fsPHENIX Forward Tracking using PHGenFit and PHG4Hit

Jin Huang(BNL), Haiwang Yu (NMSU)

Jul 27, 2016

SpinFest2016 @ UC Riverside

Introduction

Motivation: Quickly produce fsPHENIX tracking performance with Geant simulation + Kalman Filter.

- More realistic than Sagitta calculation.
- Good estimation before the detector design finalized.
- Serves as prototype for forward sPHENIX tracking software
  - with future pattern recognition component.

Procedure:

- Simulation
  - $\Longrightarrow$ PHG4TruthInfoContainer.
  - $\Longrightarrow$ PHG4Hit.
  - $\Longrightarrow$ TGeo detector geometry in DST run node (PHGeometry by Jin)

- Feed the Kalman Filter:
  - PHG4Hit $\Longrightarrow$ Measurements: Smear PHG4Hit according to given detector resolution.
  - Measurements grouping: use MC truth information, "pseudo pattern recognition"
  - Seed: Smeared MC truth information

Outline

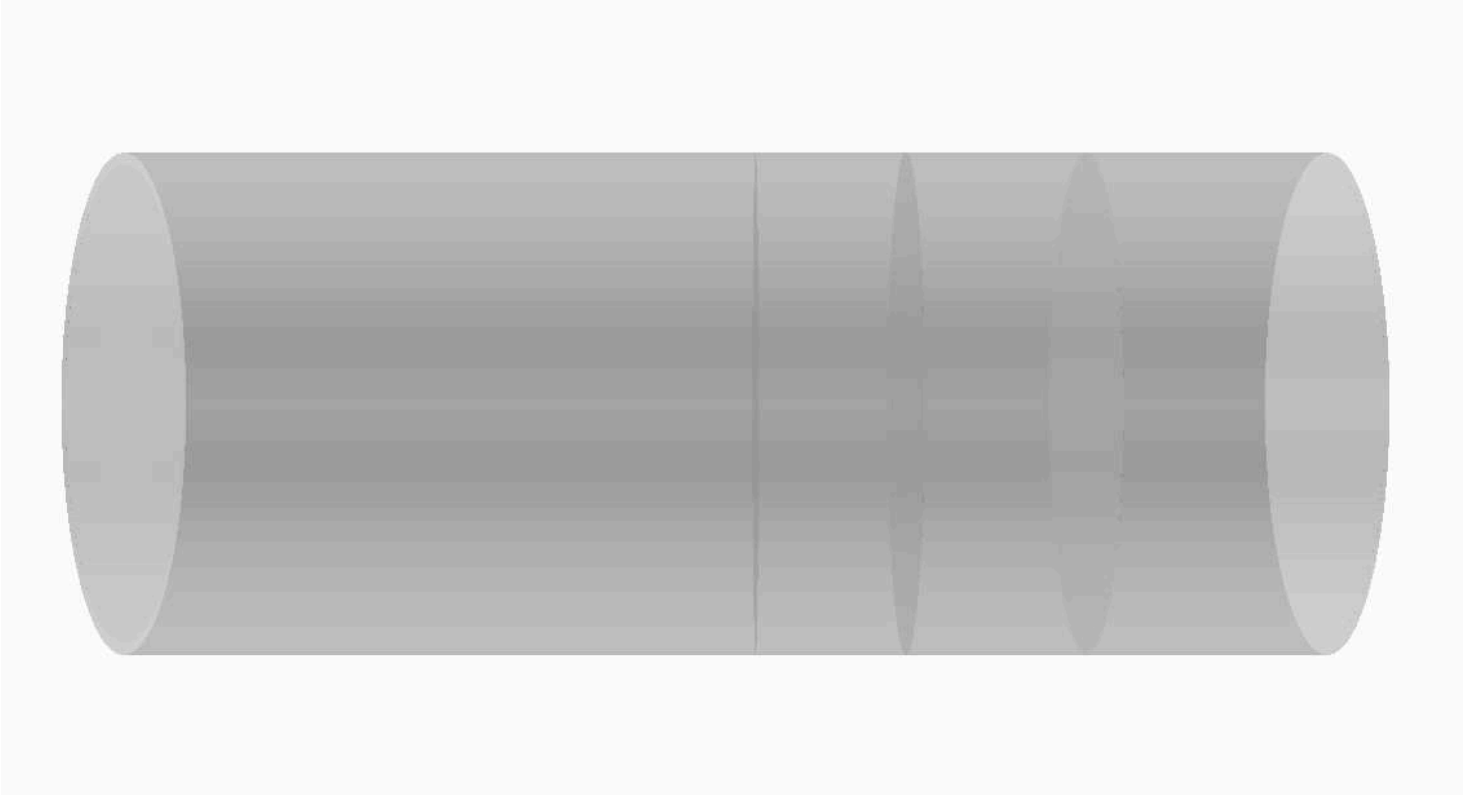
- Method Verification: test with 3-layer vacuum detector.
- Test using EIC concept FGEM, arXiv:1402.1209
- Companion with previous studies.
- Implementation of a forward tracking module based on this idea.

Method Verification:
Using 3-layer vertical plane tracking

3-layer vertical plane testing setup

First, we tested the code using 3-layer vertical planes:

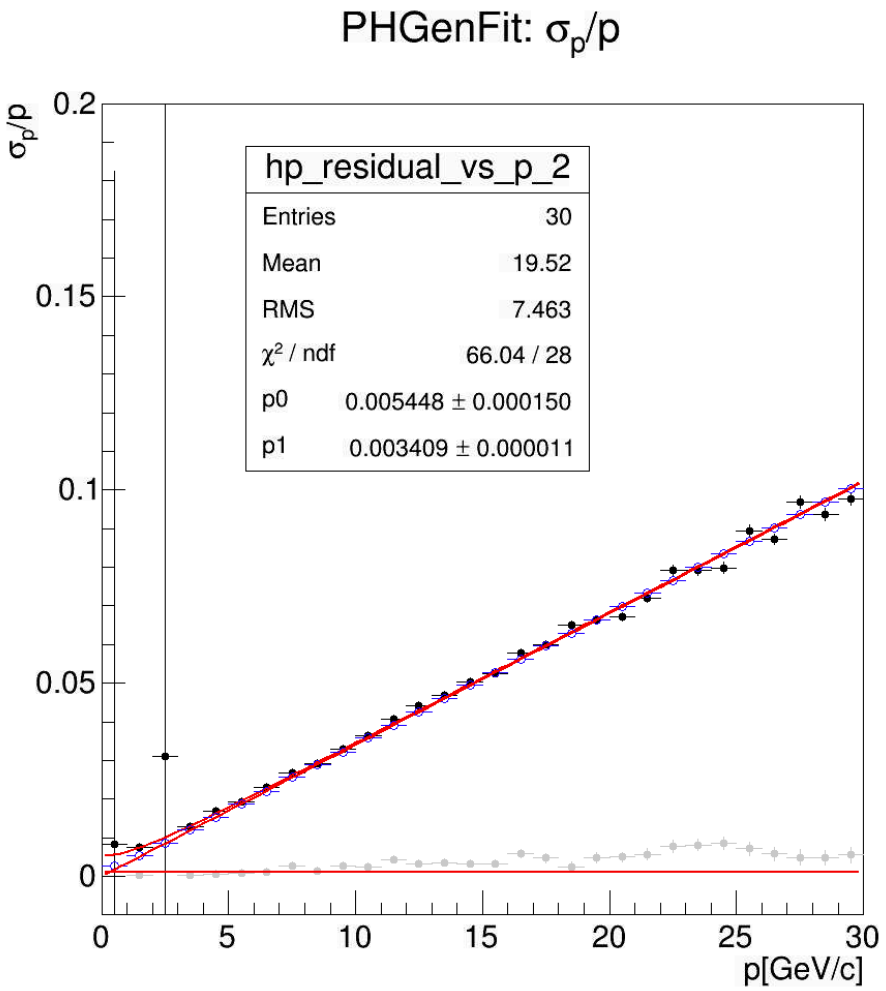
- The world and all detector material is G4_Galactic.
- 3-layer vertical planes with black holes outside.
- δr = 1cm
- r$\delta\phi$ = 100 μm



