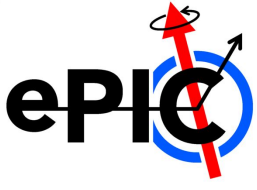
BERKELEY LAB
Bringing Science Solutions to the World

U.S. DEPARTMENT OF ENERGY
Office of Science

ePIC Track Reconstruction Status

Shujie Li
Berkeley Lab

ePIC General Meeting
Feb 10, 2023



Reconstruction Task Forces

Calorimeter Clustering	**Chao Peng** cpeng@anl.gov	**Dmitry Kalinkin** dmitry.kalinkin@gmail.com
Jet Reconstruction	**Derek Anderson** dmawxc@iastate.edu	
PID	**Christopher Dilks** dilks@jlab.org	
Tracking	**Shujie Li** shujieli@lbl.gov	
EICrecon	**Dmitry Romanov** romanov@jlab.org	
Modular Reconstruction	**David Lawrence** davidl@jlab.org	
Simulation Production	**Wouter Deconinck** wouter.deconinck@umanitoba.ca	

track recon task force meeting (the previous EIC-ACTS meeting): 7am PT on Thursdays

https://indico.bnl.gov/category/463/

mailing list:

https://lists.bnl.gov/mailman/listinfo/eic-projdet-trk-recon-l

Joosten
sjoosten@anl.gov

- **Next simulation campaign**:
  - **Target date**: End of March. **tbd**
  - **Goal**: Improved software stack for the reconstruction, including benchmarks.

