GEP Tracking

Weizhi Xiong
Syracuse University
07/15/2020
SBS Collaboration meeting

Outline

- General information for GEP tracking
- GEM occupancy, noise rejection and clustering
- Tracking efficiency and accuracy
- Possible improvement and to-do list

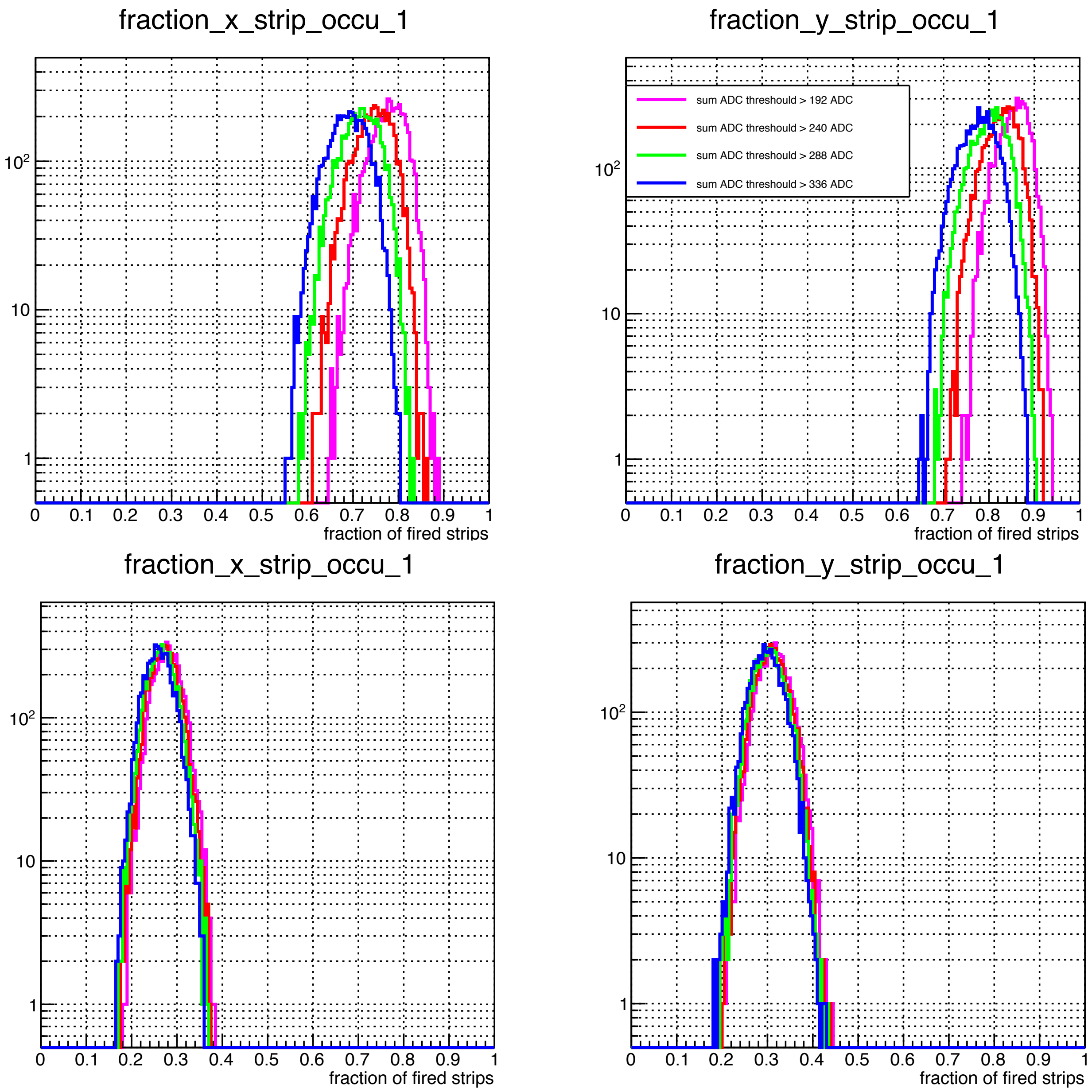
General info for GEP tracking

- At the moment, only doing tracking reconstruction for the GEP front GEM trackers (6 GEMs, each separated by about 9cm)
- Using 30cm hydrogen target
- Using electron arm info to constraint the proton track (see Prof. Andrew Puckett's slides for SBS Winter Colla. Meeting 2019)
  1. Divide target into multiple number of bins in *z*
  2. Scan these vertex-*z* bins, assume the interaction vertex is at the center of each bin and reconstruct the electron arm
  3. Using electron arm info (elastic kinematic) to constraint the proton track. Coordinates on front tracker ~+/- 1cm, slope ~ +/- 3mrad
  4. Record "good" candidates for each bin, do target reconstruction and perhaps kinematic fitting to determine the best pair
- Currently, only study the bin that contains the true MC vertex-z
- Using SBS GEP digitization provided by Eric

GEM Occupancy 100% bg

- Top row:
  1. Raw occupancy for x and y strips (longer ones)
  2. Total sum of 6 samples > 192, 240, 288, 336 ADC, corresponding to 4, 5, 6, and 7 sigma of pedestal noise

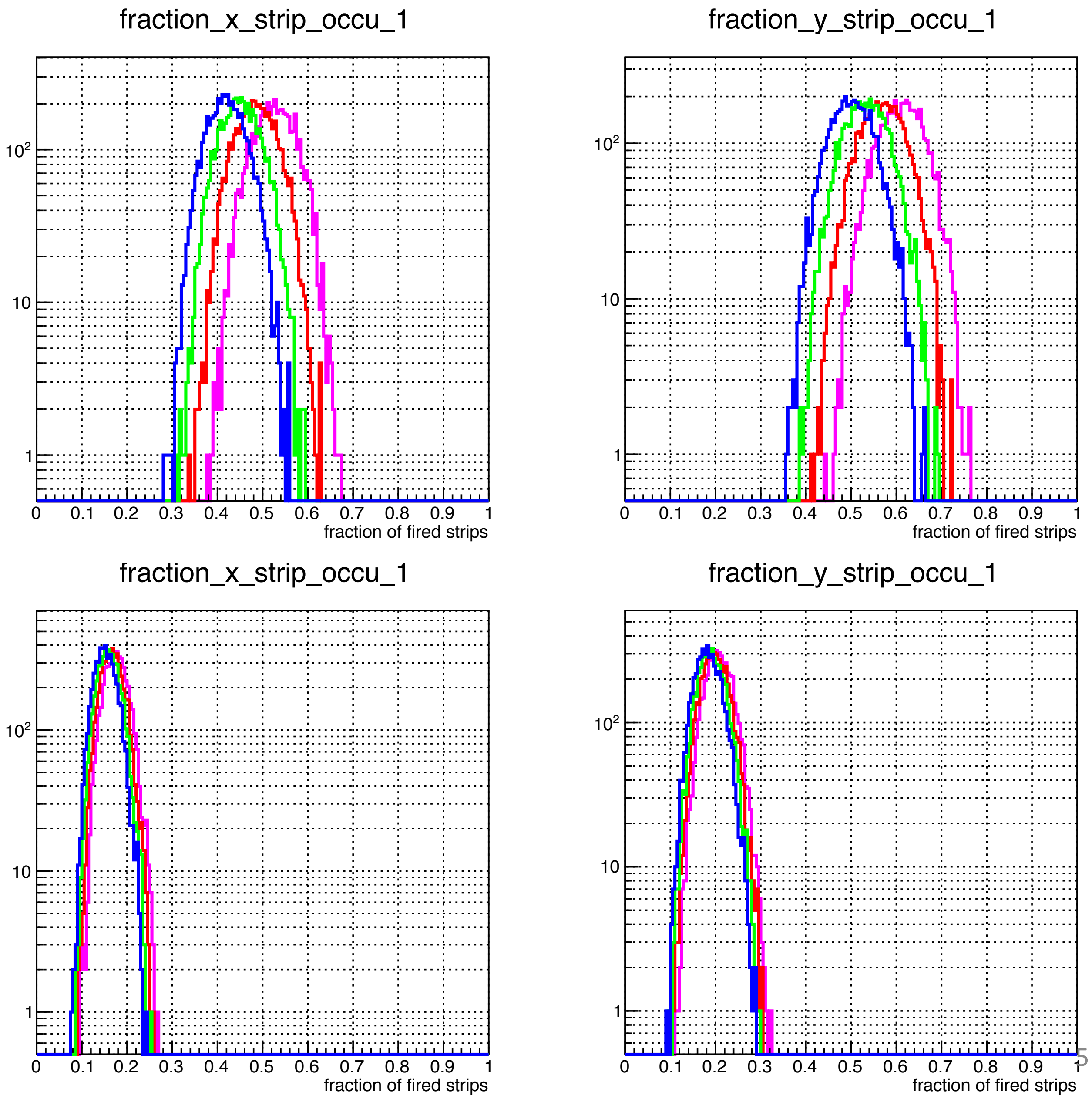
- Bottom row:
  - In addition to threshold cut, apply also the noise rejection cut (requiring raising edge of the signal)



