Fitting and Propagation in ACTS

For ePIC Collaboration Meeting

Focus of this talk

1. How ACTS CKF fitting and propagation work
2. CKF fitting directions (e.g. inside-out, outside-in, combo of both)
3. How trajectory propagation handles magnetic field and material crossed
4. How the covariance matrix is determined

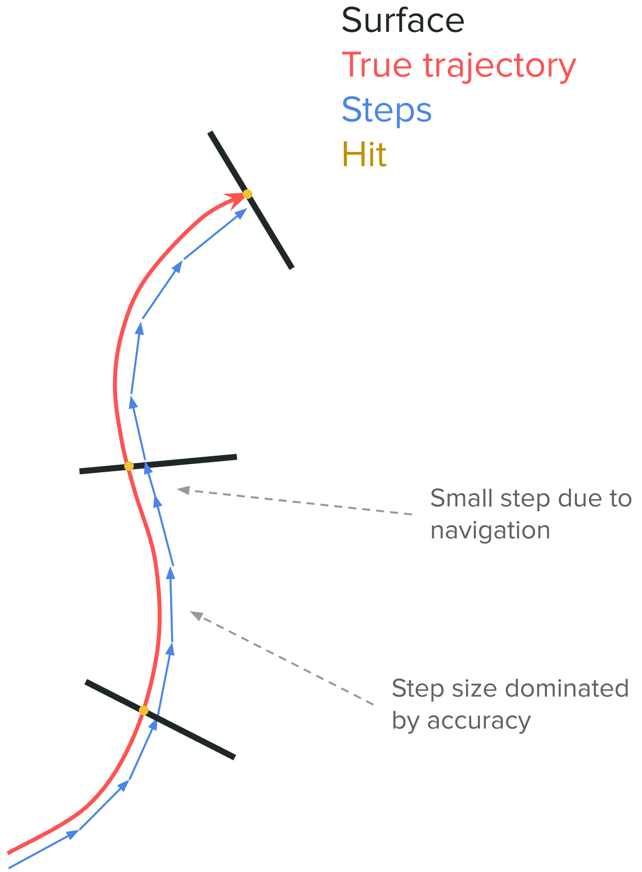
* Focusing on the current version of ACTS (v38.2.0), details might change with future versions

Quick reminder about ACTS

- Experiment independent toolkit for track reconstruction in modern C++
- Not a reconstruction framework but provides an Example one
- Example framework can be used for early studies but has limitations
- Is a community project and we invite everyone to contribute with code and discussions

Propagation

- Extrapolates given track parameters and optionally their covariance
- Is a composition of stepping and navigation
- Stepping extrapolates the parameters and optionally accumulates the transport Jacobians
- On request the stepper will use the full transport Jacobian to compute the current bound covariance
- Navigation determines the next surface to target and therefore effectively dictates the next step size
- Does **not** deal with material effects by itself



Propagation in a magnetic field

- Is handled by the stepping as it influences the numerical integration
- In case of the `EigenStepper` / `SympyStepper` we use RK4 integration
- RK4 requires 3 field queries per step
- ACTS does not deal with magnetic field uncertainties

Propagation Actors

- Actors can observe and steer the propagation
- Basically a function which is called after each step during propagation
- Can access navigation and stepping state and trigger the covariance transport
- This is used for our filters like KF, CKF
- Material effects are handled inside actors after reaching a surface
- For extrapolation you can use the `MaterialInteractor` actor
  - Example in TrackFindingAlgorithm

Covariance transport

- Is part of the propagation and automatically switched on if the input parameters carry an uncertainty
- Under the hood this is really just applying the accumulated transport Jacobian to a given starting covariance
- Material effects are **not** part of the covariance transport

Ad 3: How trajectory propagation handles magnetic field, materials crossed

- The magnetic field is directly consumed and used by the stepping
- Thereby it immediately affects the propagation

- ACTS represents material on surfaces
- If surfaces are encountered during propagation depends on the navigation
- The actor in charge of material interaction observes that we are on a material surface and acts on it

Ad 4: How is the covariance matrix determined

- The covariance matrix at any trajectory state is determined by
  - The initial covariance which is provided by the user
  - The transport Jacobian which is calculated by the stepping
  - Additional terms which are added by actors
- Intermediate covariances have to be requested by actors, otherwise you will only get a covariance at the propagation target
- Fitters have their own material interactor to avoid conflicts with other actors
- For extrapolation a material interactor has to be added to the propagation

