ProtoDUNE analysis:
Pulser study of new sticky code interpolation

ProtoDUNE sim/reco

David Adams

BNL

December 12, 2018

Pulser signals

Cold electronics provide the option to inject pulses

- At regular intervals, injected voltage is raised and then lowered
  - Controlled with 6-bit DAC ➔ 64 levels
  - Fairly linear with step size V_S = 18.75 mV
- Capacitively coupled to preamp input
  - C = 183 fF
- Voltage change corresponds to fast charge collection
  - Step size is Q_S = (183 fF) × (18.75 mV) = 3.43 fC = 21.4 ke
  - This is about 1 MIP
    - Prefer something smaller?

Signal may be generated in either of two places

- Internal: signal generated in the preamp ASIC
  - Input voltage has significant offset that varies from ASIC to ASIC
  - Variations are a large fraction of the step size
- External: signal generated on the FEMB
  - Little or no offset: for DAC setting S, $Q = S \times Q_S$

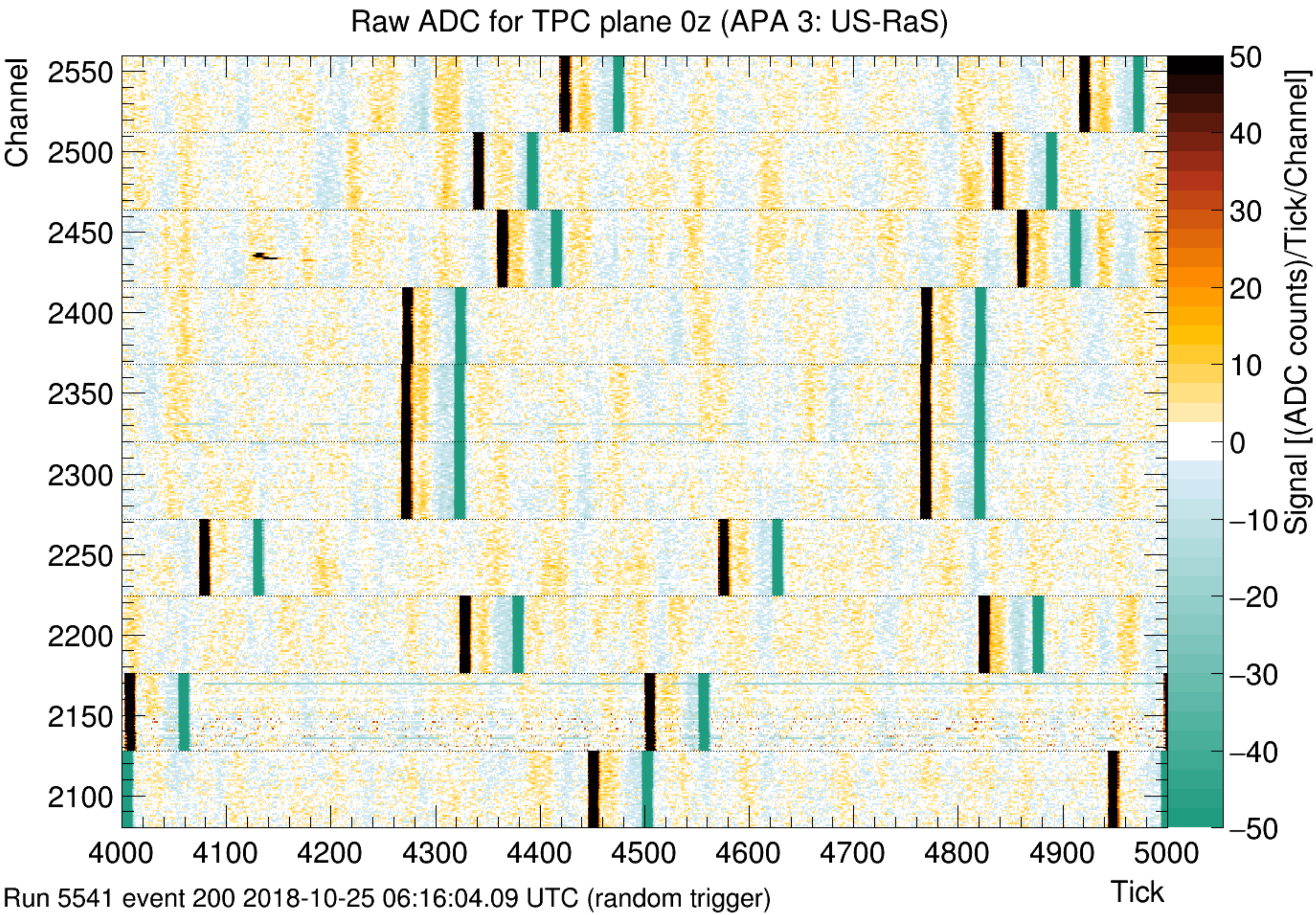
ProtoDUNE data

Some pulser amplitude scans have been taken with protoDUNE

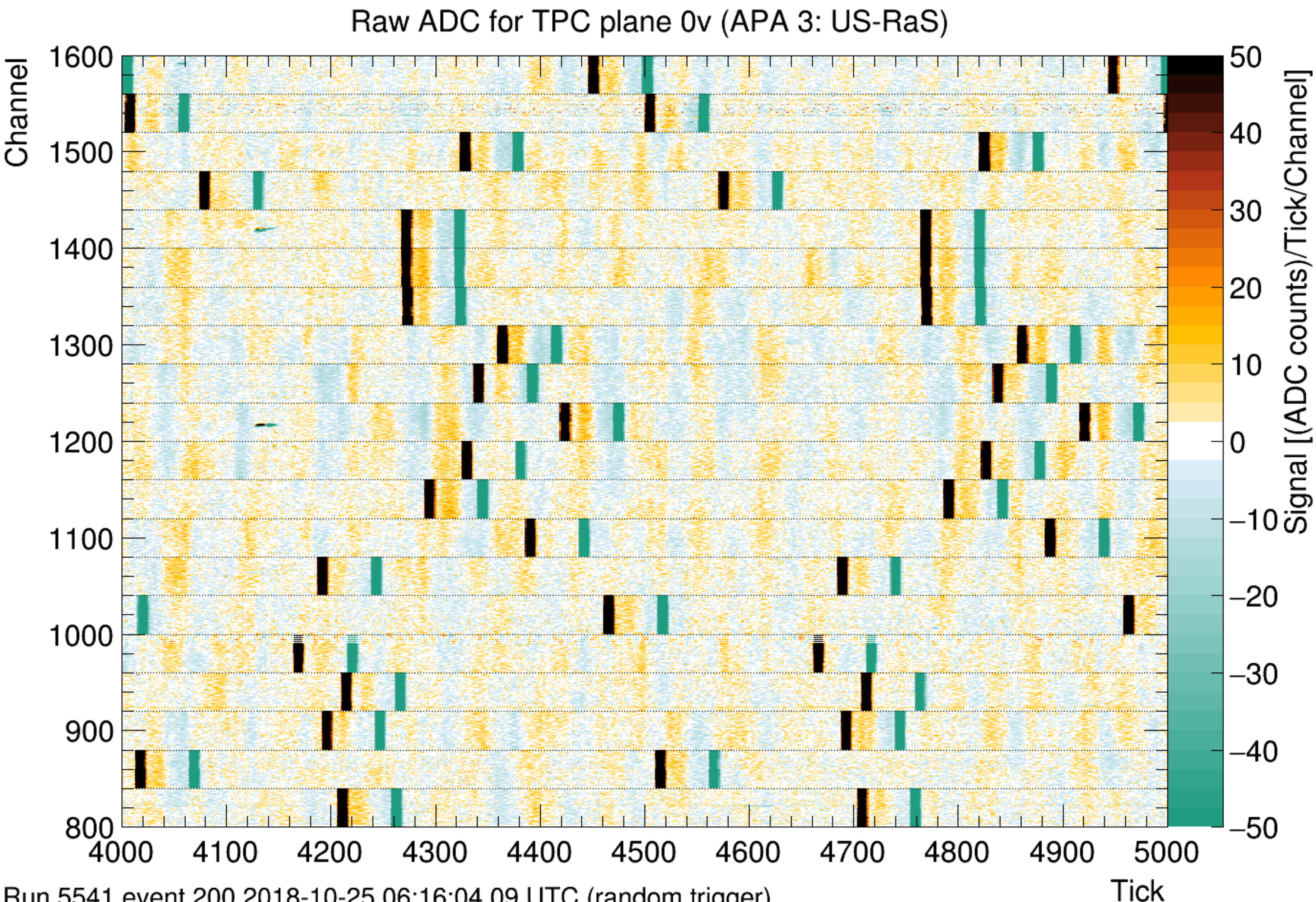
- Amplitude scan means the voltage DAC is varied
  - New run started for each DAC setting (0,1 , 2, …)
- All taken with internal pulser
- Aug 9-21
  - Detector filling
- Sep 21
  - Before baseline shift to fix “ledge effect”
- Oct 25
  - Cathode HV off, anode and grid on
  - After baseline shift (Oct 11)

I have requested more data ASAP

- Amplitude scan with all HV off and all HV on
  - Both internal and external pulser (4 scans total)
- Later multiple scans with varying timing offset



Raw ADC for TPC plane 0z (APA 3: US-RaS)
Channel
2550
2500
2450
2400
2350
2300
2250
2200
2150
2100
4000
4100
4200
4300
4400
4500
4600
4700
4800
4900
5000
Tick
50
40
30
20
10
0
−10
−20
−30
−40
−50
Signal [(ADC counts)/Tick/Channel]
Run 5541 event 200 2018-10-25 06:16:04.09 UTC (random trigger)



Raw ADC for TPC plane 0v (APA 3: US-RaS)
Channel
1600
1500
1400
1300
1200
1100
1000
900
800
4000
4100
4200
4300
4400
4500
4600
4700
4800
4900
5000
Tick
Signal [(ADC counts)/Tick/Channel]
50
40
30
20
10
0
−10
−20
−30
−40
−50
Run 5541 event 200 2018-10-25 06:16:04.09 UTC (random trigger)

Pulser studies

The pulser data can be used for many important studies

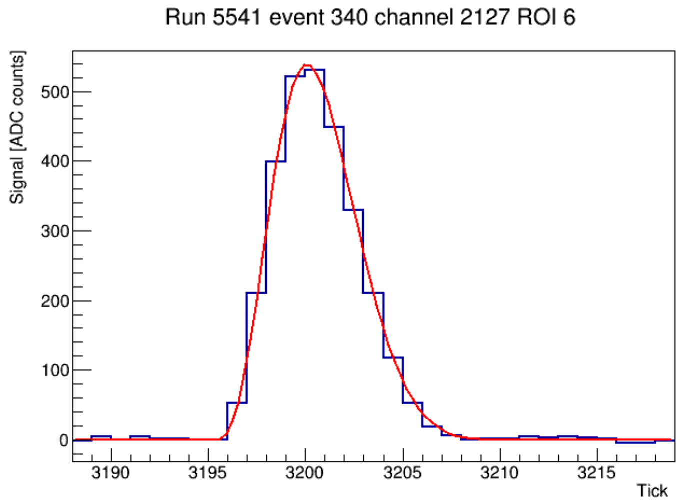
- Resolution
  - For a given amplitude setting, measure how the height and area of the processed signal vary
    - RMS, tails, etc.
    - Area measurement directly applicable to collection signals from tracks
  - At different stages and with different processing options
    - Pedestal evaluation, sticky code mitigation, tail correction, deconvolution
- Gain calibration
  - Processed signal height or area vs pulser amplitude provides gains
  - Also find the range of (roughly) linear response for each channel
    - ADC and preamp may restrict this range
- Gain nonlinearities
  - Expect signal to be the same for each tick along the pulse
  - Use tickmod = tick%497 to get a finer mapping of gain vs input charge
- Sticky codes
  - Use tickmods to search for sticky codes as was done for pedestal

Today

Tools

There are some tools useful for studying pulser data

- AdcRoiViewer
  - Waveform for each ROI (pulser signal and more)
    - Includes option to fit with CE waveform
  - Summary histograms of ROI properties
    - Area, fitted height, shaping time, …
  - Summary params (e.g. RMS) vs. channel
  - Under continuing development
    - Many changes over last week
- AdcTickModViewer
  - Histograms of ADC counts filled separately for each tickmod
  - Use mean for gain studies
  - Look at tails and sticky codes



Rough calibration

Rough height calibration

- ProtoDUNE data taken with gain 14 mV/fC
  - Pulse height for rapid charge injection
- Nominal shaping time is 2 µs—actual 4.35 ticks with variation
- ADC response is 3.0 ADC/mV with variation
  - Combining these, the rapid pulse gain is 42 ADC/fC = 6.7 ADC/ke
  - I.e. pulse height calibration is 150 e/ADC

Area calibration

- Detector signals do not have rapid charge injection
  - See example waveforms on following pages
- Best estimate of FE input charge is presumably the area in an ROI
- Rapid pulse area = 1.269 × Height × (Shaping time)
  - From CE response function checked with pulser data
- Nominal FE-ADC calibration is 27.2 e/(ADC-tick)
  - Channel variations from ADC response, preamp gain and shaping

Sticky code mitigation

Search for stick codes described previously

- I looked at waveforms for all channels
- Record (in fcl) codes that look sticky
  - Only in pedestal region

Mitigation is applied to those codes

- For release v07_12_00 and earlier, this was linear interpolation
  - Between nearest good samples below and above
- New in v07_13_00 (now improved), constant-curvature interpolation
  - I.e. quadratic function of tick
  - Still required to pass through two nearest samples
  - Choose curvature to come close to the second closest good sample below and above the mitigated sample
    - Could use 3rd order polynomial and cross both but too many wiggles
    - Last release, choose equally close to the 2nd nearest samples (call this ccv1)
      - » But sample in a signal can push the other side away from the baseline
    - Now weight the 2nd samples to favor one close to the 1st sample on its side

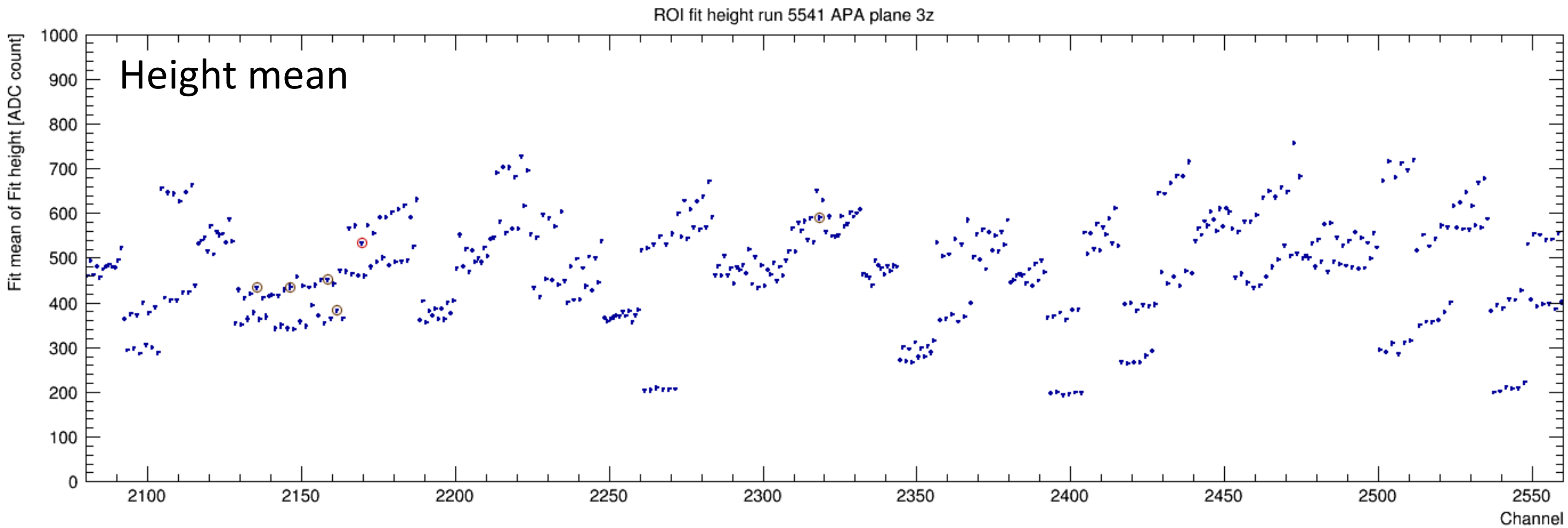
Analysis

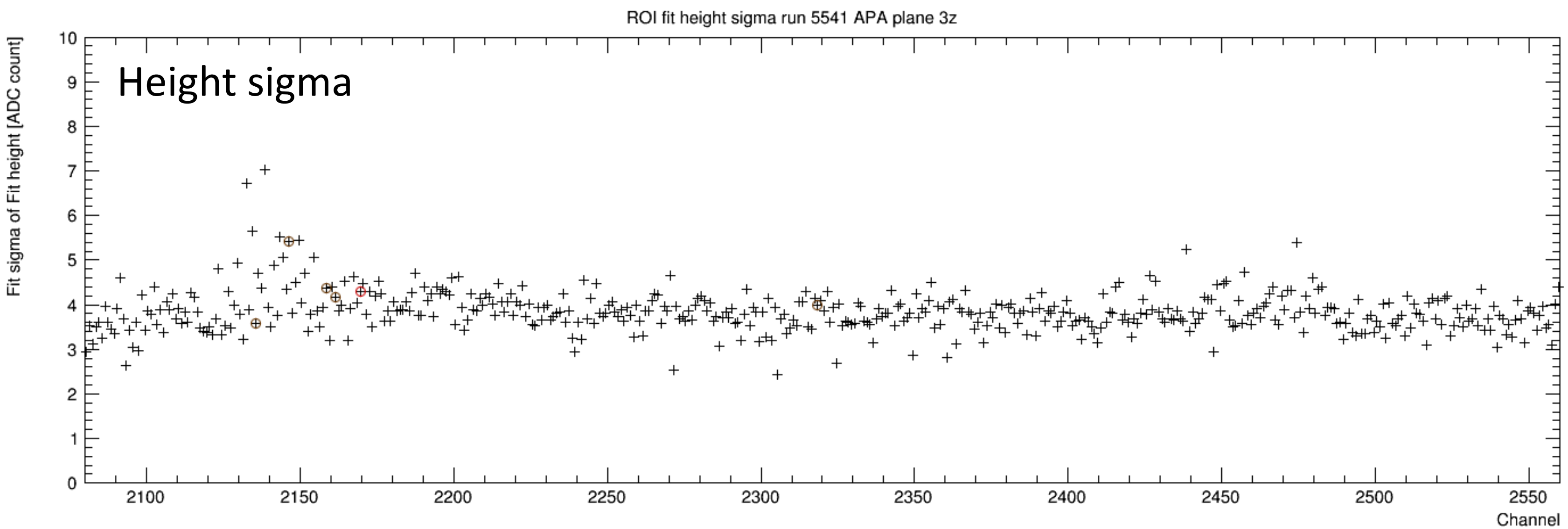
Use AdcRoiViewer to study resolution

- Without and with ADC mitigation
- Results shown for old (linear interpolation) and new mitigation

