HBD Update

Ermias Atomssa

H/L meeting

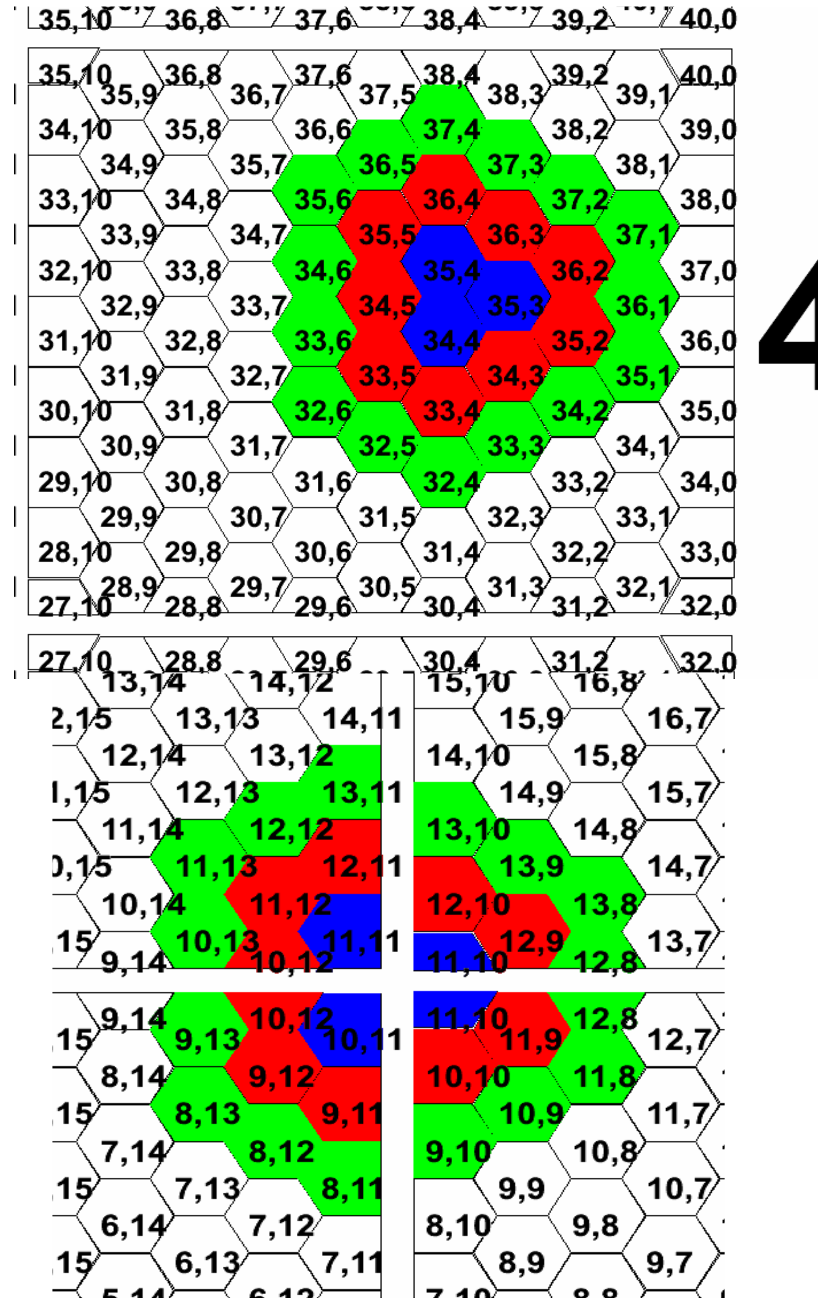
2010.12.15

Plan

- How the algorithm works
- Efficiency from MC
- Efficiency from embedding
  - Signal tracks and background tracks
- Data driven efficiency
  - Comparison with embedding efficiency
- Problems being currently worked on
- Disclaimer :
  - This presentation focuses on an algorithm that is being developed and tested at SBU by B. Bannier, J. Sun, S. Lee and me, it is NOT the only algorithm around.
  - Work on the algorithm is still under progress. There are still some kinks that we are aware of and that are included in this presentation.

Local background subtraction algorithm

- First step of the algorithm is the selection of preclusters.
  - Candidates for preclusters are all possible triplets in the HBD
  - Background is estimated for triplets as the median per pad charge of first and second neighbors.
  - Only triplets with a sufficient net charge are kept to the next step
- They have the following props.
  - Charge & area of members
  - Charge & area of 1st and 2nd neighbors
  - Net signal in member zone
  - Topology of charge distribution

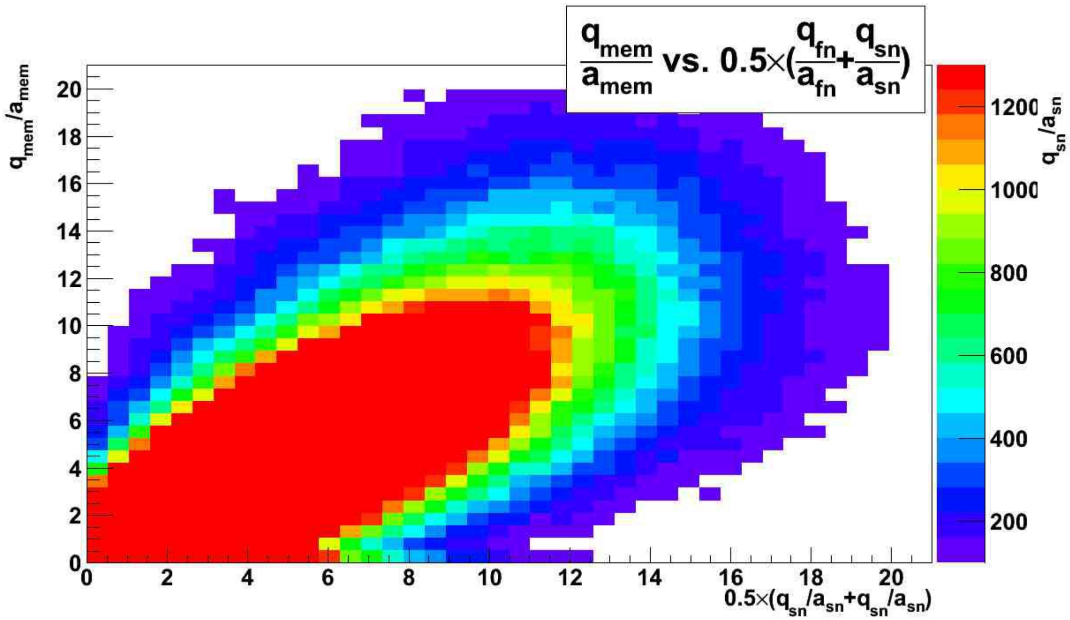


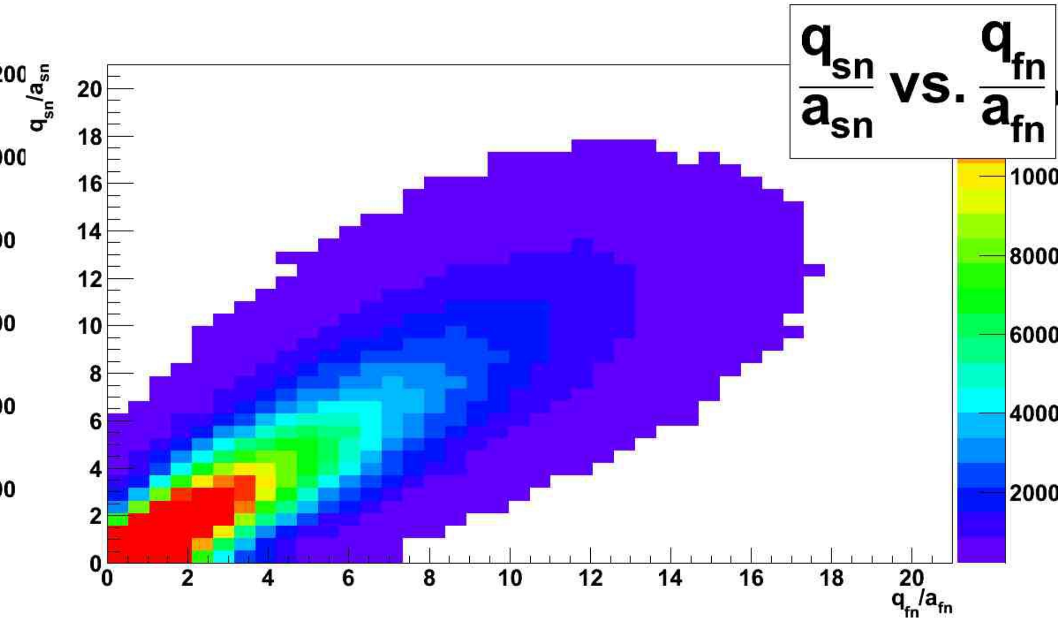
Justification of background estimation

- Basic assumption of the method
  - Scintillation background varies continuously over HBD surface
  - Background in any compact group of pads can be estimated from the average rate of npe in its neighboring pads

$$bkg = a_{mem} * \left(\frac{w_{fn} * q_{fn}}{a_{fn}} + \frac{(1 - w_{fn}) * q_{fn}}{a_{sn}} \right)$$

