# Track Reconstruction for the LOI

Richard Partridge
Brown / SLAC

October 10, 2008

# SiD Reconstruction Plans

- The current “wish list” is to fully simulate and reconstruct ~40M events using the SiD02 detector
  - These events will largely be used for benchmarking studies
  - Expect that there will be additional small samples (for example, with a planar tracker geometry) produced for tracking studies
- SLIC / GEANT4 simulation jobs are currently running at SLAC (and perhaps elsewhere?)
- Plan is to take a “snapshot” of the org.lcsim software around the end of October that will be used for reconstruction
  - Goal is to run PFA with full tracking provided performance is acceptable
  - From Matt’s results, it looks like the key tracking issue is to lower the $p_T$ cut below the 1 GeV value used for early tracking studies

# Track Reconstruction Decisions

- Parameters for virtual segmentation parameters
  - Segmentation dimensions
  - Tracker endcap stereo angle
  - Strip pitch / pixel size
- Tracking parameters
  - Minimum number of hits
  - Minimum $p_T$
  - maximum $|d_0|$ (distance of closest approach in the x-y plane)
  - maximum $|z_0|$ (z coordinate at the distance of closest approach)
  - Maximum $\chi^2$
  - Bad hit $\chi^2$ – threshold for logic that tries bypassing a layer with marginal hits
- Strategy list
  - This proceeds fairly automatically given the above cuts
  - Typically have required 7 hits - tried adding by hand a 6-hit barrel only strategy to pick up low $p_T$ central tracks, but this may have problems

# Virtual Segmentation Parameters

- Dima’s Example for the Barrel Pixels

```
CylindricalBarrelSegmenter vtxBarrelSegmenter = new CylindricalBarrelSegmenter("VertexBarrel");
vtxBarrelSegmenter.setStripLength(25.*SystemOfUnits.micrometer);
vtxBarrelSegmenter.setStripWidth(25.*SystemOfUnits.micrometer);
setSegmenter("VertexBarrel", vtxBarrelSegmenter);
```

- Dima’s Example for the Endcap Pixels

```
DiskTrackerToRingsSegmenter vtxEndcapSegmenter = new DiskTrackerToRingsSegmenter("VertexEndcap");
vtxEndcapSegmenter.setStripLength(25.*SystemOfUnits.micrometer);
vtxEndcapSegmenter.setStripWidth(25.*SystemOfUnits.micrometer);
setSegmenter("VertexEndcap", vtxEndcapSegmenter);
```

- Are Forward Pixels the same as the Endcap Pixels?

```
DiskTrackerToRingsSegmenter trackerForwardSegmenter = new DiskTrackerToRingsSegmenter(“TrackerForward");
trackerForwardSegmenter.setStripLength(25.*SystemOfUnits.micrometer);
trackerForwardSegmenter.setStripWidth(25.*SystemOfUnits.micrometer);
setSegmenter(“TrackerForward", trackerForwardSegmenter);
```

# Virtual Segmentation Parameters

- Dima’s example for Barrel Strips

```
CylindricalBarrelSegmenter trackerBarrelSegmenter = new CylindricalBarrelSegmenter("TrackerBarrel");
trackerBarrelSegmenter.setStripLength(10.*SystemOfUnits.cm);
trackerBarrelSegmenter.setStripWidth(25.*SystemOfUnits.micrometer);
setSegmenter("TrackerBarrel", trackerBarrelSegmenter);
```

- Dima’s example for Endcap Strips

```
DiskTrackerToWedgesSegmenter trackerEndcapSegmenter = new
DiskTrackerToWedgesSegmenter("TrackerEndcap");
trackerEndcapSegmenter.setNumberOfRadialSlices(new int[]{3,5,8,10, 10});
trackerEndcapSegmenter.setStripWidth(25.*SystemOfUnits.micrometer);
trackerEndcapSegmenter.setNumberOfPhiSlices(24);
setSegmenter("TrackerEndcap", trackerEndcapSegmenter);
```

