SoLID Tracking Reconstruction

Weizhi Xiong

Duke University

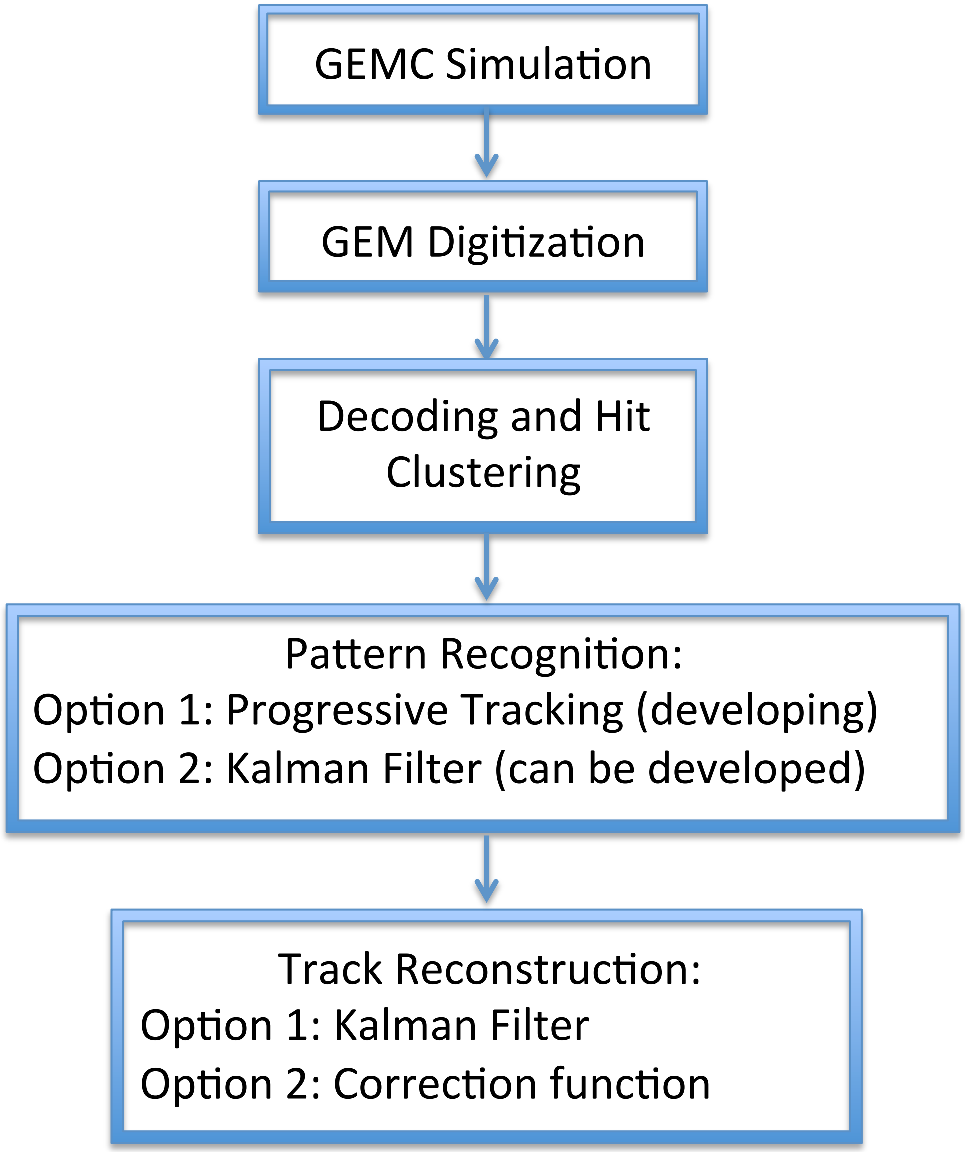
SoLID Collaboration Meeting

September 11-12, 2015

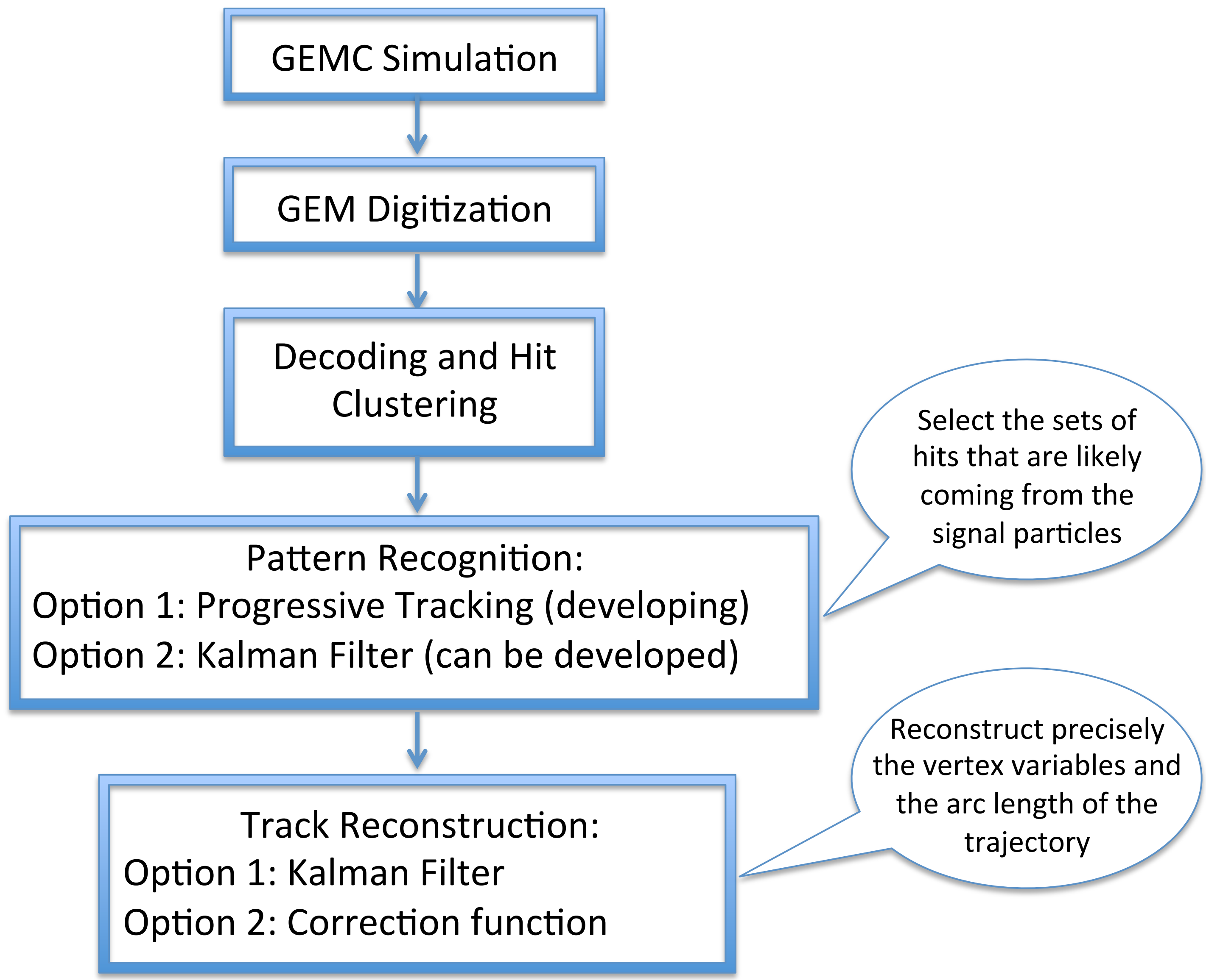
Outline

- Tracking framework overview
- Introduction to Kalman filter
- Details of the track reconstruction program
- Preliminary Tracking reconstruction results using Kalman filter
- Conclusion and plan

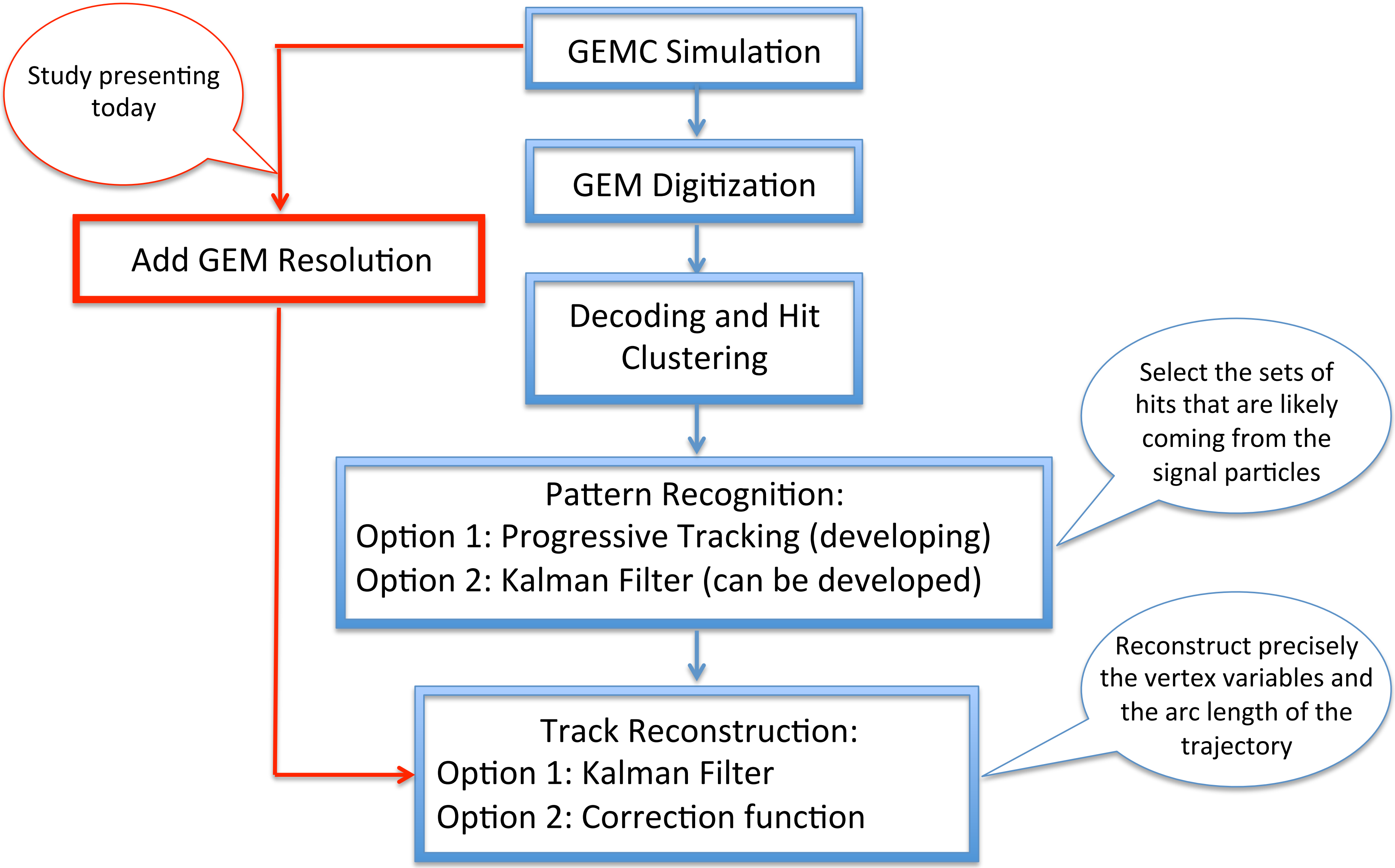
Tracking Framework Overview



Tracking Framework Overview

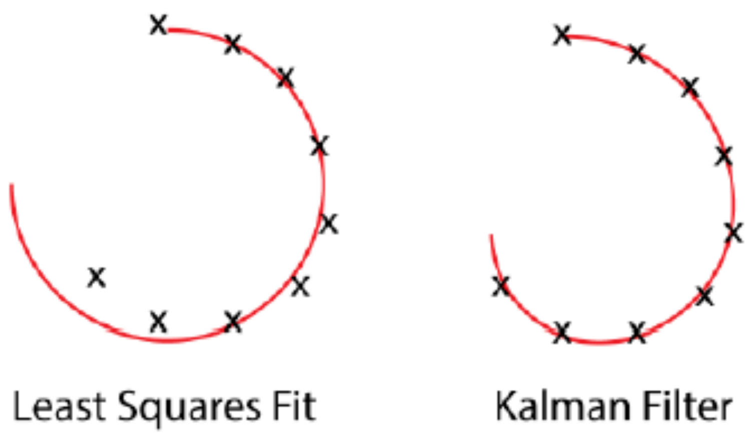


Tracking Framework Overview



Introduction on Kalman Filter

- Kalman filter vs Least square fit:
  - Both are χ^2 minimizing method
  - Kalman filter is a recursive fitting algorithm
  - Kalman filter allows the fitting parameters (state vector) to change along the trajectory (easy to make correction due to energy loss and inhomogeneous field)



Introduction on Kalman Filter

- Question: If I have an optimal state vector a_{k-1} on detector k-1, how to get the optimal state vector on detector k?

