Comparison of KF and GBL-Refit Tracks

Matt Graham
SLAC

June 5, 2023

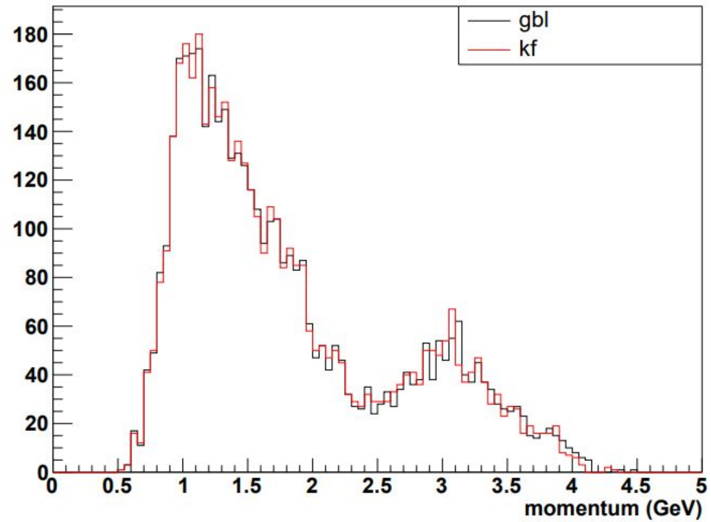
Intro

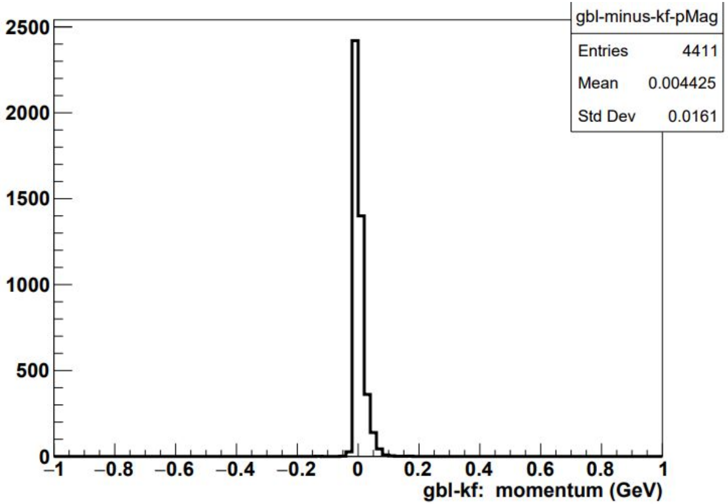
- Alignment people seeing an issue with KF->GBL tracks not matching up
    - first do track finding with KF, then refit same hits with GBL to get input for millepede
    - most striking is the KF chi2 is much larger than GBL
- This is concerning, for sure…is there something wrong with KF? GBL? Just a “feature”? We need to get an answer
- I did a track-by-track comparison of track parameters, residuals+errors etc
    - there are two classes in hps-java that takes in KF tracks and makes GBL tracks
        - SimpleGBLTrajAliDriver (C++ jna) and KalmanToGBLDriver (java port)
        - these give ~the same gbl tracks so I’m not going to discuss … this talk uses plots from KalmanToGBLDriver
- I’m using a file reconn’ed with one of Cameron’s iterations from run14166 for data, and an “ideal detector” MC (2019)
    - IMO it shouldn’t really matter what data/MC or aligned/misaligned for what I’m doing…but I want to check that!
- Much of this info is in JIRA and plots & root files on the web