GEM Occupancy 50% bg

- Top row:
  1. Raw occupancy for x and y strips (longer ones)
  2. Total sum of 6 samples > 192, 240, 288, 336 ADC, corresponding to 4, 5, 6, and 7 sigma of pedestal noise

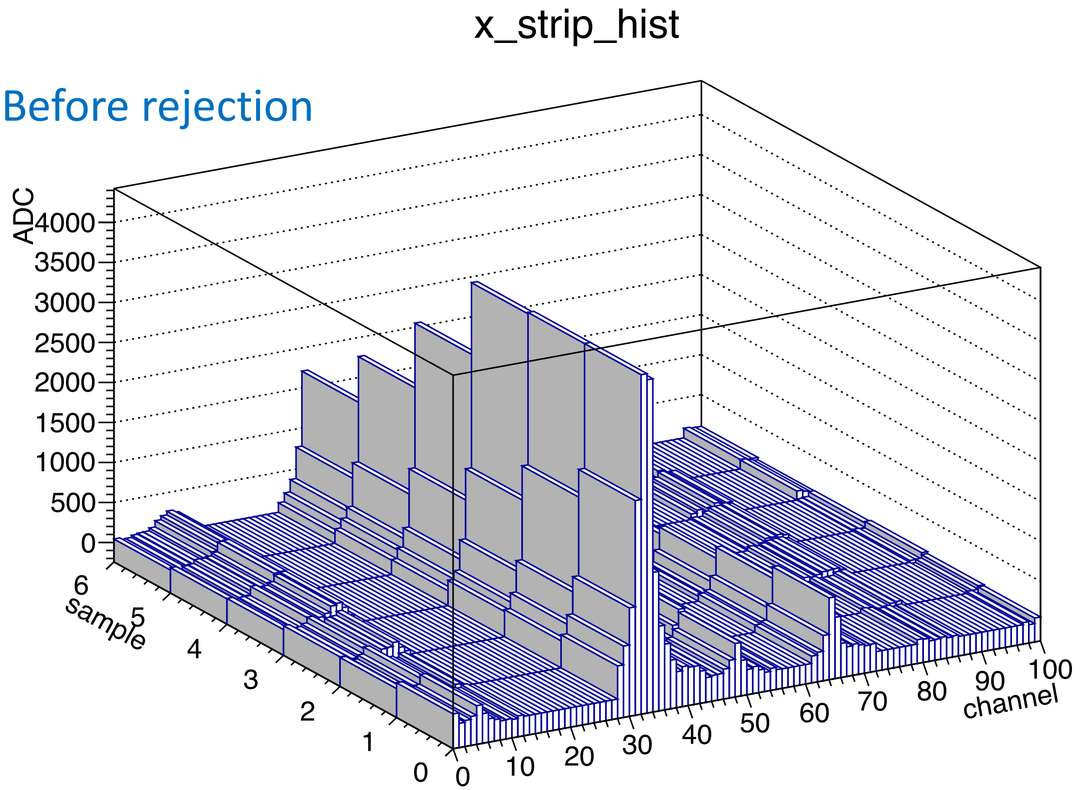
- Bottom row:
  - In addition to threshold cut, apply also the noise rejection cut (requiring raising edge of the signal)



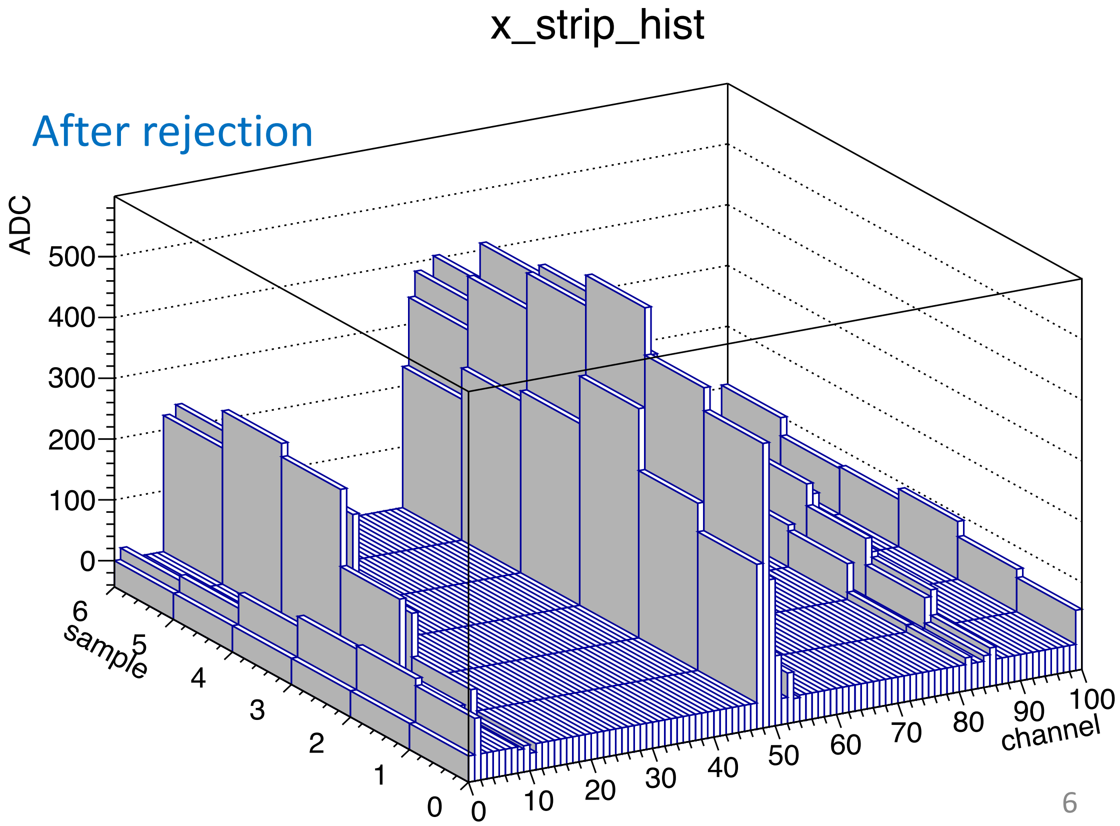
GEM Signal noise rejection

- Trigger latency setting is that the maximum of a on-time signal appear mostly at the 3rd sample
- cut the ratio of the first 3 samples: r1 = s1/s3, r2=s2/s3, accept if r1 < 1.3, r2 < 1.1 and r1 < r2
- Reasonably effective in rejecting out-of-time backgrounds
- Will cause GEM inefficiency at high occupancy (10% loss at 50% GEP bg), but reduce the total number of recon hits by a factor of ~5

