Vertexing @ ePIC

Xin Dong (LBNL)

Outline

- Current Workforce and Focus Meetings
- Vertexing Algorithm and Performance Test Status
- Diagnostic Progresses
- Summary and Plans

Workforce and Meetings

Current Work Force:

Lokesh Kumar (Panjab U., India)
Harsimran Singh (Panjab U.) - master student started ~ 4 months ago
Khushi Singla (Panjab U.) - master student just getting started

Sooraj Radhakrishnan (KSU/LBNL)
Joe Osborn (BNL)
Xin Dong (LBNL)

+ Shujie Li, Barak Schmookler (Reconstruction WG)
+ Ernst Sichtermann (Tracking WG)

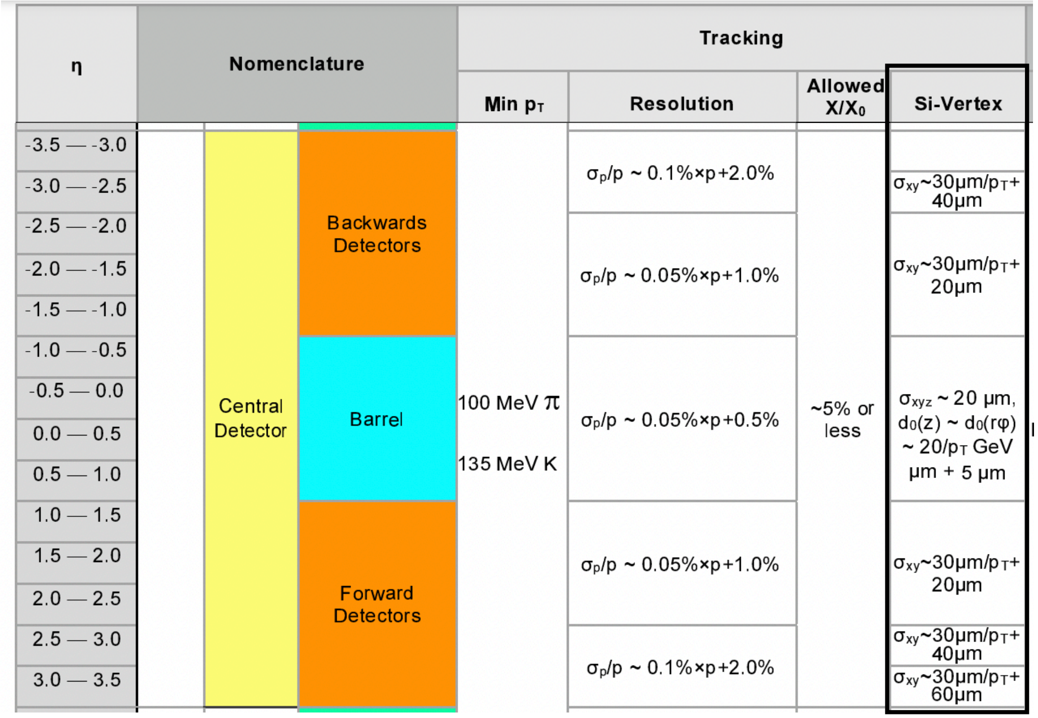
Bi-Weekly Focus Meeting:

Thursdays, 12pm BNL Time (next one Jan. 25)

+ Track Reconstruction weekly meeting (Thursdays, 10am BNL Time)

“Vertexing” for Yellow Report

<u>Detector Performance Matrix</u>



η	Nomenclature			Tracking			
				Min p_T	Resolution	Allowed X/X_0	Si-Vertex
-3.5 — -3.0		Central Detector	Backwards Detectors	100 MeV π 135 MeV K	σ_p/p ~ 0.1%×p+2.0%	~5% or less	
-3.0 — -2.5							σ_{xy}~30µm/p_T+ 40µm
-2.5 — -2.0					σ_p/p ~ 0.05%×p+1.0%		σ_{xy}~30µm/p_T+ 20µm
-2.0 — -1.5							
-1.5 — -1.0							
-1.0 — -0.5			Barrel		σ_p/p ~ 0.05%×p+0.5%		σ_{xyz} ~ 20 µm, $d_0(z)$ ~ $d_0(r\varphi)$ ~ 20/p_T GeV µm + 5 µm
-0.5 — 0.0							
0.0 — 0.5							
0.5 — 1.0							
1.0 — 1.5			Forward Detectors		σ_p/p ~ 0.05%×p+1.0%		σ_{xy}~30µm/p_T+ 20µm
1.5 — 2.0							
2.0 — 2.5							
2.5 — 3.0					σ_p/p ~ 0.1%×p+2.0%		σ_{xy}~30µm/p_T+ 40µm
3.0 — 3.5							σ_{xy}~30µm/p_T+ 60µm

