Silicon Tracking at CEPC

FU Chengdong
IHEP, CAS

The 2023 international workshop on the High Energy Circular Electron-Positron Collider
October 27, 2023

Outline

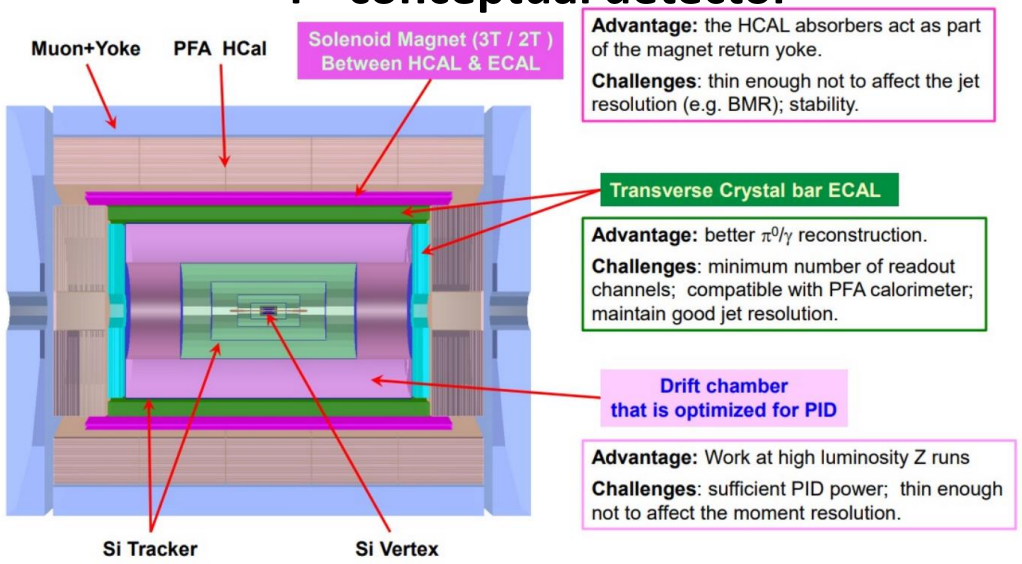
- Introduction
- Tracking software
- Application
  - Performance estimation for silicon tracker
  - Simulation test for beam test of vertex detector
- Summary

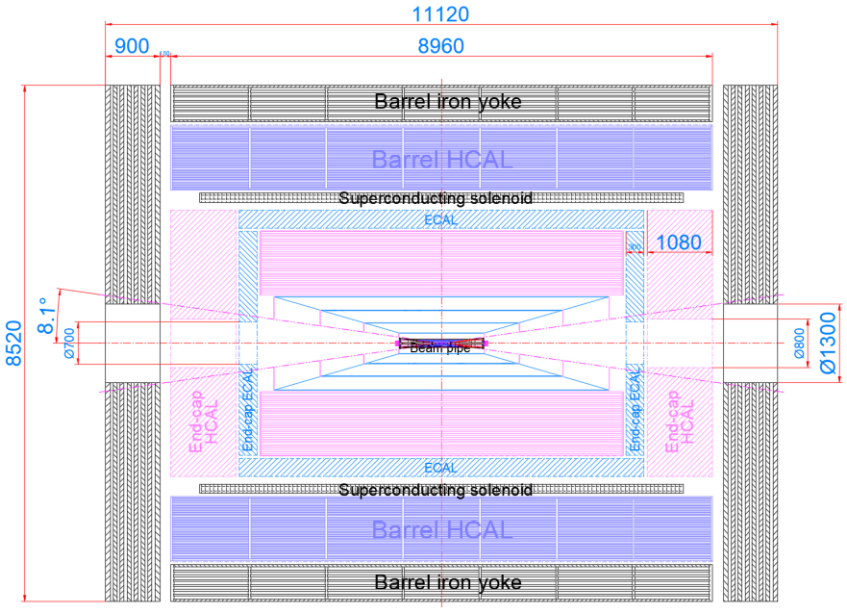
Introduction

Physics process	Measurands	Requirement on tracker
ZH, $Z\to e^+e^-(\mu^+\mu^-)$, $H\to\mu^+\mu^-$	m_H, $\sigma(ZH)$, BR($H\to\mu^+\mu^-$)	$\Delta(1/p_T) = 2\times10^{-5} \oplus \frac{0.001}{p(GeV)\sin^{3/2}\theta}$

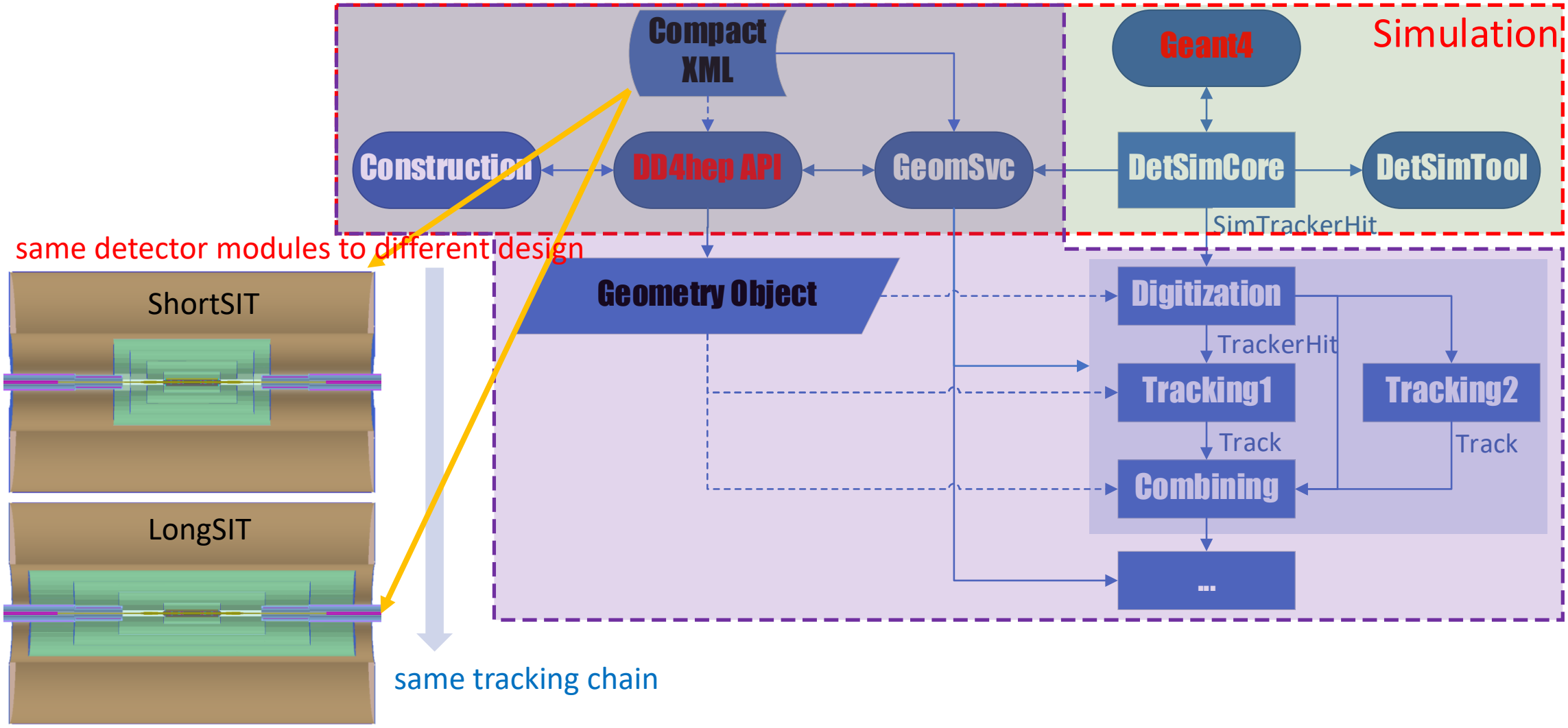
- CEPC being designed as Higgs&Z factory, has basic physics requirements, such as tracking resolution.
  - Good tracker design
  - Good tracking software
- Three detector concepts were designed at CDR stage, and the 4th conceptual detector design has been proposed since 2021, continuing to be optimized.
  - The silicon detectors are designed as part of all detector concept!
- From CDR to TDR, the software platform is being switched from cepcsoft (Mokka&Marlin) to CEPCSW (DD4hep&Gaudi) step by step. The simulation and reconstruction for the silicon tracker have been completely implemented in CEPCSW.
  - Exactly as one of tracker estimation tools
  - Developing to improve the reliability
- From last workshop (validation on tracking performance of single particle and efficiency of $b\bar{b}H$, $\tau\to$3prong), more optimizations are ongoing to support more application, such as
  - Validation on the new endcap silicon tracker design
  - Analysis of beam test of the MOST2 vertex detector
- Test feasibility and developing... For software, performing beam test also can meet some problem on real data.

