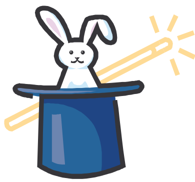
Marlin-based Algorithm for Geometry-Independent Clustering

MAGIC : v01-02



Chris Ainsley

University of Cambridge

<ainsley@hep.phy.cam.ac.uk>

General CALICE meeting

12-13 October 2005, DESY, Hamburg, Germany

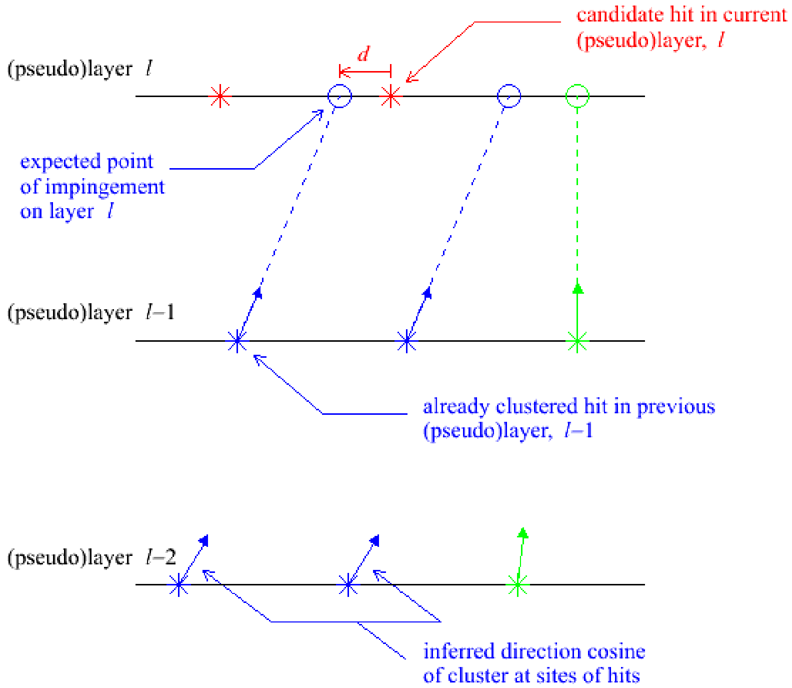
Order of service

- Reminder of the (3-stage) clustering algorithm.
- Where to get the code and how to get started with it.
- Studies of charged/neutral shower separation at normal incidence.
- Studies of cluster reconstruction vs solid angle in full detector simulation.
- Running the algorithm on the Ecal prototype data.
- Summary.

The algorithm and how to use it…

Clustering with MAGIC: stage 1

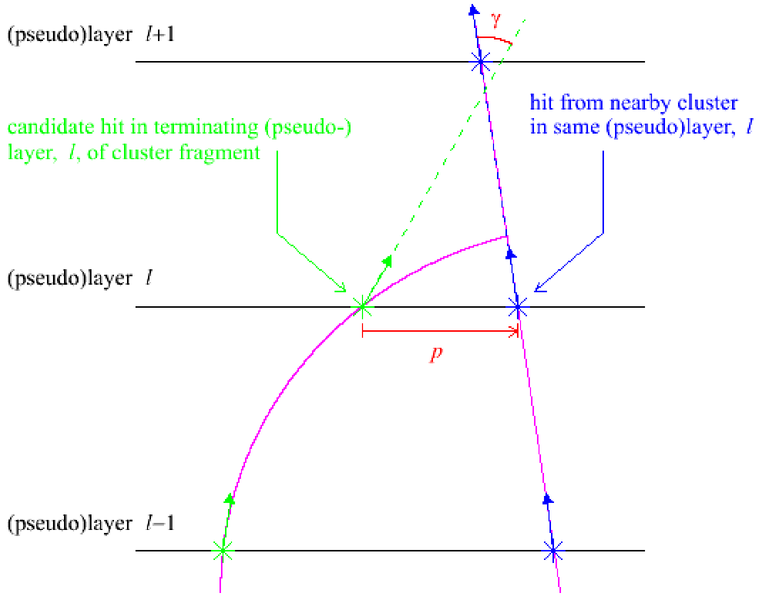
- Form coarse clusters by ***tracking*** closely-related hits ***layer-by-layer*** through the calorimeter:
  - for a candidate hit in a given layer, ***l***, minimise the distance, ***d***, w.r.t all (already clustered) hits in layer ***l***-1;
  - if ***d*** < **distMax** for minimum ***d***, assign candidate hit to same cluster as hit in layer ***l***-1 which yields minimum;
  - if not, repeat with all hits in layer ***l***-2, then, if necessary, layer ***l***-3, *etc.*, right through to layer ***l***-**layersToTrackBack**;
  - after iterating over all hits in layer ***l***, seed new clusters with those still unassigned, grouping those within **proxSeedMax** of hit of highest remaining density into same seed;
  - assign a direction cosine to each layer ***l*** hit:
    - if in Ecal, calculate density-weighted centre of each cluster's hits in layer ***l***; assign a direction cosine to each hit along the line joining its cluster's centre in the seed layer (or (0,0,0) if it's a seed) to its cluster's centre in layer ***l***;
    - if in Hcal, assign a direction cosine to each hit along the line from the hit to which each is linked (or (0,0,0) if it's a seed) to the hit itself;
  - iterate outwards through layers.



Clustering with MAGIC: stage 2

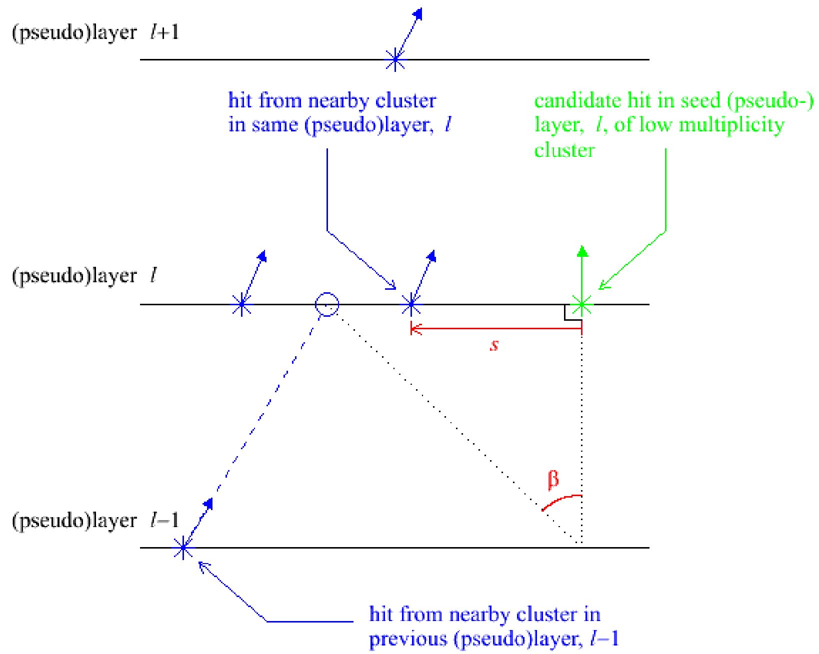
- Try to merge backward-spiralling track-like cluster-fragments with the forward propagating clusters to which they belong:
  - for each hit in the terminating layer, ***l***, of a candidate cluster fragment, calculate the distance, ***p***, to each hit in nearby clusters in the same layer, and the angle, γ, between their direction cosines;
  - loop over all pairs of hits;
  - if, for any pair, both:
    - ***p*** < **proxMergeMax** *and*
    - **cos γ** < **cosGammaMax**

    are satisfied, merge clusters together into one;
  - iterate over clusters.



Clustering with MAGIC: stage 3

- Try to merge low multiplicity cluster "halos" (hit multiplicity < **clusterSizeMin**) which just fail the stage 1 cluster-continuation cuts:
  - for the hit of highest density in the seed layer, ***l***, of a low multiplicity cluster, minimise the angle, β, w.r.t all hits in layer ***l***–1;
  - if **tan** β < **tanBetaMax** for minimum β, merge the clusters containing the repsective hits into one;
  - if not, repeat with all hits in layer ***l***–2, then, if necessary, layer ***l***–3, *etc.*, right through to layer ***l***–**layersToTrackBack**;
  - if still not, repeat above steps with the candidate hit in the seed layer of the low multiplicity cluster of next highest density, *etc.* ;
  - if still not, merge the low multiplicity cluster into the nearest cluster with hits in the same layer as the low multiplicity cluster's seed layer, provided the two clusters contain hits separated by ***s*** < **proxMergeMax**;
  - iterate over clusters.



Code organisation within LCIO/MARLIN

- Code structured as a series of 5+1 **MARLIN** "***processors***", together with a **steering file**: `cluster.steer` (read at *run-time*).
- Reads hits collections from **LCIO** file, adds **LCIO** clusters collections (essentially pointers back to component hits) and writes everything to new **LCIO** output file.
- Processors to do the reconstruction:
  - `CalorimeterConfigurer`
    → allows user to define geometrical layout of calorimeter;
  - `CalorimeterHitSetter`
    → applies hit-energy threshold and adds pseudolayer and pseudostave indices to hits collection (encoded in CellID1 akin to encoding of layer and stave indices in CellID0) as well as hit weights (= local hit density);
  - `CalorimeterStage1Clusterer`
    → performs coarse cluster reconstruction;
  - `CalorimeterStage2Clusterer`
    → recovers backward-spiralling track-like cluster fragments;
  - `CalorimeterStage3Clusterer`
    → recovers low multiplicity cluster fragments.
- Additional processor to access MC truth (if simulation):
  - `CalorimeterTrueClusterer`
    → constructs true clusters, where a true cluster is considered to comprise all hits attributable to either:
    (i) the same generator primary or any of its non-backscattered progeny, or
    (ii) the same backscattered daughter or any of its non-backscattered progeny.

User-controlled steering with MARLIN

- Detector parameters and clustering cuts set in **cluster.steer** (e.g. Mokka D09 model):

```
ProcessorType CalorimeterConfigurer
     detectorType               full       # "full" => barrel+endcaps; "prototype" => layers perp'r to +z
     iPx                        0.         # x-coordinate of interaction point (in mm)
     iPy                        0.         # y-coordinate of interaction point (in mm)
     iPz                        0.         # z-coordinate of interaction point (in mm)
     ecalLayers                 40         # number of Ecal layers
     hcalLayers                 40         # number of Hcal layers
     barrelSymmetry             8          # degree of rotational symmetry of barrel
     phi_1                      90.0       # phi offset of barrel stave 1 w.r.t. x-axis (in deg)
ProcessorType CalorimeterHitSetter
     ecalMip                    0.000150   # Ecal MIP energy (in GeV)
     hcalMip                    0.0000004  # Hcal MIP energy (in GeV)
     ecalMipThreshold           0.3333333  # Ecal hit-energy threshold (in MIP units)
     hcalMipThreshold           0.3333333  # Hcal hit-energy threshold (in MIP units)
ProcessorType CalorimeterStage1Clusterer
     layersToTrackBack_ecal     3          # number of layers to track back in Ecal
     layersToTrackBack_hcal     3          # number of layers to track back in Hcal
     distMax_ecal               20.0       # distance cut in Ecal (in mm)
     distMax_hcal               30.0       # distance cut in Hcal (in mm)
     proxSeedMax_ecal           14.0       # maximum cluster-seed radius in Ecal (in mm)
     proxSeedMax_hcal           50.0       # maximum cluster-seed radius in Hcal (in mm)
ProcessorType CalorimeterStage2Clusterer
     proxMergeMax_ecal          20.0       # Ecal proximity cut for cluster merging (in mm)
     proxMergeMax_hcal          30.0       # Hcal proximity cut for cluster merging (in mm)
     cosGammaMax                0.5        # angular cut for cluster merging
ProcessorType CalorimeterStage3Clusterer
     clusterSizeMin             10         # minimum cluster size to avert potential merging
     layersToTrackBack_ecal     39         # number of layers to track back in Ecal for merging
     layersToTrackBack_hcal     79         # number of layers to track back in Hcal for merging
     tanBetaMax                 6.0        # angular cut for cluster merging
     proxSeedMax_ecal           400.0      # Ecal proximity cut for cluster merging (in mm)
     proxSeedMax_hcal           400.0      # Hcal proximity cut for cluster merging (in mm)
```

Getting started with MAGIC

- Install **LCIO** (≥ v01-05) and **MARLIN** (≥ v00-07).
- Download *MAGIC* tar-ball from
  http://www.hep.phy.cam.ac.uk/~ainsley/MAGIC/MAGIC-v01-02.tar.gz
- **Two directories** and a **README** file (read this first!).
- The **`clustering` directory** contains the cluster-reconstruction (and cluster-truth) code (i.e. all processors and steering file mentioned earlier).
- Takes **.slcio input** files containing **`CalorimeterHits`** (data) or **`SimCalorimeterHits`** (MC):
  - must be generated with hit-positions stored, *i.e.* **`RCHBIT_LONG=1`** (data) or **`CHBIT_LONG=1`** (MC);
  - collection names must contain the string "**`ecal`**" or "**`hcal`**" (in upper or lower case, or in some combination of these) to identify the type of hit (for energy-threshold application).
- Produces **.slcio output** file with cluster-related collections added:
  - **`CalorimeterHits`** ⇒ hits above energy threshold;
  - **`CalorimeterHitRelationsToSimCalorimeterHits`** (**MC only**) ⇒ pointers to original simulated hits;
  - **`CalorimeterStage1Clusters`** ⇒ clusters after stage 1 of algorithm;
  - **`CalorimeterStage2Clusters`** ⇒ clusters after stage 2 of algorithm;
  - **`CalorimeterStage3Clusters`** ⇒ clusters after stage 3 of algorithm;
  - **`CalorimeterTrueClusters`** (**MC only**) ⇒ true clusters;
  - **`CalorimeterTrueClusterRelationsToMCParticles`** (**MC only**) ⇒ pointers to original MC particles.
- The **`examples` directory** contains example analysis code which performs simple manipulations with the clusters (e.g. processors which add calibrated energies to clusters, produce the plots shown earlier, calculate the reconstruction quality… and an accompanying steering file).

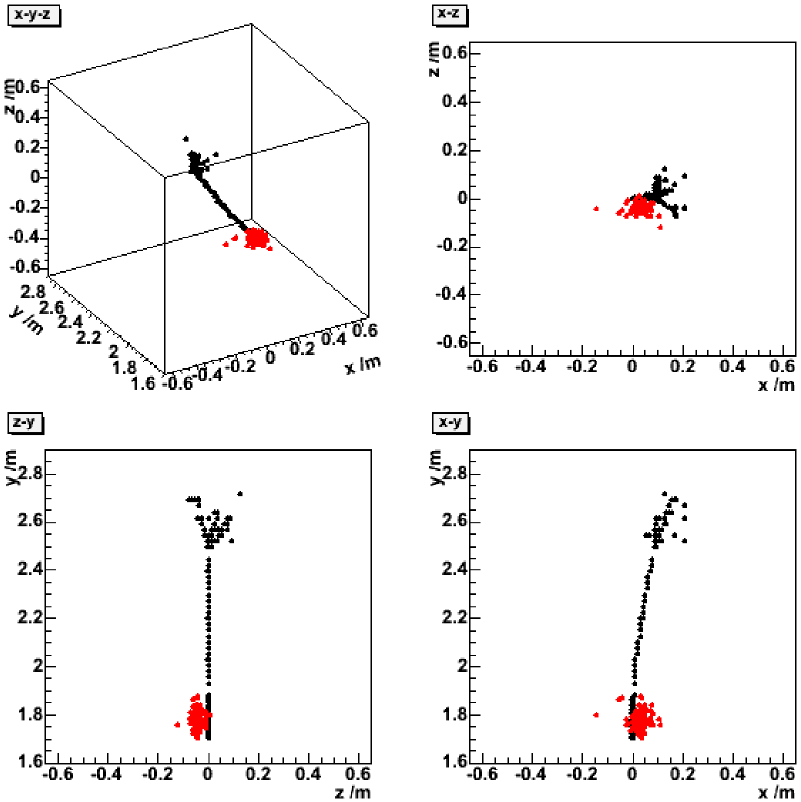
Charged/neutral shower separation...