Introduction on Kalman Filter

- Question: If I have an optimal state vector a_{k-1} on detector k-1, how to get the optimal state vector on detector k?

1. Prediction: Using a_{k-1}, predict what the state vector will be on detector k, by means of a propagation method

Introduction on Kalman Filter

- Question: If I have an optimal state vector a_{k-1} on detector k-1, how to get the optimal state vector on detector k?

1. Prediction: Using a_{k-1}, predict what the state vector will be on detector k, by means of a propagation method

2. Error Propagation: propagate the covariance matrix of a_{k-1} toward detector k. A covariance matrix of the process noise (Q matrix) is also calculated based on the material the particle is passing through

Introduction on Kalman Filter

- Question: If I have an optimal state vector a_{k-1} on detector k-1, how to get the optimal state vector on detector k?

1. Prediction: Using a_{k-1}, predict what the state vector will be on detector k, by means of a propagation method

2. Error Propagation: propagate the covariance matrix of a_{k-1} toward detector k. A covariance matrix of the process noise (Q matrix) is also calculated based on the material the particle is passing through

3. Filtering: the weighted mean of the predicted state vector and measurement vector on detector k is calculated, yielding an optimal estimate of the state vector on detector k

Details of the Track Reconstruction Program

- The state vector used in the current program[2]:
  - (x, y, t_x, t_y, q/p)
- The propagation of state vector and its covariance matrix is done based on 4th order classical Runge-Kutta method (similar to RKClassicalRK4 class in Geant4)
- This approach is designed to work for both SIDIS and PVDIS. Changing from one configuration to the other requires nothing but redefining detector locations
- For SIDIS and J/ψ, a helical state vector and associated propagation method is also available[3]

Details of the Track Reconstruction Program

- Currently the program has treatments for three types of process noises:
  - Coulomb multiple scattering (Molière formula)
  - Ionization (Bethe-Bloch formula)
  - Bremsstrahlung radiation (Bethe-Heitler formula)
- Calculation for the Q matrix and correction for energy are done step by step along the propagation, based on what material the particle is in

Details of the Track Reconstruction Program

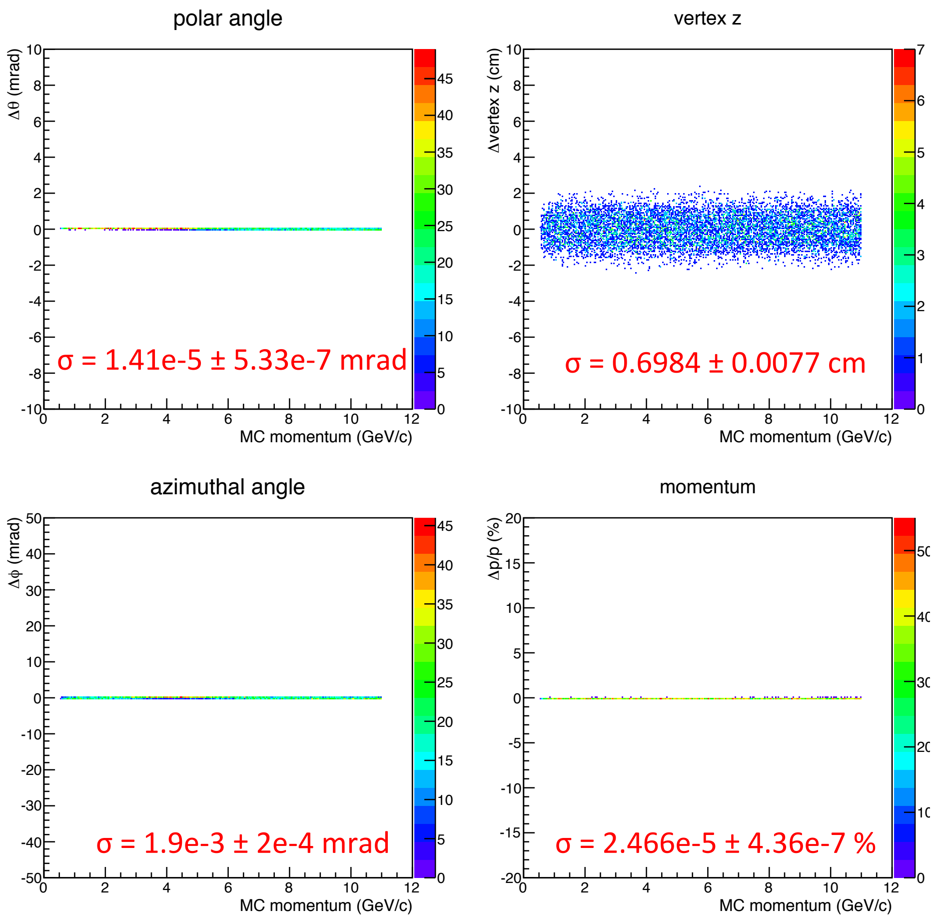
- Currently the program has treatments for three types of process noises:
  - Coulomb multiple scattering (Molière formula)
  - Ionization (Bethe-Bloch formula)
  - Bremsstrahlung radiation (Bethe-Heitler formula)
- Calculation for the Q matrix and correction for energy are done step by step along the propagation, based on what material the particle is in
- No process noise is simulated in the GEMC simulation presenting today, correspondingly this part of the code is switched off in the program

Tracking Reconstruction Basic Procedure

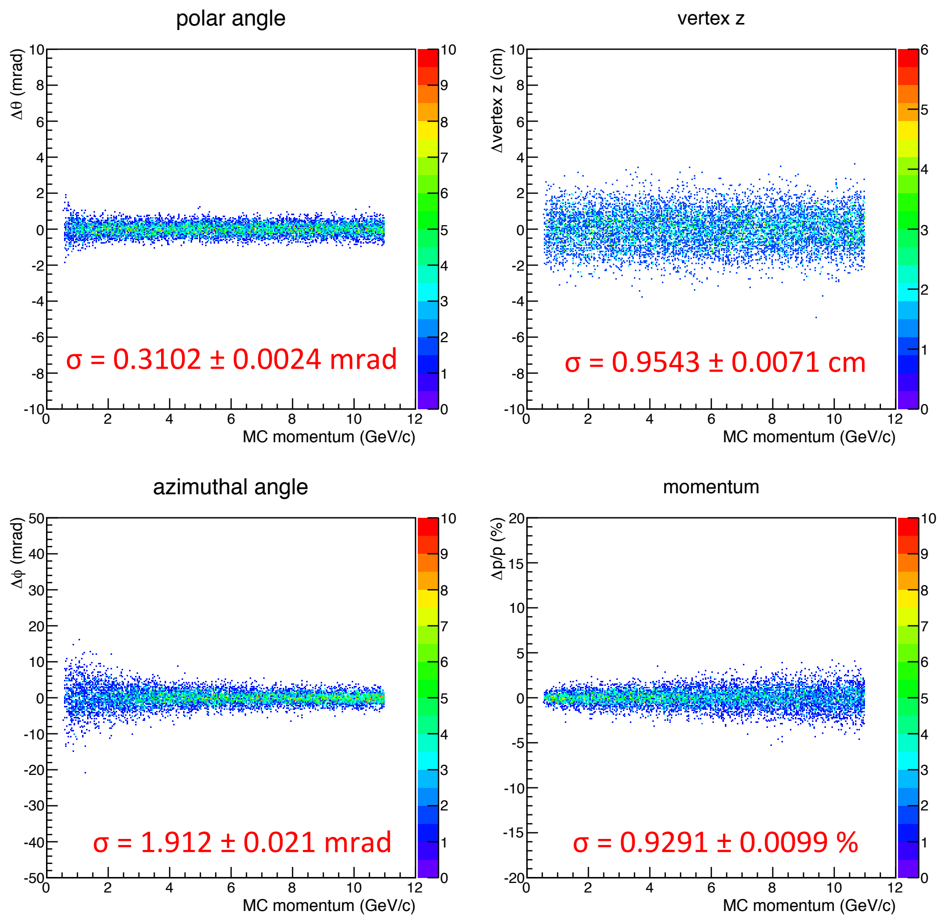
- Initializing the Kalman filter
- Fitting starts from the last GEM and moves towards the first one
- Propagate the state vector on the first GEM toward the target
- Find the interaction vertex
- Add the interaction vertex to the fit

Preliminary Tracking Reconstruction Results – SIDIS FA

GEM Resolution = 0 um
Interaction vertex not use in the fit



GEM Resolution = 90 um
Interaction vertex used in the fit

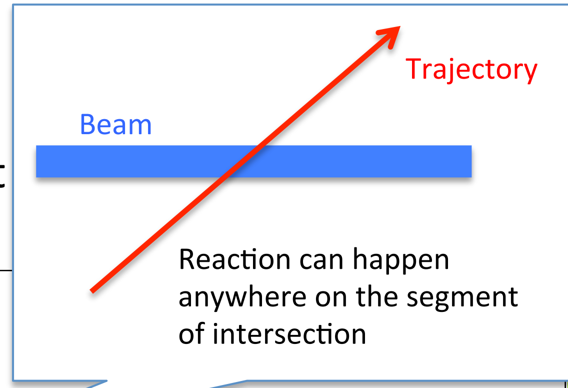


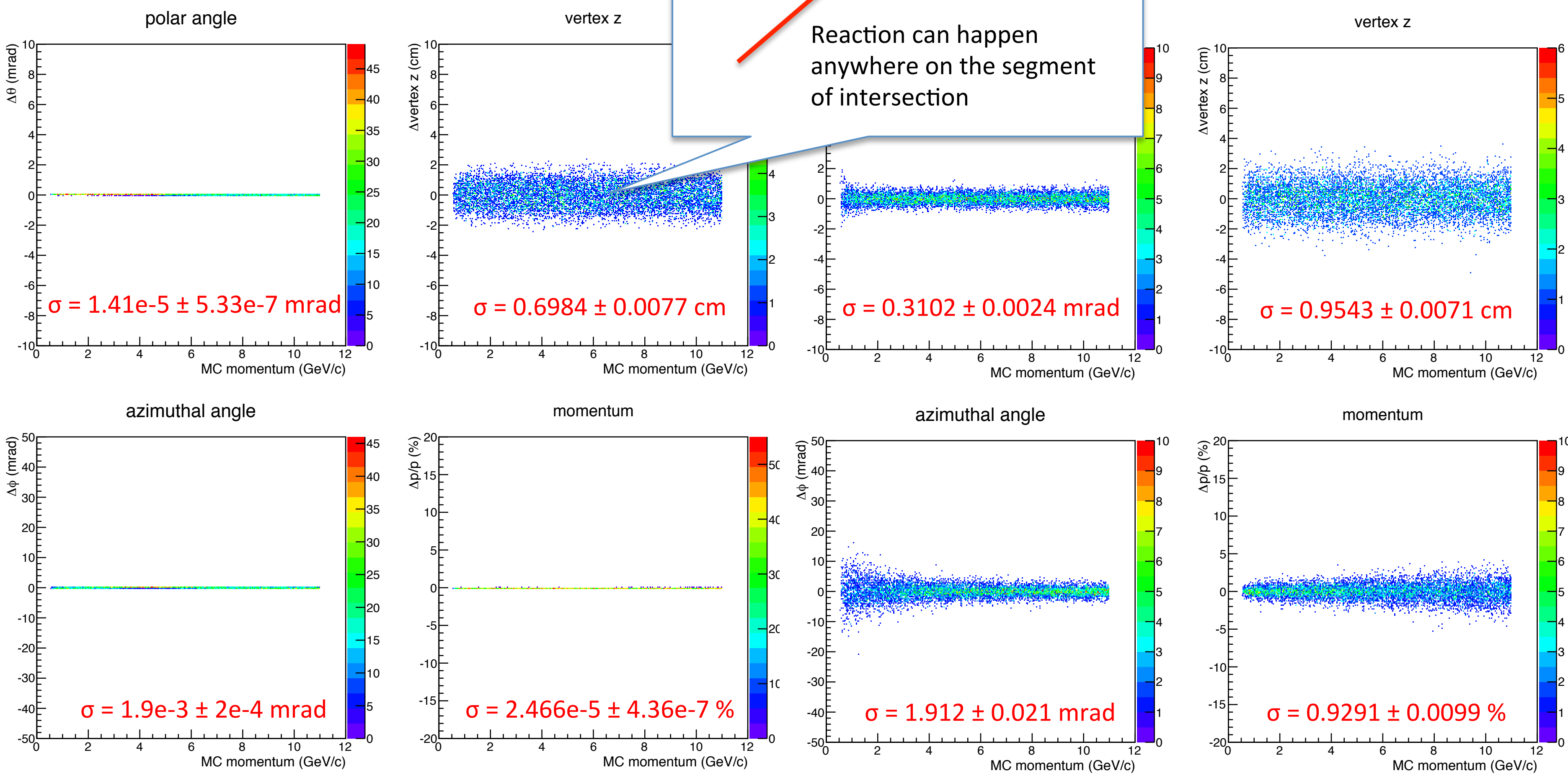
No process noise is simulated in this simulation. Generator is uniform.

