中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

Progress in CEPC Drift Chamber Software

Mengyao Liu[1], Tao Lin[2], Weidong Li[2], Wenxing Fang[2], Xingtao Huang[1], Xueyao Zhang[1], Yuan Ye[2], Yao Zhang[2]

1.Shandong University

2.Institute of High Energy Physics

CEPC workshop at Edinburgh

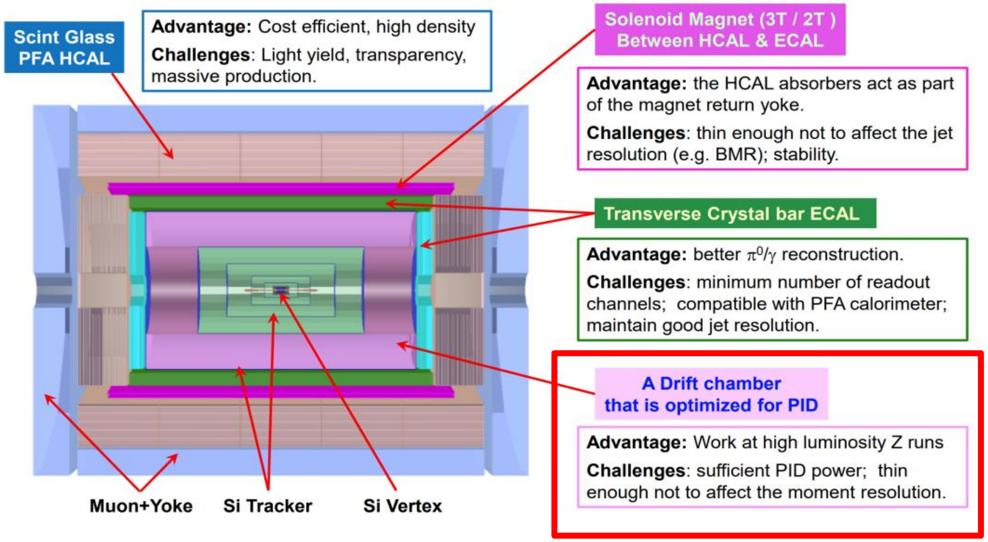
05-July-2023

Contents

- Introduction
- Detector simulation
- Track reconstruction
- Summary

Detector

- The CEPC experiment mainly aims to precisely measure the property of the Higgs boson.
- Physics requirements: high track efficiency (~100%), momentum resolution (<0.1%), PID (2σ p/K separation at P < ~ 20 GeV/c), etc.

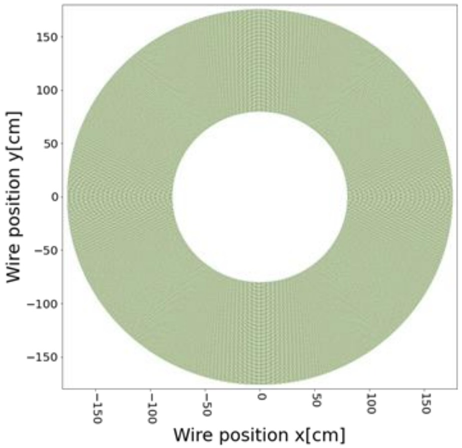


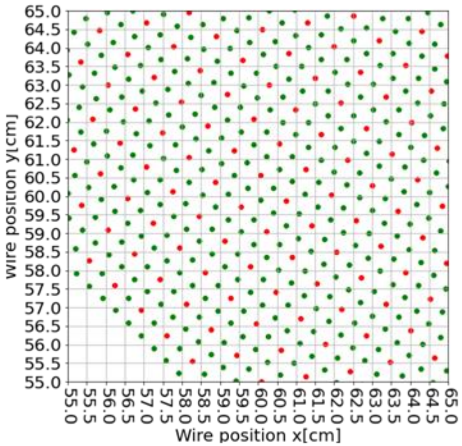
- For the 4th conceptual detector, silicon detector and drift chamber (DC) are designed to provide both tracking and PID for charged particles.
- Both detector design and physics potential studies needs strong support of simulation and reconstruction software.

Drift Chamber

- The baseline configuration of DC in CEPCSW

Half length	2980 *mm*
Inner and outer radius	800mm to 1800 *mm*
# of Layers	100/55
Cell size	~10mmx10mm/18mmx18mm
Gas	He:iC_4H_{10}=90:10
Single cell resolution	0.11 *mm*
Sense to field wire ratio	1:3
Total # of sense wire	81631/24931
Stereo angle	1.64~3.64 *deg*
Sense wire	Gold plated Tungsten ϕ=0.02*mm*
Field wire	Silver plated Aluminum ϕ=0.04*mm*
Walls	Carbon fiber 0.2 *mm*(inner) and 2.8 *mm*(outer)