Charged/neutral shower separation studies

- Fire nearby charged/neutral particles into calorimeter.
- Perform standalone clustering on calorimeter hits with *MAGIC*.
- Extrapolate helix from charged track through calorimeters.
- Associate clusters/cluster fragments with charged particle if seeded within pad-size (= 1 cm) of projected helical trajectory.
- Remove corresponding calorimeter hits from further consideration; assume remainder to be the neutral shower.
- Apply energy calibration to leftover hits to reconstruct neutral particle energy.

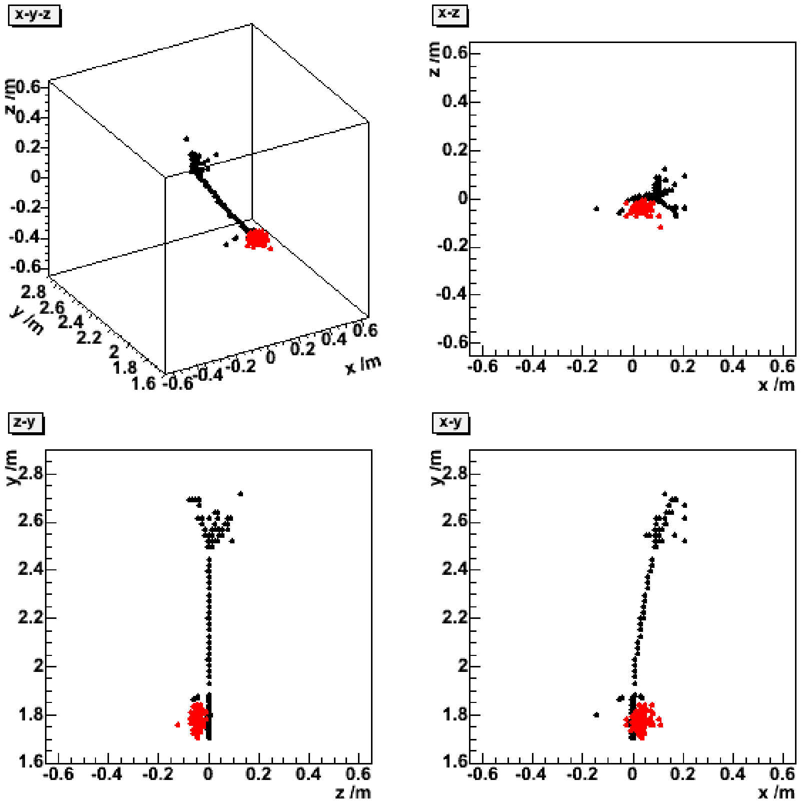
π^+/γ separation: D09 model (1)

True clusters



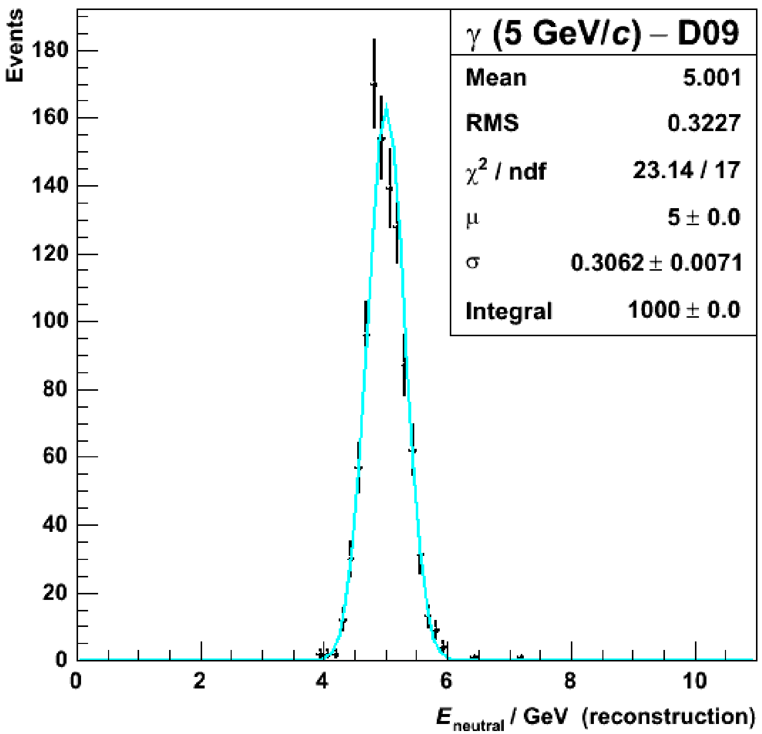
- **Black** cluster = 5 GeV/*c* π^+.
- **Red** cluster = 5 GeV/*c* γ.

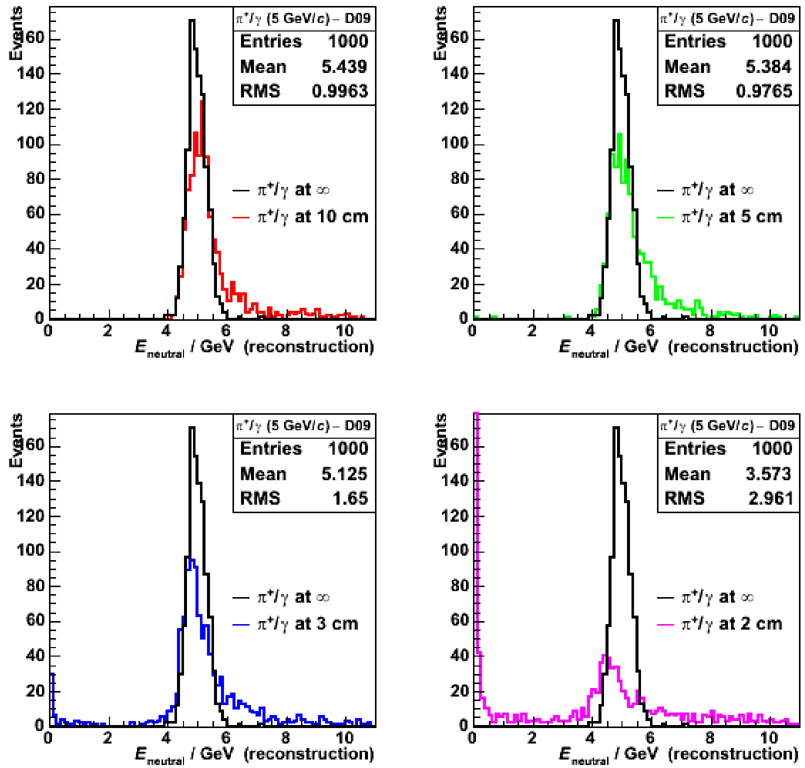
Reconstructed clusters



- **Black** cluster matched to charged track.
- **Red** cluster left over as neutral $\Rightarrow \gamma$ energy well reconstructed.

π^+/γ separation: D09 model (2)



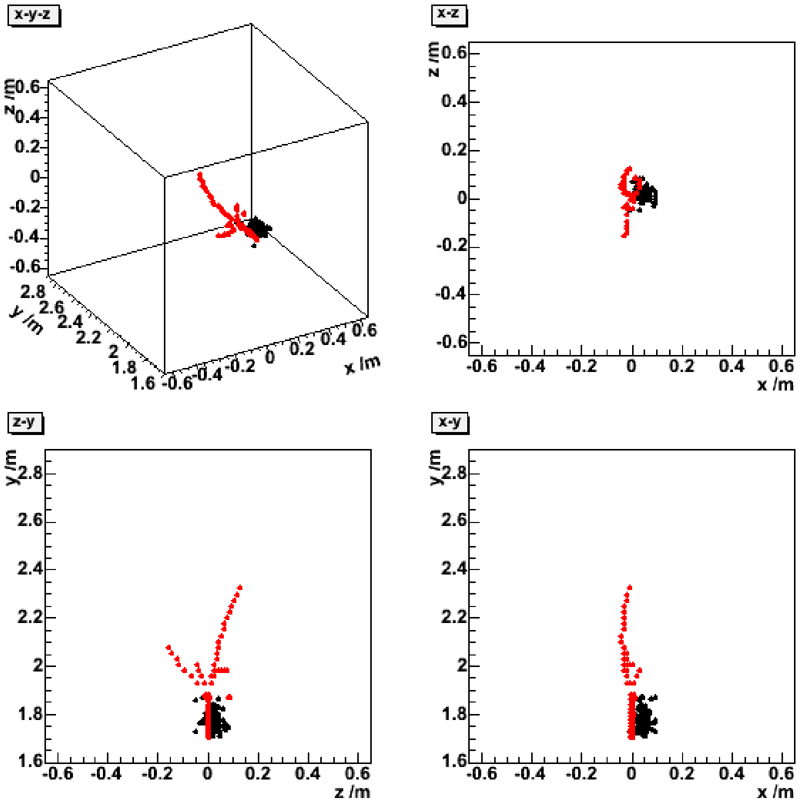


- 1k single γ at 5 GeV/*c*.
- Fit Gaussian to energy distribution, calibrated according to:
  $E = \alpha[(E_{\text{Ecal; 1-30}} + 3E_{\text{Ecal; 31-40}})/E_{\text{Ecal mip}} + 20N_{\text{Hcal}}]$.
- Fix factors α, 20 by minimising χ^2/dof.
- $\sigma/\sqrt{\mu}$ ~ 14% $\sqrt{\text{GeV}}$.

- 1k γ with nearby π^+ (at 10, 5, 3, 2 cm from γ).
- Peak of photon energy spectrum well reconstructed; improves with separation.
- Tail at higher $E \rightarrow$ inefficiency in π^+ reconstruction (next page…).
- Spike at $E = 0$ below 3 cm $\rightarrow$ clusters not distinguished.

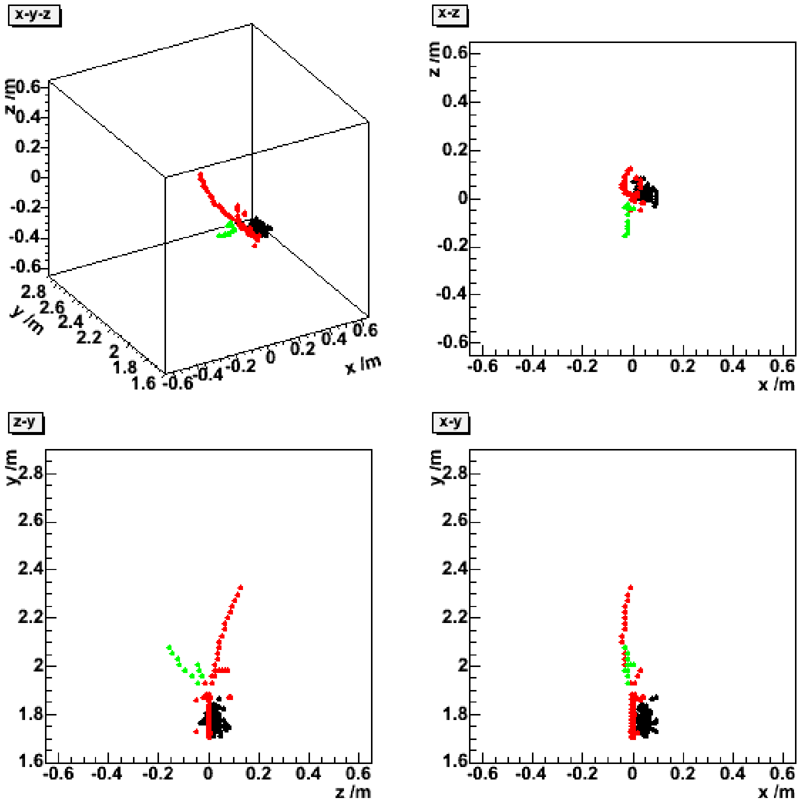
π^+/γ separation: D09 model (3)

True clusters



- **Red** cluster = 5 GeV/*c* π^+.
- **Black** cluster = 5 GeV/*c* γ.

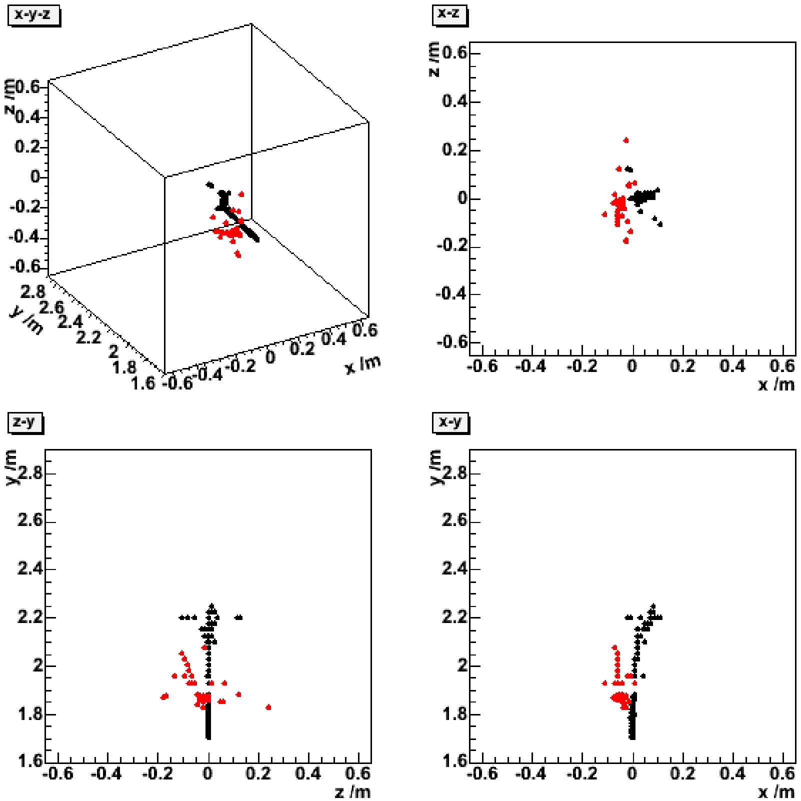
Reconstructed clusters



- **Red** cluster matched to charged track.
- **Black** ***and*** **green** clusters left over as neutral $\Rightarrow$ γ energy overestimated.

