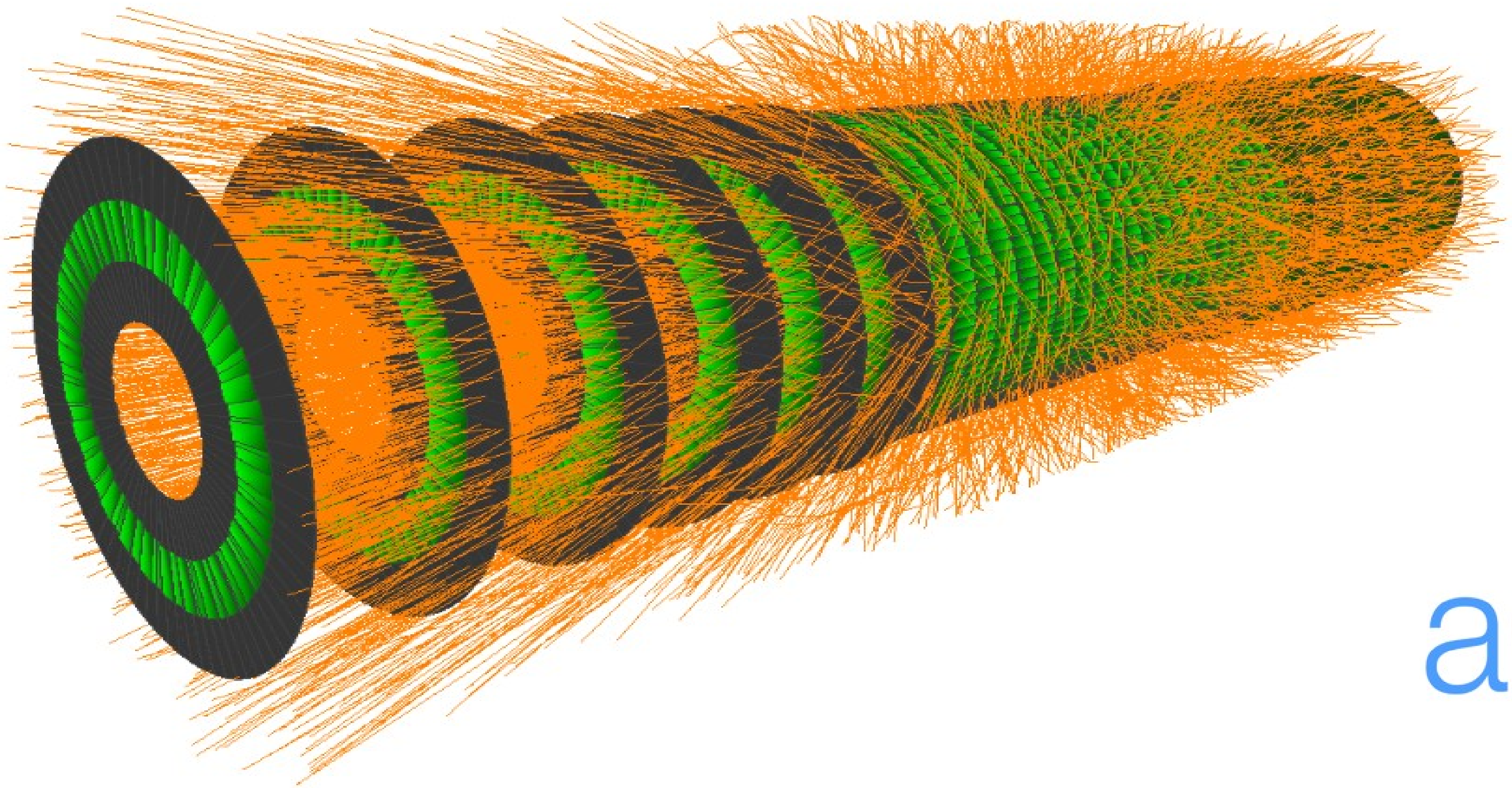


Acts: A common tracking software

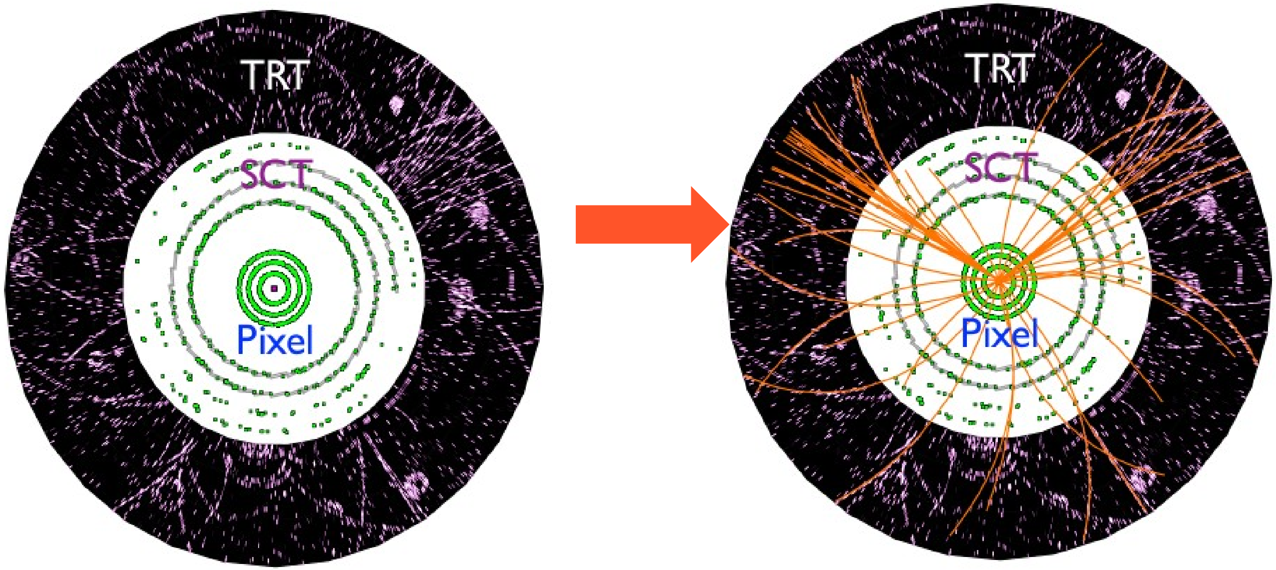
Xiaocong Ai for the ACTS developers
UC Berkeley

DPF 2019, Boston, Jul 31, 2019

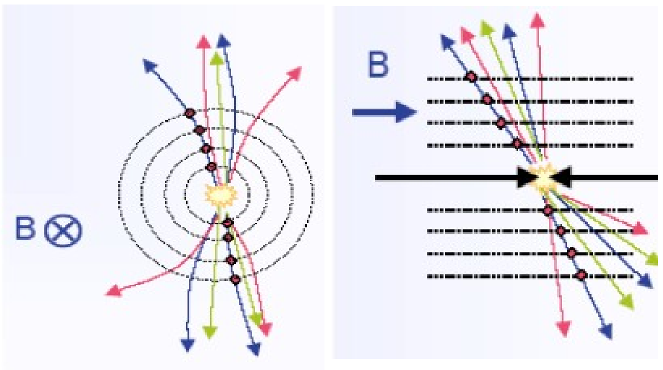
Tracking concept

- Particles traverse the detector resulting in electronic signals due to interaction with material
- Tracking system is used to measure position and momentum (by curvature in magnetic field) of charged particle by recording signals along the trajectory

e.g. **ATLAS tracking system (Inner Detector)**



Helical track in solenoidal B field

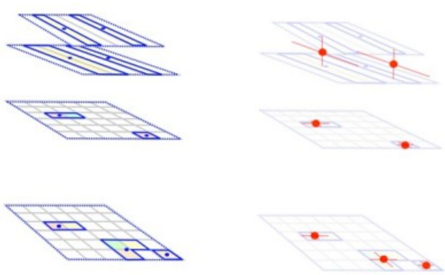


- **Track reconstruction (of charged particle)**
  - Identify the group of hits that correspond to the charged particle trajectory
  - Estimate trajectory parameter (position, momentum)
- Reconstructed tracks/vertexes are ingredients for physics object reconstruction

Tracking recap

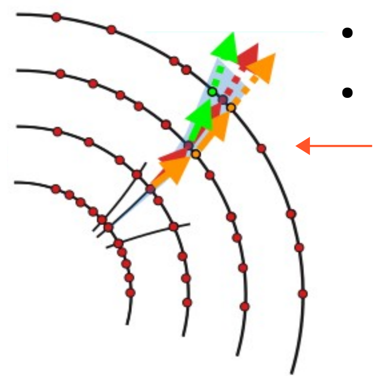
Input data preparation

- Usually includes:
  - Raw data converted to cluster/drift circle
    - need calibration constants
  - 3D SpacePoint formation
    - need alignment constants



Track finding (pattern recognition)

- Identify the group of his corresponding to a track

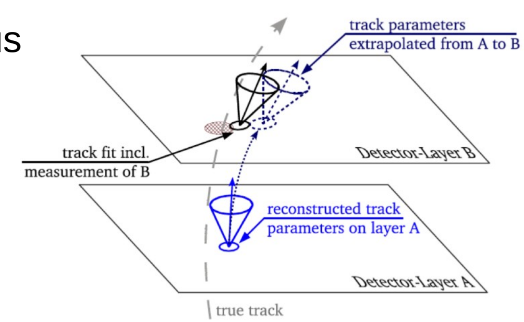


- Global (all hits considered simultaneously)
- Local (progressively adding hits)
  - e.g. track seeding + track following
    - Track seeding: find seeds from 2/ 3 hits and provide initial track parameter
    - Track following: propagate track parameter to search for additional hits