Before rejection

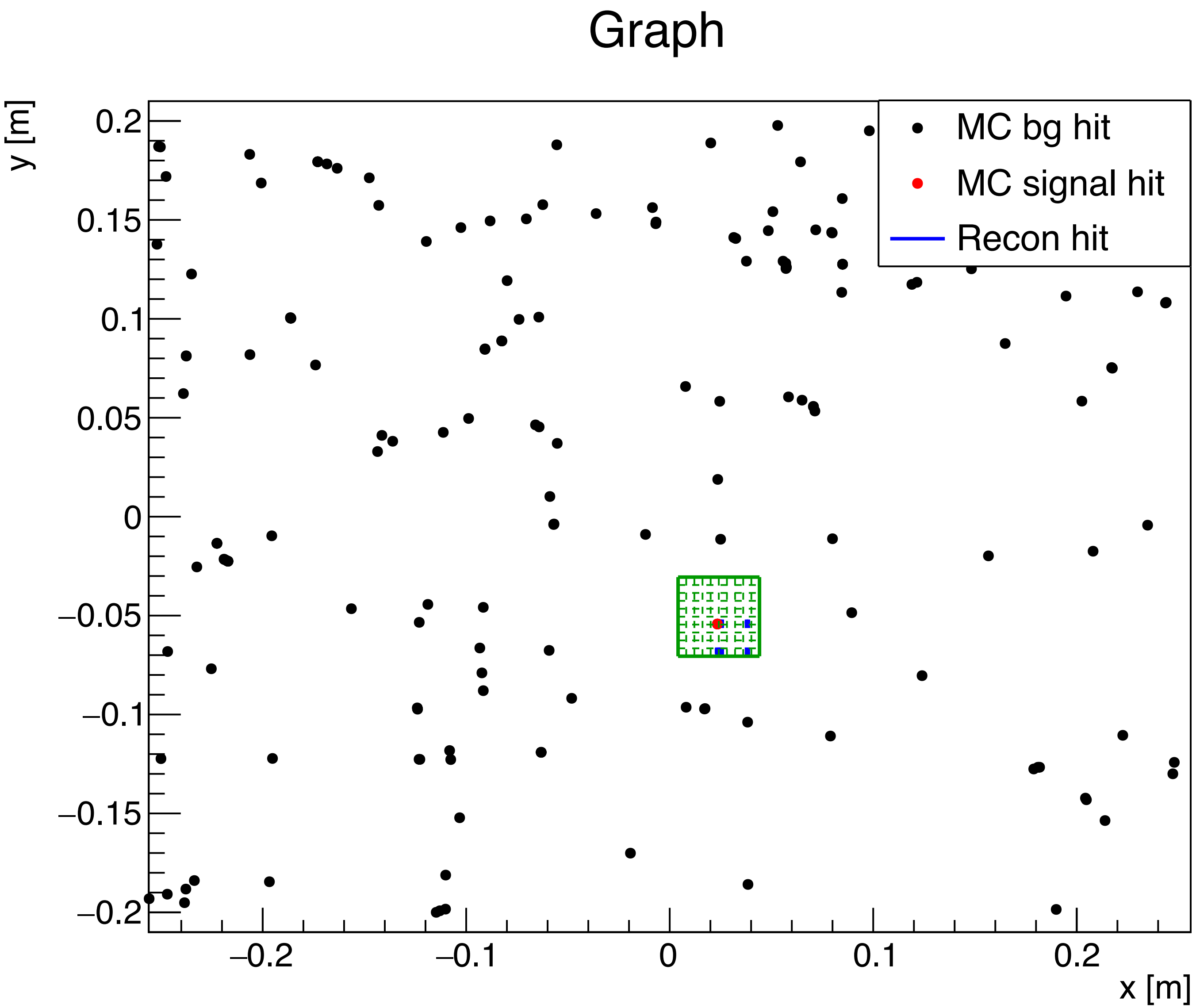


After rejection



GEM Clustering – search region

- Using electron arm to constraint the position of the proton track
- Open a +/- 2cm search region around the expected proton track (green box)
- GEM clustering and reconstruction is only performed for strips inside this region
- Drastically improves the execution speed for GEM clustering algorithm