Preliminary Tracking Reconstruction Results – SIDIS FA

GEM Resolution = 0 um
Interaction vertex not use in the fit

olution = 90 um
ertex used in the fit

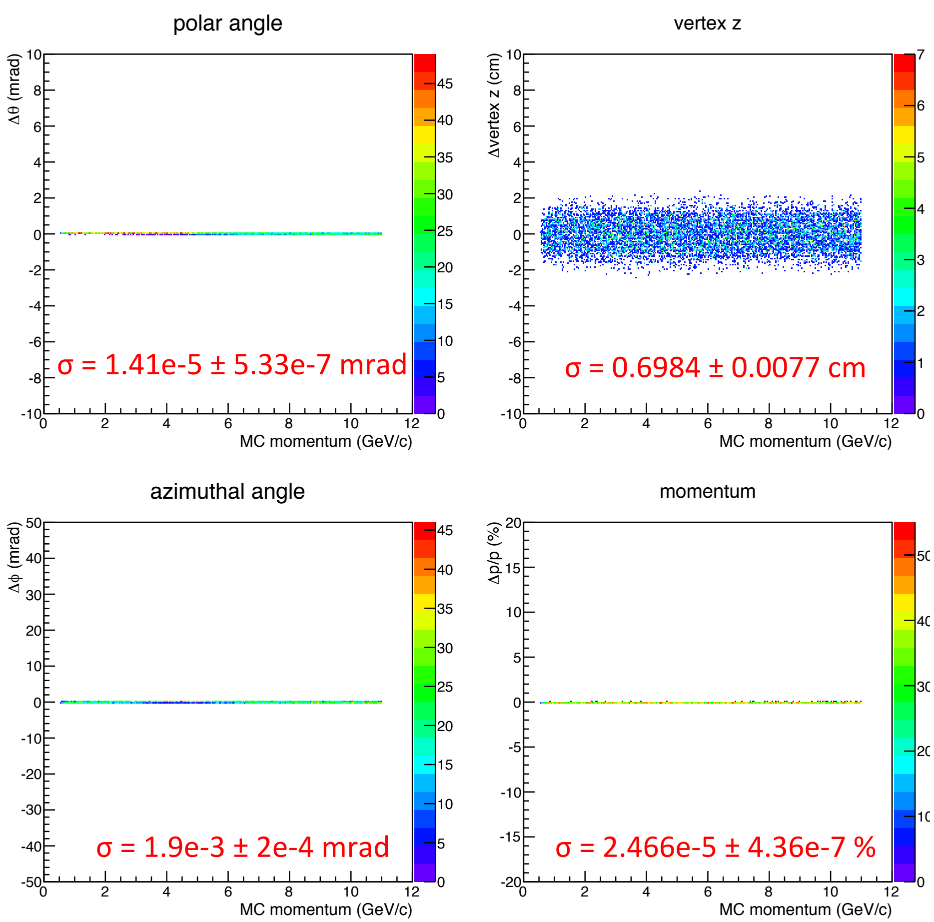




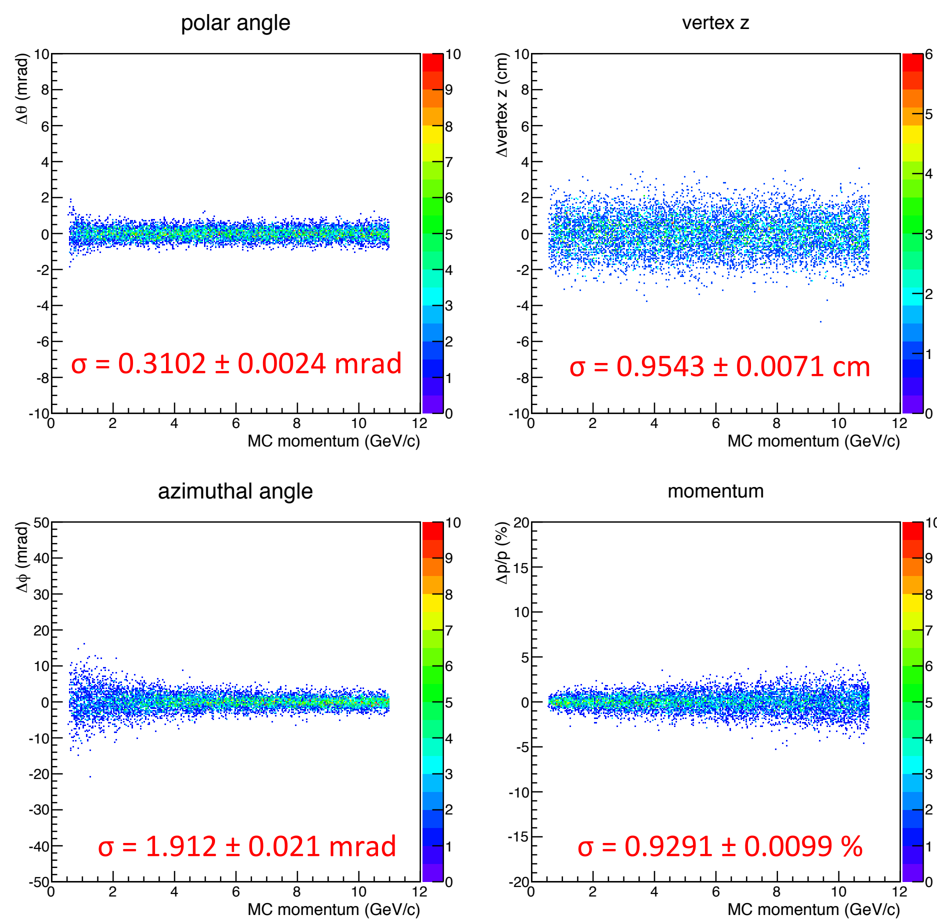
No process noise is simulated in this simulation. Generator is uniform.

Preliminary Tracking Reconstruction Results – SIDIS FA

GEM Resolution = 0 um
Interaction vertex not use in the fit



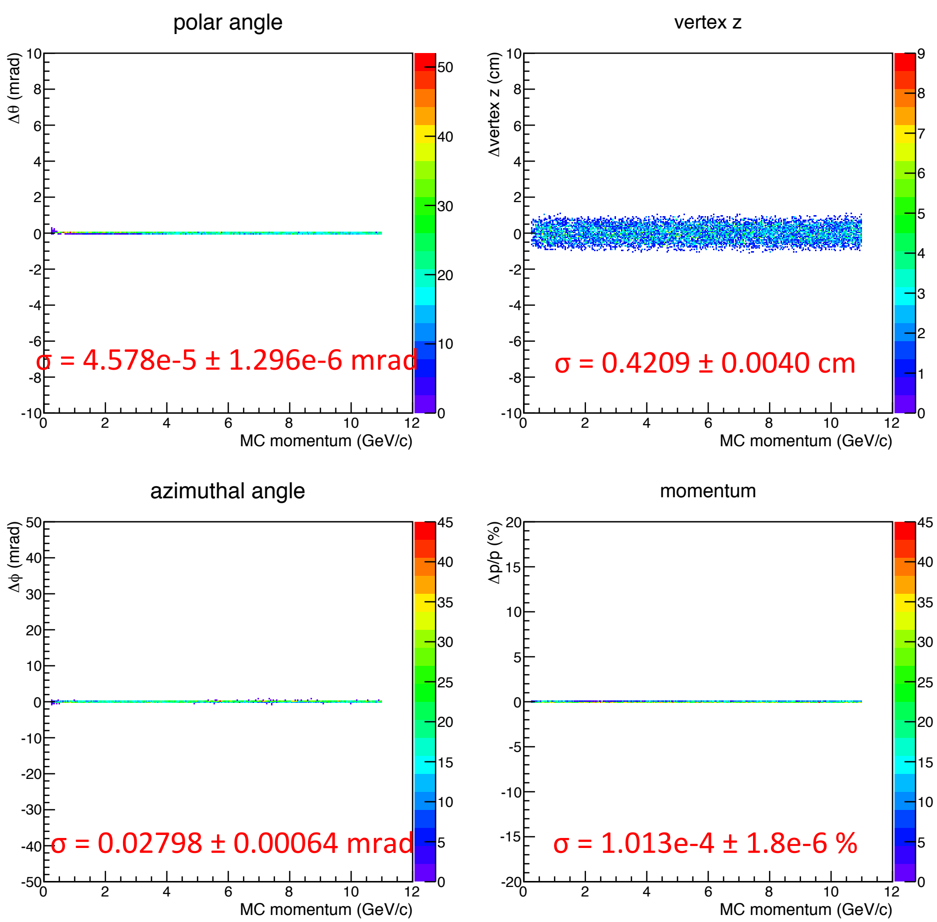
GEM Resolution = 90 um
Interaction vertex used in the fit



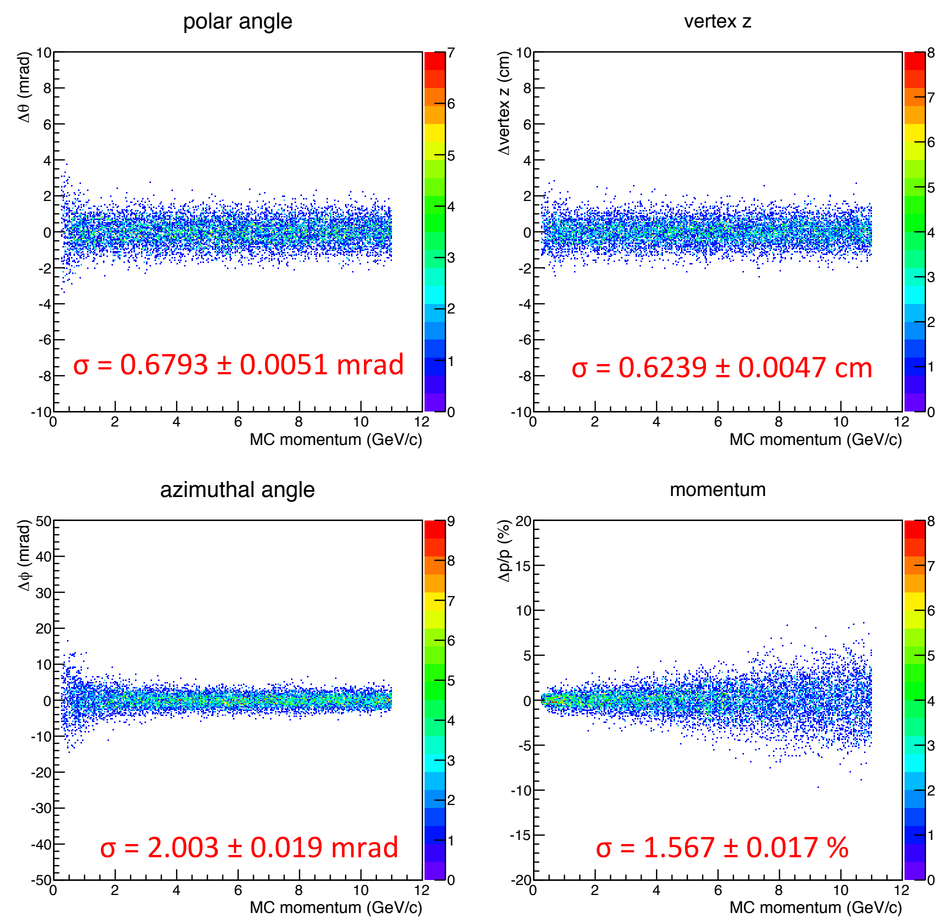
No process noise is simulated in this simulation. Generator is uniform.