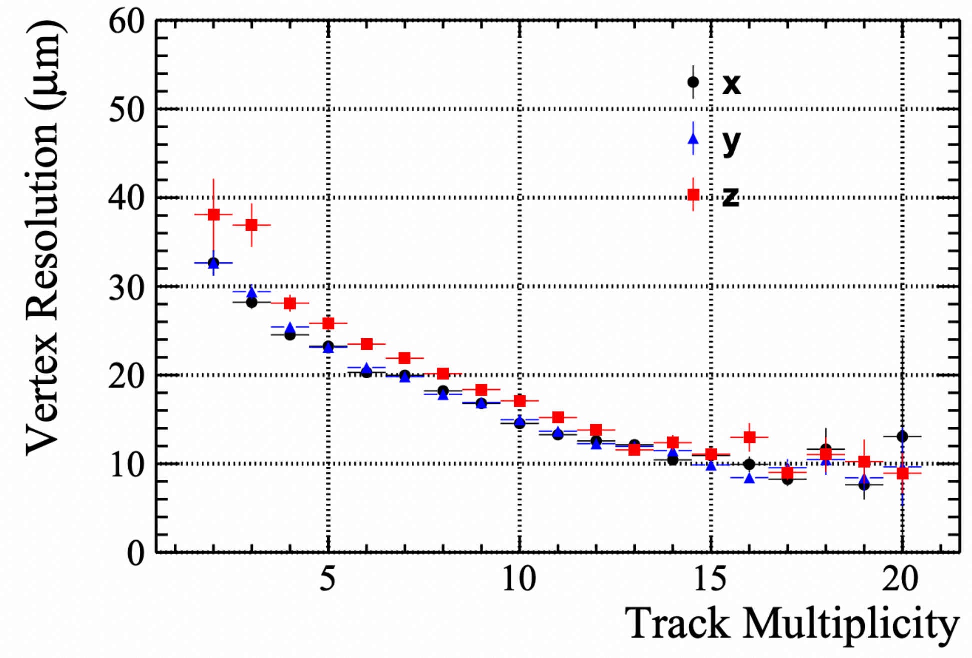
Yellow Report

<u>Fast simulation:</u>

DCA point smearing according to YR DCA resolution

Simple average of DCA position of primary tracks

- no error or covariance



M. Kelsey et. al., PRD 104 (2021) 054002

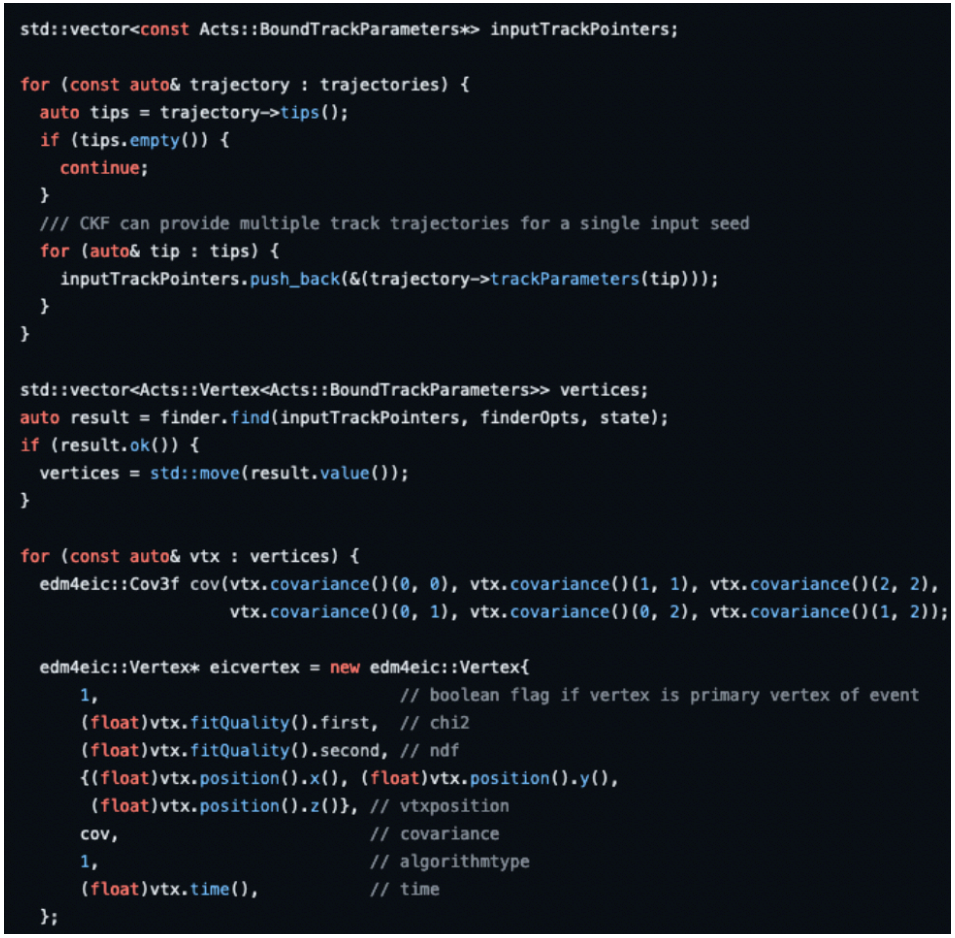
Current Vertexing Algorithm in EICrecon

See details in Joe Osborn’s presentation on May 18 at ePIC Track Reconstruction Meeting

https://indico.bnl.gov/event/19358/contributions/76588/attachments/47593/80693/vertexing.pdf

- Acts::IterativeVertexFinder implemented in EICrecon
- Trajectories used as input to Acts
- Fitted vertices filled into edm4eic::Vertex objects, stored in PODIO output

edm4eic::Vertex struct missing key fields
- to-be-udpated (see discussion later)



```
std::vector<const Acts::BoundTrackParameters*> inputTrackPointers;

for (const auto& trajectory : trajectories) {
  auto tips = trajectory->tips();
  if (tips.empty()) {
    continue;
  }
  /// CKF can provide multiple track trajectories for a single input seed
  for (auto& tip : tips) {
    inputTrackPointers.push_back(&(trajectory->trackParameters(tip)));
  }
}

std::vector<Acts::Vertex<Acts::BoundTrackParameters>> vertices;
auto result = finder.find(inputTrackPointers, finderOpts, state);
if (result.ok()) {
  vertices = std::move(result.value());
}

for (const auto& vtx : vertices) {
  edm4eic::Cov3f cov(vtx.covariance()(0, 0), vtx.covariance()(1, 1), vtx.covariance()(2, 2),
                     vtx.covariance()(0, 1), vtx.covariance()(0, 2), vtx.covariance()(1, 2));

  edm4eic::Vertex* eicvertex = new edm4eic::Vertex{
      1,                              // boolean flag if vertex is primary vertex of event
      (float)vtx.fitQuality().first,  // chi2
      (float)vtx.fitQuality().second, // ndf
      {(float)vtx.position().x(), (float)vtx.position().y(),
       (float)vtx.position().z()}, // vtxposition
      cov,                         // covariance
      1,                           // algorithmtype
      (float)vtx.time(),           // time
  };
```

Performance Tests

Two Simulation Tests

1) A-Few-Track Simulation:

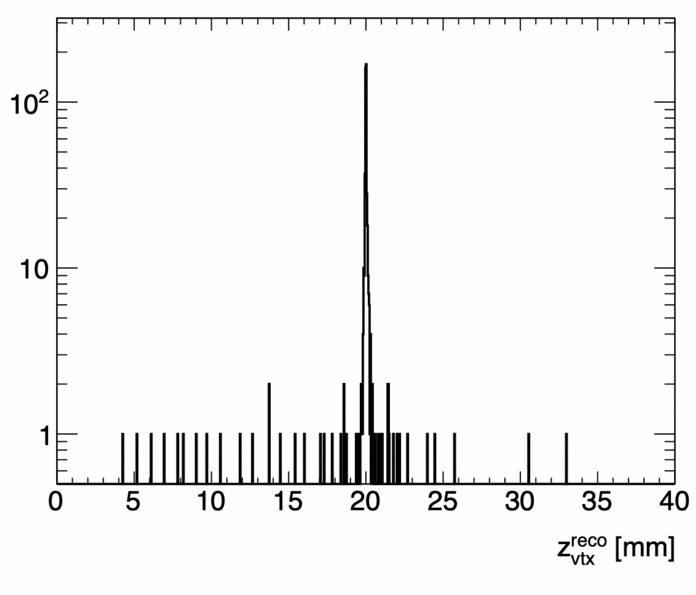
N pions thrown flat in acceptance and flat in 0.2<pT<5 GeV at a fixed vertex (0,0,20) mm

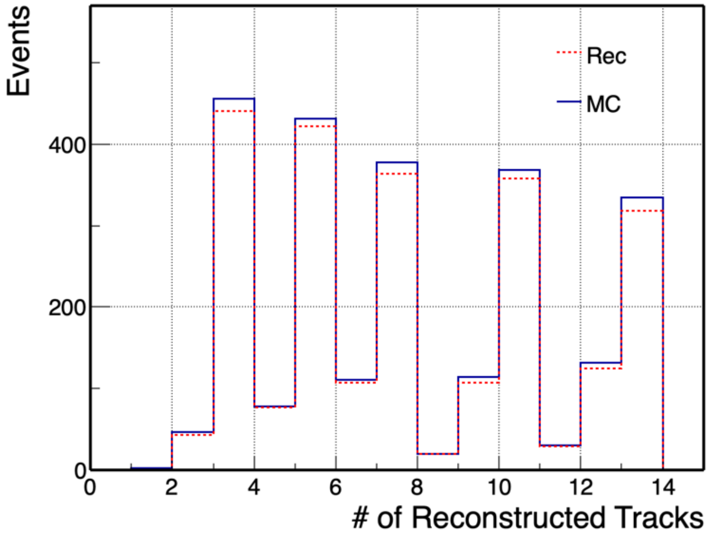
2) DIS sim events on S3: *eictest/EPIC/FULL/23.xx.x/epic_YYYYYY/*
*…/10x100/minQ2=10/pythia8NCDIS_10x100_minQ2=10_beamEffects_xAngle=-0.025_hiDiv_**

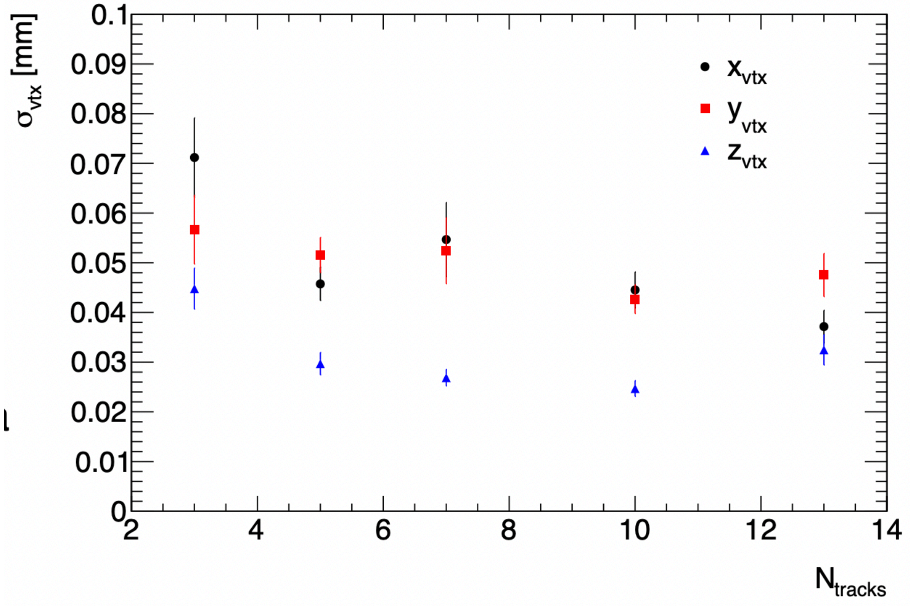
xx.xx/epic_YYYYYY:
23.05.2/epic_brycecanyon
23.08.0/epic_craterlake

Test with a-few-track Events

Simulation: N pions thrown flat in acceptance and flat in 0.2<pT<5 GeV at a fixed vertex (0,0,20) mm



