Kalman Filter Pattern Recognition and Fitting Status Update

Robert Johnson

U.C. Santa Cruz

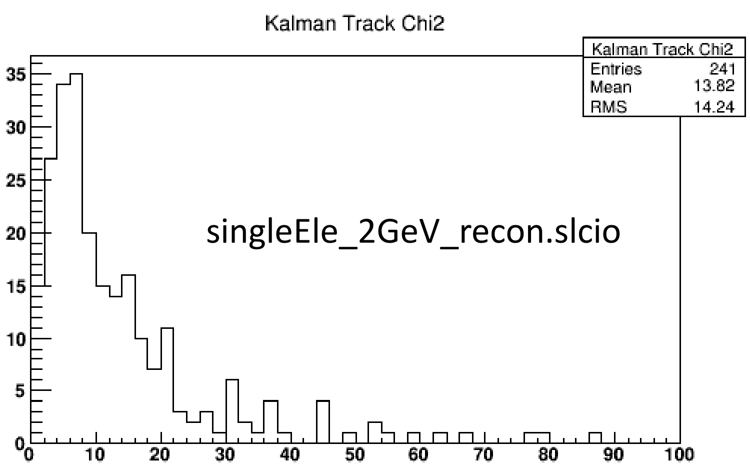
January 28, 2020

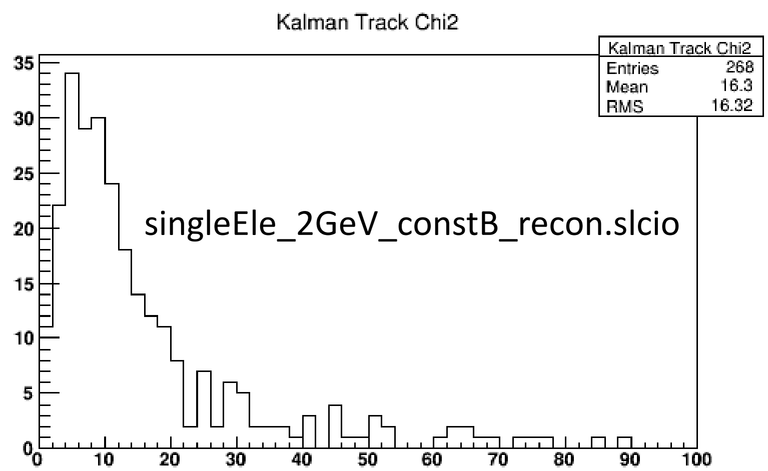
Testing the effect of field non-uniformity

- Using the toy Monte Carlo, I generated tracks using the full field map and then fit them assuming a uniform field, compared with fitting using the full field map.
  - No significant differences in the pull distributions of helix parameters.
- Using a sample of 2015 A' MC, I fit the tracks using the full field map and using a constant field.
  - No significant differences in the fit chi^2 distribution, the tracking efficiency, the momentum resolution, vertex resolution, or hit residual distributions.

Testing the effect of field non-uniformity

- Using two files of 2016 single-electron MC generated by Cameron:
  - singleEle_2GeV_recon.slcio
  - singleEle_2GeV_constB_recon.slcio
- Fitting both with the full field map yields similar results, *maybe* only slightly better with the non-constant B:



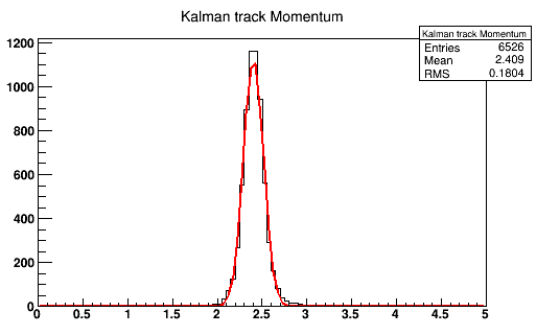


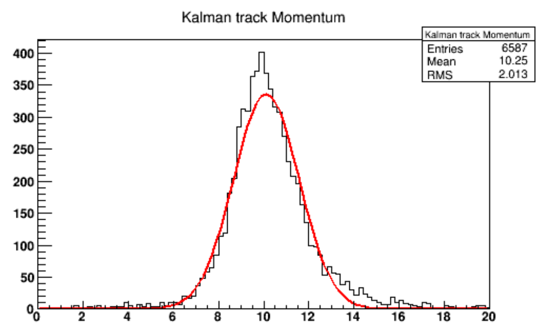
- But fitting the constant-B file with a constant field does not yield any noticeable improvement, so I think the small difference seen above is not statistically significant but is only a result of looking at two statistically independent samples.

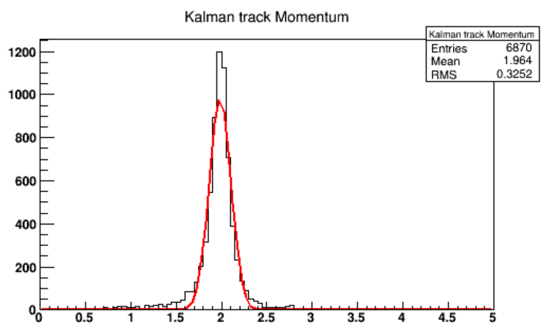
Testing with single-particle MC events

I used the following 3 MC 2016 simulated data sets generated by Cameron Bravo:

- users/bravo/sim/det16/singleMuon/slic/slicSingleMu4deg_recon.slcio
- singleMuMinus_10GeV_recon.slcio
- singleEle_2GeV_recon.slcio





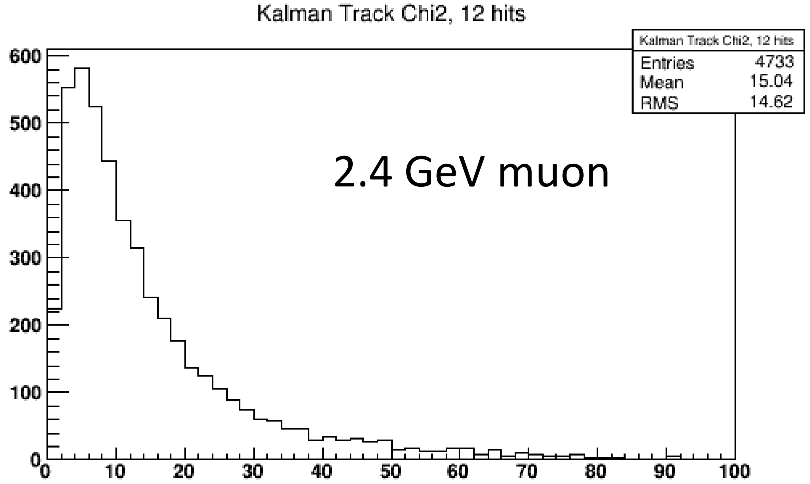


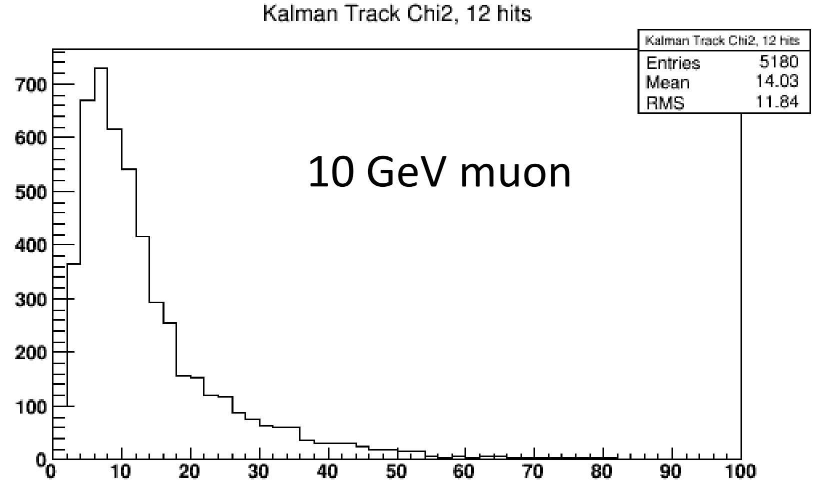
The Gaussian fits to the momentum distributions give sigmas of

