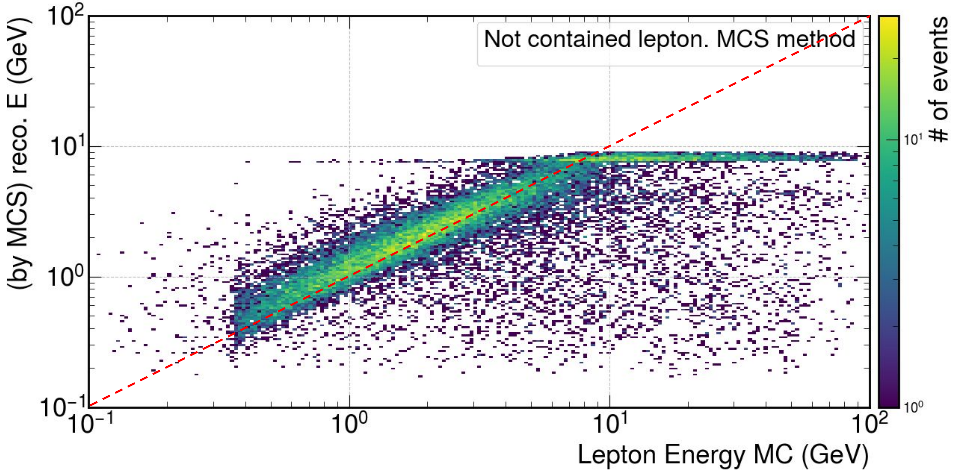
Atmospheric Neutrino Energy and Angle Reconstruction

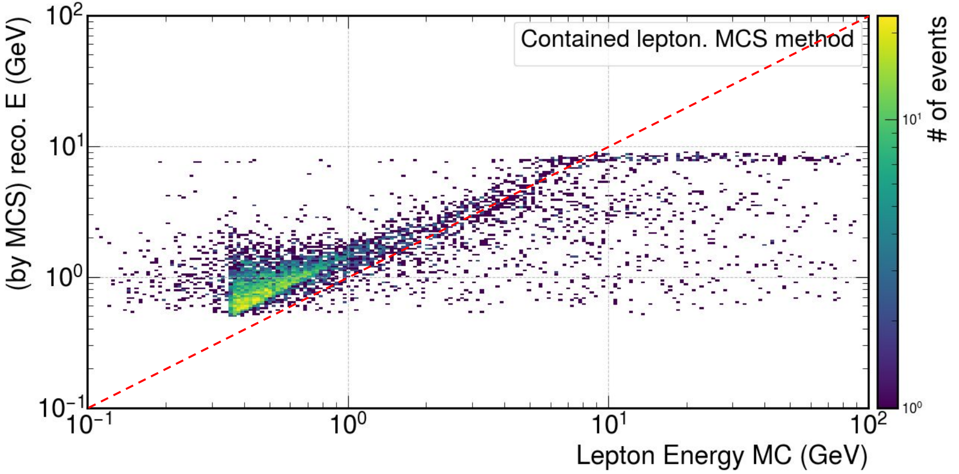
Henrique Souza, Pierre Granger

06/11/2023

Energy reconstruction

- **Last meeting slides:** link
- **CM slides:** link
- **Multiple Coulomb Scattering (MCS)**:
  - Why values are stopping before 10 GeV ?
  - Why MCS method is (slightly) overestimating the muon energy?
  - Why do we have a poor resolution?
  - What should be an appropriate threshold to choose range or mcs?





Understanding MCS method

- There are two methods of MCS, which I denote here **Chi2** and **LLHD** (next slides)
- **Chi2,** examples of use: in Opera and ICARUS:
  - Opera
  - ICARUS
- **LLHD,** examples of use in uboone and ProtoDUNE
  - uboone
  - ProtoDUNE
- **Current state of LArSoft:**
  - Only Chi2 method is used.
  - LLHD is not fully implemented
  - Codes have basically no comments (took me a while to understand them)

- Outline:
  - How Chi2 and LLHD methods work
  - How they are implemented and results

How to retrieve momentum?

- Particle scattering is modeled with a Gaussian distribution centered at zero and RMS given by the Highland formula:

$$\theta_0 = \theta_{\text{plane}}^{\text{rms}} = \frac{1}{\sqrt{2}} \theta_{\text{space}}^{\text{rms}} ,$$

$$\theta_0 = \frac{13.6 \text{ MeV}}{\beta c p} z \sqrt{\frac{x}{X_0}} \left[1 + 0.038 \ln\left(\frac{x z^2}{X_0 \beta^2}\right) \right]$$

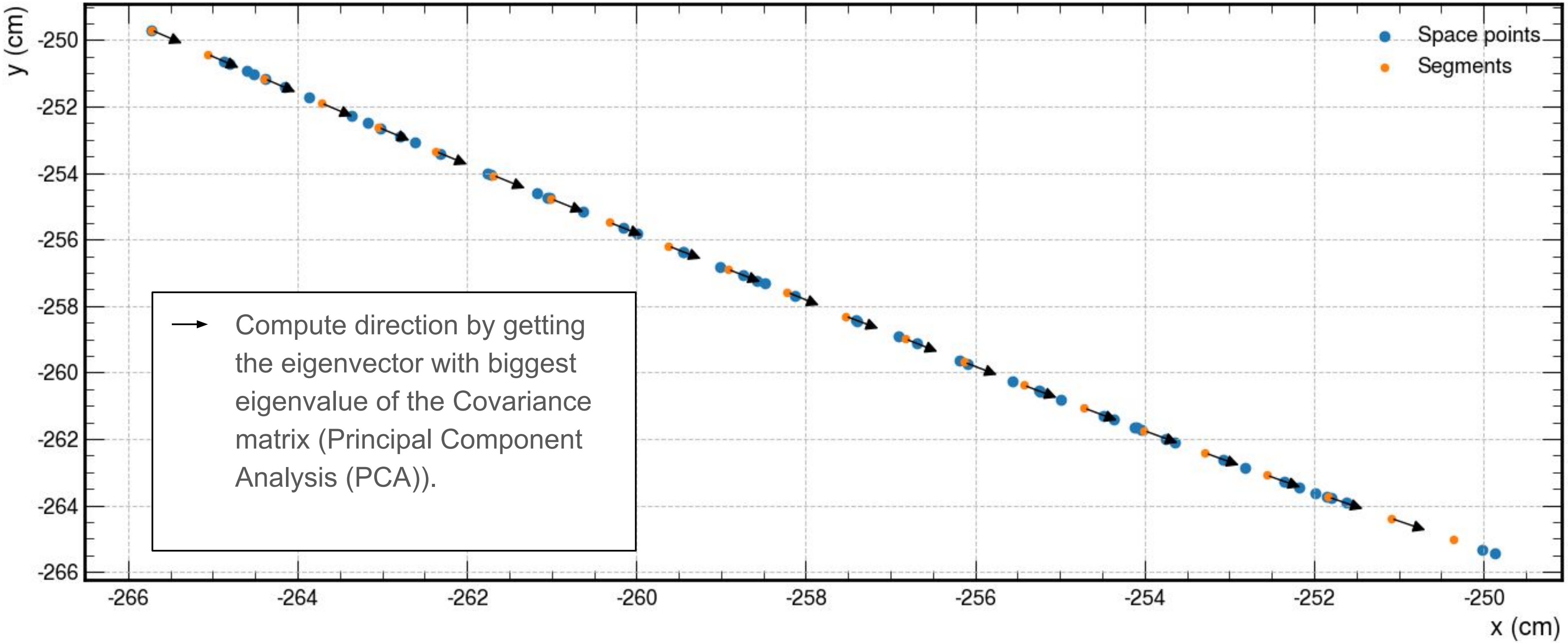
- Muons: z = 1
- LAr: Xo = 14 cm
- x: segment length

- **Chi2:**
  - Evaluates θ_0 (the rms of scattered angles) for different segment lengths
  - Find momentum by minimizing measurement vs expected value
- **LLHD:**
  - Evaluates the log-likelihood for each segments scattered angle and minimize it.
- **More details in backup slides!**

Implementation: Chi2 and LLHD

- **First step** (Common to **Chi2** and **LLHD**):
  - Evaluate particle direction breaking into segments of fixed length
  - All space points inside a segment are stored for next step

In the example:
Segments of 1 cm for visualization



→ Compute direction by getting the eigenvector with biggest eigenvalue of the Covariance matrix (Principal Component Analysis (PCA)).

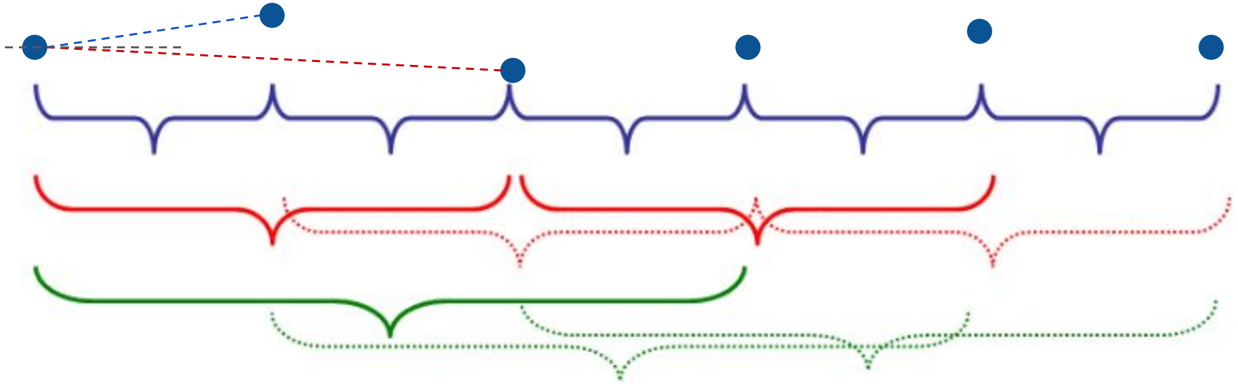
Implementation: Second step – Chi2

- **Second step (Chi2)**
  - Evaluate the RMS for different segments lengths:

10 cm segments

20 cm segments

30 cm segments



Remember:

Each blue point is the segment computed in the first step

Computes the scattered angle rms (θ_o^{RMS}) using 5 different points

Computes the scattered angle rms (θ_o^{RMS}) using 4 different points

Computes the scattered angle rms (θ_o^{RMS}) using 3 different points

In this example: we would have 3 points to fit

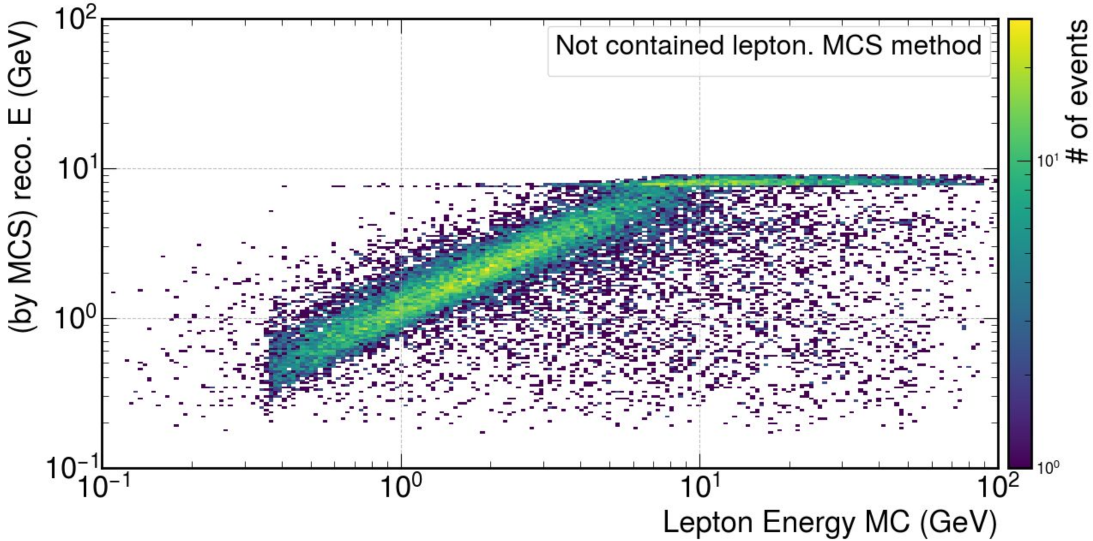
Implementation: comments on Second step – Chi2

- **The computation of rms is done twice.**
  - First time it takes all the points
  - A second one accepting only the points within 2.5σ (98%)

- **In current version of LArSoft**:
  - The segment length of 10 cm was hardcoded.
  - The number of steps is hard coded to 6 (so 6 points to be fitted only)
  - Only the xz-plane is been used
  - (side note) A systematic uncertainty of 5% rms is added quadratically with the notes: “Systematic error to fix chi2 behaviour”

