Kalman Filter-based reconstruction in SAND tracker

V.Pia, M.Pozzato, M.Tenti, G.Lupi

DUNE Italia collaboration meeting – Frascati

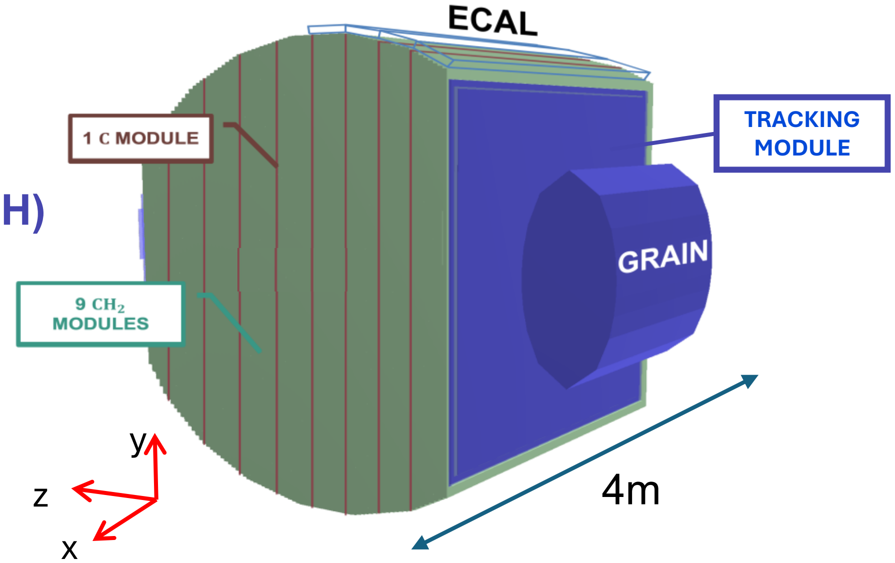
11/11/2025

INFN BOLOGNA
Istituto Nazionale di Fisica Nucleare
Sezione di Bologna

SAND Inner Tracker

The full geometry comprises **84 modules** each containing

- **Target slabs** (**1 C** every **9 CH_2**)
- **Radiator** 105 polypropylene foils alternated with air gaps
- **Detector layers: 4 (STT option) or 3 (DCH)**



Extended Kalman Filter for SAND

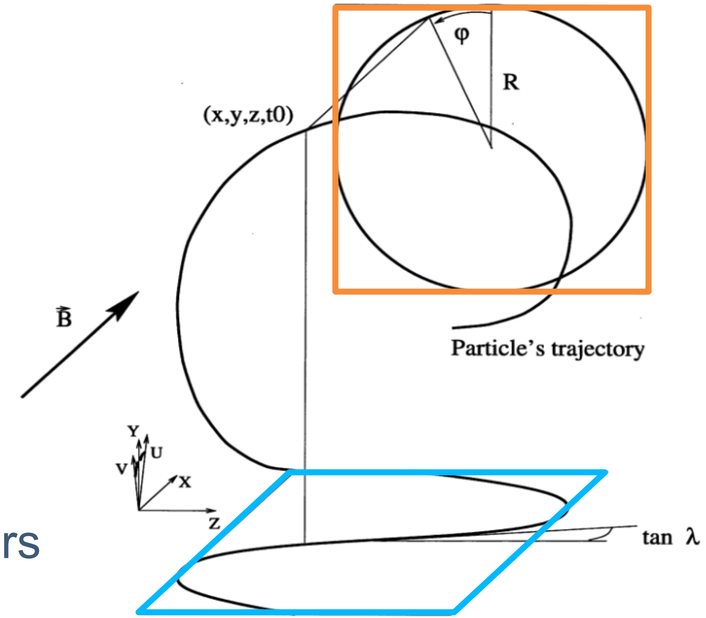
Non–linear dynamic track model including:

Magnetic field along the x-axis

- **Sinusoidal** in the xz-plane
- **Circlular trajectory** in the yz-plane

Passive targets

- Energy loss between subsequent tracker layers
- Multiple Coulomb Scattering

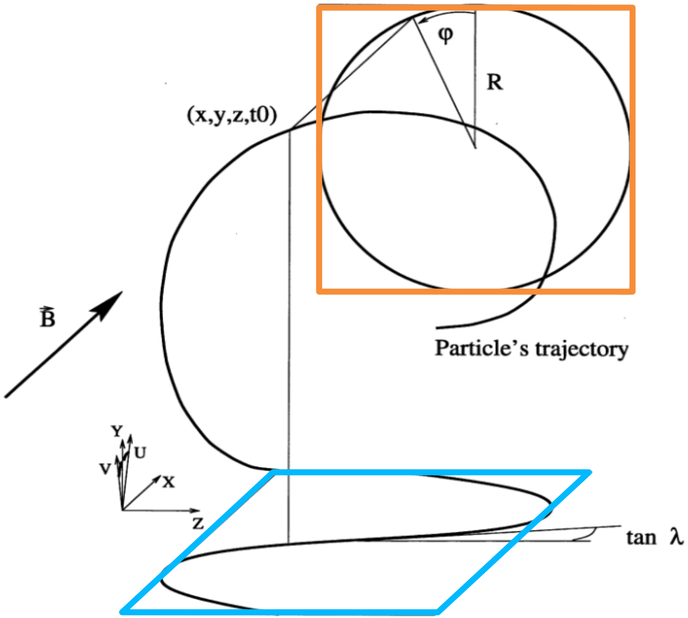


Trajectory parametrization

State vector $\mathbf{a_k}$

$$\begin{pmatrix} x \\ y \\ q/R \\ \tan\lambda \\ \phi \end{pmatrix}$$

X coordinate
Y coordinate
Signed inverse radius
Tangent of dip angle
Rotation angle



Measurement vector $\boldsymbol{m}_\mathbf{k}$

$$m_x = \begin{pmatrix} x \\ \theta_{xz} \end{pmatrix}$$ Horizontal plane wrt z-axis

$$m_y = \begin{pmatrix} y \\ \theta_{yz} \end{pmatrix}$$ Vertical plane wrt z-axis

Recap of last year presentation

Most of the **main steps** of the KF algorithm **implemented**

Statistical test and single-track reconstruction performed **to validate the KF** implementation

Complete reconstruction pipeline developed to work for both STT and Drift geometries:

- Digitization
- Clustering
- Tracklet finder

From last year - Consistency Checks

Innovation vector $r_k = m_k^{pred} - m_k^{true}$

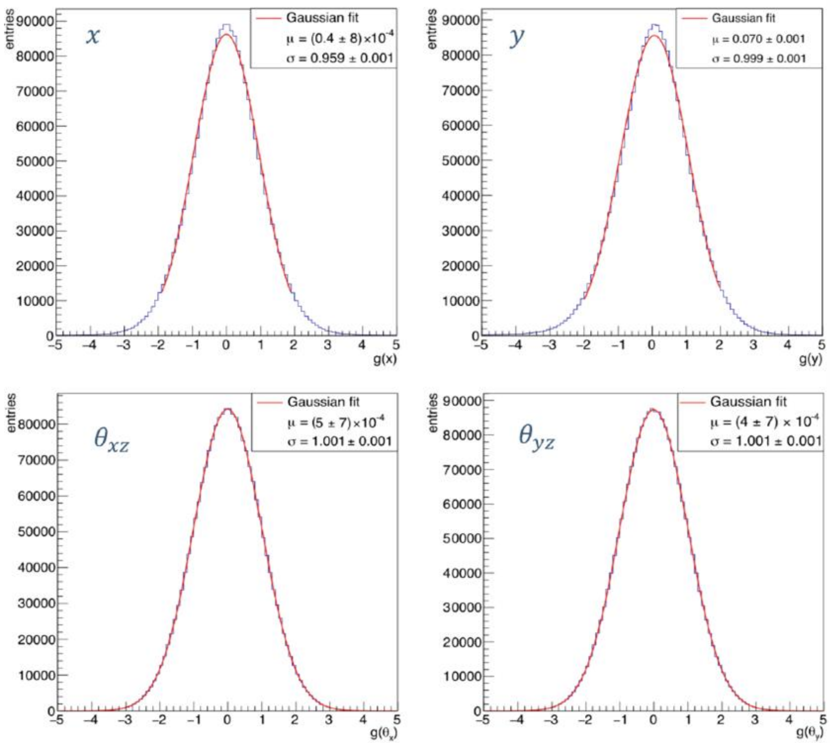
$$g(i)_k = \frac{r(i)_k}{\sqrt{C(i)_k}}$$

r_k^i = i-th element of the innovation at step k

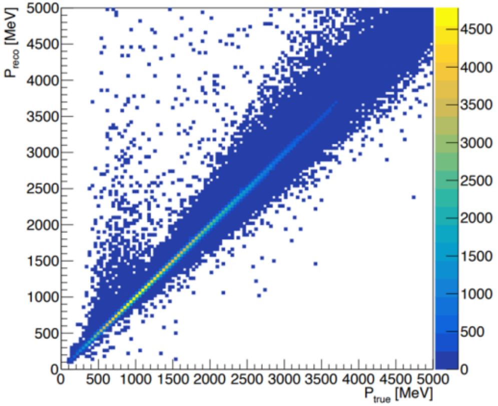
C_k^i = corresponding element of the measurement covariance matrix

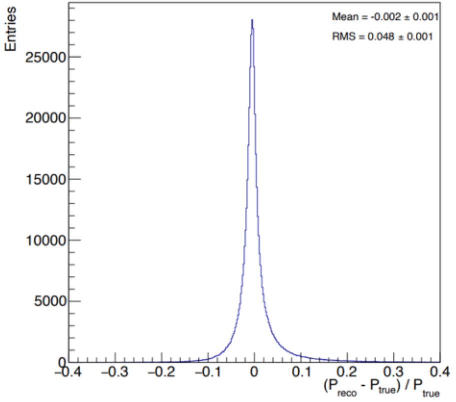
If the prediction is correct each g^i should be distributed as a **standard gaussian distribution** $(\mu = 0,\ \sigma = 1)$

