VNIVERSITAT DE VALÈNCIA

Facultat de Física

Departament de Física Atòmica, Molecular i Nuclear

Institut de Física Corpuscular (UV-CSIC)

Triggering new discoveries: development of advanced HLT1 algorithms for detection of long-lived particles at LHCb

Brij Kishor Jashal

Directed by:

María Aránzazu de Oyanguren Campos

PhD thesis

Doctorado en Física

Valencia, España

August, 2023

Dra. MARÍA ARÁNZAZU DE OYANGUREN CAMPOS
Profesora Titular, Universitat de València

La Dra. María Aránzazu de Oyanguren Campos, profesora titular del Dpto. de Física Atómica, Molecular y Nuclear de la Universitat de València
CERTIFICA:

que la presente memoria titulada **Triggering new discoveries: development of advanced HLT1 algorithms for detection of long-lived particles at LHCb**, ha sido realizada bajo mi dirección en la Universitat de València por Brij Kishor Jashal y constituye su tesis para optar al título de Doctor por la Universitat de València.

Y para que así conste, en cumplimiento con la legislación vigente, firmo el presente certificado.

Firmado
Dra. María de Aránzazu Oyanguren Campos

Declaration

This dissertation is the result of my own work, except where explicit reference is made to the work of others, and has not been submitted for another qualification to this or any other university.

Brij Kishor Jashal

Acknowledgements

I want to take this opportunity to express my deep appreciation and gratitude to the individuals and institutions who have been instrumental in helping me complete my PhD thesis. Without their support, this journey would not have been possible.

My heartfelt gratitude to my supervisor, Prof MARÍA ARÁNZAZU DE OYANGUREN CAMPOS, for her dedication, her unwavering support, her patience and invaluable guidance throughout this research endeavor. Her wisdom and mentorship have been the cornerstone of my academic growth.

The camaraderie and collaboration of my fellow researchers and colleagues have been a source of inspiration and intellectual growth. I am grateful for the stimulating discussions and shared experiences that enriched my academic journey. I especially thank the members of my IFIC-LHCb group (former and current), Jiahui, Luismi, Louis, Carlos, Fernando, Clara, Jose, Joan and Izaac who I made lifelong memories with and who supported me throughout this adventure.

I would like to thank the LHCb collaboration, the RTA and Allen team leaders and members for providing the opportunities, guidance and support without which this work could not have been conceived.

I would like to specially express my deep gratitude to my seniors and colleagues at the Department of High Energy Physics at Tata Institute of Fundamental Research, Mumbai for their unparalleled patience and support.

I would also like to extend my warm appreciation to the members of

my thesis committee, Prof Xavier Vilasis Cardona, Prof José Salt Cairols, Prof Kajari Mazumdar, Dr Peter Elmer, Dr Miriam Calvo Gómez, Dr Emma Torró. Their thoughtful insights and critical feedback have played an important role in shaping the quality of this thesis.

The financial and logistical support provided by CSIC, Ministerio de Ciencia e Innovación Spain, Faculty of Physics U.Valencia, Department of Atomic Energy India, CERN, IRIS-HEP NSF, and CNRS has been essential to the successful completion of this research. I am deeply grateful for the resources they offered, which allowed me to carry out this work.

My family have been unwavering in their belief in me. Their support during the challenging moments has been my source of strength. I can't thank them enough for their encouragement and understanding.

I also want to acknowledge any other individuals, organizations, or resources that have played a role in supporting my academic and research pursuits. Your contributions have not gone unnoticed or unappreciated.

The support and encouragement I've received from these individuals and institutions have made it all possible. I am deeply thankful for their contributions and for being part of this significant journey.

With heartfelt appreciation.

Preface

The physics opportunities offered by the next generation of Large Hadron Collider (LHC) based experiments come with challenges. The large number of proton-proton collisions due to the high luminosity means having to deal with higher pile-up and high data-rates. To cope with these conditions, the LHCb experiment has developed sophisticated two-stage trigger systems to select interesting events for analysis. For LHCb Run3 and beyond, the first stage of the trigger, the High Level Trigger 1 (HLT1), has been implemented on Graphic Processor Units (GPUs) and is capable of reducing the visible collision rate from 30 MHz to 1 MHz. These triggers are designed to identify events that could be of scientific interest and to discard events that are not relevant.

One alluring research avenue in particle physics is the study of long-lived particles (LLPs) of the Standard Model (SM) as well as beyond the standard model (BSM). Many interesting decay modes involve strange particles with large lifetimes such as Λ or K_S^0. Exotic LLPs are also predicted in many new theoretical models. The selection and reconstruction of these LLPs is a challenge. These particles can decay far from the primary interaction vertex and are hard to select by the trigger systems of the experiments and difficult to isolate from the SM backgrounds.

In this thesis the new trigger system of LHCb is introduced. Some of the key reconstruction and selection algorithms, which have been developed for highly parallel computing architectures, are presented. How these algorithms are instrumental in the studies and searches of LLPs from the SM and BSM are discussed.

This thesis is organised as follows:

Chapter 1 gives the theoretical framework in the context of this thesis. It is devoted to a brief introduction of the Standard Model of particle physics emphasising on particles with long lifetimes, and explaining some models beyond the SM which also accommodate new long-lived particles.

In Chapter 2 the experiment where this work has been carried out, LHCb at LHC, is described in detail, and differences between the previous detector and the one taking data at present is outlined.

One of the main challenges of the new LHCb detector is the design and commissioning of a new full-software-based trigger system.

The development of the first level of this new trigger (HLT1) is explained in Chapter 3. The Allen project and its software framework are described, highlighting the main design principles and features. The development of the Allen portability layer, which is one of the contributions of this thesis, is explained.

Chapter 4 outlines the algorithms operating inside Allen, and focuses on the contribution to two of them: the `HybridSeeding` algorithm, to reconstruct tracklets in the last tracker of LHCb, the SciFi, and the `Matching` algorithm, which is reconstructing *long* tracks that traverse all LHCb subdetectors.

In Chapter 5, a major achievement of this thesis is presented. It explains in detail the development and performance of a new `Downstream` algorithm, which is included for the first time in HLT1. The challenges and difficulties posed by the computing requirements, including throughput and the need for ghost track rejections are discussed. The solutions adopted to accomplish this task are reported and the performance of this new algorithm is highlighted.

Chapter 6 presents the plan to commission the recently installed Upstream detector of LHCb using K^0_S and Λ particles reconstructed by the `Downstream` algorithm. Selection and monitoring lines are described, which can be used for the alignment and calibration of the detector.

In Chapter 7 the implications of the implementation of the `Downstream` algorithm are outlined. It opens a new scale of detection for long-lived particles beyond 100 ps, and the impact on physics in the SM and beyond

is discussed.

Chapter 8 presents a study of a rare decay channel which is relevant in the context of this thesis: the $\Lambda_b^0 \to \Lambda\gamma$ decay channel. Observables such as the photon polarisation and the branching ratio are also sensitive to new physics scenarios. Since the decay involves Λ baryons, the work of this thesis is crucial to increase the available statistics for these types of events. The analysis of the $\Lambda_b^0 \to \Lambda\gamma$ branching fraction, relative to the $B^0 \to K^{*0}\gamma$ normalisation mode, is described in detail.

In Chapter 9 the conclusions of this thesis are summarised.

Contents

1 Introduction

This chapter is devoted to a brief introduction to the Standard Model (SM) and some of the interesting models beyond it which are relevant in the context of this thesis. After a compact description of the main features of the SM (Sec. 1.1), the need of new physics is addressed in Sec. 1.3. Some of the possibles scenarios which could help to fill the gaps in the SM, which involve long-lived particles, are described in this section. In addition, Sec. 1.4 explains how long-lived particles in the SM can also probe new physics through the precision measurement of observables in radiative b-baryon decays.

1.1 The Standard Model

The Standard Model of particle physics is a theoretical framework that describes the fundamental particles and forces that make up our universe. It is a highly successful model that has been rigorously tested and verified by experiments. The Standard Model describes three of the four fundamental forces in nature: the electromagnetic force, the weak force, and the strong force. It does not include gravity, which is described by the theory of general relativity. According to the Standard Model, matter is made up of particles called *fermions*, with spin 1/2, which include *quarks* and *leptons*. Quarks are the building blocks of protons and neutrons, which make up atomic nuclei, while leptons include particles such as electrons and neutrinos.

The SM also describes the interactions between these particles through the exchange of other particles, called *bosons*, which have integer spin.

The electromagnetic force is mediated by the *photon*, while the weak force is mediated by the $W^{\pm}$ and Z bosons. These two forces are joined together and known as the Electroweak (EW) force. The strong force, which holds quarks together inside protons and neutrons, is mediated by particles called *gluons*. The theory explaining these interactions is called Quantum Chromodynamics (QCD). In addition to fermions and bosons, the Standard Model predicts the existence of the Higgs boson, which is responsible for the mass of a given particle, including itself. This was a major discovery in 2012 at the Large Hadron Collider (LHC) [1, 2]. Measurements at the LHC have established that the properties of this Higgs boson match with the predictions of the SM.

In Fig. 1, a schematic view of the SM content with the three generations of fermions, the mediator spin-1 bosons as well as the scalar Higgs boson is presented along with their basic properties. Each particle has a corresponding antiparticle, with the same mass but opposite quantum numbers. Particles with vanishing quantum numbers, such as the Z or γ bosons, are their own antiparticles. There are six types of quarks, organised in three generations: *up* and *down*; *charm* and *strange*; and *top* and *bottom*. These quarks can be combined in different ways to form various particles. *Mesons* are composed of a quark and an anti-quark, while *baryons* consists of three quarks. For example, protons are composed of $[uud]$, B^0 of $[\bar{b}d]$ and Λ_b^0 of $[udb]$. From the theoretical point of view, the SM is described by a relativistic Quantum Field Theory (QFT) [3]. In this formulation, space-time is explained as being filled with different types of fields. The elementary particles are manifestations of the excitations of these quantum fields, and the interactions between the fields correspond to the fundamental forces explained above (electromagnetic, strong and weak). Mathematically, those interactions are described by the Lagrangian

$$\mathcal{L}_{\text{SM}} = \mathcal{L}_{\text{EW}} + \mathcal{L}_{\text{QCD}} + \mathcal{L}_{\text{Higgs}} + \mathcal{L}_{\text{Yukawa}}\,, \tag{1.1}$$

where the first two terms correspond to the electroweack (EW) and strong (QCD) fundamental forces, respectively. The third (Higgs) term is responsible for the interactions between the Higgs and the other massive gauge bosons, whereas the last term refers to the Yukawa couplings, describing

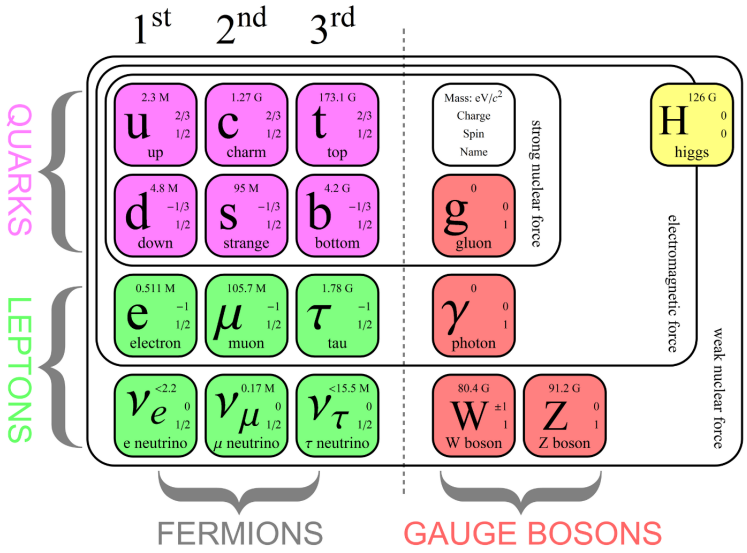


Figure 1: The Standard Model (SM) of particle Physics. The three generations of matter (fermions) is shown in the left, while interactions, with the force carriers, are shown in the right (bosons). The Higgs boson occupies a special place since its field is responsible for giving mass to particles.

the interactions between the Higgs and the fermions. Even if the mass of a particle is still a property somewhat unclear, it is the interaction with the Higgs boson what provides masses to fermions and to the Z and $W^{\pm}$ bosons through the spontaneous symmetry breaking mechanism [4, 5].

