CVT Status

V. Ziegler

M. Defurne

March 2020 collaboration meeting

CVT Tracking Status

- Package developed as joint effort between JLAB and Saclay.
  - Saclay → implementation of CA pattern recognition
- Second/Exploratory package branching off main branch to study/develop alignment of the CVT and possible seeding/fitting improvements
- Review of the two packages
  - Limitations of current code: momentum dependence as a function of phi for different regions of PS. Alignment missing. Some biases.
  - Features of current codes identified during the review addressed.
- Current effort to merge improvements from second package into the first.

Addressing the Recommendations

Recommendation 1

- Correct the geometry in the old code, to read from the geometry service (so that it
- has the same geometry as the simulations). Note that the new code already does this.

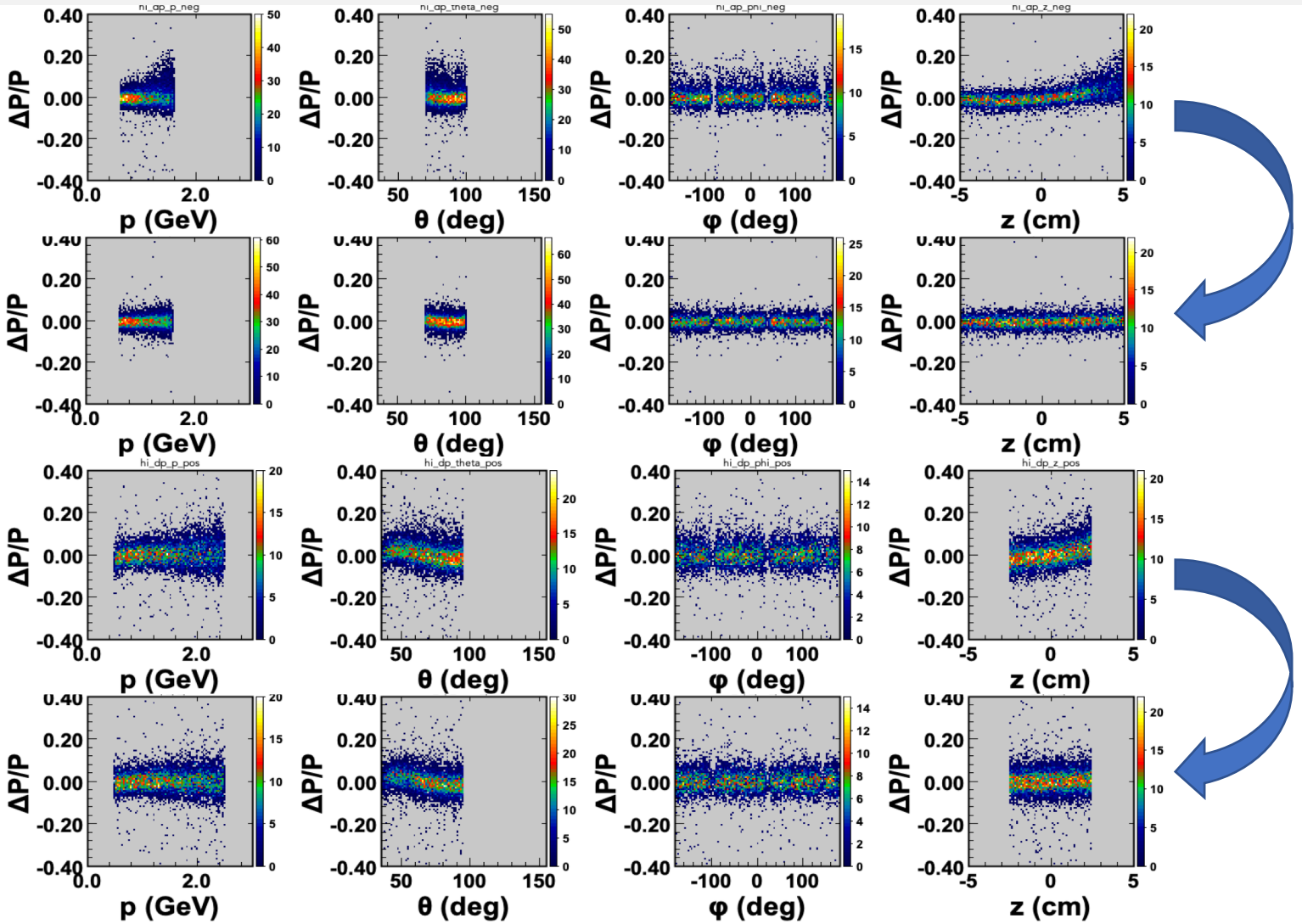
Implementation of recommendation 1

- The geometry package has the capability of doing translations and rotations of the SVT modules and to import survey numbers. The results using this package versus using the hardcoded geometry has been validated in MC.
- The code implements rotations and translations using alignment numbers put in CCDB for SVT only at this stage.
- The code now reads all constants from CCDB for both SVT and BMT systems.

Z-dependence of momentum resolution resolved

- Simulated Muons
- Simulated Protons

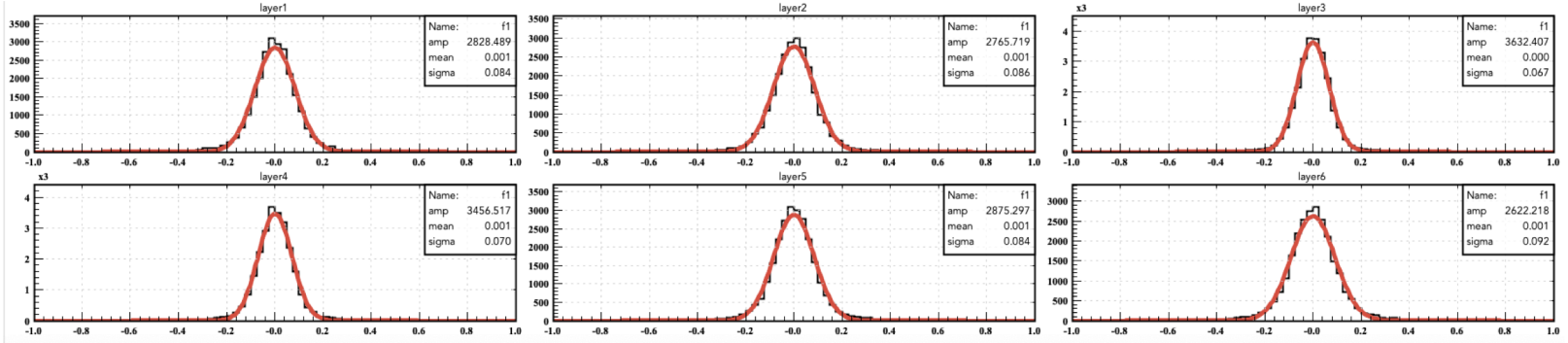
Remaining bias in theta needs to be understood



