Track reconstruction and performance at LHCb

Peilian Li (李佩莲)

Workshop of Tracking in Particle Physics Experiments

Zhengzhou, 2024-05-17

University of Chinese Academy of Sciences

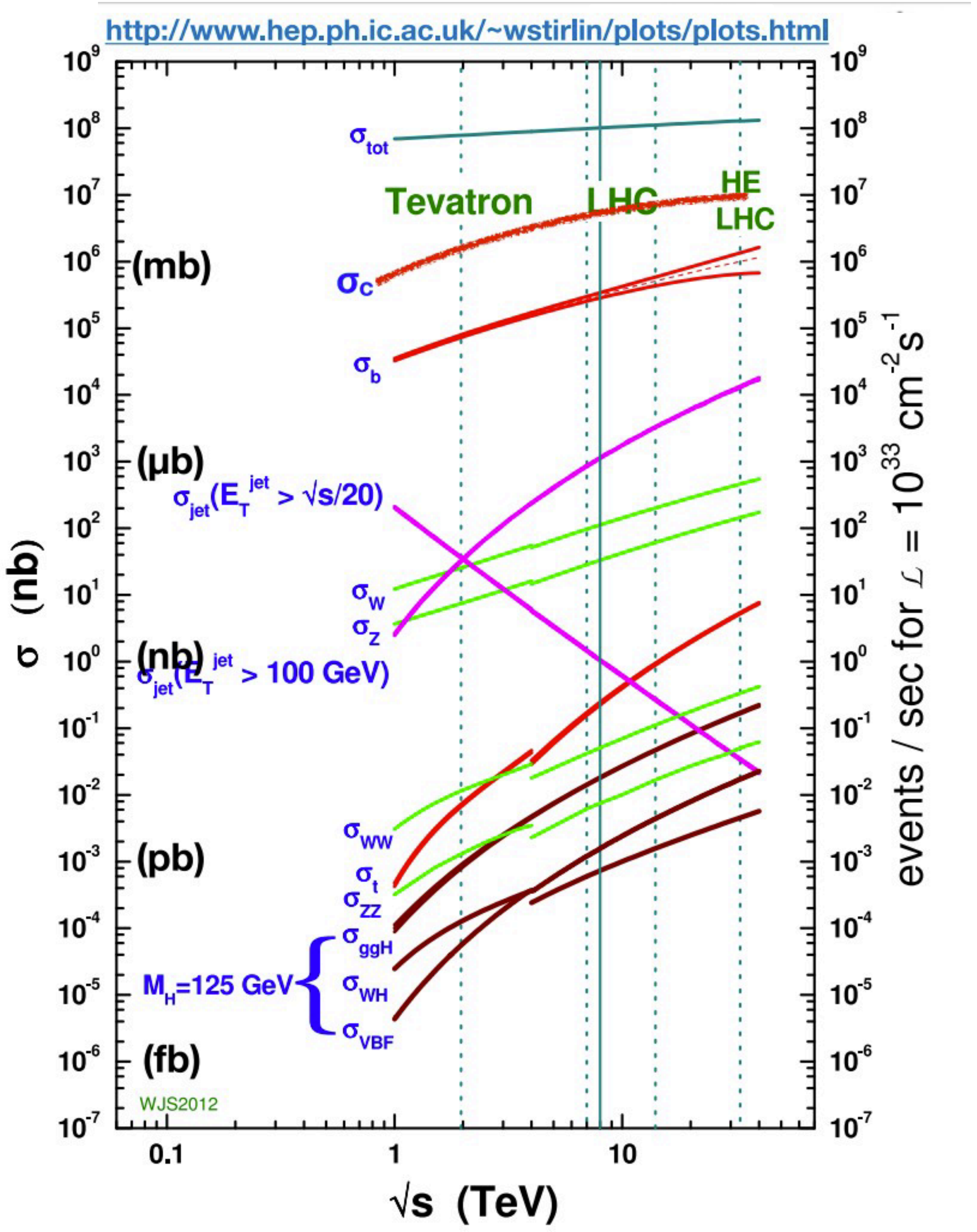
Outline

- Overview of the trigger at LHCb
- Track Reconstruction with GPU
- Track Reconstruction with CPU
- Clustering & Tracking with FPGA
- Summary

Many materials from D. Vom Bruch, V. Gligorov etc, thanks!

LHCb Upgrade

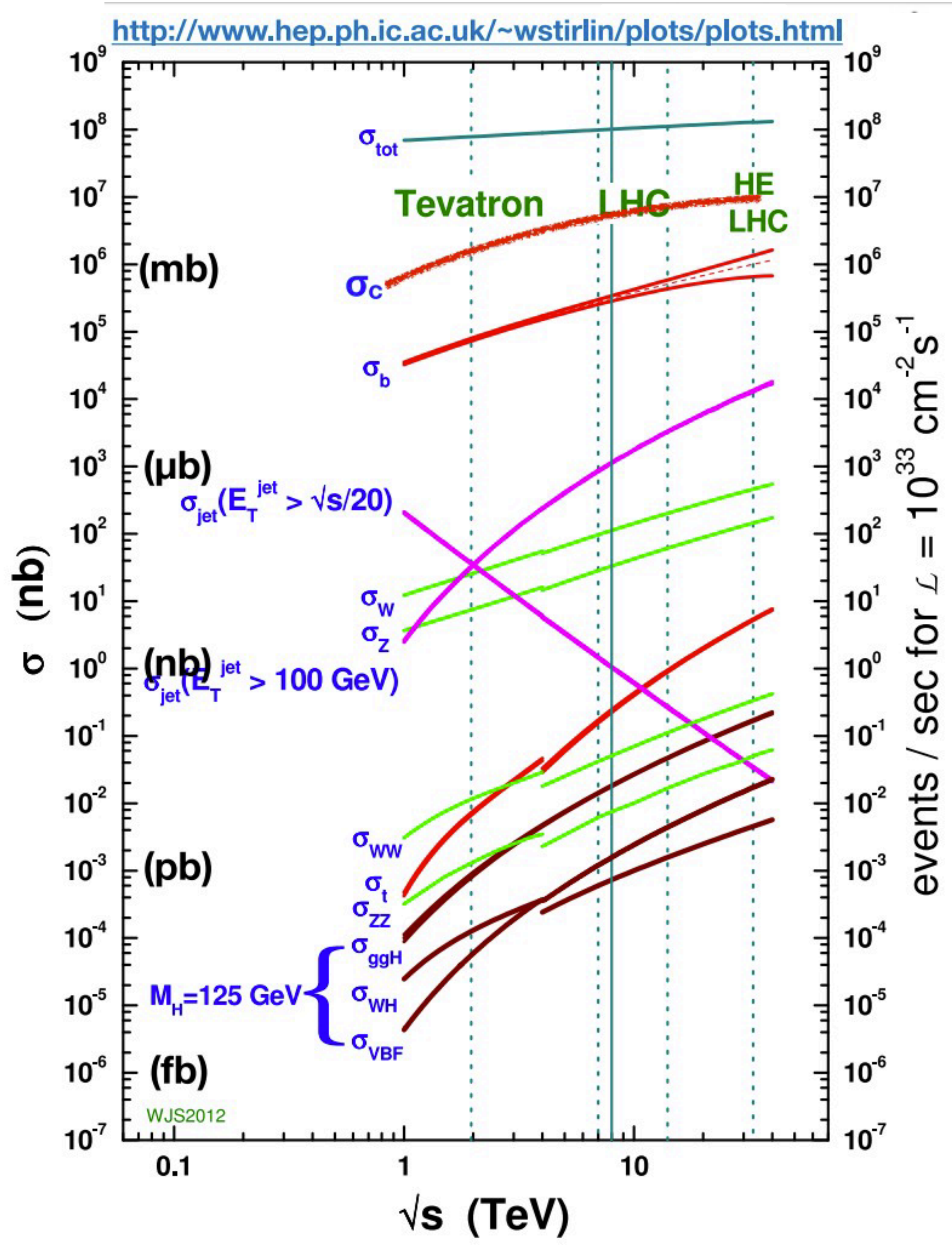
- Luminosity of 2x10^{33} cm^{-2}s^{-1}, $\sqrt{s}$ = 14 TeV, visible collisions per bunch $\mu \sim 5$



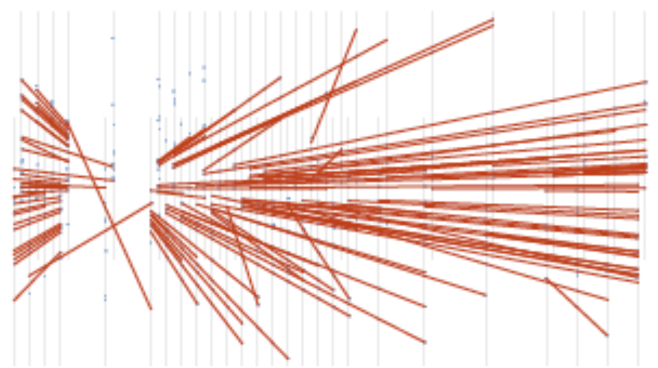
39

LHCb Upgrade

- Luminosity of 2×10^{33} cm^{-2}s^{-1}, $\sqrt{s}$ = 14 TeV, visible collisions per bunch $\mu \sim 5$



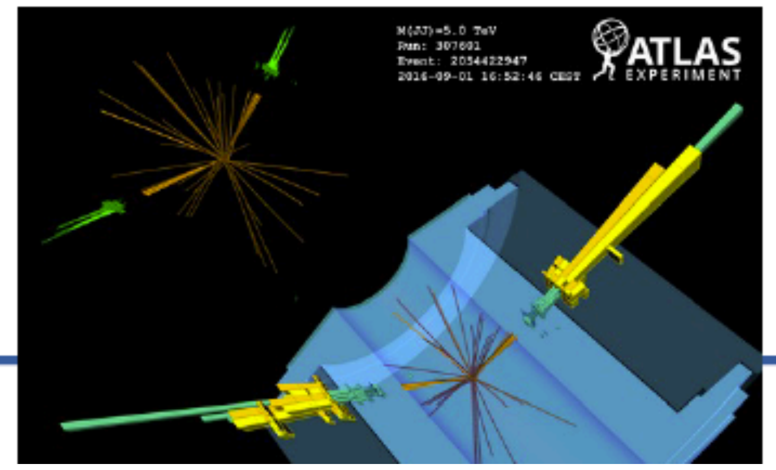
LHCb: Mainly beauty and charm physics



- Signal rates at MHz level
- Signal characteristics: Displaced vertices, momentum, particle type
- → No optimal local criteria for selection

ATLAS & CMS: Mainly Higgs properties, high p_T new phenomena

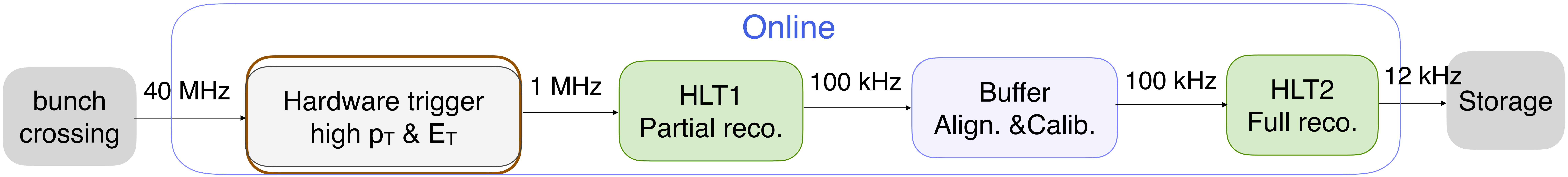
- Signal rates up to hundreds of kHz
- Signal characteristics: high pT / transverse energy
- → Local criteria for selection possible



Challenges for the Upgrade

Hardware trigger: 40→1 MHz read-out limits (fixed-latency trigger)

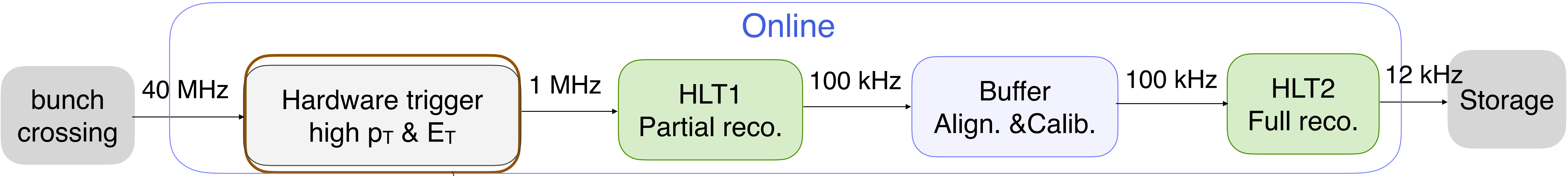
→ based on muon detector and calorimeters



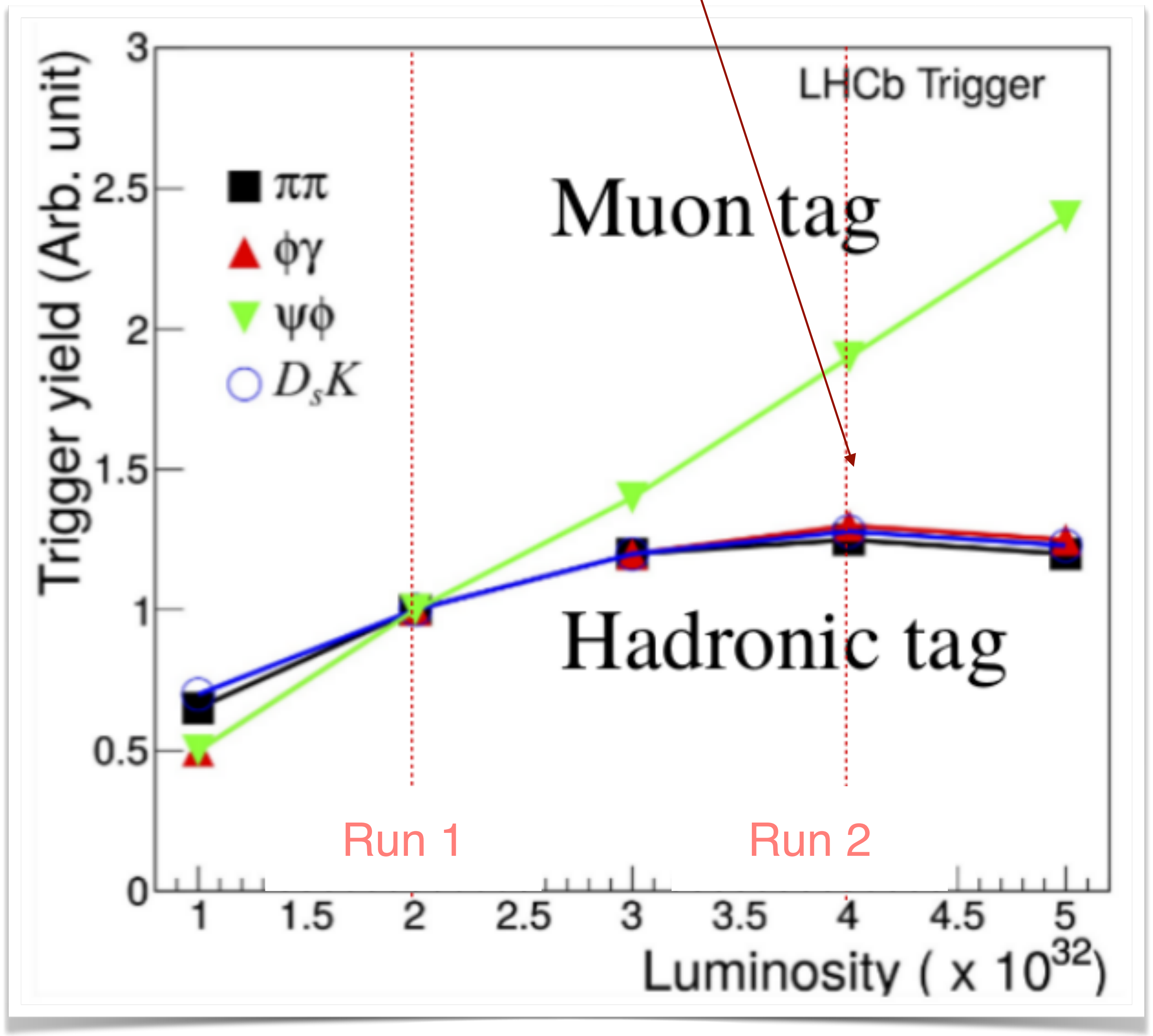
Challenges for the Upgrade

Hardware trigger: 40→1 MHz read-out limits (fixed-latency trigger)

→ based on muon detector and calorimeters

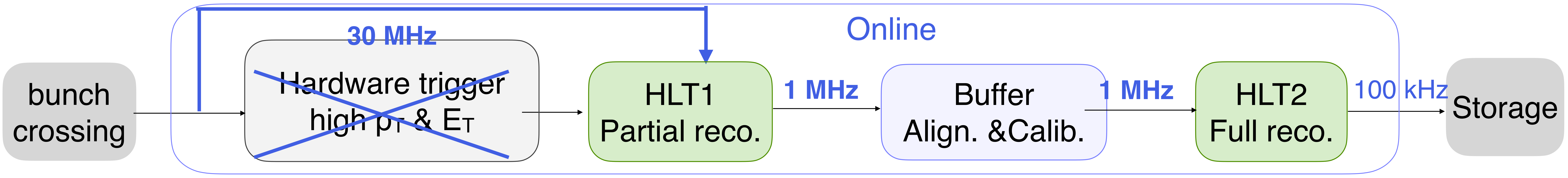


Conf. Series 878(2017)012012



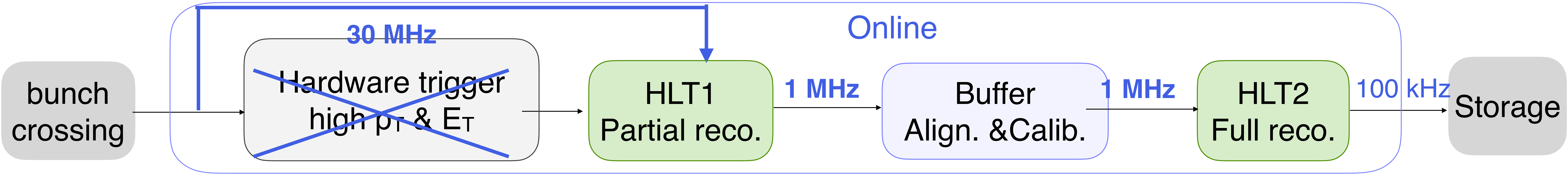
- Hardware trigger is not an option, as rate limit of 1 MHz saturates fully hadronic modes

LHCb Upgrade Trigger

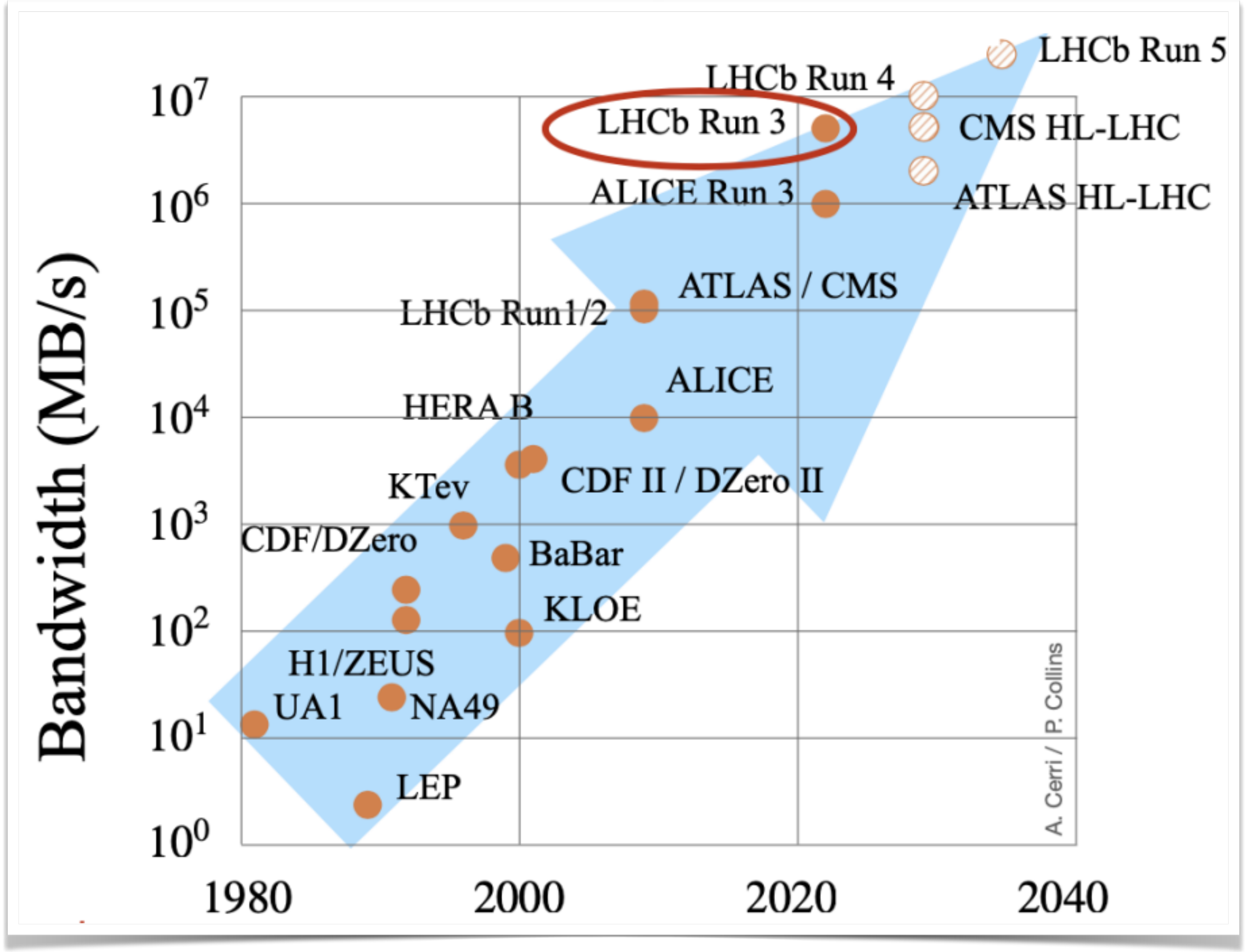


- Remove hardware trigger, fully software trigger
- Read out the full detector at 30 MHz in HLT1
- Real time alignment and calibration with 10x higher data rate than Run 2
- Full offline-quality reconstruction in "real-time"
- Increase of hadronic trigger efficiency by 2~4 w.r.t. Run 2

LHCb Upgrade Trigger

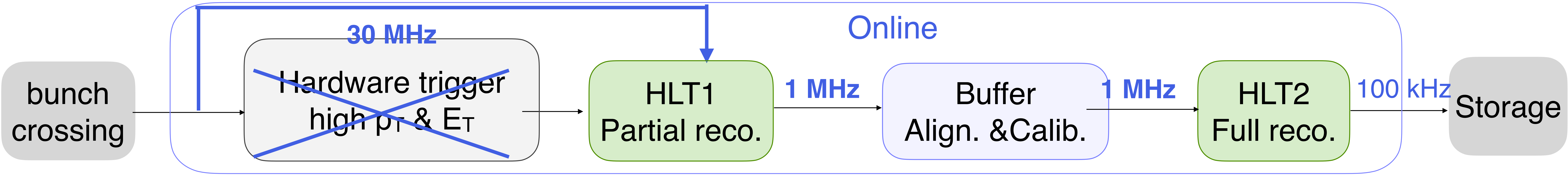


- Remove hardware trigger, fully software trigger
- Read out the full detector at 30 MHz in HLT1
- Real time alignment and calibration with 10x higher data rate than Run 2
- Full offline-quality reconstruction in “real-time”
- Increase of hadronic trigger efficiency by 2~4 w.r.t. Run 2



Highest data processing rate of any HEP experiment!

LHCb Upgrade Trigger



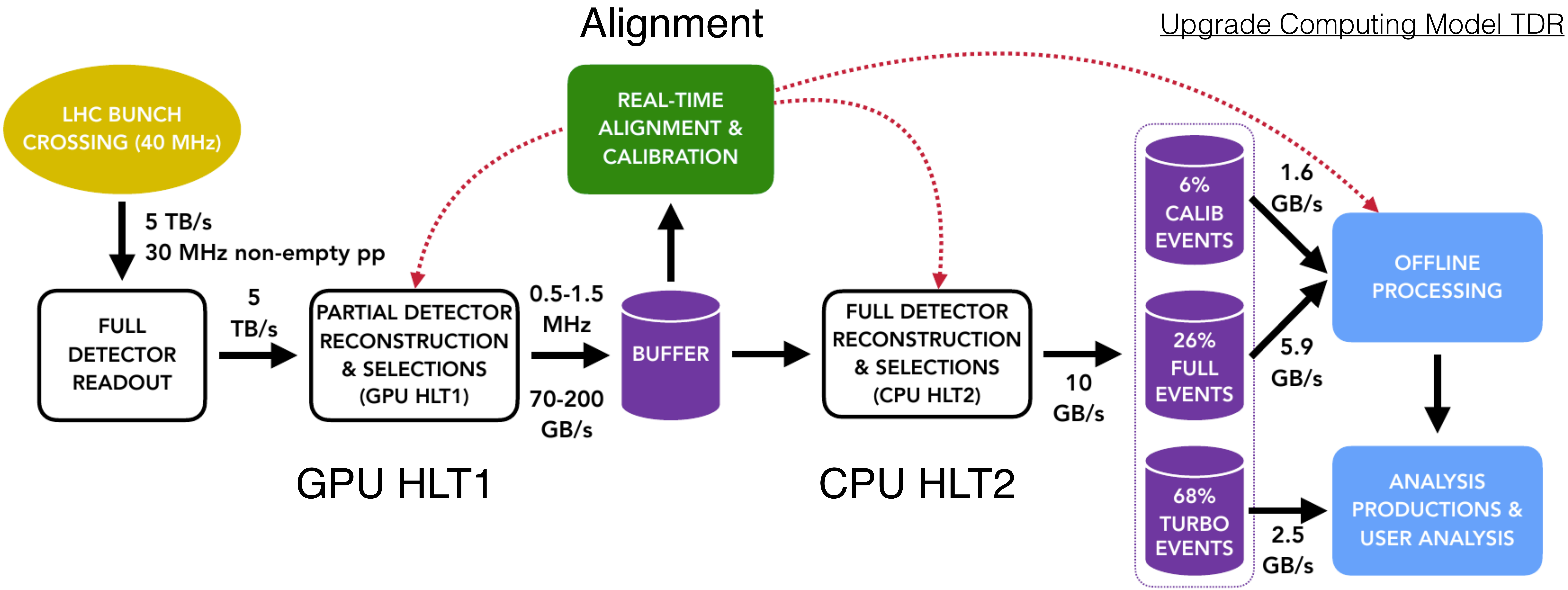
Online - Real Time Analysis

Run 1 & 2 trigger:
background rejection

Upgrade trigger:
background rejection & signals classification

LHCb Data Flow

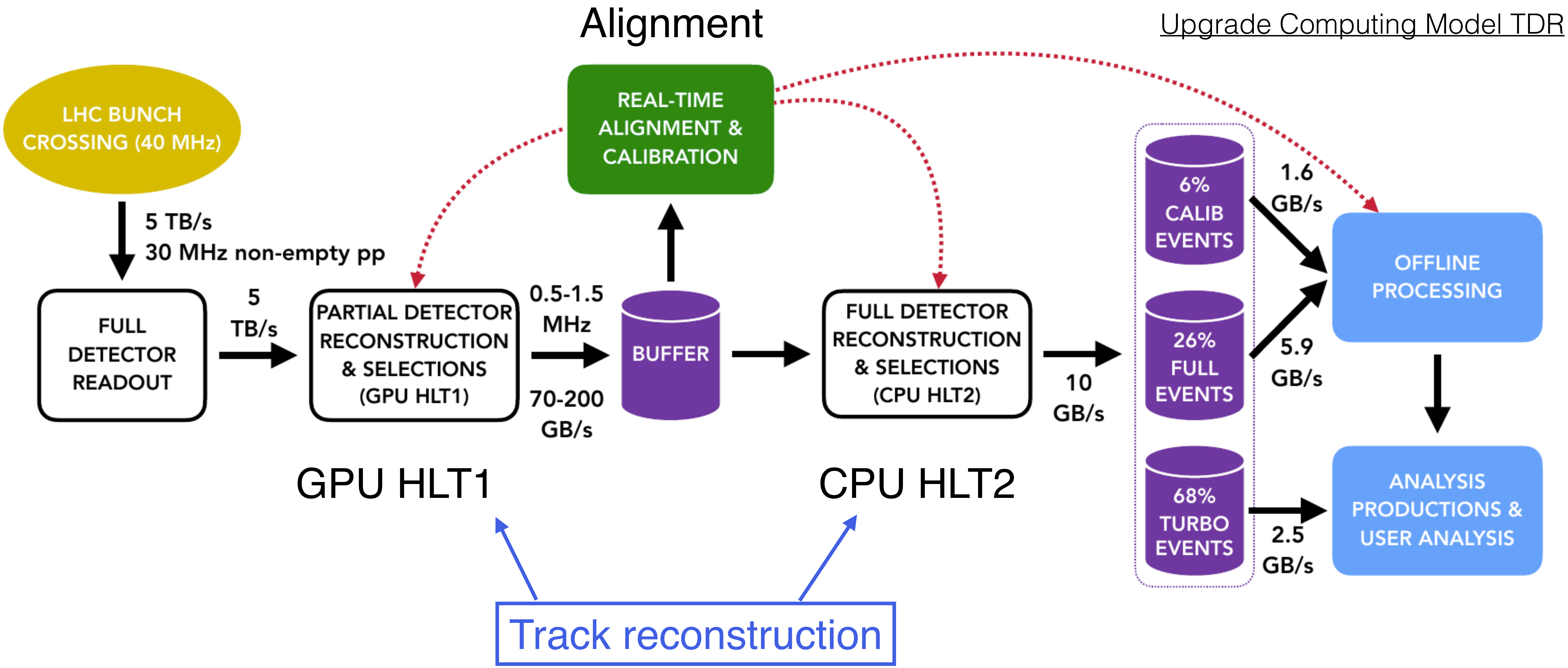
All numbers related to the dataflow are taken from the LHCb
Upgrade Trigger and Online TDR
Upgrade Computing Model TDR



First complete high-throughput GPU Trigger for a HEP experiment!

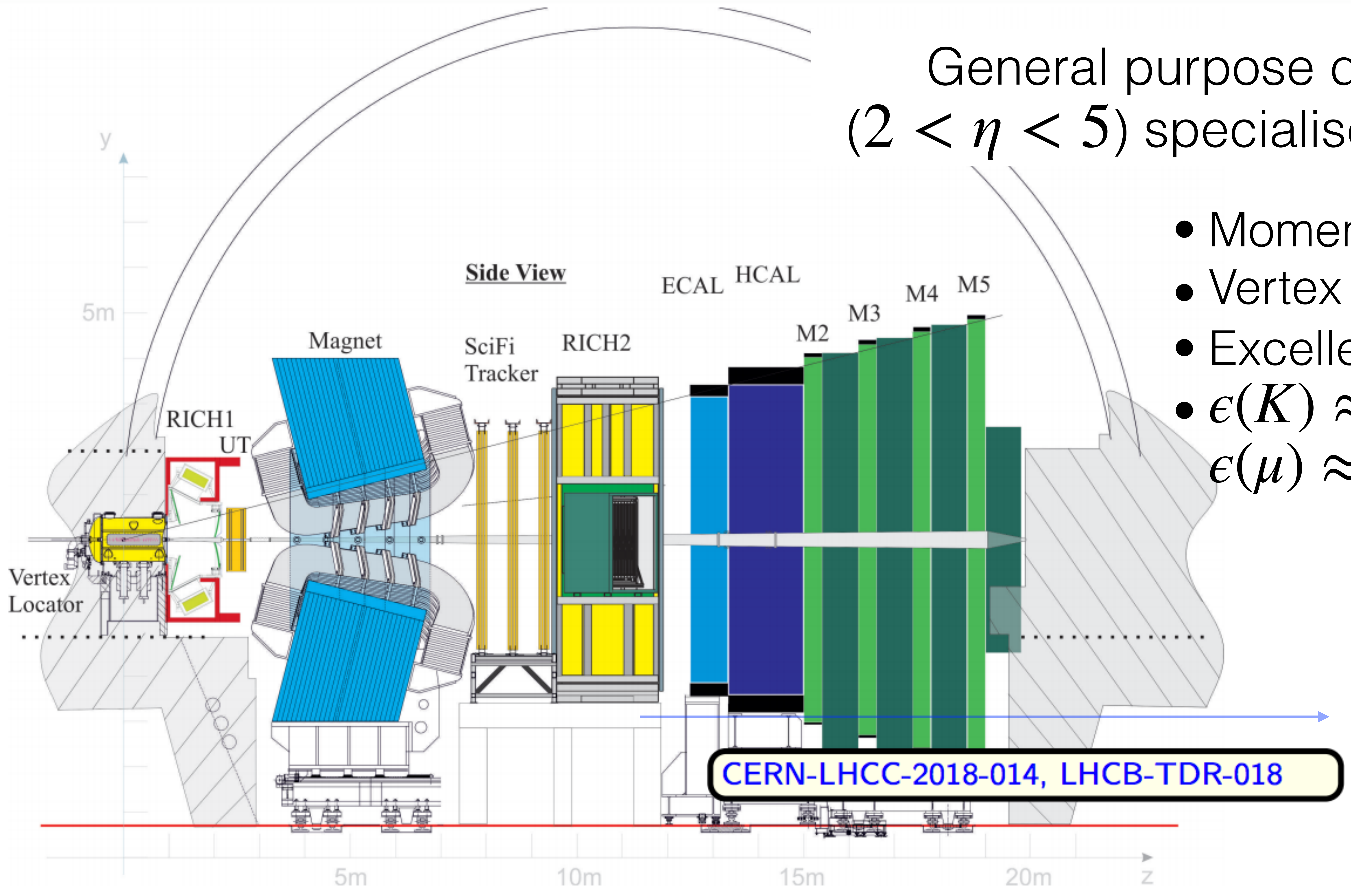
LHCb Data Flow

All numbers related to the dataflow are taken from the LHCb
Upgrade Trigger and Online TDR
Upgrade Computing Model TDR



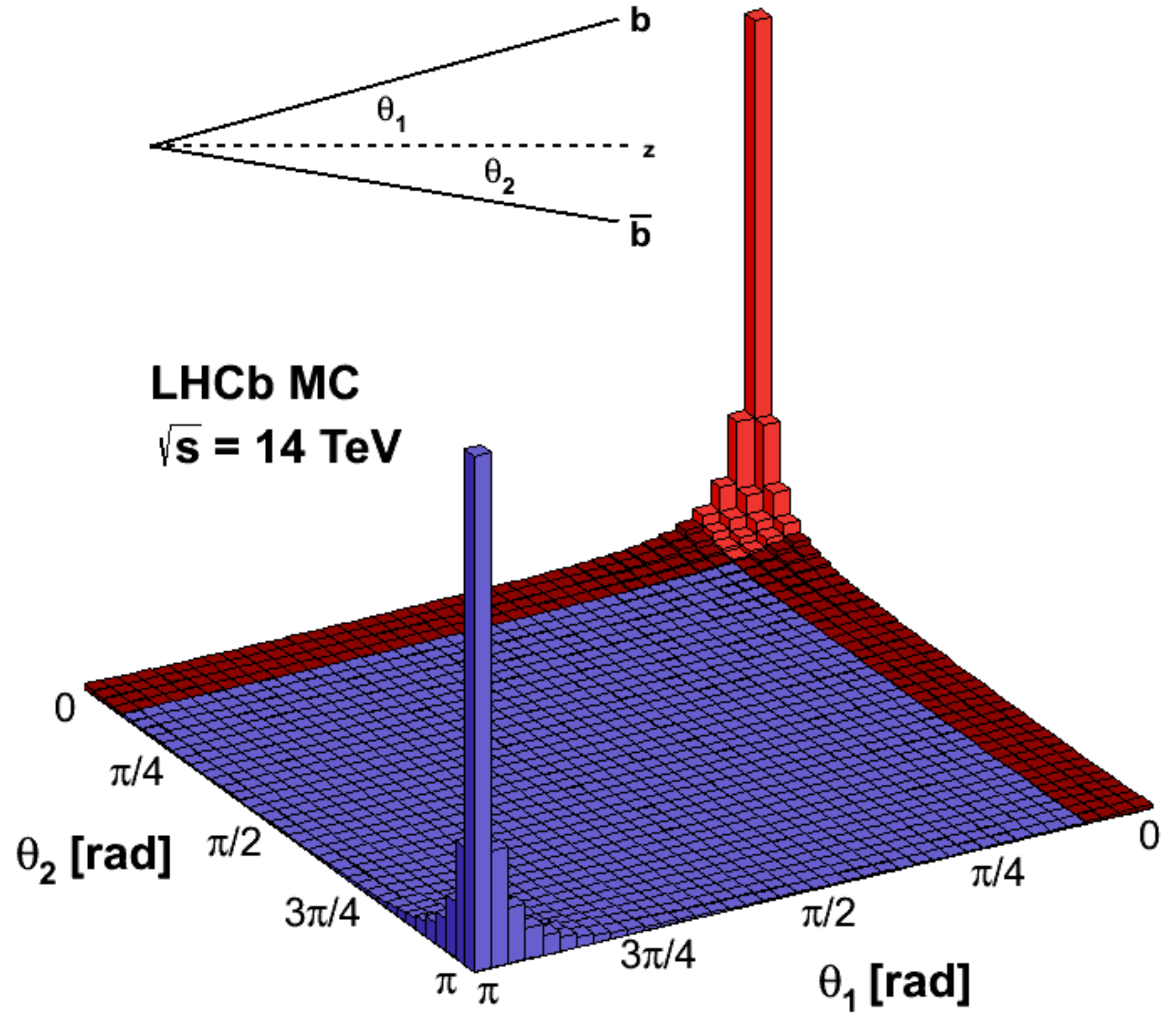
First complete high-throughput GPU Trigger for a HEP experiment!

LHCb Detector

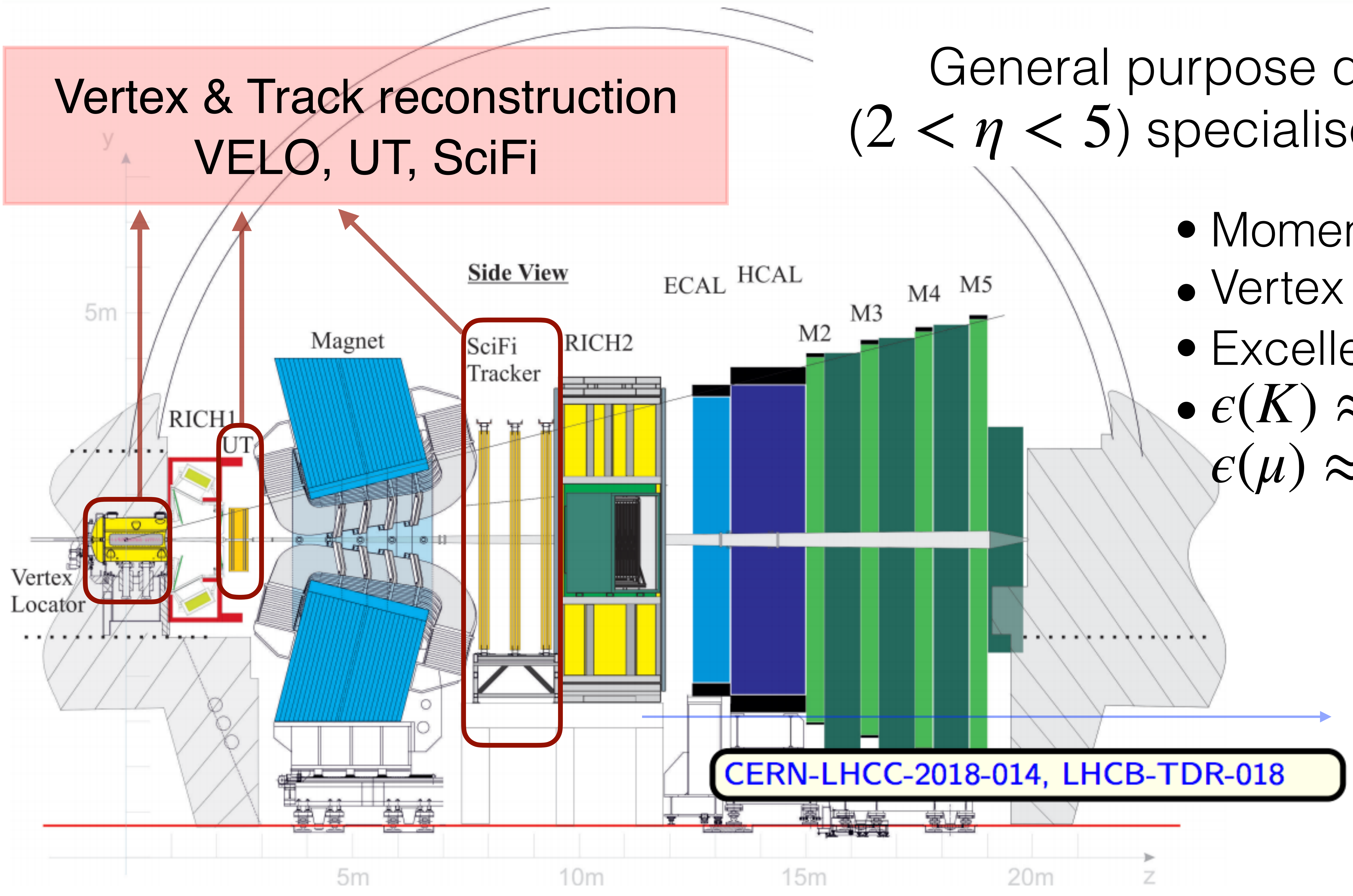


General purpose detector in the forward region $(2 < \eta < 5)$ specialised in beauty and charm physics

- Momentum resolution: 0.5%~1%
- Vertex resolution: $\sigma_{IP} \sim 35\mu m$
- Excellent particle identification
- $\epsilon(K) \approx 95\,\%$, misID $p(\pi \to K) \approx 5\,\%$
  $\epsilon(\mu) \approx 97\,\%$

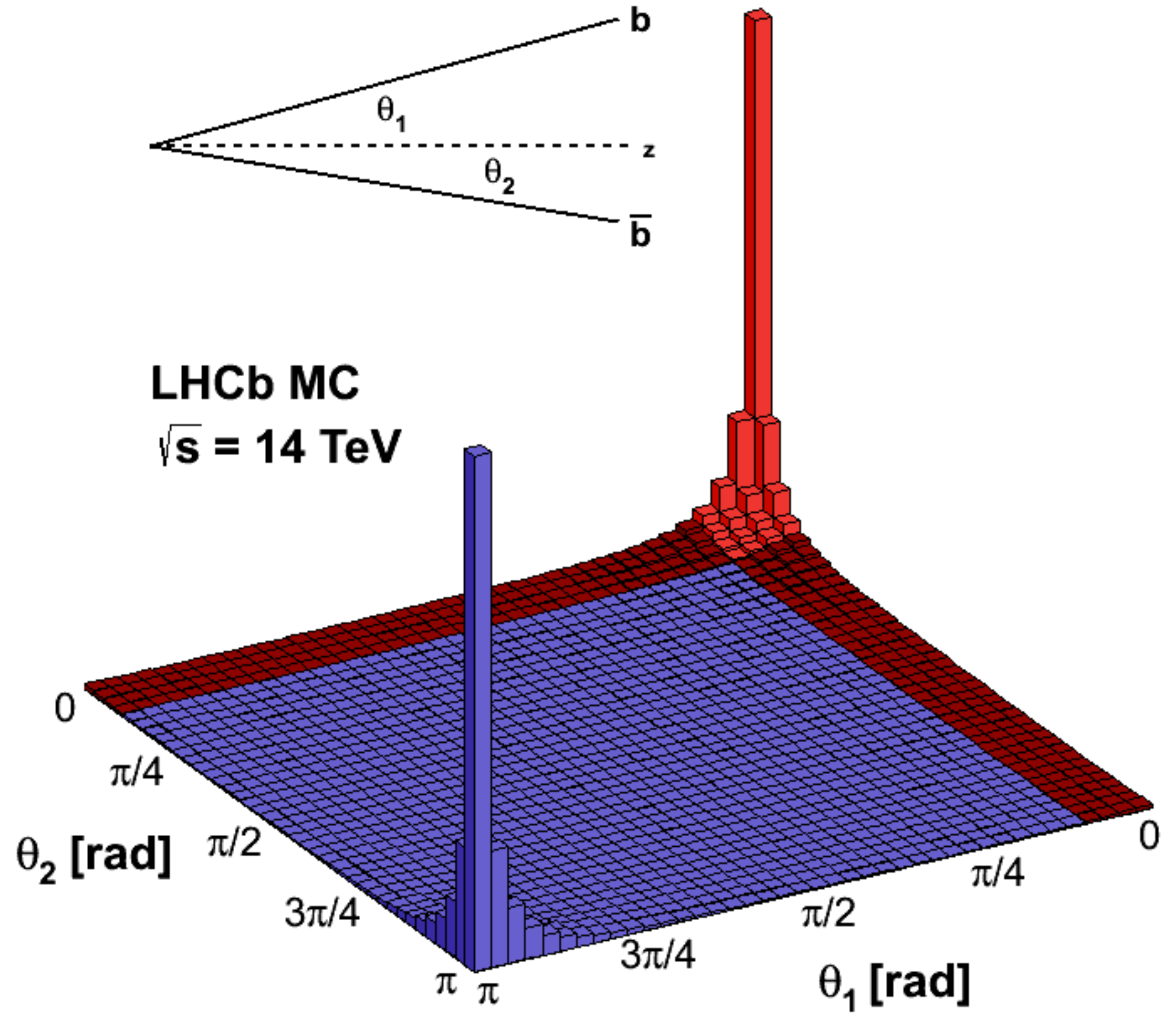


LHCb Detector



General purpose detector in the forward region ($2 < \eta < 5$) specialised in beauty and charm physics