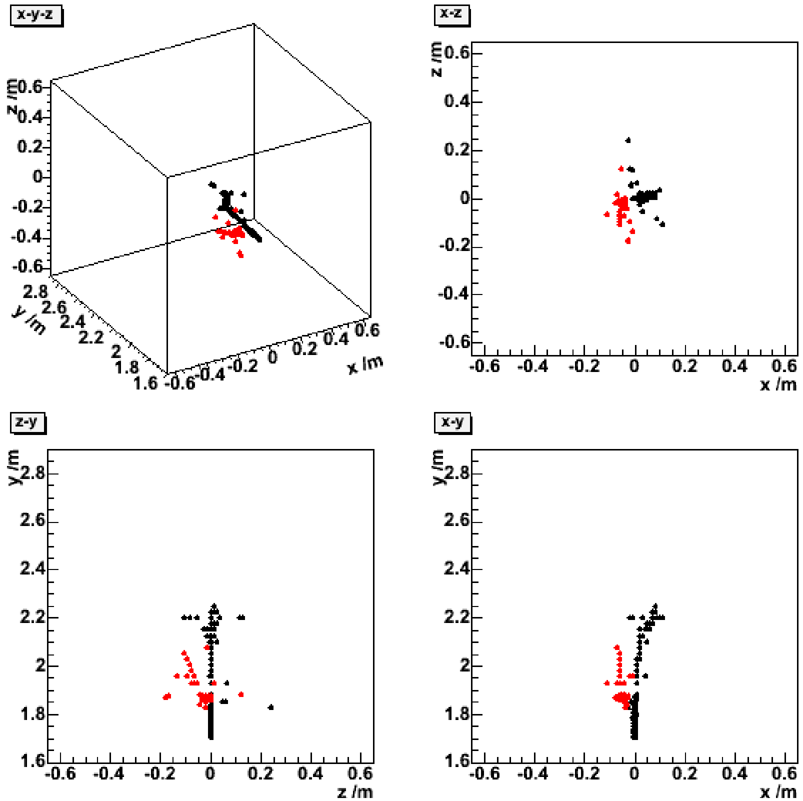
π^+/n separation: D09 model (1)

True clusters



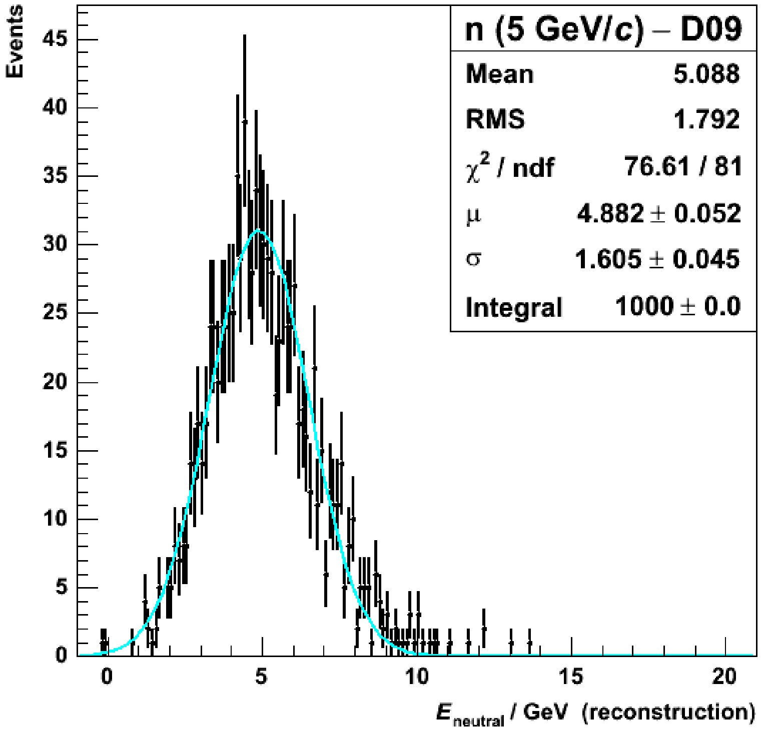
- **Black** cluster = 5 GeV/*c* π^+.
- **Red** cluster = 5 GeV/*c* **n**.

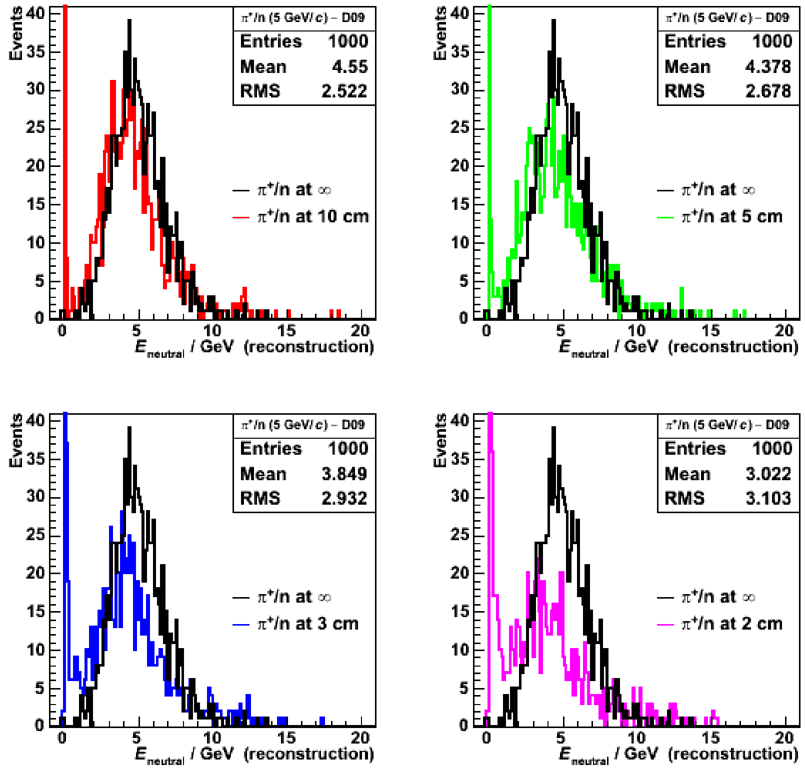
Reconstructed clusters



- **Black** cluster matched to charged track.
- **Red** cluster left over as neutral ⇒ n energy well reconstructed.

π^+/n separation: D09 model (2)



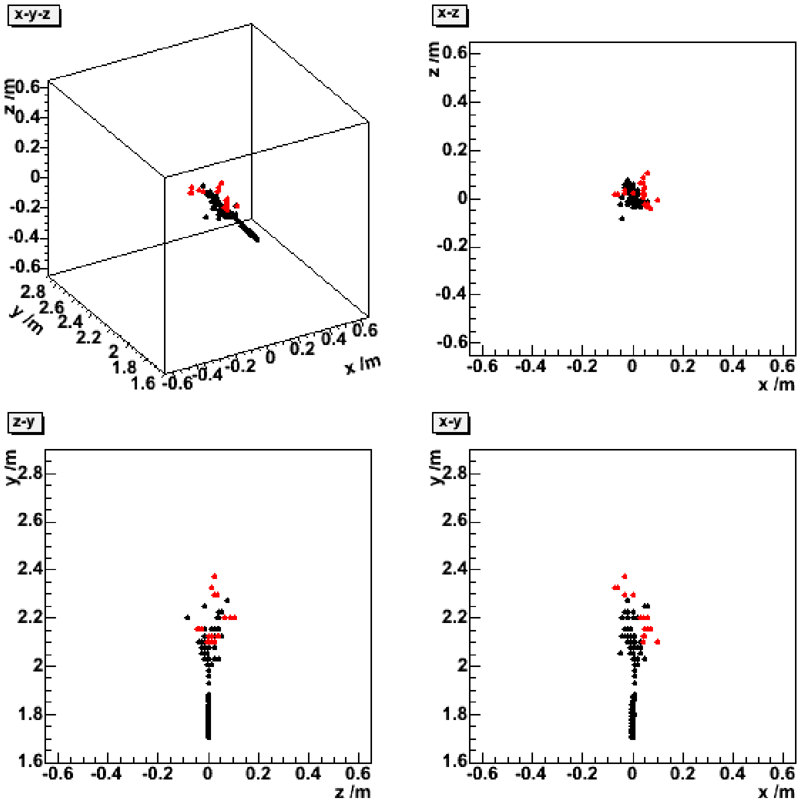


- 1k single **n** at 5 GeV/*c*.
- Fit Gaussian to energy distribution, calibrated according to:
  $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\,mip}} + 20N_{\mathrm{Hcal}}]$.
- Fix factors α, 20 by minimising χ^2/dof.
- $\sigma/\sqrt{\mu} \sim 73\%\ \sqrt{\mathrm{GeV}}$.

- 1k **n** with nearby π^+ (at 10, 5, 3, 2 cm from n).
- Peak of neutron energy spectrum well reconstructed; improves with separation.
- Spike at $E = 0$ even at 10 cm → clusters not distinguished (next page…).

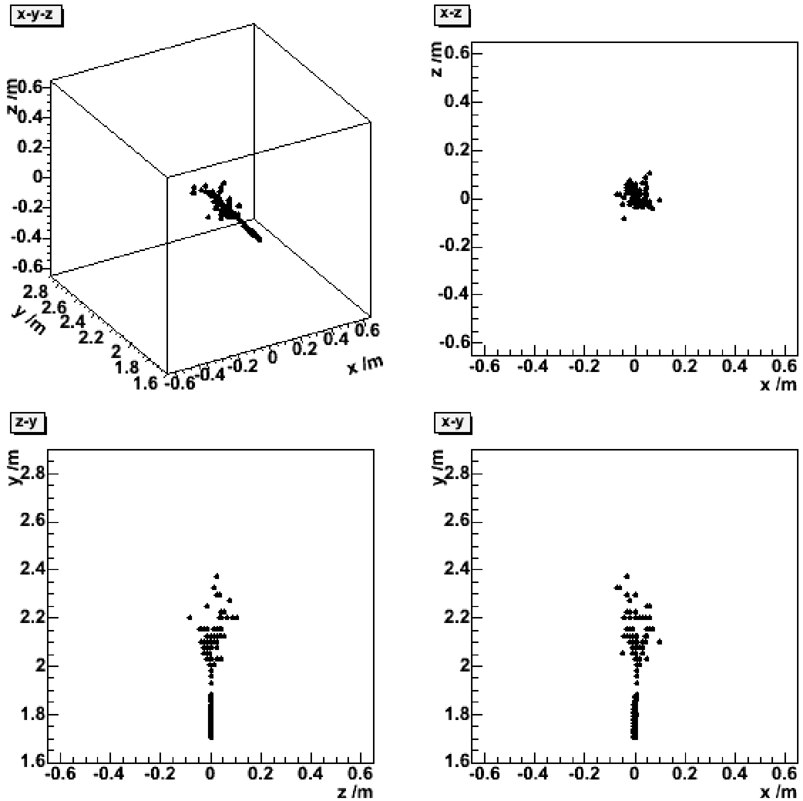
π^+/n separation: D09 model (3)

True clusters



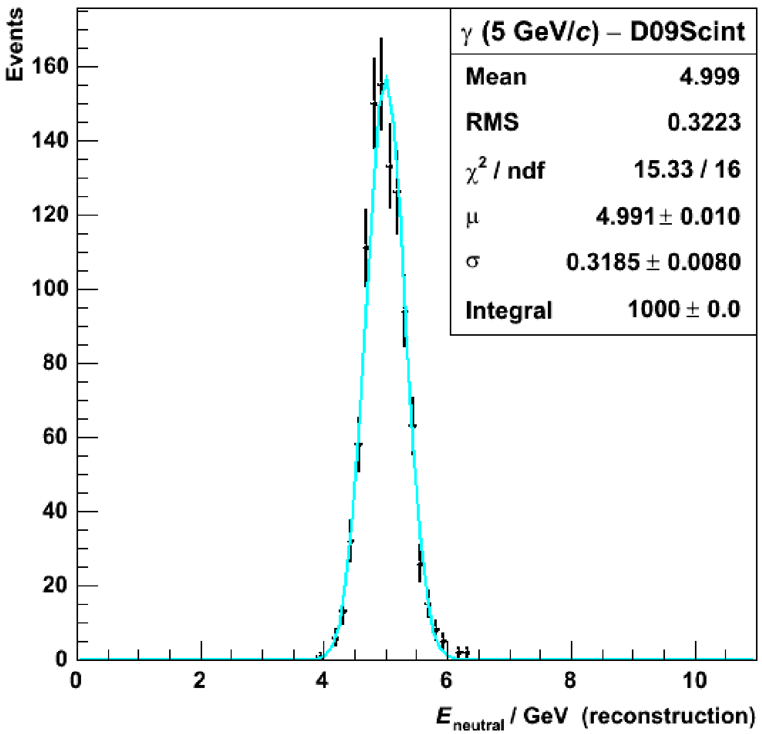
- **Black** cluster = 5 GeV/*c* π^+.
- **Red** cluster = 5 GeV/*c* **n**.

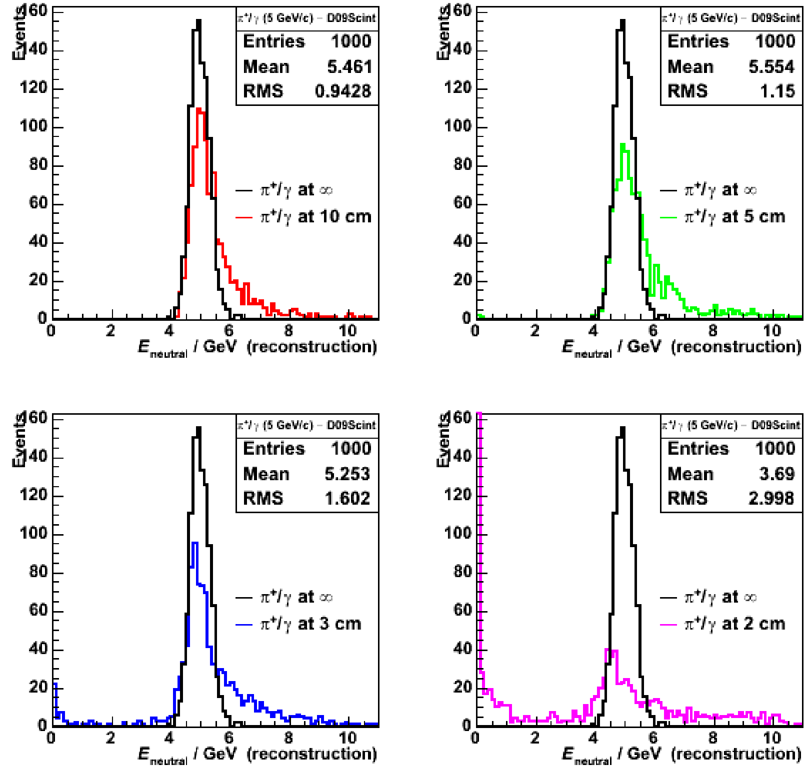
Reconstructed clusters



- **Black** cluster matched to charged track.
- Nothing left over as neutral ⇒ n not reconstructed (*i.e.* $E = 0$).