Vacuum, 3-layer, η = 3.0, r$\delta\phi$ = 100 μm, δr = 1cm

PHGenFit results are very similar with the Geant4 Sagitta calculation:

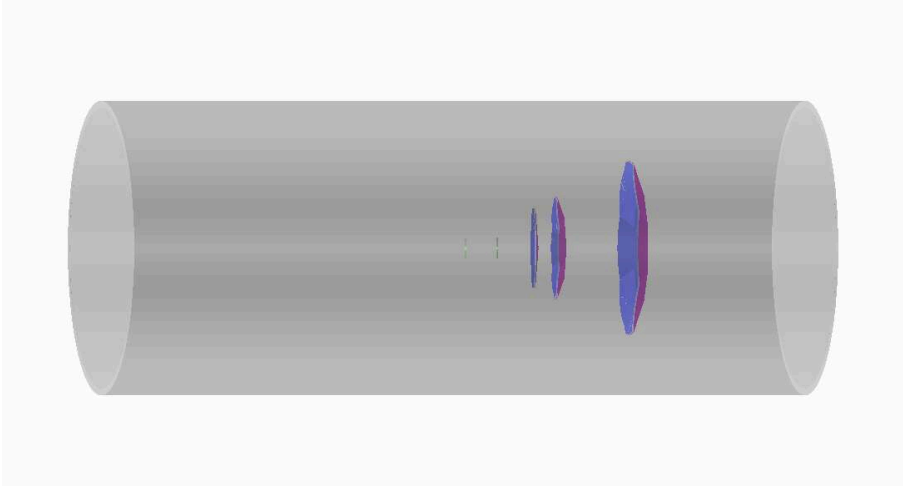
- Blue circle: $\sigma_{Sagitta}$/Sagitta vs. p from Geant4 simulation
- Black dot: σ_p/p vs. p from PHGenFit Kalman fitting.

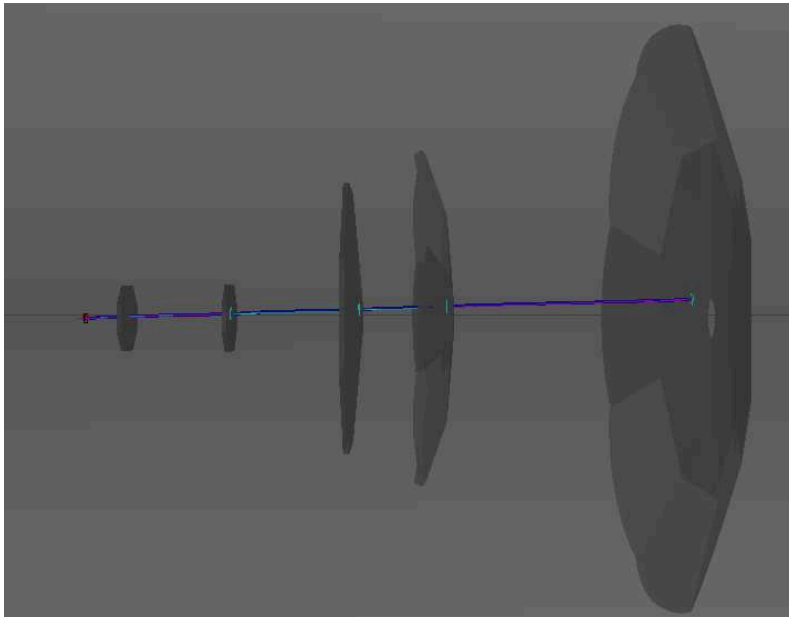


EIC Concept FGEM tracker: Momentum resolution σ_p/p

5(4)-station Forward GEM Detector

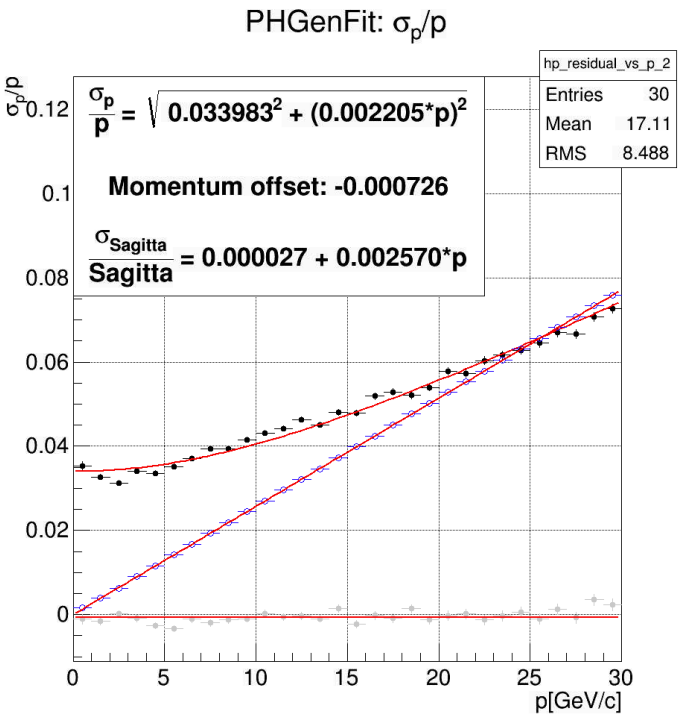
- Detector setup using G4_FGEM_fsPHENIX.C in sPHENIX master branch.
- z = 17 cm (1.01 < η < 2.7), 62 cm (2.15 < η <4.0), 120, 160, 275 (1.45 < η < 4) cm
- δr = 1 cm, r$\delta\phi$ = 100 μm (η = 1.5~2.5) 50 μm (η = 2.5~4)
- sPHENIX field, no passive piston.
- **Tracking used vertex** from smeared truth vertex (0,0,0) and 50 μm resolution.