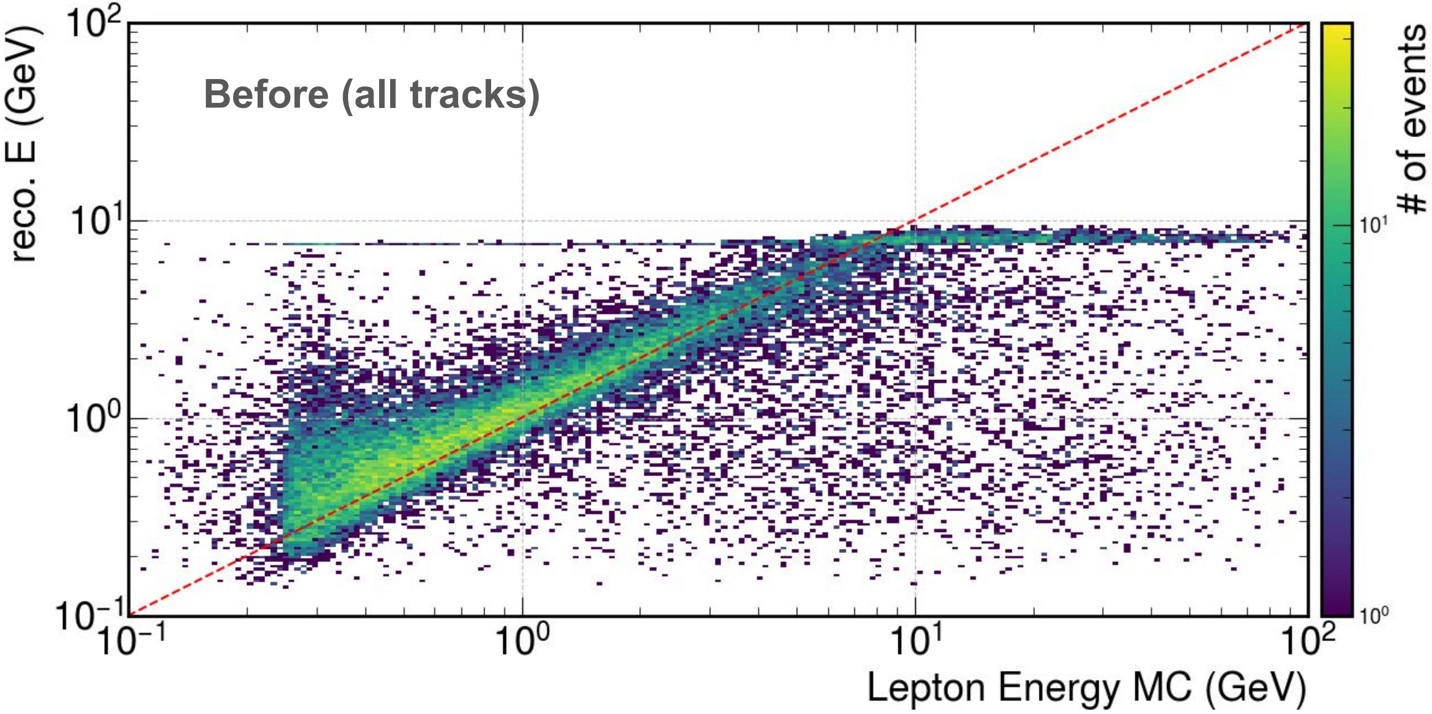
Implementation: Final step – Chi2

- **Minimize:** $\chi^2 = \sum_i \left(\frac{\theta^{\mathrm{rms}}_{\mathrm{meas}\,i} - \theta^{\mathrm{rms}}_{\mathrm{theo}\,i}}{\Delta\theta^{\mathrm{rms}}_{\mathrm{meas}\,i}} \right)^2 ,$ **assuming:** $(\theta^{\mathrm{rms}}_{\mathrm{meas}})^2 = (\theta^{\mathrm{rms}}_{0})^2 + (\theta^{\mathrm{rms}}_{\mathrm{noise}})^2 ,$
- Assumes **Beta** = 1 for all energies
- **The momentum and noise rms are fitted at the same time**
  - There was a penalty added to chi2 as: Chi2 += 2 * $\theta^{\mathrm{rms}}_{\mathrm{noise}}$ /4.6
  - However, the noise rms was constrained between 0 to 45 mrad.
    - This cause the fit to return rms noise << 10^-4 mrad **(check backup slides)**
- **Momentum is constrained between 10 MeV and 7.5 GeV**
- The energy lost by track was added at the end by
  - taking the length at the last segment
  - times 2.4 MeV/cm
  - divided by two.
    (Added to the momentum)



Changes in Chi2 method

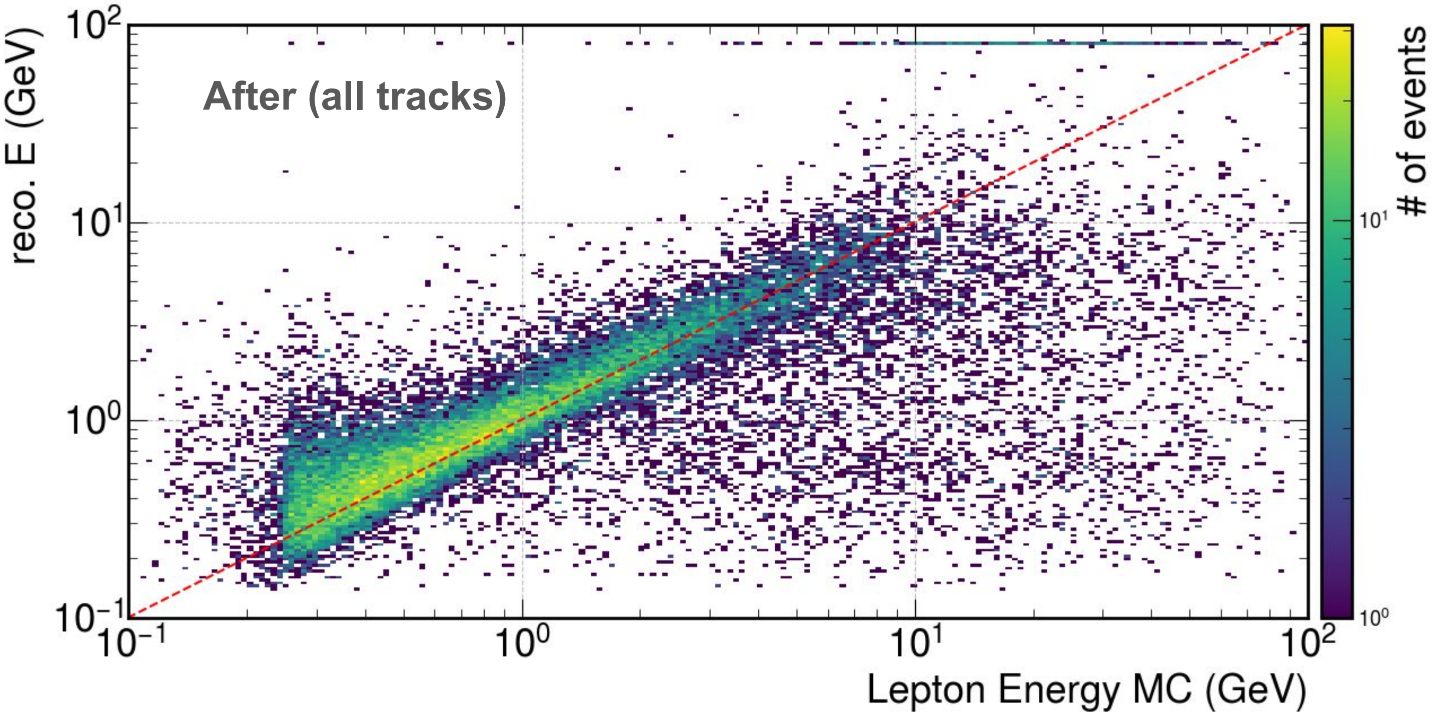
- **Trying to improve:**
  - Fix $\theta^{\mathrm{rms}}_{\mathrm{noise}}$ to 2 mrad (more study necessary)
  - Removing penalty
  - Changed upper limit from 7.5 to 80 GeV
  - Change minimum length from 100 to 30 cm
  - Increase to 18 segments, starting at 5 cm and steps of 2.5 cm



For not contained tracks, check backup

Changes in Chi2 method

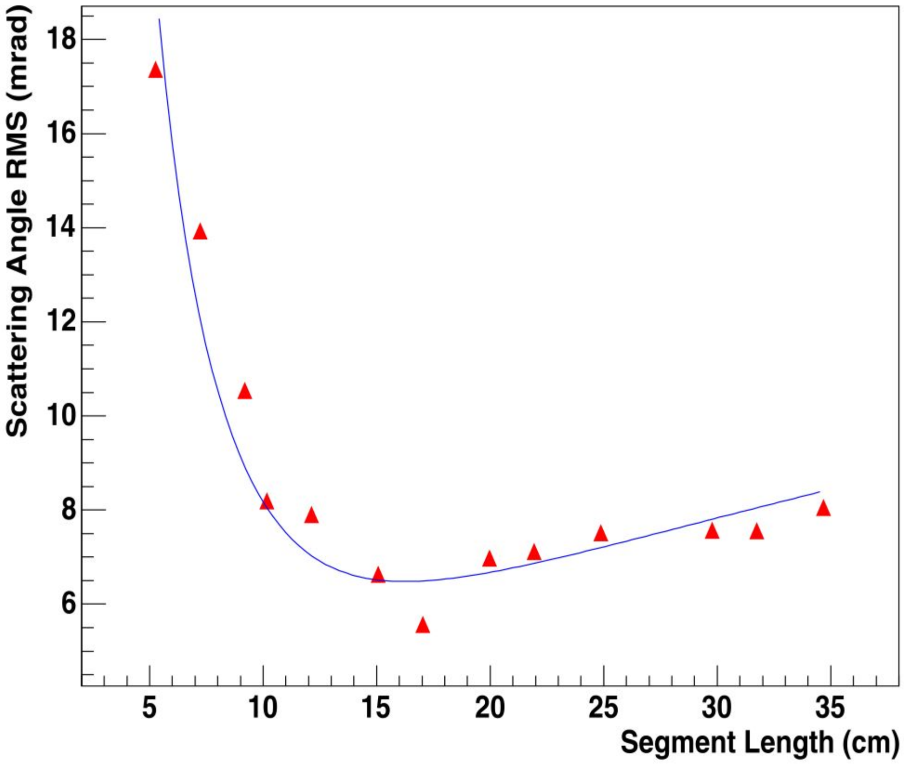
- **Trying to improve:**
  - Fix $\theta_{\rm noise}^{\rm rms}$ to 2 mrad (more study necessary)
  - Removing penalty
  - Changed upper limit from 7.5 to 80 GeV
  - Change minimum length from 100 to 30 cm
  - Increase to 18 segments, starting at 5 cm and steps of 2.5 cm



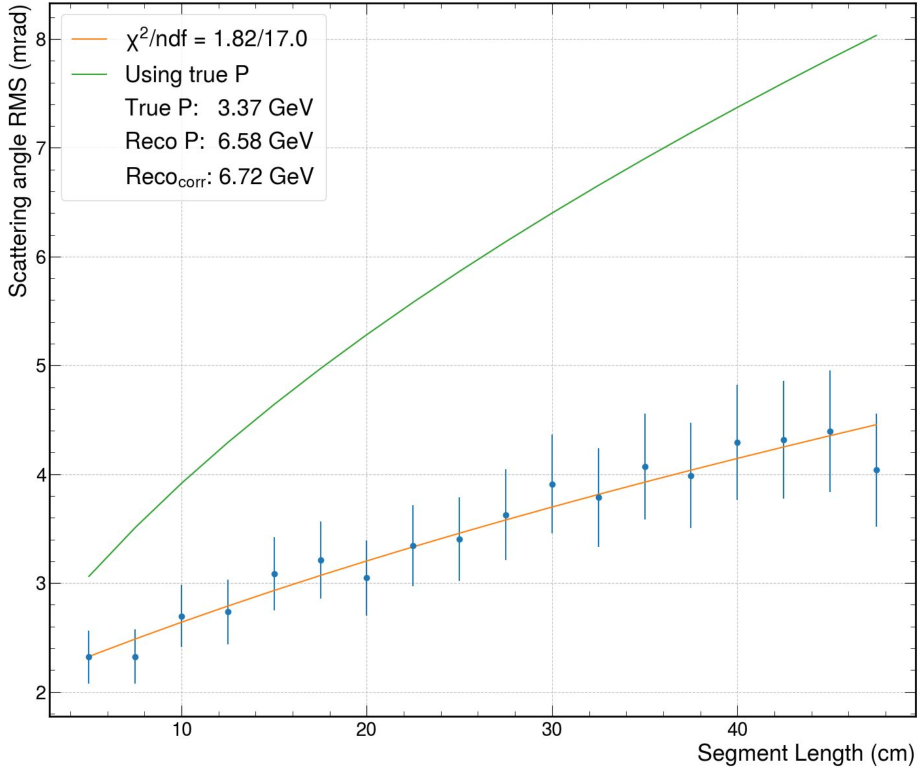
For not contained tracks, check backup

Quick note (more examples as backup)