- Momentum resolution: 0.5%~1%
- Vertex resolution: $\sigma_{IP} \sim 35\mu m$
- Excellent particle identification
- $\epsilon(K) \approx 95\,\%$, misID $p(\pi \to K) \approx 5\,\%$
$\epsilon(\mu) \approx 97\,\%$

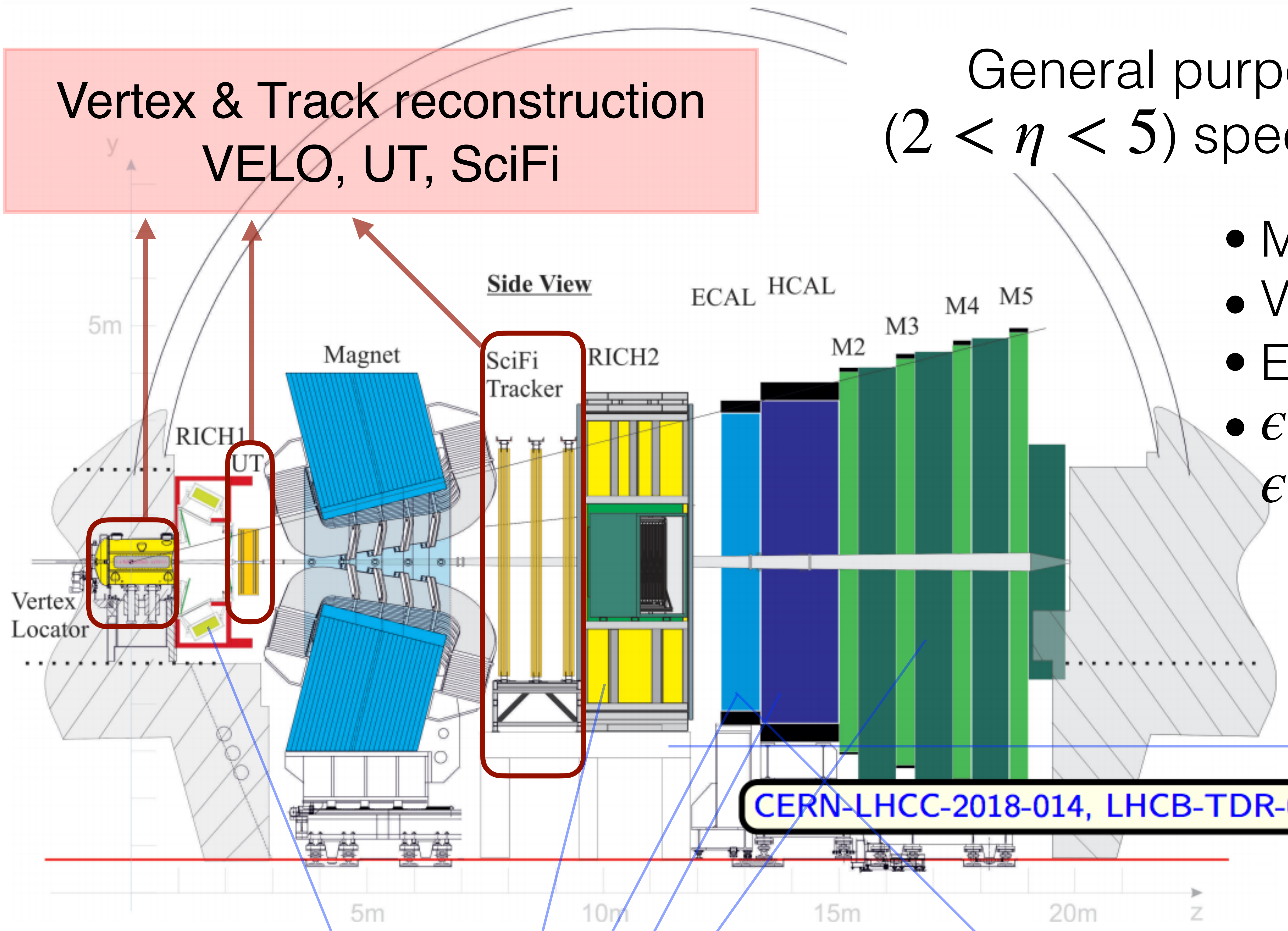


LHCb Detector

General purpose detector in the forward region
$(2 < \eta < 5)$ specialised in beauty and charm physics

- Momentum resolution: 0.5%~1%
- Vertex resolution: $\sigma_{IP} \sim 35\mu m$
- Excellent particle identification
- $\epsilon(K) \approx 95\,\%$, misID $p(\pi \to K) \approx 5\,\%$
$\epsilon(\mu) \approx 97\,\%$

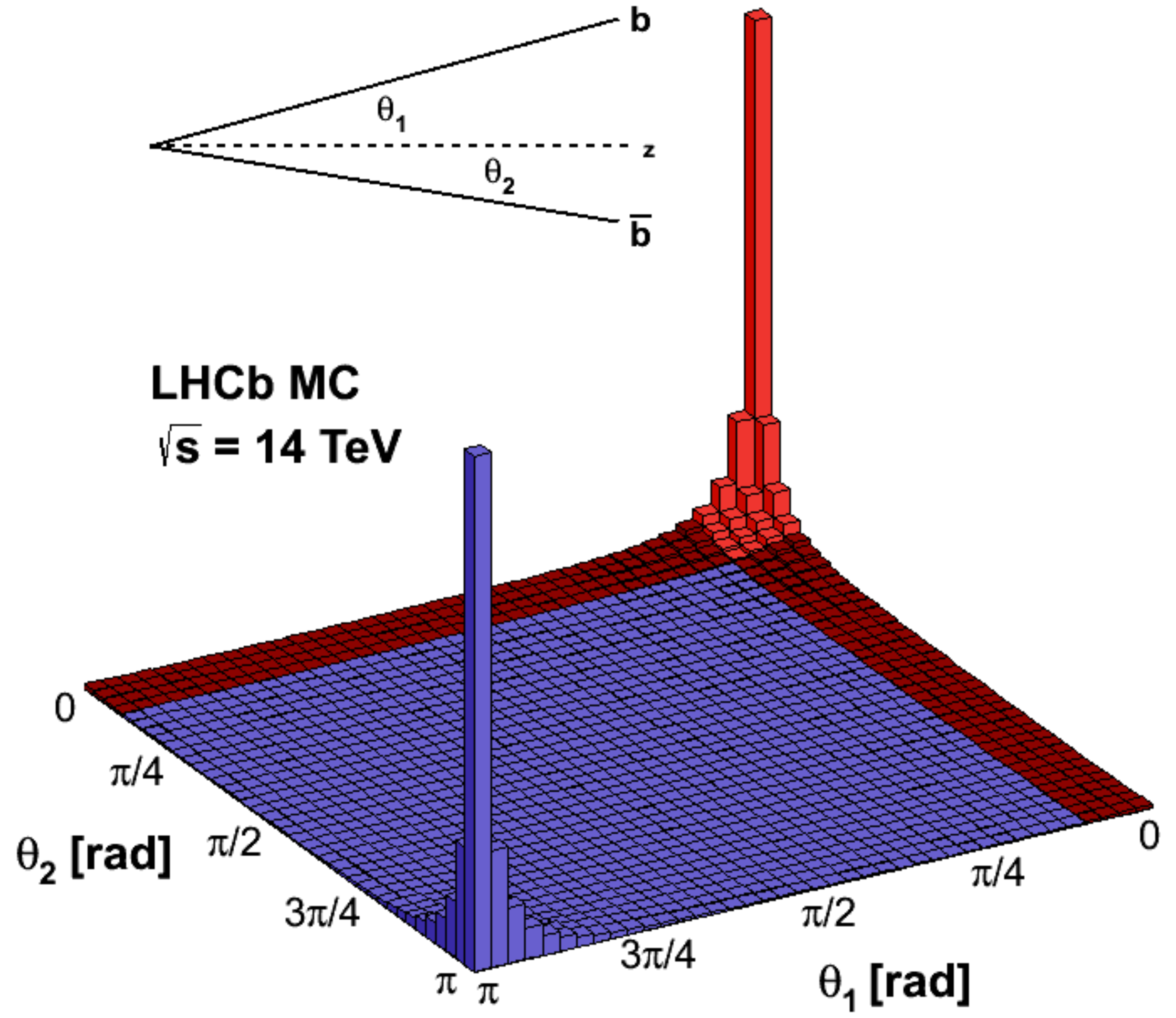
Vertex & Track reconstruction
VELO, UT, SciFi



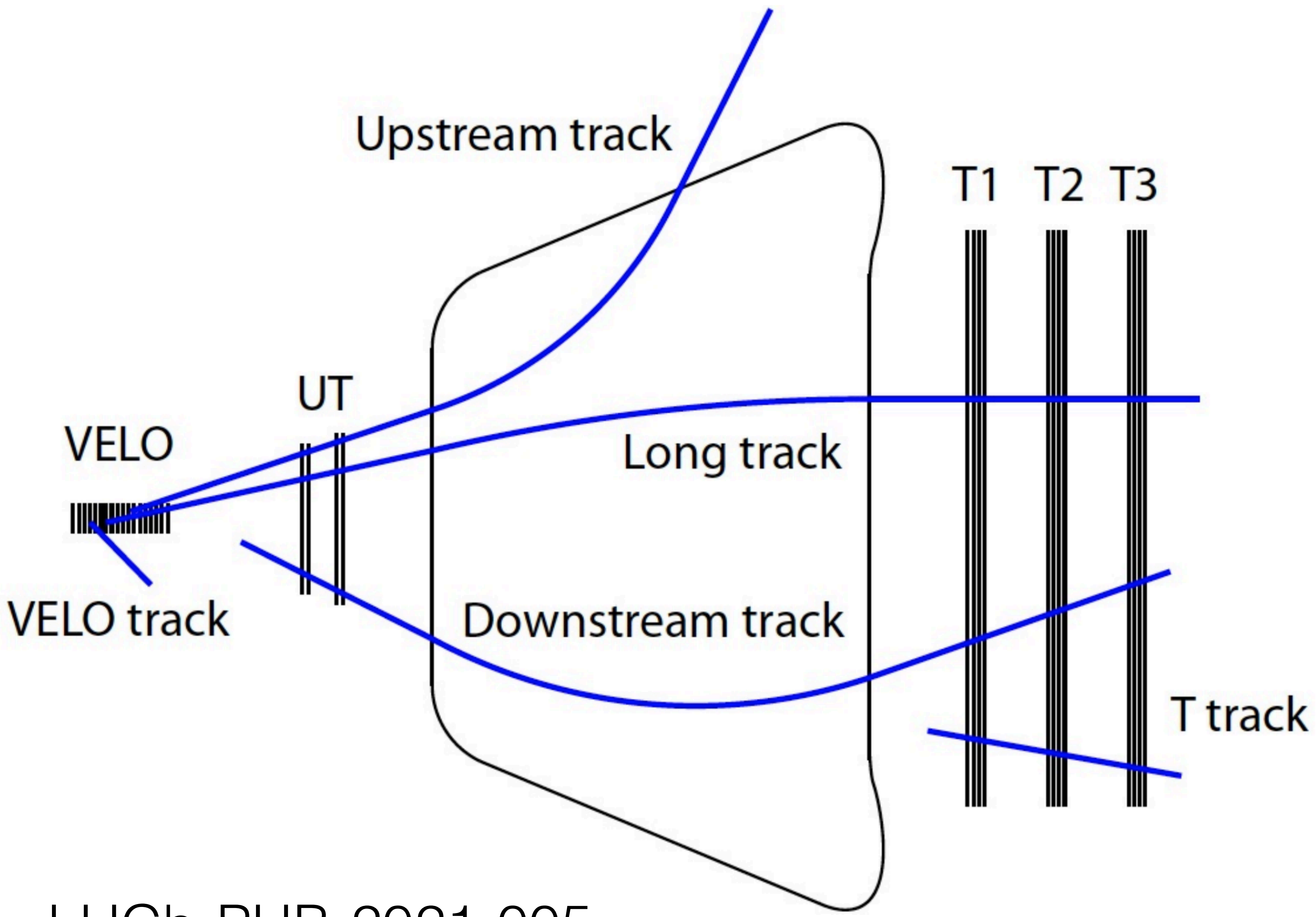
CERN-LHCC-2018-014, LHCB-TDR-018

Particle identification:
RICH, MUON, ECAL

Neutral reconstruction:
ECAL



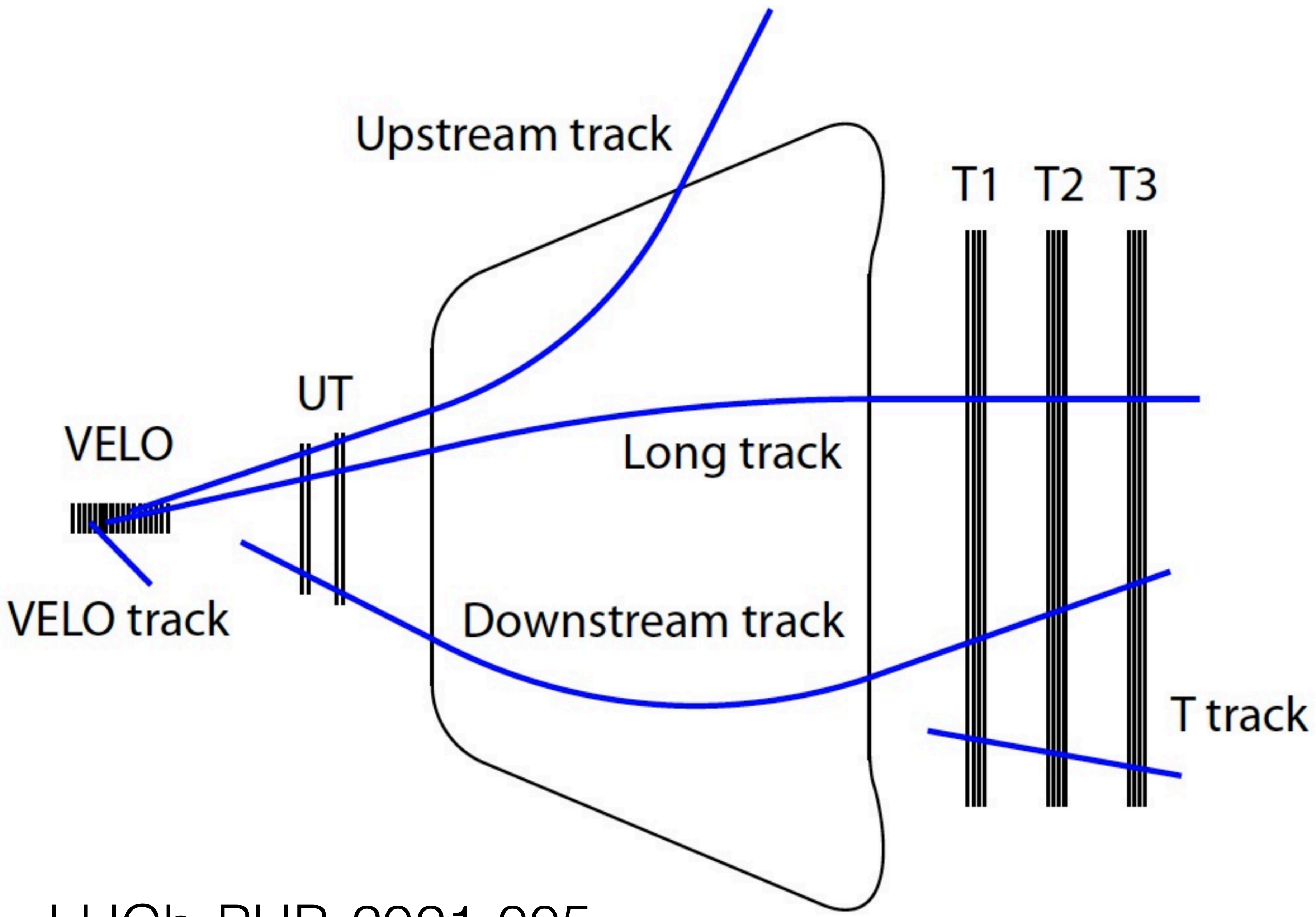
Track Types



LHCb-PUB-2021-005

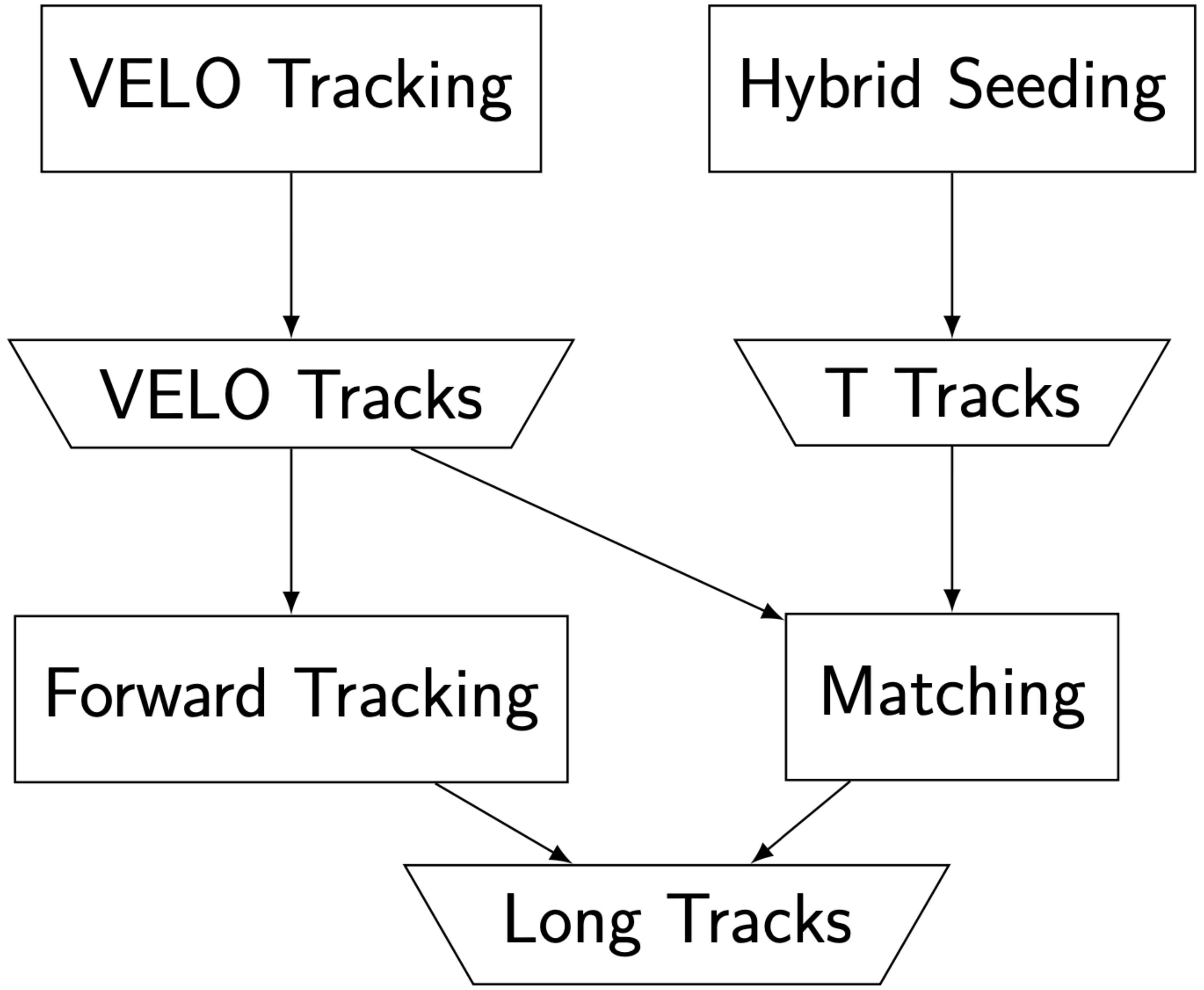
- Long tracks: best resolution
  - **Forward Tracking:** VELO (→ UT) →SciFi
  - **Matching:** VELO Tracks + T tracks + (UT)
- Downstream tracks for long-lived particles
  - T tracks + UT hits

Track Types

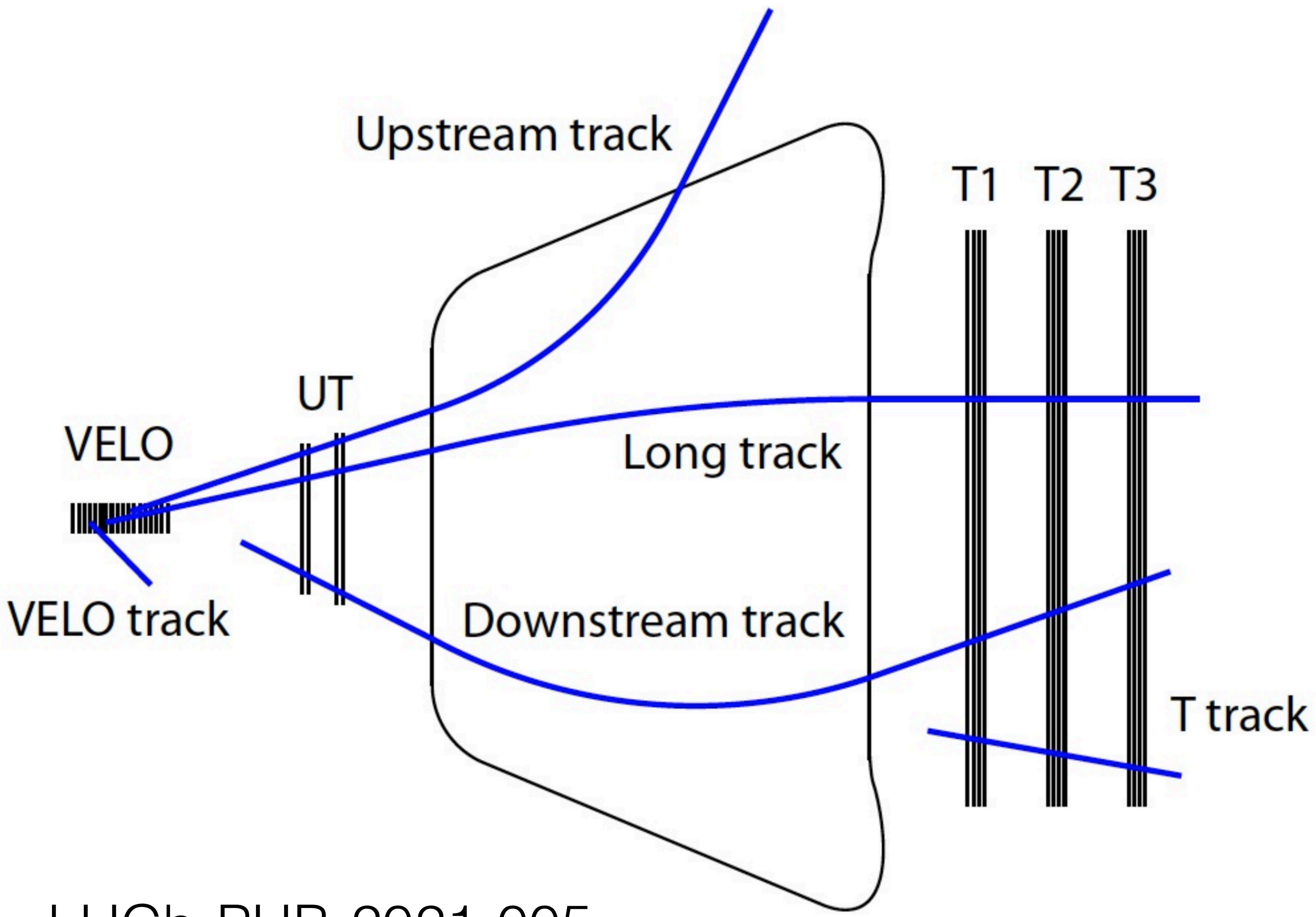


LHCb-PUB-2021-005

- Long tracks: best resolution
  - **Forward Tracking:** VELO (→ UT) →SciFi
  - **Matching:** VELO Tracks + T tracks + (UT)
- Downstream tracks for long-lived particles
  - T tracks + UT hits

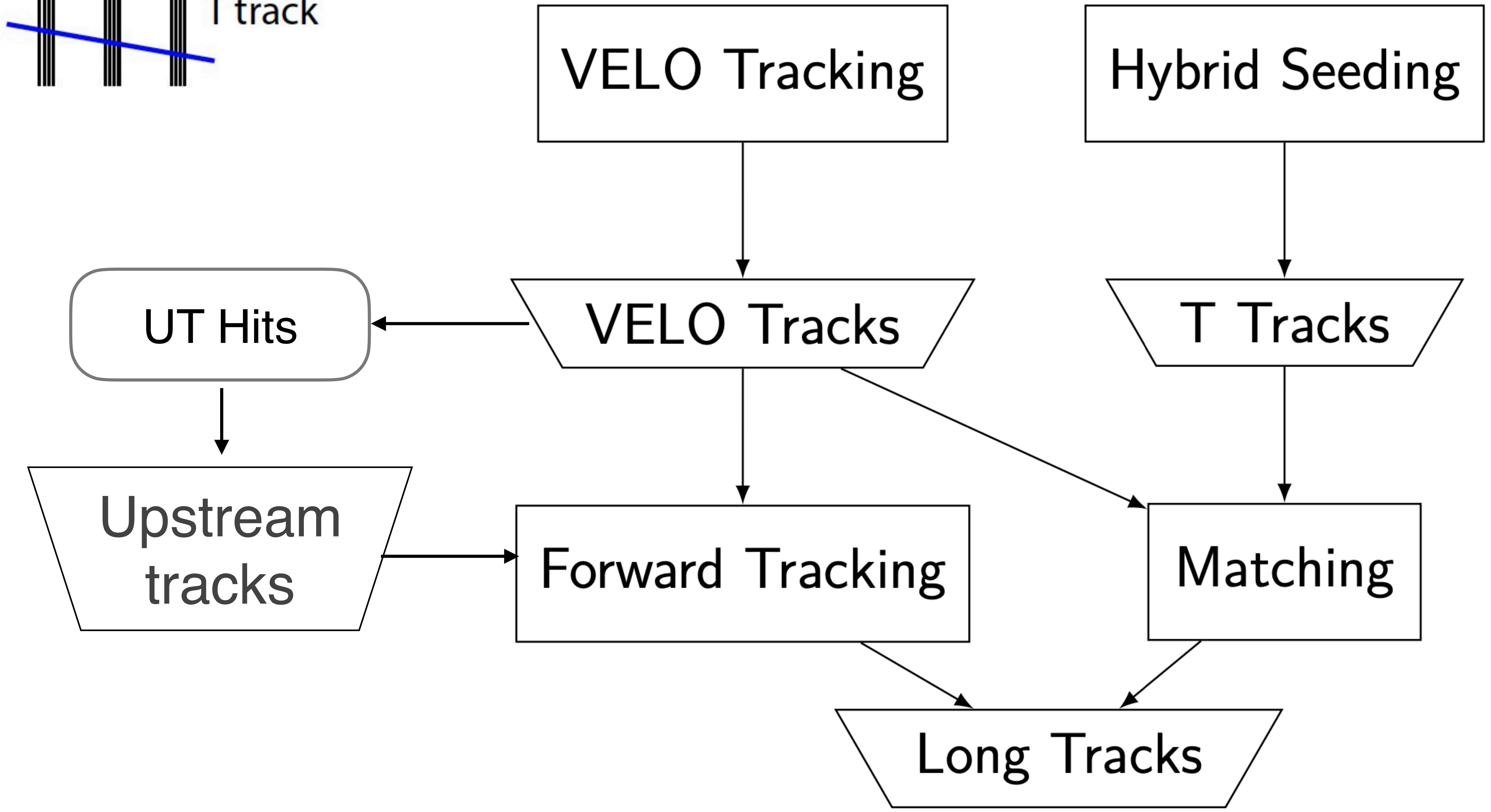


Track Types

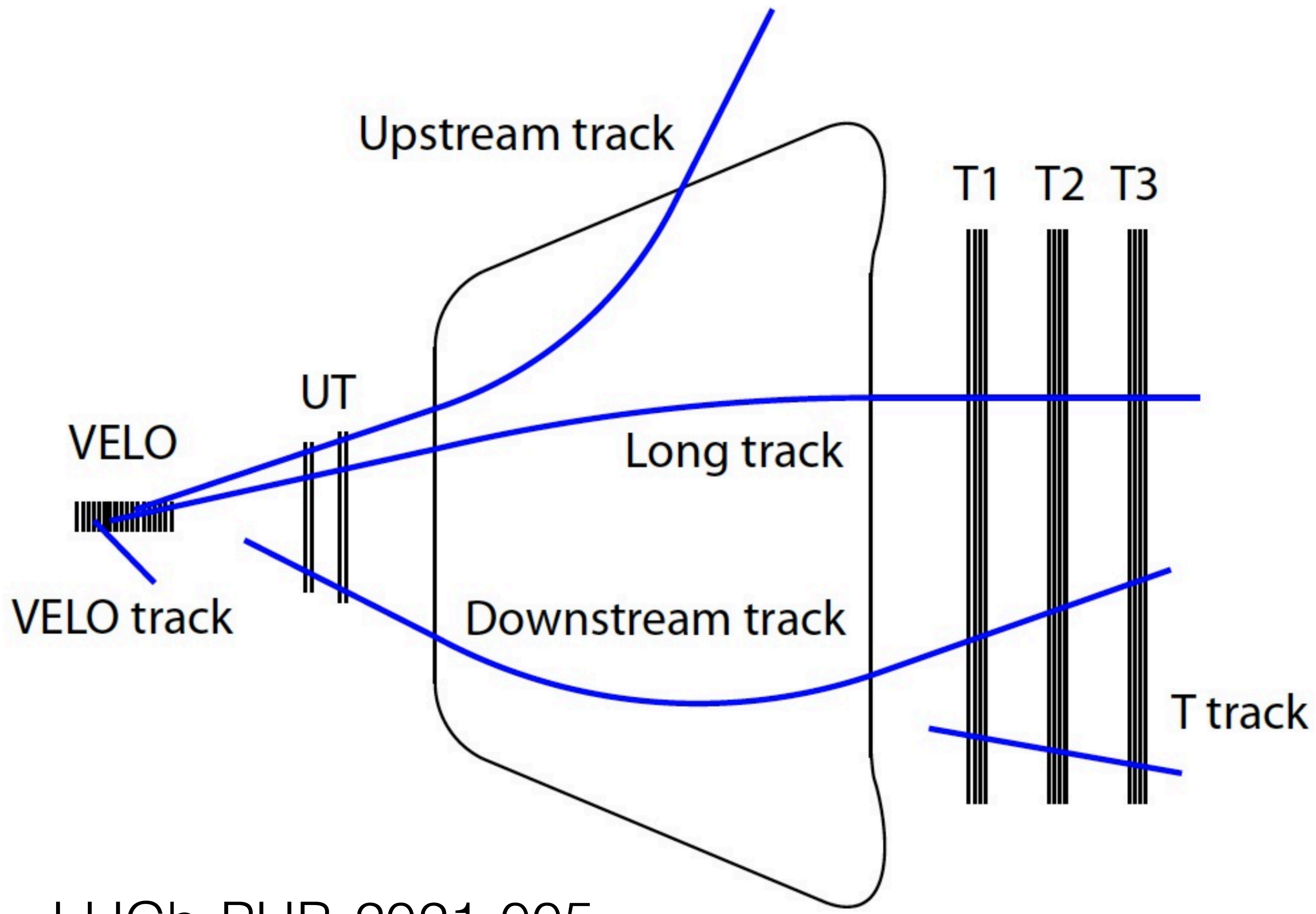


LHCb-PUB-2021-005

- Long tracks: best resolution
  - **Forward Tracking:** VELO (→ UT) →SciFi
  - **Matching:** VELO Tracks + T tracks + (UT)
- Downstream tracks for long-lived particles
  - T tracks + UT hits

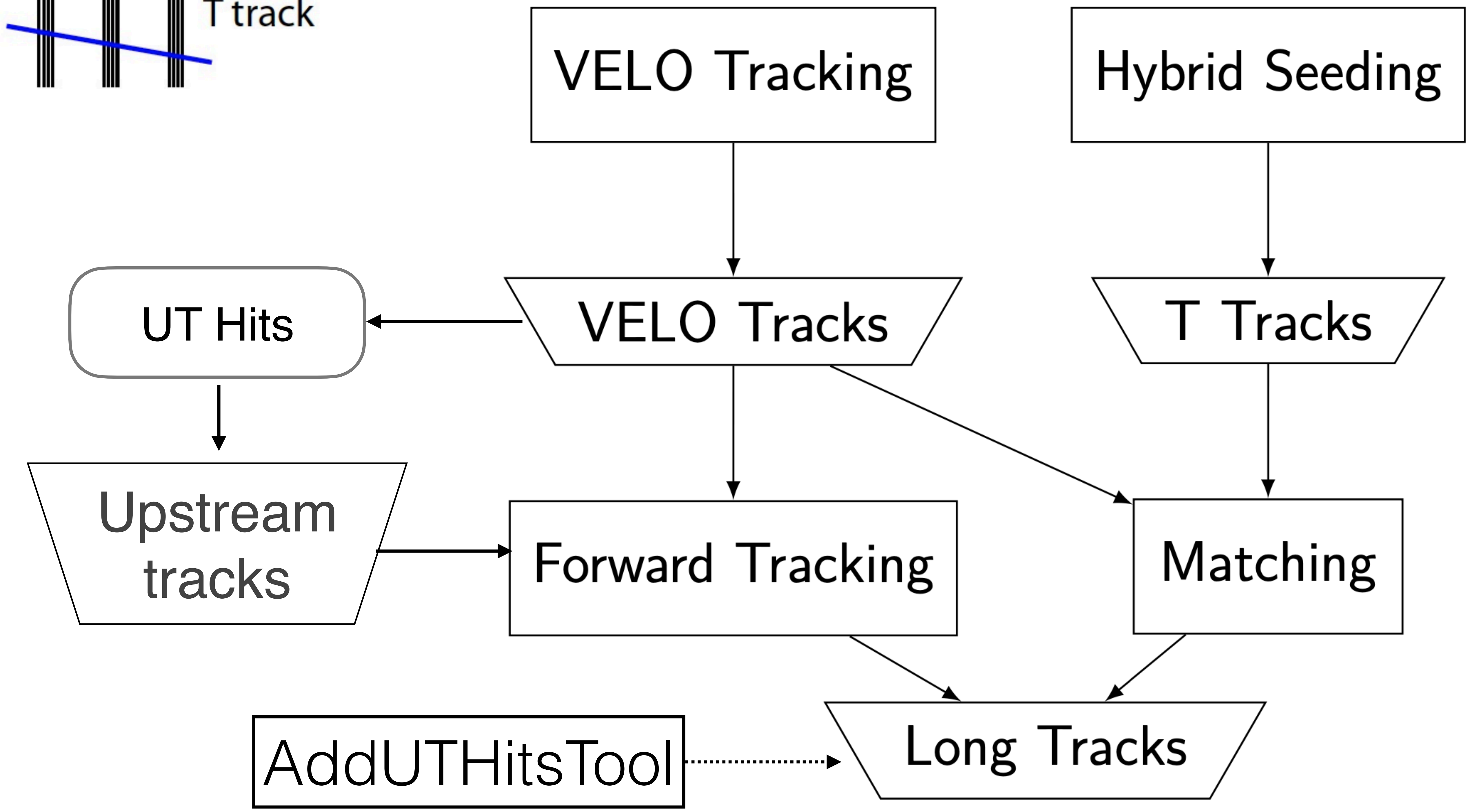


Track Types

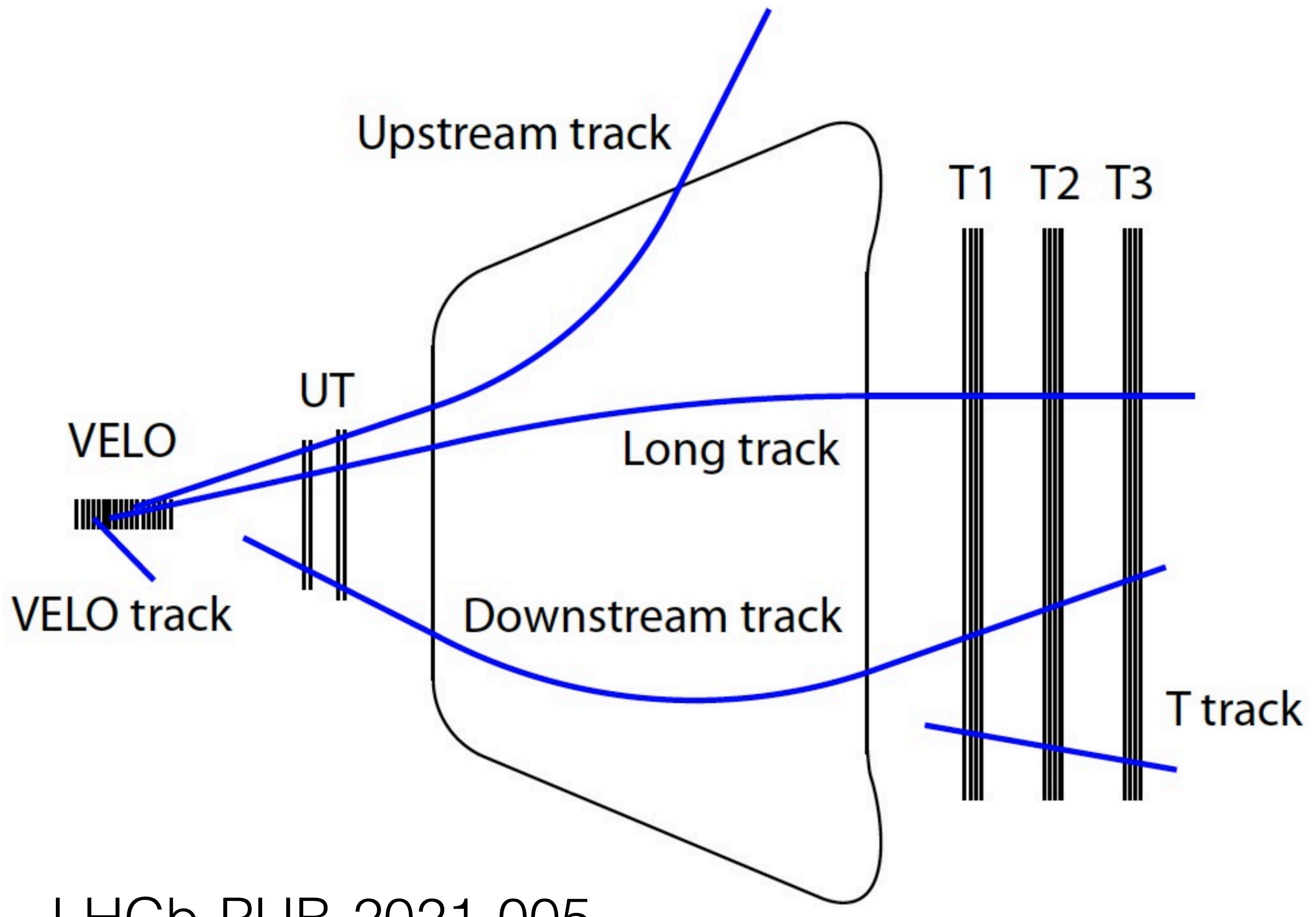


LHCb-PUB-2021-005

- Long tracks: best resolution
  - **Forward Tracking:** VELO (→ UT) →SciFi
  - **Matching:** VELO Tracks + T tracks + (UT)
- Downstream tracks for long-lived particles
  - T tracks + UT hits

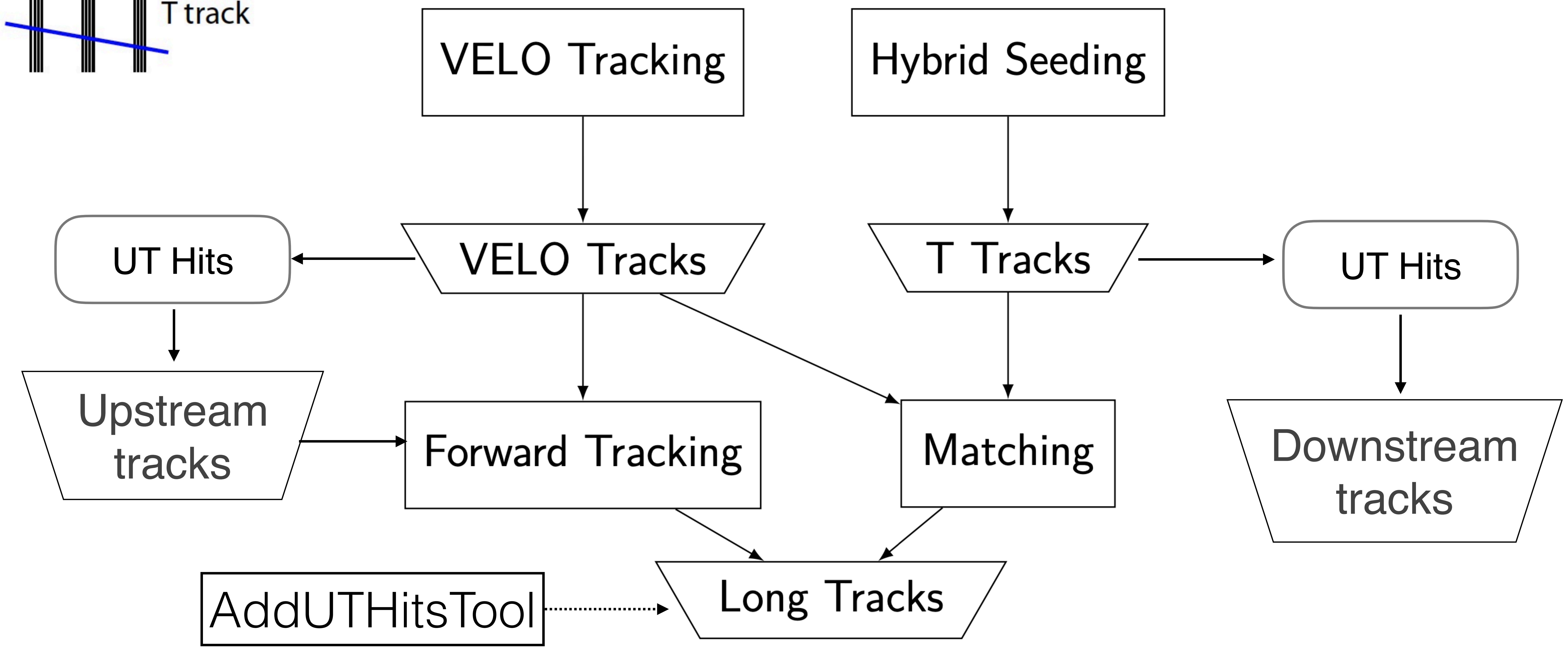


Track Types

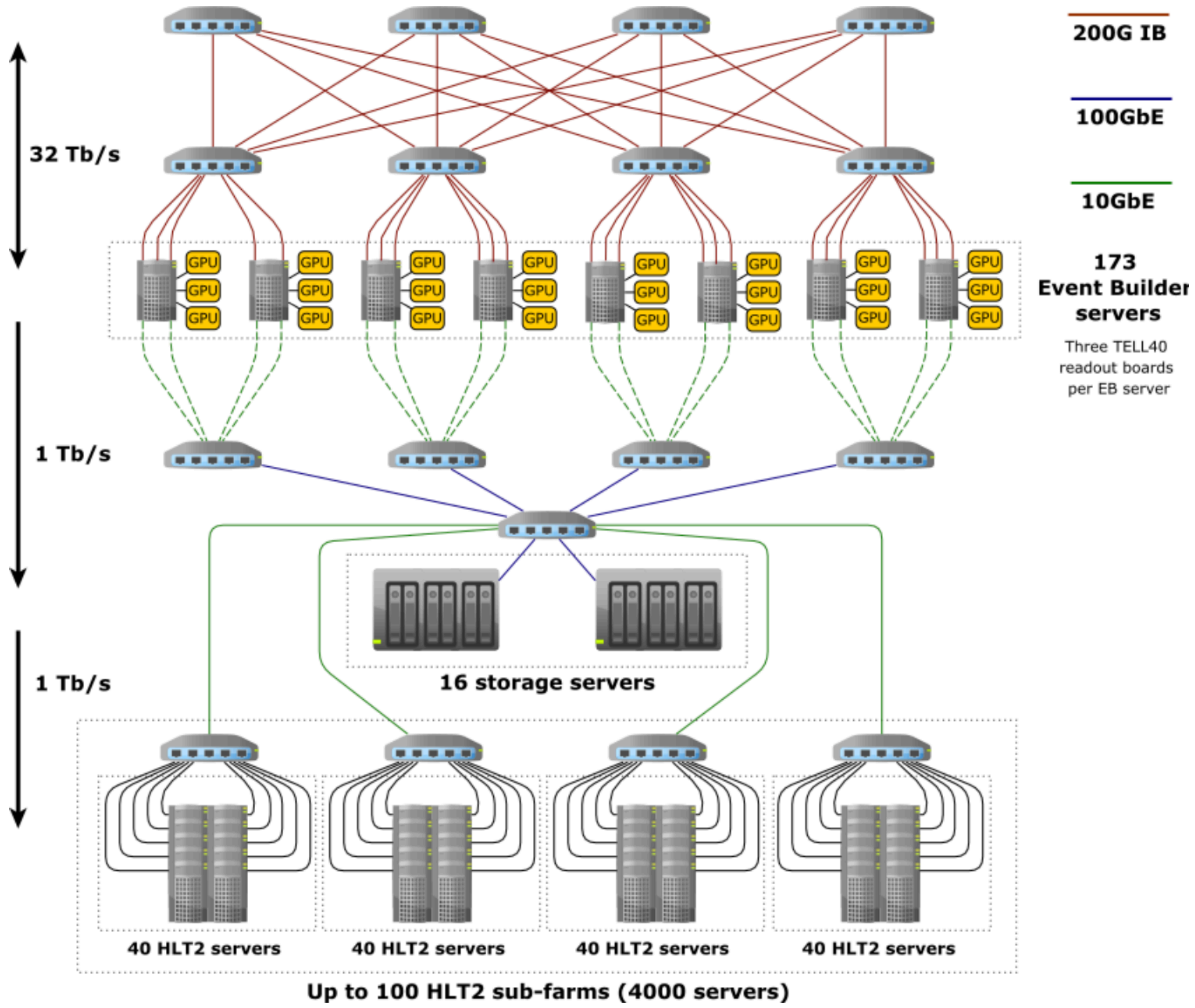


LHCb-PUB-2021-005

- Long tracks: best resolution
  - **Forward Tracking:** VELO (→ UT) →SciFi
  - **Matching:** VELO Tracks + T tracks + (UT)
- Downstream tracks for long-lived particles
  - T tracks + UT hits