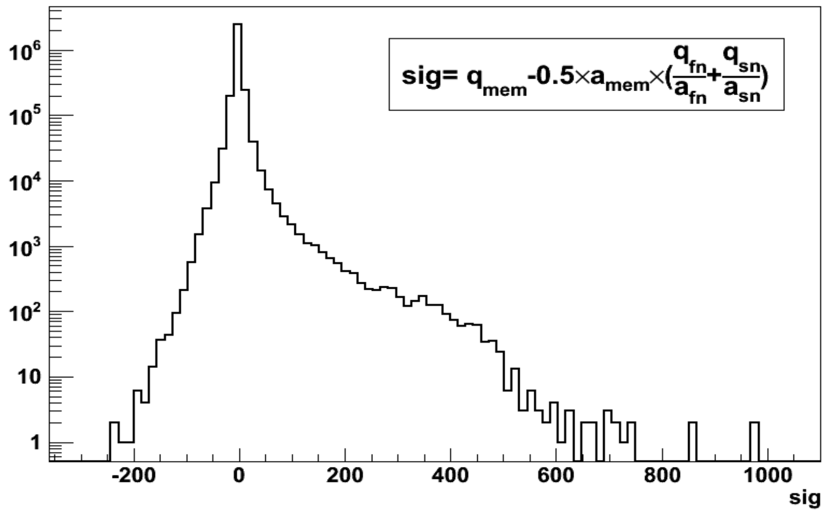
mem=triplet member fn=first neighbor, sn=second neighbor
a=area, q=number of photoelectrons
w= weight, for now set to 0.5





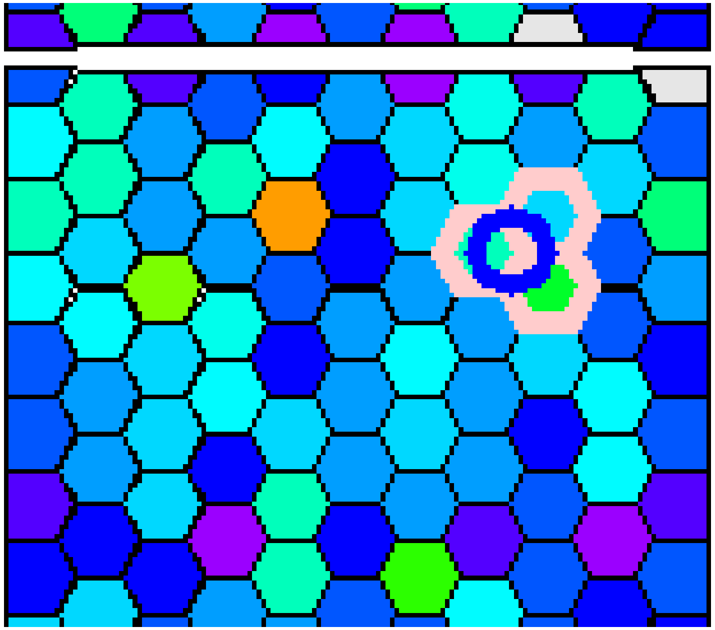
Precluster selection

- Don't want to keep everybody
  - Code will be slow
  - Will end up with superbig clusters
- What to keep?
  - Reasonable net signal
    - For now keeping 1<sig(npe)<50
    - This spans both the singles and doubles expected charge in a triplet
  - Shape cut (not implemented yet)
    - One idea is to require that the track points into the highest firing pad in the cluster



Merging

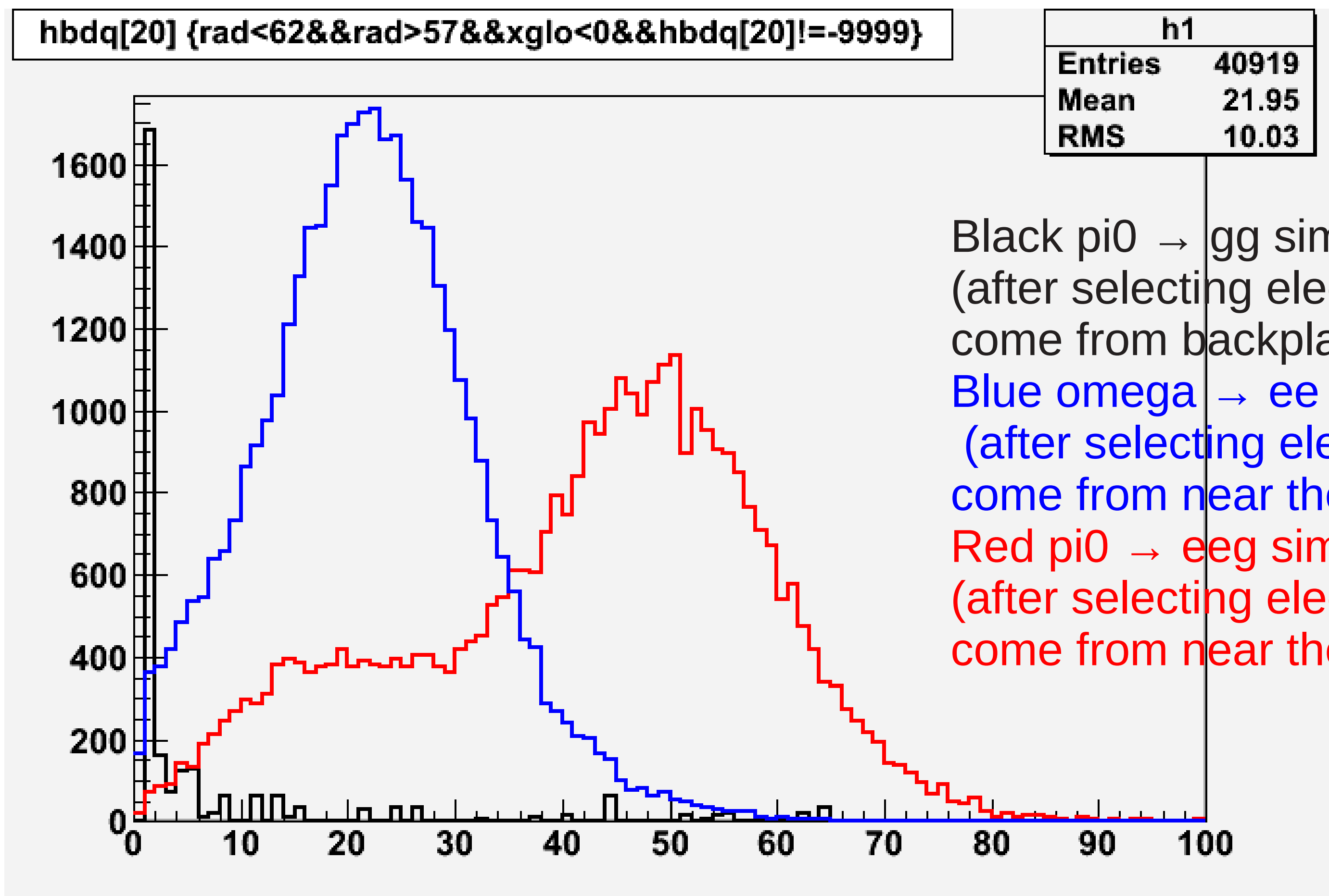
- Once a subset of good preclusters is selected, all good preclusters (triplets) within an adjustable radius of an electron track projection are merged to form the final cluster
- The merging radius should be
  - Big enough to catch all pads from which the charge was deposited
  - Small enough not to include pads from scintillation background
- By construction, final cluster to track matching distribution is centered at zero.
- Caution: The geometry description should be exact for this approach to wrok.