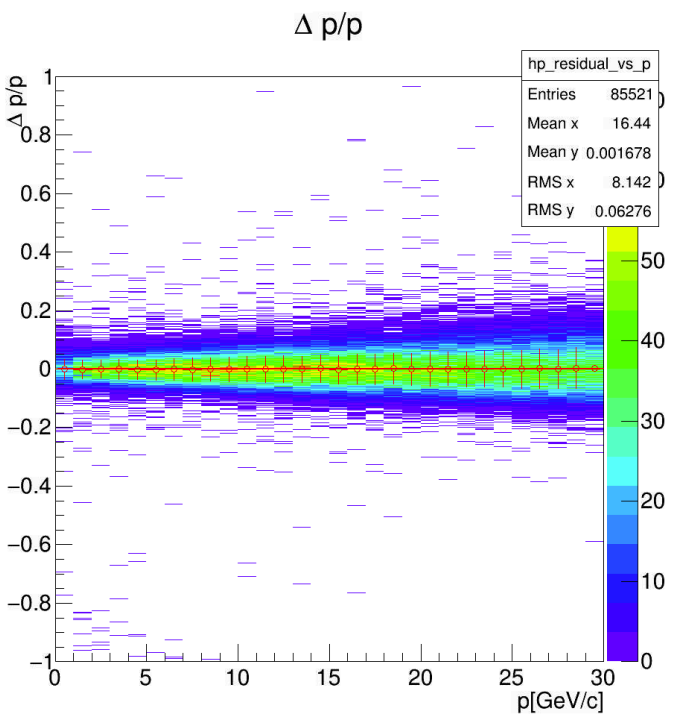




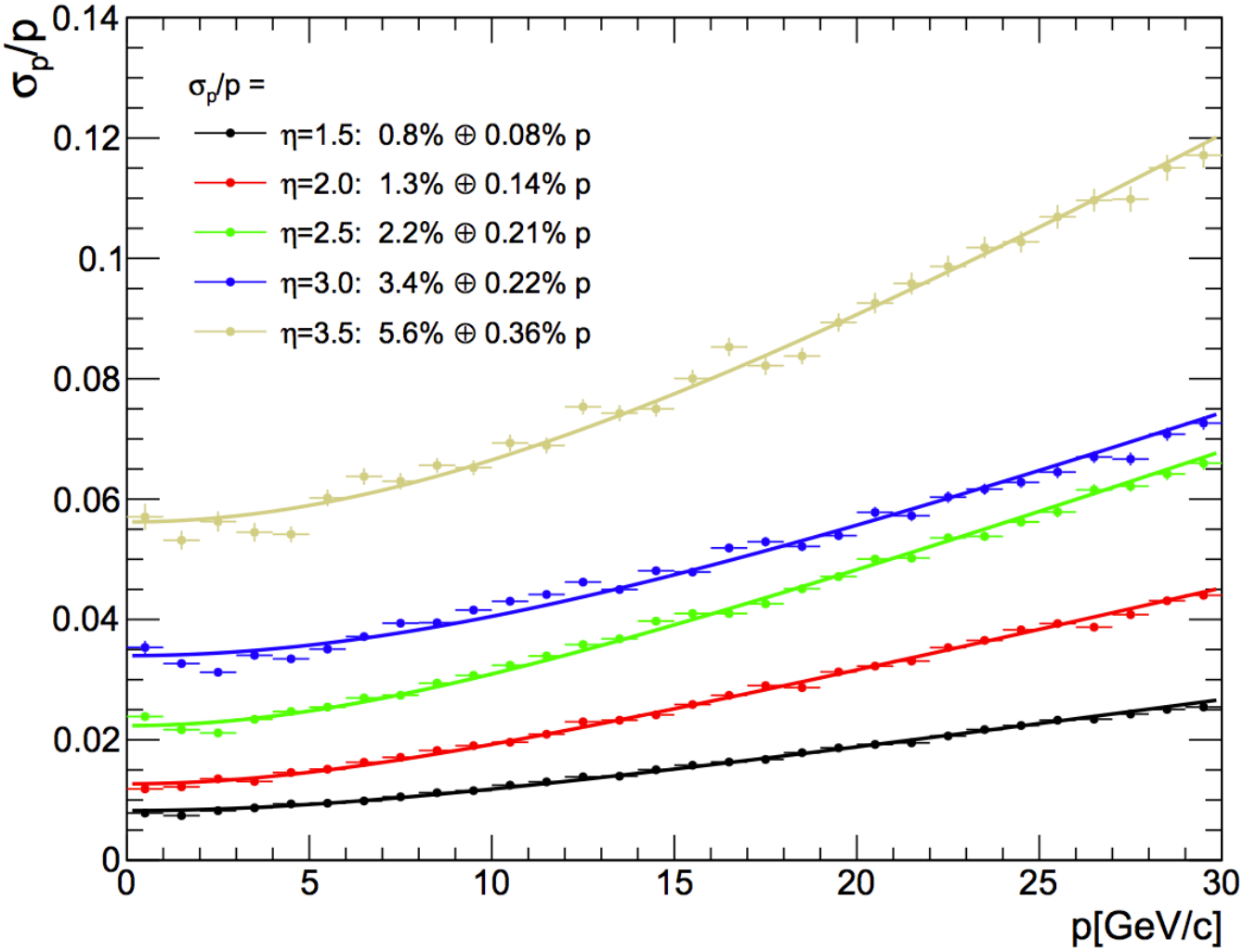
how to calculate σ_p/p

- 2D histo: $(p_{Reco} - p_{True})/p_{True}$ vs p_{True}, (right plot)
- For each slice of p_{True}, fit with Gaussian, extract mean as offset (Grey dots), sigma as momentum resolution (σ_p: Black dots), left plot.
- We also calculated $<\sigma_{Sagitta}/Sagitta>$ as a reference (Blue circles): Sagitta is calculated using vertex, station at 120cm and station at 275 cm.





Result: σ_p/p for different pseudo-rapidity

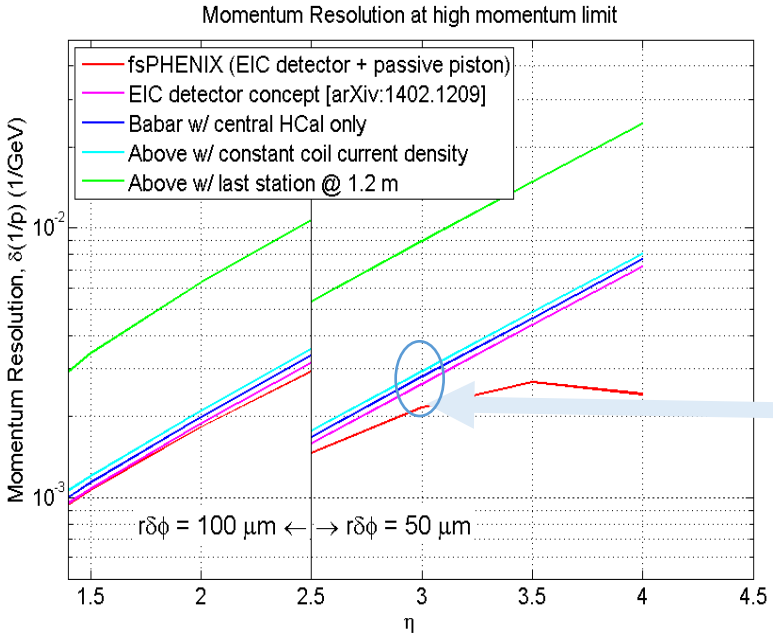


Details for each curve are in the backups.
PDF format: https://www.phenix.bnl.gov/WWW/p/draft/yuhw/fsPHENIX/FGEM_PHGenFit.pdf

Compare with previous studies

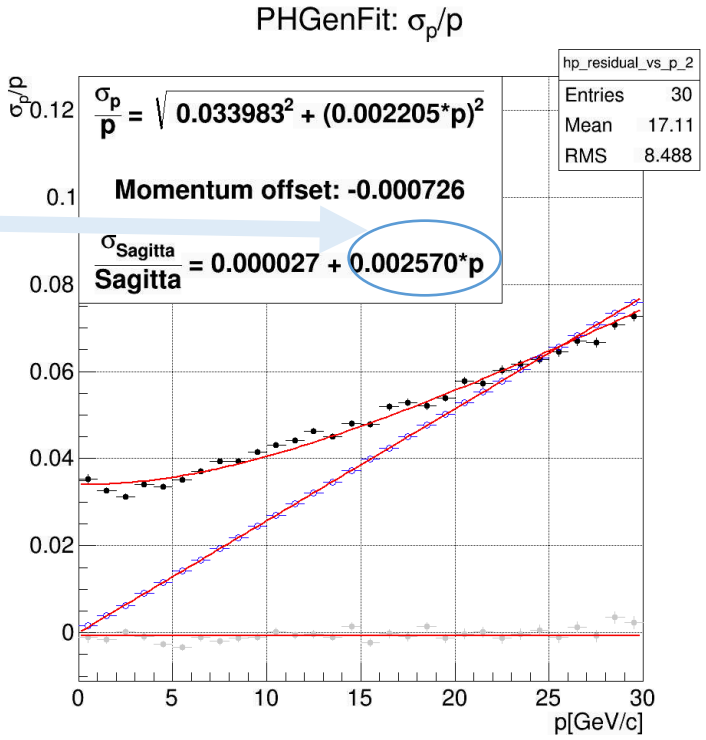
Compare with previous studies, Jin's MatLab calculation

Jin's calculation based on vertex + optimum Sagitta plane + 300cm last station.