Despite its successes, the SM does not explain all phenomena in the Universe, such as the missing dark matter and dark energy. It is also not consistent with the theory of general relativity, which describes gravity, since the *graviton*, the expected mediator particle, has not been observed. The matter-antimatter asymmetry in our Universe remains a mystery that cannot be explained by the SM. Therefore, physicists are still searching for a more complete theory that can unify all forces in nature and provide a more comprehensive understanding of the universe. Many of the proposed theories involve what it is called "long-lived particles (LLP)", and they are explained below.

1.2 Long-lived Particles (LLPs)

The *lifetime* of a particle is a measure of how long an unstable particle is expected to exist before decaying into other particles. It is often represented by the symbol τ. The lifetime of a particle is inversely proportional to its decay width Γ ($\Gamma = 1/\tau$). The latter is a quantity that represents the probability of a particle to decay into a specific set of final-state particles per unit time. Denoting i as one particular decay channel, the total decay width is the sum of the partial decay widths for all possible decay channels: $\Gamma = \Sigma\Gamma_i$. The decay width depends on factors such as the mass of the decaying particle, the masses of the final-state particles, the strength of the interactions involved, and the phase space available for the decay process. A larger decay width indicates a higher probability for that decay to occur.

Particles that decay via stronger interactions, such as the strong nuclear force or the electromagnetic force, generally have shorter lifetimes, while those decaying via weaker interactions, like the weak nuclear force, have longer lifetimes. The probability that a particle with a given lifetime will decay within a certain time interval is described by an exponential decay law: $P(t) = 1 - e^{-\Gamma t}$, where $P(t)$ is the probability of decay within the time interval t.

In the Standard Model, a vast majority of particles are unstable and decay into lighter particles after a very short time. The range of lifetimes is very wide, from 10^{-25} s to 10^{35} years. Some of them are considered stable, such as the electron or the proton. In Fig. 2 a plot of the lifetime dispersion in the SM vs the particle mass is presented. The shadow areas show the SM particles that decay promptly or are stable from the experimental point of view.

Given this wide range, the term "long-lived particle" is ambiguous, and it usually depends on the experimental framework one is working on. In proton-proton colliders, and in the context of this thesis, long-lived particles refer to those with lifetimes starting from 100 ps and beyond, which in the SM refers mainly to the strange hadrons K^0_S and Λ.

Similarly, in case of models beyond the SM, an extensive variety of

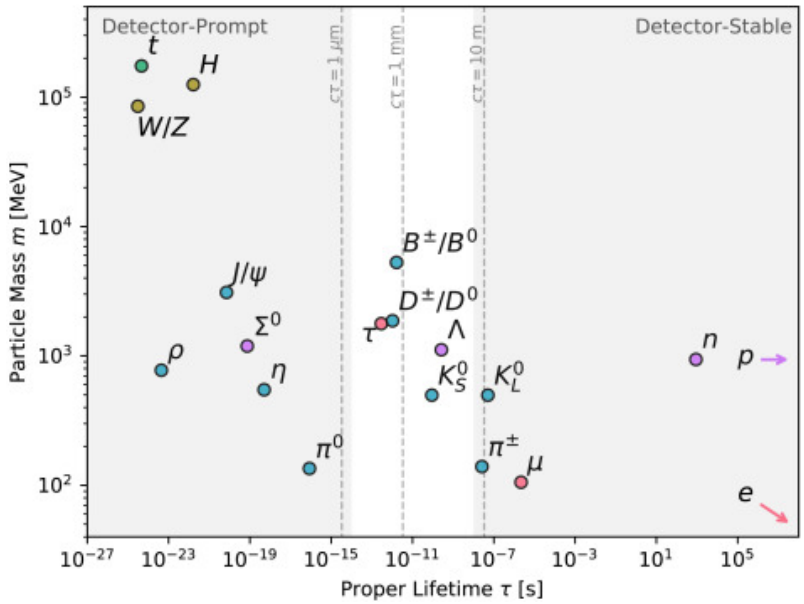


Figure 2: Lifetimes of particles in the SM vs their mass. The shadow areas indicate the regions where these particles are stable or decay promptly, according to the detector reconstruction (figure from Ref. [6]).

particle lifetimes is predicted in a new weak scale. This is supported by approximate symmetries that stabilize the LLP, with small couplings between the LLP and lighter states, or suppressed phase space available for decays. In the context of this thesis, the sensitivity to LLPs is limited by the geometrical acceptance of the LHCb detector and the studies here are delimited to lifetimes below several nanoseconds.

1.3 Long Lived Particles Beyond the Standard Model (BSM)

Many theories beyond the Standard Model accommodate easily and predict particles with long lifetimes. Their presence is strongly motivated because they typically are expected to be feebly interacting and thus very challenging to be detected by experiments. LLPs are also good candidates to understand dark matter and very relevant in cosmology since they can be related to the leptogenesis and baryogenesis processes in the early Universe. In general, for a new LLP of mass m decaying in a process involving a heavy off-shell particle with mass M, one can express

its decay width as:

$$\Gamma \sim \frac{\epsilon^2}{(8\pi)^{a-1}} \frac{m^n}{M^{n-1}} \tag{1.2}$$

with ϵ a potentially small coupling constant and n a positive integer that depends on the symmetries of the theoretical framework. The parameter a is also a positive integer and indicates the number of final state particles, the lifetime of the LLP will be the inverse of Eq. 1.2.

Some examples of BSM theories which predict the existence of LLPs include:

- **Hidden Sector particles:**

  Models in the Hidden Sector predict the existence of new particles and forces that do not interact with the SM particles or do it very weakly. They should be accessible at the current experimental energies but due to the small couplings they are very difficult to detect. Different vector and scalar "portals" to this new physics can appear, a Dark Photon (A') and a new Higgs boson (H') being some of the favorite candidates to couple to the new particles. Since these new particles are hardly visible by experiments, they are good candidates to dark matter and these type of models are often called "Dark Sectors". In Fig. 3 an example of a Feynman diagram where a heavy scalar (Φ or a Higgs) decays into long-lived scalars S.

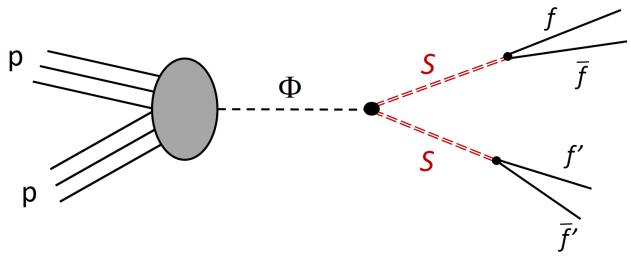


Figure 3: Example of an interaction mediated by dark scalars in the Hidden Sector with several fermions (f) in the final state.

- **Heavy Neutral Leptons (HNLs):**

  Since the origin of neutrino masses is not known in the SM, neither integrated in its first formulation, theories are being developed to include new heavy right-handed neutrinos, with masses larger than

the eV scale, that enter in the Yukawa couplings with the leptonic doublet and the Higgs field. They would be sterile particles, not interacting and being very difficult to detect. These new leptons, called Heavy Neutral Leptons (HNL), would help to understand why the SM neutrinos have smaller masses as compared to other fermions, and why the mixing between leptons (described by the PMNS mixing matrix[1]) presents a different structure as compared to the quarks one (described by the CKM mixing matrix). The existence of HLNs could explain the baryon asymmetry of the Universe after the leptogenesis period, with a CP violation[2] increase due to the neutrino oscillations in the early Universe. It would also constitute a dark matter candidate, if there are enough HNL and they are heavy enough. Due to their weak interaction with the SM particles, they could have long lifetimes. Fig. 4 shows an example where a HNL is interacting with SM particles.

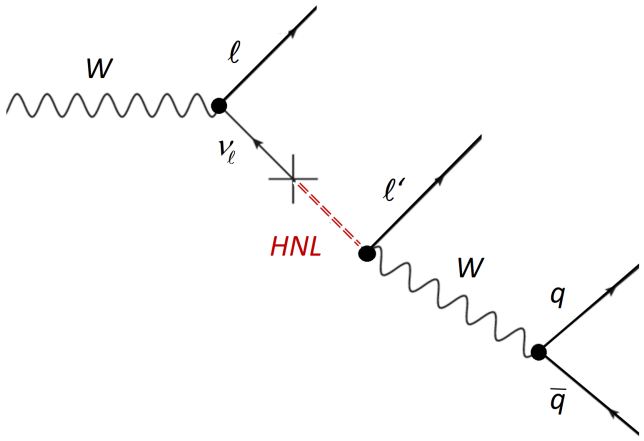


Figure 4: Example of the production of a HNL via mixing with a SM ν from the decay of a W boson and the semileptonic decay of the HNL into a lepton and two quarks.

- **Supersymmetry (SUSY)**:

  Supersymmetry is a theoretical framework that predicts a new set of particles called superpartners for each known particle in the

[1]The PMNS (Pontecorvo-Maki-Nakagawa-Sakata) matrix is a unitary matrix which describes the mixing of neutrino flavor states.

[2]CP violation is explained in Sec. 1.4.1.

Standard Model. Its fundamentals relates fermions with bosons via supersymmetric transformations, which keep same quantum numbers but different spin. Thus, for example, since electrons exist there should also be *selectrons*, with spin 0. There should also be *photinos* with spin 1/2, to take their place alongside photons, and so on. If supersymmetry were exact, these particles would have the same mass as their partners and would have been discovered, but since none has yet been discovered, it is expected that supersymmetry is broken at a few hundred GeV scale or higher. Two popular SUSY breaking mechanisms are the gravity-mediated and the gauge-mediated (GMSB). The superpartners are expected to have masses less than TeV for the SUSY to solve the hierarchy problem [7], and are expected to be discovered at the LHC. Some SUSY models predict long-lived particles, such as the lightest supersymmetric particle (LSP), which is also a candidate for dark matter. The LSP is stable because of a new conserved quantum number called R-parity, which prevents it from decaying into Standard Model particles. Fig. 5 shows and example of a supersymmetric transition.

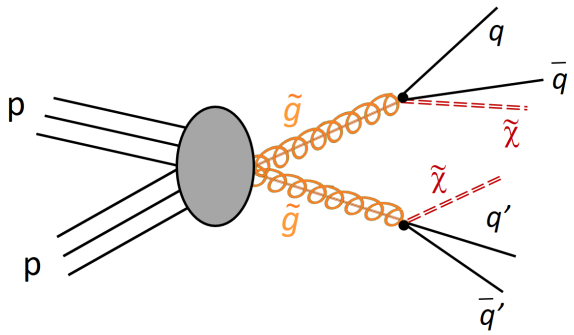


Figure 5: Example of the production of two *gluinos* ($\tilde{g}$), which are long-lived, and create two SUSY-hadrons composed of quarks (q) and *charginos* ($\tilde{\chi}^{\pm}$) or *neutralinos* ($\tilde{\chi}^{0}$).

- **Axion-Like Particles (ALPs):**

  Axion-like particles (ALPs) are light, neutral, pseudo-scalar bosons predicted by several extensions of the Standard Model, such as String theory, which are expected to interact primarily with two

photons. They are a generalisation of the *axion*, the pseudo-Goldstone boson which is related to the global Peccei-Quinn symmetry $U(1)_{PQ}$, proposed as a natural solution to the strong CP problem [8], to understand why the neutron electric dipole moment (eDM) is so small. While axions can interact with fermions, gluons, and photons, and a strict relationship between the axion mass and the coupling constant exists. ALPs are supposed to couple primarily only to two photons and the mass m_a and the coupling constant are unrelated parameters. ALPs are predicted to be very light and interact very weakly,leading to long lifetimes. If they exist, they could also be a component of dark matter. Figure 6 shows an example of a process mediated by an ALP.

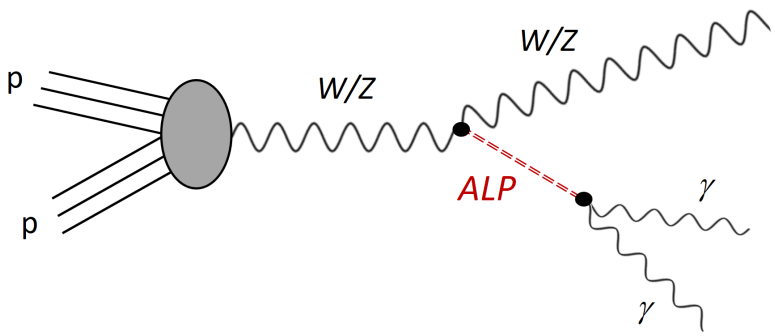


Figure 6: Example of a ALP-strahlung process, with two photons in the final state.