- **I was expecting a similar plot to ICARUS:**



- **The result was this instead:**



- **Any comments on this?**

Implementation: Last Steps LLHD

- **Correction (LLHD)**
  - Instead of using the formula:

$$\theta_0 = \frac{13.6\ \text{MeV}}{\beta c p}\, z \sqrt{\frac{x}{X_0}} \left[1 + 0.038 \ln\left(\frac{x\, z^2}{X_0 \beta^2}\right)\right]$$

Uboone paper present a simulation based tuning of the **13.6 MeV** constant, **replacing it** by:

$$\kappa(p) = \frac{0.105\ \text{MeV}}{(p(\text{GeV}))^2} + 11.004\ \text{MeV}.$$

- **Last step:**
  - Minimize for all segments and $\Delta\theta_{xz}$ and $\Delta\theta_{yz}$. (More comments on this)
  - For each segment, compute the momentum considering the energy lost as 2.1 MeV/cm

$$-l(\sigma_{o,1}, \ldots, \sigma_{o,n}; \Delta\theta_1, \ldots, \Delta\theta_n) = -\ln(L) = \frac{n}{2}\ln(2\pi) + \sum_{j=1}^{n} \ln(\sigma_{o,j}) + \frac{1}{2}\sum_{j=1}^{n}\left(\frac{\Delta\theta_j}{\sigma_{o,j}}\right)^2$$

Implementation: comments on LLHD

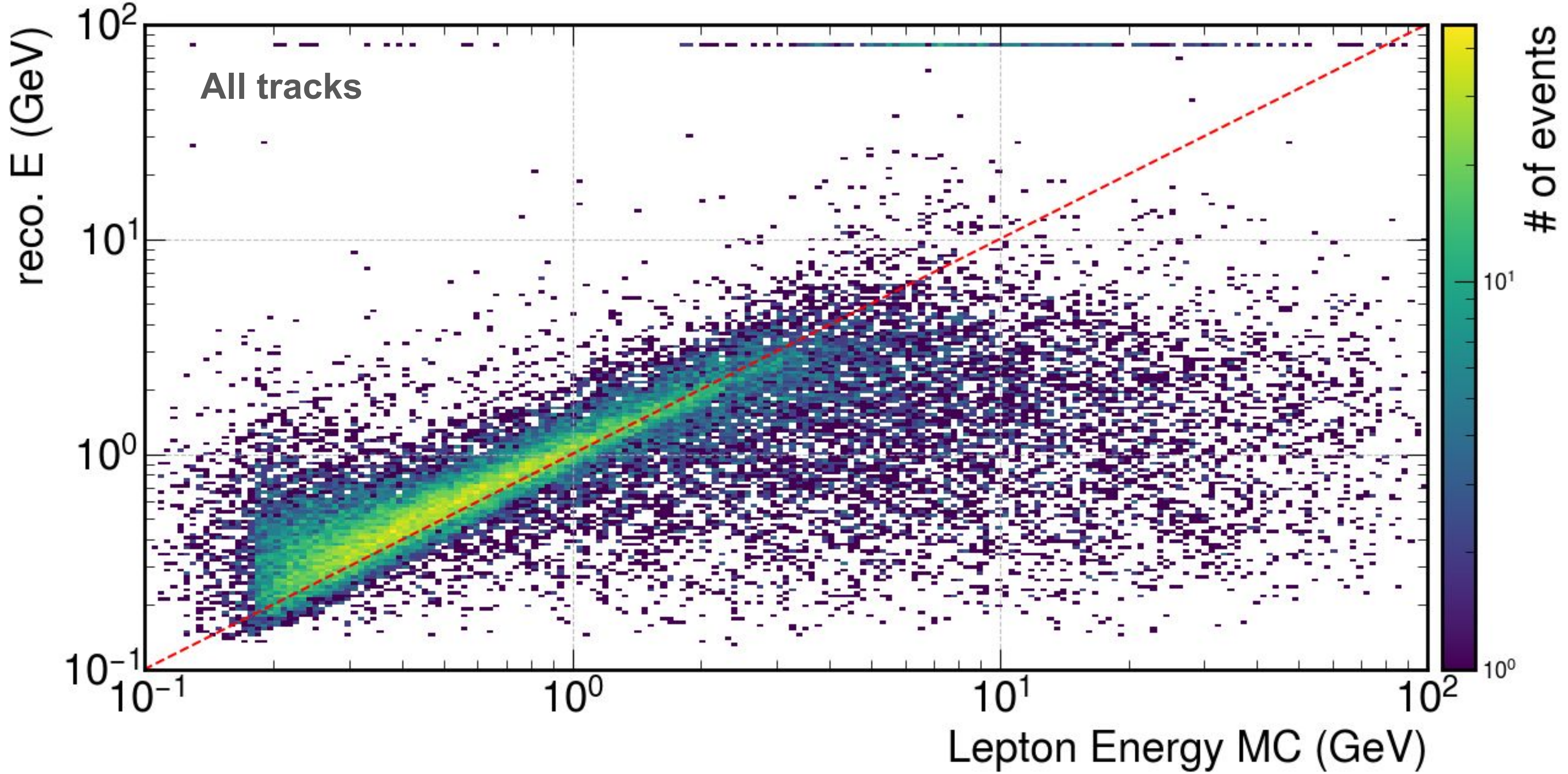
- **In current version of LArSoft**
  - LLHD method is **partially** implemented. **The code as it is does not work.**
  - We found the git repository from which the authors of the uboone paper derived their results:
    - https://github.com/kaleko/KalekoAna/tree/master
  - **Their version differed from current LArSoft by:**
    - Applying a verification on hits to avoid invalid points (I recently added it to larreco)
    - Using both angles $\Delta\theta_{xz}$ and $\Delta\theta_{yz}$ to perform a fit
    - Keeping beta in analytical form as function of p and m (p/E = B/c) -> …
    - Adding the tuning K(p) described on the paper
    - Scan values of p with steps of 10 MeV (no fit)

- **What have I changed:**
  - **Instead** of **scanning values,** I implemented the minimization
  - I set a different minimum momentum for each track depending on the number of segments. This is done to prevent p = 0 in the highland formula
    - Original code was returning 999999 to the fittier instead, making it not a smooth function and failing quite often.

Implementation: LLHD

- **Result obtained with segments of 10 cm**

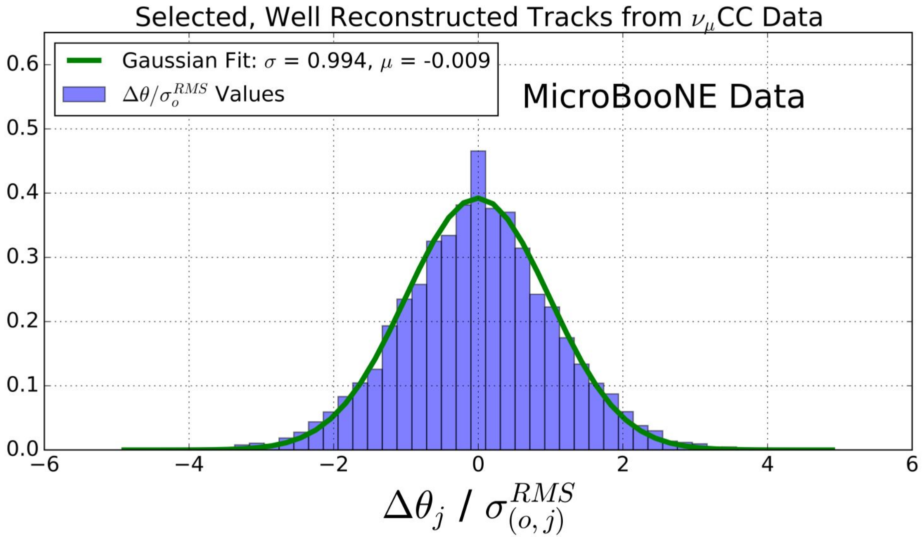
For not contained tracks, check backup



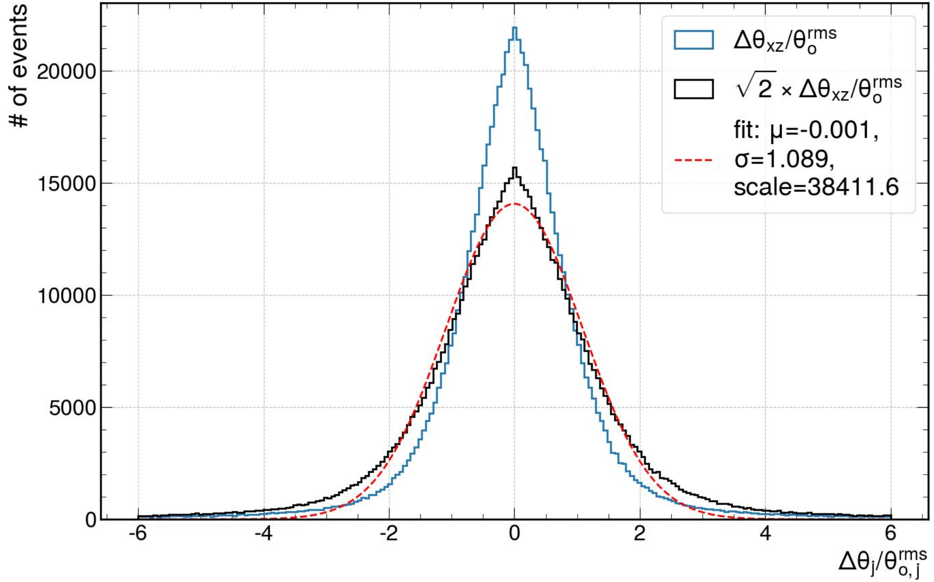
Implementation: comments on LLHD

- LLHD was done using **segments of 10 cm**
- $\Delta\theta_j$ values are lower than expected from θ_{rms}.

- **I was expecting this from uboone paper:**



- **Got this instead** (xz-plane)

