Standalone Tracking in the Outer Barrel

Tim Nelson

SLAC

Why?

An SiD sore point: what if VXD is beset by backgrounds/problems?

Can standalone in outer tracker work?

- How many layers would be required?
- Does it require stereo?
- How much does module length matter?
- Do we even need to consider such a scenario?

Damn the controversy: give it a try!

Reconnaissance

- org.lcsim now has good facilities for getting SimTrackerHit information needed for this
- org.lcsim has long had a simple circle fitter (V. Karamaki port)
- **Try simple algorithm in the barrel and see what happens...**

Standalone Tracking “v0.1”

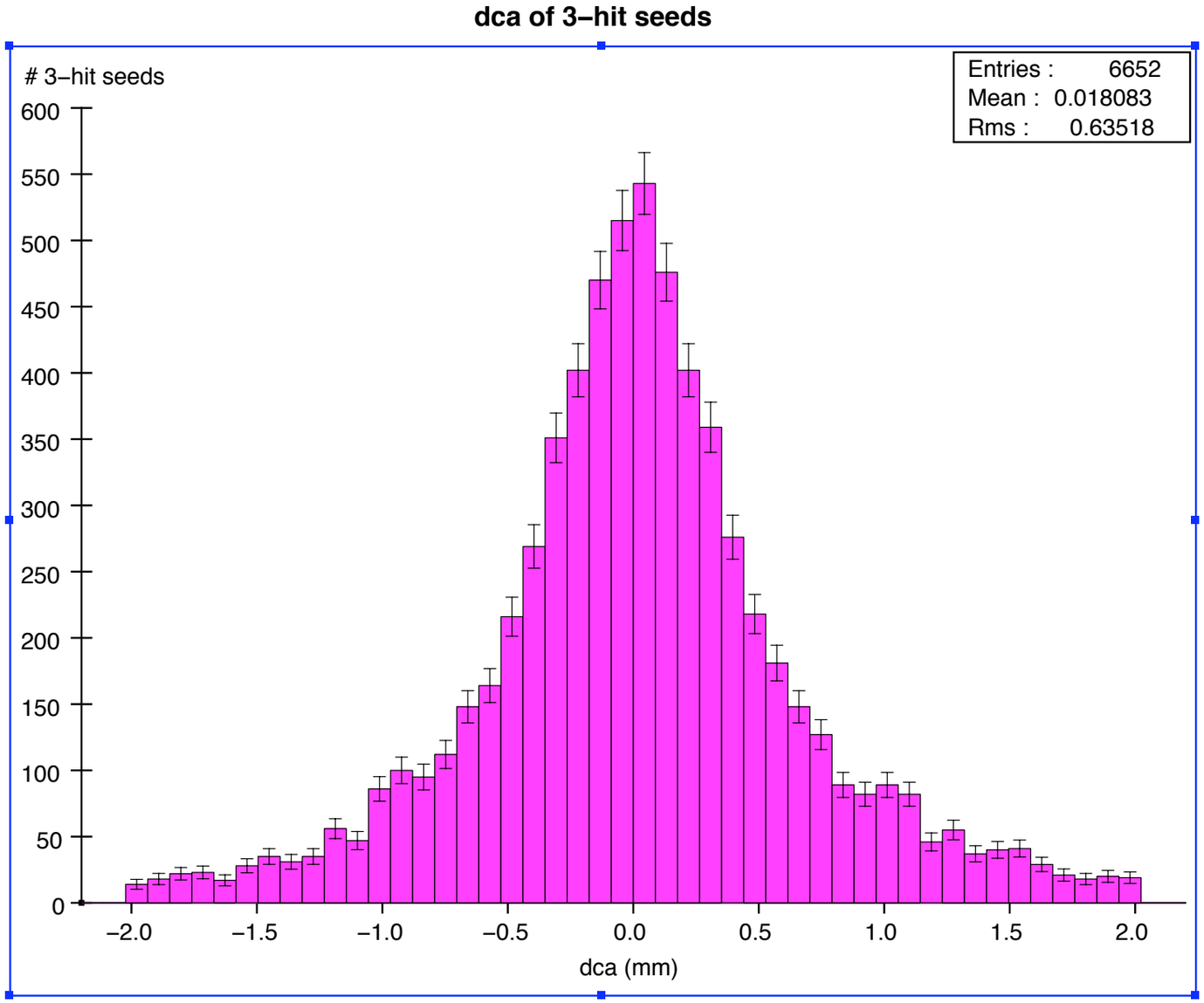
- Take all triads of three layers beginning from outside and walking inward
  - Take all combinations of three hits that have no nearby (<0.5mm) hits
  - Tried both 10 cm and +/- z modularity
  - Create circle and require that it passes close to IP (dca < 2.0mm)
  - Parse remaining layers from outside in and attach hits close to circle (nearest<0.5mm) that have no nearby hits (next_nearest>1.0mm) and refit
  - Parse remaining layers and attach **any** hits very close to the circle (<0.25mm)
  - Make very loose chisq cut and call these tracks “found”
- Create Tracks and stick them into event record

Standalone Tracking “v0.1”

- Two pieces of code:
  - TrackFinder extends Driver (finds tracks)
  - StandaloneTrack implements Track (represents tracks in event for display and further processing)
- Code is reasonably well commented, but not very pretty.
- Only minor attention paid to optimization: 1sec / event on ttbar events. An order of magnitude faster should be easily achievable.

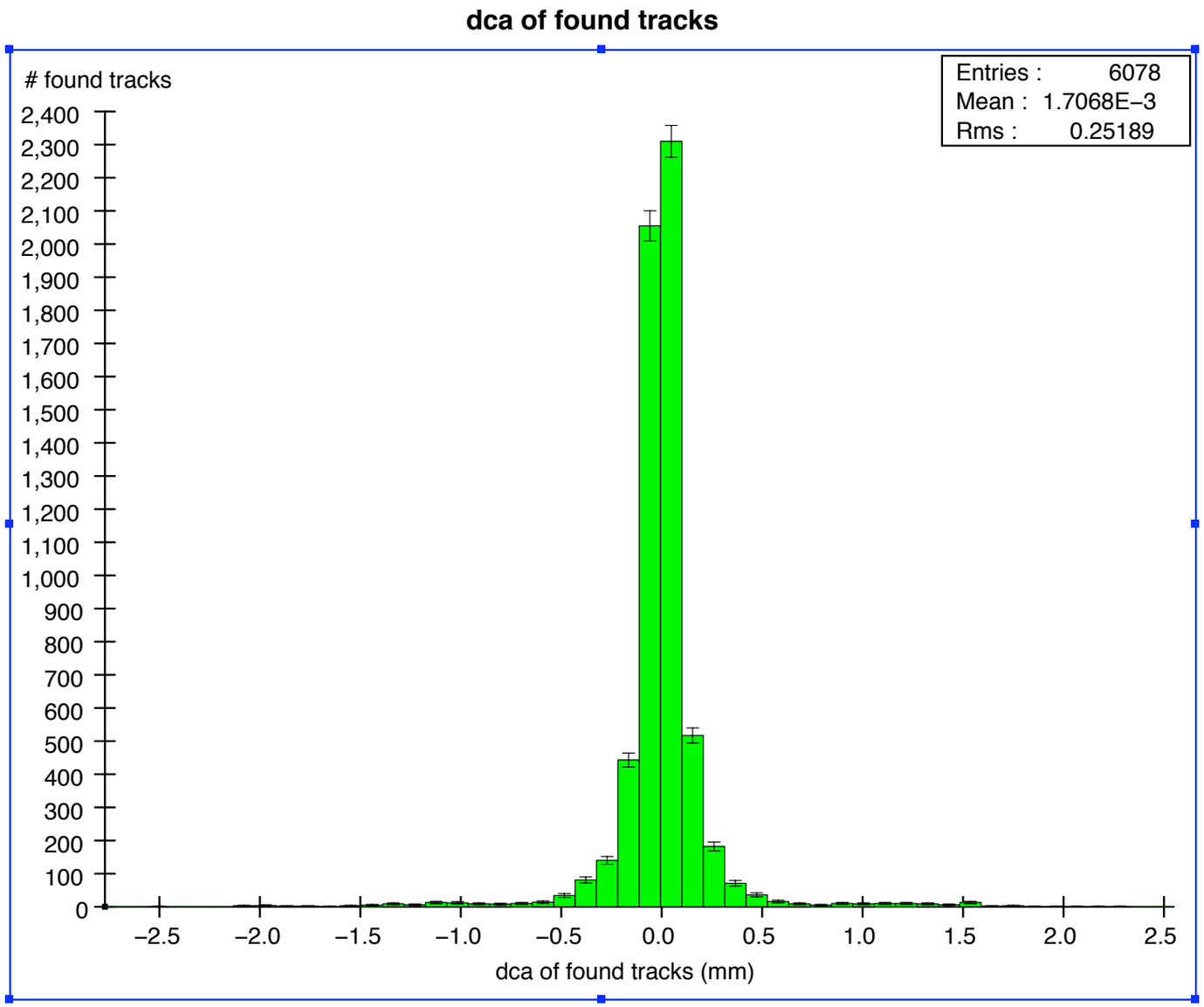
Single Muons (easy)

