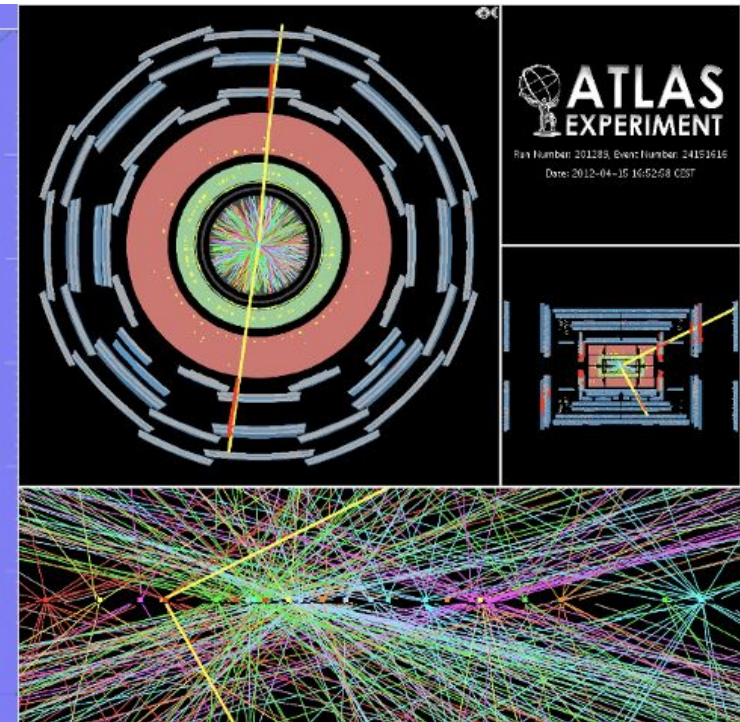
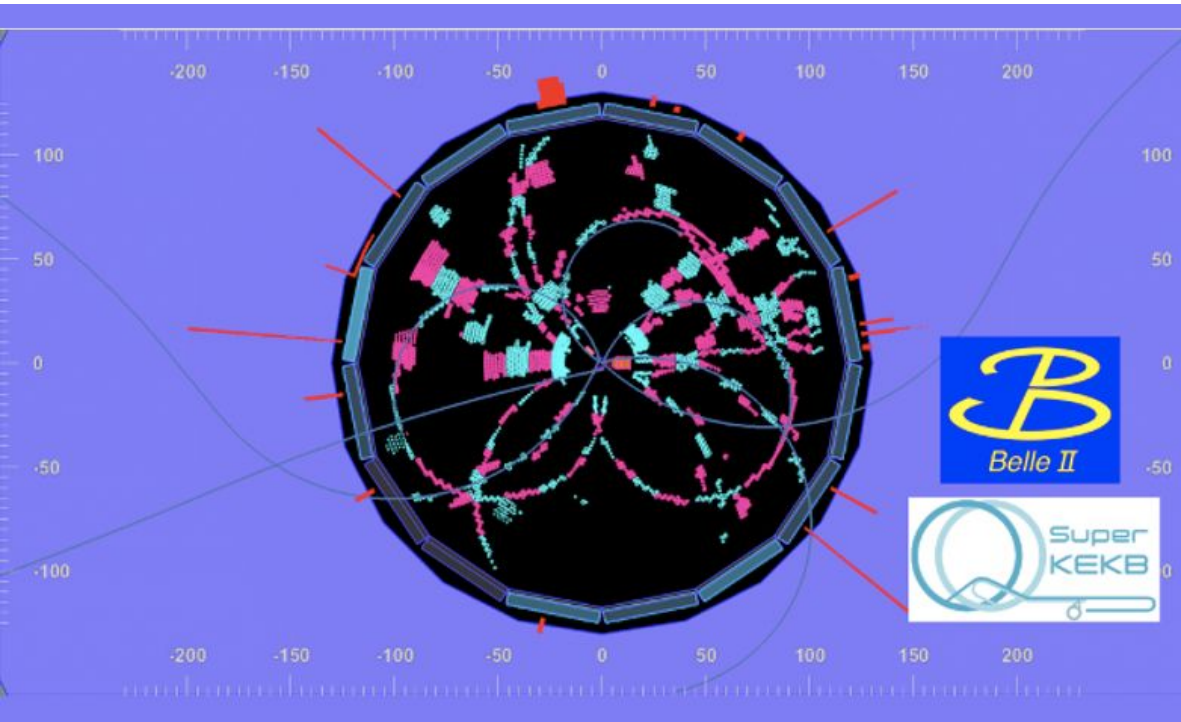