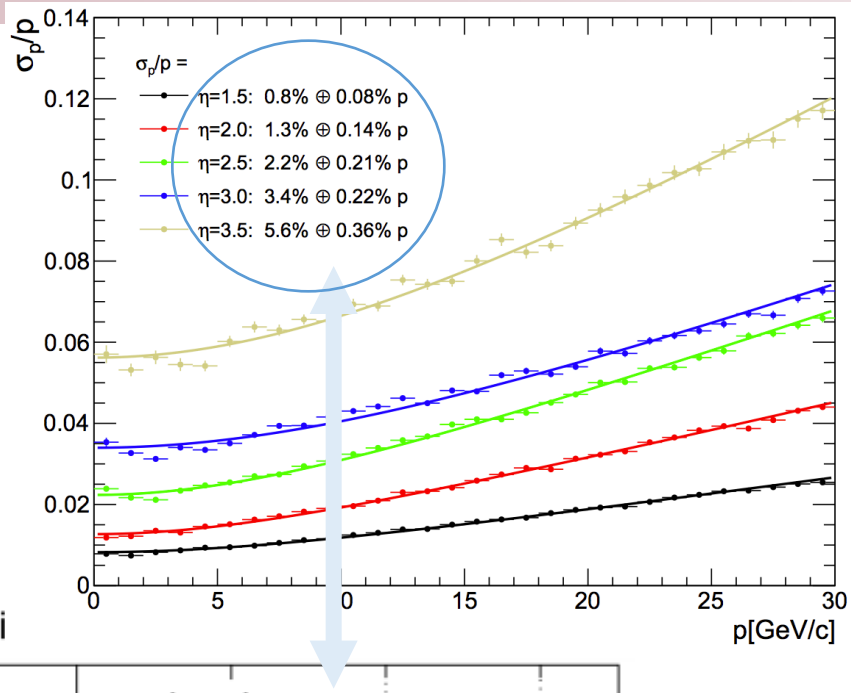
GenFit fitting for η = 3.0, corresponding to magenta curve in left plot.

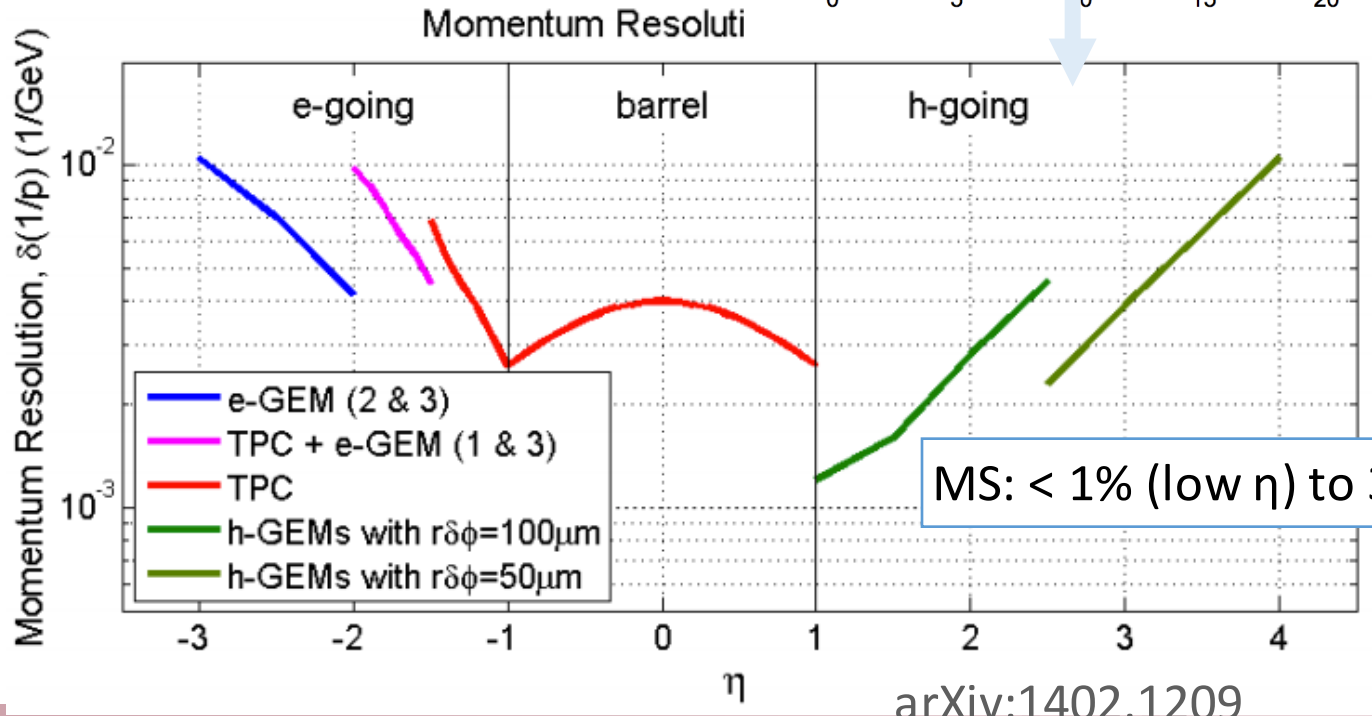
- The linear term, p1, from the σ_S/S **is consistent with left plot,** both are ~0.25%.
- The p1 term from full GenFit Kalman is better than σ_S/S . That could be caused by that we have more stations in full Kalman.



Compare with previous studies, arXiv:1402.1209

- Linear term p1 is better than previous Sagitta tracking study by **using a vertex in the fitting**.
- Constant term p0, (Multiple Scattering Effect) is constant with previous Sagitta tracking study.





MS: < 1% (low η) to 3% (η = 3)

arXiv:1402.1209

A forward tracking module based on this idea: ForwardTracking

ForwardTracking module

- In sPHENIX git - analysis: analysis/ForwardTracking

- Measurements: PHG4Hits + smearing
  - Smear PHG4Hits according to given detector resolution
  - No detector digitization and cellularization.

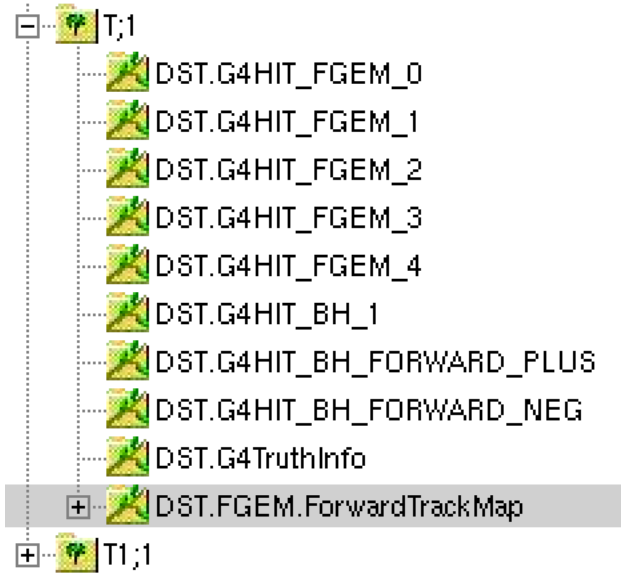
- Pseudo pattern recognition using Monte Carlo truth information
  - "track_id" matching between PHG4Hits and PHG4Particle in PHG4TruthInfoContainer
  - With manually input hit finding efficiency and noisy finding probability

- The TGeo geometry required by GenFit is provide by Jin's PHG4Geometry machineries.

