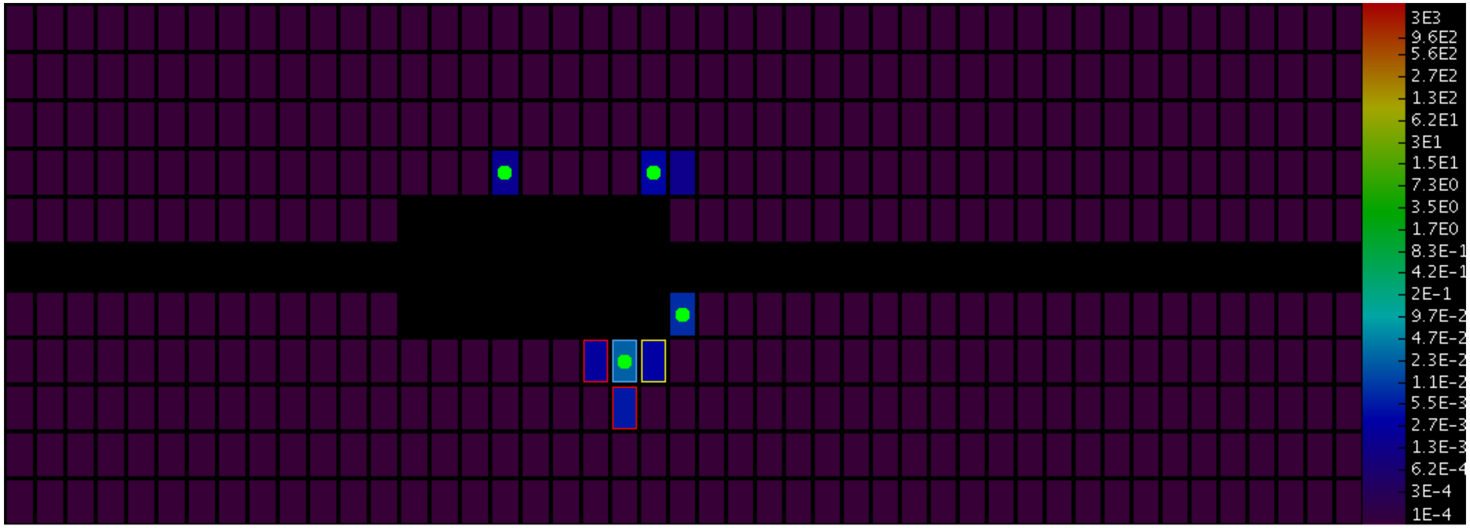
Ecal Clustering: Implementation of CLAS IC algorithm

06 March 2014

Clustering Method

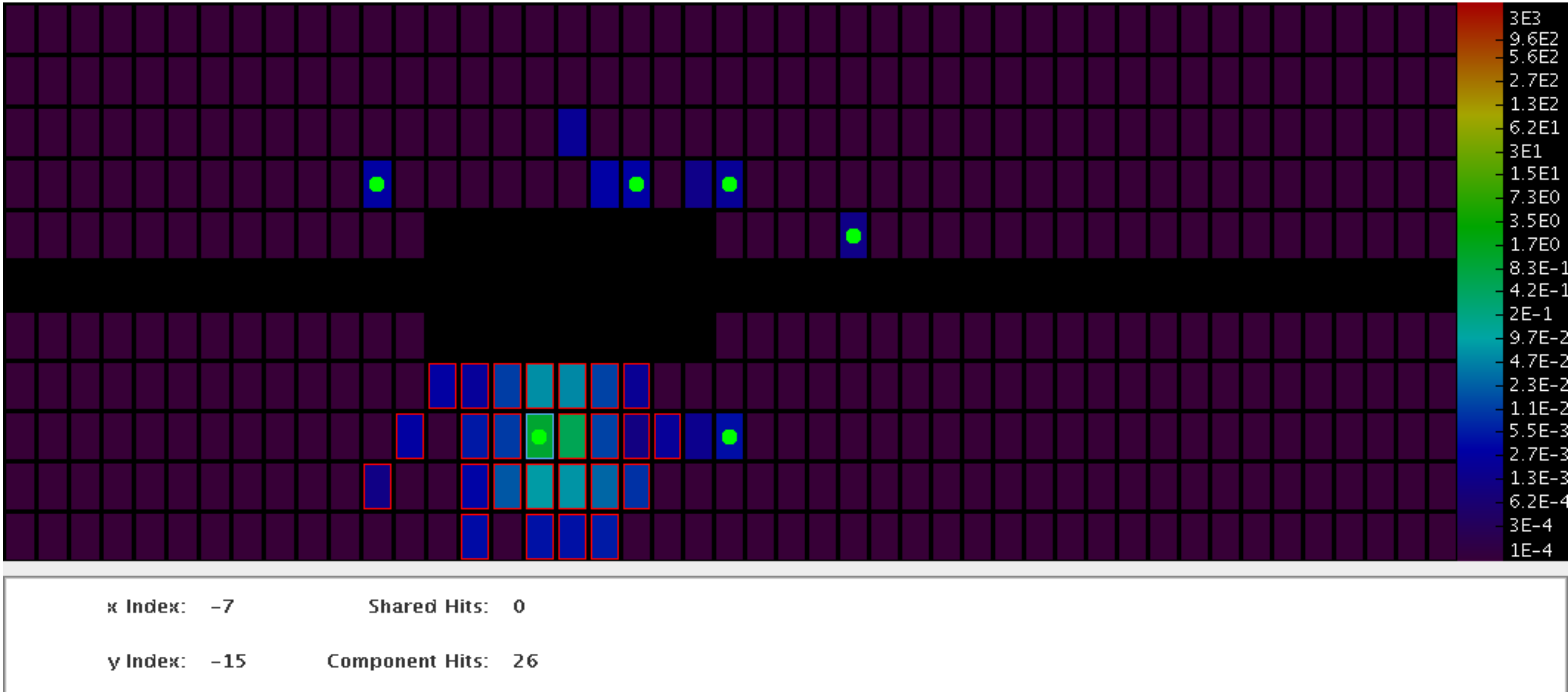
1. Sort all hits from highest energy to lowest energy
2. Loop over hit list:
    1. If: local maxima-> “Seed”
    2. Else: Neighbor to seed, add to seed cluster (if neighbor to two seeds, then add to common hit of both clusters). Otherwise, ignore