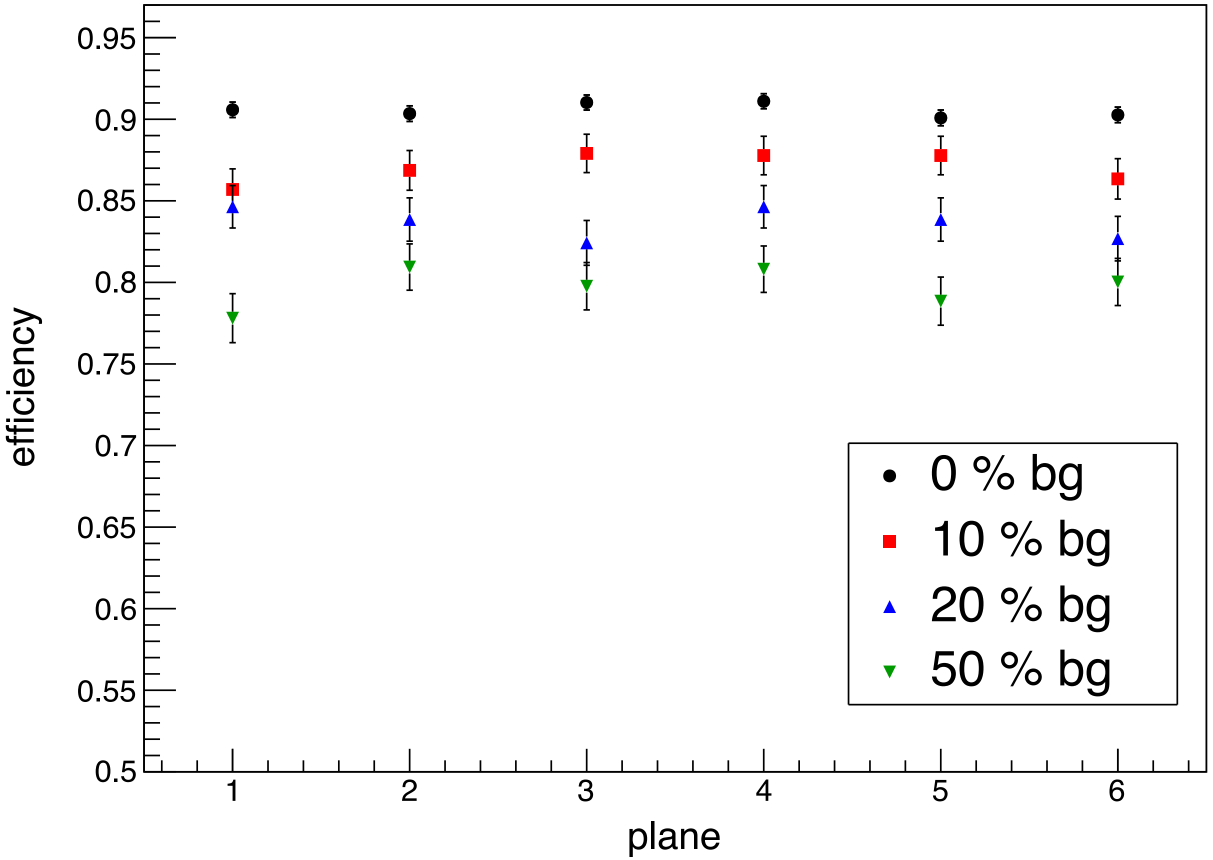


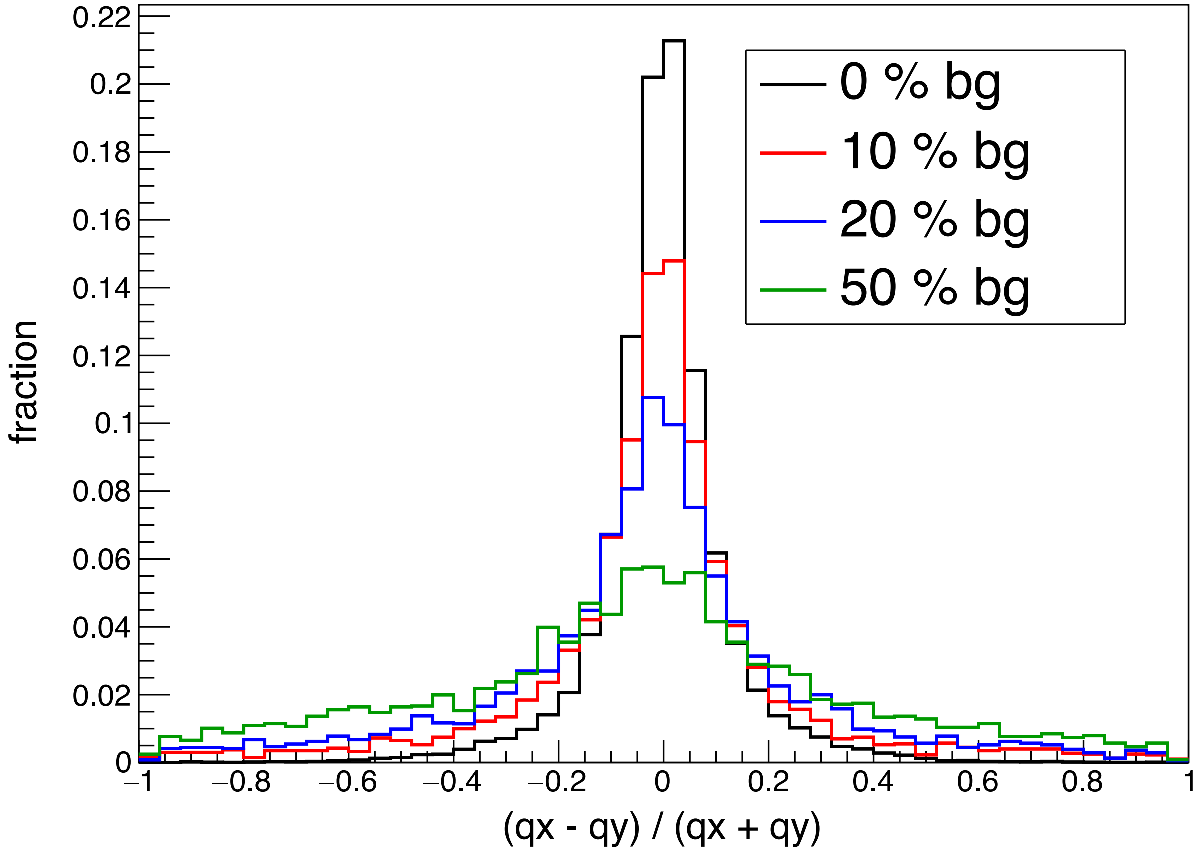
GEM Clustering – algorithm

- Based on Prof. Andrew Puckett’s standalone tracking code
- Seeding a GEM cluster for each local maxima of ADCx * ADCy
- Grow the cluster by attaching consecutive fired strips
- Stop growing once we reach a local minimum of ADC or maximum allowed cluster size
- Split the ADC at minimum by half

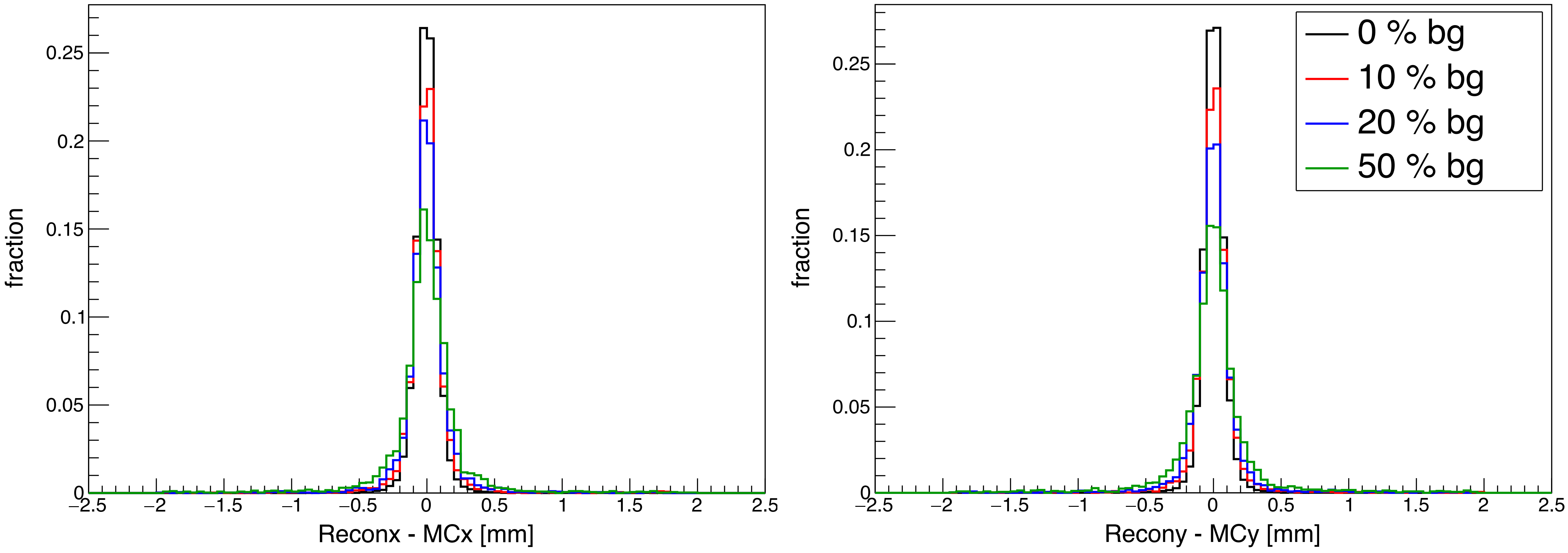
GEM reconstruction – efficiency and charge asymmetry

- Applied 5 sigma zero-suppression and noise rejection cuts
- A GEM is efficient if there is a reconstructed hit fall within +/- 5 strips around the MC true signal hit, for both x and y strips





