Recent Improvements to Forward Tracking

T. Cao (Jefferson Lab)

CLAS12 Collaboration Meeting
Nov 7 - 10, 2023

Introduction

- Main updates in PR#107 under review:
  - Refactored Kalman Filter (KF) package
  - Reset conditions for iteration termination
  - Improved pattern recognition and uncertainties of initial state for seed construction
  - Developed DC-uRWell tracking package
  - Reset initial covariance matrix (CM) for each iteration in KF tracking (Discussed next)
- Recent project for Application of Deterministic Annealing Filter (DAF) (Discussed next)

Why Reset Initial CM for Iterations of Tracking?

- In the old tracking, final state from previous iteration is input as initial state for next iteration, including state vector and CM. It causes that CM becomes smaller and smaller along iterations since measurement errors are repeatedly counted in the filter equation $C_k = \left[\left(C_k^{k-1}\right)^{-1} + H_k^T (V_k)^{-1} H_k \right]^{-1}$ along iterations.

- In spirit of Kalman Filter, initial CM for each iteration should give enough space for tracking[1].

- Therefore, final CM from previous iteration should be blown up to set initial CM for next iteration.

- Referring to Acts common tracking software[2], diagonal items in final CM of previous iteration are scaled to form initial CM of next iteration.

- It is critical to tune factor for inflation of initial CM. If factor is too small, inflation is not enough. If factor is too large, resolution of tracking results becomes worse.

[1] R. Frühwirth, Application of Kalman filtering to track and vertex fitting, Nuclear Instruments and Methods in Physics Research Section A, 1987

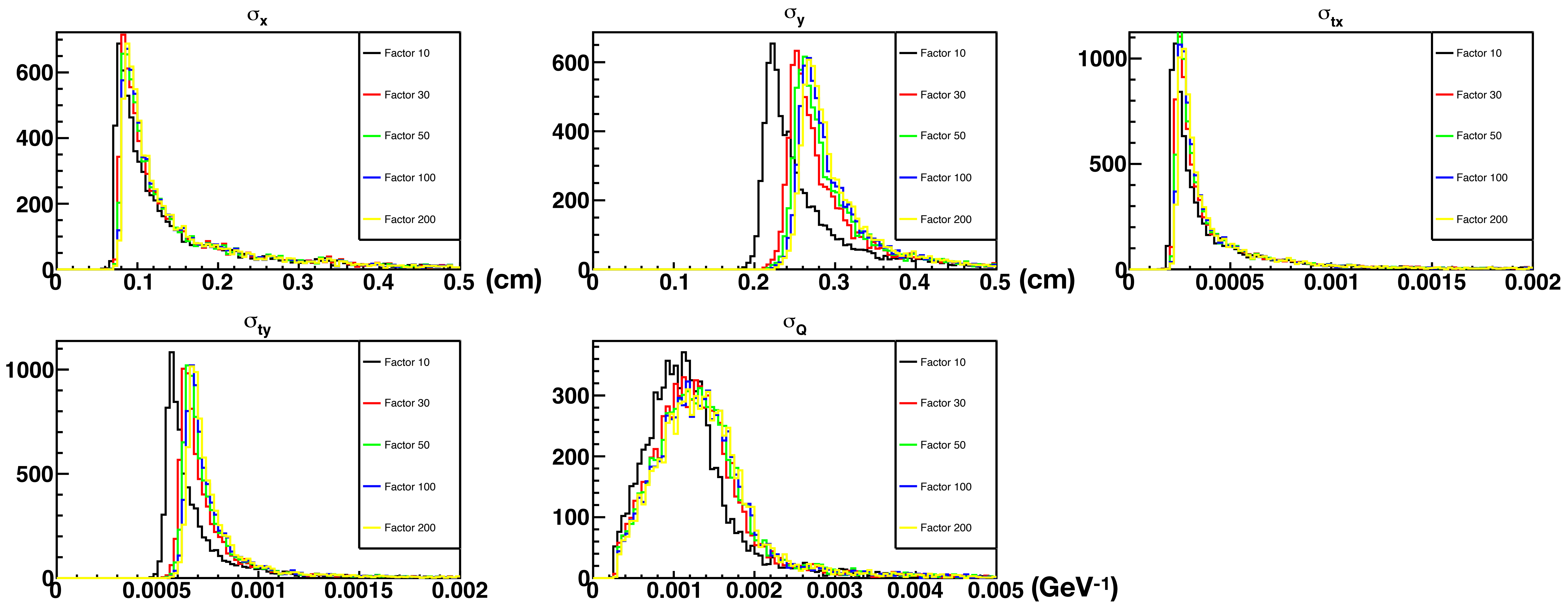
[2] https://acts.readthedocs.io/en/latest/

Pure muon sample, zero solenoid filed

Tune factor

Uncertainties of Track Parameters at Vertex for Different Factors

Uncertainties are calculated by covariance matrix of TB state at vertex



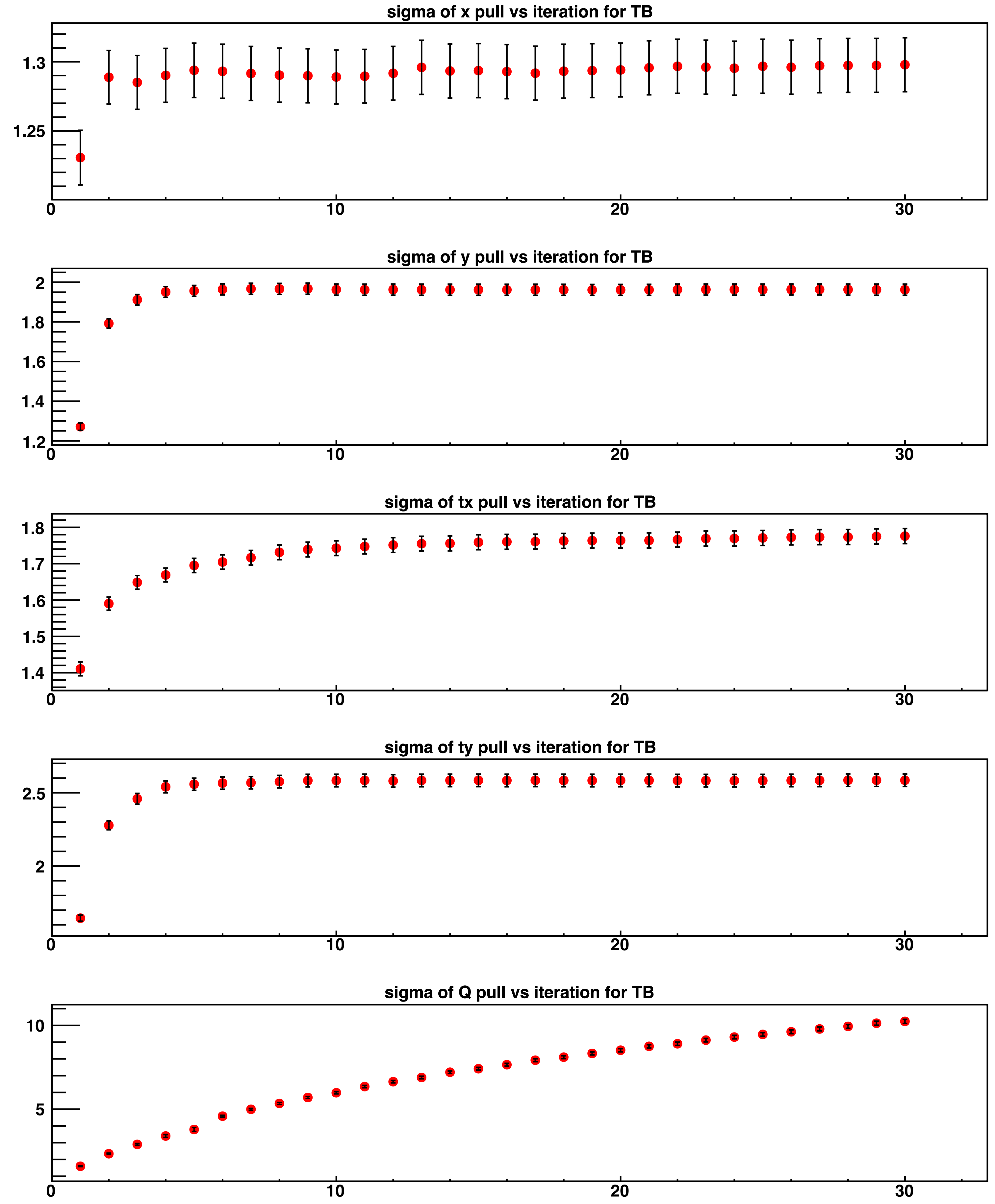
As the factor increases, uncertainties become larger until the factor reaches big enough.

Pure muon sample, zero solenoid filed

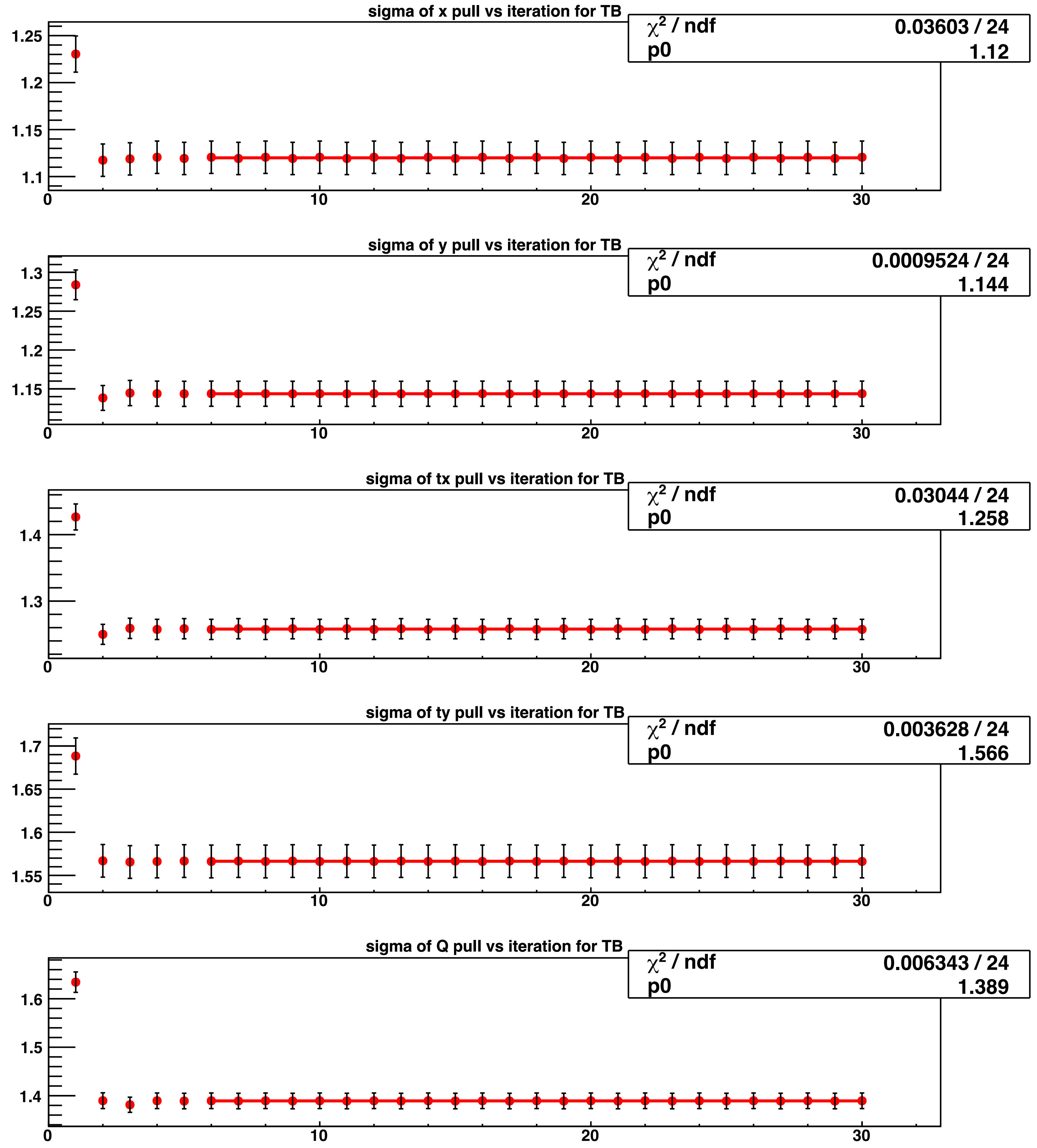
Pull = (state from tracking - truth) / uncertainty from covariance matrix of state

Sigma for Pull of Final State along Iterations

Old



New with factor of 100

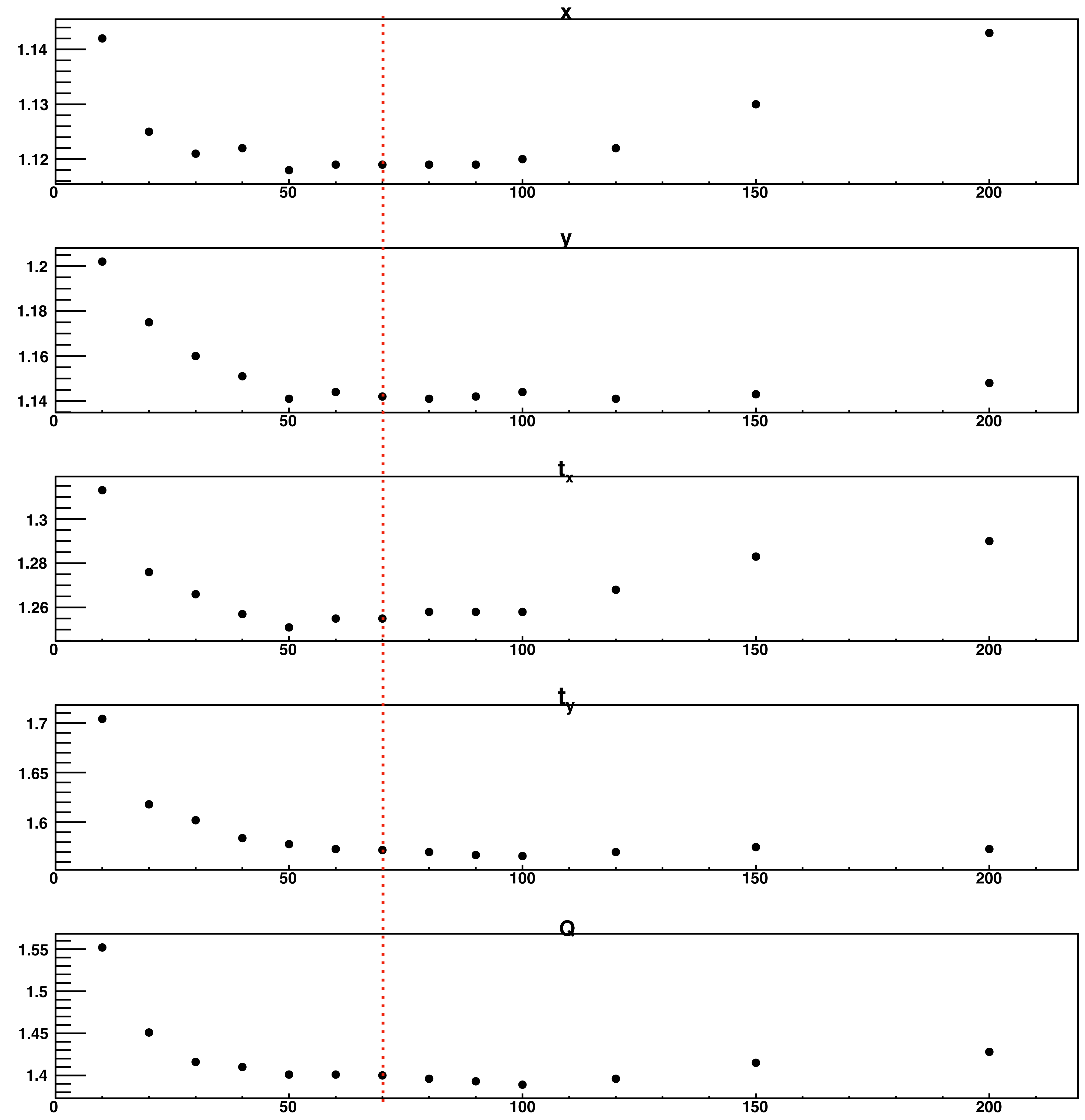


Pulls for the old tracking are not reasonable, while sigmas of pulls for the new tracking are well converged and close to 1.