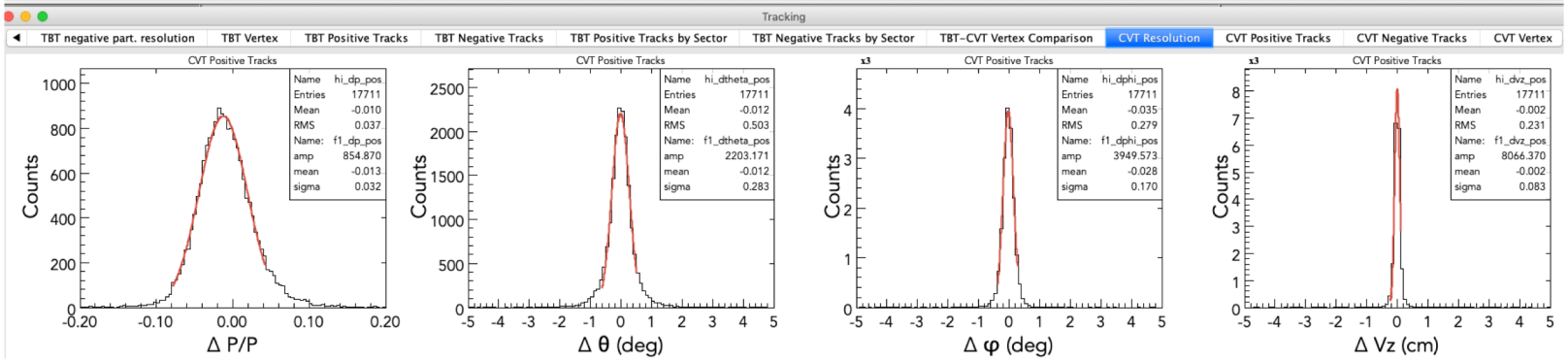
A bug was found in the calculation of the space residuals from the fit. The code produces a trajectory from the track parameters and reports points at each surface. For the SVT these are planes which are defined from a direction and a reference point. The reference point was not at the correct radius, which produced a parallax effect in the residuals. This has been fixed.

- Simulated Protons @ Vtx = (0,0,+/-5) cm

Spacial residuals



Resolutions



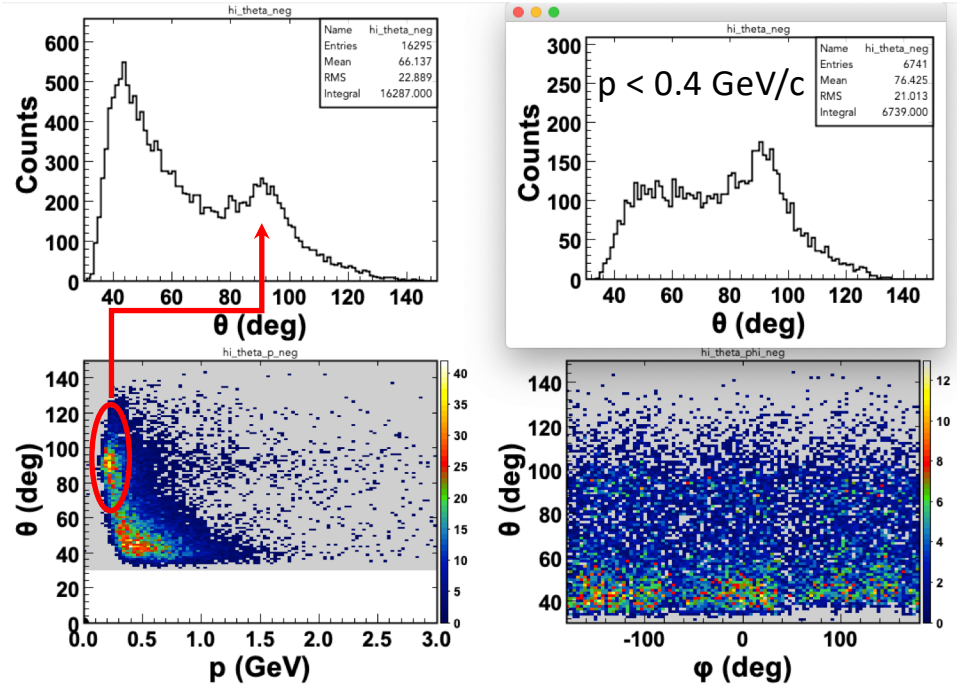
- Residuals still larger than expected. Next step: implement Maxime's algorithm for propagation to module surfaces.

Recommendation 2

- In the old code, the Kalman filter returns a strange default value when it fails.
- This results in a strange peak at 90 degrees in the Theta distribution. This should be corrected.

