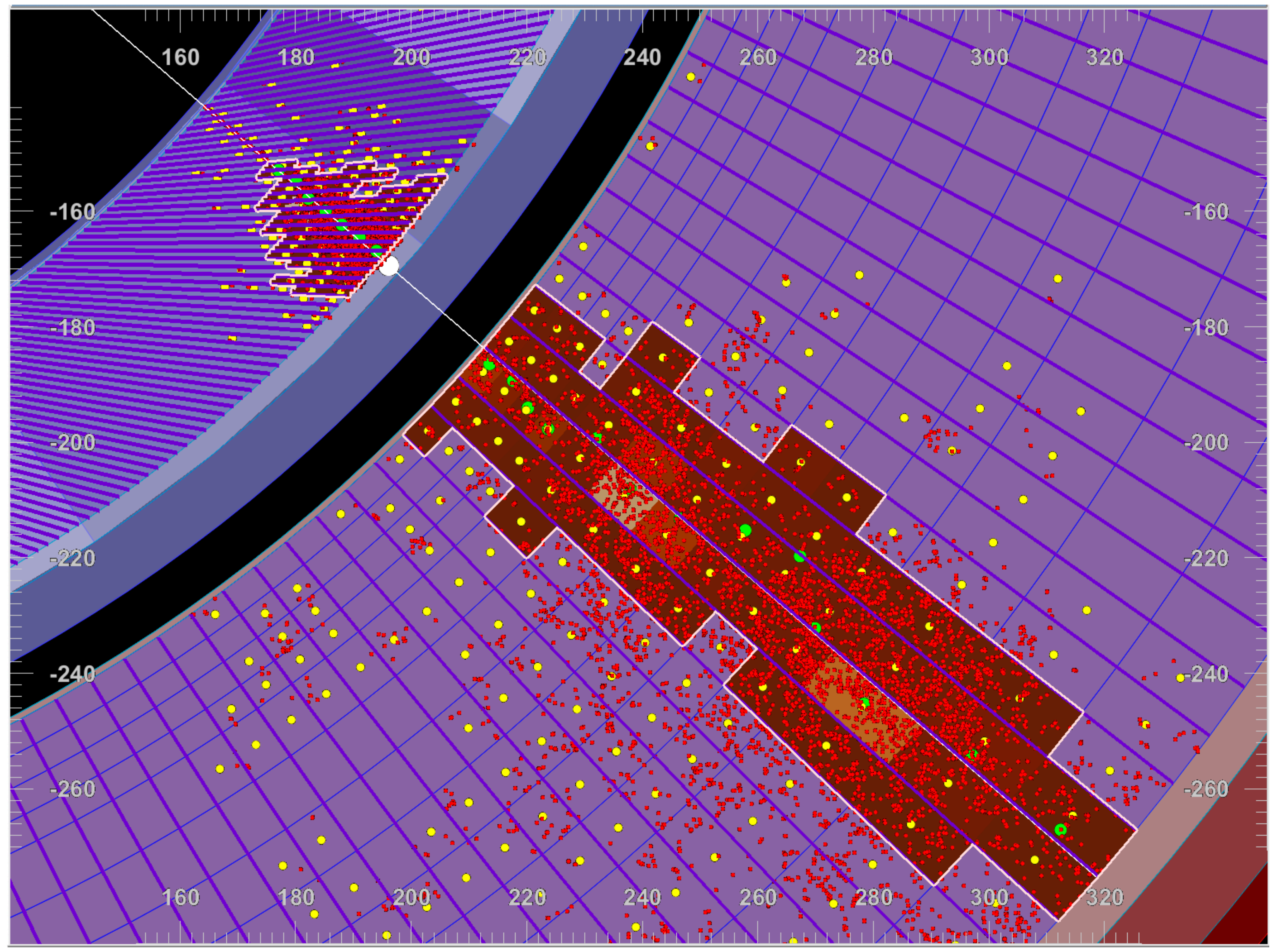
Progress on ALLEGRO ECAL barrel full simulation

Giovanni Marchiori, Tong Li, Alexis Maloizel (APC Paris)
with G. Bernardi (APC Paris), N. Morange (IJCLab Orsay)

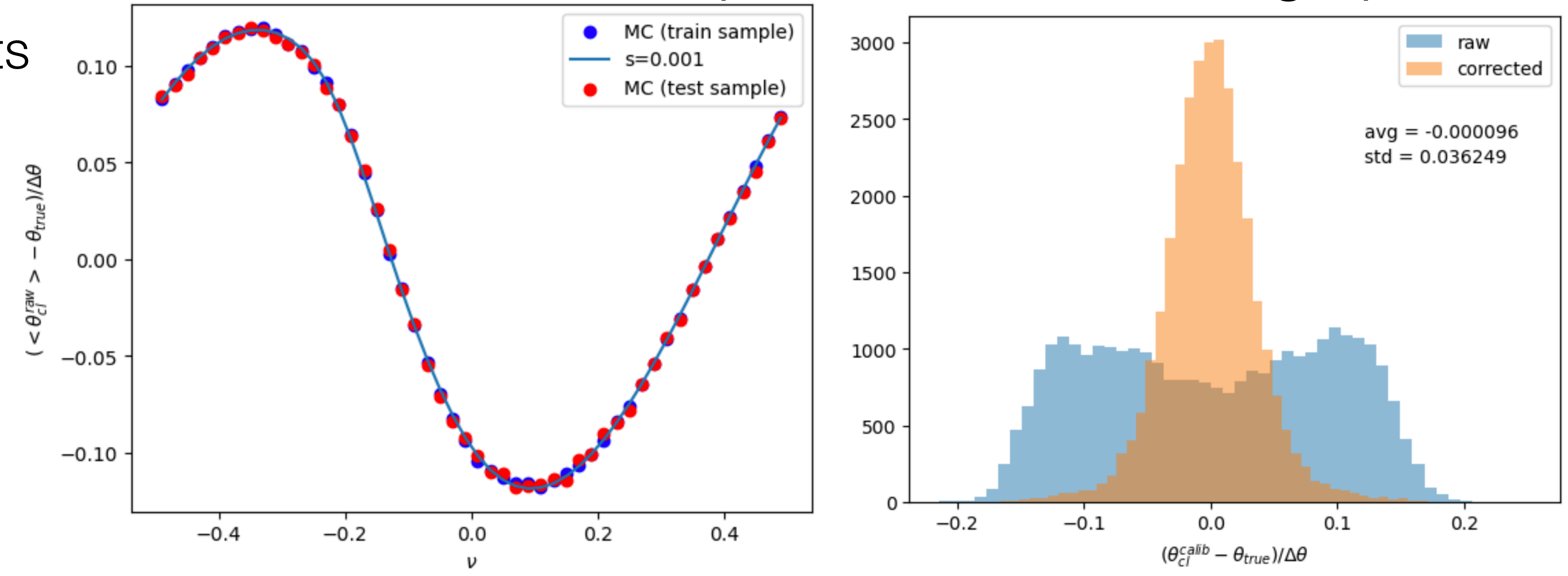
WP2 parallel session in ECFA DRD-calo workshop
CERN, 10 April 2024

Outline

- Latest presentation in the ALLEGRO meetings: 7/12/2023 (link)
- Some updates shown in FCC physics week in Annecy: (link)
  - Implemented topological and SW clustering for ECAL based on new segmentation
  - Implemented ECAL+HCAL topoclustering
  - Migrated detector model/segmentation to k4geo
  - Initial S-shape studies
- Activities performed in the past ~3 months or ongoing
  - Calculation of cluster per-layer barycentre with log(E) weights
  - Calculation and persistence of layer energy and barycenter
  - MVA-based e/gamma energy calibration
  - Photon/π^0 discrimination
  - Updated event display