Pure muon sample, zero solenoid filed

Tune factor

Converged Values for Sigma of Pull vs. Factor

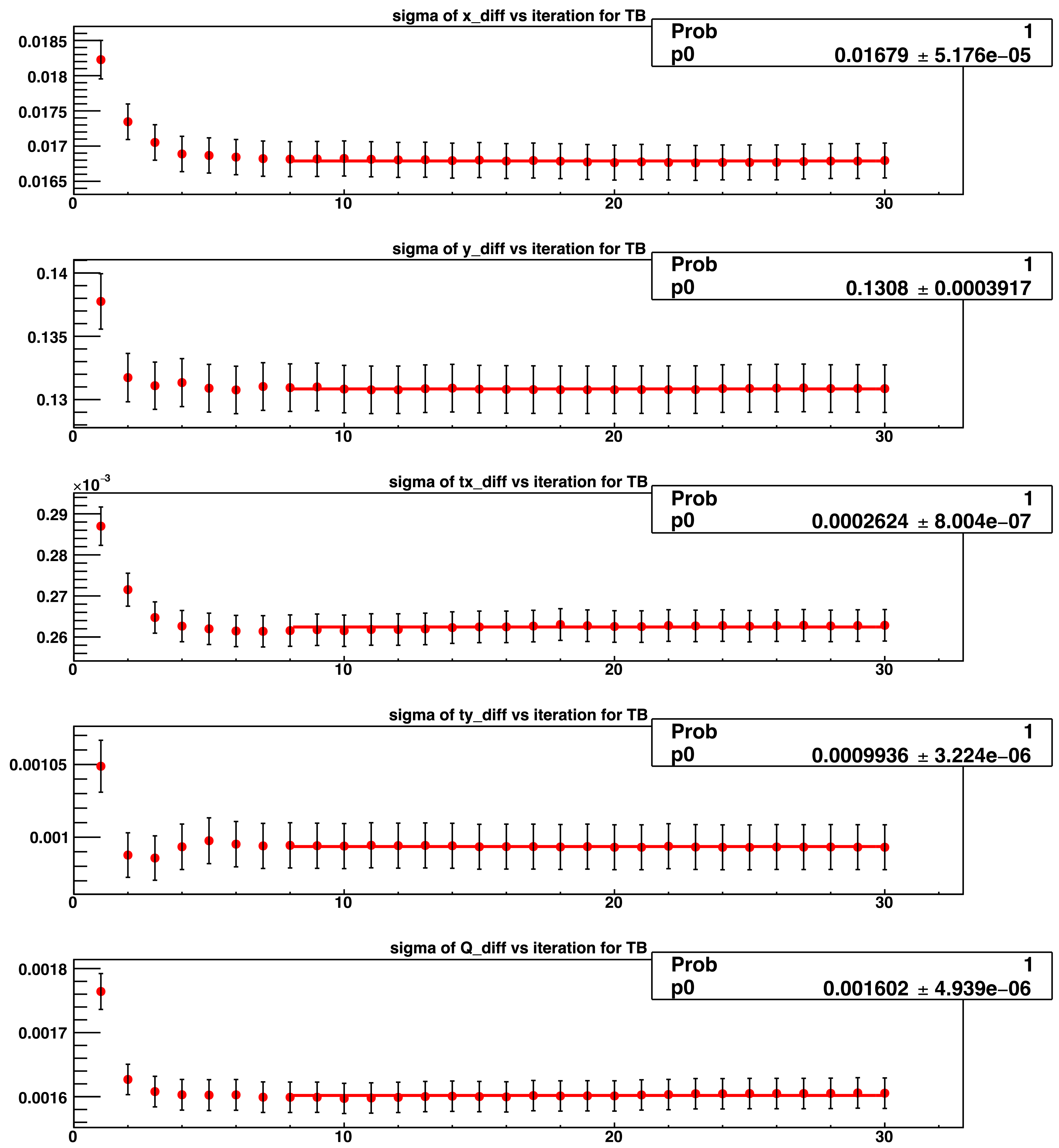


Overall, sigma of pull reaches minimum for all track parameters when factor = 70.

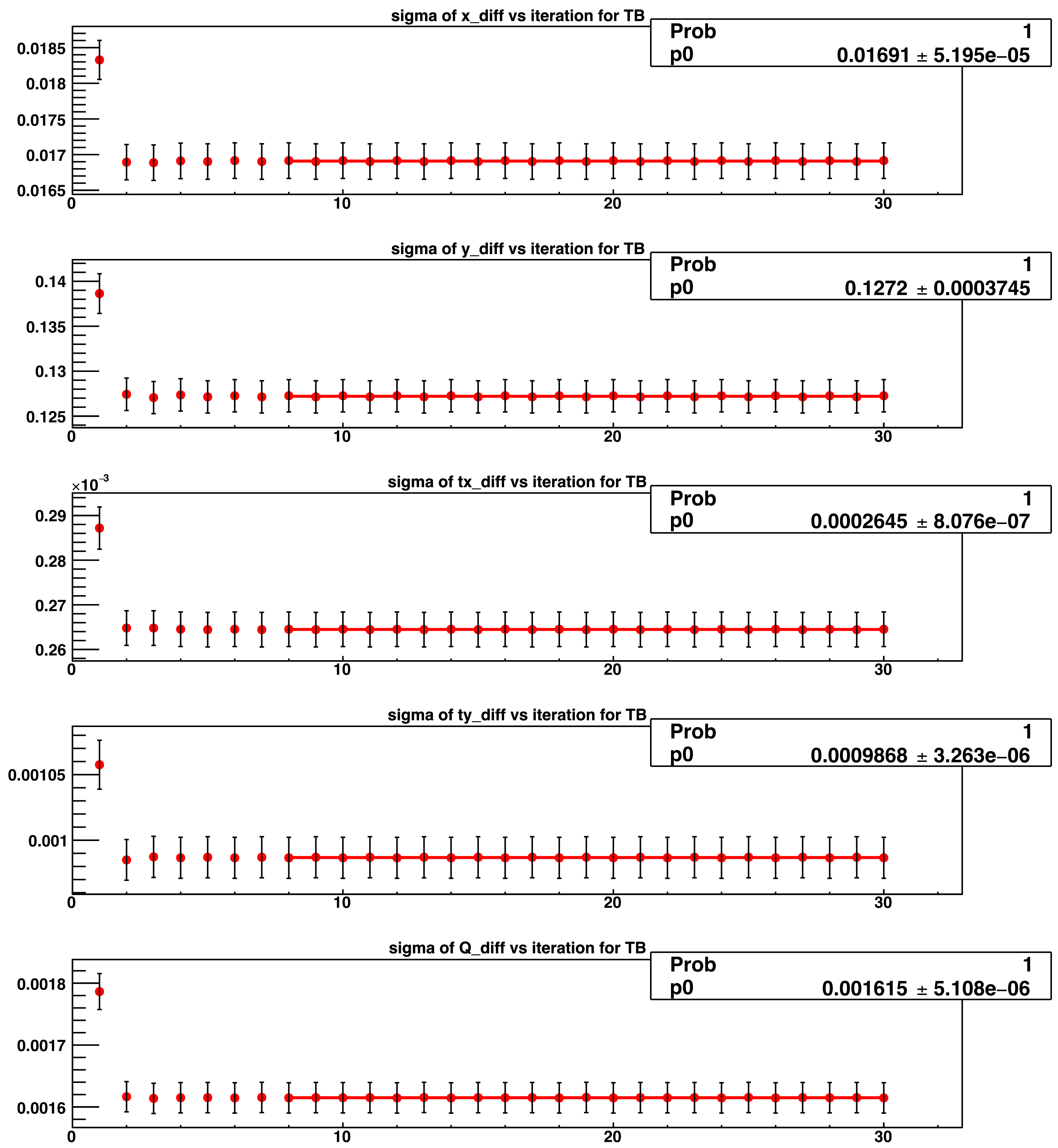
Pure muon sample, zero solenoid filed

Resolution: Sigma for Difference between Final State and Truth along Iterations

Old



New with factor of 70



- **The new package is more quickly converged.**
- **Converged values between the old and new packages are very close.**

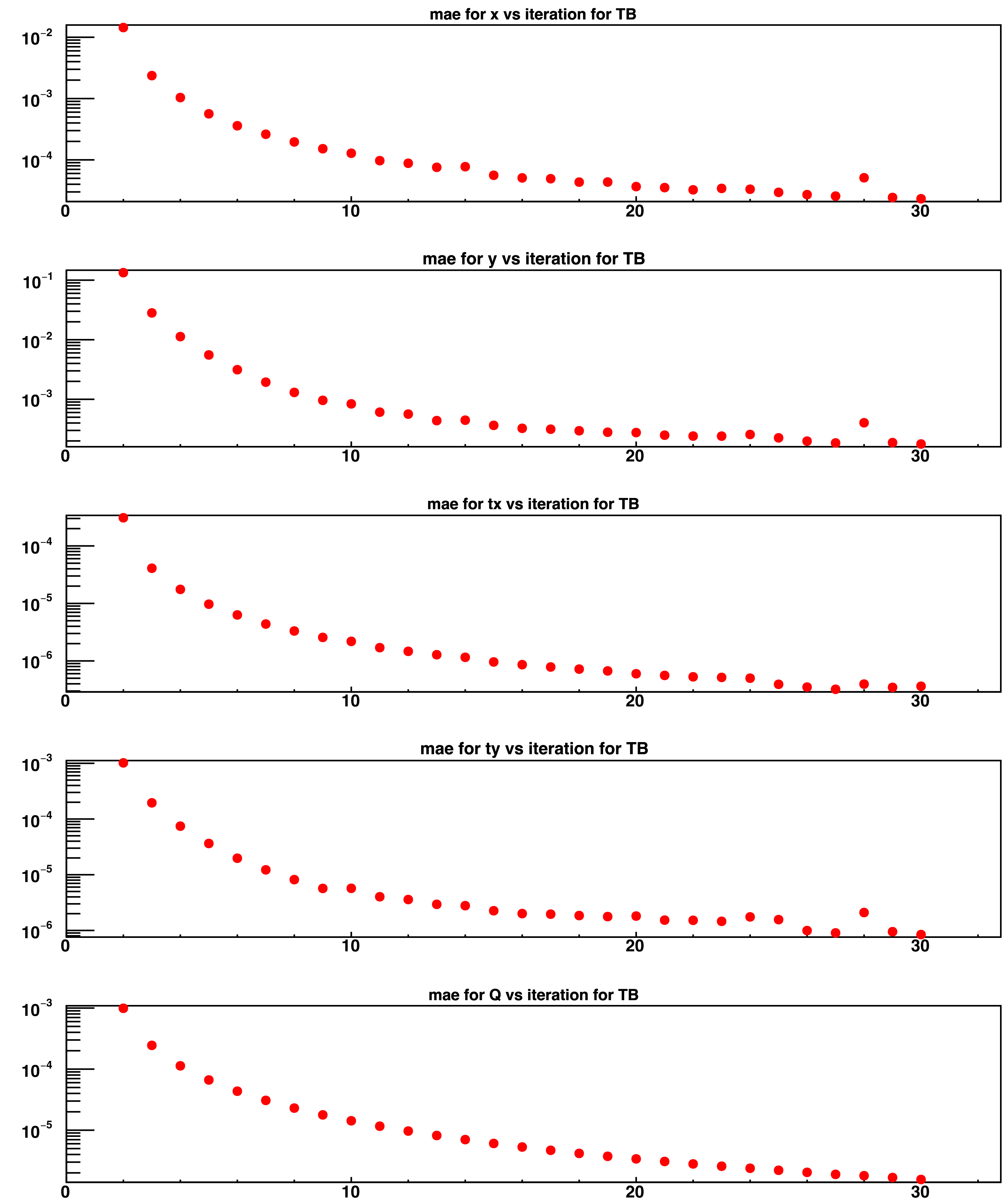
Old/New	x	y	t_x	t_y	Q
Ratio of σ	99.3%	102.8%	99.2%	100.7%	99.2%