Implementation of recommendation 2

- The 90 degree peak corresponds to track candidate with too few hits providing z coordinate information
- Fix of projector unphysical values at edges of fiducial regions
- These are low momentum tracks (p < 0.4 GeV/c)



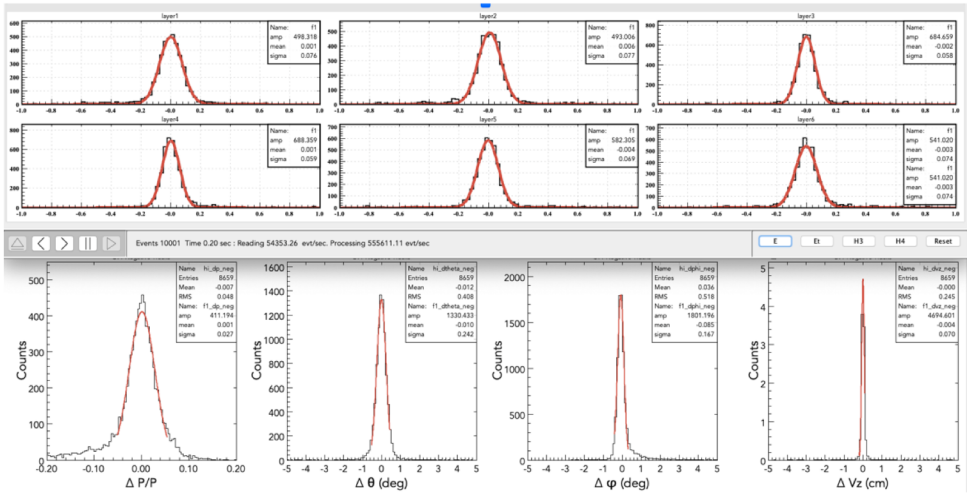
Recommendation 3

- Most importantly, the beam spot position should be an input to the old code,
- rather than just using a default value of (x,y)=(0,0). The beam position on target
- is likely to be stable, run-to-run, and needs to be determined empirically. Then the real beam position (x,y) can be used in the old code.

Implementation of recommendation 3

- The code reads the beam spot parameters from CCDB and uses this in the fit. The validation for this was done in MC by producing a sample of tracks generated at vx = 2 cm, vy = 3 cm. The resolutions are as expected.
- The effect of the beam offset in central track fitting was studies using 6.5 and 10.6 GeV elastic events (next slides).

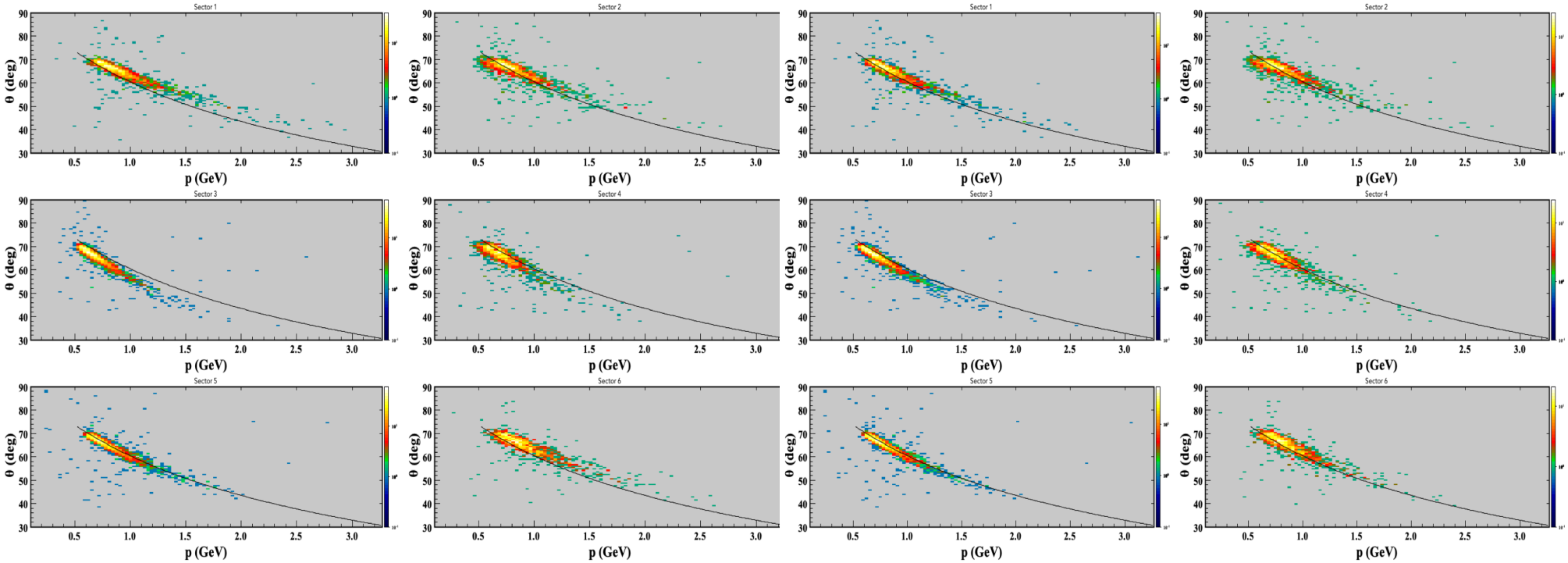
Electrons @ Vtx = (2,3,+/-5) cm



Run 5990 (6.5 GeV)