See https://indico.bnl.gov/event/18274/#1-announcement for details

Track Reconstruction in EICrecon

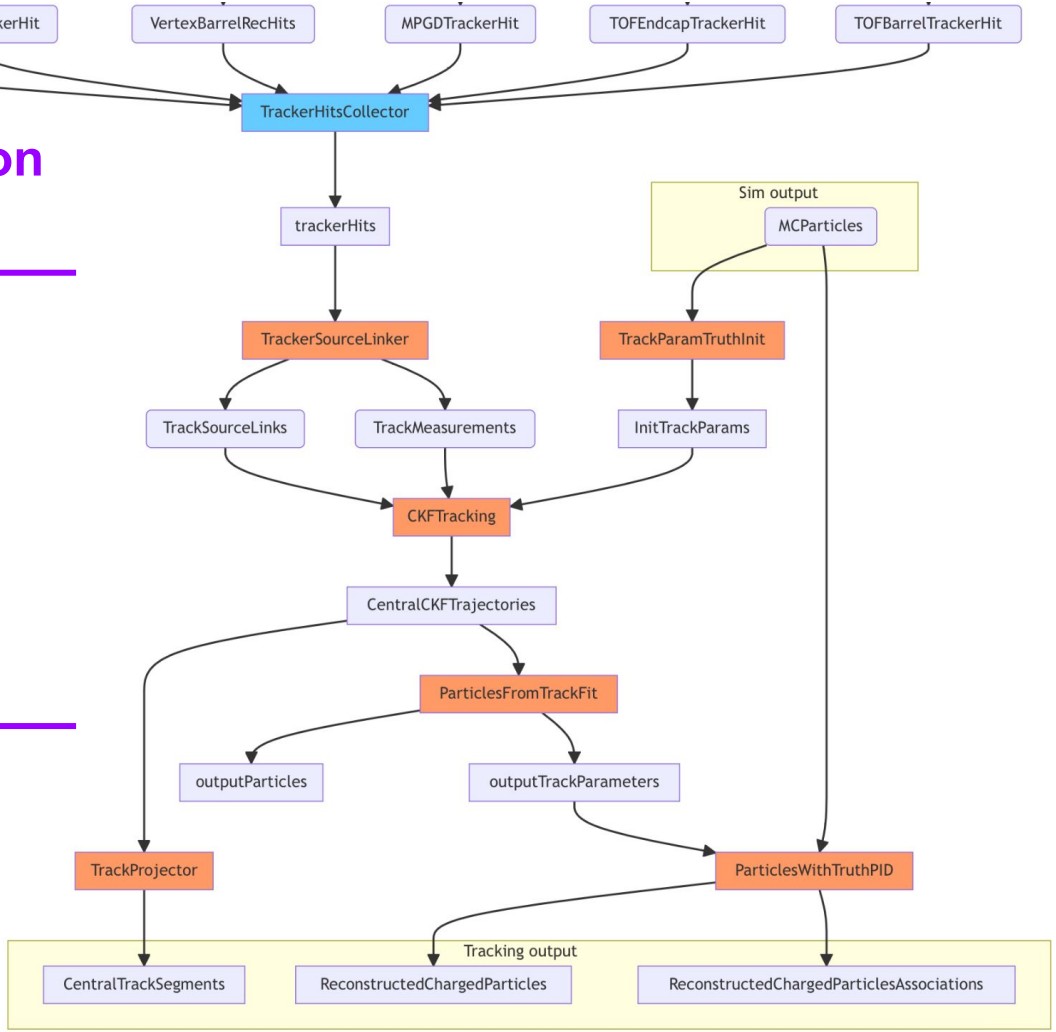
contact: Dmitry Romanov

Full diagram at https://eic.github.io/EICrecon/#/design/tracking?id=full-diagram

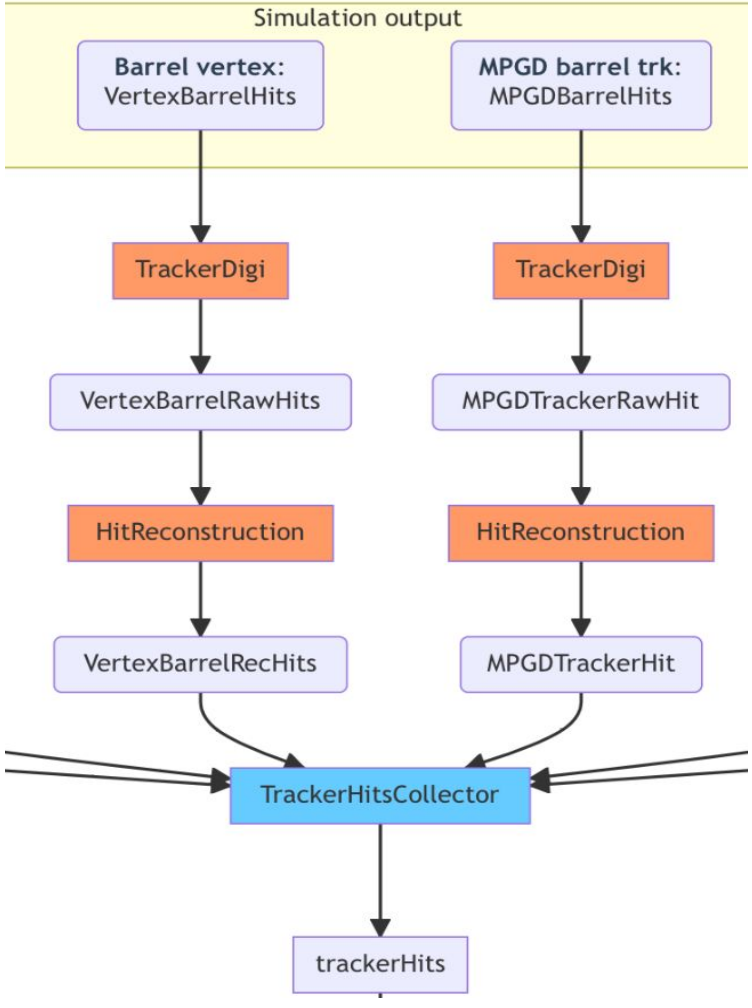
Space point formation

Track finding/fitting with

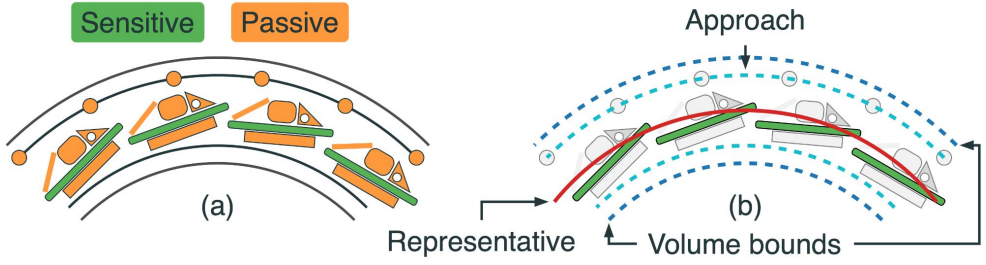
Track info in output



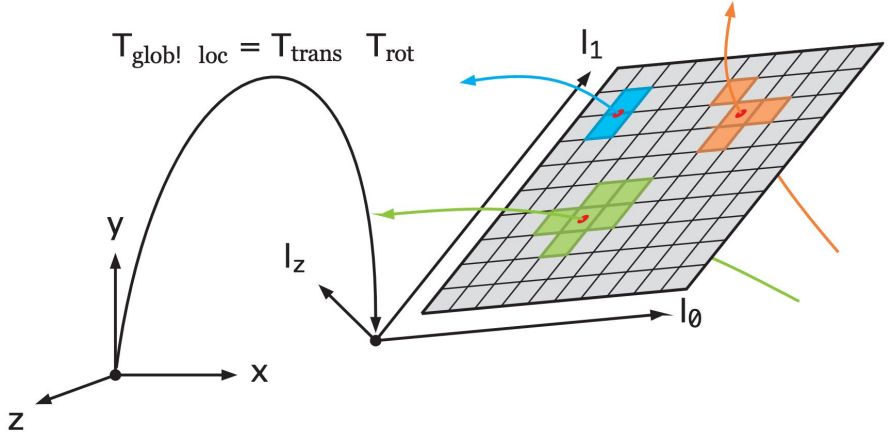
Space Point Formation



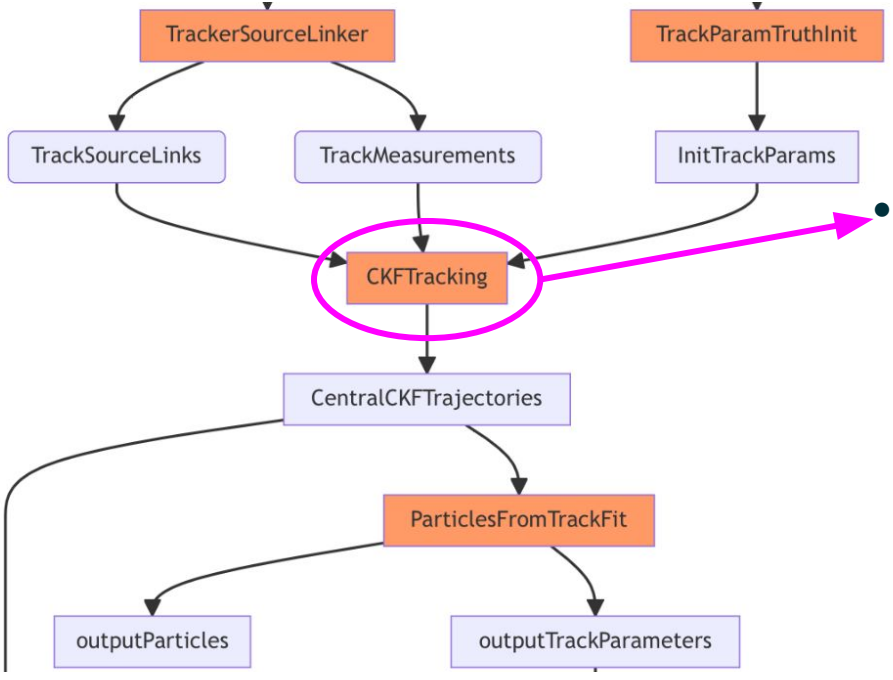
Simplified geometry in ACTS:
detector **volumes** → representing **surfaces**



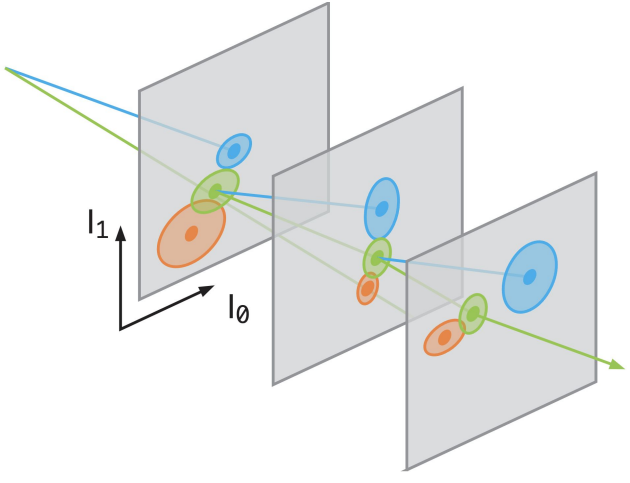
- Global / local coord. transformation
- Associate hits to surfaces
  - Raw hit positions are used, no clusterization
  - ACTS “on surface” check for B0 tracker, see [Sakib Rahman’s work]



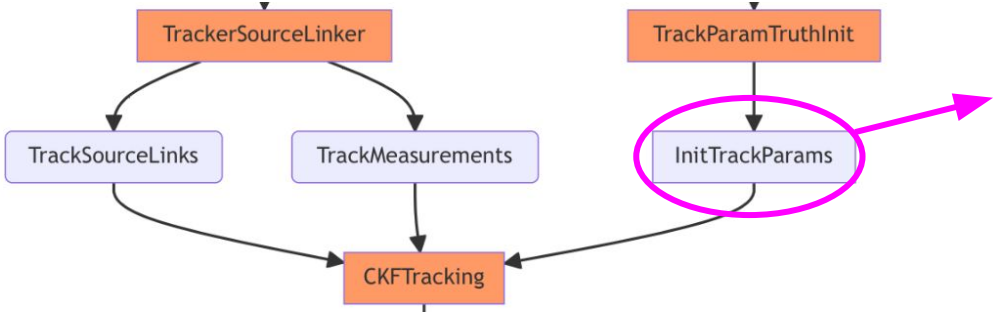
Track Finding/Fitting with ACTS



- **Combinatorial Kalman Filter (CKF)**
  - combine track finding and fitting
  - allows track branching
    - → user-defined measurement selector (number, chi2)
  - high efficiency
  - **Need a reasonable "initial guess"**