d0 of all 3-hit seeds



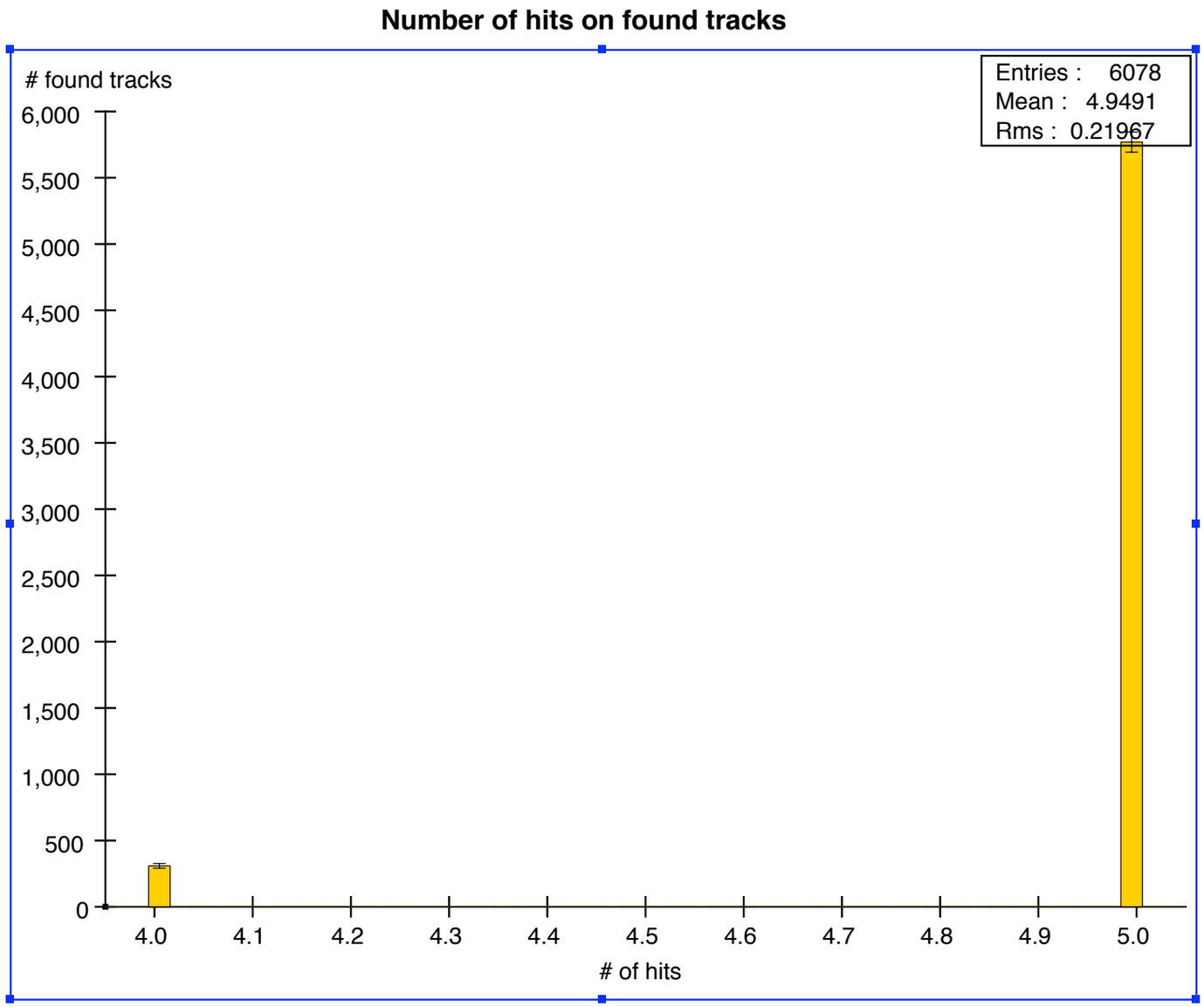
Single Muons (easy)

d0 of final tracks



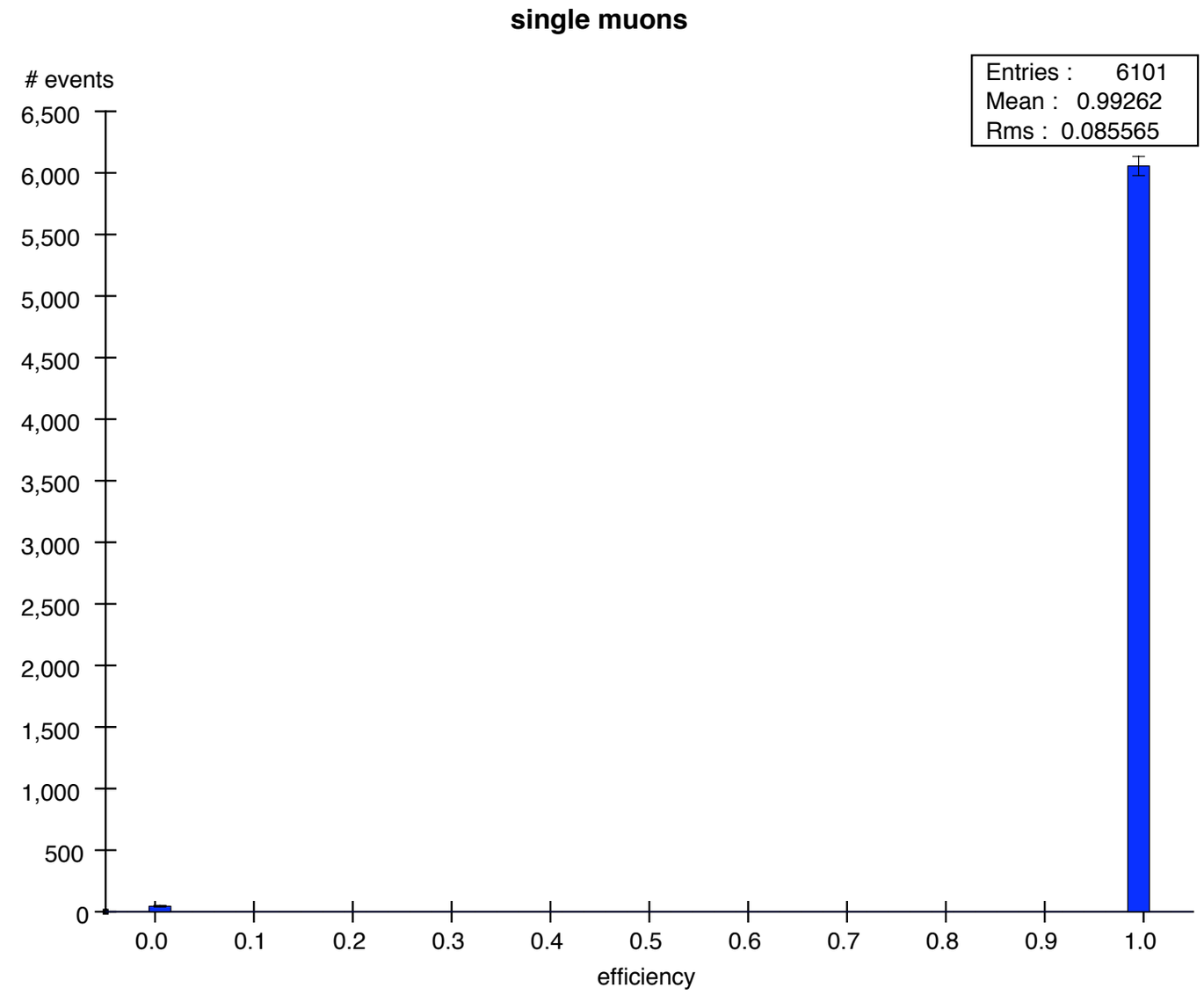
Single Muons (easy)

All five hits found 95% of time



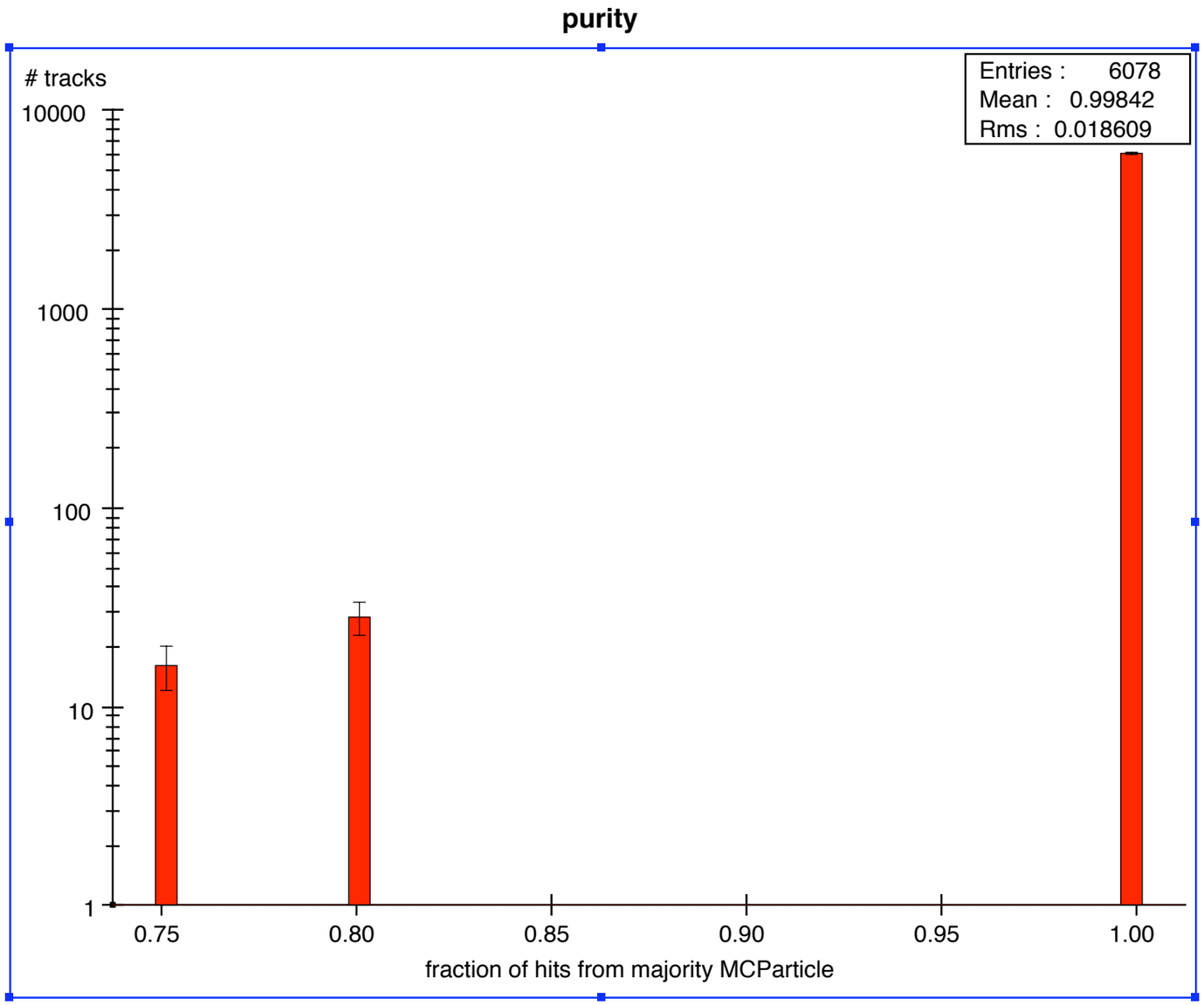
Single Muons (easy)