# Tracking Parameters

- Minimum number of hits
  - Have been using 7 for code development
- Minimum $p_T$
  - Have been using 1 GeV for code development
  - Based on PFA results, goal is to push $p_T$ cut down to 200 MeV
- Maximum $|d_0|$ (DCA in the x-y plane)
  - Have been using 10 mm for code development
- Maximum $|z_0|$ (z coordinate at the DCA)
  - Have been using 10 mm for code development
- Maximum $\chi^2$
  - Have been using 50 for code development
- Bad hit $\chi^2$ – threshold for bypassing a layer with bad hits
  - Have been using 15 for code development

Proposal: focus on pushing down $p_T$ cut, leave other cuts alone

# Reducing the Minimum $p_T$ Cut (Tues)

- Matt and Ron have shown that the 1GeV $p_T$ cut is the primary source for degradation of PFA resolution with full tracking
- Goal is to push the tracking $p_T$ cut to 200 MeV
- Since the strategies were created for the 1 GeV cut, re-ran the strategy builder to generate new strategies using a ttbar sample
  - First pass nearly doubled the number of strategies – Yikes!
  - With help from Cosmin, turned off the inefficiency option – 26 strategies
  - Small increase from old strategy file (20 strategies) may be due in part to use of ttbar training sample instead of a Z pole sample
  - Some of these strategies may be adding negligibly to the efficiency – changing the cut on where to stop reduced the number of strategies to 19
  - Cosmin is modifying the strategy builder to identify the incremental number of tracks found by each strategy added

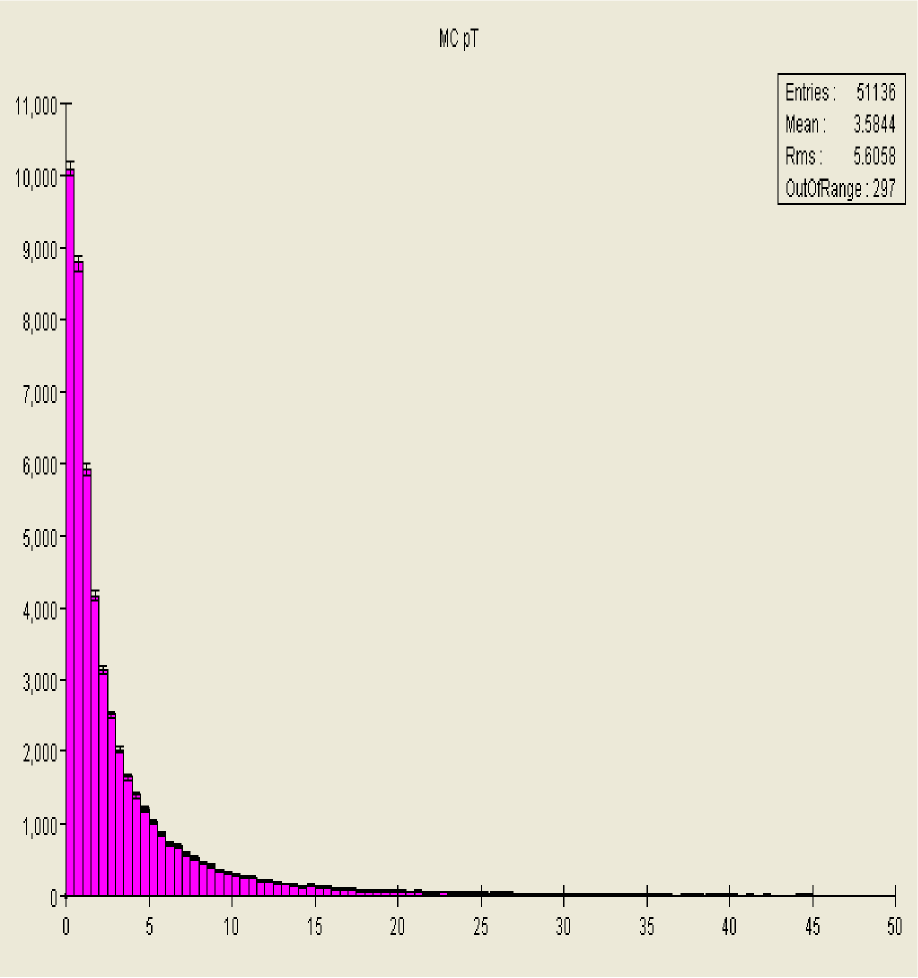
# Initial Results (Tues)