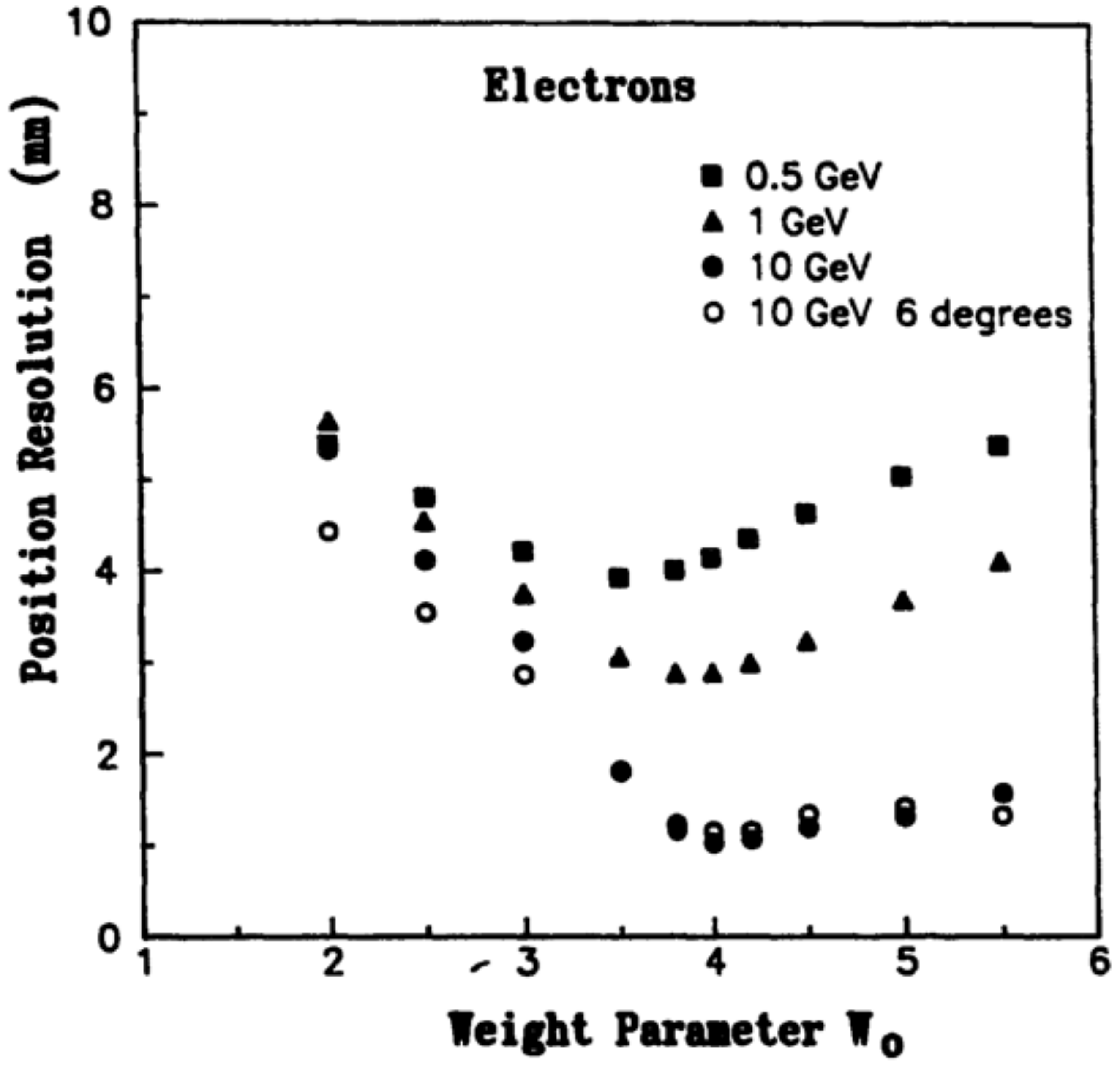
Topocluster from charge pion

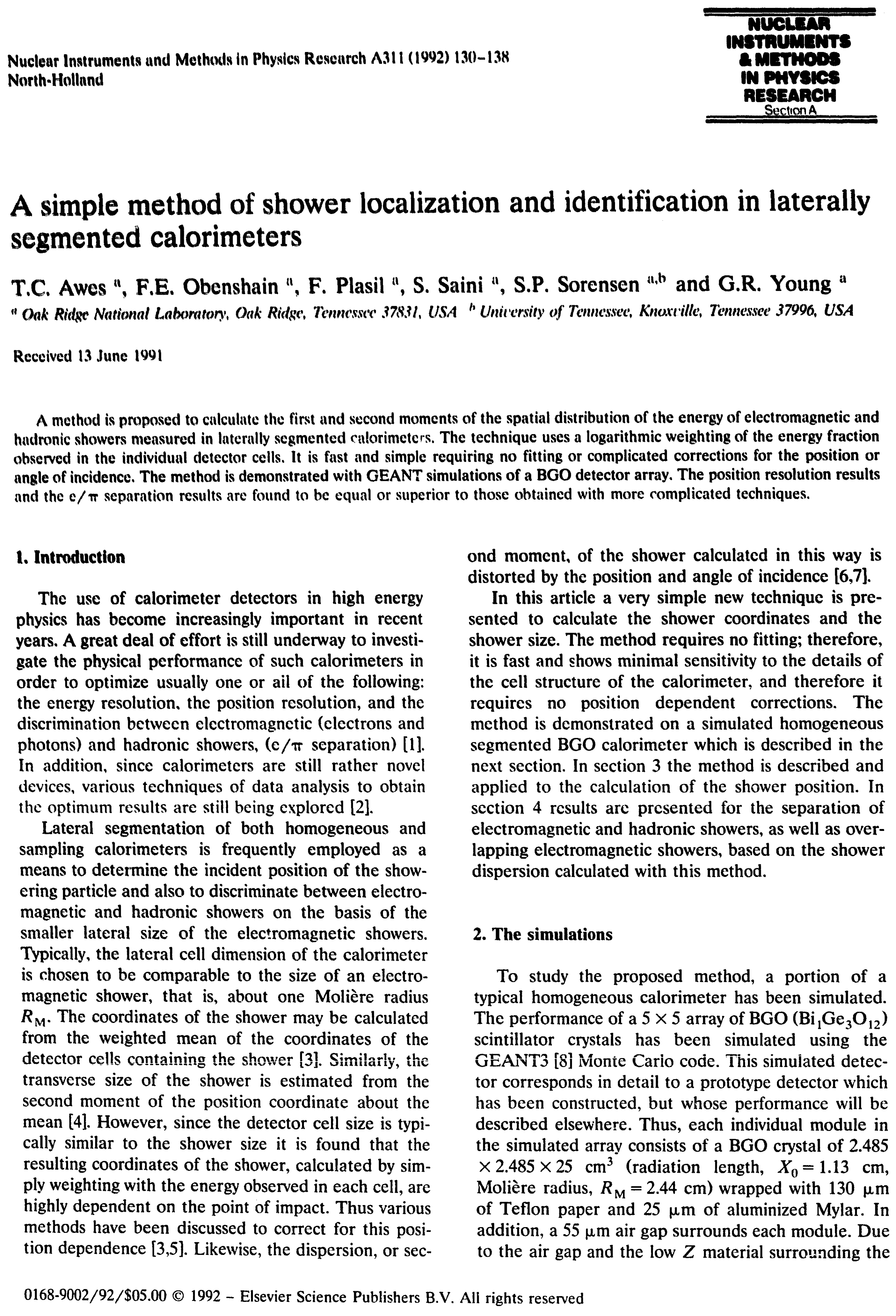


Position/direction reconstruction: S-curve θ correction and resolution

ECAL cluster layer barycentres calculation with log(E) weights

- Calculation of cluster and layer barycenter in θ is biased by limited granularity of detector and nominal positioning of digitised hits at centre of cell => S-curve effect (see plot in previous slide)
- Effect can be cured with dedicated S-curve correction (parametrisation or NN regression shown in previous meetings)
- Alternative way to restore a Gaussian response: use log(E) weights (reference)
    - Each cell *c* in layer *i* is weighted by $w_{c,i} = max(0, w_0 + log (E_c/E_i))$
    - Offset w_0 is chosen to minimise the theta position resolution




Nuclear Instruments and Methods in Physics Research A311 (1992) 130–138
North-Holland

NUCLEAR INSTRUMENTS & METHODS IN PHYSICS RESEARCH
Section A

A simple method of shower localization and identification in laterally segmented calorimeters

T.C. Awes [a], F.E. Obenshain [a], F. Plasil [a], S. Saini [a], S.P. Sorensen [a,b] and G.R. Young [a]
[a] Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA [b] University of Tennessee, Knoxville, Tennessee 37996, USA

Received 13 June 1991

A method is proposed to calculate the first and second moments of the spatial distribution of the energy of electromagnetic and hadronic showers measured in laterally segmented calorimeters. The technique uses a logarithmic weighting of the energy fraction observed in the individual detector cells. It is fast and simple requiring no fitting or complicated corrections for the position or angle of incidence. The method is demonstrated with GEANT simulations of a BGO detector array. The position resolution results and the e/π separation results are found to be equal or superior to those obtained with more complicated techniques.

1. Introduction

The use of calorimeter detectors in high energy physics has become increasingly important in recent years. A great deal of effort is still underway to investigate the physical performance of such calorimeters in order to optimize usually one or ail of the following: the energy resolution, the position resolution, and the discrimination between electromagnetic (electrons and photons) and hadronic showers, (e/π separation) [1]. In addition, since calorimeters are still rather novel devices, various techniques of data analysis to obtain the optimum results are still being explored [2].

Lateral segmentation of both homogeneous and sampling calorimeters is frequently employed as a means to determine the incident position of the showering particle and also to discriminate between electromagnetic and hadronic showers on the basis of the smaller lateral size of the electromagnetic showers. Typically, the lateral cell dimension of the calorimeter is chosen to be comparable to the size of an electromagnetic shower, that is, about one Molière radius R_M. The coordinates of the shower may be calculated from the weighted mean of the coordinates of the detector cells containing the shower [3]. Similarly, the transverse size of the shower is estimated from the second moment of the position coordinate about the mean [4]. However, since the detector cell size is typically similar to the shower size it is found that the resulting coordinates of the shower, calculated by simply weighting with the energy observed in each cell, are highly dependent on the point of impact. Thus various methods have been discussed to correct for this position dependence [3,5]. Likewise, the dispersion, or second moment, of the shower calculated in this way is distorted by the position and angle of incidence [6,7].

In this article a very simple new technique is presented to calculate the shower coordinates and the shower size. The method requires no fitting; therefore, it is fast and shows minimal sensitivity to the details of the cell structure of the calorimeter, and therefore it requires no position dependent corrections. The method is demonstrated on a simulated homogeneous segmented BGO calorimeter which is described in the next section. In section 3 the method is described and applied to the calculation of the shower position. In section 4 results are presented for the separation of electromagnetic and hadronic showers, as well as overlapping electromagnetic showers, based on the shower dispersion calculated with this method.

