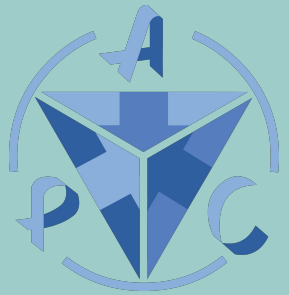
Energy reconstruction updates

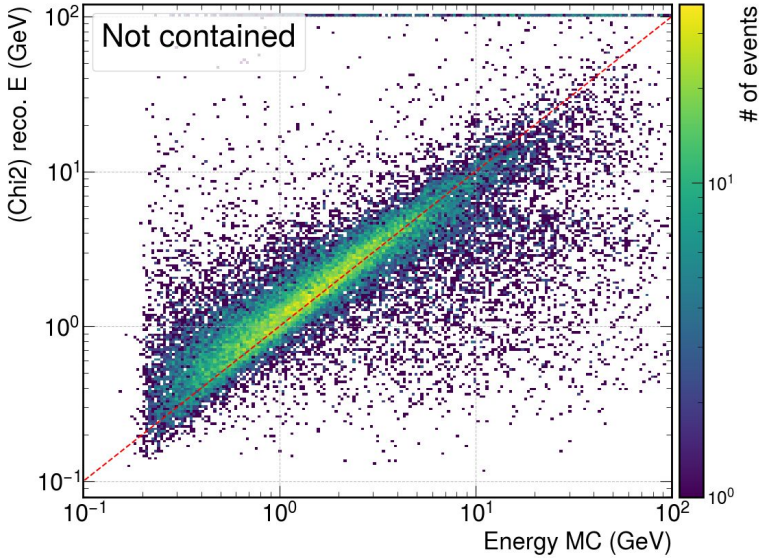
Henrique Souza for the APC group

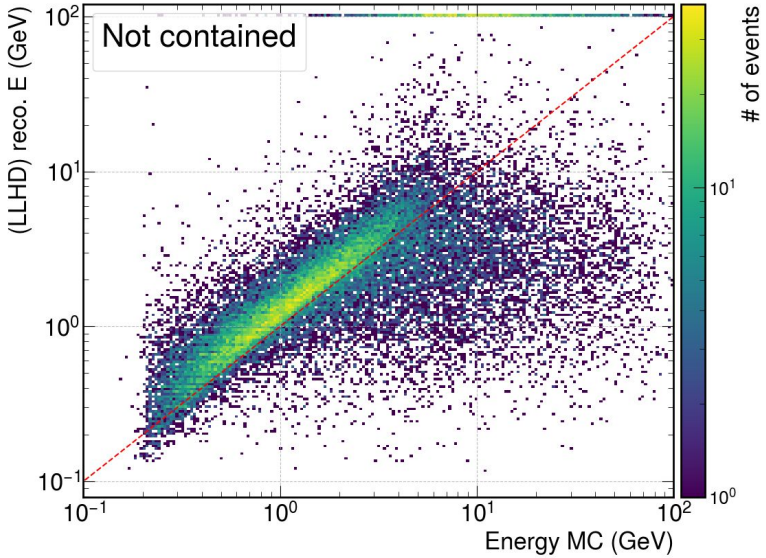
29/10/2024



Summary

Tried to improve current Multiple Coulomb Scattering algorithms to retrieve energy for escaping muons





1. Verified performance and adjusted algorithms based on Monte Carlo trajectory
2. Back to recob::Track -> Adjusted methods based on expected results

Past presentations showing methods:
#1, #2

NOTE: Only considering longest tracks that are muons, containment checked with MC(no purity/completeness evaluation)

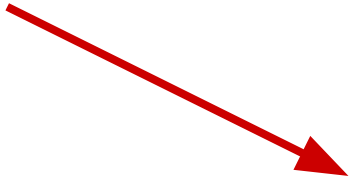
MC Trajectory

True trajectory, current version

- **Validation of methods Chi2 and LLHD using Truth information:**
  - Apply method of Multiple Coulomb Scattering (MCS) in **MCParticle trajectory**

