Fast track seed selection for track following in the Inner Detector Trigger track reconstruction

ATLAS EXPERIMENT

1. Inner Detector (ID)

- ATLAS subdetector for track & vertex reconstruction
- 3 types of layers: **Pixel**, Semiconductor tracker (**SCT**), Transition Radiation Tracker (**TRT**)
- All detector layers either **barrel** or **endcap**

- Fast Track Finder (**FTF**) in ID uses **"triplet-seed"** method:
  - Triplets of detector hits/spacepoints (**SPs**) taken
  - Cut applied on selection
  - Track following procedure performed
  - Track selected or rejected

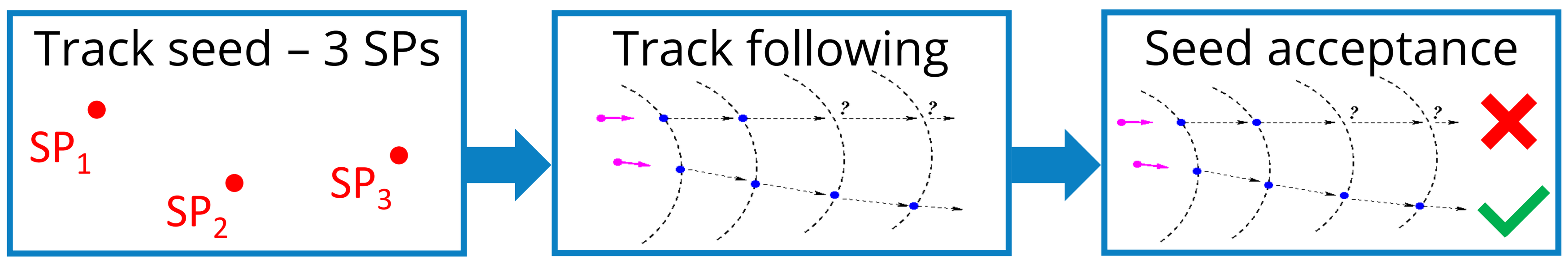

Figure 1: Approximate diagram of the track following algorithm used in the ID

2. Improving Fast Track Finder

- Track following is **computationally expensive**
- Many **Pixel-only** seeds end up being rejected
  - ➢ A lot of **CPU time** is spent on combinatorial track following for bad seeds
- To speed up FTF, introduce **pre-filtering** procedure to identify and reject bad Pixel seeds
- Ensure no significant loss of **track finding efficiency**

3. Approximate Track Extrapolation

- For Pixel-only seeds, spiral track trajectory is approximated as **linear** in RZ and **parabolic** in XY coordinate systems (**c.s.**)[1]:
  - In **RZ**, straight line is fitted through SP_1 and SP_3
  - In **XY**, UV c.s. used: SP_3 is at origin, U-axis passes through SP_2, parabola fitted such that: $v(u) = au^2 + bu$
- Trajectory Intersection Points (**IPs**) with SCT detector layers are found and stored

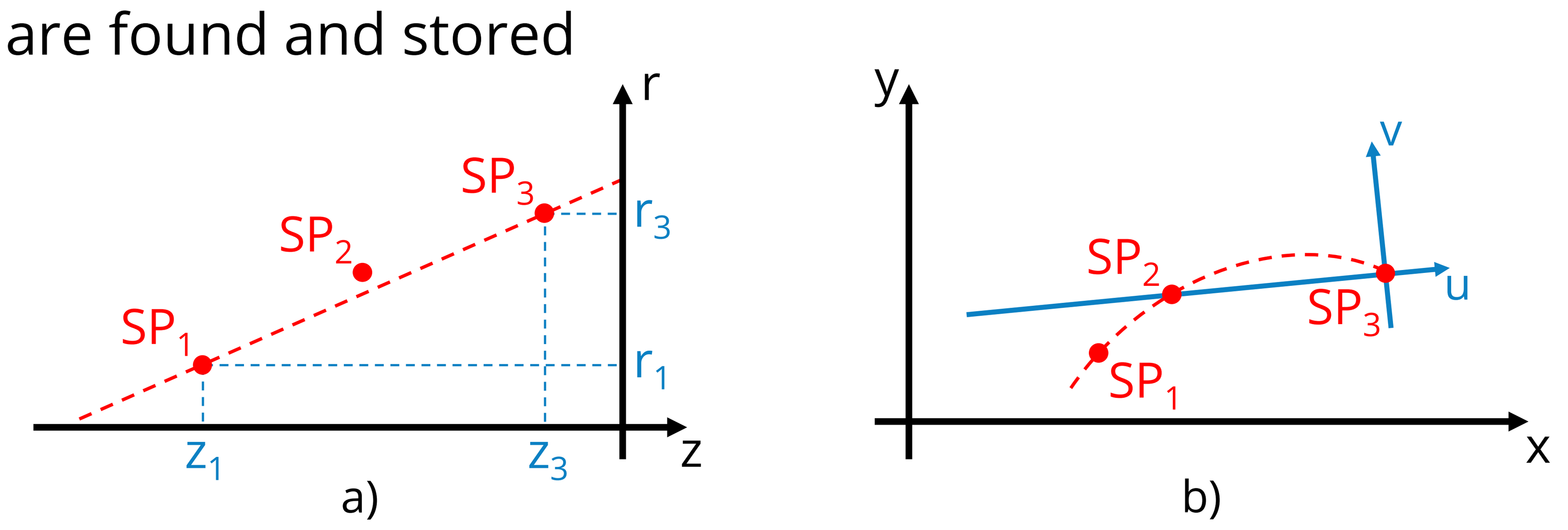

Figure 2: Approximate track extrapolation. a) straight line extrapolation in RZ c.s., b) parabolic track trajectory approximation in XY c.s.

4. Reference-related coordinates

- To simplify Kalman filter (**KF**) calculations, **reference-related** coordinate system is defined
- Idea is to use **sequence of IPs** as reference trajectory
  - ➢ Track is then described as almost **straight line**
- New c.s.: (s, Δ),
  - s is **track path length**
  - Δ is **perpendicular distance** from IP (in 2D, $\Delta = (\Delta_x, \Delta_y)$)
- Eliminates requirement for **magnetic field** modelling

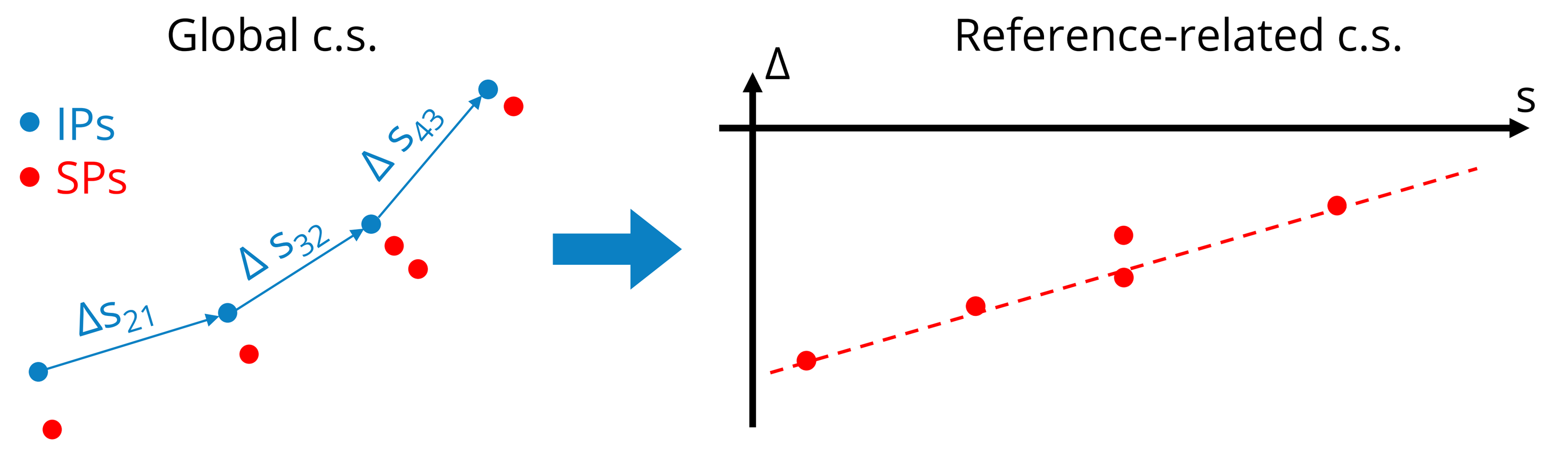

Figure 3: Conversion of spacepoint coordinates from global c.s. to reference-related one. Note that Δ is two-dimensional, but visualised as one-dimensional

[1]ATLAS uses a right-handed coordinate system with its origin at the nominal interaction point (IP) in the centre of the detector and the z-axis along the beam pipe. The x-axis points from the IP to the centre of the LHC ring, and the y-axis points upwards. Cylindrical coordinates (r, φ) are used in the transverse plane, φ being the azimuthal angle around the z-axis.

5. Kalman Filtering

Algorithm (from initial state at first IP):

1. Track state **propagated** to the next layer / IP
2. For each gathered SP, a new track candidate is created:
   - Based on χ^2, hit is either **successful** or **missed**
   - For successful hits, track state is **corrected** based on SP spatial information
3. Only N **best tracks** selected, rest discarded
4. Best track states propagated to the next layer / IP, etc…

Repeated until track candidates propagated through **all IPs**

6. Classifying Track Seeds

- To classify seeds for acceptance/rejection, two **features** were generated for each track candidate:
  - **Track Quality**: calculated during KF, successful hits rewarded, missed hits penalised
  - **Hole Value**: scalar value based on length and position of the largest streak of missed hits in a row (see Fig. 4)

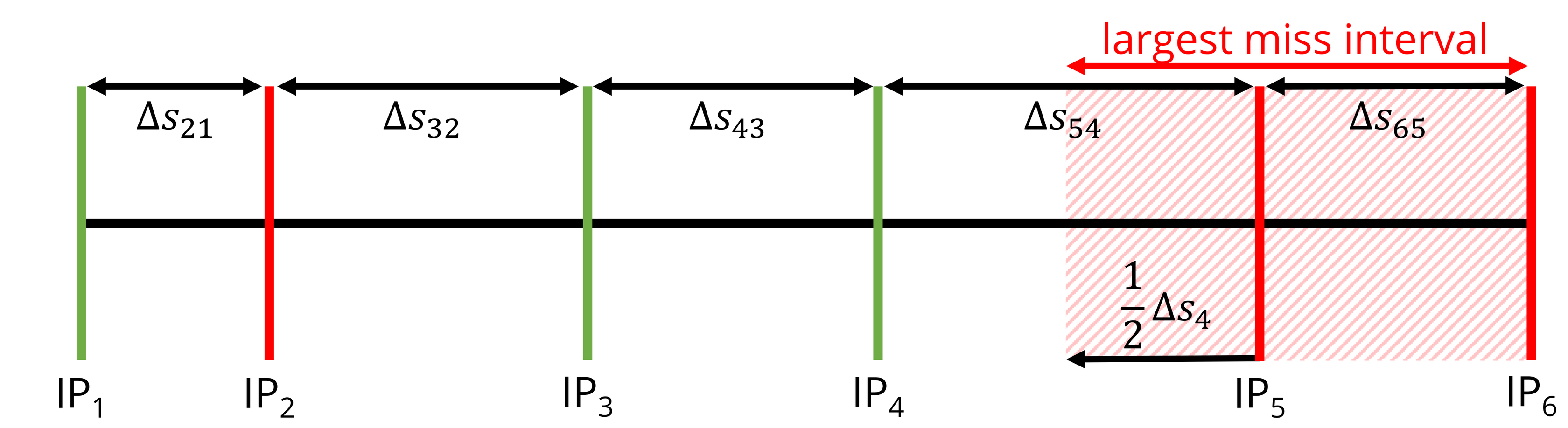

Figure 4. Visualisation of the hole value feature. Successful (missed) hits shown in green (red). Hole value is the integral of importance function over miss interval.

- Using above features, **Support Vector Classifier** was trained
  - Polynomial kernel of order 2 was selected
  - Class weights were adjusted for **prioritising good seeds** acceptance, to limit loss of track finding efficiency
  - Parameters optimised using grid search, maximising True Negative Rate (**TNR**) for True Positive Rate (**TPR**) ≈ 0.95
- Resultant classification:
  - Achieved **TPR** $= 0.96 \pm 0.02$, **TNR** $\approx 0.61 \pm 0.01$
  - Saved as Look-up Table (**LUT**) for fast inference during online Fast Track Finder within Athena Framework

7. Results

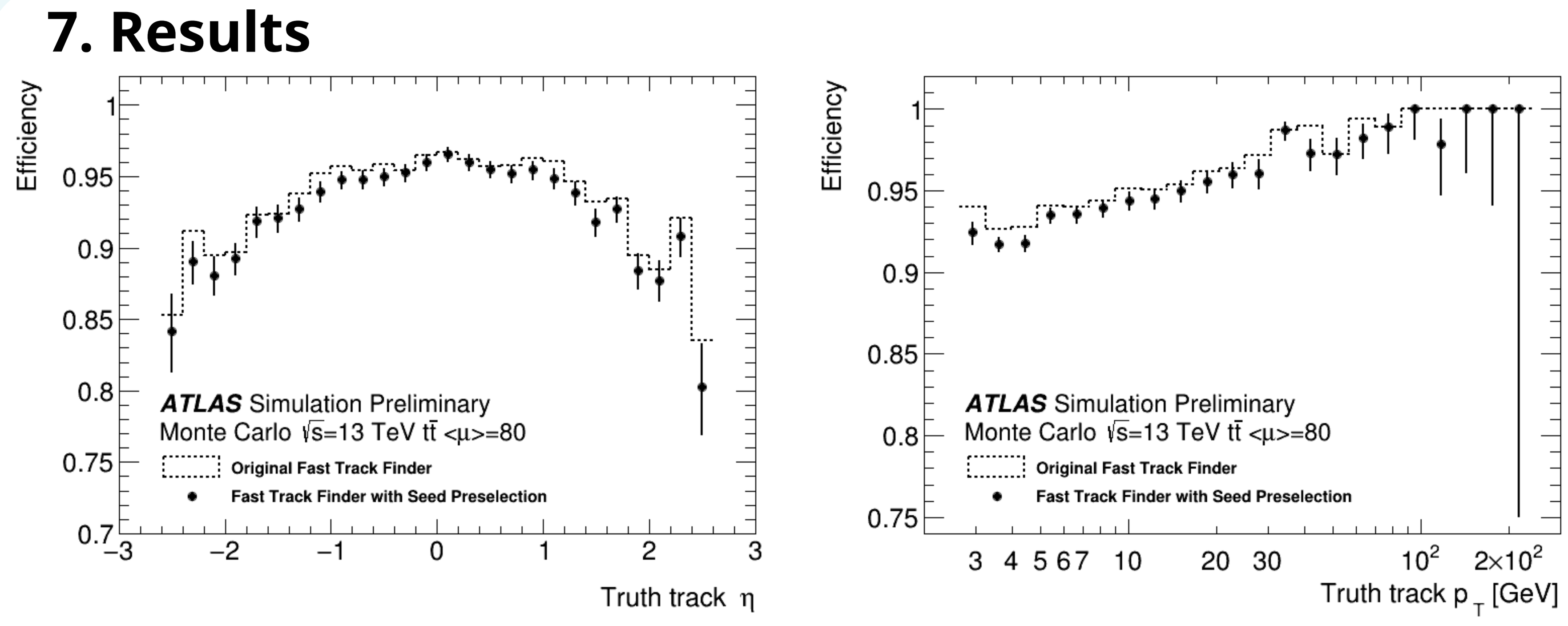

Figure 5: Track Finding efficiency dependency on track η (left) and p_T (right) [1].

For $t\bar{t}$ full detector tracking with pile-up $\langle\mu\rangle = 80$:

- Mean **efficiency loss**: **0.7%** (93.2% → 92.5%)
- Relative mean CPU **time**: **0.89** (speed-up factor: **1.12**)

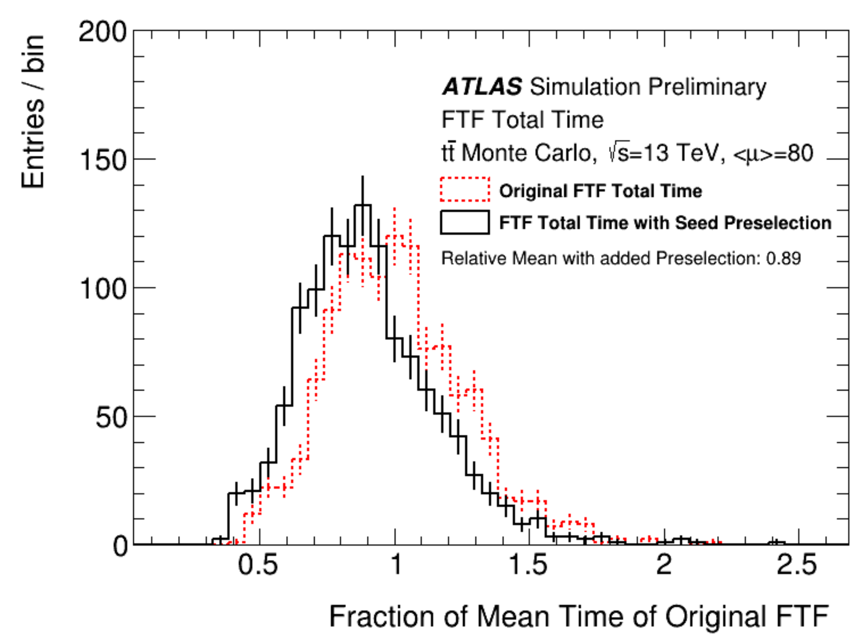

Figure 6: Relative time improvement of FTF with preselection [1].

- Result can be further modularly improved via better LUT
- Can be adapted to new ATLAS Tracker geometry for HL-LHC

[1] ATLAS Collaboration, "Fast track seed selection for track following in the Inner Detector Trigger track reconstruction," CERN, Geneva, Plot. ATL-COM-DAQ-2022-091, Oct 2022. [Online]. Available: https://cds.cern.ch/record/2834109/.

Andrius Vaitkus, University College London, on behalf of the ATLAS Collaboration

UCL