

FCAL Update

- Energy resolution in data
- FCAL geometry
- Inefficiencies due to poorly-determined gain constants
- Time slewing and timing resolution



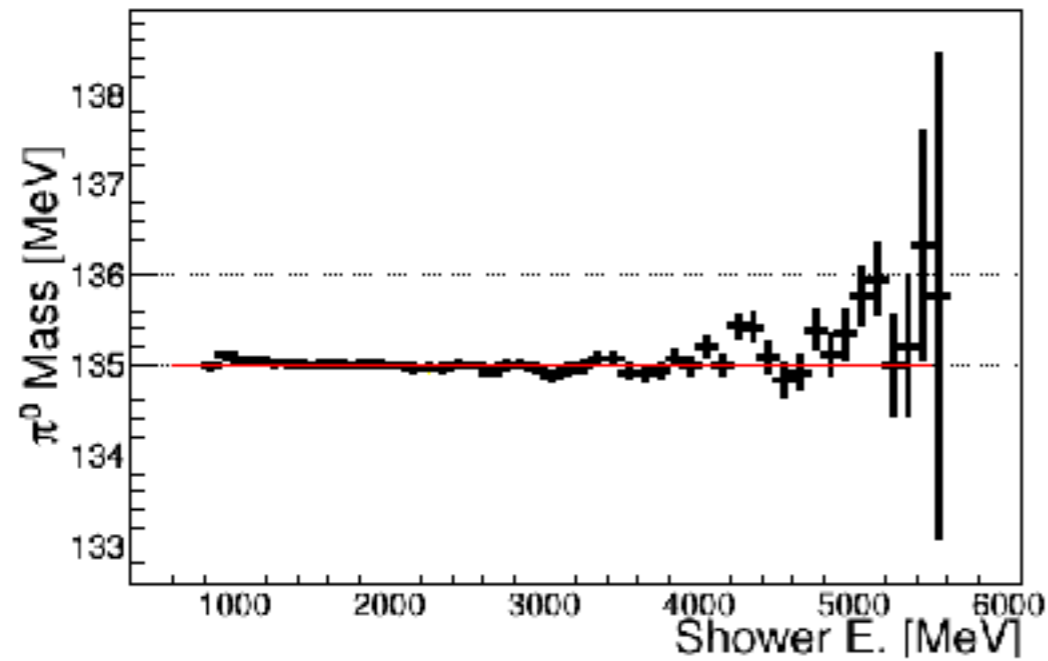
FCAL Energy Resolution

- Gain balancing done by Adesh (plots on the next page)
 - corrected block size
 - using only photons with $1.0 < E < 1.5$ GeV
 - “floor term” still appears too high
- Possible issues:
 - position resolution begins to contribute at high photon energies (η should be better)
 - poor background assumptions in fitting π^0 peak
 - many of Adesh’s fits used a linear background over a very restricted range — OK for getting peak position but not width
 - other: the resolution just isn’t as good as we expected... why?
- May need to resort to different (cleaner) event sample to validate MC resolution