`Point_t const& recob::Track::LocationAtPoint ( size_t i )`

Past presentations showing methods: #1, #2

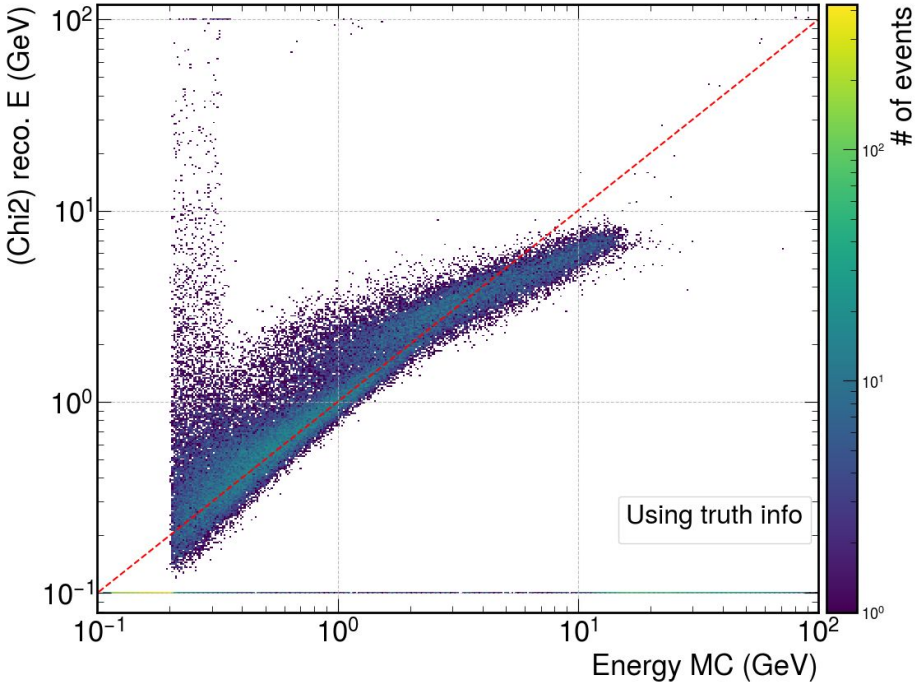


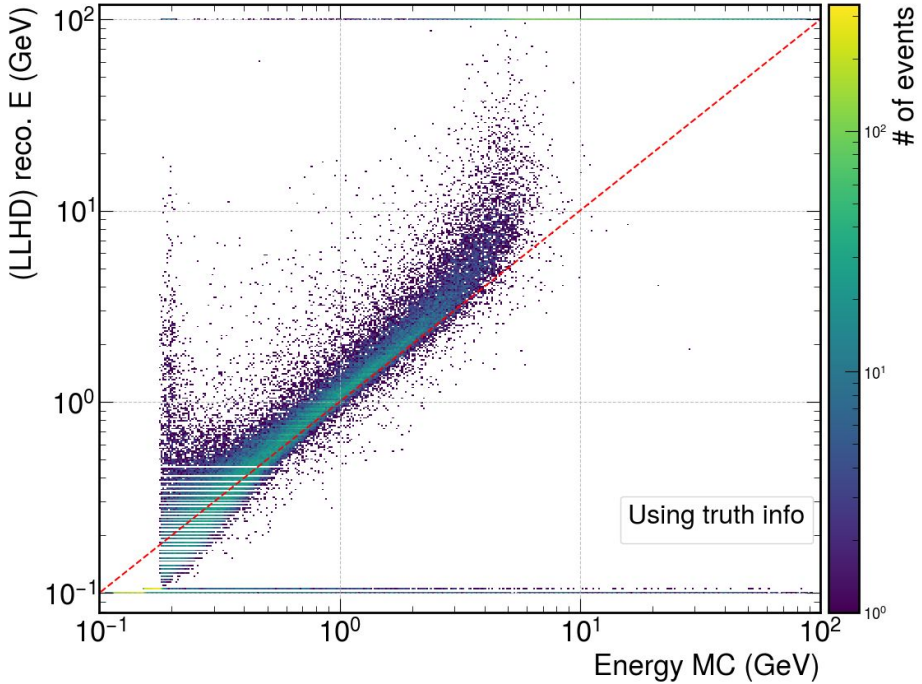
`const simb::MCTrajectory & simb::MCParticle::Trajectory ( )`

Based on studies from uboone, where they used **MCTrack**

True trajectory, current version

- **Validation of methods Chi2 and LLHD using Truth information:**
  - Apply method of Multiple Coulomb Scattering (MCS) in **MCParticle trajectory**
- Results using **current version (v09_91_02d01)** (contained and uncontained)

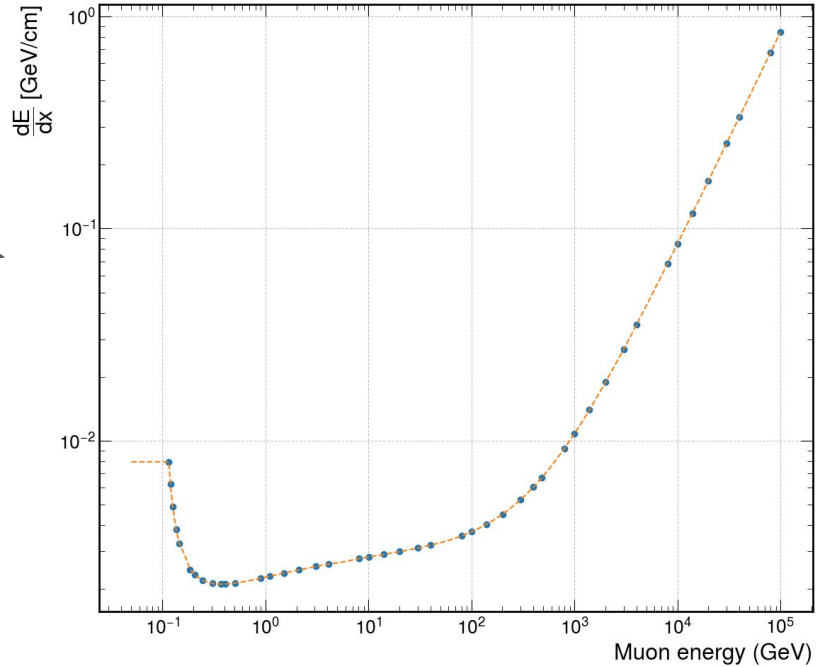


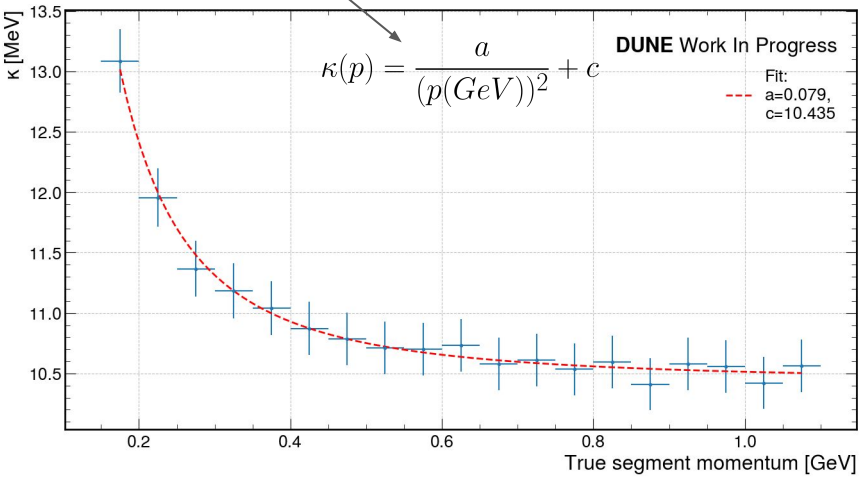


Adjustments

LLHD

- **Implemented better energy loss:**
  - Assuming energy loss following Bethe-Bloch.
  - TSpline3 to retrieve energy lost in every segment of 10 cm
- **Added space angle as option and apply corrections for the fit**
- **Reevaluated Highland formula**





$$\kappa(p) = \frac{a}{(p(GeV))^2} + c$$

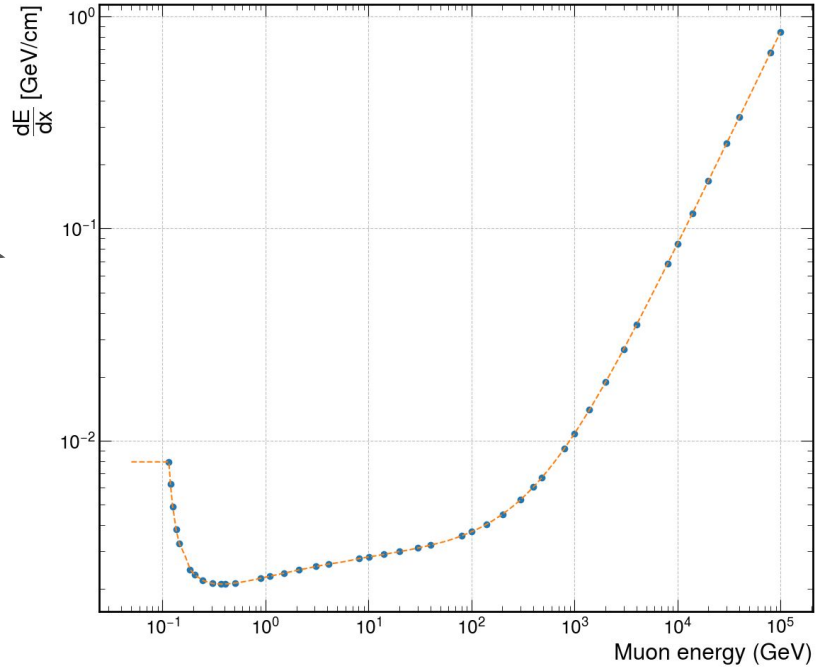
$$\theta_0 = \frac{\kappa(p)}{\beta c p}\, z \sqrt{\frac{x}{X_0}} \left[1 + 0.038 \ln \frac{x z^2}{X_0 \beta}\right]$$