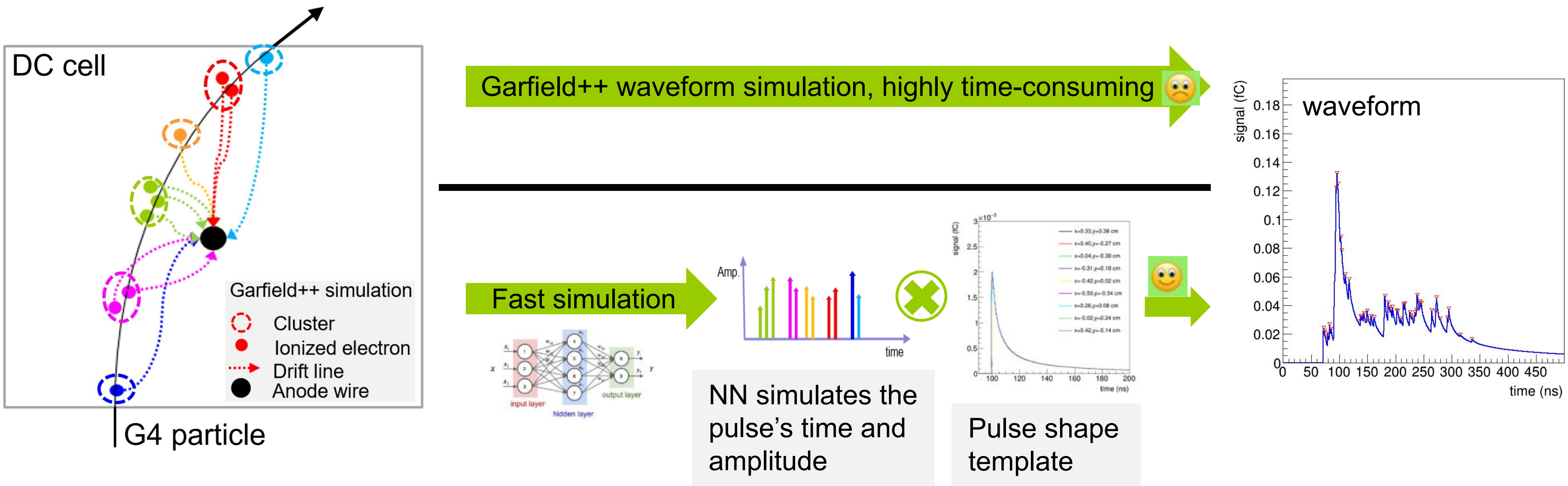




Cell structure

- Introduction
- Detector simulation
- Track reconstruction
- Summary

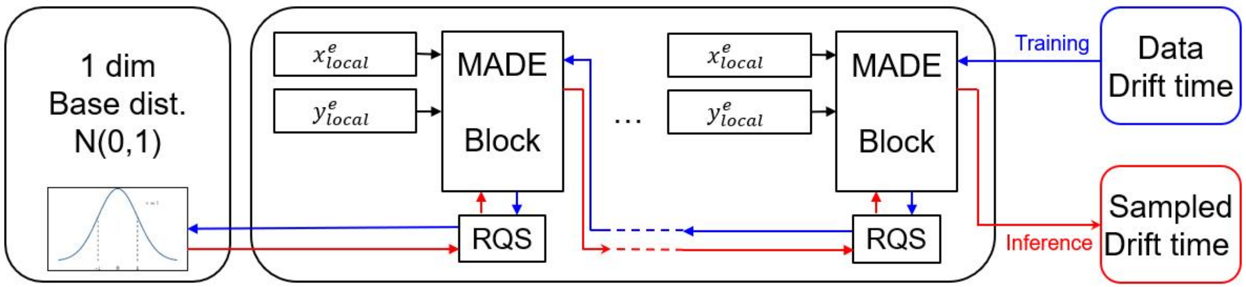
Simulation of Gaseous Detector



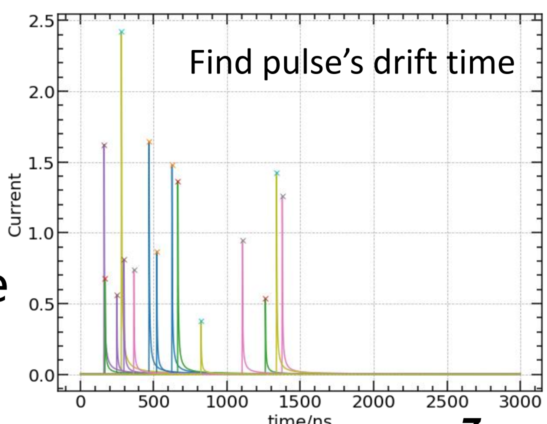
- TrackHeedSimTool (Gaudi tool) was implemented by combining Geant4 and Garfield++ to simulate the complete response of the gaseous detector
  - Input: G4Step information (particle type, initial position, momenta, and step length)
  - Using TrackHeed(from Garfield++) to create the ionization electron-ion pairs (for both primary and secondary ionizations), the deposited energy will be used to update the energy of the G4Particle
  - Using NN to simulate the time and amplitude of each pulse for each ionized electron (for fast waveform simulation)
  - Output: primary, total ionization, and pulse information, saved in EDM

ML-based Simulation Method (1)

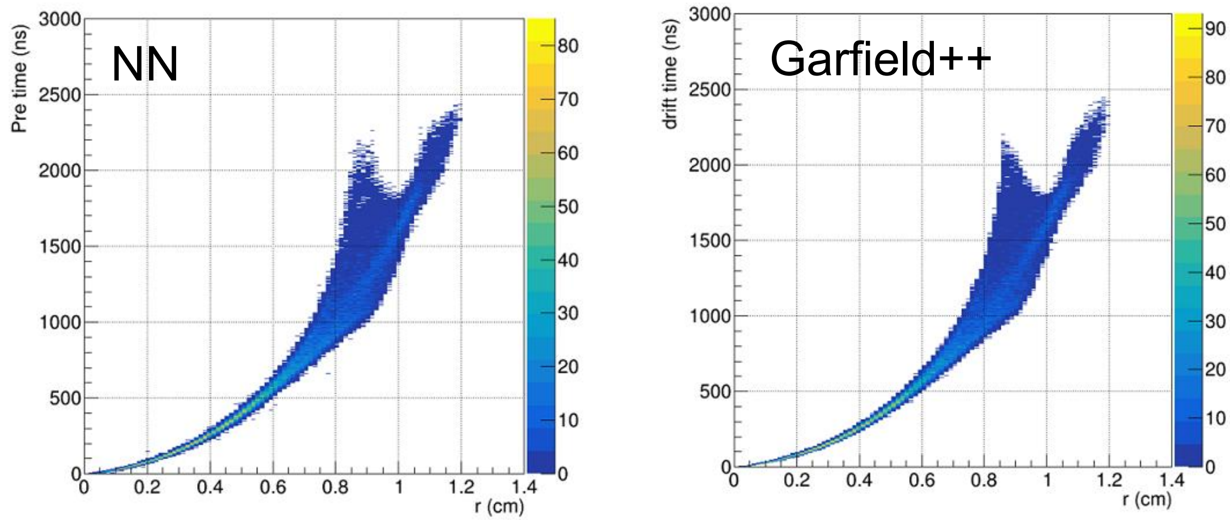
- We studied learning the drift time distribution from Garfield++ to achieve precise drift time simulation
- Normalizing Flow network was adopted
  - A similar model to CaloFlow is used, RQS (for transformation)+MADE block (for learning the parameters of RQS)

