INTRODUCING AN ALICE BASED KALMAN FILTER FOR ND-GAR

OCTOBER 2022

UNIVERSITY OF OXFORD

Author:
Federico Battisti

INTRODUCING AN ALICE BASED KALMAN FILTER FOR ND-GAR

- This is an expansion on previous work done on a Kalman Filter study for ND-GAr-Lite:

1. Dune Collaboration meeting 26th January 2022 Nd-GAr parallel session: https://indico.fnal.gov/event/50215/contributions/232480/
2. ND-GAr weekly meeting 15th March 2022: https://indico.fnal.gov/event/53600/contributions/236685/
3. DUNE Collaboration meeting 18th May 2022: https://indico.fnal.gov/event/50217/contributions/241519/
4. ND-GAr weekly meeting 9th August 2022: https://indico.fnal.gov/event/55842/

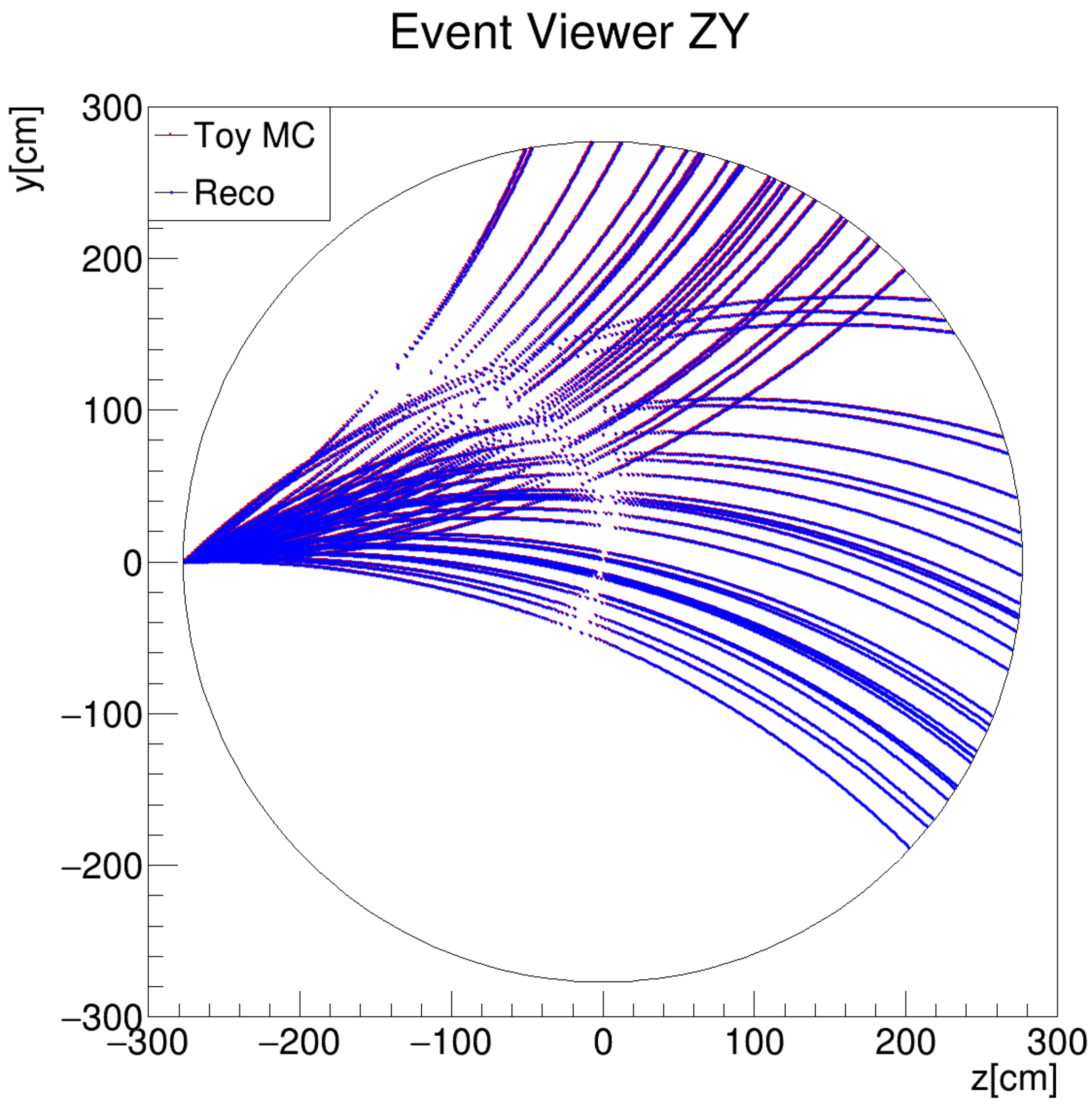
- In today's presentation:

1. Introduce Toy Monte-Carlo tool used for the study (fastMCKalman)
2. Introduce concept for an ALICE-based "radial" Kalman Filter for ND-GAr
3. Show results of early tests and compare with ND-GAr-Lite as a sanity check

SIMULATION

TOY MONTE CARLO

- fastMCKalman: Toy Monte Carlo tool created to test and develop reconstruction algorithms for TPC detectors (credit to Professor Marian Ivanov https://github.com/miranov25/fastMCKalman):
  - Generate particles with given initial total momentum, charge, angle and initial position
  - Propagate step by step the helix parameters $(y, x, sin\phi, \tan\lambda, \frac{q}{p_T})$ until particle leaves inner tracking volume (Note: ϕ azimuthal angle, λ dip-angle, p_T transverse momentum in yz plane)
  - At each step simulate Energy Loss and Multiple Scattering (both can be switched on and off)
- The 10k muon test sample was produced using the same charges, momenta and initial xy positions as the sample analyzed for latest KF study on ND-GAr-Lite (https://indico.fnal.gov/event/55842/)



TOY MONTE CARLO: ENERGY LOSS

Bethe-Bloch (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\frac{1}{\rho}\frac{dE}{dx} = K \times \frac{Z}{A} \times \frac{z^2}{\beta^2}\left[\frac{1}{2} ln\left(\frac{2m_e\gamma^2\beta^2T_{max}}{I^2}\right) - \beta^2 - \frac{\delta}{2}\right] \quad \text{[GeV/(g/cm^2)]}$$

- Energy loss simulated in three steps:

  1. Calculate dE/dx with Bethe-Bloch and convert to dP/dx
  2. Calculate momentum loss over trajectory in small "momentum-loss" steps: $n_{steps} = 1 + (dp/dx \times \Delta x)/step$ $(step = 0.005\ GeV/c)$
  3. Convert momentum loss first into energy loss $\Delta \mathrm{E} = E_{out} - E_{in}$ then into multiplicative factor to update q/p_T:

$$\frac{q}{p_T} *= cP4 = \left(1 + \frac{\Delta E}{p_{mean}^2}(\Delta E + 2 \times E_{in})\right)$$

- Note: These formulas are the same as the ones used by Geant4

TOY MONTE CARLO: MULTIPLE SCATTERING

Molière Formula (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\theta_0 = \frac{13.6MeV}{\beta p} z\sqrt{x/X_0}[1 + 0.038\ln(x/X_0)]$$

- Multiple Scattering smearing simulated in three steps:

  1. Calculate width of the angular gaussian distribution produced by MS: θ_0 from Molière formula
  2. Propagate the error to the relevant Helix parameters, obtaining their respective σ's $(\sigma_{sin\phi}, \sigma_{tan\lambda}, \sigma_{q/p_T})$
  3. Smear parameters with Gauss distribution having for width the respective σ's

- Note: These formulas are the same as the ones used by Geant4

RECONSTRUCTION

KALMAN FILTER BASICS

$$\boxed{\hat{s}_k^+ = \hat{s}_k^- + K_k \tilde{r}}$$

Kalman Filter Update Equation

A Posteriori

A Priori

Kalman Gain: small if confidence in model (determined by covariance matrix P) high, large if confidence low

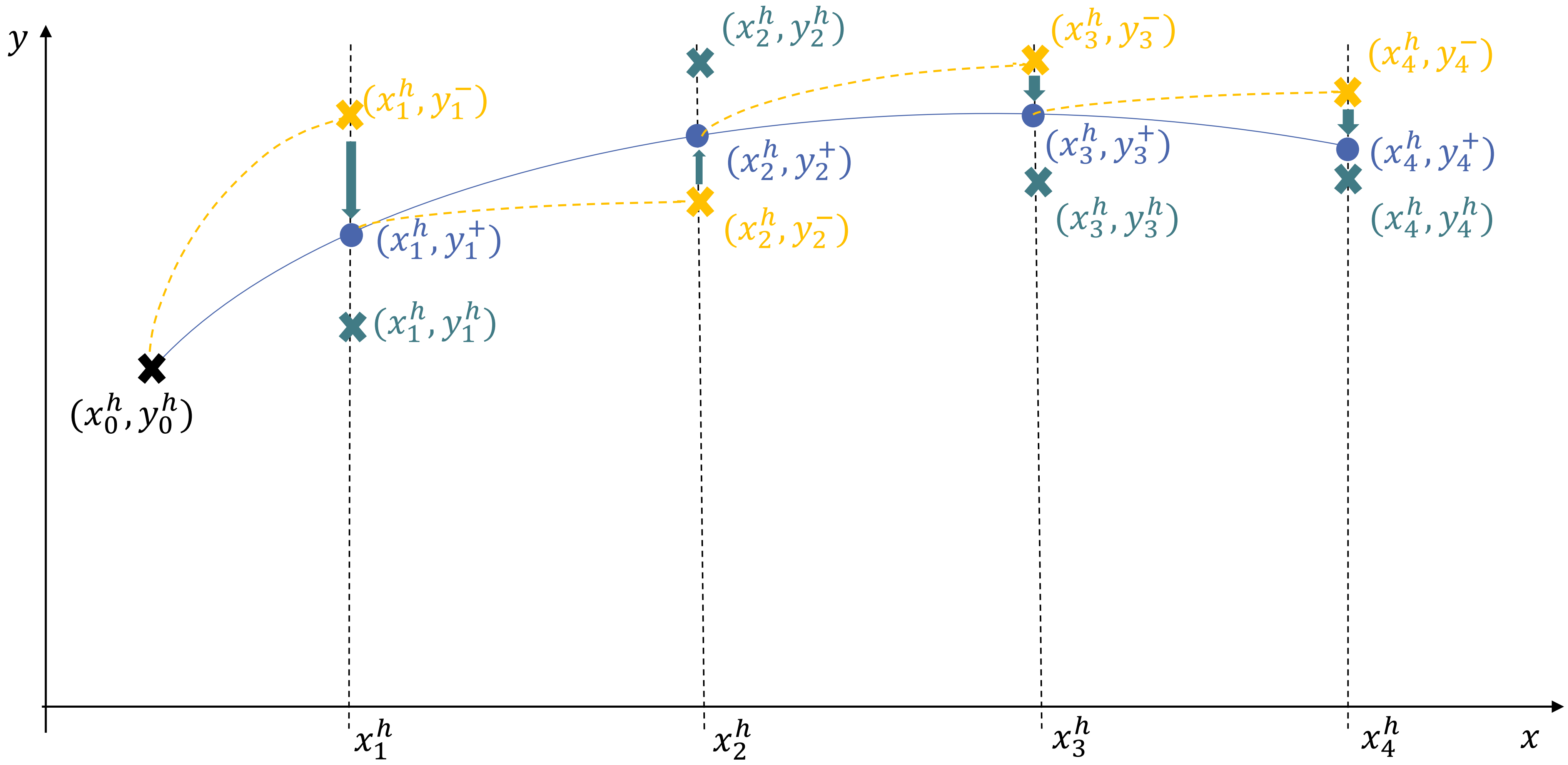
Residual: distance between measurement and a priori

- **Kalman filter:** iterative Bayesian algorithm which mediates between system knowledge and measurement. Each iteration divided in three steps:

  1. Make **A Priori prediction** of the state of the system using evolution model for the particle's trajectory
  2. Calculate **Residual:** distance between measurement and prediction
  3. Mediate between the a priori prediction and the measurement calculating **Kalman Gain** and produce **A Posteriori estimate**

Note: See back-up for further reading

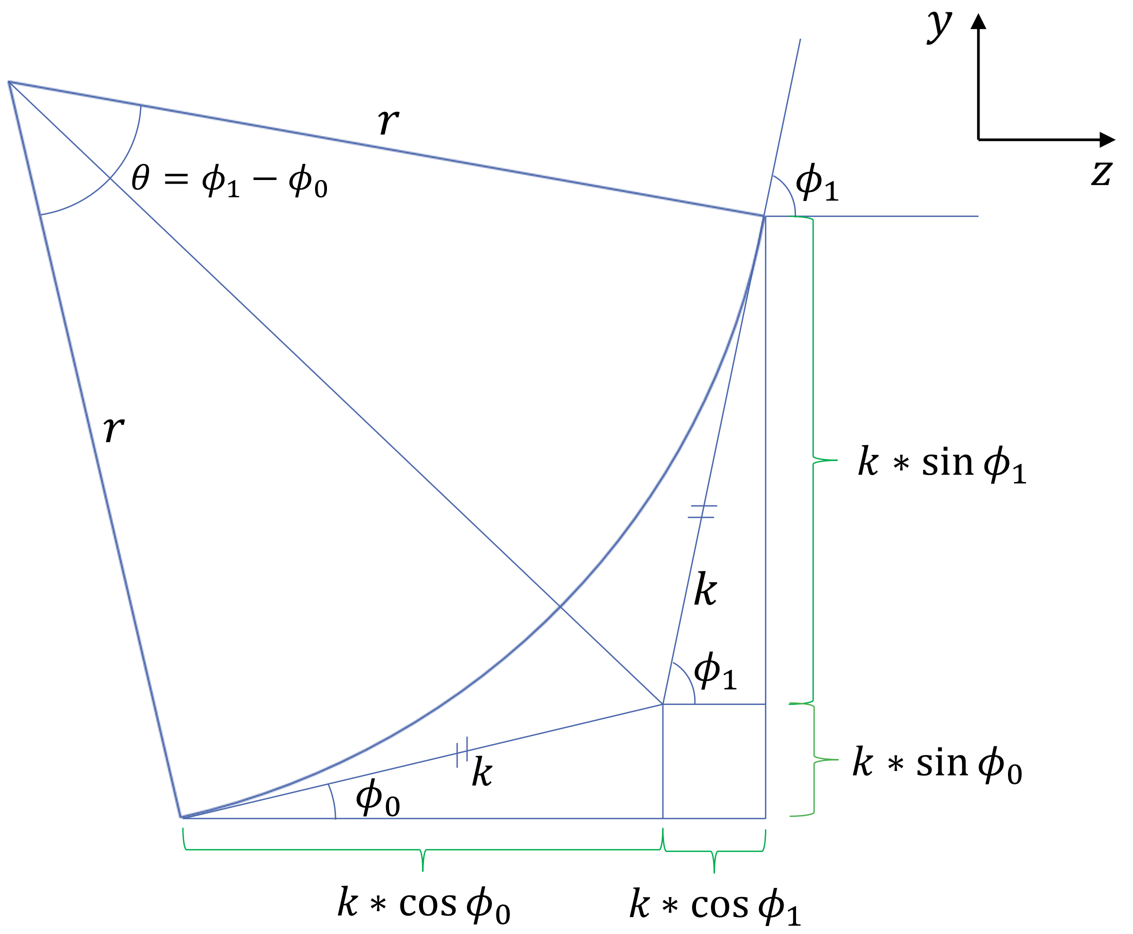
KALMAN FILTER BASICS



KALMAN FILTER MODEL AND APPLICATION

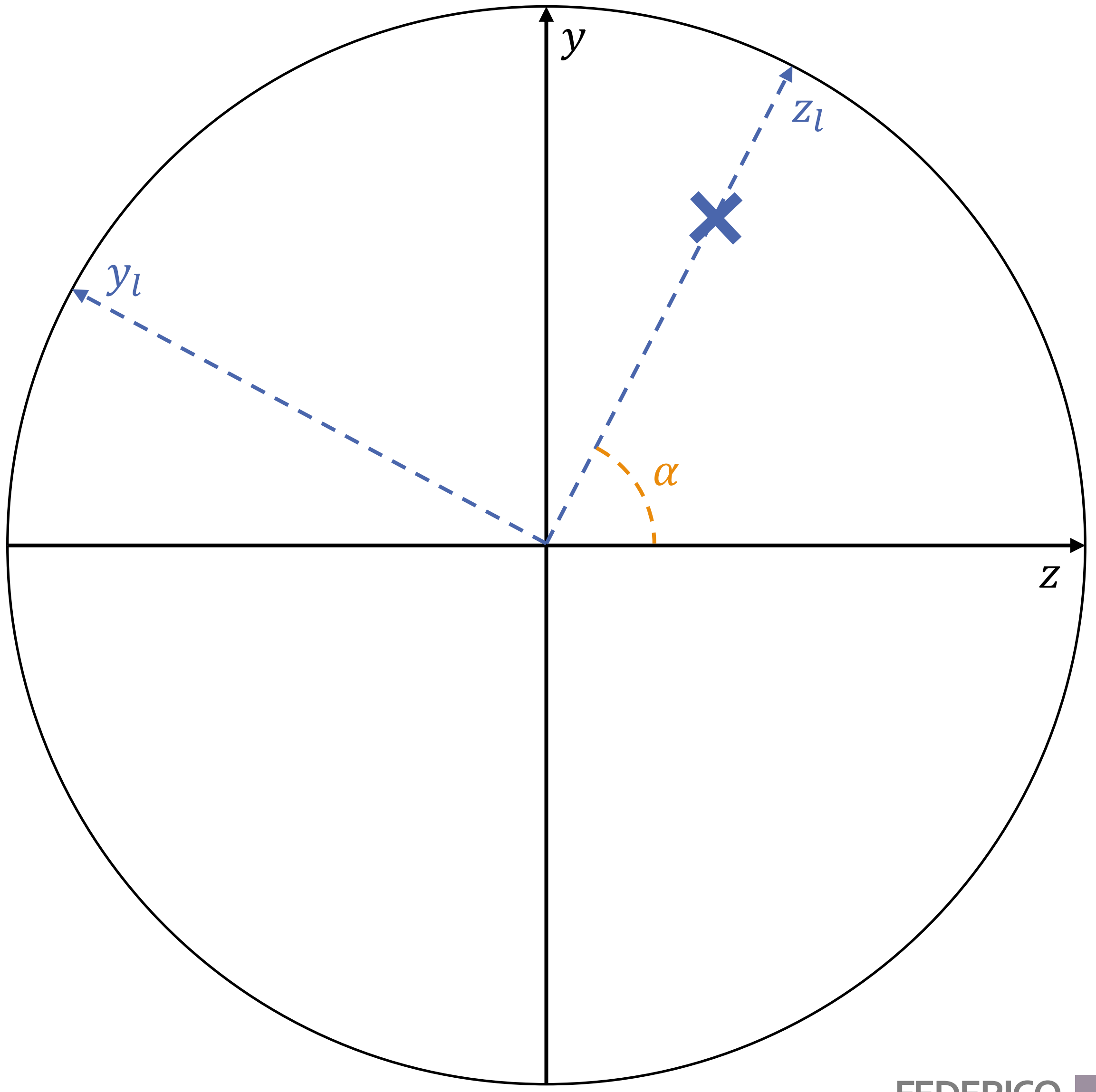
- Use parametrization used in ALICE: state vector updated by the Kalman filter is $s = (y, x, sin\phi, \tan\lambda, \frac{q}{p_T})$