0.112 GeV, 1.45 GeV, 0.120 GeV

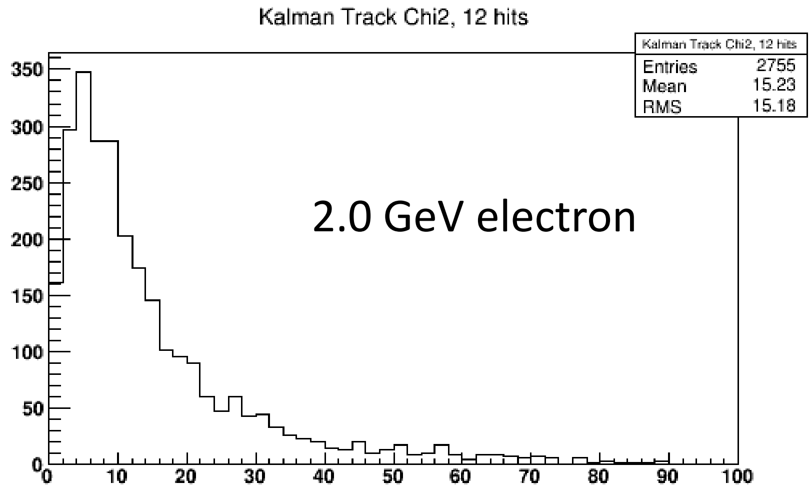
4.7%, 14.5%, 6.0%

Chi-squared Distributions

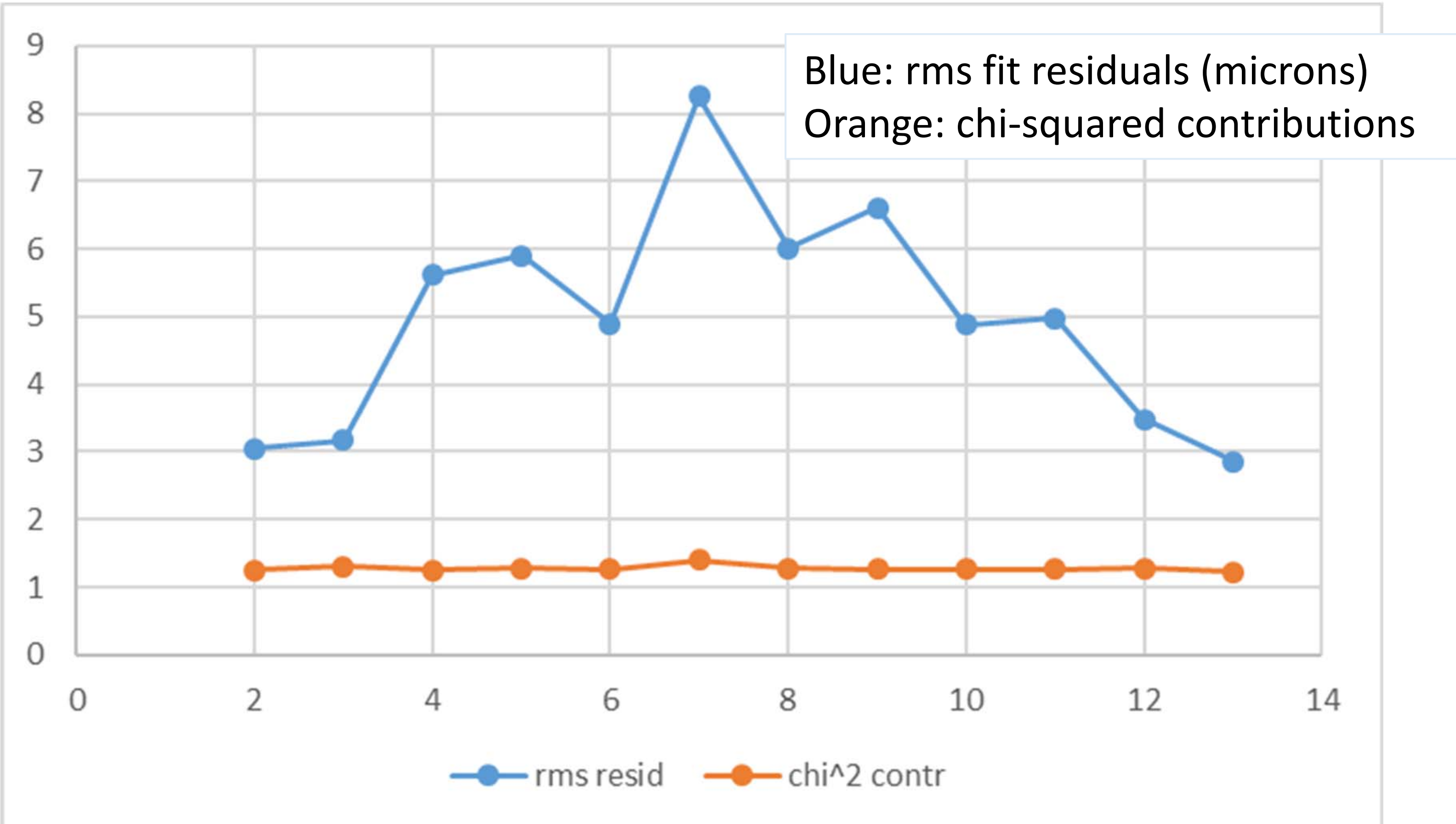




These all look about the same and also all have means significantly above the expected value of 12.

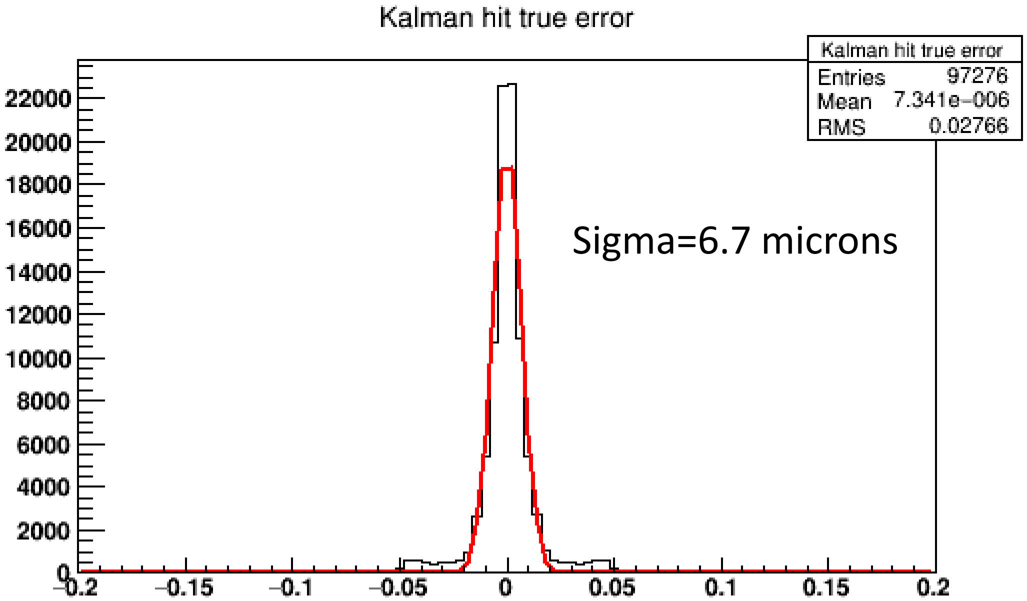


2 GeV Electrons



In all cases the chi-squared contributions are about the same from all 12 layers, as they should be, but are 20% to 30% higher than unity.

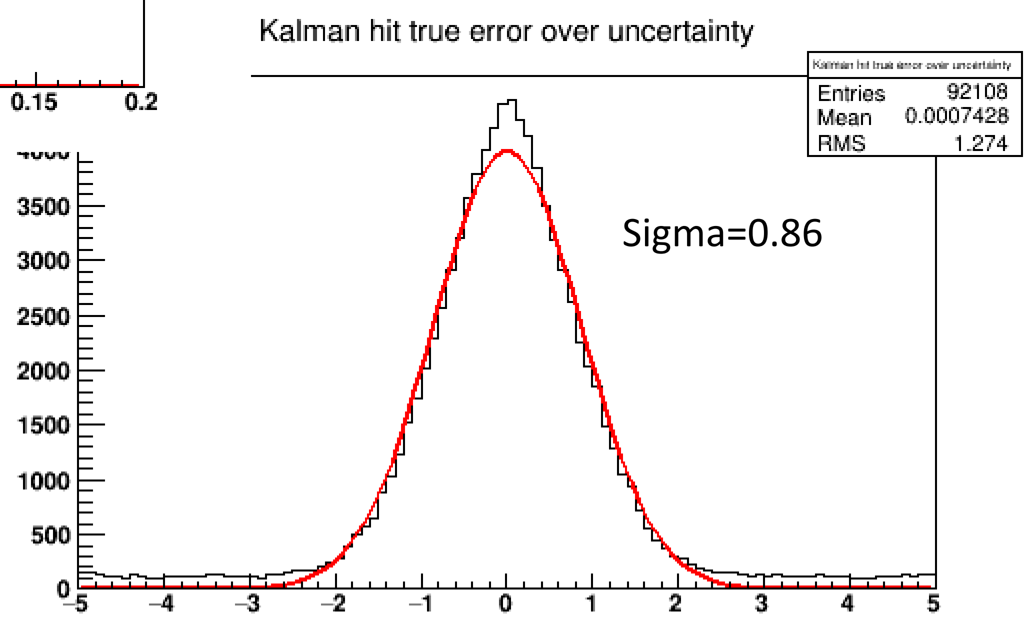
True Hit Errors



These are differences between the recon hits and the sim hits, of course along the measurement direction only.