GEM reconstruction – position resolution

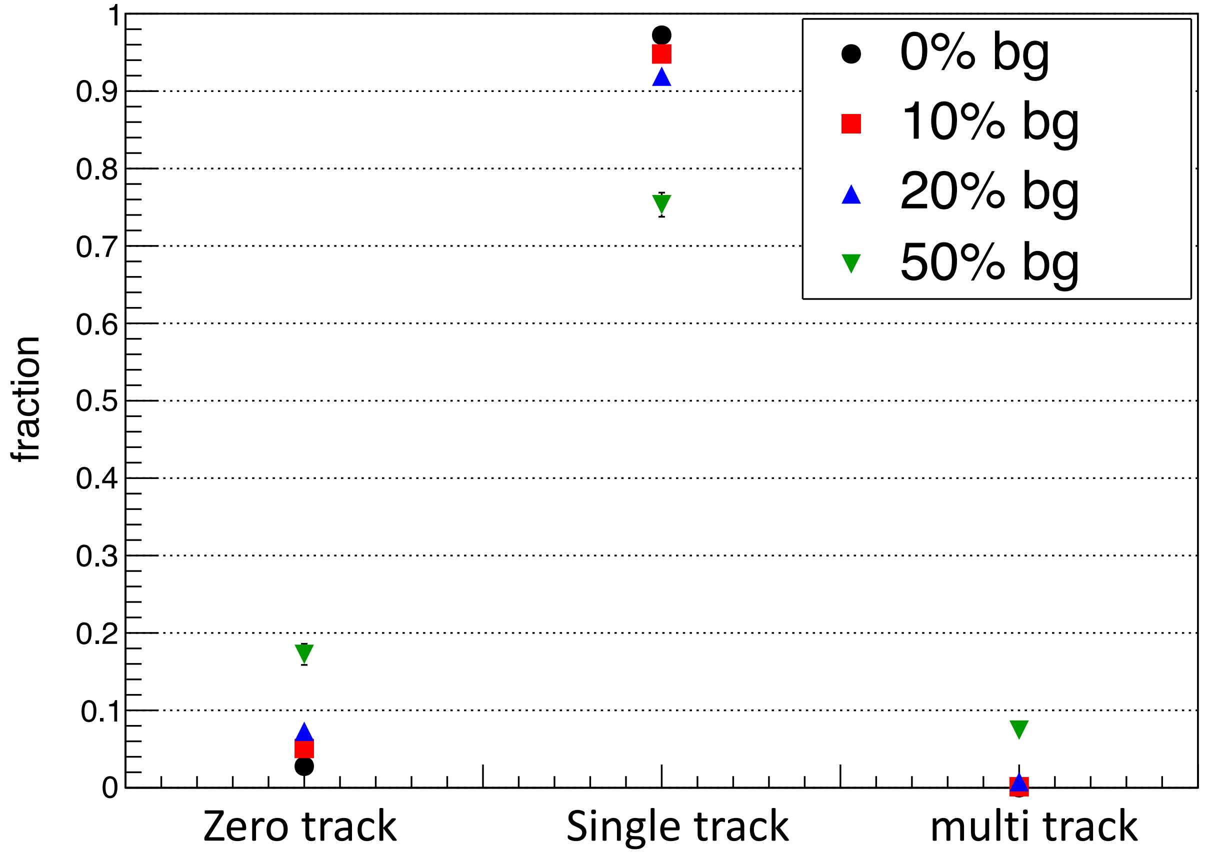


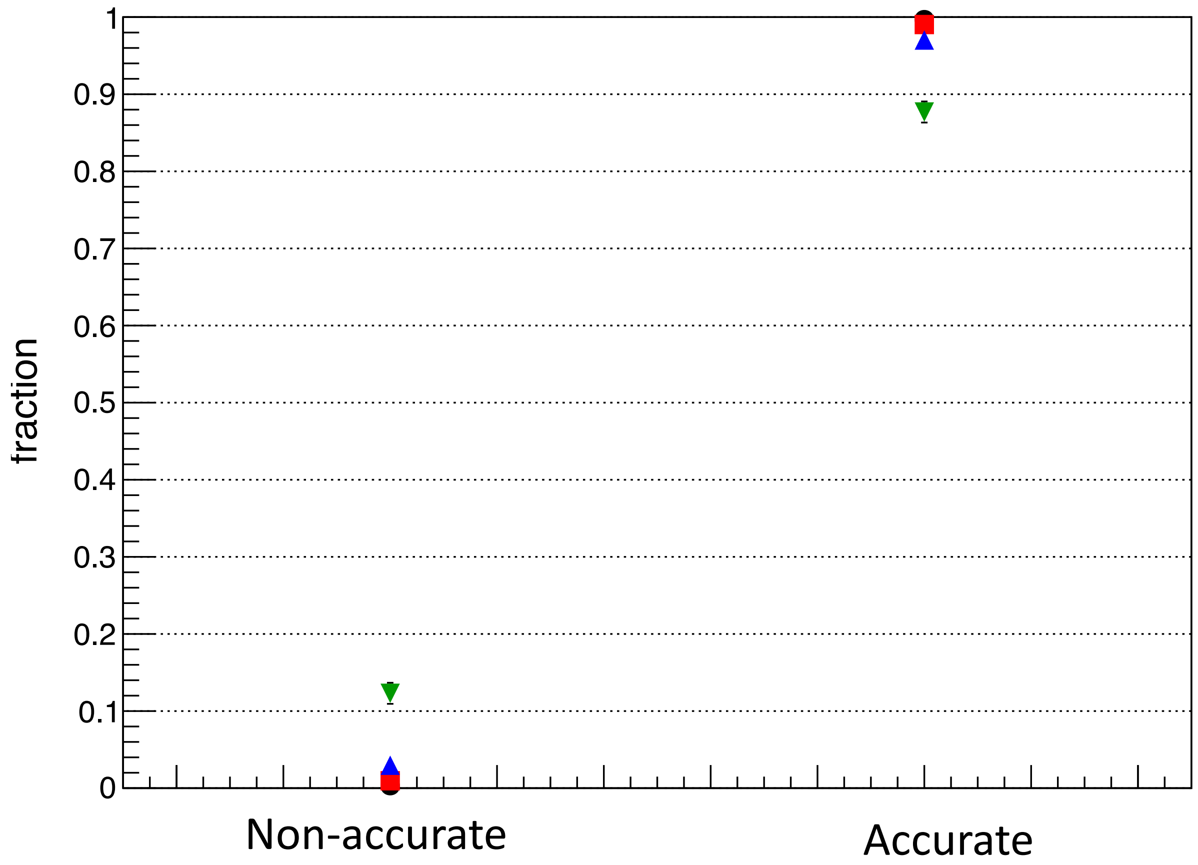
Tracking reconstruction -- algorithm

- Based on Prof. Andrew Puckett’s standalone tracking code
- Use “Kalman Filter” like approach:
  1. Loop through all combinations of 2D GEM hits on all GEM trackers
  2. For the selected hits, fit them using linear least square method and update track parameters
  3. Use the updated track parameters to project the next hit on the next GEM, adding the next hit if it fall within certain range around the projected hit
- Requirement:
  1. At least 4 out of 6 GEM have hits
  2. Track slope agree within +/- 5mrad around the projection from electron arm
- Additional Requirement (for multi-track events):
  1. At least 5 out of 6 GEM have hits
  2. Chi2 / ndf < 10
  3. At least 3 hits have good charge asymmetry $|(qx - qy) / (qx + qy)| < 0.5$
  4. If still more than one track satisfy these, use the one that has more hits