- ALICE uses no approximations in the propagation, unlike current ND-GAr model which uses small angle approximation (for full description check back-up and first ND-GAr-Lite presentation https://indico.fnal.gov/event/50215/contributions/232480/)



KALMAN FILTER MODEL AND APPLICATION

- Use parametrization used in ALICE: state vector updated by the Kalman filter is $s = (y, x, sin\phi, \tan\lambda, \frac{q}{p_T})$
- ALICE uses no approximations in the propagation, unlike current ND-GAr model which uses small angle approximation (for full description check back-up and first ND-GAr-Lite presentation https://indico.fnal.gov/event/50215/contributions/232480/)
- Kalman filter propagated radially: before each propagation, the coordinate system is rotated by an angle $\alpha = \tan(y/z)$, so that the track point "sits" on the local z axis (i.e. z coordinate becomes the radius from center of the detector)



KALMAN FILTER MODEL AND APPLICATION

- Local $sin\phi$ defines two yz semi-planes with "mirrored representations": the line separating the two is the one connecting the center of the detector and the center of curvature of the track

- As the track approaches one of the two semi-planes, $sin\phi$ reaches a point where it cannot be propagated further: $\sin\phi \in [-1,1]$

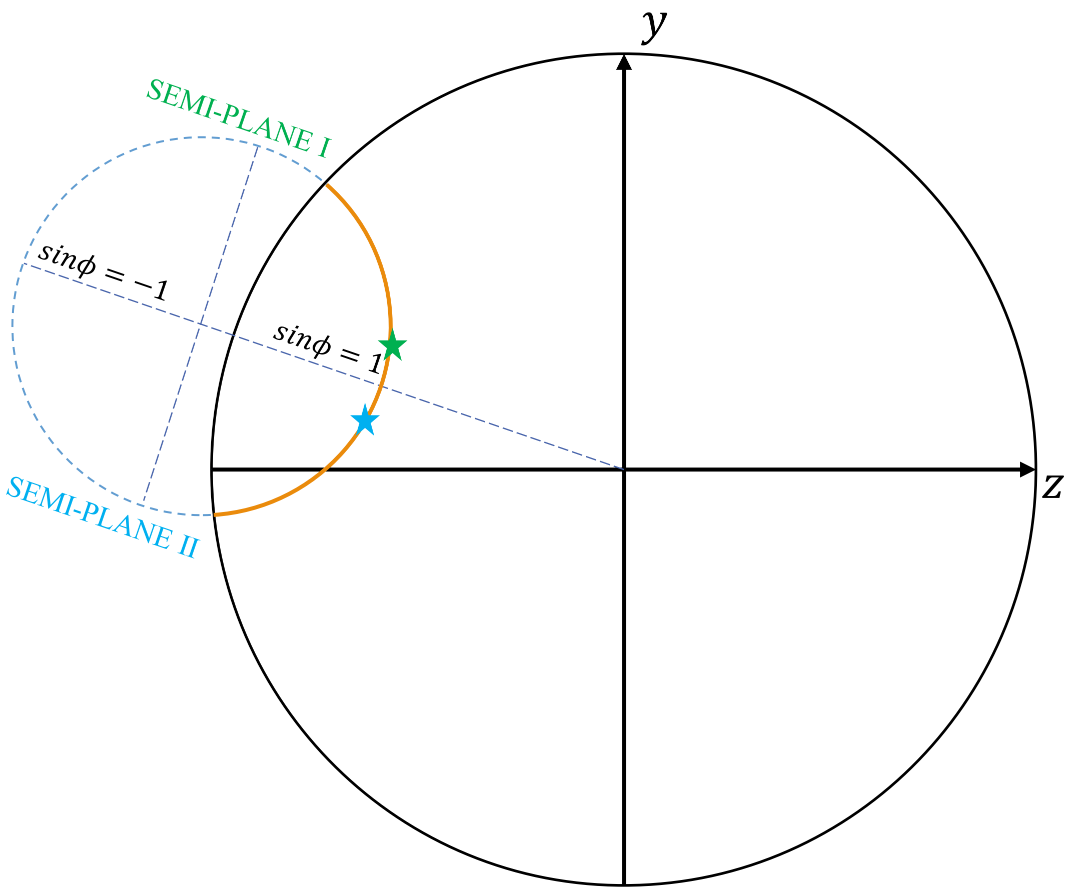
- Once the limit is reached, the state-vector and Covariance associated with the last reconstructed track point are "mirrored":

$$s_{k+1}^{-} = R s_k^{+} \qquad P_{k+1}^{-} = R P_k^{+} R^T$$

$$with\ R = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

- Finally, the local x coordinate is propagated by calculating the arch between the two mirrored points:

$$x_{k+1}^{-} = x_k^{+} + arch * tan\lambda$$



ENERGY LOSS CORRECTION

Bethe-Bloch (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\frac{1}{\rho}\frac{dE}{dx} = K \times \frac{Z}{A} \times \frac{z^2}{\beta^2}\left[\frac{1}{2} ln\left(\frac{2m_e\gamma^2\beta^2T_{max}}{I^2}\right) - \beta^2 - \frac{\delta}{2}\right] \quad [\mathrm{GeV/(g/cm^{\wedge}2)}]$$

- Energy loss correction applied to helix fit:

  1. Get dE/dx with Bethe-Bloch and evaluate momentum loss over trajectory in small "momentum-loss" steps
  2. Calculate multiplicative factor to update q/p_T:

$$\frac{q}{p_T} *= cP4 = \left(1 + \frac{\Delta E}{p_{mean}^2}(\Delta E + 2 \times E_{in})\right)$$

  2. Add factor to diagonal element of 5x5 Covariance Matrix P correspondent to q/p_T (found through error propagation):

$$P[4][4] += \left(\frac{\sigma_E}{p_{mean}^2} \times \frac{q}{p_T}\right)^2$$

- Note 1: These formulas are the same as the ones used by Geant4
- Note 2: Applied to both Kalman Filter "step-by-step" and Seeding "globally"

MS CORRECTION

Molière Formula (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\theta_0 = \frac{13.6MeV}{\beta p} z\sqrt{x/X_0}[1 + 0.038\ln(x/X_0)]$$

- Multiple Scattering correction applied to Helix fit:

  1. Calculate width of the angular gaussian distribution produced by MS: θ_0 from Molière formula
  2. Propagate the error to the relevant Helix parameters, obtaining their respective σ's ($\sigma_{sin\phi}, \sigma_{tan\lambda}, \sigma_{q/p_T}$)
  3. Update covariance matrix diagonal elements:

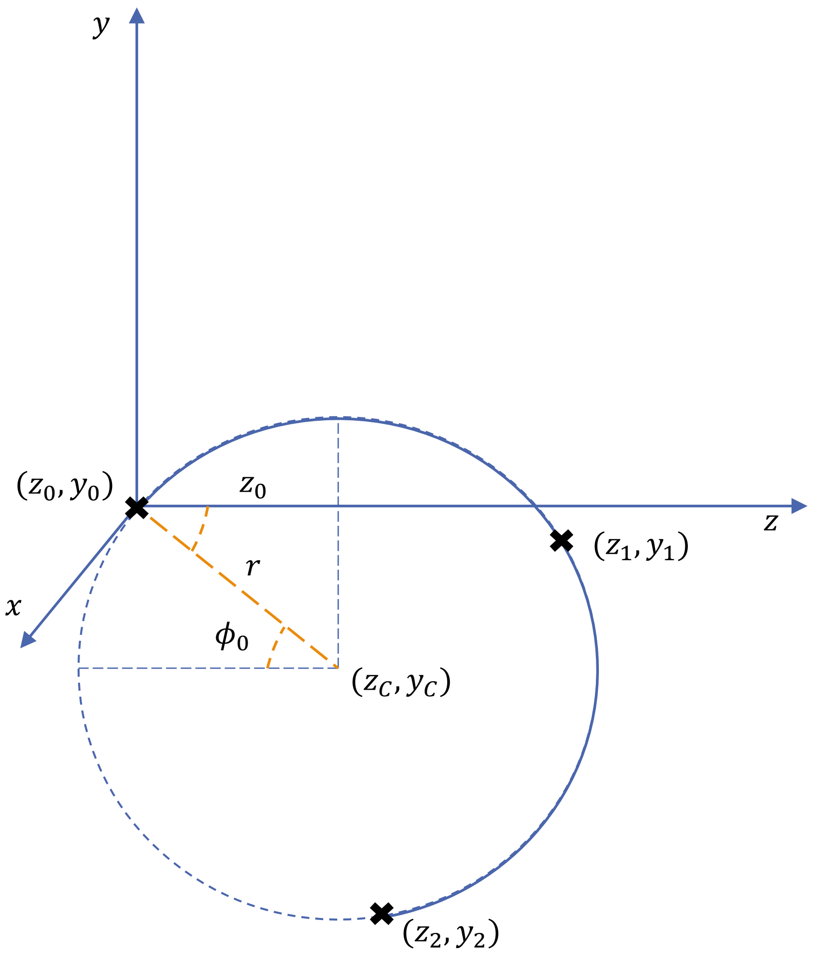
$$\begin{cases} P[2][2] \mathrel{+}= \sigma^2_{sin\phi} \\ P[3][3] \mathrel{+}= \sigma^2_{\tan\lambda} \\ P[4][4] \mathrel{+}= \sigma^2_{q/p_T} \end{cases}$$

- Note 1: These formulas are the same as the ones used by Geant4
- Note 2: Applied to both Kalman Filter "step-by-step" and Seeding "globally"

GLOBAL HELIX FIT AND INITIAL COVARIANCE ESTIMATION

- Seeding for Kalman done with simple 3-point helix fit:
  - $c = 1/r$ and $\sin \phi_0$ estimated by finding (z_c, y_C) and r of the yz plane circumference:

$$\boxed{c = 1/r} \qquad \boxed{\sin \phi_0 = \frac{z_0}{r}}$$



GLOBAL HELIX FIT AND INITIAL COVARIANCE ESTIMATION

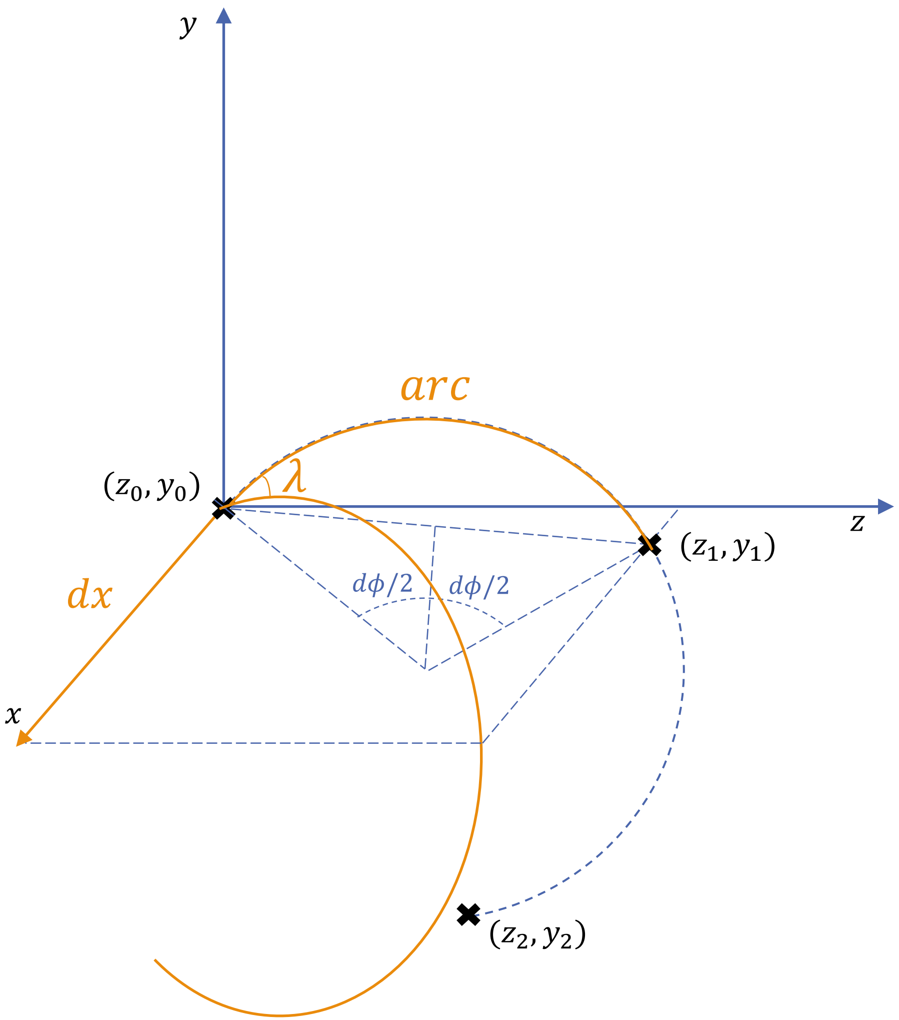
- Seeding for Kalman done with simple 3-point helix fit:
  - $c = 1/r$ and $\sin\phi_0$ estimated by finding (z_c, y_c) and r of the yz plane circumference:

$$\boxed{c = 1/r} \qquad \boxed{\sin\phi_0 = \frac{z_0}{r}}$$

  - $\tan\lambda$ from the yz plane arc between the first two points and the correspondent movement in the x direction:

$$\boxed{\tan\lambda = \frac{dx}{arc} = \frac{dx}{d\phi * r}}$$

- Note: Energy loss and MS corrections applied similarly to Kalman Filter



TESTS AND RESULTS

	Toy-MC			Helix			Kalman Filter		
	Point Smear	dE/dx	MS	Helix Result	dE/dx Corr	MS Corr	Helix Seed	dE/dx Corr	MS Corr
0.5	✓ $\sigma_{yz} = 0.1cm$			✓					
0.5a	✓ $\sigma_{yz} = 0.1cm$						✓		
1.5	✓ $\sigma_{yz} = 0.1cm$	✓		✓					
1.5a	✓ $\sigma_{yz} = 0.1cm$	✓					✓		
1.5b	✓ $\sigma_{yz} = 0.1cm$	✓					✓	✓	
1.5.1	✓ $\sigma_{yz} = 0.1cm$	✓		✓	✓				
1.5.1a	✓ $\sigma_{yz} = 0.1cm$	✓			✓		✓		
1.5.1b	✓ $\sigma_{yz} = 0.1cm$	✓			✓		✓	✓	
2.5	✓ $\sigma_{yz} = 0.1cm$		✓	✓					
2.5a	✓ $\sigma_{yz} = 0.1cm$		✓				✓		
2.5d	✓ $\sigma_{yz} = 0.1cm$		✓				✓		✓
2.5.3	✓ $\sigma_{yz} = 0.1cm$		✓	✓		✓			
2.5.3a	✓ $\sigma_{yz} = 0.1cm$		✓			✓	✓		
2.5.3d	✓ $\sigma_{yz} = 0.1cm$		✓			✓	✓		✓
3.5	✓ $\sigma_{yz} = 0.1cm$	✓	✓	✓					
3.5a	✓ $\sigma_{yz} = 0.1cm$	✓	✓				✓		
3.5b	✓ $\sigma_{yz} = 0.1cm$	✓	✓				✓	✓	
3.5c	✓ $\sigma_{yz} = 0.1cm$	✓	✓				✓	✓	✓
3.5d	✓ $\sigma_{yz} = 0.1cm$	✓	✓				✓		✓
3.5	✓ $\sigma_{yz} = 0.1cm$	✓	✓	✓	✓				
3.5a	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓		✓		
3.5b	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓		✓	✓	
3.5c	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓		✓	✓	✓
3.5d	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓		✓		✓
3.5.2	✓ $\sigma_{yz} = 0.1cm$	✓	✓	✓	✓	✓			
3.5.2a	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓	✓	✓		
3.5.2b	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓	✓	✓	✓	
3.5.2c	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓	✓	✓	✓	✓
3.5.2d	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓	✓	✓		✓

