

Vertexing and Tracking Software at LHCb

Espen Eie Bowen^{*†}

Universitaet Zuerich

E-mail: espen.eie.bowen@cern.ch

The LHCb experiment is a dedicated heavy flavour experiment at the LHC. Its primary goal is to search for indirect evidence of New Physics in CP violation and rare decays of beauty and charm hadrons. The detector includes a high granularity silicon-strip vertex detector, a silicon-strip detector upstream of the magnet and three stations of silicon-strip detectors and straw drift tubes downstream of the magnet. The software used to perform the track reconstruction and primary vertex reconstruction is described in detail along with a discussion of its performance.

The 23rd International Workshop on Vertex Detectors

15-19 September 2014

Macha Lake, The Czech Republic

^{*}Speaker.

[†]On behalf of the LHCb collaboration.

1. Introduction

The LHCb experiment [1] is a single arm forward spectrometer that covers a pseudo-rapidity range of $2 < \eta < 5$, with the primary purpose of looking for indirect evidence of New Physics in CP violation and rare decays of beauty and charm hadrons. The unique geometry takes advantage of the large b - and c -quark production in the forward region at the LHC. The following requirements are crucial in order to fulfill the LHCb physics programme: excellent tracking (momentum, impact parameter and primary vertex resolution), precise decay time resolution and excellent particle identification.

The tracking system consists of a high granularity vertex detector (VELO) surrounding the pp interaction region [2], a large area silicon-strip detector located upstream of a dipole magnet with a bending power of about 4 Tm, and three stations of silicon-strip detectors and straw drift tubes placed downstream of the magnet [3]. The VELO is formed from 42 layers of silicon-strip detectors with a r - ϕ geometry. It has two retractable halves which are only 8 mm from the beam when closed. The TT and each T-station consist of 4 layers with a $(0^\circ, -5^\circ, +5^\circ, 0^\circ)$ layout. The inclined layers allow stereo measurements to be made.

2. Track reconstruction

The track reconstruction in LHCb is performed by several different algorithms. In order to describe the process, it is first necessary to introduce the notion of track types and track states which are described in Sections 2.1 and 2.2 respectively. Each of the tracking algorithms used in LHCb are described in detail in Section 2.3. Finally, the duplicate track removal and track fit procedures are described in Sections 2.4 and 2.5 respectively.

2.1 Track types

The geometry of LHCb, allows the definition of several different track types depending on the sub-detectors in which the tracks have measurements. A schematic diagram of the LHCb tracking system along with each of the track types is shown in Figure 1. The following definitions are used:

- **Long tracks:** Traverse full tracking system from VELO to the T-stations. As they have the most accurate momentum estimate, they are often the most useful for physics analysis.
- **VELO tracks:** Have hits in both the r - and ϕ -sensors of the VELO but are not matched to hits in other sub-detectors. They can be at large polar angles or backwards and are used for primary vertex reconstruction.
- **Upstream tracks:** Have hits in the VELO and TT only. Often low momentum particles that are bent out of acceptance by the magnetic field.
- **T tracks:** Only reconstructed in the T-stations. Can originate from very long-lived particles or material interactions.
- **Downstream tracks:** Have hits in the TT and T-stations. Allow the reconstruction of charged daughters of long-lived particles (K_S^0 , Λ) with a decay vertex displaced from the interaction point.

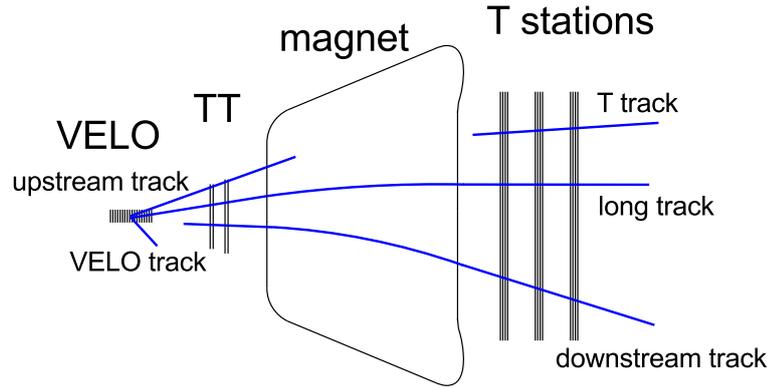


Figure 1: Schematic diagram of the LHCb tracking system. The various track types reconstructed by the different tracking algorithms are shown.