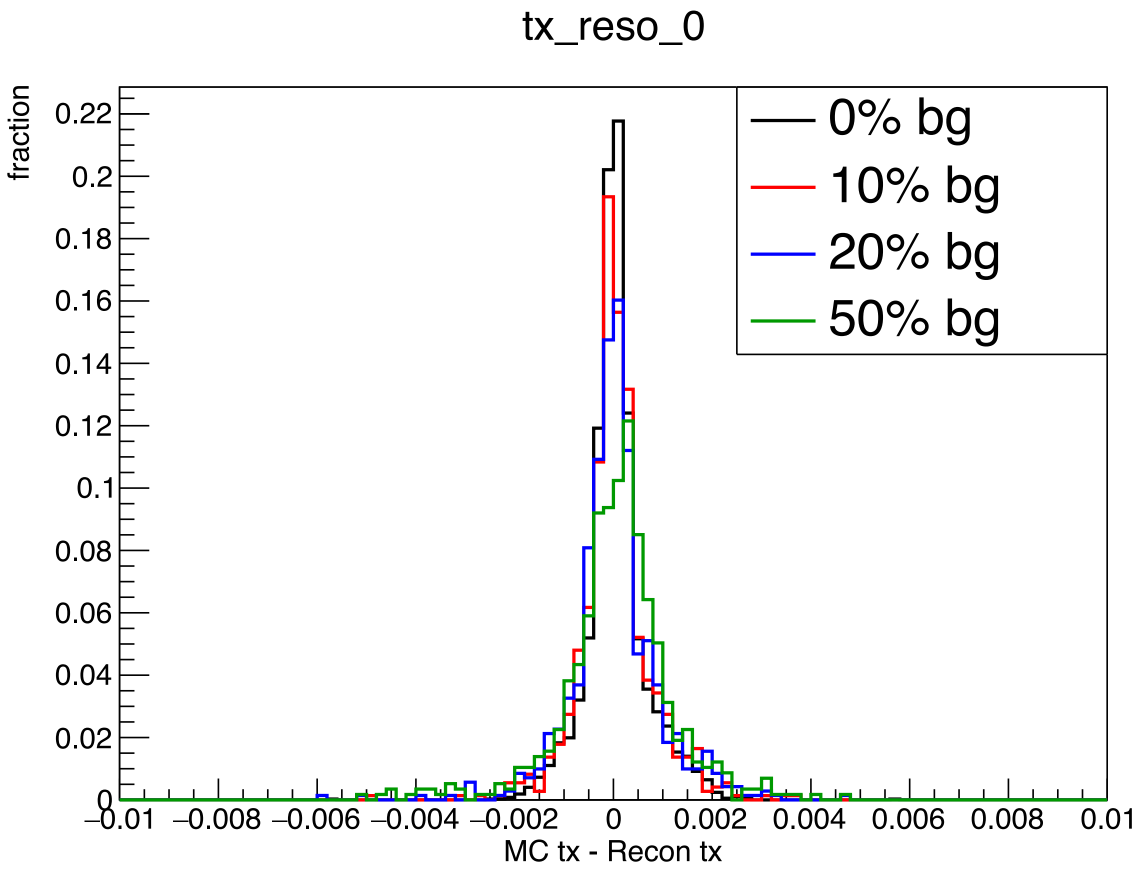
Tracking reconstruction -- algorithm

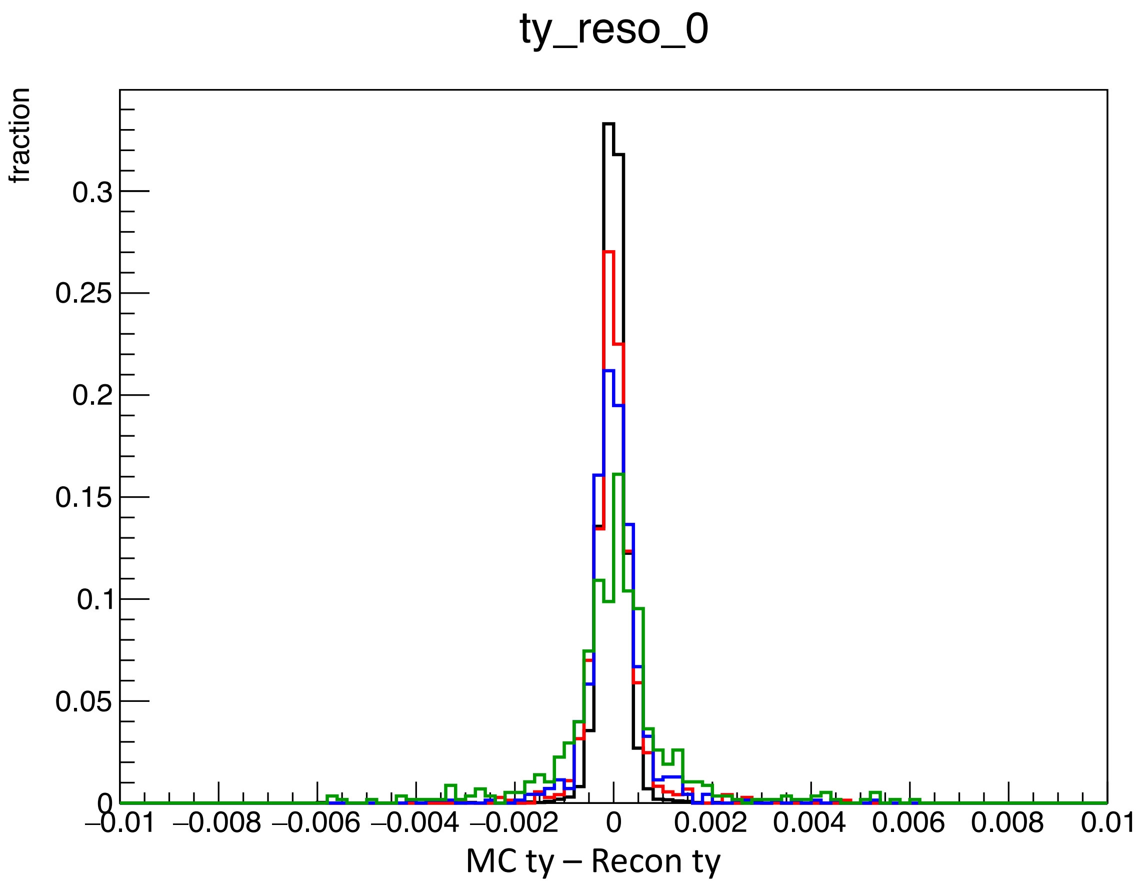
- If multi-track events has a track satisfy the “additional requirement”, it is counted as single-track event
- Accuracy is calculated for single track events only
- A track is considered accurate if it is consist of all “good” hits
- “good” hit: the closest recon hit to the true MC hit, and it must fall within +/- 5 strips around the true MC hit
- Results with 50% bg suffers from too many hit combinations (> 1e9), ~10% loss in efficiency due to this





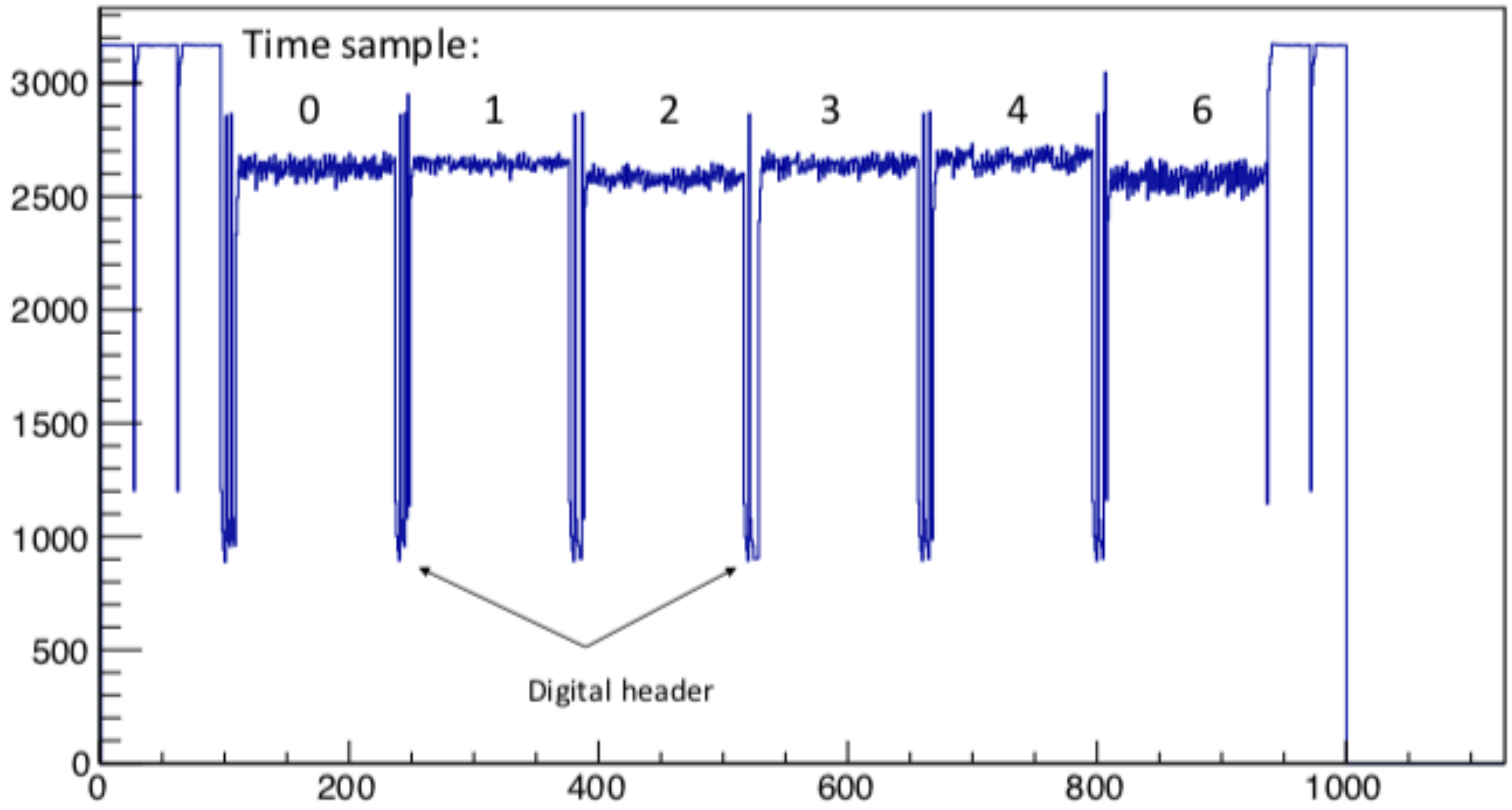
Tracking reconstruction – recon track slopes





