technische universität
dortmund

Arbeit zur Erlangung des akademischen Grades
Master of Science

Optimisation of the software-based muon identification at the LHCb experiment

Kevin Dungs

18. September 2015

Lehrstuhl für Experimentelle Physik V
Fakultät für Physik
Technische Universität Dortmund

Erstgutachter: Dr. Johannes Albrecht
Zweitgutachter: Prof. Dr. Kevin A. Kröninger

Abstract

The LHCb experiment performs precision tests of the standard model of particle physics. A significant part of the physics programme of the experiment is based on final states containing muons. In this work, the performance of the LHCb experiment's muon identification in Run I is examined. Using simulated $B^0 \to K^{*0}\mu^+\mu^-$ candidates, the relative inefficiency of the first stage of the software trigger with respect to the offline reconstruction is evaluated. For single muon tracks, an inefficiency of $(9.4 \pm 0.4)\,\%$ which stems only from the reconstruction software is found. Thanks to an upgraded computing farm, more CPU time can be spent on reconstruction. Through removal of simplifications and other optimisations, the software-based inefficiency is reduced by $(8.43 \pm 0.30)\,\%$ for Run II. This means, that $(89.7 \pm 5.0)\,\%$ of the software-based inefficiency are removed. The underlying software is modularised such that in the online farm the identical algorithm from offline processing can be used. The software developed in this thesis is used in all future processing and triggering aiming to identify muon. As a side effect, the offline reconstruction is now roughly 2.5 times faster.

Abstract (Deutsch)

Das LHCb-Experiment beschäftigt sich mit Präzisionstests des Standardmodells der Teilchenphysik. Ein wichtiger Teil des Physikprogramms sind dabei Teilchenzerfälle deren Endzustände Myonen enthalten. In dieser Arbeit wird die Leistungsfähigkeit der Myonenrekonstruktion des LHCb-Experimentes in der ersten Datennahmephase des LHC untersucht. Mit Hilfe simulierter $B^0 \to K^{*0}\mu^+\mu^-$ Zerfälle wird die relative Ineffizienz der ersten Stufe des Softwaretriggers relativ zur Offlinerekonstruktion ermittelt. Für einzelne Myonenspuren wird eine Ineffizienz von $(9.4 \pm 0.4)\,\%$ gefunden, die sich allein aus der Rekonstruktionssoftware ergibt. Dank einer verbesserten Computerfarm kann mehr CPU-Zeit für die Rekonstruktion verwendet werden. Durch eine Optimierung der Rekonstruktionssoftware wird diese Ineffizienz für die kommende Datennahmephase um $(8.43 \pm 0.30)\,\%$ reduziert. Das bedeutet, dass der Großteil aller Ineffizienzen in der Onlinerekonstruktion von Myonen entfernt werden konnte. Die zugrundeliegende Software wird soweit modularisiert, dass in der Onlinefarm der identische Algorithmus der Offlinerekonstruktion benutzt werden kann. Die in dieser Arbeit entwickelte Software wird in Zukunft in der standard Online- und Offlineprozessierung des LHCb-Experimentes eingesetzt, wenn es darum geht, Myonen zu identifizieren. Ein Nebeneffekt dieser Arbeit ist, dass die Offlinerekonstruktion um einen Faktor zweieinhalb schneller läuft.

Contents

1 Introduction

The LHCb experiment is one of the four big experiments at CERN, the European Organisation for particle physics. It is, among other topics, concerned with the analysis of beauty and charm decays in order to find deviations from the standard model of particle physics. Many of the decays sensitive to the so called "new physics" are characterised by final states that contain muons. In Run I of the LHC, no significant deviations from the standard model have been found. Some channels, e.g. the $B^0 \to K^{*0}\mu^+\mu^-$ decay[1] [1], show some slight excesses that could be a hint for new physics. In order to achieve statistical significance, more data is needed. Especially for muons, excellent reconstruction efficiencies are necessary. Thus improving the performance of the muon reconstruction directly improves the physics performance of the aforementioned and many other analyses.

The trigger system of the LHCb experiment is of crucial importance in handling the vast amounts of data produced at the LHC. At 20 MHz inelastic collision rate, the trigger reduces the rate to 12.5 kHz at which data is written to storage. Thus 99.90 % of the visible events are rejected. If an event is discarded that contains muons which could have otherwise been reconstructed, possible physics output is lost. As can be seen in Figure 1.1, previous studies with $B^+ \to J/\psi K^+$ candidates found significant inefficiencies in the online muon reconstruction in Run I (2012–2012). Most of those inefficiencies can be attributed to simplifications that were made due to the very tight timing budget. Improved computing resources for Run II (2015–2018) allow to re-evaluate the choices.

In this work, the performance of the muon triggers in Run I is analysed in detail. Based on those values, several aspects of the muon identification are optimised for the larger timing budget in Run II. Additionally, the code is modularised to be used both in the software trigger as well as the offline reconstruction.

[1] Whenever applicable, charge conjugation is assumed implicitly.

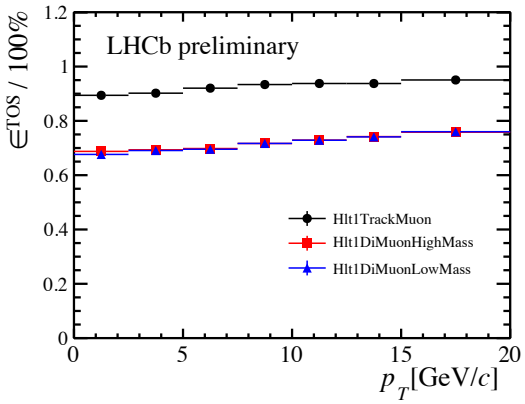


Figure 1.1: HLT1 muon trigger performance in Run I: Efficiency for finding $B^+ \to J/\psi K^+$ candidates with respect to transverse momentum of the B^+. Figure reproduced from [2].

2 The LHCb experiment at the LHC

Located at the border between Switzerland and France close to Geneva, CERN, the European organisation for particle physics, is home to the Large Hadron Collider (LHC). The LHC is a circular particle accelerator capable of colliding proton bunches separated by 25 ns. At the start of Run II, the centre-of-mass energy is 13 TeV. Large quantities of particles containing beauty and charm quarks which are of interest to the LHCb experiment are produced.

Named after the b-quark, LHCb is one of the four big experiments at CERN. Its main focus is the study of particle decays involving beauty and charm quarks. Other than the general purpose detectors ATLAS and CMS, LHCb is a forward spectrometer with a limited angular acceptance of $1.8 < \eta < 4.9$ designed for precision in that region. At $\sqrt{s} = 13\,\text{TeV}$, that region is expected to contain 24 % of all $b\bar{b}$ quark pairs produced in the collisions at the interaction point as can be seen in Figure 2.1. As shown in Figure 2.2, the LHCb detector is comprised of several layers. It can be fully read out at a rate of 1 MHz. A comprehensive description of the detector and its performance can be found in references [3, 4].

In 2009, the LHCb collaboration published a roadmap naming six of the key measurements for the experiment [5]. Half of the quoted decay modes contain muons in their final states and thus benefit greatly from an improved muon reconstruction. Namely the decay modes are $B_s^0 \to J/\psi\phi$, $B_s^0 \to \mu^+\mu^-$, and $B^0 \to K^{*0}\mu^+\mu^-$.

2.1 Track reconstruction

The muon reconstruction of the LHCb experiment relies on the reconstruction of tracks produced by charged particles in the detector. The tracking system of the LHCb detector has an average efficiency of over 96 % in the momentum range $5\,\text{GeV/c} < p < 200\,\text{GeV/c}$ for tracks that pass through the whole tracking

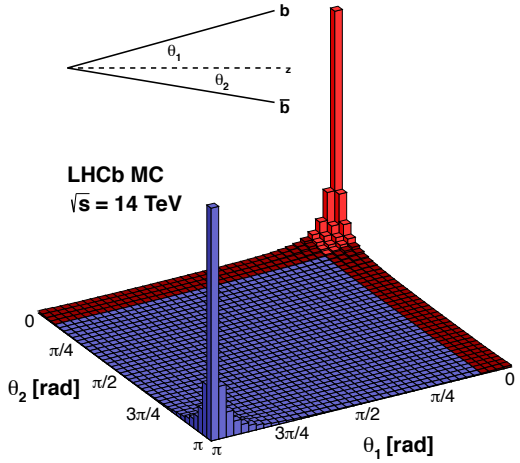


Figure 2.1: Angular distribution of $b\bar{b}$ production at the LHC from simulation for $\sqrt{s} = 14\,\text{TeV}$. Figure reproduced from [6]. It can be seen that the distribution has strong peaks in the forward and backward directions. The acceptance of the LHCb detector is highlighted in red. It covers 24 % of the produced $b\bar{b}$ pairs.

system [4]. It consists of the Vertex Locator (VELO) and the four tracking stations (TT, T1–T3) together with the magnet. The magnet is a warm dipole magnet providing an integrated field of about 4 Tm. It is needed for the momentum measurement of charged particles as it deflects them in the horizontal plane. The VELO is built around the interaction point and able to get as close as 7 mm to the beam line. It provides excellent primary vertex resolution as well as a decay time resolution of 45 fs. For the Tracker Turicensis (TT), which is located upstream of the magnet, silicon microstrips are used. The other three tracking stations T1–T3 are located downstream of the magnet. Close to the beam pipe they also consist of silicon strips while further out straw tubes are used. The inner part is named the Inner Tracker (IT) while the outer part is called the Outer Tracker (OT). Only particles with a momentum above 1.5 GeV/c reach the tracking stations T1–T3 as the tracks low momentum particles are bent out of the detector acceptance by the magnet.

As can be seen in Figure 2.3, different track types are defined depending on which parts of the detector corresponding hits were found in. The `FastVelo` algorithm [8] creates track segments from hits in the VELO in two stages. The first stage uses quadruplets of hits and is very fast and clean. The second stage picks up left over hits and tries to reconstruct additional tracks from triplets of hits. Long tracks are built from VELO tracks by the forward tracking algorithm [9] or the track matching algorithm [10]. The former combines a VELO track with a single hit in

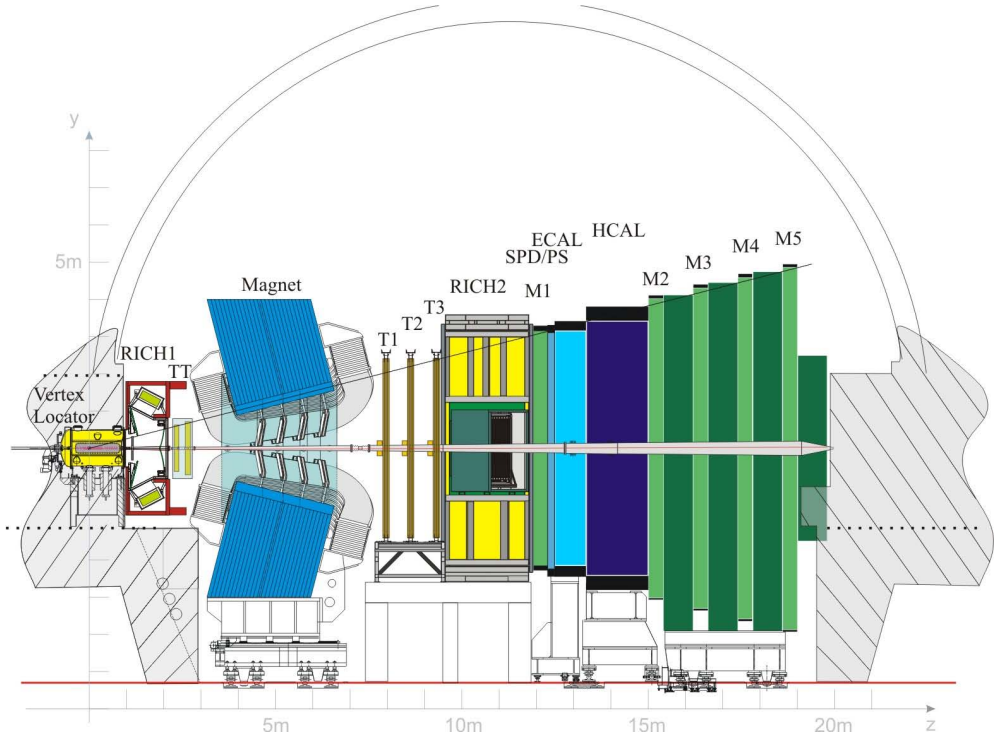


Figure 2.2: Schematic view of the LHCb detector. Figure reproduced from [7]. The beam pipe traverses the detector in the middle. The interaction point is located at the left edge, surrounded by the VELO. A right-handed cartesian coordinate system is defined with its origin in the interaction point and its z-axis pointing down the detector along the beam axis. The y-axis is chosen to point vertically upwards from the floor. The magnet bends tracks of charged particles in the x-z-plane.

the T stations to determine a trajectory and momentum and adds additional hits successively. The track matching combines a VELO track with an independently reconstructed track segment in the T stations. This is done by the `PatSeeding` algorithm [11]. It adds TT information to improve the momentum estimate. Downstream tracks are reconstructed by extrapolating T track segments backwards through the magnetic field and searching for matching hits in the TT. All tracks are fitted using a Kalman filter [12] that takes into account the detector material. The fit yields a χ^2 that can be used for quality cuts. More detailed information can be found in the above mentioned references as well as in reference [3].