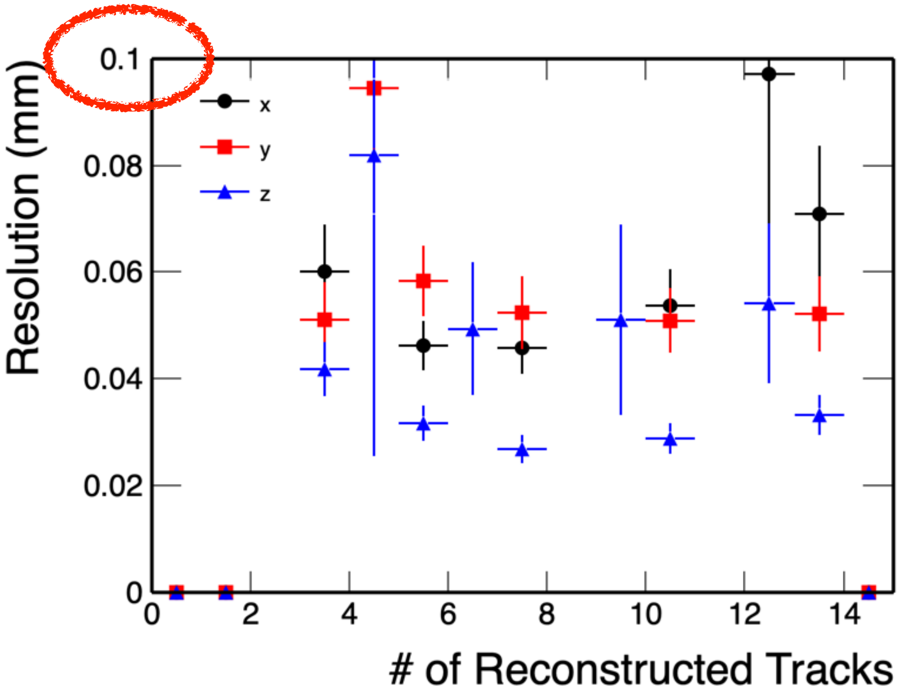


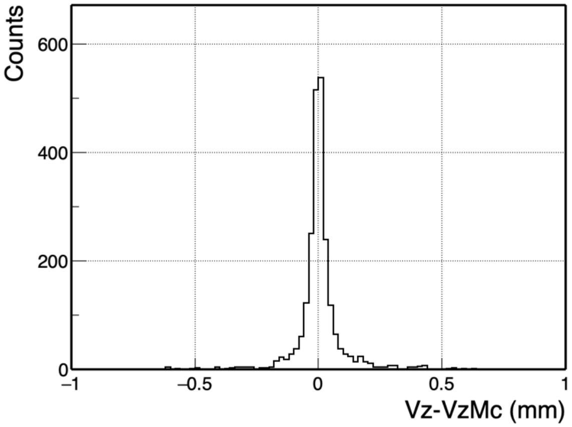


Joe Osborn, XD

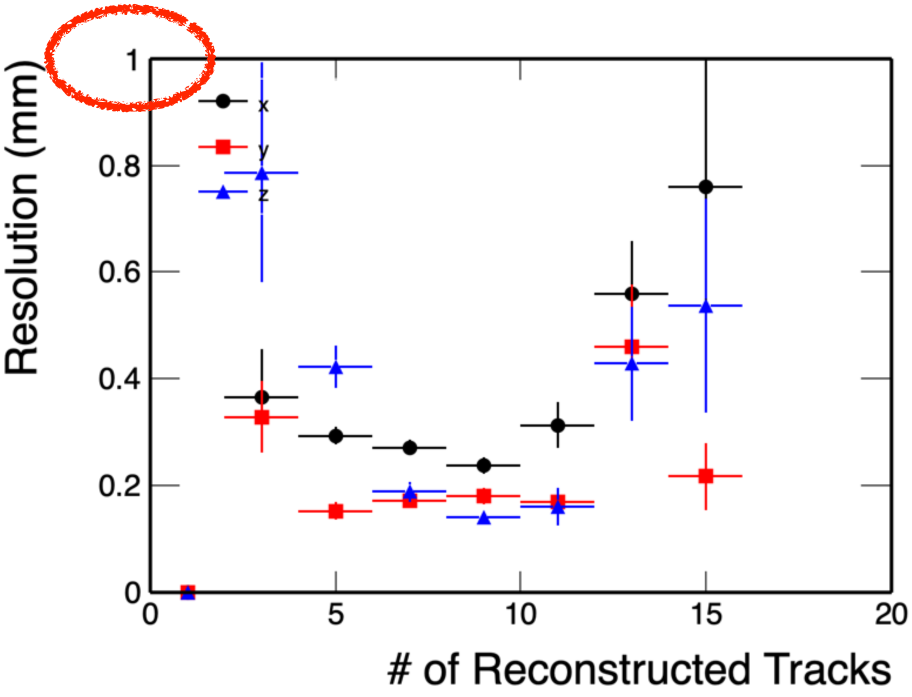
Test with DIS Events on S3

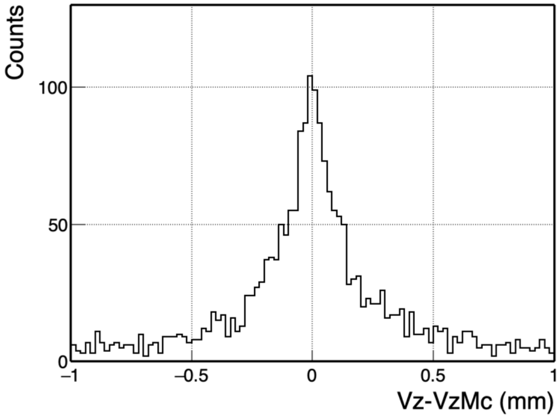
Few-track Simu





PYTHIA DIS S3 Events

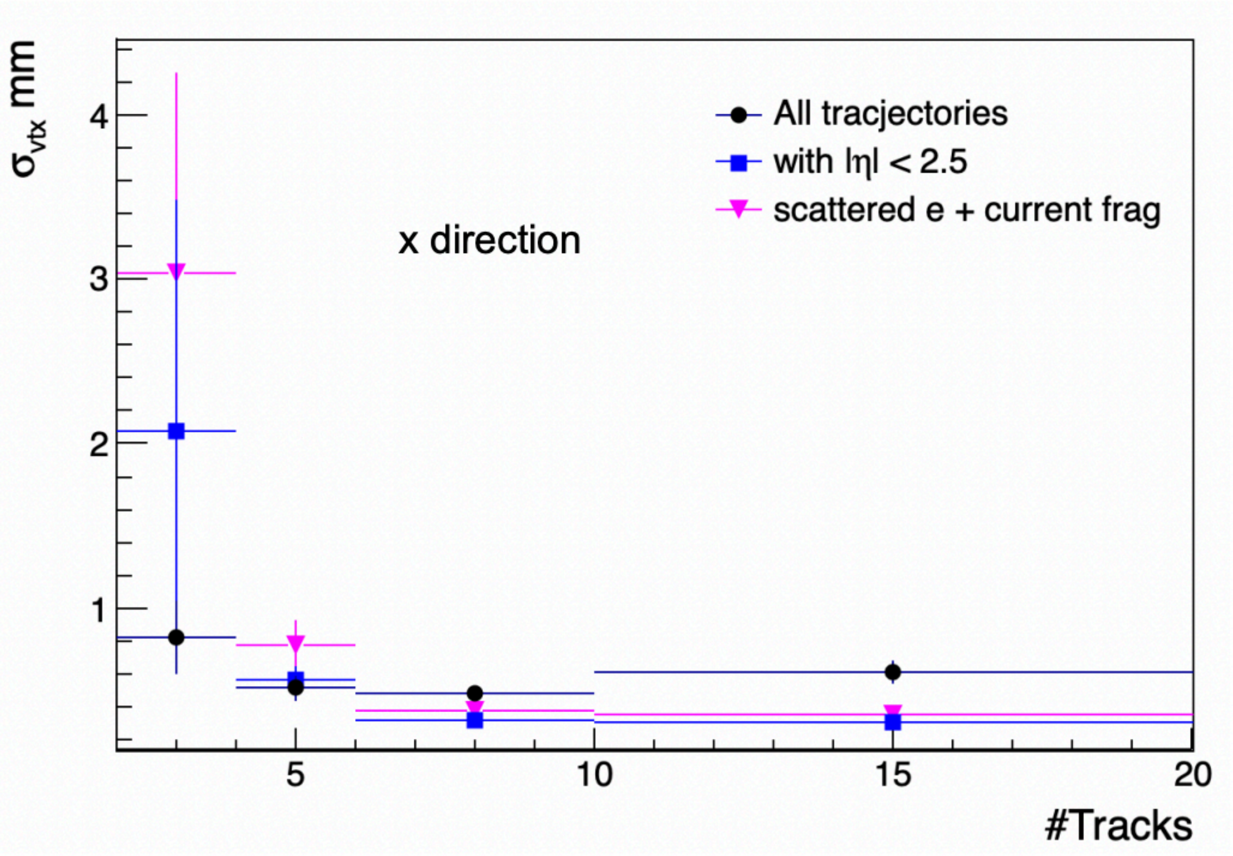




Vertex resolutions: a factor of ~4-5 worse in DIS events compared to few-track events

