Structural PFA: Status & Results

Mat Charles
The University of Iowa

Overview

- Structural algorithm: Recap & code status
- Tools I use to assess the PFA
- Structural PFA performance
  - What works well
  - What needs improving

All plots: Z-pole events in sidaug05

Structural Algorithm: Recap

- Previously described at Snowmass
- Hadronic clusters are composed of
  (1) Track segments
  (2) Dense clumps
  (3) Halo
  (4) Secondaries (neutral)
- Find these components and link them
  – Likelihood selector for (1) and (2)
  – MST for (3)
  – Special fragment handling for (4)

Fragment Handling

- Two parts:
  - (1) Is this cluster a fragment?
  - (2) Which is the parent cluster of this fragment?
- Current simplistic implementation:
  - (1) Fragment ID:
    - Not a fragment if there's an associated track. Else:
    - Fragment if < 4 hits
    - Fragment if 4-10 hits and DOCA to IP > 10 cm
  - (2) Fragment assignment:
    - Merge with nearest (hit-hit) non-fragment cluster

Code Status & Locations

- Code is in CVS:
  - org.lcsim.recon.cluster.mipfinder
  - org.lcsim.recon.cluster.mst
  - contrib/uiowa/structural
- Package “structural” is the structural algorithm plus an example PFA that uses it
- First two are stable; hopefully finalize interface for structural package at this workshop

Tools for studying the PFA

- Many pieces in a PFA
- Need to find & improve the weak links
- Some tools I use:
  - Modular design
    - Cheating can be turned on & off in steps
    - Plots for individual steps (e.g. fragment ID)
  - Check amount of charged/neutral confusion
  - Simple toy MC (described shortly)

Some Modules

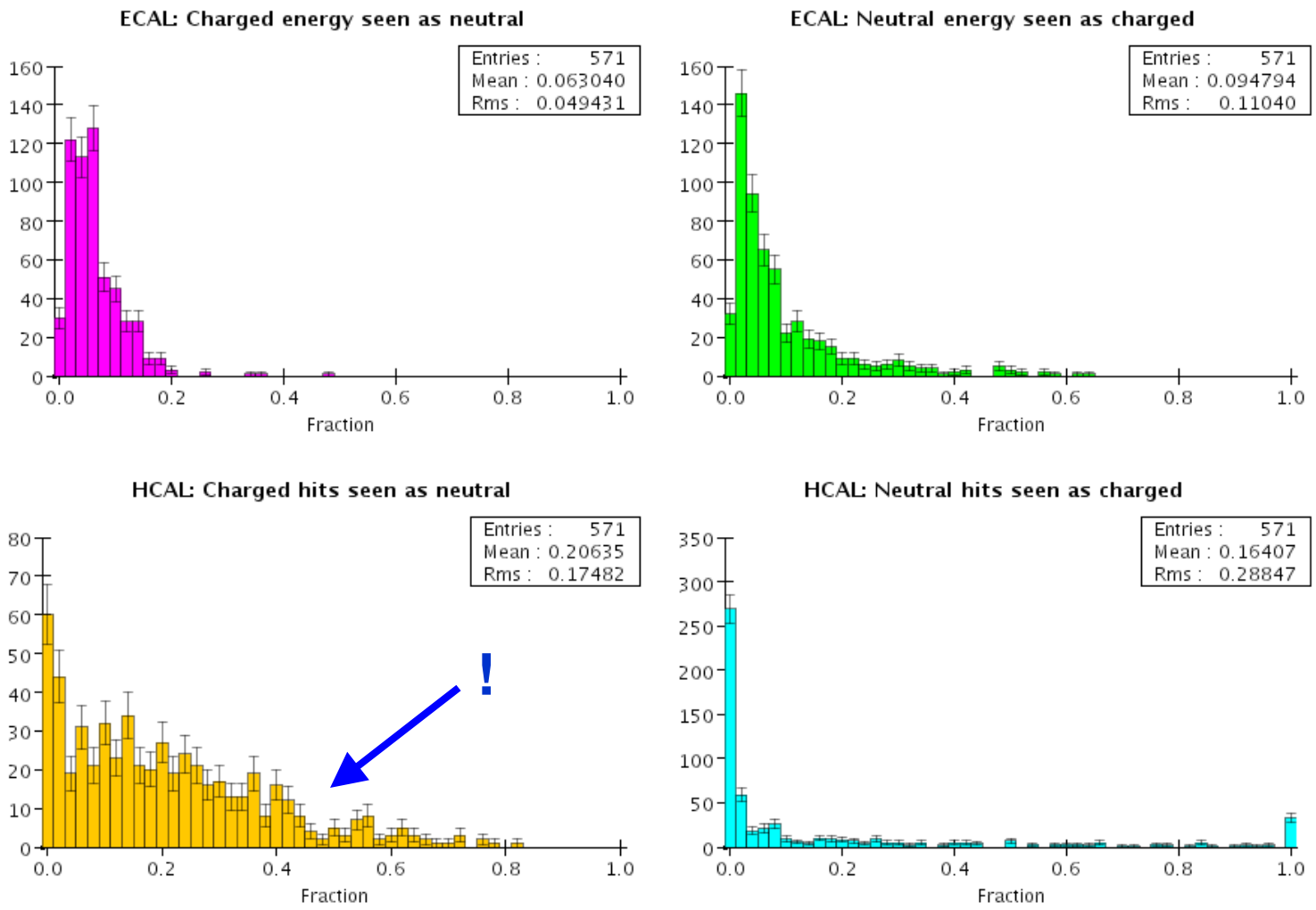
- FragmentIdentifier interface
  - SimpleFragmentIdentifier
  - CheatFragmentIdentifier
  - TestFragmentIdentifier
- FragmentMerger driver
  - CheatFragmentMerger extension
  - TestFragmentMerger extension
- EventEnergySum driver

