BMOD Data Inclusion Story

Beam Modulation causes the beam to dither by ~mm scales in X, Y, and energy (BPM 12X, etc.). While modulating the position differences are often much larger than when not modulating, as the phase of the modulation is not locked to the QRT pattern sync (but could have been and should be in future experiments). Several metrics of the data, from monitors, asymmetries, and corrections, are investigated to motivate including the BMOD active data into the final Apv dataset.

Outliers and their impact on the data set, with and without including BMOD

Multipletwise

Throughout CREX there are occasional outliers in the multipletwise distribution at the several thousand ppm level, which is not typically seen during PREX II. These outliers show up in the us_avg distribution, but not the us_dd distribution, leading one to believe that they are due to correlated noise in either the Y or Energy BPMs. A dedicated study of individual runs and events is required to carefully evaluate whether events need to be thrown out and if they have any noteworthy negative impact on the miniruns in which they reside (as our averaging technique looks at minirunwise data everywhere).

The multipletwise pull plots for ErrorFlag data set (ErrorFlag == all good events, not including BMOD), IncludeBMOD (ErrorFlag data set + BMOD Active) and OnlyBMOD (only the BMOD Active dataset) are included below:

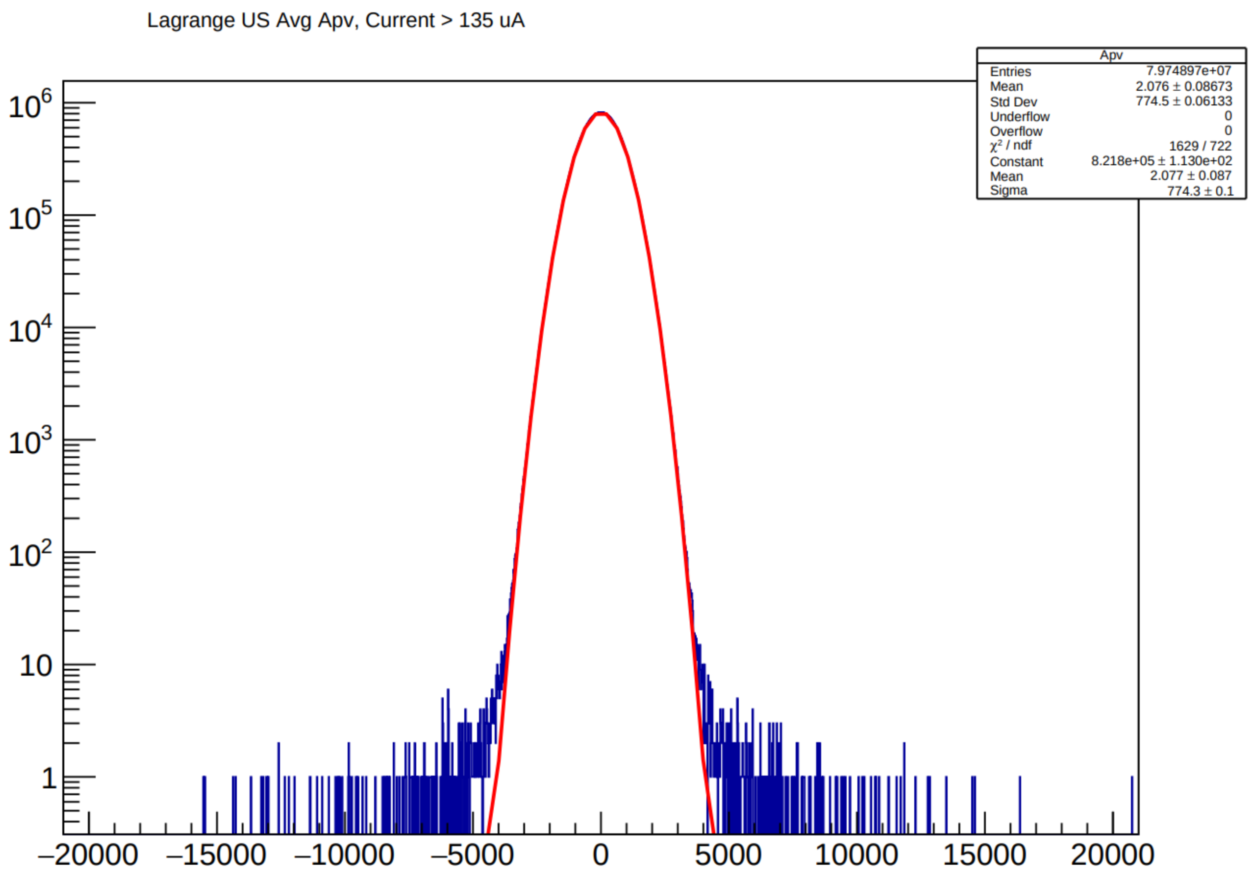


Fig 1. Lagrange US Avg Apv multiplet plot with current cut >135 uA. ErrorFlag dataset cut.

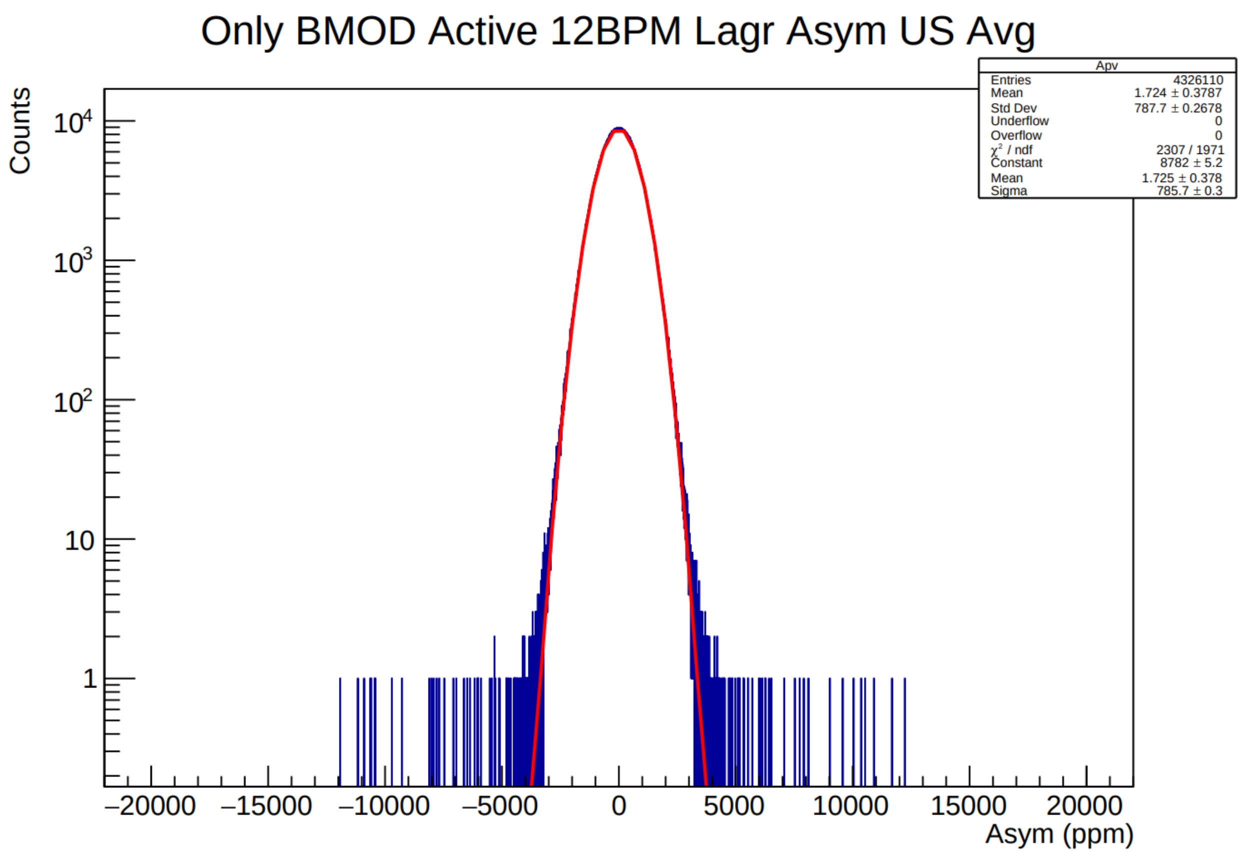


Fig 2. Lagrange US Avg Apv multiplet plot with current cut > 135 uA. OnlyBMOD dataset cut.

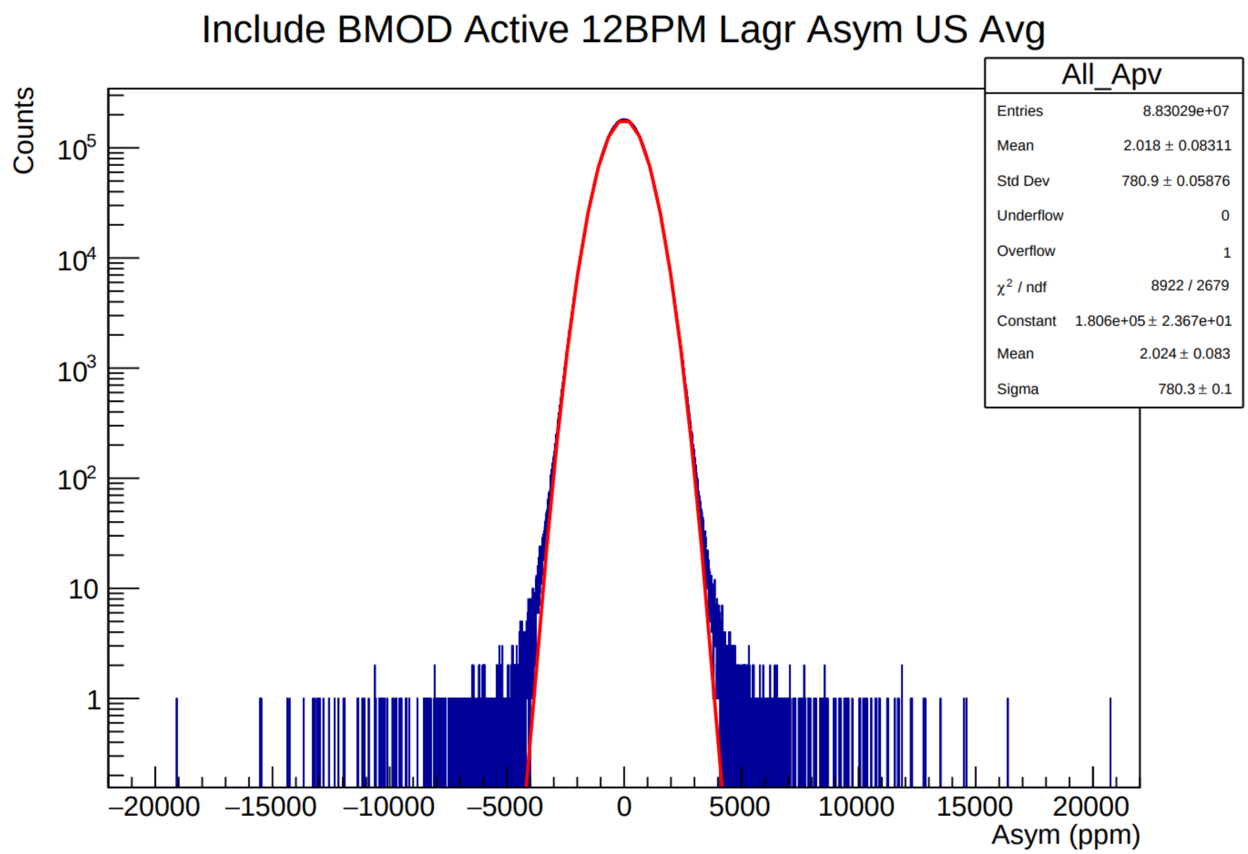


Fig 3. Lagrange US Avg Apv multiplet plot with current cut > 135 uA. IncludeBMOD dataset cut.