Data: run 6666

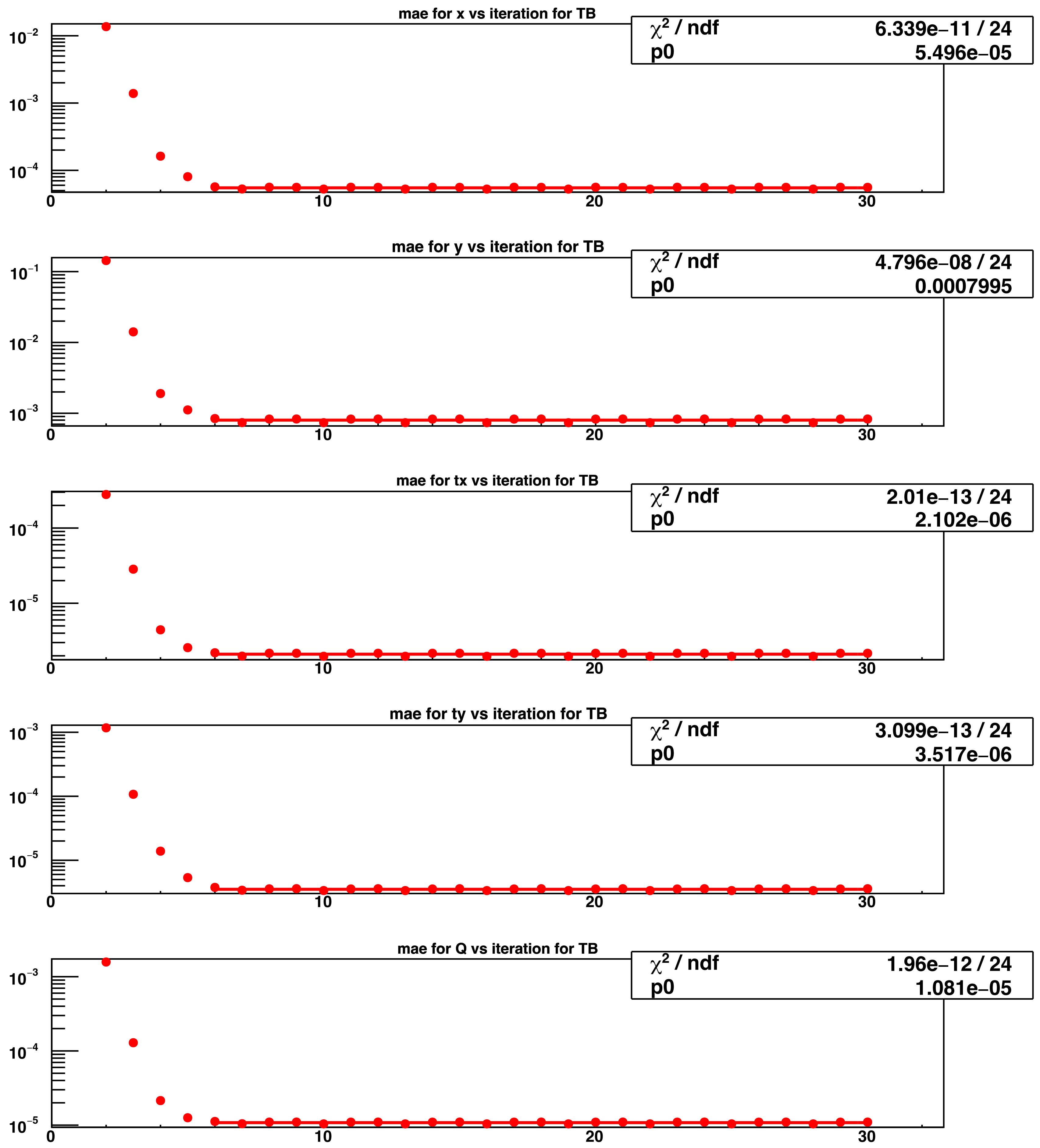
Bank: TimeBasedTrkg::TBTracks

Mean Absolute Error (MAE) of Neighbored Iterations along Iterations

Old



New with factor of 70



- **MAE of neighbored iterations is well converged, and converged values are much smaller.**
- **Converged values can be directly set as new conditions for iteration termination.**

Run 6666

Reset Iteration Termination Conditions

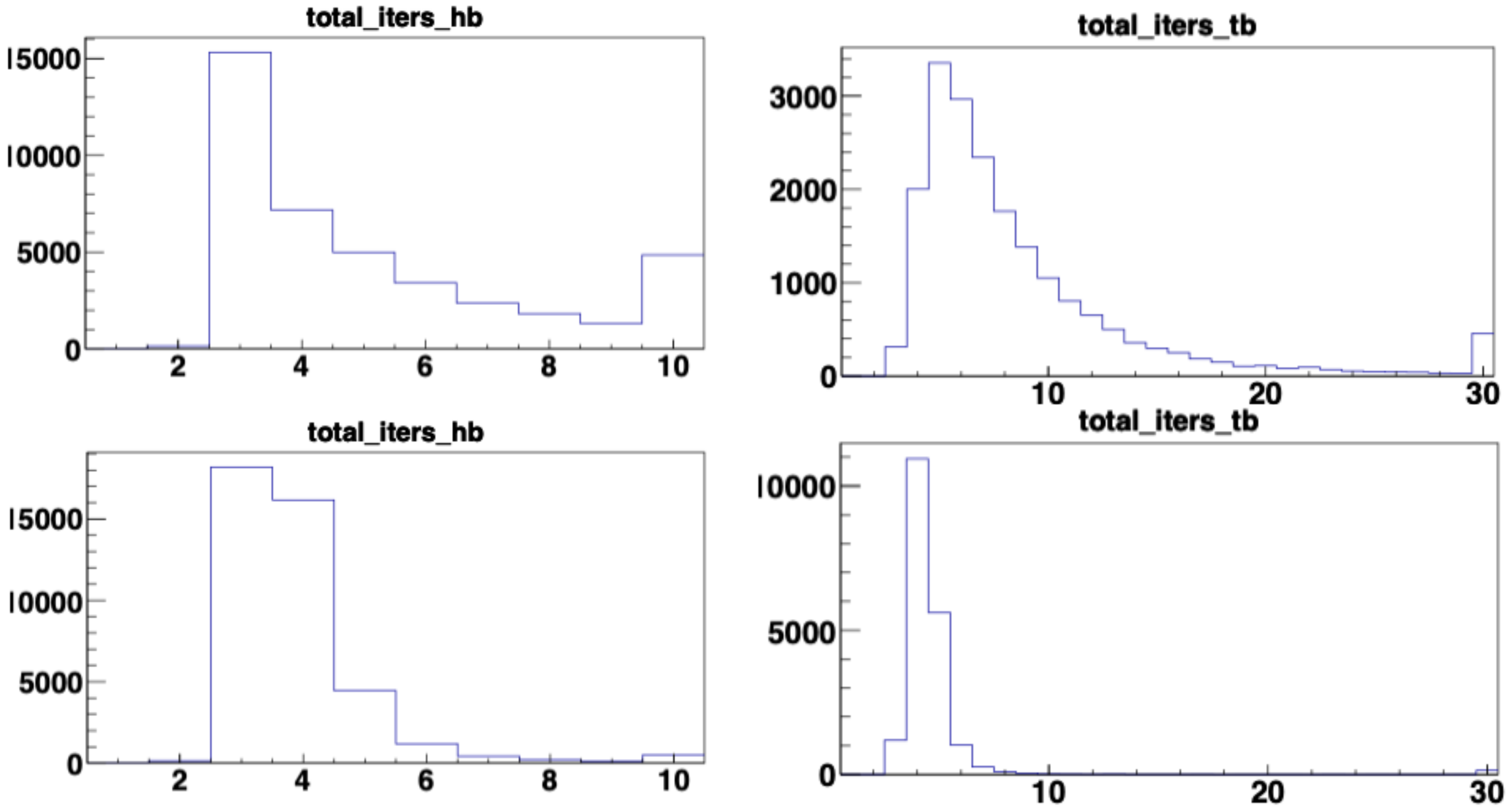
	|x_diff|	|y_diff|	|tx_diff|	|ty_diff|	|Q_diff|
Old HB	7.3E-02	4.3E-01	9.2E-04	2.1E-03	3.3E-03
New HB	1.2E-02	1.4E-01	2.5E-04	1.0E-03	1.6E-03
Old TB	5.0E-04	2.1E-03	8.8E-06	1.4E-05	5.5E-05
New TB	5.5E-05	8.0E-04	2.1E-06	3.5E-06	1.1E-05

HB

TB

Old

New

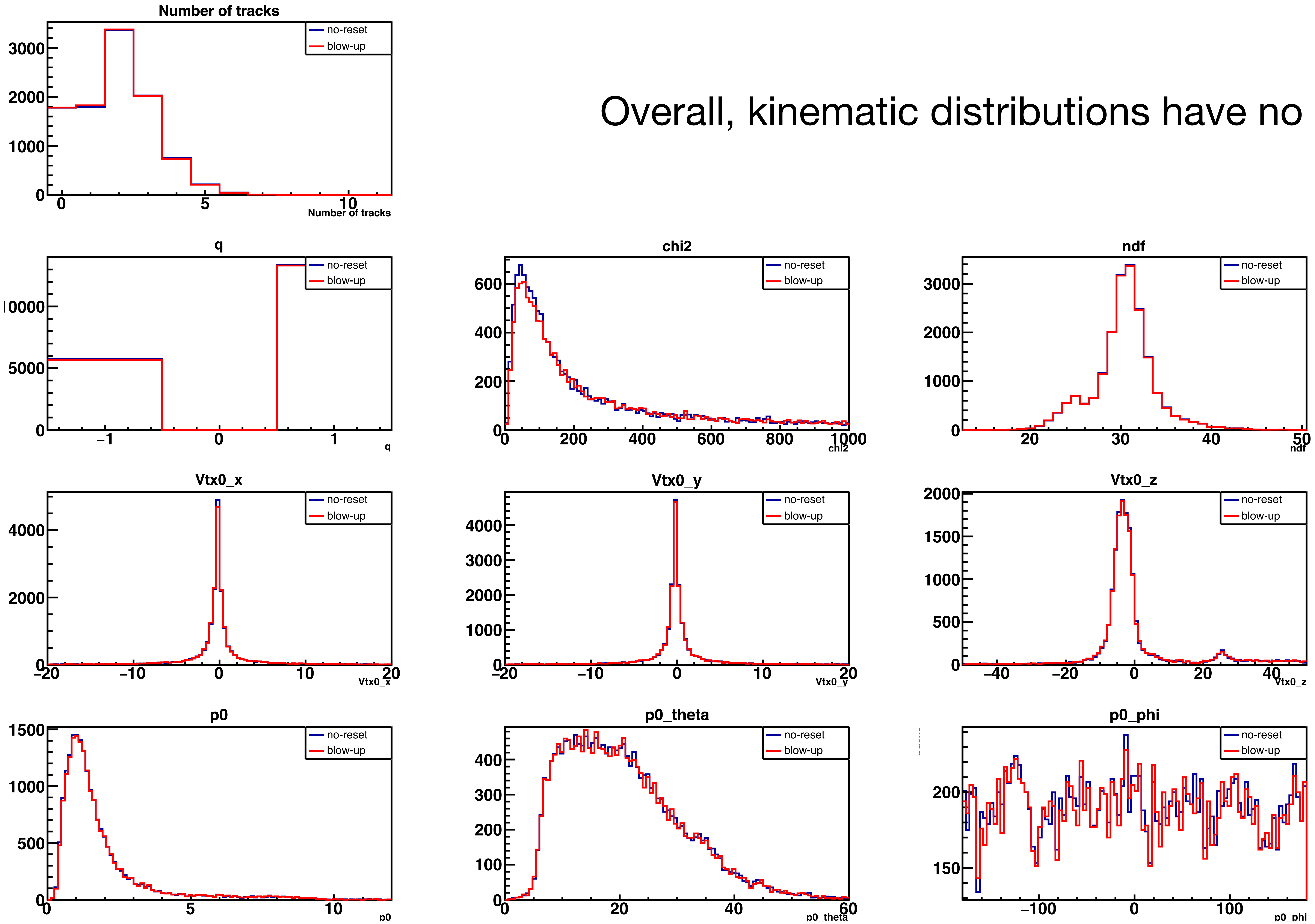


- Although new conditions are much tighter than old conditions, total number of iterations for new tracking is much less.
- Besides, distribution for total number of iterations is much more centralized for new tracking.
- On average, much less iterations are needed for the new package, so CPU efficiency is significantly improved.

Data: run 6666

Bank: TimeBasedTrkg::TBTracks

Comparison of TB Tracks



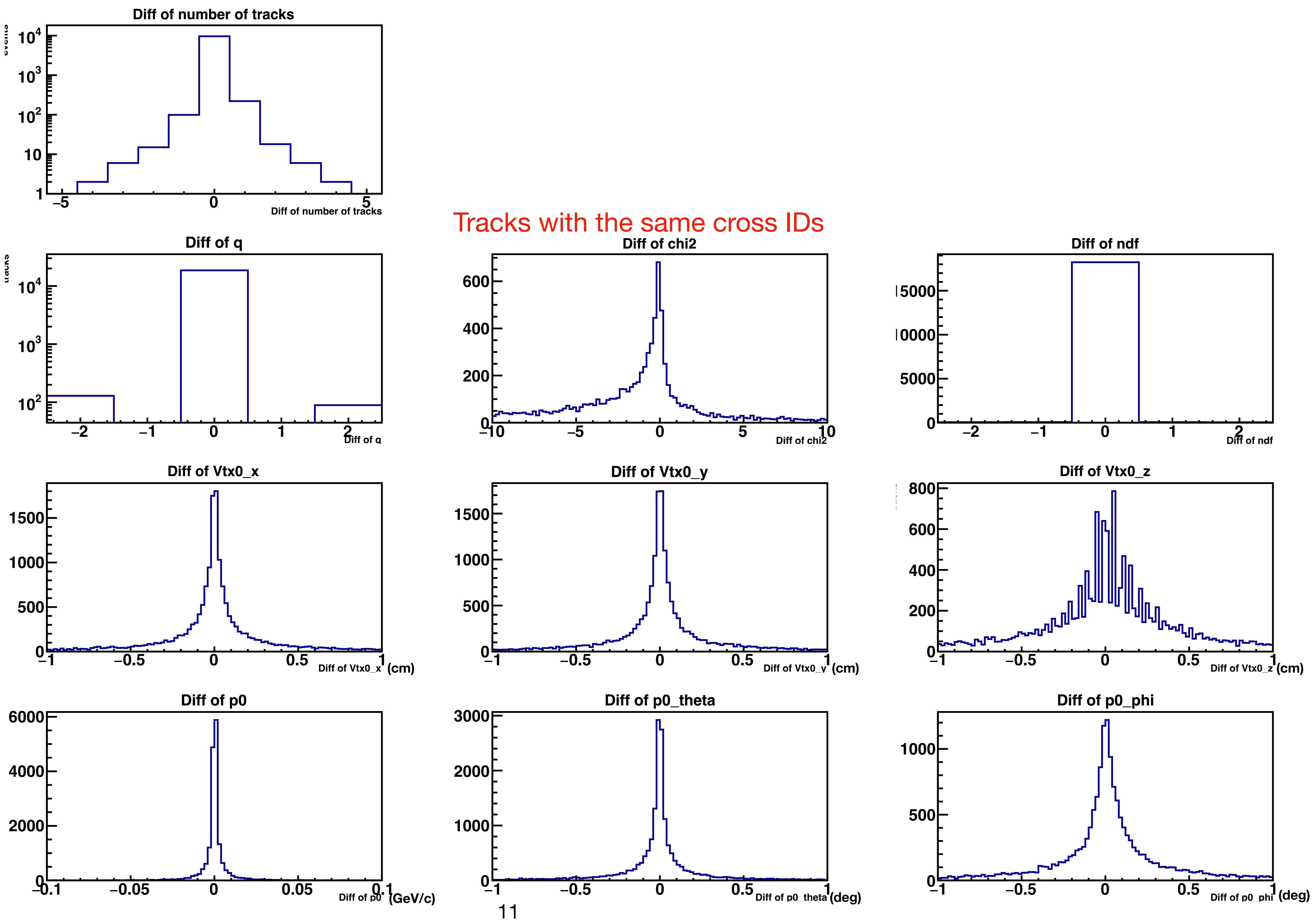
Overall, kinematic distributions have no big difference.

Difference of TB Tracks

~5.3% events have different number of tracks.

For ~1.2% tracks, charge is opposite even if cross IDs are the same.

