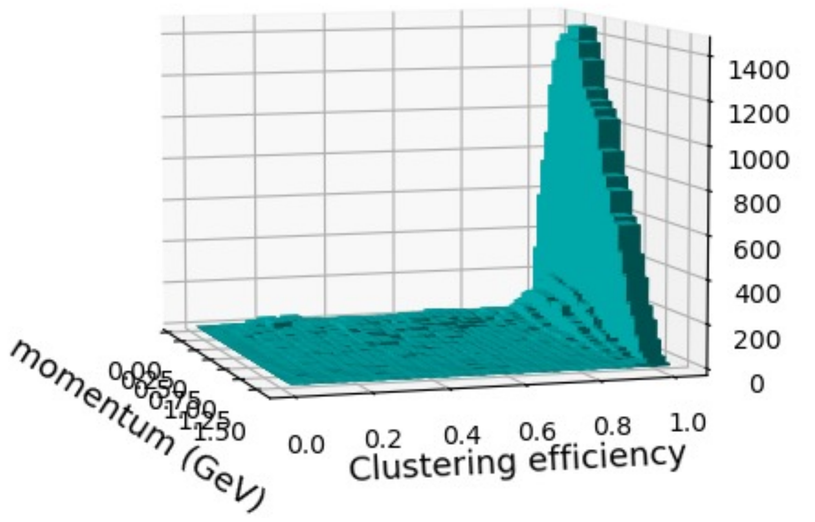


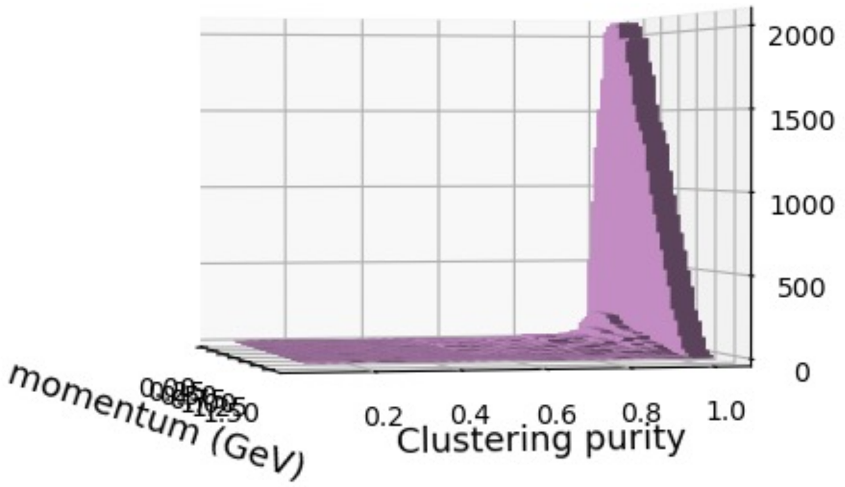



# Preliminary tracking performance at BESIII

- Particle reconstructed performance
  - $J/\psi \to \rho^0\pi^0 \to \gamma\gamma\pi^+\pi^-$ from MC simulation
  - track eff = $\frac{N_{\text{rec tracks}}}{N_{\text{total tracks}}}$