π^+/γ separation: D09Scint model



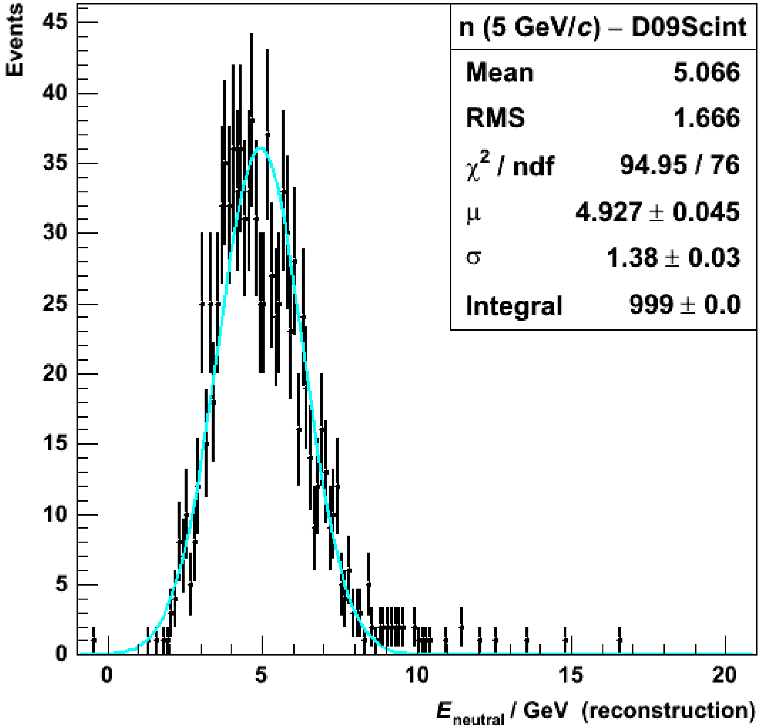


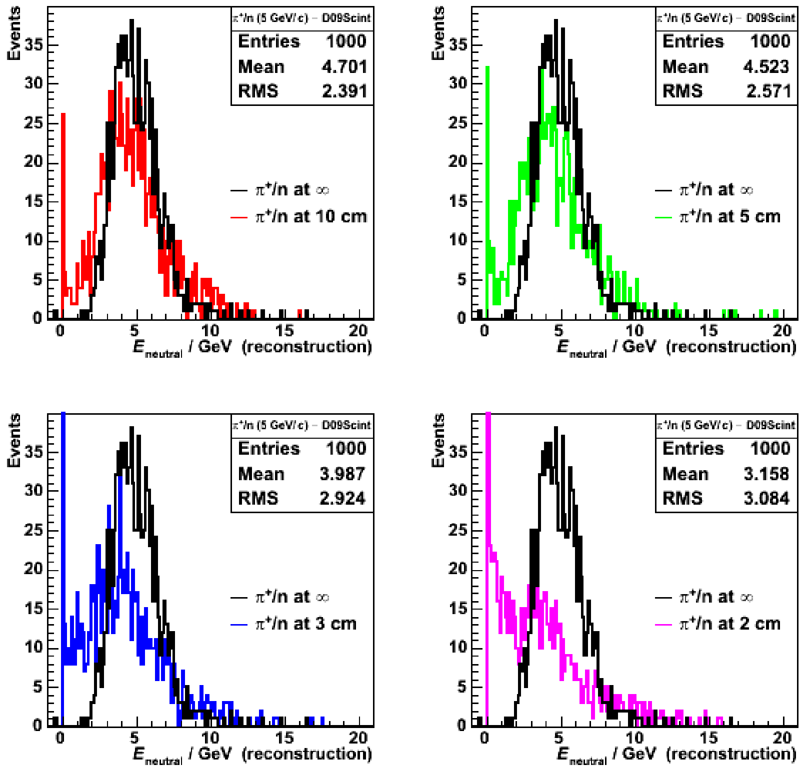
- 1k single γ at 5 GeV/*c*.
- Fit Gaussian to energy distribution, calibrated according to:
  $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\,mip}} + 5E_{\mathrm{Hcal}}/E_{\mathrm{Hcal\,mip}}]$.
- Fix factors α, **5** by minimising χ^2/dof.
- $\sigma/\sqrt{\mu} \sim 14\% \sqrt{\mathrm{GeV}}$ (as for D09 model).

- 1k γ with nearby π^+ (at 10, 5, 3, 2 cm from γ).
- General trends much as for D09 model.

Chris Ainsley
<ainsley@hep.phy.cam.ac.uk>

π^+/n separation: D09Scint model



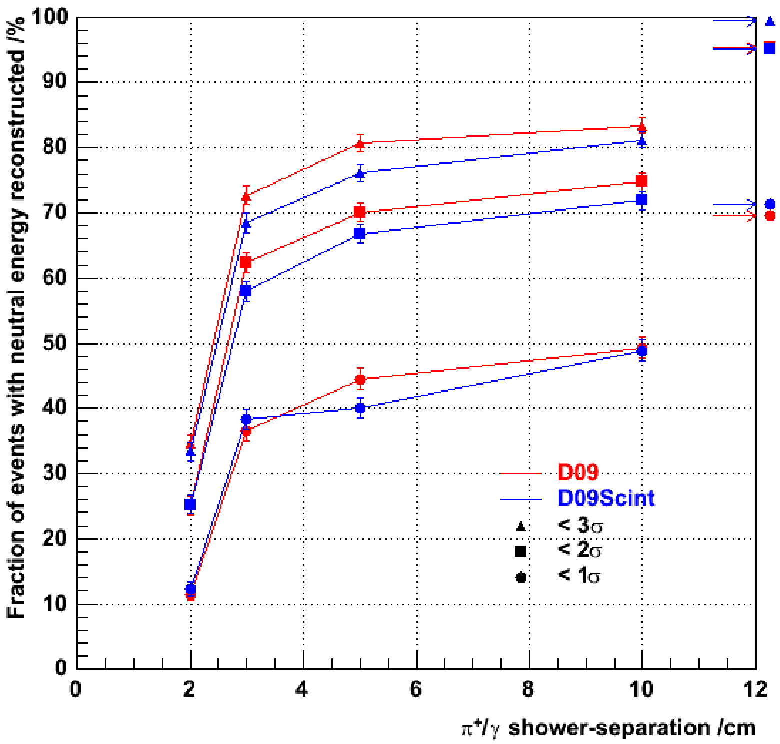


- 1k single **n** at 5 GeV/*c*.
- Fit Gaussian to energy distribution, calibrated according to:
  $E = \alpha[(E_{\text{Ecal; 1-30}} + 3E_{\text{Ecal; 31-40}})/E_{\text{Ecal mip}} + \mathbf{5}E_{\text{Hcal}}/E_{\text{Hcal mip}}]$.
- Fix factors α, **5** by minimising χ^2/dof.
- $\sigma/\sqrt{\mu}$ ~ 62% √GeV (cf. 73% √GeV for D09 model).

- 1k **n** with nearby π^+ (at 10, 5, 3, 2 cm from n).
- General trends much as for D09 model.

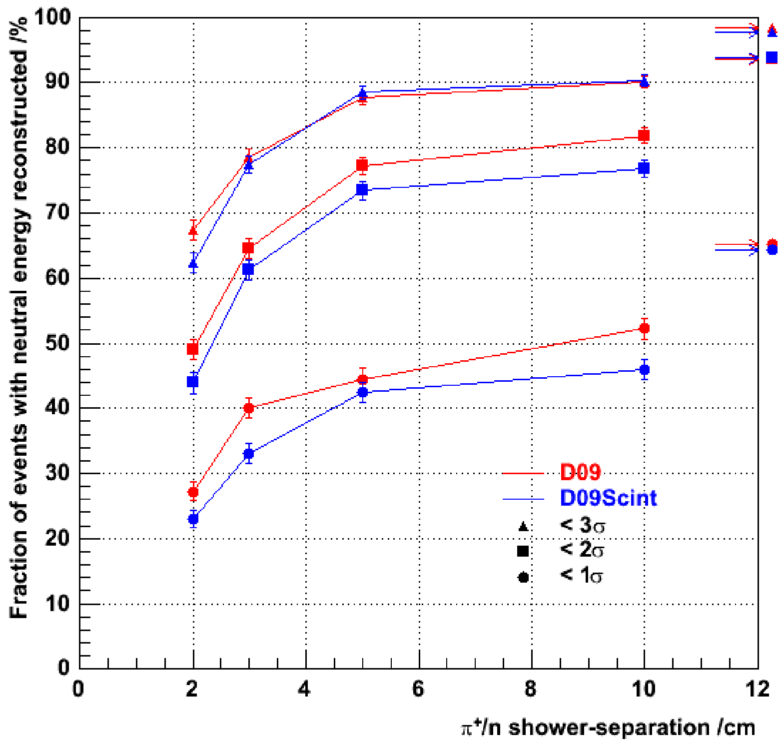
π^+/neutral cluster separability vs separation

5 GeV/c π^+/γ



- Fraction of events with photon energy reconstructed within 1,2,3σ generally higher for D09 than for D09Scint…
- …and absolute γ resolution similar.

5 GeV/c π^+/n

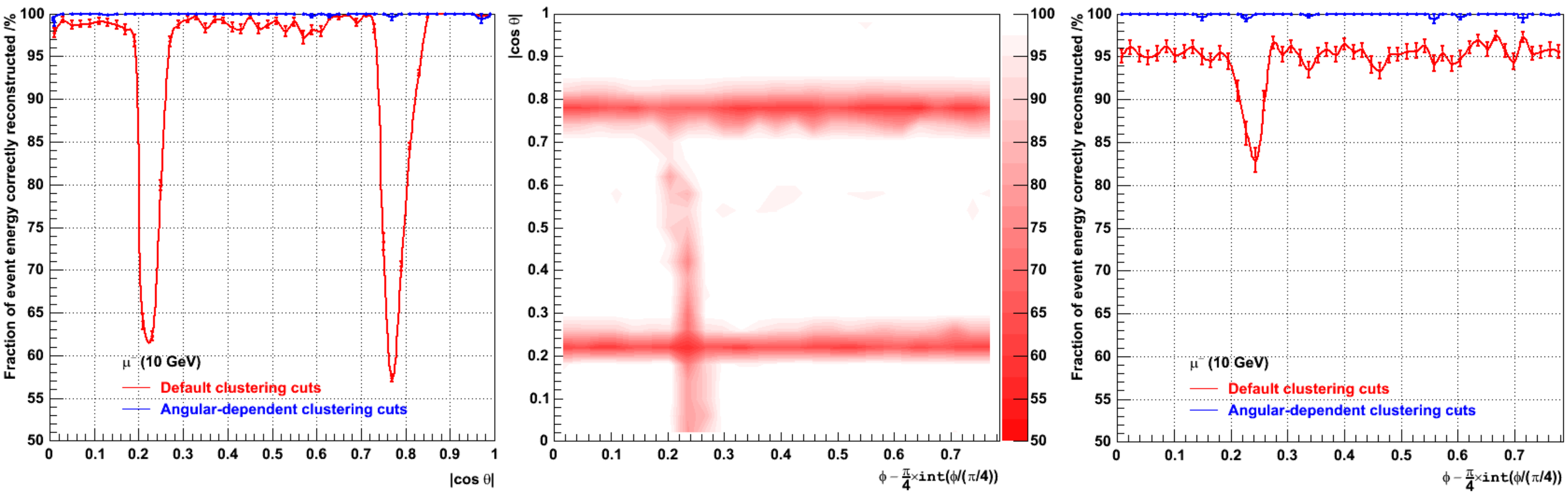


- Fraction with neutron energy reconstructed within 1,2,3σ also generally higher for D09…
- …*but*, absolute ***n*** resolution is better for D09Scint.

Clustering vs detector solid angle...

Detector scan: μ^- (10 GeV)

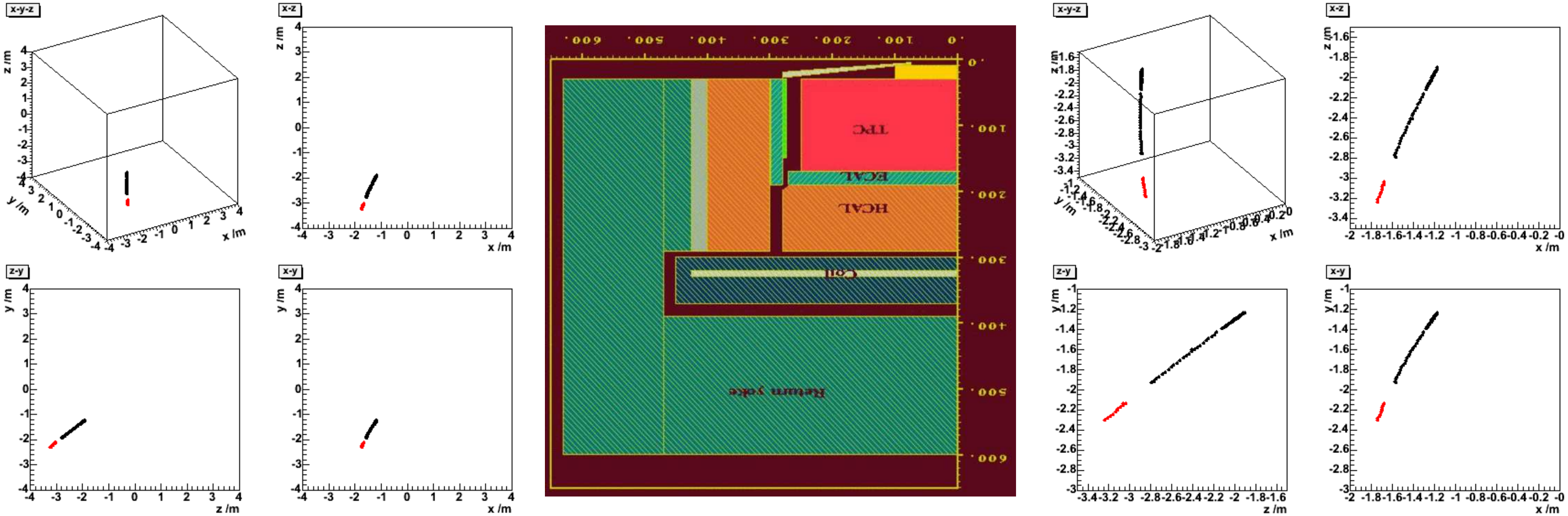
- μ^- fired isotropically into (analogue) **Si/W Ecal**, (digital) **rpc/Fe Hcal** (Mokka "D09" model).
- Cluster energies calibrated according to: $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\ mip}} + 20N_{\mathrm{Hcal}}]$ GeV.
- Fraction of event energy in highest-energy reconstructed cluster plotted vs $|\cos\theta|$ and vs ϕ (folded into first octant: $0 \le \phi < \pi/4$) at (0,0,0).



- Default clustering cuts $\rightarrow$ μ^- track fragmented at $|\cos\theta| \sim 0.78$, $|\cos\theta| \sim 0.23$ and $\phi \sim 0.20$–0.24 ($\cos\theta$–dependent) $\Rightarrow$ algorithm needs to know some geometry to overcome this!
- Angular-dependent clustering cuts $\rightarrow$ μ^- track reconstructed with ~100% efficiency $\forall$ (θ, ϕ).
- What detector features do these regions correspond to?

Detector scan: μ^- (10 GeV)

- μ^- at **cos θ = −0.75** traverses Ecal barrel, Hcal barrel and Hcal endcap.
- Track breaks on crossing **from barrel to endcap** → layers of active material "missing" in the gap.

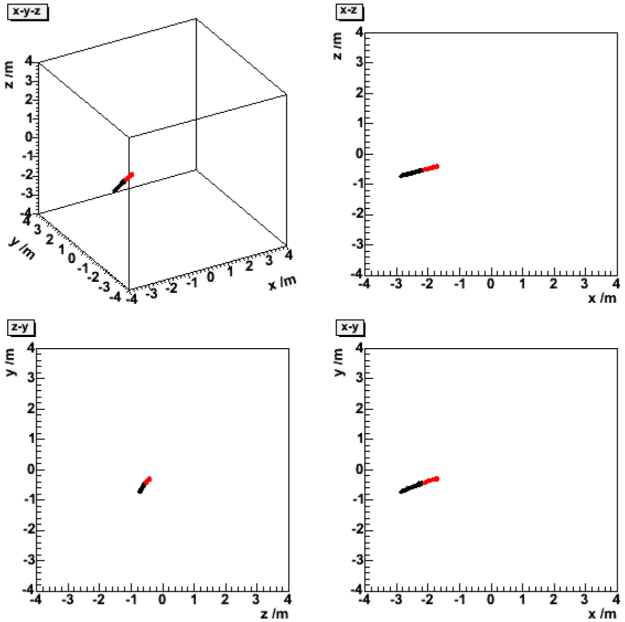


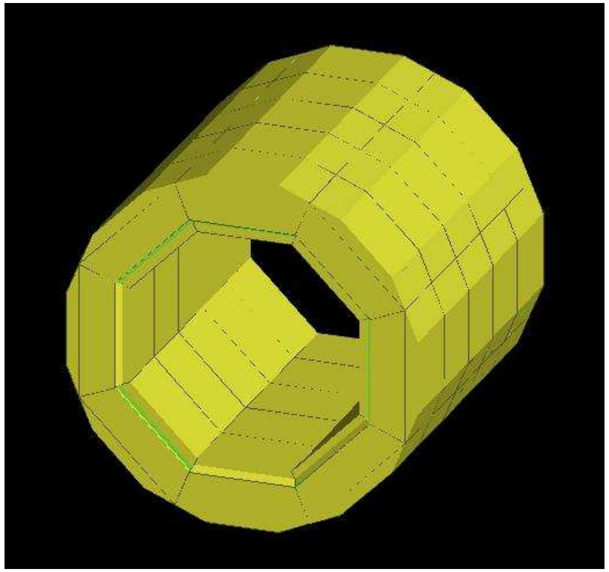
- Relax `layersToTrackBack_ecal` cut for 0.81 < |cos θ| < 0.85 and `layersToTrackBack_hcal` cut for 0.72 < |cos θ| < 0.85 to prevent this.
- Design the detector with as small a barrel-endcap gap as possible!