99.2% efficient for 5-hit muons



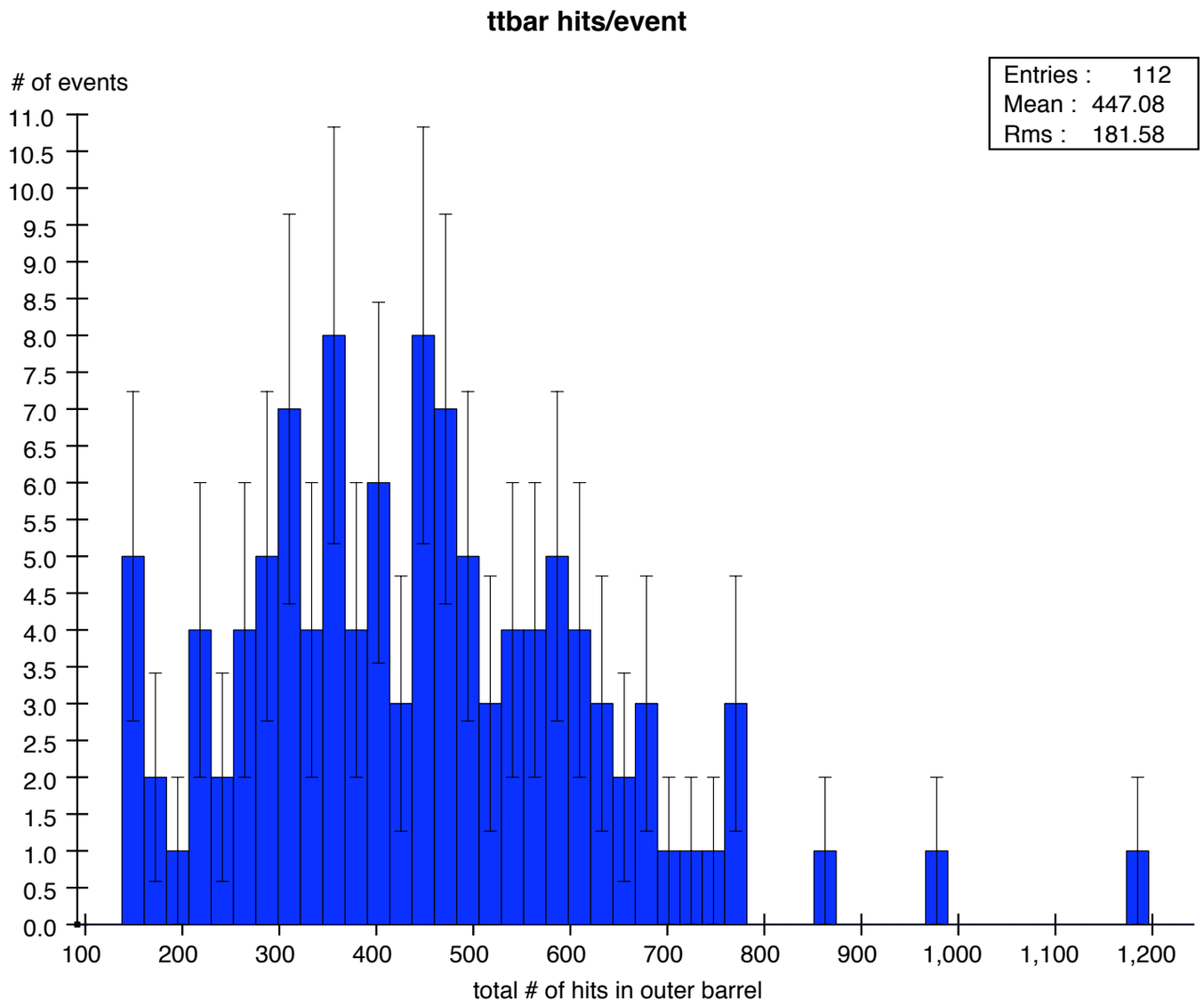
Single Muons (easy)

~1% have a hit from another MCParticle



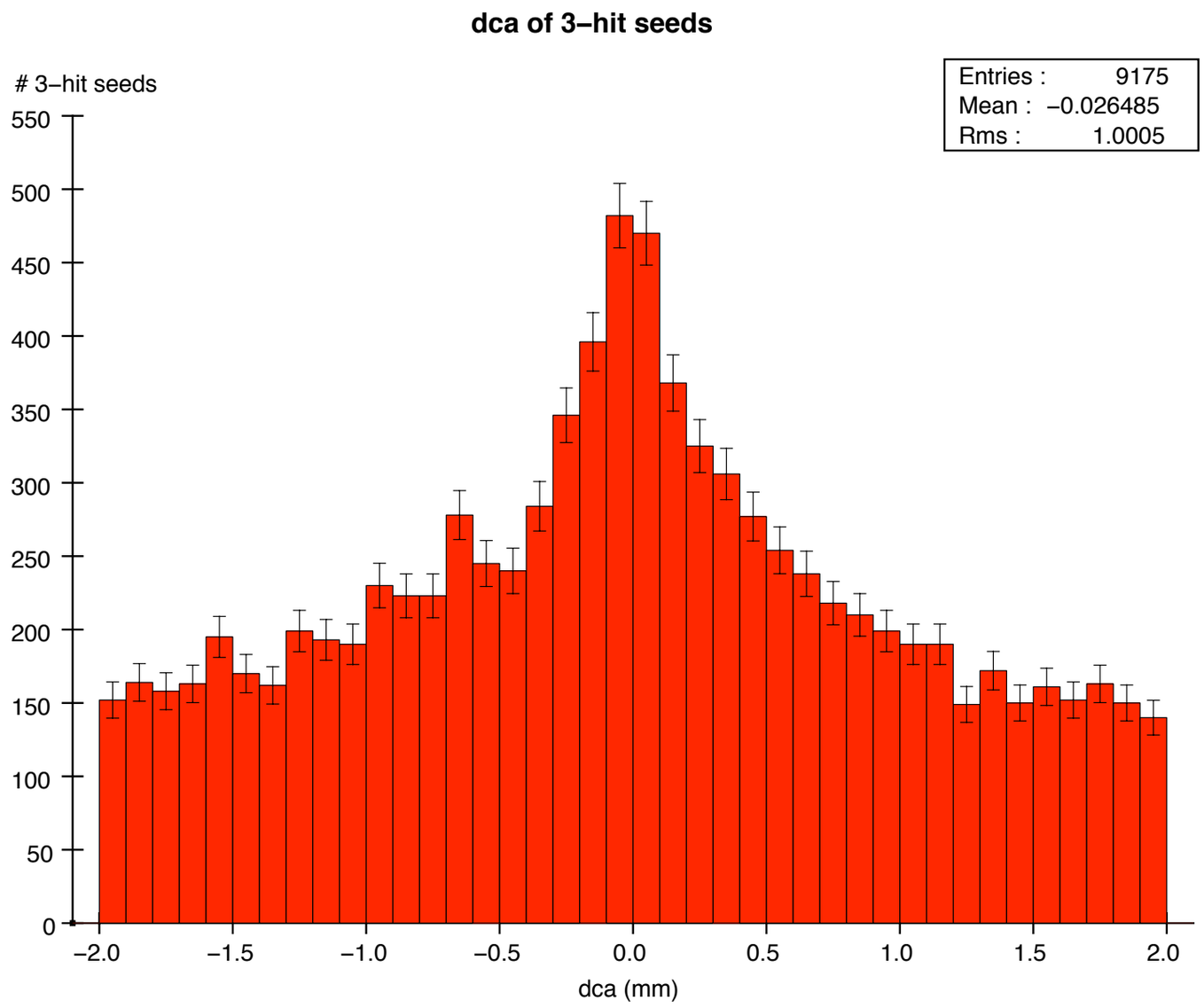
ttbar (hard)