1.3.1 Experimental status

Several searches have been performed by the LHCb collaboration with the data collected during 2011 to 2012 (Run1) and 2015-2018 (Run2) of the LHC. The experimental signatures include:

- events with a lepton from a high-multiplicity displaced vertex [9],
- events with two displaced high-multiplicity vertices [10, 11],
- decays of B mesons (mediated by a Majorana neutrino) to a final state with two same-sign leptons associated to different vertices [12, 13],
- B meson decays to a final state with two opposite-sign leptons forming a displaced vertex [14, 15],

- prompt tracks (charged massive stable particles) extending to the muon stations with velocity lying below the threshold for producing Cherenkov light in the RICH detector [16].
- prompt muons which form a high-quality displaced vertex [17].

Figure 7 shows the present exclusion limits obtained by the LHCb experiment for the search of two prompt muons coming from a dark photon, together with the limits obtained by other experiments. LHCb offers a unique coverage amongst all experiments.

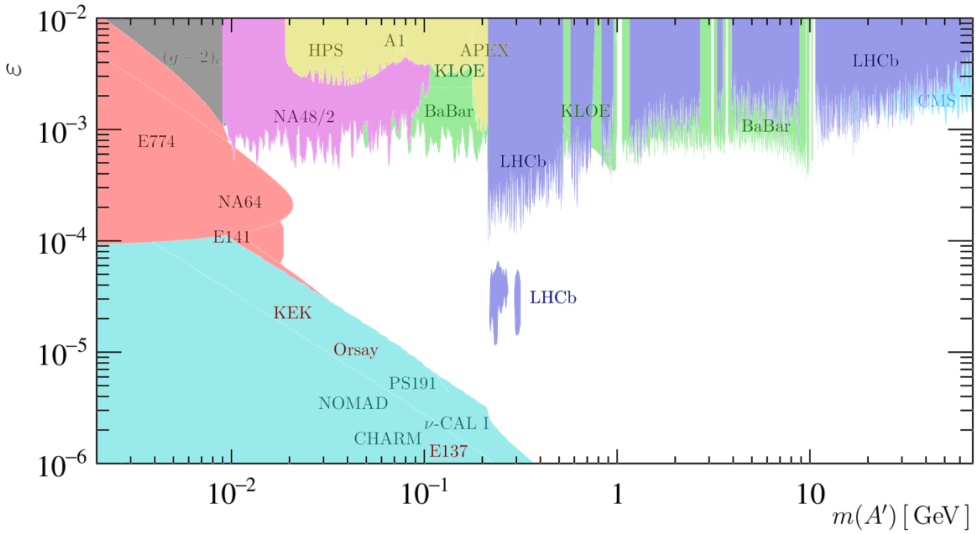


Figure 7: Exclusion limits for the search of a dark photons (A') in the parameter space of the γ-A$'$ mixing parameter ϵ versus the dark photon mass. The LHCb coverage is based on a displaced signature in the decay of A'$\to \mu^+\mu^-$.

Extensive LLP searches have also been carried out by the ATLAS and CMS experiments, reconstructing several signatures such as detached muons, photons not pointing to the primary vertices, displaced di-electromagnetic vertices ($H \to \gamma\gamma$ or $Z \to e^+e^-$), trackless and delayed jets, and multi-charged particles. A summary of the searches and reach obtained by ATLAS is shown in Fig. 8. The model and the expected signature, together with the analysed luminosity is quoted in the table. References for each analysis is also quoted in the last row.

Apart from the LHC general-purpose detectors, ATLAS and CMS, there are several other dedicated experiments which have been designed

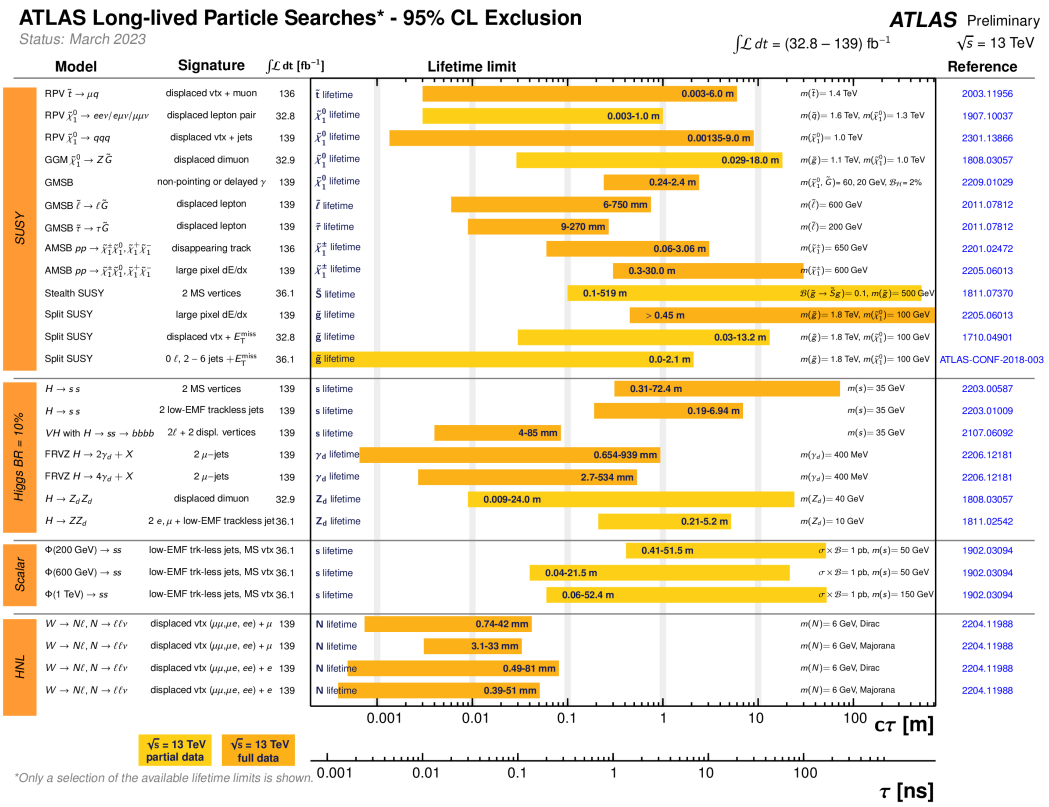


Figure 8: 95% CL exclusion limits obtained by the ATLAS experiment on different models. Figure from [18].

and proposed for LLP searches such a MoEDAL-MAPP [19], MATHUSLA [20], CODEX-b [21], FASER [22] and SHIP [23]. A comprehensive review of the current status and future proposals of LLP searches across the experiments outlines the importance of LLPs and can be found in Ref. [24]. The lifetime of the hypothetical particle plays an important role in the design of the experiments. Until now the majority of searches for new physics have been based on the underlying idea of prompt decays of particles wherein particles decay close to the Interaction Point (IP) and decay products traverse through rest of the detector layers. Most of the new physics (NP) searches are based on signatures which are built using properties of candidate particles traversing outwards of vertex detectors such as tracking information, missing transverse energy, *etc.* The newly proposed experiments aim to extend their reach by being able to search for new particles with longer lifetimes than before. Alongside these dedicated experiments, ongoing upgrades of ATLAS, CMS and LHCb are expected to provide enhanced sensitivity for the LLP searches.

The work developed in this thesis is devoted to widen the physics potential of the present LHCb experiment to detect BSM particles with lifetimes beyond 100 ps. As explained in the following chapter, the development of reconstruction algorithms at early stages of the trigger is crucial.

1.4 Long Lived Particles in the SM

The dispersion in the lifetimes of SM particles has already been stated in Fig. 2. Particles with $\tau \leq 10^{-15}$ s decay via the strong or electromagnetic interactions and are considered here to have short lifetimes. They decay promptly in the LHC detectors. Particles with lifetimes between 10^{-15} s and 10^{-10} s are decaying weakly and are considered long-lived particles, specially those involving strange quarks, like the Λ and K^0_S. For particles with lifetimes larger than 10^{-8} s, since they are traversing all detector volume before decaying, they are considered here as stable particles. Long-lived particles in these range, from 10^{-15} s to 10^{-10} s, are of special interest in the SM phenomenology since they are suffering from the violation of one of the fundamental conservation laws in nature, the CP-symmetry. The concept of CP (charge-parity) symmetry in particle physics is based on the idea that the behavior of a physical system should be the same if all particles are replaced with their antiparticles and the spatial coordinates are flipped. In other words, CP symmetry predicts that there should be no difference between matter and antimatter at the fundamental level. CP violation has been observed in the decays of neutral systems for b-, charm- and strange-hadrons. The study of CP asymmetries is crucial for understanding the baryon asymmetry observed in the Universe, as it relates to one of Sakharov's conditions [25]. Channels involving K^0_S are particularly relevant when studying CP violation in both the b and charm sectors, given that it is a very common decay product (eg. $B^0 \to K^0_S K^0_S$, $B^0 \to K^0_S \pi^+ \pi^-$, $D^0 \to K^0_S K^0_S$, $D^0 \to K^0_S \pi^+ \pi^-$, *etc.*).

1.4.1 CP violation in the SM

In the late 1960s, M. Kobayashi and T. Maskawa proposed a mechanism to explain CP violation in the weak interactions of quarks. They introduced the concept of quark mixing, where quarks of different generations can transform into one another through weak interactions. This mixing is described by the CKM (Cabibbo-Kobayashi-Maskawa) matrix, a unitary matrix that specifies the probabilities of one quark flavor changing into another. They proposed that there were more than two generations of quarks, and that the mixing of these quarks in weak interactions could account for the observed CP violation. They also predicted that CP violation would be most significant in the interactions involving the third generation of quarks (top and bottom). The CKM matrix contains a complex phase which is the only source to account for CP violation (CPV) in weak interactions, which was first observed in experiments with neutral kaons in the 1960s [26]. Later, in the 1980s, CPV was observed in the B meson systems by the B-factories at SLAC [27] and KEK [28] and, just a few years ago, CPV was observed in the charm sector by the LHCb experiment [29].

The CKM matrix is unitary and complex by construction. It connects the mass eigenstates *(d, s, b)* with the flavour eigenstates *(d′, s′, b′)* in the way:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = V_{\text{CKM}} \begin{pmatrix} d \\ s \\ b \end{pmatrix} , \tag{1.3}$$

with

$$V_{\text{CKM}} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} . \tag{1.4}$$

The V_{ij} matrix elements represent the strength of the transition from an *up*-type quark (i) to a *down*-type quark (j) via the weak interaction. The probability of this transition is proportional to $|V_{ij}|^2$. The parameters of the CKM matrix are not predicted by the SM, they are fundamental

parameters that must be experimentally determined. Due to unitary constraints, This matrix depends only on four parameters, three angles and one complex phase. This phase is responsible for CP violation in the SM. There are several parameterisations for the CKM matrix, one of them is the Wolfenstein [30], defined as

$$V_{\text{CKM}} = \begin{pmatrix} 1-\lambda^2/2 & \lambda & A\lambda^3(\rho - i\eta) \\ -\lambda & 1-\lambda^2/2 & A\lambda^2 \\ A\lambda^3(1-\rho-i\eta) & -A\lambda^2 & 1 \end{pmatrix} + \mathcal{O}(\lambda^4)\,, \qquad (1.5)$$

where the A, λ, ρ and η parameters are real parameters with measured values of $\lambda \simeq 0.22$, $A \simeq 0.81$, $\rho \simeq 0.14$ and $\eta \simeq 0.35$ [31]. The CP violation of the SM is encoded in the η parameter, which only appears in the most suppressed terms $\left(\sim \mathcal{O}(\lambda^3)\right)$. The Wolfenstein parameterisation highlights that the strongest coupling happens between quarks of the same family. Transitions between the first and the second families are proportional to λ, whereas transitions from the second to the third family are of $\mathcal{O}(\lambda^2)$. The least probable transition occurs between the first and the third family and it is of $\mathcal{O}(\lambda^3)$. Studying different physics observables and measuring the CKM parameters from different decay channels one can check for inconsistencies and probe the SM phenomenology in the quark sector. If measurements are incompatible among them, this would be an evidence of the effect of new particles and mechanisms not included in the SM. Figure 9 shows the experimental constraints obtained by the CKMFitter group [31] on different observables and CKM matrix elements. These pictures give an image of the good agreement between the SM flavour phenomenology and the up-to-date experimental results. Some of the observables are linked to b-hadron rare decays, which are explained in the following.