Matching distribution (from MC)

Charge distribution in MC

- Three MC samples were sent from Weismann (Only Cerenkov light is simulated, no scintillation)
  - pi0->gg (used to study response to backplane conversions)
  - omega->ee (used to study response to single electron hits)
  - pi0->eeg (used to study response to double electron hit
- Ancestry information is used in what follows to make sure that we are looking at the “right” electrons



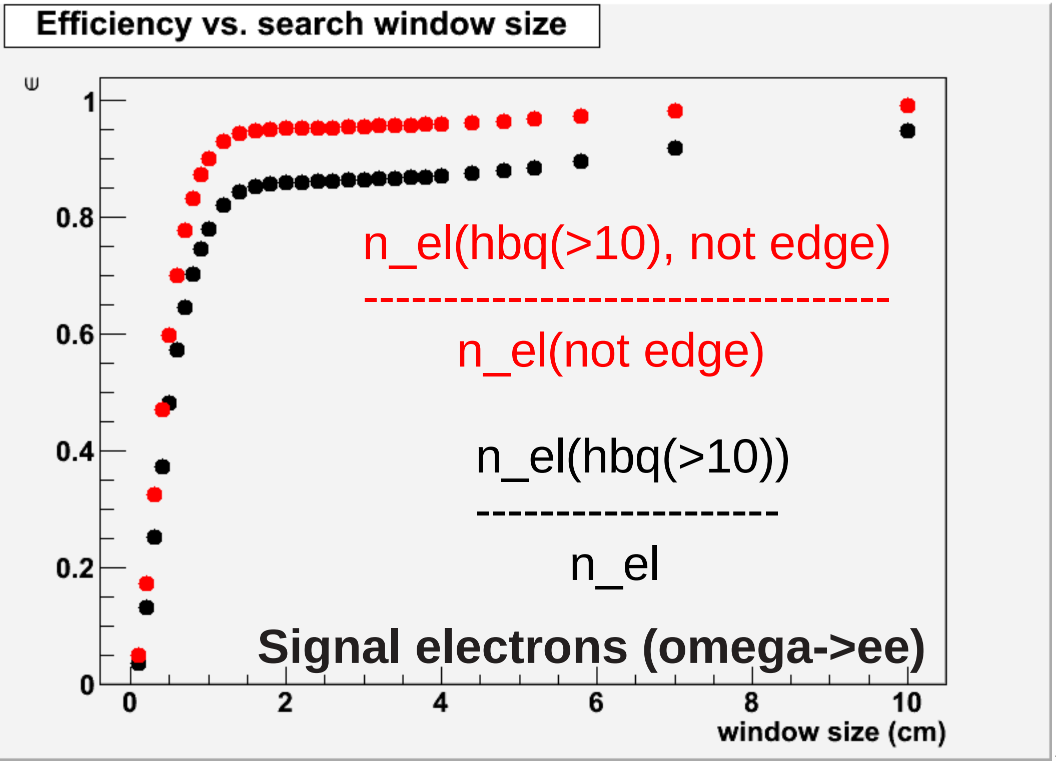
Black pi0 → gg simulation (after selecting electrons that come from backplane of HBD)

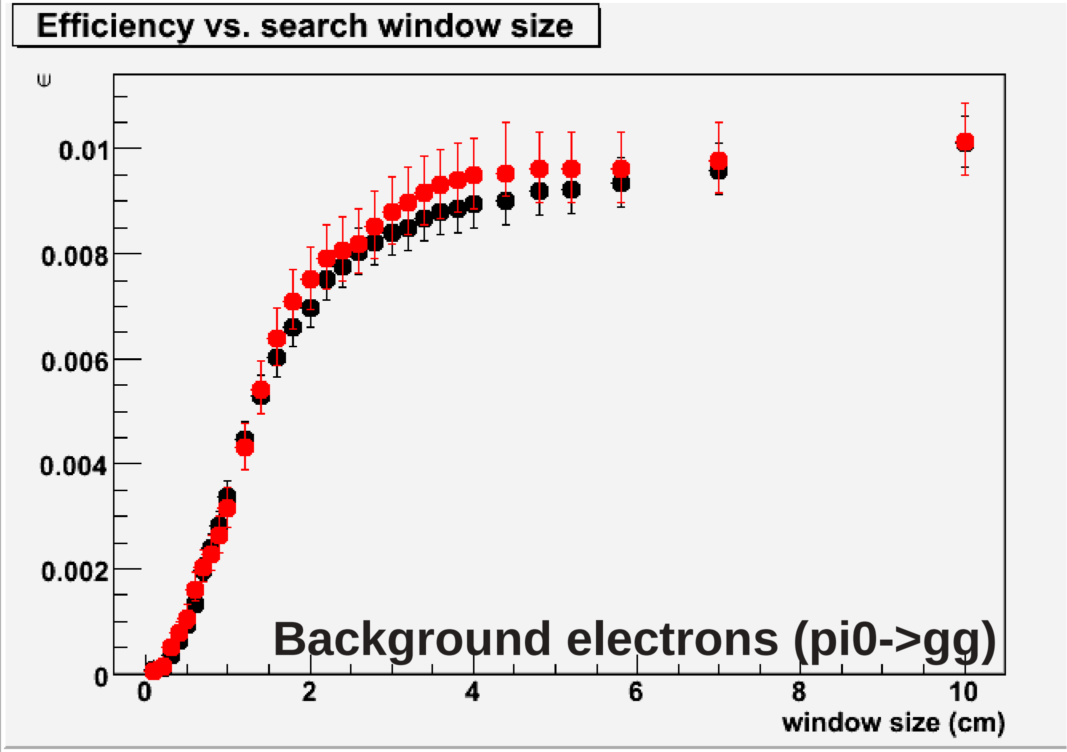
Blue omega → ee simulation (after selecting electrons that come from near the vertex)

Red pi0 → eeg simulation (after selecting electrons that come from near the vertex)

Efficiency from MC (vs. merging window rad)

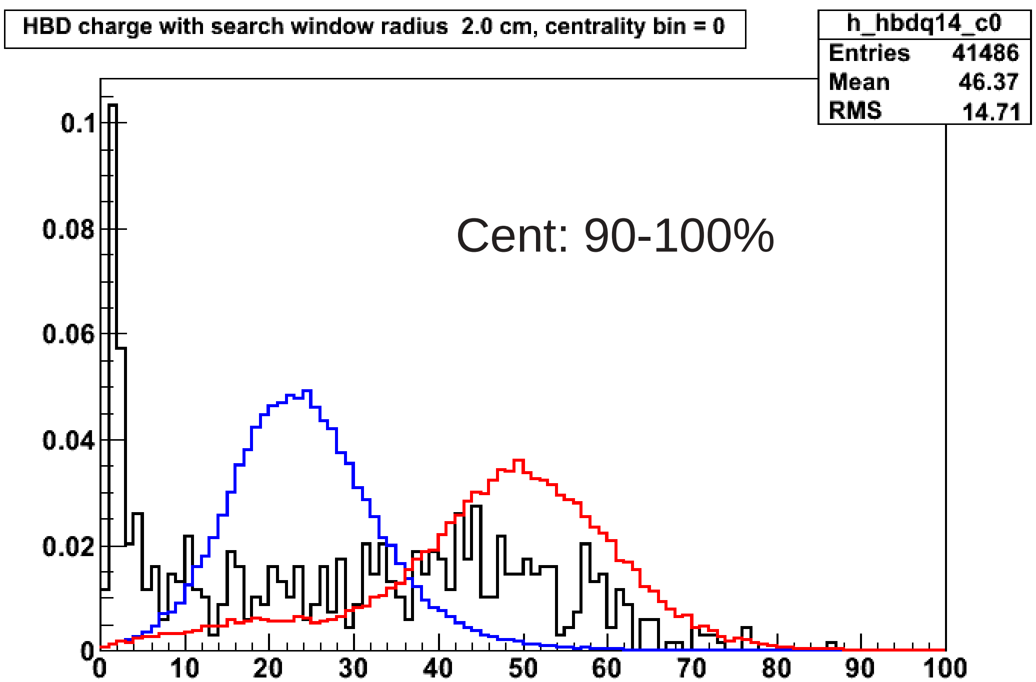
- The efficiency for charge cut of >10 saturates at merging window radius of about 1cm.
- It saturates at ~85% for all electrons and ~95% if we cut the HBD edge areas off with a fiducial cut (3cm used here)