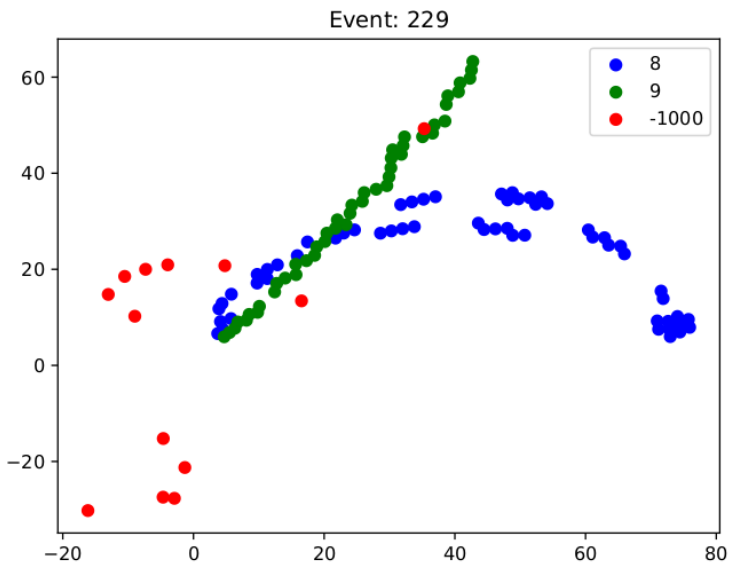


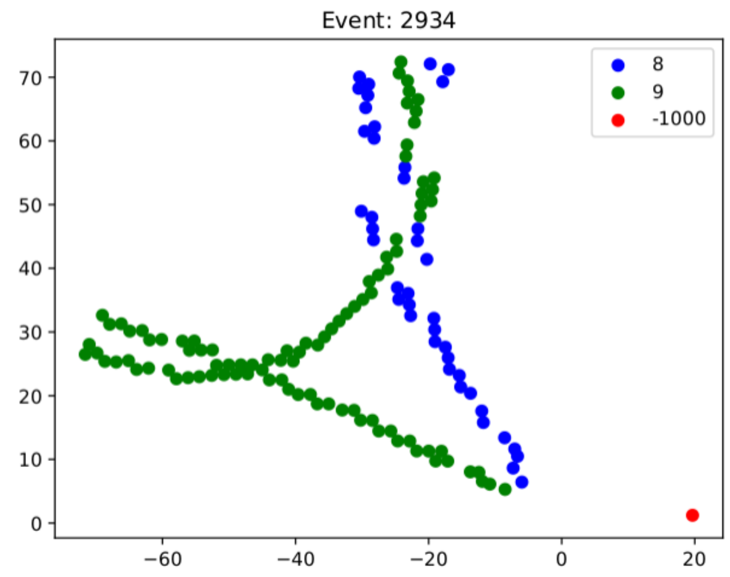



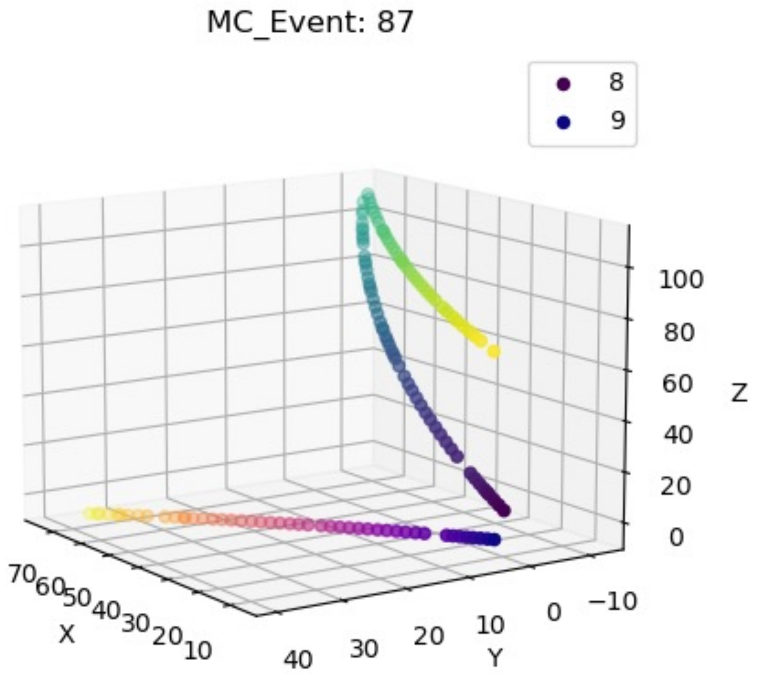



# Sources of efficiency loss

- Efficiency loss during track finding(clustering):
  - multi-circular, decays,
  - interaction with detector boundary/material, scattering
  - 2D crossing tracks or too close to each other

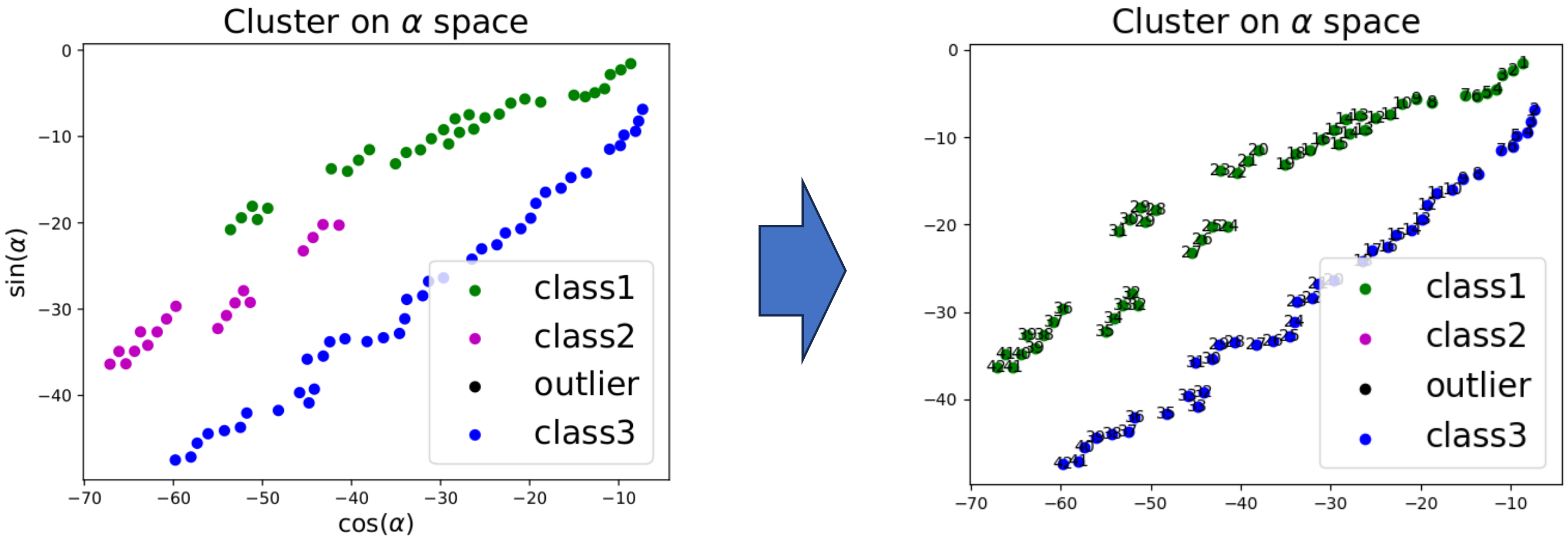


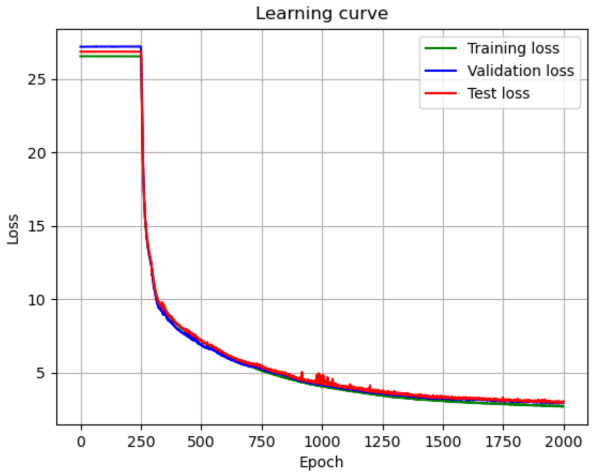
# Further modification of track finding

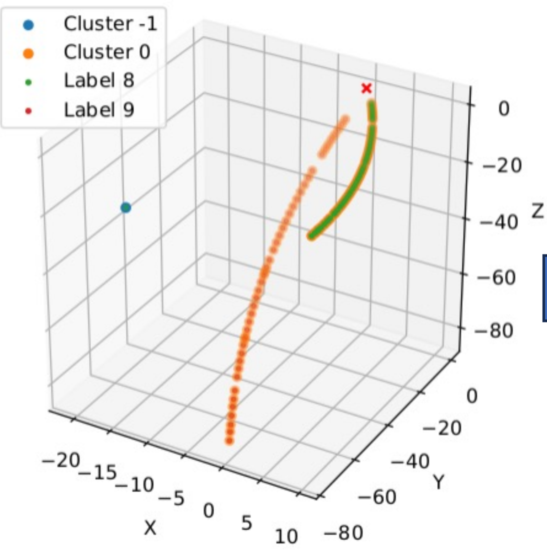
- Since z position of hits is unknown, 2D information has large deviation for stereo wires
- Breaking into parts especially for tracks with large polar angle
- Re-combination at super layers level