Confusion

- Confusing charged and neutral energy degrades resolution.
- Three categories to describe the hits:
  - (1) From a charged particle with a found track
  - (2) From a charged particle without a track
  - (3) From a neutral particle
- Measure how much of each type winds up in clusters with
  - (a) A matched track (safe for 1; wrong for 2,3)
  - (b) No matched track (safe for 2,3; wrong for 1)

Example results

No cheating; simple fragment ID & assignment



A problem identified: charged energy in the HCAL seen as neutral

HCAL: Charged → neutral energy

- We found a track at the ECAL surface
- HCAL hits in separate cluster (fragment)
- Possible issues:
  - (1) We mis-identified it as a primary neutral
  - (2) We identified it as a fragment but attached it to a wrong (neutral) cluster
  - (3) Should it have been part of the primary cluster in the first place?
  - (4) Track misassigned?

Fragment Performance

- Fragment ID:
  - 99% of fragments in ECAL correctly ID'd
  - 93% of fragments in HCAL correctly ID'd
  - 68% of primaries in ECAL correctly ID'd
  - 81% of primaries in HCAL correctly ID'd
- Fragment merging:
  - 42% of fragments in ECAL assigned to parent
  - 72% of fragments in HCAL assigned to parent

Merging problems? But not conclusive…

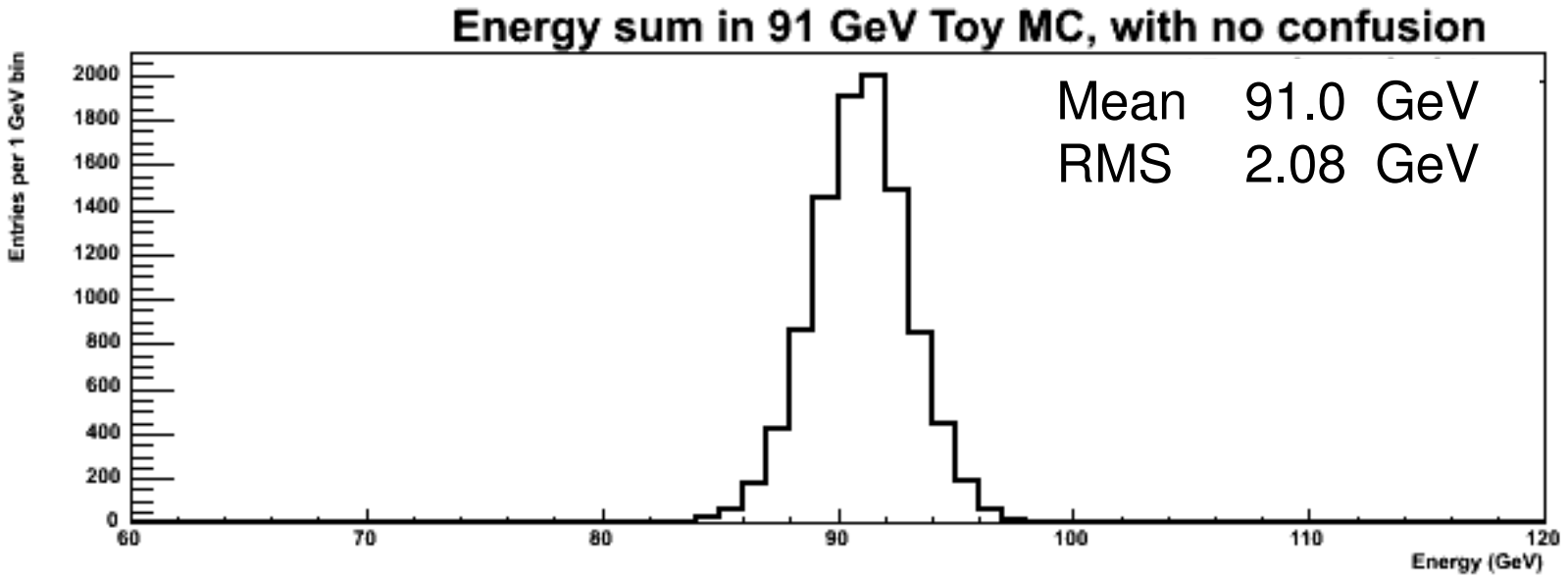
Building A Simple Toy MC

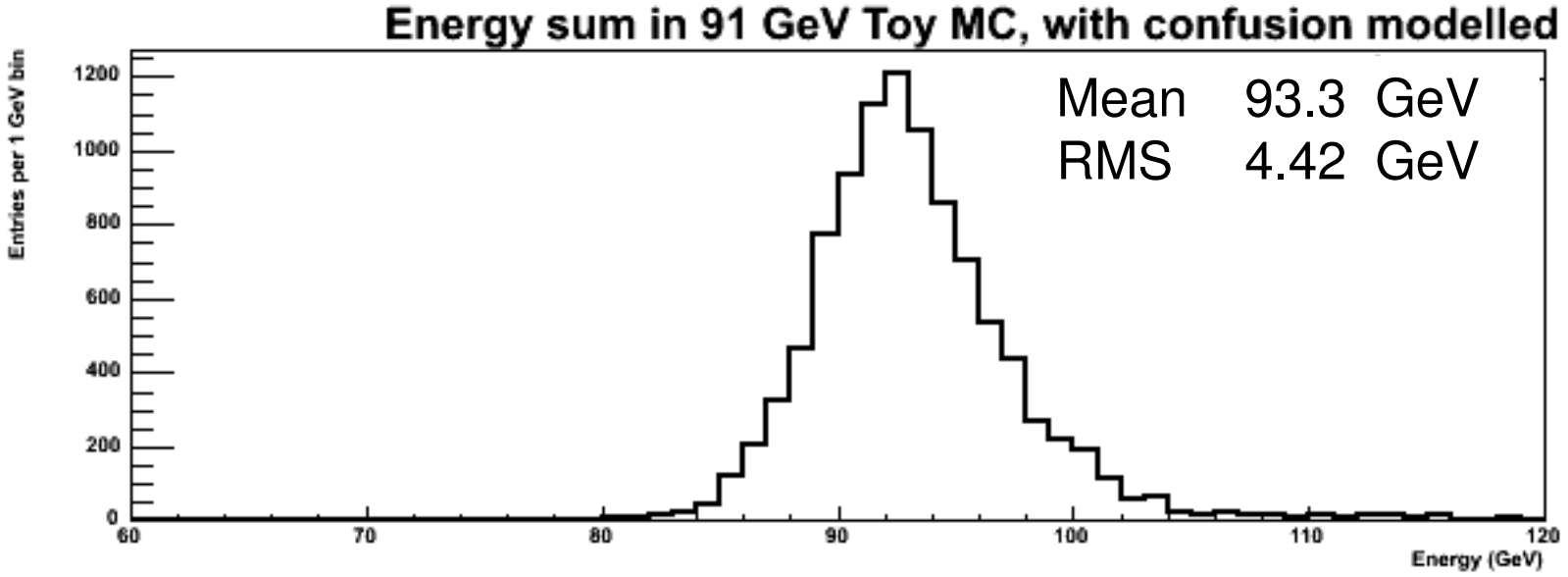
- Goal is to simplify the problem so I can study PFA performance quantitatively
- Relevant parts of resolution for clustering:
  (1) Confusion term
  (2) Fragment ID if calibration is non-linear
- Use confusion plots as input for (1)
- Ignore (2) for the moment