4th conceptual detector



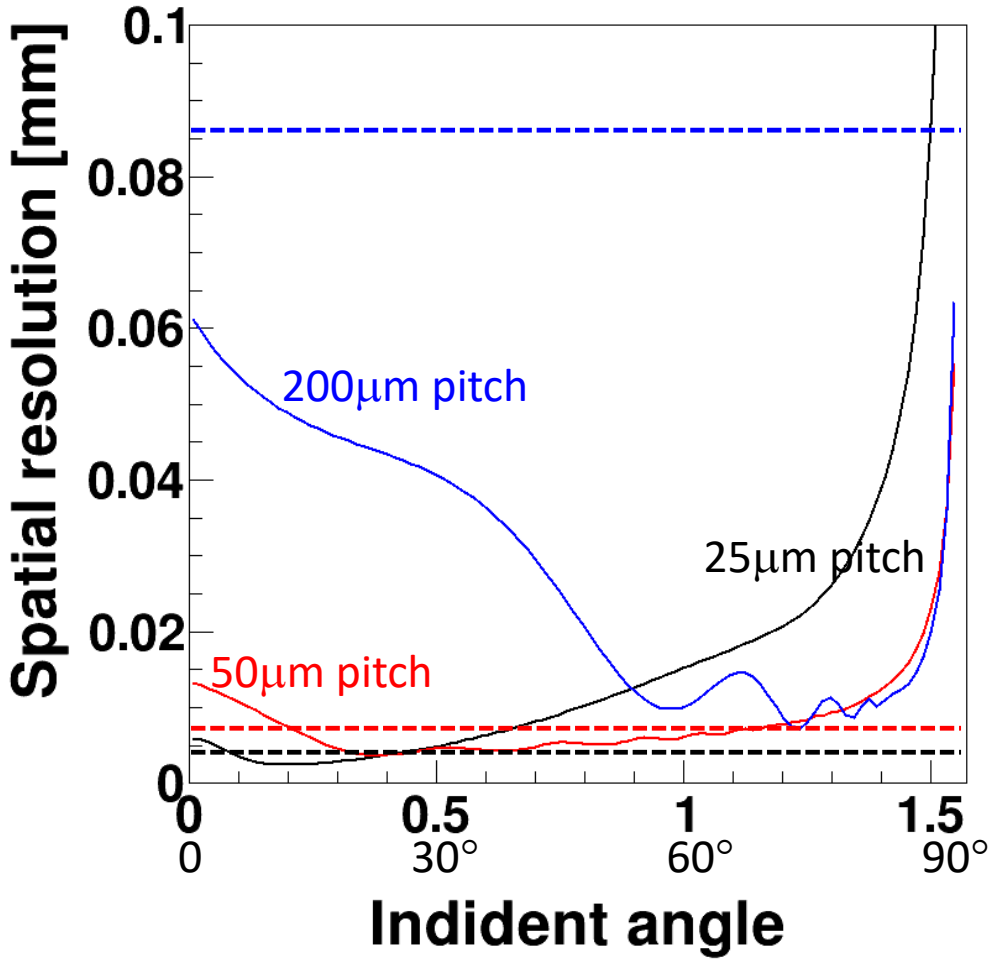


Tracking in CEPCSW

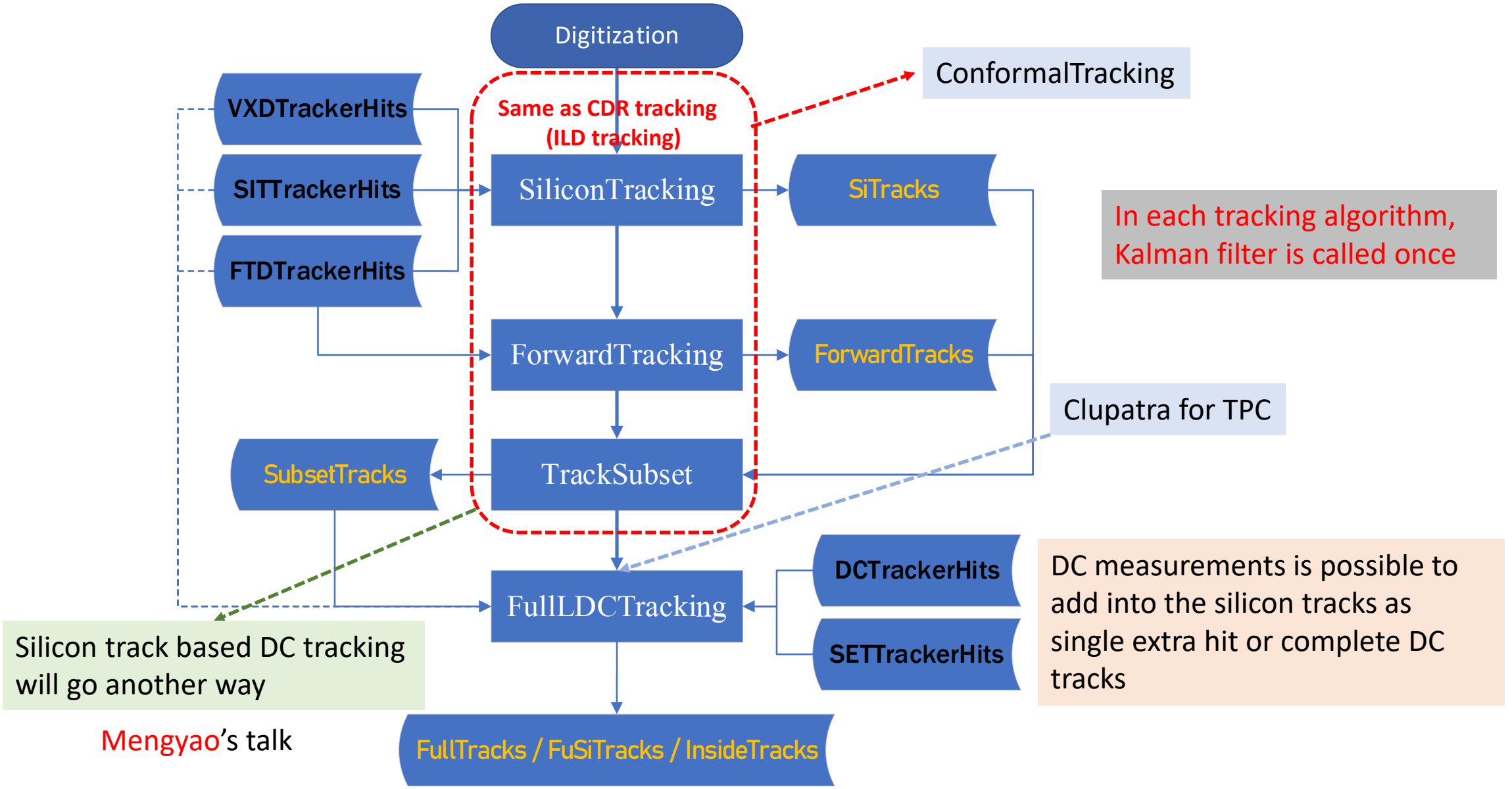


Digitization

- Gaussian smearing on SimTrackerHit at measurement dimension (u,v)
  - pixel: 2D (u,v)
  - strip: 1D (u,0) or (0, v)
- Fixed spatial resolution
  - VXD
    - ✓ $\sigma_{rphi,z}$=4μm(2.8μm), 4μm(6μm), 4μm, 4μm, 4μm, 4μm
  - SIT
    - ✓ σ_{rphi} =7.2μm, σ_z=86μm
  - SOT/SET
    - ✓ σ_{rphi} =7.2μm, σ_z=86μm
  - Endcap tracker
    - ✓ σ_{rphi} =7.2μm, σ_z=86μm
- Parameterized spatial resolution
  - Riccardo del Burgo's parametrization model
    - ✓ $\sigma_{u,v} = p_0 + p_1 x + p_2 e^{-p_9 x}\cos(p_3 x + p_4) + p_5 e^{-\frac{(x-p_6)^2}{2p_7^2}} + p_8\sqrt{x}$
  - parameters are relative to pitch size (for CMS PhaseII)
    - ✓ VXD: 25μm×25μm
    - ✓ others: 50μm×200μm