1.4.2 Rare decays of heavy hadrons

Rare decays of b-hadrons provide a powerful way of identifying contributions from physics beyond the SM, being quite sensitive to the existence of new heavy particles. The elements of the CKM matrix described in

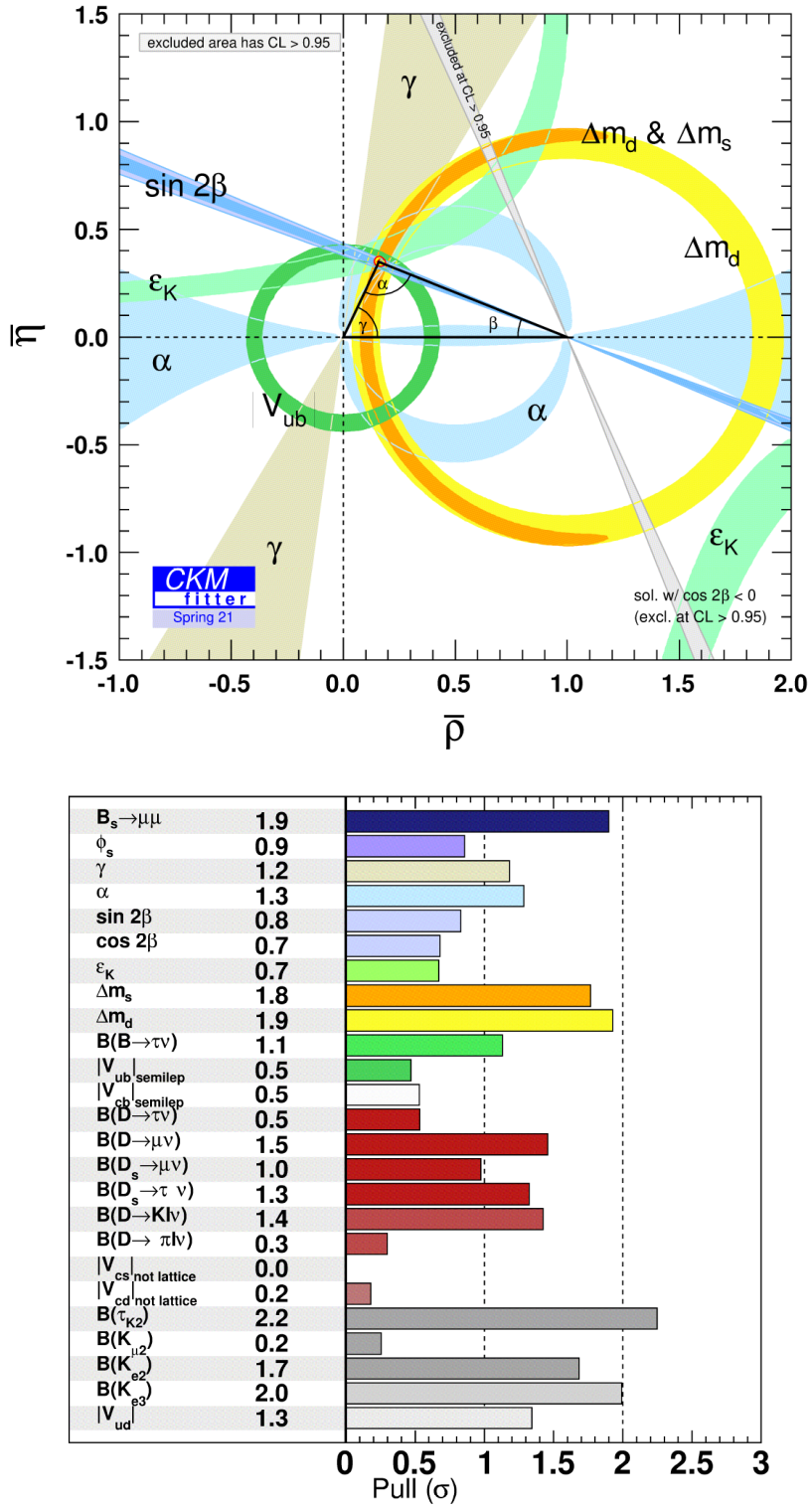


Figure 9: Constraints given by the global fit results of the CKM parameters on the ($\bar{\eta}$, $\bar{\rho}$) plane (left) and the pull distributions of the input observables (right) [31].

the previous section describes the tree level[3] transitions among quarks

[3]In quantum field theory, where the involved particles exchange virtual particles, tree level refers to direct diagrams without loop corrections.

with charged currents, via the $W^{\pm}$ boson. The quark transitions between quarks with the same electric charge, such as $b \to d$ or $b \to s$, do not happen at tree level in the SM as the Z boson does not couple to quarks of different flavour. These processes, called Flavour Changing Neutral Currents (FCNC), only can occur via *loop* level and higher order transitions, thus are very sensitive to NP.

The common theoretical approach to rare decays is model independent, the underlying physics being parameterised in terms of an effective Hamiltonian describing the transition amplitude of an initial state $|i\rangle$ to a final state $|f\rangle$:

$$\langle f|\mathcal{H}_{eff}|i\rangle = -\frac{G_F}{\sqrt{2}} V_{\mathrm{CKM}} \sum_{k=1}^{10} \mathcal{C}_k(\mu)\langle f|\mathcal{O}_k(\mu)|i\rangle \,. \tag{1.6}$$

to In this equation, G_F is the Fermi constant, which is proportional to the strength of the electroweak interaction, and V_{CKM} are the corresponding matrix elements involved in the decay process. The long-distance contributions[4] are encoded in the $O_k(\mu)$ operators, depending on the energy scale μ, whereas the Wilson coefficients, $\mathcal{C}_k(\mu)$, portray the short-distant effects, and absorb the effects of the W, Z bosons and *top* quark.

Each operator represents a different process:

- $\mathcal{O}_1$ and $\mathcal{O}_2$ are current-current operators;
- $\mathcal{O}_3$-$\mathcal{O}_6$ are strong penguin operators;
- $\mathcal{O}_7$ is the electromagnetic penguin operator;
- $\mathcal{O}_8$ is the chromomagnetic operator;
- $\mathcal{O}_9$ and $\mathcal{O}_{10}$ are the semileptonic operators.

Of most interest in rare decays are the suppressed operators O_7, O_9 and O_{10}, since a comparison from the SM prediction of the corresponding Wilson coefficients is very sensitive to new particles.

[4] "Long-distance" contributions come from processes at scales comparable to or larger than a hadron and involve non-perturbative QCD effects. In contrast, "short-distance" contributions refer to quantum effects arising from processes at small length scales, typically much smaller than a hadron, that can be calculated using perturbative QCD.

1.4.3 Radiative b-hadron decays

In the SM, the radiative transition $b \to s$ proceeds via a loop process as shown in Fig. 10. The O_7 operator dominates the process resulting in a decay width,

$$\Gamma(b \to s\gamma) = \frac{G_F^2\, \alpha_{EM}\, m_b^5}{32\, \pi^4} |V_{ts}^* V_{tb}|^2\, |C_7|^2 + \text{corrections}, \tag{1.7}$$

where G_F is the Fermi constant, α_{EM} is the electromagnetic constant, m_b is the b quark mass, and V_{ij} are the corresponding parameters of the CKM matrix. A measurement of the $b \to s\gamma$ branching fraction thus provides a direct constraint on the C_7 Wilson coefficient.

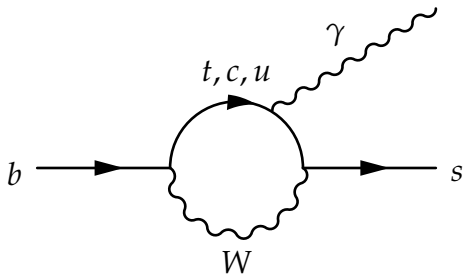


Figure 10: The $b \to s\gamma$ Feynman diagram.

If one does not assume the V-A structure of weak interactions[5], new primed operators in Eq. 1.6, with flipped helicities, appear, which generate a right-handed photon in $b \to s\gamma$ decays (i.e. C_7'). The non-zero value of this primed coefficient would be a sign of NP and thus the measurement of the photon polarisation in these transitions is very relevant, since in the SM the photon is predicted to be almost left-handed in the $b \to s\gamma$ transition, because it is mediated by the W^- boson[6]. A small right-handed contribution, of the order of 10^{-4}, is expected from the effect of the b and s quark masses.

In terms of the Wilson coefficients, the measurement of branching fractions grants access to $|C_7|^2 + |C_7'|^2$ and, hence, allows to impose circular constraints in the $(\mathcal{C}_7, \mathcal{C}_7')$ plane. Since the ratio of right- and left-

[5]It refers to the left-handed (V) and right-handed (A) chiral components of particles, with a preference for left-handed couplings in the SM.

[6]The $\bar{b} \to \bar{s}\gamma$ transition is mediated by the W^+ boson, which only couples with right-handed quarks, thus the photons emitted are predominantly right-handed.

handed contributions encoded by the Wilson coefficients satisfies $|r| = \frac{C_7'}{C_7}$, this fraction is expected to be almost zero.

1.4.4 Experimental status

Since the 1980s, the $b \to s\gamma$ transition has played a major experimental role, primarily due to the dependence in Eq. 1.6 on the *top* quark mass[7]. The first observation of the $b \to s\gamma$ decay was reported in 1993 by the CLEO collaboration [32], observing a signal of the exclusive decay $B^0 \to K^{*0}\gamma$, and measuring the branching fraction to be around $\dot{1}0^{-5}$. The Belle and BaBar experiments have measured the branching faction of the $B^0 \to K^{*0}\gamma$ and $B_s^0 \to \phi\gamma$ decay channels [33–37], and the LHCb experiment has studied the $B_s^0 \to \phi\gamma$ decay with much more statistics [38]. The photon polarisation in $b \to s\gamma$ transitions has been observed by first time at LHCb by measuring the up-down asymmetry in $B^+ \to K^+\pi^-\pi^+\gamma$ decays [39] and by studying time-dependent CP asymmetries in $B_s^0 \to \phi\gamma$ decays [40, 41]. At present all measurements are in agreement with the SM predictions.

Of special interest is the study of the b-baryon decay channel $\Lambda_b^0 \to \Lambda\gamma$. It offers a much more rich spin structure, since the ground state spin is $\frac{1}{2}$, which makes it ideal to probe the helicity anatomy in $b \to s\gamma$ transitions[8]. This decay channel was observed for first time at LHCb [42, 43], and its branching fraction has been measured using a limited data sample of 1.7 fb^{-1}. The value $\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma) = (7.1 \pm 1.5 \pm 0.6 \pm 0.7) \times 10^{-6}$ where the first uncertainty is statistical, the second systematic and the third is the systematic from external measurements - is compatible with theoretical predictions [44–46], and at present is limited by the statistical uncertainty.

Constraints in the complex Wilson coefficient plane (C_7, C_7') provided by different measurements are shown in Fig. 11. The results agree with the SM predictions. It is worth noting that the measurement of the angular distribution of $B^0 \to K^{*0}e^+e^-$ decays, where photons are virtual,

[7]The top quark was discovered at the Tevatron in 1995 and its mass measured, which determined the Standard Model decay rate of $b \to s\gamma$ to be a few 10^{-4}.

[8]The spin of the B^0 and B_s^0 meson is zero.

provides the most stringent constraints [47].

Experimentally, the main difficulty comes from the fact that the decay involves a neutral long-lived particle Λ and a photon. As a result, the selection and reconstruction of these decays is very challenging. At present the LHCb experiment has only been able to reconstruct these decays with specific signal requirements. The work of this thesis aims to increase the statistics of the selection of radiative b-baryon decays, which will allow to measure with more precision the branching fraction and the photon polarisation, both being key observables to probe the SM phenomenology.

