Summary of track reconstruction workfests in July 2024 ePIC collaboration meeting

Barak Schmookler

Session Overview

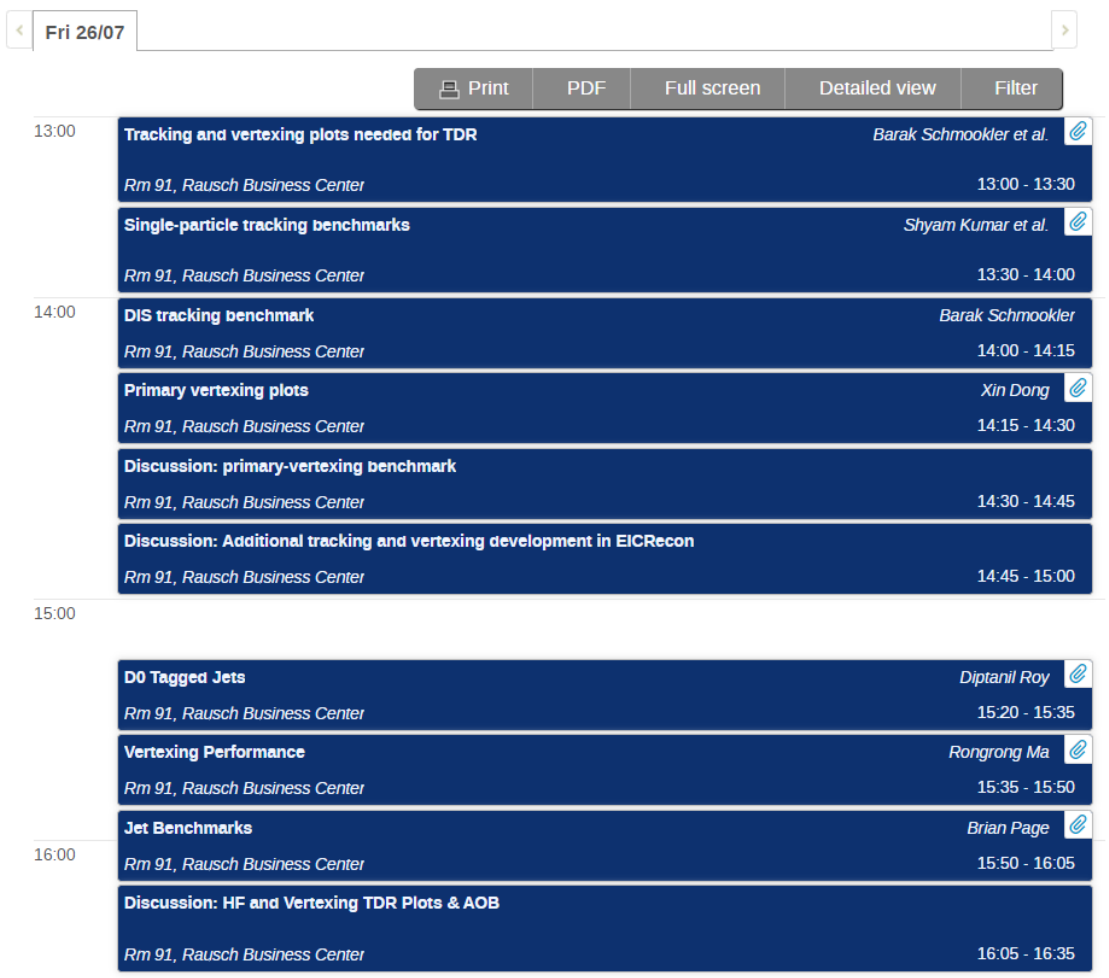
Thu 25/07

Print | PDF | Full screen | Detailed view | Filter

08:00
- **Introduction** — *Rm 137, Rauch Business Center* — 08:00 - 08:15
- **Tracking (reconstruction workflow)** — *Shujie Li* — *Rm 137, Rauch Business Center* — 08:15 - 08:30
- **Calorimetry and Particle Flow (reconstruction workflow)** — *Derek Anderson* — *Rm 137, Rauch Business Center* — 08:30 - 08:45
- **Vertexing (reconstruction workflow)** — *Nicolas Schmidt et al.* — *Rm 137, Rauch Business Center* — 08:45 - 09:00