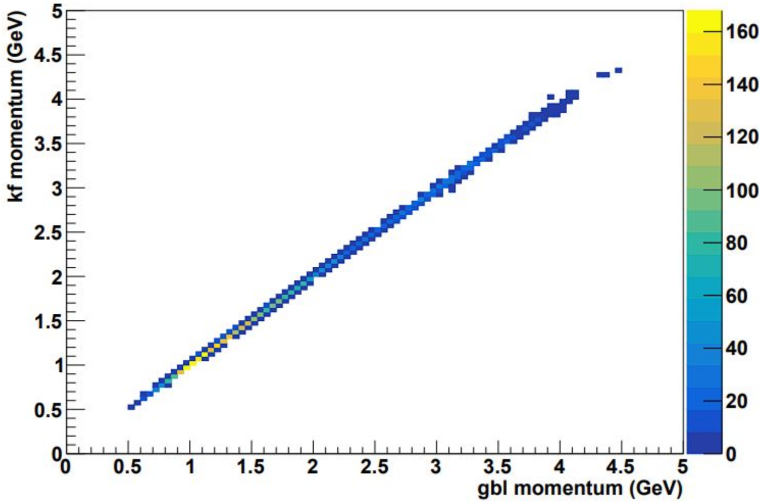
Track Parameter Comparisons

I'll show a few slides like this…
Top Right: raw momentum distribution for KF and GBL-refit
Bot Left: GBL-KF momentum
Bot Right: GBL vs. KF momentum