Further Test on DIS Events

Sooraj Radhakrishnan

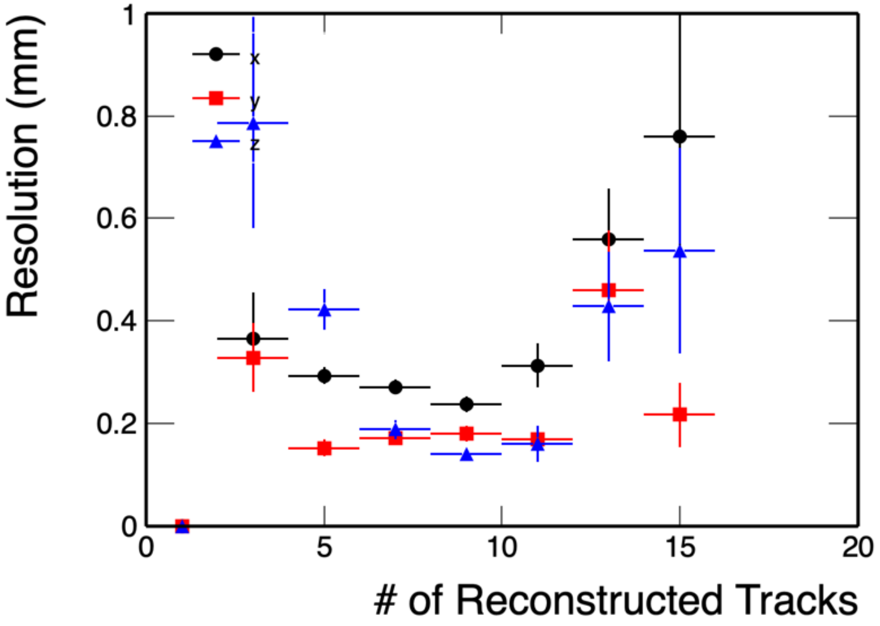


Different filters for selecting the trajectories used for vertexing:

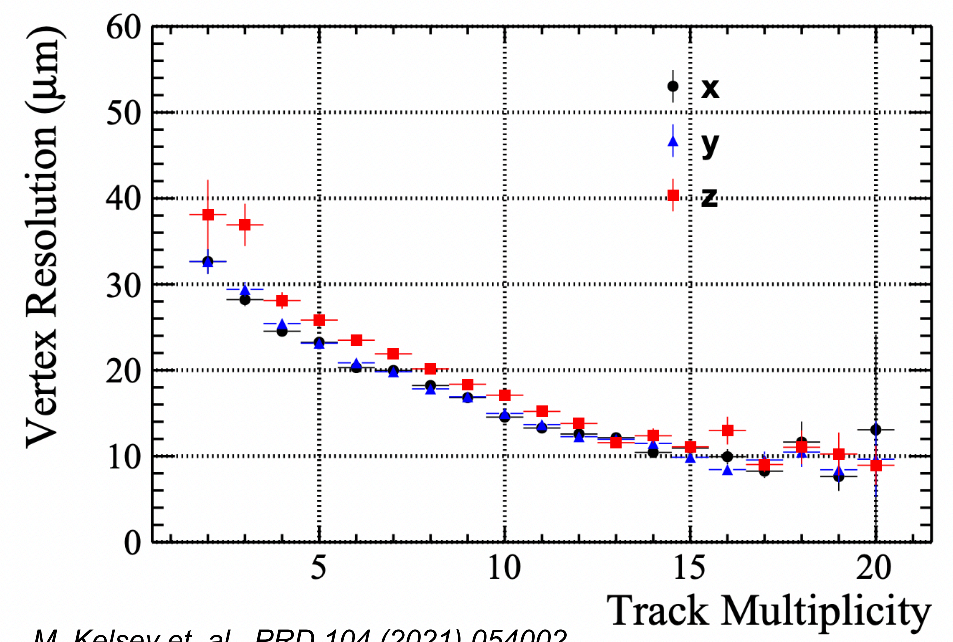
1) Restricting tracks within |eta|<2.5 helps on the resolution - as expected
2) Using scattered electrons and leading hadron also leads to a better resolution
   - *Scattered electron taken as track with charge = -1 at most negative rapidity*
   - *Leading hadrons in transverse plane opposite to it within delta_phi of π/4*

Recap on DIS Event Vertexing

10x100 GeV GEANT + EICrecon



18x275 GeV   Fast simulation



M. Kelsey et. al., PRD 104 (2021) 054002

Fast simulation, 3T field, 18x275 GeV, Q^2>1 GeV^2,
Yellow Report performance / simple DCA average

Strategies:

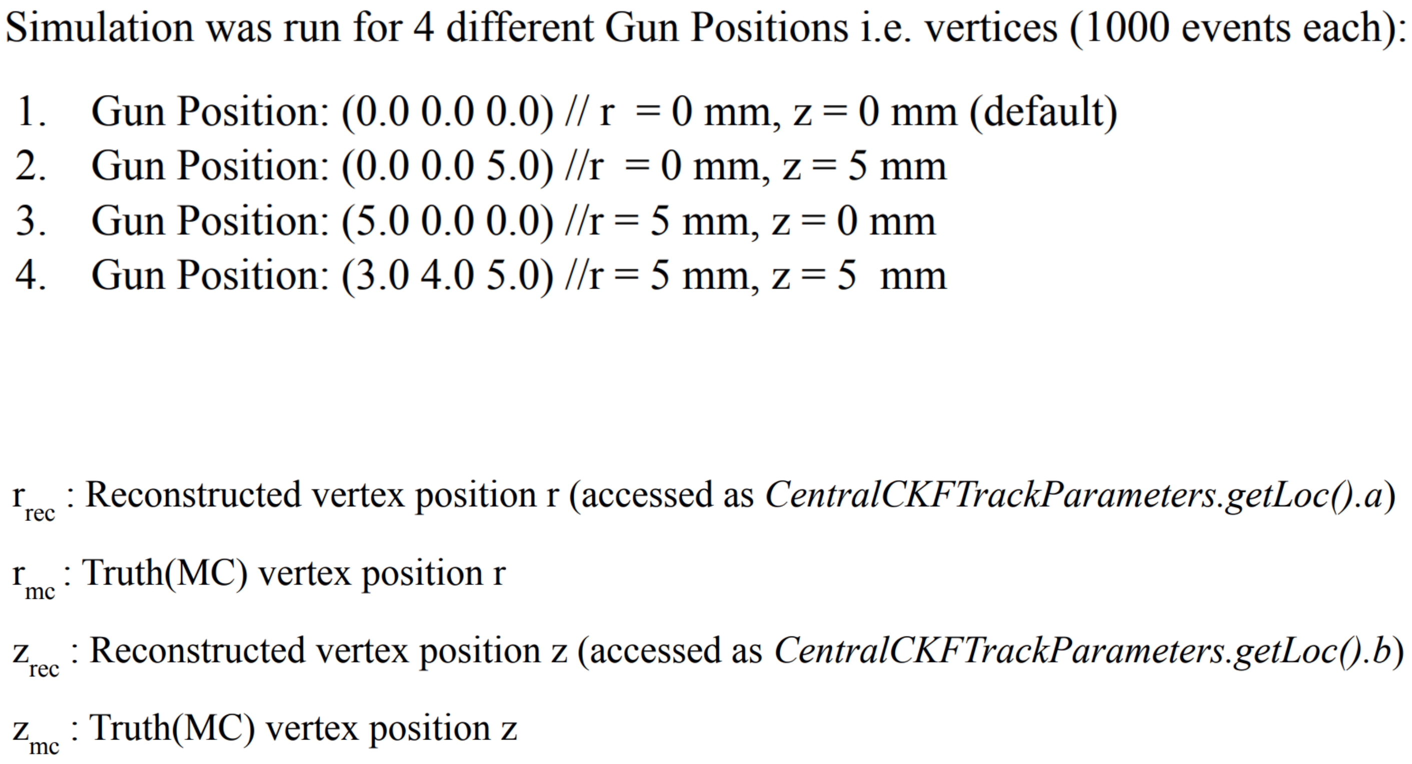
1) Few-track simulation, check single track performances (resolution, efficiency, DCA), then moving away from (0,0) in (x,y)   - Harsimran S.