Fitting algorithms

- Fitters are implemented as Actors in ACTS
- They observe and change the state of the propagation
- This aligns very well with iterative fitters like the Kalman Filter
- The KF really only relies on the current state of the propagation (predicted) and updates it given a measurement to derive a “better”, updated state (filtered)
- This only feeds information in the direction of the propagation and can be reversed afterwards to obtain optimal estimates for all track state (smoothed)

Ad 1: How ACTS CKF fitting and propagation work

- CKF fitting is using the Kalman formalism to update predicted states using encountered measurements to derive filtered states
- This is done within the propagation using the actor mechanism
- Material effects are handled inside the fitting actor

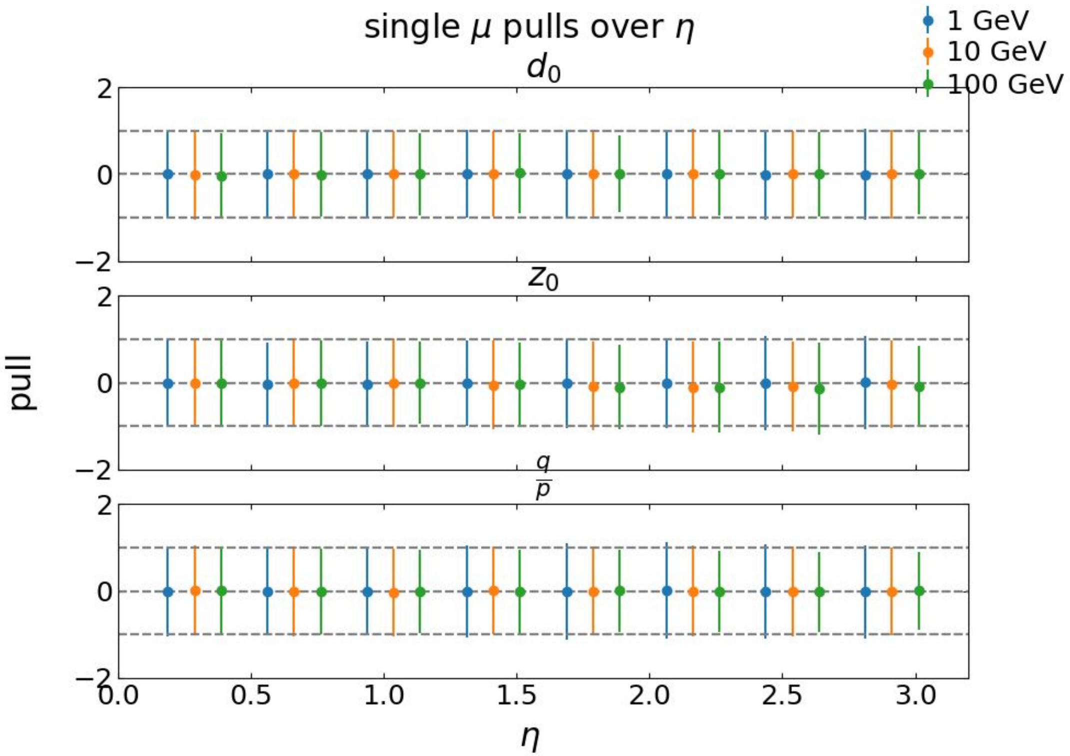
- Propagation is composition of navigation and stepping
- The navigation selects the next surface we should encounter during the propagation
- The stepping advances the track parameters towards the next surface

Acts+ODD Performance

- Validates correct handling of uncertainties in reconstruction

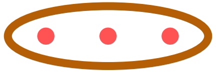
➢ Good control of material effects and cluster uncertainties