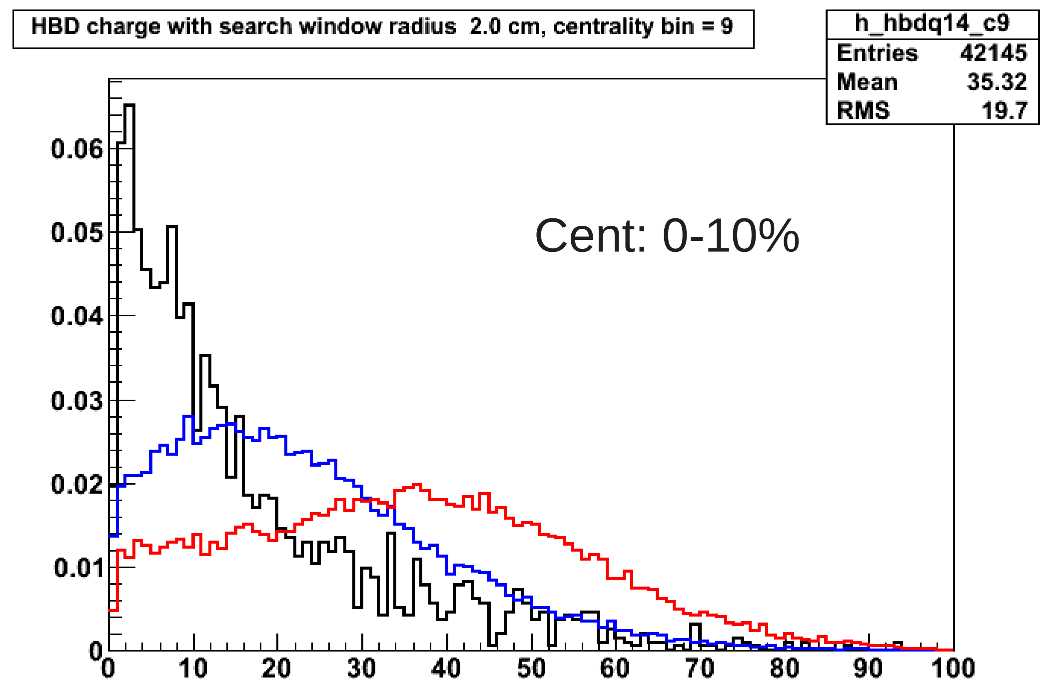




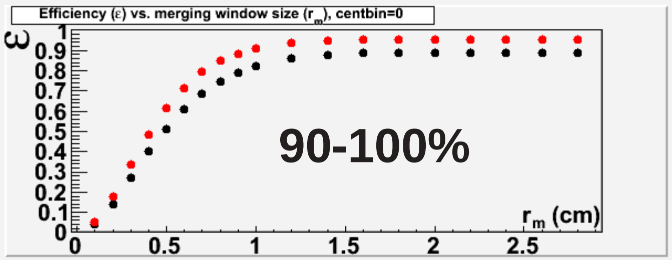
Embedding

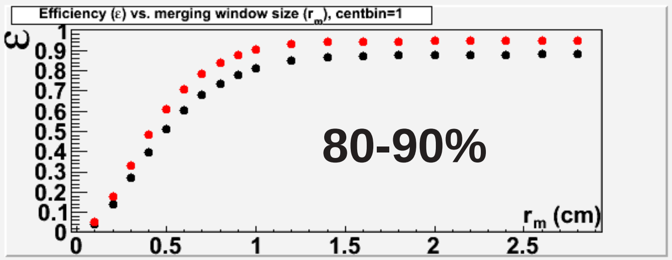
- We have a tool to embed MC HBD response into real data hbd response
  - Objective is to quantify the performance of HBD algorithms in realistic situation (ie with underlying background from scintillation/curlers)
  - Procedure is simple pad-by-pad summation of charge from MC and RD before running the clusterization algorithm
  - Then original MC charged tracks are matched to clusters found in the HBD after embedding
  - (Poor man's) centrality is determined by subdividing the events into 10 equal sized groups following the distribution of number of reconstructed charged tracks per event

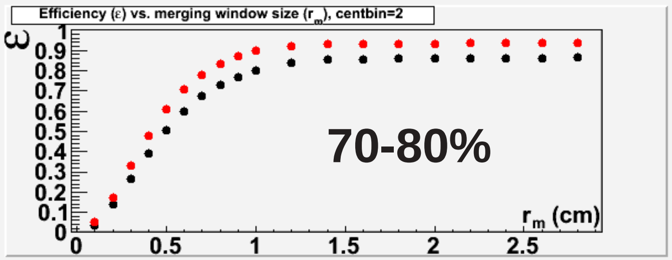


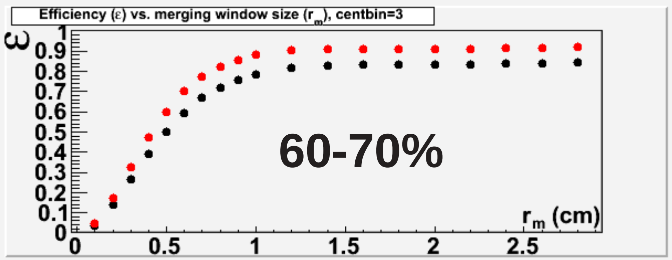


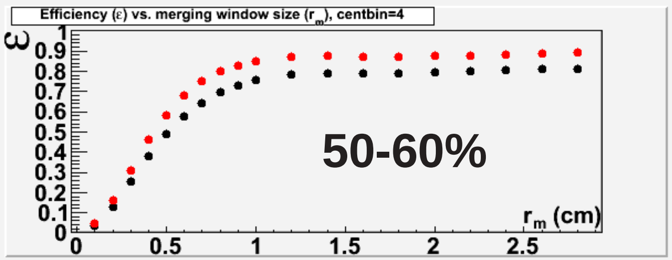
Efficiency for signal electrons (omega->ee)