Track fitting

- Estimate trajectory's parameter

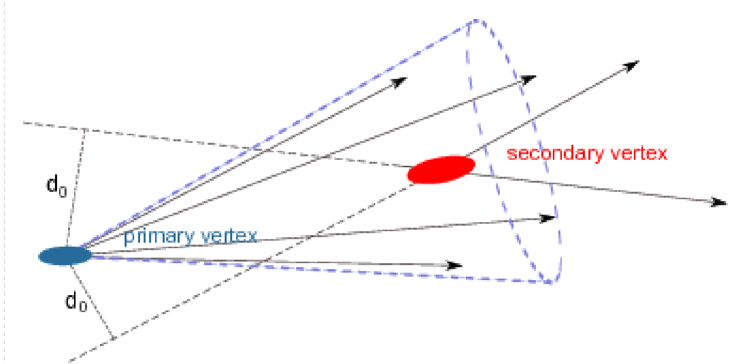
- Least squares estimation (simultaneous consideration of all measurements)
- Kalman Filter (progressively from one measurement to the next)
  - Could be merged with track following
  - Need non-Gaussian extension
    - e.g. Gaussian-sum filters (GSF)to handle non-Gaussian energy loss of electron Bremsstrahlung



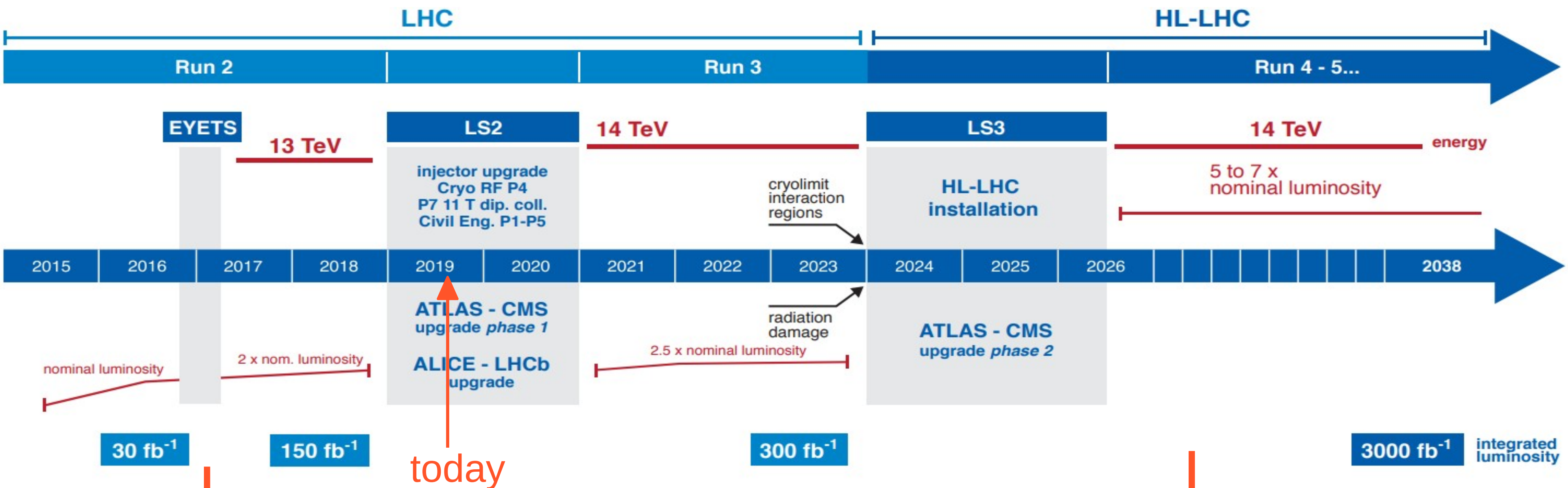
S. Fleischmann's thesis

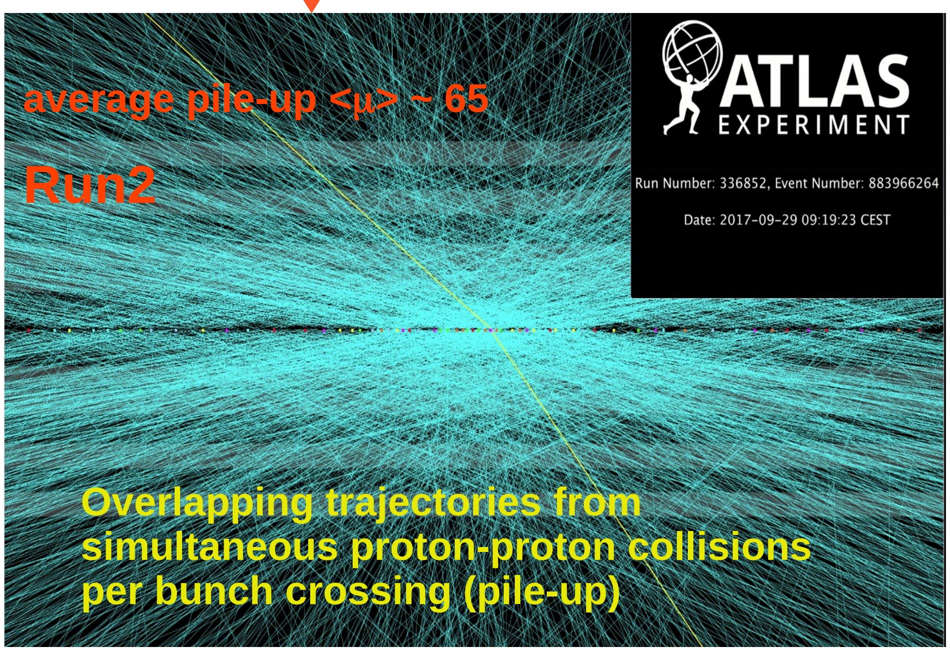
Vertex finding

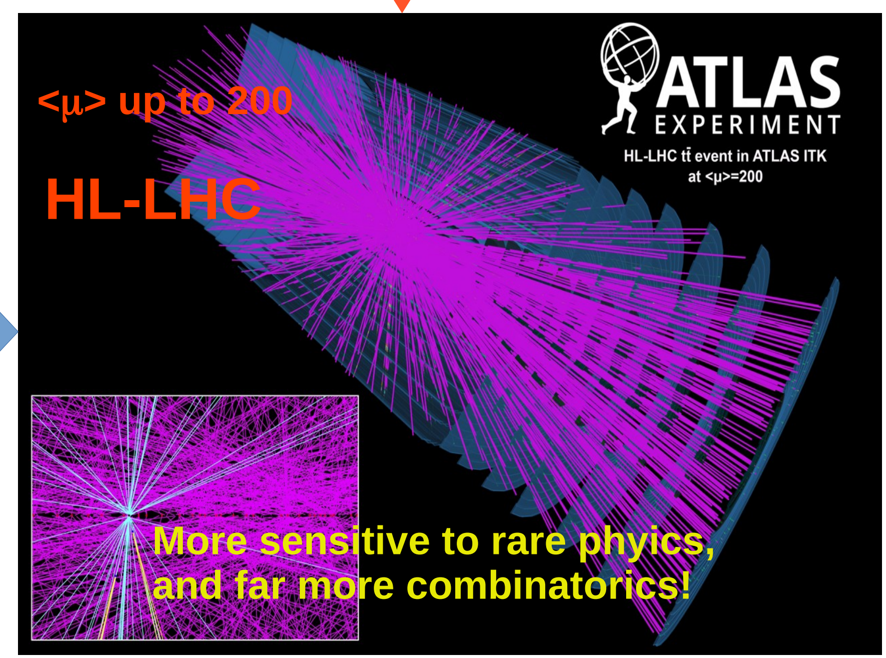
- Associate sets of tracks to the same interaction point
- Estimate vertex properties



High-Luminosity LHC (HL-LHC)

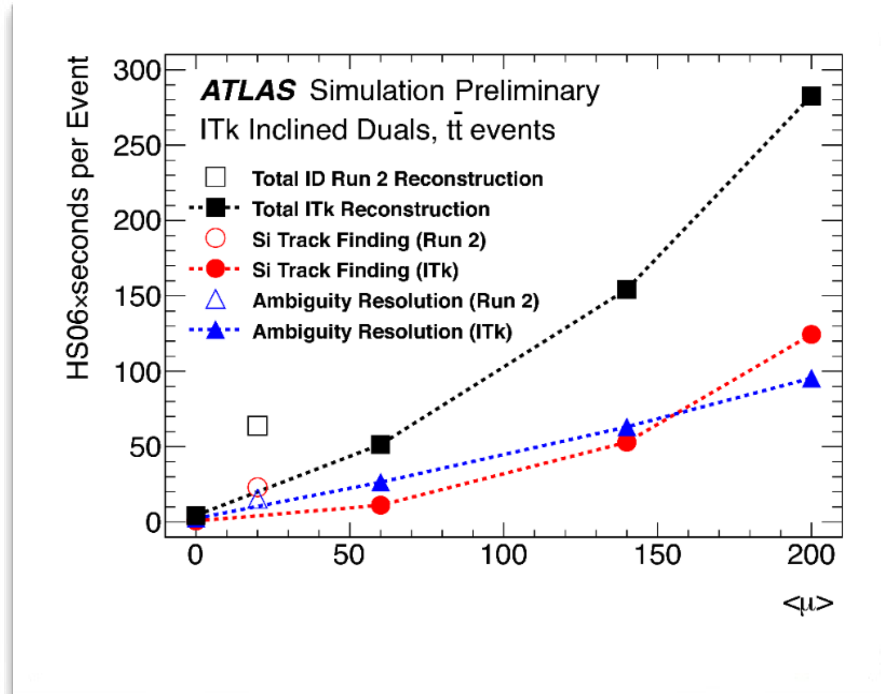




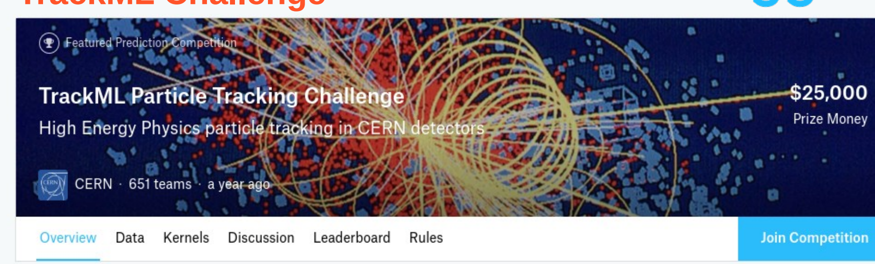


Tracking challenge

- Tracking presents the greatest challenge to CPU at future hadron collider, e.g. HL-LHC, FCC-hh
  - Explosion in hit combinatorics from ~1000 overlaid pileup tracks
- Fast and accurate tracking software is necessary
  - Write **thread-safe** code which can be executed concurrently to maximize memory access
    - e.g. involved on-going migration of ATLAS's software framework (Athena) to a multi-threading framework (AthenaMT)
  - Use performant software technology
    - e.g. Eigen algebra library to optimize data structure
  - Improve tracking strategy
    - e.g. Machine Learning techniques
  - Adapted to modern hardware architectures
    - e.g. **multi-core**, GPU



TrackML Challenge



Solutions at arXiv:1904.06778

Acts: a common tracking software

- A tracking toolkit for future detectors and modern computing infrastructure based on LHC tracking experience
- An open-source platform for tracking algorithms R&D
  - Collaboration across experiments: ATLAS, LHCb, FCC-hh, FASER, CEPC and external contributors, e.g. machine learning experts

http://acts.web.cern.ch/ACTS/

ATLAS tracking software (Athena)

- Performance tested and known
- Hard to maintain after > 15 years
- Design not thread-safe

→

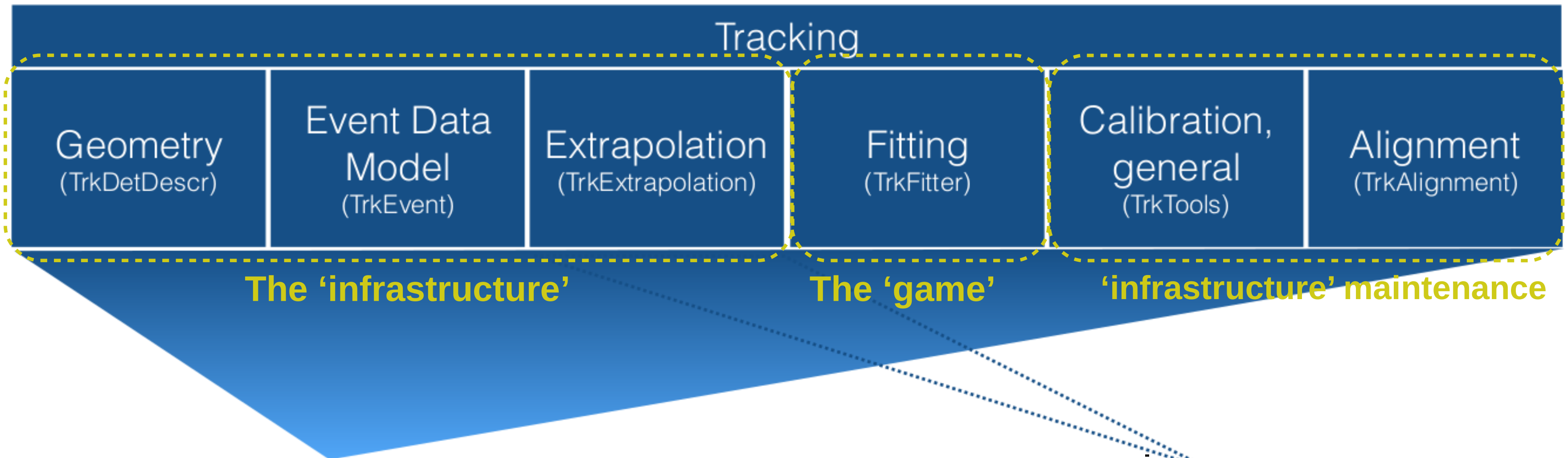
Acts

- Preserve and improve ATLAS tracking code
  - Modern C++ 17 concepts
  - Facilitate usability and maintenance
  - Thread-safe design
  - Minimal dependencies to simplify integration with software framework

ACTS components

Most figures from [A. Salzburger's slides]

ATLAS Tracking modules



acts-core

Surfaces | Geometry | Material | EventData

Propagator | Fitter | Seeding | Vertexing

Magnetic field | Utilities

acts-fatras

Simulate particle trajectories through detector with simplified material effects

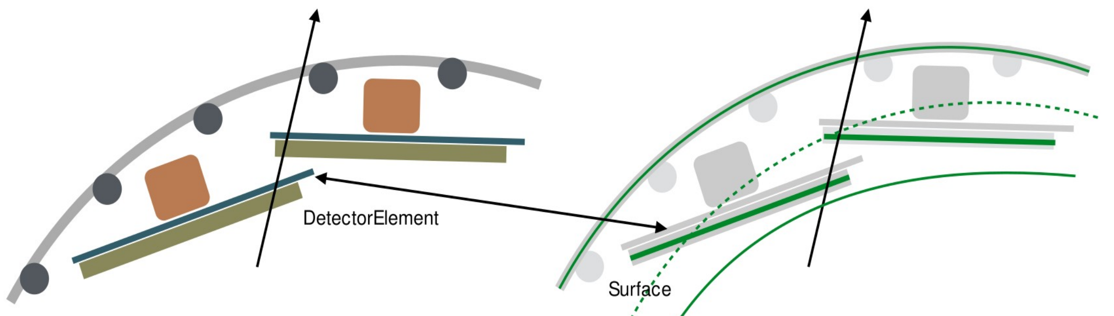
acts-framework

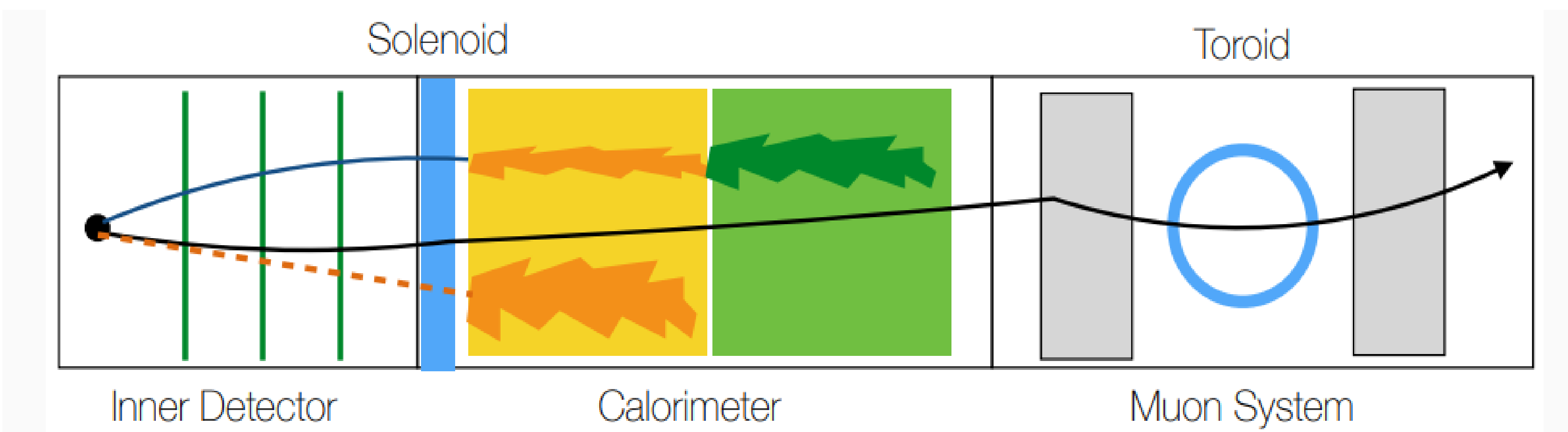
A light-weight Gaudi style framework for integration and concurrency test

Tracking geometry

How to describe detector geometry for tracking

- A simplified geometry from full detector geometry
  - Basic concepts: surfaces
- Geometry plugins provide converters from different geometry description formats (DD4hep, TGeo)
- Detector geometry implementation
  - Supporting silicon tracker & calorimeter & Muon System





Volumes with layers | Volumes with dense material | Volumes with floating objects

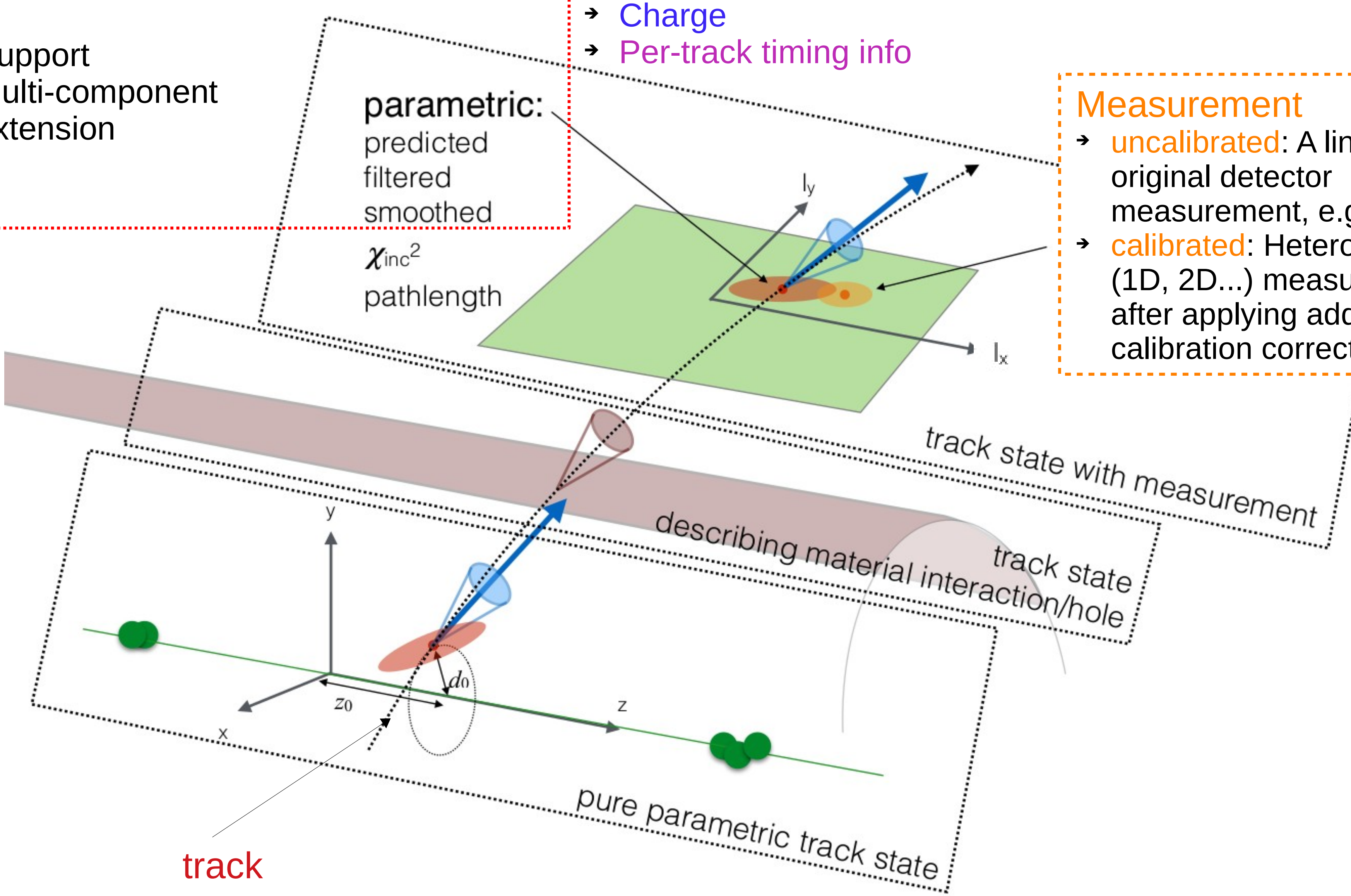
Event Data Model (EDM)

How to describe track parameter, measurement, track state and track...

Track parameter: (l_1, l_2, ϕ, θ, q/p, t)

Support multi-component extension

- Local position on surface
- Momentum
- Charge
- Per-track timing info



Measurement

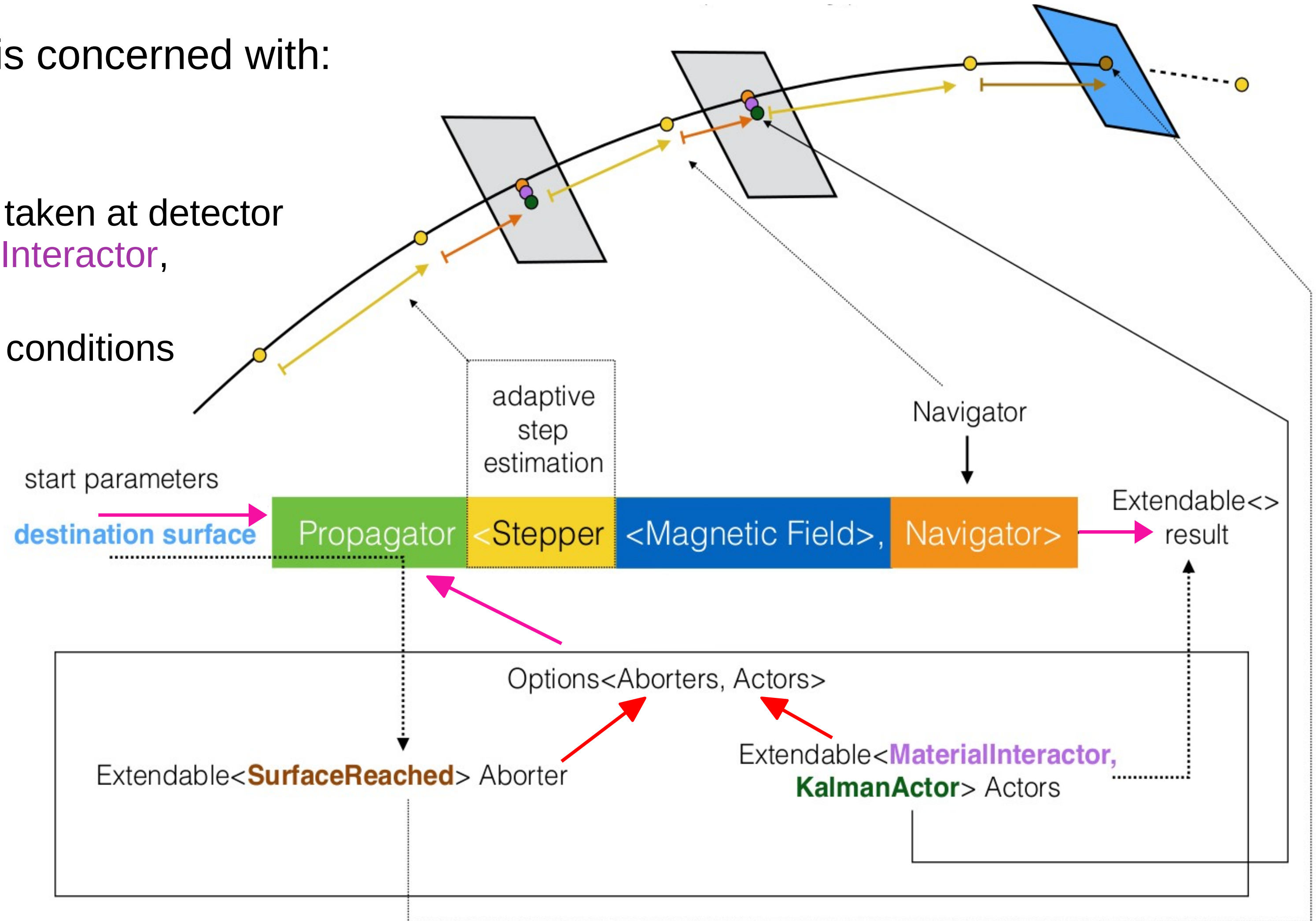
- uncalibrated: A link to original detector measurement, e.g. cluster measurement
- calibrated: Heterogenous (1D, 2D...) measurement after applying additional calibration corrections

Propagator

How to propagate track parameter from one detector part to next

- Different propagation scenario determined by the Stepper and Navigator:
  - Stepper: integration of equation of motion for the charged particle (in a magnetic field) and provide step estimation
  - Navigator: determine navigation step size to reach detector according to tracking geometry

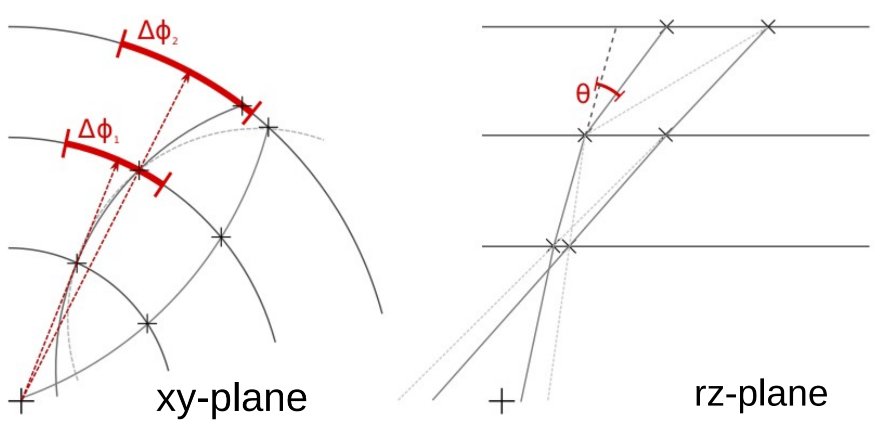
- A specific propagation is concerned with:
  - Start parameters
  - Option list:
    - Actors: list of actions taken at detector surface, e.g. MaterialInteractor, KalmanActor
    - Aborters: list of abort conditions
  - Result:
    - propagation status
    - track parameter
    - action outputs



Pattern Recognition

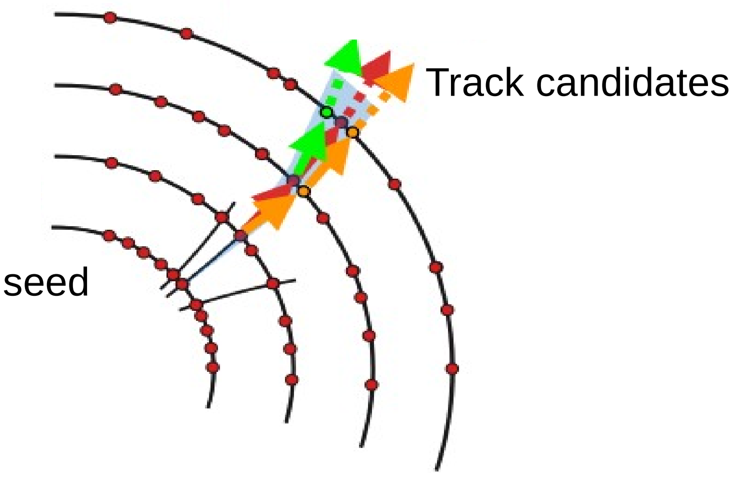
- Strategy: transcribe ATLAS pattern recognition code into ACTS code design
  - A simple **combinatorial seed finder** is available in ACTS. Tested and validated against AthenaMT
  - **Combinatorial Kalman Filter** for track following to be imported

The combinatorial seed finder in ACTS



- Starting from hits on first layer, search hits on the second and third layer within an azimuth angle window
- Associate two hit doublets based on polar angles

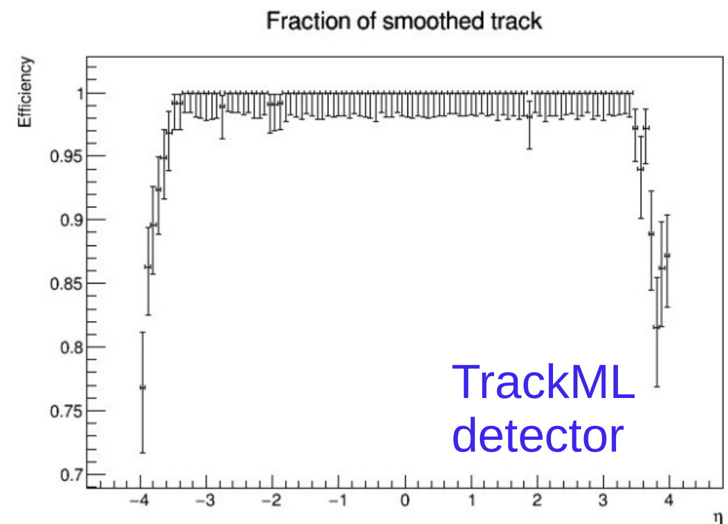
The combinatorial track finder



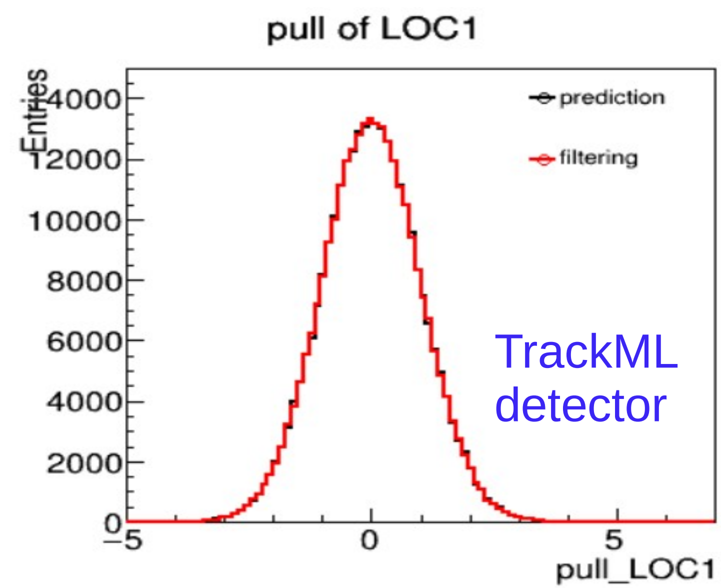
- Starting from seeds, incorporate additional space-points from the remaining layers
- Multiple track candidates per seed
- Ambiguity solving is usually needed to remove track candidates with bad quality

Track fitting and Vertexing

- Kalman Filter prototype in validation/improvement
  - 100% efficiency for truth particle trajectories (sets of truth hits from simulation)
  - Parameter pull is (approximately) Gaussian distributed
  - Hole (missing measurement on sensitive detector region) search and outlier (measurements not compatible with track hypothesis) logic in progress



- GSF (approximates non-Gaussian energy loss of electron Bremstrahlung as Gaussian mixture) prototype in progress
  - Multi-component stepper and material interactor complete
  - Multi-component Kalman filtering and smoothing in progress



- Vertex finder prototype (Iterative vertex finder and multi-adaptive vertex fitter) in optimization

Alignment/Calibration

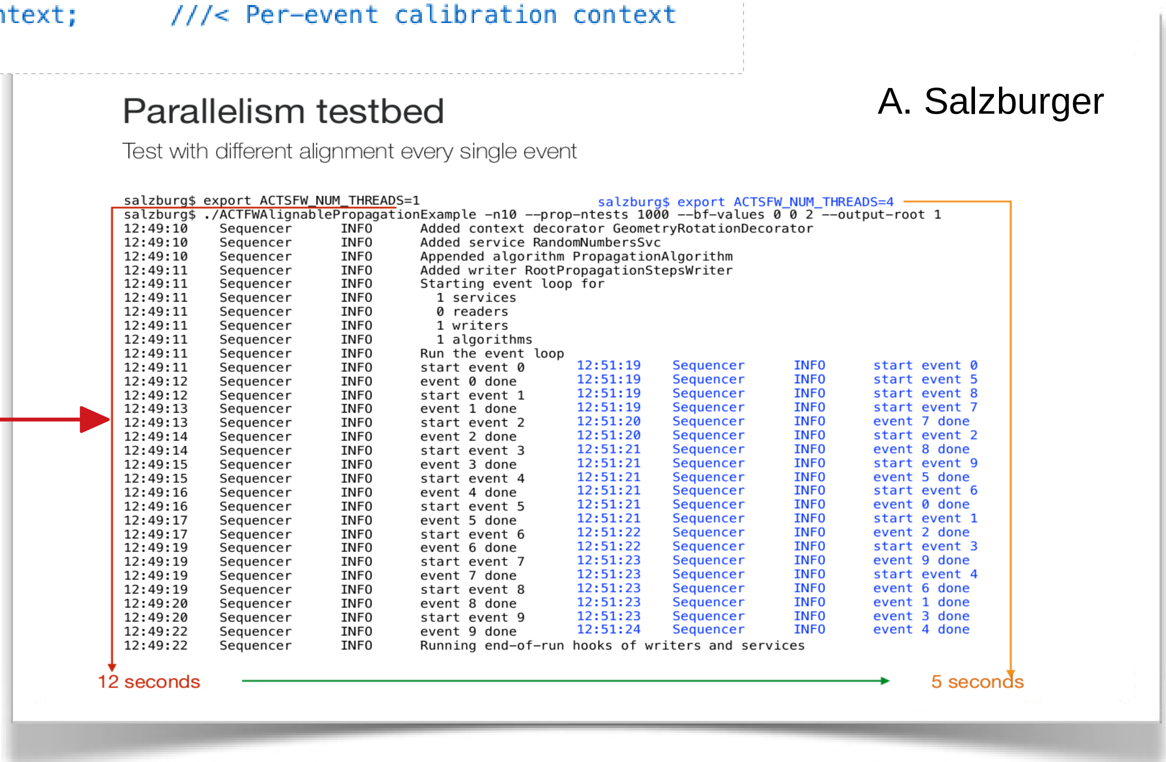
Capabilities to handle detector geometry and condition change

- Multi-threading event processing needs to allow track reconstruction with **multiple** alignment constant, calibration constant or even magnetic field **simultaneously**
- ACTS uses contextual conditions concept to support on-the-fly event-dependent changes of alignment/calibration/magnetic field

```
/// Aggregated information to run one algorithm over one event.
struct AlgorithmContext
{
  size_t                       algorithmNumber;    ///< Unique algorithm identifier
  size_t                       eventNumber;       ///< Unique event identifier
  WhiteBoard&                  eventStore;        ///< Per-event data store
  Acts::GeometryContext        geoContext;        ///< Per-event geometry context
  Acts::MagneticFieldContext   magFieldContext;   ///< Per-event magnetic Field context
  Acts::CalibrationContext     calibContext;      ///< Per-event calibration context
};
```

- Contextual calibration tool will be implemented

Propagation test with contextual alignment constant in both single-thread and multi-thread event processing



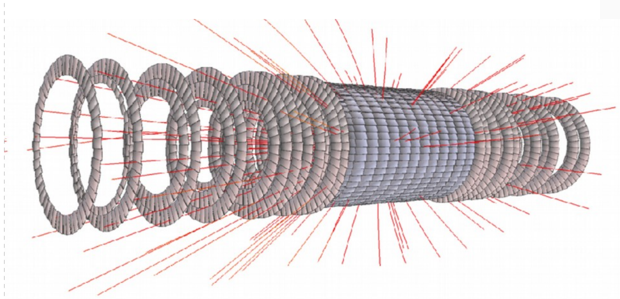
Integration with AthenaMT/FCCSW

Plugins (e.g. geometry, magnetic field, EDM) are needed to interface ACTS to experiment software

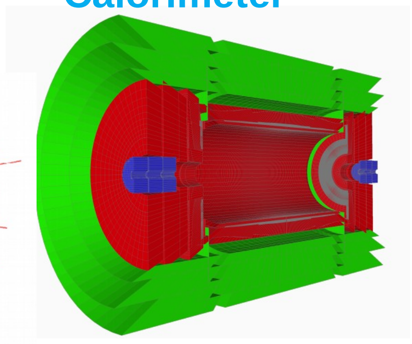
Integration to ATLAS AthenaMT

- Geometry implemented for Inner Detector and Calorimeter (prototype)
- MT-ready or MT-prepared seeding, alignment, extrapolation and fitter

ATLAS current Pixel+SCT



ATLAS current Calorimeter

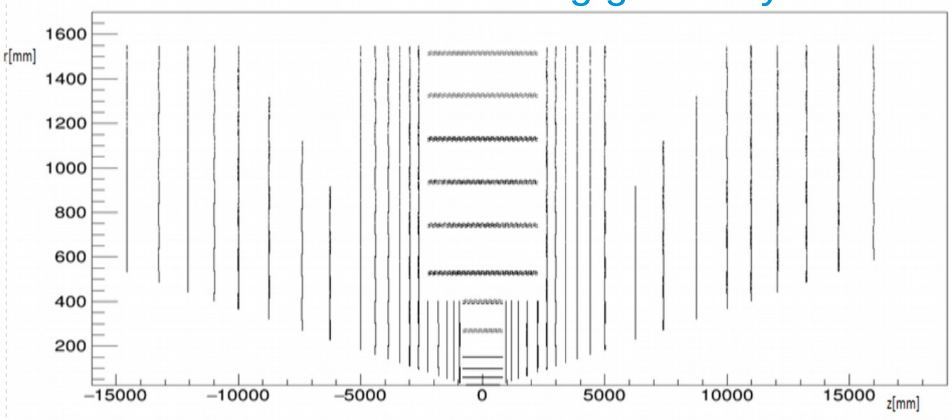


P. Gessinger

Integration to FCC FCCSW

- ACTS provides fast sim/digi + tracking toolkit for FCCSW
- DD4HepPlugin helps link back to ACTS tracking geometry

FCC-hh tracking geometry

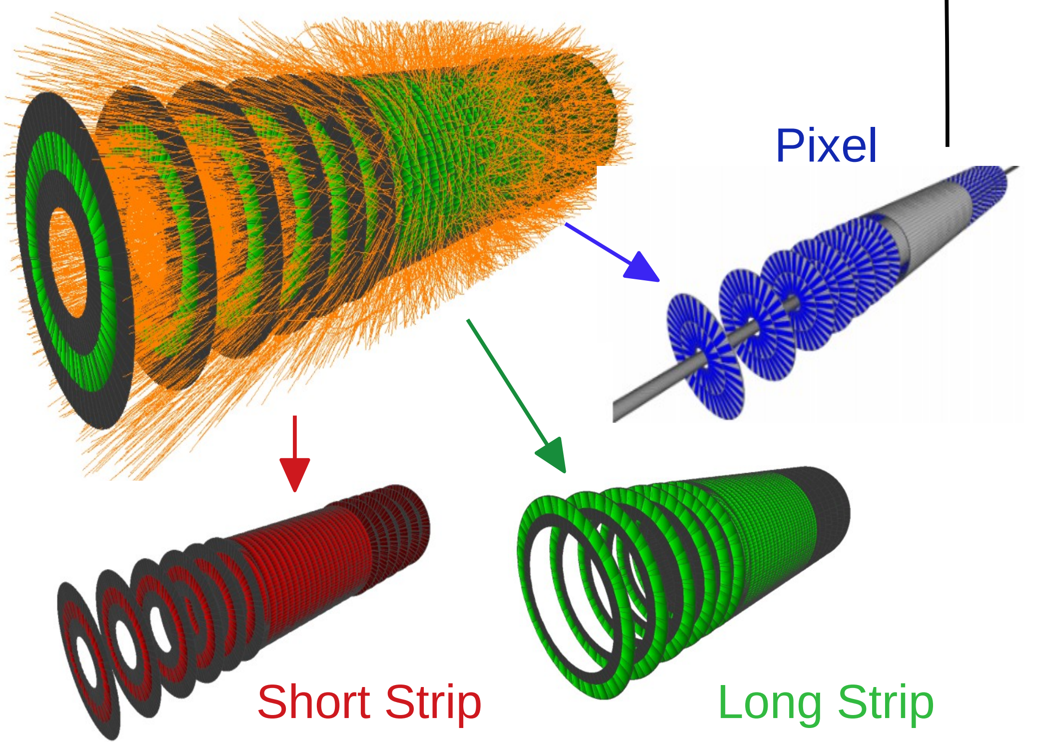


J. Hdrinka

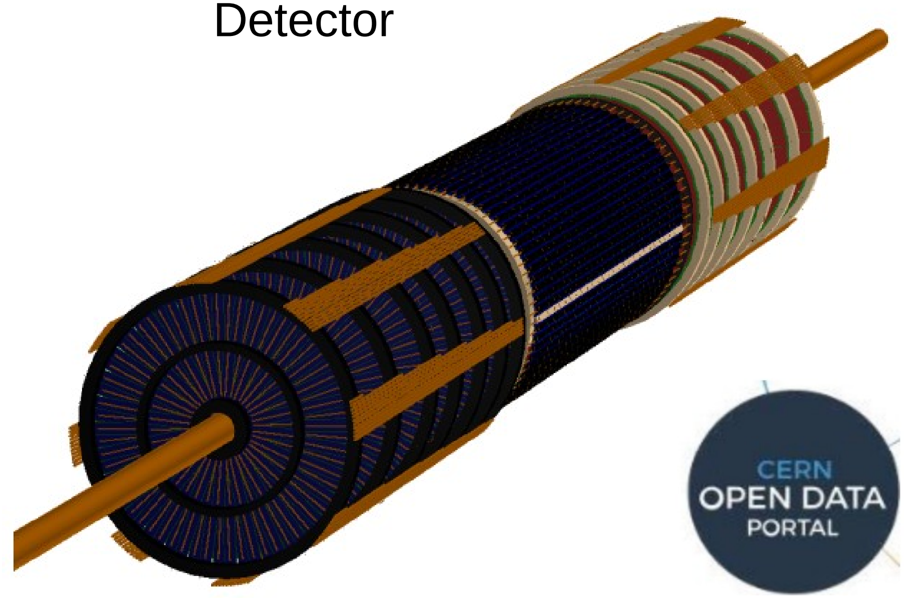
Application: TrackML/OpenData Detector

More details in
[A. Salzburger's slides]

Tracking ML detector with 1000 events from ACTS fast simulation



OpenData Pixel Detector



- Dataset is simulated with ACTS fast simulation
  - Includes multiple scattering, energy loss and hadronic interactions (no particle decay)
  - Not enough material
- TrackML dataset is also used for other projects:
  - e.g. Hep.TrkX & Exa.TrkX project (talk by X. Ju later today)

- Simulating a realistic detector
  - Realistic material description
  - Full material interaction
- Large benchmark dataset to be released as CERN OpenData

Summary

- Large increase in track multiplicity at future colliders needs high performance tracking software with ability to exploit parallel architectures
- ACTS provides a **framework-independent** and **detector-independent** tracking toolkit tested for strict **thread-safety** with developers across experiments
  - Basic tracking modules (geometry, extrapolation engine, EDM, fitter) prototypes are working and optimized
  - Concurrent alignment and calibration infrastructure in progress
  - Working towards a full tracking solution
- ACTS is also designed as a testbed for algorithm R&D
  - e.g. used as fast simulation engine and provide reference implementation of solutions for Tracking ML

Don't hesitate to get in touch if experiments are interested in collaboration with or using ACTS!

Q&A

ACTS members/developers

Members with access to **acts** 39

Andreas Salzburger @asalzbur
Given access 3 years ago

Julia Hrdinka @jhrdinka
Given access 3 years ago

Noemi Calace @ncalace
Given access 3 years ago

Valentin Volkl @vavolkl
Given access 3 years ago

Moritz Kiehn @msmk
Given access 2 years ago

Robert Johannes Langenberg @rlangenb
Given access 2 years ago

Hadrien Benjamin Grasland @hgraslan
Given access 2 years ago

ATS Jenkins @atsjenkins
Given access 2 years ago

Stewart Martin-Haugh @smh
Given access 2 years ago

Karolos Potamianos @karolos
Given access 2 years ago

Edward Moyse @emoyse
Given access 2 years ago

Nicholas Styles @nstyles
Given access 2 years ago

Dmitry Emeliyanov @demelian
Given access 2 years ago

Paolo Calafiura @calaf
Given access 2 years ago

Benedikt Hegner @hegner
Given access 2 years ago

Sarka Todorova @nova
Given access 2 years ago

Wolfgang Liebig @liebig
Given access 2 years ago

Shaun Roe @sroe
Given access 2 years ago

Vincenzo Innocente @innocent
Given access 2 years ago

David Rousseau @droussea
Given access 2 years ago

Markus Elsing @elsing
Given access 2 years ago

Frank-Dieter Gaede @fgaede
Given access 2 years ago

David Chamont @chamont
Given access 2 years ago

Tobias Golling @golling
Given access 2 years ago

Felice Pantaleo @fpantale
Given access 2 years ago

Paul Gessinger-Befurt @pagessin
Given access 1 year ago

Marco Rovere @rovere
Given access 1 year ago

Fabian Klimpel @fklimpel
Given access 1 year ago

John Smith @jrsmith
Given access 11 months ago

Shih-Chieh Hsu @schsu
Given access 11 months ago

Heather Gray @hgray
Given access 10 months ago

Lauren Alexandra Tompkins @tompkins
Given access 10 months ago

Jin Zhang @jinz
Given access 9 months ago

Bastian Schlag @bschlag
Given access 8 months ago

Jovan Mitrevski @jmitrevs
Given access 7 months ago

Xiaocong Ai @xai It's you
Given access 7 months ago

Simone Pagan Griso @spagan
Given access 5 months ago

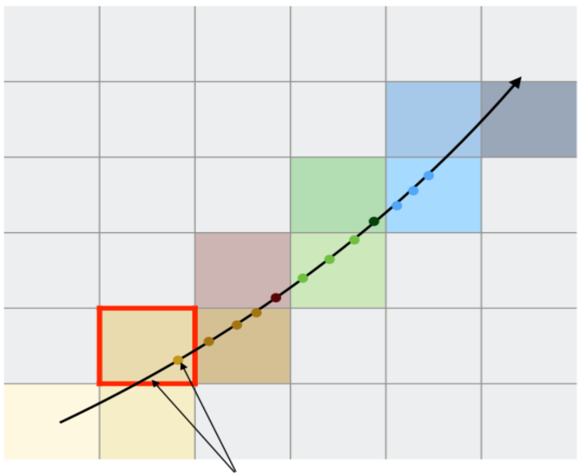
Gang Zhang @gang
Given access 3 months ago

Tim Adye @adye
Given access 4 weeks ago

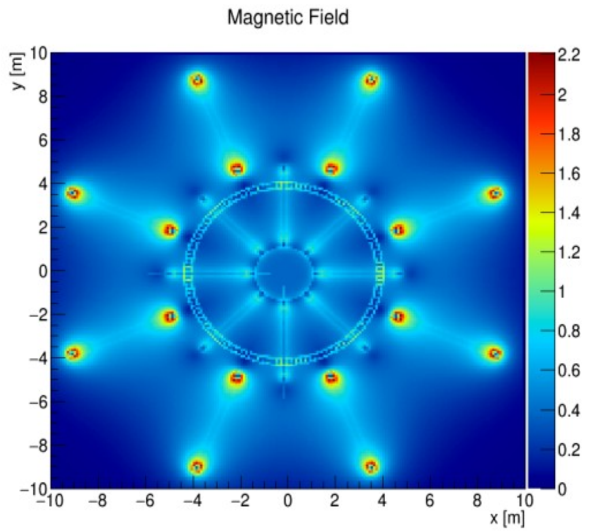
ATLAS
LHCb
FCCSW
CEPC
FASER

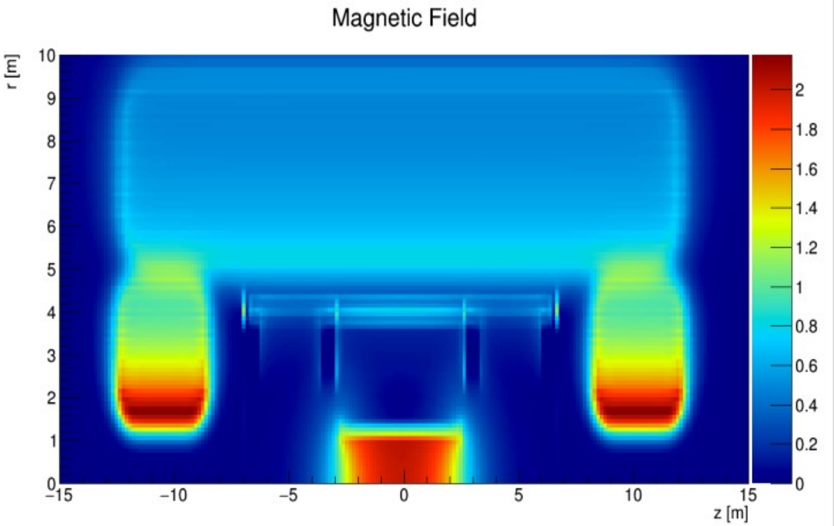
Magnetic field

- Simulation and reconstruction needs extensive lookup to magnetic field
- Cache of filed value could make the access less expensive
  - Facilitate repetitive access to similar locations
- In ACTS, the cache is passed between magnetic field service and client via client function argument
  - Cache is thread-local thus thread-safe



ATLAS Magnetic field in ACTS

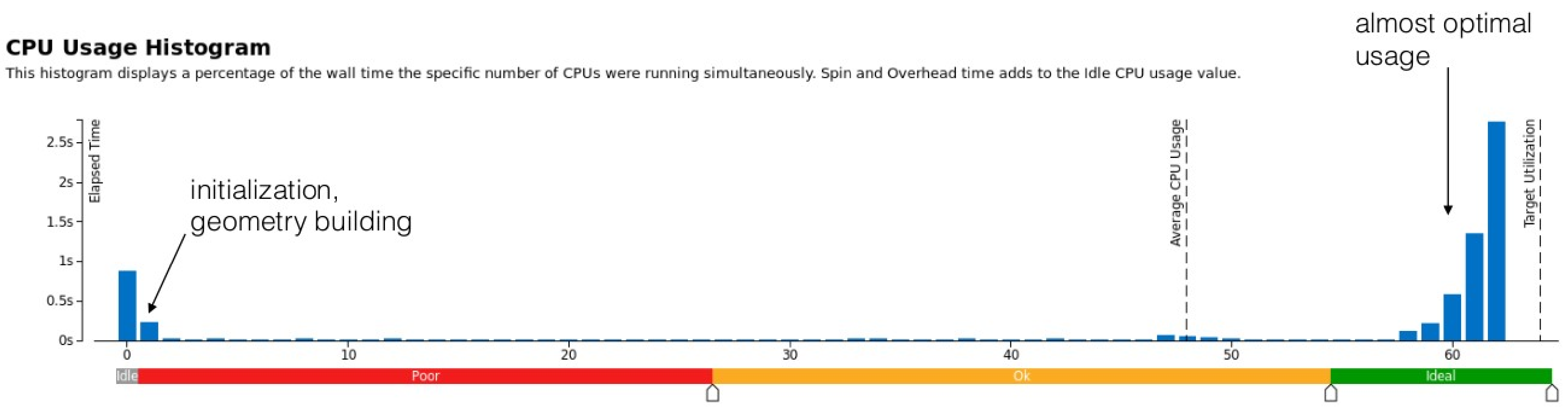




Concurrency test

Intel Xeon e5-2698 v3, 2 sockets
32 Cores, 2 threads per core
64 Processors(cpu's)

- Strategies for thread-safety
  - const-correctness
    - Member method of tools has to be const to forbid altering of internal state
  - stateless tools and visitor cache
    - Tools have no internal cache. Client provides the cache in function call
- Run full propagation test (geometry description, geometry navigation, and particle propagation) with Intel Thread Building Block (TBB) multi-threaded mode
  - part of acts-framework CI test



Vertex Finding

acts › acts-core › Merge Requests › !535

Merged Opened 2 months ago by **Bastian Schlag** Edit Report abuse

Iterative vertex finder

Implements the Iterative Vertex Finder together with the ZScanVertexFinder used as the vertex seeding algorithm. Already adapted to 4D vertexing, ready to be merged after !576 (merged).

Edited 1 week ago by Bastian Schlag

Request to merge `iterative_vertex_finder` **into** `master`

acts › acts-core › Merge Requests › !566

Open Opened 1 month ago by **Bastian Schlag** Edit Close merge request

WIP: Multi adaptive vertex fitter

Implements the multi adaptive vertex fitter, depends on some features from !535 (merged)

Edited 1 month ago by Bastian Schlag

Request to merge `MultiAdaptiveVertexF…` **into** `master` Open in Web IDE Check out branch

The source branch is 62 commits behind the target branch

Now iterating on the design and optimising

InDetPriVxFinder

(Fitting after finding approach)

Track Pre-selection

Tracks compatible with a beam spot are selected.

→

Seed finding

A search for clusters of tracks in Z-impact parameter

→

Vertex fit of all candidates

↓

Rejection of tracks incompatible with vertex candidates.

One track at the time or a fixed fraction.

(loops back to Vertex fit of all candidates)

↓

Fit is completed

Can be used together with any ATLAS vertex fitter.

- A maximal number of primary vertex candidates is fully determined at the seeding stage.
- No possibility of re-using rejected tracks.
- The signal (tagged) primary vertex is selected according to the highest Σp_t of tracks.

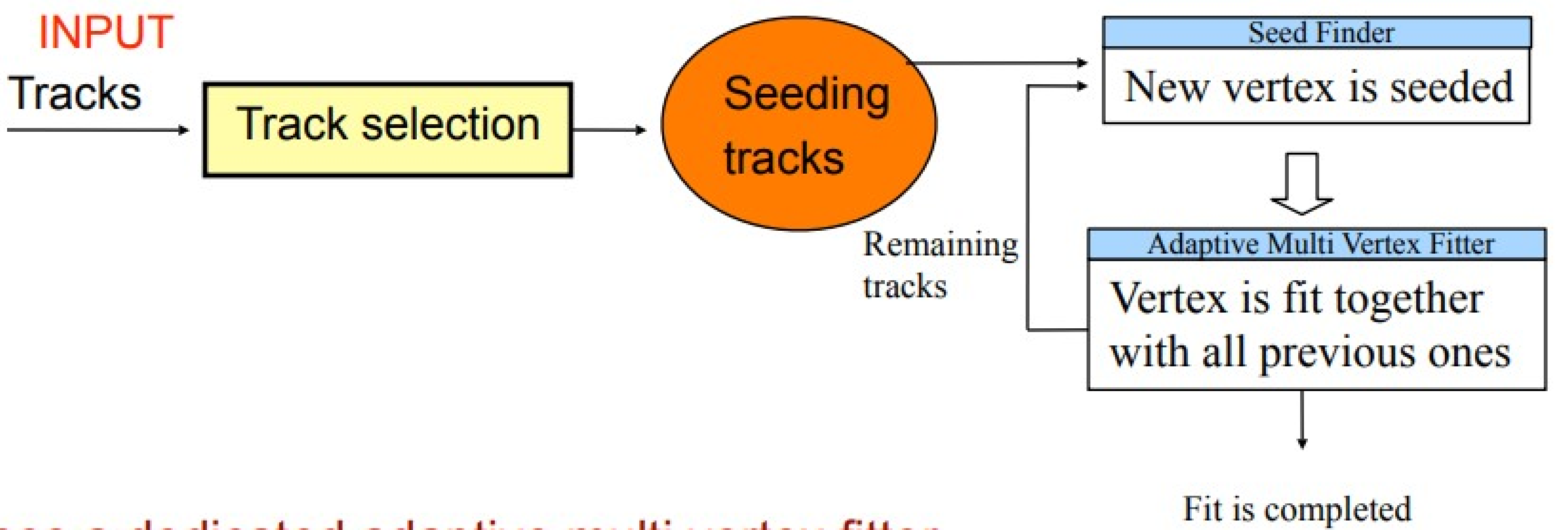
talk by A.Salzburger

AdaptiveMultiVertexFinder

(Finding through fitting approach)

talk by A.Salzburger

- Adaptive multi vertex finder: default ATLAS reco. algorithm



- Uses a dedicated adaptive multi vertex fitter
  - Several vertices are fitted simultaneously, competing against each other in order to get a certain track assigned to them.
  - An annealing procedure is used: the assignement of tracks to vertices gets harder as the fit iteration number increases and the vertex position is known with more precision.
- The signal (tagged) vertex is selected according to the highest $\Sigma p_t^2/N_{trk}$

TrackML in a nutshell

- Accurate simulation engine (ACTS https://gitlab.cern.ch/acts/acts-core) to produce realistic events
  - One file with list of 3D points
  - Ground truth : one file with point to particle association
  - Ground truth auxiliary : true particle parameter (origin, direction, curvature)
  - Typical events with ~200 parasitic collisions (~10.000 tracks/event)
- Large training sample 10k events, 0.1 billion tracks, 1 billion points, ~100GByte
- Accuracy phase (May to August 2018) on Kaggle
  - Participants are given the test sample (with usual split for public and private leaderboard) and run the evaluation to find the tracks
  - They should upload the tracks they have found
  - A track is a list of 3D points
  - Score : fraction of points correctly grouped together
  - Evaluation on test sample with per-mille precision on 100 event
- Throughput phase Sep to Mar 2019 on Codalab
  - Strong CPU incentive

TrackML Challenge

R&D

Accuracy Phase

- **First : *Top Quarks***
  - Johan Sokrates is an industrial Mathematics master student
  - Pair seeding, triplet extension, trajectory following, track cleaning, all **with machine learning for quality selection**
- **Second :**
  - Pei-Lien Chou is a software engineer in image-based deep learning in Taïwan
  - Machine learning to **predict the adjacency matrix**
- **Thirds :**
  - Sergey Gorbunov is a p**hysicist, expert in tracking**
  - Triplet seeding, trajectory following

- **Jury Innovative prize**
  - Yuval Reina is an electronic engineer and Trian Xylouris is an entrepreneur
  - Marginalized Hough transform with **machine learning classifier**
- **Jury Clustering prize**
  - Jean-François Puget CPMP is a software engineer at IBM. He is both competition and discussion Kaggle grandmaster
  - **DBSCAN clustering** with iterative Hough transform
- **Jury Deep Learning prize**
  - Nicole and Liam Finnie are software engineers
  - DBSCAN seeding, **trajectory following with LSTM**

- **Organization pick**
  - Diogo R. Ferreira is a professor/researcher, focusing on data science and nuclear fusion
  - **Pattern matching**

Throughput Phase

- First :
  - JSergey Gorbunov is a physicist, expert in tracking
  - Triplet seeding, multiple passes trajectory following
- Second :
  - Dmitry Emeliyanov is a physicist
  - Connection graph, Cellular automaton, graph traversal with Kalman Filter
- Thirds :
  - Marcel Kunze is a computer scientist
  - Solution based on top quark, trained navigation on DAG of voxels to find doublets and triplets

- No jury selection yet

Sponsors

→ Porting several of the solutions into ACTS

→ TrackML Workshop at CERN on July 1-2 2019

https://indico.cern.ch/event/742793/contributions/3291192

Bremstrahlung Electron loses energy by a fraction 1 −z

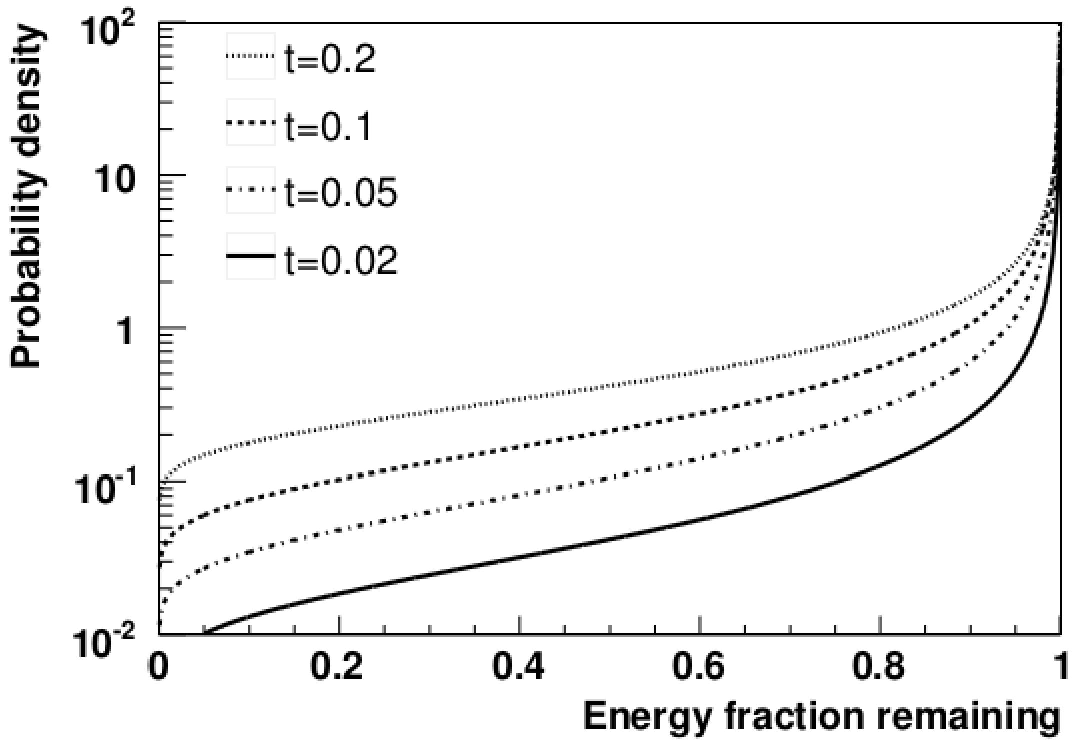


Figure 5.14: *The Bethe-Heitler distribution, which describes the energy loss of an electron when it traverses a material layer. t is the thickness of the layer in radiation lengths, i.e. $t = \frac{x}{X_0}$, with x the thickness in mm, and X_0 the radiation length in mm.*