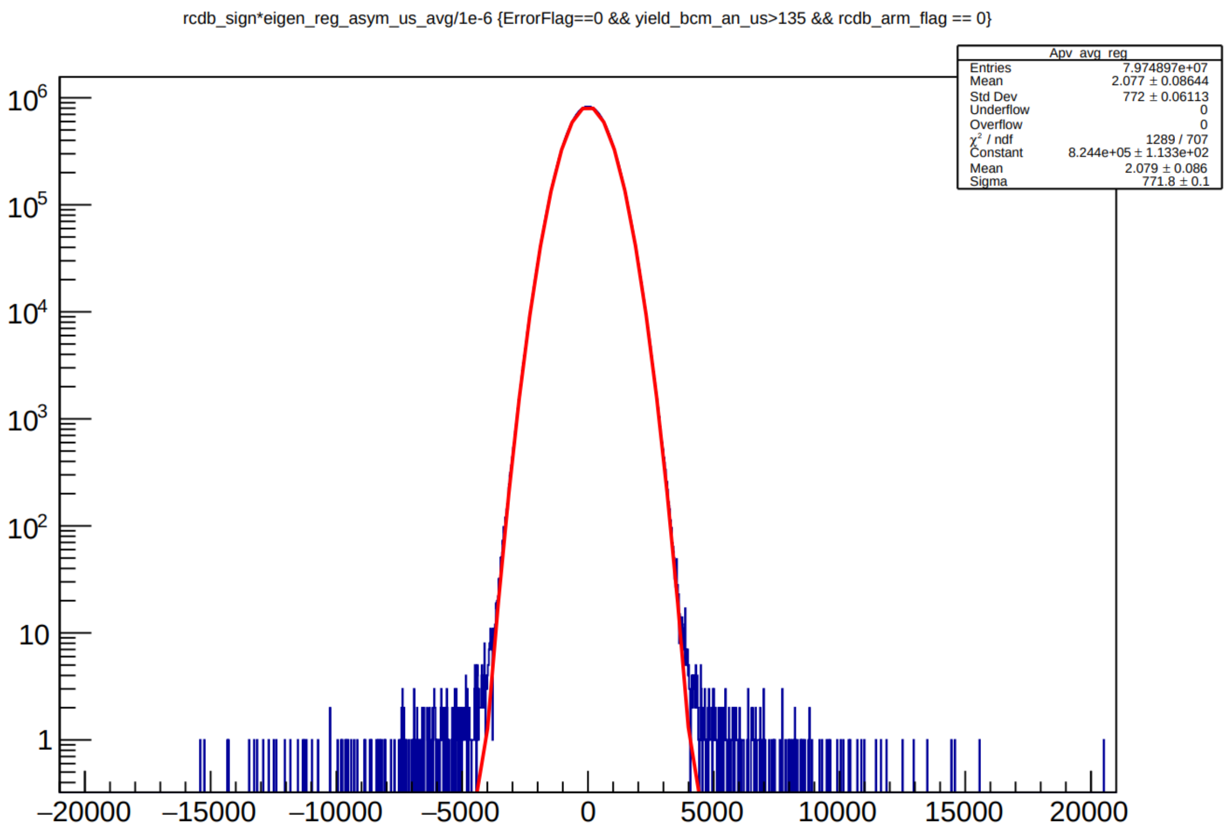


Fig 4. Regression US Avg Apv multiplet plot with current cut > 135 uA. ErrorFlag dataset cut.

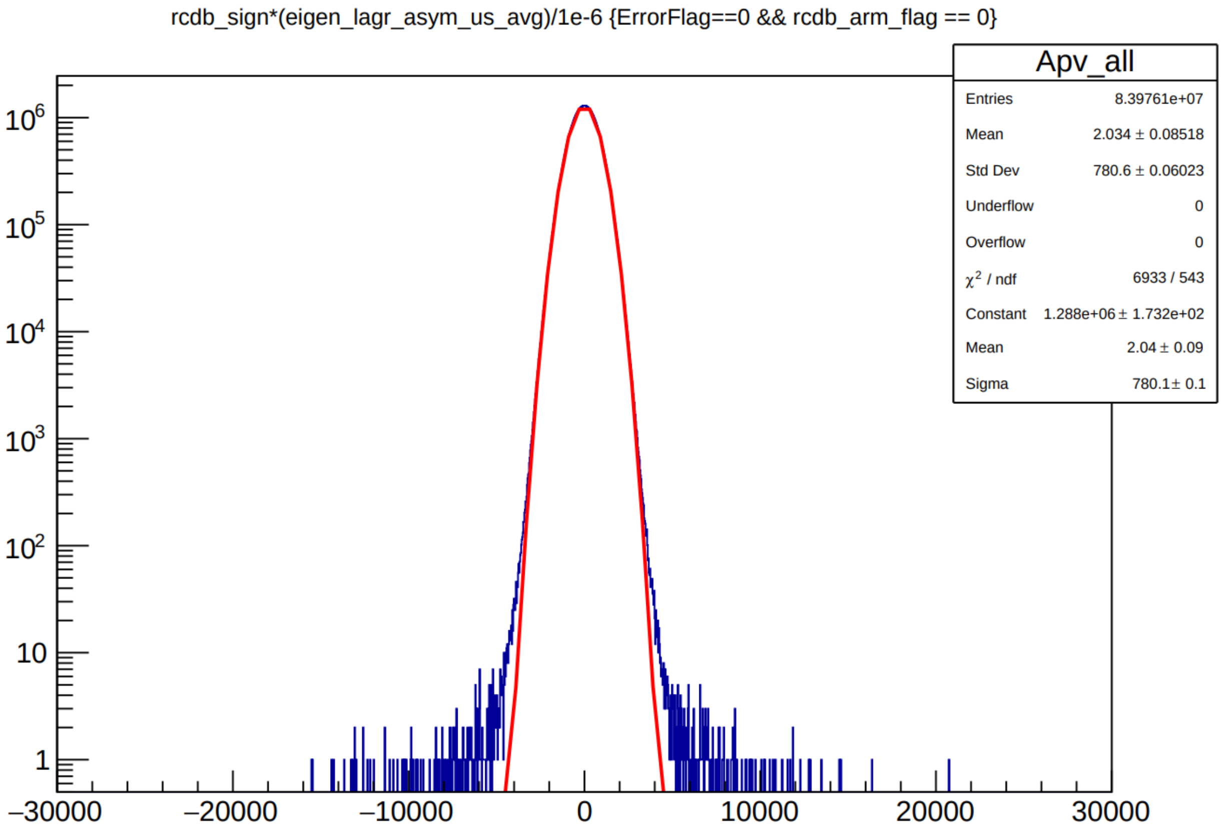


Fig 5. Lagrange US Avg Apv multiplet plot with no current cut. ErrorFlag dataset cut.

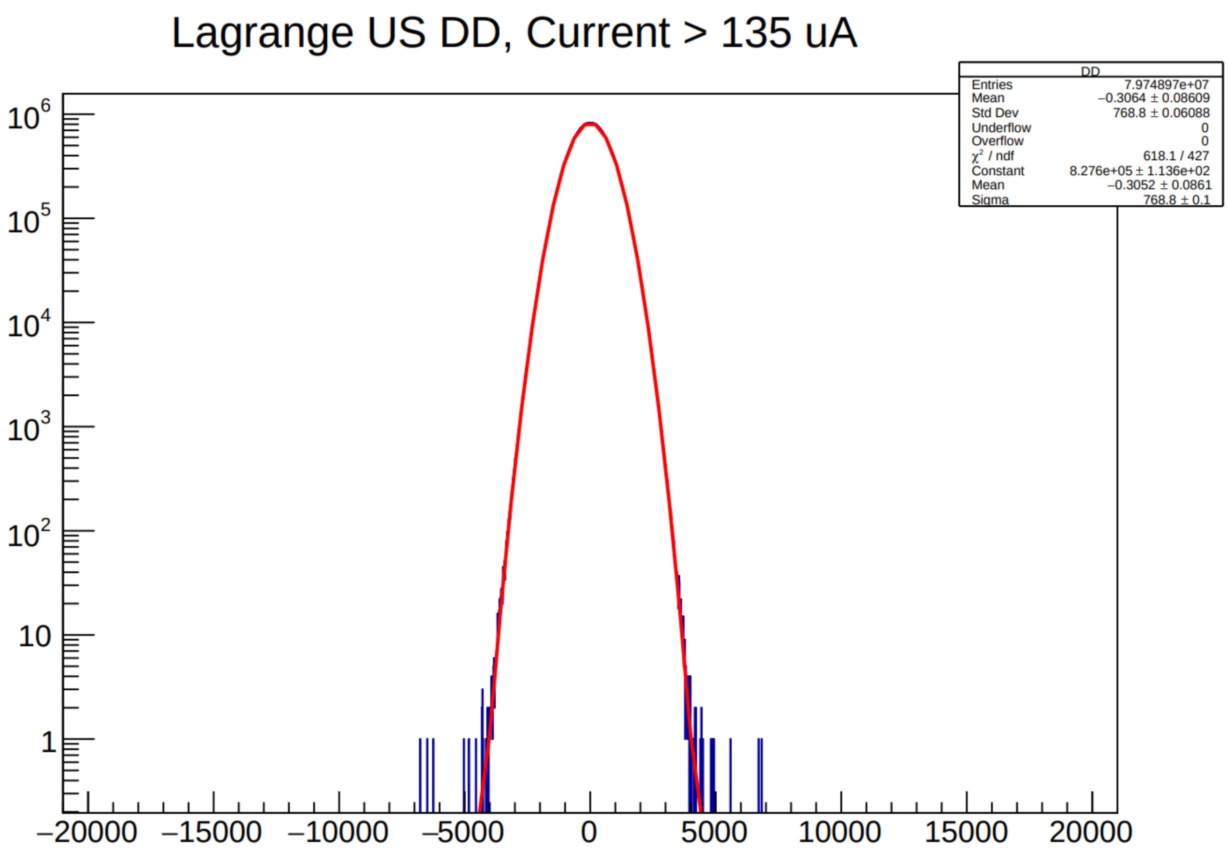


Fig 6. Lagrange US DD multiplet plot with > 135 uA current cut. ErrorFlag dataset cut.