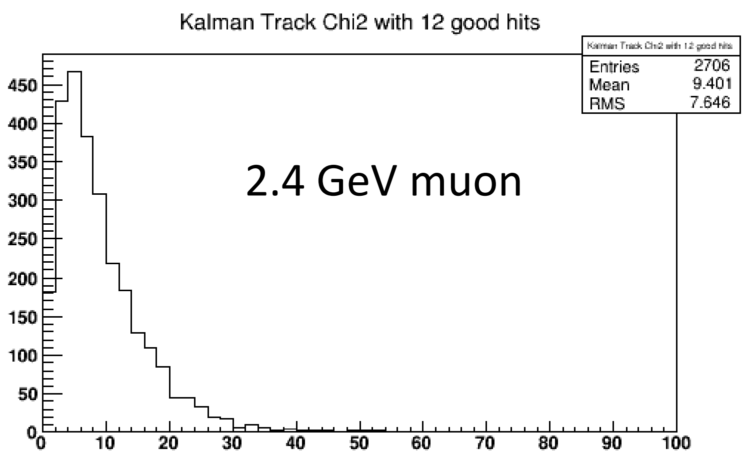
The hit errors have significant tails, which I think likely explain some of the deviation from the ideal value of the fit mean chi-squared.

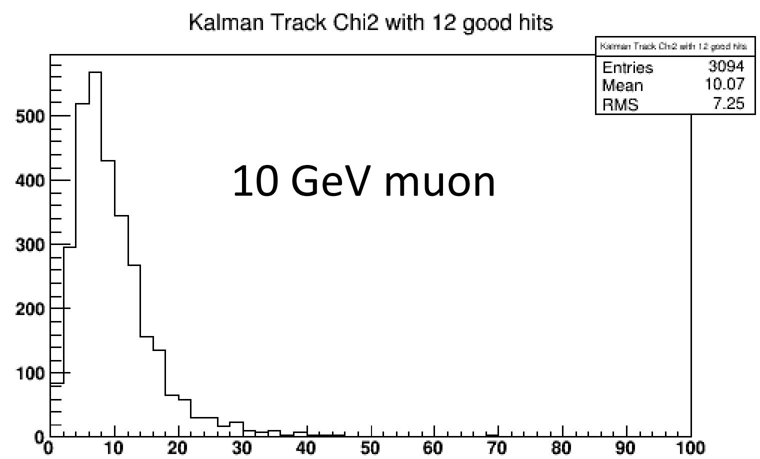
The rms error is significantly (~25%) larger than the uncertainty applied in the Kalman code when calculating chi-squared.



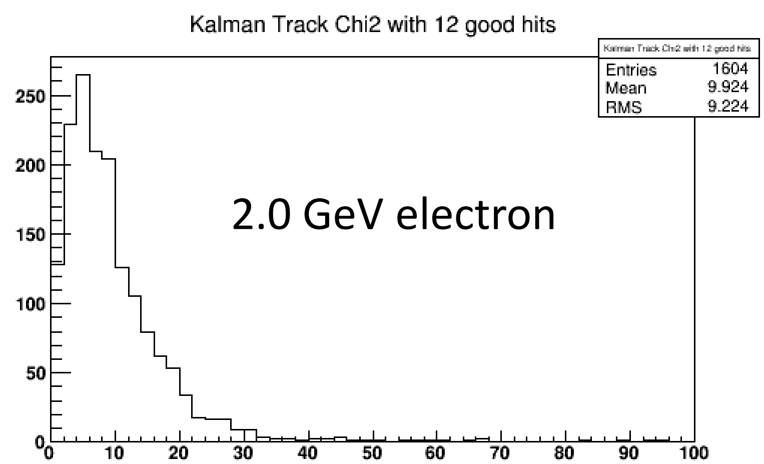
Tracks with only "good" hits

When I reject all events with any true hit error larger than 20 microns, then the mean fit chi-squared is much lower:



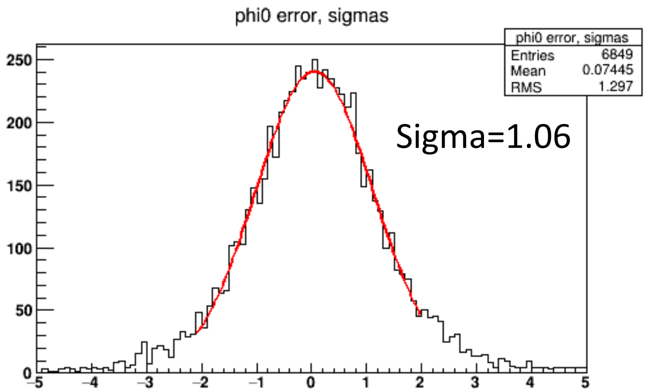


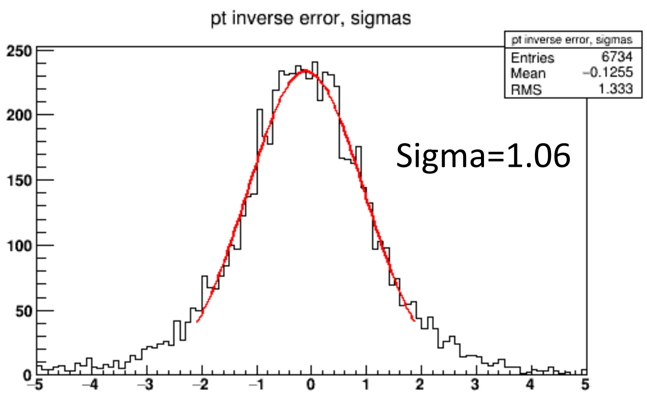
Conclusion: I think the fit is working correctly at the statistical level, as it does with the "toy" Monte Carlo.

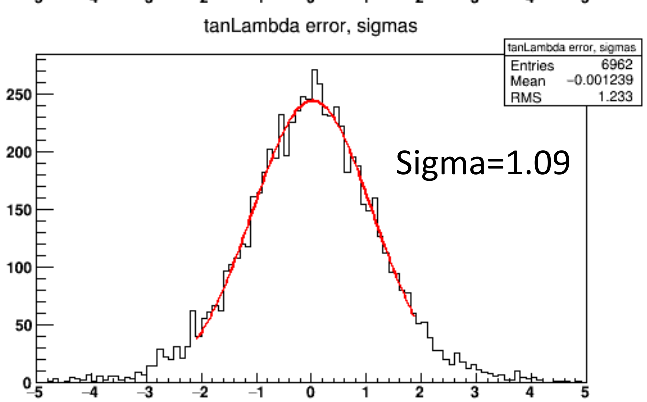


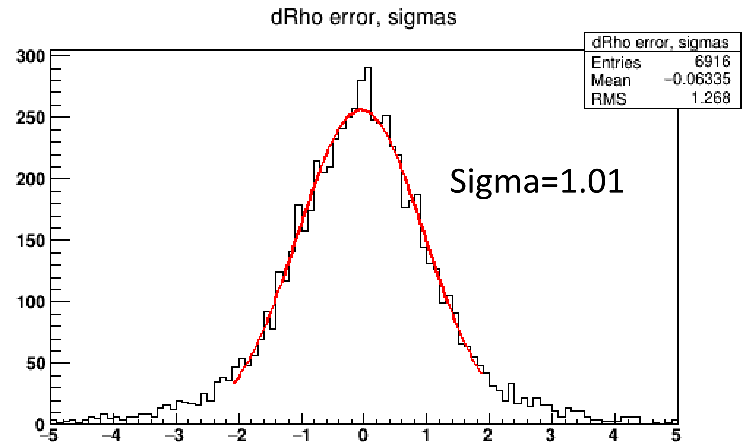
Note that these events were all pattern-recognized by the Kalman code, but of course they are trivial, with only a single track each.