09:00
- **Electron ID (reconstruction workflow)** — *Tyler Kutz* — *Rm 137, Rauch Business Center* — 09:00 - 09:15
- **PID (reconstruction workflow)** — *Chandradoy Chatterjee et al.* — *Rm 137, Rauch Business Center* — 09:15 - 09:30
- **Far forward (reconstruction workflow)** — *Alexander Jentsch* — *Rm 137, Rauch Business Center* — 09:30 - 09:45
- **Coffee break** — *Rm 137, Rauch Business Center* — 09:45 - 10:00

10:00
- **Track-cluster matching (integration discussion)** — *Tyler Kutz* — *Rm 137, Rauch Business Center* — 10:00 - 10:20
- **Cluster shape parameters (integration discussion)** — *Rm 137, Rauch Business Center* — 10:20 - 10:40



Fri 26/07

Print | PDF | Full screen | Detailed view | Filter

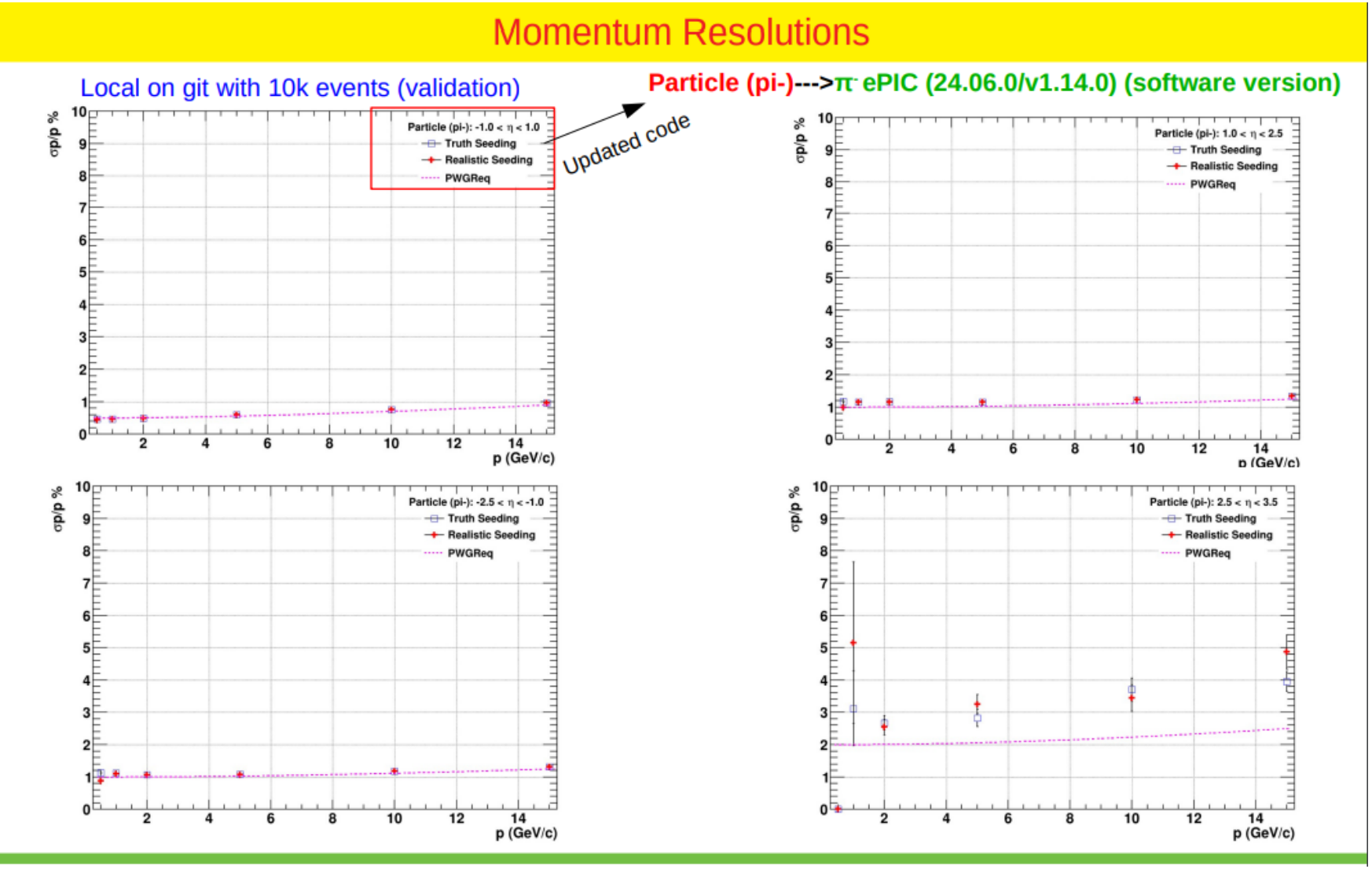
13:00
- **Tracking and vertexing plots needed for TDR** — *Barak Schmookler et al.* — *Rm 91, Rausch Business Center* — 13:00 - 13:30
- **Single-particle tracking benchmarks** — *Shyam Kumar et al.* — *Rm 91, Rausch Business Center* — 13:30 - 14:00