And so we see that the outliers are not a function of Lagrange (as they show up in regression as well) and are not an artifact of low current running, but are correlated between the two arms, as they go away when looking at the double difference (fig 6). The BMOD inclusion has a negligible inclusion of this effect, though it is still there in this additional data. The counts of number of events with these large outliers are not too large: the ErrorFlag dataset has 46 multiplets outside of 10,000 ppm from 0, while the OnlyBMOD dataset has 13 entries outside of 10,000, though 11 of them come from one run, run 6564's minirun 4 (5th minirun in the run) ***which definitely needs to be investigated***. The ErrorFlag dataset has 210 entries that are 6,000 ppm or more outliers, while the OnlyBMOD dataset has 45 outliers at 6,000 ppm, of which 36 are from the same run 6564.4. So including BMOD in the dataset does not have any noticeably worse effect on the multipletwise outliers, other than this one clearly problematic run which deserves its own investigation. The full lists of 6,000 and 10,000 ppm outliers run+minirun information for ErrorFlag and OnlyBMOD dataset cuts are available as 6/10ppm_ErrorFlag/OnlyBMOD_multipletwise_outliers.txt files

Minirunwise

Minirunwise, the main detector pull plot looks very good:

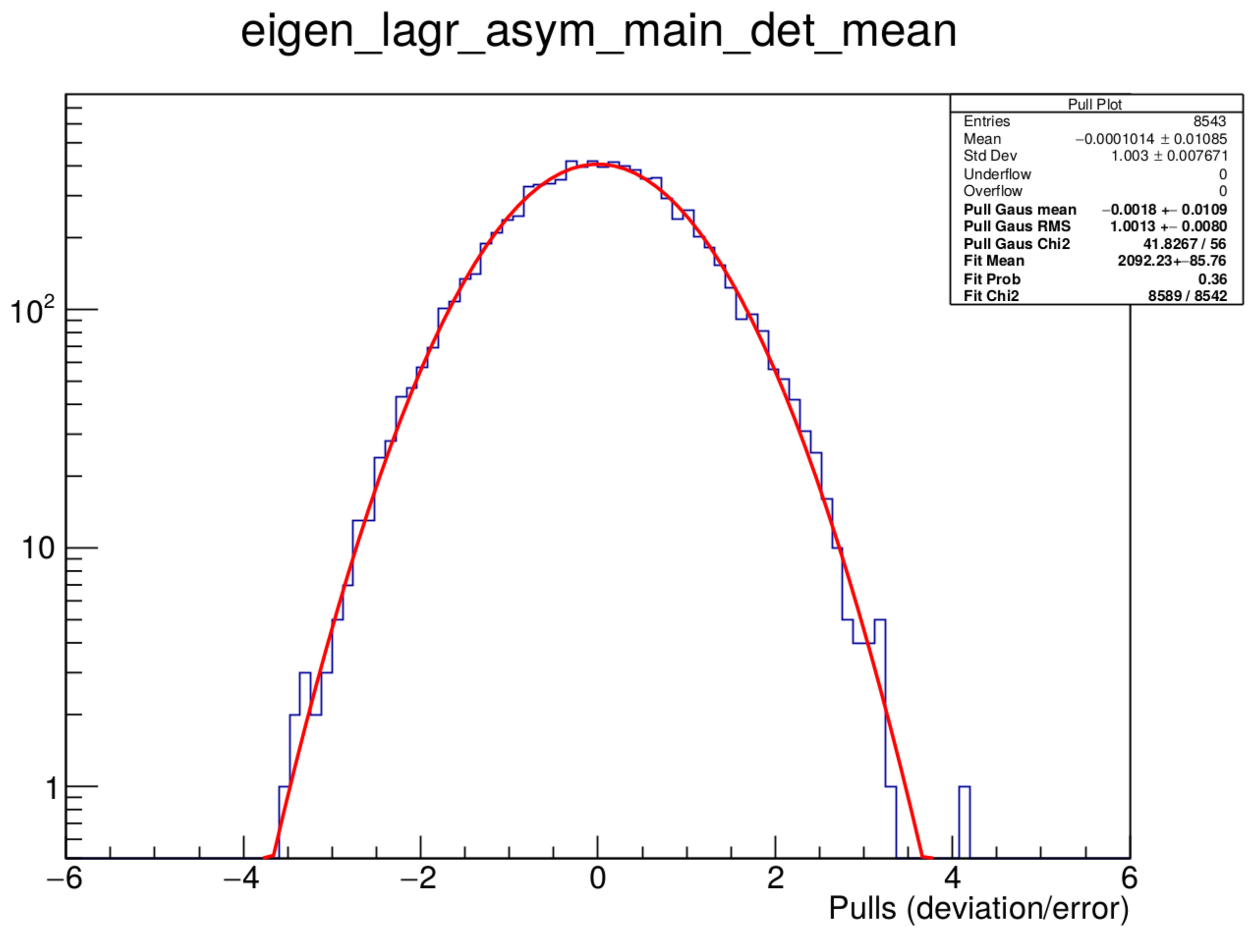
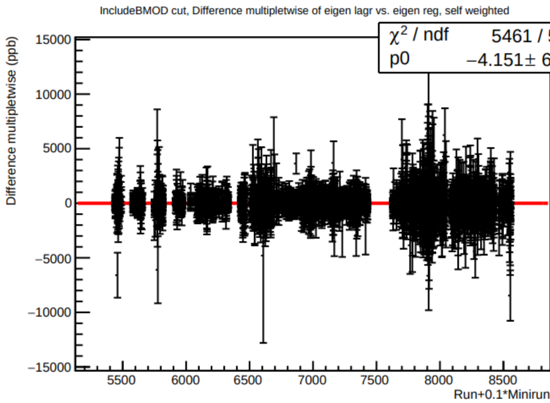


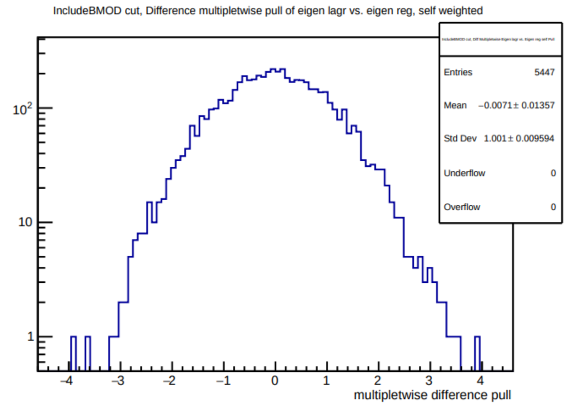
Fig 7. Minirunwise pull plot of main detector lagrange corrected asymmetry distribution.

But, what about the disagreement between regression vs. lagrange corrected asymmetry distributions minirunwise, and the disagreement between ErrorFlag and OnlyBMOD or IncludeBMOD cuts.

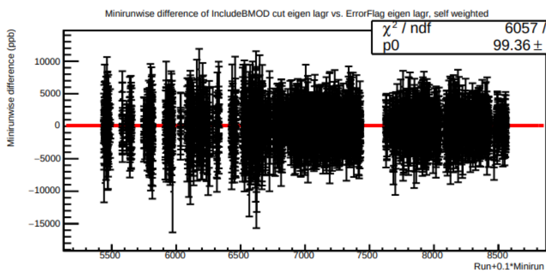
For the IncludeBMOD vs. ErrorFlag case:

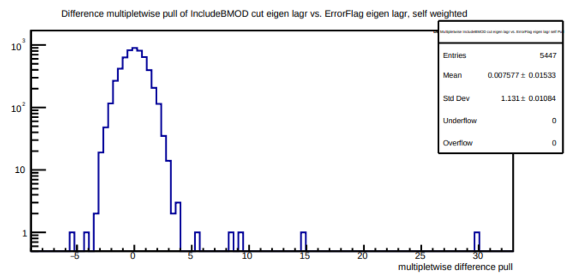
Lagr vs. Reg comparison:





Cut comparison:





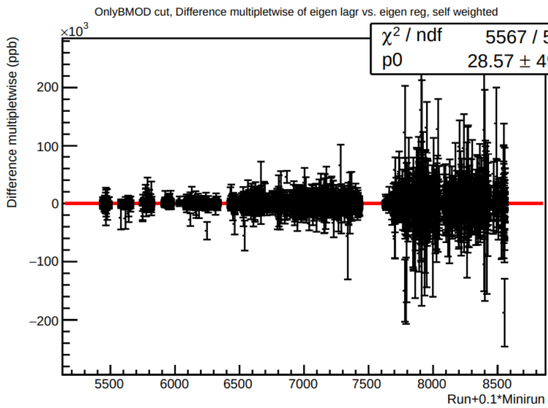
Difference in main det mean value PULL (quadrature error difference) between BMOD data set and standard ErrorFlag dataset lagrange 12bpm data, pull > 4

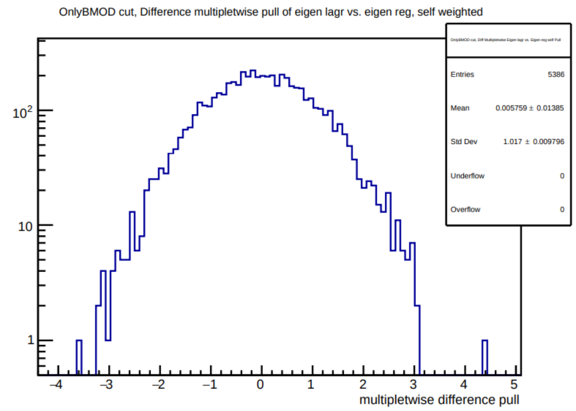
Row	rcdb_slug	rcdb_flip	rcdb_ihwp	rcdb_sign	run_numbe	minirun_n	nentries	rcdb_sign*pull
3624	159	2	2	1	7014	1	9277	-5.608746
6195	195	1	1	1	7835	1	9551	9.1458024
6401	196	1	2	-1	7896	0	9010	8.3561752
7684	210	1	2	-1	8191	6	9004	14.61033
7786	212	1	1	1	8214	1	9005	29.708971
8723	221	1	1	1	8536	2	9009	5.4275504
8793	223	1	2	-1	8558	2	9095	-4.083957

==> 7 entries (and note, a normal ErrorFlag minirun is 9000 nentries).