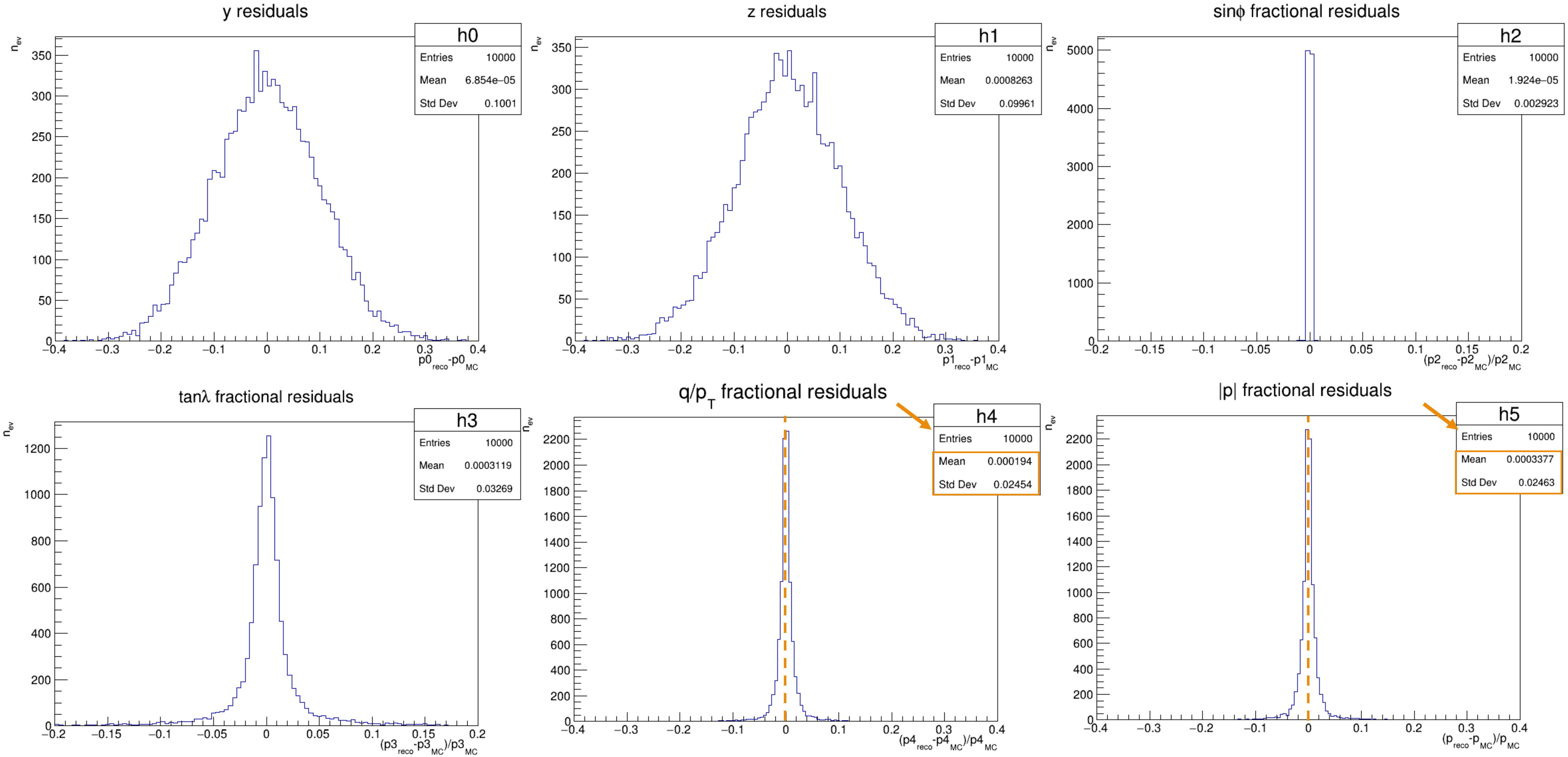
- Test naming convention:
  - n = Kalman Filter using ideal seed
  - $n.5$ = ALICE 3-points Helix Fit
  - $n.x.1$ = Helix Fit E-loss correction
  - $n.x.2$ = Helix Fit E−loss+MS correction
  - $n.x.3$ = Helix Fit MS correction
  - $n.x.y\ a$ = Kalman Filter using Helix Seed
  - $n.x.y\ b$ = Kalman Filter + E-loss correction using Helix Seed
  - $n.x.y\ c$ = Kalman Filter + E-loss + MS corrections using Helix Seed
  - $n.x.y\ d$ = Kalman Filter+ MS corrections using Helix Seed
- Note: Same ND-GAr-Lite sample used for all the tests; For different n we have different Toy Monte Carlo set-ups (E-loss, MS etc.)

TEST 0.5

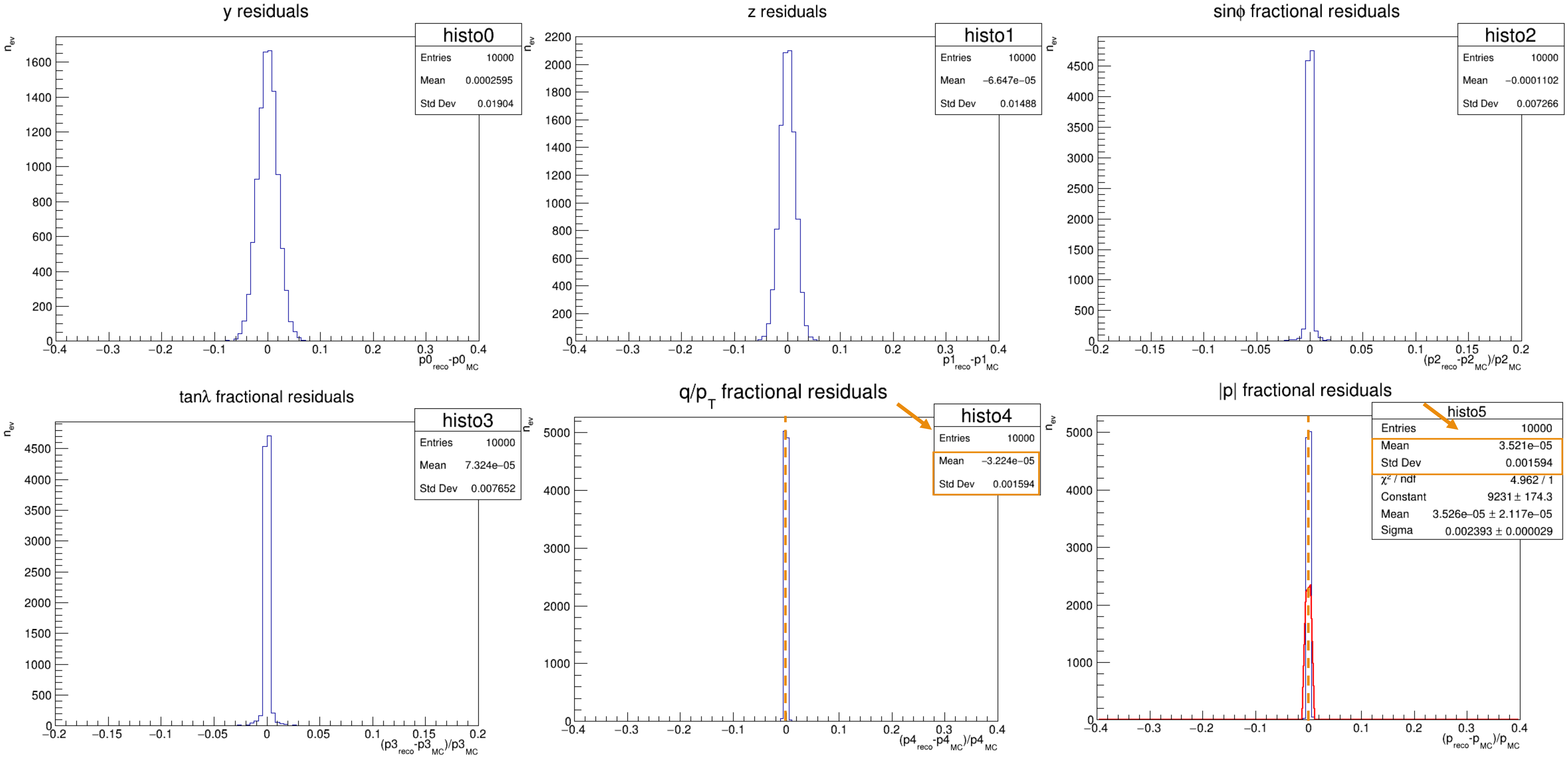
	Toy-MC			Helix			Kalman Filter		
	Point Smear	dE/dx	MS	Helix Result	dE/dx Corr	MS Corr	Helix Seed	dE/dx Corr	MS Corr
0.5	✓ $\sigma_{yz} = 0.1cm$			✓					
0.5a	✓ $\sigma_{yz} = 0.1cm$						✓		

- Compare results in terms of fractional residuals for the helix parameters $(y, x, sin\phi, \tan\lambda, \frac{q}{p_T})$ and the total momentum p and check that the Covariance Matrix describes the sample
- For this set of tests, no energy loss nor multiple scattering are simulated in the Toy Monte Carlo and a gaussian smearing $\sigma_{xy} = 0.1cm$ is applied to the points
- Compare 2 reconstruction results:
  - Simple ALICE 3-point method with no corrections (Test 0.5)
  - Kalman Filter applied over simple ALICE 3-point method with no corrections in either (Test 0.5a)

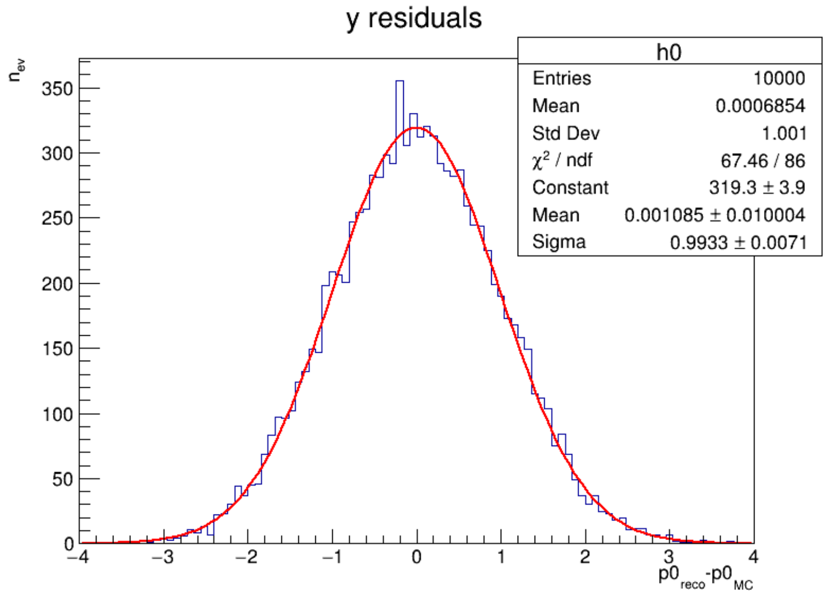
TEST 0.5: HELIX FIT

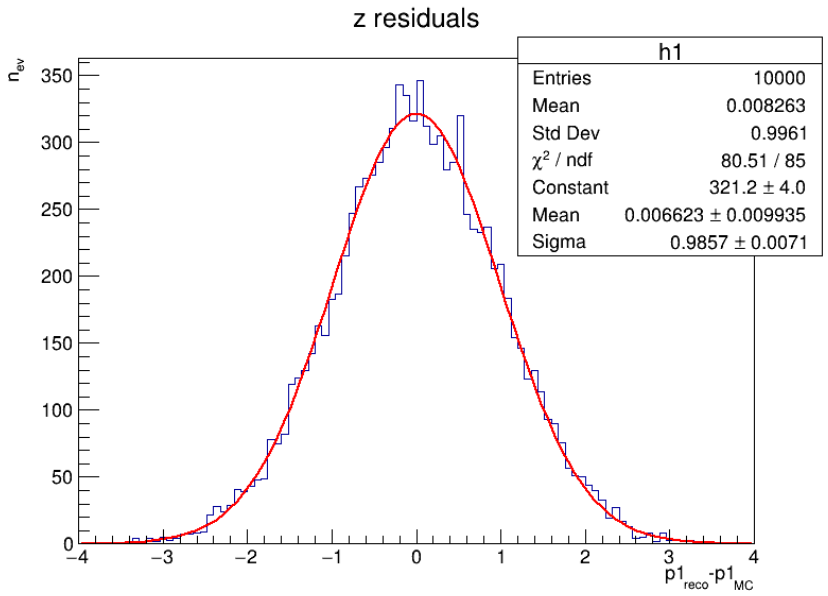


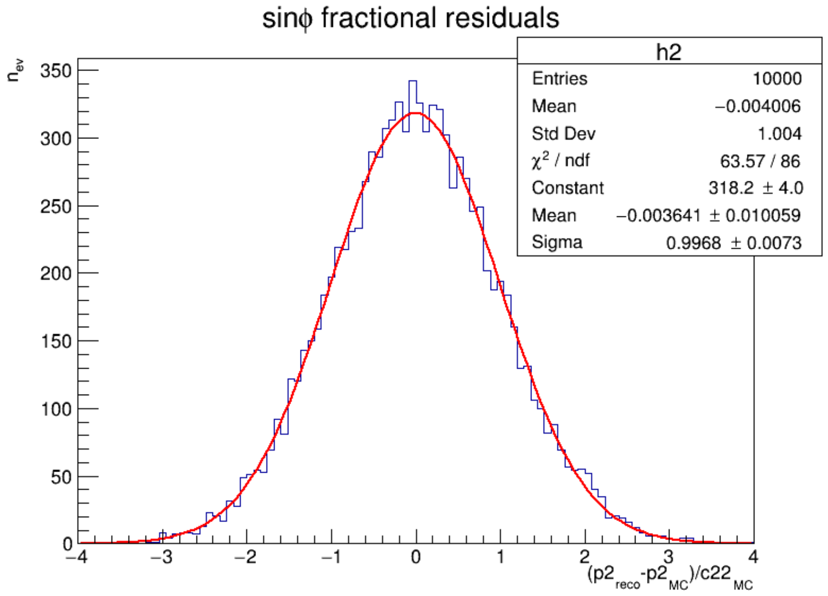
TEST 0.5A: KALMAN FILTER



PULL TEST: HELIX



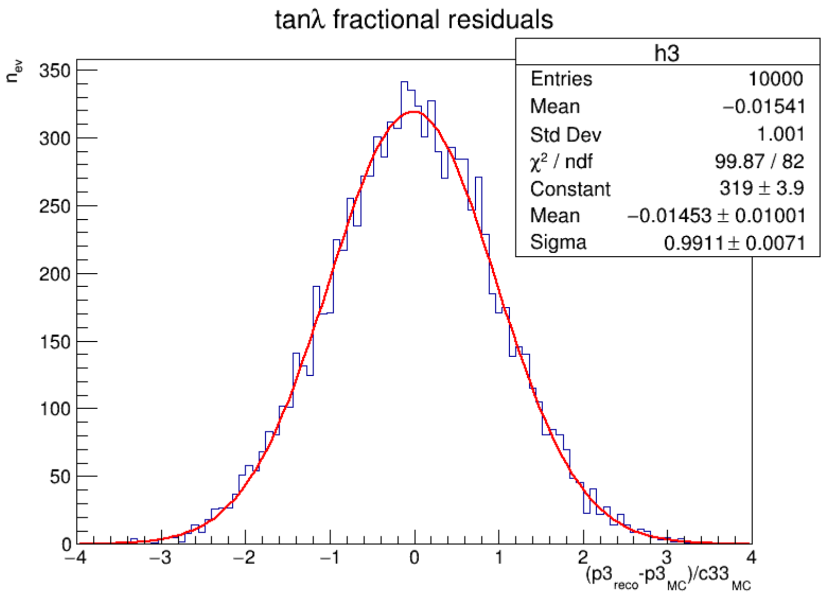


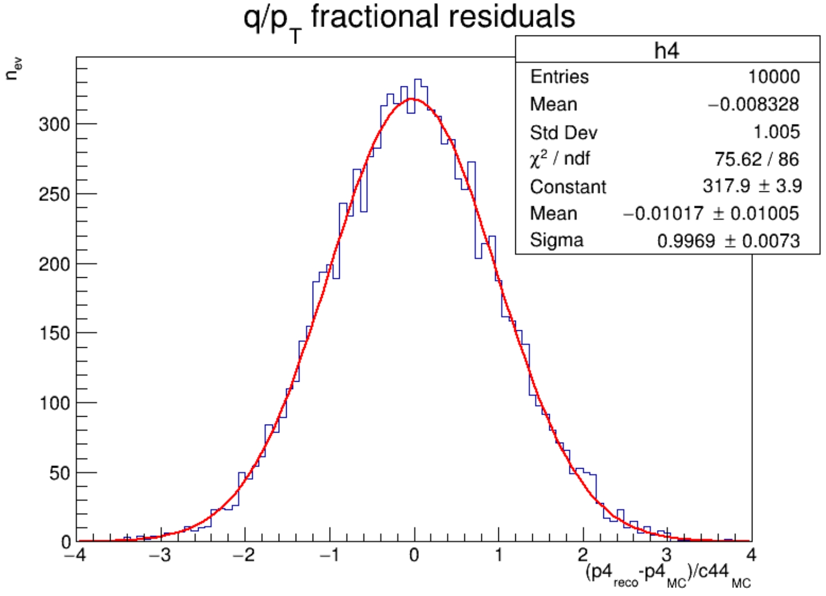


- To check the quality of the estimation of the covariance matrix we perform a Pull Test