Tracking Chain

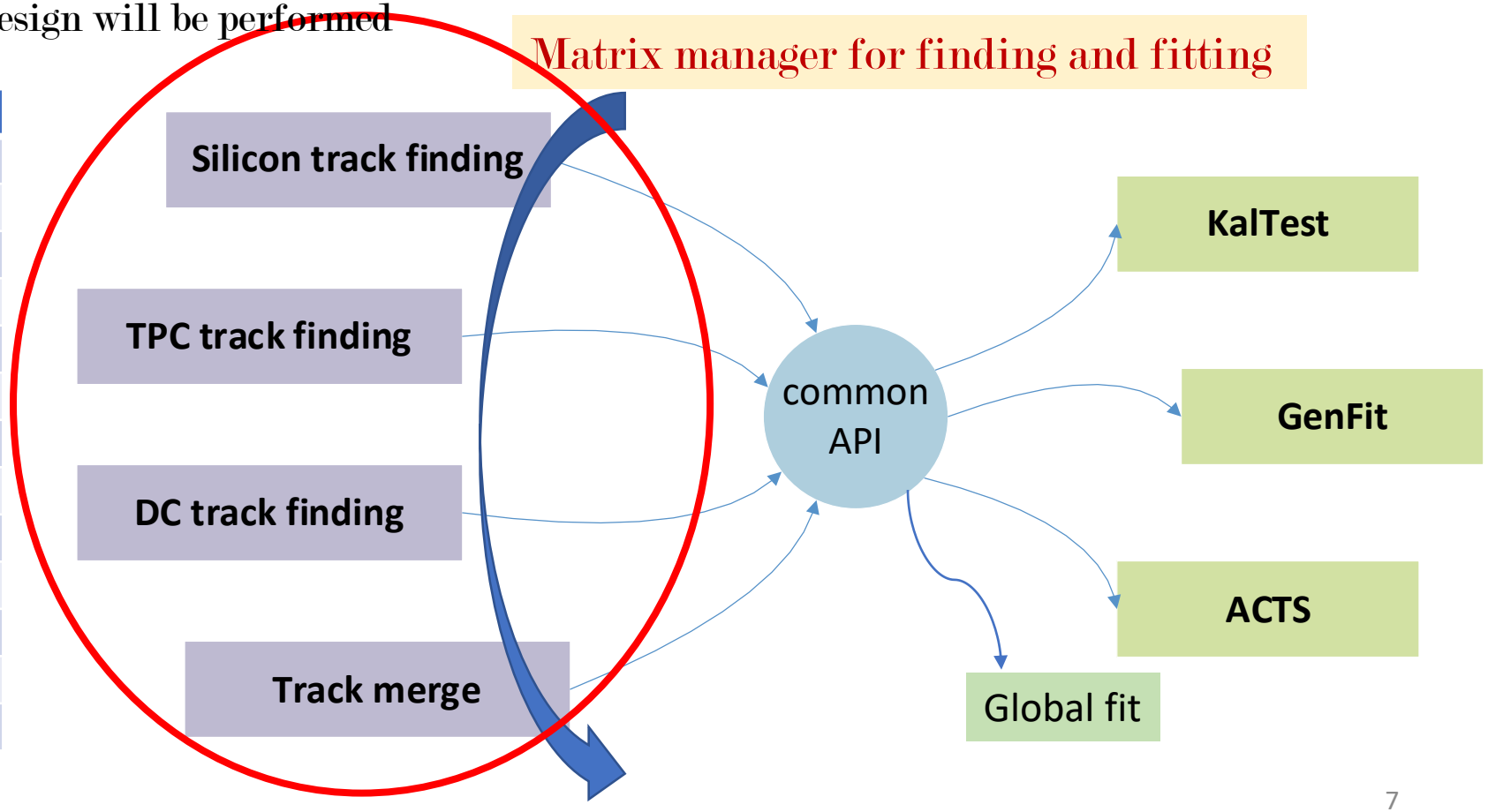


Propose of Tracking Chain

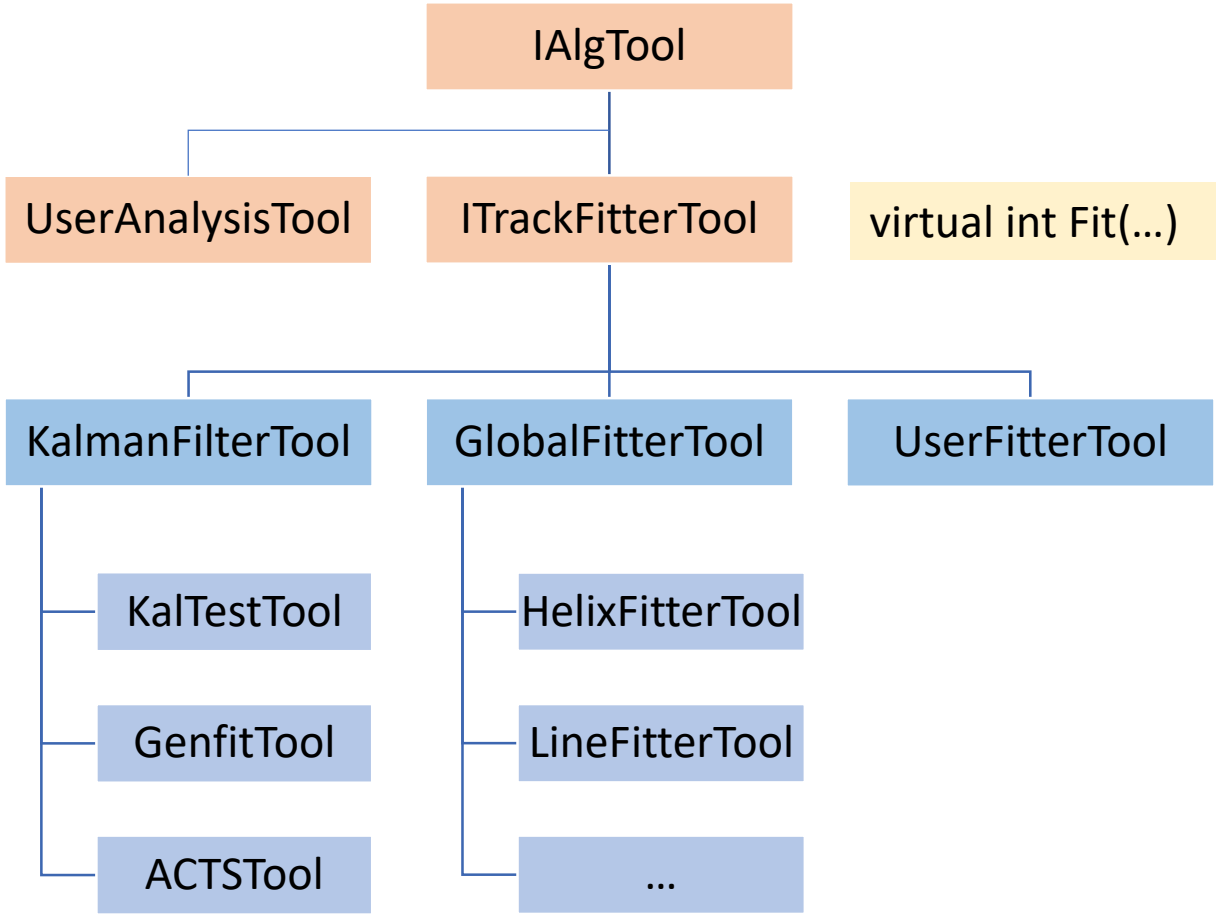
- Call the common API after track finding in all algorithm, and choose fitter according to option
- To choice best combination of track finding and fitting (global fit or Kalma filter?)
  - For middle tracking, low CPU time
  - For final tracking, high performance
- Comparison on same detector design will be performed

BEST: appropriate performance and CPU time

Content	Package	Status
Common API	Base API	done
	Kalman filter API	done
Fitter	KalTest	done
	DDKalTest	ongoing
	GenFit	ongoing
	ACTS	ongoing
	Global fitter	ongoing
Algorithm	SiliconTracking	done
	TrackSubset	done
	Clupatra	done
	FullLDCTracking	done
	ConformalTracking	done
	…	



Fitter API




```
# reconstruction
from Configurables import KalmanFilterTool
# Close multiple scattering and smooth, used by clupatra
kt010 = KalmanFilterTool("KalTest010")
kt010.MSOn = False
kt010.Smooth = False
#kt010.OutputLevel = DEBUG

# Open multiple scattering, energy loss and smooth (default)
kt111 = KalmanFilterTool("KalTest111")
kt111.MagneticField = 1
#kt111.OutputLevel = DEBUG

# Close smooth
kt110 = KalmanFilterTool("KalTest110")
kt110.Smooth = False
#kt110.OutputLevel = DEBUG

from Configurables import LineFitterTool
lft = LineFitterTool("GlobalLine")
#lft.OutputLevel = DEBUG
```

Choose according to tool name

```
tracking.FitterTool = "KalmanFilterTool/KalTest111"
```

```
tracking.FitterTool = "LineFitterTool/GlobalLine"
```

Output

edm4hep::Track

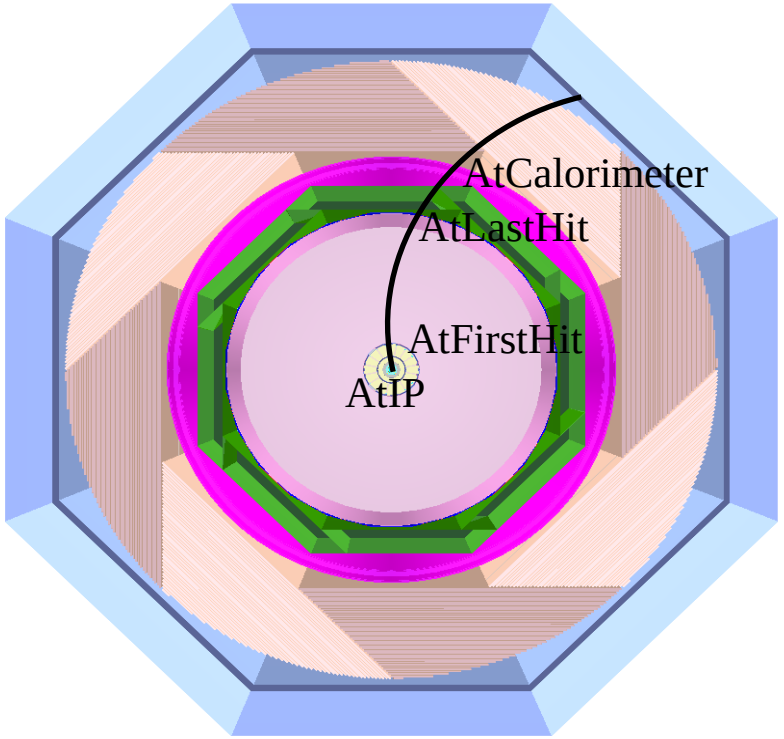
.....

edm4hep::TrackerHit s

.....

edm4hep::TrackState s

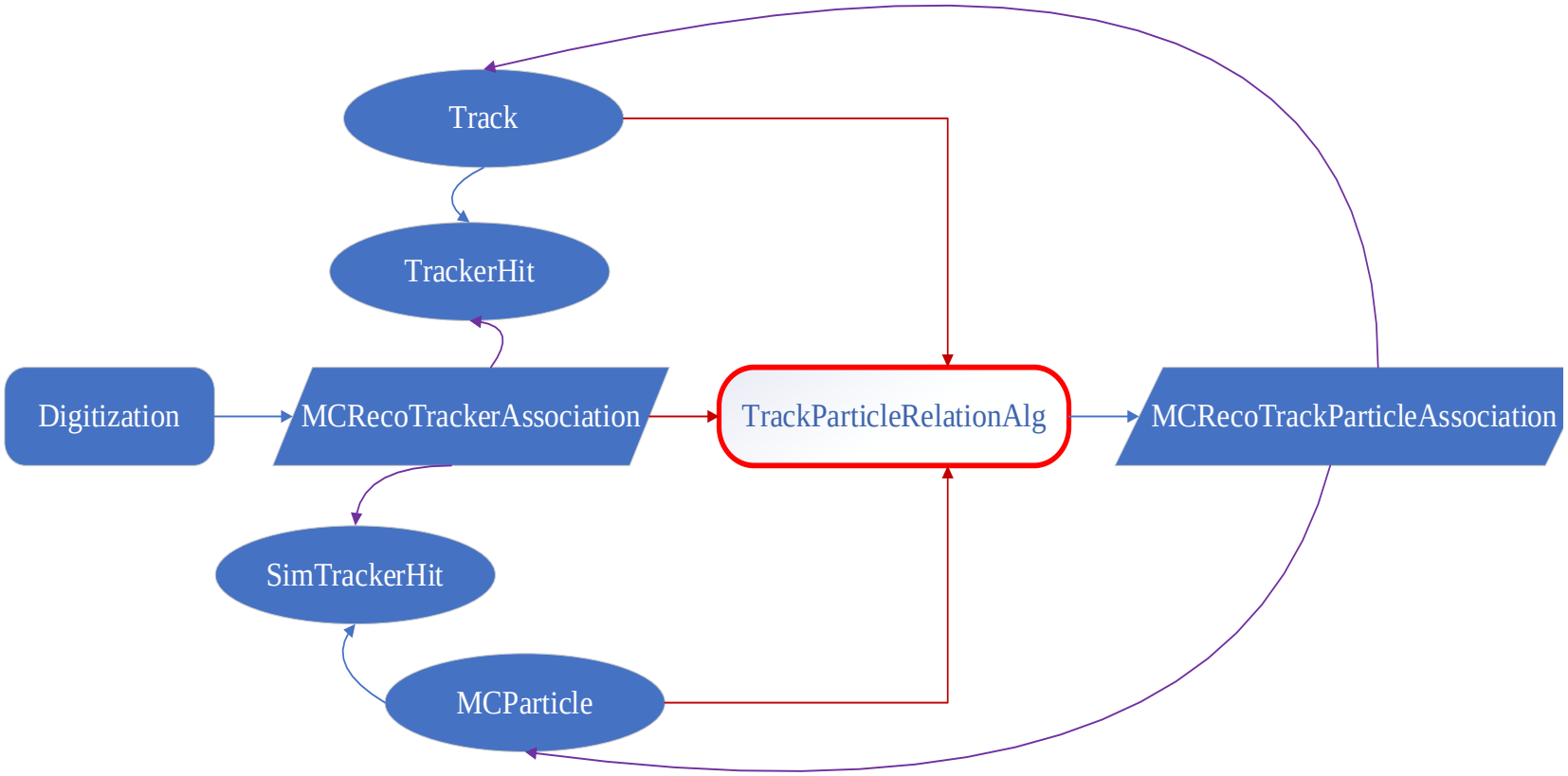
- std::int32_t location
- float D0
- float phi
- float omega
- float Z0
- float tanLambda
- float time
- edm4hep::Vector3f referencePoint
- std::array<float, 21> covMatrix