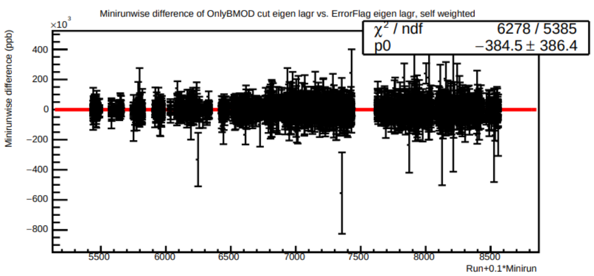
And for the OnlyBMOD vs. ErrorFlag case:

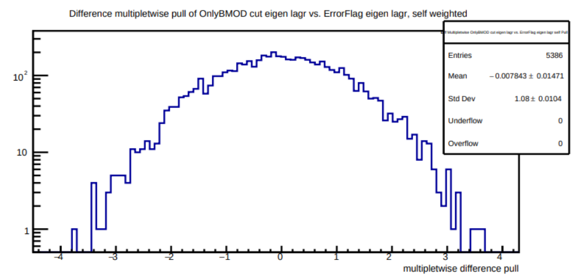
Lagr vs. Reg comparison:





Cut comparison:





==> No noteworthy outliers

Net disagreements between dataset cuts and ratios of corrected asyms' RMSs, and lagr vs. reg

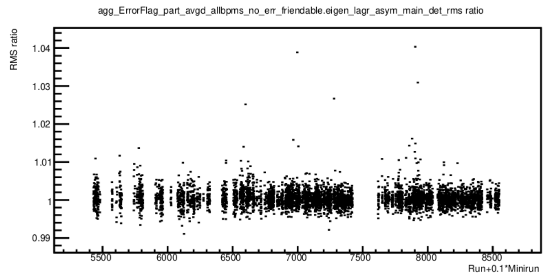
Net difference between regression and lagrange:

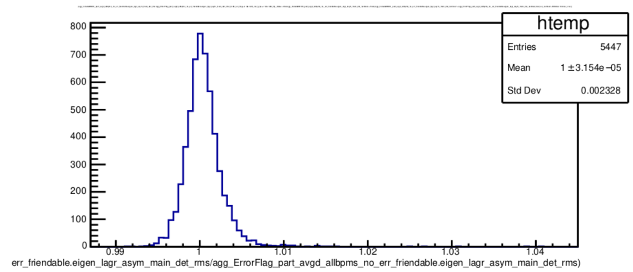
	Global,	Part 1,	Part 2,	Part 3
ErrorFlag cut:	??? +- ???	, 34.8 +- 25.6,	-8.9 +- 13.1,	-3.9 +- 13.9
IncludeBMOD cut:	-0.87 +- 11.0 ppb	, 43.6 +- 25.1,	0.15 +- 11.51,	-19.6 +- 22.4
OnlyBMOD cut:	-95.77 +- 121.2 ppb,	101.8 +- 119 ,	17.49 +- 130.6,	-493.3 +- 374.2

Net difference between lagrange of two different cuts:

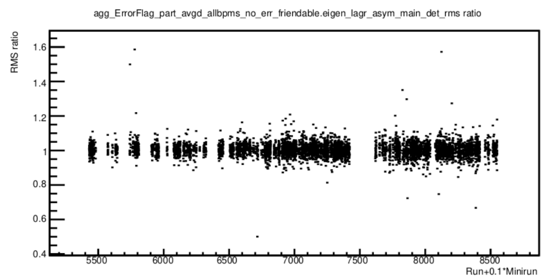
	Global,	Part 1,	Part 2,	Part 3
IncludeBMOD vs. ErrorFlag:	-17.65 +- 26.49 ppb,	-33.02 +- 81.1,	-5.1 +- 39.7 ,	-26.3 +- 37.9
OnlyBMOD vs. ErrorFlag:	-143.5 +- 540.9 ppb,	-430.6 +- 1119,	-7.5 +-781.8,	-190.2 +- 947.2

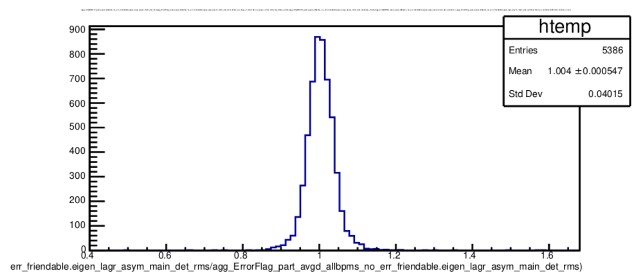
IncludeBMOD ratio of lagrange corrected asym distribution RMS / ErrorFlag version:





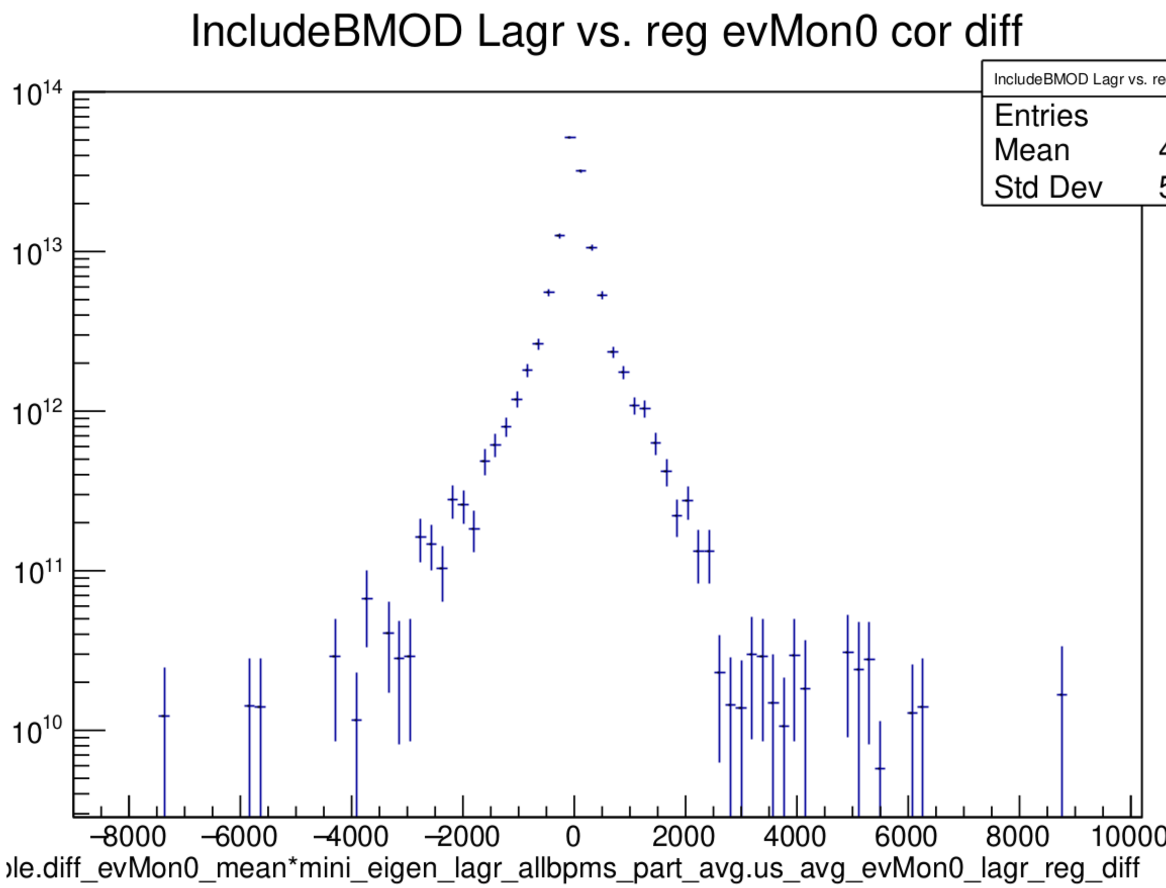
OnlyBMOD ratio of lagrange corrected asym distribution RMS / ErrorFlag version:



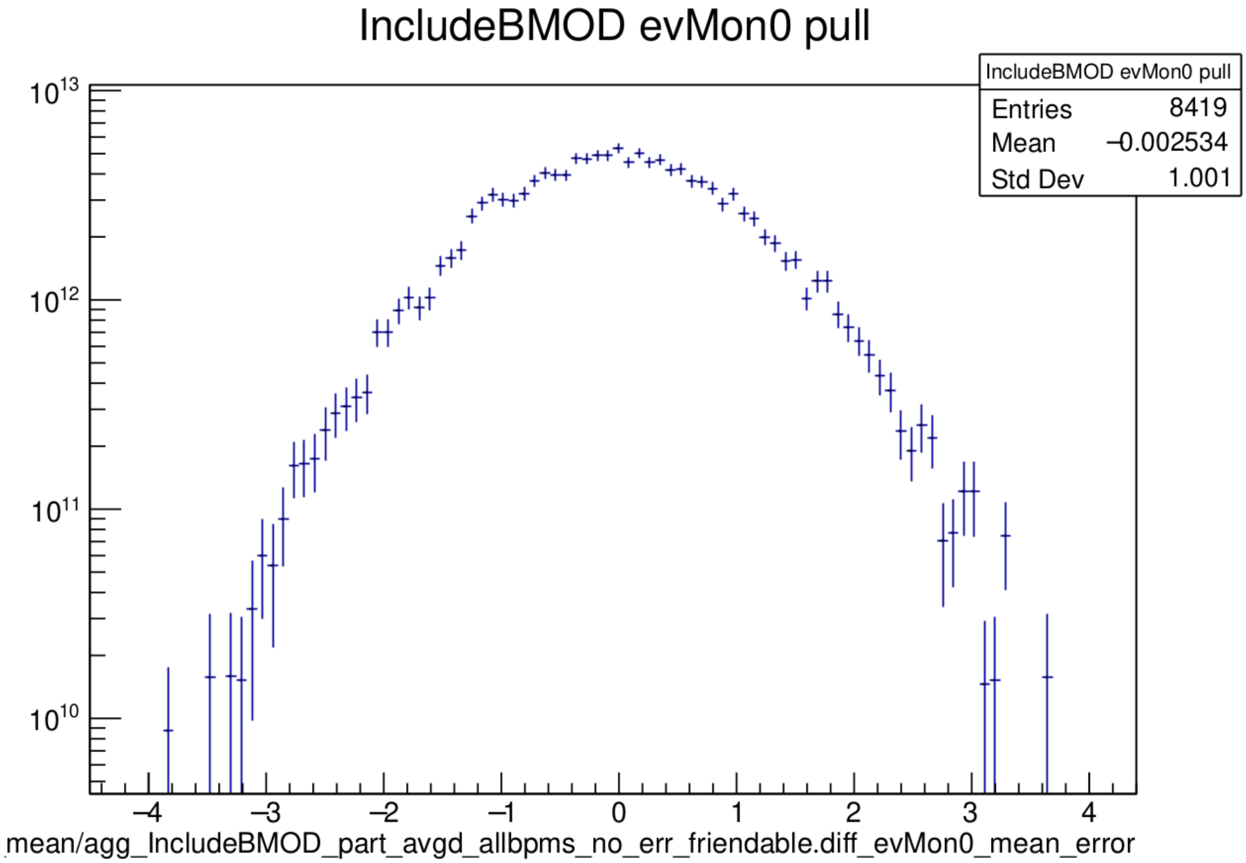


Minirunwise evMon diffs and corrections disagreements between dataset cuts, and lagr vs. reg