14:00
- **DIS tracking benchmark** — *Barak Schmookler* — *Rm 91, Rausch Business Center* — 14:00 - 14:15
- **Primary vertexing plots** — *Xin Dong* — *Rm 91, Rausch Business Center* — 14:15 - 14:30
- **Discussion: primary-vertexing benchmark** — *Rm 91, Rausch Business Center* — 14:30 - 14:45
- **Discussion: Additional tracking and vertexing development in EICRecon** — *Rm 91, Rausch Business Center* — 14:45 - 15:00

15:00
- **D0 Tagged Jets** — *Diptanil Roy* — *Rm 91, Rausch Business Center* — 15:20 - 15:35
- **Vertexing Performance** — *Rongrong Ma* — *Rm 91, Rausch Business Center* — 15:35 - 15:50
- **Jet Benchmarks** — *Brian Page* — *Rm 91, Rausch Business Center* — 15:50 - 16:05

16:00
- **Discussion: HF and Vertexing TDR Plots & AOB** — *Rm 91, Rausch Business Center* — 16:05 - 16:35

Single-particle track reconstruction benchmark



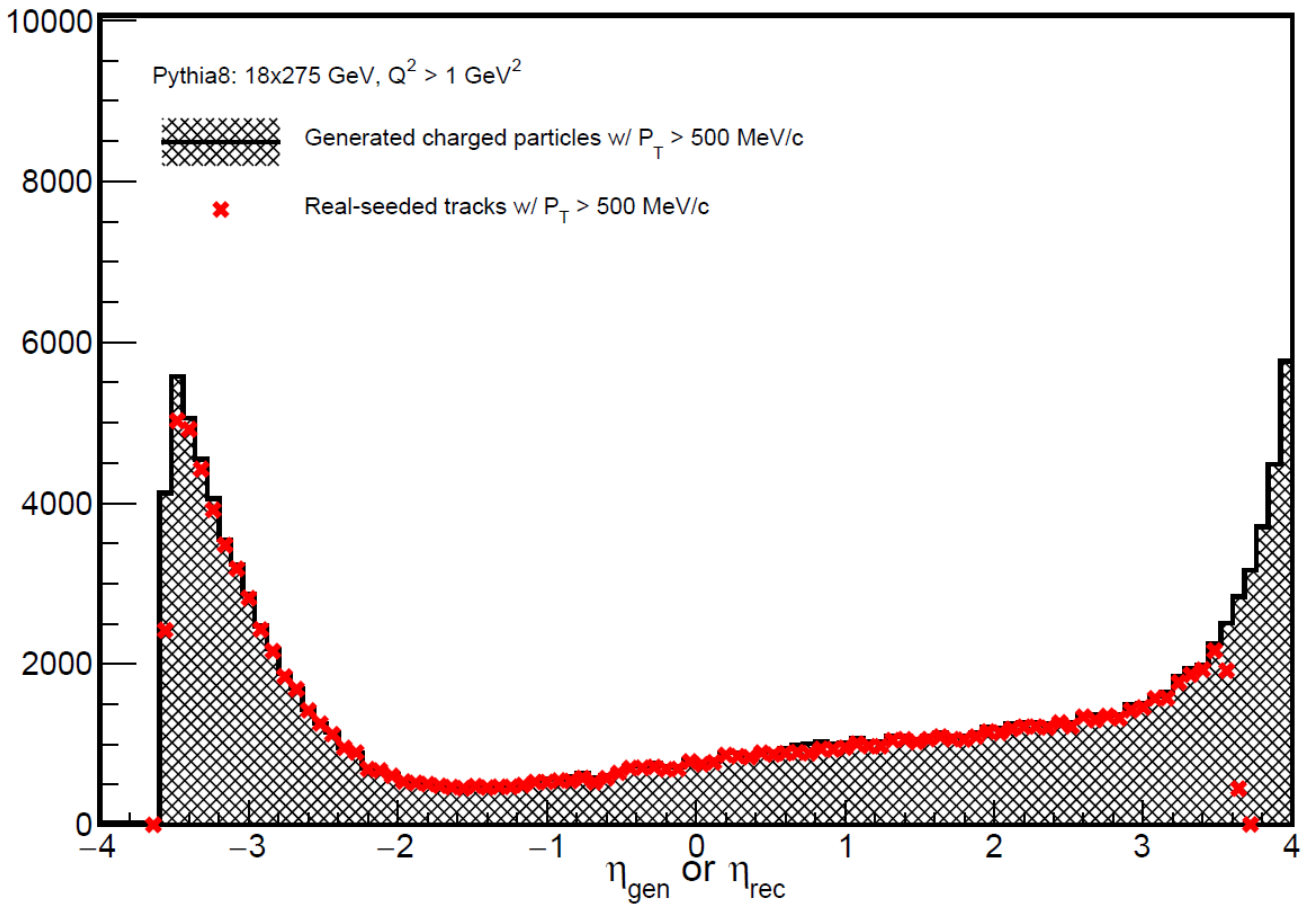
- Tracking benchmark runs on EICweb and produces tracking performance results.
- Benchmark produces single-particle momentum and pointing resolution plots.
- Now that all the machinery exists, the benchmark will be extended to include additional analysis codes for efficiency, angular resolutions, etc...
- Similar set of analysis codes will be run on monthly single-particle simulation campaigns.