- Output node: a SvtxTrackMap "FGEM/ForwardTrackMap"

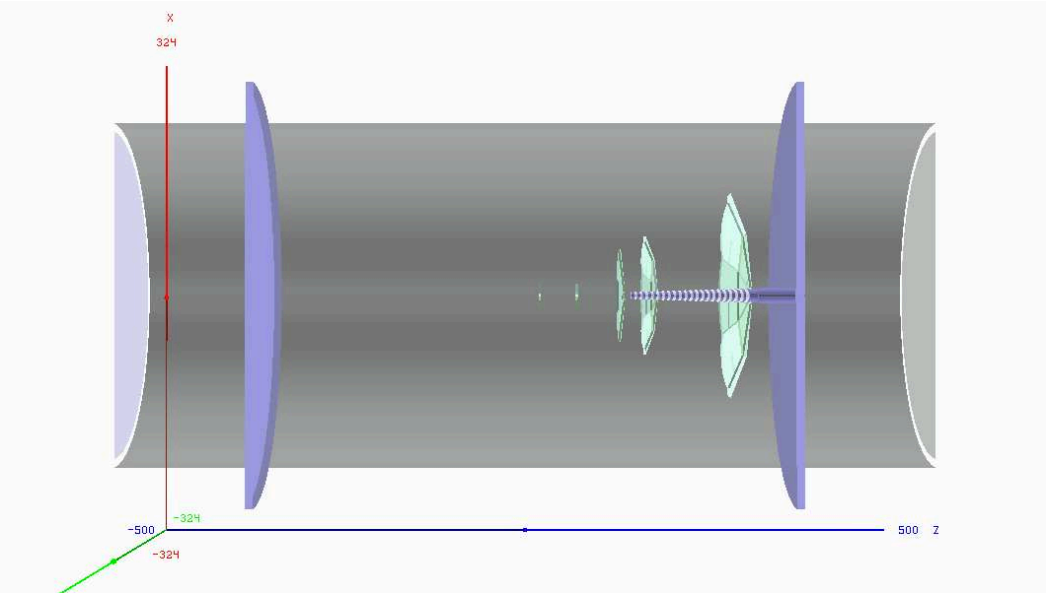
Using this module

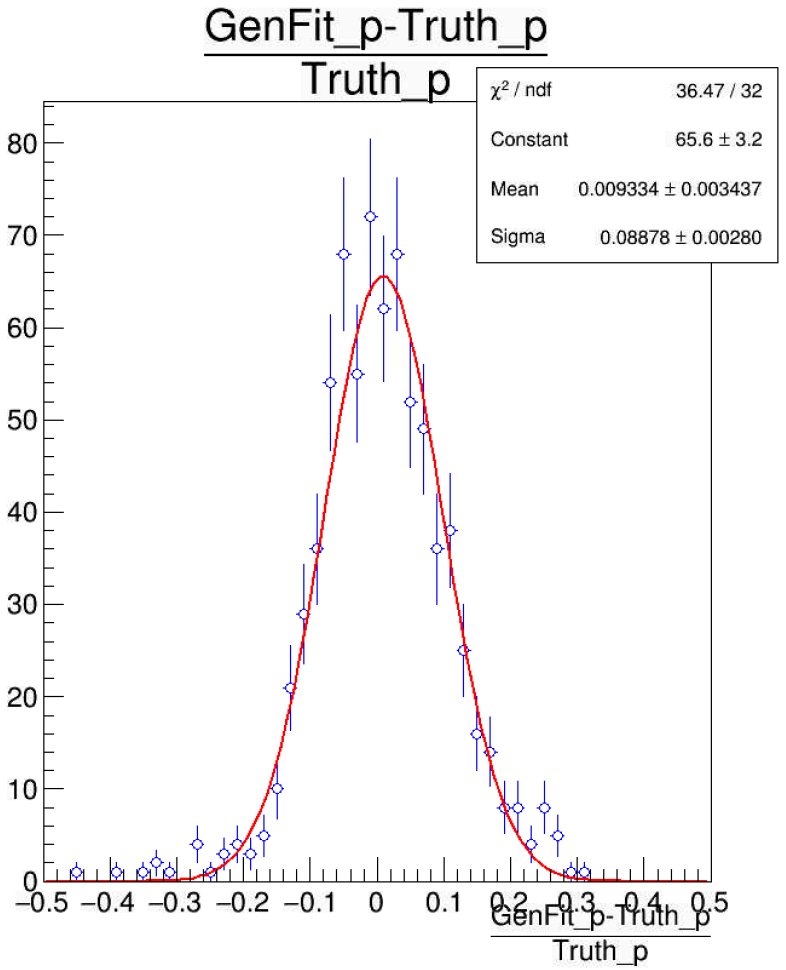
```
void FGEM_Reco(int verbosity = 0) {

  //---------------
  // Load libraries
  //---------------

  gSystem->Load("libfun4all.so");
  gSystem->Load("libForwardTracking.so");

  //---------------
  // Fun4All server
  //---------------

  Fun4AllServer *se = Fun4AllServer::instance();

  PHG4HitKalmanFitter* kalman = new PHG4HitKalmanFitter("PHG4HitKalmanFitter");

  kalman->set_mag_field_file_name("/phenix/upgrades/decadal/fieldmaps/fsPHENIX.2d.root");
  kalman->set_FGEM_phi_resolution(50E-4);
  kalman->set_FGEM_r_resolution(1.);
  kalman->set_pat_rec_hit_finding_eff(1.);
  kalman->set_pat_rec_nosise_prob(0.);
  kalman->set_do_evt_display(false);

  se->registerSubsystem(kalman);
}
```



First look at the output

TGeo used by GenFit in the DST RUN node





Summary

- The idea of GenFit Kalman Filter with smeared G4 PHG4Hits tested. Results consistent with previous studies.
- An forward tracking module implemented based on this idea. Under testing.

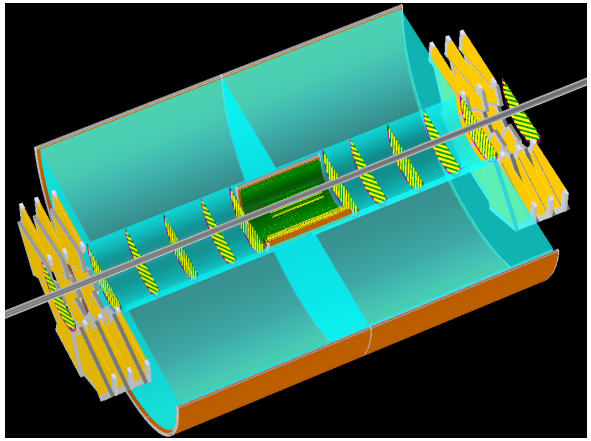
Next:

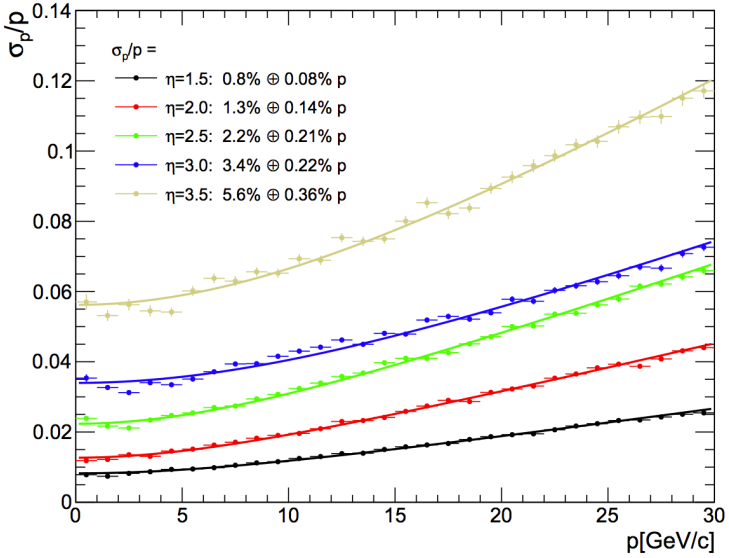
- Test and make sure every thing is working as expected.
- Output some evaluation NTuples.
- Simulations with physics motivations!

Backups:

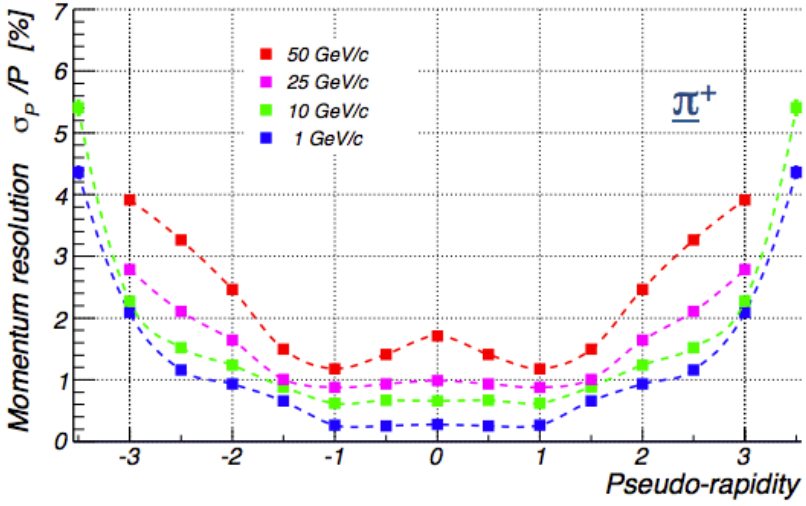
Compare with BeAST:

- Resolution is about x2 of the BeAST, despite BeAST use x2 stronger magnetic field + silicon tracker, as we used a much longer tracking arm.
- Our higher momentum resolution could be improved by also switching GEM to high precision silicon tracker.