Difference of kinematics is not negligible for both vertex and momentum.

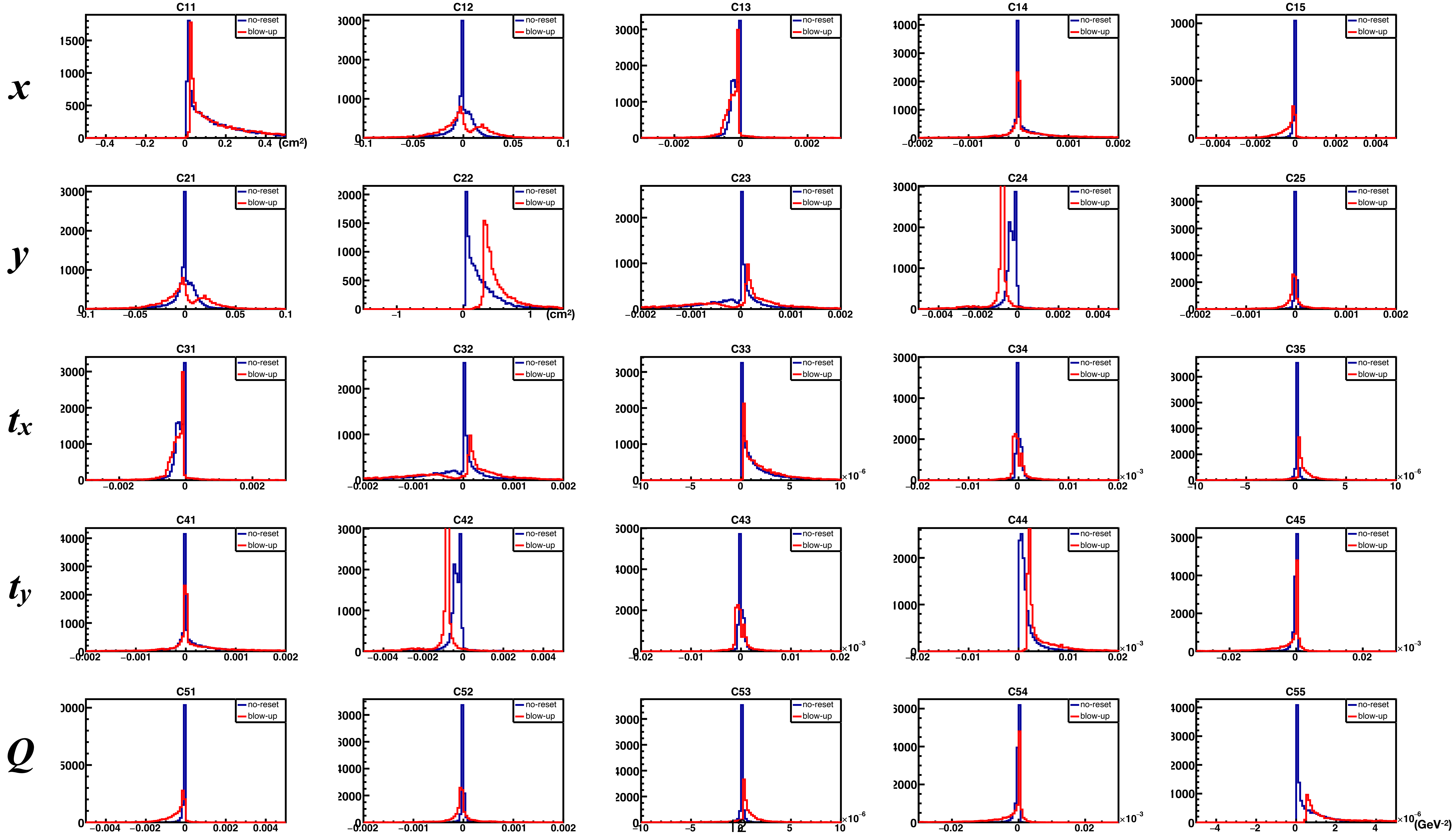


Data: low lumin. run 5700

Bank: TimeBasedTrkg::TBTracks

Comparison of Covariance Matrix at Vertex for TB Tracking

Track representation: x y t_x t_y Q



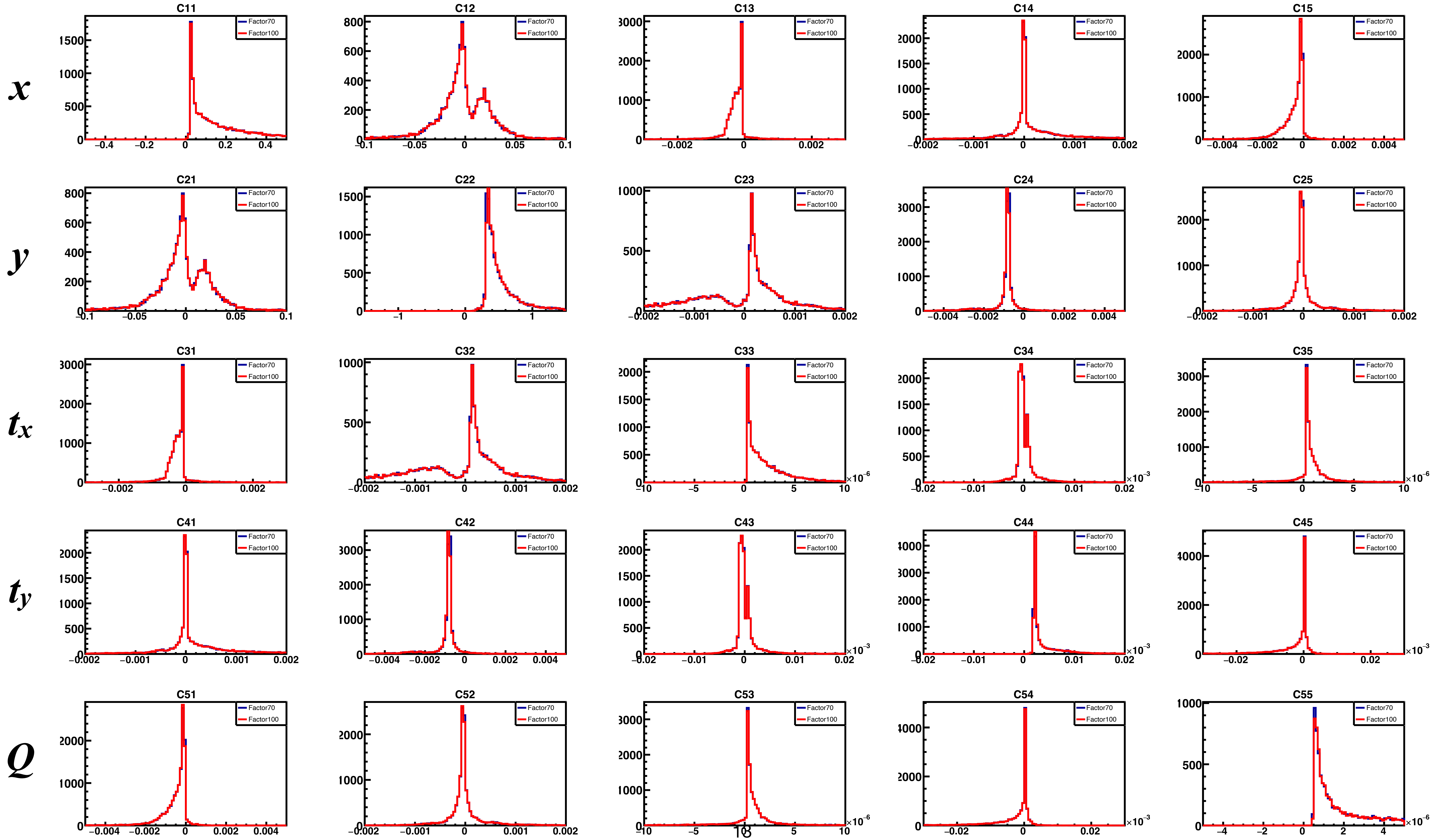
Data: run 5700

Bank: TimeBasedTrkg::TBTracks

Comparison of Covariance Matrix at Vertex between factors of 70 and 100

Track representation: x y t_x t_y Q

Factor of 70 is enough.



Data: run 5700

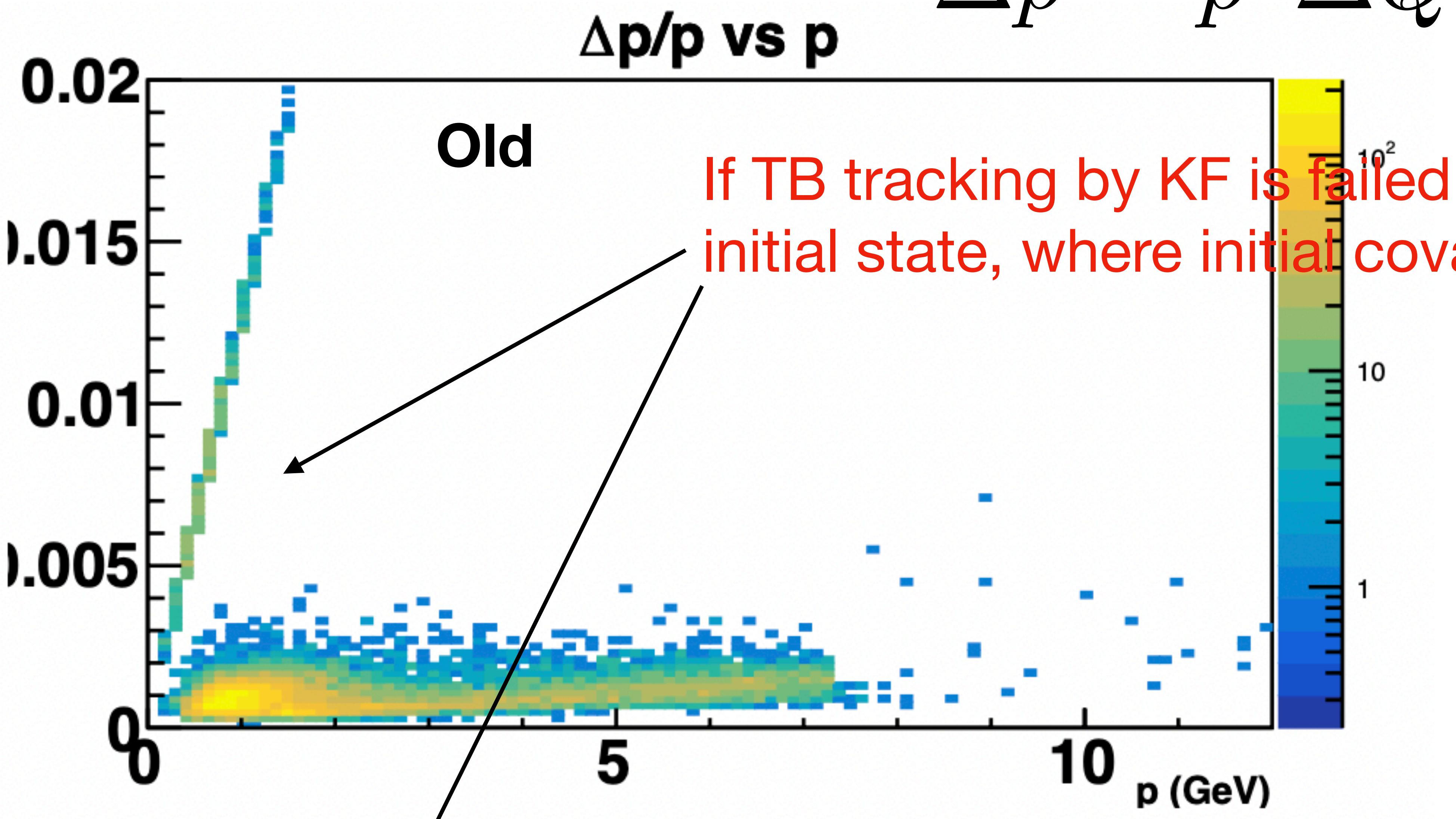
Bank: TimeBasedTrkg::TBTracks

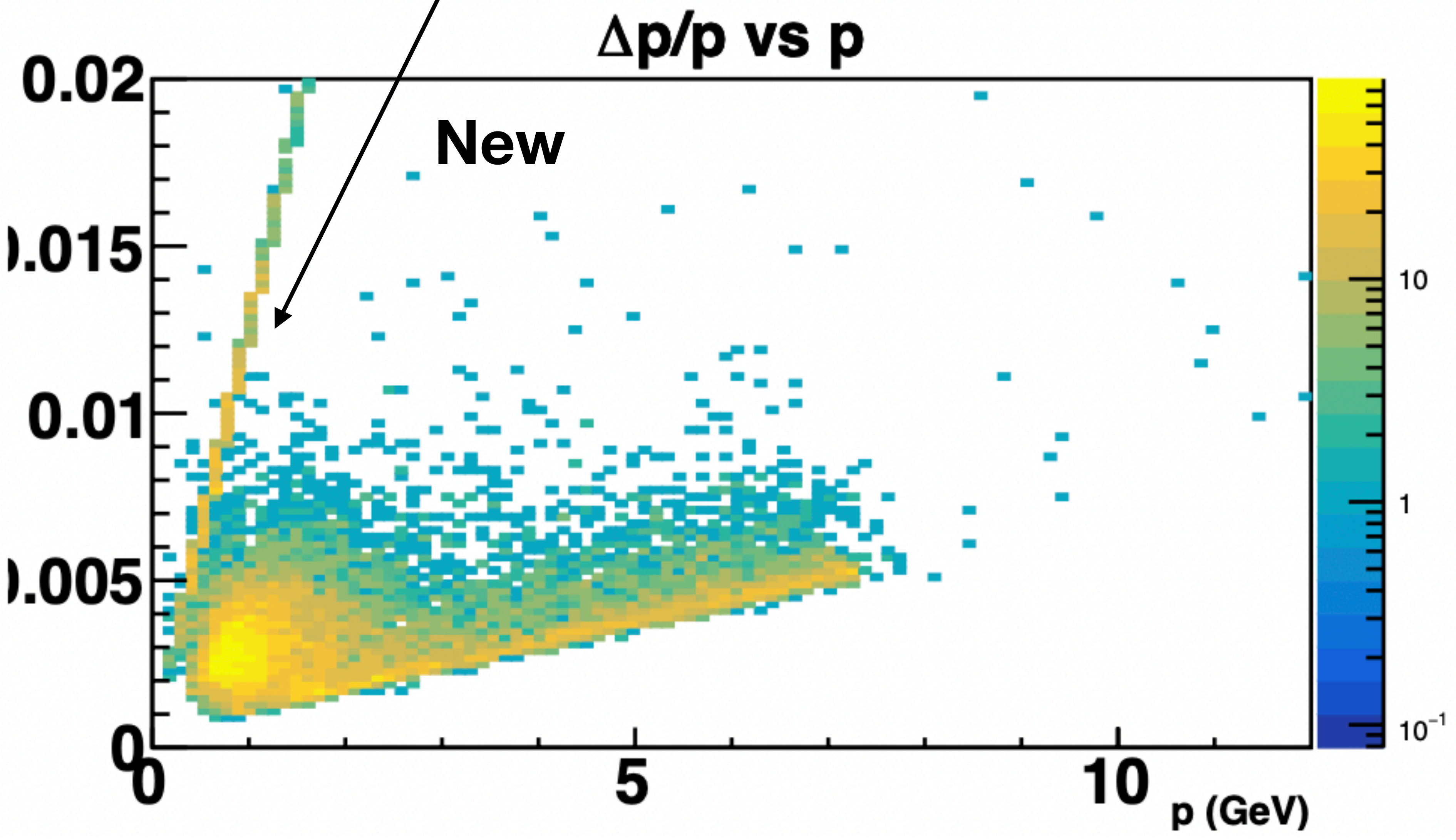
Preliminary Study of C55 (Variance of Q) in Final Covariance Matrix