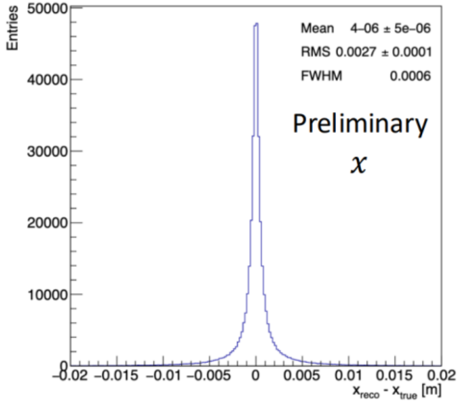
$m_x = \begin{pmatrix} x \\ \theta_{xz} \end{pmatrix}$ $m_y = \begin{pmatrix} y \\ \theta_{yz} \end{pmatrix}$

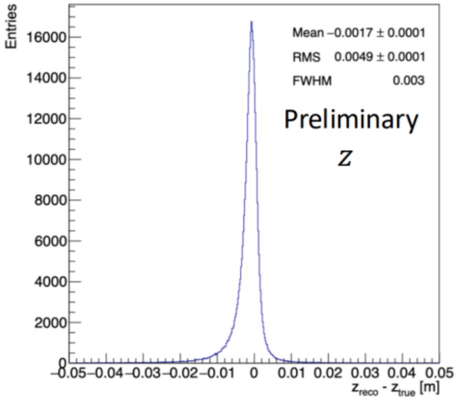


From last year - Track and vertex reconstruction









Performances for single particles by comparing true and reco values at the most upstream measurement layer

Large number of **low momentum particles with overestimated reconstructed energies** need investigation

x [mm]		$\tan\lambda$		ϕ[mrad]		P [%]	
Mean	RMS	Mean	RMS	Mean	RMS	Mean	RMS
$< 10^{-6}$	0.4	$< 10^{-6}$	0.006	1.1	3.5	−0.2%	4.9%

Preliminary attempt for a vertex reconstruction

	Mean [mm]	RMS [mm]	FWHM [mm]
x, y	< 0.1	3	0.6
z	~ -2	5	3

From last year - Next steps

DUNE ITALIA 2024

- **Merge** the tracking pipeline in the **sand-reco framework** to start use it

- **Validate the KF** algorithm with the new software
    - Perform the **statistical tests** and the particle reconstruction again

- Develop a realistic **seeding** algorithm

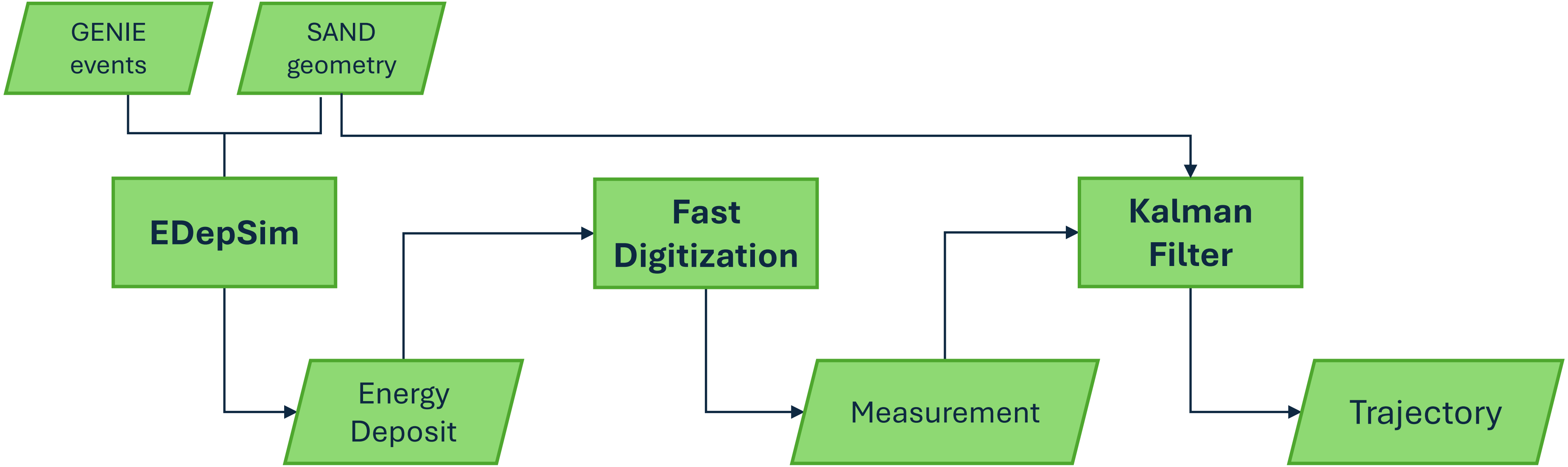
- Lot of other technical things

From last year - Next steps

[DUNE ITALIA 2024]

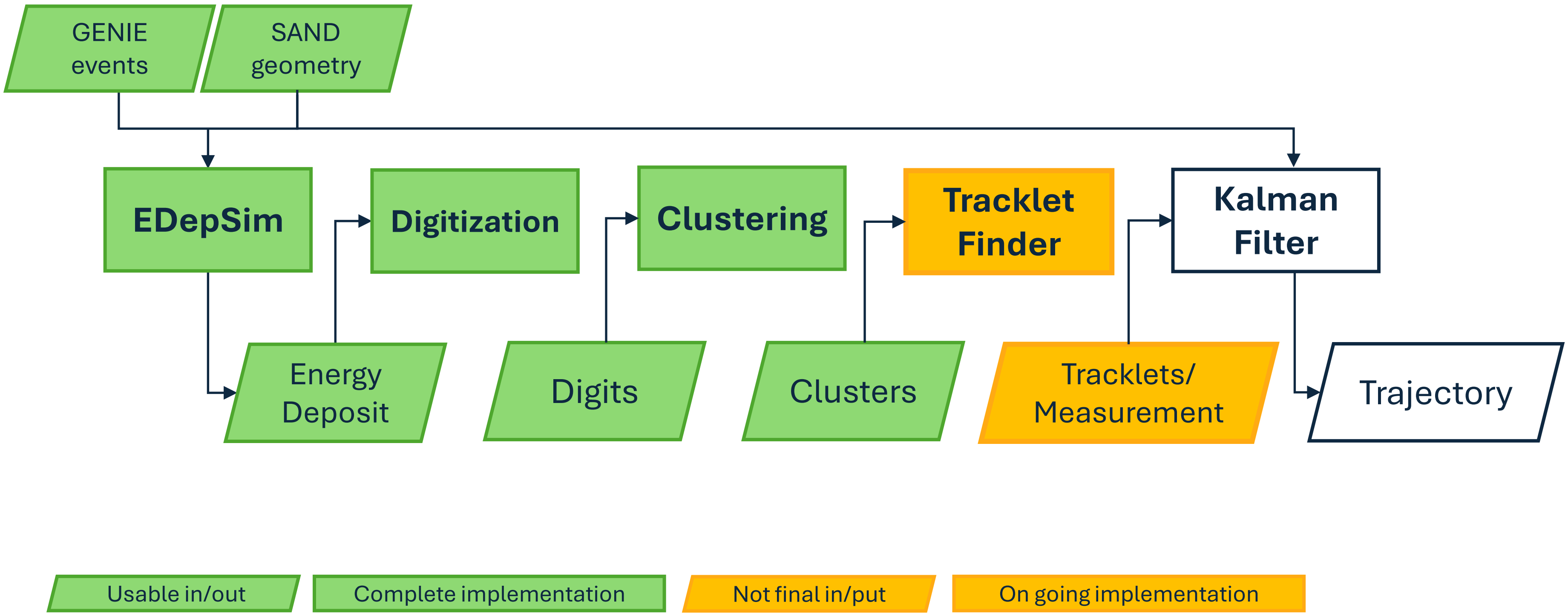
- **Merge** the tracking pipeline in the **sand-reco framework** to start use it → DONE
- **Validate the KF** algorithm with the new software → ALMOST DONE
  - Perform the **statistical tests** and the particle reconstruction again
- Develop a realistic **seeding** algorithm → DONE, NOT USED
- Lot of other technical things → IT WILL NEVER END!

Old Tracking Flow Chart



Usable in/out | Complete implementation | Not final in/put | On going implementation

Current Tracking Flow Chart



New consistency checks

Last year

Fast digitization:
Hits from MC-truth(TrajectoryPoint) smeared and sampled at fixed steps

Seeding from MC-truth

PID from MC-truth

Now

Full digitization:
Track segments (tracklets)
from full reco (smeared SegmentDetector)
→ required to modify edepsim and related SW!

