Greedy ambiguity resolution solver in EICrecon

Joint track and vertex reconstruction and tracking WG meeting

Minjung Kim (UC Berkeley / LBNL / CFNS, Stony Brook U.)

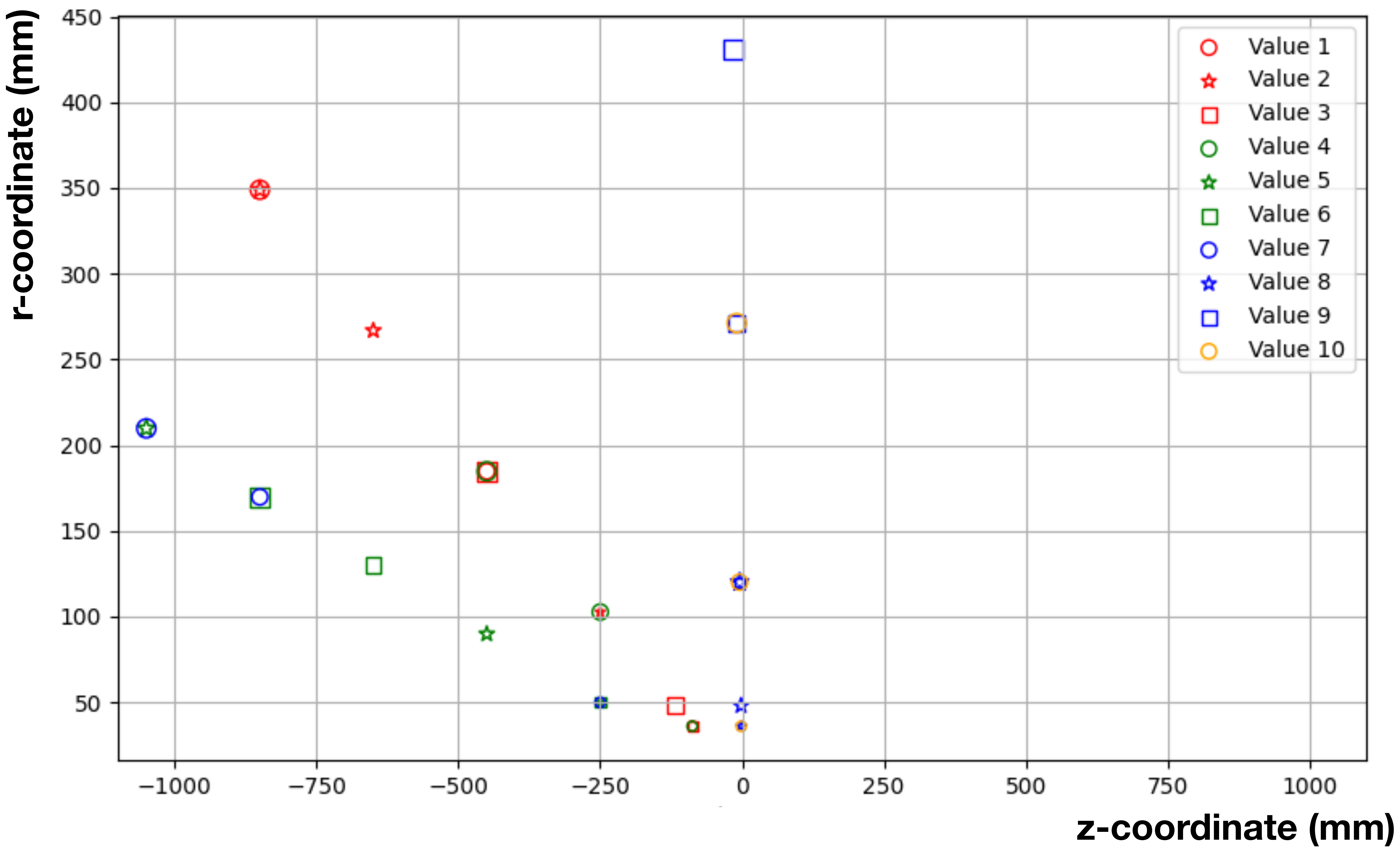
18. Apr. 2024 (Thu.)

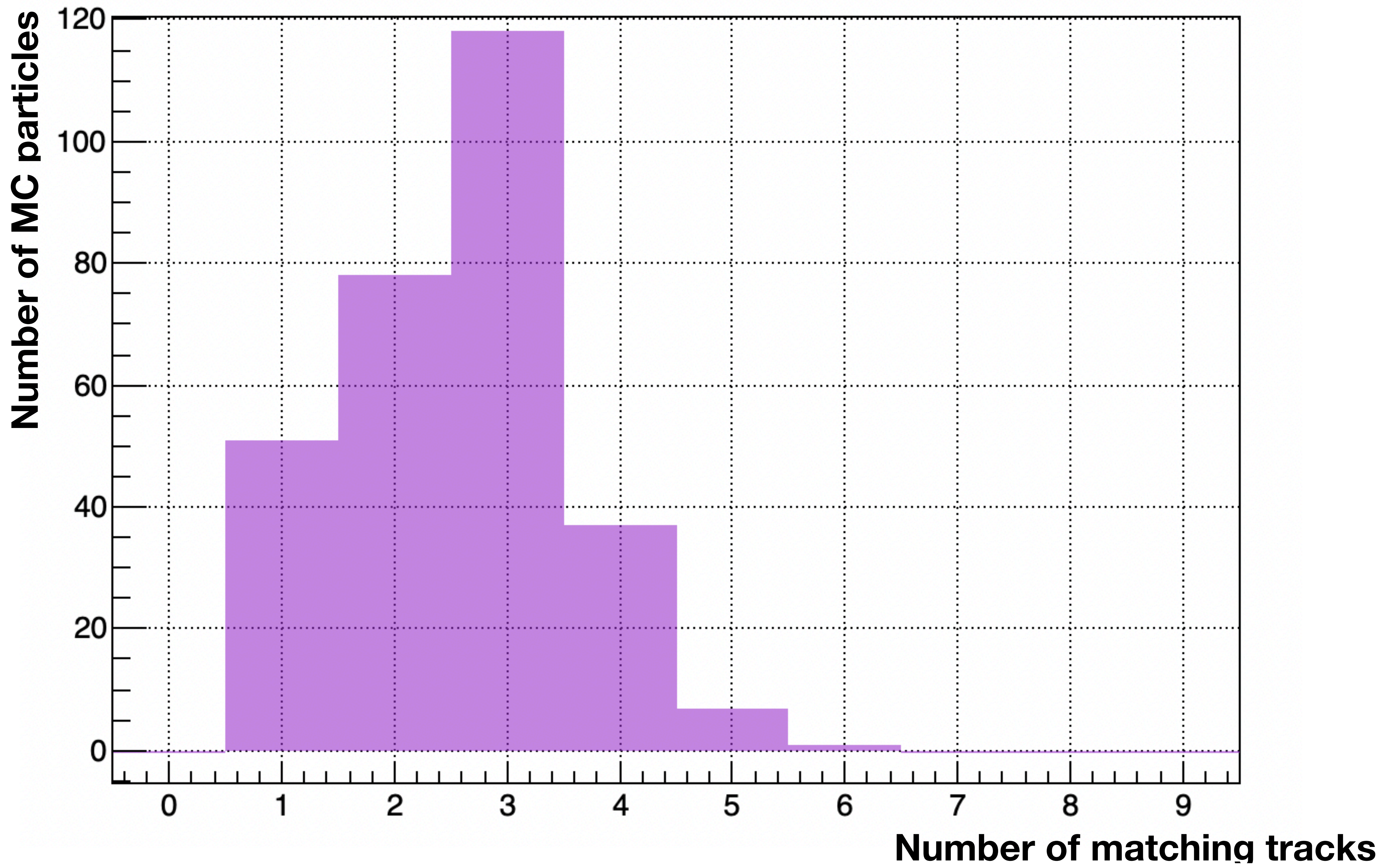
Duplicate tracks in realistic seeding

- Seed is made with 3 hits → several seeds can be found along one trajectory, leading duplicate tracks
- Correlation between a number of generated hits (N_{hit}) and a number of duplicate tracks (N_{trk}):

$$N_{trk} \propto {}_{N_{hits}}C_3$$

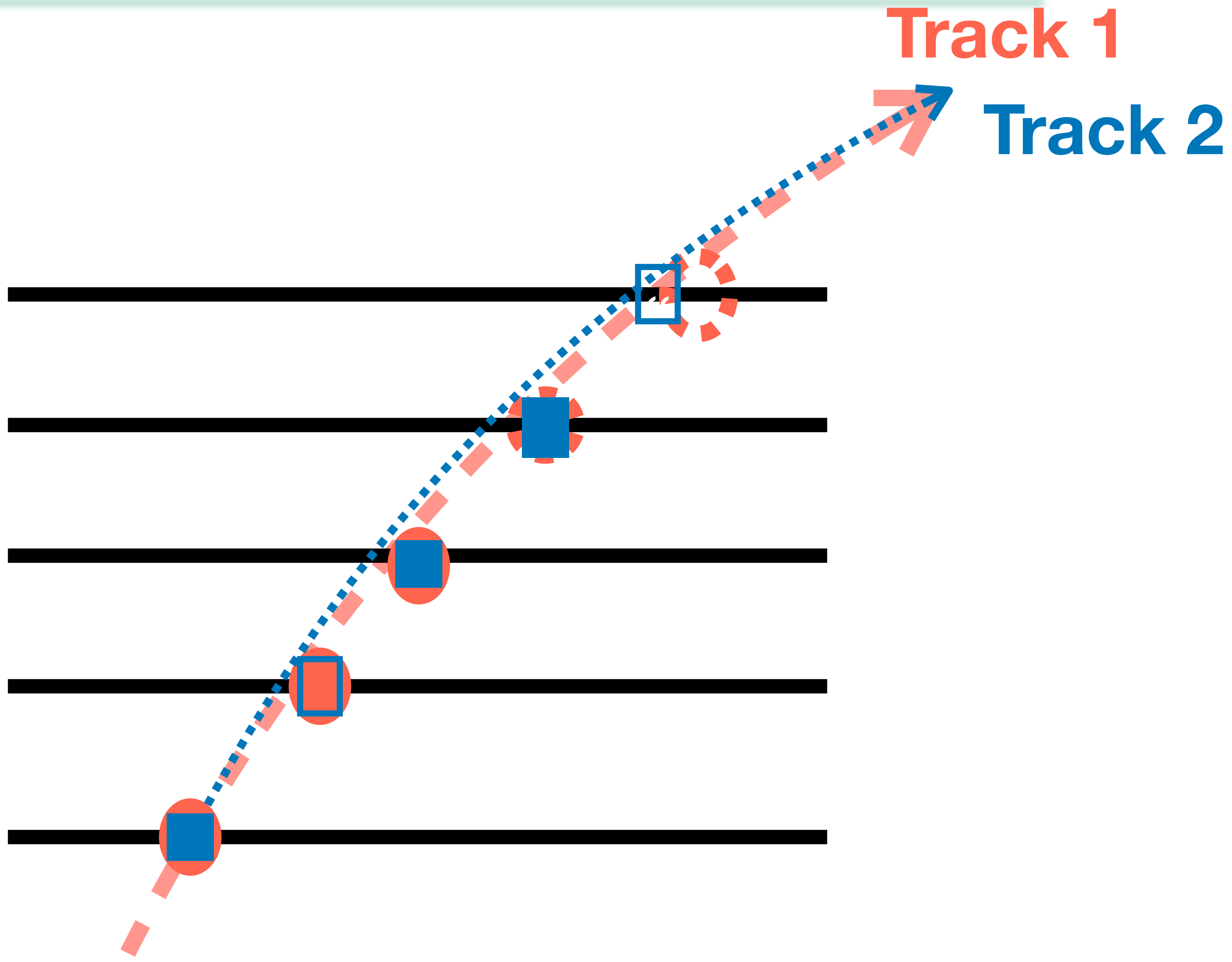
- Reduction by requiring using middle space point only once (as a middle space point), but still there





Greedy ambiguity resolution solver

- **Greedy ambiguity resolution solver**:
  1. Iterate trajectories and find the trajectory having number of shared hits larger than certain threshold
  2. Find the competetors and keep better quality trajectory only
  3. Repeat till you have trajectories having shared hits below certain threshold



- **Implementation to EICrecon**

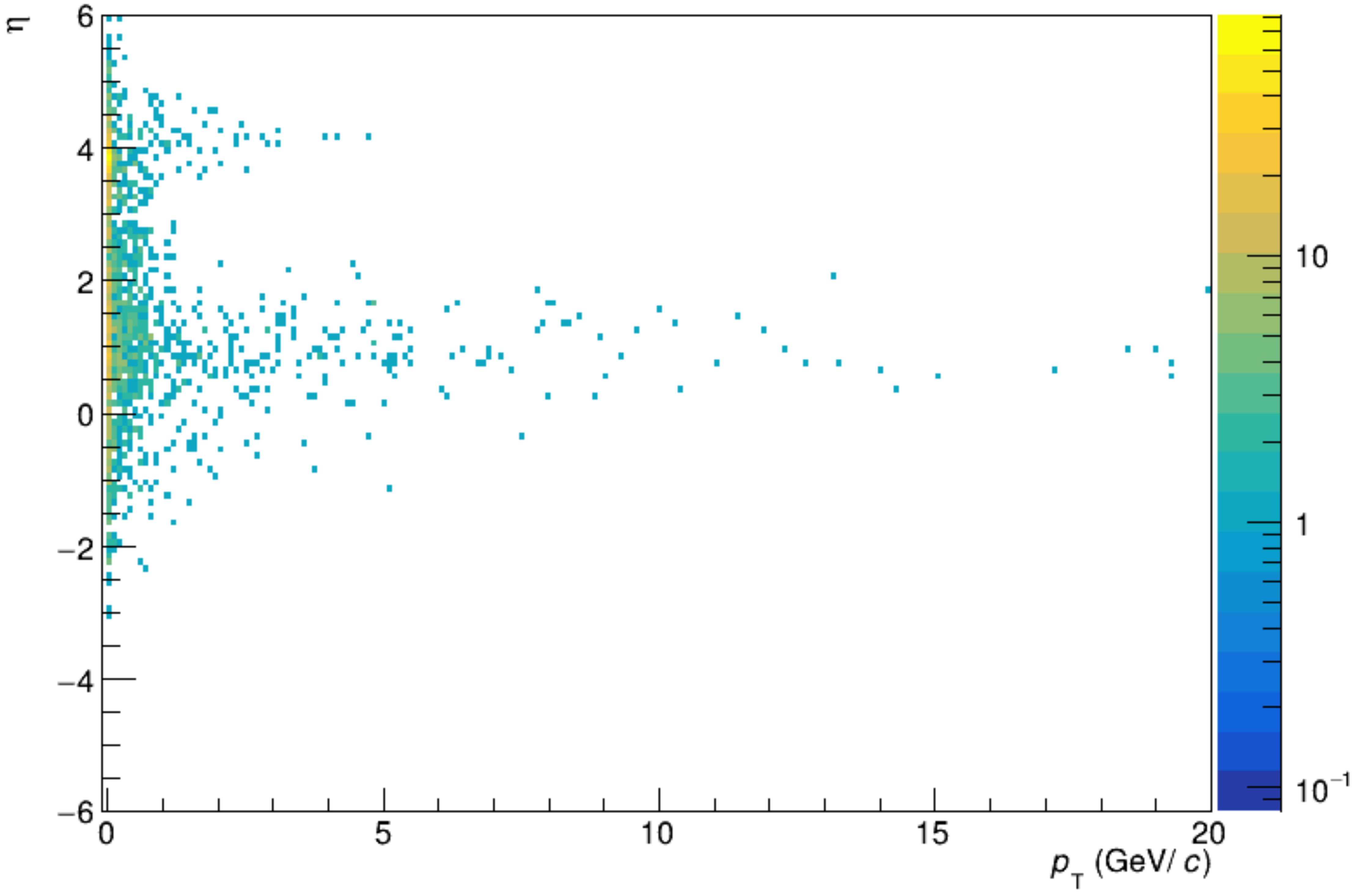
From ACTS: Core/include/Acts/AmbiguityResolution/GreedyAmbiguityResolution.ipp ➡Merge request ongoing!!