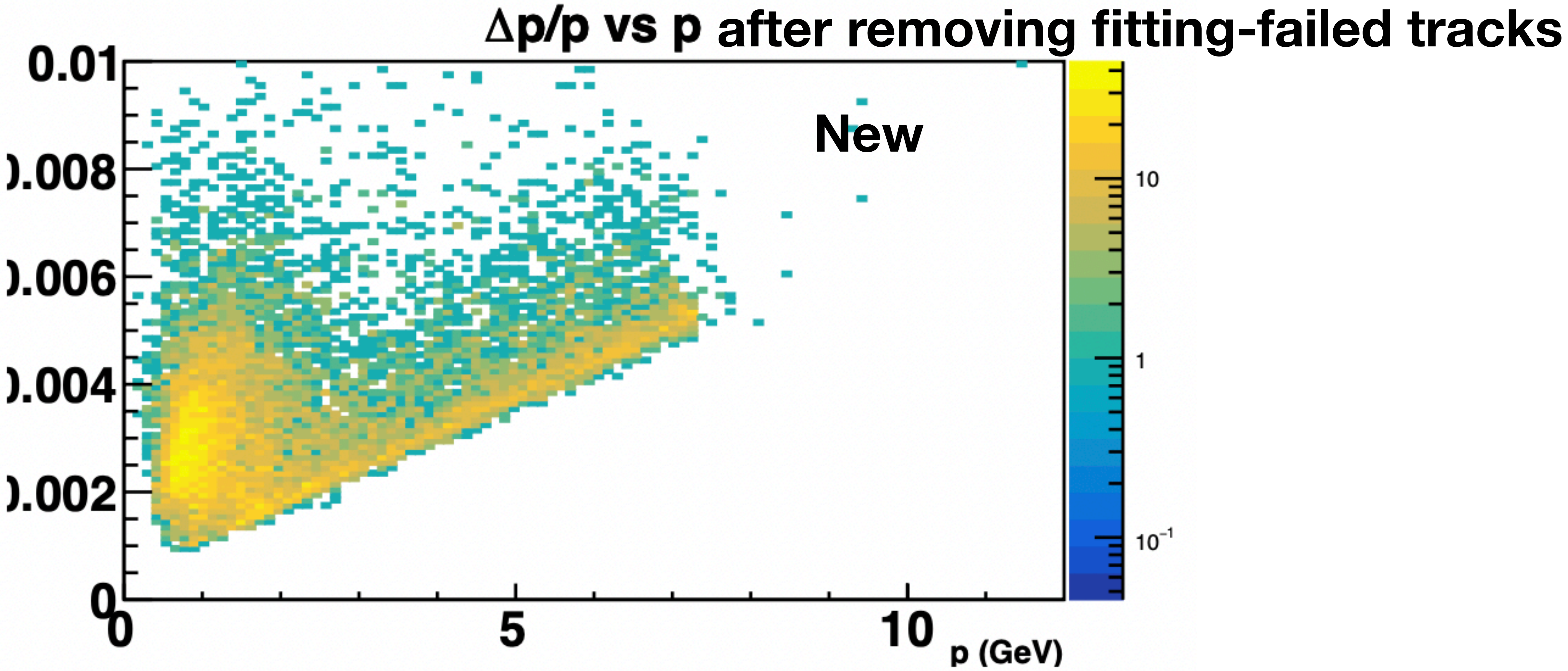
$$\Delta p = p^2 \Delta Q = p^2 \sqrt{C55}$$

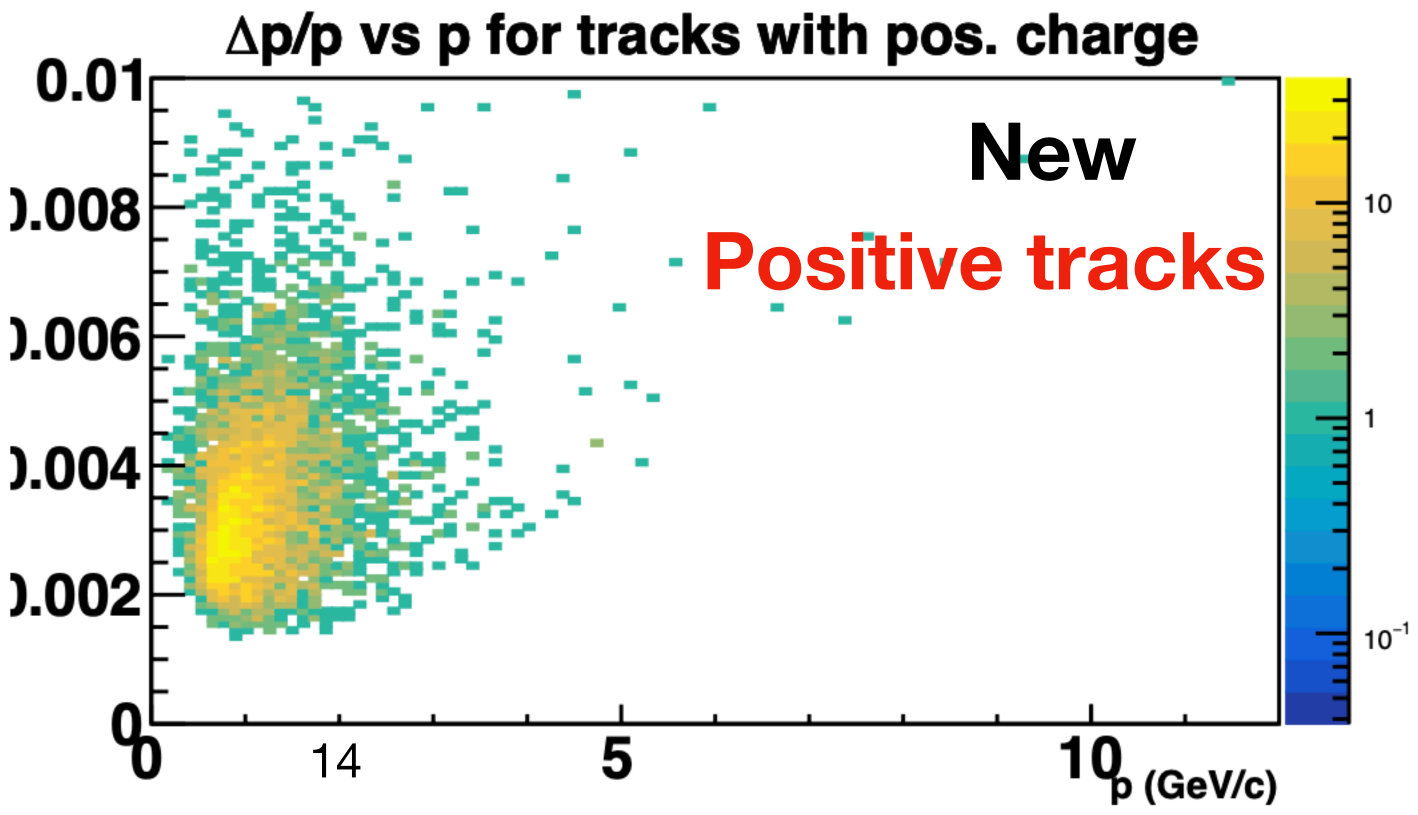
```
if (TBT == true) {
    if (chi2 > initChi2) { // fit failed
        this.finalStateVec = this.initialStateVec;
        sv.trackTrajT.put(svzLength - 1, value: this.initialStateVec);
        this.calcFinalChisq(sector, nofilter: true);
    }
}
```

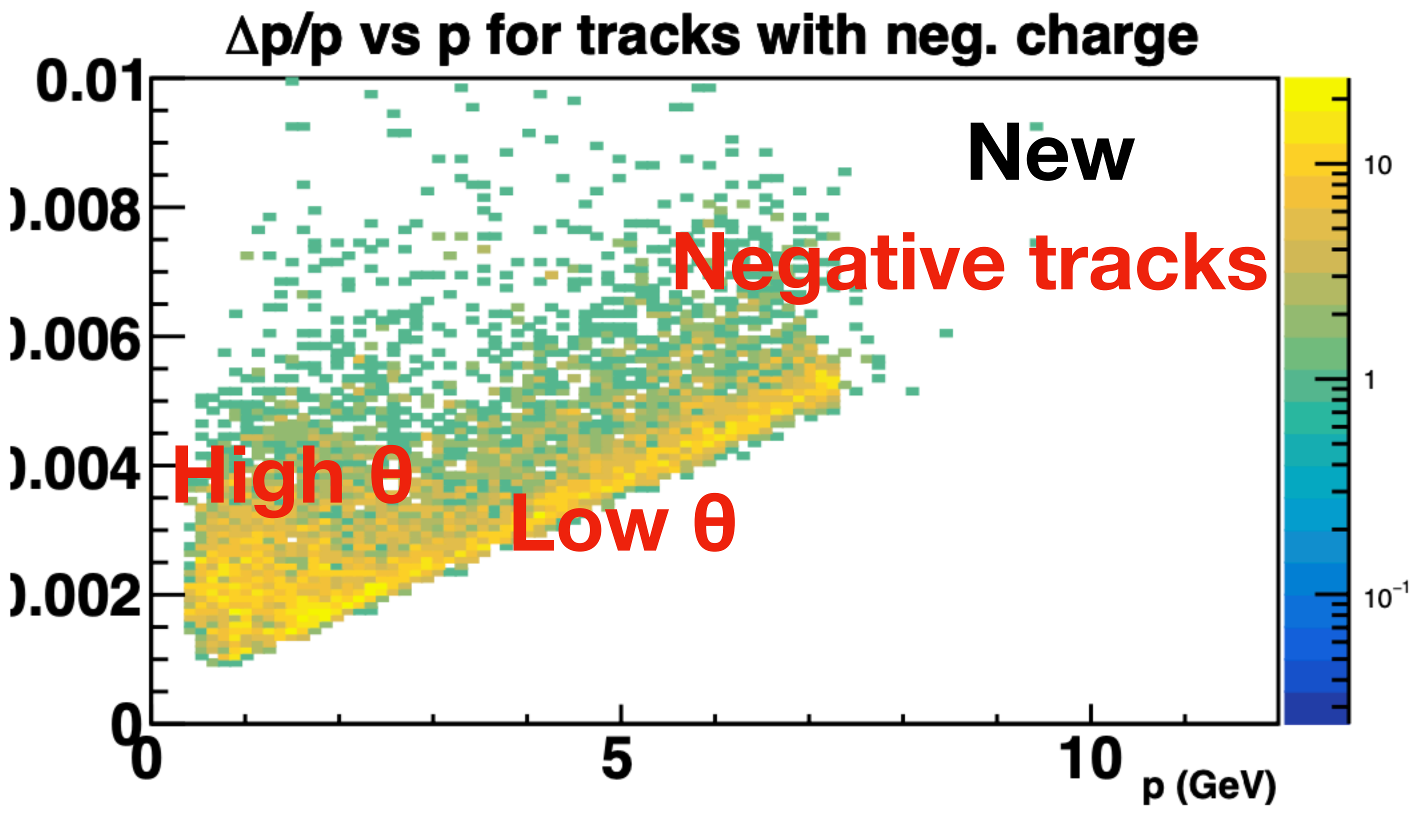
If TB tracking by KF is failed, TB final state is set as initial state, where initial covariance matrix is constant.











Introduction to DAF

- DAF is a proper way to handle hit ambiguity, noise hits, and DC layers with double hits based on papers [1, 2].
- After tracking at one iteration, each measurement is assigned a weight based on its χ^2 calculated by its residual and error. An effective hit for each DC layer with one or double hits is calculated based on measurements and weights. Effective hits are applied into next iteration. As tracking results are updated along iterations, weights and effective hits are updated. Originally, we set weight as 1 for single hit, and (0.5, 0.5) for double hits on layers.
- Referring to a thesis[3] for the HADES experiment, definitions of effective hits and weights are as follows. There are two important parameters in formulas for weights: cut-off parameters χ^2_{cut} and annealing factor T. The parameter χ^2_{cut} is equivalent to a χ^2 cut for low T. The annealing factor basically inflates the measurement errors. Along iterations, T changes from large to small, and stops at 1[1].