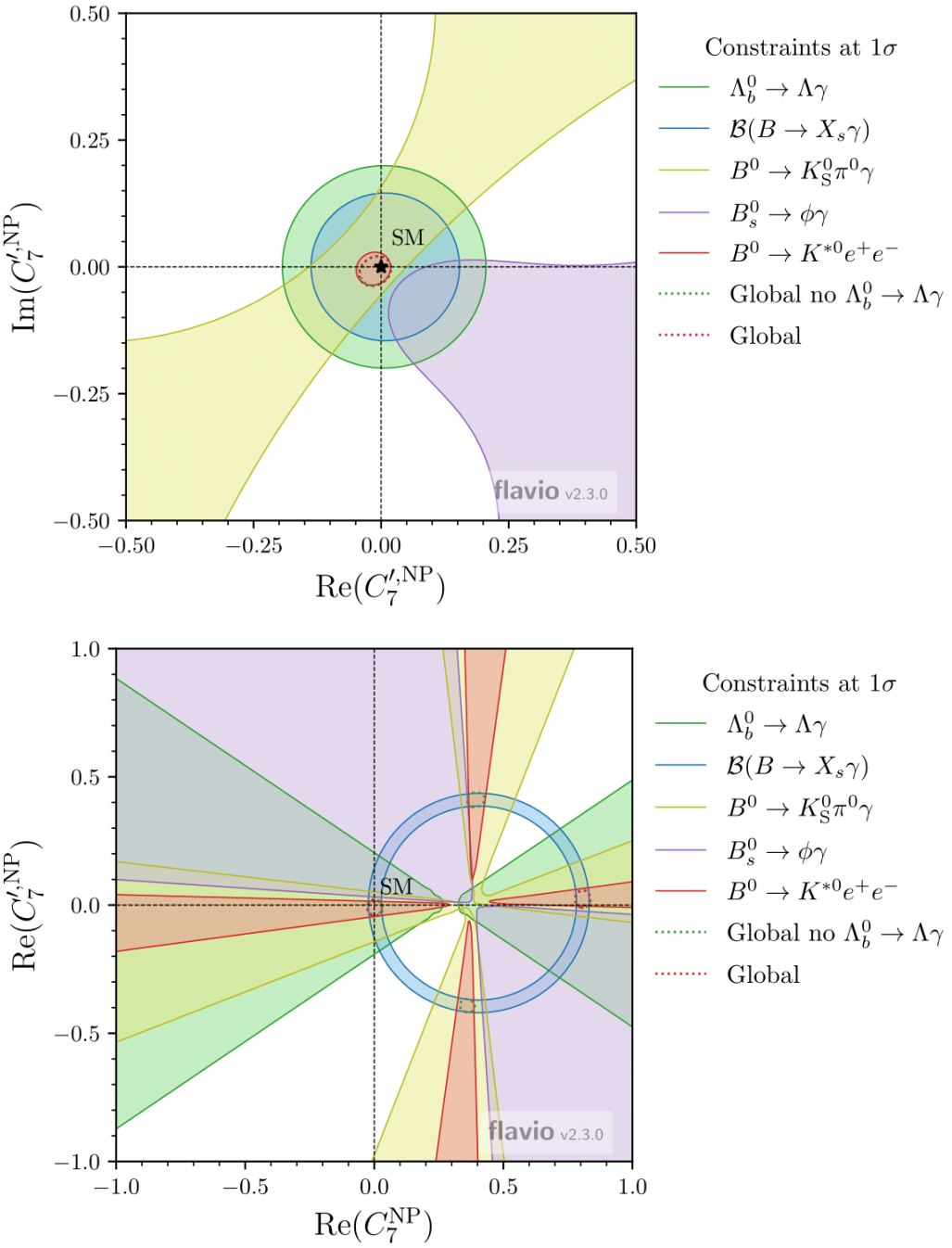


Figure 11: Constraints to the C_7' Wilson coefficient in the complex plane from the b-hadron radiative measurements [48].

2 The LHCb experiment

The experimental framework where the thesis has been developed is explained in this chapter. The proton-proton Large Hadron Collider (LHC) is described in Sec. 2.1. The LHCb detector is covered in Sec. 2.2.2, where the different subsystems are described, with special emphasis in the differences between the detector in Run2 and the upgraded detector recently installed for Run3. The detectors involved in the tracking system, which are relevant in the context of this thesis, are detailed. An essential part for this thesis is the data acquisition and the trigger systems and they are described in Sec. 2.4 and Sec. 2.5, respectively. The procedure for data processing and analysis is also explained at the end of this chapter.

2.1 The LHC machine

The LHC is located about 100 meters underground at CERN (the European Organisation for Nuclear Research), near Geneva. It is a circular accelerator with a 27-kilometer circumference, making it the largest and most powerful particle accelerator in the world. It has been operating since 2010 in three runs: Run1 (2011-2012) at 7 and 8 TeV centre-of-mass energy ($\sqrt{s}$), Run2 (2015-2018) at 13 TeV, and Run3, which started in spring 2022 at $\sqrt{s}$=13.6 TeV.

The LHC beam contains roughly 2800 bunches of protons, each having around 1.15×10^{11} protons. The instantaneous luminosity (L_{inst}) of LHC, which is proportional to the rate of collisions (and thus a critical

collider parameter) can be expressed as

$$L_{\text{inst}} = \frac{N^2 \cdot f \cdot n_b}{4\pi\sigma_x\sigma_y} \tag{2.1}$$

wherein for LHC, N is the number of protons per bunch ($N \approx 1.15 \times 10^{11}$). f is the revolution frequency which is approximately $f \approx 11.245$ kHz., n_b is the number of bunches per beam $n_b \approx 2800$, and σ_x and σ_y are the horizontal and vertical beam sizes at the interaction point. The concept of *pileup* (μ), the average number of simultaneous interactions per bunch crossing, can be expressed as

$$\mu = \frac{L_{\text{inst}} \cdot \sigma_{\text{inel}}}{f} \tag{2.2}$$

where σ_{inel} is the inelastic proton-proton cross-section, which is about 80 mb at 14 TeV. Limiting the *pileup* is important to perform precision physics, since a large amount of interactions provides a large quantity of primary vertices which are difficult to disentangle from the decay of interest.

Figure 12 displays the layout of the LHC collider complex, highlighting the positions of the different experiments. Four detectors, ATLAS, CMS, ALICE, and LHCb, are situated at various points around the LHC ring, each tailored for a specific research focus.

ATLAS and CMS are general-purpose detectors designed to explore a broad spectrum of physics, including the search for the Higgs boson. While they share similar scientific objectives, they employ distinct technical solutions and design concepts, enabling mutual cross-checks. ALICE is optimised to study the quark-gluon plasma, a state of matter thought to have existed just microseconds after the big bang, by colliding lead ions instead of protons. LHCb is a specialised flavour physics experiment studying decays of b and c quarks, investigating the differences between matter and antimatter particle decays, and also focusing in hadron spectroscopy.

2.2 The LHCb experiment

The Large Hadron Collider beauty (LHCb) experiment is dedicated to the study of flavor physics. LHCb focuses on investigating the behavior of b

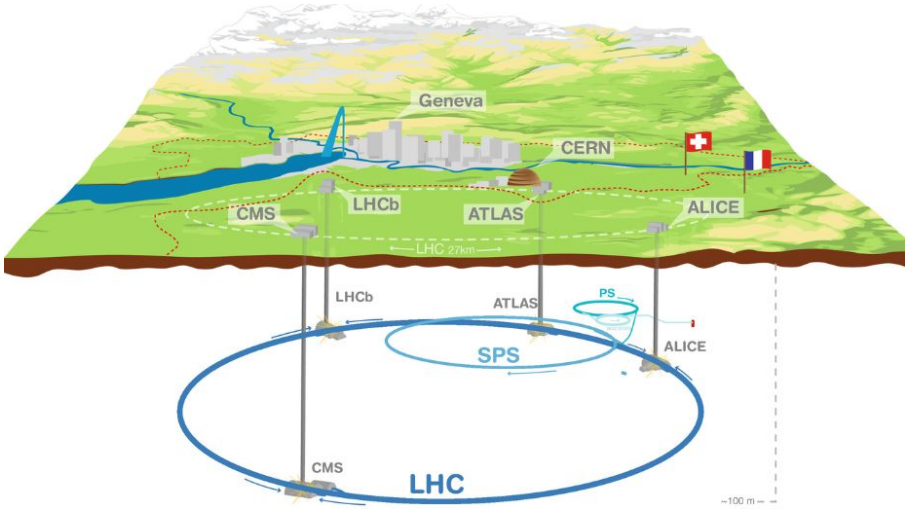


Figure 12: Layout of the LHC accelerator complex, showing the collision points with the ATLAS, CMS, LHCb, and ALICE detectors.

and c quark decays. These decays can proceed via *loop* or *box* diagrams, which could involve new particles. By comparing the measurements of these processes with theoretical predictions, potential discrepancies might be indicative of new physics.

Since the experiment's inception, it has contributed to a great extent to our understanding of the Standard Model. Measurements of b-meson oscillations, CP violation parameters, CKM metrology and rare decays in the heavy sector have been performed. LHCb has also observed a plethora of new hadronic states.

2.2.1 $b\bar{b}$ production at LHC

The production of heavy flavor quark pairs, (b and $\bar{b}$), in high-energy collisions stems from strong interaction processes. The various mechanisms in which $b\bar{b}$ pairs can be produced are:

- Flavor creation through gluon-gluon fusion ($gg \to b\bar{b}$): two gluons from the protons can directly interact, leading to the creation of $b\bar{b}$ pairs as shown in Fig. 13a this is the dominant mechanism for $b\bar{b}$ production at the LHC.

- Flavor creation through quark-antiquark annihilation ($q\bar{q} \to g \to b\bar{b}$): $b\bar{b}$ pairs can be created when a quark from one proton and an an-

tiquark from another proton annihilate, resulting in $b\bar{b}$ production, as shown in Fig. 13b.

- Gluon splitting: gluons can spontaneously split into a $b\bar{b}$ pair, contributing to the overall rate of $b\bar{b}$ production, as shown in Fig. 13c.
- Flavor excitation through gluon-induced processes: in flavor excitation, gluons interact with quarks in the protons, converting some of their energy into $b\bar{b}$ pairs through complex QCD processes, one example of such a process is shown in Fig. 13d.

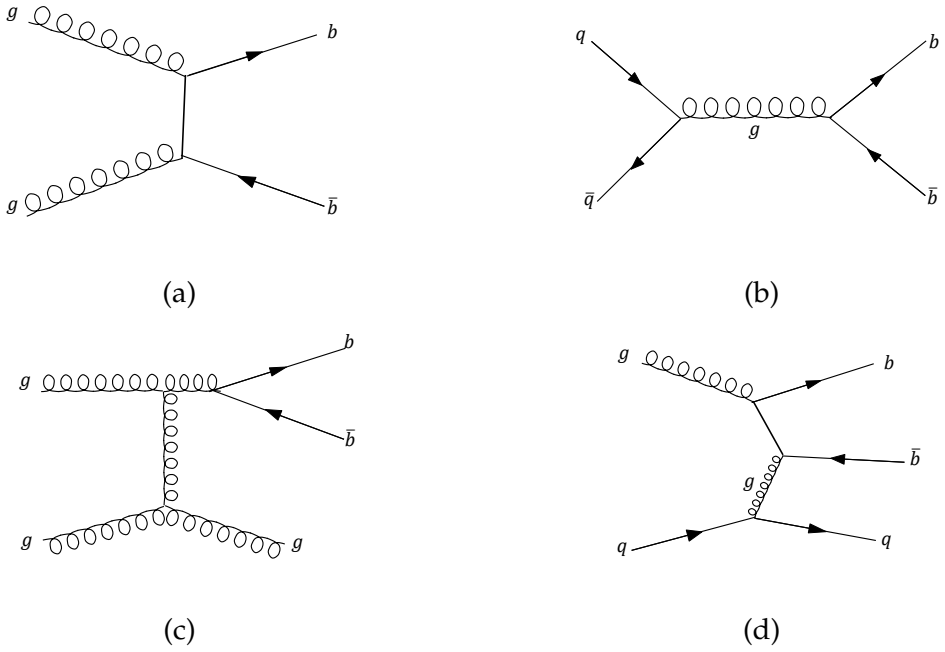


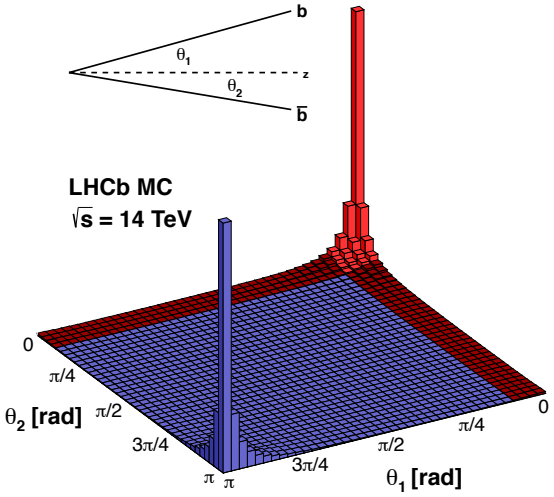
Figure 13: (a) Gluon-gluon fusion. (b) Quark-antiquark annihilation. (c) Gluon splitting. (d) Gluon-induced processes.

In these processes, a significant amount of energy is transferred to the quark-antiquark pair, which typically results in these particles being emitted close to the direction of the initial colliding protons.