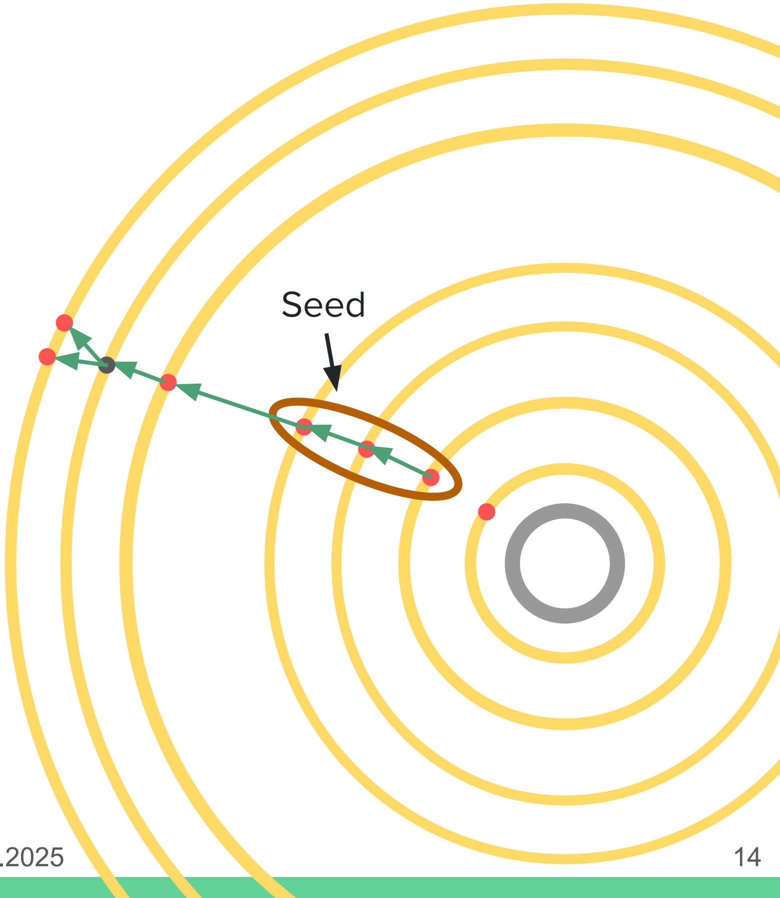
https://indico.cern.ch/event/1252748/contributions/5521502



Track finding direction

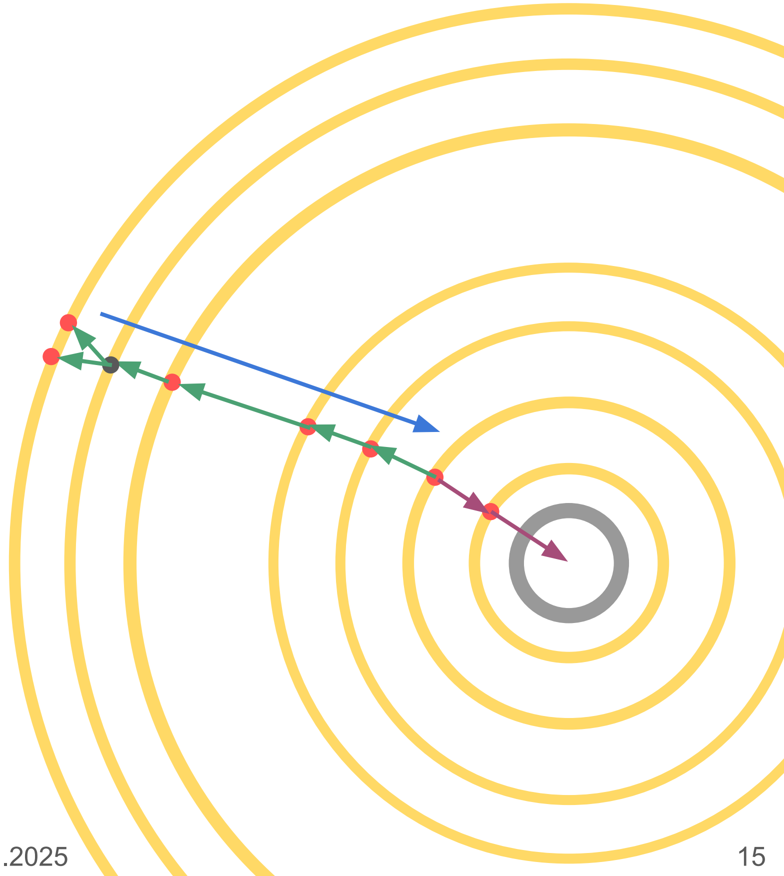
Track finding direction

1. Track finding starts with estimated track parameters at the innermost cluster (in case of pixel seed) 
2. Extrapolate, select measurements, branch, repeat →