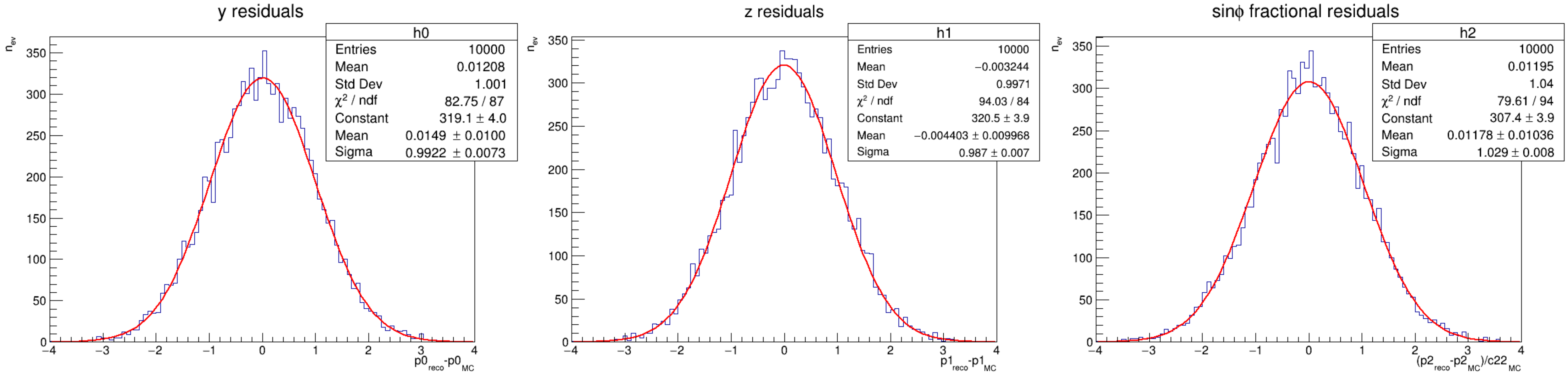
- Pull test: residuals of parameters divided by the square-root of the correspondent diagonal matrix, should form a Gauss distribution with $\sigma \sim 1$

- Helix seed estimates uncertainties effectively for all 5 parameters

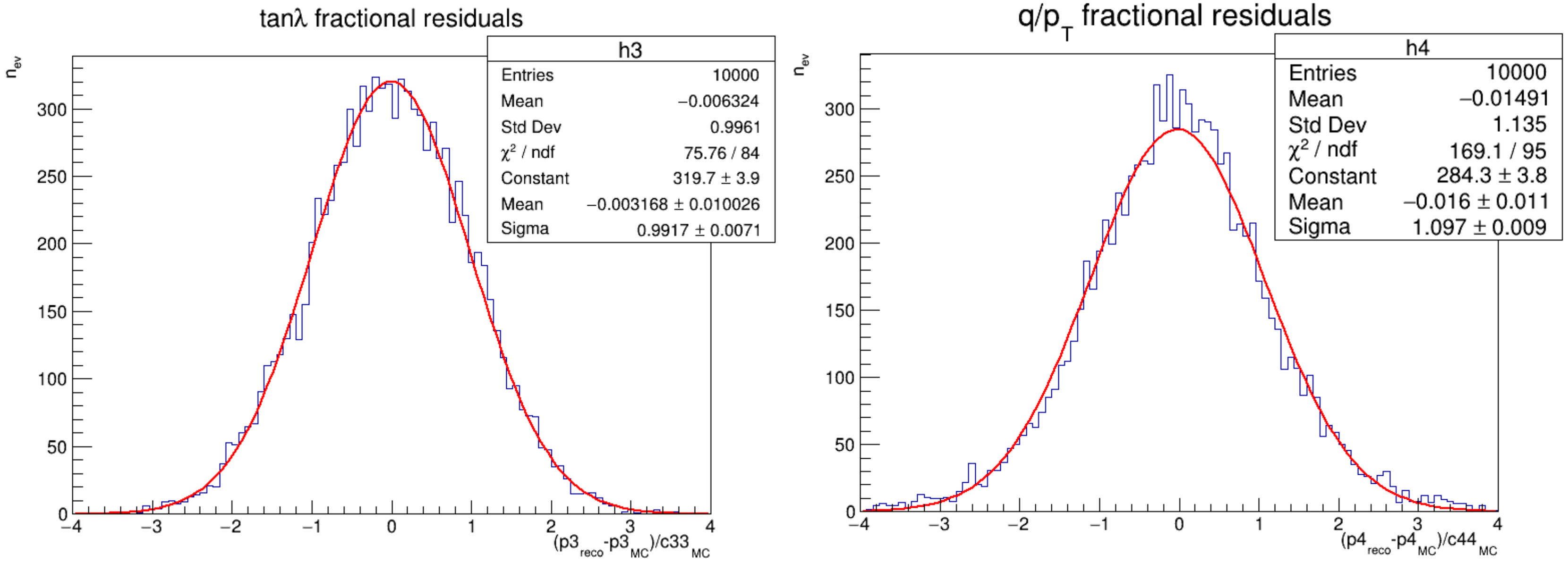




PULL TEST: KALMAN FILTER AT END OF RECO



- To check the quality of the estimation of the covariance matrix we perform a Pull Test
- Pull test: residuals of parameters divided by the square-root of the correspondent diagonal matrix, should form a Gauss distribution with $\sigma \sim 1$
- Kalman Filter propagates uncertainties effectively for all 5 parameters



MAHALANOBIS DISTANCE TEST

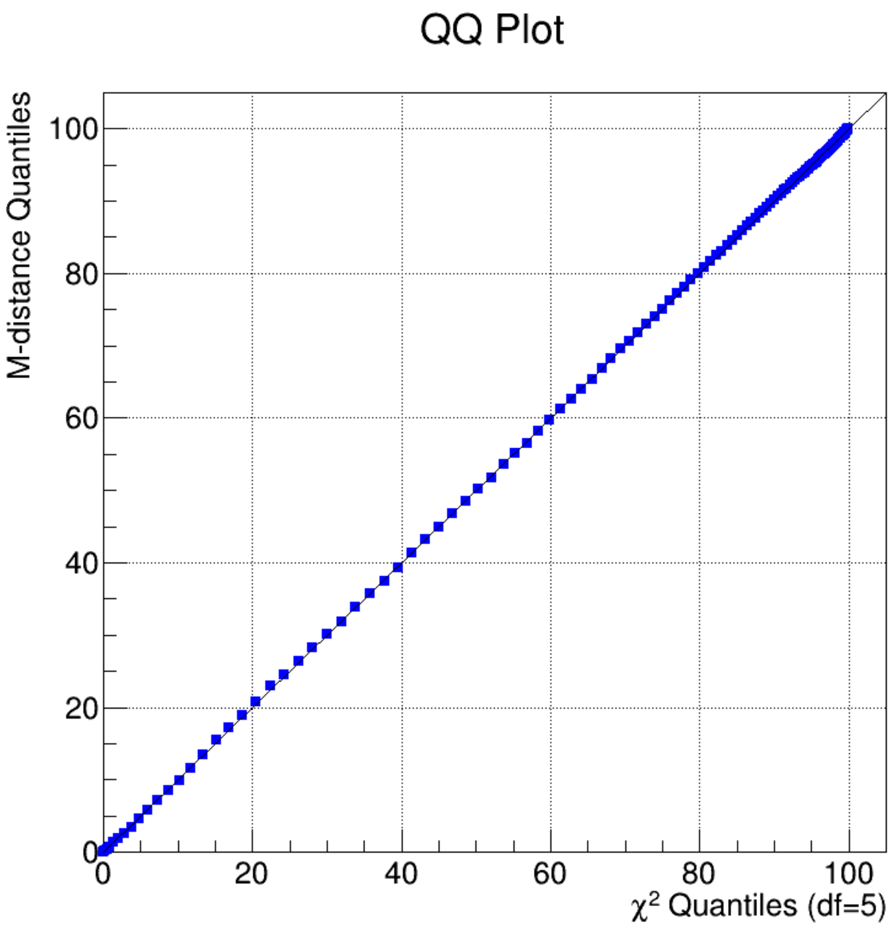
- Given a probability distribution Q on R^N with mean $\vec{\mu}$ and positive-definite covariance matrix P the Mahalanobis distance (M-Distance) of a point $\vec{s}$ from Q is defined as:

$$d_M = \sqrt{(\vec{s} - \vec{\mu})^T P (\vec{s} - \vec{\mu})}$$

- The M-Distances of a set of points belonging to the distribution Q will follow a χ^2 distribution with N degrees of freedom

- To check if a covariance matrix of a distribution Q is correctly estimated one can calculate d_M for a certain number of "points" (in our case state vectors of tracks) and check if they follow the correct χ^2 distribution (NB: this checks the whole matrix including correlations, unlike standard Pull-Test. **Thanks to Lukas Koch for the suggestion**)

- Easy way to visualize this is a Quantile VS Quantile (QQ) plot, in our case quantiles of the d_M distribution VS quantiles of the χ^2 distribution: if we get a straight line the estimated Covariance describes the distribution



QQ-plot for 3-point Helix Fit for test 0.5

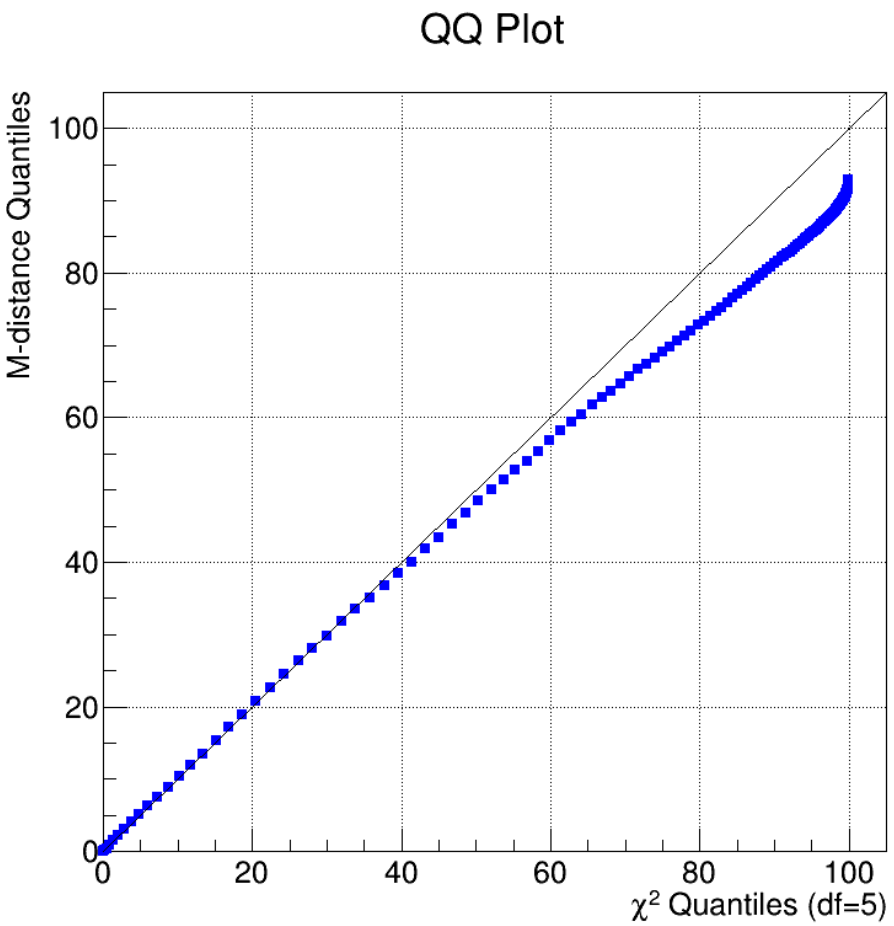
MAHALANOBIS DISTANCE TEST

- Given a probability distribution Q on R^N with mean $\vec{\mu}$ and positive-definite covariance matrix P the Mahalanobis distance (M-Distance) of a point $\vec{s}$ from Q is defined as:
$$d_M = \sqrt{(\vec{s}-\vec{\mu})^T P(\vec{s}-\vec{\mu})}$$

- The M-Distances of a set of points belonging to the distribution Q will follow a χ^2 distribution with N degrees of freedom

- To check if a covariance matrix of a distribution Q is correctly estimated one can calculate d_M for a certain number of "points" (in our case state vectors of tracks) and check if they follow the correct χ^2 distribution (NB: this checks the whole matrix including correlations, unlike standard Pull-Test. **Thanks to Lukas Koch for the suggestion**)

- Easy way to visualize this is a Quantile VS Quantile (QQ) plot, in our case quantiles of the d_M distribution VS quantiles of the χ^2 distribution: if we get a straight line the estimated Covariance describes the distribution



QQ-plot for 3-point Kalman Filter for test 0.5a. Slight under-estimation in the tails probably due to approximations done in "mirror" portion of the tracking

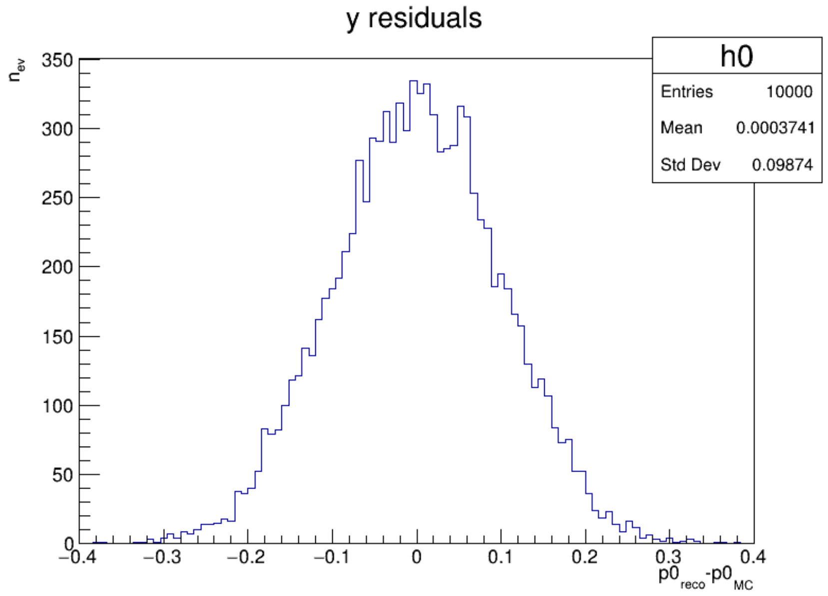
TEST 1.5

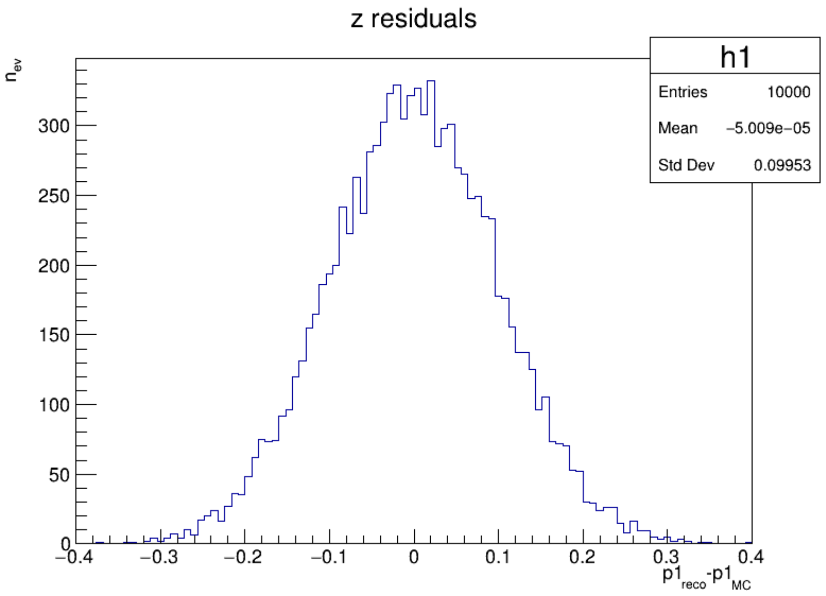
	Toy-MC			Helix			Kalman Filter		
	Point Smear	dE/dx	MS	Helix Result	dE/dx Corr	MS Corr	Helix Seed	dE/dx Corr	MS Corr
1.5.1	✓ $\sigma_{yz} = 0.1cm$	✓		✓	✓				
1.5.1b	✓ $\sigma_{yz} = 0.1cm$	✓			✓		✓	✓	