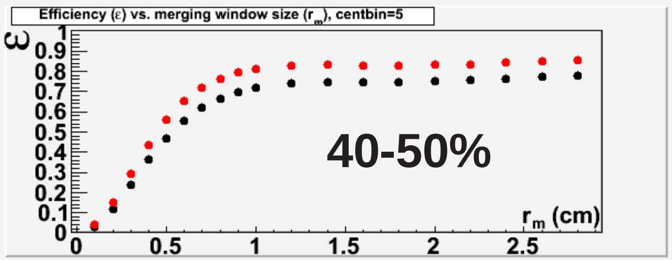


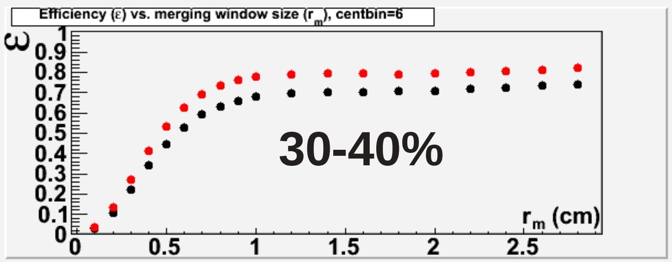


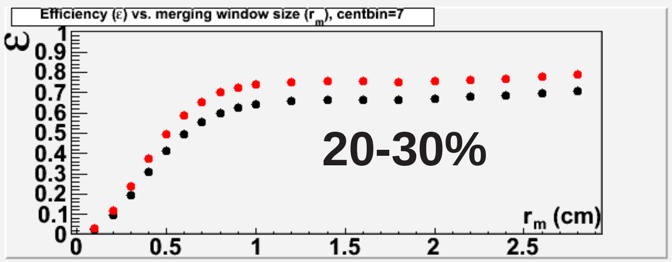


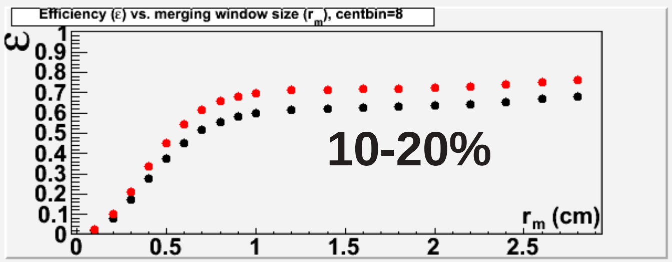


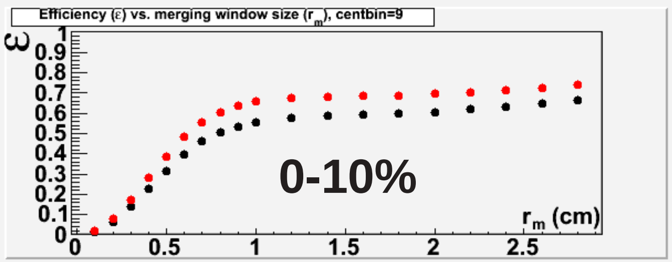






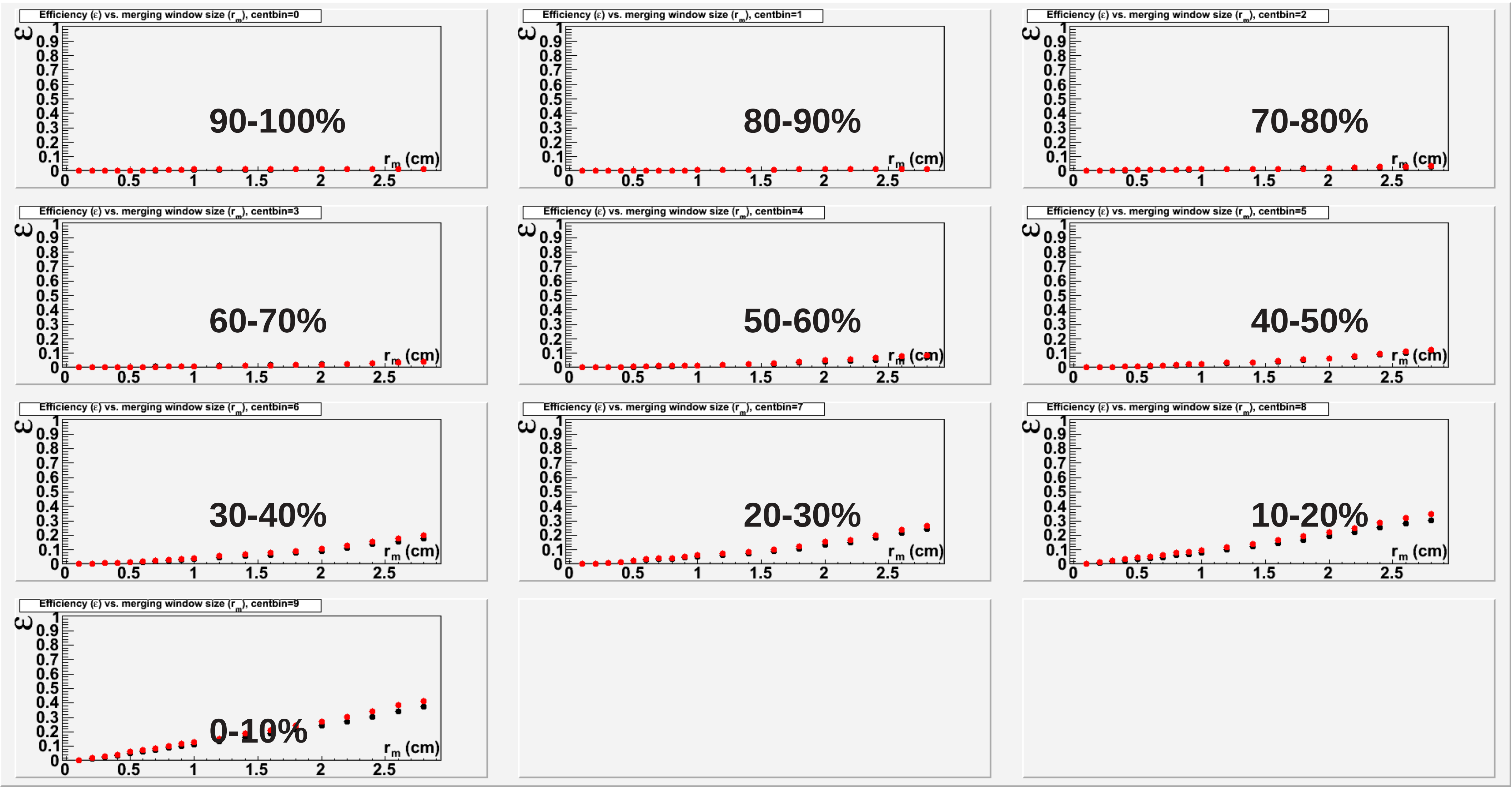






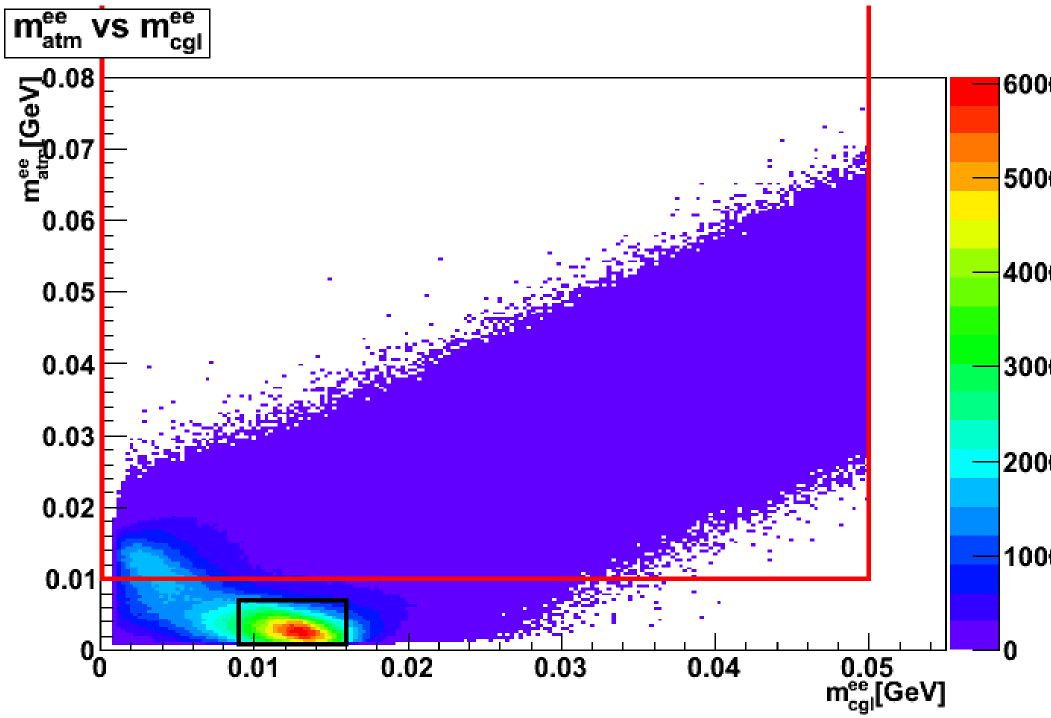
- centrality

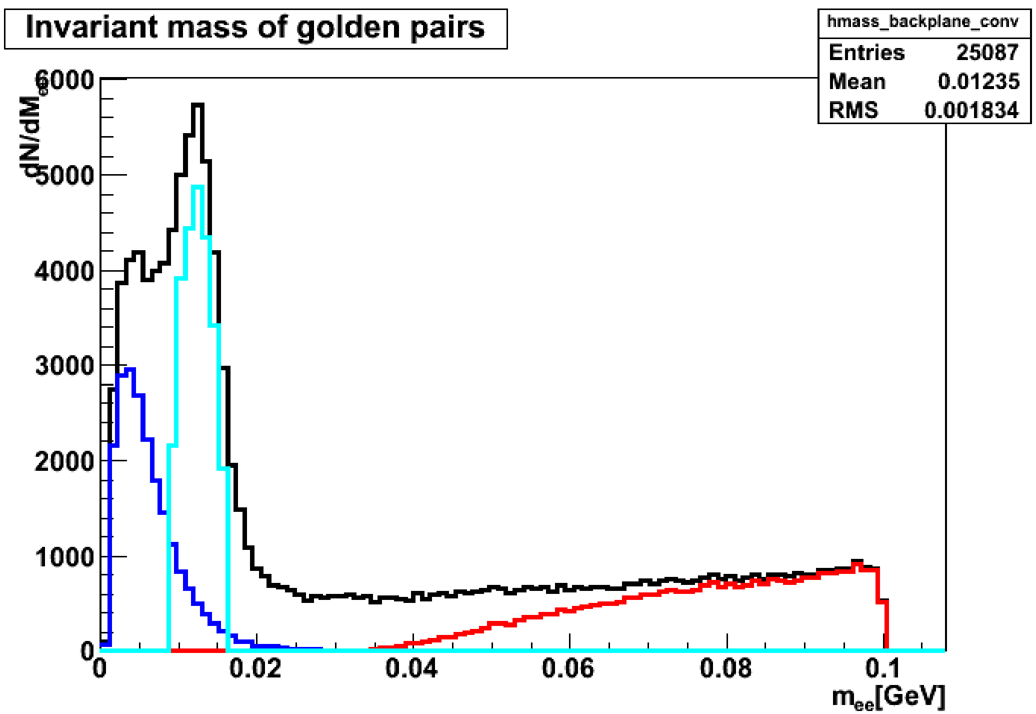
Efficiency for background electrons (pi0->gg)