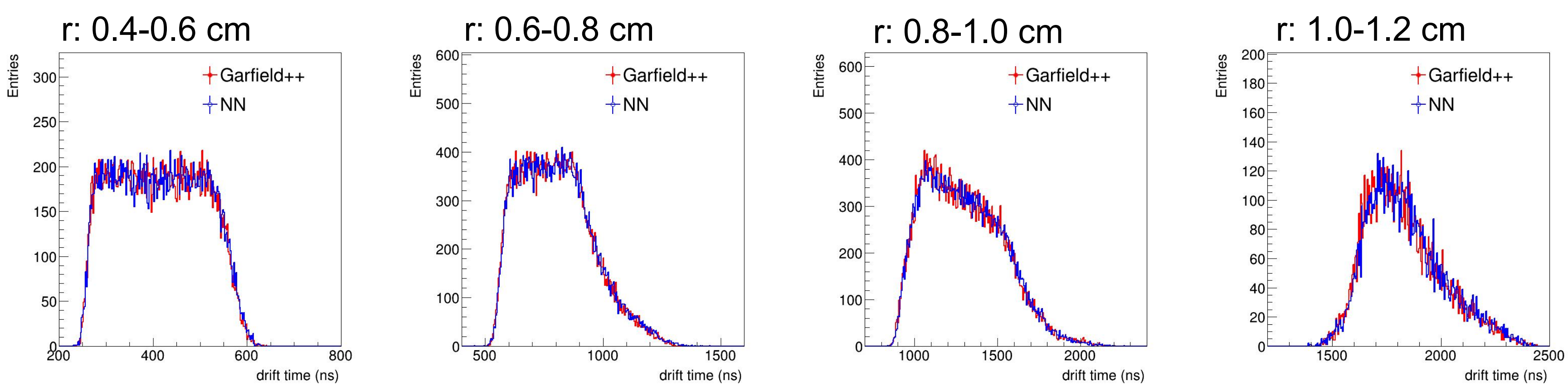


- Training data is from Garfield++ simulation:
  - Gas: 90%$\mathrm{H_e}$+10%$\mathrm{C_4H_{10}}$
  - For each event, an ionized electron is uniformly generated in the DC cell (x_{local}, y_{local}) and the pulse is simulated. Then a peak finding algorithm (scipy.signal.find_peaks()) is used to get drift time value



ML-based Simulation Method (2)





- Good agreement between the NN and Garfield++ simulation

Method Validation with BESIII Data (1)

- To investigate the possibility of applying ML to simulation, the real data from the BESIII experiment was used evaluate the performance of the chosen neural network.
- Radiative bhabha events were selected to study the simulation of drift time in the chamber cell
  - X-T relation:
    - doca(distance of closest approach) v.s. drift time

Prepare experimental data for training

↓

Learn drift time distribution: $D_{drift\ time} = f(doca, eangle, layer)$

↓

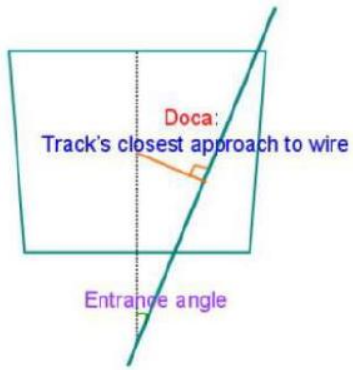
Check $D^{data}_{drift\ time}$, $D^{NN\ sim.}_{drift\ time}$

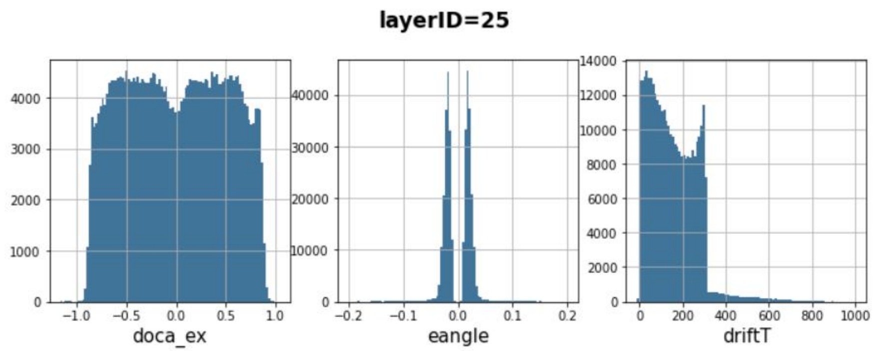
↓

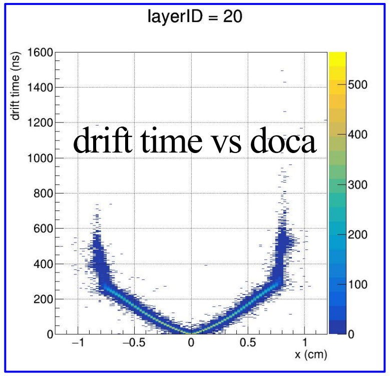
check the X-T cure: C^{data}_{X-T} , $C^{NN\ sim.}_{X-T}$

