

# ***ECAL Energy Calibration Updates***

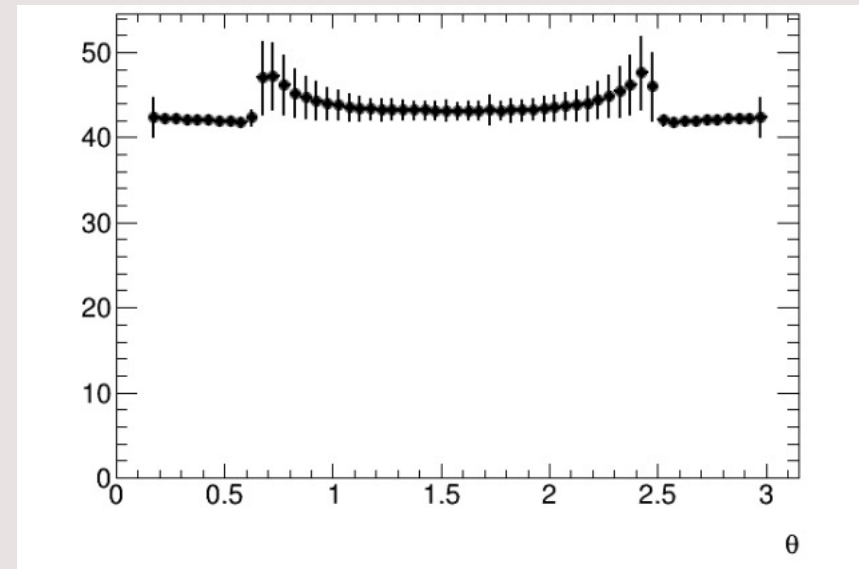
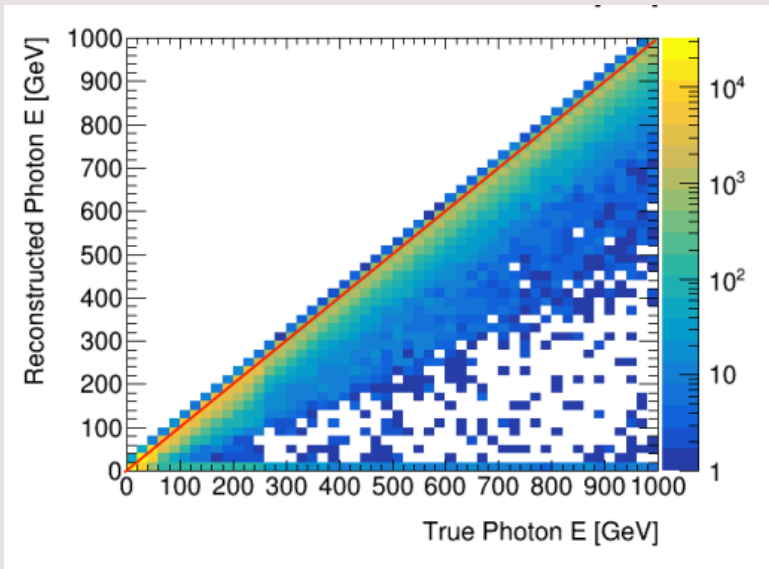
8 May 2024