$$\bar{\boldsymbol{m}}_k = \bar{V}_k \left(\sum_{i=1}^{n} p_k^i \left(V_k^i\right)^{-1} \boldsymbol{m}_k^i \right)$$

$$\bar{V}_k = \left(\sum_{i=1}^{n} p_k^i \left(V_k^i\right)^{-1} \right)^{-1}$$

$$p_k^i = \frac{\Phi_k^i}{\sum_{j=1}^{n} \Lambda_k^j + \Phi_k^j}$$

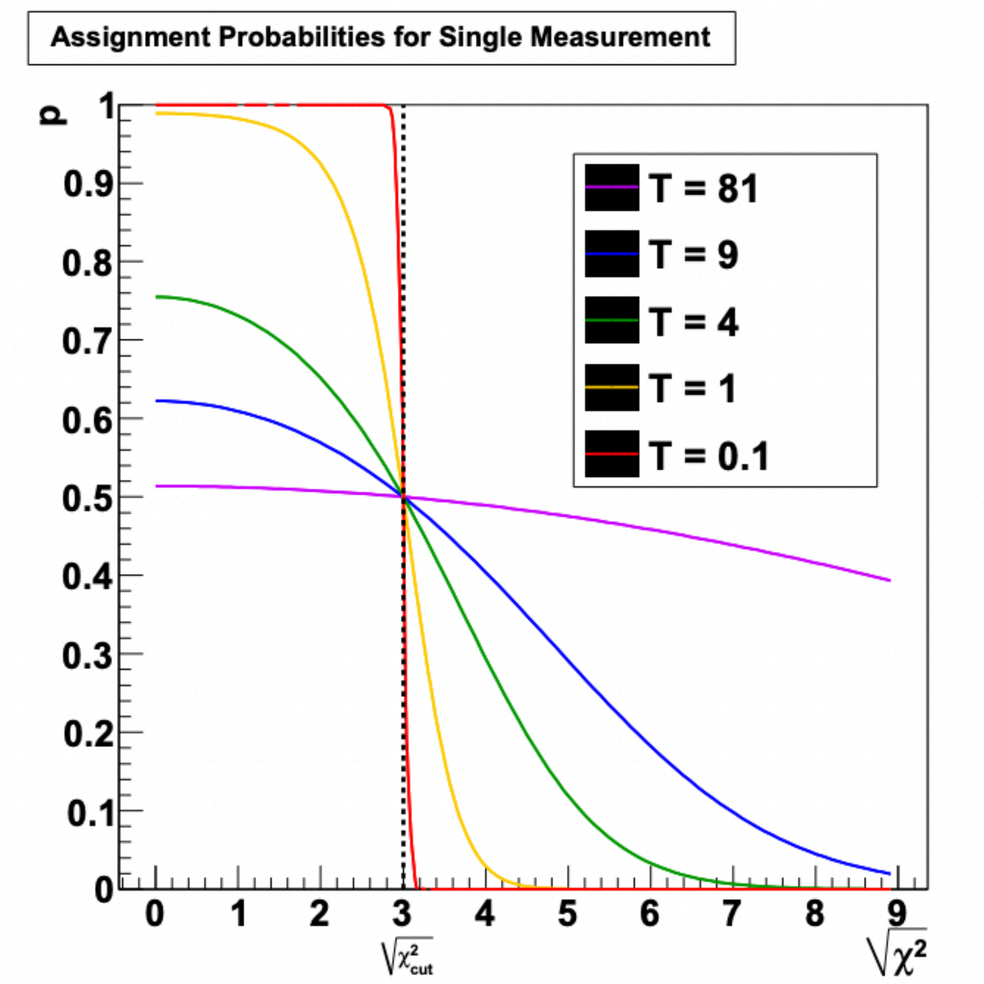
$$\Phi_k^i = \frac{1}{(2\pi)^{\frac{r}{2}} \sqrt{T \cdot |V|}} \cdot \exp\left(-\frac{1}{2T} \left(\boldsymbol{m}_k^i - \boldsymbol{h}_k\left(\boldsymbol{x}_k^n\right)\right)^T \left(V_k^i\right)^{-1} (\dots) \right)$$

$$\Lambda_k^i = \frac{1}{(2\pi)^{\frac{r}{2}} \sqrt{T \cdot |V|}} \cdot \exp\left(-\frac{\chi^2_{\text{cut}}}{2T} \right)$$

χ^2

Cut-off parameter

Annealing factor



[1] R. Frühwirth, A. Strandlie, Application of adaptive filters to track finding, Nuclear Instruments and Methods in Physics Research A, 2006
[2] R. Frühwirth, A. Strandlie, Track fitting with ambiguities and noise: A study of elastic tracking and nonlinear filters, Computer Physics Communications, 1999
[3] Erik Krebs, Application of a Kalman filter and a Deterministic Annealing filter for track reconstruction in the HADES experiment, 2012

Definition of Effective Doca for Single Hit

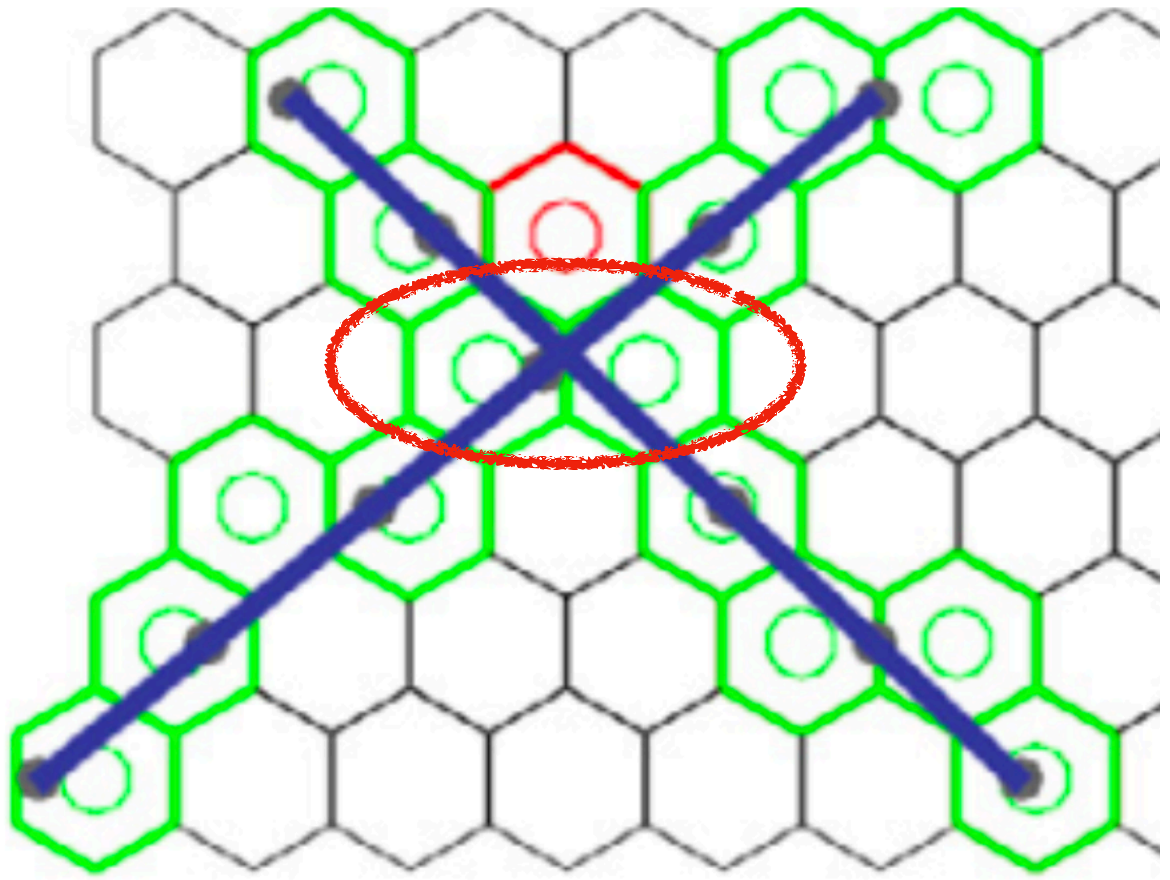
$$doca_{eff} = doca \qquad V_{eff} = V/p$$

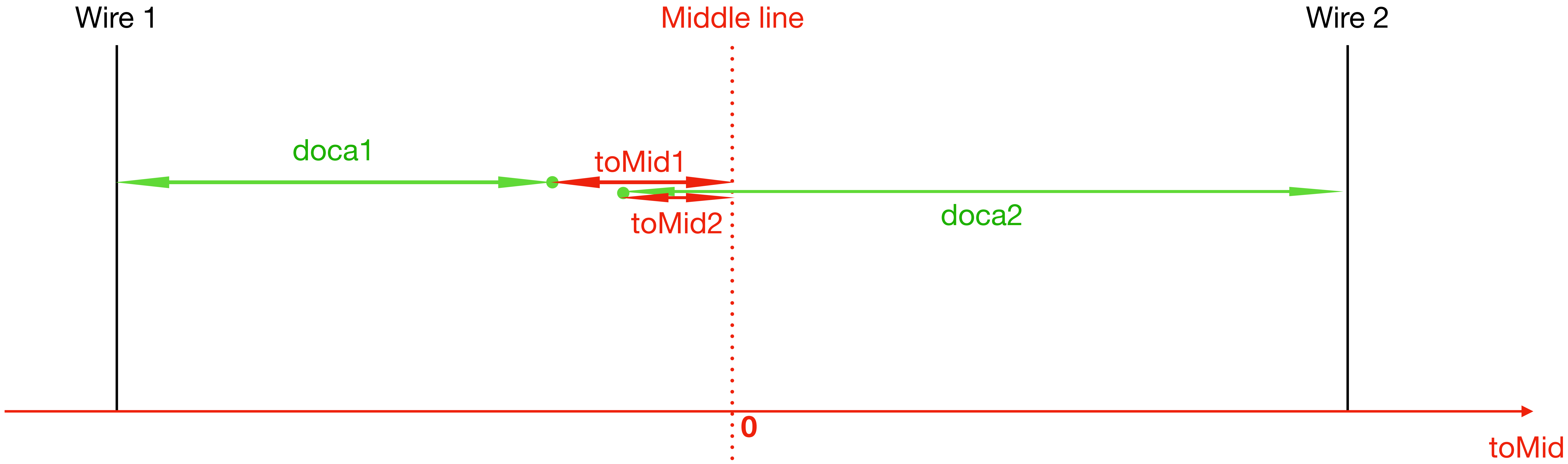
- Finally, when $T = 1$, weight p is close to 1 for real hits, and close to 0 for noise hits.
- It let V_{eff} very large for noise hits. Further it let filtered CM close to CM from propagation, and Kalman gain matrix close to 0.
- It means that DAF let effect of noise hits on tracking by Kalman Filter negligible.

Filtered CM: $$C_k = \left[\left(C_k^{k-1}\right)^{-1} + H_k^T \left(V_k\right)^{-1} H_k \right]^{-1}$$

Kalman gain matrix: $$K_k = C_k H_k^T \left(V_k\right)^{-1}$$

Definition of Effective Doca for Double Hits





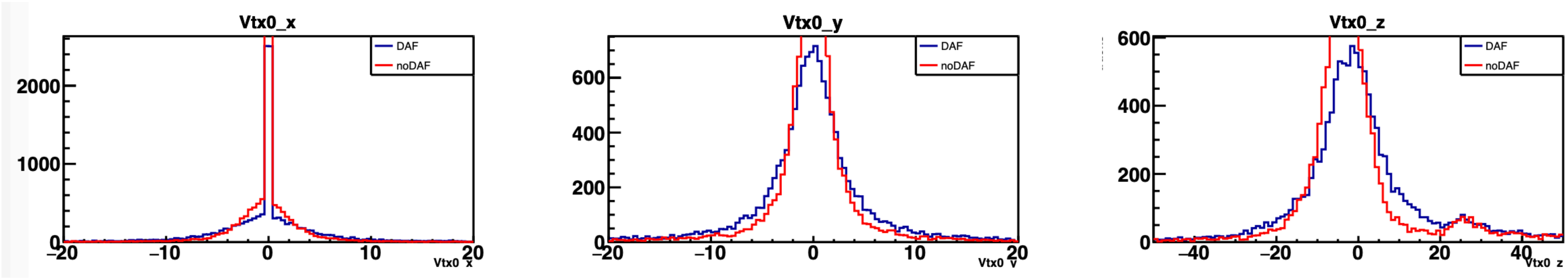
- Docas for double hits are referred to two neighbored wires, so effective doca can not be directly expressed. How to calculate effective doca:
  - For each hit, a weight is assigned at the previous iteration.
  - Based on docas, distances to the middle line is calculated with sign. Errors of distances are the same as errors of docas.
  - Input distances and weights into DAF formulas to calculate an effective distance to the middle line.
  - Finally, calculate an effective doca referring to a wire with higher weight. (Actually, it is equivalent to choose any of the two wires as a reference line).
- About effective doca:
  - If both hits are real, a proper effective hit is calculated and applied into tracking.
  - If one hit is noise, its weight is much less than the other one. Then its contribution on effective doca is much less than the other one.
  - If both hits are noise, both weights are very small. Like noise single hit, their effects on Kalman Filter are negligible.

Run 5700

Challenge for DAF Applied into HB Tracking

DAF does not work well for HB tracking due to two reasons:

- HB doca is rough.
- Initial state by the pattern recognition is rough.



Pure muon sample

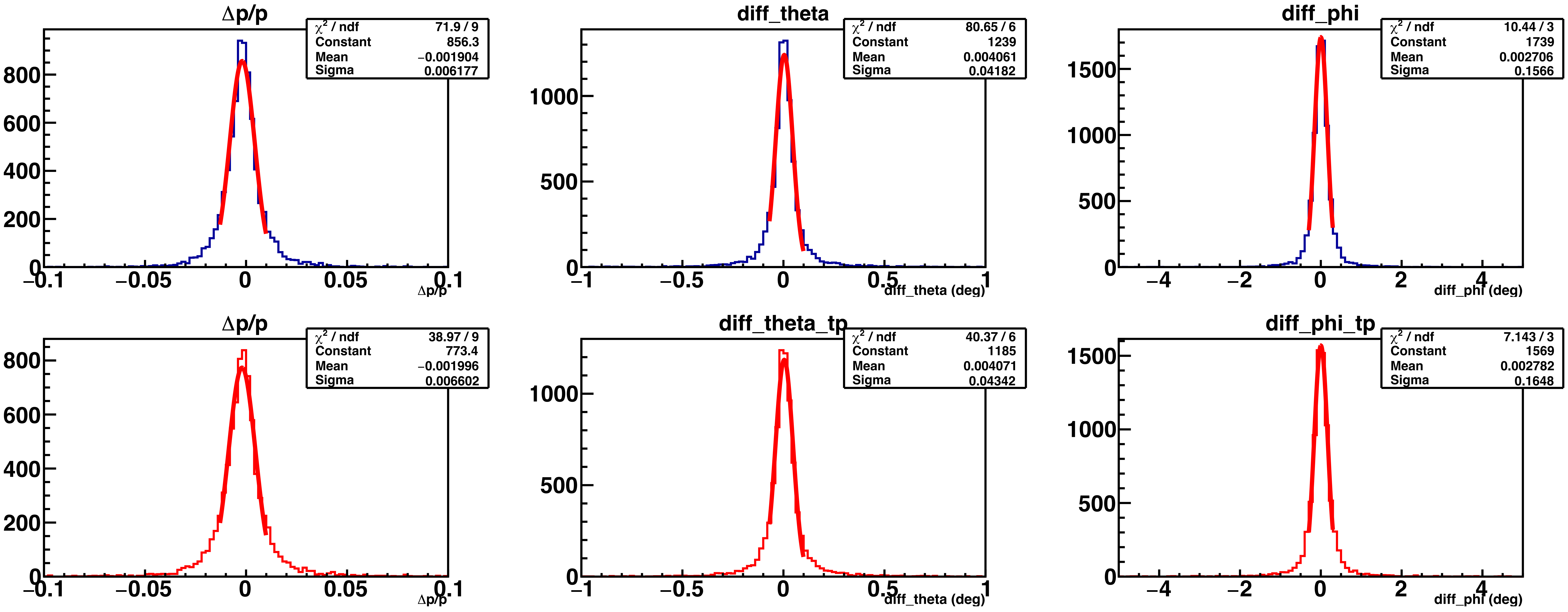
DAF for TB Tracking

- Temporarily, $\chi^2_{cut} = 8$, $T = (81, 9, 4, 1, 1, \ldots)$.
- The parameters will be tuned.