↓

check spatial resolution: using C^{data}_{X-T} , $C^{NN\ sim.}_{X-T}$

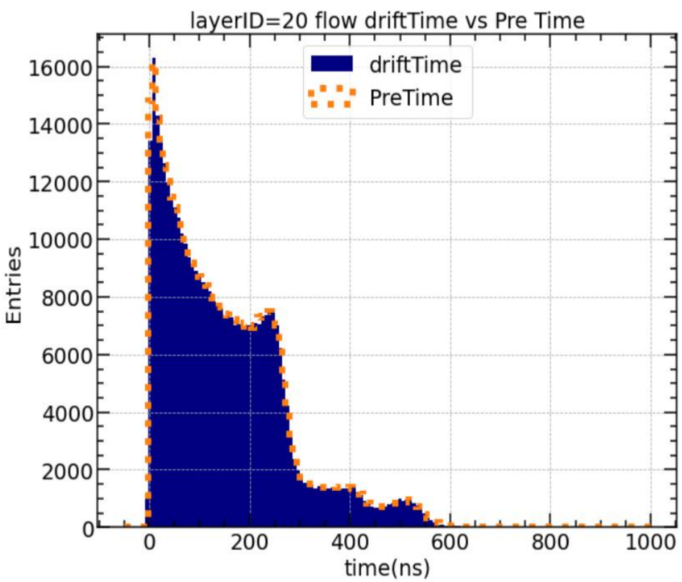


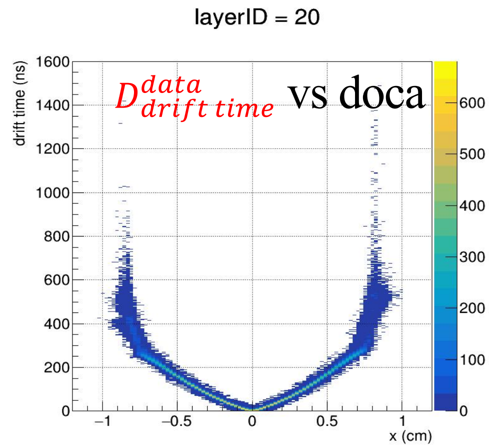


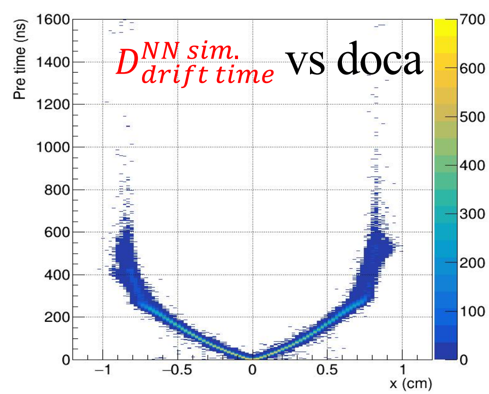


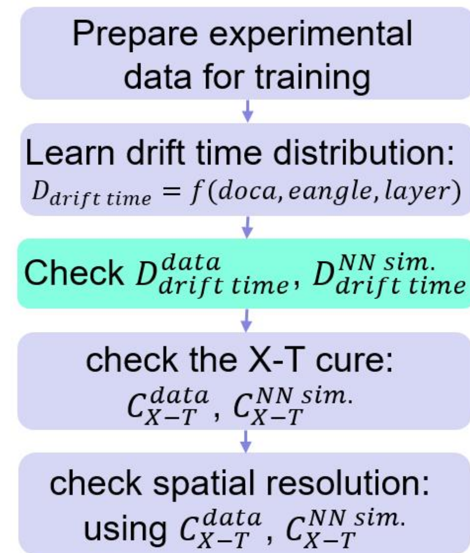
Method Validation with BESIII Data (2)

- Comparison of drift time distributions between real data and ML-based simulation





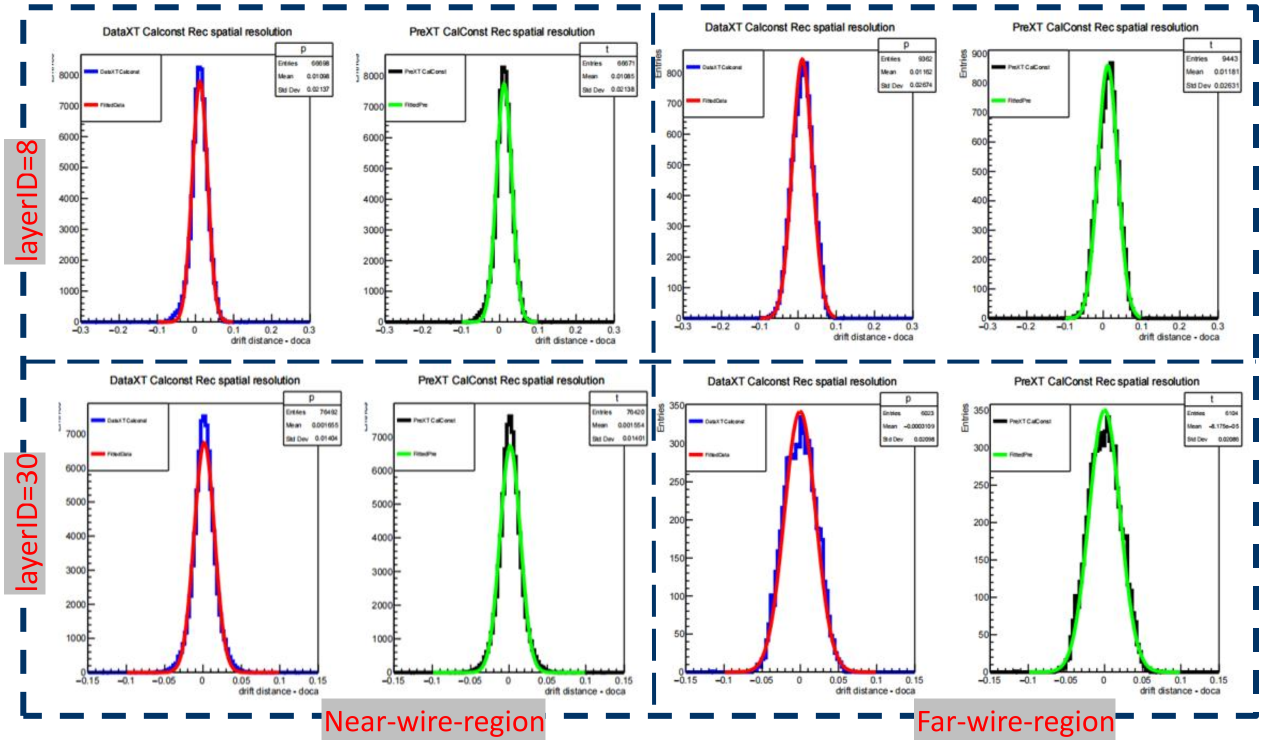




- Good agreement was found

Method Validation with BESIII Data (3)

- Checked the spatial resolution of reconstructed tracks, using C_{X-T}^{data} and $C_{X-T}^{NN\ sim.}$ (calibration constants)



layerID	Near-wire-region (C_{X-T}^{data} v.s. $C_{X-T}^{NN\ sim.}$)		Far-wire-region (C_{X-T}^{data} v.s. $C_{X-T}^{NN\ sim.}$)	
	mean	sigma	mean	sigma
8	0.01096/0.01085	0.02137/0.02138	0.01162/0.01181	0.02674/0.02631
30	0.001055/0.001564	0.01404/0.01401	-3.1e-4/-5.1e-5	0.02060/0.02086