Without beam offset

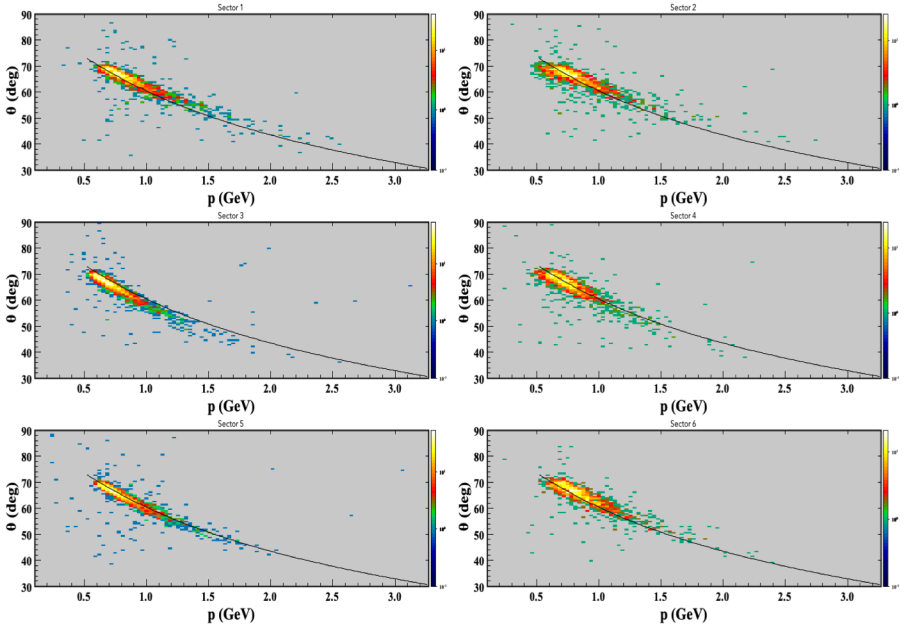
With beam offset



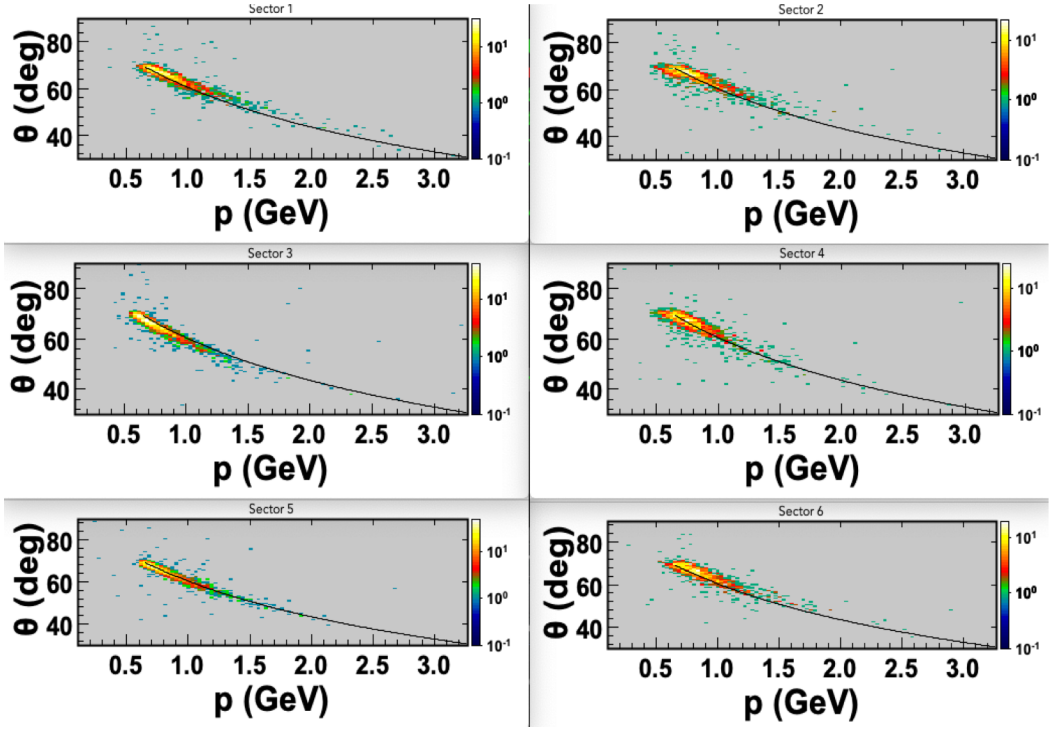
Using SVT surveys has ~no effect in reco.

Run 5990 (6.5 GeV)