- $\sigma_{pos} = 200\ \mu m$
- $\sigma_{ang} = 0.02$ rad

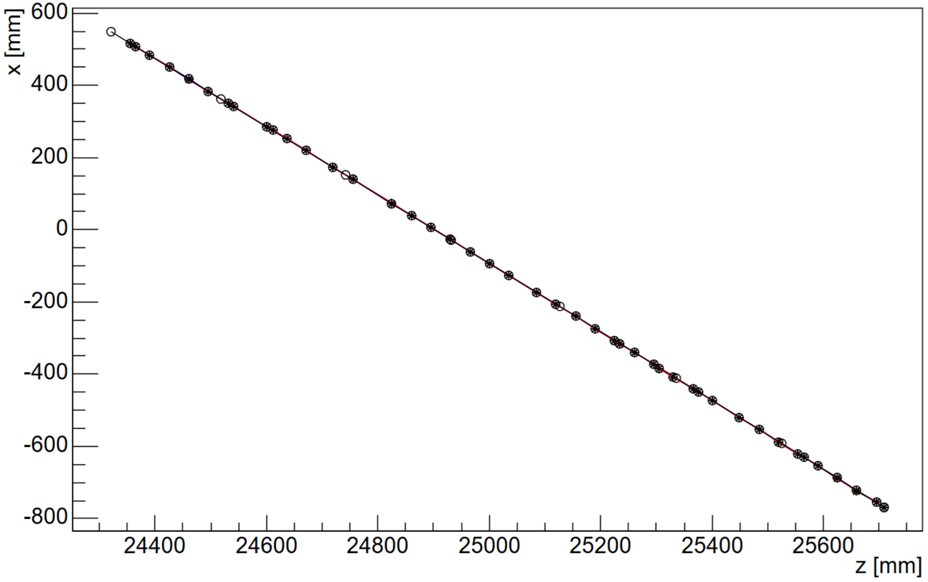
Seeding from MC-truth

PID from MC-truth

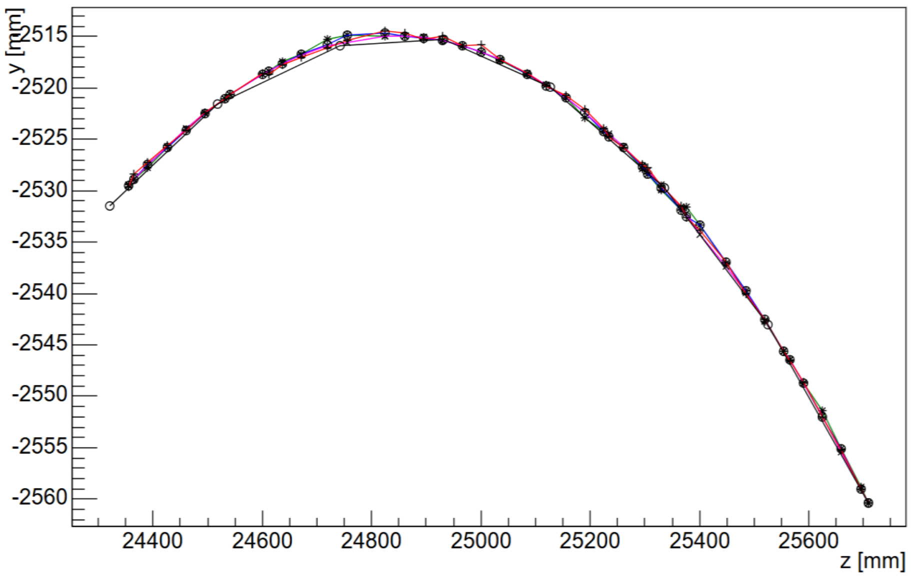
Muon reconstruction

Production: 1k muons with starting point in the tracker volume

Horizontal view

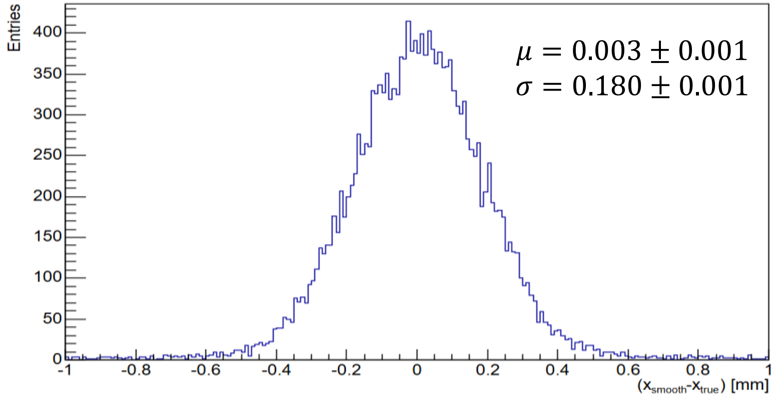


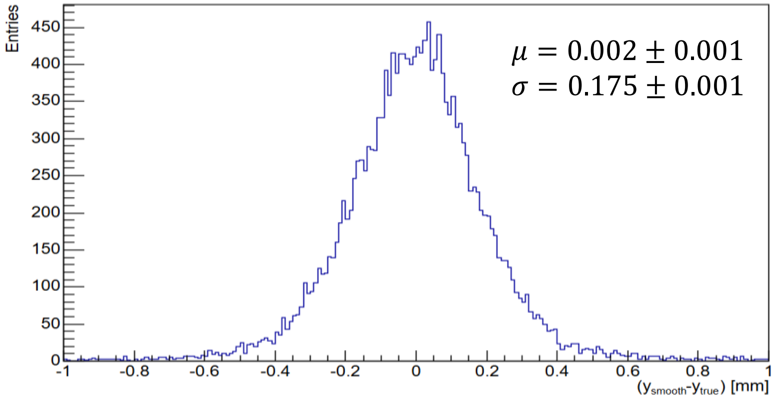
Vertical view

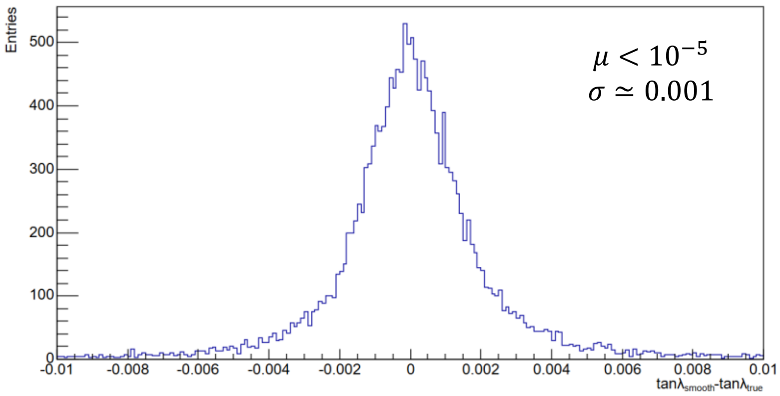


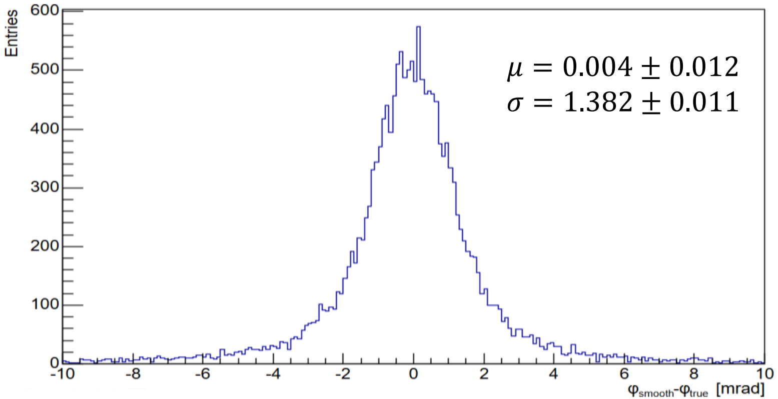
Residuals on track parameters

$$\begin{pmatrix} x \\ y \\ q/R \\ \tan\lambda \\ \phi \end{pmatrix}$$

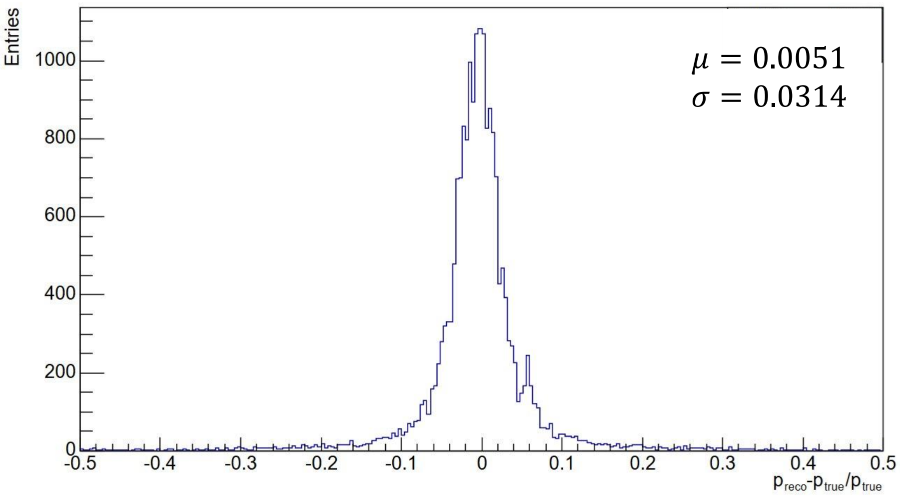


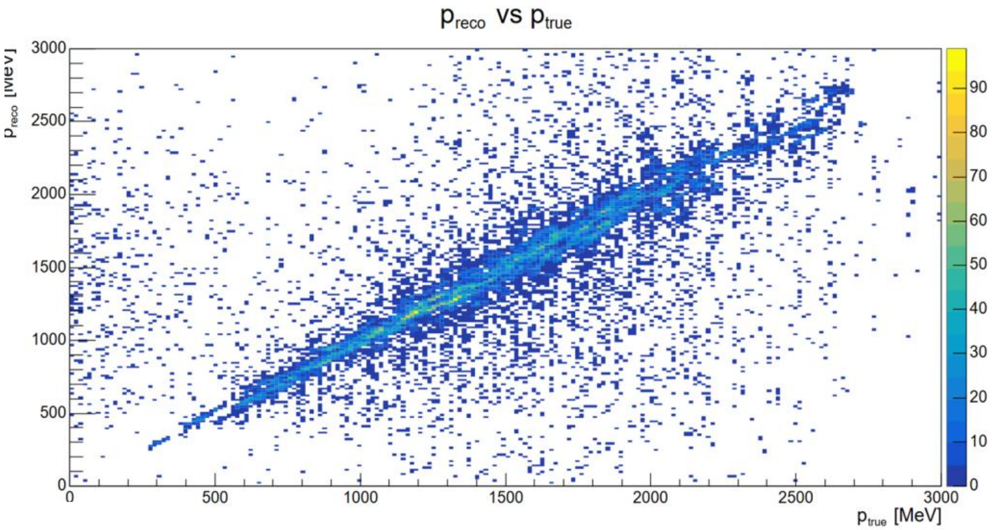






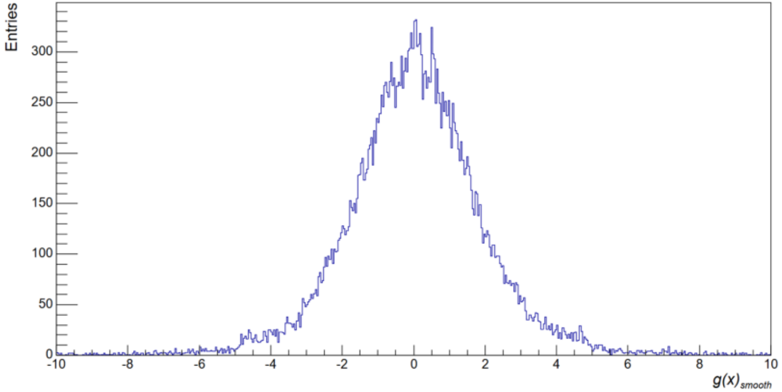
Residuals on reconstructed momentum

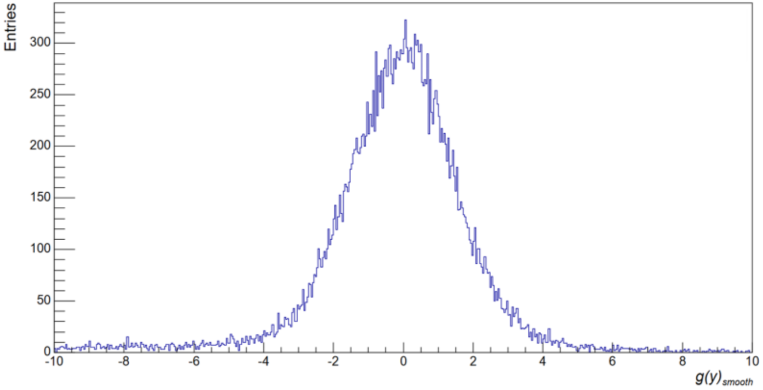


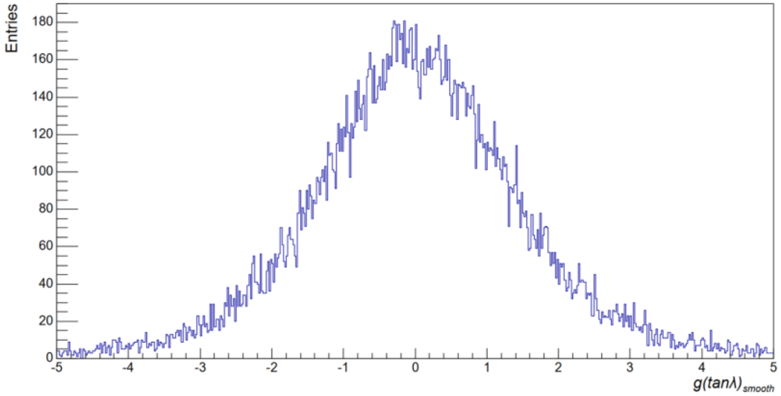


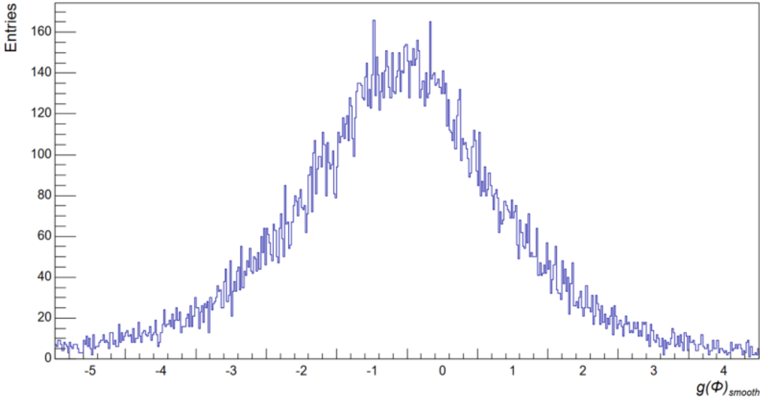
Pull tests on track parameters

$$g(i)_k = \frac{r(i)_k}{\sqrt{C(i)_k}}$$