Preliminary Tracking Reconstruction Results – SIDIS LA

GEM Resolution = 0 um
Interaction vectex not use in the fit



GEM Resolution = 90 um
Interaction vertex used in the fit

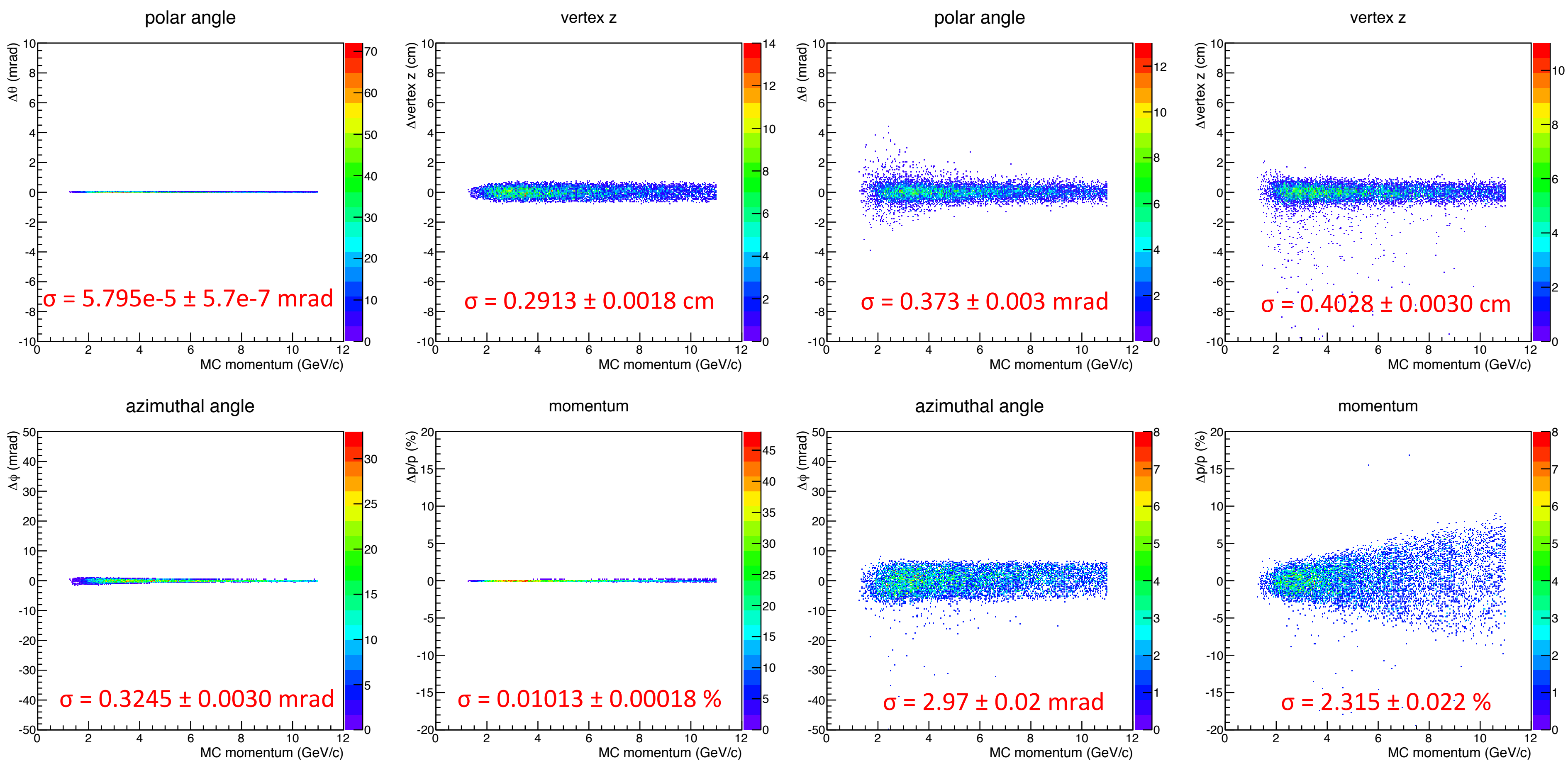


No process noise is simulated in this simulation. Generator is uniform.

Preliminary Tracking Reconstruction Results – PVDIS

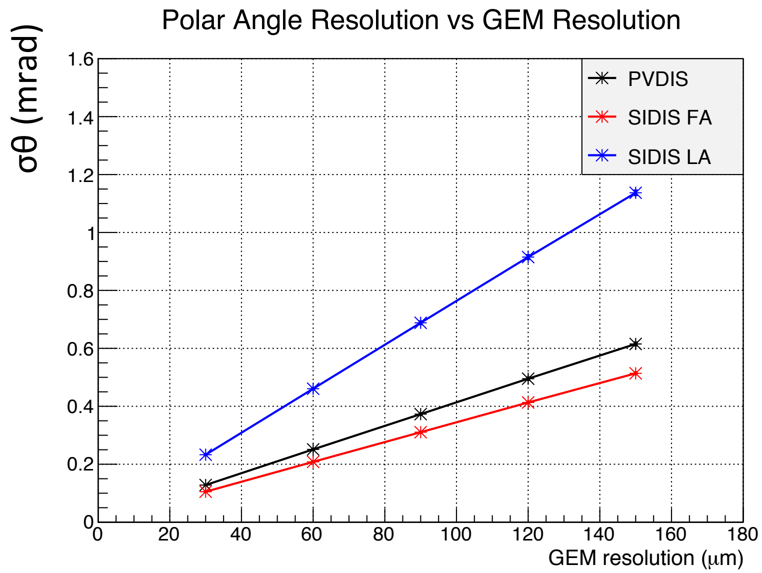
GEM Resolution = 0 um
Interaction vertex not use in the fit

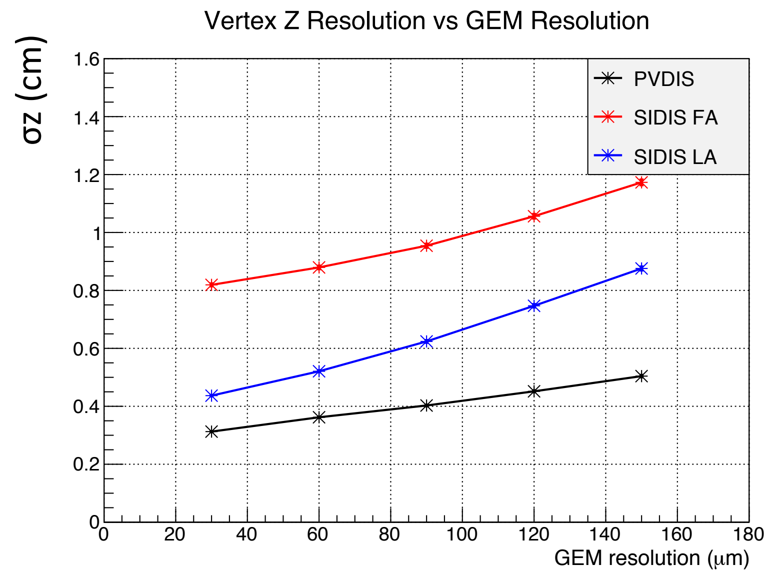
GEM Resolution = 90 um
Interaction vertex used in the fit

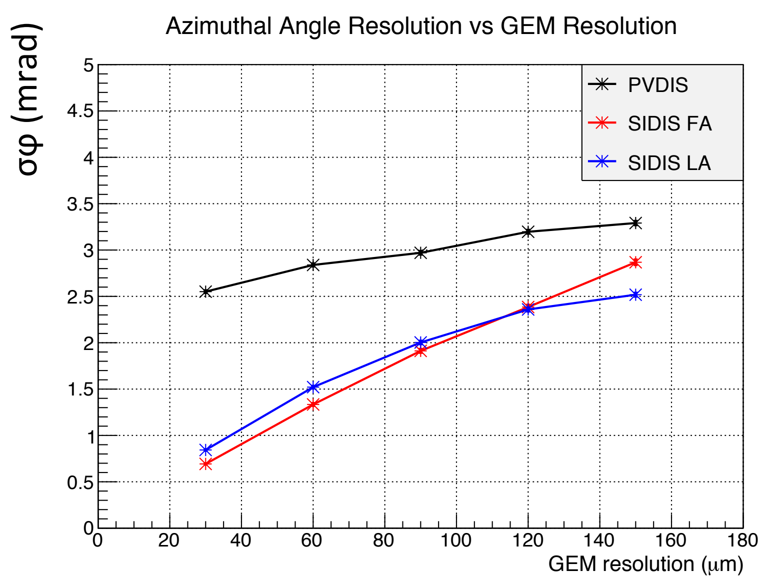


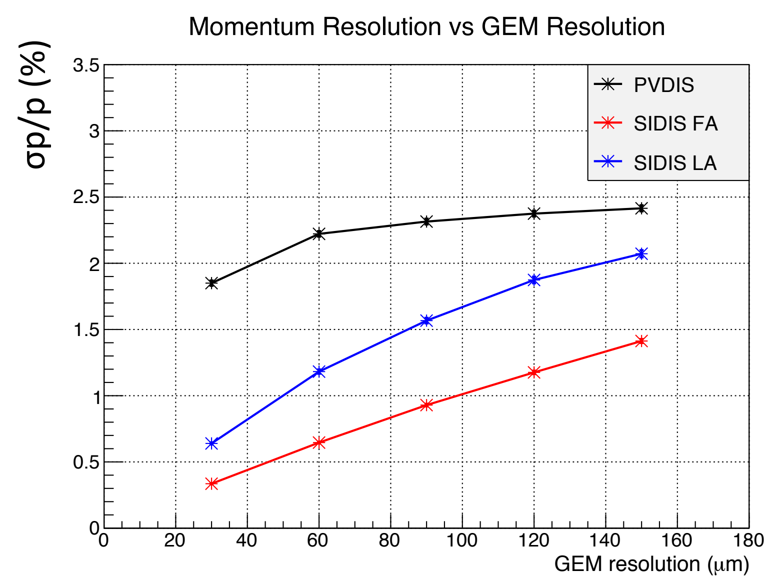
No process noise is simulated in this simulation. Generator is uniform.