Initial Guess for CKF: 1. truth seeding



Use truth initial momentum, angle etc. from simulation

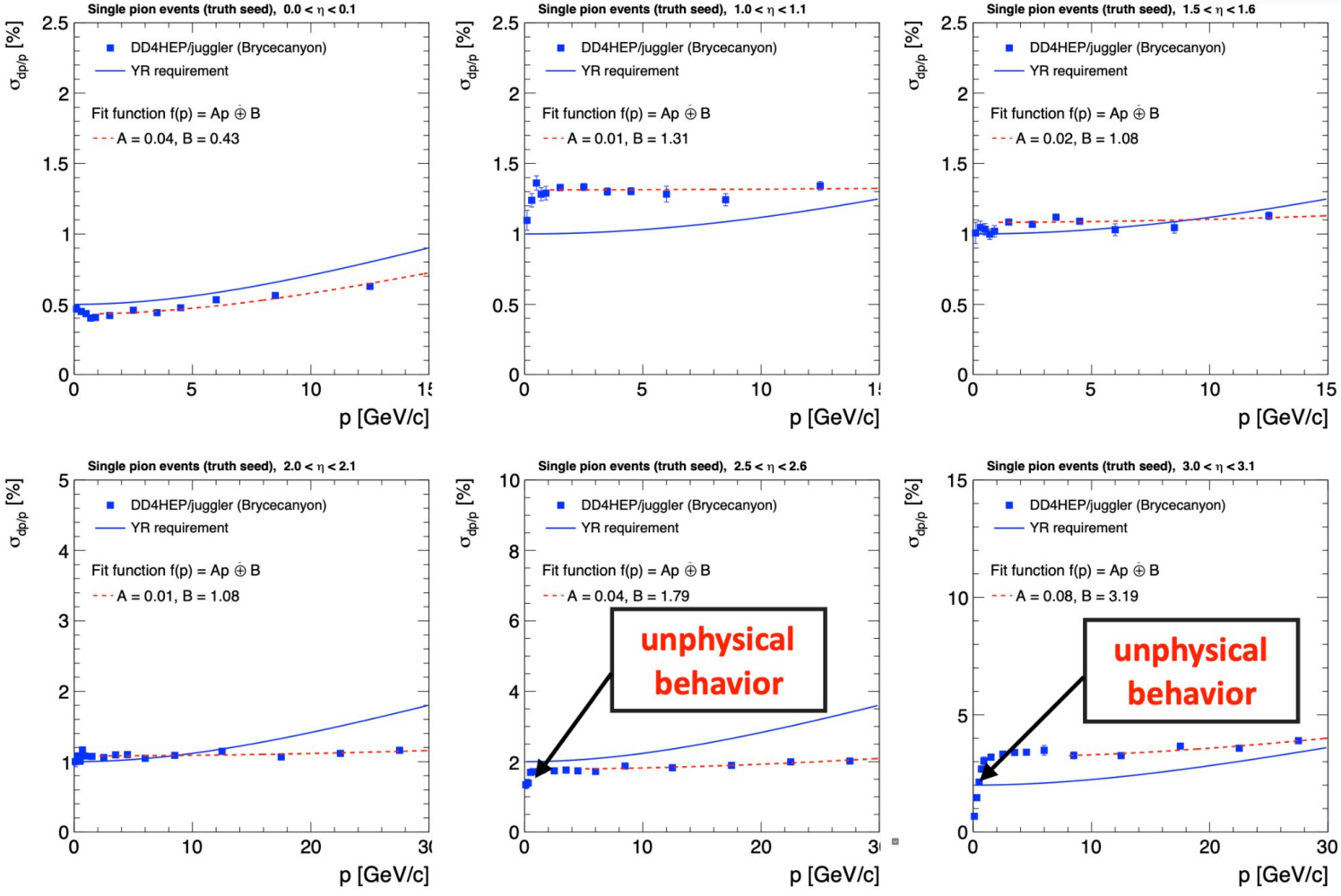
```
// build some track cov matrix
Acts::BoundSymMatrix cov                    = Acts::BoundSymMatrix::Zero();
cov(Acts::eBoundLoc0, Acts::eBoundLoc0)     = 1000*um*1000*um;
cov(Acts::eBoundLoc1, Acts::eBoundLoc1)     = 1000*um*1000*um;
cov(Acts::eBoundPhi, Acts::eBoundPhi)       = 0.05*0.05;
cov(Acts::eBoundTheta, Acts::eBoundTheta)   = 0.01*0.01;
cov(Acts::eBoundQOverP, Acts::eBoundQOverP) = (0.1*0.1) / (GeV*GeV);
cov(Acts::eBoundTime, Acts::eBoundTime)     = 10.0e9*ns*10.0e9*ns;

Acts::BoundVector  params;
params(Acts::eBoundLoc0)   = 0.0 * mm ;  // cylinder radius
params(Acts::eBoundLoc1)   = 0.0 * mm ;  // cylinder length
params(Acts::eBoundPhi)    = phi;
params(Acts::eBoundTheta)  = theta;
params(Acts::eBoundQOverP) = charge / (pinit * GeV);
params(Acts::eBoundTime)   = part->getTime() * ns;
```

https://github.com/eic/EICrecon/blob/main/src/algorithms/tracking/TrackParamTruthInit.cc