Performance QA setup

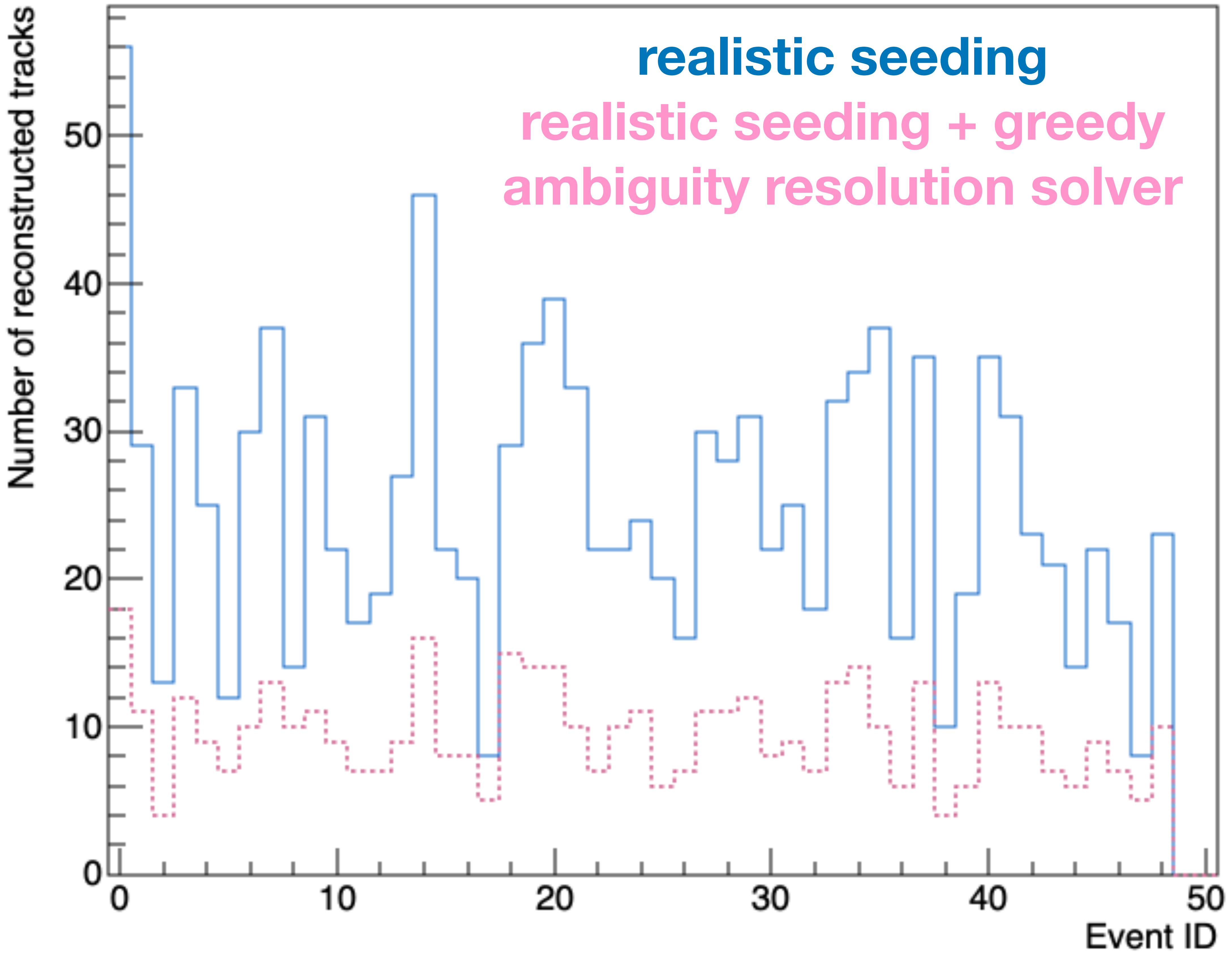
- ep collisions at 18x275 with minQ2=1000 + beam Effects
- 50 events simulated with “epic_craterlake.xml”
- Matching between MC particle and reconstructed track:

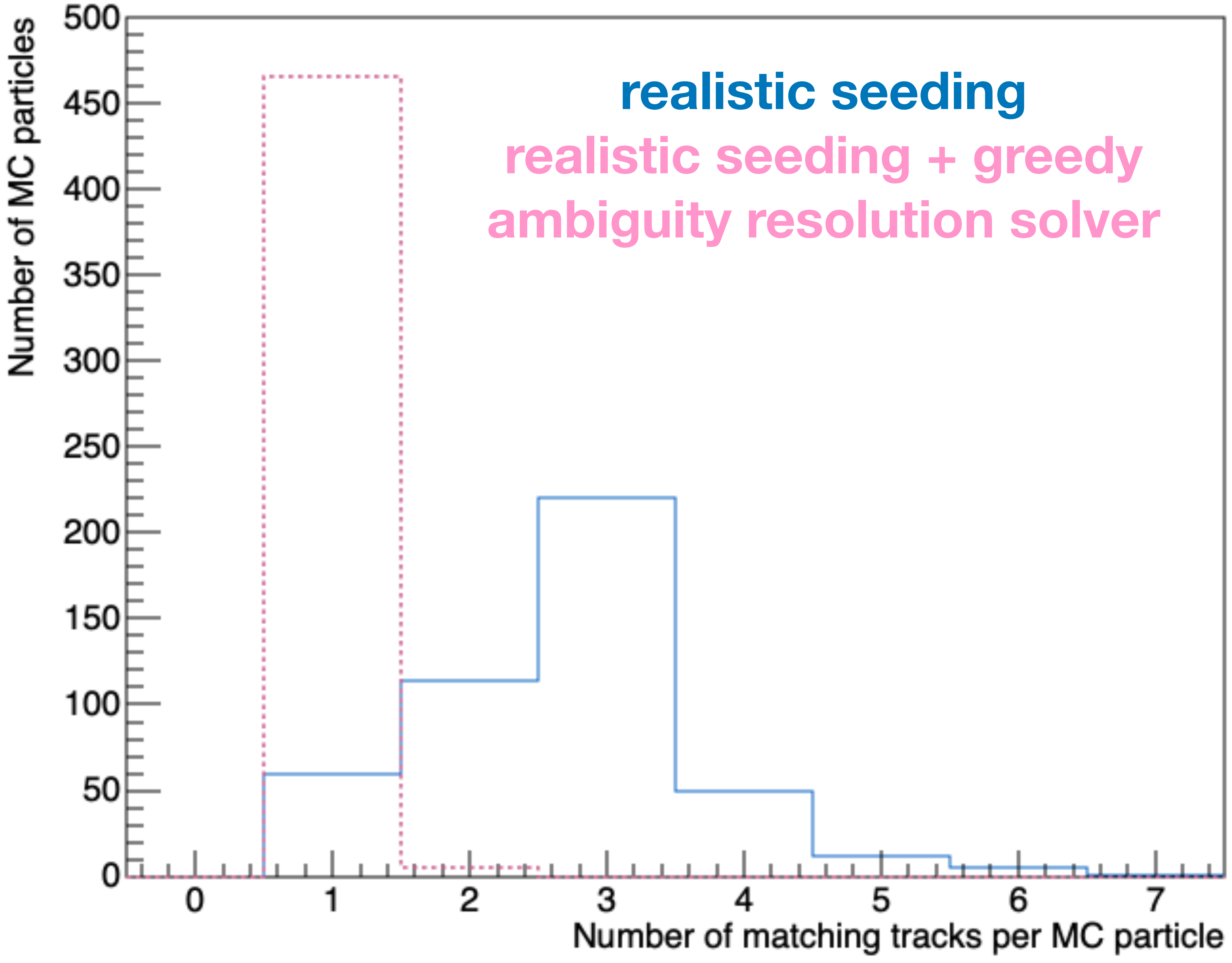
  Hit level matching: check the source of hits associated with the track and matching to the particle giving maximum contribution

charged particle kinematic distribution



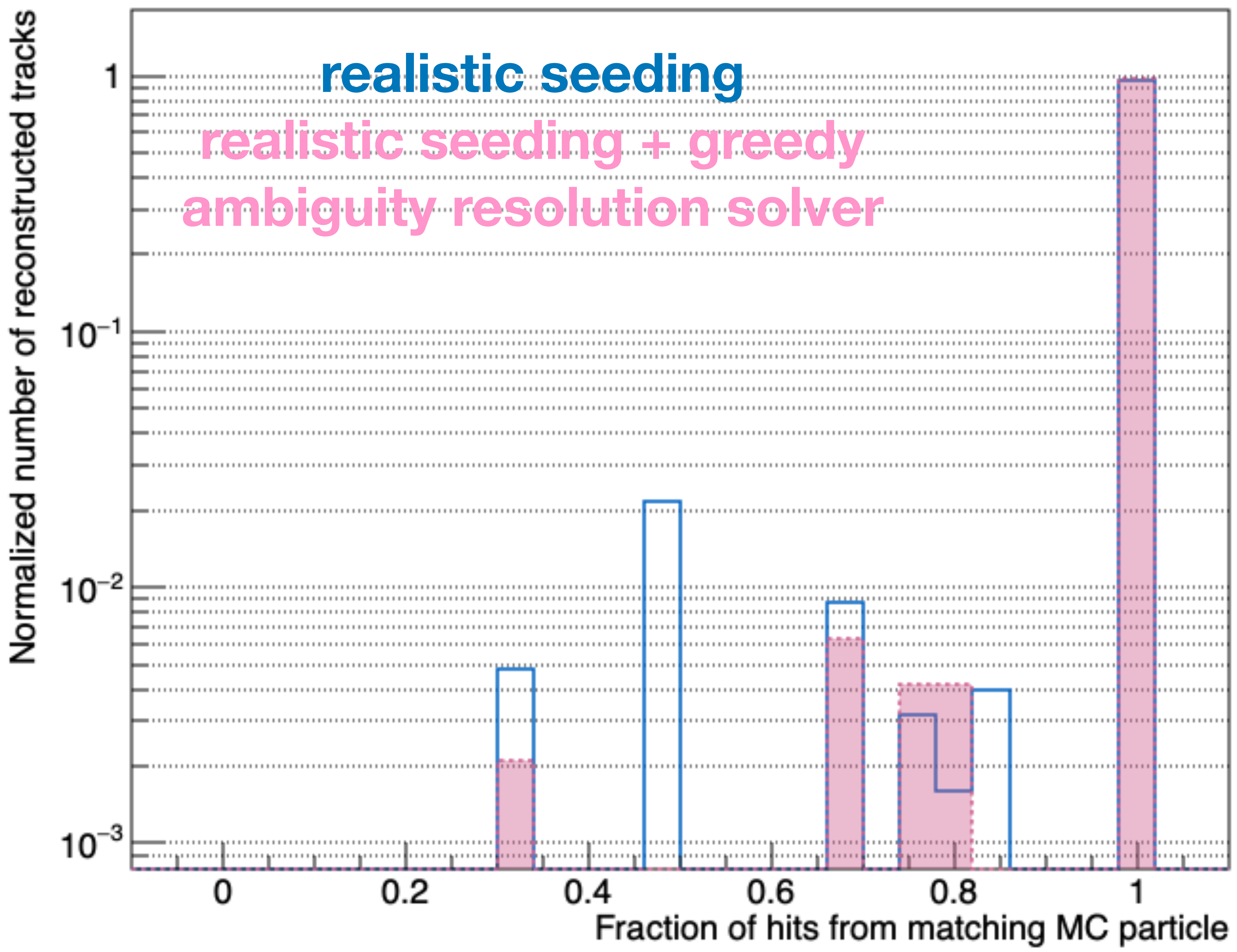
Impact of greedy ambiguity resolution solver

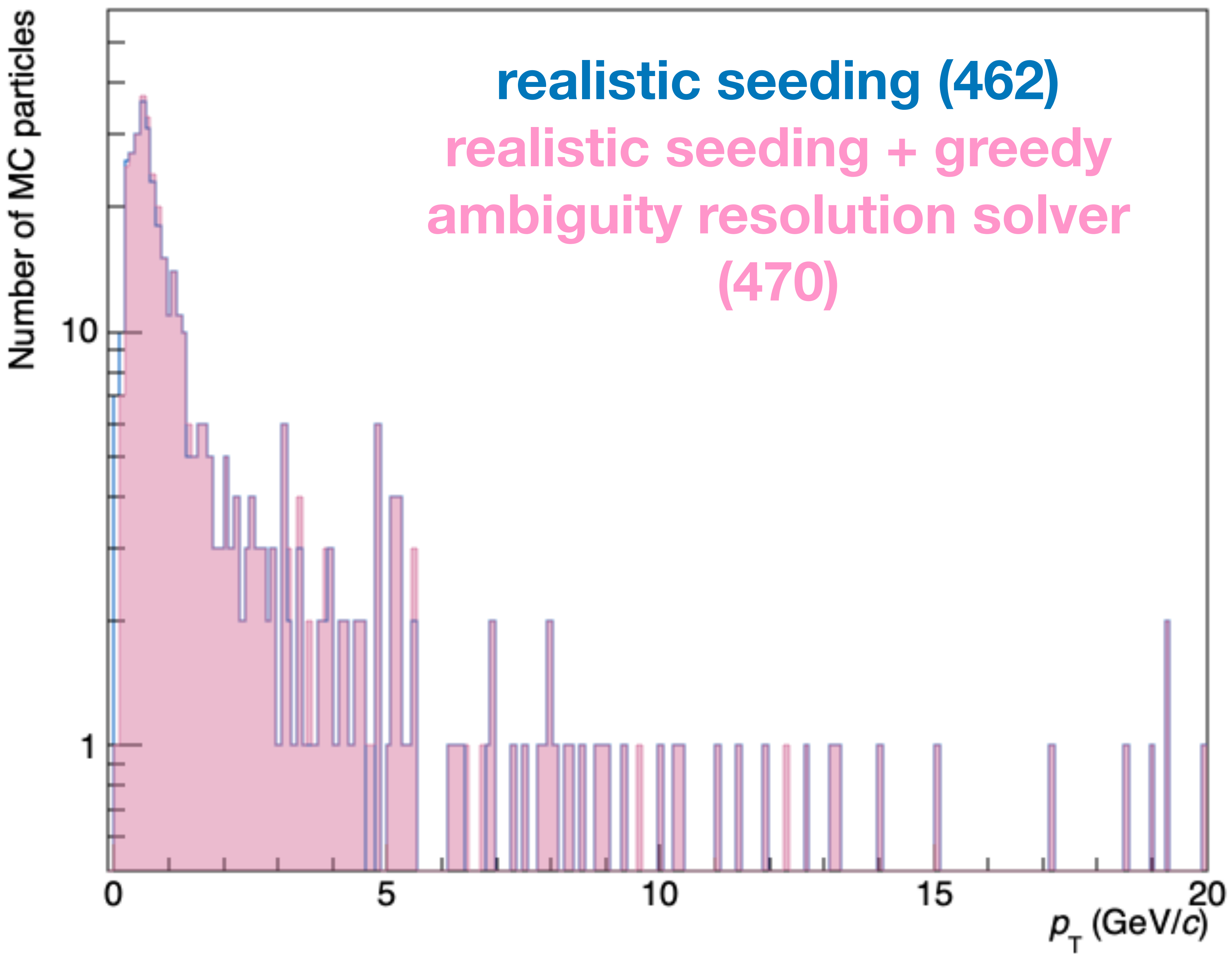




- In average, about 1/3 of tracks left after filtering using greedy ambiguity resolution solver
- Less than 1% of duplicate tracks found after filtering

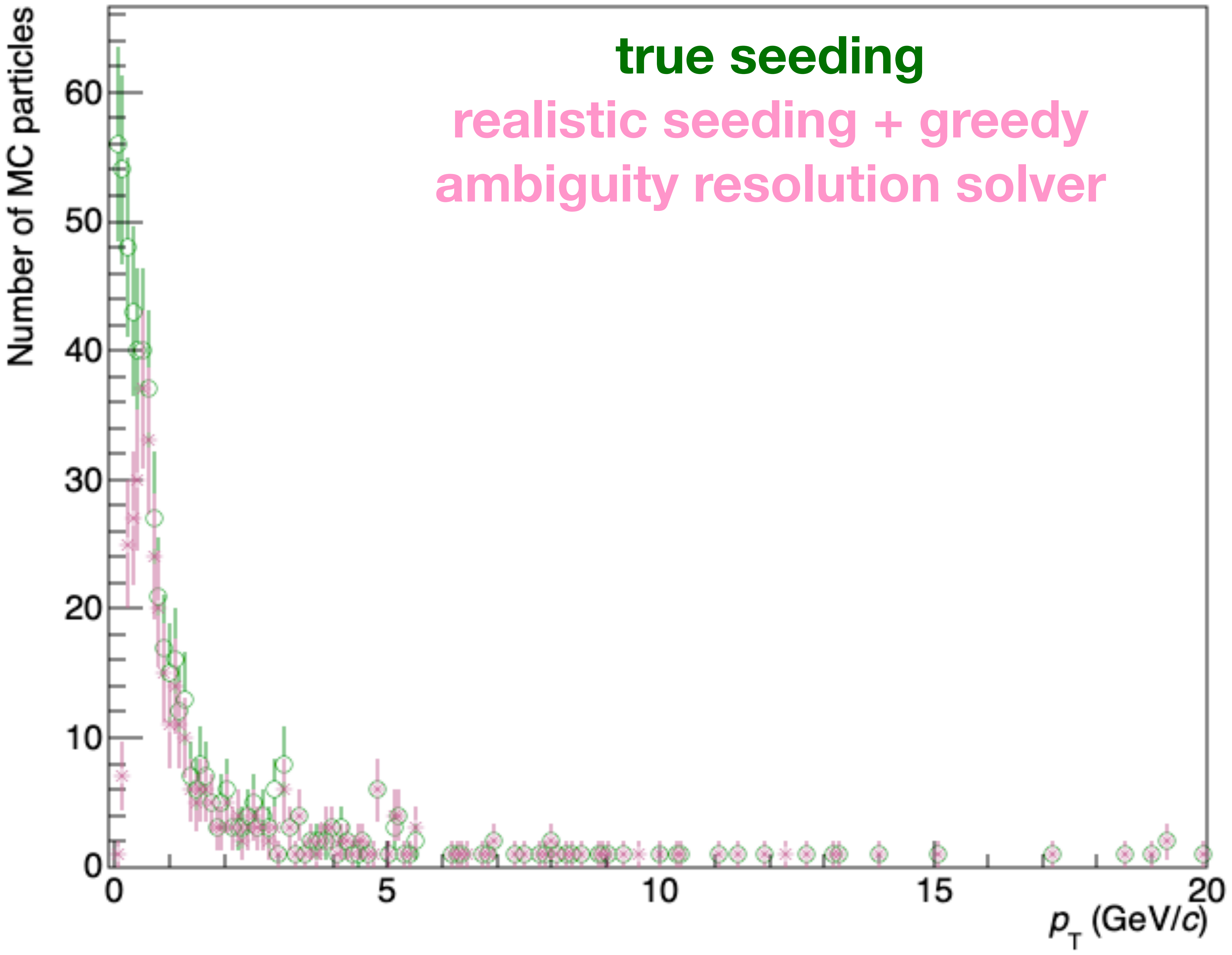
Impact of greedy ambiguity resolution solver