Preliminary Tracking Reconstruction Results Comparison

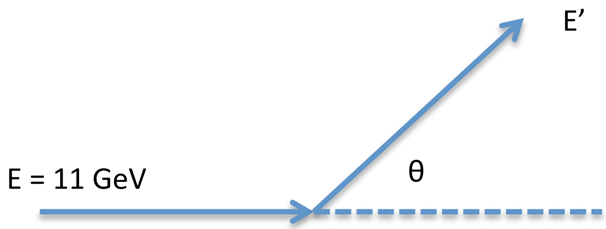




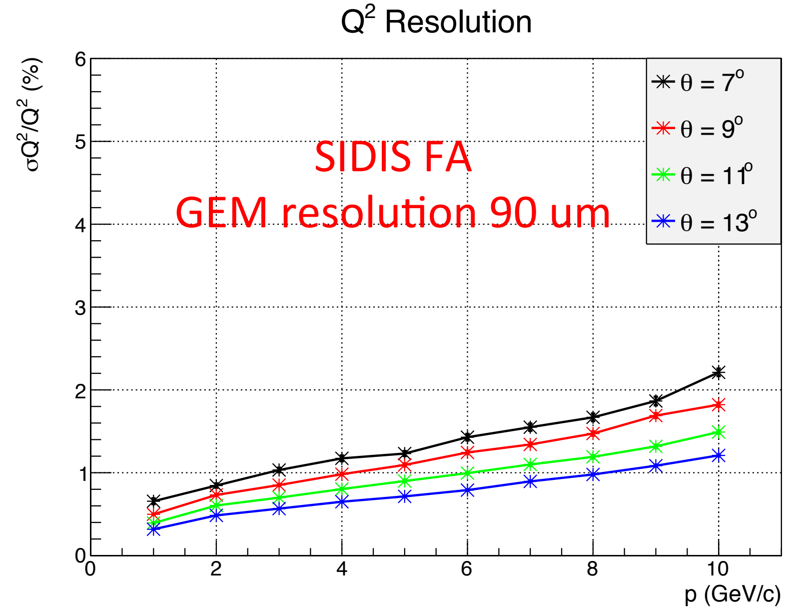


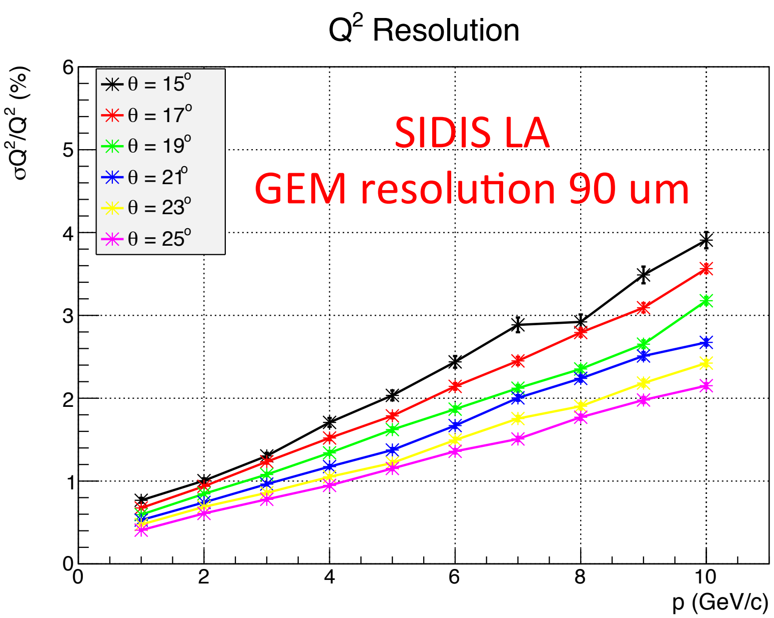


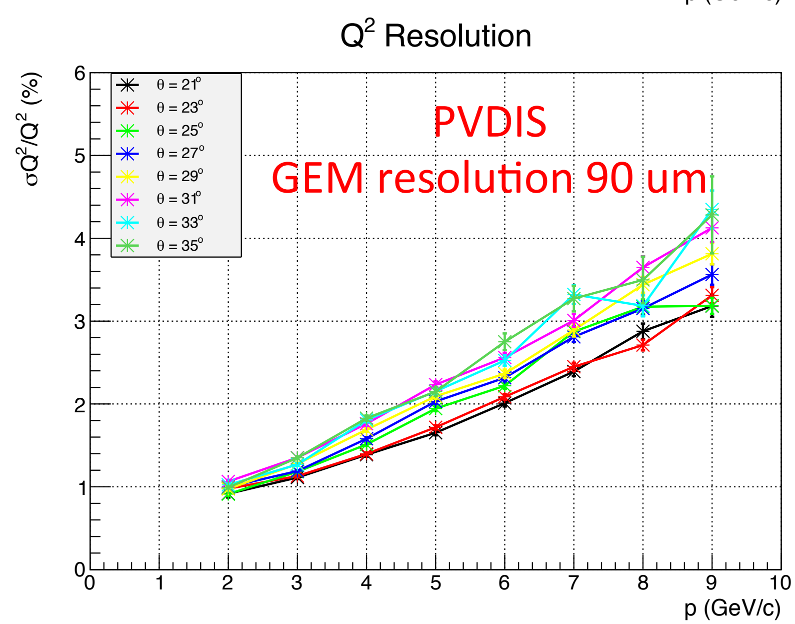
Preliminary Tracking Reconstruction Results
Q^2 Resolution



$$Q^2 = 2EE'(1-\cos\theta)$$







Conclusion

- Kalman filter with current track model and propagation method "works" for both SIDIS and PVDIS
  - Precise reconstruction in the case of no process noise and GEM smearing
  - Reasonable resolution under GEM smearing
- Only a starting point for this development. Further improvement quite necessary:
  - Better error estimation (initial covariance matrix, error in the interaction vertex…)
  - Better vertex finding algorithm
  - Better software structure
  - More functionalities
- This algorithm can be developed into a pattern recognition algorithm.

Plan

- Further Development for the Kalman filter reconstruction method
- Progressive tracking for PVDIS (curved track) is currently being developed, have a preliminary version, need more tests (digitization)
- Develop a Kalman filter pattern recognition algorithm if necessary

- Reference:

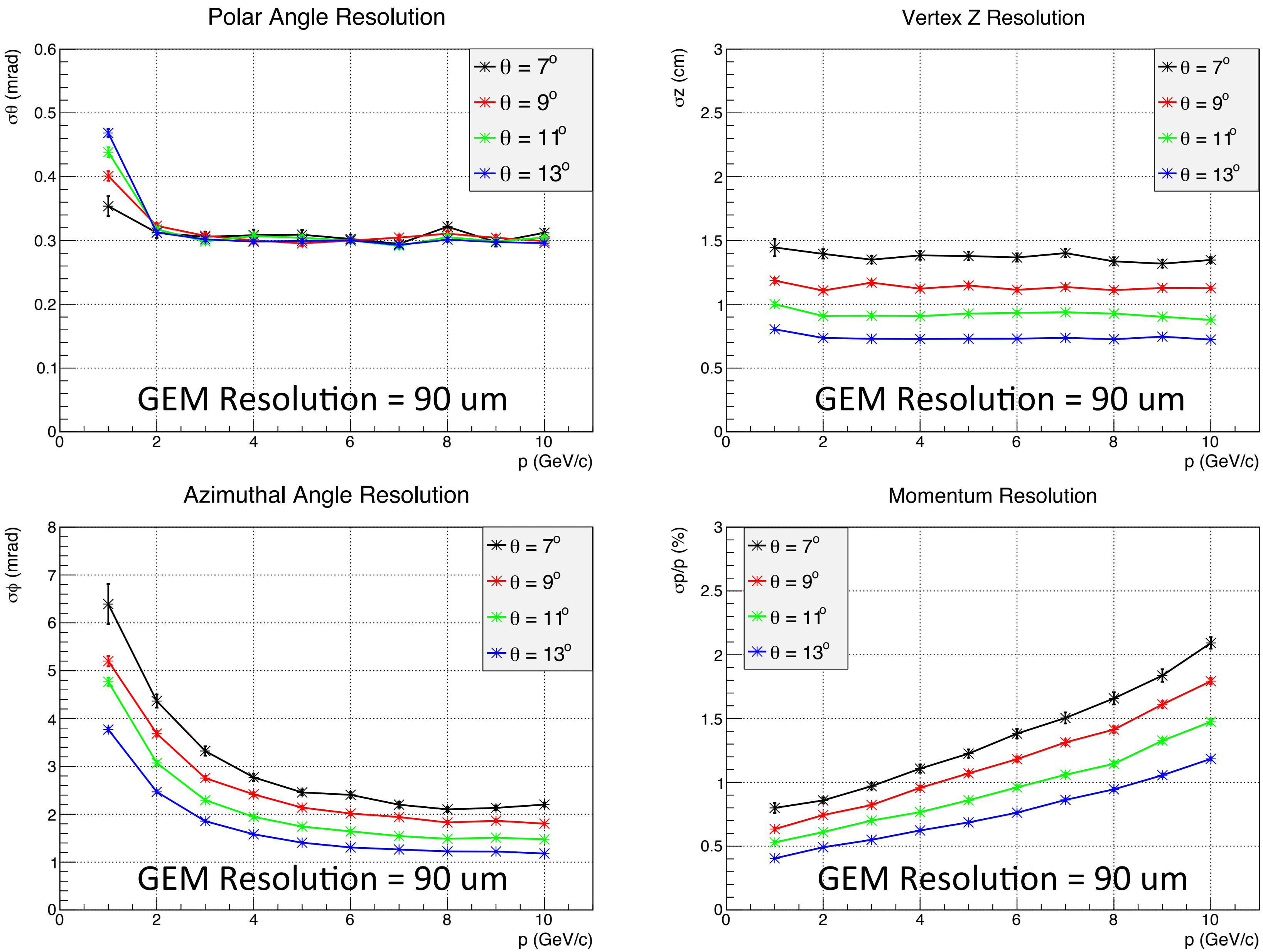
[1] Extended Kalman Filter. http://www-jlc.kek.jp/subg/offl/kaltest/doc/ReferenceManual.pdf

[2] Application of a Kalman Filter and a Deterministic Annealing Filter for Track Reconstruction in the HADES Experiment. https://www-alt.gsi.de/documents/DOC-2013-Aug-7-1.pdf

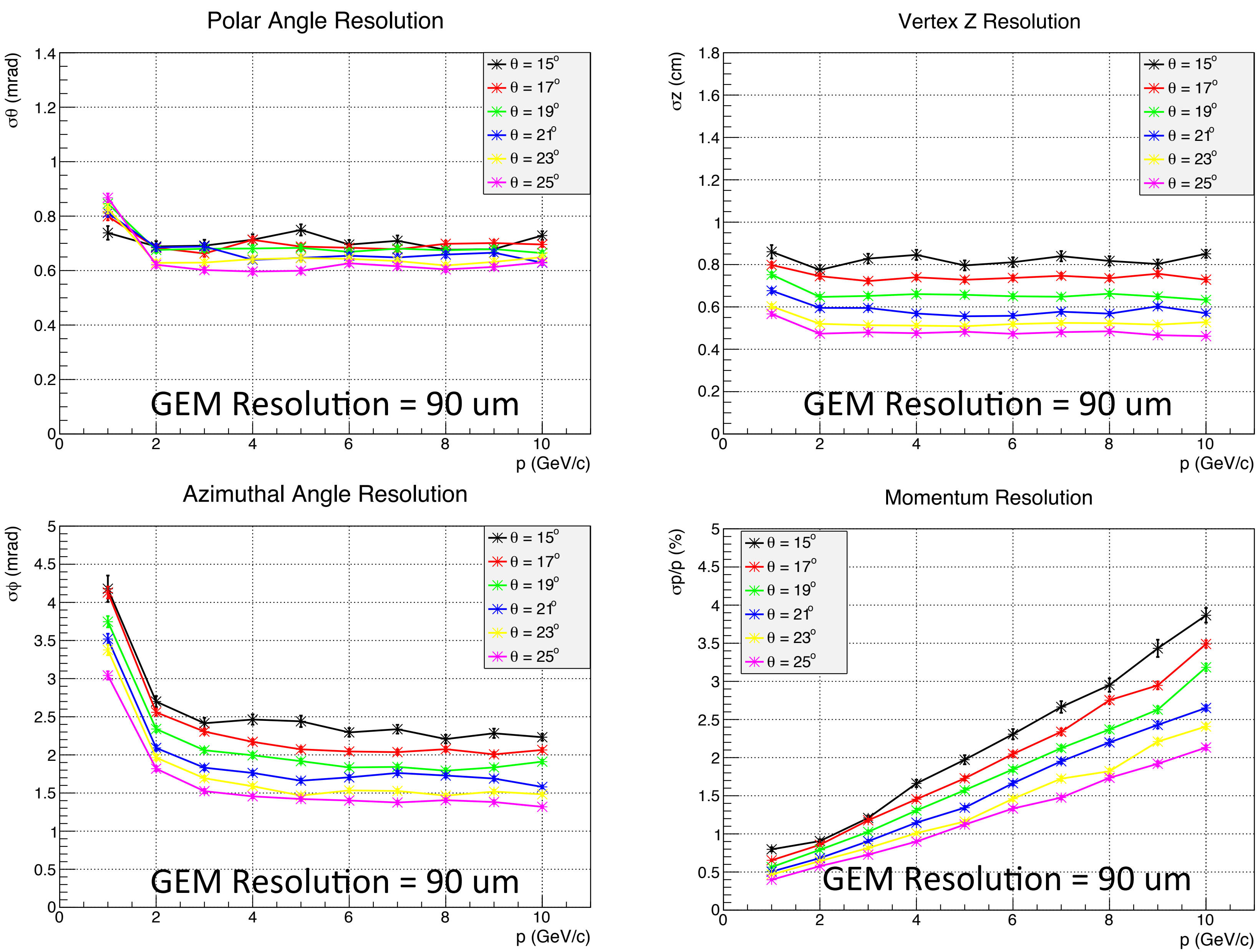
[3] Robert Kutschke. Billoir Fitter for CELO II