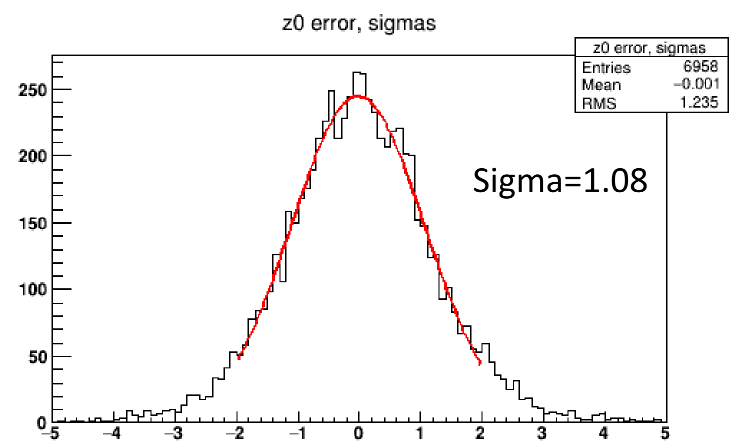
Helix Parameter Pull Distributions



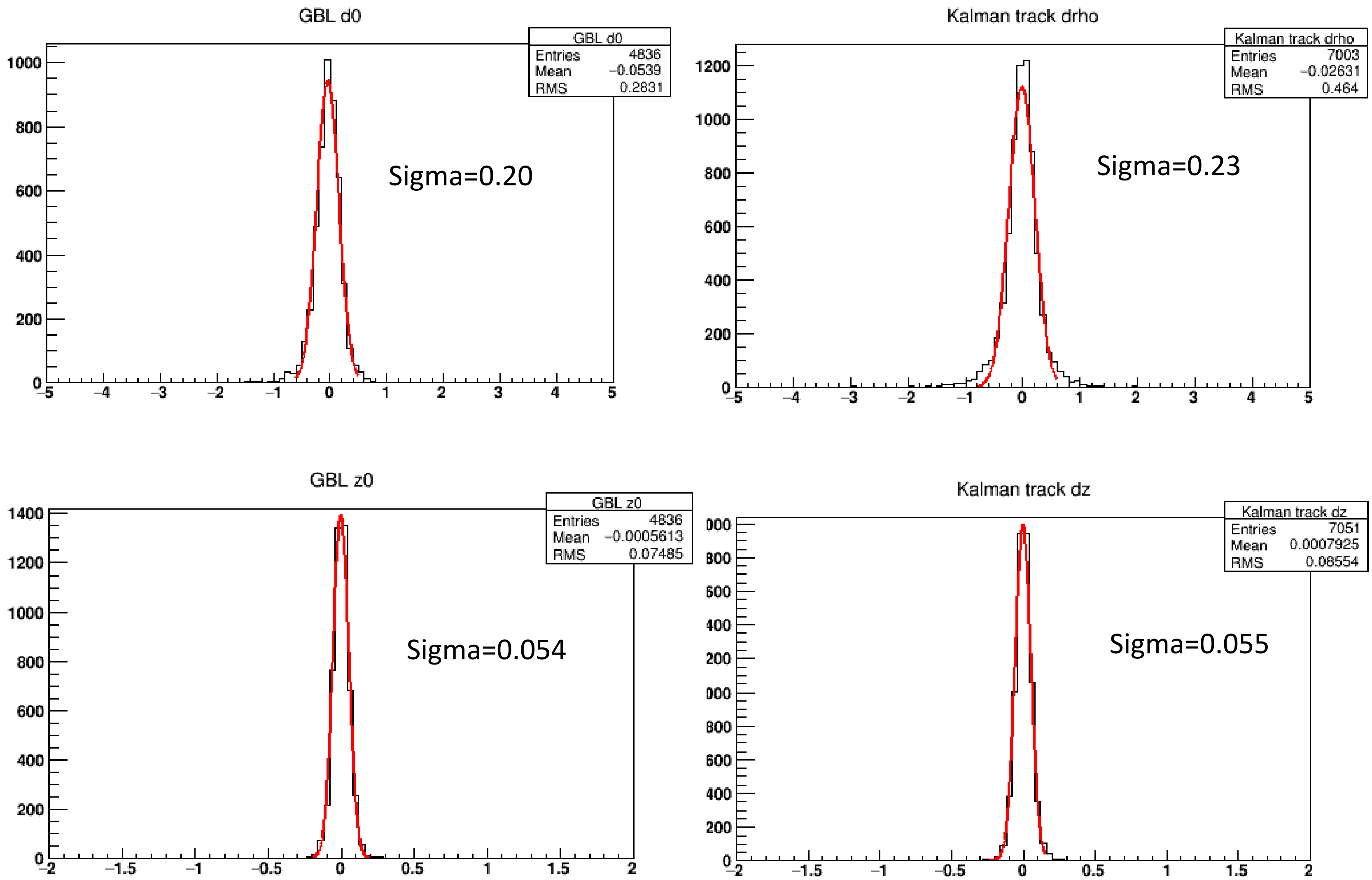






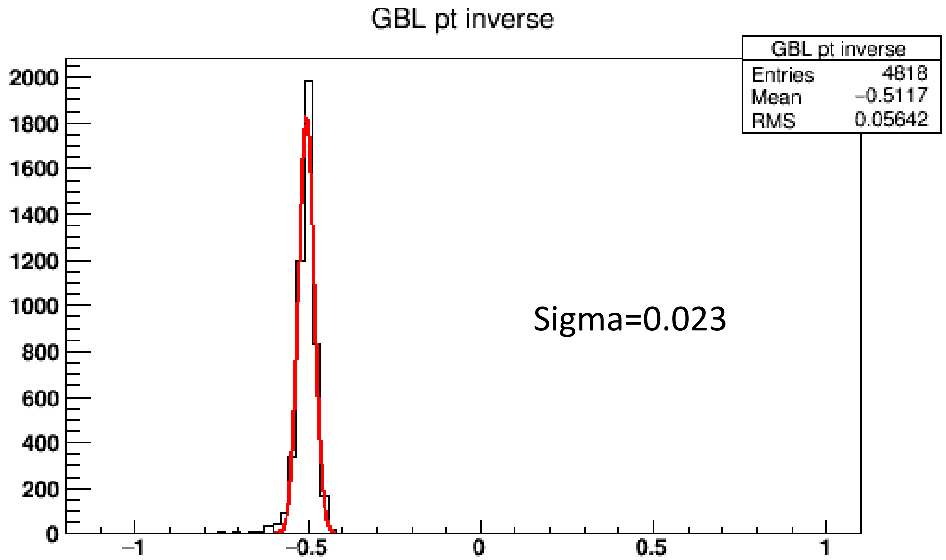


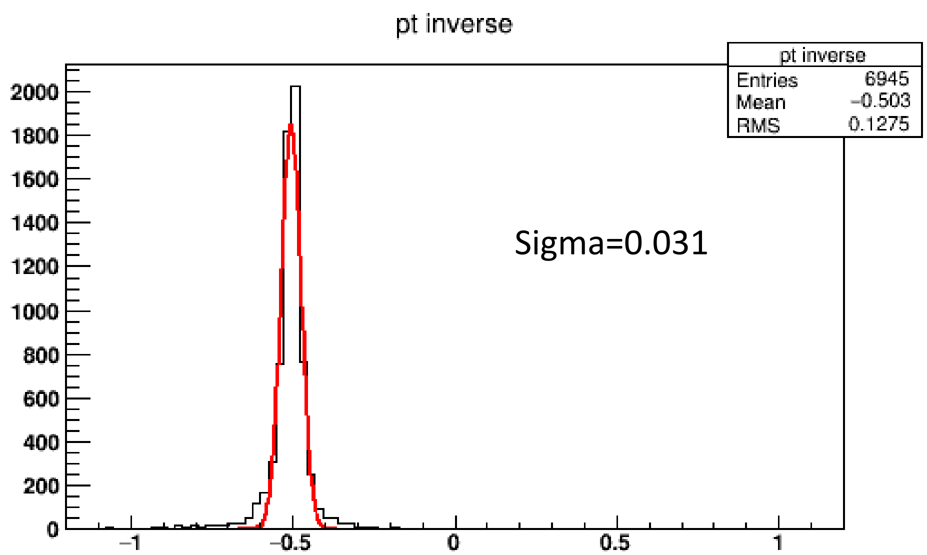
Comparison with GBL Fits (2 GeV electrons)



Similar quality, but GBL a bit better, but with fewer entries.

Comparison with GBL Fits (2 GeV electrons)





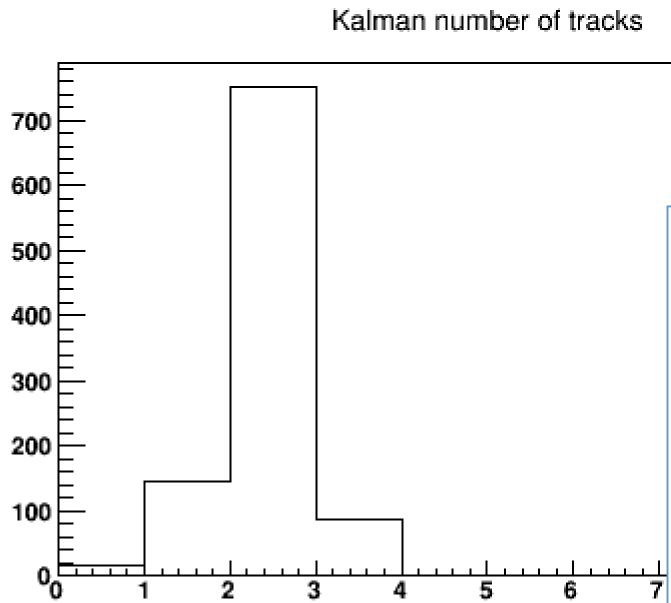
The Kalman distribution has much larger tails.