2.2.2 Design considerations

Since the quarks' transverse momentum is typically smaller than their longitudinal momentum, the distribution of the produced pairs as a function of the polar angle θ (the angle between the momentum of the particle with respect to the beam axis) peaks in the forward $(cos(\theta) = 1)$

and backward $(cos(\theta) = -1)$ directions. For this reason the LHCb detector is a single-arm spectrometer with a forward angular coverage from approximately 15 mrad to 300 (250) mrad in the $x - z$ $(y - z)$ plane, corresponding to a pseudorapidity[1] range $2 < \eta < 5$. Distribution of b and $\bar{b}$ at $\sqrt{s} = 14\,\text{TeV}$ is shown in Fig. 14. The design of LHCb allows access to 27% of b and $\bar{b}$ quark production. In order to perform precision



(a)

Figure 14: Distribution of b and $\bar{b}$ quarks at LHC. Figure from Ref. [49].

measurements, LHCb operates at a reduced luminosity than the nominal LHC luminosity, with lower *pileup*. This is achieved by defocusing the beam at the LHCb interaction point, having less instantaneous luminosity, of the order of $10^{32}\,\text{cm}^{-2}\text{s}^{-1}$, thereby making each bunch crossing (or event) dominated by few pp interactions. As compared to ATLAS and CMS wherein μ ranged from 25 to 40, μ at LHCb was less than 2 during the Run1 and Run2. The integrated luminosity acquired by LHCb during the Run1, Run2 and Run3 is $3\,\text{fb}^{-1}$, $6\,\text{fb}^{-1}$ and $0.4\,\text{fb}^{-1}$, respectively. At the end of Run3, and thanks to the upgrade of the LHCb detector in the past years, an integrated luminosity of $23\,\text{fb}^{-1}$ is envisaged.

[1]The pseudorapidity η is defined as $\eta = -\ln\tan(\theta/2)$.

2.3 The LHCb detector and subsystems.

The LHCb detector and its subsystems is shown in Fig. 15. A right-handed coordinate system is defined with z-axis aligned with the beam direction, y vertical and x horizontal. LHCb detector is 21 m long in the forward direction (z), 10 m high (y) and 13 m wide (x).

In the following, the detector and its components are described. This is based on the current state of the LHCb detector after the first major upgrade for Run3. The major differences in comparison to the detector configuration during Run1 and Run2 are highlighted in the subsections. The main improvements introduced by the upgrade at detector level, as compared to the previous detector, are the new tracking system and the read out architecture. The new readout permits a software-based trigger system, allowing data to be recorded at a rate five times higher than before. This enhances the acceptance of purely hadronic b decays by up to a factor of two. The subsystems of the detector comprise:

1. **Tracking detectors**: which can detect charged particles and pinpoint the decay vertex. These are: VErtex LOcator (VELO), Upstream Tracker (UT), and Scintillating Fiber Tracker (SciFi).
2. **A dipole magnet**: which bends the path of the charged particle and enables momentum measurements.
3. **Particle Identification (PID) system**: for particle identification, differentiation, and energy measurement. The setup incorporates Ring Imaging Cherenkov detectors (RICH1 and RICH2), muon chambers, and electromagnetic and hadronic calorimeters.

2.3.1 Tracking detectors

The VErtex LOcator (VELO)

The VELO (Vertex Locator) is tasked with detecting charged particles decaying close to the interaction point. Its main function is the accurate reconstruction of primary and secondary vertices with a spatial resolution less than the typical decay lengths of b and c hadrons i.e. $c\tau \approx 0.01\,\text{cm}$–$1\,\text{cm}$. This precision is indispensable for distinguishing between heavy

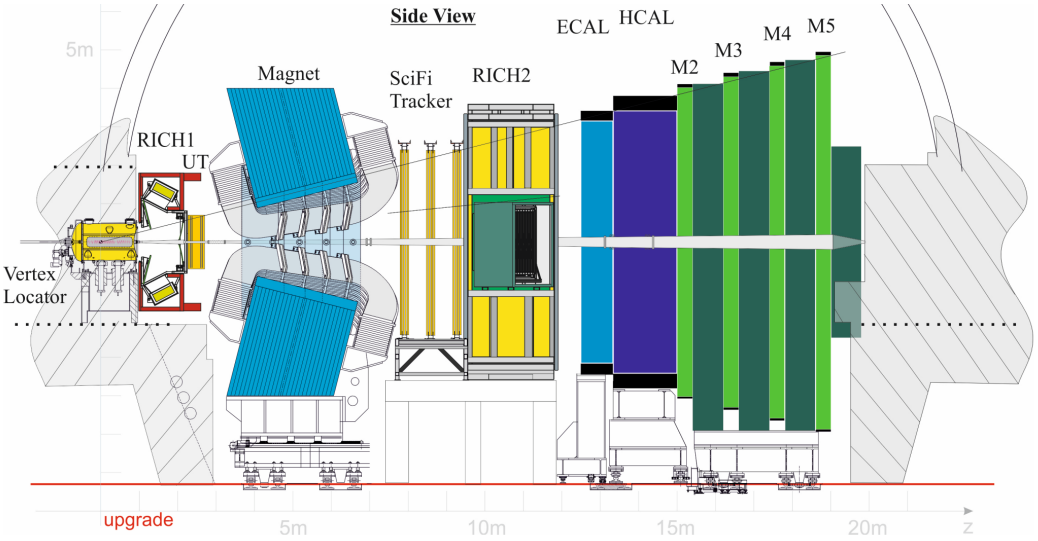


Figure 15: The LHCb detector.

flavor signals from background events. This detector was made of silicon micro-strip layers during Run2, and it has been upgraded to a silicon hybrid pixel technology for Run3. The principal metric for a vertex detector design is the impact parameter resolution, the precision with which the distance of a track to a point is measured. For Run2 it was σ_{IP}=(15 + 29/p$_T$) μm, where p_T is the component of the momentum transverse to the beam, in GeV/c. The VELO design for the upgraded Run3 has been optimised to maintain, within the standard LHCb acceptance, a performance equivalent to or better than the Run2 VELO. This is in terms of both σ_{IP} and track-finding efficiency, even with the anticipated increase in instantaneous luminosity for Run3.

The core technology of the new VELO is pixelated hybrid silicon sensors of 55μm $\times$ 55μm, which are arranged into 52 modules with 4 sensors each, and cooled by a bi-phase CO_2 microchannel system embedded in the silicon substrate. The VELO services have been redesigned reducing both the material budget and the inner radius of the VELO along the beamline. The modules are arranged into two movable halves, the Side C ($x < 0$) and Side A ($x > 0$). Both sides have a common z distribution but C-side modules are shifted 12.5 mm along the z axis to assure mechanical compatibility and sensor overlap when closed, providing a complete azimuthal coverage. Figure 16 shows a sketch of the VELO detector. The major differences between the Run2 and Run3 VELO detectors are

summarised in Table 1.

VELO dertector	2009–2018	2022
RF box inner radius (min. thickness)	5.5 mm (300 μm)	3.5 mm (150 μm)
Inner radius of active si detector	8.2 mm	5.1 mm
Total fluence (silicon tip) [n_{eq}/cm^2]	$4 * 10^{14}$	$\sim 8 * 10^{15}$
Sensor segmentation	$r - \phi$ strips	square pixels
Total active area of Si detectors	0.22 m^2	0.12 m^2
Pitch (strip or pixel)	37–97 μm	55 μm
Technology	n-on-n	n-on-p
Number of modules	42	52
Total number of channels	172 thousand	41 million
Readout rate [MHz]	1, analogue	40, zero suppressed
Whole-VELO data rate	150 Gbit/s	$\sim$ 2 Tbit/s
Total power dissipation (in vacuum)	800 watts	$\sim$ 2 kwatt

Table 1: Specifications of the Run3 VELO compared to the Run2.

Upstream Tracker (UT)

The second tracker in LHCb, named Tracker Turicensis (TT) in Run2 and located right before the magnet, was composed of four stations of silicon-strip detectors. The sensors have been upgraded to increased granularity, radiation hardness and acceptance for Run3, and now it is called Upstream Tracker (UT). It is one of the most important detectors concerning the work in this thesis.

The UT is positioned between the RICH1 detector and the dipole magnet. It plays a critical role in charged-particle tracking [50]. The setup comprises four planes of silicon strip detectors and it is shown in Fig. 17. The planes are referred to as UTaX, UTaU, UTbV and UTbX. The strips in UTaX and UTbX plane are arranged vertically along the y-axis and in UTaU and UTbV are inclined at stereo angles of 5°. The UT provides a preliminary momentum estimation for tracks with $p_T > 0.2$ GeV/c, by using fringe magnetic field between the interaction region and the UT itself. This results in moderate precision ($\sim$ 15%) which allows to expedite the matching with Scintillating Fibre Tracker (SciFi) hits. Furthermore, the UT provides essential information about particles that decay after

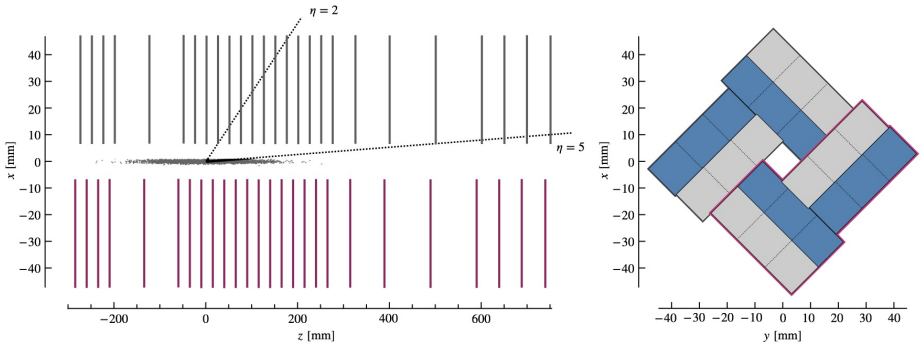


Figure 16: Left: A top-view on the $z - x$ plane at $y = 0$ shows the luminous region z-extent and pseudorapidity acceptance ($2 < \eta < 5$). Right: A sketch presents the ASICs' standard layout around the z-axis in the closed VELO. ASICs are split between the upstream (grey) and downstream (blue) module faces. Side C modules are highlighted in purple in both images.

the VELO such as the long-lived K_S^0 and Λ, contributing to the *downstream* tracking described in Chapter 5. To meet occupancy requirements

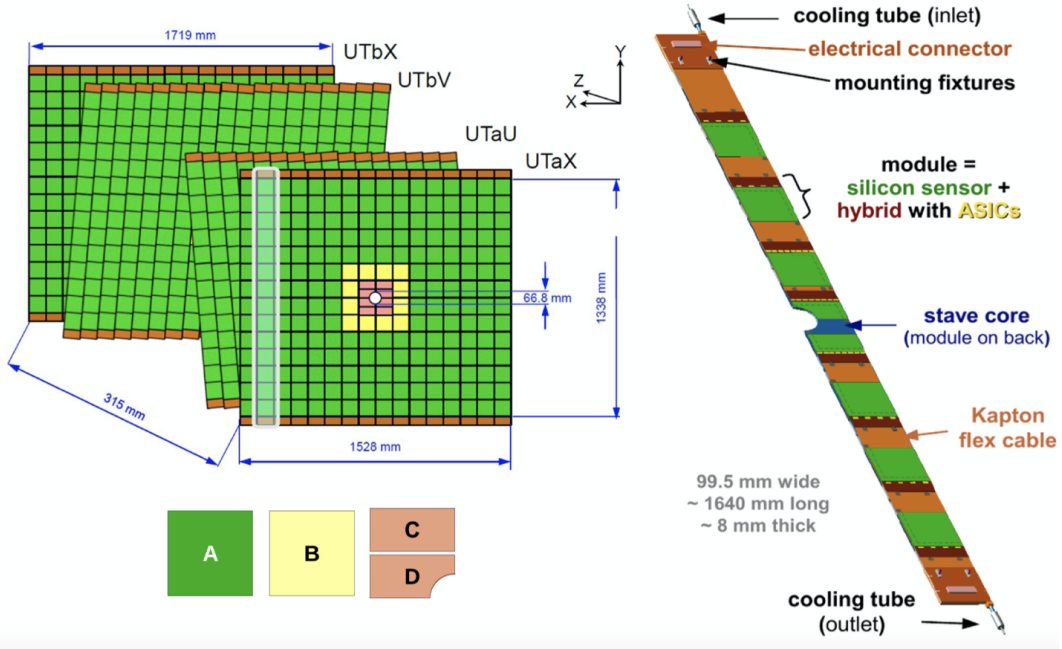


Figure 17: The configuration of the UT's four layers is illustrated. Distinct colors represent the various sensor types. On the right, the design of an UT stave is displayed. Figure from Ref. [51].

in different regions of the detector, four sensor types (named A, B, C, and D) with varying strip pitch and strip lengths are used. The sensors

are arranged in vertical columns called *staves*, a mechanical structure composed of 14 sensors with a hybrid PCB[2] that provides electrical connection. The internal area required a finer granularity due to the higher occupancy, therefore, the four central sensor positions are populated with a combination of C and D sensors, while the 12 sensors around them are of type B, as illustrated in Fig. 17.