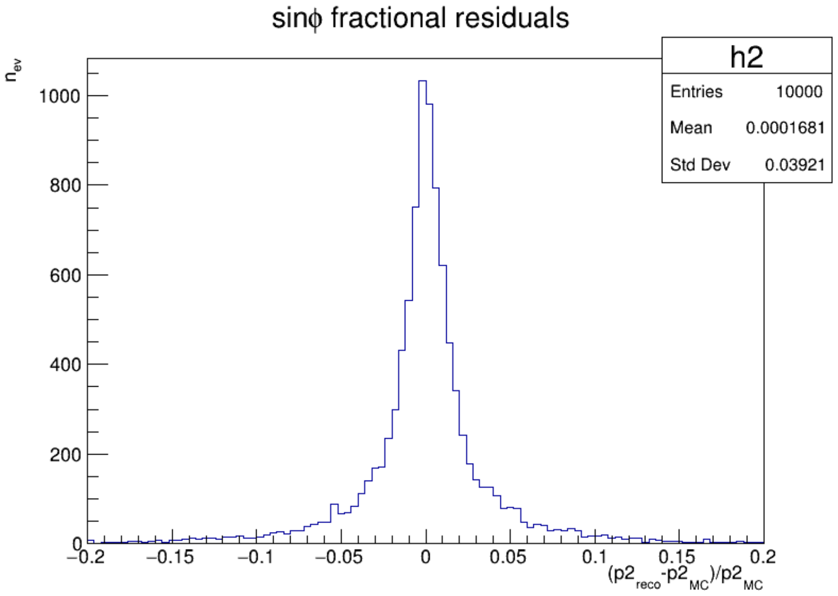
- For this set of tests E-loss is introduced in the Toy MC simulation

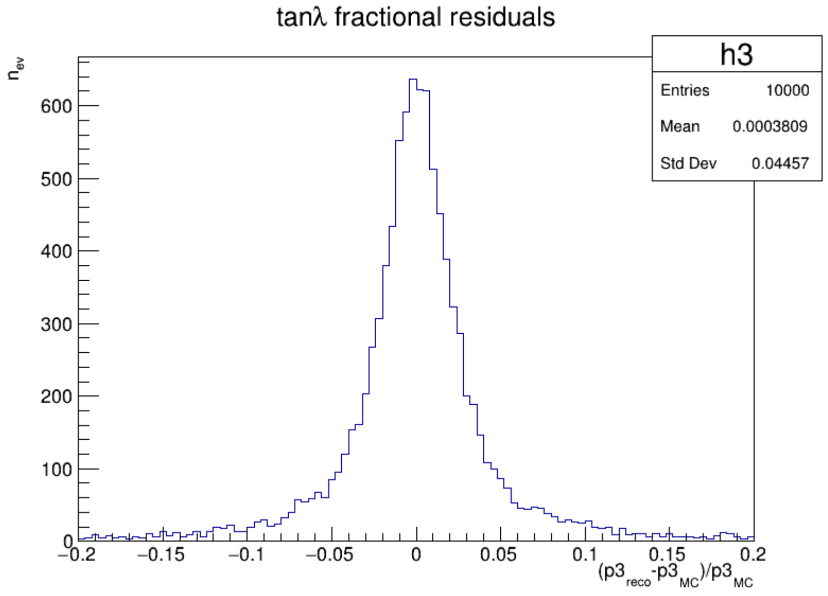
- Compare 2 reconstruction results:
  - Helix Fit with E-Loss corrections (Test 1.5)
  - Kalman Filter applied over simple Helix Fit with E-loss corrections in both (Test 1.5a)

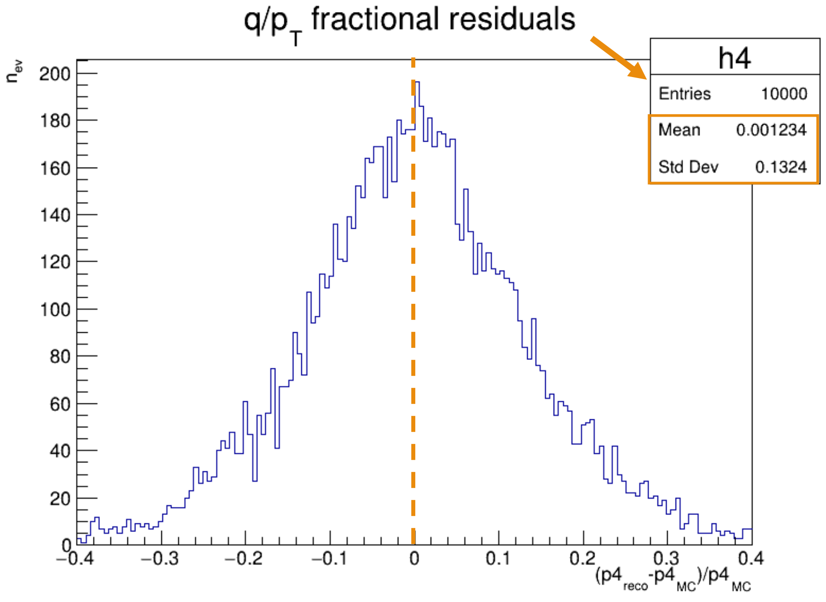
TEST 1.5.1: HELIX FIT+ E-LOSS CORR

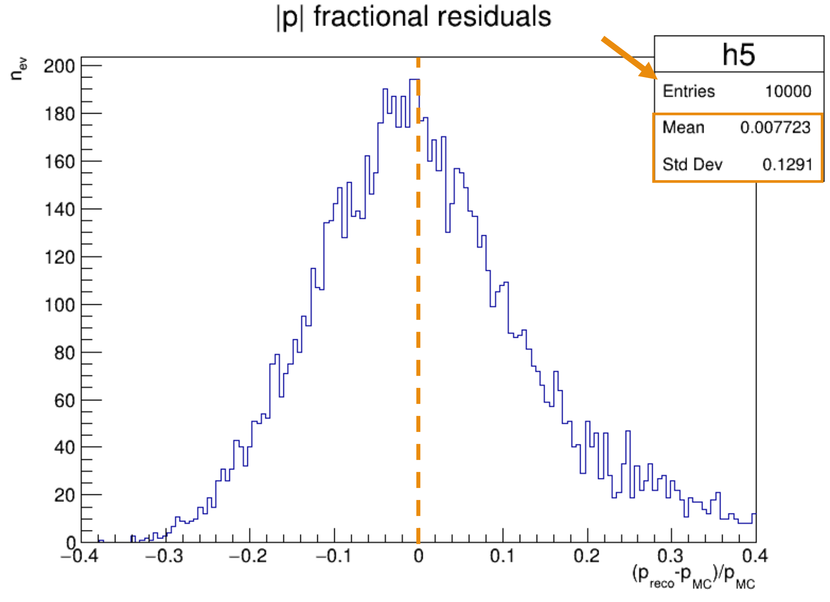




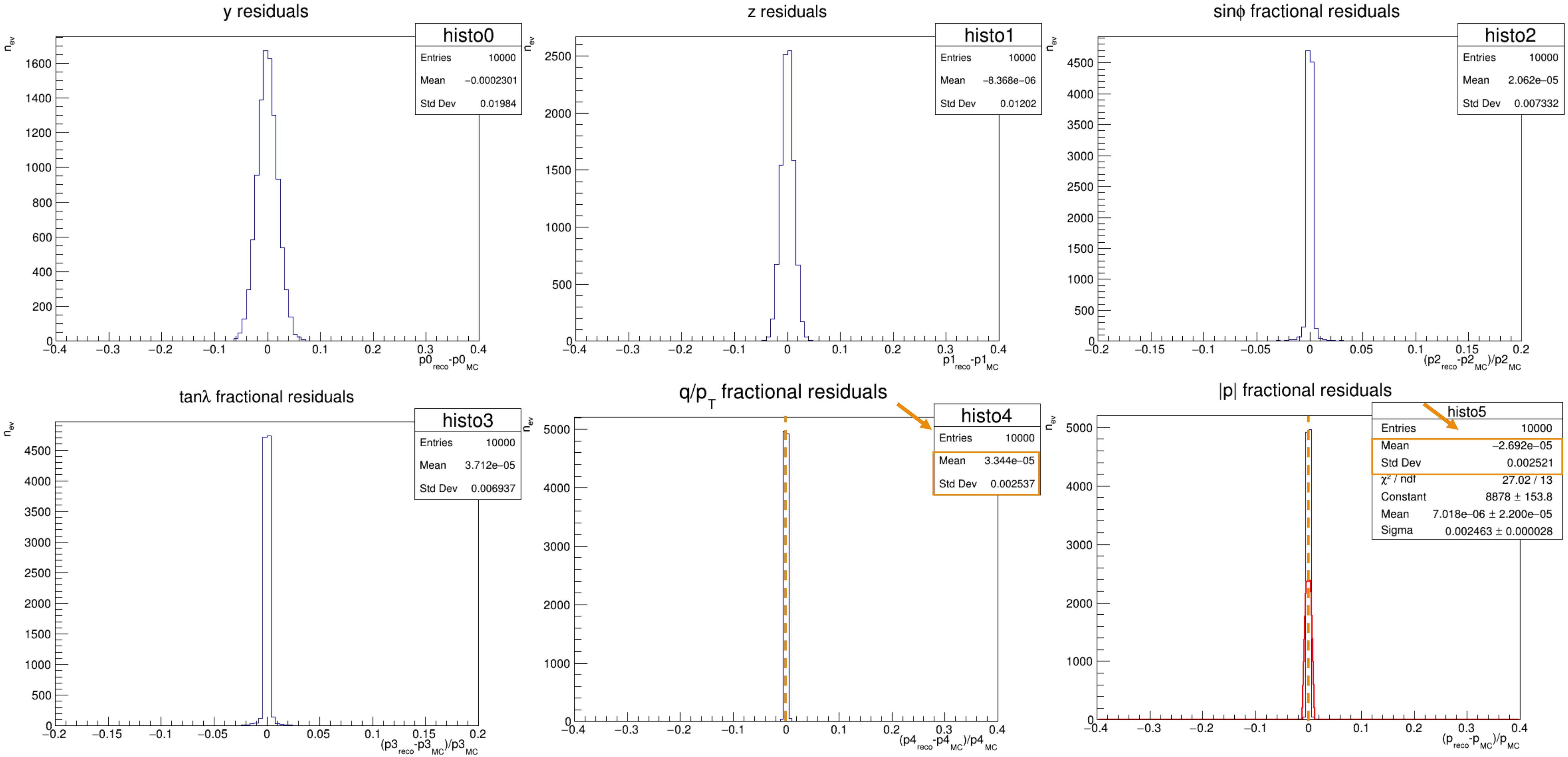








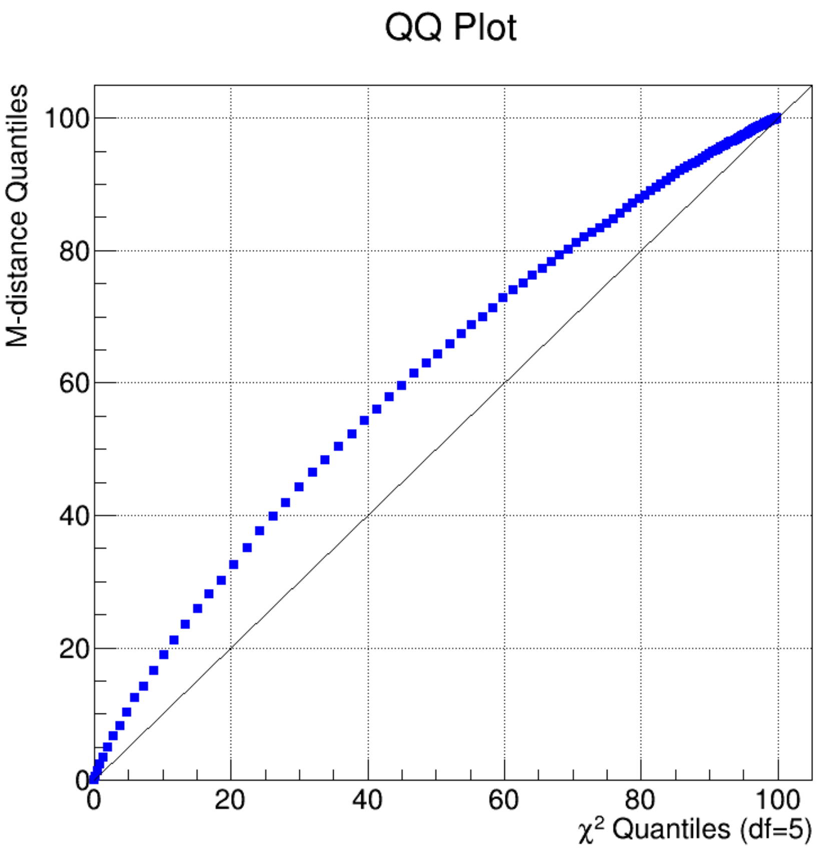
TEST 1.5.1B: KALMAN FILTER+E-LOSS CORRECTION



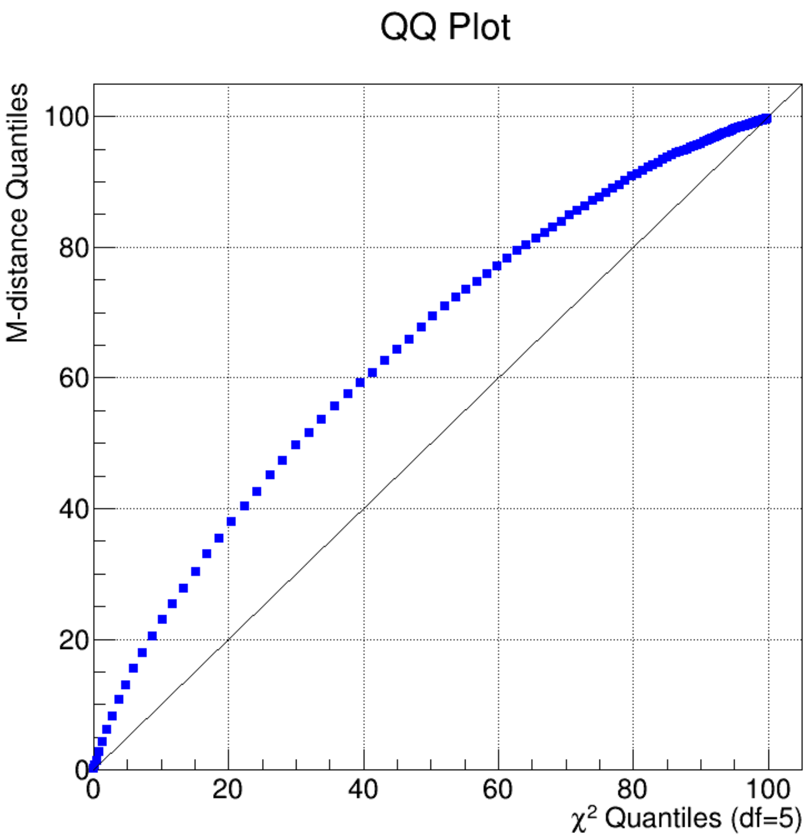
M-DISTANCE TEST: 1.5.1-1.5.1B

- The M-Distance plots for these tests show a slight over-estimation of the errors

- This might be due to implementation of the E-loss correction in the Seeding (probably not in the KF propagation, see next test) or to the approximations made in the mirroring step of the propagation. Further investigation needed



QQ-plot for Helix Fit + E-loss correction Covariance Estimation (Test 1.5.1)



QQ-plot for Kalman Filter + E-loss correction Covariance Estimation (Test 1.5.1b)