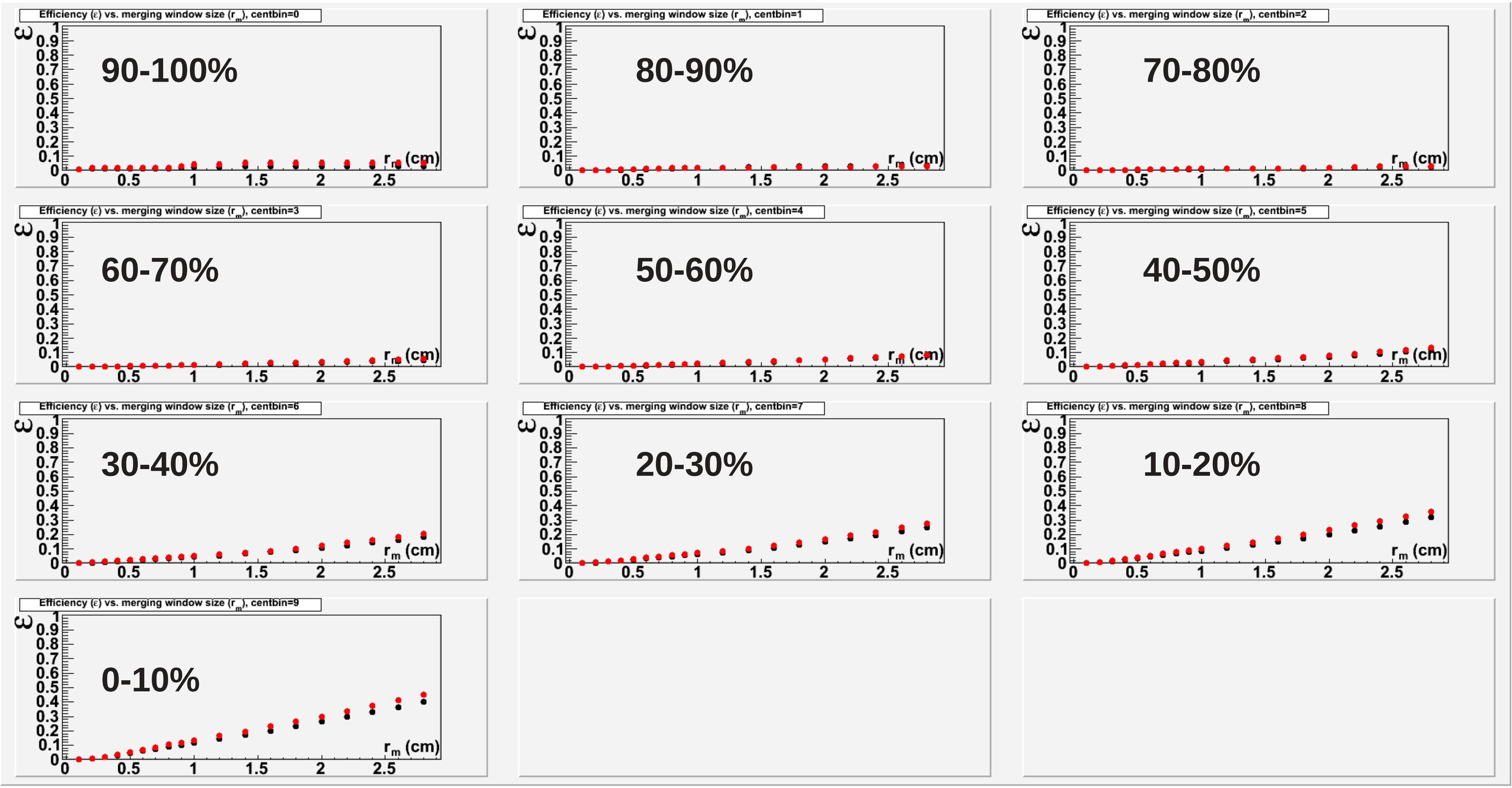
Real data golden track selection

- Compact CNT files with full events (electron trk + hbd info):
  - MB events |zvtx| < 10 cm, and containing clearly identified electron pairs
    - Backplane conv. : eid && mass < 7 MeV && 9 Mev < mass(cgl) < 16 Mev && oa(atm) < 30mr
    - Open Dalitz pairs : eid + mass(atm) > 10 MeV + mass(cgl) < 50 MeV && oa(cgl) > 150 mr
    - Closed Dalitz pairs : eid + mass(atm) > 10 MeV && mass(cgl) < 50 MeV && oa(cgl) < 30 mr
  - HBD clusterization algorithms can thus be run to cross check embedding efficiencies