typically hundreds of barrel hits per event



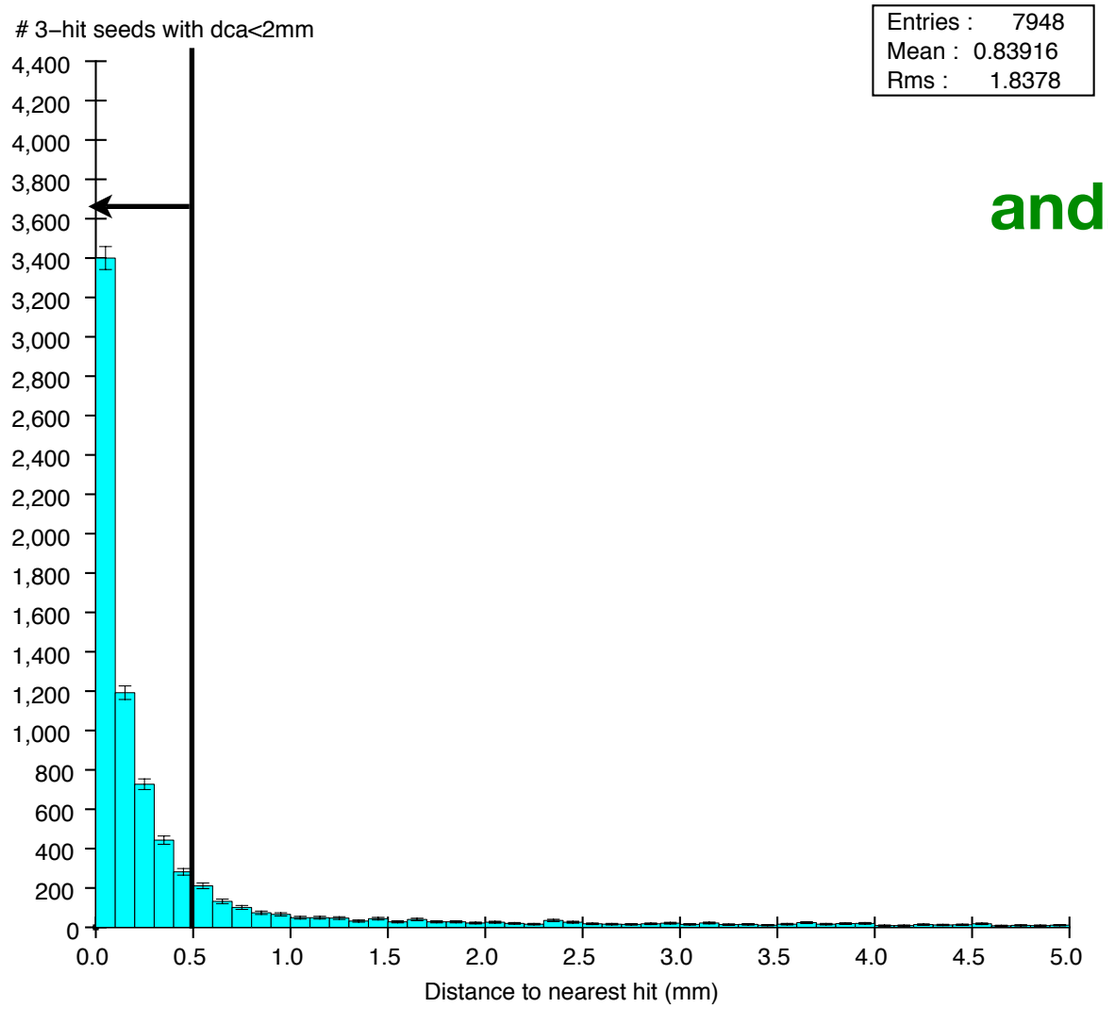
ttbar

not difficult to see IP in 3-hit seeds with 10cm z segments



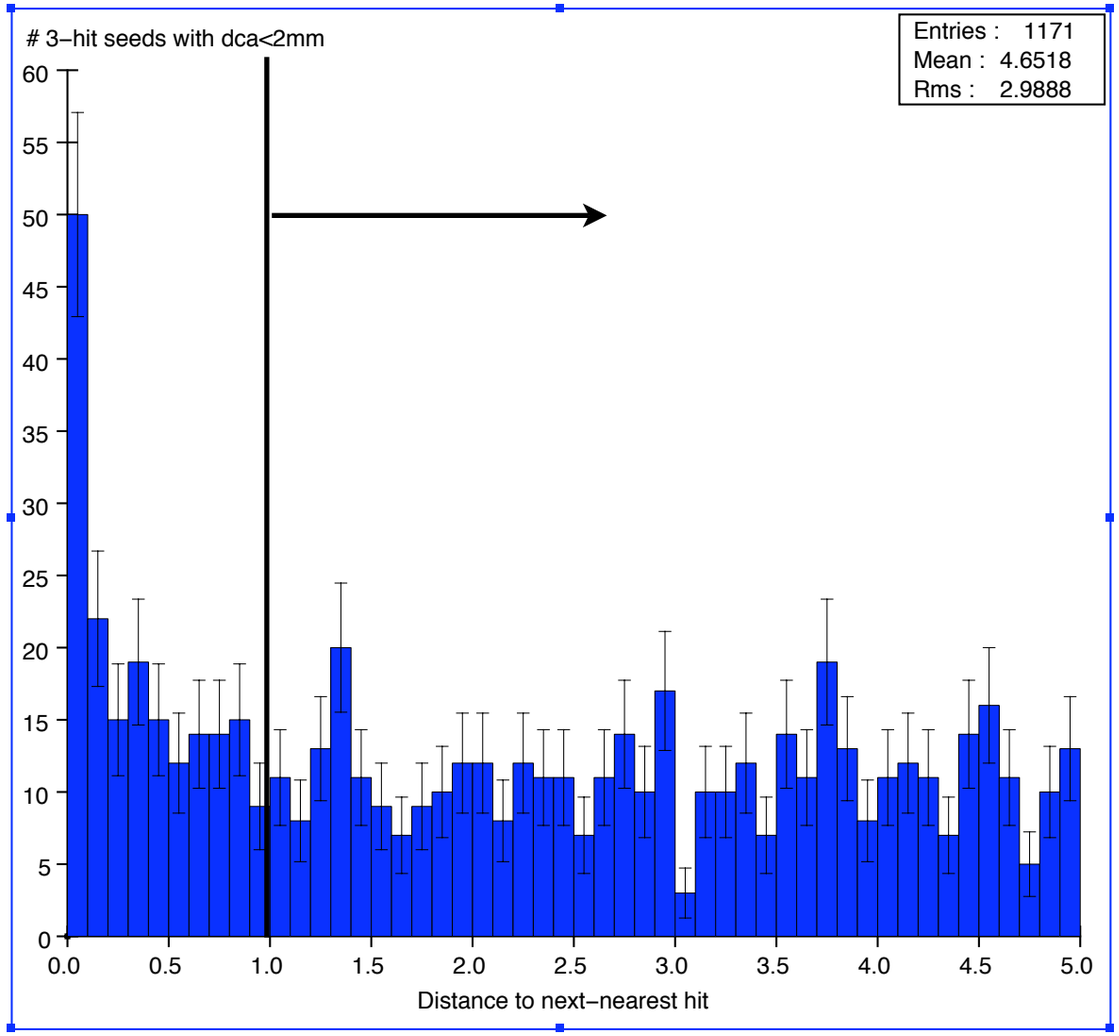
ttbar

Nearest additional hit on pass 1



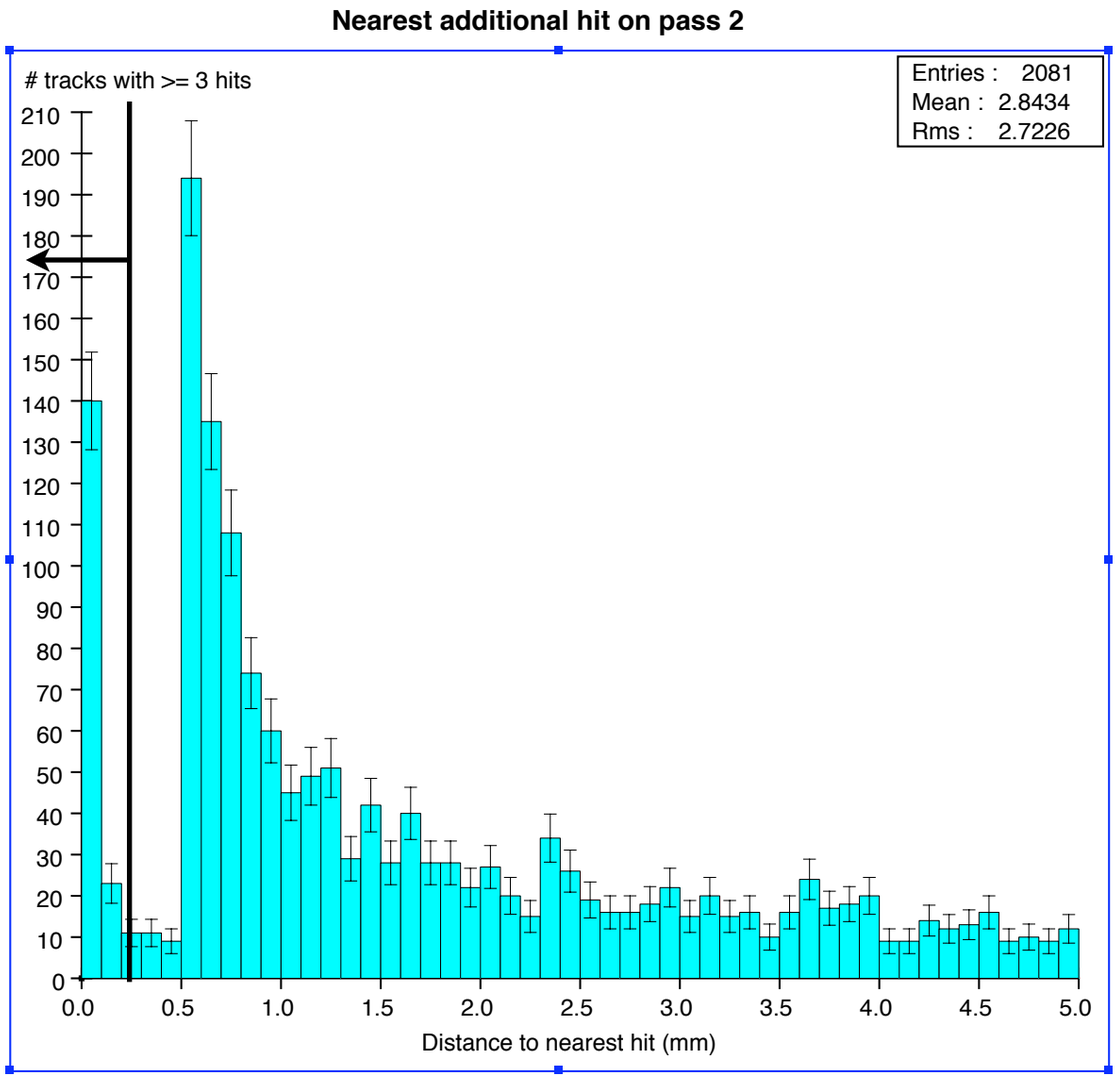
and...