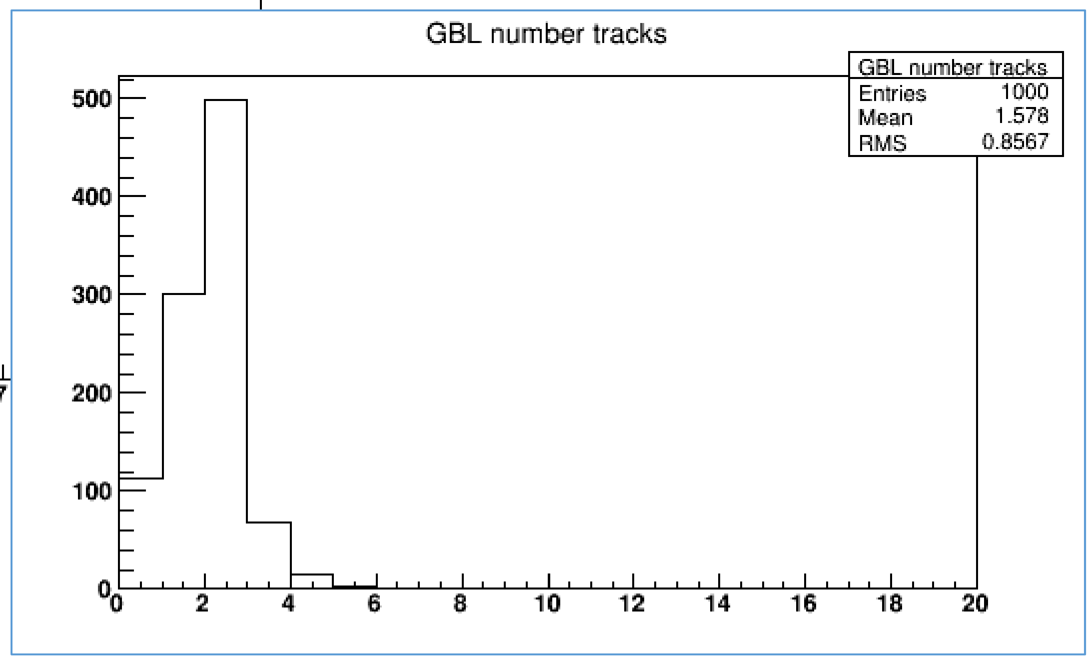
However, it also has 44% more entries from these 10,000 events analyzed.

Pattern Recognition Tests on A’ MC

Detector: HPS-EngRun2015-Nominal-v5-0-fieldmap

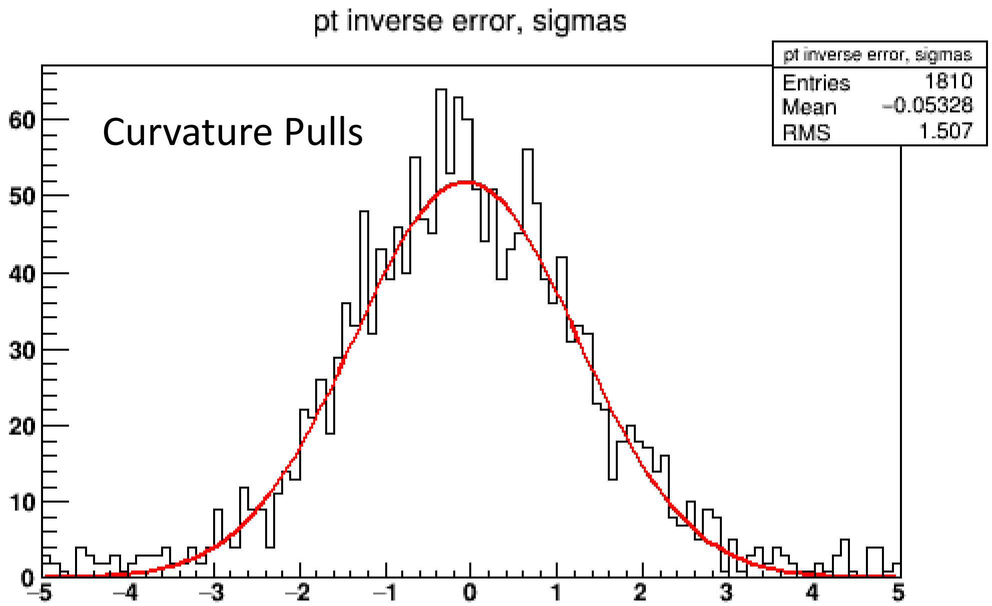
Input: KalmanTest_fullGBL_MC.slcio (from Miriam), 1000 events





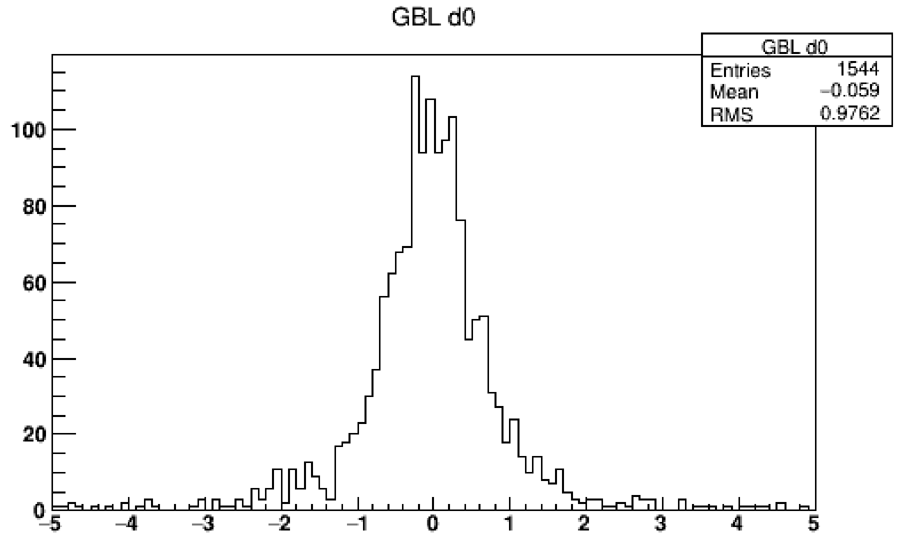
The code already finds more tracks on average than what are in the recon file, and it never finds events with > 3 tracks.

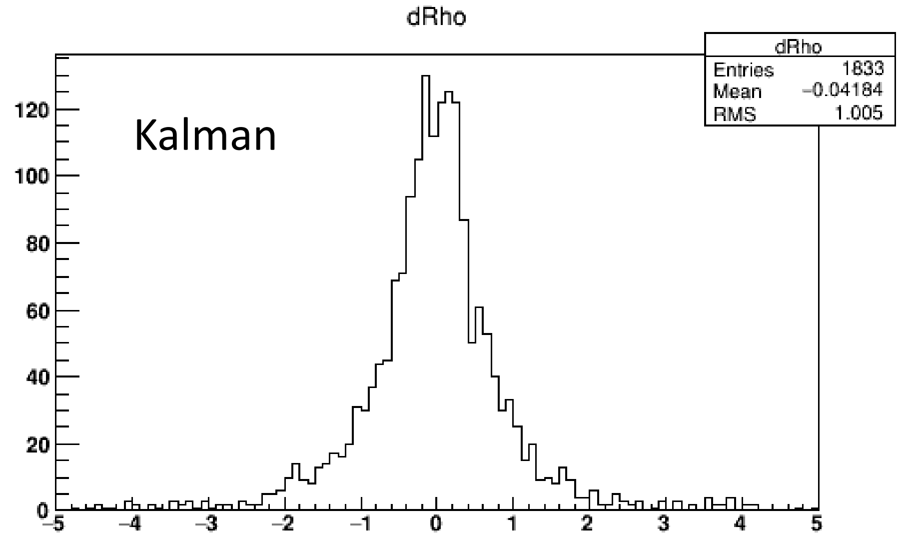
Curvature Measurement in A’ Events

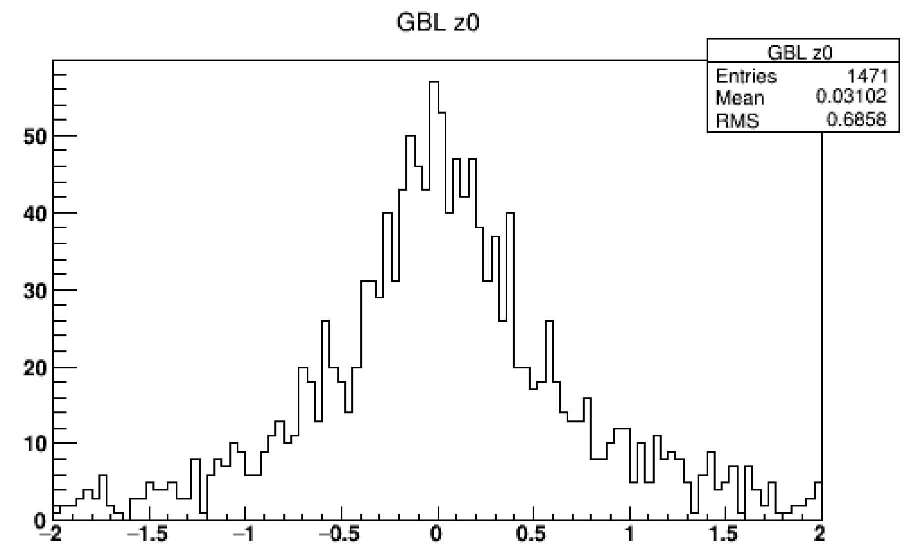


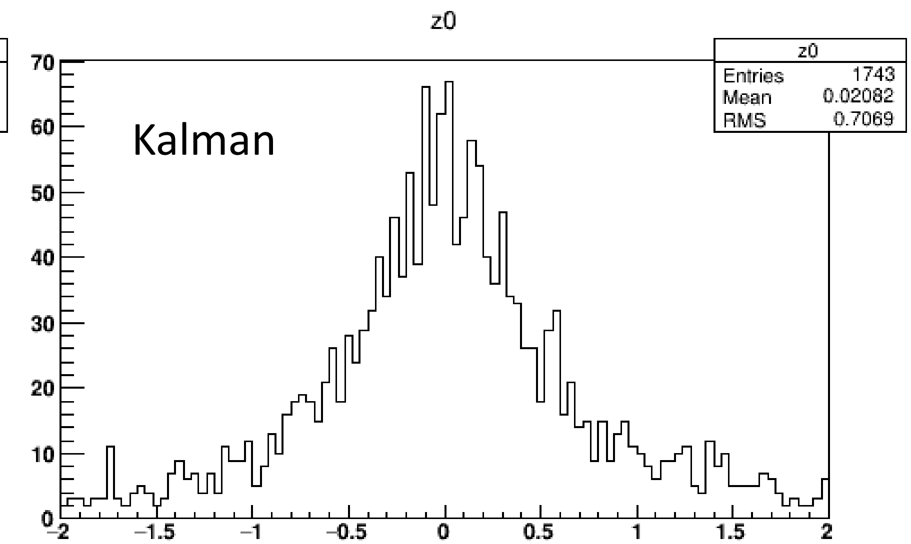
This includes all tracks that were found and fit.
The Gaussian fit has a sigma of 1.3.