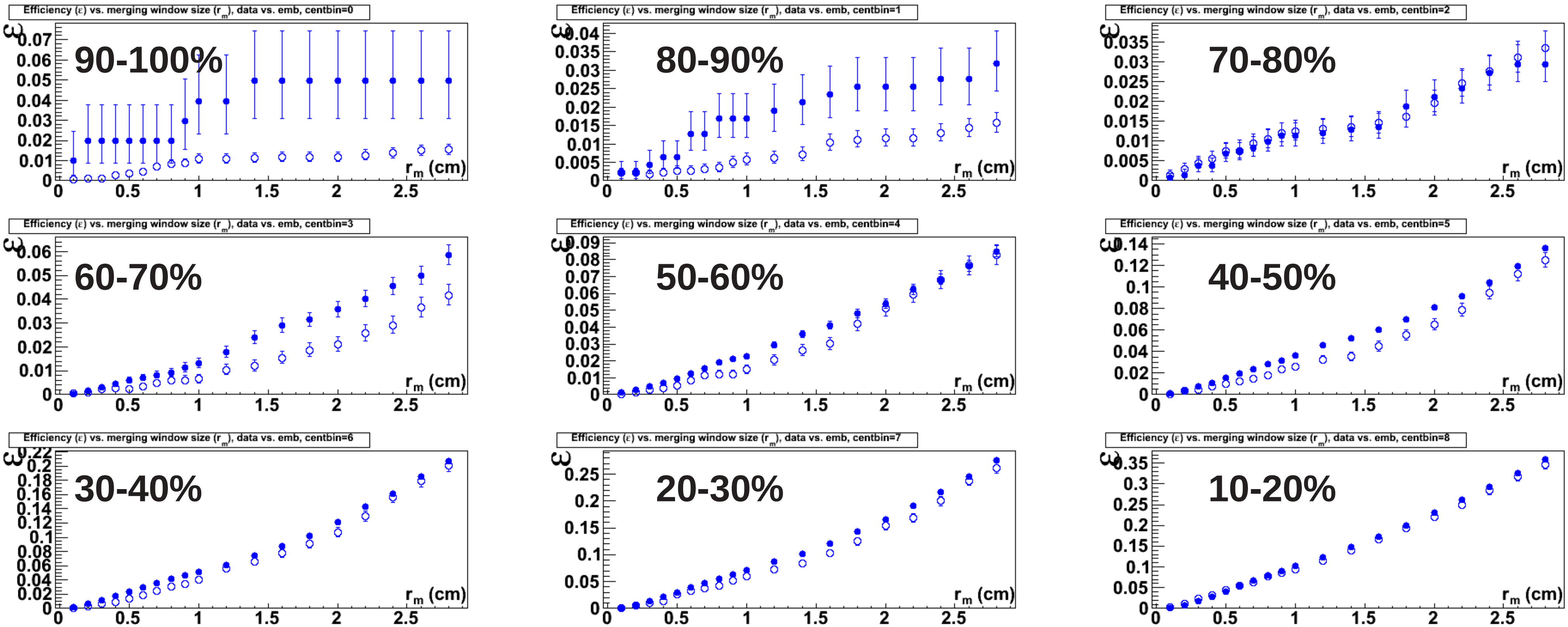


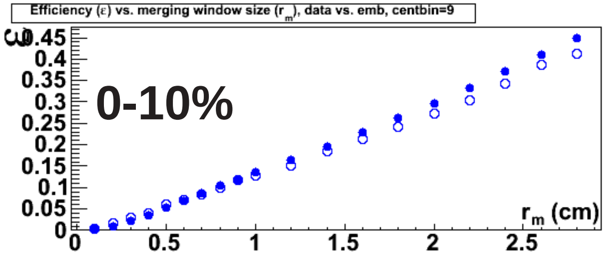


Efficiency for background in real data



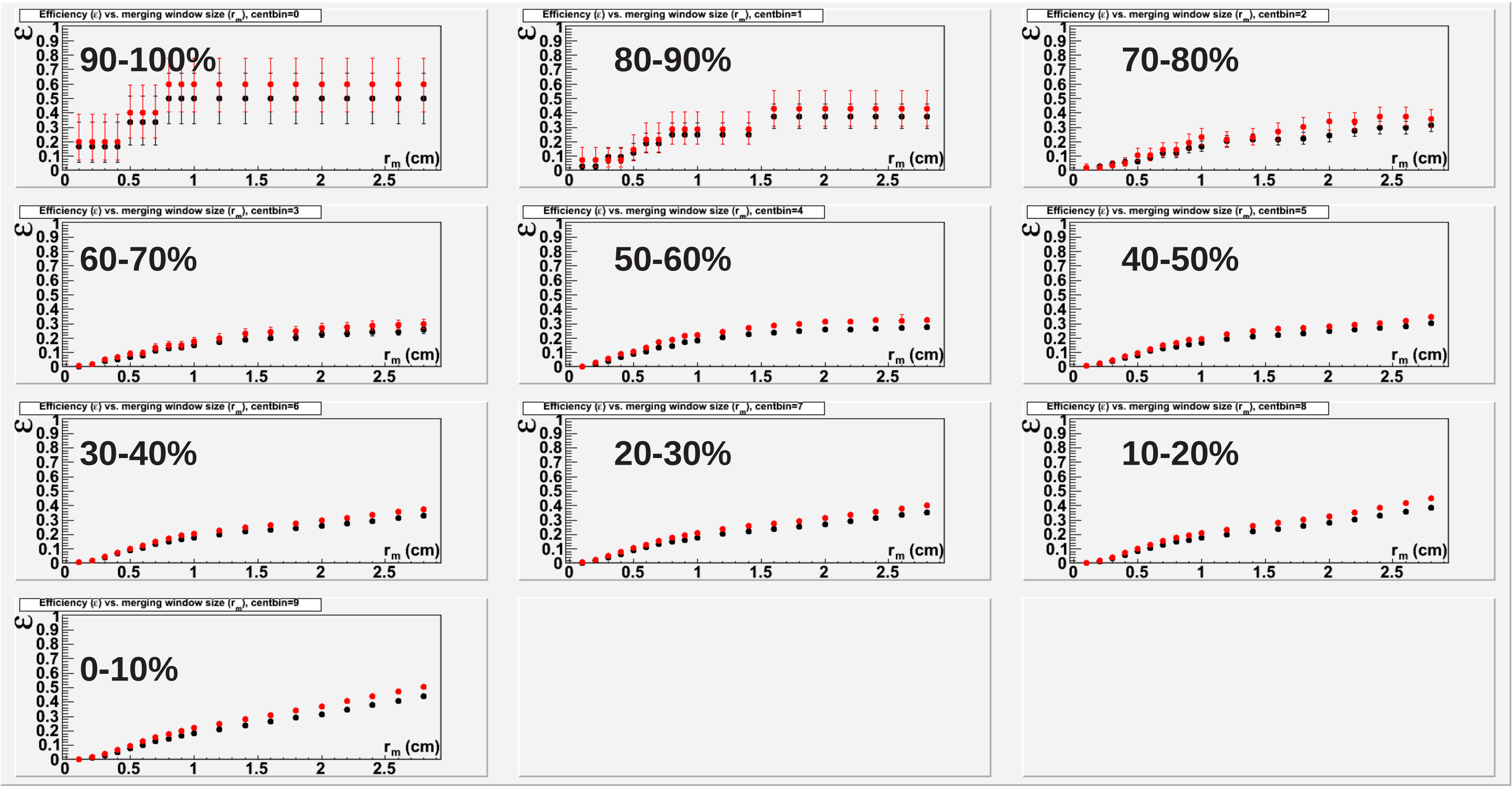
Comparison to embedded MC





Open symbol : Efficiency from embedded backplane conv.
Closed symbol : Efficiency from real data backplane conv.

Efficiency for signal electrons in real data



Discussion

- The embedding efficiency getting near our objective
  - Single e ~ 70% for and backplane conv. ~ 10% for 0-10%
  - Objective (first step atleast): reject backplane conversions by a factor of ~40 (*) or efficiency for backplane conv of ~2.5% while keeping a reasonable single electron efficiency
  - We still have some way to go, and there are ideas we are testing
    - Use charge topology: ex. Request that track points to pad with maximum charge
    - Use central arm tracking: Backplane conversions are mis-reconstructed
- The real data cross check of efficiencies:
  - The backplane conversion efficiency seems to match pretty well (between data and embedded simulation) taking into account that the data sample can have some contamination from real electrons
  - The signal electron efficiency from real data is much lower than that found in embedding. This can be due to either of two causes:
    - The real data “open Dalitz” sample has a strong contamination from backplane conv.
    - Real data HBD geometry misalignment. This is a known issue that is being worked on actively

(*) The radiation length from HBD backplane in run 10 is ~4x that in run4 from air and beam pipe.
In order to reduce the backplane conversions to a fraction x of the run 4 level, we need to have a rejection factor of 4/x*. Say to get x=10%, we need a rejection of about 40.

Summary

- We have developed at SBU an alternate clusterization for the HBD which we think is particularly useful in high background
  - Basic idea is to estimate the scintillation background level from the immediate vicinity of the clusters and subtract out after summing the charges of pads identified as constituting the cluster
  - Track projections are used to facilitate the search of clusters
- We have developed a set of tools to evaluate the effectiveness of our algorithm
  - Embedding MC HBD response in real data
  - Work with a set of well identified tracks with distinct HBD response
- The results of testing our algorithm is ongoing
  - We are close to the first objective of effectively rejecting backplane conversions though efficiency from data driven method doesn't match embedded MC for single electrons
  - Need some more work on the single/double identification front (which remains the main objective of using the HBD for low mass dilepton spectrum analysis)