5/7/24



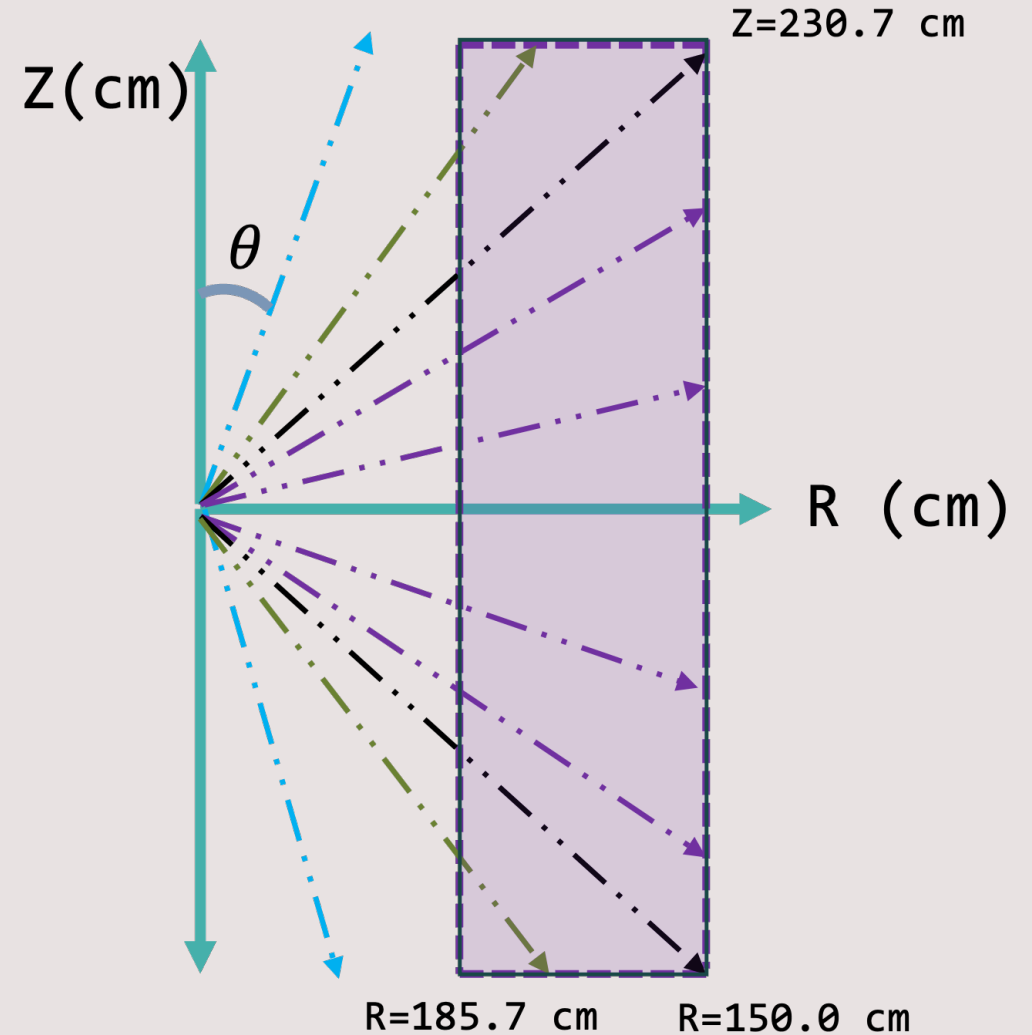
# Review of the Problem

- Experiencing poor photon resolution and globally underestimated  $E_{\text{reco}}$
- Tova confirmed non-uniform **theta-dependence** of  $E_{\text{truth}}/E_{\text{reco}}$  (see her super helpful plots below)
- Conclusion: we need a theta-dependent calibration at the clustering/reco stage



# A First-Principles Approach

- Angular dependence originates from the solenoid geometry
- With the assumption that photons begin showering in the magnet, use trig to find a model for angular-dependent energy loss
- First: determine how many **radiation lengths** ( $X_0$ ) of solenoid material photons see
- Three regions:
  - + Doesn't interact with solenoid (bounded by inner R-limit)
  - + Bounded by z-limit
  - + Bounded by outer R-limit



# Piecewise Energy Loss

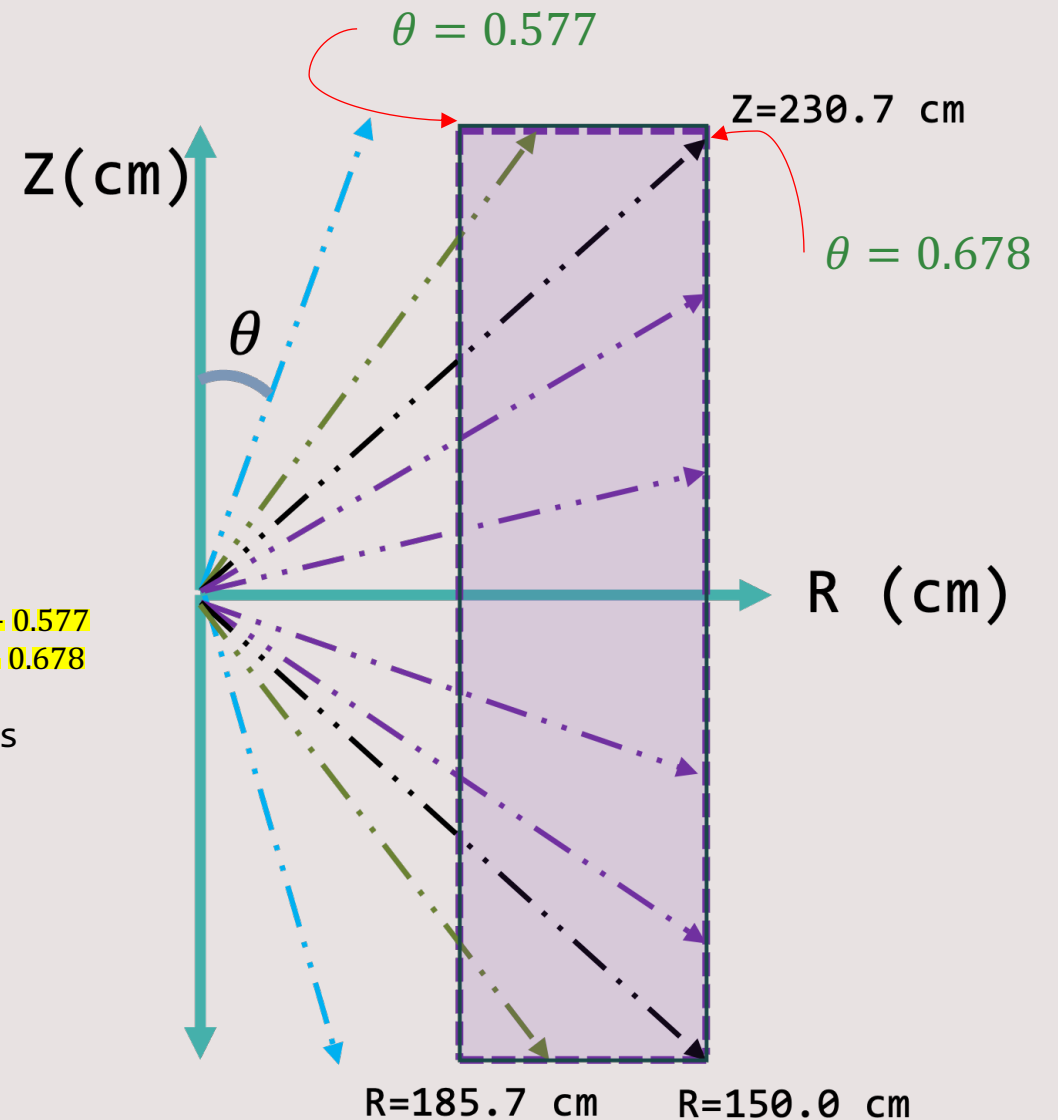
- Assumptions:
  - + Photons fired from the origin
  - +  $X_0$  in Al  $\sim 8.897$  cm
  - + Pair production and Brem dominate energy loss
  - +  $E(N) = E_0 2^{-N}$  (where N is # of rad lengths)