Track finding direction

3. Get tracks from this outwards pass, smooth them →
4. Start inwards pass with smoothed params at innermost measurement state →
5. Terminate at perigee surface, extract parameters →
6. Reverse the inner tracklet, stitch them together
7. Output tracks



Track finding direction

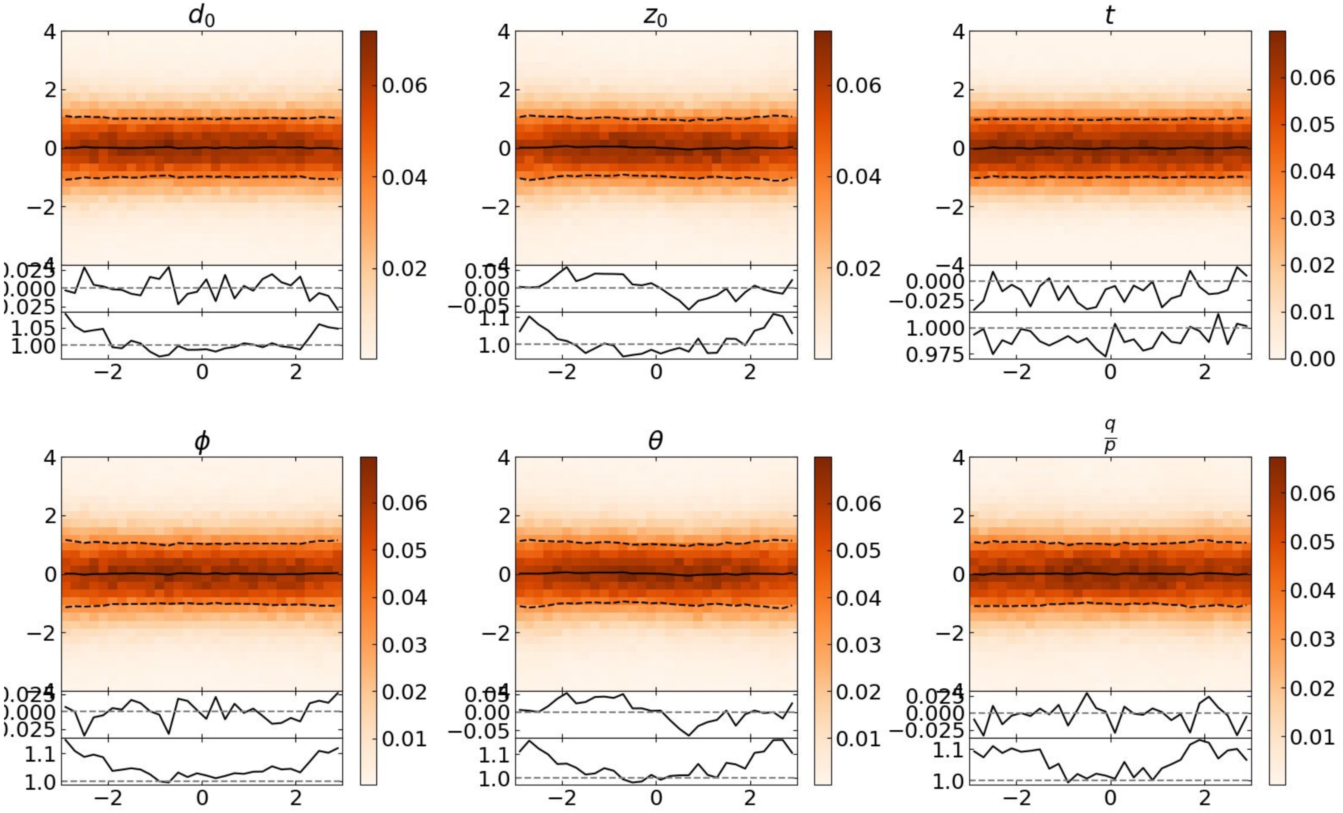
- To obtain all measurements on track we need to look outwards and inwards from the seed if we do not start from the first layer
- This can be achieved with the ACTS CKF by running first in one direction and then in the other direction
- Note that only the innermost track state and the extrapolated track state will have optimal resolution due to combining all measurements
- To obtain optimal resolution on all track states a refit or second smoothing pass is required