$\sigma \neq 1$

We have some hypothesis, currently under investigation

Preliminary seeding algorithm

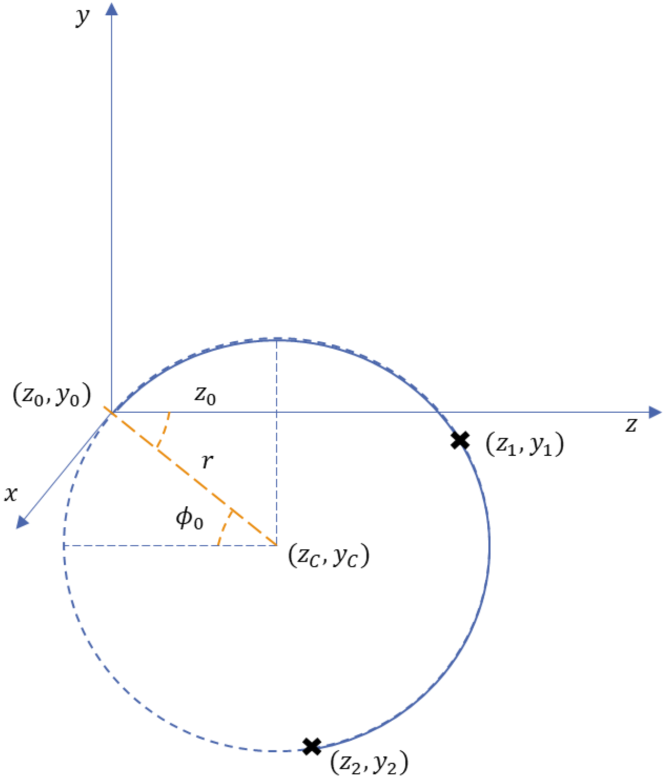
The **curvature** and initial **azimuthal** angle are estimated by **finding the center and radius** of the circumference in the plane perpendicular to the magnetic field

If we consider three points along the trajectory, the circumference is uniquely defined

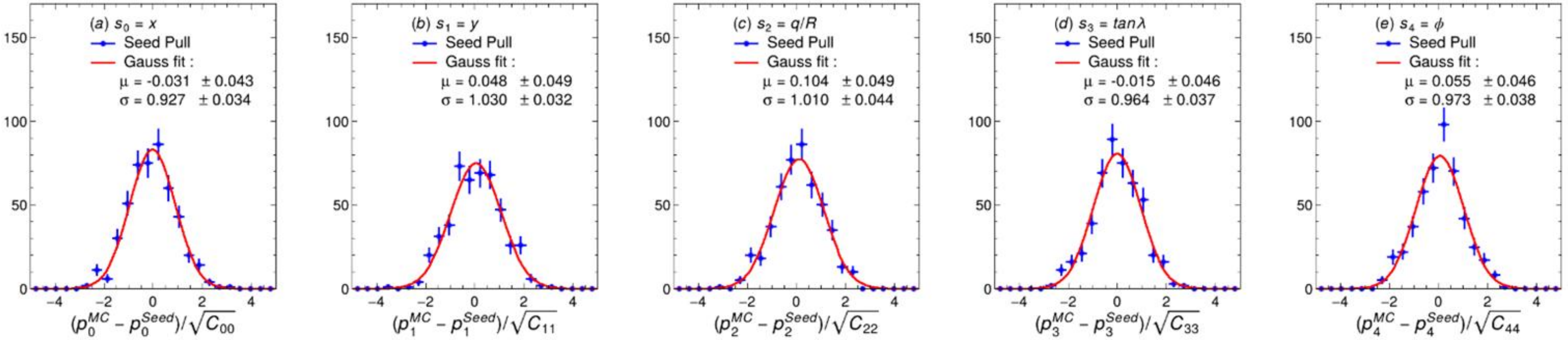
$$\begin{cases} z_C^2 + y_C^2 = r^2 \\ (z_1 - z_C)^2 + (y_1 - y_C)^2 = r^2 \\ (z_2 - z_C)^2 + (y_2 - y_C)^2 = r^2 \end{cases}$$

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

$$\boxed{\begin{cases} c = \dfrac{1}{r} \\ \phi_0 = \dfrac{x_0}{r} \end{cases}}$$



Preliminary seeding algorithm - Consistency Checks



- The seeding **algorithm is working**
- We are still using **MC seeding during the validation phase**
- We'll start using it as soon as the validation is completed