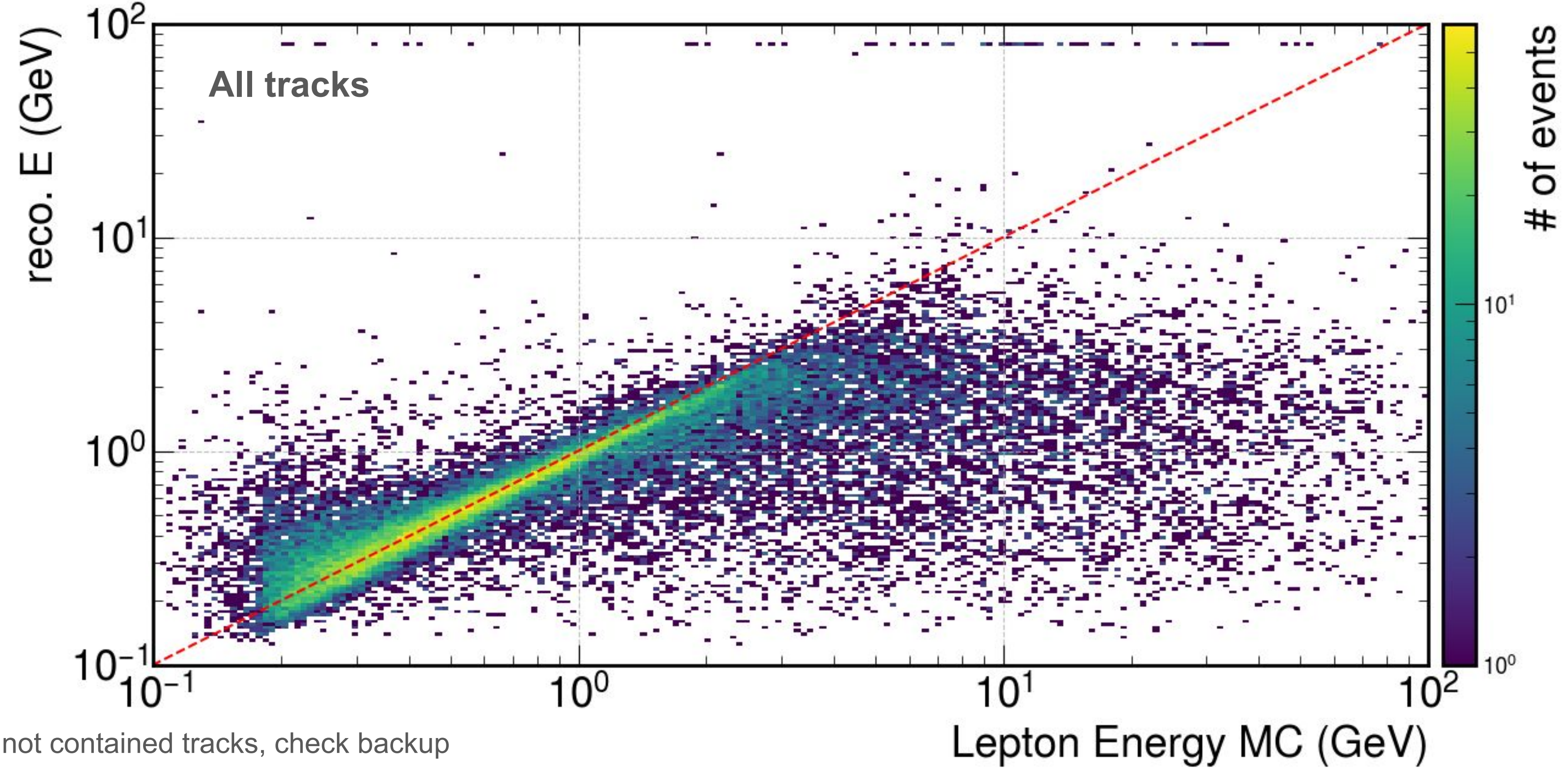


yz-plane as backup

Implementation: comments on LLHD

- **Using space angle (without √2) - $\Delta\theta_{space} = \sqrt{(\Delta\theta_{xz}^2 + \Delta\theta_{yz}^2)}$**

$$\theta_0 = \theta^{rms}_{plane} = \frac{1}{\sqrt{2}} \theta^{rms}_{space} \, ,$$



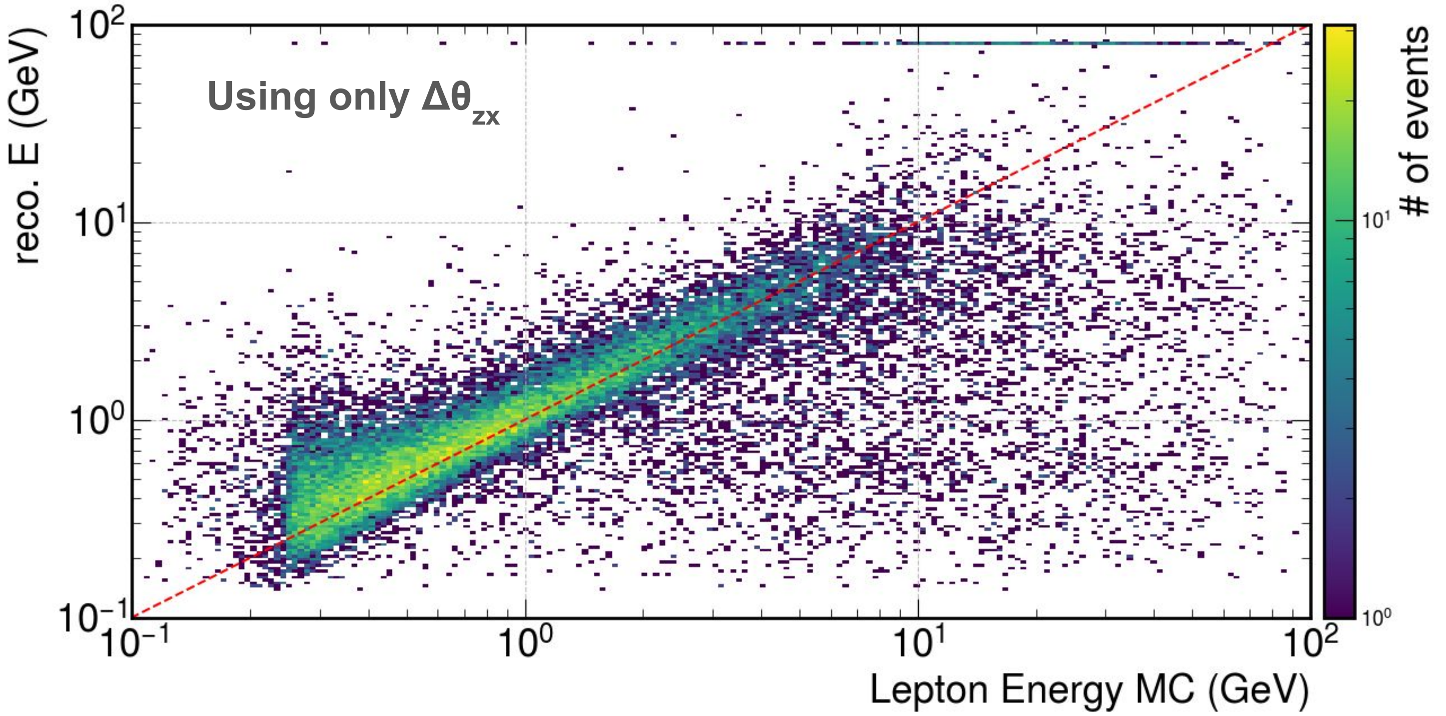
For not contained tracks, check backup

MCS method

- **BACK TO Chi2**
  - Try to use space angle

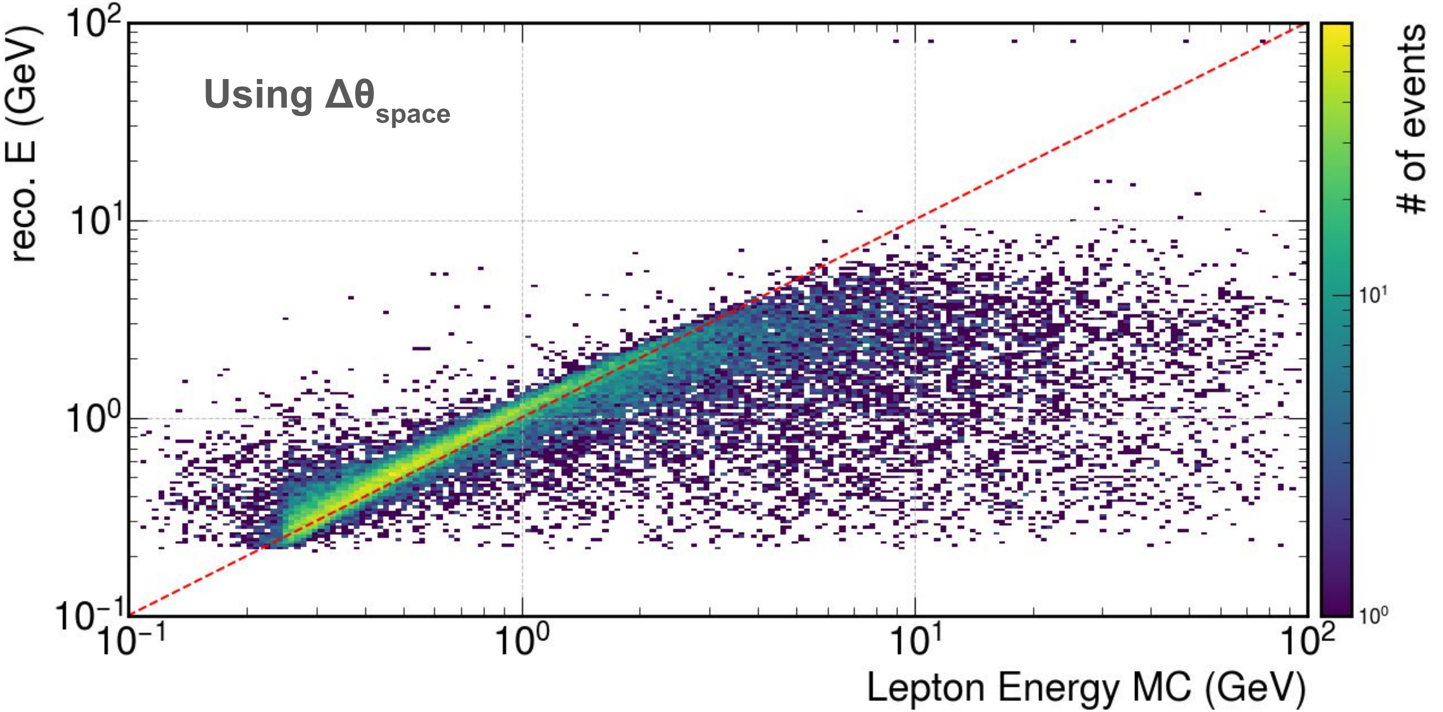
MCS method - Chi2 space angle

- Using $\Delta\theta_{space}$ **(with √2 correction)**
- and adding energy lost by track (instead of adding half of it to the momentum)



MCS method - Chi2 space angle

- Using $\Delta\theta_{space}$ **(with √2 correction)**
- and adding energy lost by track (instead of adding half of it to the momentum)



MCS method - Summary

- **Multiple Coulomb Scattering (MCS)**:
  - Why values are stopping before 10 GeV ? **(solved)**
  - Why MCS is method is (slightly) overestimating the muon energy? **(“solved”, added energy)**
  - Why we have a poor resolution? **(solved, llhd has narrower response)**
  - What should be an appropriate threshold to choose range or mcs? **(forgot about it)**
- **Main points:**
  - **Why do we have smaller Δθ ?**
  - **(Chi2) why we add half of the energy lost to the momentum at the end?**
  - (Chi2) Should tuning also be applied ?
  - (LLHD) gaps in the tracks are interpolated to make segments of fixed size. These leads to very small Δθ that might impact the fit. Need to check if it improves
  - **(Chi2 and LLHD) plane or space angle?**
- **Ongoing (todo):**
  - New sample of only CC numu and nue events, with flat En. distributions: half sample between 100 MeV to 10 GeV, and half from 100 MeV to 20 GeV
  - Small sample of muon only
  - Need to come to a conclusion of what method/parameters to use

Angle reconstruction

- **First (very basic) version of angle reconstruction has been implemented**
  - The producer can be run for nue or numu reconstruction
  - For numu:
    - it reconstructs the momentum of the neutrino using the direction of the longest track
  - For nue:
    - it uses the direction of the shower with the highest charge deposition
  - Output: is the normalized reconstruction directions

Backup

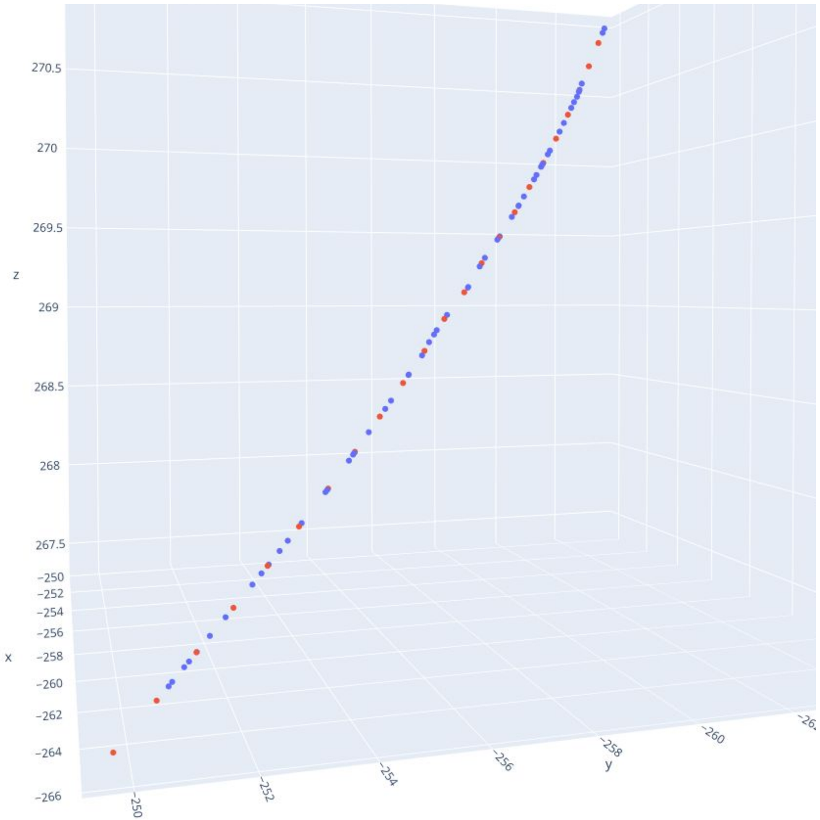
Implementation: Chi2 and LLHD

- **First step** (Common to **Chi2** and **LLHD**):
  - Compute direction by getting the eigenvector with biggest eigenvalue of the Covariance matrix (Principal Component Analysis (PCA)).