Momentum Difference between tracking results and truth

With DAF

No DAF

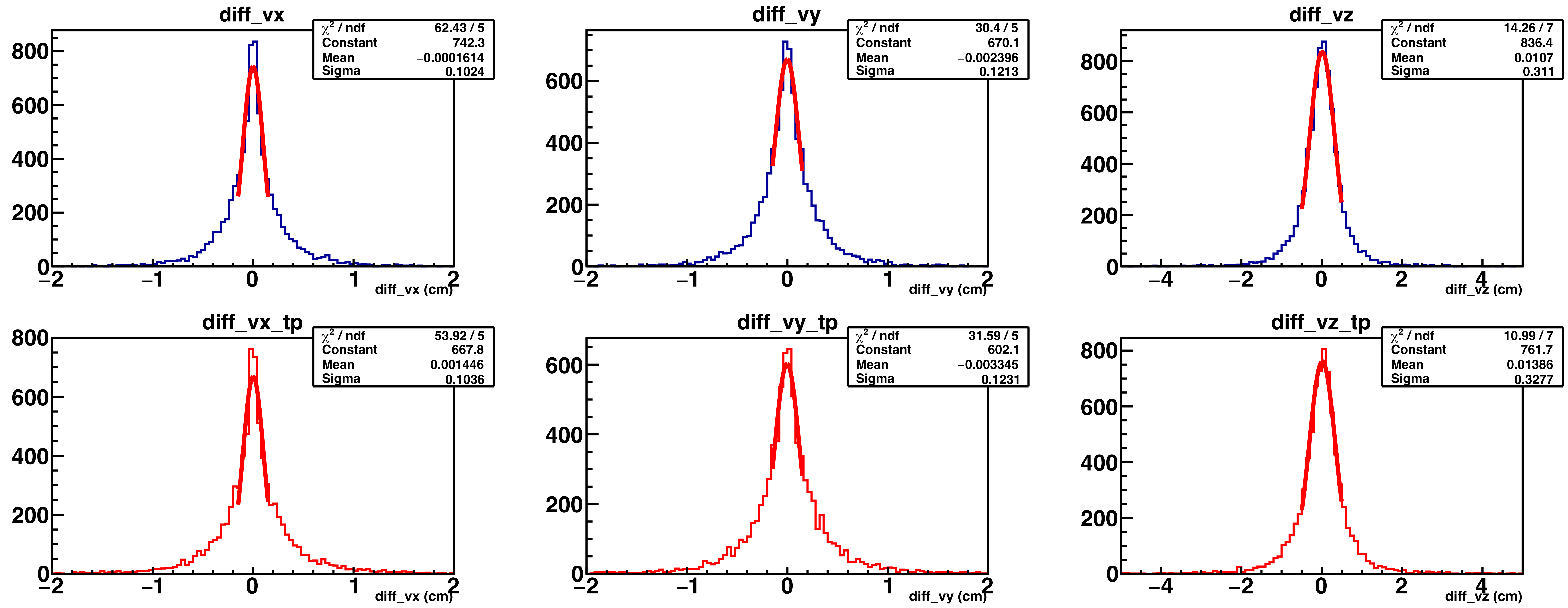


Ratio of σ = 93.6% **Ratio of σ = 96.3%** **Ratio of σ = 95.0%**

Vertex Difference between tracking results and truth

With DAF

No DAF



Ratio of σ = 98.8% **Ratio of σ = 98.5%** **Ratio of σ = 94.9%**

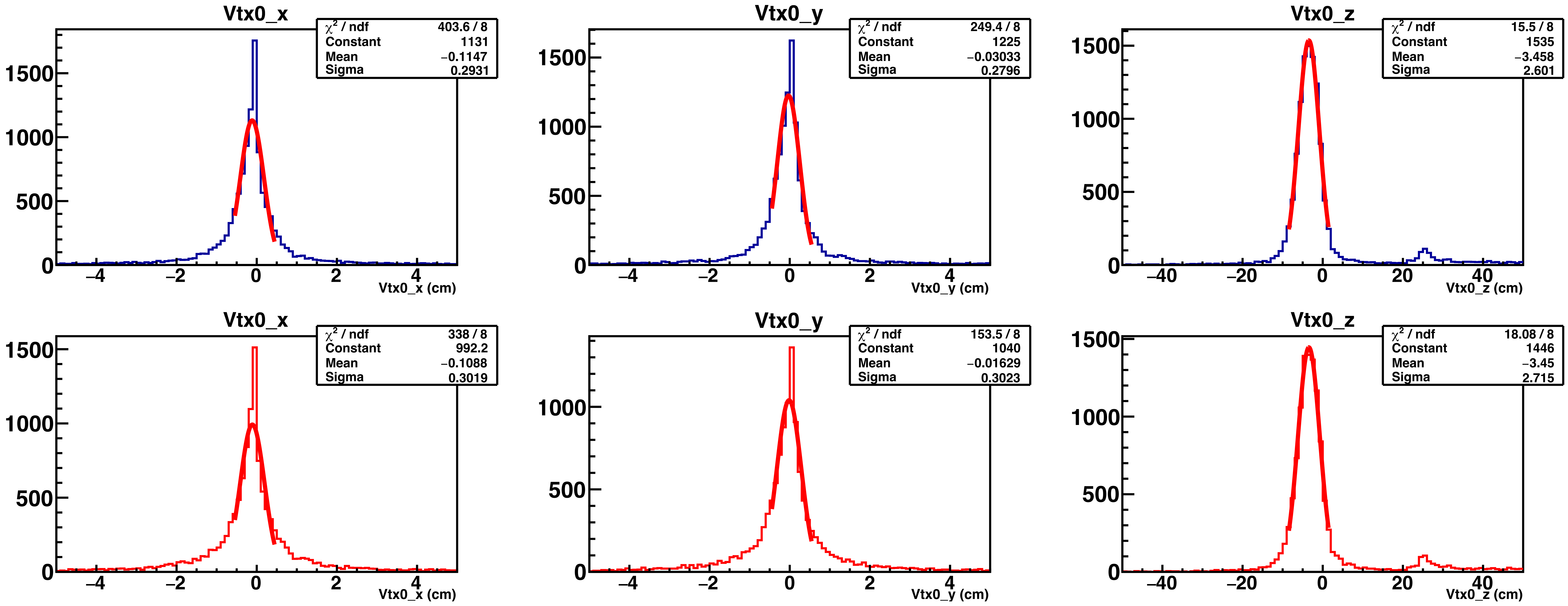
- Resolutions for both momentum and vertex are improved.
- For the pure MC sample without background, the main reason for resolution improvement is that DAF properly handles cases with double hits.

Run 5700

Vertex for Data

With DAF

No DAF

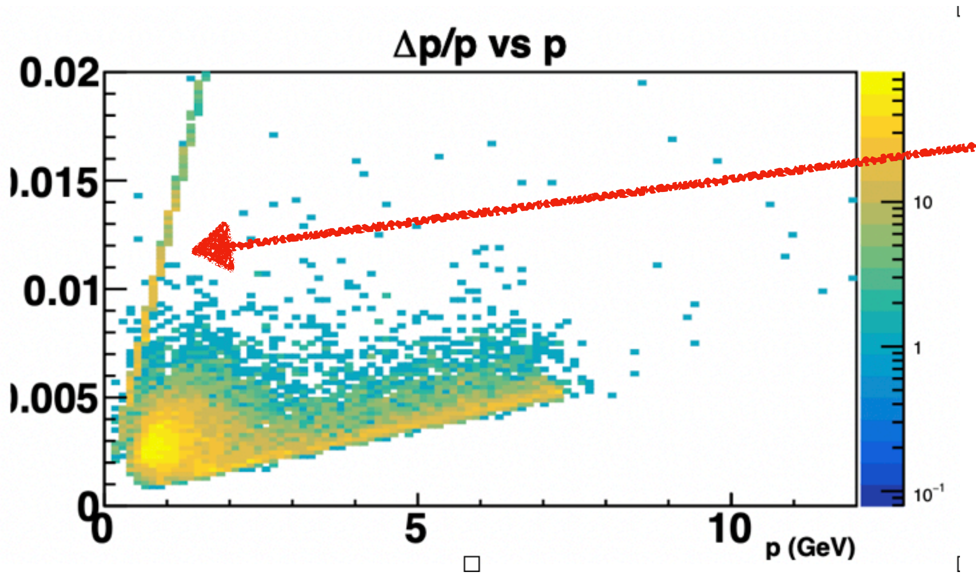


Ratio of σ = 97.1% **Ratio of σ = 92.5%** **Ratio of σ = 95.8%**

Vertex resolution is improved.

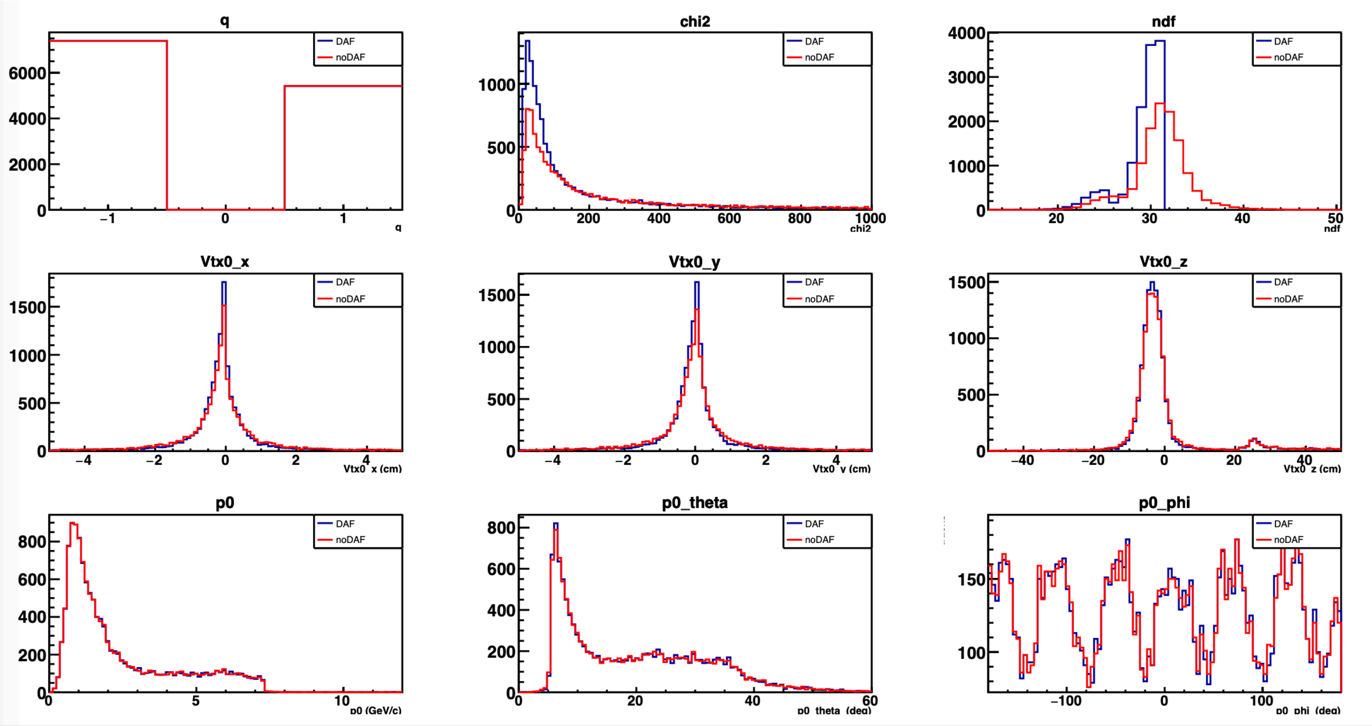
Run 5700

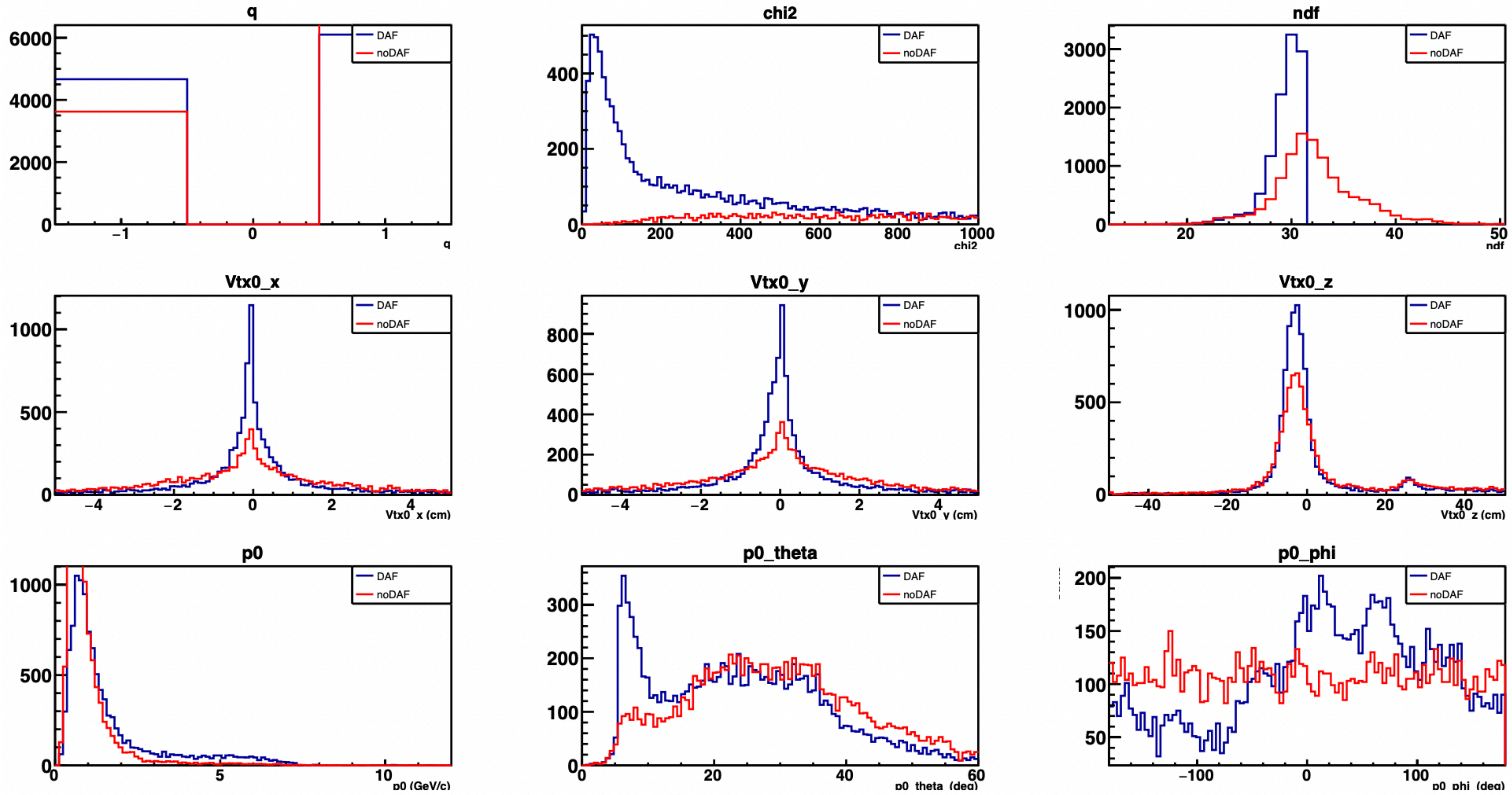
Reconstruct Failed Tracks in the Old Package with DAF



~8% failed Tracks in the old package in low-luminosity run 5700
(Conventional tracking without AI-denoising)

All Tracks





DAF saves plenty of failed tracks in the old package since effect of noise hits is well degraded by DAF.