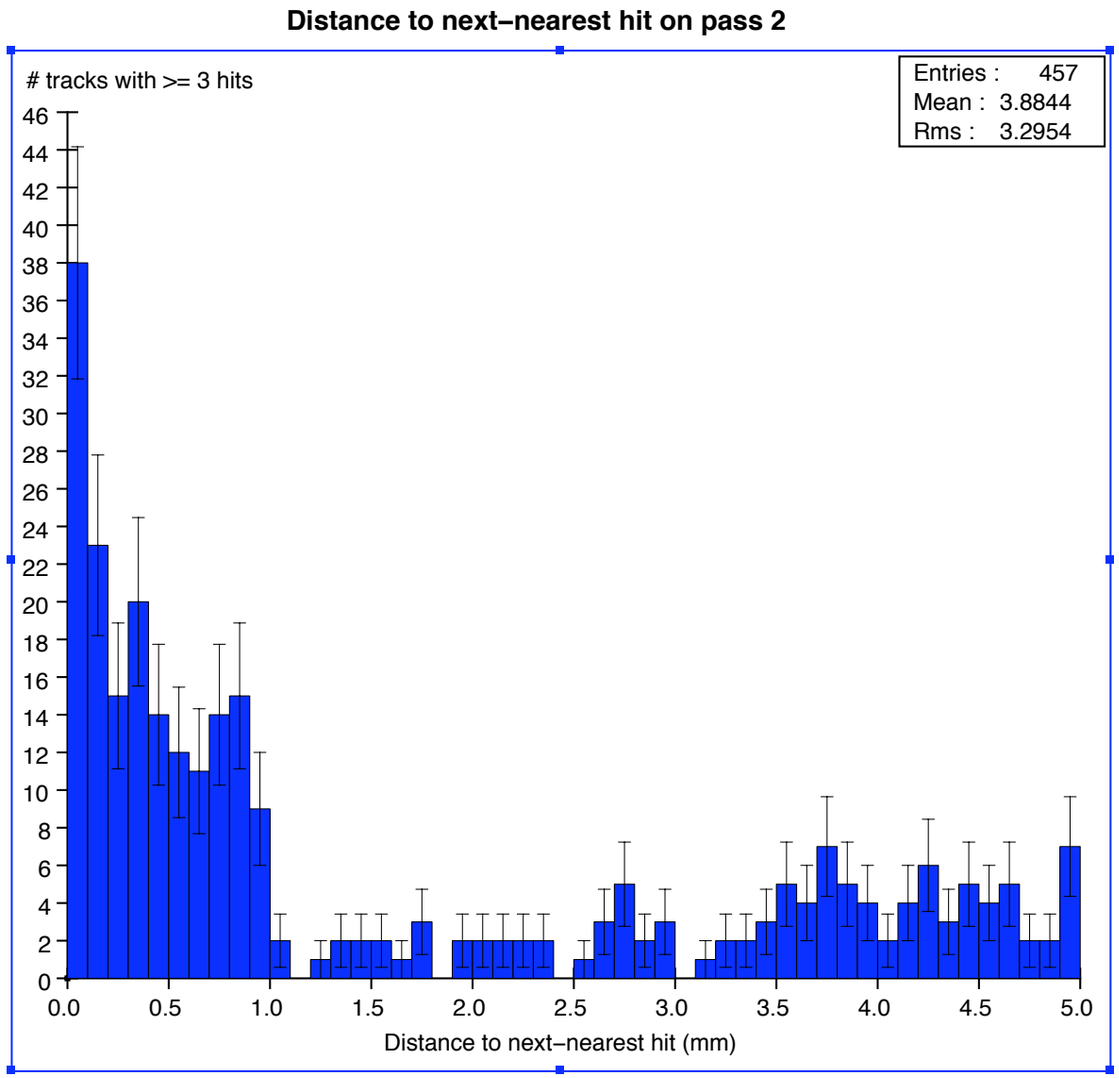
Next–nearest additional hit on pass1



pass1 - adding isolated hits and refitting

ttbar

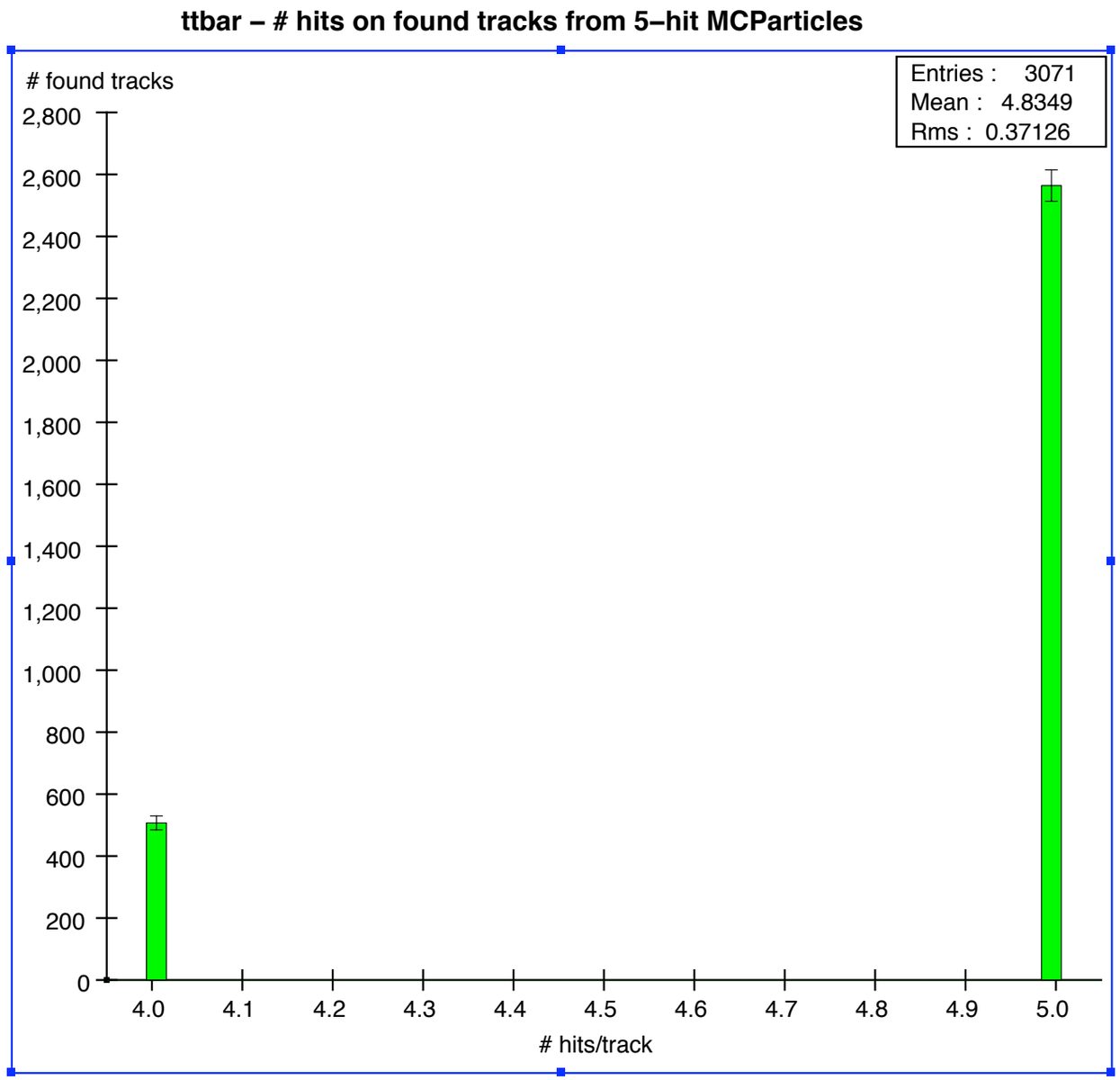




pass2 - add only hits very near trajectory

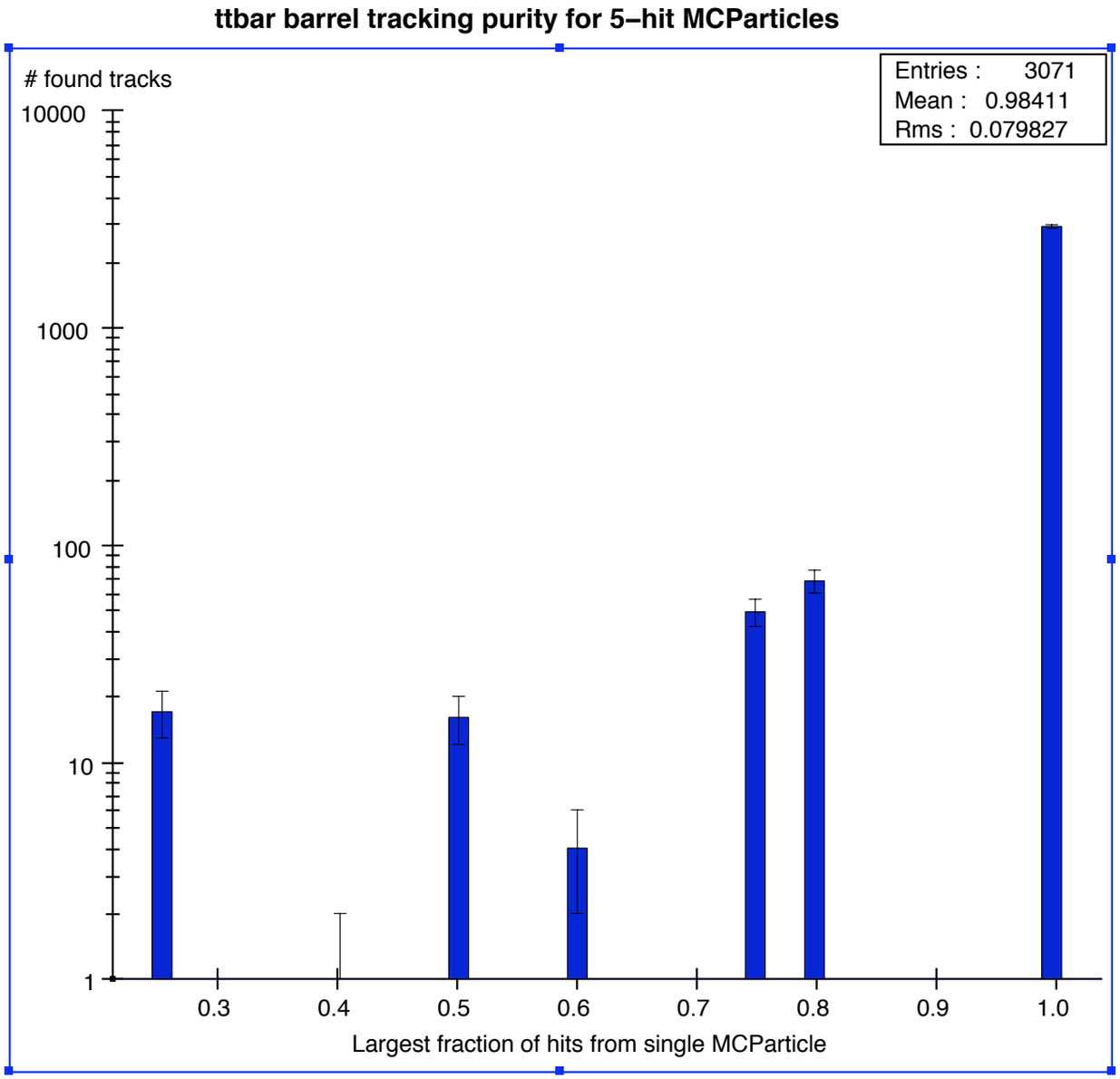
ttbar

all hits found
83% of time



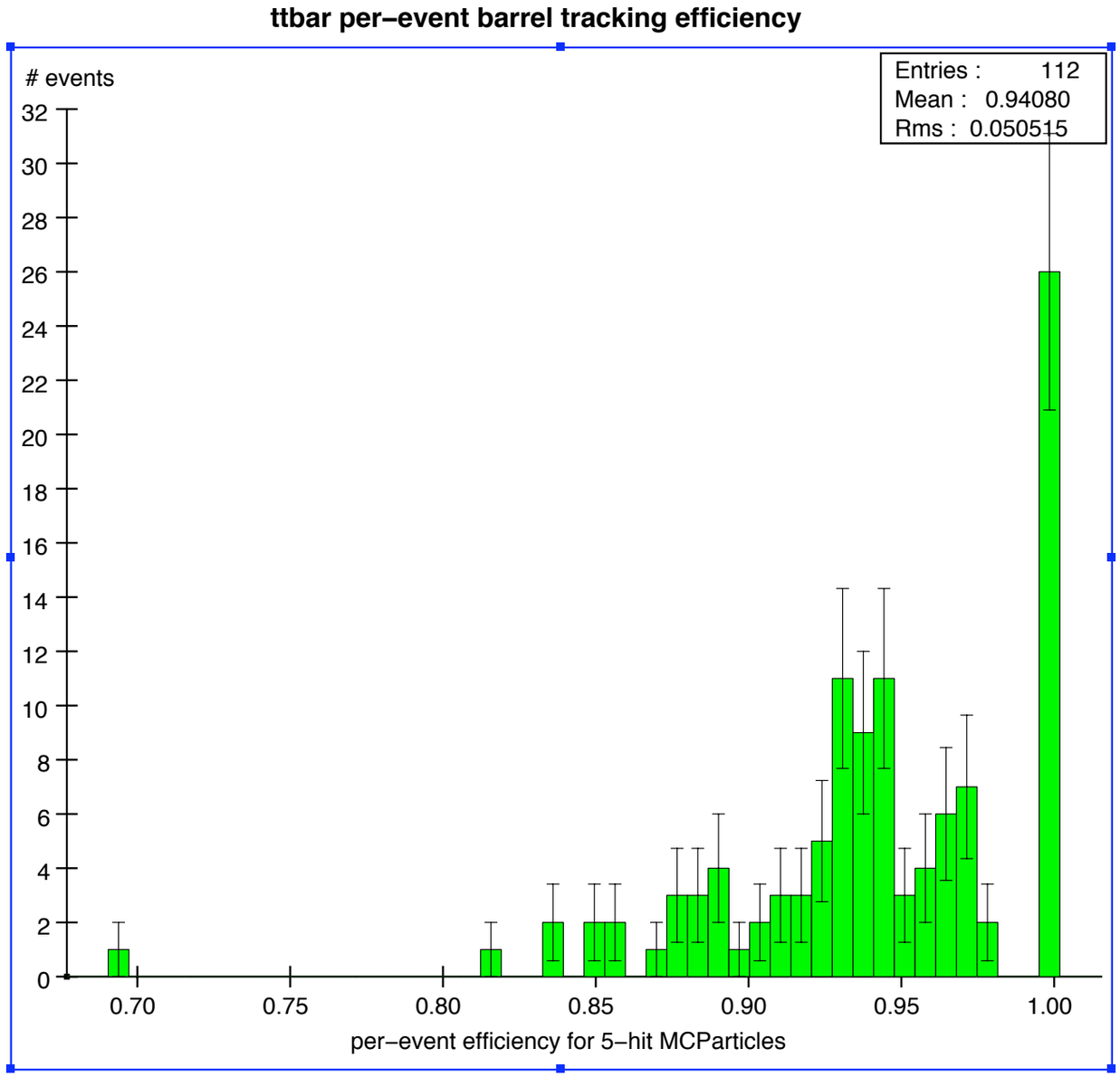
ttbar

- 1-2% get one wrong hit
- A few tenths of a % are garbage
  - no serious garbage with 5 hits