Work by Shyam Kumar

DIS tracking benchmark

Benchmark under development

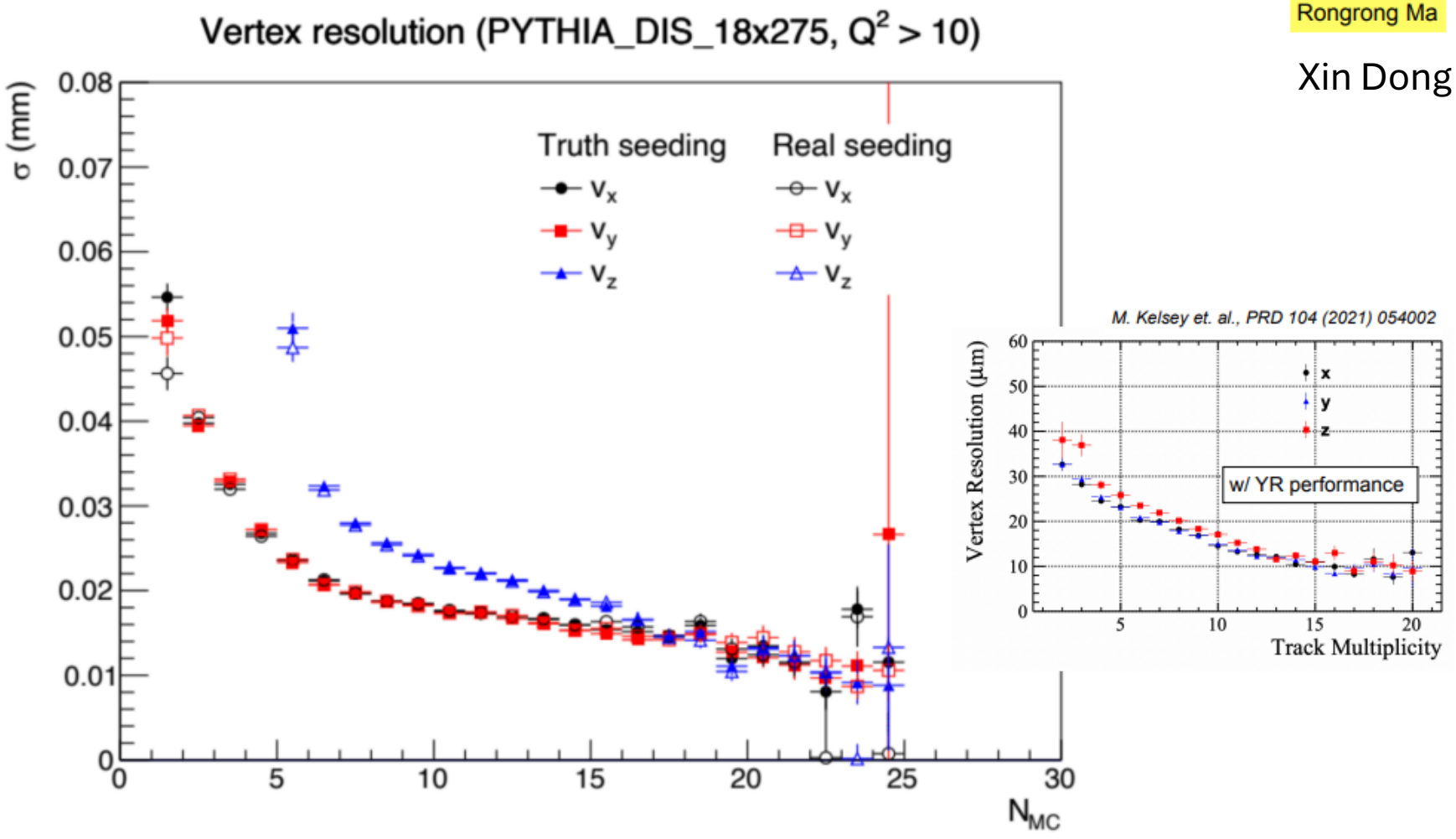
- Simulation of 50k *Pythia8* events with $Q^2 > 1$ GeV^2 at the 18x275 GeV beam energy setting.
- **Black curve**: true pseudo-rapidity distribution of all generated, final-state charged particles with true transverse momentum >500 MeV/c.
- **Red points**: reconstructed pseudo-rapidity distribution of all real-seeded tracks with reconstructed transverse momentum >500 MeV/c.