On uboone: a = 0.1049, c = 11.0038

Adjustments

LLHD

- **Implemented better energy loss:**
  - Assuming energy loss following Bethe-Bloch.
  - TSpline3 to retrieve energy lost in every segment of 10 cm
- **Added space angle as option and apply corrections for the fit**
- **Reevaluated Highland formula**



Chi2

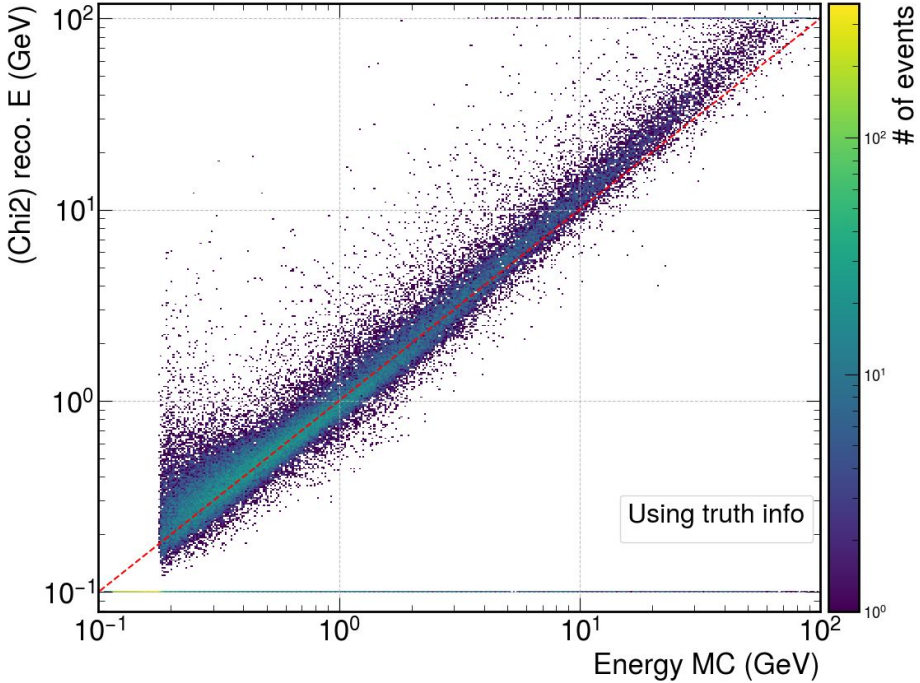
- **Added space angle as option and apply corrections**
- **Adjusted energy addition after fit**
- **Fixed some issues in angle computation**

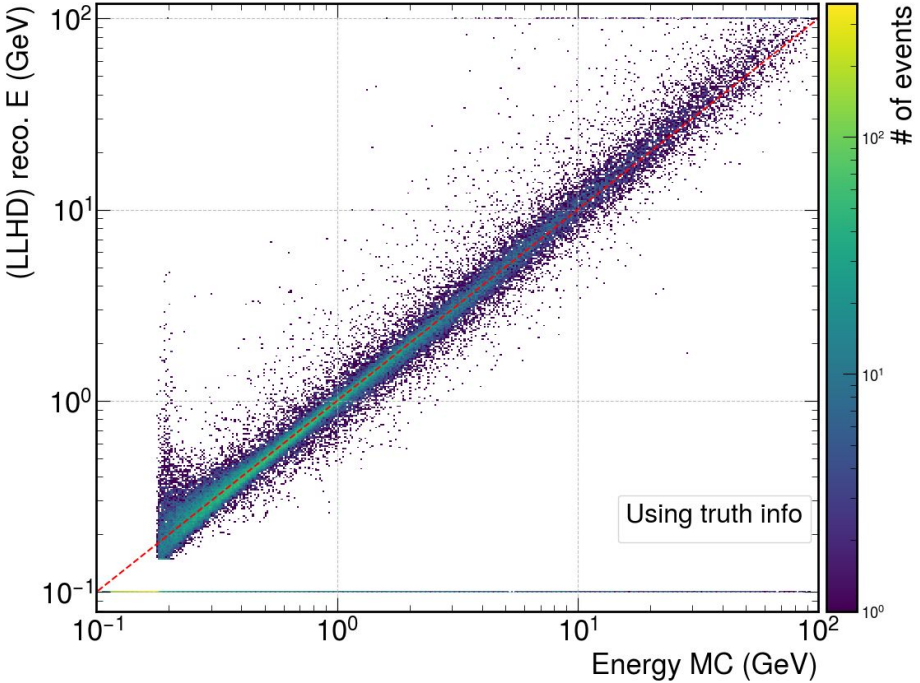
For both

- **(As option) removed fitting of angle resolution**

True trajectory, improved version

- **Validation of methods Chi2 and LLHD using Truth information:**
  - Apply method of Multiple Coulomb Scattering (MCS) in **MCParticle trajectory**
- Results using **new version** (contained and uncontained)