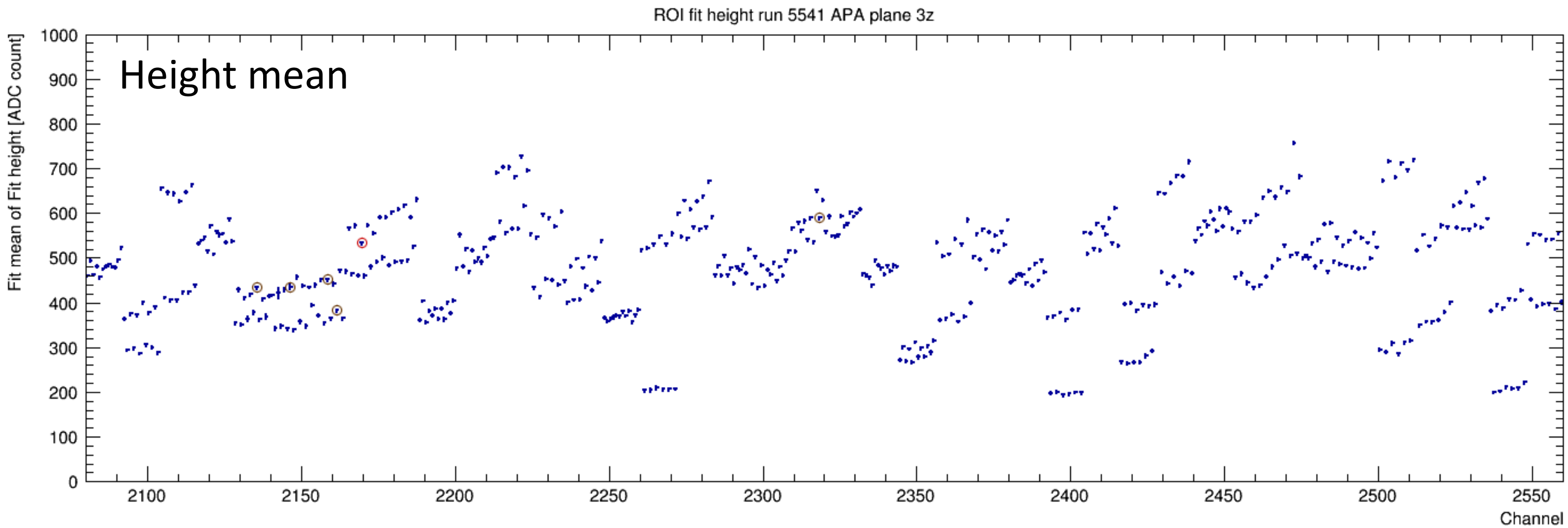
Channel summary plots for APA3z

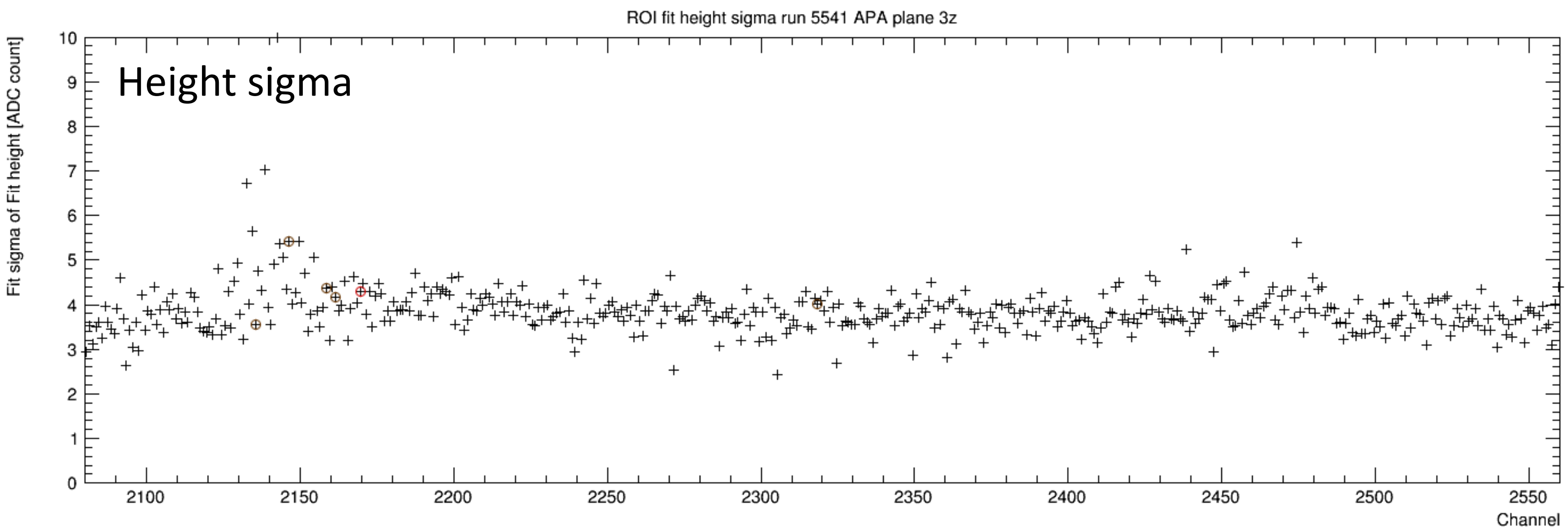
Fit height unmitigated



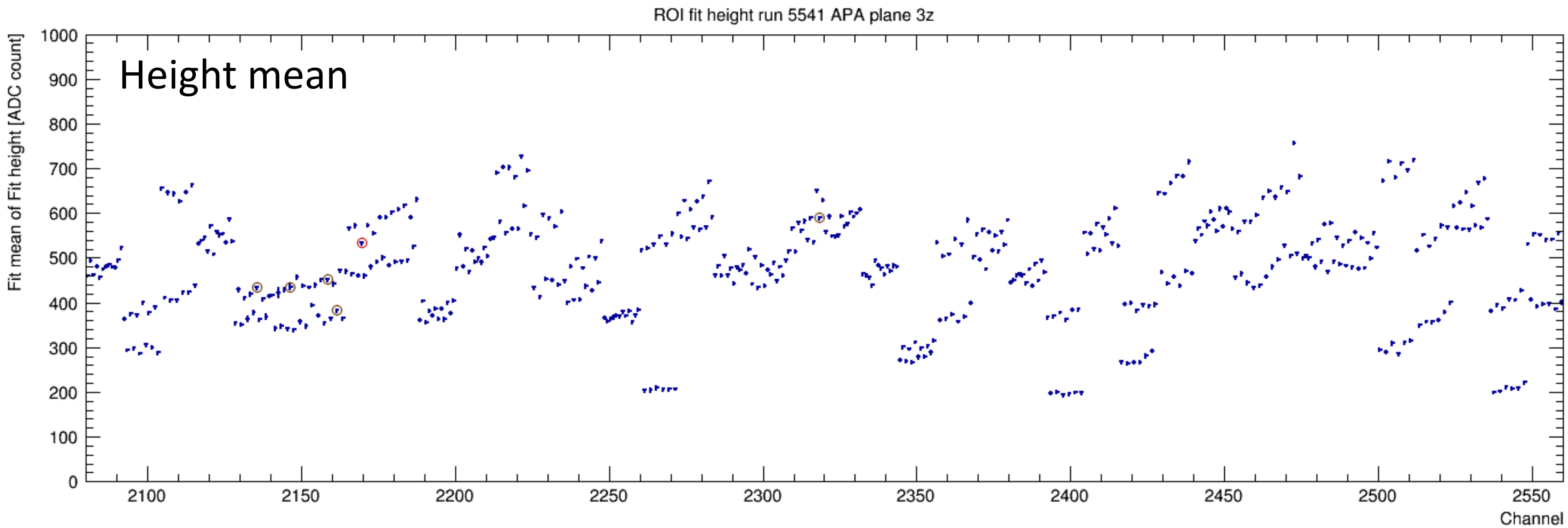


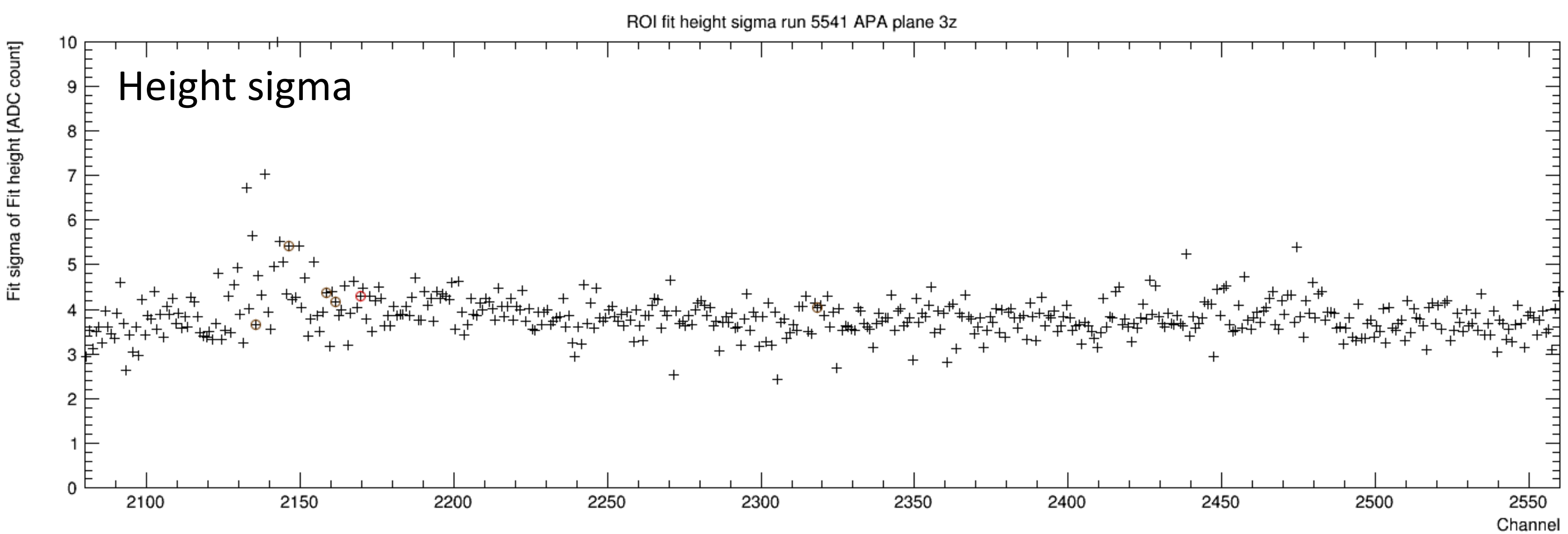
Fit height with new mitigation



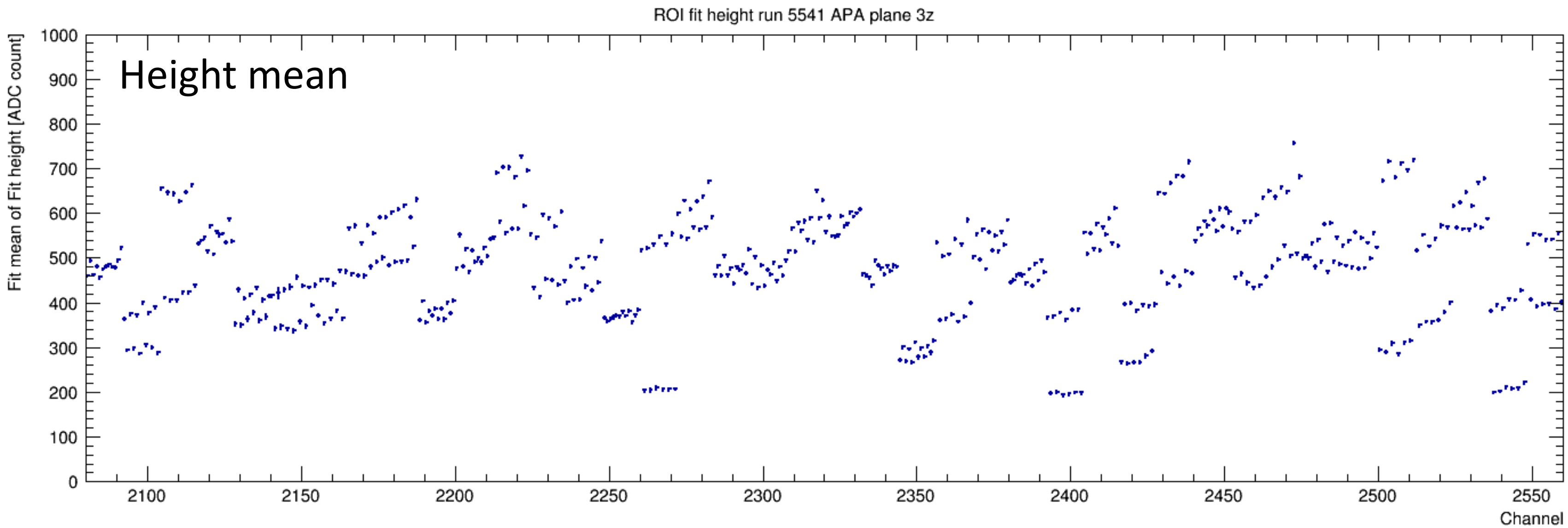


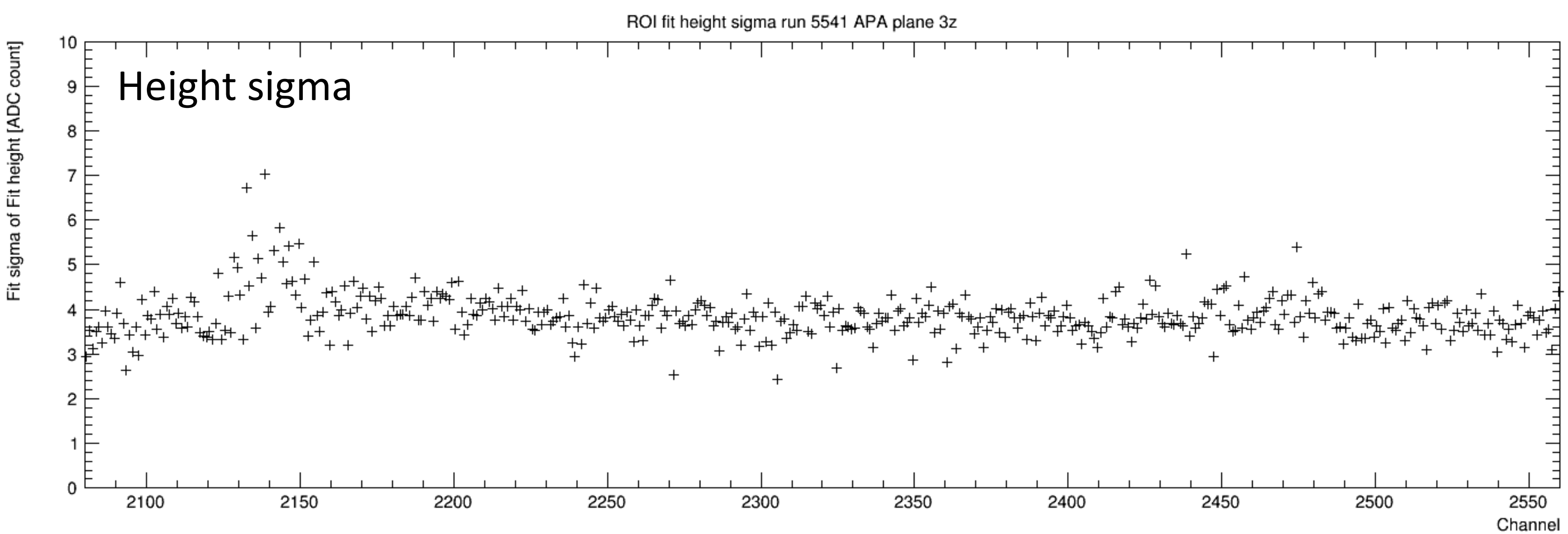
Fit height with ccv1 mitigation



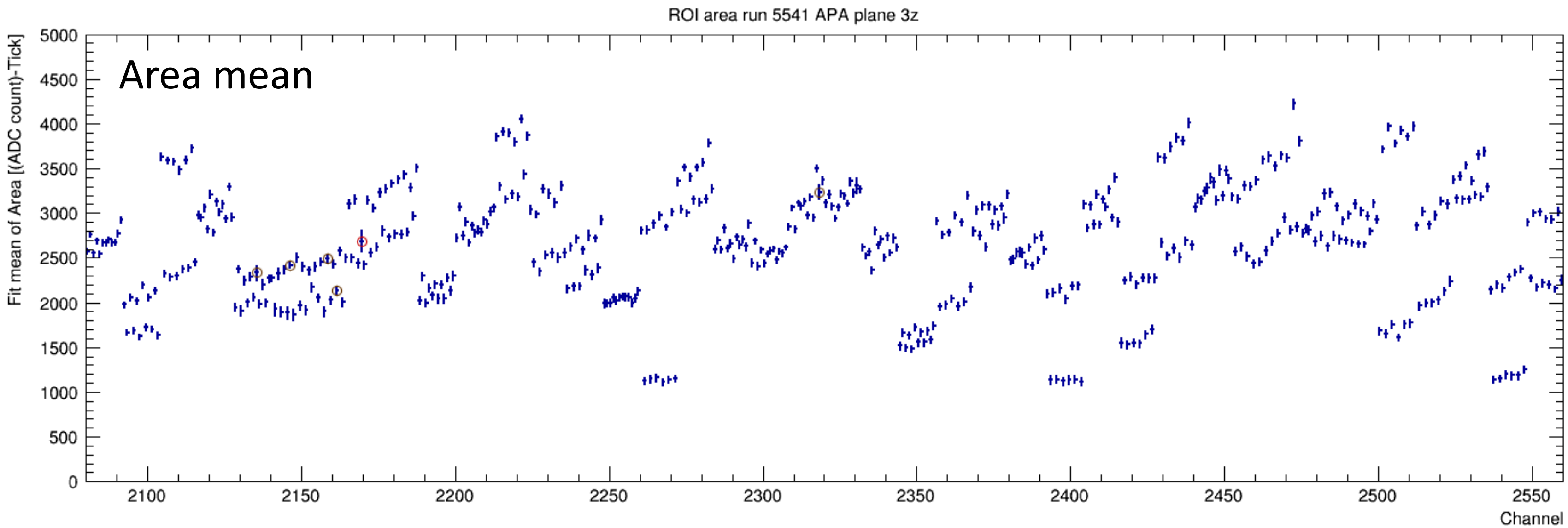


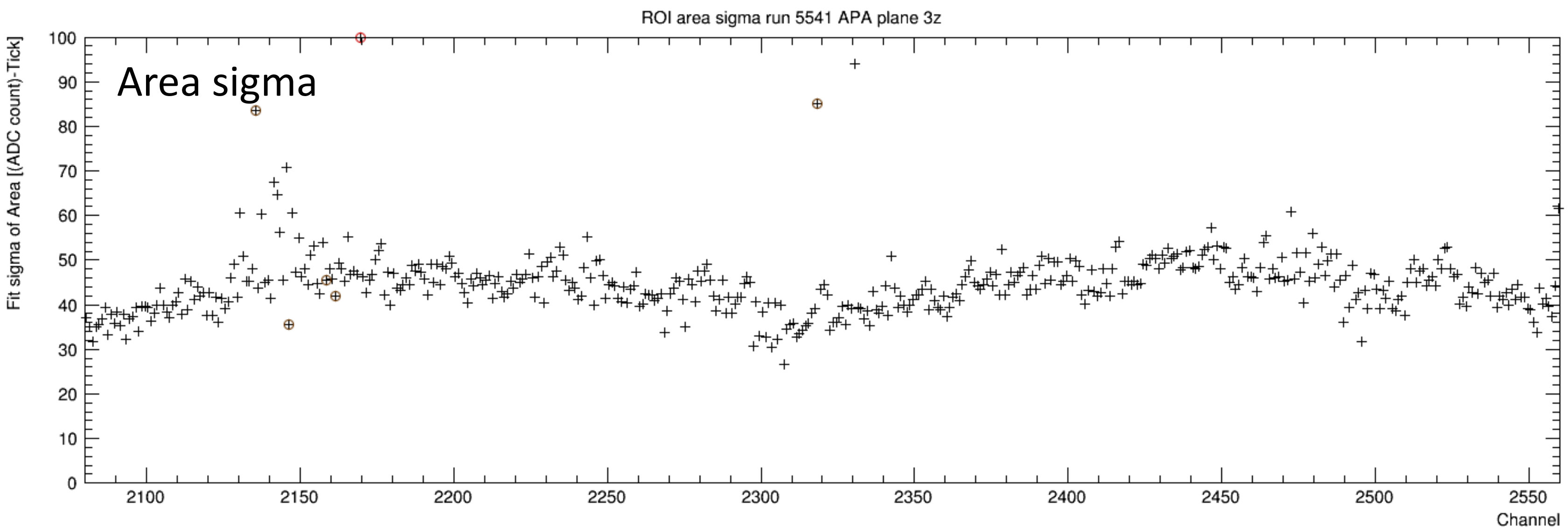