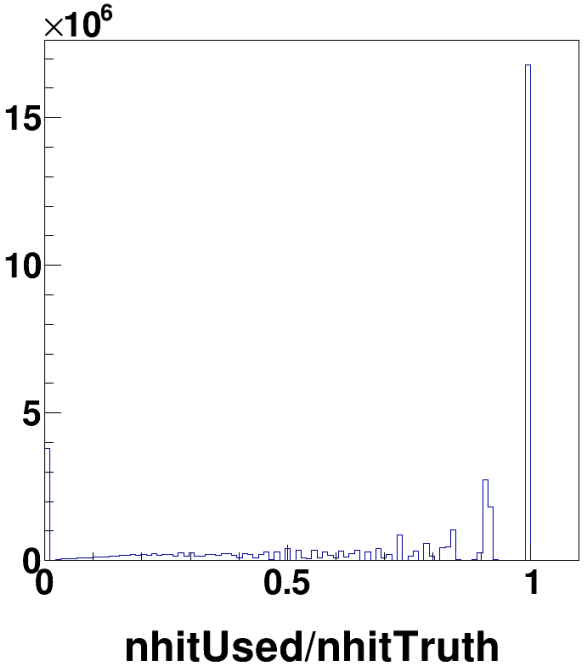


Currently, four TrackStates are in storage:
if(location==edm4hep::TrackState::AtIP)
if(location==edm4hep::TrackState::AtFirstHit)
if(location==edm4hep::TrackState::AtLastHit)
if(location==edm4hep::TrackState::AtCalorimeter)

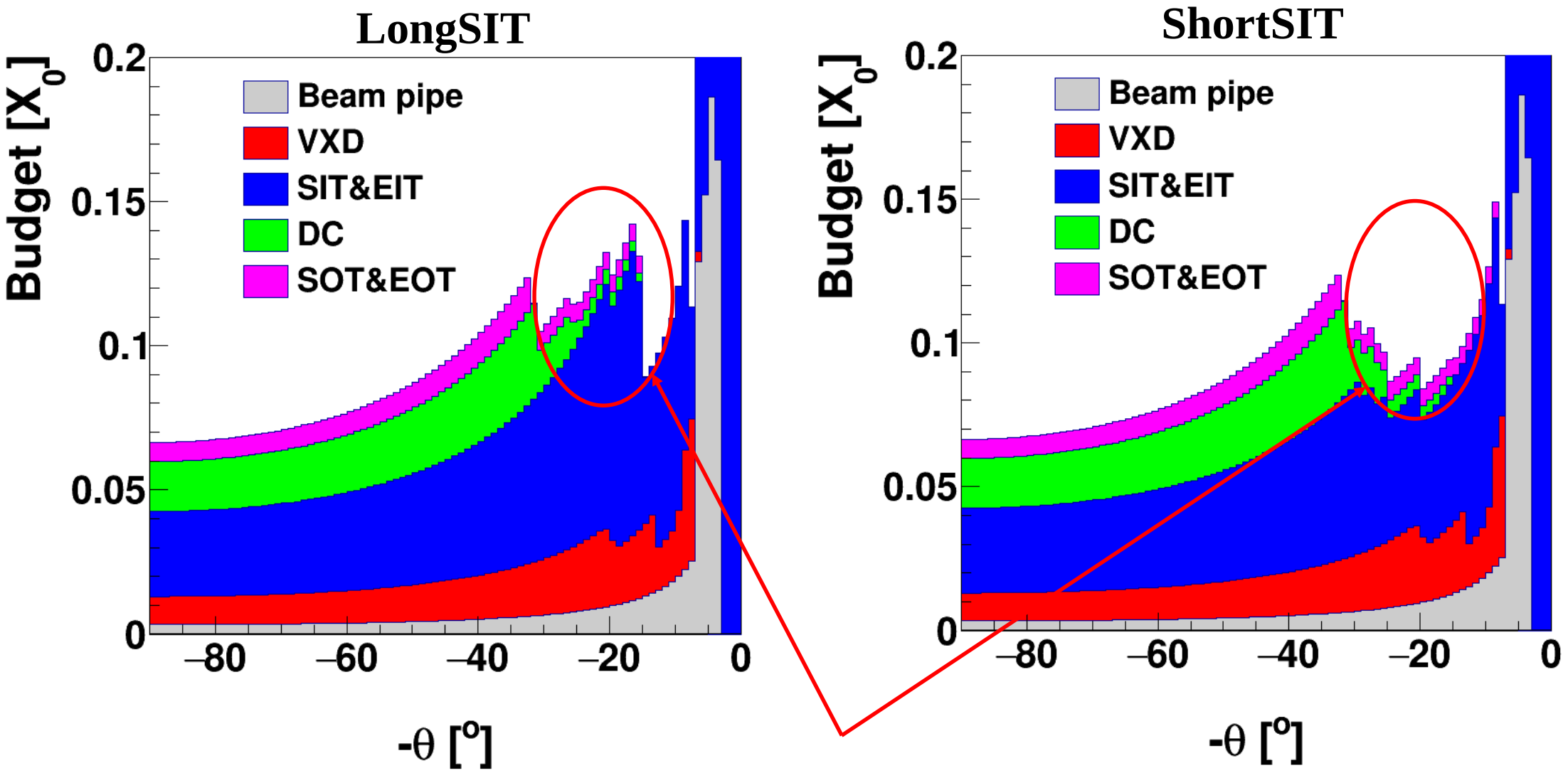
Association

- MCRecoTrackParticleAssociation
  - Track
  - MCParticle
  - weight: number of tracker hit linked between MCParticle and Track (NL), for a particle, found track (minimum requirement: NLmaximum≥4)