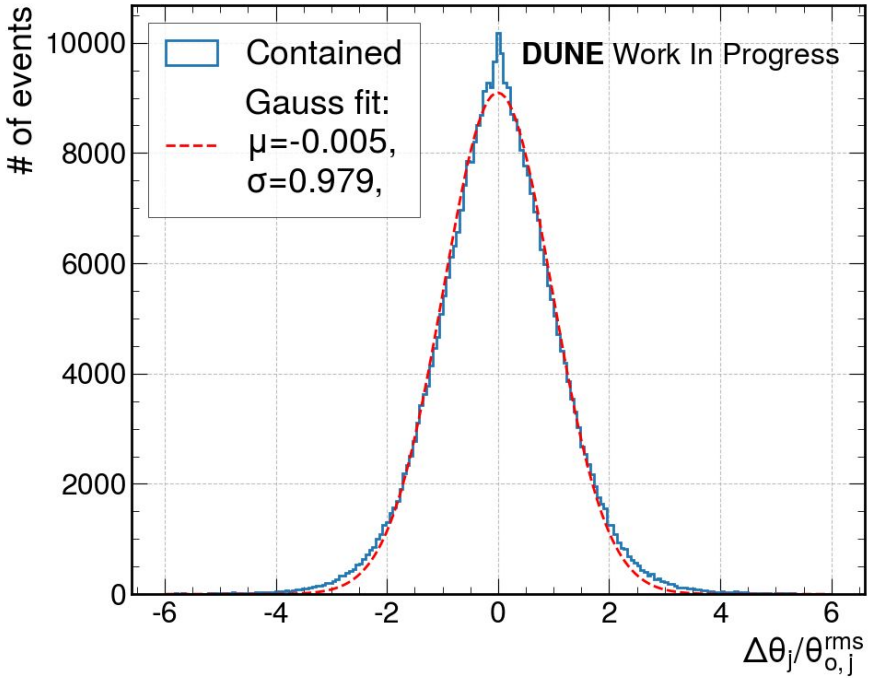


Reconstructed track

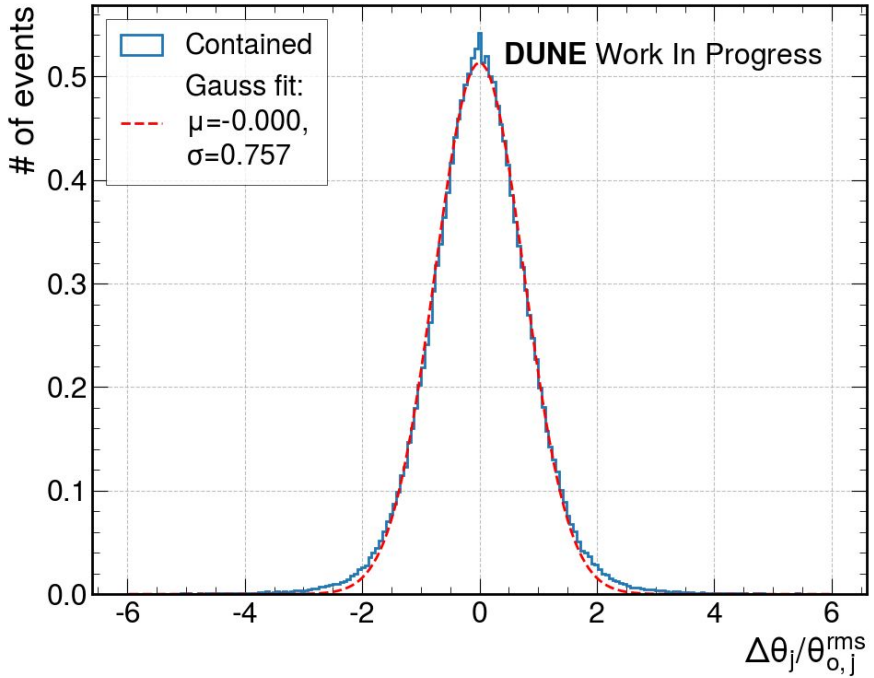
LLHD: adjustment on scatter angle

- Scattered angle smaller for reconstruction
  - Possibly due to over smoothing of tracks
  - Space angles have to be corrected by a factor of x 1/0.757