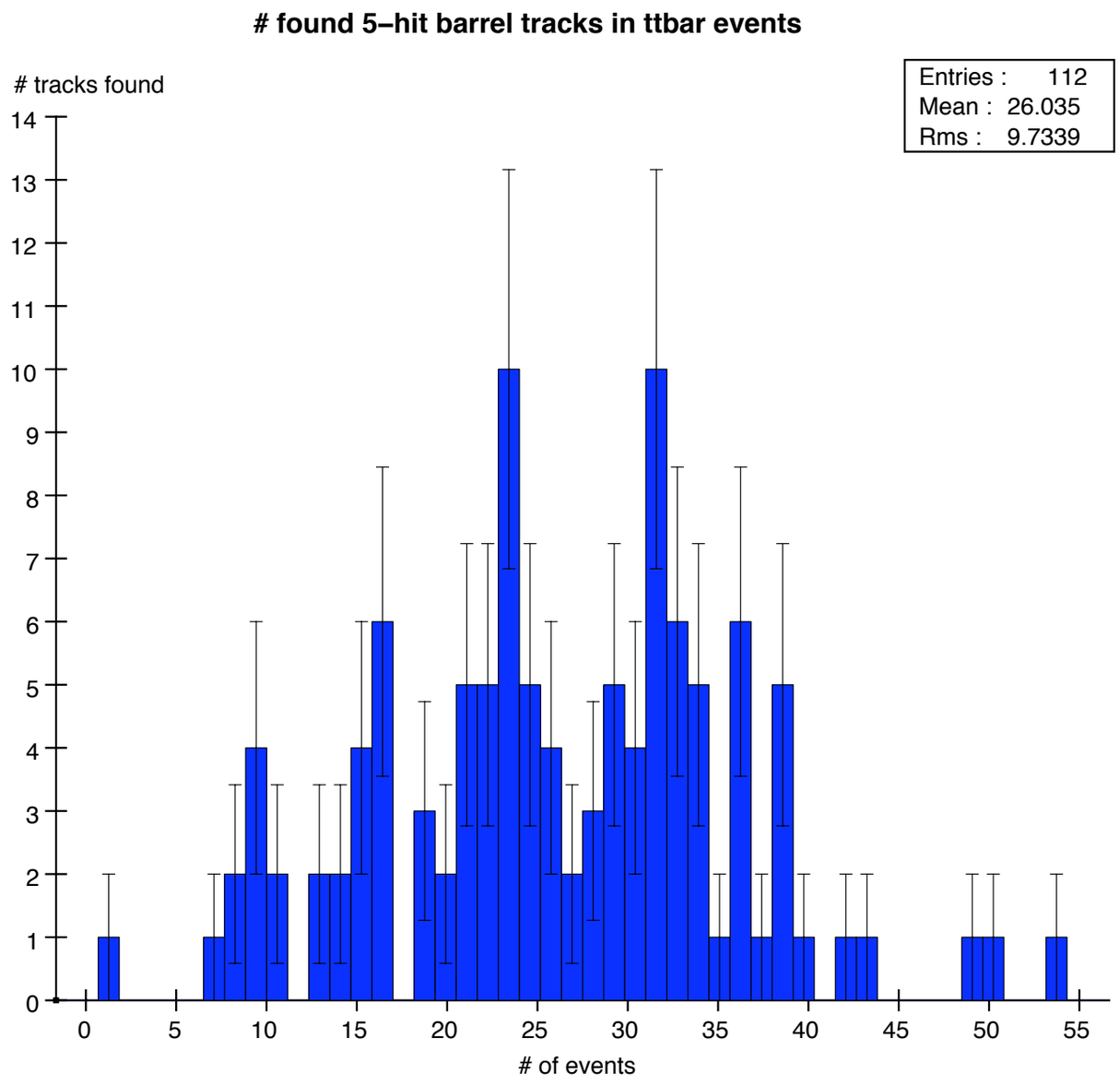
ttbar

- **Most likely outcome: all tracks found**
- **Average per-event efficiency is 94%**
- **Struggling with jas3 to get plots as function of momentum, etc.**