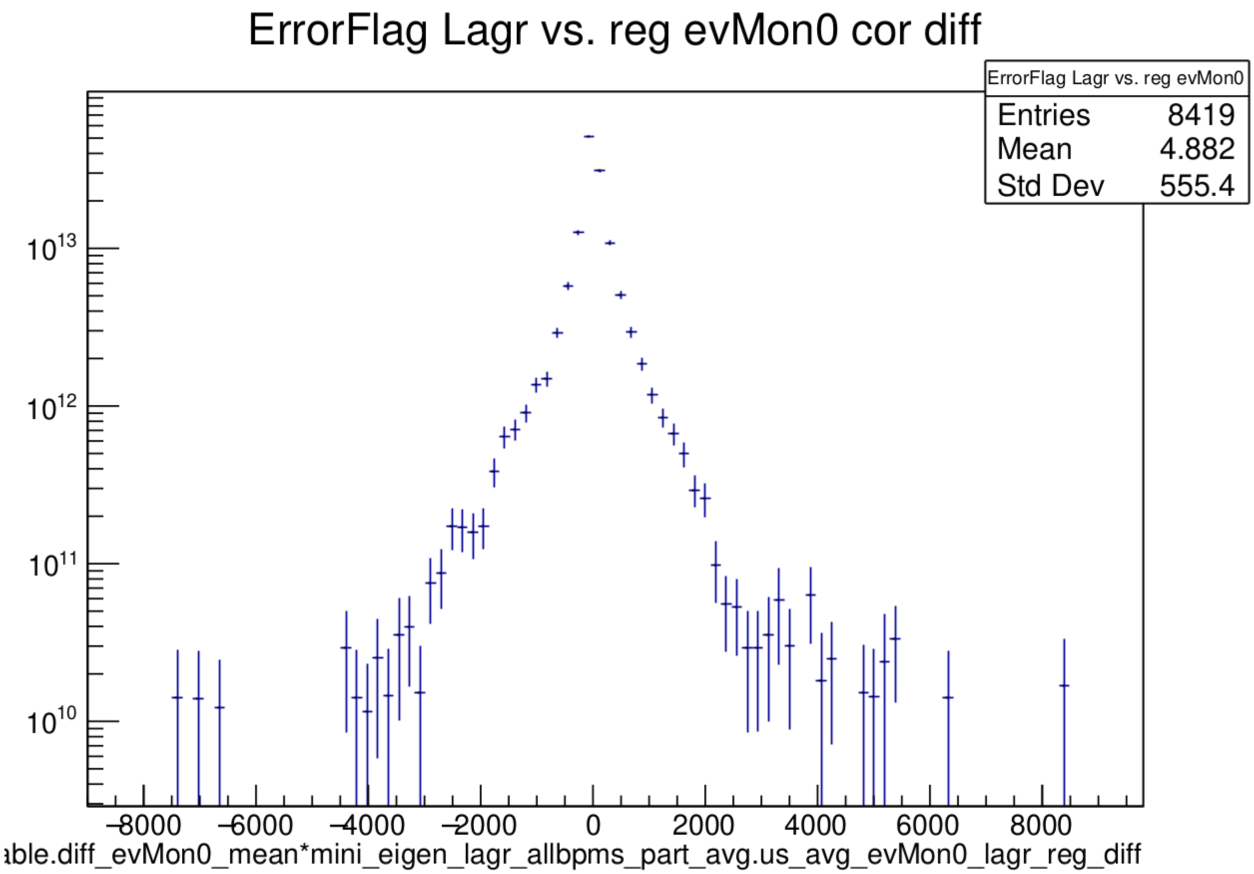
Looking at the difference in correction per monitor, minirunwise, the corrections all seem to be net small, and the monitors themselves are consistent between cuts. For example, looking at IncludeBMOD evMon 0 correction and diff pull plot:



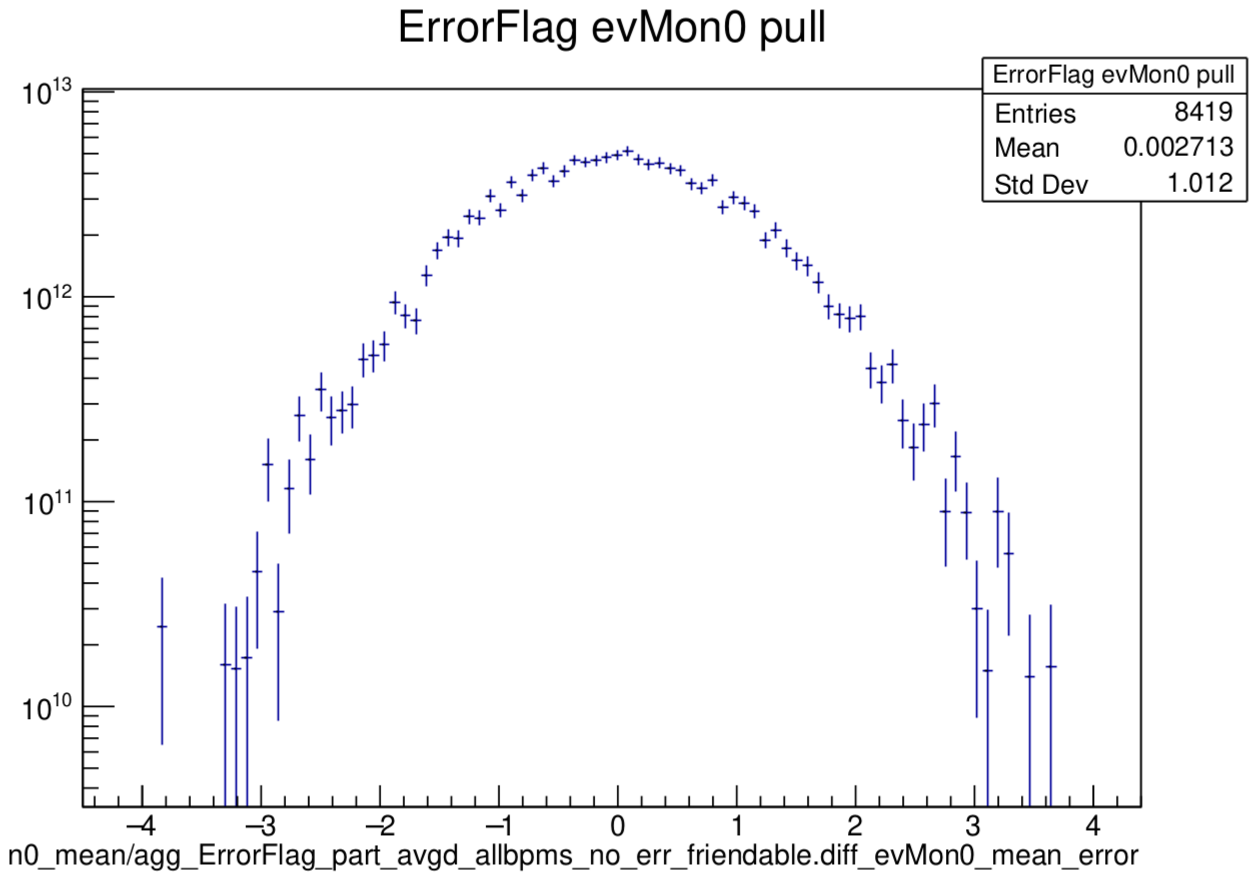
evMon 0 IncludeBMOD correction value difference between lagr and reg