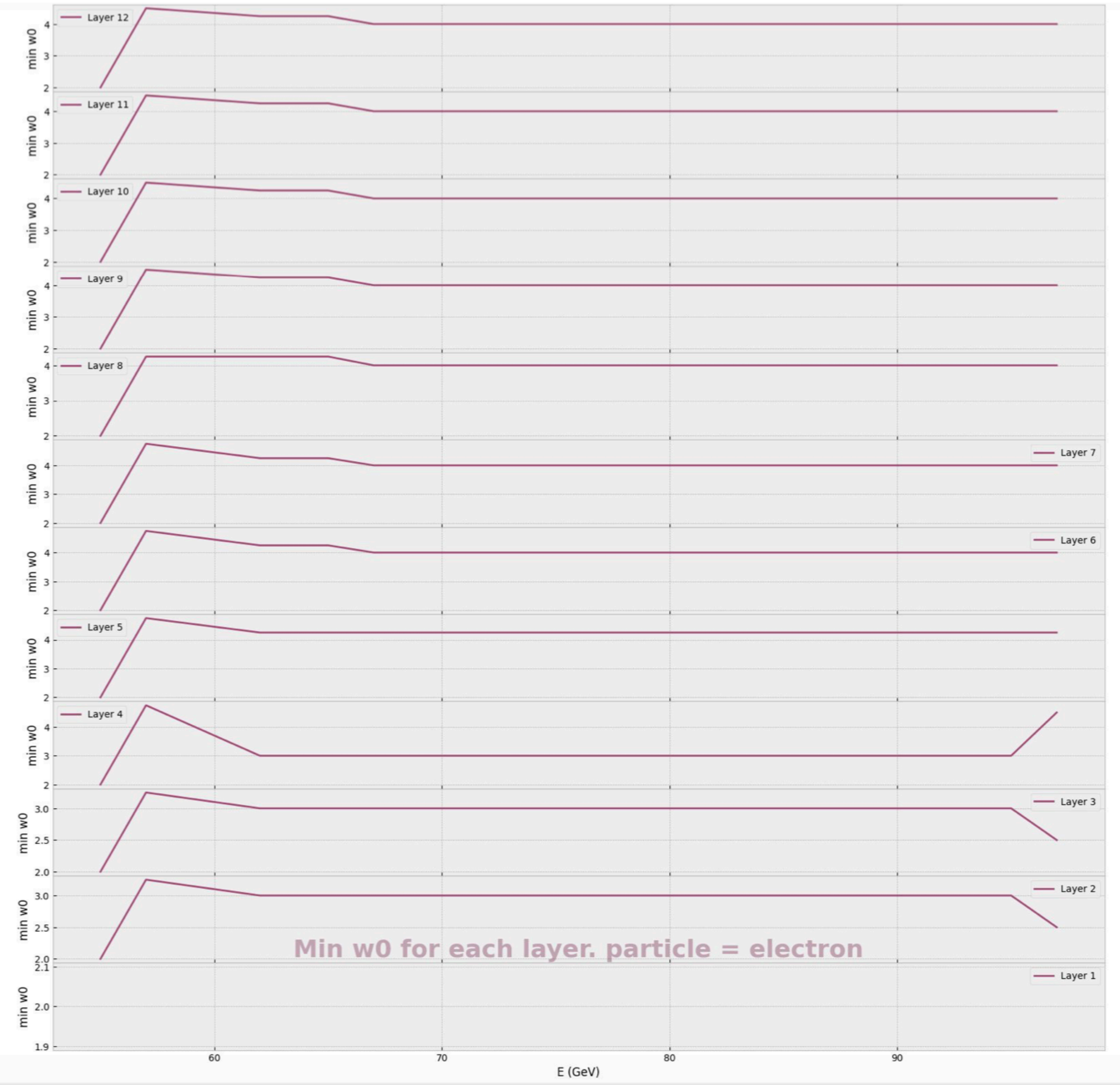
2. The simulations

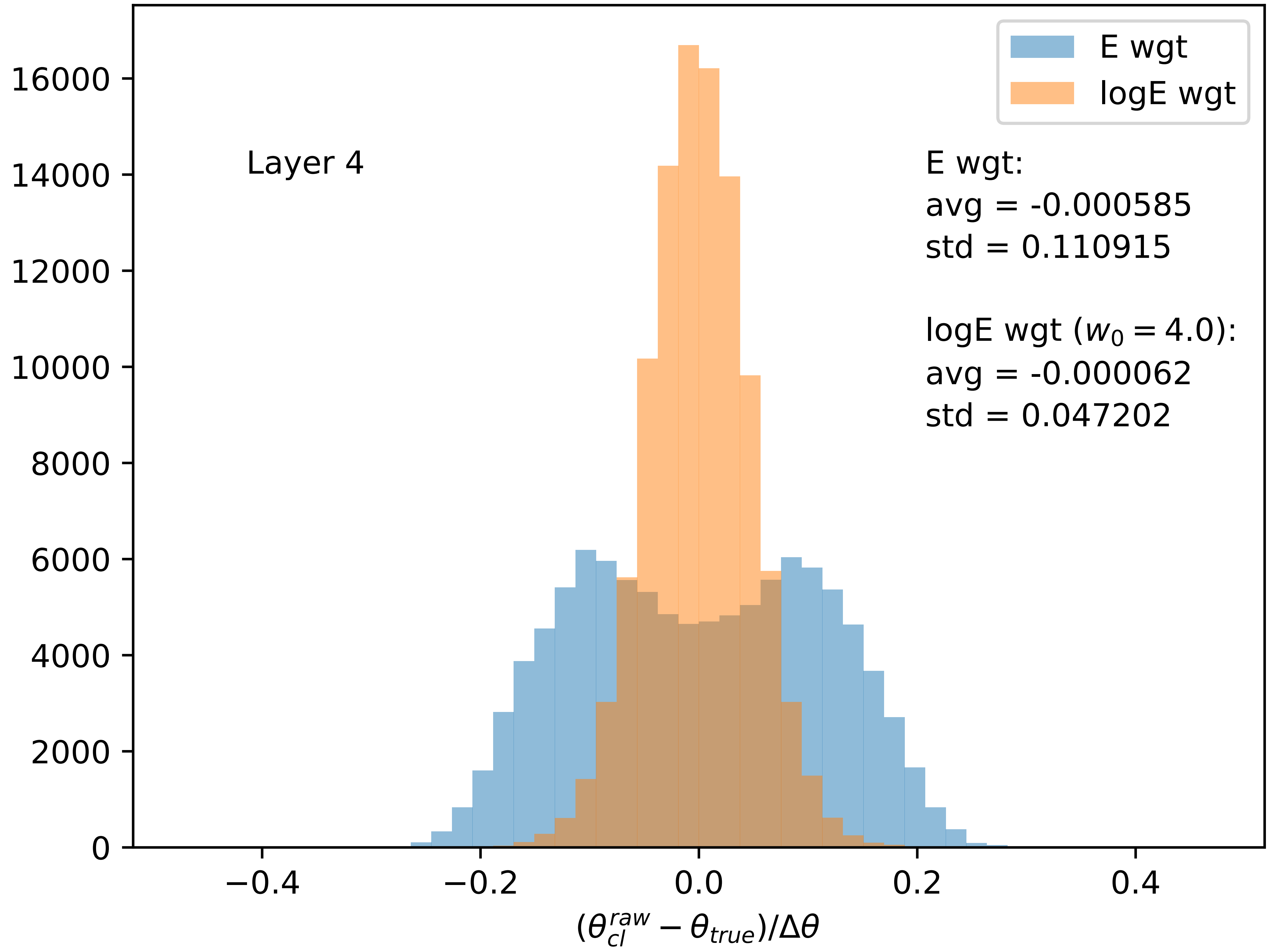
To study the proposed method, a portion of a typical homogeneous calorimeter has been simulated. The performance of a 5 × 5 array of BGO ($Bi_1Ge_3O_{12}$) scintillator crystals has been simulated using the GEANT3 [8] Monte Carlo code. This simulated detector corresponds in detail to a prototype detector which has been constructed, but whose performance will be described elsewhere. Thus, each individual module in the simulated array consists of a BGO crystal of 2.485 × 2.485 × 25 cm^3 (radiation length, $X_0 = 1.13$ cm, Molière radius, $R_M = 2.44$ cm) wrapped with 130 μm of Teflon paper and 25 μm of aluminized Mylar. In addition, a 55 μm air gap surrounds each module. Due to the air gap and the low Z material surrounding the

ECAL cluster layer barycentres calculation with log(E) weights

- Study done for topoclusters, to be repeated for SW clusters, and to be updated for new ECAL barrel readout (see slide 10)
- w_0 found to be reasonably independent of particle hypothesis and energy. Calculation implemented in Gaudi alg (next slide)

Alexis, GM



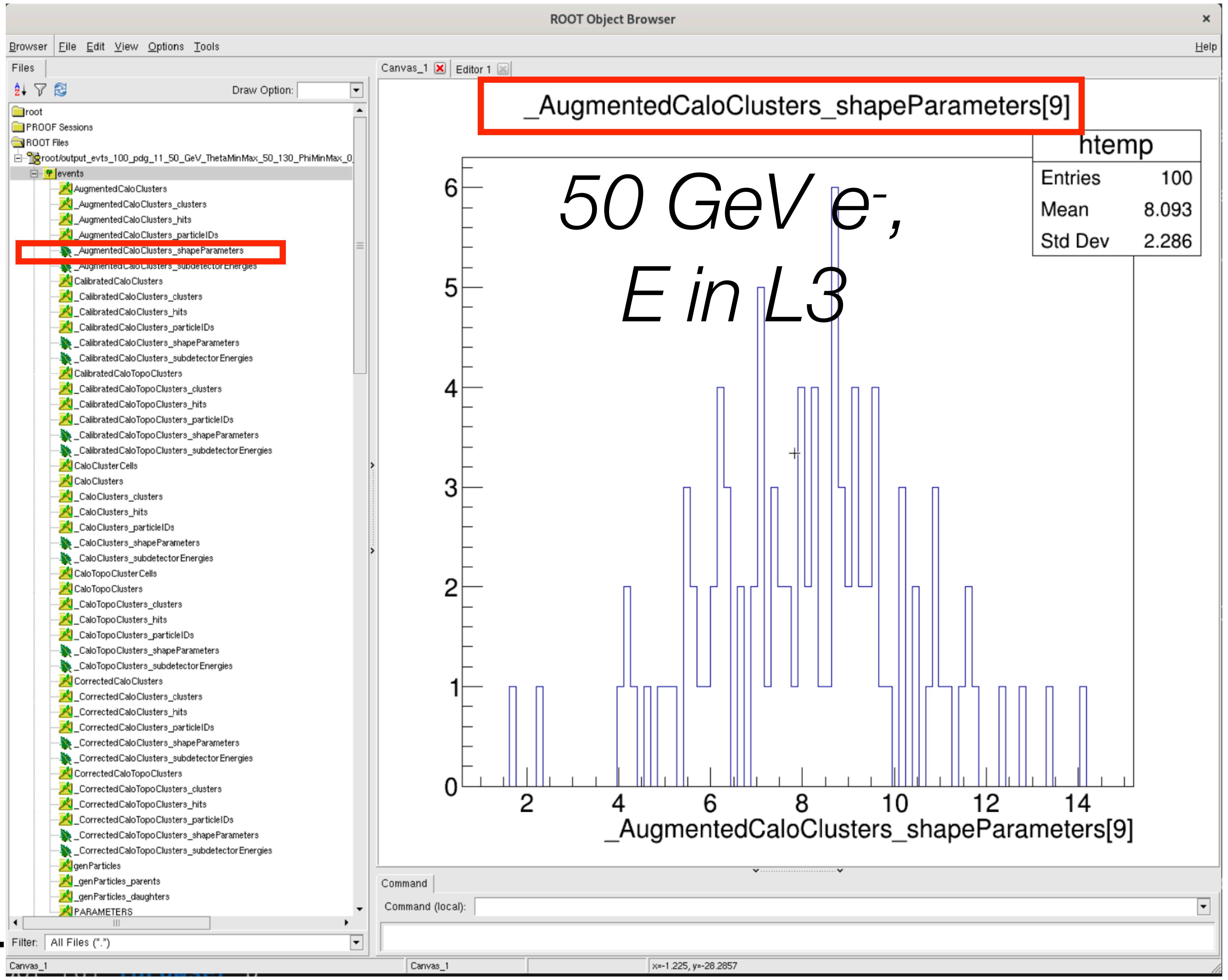


- Ongoing/to-do:
  - compare with regression and with dedicated S-curve correction
  - parametrise angular resolution vs E and use in reconstruction of particle direction

