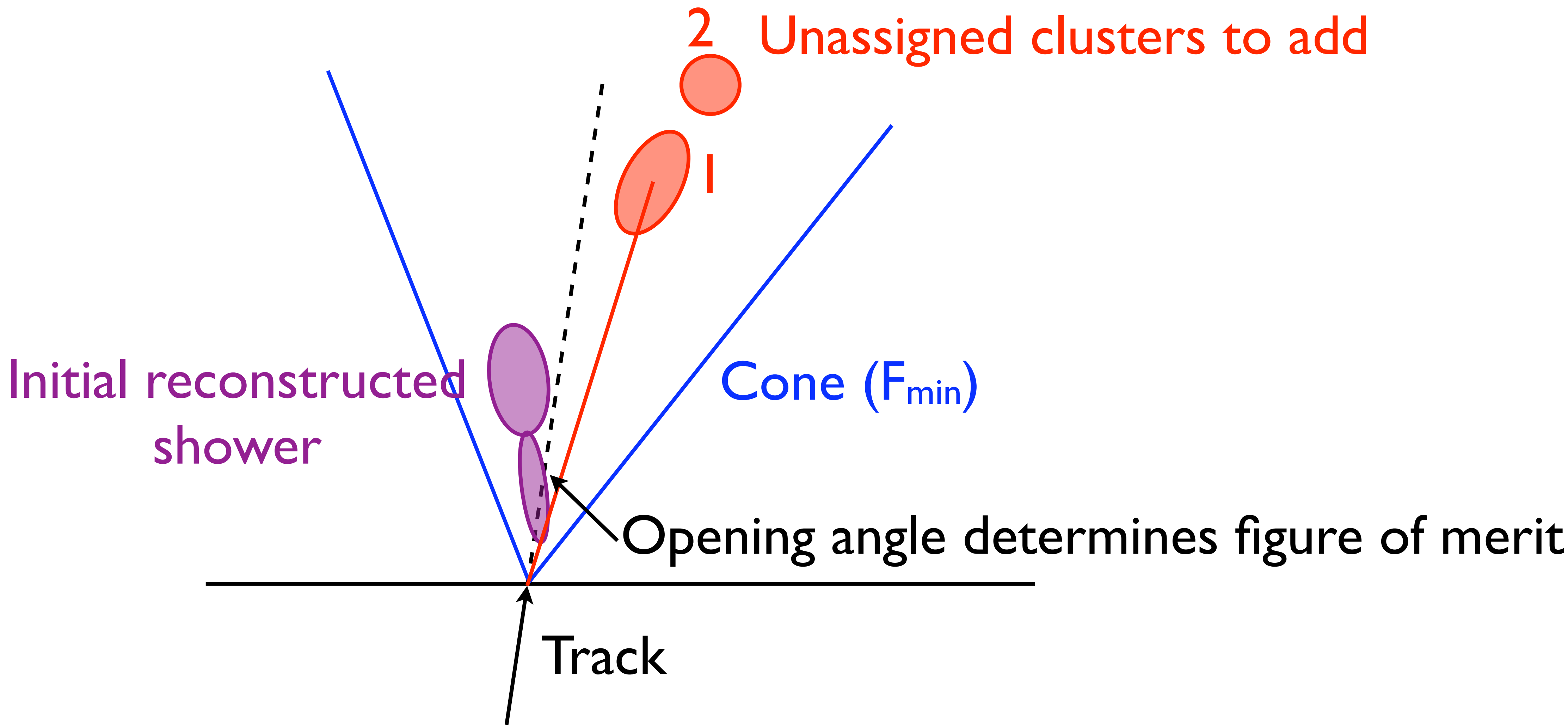
Update on PFA development at Iowa

Mat Charles
Tae Jeong Kim
Usha Mallik

What we're working on (1)