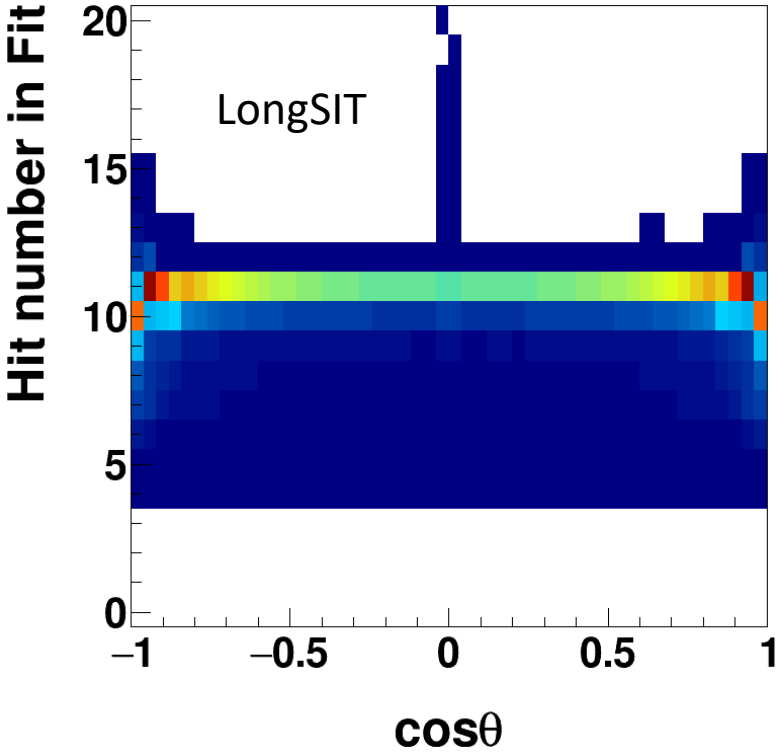


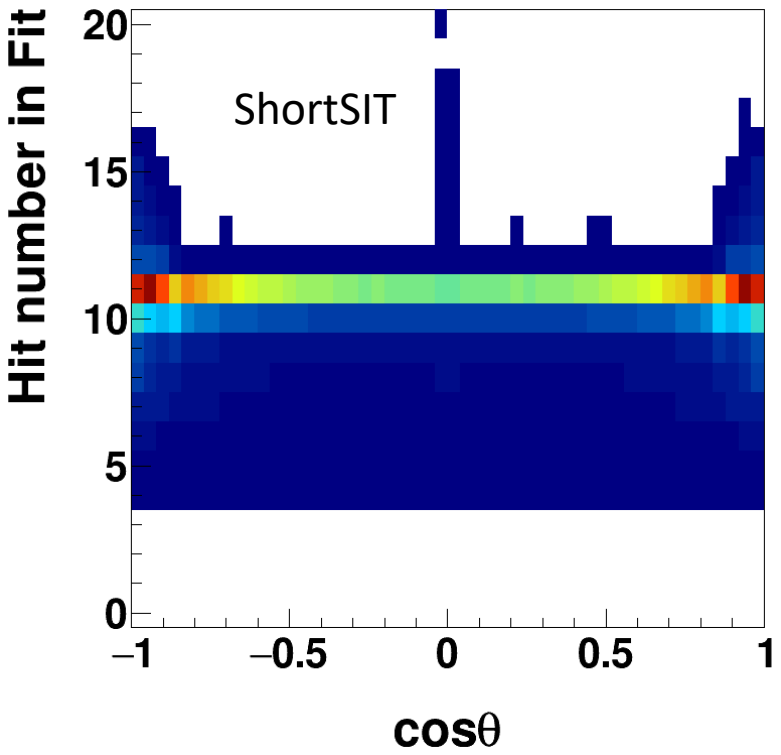


Material Budget

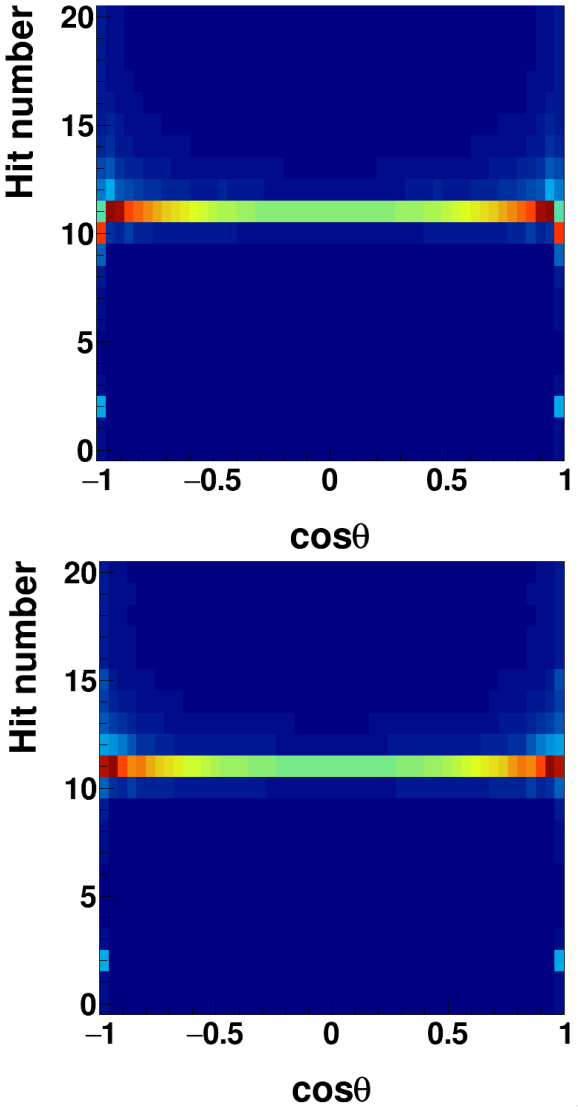


Hits Number





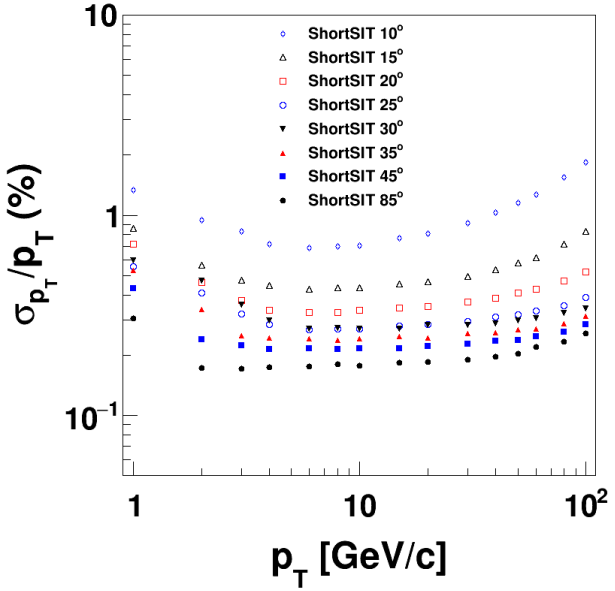
MC tructh

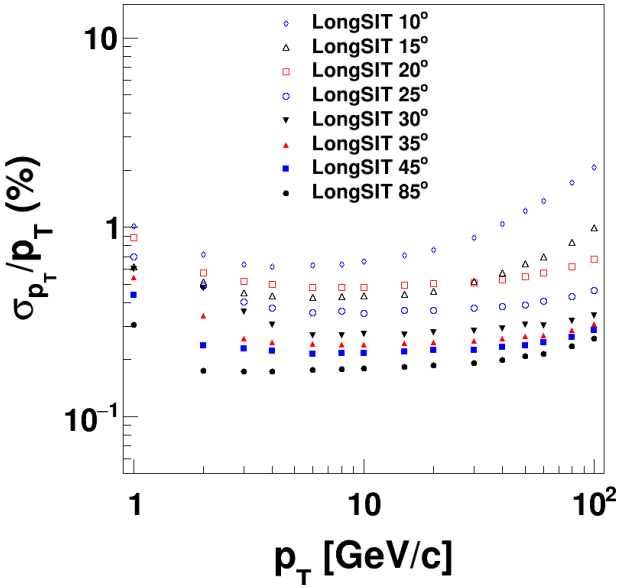


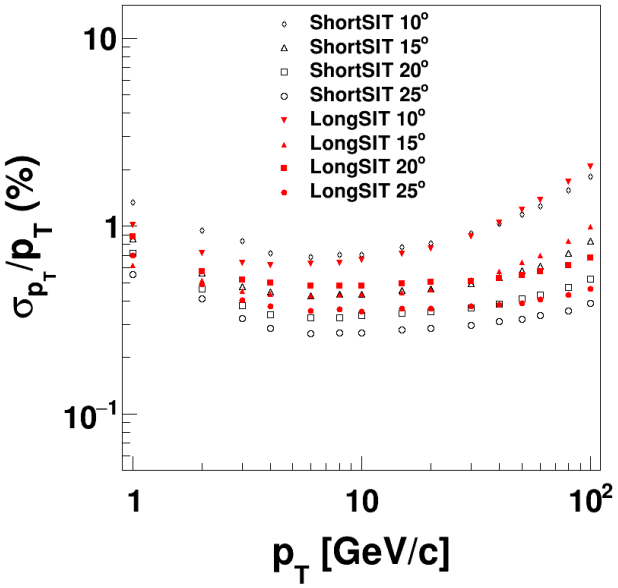
Resolution of p_T

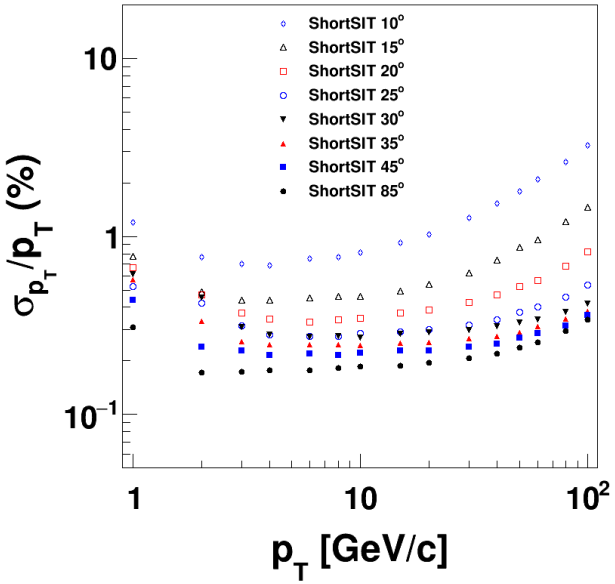
Fixed

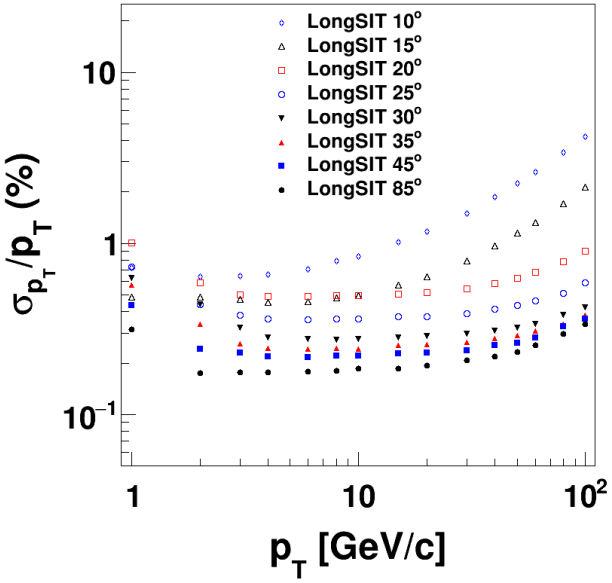
Parametrization

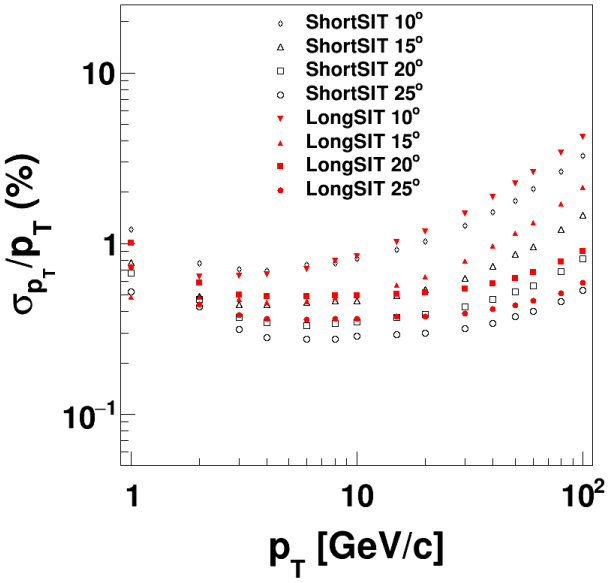