ECAL cluster shape parameters

- In key4hep, clusters can be augmented with “shapeParameters" vectors that can hold user-computed variables of interest
- Metadata in ROOT file can be used to record mnemonic string listing the variables added to the shape parameters
- Written Gaudi algorithm that calculates for each ECAL cluster a set of quantities and saves them as shapeParameters
  - Energy, theta and phi barycentre of each layer

```
gmarchio@apcatlas01:~/work/fcc/allegro/fullsim/LAr_scripts/FCCSW_ecal % python
Python 3.10.13 (main, Mar  8 2024, 20:43:16) [GCC 11.3.1 20221121 (Red Hat 11.3.1-4)
] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from podio.reading import get_reader
>>> reader = get_reader("root/output_evts_100_pdg_11_50_GeV_ThetaMinMax_50_130_PhiMi
nMax_0_6.283185307179586_MagneticField_False_NoiseFalse.root")
>>> frame = reader.get("metadata")[0]
>>> decorations = frame.get_parameter('AugmentedCaloClusters__shapeParameterNames')
>>> print(decorations)
E0:theta0:phi0:E1:theta1:phi1:E2:theta2:phi2:E3:theta3:phi3:E4:theta4:phi4:E5:theta5
:phi5:E6:theta6:phi6:E7:theta7:phi7:E8:theta8:phi8:E9:theta9:phi9:E10:theta10:phi10:
E11:theta11:phi11
```



Alexis, GM

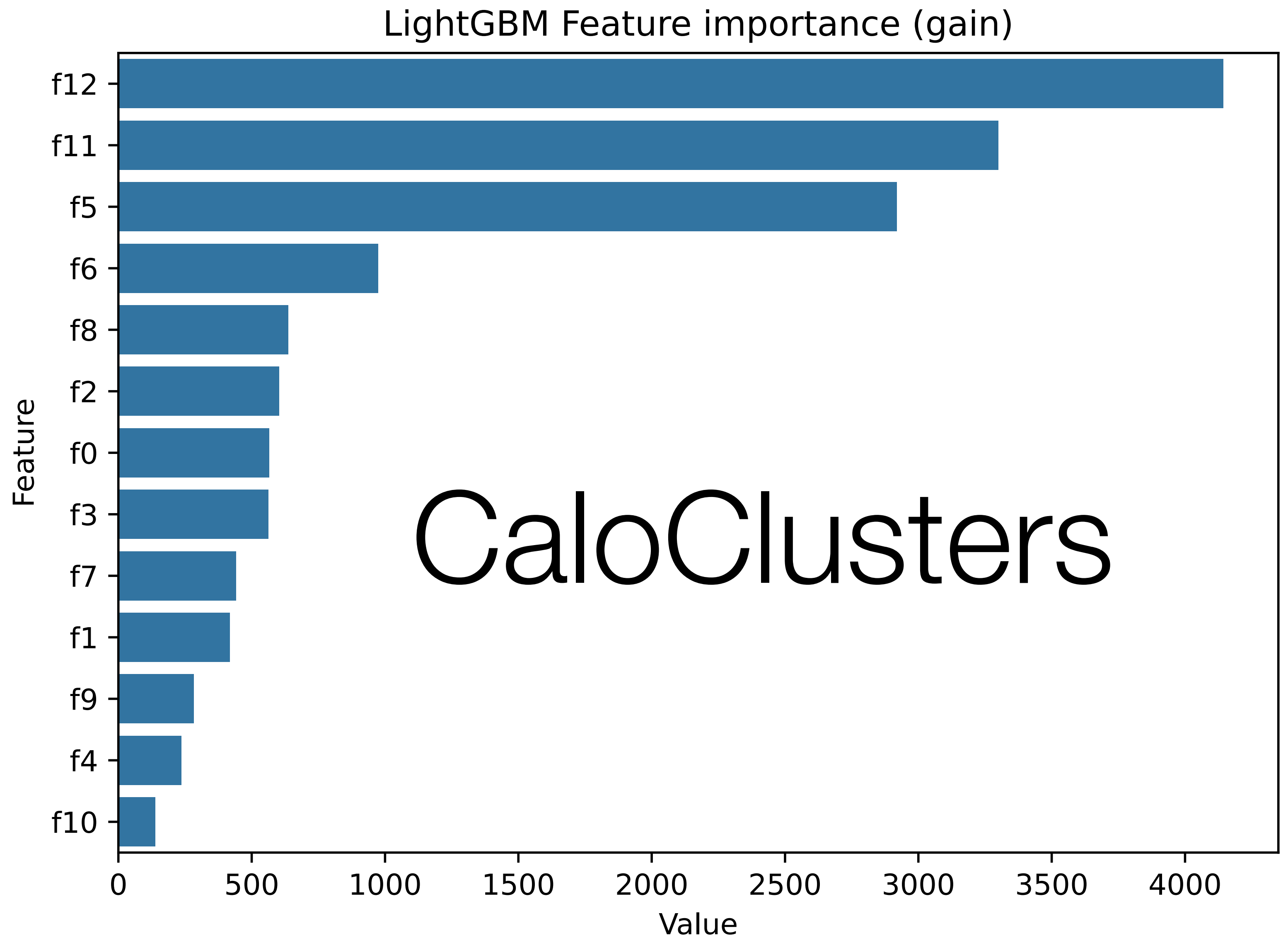
- Pull requests: k4RecCalorimeter, LAr_scripts
- Ongoing/to-do: expand alg to calculate more variables of interest (e.g. shower shape variables for particle identification). Re-use calculated quantities in downstream algos

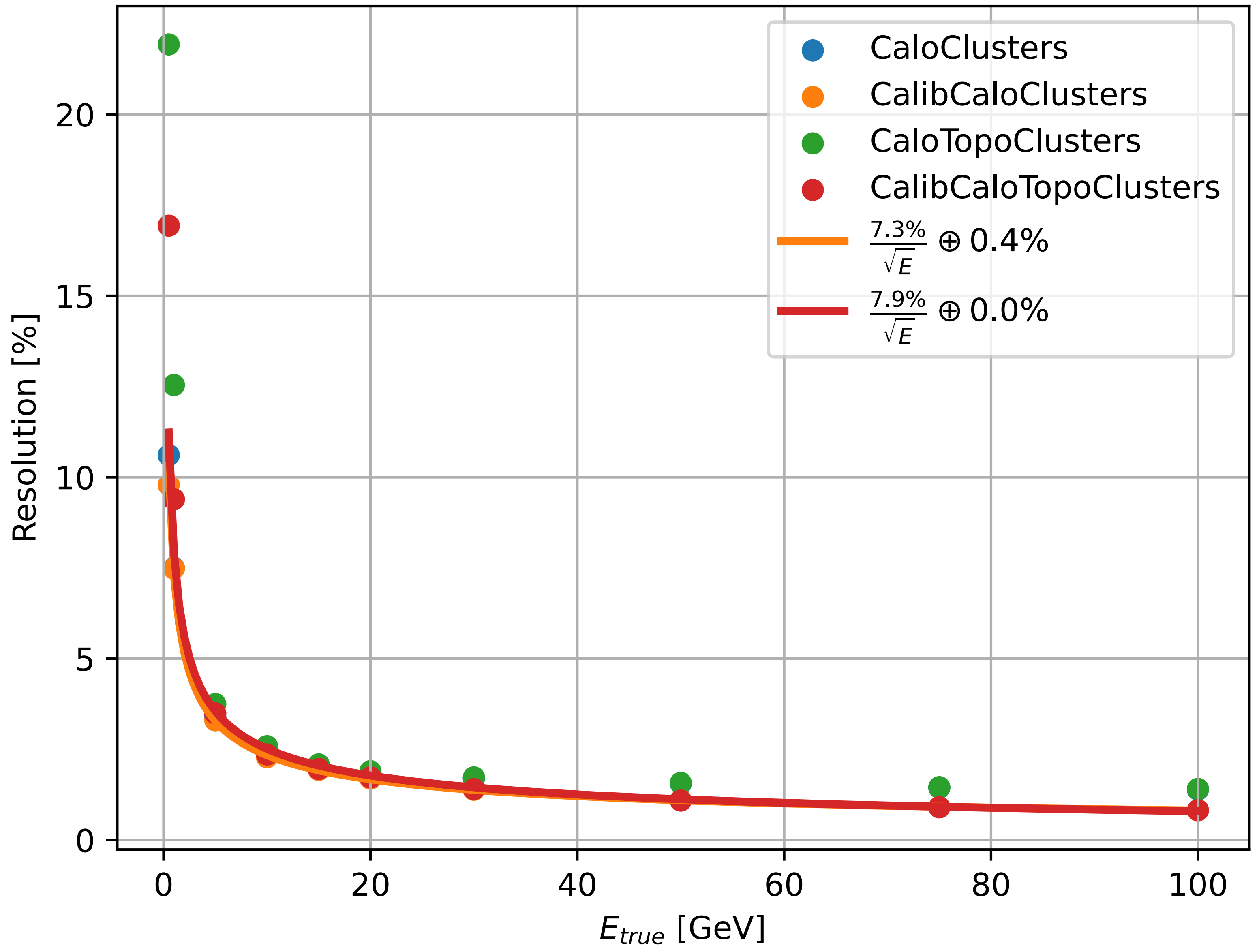
MVA-based e/gamma energy calibration

- BDT-based calibration based on measured energies in each layer already shown to perform better than simpler parametric correction for estimated upstream and downstream energy losses
- So far, BDT regression can be trained (using XGBoost) and applied offline running some python script on the ROOT files produced by the full simulation
- Code to convert the XGBoost model to a portable ONNX format (package) and to import it and apply it to clusters in a Gaudi algorithm running inside FCCSW (PRs: k4RecCalorimeter, LAr_scripts has been implemented
- The package with the XGB -> ONNX converter also include a couple of scripts for BDT training and evaluation with an alternative tool (LightGBM)
    - similar response, very fast training, smaller output files
    - Also LGBM model can be saved to ONNX format and used in the simulation, in alternative to XGBoost
    - Currently the LGBM training uses the energies per layer calculated by the offline analysis script used also for the XGBoost training
    - To-do: migrate the training code to using the energies per layer stored in the shapeParameters vector if the decorations are found in the input file metadata.