Prepare experimental data for training

↓

Learn drift time distribution: $D_{drift\ time} = f(doca, eangle, layer)$

↓

Check $D_{drift\ time}^{data}$, $D_{drift\ time}^{NN\ sim.}$

↓

check the X-T cure: C_{X-T}^{data} , $C_{X-T}^{NN\ sim.}$

↓

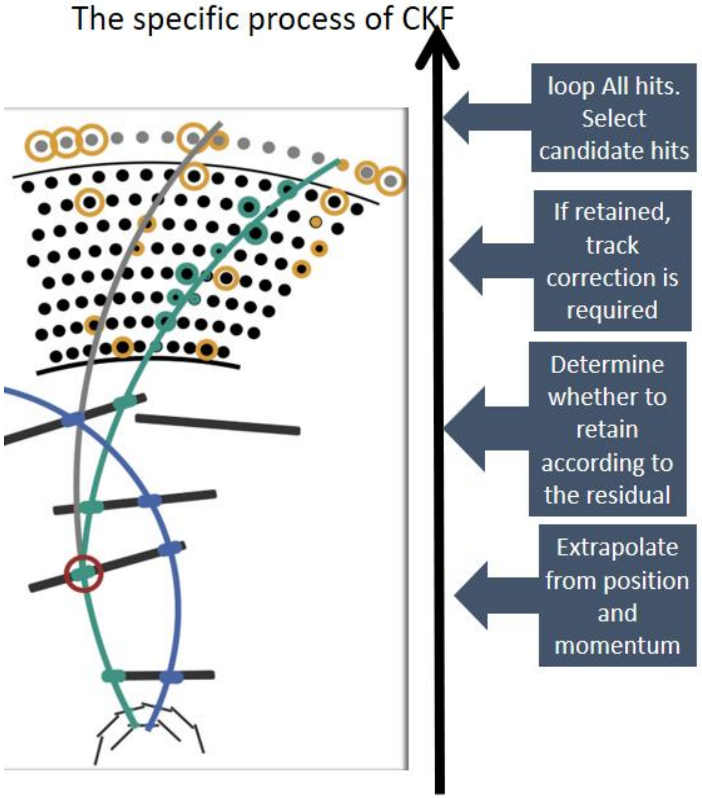
check spatial resolution: using C_{X-T}^{data} , $C_{X-T}^{NN\ sim.}$

- Consistent spatial resolution results obtained for C_{X-T}^{data} and $C_{X-T}^{NN\ sim.}$
- The NN can learn the drift time from the data

- Introduction
- Detector simulation
- Track reconstruction
- Summary

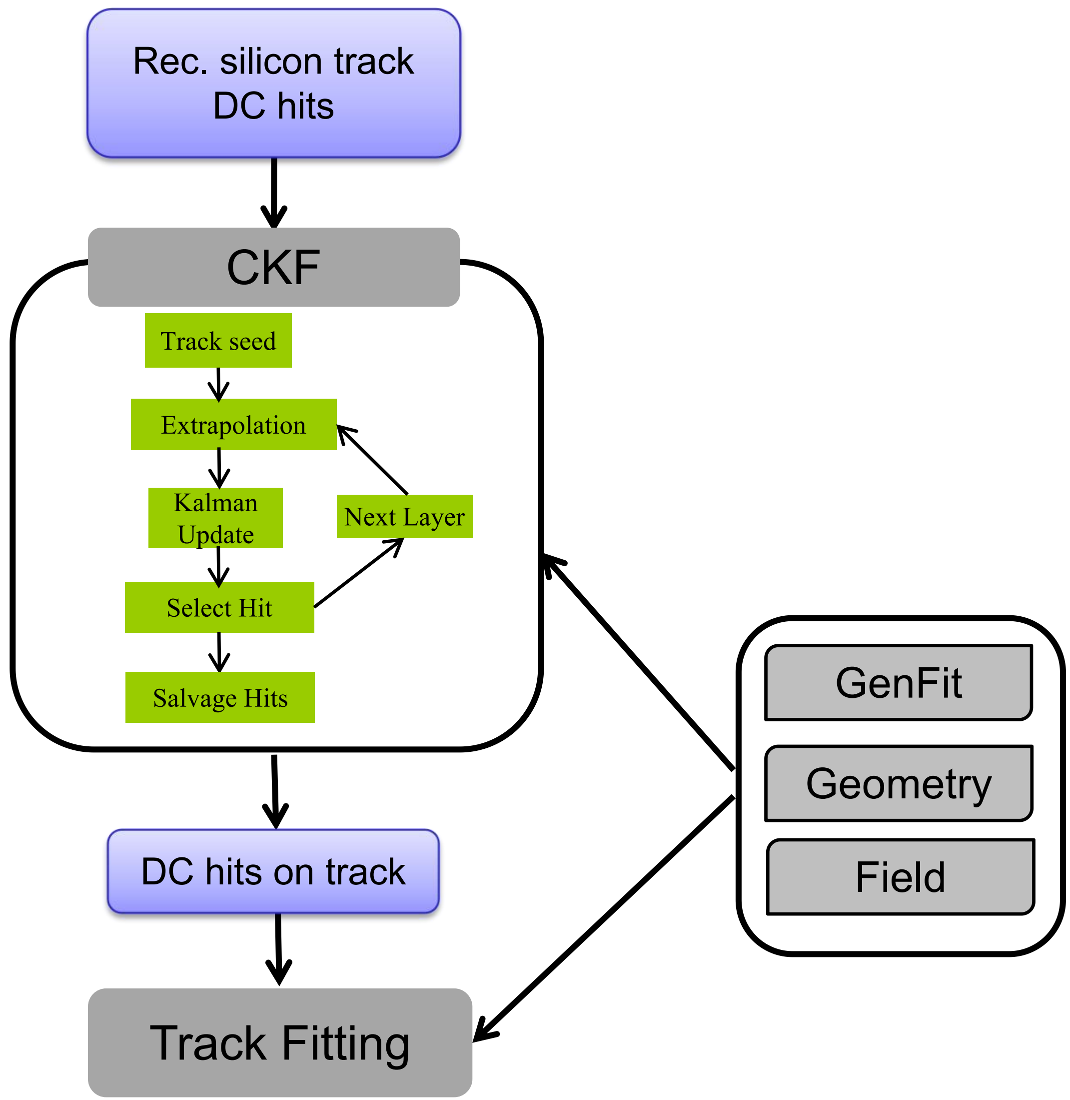
Track Reconstruction

- Tracking with Combinatorial Kalman Filter (CKF) method
  - Used by many high energy physics experiments
- Track finding with CKF in drift chamber
  - Migrate from Belle2
  - Track segments reconstructed in the silicon detector, called seeds, are extrapolated to the DC and all the DC hits belonging to the track are collected
- Track fitting tool: Genfit
  https://github.com/GenFit/GenFit/
  - Experiment-independent generic track fitting toolkit
  - Official track fitting for BelleII, also used by PANDA, COMET, GEM-TPC etc.
  - Using DAF kalman filter