Building A Simple Toy MC

- Other inputs: amount and resolution for
  - E/M neutrals (photons, pi0) [20%/√E]
  - E/M charged (electrons)
  - Hadronic neutrals (K0, n, …) [68%/√E]
  - Hadronic charged (pi+, p, … -- also muons)
- Total visible energy is fixed to 91.0 GeV
- … and throw a lot of dice

Example Output Plots





Toy MC Results

	Mean (GeV)	RMS (GeV)
No confusion	91.0	2.04
Cheating on fragment ID and fragment assignment	91.5	2.36
Cheating on fragment ID but not fragment assignment	93.0	3.52
Not cheating at all	93.3	4.42

Not cheating on cluster core reconstruction, track matching etc. to get confusion PDFs

Subtracting in quadrature:
Bad ID: 2.6 GeV
Bad merging: 2.7 GeV

Another Approach To Fragments

- Suggested by Ron…
- We identify fragments like before…
- … but instead of trying to merge them with their parent, we drop them instead
- In effect, trade off hit efficiency for purity
- No need to do fragment assignment

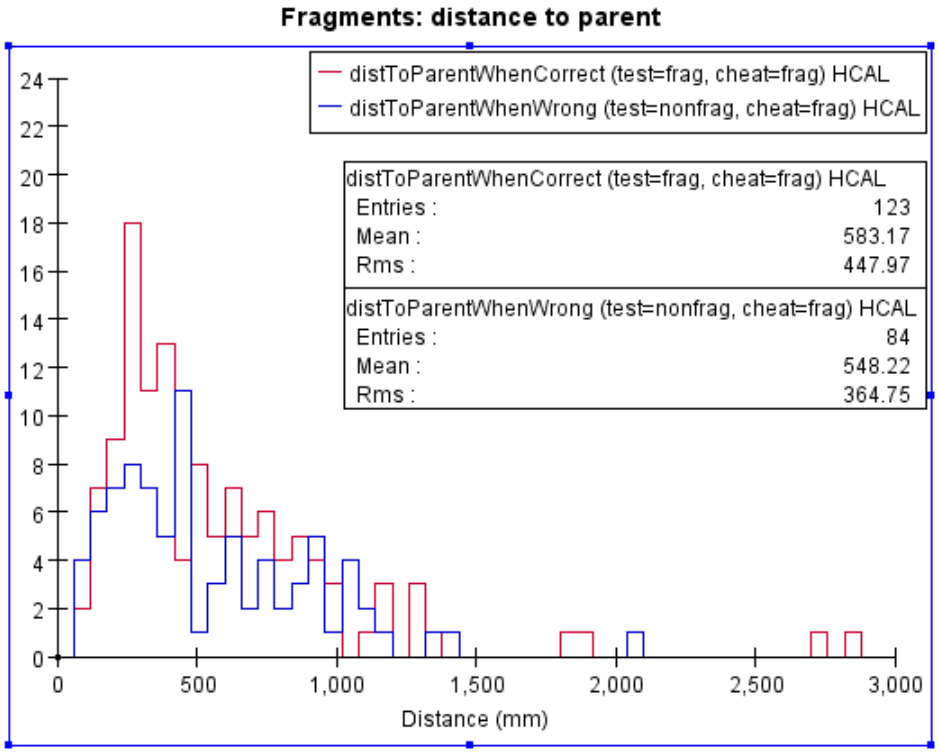
- Preliminary and wrong study shows it might just about work IF we have really good fragment ID.