MVA-based e/gamma energy calibration

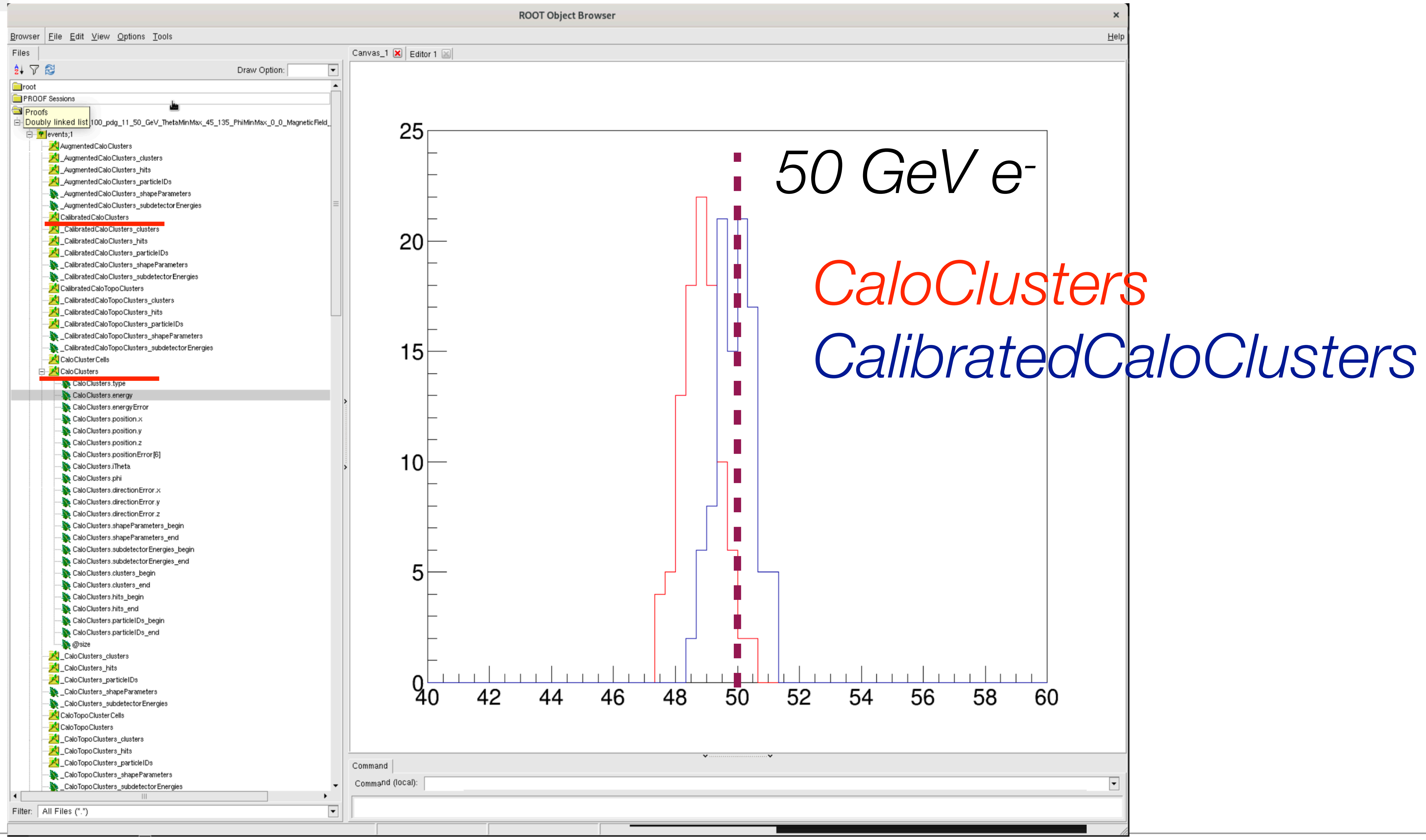
- Training with LGB





GM

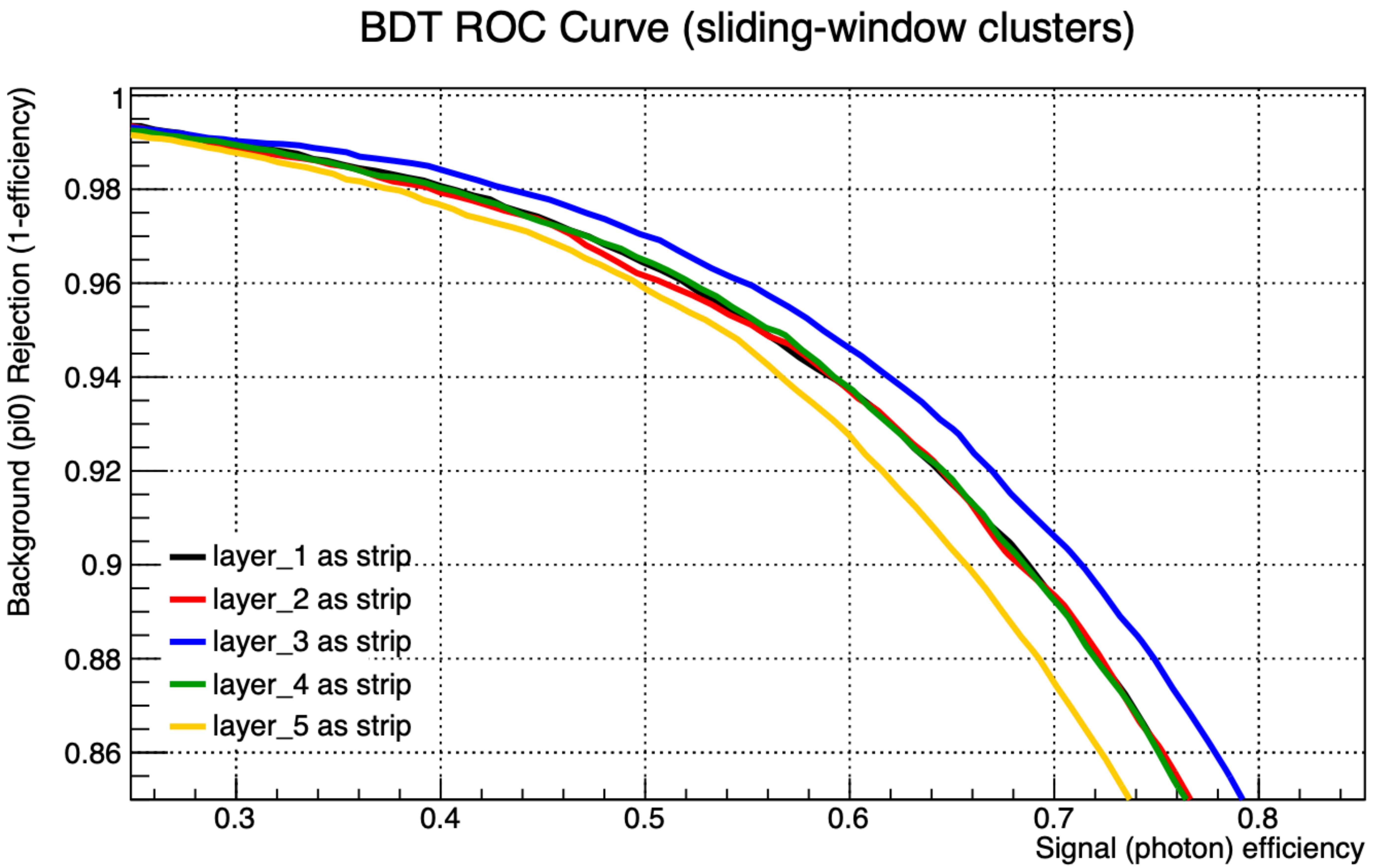
```
calibrateCaloClusters = CalibrateCaloClusters(
    "calibrateCaloClusters",
    inClusters="Augmented" + createClusters.clusters.Path,
    outClusters="Calibrated" + createClusters.clusters.Path,
    systemIDs=[4],
    numLayers=[11],
    firstLayerIDs=[0],
    readoutNames=[
        ecalBarrelReadoutName],
    layerFieldNames=["layer"],
    calibrationFile="lgbm_calibration-CaloClusters-v03.onnx",
    OutputLevel=INFO
)
```



