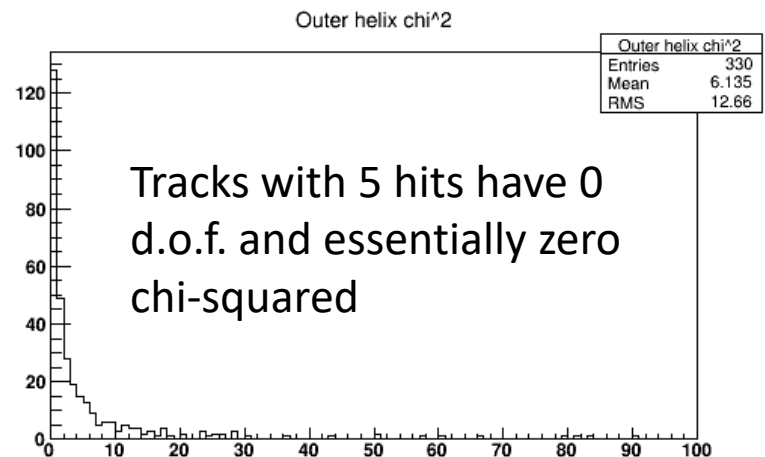
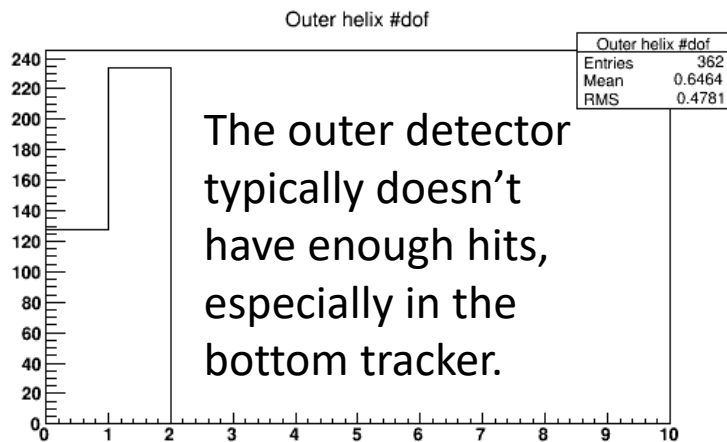
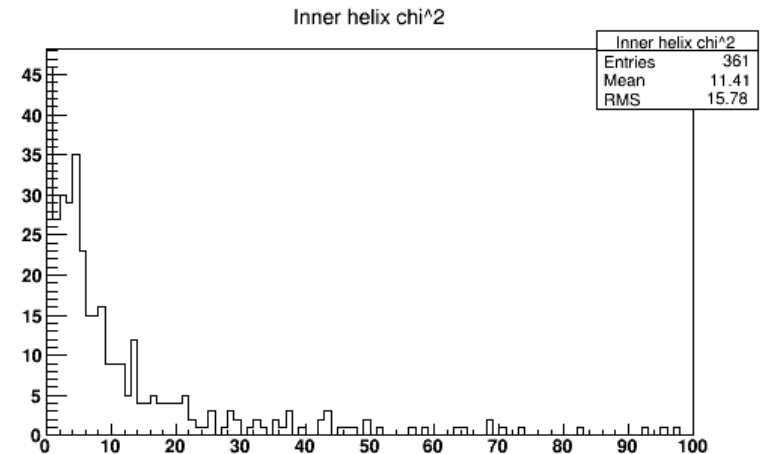
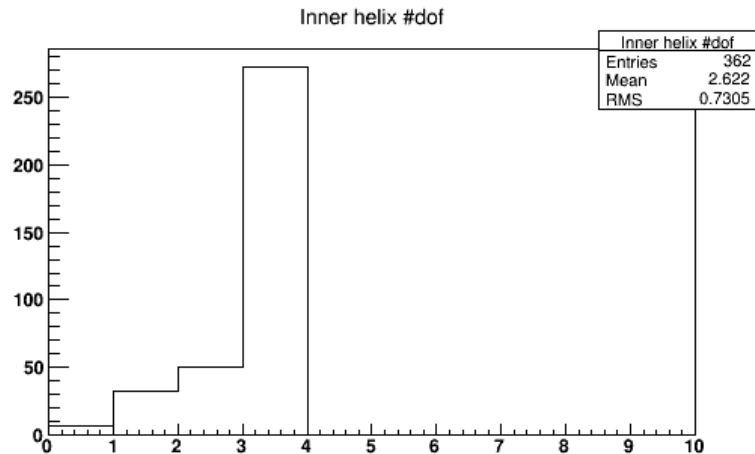