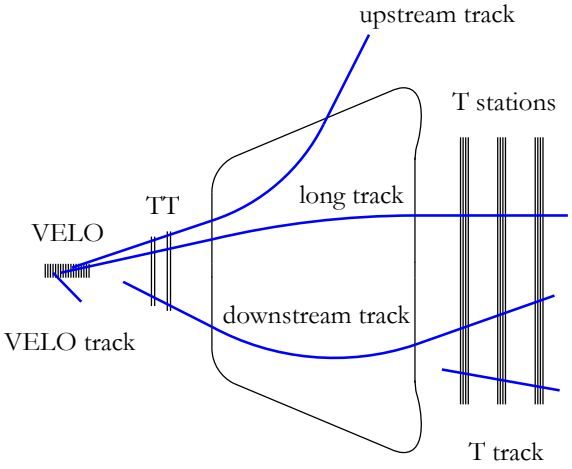


Figure 2.3: Overview of track types in LHCb. Figure reproduced from [13]. For the muon reconstruction, only long tracks and downstream tracks are used.

2.2 Muon system

The muon system is located further downstream from the tracking system. It comprises five stations (M1–M5). M1 and M2–M5 are separated by the calorimeter system (comprised of the ECAL and the HCAL as well as the SPD and the PS) which in this context function only as shielding. Additionally, the stations M2–M5 are interspersed with iron filters as shown in Figure 2.4. The muon system has an angular acceptance of 20 mrad to 206 mrad in the bending plane and 16 mrad to 258 mrad in the non-bending plane [14]. Compared to the full solid angle this corresponds to a geometrical acceptance of 20 % for b quarks. The stations M2–M5 contain multi wire proportional chambers (MWPC) that can be read out in terms of logical strips in two dimensions. The intersection of a vertical and a horizontal

strip defines a rectangular logical pad. The number of logical pads in each region of each station is chosen according to a projective geometry that takes into account the interaction of particles with the absorber material: The sizes of each region and channel scale with a factor two from one region to the next while the transverse dimensions of the stations scale with the distance to the interaction point. The separation in regions, chambers, and pads is visualised in Figure 2.5. As shown in Figure 2.6, hits in logical pads that are confirmed by both the horizontal and vertical readout channel are called crossed hits while those for which only one of the two has produced a signal are called uncrossed hits. This definition is important for the muon reconstruction algorithms. Further details on the muon system can be found in the corresponding technical design report [15] and the two addenda [16, 17].

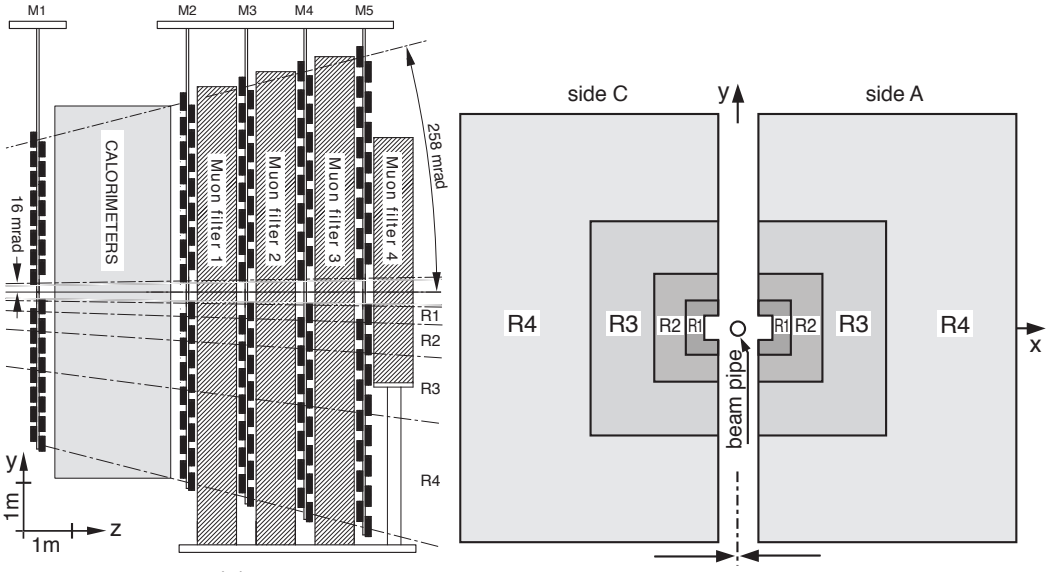


Figure 2.4: Side view of the LHCb Muon Detector (left) as well as a schematic view of the separation into regions (right). Figures reproduced from [18]

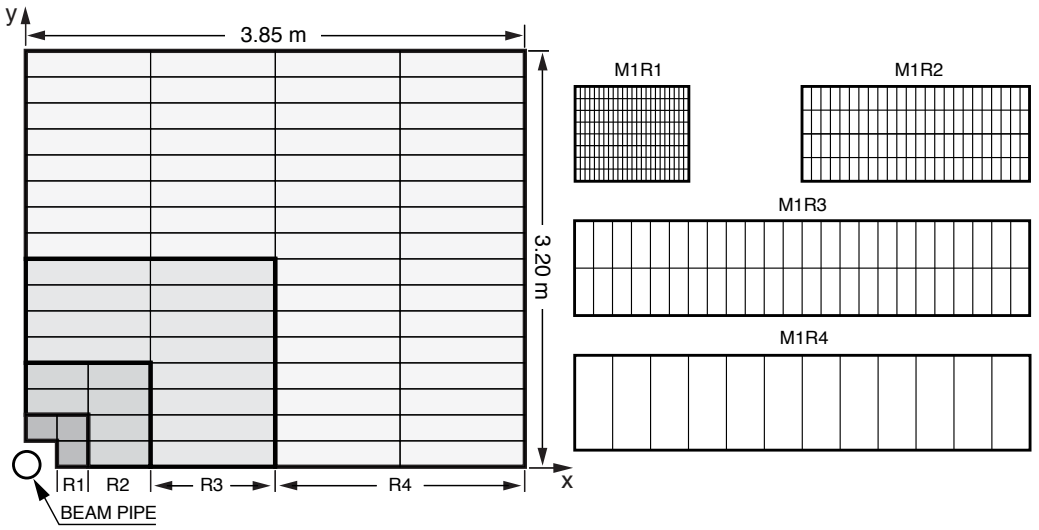


Figure 2.5: Separation of M1 into regions, chambers, and pads. Figure reproduced from [18]. Each station is separated into four regions. The dimensions of the regions scale with a factor two. R1, close to the beam pipe, is the smallest region, while R4 is the largest. Each region consists of chambers (MWPC for M2–M5) which can be read out in logical channels both horizontally and vertically. Pads are defined as the intersecting area of two crossing channels. For M2 and M3, the number of pad columns per chamber is double the number in M1 while for M4 and M5 it is half.

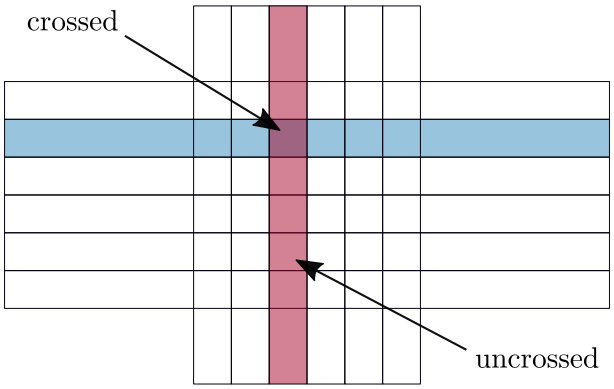


Figure 2.6: Visualisation of the difference between crossed and uncrossed hits. A hit in a muon station is considered a crossed hit if it is registered both by a horizontal and a corresponding vertical strip. If a hit is only seen by either, it is considered uncrossed. Uncrossed hits have, by construction, much larger pad sizes and thus uncertainties.

3 Muon reconstruction

The idea behind the muon reconstruction algorithm at the LHCb experiment can be summarised as follows: A reconstructed track is extrapolated into the muon stations. Around that extrapolation in each station, hits are searched for within a window. If hits are found in a sufficient number of stations, the track is associated with a muon. Further quantities are calculated per track in order to help distinguish muons from non-muonic background. For each event, containing a number of tracks, the software inspects tracks individually. However, one of the quantities, `nShared`, reflects relationships between the tracks. All of the variables are described in detail below. Of those, only `IsMuon` is used in the first stage of the software trigger. Table A.3 in the appendix contains an overview of all the variables, their types, and their locations.

3.1 IsMuon

The central logic of the muon reconstruction is an algorithm called `IsMuon` that calculates the corresponding boolean variable `IsMuon`. Based on an extrapolation of a long track into the muon stations it makes a statement about whether a track is consistent with a muon hypothesis. The extrapolation yields expected coordinates in the muon stations M2, M3, M4, and M5. Only extrapolations within the acceptance of the muon stations are considered. For each of the stations, a search for hits within an elliptic, momentum dependent field of interest (FoI) around the extrapolated hit is performed. Three parameters for each dimension $\rho_{\{x,y\}\times\{1,2,3\}}$ define a FoI,

$$\mathrm{FoI}_a(p) = \rho_{a,1} + \rho_{a,2} \cdot \exp\left(\frac{-\rho_{a,3} \cdot p}{\mathrm{GeV/c}}\right), \tag{3.1}$$

with $a \in \{x, y\}$. A hit (h_x, h_y) with corresponding pad dimensions $\mathrm{pad}_{\{x,y\}}$ is considered to be within the field of interest around an extrapolation (e_x, e_y) if

$$|h_x - e_x| < \mathrm{pad}_x \cdot \mathrm{FoI}_x \quad \text{and} \quad |h_y - e_y| < \mathrm{pad}_y \cdot \mathrm{FoI}_y\,. \tag{3.2}$$

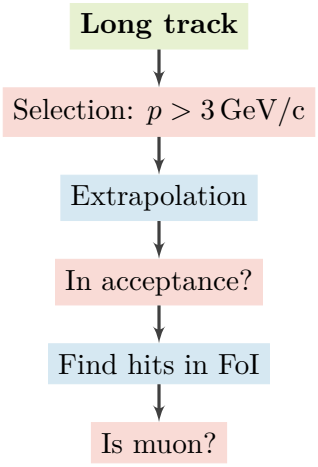


Figure 3.1: Simplified flow of the `IsMuon` algorithm. A long track with $p > 3\,\mathrm{GeV/c}$ is extrapolated into the muon stations. If the extrapolation is within the acceptance, hits within the FoI are searched for. If enough stations have hits, the attribute `IsMuon` is assigned.

It is important to note that the pad size depends on whether the hit is a crossed or an uncrossed hit (see Figure 2.6). Therefore the dimensions of the pad need to be obtained in a way that guarantees different treatment of crossed and uncrossed hits. Depending on the momentum of the long track, the track is given the attribute `IsMuon` if enough stations are found to have at least one hit within the FoI. Table 3.1 gives an overview of the required stations with respect to the track's momentum. The selection cut of $p > 3\,\mathrm{GeV/c}$ is chosen because that is the momentum at which roughly 50 % of the muons reach M3.

Similar to `IsMuon`, `IsMuonLoose` makes a statement about whether a track is consistent with a muon hypothesis. The requirements are less strict than for `IsMuon`. They are also listed in Table 3.1. A third variable, `IsMuonTight` can be calculated with the same algorithm and requirements as `IsMuon` but in this case the search is limited to crossed hits only.

3.2 NShared

Each muon track has a property called `nShared`. It is an unsigned integer that helps distinguish between actual tracks and potential ghost tracks. For each hit within the FoI of a given extrapolation into the muon chambers, the algorithm

Table 3.1: Required stations with hits within FoI for `IsMuon` and `IsMuonLoose` with respect to track momentum [14]. When `IsMuonLoose` was first introduced, probabilistic weights were used for each hit. Since they depend on random number generation, reproducibility is lost. When they were removed, the separation of the lower momentum bin was introduced in order keep the underlying hypothesis that hits in M4 and M5 associated to a track with $p < 3.5\,\mathrm{GeV/c}$ are noise.

p [GeV/c]	Required stations	
	`IsMuon`	`IsMuonLoose`
$<$ 3	*Always false*	*Always false*
$\leq$ 3.5	–	M2 & M3
$<$ 6	M2 & M3	At least two of M2–M4
<10	M2 & M3 & (M4 | M5)	–
	M2 & M3 & M4 & M5	At least three of M2–M5

checks whether any of the other tracks has already used that hit. If so, the value of `nShared` for the track with the larger sum of squared distances between the extrapolation and the corresponding closest hit is incremented. This principle is illustrated in Figure 3.2. It follows that a track with an `nShared` of zero is likely to be a real muon while one with a high `nShared` is likely to be a fake muon.