IncludeBMOD evMon 0 diff pull

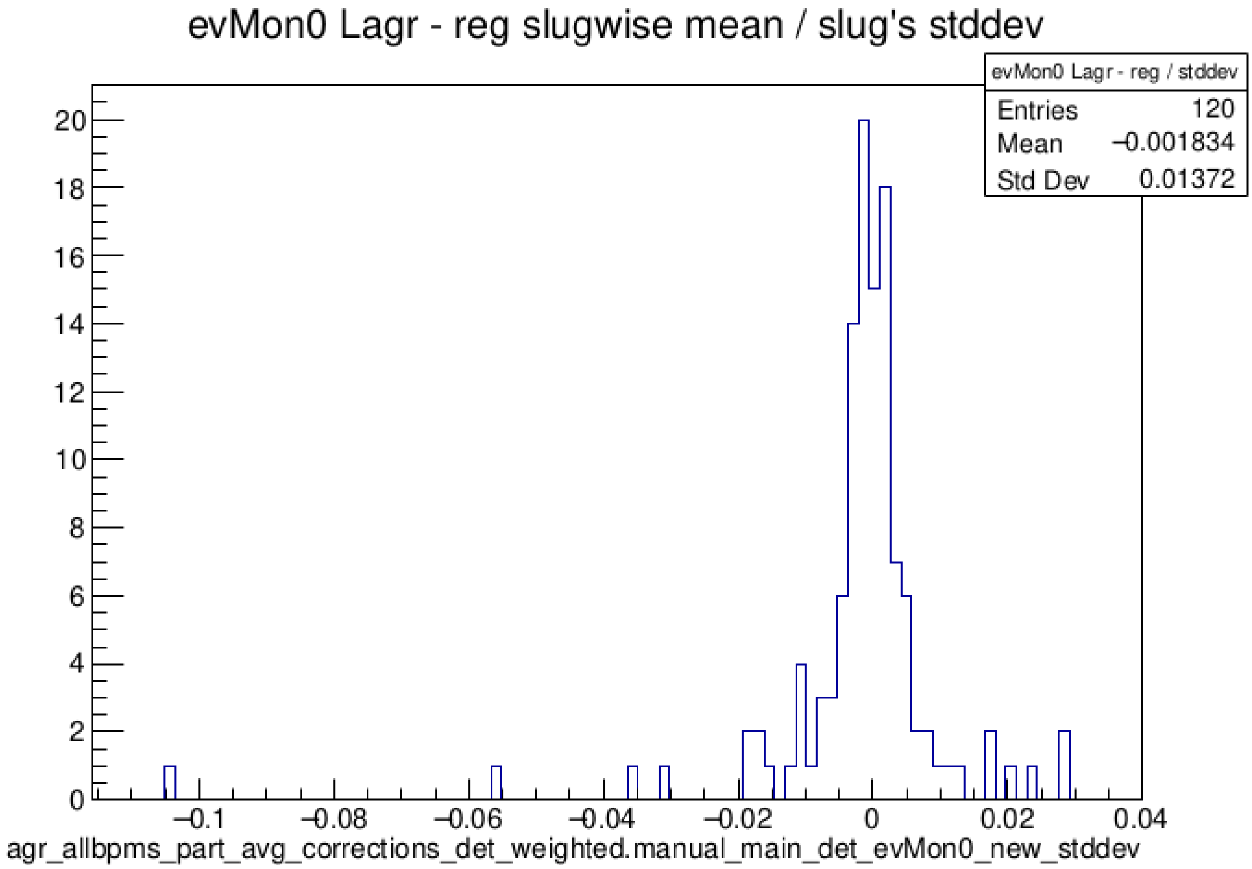


The same but in the ErrorFlag cut

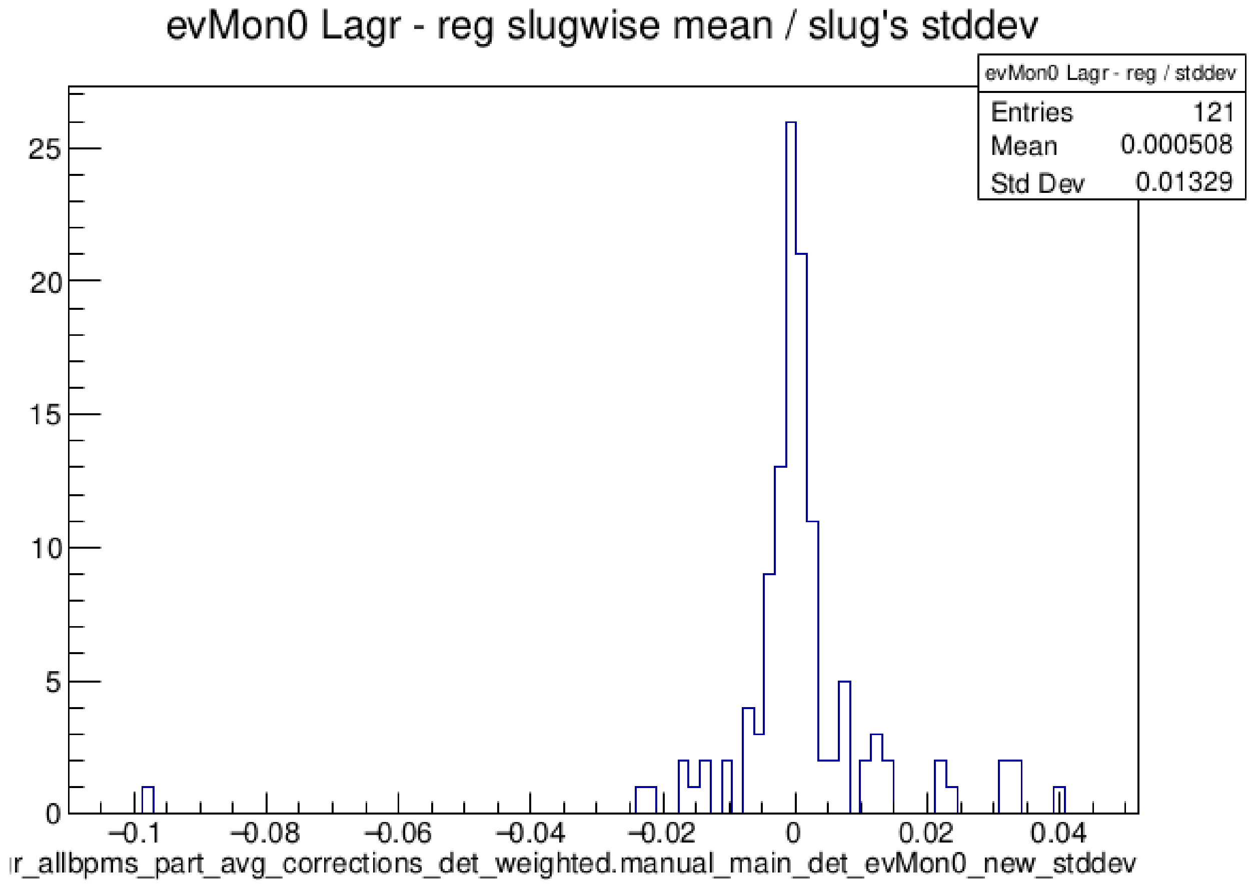


ErrorFlag evMon 0 diff pull

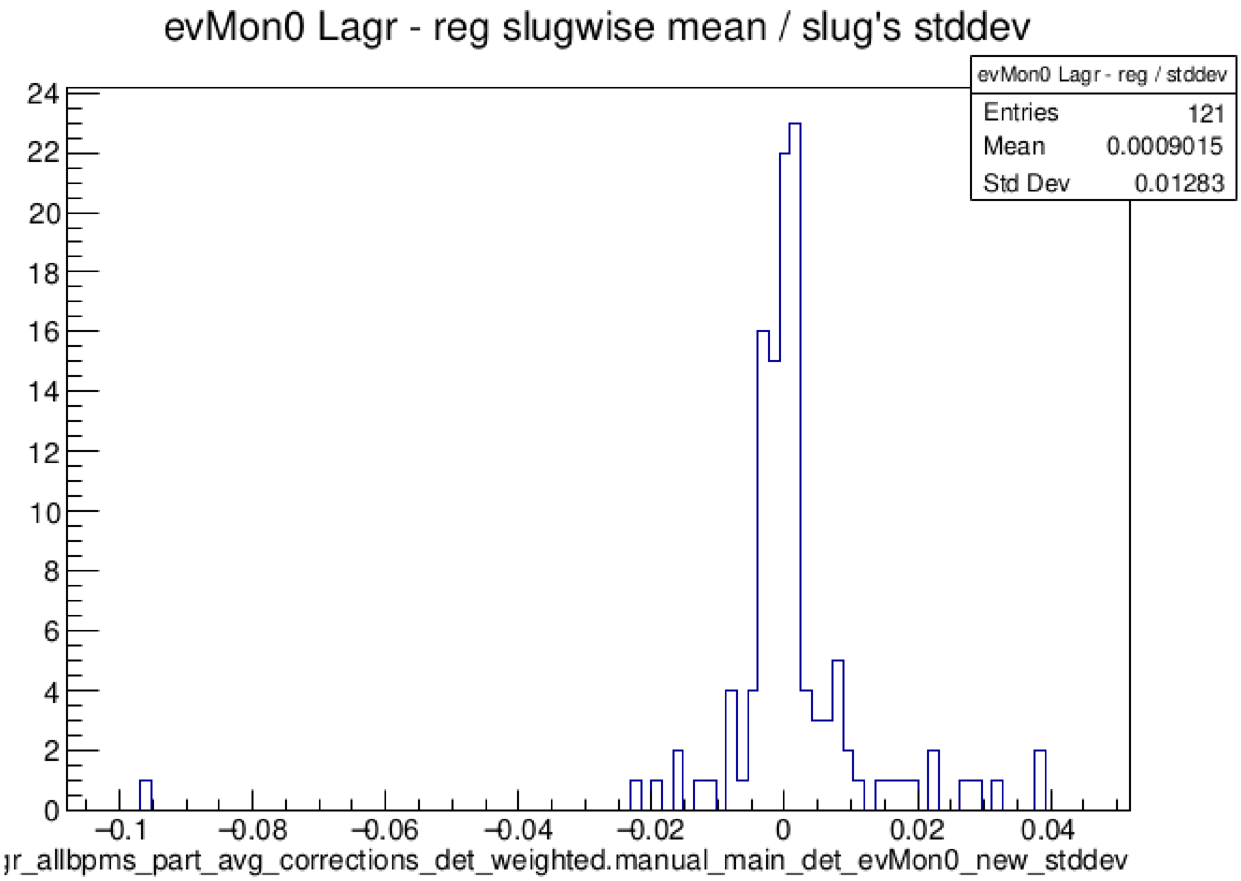
And, we can also look at the slugwise mean value of the correction vs. the slug's standard deviation width of minirun's means (the breadth of the distribution of miniruns' corrections). This shows us that the mean across a slug is happily situated within the standard deviation of it's points, meaning that there are no strong outlier points outside of their own spread.



evMon 0 spread of slug’s means/(stddev of minirun’s means per slug) for OnlyBMOD cut



evMon 0 spread of slug’s means/(stddev of minirun’s means per slug) for IncludeBMOD cut

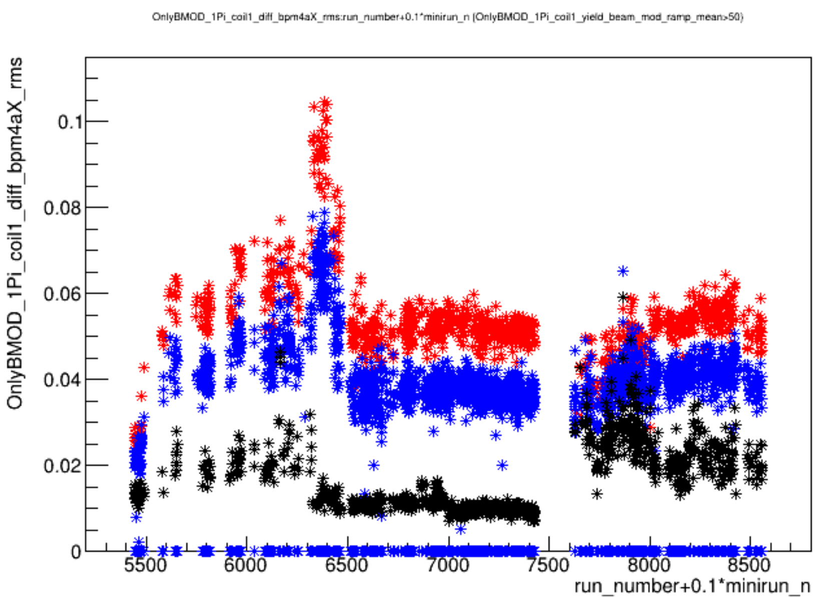


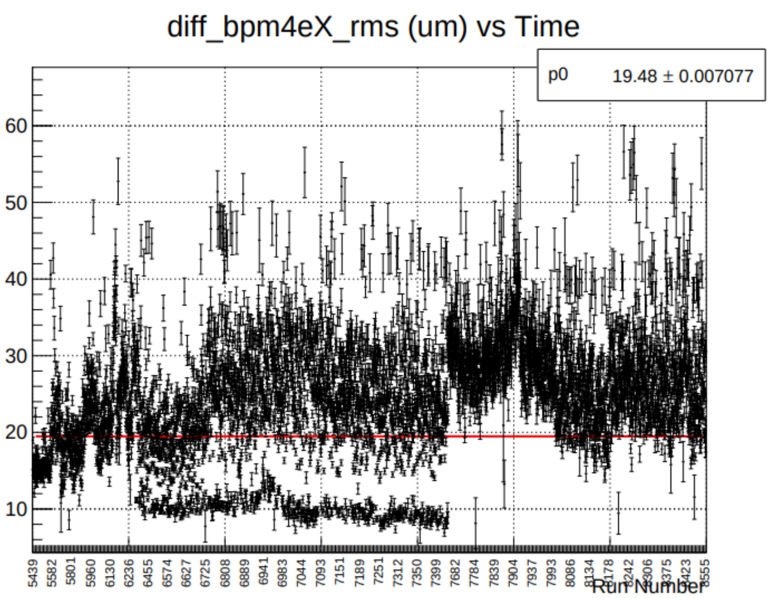
evMon 0 spread of slug's means/(stddev of minirun's means per slug) for ErrorFlag cut

Multiplet out of phase with BMOD cycles, increased BPM RMSs and Regression Corrected RMSs