Fit height with old mitigation



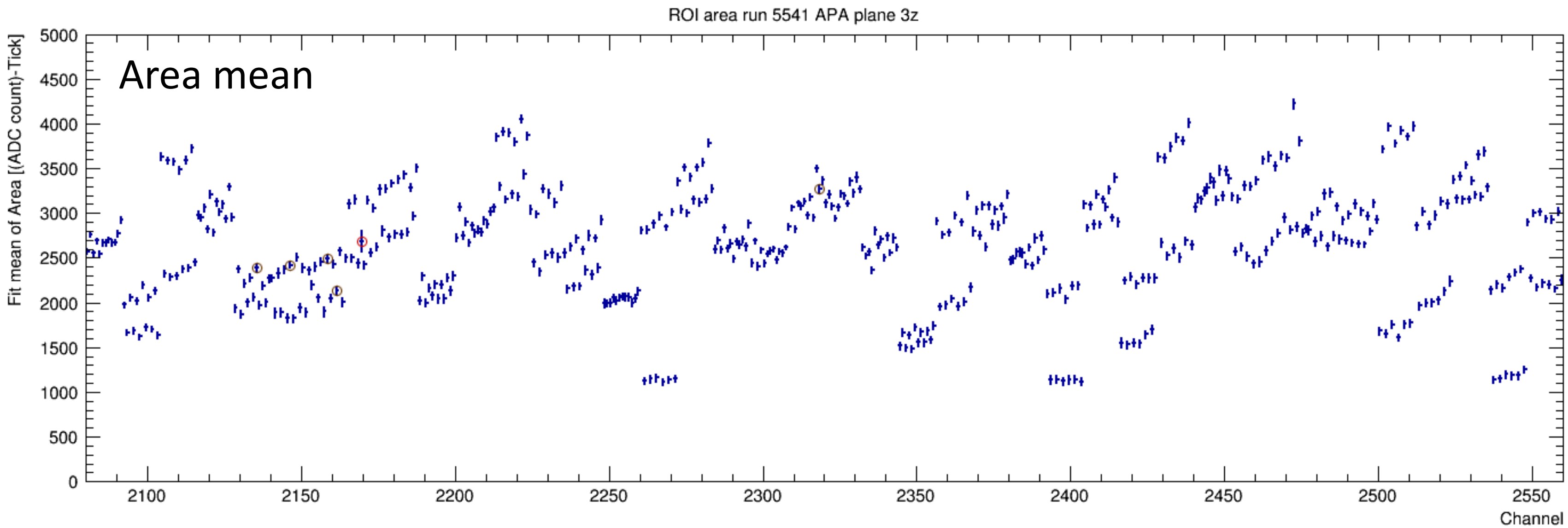


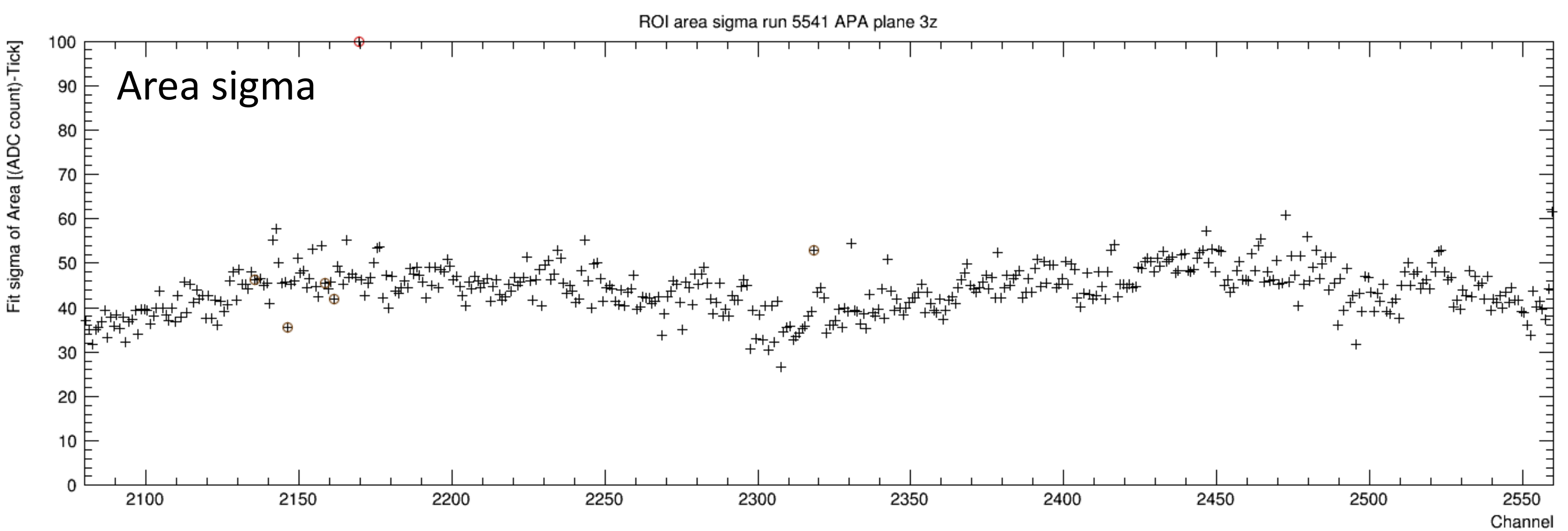
Signal area unmitigated



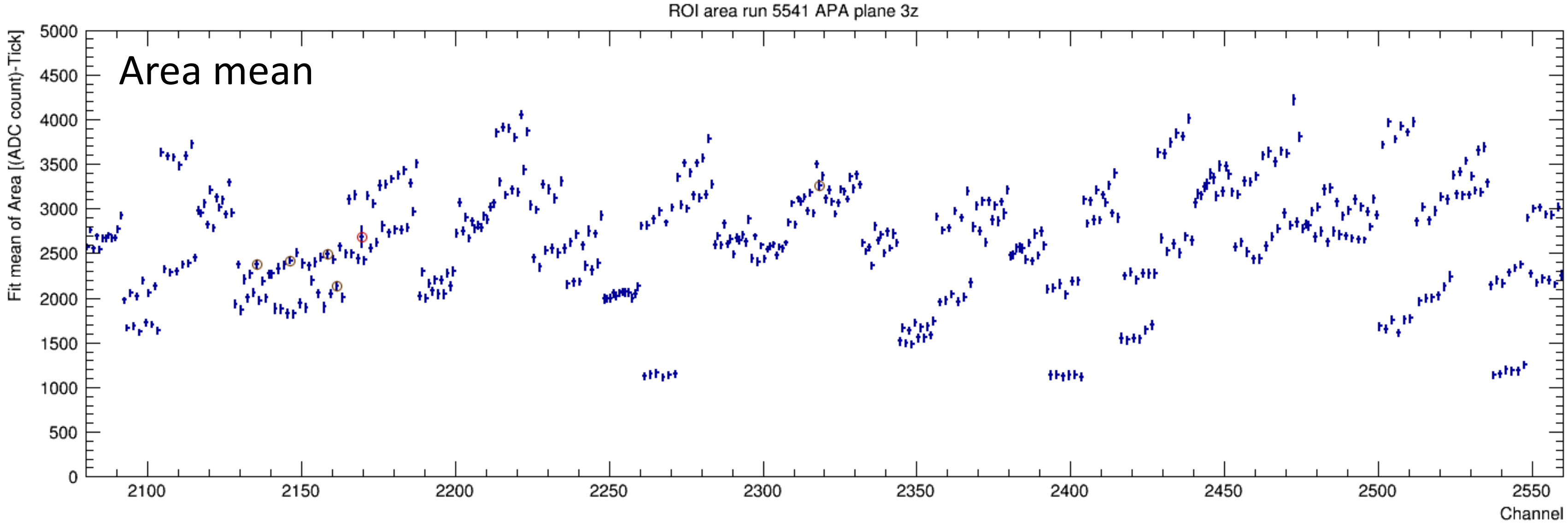


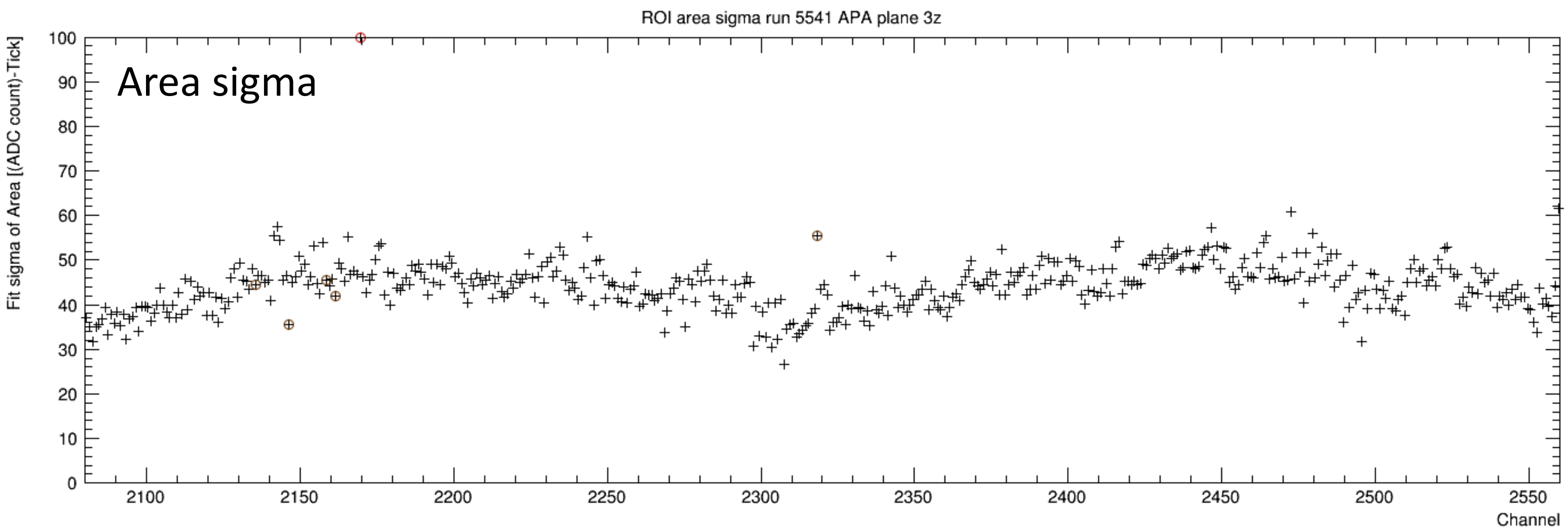
Signal area with new mitigation



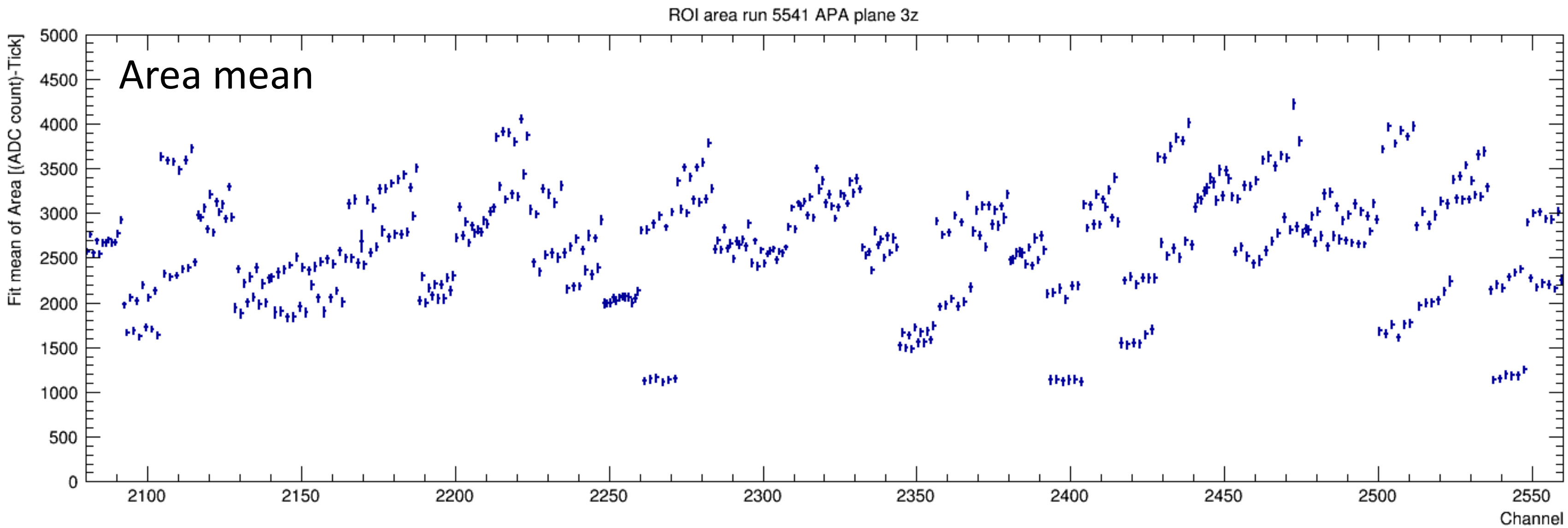


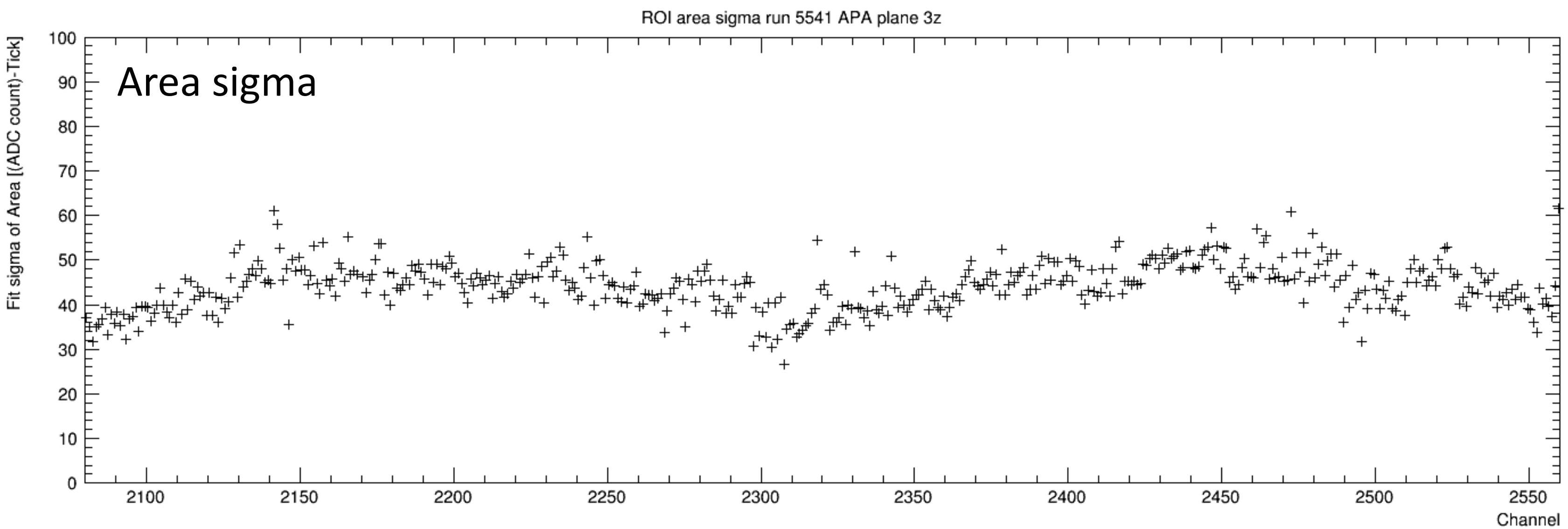
Signal area with ccv1 mitigation



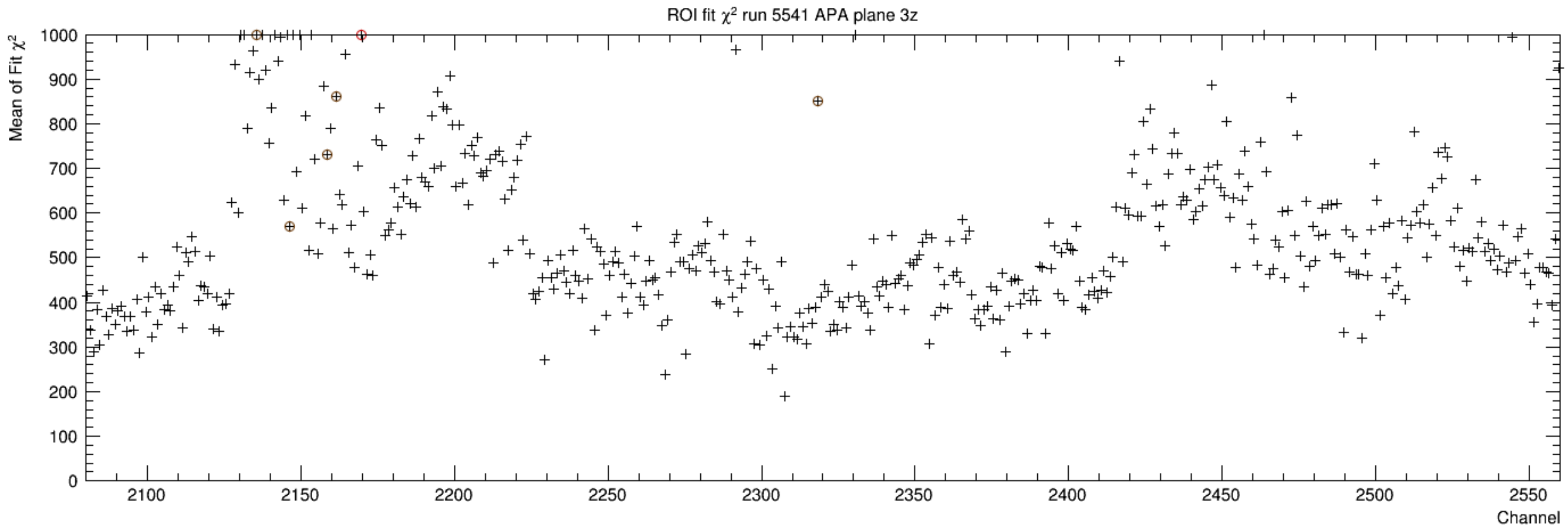


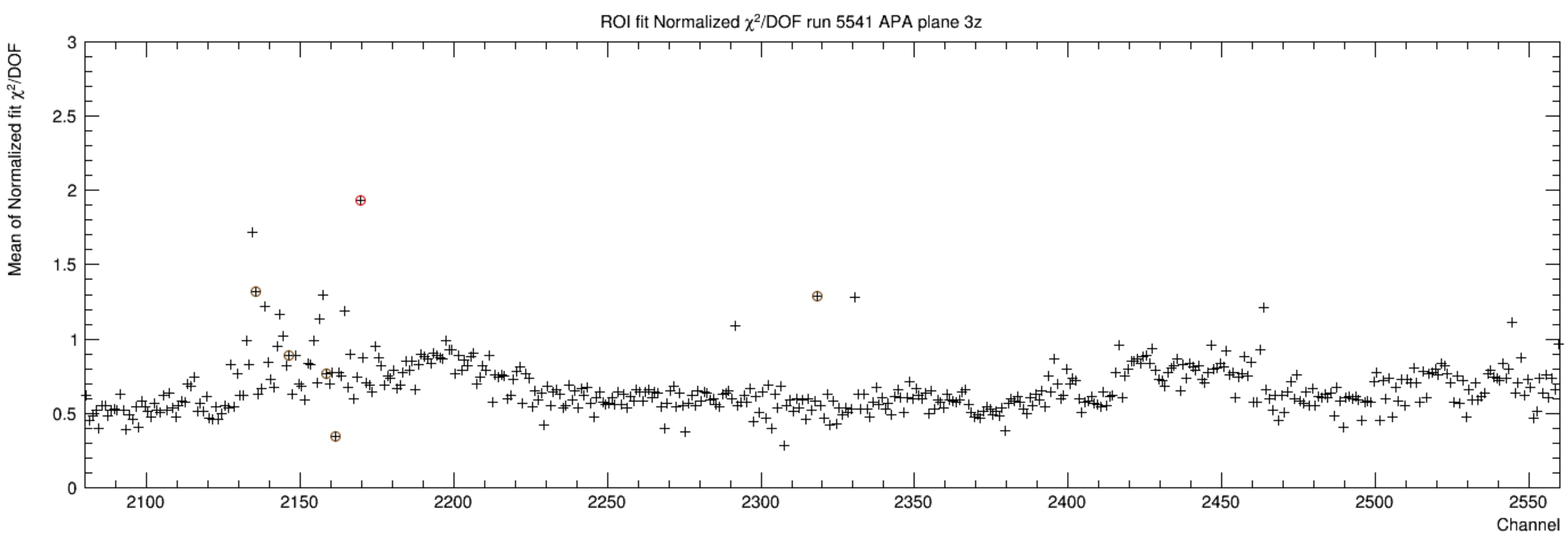
Signal area with old mitigation