Track distances from the origin

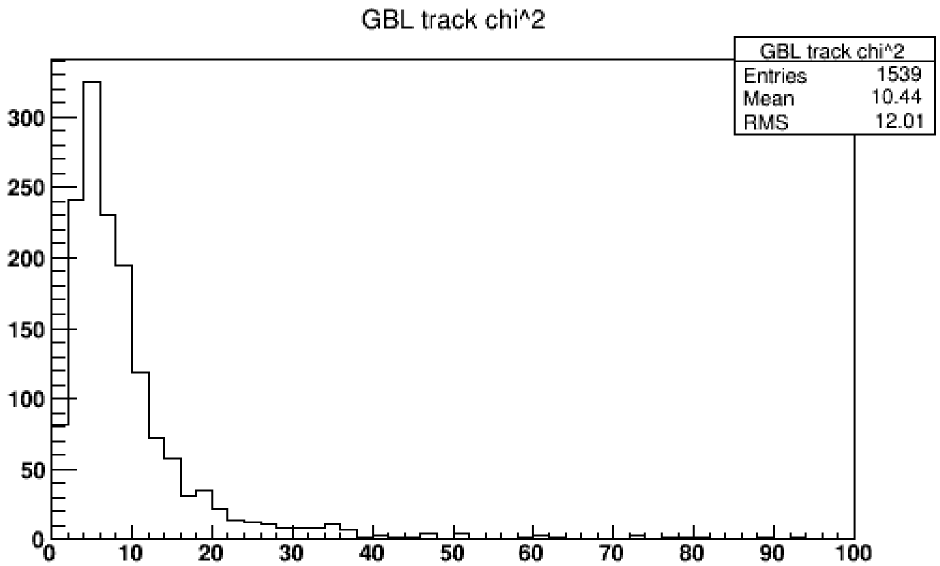


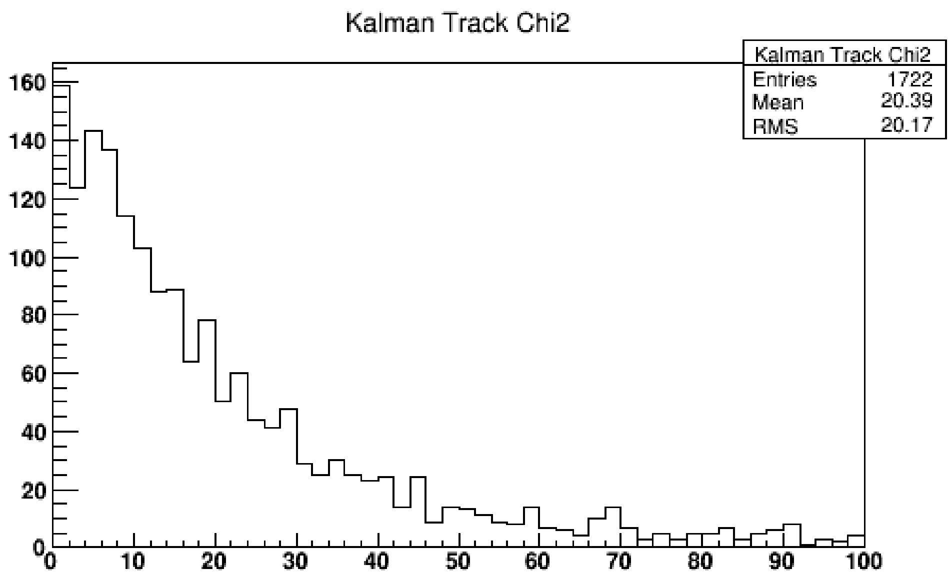






Fit Quality

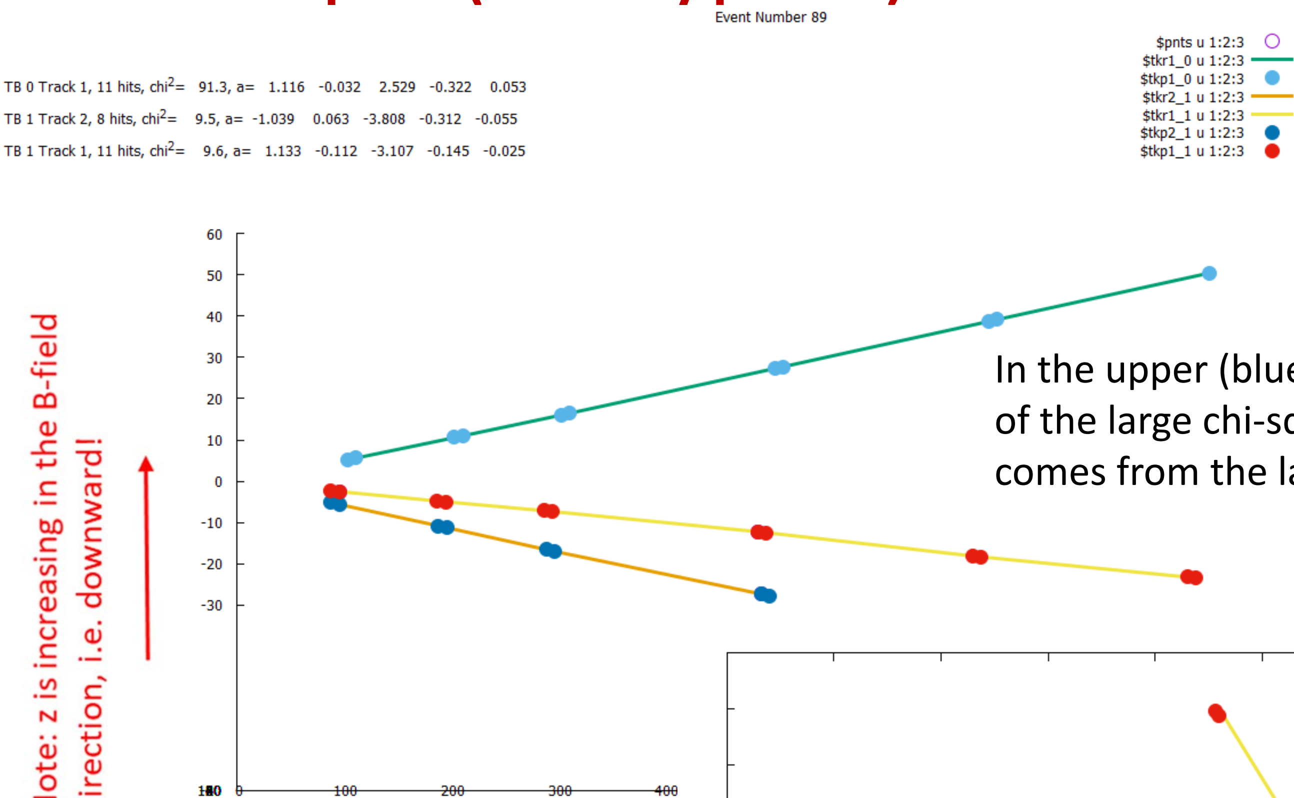




The Kalman chi-squared is generally larger than that of the GBL fit, but I believe this mostly reflects a difference definition of the test statistic between the two.

Nevertheless, the Kalman chi-squared has a significant tail that probably reflects errors in the pattern recognition as well as scattering processes.