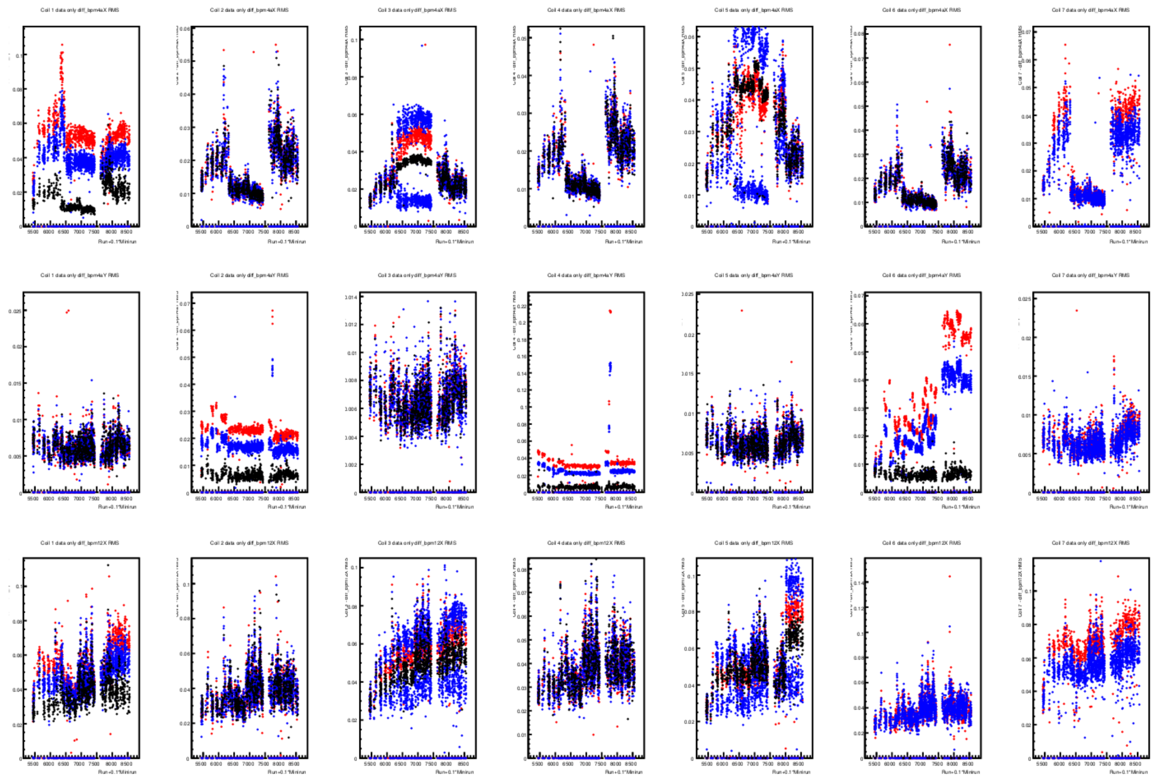
I can say with a high degree of certainty that the jumping around RMSs during BMOD only running are caused by the relative phase of the BMOD ramp 15 Hz signal w.r.t. the phase of the 30Hz helicity difference calculation -> when the helicity differences measure the 15 Hz signal with cancellation the RMS is very small, when they sample it with minimal cancellation the RMS is very large, and this is determined by the average phase over which the helicity correlated difference is calculated:

Black = half pi phase shift. Blue = quarter pi phase shift, and Red = 0pi phase shift of helicity quartet integration vs. 15 Hz BMOD ramp: this plot is for BPM 4aX during coil 1 modulation only, separated separately into the 3 relative phase configurations that our quartets could see

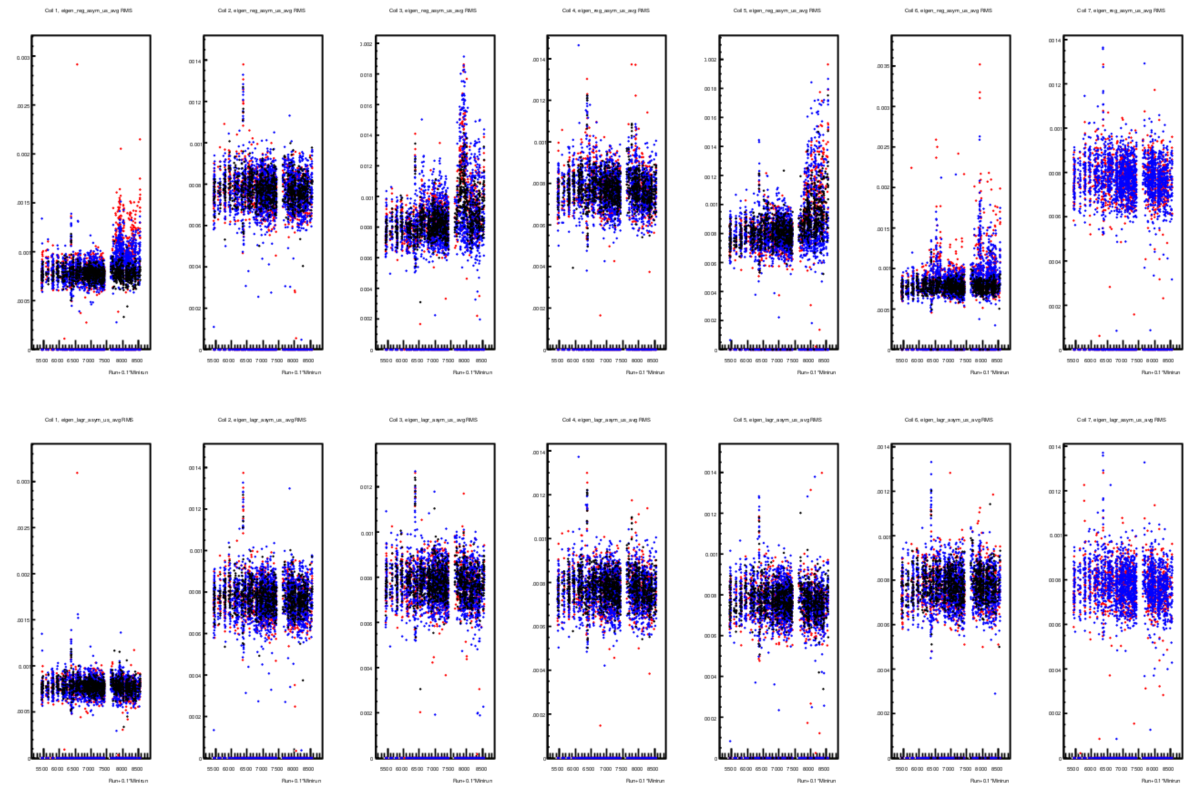




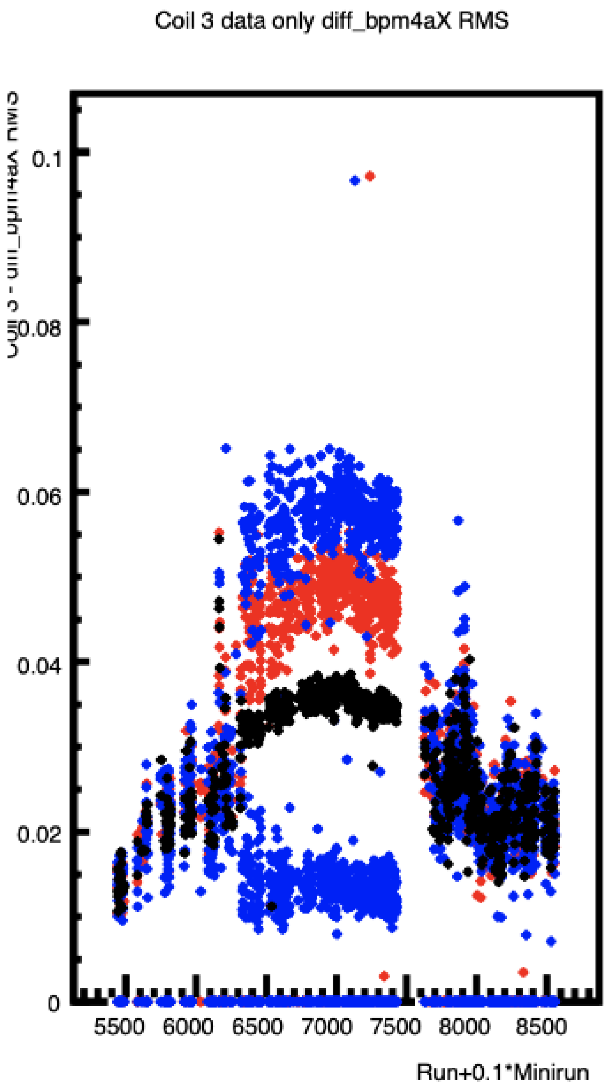
Which is at least part of the story for explaining this plot ^^



And here is the same thing but for reg and lagr corrected asymmetries (allbpm, part avg eigenvectors) analyses



What is up with the blue phase data here?



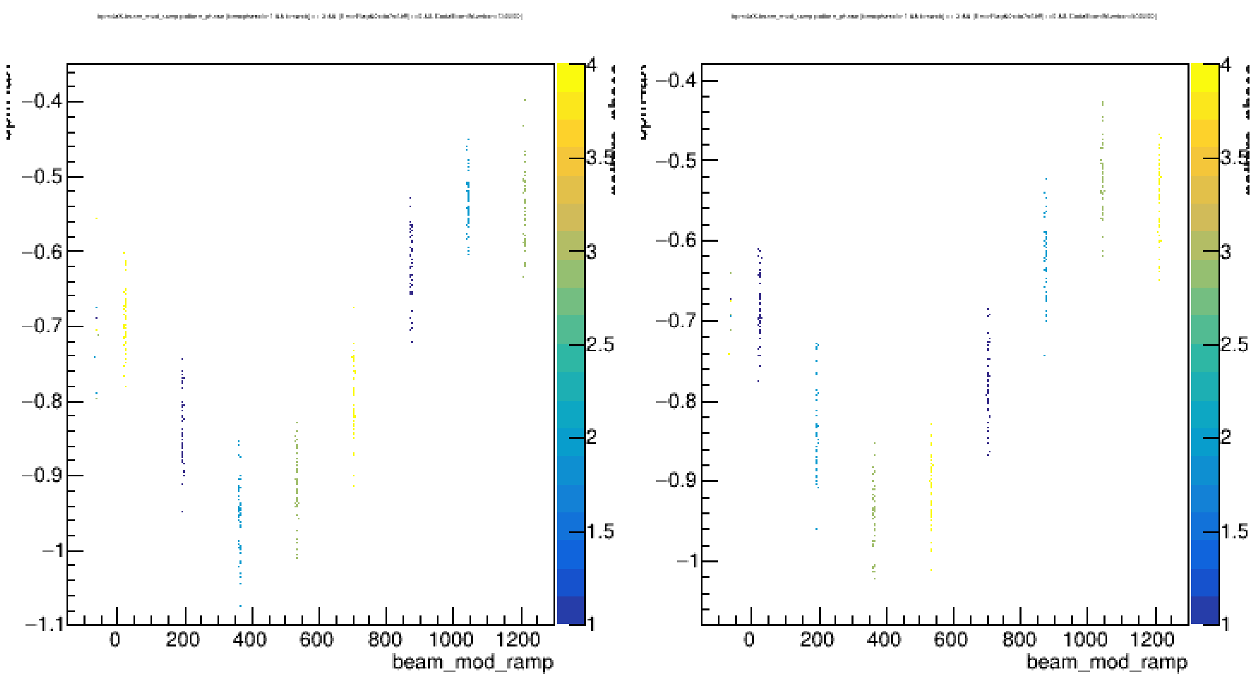
My guess is that the phase slipped a bit for coil3 relative to the others (and to the signal being sent) during that part of the experiment. How or why is another question - pick a blue one from the bottom caterpillar in segment 2, and a blue from the top caterpillar, and show that the phase of the motion is different relative to ramp.