2.2 Track states

In the LHCb software, a track is defined as of a series of straight line segments called track states. As tracks are regarded as going in either the forward or backward direction, track states are parameterised as a function of z by a state vector:

$$\vec{x} = \begin{pmatrix} x \\ y \\ t_x \\ t_y \\ q/p \end{pmatrix} \quad \text{with } t_x = \frac{\partial x}{\partial z} \text{ and } t_y = \frac{\partial y}{\partial z} \quad (2.1)$$

where q and p are the charge and momentum of the track respectively, and a corresponding 5×5 state covariance matrix.

2.3 Tracking algorithms

In order to reconstruct the different track types, several tracking algorithms are employed. The two stand-alone algorithms, VELO tracking and track seeding, are described in Sections 2.3.1 and 2.3.4 respectively. The other algorithms use input from these two algorithms in order to perform a further track reconstruction.

2.3.1 VELO tracking

The VELO tracking algorithm [4] is used to find tracks in the VELO. As there is no magnetic field in the VELO, tracks are expected to be approximately straight lines. The track search begins in the most downstream layer of the VELO. Quadruplets of hits are searched for in the r -sensors as shown in Figure 2. If they are found, they are extended back to smaller z adding hits that are consistent with coming from the same track. Next, the same quadruplet search is performed for backward-going tracks. Triplets are then searched for, first backward-going and then forward-going, requiring that the hits have not been used in the quadruplet search.

Starting from the longest r - z track, ϕ hits are searched for that are consistent with coming from the same track. These 3D tracks are then fitted with a χ^2 minimization.

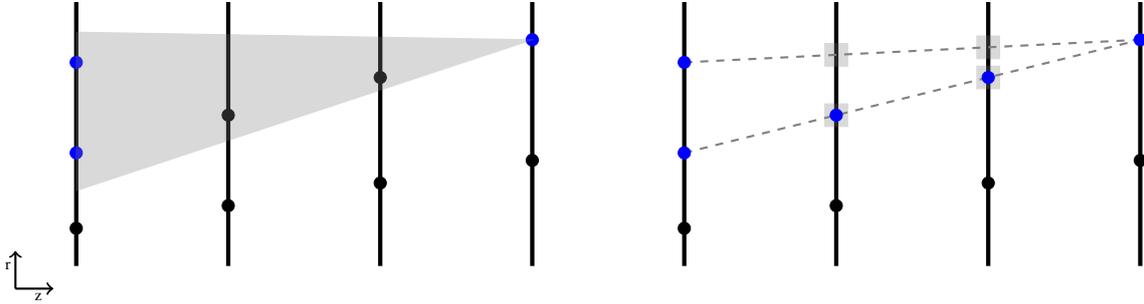


Figure 2: Quadruplets of hits are searched for in the VELO starting from the most downstream layer. Starting with a hit in that layer, a search window is opened in the fourth most downstream layer. From hits found within the window, the expected position of hits in the intermediate layers can be predicted assuming the track is a straight line in the r - z projection. If hits fall within a tolerance of the expected positions, quadruplets are formed and a track is created.

2.3.2 Forward tracking

The Forward tracking algorithm [5] is used to find long tracks. A Hough transform is utilised to associate hits in the T-stations to each VELO track. The VELO track is linearly extrapolated to the T-stations and a search window is opened in each x layer. The VELO track state and knowledge of the \vec{B} field are used to project each selected hit to the z position of a reference plane. Hits from the same particle are expected to be projected to the same x position while random hits should be uniformly distributed. This procedure is shown schematically in Figure 3. The resulting clusters are fitted and outliers are removed using a χ^2 criterium. An additional cluster search is used to add stereo hits that are consistent with the x - z track. This 3D track is then fitted, outliers are removed and the best track candidate is chosen based on its χ^2/dof .