Backup Slides

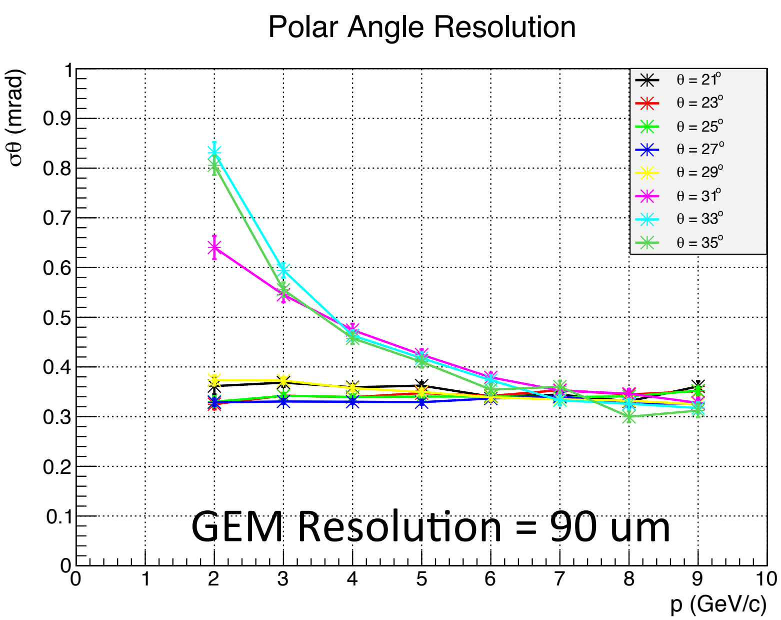
Preliminary Tracking Reconstruction Results – SIDIS FA

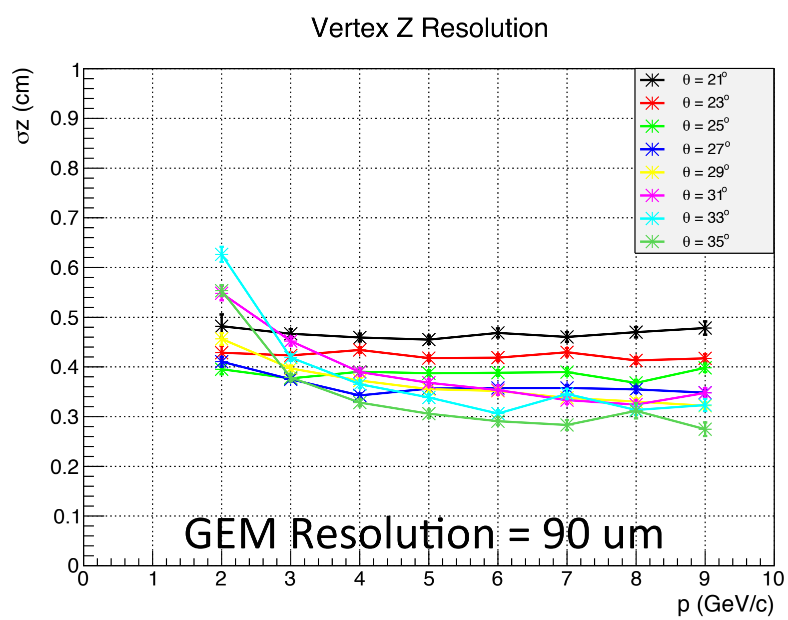


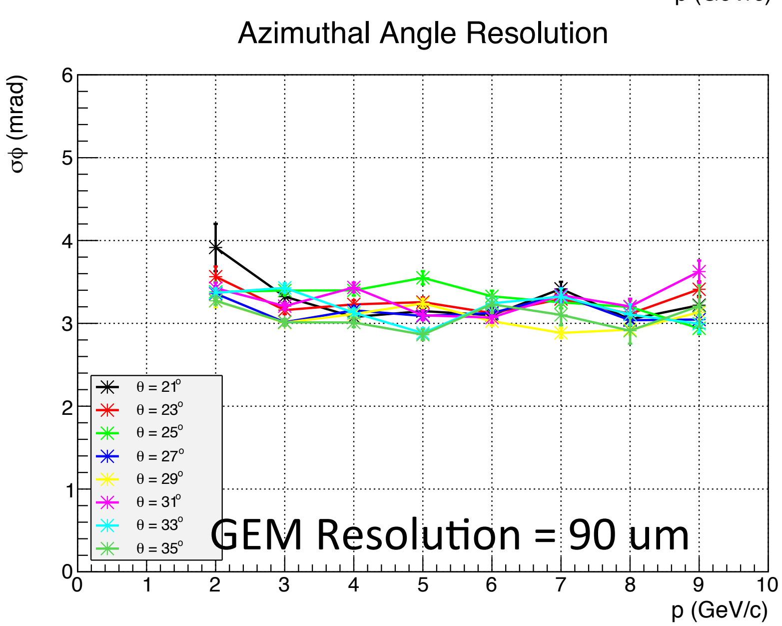
Preliminary Tracking Reconstruction Results – SIDIS LA

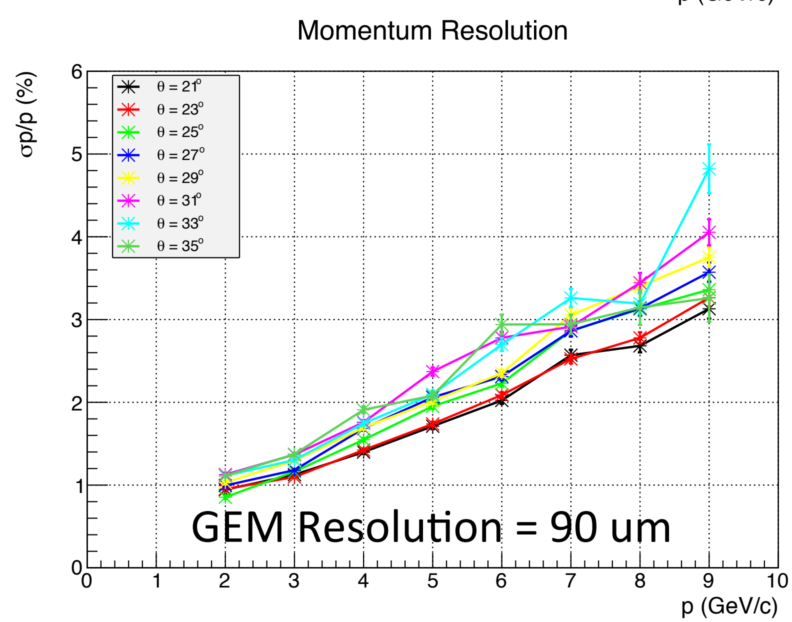


Preliminary Tracking Reconstruction Results – PVDIS

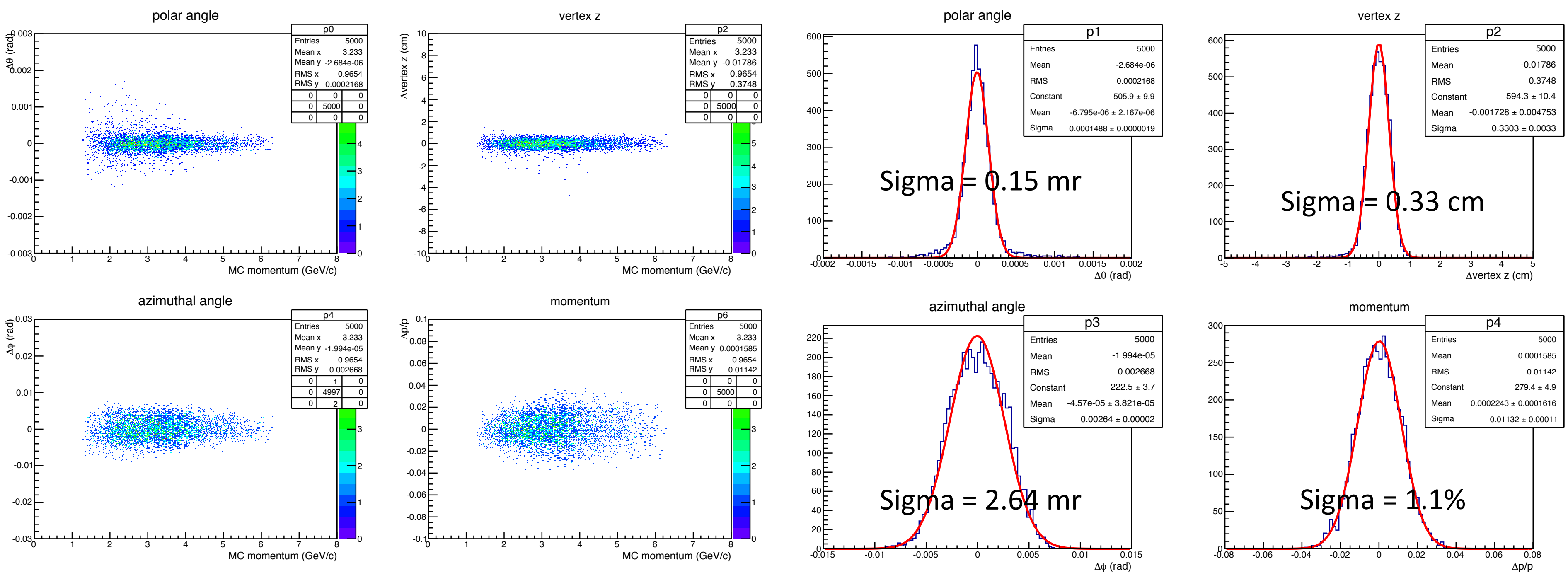








PVDIS Vertex Reconstruction



No process noise is simulated in this simulation. Generator is eDIS.