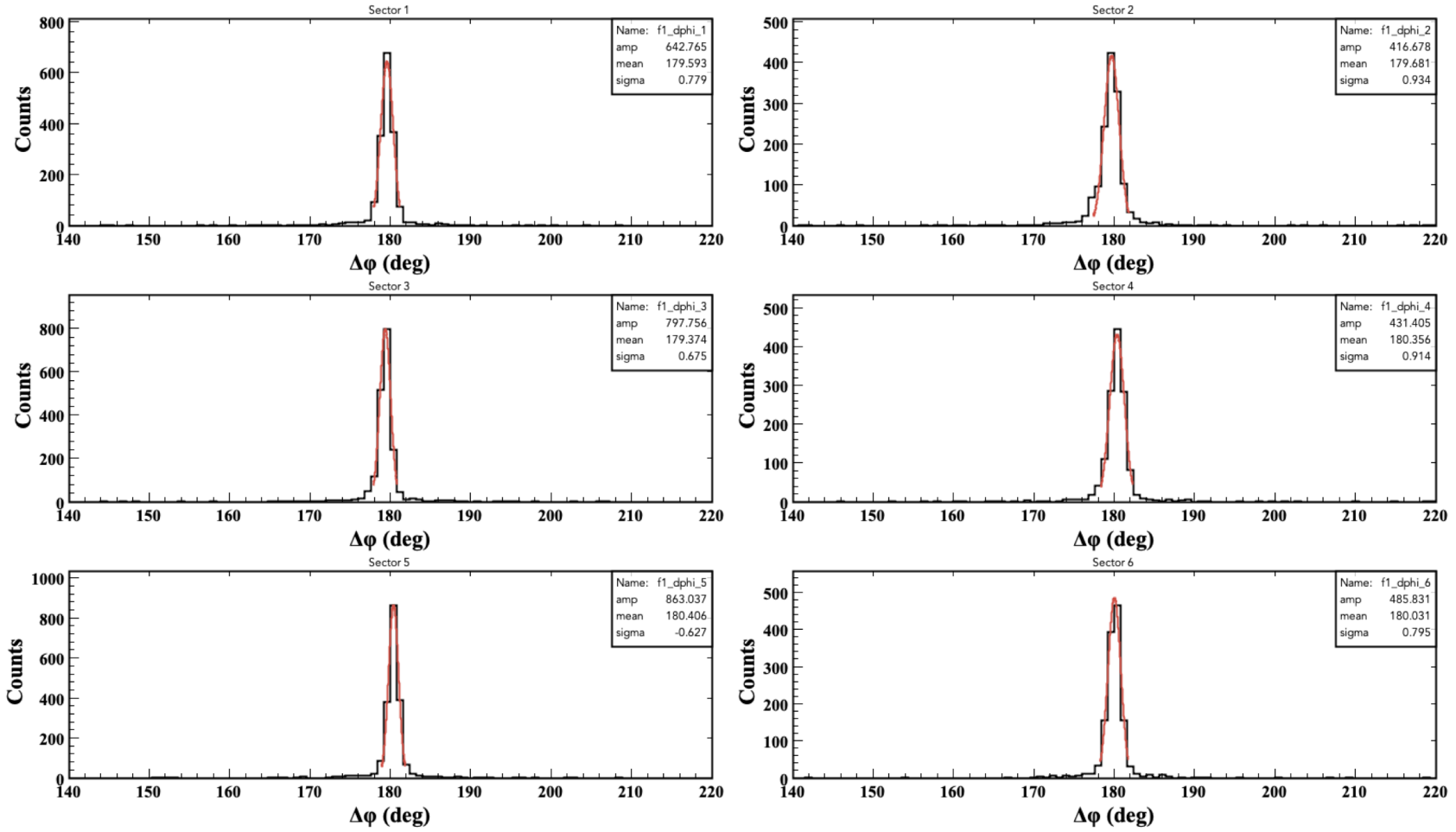
With beam offset & without SVT survey



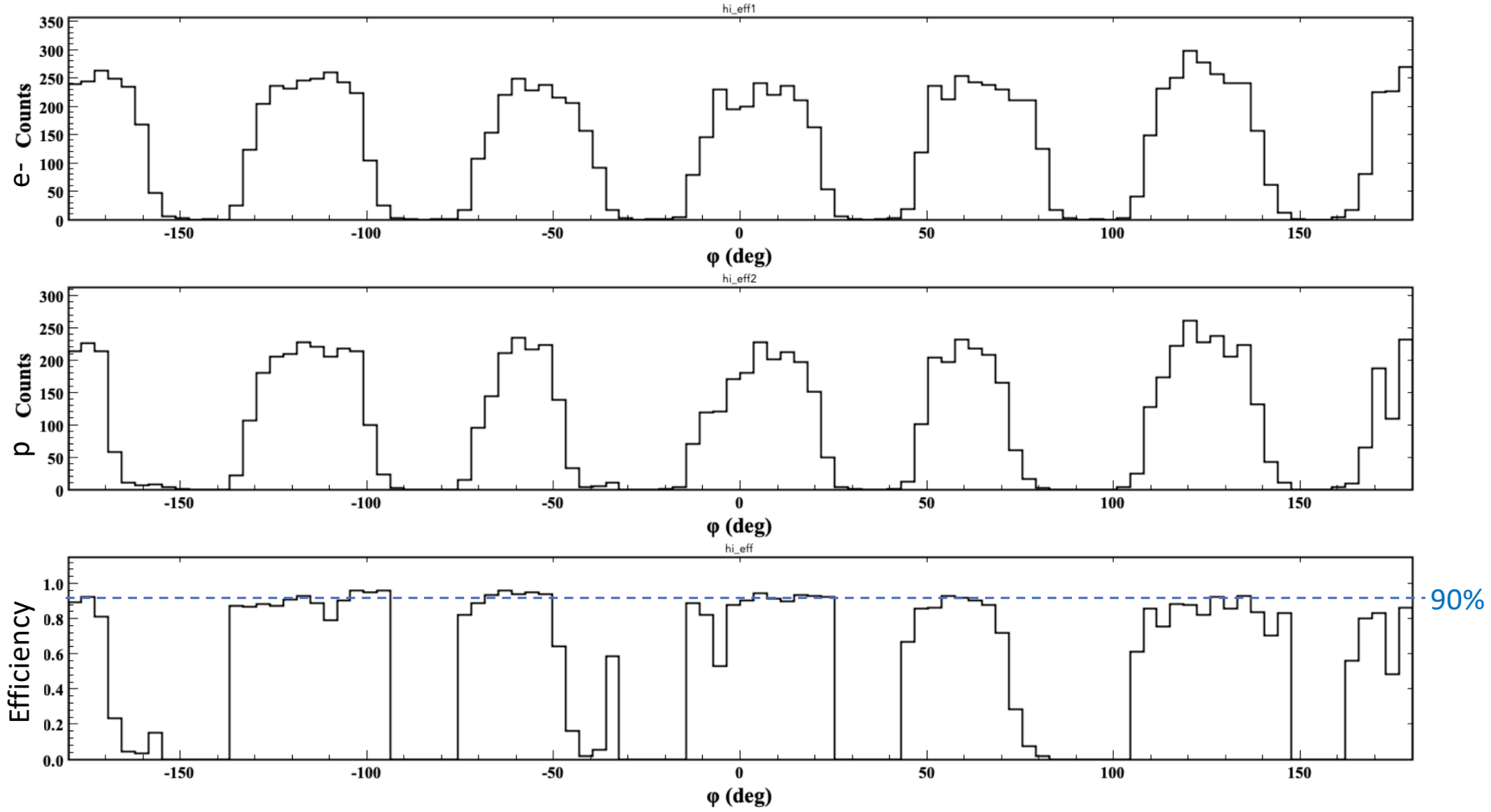
With beam offset & using SVT survey data