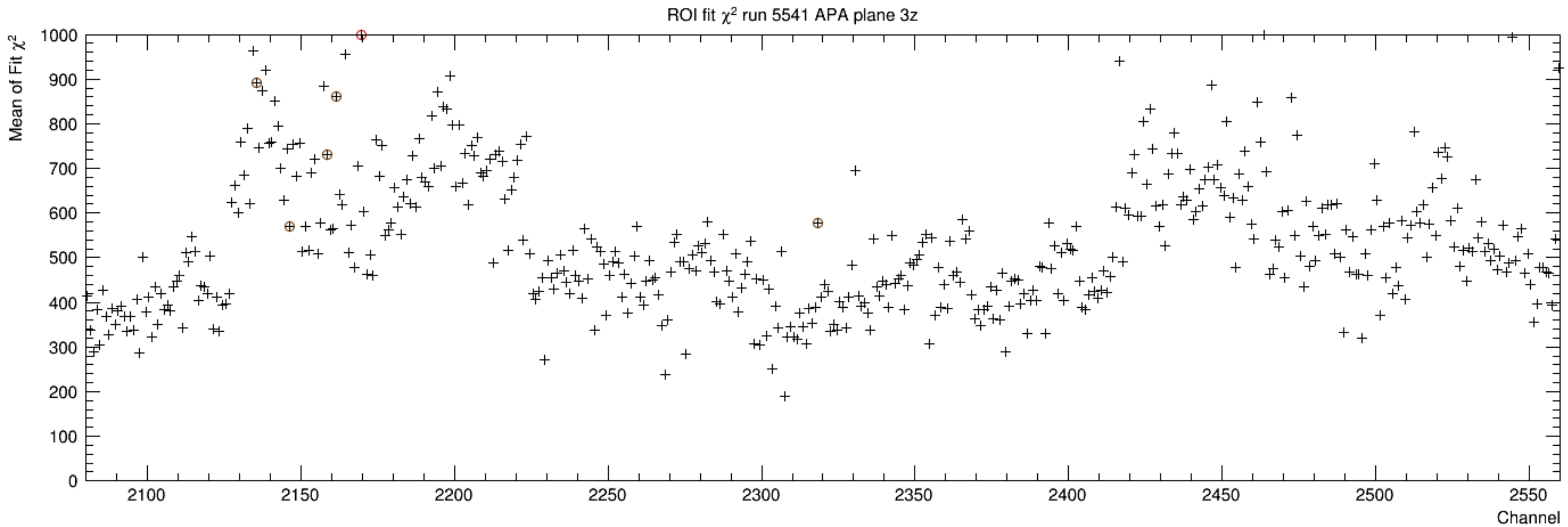


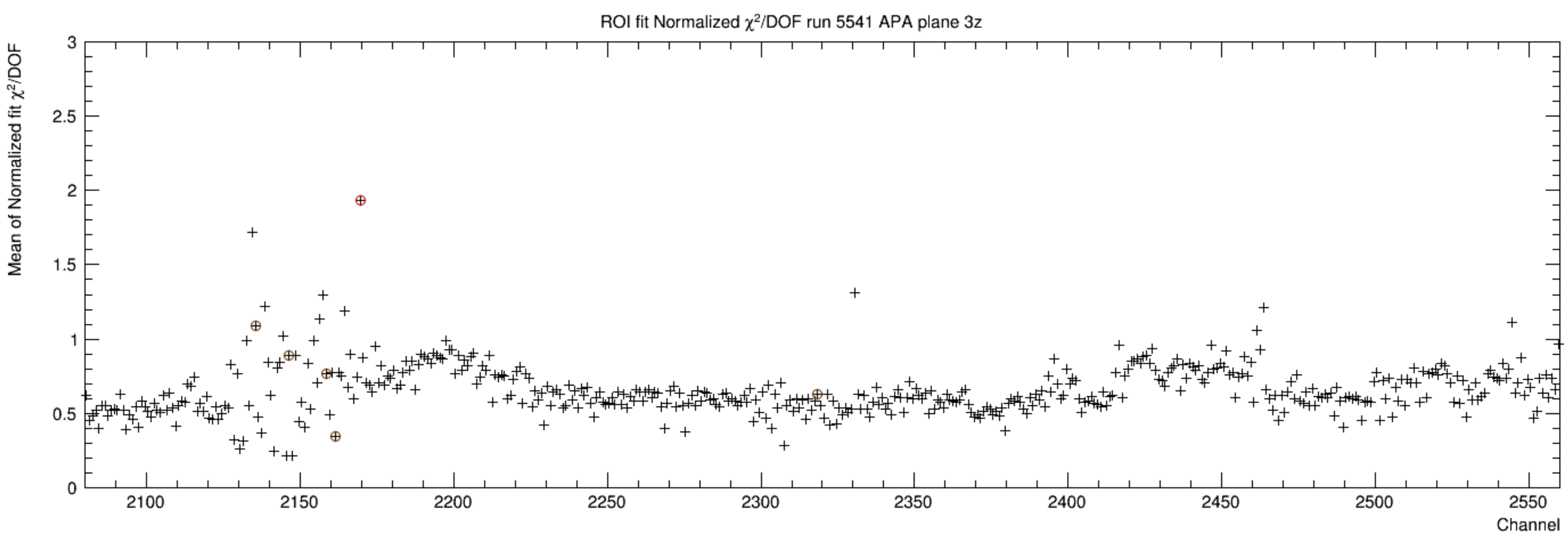
Fit chi-square unmitigated



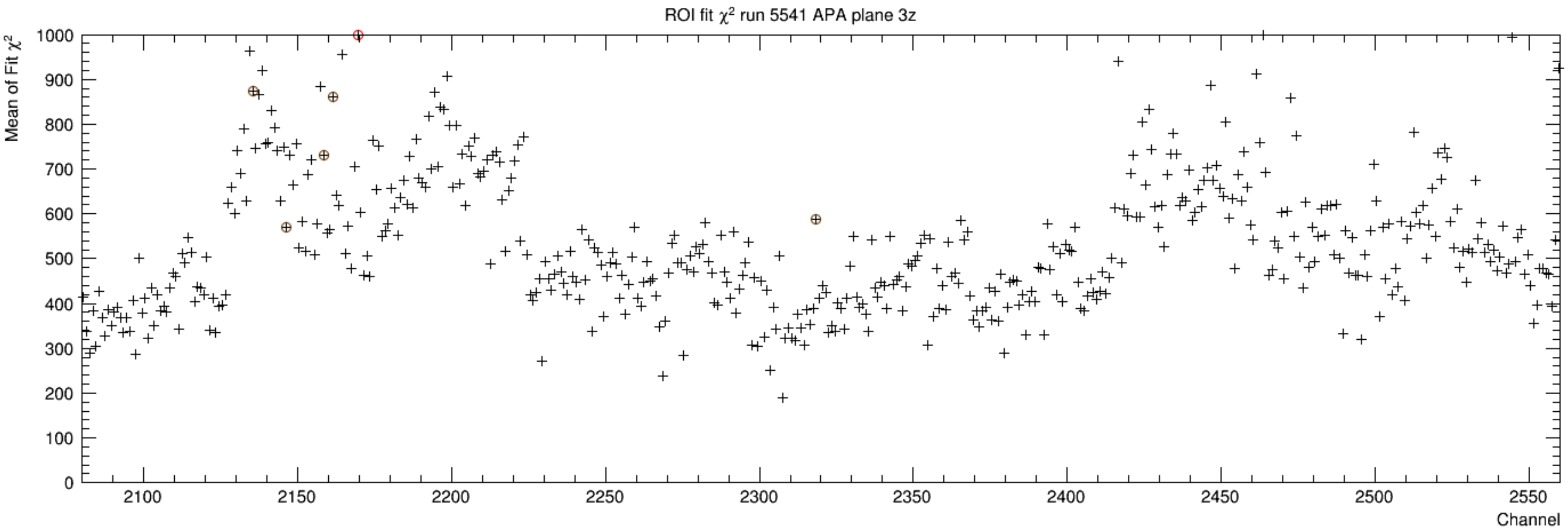


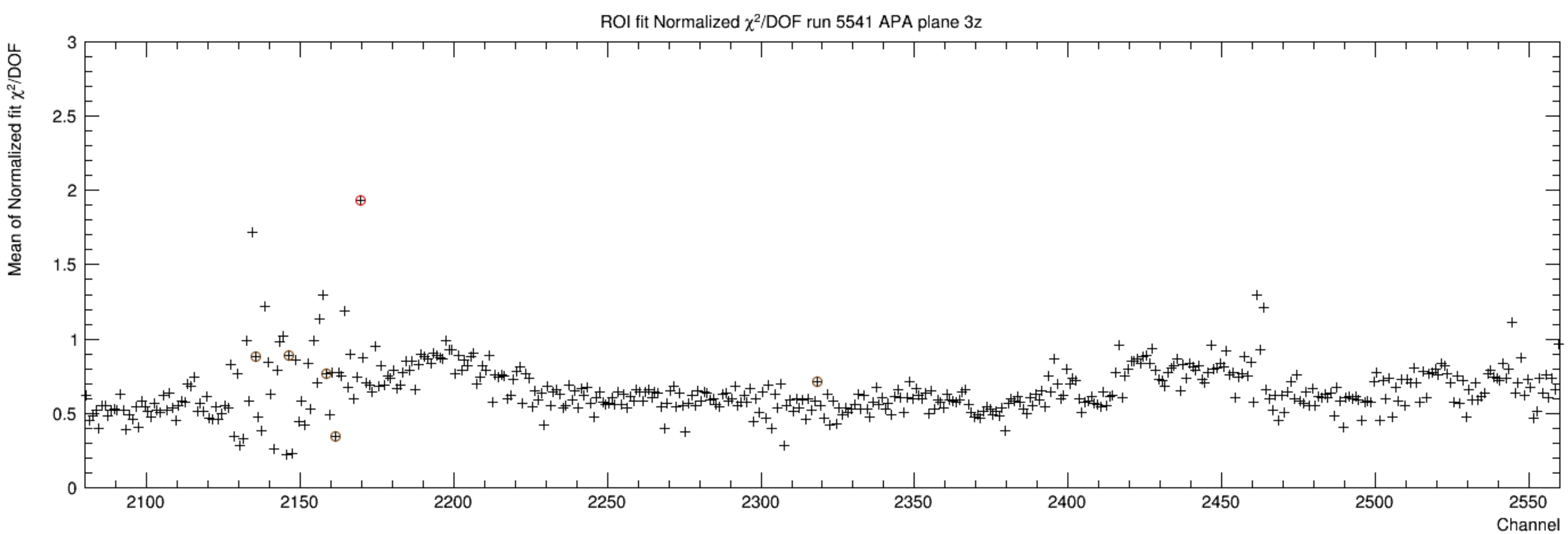
Fit chi-square with new mitigation



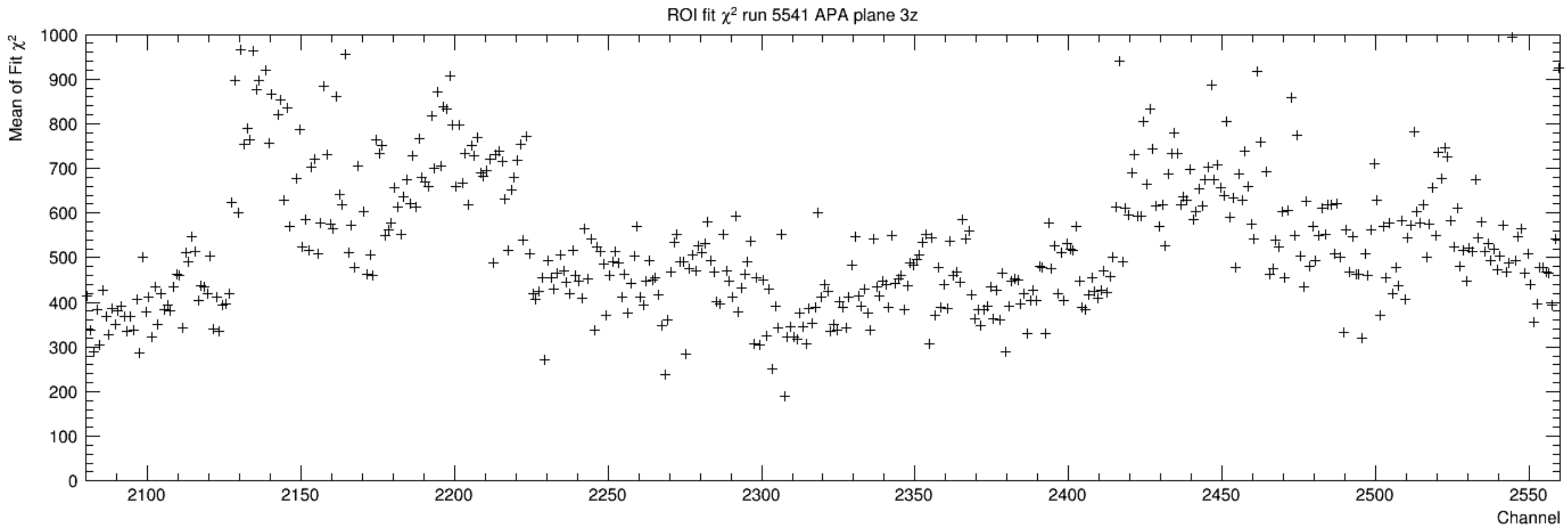


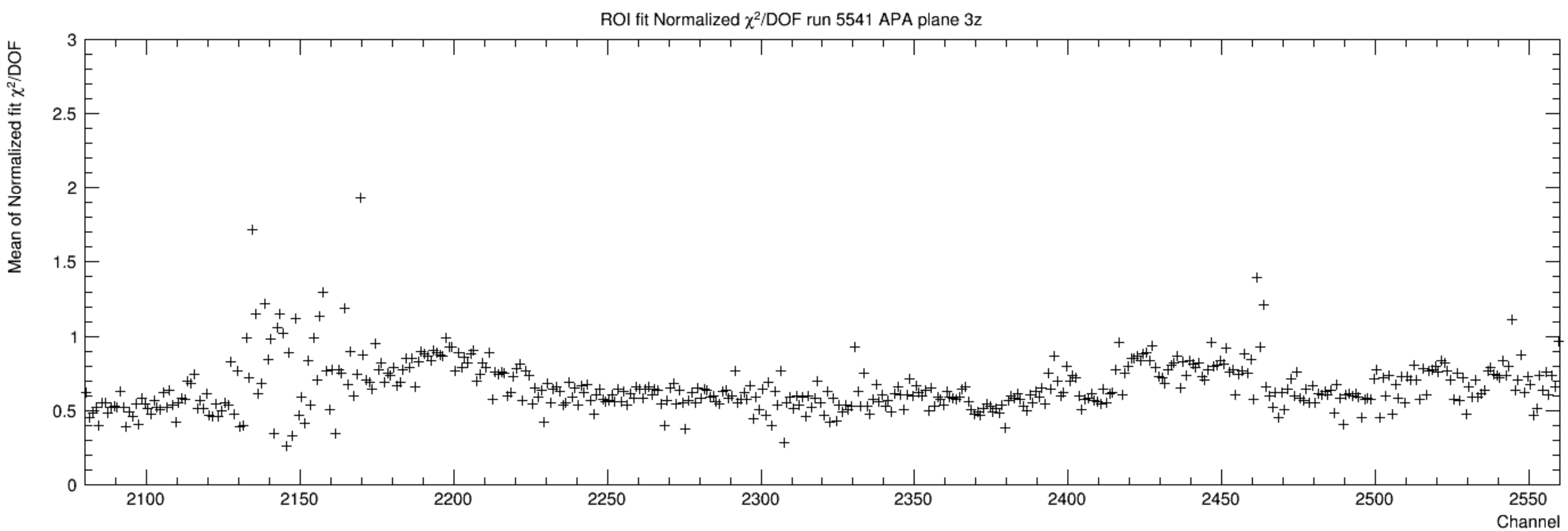
Fit chi-square with ccv1 mitigation





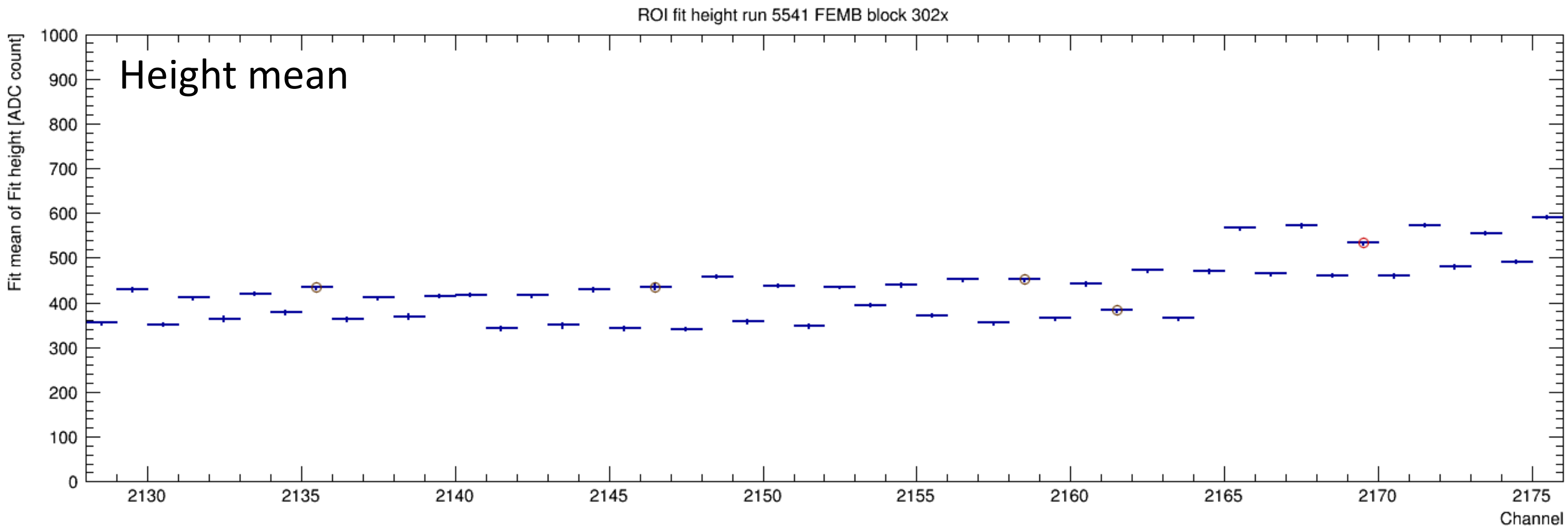
Fit chi-square with old mitigation

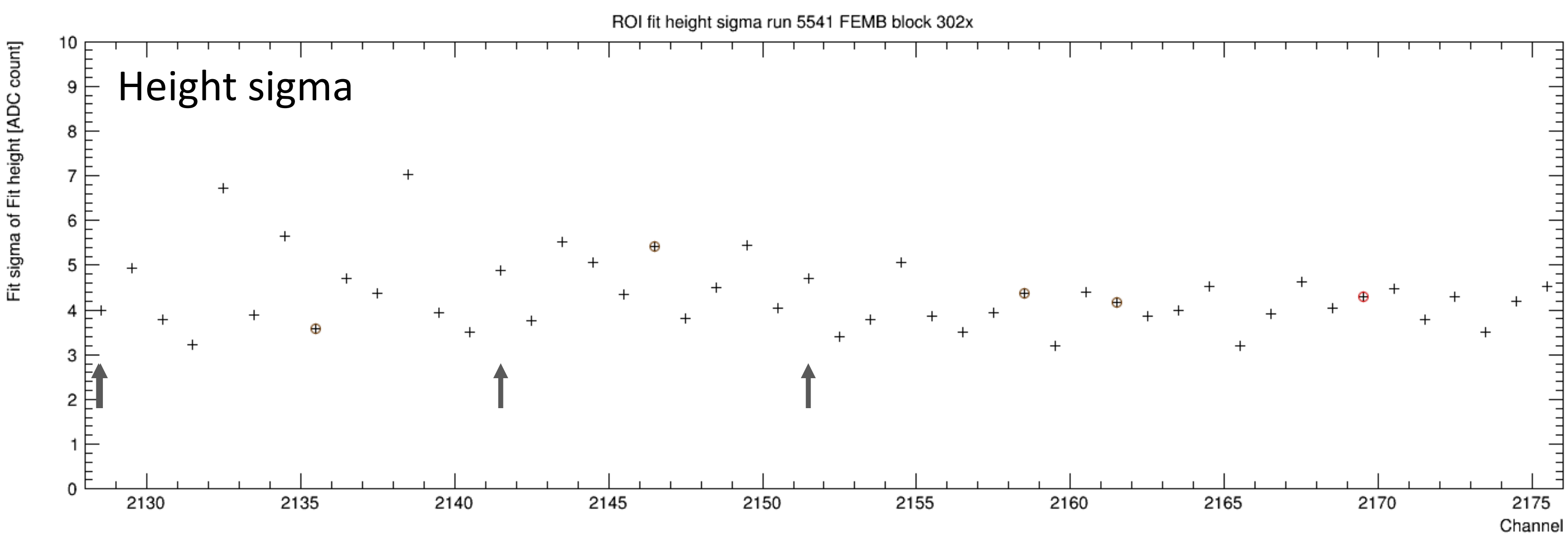