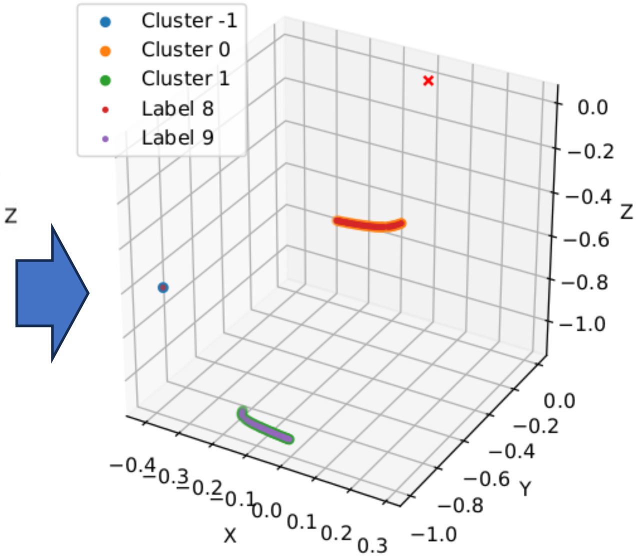


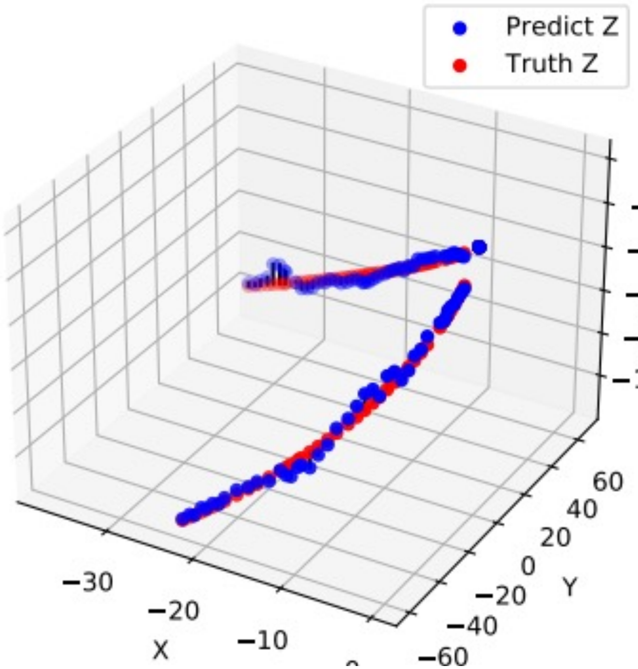
# Z regression and 3D clustering

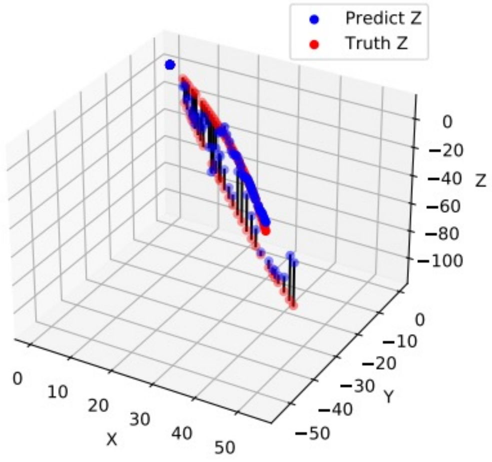
- GNN regression for z coordinate prediction
  - structure similar as edge classification GNN
  - Lost function:
    - averaged distance between predicted and real position
- Clustering:
  - 3D parameter space: sinα， cosα， z/r



# Potential of 3D clustering

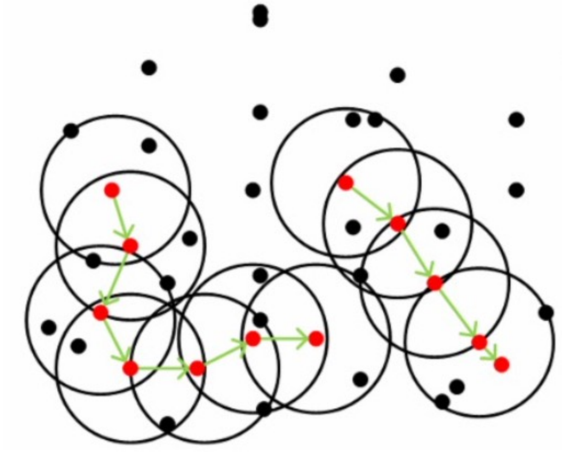
- Parameter space clustering is better than original coordinate space.
- Potential track finding efficiency can reach 97.5% via 3d parameter space clustering
- Good event definition: #hits per track is between 5 and 50, might get rid of circular and large angle scattering

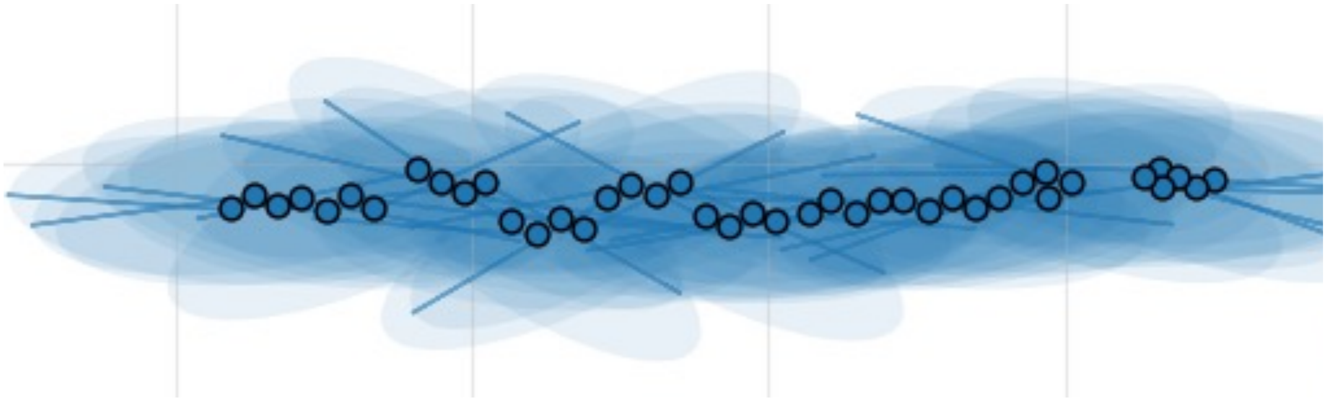
| D+R | 2D | | 3D | | | | |
|---|---|---|---|---|---|---|---|
| | wire XY | Truth XY good event | PredZ wire XY good event | TruthZ wire XY good event | Truth XYZ good event | Truth parameter space | Truth parameter space good event |
| Efficiency (%) | 97.3 | 98.3 | 98.1 | 97.9 | 98.1 | 98.2 | 98.3 |
| Purity (%) | 96.8 | 97.6 | 96.1 | 96.7 | 98.0 | 99.1 | 99.2 |
| Finding Success rate (%) | 83.3 | 90.2 | 71.6 | 84.3 | 91.8 | 92.0 | 97.5 |