Demonstration of a Track, Failed in the Old Package, but Successful with DAF

No DAF

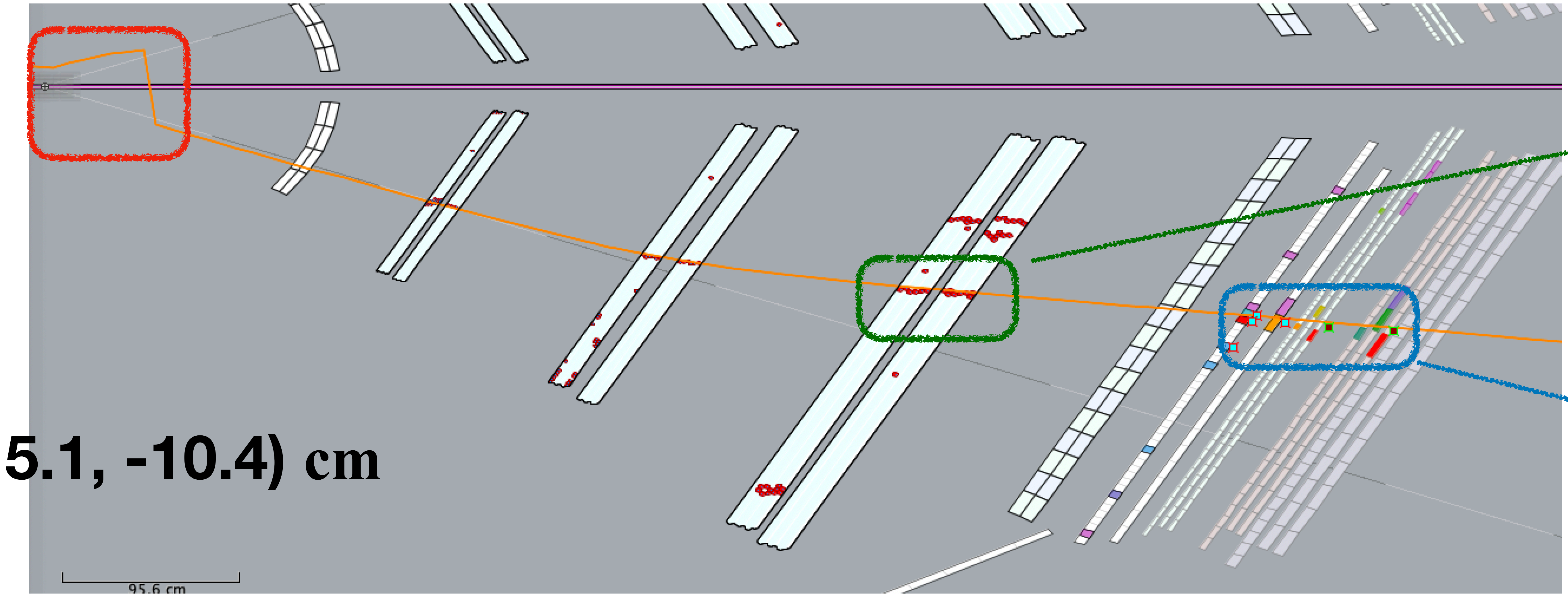
χ^2 = 4834.2

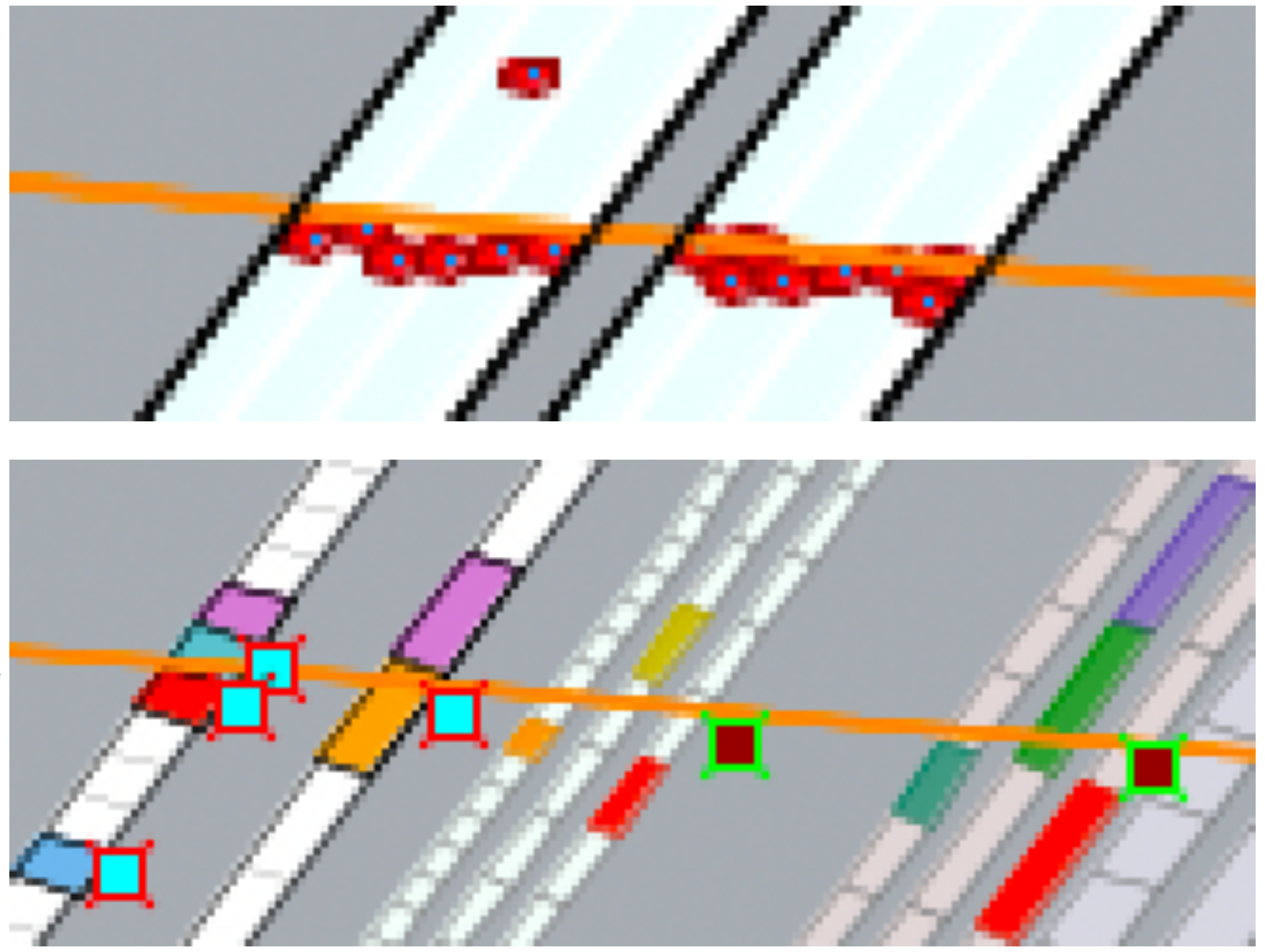
p = 1.29 GeV/c

θ = 32.0 deg

φ = 149.9 deg

Vertex = (8.6, 15.1, -10.4) cm





Track in the old package shifts upwards.

With DAF

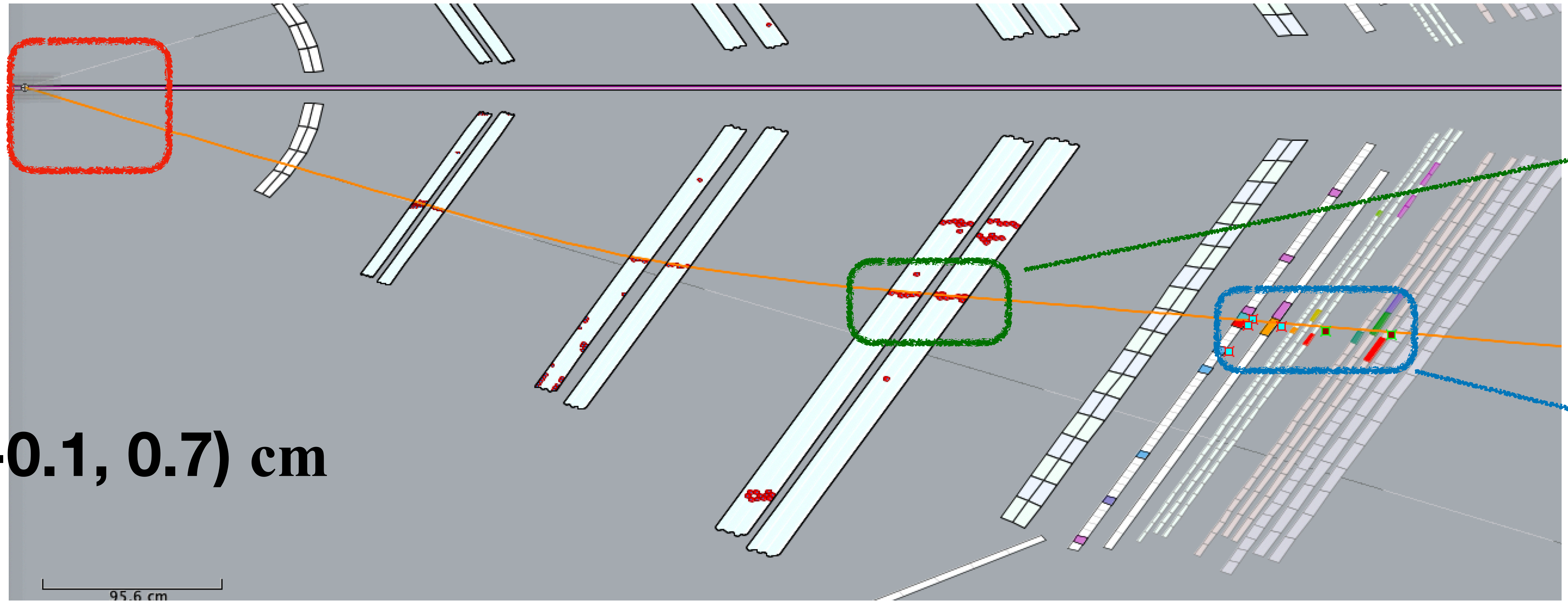
χ^2 = 40.8

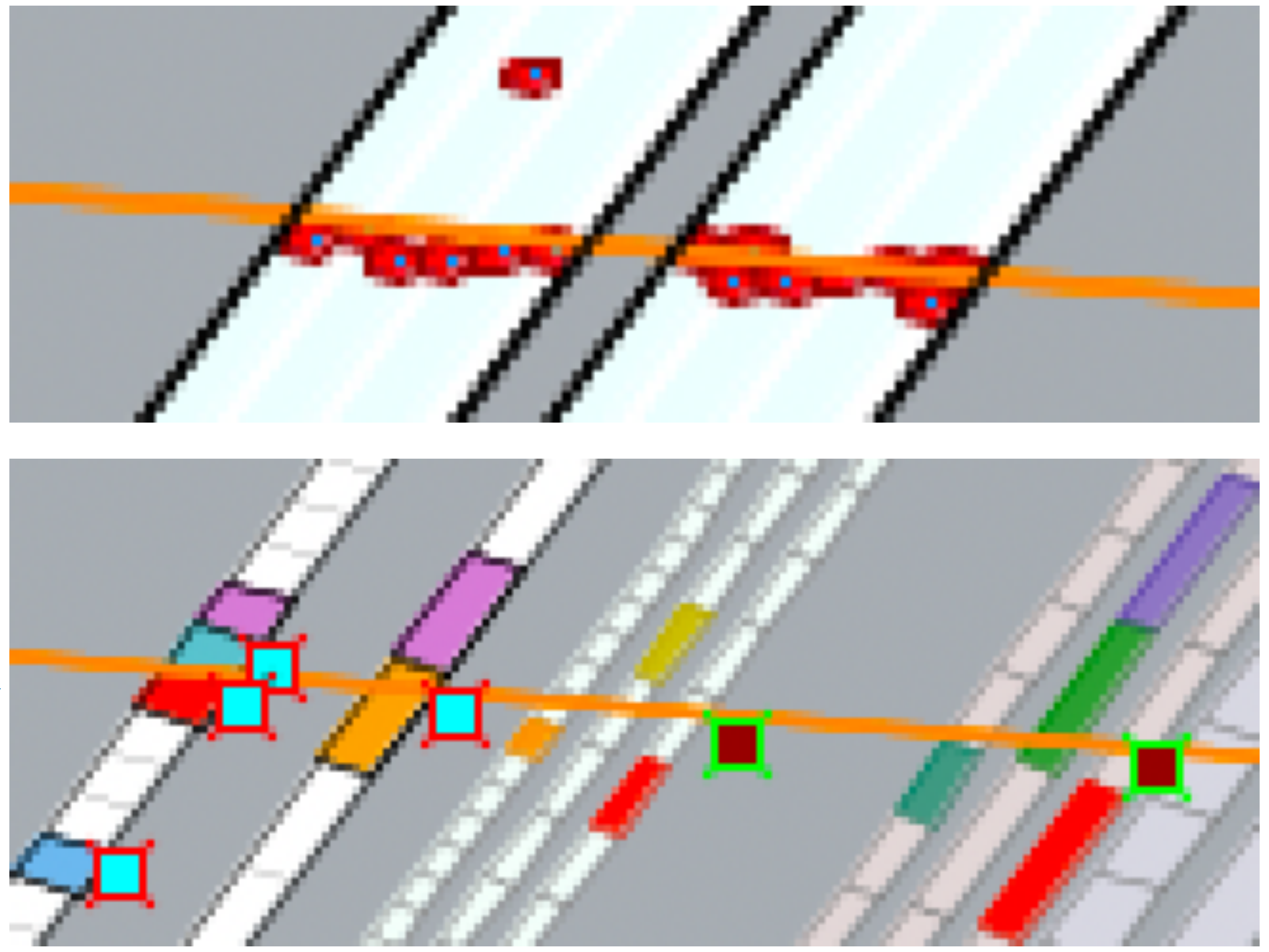
p = 1.20 GeV/c

θ = 26.1 deg

φ = 153.7 deg

Vertex = (-0.2, -0.1, 0.7) cm





Summary and Next Step

- Resetting initial CM brings benefits:
    - Tracking is well converged along iterations.
    - Reasonable pull and covariance matrix are obtained.
    - CPU efficiency is significantly improved.
- Since DAF properly handles double-hit cases and well degrades effect of noise hits, it brings benefits:
    - Resolution is improved.
    - Very promising that DAF improves tracking efficiency.
- DAF project will be continued:
    - More detailed studies for how DAF affects tracking will be taken.
    - LR-ambiguity and error for DC hits will be revisited since they closely correlated with DAF.
    - Parameters of χ^2_{cut} and T will be tuned to optimize resolution and tracking efficiency.