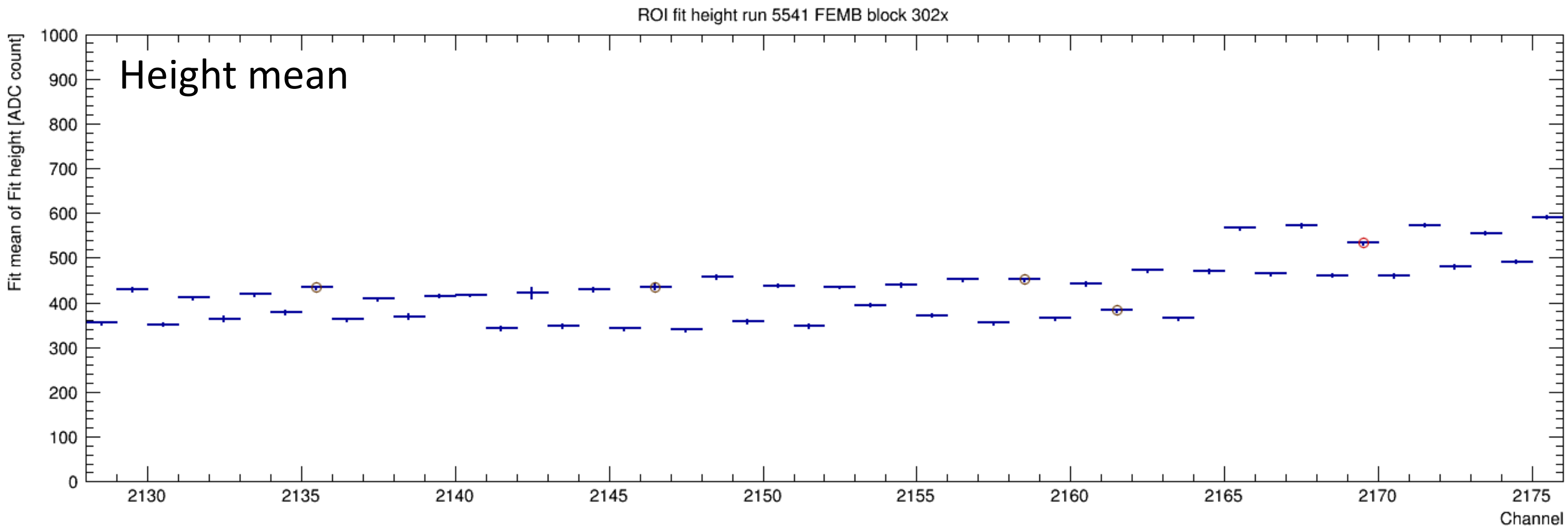
Channel summary plots for FEMB 302x

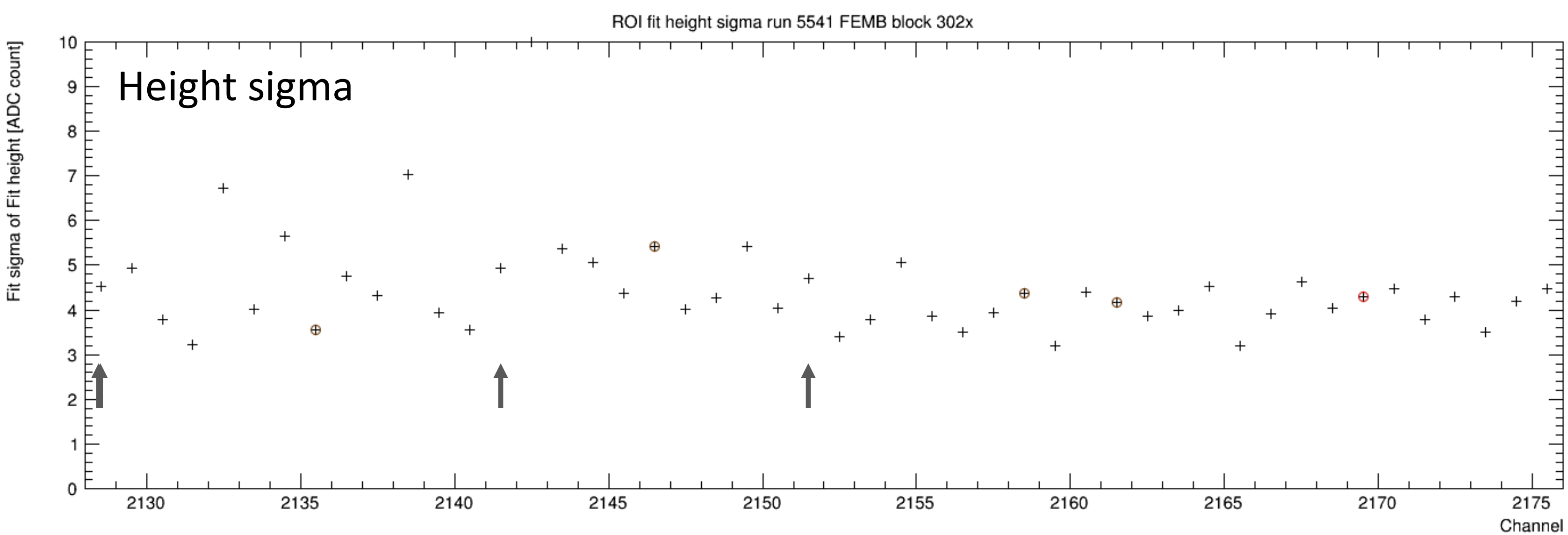
Fit height unmitigated



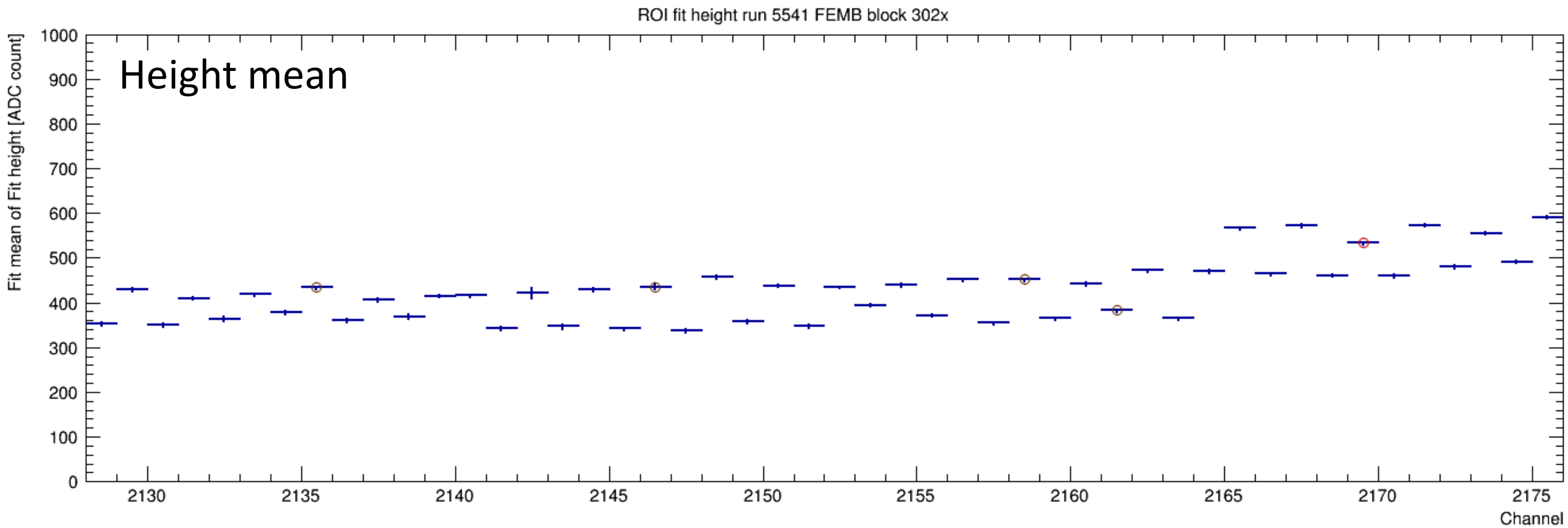


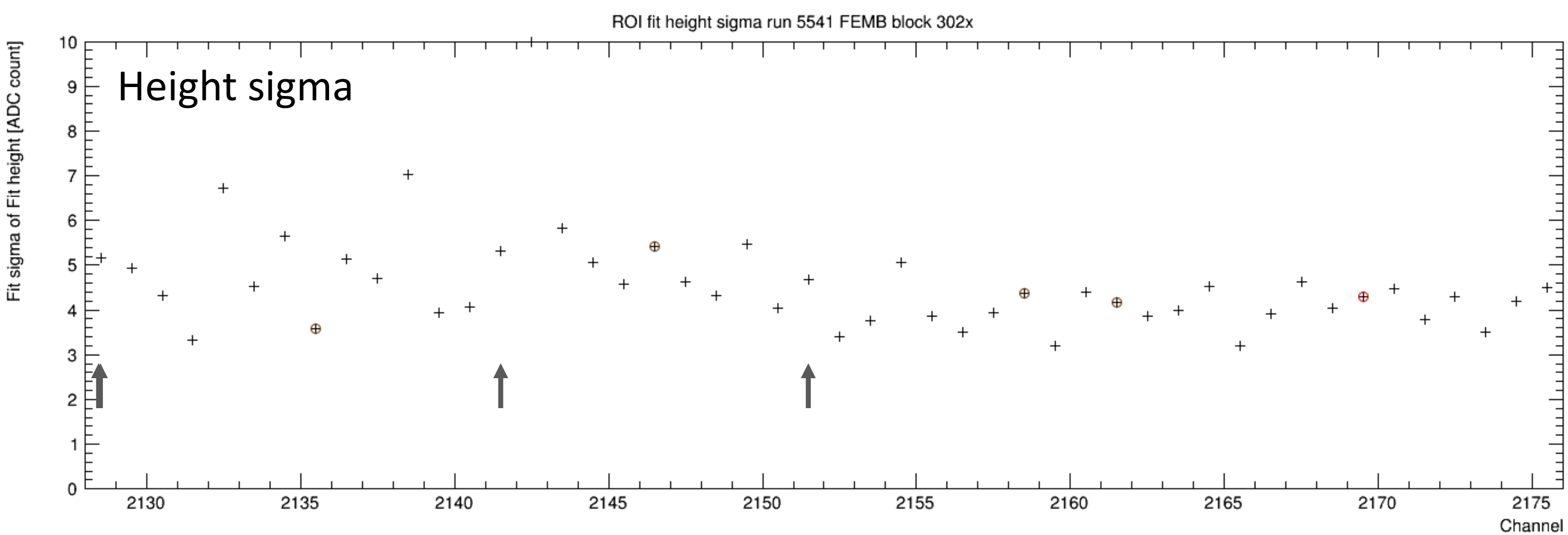
Fit height with new mitigation



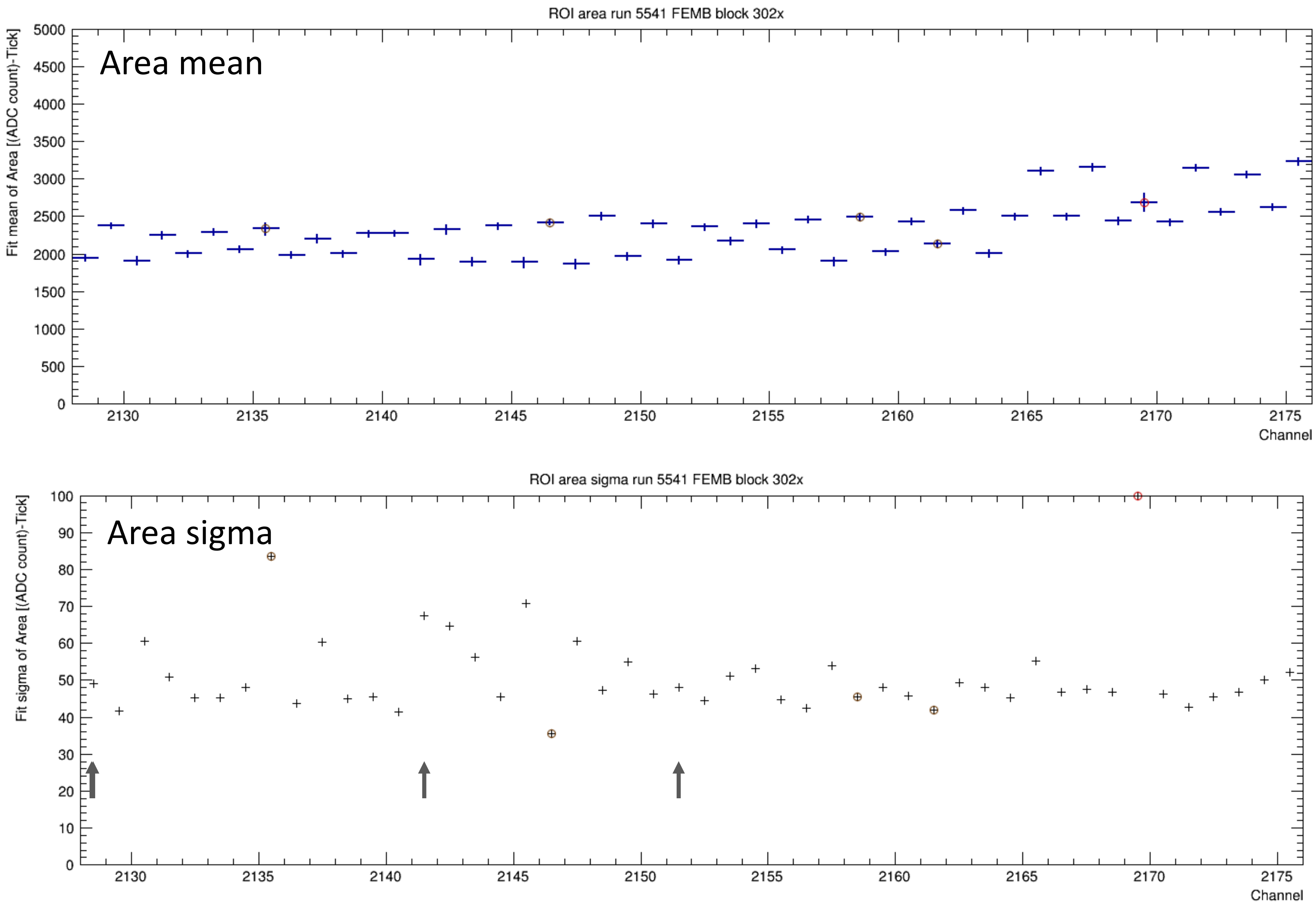


Fit height with old mitigation

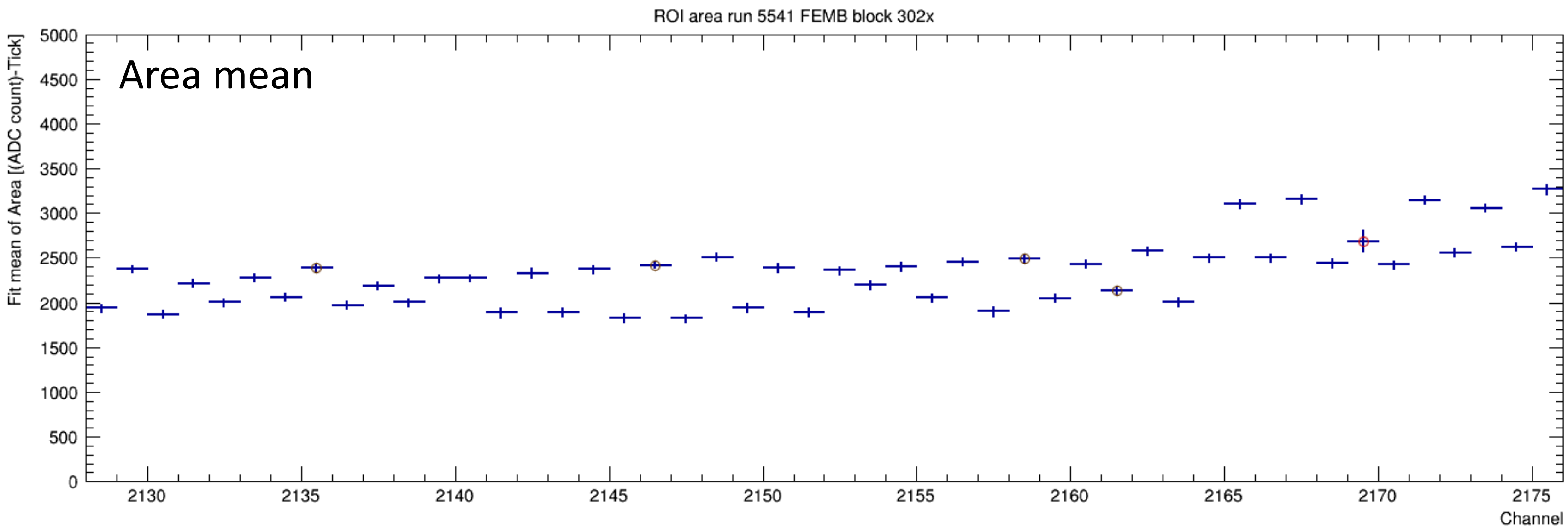


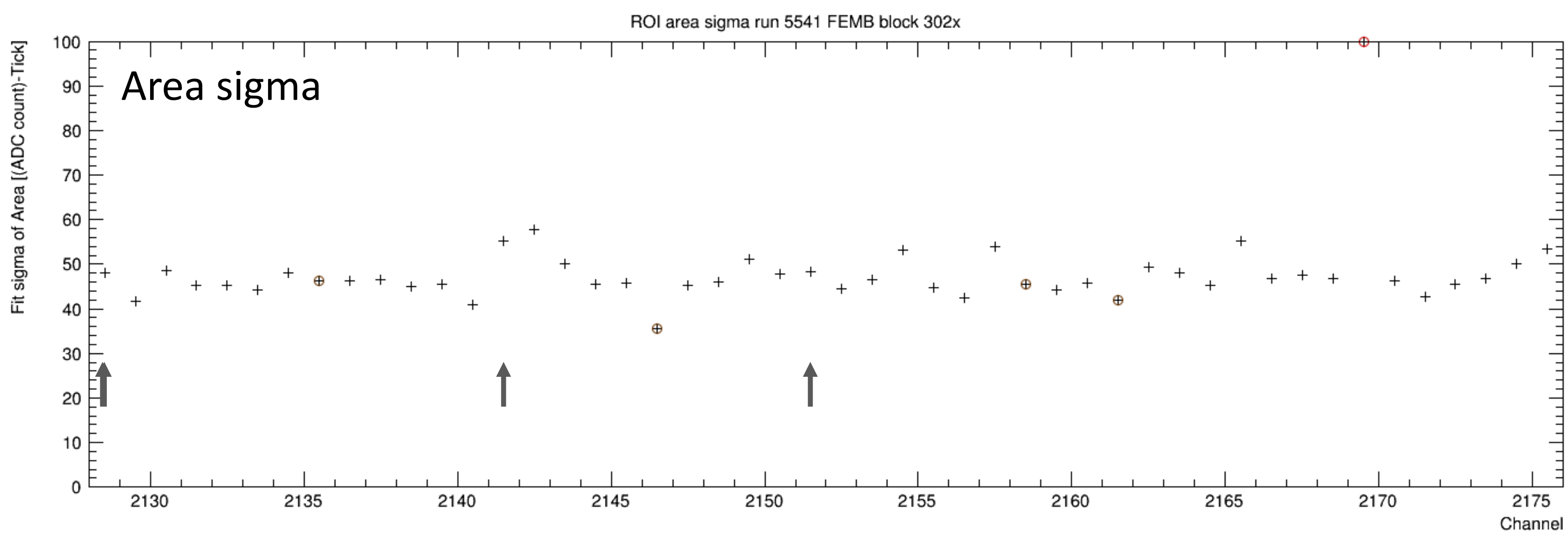


Signal area unmitigated