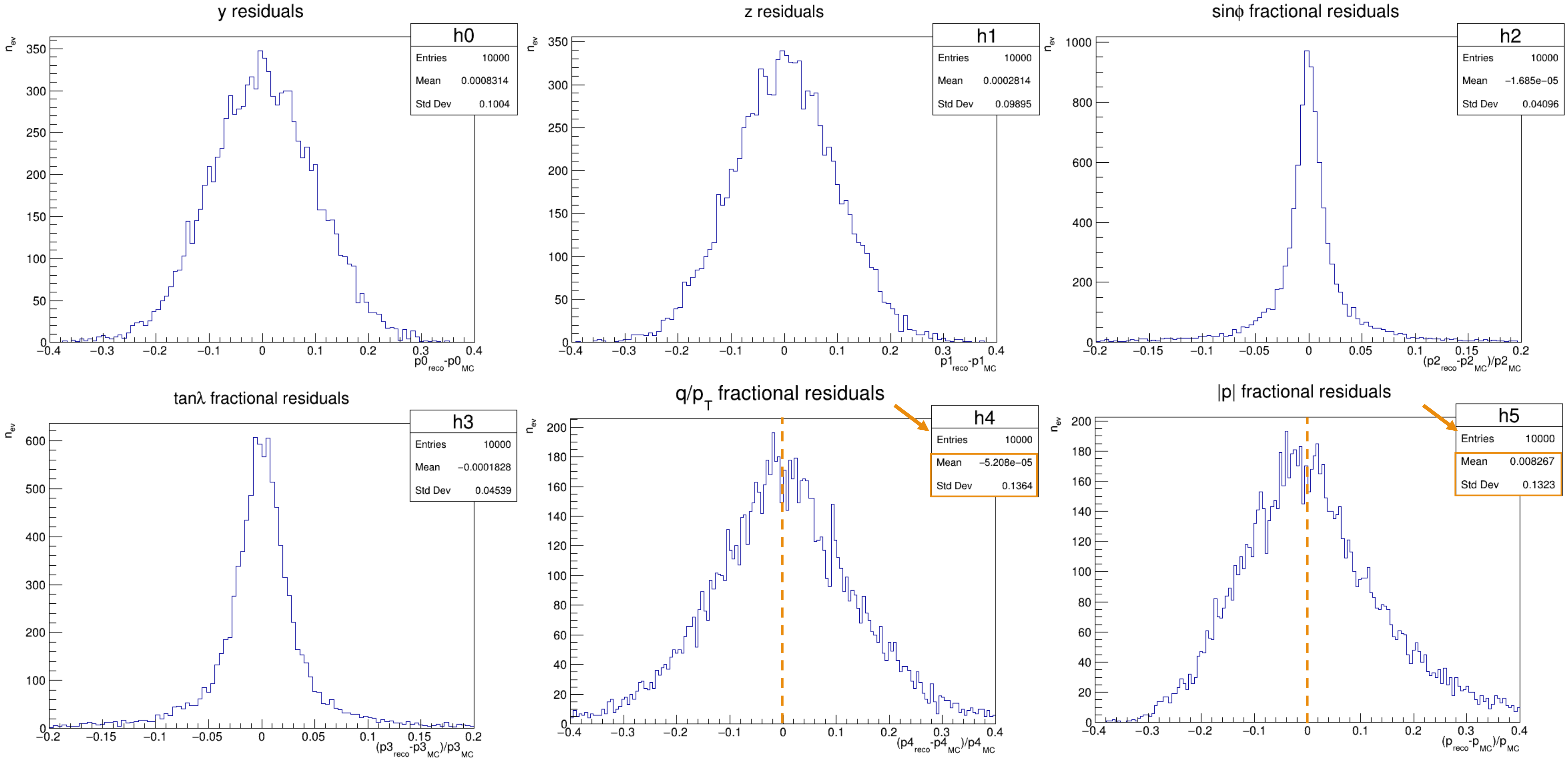
TEST 2.5

	Toy-MC			Helix			Kalman Filter		
	Point Smear	dE/dx	MS	Helix Result	dE/dx Corr	MS Corr	Helix Seed	dE/dx Corr	MS Corr
2.5.1	✓ $\sigma_{yz} = 0.1cm$	✓	✓	✓	✓	✓			✓
2.5.1c	✓ $\sigma_{yz} = 0.1cm$	✓	✓		✓	✓	✓	✓	✓

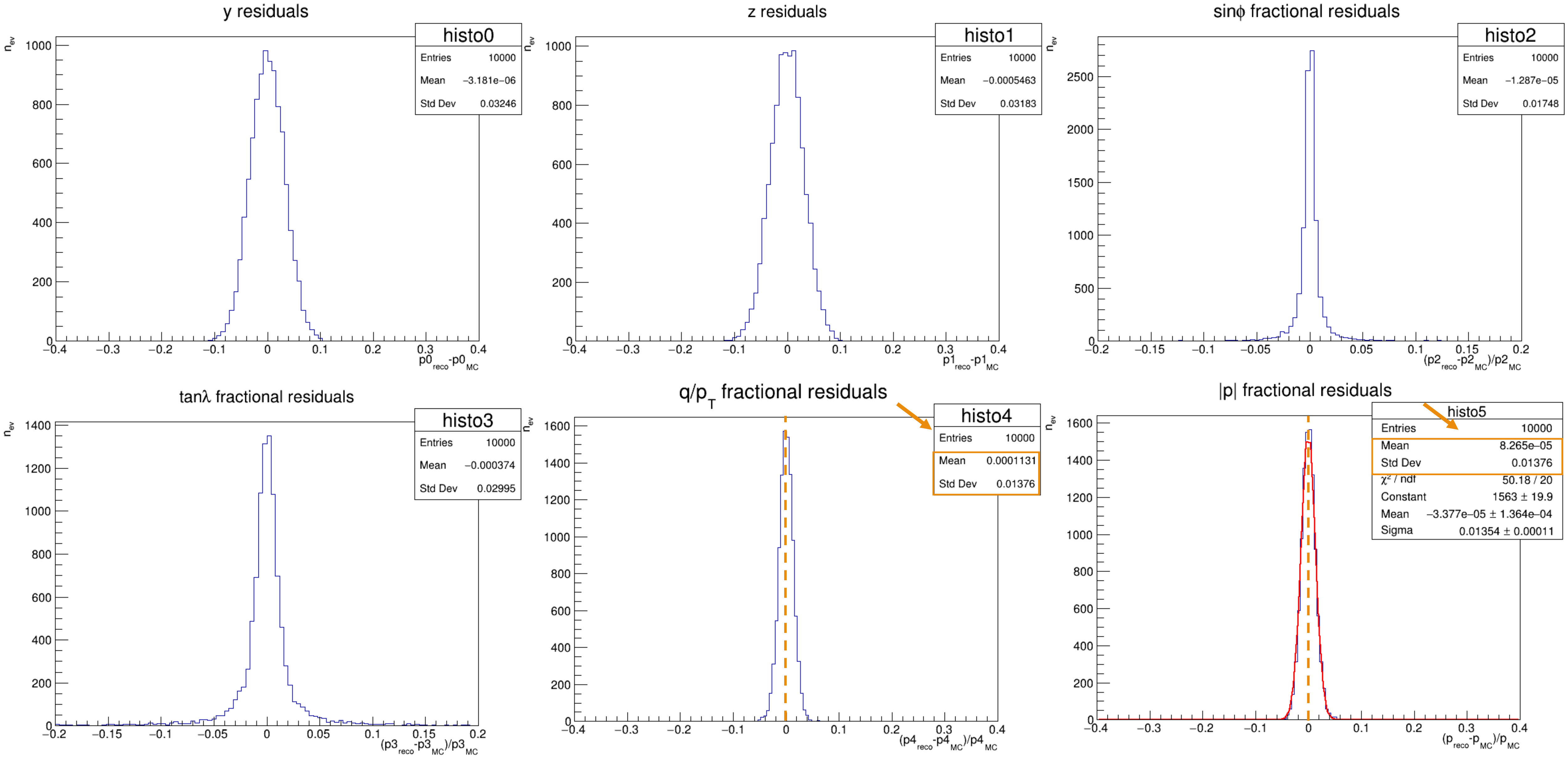
- For this set of tests E-loss and Multiple Scattering are introduced in the Toy MC simulation

- Compare 2 reconstruction results:
    - Helix Fit with E-Loss + MS corrections (Test 1.5)
    - Kalman Filter applied over simple Helix Fit with E-loss + MS corrections in both (Test 1.5a)

TEST 2.5.1: HELIX FIT+ E-LOSS +MS CORR



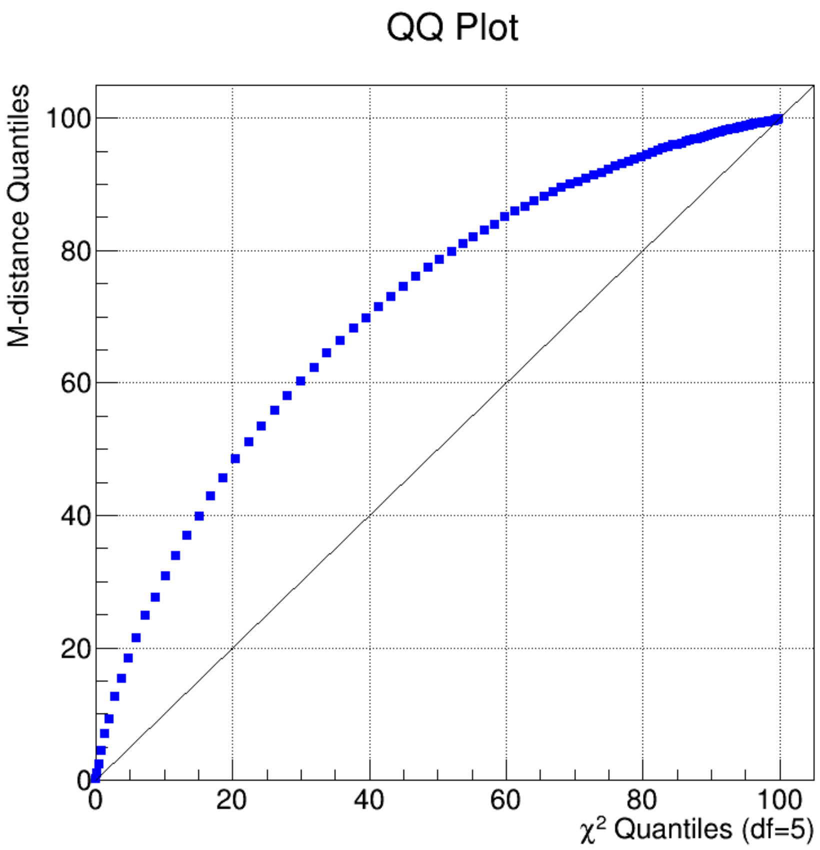
TEST 1.5.1B: KALMAN FILTER+E-LOSS +MS CORRECTION



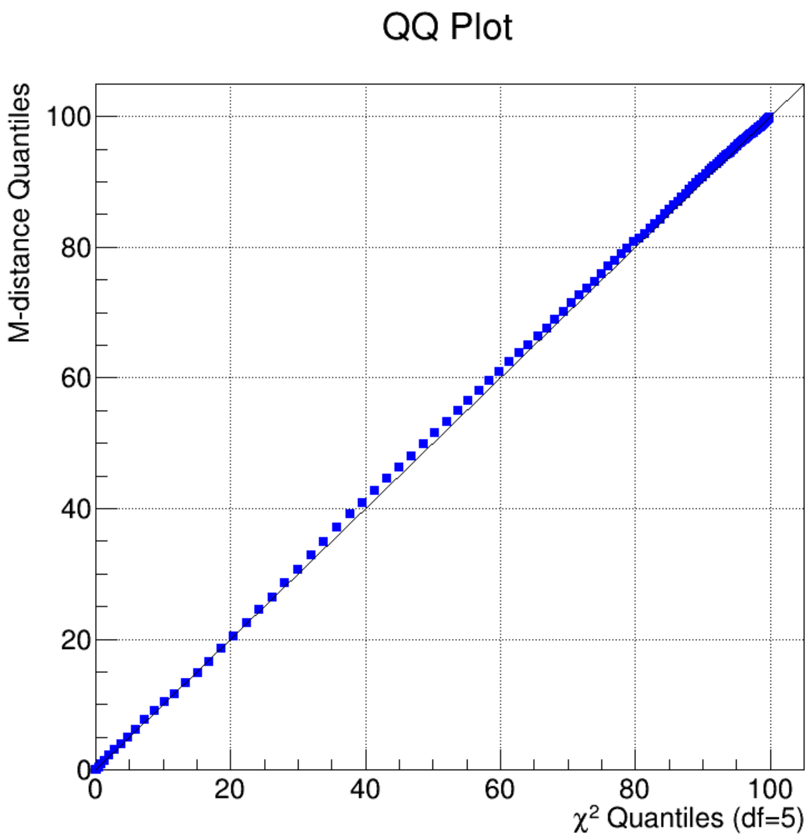
M-DISTANCE TEST: 1.5.1-1.5.1B

- The M-Distance plots for these tests show a significant over-estimation of the errors in the Seeding, that is then smoothed out by the Kalman Filter propagation

- This supports the hypothesis that the problem in the implementation resides with the seeding portion of the algorithm

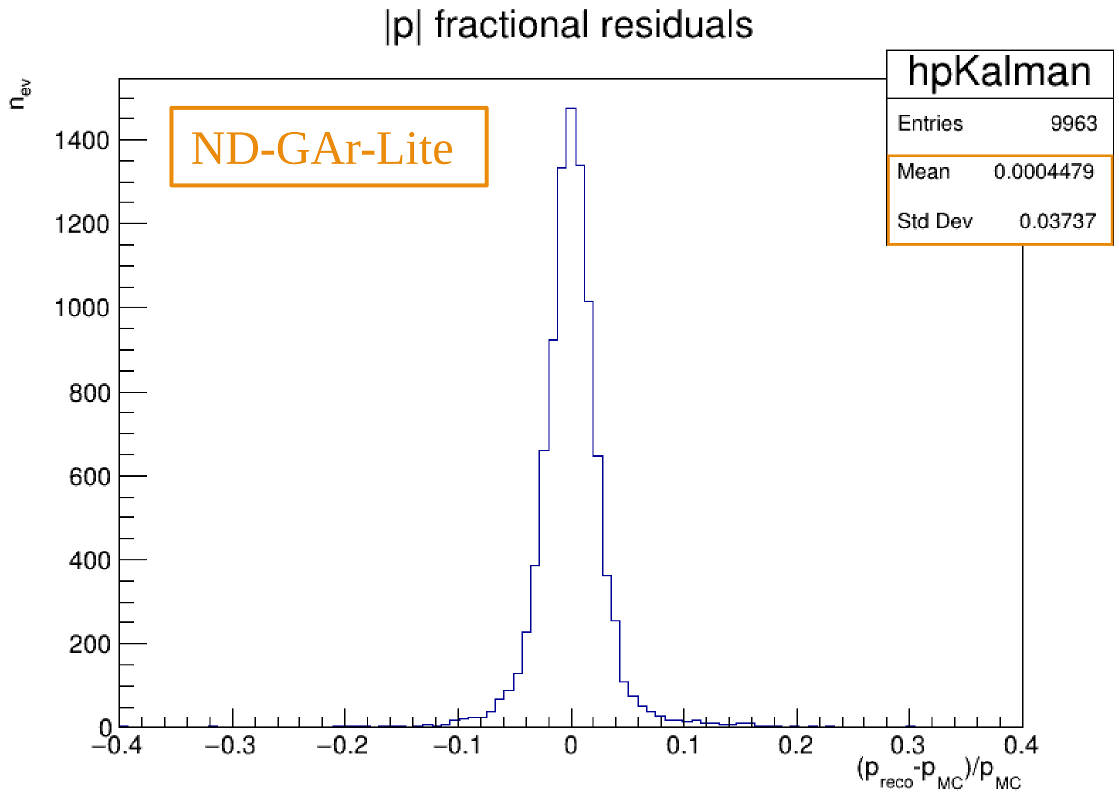


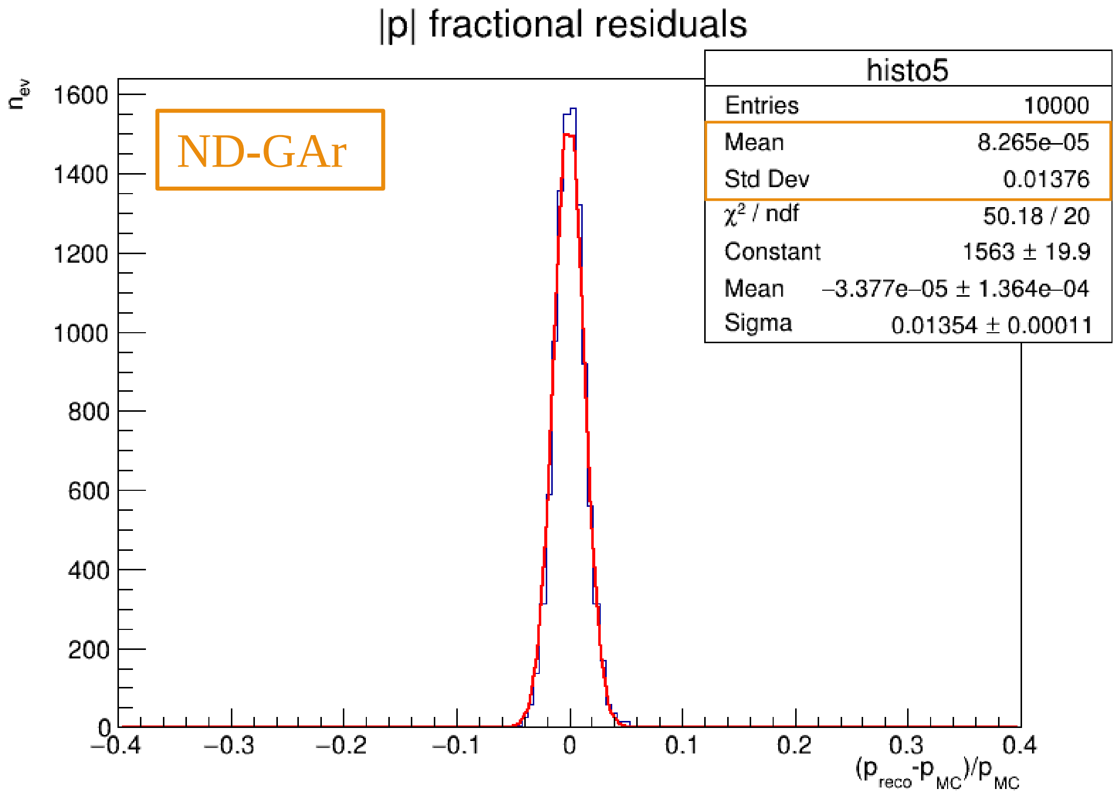
QQ-plot for Helix Fit + E-loss correction Covariance Estimation (Test 1.5.1)