# DBSCAN using elliptical neighborhood

- Circular neighborhood is replaced with an elliptical neighborhood
- Local orientation of each point is determined based on PCA considering points within its neighborhood

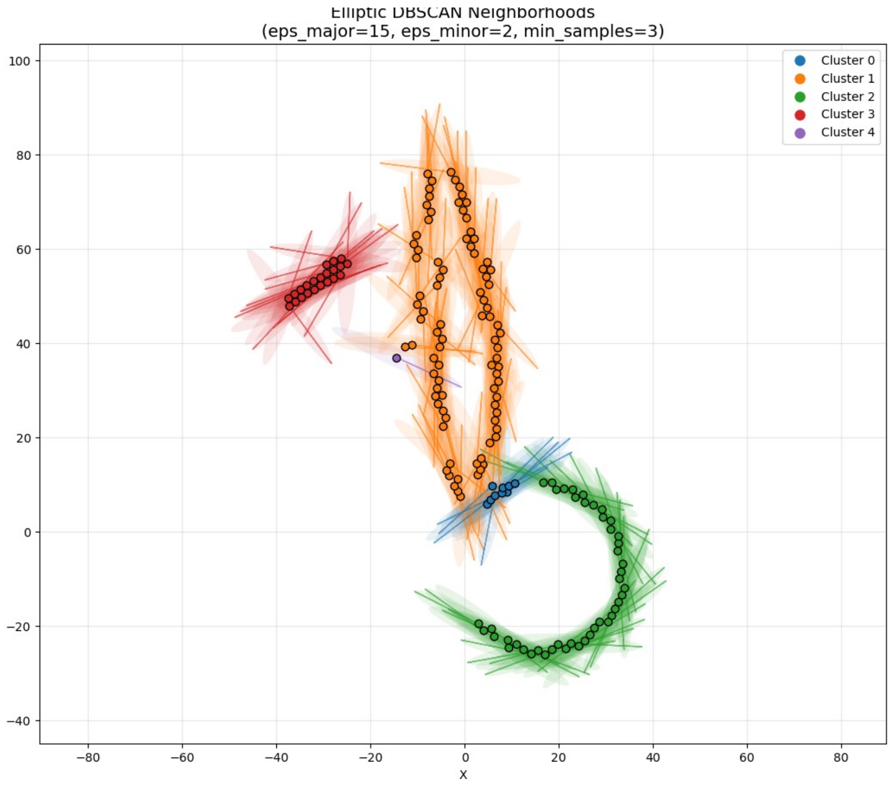
| Parameters | Meaning | value |
|---|---|---|
| eps_major | long axis | 15cm |
| eps_minor | short axis | 10cm |
| min_samples | min neighbors | 3 |
| k (orientation calculation) | neighbors for PCA | 5 |