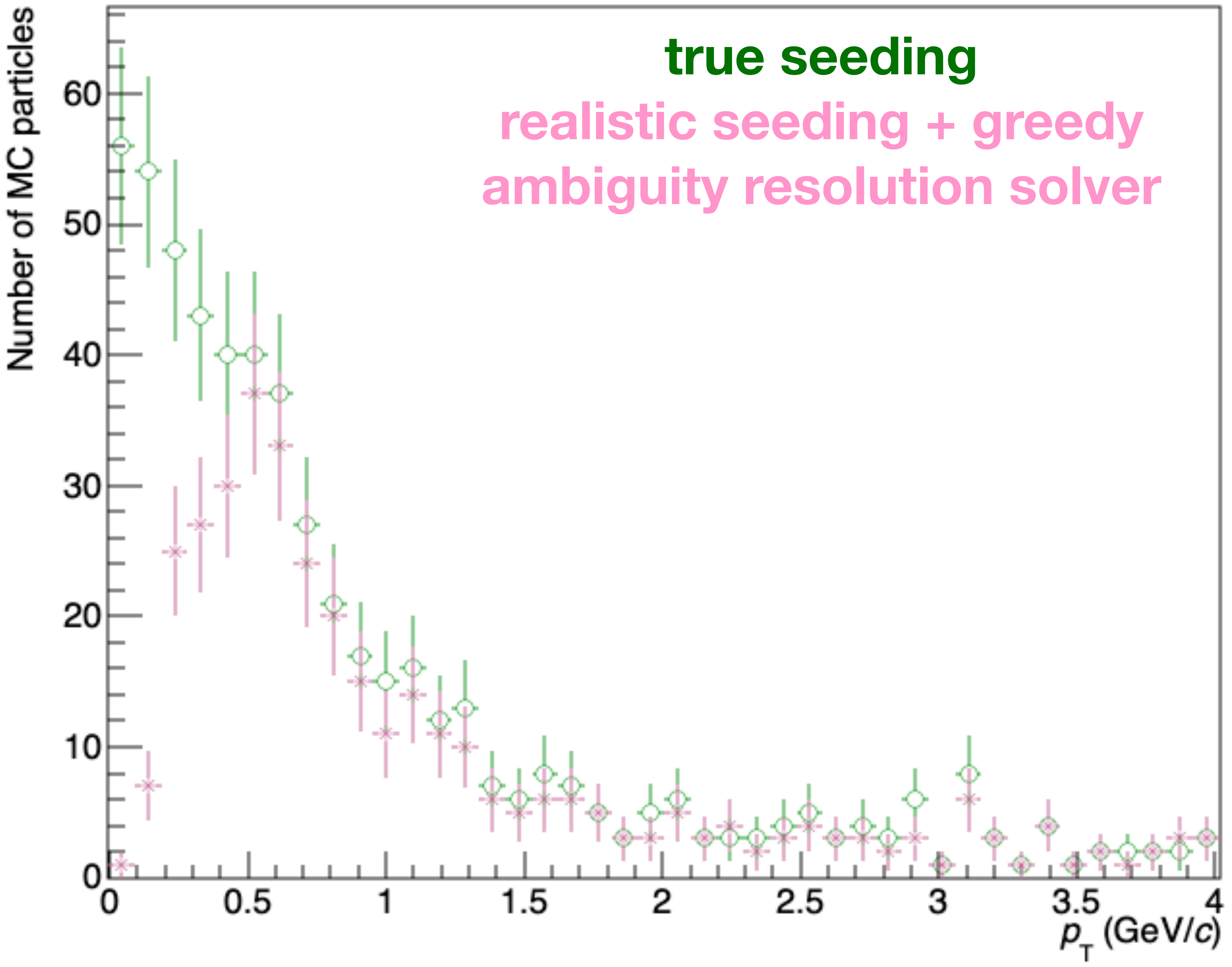




- No qualitative change in purity of reconstructed tracks
- No efficiency loss in reconstruction

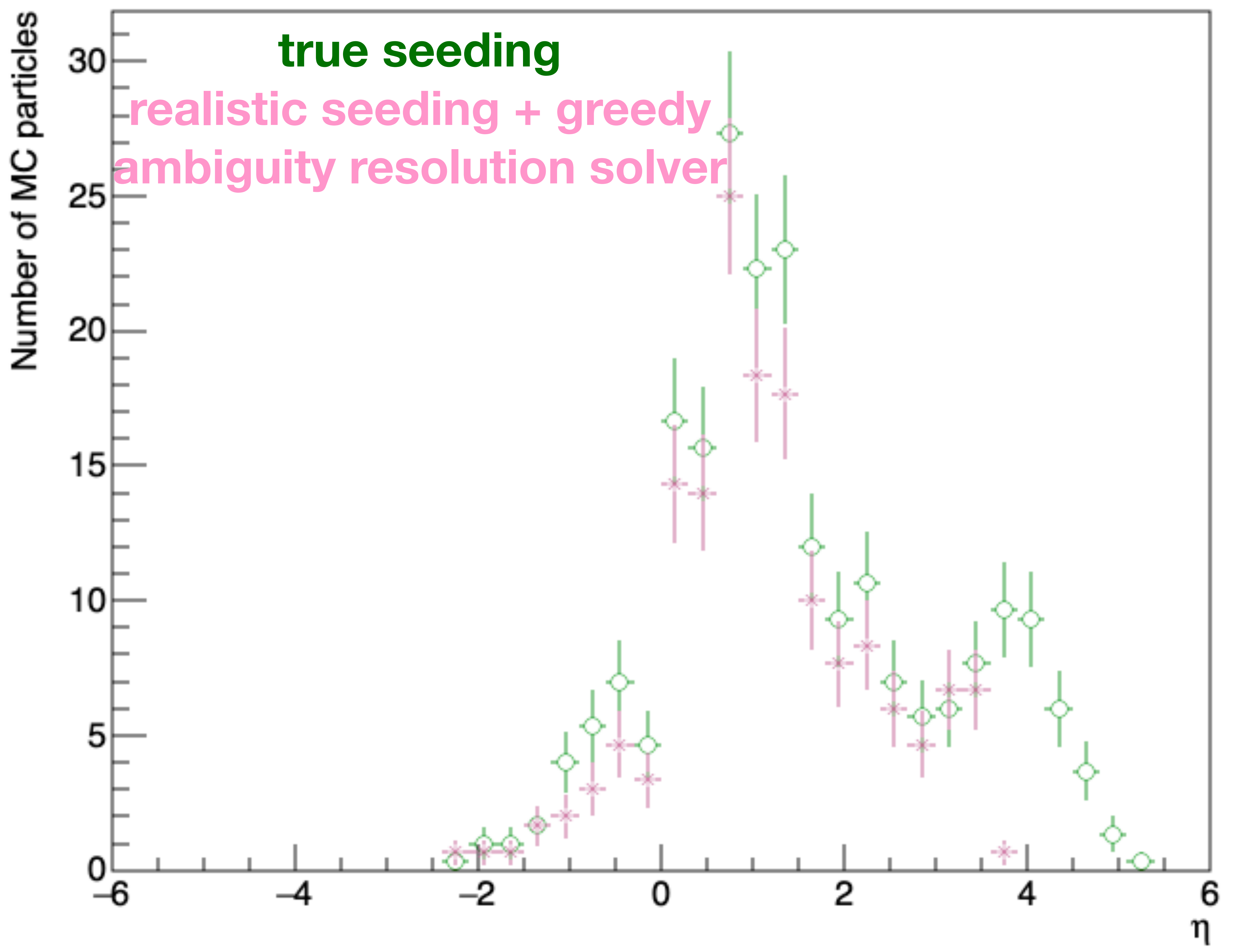
Realistic seeding vs. true seeding: pT

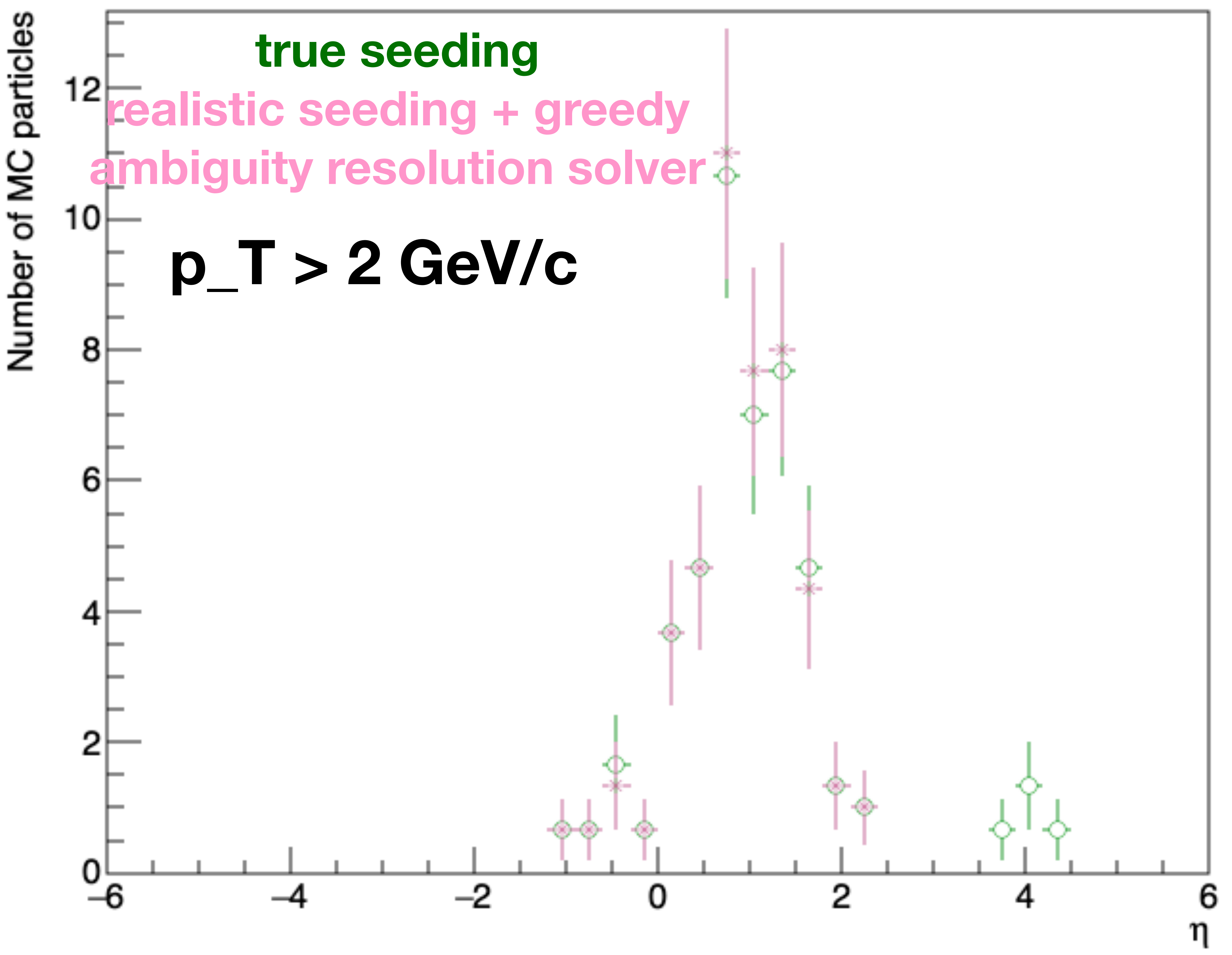




- Large discrepancy at low pt below 0.5 GeV/c
- Compatible in higher pT

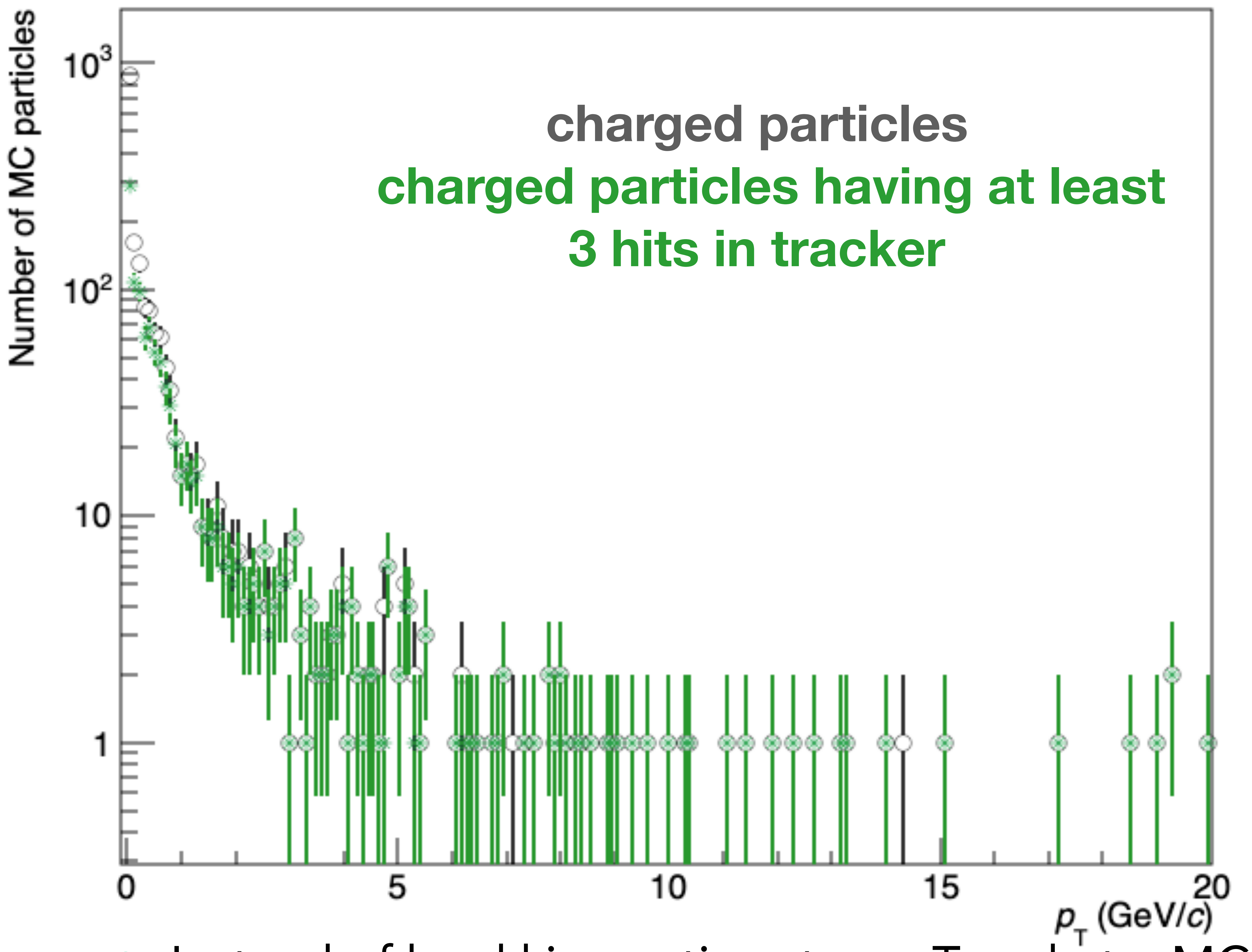
Realistic seeding vs. true seeding: eta

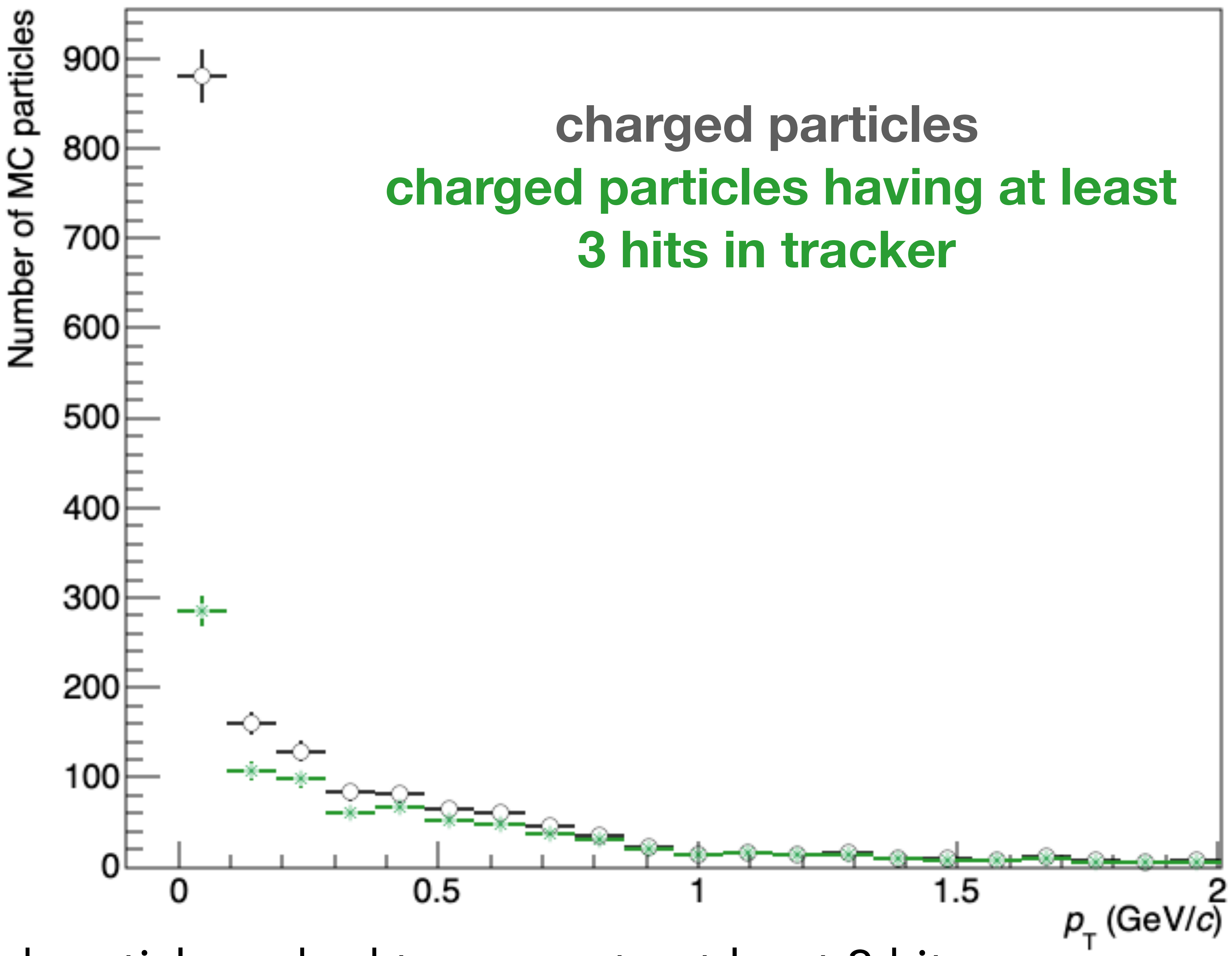




- Eta dependent discrepancy mainly comes from the correlation of pT
- Reco. performance compatible with each other in higher pT