This is done in three dimension!

For all next steps:

- **z** coordinate becomes the particle direction $\mathbf{r}_{\mathbf{particle}}$
- **y** is taken by finding the normal between $\mathbf{r}_{\mathbf{particle}}$ **-** $\mathbf{x}_{\mathbf{TPC}}$ **plane**
- **x** is the cross product between **y** and **z**



MCS method - How to retrieve momentum

- "For many small-angle scatters the net scattering and displacement distributions are Gaussian via the central limit theorem. Less frequent "hard" scatters produce non-Gaussian tails. These Coulomb scattering distributions are well-represented by the theory of Molière". The Highland formula for the scattering angle rms is:

$$\theta_0 = \theta_{\text{plane}}^{\text{rms}} = \frac{1}{\sqrt{2}} \theta_{\text{space}}^{\text{rms}} ,$$

$$\theta_0 = \frac{13.6 \text{ MeV}}{\beta c p} z \sqrt{\frac{x}{X_0}} \left[1 + 0.038 \ln\left(\frac{x z^2}{X_0 \beta^2}\right) \right]$$

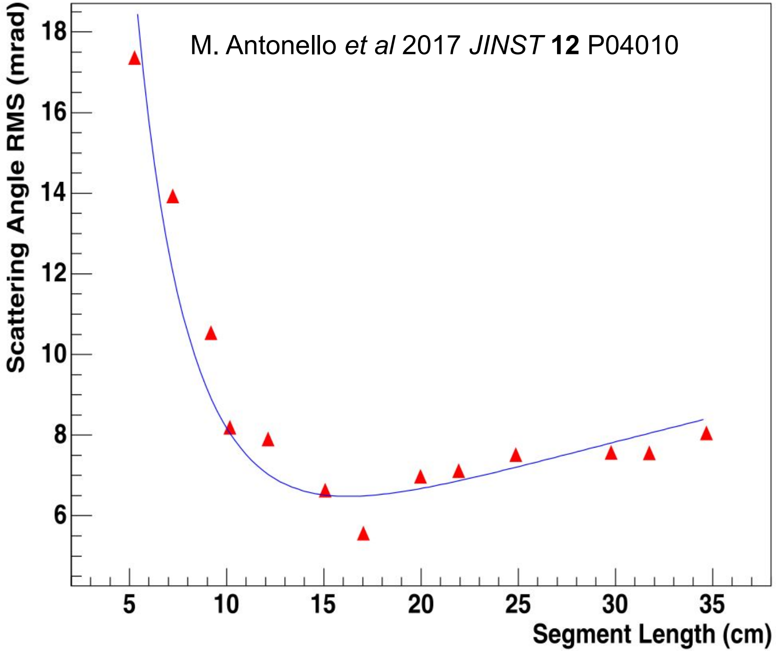
- Muons: z = 1
- LAr: Xo = 14 cm

MCS method - How to retrieve momentum

- **Chi2:**
  - Evaluates θ_0 (the rms of scattered angles) for different 'x' (segments)
  - Find momentum by minimizing measurement vs expected value

$$\theta_0 = \frac{13.6\ \mathrm{MeV}}{\beta c p} z \sqrt{\frac{x}{X_0}} \left[1 + 0.038 \ln\left(\frac{x z^2}{X_0 \beta^2}\right) \right]$$

$$\begin{aligned} \theta_{\mathrm{meas}}^{\mathrm{rms}} &= \sqrt{(\theta_0^{\mathrm{rms}})^2 + (\theta_{\mathrm{noise}}^{\mathrm{rms}})^2} \\ &= \left\{ \left(\frac{13.6\ \mathrm{MeV}}{\beta c p} z \sqrt{\frac{l}{X_0}} \left[1 + 0.038 \ln\left(\frac{l}{X_0}\right) \right] \right)^2 \right. \\ &\quad \left. + \left(C \cdot l^{-3/2} \right)^2 \right\}^{1/2}, \qquad (3) \end{aligned}$$



NOTE: For OPERA and uboone, the noise was fixed ~2 mrad

Implementation: Final step – Chi2

- **Minimizing**:

$$\chi^2 = \sum_i \left(\frac{\theta^{\rm rms}_{{\rm meas}\, i} - \theta^{\rm rms}_{{\rm theo}\, i}}{\Delta\theta^{\rm rms}_{{\rm meas}\, i}} \right)^2$$

- **In current LArSoft:**

$$\theta^{\rm rms}_{\rm meas} = \sqrt{(\theta^{\rm rms}_0)^2 + (\theta^{\rm rms}_{\rm noise})^2}$$

$$= \left\{ \left(\frac{13.6\ {\rm MeV}}{p} z \sqrt{\frac{l}{X_0}} \left[1 + 0.038 \ln\left(\frac{l}{X_0}\right)\right]\right)^2 + \theta^{\rm rms}_{\rm noise}{}^2 \right\}^{1/2}$$

NOTE: Current version assumes Beta = 1 for all energies

MCS method - How to retrieve momentum

- **LLHD:**
  - "The normal probability distribution for a scattering angle in either the x_0 or y_0 direction, $\Delta\theta$, with an expected Gaussian uncertainty σ_0 and mean of zero is given by":

$$f_X(\Delta\theta) = (2\pi\sigma_o^2)^{-\frac{1}{2}} \exp\left[-\frac{1}{2}\left(\frac{\Delta\theta}{\sigma_o}\right)^2\right]$$

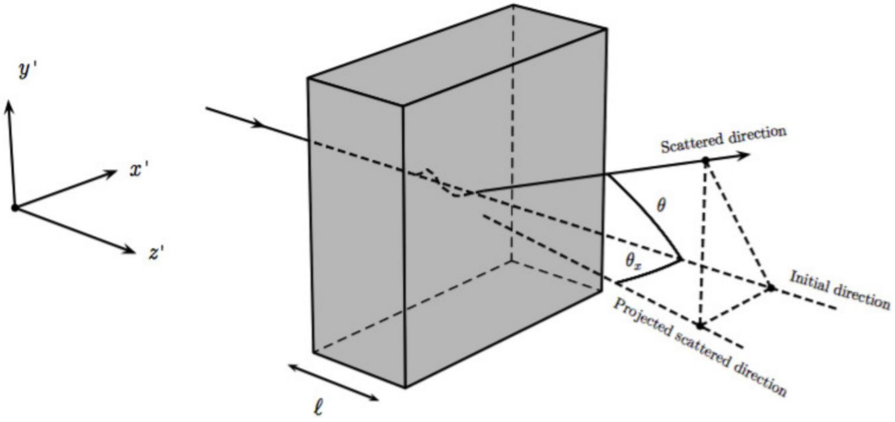
  - "Since energy is lost between segments along the track, σo increases for each angular measurement along the track. We therefore replace σ_o with $\sigma_{o,j}$, where j is an index representative of the segment. To obtain the likelihood, we take the product of $f_X(\Delta\theta_j)$ over all n of the $\Delta\theta_j$ segment-to-segment scatters along the track." The function to be minimized became:

$$L(\sigma_{o,1},\ldots,\sigma_{o,n};\Delta\theta_1,\ldots,\Delta\theta_n) = (2\pi)^{-\frac{n}{2}} \times \prod_{j=1}^{n}(\sigma_{o,j})^{-1} \times \exp\left[-\frac{1}{2}\sum_{j=1}^{n}\left(\frac{\Delta\theta_j}{\sigma_{o,j}}\right)^2\right]$$

$$-l(\sigma_{o,1},\ldots,\sigma_{o,n};\Delta\theta_1,\ldots,\Delta\theta_n) = -\ln(L) = \frac{n}{2}\ln(2\pi) + \sum_{j=1}^{n}\ln(\sigma_{o,j}) + \frac{1}{2}\sum_{j=1}^{n}\left(\frac{\Delta\theta_j}{\sigma_{o,j}}\right)^2$$

Implementation: Second step – LLHD

- After getting the particle direction after each segment (MCS method - Implementation (Chi2 and LLHD))
- Compute $\Delta\theta_{xz}$ and $\Delta\theta_{yz}$ for each segment.



  - Link this in the formula to minimize:

$$-l(\sigma_{o,1},\ldots,\sigma_{o,n};\Delta\theta_1,\ldots,\Delta\theta_n) = -\ln(L) = \frac{n}{2}\ln(2\pi) + \sum_{j=1}^{n}\ln(\sigma_{o,j}) + \frac{1}{2}\sum_{j=1}^{n}\left(\frac{\Delta\theta_j}{\sigma_{o,j}}\right)^2$$

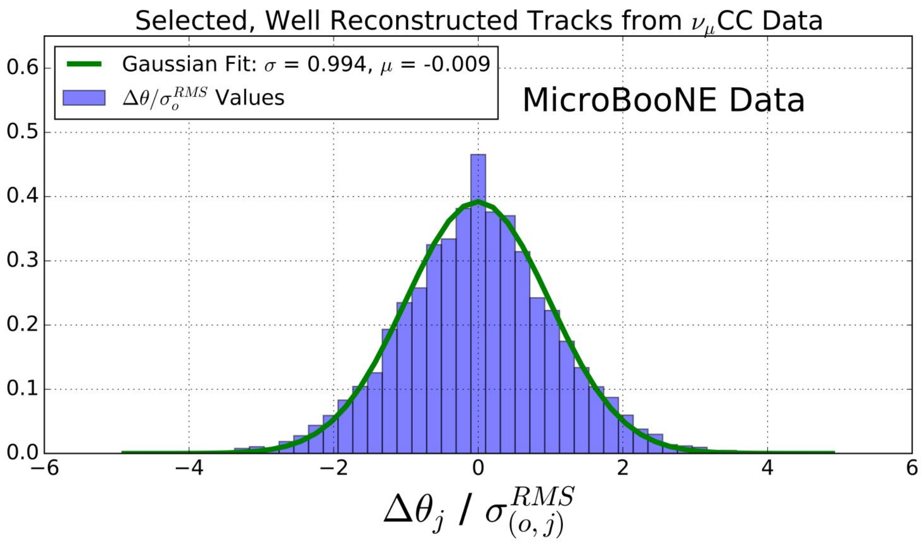
  - For each step, computes $\sigma_{o,j}$ expected (next slide)

Implementation: comments on LLHD

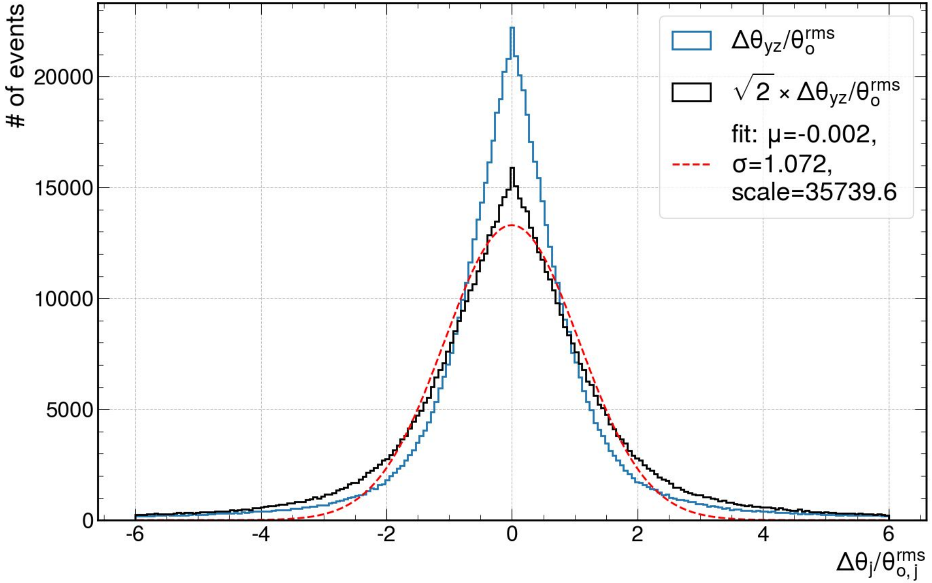
- LLHD was done using **segments of 10 cm**.
- In one of my many attempts to debug, I noticed the that Δθ was usually lower then θ_{rms}.
- I tried to use the space angle, but I forgot √2
  - **This confirmed second bullet and lead to an improved result**

$$\theta_0 = \theta_{\text{plane}}^{\text{rms}} = \frac{1}{\sqrt{2}}\,\theta_{\text{space}}^{\text{rms}} \,,$$