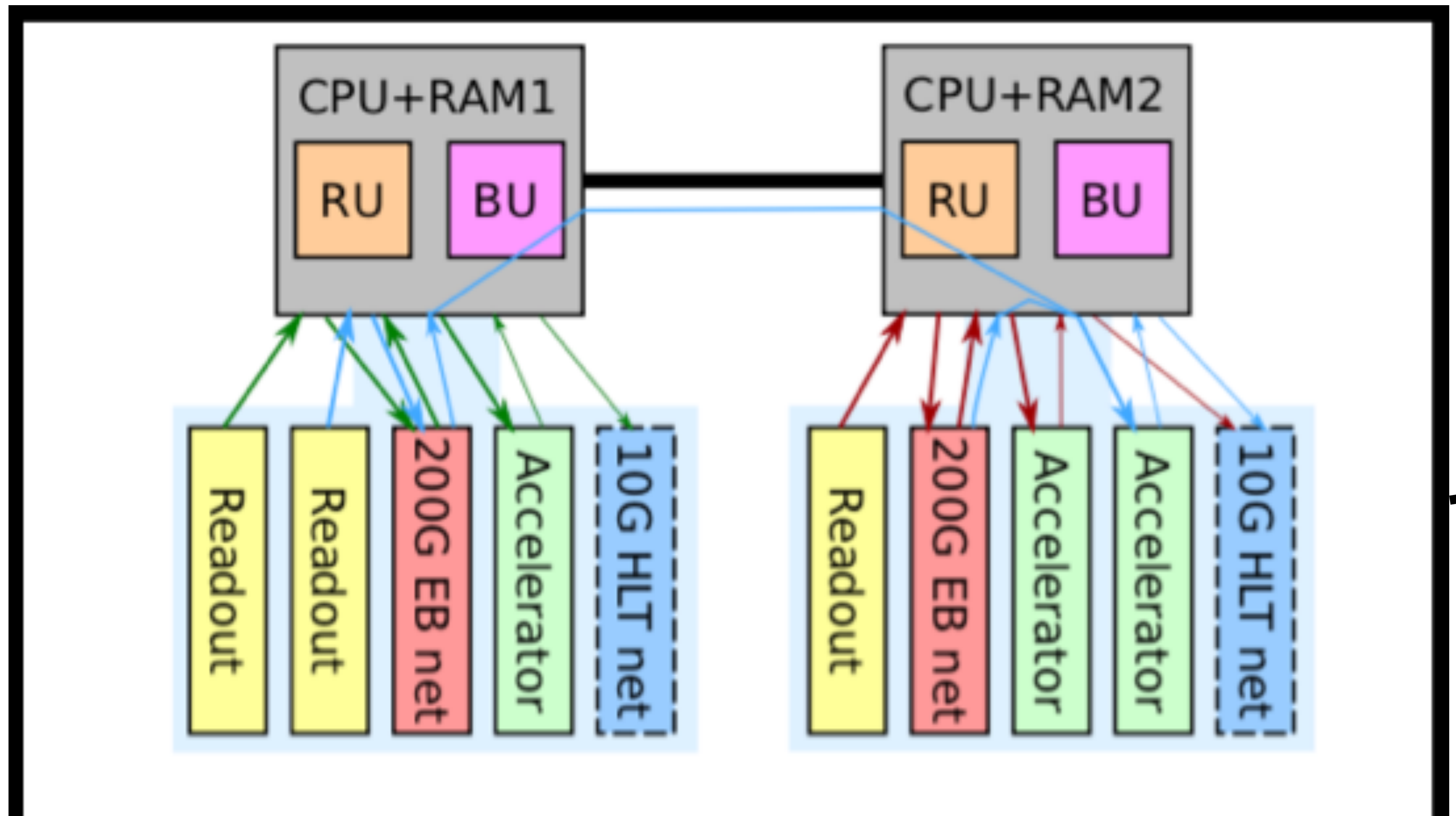


Track reconstruction with GPU

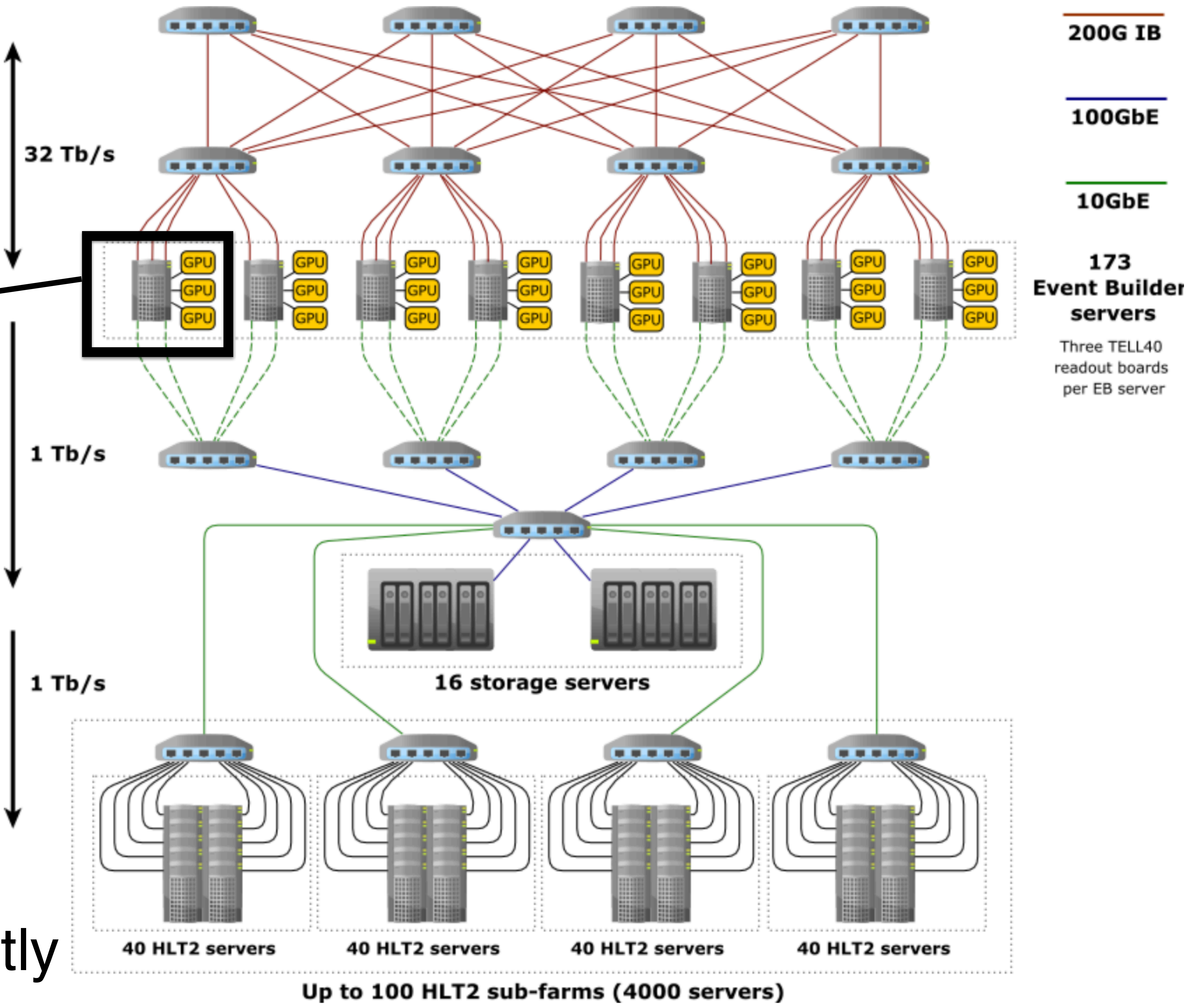


Computing and Software for Big Science(2020)4:7

Track reconstruction with GPU



GPU-equipped event builder PC, with traffic of all three readout cards.



- Each Event-builder hold 2 GPU cards
- 173 EBs → 346 GPUs
- Reduce data volume by a factor 30-60, significantly reducing the networking from EB to CPU farms

Computing and Software for Big Science(2020)4:7

The Allen software project (GPU HLT1)

- Named after Frances E. Allen
- Fully standalone software project: https://gitlab.cern.ch/lhcb/Allen
- Framework developed for processing LHCb's HLT1 on GPUs

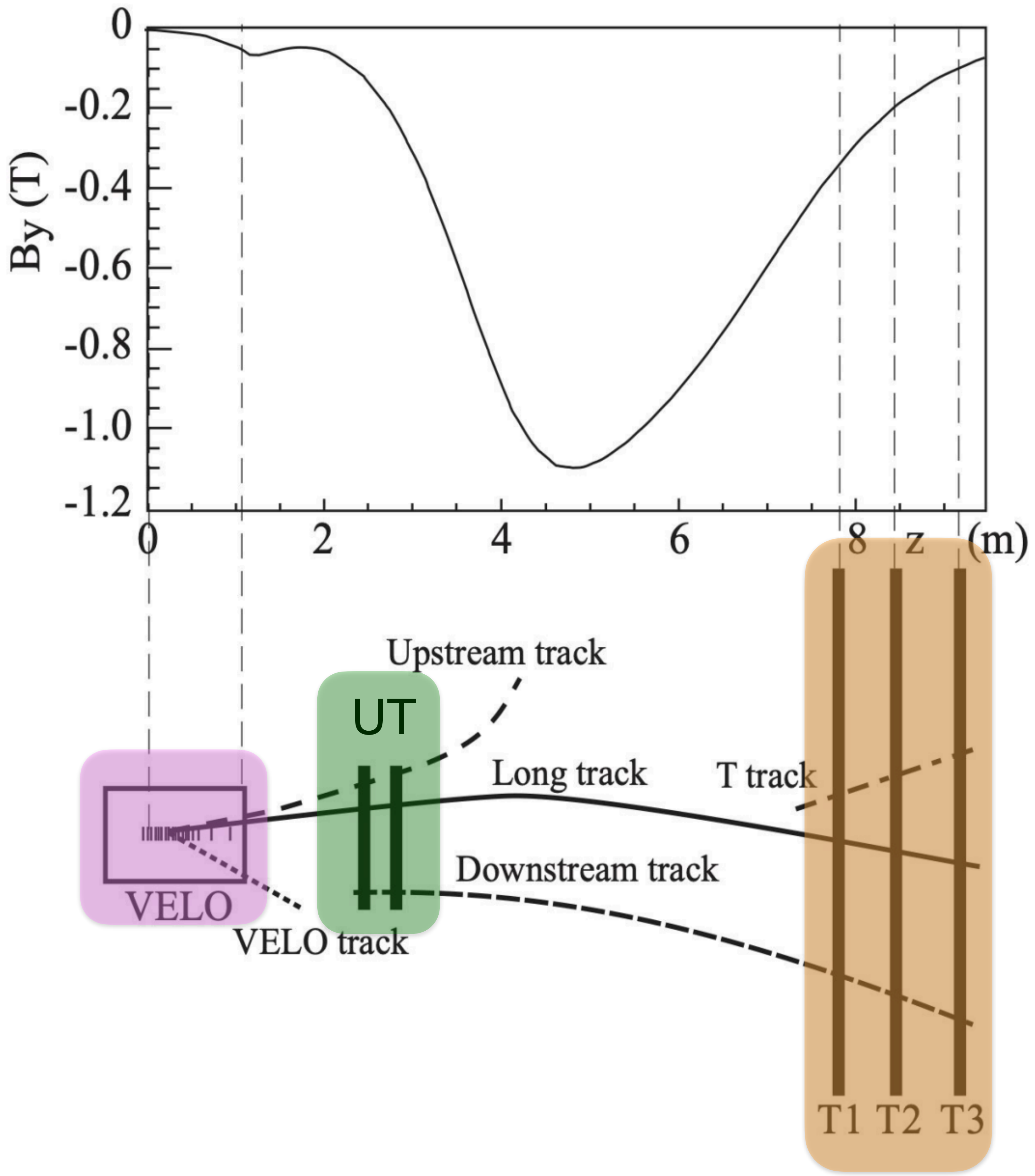
First application of GPU in Trigger of HEP experiment!

- Cross-architecture compatibility via macros & few coding guide lines
  - GPU code written in CUDA, runs on CPUs, Nvidia GPUs(CUDA), AMD GPUs (HIP)
- Algorithms sequences defined in python and generated at run-time
- Multi-event processing with dedicated scheduler
- Memory manager allocates large chunk of GPU memory at start-up
- Reconstruction algorithms re-designed for parallelism and low memory usage: O(MB) per core

Track reconstruction with GPU (HLT1)

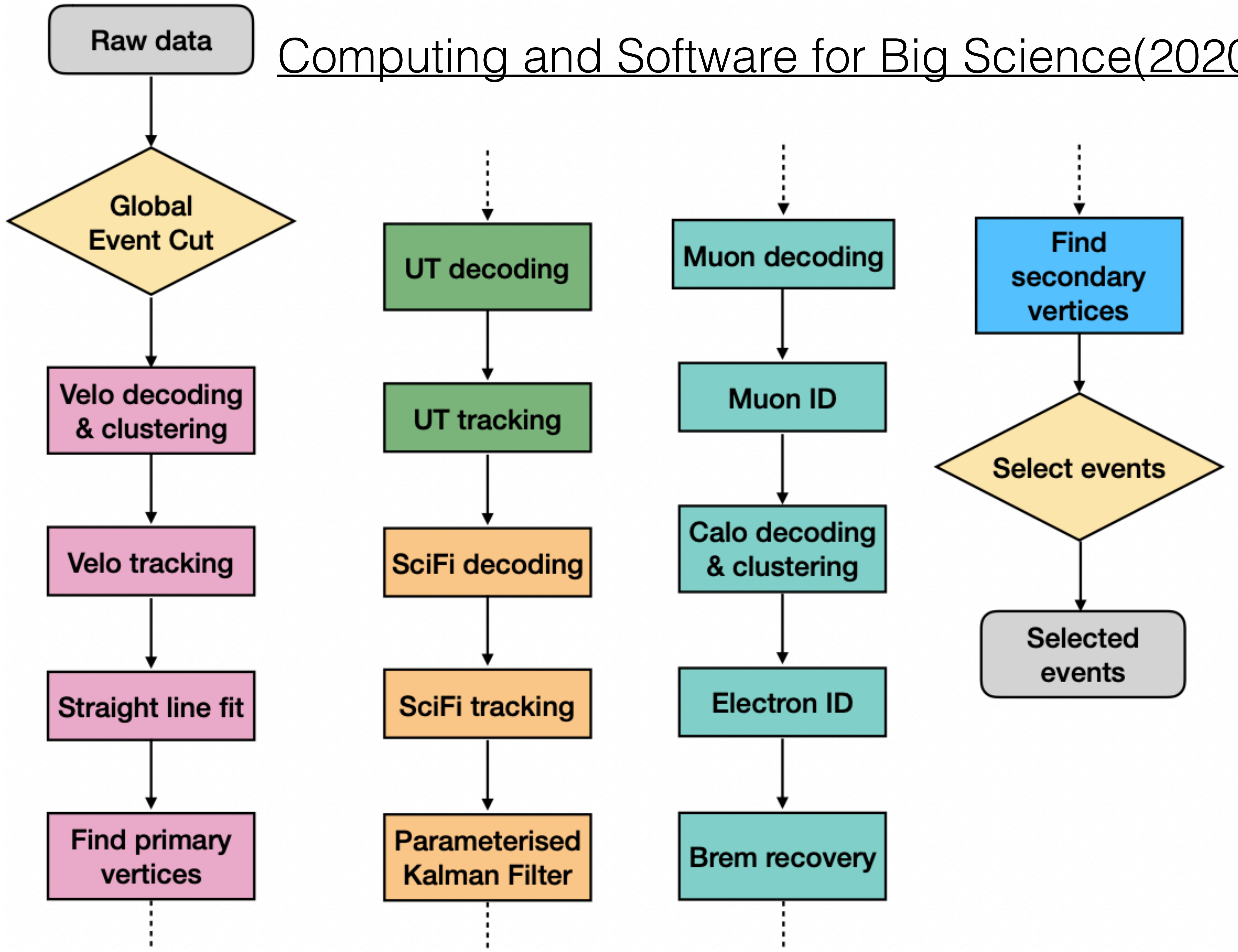
- HLT1 filters the 30 MHz pp collision to 1 MHz with GPU architecture
- Partial reconstruction using hits from VELO, (UT), SciFi & Muon
  - → High momentum long charged track reconstruction & muon identification
  - → Few inclusive single and two-track selections to reduce rate

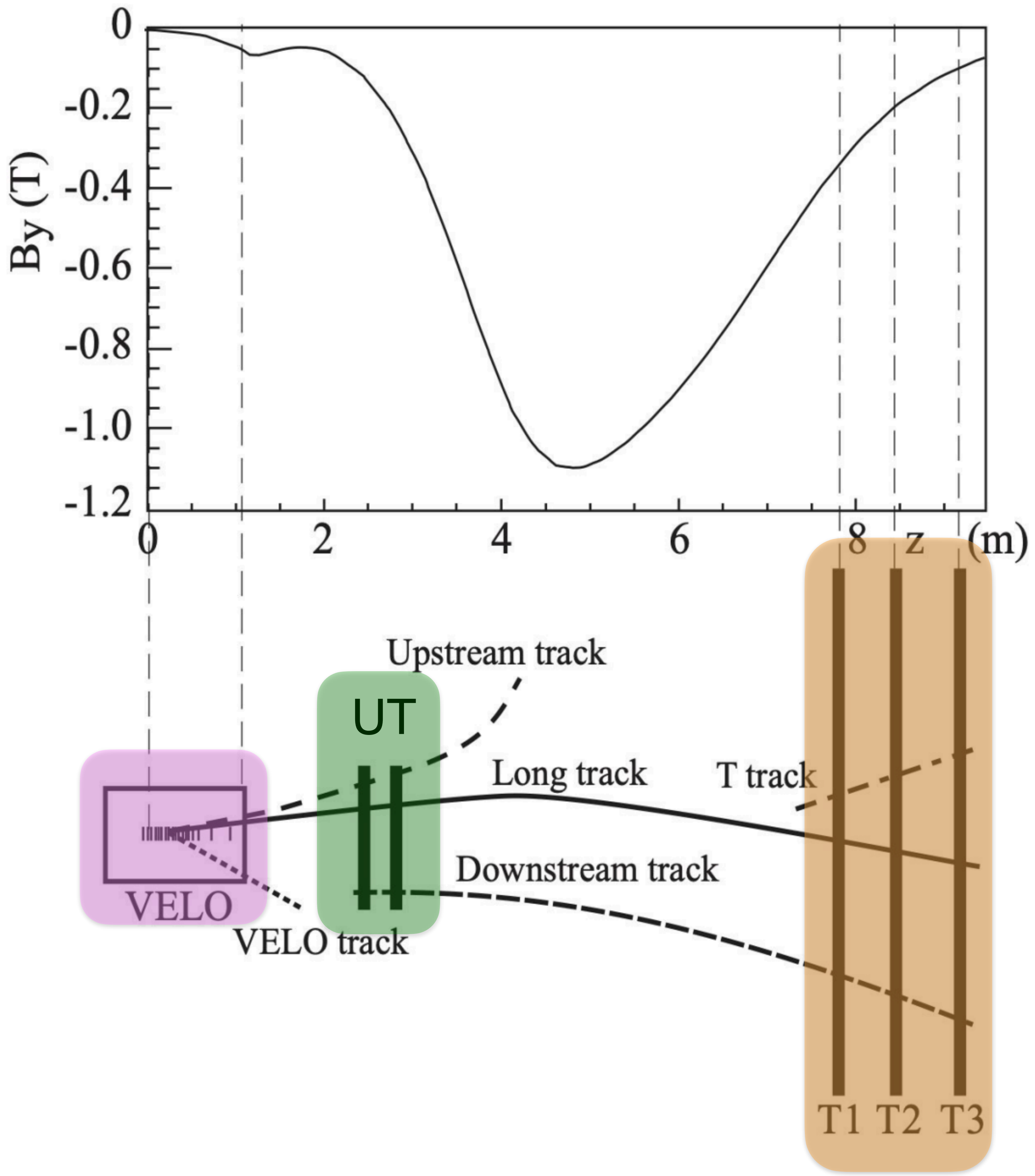
Computing and Software for Big Science(2020)4:7



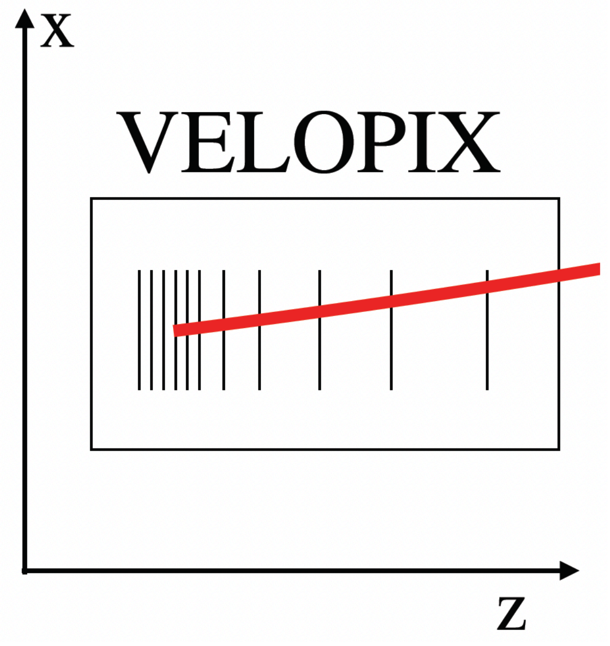
Track reconstruction with GPU (HLT1)

- HLT1 filters the 30 MHz pp collision to 1 MHz with GPU architecture
- Partial reconstruction using hits from VELO, (UT), SciFi & Muon
  - → High momentum long charged track reconstruction & muon identification
  - → Few inclusive single and two-track selections to reduce rate

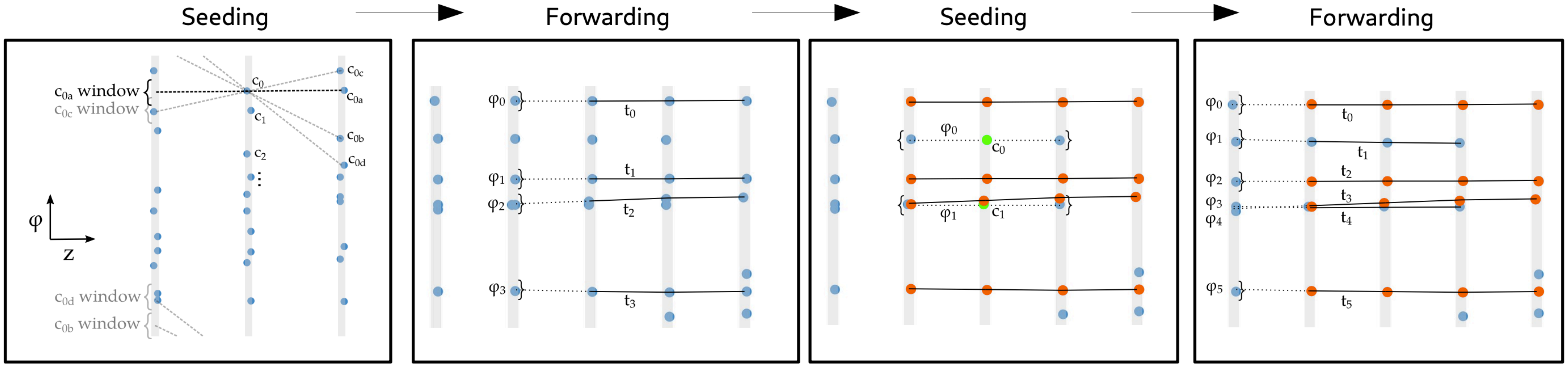




VELO: Tracking



- 26 layers of silicon pixels detector

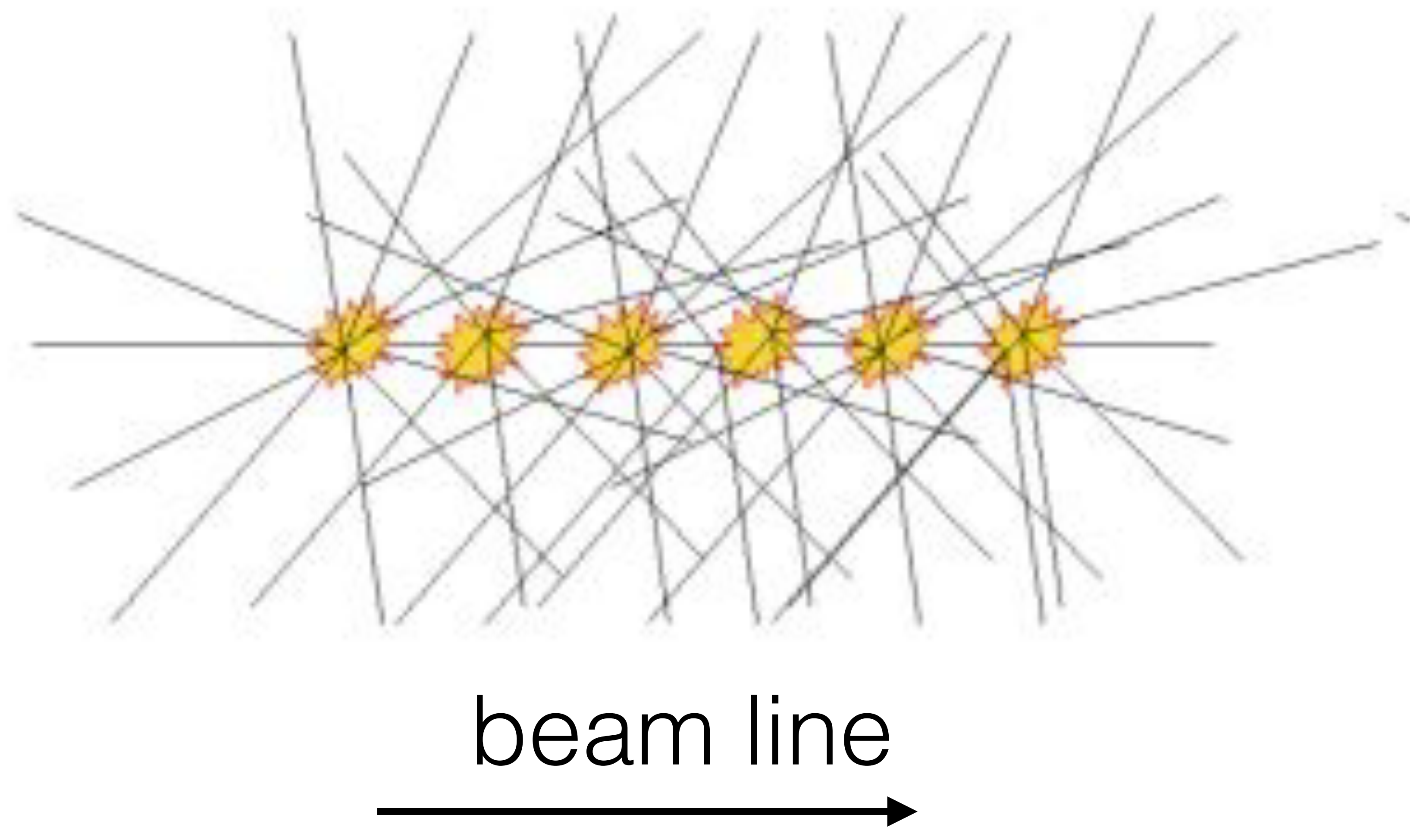


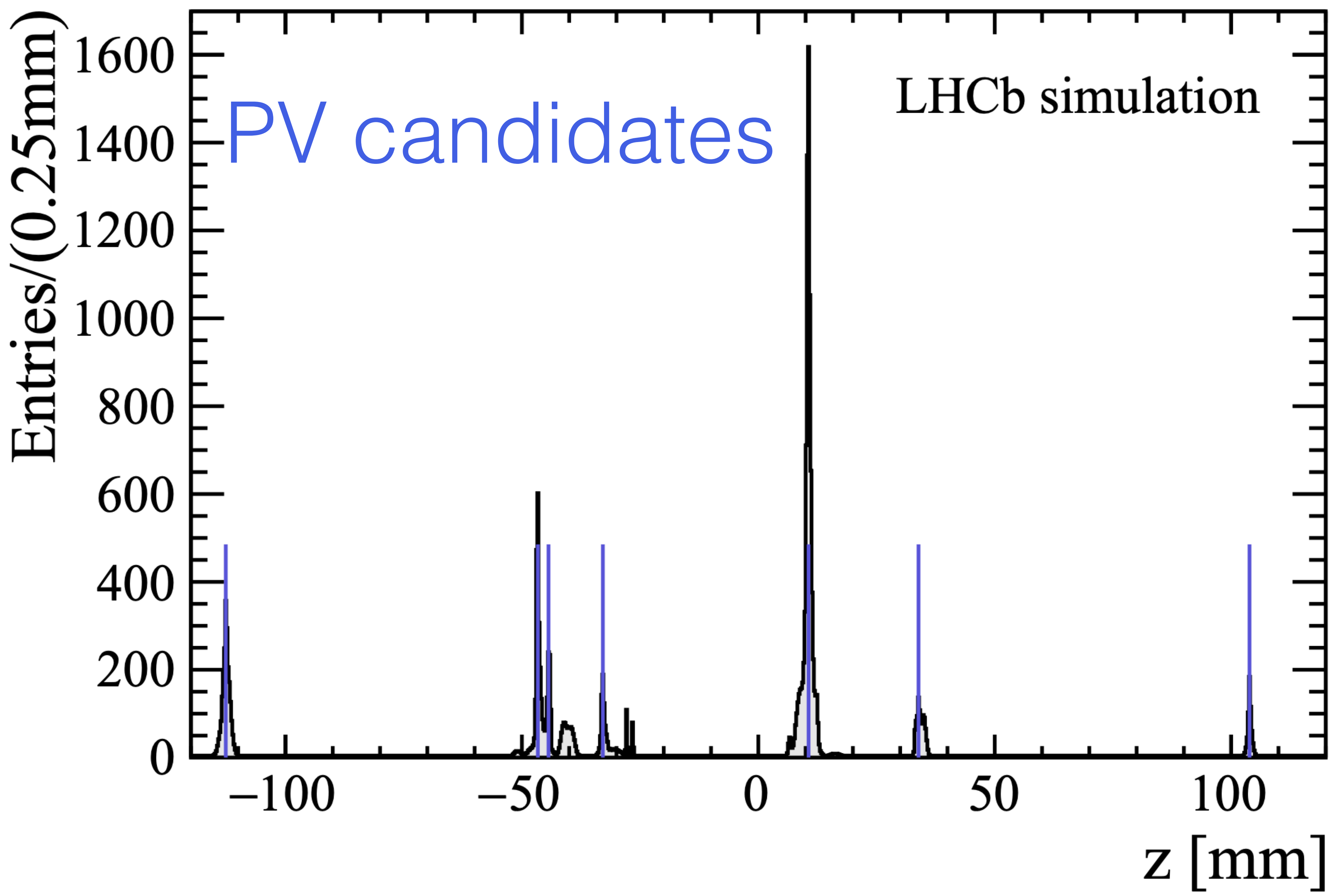
- Build “triplets” of three hits on consecutive layers → parallelisation
- Choose them based on alignment in phi
- Hits sorted by phi →memory accesses as contiguous as possible: data locality
- Extend triplets to next layer → parallelisation

D D. Campora, N. Neufeld, A. Riscos Núñeez: “A fast local algorithm for track reconstruction on parallel architectures”, IPDPSW 2019

VELO: Vertex reconstruction

LHCb-Figure-2020-005

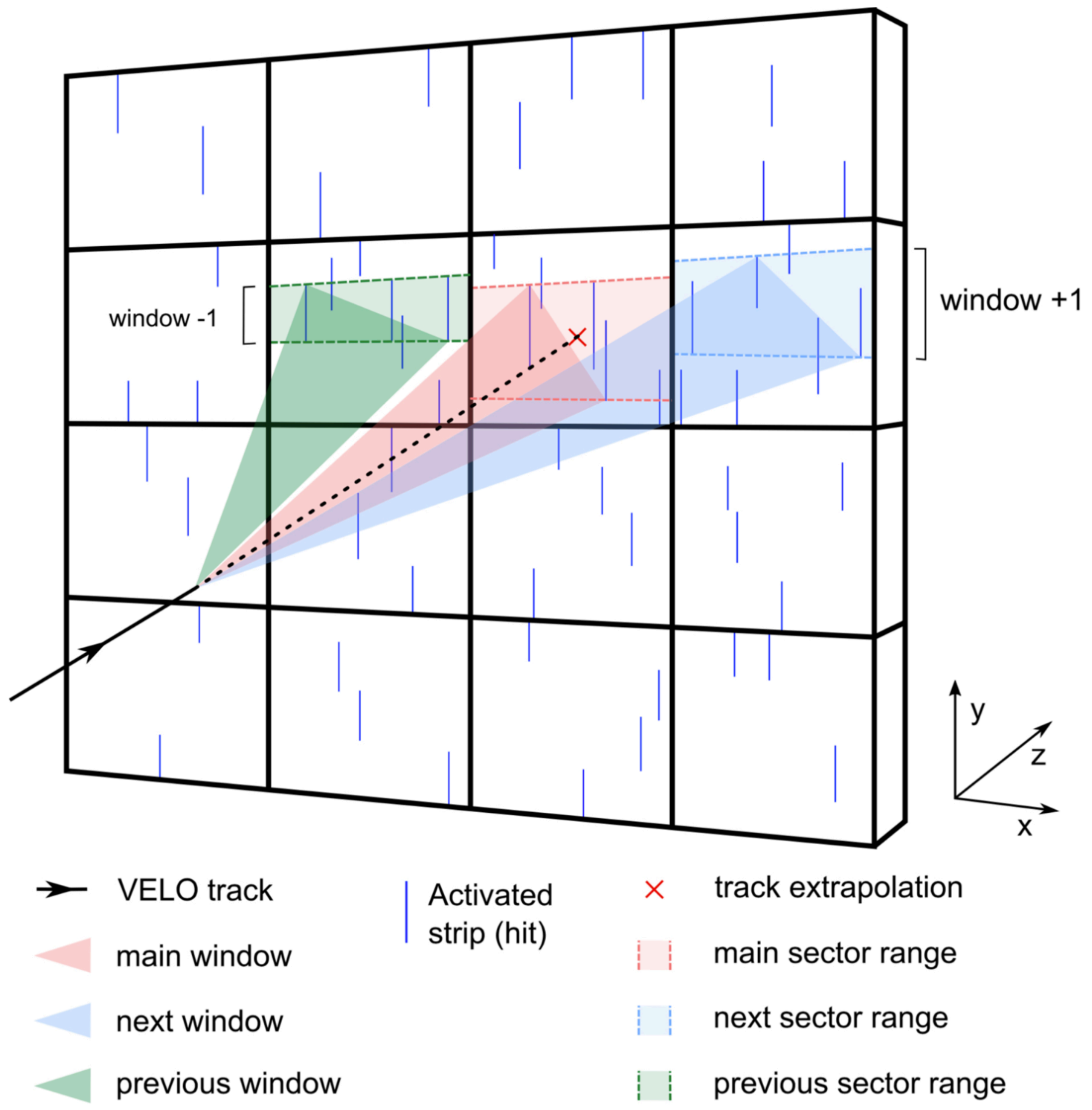




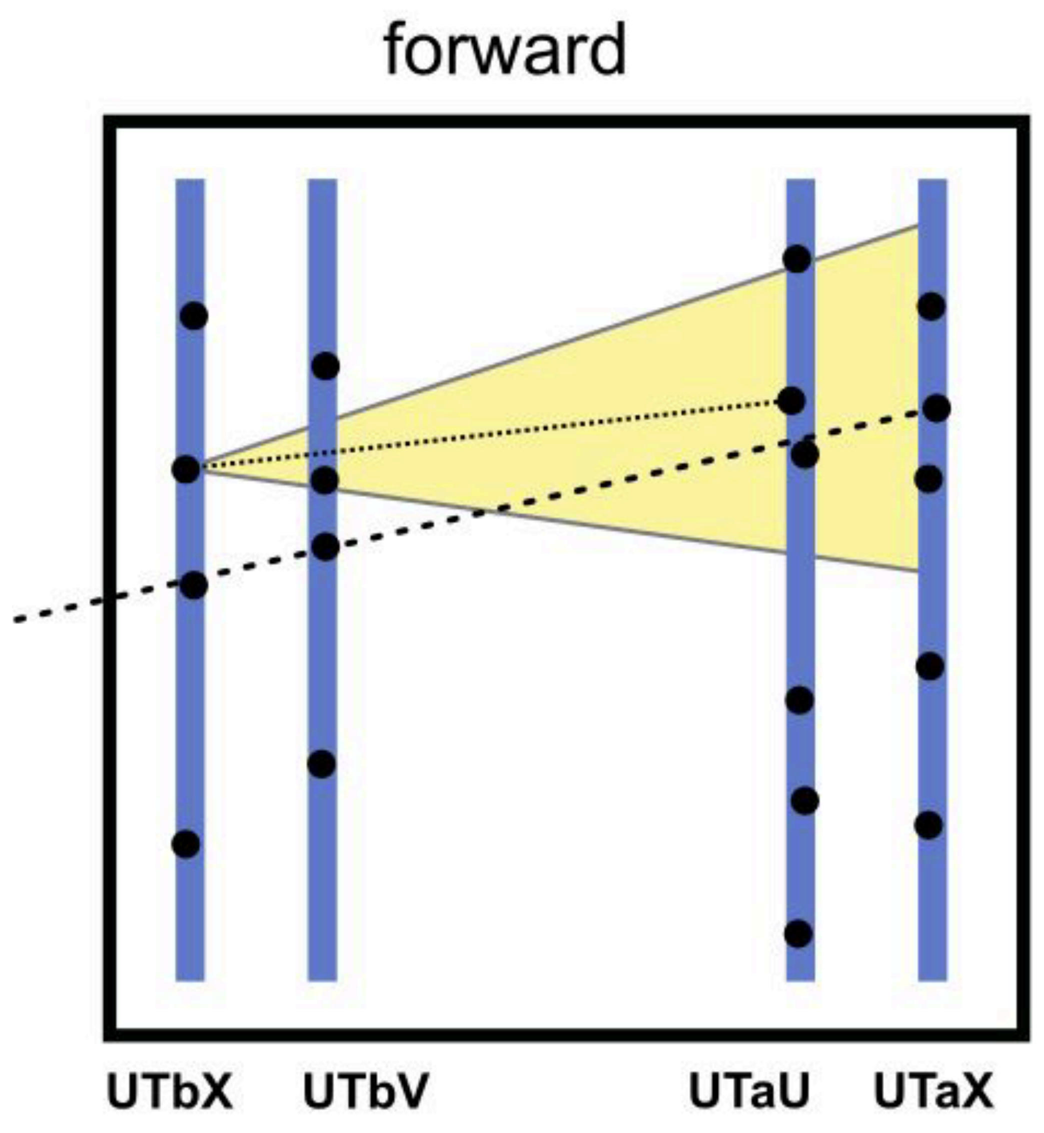
- Primary vertices (PVs) are extended along the beam direction (z-axis)
- Histogram the tracks' z position closest to the beam line
- Every track contributes to every PV candidate with a weight
- No inter-dependence between PV candidates, as every track contributes to every PV
- PV fitting can be done in parallel for every candidate

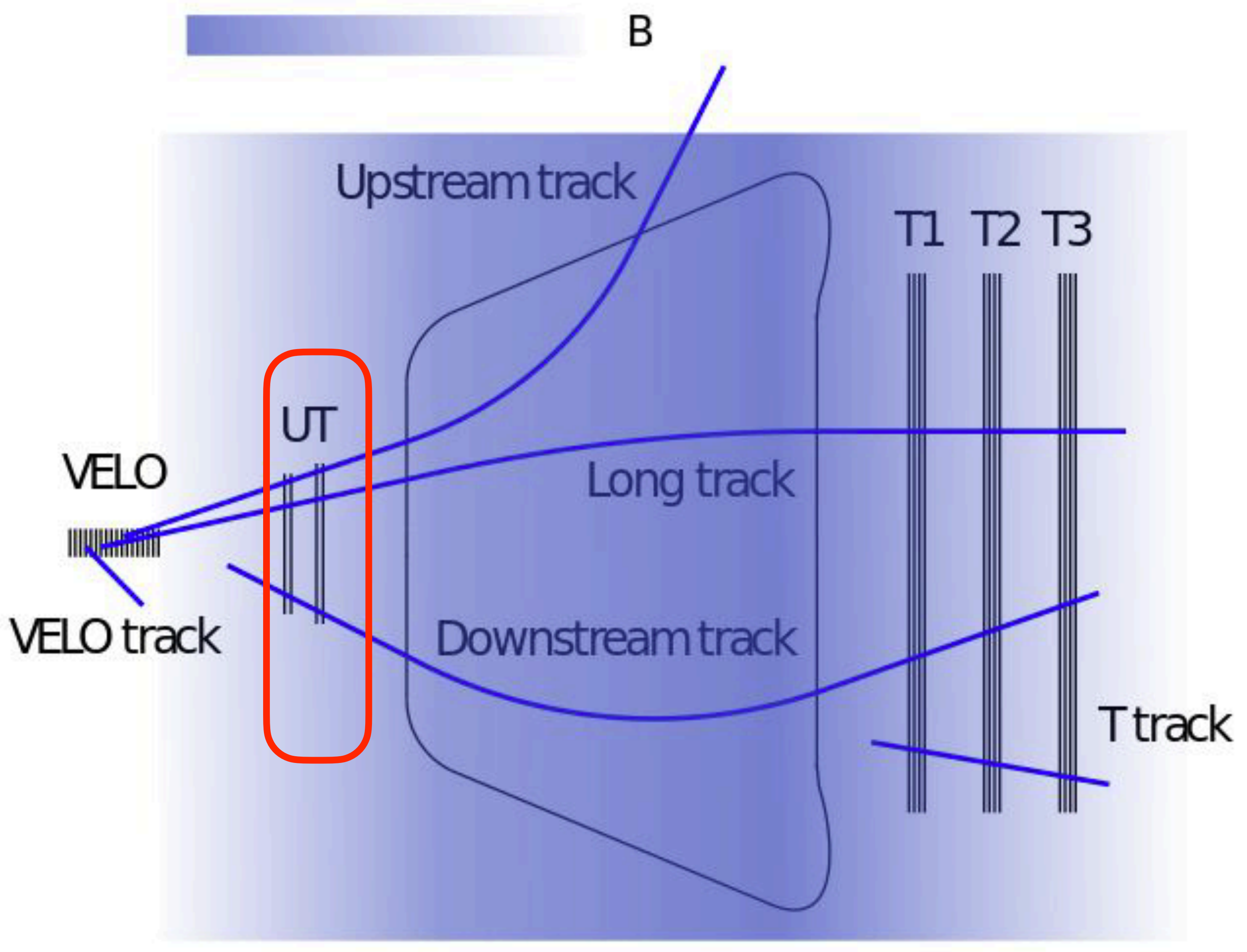
UT: Tracking

- Four layers of silicon strip detector



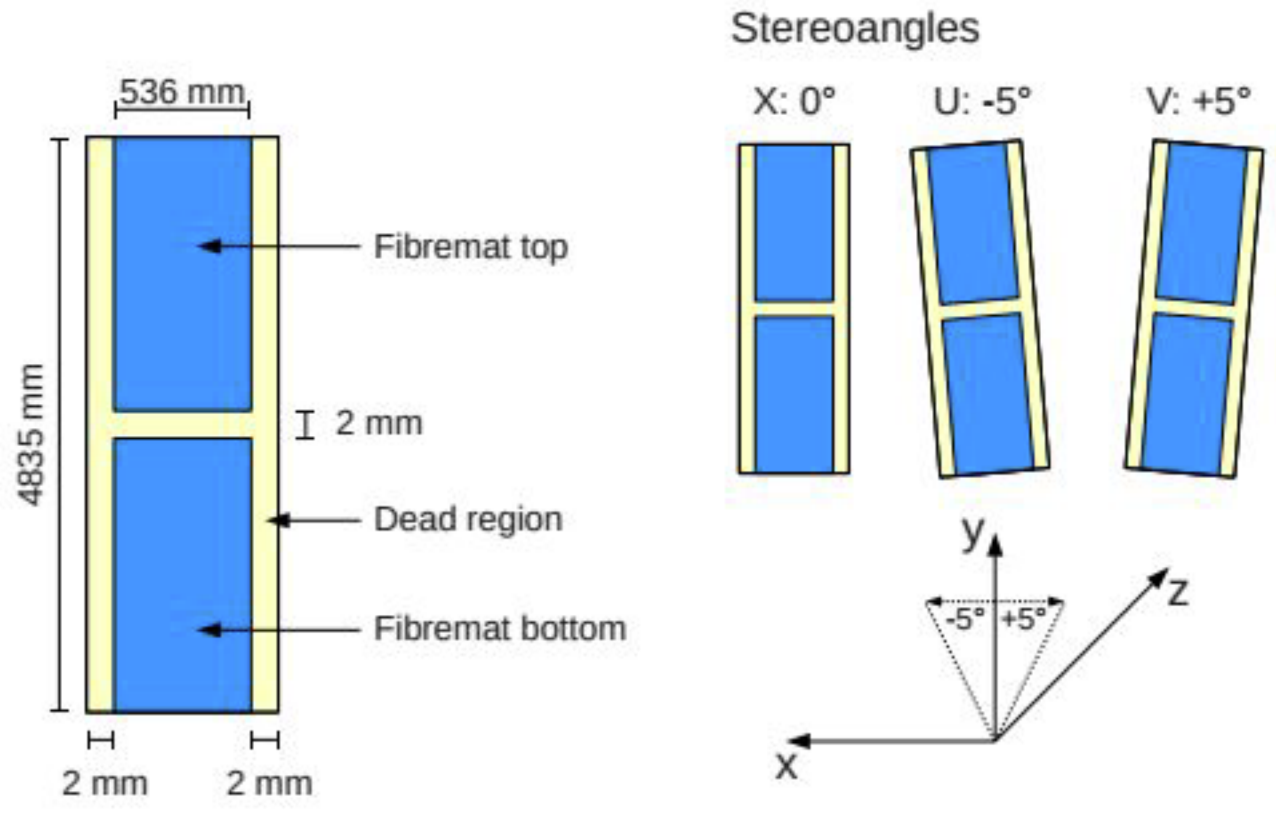
- Extrapolate VELO tracks to the UT planes based on lookup table for minimum momentum requirement
- Define search regions in each UT plane → parallelisation
- Trackless finding inside windows from 4 layers building combinatorics → parallelisation