Run 5990, beam offset from CCDB

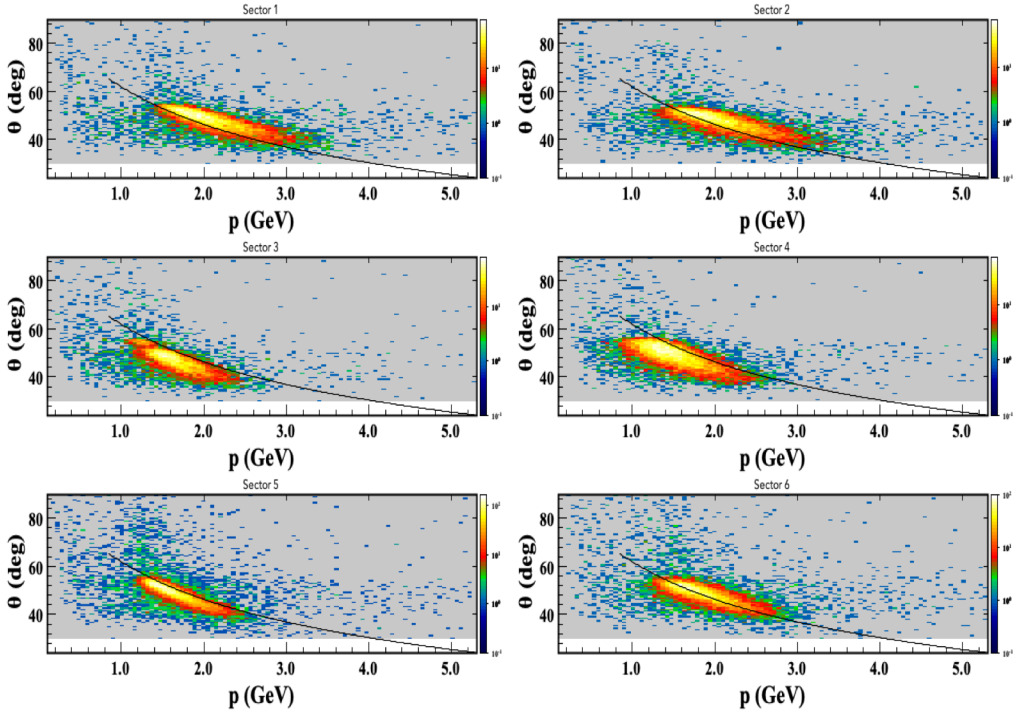


Run 5990, beam offset from CCDB: Efficiency

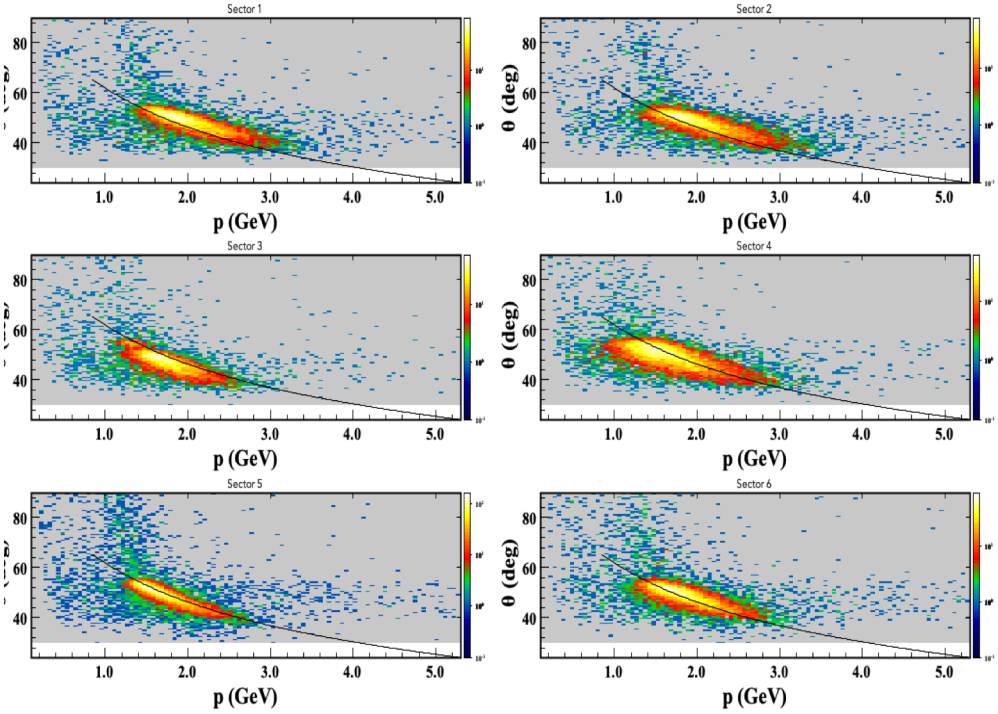


Run 5038 (10.6 GeV)