The UT also includes several improvements in terms of mechanics and services. A crucial improvement for Run3 is the rapid data processing by the front-end (FE) electronics. This is done through an intelligent iterator which returns the strip with the highest pulse-height after checking the neighboring strip hits.

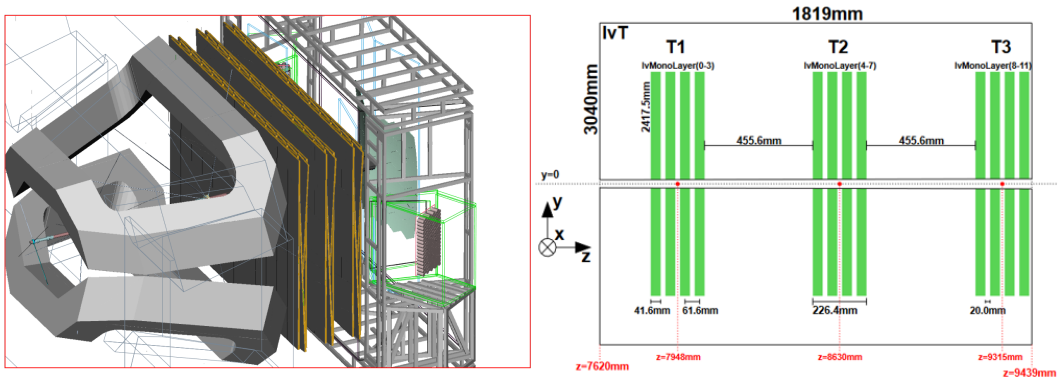
Scintillating Fibre Tracker (SciFi)

The Scintillating Fibre Tracker (SciFi), placed downstream of the LHCb dipole magnet as shown in Fig. 18a, is the most innovative detector of the LHCb upgrade. It uses a cutting-edge technology of long scintillating fibres. The detector is tasked with charged particle tracking and momentum estimation, and together with the UT, it is one of the most relevant detectors in the work of this thesis. It is required to provide momentum resolution and tracking efficiency for b- and c-hadrons comparable to those obtained during Run1 and Run2. To meet the nominal LHCb acceptance, the tracker has to cover an area of roughly $6\,\mathrm{m} \times 5\,\mathrm{m}$ in the xy plane.

The acceptance of the new *SciFi* detector ranges from approximately 20 mm from the beam pipe edge to distances of ± 3186 mm and ± 2425 mm in the horizontal and vertical directions, respectively. The detector relies on 250 μm diameter plastic scintillating fibres in multi-layered fibre mats, arranged in 12 detection planes across 3 stations (named T1, T2, T3) with four layers in an *x-u-v-x* configuration, for determining the particle trajectories in x and y coordinates.

The stations are built from identical SciFi modules that are approximately 52 cm wide, spanning the full height. Figure 18b shows an sketch of the different stations. Each station has four layers with each layer

[2]Printed Circuit Board.



(a) (b)

Figure 18: (a) The three stations of the SciFi shown between the dipole magnet on the left and RICH2 on the right. (b) Sketch of the SciFi detector showing the three stations and the four layers in each station.

having two independently movable structures, referred to as C-frames, on either side of the beam pipe. The scintillating fibres light signals are detected by 128-channel arrays of silicon photomultipliers (SiPM) with a channel pitch of 250 µm.

The equivalent Run2 detector was divided in two subsystems, the Outer Tracker (OT) and the Inner Tracker (IT). The OT, constructed using large straw detectors, occupied around 99% of the 30 m^2 detector surface. The IT, a silicon micro-strip detector, covered an area of 0.35 m^2, specifically focusing on the region around the beam-pipe with high track density. Each T-station comprised four detection planes (x, u, v, x), which provided precise coordinate measurements with strips or straws oriented at (0°, +5°, −5°, 0°) relative to the vertical axis.

For the Run2 tracking system, the momentum resolution, $\Delta p/p$, ranged from 0.4% at 5 GeV/c to 0.6% at 100 GeV/c, with an invariant B mass resolution of approximately 8 MeV/c^2. The performance of the tracking system for Run3 is explored in detail in Chapters 4 and 5.

2.3.2 Magnet

A dipole magnet, located between the UT and SciFi, induces a deviation in particles trajectory in the horizontal plane from which it is possible to extract their momentum. The integrated magnetic field between the

UT and SciFi is 4 Tm. The polarity of the magnet is reversed periodically to minimize possible systematic uncertainties due to the detector asymmetries, which average out by using each half of the data in a different configuration (i.e magnet polarity up and down). Fig. 19 shows the magnetic field distribution.

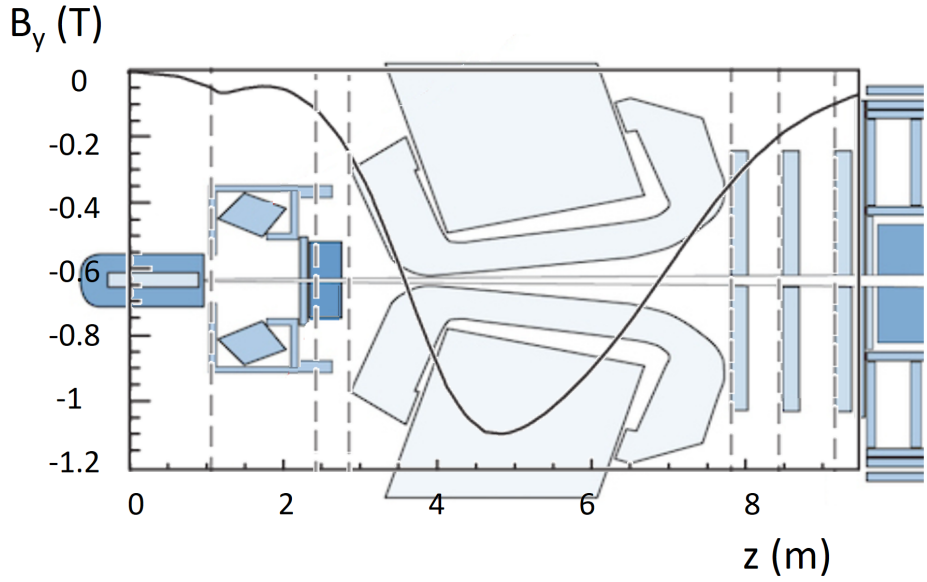


Figure 19: Distribution of the magnetic field as function of z created by the LHCb magnet. The VELO, RICH1, UT, magnet and the SciFi are superimposed.

2.3.3 Particle Identification

The RICH detectors

The discrimination of charged hadrons, in particular the distinction between pions, kaons, and protons, is an essential requirement for the LHCb physics programme. The LHCb utilizes the RICH (Ring Imaging Cherenkov) system for hadron Particle Identification (PID) in the 2.6 - 100 GeV/c momentum range.

Despite retaining the overall concept and layout of the RICH system from Run1 and Run2 [52], essential modifications have been needed to enable the system to function at a higher design luminosity, maintaining a performance on par with that of the previous runs [53]. The system is composed of two detectors, RICH1 and RICH2, as depicted in Fig. 20, which cover different particle momentum ranges.

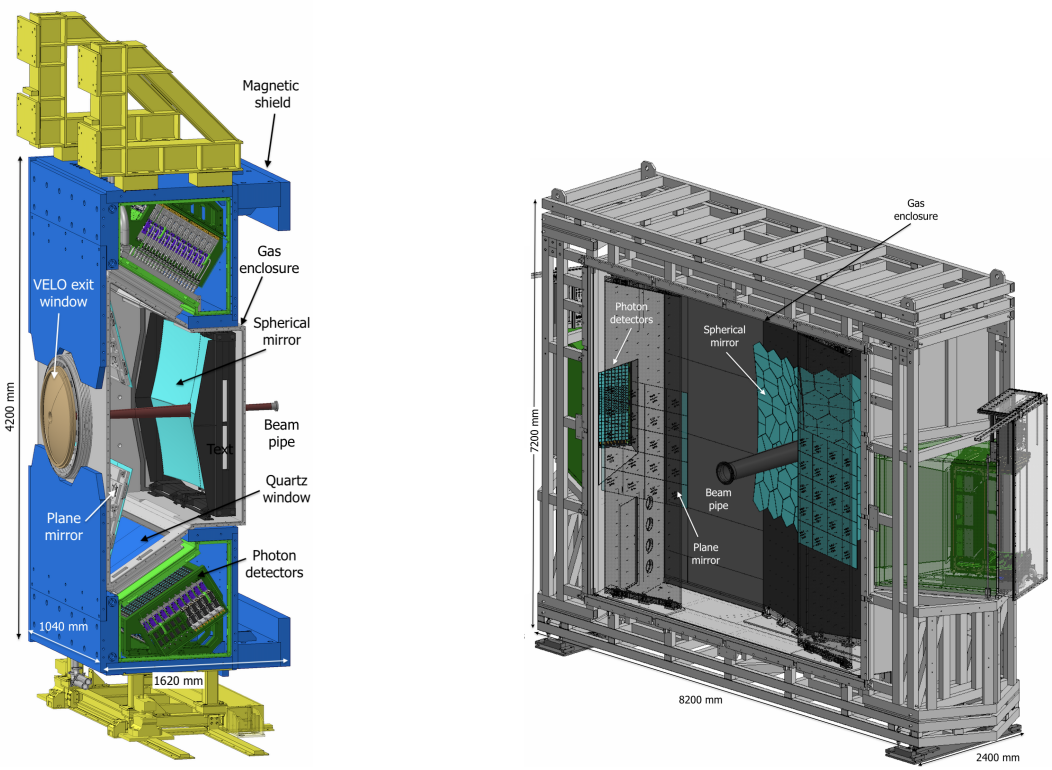


Figure 20: Schematic view of the Run3 (left) RICH1 and (right) RICH2 detectors.

- **RICH1**: the upgraded detector, positioned upstream of the LHCb dipole magnet between the VELO and the UT, has undergone substantial redesign and subsequent rebuilding. The geometry and the optical system has been upgraded to reduce the number of photons in the hottest inner region. The Run2 design has been modified by increasing the focal length of the spherical mirrors, reducing mirror aberrations, and moving the photon detector planes. The detector kept the fluorocarbon C4F10 gas radiator from the previous design, appropriate for momentum ranges between 2 GeV/c and 60 GeV/c, and the angular acceptance remains unchanged, covering 25 mrad - 300 mrad.

- **RICH2**: situated downstream of the dipole magnet right after the SciFi, it covers an angular acceptance of 15 to 20 mrad. Filled with CF4, it covers the 30 GeV/c - 100 GeV/c momentum range. The system remains similar to the one used in Run2, but a new opto-electronics chain and a new a mechanic framework to support it have been developed, for both RICH2 and RICH1.

Calorimeters

The LHCb calorimeter system comprises an Electromagnetic calorimeter (ECAL) followed by a Hadronic one (HCAL). They are in charge of detecting and identifying photons, electrons and hadrons. During the Run1 and Run2 these detectors were key since they participated in the hardware trigger. For Run3 the configuration of both calorimeters has been retained[3], and the main changes comes from the FE electronics boards, which have undergone a complete redesign to meet the 30 MHz readout frequency. Figure 21 shows the ECAL and HCAL layouts.

- **Electromagnetic Calorimeter (ECAL)**: the ECAL is responsible for measuring the energy of photons and electrons. It is situated approximately 12.5 m from the interaction point and is formed by scintillating pads (4mm) separated by thick lead absorbers (2mm).

[3]The calorimeter modules, photomultiplier tubes, bases, and coaxial cables have remained unchanged.