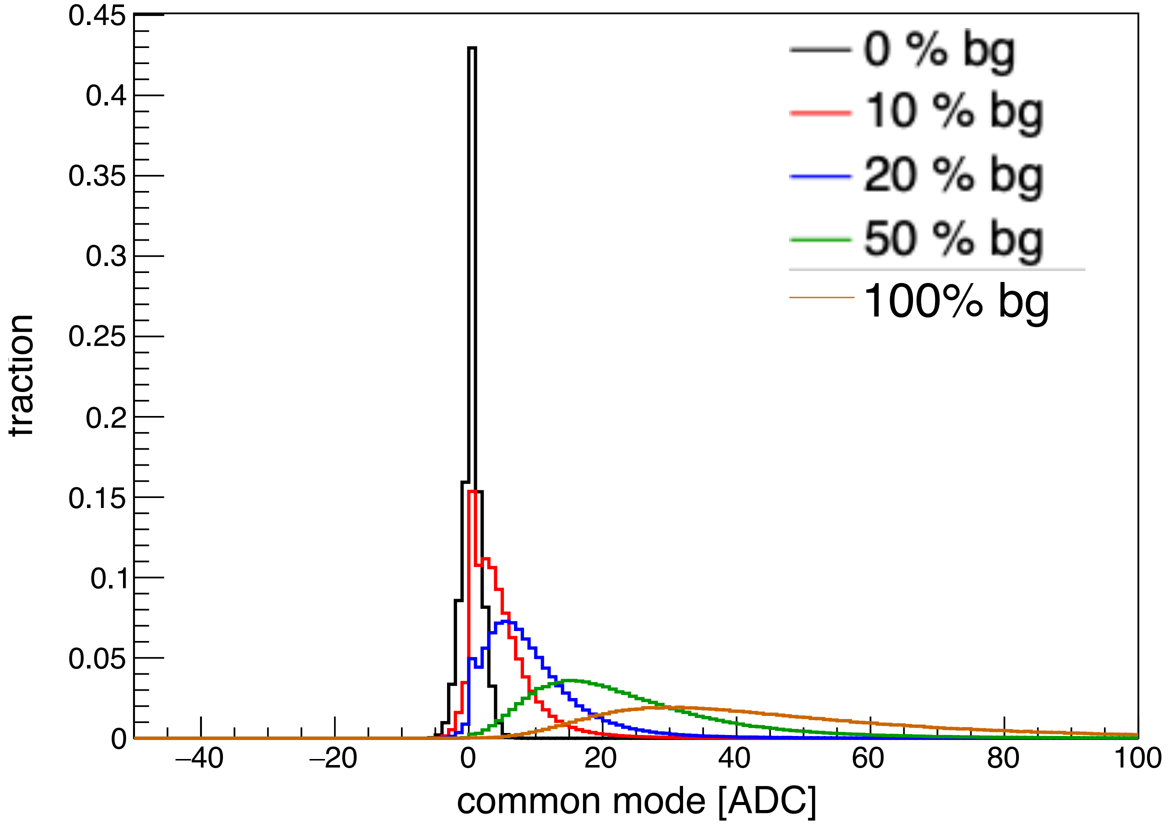
Common Mode

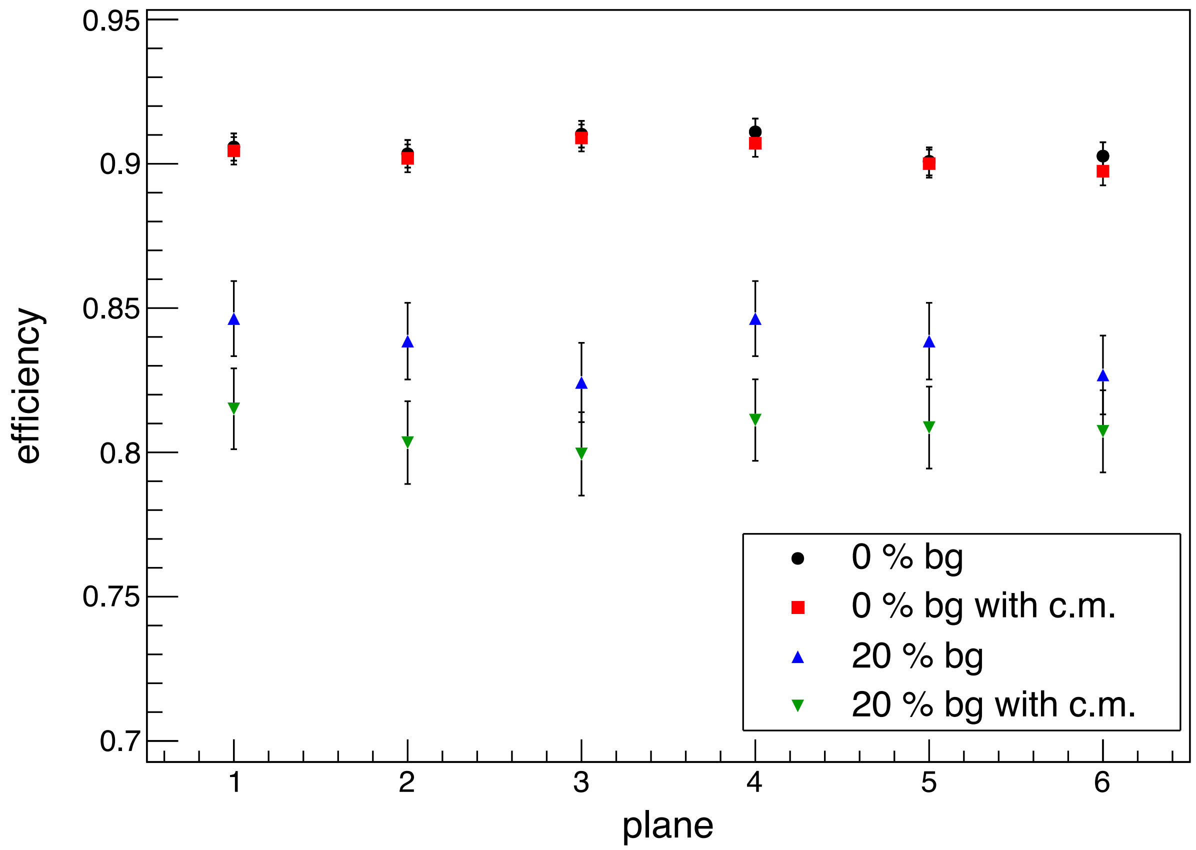
- The following plot shows the raw pedestal data from one APV25. There are 6 consecutive time samples, separated by digital headers
- For each time-sample, one can see the pedestal noises, which are small fluctuations around the baseline
- One can also see the baselines shift between different samples, these baselines are the common modes