2) PYTHIA DIS simulation, starting from (0,0,0)   - Khushi S. just getting started

Progresses and Findings

Harsimran Singh

10 muons per event, flat theta, pT



Simulation was run for 4 different Gun Positions i.e. vertices (1000 events each):

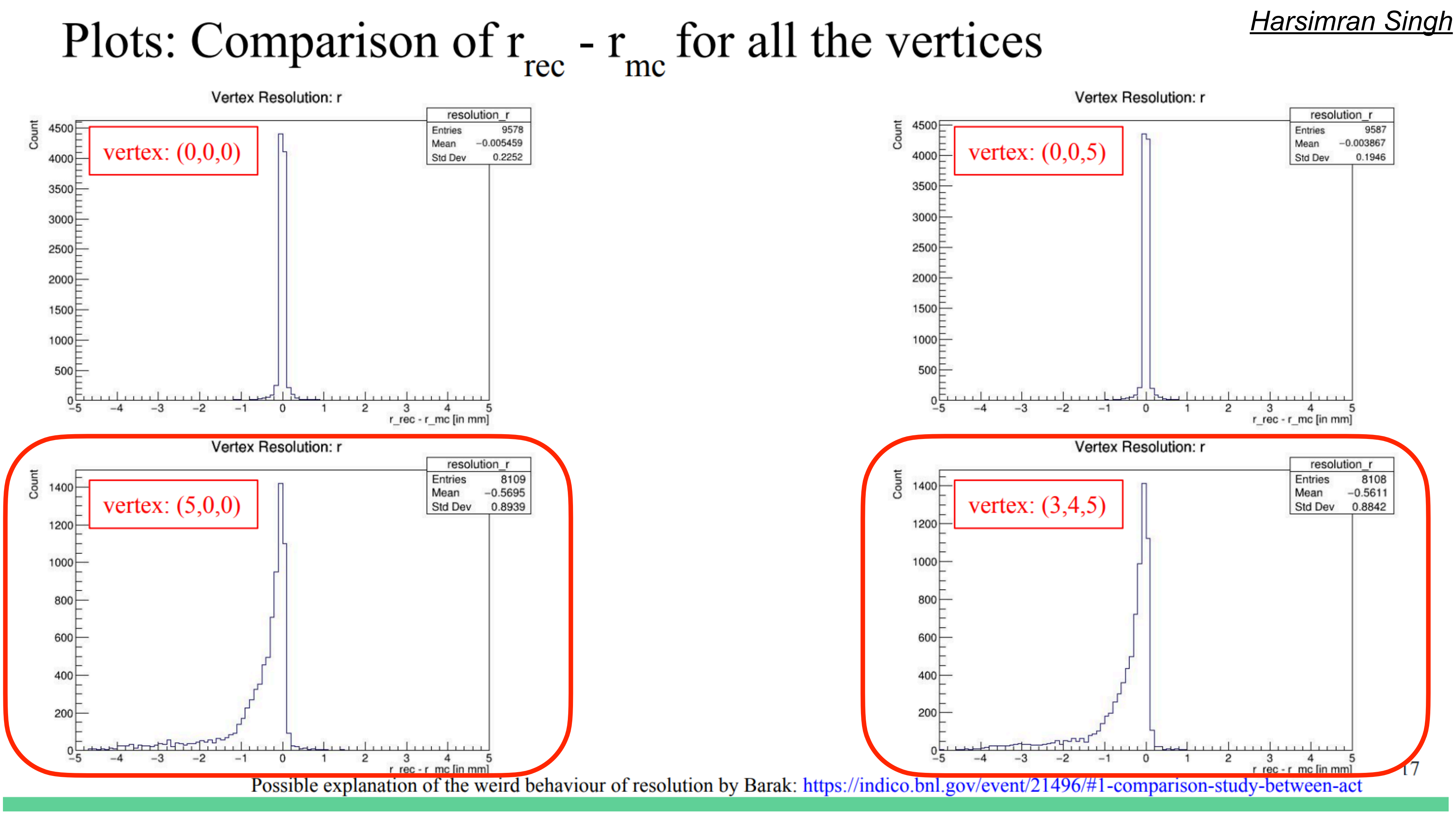
1. Gun Position: (0.0 0.0 0.0) // r = 0 mm, z = 0 mm (default)
2. Gun Position: (0.0 0.0 5.0) //r = 0 mm, z = 5 mm
3. Gun Position: (5.0 0.0 0.0) //r = 5 mm, z = 0 mm
4. Gun Position: (3.0 4.0 5.0) //r = 5 mm, z = 5 mm

r_{rec} : Reconstructed vertex position r (accessed as *CentralCKFTrackParameters.getLoc().a*)

r_{mc} : Truth(MC) vertex position r

z_{rec} : Reconstructed vertex position z (accessed as *CentralCKFTrackParameters.getLoc().b*)

z_{mc} : Truth(MC) vertex position z



Plots: Comparison of r_{rec} - r_{mc} for all the vertices

Harsimran Singh

Possible explanation of the weird behaviour of resolution by Barak: https://indico.bnl.gov/event/21496/#1-comparison-study-between-act

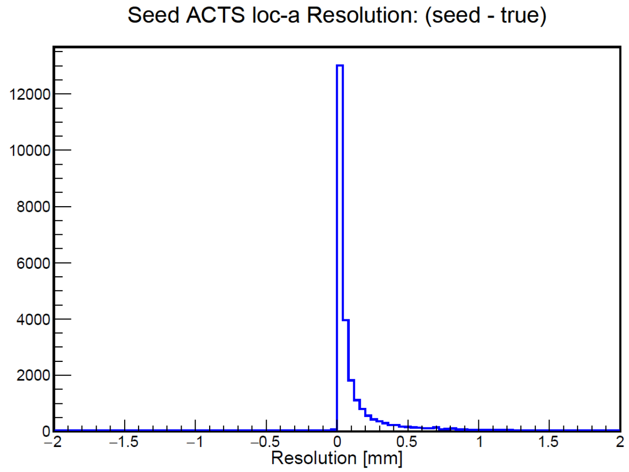
Δr distributions are distorted for collisions originated 5mm away from (0,0)

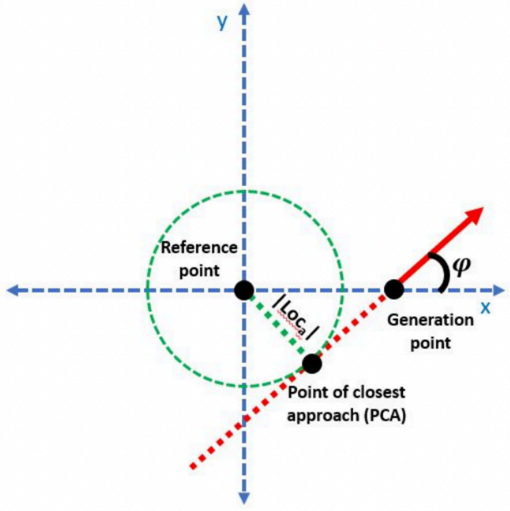
Possible causes: 1) Tracking issue for off-axis tracks in ACTS (EICrecon, PR#1185, merged)
2) Default TrackParameters calculated w.r.t (0,0)

PCA with PR#1185 Fix

ACTS issue in Loc_a value - always positive, addressed in PR#1185 (Barak S.)

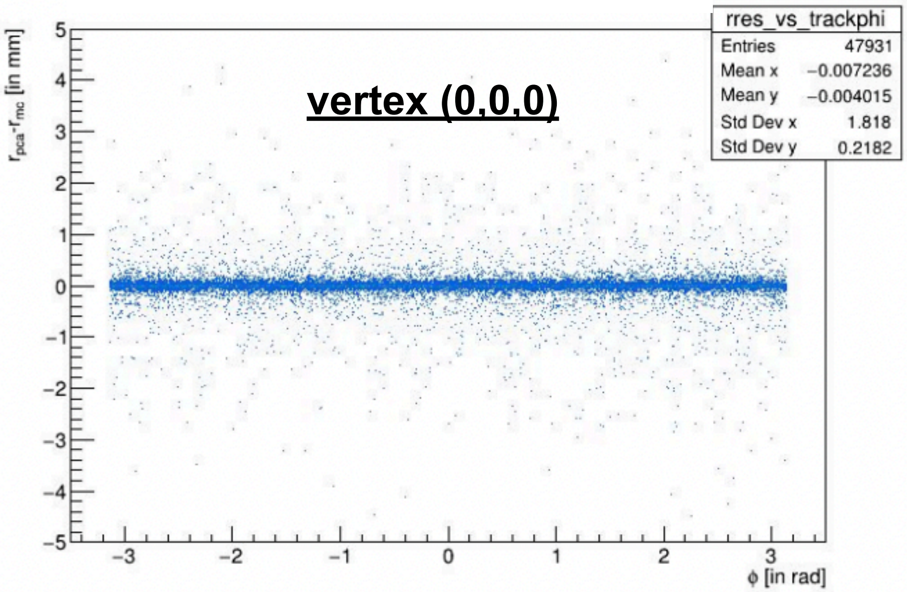
Barak S.