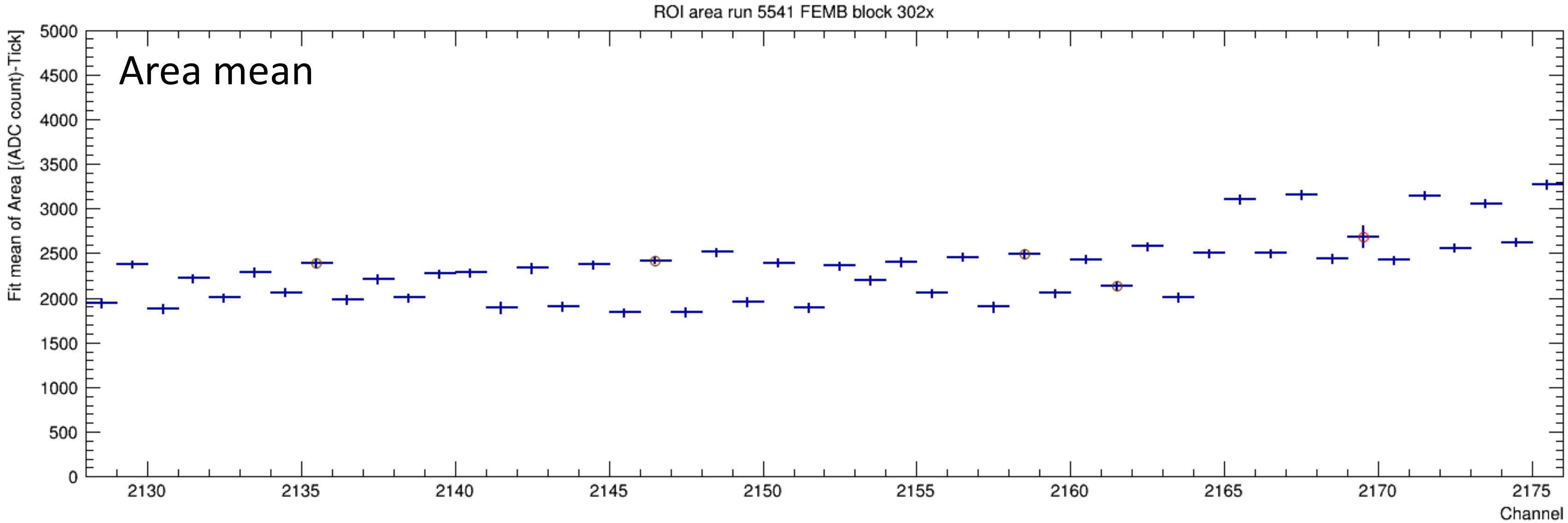


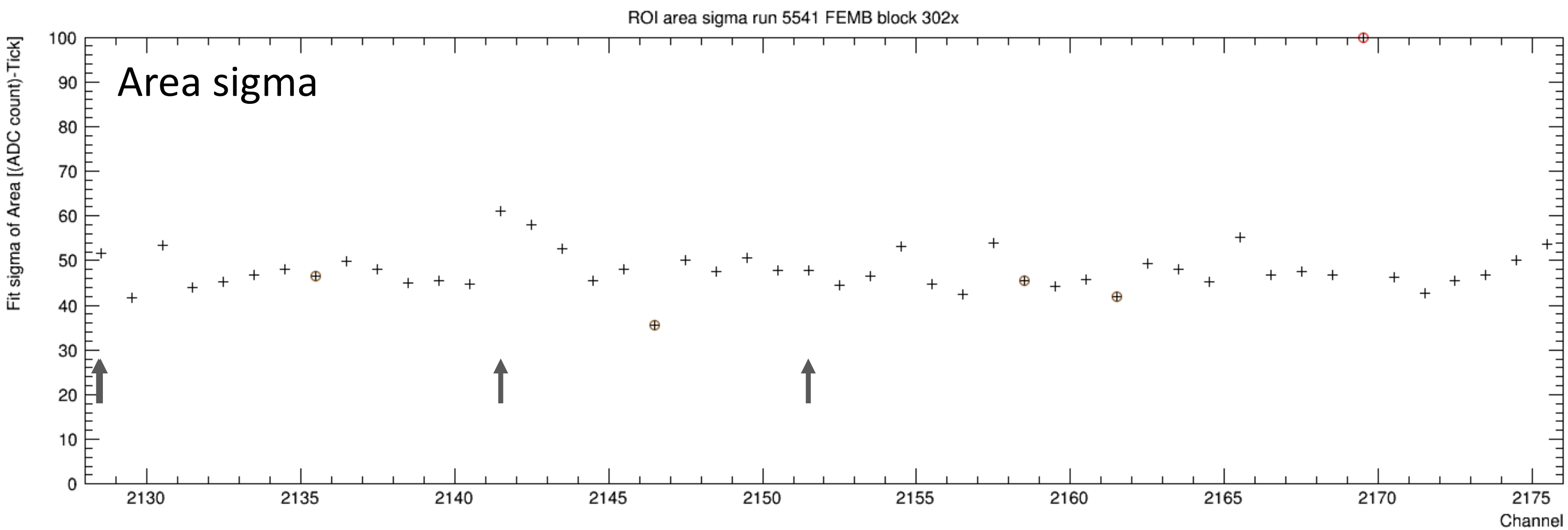
Signal area with new mitigation



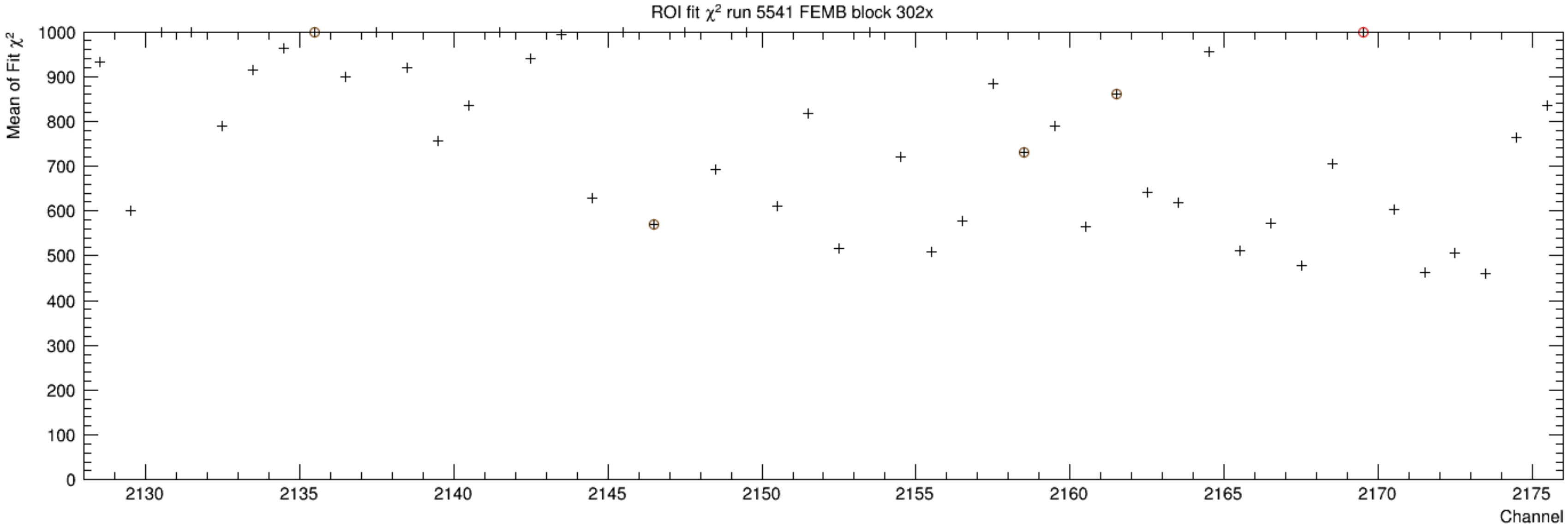


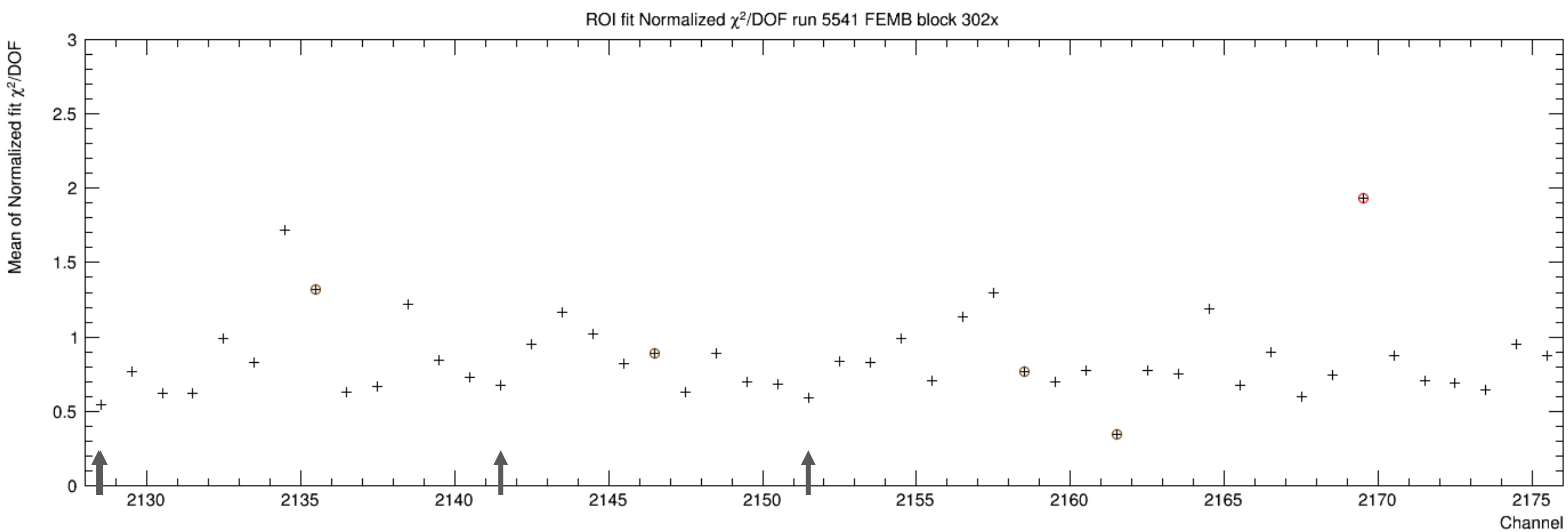
Signal area with old mitigation



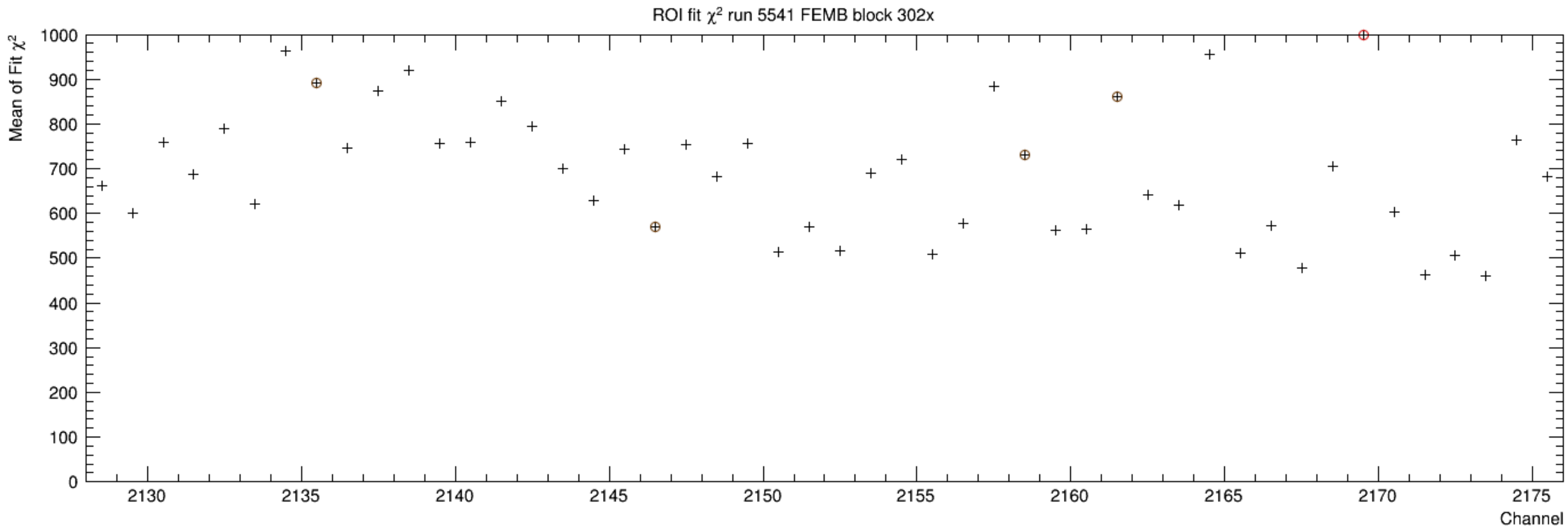


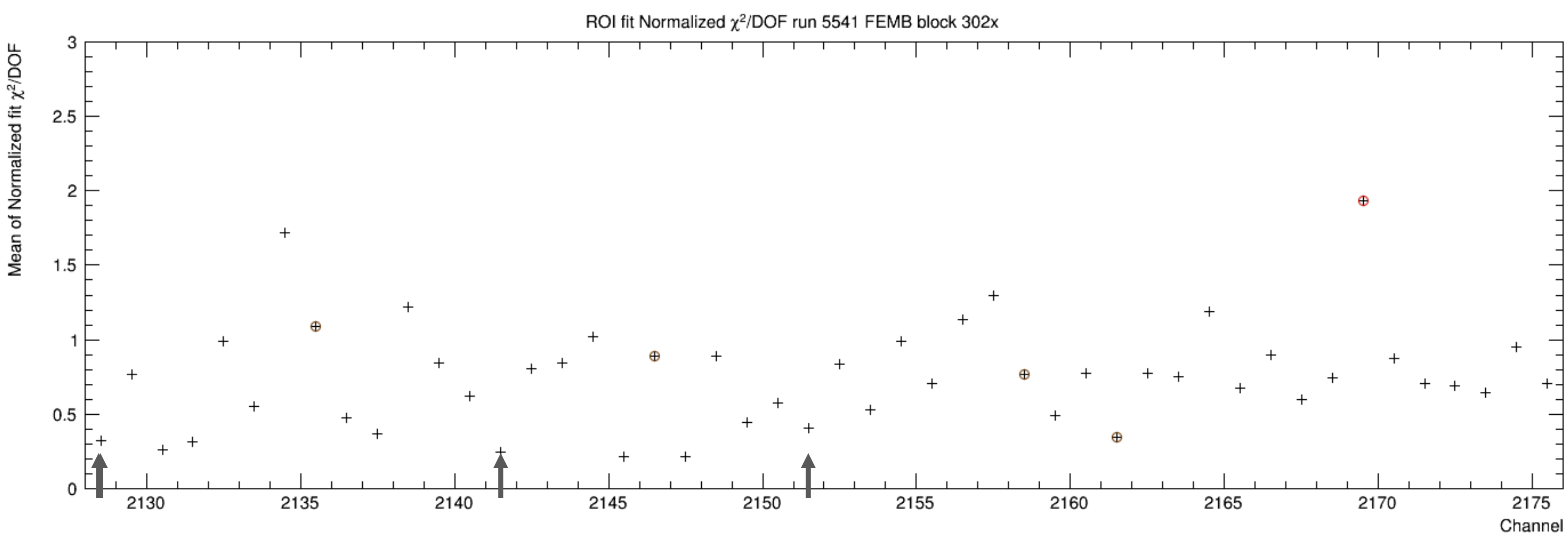
Fit chi-square unmitigated



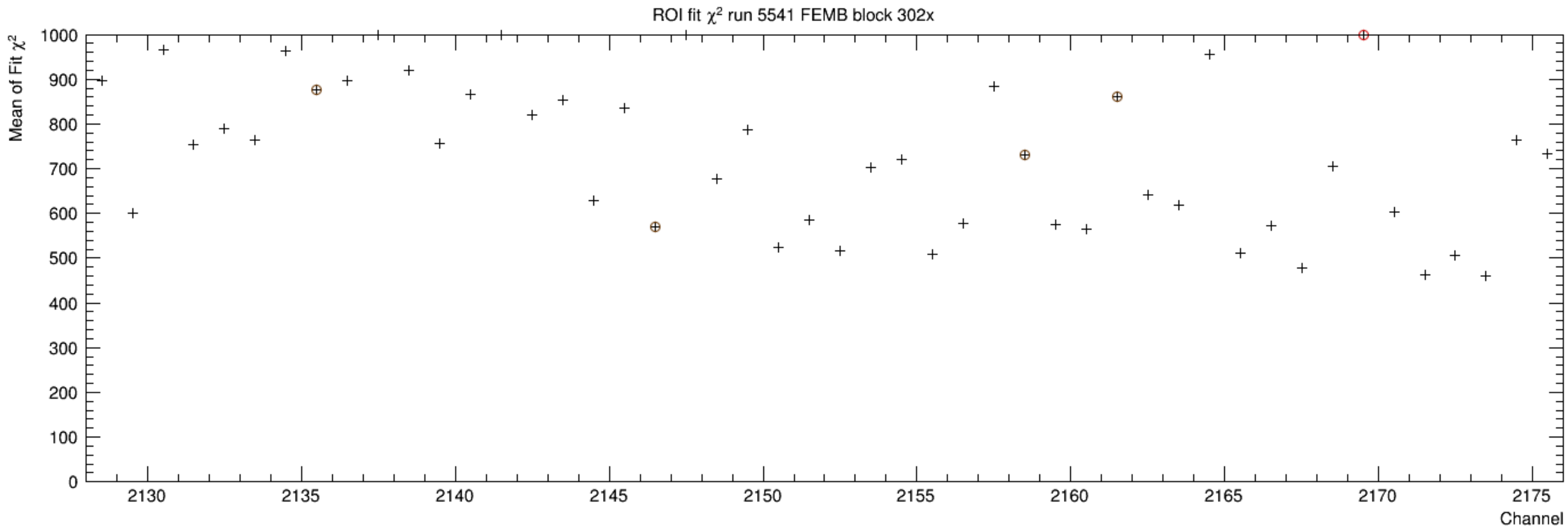


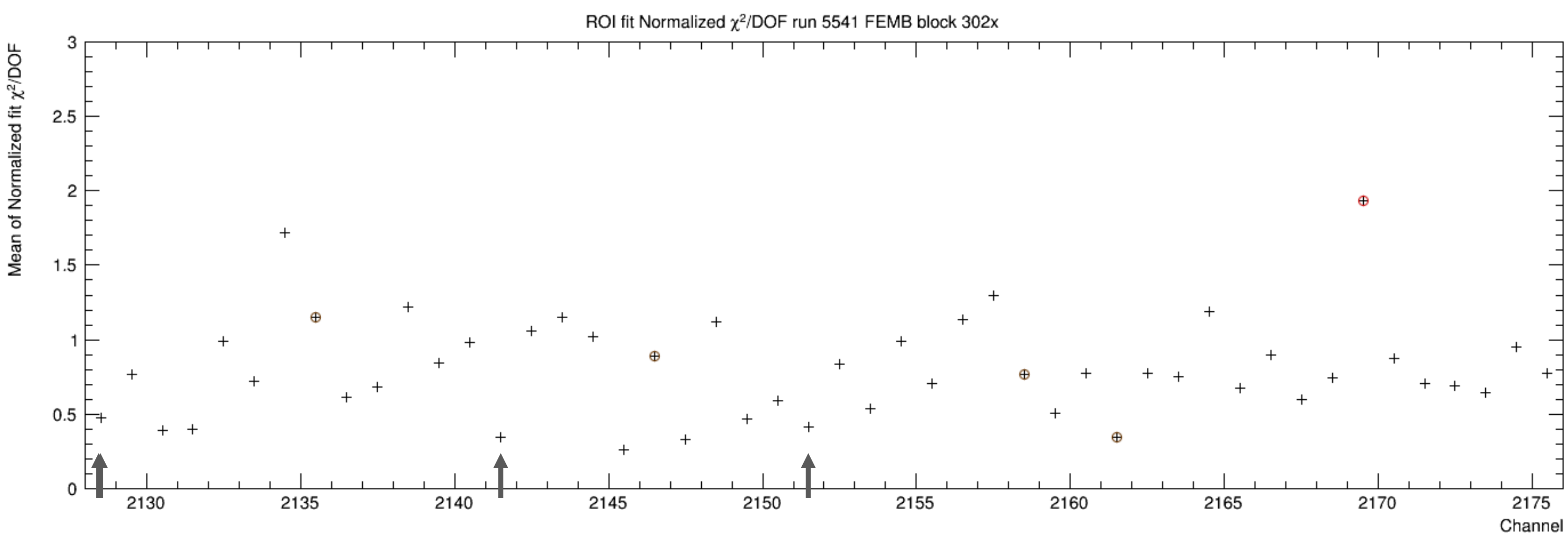
Fit chi-square with new mitigation