3.3 Combined information

As shown in reference [14], the average squared distance in units of pad size between the track extrapolation into the muon stations and the corresponding closest hit for each station that fired yields a good separation between muons and non-muons when used for a hypothesis test. It is defined as

$$D^2 = \frac{1}{N}\sum_{i=0}^{N}\left(\left(\frac{x_{\mathrm{closest},i} - x_{\mathrm{track},i}}{\mathrm{pad}_{x,i}}\right)^2 + \left(\frac{y_{\mathrm{closest},i} - y_{\mathrm{track},i}}{\mathrm{pad}_{y,i}}\right)^2\right), \tag{3.3}$$

where $\{x,y\}_{\mathrm{closest},i}$ denotes the $\{x,y\}$ coordinate of the closest hit in station i. Likewise, $\{x,y\}_{\mathrm{track},i}$ is the $\{x,y\}$ coordinate of the track in station i. The value of $\mathrm{pad}_{\{x,y\},i}$ is the $\{x,y\}$ dimension of the pad traversed by the track. Due to multiple scattering it depends on the candidate's momentum as well as the traversed material. The material in turn depends on the polar angle. Due to those dependencies, the hypothesis tests are performed in bins of detector region (which corresponds to the

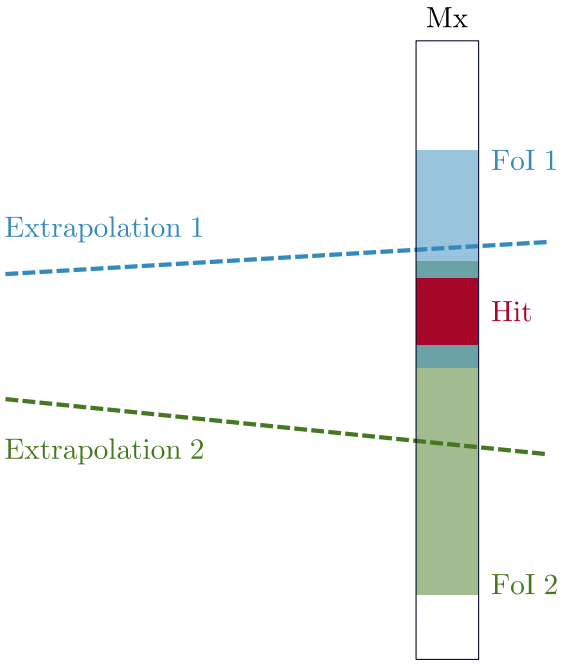


Figure 3.2: Example scenario for nShared. Two extrapolations into the muon station Mx (where $x = 2, \ldots, 5$) share the same hit because it is contained in both fields of interest around the extrapolated hit. For the track which has a larger sum of squared distances between the extrapolations and the corresponding closest hits, the value of `nShared` is incremented. Please note that this sketch is not to scale.

polar angle) and momentum. For a given bin, two tests are performed that yield $P(\mu)$, the probability of the candidate being a muon and, respectively, $P(\text{not}\,\mu)$, the probability of the candidate not being a muon. From those quantities the delta log likelihood, DLL, is calculated as

$$\text{DLL} = \log\left(\frac{P(\mu)}{P(\text{not}\,\mu)}\right) = \log\left(P(\mu)\right) - \log\left(P(\text{not}\,\mu)\right) . \tag{3.4}$$

Details on the procedure and the binning can be found in reference [14]. It should be noted here, that due to the two-dimensional binning, many calibration constants are needed. D^2 and the DLL are saved in the muon track object as `MuonDist2` and `MuonDLL`. In the offline reconstruction, the quantities $\log(P(\mu))$ and $\log(P(\text{not}\,\mu))$ are stored as `MuonLLMu` and `MuonLLBg` in the muon PID object. Additionally, a track fit is performed on the extrapolation using only the closest hits. The resulting χ^2/ndof is stored in the muon track object as `MuonChi2perDoF`.

Another quantity that is often used is called `ProbNN`. It is a particle identification variable determined by a specifically trained machine learning algorithm. It is, however, not part of the default muon reconstruction and therefore will not be discussed in detail. A description of `ProbNN` can be found in reference [4].

4 Trigger system

In Run I, with an inelastic collision rate of 13 MHz from the LHC and an average event size of 70 kB, 0.87 TB of data were produced each second within the acceptance of the LHCb experiment. For Run II the numbers are 50 kB at 20 MHz which corresponds to a data rate of 1 TB per second. At 4×10^6 s of successful running per year, this translates to 4 EB per year. Not only is this way too much data to be stored given today's computing resources, also a lot of that data is background that would later be discarded anyway. Thus the rate needs to be reduced in real time. In an ideal world, analysis selections could be run directly on incoming data using all the information available in the event and through reconstruction. However, more complex information is computationally expensive and – given the state of the computing power of the LHCb experiment at the time of this writing – can not be computed in real time. The trigger system aims to give the best possible approximation given the limited resources. It uses simplified information to minimise the CPU time consumption while maximising the physics yield. For Run I as well as for Run II, LHCb utilises a two-stage trigger system with a hardware stage and a software stage.

Hardware trigger

The first stage of LHCb's trigger system (called Level 0 or L0) is a fixed-latency hardware solution that takes 4 µs to decide whether an event is rejected or passed on to the software trigger. To make this decision, it uses information from the calorimeters and muon stations that can be accessed quickly. Hadrons, as well as electrons and photons are selected based on transverse energy clusters. In addition, information from the SPD and PS is used to separated electrons from photons. Muons are selected through a simplified track reconstruction in the muon stations. The L0 reduces the data rate to 1 MHz at which the whole detector can be read out. A detailed description of the Level 0 trigger can be found in reference [19].

Software trigger

Events that are accepted by the hardware trigger are passed on to the high level software trigger (HLT) that runs on the event filter farm (EFF). The EFF is a large computer farm dedicated to event reconstruction and software-based triggering. In Run I, the farm consisted of 29 000 logical CPU cores, while for Run II it has been upgraded to 50 880 cores. Given the advances in technology, this corresponds to a 90 % increase of the available computing power for Run II with respect to Run I. A detailed description of the EFF infrastructure can be found in reference [20].

In the first stage of the high level trigger, called HLT1, events are partially reconstructed. Using those partial reconstructions, events are either rejected or passed on to the second stage of the HLT. This reduces the event rate from 1 MHz down to 80 kHz (150 kHz in Run II).

The second stage of the software trigger, HLT2, performs a full offline-like event reconstruction and a mixture of inclusive and exclusive selections. In Run II it utilises the same reconstruction as is used offline. The rate at which events are written to storage is reduced to 5 kHz (12.5 kHz). This corresponds to a data rate of about 350 MB/s (625 MB/s) at an average event size of 70 kB (50 kB). The data flow in the whole trigger is visualised in Figure 4.1.

In Run I, the LHC operated at centre of mass energies of 7 TeV and 8 TeV with about 1300 bunches at a 50 ns spacing. At a fixed instantaneous luminosity of $\mathcal{L} = 4 \times 10^{32}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ this converts to $\mu = 1.7$ visible interactions per bunch crossing in LHCb on average. Given the size of the EFF, some trade-offs had to be made in the trigger to handle the high data rate. In Run II the LHC operates with 1100 bunches at a 25 ns spacing and a centre of mass energy of 13 TeV. Thus the average number of visible interactions per bunch crossing goes down to $\mu = 1.1$. This and the much larger EFF allow the trigger to take more time per event. This is an opportunity to improve the muon identification: For Run I with its very strict timing constraints, trade-offs were made that led to losses in the muon reconstruction efficiency in the trigger. Those inefficiencies are studied in this work in order to regain them in the context of Run II.

Deferred triggering

The LHC provides stable beams only about 30 % of the time in a non-predictable pattern. This results in the EFF being idle 70 % of the time. In order to prevent wasting a lot of computing power, the concept of deferred triggering was introduced in Run I: 25 % of the events that passed the hardware stage were buffered to the local disks of EFF nodes. During the idle time of the LHC, data was processed from the local disks resulting in effectively 25 % extra CPU power. This allowed for greater flexibility in exclusive selections as well as lower thresholds in the reconstruction. For Run II this concept was taken even further. After the first stage of the HLT, all events are written to disk. This improves CPU usage and increases the CPU time available for event filtering by a factor of 3. Additionally, it allows for alignment and calibration to be performed before the second stage of the HLT runs. Among other things, this makes particle identification based on the information from the Ring Imaging Cherenkov detectors (RICH) possible. In Run I this was only possible offline. Through the online alignment and calibration, the high level trigger can perform the same reconstruction as is done offline. Technical details of the deferred high level trigger can be found in reference [21]. The online alignment and calibration procedure is described in reference [22].

Turbo stream

In Run II the high lever trigger produces offline quality candidates that are used in the selections. If written out, physics analyses can be performed using those candidates. This is the idea of the Turbo stream. Normally, if the trigger accepts an event, the whole raw event is saved to storage and processed again by the offline reconstruction. That process takes additional time: for a 3 GB raw file, the entire offline reconstruction process takes roughly 30 hours. At the beginning of Run II, 2.5 kHz of the 12.5 kHz output rate of the HLT2 are provided by the Turbo stream and contain information on the trigger reconstruction in addition to the raw event. The resulting files can directly be used for analyses. Once the Turbo stream is fully validated, it is possible to remove the sub-detector banks from the raw event, and replace them with the specific trigger candidates. This reduces the event size from around 50 kB to the level of 5 kB. More details on the Turbo stream and its design can be found in reference [23].

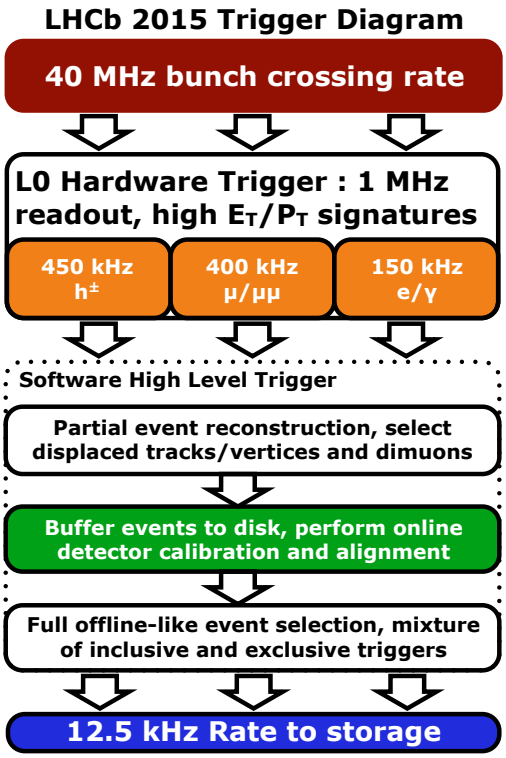


Figure 4.1: Flowchart of the LHCb trigger system in Run II. Figure taken from [6]. After the hardware stage with an output rate of 1 MHz, a first software stage selects events based on a partial reconstruction. Those events are buffered to disk at a rate of 150 kHz. After the real-time alignment and calibration are performed, the second software stage performs an offline-like reconstruction. At an average event size of 50 kB, the output rate corresponds to 625 MB/s.

5 Muon triggers

5.1 Run I

The Level 0 muon trigger utilises four dedicated processing units, one for every quadrant of the muon system. Each of those processors performs a simplified track reconstruction to identify the two muons with highest transverse momentum by searching for hits on a straight line pointing towards the interaction point in the y-z-plane. In the x-z-plane, the search is limited to $p_T > 0.5\,\mathrm{GeV/c}$. This methods yields an estimate of the transverse momentum with a resolution of about 20 % with respect to offline. The `L0Muon` line selects events where the single track with the highest p_T has $p_T > 1.76\,\mathrm{GeV/c}$. The `L0DiMuon` line selects events where the product of transverse momenta is larger than $(1.6\,\mathrm{GeV/c})^2$ for the two tracks with the highest and second highest p_T. Those thresholds correspond to the configuration in 2012. In 2011, the values were $1.48\,\mathrm{GeV/c}$ and $(1.6\,\mathrm{GeV/c})^2$. As shown in references [2, 19], the integrated efficiency of both L0 muon triggers is 89 %. Whenever efficiencies of the software trigger are given in this work, the events in question are required to have passed the Level 0 and thus the efficiencies are those of the software trigger only.

The muon reconstruction in HLT1 is limited by tight timing constraints. Therefore some simplifications with respect to the offline reconstruction are made. At the start of the HLT1 and available to all lines, VELO candidates are created by the `FastVelo` algorithm [8] which is limited to its first stage only. Additionally a full reconstruction of primary vertices is performed based on the VELO tracks. Muon lines first match the VELO candidates to hits in the muon stations in order to validate potential muon candidates. This is done by the `MatchVeloMuon` algorithm [24]. The remaining VELO candidates are run through the forward tracking algorithm [9]. In contrast to the offline reconstruction, the track matching algorithm is not used. For the forward tracking the same configuration as in the offline reconstruction is applied, only the thresholds are adjusted to decrease the CPU time consumption. On the resulting long tracks, a specialised version of the offline muon identification is run to filter out tracks that do not pass as muons.

This specialised version, contained in the `IsMuonTool`, was created for Run I by taking the offline muon reconstruction code and stripping away parts that were not used in HLT1. Logically, it performs the same steps as the offline `IsMuon` algorithm but the code is optimised for performance in the high level trigger. Afterwards, a Kalman filter based track fit with a simplified detector geometry is performed in order to filter tracks based on the χ^2 of the track fit. In the course of this work, the efficiency of this procedure is analysed. For each of the steps, the individual contribution to the total inefficiency is obtained. Using this information, the procedure is tuned for Run II. Additionally, the code of the `IsMuonTool` is revised in order to remove possible inconsistencies with the offline reconstruction.

Muon triggers in HLT1 can be divided in two categories: single muon triggers and dimuon triggers. The former are used for inclusive and electroweak selections while the latter are used as signal triggers for analyses with two muons in the final state. Both kinds require at least one of the aforementioned L0 decisions. For detailed performance studies in this work the `Hlt1TrackMuon` single muon line is used. The data flow within the line is visualised in Figure A.1a in the appendix. A detailed description can be found in reference [25]. In addition, section A.1 in the appendix also contains flow charts of the single muon lines `Hlt1SingleMuonHighPT` and `Hlt1SingleMuonNoIP` as well as the dimuon lines `Hlt1DiMuonLowMass` and `Hlt1DiMuonHighMass`. More details on the muon lines can be found in references [2, 19, 26].