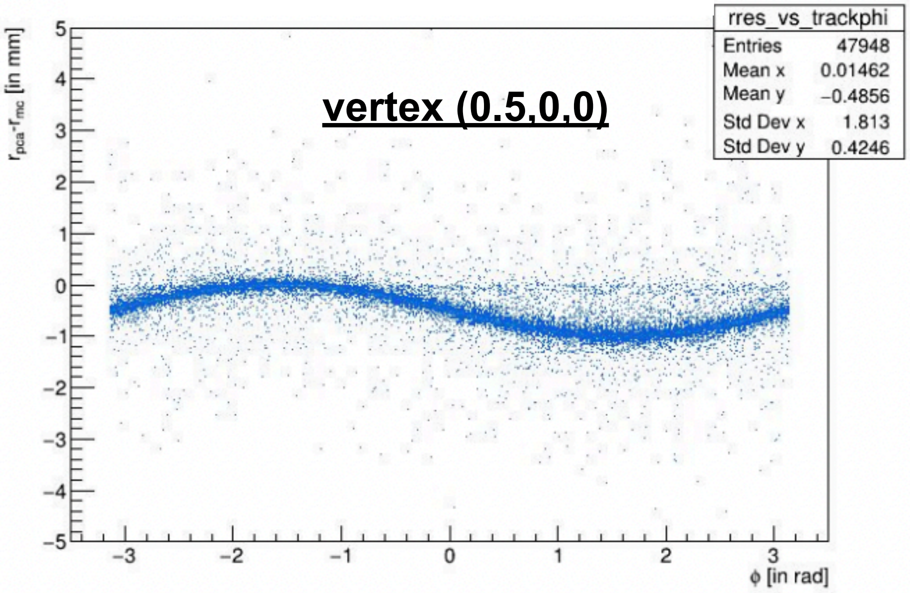




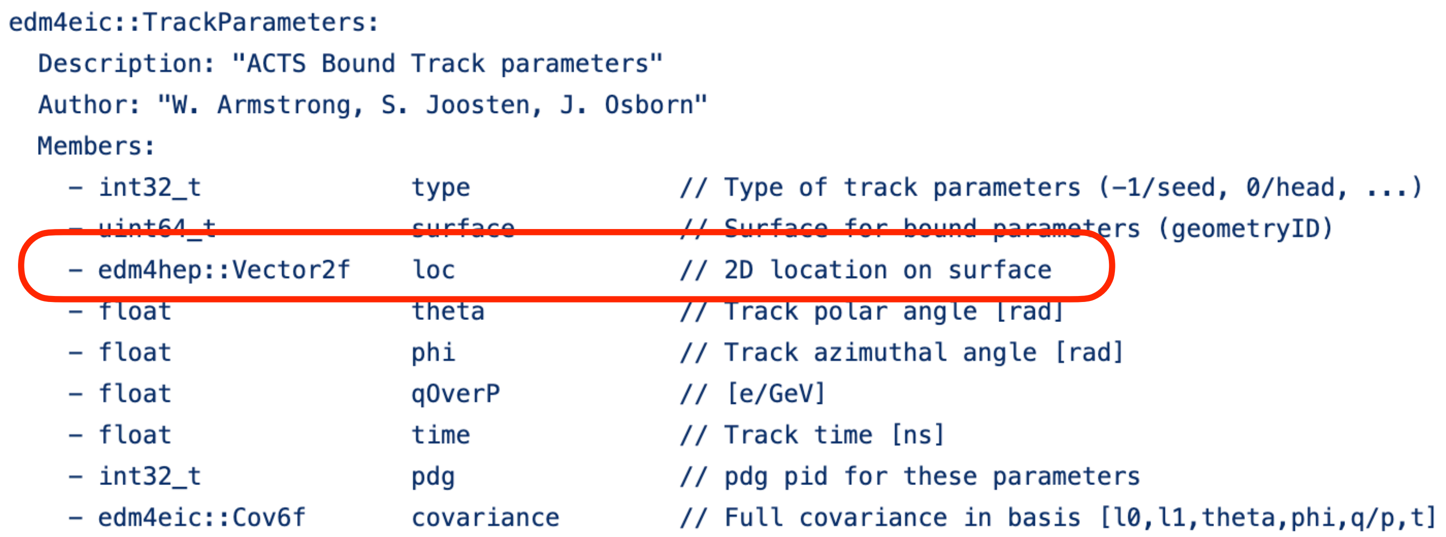
Harsimran used the ad hoc fix from Barak and recalculated the r_PCA

Harsimran Singh



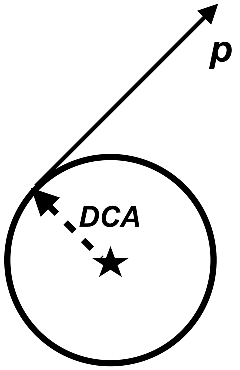


DCA to Vertex Calculation



```
edm4eic::TrackParameters:
  Description: "ACTS Bound Track parameters"
  Author: "W. Armstrong, S. Joosten, J. Osborn"
  Members:
    - int32_t              type             // Type of track parameters (-1/seed, 0/head, ...)
    - uint64_t             surface          // Surface for bound parameters (geometryID)
    - edm4hep::Vector2f    loc              // 2D location on surface
    - float                theta            // Track polar angle [rad]
    - float                phi              // Track azimuthal angle [rad]
    - float                qOverP           // [e/GeV]
    - float                time             // Track time [ns]
    - int32_t              pdg              // pdg pid for these parameters
    - edm4eic::Cov6f       covariance       // Full covariance in basis [l0,l1,theta,phi,q/p,t]
```

Vector2f: a - projection to (0,0) (2D signed dca to (0,0) line)
b - z component of the projection
direct information of the position phi information hidden



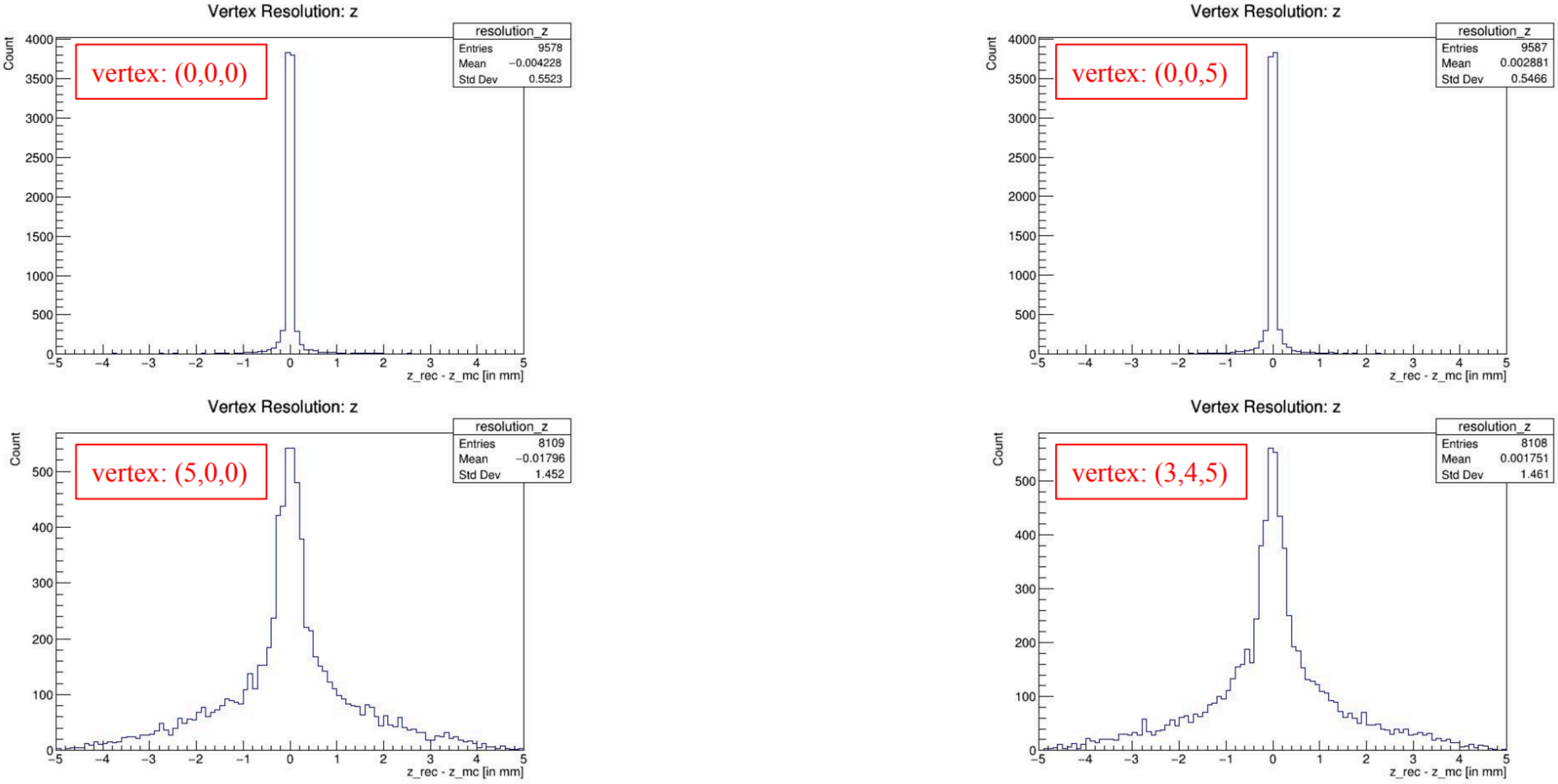
Input from Barak:
one can derive the position phi based on the momentum direction