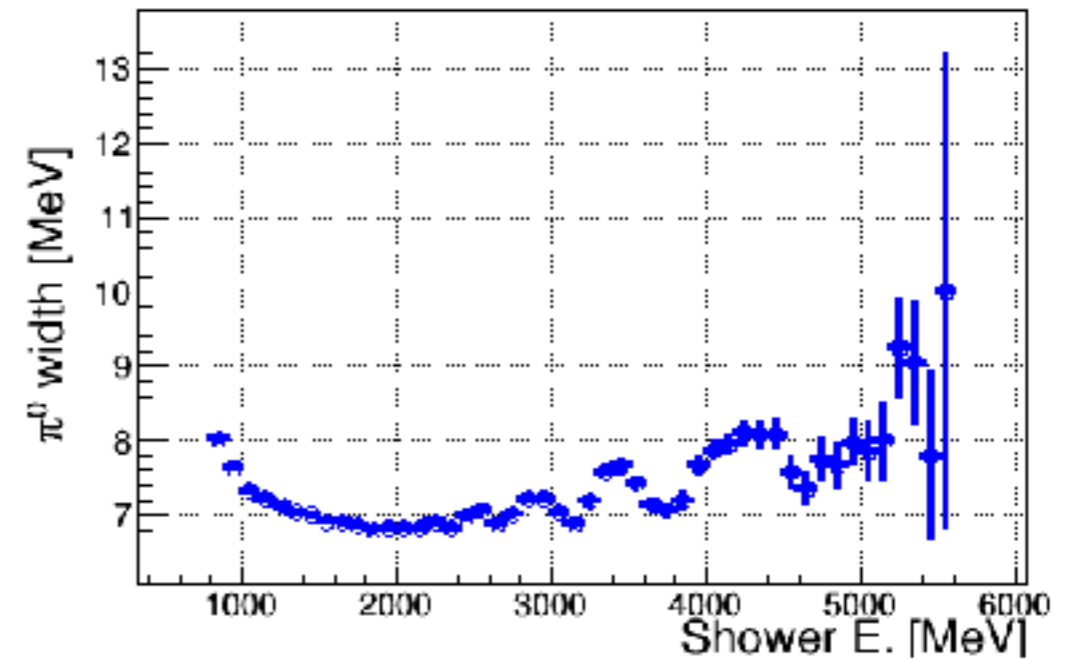


FCAL Energy Resolution

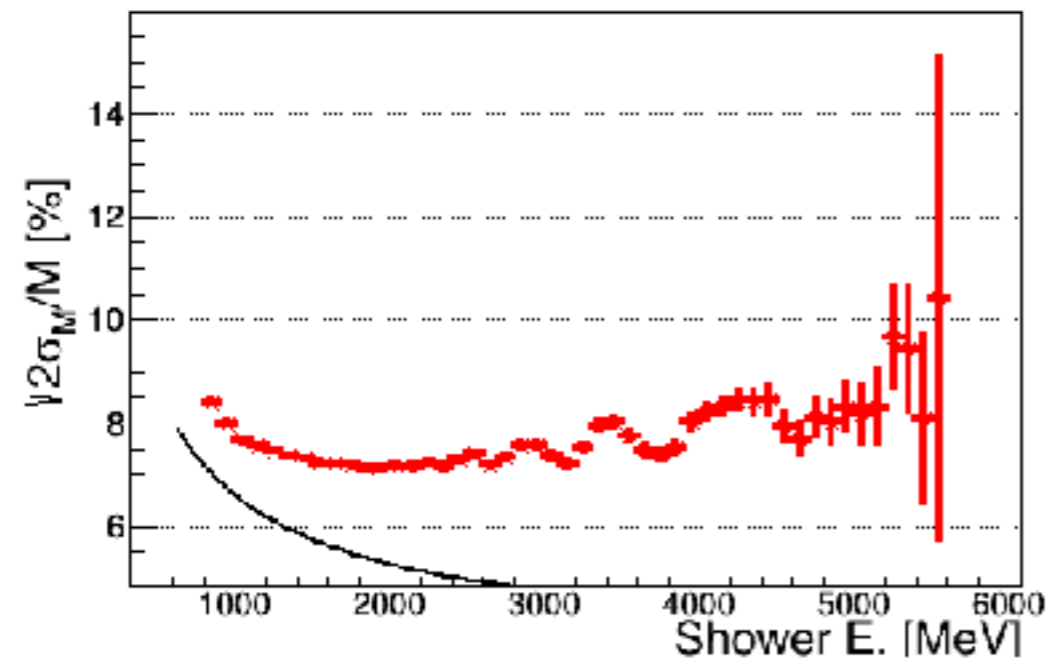
Mass Variation



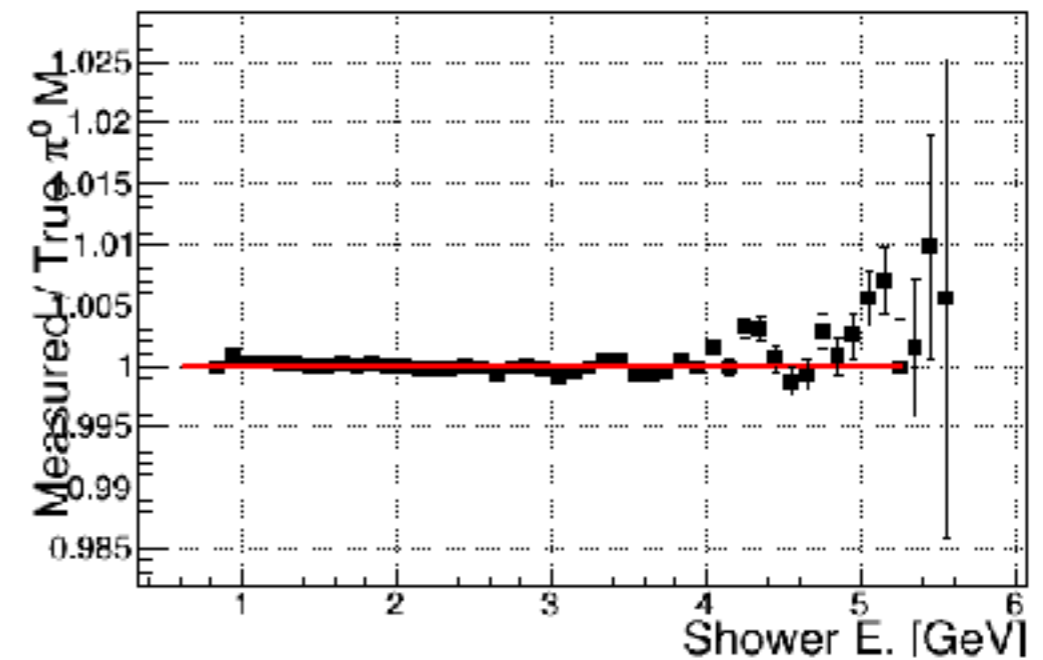
Width Variation



Resolution

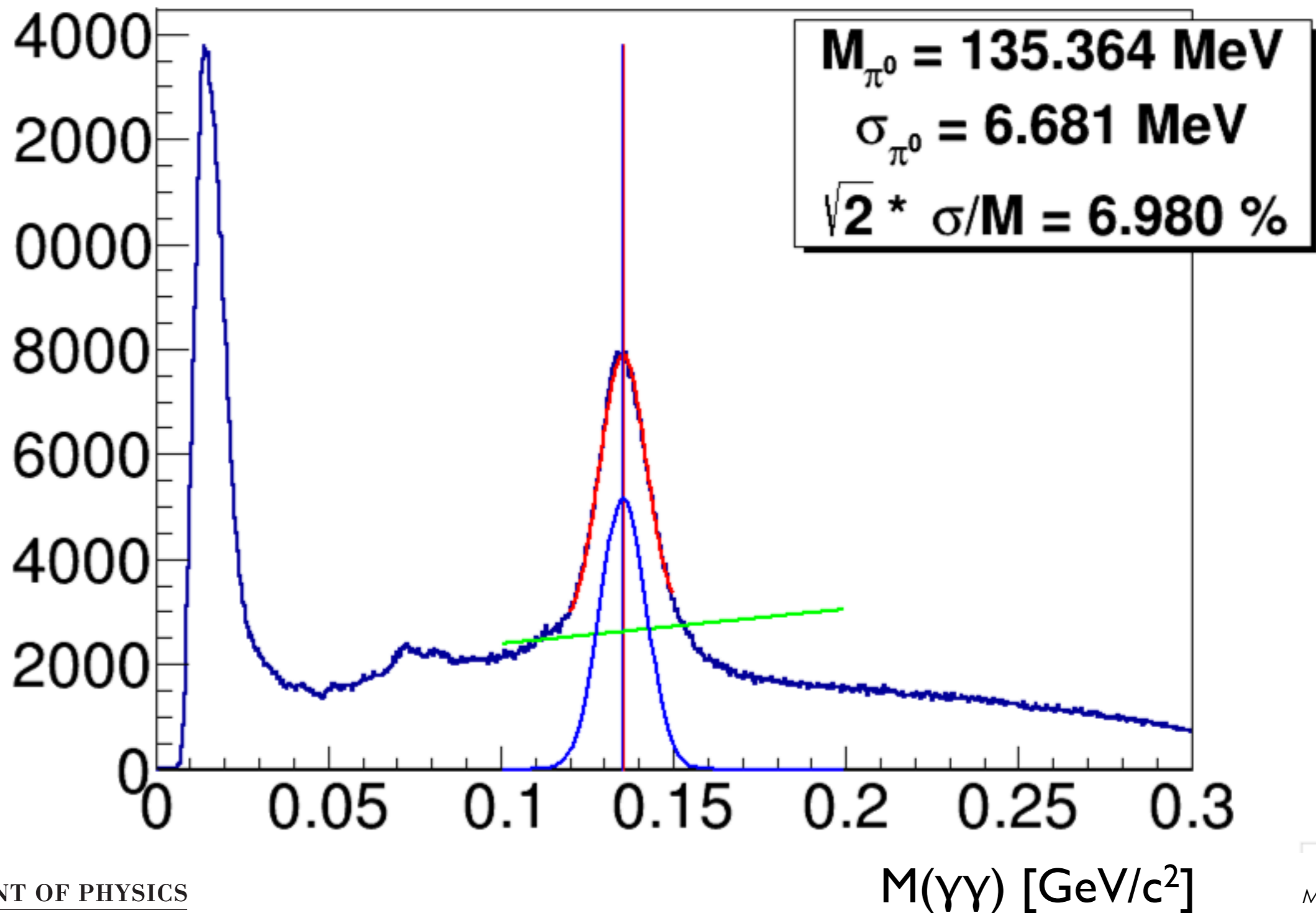


Non-linearity function: $y(x) = p_0 + \exp(-p_1 \cdot x + p_2)$



FCAL Energy Resolution

1.40 < Shower E. [GeV] < 1.50



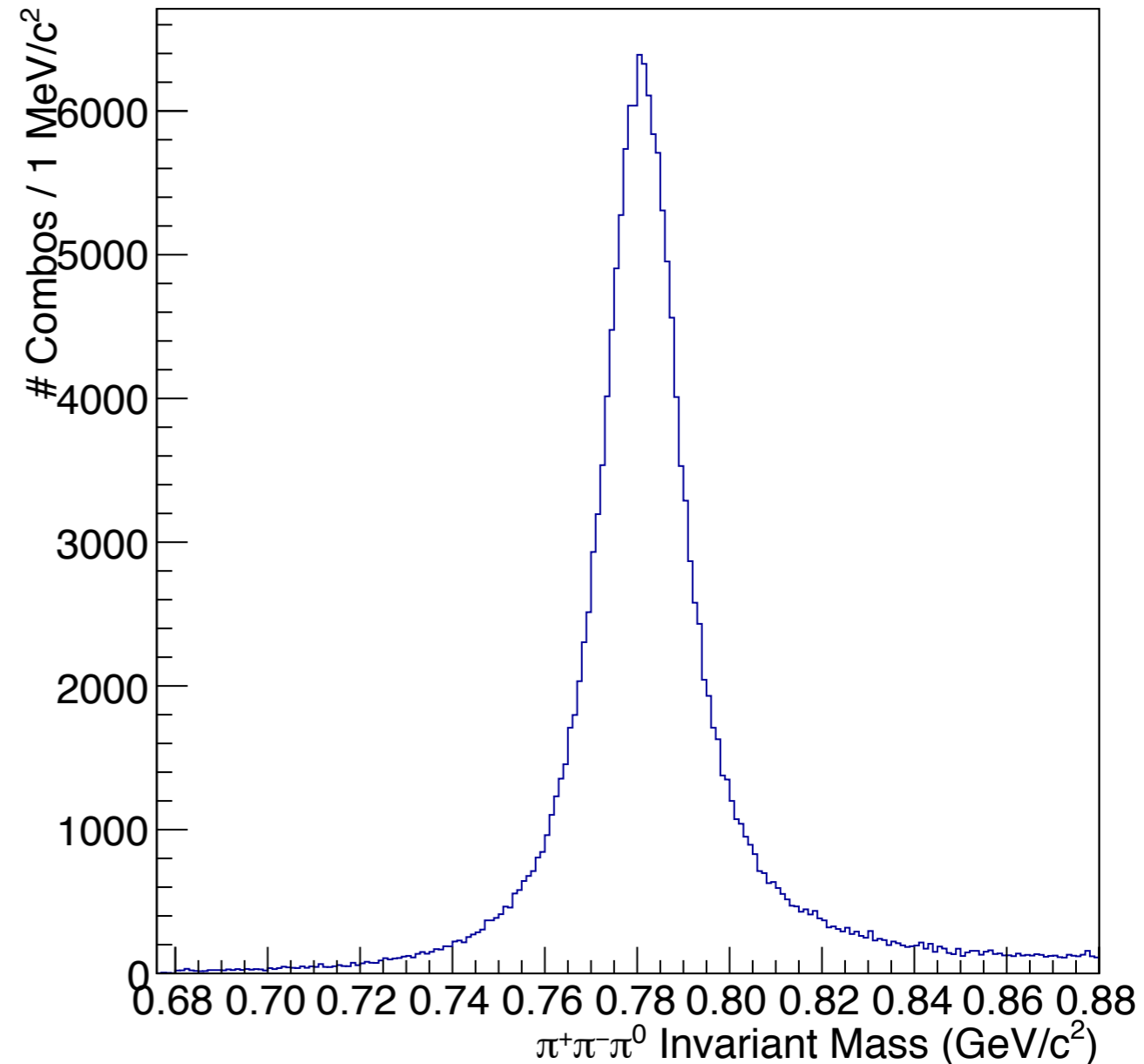
FCAL Geometry Updates

- Key change: the unit cell in the FCAL was assumed to be 4.000 cm; in reality “as built” it is 4.016 cm
 - affects both data reconstruction and MC
 - change in [DFCALGeometry.cc](#) mandates a change of gain constants as well
 - not committed yet
- MC modifications (all committed?):
 - incorporate change to unit cell size
 - add material for upstream plate and straps
 - add material for plastic light tight cover
 - add light guide sensitive volume (studies with data suggest MIPs that hit the light guide have different energy and timing response)