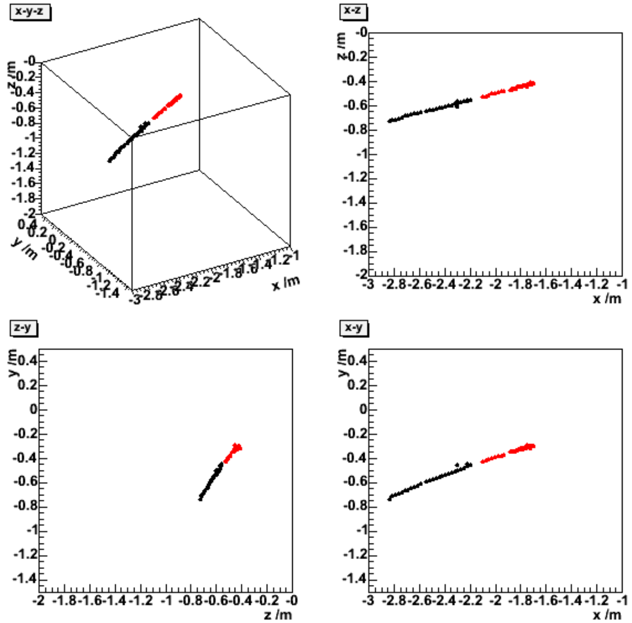
Chris Ainsley
<ainsley@hep.phy.cam.ac.uk>

Detector scan: μ⁻ (10 GeV)

- μ^- at **cos θ = −0.24** traverses Ecal barrel module 3, Hcal barrel module 3 and Hcal barrel module 2.
- Track breaks on crossing **between barrel modules** at |z| ~ 0.56 m → active cells "missing" near the module edges.



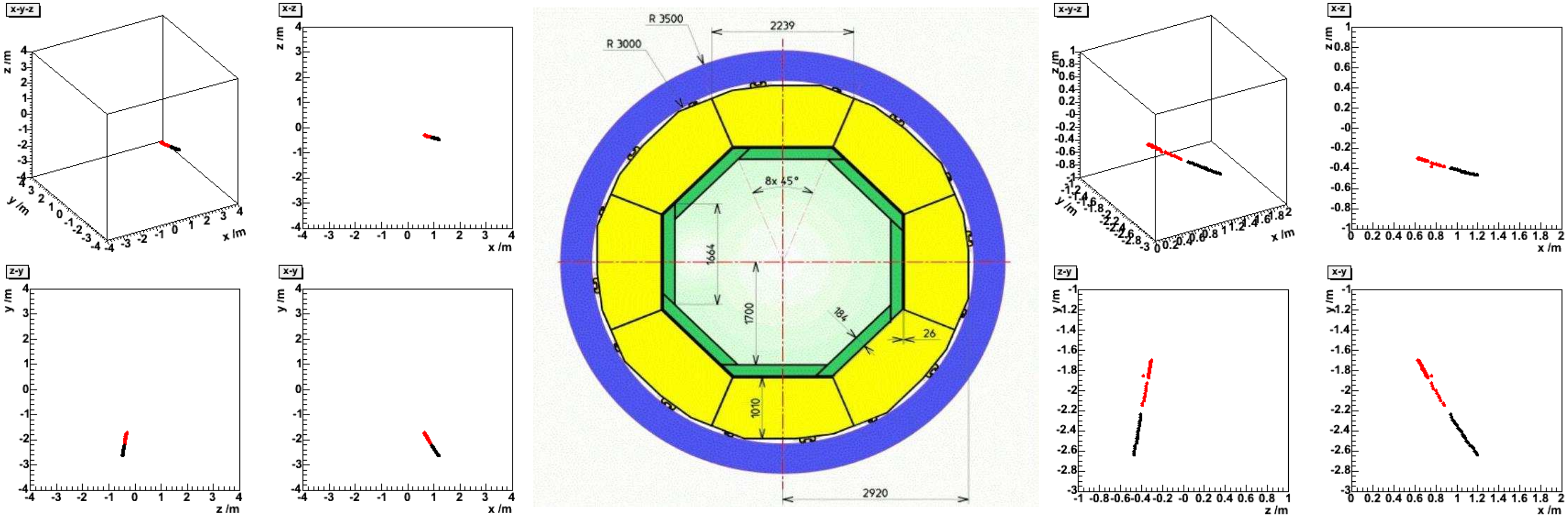




- Relax **distMax_ecal** and **distMax_hcal** cuts for 0.18 < |cos θ| < 0.28 to prevent this.
- Much less severe, but similar, effect at |z| ~ 1.68 m (0.47 < cos θ < 0.65) treated in the same way.
- Design the detector with as small an inter-module gap as possible!

Detector scan: μ^- (10 GeV)

- μ^- at ϕ = **1.58**π traverses Ecal barrel stave 5, Hcal barrel stave 5 and Hcal barrel stave 6.
- Track breaks on crossing **between Hcal barrel staves** at ϕ-($6\times\pi/4$) = $\pi/8$ = 0.39 (curves in B-field) $\rightarrow$ active cells "missing" near the stave edges.

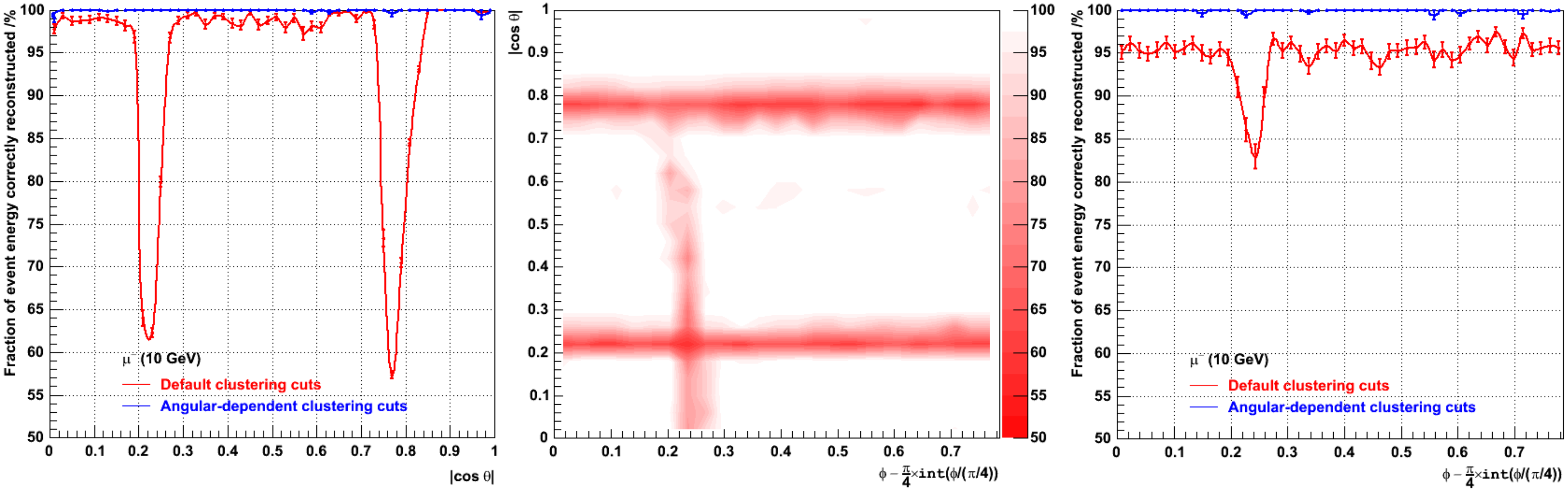


- Relax **distMax_hcal** and **layersToTrackBack_hcal** cuts for 0.36 < ϕ < 0.42 if $|\cos\theta|$ < 0.82 (Hcal barrel) to prevent this.
- No problem in Ecal $\rightarrow$ staves overlap.
- Design the Hcal with no pointing cracks (*e.g.* like the Ecal)!

Chris Ainsley
<ainsley@hep.phy.cam.ac.uk>

Detector scan: μ⁻ (10 GeV)

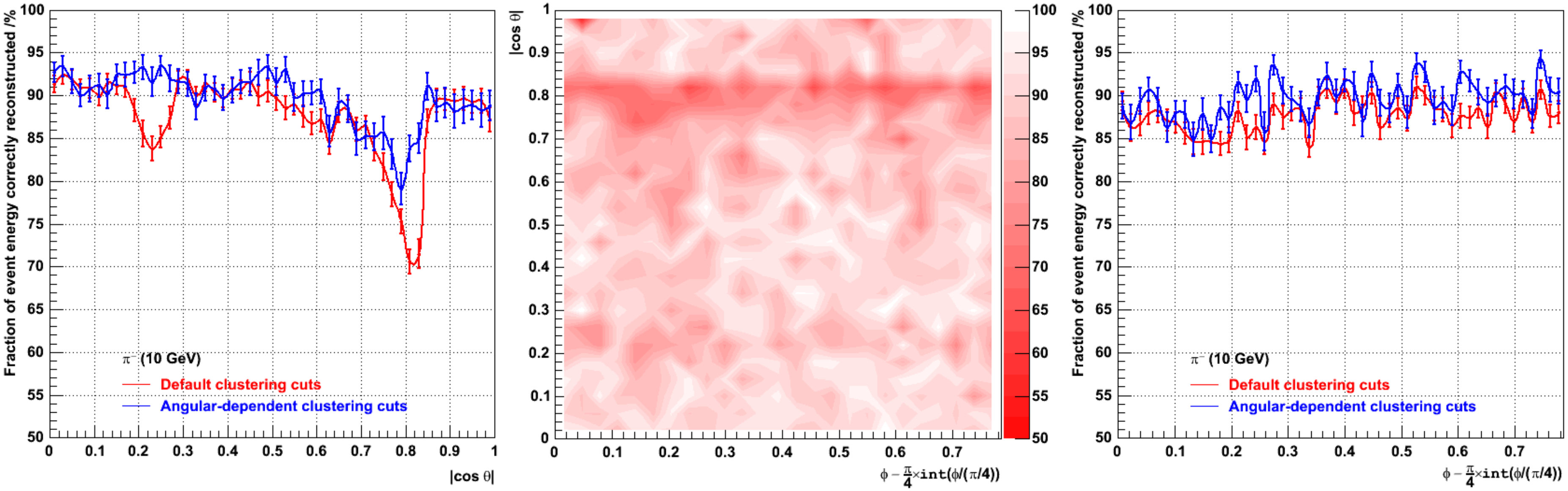
- μ⁻ fired isotropically into (analogue) **Si/W Ecal**, (digital) **rpc/Fe Hcal** (Mokka "D09" model).
- Cluster energies calibrated according to: $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\ mip}} + 20N_{\mathrm{Hcal}}]$ GeV.
- Fraction of event energy in highest-energy reconstructed cluster plotted vs |cos θ| and vs ϕ (folded into first octant: $0 \leq \phi < \pi/4$) at (0,0,0).



- Default clustering cuts → μ⁻ track fragmented at |cos θ| ~ 0.78 (barrel/endcap overlap), |cos θ| ~ 0.23 (gap between barrel modules) and ϕ ~ 0.20–0.24 (gap between Hcal barrel staves).
- Angular-dependent clustering cuts → μ⁻ track reconstructed with ~100% efficiency ∀ (θ, ϕ).
- Does relaxing cuts near dead zones impact on charged/neutral cluster separability though?

Detector scan: π^- (10 GeV)

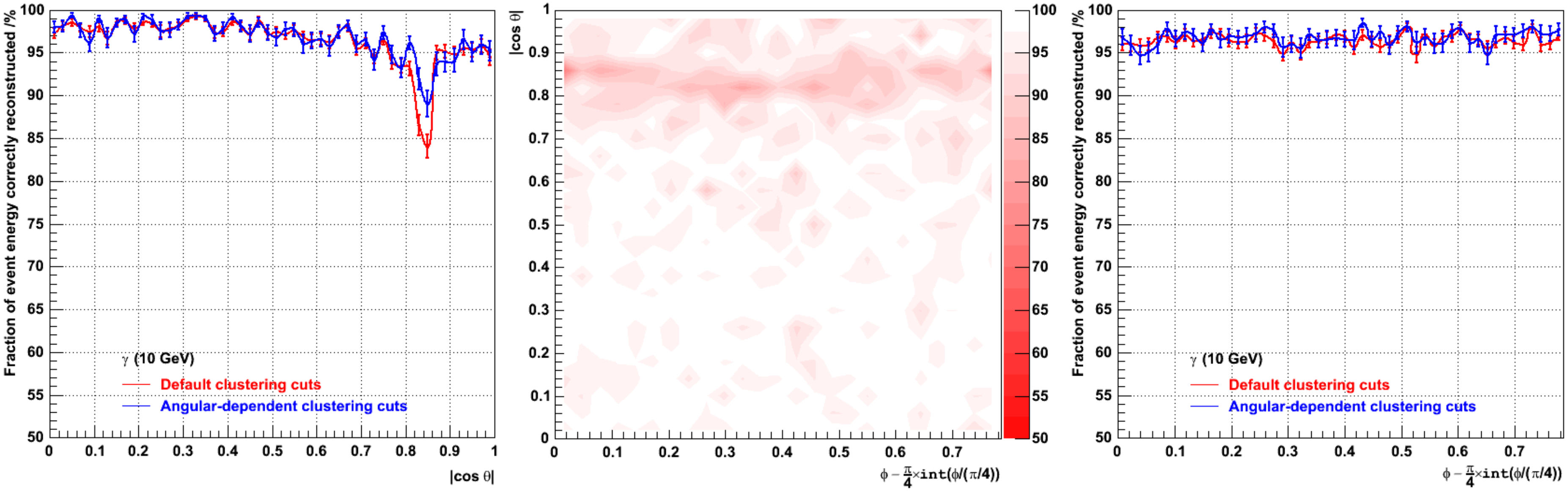
- π^- fired isotropically into (analogue) **Si/W Ecal**, (digital) **rpc/Fe Hcal** (Mokka "D09" model).
- Cluster energies calibrated according to: $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\ mip}} + 20N_{\mathrm{Hcal}}]$ GeV.
- Fraction of event energy in highest-energy reconstructed cluster plotted vs $|\cos\theta|$ and vs ϕ (folded into first octant: $0 \leq \phi < \pi/4$) at (0,0,0).



- Default clustering cuts → π^- shower fragmented at $|\cos\theta| \sim 0.83$ (barrel/endcap overlap) and $|\cos\theta| \sim 0.23$ (gap between barrel modules).
- Angular-dependent clustering cuts → π^- shower reconstructed with improved efficiency.
- Does relaxing cuts near dead zones impact on charged/neutral cluster separability though?