$$\overrightarrow{DCA} \perp \vec{p}$$

Next: Reconstruct full track helices and re-calculate the DCA to off-axis vertices
- important to have a common tool to perform such calculations

Harsimran Singh

Plots: Comparison of z_{rec} - z_{mc} for all the vertices



18

Infrastructure Updates - edm4eic::Vertex

1) Current edm4eic::Vertex definition - several information missing

2) Discussed in the Track Recon group and then the S&C group

- discussions regarding a generic vertex structure for both Rec. and MC or Generated vertices

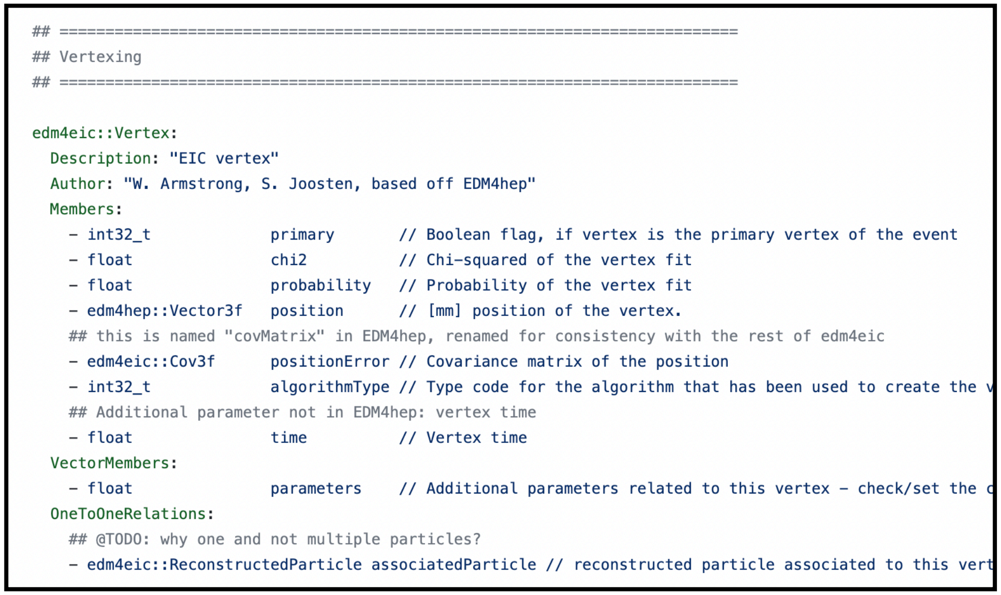
3) Joe and I discussed further on this, we prefer to separated structure for Rec and MC vertices. Proposal on changes for Rec Vertex in PR #61.

- also request to upstream edm4hep (PR #248 merged)

Reconstructed Vertex
Simulated Vertex:

- Generator level vertices (HepMC data)
- GEANT Vertices

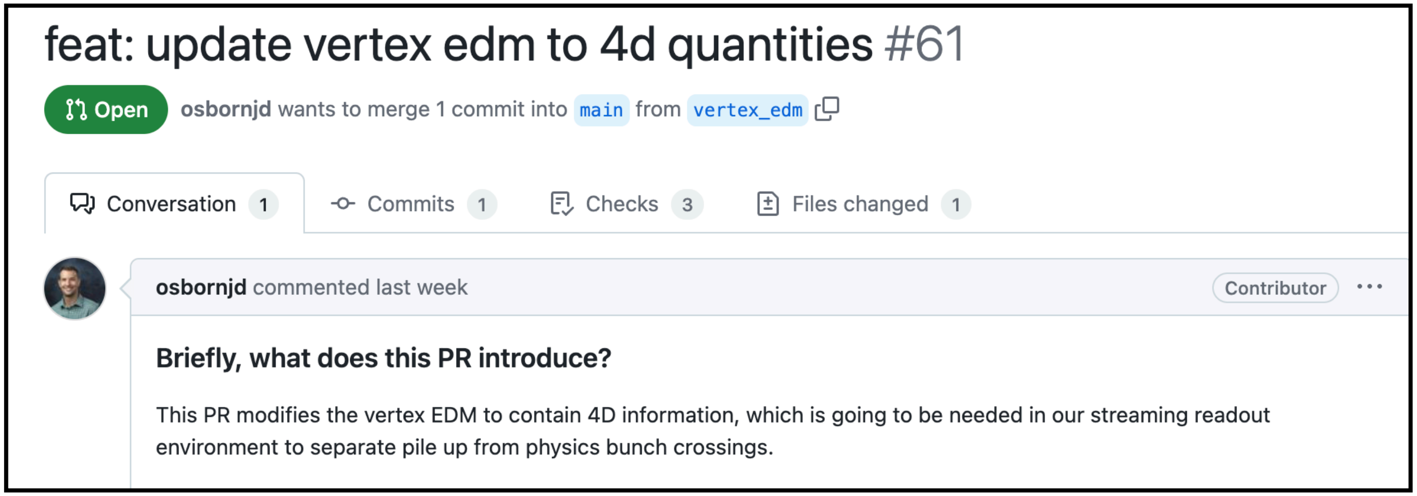
Current edm4eic::Vertex



```
## ========================================================================
## Vertexing
## ========================================================================

edm4eic::Vertex:
  Description: "EIC vertex"
  Author: "W. Armstrong, S. Joosten, based off EDM4hep"
  Members:
    - int32_t             primary       // Boolean flag, if vertex is the primary vertex of the event
    - float               chi2          // Chi-squared of the vertex fit
    - float               probability   // Probability of the vertex fit
    - edm4hep::Vector3f   position      // [mm] position of the vertex.
    ## this is named "covMatrix" in EDM4hep, renamed for consistency with the rest of edm4eic
    - edm4eic::Cov3f      positionError // Covariance matrix of the position
    - int32_t             algorithmType // Type code for the algorithm that has been used to create the v
    ## Additional parameter not in EDM4hep: vertex time
    - float               time          // Vertex time
  VectorMembers:
    - float               parameters    // Additional parameters related to this vertex - check/set the c
  OneToOneRelations:
    ## @TODO: why one and not multiple particles?
    - edm4eic::ReconstructedParticle associatedParticle // reconstructed particle associated to this vert
```

Joe Osborn



feat: update vertex edm to 4d quantities #61

osbornjd wants to merge 1 commit into main from vertex_edm

osbornjd commented last week

Briefly, what does this PR introduce?

This PR modifies the vertex EDM to contain 4D information, which is going to be needed in our streaming readout environment to separate pile up from physics bunch crossings.