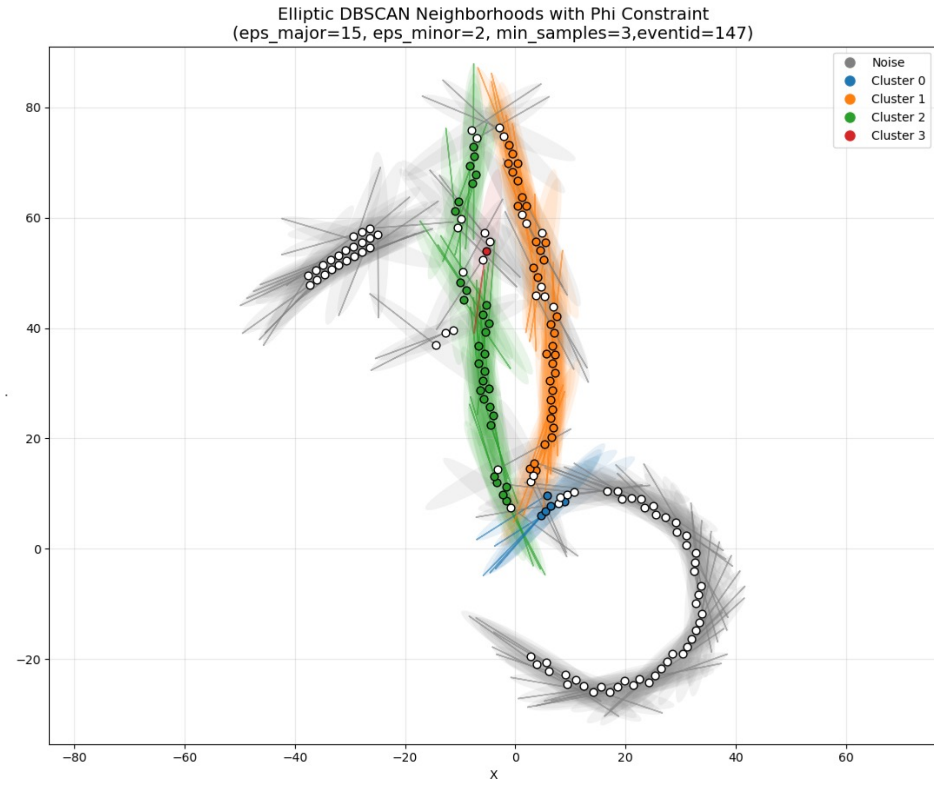




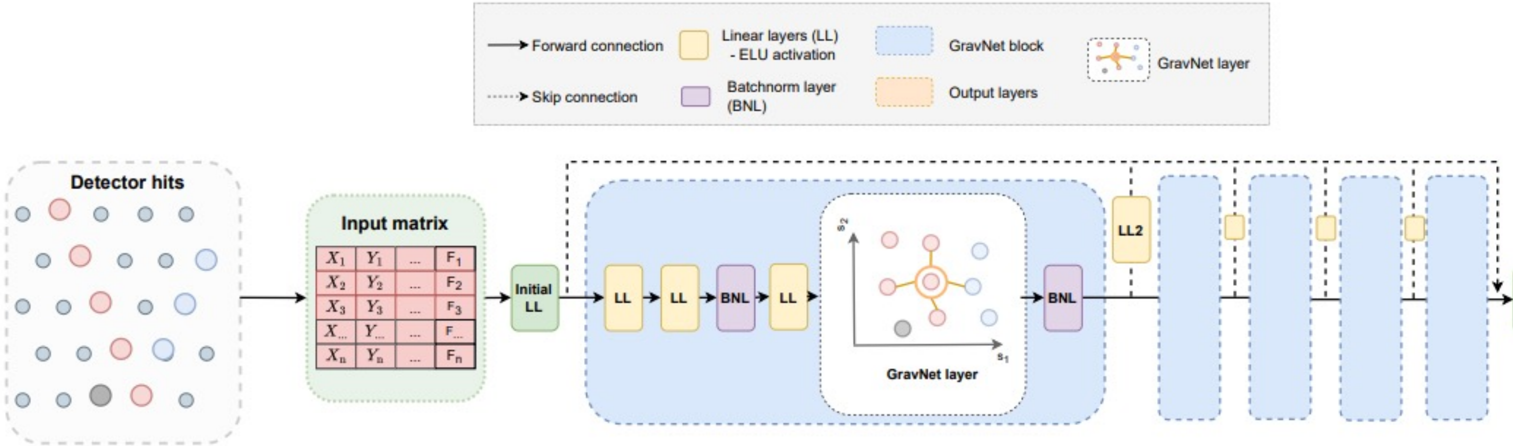
# Split cluster via abnormal point

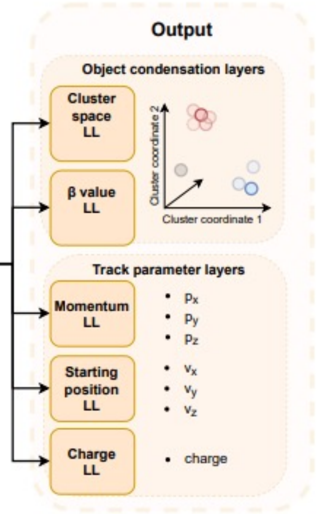
- Abnormal point detection:
  - large di-angle between direction of PCA and direction to the IP
- Further attempts:
  - assign large weight for hits in the same superlayer
  - veto hits in the same payer, same superlayer but another group



# Object Condensation(OC) using GNN



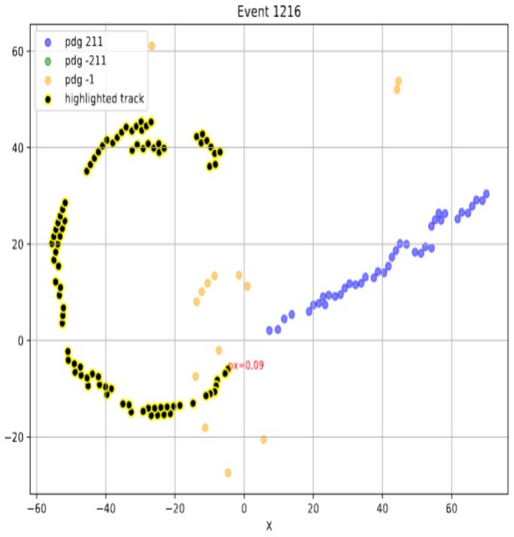

- Input feature： x and y coordinates of hits, layer, superlayer, ADC and TDC
- Network structure：
  - Initial LL with ELU activation and batch normalization.
  - Multiple stacked GravNet blocks with skip connections. Each block includes ：
    - A GravNet layer(GNN layer) that learns a latent space to determine neighborhood relationships and pass message.
    - Linear layers (LL) and batch normalization layers (BNL).
  - Final LL to generate output representations

*Comput.Softw.Big Sci.* 9 (2025) 1, 6

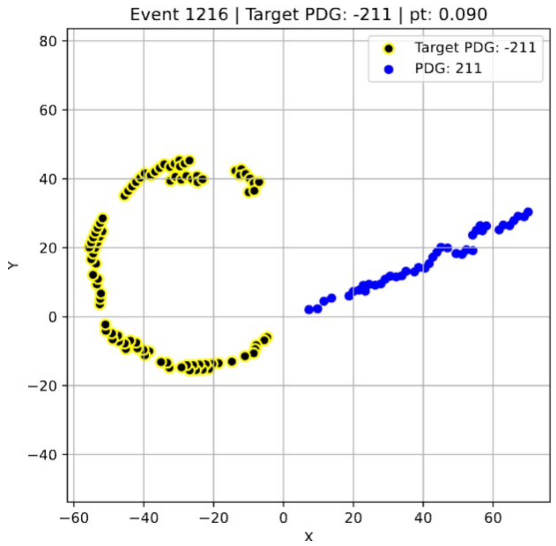
- Output：
  - Object condensation layers ：
    - Cluster coordinates (for grouping hits belonging to the same track).
    - β-values (confidence scores for cluster centers).
  - Track parameter prediction layers： $q, p_x, p_y, p_z, v_x, p_y, v_z$

