Acts Greedy Ambiguity Resolution Solver in EICRecon

Barak Schmookler, Minjung Kim

Thanks to Dmitry, Shujie, and Wouter for reviewing the PR!

Implementation in EICRecon

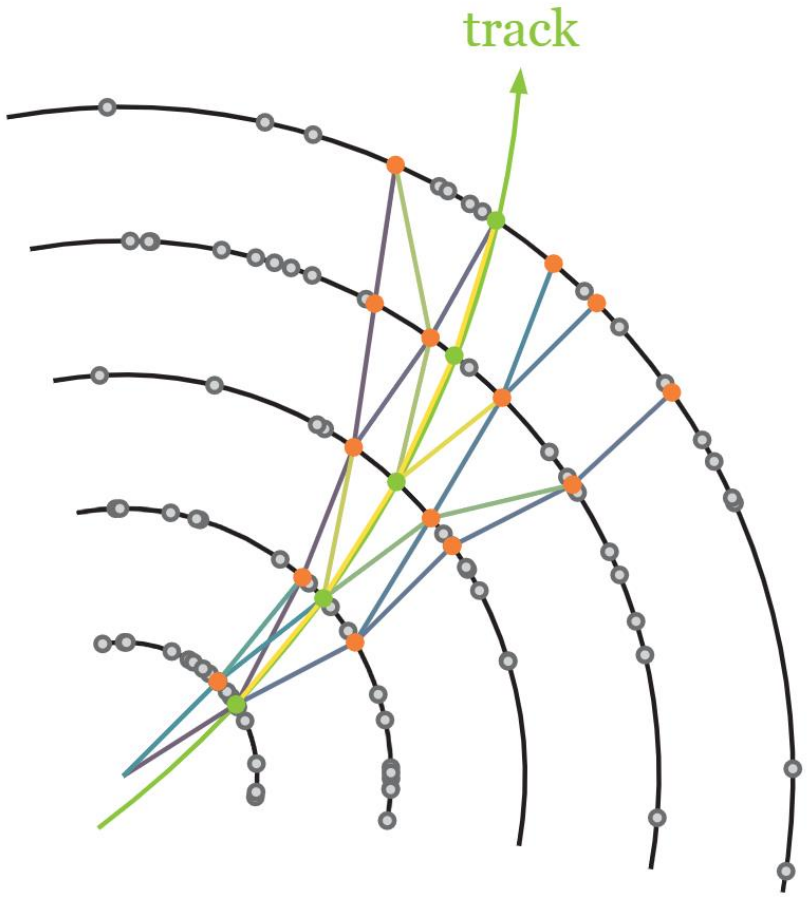
- The Acts Greedy Ambiguity Resolution Solver is now included in EICRecon. (PR was merged into main branch last night.)
- The solved tracks that are the output from the ambiguity resolution solver are now the default tracks that will be used for analysis and in downstream algorithms.
- These solved tracks combine the input tracks which contain a minimum number of shared hits. This is important for removing duplicate seeds. In addition, the input tracks will be required to have a minimum of number of tracker measurement hits.
- The ambiguity solver is used for both truth and real seeded tracking.
- The unsolved tracks are still saved to the EICRecon output as 'unfiltered' track collections.

```
namespace eicrecon {
struct AmbiguitySolverConfig {
  /// Maximum amount of shared hits per track.
  std::uint32_t maximum_shared_hits = 1;
  /// Maximum number of iterations
  std::uint32_t maximum_iterations = 100000;
  /// Minimum number of measurement to form a track.
  std::size_t n_measurements_min = 3;
};
} // namespace eicrecon
```

https://github.com/eic/EICrecon/blob/main/src/algorithms/tracking/AmbiguitySolverConfig.h

Differences between previous and new output

- Previously, all track candidates found by the CKF were grouped into trajectories based on the corresponding seed. Then we saved one of the track candidates for each trajectory/seed. If a trajectory had multiple tracks (trackTips), we saved the first one in the array – which seems to have been set somewhat arbitrarily.
- In the current implementation, all the track candidates (trackTips) are saved to the unfiltered output. The ambiguity filter works on those trackTips and returns a list of those trackTips that survive the filter. We then create a trajectory which corresponds to a given filtered trackTip.



Example: 10 single muon events with truth-seeded tracking

Reconstructed polar angle for unfiltered and final tracks

```
root [4] events->Scan("CentralCKFTrackParametersUnfiltered.theta:CentralCKFTrackParameters.theta","")
************************************************
*    Row   * Instance * CentralCK * CentralCK *
************************************************
*        0 *        0 * 3.0530140 * 3.0530140 *
*        1 *        0 * 2.8717191 * 2.8717191 *
*        2 *        0 * 2.8191428 * 2.8191428 *
*        3 *        0 * 2.8773849 * 2.8773849 *
*        4 *        0 * 2.6493008 * 2.6493008 *
*        5 *        0 * 0.7352977 * 0.7352977 *
*        6 *        0 * 1.9785047 * 1.9785047 *
*        7 *        0 * 2.1588175 * 2.1588175 *
*        8 *        0 * 2.4671847 * 2.4616904 *
*        8 *        1 * 2.4616904 *           *
*        9 *        0 * 0.8032970 * 0.8032970 *
************************************************
```

Example: 10 single muon events with truth-seeded tracking

Reconstructed polar angle for unfiltered and final tracks

```
root [4] events->Scan("CentralCKFTrackParametersUnfiltered.theta:CentralCKFTrackParameters.theta","")
************************************************
*    Row   * Instance * CentralCK * CentralCK *
************************************************
*        0 *        0 * 3.0530140 * 3.0530140 *
*        1 *        0 * 2.8717191 * 2.8717191 *
*        2 *        0 * 2.8191428 * 2.8191428 *
*        3 *        0 * 2.8773849 * 2.8773849 *
*        4 *        0 * 2.6493008 * 2.6493008 *
*        5 *        0 * 0.7352977 * 0.7352977 *
*        6 *        0 * 1.9785047 * 1.9785047 *
*        7 *        0 * 2.1588175 * 2.1588175 *
*        8 *        0 * 2.4671847 * 2.4616904 *
*        8 *        1 * 2.4616904 *           *
*        9 *        0 * 0.8032970 * 0.8032970 *
************************************************
```

Note how event 8 contains 2 track candidates. Since this is truth-seeded tracking, both track candidates correspond to the same seed. This can happen in single-particle simulation if there are additional tracker hits caused by secondary particles.