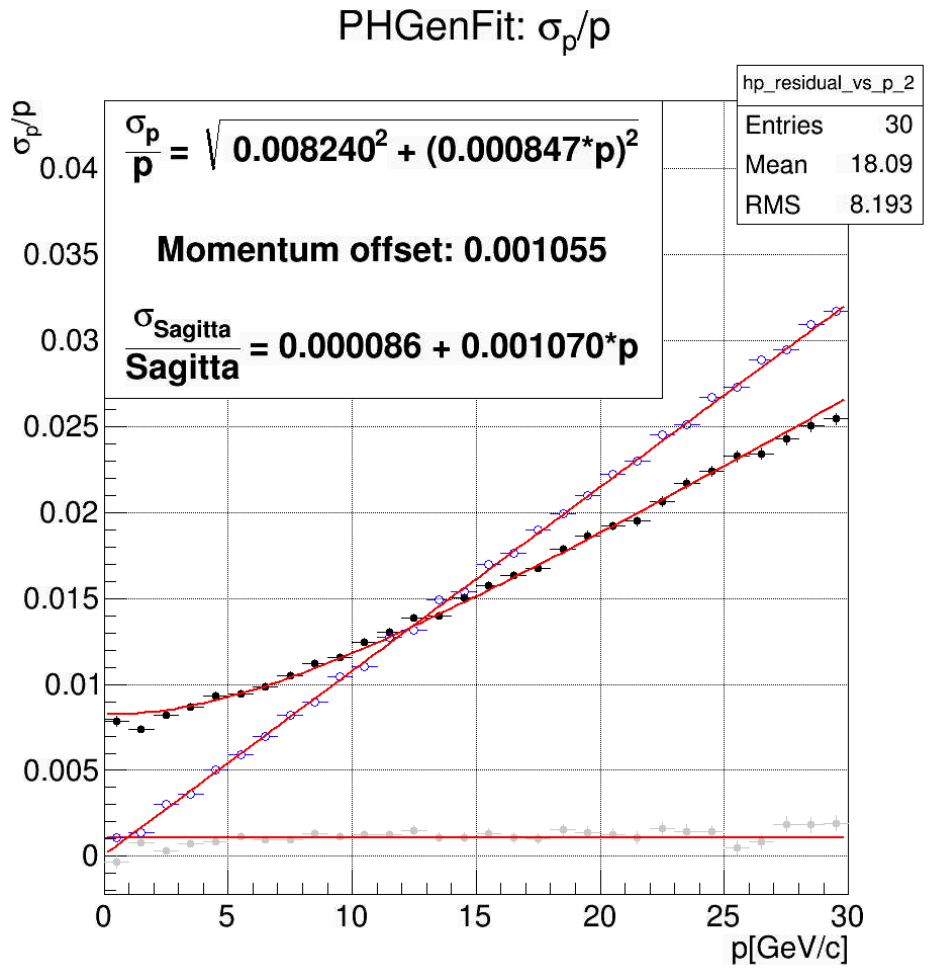


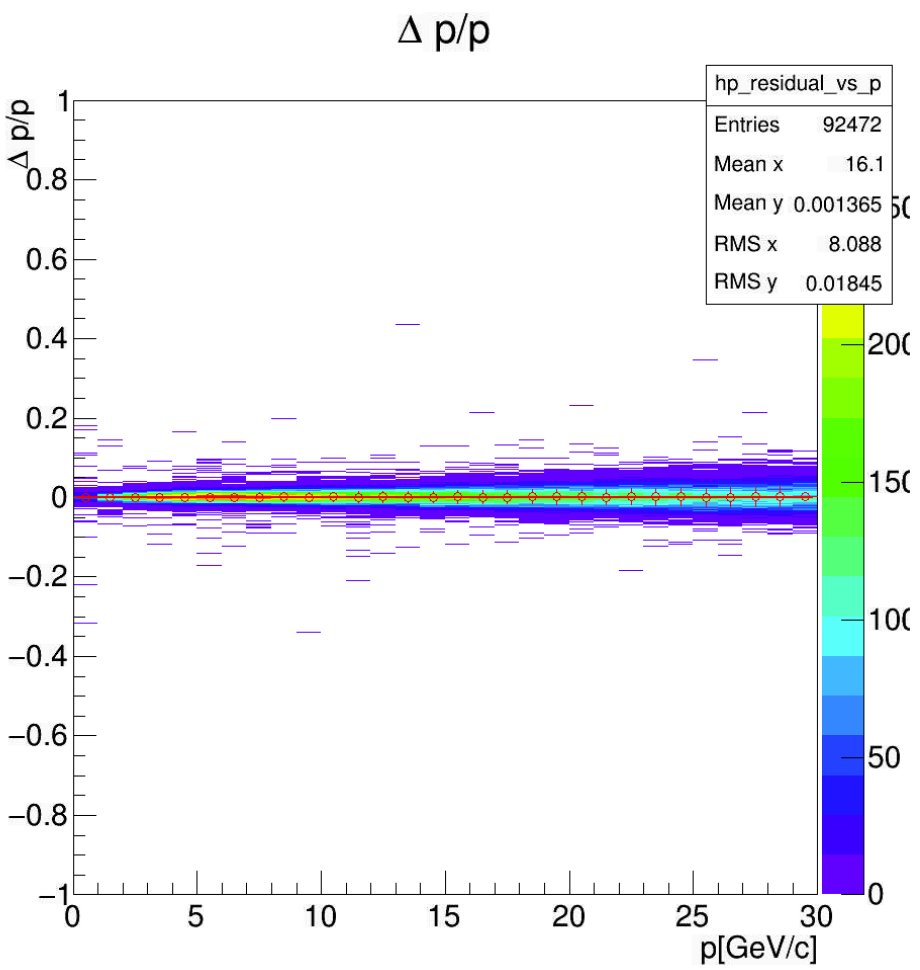
sPHENIX FGEM



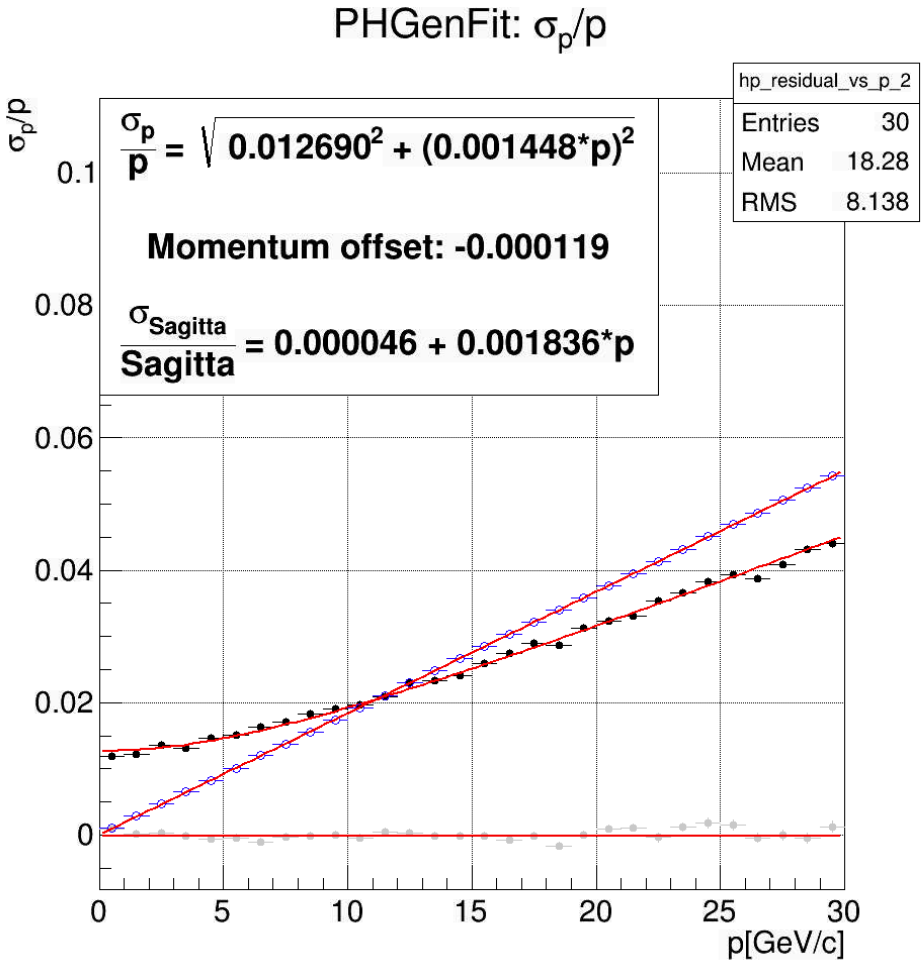
BeAST

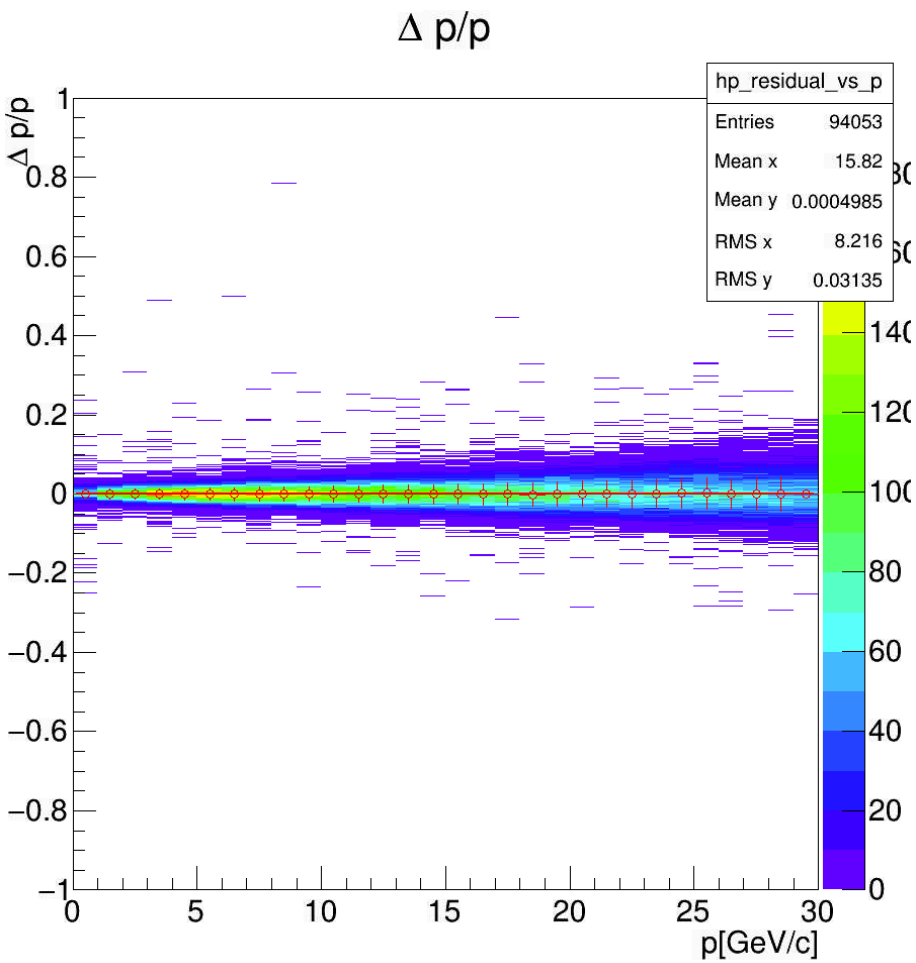