γ/π^0 discrimination

- Goal: discriminate photons vs π^0 clusters based on shower shapes, exploiting the different transverse energy profiles
    - Start with simpler algorithms (e.g. BDTs based on shower shape variables computed from cells), obtain a baseline reference algorithm, before looking into fancier/more refined tools
    - Compare performance of different detector design
- Currently implemented the calculation of shower-shape variables (energy fraction, shower width in phi and theta, relative difference between local minimum and 2nd local max in energy profile vs theta, …) offline, from the cell collections saved in the ROOT files produced by the simulation
- TMVA then used to train a BDT and tune hyperparameters
- Ongoing work/to-do:
    - Consolidate list of input variables and BDT settings and assess performance of baseline algorithm
    - Move calculation of shower shape variables to Gaudi algorithm and save them as cluster shapeParameters
    - Train BDT with alternative frameworks (eg LGBM) and implement photon-ID algorithm in Gaudi
    - Check impact of detector parameters
    - Compare to more advanced algorithms

γ/π⁰ discrimination

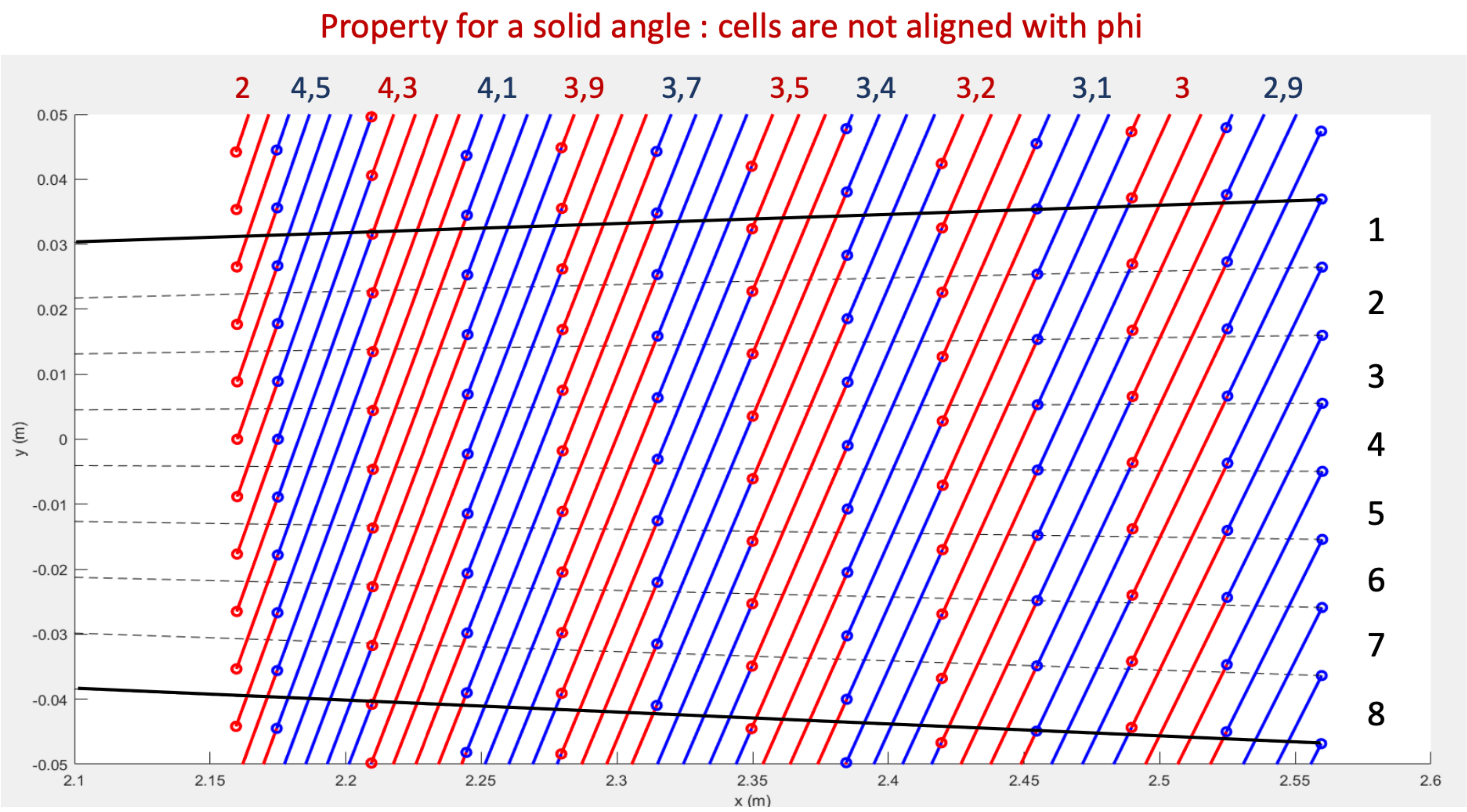


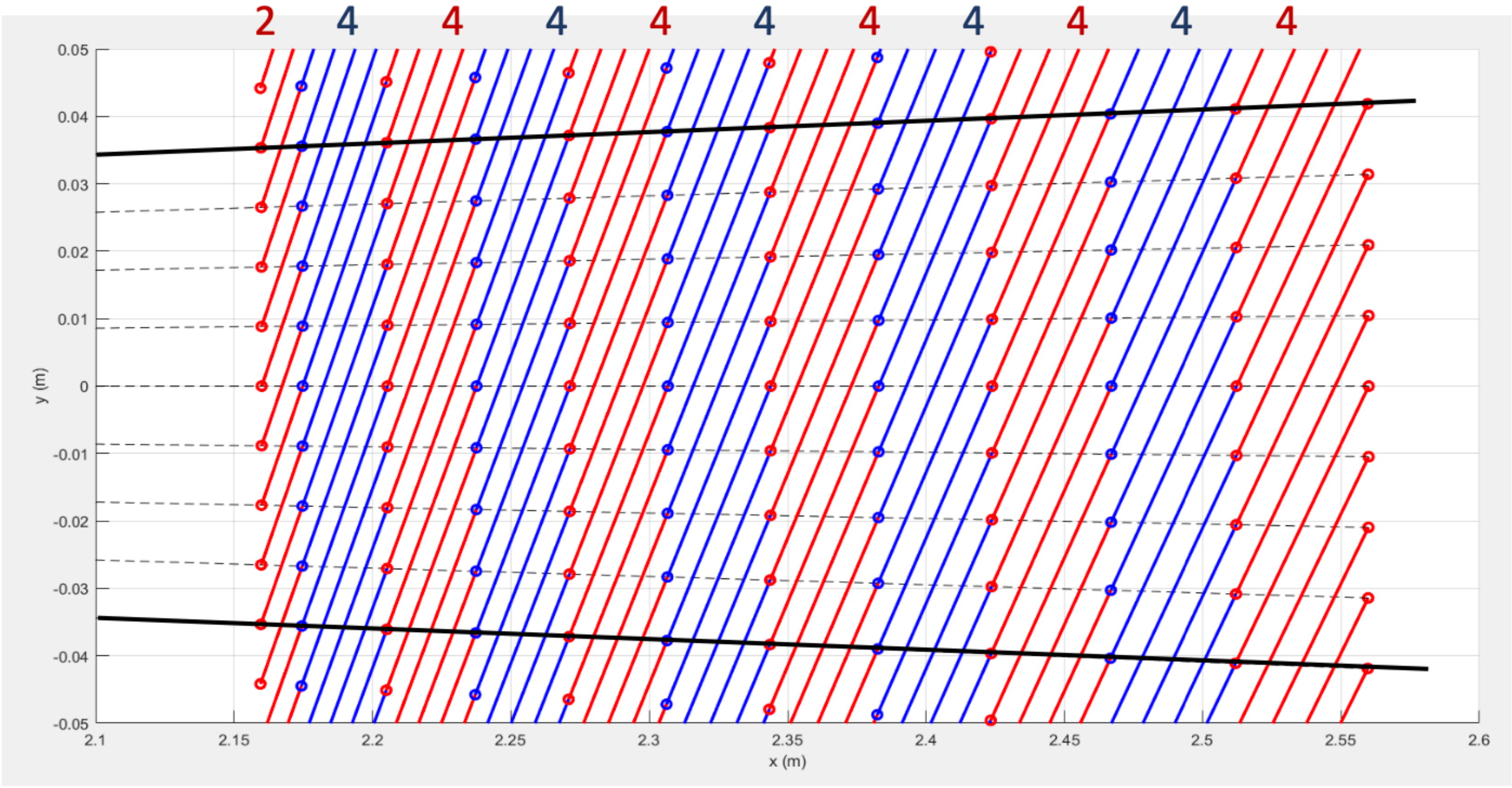
Tong

Very preliminary, results still under scrutiny

Readout with cell corners projective in ϕ

- Current ECAL parameters leads to cell corners that are not projective in ϕ => particles from the IP do not cross the same number of modules per layer depending on phi
- With 11 layers instead of 12, properly segmented longitudinally and slight change of inclination (~0.5º) cells can be aligned





R. Chiche,
D. Fournier,
N. Morange

- Used Nicolas' code to calculate geometrical parameters (angle, layer length) and LAr gap given the active ECAL r_in, r_out in the current full sim and desired number of electrodes (1536), and created new xml version of detector (ALLEGRO_o1_v03)
- Re-ran the simulation with new geometry to calculate all parameters needed for reconstruction (sampling fractions, expected noise and list of neighbours for topoclustering, MVA calibration, …) => PR: k4geo, LAr_scripts: code ready but PR not yet created - waiting for others to be merged)