Initial Guess for CKF: 1. truth seeding

Tracking performance check with single pion events. See Wenqing Fan's work



Initial Guess for CKF: 1. truth seeding

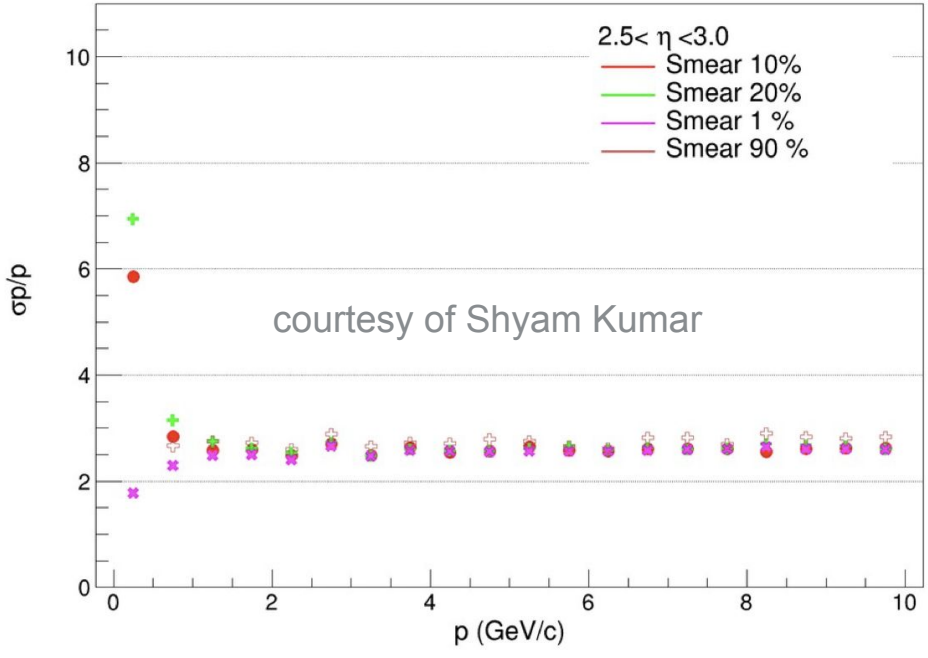
Solutions to the “unphysical behavior” at low momentum:

Short term: add artificial smearing to init parameters. See Shyam Kumar’s work

- address the immediate concern of the reconstruction performance at low p from physics WG
- no clear guidance to a “reasonable” smearing

Long term: use realistic init parameters from seeding

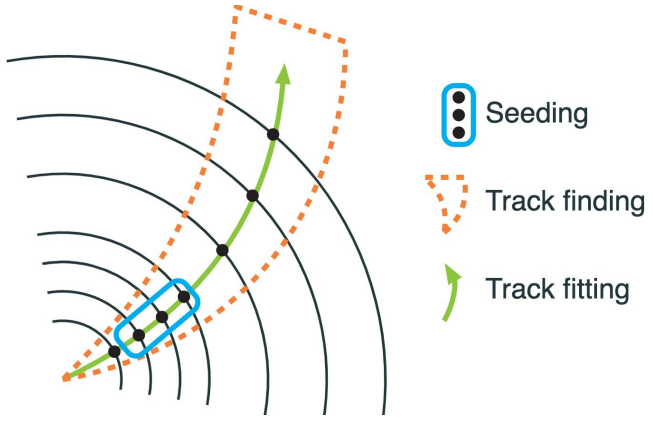
- use smearing parameters from realistic seeding study
- switch to realistic seeding completely



Initial Guess for CKF: 2. realistic seeding

Seeder: a set of three space points to estimate initial track parameters

- **Binned seeder**: loop over φ-z binning to try all combinations. Slow at large η
  - tested and bugs fixed. See YueShi Lai's work
- **Orthogonal seed finder**: can efficiently search for space points within a given range.
  - Initial implementation in EICrecon - Joe Osborn
  - Seeder configuration:
    - default parameters from binned seeder
    - parameter optimization - See Rey Cruz-Torres's work
  - Seeder confirmation/filter