…this looks fine to me…

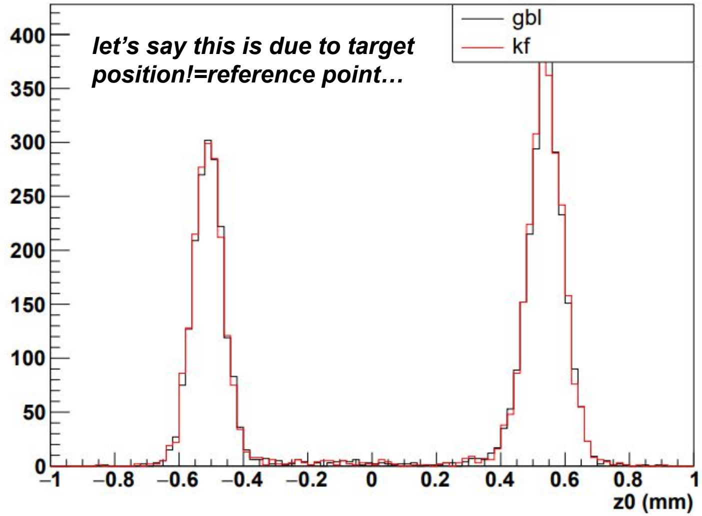




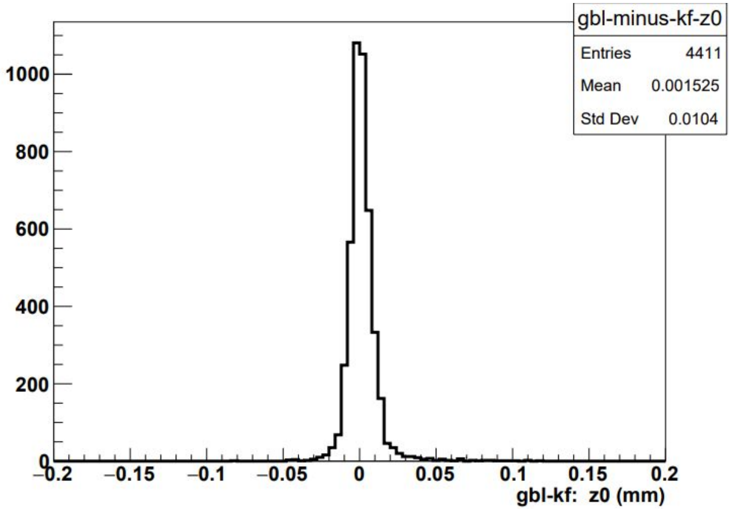


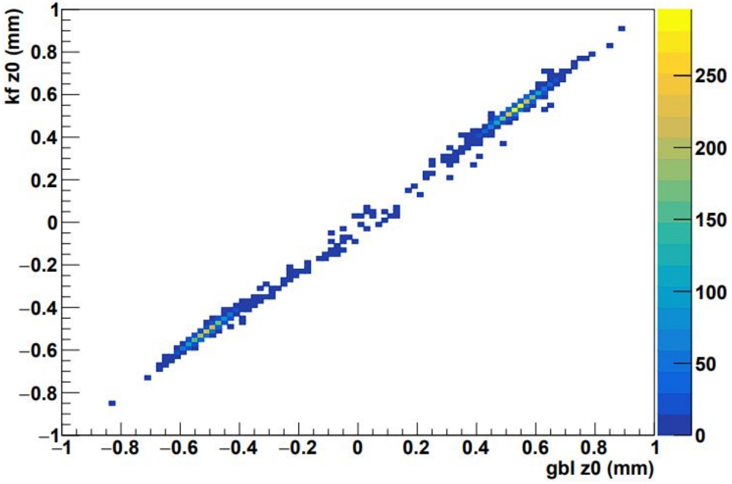
Track Parameter Comparisons

This is z0…the curvature and directions (tanLambda, phi0) show very little spread gbl-kf; the positions (d0, z0) seem to have a bit larger spread. Still small enough that I'll call it ok.



let's say this is due to target position!=reference point…

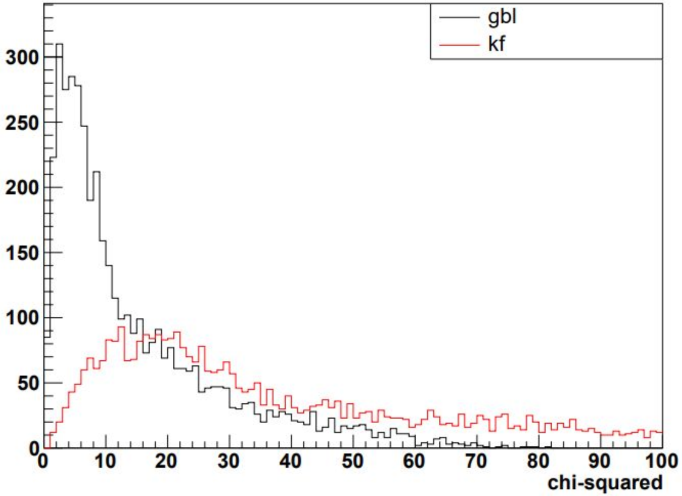


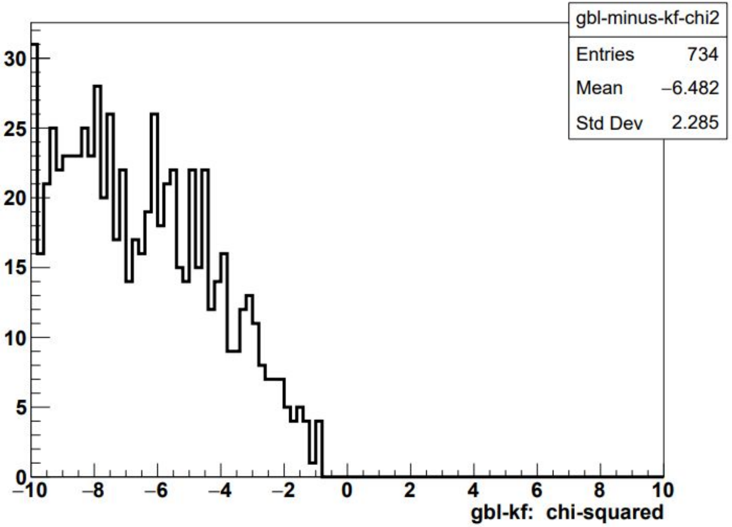


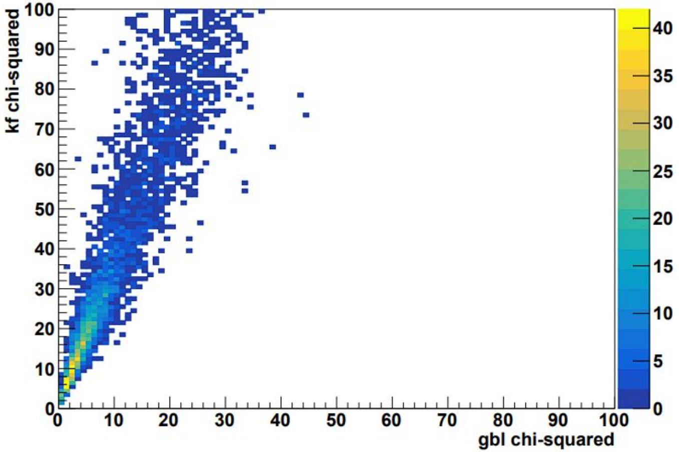
Chi2 Comparisons

Here's the good stuff…chi2 in GBL-refit is much lower than KF (GBL is roughly correct for this NDF)

What is going on?



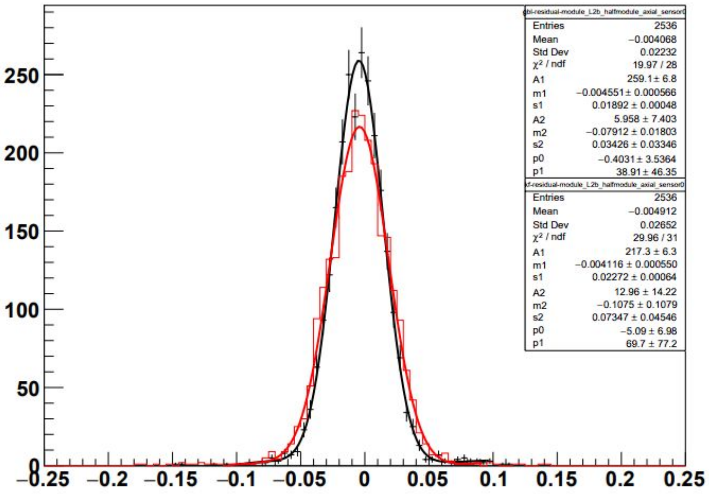


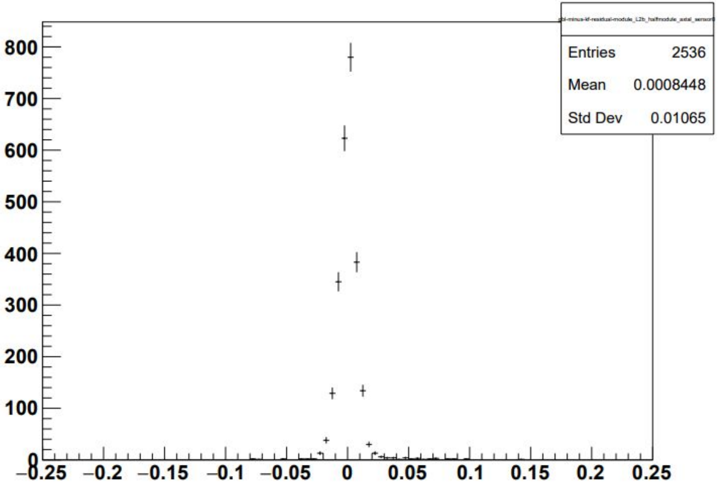


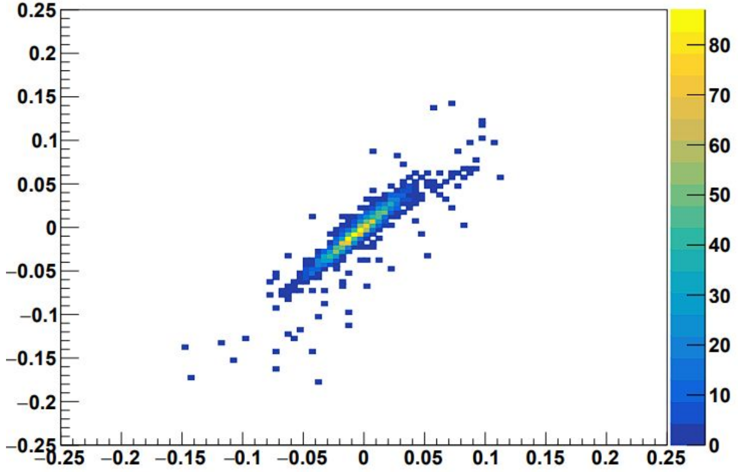
Unbiased Residual Comparison

This is a typical residual comparison for example…it's from L2b-axial, but others have same features**

KF residuals are a bit broader than GBL-refit…~10-20%





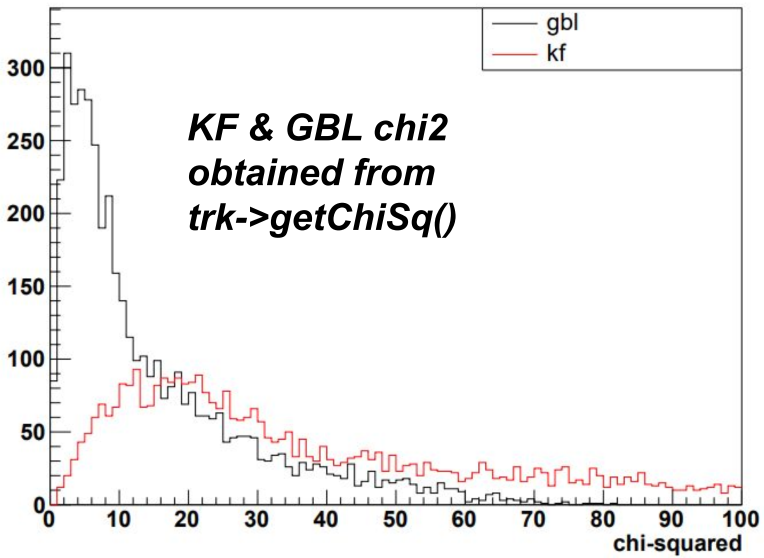


Chi2 from-unbiased-residuals

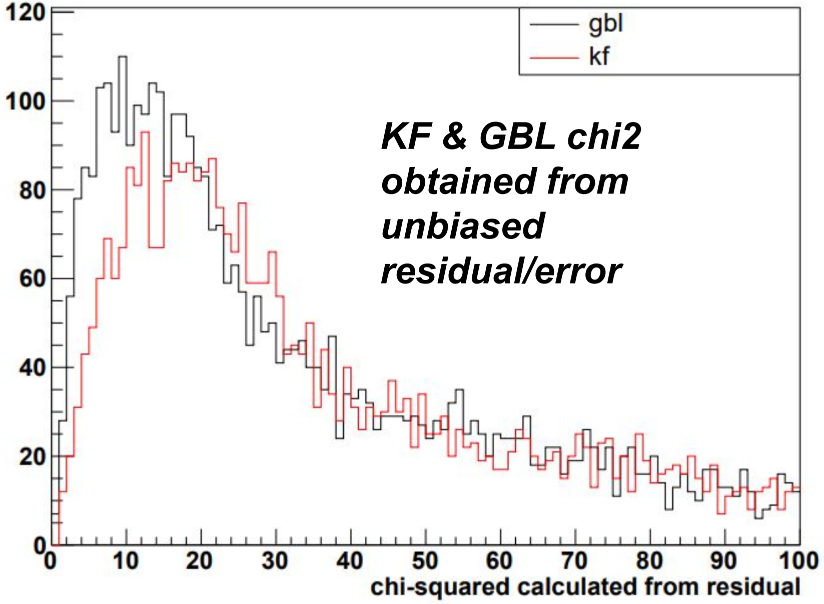
On right, I calculate chi2 as sum_over_hits <unbiased-residual>/<error-on-residual>

Why would I do this? It's not the correct way to calculate chi2 (thanks to PF for setting my straight).
…anyway I did it. GBL matches much better to KF now (but not perfectly).

you can see from the plot what happens to GBL chi2…KF it's harder to see



KF & GBL chi2 obtained from trk->getChiSq()