Track Reconstruction

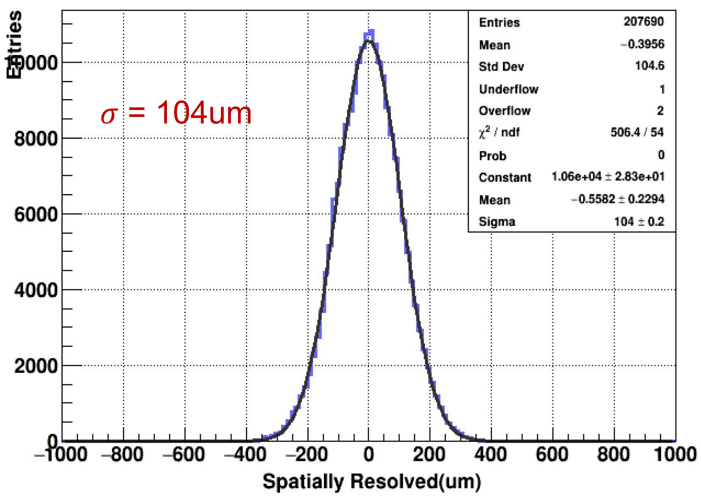
- Implementation
  - Extrapolation based on GenFit
  - Field, material and geometry from DD4hep
  - A data format converter between TrackerHit(from EDM4hep) and CDCWireHit
- Status
  - Performance validation completed



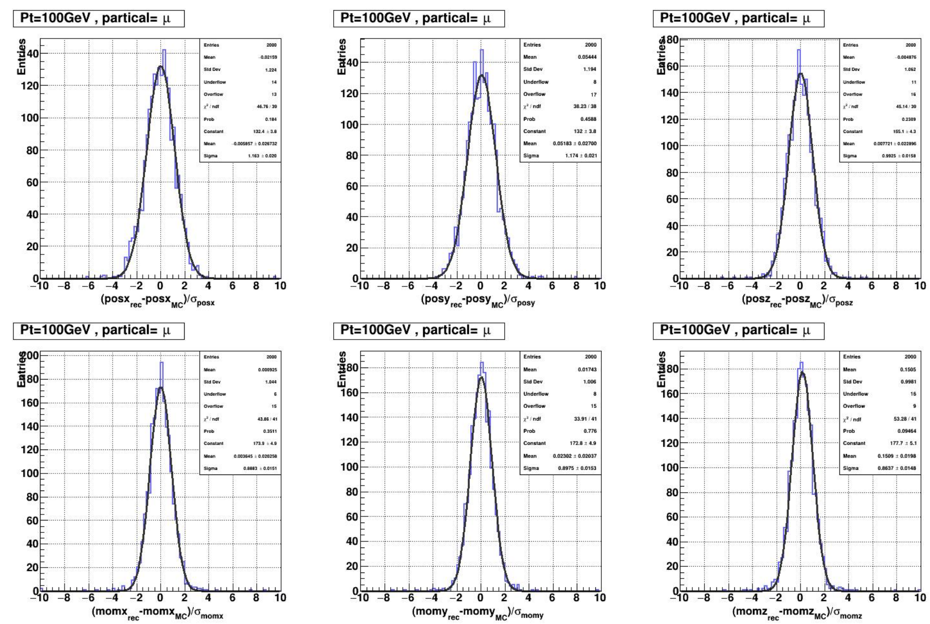
Workflow of track reconstruction

Quality of track fitting

- Data Sample: Single $\mu^-, \theta = 50°, p_T = 10GeV/c$ with single cell resolution of $110\mu m$
- Track pull distribution
  - posx , posy , posz , momx , momy , momz follows N~(0,1)
- Spatial resolution consistent with the simulation