MC Trajectory

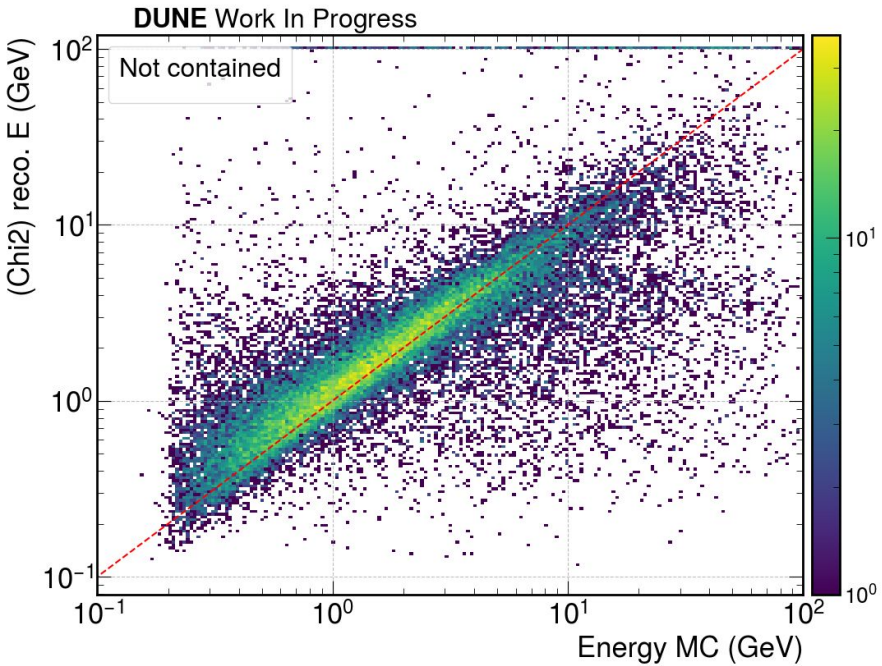


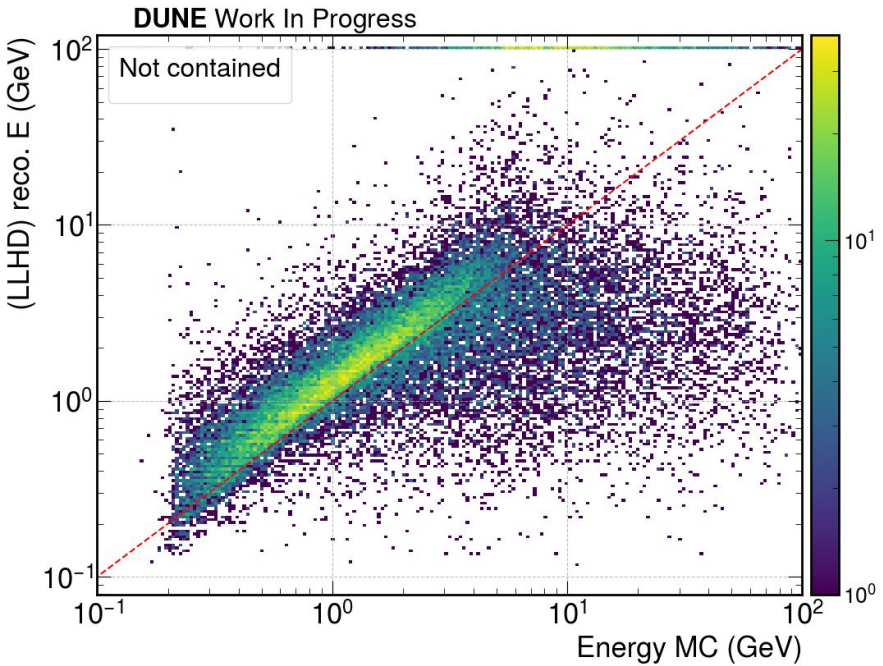
Reconstructed track



Reconstructed energy, current version

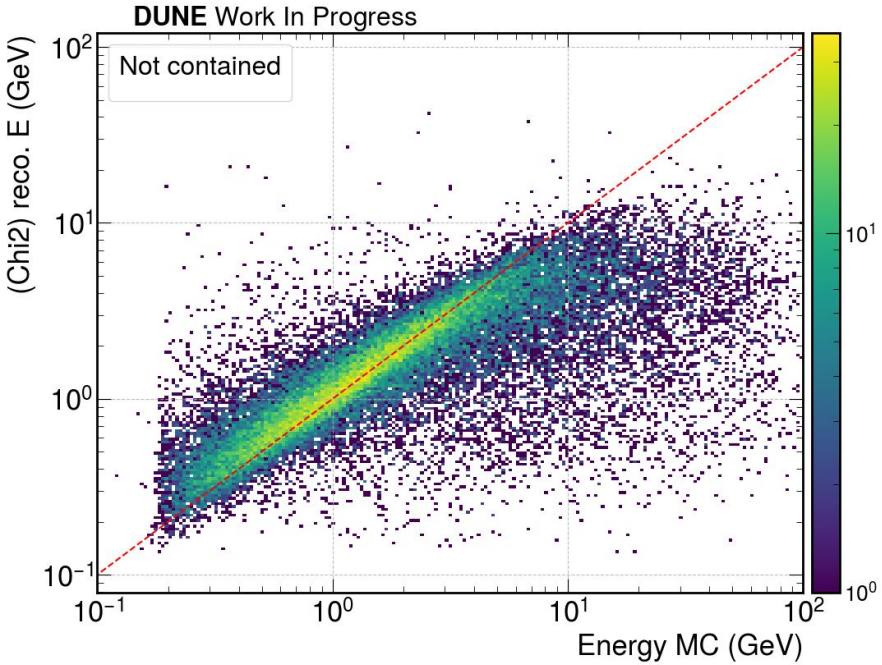
- Results using **current version (v09_91_02d01)**

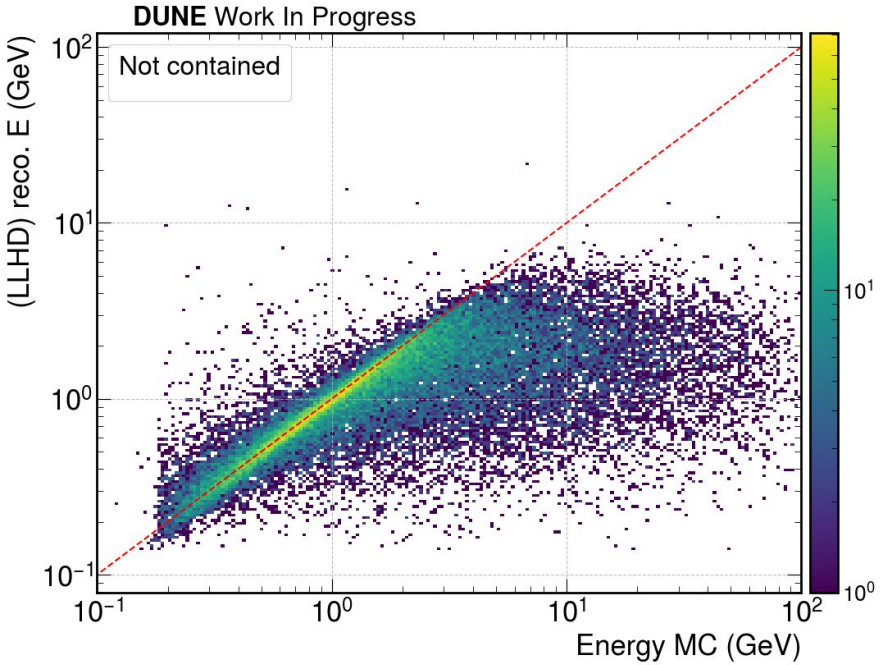




Reconstructed energy, new version

- Results using **new version**





Summary of changes

- Default values were respected:
  - If nothing is passed (standard in all codes), the results will be quite similar

```
TrackMomentumCalculator(double minLength = 100.0,
                        double maxLength = 1350.0,
                        double steps_size = 10.);
```

```
TrackMomentumCalculator(double minLength = 100.0,
                        double maxLength = 1350.0,
                        double steps_size = 10.,
                        int angleMethod = 1,
                        int nsteps = 6);
```

```
double GetMomentumMultiScatterChi2(art::Ptr<recob::Track> const& trk,
                                   const bool checkValidPoints = false,
                                   const int maxMomentum_MeV = 7500);
```

```
double GetMomentumMultiScatterChi2(art::Ptr<recob::Track> const& trk,
                                   const bool checkValidPoints = false,
                                   const int maxMomentum_MeV = 7500,
                                   const double min_resolution = 0,
                                   const double max_resolution = 45);
```

```
double GetMomentumMultiScatterLLHD(art::Ptr<recob::Track> const& trk,
                                   const bool checkValidPoints = false,
                                   const int maxMomentum_MeV = 7500,
                                   const int MomentumStep_MeV = 10,
                                   const int max_resolution = 0);
```

```
double GetMomentumMultiScatterLLHD(art::Ptr<recob::Track> const& trk,
                                   const bool checkValidPoints = false,
                                   const int maxMomentum_MeV = 7500,
                                   const double min_resolution = 0.001,
                                   const double max_resolution = 800,
                                   const bool check_valid_scattered_ = false,
                                   const bool angle_correction_ = 0.757);
```

Conclusions

MC

- Original algorithms do not perform well using Monte Carlo Trajectory
- **After adjustments,** both methods (LLHD and Chi2) perform well over all energy range

Reco

- Major improvement in **LLHD** method
- Minor improvement in **Chi2** method

- **Updates in DUNERECO** made to allow changing parameters
  - pull request depending on **LARRECO**