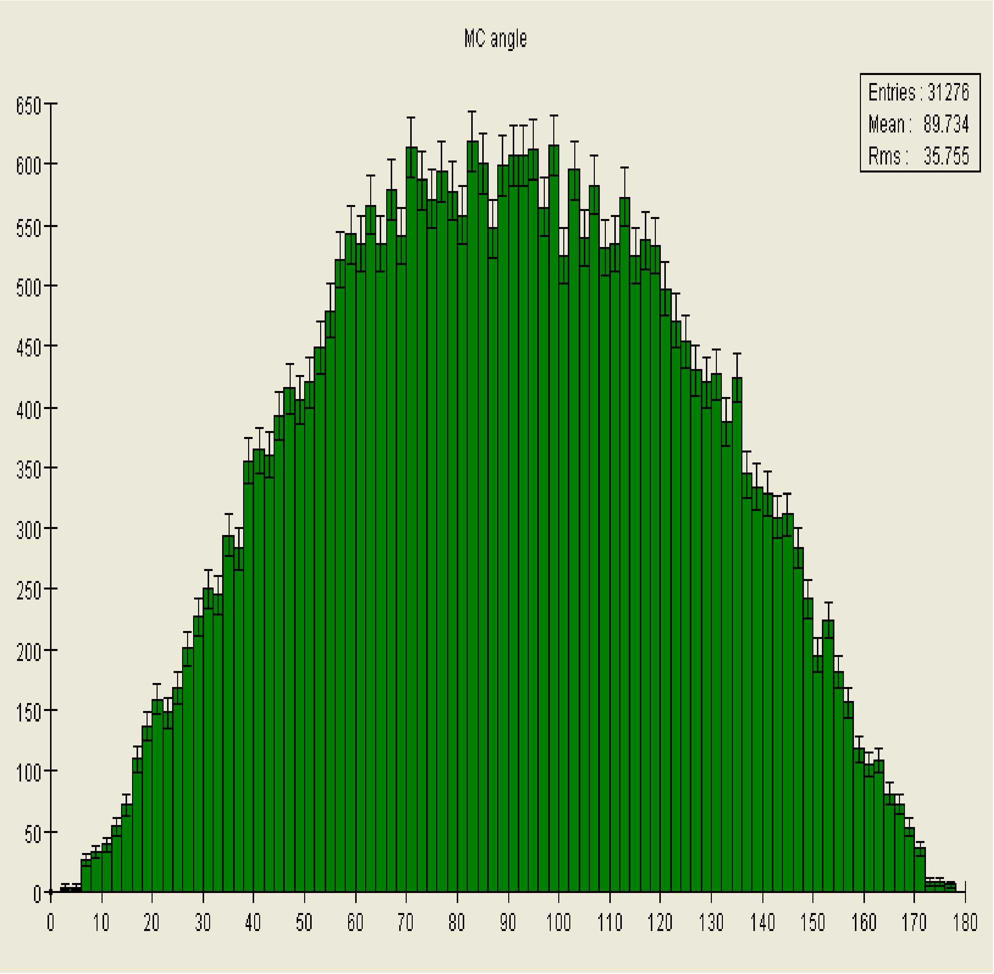
- Tracking is slowed down by a large factor (>10)
- A known memory issue has re-appeared
  - We think we know how to fix this – not a big deal
- Next step: back off on $p_T$ cut from 200 MeV cut to 500 MeV cut and try to get back to a reasonable performance level
  - Past profiling has shown most of the time is spent in calculating MS errors – significant speedup is possible by not recalculating these errors at every iteration

# Latest Results

- Re-structured code to reduce memory usage
  - Perform the merging algorithm that selects the best track candidate each time we find a new track candidate
  - Previously, this was done after all track candidates have been found
  - Associated changes may also have improved performance
- Good results with restructured code
  - Memory usage is reasonably stable (154 MB max in JAS memory box)
  - Ran through an entire file of tt events with a 200 MeV $p_T$ cut
  - Code is noticably faster: average 15 sec/evt for a 200 MeV $p_T$ cut
- Working on speeding up multiple scattering code
  - Track is refit each time we try adding a hit
  - Currently the code calculates MS errors right before doing the helix fit
  - Track fitter is typically called repeatedly with all but one “new” hit unchanged
  - Should be much more efficient to do MS calculation for unchanged hits only one time (after the track fitting, before trying to add a new hit)
  - Hope to finish this up later today