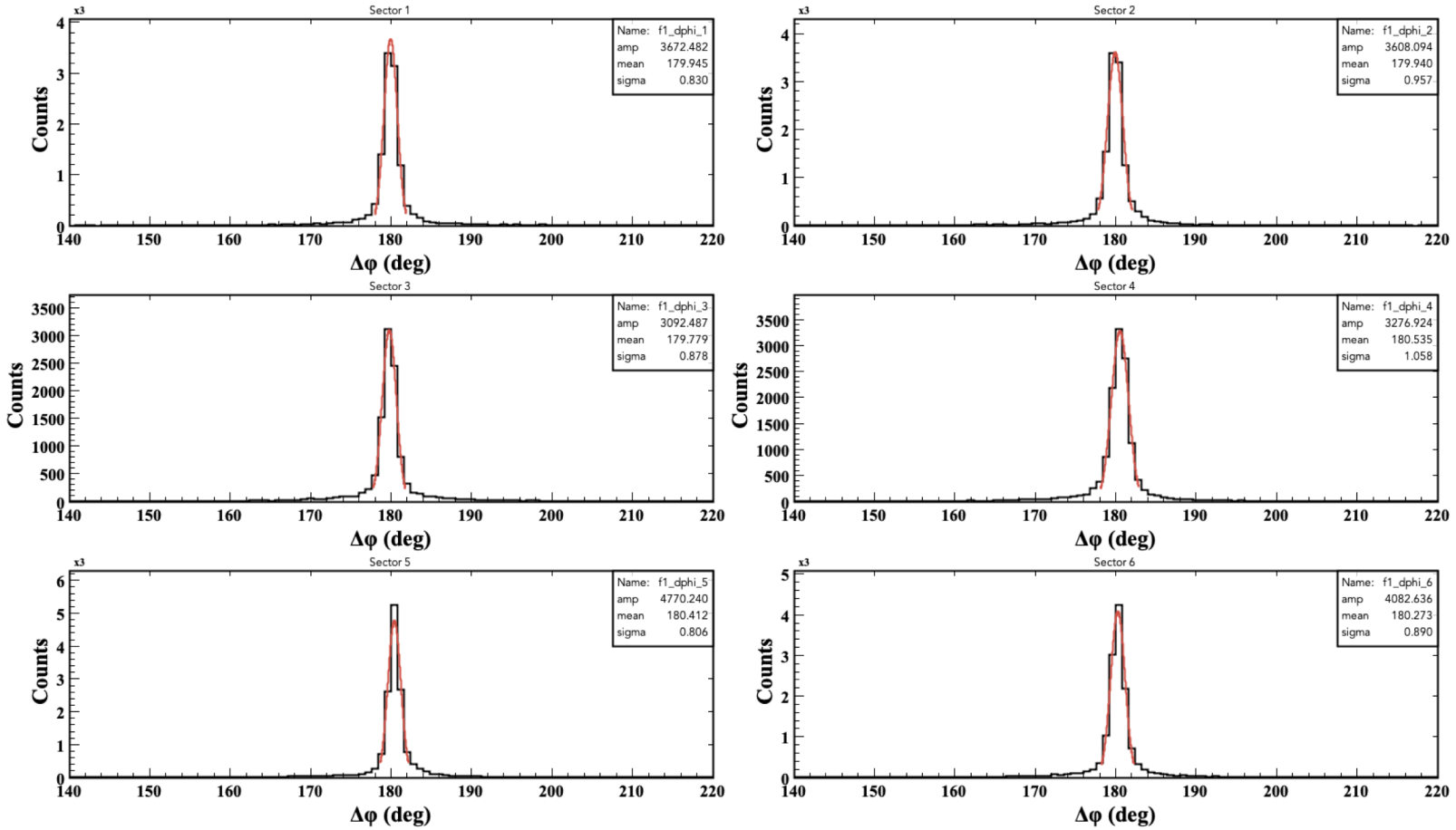
Without beam offset



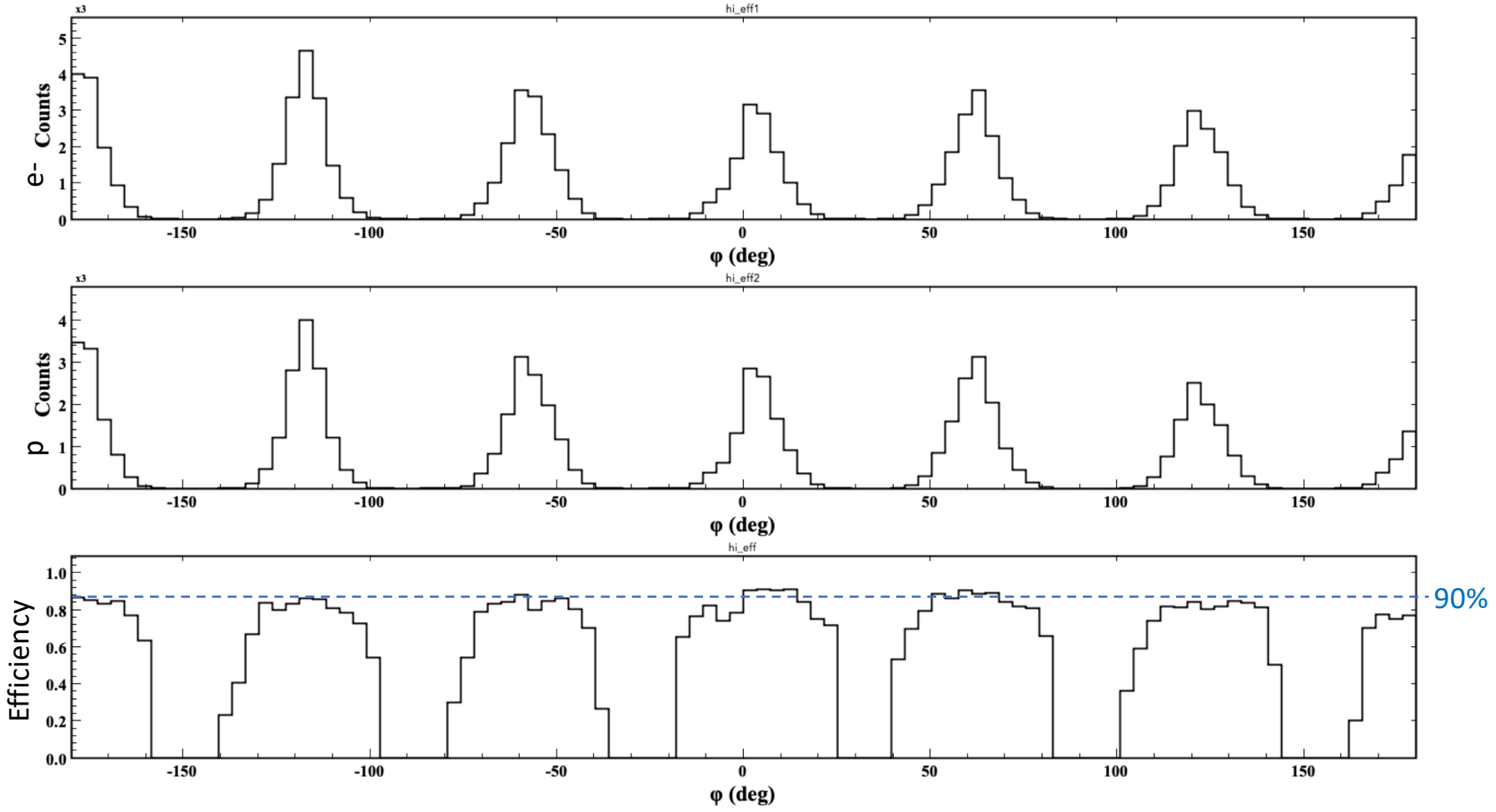
With beam offset



Run 5038, beam offset from CCDB



Run 5038, beam offset from CCDB: Efficiency



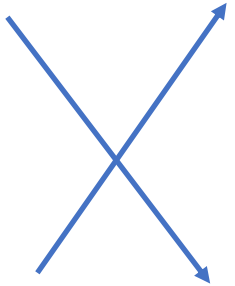
SVT Reconstruction Status

- The geometry package has the capability of doing translations and rotations of the SVT modules and to import survey numbers.
- The survey numbers are implemented using a CCDB variation. No effect in reconstruction observed from using the survey numbers in reconstruction. Alignment is needed.
- The code now reads all constants from CCDB for both SVT and BMT systems.
- The code reads the beam spot parameters from ccdb and uses this in the fit. The validation for this was done in MC by producing a sample of tracks generated at vx = 2 cm, vy = 3 cm. The resolutions are as expected.
- A bug was found in the calculation of the space residuals from the fit. The code produces a trajectory from the track parameters and reports points at each surface. For the SVT these are planes which are defined from a direction and a reference point. The reference point was not at the correct radius, which produced a parallax effect in the residuals. This has been fixed.
- Service init exceptions fix (BMT geometry constants loading on multiple threads).