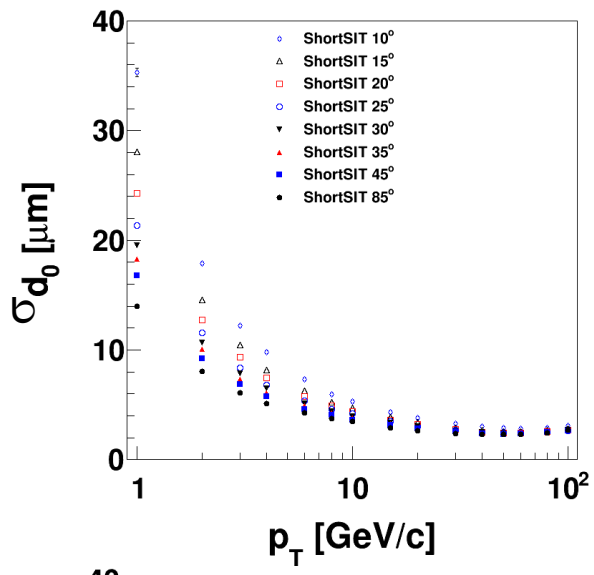


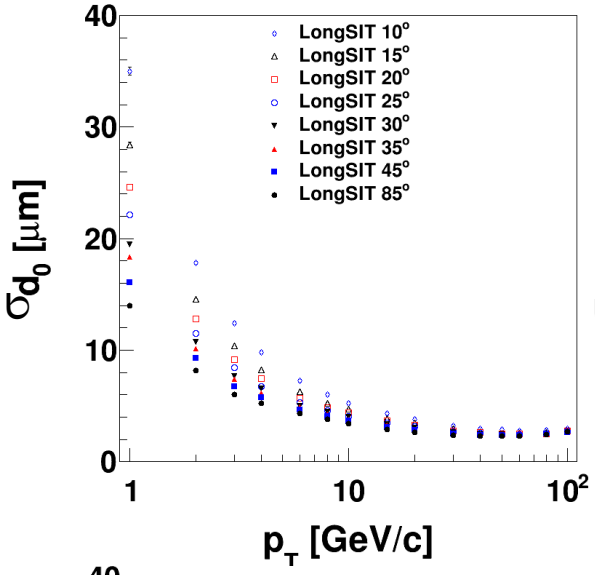


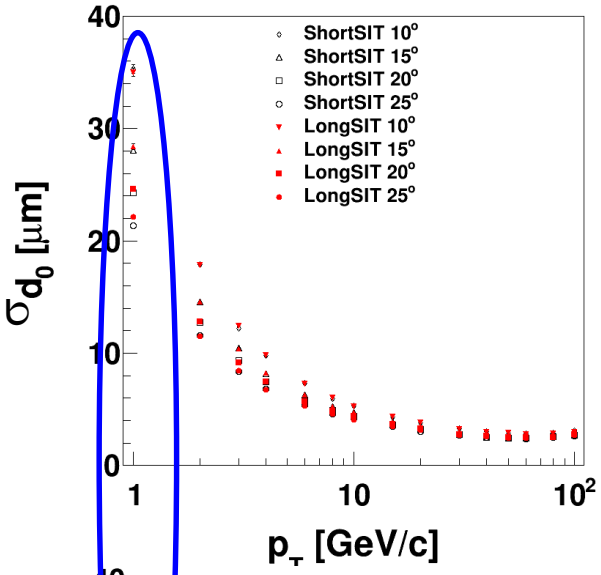
Resolution of d0

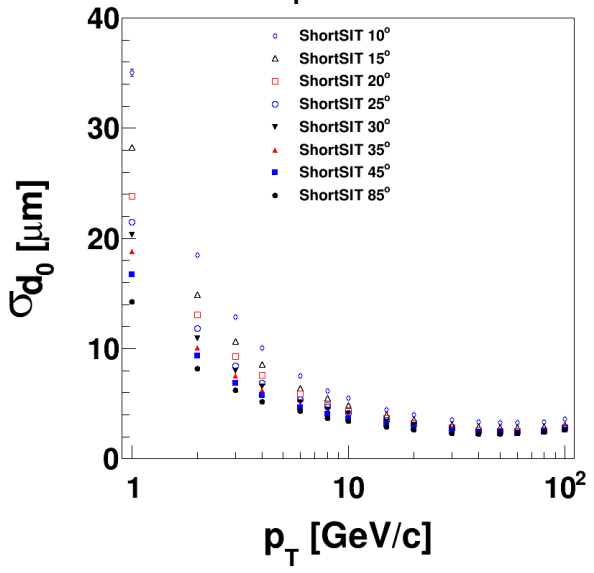
Fixed

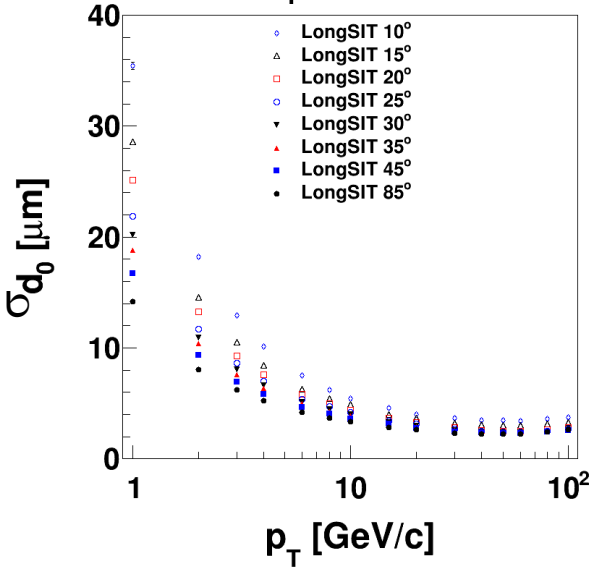
Parametrization

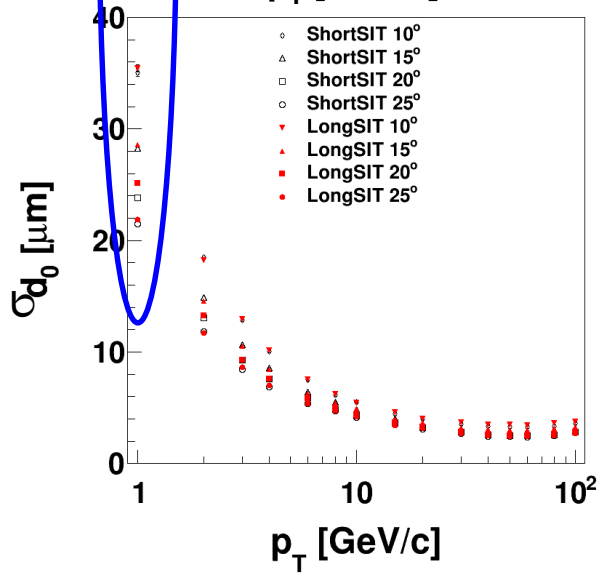




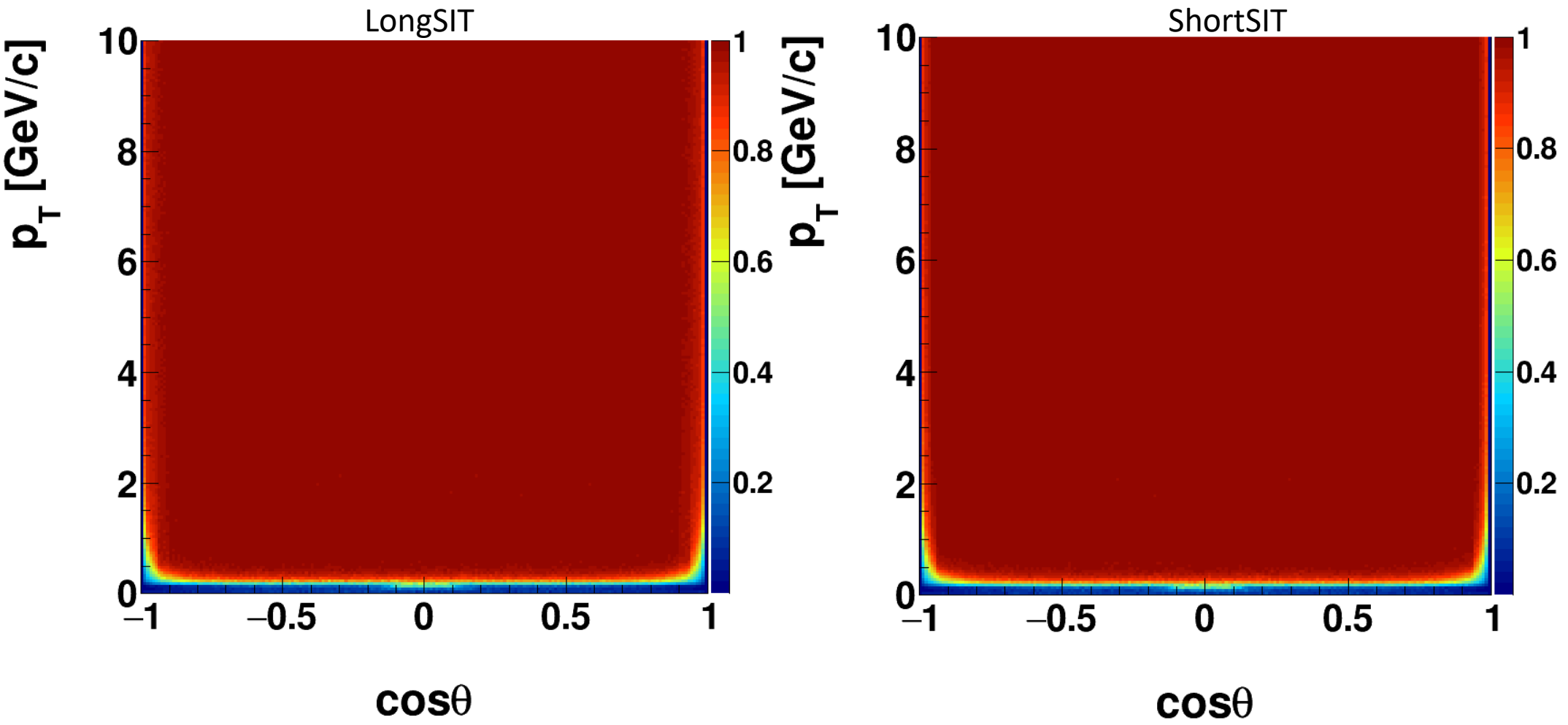








Efficiency

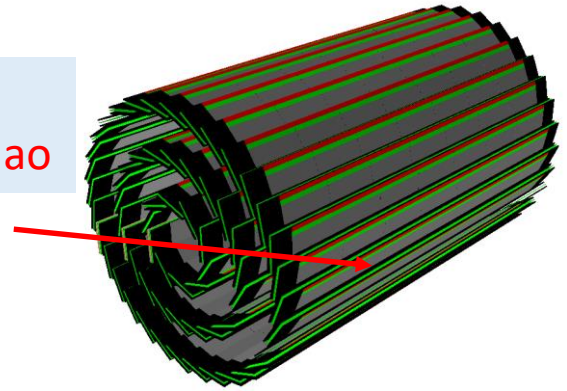


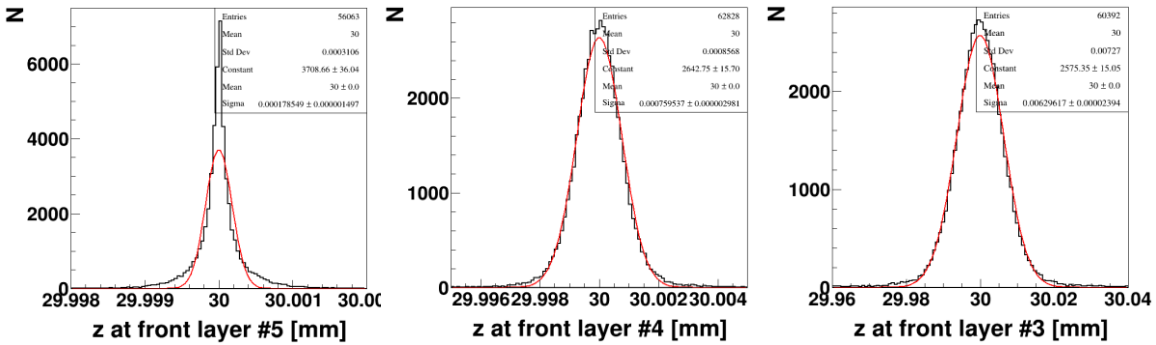
- ~86% @cosθ=(0.98-0.99)