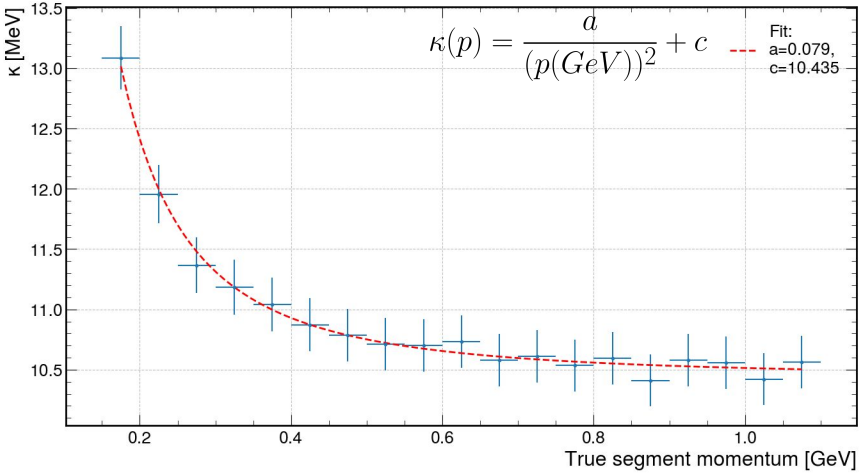
Backups

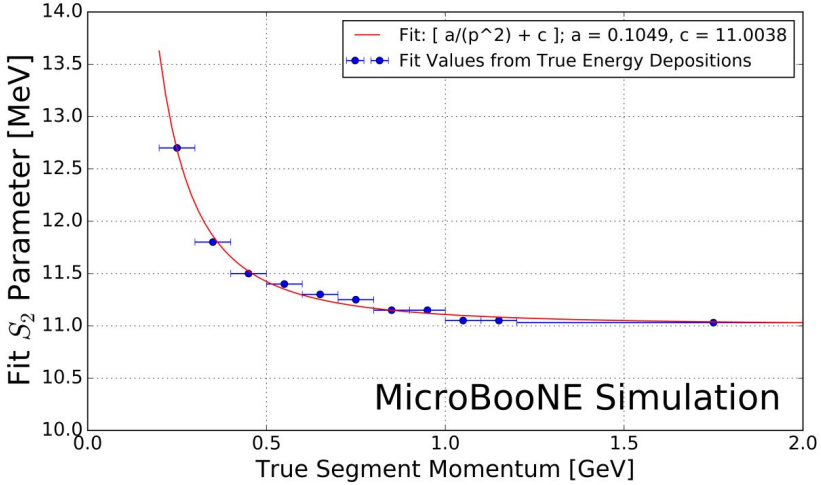
Adjustments

LLHD

- **Reevaluated Highland formula**

$$\theta_0 = \frac{\kappa(p)}{\beta c p}\, z \sqrt{\frac{x}{X_0}} \left[1 + 0.038 \ln \frac{x z^2}{X_0 \beta}\right]$$





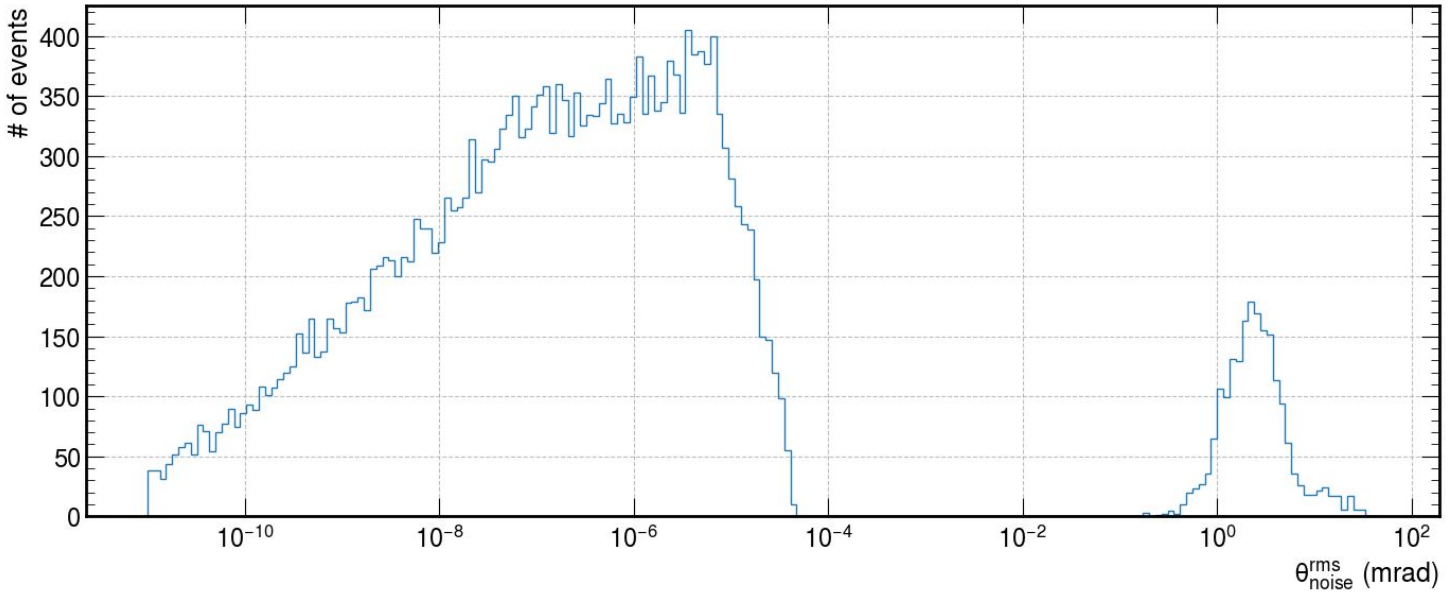
On uboone: a = 0.1049, c = 11.0038

How does the resolution impacts the fit?