Primary vertex resolutions – to go into benchmark

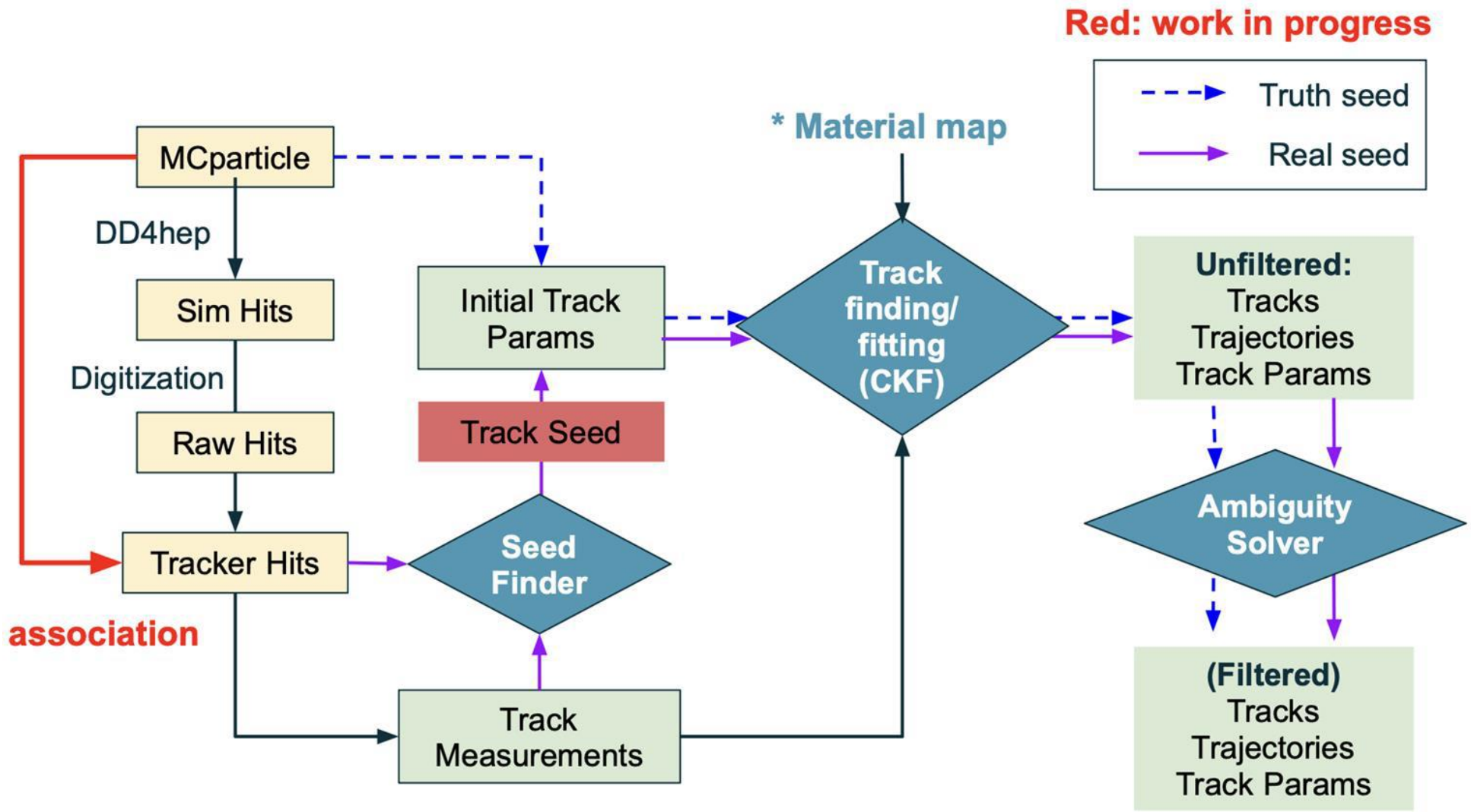
Rongrong Ma

Xin Dong



Vertex resolution reaches to < 20 μm with $N_{MC} > 10$

Track reconstruction / vertexing – status



EICRecon – work in progress

- Fixes to the seed finder to address observed inefficiencies for |z|>50mm. EICRecon branch; PR almost ready.
- Implementation of hit-based track to MC particle matching. Relation between rec and raw tracker hits added. All needed associations/relations