Applications of *A-Common-Tracking-Software* in the Future Circular Collider Design Study

B. Hegner [1], J. Hrdinka [2], A. Salzburger [1], V. Volkl [3]
1: CERN, Geneva, Switzerland, 2: TU Wien, Austria, 3: Universität Innsbruck, Austria

The FCC Design Study

The Future Circular Collider (FCC) design study plans a high-energy frontier particle accelerator of 100 km circumference to succeed the Large Hadron Collider currently in operation. The physics program of the hadron-hadron accelerator, reaching a collision energy of up to **100 TeV** is outlined in [1]. A conceptual design report is planned for end of 2018.

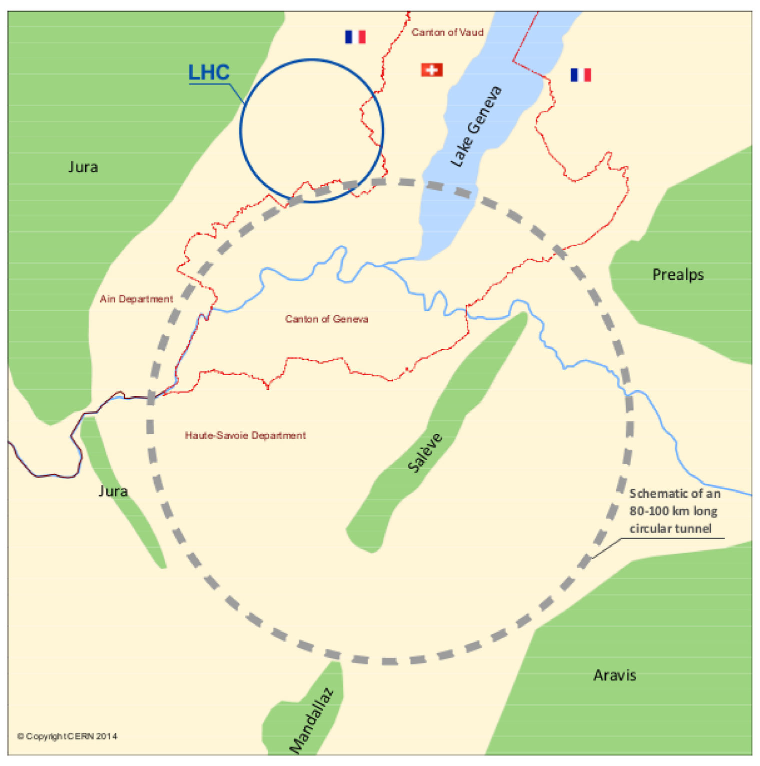


Figure 1. Schematic of the proposed FCC accelerator infrastructure

FCC Software Framework

Ongoing work on hadronic (hh) and leptonic (ee) options for the beams of the accelerator are supported by a common software framework (**FCCSW**), based on Gaudi, developed at CERN. Great emphasis is put on collaboration and synergies, as for example with the adoption of the *A Common Tracking Software* (ACTS) [3] project that aims to extract and package the tracking code of the ATLAS experiment as a toolkit for tracking in FCCSW. For seeding, a cellular automaton based code is developed in the standalone library *TrickTrack* (see below) [4].

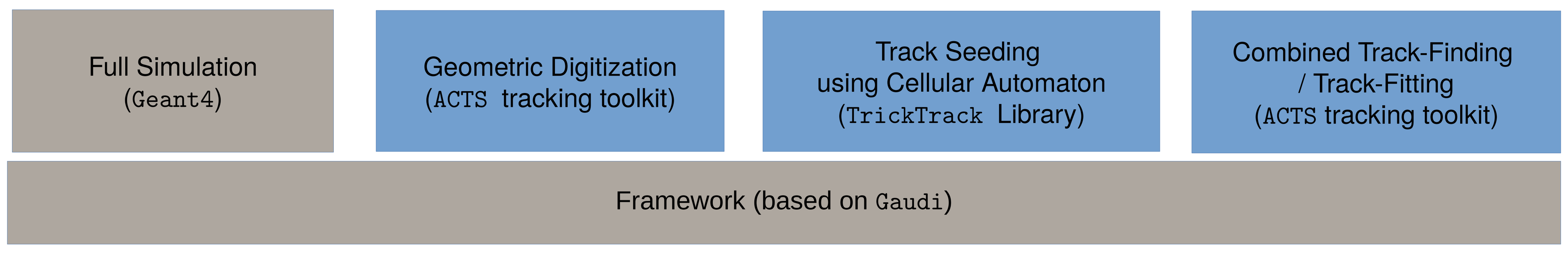


Figure 2. Proposed components of the FCC-hh track reconstruction in FCCSW

Pileup Mitigation and 4D-Tracking

The luminosity goals for FCC-hh imply an average **pileup rate** of ~1000 proton-proton collisions per event, assuming 25 ns between bunch crossing, or ~200 proton-proton collisions per event assuming a bunch crossing every 5 ns [2].
Full simulation and reconstruction of events of this size is prohibitively expensive.

The reconstruction has to exploit new detector technologies to suppress the **combinatorics** of track candidates introduced by the pileup.

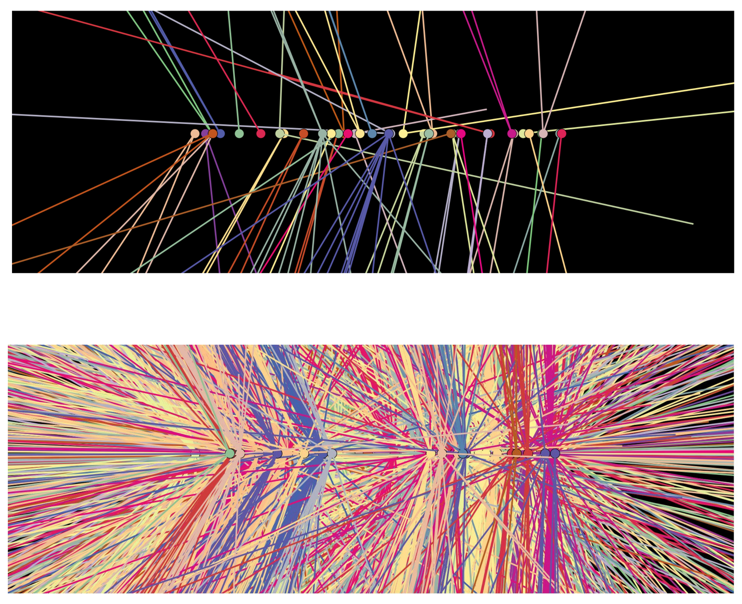


Figure 3. Visualization of typical event track densities in LHC conditions (25 pileup vertices, top) vs. those expected at FCC (1000 pileup vertices, bottom)

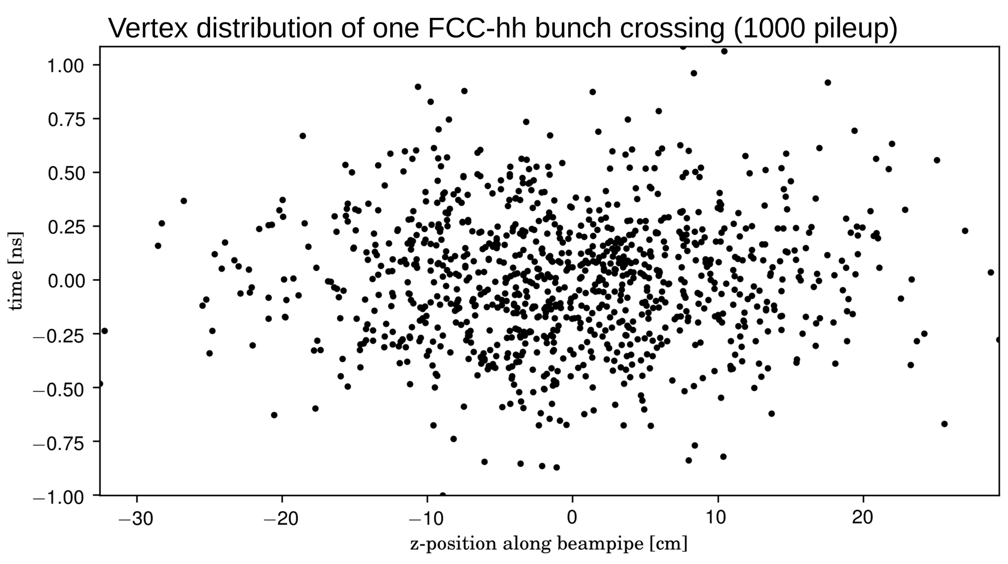
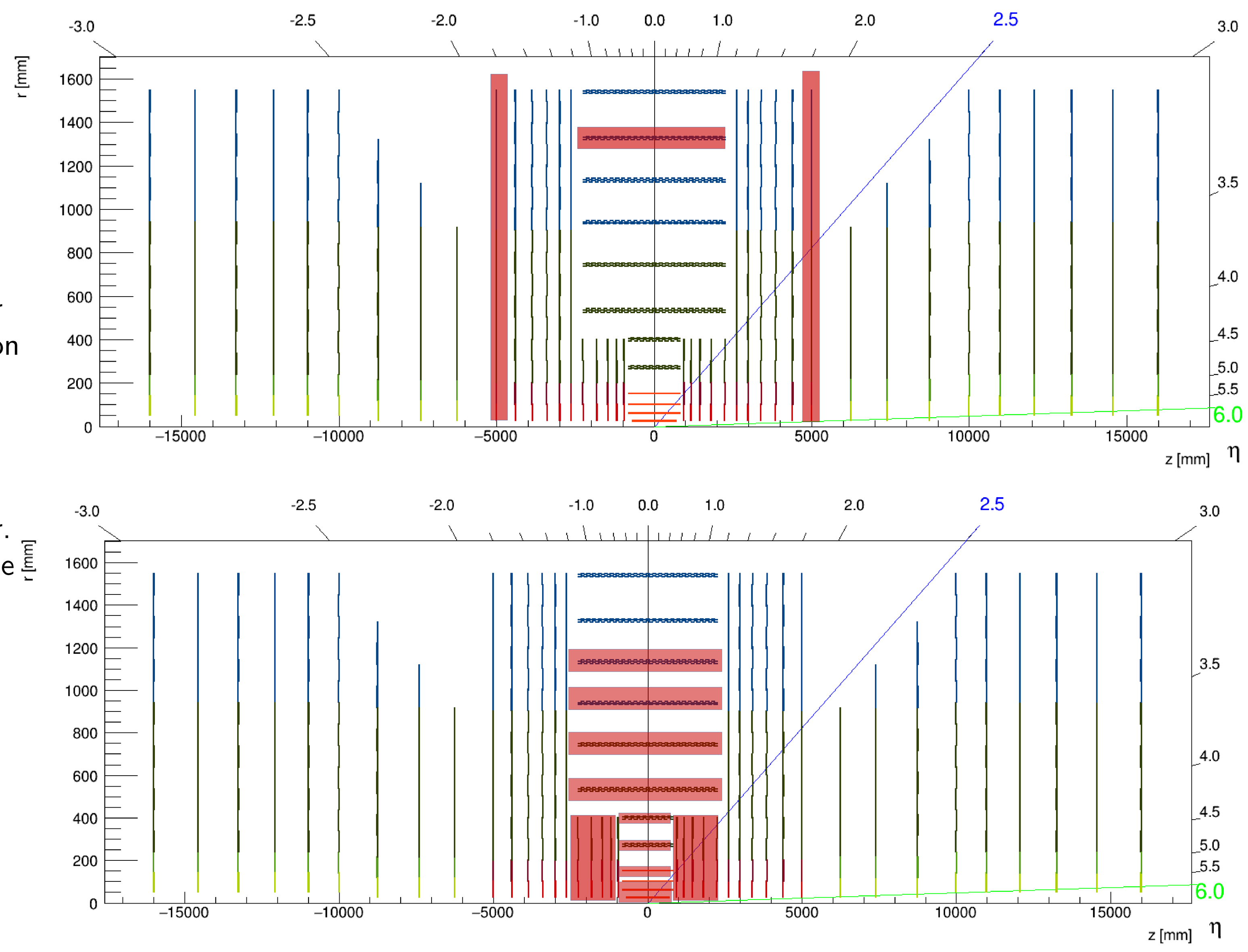


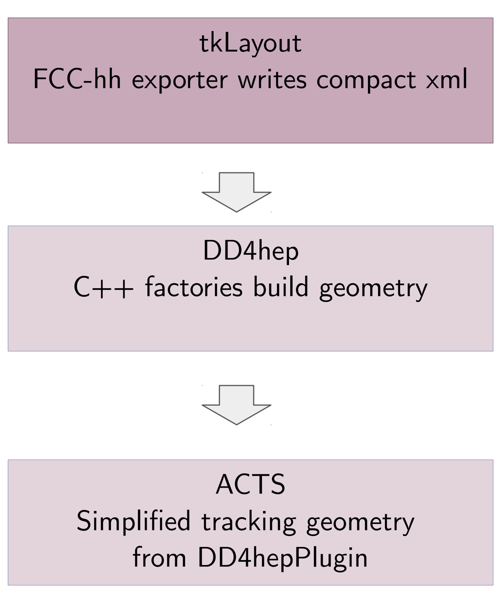
Figure 4. Simulated distribution of vertices in one typical bunch crossing at FCC-hh.

Figure 5. Possible timing layer configurations of the FCC-hh tracker. Above: one timing layer. Below: Time detectors throughout the inner detector to aid in seeding



Time information as a fourth dimension in addition to the three spatial coordinates usually measured can be used to help distinguish pileup and signal vertices. This is possible even in a light configuration with time detectors in just one layer, by propagating the time measurement of each track back to the luminous region. To reduce the combinatorics in the seeding step, timing capabilities are required for all seeding layers. The event data model of ACTS is flexible with regards to the number of track parameter, and the propagation package is currently being adapted for measurements of timing detectors.

DD4hep Plugin and Geometry Workflow



The detector description toolkit DD4hep is used in FCCSW as a central source of detector information. For faster turnaround in detector design, tkLayout was adapted to export central parameters directly to a DD4hep-readable compact file which can be used in simulation, and via the DD4hep-plugin of ACTS, in reconstruction.

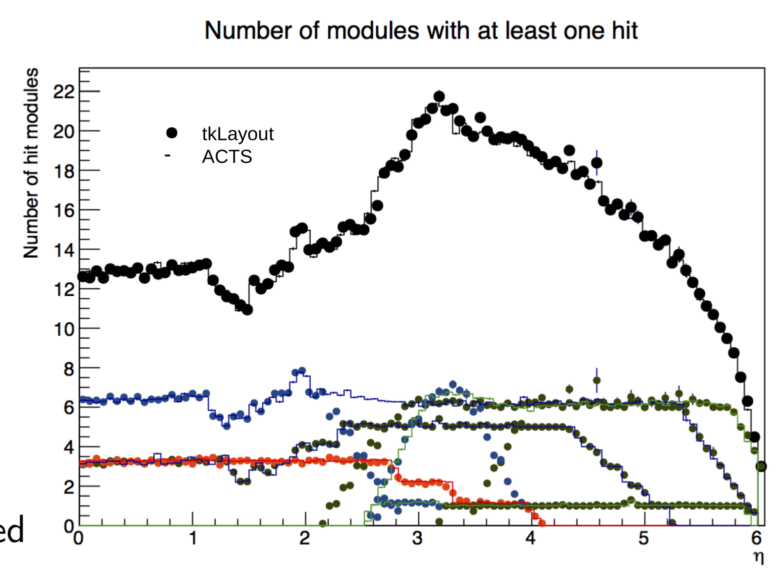


Figure 6. Comparison of the number of hits along a straight ideal track in tkLayout and derived geometries

Seeding: *TrickTrack*

Complementing the track fitting and propagation functionalities, FCC software uses the *TrickTrack* library to identify track candidates. In a cellular-automaton-based approach originally developed for CMS, *TrickTrack* connects all geometrically compatible doublets of hits and treats them as cells of a cellular automaton to find N-tuples of hits. Due to the use of a **KD-Tree** as data structure for the tracker hits, the geometric cuts can easily be extended to other dimensions as for example time, to reduce the combinatorics of track seeds in this stage of track reconstruction.

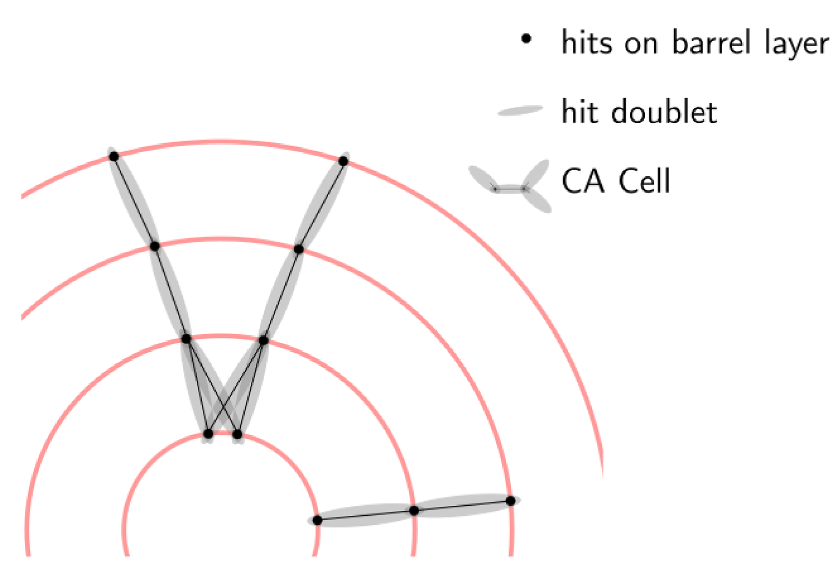


Figure 7. Schematic of the cellular-automaton based seeding procedure used in the *TrickTrack* library. See text for details.

References

[1] Mangano, M. *Physics at the FCC-hh, a 100 TeV pp collider* (2017). doi:10.23731/CYRM-2017-003
[2] Benedikt, M., Schulte, D. & Zimmermann, F. *Optimizing integrated luminosity of future hadron colliders.* Phys. Rev. ST Accel. Beams 18, 101002 (2015).
[3] Gumpert, C., Hrdinka, J., Salzburger, A., Calace, N. & Kiehn, M. *ACTS: from ATLAS software towards a common track reconstruction software*. ATL-SOFT-PROC-2017-030
[4] TrickTrack source code repository. `https://github.com/HEP-SF/TrickTrack`