Clustering Method

3. Build remainder of clusters

    -Go through neighbors of cluster hits, adding to cluster if less energy



x Index: -7 Shared Hits: 0

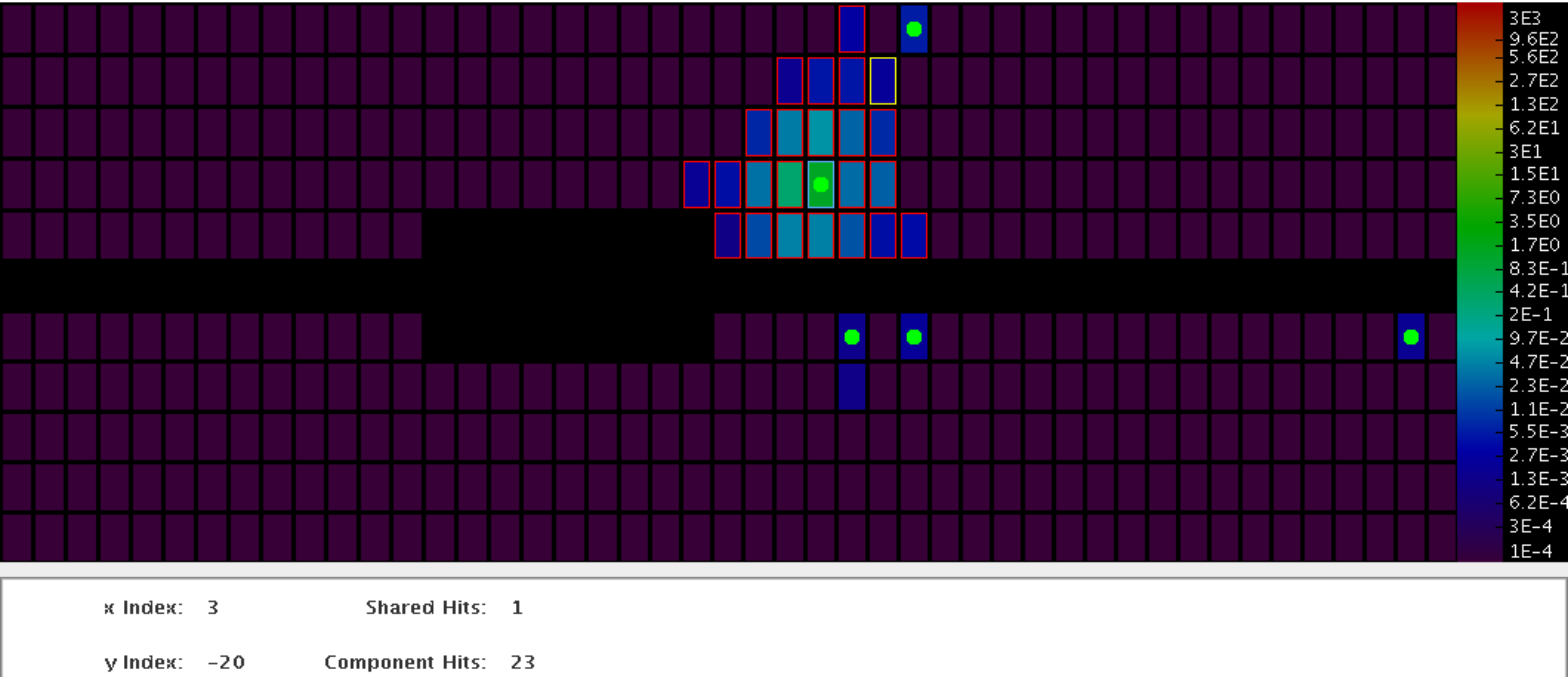
y Index: -15 Component Hits: 26

Clustering Method

4. Common hits

-Lie on borders of two clusters (have neighbors belonging to different seeds)

- Less energy than some surrounding hits



x Index: 3 Shared Hits: 1

y Index: -20 Component Hits: 23

Cluster Energy

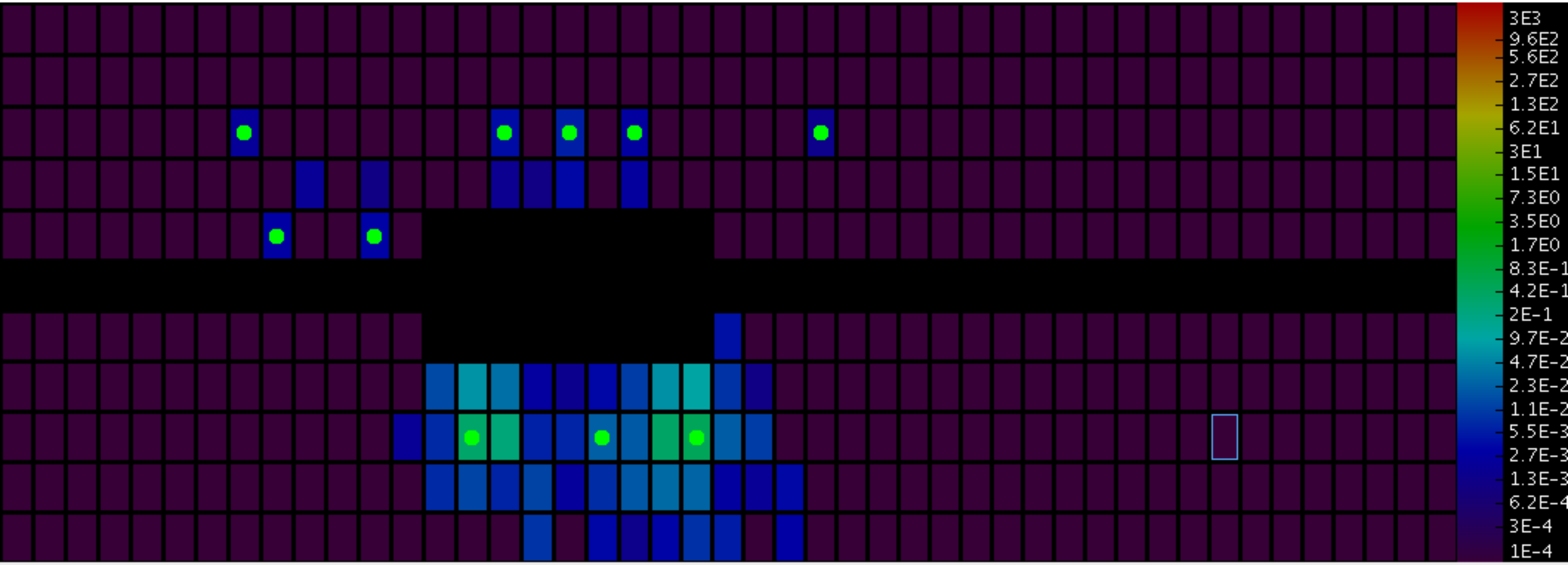
- Energy of cluster includes all cluster hits (excluding common hits)
- Add back in energy from Common Hits:

$$E_i = \frac{E_A}{E_A + E_B} E_{CH}$$

E_A = Energy of Cluster A
E_B = Energy of Cluster B
E_{CH}=Energy of common hit
E_i = Energy contribution of common hit to cluster (A in this case)

Shortcomings

- Does not account for more than two seeds in hit cluster
- Do we want to keep 3x3 around seed or can this become common hit?



Next Step and Questions

- Implement into recon driver format (input arguments), energy cut option
- Sampling fraction
  - Dependent on location of Ecal that the seed hit occurs
  - Energy distributions fitting->What percent can we reasonably reconstruct that is lost on edges?
  - What MC data should I be using?
- Why does test data only give 1-2 crystals in clusters?