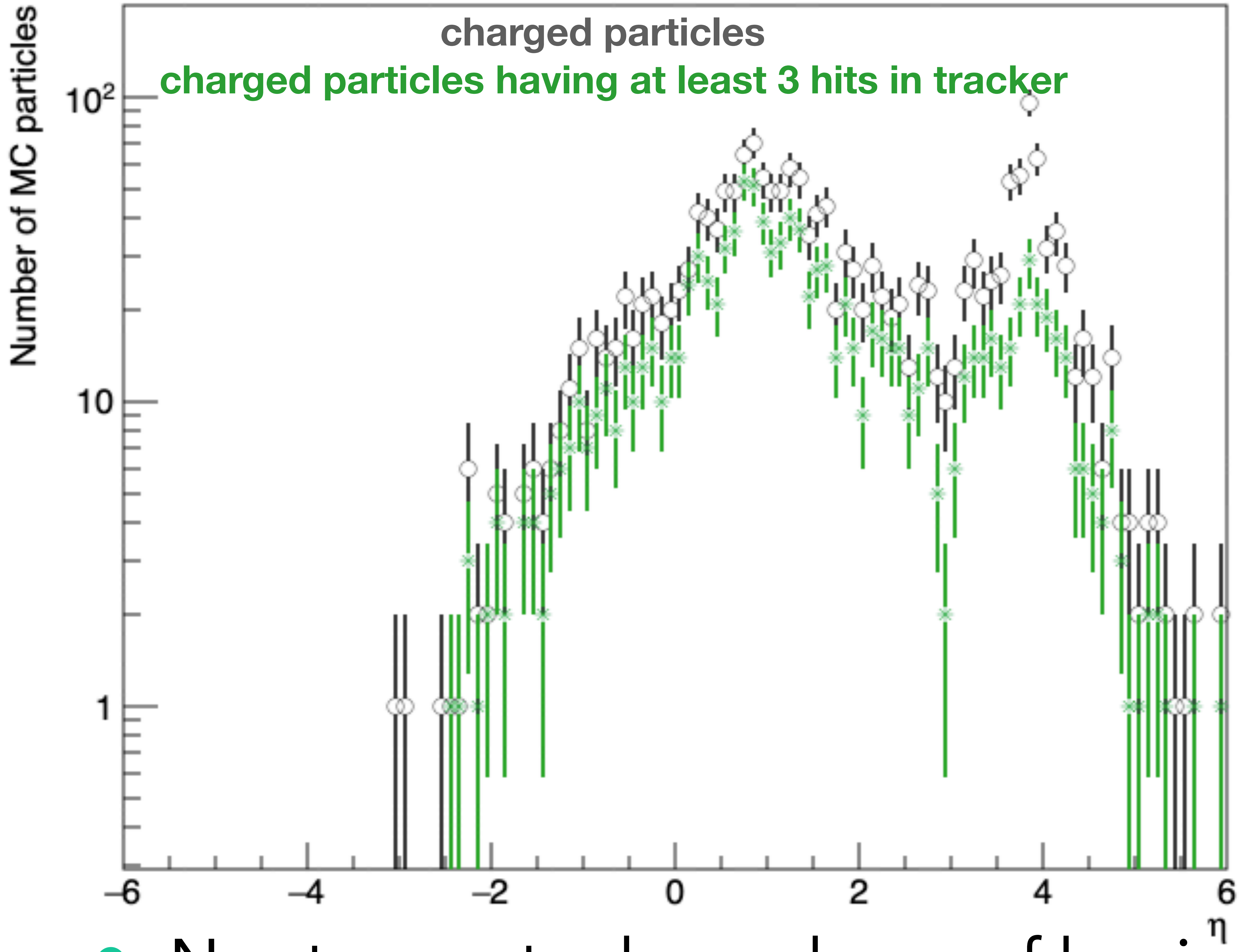
Tracking efficiency: denominator from MC particles

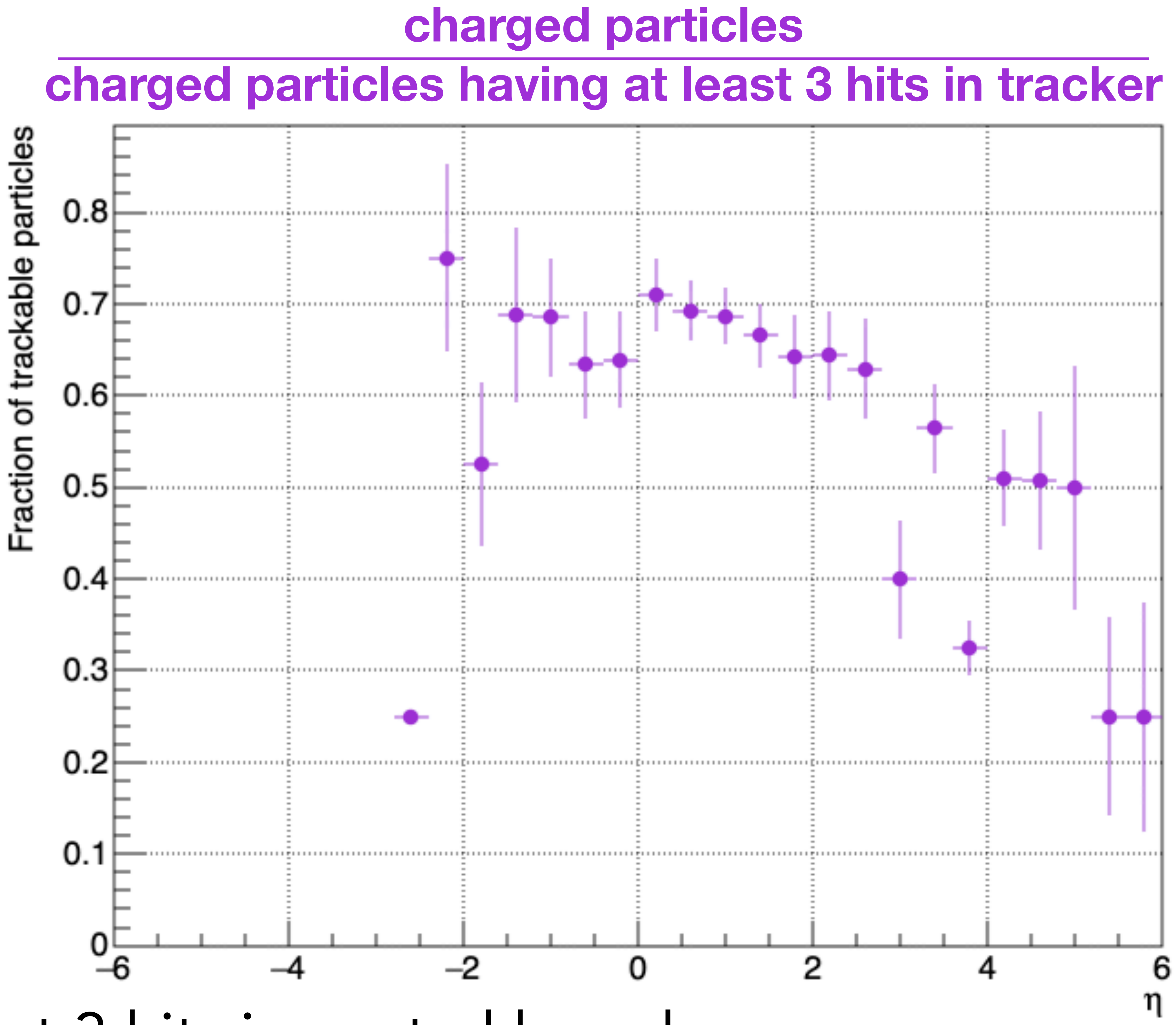




- Instead of hard kinematic cut on pT and eta, MC charged particles asked to generate at least 3 hits on any layer of trackers (including B0)
- Low pT drops naturally

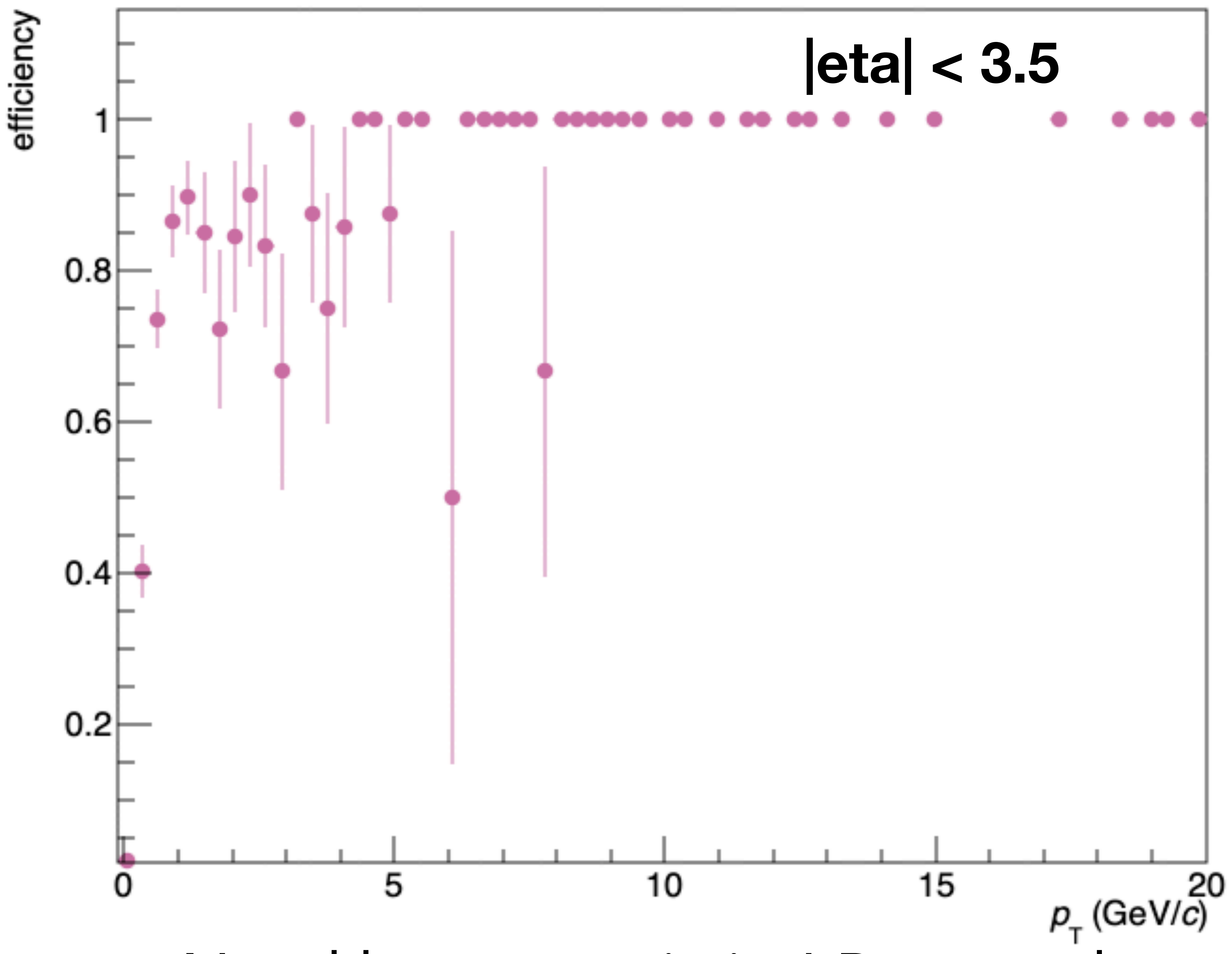
Tracking efficiency: denominator from MC particles

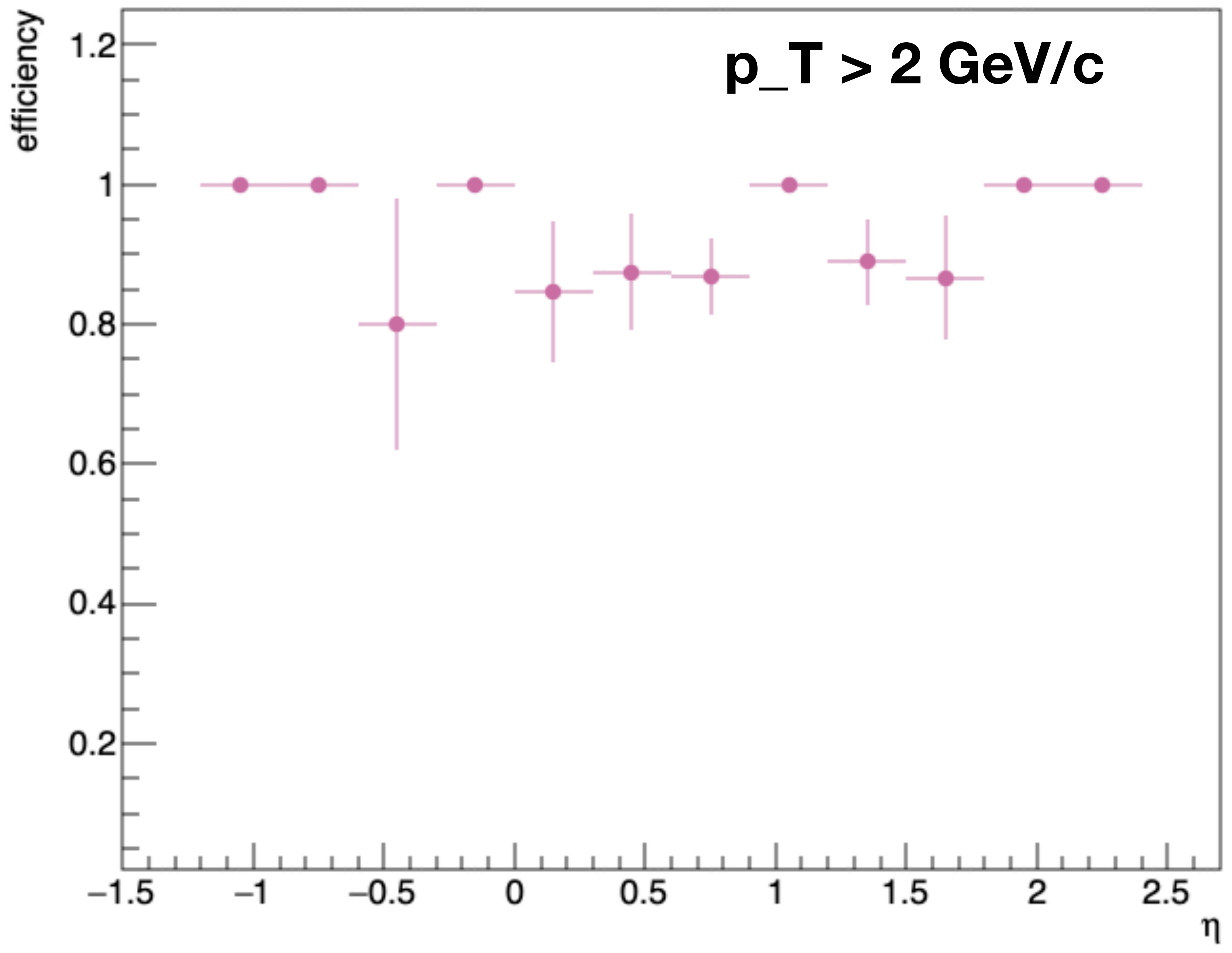




- No strong eta dependence of leaving at least 3 hits in central barrel

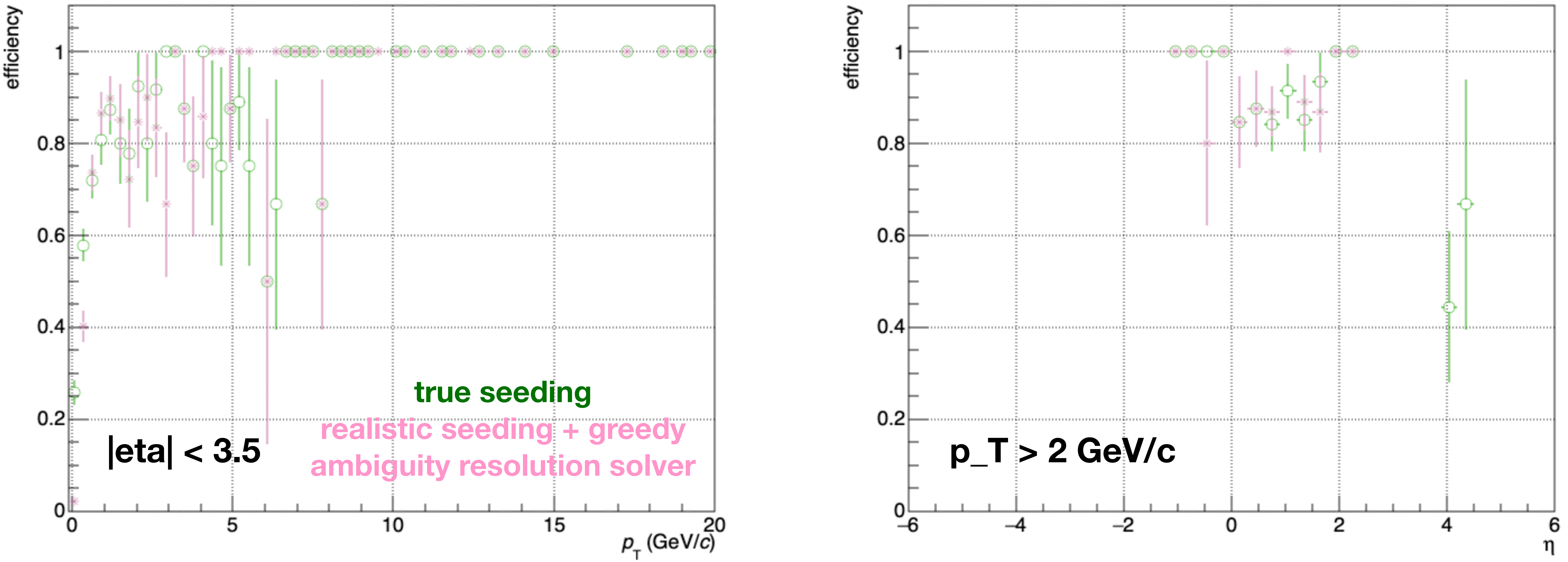
Tracking efficiency from realistic seeding