# Track finding result via OC

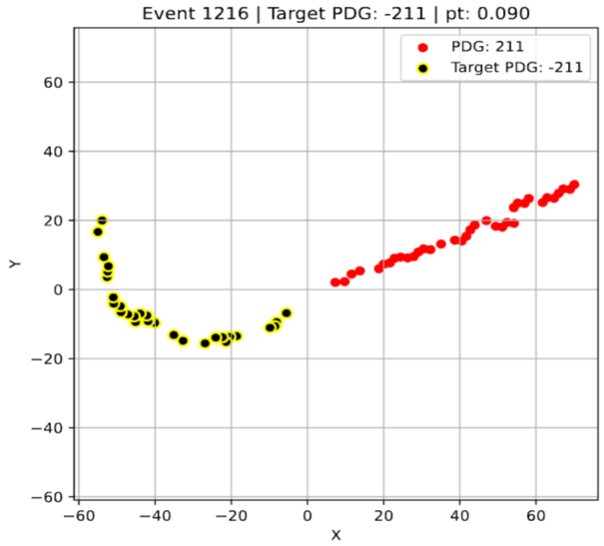
- circling track




truth

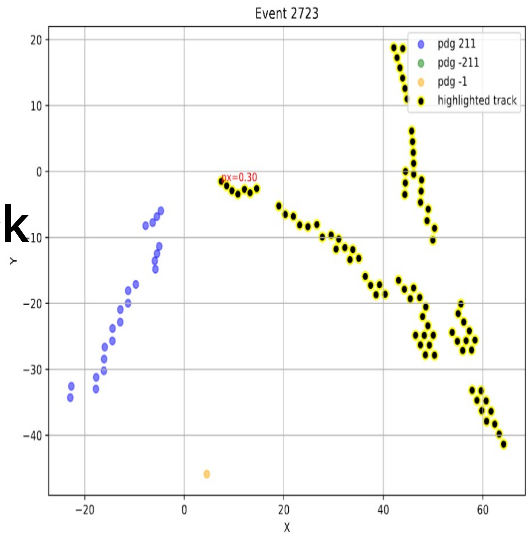



OC clustering results

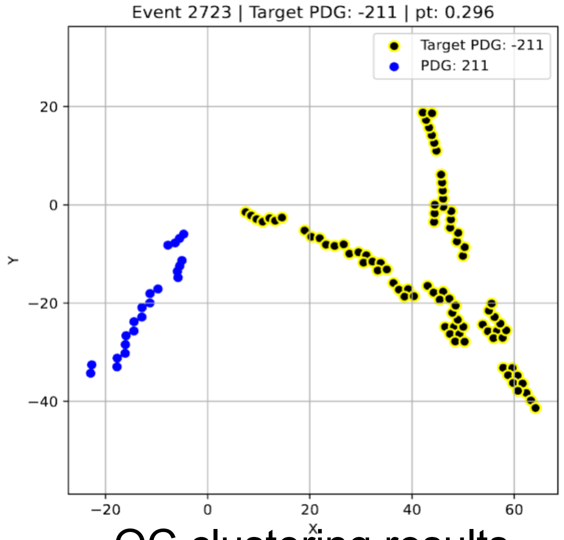



BESIII PATTSF

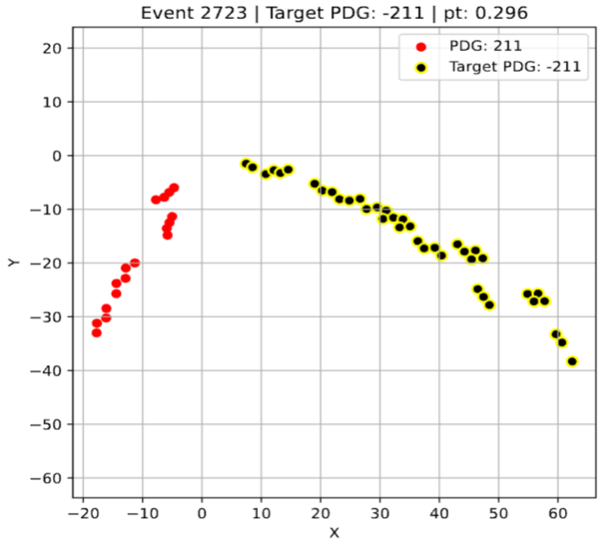
- scattered track




truth




OC clustering results




BESIII PATTSF

# Comparison of GNN and OC

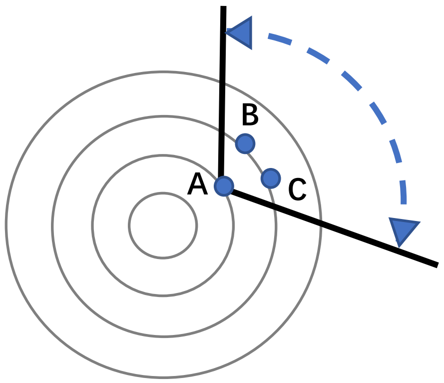
- OC clustering efficiency is higher than GNN
- Attempts to improve fitting rate: keep 1st circle or the part before scattering

| | GNN | OC |
|---|---|---|
| low momentum (pt<200MeV) | circling track event: bad clustering quality | better short track reconstruction ; circling track event: good clustering efficiency but might fail in fitting |
| high momentum | large angle scattering fail in clustering，2%-3% crossing tracks fail in clustering，1% | large angle scattering fail in fitting，3%-4% |

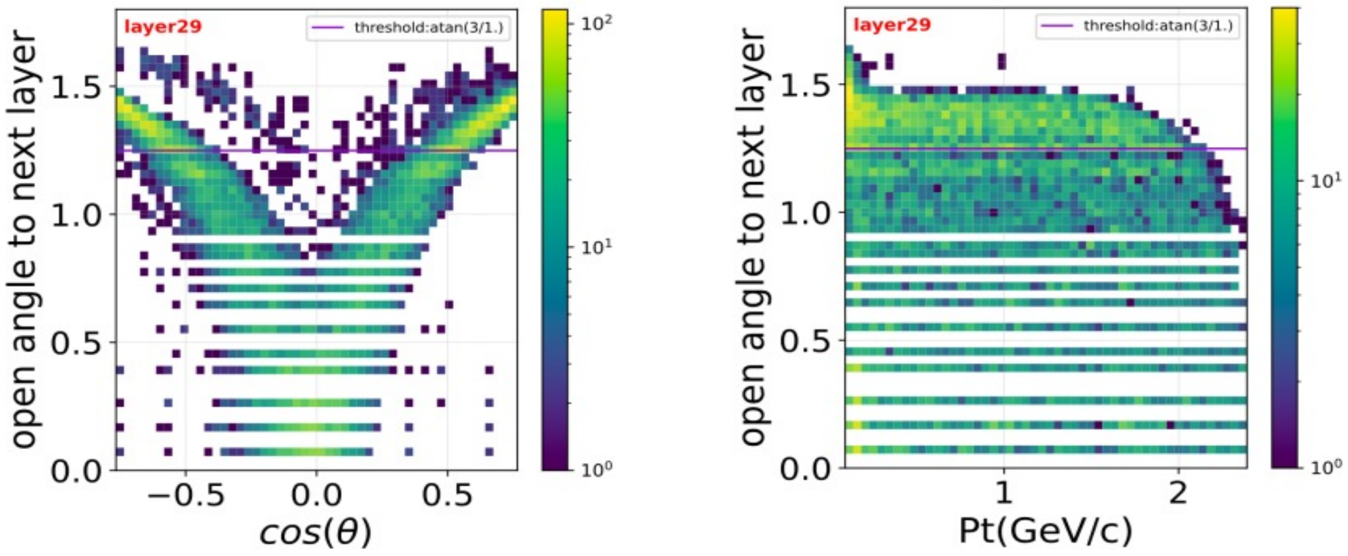
# Graph construction at STCF

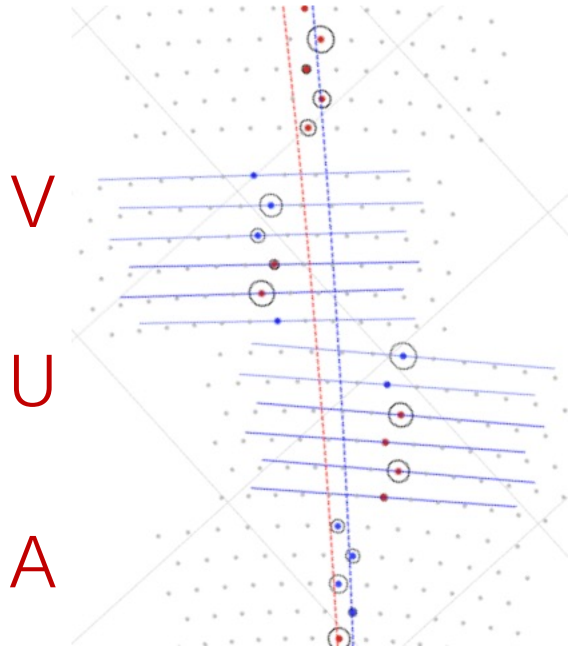
Geometric cut at STCF

- Edge assignment
  - Hit and two adjacent hits on the left and right sides (same layer)
  - Within a certain opening angle (the next layer and one layer apart)
  - The junction of U-V superlayers (layers 11 and 29) appropriately amplify the threshold