- In both methods, a noise is added to the rms due to the detector resolution as:

$$(\theta_{\text{meas}}^{\text{rms}})^2 = (\theta_0^{\text{rms}})^2 + (\theta_{\text{noise}}^{\text{rms}})^2 ,$$

- For **uboone** this was set to 2 mrad. We are fitting it in the Chi2 method.
- For **Chi2,** there is no major impact as the noise results to zero in most of the events

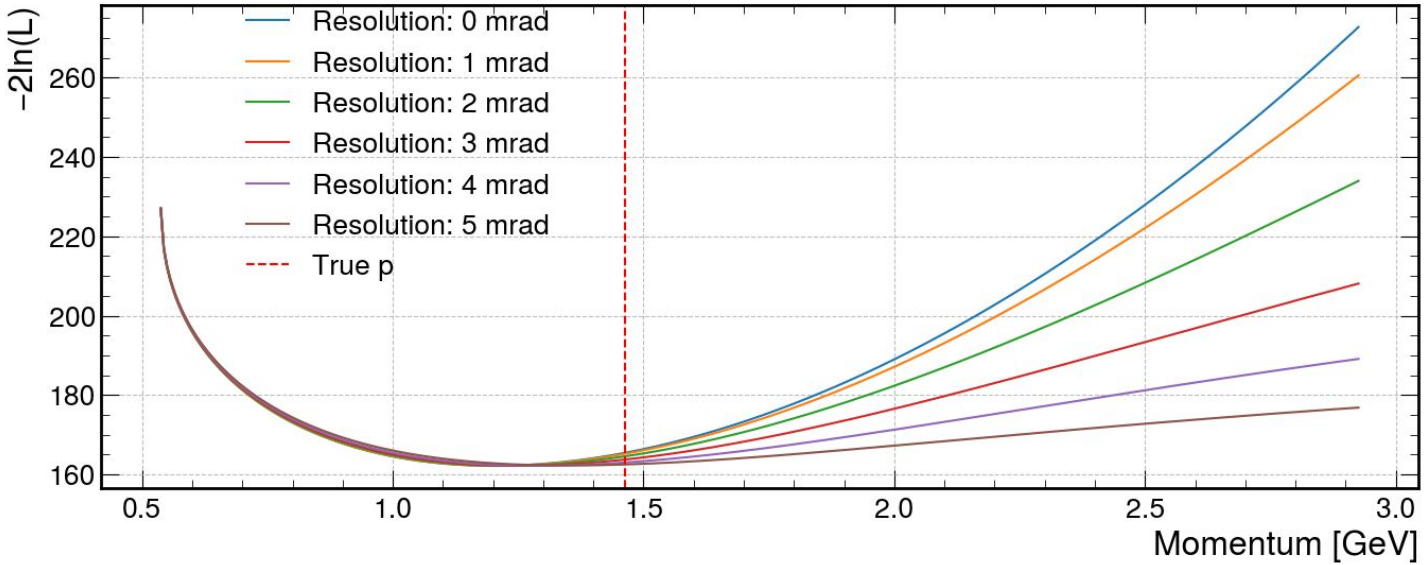


How does the resolution impacts the fit?

- In both methods, a noise is added to the rms due to the detector resolution as:

$$(\theta_{\text{meas}}^{\text{rms}})^2 = (\theta_0^{\text{rms}})^2 + (\theta_{\text{noise}}^{\text{rms}})^2 ,$$

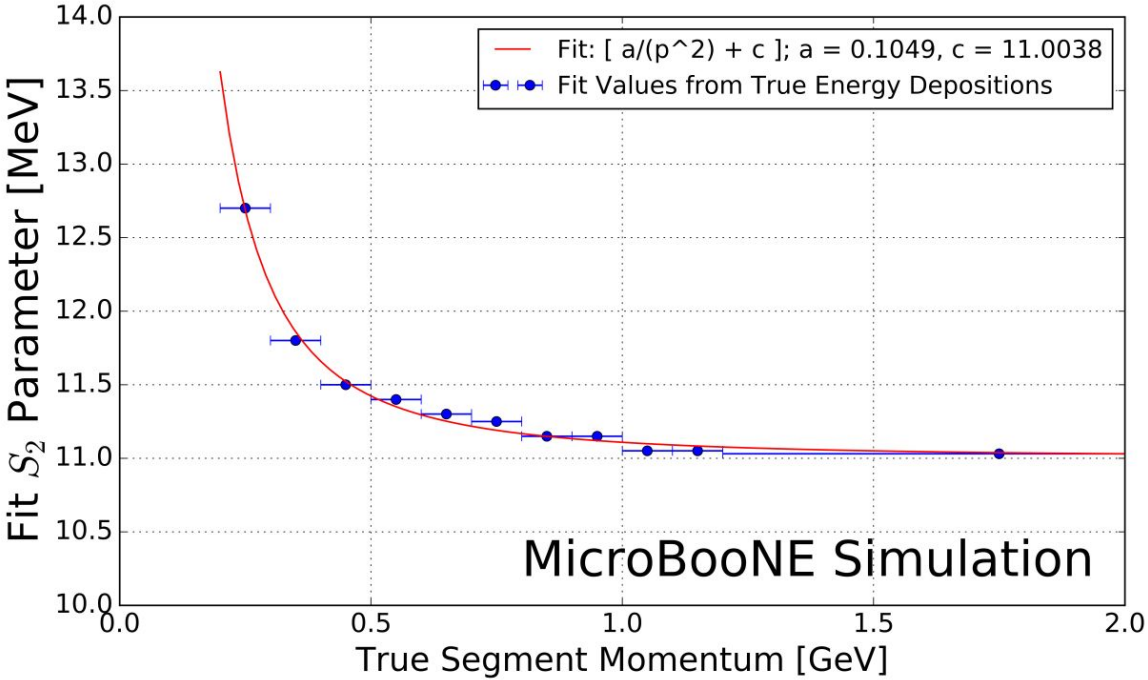
- For **uboone** this was set to 2 mrad. We are fitting it in the Chi2 method.
- For **Chi2,** there is no major impact as the noise results to zero in most of the events
- For **LLHD,** the noise **should not be fitted** because it has a major impact in the fit:



Uboone adjustments

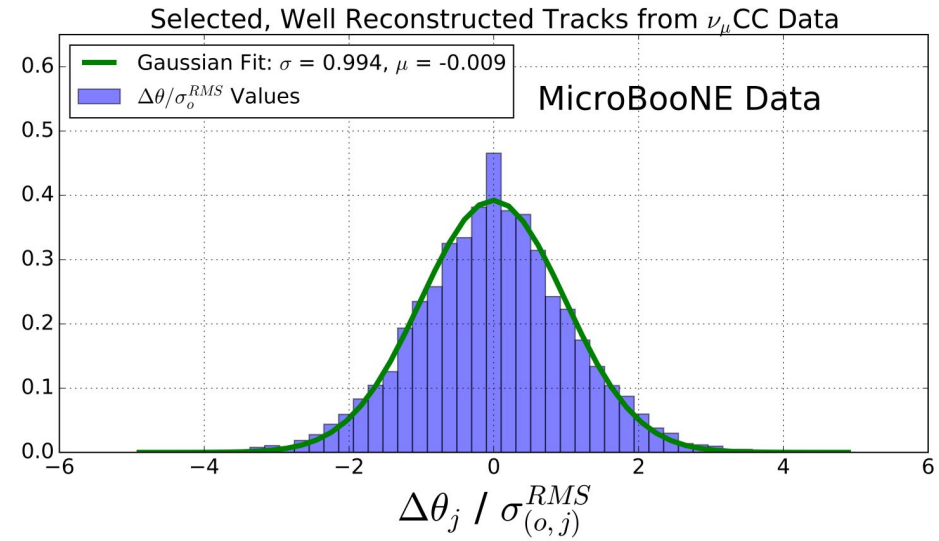
Values depend on the method used to extract scattered angle.