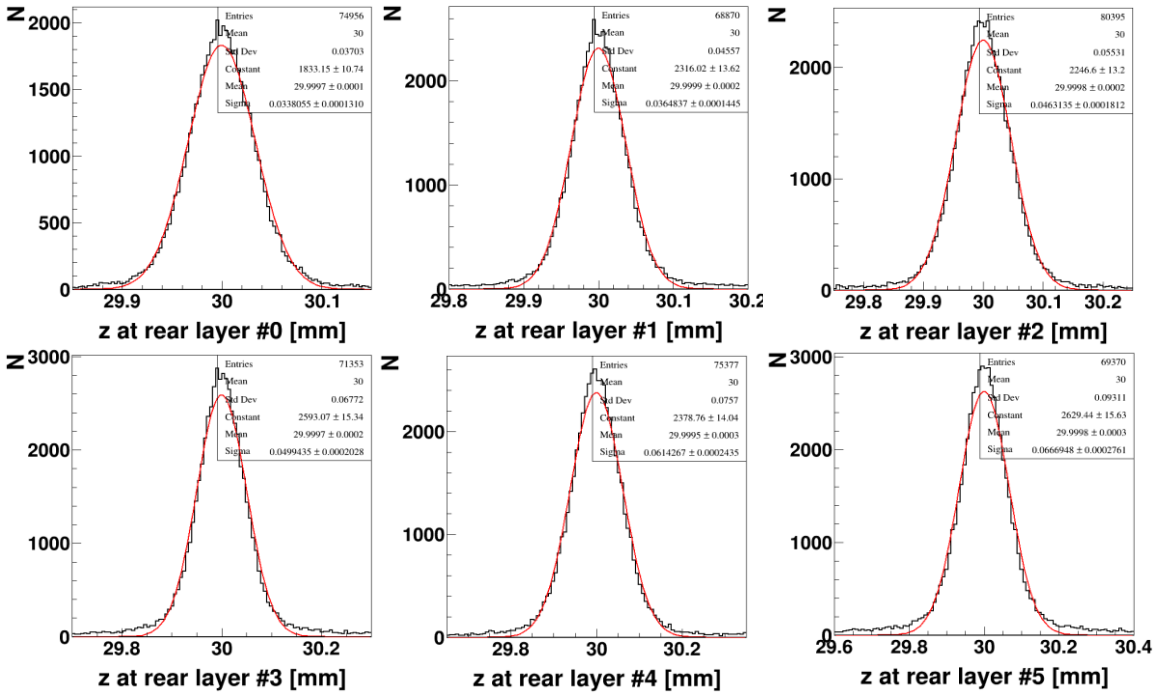
Geant4 Simulation for MOST2 Vertex

- ParticleGun
  - 5GeV electron
  - z = 30mm & θ=90°
  - x,y & ϕ to make sure
    - ✓ pass through (0,0)
    - ✓ pass through one ladder for each layer

Based on the module implemented by ZHENG Hao

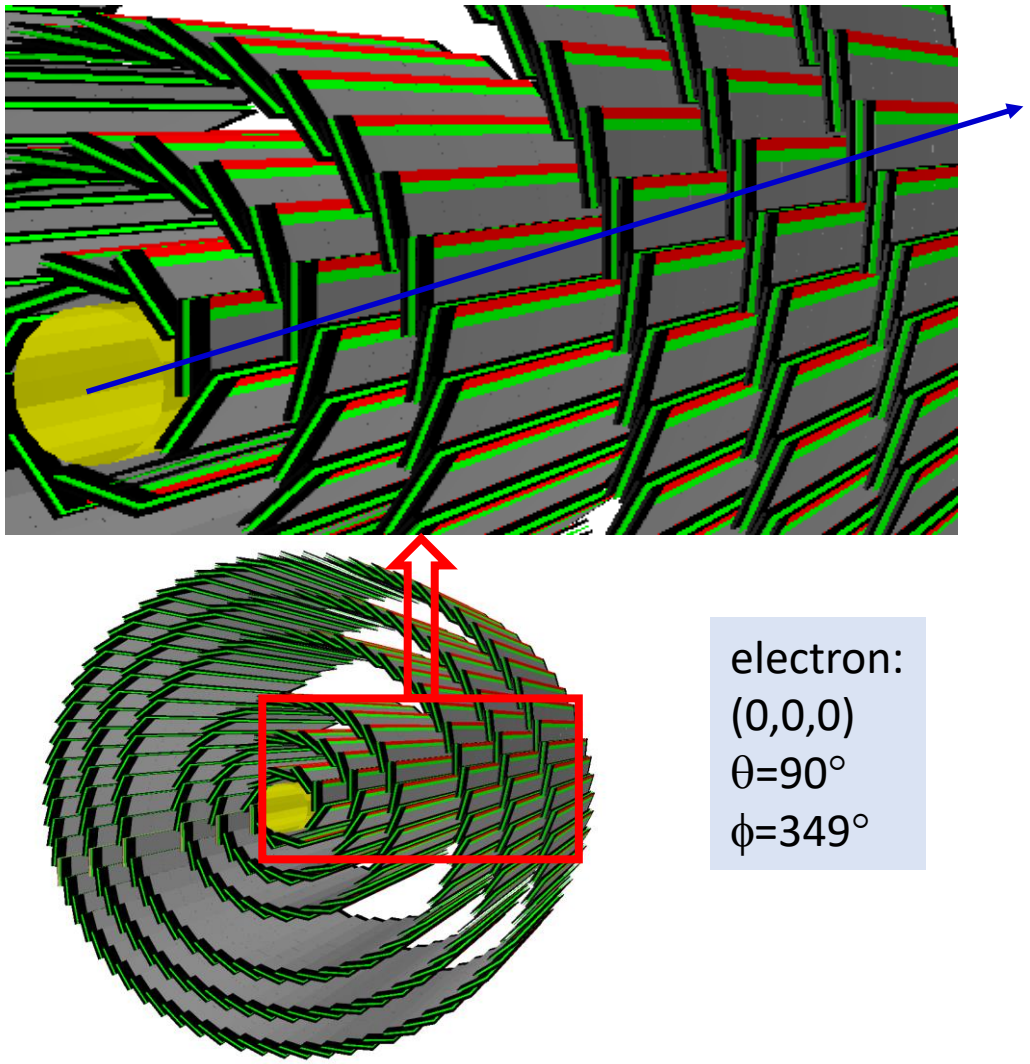




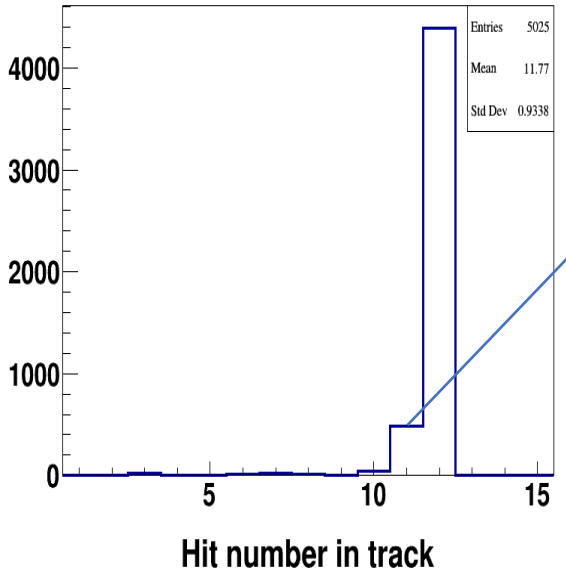


CEPCSW Tracking for MOST2 Vertex

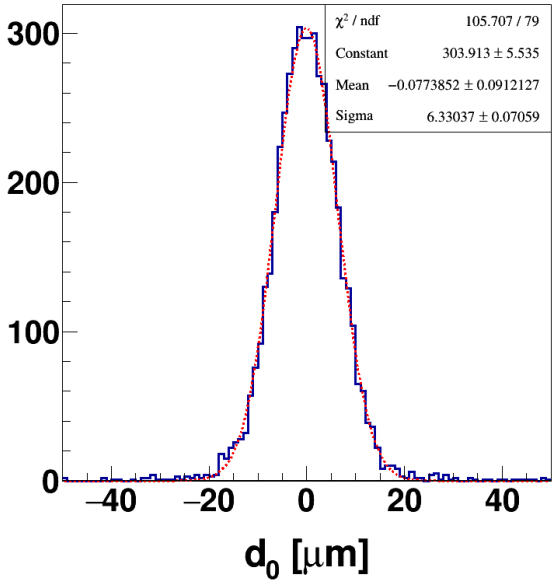
- Normal silicon tracking will obtain two tracks per beam particle, two ways to resolve
  - Merge algorithm
  - Virtual additional layout
- Virtual vertex
  - 6 doubly supper layer
  - Distance between 3rd and 4th equal to double of (0,0,0) to 1st
  - Same as beam test case
- Normal silicon tracking is possible to perform on beam test data
  - Coordinate transformation needed
- Use common fitter API, other user fitter or analysis can be called after track finding
  - Currently, the tracking software cannot work for zero magnetic field, so cheat algorithm a uniform magnetic field, such as 1T, while simulating in zero magnetic field

