Track EDM Update

Joe Osborn
BNL
November 2, 2023

Overview

- Issue - currently 3 outputs from the fitter, all of which contain (effectively) the same information
- Has downstream implications for all other reconstruction algorithms using tracks (electron finder, PID, vertexing…)
- Have proposed a solution for the fitting output in the S&C meeting several weeks ago, PR already created and reviewed and ready to be merged

CKF Output

- Current output:
    1. edm4eic::TrajectoryCollection - contains all track states
    2. edm4eic::TrackParametersCollection - contains track parameters at the target surface
    3. std::vector<ActsExamples::Trajectories> - An Acts EDM object which packages parameters and states for use by other Acts algorithms
        1. Note: Acts development team moving away from this object
- Proposal from S&C meeting:
    - Single output of edm4eic::TrackCollection
        - This will contain the track parameters at the target surface, all track states and associated measurements
        - Missing a field for track position at target surface, which should be added
        - Additional fields from edm4eic::Trajectory can be added, e.g. nStates, nOutliers etc. (these are inspired from the Acts trajectory object)

```
edm4eic::Track:
  Description: "Track information at the vertex"
  Author: "S. Joosten"
  Members:
    - int32_t           type            // Flag that defines the type of track
    - float             chi2            // Total chi2 (sum) of the track fit
    - int32_t           ndf             // Numbers of degrees of freedom of the track fit
    - edm4hep::Vector3f momentum        // Track 3-momentum at the vertex [GeV]
    - edm4eic::Cov3f    momentumError   // Covariance matrix on the momentum
    - float             time            // Track time at the vertex [ns]
    - float             timeError       // Error on the track vertex time
    - float             charge          // Particle charge
  OneToOneRelations:
    - edm4eic::Trajectory trajectory    // Trajectory of this track
    - edm4eic::Vertex   vertex          // Track vertex of this track
  OneToManyRelations:
    - edm4eic::TrackerHit trackerHits   // Hits that were used for this track
    - edm4eic::Track    tracks          // Tracks (segments) that have been combined to create
```

Advantages/Disadvantages

- Advantages
  - Single output container that is defined within our EDM, so we maintain control and are not affected by external changes (e.g. Acts updates)
  - Contains all track information that will realistically be needed by any downstream algorithm or analysis
- Disadvantages
  - Have to use additional CPU time to swap in between edm4eic and Acts::EDM for other tracking algorithms (e.g. vertexing, track projections, whatever else comes along)
    - Ultimately a small price to pay to insulate ourselves from external changes

Implementation - TrackParameters

```
edm4eic::TrackParameters:
  Description: "ACTS Bound Track parameters"
  Author: "W. Armstrong, S. Joosten, J. Osborn"
  Members:
    - int32_t             type            // Type of track parameters (-1/seed, 0/head, ...)
    - uint64_t            surface         // Surface for bound parameters (geometryID)
    - edm4hep::Vector2f   loc             // 2D location on surface
    - float               theta           // Track polar angle [rad]
    - float               phi             // Track azimuthal angle [rad]
    - float               qOverP          // [e/GeV]
    - float               time            // Track time [ns]
    - int32_t             pdg             // pdg pid for these parameters
    - edm4eic::Cov6f      covariance      // Full covariance in basis [l0,l1,theta,phi,q/p,t]
```

- TrackParameters are a set of track parameters associated to a surface
- Closely model the Acts EDM since this is what we use
- Cov6f is a packed 21 top half triangular covariance

Implementation - Trajectory

```
edm4eic::Trajectory:
  Description: "Raw trajectory from the tracking algorithm. What is called hit here is 2d measurement indeed."
  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          nHoles            // Number of missing hits
    - 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
  OneToManyRelations:
    - edm4eic::TrackParameters trackParameters // Associated track parameters, if any
```

- Trajectory contains tracking expert information
- Global track trajectory state information (e.g. nHoles, nOutliers, etc)
- Relation to track state parameters, for track state info at each measurement surface visited by the Kalman Filter

Implementation - Track

```
edm4eic::Track:
  Description: "Track information at the vertex"
  Author: "S. Joosten, J. Osborn"
  Members:
    - int32_t            type                          // Flag that defines the type of track
    - edm4hep::Vector3f  position                      // Track 3-position at the vertex
    - edm4hep::Vector3f  momentum                      // Track 3-momentum at the vertex [GeV]
    - edm4eic::Cov6f     positionMomentumCovariance    // Covariance matrix in basis [x,y,z,px,py,pz]
    - float              time                          // Track time at the vertex [ns]
    - float              timeError                     // Error on the track vertex time
    - float              charge                        // Particle charge
    - float              chi2                          // Total chi2
    - uint32_t           ndf                           // Number of degrees of freedom
    - int32_t            pdg                           // PDG particle ID hypothesis
  OneToOneRelations:
    - edm4eic::Trajectory                   trajectory      // Trajectory of this track
  OneToManyRelations:
    - edm4eic::Measurement2D measurements      // Measurements that were used for this track
    - edm4eic::Track      tracks            // Tracks (segments) that have been combined to create this track
```

- Track contains parameters at vertex/PerigeeSurface in global coordinates for analysis
- Relation to trajectory to get track state information
- Relation to measurements that compose the track

Summary

- Track EDM changes complete and ready to be merged into edm4eic
- Currently as a part of the Acts update Wouter has the CKF now outputting a container of edm4eic::Track in addition to other containers
- Procedure:
    1. Change the CKF to conform to new EDM
    2. Update downstream algorithms to read from edm4eic::TrackCollection instead of one of the other 3 containers
    3. Remove creation of other 3 containers from CKF fitting