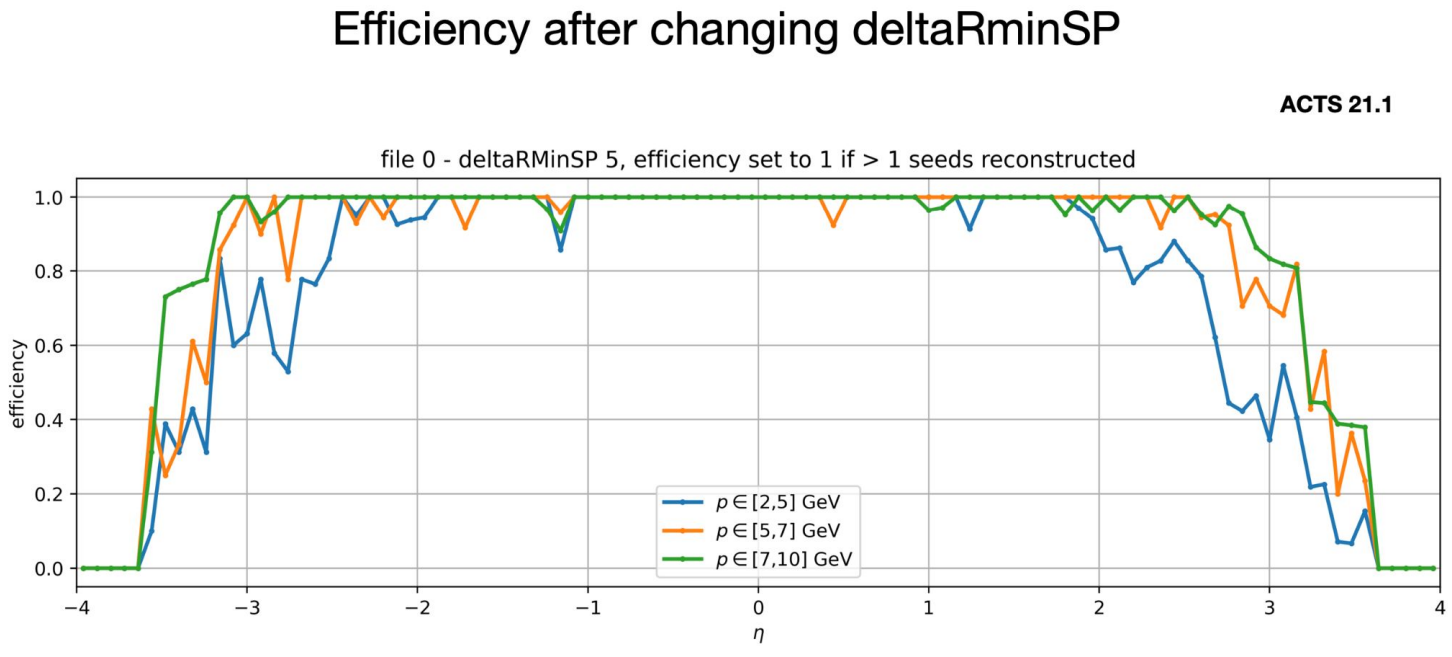


Parameter	Description
bFieldInZ	z component of magnetic field
rMax	Maximum r value to look for seeds
rMin	Minimum r value to look for seeds
zMin	Minimum z value to look for seeds
zMax	Maximum z value to look for seeds
beamPosX	Beam offset in x
beamPosY	Beam offset in y
deltaRMinTopSP	Min distance in r between middle and top SP in one seed
deltaRMinBottomSP	Min distance in r between middle and bottom SP in one seed
deltaRMaxTopSP	Max distance in r between middle and top SP in one seed
deltaRMaxBottomSP	Max distance in r between middle and top SP in one seed
collisionRegionMin	Min z for primary vertex
collisionRegionMax	Max z for primary vertex
cotThetaMax	Cotangent of max theta angle
minPt	Min transverse momentum
maxSeedsPerSpM	Max number of seeds a single middle space point can belong to - 1
sigmaScattering	How many standard devs of scattering angles to consider
radLengthPerSeed	Average radiation lengths of material on the length of a seed

Initial Guess for CKF: 2. realistic seeding

Seeder Configuration

Example of parameter tuning. Use loose settings to obtain seeds for entire eta range despite of high chance to have duplicated seeds. **Ongoing** work of Rey



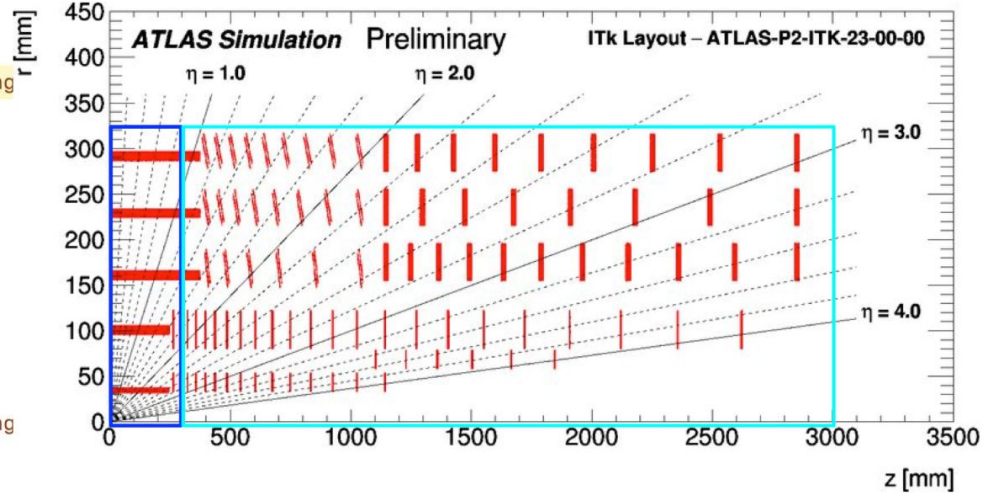
Initial Guess for CKF: 2. realistic seeding

Seed Confirmation/Filter

Individual filter settings for each geometry region.

- Experience from ATLAS-ITK , see Luis Falda Coelho's work
- implementation in EICrecon, **TBD** - Rey, Barak Schmookler