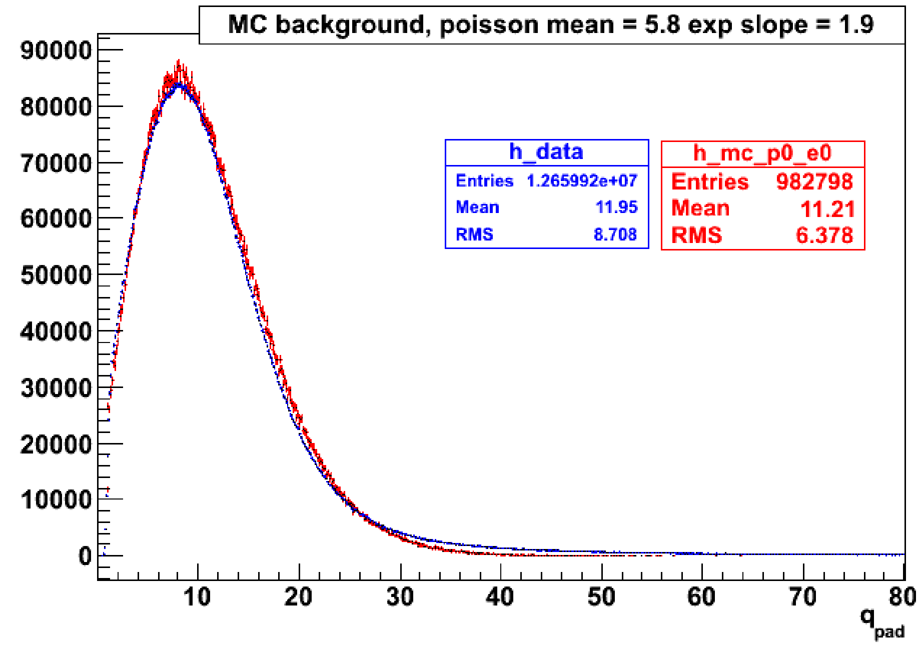
Backup

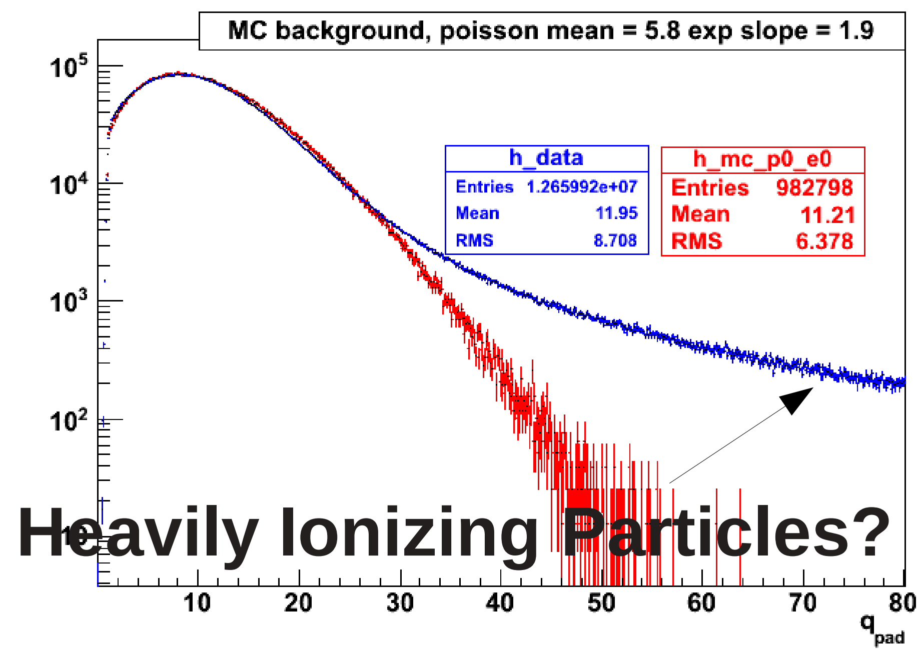
Mimic the real data backgrond

- Attempt to generate RD like background

$$q=\sum_{0}^{P(M)} \exp(\tau)$$

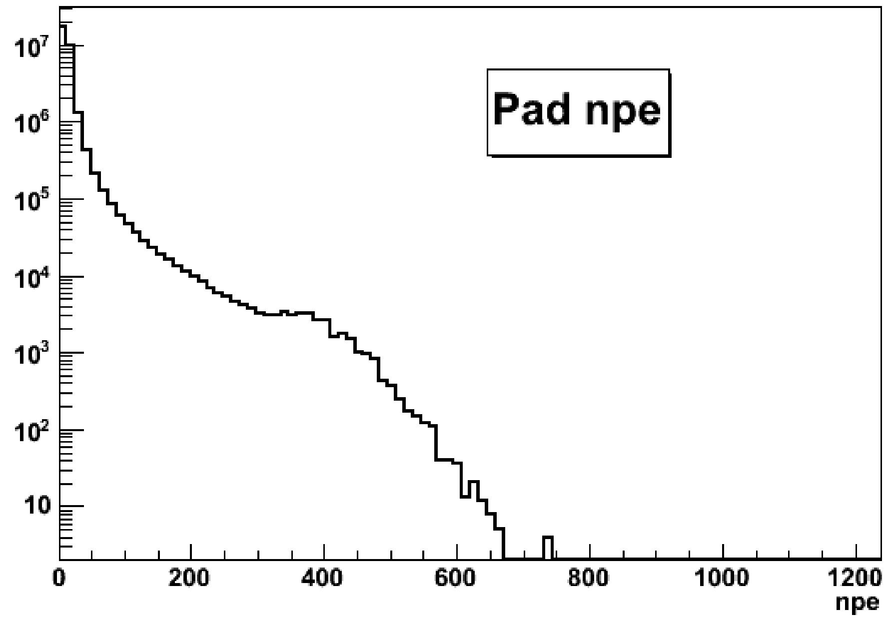
  - M (Poisson RV mean) and tau (Exp. RV decay const.) are hand tuned to match the RD pad charge distribution
    - Ten centrality bins of 10%
    - The long tail in RD is hard to reproduce (probably coming from jets? If so maybe can be added with some effort.)
    - This kind of detail matters for clusterizing
  - Using temporarily as a rough approximation to scintillation background

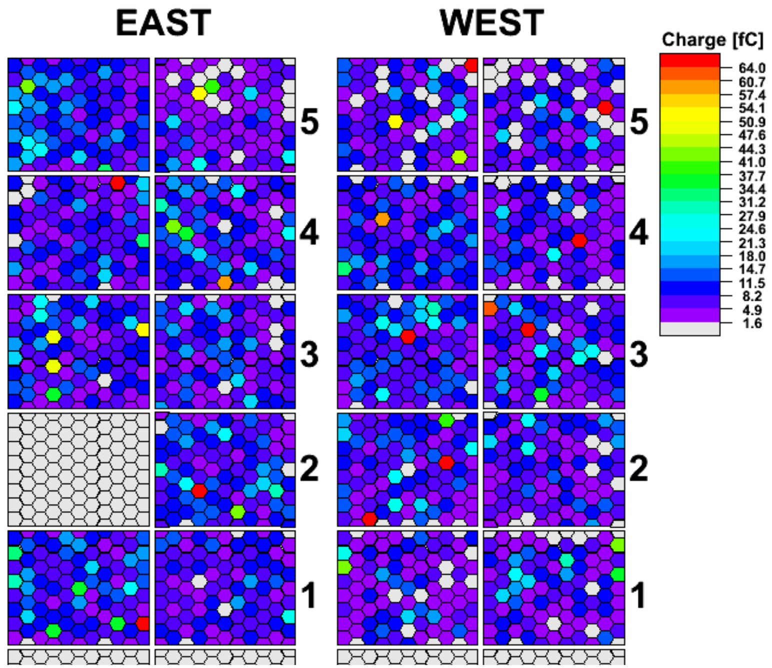




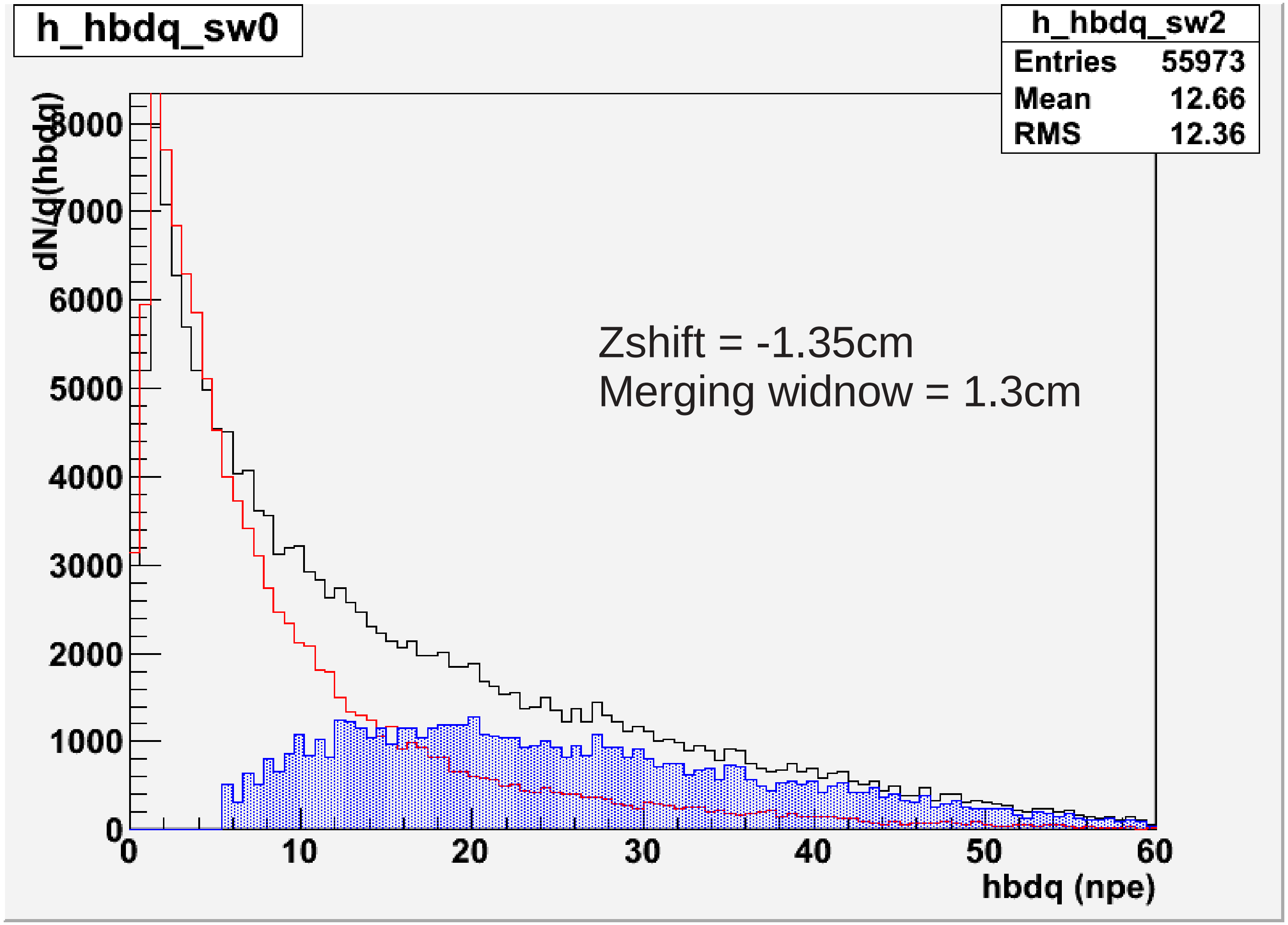
HIPs: an issue with a solution

- The pad by pad charge distribution has a very long tail
  - Caused by physics processes that deposit a huge amount of energy
  - Much more than typical per pad charge expected from either scintillation or Cerenkov
  - Rate is proportional to intensity
  - X-ray, neutrons heavy particles?
- These pads if left alone are a big problem for any clusterization algorithm, because they can seed fake clusters.
- Fortunately, event by event, they cover only a very small fraction of the active HBD area





Charge distribution for projection based merging



- Subtracted distribution (electrons that leave a hit) has a higher charge than the swapped (random clusters)
- But there is still some contamination

Rejection of strut conversions