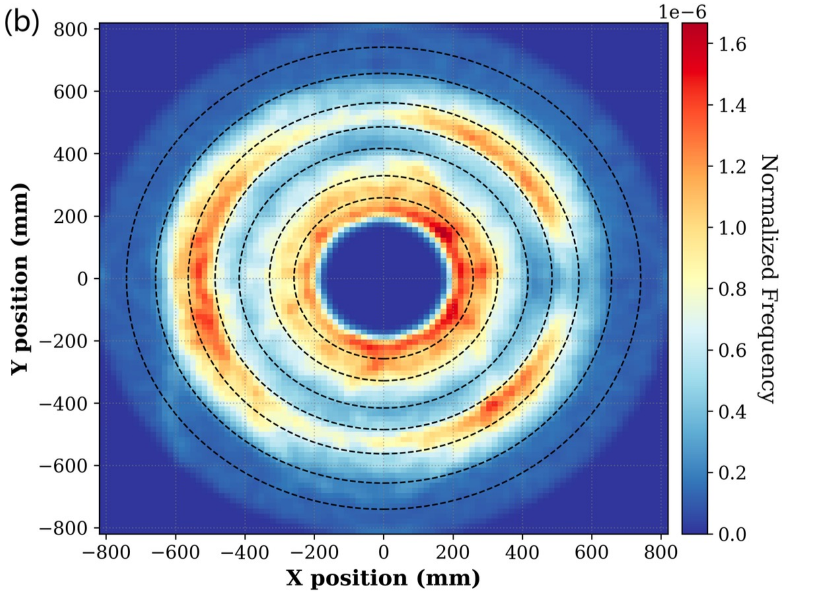

- Graph representation
  - Node features (raw time, position coordinates r, φ of the sense wires), adjacency matrices, edge labels

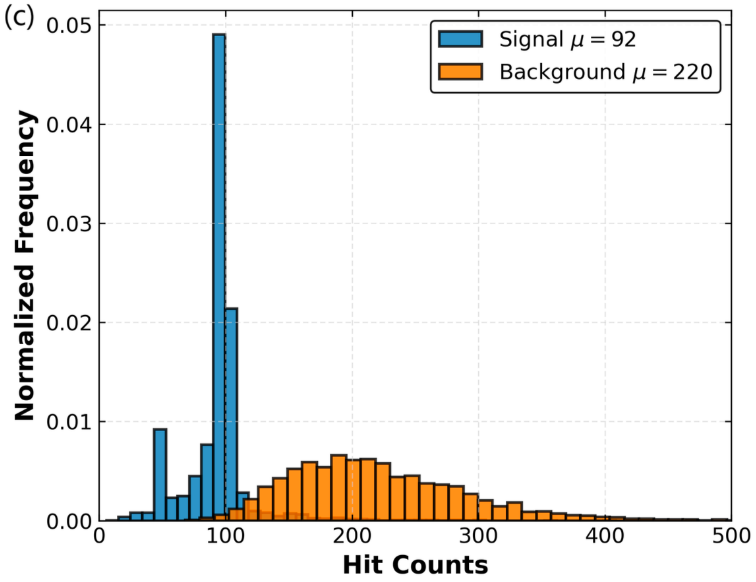


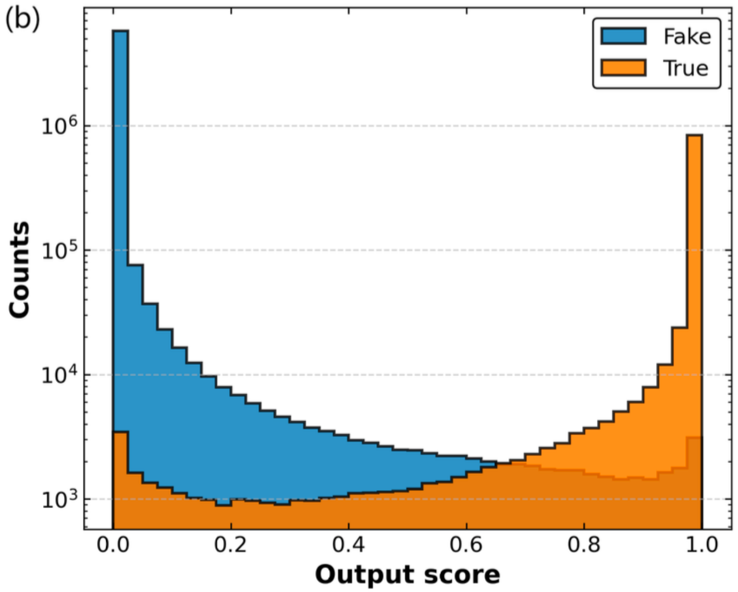



# Performance of noise filter

- PHSP $J/\psi \to \rho^0\pi^0 \to \gamma\gamma\pi^+\pi^-$
- Noise rejection rate of 86.8% while maintaining 98.2% signal selection efficiency