# Tracks Distributions in the tt Sample



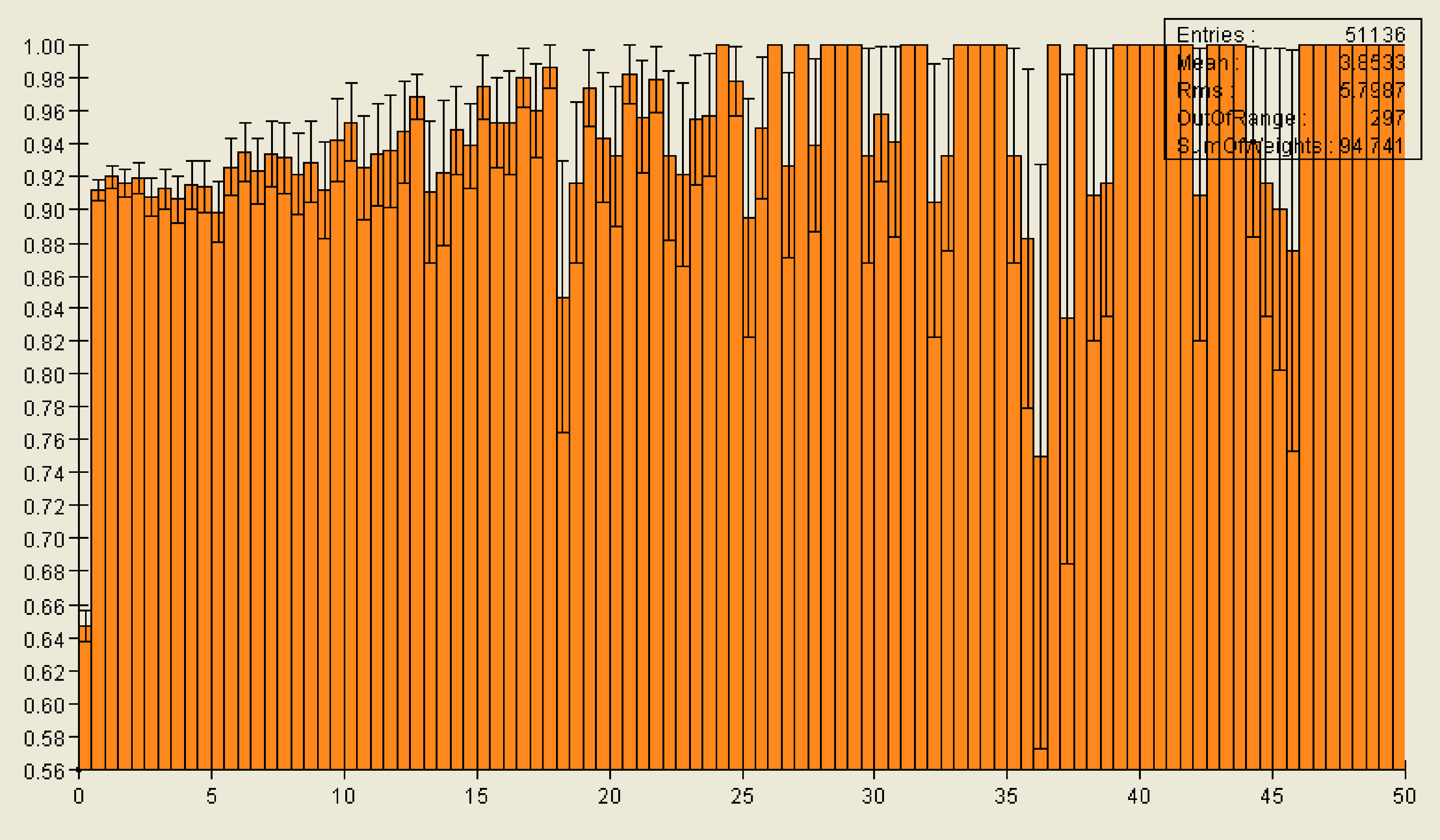


$p_T$

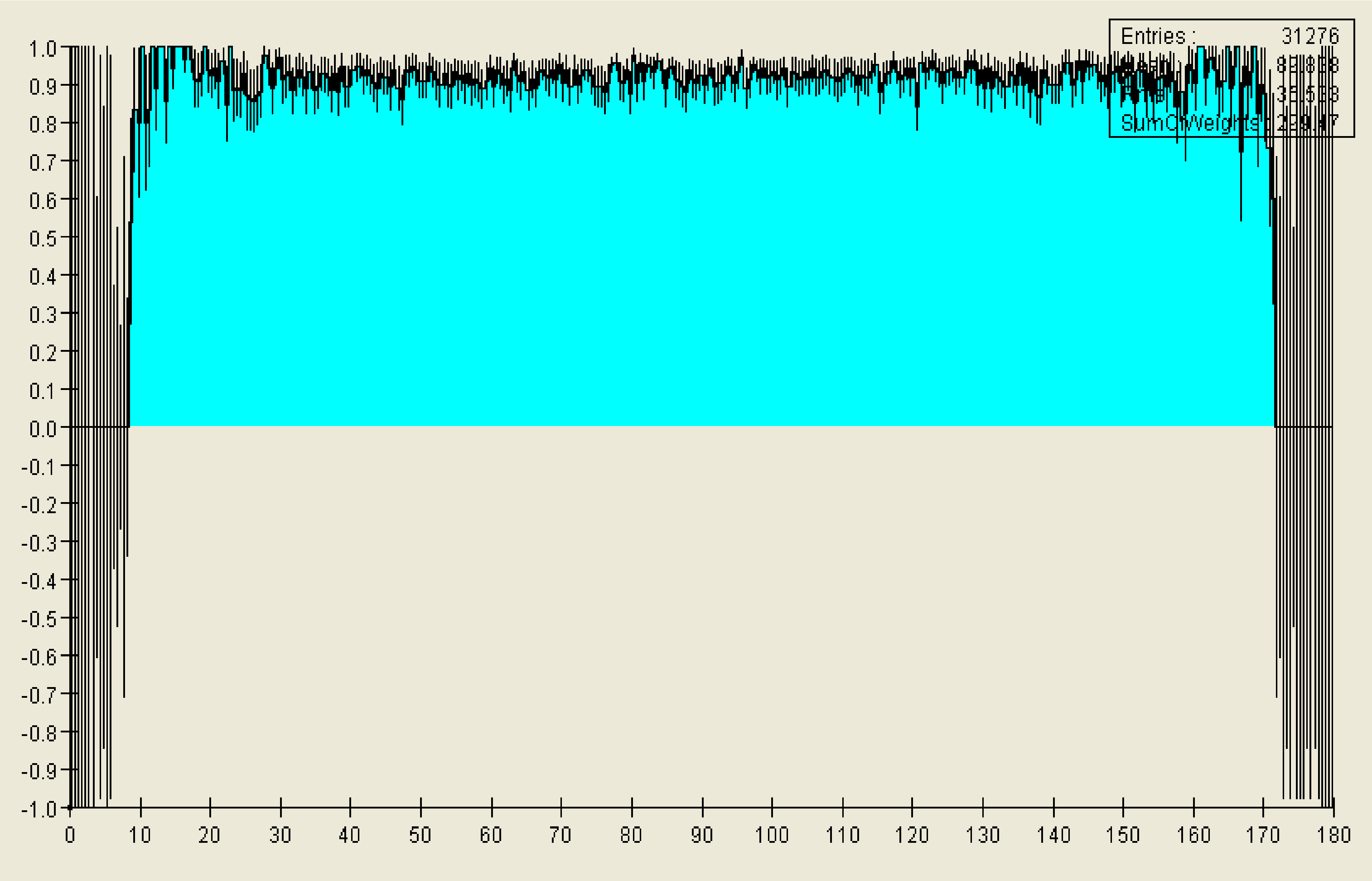