η = 1.5, r$\delta\phi$ = 100 μm



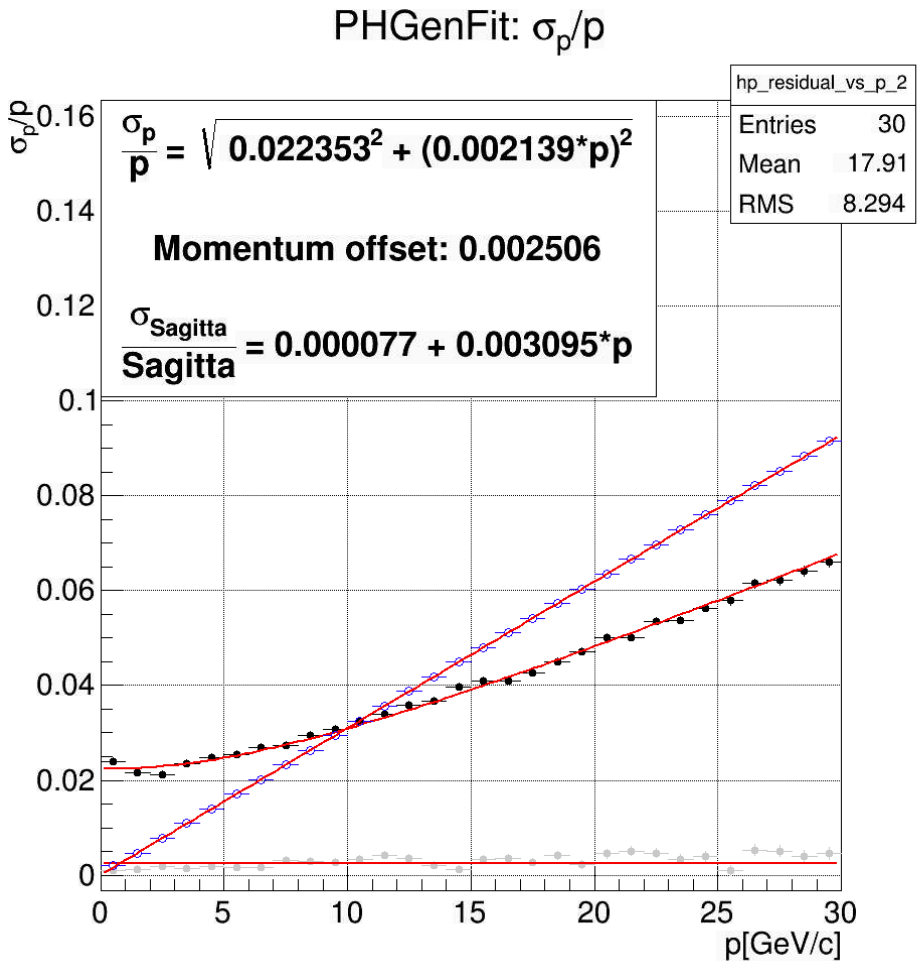


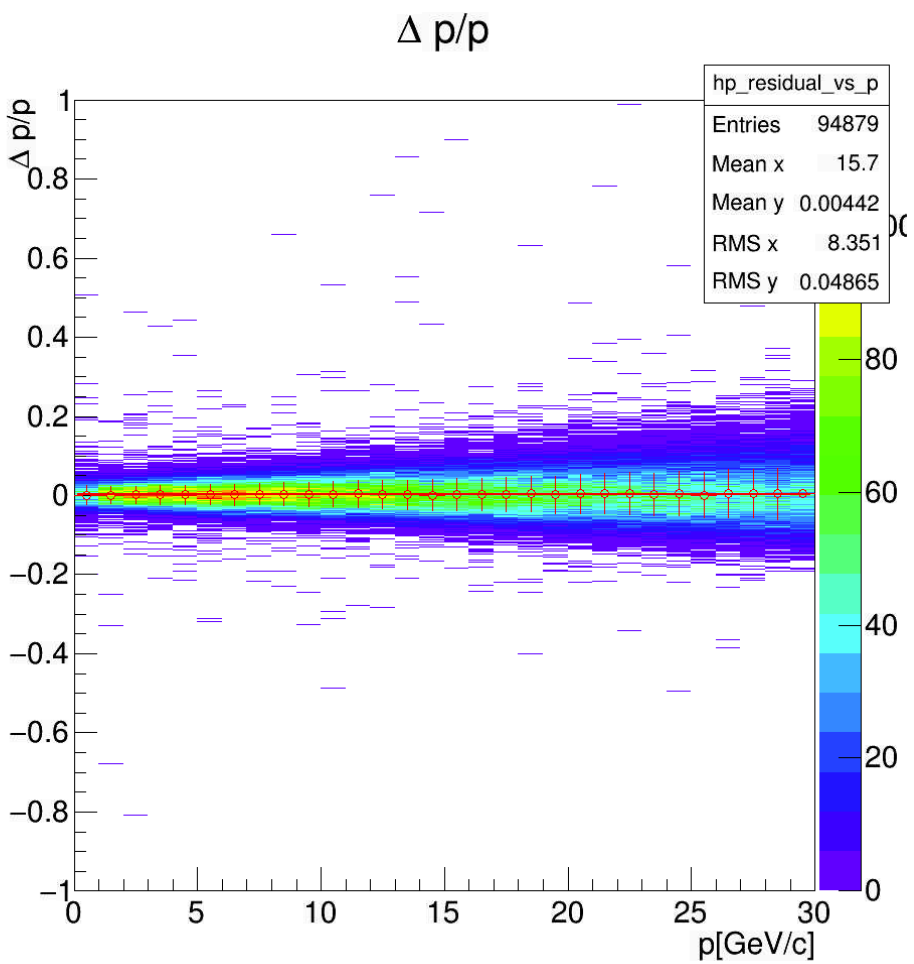
η = 2.0, r$\delta\phi$ = 100 μm