Two codes with strengths and weaknesses

Official Version

- Fast
- Fairly resilient to misalignments between detectors
- Need the beam position as input (bias)
- Remaining dependencies possibly related to track intercept issues.
- Not handling misalignments corrections



Exploratory Version

- Fully Unbiased
- Robust track intercept computations
- Handle alignment corrections
- More sensitive to detector misalignments
- Slower to execute on data.

The perfect code can be written from both existing codes.

Alignment status

- Preliminary alignment procedure compatible with exploratory branch.

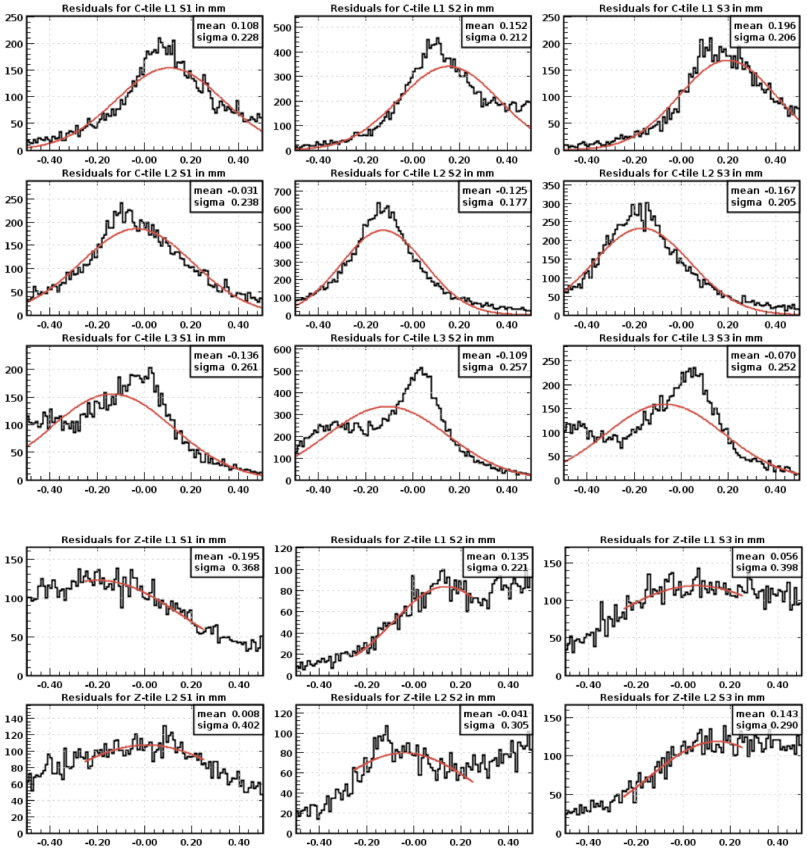
- Fairly simple approach:
  - Do tracking on cosmics or alignment data by ignoring a detector.
  - Find translations/rotations to minimize residuals of ignored detector.
  - Proceed to next detector to align and iterate over all detectors as long as translations/rotations are not stable.

- Drawback: Long to perform with many detectors and many local minima (Curvature of tracks on data reveals that alignment is not optimal)
  => Conclusion validated by my CERN colleagues.

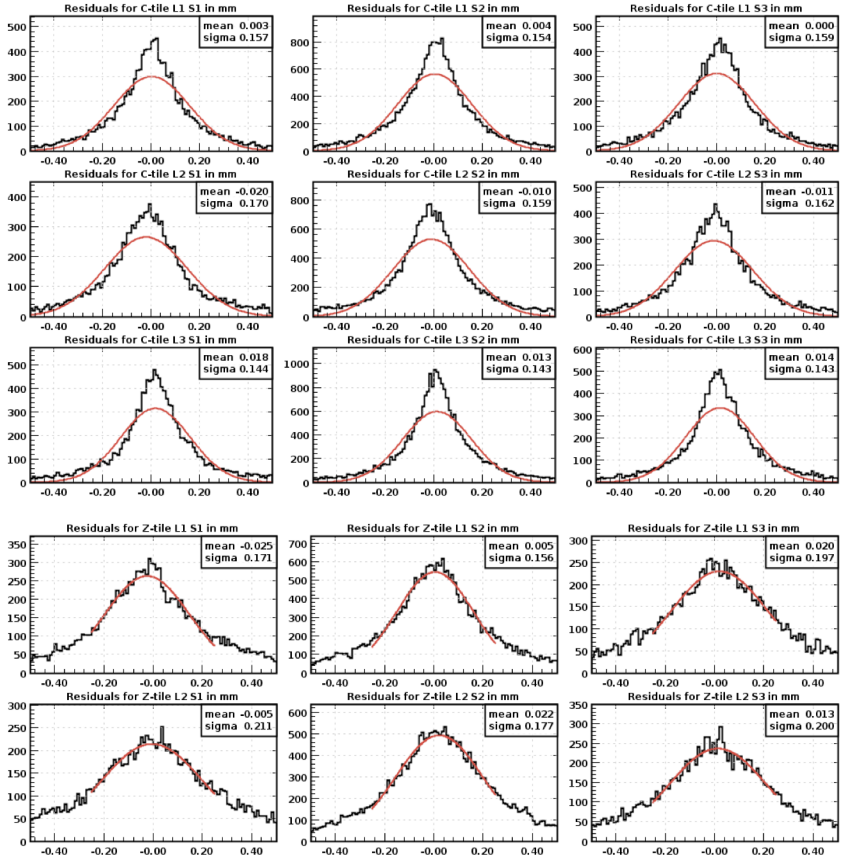
- Nevertheless we have demonstrated that there are significant misalignments and we can correct for them.

Alignment status: Example on micromegas

Before Alignment



After alignment



Alignment perspectives

- Taking as example our colleagues from CERN:
  - > Millepede
  Advantages: Wide community, well tested and quasi-one shot.
  Drawbacks: Need a well-defined parameterization of tracks (Not easily compatible with KF and written for straight tracks.)

  -> Kalman Filter approach
  Advantages: Veronique and I know KF very well and it is compatible with KF tracking (can use beam data!)
  Drawbacks: Cumbersome to keep covariance matrix positive definite + not-as-widely-used-as Millepede.

- Both has been compared by CERN colleagues and give similar results.