$\theta$

# Efficiency vs $p_T$

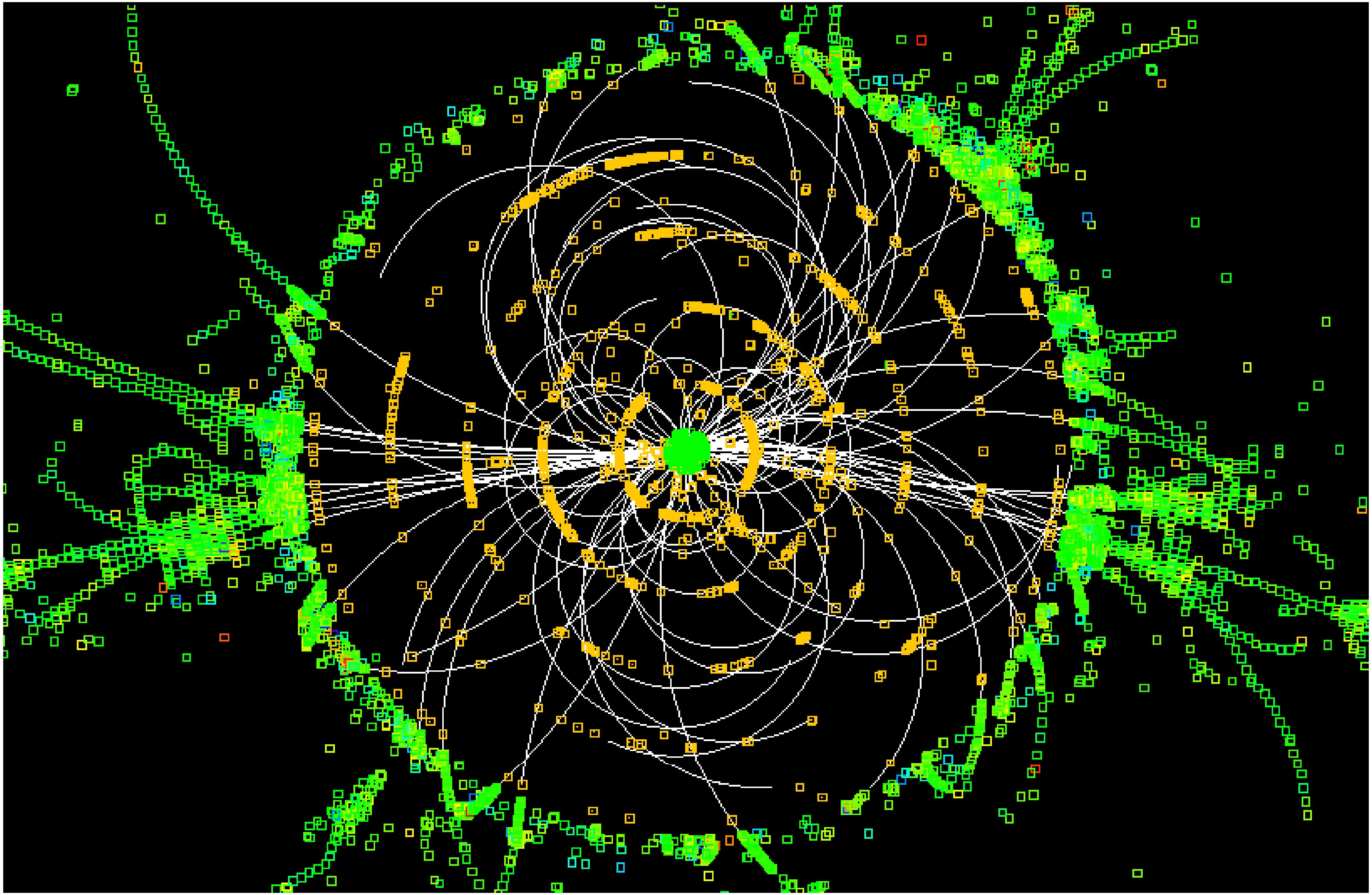
- Most of the inefficiency is due to non-prompt tracks



# Efficiency vs θ



# Example of a tt Event with >1700 Hits



# Some Problem with 6-Hit Tracks