Data MC Comparison

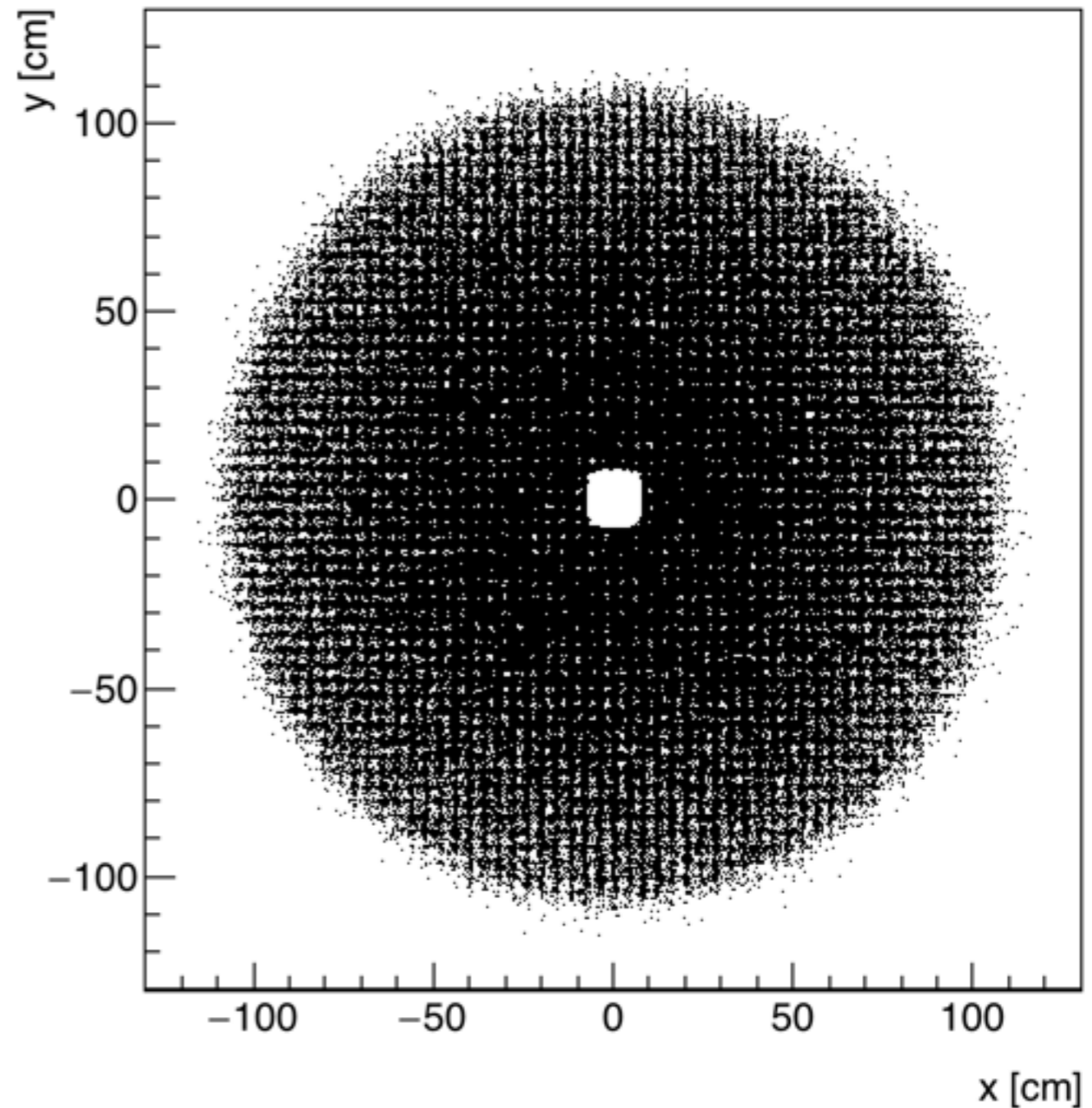
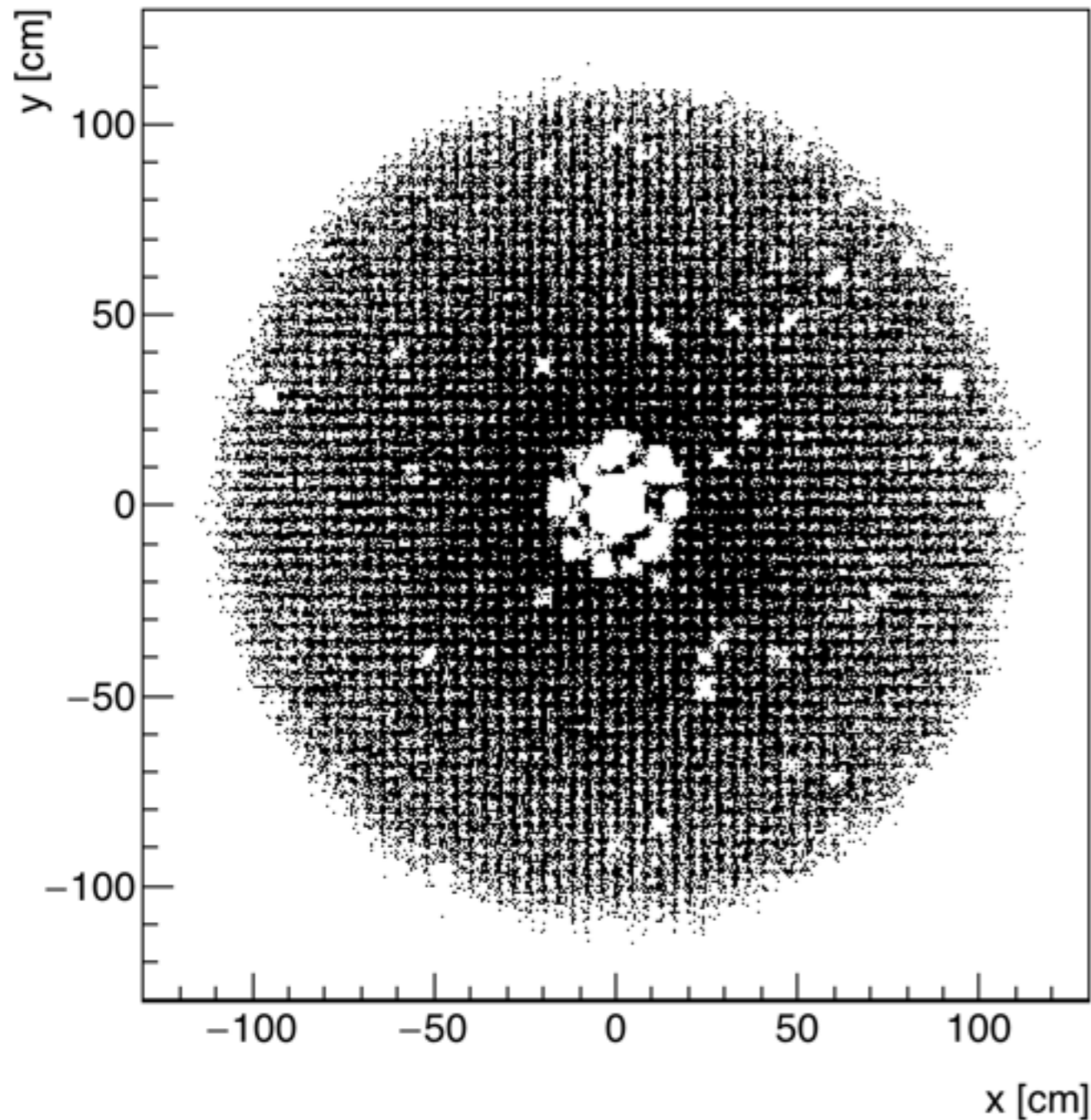
- Reconstruct events of the type:
 - $\gamma p \rightarrow \omega p$ where $\omega \rightarrow \pi^+ \pi^- \pi^0$
 - select with a kinematic fit that includes π^0 mass constraint and 5% cut on the confidence level
- "Tag" showers produced by the π^0 decay as true photons
 - use these to study calorimeter performance
 - avoids MIP/splitoff contamination
- Future: relax π^0 mass constraint and do data/MC comparison of π^0 width



