LHCb's Real-time Analysis Trigger for Run 3

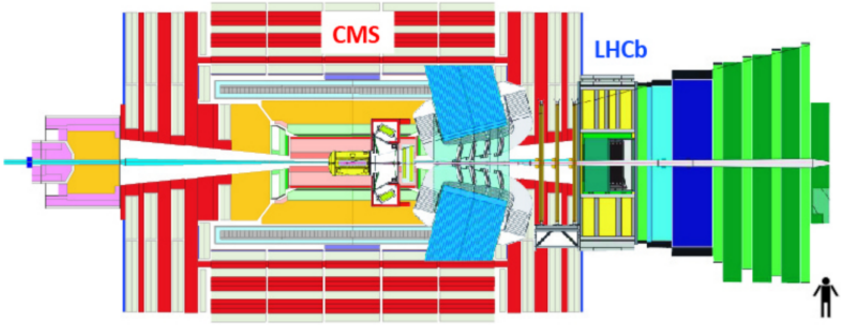
(CPU-based) Online Track Reconstruction

André Günther[1]

[1]Physikalisches Institut Heidelberg

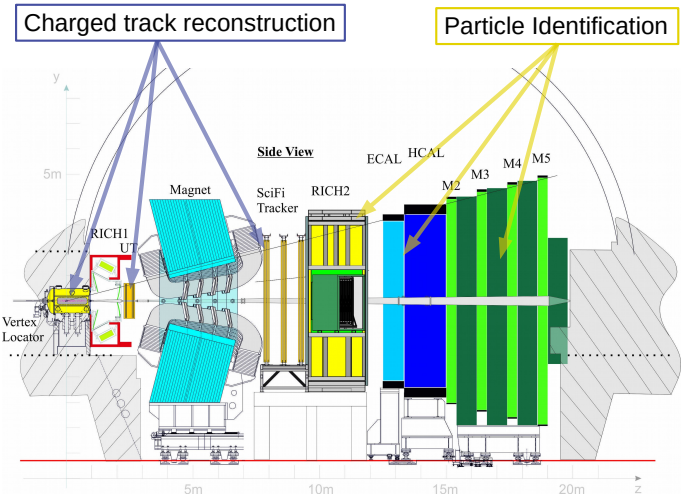
HighRR Seminar 15th June 2022

LHCb - A Flavourful Experiment



- forward spectrometer with $2 < \eta < 5$
- optimised for flavour physics of B and D mesons
  - $\rightarrow$ precise tracking (high single hit efficiency and resolution)
  - $\rightarrow$ high vertex resolution
- by now "general purpose" (EW physics, Exotics, LFV, DM, ...)

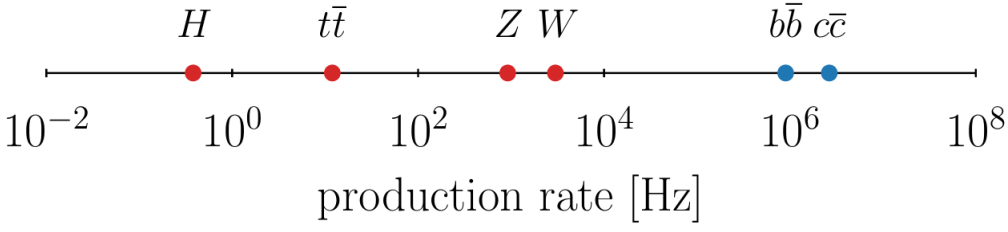
The LHCb Detector for Run 3 (now!)



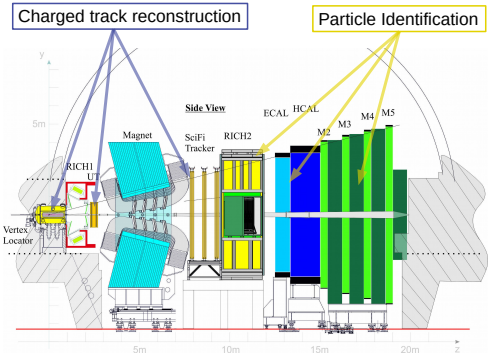
Enter the MHz Signal Era

$\mathcal{L} = 2 \times 10^{34}\ \mathrm{cm}^{-2}\mathrm{s}^{-1}$ (ATLAS/CMS)
$\mathcal{L} = 2 \times 10^{33}\ \mathrm{cm}^{-2}\mathrm{s}^{-1}$ (LHCb)

$\sqrt{s} = 14$ TeV



Trigger? - What are you talking about?



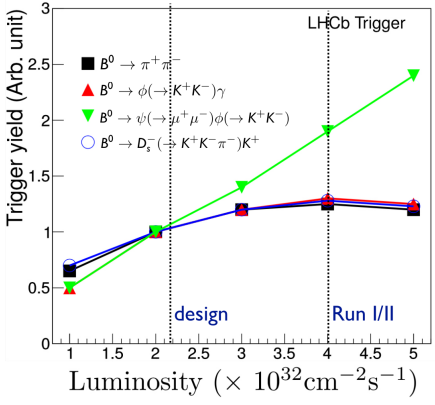
Insufficiency of Hardware/Low-Level Triggers

LLT: FPGAs on calorimeter and muon detectors

→ fast check for high E_T (or p_T)

→ done by LHCb's Level-0 trigger in Run I+II

→ reduced event rate to 1 MHz



- Example: Run 1 already 0.5 light, long-lived hadrons per event
  → signatures saturate trigger
- but want to take more data in Run 3 → luminosity $\times 5$

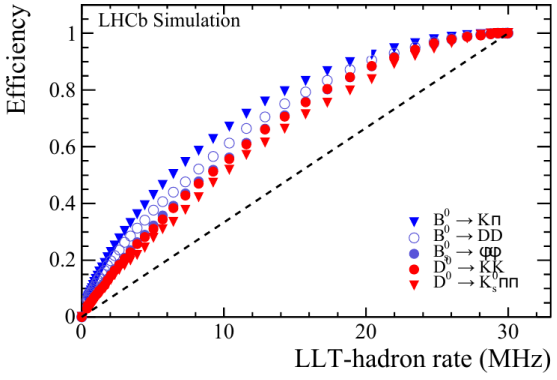
Insufficiency of Hardware/Low-Level Triggers

LLT: FPGAs on calorimeter and muon detectors

→ fast check for high E_T (or p_T)

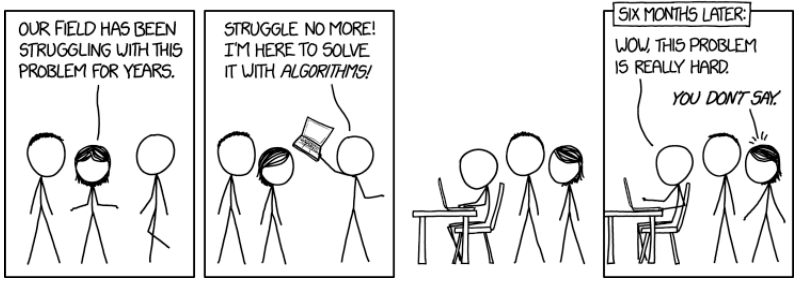
→ done by LHCb's Level-0 trigger in Run I+II

→ reduced event rate to 1 MHz



- Example: Run 1 already 0.5 light, long-lived hadrons per event
  → signatures saturate trigger
- but want to take more data in Run 3 → luminosity $\times 5$

Purely Software-based Trigger



Challenges of the Software Trigger

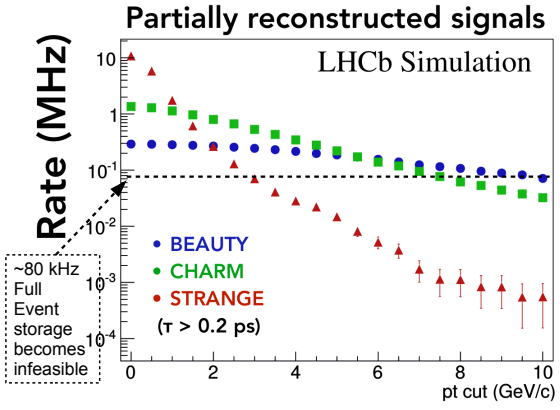
- 5× luminosity vs. Run 2
- bunch crossing every 25 ns

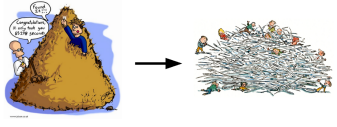
→ detector read-out @ 40 MHz

→ process 30 MHz non-empty events

- squeeze 5 TB/s raw data into 10 GB/s written to disk
- cannot store full event

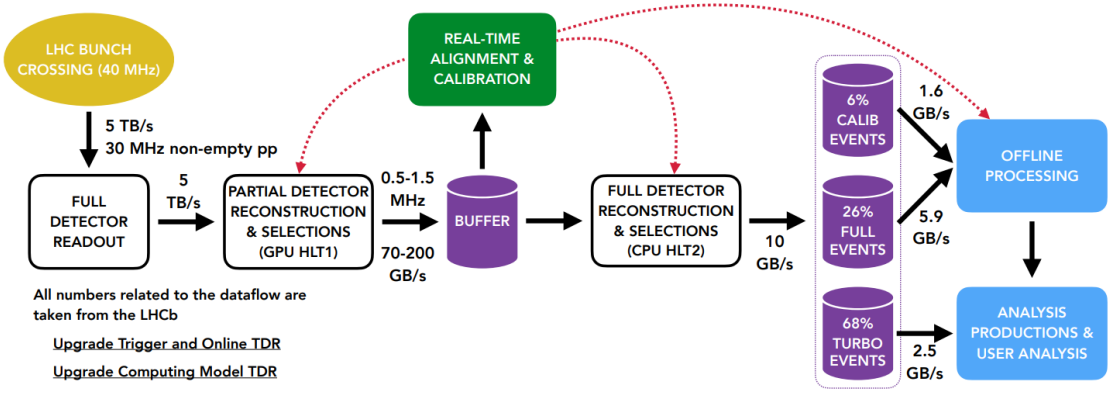
→ real-time analysis



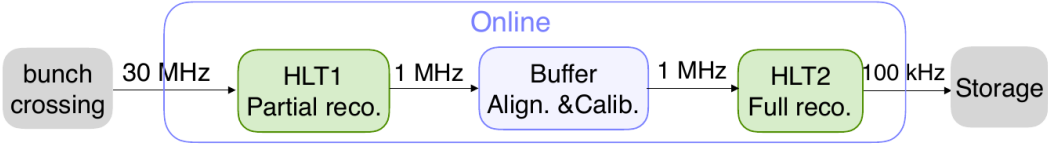


Real-time Data Analysis @ LHCb

- purely software-based two-stage trigger



Online Event Reconstruction

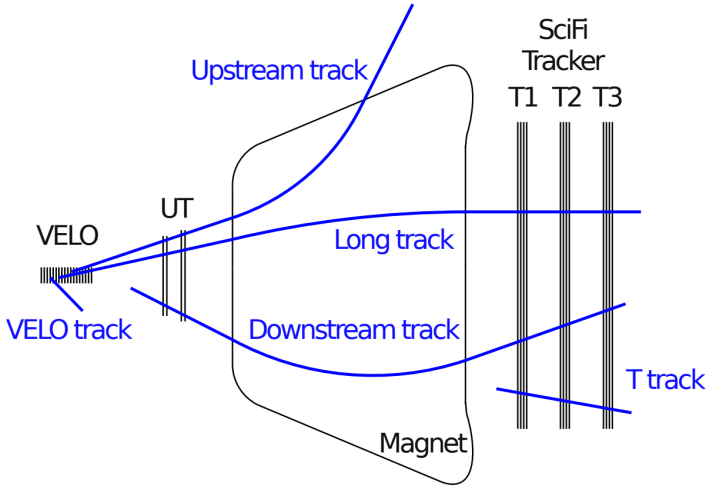


Online Track Reconstruction - Track Types

- reconstruction efficiency (from simulation):

$$\varepsilon_{reco} = \frac{\text{reconstructed}}{\text{reconstructible}}$$

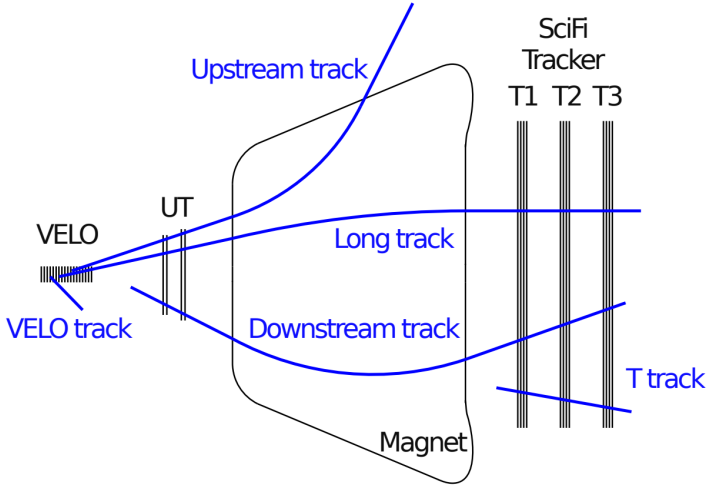
- Long Track reconstructible if MC particle leaves 3 VELO hits and 6 SciFi hits



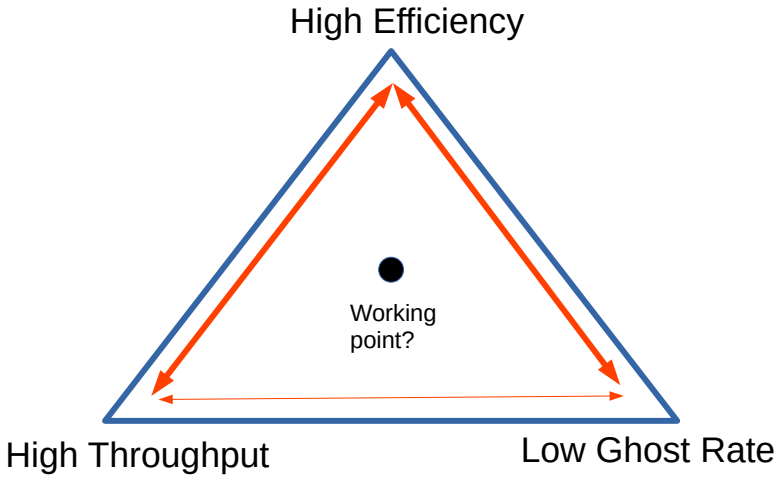
Online Track Reconstruction - Fake/Ghost Tracks

- reconstructed track has less than 70% hits from one MC particle

→ fake track (a.k.a. ghost)



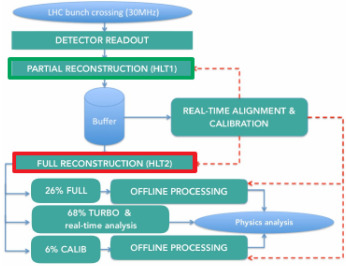
Online Track Reconstruction - Metrics



Two Years Ago

Software Trigger Status

- proof of concept already during Run II
- pre Run III challenges:
  - basically new detector
  - sixfold performance increase needed
- milestones per node (2 CPUs):
  - 30 000 Events/s in HLT1
  - 500 Events/s in HLT2



- [1]HLT1: $\texttt{throughput} = 35\,717$ Events/s
- HLT2: $\texttt{throughput} = 143.5$ Events/s

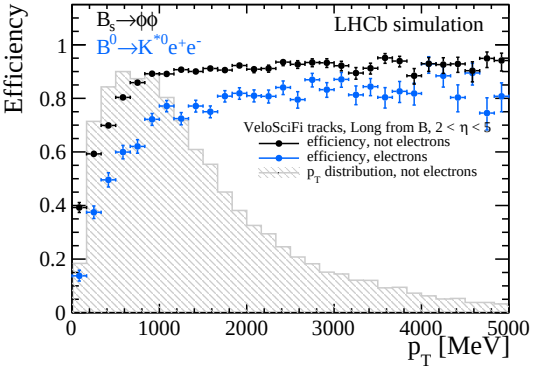
$\rightarrow$ focus shifting towards HLT2

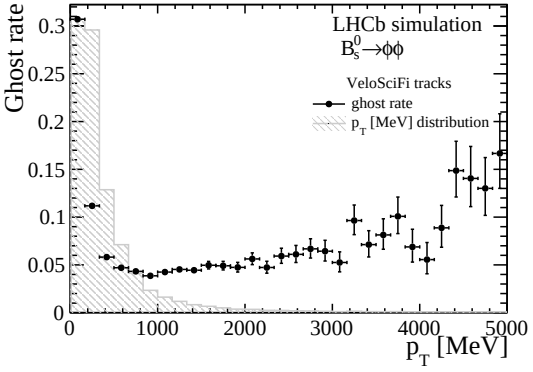
[1] 2×20 threads on Intel Xeon E5-2630-v4

Online Event Reconstruction - HLT1

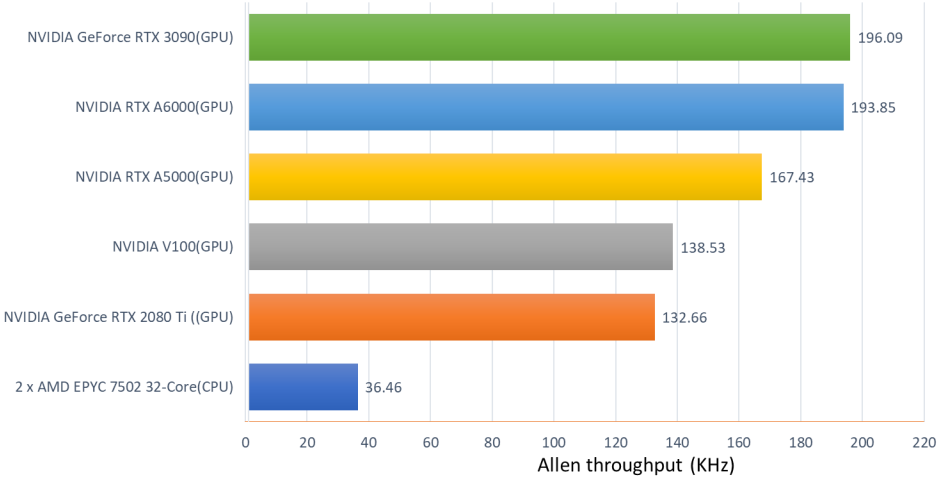
- partial event reconstruction using GPUs $\rightarrow$ new and a first!
  - find charged particle tracks $\rightarrow$ focus on tracks from heavy flavour
  - track parameter estimation using simplified Kalman filter
  - find primary vertices
  - muon + electron identification
  - **no** hadron identification
- allows to have $\mathcal{O}(100)$ flexible selections

HLT1 - Allen Reconstruction Efficiency





HLT1 - Allen Event Throughput



Online Event Reconstruction - HLT1

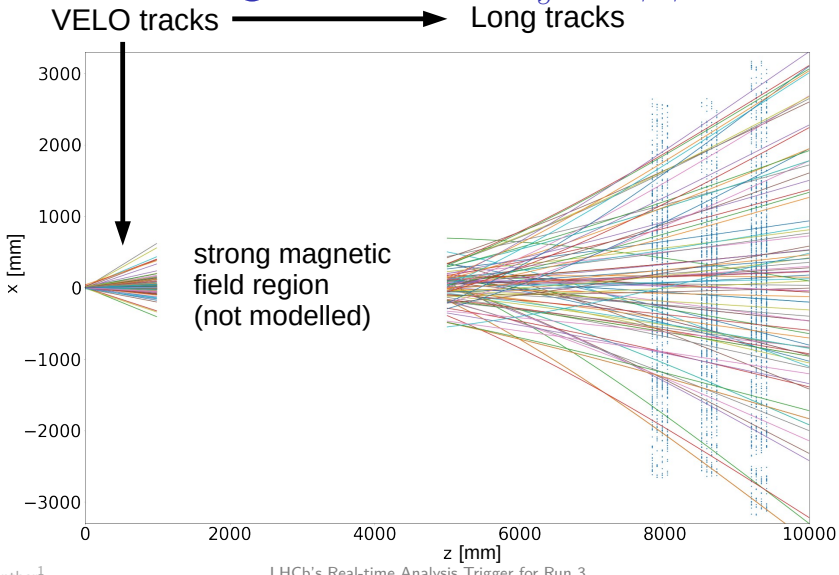
- partial event reconstruction using GPUs $\rightarrow$ new and a first!
  - find charged particle tracks $\rightarrow$ focus on tracks from heavy flavour
  - track parameter estimation using simplified Kalman filter
  - find primary vertices
  - muon + electron identification
  - **no** hadron identification
- allows to have $\mathcal{O}(100)$ flexible selections
- **big achievement and pioneering:** will process 30 MHz event rate using ~ 170 NVIDIA A5000 cards

Online Event Reconstruction - HLT2

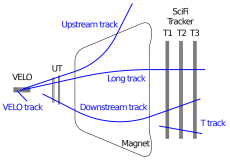
- full offline-quality event reconstruction on CPUs
  - full track reconstruction → that's my domain
  - track parameter estimation using Kalman Filter
  - full particle identification (hadrons, muons, electrons)
  - neutrals reconstruction
  - composite particle building and selection
  - selective persistence

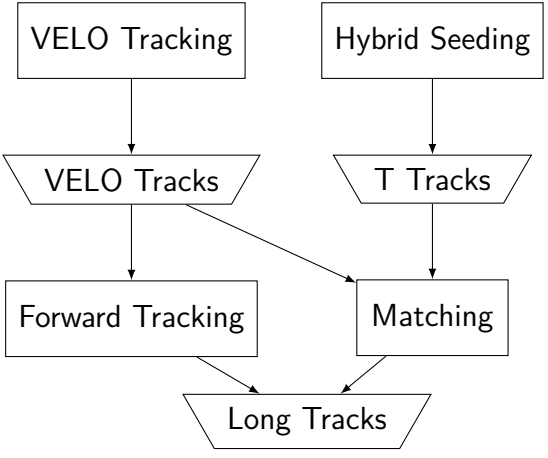
→ full analysis of the event in real-time!

Reconstructed Long Tracks in $B_s^0 \to \phi\phi$ Event



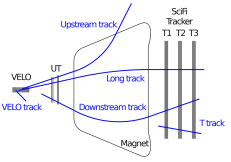
HLT2 Track Reconstruction Sequence

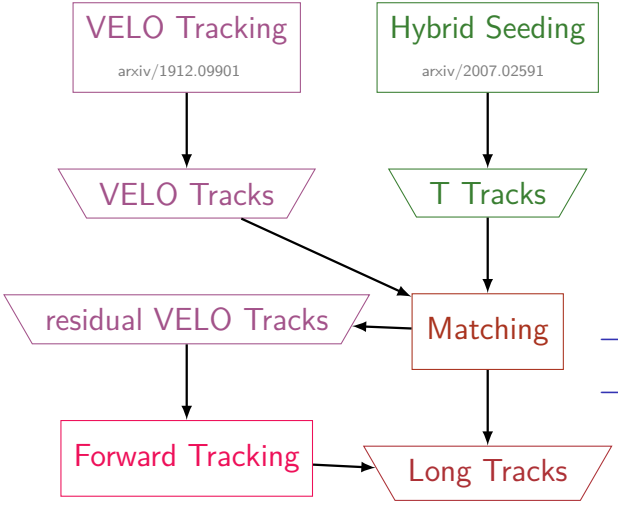




- legacy Run-2-like sequence
- redundancy in finding Long Tracks
- slow because too much work done + need to remove tracks found twice

HLT2 Track Reconstruction Sequence





- two algorithms find Long Tracks
  1. Neural Network (MLP) Matching VELO to T Tracks
  2. Forward Tracking

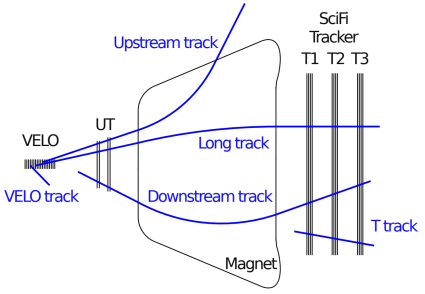
→ avoid redundancy for speed

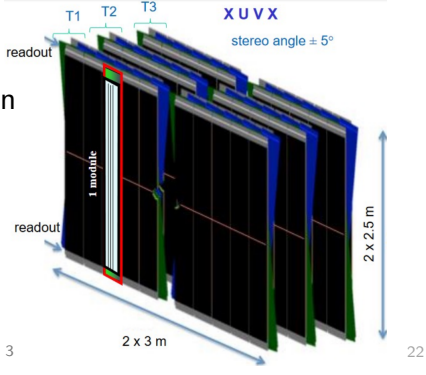
→ run Forward Tracking on not-matched VELO tracks

Forward Tracking - Intro

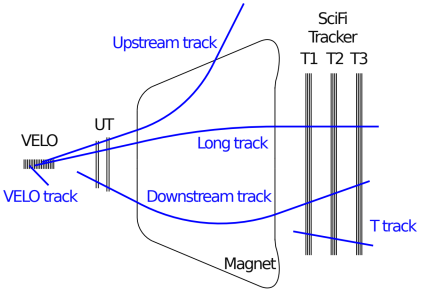
Goal: find "Forward" extension
to VELO track in SciFi Tracker
($\rightarrow$ Long Track)
+ estimate momentum

- extension is set of 10-12 SciFi hits
  - 99% single hit efficiency - 1% dead region
  - $\rightarrow$ 78.5% of all tracks leave 12 hits
  - $\rightarrow$ 99.8% at least 10 hits





Forward Tracking - Intro



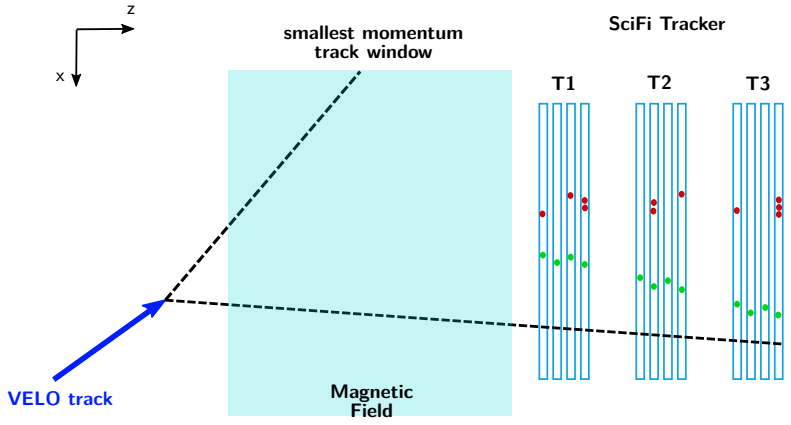
Goal: find "Forward" extension
to VELO track in SciFi Tracker
($\rightarrow$ Long Track)
+ estimate momentum

- extension is set of 10-12 SciFi hits
- fringe magnetic field in SciFi Tracker $\rightarrow$ hits on slightly curved line

$\rightarrow$ Hough-like transform algorithm to recognise lines

Forward Tracking - Algorithm

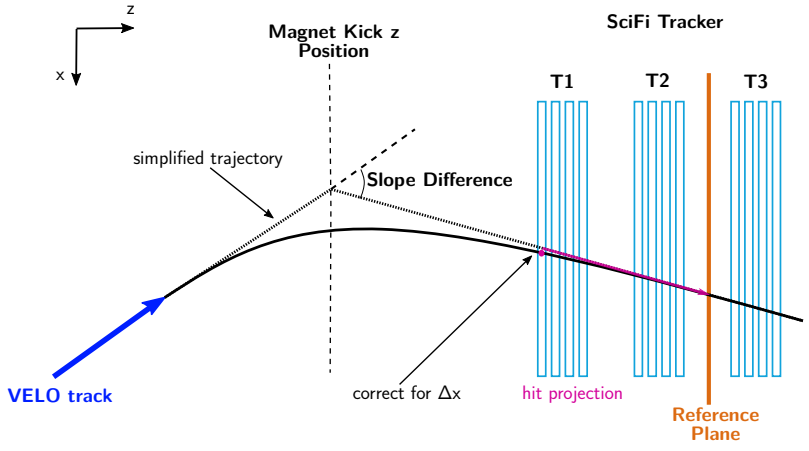
1. define hit search window for VELO track state $(x, y, \frac{\partial x}{\partial z}, \frac{\partial x}{\partial z}, \frac{q}{p})$
   $\frac{q}{p}$ unknown, assume $p > 1.5\,\text{GeV}$ use Polynomial$(\frac{\partial x}{\partial z}, \frac{\partial x}{\partial z}, p)$



Forward Tracking - Algorithm

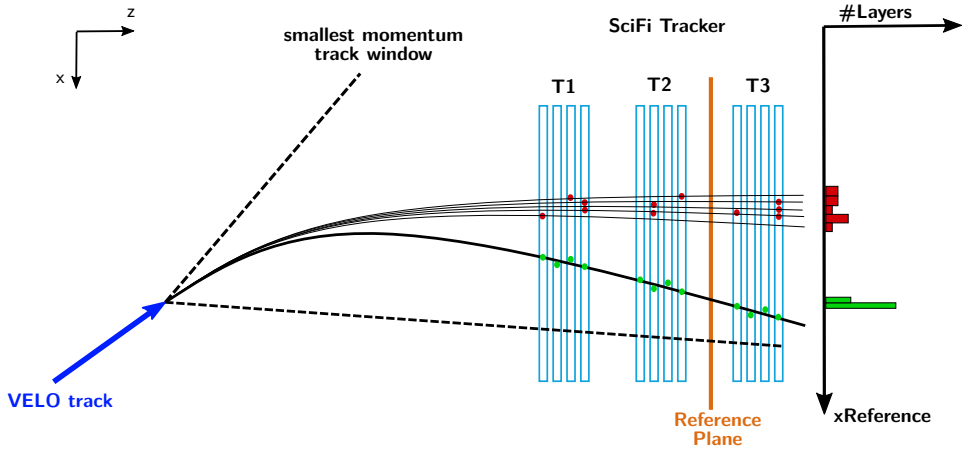
LHCb-2002-008

2. treat magnet as optical lens to simplify track and hit projection



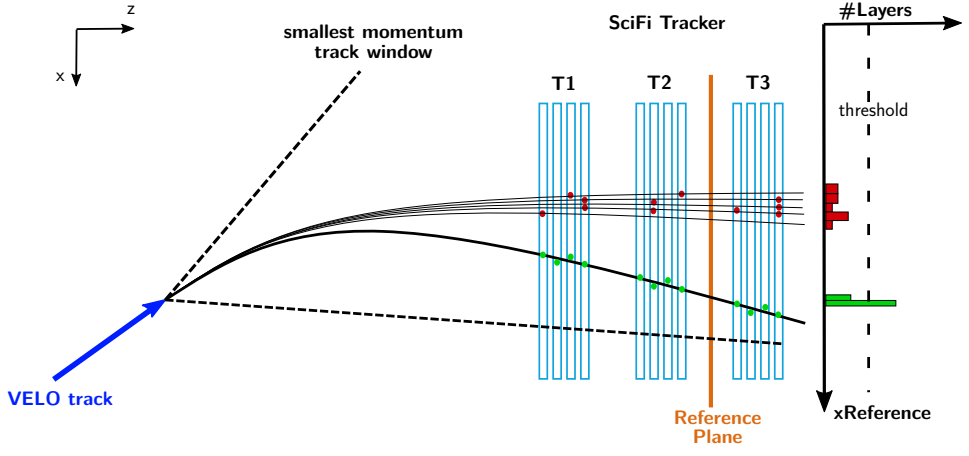
Forward Tracking - Algorithm

3. Hough-like transform: project all hits in window to reference plane and count number of SciFi layers in histogram



Forward Tracking - Algorithm

4. scan histogram, collect hits from bins above threshold



Forward Tracking - Algorithm Summary

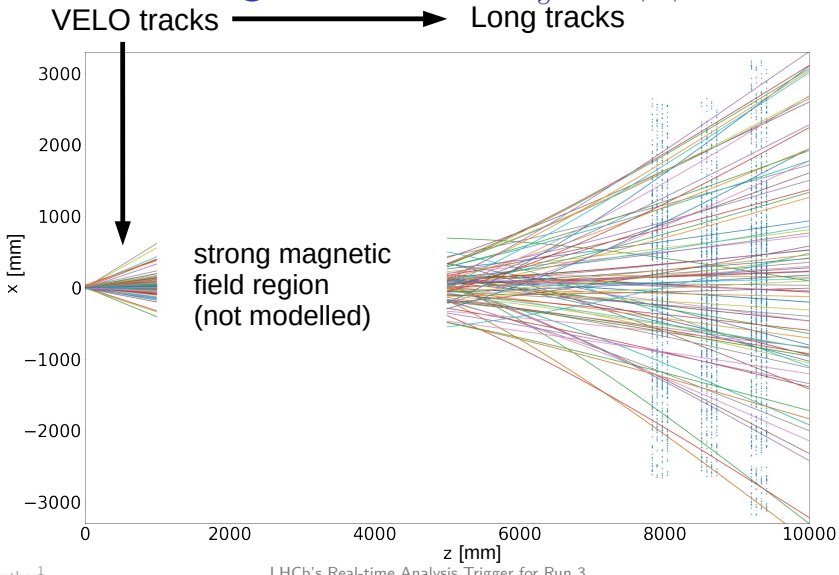
1. define hit search window
2. treat magnet as optical lens to simplify track and hit projection
3. Hough-like transform: project all hits in window to reference plane and count number of SciFi layers in histogram
4. scan histogram, collect hits from bins above threshold

$\rightarrow$ found set of SciFi hits extending VELO track

---

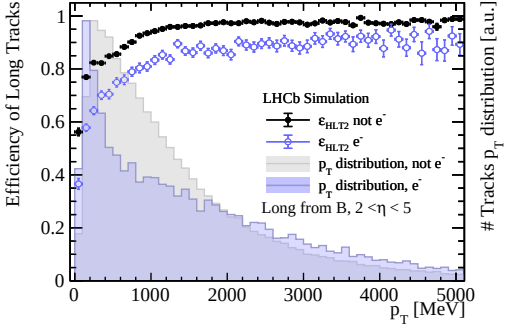
5. clean-up hit set and fit using 3rd order polynomial
6. estimate q/p from fit result

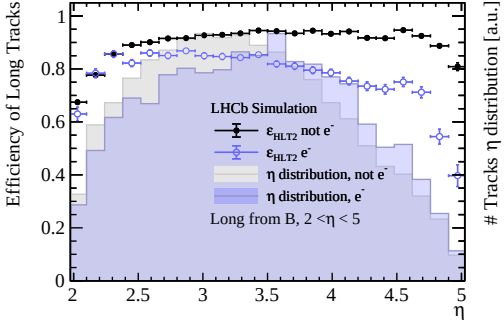
Reconstructed Long Tracks in $B_s^0 \to \phi\phi$ Event



Forward Tracking - Reconstruction Efficiency

- more than 95% efficiency for high momentum tracks from B meson



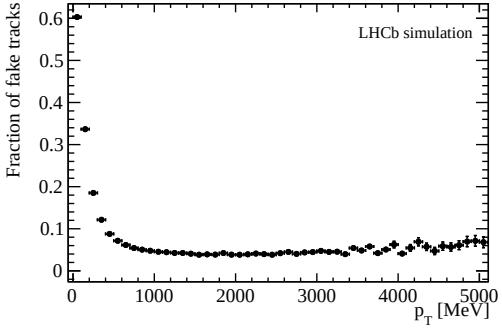


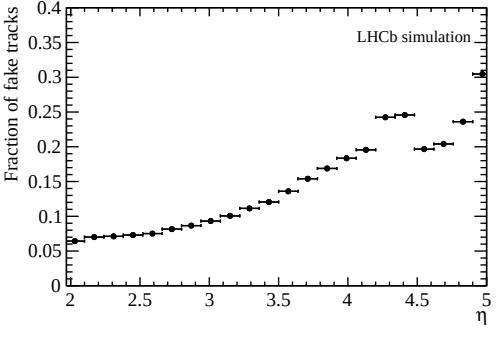
LHCb-FIGURE-2022-005

Forward Tracking - Fake Track Fraction

- high pseudorapidity region has high hit density

→ more combinatorics to form fake tracks





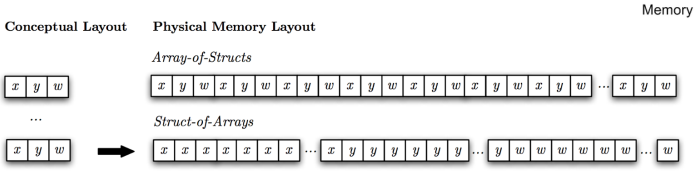
LHCb-FIGURE-2022-005

Forward Tracking - Event Throughput Optimisation

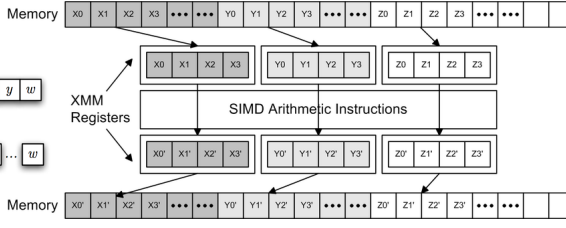
- typically trade-off between throughput, track reconstruction efficiency, fake track fraction

But: can improve throughput using modern CPU capabilities

→ Single Instruction Multiple Data (SIMD) with Structure-of-Arrays (SOA) data layout, "Vectorisation"



Arthur Hennequin. Performance optimization for the LHCb experiment



Ma, Wan-Chun, Yang, Chia-Lin. (2003) 10.1007/3-540-36228-2 134.

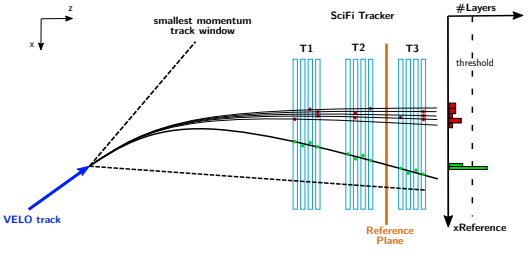
Forward Tracking - Vectorisation

- hits and #Layers stored in SOA layout → apply SIMD

→ project multiple hits

→ threshold scan multiple #Layers

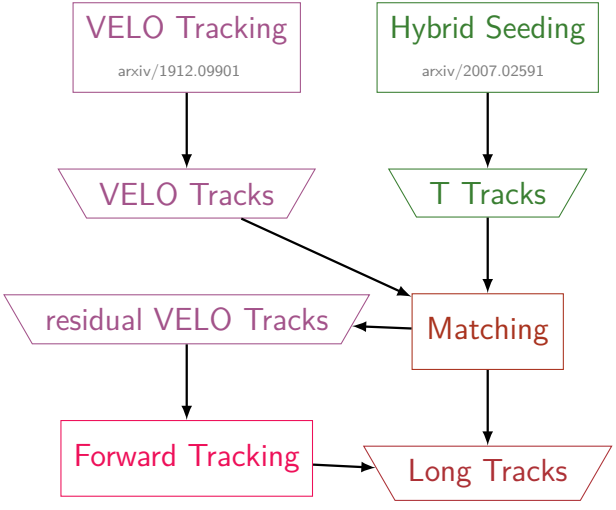
- 8 single precision floats/integers in parallel (AVX2)



→ increases event throughput of Forward Tracking by 60%

HLT2 Reconstruction Performance

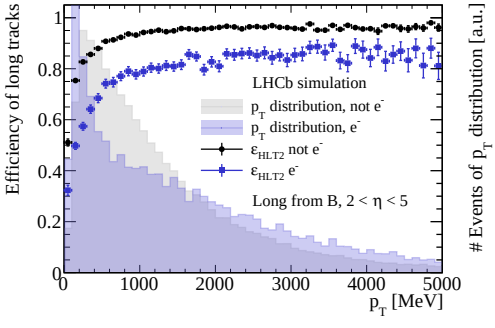
HLT2 Reconstruction Performance

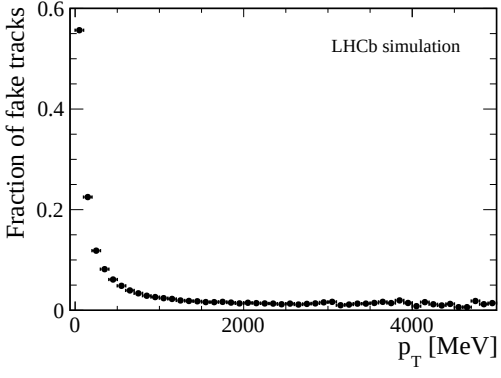


- removing redundancy loses $\sim$1% reconstruction efficiency
- "Best" Long Tracks found using Kalman Filter

HLT2 Reconstruction Performance

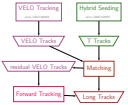
- "Best" Long Tracks for physics after Kalman Filter





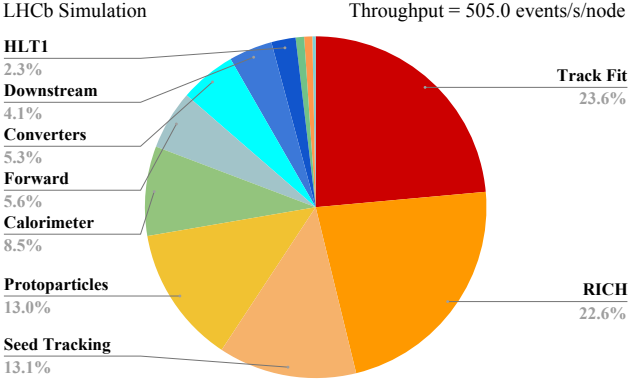
LHCb-FIGURE-2021-003

HLT2 Reconstruction Sequence Throughput



LHCb-FIGURE-2022-005

- throughput break down of track reconstruction and particle identification
- not including candidate selections



→ Forward Tracking not dominating component for throughput

→ **milestone of 500 Events/s/node reached**

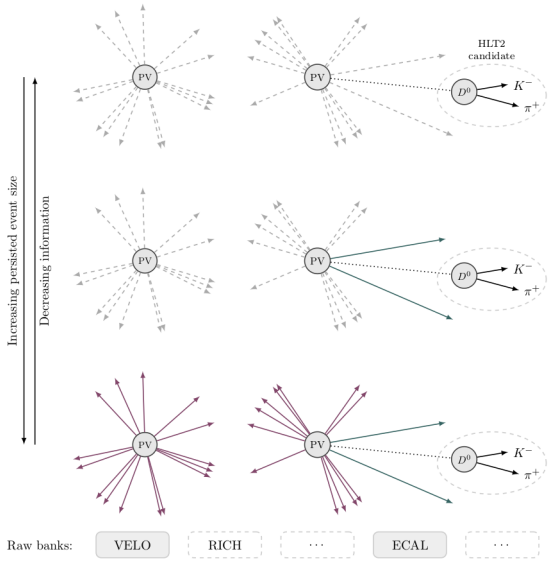
Summary

- LHCb enters MHz signal era → need for software trigger
- HLT1 partial event reconstruction accelerated using GPUs
  → pioneering software reducing of 30 MHz raw collision data
- HLT2 full event reconstruction reached milestone 500 Events/s/node
  - Forward Tracking algorithm as performance example

→ HLT1 and HLT2 event reconstruction ready for data taking!

Backup

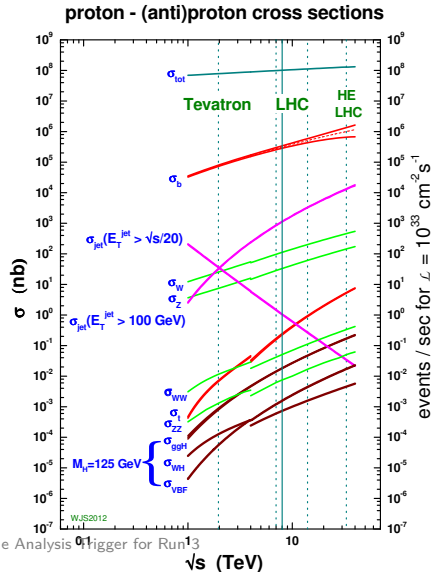
Turbo Persistence Model



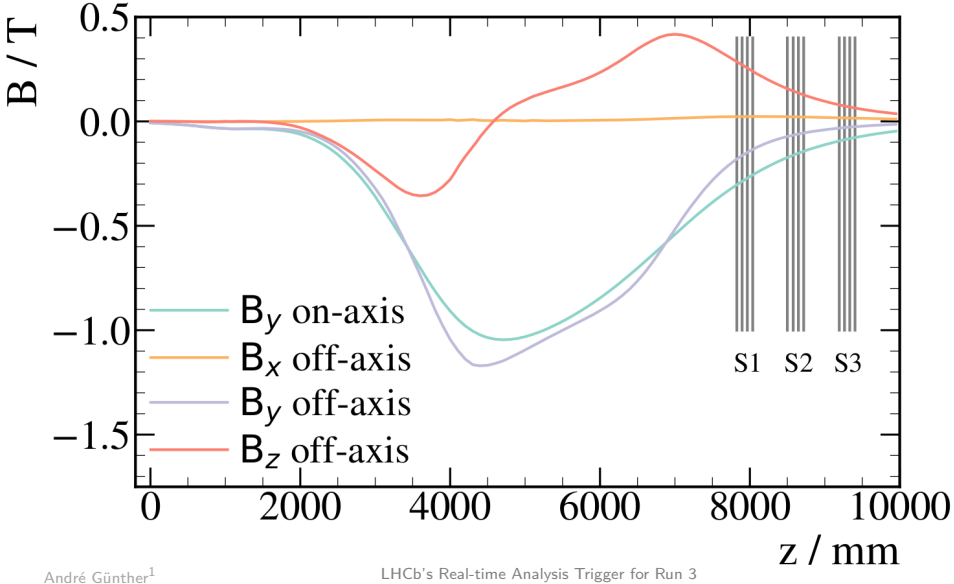
Cross sections at LHC(b)

- large b and c quark production cross section

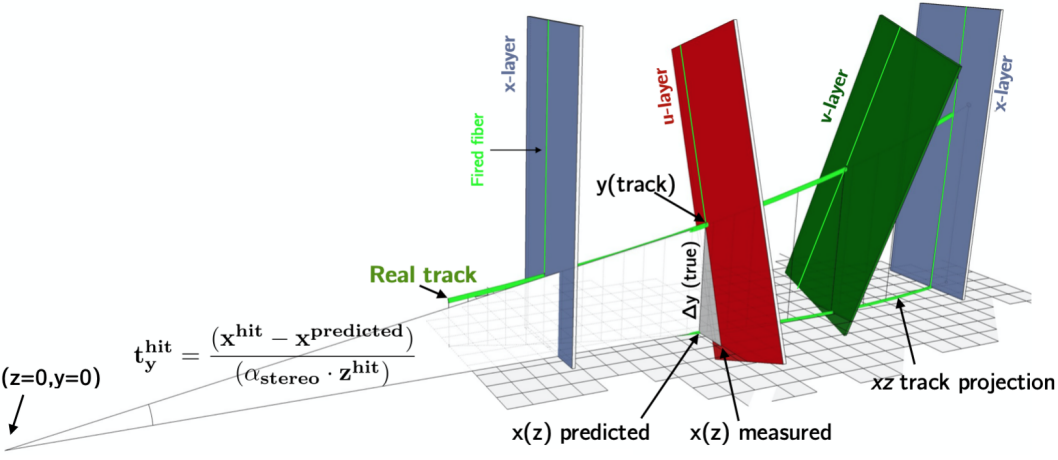
$\rightarrow$ many pp collisions potentially interesting for LHCb!



Magnetic Field Components



Trajectory Stereo Layers

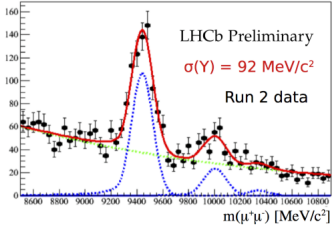


Real-time Alignment & Calibration

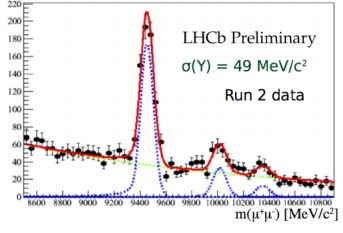
- Efficient and pure selections require offline-quality reconstruction at the HLT2 level ⟵ Aligned and calibrated detector
- Use output bandwidth more efficiently
  - Better mass resolution
  - Better particle identification
  - Less background

Journal of Physics:
Conference Series, 664 (2015)

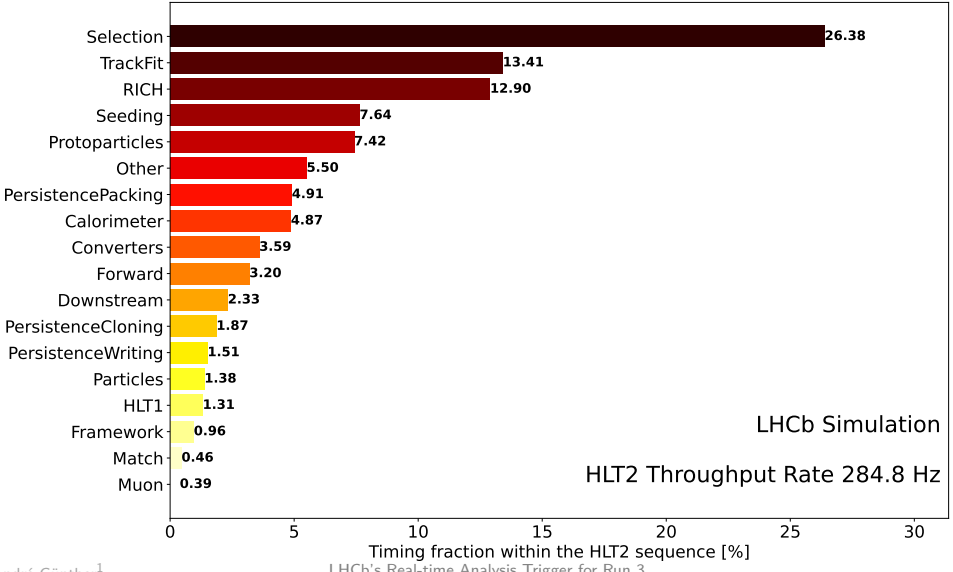
Before alignment



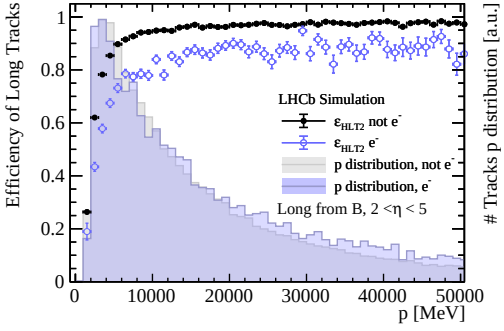
After alignment

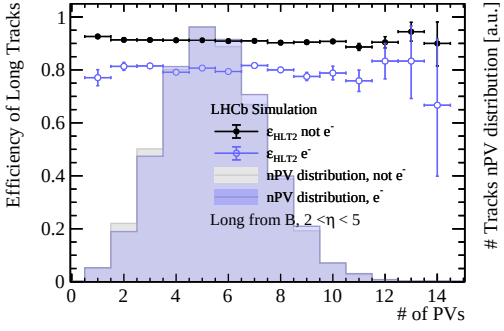


HLT2 Throughput Performance with Selections



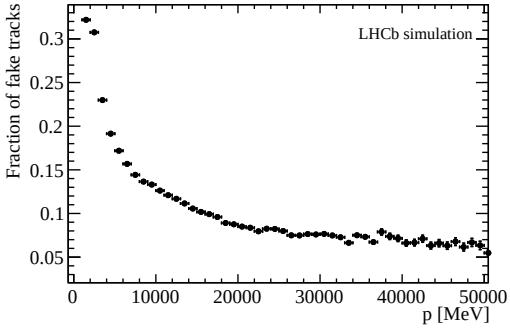
Forward Tracking - Reconstruction Efficiency

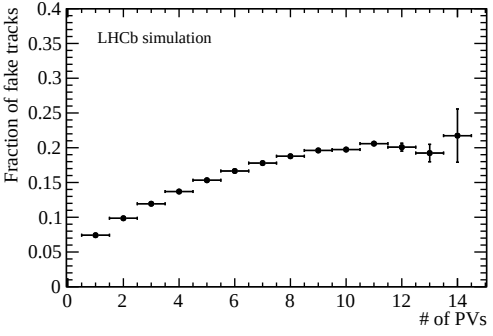




LHCb-FIGURE-2022-005

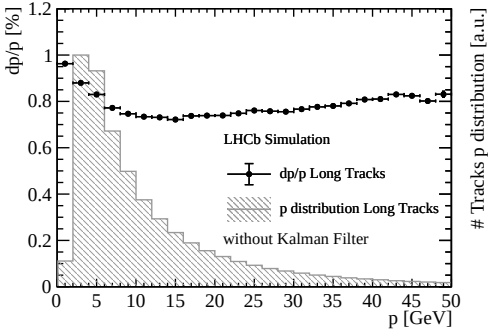
Forward Tracking - Fake Track Fraction

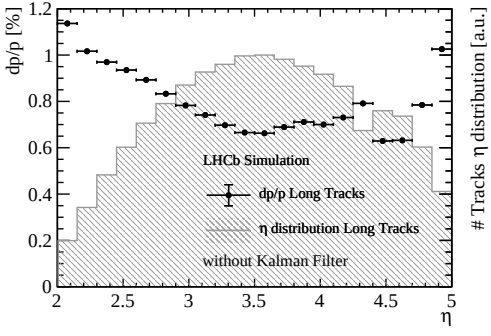




LHCb-FIGURE-2022-005

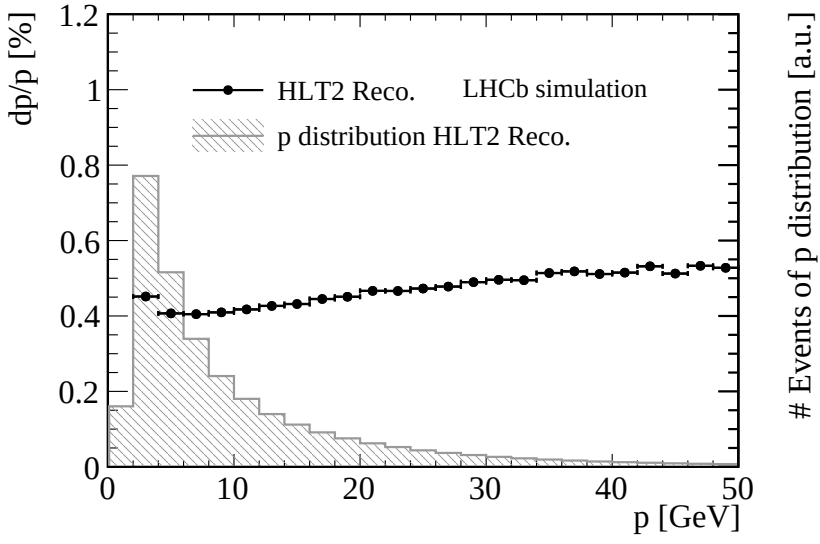
Forward Tracking - Momentum Resolution





LHCb-FIGURE-2022-005

HLT2 Momentum Resolution

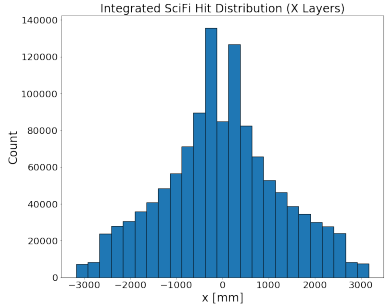


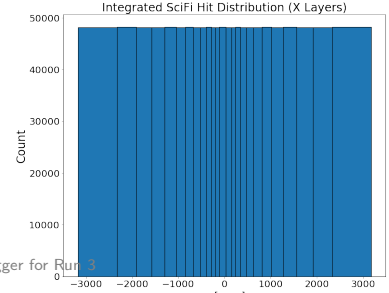
Hough Histogram Binning

→ choose binning such that distribution is uniform

→ i-th bin edge x_Q^i defined by (quantiles):

$$F(x_Q^i) = \int_{-\infty}^{x_Q^i} f(t)\mathrm{d}t = \frac{i}{n_{bins}}$$





Hough Histogram - Hit Position to Bin Mapping

- calculate bin for each projected hit, i.e. non-linear mapping $x \mapsto i$
- function should be fast to evaluate and (easily) invertible
- possible choice:

$$i(x) = p_0 + \frac{p_1 \cdot x}{1 + |p_2 \cdot x|}$$

- with $n_{bins} = 1000$ gives widths $\mathrm{w}_{\mathrm{min}} = 1.625\,\mathrm{mm}$ and $\mathrm{w}_{\mathrm{max}} = 45\,\mathrm{mm}$