ACTS for ITk in ATLAS

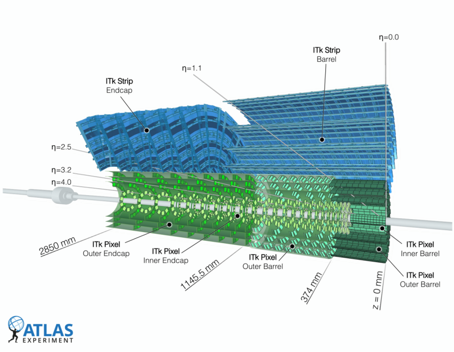
Luthien Mlynarska[1], Tomasz Bold[1], Petr Balek[1], Alexis Vallier[2], Carlo Varni[3], Noemi Calace[4]

[1]AGH Univerity of Krakow
[2]Laboratoire des 2 Infinis, Toulouse
[3]Berkeley University of California
[4]CERN

18th November 2024

Run 4

- In Run 4 current ATLAS Inner Detector will be replaced by the ITk, which will have higher η coverage.
- The number of events per bunch crossing will not increase in the case of Pb+Pb collisions, but the amount of signals in ITk will be comparable to HL LHC pp collisions.



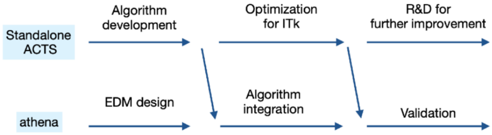
https://cds.cern.ch/record/2908925

Tracking in ATLAS

- Track reconstruction in ATLAS is conducted using the Athena software.
- **Tracking is very CPU-consuming**; scales with the number of interactions (up to 200 in Run 4), or with centrality in Pb+Pb.
- Tracking procedure used at ATLAS:
  - Pre-processing step: clustering and space point (3D representations of clusters) formation
  - Identification of seeds, e.g. triplets of space points compatible with a helix trajectory
  - Iterative Combinatorial Kalman Filter (CKF) extending track seeds into tracks candidates using all clusters compatible with the estimated trajectory
  - Ambiguity solver to resolve overlaps among track candidates

ACTS integration into Athena

- Tracks can have different topologies, but algorithms are the same. Many strategies to choose from, e.g. quality of seeds - flexibility ACTS can provide.
- **Benefits from using ACTS are not for free!**
- Extensive work is required from the side of the experiment.
  - Definition of an Event Data Model, connecting experiment and ACTS' representation of the core objects
  - Integration of the ACTS components into the reconstruction chain
  - Possible adaptation of downstream objects reconstruction to the above changes



Carlo Varni @ CHEP-24

Event Data Model

Internal ACTS EDM

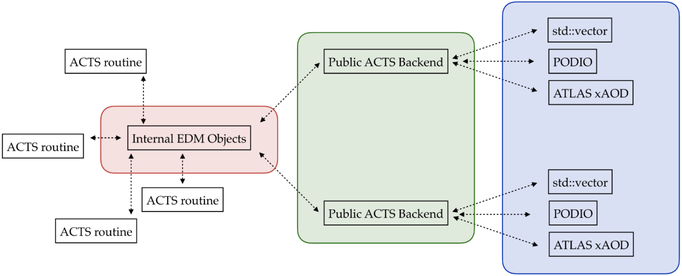
- Data objects to pass around between different parts of the ACTS library
- Library-specific, tightly coupled to the algorithm

Public ACTS EDM

- Data objects clients directly interact with
- Should be experiment agnostic
- Extensible by experiment, easy integration

Experiment ACTS EDM

- Data objects experiments use, can save to file



Paul Gessinger @ CHEP 2023

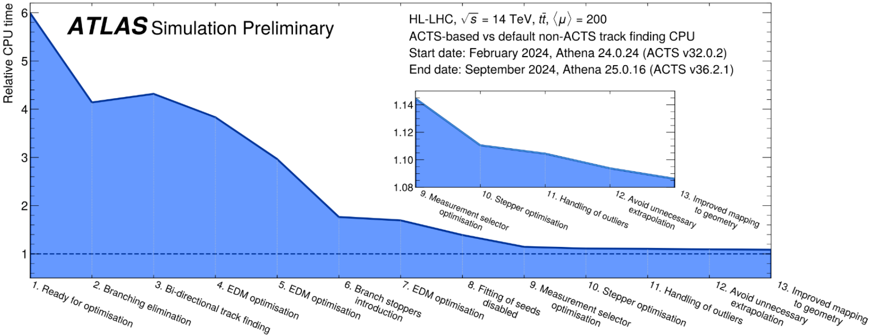
Integration challenges

- The ACTS toolkit is actively developed and optimized.
- In Athena, we monitor the effects ACTS newest changes have on Athena and perform the Athena test for every pull request in ACTS before creating a new ACTS tag.
- We have a mechanism in place in Athena that allows to re-compile the software with a custom version of ACTS, add dedicated patches for Athena, and to produce a dedicated main—ACTS nightly release.
- For the monitoring of all the ACTS pull requests, we use Apogee to check if they are not breaking Athena.
- This strategy worked well for ATLAS - happy to share know-how with other ACTS customers.

Optimization

- Aiming at a complete demonstrator by the end of this year, i.e. finalize integration of all the required components in the reconstruction chain.
- Already starting tuning and optimization of algorithms.
- Target: similar or faster execution time than current ATLAS software counterparts, while achieving same or better physics performance.
- Integration is fast in the case of ATLAS, as most of the ACTS developers are ATLAS members.
- **A strong participation from other experiments will benefit ACTS as an experiment-independent toolkit.**

CPU optimization in Athena

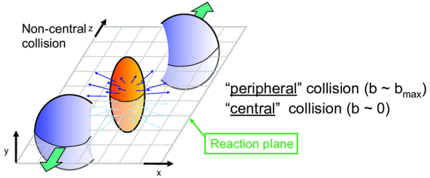


ATL-PHYS-PUB-2024-017

Heavy Ion related contributions

HI-specific research

- During the heavy ion LHC run, the luminosity is low, and we don't observe collision pileup (1 event per bunch crossing).
- We only see one vertex with very high multiplicity.
- We categorize events by their centrality (central, peripheral, ultra-peripheral).
- We can go from analogous 2009 *pp* event conditions to Run 4 *pp* conditions in a matter of single collisions.



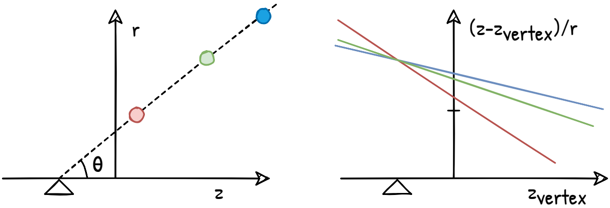
https://www.researchgate.net/publication/224904960_QCD_in_heavy_ion_collisions

Heavy Ion physics configuration

- My qualification project in ATLAS involved implementation of ACTS to reconstruct heavy ion events.
- I was able to run it natively; only parameters needed to be tuned.
- **Tracking is much slower for HI (about 10$\times$).**
- High hit density in central events results in many seeds and high combinatorics.
- Currently, I am working on the seed persistifying algorithm to optimize seeding.

Vertex finding using Hough transform

- Having vertex z known prior to seeding will allow for an additional constraint
- How to get a vertex coordinates before tracks are reconstructed?
  $\Rightarrow$ We can use Hough transform to get it from spacepoints.
- In ATLAS uniform solenoid field, a track has some angle θ and originates at z_{vertex}.
- Apply Hough transform, so each spacepoint is projected into this parameter space.

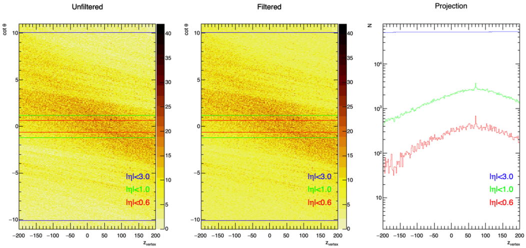


Vertex finding using Hough transform

- Instead of θ, we use $\cot(\theta) = (z - z_{vertex})/r$, which is more computationally effective (only addition/multiplication, no trigonometric functions).
- Identify the crossing (peaks in parameter space), z_{vertex} shared by all tracks, the crossings will make a peak
- Projection to z_{vertex} further amplifies the peak.
- Position of the peak indicates a vertex z coordinate.

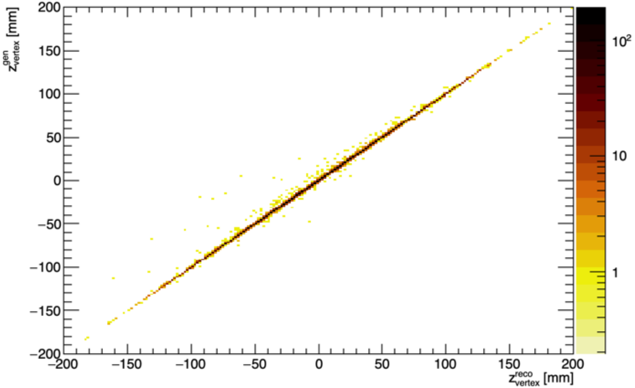
1 event in OpenData Detector, 69.9k spacepoints

- Algorithm needs to be flexible to handle ultra-central and peripheral HI collisions.
- Example challenge in ultra-central collision - too many random coincidences in parameter space.
- Solution: variable image space (granularity) and mapping (dynamic selection of space points e.g. $|\eta|$).



- Large $|\eta|$ range has no visible peak
- Smaller $|\eta|$ ranges have also finer $\cot(\theta)$ bins

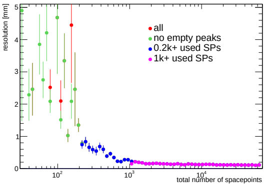
Optimal strategy and results

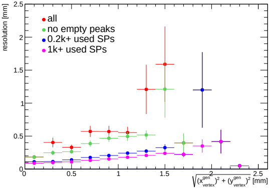


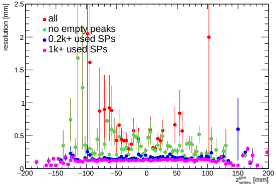
- Target is to have ∼10k spacepoints within the $|\eta|$ range.
- Limit $|\eta|$ if we have too many spacepoints.
- If there are less than 10k spacepoints, keep $|\eta| < 3.0$.
- Vertex was always found with 10k spacepoints.

Resolution

- Resolution is σ of $z^{gen}_{vertex} - z^{reco}_{vertex}$.
- Always assuming $x_{vertex} = 0\,\mathrm{mm}$, $y_{vertex} = 0\,\mathrm{mm}$, algorithm is highly sensitive to unknown beamspot.
- Algorithms optimizations are applied depending on multiplicity (left).
- Varying algorithm performance, excellent resolution where needed the most (central).







Vertex finding summary

- It is possible to estimate vertex position using only spacepoints.
- Might have positive impact on seeds formation or filtering.
- Algorithm is very fast.
  - Background in high-multiplicity events easily mitigated.
  - Lack of entries in low-multiplicity events is not a crucial problem.
- Many opportunities for further tuning:
  - How to limit $|\eta|$ depending on number of spacepoints?
  - What amount of bins to use in $\cot(\theta)$?
  - How exactly to fill the Hough space?
- ... and for further improvements:
  - Finding more peaks?
  - Finding the peak iteratively, so X and Y position of the vertex can be also estimated?
- Will be implemented in ACTS.
- ... and hopefully adopted by ATLAS for Run 4.
- Read more here.

Conclusions

- ACTS is an experiment-independent tracking software, but adapting it requires effort.
- A work is required from the experiment side to implement ACTS, e.g. communication with Event Data Model.
- Experiments using ACTS would benefit from an automated integration procedure, as does by ATLAS.
- ACTS is able to easily accommodate changes coming from heavy ion collisions.
- HI specific conditions open possibility of optimisations, fast HT based vertex z position finder preceding actual tracking sequence will benefit seeding and downstream processing.