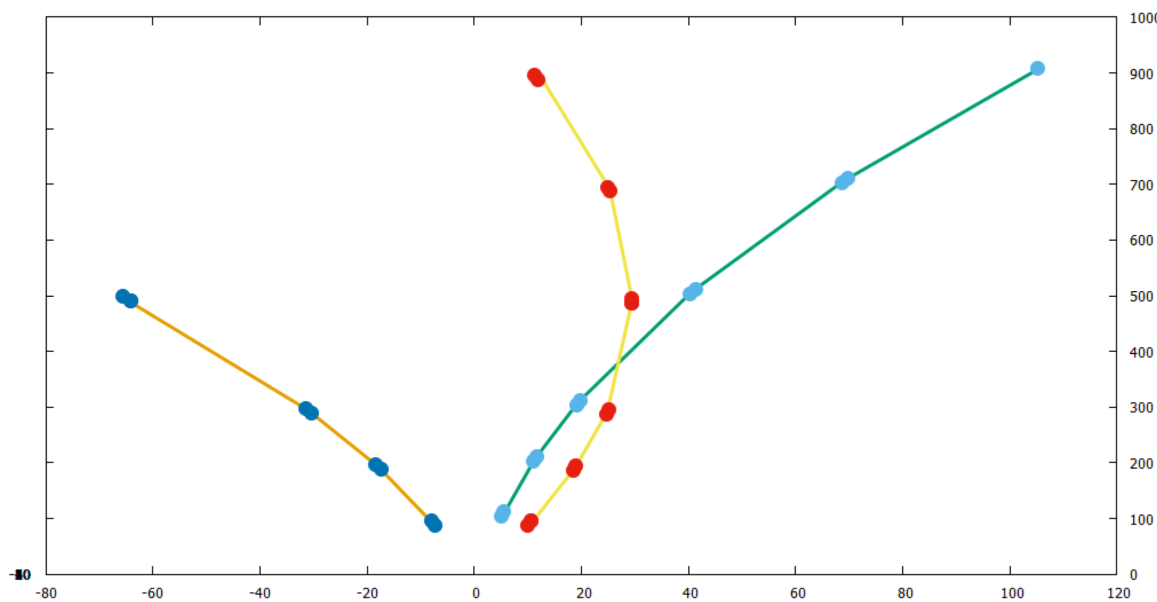
Example (not typical) 2015 A’ MC event



Note: in the non-measured coordinate the hits are placed at the MC true values.

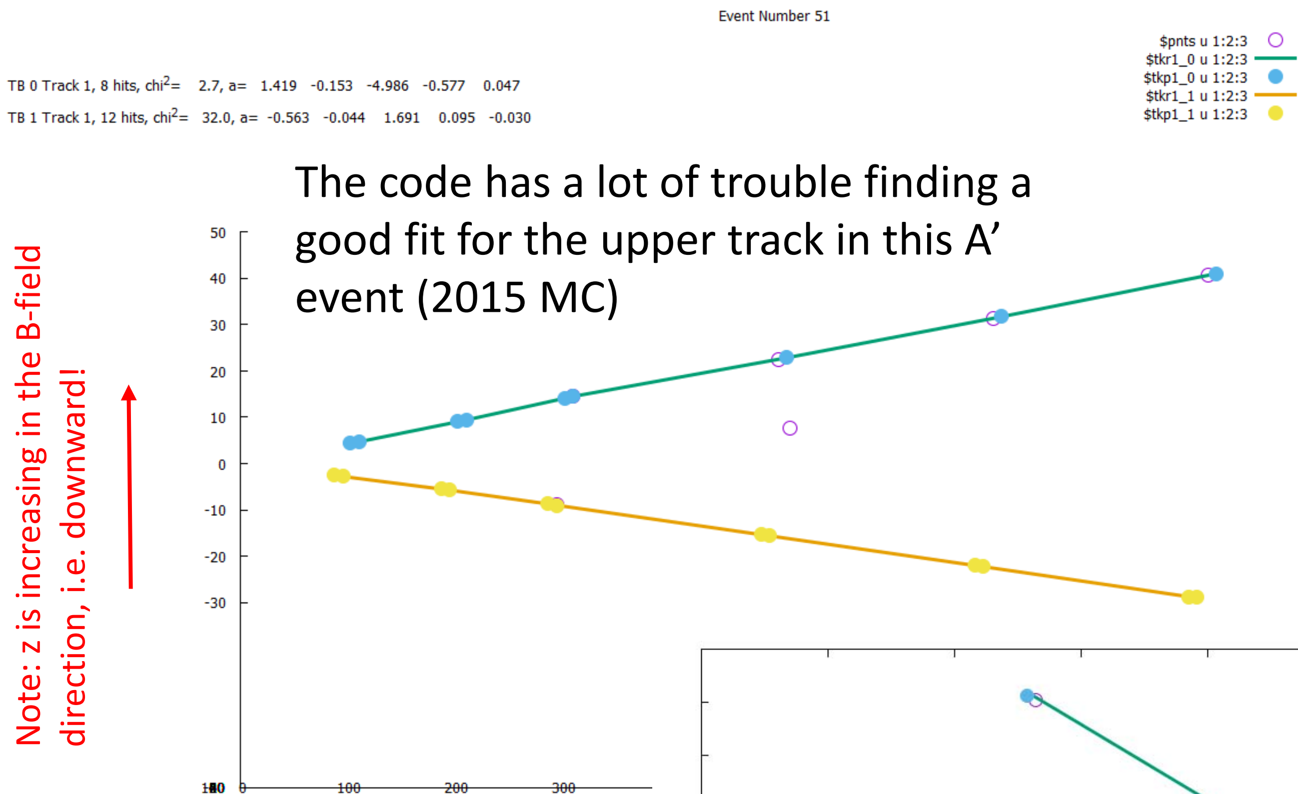
In the upper (blue-green) track, most of the large chi-squared of 91.3 comes from the last, isolated hit.

Note: z is increasing in the B-field direction, i.e. downward!



I intend to do more work on bad-hit removal to improve tracks such as the upper one in this event.

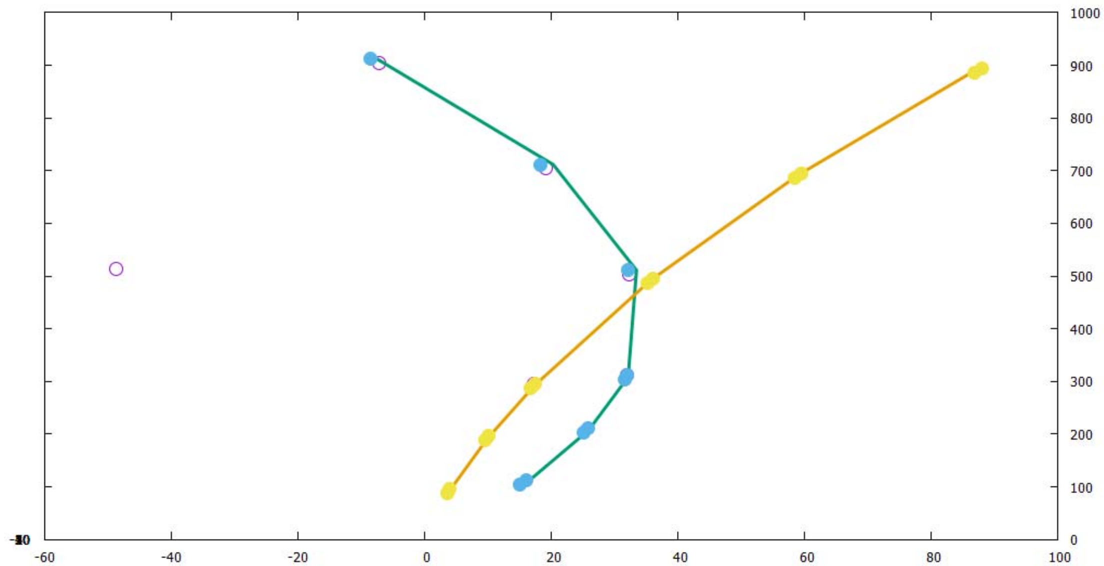
Problematic Events



The code has a lot of trouble finding a good fit for the upper track in this A' event (2015 MC)