QQ-plot for Kalman Filter + E-loss correction Covariance Estimation (Test 1.5.1b)

COMPARISON WITH ND-GAR-LITE RECONSTRUCTION





- The 10k muon test sample was produced using the same charges, momenta and initial xy positions as the sample analyzed for latest KF study on ND-GAr-Lite (https://indico.fnal.gov/event/55842/)
- As a sanity check we can compare the momentum reconstruction performance found for ND-GAr, with the one found for ND-GAr-Lite: as expected the performance in ND-GAr is significantly improved (resolution spread reduced by a factor of about $\sim$ 2.5 and bias reduced by a factor of $\sim$ 5)
- NB: some events that couldn't be reconstructed in ND-GAr-Lite due to lack of hit points, are instead reconstructed in ND-GAr, but this will have to be checked on a proper MC

SUMMARY AND CONCLUSIONS

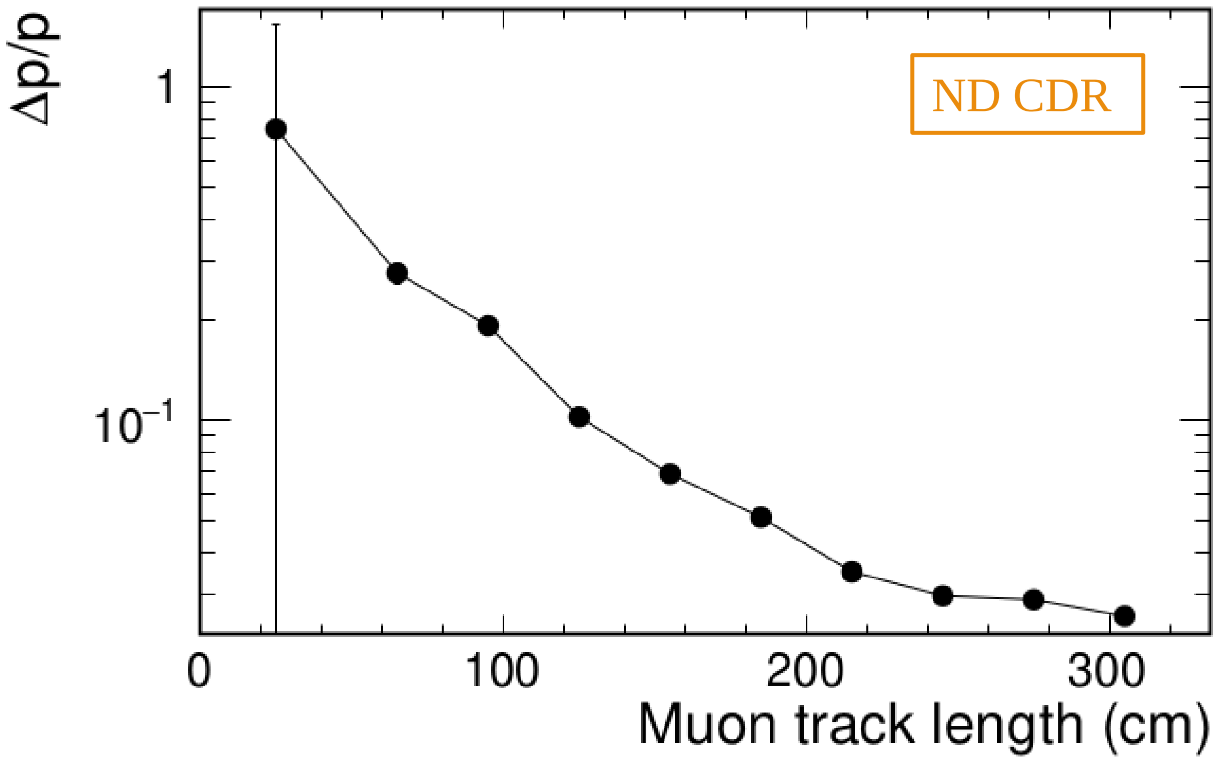
- Introduced a concept for a ALICE-based Kalman Filter for ND-GAr and a toy Monte Carlo tool (fastMCKalman) which allows easy development for reconstruction algorithms in a TPC environment

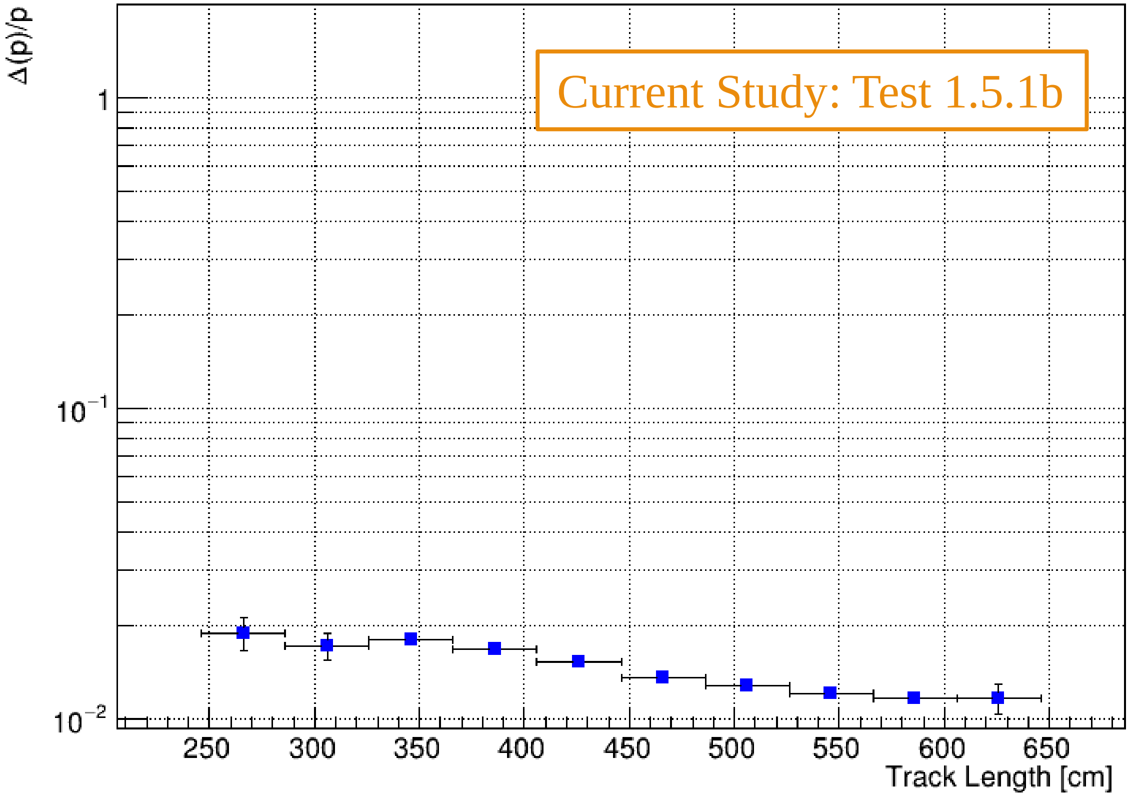
- Main Takeaways:
  1. Toy Monte-Carlo Tests give mostly consistent and encouraging results
  2. Comparisons performed against ND-GAr-Lite show a very significant performance improvement as expected

- Next steps:
  1. Apply Kalman Filter to garsoft Monte Carlo data, ideally particles produced in neutrino on GAr interactions (if you know of trusted samples that already exist, please point me towards it!)
  2. If the testing is successful, implement the new Kalman Filter in garsoft (enable/disable with a fhicl parameter?) and write a technical paper on the full algorithm
  3. Perform physics sensitivity studies using the new algorithm

BACK-UP

COMPARISON WITH ND-GAR CDR





- Direct comparison between the results of this study and the CDR results is not appropriate as the momentum spectra are quite different (Sample for this study is mono-energetic with initial momenta $p = 1GeV/c$, same as the one used in (https://indico.fnal.gov/event/55842/)
- Note: tracks in test sample are consistently about double the length as the ones in CDR, which is unexpected

ENERGY LOSS AND MS

ENERGY LOSS: BETHE-BLOCH FORMULA

Bethe-Bloch (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\frac{1}{\rho}\frac{dE}{dx} = K \times \frac{Z}{A} \times \frac{z^2}{\beta^2}\left[\frac{1}{2} ln\left(\frac{2m_e c^2 \gamma^2 \beta^2 W_{max}}{I^2}\right) - \beta^2 - \frac{\delta}{2}\right] \quad [\mathrm{GeV/(g/}cm^2)]$$

- $\rho = 1.032\ g/cm^3$ — Plastic scintillator density
- $K = 4\pi N_A r_e^2 m_e c^2 = 0.307\ 075\ MeV\ mol^{-1} cm^2$ — Bethe Bloch constant coefficient
- $Z/A = 0.54141\ mol/g$ — Mean atomic number/mass of plastic scintillator
- z — Atomic number of incident particle
- $m_e c^2 = 0.511$ MeV — Mass of electron
- $W_{max} = 2 m_e c^2\ \beta^2 \gamma^2$ — Low energy approximation of maximum energy transfer
- $I = 64.7 \times 10^{-9}\ GeV$ — Mean excitation energy

$$\frac{\delta}{2} = \begin{cases} 0 & ln\beta\gamma < 2.303x_0 \\ \ln\beta\gamma - 1/2\,C & ln\beta\gamma > 2.303x_1 \\ \ln\beta\gamma - 1/2\,C + (1/2\,C - 2.303X_0) \times \left(\frac{2.303X_1 - \ln\beta\gamma}{2.303(X_1 - X_0)}\right)^3 & ln\beta\gamma \in [2.303x_0, 2.303x_1] \end{cases}$$

DENSITY CORRECTION

$$with\ C = 2 - \ln\left(\frac{28.816 \times 10^{-9}\sqrt{\rho(Z/A)}}{I}\right)$$

$x_0 = 0.1469 \qquad x_1 = 2.49$

1st and 2nd junction points for plastic scintillator

ENERGY LOSS CORRECTION

Bethe-Bloch (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\frac{1}{\rho}\frac{dE}{dx} = K \times \frac{Z}{A} \times \frac{z^2}{\beta^2}\left[\frac{1}{2} ln\left(\frac{2m_e c^2\gamma^2\beta^2 W_{max}}{I^2}\right) - \beta^2 - \frac{\delta}{2}\right] \quad [\mathrm{GeV}/(\mathrm{g}/cm^2)]$$

- Step by step procedure:

1. Convert into: $dp/dx = dE/dx \times \beta^{-1}$
2. Calculate number of steps: $n_{steps} = 1 + (dp/dx \times \Delta x)/step$ with $step = 0.005$
3. Calculate step-wise total momentum loss: $\Delta p_{tot} = \sum_{i=0}^{n_{steps}} \Delta p_i = \sum_{i=0}^{n_{steps}} \frac{\mathrm{dp}}{\mathrm{dx}_\mathrm{i}} \Delta x_i$
4. Calculate total energy loss $\Delta E = E_{in} - \sqrt{p_{out}^2 + m^2}$ with $p_{out} = p_{in} - \Delta p_{tot}$
5. Apply multiplicative factor:
$$\frac{q}{p_T} *= cP4 = \left(1 + \frac{\Delta E}{p_{mean}^2}(\Delta E + 2 \times E_{in})\right)$$
6. Apply correction to covariance matrix:
$$P[4][4] += \left(\frac{\sigma_E}{p_{mean}^2} \times \frac{q}{p_T}\right)^2$$

KALMAN FILTER: MS CORRECTION

Molière Formula (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\theta_0 = \frac{13.6 MeV}{\beta p} z \sqrt{x/X_0}\left[1 + 0.038 \ln(x/X_0)\right]$$

- $X_0 = 42.54cm$ Radiation length of plastic scintillator in cm
- x is the step length
- z is the charge of incident particle
- Formulas for propagated σ's:

$$\begin{cases} \sigma_{\sin\phi} = \theta_0 \cos\phi \sqrt{1 + \tan^2\lambda} \\ \sigma_{\tan\lambda} = \theta_0 (1 + \tan^2\lambda) \\ \sigma_{q/p_T} = \theta_0 \tan\lambda \dfrac{q}{p_T} \end{cases}$$

KALMAN FILTER: ENERGY LOSS CORRECTION

Bethe-Bloch (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\frac{1}{\rho}\frac{dE}{dx} = K \times \frac{Z}{A} \times \frac{z^2}{\beta^2}\left[\frac{1}{2} ln\left(\frac{2m_e\gamma^2\beta^2 T_{max}}{I^2}\right) - \beta^2 - \frac{\delta}{2}\right] \quad \text{[GeV/(g/cm^2)]}$$

- Energy loss correction:

  1. Use multiplicative factor $cP4$ (see slide 7) to update q/p_T

  2. Add factor to diagonal element of 5x5 Covariance Matrix P correspondent to q/p_T (found through error propagation):

$$P[4][4] += \left(\frac{\sigma_E}{p_{mean}^2} \times \frac{q}{p_T}\right)^2$$

- **NOTE:** $\sigma_E = k \times \sqrt{|\Delta E|}$ where k is a tunable parameter set at 0.07

KALMAN FILTER: MS CORRECTION

Molière Formula (PDG)
https://pdg.lbl.gov/2005/reviews/passagerpp.pdf

$$\theta_0 = \frac{13.6MeV}{\beta p} z\sqrt{x/X_0}[1 + 0.038\ln(x/X_0)]$$

- Multiple Scattering smearing simulated in three steps:

  1. Obtain parameter σ's ($\sigma_{sin\phi}, \sigma_{tan\lambda}, \sigma_{q/p_T}$) through error propagation as described in slide 6

  2. Update covariance matrix diagonal elements:

$$\begin{cases} P[2][2] += \sigma^2_{sin\phi} \\ P[3][3] += \sigma^2_{\tan\lambda} \\ P[4][4] += \sigma^2_{q/p_T} \end{cases}$$

KALMAN FILTER

KALMAN FILTER IN GENERAL

1. Make a priori predictions for the current step's state and covariance matrix using the a posteriori best estimate of the previous step (i.e. updated using measurement)

STATE VECTOR $s_k^- = f(s_{k-1}^+, X_{k-1})$

COVARIANCE MATRIX $P_k^- = F_{k-1} P_{k-1}^+ F_{k-1}^T + Q$

$$F_{k-1} = \left.\frac{\partial f}{\partial s}\right|_{s_{k-1}^+, X_{k-1}}$$

JACOBIAN

$$Q$$

PROCESS NOISE COVARIANCE

Note: In the first iteration step we use step 0 estimates for the state vector and the covariance matrix (s_0, P_0), which can be made very roughly

KALMAN FILTER IN GENERAL

2. Calculate the measurement residual and the Kalman Gain

RESIDUAL	$\tilde{y}_k = m_k^h - H(s_k^-)$
KALMAN GAIN	$K_k = P_k^- H^T (R + H P_k^- H^T)^{-1}$

R

MEASUREMENT NOISE COVARIANCE

H

CONVERSION MATRIX

3. Update the estimate

STATE VECTOR	$s_k^+ = s_k^- + K_k \tilde{y}$
COVARIANCE MATRIX	$P_k^+ = (1 - K_k H) P_k^-$

Note: in the case where R is a null matrix $s_k^+ = s_k^h$ and $P_k^+ = 0$

Note: the conversion matrix is needed to make the dimensions of vectors and matrixes turn out right. For exemple if s_k^h is a 2-D vector and s_k^- is 5-D, then H would be a 2×5 matrix:

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

KALMAN FILTER MODEL

- Use parametrization used in ALICE: free parameter z, state vector s = $(y, x, sin\phi, \tan\lambda, \frac{q}{p_T})$ (ϕ azimuthal angle, λ dip-angle, p_T transverse momentum in yz plane), evolution function:

0

$$\frac{dy}{dz} = \frac{k * (sin\phi_0 + sin\phi_1)}{k * (\cos\phi_0 + \cos\phi_1)}$$