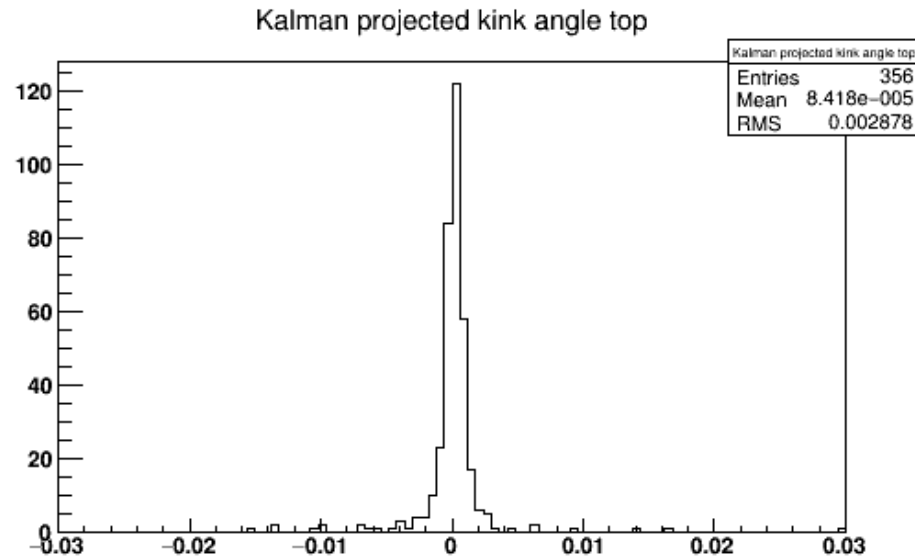
Fitting the kink between front and back

- KalmanKinkFit.java and KalmanKinkFitDriver.java
- Already merged into Master in Github
- Can operate from slcio files, using the above driver, or can be called following the Kalman pattern recognition
- Algorithm:
 - Create the Kalman SiModule data structure for each sensor, if not already done.
 - Read in 1D hits and Kalman hit relations, and fill the hits into the Kalman data structure, making separate lists for the front (layers 1-8) and back halves (layers 9-14) of the detector
 - Require at least 3 stereo and 2 axial hits in each half
 - Run the Kalman fit separately for the two halves
 - Calculate the momentum for each half, at a plane midway between layers 8 and 9
 - Calculate the angle in the y,z plane between the two halves
 - Would be easy to add calculation of offsets.

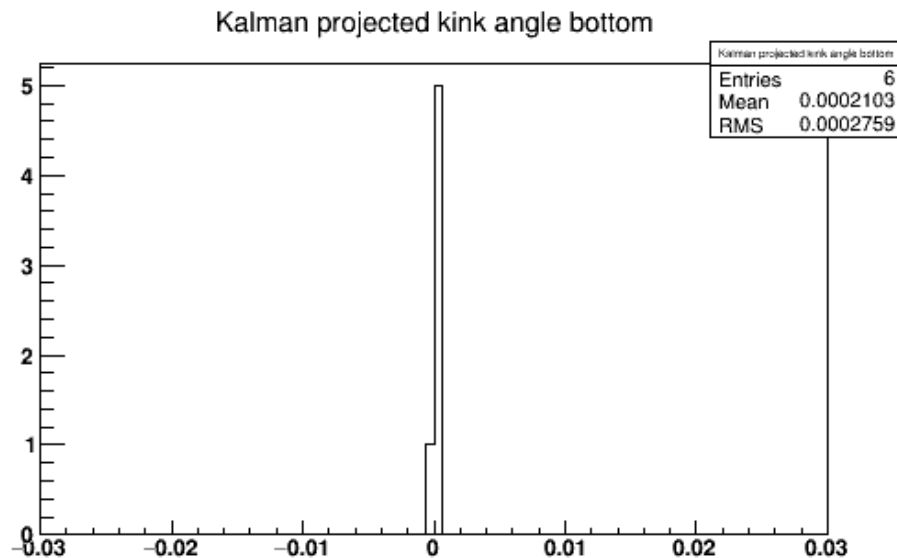
Testing on 2019 calibrated data, run 10030