exist; now need a factory to make association between track and MC Particle.
- Calculation of track distances/parameters w.r.t measured primary vertex. See next slide.
- Inclusion of option for noise hits and dead pixels in the SVT detector (also sensor unit with inactive area).

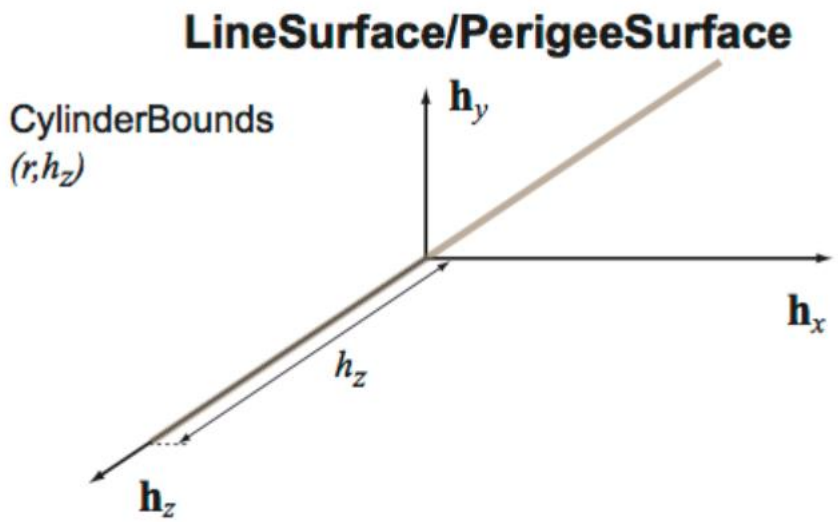
Track parameters w.r.t primary vertex

Perigee surface

class `PerigeeSurface` : *public* Acts::LineSurface

Class describing the Line to which the Perigee refers to.