Detector scan: γ (10 GeV)

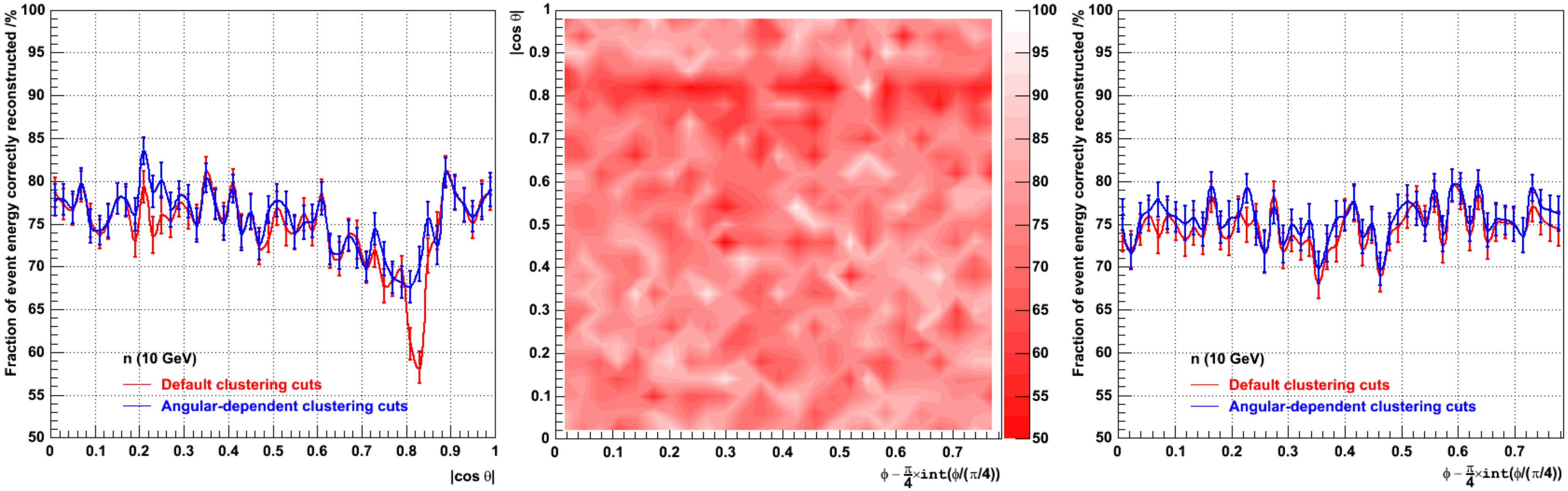
- γ fired isotropically into (analogue) **Si/W Ecal**, (digital) **rpc/Fe Hcal** (Mokka "D09" model).
- Cluster energies calibrated according to: $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\ mip}} + 20N_{\mathrm{Hcal}}]$ GeV.
- Fraction of event energy in highest-energy reconstructed cluster plotted vs $|\cos\theta|$ and vs ϕ (folded into first octant: $0 \le \phi < \pi/4$) at (0,0,0).



- Default clustering cuts → γ shower fragmented at $|\cos\theta| \sim 0.85$ (Ecal barrel/endcap overlap).
- Angular-dependent clustering cuts → γ shower reconstructed with improved efficiency.
- Does relaxing cuts near dead zones impact on charged-neutral cluster separability though?

Detector scan: n (10 GeV)

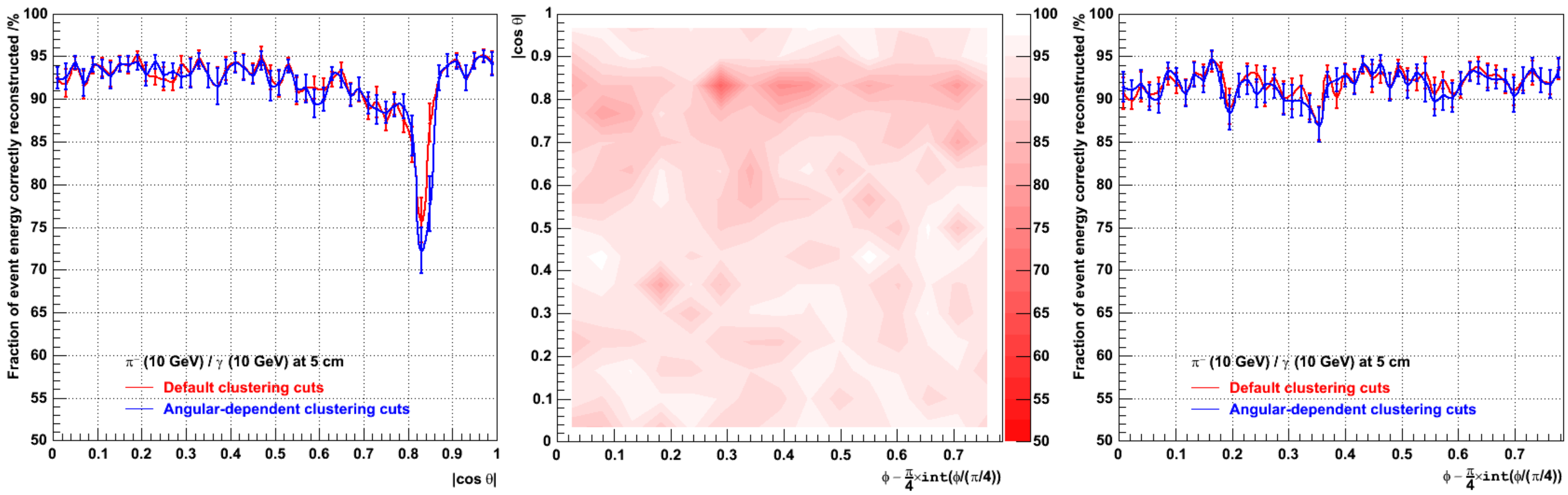
- **n** fired isotropically into (analogue) **Si/W Ecal**, (digital) **rpc/Fe Hcal** (Mokka "D09" model).
- Cluster energies calibrated according to: $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\ mip}} + 20N_{\mathrm{Hcal}}]$ GeV.
- Fraction of event energy in highest-energy reconstructed cluster plotted vs $|\cos\theta|$ and vs ϕ (folded into first octant: $0 \leq \phi < \pi/4$) at (0,0,0).



- Default clustering cuts → n shower fragmented at $|\cos\theta| \sim 0.83$ (barrel/endcap overlap).
- Angular-dependent clustering cuts → n shower reconstructed with improved efficiency.
- Does relaxing cuts near dead zones impact on charged-neutral cluster separability though?

Detector scan: π^-/γ at 5 cm (10 GeV)

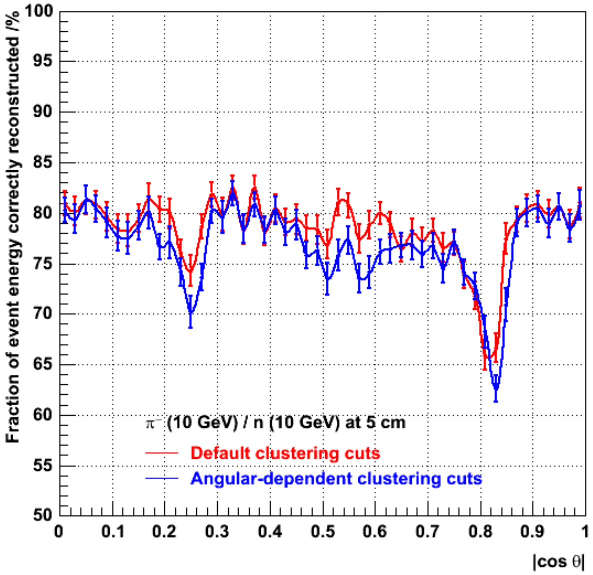
- π^-/γ fired 5 cm apart isotropically into (analogue) **Si/W Ecal**, (digital) **rpc/Fe Hcal** (Mokka "D09" model).
- Cluster energies calibrated according to: $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\ mip}} + 20N_{\mathrm{Hcal}}]$ GeV.
- Fraction of event energy in 1:1 correspondence between reconstructed and "true" clusters plotted vs $|\cos\theta|$ and vs ϕ (folded into first octant: $0 \le \phi < \pi/4$) on entry to Ecal.

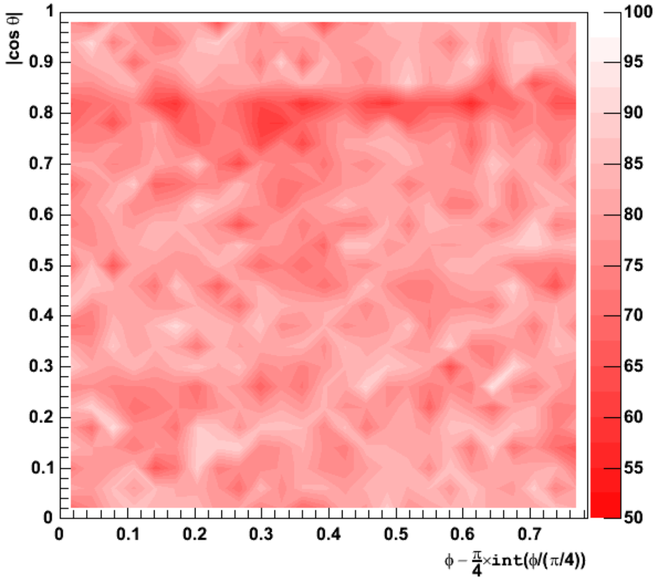


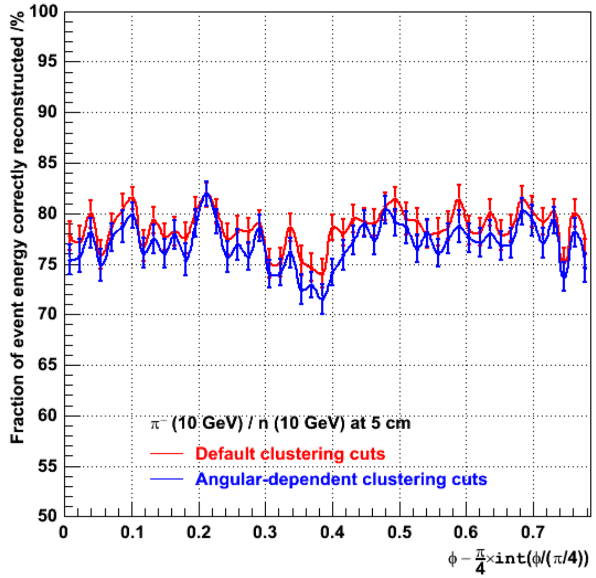
- Default clustering cuts → shower reconstruction/separability harder near $|\cos\theta| \sim 0.83$ (barrel/endcap overlap).
- Angular-dependent clustering cuts → improves single-particle reconstruction, but increases potential charged/neutral confusion (cuts relaxed).
- On balance, seems beneficial → separability barely affected.

Detector scan: π⁻/n at 5 cm (10 GeV)

- **π⁻/n** fired 5 cm apart isotropically into (analogue) **Si/W Ecal**, (digital) **rpc/Fe Hcal** (Mokka "D09" model).
- Cluster energies calibrated according to: $E = \alpha[(E_{\mathrm{Ecal;\,1\text{-}30}} + 3E_{\mathrm{Ecal;\,31\text{-}40}})/E_{\mathrm{Ecal\ mip}} + 20N_{\mathrm{Hcal}}]$ GeV.
- Fraction of event energy in highest-energy reconstructed cluster plotted vs $|\cos\theta|$ and vs ϕ (folded into first octant: $0 \le \phi < \pi/4$) on entry to Ecal.







- Default clustering cuts → shower reconstruction/separability harder near $|\cos\theta| \sim 0.83$ (barrel/endcap overlap) and $|\cos\theta| \sim 0.25$ (gap between barrel modules).
- Angular-dependent clustering cuts → improves single-particle reconstruction, but increases potential charged/neutral confusion (cuts relaxed).
- On balance, may again be beneficial, but need to be careful.

Clustering the prototype data...

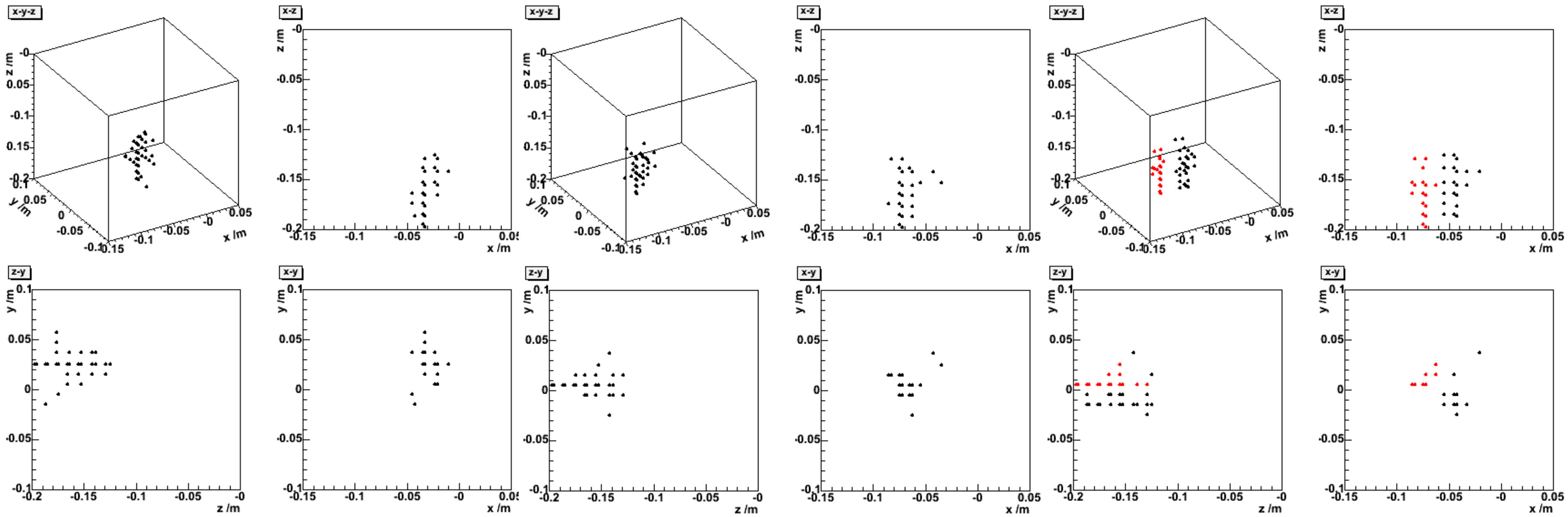
Steering file for the prototype

- Detector parameters and clustering cuts set in **cluster.steer**:

```
ProcessorType CalorimeterConfigurer
     detectorType              prototype   # “full” => barrel+endcaps; “prototype” => layers perp’r to +z
     iPx                       0.          # x-coordinate of interaction point (in mm)
     iPy                       0.          # y-coordinate of interaction point (in mm)
     iPz                       -99999.     # z-coordinate of interaction point (in mm)
     ecalLayers                30          # number of Ecal layers
     hcalLayers                40          # number of Hcal layers
     barrelSymmetry            8           # degree of rotational symmetry of barrel
     phi_1                     90.0        # phi offset of barrel stave 1 w.r.t. x-axis (in deg)
ProcessorType CalorimeterHitSetter
     ecalMip                   0.000150    # Ecal MIP energy (in GeV)
     hcalMip                   0.0000004   # Hcal MIP energy (in GeV)
     ecalMipThreshold          0.3333333   # Ecal hit-energy threshold (in MIP units)
     hcalMipThreshold          0.3333333   # Hcal hit-energy threshold (in MIP units)
ProcessorType CalorimeterStage1Clusterer
     layersToTrackBack_ecal    3           # number of layers to track back in Ecal
     layersToTrackBack_hcal    3           # number of layers to track back in Hcal
     distMax_ecal              20.0        # distance cut in Ecal (in mm)
     distMax_hcal              30.0        # distance cut in Hcal (in mm)
     proxSeedMax_ecal          14.0        # maximum cluster-seed radius in Ecal (in mm)
     proxSeedMax_hcal          50.0        # maximum cluster-seed radius in Hcal (in mm)
ProcessorType CalorimeterStage2Clusterer
     proxMergeMax_ecal         20.0        # Ecal proximity cut for cluster merging (in mm)
     proxMergeMax_hcal         30.0        # Hcal proximity cut for cluster merging (in mm)
     cosGammaMax               0.5         # angular cut for cluster merging
ProcessorType CalorimeterStage3Clusterer
     clusterSizeMin            10          # minimum cluster size to avert potential merging
     layersToTrackBack_ecal    39          # number of layers to track back in Ecal for merging
     layersToTrackBack_hcal    79          # number of layers to track back in Hcal for merging
     tanBetaMax                6.0         # angular cut for cluster merging
     proxSeedMax_ecal          400.0       # Ecal proximity cut for cluster merging (in mm)
     proxSeedMax_hcal          400.0       # Hcal proximity cut for cluster merging (in mm)
```

Prototype data (Run 100121) : e^- (1 GeV)

Event 803 | Event 59992 | Event 811



- 14 layers (analogue) **Si/W Ecal**; > 50k 1 GeV e^- events.
- Default clustering cuts → events generally reconstruct as single clusters (no tracking info used).
- On average, **98.93 ± 0.03 %** of event energy contained in highest energy reconstructed cluster (cluster energies calibrated according to: $E = \alpha(E_{\text{Ecal; 1-10}} + 2E_{\text{Ecal; 11-14}})$ GeV).

Conclusion

Conclusion...

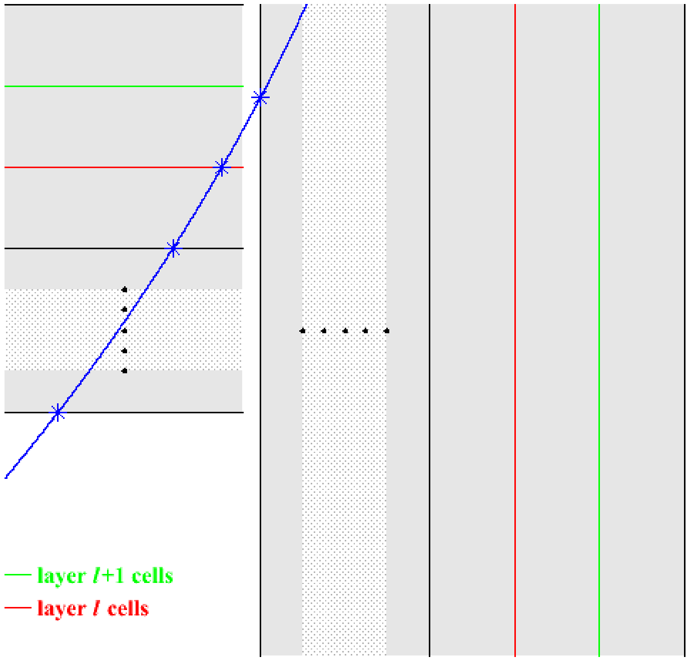
Summary & outlook

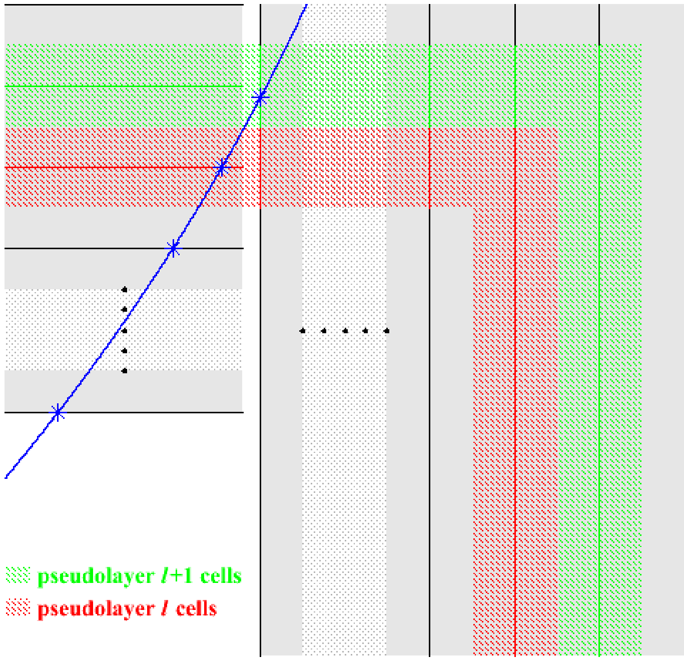
- Current version of *M*arlin-based *A*lgorithm for *G*eometry-*I*ndependent *C*lustering available from:
  http://www.hep.phy.cam.ac.uk/~ainsley/MAGIC/MAGIC-v01-02.tar.gz
- Will also put into CVS.
- Compliant with **LCIO** (≥ v01-05) / **MARLIN** (≥ v00-07) ⇒ input parameters (set at run-time) kept distinct from reconstruction (pre-compiled).
- Code straightforwardly **applicable to *any* detector geometry** comprising an ***n***-fold rotationally symmetric barrel closed by endcaps → just need to specify ***n***, barrel orientation, and layer positions as input.
- User specifies **geometry** and **clustering cuts** (user-defined angular-dependence in next version) at **run-time**.
- Algorithm can be used to compare different calorimeter designs straightforwardly (early hints of a preference for rpc over scintillator for Hcal using Mokka models).
- Please try it out!

The end

That's all folks…

Generalising the calorimeter (1)

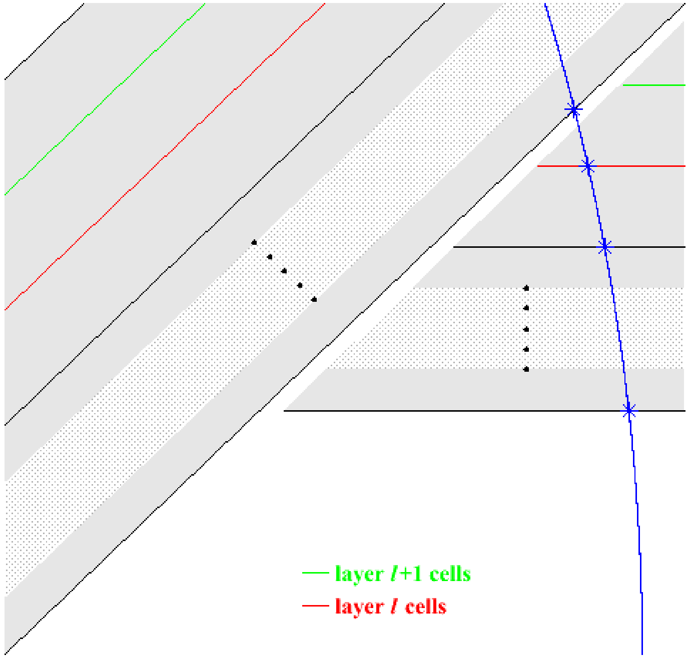


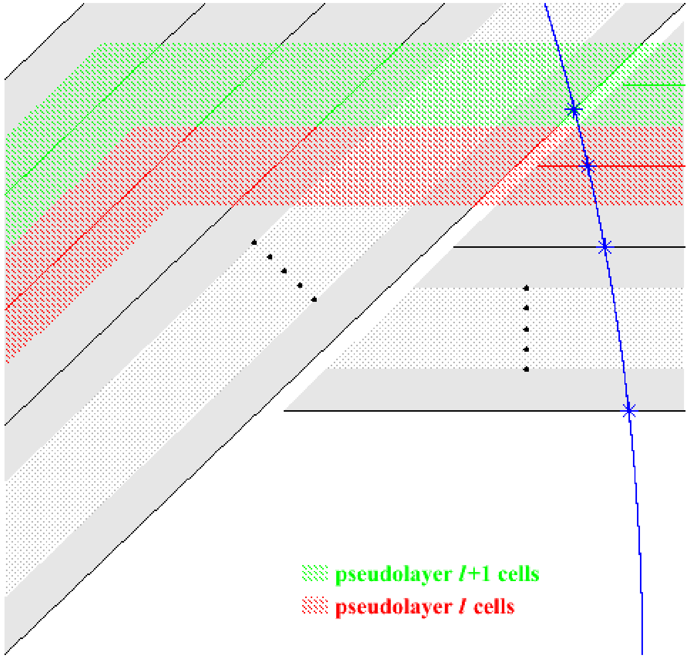


- Layer index changes discontinuously at barrel/endcap boundary.
- On crossing, jumps from *l* to 1 (first Ecal layer).

- Define a "***pseudolayer***" index based on projected intersections of physical layers.
- Index varies smoothly across boundary.
- Pseudolayer index = layer index, *except* in overlap region.

Generalising the calorimeter (2)

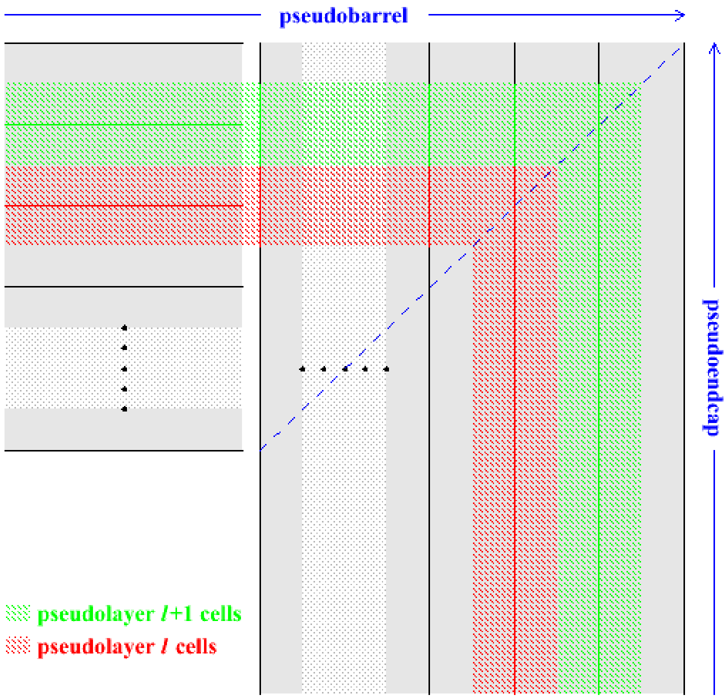


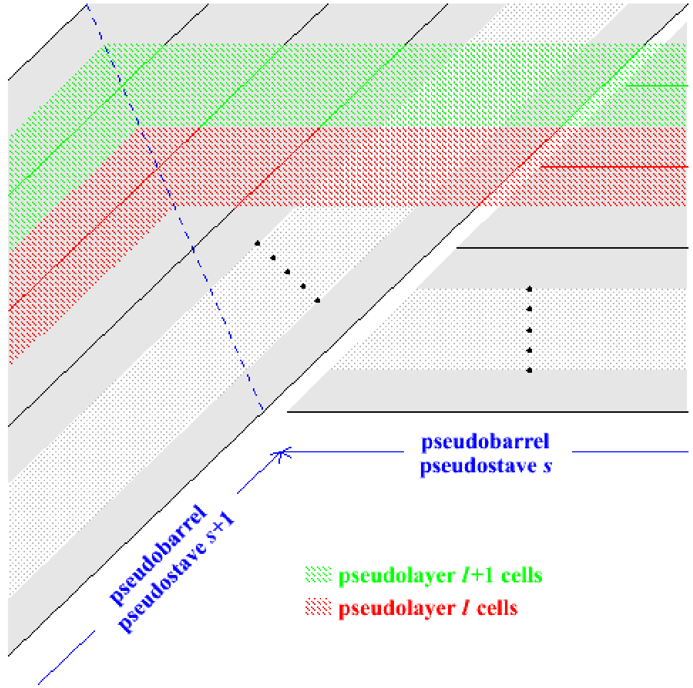


- Layer index changes discontinuously at boundary between overlapping barrel staves.
- On crossing, jumps from *l* to 1 (first Ecal layer.

- Again, define ***"pseudolayer"*** index from projected intersections of physical layers.
- Again, index varies smoothly across boundary.
- Again, pseudolayer index = layer index, *except* in overlap region.

Generalising the calorimeter (3)





- Define a ***"pseudostave"*** as a plane of parallel pseudolayers.
- ***"Pseudobarrel"*** pseudostaves meet boundaries with left- and right-hand ***"pseudoendcap"*** pseudostaves along 45° lines (if layer-spacings equal in barrel and endcaps).

- ***"Pseudobarrel"*** pseudostaves meet boundaries with other ***"pseudobarrel"*** pseudostaves along 360°/2***n*** lines (for an ***n***-fold rotationally symmetric barrel).
- Calorimeter divides naturally into ***n*+2** pseudostaves.

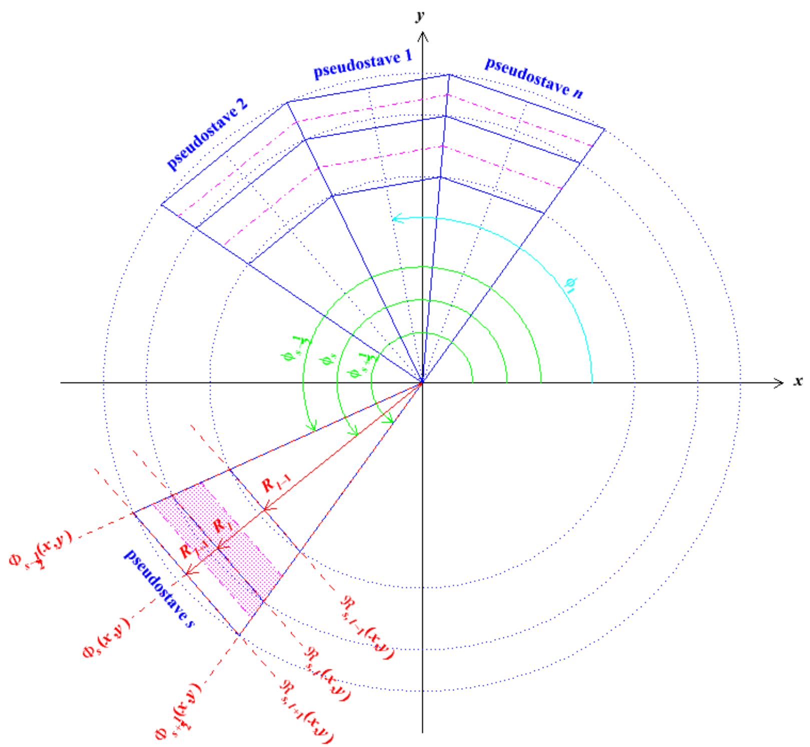
Generalising the calorimeter (4)

- Code recasts any layered calorimeter composed of a rotationally symmetric barrel closed by two endcaps into this standard, generalised form comprising layered shells of rotationally-symmetric *n*-polygonal prisms, coaxial with *z*-axis.
- Layers and staves from which calorimeter is built translated into pseudolayers and pseudostaves with which algorithm works.
- Only required inputs as far as algorithm is concerned are:
  - **`barrelSymmetry`** = rotational symmetry of barrel (***n***);
  - **`phi_1`** = orientation of pseudobarrel pseudostave 1 w.r.t. *x*-axis;
  - **`distanceToBarrelLayers[ecalLayers+hcalLayers+2]`**
    = layer positions in barrel layers ("**+2**" to constrain inside edge of first pseudolayer and outside edge of last pseudolayer); and
  - **`distanceToEndcapLayers[ecalLayers+hcalLayers+2]`**
    = layer positions in endcap layers;

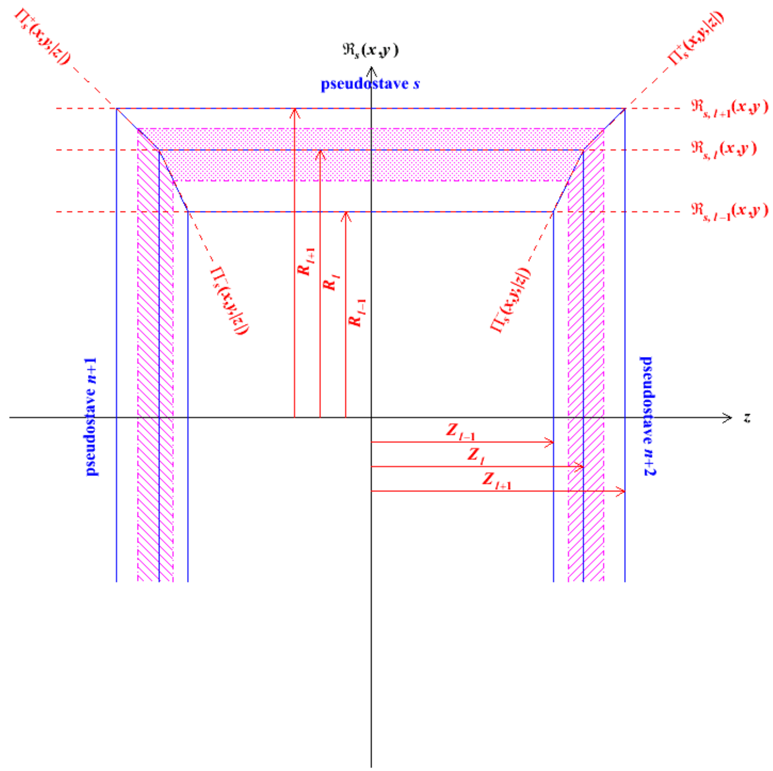
→ as geometry-independent as it's likely to get!