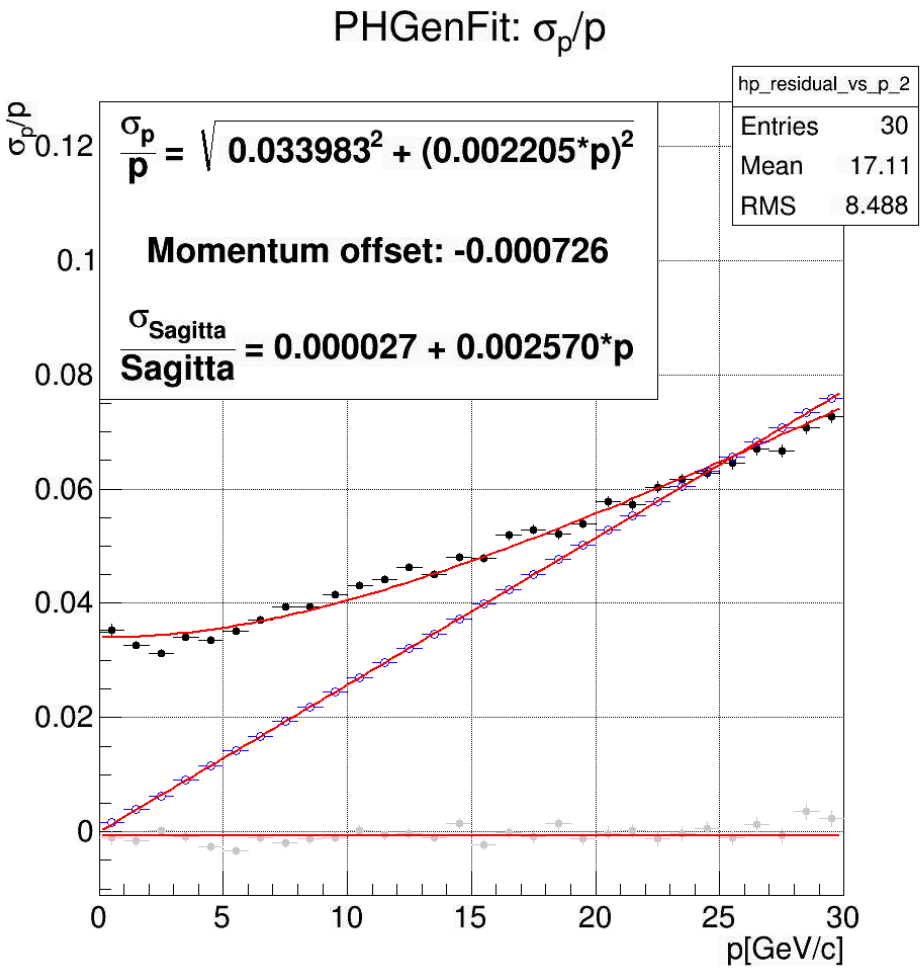


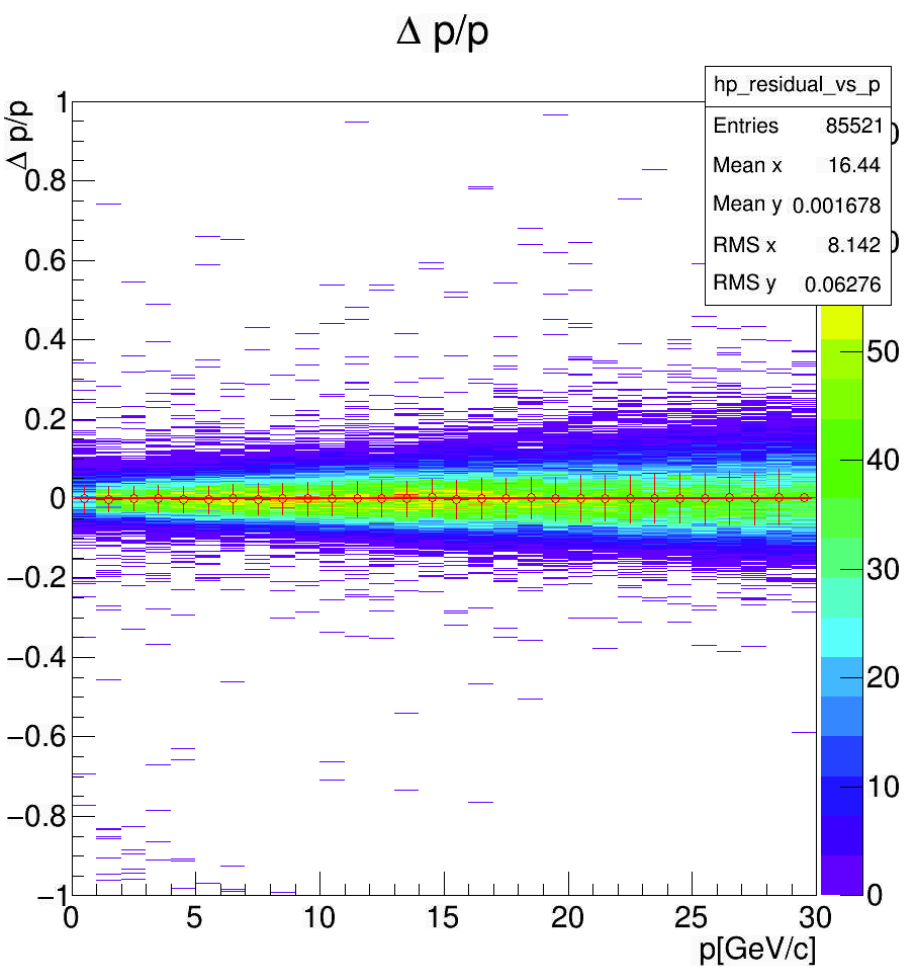
η = 2.5, r$\delta\phi$ = 100 μm





η = 3.0, r$\delta\phi$ = 50 μm





η = 3.5, r$\delta\phi$ = 50 μm