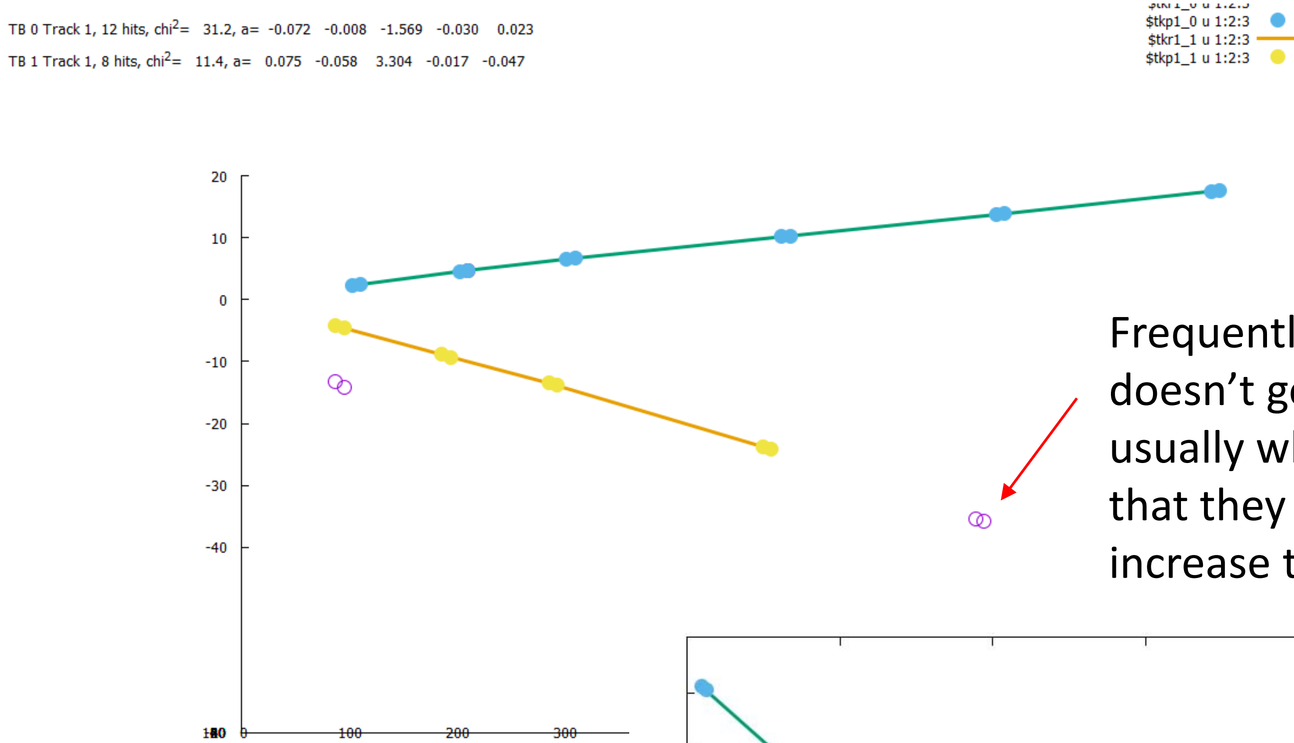
Note: in the non-measured coordinate the hits are placed at the MC true values.

Note: z is increasing in the B-field direction, i.e. downward!

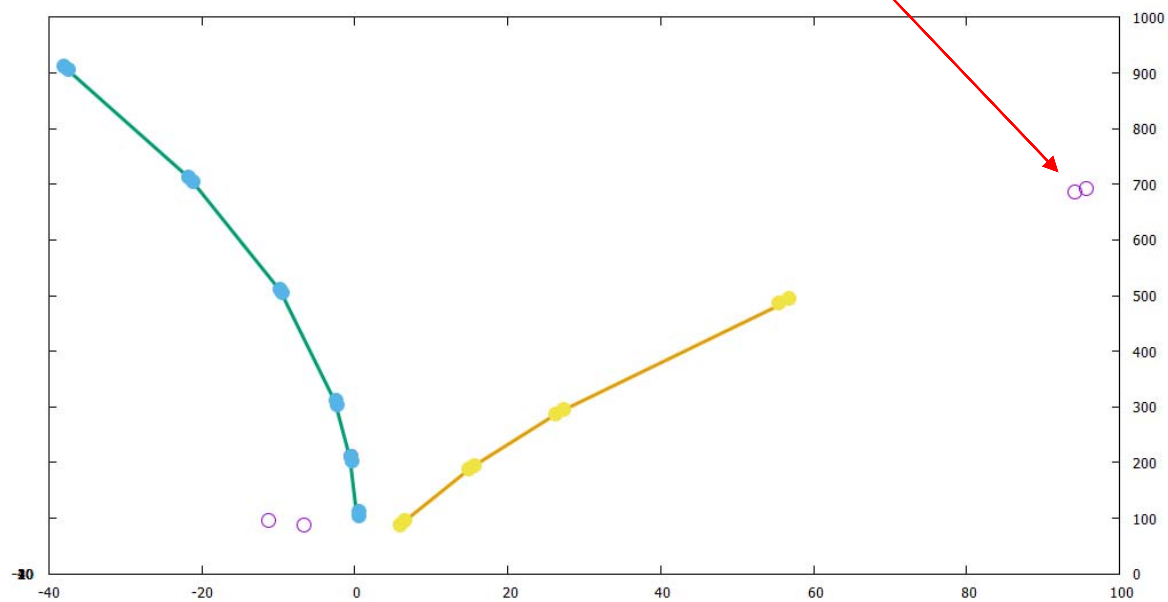


In this event the chi-squared of the upper track jumps into the thousands if the last 3 axial hits are included. I assume there must have been a hard scatter. The stereo hits still fit okay because of the freedom of the curvature (*but I should look at how much the scatter pulled the momentum measurement*).

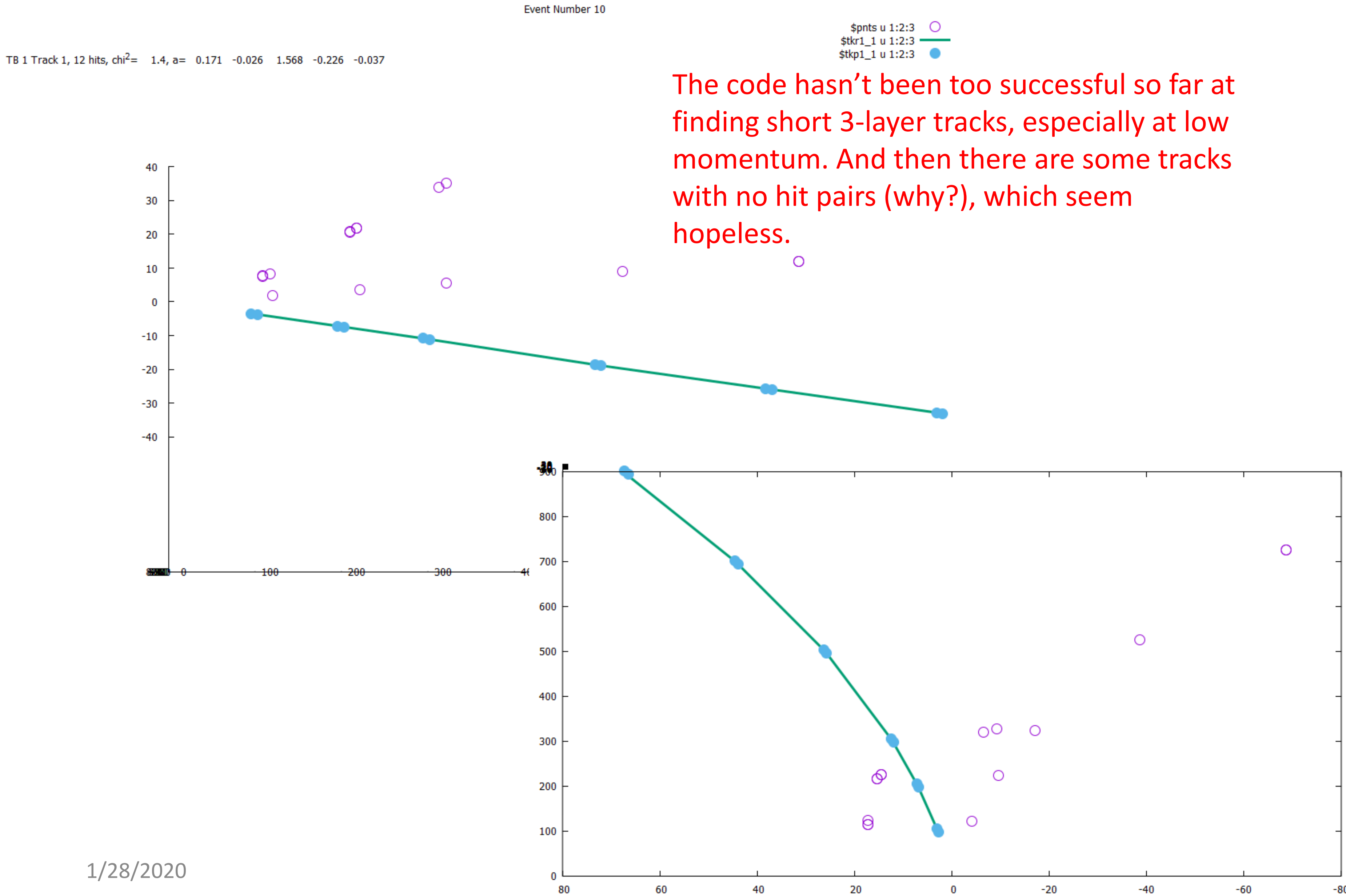
Problematic Events



Frequently the last set of hits doesn't get included, but usually when I look into it I find that they would drastically increase the chi-squared.



Problematic Events



The code hasn't been too successful so far at finding short 3-layer tracks, especially at low momentum. And then there are some tracks with no hit pairs (why?), which seem hopeless.

Conclusions

- The Kalman-Filter code probably doesn't need the non-uniform field complication after all, but it is there and is working…
- The Kalman fit quality is reasonable for single-particle events, although no better than what the GBL fit achieves. The chi-squared mean, unlike in the case with my idealized simulation, is 20% to 30% high, but that is probably due to
  - Non-Gaussian scattering contributions
  - Tails in the hit error distributions, as seen in MC truth.
- The Kalman-fit based pattern recognition works reasonably well on MC A' events:
  - It finds more tracks on average than the existing pattern recognition does.
  - The track quality is about the same as with the GBL fit.
  - I think that some more work tuning it will improve recognition of short, low-momentum tracks and will improve removal of bad hits.