Photon Locations

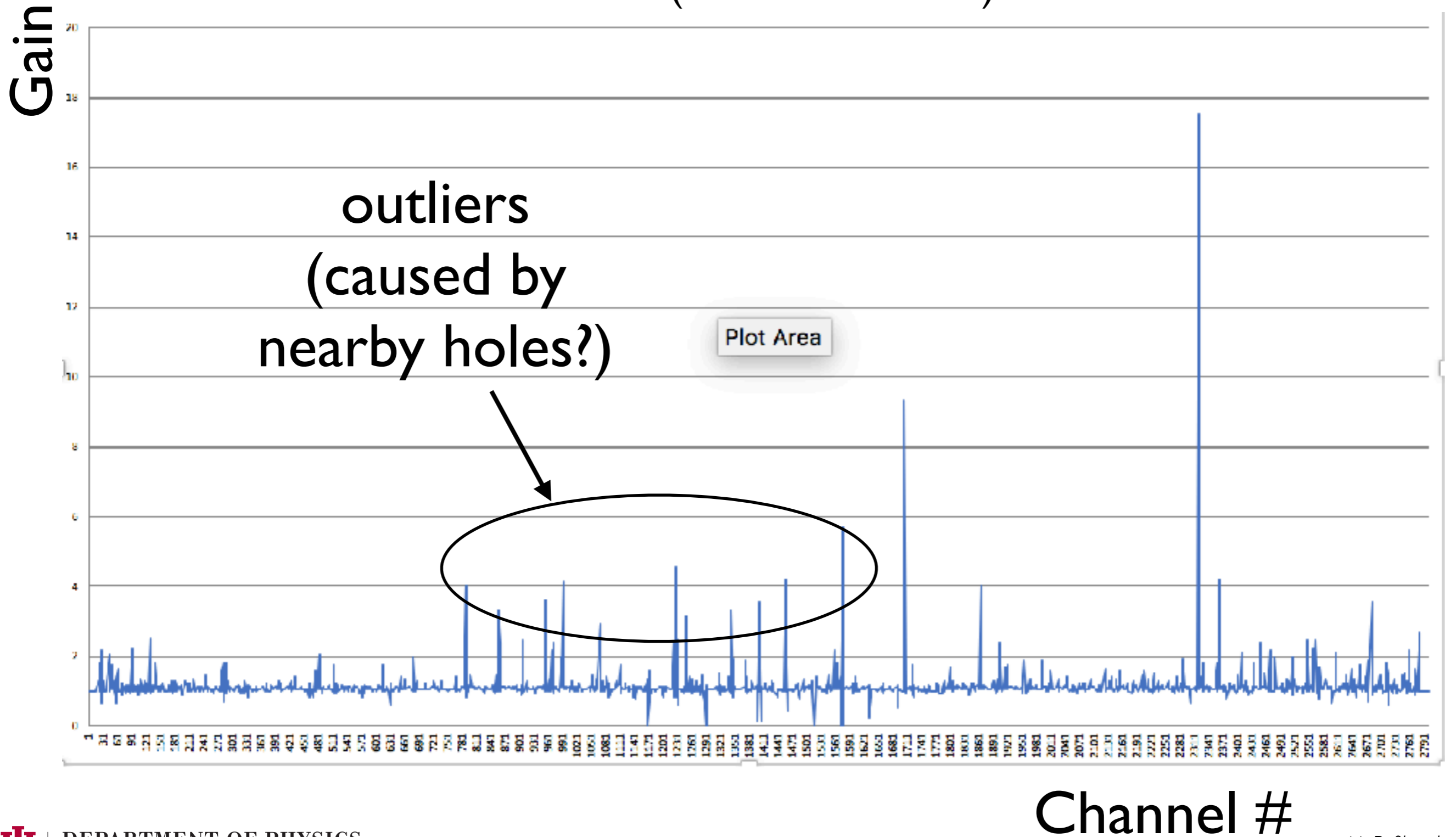
Data

MC



FCAL Gains

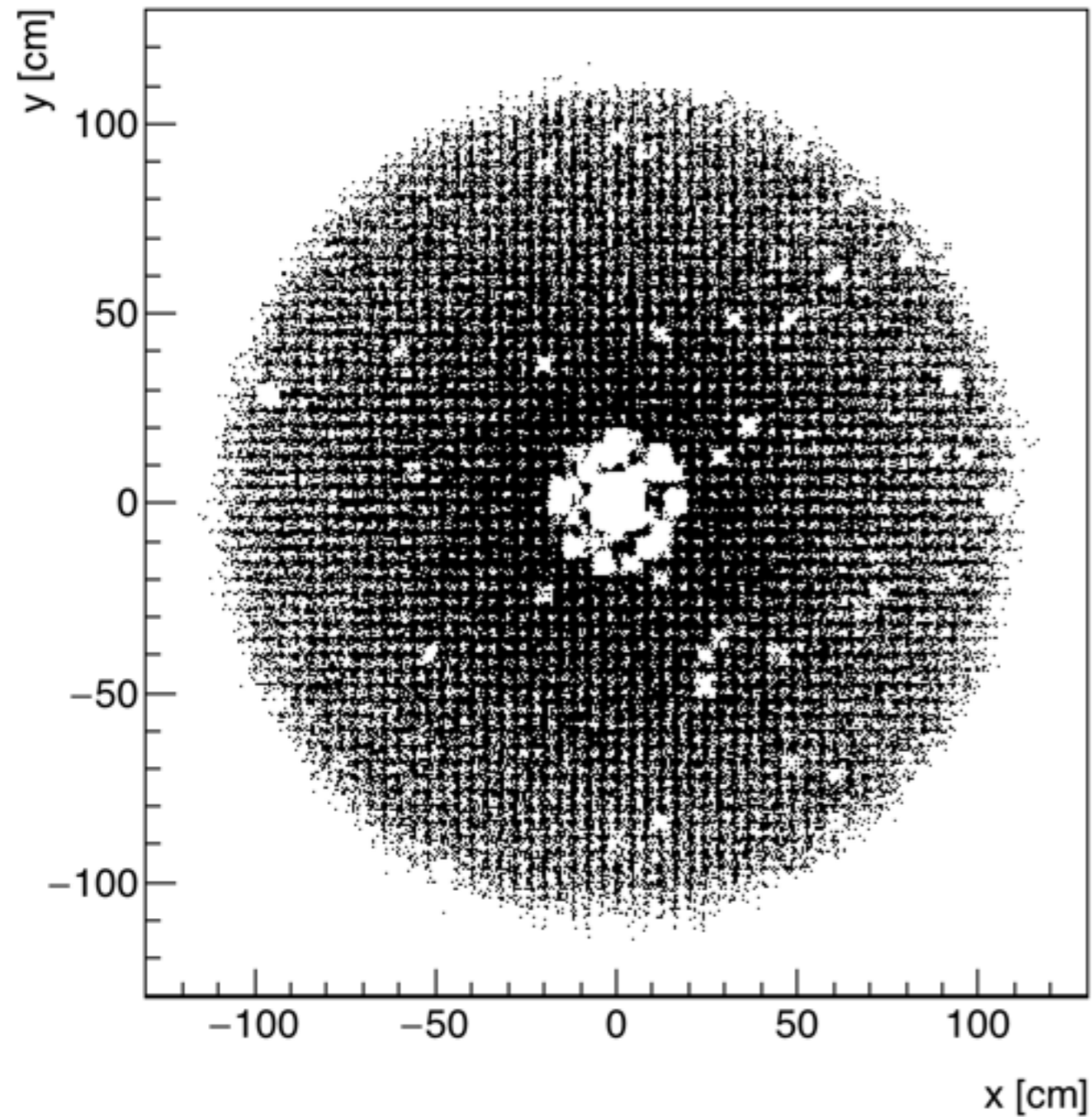
from ccdb (thanks Mike Staib)



