Recent updates to Truth Seeding in EICRecon

Barak Schmookler

(most of the actual work done by Wouter and Dmitry)

https://github.com/eic/EICrecon/pull/766 (merged)
https://github.com/eic/EICrecon/pull/783 (ready to merge)

Updates to truth seeding

➢ The CKF requires us to pass a set of seed parameters – charge, momentum, theta, phi, ACTS local coordinates, and a covariance matrix – as well as a reference point (perigee surface) in global coordinates.

```
// Create parameters
acts_track_params.emplace_back(pSurface, params, charge, cov);
```

➢ Previously, the truth seeding was writing to the *eicrecon::TrackParameters* type that is needed for the CKF (shown above). The truth-seeded parameters could not be saved to the EICRecon output ROOT file.

➢ Now, the truth seeding is writing to the *edm4eic::TrackParameters* type. This can be written to the output ROOT file, and we have the same structure and algorithm flow for the truth and real seeds. (We then covert to the format needed for the CKF within the CKF factory.)

➢ We don't want to go back to the previous situation because that would prevent merging of the real seeded tracking branch (or require some additional changes to the data model).

Updates to truth seeding – previous version

➢ As mentioned above, the ACTS requires us to pass the seed local coordinates and the reference point (perigee surface). Previously for the truth seeding, we defined the reference point to be the particle's generation vertex and set the local coordinates to zero.

```
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;


//// Construct a perigee surface as the target surface
auto pSurface = Acts::Surface::makeShared<Acts::PerigeeSurface>(
        Acts::Vector3{part->getVertex().x * mm, part->getVertex().y * mm, part->getVertex().z * mm});

auto result = new eicrecon::TrackParameters({pSurface, params, charge, cov});
```

All this is done in the track seeding code. Without smearing, this should give the ideal track seed for a given particle.

3

Updates to truth seeding – proposed version

➢ Now, we set the reference point to (0,0,0) and use the particle's generation vertex as the local coordinates.

```
auto v = mcparticle.getVertex();
track_parameter.setLoc({static_cast<float>(std::hypot(v.x, v.y)), static_cast<float>(v.z)}); // 2d location on surface [mm]
```

In track seeding code. Perigee surface cannot be saved into the *edm4eic::TrackParameters* data container.

```
// Construct a perigee surface as the target surface
auto pSurface = Acts::Surface::makeShared<const Acts::PerigeeSurface>(Acts::Vector3(0,0,0));
```

In CKF factory which will be used for both truth and real seeded tracking. The reference point is set to the origin. Note that for the real seeds, we fit 3 points and find the point of closest approach in the (x,y) plane, and extract the z value when r=0 from a linear fit. So, using the origin as the reference point makes sense.

Updates to truth seeding – one thing to check

➢ For particles with a generation vertex on the z-axis, the results should be equivalent for the previous and proposed versions.

➢ However, for non-zero (x,y) generated vertex coordinates, note that the old version gives a specific point in space, while the new version gives a specific z-value but only specifics a circle in the (x,y) plane.