The ambiguity solver keeps the 2nd track candidate. Previously, we would have kept the first one.

Example: 10 single muon events with truth-seeded tracking

Reconstructed polar angle, number of measurements and number of degrees of freedom for unfiltered tracks

```
root [11] events->Scan("CentralCKFTrackParametersUnfiltered.theta:CentralCKFTrajectoriesUnfiltered.nMeasurements:CentralCKFTracksUnfiltered.ndf","")
************************************************************
*    Row   * Instance * CentralCK * CentralCK * CentralCK *
************************************************************
*        0 *        0 * 3.0530140 *         6 *        12 *
*        1 *        0 * 2.8717191 *         4 *         8 *
*        2 *        0 * 2.8191428 *         6 *        12 *
*        3 *        0 * 2.8773849 *         7 *        14 *
*        4 *        0 * 2.6493008 *         3 *         6 *
*        5 *        0 * 0.7352977 *         7 *        14 *
*        6 *        0 * 1.9785047 *         8 *        16 *
*        7 *        0 * 2.1588175 *         8 *        16 *
*        8 *        0 * 2.4671847 *         2 *         4 *
*        8 *        1 * 2.4616904 *         4 *         8 *
*        9 *        0 * 0.8032970 *         6 *        12 *
************************************************************
```

The first track candidate in event 8 only has 2 measurements, so it is removed by the ambiguity solver.

Of these two hits zero, one, or two may have been shared with the other candidate. So, even if it had 3 measurements, it may have been removed if it shared a hit with the other candidate.

Example: 10 single muon events with real-seeded tracking

Reconstructed polar angle for unfiltered and final tracks

```
root [5] events->Scan("CentralCKFSeededTrackParametersUnfiltered.theta:CentralCKFSeededTrackParameters.theta","")
*******************************************
*    Row   * Instance * CentralCK * CentralCK *
*******************************************
*        0 *        0 * 3.0530142 * 3.0530138 *
*        0 *        1 * 3.0530133 *           *
*        0 *        2 * 3.0530138 *           *
*        1 *        0 * 2.8718557 * 2.8717186 *
*        1 *        1 * 2.8717186 *           *
*        1 *        2 * 2.8717188 *           *
*        1 *        3 * 2.8717176 *           *
*        1 *        4 * 2.8717188 *           *
*        2 *        0 * 2.8191442 * 2.8191428 *
*        2 *        1 * 2.8191428 *           *
*        2 *        2 * 2.8191428 *           *
*        2 *        3 * 2.8191428 *           *
*        3 *        0 * 2.8774066 * 2.8773849 *
*        3 *        1 * 2.8773849 *           *
*        3 *        2 * 2.8773849 *           *
*        3 *        3 * 2.8773849 *           *
*        3 *        4 * 2.8773849 *           *
*        4 *        0 * 2.6492986 * 2.6492984 *
*        4 *        1 * 2.6492986 *           *
*        4 *        2 * 2.6492984 *           *
*        5 *        0 * 0.7352977 * 0.7352977 *
*        5 *        1 * 0.7352977 *           *
*        5 *        2 * 0.7352977 *           *
*        6 *        0 * 1.9785049 * 1.9785049 *
*        6 *        1 * 1.9785049 *           *
Type <CR> to continue or q to quit ==>
*        6 *        2 * 1.9785047 *           *
*        7 *        0 * 2.1588177 * 2.1588175 *
*        7 *        1 * 2.1588175 *           *
*        7 *        2 * 2.1588175 *           *
*        7 *        3 * 2.1588175 *           *
*        8 *        0 * 2.4672789 * 2.4616906 *
*        8 *        1 * 2.4616901 *           *
*        8 *        2 * 2.4673416 *           *
*        8 *        3 * 2.4616906 *           *
*        9 *        0 * 0.8032970 * 0.8032970 *
*******************************************
```

Tracks corresponding to the same particle but coming from different seed triplets (‘duplicate’ seeds) are combined.