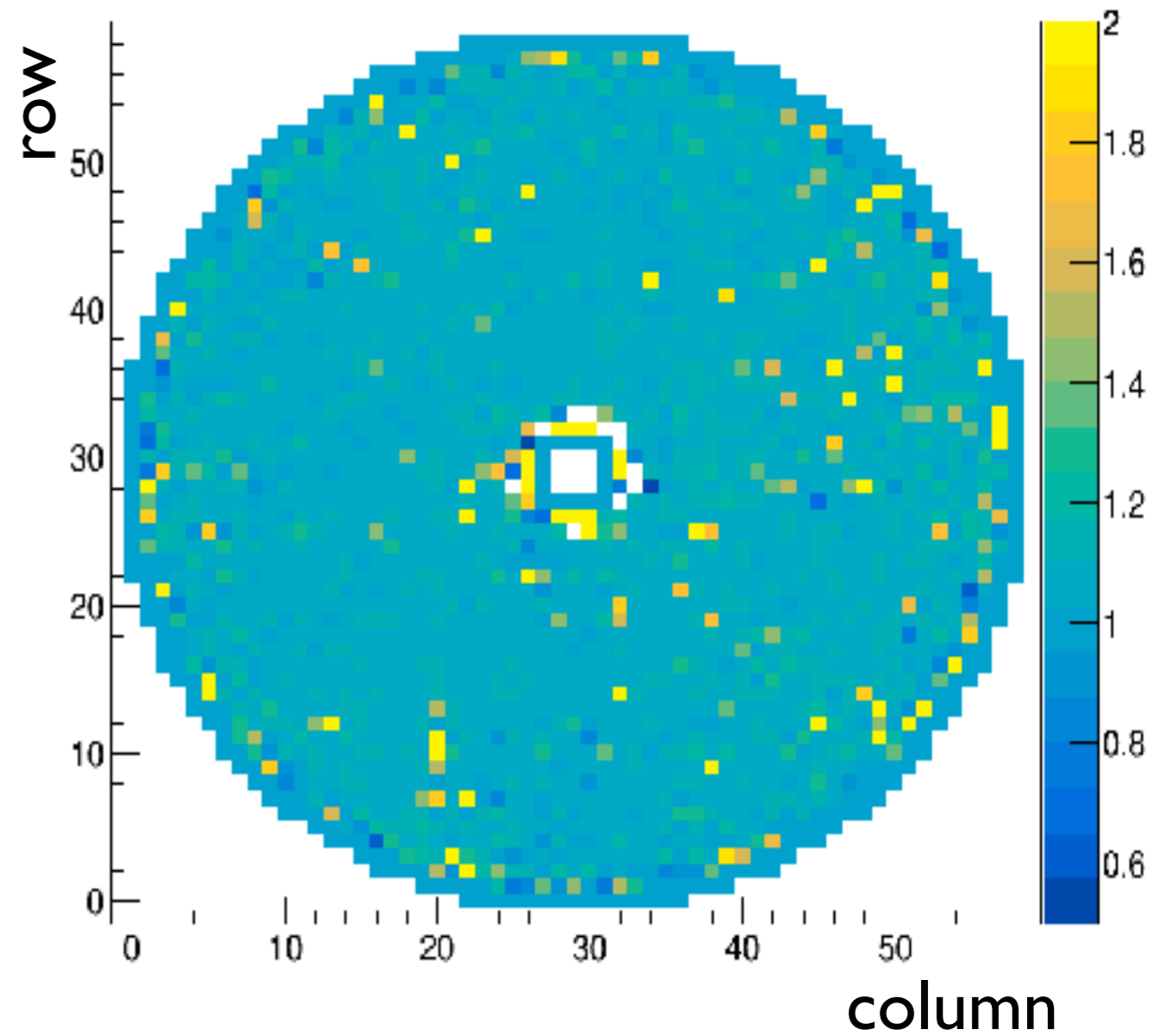
DEPARTMENT OF PHYSICS

INDIANA UNIVERSITY
College of Arts and Sciences
Bloomington

FCAL Gains



From Mike

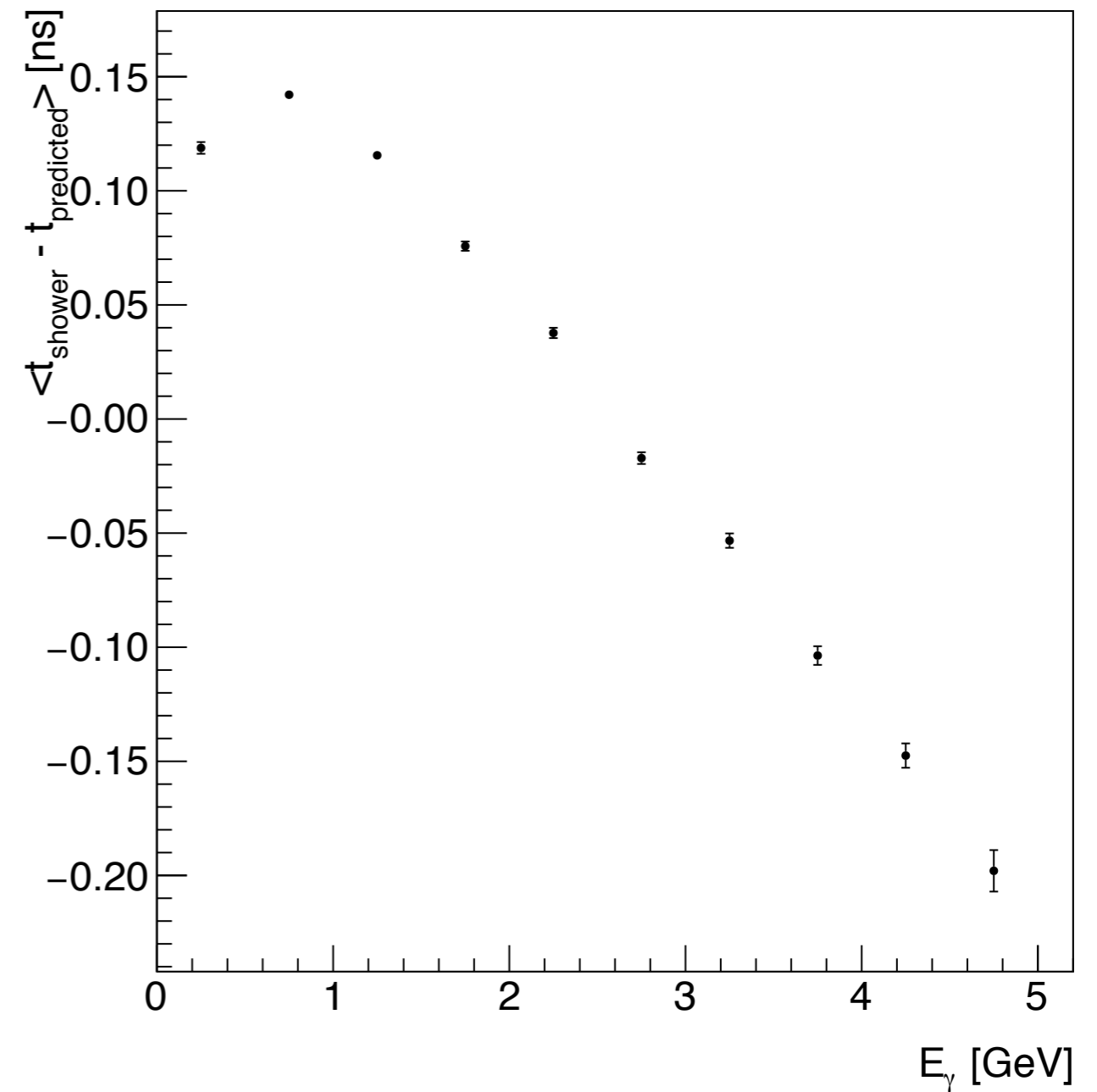
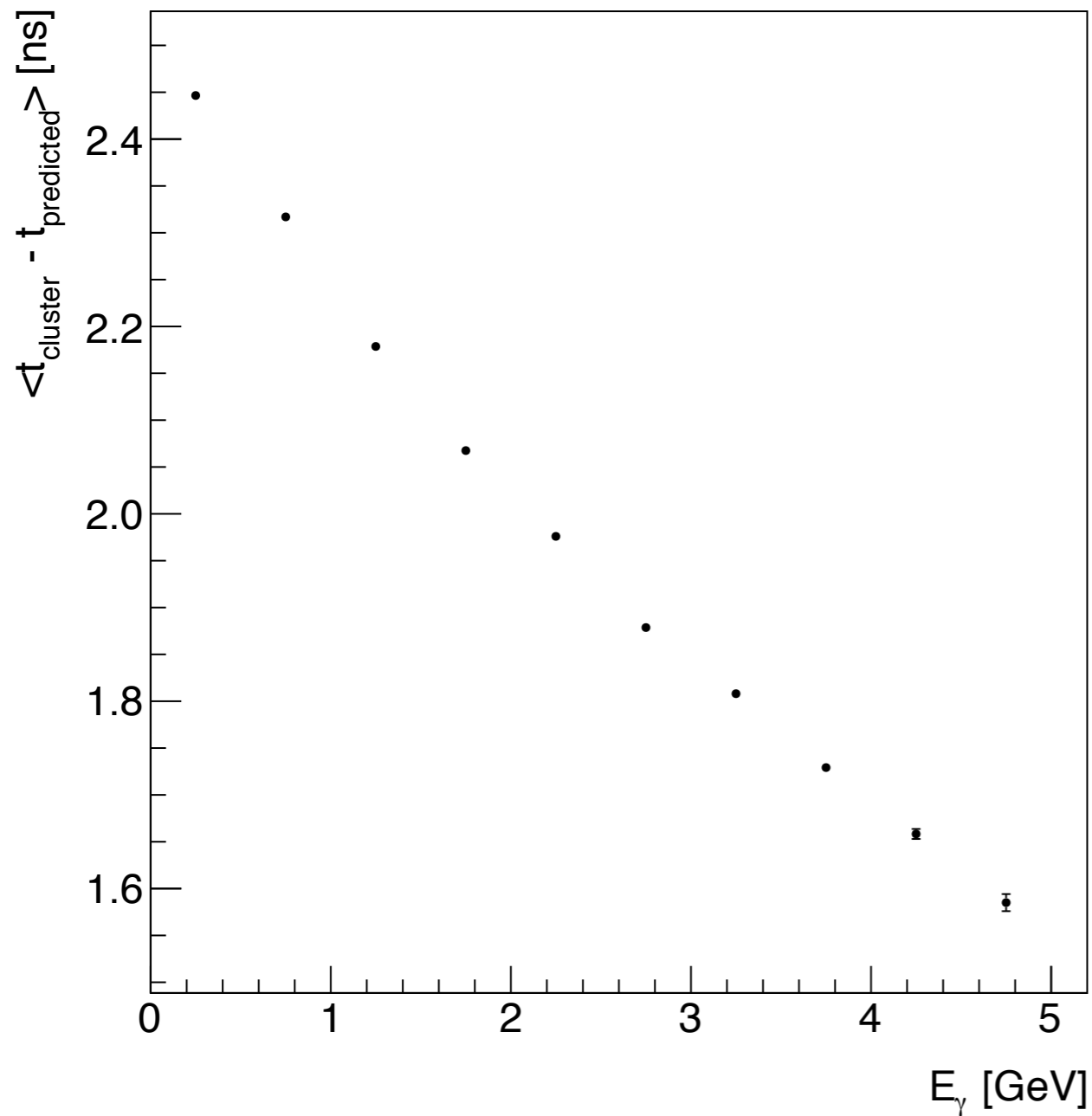