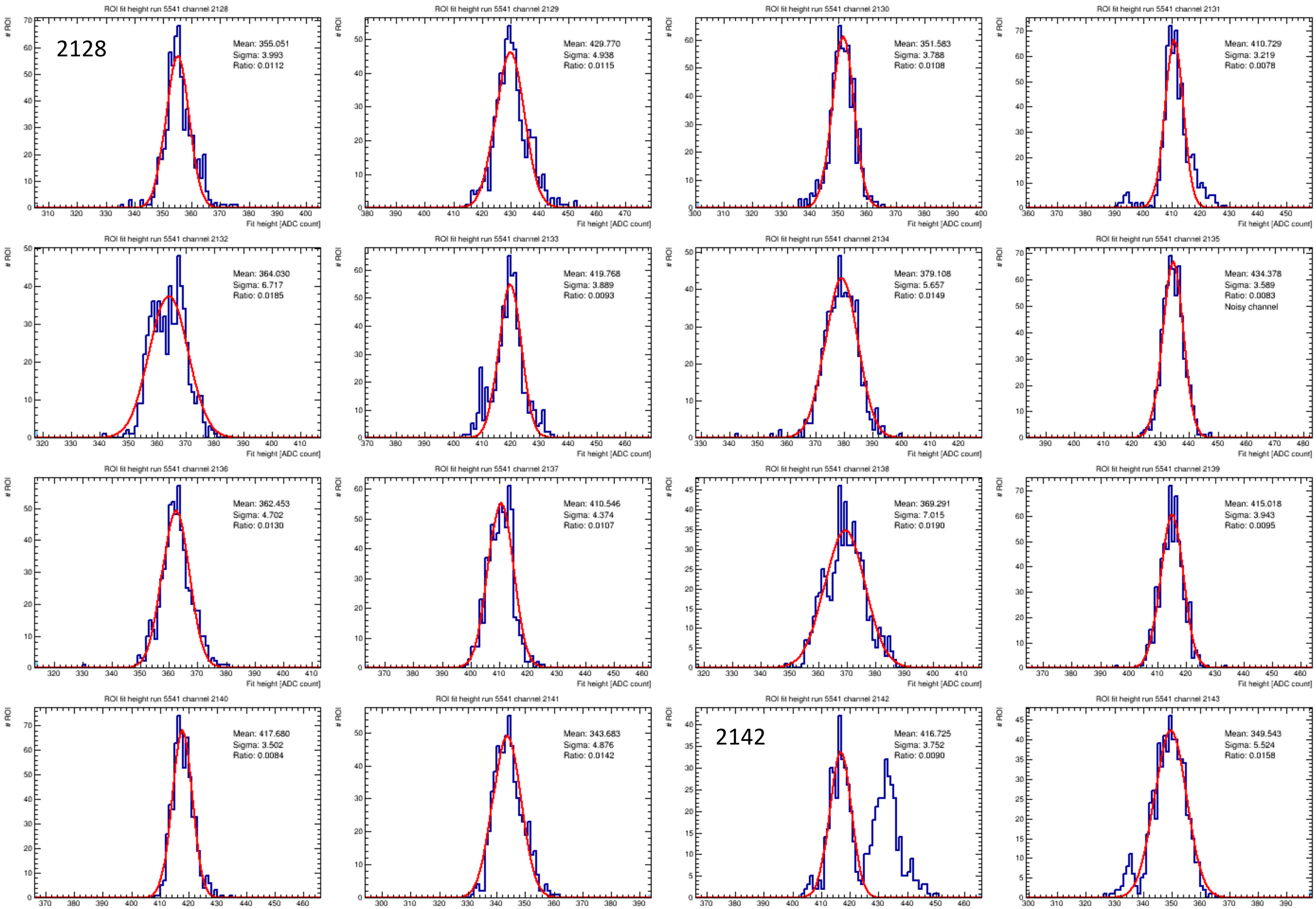
Fit chi-square with old mitigation



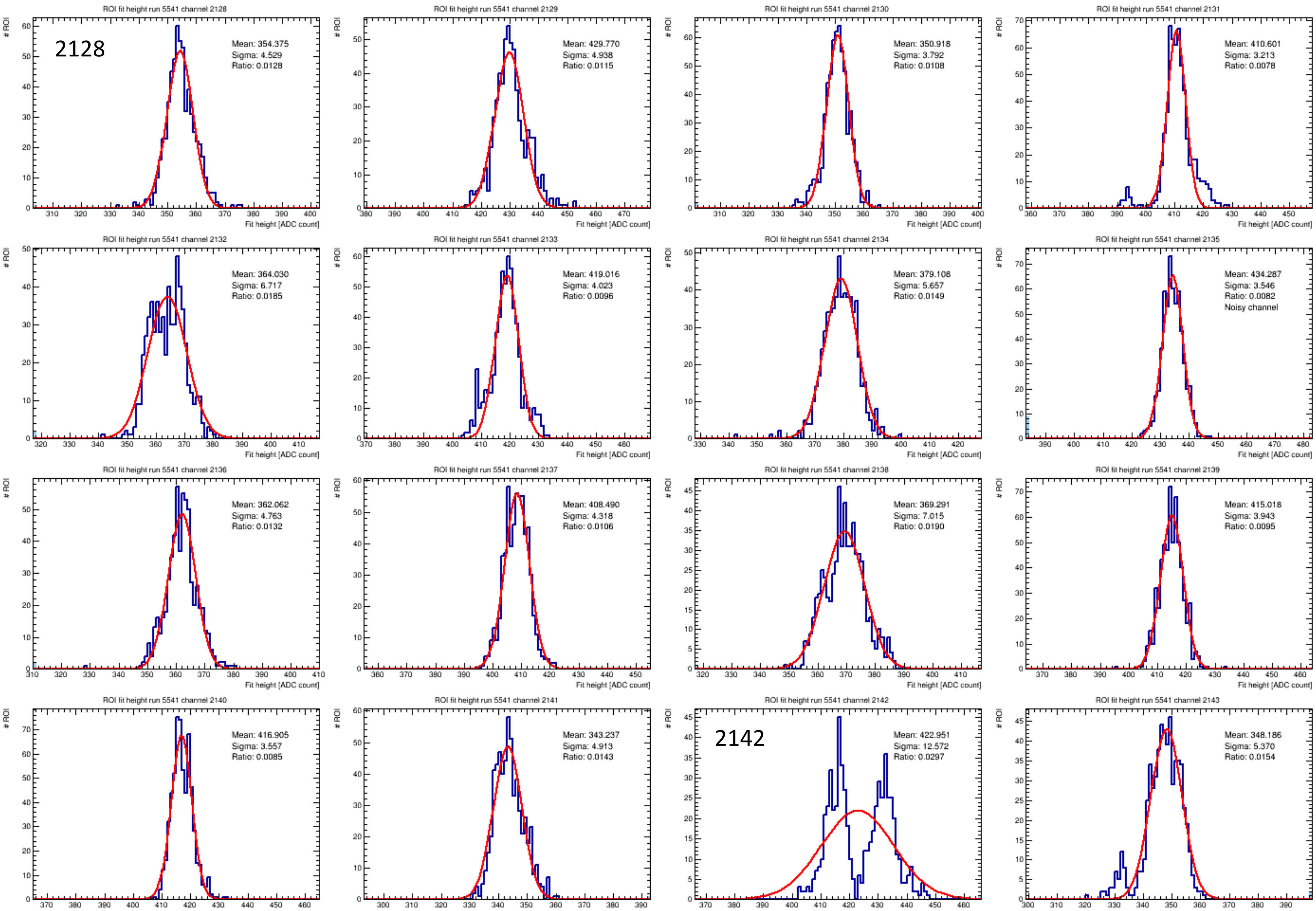


ROI summary plots for FEMB 302x

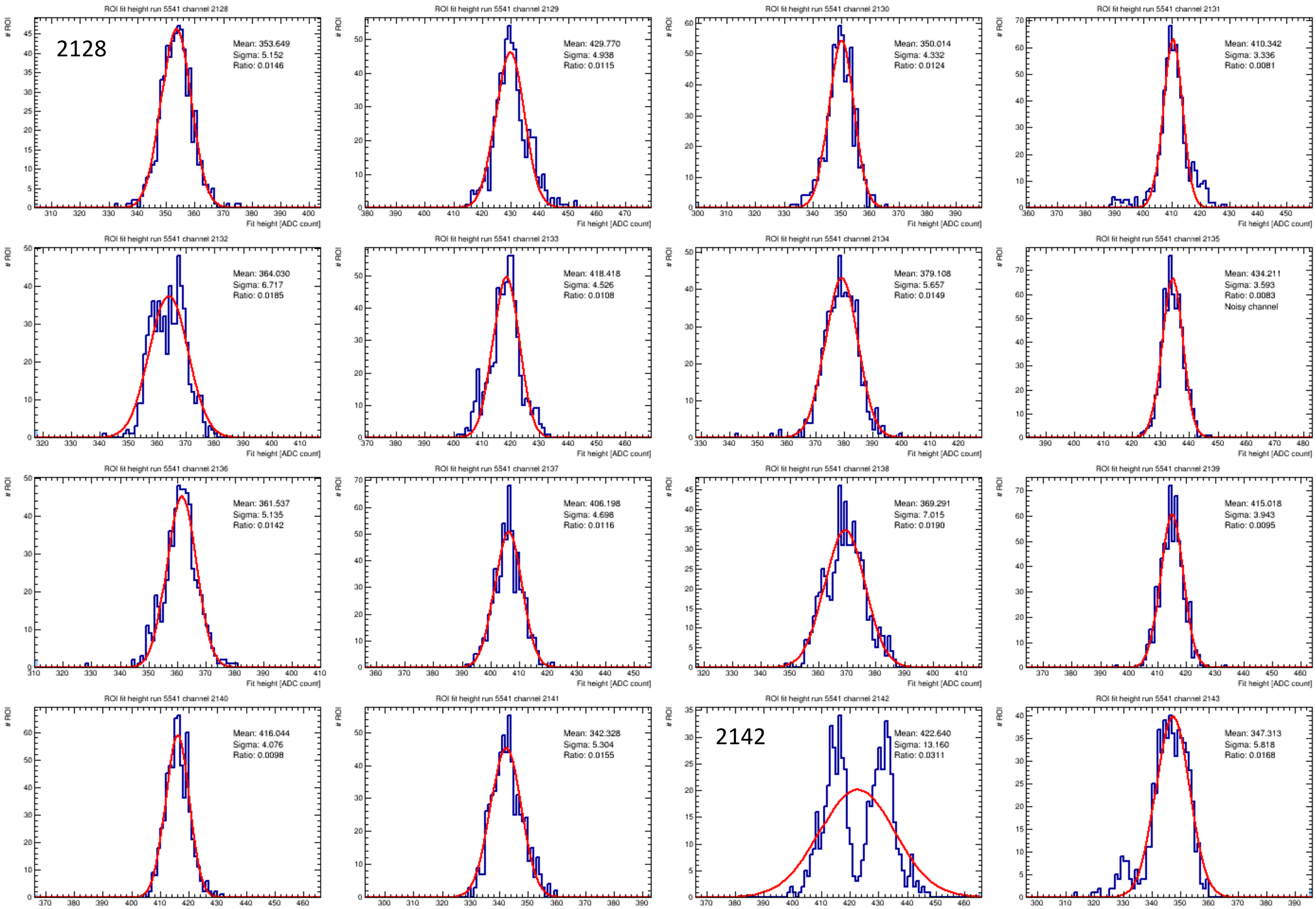
Height unmitigated



Height with new mitigation

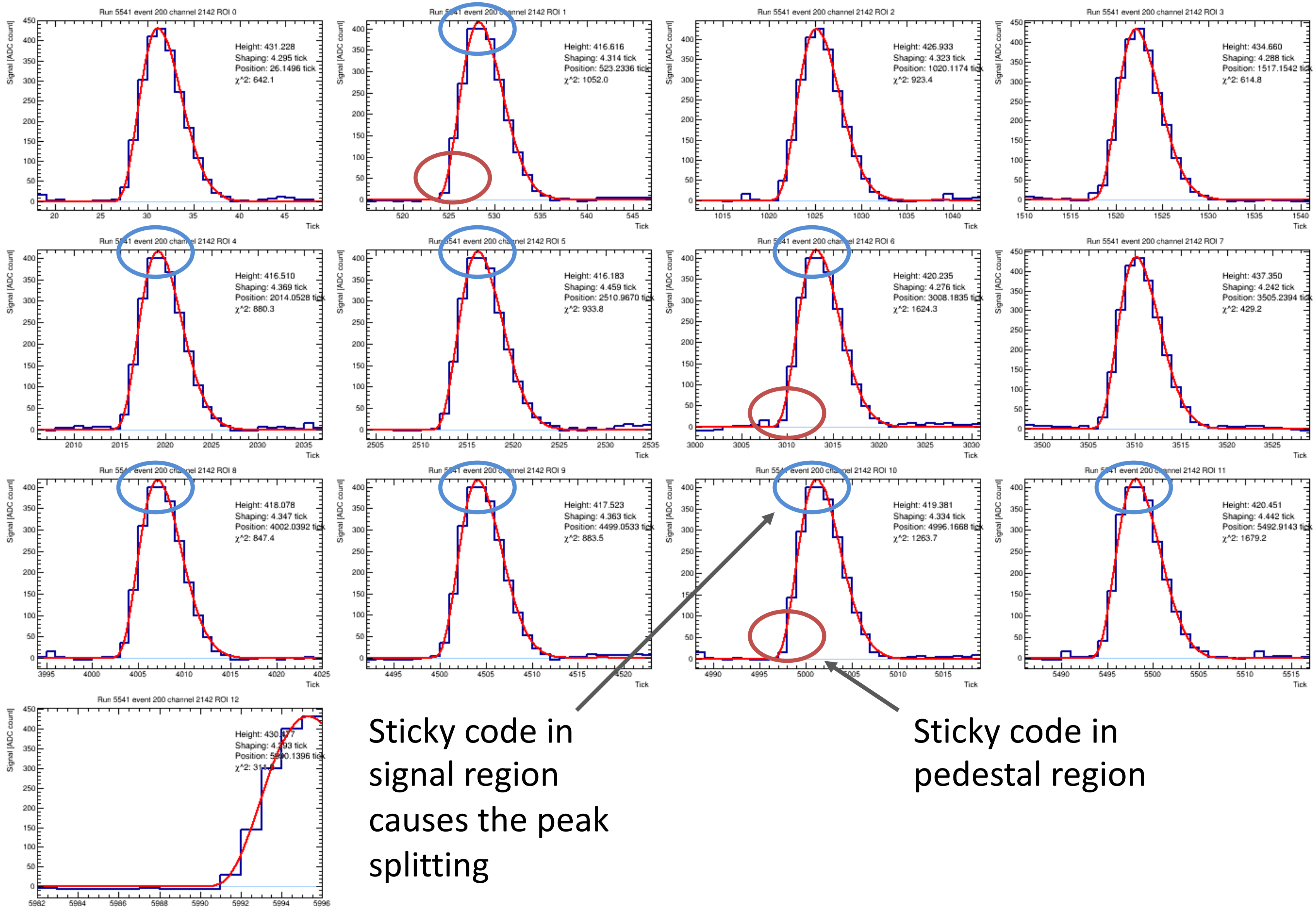


Height with old mitigation



ROIs for channel 2142

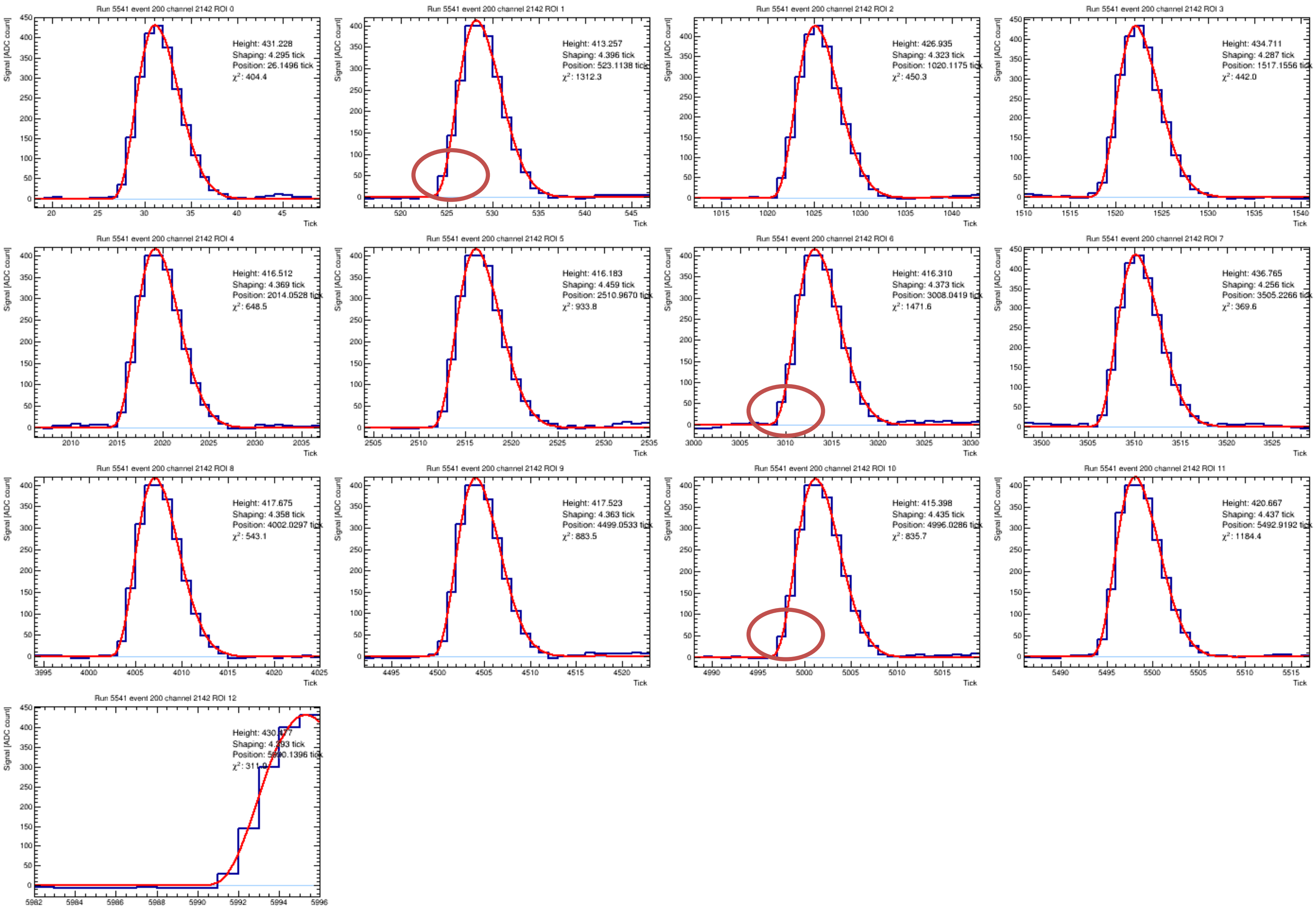
Unmitigated



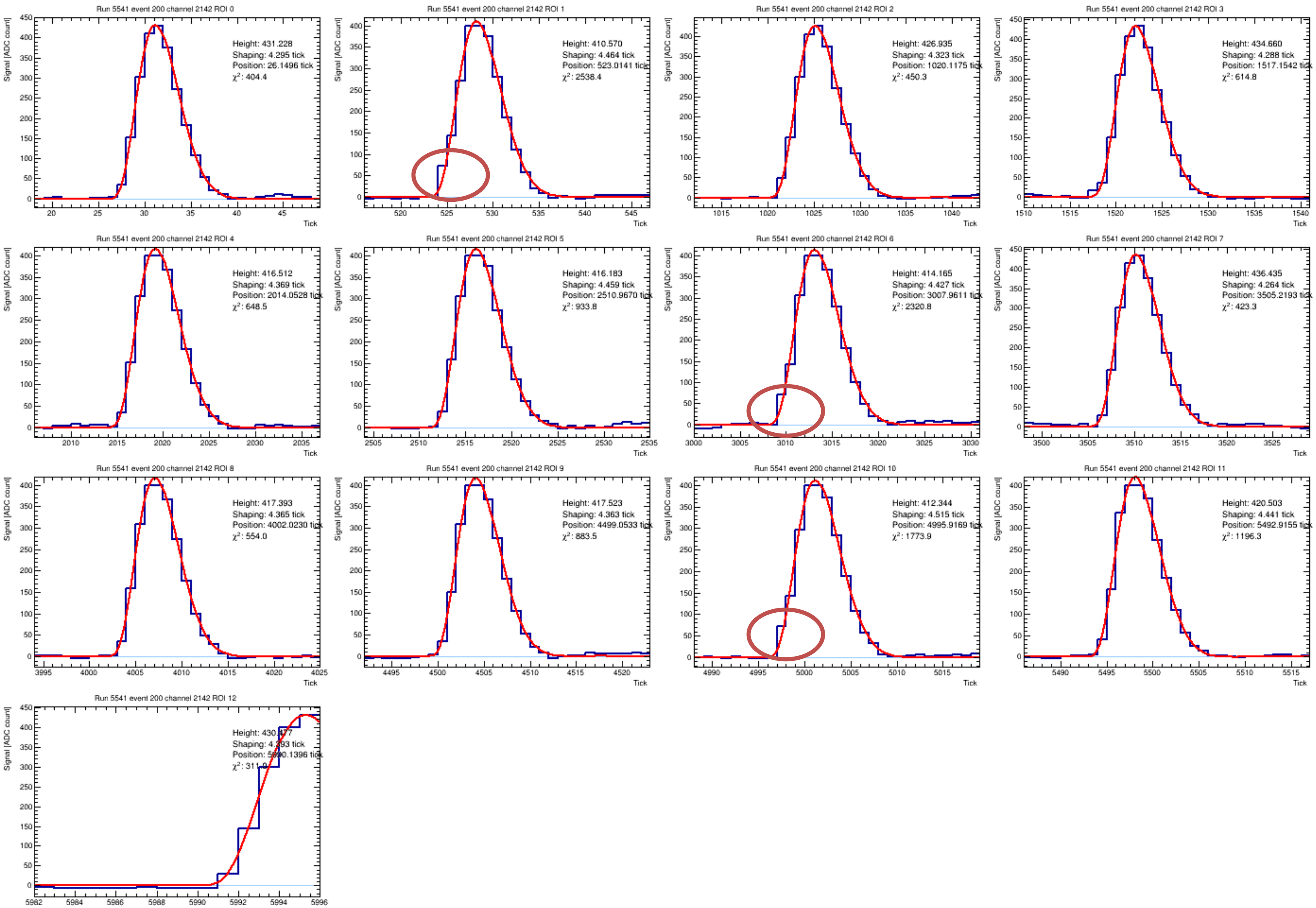
Sticky code in signal region causes the peak splitting