Reconstruction multiplicity/efficiency: truth-seeded tracking

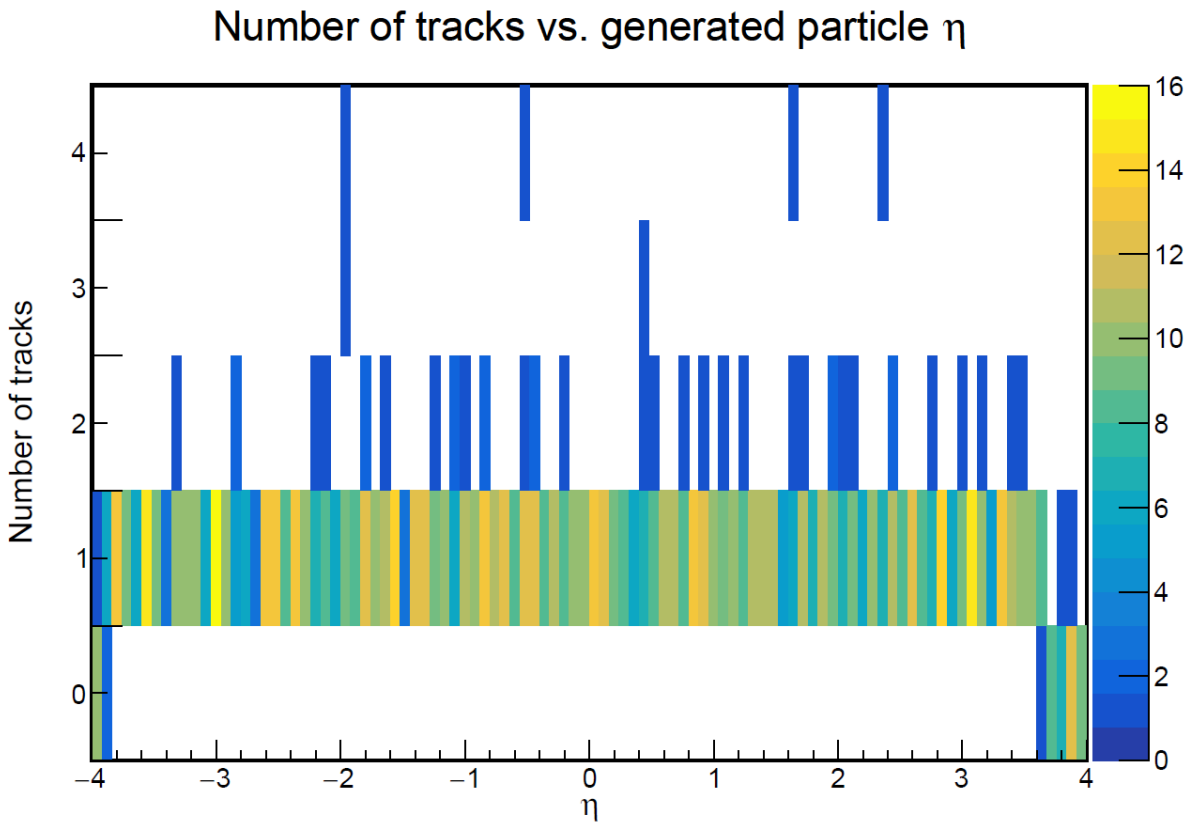
Single μ^- generated:

0.5 GeV/c < P < 20 GeV/c

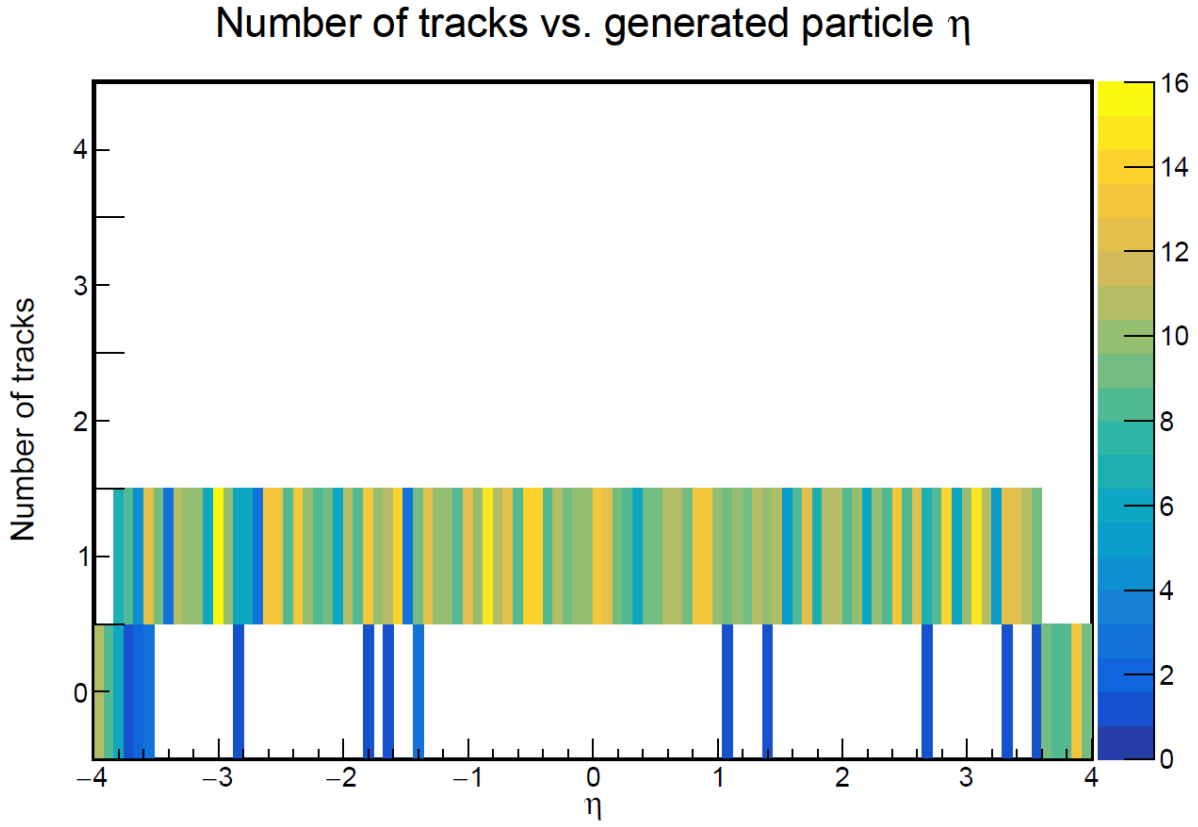
-4 < η < 4

Generated vertex: (0,0,0) mm

Unfiltered tracks



Final tracks



Reconstruction multiplicity/efficiency: truth-seeded tracking

Single μ^- generated:

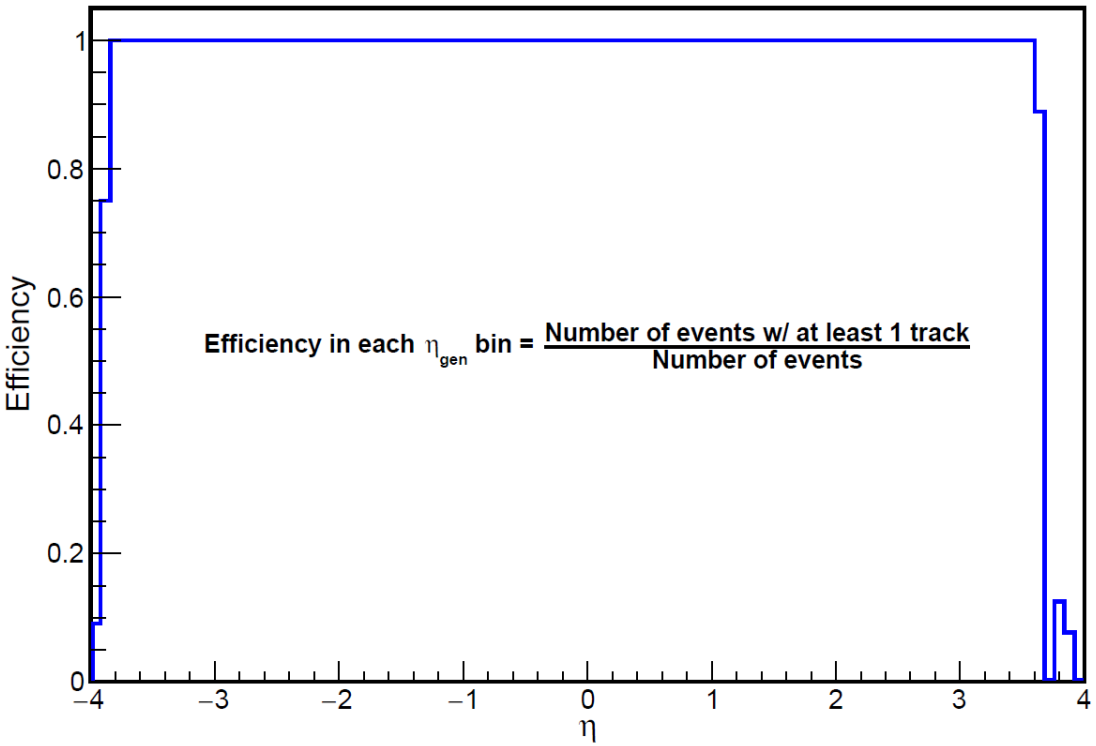
0.5 GeV/c < P < 20 GeV/c

-4 < η < 4

Generated vertex: (0,0,0) mm

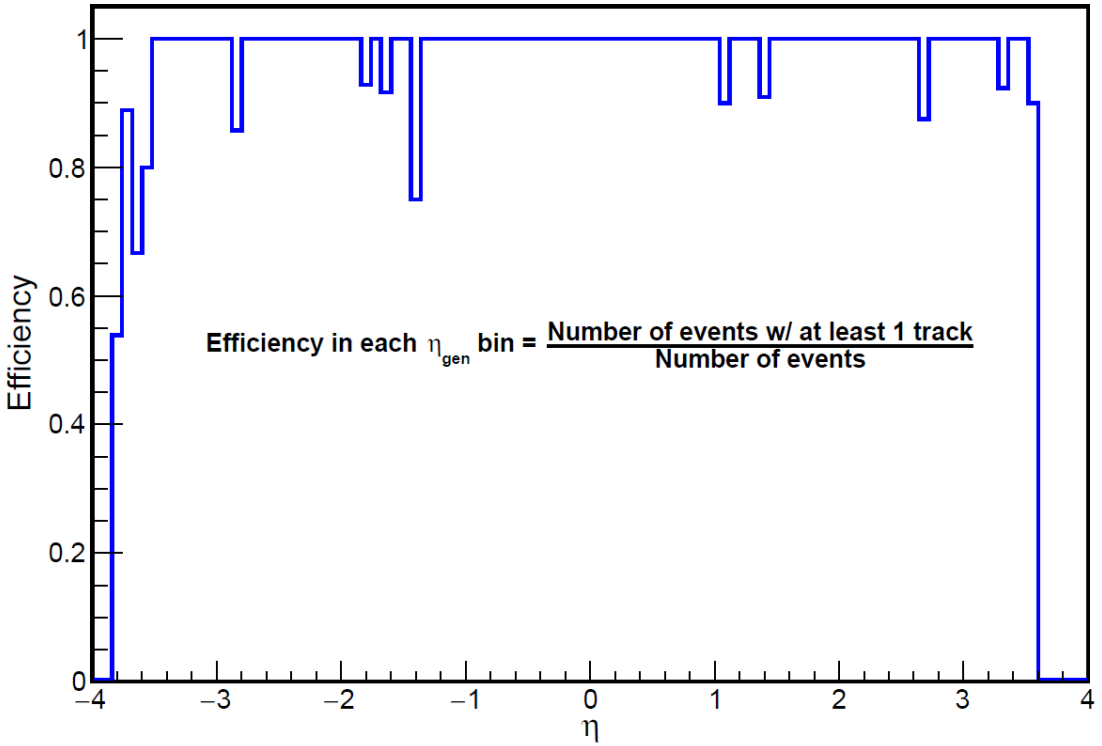
Unfiltered tracks

Tracker Efficiency vs. generated particle η



Final tracks

Tracker Efficiency vs. generated particle η



Reconstruction multiplicity/efficiency: real-seeded tracking

Single μ^- generated:

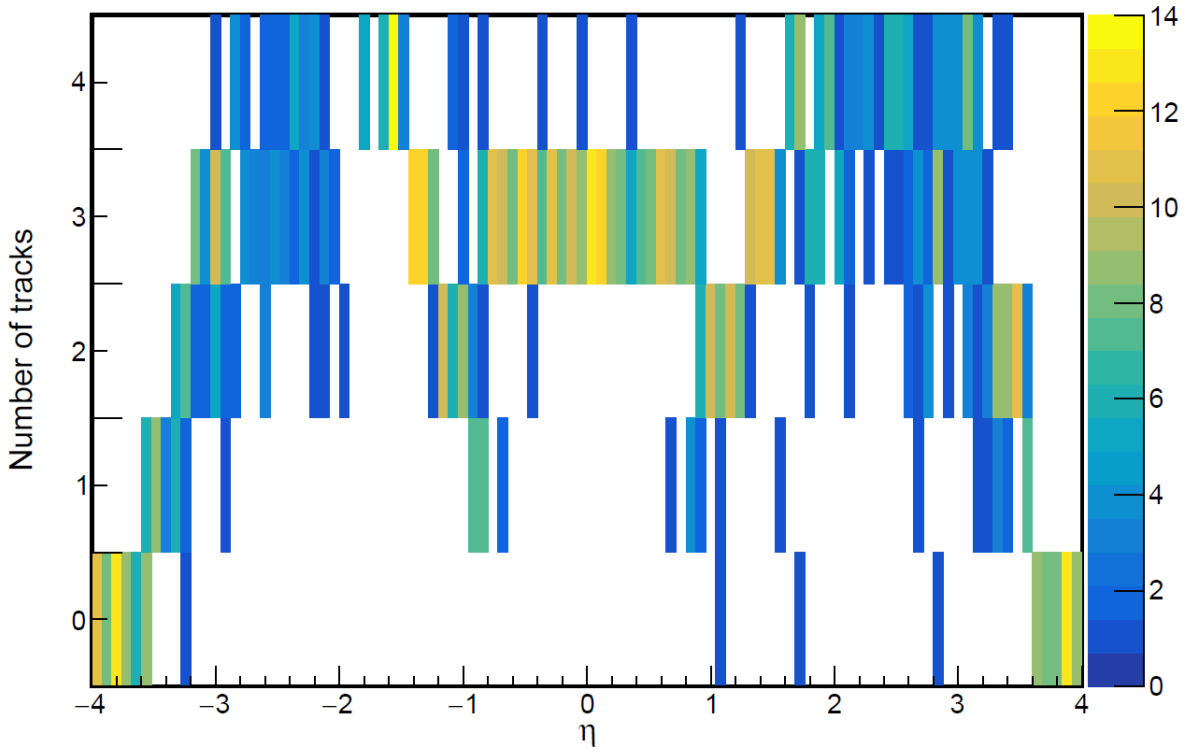
0.5 GeV/c < P < 20 GeV/c

-4 < η < 4

Generated vertex: (0,0,0) mm

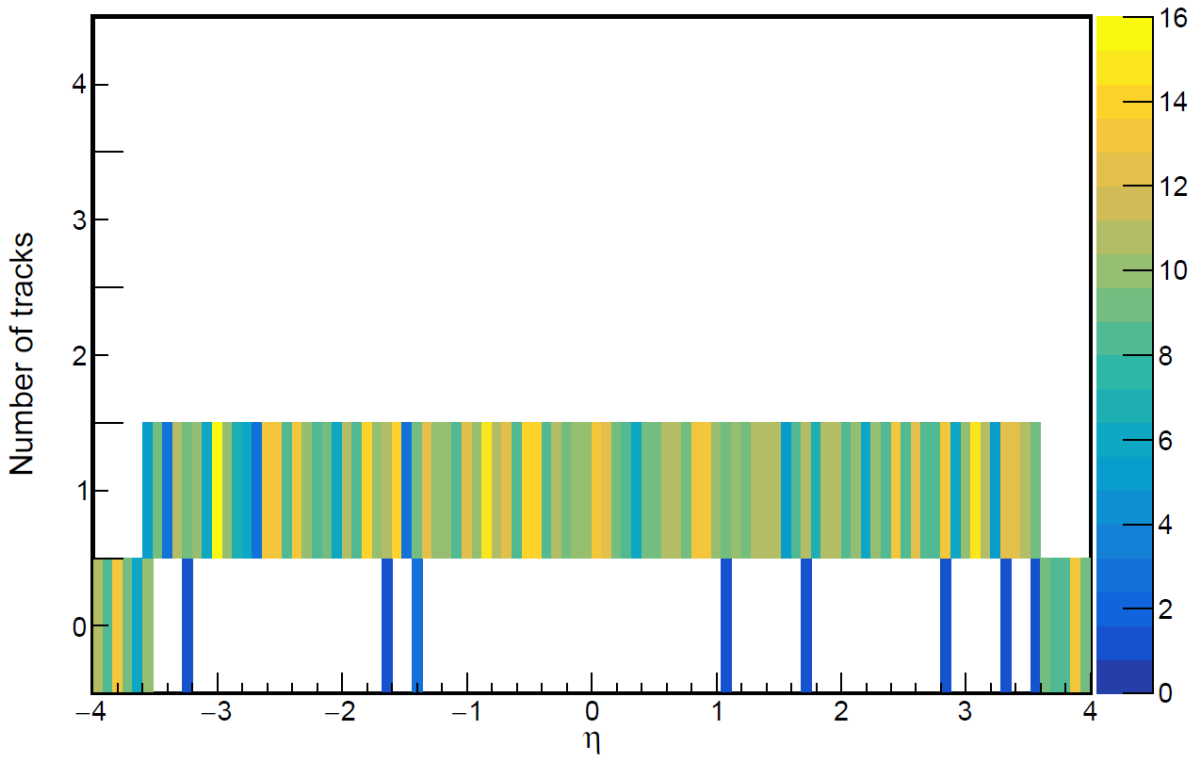
Unfiltered tracks

Number of tracks vs. generated particle η



Final tracks

Number of tracks vs. generated particle η



Reconstruction multiplicity/efficiency: real-seeded tracking