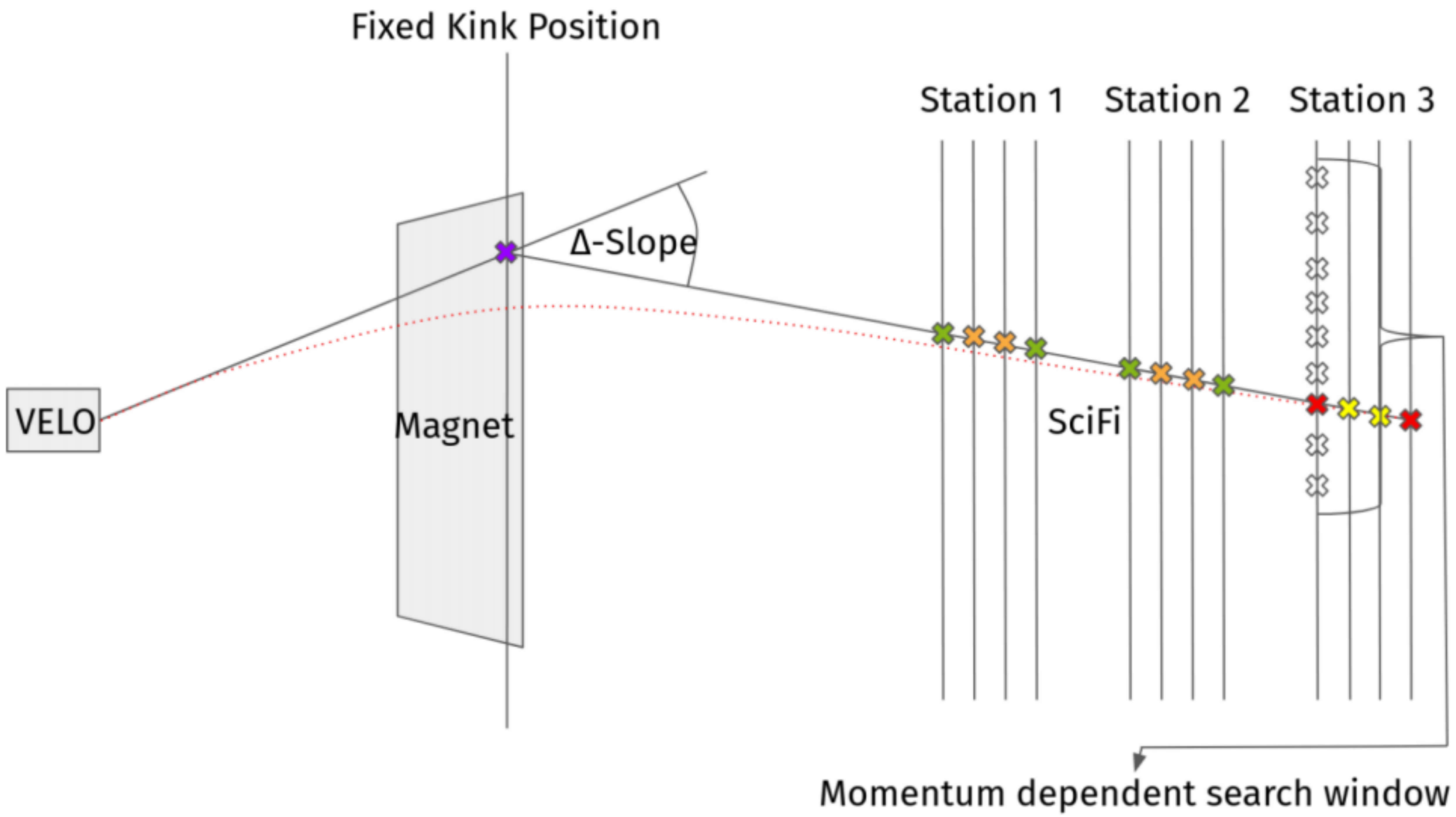




SciFi: Long track reconstruction

- 3 stations with 4 layers scintillating fibres each (*xuvx* configuration)
  - Extrapolate each Upstream track in the 12 layers of the SciFi
  - Build triplets combinations using T1/2/3, Best triplets selected according to local parameterisation of magnetic field
  - Forward all triplets to remaining layers with an extra parameterised corrections in the non-bending plane

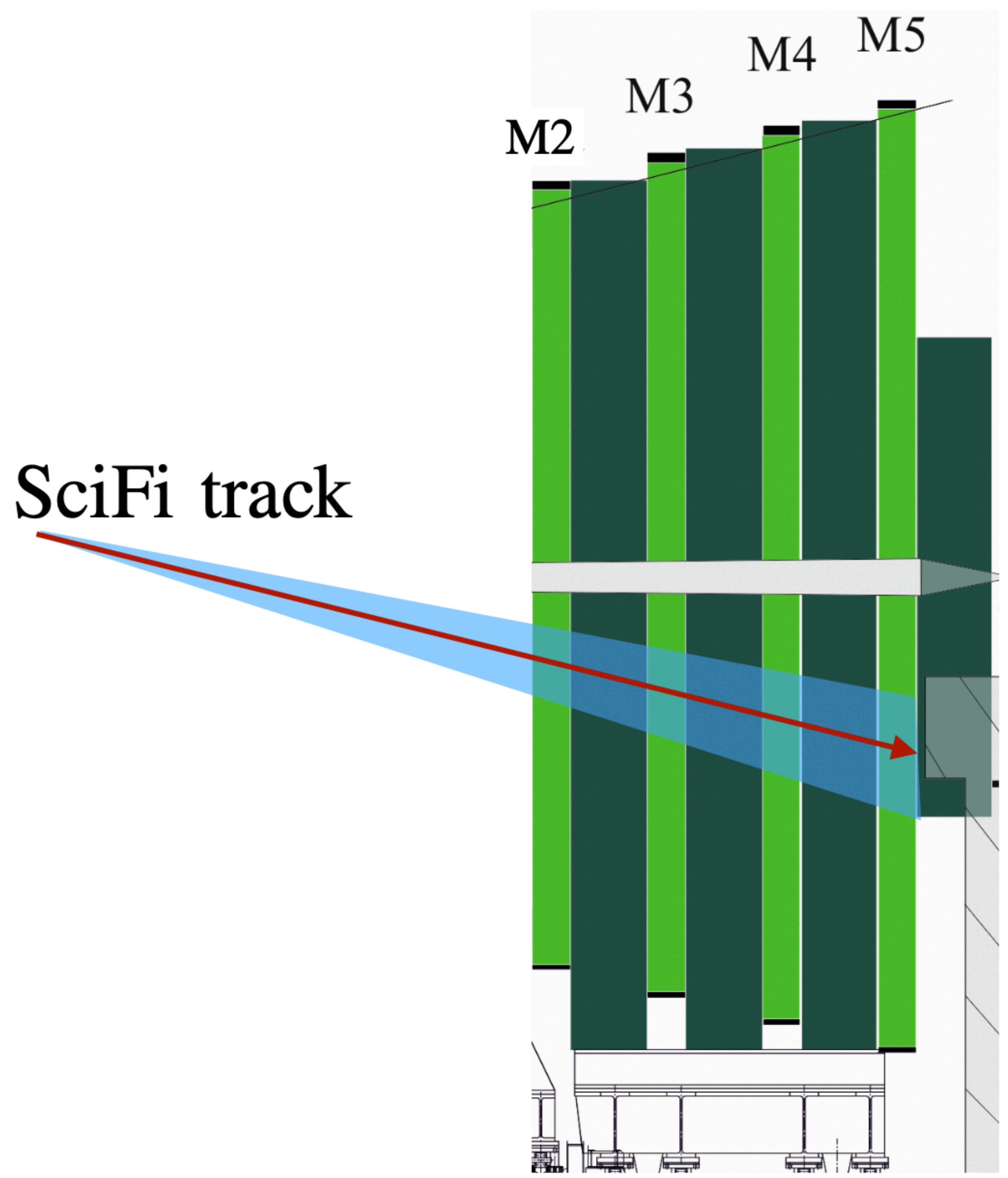




Muon identification & track fit

Muon identification

- Project Long tracks to 4 layers of Multi-wire proportional Muon chambers
- Find hits in side the FoI for μ identification
- Parallelise across tracks and muon chambers



Track fit

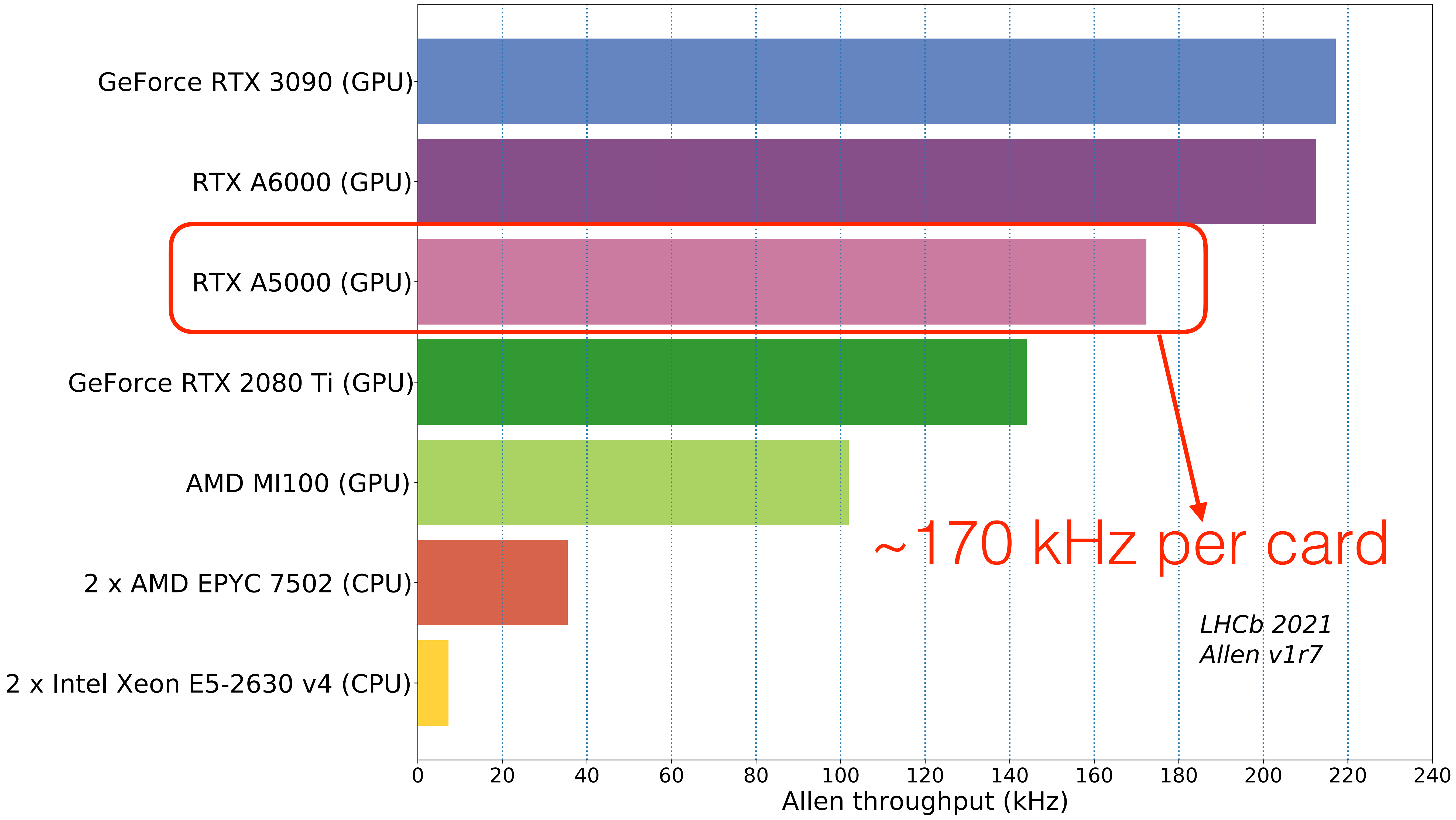
- Goal: improve track description close to the beam line for precise determination of the impact parameter
- Only fit part of the track within the Velo detector
- Parameterized Kalman filter → no need for magnetic field map and detector material description

LHCb Performance Definitions

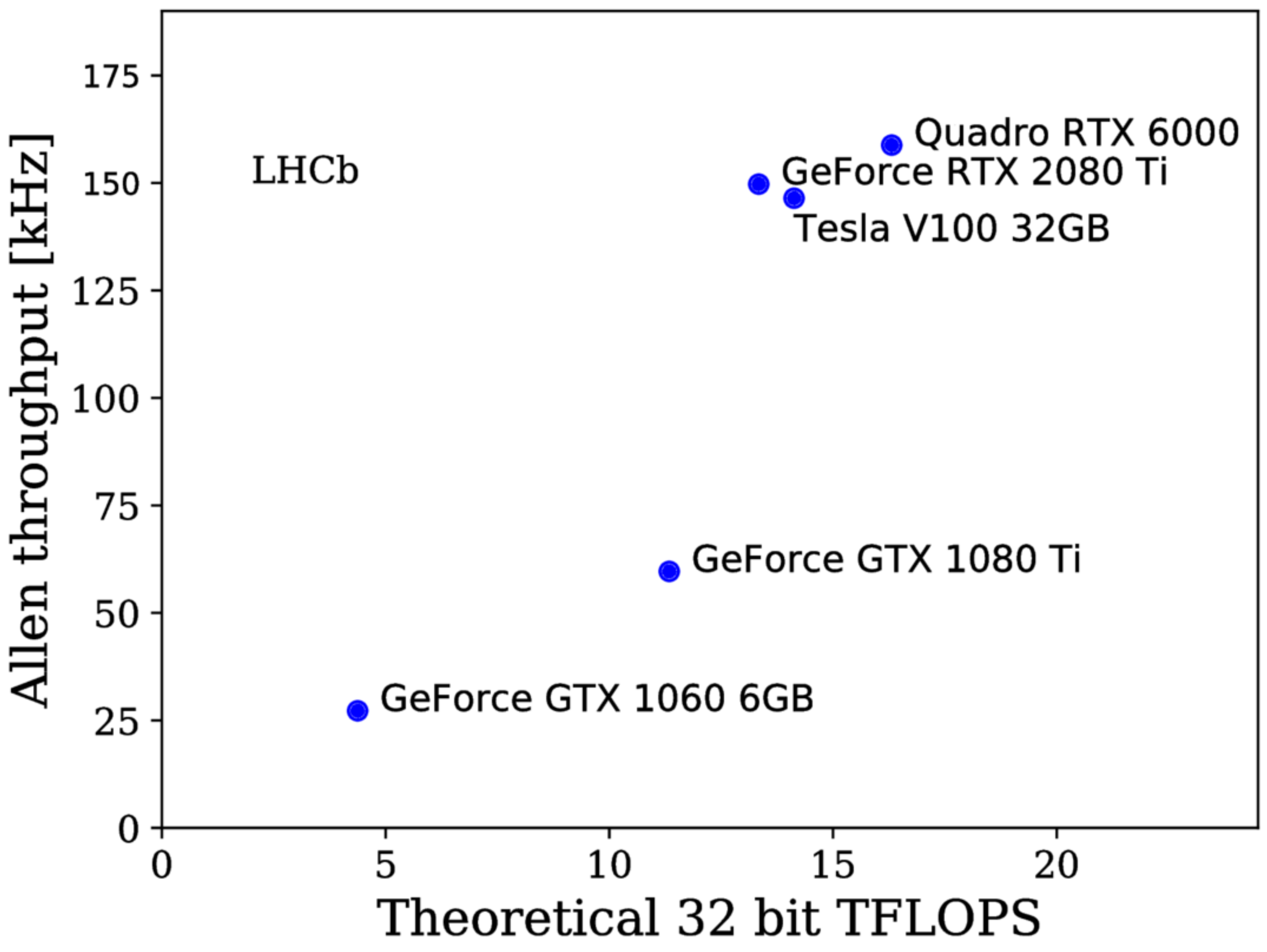
LHCb-Figure-2020-014

- A **track** is **matched** to a simulated particle if **at least 70% of the hits** come from the same simulated particle
- **Efficiency:** number of matched reconstructed tracks divided by number of reconstructible particles
- **Reconstructible particles** have a minimum number of hits in the sub-detectors for which the efficiency is being determined
- A **PV** is matched to a simulated PV if the **distance along the z-axis is less than five times the uncertainty** of the reconstructed PV
- **Muon identification efficiency** is determined with respect to all tracks matched to a simulated track
- **Computational performance (throughput)** measured with events representative of the Run 3 conditions on several GPU cards

HLT1 Throughput Performance



LHCb-Figure-2020-014



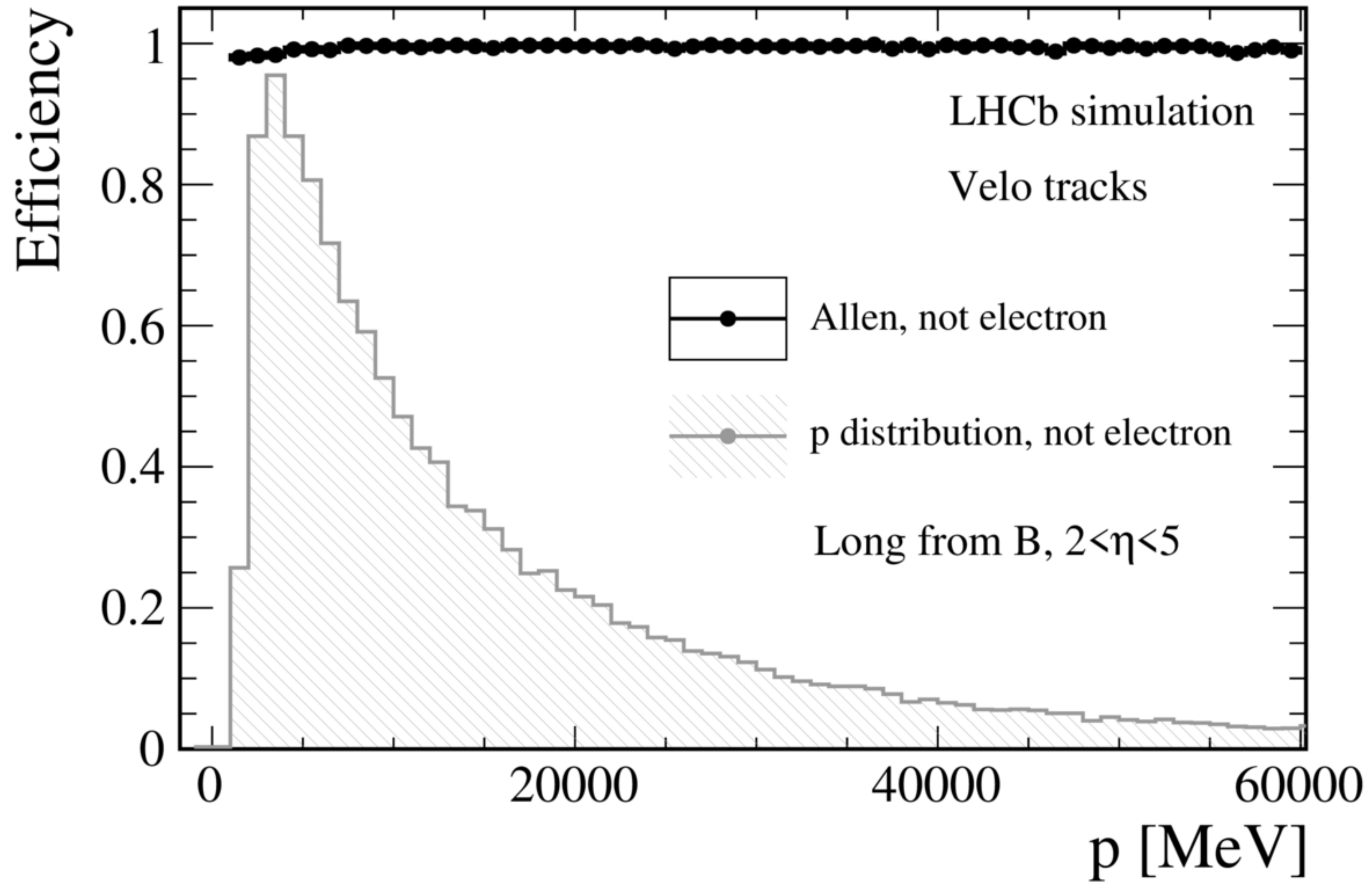
O(200) Nvidia RTX A5000 GPUs implemented to reach 30 MHz, so there is plenty of spare capacity!

Excellent throughput scaling with theoretical TFLOPS of GPU card

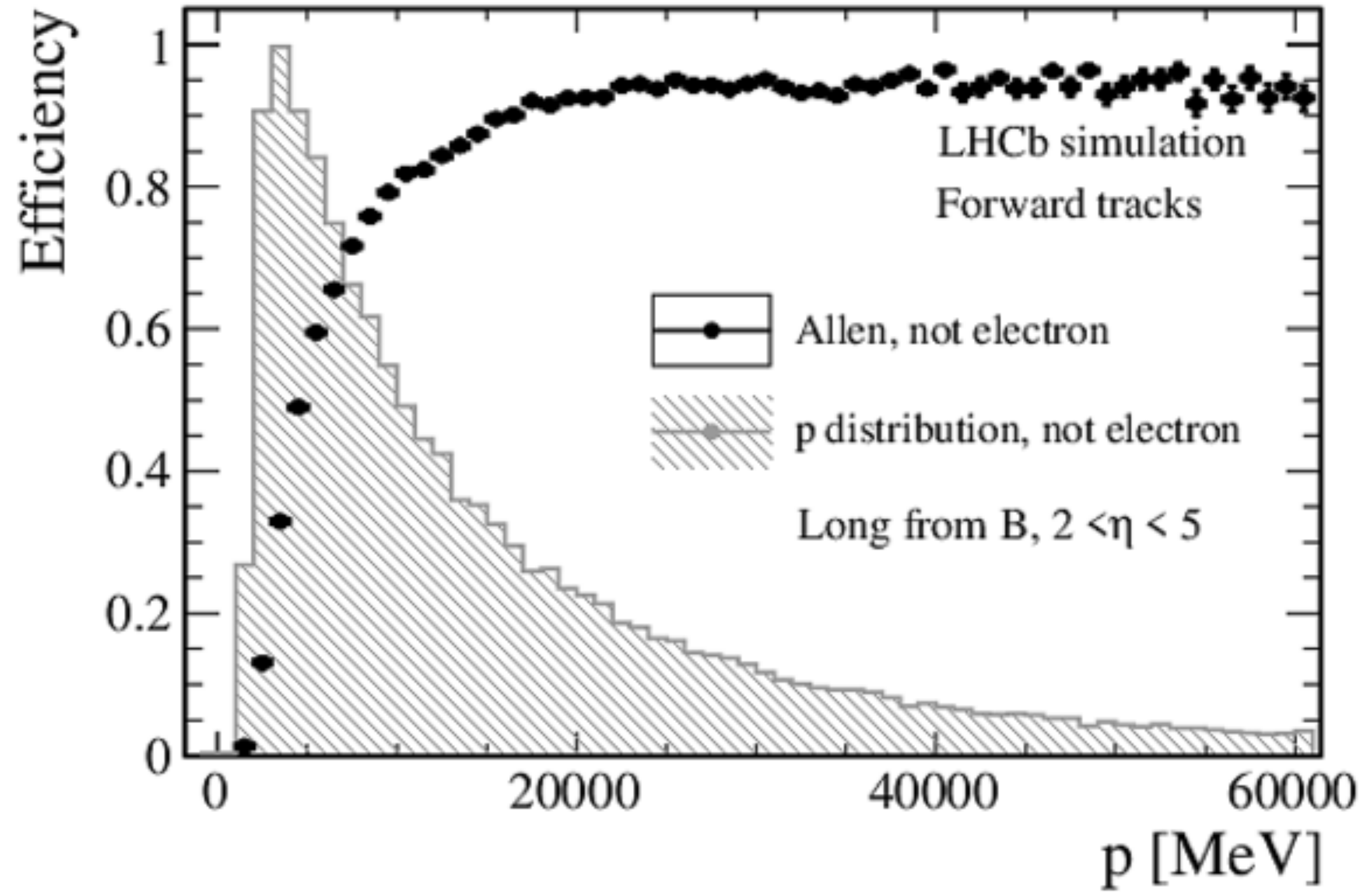
HLT1 Tracking Performance

LHCb-Figure-2020-014

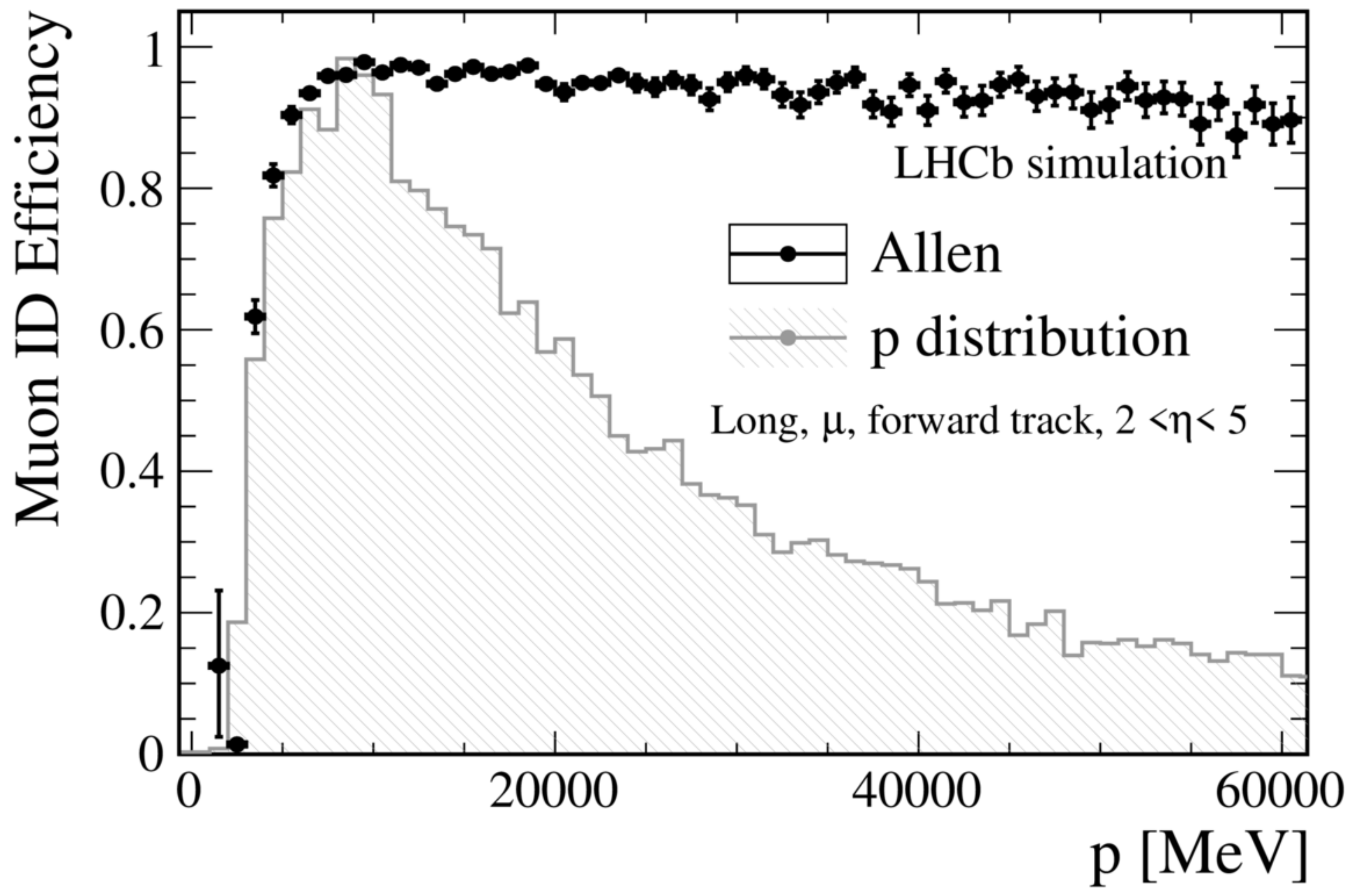
VELO efficiency (>99%)

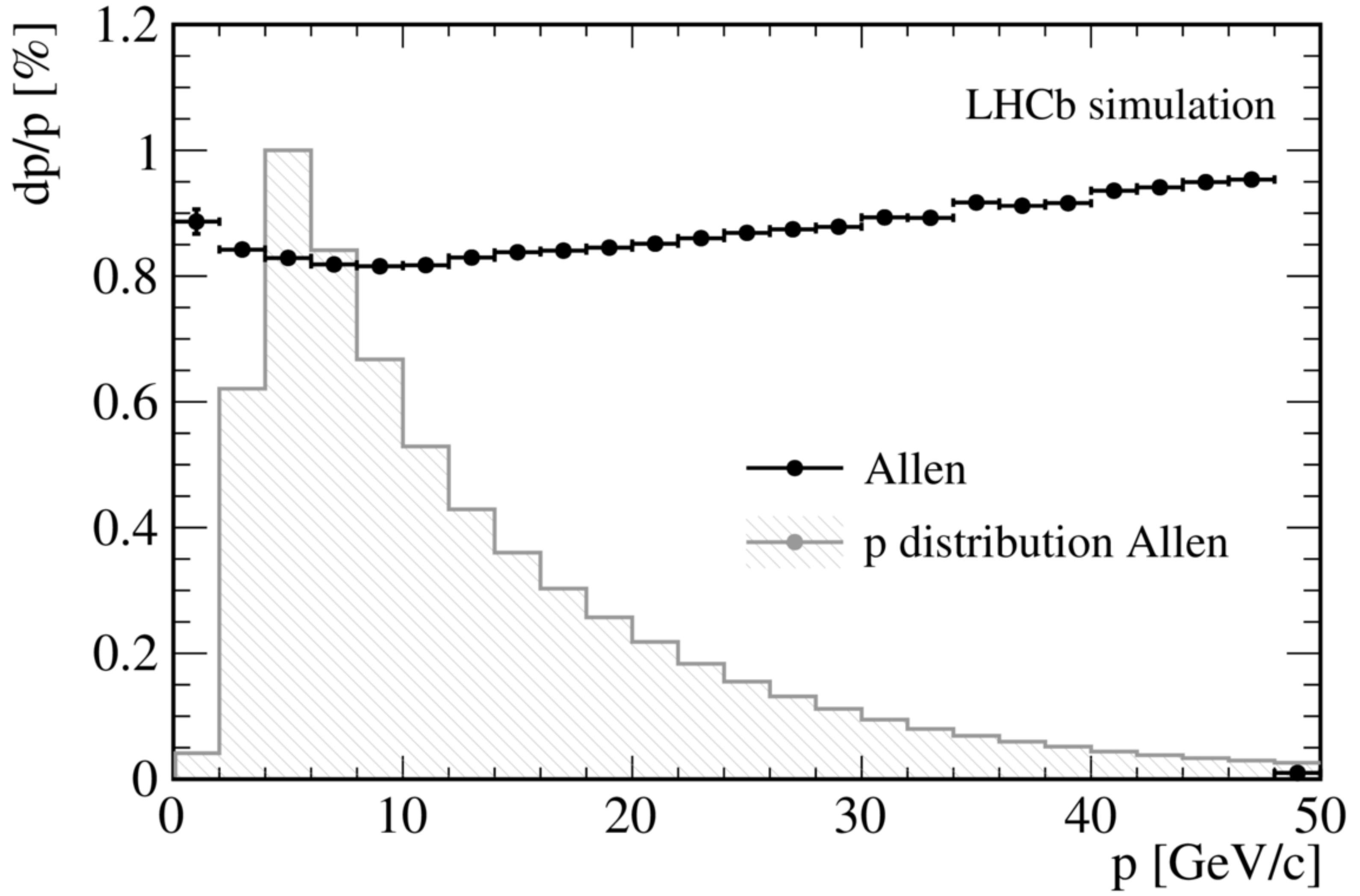


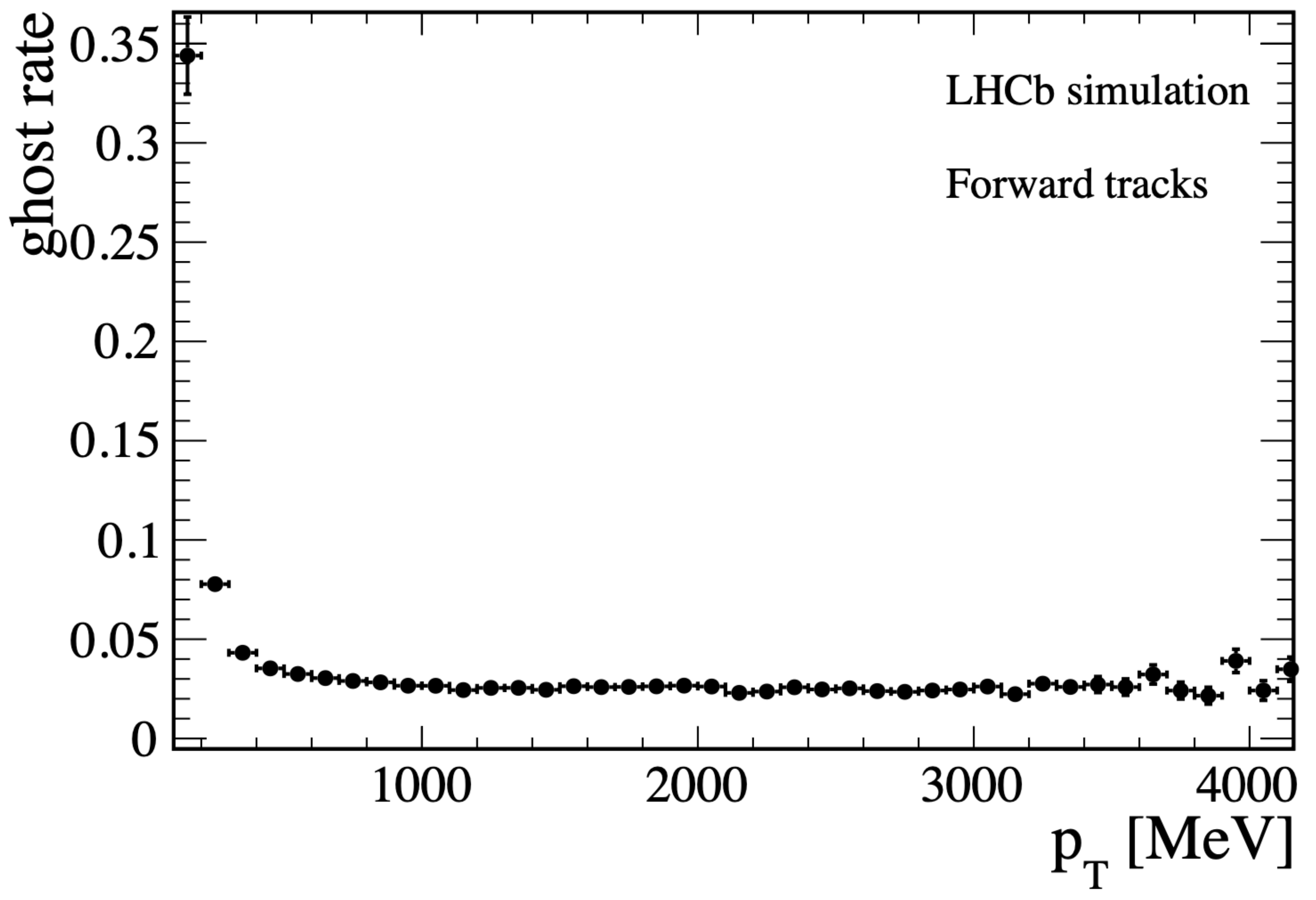
VELO-UT efficiency



Forward (long track) efficiency

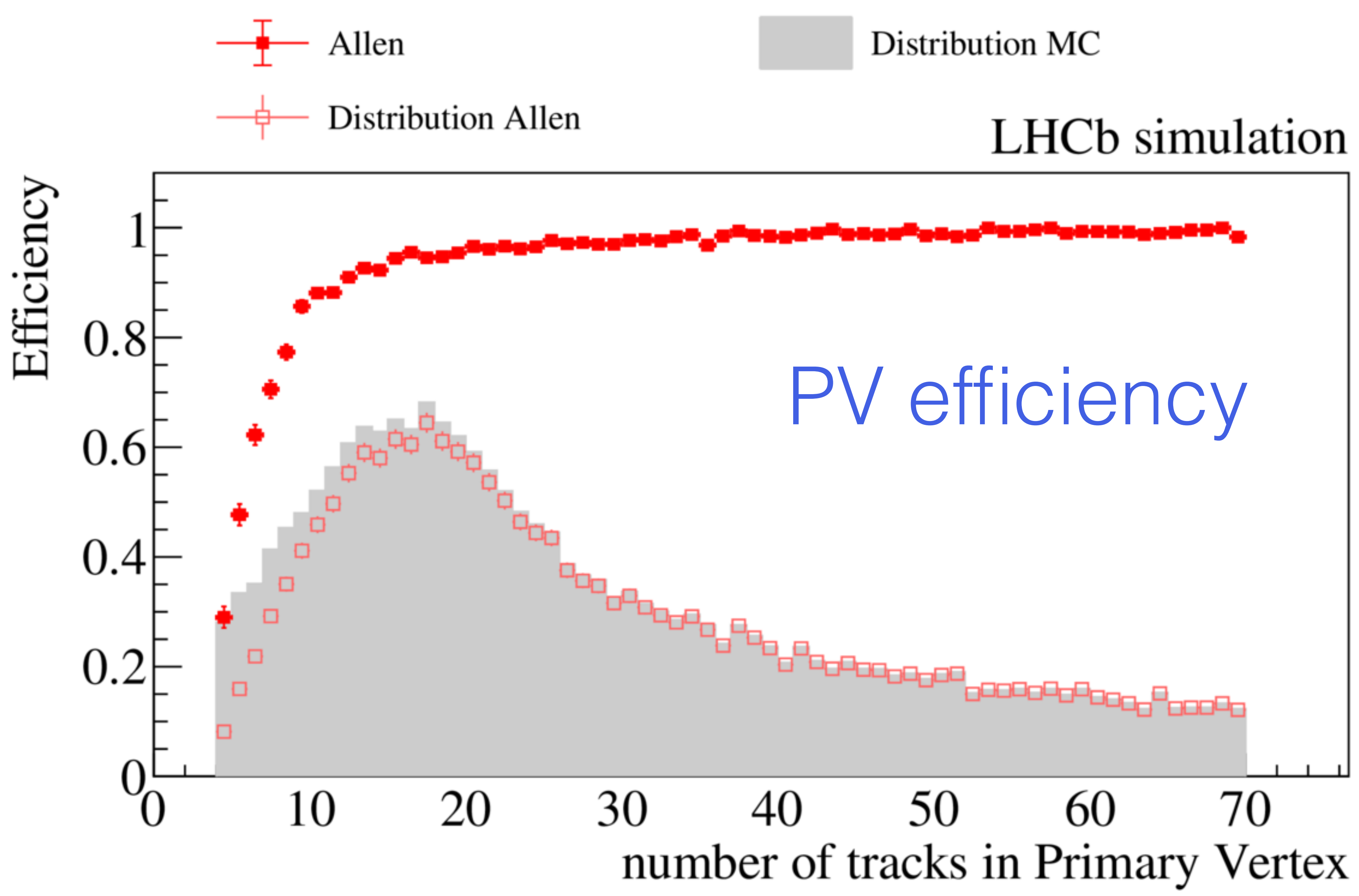




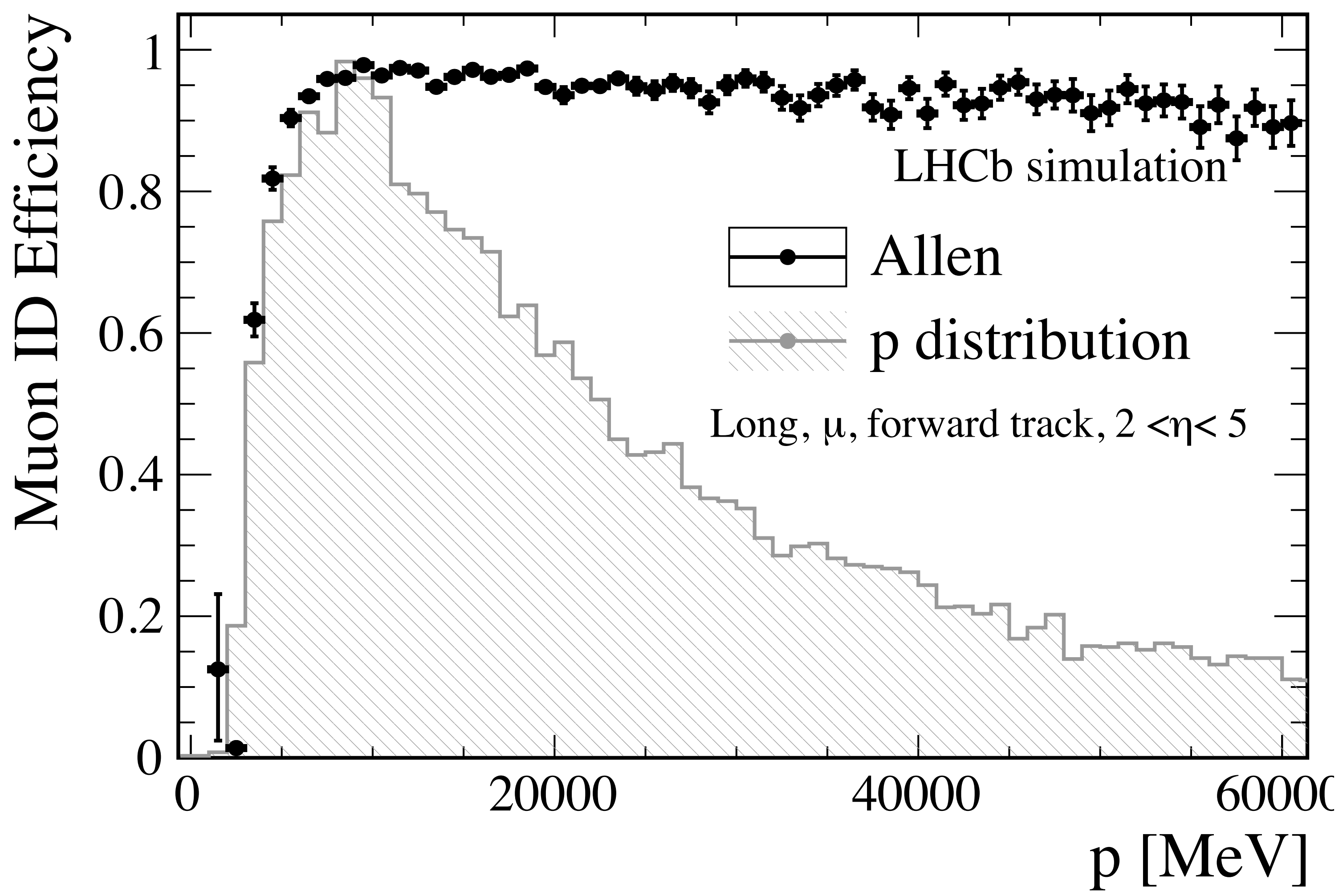


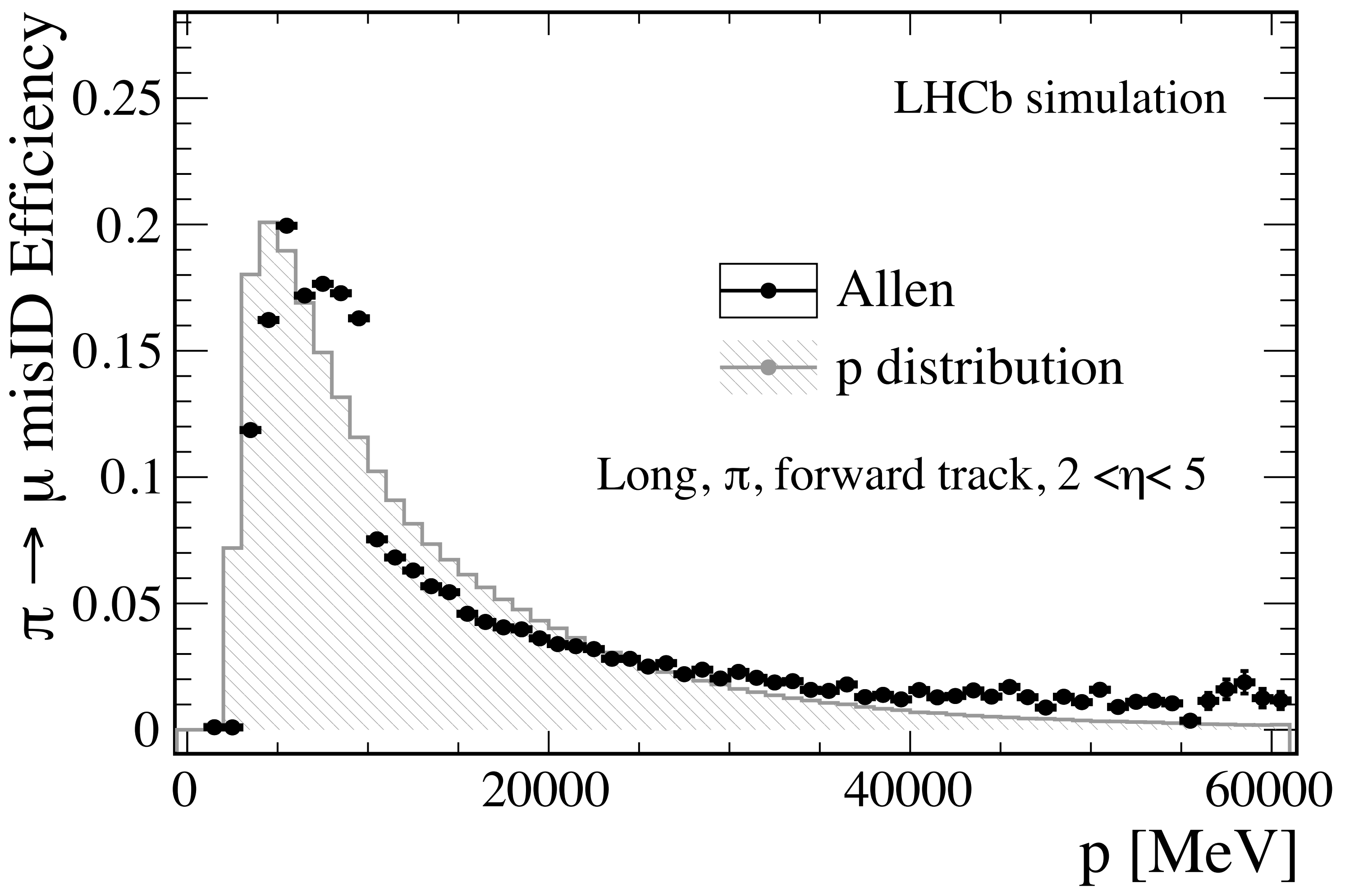
- About 95% long track efficiency about p>5 GeV
- Fake rate < 5% with p_T>250 MeV
- Momentum resolution < 1%

HLT1 Performance



LHCb-Figure-2020-014



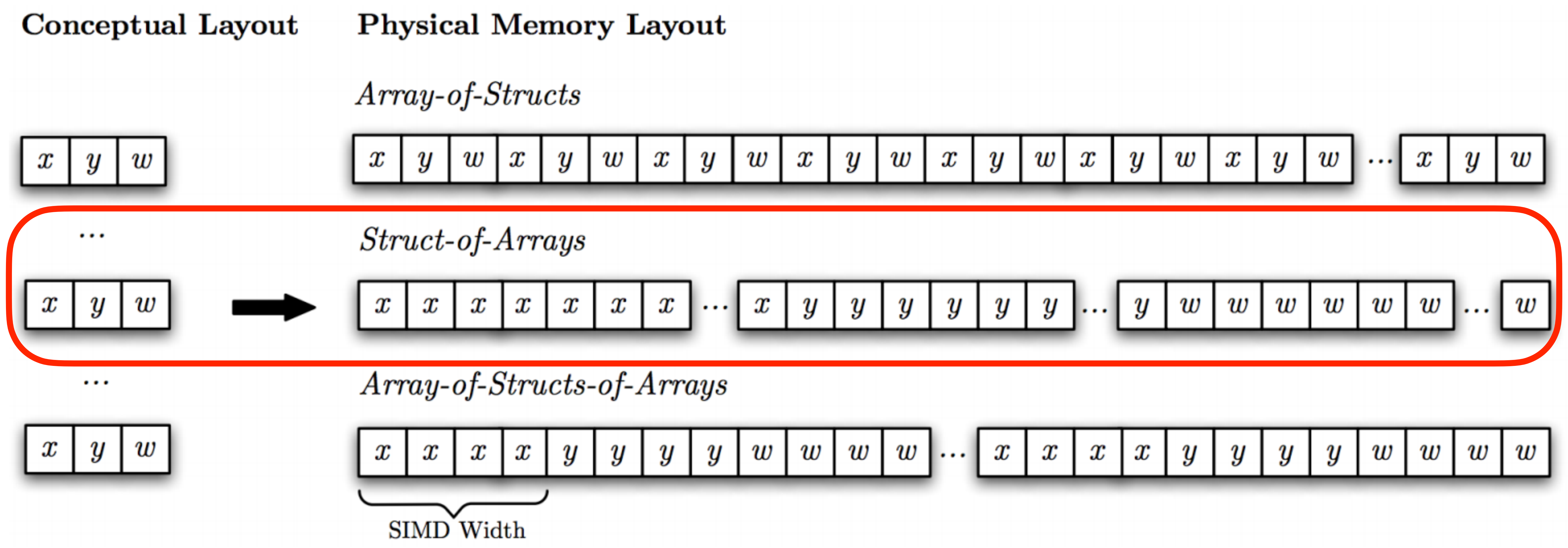


- More than 90% PV reconstruction efficiency with number of tracks larger than 10
- More than 95% Muon identification
- About 2-3% $\pi \rightarrow \mu$ misidentification when momentum > 20 GeV

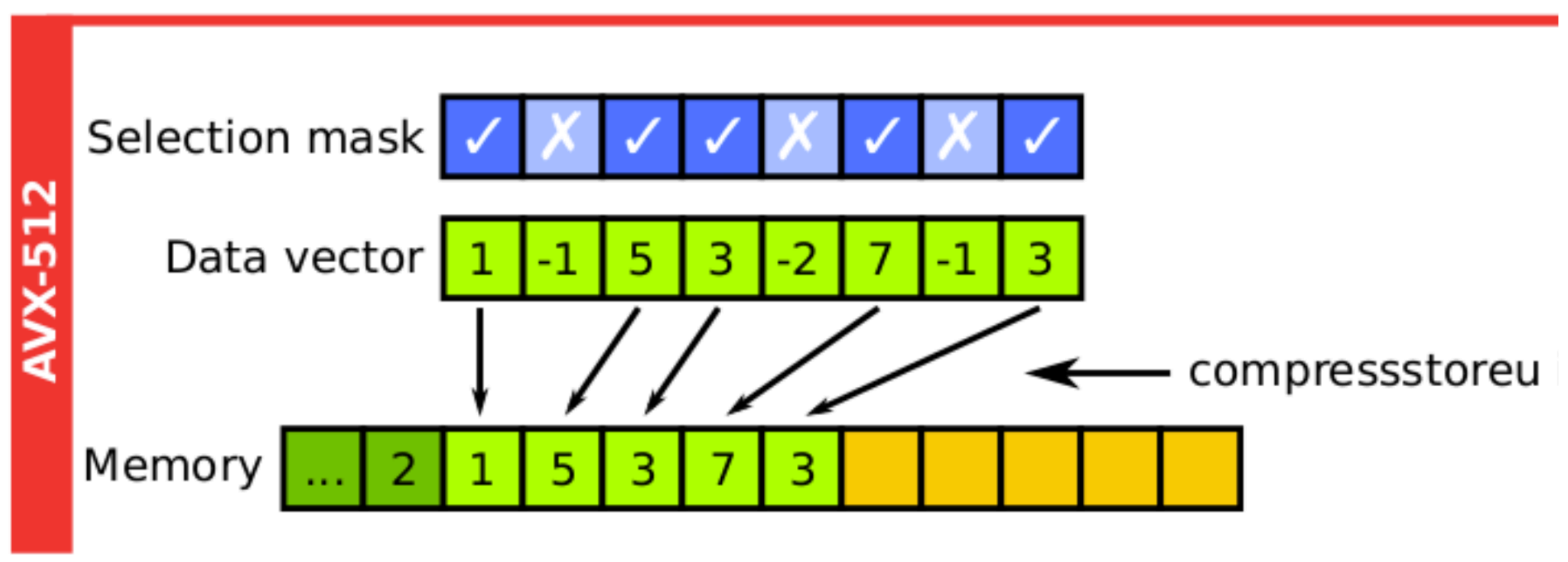
Full track reconstruction with CPU (HLT2)

- HLT2 reconstruction is critical to both physics output and physics quality
  - → Full, offline-fidelity event reconstruction on at least 1 MHz
  - → Charged track reconstruction with full momentum range & deliberate Kalman fit

- Common intra-event parallelisation techniques as in GPU



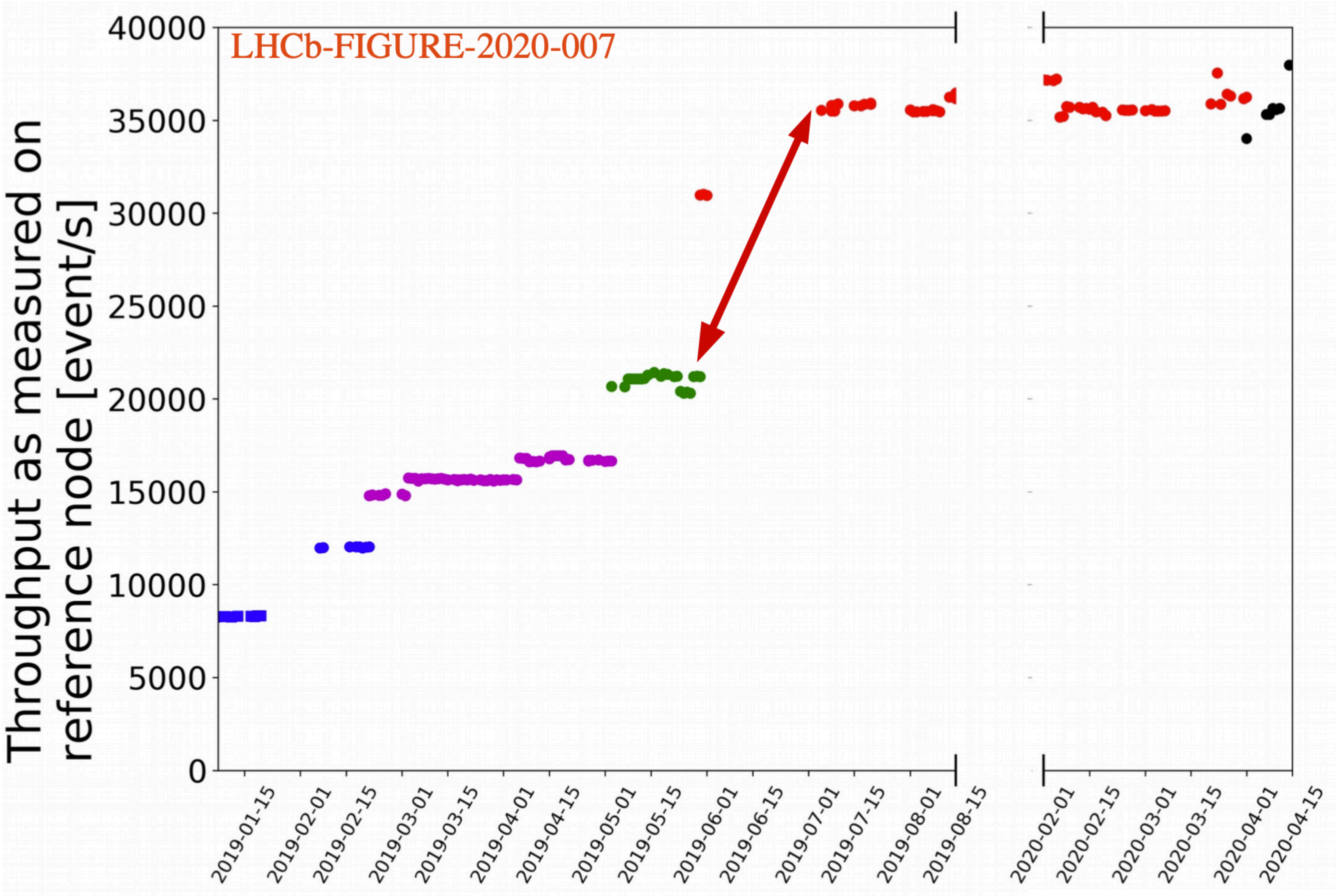
arXiv:1912.09901



- Rewrote all reconstruction algorithms with SOA structure
- Developed custom SIMD wrappers to support all the backends (SSE, AVX2..)

Full track reconstruction with CPU (HLT2)

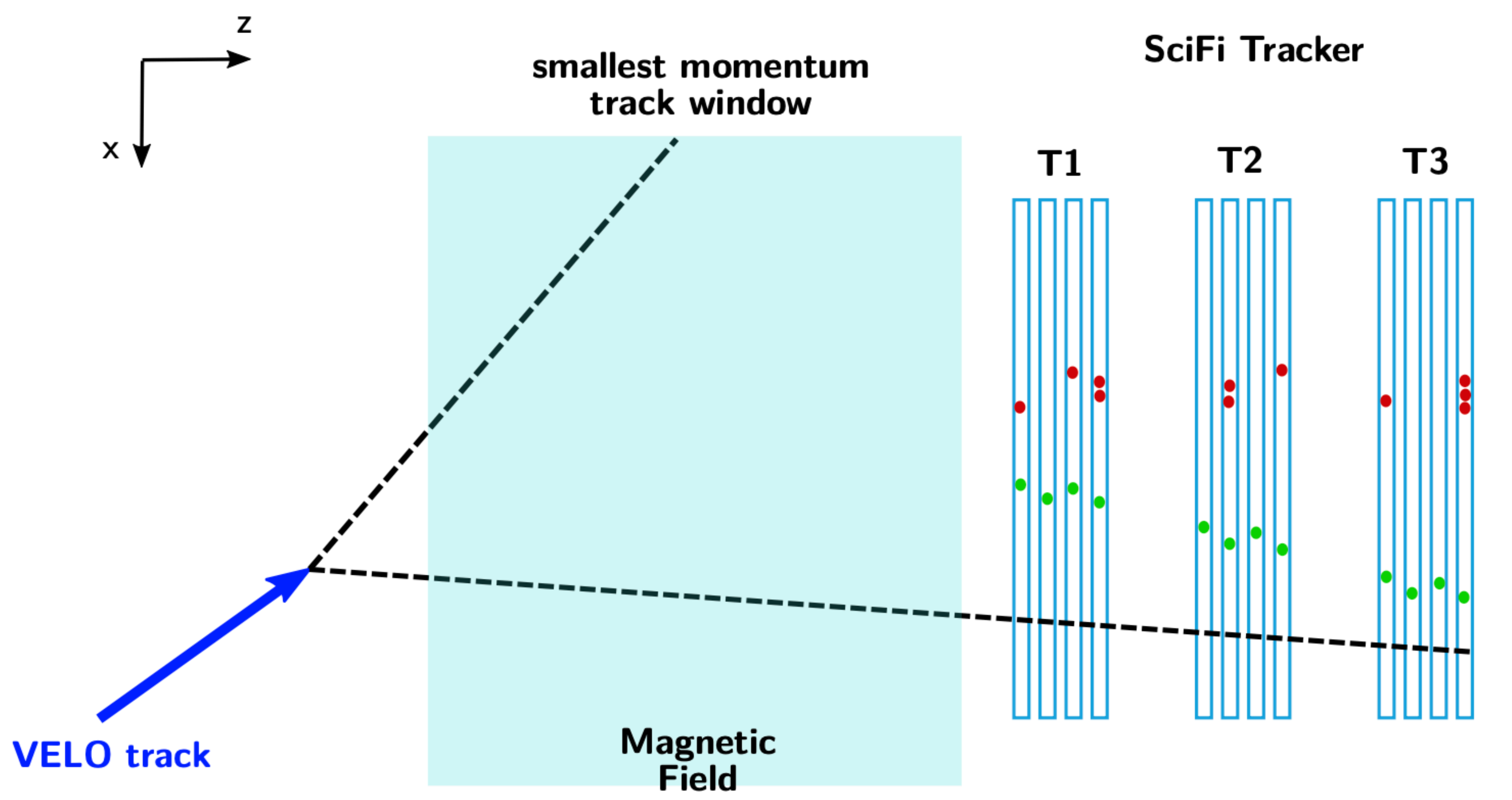
- Significantly speed up the reconstruction in HLT2



Forward Tracking in HLT2

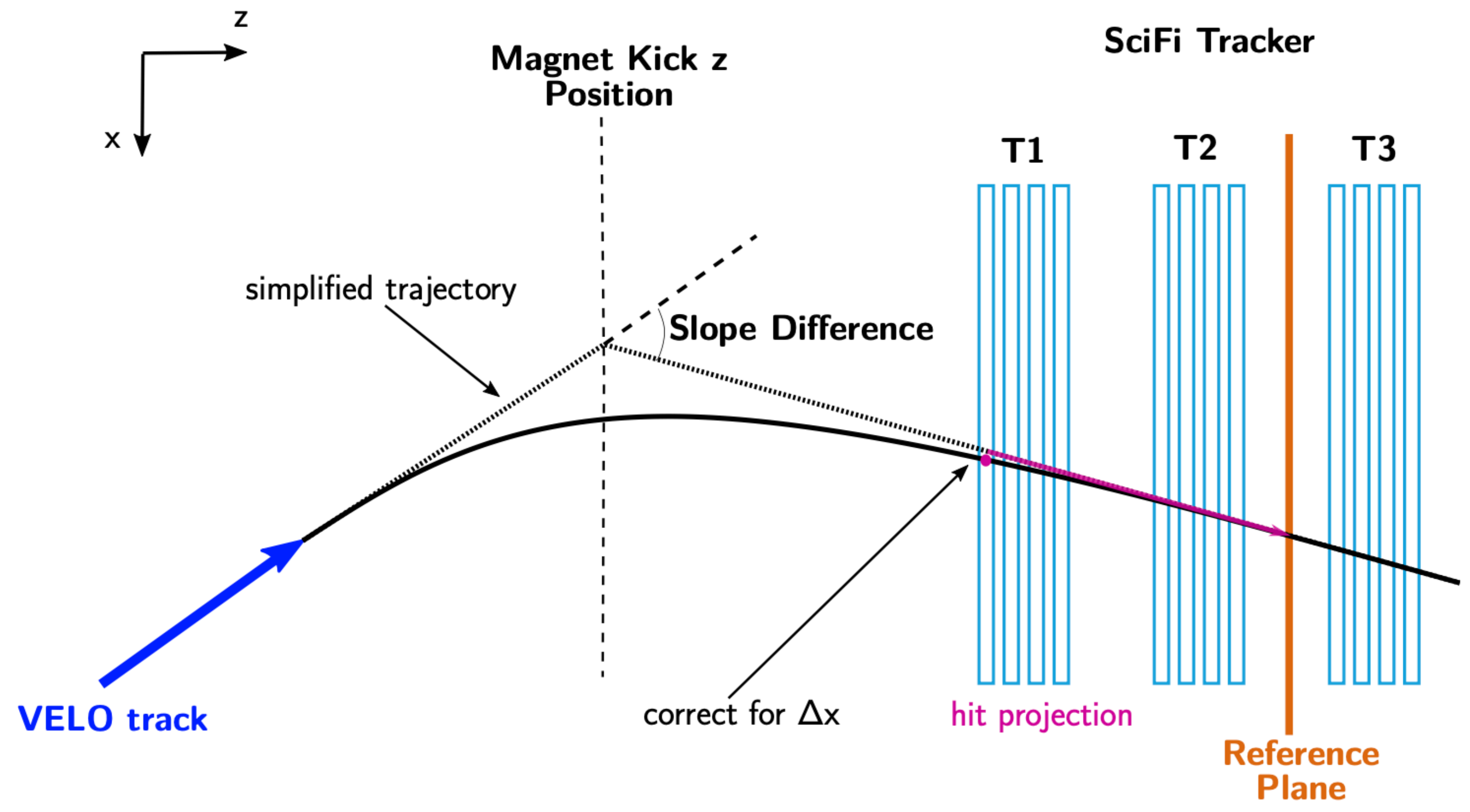
PrForwardTracking

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 $p > 1.5\,\mathrm{GeV}$ use Polynomial$(\frac{\partial x}{\partial z}, \frac{\partial x}{\partial z}, p)$



*Velo / upstream tracks as input

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

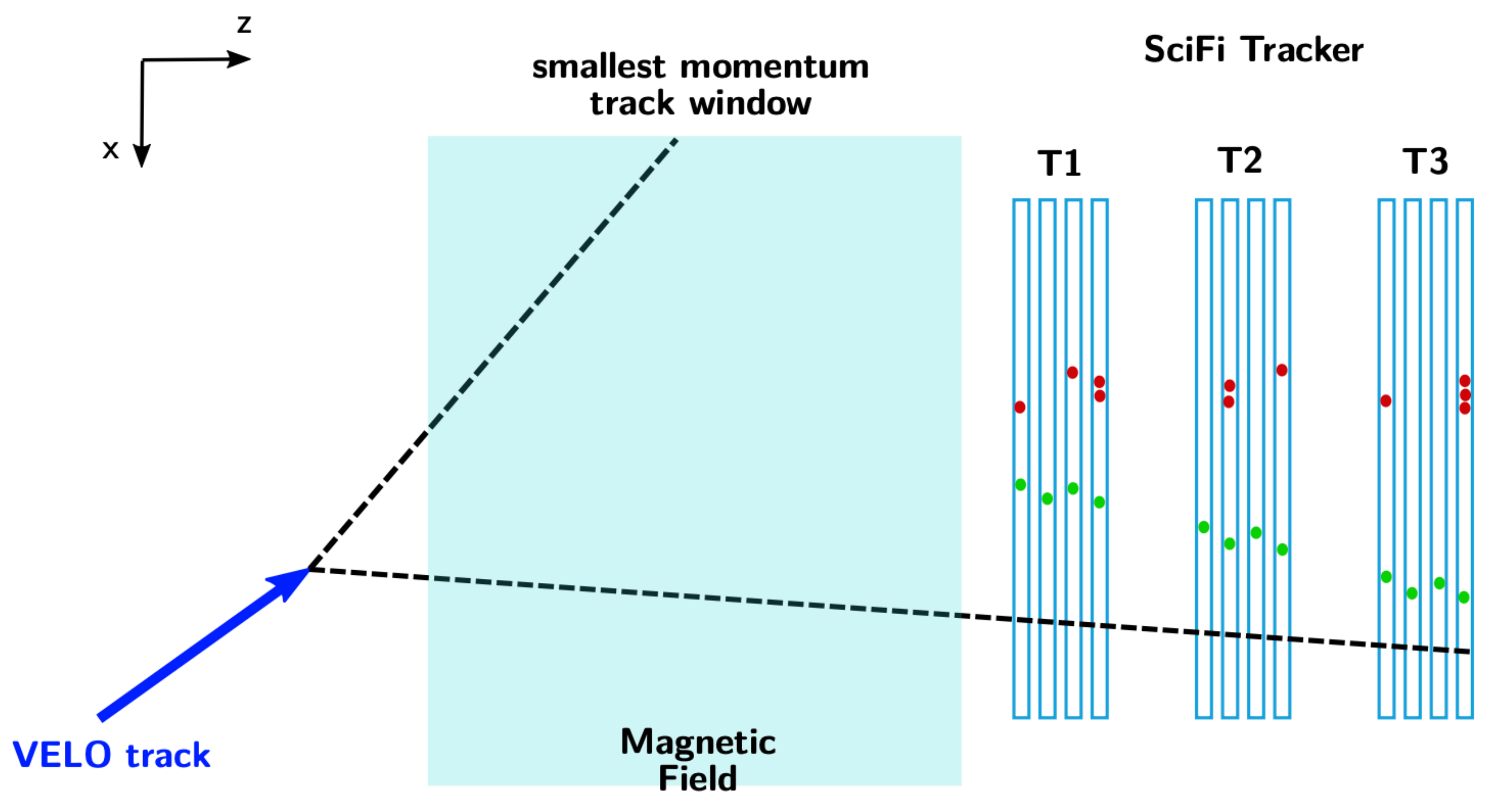


5. clean-up hit set and fit using 3rd order polynomial
6. estimate q/p from fit result

Forward Tracking in HLT2

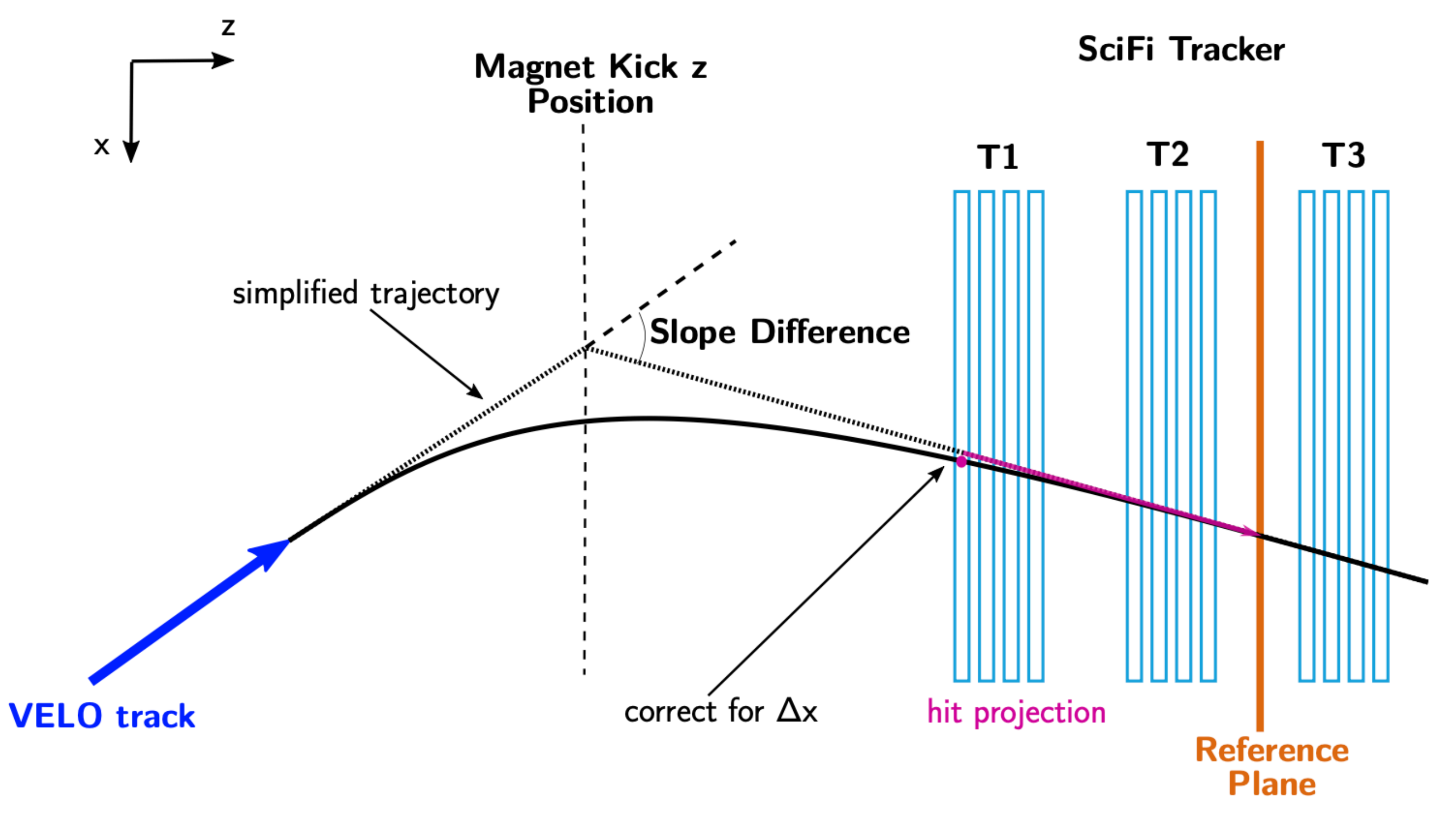
PrForwardTracking

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 $p > 1.5\,\mathrm{GeV}$ use Polynomial$(\frac{\partial x}{\partial z}, \frac{\partial x}{\partial z}, p)$

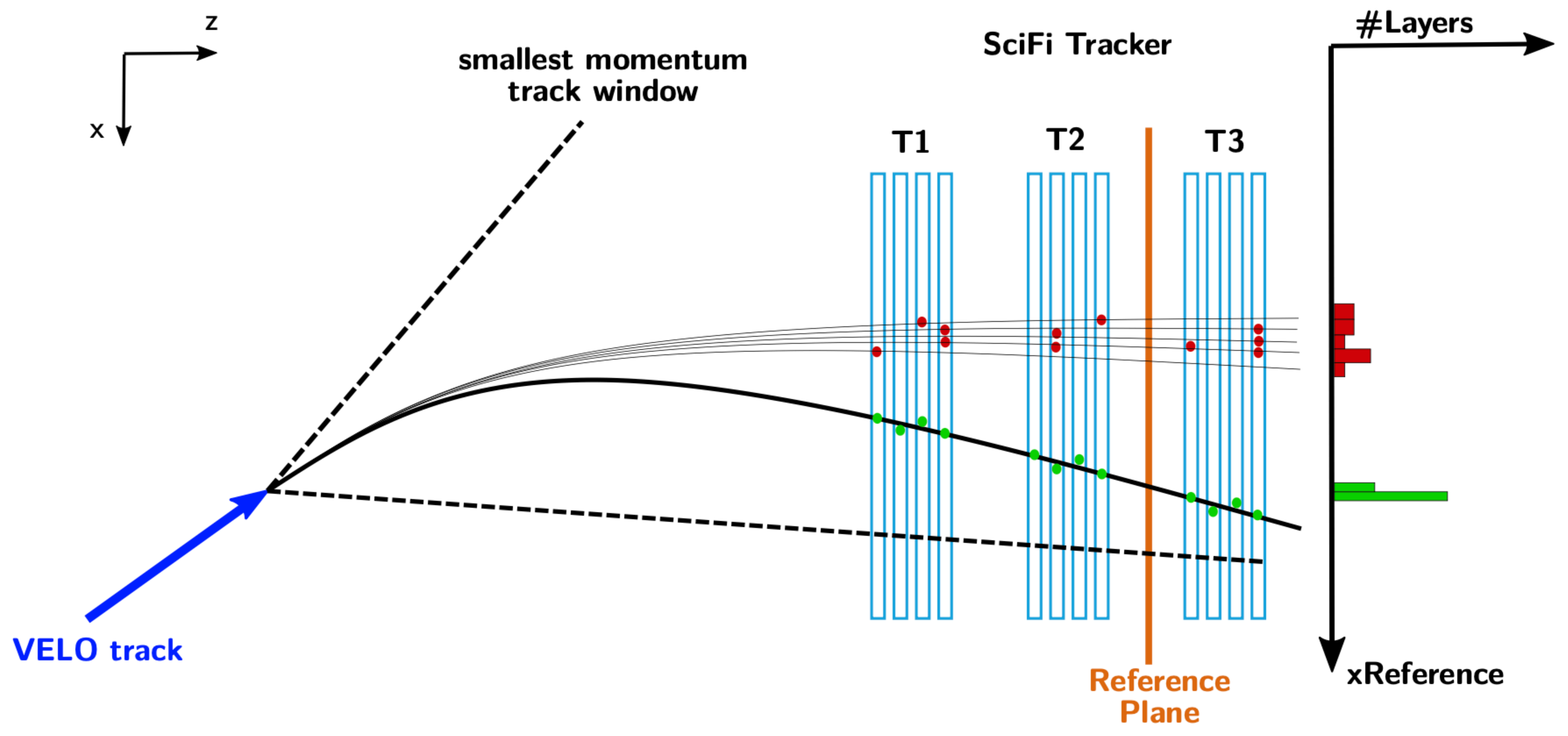


*Velo / upstream tracks as input

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

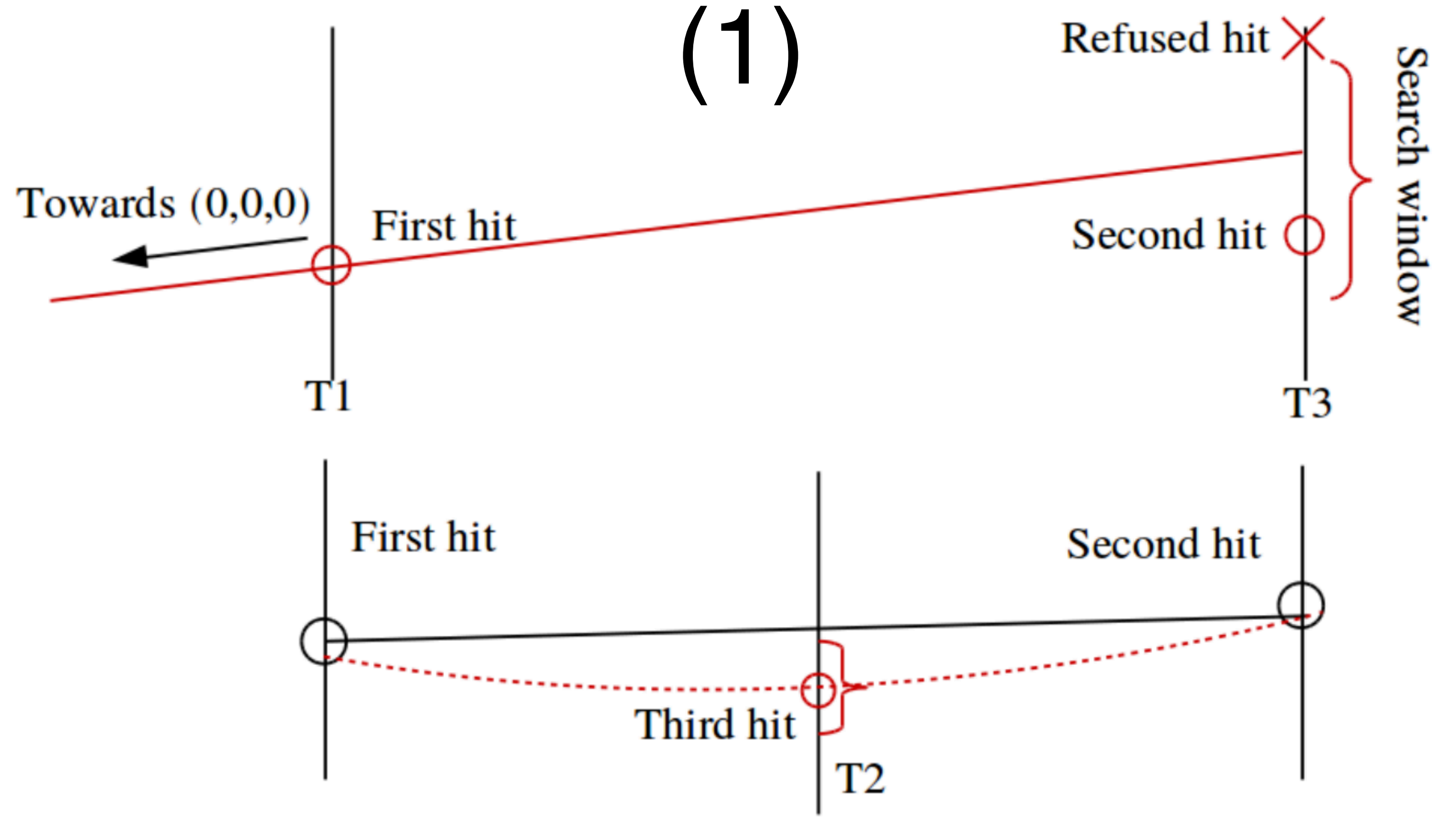
5. clean-up hit set and fit using 3rd order polynomial
6. estimate q/p from fit result

Seed Track reconstruction

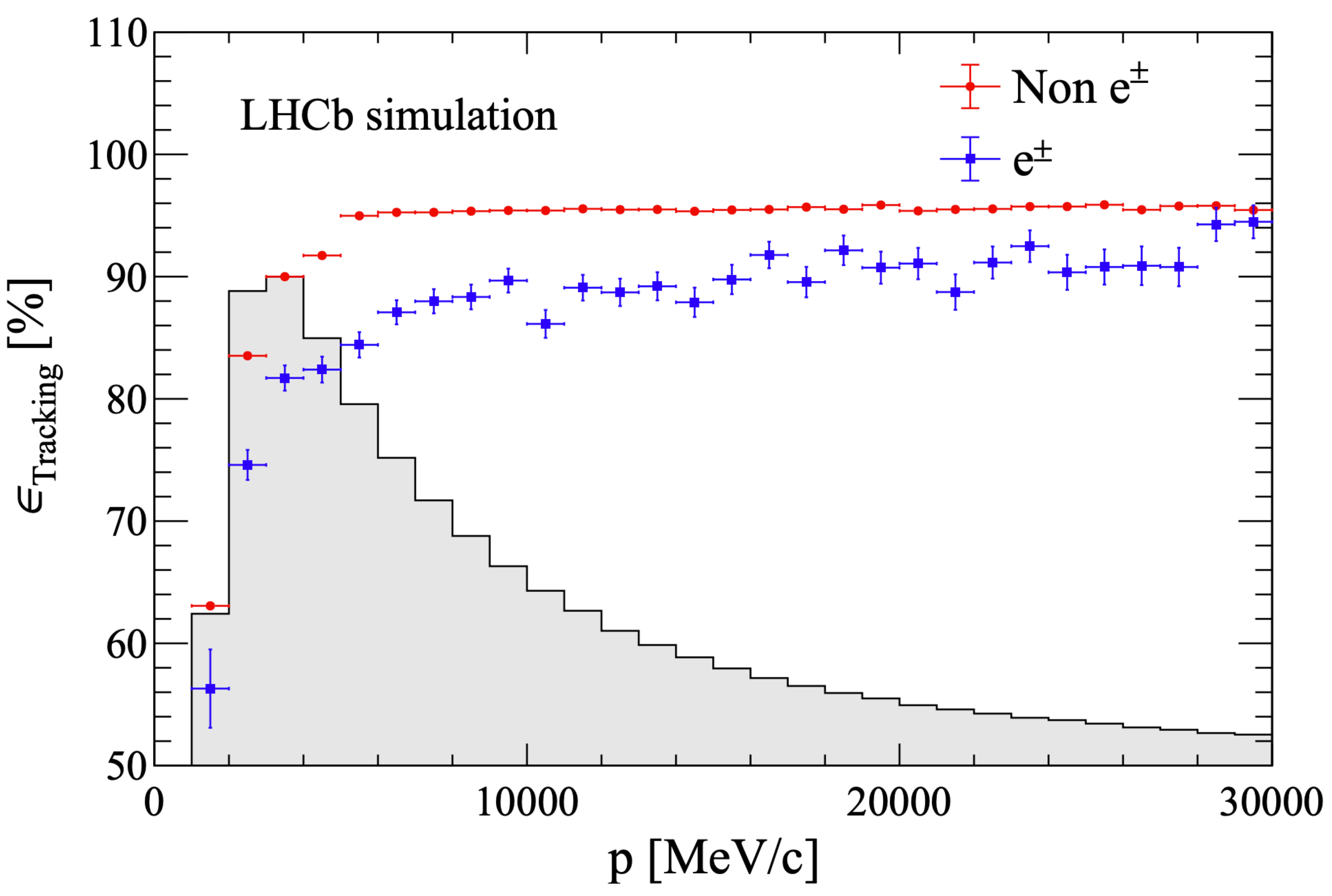
arXiv:2007.02591

- Standalone algorithm: using SciFi hits only
  - Input tracks to Matching & Downstream tracking
  - (1) Seeding_XZ track: look at 6 X layers only
  - (2) Seeding_Track: add u/v layers by fitting to the y(z) with at least 4 hits

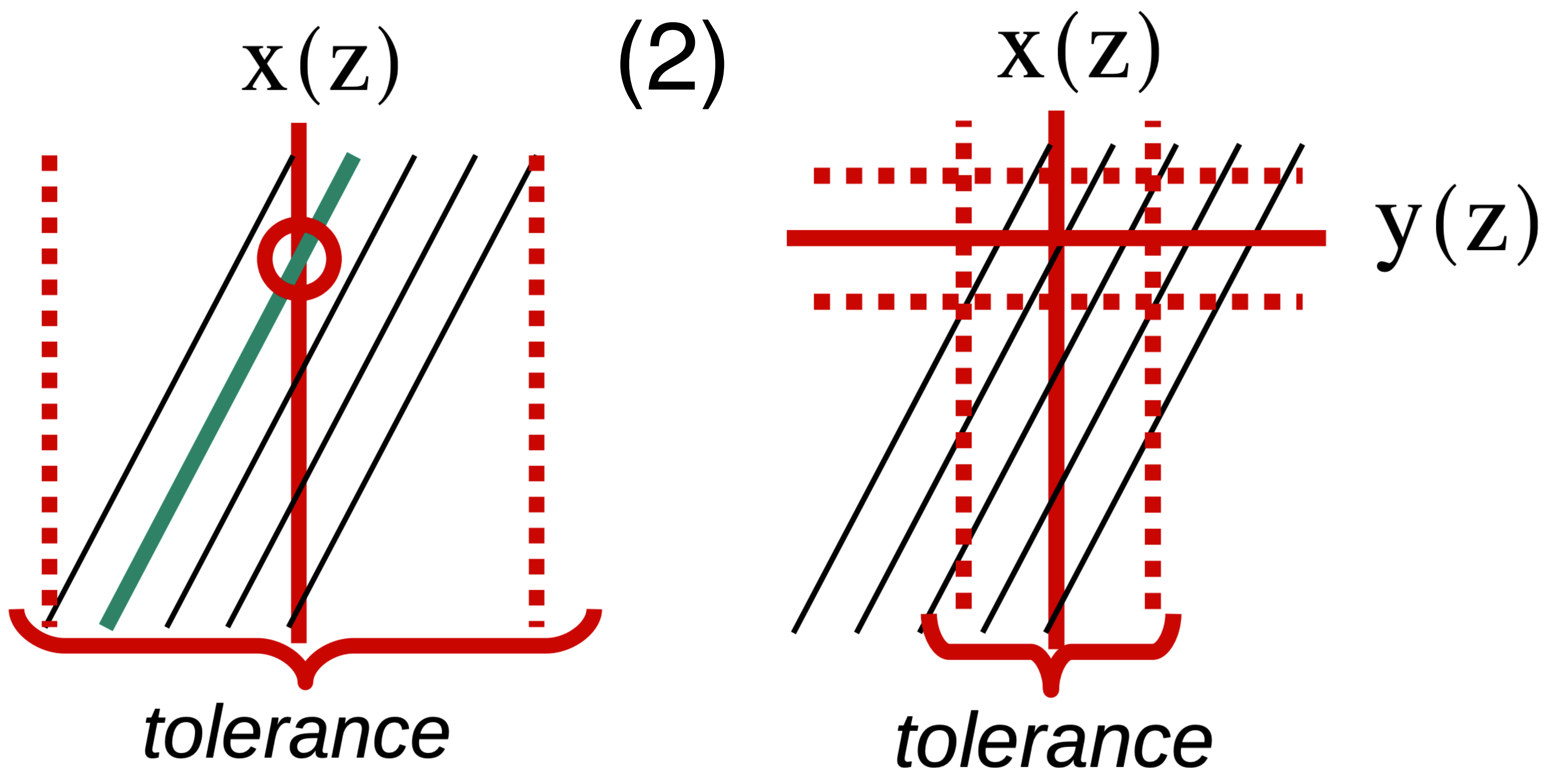
(1)



Iteration	1	2	3
First x-layer	T1x1	T1x2	T1x1
Second x-layer	T3x2	T3x1	T3x1
Minimum momentum	5 GeV/c	2 GeV/c	1.5 GeV/c

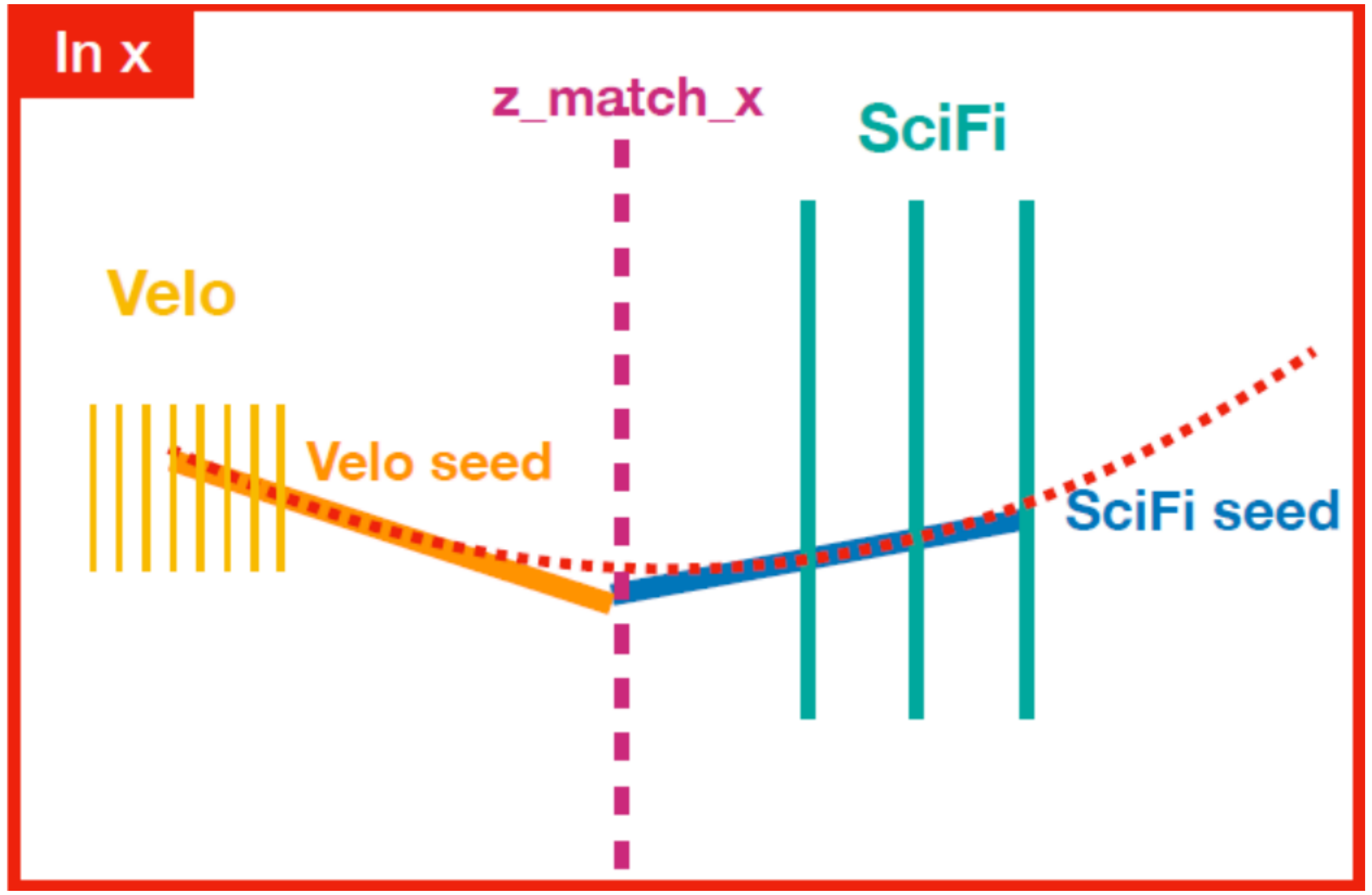


(2)

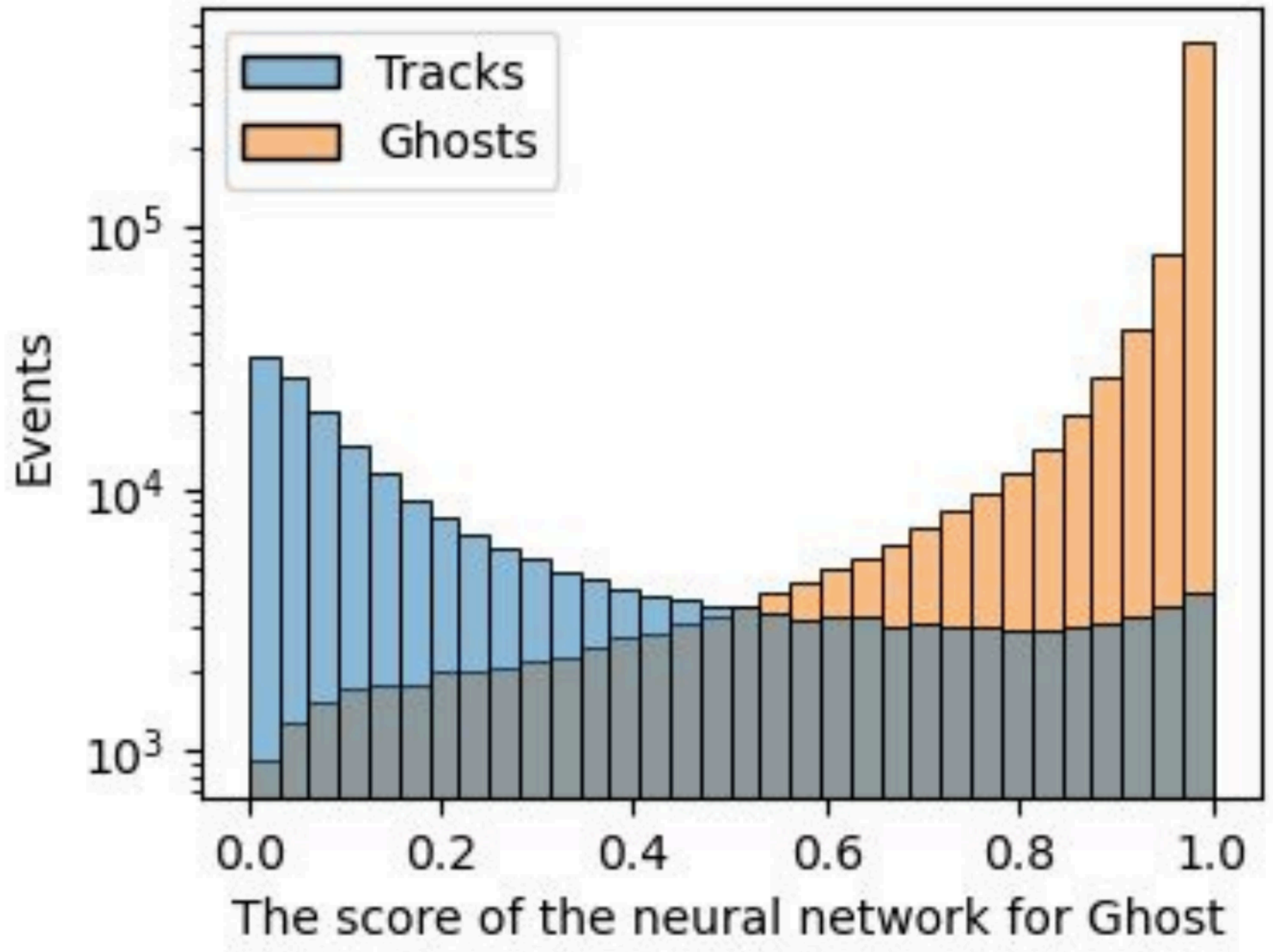


Matching & Downstream tracking

- Matching algorithm: VELO + Seed tracks
  - Second Long track reconstruction alg.
  - Together with Forward Tracking to maximise the Long track reconstruction efficiency



- Downstream tracking: SciFi tracks + UT hits
  - Important for the Long-lived particles, e.g.$K_S^0,\ \Lambda^0$
  - Neural-Network based classifier to reduce fakes



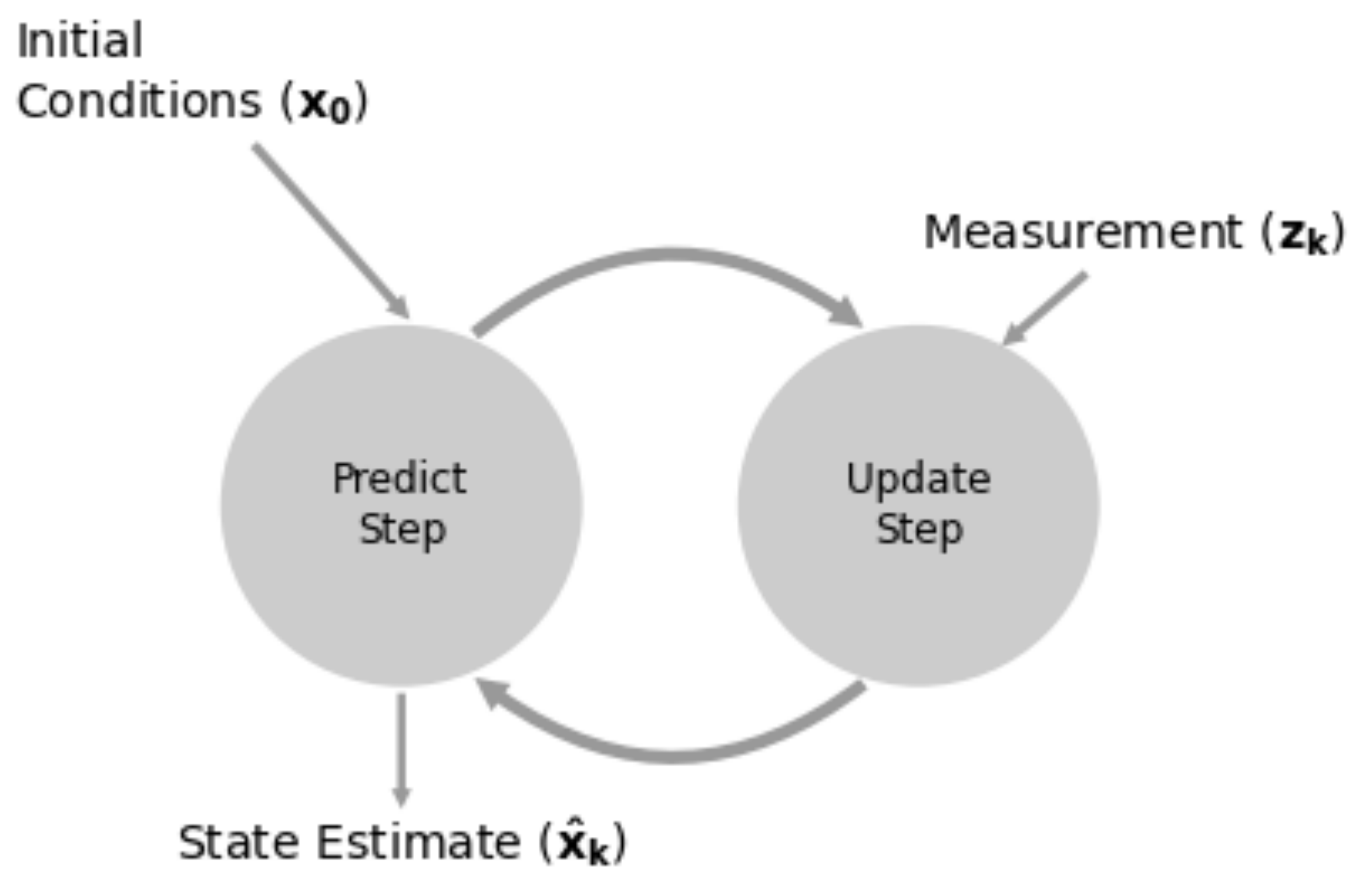
- Both Matching & Downstream tracking are now also implemented in GPU!

HLT2 Kalman fit

- Determine the most accurate estimates of the track parameters together with the corresponding covariances
  - Track state at given z: $\overrightarrow{x} = (x, y, t_x, t_y, q/p)^{\mathrm{T}},\ t_x = \dfrac{\partial x}{\partial z},\ t_y = \dfrac{\partial y}{\partial z}$
  - Track state + measurement = node (stage for Kalman Filter)
  - Principle: add measurements one-by-one and each time update the track state at the particular node

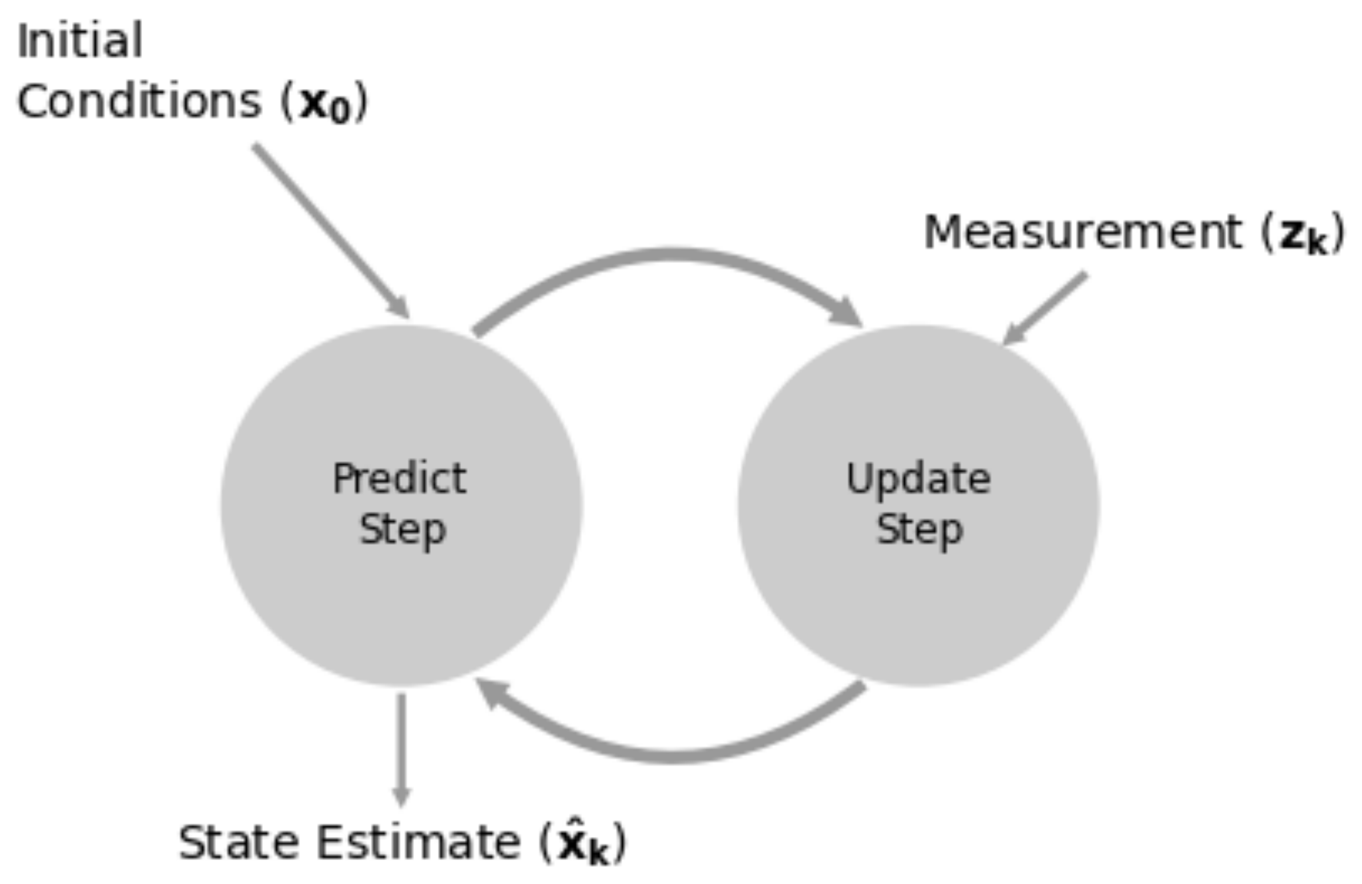
Prediction (Runge-Kutta extrapolator) $\rightarrow$ Parameterisation

- $\overrightarrow{x}_k^{k-1} = F_k(\overrightarrow{x}_{k-1}),\ C_k^{k-1} = F_k C_{k-1} F_k^T + Q_k$
- Noise - zero mean multivariate normal distribution with covariant matrix Q_k for including multiple scattering



HLT2 Kalman fit

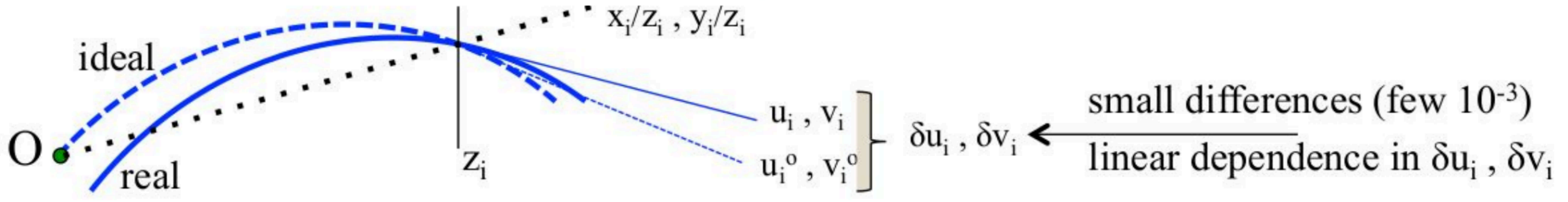
- Determine the most accurate estimates of the track parameters together with the corresponding covariances
  - Track state at given z: $\vec{x} = (x, y, t_x, t_y, q/p)^{\mathrm{T}},\ t_x = \frac{\partial x}{\partial z},\ t_y = \frac{\partial y}{\partial z}$
  - Track state + measurement = node (stage for Kalman Filter)
  - Principle: add measurements one-by-one and each time update the track state at the particular node



Prediction (Runge-Kutta extrapolator) $\rightarrow$ Parameterisation

- $\vec{x}_k^{k-1} = F_k(\vec{x}_{k-1}),\ C_k^{k-1} = F_k C_{k-1} F_k^T + Q_k$
- Noise - zero mean multivariate normal distribution with covariant matrix Q_k for including multiple scattering

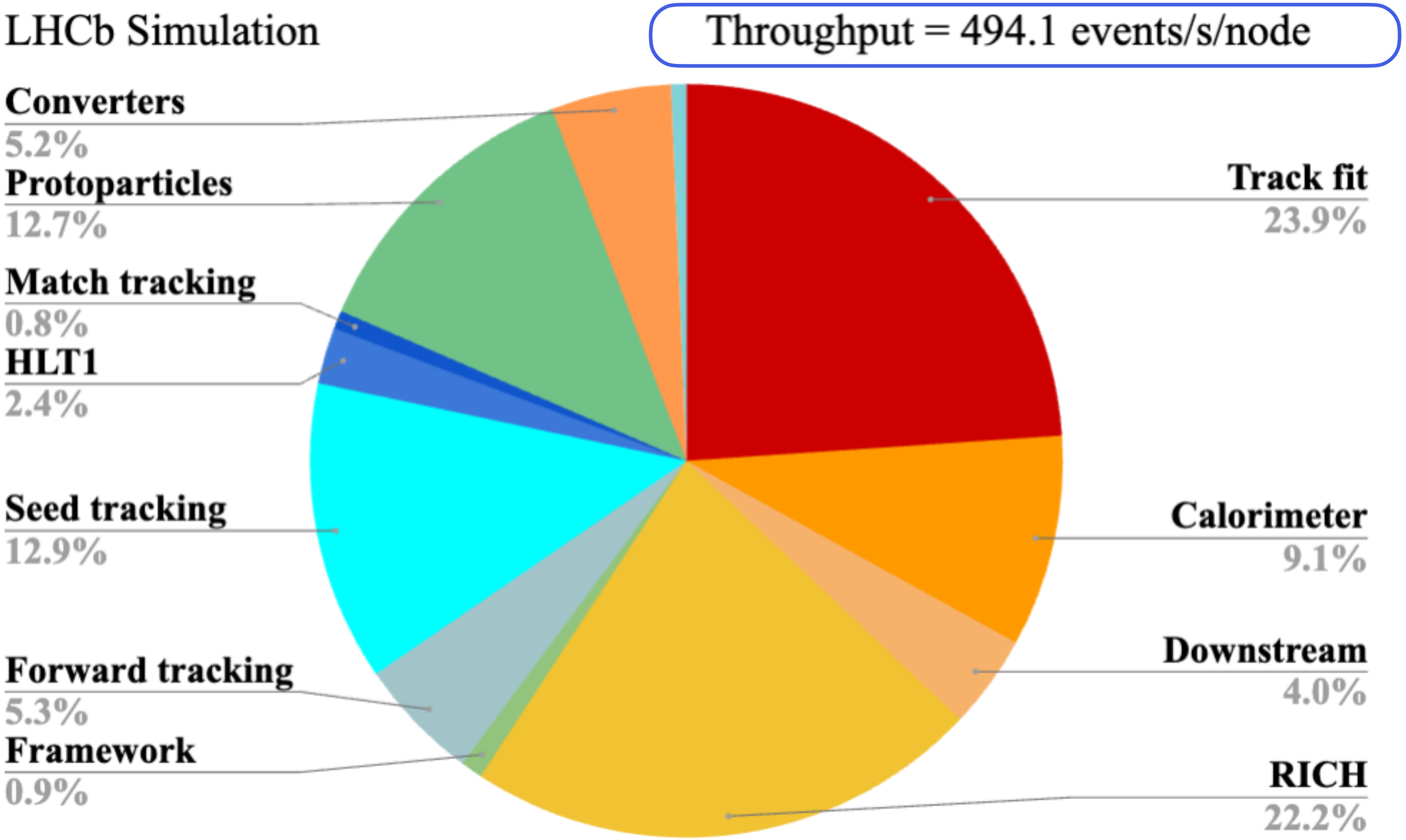
$$\mathbf{f}(\mathbf{x}_i) = \sum_{k=1}^{K_1} \mathbf{A}_k(x_i, y_i) \left(\frac{q}{p}\right)^k + \sum_{k=1}^{K_2} \left(\mathbf{B}_k(x_i, y_i)\,\delta u + \mathbf{C}_k(x_i, y_i)\,\delta v\right) \left(\frac{q}{p}\right)^k$$



Comput. Phys. Commun. 265,108026 (2021)

HLT2 Throughput for reconstruction

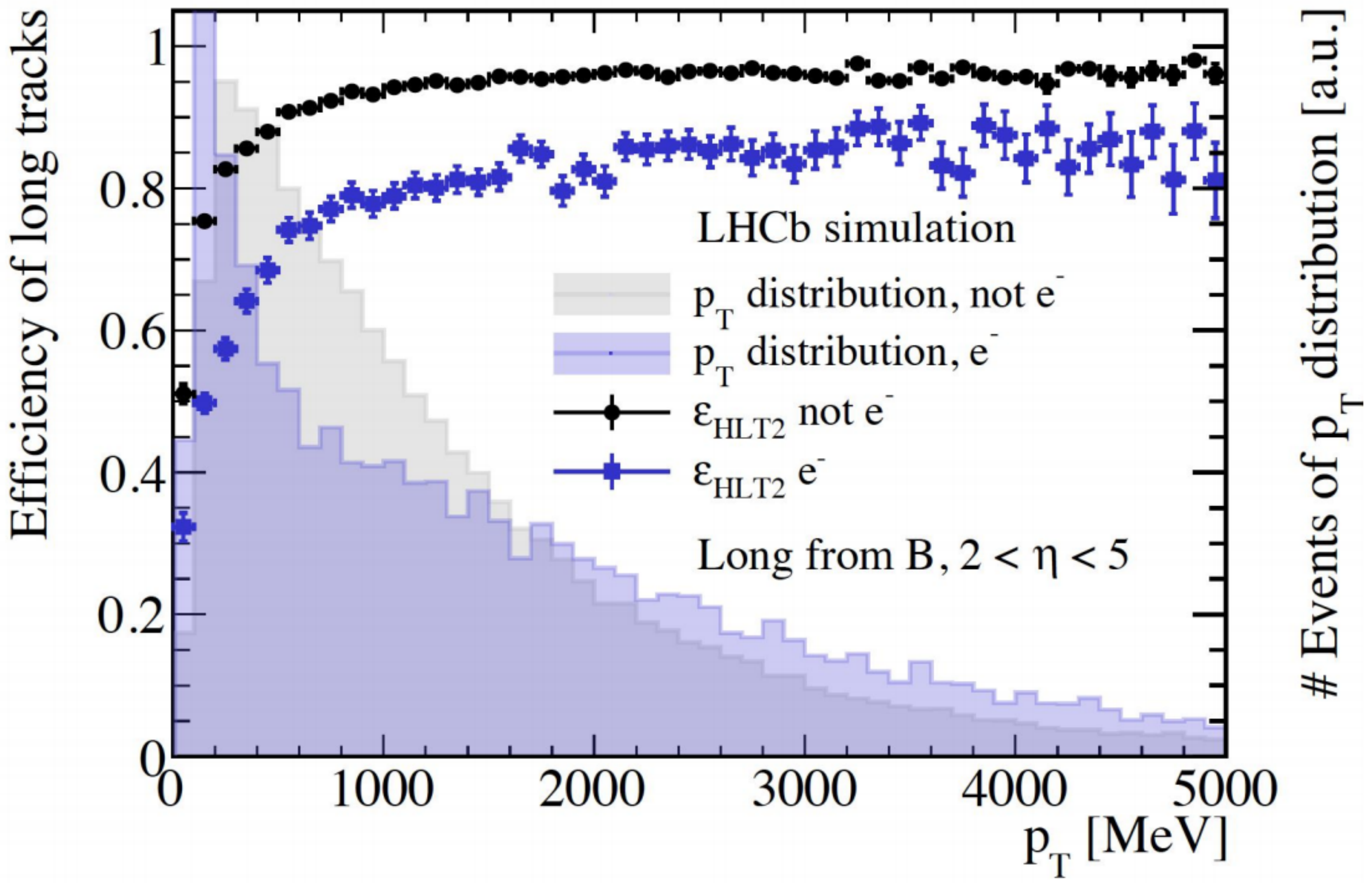
Allowed maximum HLT1 output rate: > 2 MHz

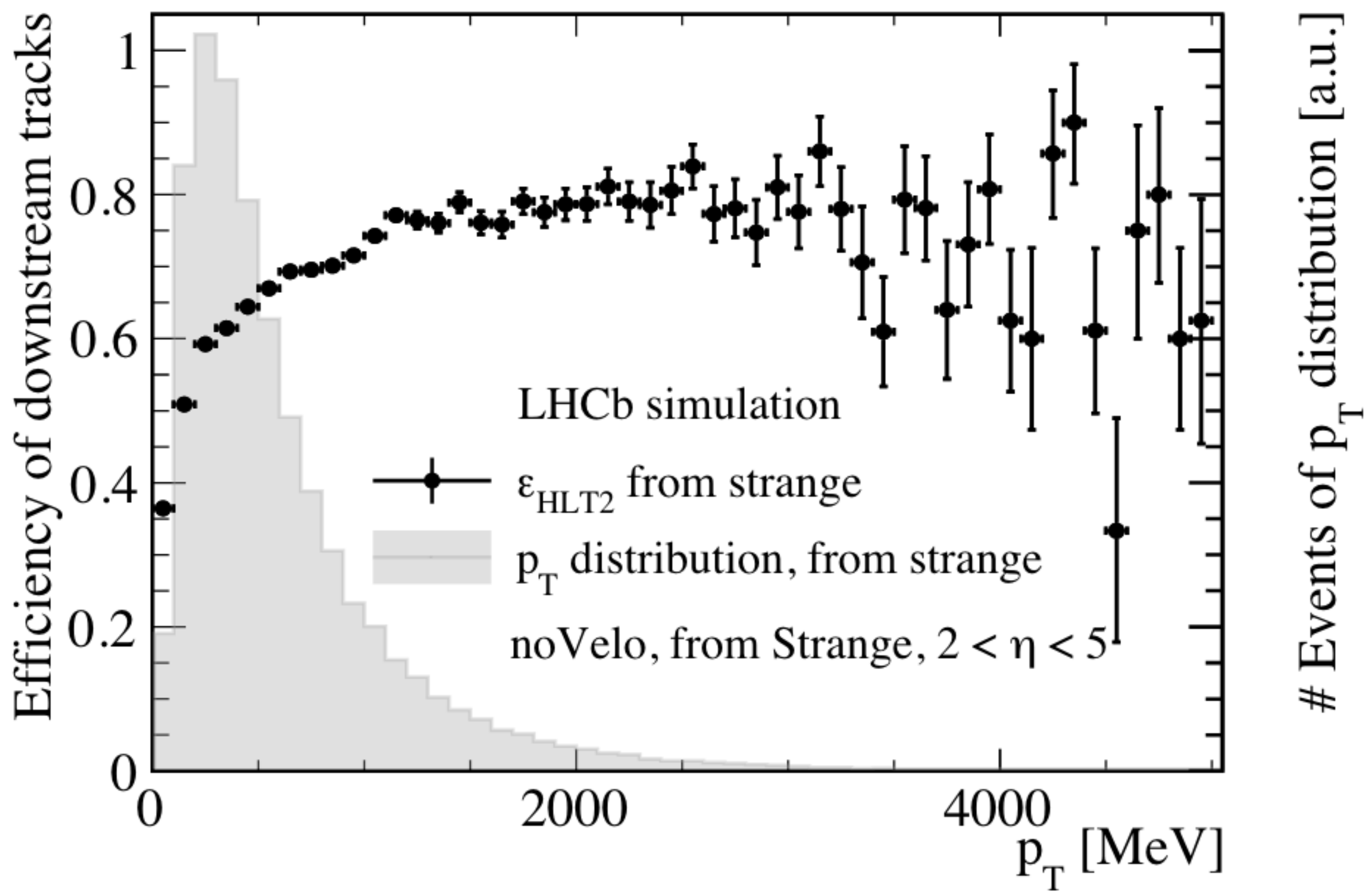


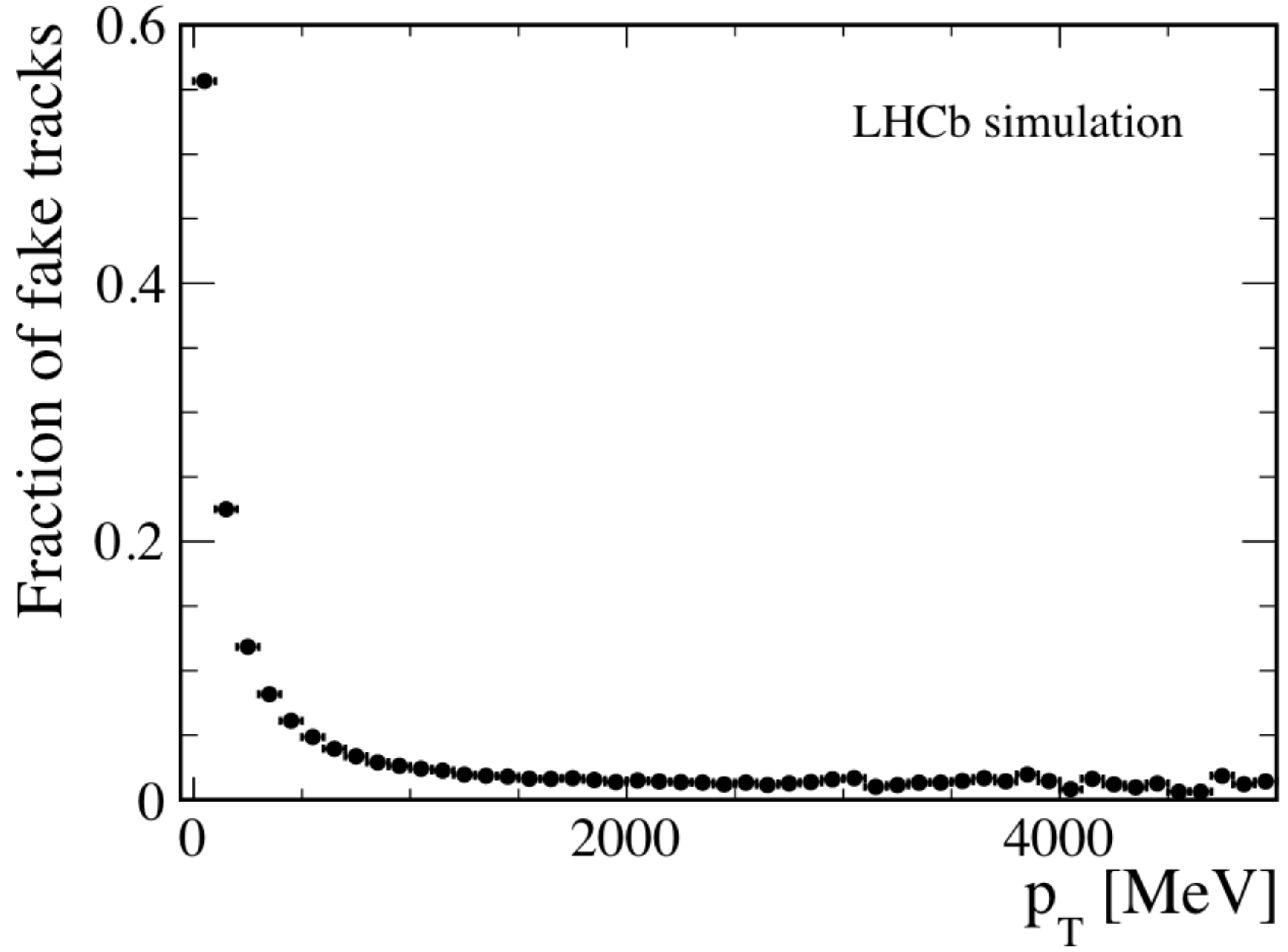
LHCb-FIGURE-2022-005

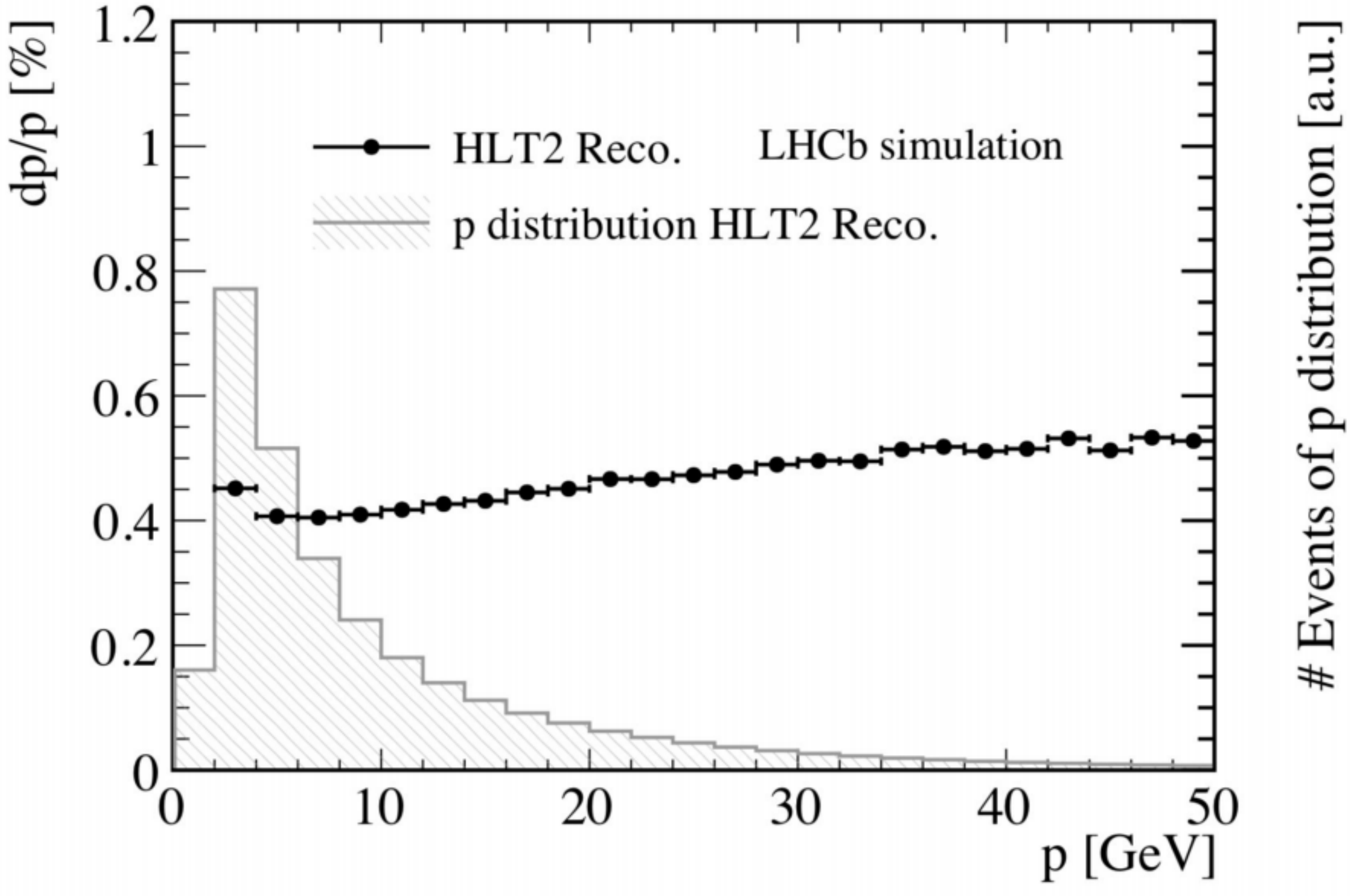
Caveat: no selection and persistency included

HLT2 Tracking performance





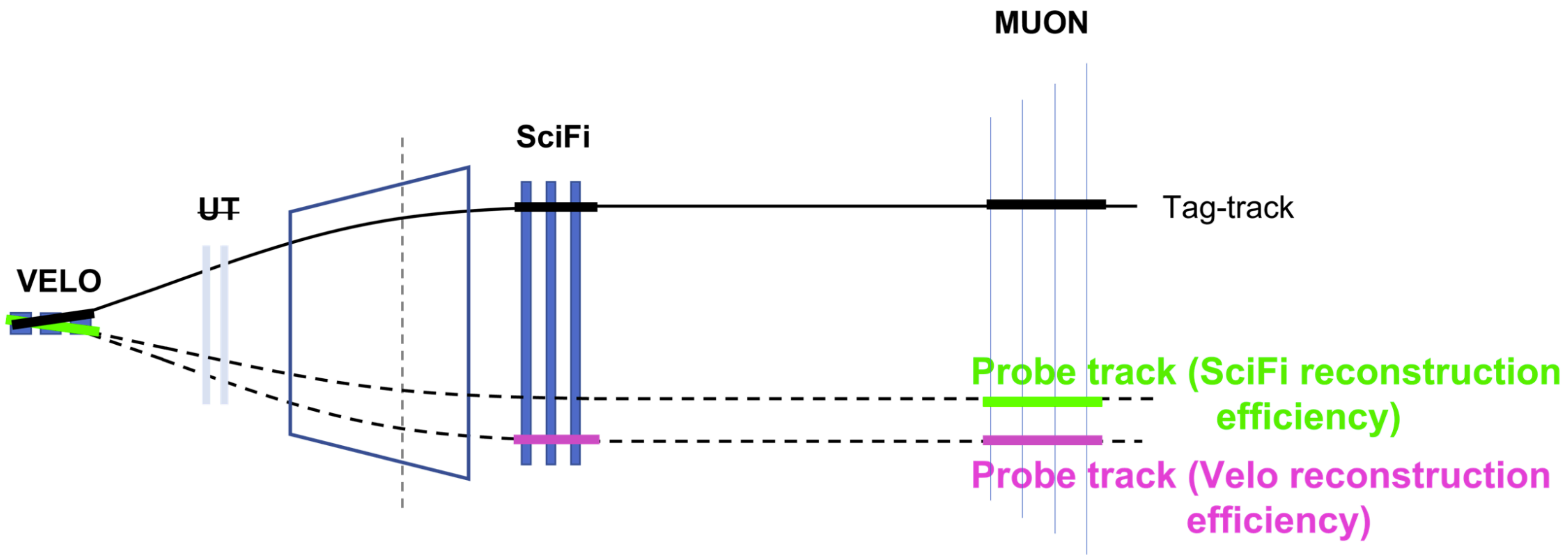




- High Efficiency in whole p_T region
- Low Ghost/fake rate
- Excellent momentum resolution ~0.5%

Tracking efficiency calibration

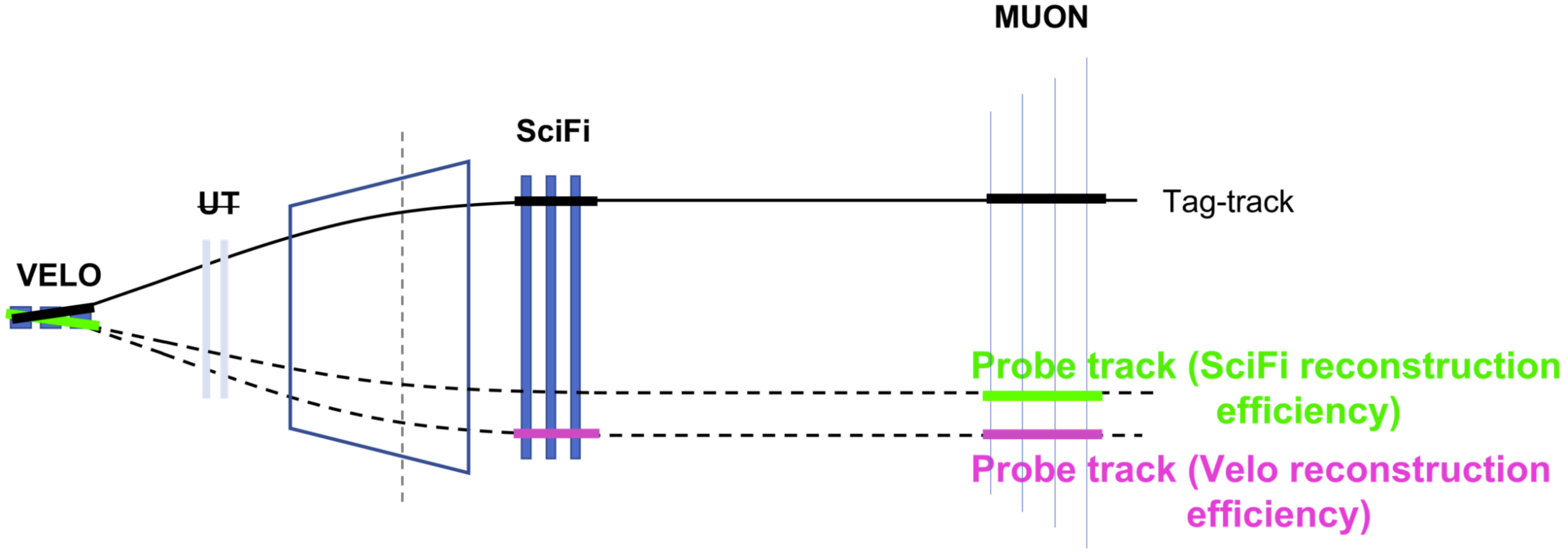
- Data-driven method to determine the tracking efficiency with $J/\psi \to \mu^+\mu^-$ candidates
  - Tag & probe method with fully reconstructed **tag muon** + partially reconstructed **probe muon**
  - Determine **track reconstruction efficiency** by trying to match the probe track to fully reconstructed long track → **matched or failed**



Tracking efficiency calibration

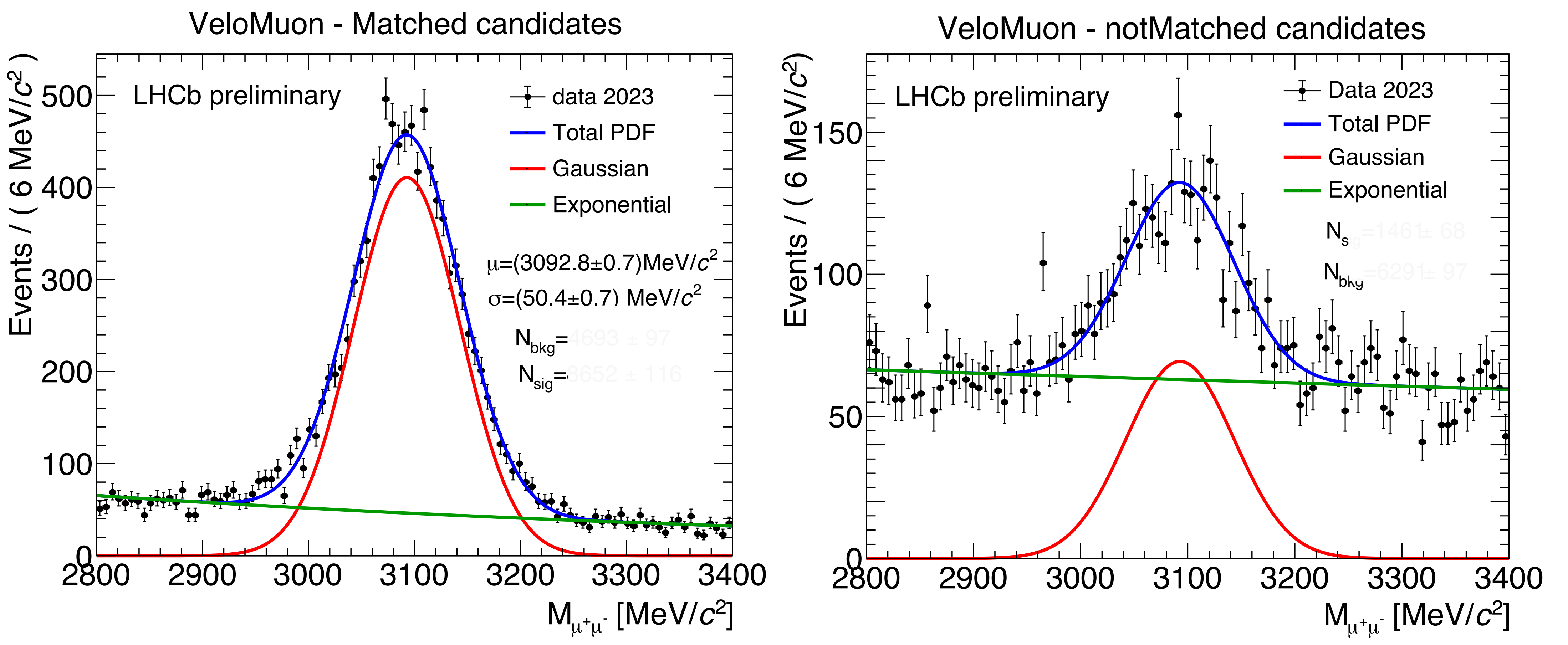
- Data-driven method to determine the tracking efficiency with $J/\psi \to \mu^+\mu^-$ candidates
  - Tag & probe method with fully reconstructed **tag muon** + partially reconstructed **probe muon**
  - Determine **track reconstruction efficiency** by trying to match the probe track to fully reconstructed long track → **matched or failed**

$$\epsilon_{track\ reconstruction\ efficiency} = \frac{N_{sig,matched}}{N_{sig,matched} + N_{sig,failed}}$$



Simultaneous fit

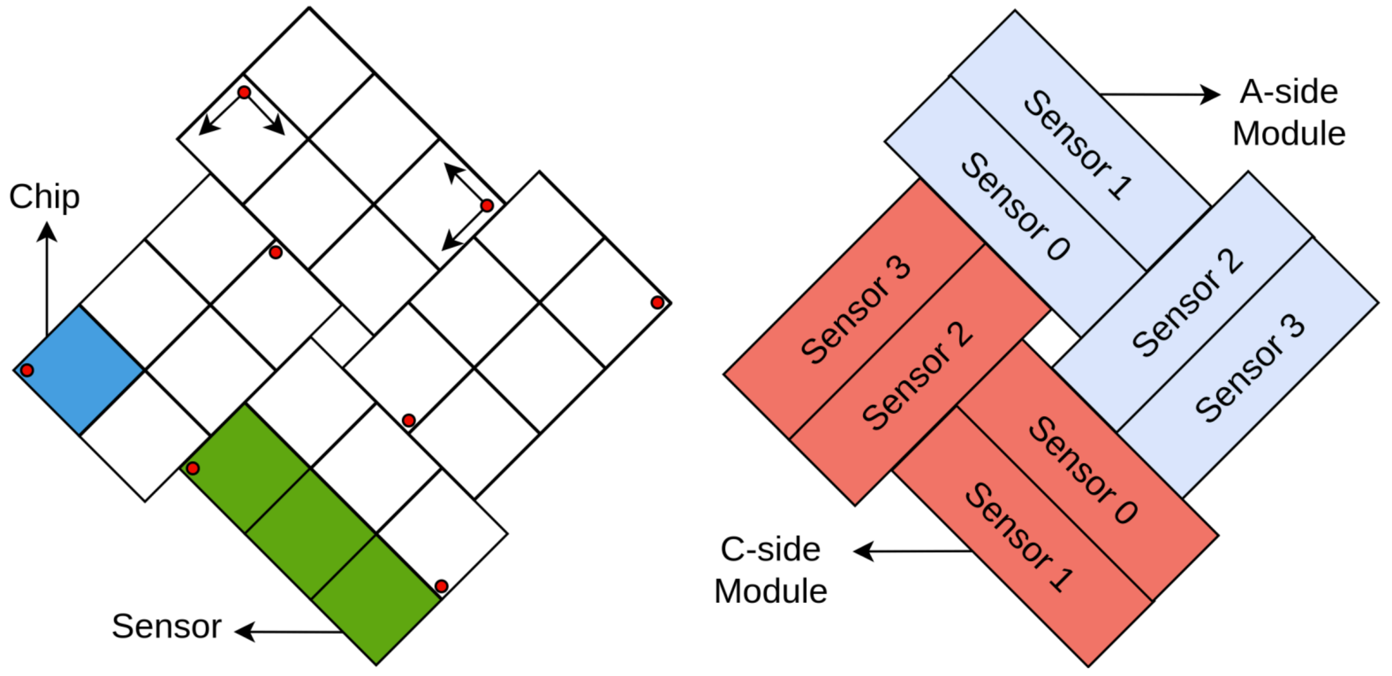
Unofficial, illustration only!

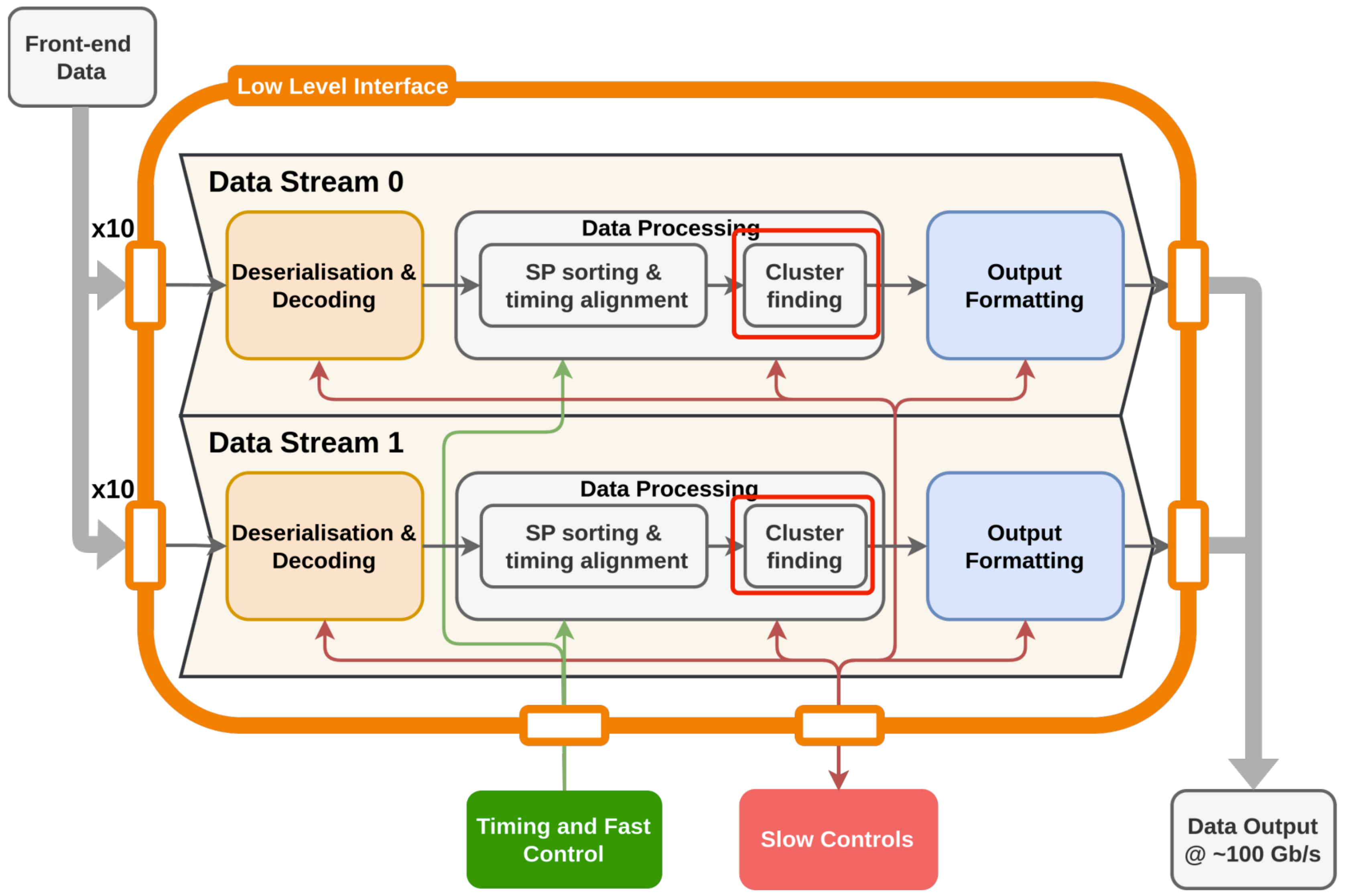


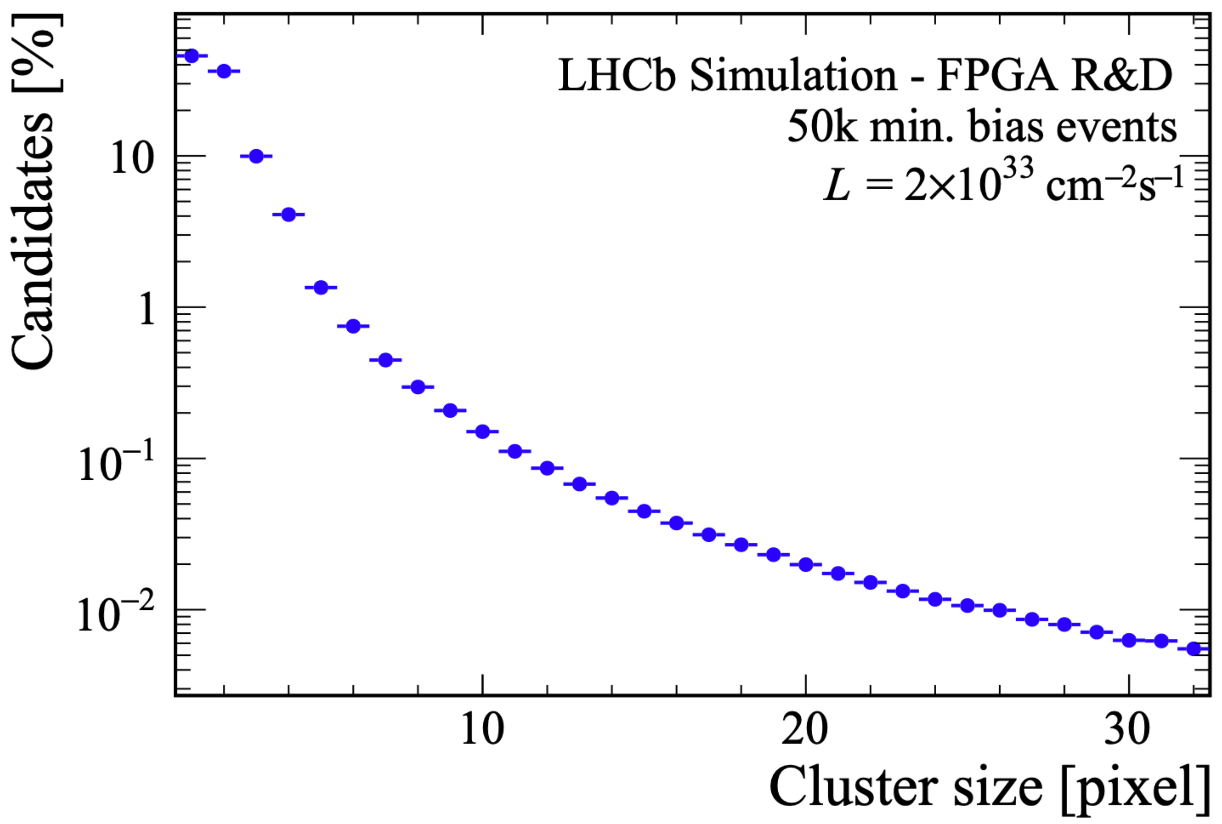
Clustering and tracking with FPGA

arXiv:2302.03972

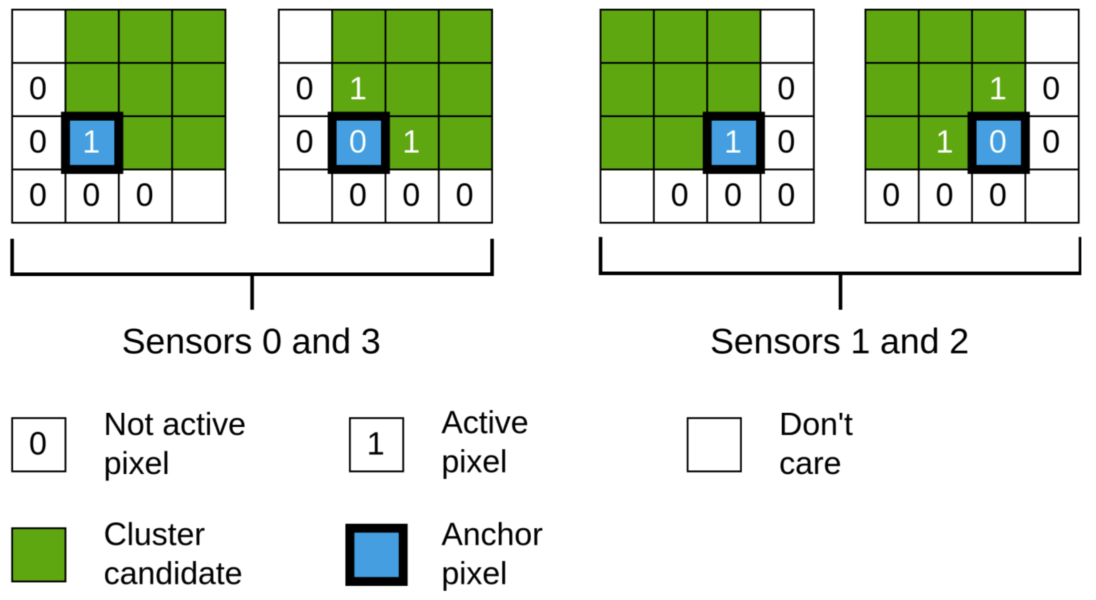
- Clustering with FPGA (Retina cluster) is applied in LHCb Run 3 data taking successfully
  - Cluster efficiency and tracking efficiency comparable with reconstruction in CPU

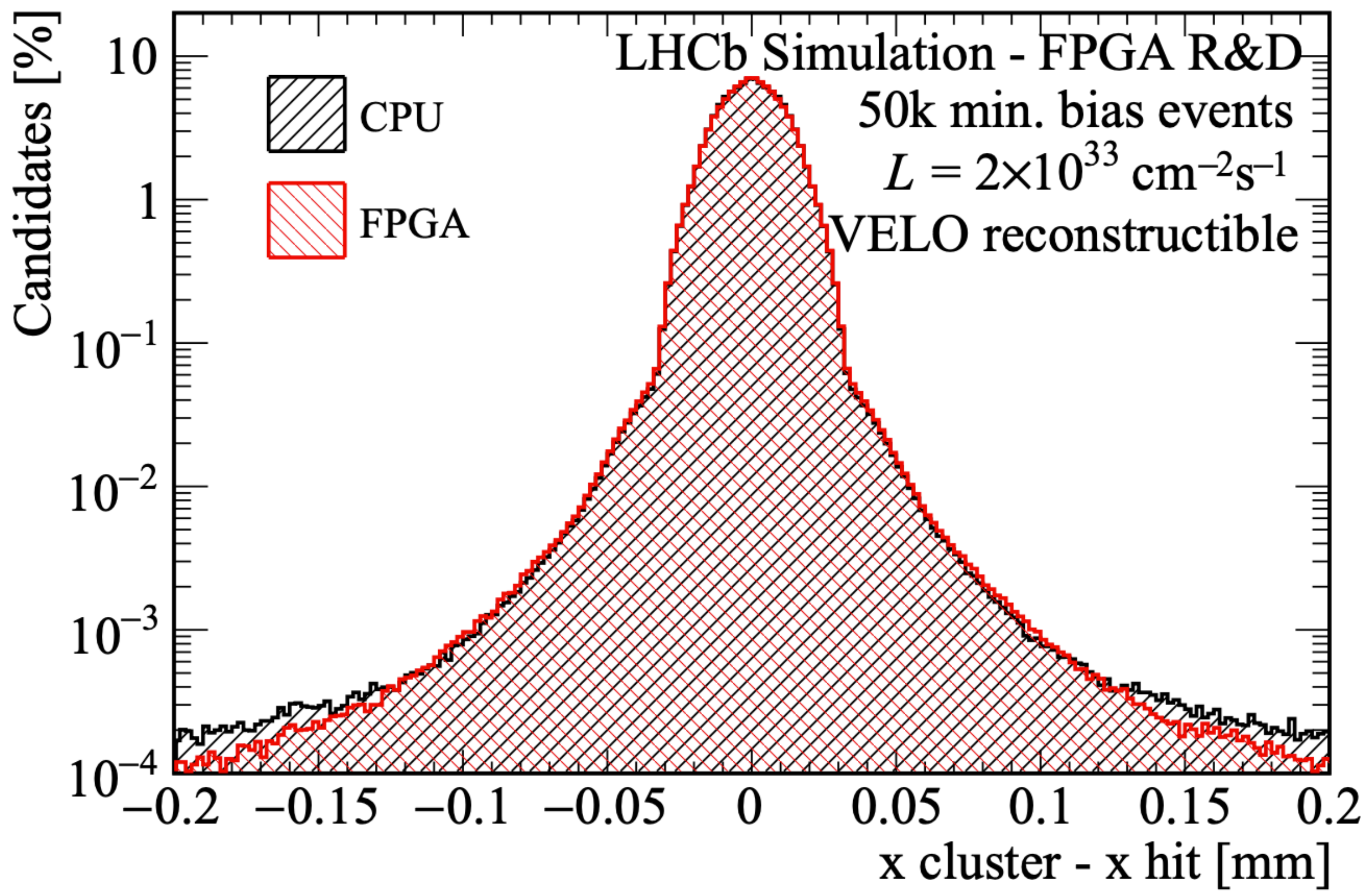


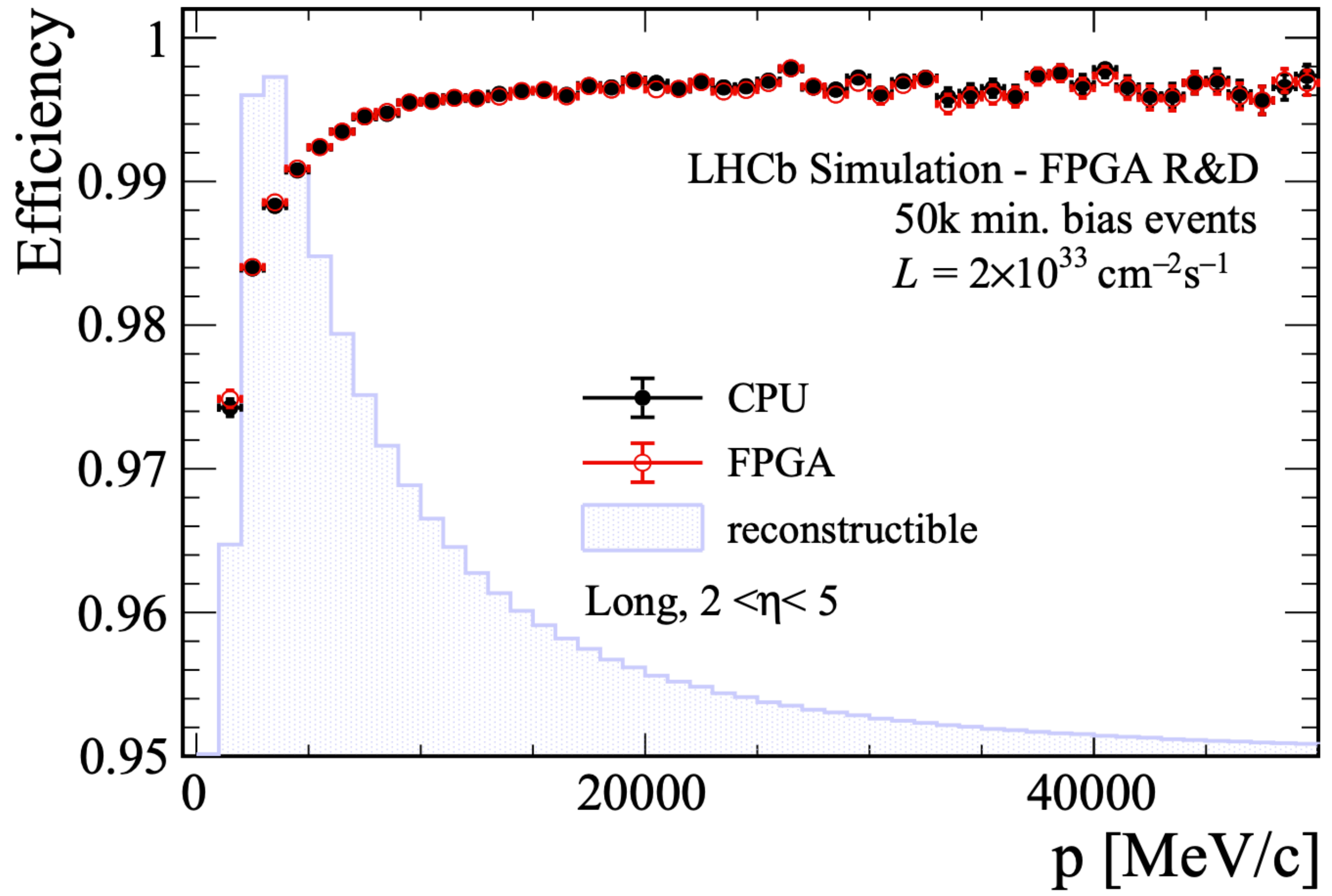




- 53% Lookup table for isolated clusters
- non-isolated clusters





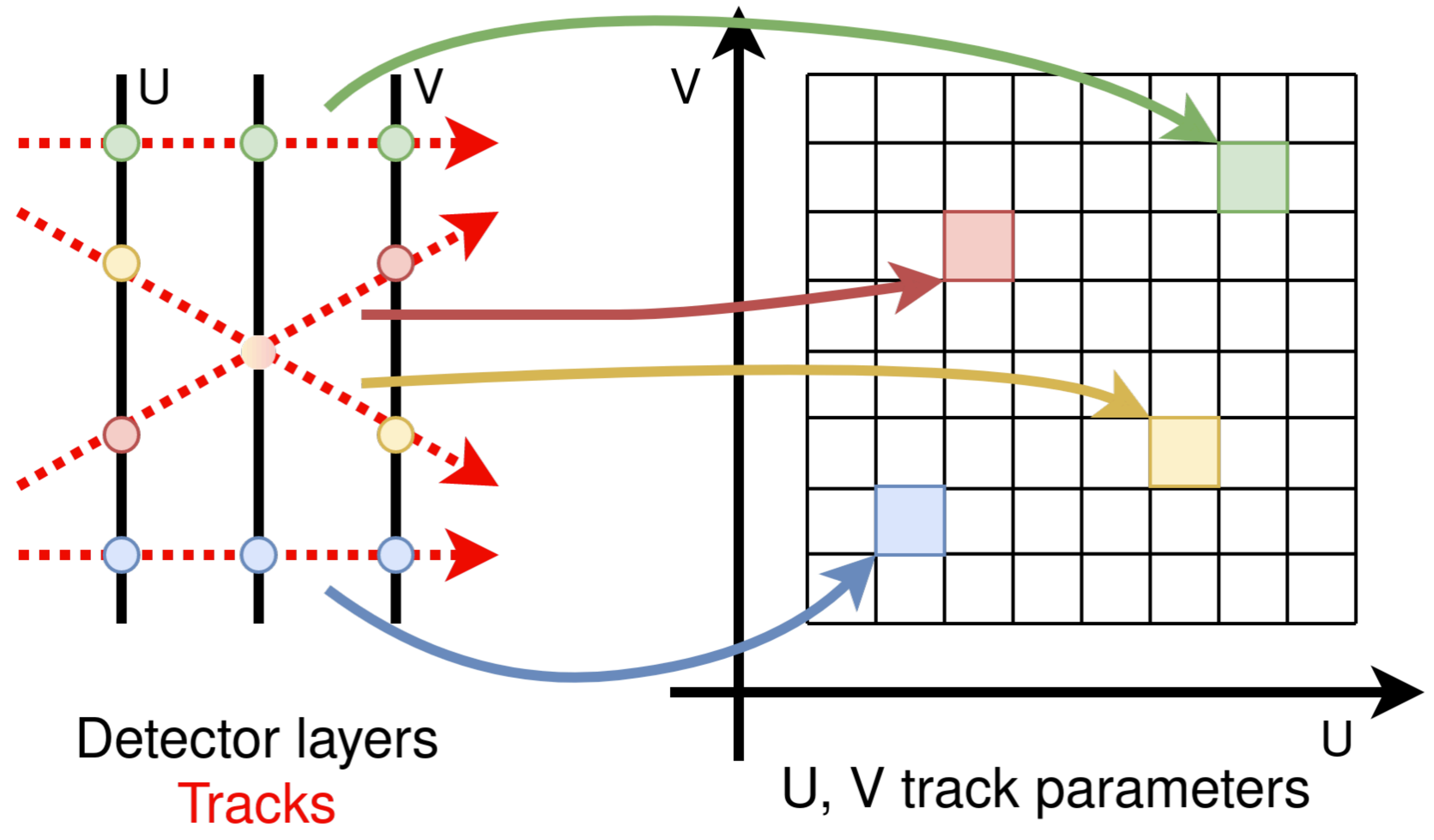


Track reconstruction with FPGA

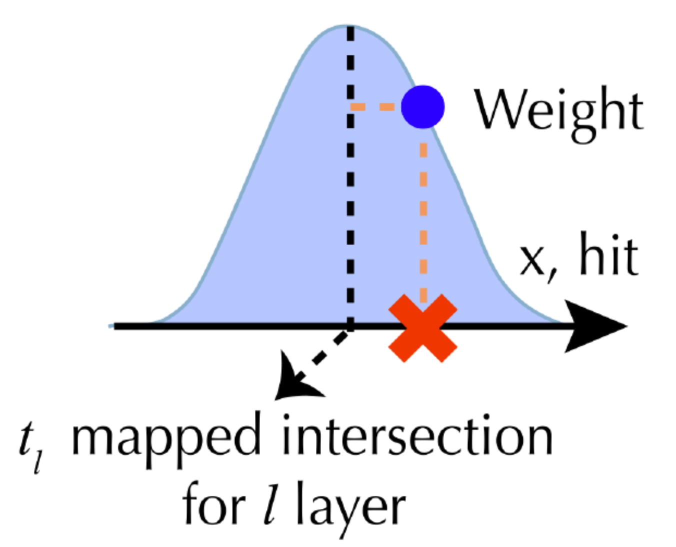
- Segment both the distribution of network and the cell matrix into smaller blocks to FPGAs
- Find the T-tracks primitives with FPGA, encoding the recalculated primitives to Retina Raw banks
- Replace the T tracks pattern recognition in HLT1 GPU with the decoding of the Retina Raw bank, expected to largely speed up the tracking in HLT1

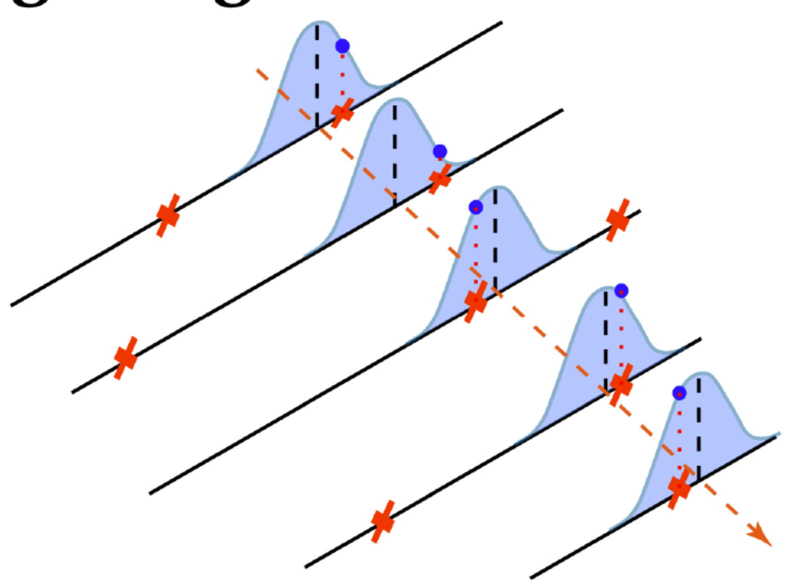
LHCb-Pub-2024-001

Step1: Parametrisation of reference tracks



Step 2: Accumulating weights (each cell)

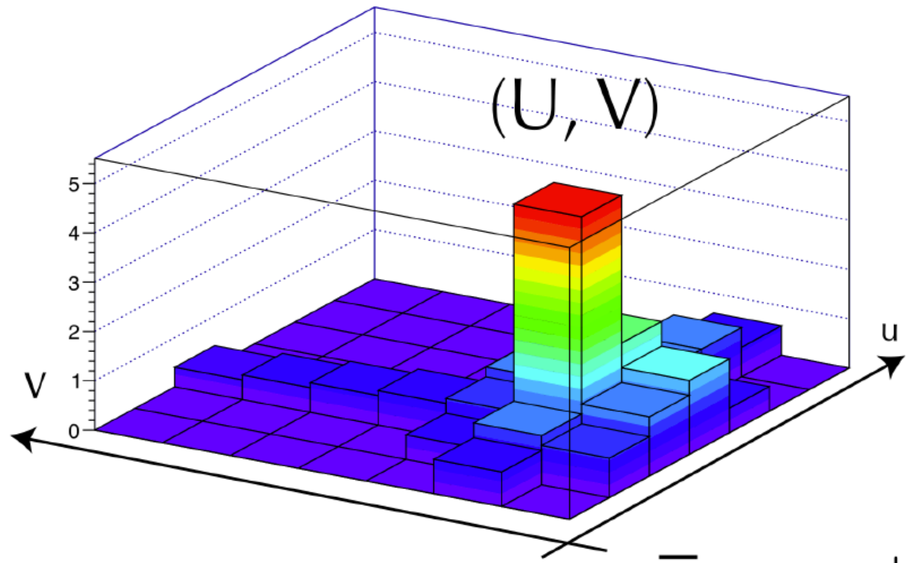


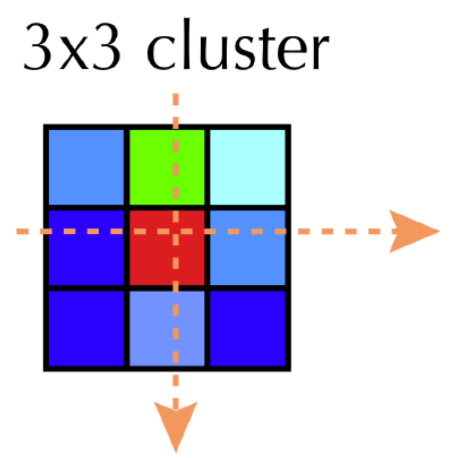


$$R = \sum^{\text{hits}} e^{-\frac{(x_l - t_l)^2}{2\sigma}}$$

R is close to N (# of layers) only if we have a set of hits near the mapped track

Step 3: Find the local maxima and compute centroid





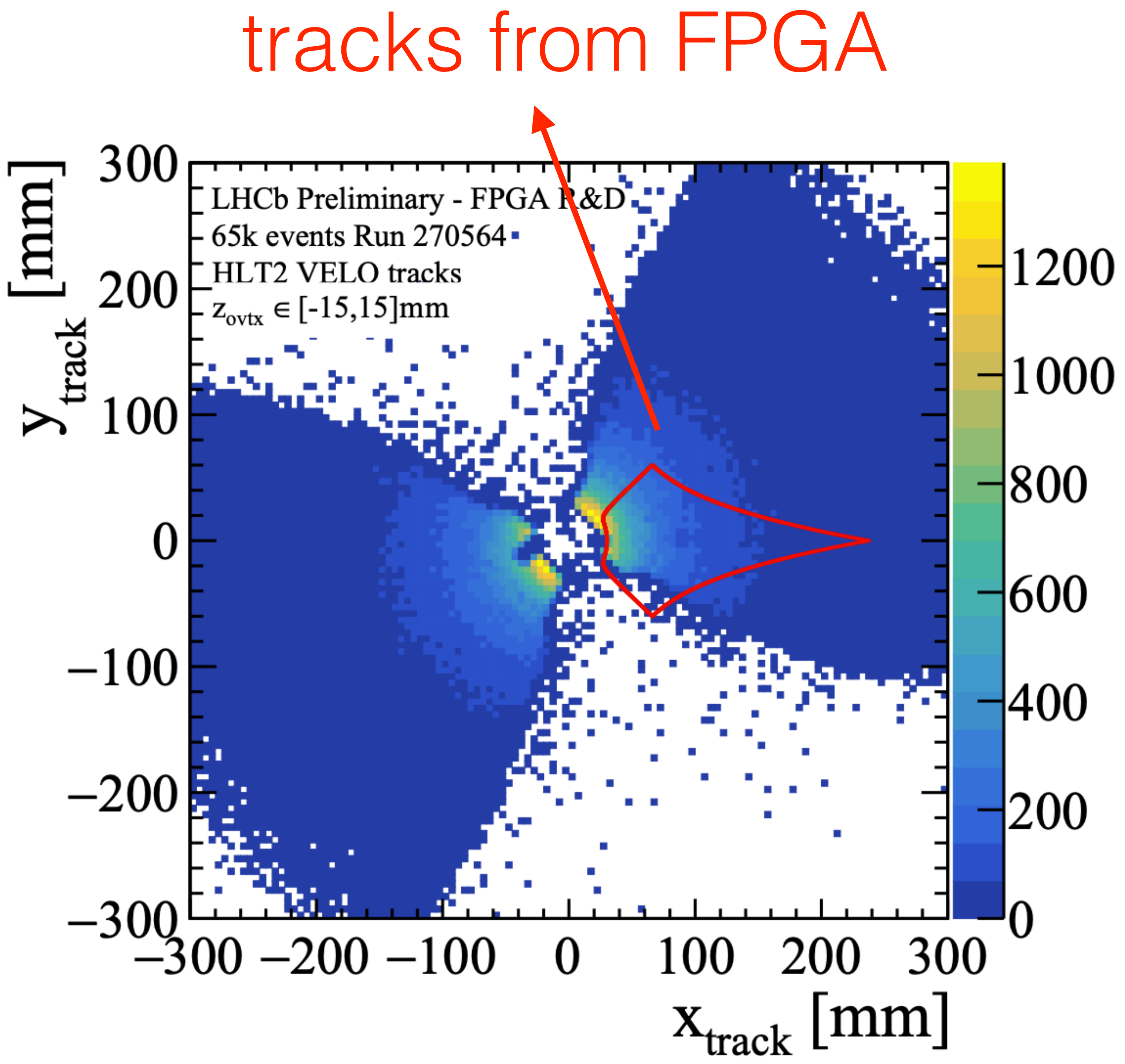
$$\overline{u} = u_0 + \frac{\sum_{ij} iR_{i,j}}{\sum_{ij} R_{i,j}} \qquad \overline{v} = v_0 + \frac{\sum_{ij} jR_{i,j}}{\sum_{ij} R_{i,j}}$$

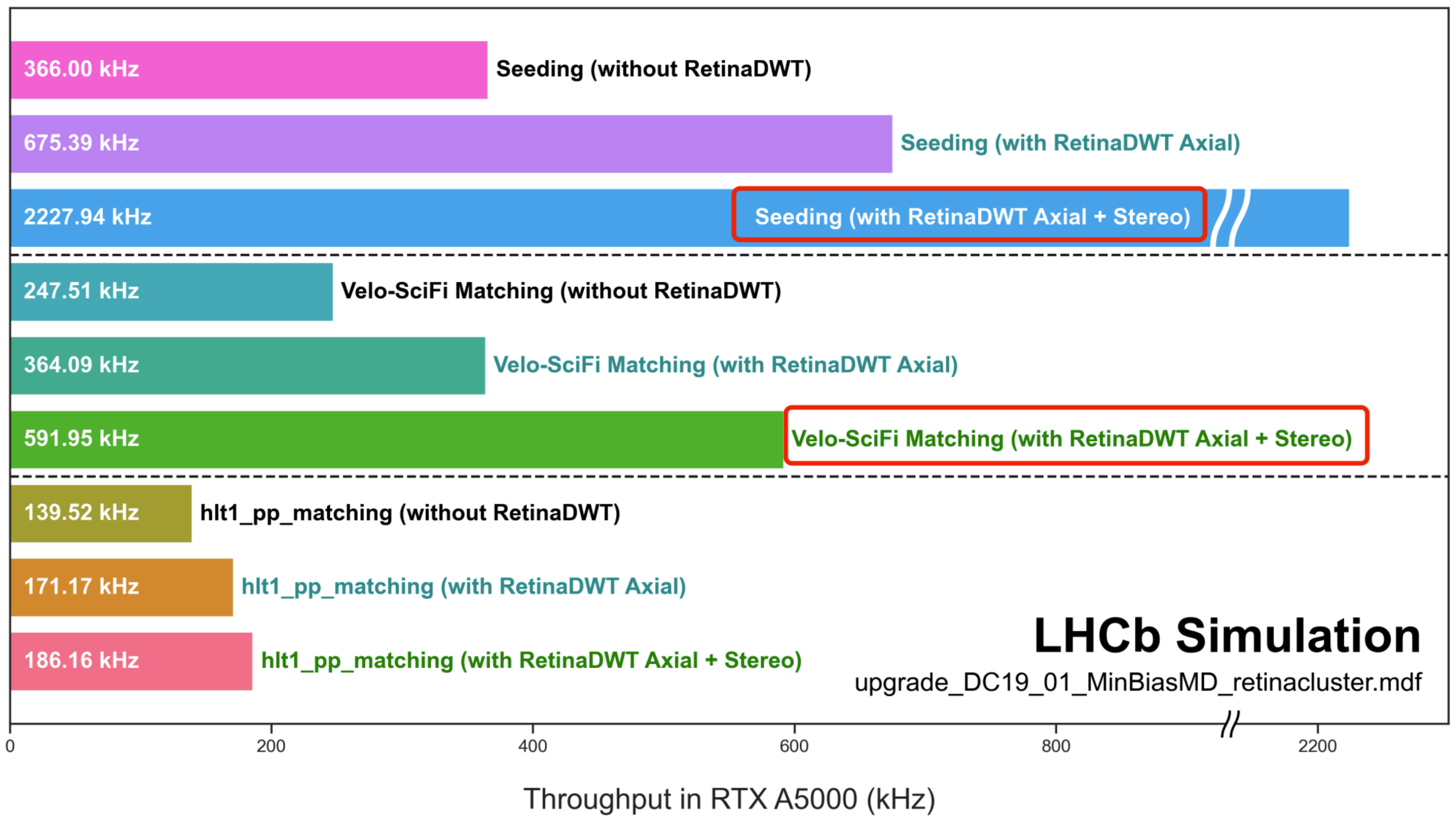
$i = U-1, U, U+1$
$j = V-1, V, V+1$

Track reconstruction with FPGA

- A demonstrator of VELO tracks successfully implemented
- Emulator of the GPU track reconstruction with Retina T-track primitives (RetinaDWT), significant speed up in the throughput test
- Proposal of Retina DWT approved by LHCb and in review with LHCC

LHCb-Pub-2024-001

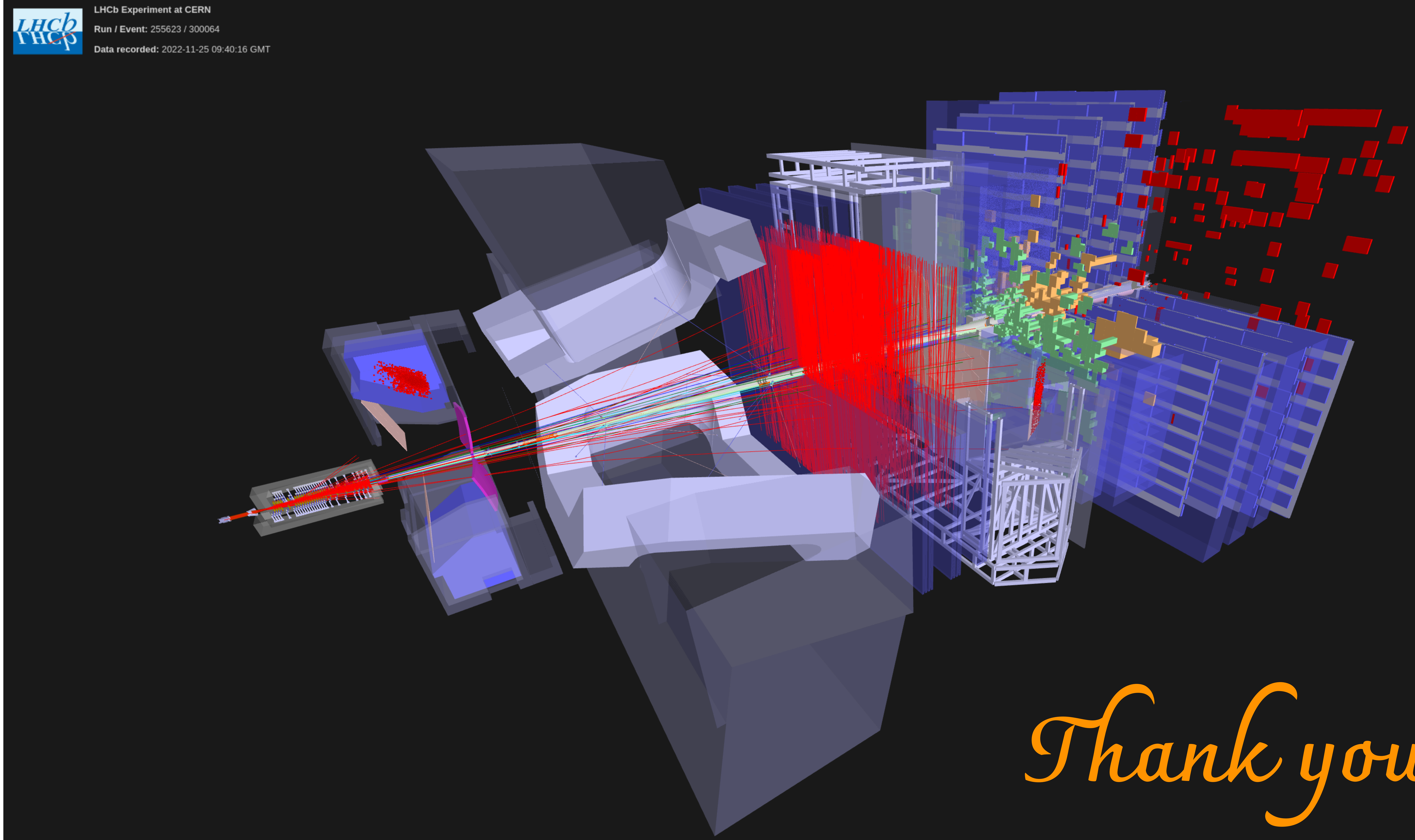




Summary

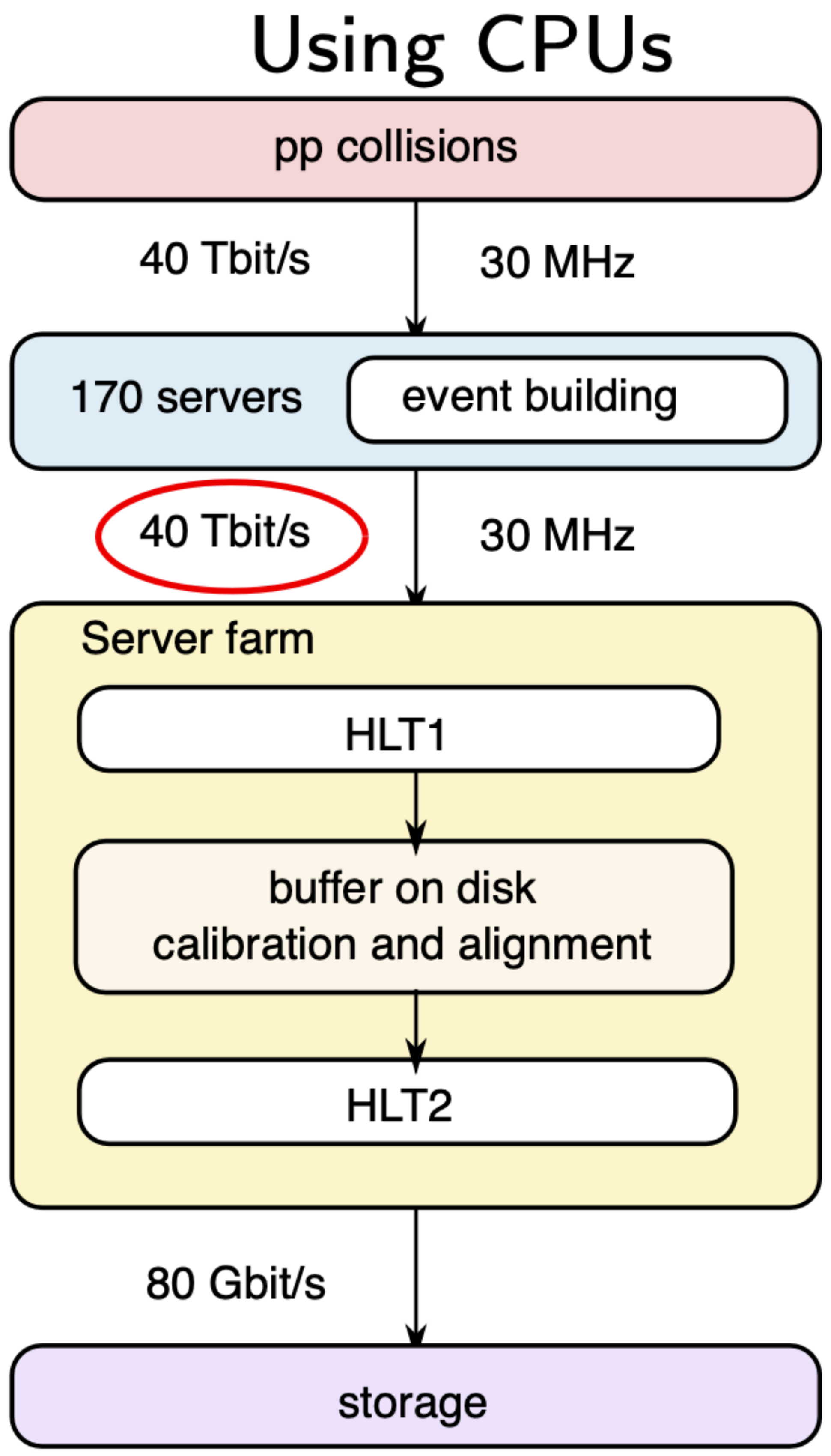
- LHCb Run 3 changes the trigger paradigm with software only, pioneering in the real time processing
- Partial tracking reconstruction at 30 MHz input rate using GPUs
- Full offline-quality reconstruction at 1 MHz input rate using CPUs
- FPGA clustering applied in VELO and downstream tracking in good progress
- GNN based tracking in VELO being studied

- Hybrid architecture (GPU+CPU) in Run 3 would prepare us better for future upgrade
- R&D studies on optimal use of hybrid architectures (GPU/CPU/FPGA), remain flexible

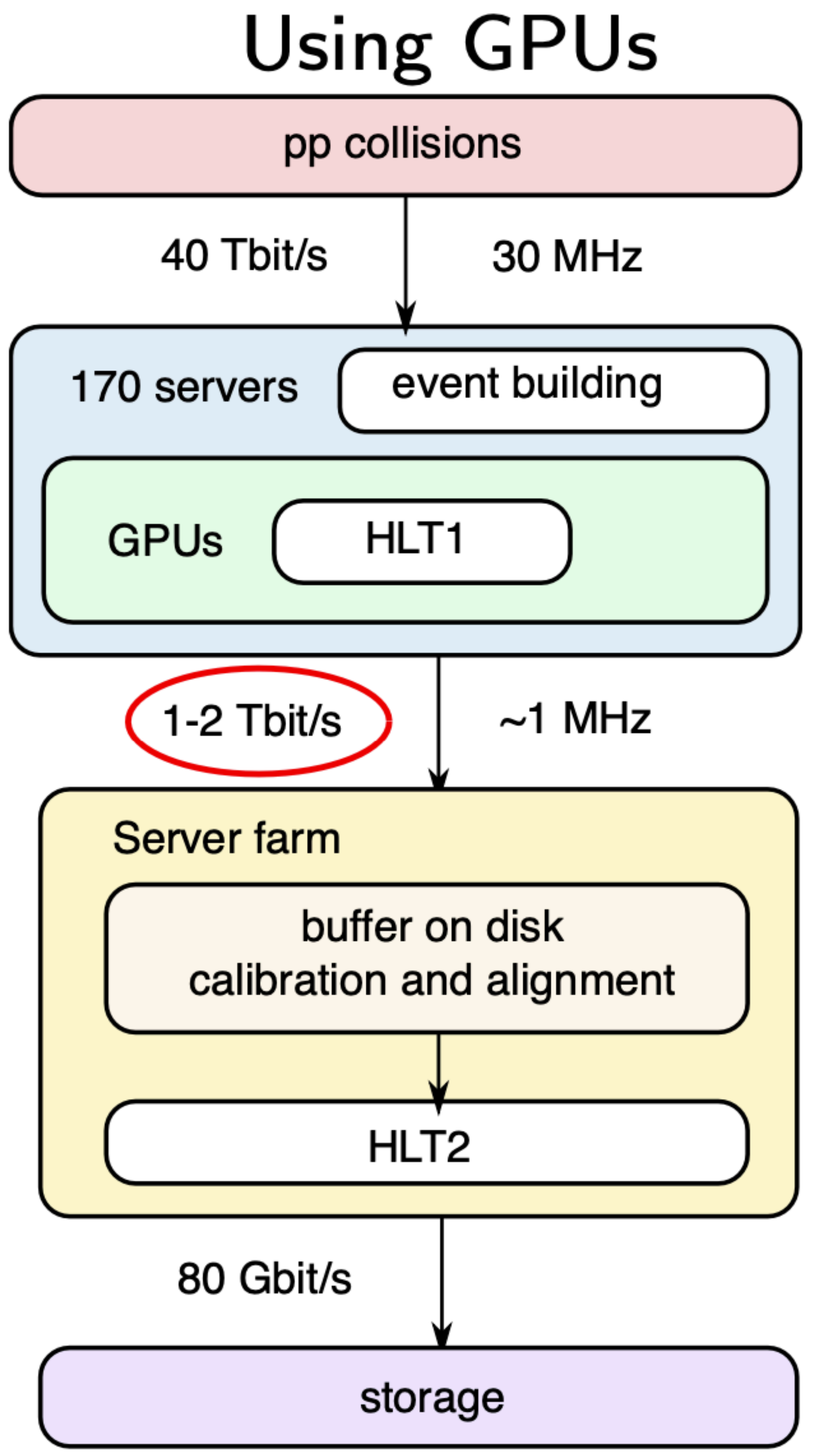

LHCb Experiment at CERN
Run / Event: 255623 / 300064
Data recorded: 2022-11-25 09:40:16 GMT

Thank you!

Hardware



Trigger TDR (2014)

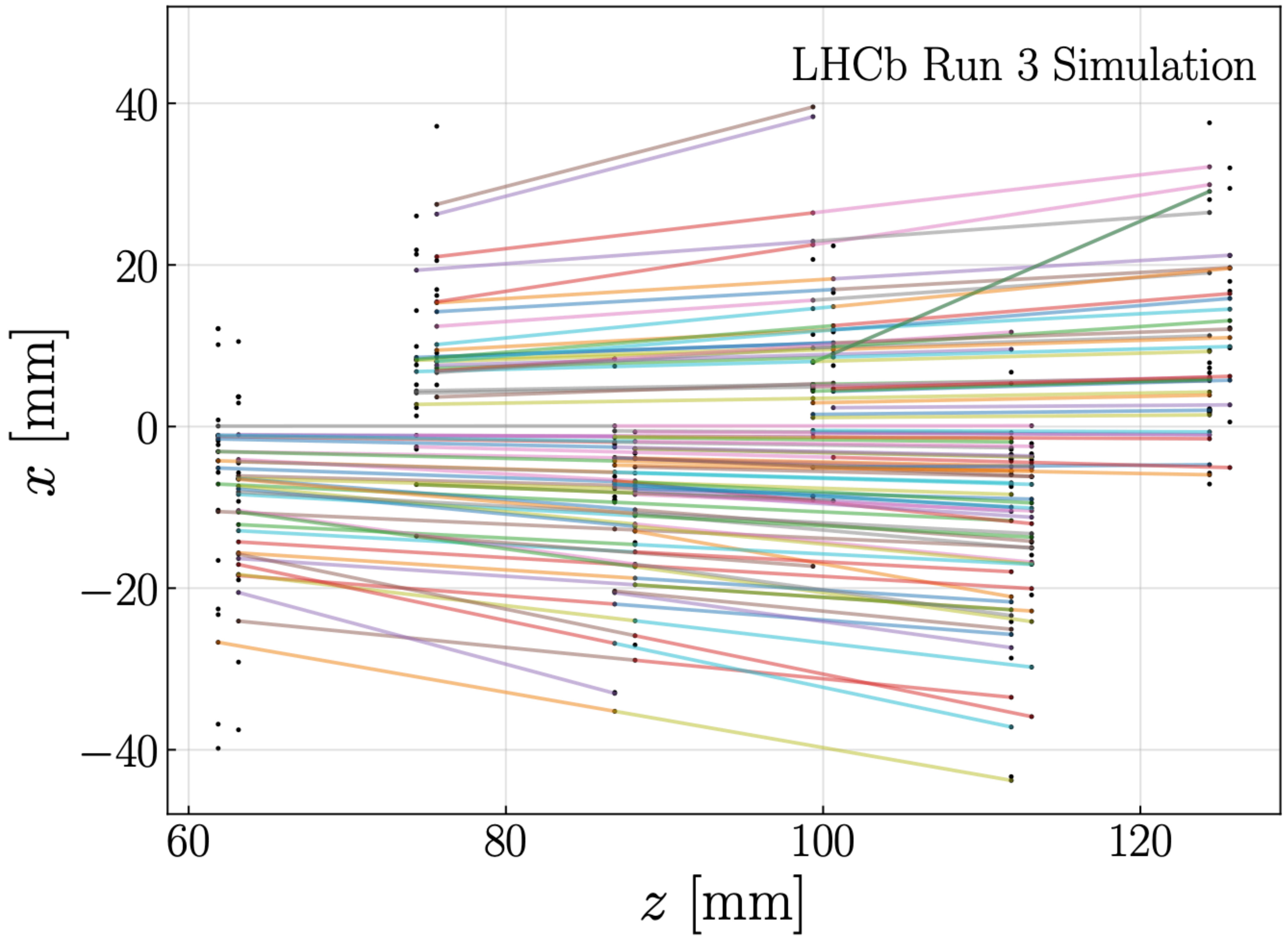


GPU HLT TDR (2020)
Allen project

LHCB-TDR-016.pdf

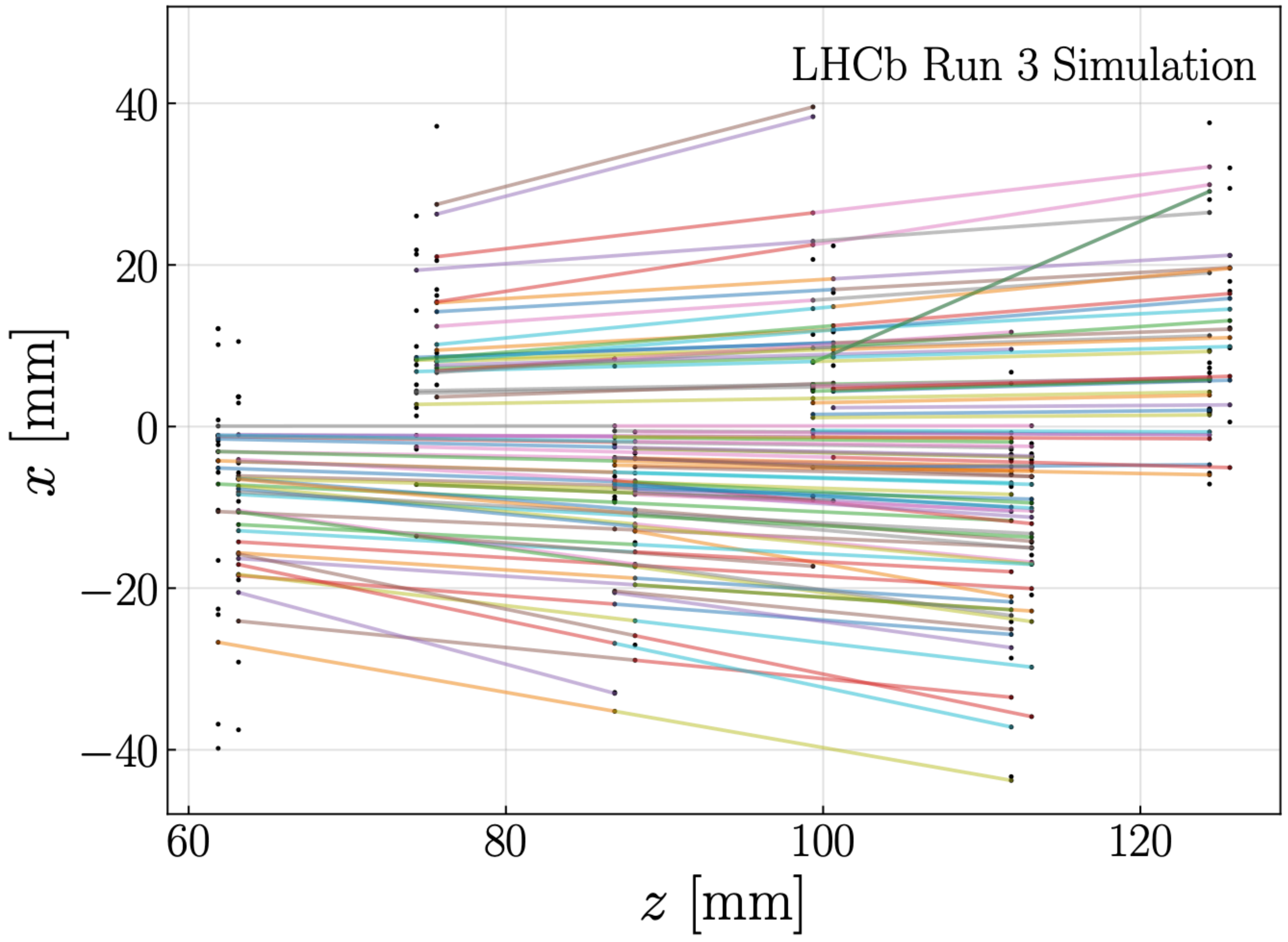
LHCB-TDR-021.pdf

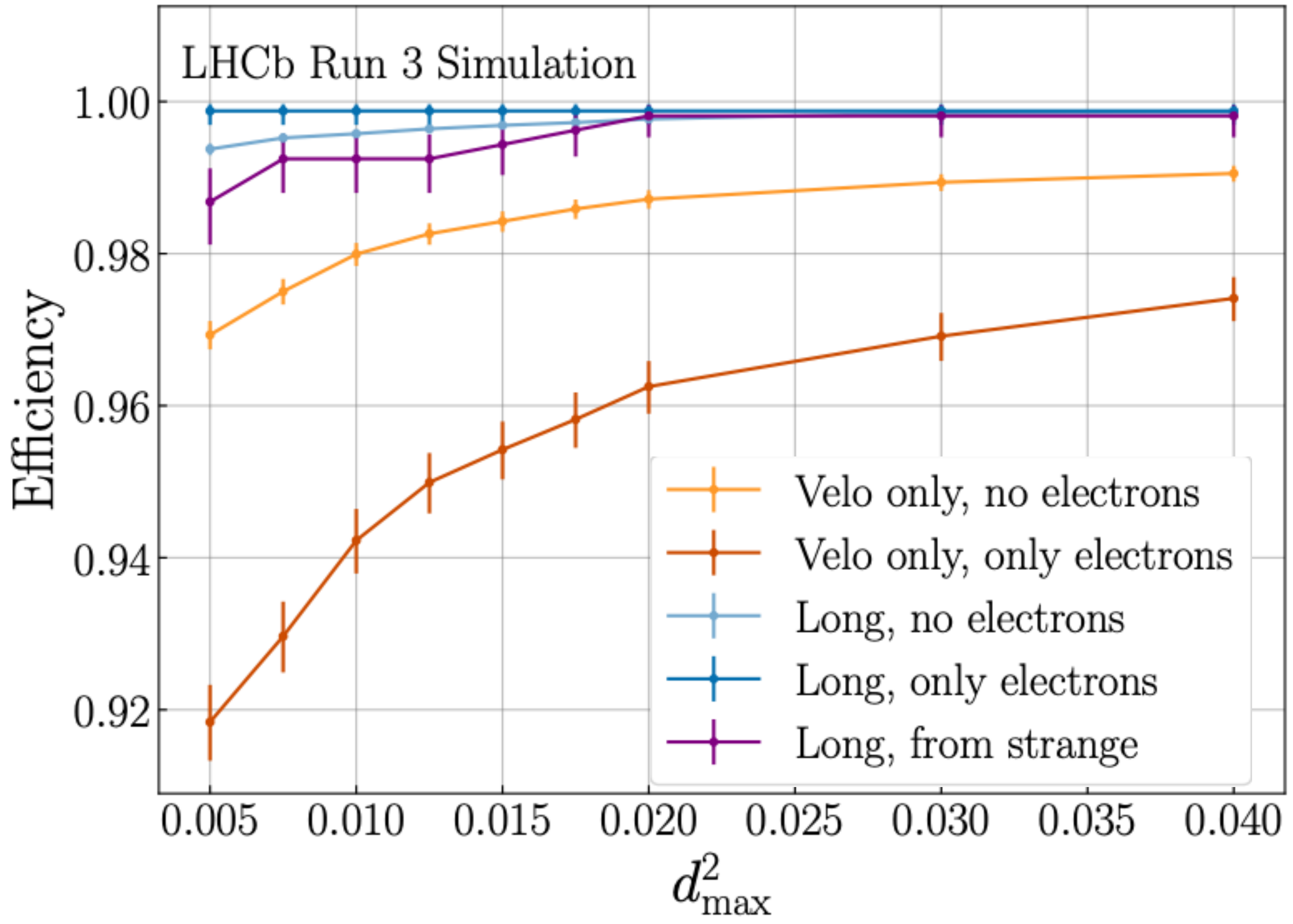
GNN based track finding in VELO



GNN based track finding in VELO

LHCb-FIGURE-2023-024

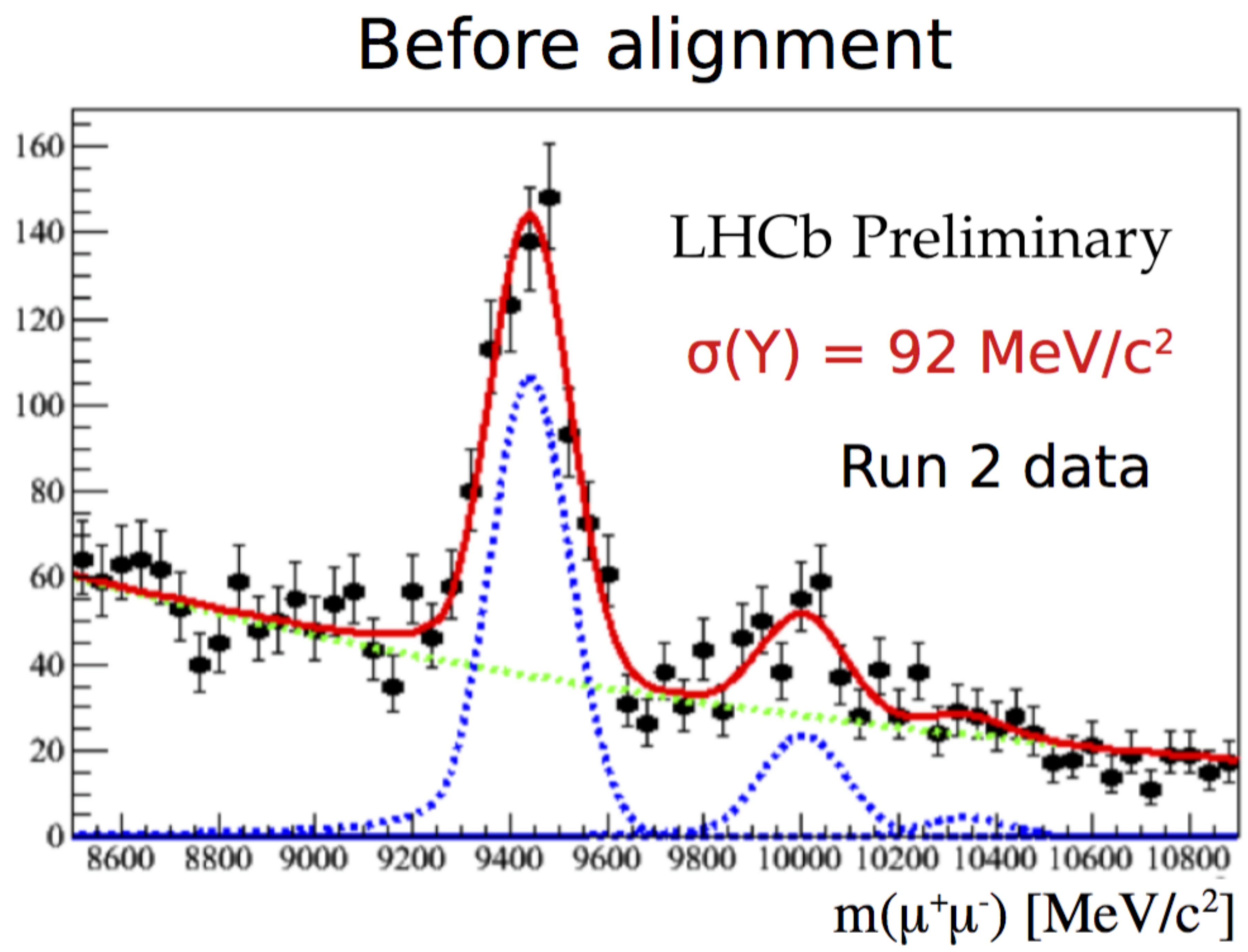


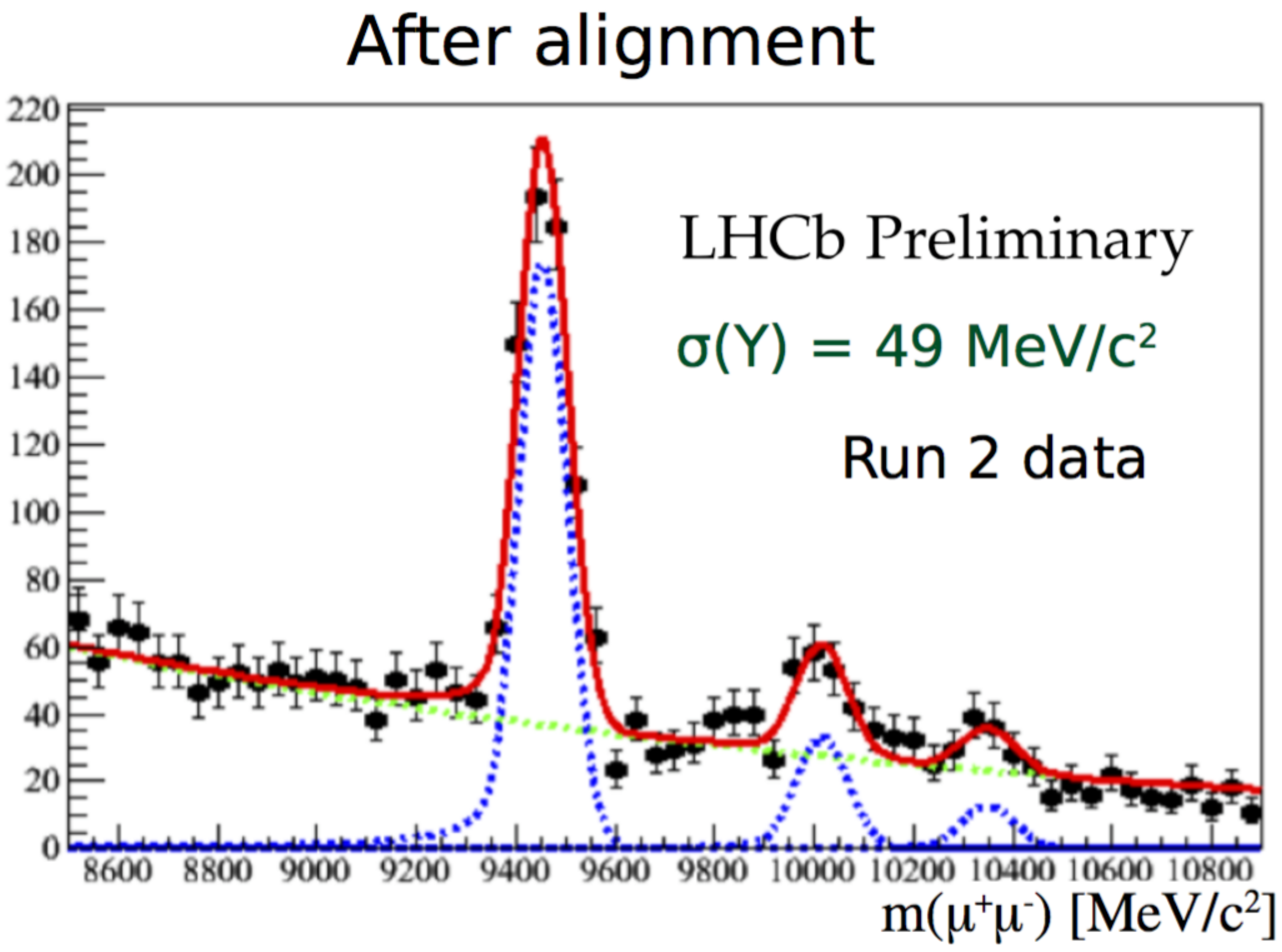


Alignment & Calibration

- Crucial for efficient and pure selections require offline-quality reconstruction at the HLT2 level
- Use output bandwidth more efficiently
  - Better mass resolution
  - Better particle identification
  - Less background

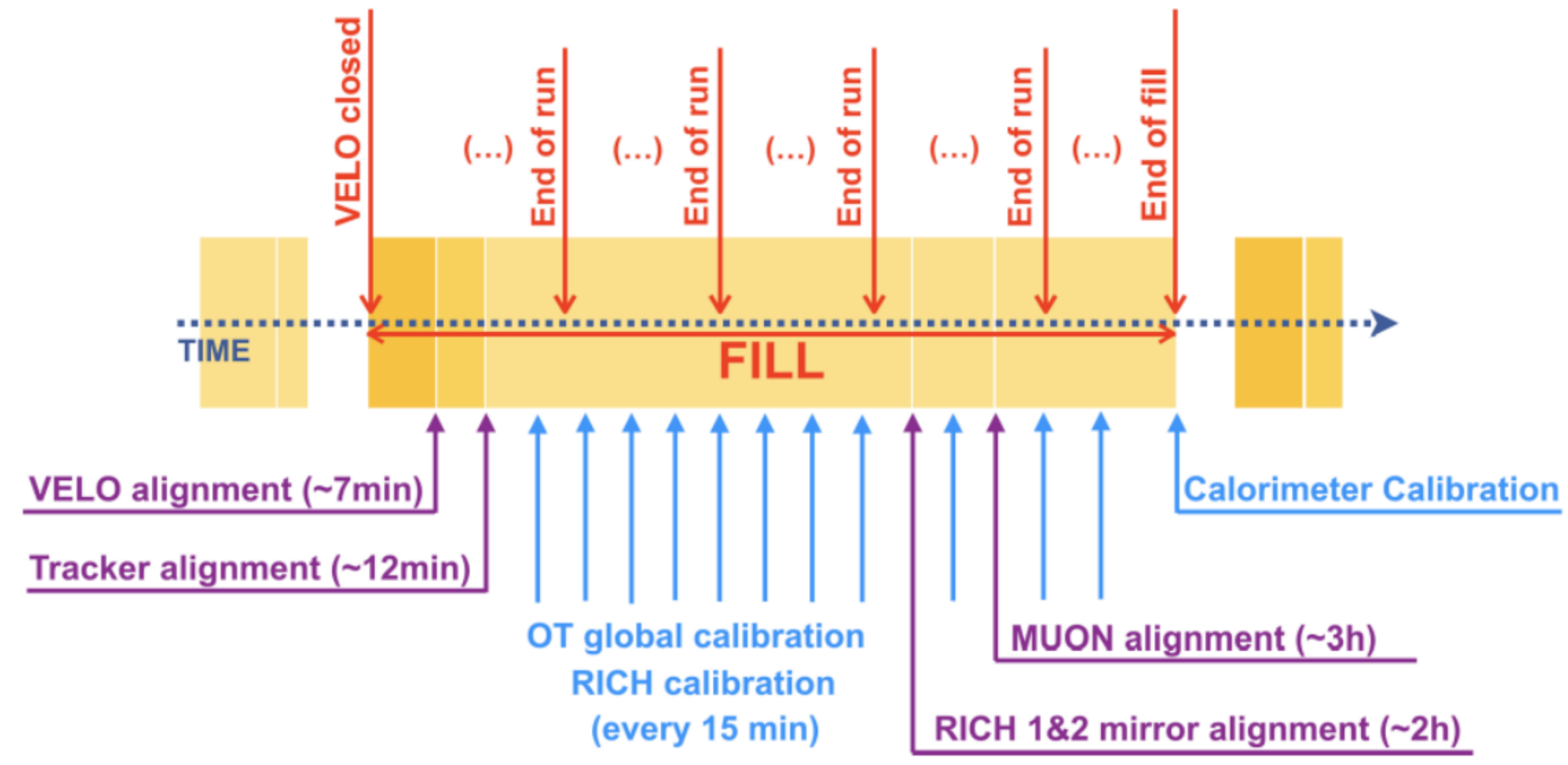
Journal of Physics: Conference Series, 664 (2015)



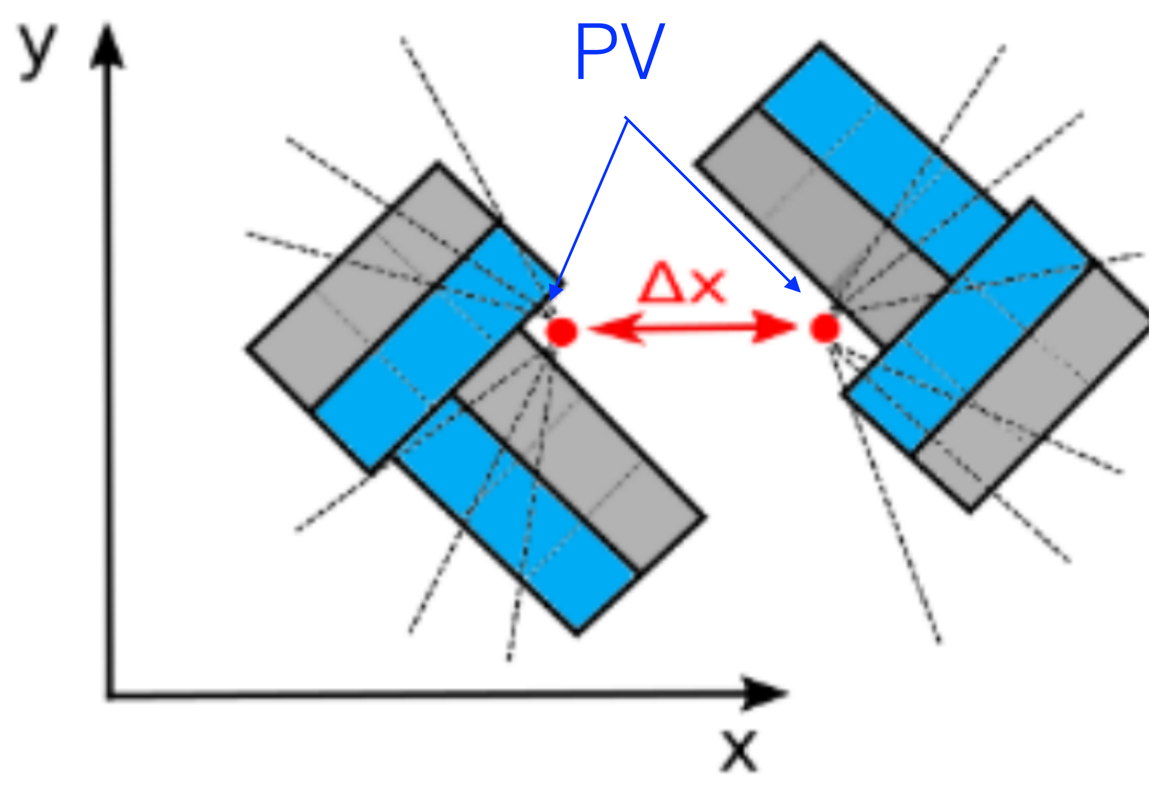


Alignment & Calibration

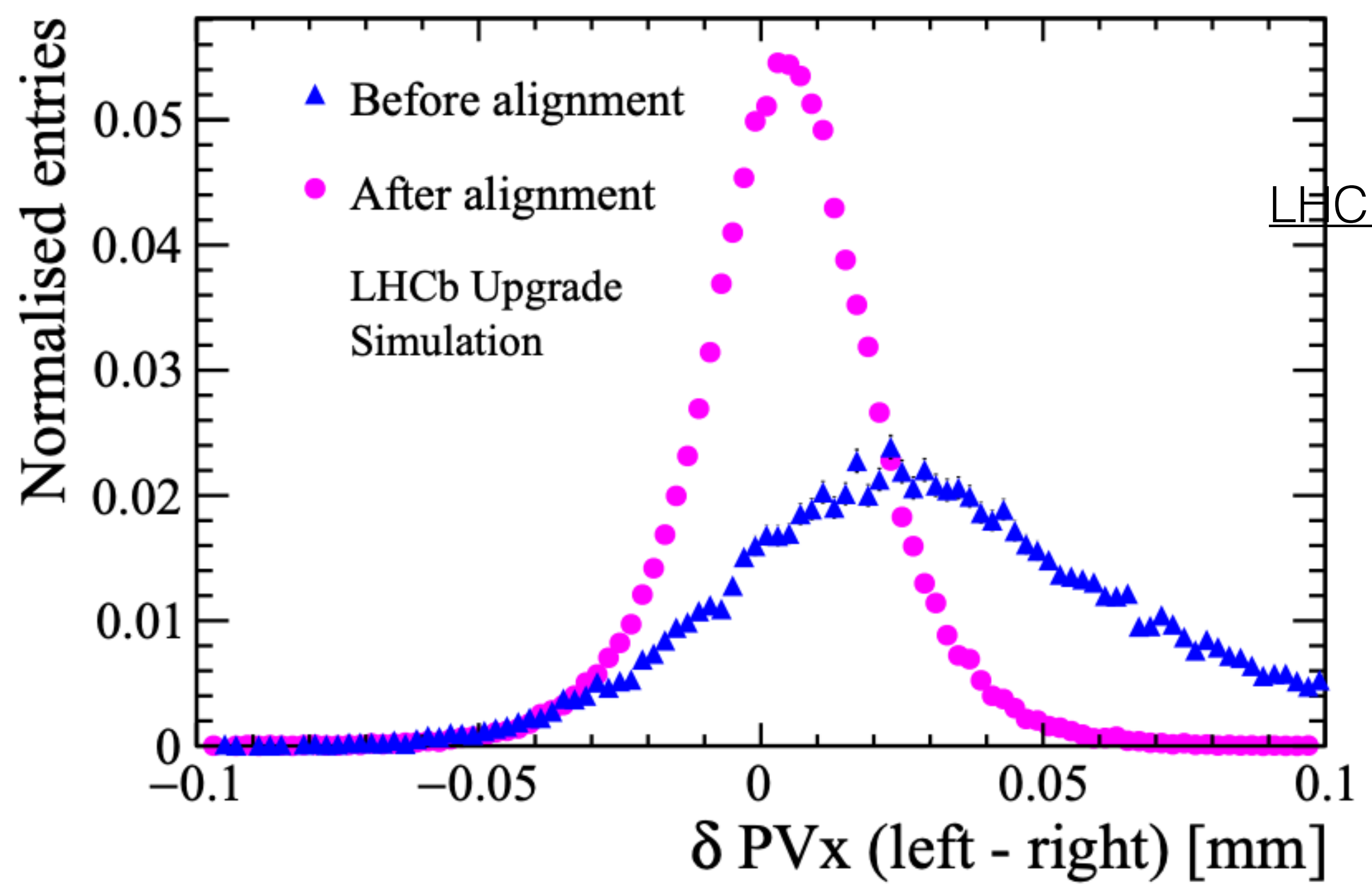
- Same disk buffer as Run 2 but 10x more data (Should be very fast!)
  Several minutes in Trackers & several hours for RICH & MUON



((~7min),(~12min),(~3h),(~2h)) - time needed for both data accumulation and running the task



- Test the alignment algorithm for each module with random misalignment in VELO



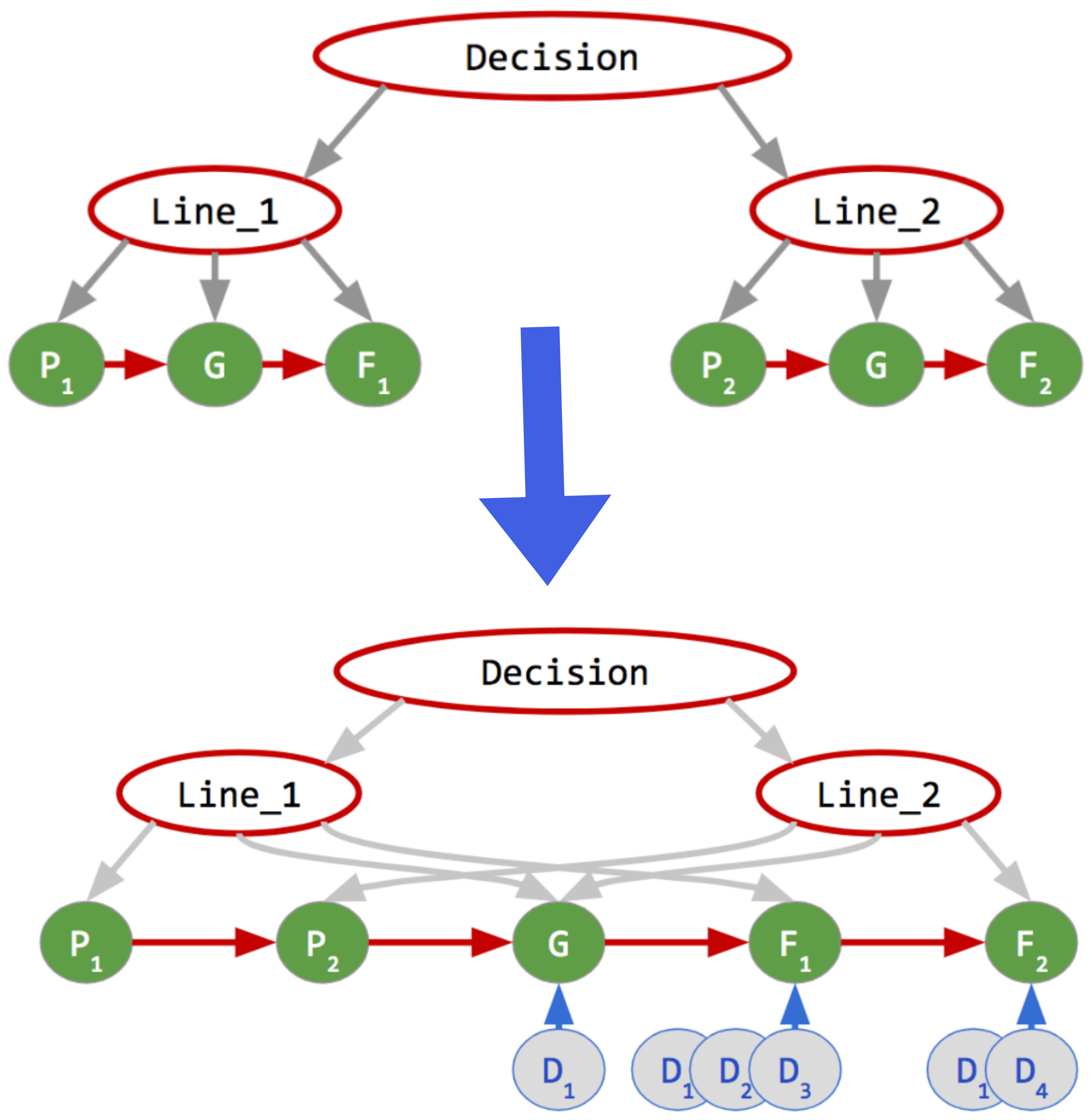
LHCb-FIGURE-2019-003

Managing O(1000) HLT2 Selection

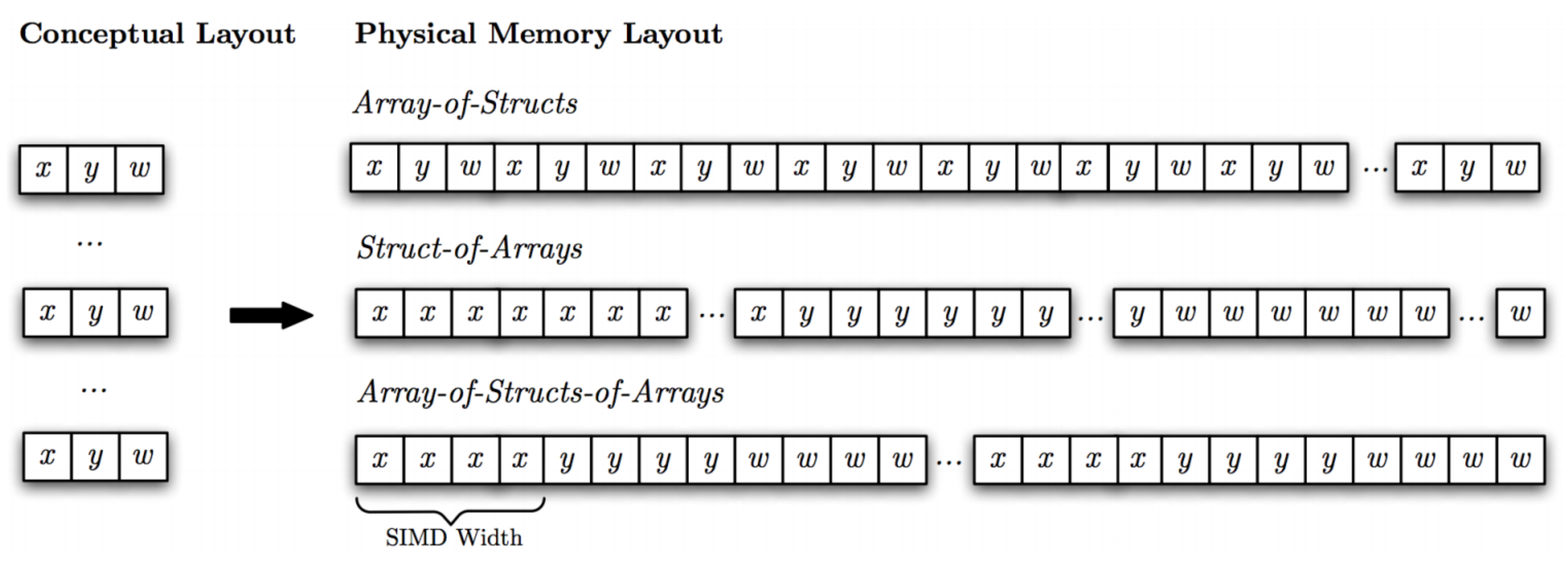
LHCb-Proc-2020-003

- Real-time selections with offline quality
- New algorithm scheduler with multithread

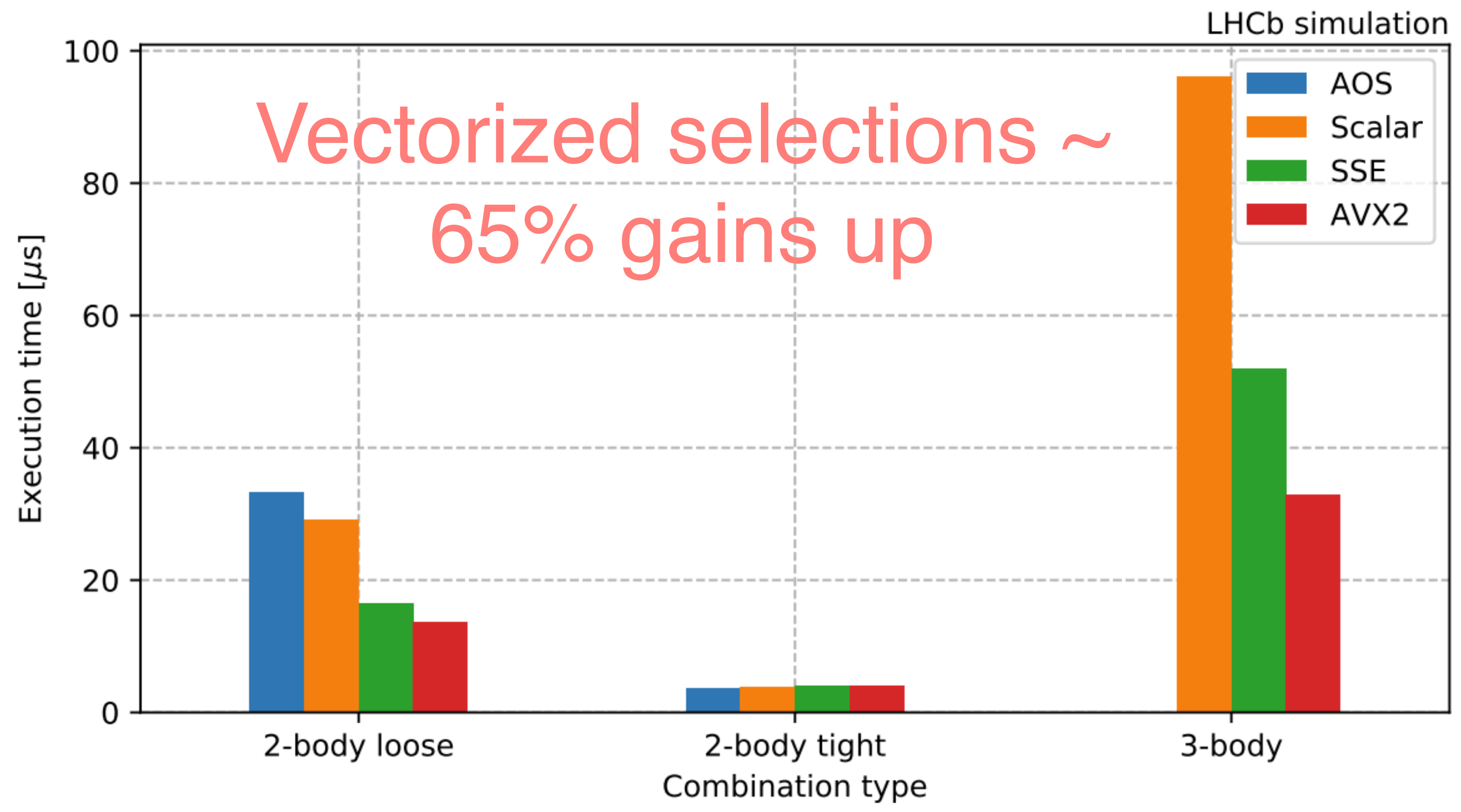
example dependency tree with two lines



static graph with ordered nodes



* SIMD-based algorithms allow for more efficient selections at a lower cost in CPU time



Vectorized selections