Tracking Workshop for HEP - January 2019



CKF Project

Project Idea

Implement a CKF

Project Goal: Wrap a CKF around Kalman

Project Roadmap:

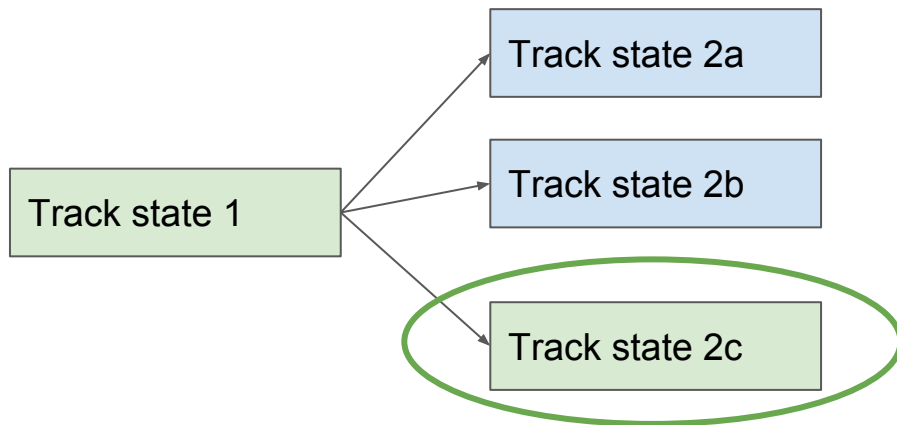
1. understand what is already there
2. brainstorm and develop interface
 - a. think about filter possibilities and how they should be interfaced
 - b. think about seed possibilities and how they should be interfaced
3. brainstorm and develop unittests
4. CKF without filters and final candidate selection
5. Implement first possible filters and final candidate selection
6. (Physics) Performance studies
7. (Computation) Performance studies

Project Participants: Nils Braun, Aaron Soffa, Nick Styles, Sean Conlon, you?

- In the end, decided to start with a Sequential Kalman Filter, as a more manageable task
 - No branching, just take best measurement on the surface

Implementation Details

- First challenge - how to store input measurements
 - Started with `std::multimap` with `geomID` as key to store measurements per layer
 - Eventually went with `map` of `vectors` for simplicity
 - May want to change this to `std::set`



Compute χ^2 for each new hit added
→ Update with lowest χ^2

Achieved Goals and Work to be done

- Sequential Kalman Filter code implemented
 - Needs some cleanup and refactoring (mostly the test), but interface is there
 - Only using χ^2 sorting in the moment (could be configurable)
- First unit test based on the Kalman test successfully
 - More tests to come (e.g. including multiple tracks, different geometry)
- Combinatorial part is a bit harder to implement
 - Plans exist:
 1. Perform loop over track propagations using current sequential implementation
 2. Use new KF actor with trajectory branching
- Configurable options for determining which partial track candidates to keep
 - Keep list of lowest χ^2 candidates
 - Change selection algorithm

Where to find it

Currently uploaded to

<https://gitlab.cern.ch/nibraun/acts-core/commits/feature/sequential-kalman-filter>

Time schedule/Requirements achieved for a pull request?