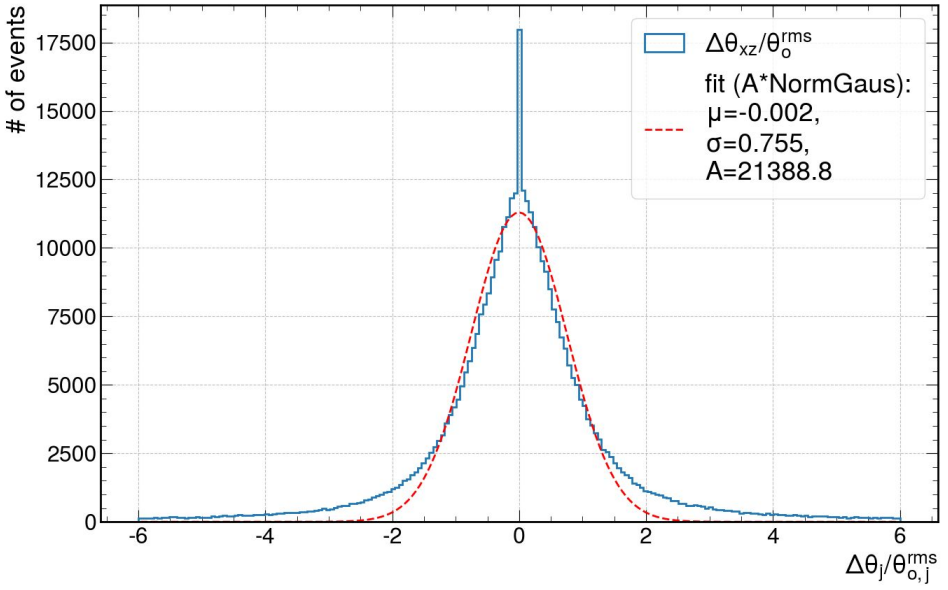
It also depends on the segment length



Are we over smoothing tracks?

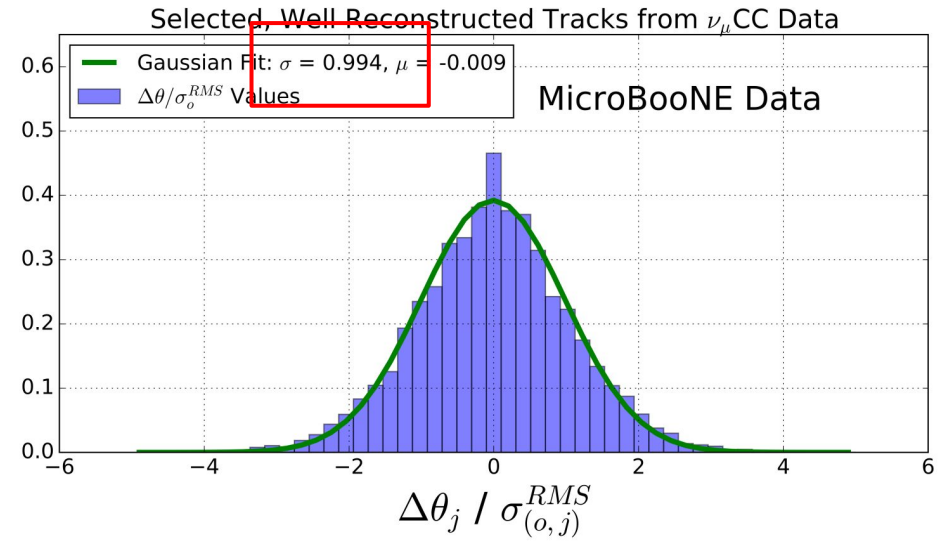
- In last presentation (ages ago), it was suggested that we might be over fitting the tracks:
  - **Scatter angles are often smaller than what we expect**

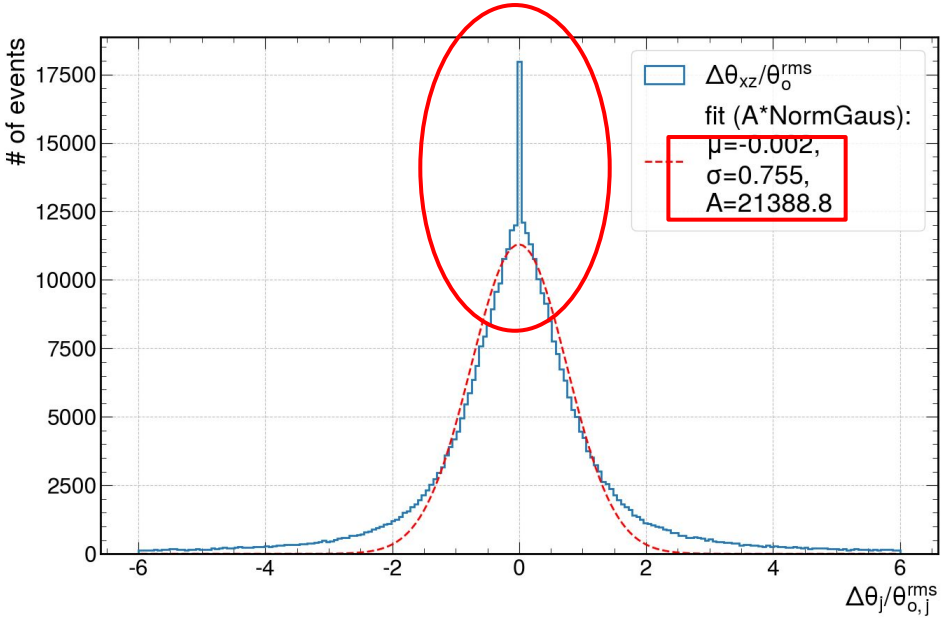




Are we over smoothing tracks?

- In last presentation (ages ago), it was suggested that we might be over fitting the tracks:
  - **Scatter angles are always smaller than what we expect**





Can be over fitting of tracks or the method that retrieves scatter angles (bad reconstruction also affects it)

Where can we improve?

- **Better selection of muon track candidate**