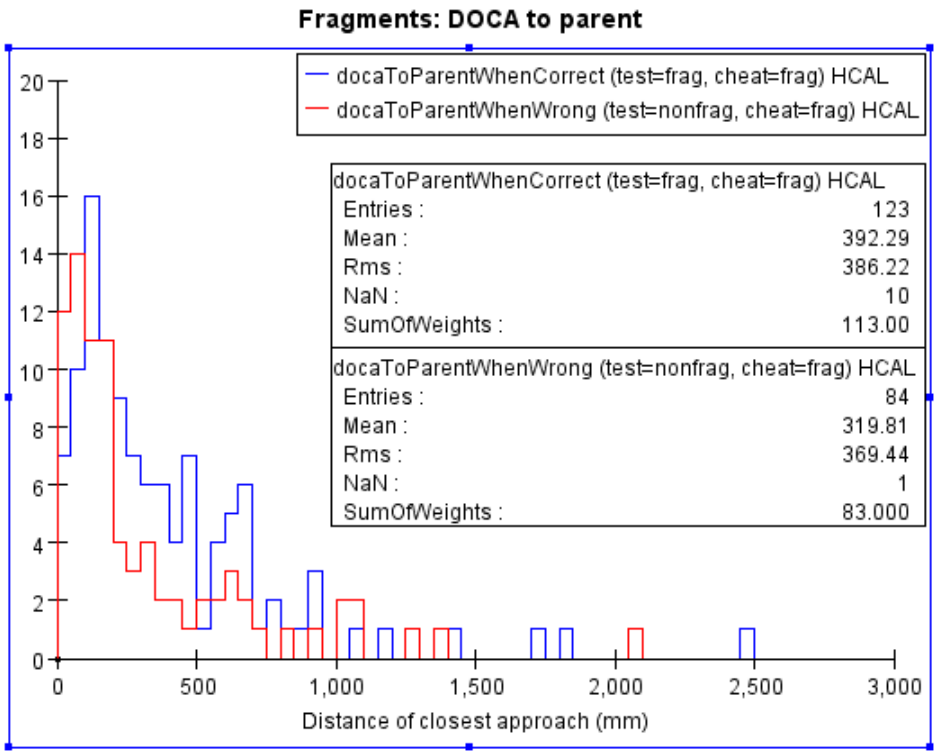
Conclusions & Thoughts

- Need better fragment handling!
  - I have a few ideas for ID
  - Merging seems to be a hard and non-scaleable problem…
  - Keen to try out other people's algorithms
- Does track-matching need improving?
- Make use of E/p?
- Are the resolutions from toy MC achievable in real life?
- Better modelling of fragment-dropping?