Spatial Resolution

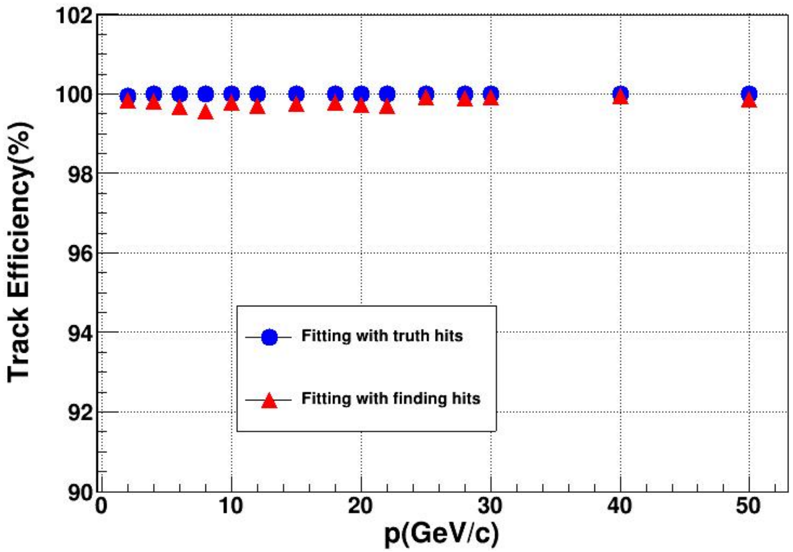


Normalized parameter residual distributions

- The estimation of the track parameter and error is reliable

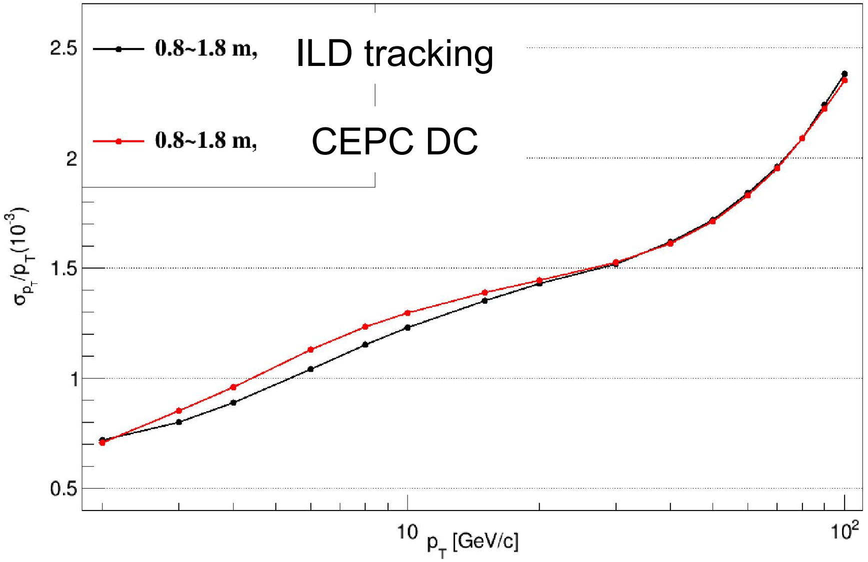
Tracking Efficiency

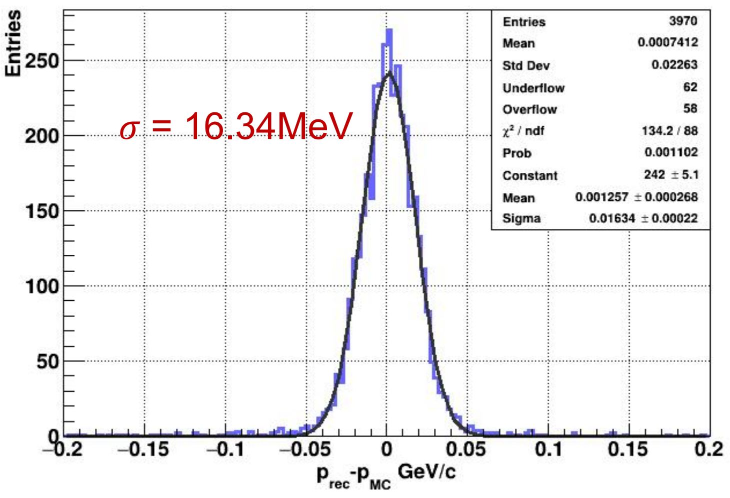
- Data sample: Single particle $\mu^{-}, \theta = 50^{\circ}$
- Track Efficiency = N_1/N_2
  - N_1 is the number of track satisfying:
    - $chi^2 < 400$
    - $N_{DC\ hits\ on\ track} > 50$
  - N_2 is the numbre of silicon track
- The efficiency is over 99% and closely aligns with the results using truth hits



Momentum Resolution

- Data Sample: Single particle $\mu^{-}, \theta = 85°$
- Combined measurement of Silicon and Drift Chamber
- Momentum resolution is reasonable and consistent with ILD tracking

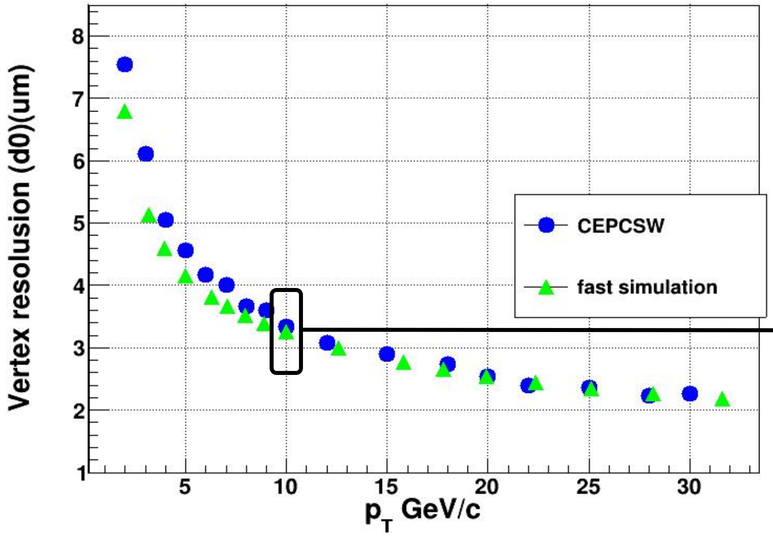


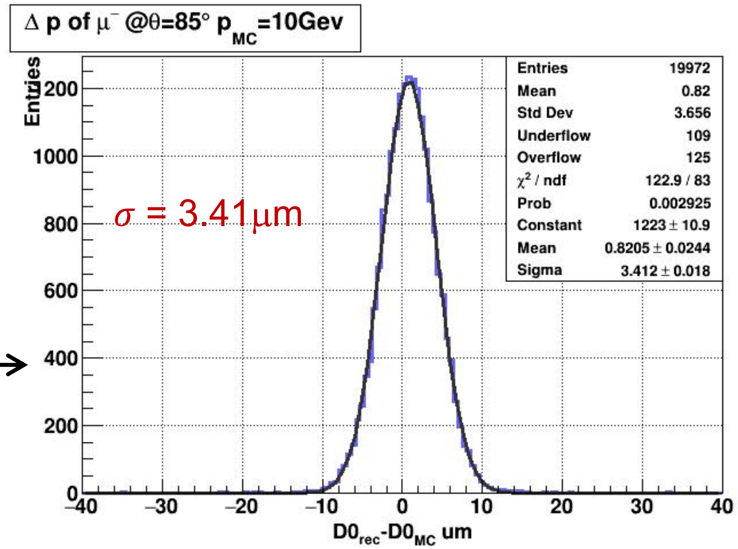


Momentum distribution of 10GeV/c μ^{-}

Impact Parameter

- Data Sample: Single particle $\mu^-, \theta = 85°$
- Impact parameter
  - $\sigma_{d0} = 3.41\mu m\ with\ p_T = 10GeV/c$
  - Consistent with fast simulation