Conclusions

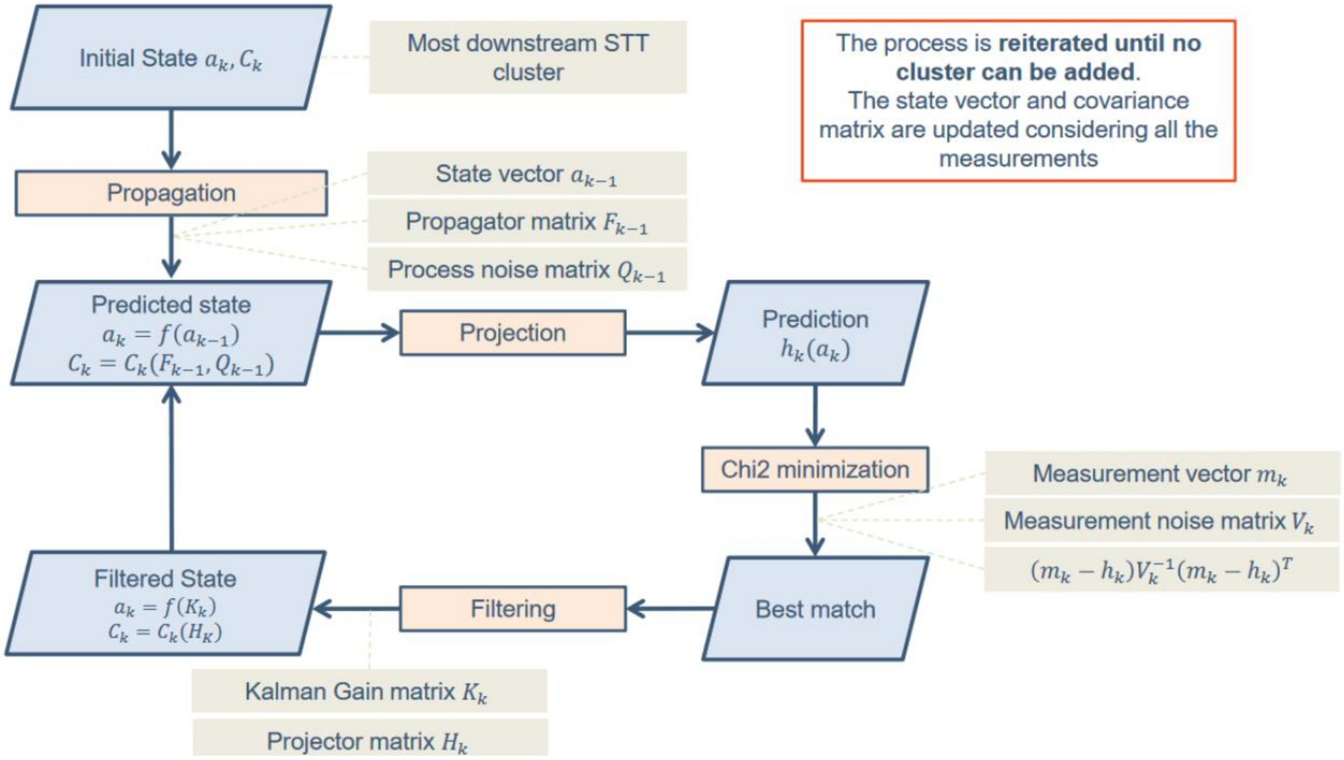
- KF with tracking pipeline has been **integrated in sand-reco framework** and already been used in some analysis with both geometries
- Developed **new seeding algorithm** to be used as starting point of the KF
- Performed **new statistical tests**: some distributions show discrepancies from expectation ($\sigma > 1$)
- Despite some discrepancies, the **algorithm is working**:
  - Momentum resolution $\sim 3\%$
  - Angular residual $\sim 3\ \mathrm{mrad}$
  - Position residual $\sim 190\ \mu\mathrm{m}$

Next steps

- Investigate pull test results → ongoing and almost done
- Working on a better tracklet finder algorithm
  - The current one has some limitations with tracks with large angles
- Integrate KF + seeding in the new sandreco ufw

BACKUP

Kalman Filter Algorithm



The process is **reiterated until no cluster can be added**.
The state vector and covariance matrix are updated considering all the measurements

Measurement vector on STT

- Measurement vector:

$$m_k = \begin{pmatrix} x \\ \theta_{xz} \end{pmatrix} \quad \theta_{xz} = -\kappa \cdot \arctan \frac{\tan\lambda}{\sin\phi}$$

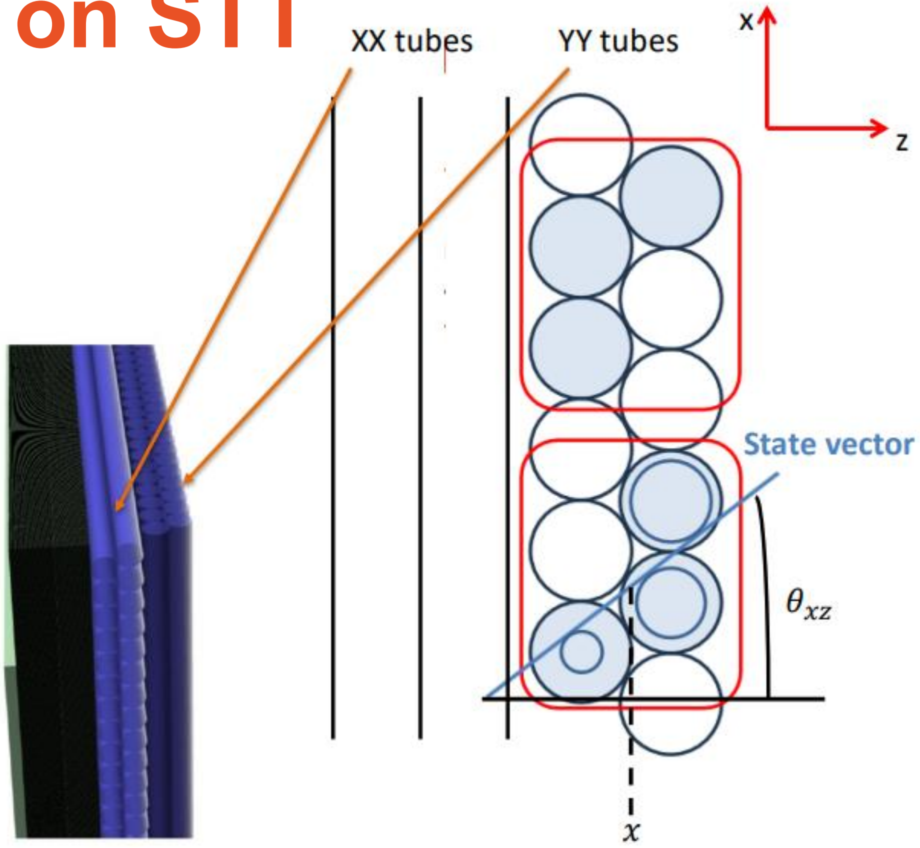
$$m_k = \begin{pmatrix} y \\ \theta_{yz} \end{pmatrix} \quad \theta_{yz} = \phi + \kappa \cdot \frac{\pi}{2}$$

Either one depending on the STT plane

Initial state (seed)

Initial state vector and covariance matrix obtained from most downstream STT cluster

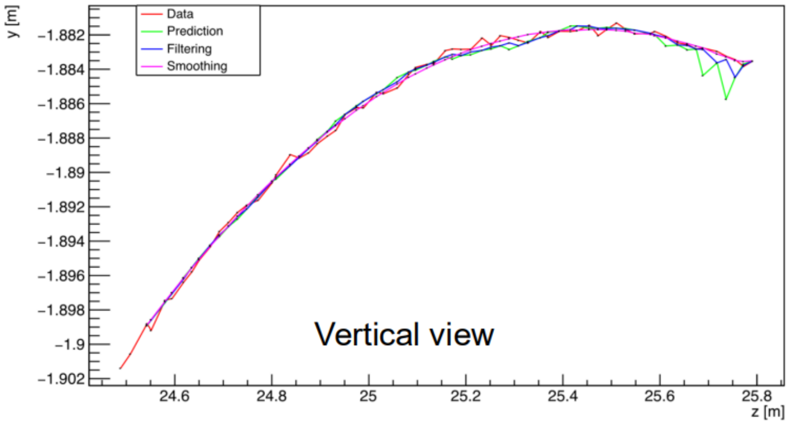
- Input geometry is read: tubes and planes info stored and organized
- Input tube-digits are grouped in plane
- Within plane adjacent tube-digits are clustered
- Reconstruct radius for tube-digits
- Evaluate common tangents and take the best one according to a likelihood
- **Clusters are reconstructed: m, q, t0, quality**
- **Most downstream cluster is the initial state for the Kalman filter**

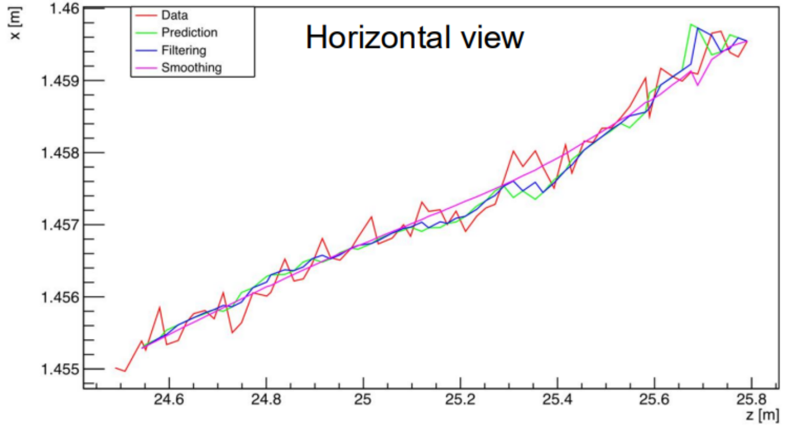


Past year activities

Kalman Filter tested with 10^6 ν_μCC events in STT:

- Energy loss and MCS included
- μ, π, p trajectory reconstruction
- Preliminary vertex reconstruction
- Statistical test to validate the algorithm





Pending Objectives DUNE ITALIA 2024

Fast digitization : MC info smeared and

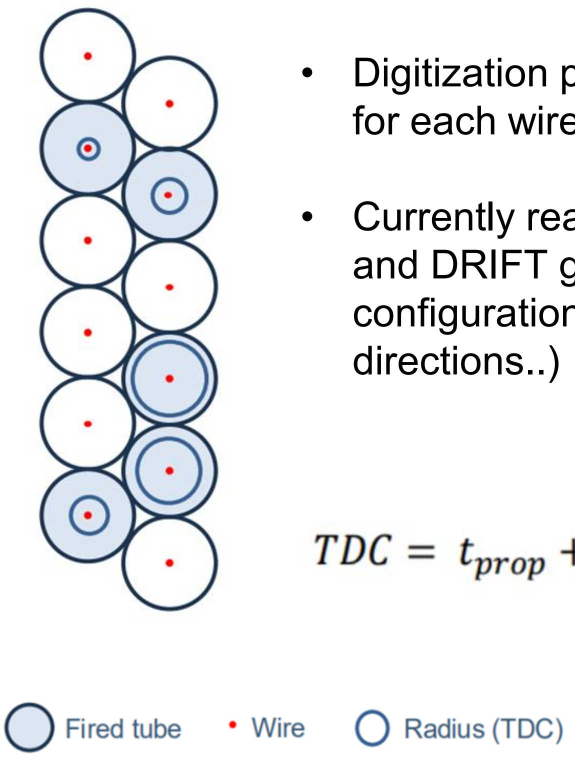
sampled at fixed steps

No **track finding**

Seeding from MC-truth

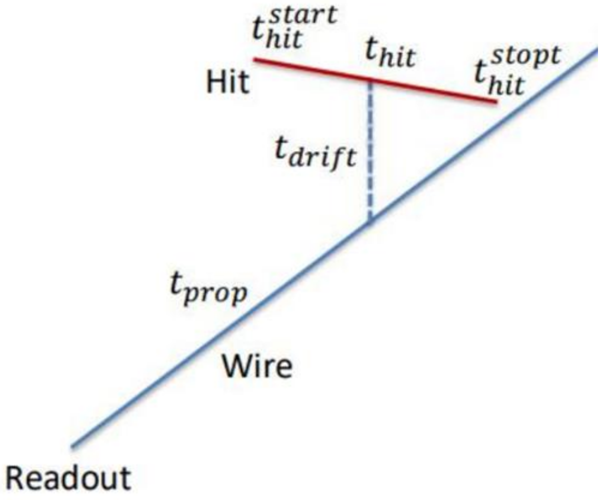
PID from MC-truth

Digitization



- Digitization provides a TDC and ADC for each wire and energy deposit

- Currently ready to work with both STT and DRIFT geometries and any configuration (pitch, number of wires, directions..)

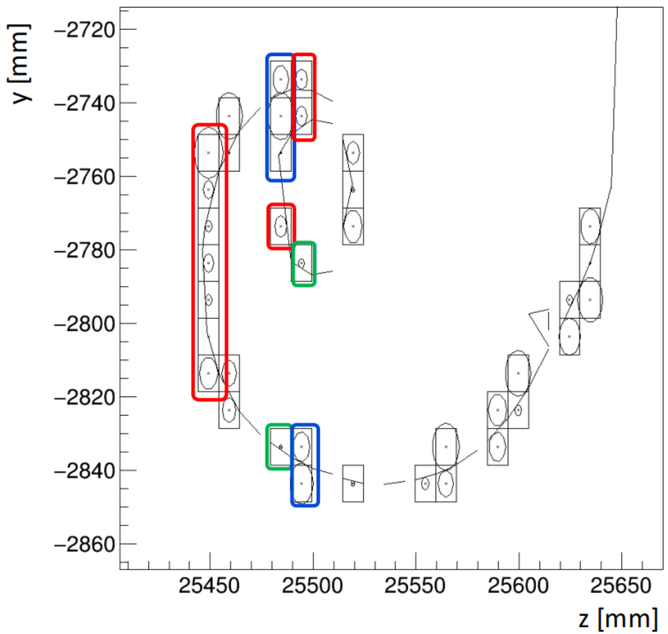


$$TDC = t_{prop} + t_{drift} + t_{hit} + 3.5\ \text{ns smearing}$$

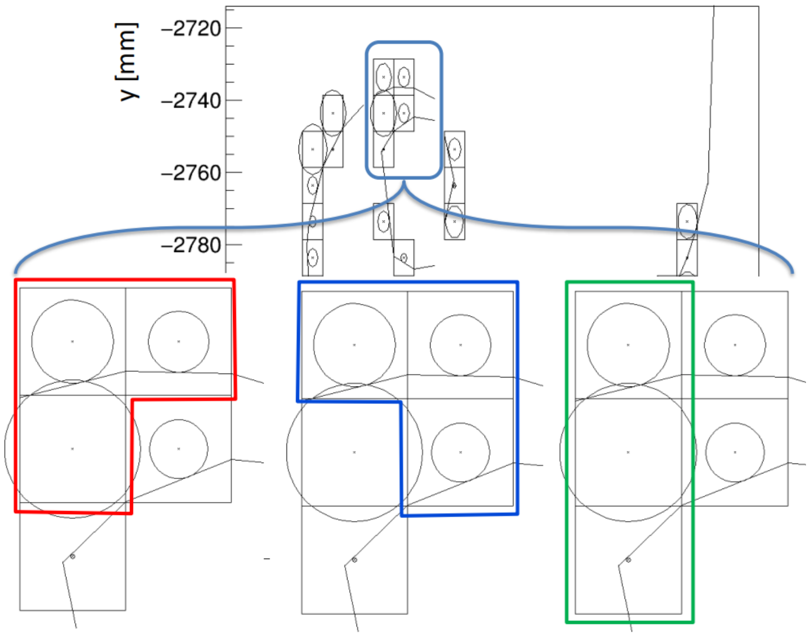
Clustering

Should group digits in cluster based on some criteria :

Close wires in the same plane



All unique triplets of close wires



Tracklet Finder

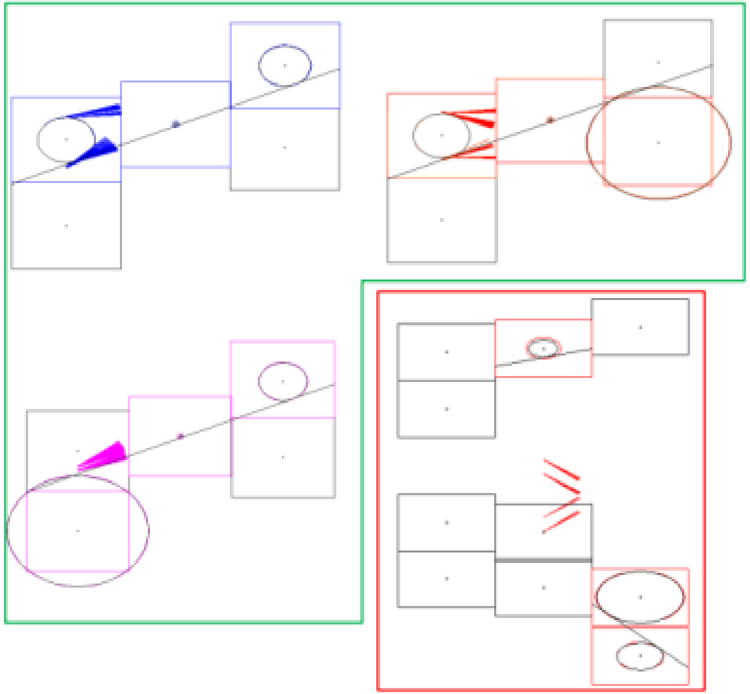
Minimization with migrad

$$D = \sum_i (d_i - r_i)^2$$

- d_i is the distance between the i-th fired wire and the trial track
- r_i is the radius of the i-th fired wire (from t_{drift}^{reco})

Scan over 4 parameters:

- xz, yz **angles**, limited in the z+ quadrants
- x,y **positions**, limited in the region defined by the cell intersection



Tracklet Finder

1. Set of tracklet candidates from **TrackletFinder** algorithm

2. True particle trajectories from MC

3. For each tracklet both **position and direction** are compared to the MC truth $\boxed{m_x = \begin{pmatrix} x \\ \theta_{xz} \end{pmatrix} \quad m_y = \begin{pmatrix} y \\ \theta_{yz} \end{pmatrix}}$

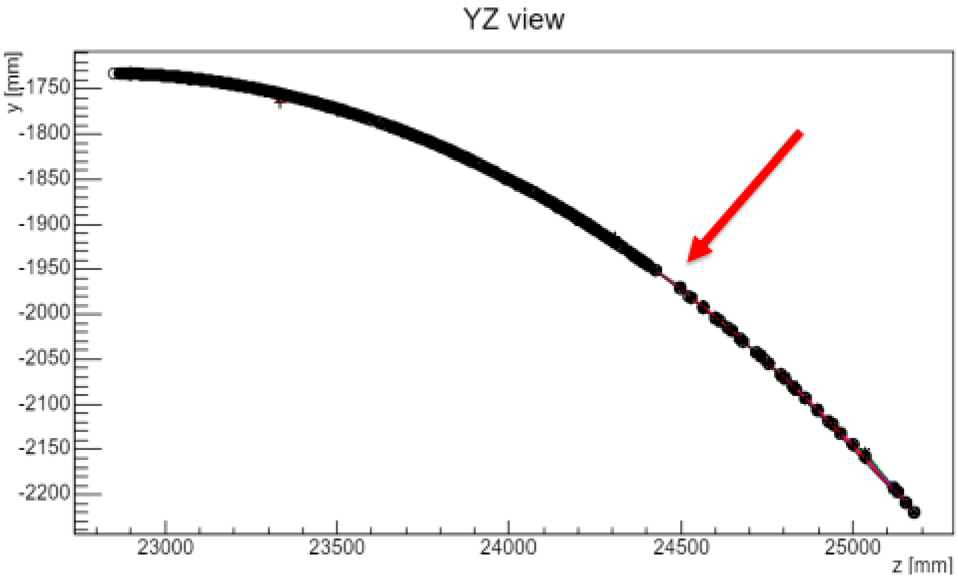
4. A **score** is assigned to each tracklet to select the **best match**

Tracklet Finder

- Evaluate best tracklet based on **position selection**

- Edepsim **sampling is not homogeneous** → find **two closest adjacent points** of the trajectory at the corresponding $z_{tracklet}$ and perform a **linear interpolation**

- True trajectory **position** and **momentum** are computed



Tracklet Finder

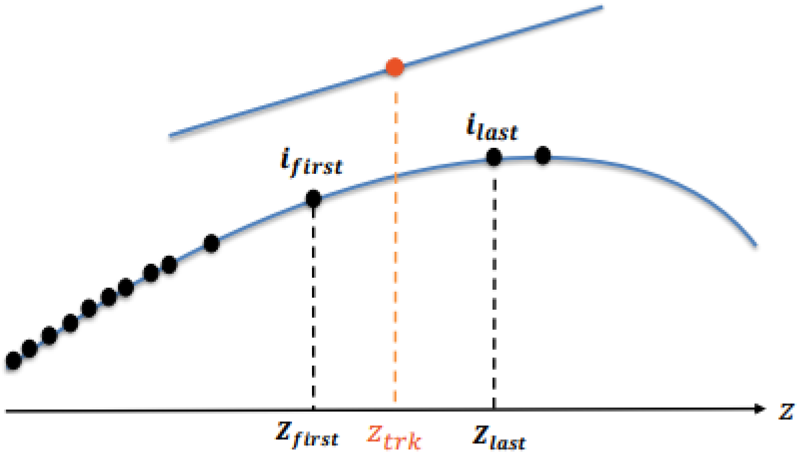
- Evaluate best tracklet based on **position and direction selection**
- Edepsim **sampling is not homogeneous** → find **two closest adjacent points** of the trajectory at the corresponding $z_{tracklet}$ and perform a **linear interpolation**
- True trajectory **position** and **momentum** are computed

i_{first} closest smaller Z (point before z_tracklet)
i_{last} closest larger Z (point after z_tracklet)

$$X_{trj} = X_{first} + \alpha\left(X_{last} - X_{first}\right)$$
$$Y_{trj} = Y_{first} + \alpha\left(Y_{last} - Y_{first}\right)$$

where

$$\alpha = \frac{Z_{trk} - Z_{first}}{Z_{last} - Z_{first}}$$



Tracklet Finder

- Evaluate **best tracklet direction** searching for the **most parallel** vector
- Compute tracklet and trajectory direction with the respect of z-axis
- Direction is computed as the scalar product between tracklet and trajecotry momentum vectors

Similarly the true momentum at z_tracklet is

$$\boldsymbol{P}^{true} = \boldsymbol{P}_{first} + \alpha\left(\boldsymbol{P}_{last} - \boldsymbol{P}_{first}\right)$$

From which the direction for each plane orientation can be computed as

$$\theta_{XZ} = \tan^{-1}\left(\frac{P_X^{true}}{P_Z^{true}}\right) \qquad \theta_{YZ} = \tan^{-1}\left(\frac{P_Y^{true}}{P_Y^{true}}\right)$$

Tracklet Finder

Position Matching

Compute Euclidean distance between interpolated trajectory position and tracklet position

$$d_{pos} = \sqrt{(x_{trk} - x_{true})^2 + (y_{trk} - y_{true})^2}$$

Direction Matching

The angular deviation between the tracklet and true trajectory is computed using the dot product

$$\cos(\theta) = \frac{\vec{p}_{trk} \cdot \vec{p}_{true}}{|\vec{p}_{trk}||\vec{p}_{true}|}$$

Score

To select the best tracklet, we define a **combined score** based on position and angle deviations

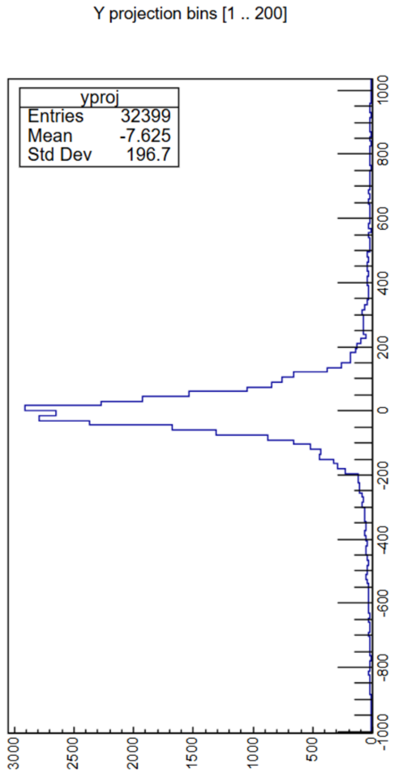
$$D = \frac{d_{pos}}{\sigma_{pos}} + \frac{\cos^{-1}(\cos\theta)}{\sigma_{ang}}$$

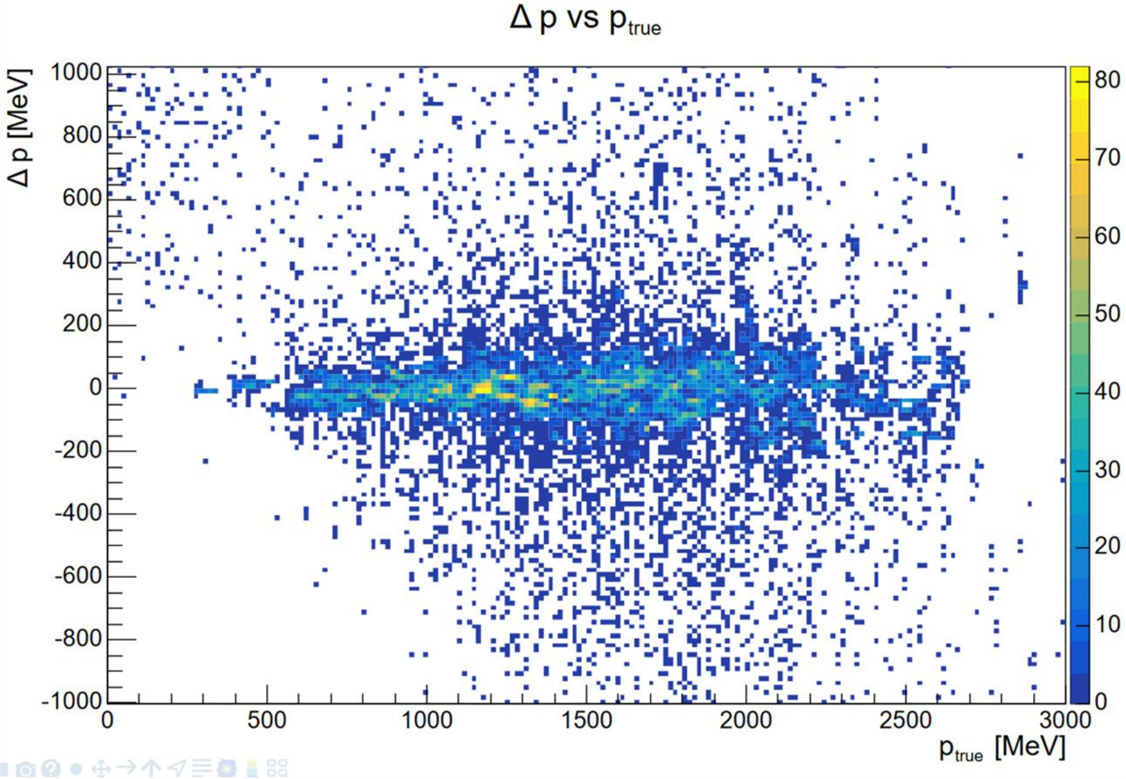
- $\sigma_{pos} \sim 200\ \mu m$
- $\sigma_{ang} \sim 0.2$ rad

New pull tests

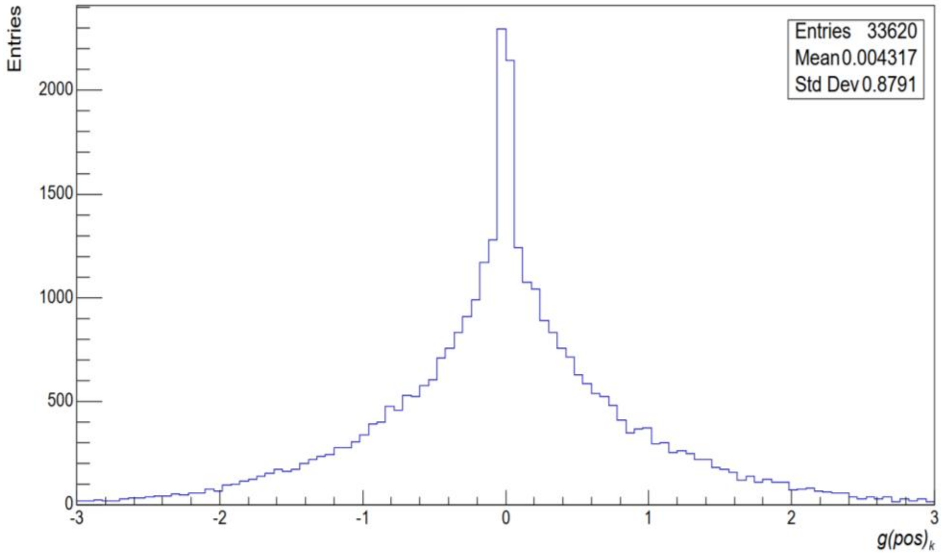
- Edepsim stores:
  - **TrajectoryPoints** with MC info of the trajectories → **particle**
  - **DetectorSegments** with MC info of the energy deposits → **hit**
- In the past we could compare reco and true quantities as we were smearing the info of the TrajectoryPoints (position and momentum)
- Currently **the reco is obtained from the DetectorSegments** (no momentum info in the hits and no 1:1 correspondency between points and hit)
- We **modified the SegmentDetector class** to also store the true momentum at the start and stop point of each segment
- **Modified EDEPReader** (in the legacy version) to handle this change