Estimated common mode

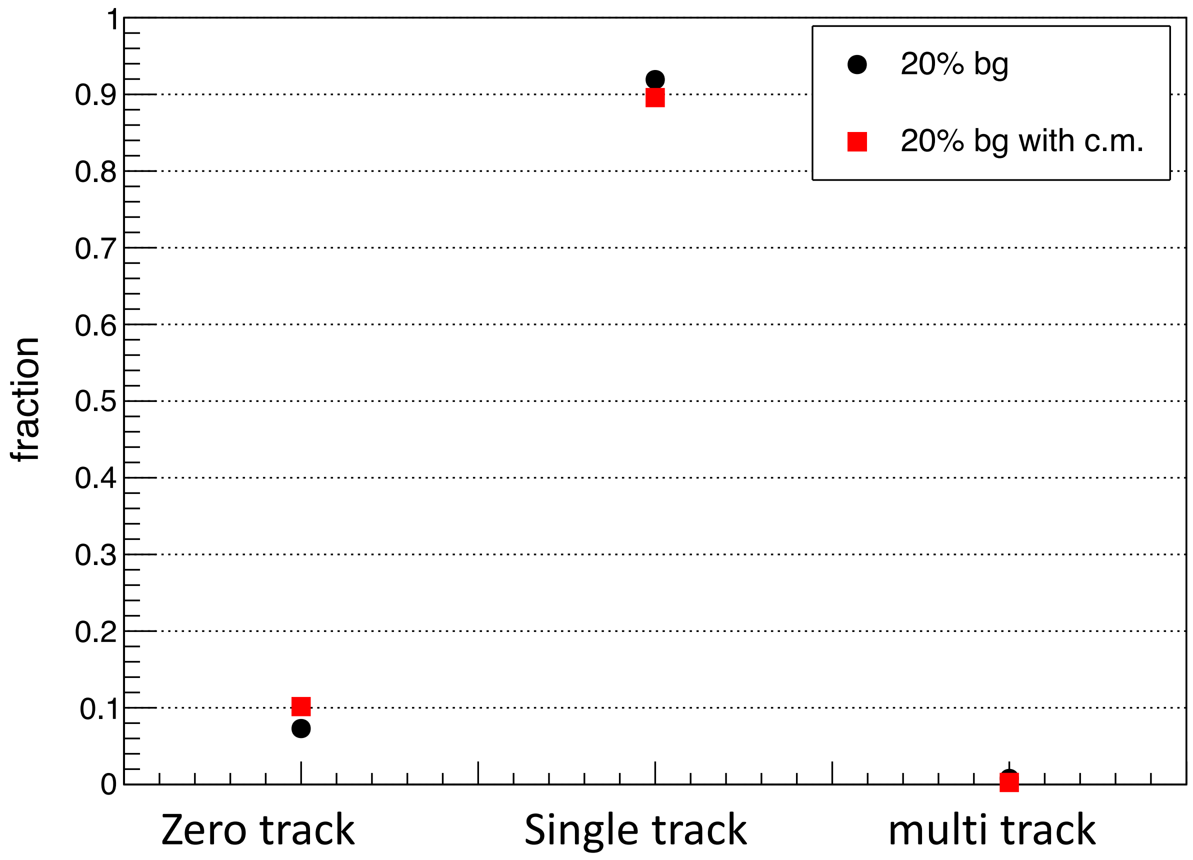
- True common in the simulation always at 0
- Applied Danning's algorithm to obtain estimated common mode values
- At high occupancy, estimated values tend to be larger than the true value as expected
- It reduces the GEM efficiency as we more likely will subtract more than we suppose to

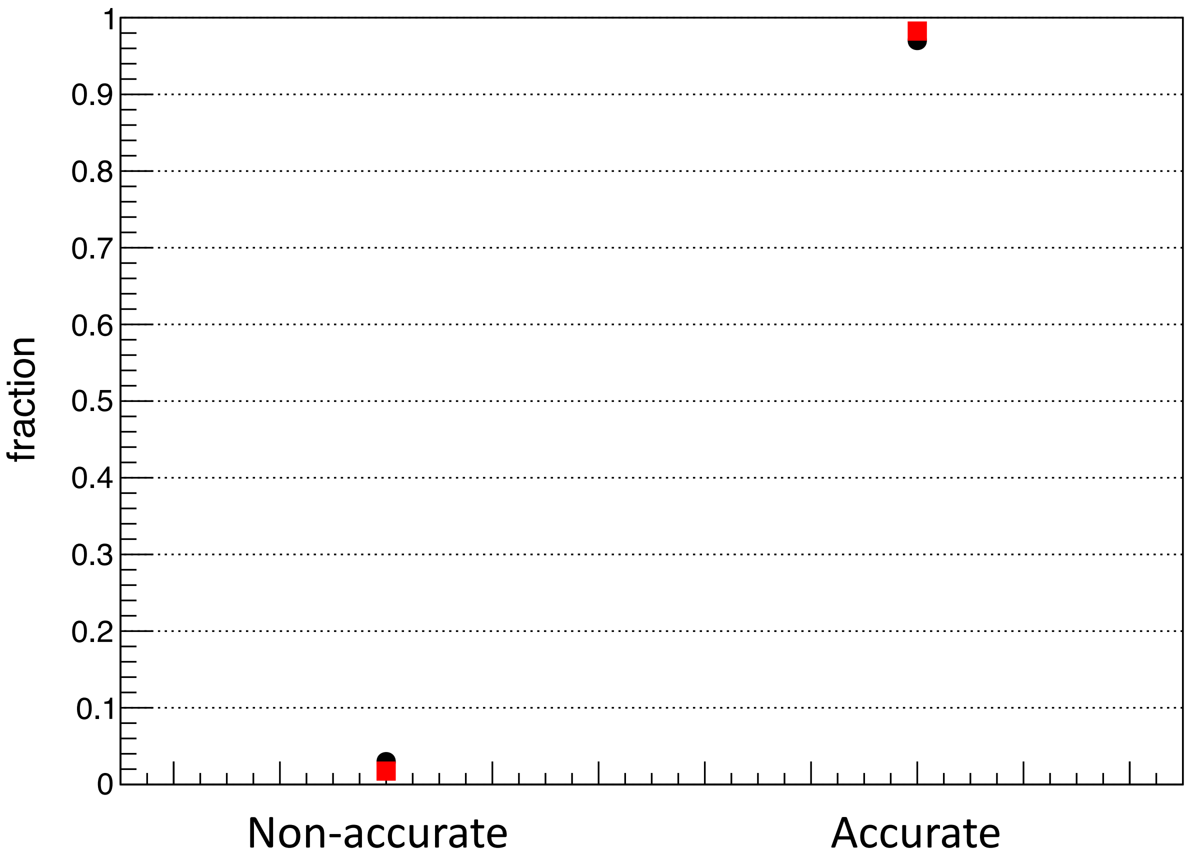




Effect due to common mode

- Tested up to 20% bg at the moment
- Difference is less at lower bg level





Possible Improvement and to-do list

1. Optimizing cuts – may have percentage level improvement on the results
2. Deconvoluting the GEM signal
   a) Reduce pile-up effect but increase pedestal noise level
   b) Improve GEM efficiency in high bg cases, but may introduce more bg/false hits
3. Better clustering algorithm and splitting
   - Maybe some ML algorithm?
4. Improve tracking algorithm speed – right now it takes about 8 hours to process 760 events at 50%
5. …