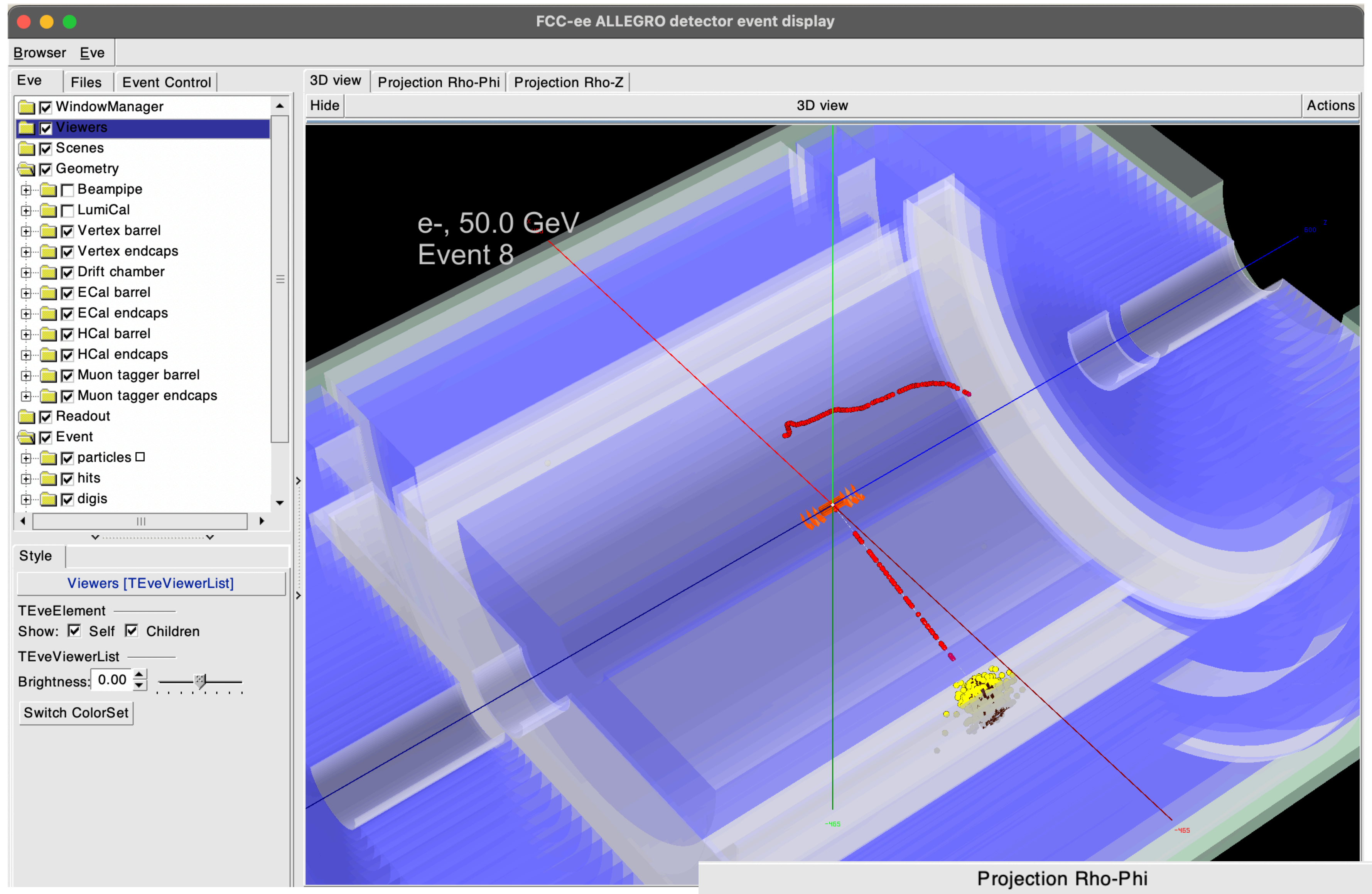
GM

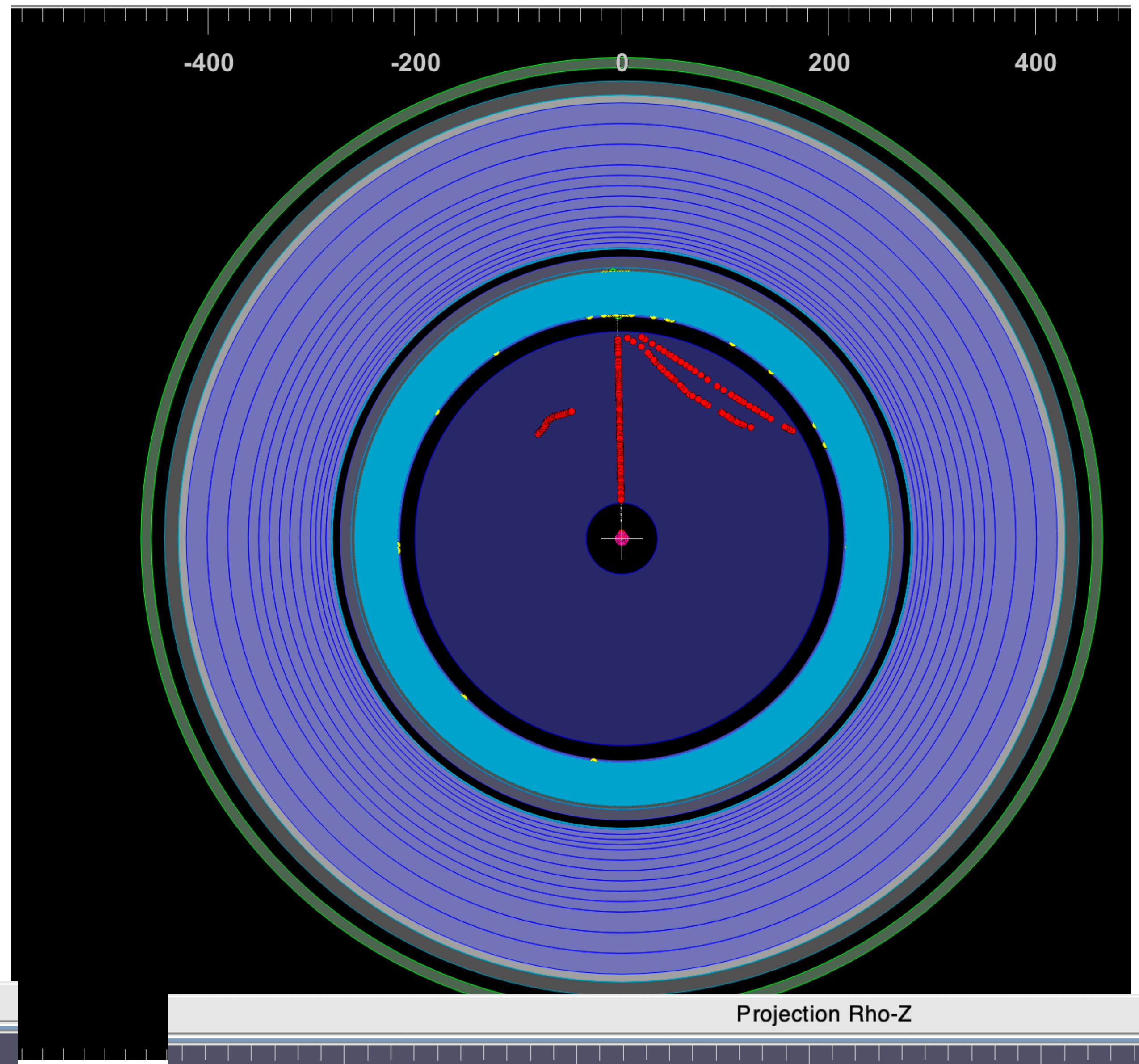
Event display updates

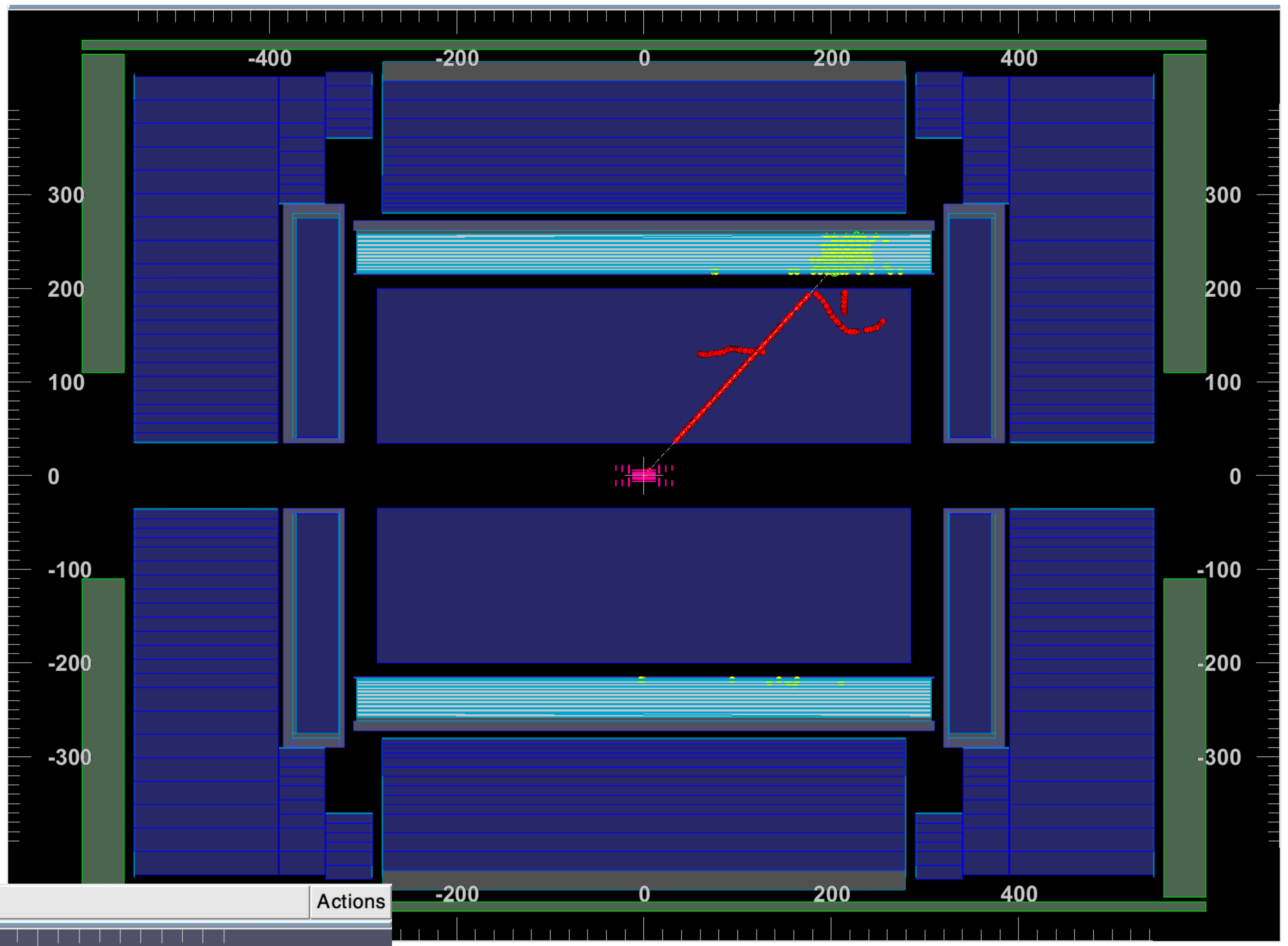
- Better rendering of some volumes (more polygons)
- Included option to show all ALLEGRO sub detectors (including ECAL endcap, HCAL barrel and endcaps, others)
    - Conveniently grouped in main window, with some elements (beam pipe, lumical) off by default
- Can now also draw hits in vertex and drift chamber (in addition to ECAL and HCAL)
- Improved rho-phi and rho-z views (do not show endcaps in rho-phi projection..)
- Make objects to draw, names of collections, cluster energy thresholds run-time options
- Store event display configuration in json file that can be passed at run time as an argument to the main executable
- Event display can work with both full-sim files produced with legacy configuration (gen/sim/digi with k4run) or new config based on ddsim + k4run
- probably some more changes that I forgot