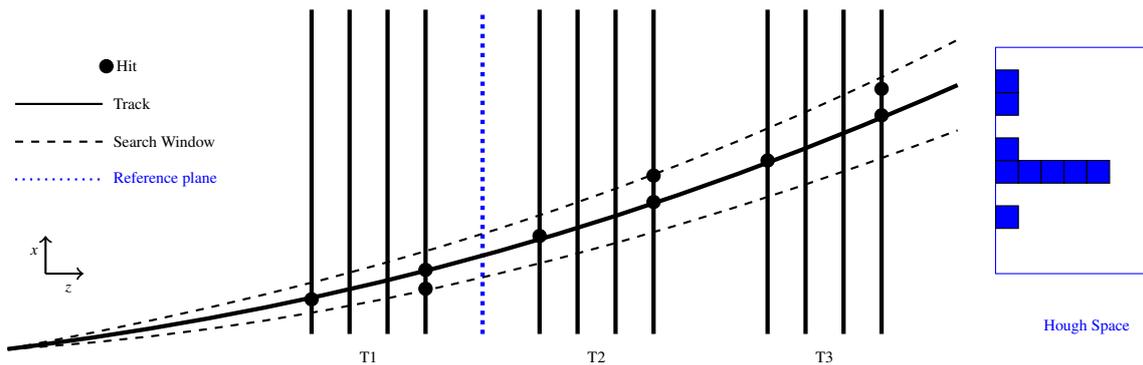


Figure 3: A Hough transform is used to associate hits in the T-stations to a VELO track. Each hit within a search window around the extrapolated track is projected to the z position of a reference plane. Hits from the same particle are expected to be projected to the same x position while random hits should be uniformly distributed.

2.3.3 Upstream tracking

The upstream tracking algorithm [6] is used to find upstream tracks. Each VELO track is linearly extrapolated to the TT. Search windows are opened in each layer and the distance Δx between the track and each hit is calculated. These Δx values are scaled to a reference plane at the center of the TT. Clusters are chosen that are consistent with coming from the same VELO track. Each track candidate is fitted with a χ^2 minimisation and the q/p of the track is estimated. Due to the fringe \vec{B} field between the VELO and the TT a momentum estimate of $\delta p/p \sim 15\%$ is possible. The best track candidate is chosen based on the number of TT layers containing measurements and the χ^2 of the fit.

2.3.4 T seeding

The T seeding algorithm [7] is used to find T tracks. Track candidates are first searched for in the x - z projection. A straight line is formed between suitable pairs of hits in T1 and T3. A compatible hit in T2 is added to form a parabola. Further hits in x layers consistent with this parabola are added to the track candidate. A Hough transform is used to add stereo hits. A weighted least squares fit is then applied to each candidate.

2.3.5 Track matching

The track matching algorithm is also used to form long tracks. It takes both VELO and T tracks as input (seeds). The difference in x and y of the two seeds are calculated by extrapolating them both to the magnet bending plane (Δx) and the end of the T-stations (Δy) respectively. A matching criterion χ^2 is formed using Δx , Δy , Δt_x and Δt_y . If the track passes this criterion it is fitted and an estimate is made of its q/p .

2.3.6 Downstream tracking

The downstream tracking algorithm [8] forms downstream tracks. Each T track is extrapolated back to find the corresponding (x, y) point at the center of the magnet. A track estimate is formed using this point and the nominal interaction point. Hits in the TT consistent with the track estimate are selected. For each TT hit, a new track estimate is formed and consistent x hits are collected. The collection of x hits is fitted in the x - z projection and outliers are removed. Stereo hits are added, the track is refitted and further outliers are removed. Finally, the best track candidate is chosen according to the number of hits it contains and the value of the χ^2 from the fit.

2.4 Duplicate track removal

As there are two independent algorithms to produce long tracks and several track types are subtracks of other types, it is necessary to avoid or remove duplicate tracks found by multiple algorithms. This is accounted for in two different ways. Some algorithms are only allowed to use tracks or hits that have not been previously used. When there is a significant overlap of hits between two tracks, the track with the smaller number of hits is discarded.

2.5 Kalman filter

The purpose of the track fit is to obtain the most accurate estimates of track parameters along with their corresponding covariances. Track parameters are used to match to particle identification objects (e.g. Cherenkov rings), find primary and secondary vertices and calculate the kinematics and invariant masses of particle combinations. The track χ^2 is used to select good quality tracks.

A Kalman filter is used to fit the tracks. With this approach, multiple scattering is taken into account as process noise and corrections due to energy losses are applied [9]. The transport through the magnetic field is evaluated using a Runge-Kutta method. The propagation and projection functions are linearised around a reference track state using a Taylor expansion [10].