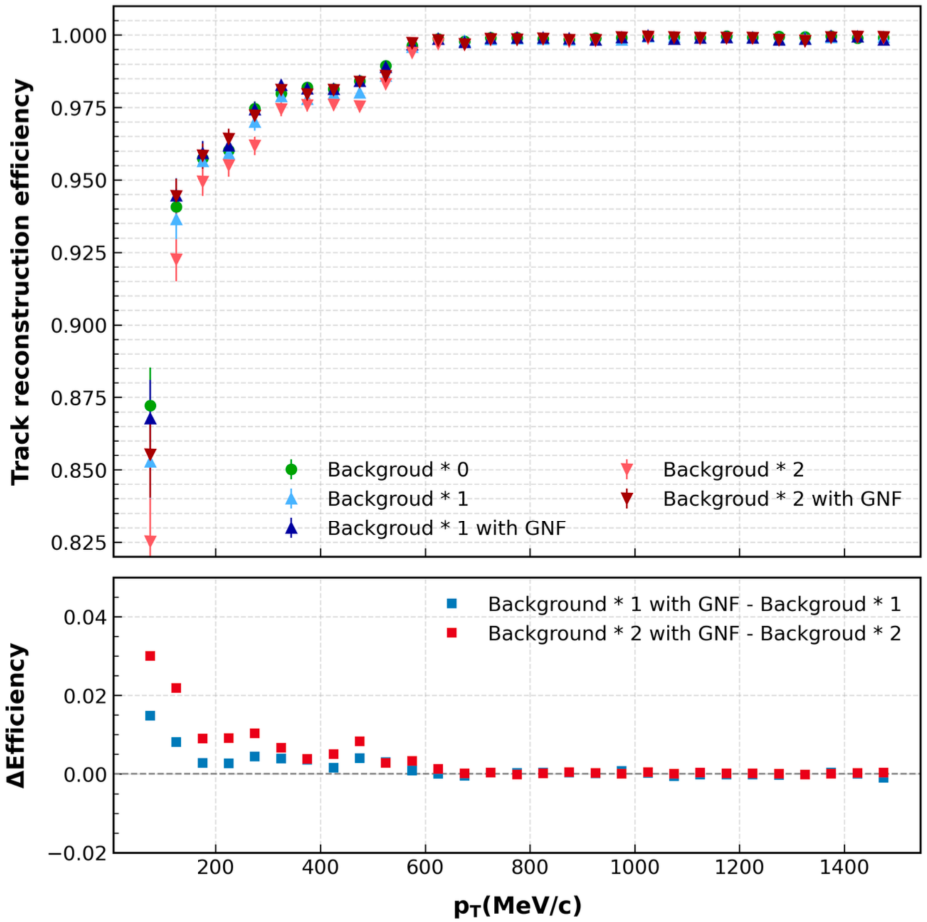



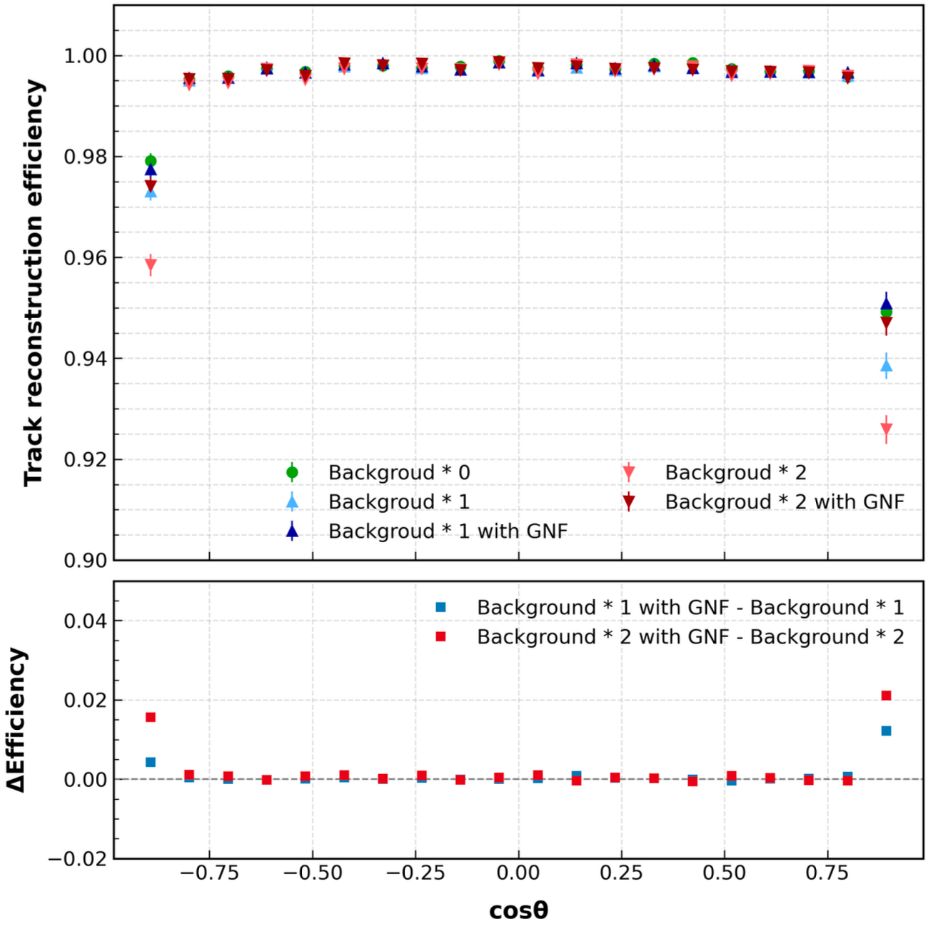




Noise hit distribution

# Tracking performance after noise filter

- Oscar version 2.6.0
- PHSP $J/\psi \rightarrow \rho^0\pi^0 \rightarrow \gamma\gamma\pi^+\pi^-$
- Improve tracking efficiency with GNF, especially for low momentum region



# Summary

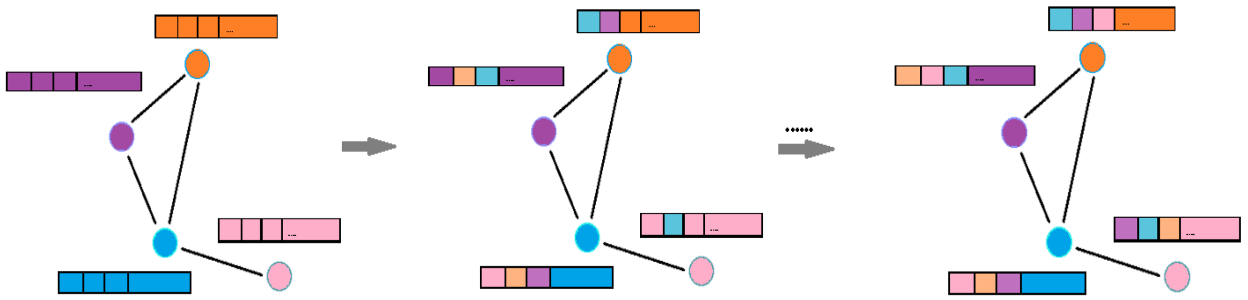
- A novel tracking algorithm prototype based on machine learning method for BESIII/STCF is under development
  - GNN to distinguish the hit-on-track from noise hits.
  - Clustering method based on DBSCAN and RANSAC to cluster hits from multiple tracks
- Preliminary results on MC data shows promising performance

- Outlook
  - Further optimization: circular, scattering..
  - Performance verification concerning events with more tracks and long lived particle
  - Check the reconstruction time consumption

Thank you!

# Graph Neural Network

- GNN key idea: propagate information across the graph using a set of learnable functions that operate on node and edge features



*Message-passing*

- Graph Neural Network edge classifier
  - High classification score
    - the edge belongs to a true particle track
  - Low classification score
    - it is a spurious or noise edge