In HLT2 a complementary step to the VELO tracking in HLT1 is run so that VELO candidates in HLT2 are identical to offline. All those VELO candidates are passed on to the full offline forward tracking with additional requirements on momentum ($p > 3\,\mathrm{GeV/c}$) and transverse momentum ($p_T > 0.3\,\mathrm{GeV/c}$). Those long tracks are used in combination with the full offline muon identification, `MuonIDAlg`, with the only limitation that no additional fit is performed on the extrapolation and thus no χ^2 is obtained. As the signal rates are still very high in HLT2, single muon triggers alone are not useful as the main signal triggers. Dimuon and even trimuon lines are used. They are described in reference [26].

5.2 Run II

At the time of this writing, the L0 thresholds are still being finalised. In the nominal scenario, the transverse momentum thresholds are expected to be $p_T > 2.8\,\mathrm{GeV/c}$ for single muons and $p_T^{\mu_1} \times p_T^{\mu_2} > (1.8\,\mathrm{GeV/c})^2$ for dimuons [27]. The tighter cuts

are necessary since at a higher centre of mass energy more b quarks are produced while the output rate of the Level 0 trigger is fixed at 1 MHz.

The most important change in the architecture of HLT1 that affects the muon identification is that long tracks with $p_T > 0.5\,\mathrm{GeV/c}$ are provided to all trigger lines. Those tracks can be used directly for muon identification and no matching of VELO tracks is necessary. In order to not lose softer muons, all VELO tracks that have not been upgraded by the forward tracking are run through an adjusted version of the VELO muon matching. The `ComplementForward` algorithm combines those tracks with the ones found in the regular forward tracking. In addition, the VELO tracks are now built utilising the full potential of the `FastVelo` algorithm using both the first and the second iteration. The muon identification used in the HLT2 is now exactly the same as offline where the newly introduced `MuonIDAlgLite` is used. Said algorithm is developed in the course of this work as part of an effort to share the same code base between the offline reconstruction and the trigger in order to remove unwanted differences. This goes hand in hand with changes to the aforementioned `IsMuonTool`. For that purpose a `CommonMuonTool` is developed that offers functionality which is used both in the `IsMuonTool` in HLT1 and in the `MuonIDAlgLite`. The following chapter is concerned with the development of those tools.

6 Unification of online and offline code

During Run I of the LHC two separate code bases for the offline (`Brunel`) and the online (`Moore`) reconstruction of muons existed. Although the offline reconstruction performs additional computations, a lot of code was effectively duplicate. This reduces maintainability and can be a possible source for discrepancies between online and offline. Ideally, `Brunel` and `Moore` would run the same code base. Additionally, the offline code was not previously optimised for CPU time consumption. This work is part of a collaborative effort to unify the code and improve its performance.

As described in chapter 3, the full muon reconstruction consists of five parts: `IsMuon`, `IsMuonLoose`, `IsMuonTight`, `nShared`, and `DLL`[1]. Only the first has traditionally been of relevance in HLT1. In order to calculate those quantities, different experimental constants as well as information concerning the event need to be known. Extracting and streamlining the core functionality from the existing code base was one of the central tasks of this work. The rationale behind the new tool is that it has to run in both stages of the software trigger as well as the offline. Therefore it has to be as fast as possible without doing unnecessary work. Only the functionality to calculate `IsMuon` and its variants is included in the newly created `CommonMuonTool`. Separate tools are introduced to calculate `DLL` and `nShared` and to create muon track objects.

6.1 CommonMuonTool

Based on previous work [24], a `CommonMuonTool` was created that can be used in Brunel as well as in Moore. Figure 6.1 visualises the intended use of the tool and additional tools for offline-only properties. This removes duplication and thus greatly improves maintainability. Additionally, an effort was made to split the tool into small functions with clearly defined responsibilities. In the past, improvements

[1] Technically also `ProbNN` which is calculated by a separate tool.

in the online muon identification code rarely made it back into the offline code. With a unified code-base both sides benefit from improvements in the same way.

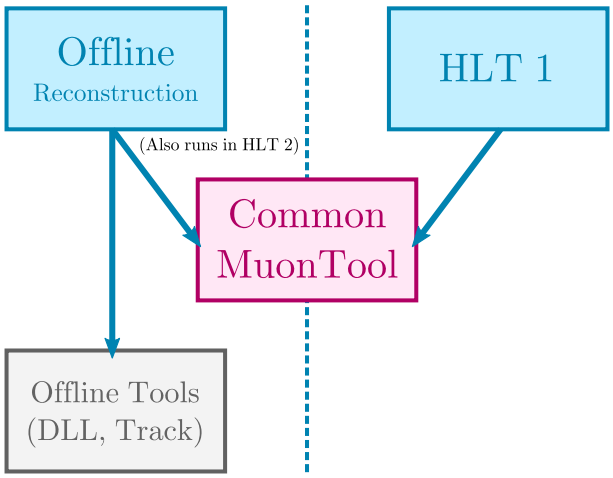


Figure 6.1: Use of the `CommonMuonTool` from online and offline code.

The tool itself does not save any state related to an event or track. The only external information it uses are constants from the conditions database. Those are a scale factor for the FoI, the pre-selection momentum value, the two edges of the momentum bins (see Table 3.1), the z-coordinates of the muon stations, the dimensions of the muon stations, and three FoI parameters for both x and y-dimension. Additionally, it makes use of a `CommonMuonHitManager`[2] and a `DeMuonDetector` in order to extract read-out information. Their task is to extract hit information from the raw detector data.

The `CommonMuonTool` offers a dedicated method for each logical step in the `IsMuon` algorithm as depicted in Figure 3.1. In addition, functions offering functionality to calculate `IsMuonLoose` and `IsMuonTight` are implemented. What follows is an overview of public methods of the tool which can be used both in the offline reconstruction from the `MuonIDAlgLite` as well as in the HLT1 from the `IsMuonTool`. All methods take their (non-primitive) input by constant reference and return by value to make use of return value optimisation.

- The `initialize` method sets up the tool. It loads additional tools and fetches the constants from the database. It takes no parameters and returns a status code.

[2]The `CommonMuonHitManager` is what was previously called `Hlt1MuonHitManager`. All other associated tools have been changed using this naming scheme.

- `preSelection` takes a track object and returns whether the track passes the pre-selection criteria. In this case it just checks if the track's momentum is larger than the cut value ($p > 3\,\mathrm{GeV/c}$).

- `extrapolateTrack` takes a track object and extrapolates it into the muon stations. It returns a container (aliased as `MuonTrackExtrapolation`) containing a point (x, y) for each station (at a fixed z) except M1 which is not used.

- `inAcceptance` uses the output of `extrapolateTrack` in order to check whether the coordinates of the extrapolated hits are be within the acceptance of the muon stations.

- `hitsAndOccupancies` takes both a track and a `MuonTrackExtrapolation` container as input and returns two containers via a `std::tuple`. The first holds all the hits that are found in the muon stations within the FoI around the extrapolations. The second container holds the number of hits in each station which is called the occupancy of the station. The latter is used to check whether a station has hits within a FoI. The FoI are calculated from the track momentum via the `foi` method.

- `extractCrossed` takes a container of hits in the muon stations as input and extracts the crossed hits. Additionally it also calculates the occupancies considering only the crossed hits. Both containers are returned in a `std::tuple`.

- `isMuon` uses occupancies and the track's momentum to classify it according to Table 3.1. If it obtains a container of occupancies from crossed hits, it calculates `IsMuonTight` by definition.

- `isMuonLoose` also takes a container of occupancies and a track momentum and calculates `IsMuonLoose` according to Table 3.1.

- `foi` is a helper method that, for a given station, region, and momentum returns a point (`std::pair`) that marks the edge of the field of interest.

In HLT1, muon lines make use of the `IsMuonTool`. It has been adjusted to use the functionality offered by the `CommonMuonTool`. Like every tool that is used from a trigger line, it exposes a method `tracksFromTrack` that takes the current trigger track as input and only writes to an output container if the `IsMuon` criterion

is met. It uses the functions `preSelection`, `extrapolateTrack`, `inAcceptance`, `hitsAndOccupancies`, and `isMuon` in sequence and thus perfectly resembles the flow depicted in Figure 3.1. Since the use of the `CommonMuonTool` makes it so concise, the `tracksFromTrack` method of the `IsMuonTool` is shown in its entirety in section A.3 in the appendix.

An algorithm in the offline code receives a whole event as input. Other than in the trigger where one track is dealt with at a time, the event contains a collection of tracks. The calculation of `IsMuon` is embedded inside a loop over all the tracks in the event. For the purpose of backwards compatibility, the old `MuonIDAlg` has not been removed. Instead, a new algorithm called `MuonIDAlgLite` was added and is now used instead. A difficulty arises from the fact that the calculation of the `nShared` relies on relationships between the tracks in an event: Track-specific information needs to be kept for each track in the event during the iteration. This is all done by a dedicated tool. A specific effort was made to optimise the procedure in terms of performance. One example is how hits and occupancies for crossed hits are obtained: Before, the whole read out procedure was done again with the additional requirement that hits need be crossed. This information however can be obtained from a hit which allows to simply filter the container of already extracted hits, as done by the `extractCrossed` method of the `CommonMuonTool`. Furthermore, the algorithm now accepts a collection of input locations. Therefore both long and downstream tracks can be used in combination.

6.2 Offline-only tools

Two additional tools are introduced [28] in order to provide additional information that is not used in HLT1. Those are called `DLLMuonTool` and `MakeMuonTool`. The `DLLMuonTool` is responsible for calculating delta log likelihood (DLL) of the muon hypothesis. It loads all the parameters for the hypothesis tests in different bins in momentum and region as described in section 3.3 in order to calculate $P(\mu)$ and $P(\text{not}\,\mu)$. Two different implementations can be used via a flag: `calcMuonLL_tanhist` and `calcMuonLL_tanhist_landau`. Both return a `std::tuple` that contains the likelihood for the muon hypothesis $P(\mu)$ as well as for the background hypothesis $P(\text{not}\,\mu)$. It also it contains the squared distance of the muon track D^2. The DLL is then calculated according to Equation 3.4. Additionally, the tool allows to calculates the `nShared` variable via the `calcNShared` method. In order to relate the tracks to each other, the tool saves information on each track in an event in a temporary map. The `MakeMuonTool` is intended

to create a muon track once all the necessary information is there. It exposes a function called `makeMuonTrack` that returns a pointer to an `LHCb::Track` object which contains the information from the extrapolation. If a corresponding flag is set, the tool also performs a track fit in order to obtain the χ^2 of the track.

6.3 Impact

Once these changes were put into place it has been verified that they produce identical results. This ensures that no bugs that would alter the physics output made it into the production code. The CPU time consumption in the offline reconstruction has been found to have reduced by a factor of 2.5 from once around 2.5 ms to now only 1 ms per event. One reason for that is that less redundant calculations are performed. For example, by using the `extractCrossed` method, the previously obtained container of hits can be reused.

Most importantly, the same exact code base now powers both the offline muon identification as well as the one in the software trigger. Every single muon that makes its way through the LHCb reconstruction software, online as well as offline, passes through the `CommonMuonTool` which is a central result of this work. This prevents technical subtleties from having an impact on the performance. During the course of this work it became apparent that the `IsMuonTool` in Run I did not use the correct pad sizes for uncrossed hits. The tool used an internal definition of the pad sizes for each station and region instead of the information from the hit object itself. As shown in Equation 3.2, the pad dimensions are considered when checking whether a hit is inside a FoI. With the faulty pad sizes, uncrossed hits were wrongly rejected resulting in smaller efficiencies.

7 Efficiency evaluation

For a sample of n events, the number of observed successes k – events that pass a certain criterion at hand –, follows a binomial distribution $k \sim \text{Bin}(n, \varepsilon)$ where ε denotes the efficiency. The efficiency can be calculated as

$$\varepsilon = \frac{k}{n} \tag{7.1}$$

Since efficiencies in this work tend to be close to 1, the uncertainties can not be approximated by symmetric normal distributions. Instead, exact Clopper-Pearson intervals [29] are used for the asymmetric uncertainties. Other than commonly used methods, those do not rely on any approximation of the binomial distribution. A Clopper-Pearson interval always has at least nominal coverage for any population proportion but usually tends to be rather conservative. It can be defined as

$$\begin{aligned} S_{\leq} &= \left\{ \varepsilon : P[\text{Bin}(n;\varepsilon) \leq k] > \frac{\alpha}{2} \right\} , \\ S_{\geq} &= \left\{ \varepsilon : P[\text{Bin}(n;\varepsilon) \geq k] > \frac{\alpha}{2} \right\} . \end{aligned} \tag{7.2}$$

Using the relation between the cumulative binomial distribution and the beta distribution $B(x;a,b) \sim x^{a-1}(1-x)^{b-1}$, the interval can be expressed as

$$B\left(\frac{\alpha}{2}; k, n-k+1\right) < \varepsilon < B\left(1-\frac{\alpha}{2}; k+1, n-k\right) . \tag{7.3}$$

Throughout this work $\alpha = 1\sigma$ is used.

Two different kinds of studies are performed. In order to get a very fine-grained insight into the structure of the efficiencies in Run I, Moore, the trigger software of the LHCb experiment, is run interactively on a sample of simulated $B^0 \to K^{*0}\mu^+\mu^-$ signal events. $B^0 \to K^{*0}\mu^+\mu^-$ with $K^* \to K^+\pi^-$ is a flavour-changing neutral current process that proceeds via loop and box amplitudes in the standard model. New particles can modify the angular distributions of the decay products. A recent analysis observed a local discrepancy of 3.7σ for one of the form-factor independent observables [1]. This could be a hint for physics beyond the standard model. With

more data this discrepancy could be resolved or be turned into a discovery. This makes it a prime channel to understand and improve the muon reconstruction efficiencies. For a second, comparative study, different versions of Moore are run independently on a sample of simulated $B^+ \to J/\psi K^+$ decays with $J/\psi \to \mu^+\mu^-$. This channel is used as a normalisation channel for the $B_s^0 \to \mu^+\mu^-$ decay and therefore equally suited to determine muon reconstruction efficiencies. Principally, all studies could have been done on the same sample but due to technical reasons this was not possible. Both studies use only events that have already been processed by the whole offline reconstruction chain and thus all measured inefficiencies are only due to effects in the trigger. Simply from counting candidates that caused the trigger line to give a positive decision the efficiency is obtained from Equation 7.1 with $k = N_{\mathrm{TOS}}$. In order to factor out the L0 efficiencies, all candidates (nominator and denominator) are required to have passed either `L0Muon` or `L0DiMuon`.

7.1 Analysis of 2012 inefficiencies

In order to evaluate the performance of the 2012 `IsMuon` algorithm in the trigger, Moore is run interactively. That means a custom bit of Python code inspects each event and the corresponding trigger track and writes information on the trigger decision to a text file that is evaluated afterwards.

For this study, a sample of simulated $B^0 \to K^{*0}\mu^+\mu^-$ signal events is used. Signal candidates are selected by an inclusive selection through the `B2XMuMu` stripping line. The stripping line is described in reference [30]. The second part of the selection is implemented directly in the Python script that determines the efficiencies. For each event that passes `L0Muon` or `L0DiMuon`, it fetches the trigger tracks associated with the `Hlt1TrackMuon` line as well as B^0 candidates that were selected by the stripping line. Of those candidates, only the one which has a particle reconstructed as a K^{*0} among its daughters is considered. A B^0 is required to have $p_T > 0.5\,\mathrm{GeV/c}$. In Figure 7.1, the distributions of mass, momentum, and transverse momentum in the sample after this selection are shown. It contains 19 249 candidates. The daughter muons are then matched against the trigger tracks by requiring at least 70 % overlapping LHCbIDs. This matching is equivalent to the functionality of the `Hlt1TisTosTool`. This has been verified on the whole sample where a 1:1 correspondence is found. The cut at 70 % is motivated in Figure 7.2.

The efficiency is calculated for several scenarios: Successively, more and more parts of the `Hlt1TrackMuon` line are left out so that the individual contributions

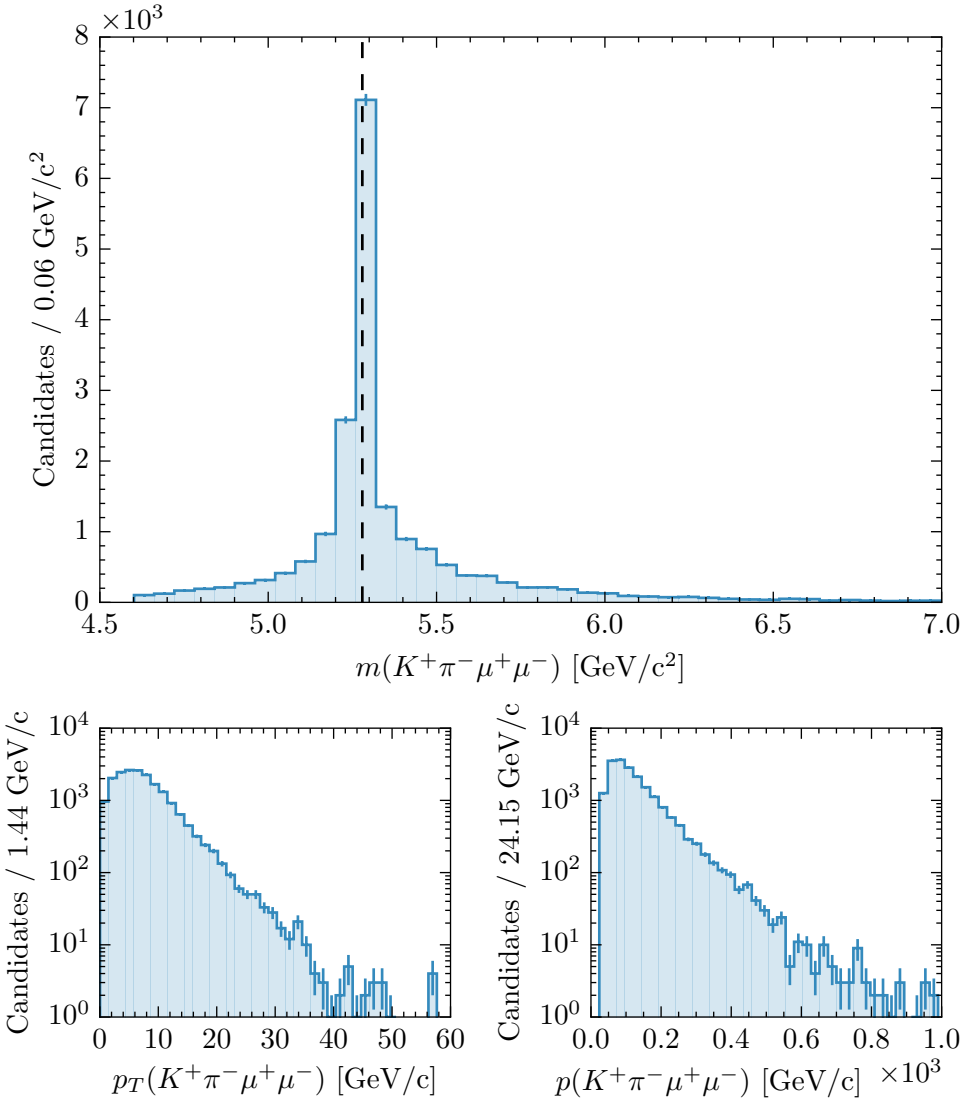


Figure 7.1: Mass (top), transverse momentum (bottom left), and momentum (bottom right) of the reconstructed $B^0 \to K^{*0}\mu^+\mu^-$ candidates in the simulated sample used to evaluate the efficiency of `Hlt1TrackMuon` in Run I. Candidates are selected by the inclusive `B2XMuMu` stripping line and required to have $p_T > 0.5\,\text{GeV/c}$ and a reconstructed daughter K^{*0}. The nominal B^0 mass from reference [31] is marked by the dashed line at $(5279.58 \pm 0.17)\,\text{MeV/c}^2$. The tails in the mass distribution are a result of the inclusive selection. Since simulated signal candidates are used, this does not influence the validity of the study.

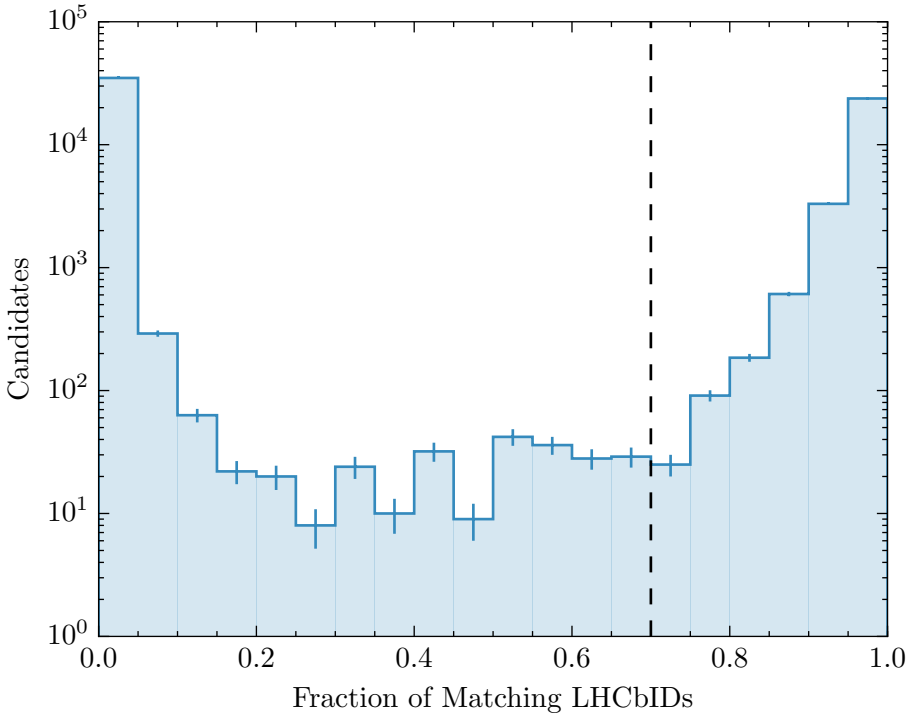


Figure 7.2: Percentage of matching LHCbIDs between trigger tracks and signal muons. On the left are tracks where no matching has been found while on the right a matching becomes more and more apparent. The threshold of 70 % is marked by the dashed line. Please note the logarithmic scale. Effectively, only the edges of the shown range contain a significant amount of events. The drop from the first to the second bin is much stronger than between the last bins.

to the total inefficiency of the trigger line can be computed. The results of this study can be found in Table 7.1 and Figure 7.3. Figure 7.4 contains a visual representation of the individual contributions to the total inefficiency. It can be seen, that cuts are responsible for a large portion of the inefficiency. This is a deliberate choice motivated by timing constraints that can easily be adjusted. More interestingly, $(9.4 \pm 0.4)\,\%$ originate from the simplified reconstruction software. Eliminating them is a central task of this thesis and the associated work. In order to properly evaluate the effect on CPU time consumption, 13 TeV data is necessary which was not available for this study. However, the observed gains in efficiency were motivation enough to adjust the trigger strategy for Run II as described in section 5.2. Checks on the total time consumption of the HLT are routinely performed and the increases found are considered small compared to the gains in signal efficiency [32].

Table 7.1: TOS Efficiencies for (parts of) `Hlt1TrackMuon` on single muons. Successively, parts of the trigger line are removed in order to be able to calculate individual contributions to the inefficiency. The total inefficiency is $(26.33 \pm 0.27)\,\%$, of which $(9.4 \pm 0.4)\,\%$ originate from the simplified reconstruction. The individual contributions are obtained as the pairwise differences $\Delta\varepsilon$. The last two points were added in order to understand the remaining inefficiency after `IsMuon` was removed. The negative contribution from using the offline cuts in the track fit is within the statistical uncertainty. The remaining $(0.79 \pm 0.07)\,\%$ are unexplained.

Scenario (shorthand)	ε(TOS) [%]	$\Delta\varepsilon$ [%]
Original 2012 configuration. (2012)	73.67 ± 0.27	
Use both iterations in `FastVelo`. (FullVelo)	76.09 ± 0.27	2.4 ± 0.4
Remove all cuts in the streamer. (- Cuts)	93.03 ± 0.16	16.94 ± 0.31
Exclude the `MatchVeloMuon` algorithm. (- MatchVeloMuon)	96.85 ± 0.11	3.82 ± 0.20
Swap `IsMuon` and the track fit. (Swapped)	97.47 ± 0.10	0.63 ± 0.15
Exclude the `IsMuon` algorithm. (- IsMuon)	97.92 ± 0.09	0.45 ± 0.14
Use the same cuts as offline in the forward tracking. (+ Offline Cuts in Fwd)	97.87 ± 0.09	-0.06 ± 0.13
Require tracks reconstructed by forward tracking. (+ Fwd tracks only)	99.21 ± 0.07	1.34 ± 0.20

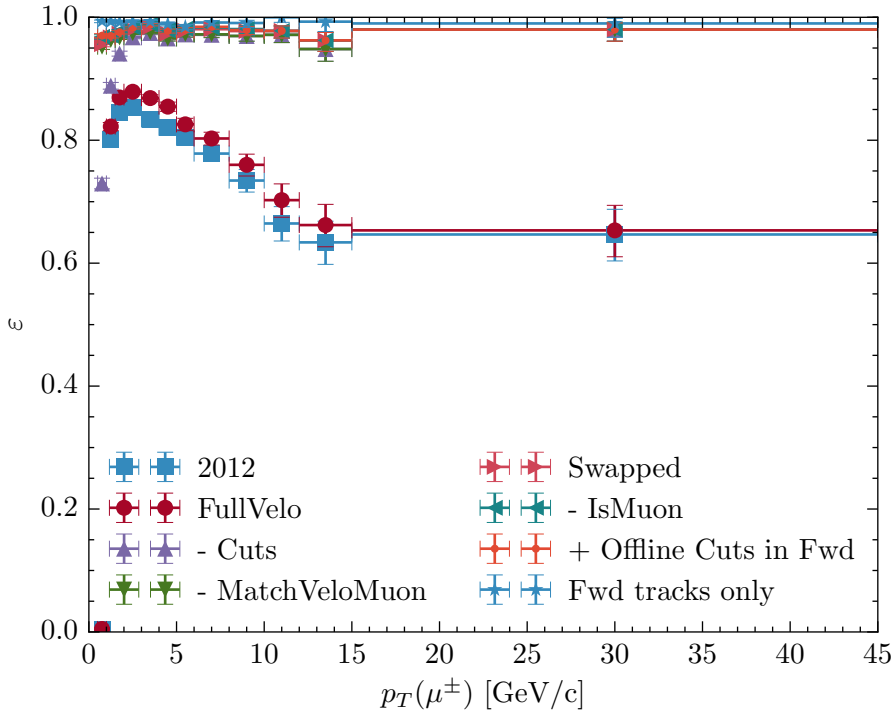


Figure 7.3: Efficiencies in bins of transverse momentum for `Hlt1TrackMuon` in the several stages of the study. It can be seen that both the cuts and the `MatchVeloMuon` algorithm introduce a p_T dependence.

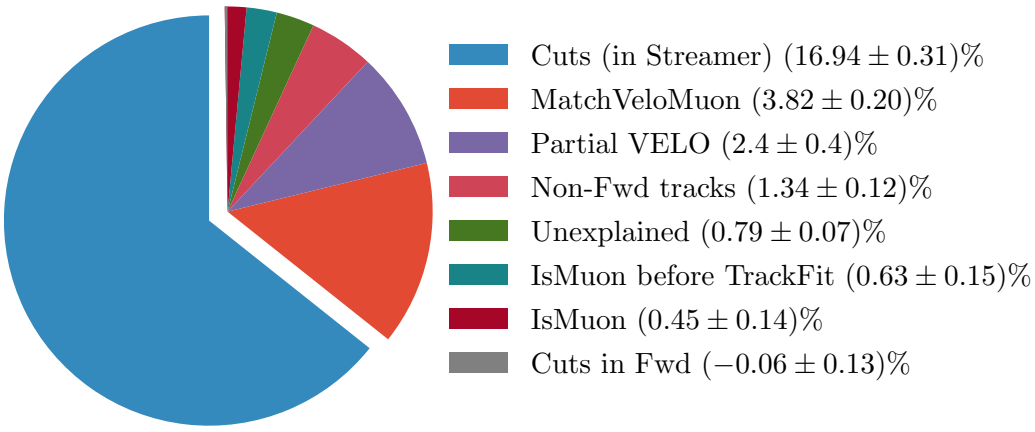


Figure 7.4: Individual contributions to the total inefficiency. It can be seen that about 17 % of inefficiency are due to the cuts in the streamer. The remaining 9 % stem from timing-driven choices in the software.

7.2 Comparative study for 2012 and 2015

In this scenario, a sample of simulated $B^+ \to J/\psi K^+$ with $J/\psi \to \mu^+\mu^-$ signal events is used. Signal candidates are selected by the `Bs2MuMuLines` stripping lines. In addition, a cut based selection is used. The corresponding cut values are listed in Table A.4 in the appendix. Figure 7.5 shows the distributions of mass, momentum, and transverse momentum in the sample. Three different versions of Moore are used to understand the changes in efficiency between Run I and Run II: The trigger software and cut values from Run I in 2012 (called 2012), the trigger software and cut values for Run II in 2015 at the time of this writing (called 2015), and the trigger software for Run II with cut values as in 2012 (called 2015'). Therefore the resulting differences in efficiencies between 2012 and 2015' stem only from changes in the software, not in the cuts. All events are required to have passed `L0Muon` or `L0DiMuon`. The TOS efficiencies are calculated in bins of transverse momentum of the single muons for `Hlt1TrackMuon` and `Hlt1SingleMuonHighPT` and in bins of p_T of the reconstructed J/ψ for the `Hlt1DiMuonLowMass` and `Hlt1DiMuonHighMass` lines. Table 7.2 contains an overview of the efficiencies obtained for the HLT1 muon trigger lines, three regarding single muons and two regarding muon pairs. They are visualised in Figure 7.6 in bins of transverse momentum. For single muon tracks, the increase in efficiency due to changes in the reconstruction software

is found to be $(8.43 \pm 0.30)\,\%$. This means that $(89.7 \pm 0.5)\,\%$ of the previously mentioned reconstruction-based inefficiencies have been removed. The efficiency for muon pairs is increased by about 15 %. This means large improvements for analyses on decay channels like the aforementioned $B^0 \to K^{*0}\mu^+\mu^-$ or $B_s^0 \to \mu^+\mu^-$. This effect is slightly mitigated by the cut values used in 2015 to reduce the CPU time consumption. The effects are of sub-percent order except for `Hlt1DiMuonHighMass`.

Table 7.2: Efficiencies of HLT1 muon lines for 2012 code, 2015 code, and 2015 code with 2012 cuts (labelled 2015'). The important difference which amounts only to changes in the software is $\Delta\varepsilon = \varepsilon(2015') - \varepsilon(2012)$. The `Hlt1SingleMuonNoIP` line is prescaled with a factor of 0.01 for 2012 and 0.1 for 2015. For easier comparison, the numbers presented in this table have been divided by those factors.

Trigger line	ε(TOS) [%]			$\Delta\varepsilon$ [%]
	2012	2015	2015'	
DiMuonLowMass	69.86 ± 0.35	85.75 ± 0.26	85.88 ± 0.26	16.0 ± 0.4
DiMuonHighMass	70.68 ± 0.34	81.28 ± 0.29	85.31 ± 0.27	14.6 ± 0.4
TrackMuon	75.25 ± 0.23	84.96 ± 0.19	83.68 ± 0.20	8.43 ± 0.30
SingleMuonHighPT	86.82 ± 0.51	95.07 ± 0.33	95.15 ± 0.33	8.3 ± 0.6
SingleMuonNoIP	74 ± 15	94.3 ± 4.4	94.4 ± 4.4	20.4 ± 15.6

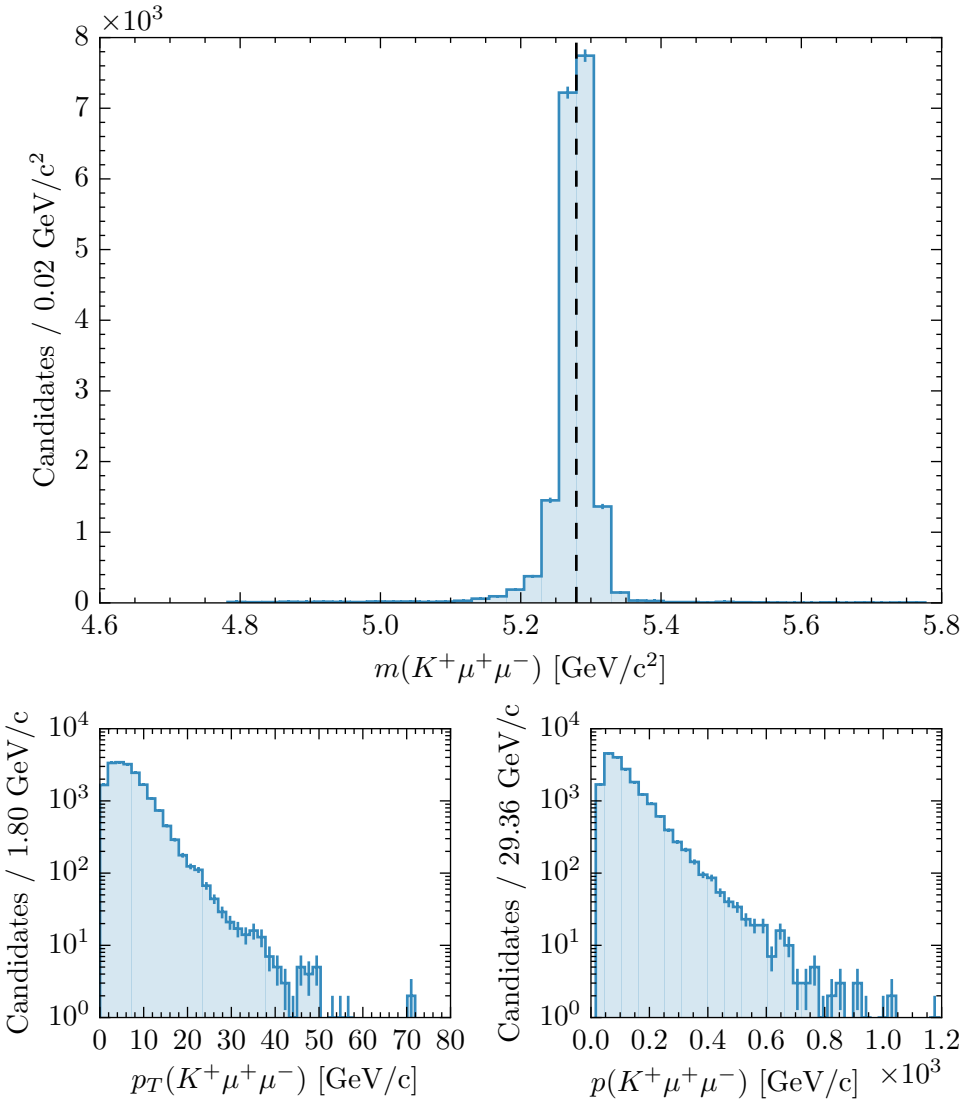


Figure 7.5: Mass (top), transverse momentum (bottom left), and momentum (bottom right) of the reconstructed $B^+ \to J/\psi K^+$ candidates in the simulated sample used to compare the efficiency of the HLT1 muon identification in Run I and Run II. The candidates are selected by the the `Bs2MuMuLines` stripping line and a cut based selection as shown in Table A.4. The nominal B^+ mass from reference [31] is marked by the dashed line at $(5279.26 \pm 0.17)\,\mathrm{MeV/c^2}$.

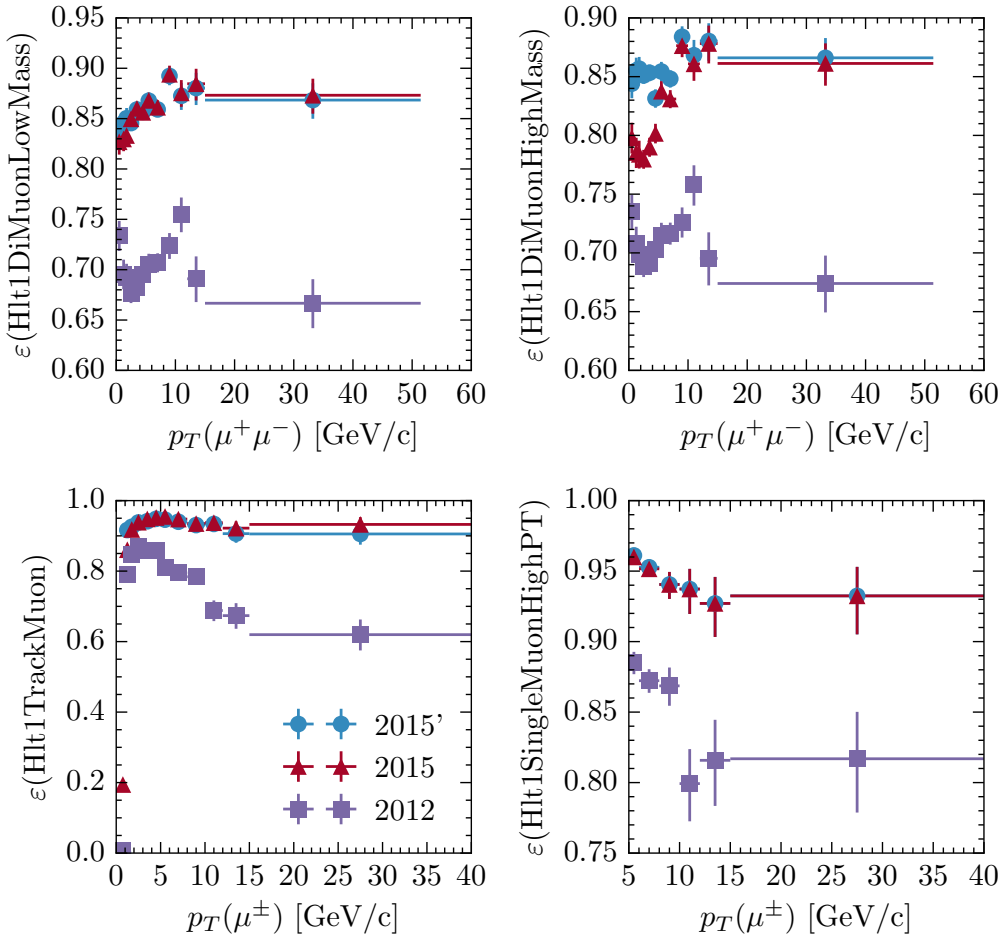


Figure 7.6: HLT1 TOS efficiencies evaluated on simulated $B^+ \to J/\psi K^+$ events. The efficiencies for 2012 are shown as purple squares, the ones for 2015 as red triangles, and those for the 2015 trigger with 2012 cuts (labelled as 2015') are shown as blue circles. It can be seen that the changes from 2012 to 2015 lead to significant gains in all discussed trigger lines. In addition, the p_T dependence improves for 2015. For the dimuon lines the curves are smoother and don't feature a sharp drop after 10 GeV/c anymore. The efficiency of the single track muon line is now almost constant for $p_T > 1$ GeV/c.

8 Conclusion

In this work, a detailed analysis of the LHCb experiment's muon identification procedure and its inefficiencies was performed. For single muon tracks in the first stage of the software trigger it was found that in Run I $(26.33 \pm 0.27)\,\%$ were not selected. $(9.4 \pm 0.4)\,\%$ are reconstruction-based and do not stem from cuts. The origin of most of those reconstruction-based inefficiencies has been precisely determined. As a consequence, the majority of the inefficiencies could be removed. This translates into an absolute increase in efficiency of about $15\,\%$ for some of the most important signal modes at the LHCb experiment, e.g. $B^0 \to K^{*0}\mu^+\mu^-$ and $B_s^0 \to \mu^+\mu^-$.

The code for the muon identification procedure has been revised for Run II. The software infrastructure has been unified such that now a common tool is used both in the offline reconstruction and in the both parts of the trigger. This improves maintainability and removes possible and unwanted differences between the online and offline muon identification. An unwanted difference in Run I between the muon identification in the first stage of the software trigger and in the offline reconstruction due to a technical subtlety has been uncovered and removed.

A Technical details

A.1 Muon trigger flowcharts

The flowcharts for Run I are created based on the TCK `0x9b0044` which in turn represents the settings for LHCb's physics programme in September 2012. Those for Run II are created from the TCK `0xfb0051` which is based on the configuration for physics on 25 ns runs in August 2015. The thresholds for Run II are not finalised at the time of this writing and might be subject to change. Table A.1 gives an overview of all flowcharts in this document. A tool for the automatic generation of flowcharts from the definition of trigger lines is a byproduct of this work. It can be found on `https://github.com/kdungs/lhcb-hltflow`. In order to obtain the configurations of the HLT1 muon lines from a TCK, the following Python script can be used with `Moore`. In the flowcharts, operations are highlighted in blue. Cuts are shown in red. The cuts are implemented in terms of so called `LoKi` functors, an overview of which can be found in Table A.2. Input and output containers are shown in green.

```
#!/usr/bin/env python

import json
import TCKUtils.utils as ut

TCK = 0x00fb0051
FILENAME = 'hlt1streamers2015.json'
STREAMERS = [
    'Hlt1TrackMuonUnit',
    'Hlt1DiMuonHighMassStreamer',
    'Hlt1DiMuonLowMassStreamer',
    'Hlt1SingleMuonHighPTStreamer',
    'Hlt1SingleMuonNoIPStreamer'
]
```

```
tree = ut.getConfigTree(TCK)
codes = [{'name': streamer.replace('Streamer', ''),
          'code': tree.leafs()[streamer].props['Code']}
         for streamer in STREAMERS]
with open(FILENAME, 'w') as jsonfile:
    json.dump(codes, jsonfile)
```

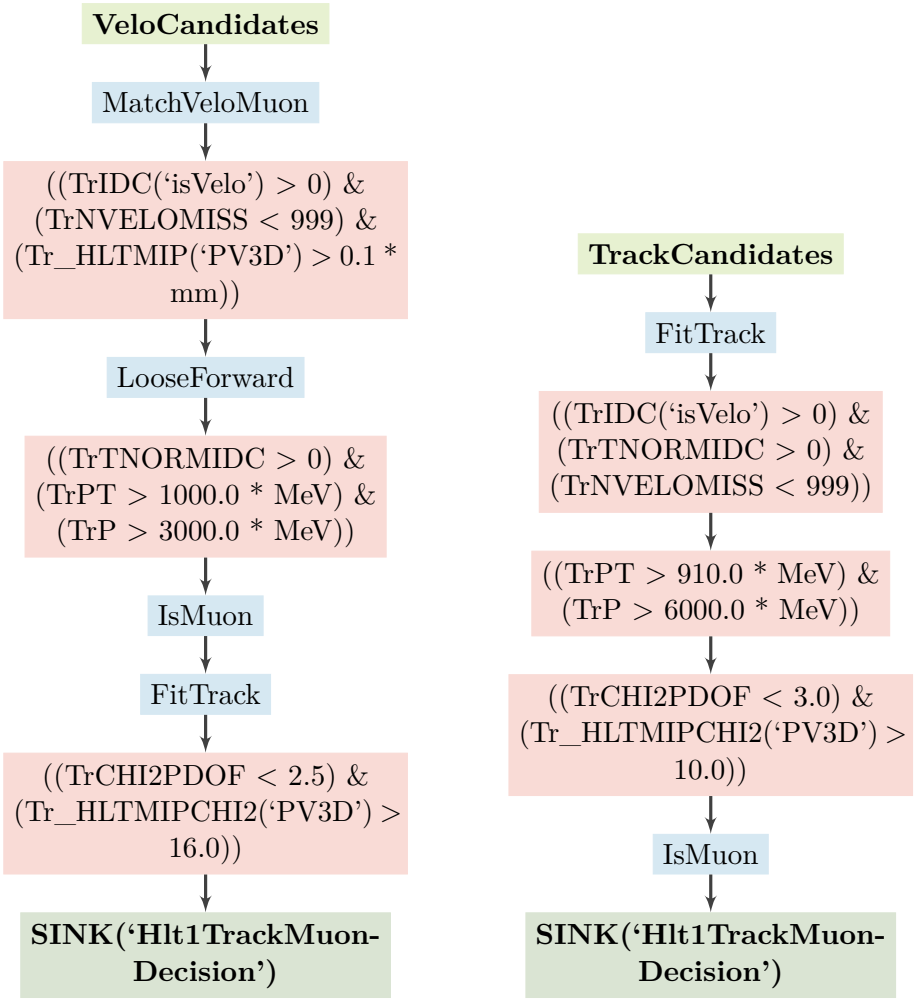
Cuts in the flowcharts are expressed using so called `LoKi` functors. Within LHCb's software they are a widely used tool to extract information about particles. An overview of the functors used in the flowcharts can be found in Table A.2.

Table A.1: Overview of all figures containing flowcharts for HLT1 muon lines.

HLT1 line	Run I	Run II
`Hlt1TrackMuon`	Figure A.1a	Figure A.1b
`Hlt1DiMuonHighMass`	Figure A.2a	Figure A.4a
`Hlt1DiMuonLowMass`	Figure A.2b	Figure A.4b
`Hlt1SingleMuonHighPT`	Figure A.3a	Figure A.5a
`Hlt1SingleMuonNoIP`	Figure A.3b	Figure A.5b

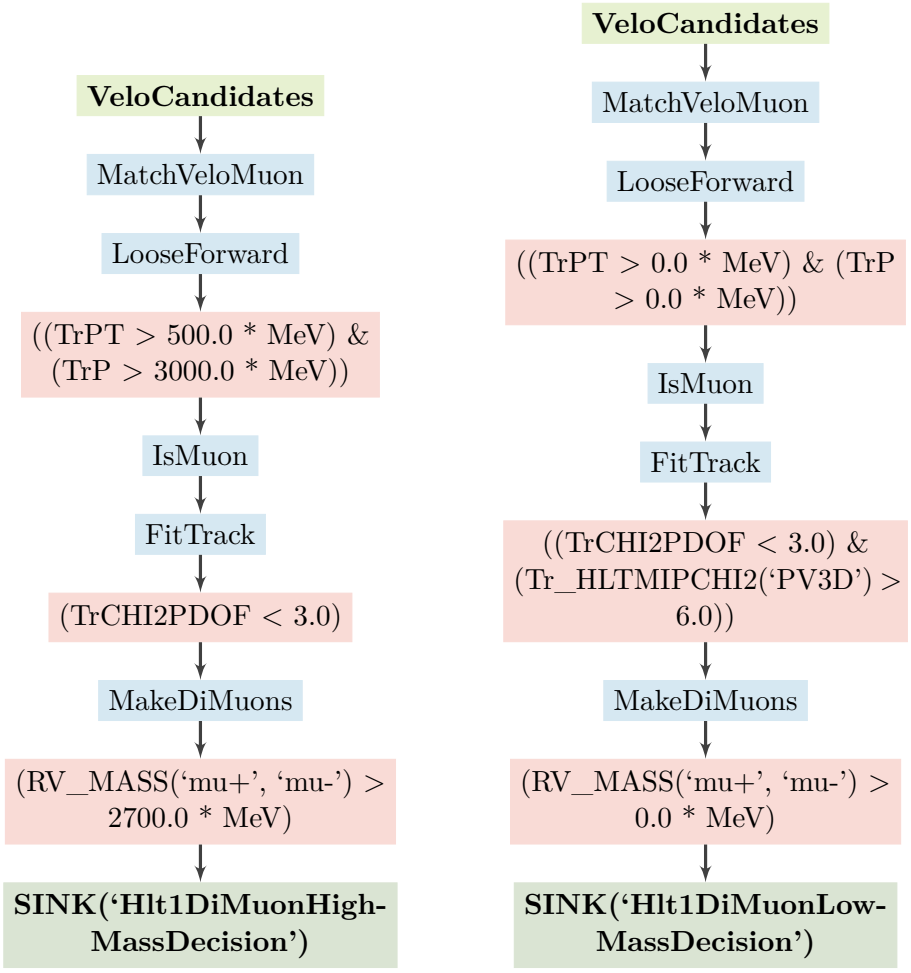
Table A.2: Overview of the `LoKi` functors that can be found in the flowcharts together with the corresponding quantities they extract.

Functor	Quantity
`TrP`	A track's momentum p.
`TrPT`	A track's transverse momentum p_T.
`TrCHI2PDOF`	The χ^2 per degrees of freedom from a track fit.
`TrNVELOMISS`	Number of VELO layers a reconstructed track misses hits in.
`TrIDC`	Counts `LHCbID`s for a given channel. Those loosely correspond with hits in a subdetector.
`TrTNORMIDC`	Effective number of hits in the T-stations. Defined as $2 \times \#\text{IT} + \#\text{OT}$
`Tr_HLTMIPCHI2`	Minimal value of χ^2 for the impact parameter of a particle in the HLT.
`RV_MASS`	Mass of a reconstructed vertex.
`Q_Prod`	Product of the charges of two combined particles.



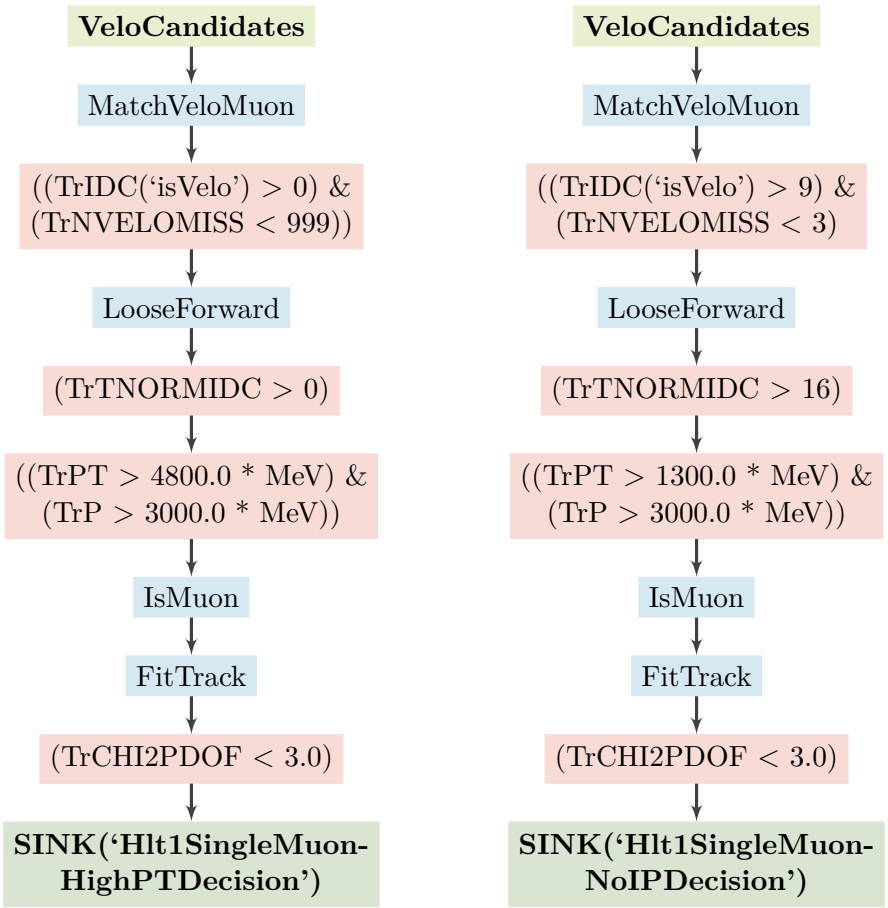
(a) `Hlt1TrackMuon` in Run I. **(b)** `Hlt1TrackMuon` in Run II.

Figure A.1: Flowcharts of `Hlt1TrackMuon` for Run I (left) and Run II (right) side by side. It can be seen that in Run II, other than in Run I, no matching of VELO tracks to the muon stations is done. Additionally, in Run II, the track fit is performed before `IsMuon` is run.



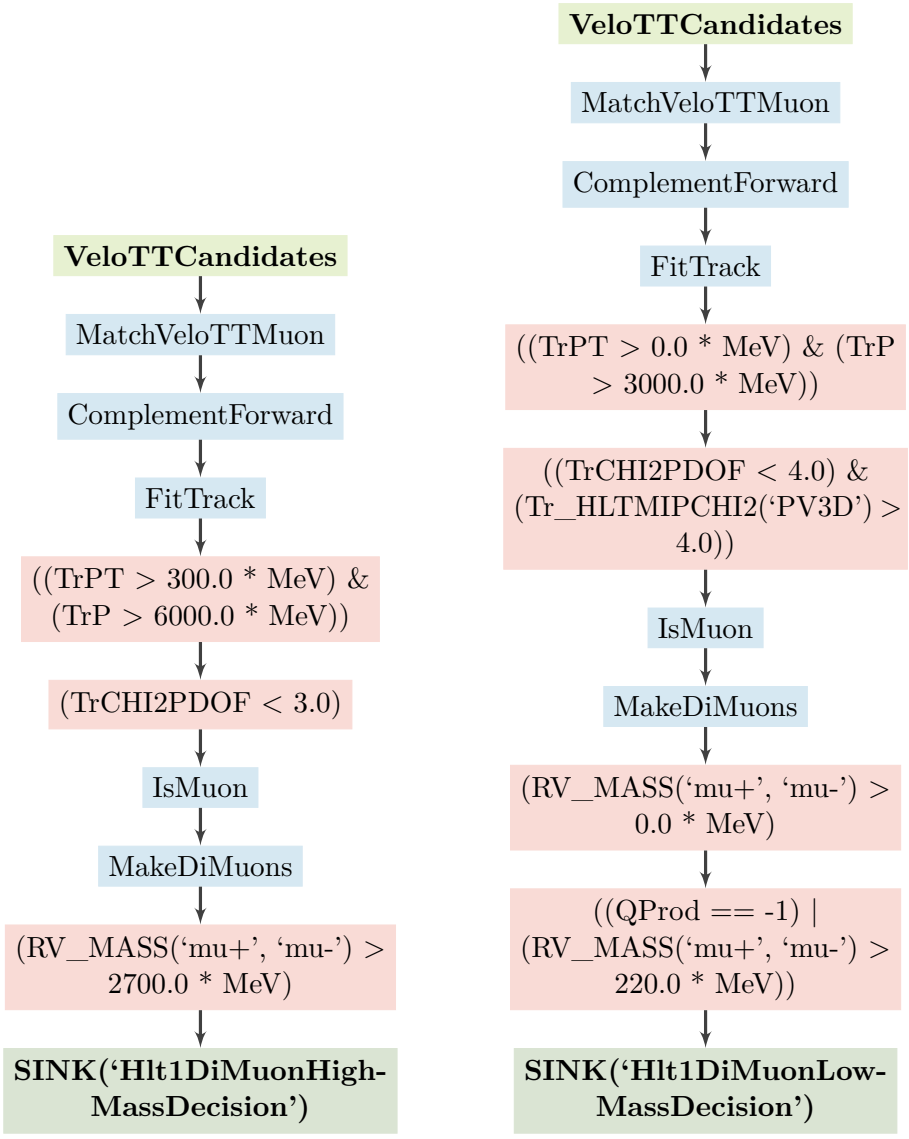
(a) `Hlt1DiMuonHighMass` **(b)** `Hlt1DiMuonLowMass`

Figure A.2: Flowcharts of HLT1 dimuon lines in Run I.



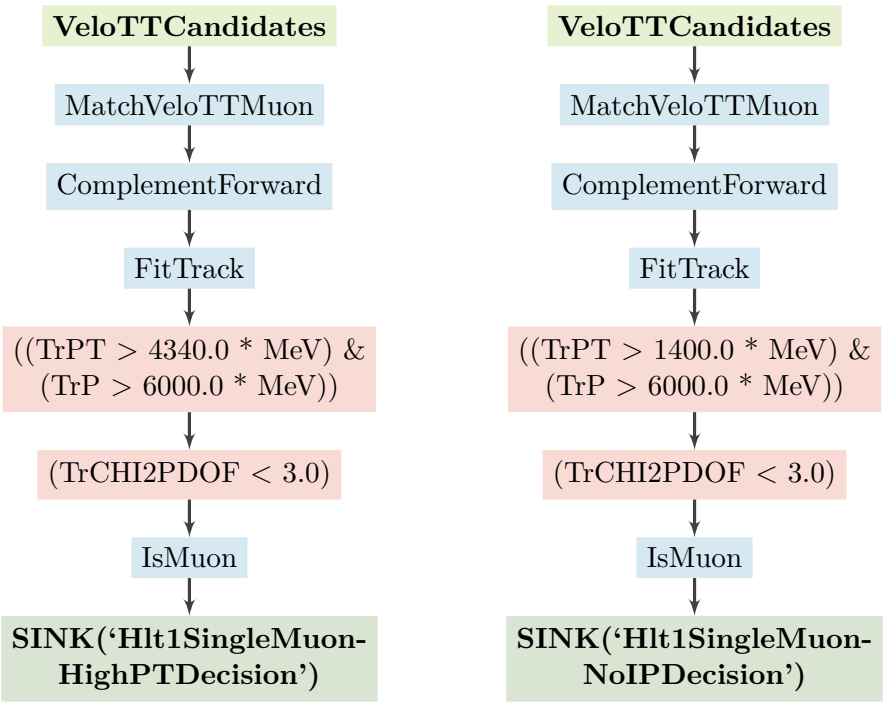
(a) `Hlt1SingleMuonHighPT` **(b)** `Hlt1SingleMuonNoIP`

Figure A.3: Flowcharts of HLT1 single muon lines in Run I.



(a) `Hlt1DiMuonHighMass` **(b)** `Hlt1DiMuonLowMass`

Figure A.4: Flowcharts of HLT1 dimuon lines for Run II.



(a) `Hlt1SingleMuonHighPT` **(b)** `Hlt1SingleMuonNoIP`

Figure A.5: Flowcharts of HLT1 single muon lines for Run II.

A.2 Muon ID Variables

Table A.3: Overview of the variables and their locations in the muon track (`LHCb::Track` with type set to `LHCb::Track::Muon`) and muon PID (`LHCb::MuonPID`) objects. In the track object they can be obtained through the `getInfo` method where the listed name corresponds to an enum variable and is used as the first parameter. The muon PID object has getter methods as listed.

Track	Muon PID	Type
`MuonMomentumPreSel`	`PreSelMomentum`	Boolean
`MuonInAcceptance`	`InAcceptance`	Boolean
`IsMuonLoose`	`IsMuonLoose`	Boolean
`IsMuon`	`IsMuon`	Boolean
`IsMuonTight`	`IsMuonTight`	Boolean
`MuonDist2`	–	Double
`MuonDLL`	–	Double
`MuonNShared`	`nShared`	(Positive) integer
`MuonChi2perDoF`	–	Double
–	`MuonLLMu`	Double
–	`MuonLLBg`	Double
–	`idTrack`	`LHCb::Track*`
this	`muonTrack`	`LHCb::Track*`

A.3 IsMuonTool

```
/** Implement signature specified by the ITracksFromTrack interface.
 * For a given track perform all necessary steps for muon id on it.
 * Save an output track if isMuon is true.
 */
StatusCode IsMuonTool::tracksFromTrack(const LHCb::Track& track,
    std::vector<LHCb::Track*>& tracks) {
  if (!m_muonTool->preSelection(track)) {
    return StatusCode::SUCCESS;
  }
  const auto extrapolation = m_muonTool->extrapolateTrack(track);
  if (!m_muonTool->inAcceptance(extrapolation)) {
    return StatusCode::SUCCESS;
  }
  CommonConstMuonHits hits;
  ICommonMuonTool::MuonTrackOccupancies occupancies;
  std::tie(hits, occupancies) =
      m_muonTool->hitsAndOccupancies(track, extrapolation);
  if (m_muonTool->isMuon(occupancies, track.p())) {
    // Add found hits to track
    LHCb::Track* output = track.clone();
    tracks.push_back(output);
    for (const auto& hit : hits) {
      output->addToLhcbIDs(LHCb::LHCbID{hit->tile()});
    }
  }
  return StatusCode::SUCCESS;
}
```

Selection criteria for the comparative study

Table A.4: Selection for $B^+ \to J/\psi K^+$ channels; DOCA is the distance of closest approach between the two tracks, VDS is the secondary vertex flight distance significance, and DLL is the combined PID likelihood to discriminate different particle hypotheses.

Variable	Applied on	Cut value
track χ^2/ndf	μ / h	< 3
ghost prob		
DOCA		$< 0.3\,\mathrm{mm}$
IPχ^2		> 25
p_T		> 0.25 and $< 40\,\mathrm{GeV/c}$
p		$< 500\,\mathrm{GeV/c}$
`IsMuon`	μ only	true
vertex χ^2	J/ψ	< 9
VDS		> 15
$\lvert M(\mu\mu) - m_{J/\psi}\rvert$		$< 60\,\mathrm{MeV/c^2}$
IPχ^2	B^+	< 25
t		$< 9 \cdot \tau(B_s^0)$
BDTS		> 0.05
$\lvert M(J/\psi K) - m_B\rvert$		$< 100\,\mathrm{MeV/c^2}$

Bibliography

[1] The LHCb collaboration, C. Langenbruch *et al.*, *Angular analysis of the $B^0 \to K^{*0}\mu^+\mu^-$ decay* (2015). LHCb-CONF-2015-002.

[2] The LHCb HLT project, J. Albrecht *et al.*, *Performance of the LHCb High Level Trigger in 2012*, J. Phys. Conf. Ser. **513** (2014) 012001.

[3] The LHCb collaboration, A. A. Alves Jr. *et al.*, *The LHCb Detector at the LHC*, JINST **3** (2008) S08005.

[4] The LHCb collaboration, R. Aaij *et al.*, *LHCb Detector Performance*, Int. J. Mod. Phys. **A30**.07 (2015) 1530022.

[5] The LHCb collaboration, B. Adeva *et al.*, *Roadmap for selected key measurements of LHCb* (2009). LHCB-PUB-2009-029.

[6] The LHCb collaboration, *Material for Presentations*, `https://lhcb.web.cern.ch/lhcb/speakersbureau/html/Material_for_Presentations.html`.

[7] R. Lindner, *LHCb layout_2. LHCb schema_2*, LHCb collection, Feb. 2008. LHCb-PHO-GENE-2008-002.

[8] O. Callot, *FastVelo, a fast and efficient pattern recognition package for the Velo*, tech. rep. LHCb-PUB-2011-001.

[9] O. Callot and S. Hansmann-Menzemer, *The Forward Tracking: Algorithm and Performance Studies*, tech. rep. LHCb-2007-015.

[10] M. Needham and J. van Tilburg, *Performance of the track matching*, tech. rep. LHCb-2007-020.

[11] O. Callot and M. Schiller, *PatSeeding: A Standalone Track Reconstruction Algorithm*, tech. rep. LHCb-2008-042.

[12] R. Frühwirth, *Application of Kalman filtering to track and vertex fitting*, Nucl. Instrum. Meth. **A262** (1987) 444–450.

[13] M. De Cian, *Track Reconstruction Efficiency and Analysis of $B^0 \to K^{*0}\mu^+\mu^-$ at the LHCb Experiment*, PhD thesis, Zürich U., Sept. 2013. CERN-THESIS-2013-145.

[14] G. Lanfranchi *et al.*, *The Muon Identification Procedure of the LHCb Experiment for the First Data* (2009). LHCb-PUB-2009-013.

[15] The LHCb collaboration, M. Adamus *et al.*, *LHCb muon system*, technical design report, 2001. CERN-LHCC-2001-010.

[16] The LHCb collaboration, *Addendum to the muon system technical design report* (2003). CERN-LHCC-2003-002.

[17] The LHCb collaboration, *Second addendum to the muon system technical design report* (2005). CERN-LHCC-2005-012.

[18] The LHCb collaboration, A. A. Alves Jr. *et al.*, *Performance of the LHCb muon system*, JINST **8** (2013) P02022.

[19] The LHCb collaboration, R. Aaij *et al.*, *The LHCb Trigger and its Performance in 2011*, JINST **8** (2013) P04022.

[20] N. Neufeld, *The LHCb Event Filter Farm (EFF)*, `https://lbonupgrade.cern.ch/doku.php?id=outreach:the_lhcb_eventfilterfarm`.

[21] M. Frank *et al.*, *Deferred High Level Trigger in LHCb: A Boost to CPU Resource Utilization*, Journal of Physics: Conference Series **513**.1 (2014) 012006.

[22] G. Dujany and B. Storaci, *Real-time alignment and calibration of the LHCb Detector in Run II* (2015). LHCb-PROC-2015-011.

[23] S. Benson *et al.*, *The LHCb Turbo Stream* (2015). LHCb-PROC-2015-013.

[24] R. Aaij, *Triggering on CP Violation Real-Time Selection and Reconstruction of $B_s^0 \to J/\psi\phi$ Decays*, PhD thesis, Vrije U., Amsterdam, Jan. 2015. CERN-THESIS-2015-102.

[25] V. V. Gligorov *et al.*, *The HLT inclusive B triggers* (2011). LHCb-PUB-2011-016.

[26] R. Aaij and J. Albrecht, *Muon triggers in the High Level Trigger of LHCb* (2011). LHCb-PUB-2011-017.

[27] M. W. Kenzie, *HLT Piquet Report*, LHCb Physics Performance, Trigger & Stripping Meeting 2015-08-31, `https://indico.cern.ch/event/400890/`.

[28] *Work predominantly done by Francesco Dettori and Ricardo V. Gomez.*

[29] C. J. Clopper and E. S. Pearson, *The use of confidence or fiducial limits illustrated in the case of the binomial*, Biometrika **26** (1934) 404–413.

[30] P. N. Schaack, *Measurement of the decay $B_s^0 \to \phi\mu^+\mu^-$ at LHCb*, PhD thesis, IC, London, Mar. 2012. CERN-THESIS-2012-064.

[31] Particle Data Group, K. A. Olive *et al.*, *Review of Particle Physics*, Chin. Phys. **C38** (2014) 090001.

[32] S. Stahl, *The LHCb High Level trigger in Run 2* (), European Physical Society Conference on High Energy Physics 2015, Vienna, Austria, 22–29 Jul 2015. LHCb-TALK-2015-182.