Track candidates can be considered as a collection of track states (initially provided by the individual tracking algorithms) and measurements (tracking station hits). The Kalman filter can be divided into two steps, shown schematically in Figure 4. Firstly, the parameters of a state at z_k are predicted given a state at z_{k-1} . Next, the state at z_k is updated with information of the measurement at this position. These two steps are repeated until all the measurements have been added. In LHCb the filter is run in both the forward and the backward directions and the average is taken for smoothing.

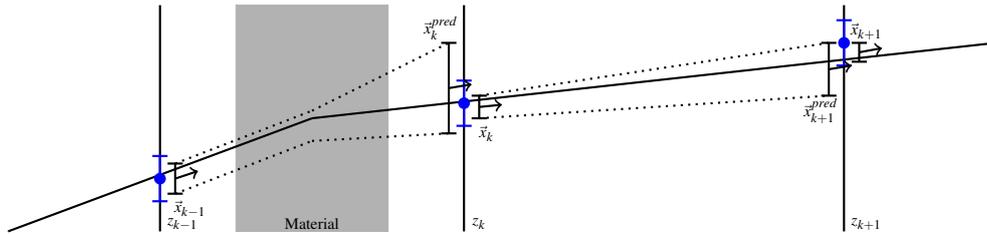


Figure 4: A schematic diagram of the Kalman filter showing the prediction of a state z_k given a state at z_{k-1} . The state at z_k is subsequently updated with information of the measurement at this position. This process is repeated until all measurements have been added.

3. Primary vertex reconstruction

Primary vertex (PV) reconstruction [11] consists of two stages: seeding and fitting. Firstly, as described in Section 3.1, a set of PV candidates is found by looking for points at which a sufficient number of tracks pass close to each other. These points become seeds for the subsequent PV fit, described in Section 3.2. Reconstructed seeds are sorted according to decreasing multiplicity in order to reduce the incorrect reconstruction of secondary B-meson vertices as primary vertices.

3.1 PV seeding

The purpose of the PV seeding is to find the PV candidates. All the reconstructed tracks in the event are looped over. For each track (base track), other tracks are selected that have a distance of closest approach below a certain threshold. If there is an insufficient number of selected tracks, the next base track is considered. For each selected track, the point of closest approach (POCA) to the base track is determined. The average POCA is determined in a two step approach. Firstly, the truncated mean is calculated iteratively. Secondly, if sufficient tracks remain after truncation,

the weighted average is calculated. All tracks used to calculate the weighted average are marked as used and the loop over the remaining tracks continues. The result is a set of (x, y, z) points which become the seeds to the PV fit.

3.2 PV fitting

The standard least squares iterative procedure is not optimal for PV fitting. At this stage tracks originating from B decays are not known and are likely to be assigned to the PV. This would cause a systematic shift of the PV toward the secondary vertex.

In order to account for this effect, an adaptive least squares method using the Tukey biweight is used. A weight is assigned to a track according to its value of its χ_{IP}^2 ¹ and the Tukey constant C_T :

$$\begin{aligned} W_T &= (1 - \chi_{IP}^2/C_T^2)^2, & \text{if } \chi_{IP}^2 < C_T^2 \\ W_T &= 0, & \text{if } \chi_{IP}^2 > C_T^2 \end{aligned} \quad (3.1)$$

The position of the reconstructed PV is determined iteratively by minimizing:

$$\chi_{PV}^2 = \sum_{i=1}^{n_{tracks}} \chi_{IP,i}^2 \times W_{T,i} \quad (3.2)$$

until the convergence criteria are satisfied: (1) the shift in the z position of the PV is below a threshold value and (2) there are a sufficient number of tracks associated to the PV.

4. Performance

4.1 Track reconstruction efficiency

The long track reconstruction efficiency can be defined as the probability that the trajectory of a charged particle that has passed through the full tracking system is reconstructed. This can be measured in data using a tag-and-probe method with $J/\psi \rightarrow \mu^+ \mu^-$ decays [12]. One of the daughters is fully reconstructed (tag), while the other is only partially (probe), although well enough to reconstruct the J/ψ invariant mass. The efficiency can then be measured by matching the probe track to a fully reconstructed long track. The average efficiency is found to be over 95% and is only slightly affected in high multiplicity events.

4.2 Mass and momentum resolution

The momentum resolution for long tracks can be measured in data using $J/\psi \rightarrow \mu^+ \mu^-$ decays [13]. For two muons of similar momentum, neglecting muon masses, the momentum resolution can be approximated as:

$$\left(\frac{\delta p}{p}\right)^2 = 2\left(\frac{\sigma_m}{m}\right)^2 - 2\left(\frac{p\sigma_\theta}{m\theta}\right)^2, \quad (4.1)$$

where m is the J/ψ mass, σ_m is the Gaussian width obtained from a mass fit and the second term corrects for the opening angle θ between the two muons where σ_θ is the mean per-event error on

¹The χ_{IP}^2 is the value by which the vertex χ^2 increases when adding the track to the vertex.

θ . The relative momentum resolution, $\delta p/p$ is found to be between 0.4% - 0.6% for tracks up to 100 GeV/c.

The mass resolution is determined from data by studying the J/ψ , $\psi(2S)$, Υ and Z resonances. The relative mass resolution, σ_m/m , is found to be about 5 per mille up to the Υ masses.

4.3 Vertexing and decay time resolution

Precise vertex resolution is important to allow the separation of primary and secondary decay vertices. The primary vertex resolution depends strongly on the number of tracks used to form it. It can be measured in data in an event-by-event manner by randomly splitting the track sample in two and reconstructing the PV using each independent set of tracks. In 2011 data, a 25-track vertex was found to have a resolution of 13 μm in x and y and 71 μm in z [2].

While the reconstructed decay time of charm and beauty hadrons is used in offline selections and for precise measurements of lifetimes, the most stringent requirement on the decay time resolution originates from the need to resolve the fast B_s^0 - \bar{B}_s^0 oscillations in mixing. The decay time resolution is topology dependent and is calibrated in data for each final state using prompt combinations that fake the signal candidates. The shape of the prompt decay time distribution is determined only by the resolution function. The typical decay time resolution is 45 fs for a 4-track vertex.

5. Conclusions

The track reconstruction and primary vertex reconstruction of LHCb perform to a very high standard in the challenging environment of the LHC. The various tracking algorithms described are able to provide a tracking efficiency above 95% with a relative momentum resolution between 0.4% - 0.6% for tracks up to 100 GeV/c, which corresponds to a relative mass resolution of about 5 per mille up to the Υ masses. The primary vertex reconstruction was found to give a resolution of 13 μm in x and y and 71 μm in z for a 25-track vertex and a typical decay time resolution of 45 fs for a 4-track vertex.

References

- [1] LHCb Collaboration, “The LHCb detector at the LHC”, *JINST* 3 (2008) S08005.
- [2] LHCb VELO group, “Performance of the LHCb Vertex Locator”, *JINST* 9 (2014) P09007.
- [3] LHCb Outer Tracker group, “Performance of the LHCb Outer Tracker”, *JINST* 9 (2014) P01002.
- [4] O. Callot, “FastVelo, a fast and efficient pattern recognition package for the Velo”, [LHCb-PUB-2011-001](#).
- [5] O. Callot and S. Hansmann-Menzemer, “The Forward Tracking: Algorithm and Performance Studies”, [LHCb-2007-015](#).
- [6] O. Callot, “VELO-TT track reconstruction”, [LHCb-2007-010](#).
- [7] O. Callot and M. Schiller, “PatSeeding : A Standalone Track Reconstruction Algorithm”, [LHCb-2008-042](#).
- [8] S. Stahl, “Reconstruction of displaced tracks and measurement of K_s production rate in proton-proton collisions at $\sqrt{s} = 900$ GeV at the LHCb experiment”, Diploma thesis, June 2010.

- [9] R. Frühwirth, “Application of Kalman filtering to track and vertex fitting”, [Nucl. Instrum. Meth. A262 \(1987\) 444](#).
- [10] J. van Tilburg, “Track simulation and reconstruction at LHCb”, PhD thesis, October 2005.
- [11] M. Kucharczyk, P. Morawski, M. Witek, “Primary Vertex Reconstruction at LHCb”, [LHCb-PUB-2014-044](#).
- [12] LHCb Collaboration, “Measurement of the track reconstruction efficiency at LHCb”, [arXiv:1408.1251](#).
- [13] LHCb Collaboration, “LHCb Detector Performance”, LHCb-DP-2014-002, In preparation.