Ad 2: CKF fitting directions (e.g. inside-out, outside-in, combo of both)

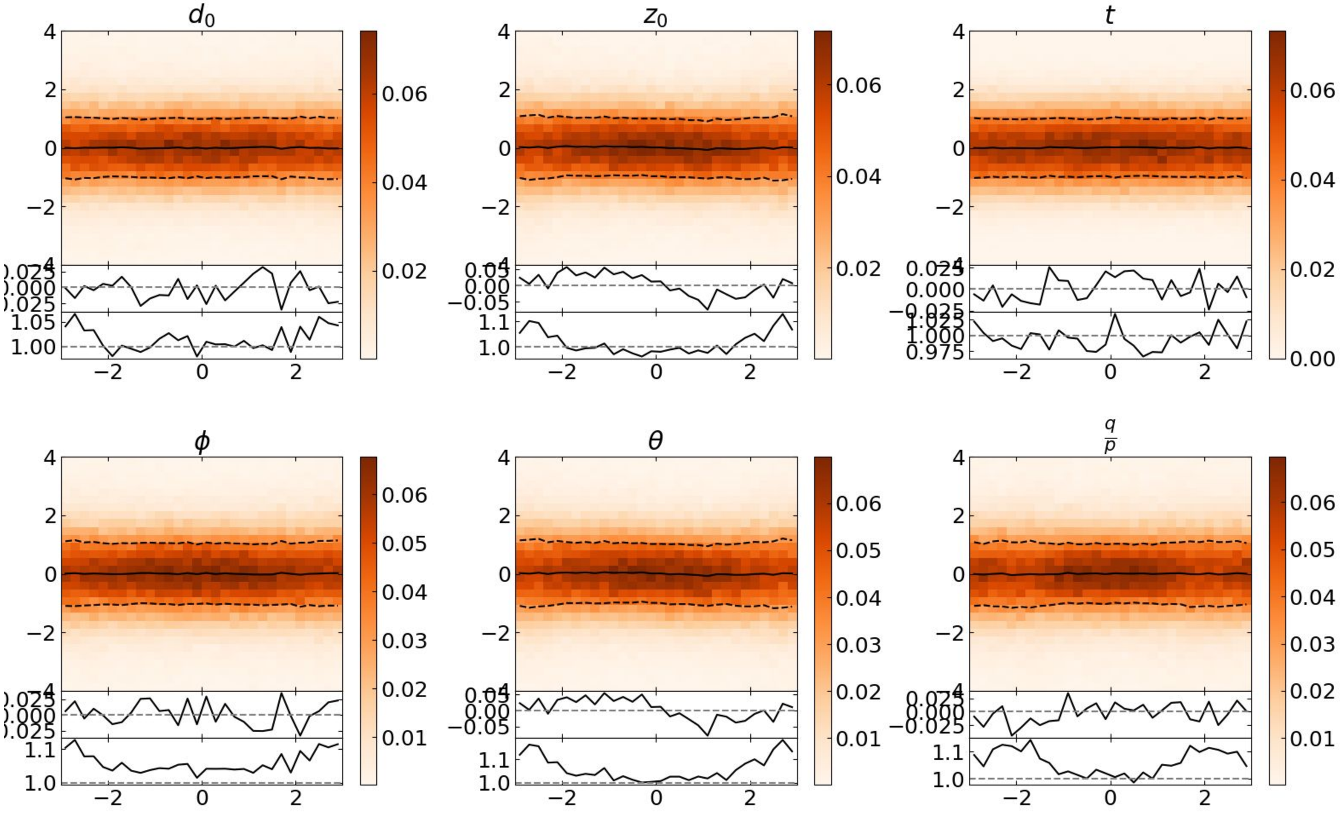
- This is important to find all measurements, primarily if we do not start the finding from the first layer
- To obtain optimal resolution on all track states a refit or second smoothing pass is necessary
- The CKF should always be guided towards the direction of more measurements first as the resolution will be better before reversing the direction
- The direction should only impact the finding performance and not the fitting performance as after smoothing all states should have optimal resolution

Backup

Single muon pulls 1 GeV pT



Single pion pulls 1 GeV pT



Single electron pulls 1 GeV pT