- Need better statistics! But trend seems reasonable

Tracking efficiency from true seeding

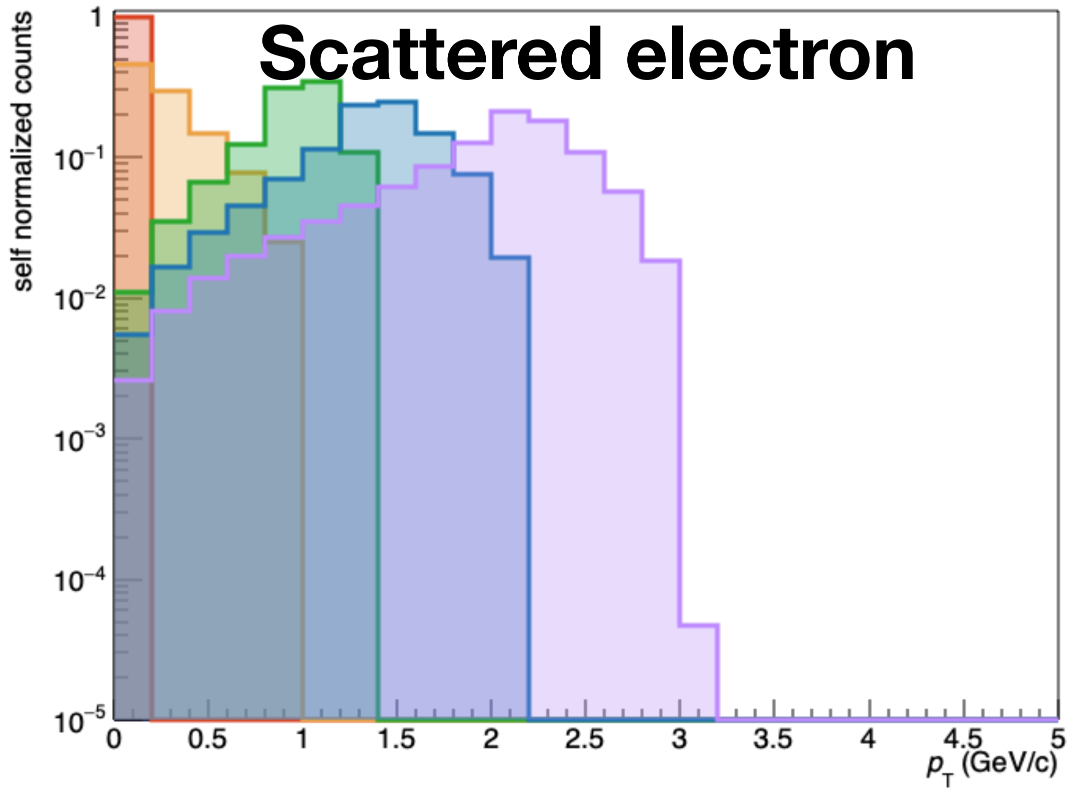


- Need better statistics! But different seeding methods give similar trend except for very low pT

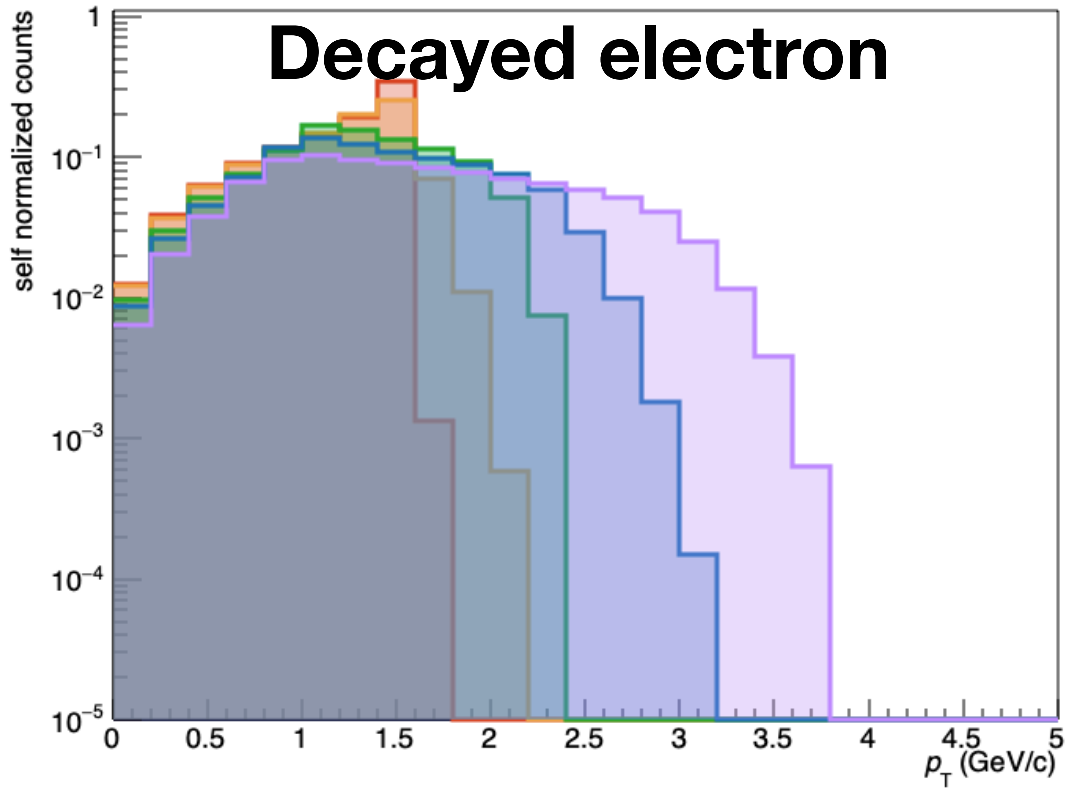
Backup

Reminder: J/ψ photoproduction (5 < Q^2 < 10) in EIC

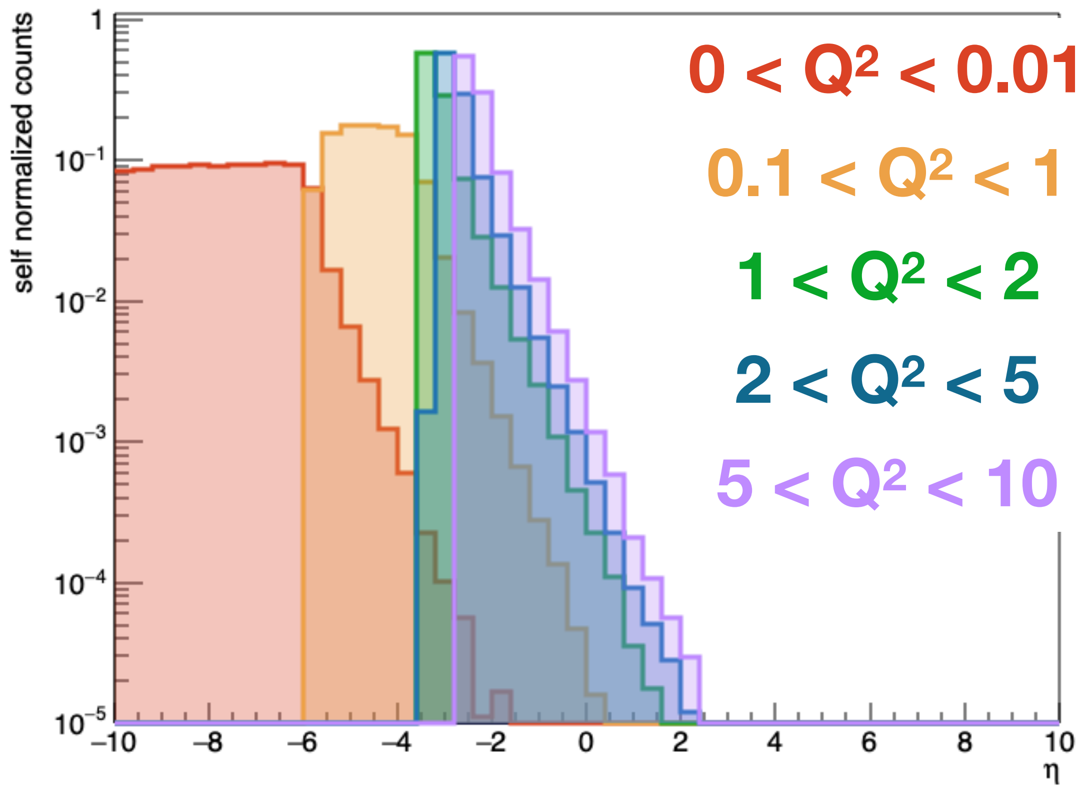
- Coherent production of eA→eA′J/ψ→e(e+e-)A′ with eSTARLight
- Final state particle kinematics are well constrained; most of cases 3 electrons



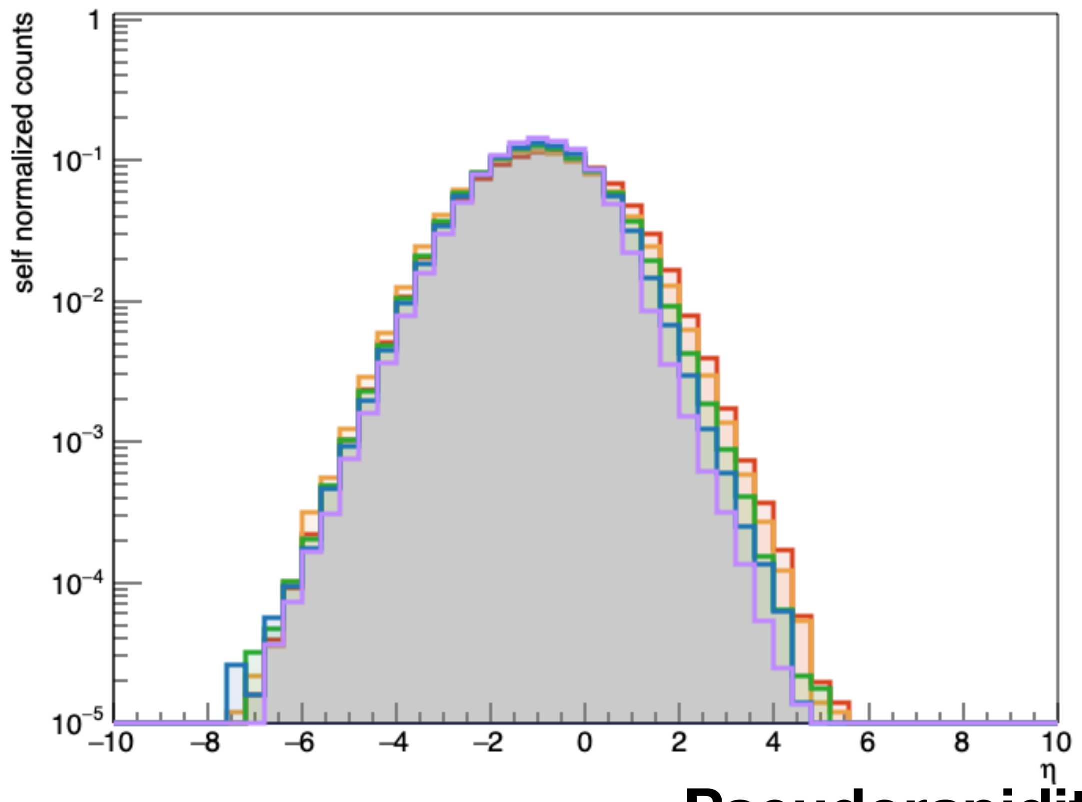
Transverse momentum (GeV/c)



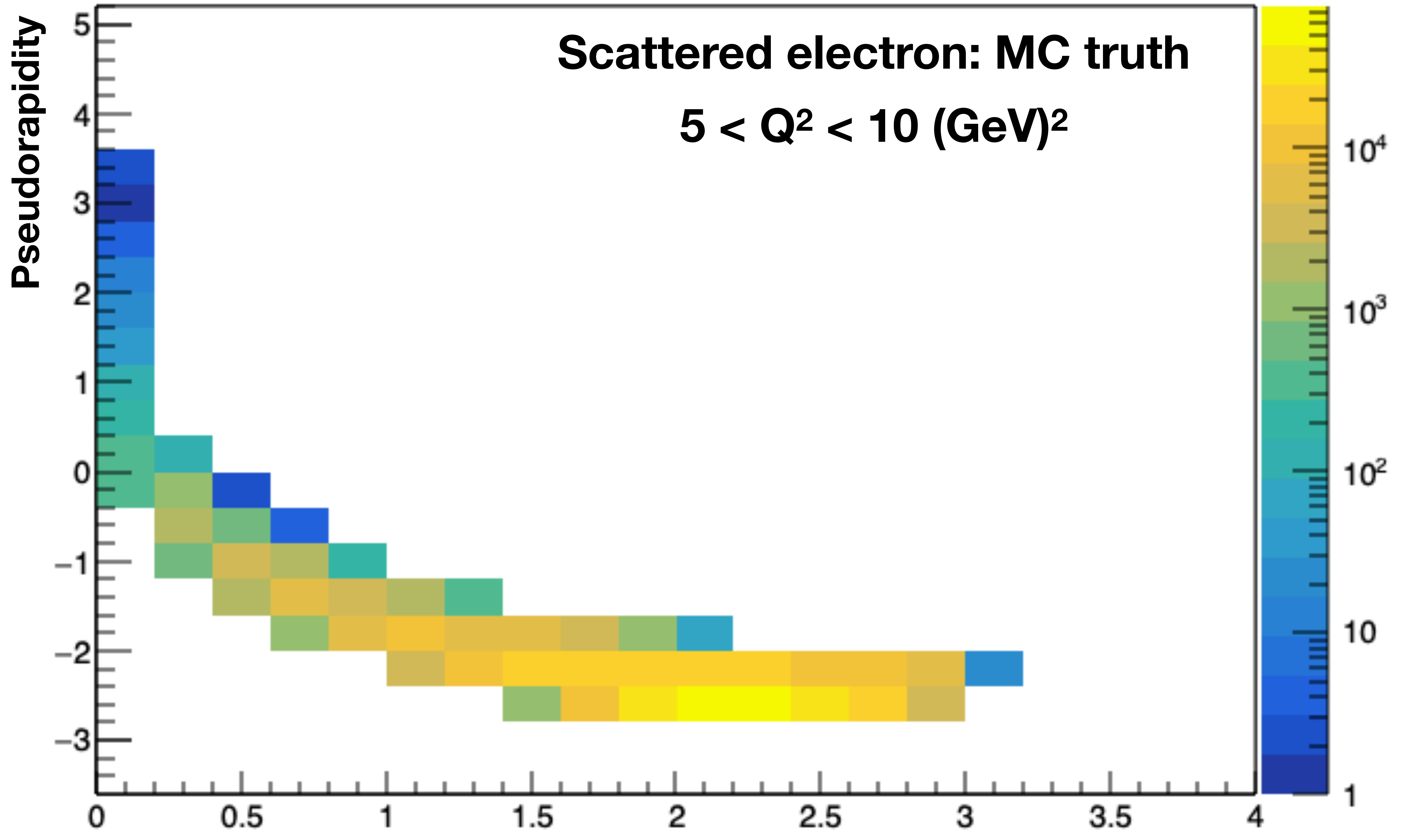
Transverse momentum (GeV/c)



Pseudorapidity



Pseudorapidity

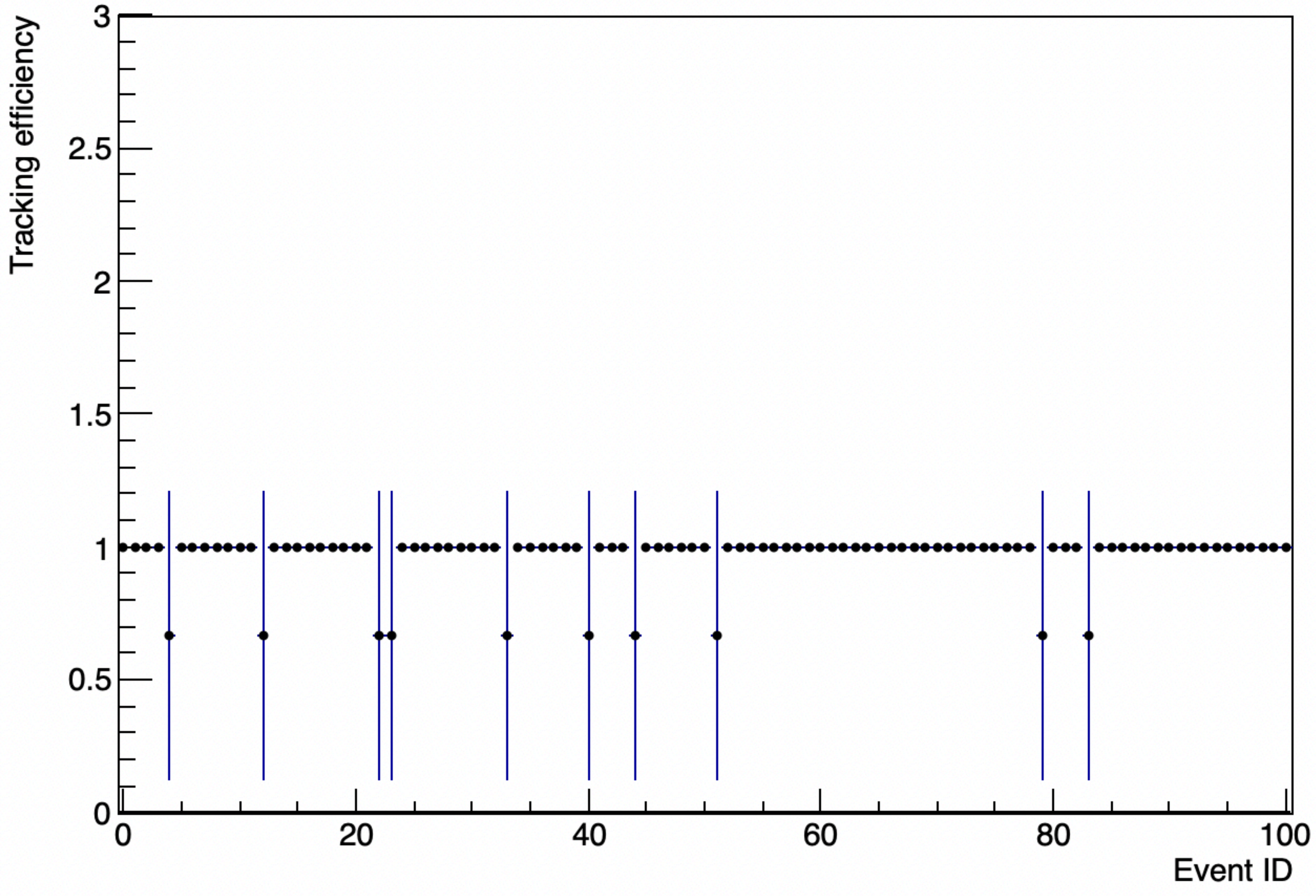


Transverse momentum (GeV/c)