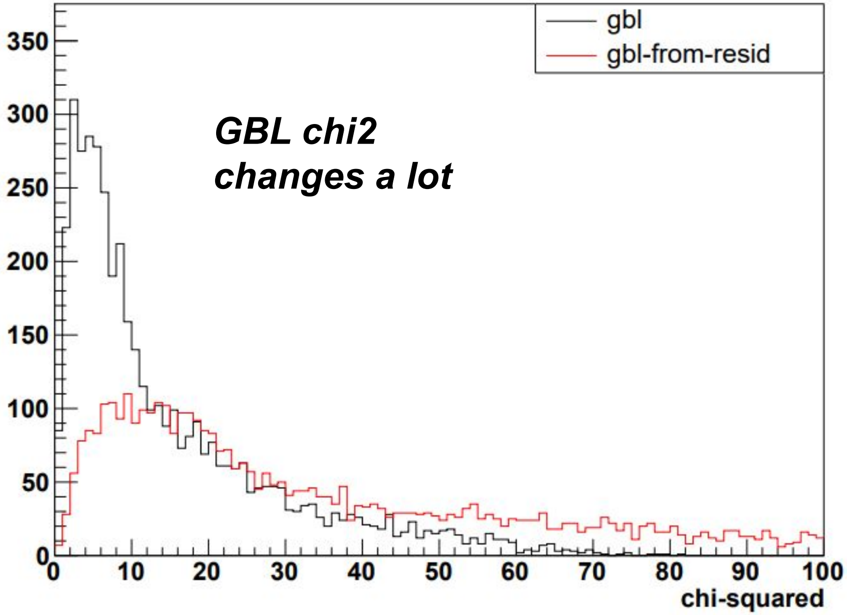


KF & GBL chi2 obtained from unbiased residual/error

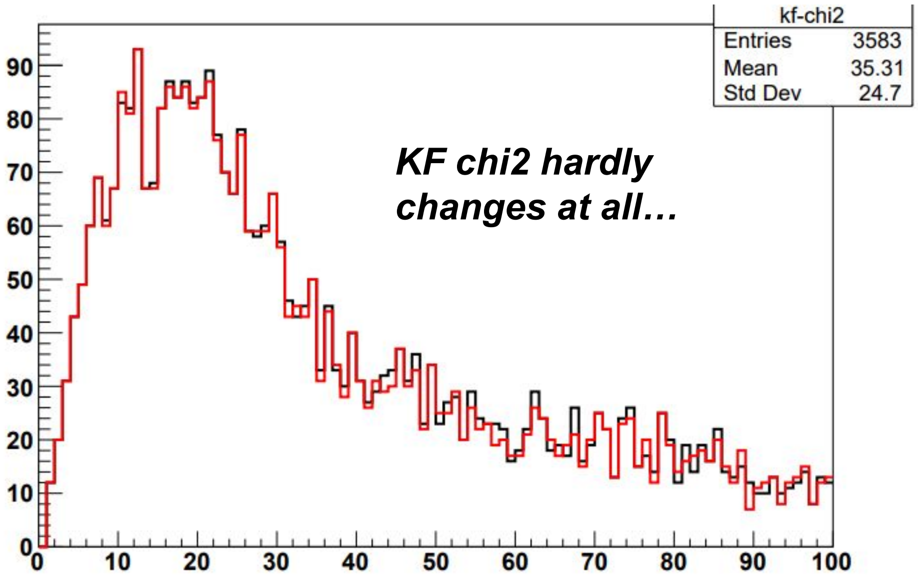
Chi2 from-unbiased-residuals: from track vs. by hand

Weirdly, the KF code seem to calculating the chi2 using the unbiased residuals. That seems wrong…it should be from ***biased*** residuals (right?).

I didn't have biased residuals saved for KF and the supposedly biased residuals for GBL were very weird…so I wasn't able to compare.



GBL chi2 changes a lot



KF chi2 hardly changes at all…

So what does it all mean…

- It looks like the gbl-refit isn’t changing the track parameters too much
- The unbiased residuals are maybe a bit weird…KF ~10-20% broader
- Big question on how chi2 is being calculated in GBL vs KF
  - have we never compared these?
  - from Robert’s toy studies, the KF chi2 came out looking fine…has something changed in code or are we using it incorrectly?
- How much does this matter for alignment using KF tracks?