Introduction on Kalman Filter

- Basic steps for Kalman Filter[1]:
  - The optimized state vector a_{k-1} on detector k-1 is extrapolated to detector k by means of a propagation method

$$a_k^{k-1} = f_{k-1}(a_{k-1})$$

  - The covariance matrix of the predicted state vector, and also the covariance matrix of the process noise between detector k-1 and k are computed by error propagation

$$C_k^{k-1} = F_{k-1} C_{k-1} F_{k-1}^T + Q_{k-1}$$

$$F_{k-1} = \frac{\partial f_{k-1}(a_{k-1})}{\partial a_{k-1}}$$

Introduction on Kalman Filter

- Basic steps for Kalman Filter:
  - The predicted state vector and its covariance matrix are projected into the measurement space by means of a projector H, thus obtain the predicted measurement vector
  - The weighted mean of the extrapolated and the actual measurement vector m_k of detector k is computed, yielding an optimal estimate of the state vector at k

$$C_k = (I - K_k H_k) C_k^{k-1}$$

$$a_k = a_k^{k-1} + K_k (m_k - H_k a_k^{k-1})$$

$$K_k = C_k^{k-1} (H_k)^T (V_k + H_k C_k^{k-1} (H_k)^T)^{-1}$$

In the simplest case, assume 1d state vector x, and we directly measure this quality, then the Kalman filter formulae can be reduced to a simple form

$$\frac{x_o}{\sigma_o^2} = \frac{x_p}{\sigma_p^2} + \frac{x_m}{\sigma_m^2}$$

Classical 4th Order Runge-Kutta

First of all, take the derivative of the state vector with respect to z, this is the differential equation that is going to be solved by the Runge-Kutta method:

$$\frac{d\vec{a}}{dz} = \begin{pmatrix} dx/dz \\ dy/dz \\ dt_x/dz \\ dt_y/dz \\ d(q/p)/dz \end{pmatrix} = \begin{pmatrix} t_x \\ t_y \\ t'_x \\ t'_y \\ 0 \end{pmatrix} = f(\vec{a}, z)$$

The solution of the 4th order Runge-Kutta method is:

$$\Delta\vec{a}_1 = h \cdot f(\vec{a}(z_0), z_0)$$

$$\Delta\vec{a}_2 = h \cdot f(\vec{a}(z_0) + \frac{1}{2}\Delta\vec{a}_1, z_0 + \frac{1}{2}h)$$

$$\Delta\vec{a}_3 = h \cdot f(\vec{a}(z_0) + \frac{1}{2}\Delta\vec{a}_2, z_0 + \frac{1}{2}h)$$

$$\Delta\vec{a}_4 = h \cdot f(\vec{a}(z_0) + \Delta\vec{a}_3, z_0 + h)$$

$$\vec{a}_f = \vec{a}_0 + \frac{1}{6}\Delta\vec{a}_1 + \frac{1}{3}\Delta\vec{a}_2 + \frac{1}{3}\Delta\vec{a}_3 + \frac{1}{6}\Delta\vec{a}_4 + O(h^5)$$

Classical 4th Order Runge-Kutta

- The propagator matrix of this process is:

$$F = \frac{d\vec{a}_f}{d\vec{a}_0} = I + \frac{1}{6}F_1 + \frac{1}{3}F_2 + \frac{1}{3}F_3 + \frac{1}{6}F_4$$

$$F_i = \frac{d\Delta\vec{a}_i}{d\vec{a}_0}$$

- Using this method, there is no need for a pre-defined geometric track. To initialize the fit, x_0 and y_0 can be taken from the last GEM tracker along the beam direction, tx_0 and ty_0 can be calculated using the last two GEM trackers (where field is low and track is almost straight), finally p is given by the calorimeter (we always have a EC hit for PVDIS)

Process Noise Treatment

- In general, there are three important process noises one will need to consider during tracking reconstruction
  - Coulomb multiple scattering:

$$C_{MS} = \left(\frac{13.6\,MeV}{\beta pc}\right)^2 t\,(1 + 0.0038\ln t)^2$$

  - Ionization:
    - For spinless particle: $$-\left\langle\frac{dE}{dx}\right\rangle = Kz^2\frac{Z}{A}\frac{1}{\beta^2}\left(\frac{1}{2}\ln\left(\frac{2m_ec^2\beta^2\gamma^2T_{max}}{I^2}\right) - \beta^2\right)$$
    - For electron: $$-\left\langle\frac{dE}{dx}\right\rangle = \frac{1}{2}K\frac{Z}{A}\left(2\ln\frac{2m_ec^2}{I} + 3\ln\gamma - 1.95\right).$$

$$K = 4\pi N_A r_e^2 m_e c^2 \qquad T_{max} = \frac{2m_ec^2\beta^2\gamma^2}{1 + 2\gamma\frac{m_e}{M} + \left(\frac{m_e}{M}\right)^2}.$$

  - Bremsstrahlung:

$$\left\langle\frac{dE}{dx}\right\rangle = -\frac{E}{X_0}.$$

More about Process Noise Treatment

- For each Runge-Kutta step that is taken, we make one correction for the process noise
- We ADD the lost energy to the track if the fitting is backward, while SUBTRACT the lost energy if the fitting goes forward
- The process noise matrix Q will mainly affected by Coulomb multiple scattering and Bremsstrahlung radiation. The effect of ionization on Q is neglected because this effect is only strong for low momentum particle, where resolution is dominated by Coulomb multiple scattering

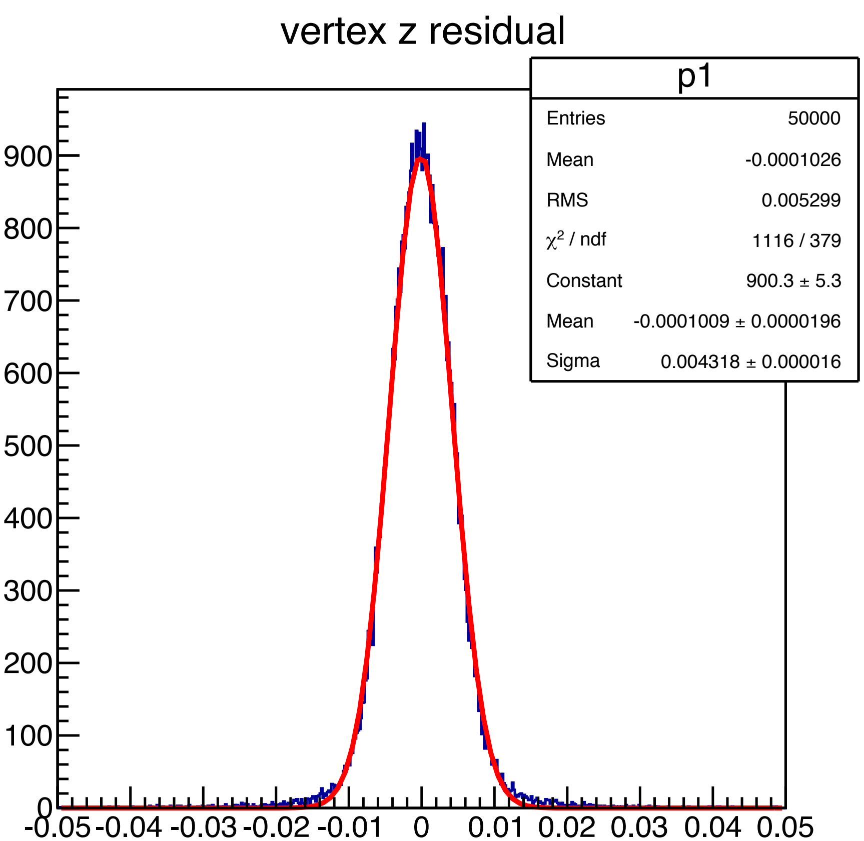
Effect on Q due to Coulomb Multiple scattering

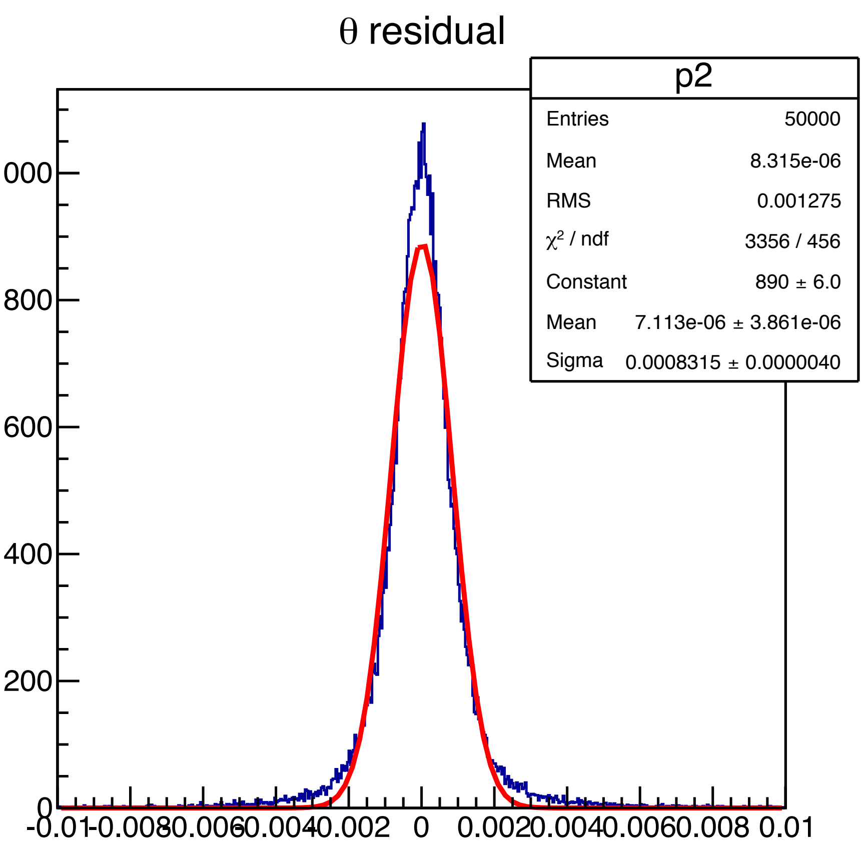
$$Q(t_x, t_x) = (1 + t_x^2) \cdot (1 + t_x^2 + t_y^2) \cdot C_{MS}$$
$$Q(t_y, t_y) = (1 + t_y^2) \cdot (1 + t_x^2 + t_y^2) \cdot C_{MS}$$
$$Q(t_x, t_y) = Q(t_y, t_x) = t_x t_y \cdot (1 + t_x^2 + t_y^2) \cdot C_{MS}.$$

Effect on Q due to Bremsstrahlung radiation

$$Q\left(\frac{q}{p}, \frac{q}{p}\right) = \left(\frac{q}{p}\right)^2 \left(e^{-t\frac{\ln 3}{\ln 2}} - e^{-2t}\right)$$