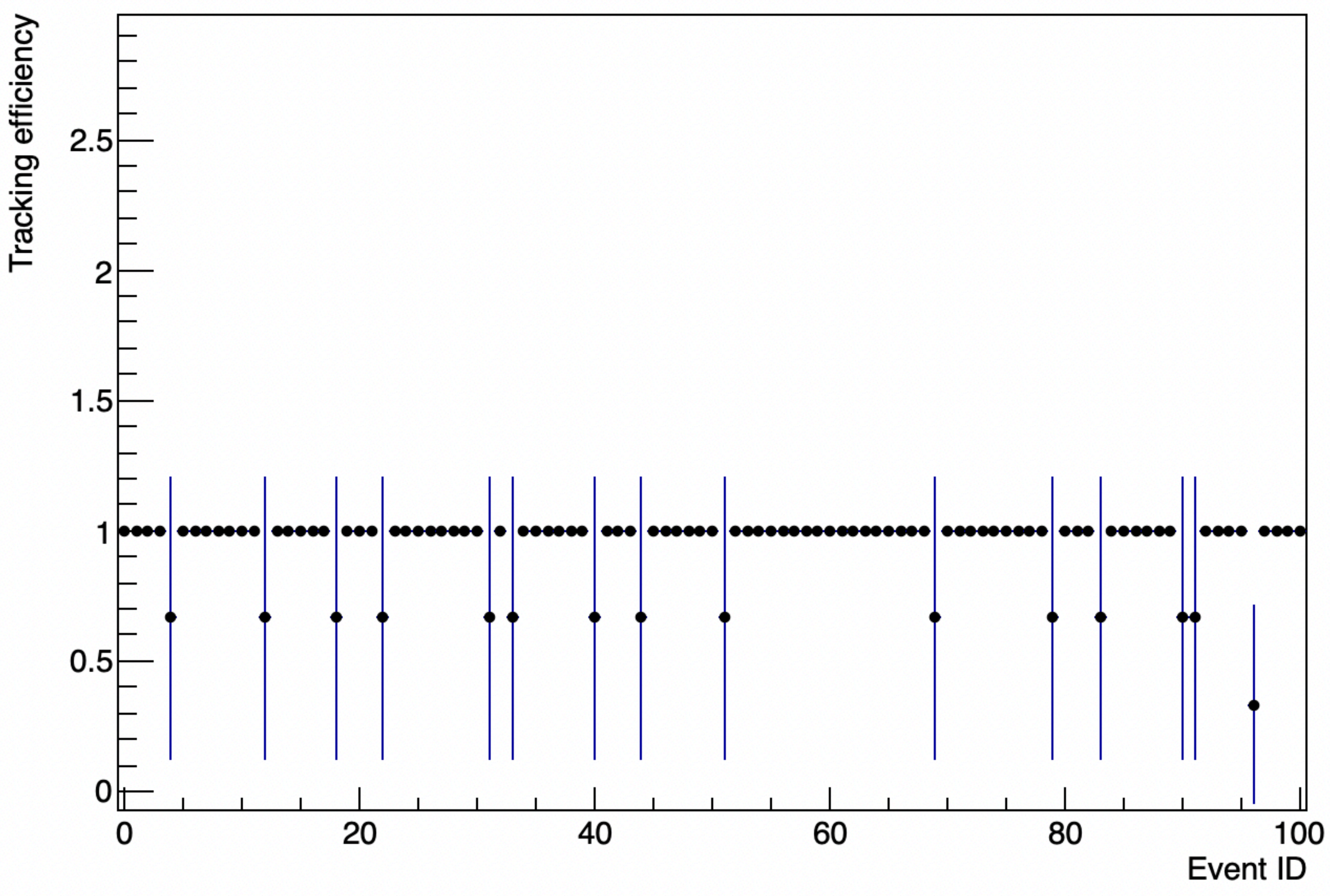
Duplicate track rejection in photoproduction events

- Event-by-event tracking efficiency estimated: $\frac{\text{number of reconstructed tracks}}{\text{number of generated final state particles}}$
- Greedy ambiguity resolution solver works well!! (No events with duplicate tracks!)

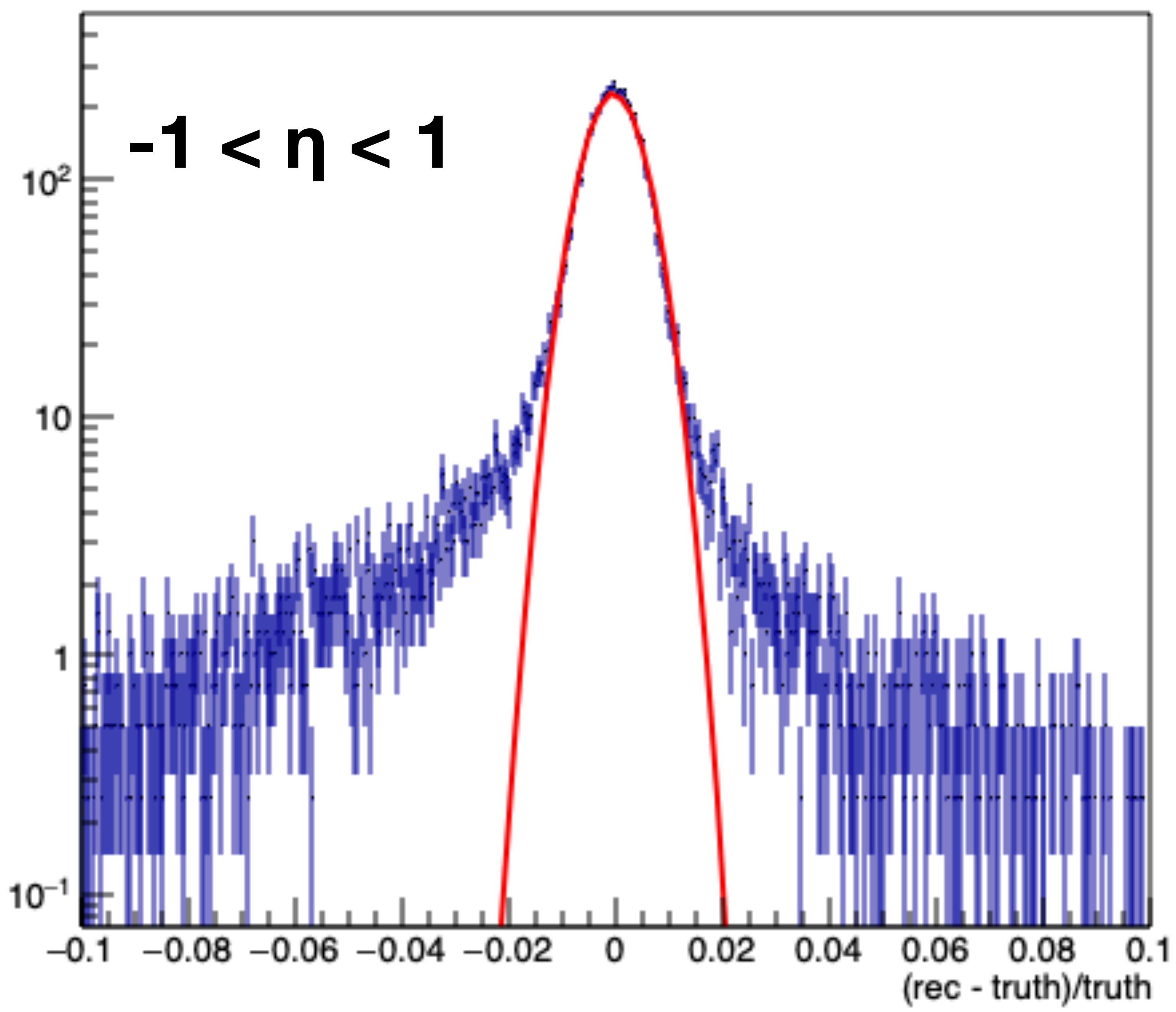
True seeding

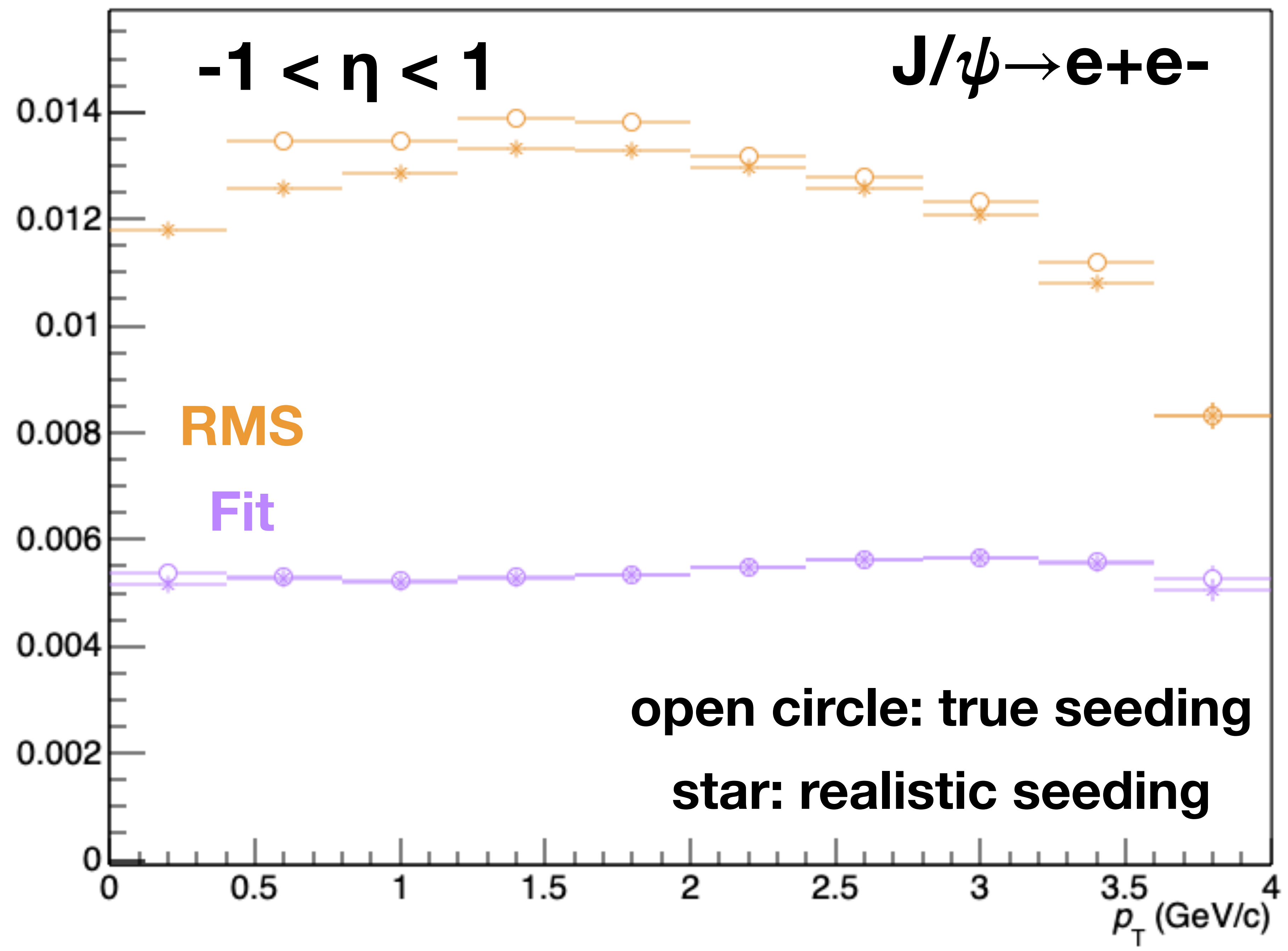


Realistic seeding



Transverse momentum resolution



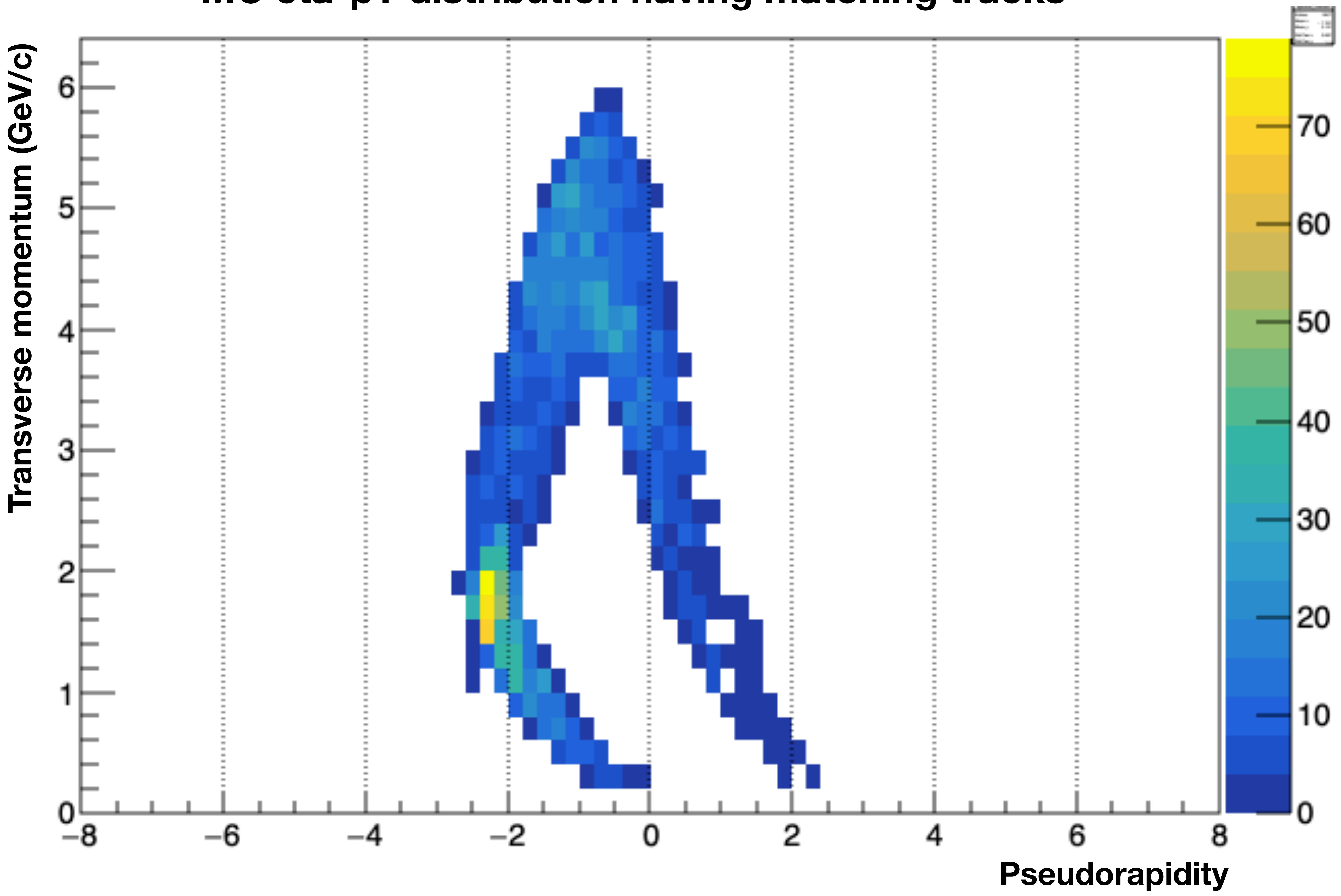


- For MC particles in small transverse momentum (p_T) range:
  - Find the matching tracks
  - Calculate the deviation of reconstructed p_T w.r.t. truth p_T
  - Take the RMS or the width from gaussian fit as the resolution
- Larger negative tail due to the multiple scattering/Bremsstrahlung

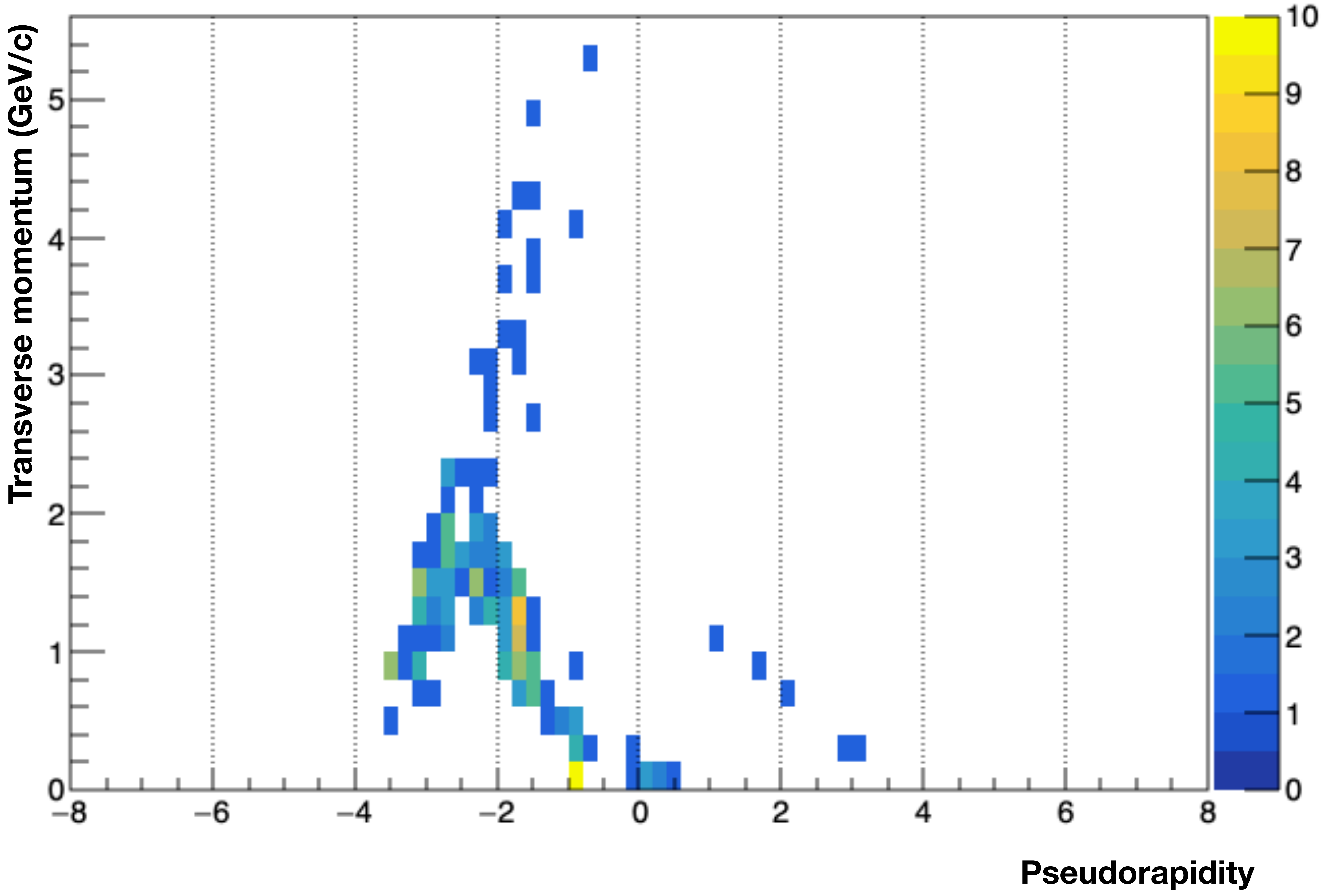
First look of tracking efficiency of the ePIC tracker