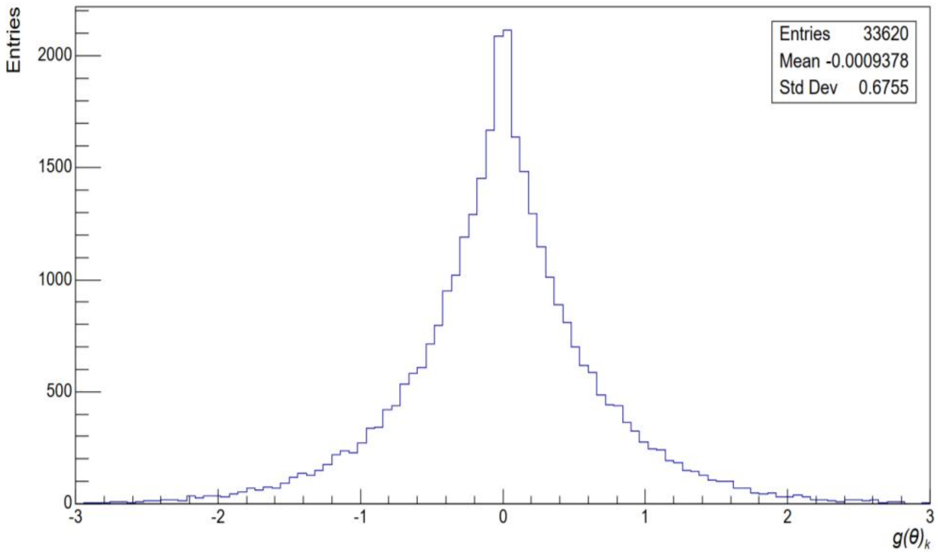
Residuals on reconstructed momentum



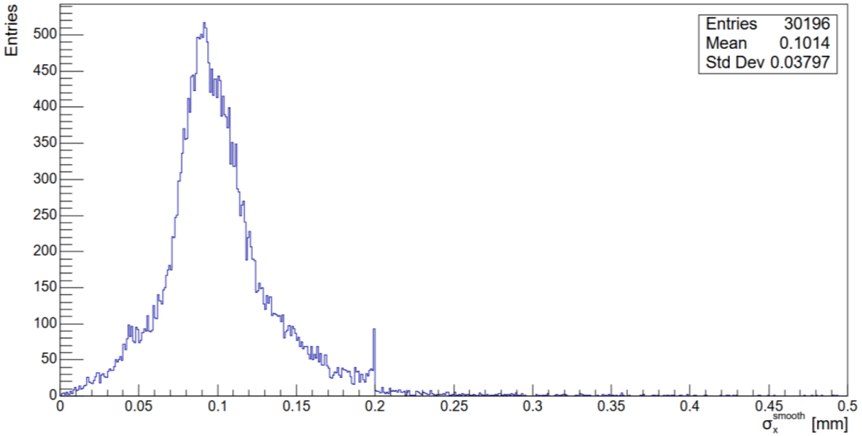


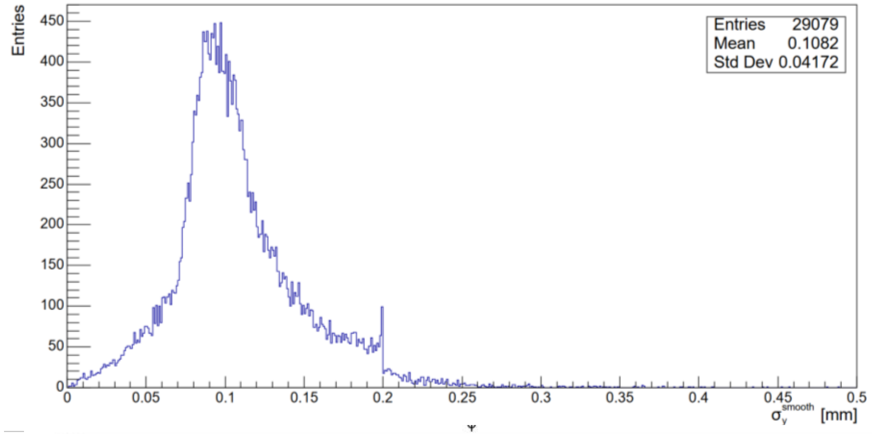
New pull tests on measurements

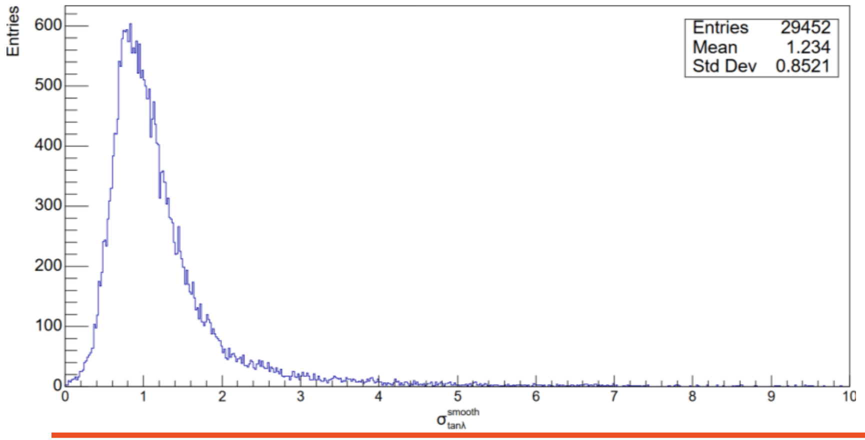


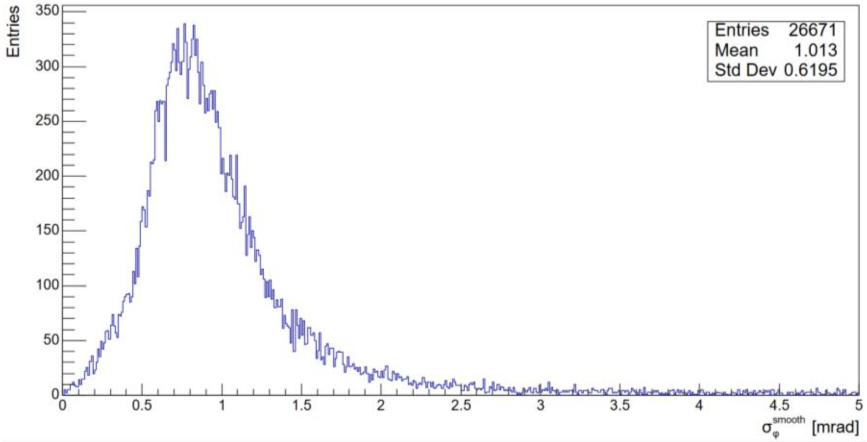


Sigmas distributions

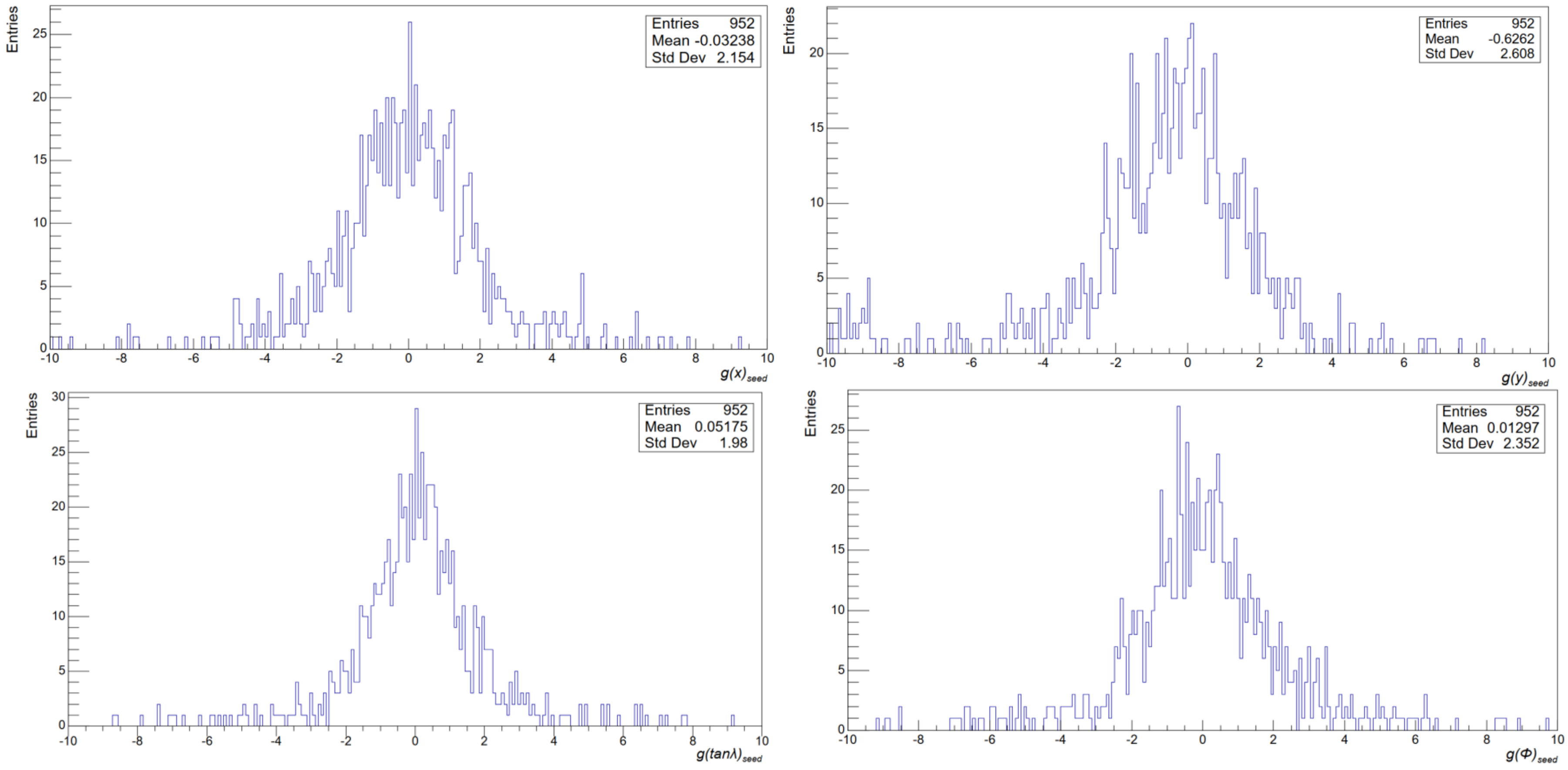








Residuals at seeding point



Residuals of momentum at seeding point