I think it is true. Here is an example, run 6966 has BMOD in a few of its miniruns. Miniruns 0 and 3 have BMOD cycles with the large RMS side of that blue caterpillar plot for 4aX coil3 data, and minirun 5 has the small RMS side of it:

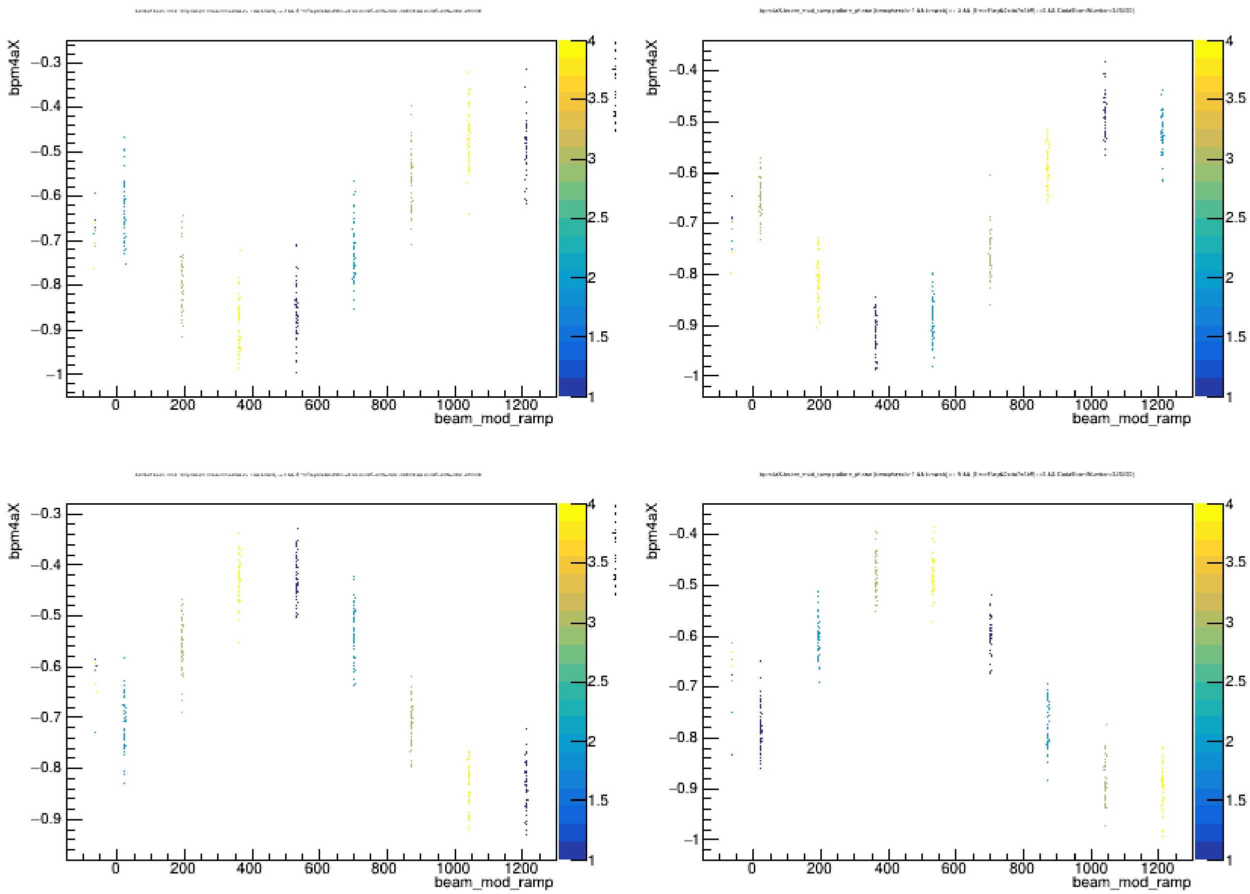
Here is what I get when I plot the BPM 4aX eventwise yield vs. BMOD ramp phase (the one measured in the ADC, looks the same with the DAQ-software number as well) - also sorry, I can't figure out how to do custom binning of a COLZ TH2:

The color axis is the Pattern phase (1, 2, 3, 4, representing which event in the Quartet we are on) and the X axis is BMOD ramp. Y axis is BPM response.

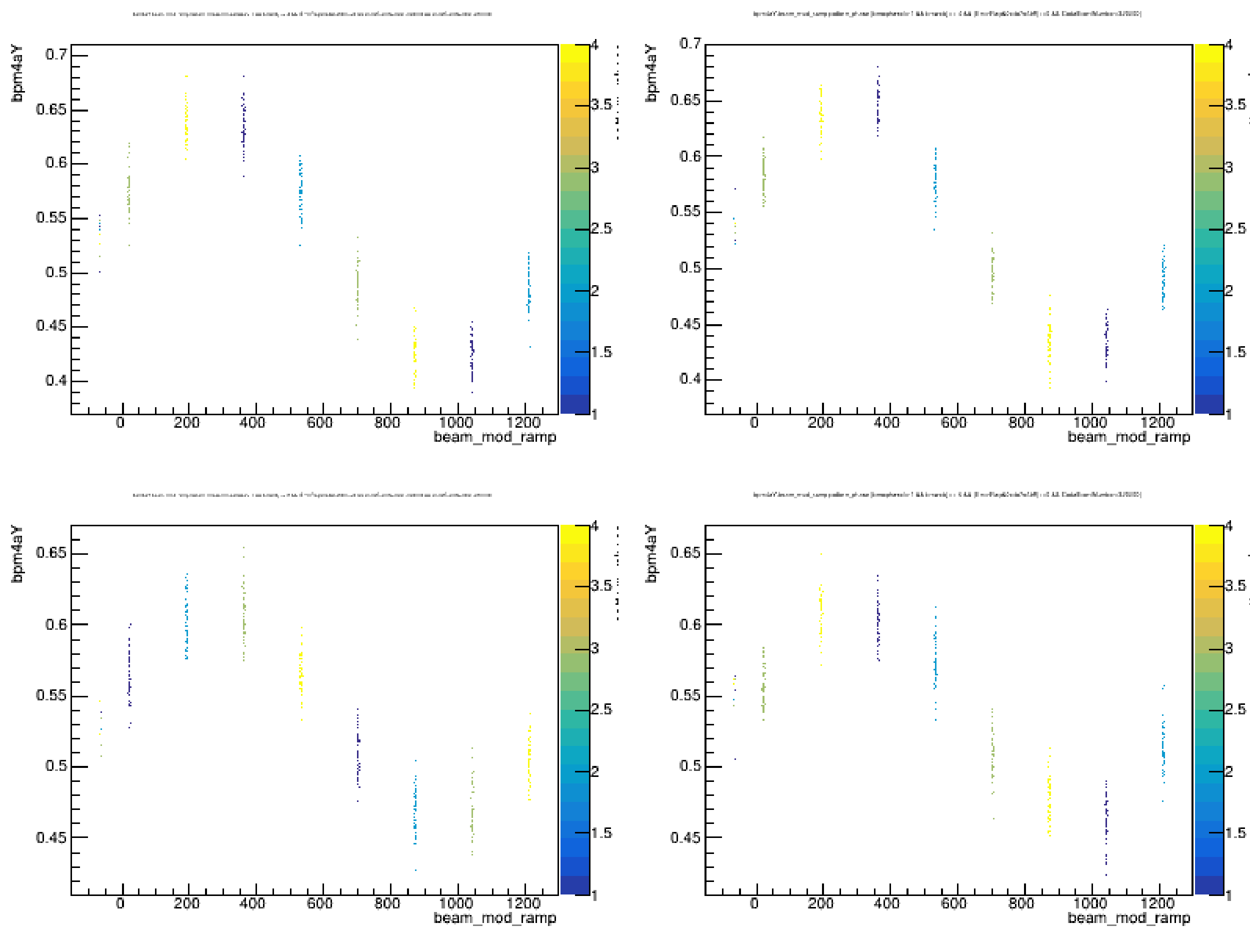
Run 6966 X coils, two supercycles in the same run:



Run 6961, looking at X and Y, separate supercycles in the same run. The left plot is the first 2 minirun's BMOD coil3 data, the right plot is the minirun 5 data:



And 6966 same cycles Y Coils:



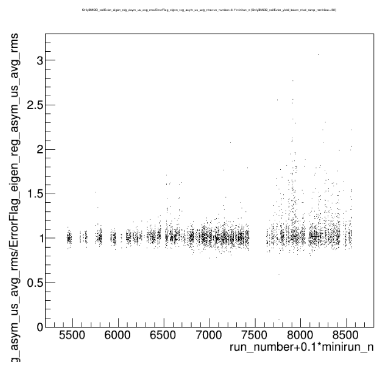
So there is phase slippage between minirun 3 and 5
But it is stable, hence the sudden jump from large to small BPM diff RMS during coil3 active

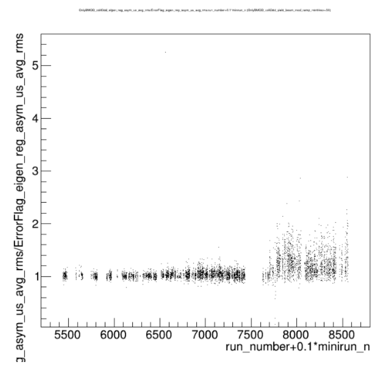
There is probably a better way to plot that, but I think this answers the question

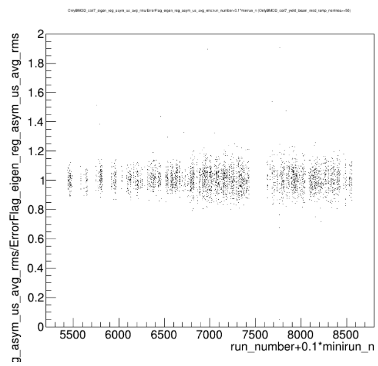
------

And note: regression does have issues with BMOD included data for X and Y coil modulation, but lagrange is fine, and this is expected due to the regression dataset including no information during the intentionally modulating period:

Reg: Y coils X coils E coils







Lagr: Y coils X coils E coils