- **I was expecting this from uboone paper:**



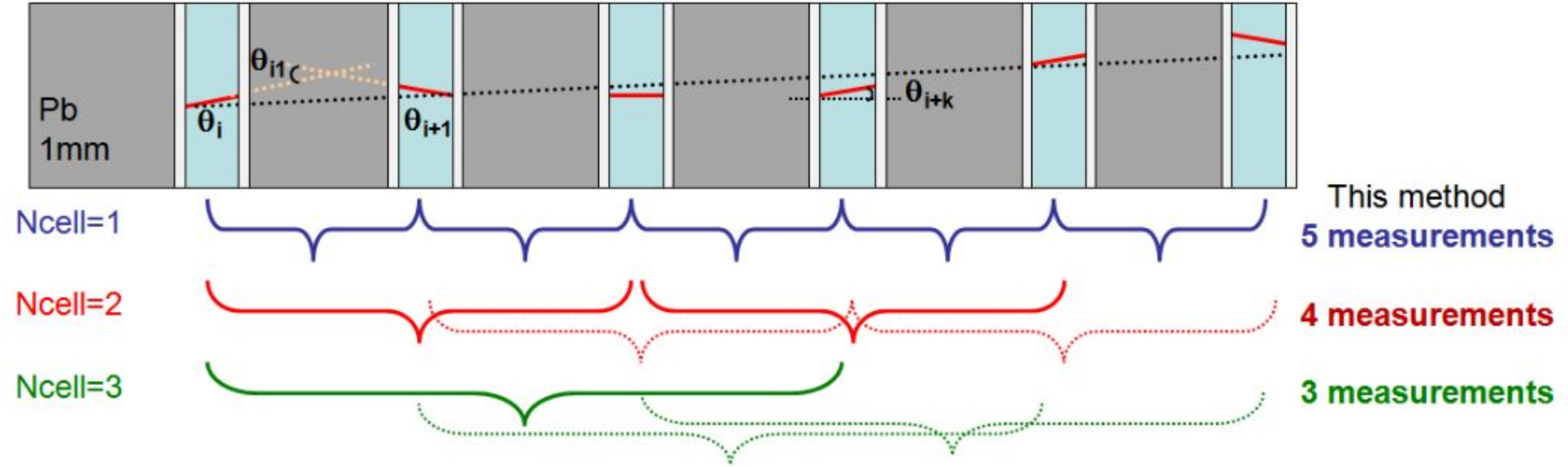
- **Got this instead** (yz-plane)



MCS method - Implementation (Chi2)

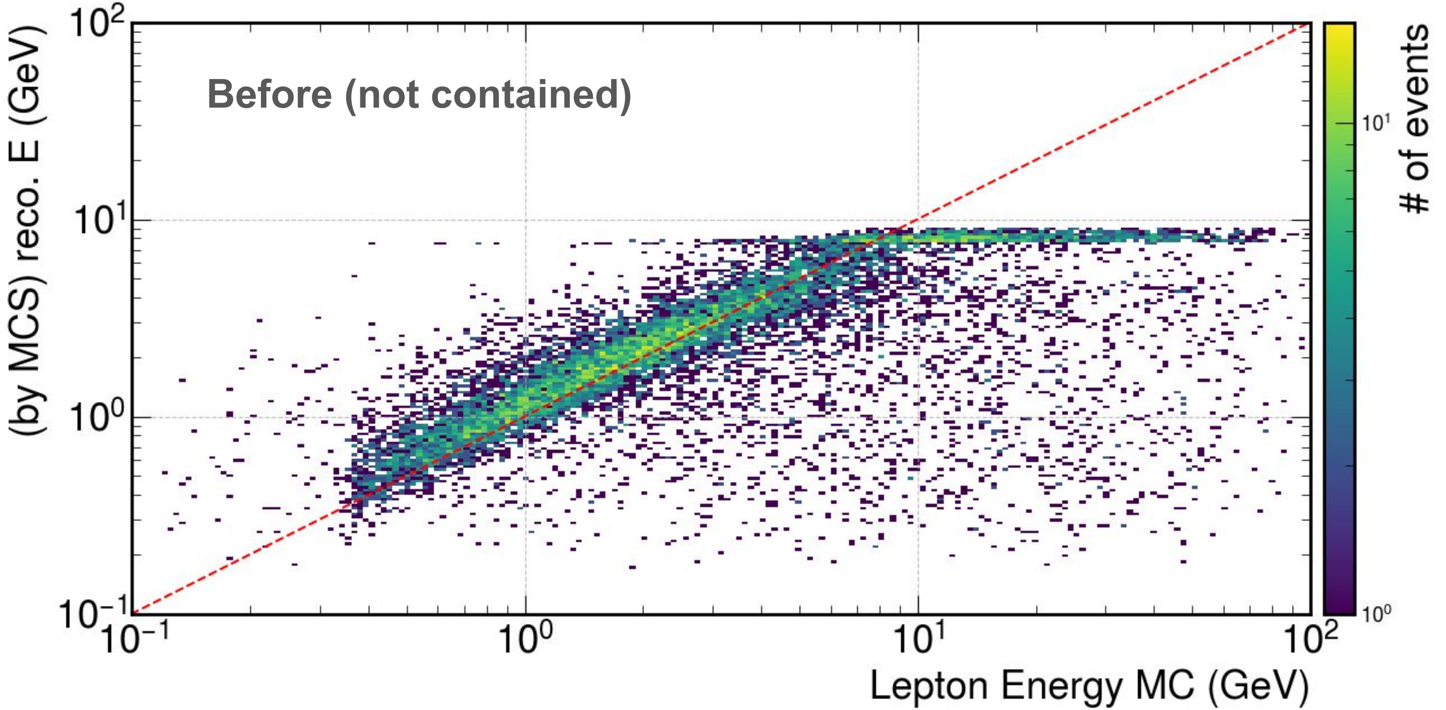
- **Second step (Chi2)**
  - Evaluate the RMS for different segments lengths:

N Agafonova *et al* 2012 *New J. Phys.* **14** 013026



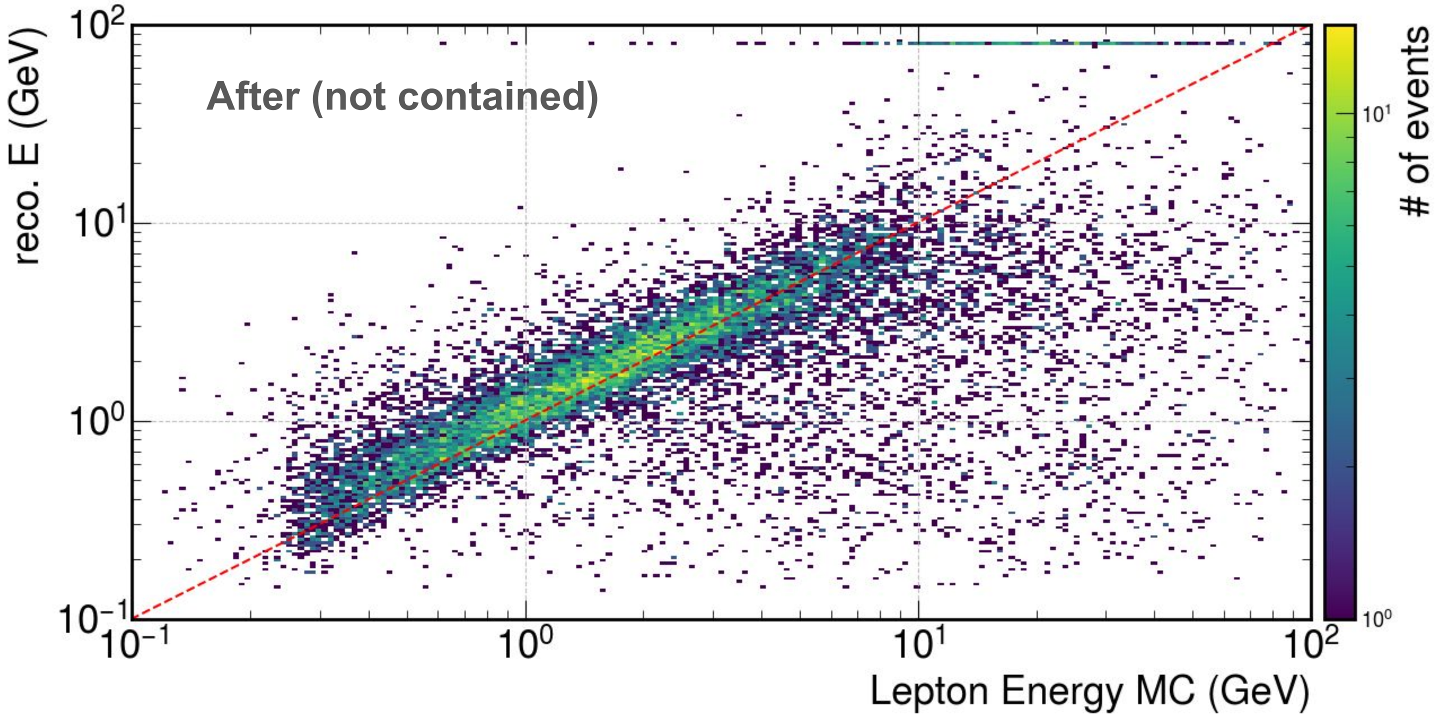
MCS method - Changes in Chi2 method

- **Trying to improve :**
  - **Fixing $\theta_{\text{noise}}^{\text{rms}}$ to 2 mrad (more study necessary)**
  - **Removing penalty**
  - **Changed upper limit from 7.5 to 80 GeV**
  - **Set minimum length to 30 cm**
  - **Increase to 18 segments, starting at 5 cm and steps of 2.5 cm**



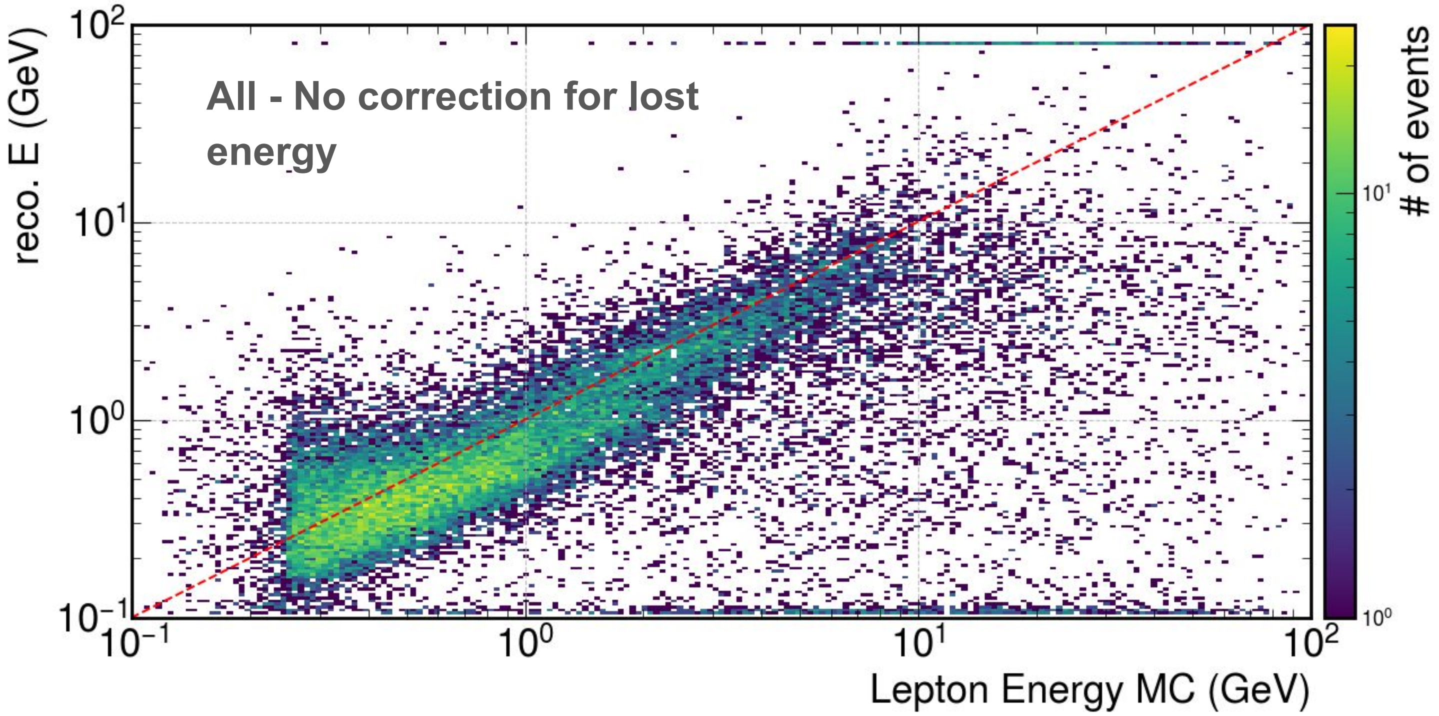
MCS method - Changes in Chi2 method

- **Trying to improve :**
  - **Fixing $\theta_{\text{noise}}^{\text{rms}}$ to 2 mrad (more study necessary)**
  - **Removing penalty**
  - **Changed upper limit from 7.5 to 80 GeV**
  - **Set minimum length to 30 cm**
  - **Increase to 18 segments, starting at 5 cm and steps of 2.5 cm**