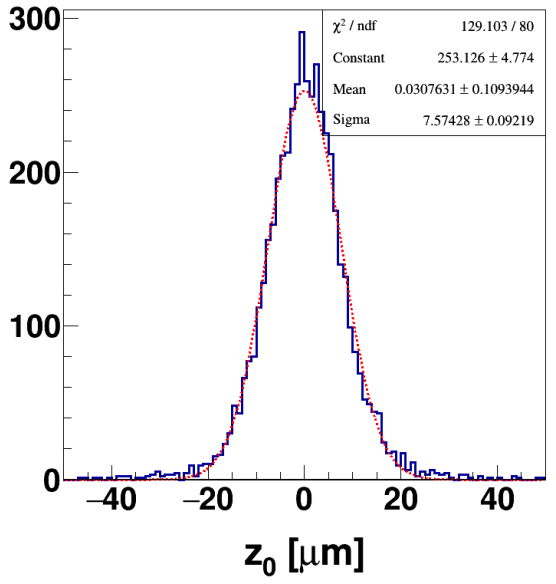


Result of Kalman Filter



Some hits are dropped in Kalman filter, only one in most of drop case



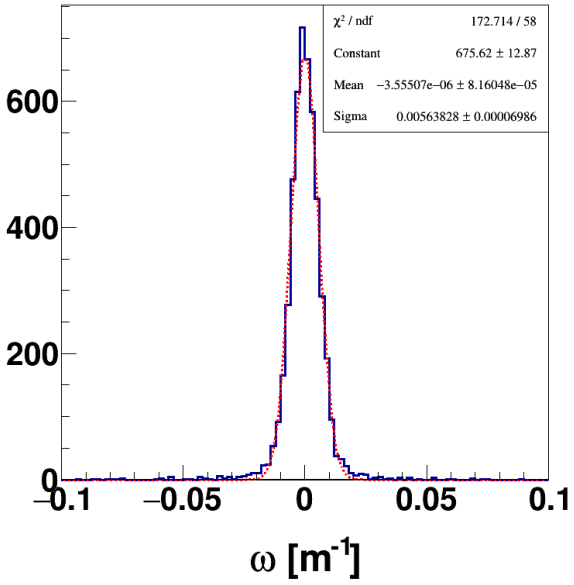


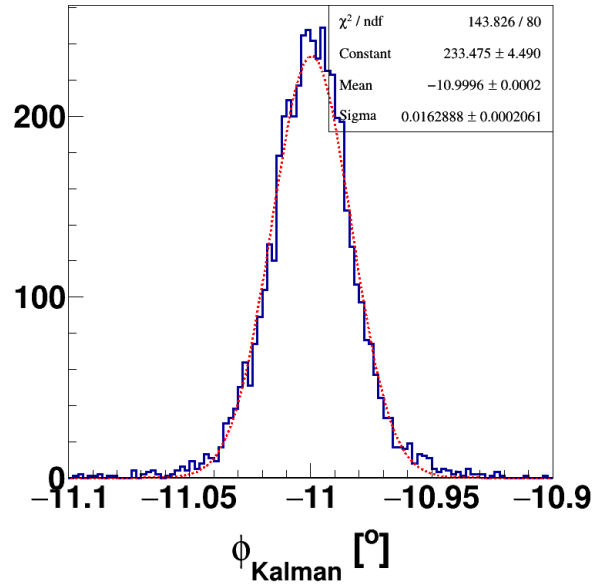
Although fitted by helix, the result shown the helix is very approximate to straight line.

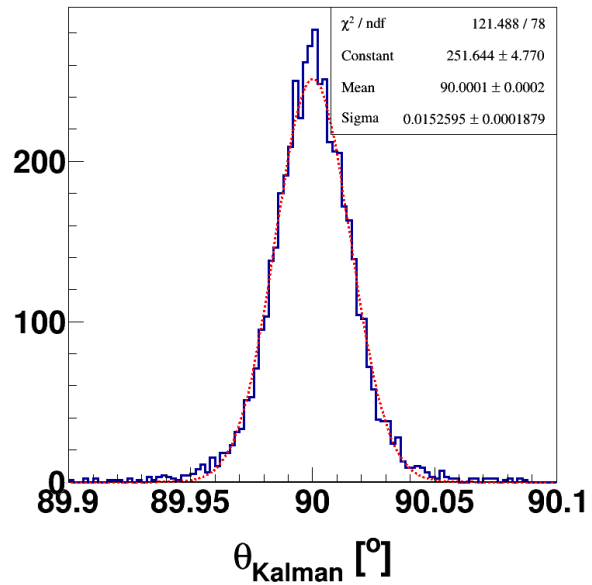
Large R of helix:

99.7%(3σ) R>59m

68.3%(1σ) R>178m







Global Line Fit

- Straight line
  - $x = x_0 + k\sin\theta\cos\phi$
  - $y = y_0 + k\sin\theta\sin\phi$
  - $z = z_0 + k\cos\theta$
- Minimization
  - $\chi^2 = \sum \frac{(u_i - u_{L2i})^2}{\sigma_{ui}^2} + \sum \frac{(v_i - v_{L2i})^2}{\sigma_{vi}^2}$
  - $\chi^2 = \sum \frac{(u_i - u_{L2i})^2}{\sigma_{ui}^2 + \Delta_{i,MS}^2} + \sum \frac{(v_i - v_{L2i})^2}{\sigma_{vi}^2 + \Delta_{i,MS}^2}$

Local position of point of line intersect to the ith layer

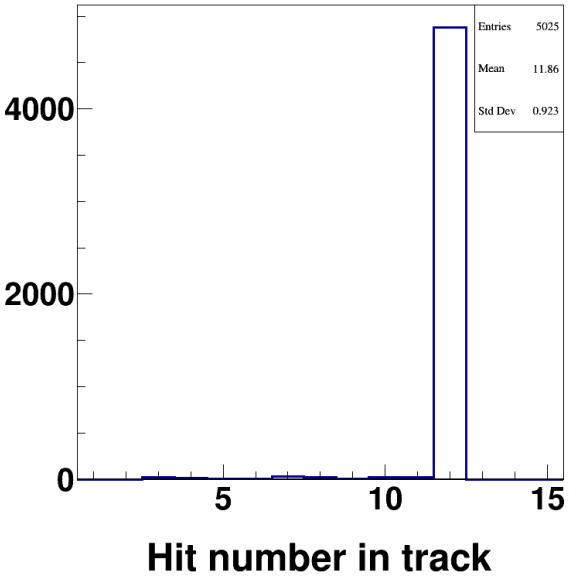
Local position of hit

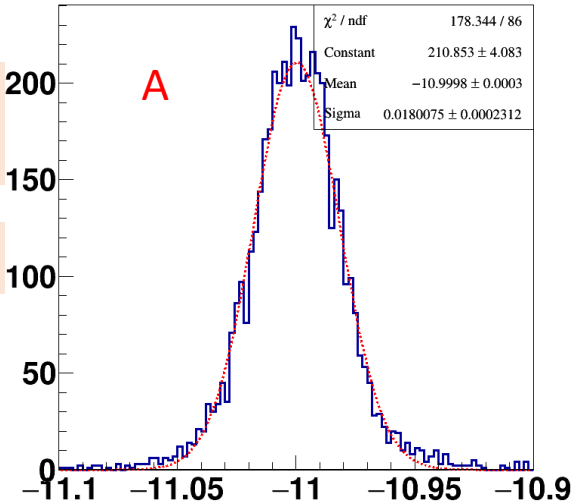
A

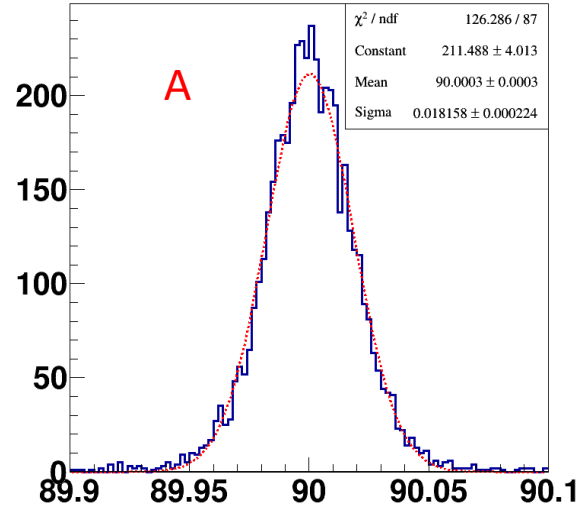
B

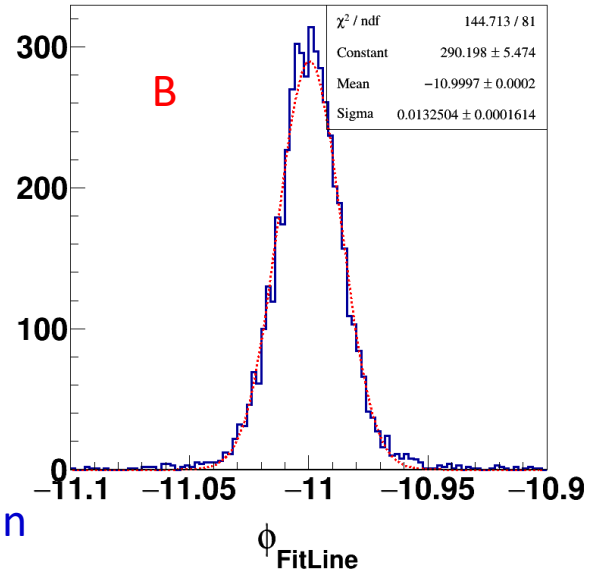
Resolution of the ith layer

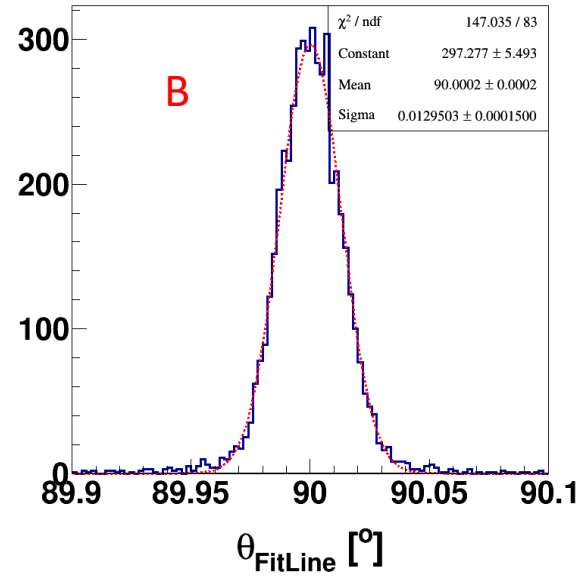
Uncertainty on the ith layer caused by multiple scattering











B obtain better resolution

Summary

- After continuing upgrade, we have more practicable silicon tracking software for various kind of application, which is validated in previous work in last workshop.
- Dominant improve
  - Parameterized resolution implement.
  - Common fitter API and tracking algorithm modification.
- Application test
  - Performance of silicon tracker design, shown understood results.
  - Simulation of Beam test for MOST2 vertex detector, good expectation to perform on the real data.

possible to run user analysis code through the common API

Thanks very much for your attention!