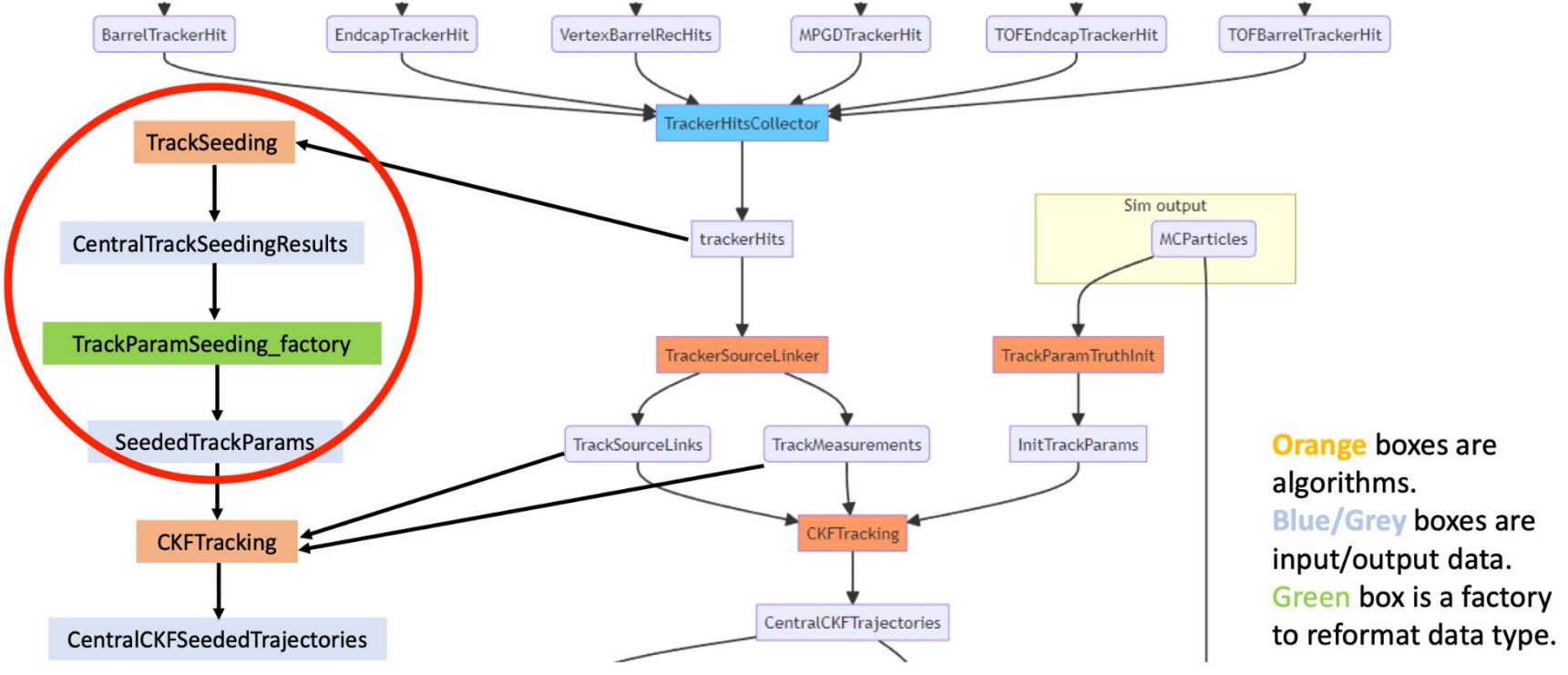
```
centralSeedConfirmationRange = acts.SeedConfirmationRang
    zMinSeedConf=-250 * u.mm,
    zMaxSeedConf=250 * u.mm,
    rMaxSeedConf=140 * u.mm,
    nTopForLargeR=1,
    nTopForSmallR=2,
    seedConfMinBottomRadius=60.0 * u.mm,
    seedConfMaxZOrigin=150.0 * u.mm,
    minImpactSeedConf=1.0 * u.mm,
)  # contains parameters for seed confirmation
forwardSeedConfirmationRange = acts.SeedConfirmationRang
    zMinSeedConf=-3000 * u.mm,
    zMaxSeedConf=3000 * u.mm,
    rMaxSeedConf=140 * u.mm,
    nTopForLargeR=1,
    nTopForSmallR=2,
    seedConfMinBottomRadius=60.0 * u.mm,
    seedConfMaxZOrigin=150.0 * u.mm,
    minImpactSeedConf=1.0 * u.mm,
)
```