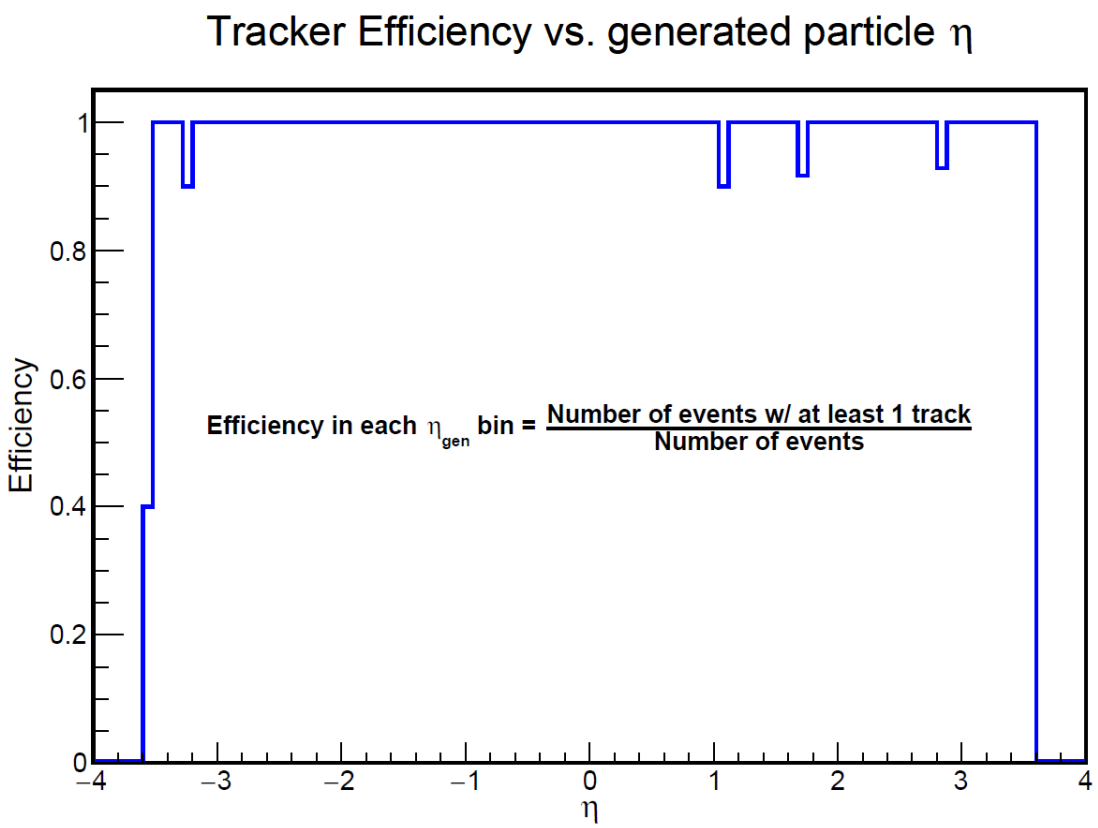
Single μ^- generated:

0.5 GeV/c < P < 20 GeV/c

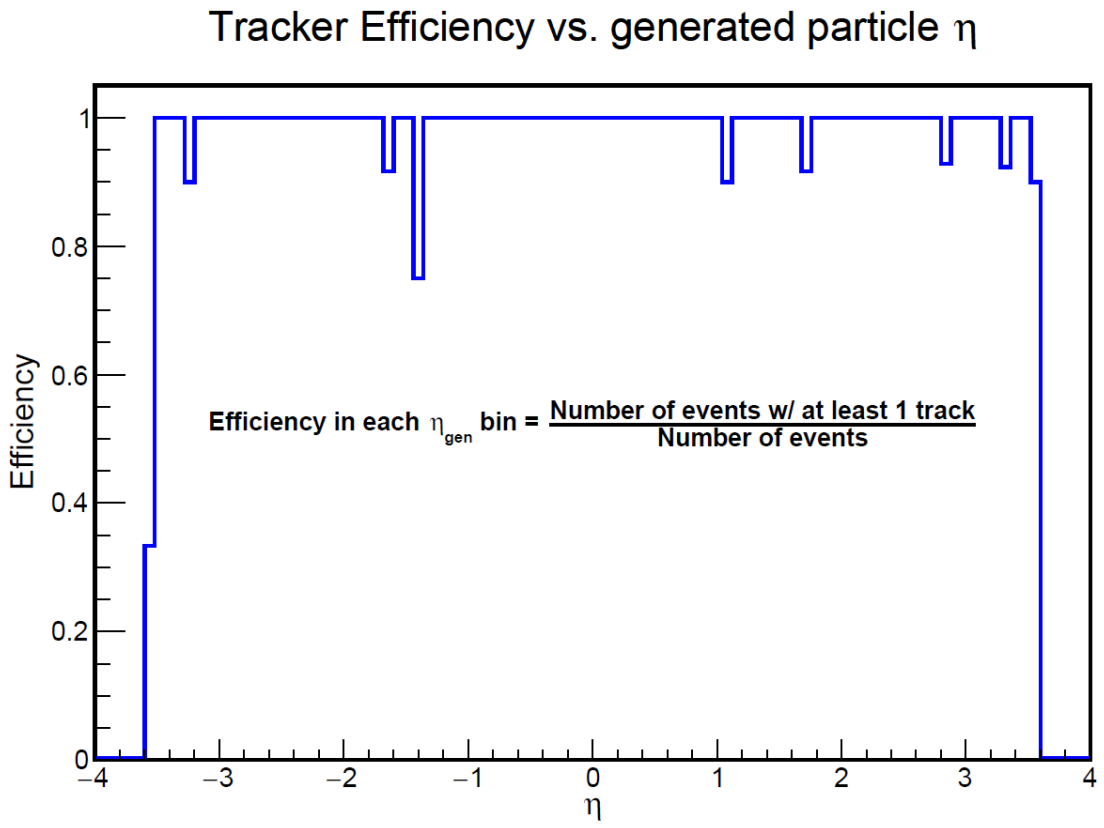
-4 < η < 4

Generated vertex: (0,0,0) mm

Unfiltered tracks



Final tracks



Reconstruction momentum resolution: truth-seeded tracking

Single μ^- generated:

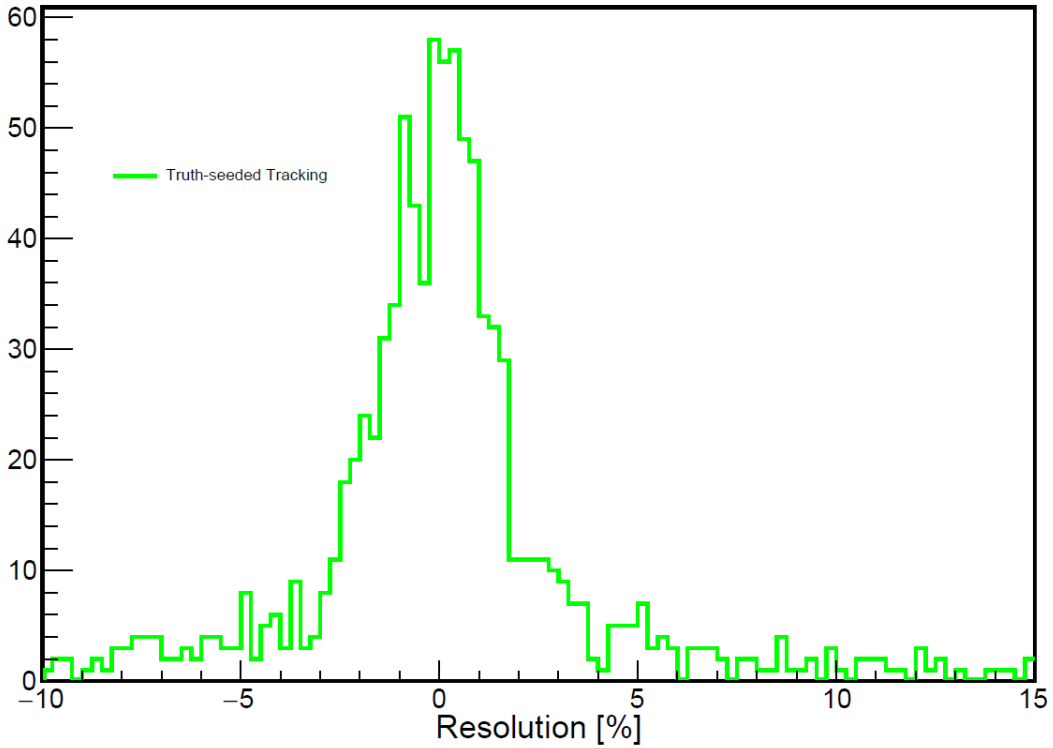
0.5 GeV/c < P < 20 GeV/c

-4 < η < 4

Generated vertex: (0,0,0) mm

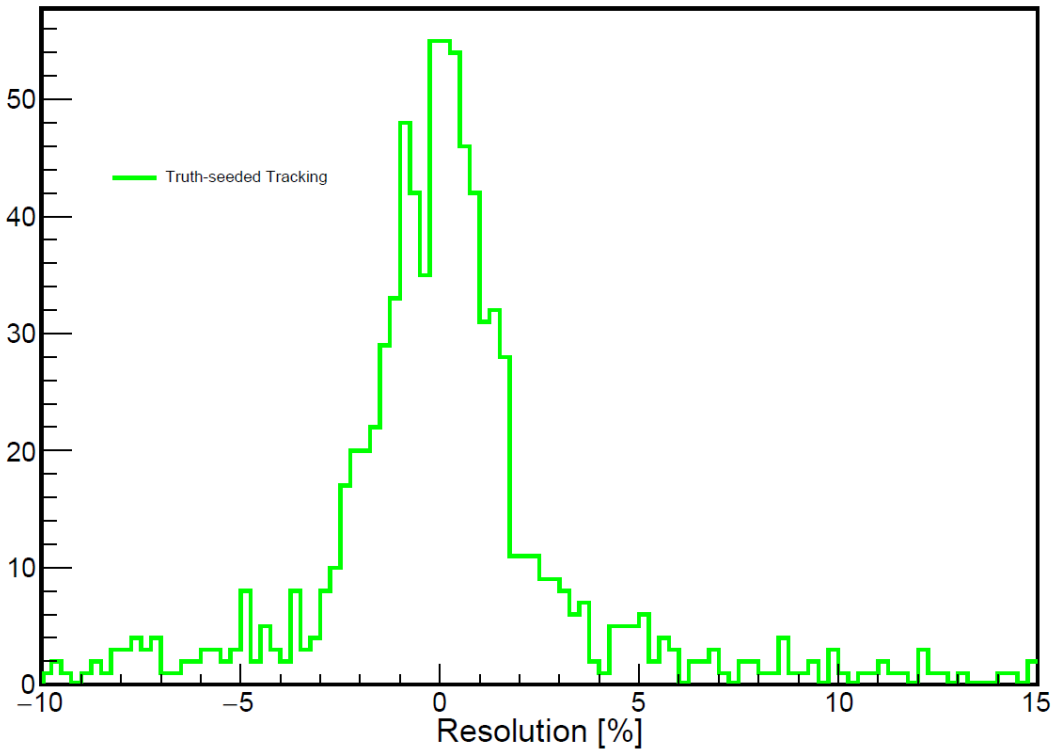
Unfiltered tracks

Momentum Resolution: (rec. - true)/true



Final tracks

Momentum Resolution: (rec. - true)/true



Reconstruction momentum resolution: real-seeded tracking

Single μ^- generated:

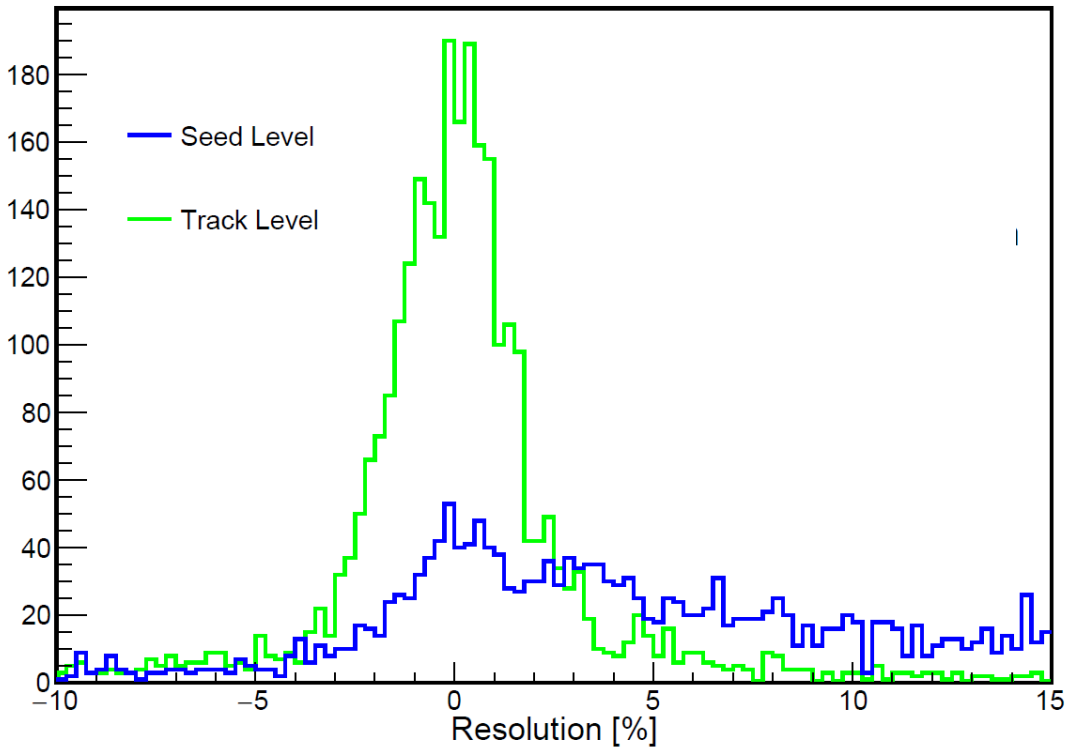
0.5 GeV/c < P < 20 GeV/c

-4 < η < 4

Generated vertex: (0,0,0) mm

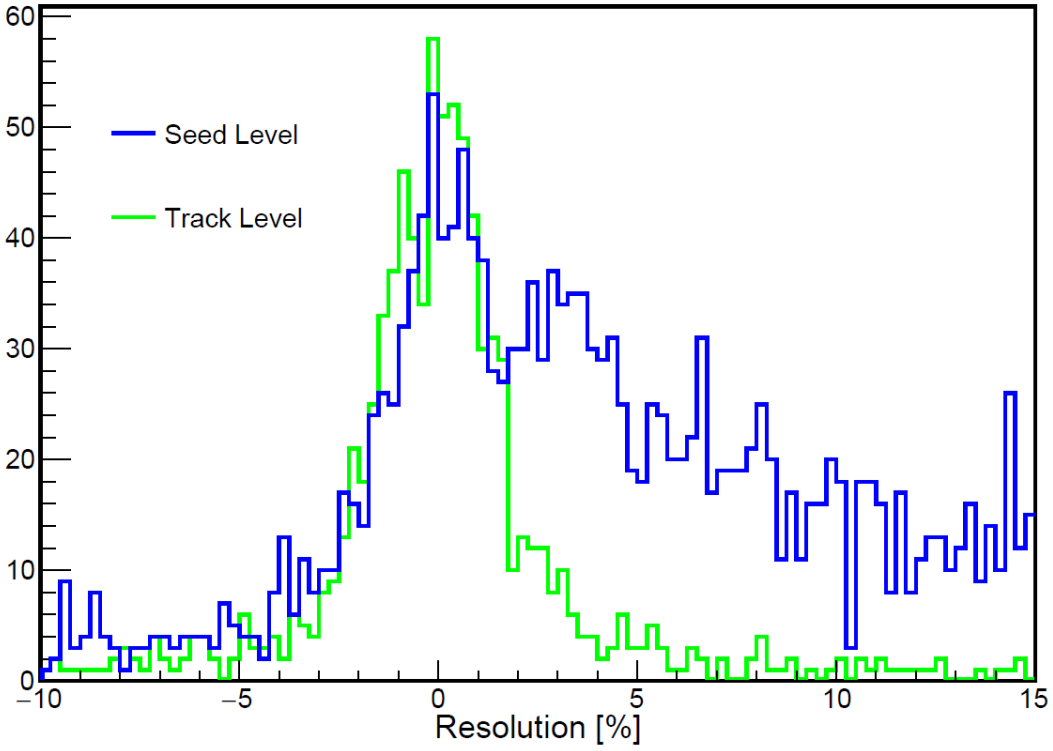
Unfiltered tracks

Momentum Resolution: (rec. - true)/true



Final tracks

Momentum Resolution: (rec. - true)/true



Summary

- The Acts Greedy Ambiguity Resolution Solver has been implemented into EICRecon. It is included in the latest official release.
- We have provided a summary of the workings and effects of the algorithm for both truth-seeded tracking and real-seeded tracking.
- Additional single-particle studies and studies with DIS events should be done as soon as possible.