Backups

Fragment Merging in HCAL

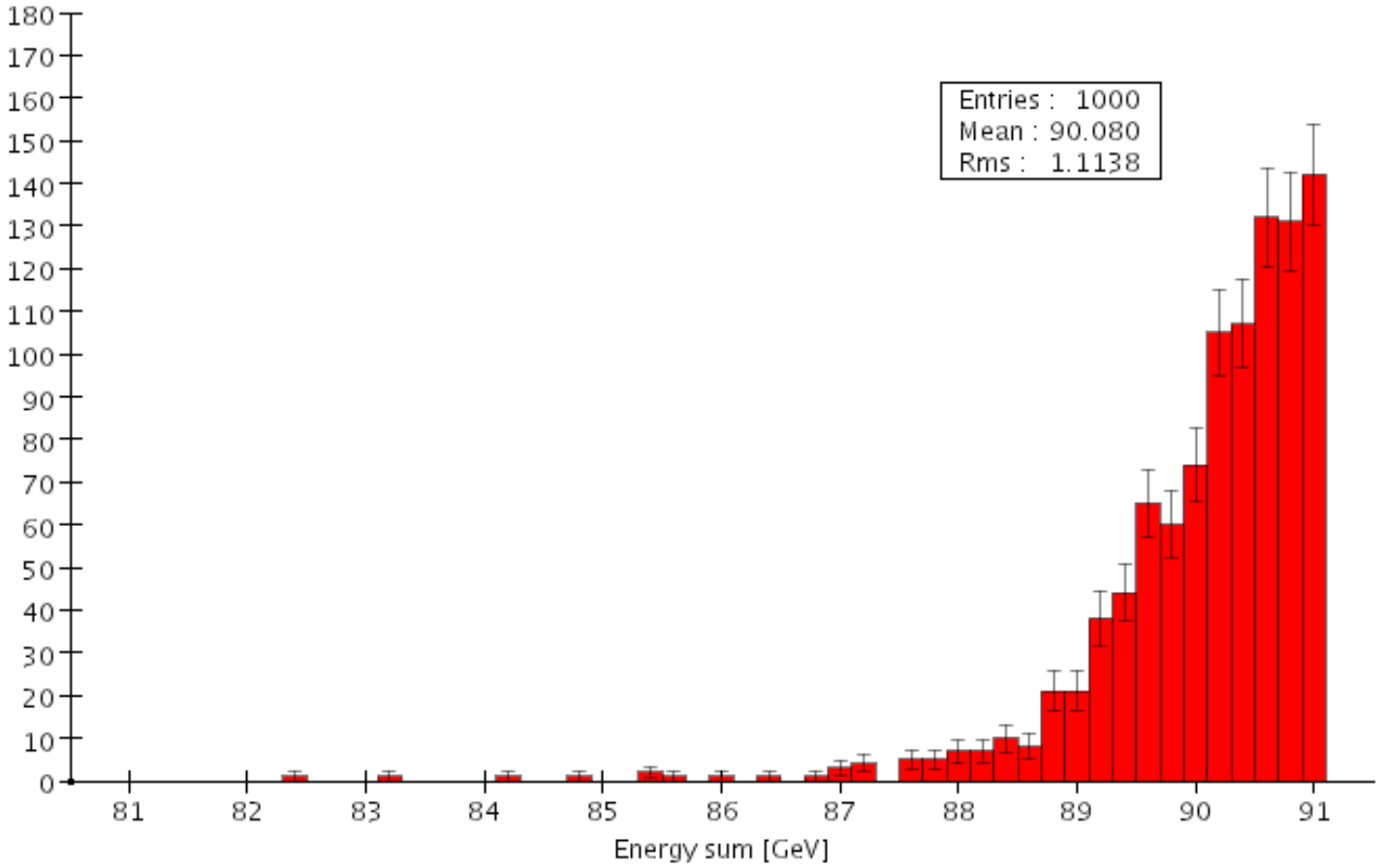




Toy MC Inputs

Type	Energy fraction	Resolution
Charged hadrons	0.659 ± 20% relative	0 (charged)
Neutral Hadrons	0.083 ± 50% relative	68% / sqrt(E)
Electrons	0.040 ± 30% relative	0 (charged)
Photons	0.218 ± 30% relative	20% / sqrt(E)

Example: Missing Energy



- Perfect PFA, counting everything that reaches the calorimeters
- Energy taken from truth (i.e. zero resolution)
- Neutrinos added in

Actual Energy Sum Plot