Simple Linear Least Square Fit

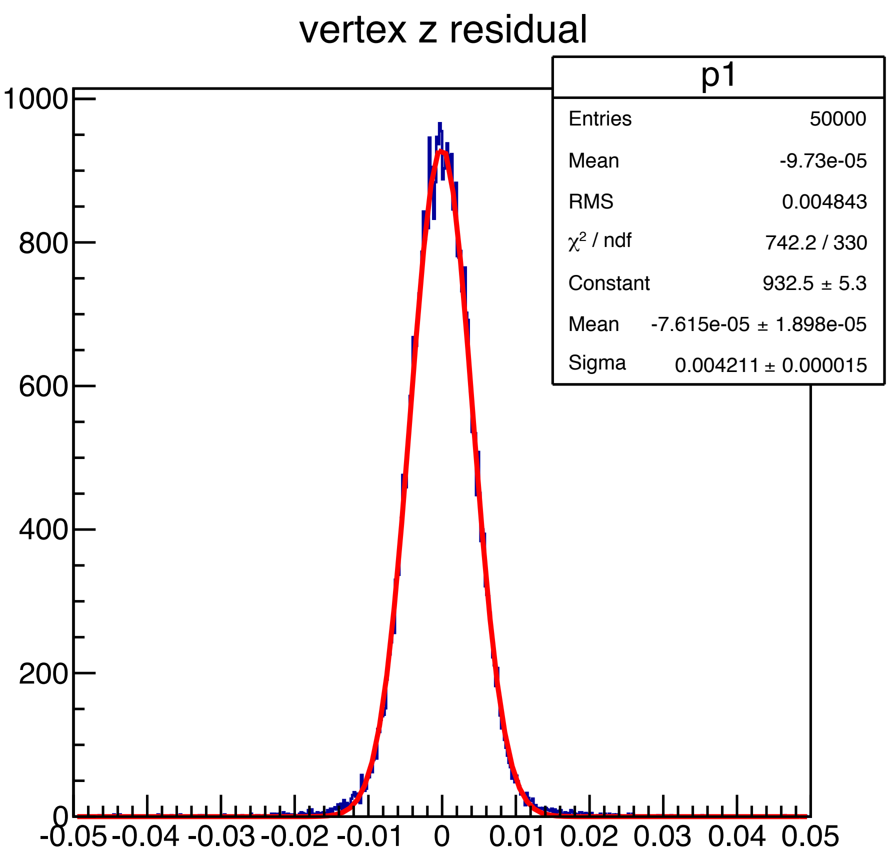




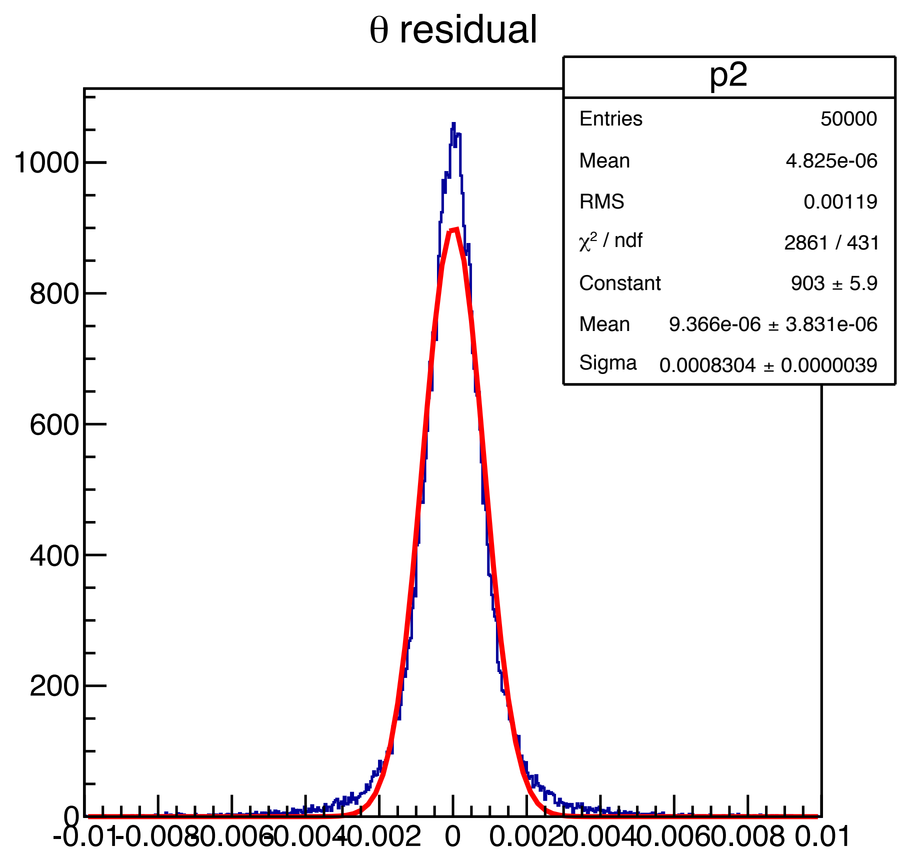
Sigma = 0.4318 cm
RMS = 0.5299 cm

Sigma = 0.8315 mrad
RMS = 1.275 mrad

Linear Least Square Fit with CMS



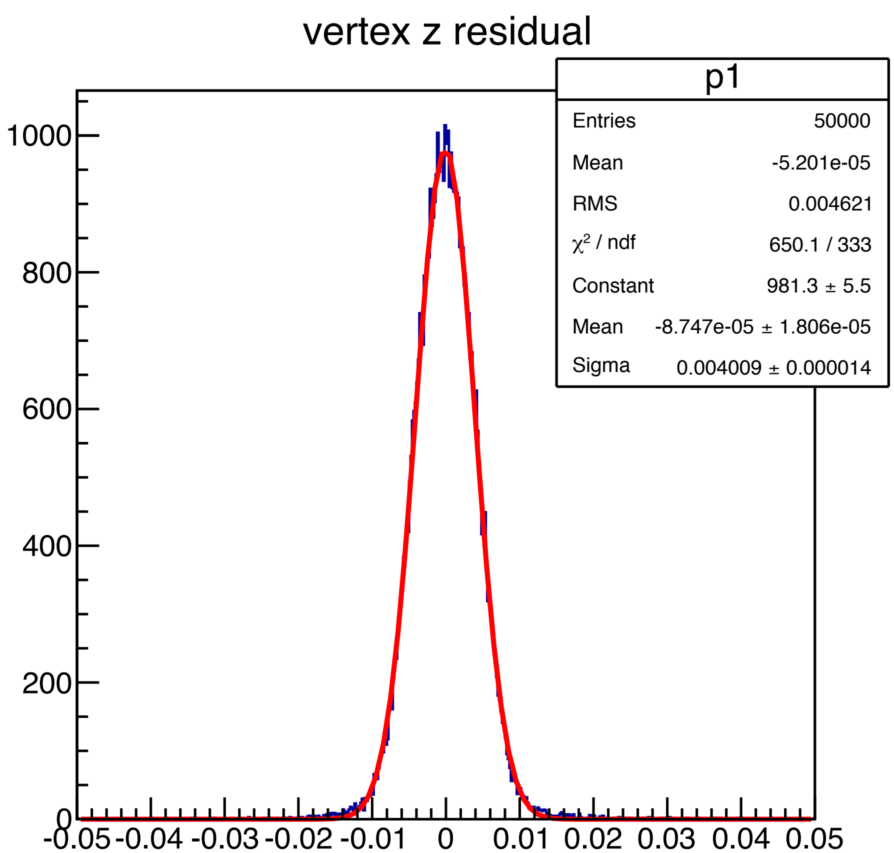
Sigma = 0.4211 cm
RMS = 0.4843 cm



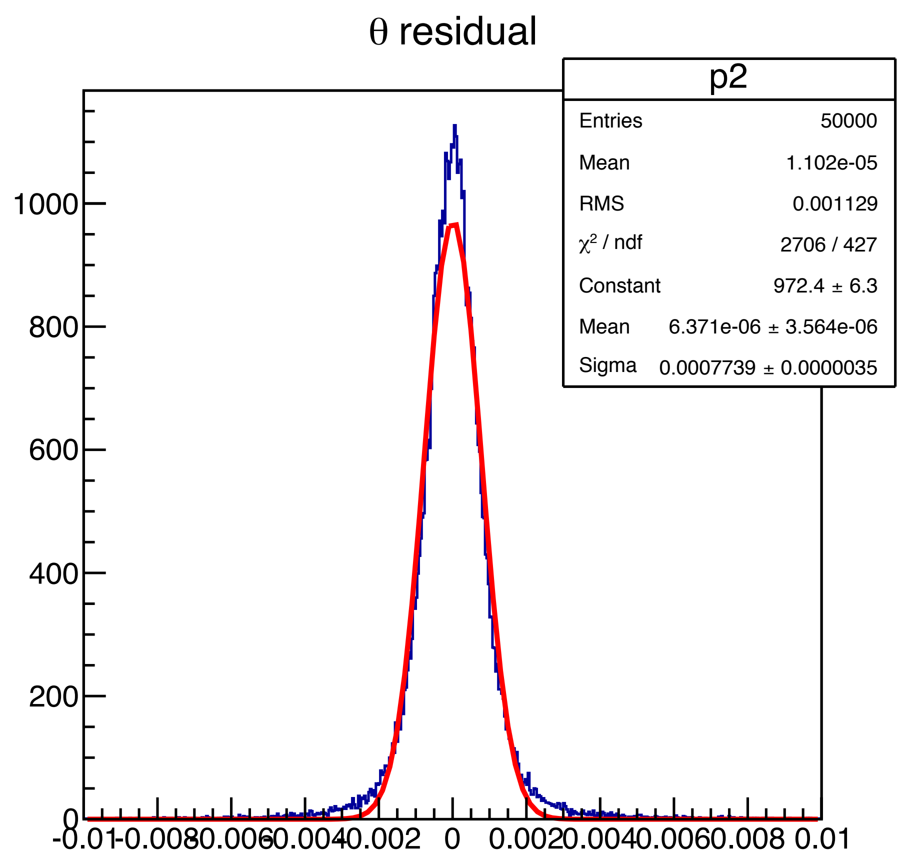
Sigma = 0.8304mrad
RMS = 1.19 mrad

Comment: Reconstructed vertex variables can be slightly better then before

Kalman Filter Fit with CMS

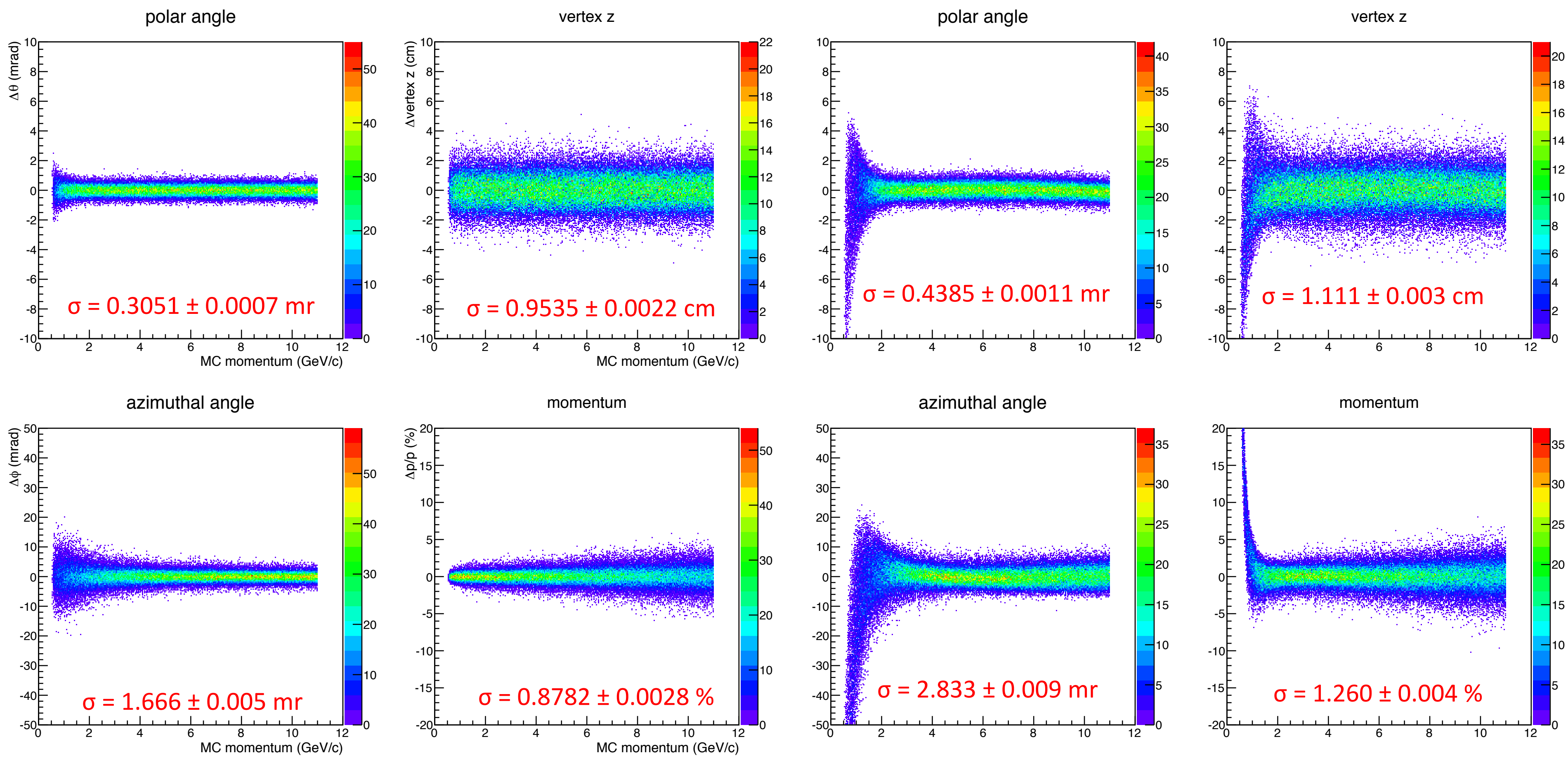


Sigma = 0.4009 cm
RMS = 0.4621 cm



Sigma = 0.7739mrad
RMS = 1.129 mrad

Comment: Best resolution for the reconstructed vertex variables among the three



All red numbers are the width of the Gaussian fit to the 1d distribution. A cut at momentum large than 1 GeV is used for both cases. Just to be consistent with the previous correction function approach. GEM resolution set at 90um.

Details of the Track Reconstruction Program

- What the Kalman filter reconstruction program should have:
  - Abstract classes for basic Kalman filter theory
  - Trajectory models and their corresponding propagation methods
  - SoLID field map
  - Various geometric objects to describe different SoLID sub-detectors (plane, cylinder…), which should store the position and material info about the objects
  - Various treatments for process noises

General Info about the Simulation

- Generator is uniform, reconstruct trajectory as long as the particle hits all the GEMs and the calorimeter
- GEM resolution is added according to: