technische universität dortmund

FSP LHCb
Erforschung von Universum und Materie

Bundesministerium für Bildung und Forschung

Understanding the Alignment of LHCb's Scintillating Fibre Tracker

Nils Breer, Biljana Mitreska, Sophie Hollitt, Johannes Albrecht

07.09.2023

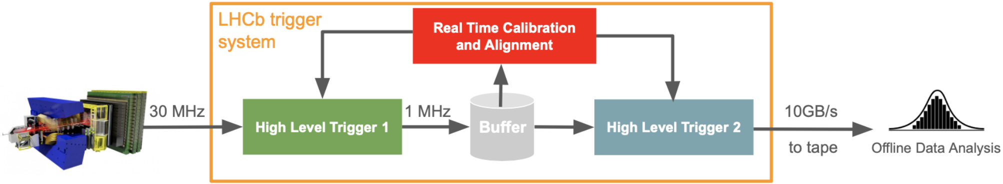
Maria Laach high energy physics school, Siegen

Overview

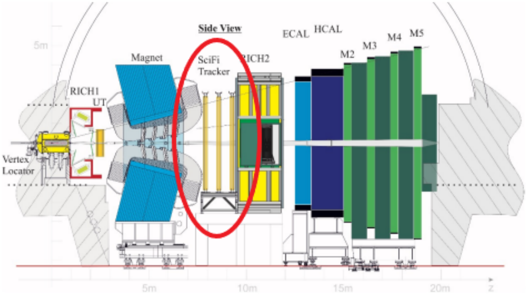
- The SciFi Detector Upgrade
- Importance of the SciFi and Alignment
- Understanding first alignments on 2022 data
- Stability measurements on 2022 data
- Joint constraints for SciFi modules

Importance of alignments

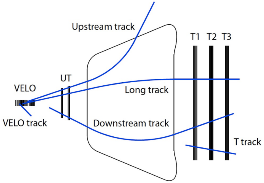
- Alignment is part of the LHCb trigger system
- Physics performance tied to alignment performance
- with optimal alignment:
    - → remove systematic biases for asymmetry measurements
    - best possible mass resolution



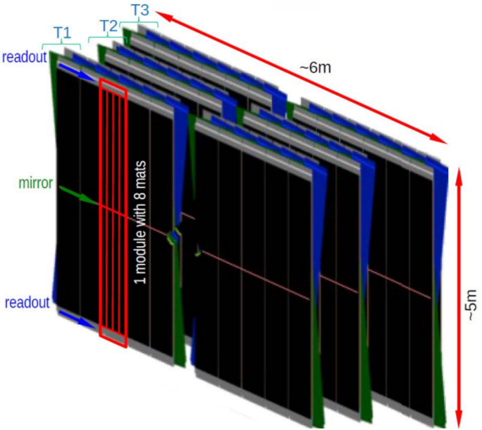
LHCb upgraded with the SciFi



- Consists of 3 stations: T1, T2, T3
- 4 layers per station: X1, U, V, X2
- replaces former IT and OT to cope with the increased instantaneous luminosity
- crucial part of tracking system



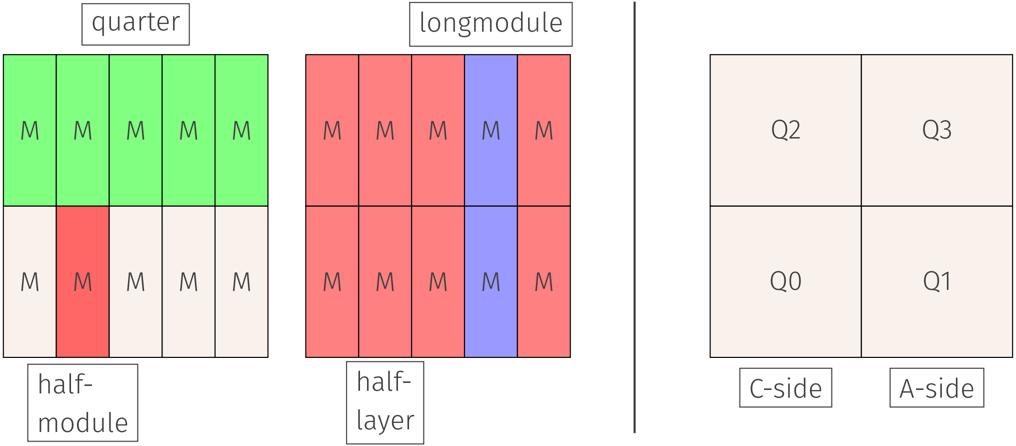
The Scintillating Fibre Tracker



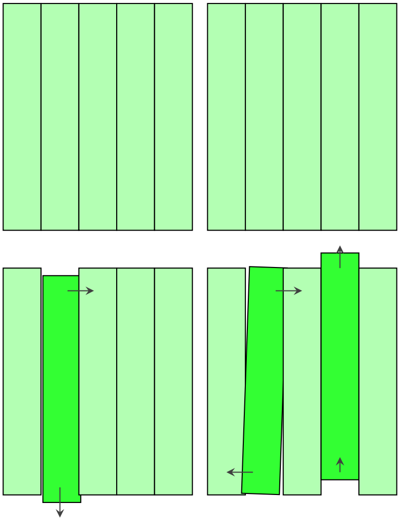
- Front two stations have 5 modules per side
- Back station has 6 modules on each side
- U, V layers have a **∓5 deg** stereo angle respectively
- → used for determining y-position of tracks by comparing hitposition at different angles

SciFi terminology

layers are divided into two halves commonly labeled as A-side and C-side



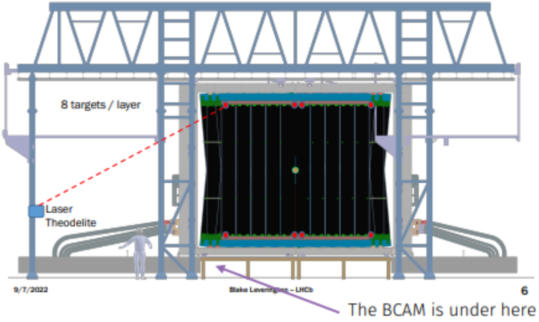
What is Alignment and why do we need it?



- top: ideal detector, bottom: physical detector
- Surveys are used to find the rotation and position of each detector component
- Are used as starting positions for software alignment
- Building tracks accurately requires positions in reconstruction to be as similar as possible to real positions

The survey: what is it and the different types

- measure distance of some points on the detector with a laser



- 2022: photogrammetry was recorded in assembly hall → not quite perfect
- 2023: photogrammetry will be recorded in cavern
- relative angles and positions between points are compared to simulation
- layer survey: performed in the cavern on the layer in the front in closed state (both halves together)
- module survey: performed inside assembly hall using reflective stickers keeping track of all positions

Alignment: track fits with the Kalman Filter

measurement m

track model h

$$r_i = m_i - h_i(x, \alpha)$$

$$\chi^2 = r^T V^{-1} r$$

covariance matrix V

Track fit with Kalman filter

↓

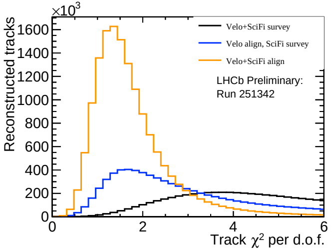
Alignment

↓

Update alignment constants

- Use survey information as starting point
- aligning the detector by minimizing the residuals of the track hits
- basically a χ^2 minimization problem with alignment parameters α
- Why Kalman Filter?
  - easily models material interactions as well as multiple scattering
- propagation of nodes, minimization, smooth error sizes by back propagation

Alignment versions in use



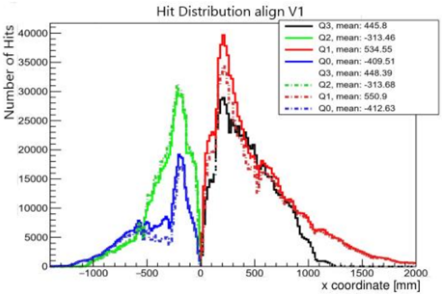
- V1: First ever SciFi alignments for the upgraded LHCb detector
- Using early tracks from comissioning
- use full length modules
- alignable degrees of freedom: Tx Rz (x translation, rotation around z →beam pipe axis)

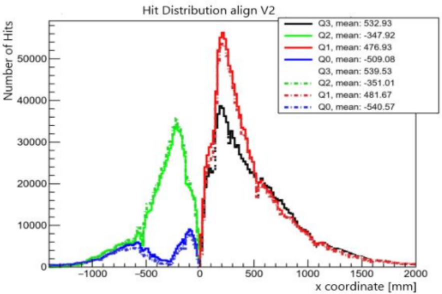
- V2: Updated alignment version with what we learned from V1
- aligned using half modules
- uses newest time alignment

Hit distribution per quarter in V1 and V2 alignment

- Improvements to V2 visible on A-side, losing some performance on C-side
- Alignment performance difference in each quarter → seperately analyse quarters!
- $\chi 2$ per quarter can provide more insights about alignment performance in each detector part

- analysis of each quarter seperately makes finding possible issues easier

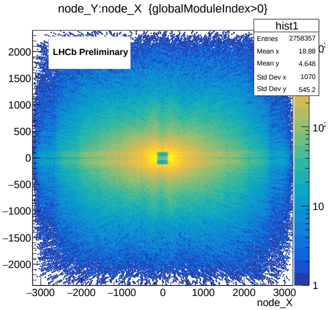


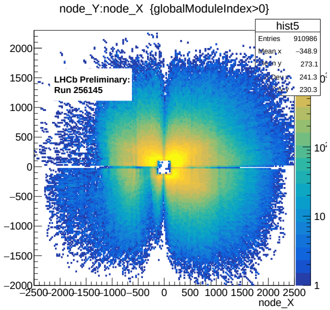


Track hits comparison of V2 and simulation

- Simulation: hits on **reconstructed** tracks fill whole detector
- data: filling tracks into A-side → good!

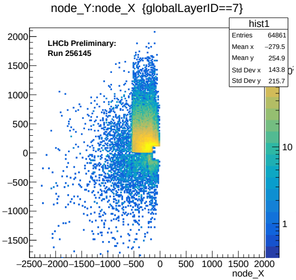
→ scan C-side quarters for possible issues in distinct layers

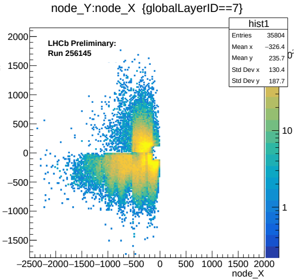


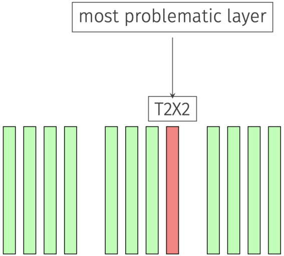


New Q0 positions in T2X2 layer

- Changes based on V2 alignment positions
- compare T2X2 constants to layers with good hit coverage →look for irregularities
- positions: translations relative to the nominal position for each module
- V2 alignment has only few tracks in Q0 because parts of the SciFi are too far out of alignment
- combining the manual scanning with a looser configuration →Alignment V3







Stability tests on 2022 data

- **How much does the SciFi move between runs?**
- **Does the magnet polarity impact playa role in the alignment?**

- Run an alignment for each of the runs on the list
- Sort the runs in ascending run number
- Compare the difference in module position for each run to the next
- Where are the modules in the local frame in all runs?

Dataset and Alignment setup

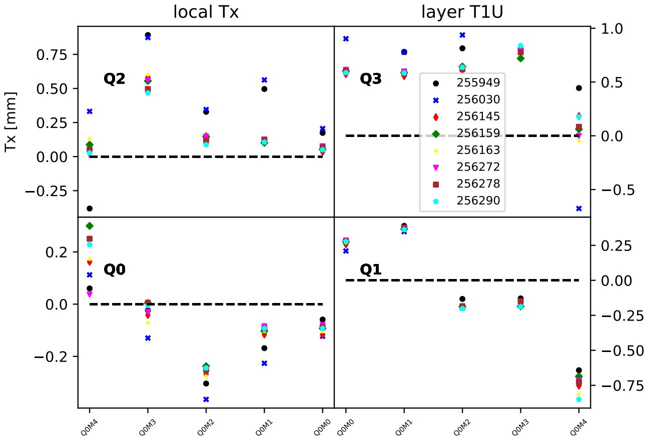
- Dataset contains magnet-up and magnet-down samples from 2022 labeled as "good" from EMTF
- Good: > 90% of datalinks are good
- Includes runs from fills: 8489, 8491, 8496

List of randomly chosen runs: 255949, 256030, 256145, 256159, 256163, 256272, 256278, 256290

- V3 Alignment from tag (loose tracking, half module alignment TxTz + Mat alignment, back layer fixed) from conditions database

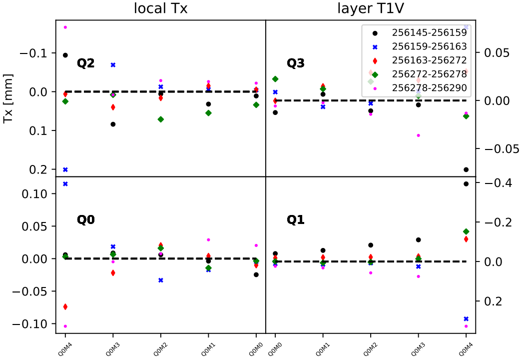
Module Positions in local half module frame

- Runs 255949 + 256030 were from fill 8489
- Optimal fine timing implemented in 256145 (afterwards)
- Positions of other runs compatible
- magUp: 256272, 256278, 256290
- magDown: 255949, 256030, 256145, 256159, 256163



Reduced dataset: removed pre timing update runs

- again: compare module positions of 2 runs but remove first 2 runs from input (different fine timing)
- Without the fine timing changes the largest movement is at max around **400μm** at most outer modules
- M4, M5 often < 1000 events (difficult for the alignment) →large movement,
- M0-M3: movement around **150μm**



stability tests: Conclusion

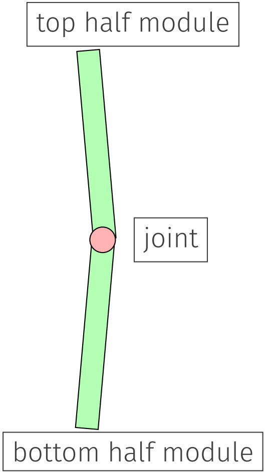
- Impactful changes like timing induces an observed movement up to 0.8mm in some cases
- The change in module position from run to run is at maximum **150μm** for the modules M0 →M3 in Tx

→only if there are no big changes between runs

- M4 moves at max **400μm** in this case
- there is no visible difference between magUp and magDown polarity
- With good SciFi timing, variation of 200 μm expected.
- A possible choice of an automatic update would be if variations of > 200 μm occur.

Joints constraints for SciFi module alignment

- Long SciFi modules: slight "banana shape"
- Half modules + joints reproduce the real shape
- Joints are implemented in the alignment by using a survey constraint (MR!368)
- it constrains parameters of 2 alignables A and B to each other with $\chi^2 = (p_A - p_B)^T V^{-1} (p_A - p_B)$
- p_A, p_B: set of parameters for half modules
- use common frame (local half modules)
- Uncertainties taken from diagonal covariance matrix →how realistic? →tuning needed
- No existing survey available for joints; tuning needed to control their χ^2



Tuning procedure

- Instead of one χ^2 $\rightarrow$look at χ^2 for joint parameters (Tx, Ty, Tz, Rx, Ry, Rz)
- Tune Uncertainties by running an alignment for each change to the respective parameter uncertainty until roughly χ^2 / DoF = 1
- make sure not to run into local minimum

- Procedure evaluated with 2023 data (run 269045, warm SciFi) and master from conditions database
- Using the alignment master

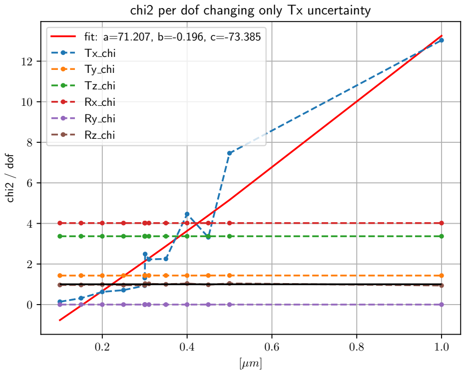
```
elements = Alignables()
elements.FTHalfModules("TxRz")

surveyconstraints = SurveyConstraints()
surveyconstraints.FT(addHalfModuleJoints=True)

constraints = []
constraints.append("BackFramesFixed : FT/T3/X2/HL.*/M. : Tx Rz")
```

Tuning of uncertainty: Tx

intersection: **0.22μm** (fit), **0.3μm** (measurement)

Initial errors:

- Tx,Ty,Tz [μm]: 1 1 1
- Rx,Ry,Rz [mrad]: 0.2 0.2 0.2
- Vary Tx uncertainty (starting at 1 μm) →run alignment →calculate χ^2/DoF values, keep other parameters at nominal!
  →Tx = **1μm** has $\chi^2 \approx$ **13**, perform a scan in a range of uncertainties to find the intersection χ^2/DoF = **1** (black line)

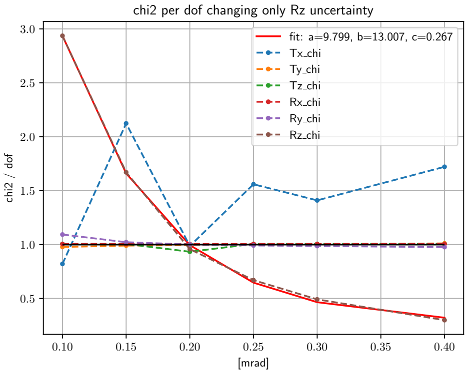


Tuning of uncertainty: Rz

intersection: **0.22μm** (fit), **0.3μm** (measurement)

Initial errors:

- In the last step Rz was tuned
- intersection at 0.2 mrad was already correctly set from nominal
- All parameters show good behaviour at chosen uncertainty
- final tunded uncertainties [μm, mrad]: 0.3 1.2 1.9 0.4 0.00000044 0.2



Summary

- Source of complications: SciFi parts too far out of alignment to be correctly updated, corrected now with new alignment version and photogrammetry
- Adapt module positions by hand: working procedure for singular low-efficiency modules
- stability tests show no substantial difference in alignment quality from magnet polarity
- sufficient statistics: **150μm** movement; rerun manually if this is exceeded
- Looser particle selection →more events, residuals comparable to strict selection, better D0 mass peak
- tuned parameters for loose selection [μm,mrad]: 0.0074 1.2 1.9 0.4 0.00044 0.22
- What does that mean for the joint constraint?
    - Constraint is mainly influenced by **Rx** and **Tz** →logical since this is the bending direction
    - **Tx** (left-to-right movement) is basically fixed since it has no impact on the constraint, same for **Ry** (rotation around vertical axis)

Backup

Loose Tracking config and D0 selection

```python
###############################################################################
# (c) Copyright 2022 CERN for the benefit of the LHCb Collaboration           #
#                                                                             #
# This software is distributed under the terms of the GNU General Public      #
# Licence version 3 (GPL Version 3), copied verbatim in the file "COPYING".   #
#                                                                             #
# In applying this licence, CERN does not waive the privileges and immunities #
# granted to it by virtue of its status as an Intergovernmental Organization  #
# or submit itself to any jurisdiction.                                       #
###############################################################################
from PyConf.Algorithms import (PrForwardTrackingVelo, PrMatchNN)
from Moore import options, run_reconstruction
from RecoConf.standalone import standalone_hlt2_light_reco_without_UT
from RecoConf.hlt2_tracking import (
    make_PrKalmanFilter_noUT_tracks,
    make_TrackBestTrackCreator_tracks,
)
from RecoConf.early_data import (
    get_loose_PrForwardTrackingVelo_params,
    get_loose_PrMatchNN_params,
    get_loose_PrKalmanFilter_params,
    get_loose_TrackBestTrackCreator_params,
)
with PrForwardTrackingVelo.bind(**get_loose_PrForwardTrackingVelo_params()), \
     PrMatchNN.bind(**get_loose_PrMatchNN_params()), \
     make_PrKalmanFilter_noUT_tracks.bind(**get_loose_PrKalmanFilter_params()), \
     make_TrackBestTrackCreator_tracks.bind(**get_loose_TrackBestTrackCreator_params()):
    run_reconstruction(options, standalone_hlt2_light_reco_without_UT)
```

pt_{min} = 800 MeV

pion, kaon required to have min pt and

IPCut = 60 μm

mass hypothesis [1760 MeV, 1960 MeV]

vertex χ^2 < 10