How the generalised detector shapes up

Transverse section



Longitudinal section

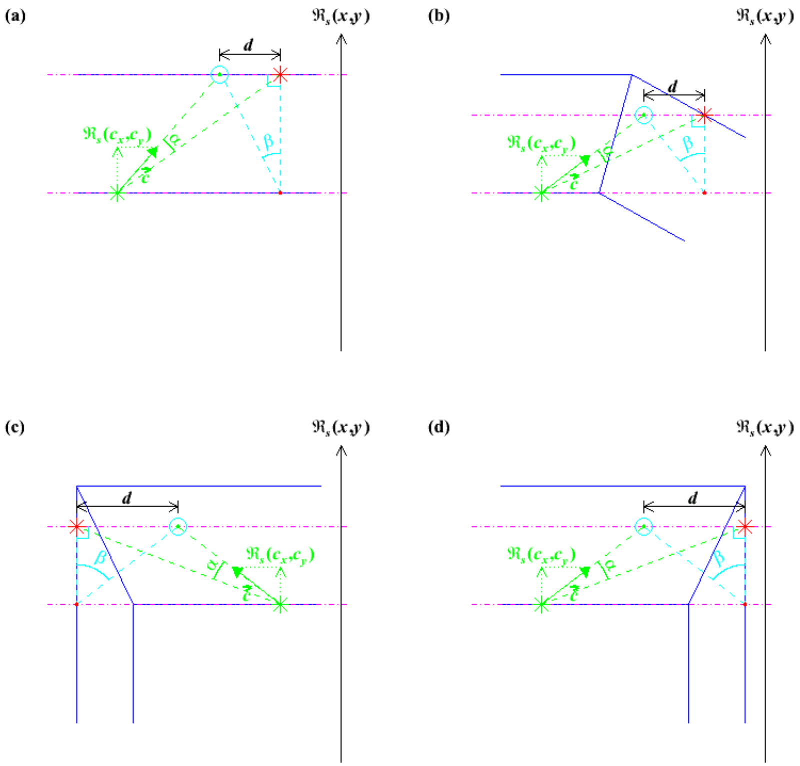


- Solid blue lines aligned along real, physical, sensitive layers.
- Dot-dashed magenta lines bound shell containing hits with same ***pseudolayer*** index, l.
- ***Pseudostaves*** automatically encoded by specifying n, ϕ_1 and R_l and Z_l ($\forall\ l$).

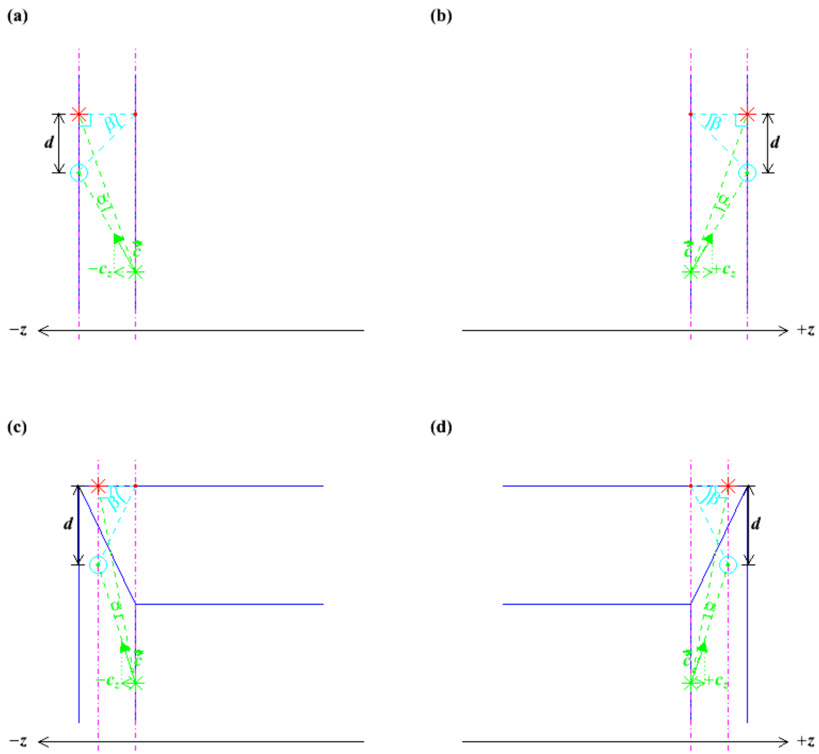
Chris Ainsley
<ainsley@hep.phy.cam.ac.uk>

Cluster-tracking between pseudolayers

From the pseudobarrel

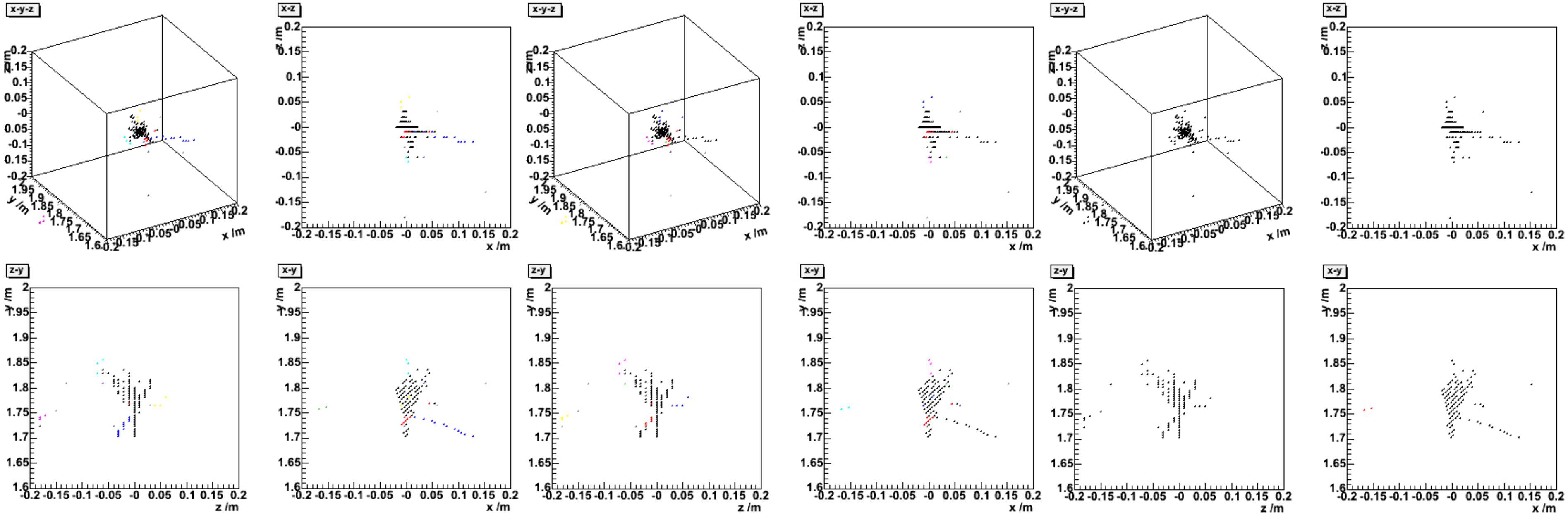


From the pseudoendcap



5 GeV π^+ event: 3 stages of clustering

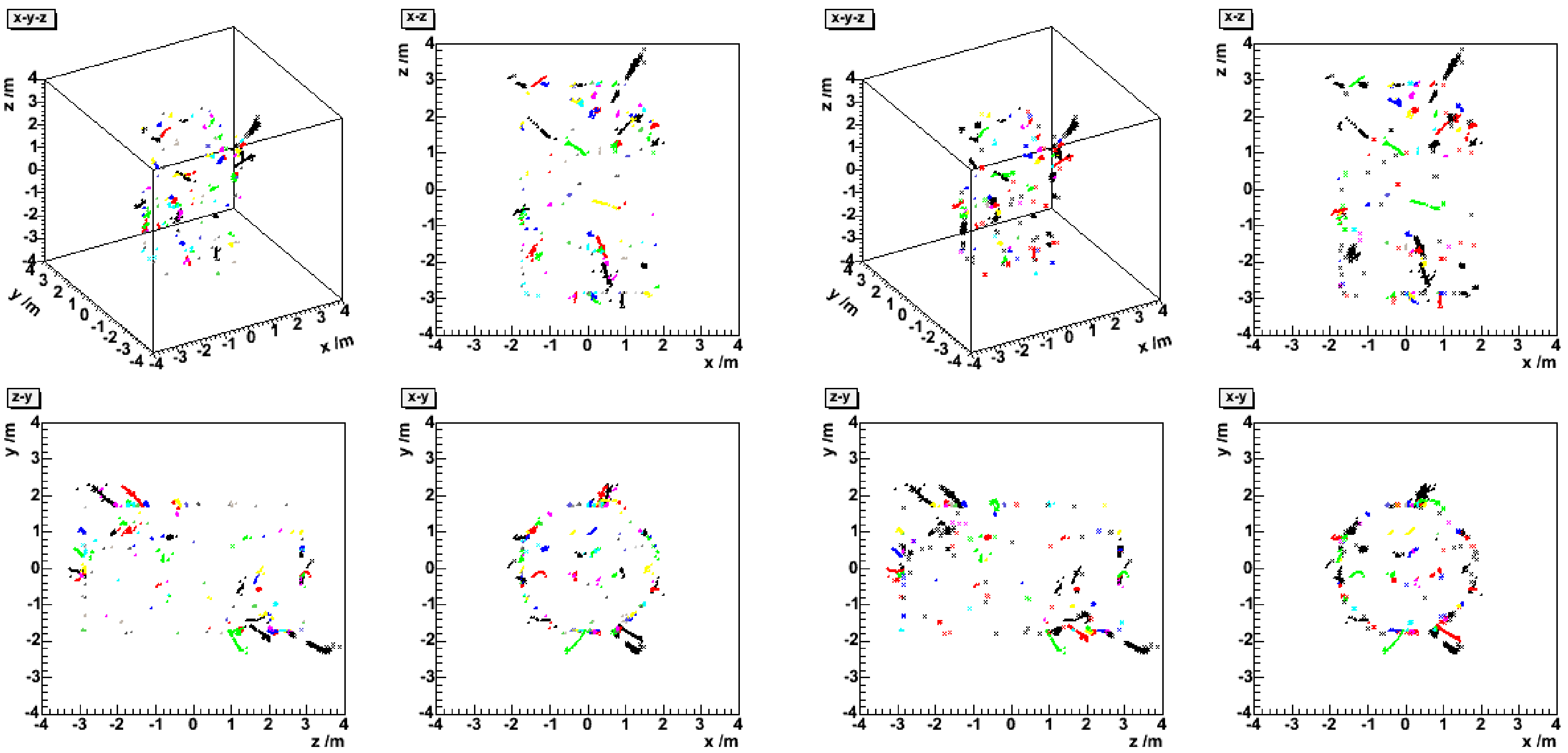
Clusters: stage 1 | **Clusters: stage 2** | **Clusters: stage 3**



- One **backward-spiralling track** and several **halo** clusters surround principal cluster.
- **Backward-spiralling track** merged with principal cluster.
- **Halo** clusters merged with principal cluster.

Example event: Z → u,d,s jets at 91 GeV

Reconstructed clusters | **True clusters**



- Reconstruction works successfully not only for ***intra***-stave, but also for ***inter***-stave clusters (*e.g.* ***black*** truth cluster spanning barrel staves 5+6 and the RH endcap correctly reconstructed).

Code organisation within LCIO/MARLIN

- Layer positions set (for convenience) in **CalorimeterConfigurer.cc**:

```
// Create collections to store the barrel and endcap layer positions
LCCollectionVec* distanceToBarrelLayersVec = new LCCollectionVec(LCIO::LCFLOATVEC);
LCCollectionVec* distanceToEndcapLayersVec = new LCCollectionVec(LCIO::LCFLOATVEC);

// Fill the collections with their positions (in mm)
for(int l=0; l<=ecalLayers+hcalLayers+1; l++) {
  LCFloatVec* distanceToBarrelLayers = new LCFloatVec;
  LCFloatVec* distanceToEndcapLayers = new LCFloatVec;
  if(detectorType=="full") {  // full detector
    if(l<=30) {  // first 30 Ecal layers at a pitch of 3.9 mm (+ layer 0)                ← edit
      distanceToBarrelLayers->push_back(1698.85+(3.9*l));                                ← edit
      distanceToEndcapLayers->push_back(2831.10+(3.9*l));                                ← edit
    }                                                                                    ← edit
    else if(l>30 && l<=ecalLayers) {  // last 10 Ecal layers at a pitch of 6.7 mm        ← edit
      distanceToBarrelLayers->push_back(1815.85+(6.7*(l-30)));                           ← edit
      distanceToEndcapLayers->push_back(2948.10+(6.7*(l-30)));                           ← edit
    }                                                                                    ← edit
    else {  // 40 Hcal layers at a pitch of 24.5 mm (+ layer 81)                         ← edit
      distanceToBarrelLayers->push_back(1931.25+(24.5*(l-41)));                          ← edit
      distanceToEndcapLayers->push_back(3039.25+(24.5*(l-41)));                          ← edit
    }                                                                                    ← edit
  }
  else if(detectorType=="prototype") { ...some more code... }  // prototype detector
  distanceToBarrelLayersVec->push_back(distanceToBarrelLayers);
  distanceToEndcapLayersVec->push_back(distanceToEndcapLayers);
}

// Save the collections
evt->addCollection(distanceToBarrelLayersVec, "distance_barrellayers");
evt->addCollection(distanceToEndcapLayersVec, "distance_endcaplayers");
```

Getting started with MAGIC

- For new LCIO `CalorimeterHits` collection can:
  - `getCellID0()`;
  - `getCellID1()` ⇒ pseudolayer/stave id encoded like layer/stave id in CellID0;
  - `getEnergy()`;
  - `getPosition()`;
  - `getType()` ⇒ "0"=ecal hit; "1"=hcal hit.
- For all new LCIO `Calorimeter*Clusters` collections, can:
  - `getCalorimeterHits()`;
  - `getHitContributions()`; and
  - `getClusters()`

  (no energy/position/shape attributes set—user can set these in own private processors as desired).
- If simulation, can also use `LCRelationNavigator` to:
  - `simHitRel->getRelatedToObjects(hit)`, and
  - `mCParticleRel->getRelatedToObjects(trueCluster)`.