- Most of efficiency lost in low p_T below 200 MeV/c
- Limited kinematic coverage of photoproduction events

MC eta-pT distribution having matching tracks

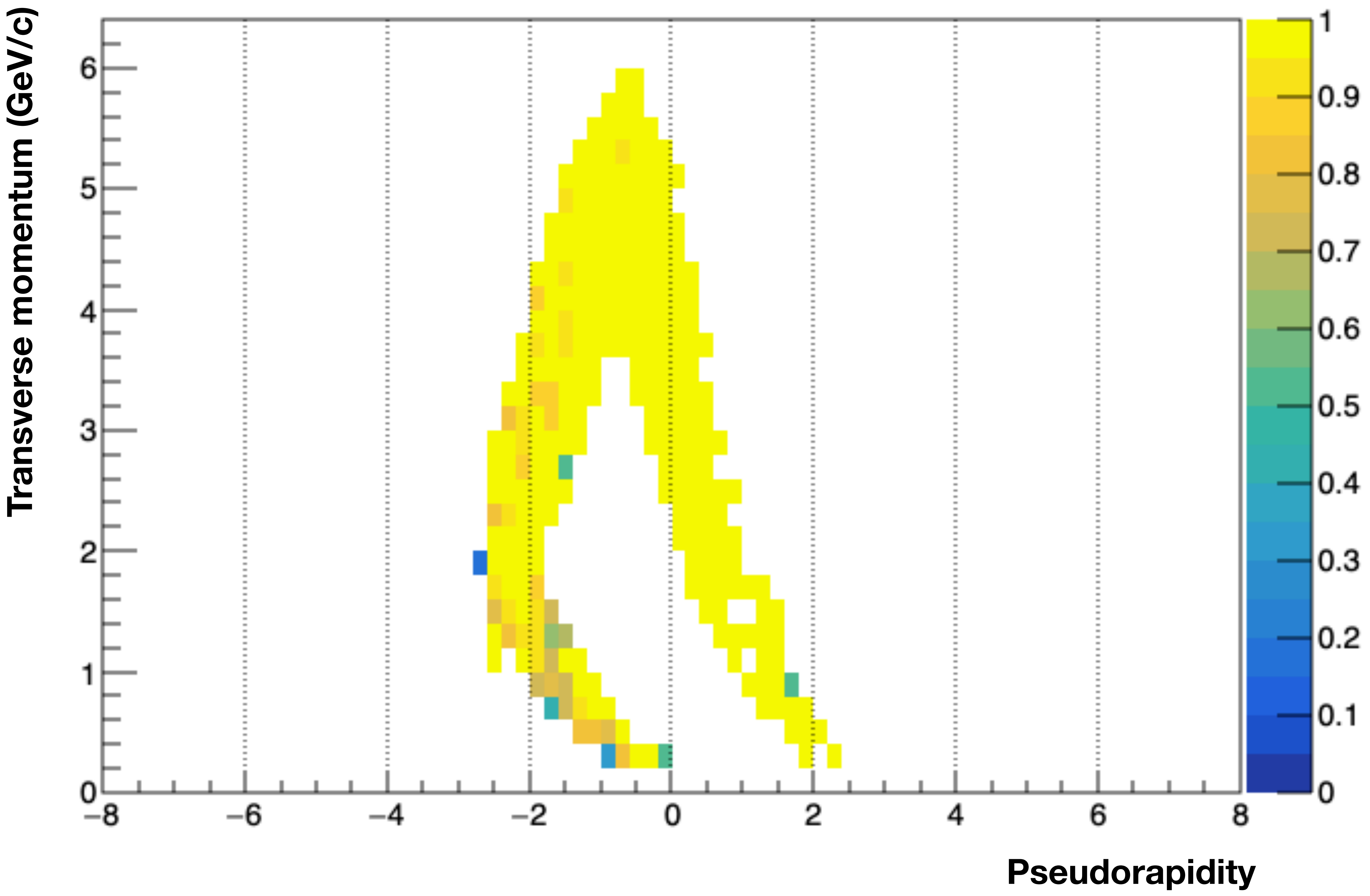


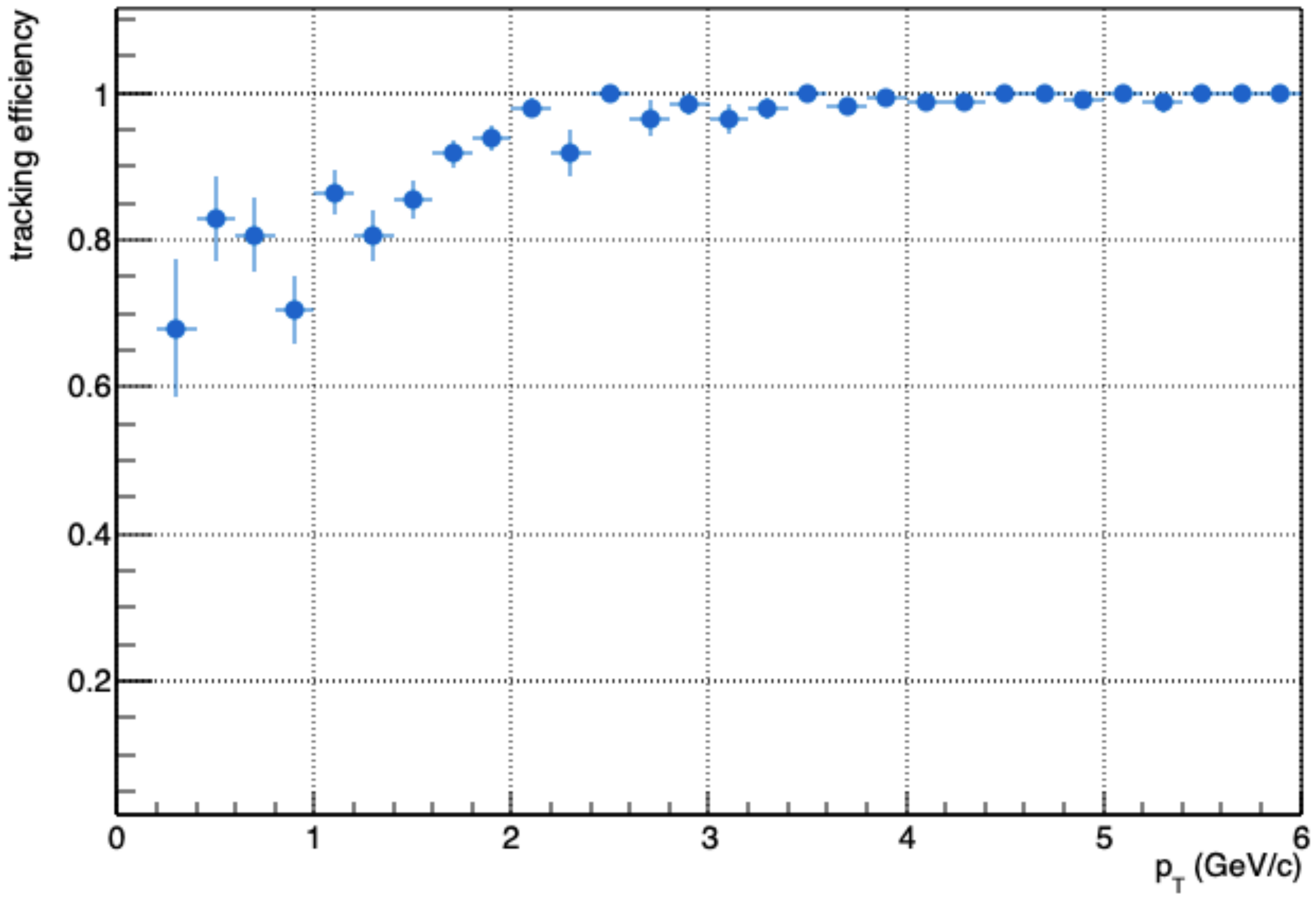
MC eta-pT distribution not having matching tracks



First look of tracking efficiency of the ePIC tracker

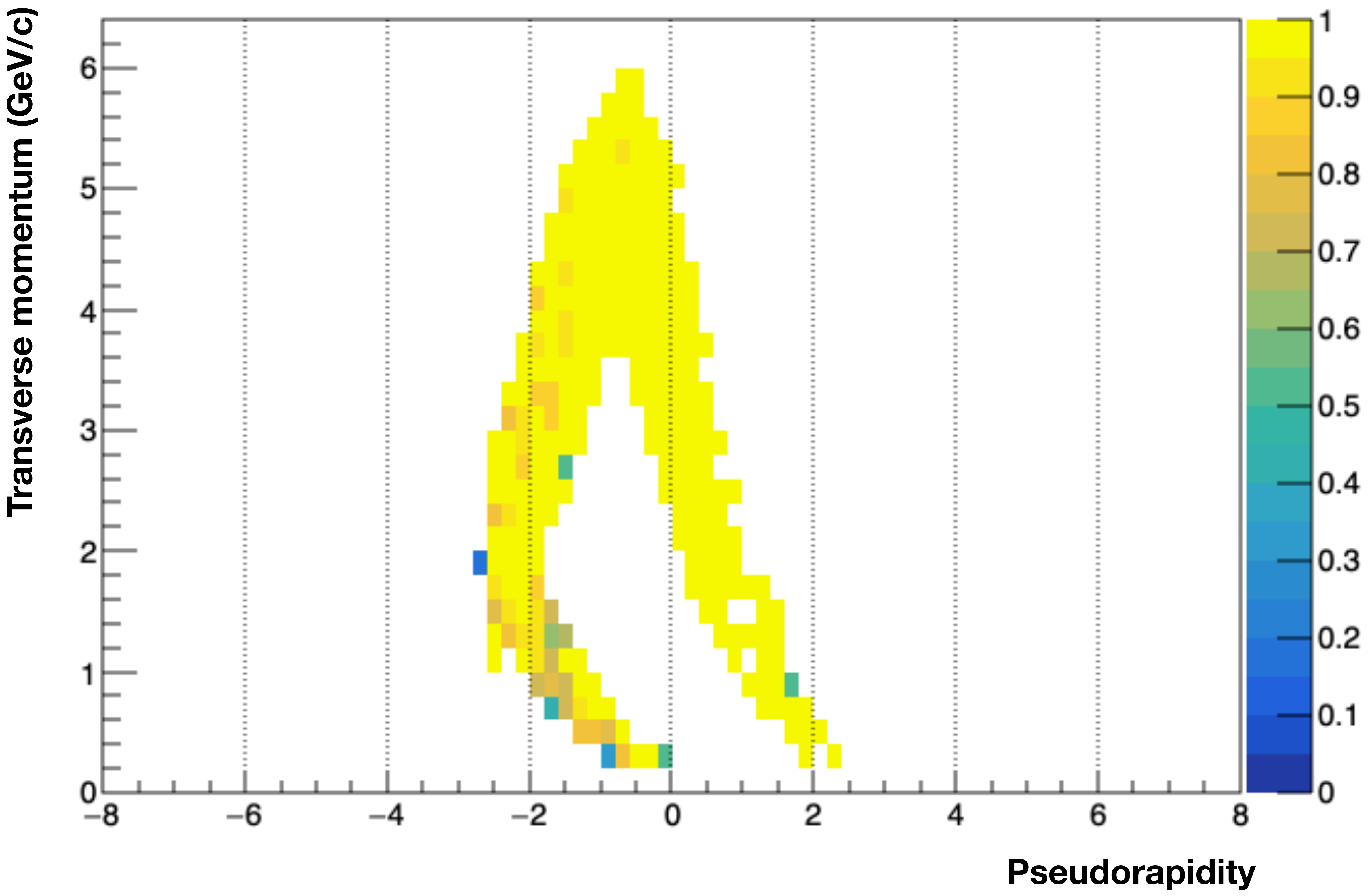
- Fully efficiency above 2 GeV/c
- Tracking efficiency: $\frac{\textbf{number of reconstructed, matched tracks}}{\textbf{number of generated final state particles}}$

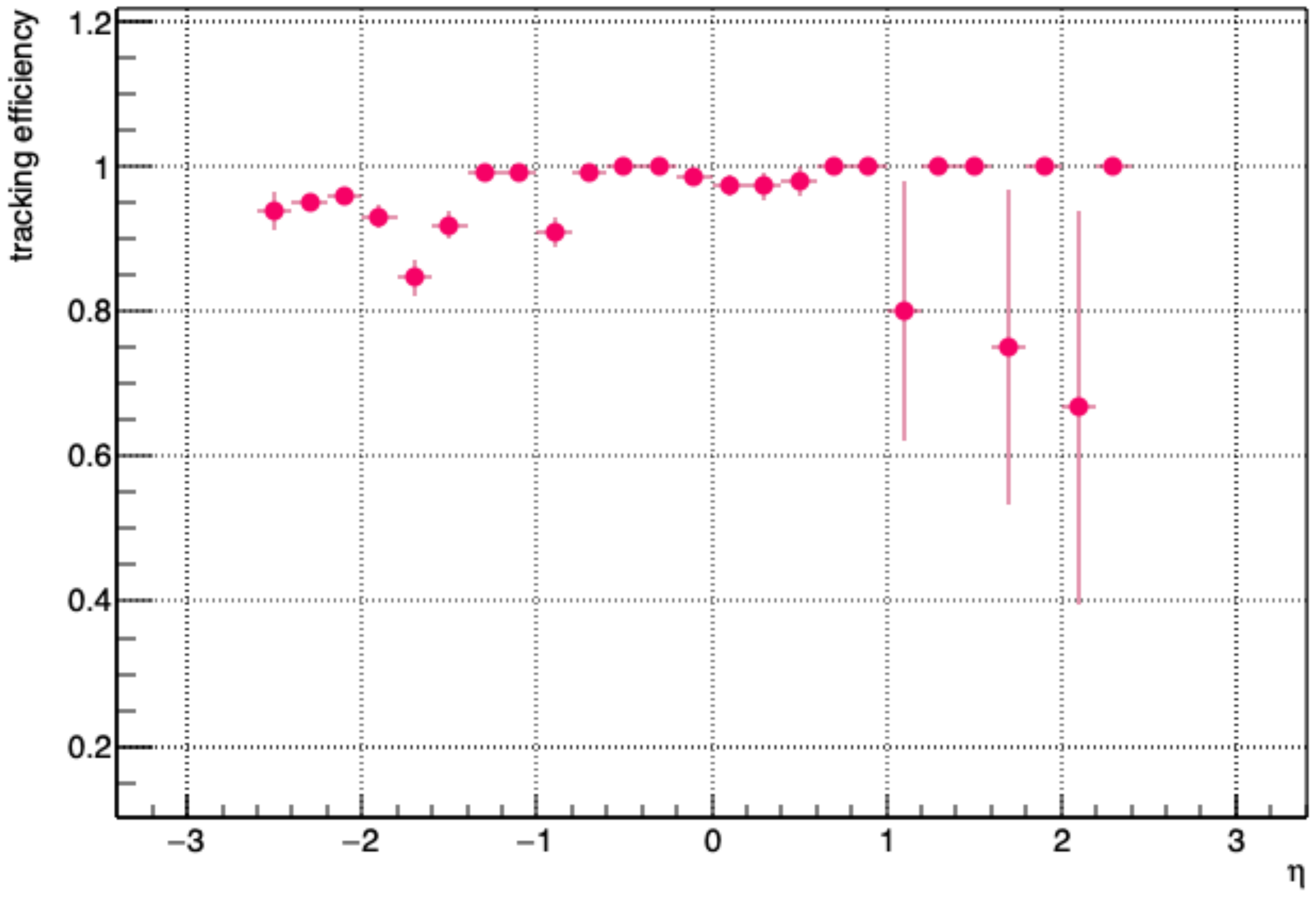




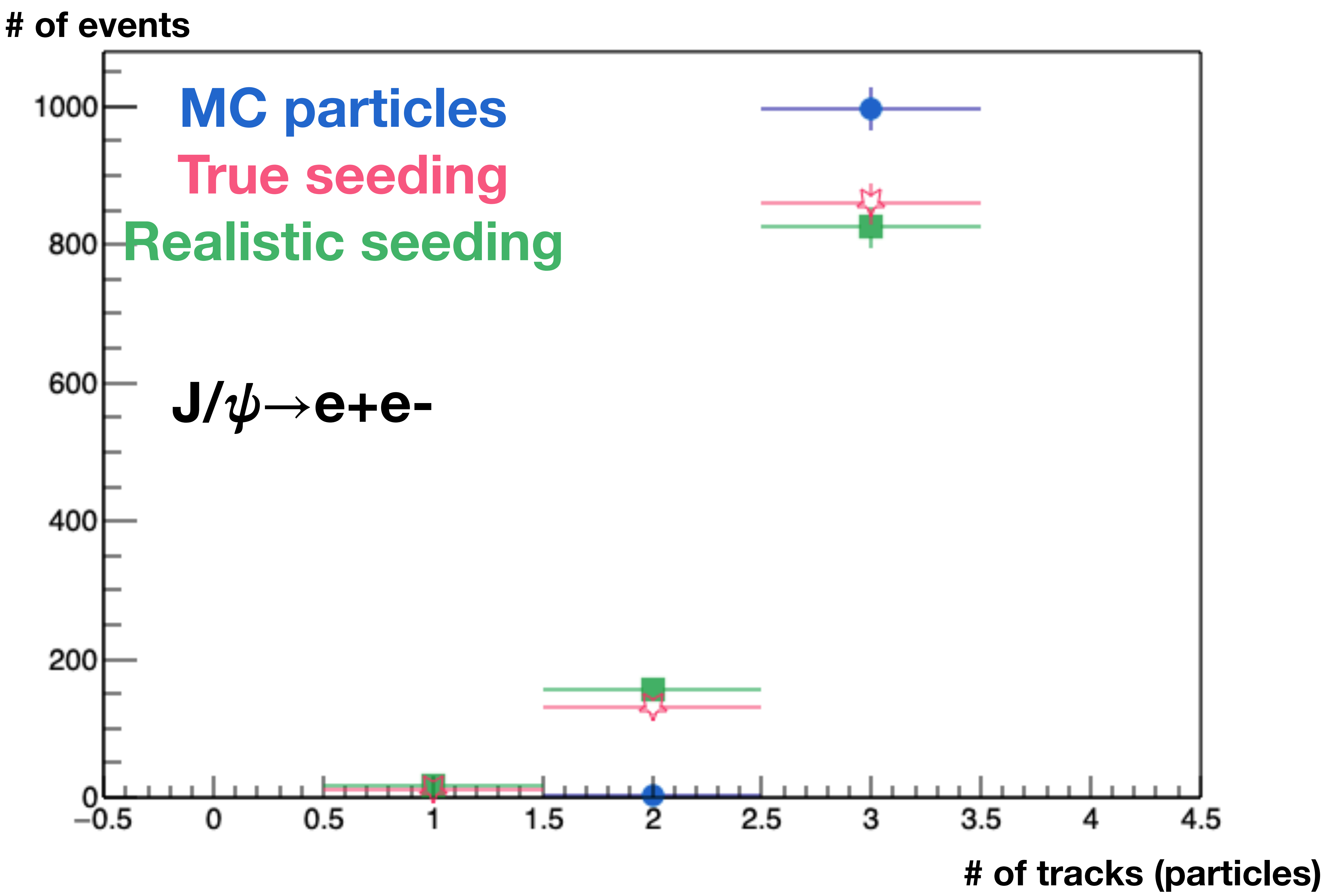
First look of tracking efficiency of the ePIC tracker