- After the reconstruction pass, we often have charged showers where E_{cal} << $|p_{track}|$ and no evidence of punch-through.
- Original handling:
  - "Obviously, something was missed..."
  - ... therefore part of shower elsewhere...
  - ... therefore to avoid double-counting, least biased method is to ignore p_{track} and assign it as a hadron of energy E_{cal}.
- Added step to try to recover missing energy:
  - Look at clusters which are not assigned to another track and are not photons
  - Assign each a figure of merit F for consistency with track
  - Add clusters in order of increasing F...
  - ... until hit limit (F_{min}) or E_{cal} goes too large for $|p_{track}|$
  - Abort if run into huge cluster (E<<p to E>>p in one step)
  - Abort if fail to satisfy E/p within errors (or find punch-through)
- Currently using simple cone for F (next slide)
- Better figure of merit in the works (later slide)

Simple cone reassignment


2
Unassigned clusters to add
1
Initial reconstructed shower
Cone (F_{min})
Opening angle determines figure of merit
Track

Very easy, very simple -- and improves performance.

		sid01	sid01_scint
Before	With E/p veto	-2.28 ± 4.45	-3.94 ± 4.15
	No E/p veto	-0.40 ± 5.00	-1.88 ± 4.92
After	With E/p veto	-2.94 ± 4.33	-4.52 ± 4.15
	No E/p veto	-1.99 ± 4.31	-3.40 ± 4.07

E/p veto no longer needed! We've effectively replaced it with the reassignment step.

What we're working on (2)

- Simple cone algorithm was proof-of-principle...
- ... but real showers don't look like that.
- Tae Jeong now working on better figure of merit based on cone starting at showering point of cluster
  - Similar to algorithm used by Steve (also suggested by Ron)
  - Envelope should catch most of shower.

    (Remember -- we've already done the first-pass shower reconstruction so we don't need to catch everything in this one step, just the missing pieces.)
  - Not finalized yet! Details are important here.
- If this works well, plan to extend its use:
  - Look for other clusters linked to those picked up by cone algorithm?
  - Consider also clusters that are currently assigned to another track.
  - Use as part of cluster sharing algorithm for halo hits & small clusters
  - Include as another type of cluster-cluster link in earlier reconstruction pass (effectively picking narrow cone up earlier in clustering)
- Return to skipped/hard cases later:
  - If find huge cluster, break it up instead of giving up.
  - Sometimes pick up "photon" clusters if no other option

Other improvements & fixes

Various improvements to code:

- Some code cleanup & refactoring
- Allow projective (easy) MIP clusters to cross between endcap & barrel subsystems and to span more than one DTree cluster
- Fix handling of punch-through to include sides of HCAL as well as back
- Be explicit about one-sided vs two-sided E/p tests
- Improve handling of halo hits for large clusters with internal structure:
  - share hit-by-hit rather than as a giant blob
  - different scale parameters for ECAL and HCAL
  - separate from handling of small clusters w/o internal structure

		sid01	sid01_scint
Before cone reassignment	With E/p veto	-2.28 ± 4.45	-3.94 ± 4.15
	No E/p veto	-0.40 ± 5.00	-1.88 ± 4.92
After cone reassignment	With E/p veto	-2.94 ± 4.33	-4.52 ± 4.15
	No E/p veto	-1.99 ± 4.31	-3.40 ± 4.07
+ fixes & improvements	With E/p veto	-2.68 ± 4.34	-3.75 ± 4.11
	No E/p veto	-1.71 ± 4.29	-2.72 ± 3.99

Psychological barrier broken...

Further down the line

Recent discussions with Ron & Usha very helpful in identifying next steps/projects:

- Ron notes that shared clusters have pretty lousy purity AND comprise about half of HCAL hits! Need to be more careful about sharing... and look for ways to find more structure / break apart further.
- Hadrons that interact late in tracker are hard for current track matching -- identify & handle these specially?
- Need a better way to identify/arbitrate cases where one track steals clusters from another shower...
- ... second pass with a different algorithm?
- + various other ideas for individual pieces

Also general code cleanup to help make it easier for human beings to read.