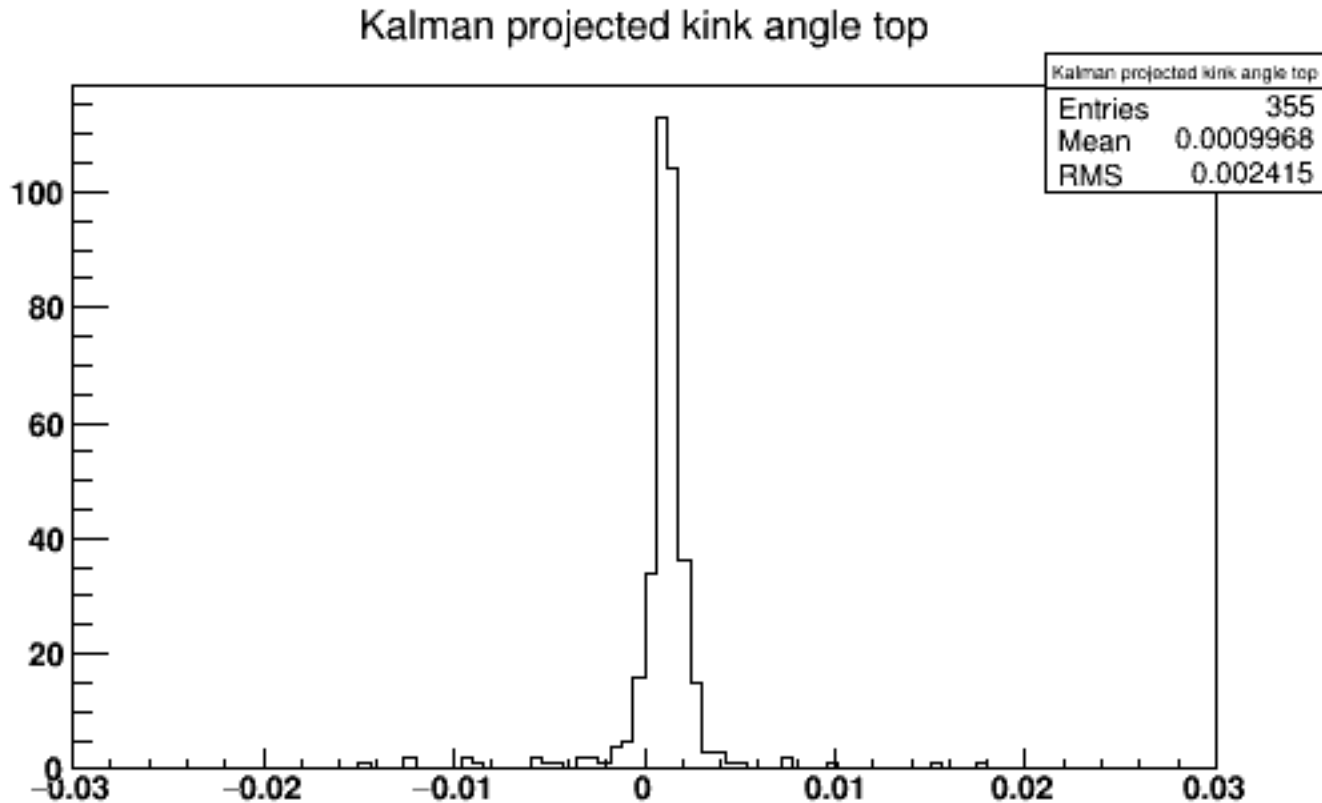
Testing on 2019 calibrated data, run 10030



This confirms that the alignment process is getting this angle correctly set to zero.



Test with top tracker kinked by 1 mrad



Sanity check: the new code reconstructs the correct kink angle when the detector is deliberately kinked by a known amount (thanks to PF).