- https://github.com/giovannimarchiori/calodisplay
- Beware: not a “serious” project, mostly for debugging / own education / fun

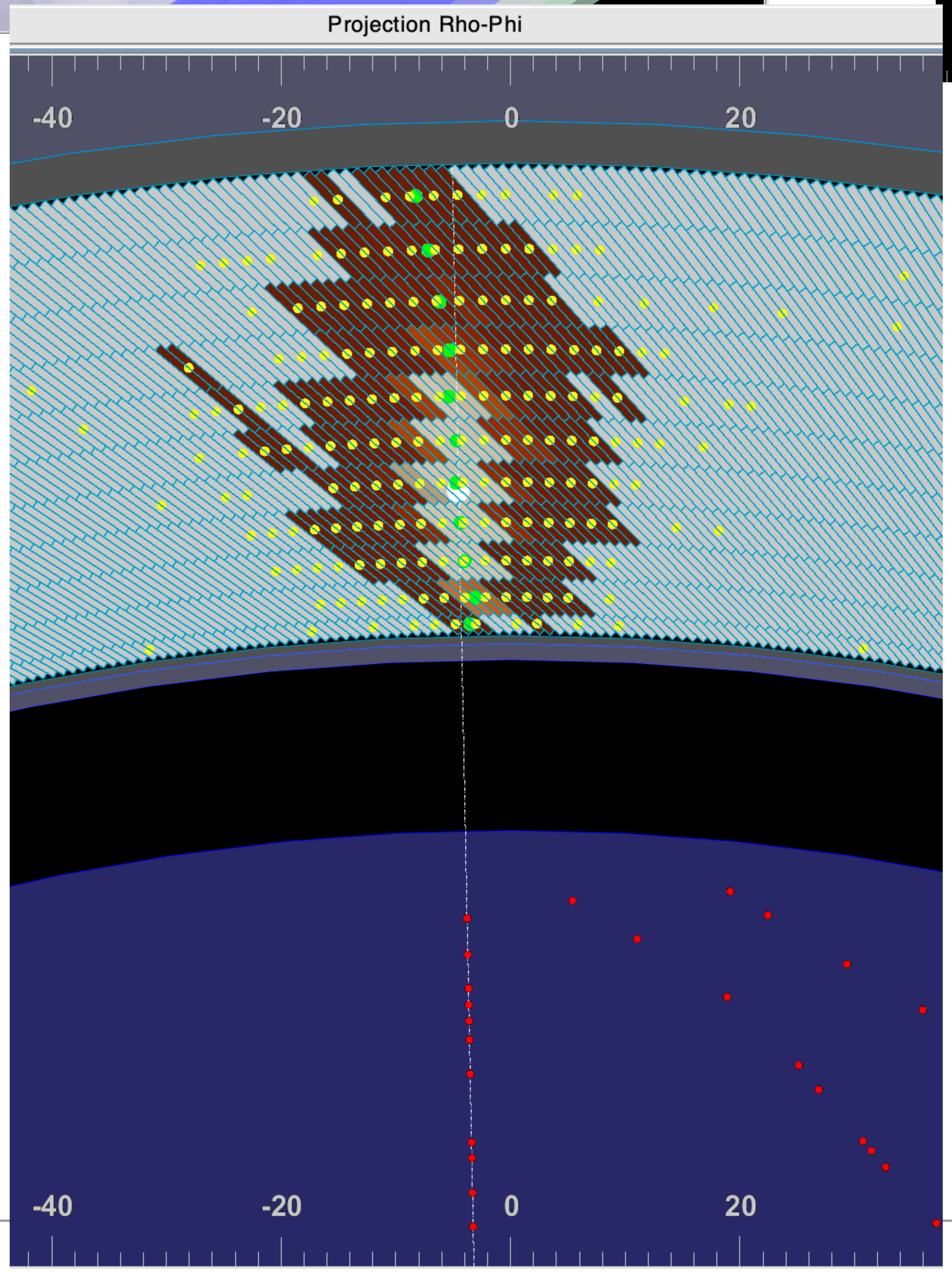
Event display updates

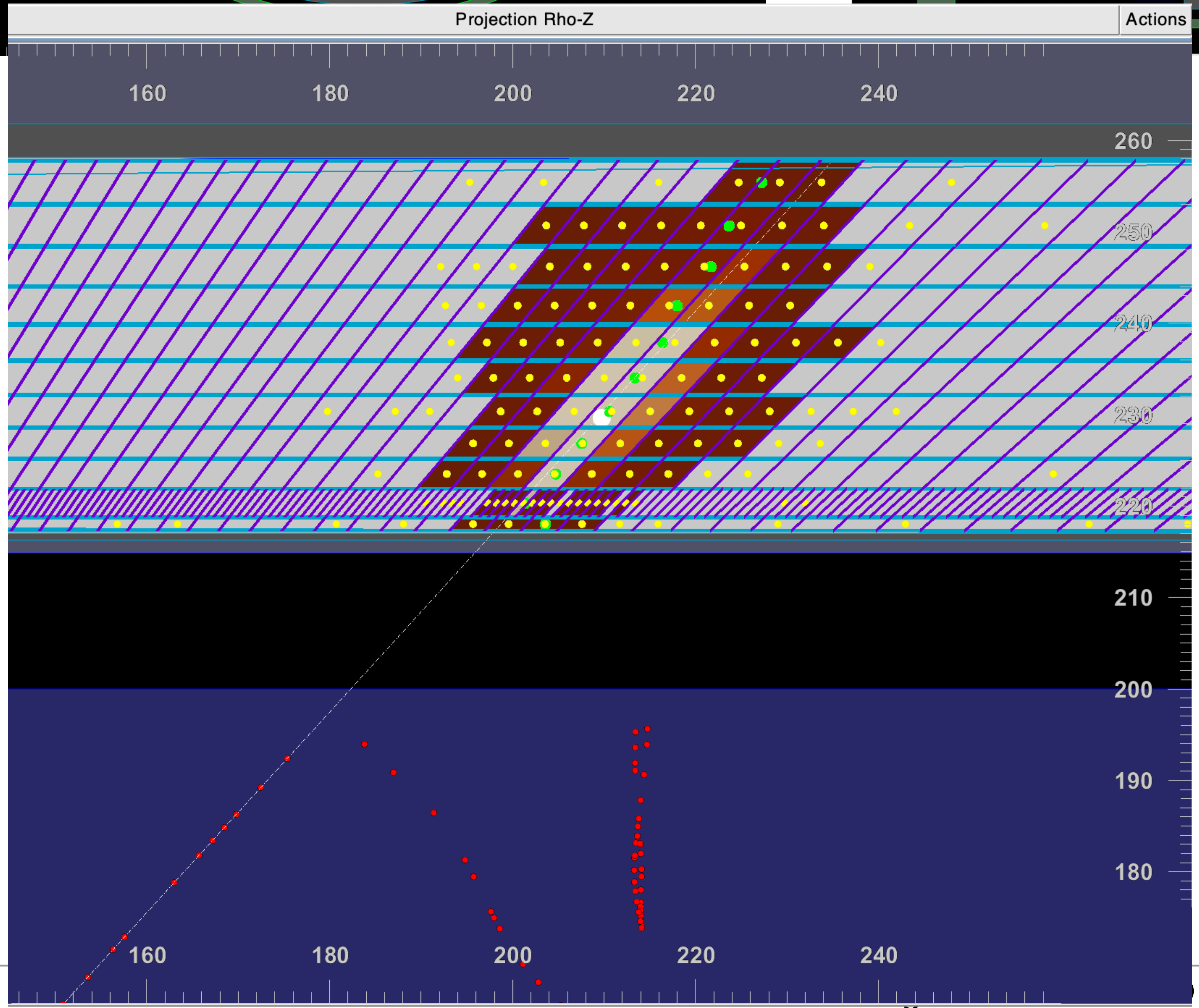






ALLEGRO o1_v03 with projective cell corners





GM

Conclusion

- Continuing progress on full simulation of ALLEGRO ECAL @ Paris
- New detector geometry with 11 layers and aligned cell corners implemented, PR created, hopefully merged soon
  - Need a decision on which geometry to use as reference for our performance studies
- Focused so far mostly on low-level details, but finally starting to have all ingredients to look at high-level performance
  - energy and direction resolution
    - MVA-based calibration can be run inside Gaudi, and can be easily re-trained from ROOT file produced from full-sim
    - plan to determine direction from layer barycentre positions; first step ongoing = study of theta and phi resolution vs E
  - particle ID (eg γ vs π^0)
    - optimisation of input variables and multivariate algorithms/parameters
  - Looking forward to have even more fun with the ALLEGRO ECAL simulation!

- *And many thanks to all the experts that help us so kindly (detector details, code review, ideas/suggestions/feedback) …*