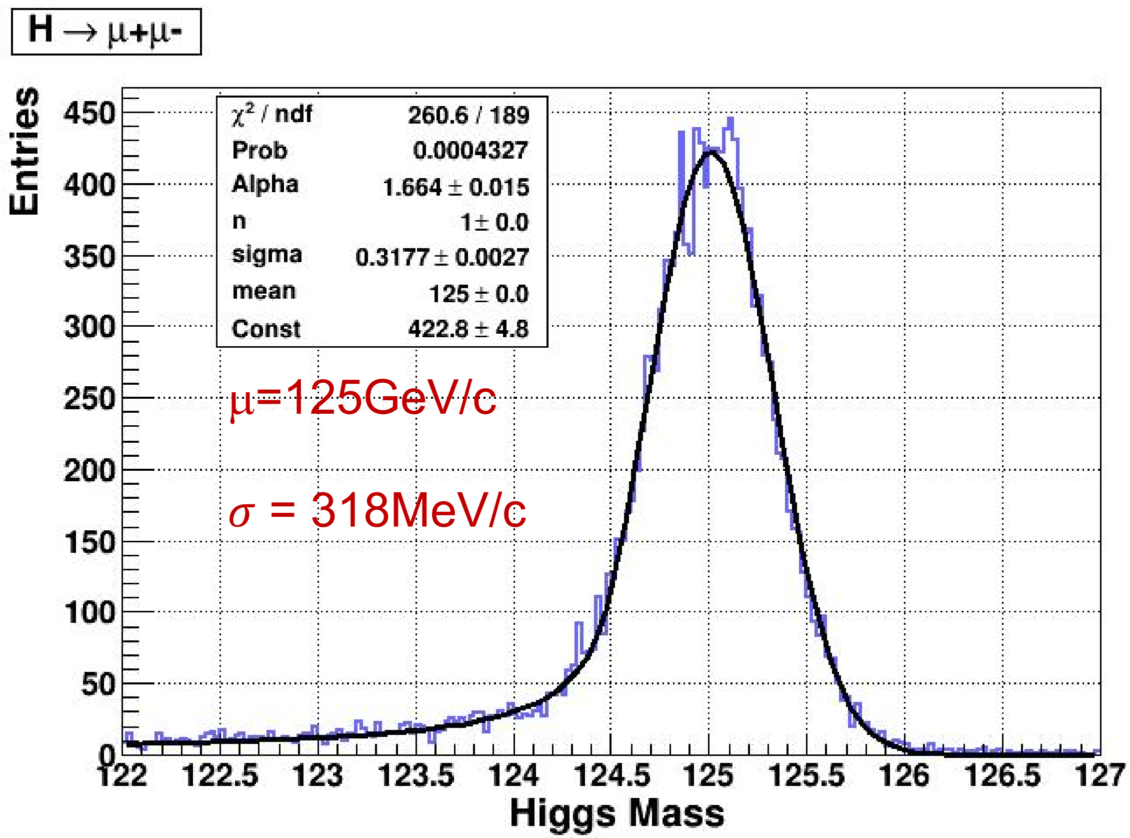




Momentum distribution of 10GeV/c μ^-

Physics Event Reconstruction

- Higgs reconstruction for $\mathrm{H} \rightarrow \mu^{+}\mu^{-}$
- Can be used for physics simulation studies



Summary

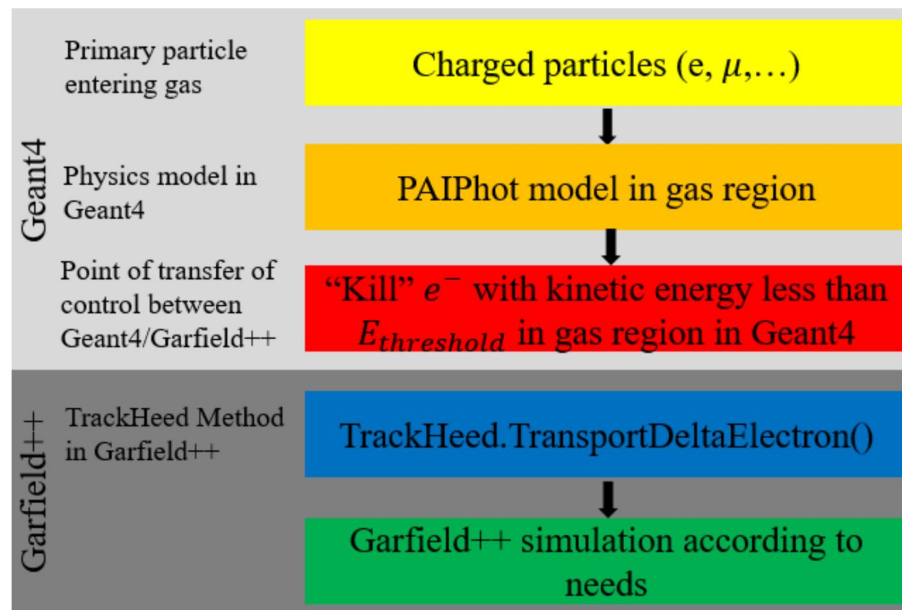
- In CEPCSW, TrackHeedSimTool was implemented by combining Geant4 and Garfield++ to simulate the complete response of the gaseous detector
  - Machine learning based algorithm was developed for waveform generation
- Tracking algorithm was implemented by reusing the code of Belle II and its performance meets expectations.

Thank You！

Back up

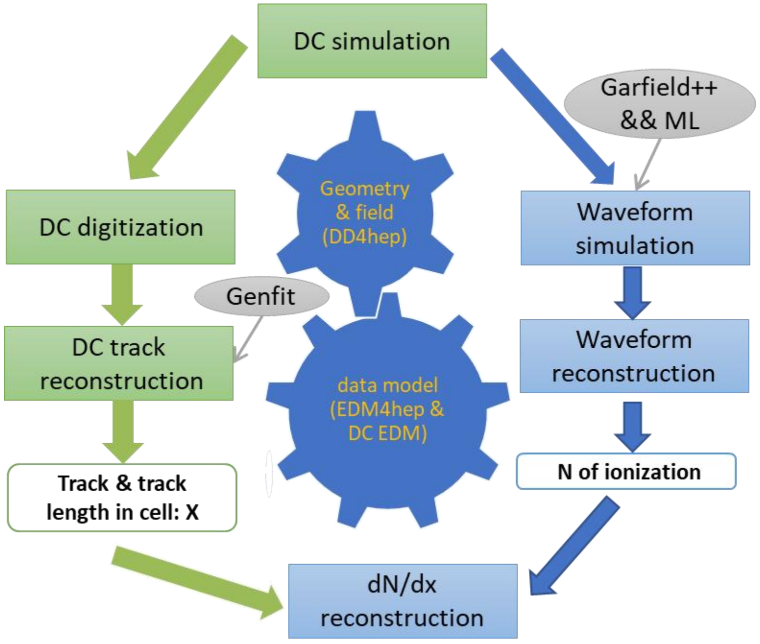
Simulation of Gaseous Detector (1)

- Since Geant4 can not be used to simulate the ionization process properly (arXiv:2105.07064), Garfield++ becomes a common tool for precise ionization simulation.
- "Interfacing Geant4, Garfield++ and Degrad for the Simulation of Gaseous Detectors" studied how to combine Geant4 and Garfield++ to get correct energy deposition or total number of ionized electrons (adopted by COMET experiment)
- Method:
  - Geant4 PAI (Photo Absorption Ionization) model to simulate primary or secondary ionization
  - TrackHeed (from Garfield++) to simulate ionization from residual delta electron



Work Flow for Simulation and Reconstruction

- Detector simulation
  - Geant4 is employed to simulate particle’ s propagation (including particle decay) in the detector, interaction with detector material, etc.
  - TrackerHeed (from Garfield++) is used to simulate ionization process of charged particles (e, μ, π, K, p, …) when they pass through the drift chamber.
  - Garfield++ was integrated with the CEPCSW to simulate but its extreme computation intensiveness makes it impossible



Drift chamber simulation and reconstruction flow

  - Machine learning (ML) based simulation: training data is created by Garfield++ and ML model is be executed to replace Garfield++ in the detector simulation.
- Reconstruction
  - Extrapolating the track segment found in the inner silicon detector to drift chamber, collecting the hits on the path, and applying a Kalman Fit to the found track .
  - dN/dx reconstruction: waveform reconstruction + path length calculation