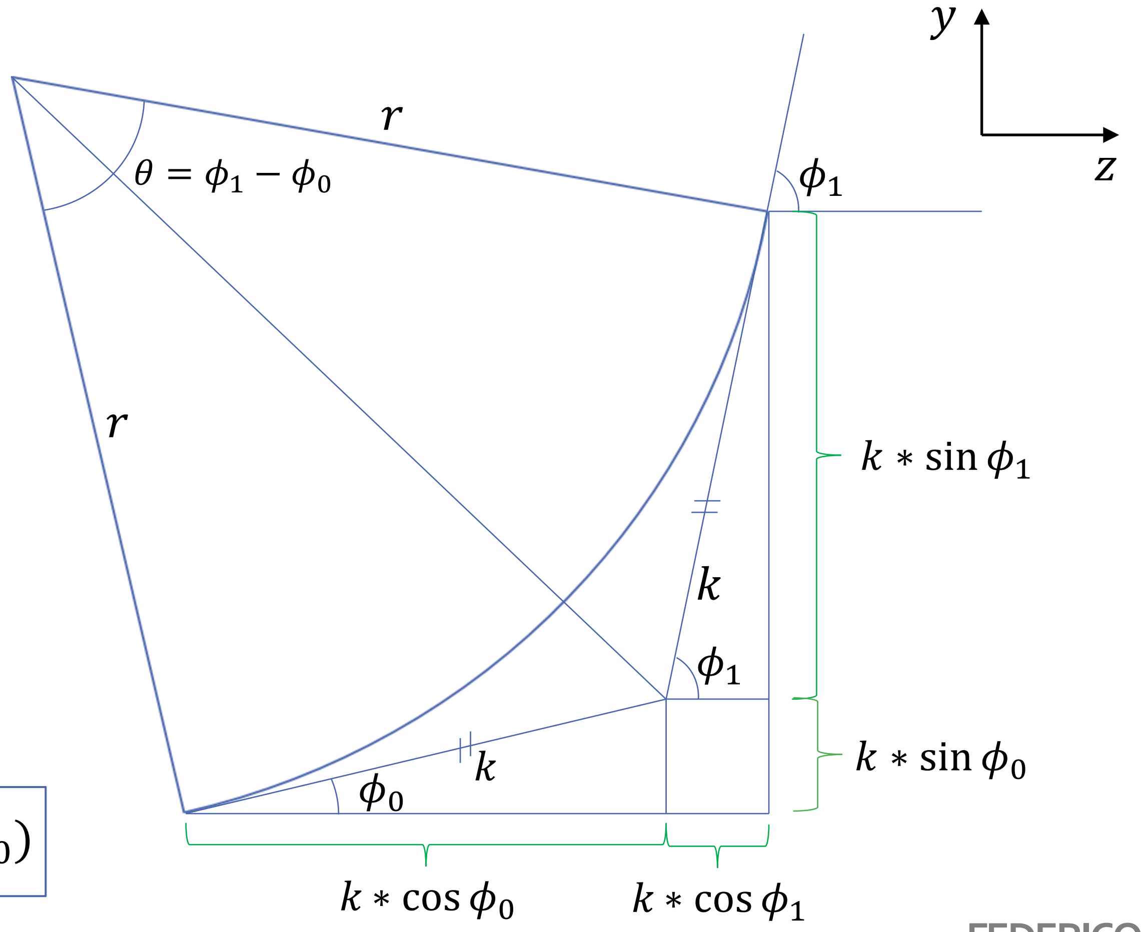
$$\boxed{y_1 = y_0 + \frac{(sin\phi_0 + sin\phi_1)}{(\cos\phi_0 + \cos\phi_1)} * dz}$$

1

$$dx = arch * tan\lambda = \theta * r * tan\lambda$$

$$\theta = \phi_1 - \phi_0 = \arcsin(\sin(\phi_1 - \phi_0)) =$$
$$= \arcsin(\cos\phi_0 \sin\phi_1 - \cos\phi_1 \sin\phi_0)$$

$$\boxed{x_1 = x_0 + \tan\lambda * \frac{r}{q} * \arcsin(\cos\phi_0 \sin\phi_1 - \cos\phi_1 \sin\phi_0)}$$



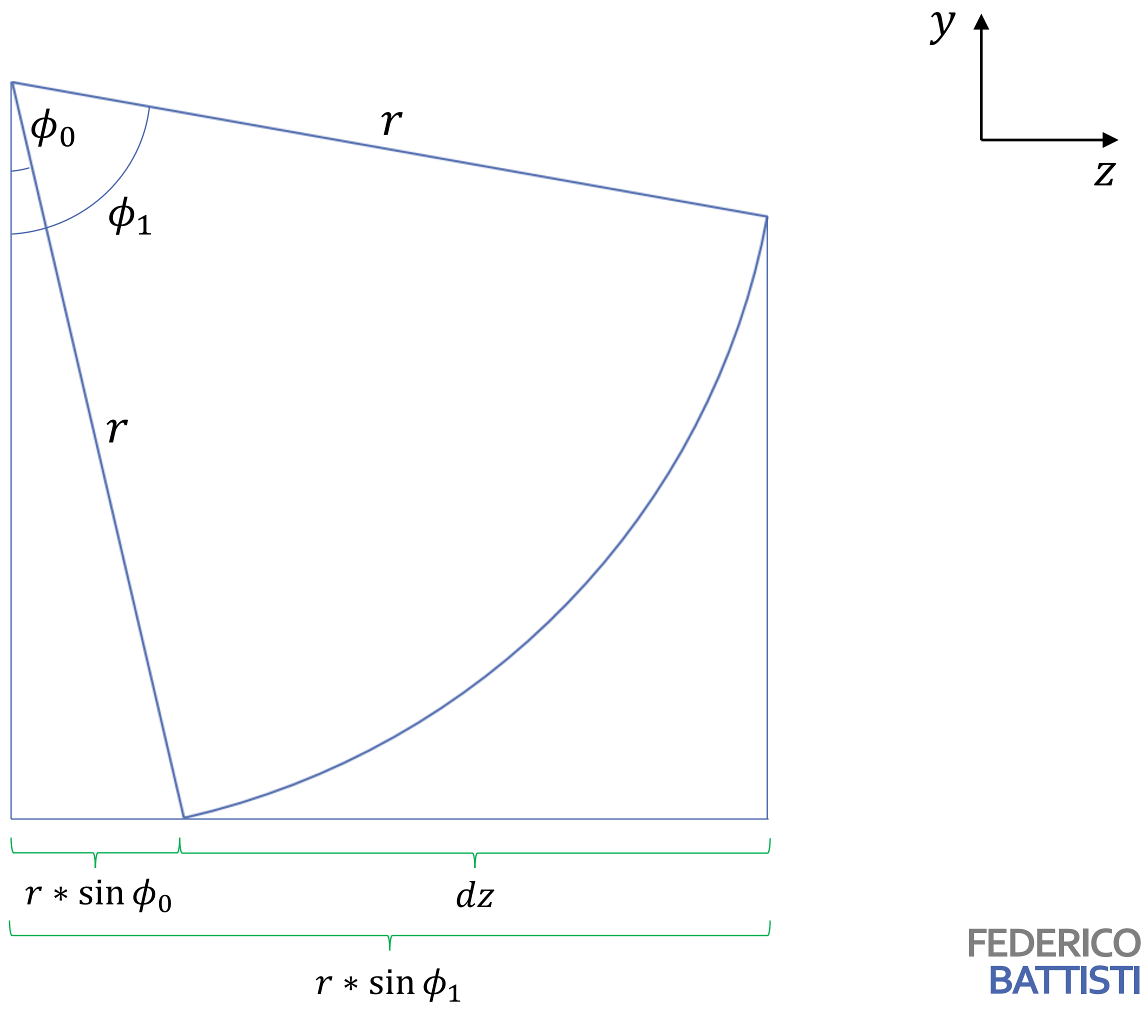
KALMAN FILTER MODEL

- Use parametrization used in ALICE: free parameter z, state vector $(y, x, sin\phi, \tan\lambda, \frac{q}{p_T})$ (ϕ azimuthal angle, λ dip-angle, p_T transverse momentum in yz plane), evolution function:

2 $\quad dz = r * \sin\phi_1 - r * \sin\phi_0$

$$\boxed{\sin\phi_1 = \sin\phi_0 + \frac{dz}{r}}$$

3 & 4 are static



HELIX FIT

K-F UPDATE: GLOBAL HELIX FIT AND INITIAL COVARIANCE ESTIMATION

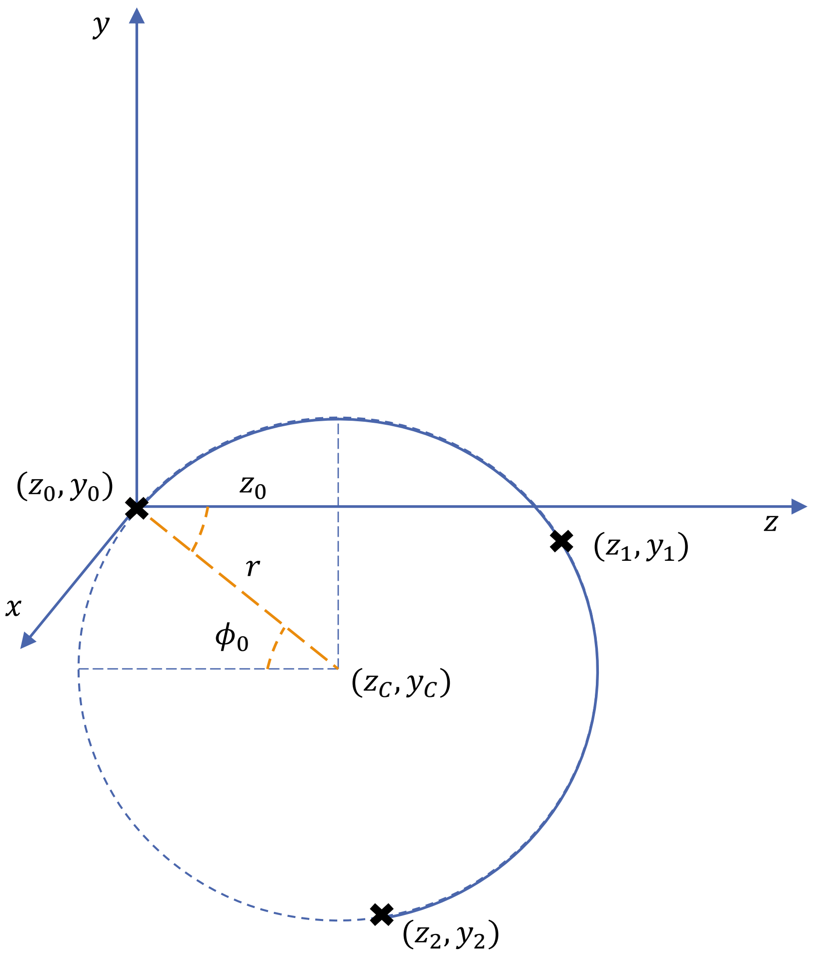
- $c = 1/r$ and $\sin\phi_0$ estimated by finding (z_c, y_C) and r of the yz plane circumference passing through the first, last and middle hit point of the particle trajectory

- After traslating the coordinate system to have the origin on the first point $(z_0, y_0) \rightarrow (0,0)$ we have the circumference equations:

$$\begin{cases} z_C^2 + y_C^2 = r^2 \\ (z_1 - z_C)^2 + (y_1 - y_C)^2 = r^2 \\ (z_2 - z_C)^2 + (y_2 - y_C)^2 = r^2 \end{cases}$$

$$\begin{cases} z_C = \frac{1}{2}\left(z_2 - y_2 \frac{z_1(z_1 - z_2) + y_1(y_1 - y_2)}{z_2 y_1 - z_1 y_2}\right) \\ y_C = \frac{1}{2}\left(z_2 - y_2 \frac{z_1(z_1 - z_2) + y_1(y_1 - y_2)}{z_2 y_1 - z_1 y_2}\right) \\ r = \sqrt{z_C^2 + y_C^2} \end{cases}$$

$$\boxed{\begin{array}{l} c = 1/r \\ \sin\phi_0 = \frac{z_0}{r} \end{array}}$$

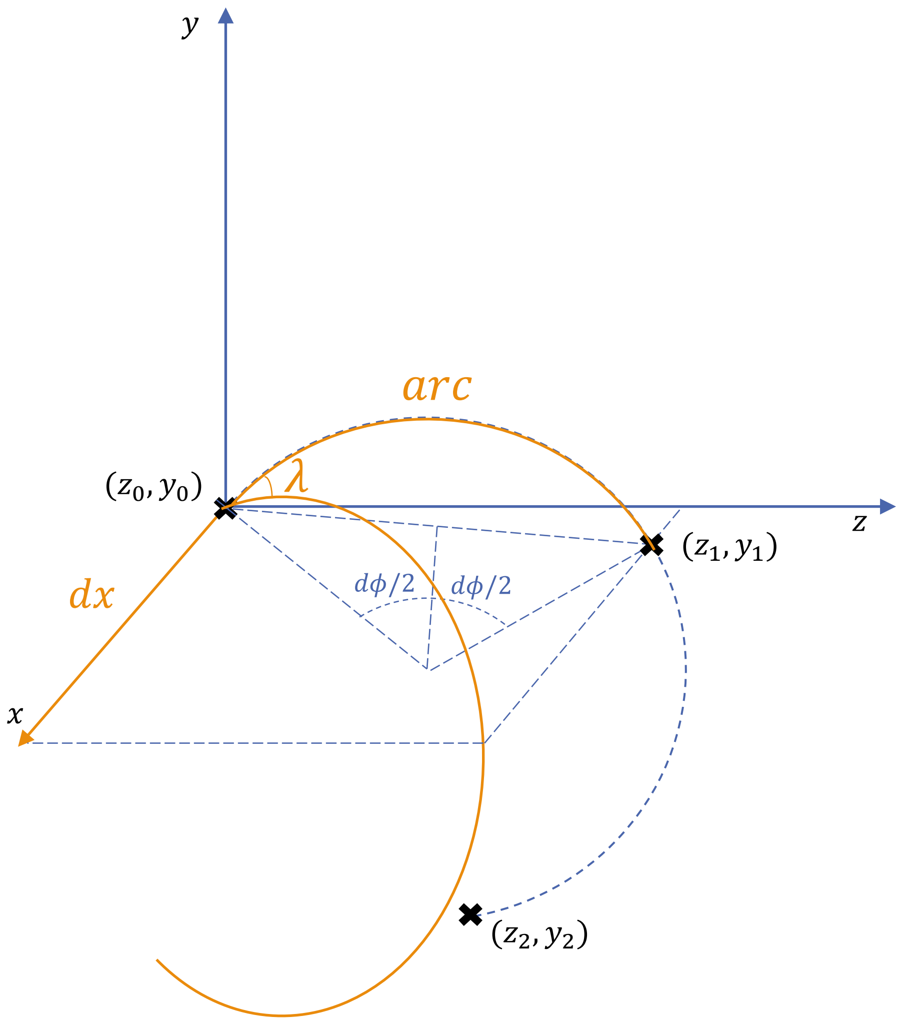


K-F UPDATE: GLOBAL HELIX FIT AND INITIAL COVARIANCE ESTIMATION

- We evaluate $\tan\lambda$ from the yz plane arc between the first two points and the correspondent movement in the x direction (magnetic field direction) using r estimate from previous step:

$$d\phi = 2\arcsin\left(\frac{chord}{2r}\right)$$
$$= 2\arcsin\left(\frac{\sqrt{(y_1-y_0)^2+(z_1-z_0)^2}}{2r}\right)$$

$$\boxed{\tan\lambda = \frac{dx}{arc} = \frac{dx}{d\phi * r}}$$



K-F UPDATE: GLOBAL HELIX FIT AND INITIAL COVARIANCE ESTIMATION

- Given parameter estimation from global helix fit, estimate uncertainties through error propagation
- Uncertainties associated with x and y: σ_{xy} ; z free parameter with no uncertainty $\sigma_z = 0$ (as in the Kalman filter)
- Formula for $\sin\phi_0$ estimation is function of $f(z_0, y_0, z_1, y_1, z_2, y_2)$ but since $\sigma_z = 0$, consider only $f(y_0, y_1, y_2) \rightarrow$ From error propagation we get:

$$\sigma_{\sin\phi_0} = \sqrt{\left(\frac{\partial f(y_0, y_1, y_2)}{\partial y_0}\right)^2 \sigma_{xy}^2 + \left(\frac{\partial f(y_0, y_1, y_2)}{\partial y_2}\right)^2 \sigma_{xy}^2 + \left(\frac{\partial f(y_0, y_1, y_2)}{\partial y_3}\right)^2 \sigma_{xy}^2}$$

- This can be approximated as:

$$\sigma_{\sin\phi_0} = \sqrt{\left(\frac{f(y_0 + \sigma_{xy}, y_1, y_2)}{\sigma_{xy}}\right)^2 \sigma_{xy}^2 + \left(\frac{f(y_0, y_1 + \sigma_{xy}, y_2)}{\sigma_{xy}}\right)^2 \sigma_{xy}^2 + \left(\frac{f(y_0, y_1, y_2 + \sigma_{xy})}{\sigma_{xy}}\right)^2 \sigma_{xy}^2}$$

K-F UPDATE: GLOBAL HELIX FIT AND INITIAL COVARIANCE ESTIMATION

- Repeat the process with other parameters to get respective uncertainties
- Estimate for covariance matrix P_0 is diagonal matrix with:

$$P_0 = \begin{pmatrix} \sigma_{xy}^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{xy}^2 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{sin\phi}^2 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{tan\lambda}^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{q/p_T}^2 \end{pmatrix}$$

- Note: off-diagonal elements could also be calculated, but are not at the moment