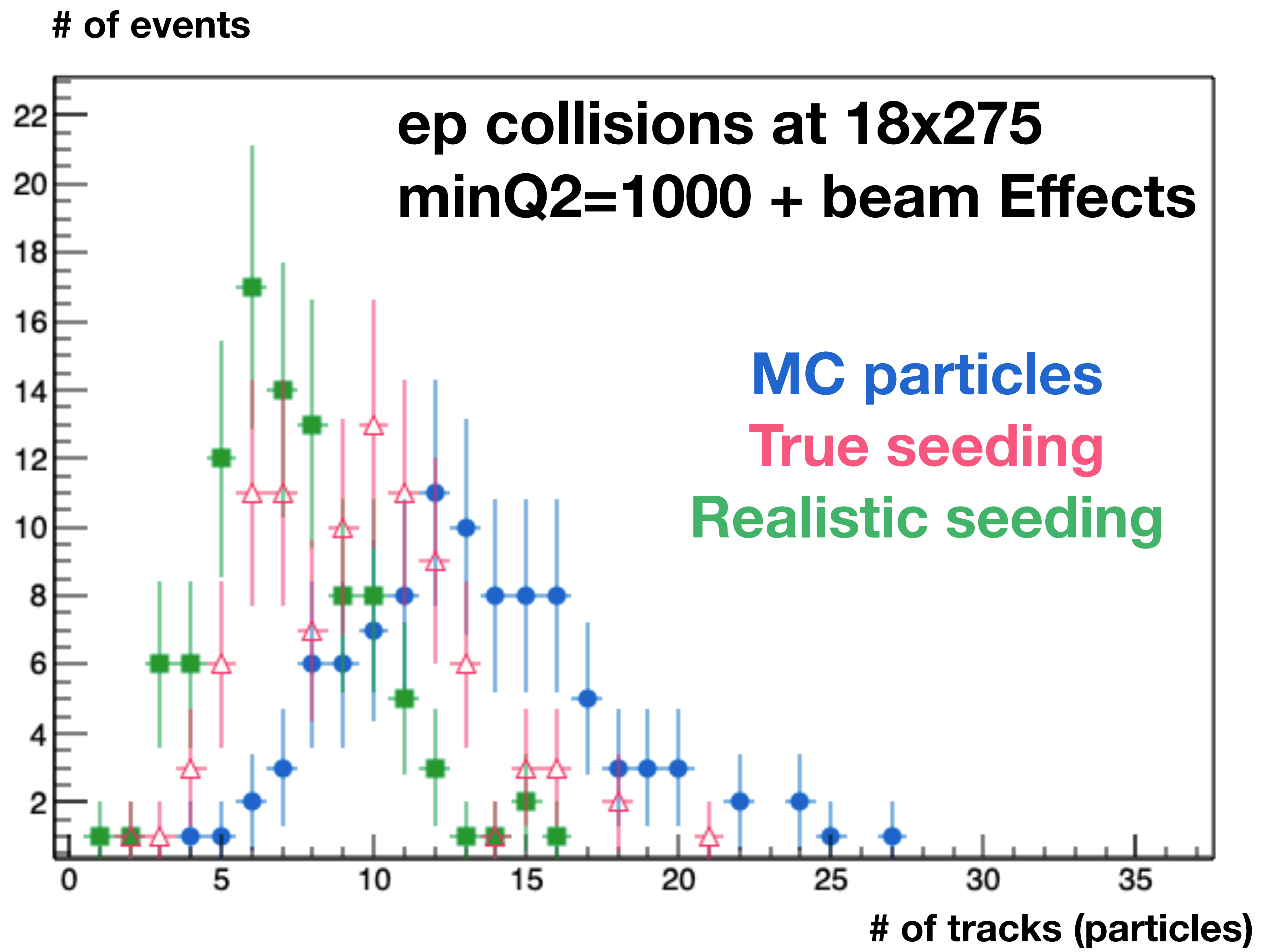
- No eta dependence found within -2.5 < eta < 2.5
- Tracking efficiency: $\frac{\textbf{number of reconstructed, matched tracks}}{\textbf{number of generated final state particles}}$





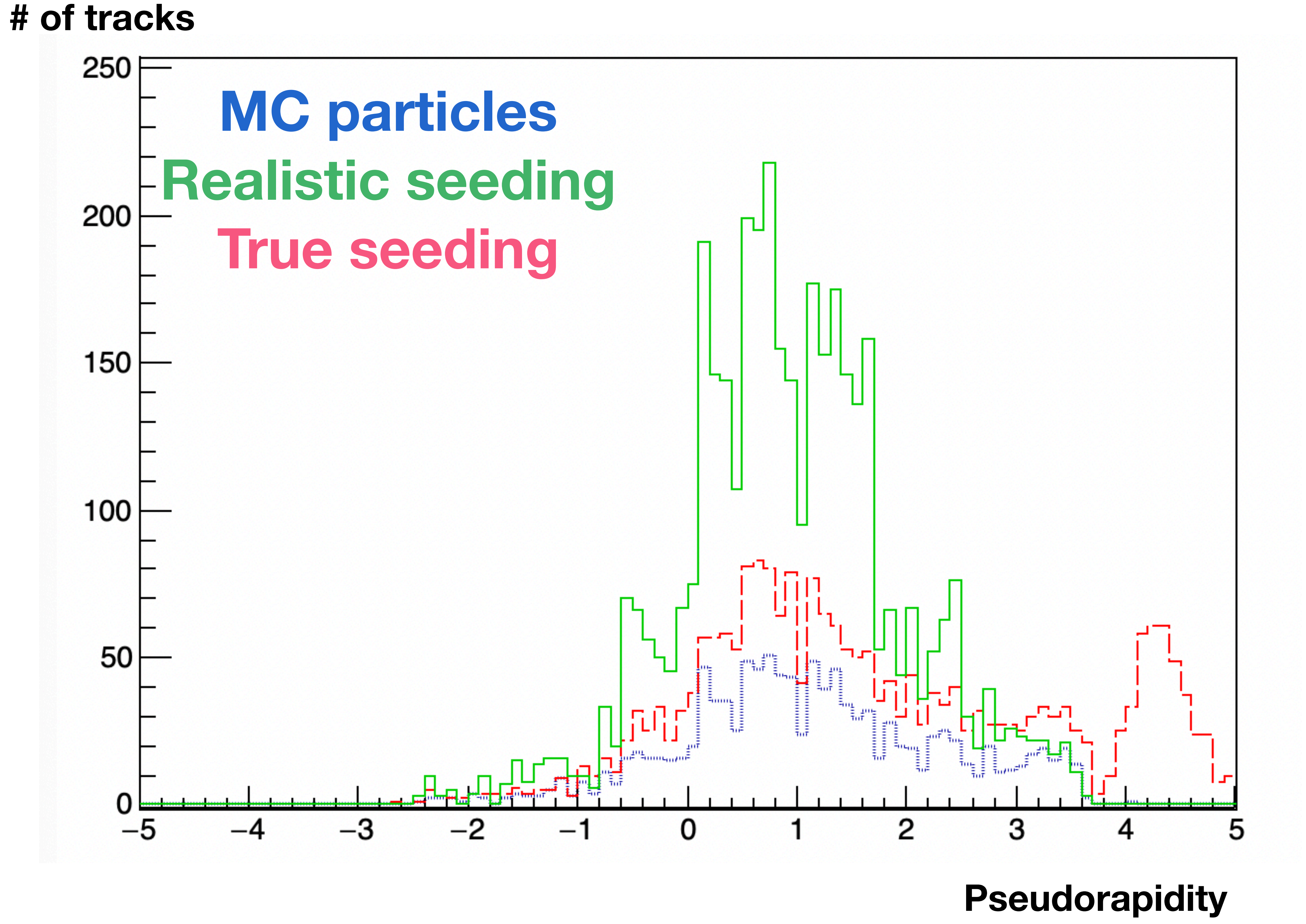
Moving towards complicated environment

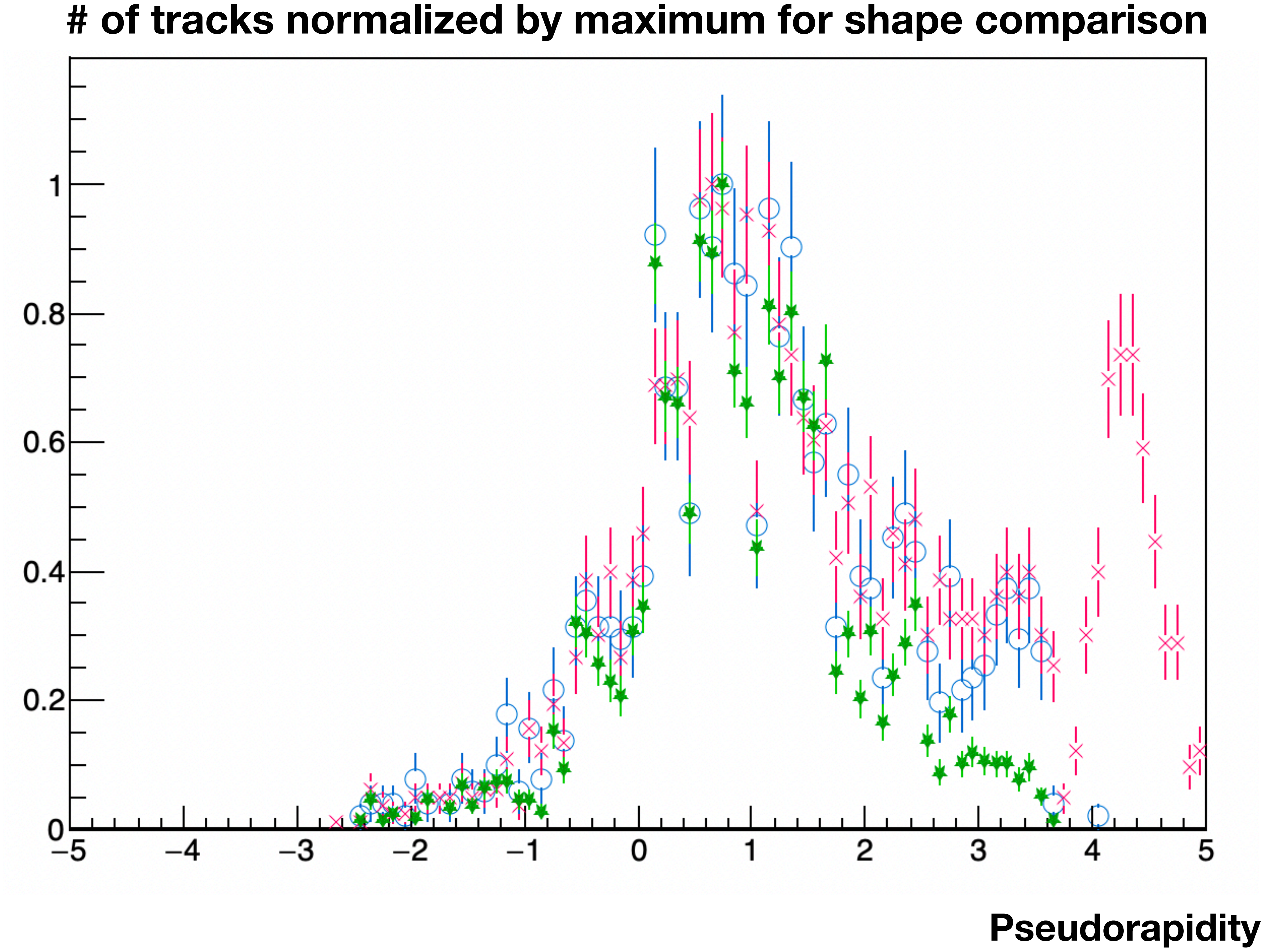




- Efficiency loss in true seeding/realistic seeding: No kinematic cuts on MC particles yet
- Slightly worse performance in realistic seeding: over-suppressed by duplicates rejection?
- More complicated in DIS

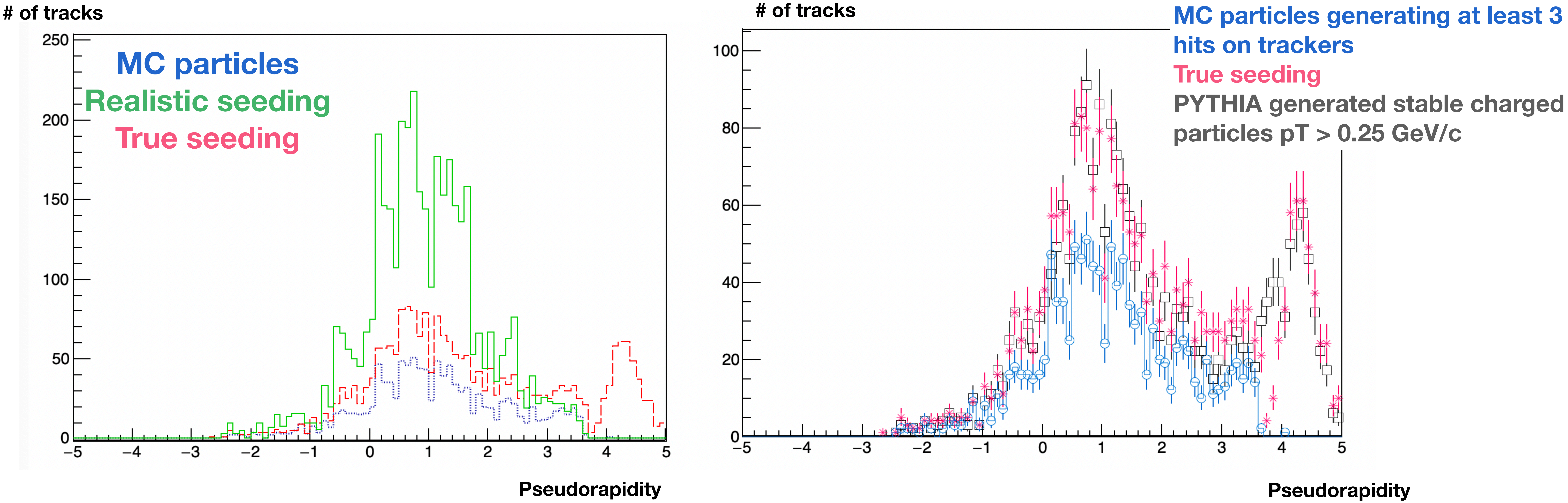
High Q^2 DIS events: seed-level QA





- MC particles: stable particles generating at 3 hits on silicon trackers
- Much larger entries w.r.t. MC particles in realistic seeding: duplicate seeds visible as expected
- Larger entries in true seeding w.r.t. MC particles in true seeding, with seeds in B0 acceptance

Caution for true seeding usage



- Larger entries in true seeding w.r.t. MC particles in true seeding, with seeds in B0 acceptance
- True seeding takes PYTHIA generated charged particle with min. pT: doesn't guarantee whether they are really trackable in the detector! **min. # of generated hits for input MC particles?**

Ambiguity Resolution in EICrecon

- algorithm/tracking/AmbiguitySolver.cc(h):

  std::vector<const ActsExamples::ConstTrackContainer*> input_container

  - Taking CKFTracking output(s)
  - Call algorithm and process
  - Convert output(s) from the algorithm to standard EICrecon output (compatible with tracking output(s) for seamless transition for the rest of the reco. chain)

```
std::tuple<
   std::unique_ptr<edm4eic::TrajectoryCollection>,
   std::unique_ptr<edm4eic::TrackParametersCollection>,
   std::unique_ptr<edm4eic::TrackCollection>,
   std::vector<ActsExamples::Trajectories*>,
   std::vector<ActsExamples::ConstTrackContainer*>
>
```

- algorithms/tracking/AmbiguitySolverConfig.h
  - Configuration helper for the algorithm:

    max. shared hits, **max. iteration** and **min. # of measurements per track**
- global/tracking/AmbiguitySolver_factory.h
  - Factory for the algorithm; to be called after “CKFtracking” inside full tracking chain