➢ So, for secondary particles where the particle's generation vertex is far from the z-axis, there may be differences between the old and new versions based on how the CKF treats the input.
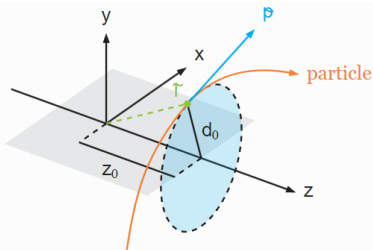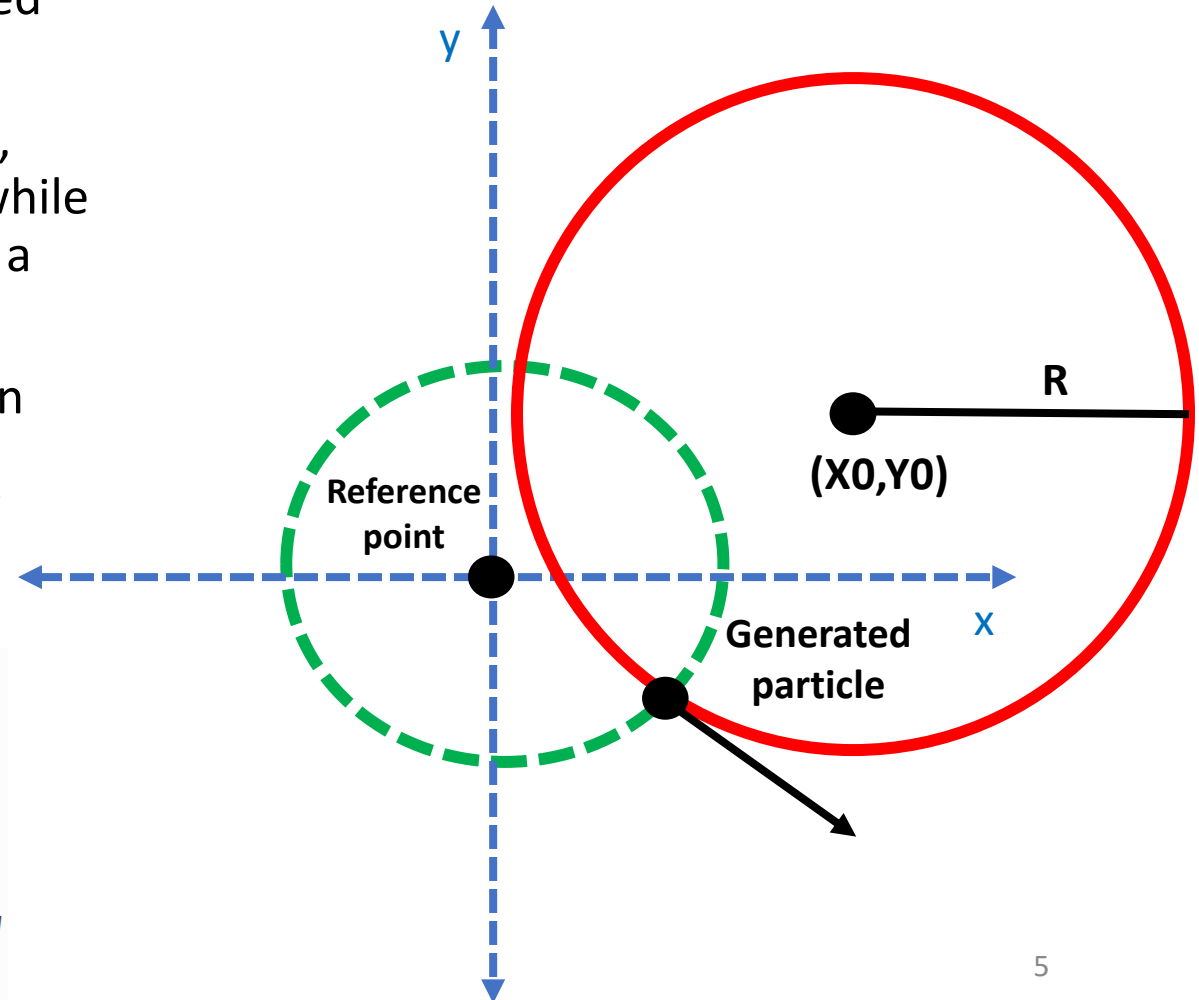

Fig. 4 *Illustration of the perigee parametrization which uses the point of closest approach relative to a reference point. The impact parameter d_0, the position l and the momentum vector $\vec{p}$ are shown.*

y

R

(X0,Y0)

Reference point

Generated particle

x

Current status

➢ From a first look at some DIS events, there do seem to be minor differences between old and new versions for some tracks.

➢ I would suggest we merge in the proposed version into the main branch, as we currently have the changes partially merged. This will also allow us to quickly merge the real-seeded tracking.

➢ Following the update, it would be good if someone would be willing to study and catalog any different.

➢ If we do find some significant differences that can be traced down to the seed input position relative to the reference point, we'll need to decide the best approach to take. For example, we can add the perigee surface to the *edm4eic::TrackParameters* data type, allowing us to effectively go back to the previous setup.