FCAL Timing

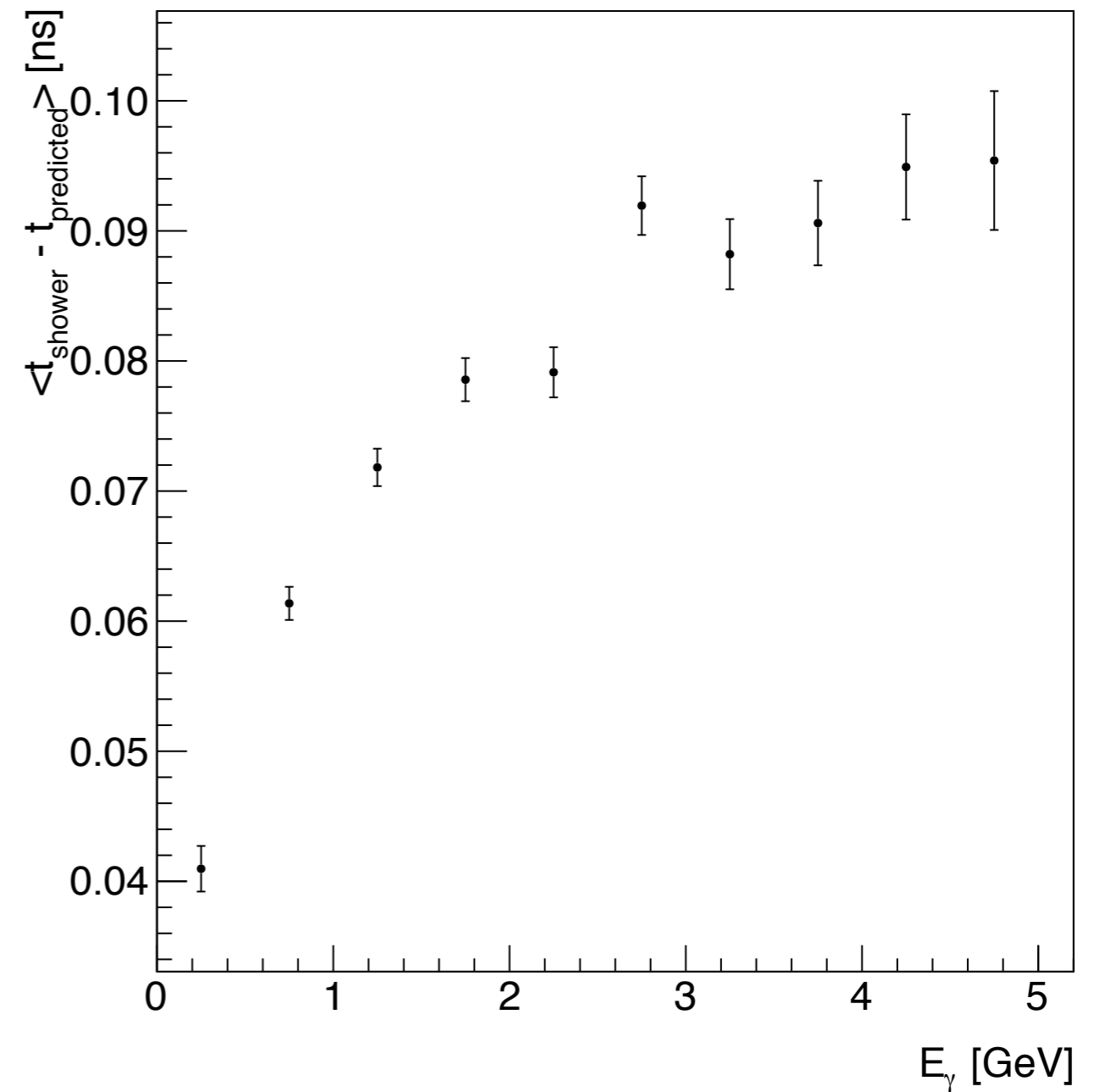
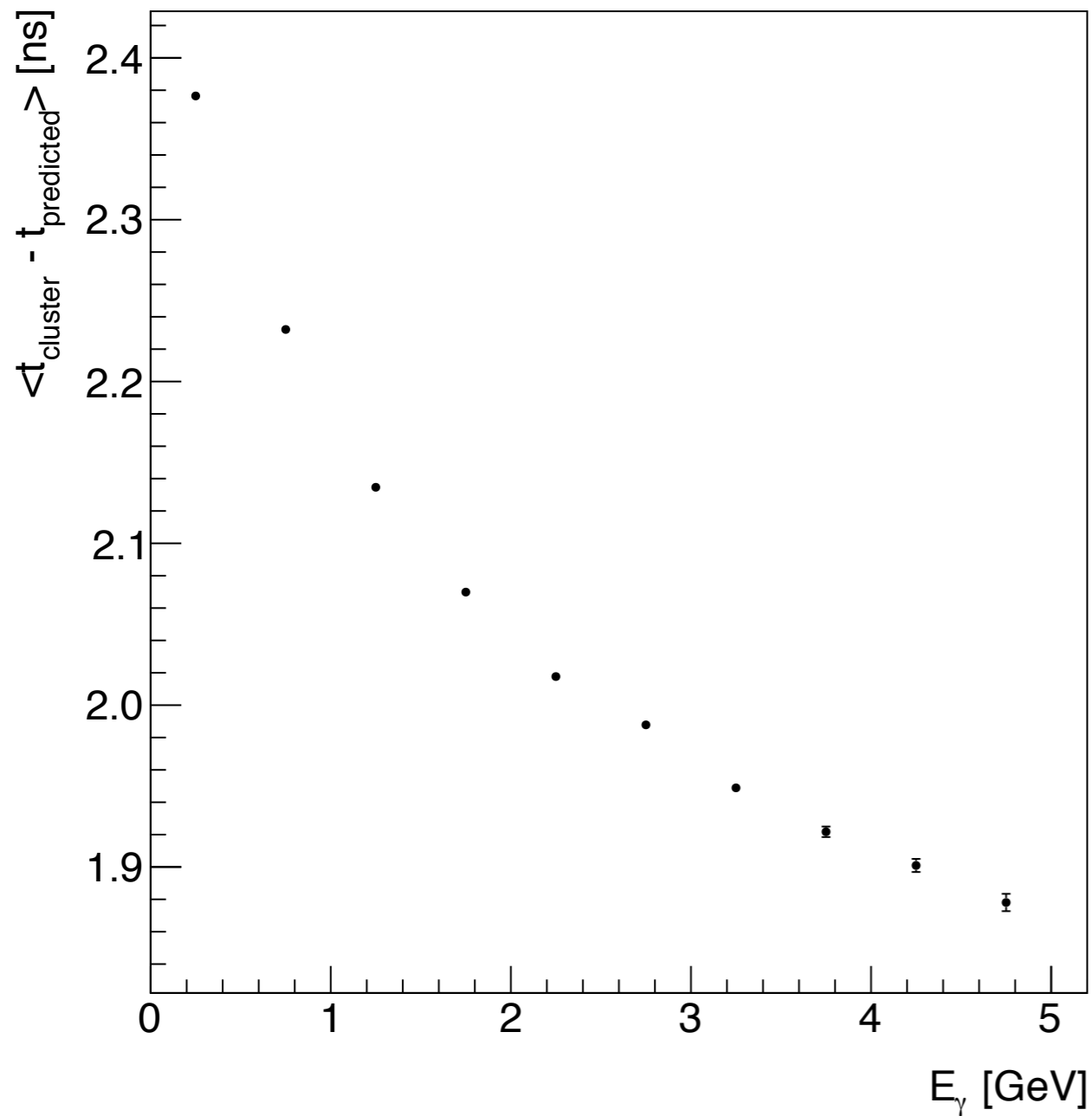
- Current algorithm:
 - create “clusters” in 2D
 - set cluster time to time of most energetic hit
 - create a “shower” from each cluster by translating z-coordinate along flight path to a depth determined by cluster energy
 - apply energy-dependent timing correction to shower (due to effective speed of light in the block)
- Cross check the timing correction with true photons:
 - predicted time = RF time at target center + (distance to depth-corrected shower center) / c
 - check existing correction: shower time - predicted time
 - derive a new correction: cluster time - predicted time