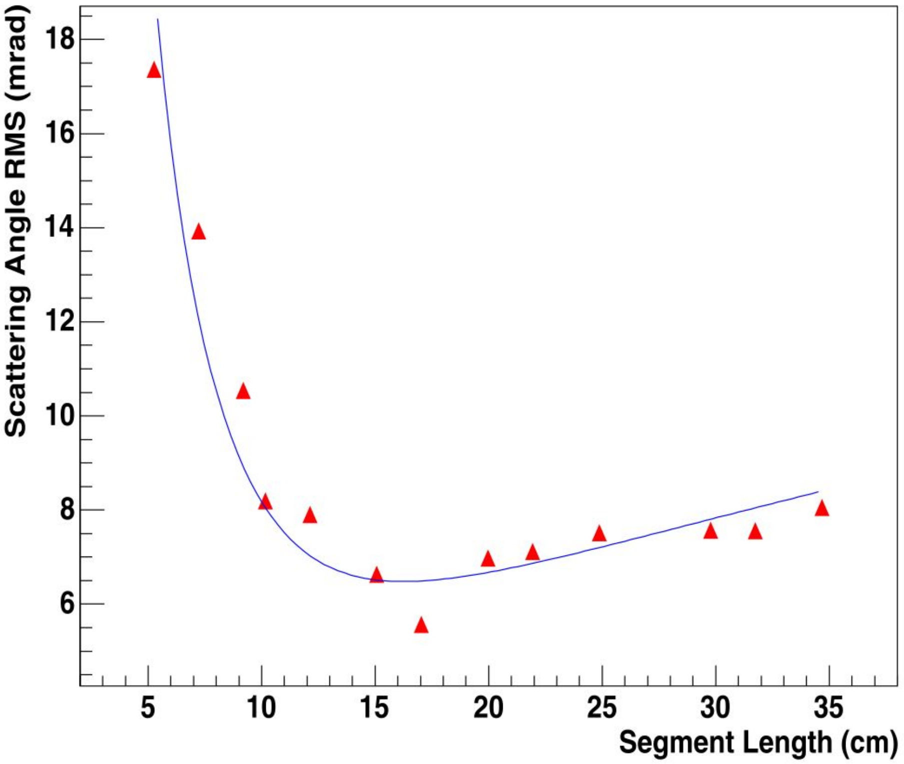
MCS method - Changes in Chi2 method

- **Trying to improve :**
  - **Fixing $\theta_{\text{noise}}^{\text{rms}}$ to 2 mrad (more study necessary)**
  - **Removing penalty**
  - **Changed upper limit from 7.5 to 80 GeV**
  - **Set minimum length to 30 cm**
  - **Increase to 18 segments, starting at 5 cm and steps of 2.5 cm**

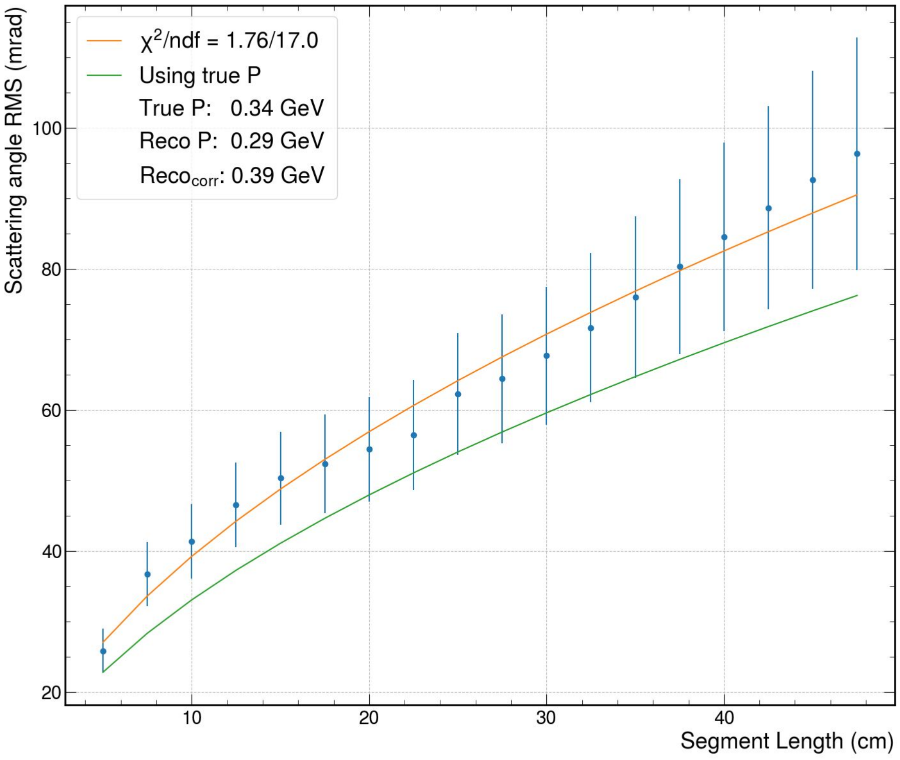


MCS method - Quick note

- **I was expecting a similar plot to ICARUS:**

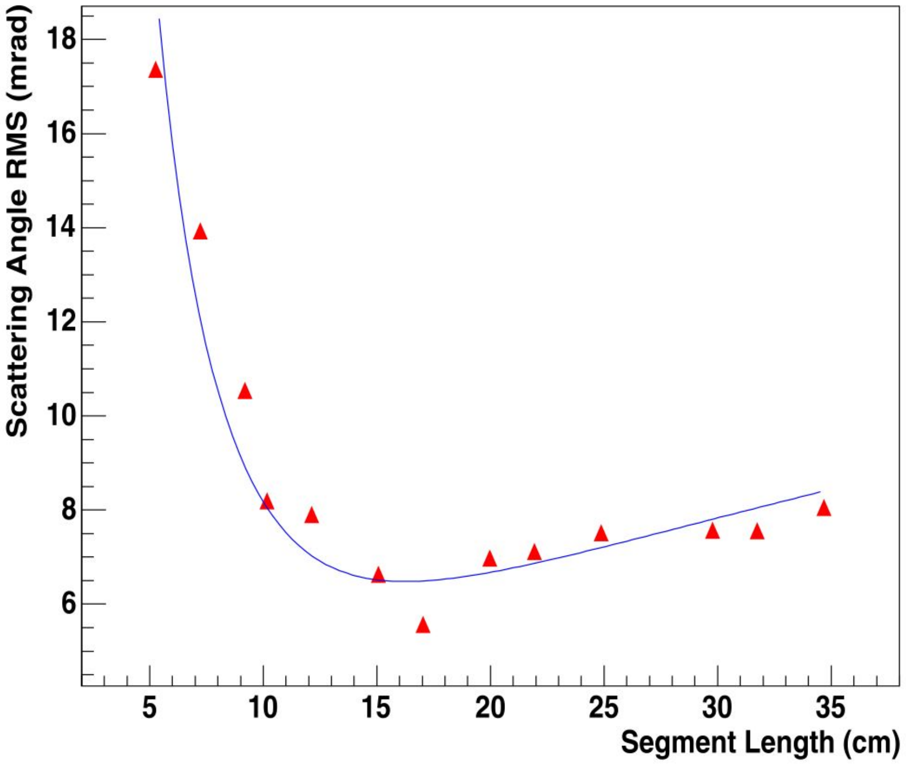


- **The result was this instead:**

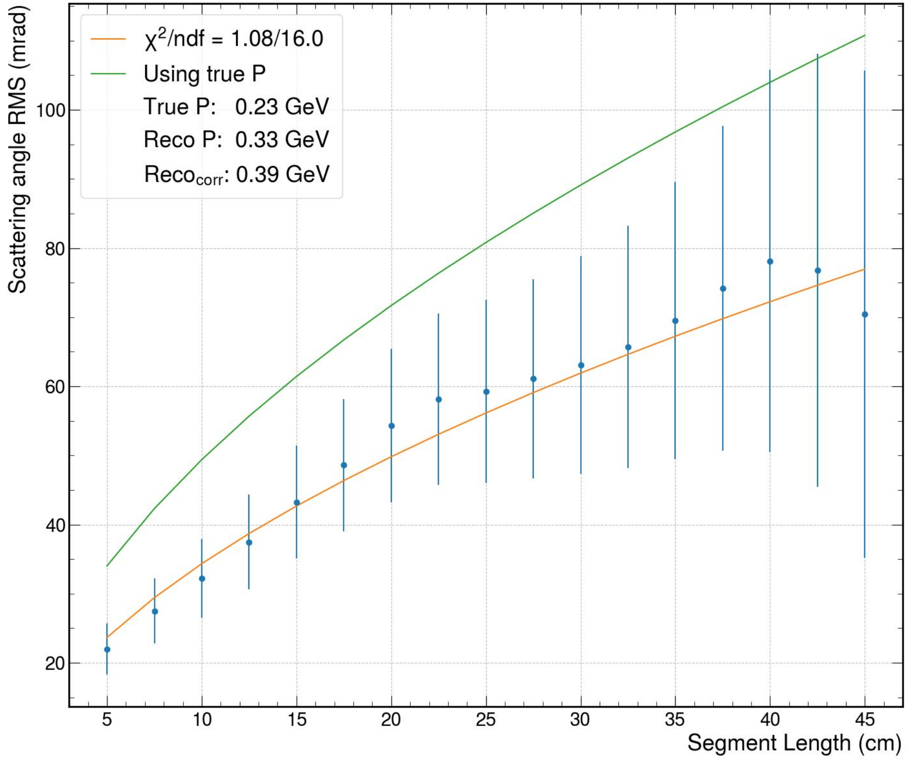


MCS method - Quick note

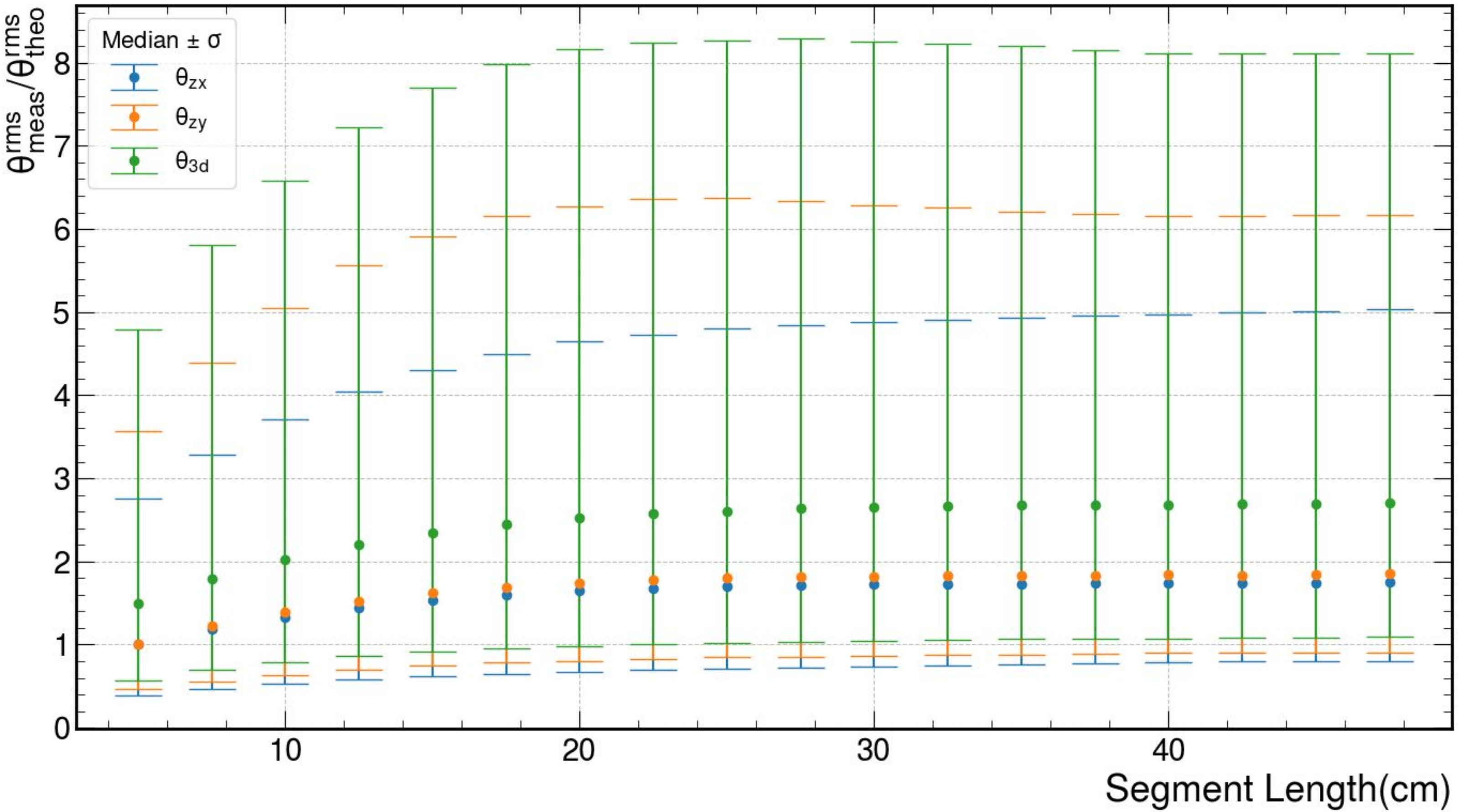
- **I was expecting a similar plot to ICARUS:**