Sticky code in pedestal region

New mitigation

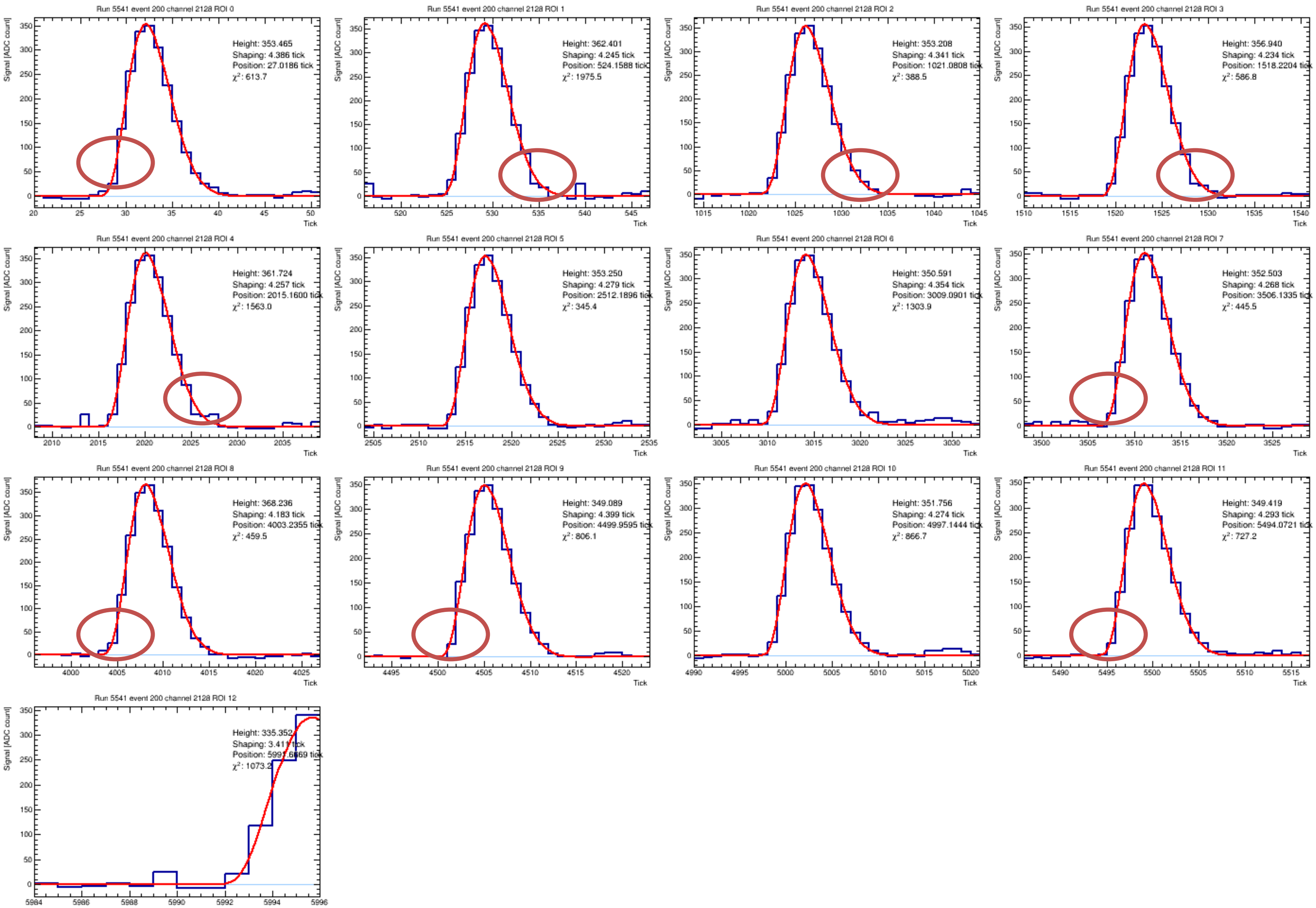


Old mitigation

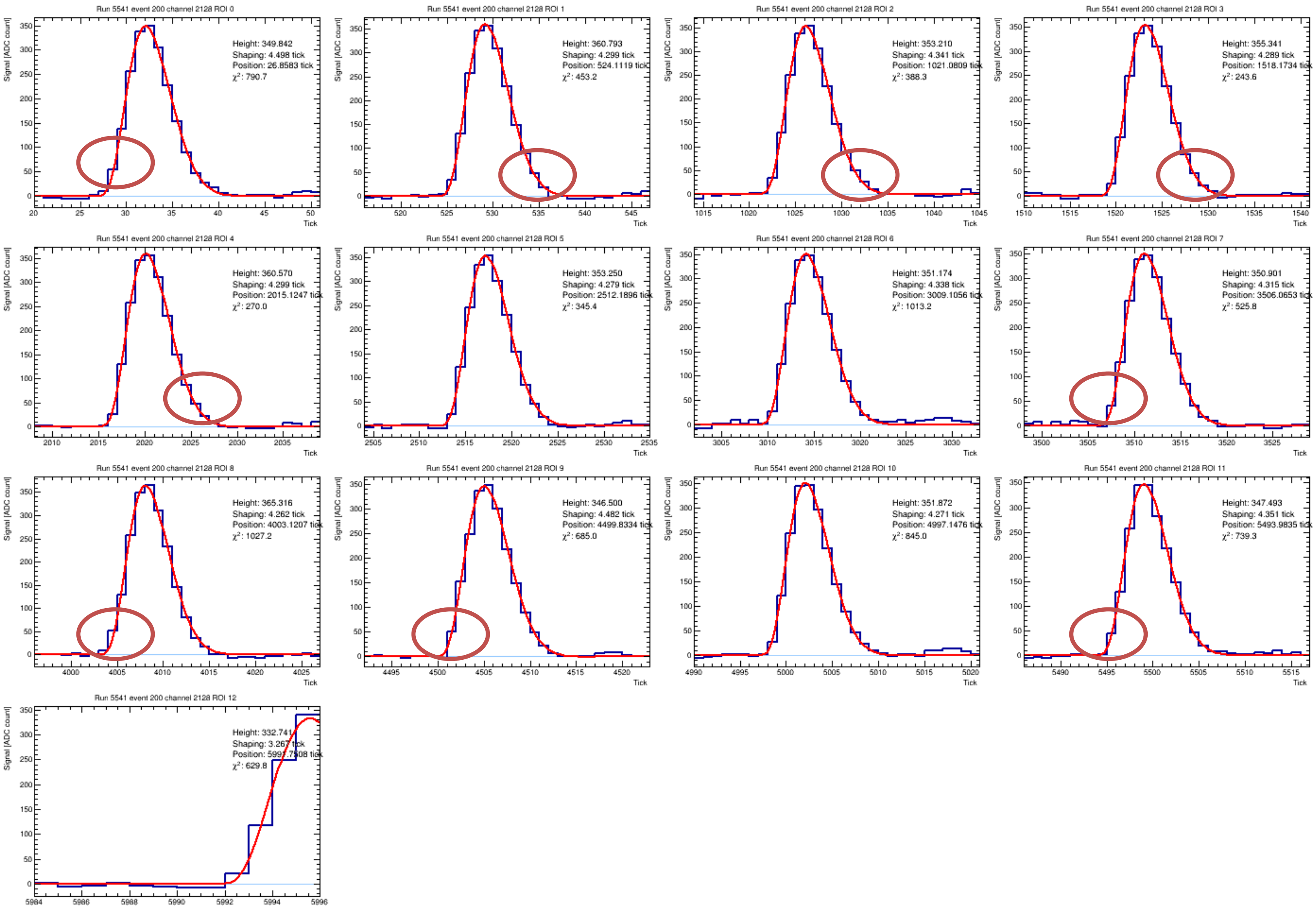


ROIs for channel 2128

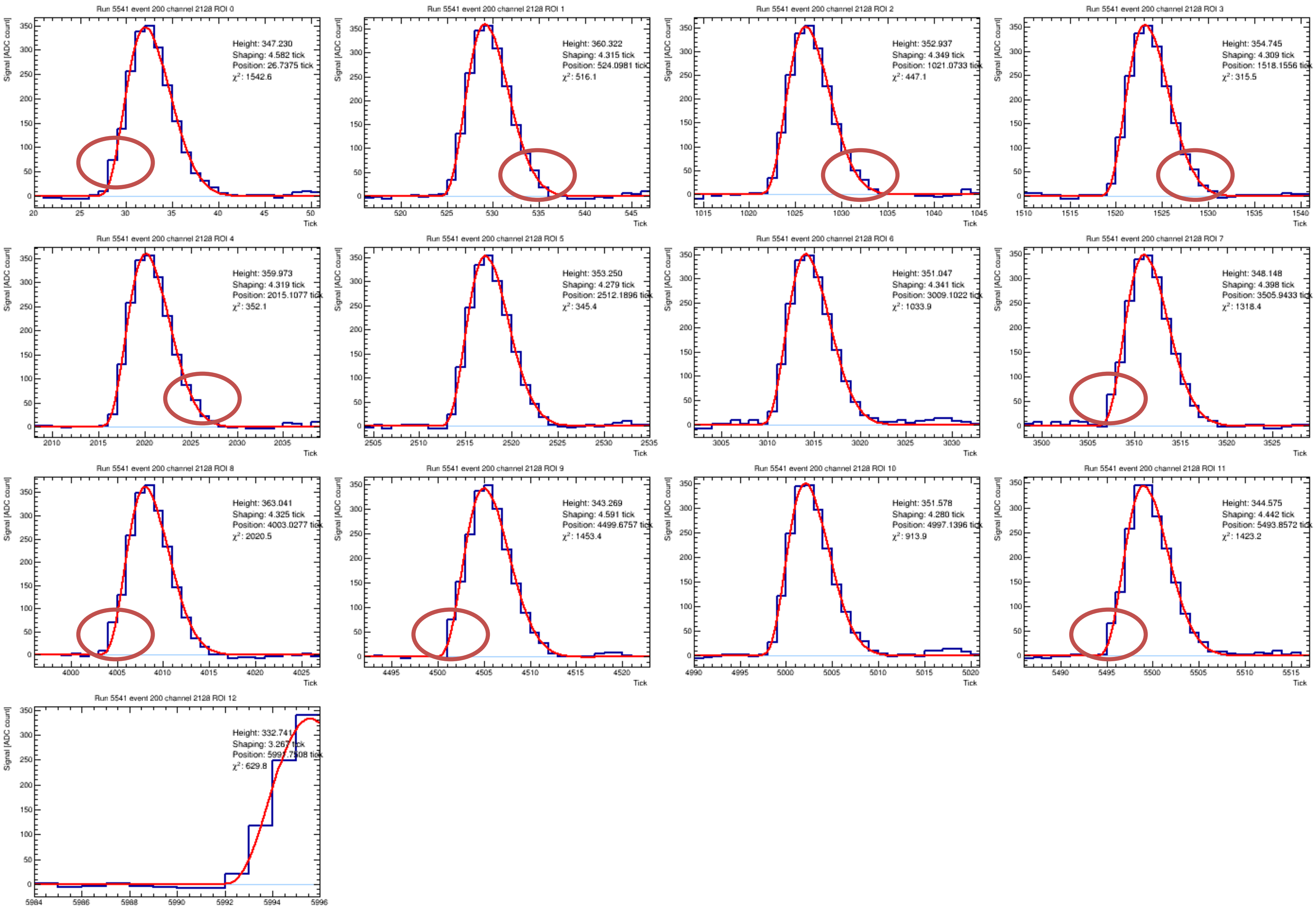
Unmitigated



New mitigation

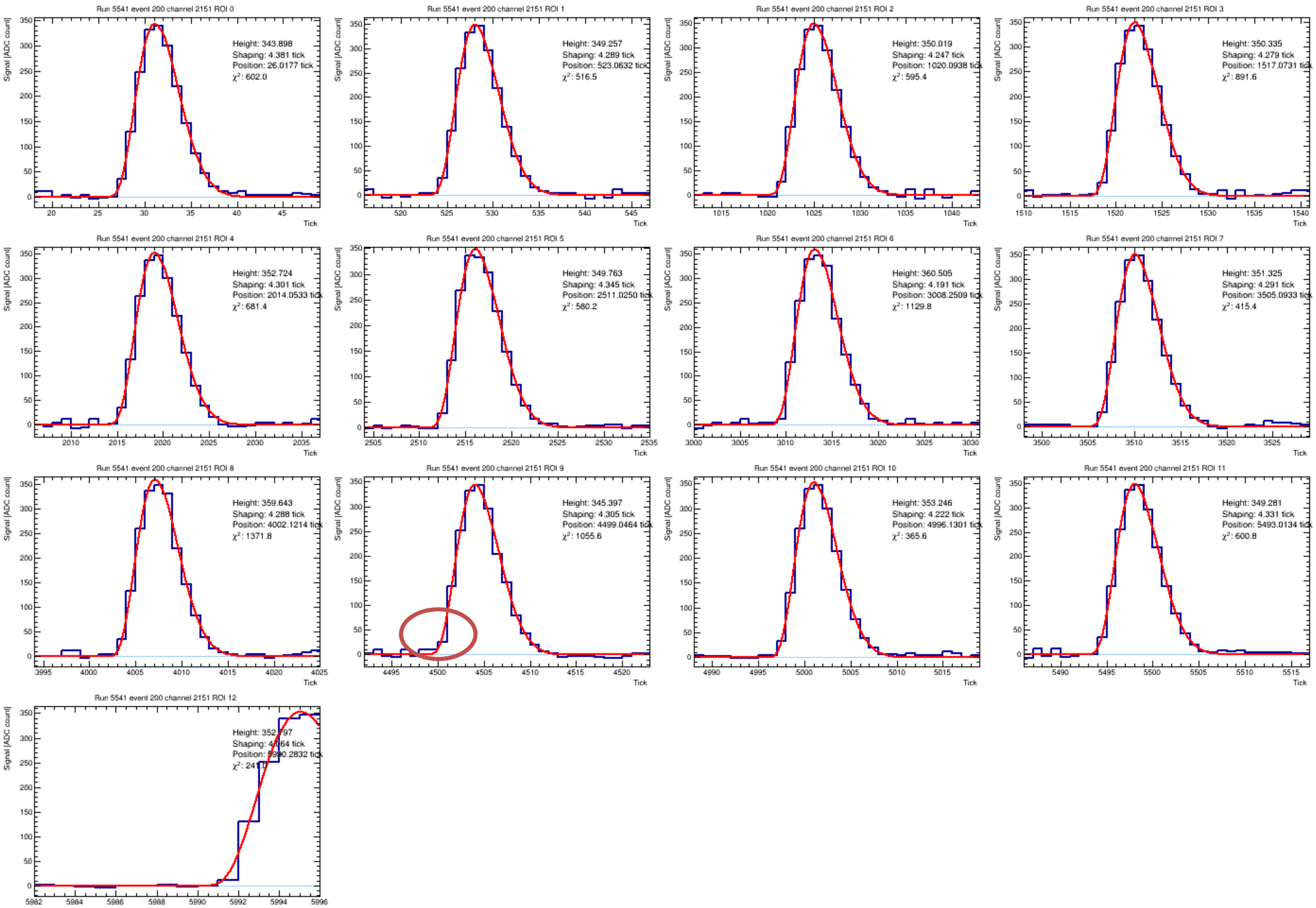


Old mitigation

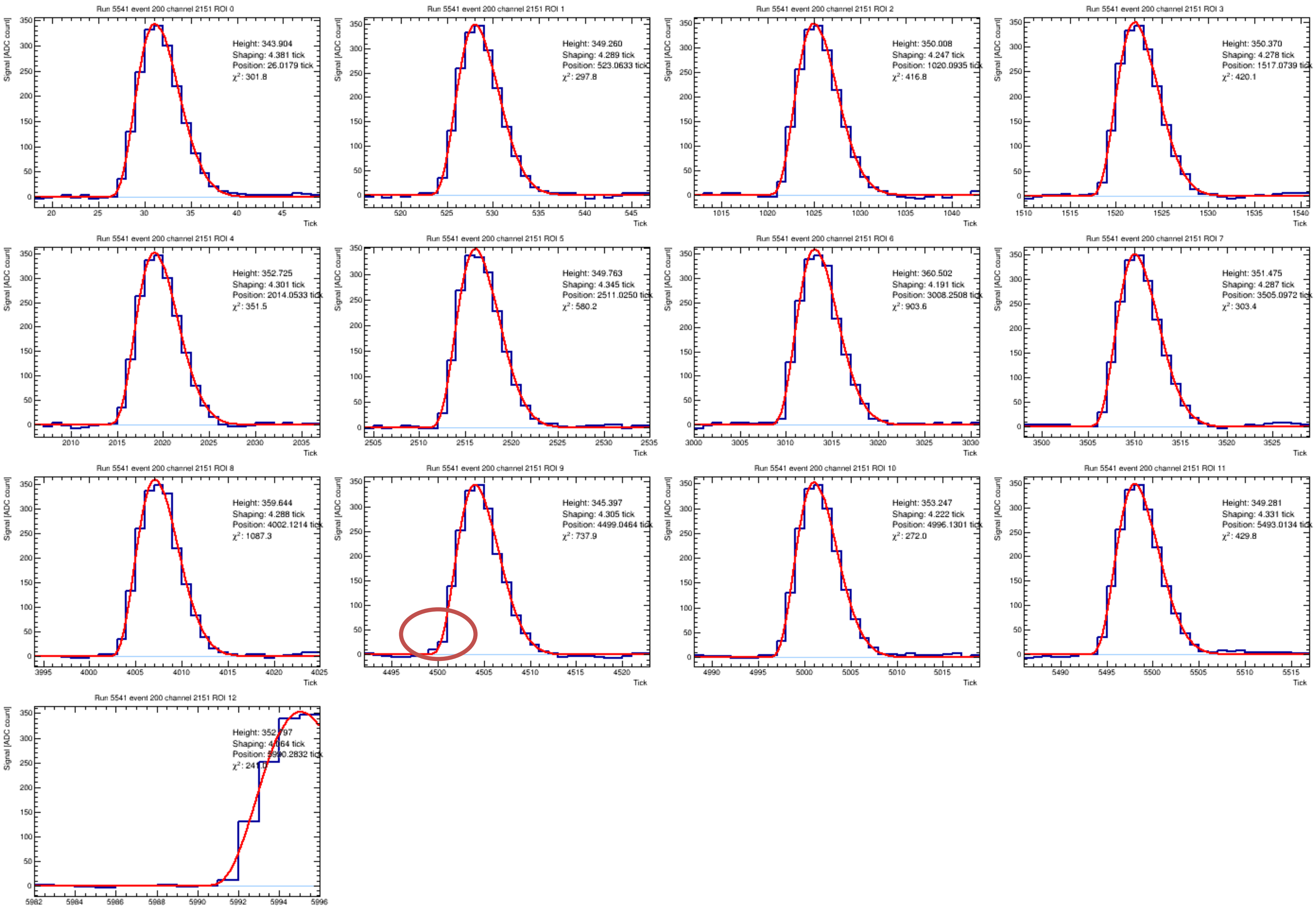


ROIs for channels 2151

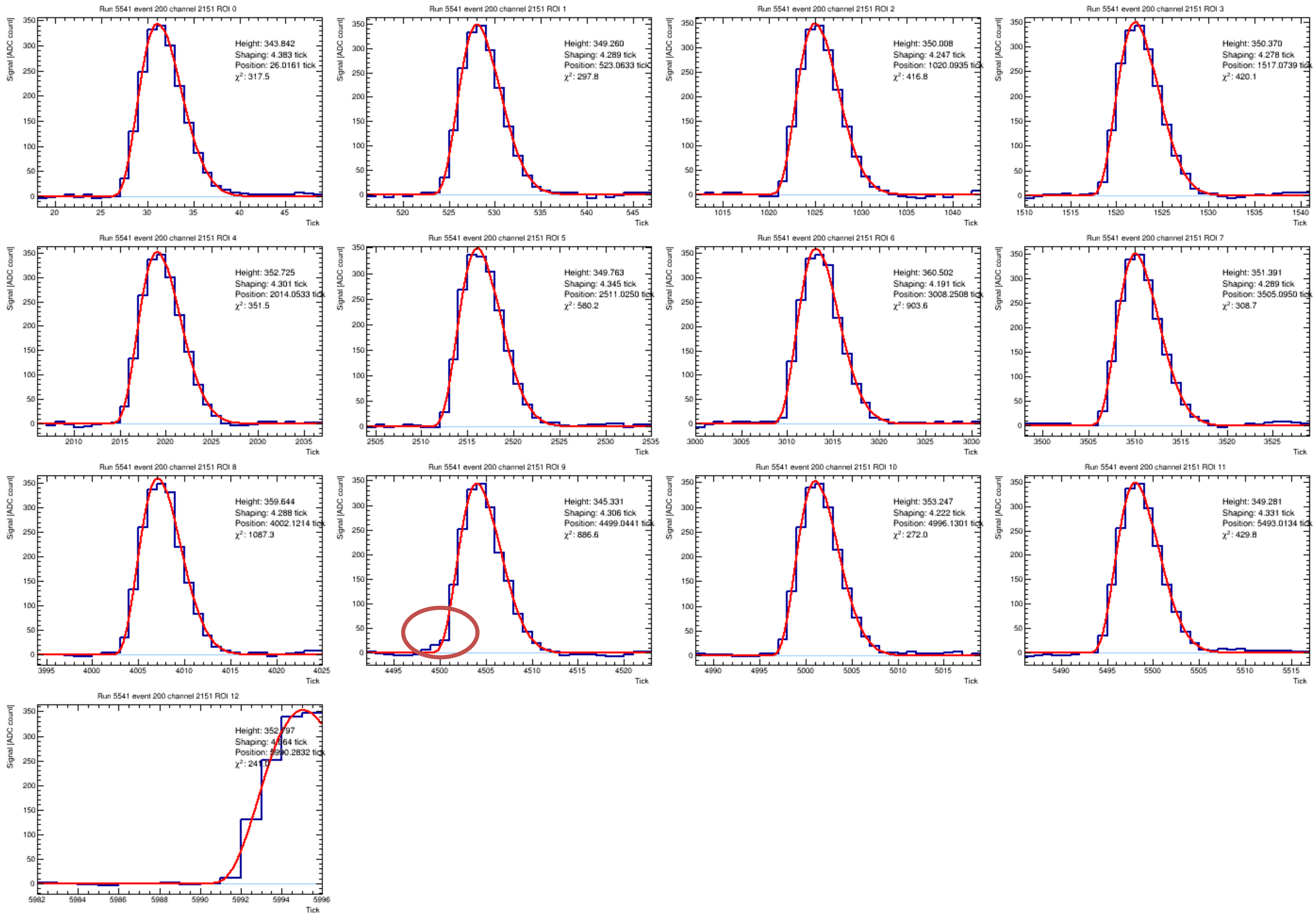
Unmitigated



New mitigation



Old mitigation



Comments

Sticky code mitigation

- New interpolation algorithm has been introduced
- Still interpolates between the two nearest good codes
- Uses fixed curvature instead of linear (curvature = 0)
- Curvature evaluated from the 2nd nearest good codes
  - Weighted to favor the side where sign changes least
- Change is in dunetpc and will go into next release if OK here

Pulser resolution study

- Shows sticky code mitigation improves resolution
  - Only looked at APA3x
  - Biggest effects in FEMB 302 where codes are sticky
- Better job with new interpolation algorithm
- Pulser data is useful for tuning and evaluating performance of steps in data preparation

Future

Future pulser resolution studies

- Check effect of mitigation on other APA planes
- Look at other dataprep steps
  - Timing mitigation
  - Tail correction
  - Deconvolution
- Try out other ideas for dataprep
  - Nice to have a quantitative measure of performance in signal region

Search for sticky codes in signal regions

- Use pulser tickmods to get better coverage
- Like also to have data with pulser timing offsets to improve coverage