Initial Guess for CKF: 2. realistic seeding

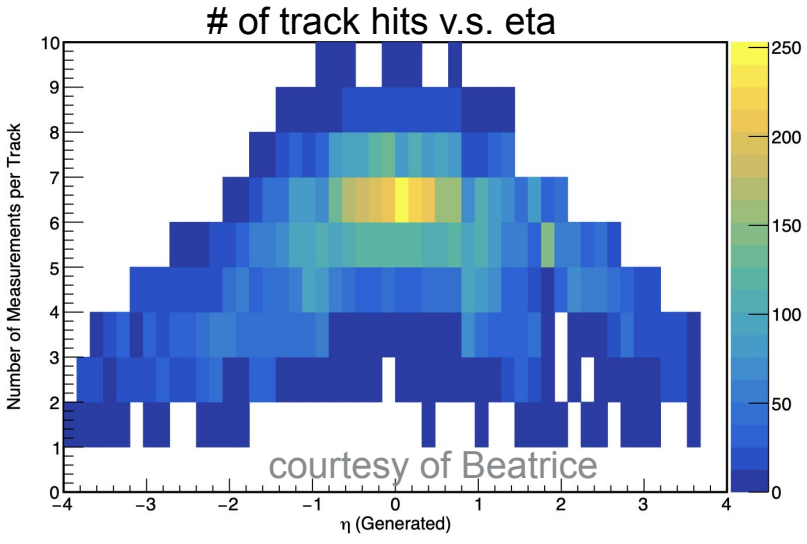
Supply realistic init parameters to CKF

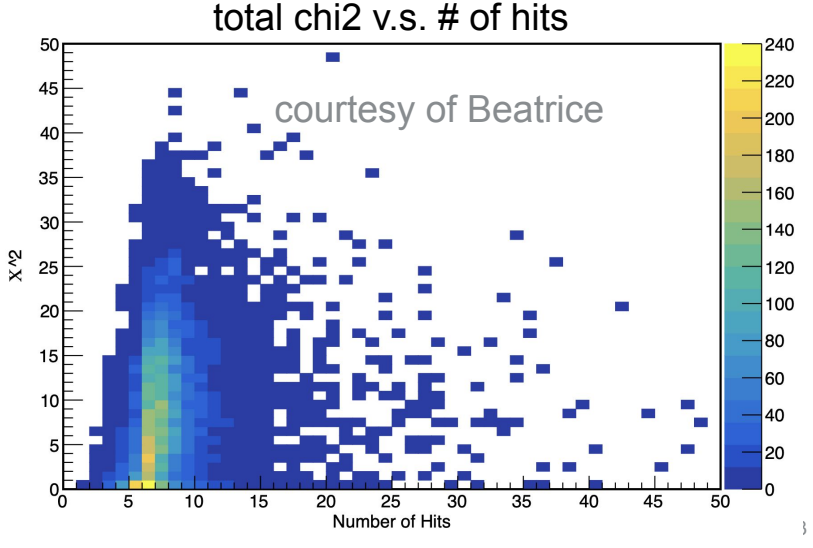
- CKF with realistic seeding in addition to truth seeding. See Barak's work
  - retain data structure for current downstream analysis
- Switch between truth / realistic seeding. TBD. See Dmitry's work



Track Info in Output

- Track parameters from fit - Done
- Track projection - Done
- Trajectory info (chi2, number of hits …)
  - save to histograms with EICrecon plugins:
    - plugin from Nicolas Schmidt https://github.com/eic/EICrecon/blob/angular_studies_tof/src/benchmarks/reconstruction/tracking_studies/tracking_studiesProcessor.cc
    - plugin from Barak https://github.com/eic/EICrecon/tree/track-qa-barak/src/tests/track_qa
    - see also Beatrice Liang-Gilman's study on number of track hits and chi2 (**ongoing**)





Track Info in Output

- Track parameters from fit - Done
- Track projection - Done

- Trajectory info (chi2, number of hits …)
  - save to histograms with EICrecon plugins
  - save to output rootfile:
    - **TBD**: write an EICrecon factory to write trajectory info into data structure

- **TBD:** Hits associated with tracks

```
struct TrajectoryState {
  size_t nStates = 0;
  size_t nMeasurements = 0;
  size_t nOutliers = 0;
  size_t nHoles = 0;
  double chi2Sum = 0;
  std::vector<double> measurementChi2 = {};
  std::vector<double> outlierChi2 = {};
  size_t NDF = 0;
  std::vector<unsigned int> measurementVolume = {};
  std::vector<unsigned int> measurementLayer = {};
  std::vector<unsigned int> outlierVolume = {};
  std::vector<unsigned int> outlierLayer = {};
  size_t nSharedHits = 0;
};
```

trajectory info from ACTS

```
eicd::Trajectory:
  Description: "Raw trajectory from the tracking algorithm"
  Author: "S. Joosten, S. Li"
  Members:
    - uint32_t          type             // 0 (does not have good track fit), 1 (has good track fit)
    - uint32_t          nStates          // Number of tracking steps
    - uint32_t          nMeasurements    // Number of hits used
    - uint32_t          nOutliers        // Number of hits not considered
    - uint32_t                           hits
    - float
    - uint32_t          ndf              // Number of degrees of freedom
    - uint32_t          nSharedHits      // Number of shared hits with other trajectories
  VectorMembers:
    - float             measurementChi2  // Chi2 for each of the measurements
    - float             outlierChi2      // Chi2 for each of the outliers
  OneToOneRelations:
    - eicd::TrackParameters trackParameters // Associated track parameters, if any
  OneToManyRelations:
    - eicd::TrackerHit  measurementHits  // Measurement hits used in this trajectory
    - eicd::TrackerHit  outlierHits      // Outlier hits not used in this trajectory
```

data structure for EICrecon

Track Recon Validation Plots

tracking performance:

- *momentum resolution,*
- *DCA_2D resolutions,*
- *polar angle resolution (at vertex and the various PID detector),*
- *azimuthal angle resolution (at vertex and the various PID detector)*

tracking hits information:

- *hit at generation level,*
- *hit at reconstruction level.*
- *hit locations (x,y,z) associated with tracks*

tracking quality variables (truth + realistic seeding):

- *chi^2 (total, per DOF),*
- *No. of hits of the reconstructed tracks*
- *No. of trajectories per event*
- *No. of seeders (for realistic seeding)*

Proposed by tracking WG, Matt Posik, and LBL group. To be discussed with software/computing WG.

Summary

- Tracker in Acts geometry:
  - central tracker in good shape. B0 tracker needs to be integrated
- Track reconstruction:
  - recon with truth info works well except at low momentum. Parameters need to be smeared.
  - realistic seeder needs to be tuned.
- Track recon quality check:
  - some tracking info is accessible via EICrecon plugins but not in rootfiles
  - need to include validation plots in benchmark tests.

Thank You

Thank You