Summary and Plans

Initial vertexing algorithm implemented:

Few-track simulation: starting vertex at (0,0,z0) - Show decent vertexing position resolution

PYTHIA DIS event: simulation campaign events with crossing angles and random vertex positions

- vertex resolution worse by ~ a factor of 5 or more

- limiting kinematic acceptance showing resolution better by ~ 2, but not enough

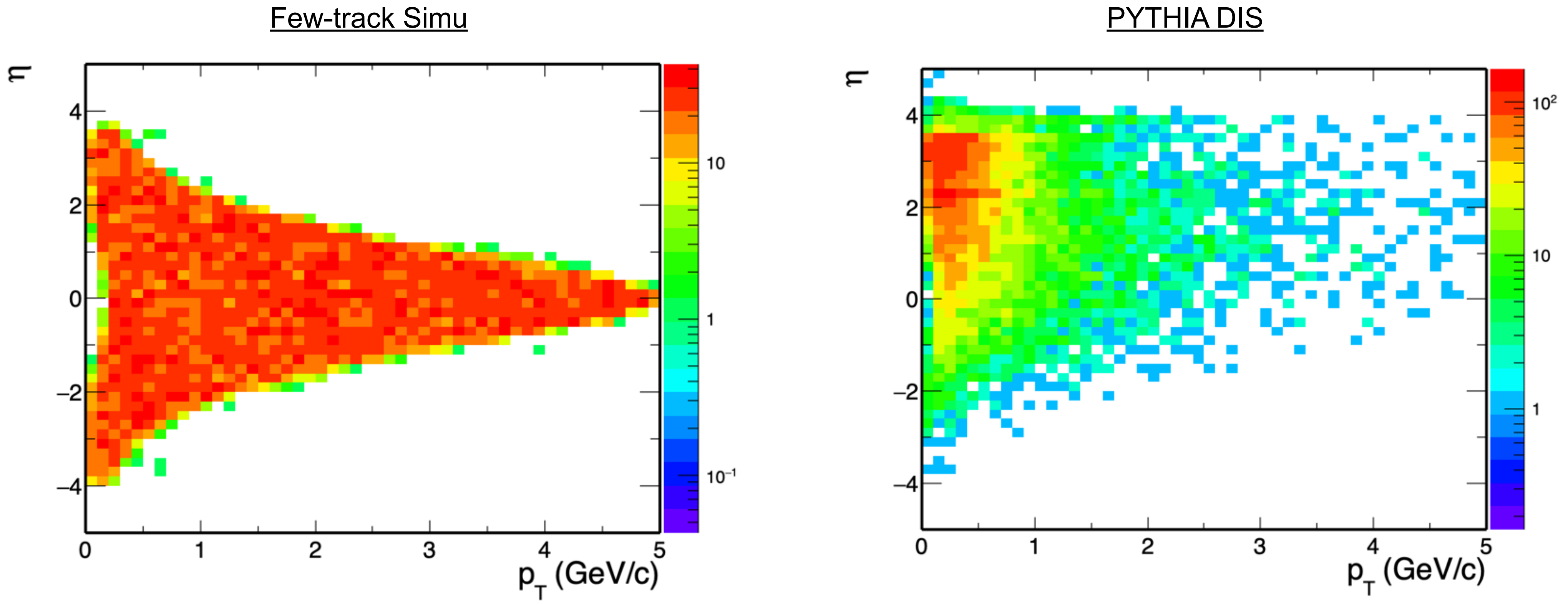
Near Term Plans:

1) Off-axis tracking performance under investigation (working together with Track Recon Team)
   - common DCA calculation tool needed
   - DCA performances to be investigated

2) DIS event performance test ongoing

3) Vertex Objects
   - edm4eic:vertex structure discussion going on, PR#61 under review
   - generated/GEANT vertices need some storage structure

Different Kinematic Distributions

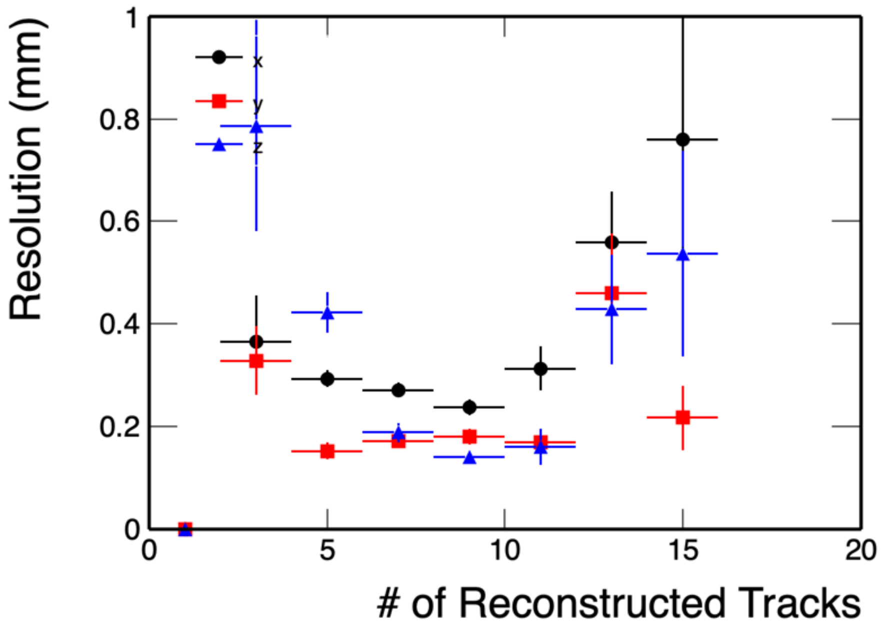


Possible Reason:

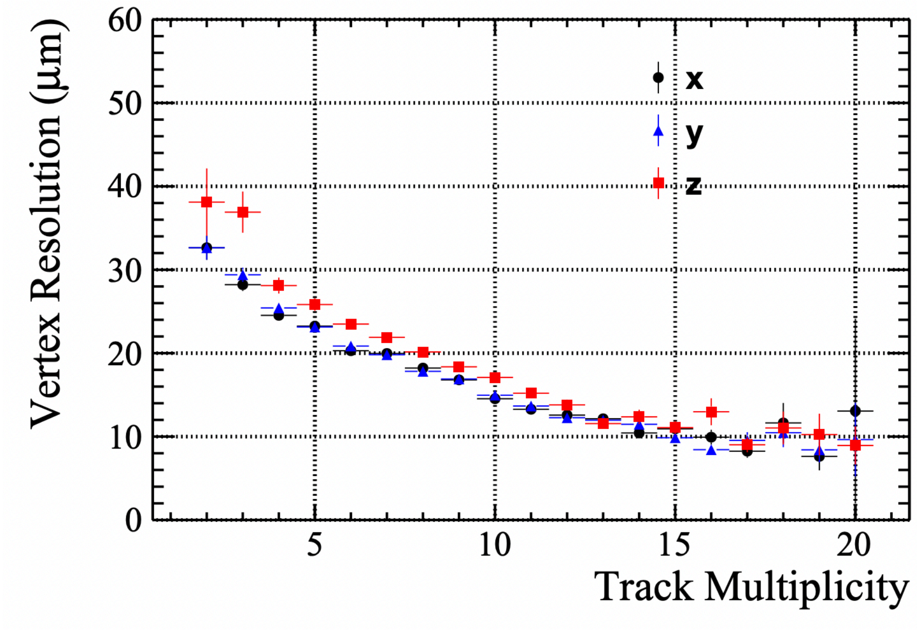
Tracks in DIS events more populated at forward eta (1<eta<3.5) and low pT

Vertex Resolution Discussions

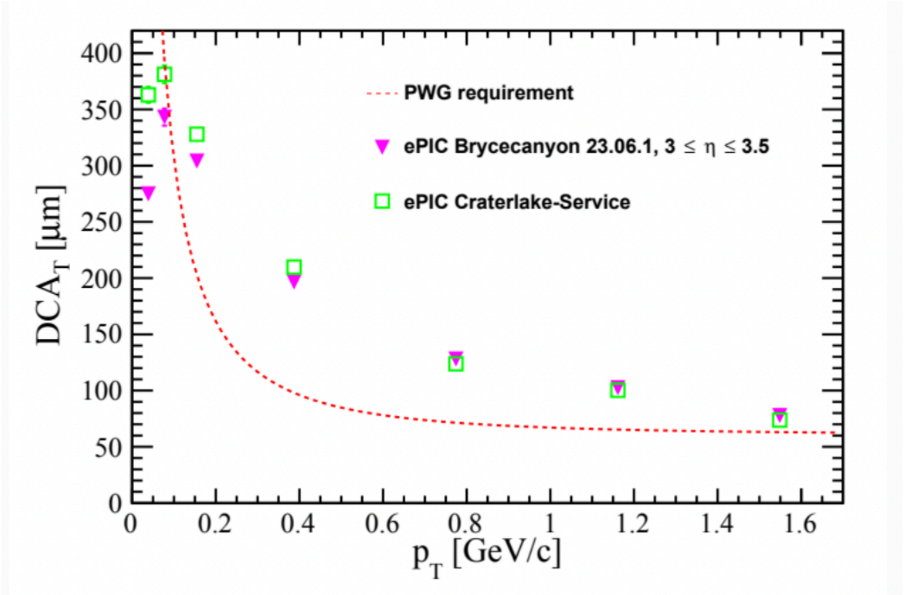
10x100 GeV GEANT + EICrecon



Fast-sim, DCA-avg, YR-parameter, 18x275, |eta|<3



Single track DCA resolution (Stephen M.) is worse than the PWG requirement in the forward/backward regions



Stephen Maple, July 13, TrkRecon Meeting