The Surface axis is fixed to be the z-axis of the Tracking frame. It inherits from StraingLineSurface.



Class Acts::ImpactPointEstimator

template<typename input_track_t, typename propagator_t, typename propagator_options_t = PropagatorOptions<>>
class Acts::ImpactPointEstimator

Estimator for impact point calculations.

Public Functions

inline ImpactPointEstimator(const Config &cfg)

Constructor.

Parameters: **cfg** – Configuration object

Result<double> calculate3dDistance(const GeometryContext &gctx, const BoundTrackParameters &trkParams, const Vector3 &vtxPos, State &state) const

Calculates 3D distance between a track and a 3D point.

Parameters:
- **gctx** – The geometry context
- **trkParams** – Track parameters
- **vtxPos** – Position to calculate distance to
- **state** – The state object

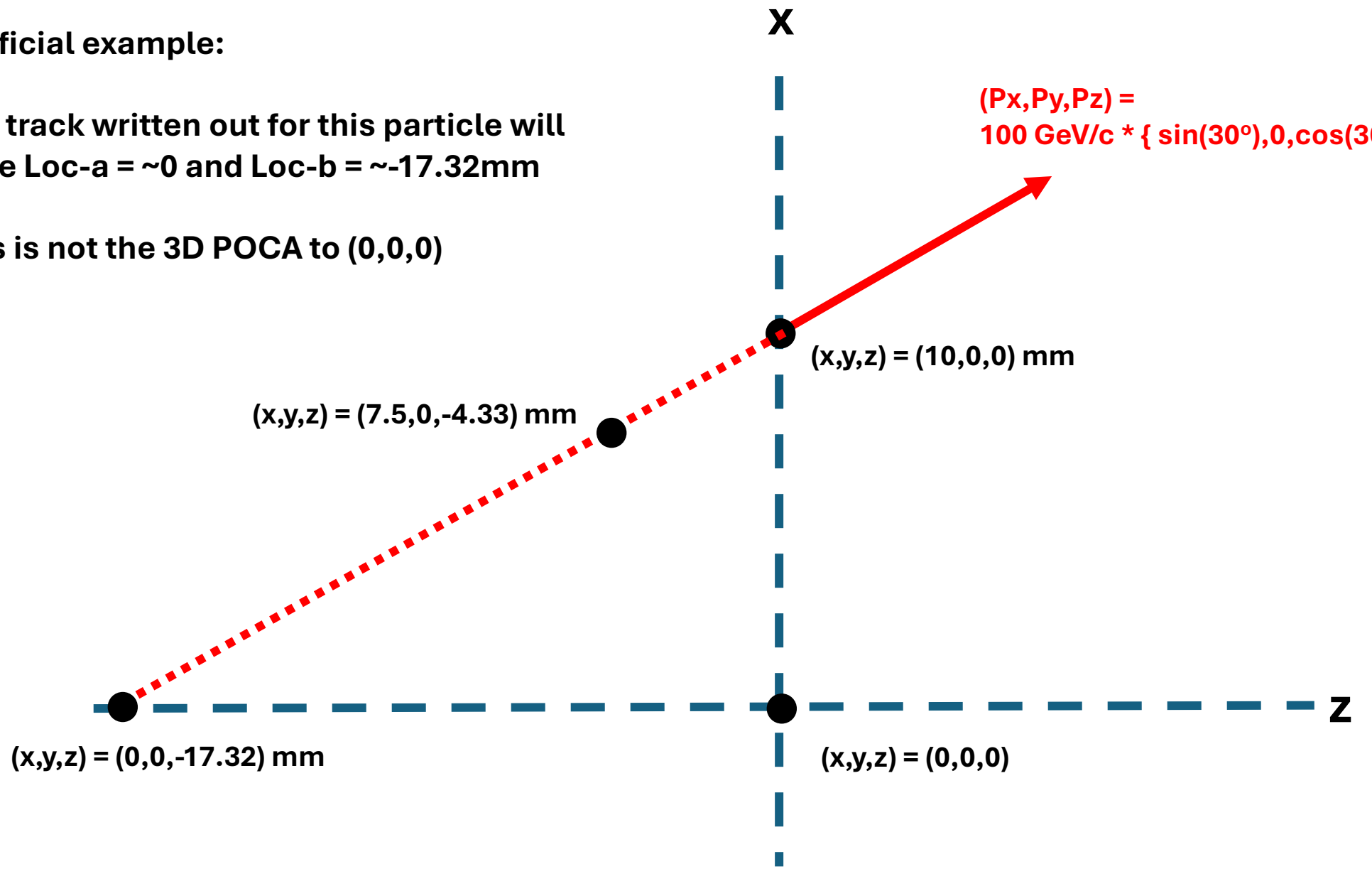
Returns: Distance

Prior to the vertexing, our track parameters are evaluated w.r.t. the perigee surface through (0,0,0)

Artificial example:

The track written out for this particle will have Loc-a = ~0 and Loc-b = ~-17.32mm

This is not the 3D POCA to (0,0,0)



Track parameters w.r.t primary vertex

For secondary tracks, should use this to get 3D DCA to vertex.

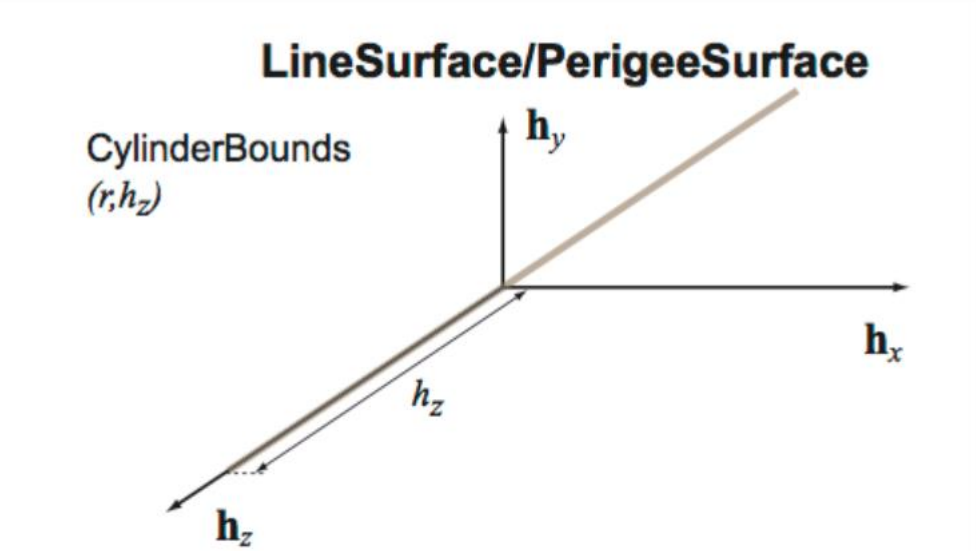
For tracks associated w/ primary vertex, can use either class. (Just make a perigee surface though the vertex.)