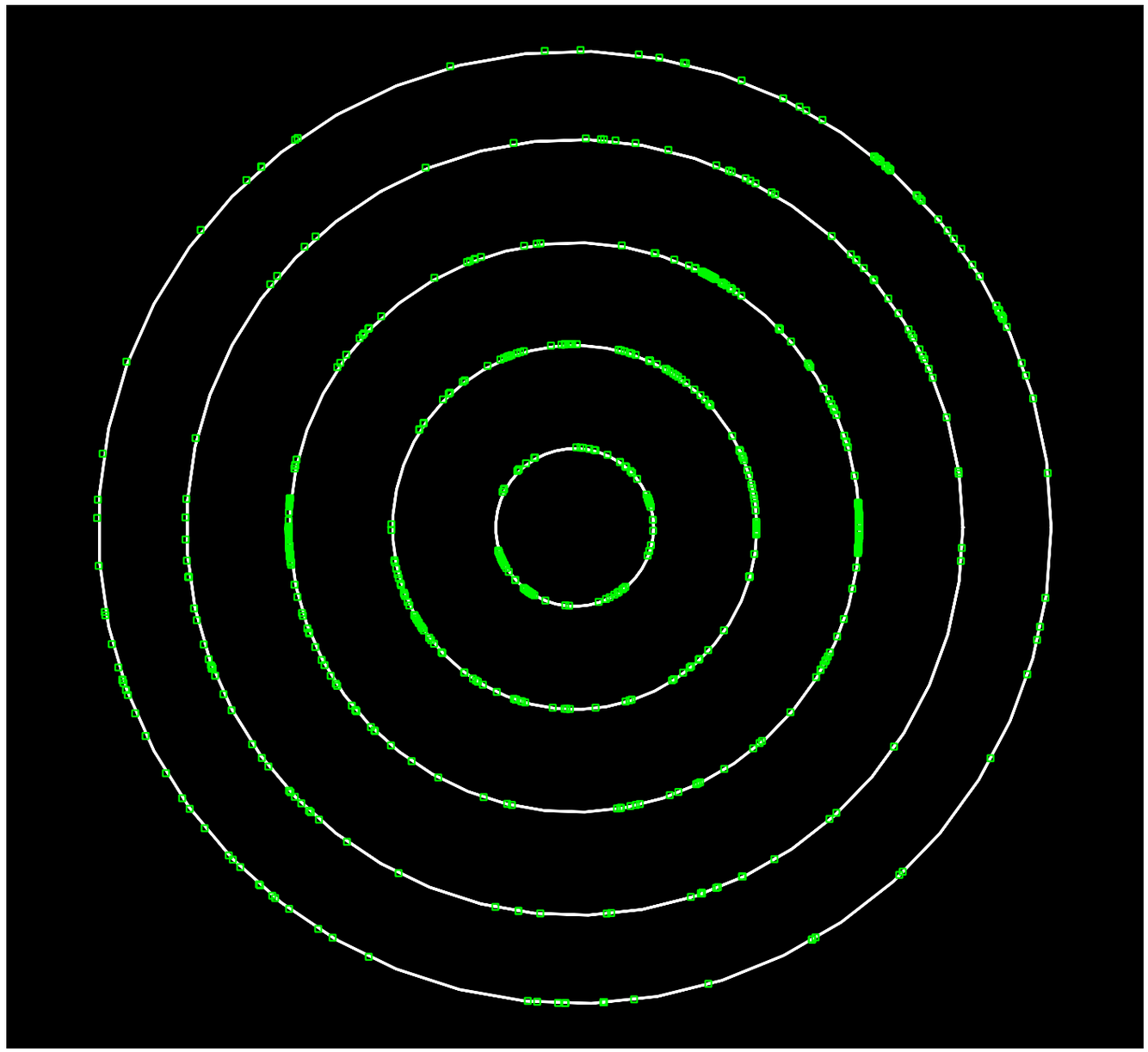


ttbar

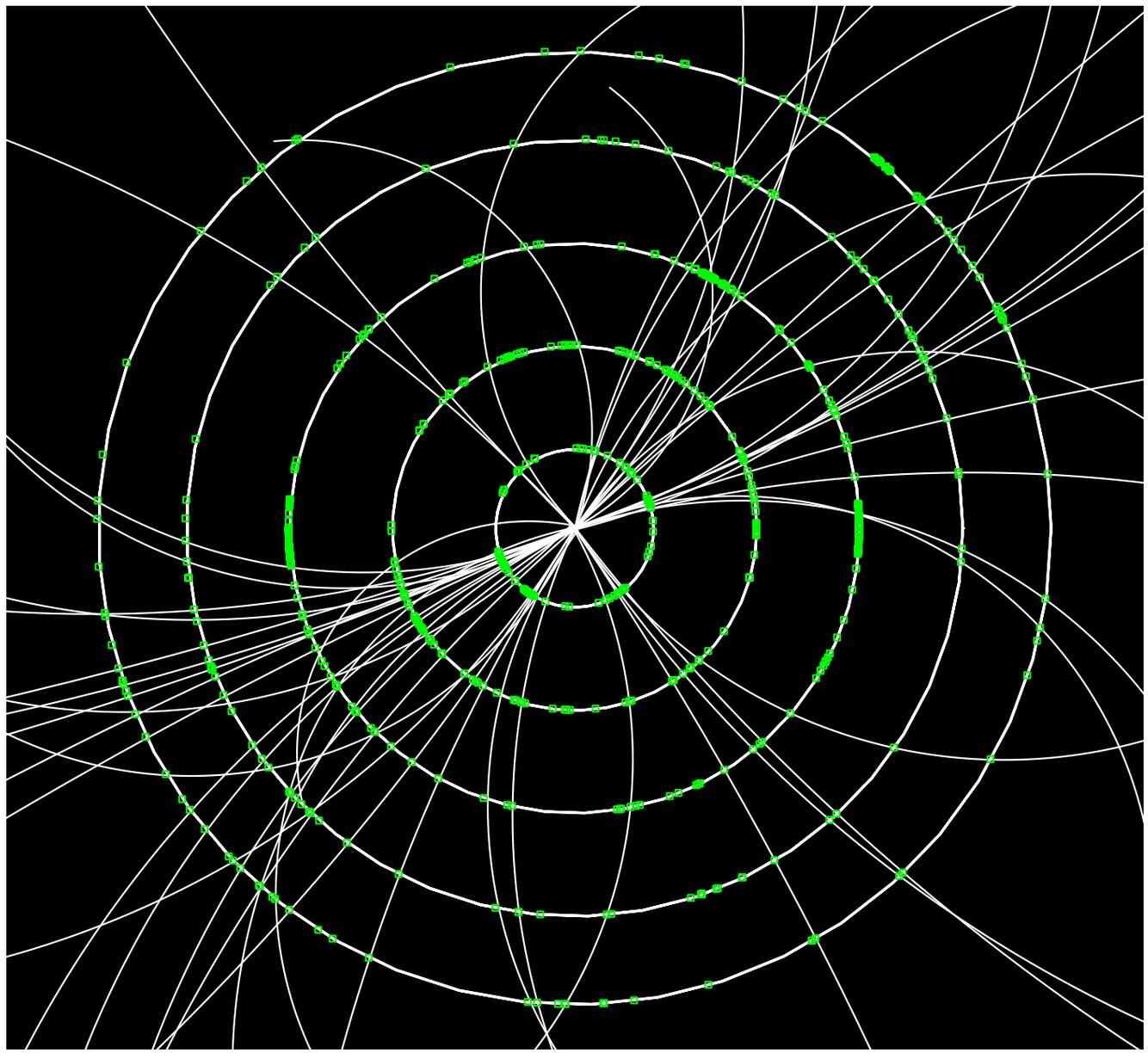
Tracks found/event



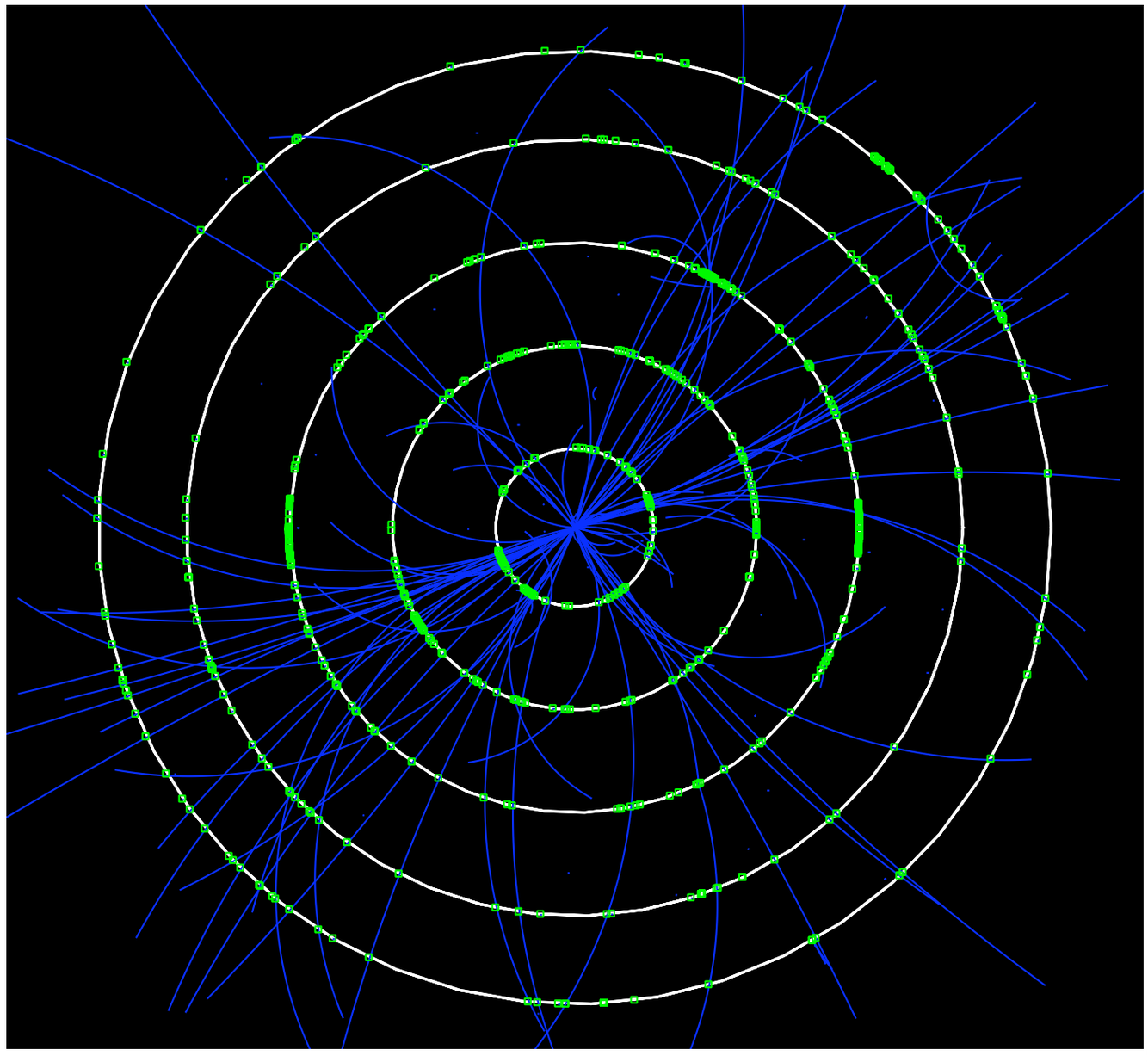
ttbar - a nasty example



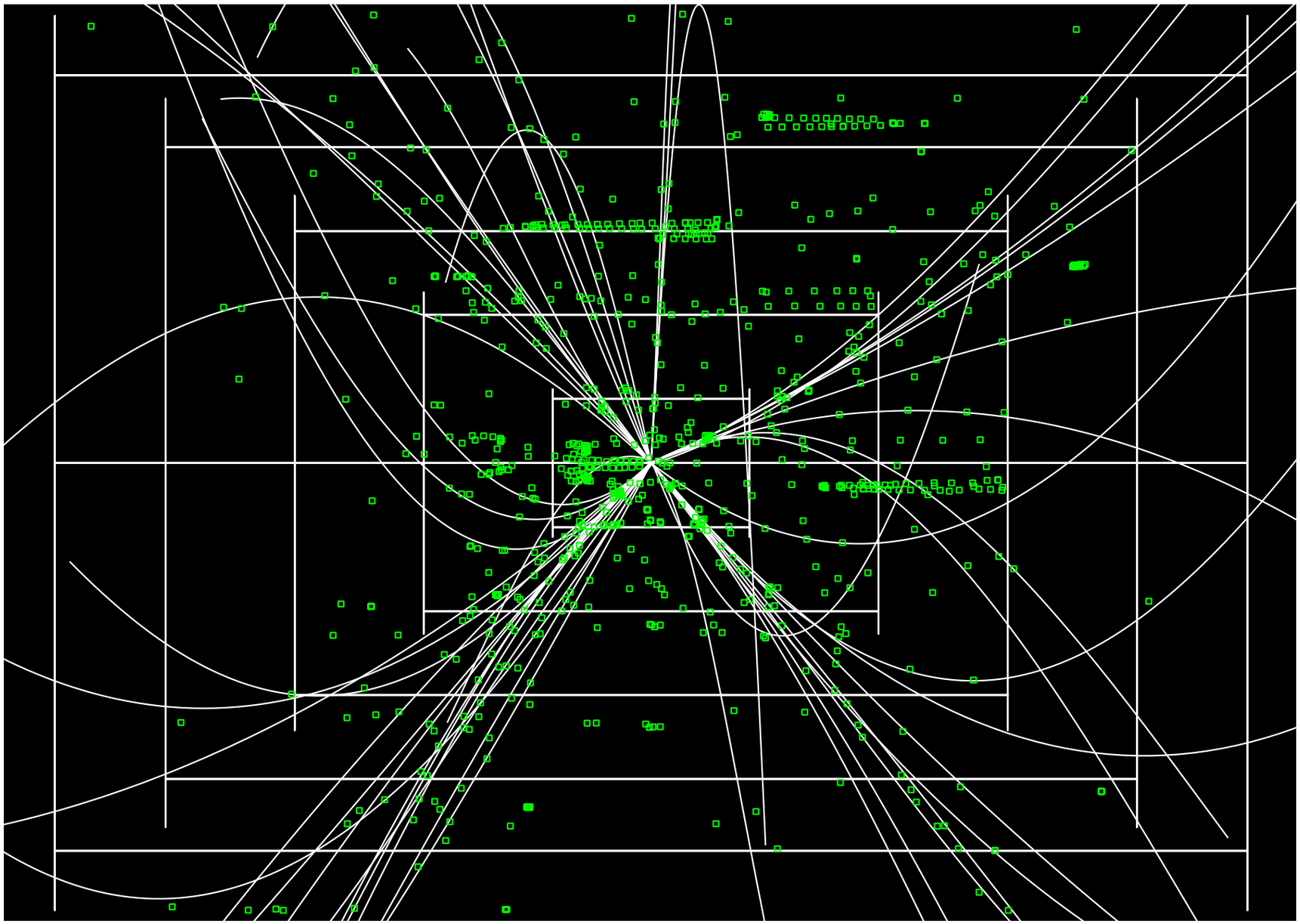
ttbar - a nasty example



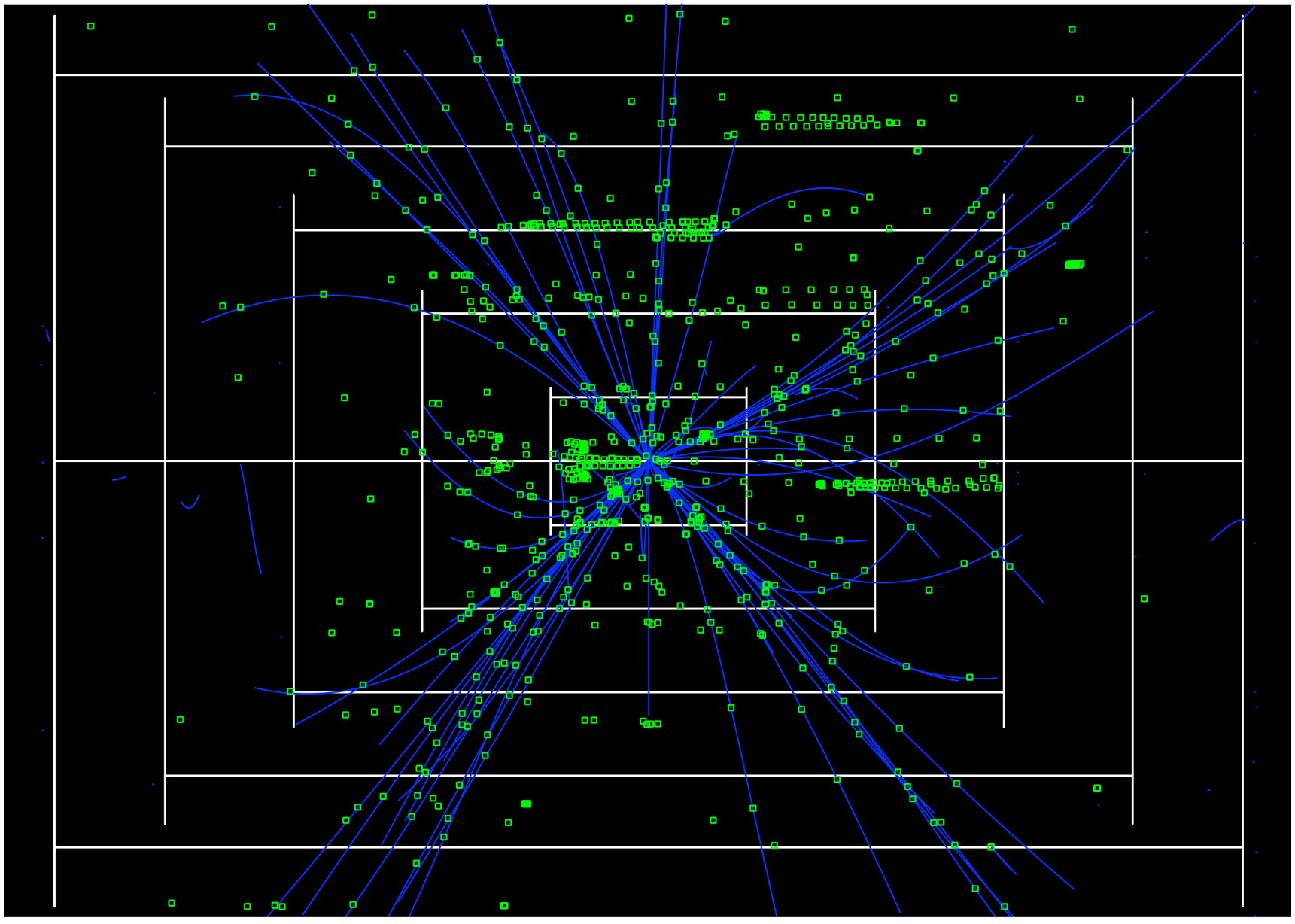
ttbar - a nasty example



ttbar - a nasty example

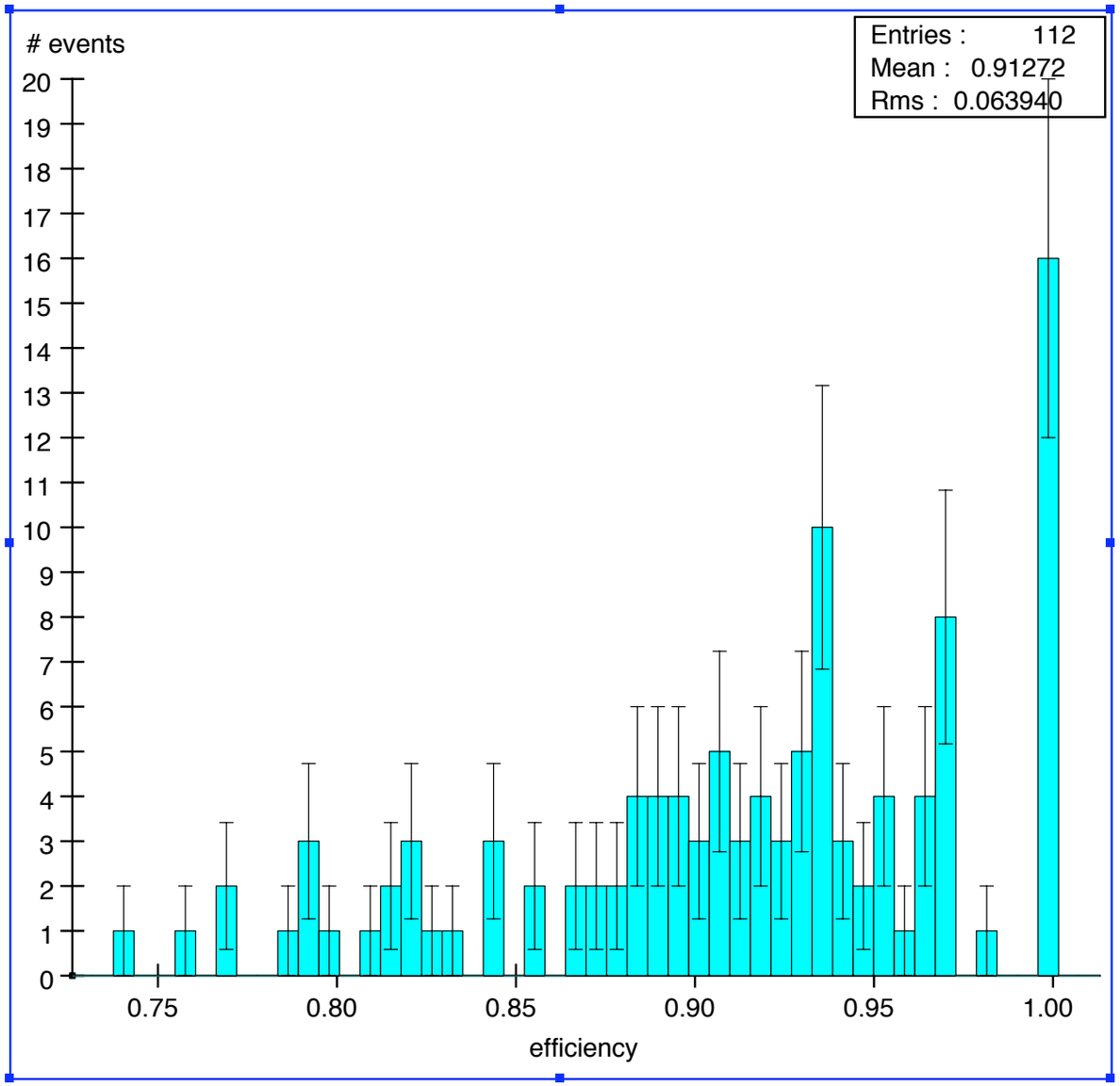


ttbar

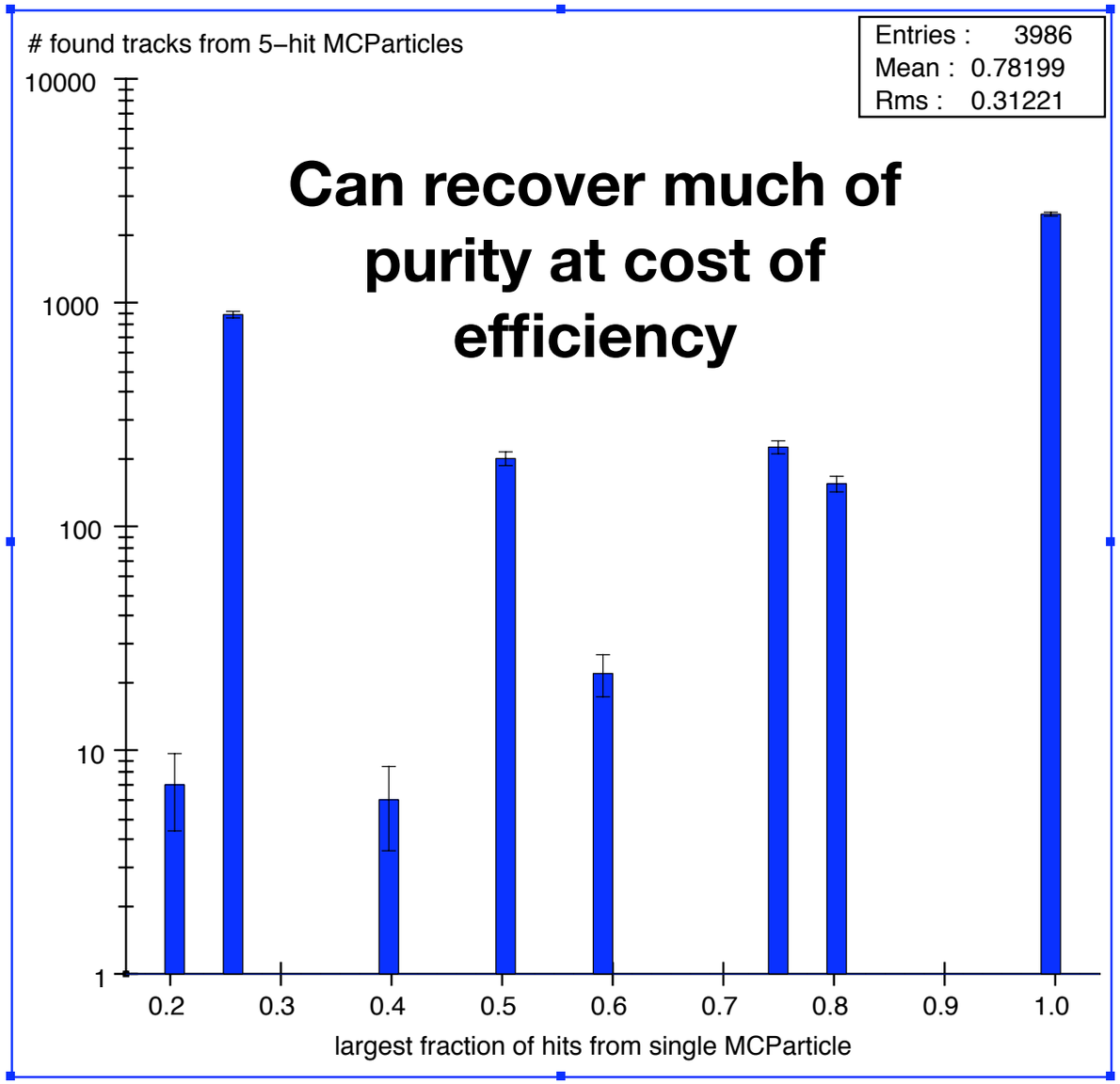


ttbar - only +/- z segments

efficiency – only two z segments



purity – only two z segments



Conclusions

- With only minimal work, standalone in outer barrel tracker works surprisingly well for our default design
- A better fitter will help, realistic silicon simulation shouldn't hurt too much until we are going after the last 5%
  - code can be run sequentially with looser cuts after removing hits to go after remaining tracks, 4-hit tracks, vees, etc...
- Code being documented, cleaned up, checked in
- On to finishing silicon simulation