- **The result was this instead:**

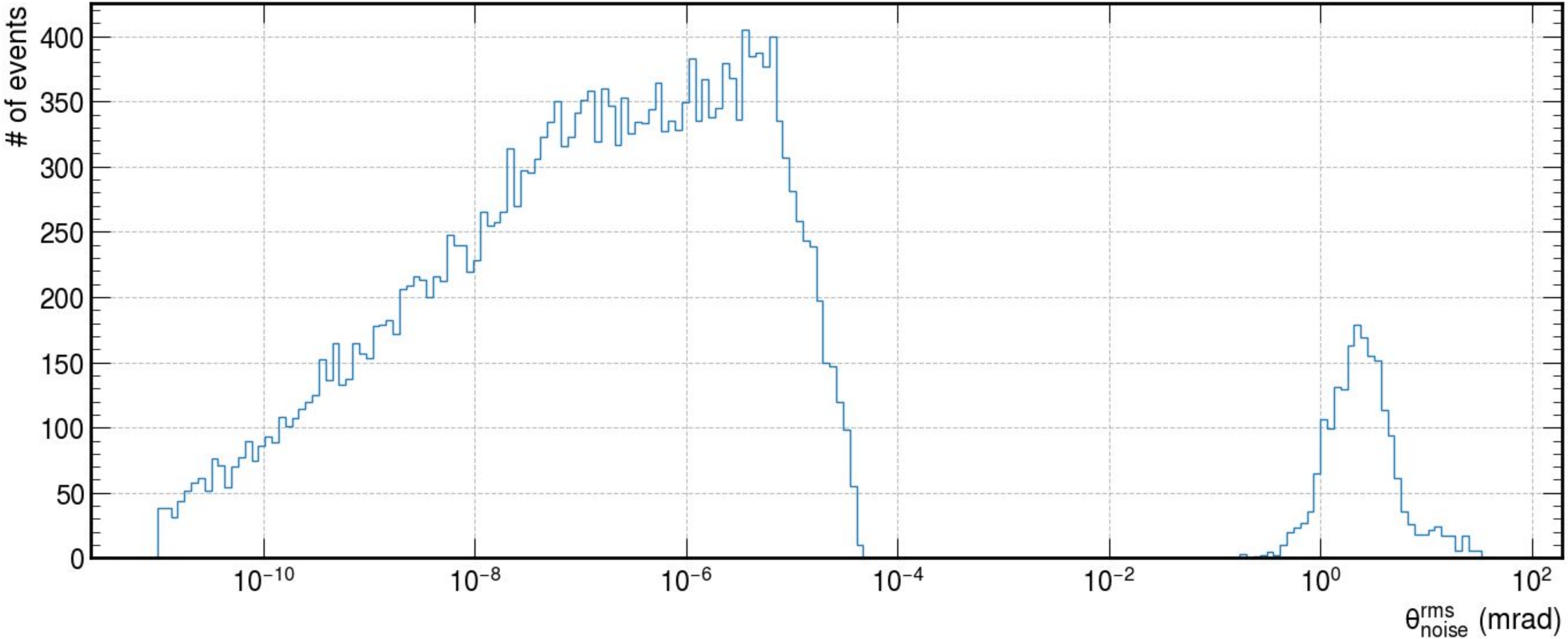


MCS method - Quick note chi2



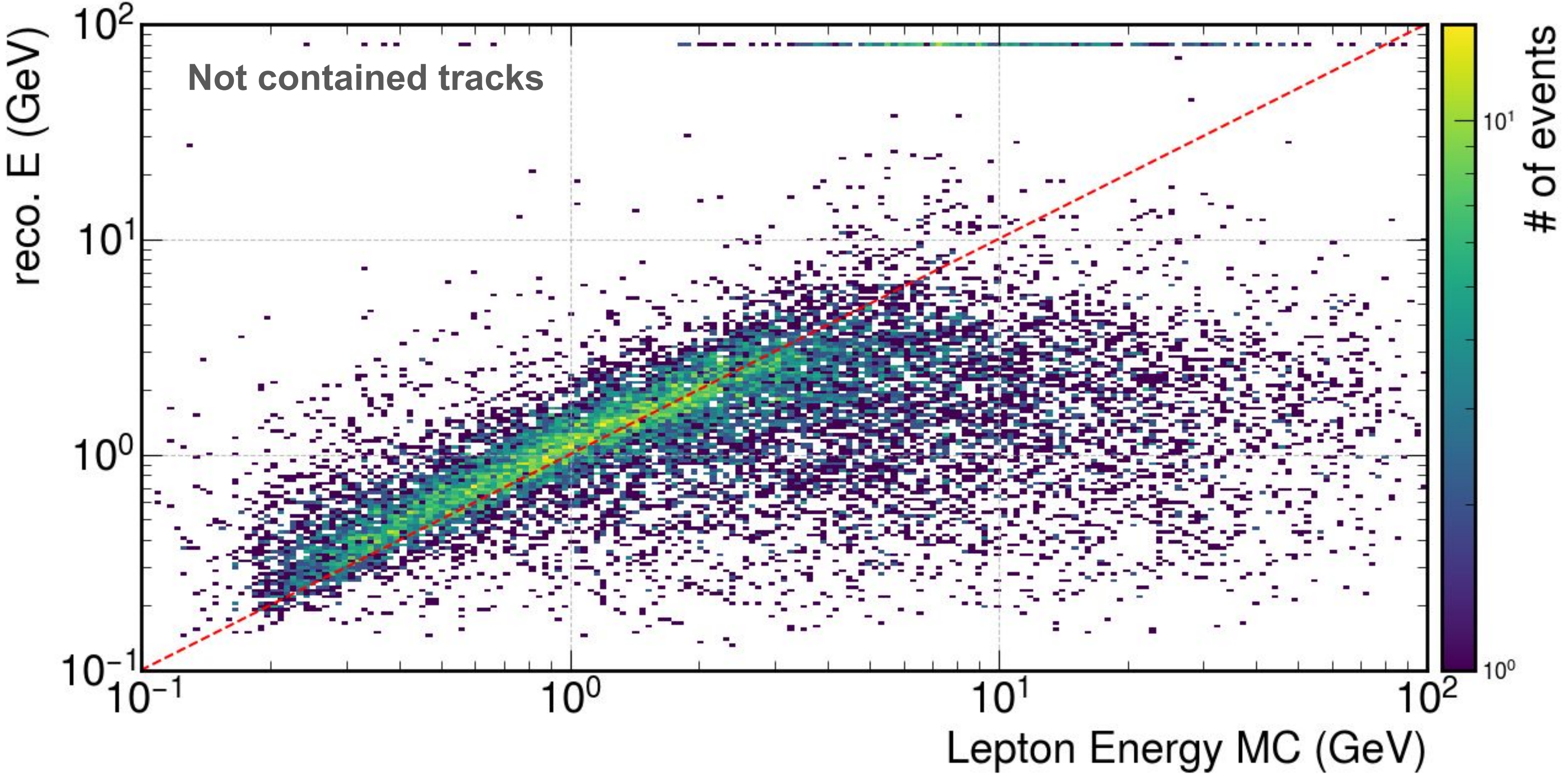
Backup

Example bad fitting of noise rms



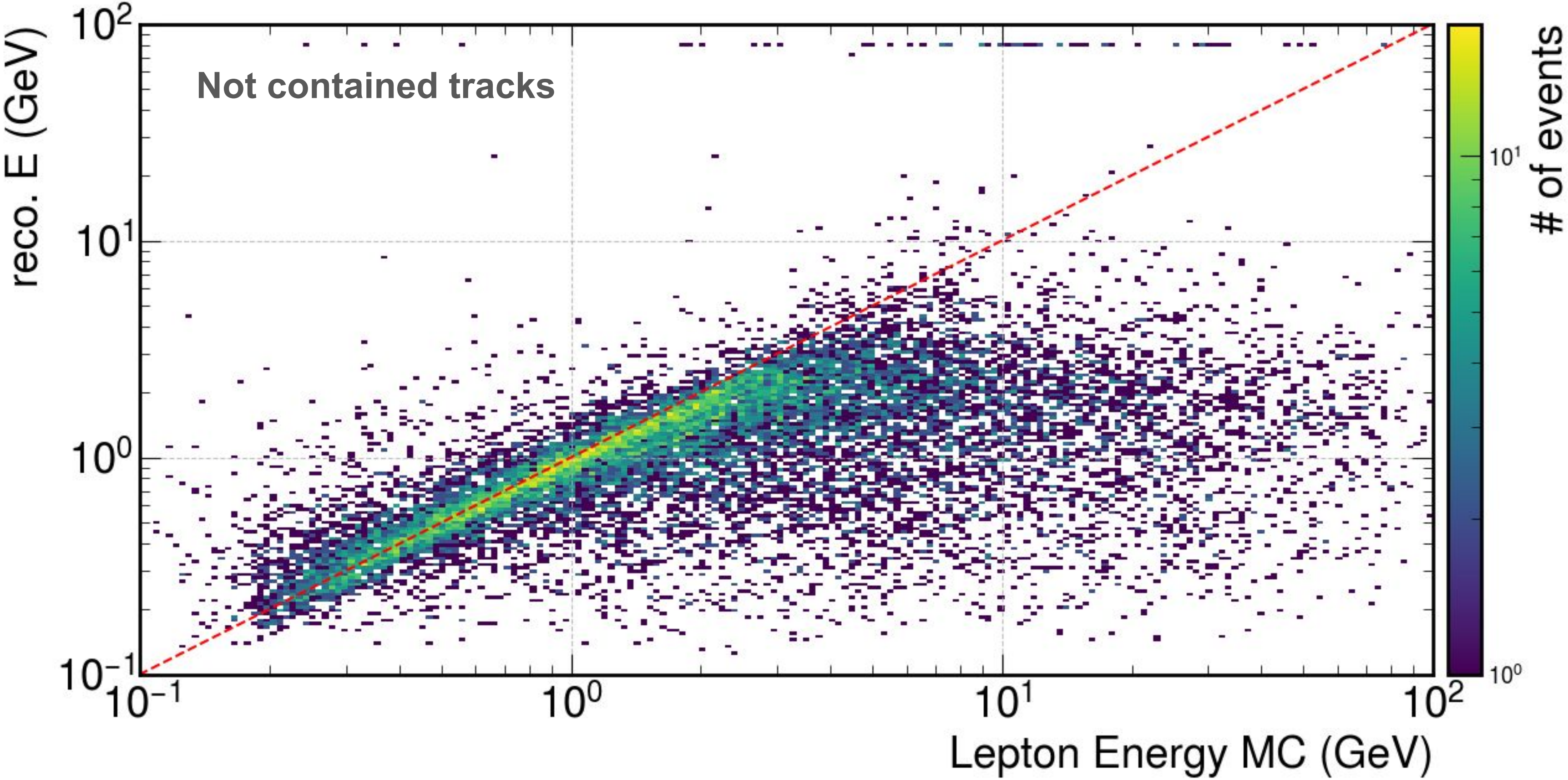
MCS method - Implementation (LLHD)

- **Result obtained with segments of 10 cm**



MCS method - Comments on LLHD

- **Using space angle (without √2) - $\Delta\theta_{space} = \sqrt{(\Delta\theta_{xz}^2 + \Delta\theta_{yz}^2)}$**



Energy reconstruction - Lepton E. reconstruction (OLD)