Perigee surface

class `PerigeeSurface` : *public* Acts::LineSurface

Class describing the Line to which the Perigee refers to.

The Surface axis is fixed to be the z-axis of the Tracking frame. It inherits from StraingLineSurface.



Class Acts::ImpactPointEstimator

template<typename input_track_t, typename propagator_t, typename propagator_options_t = PropagatorOptions<>>
class Acts::ImpactPointEstimator

Estimator for impact point calculations.

Public Functions

inline ImpactPointEstimator(const Config &cfg)

Constructor.

Parameters: **cfg** – Configuration object

Result<double> calculate3dDistance(const GeometryContext &gctx, const BoundTrackParameters &trkParams, const Vector3 &vtxPos, State &state) const

Calculates 3D distance between a track and a 3D point.

Parameters:
- **gctx** – The geometry context
- **trkParams** – Track parameters
- **vtxPos** – Position to calculate distance to
- **state** – The state object

Returns: Distance

Prior to the vertexing, our track parameters are evaluated w.r.t. the perigee surface through (0,0,0)

Sensor-oriented workfest summaries

SVT
https://indico.bnl.gov/event/20727/contributions/93074/attachments/56183/96165/20240727%20-%20ePIC%20SVT%20report.pdf

MPGD
https://indico.bnl.gov/event/20727/contributions/93067/attachments/56185/96157/MPGD_Workfest_Summary.pdf