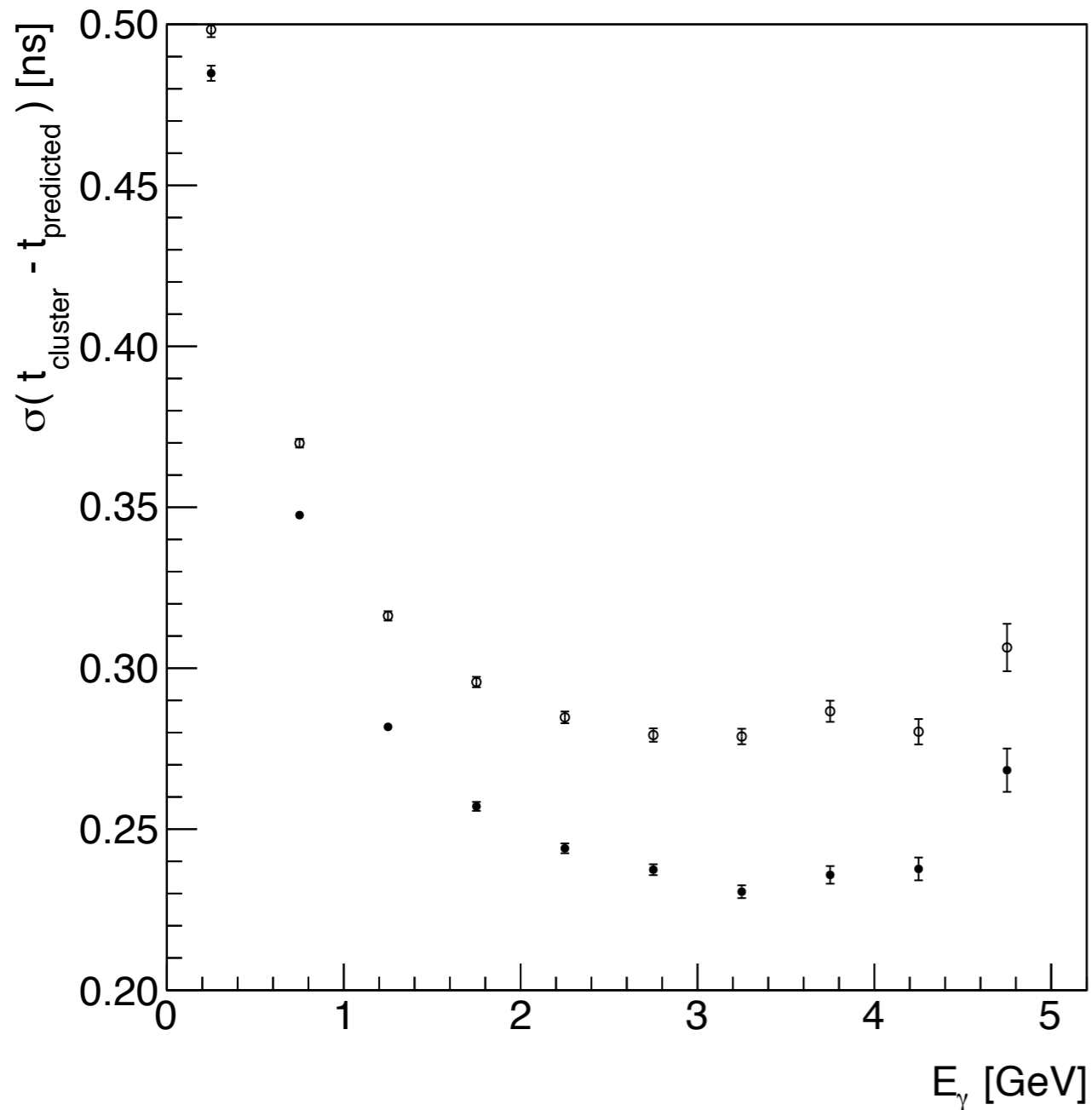
Timing Corrections (Data)



Timing Corrections (MC)



Time Resolution (Data)



- Standard algorithm (hollow circles) sets cluster time to time of maximum energy hit
- Improvement (solid circles) can be made by using energy weighted average of all hits in cluster
- Propose to implement this change first, then revise average time correction
- MC resolution has no energy dependence and is about 420 ps

Summary and Other Notes

- Energy resolution: emphasis has been on understanding it in data
 - are the techniques for measuring it sound? why is the floor term so large?
 - degrading resolution in MC is relatively straightforward
- Hit (block) efficiency:
 - in addition to dead channels from LED runs may have effectively dead channels due to poorly determined gain constants
- Geometry:
 - would like to correct/examine gain constants for revised geometry
 - push changes to ccdb and block size simultaneously
 - long term: restructure FCAL geometry class
- Timing:
 - improve time resolution by using energy-weighted time
 - implement new energy-dependent time correction