$N(\theta) =$

$$\begin{cases} 0 & \theta < 0.577, \theta > \pi - 0.577 \\ 25.93|\sec \theta| - 16.86 \csc \theta & 0.577 < \theta < 0.678, \pi - 0.678 < \theta < \pi - 0.577 \\ 4.01 \csc \theta & 0.678 < \theta < \pi - 0.678 \end{cases}$$

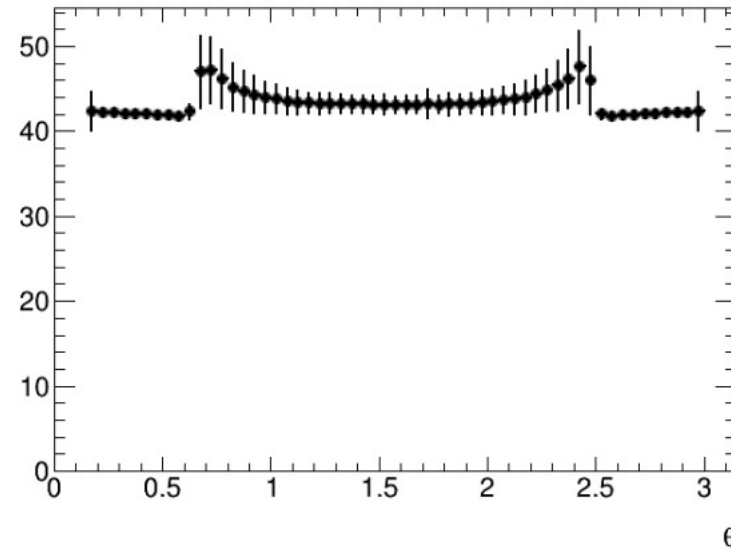
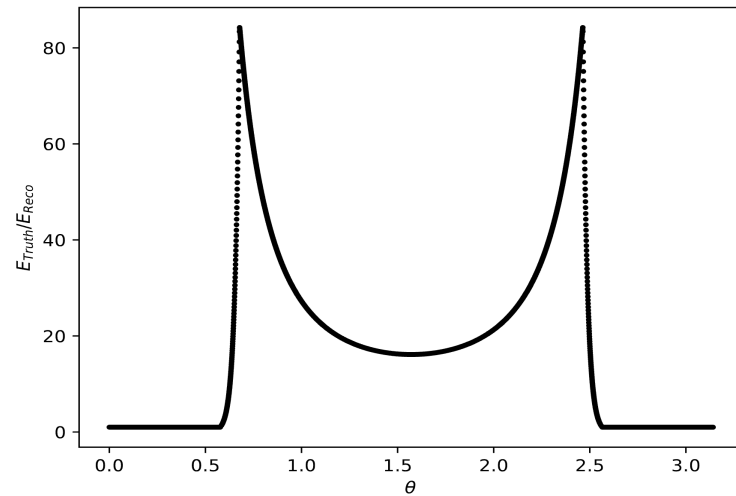
- Then we expect to model our ratio of  $E_{\text{true}}$  to  $E_{\text{reco}}$  as

$$\frac{E_{\text{truth}}}{E_{\text{reco}}} = 2^{N(\theta)}$$



## ***Expected ratio***

- Plotting this function for values of theta between 0 and pi returns a familiar shape
- Obviously not scaled correctly
- However, it looks like this may be the correct functional form



## ***Fitting to our data***

- For further convincing, dividing by 10 and shifting to combat the underline  $\sim 40x$  offset factor gives us back almost the same distribution (see below)
- Plan: reproduce the  $E_t/E_r$  profile for the latest photon sample (which should take care of the constant offset) and then fit to our custom function
- A simple calibration model: multiply reconstructed energy by our ratio function