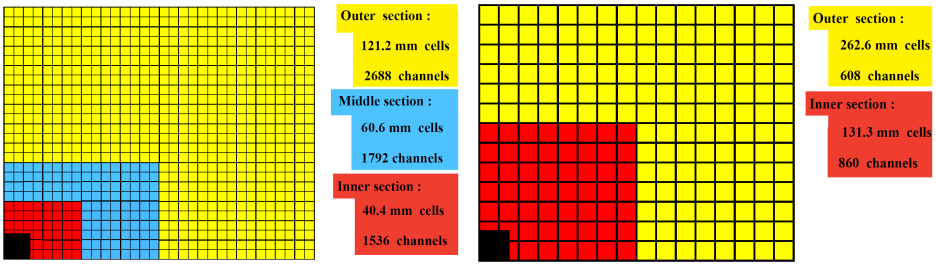


Figure 21: Lateral segmentation of the ECAL (left) and HCAL (right). A quarter of the detector front face is shown.

It is read out by dedicated photomultiplier tubes (PMTs). It consists of a total of 6016 square cells arranged in inner, middle, and outer regions, with cell sizes of 40.4 mm, 60.6 mm, and 121.2 mm sides, respectively. These sizes are designed to scale with the distance from the beam-pipe to maintain a roughly uniform particle rate per cell. The energy resolution of an individual cell, as measured with a test electron beam, is parametrised as:

$$\frac{\sigma(E)}{E} = \frac{(9.0 \pm 0.5)\,\%}{\sqrt{E}} \oplus (0.8 \pm 0.2)\,\% \oplus \frac{0.003}{E \sin\theta} \tag{2.3}$$

where E is the particle energy in GeV, and θ is the angle between the beam axis and the line from the LHCb interaction point to the centre of the ECAL cell.

For Run2, two additional subsystems were in place, the SPD (Scintillating Plane Detector) and the PS (PreShower Detector), and used to initiate the electromagnetic shower, allowing to recognise electrons, photons and neutral pions. Those systems have been removed since they are not needed anymore for the hardware trigger.

- **Hadronic Calorimeter (HCAL)**: the HCAL is a sampling tile calorimeter with a thickness of 5.6 interaction lengths. The sampling structure consists of staggered iron and plastic scintillator tiles, which are mounted parallel to the beam axis to enhance light collection. The HCAL employs the same PMT type as the ECAL for readout. It comprises 1488 cells, organised into an inner and outer region, with square cells of 131.3 mm and 262.6 mm sides, respectively. The energy resolution of the HCAL, as measured in beam tests

with pions, is parametrised as:

$$\frac{\sigma(E)}{E} = \frac{(67 \pm 5)\,\%}{\sqrt{E}} \oplus (9 \pm 2)\,\% \tag{2.4}$$

where E is the energy deposited in GeV.

Muon system

The muon subdetector of LHCb consists of 1104 multi-wire proportional chambers (MWPC) across four stations (M2 to M5), covering an area of 385 m^2. For Run3, an earlier station serving for the hardware trigger, M1, is no longer in use, but its structure now serves as neutron shielding. Each station is divided mechanically into two halves, houses MWPCs and is located downstream of the calorimeter system. These stations are also layered with thick iron absorbers for filtering low-energy particles. Figure 22 shows a schematic of the muon system layout. The read-out electronics have been overhauled to synchronize with the upgraded LHCb readout scheme, the primary upgrade of the muon system.

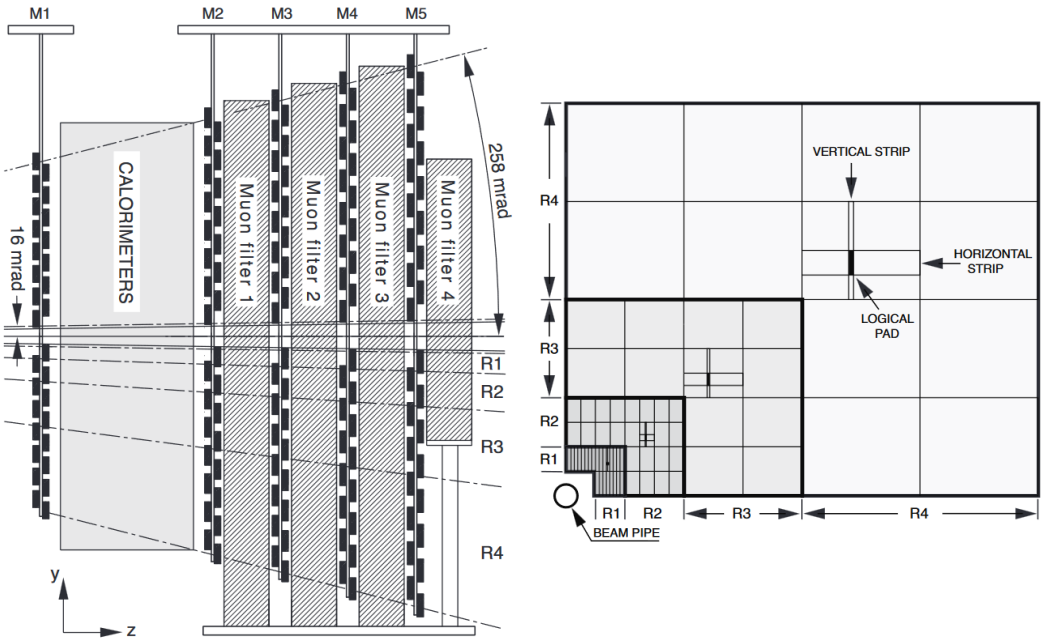


Figure 22: Schematic of the LHCb muon system (a) side view (b) front view of one quadrant of M2 [54].

2.4 The Data Acquisition System (DAQ) and Online

During Run3, LHCb is beginning to process data at an event rate of 30 MHz, where each event approximately consists of 100 kB of data. This corresponds to an expected total throughput of 40 Tb per second.

The Data Acquisition (DAQ) system of LHCb comprises the readout system, the event builder, and the event filter. A schematic overview of the system is shown in Fig. 23. Data are read from the front-end electronics of the detector and fed into about 170 event builder (EB) servers. Each server is a standard 4U rack unit type with 8 PCIe slots to host the TELL40 back-end (BE) receiver boards, network cards and Graphical Processing Units (GPUs). Using a synchronised round-robin scheme through the EB network, each server sends the individual event fragments to one server for the building. Following this, the first stage of Event filtering is carried out on GPUs which run the High Level Trigger stage one (HLT1) application and are integrated within the same EB servers. Data processed by the EB and HLT1 are then dispatched to a buffer storage. From this location, data is retrieved by the second stage of the High Level Trigger two (HLT2) for further processing. The HLT2 stage comprises up to 3000 servers of half width standard rack unit.

2.4.1 Event readout

LHCb uses a synchronous readout. After every bunch-crossing, all data acquisition components, known as front-end electronics (FEE), transmit data after zero-suppression. So, the LHCb online system essentially operates without a traditional trigger from the FEE's viewpoint.

In the LHCb's online system, readout elements form partitions. These range from parts of a subdetector to groups of them. These partitions can operate concurrently, aiding in commissioning and testing, with the Timing and Fast Control (TFC) and Experiment Control System (ECS) enabling this (see Fig 23).

Data travel from the underground detector via multi-mode optical fibers to a ground data center, using the radiation-hard Versatile Link

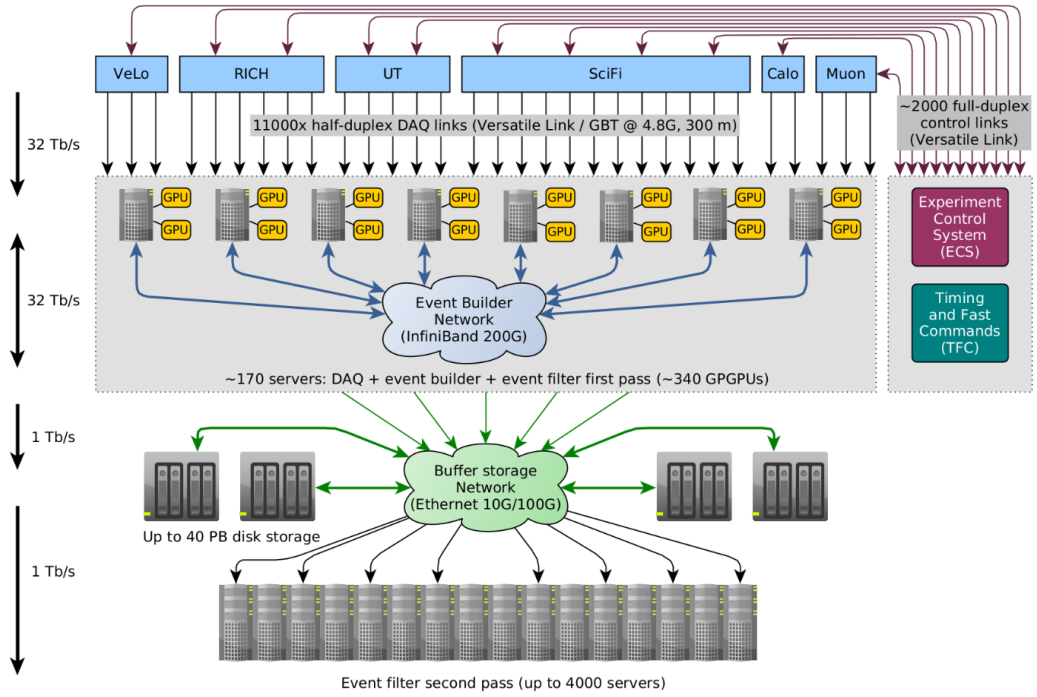


Figure 23: The LHCb online computing infrastructure and DAQ layout.

(VL) and the GigaBit Transceiver (GBT) protocol. LHCb uniquely uses half-duplex mode for data transmission links, with a single fiber connecting the FE to TELL40 boards. Separate connections handle control and monitoring. Control links, both FE and BE, operate in full-duplex. While both TFC and ECS are sent to the FE, only ECS returns via the same optical links.

2.4.2 Event building (EB)

For event selection, the fully software-based LHCb trigger requires full event information from all subdetectors. Hence, event building, which is the consolidation of all data segments belonging to the same bunch-crossing, is performed for every non-empty bunch collision at a 30 MHz rate[4].

The EB system receives data from the subdetectors. Given that each EB server only collects data from the subdetectors linked to the corresponding TELL40 boards, a high-performance network is required to

[4]Data associated with each bunch filling is synchronously transferred to the TELL40 boards by the Timing and Fast Control (TFC) system. Normally, data recorded during empty crossings are disregarded.

interconnect all the EB nodes and facilitate the transmission of complete information to the node responsible for full event assembly. The EB network employs 200 Gbps High Data Rate (HDR) InfiniBand technology with PCIe interfaces to ensure optimal data bandwidth.

There are three TELL40 cards per server. Each server is linked through two HDR ports to the EB network. Every server alternately serves as a data source and data sink in the event-building process, where cyclically each node operates as a complete event builder (sink) and collects data from all other servers (sources). For optimal performance, data from multiple bunch-crossings are bundled together and managed as unit data blocks. Once a builder receives data from all sources, it reorders them for efficient subsequent processing. Finished events are stored in a memory buffer and passed to the event filtering process.

2.4.3 Event filtering (EF)

For Run3, the event filtering is implemented in two stages of software based triggers: High Level Trigger 1 (HLT1) and 2 (HLT2). The HLT1 is run on the GPUs installed within the EB nodes. This is discussed in detail in Chapter 3. There is one PCIe based Nvidia A500 GPU installed on each of the approximately 170 event builder nodes. The events selected by HLT1 are then dispatched via a separate 10G/100G Ethernet network to temporary storage, from where they are accessed by the alignment and calibration processes and the second-stage filtering by HLT2.

2.5 LHCb software framework

The LHCb software framework and its various components have been organised under two major categories: Real Time Analysis (RTA) and Data Processing & Analysis (DPA). Figures 25 and 27 provide overviews of the RTA and DPA process flows, respectively. In the context of this thesis, contributions have been made to the software under RTA. These contributions are elaborated upon in Chapters 4, 3, 5 and 6. Addi-

tionally, Sec. 2.5.2 provides a brief description of the various software packages and tools from DPA used in this thesis.

2.5.1 RTA: the trigger system

During Run2, the trigger involved a hardware trigger, which harnessed the calorimeters and the muon system to lower the event rate from 30 MHz to approximately 1 MHz. After the hardware trigger, a two-stage software trigger was implemented on a dedicated CPU farm. The initial stage, referred to as HLT1, incorporated a partial event reconstruction strategy to further decrease the event rate. In Run2, this rate was effectively around 100 kHz. The subsequent stage, HLT2, performed a detailed event reconstruction and determined which events to retain. Throughout Run2, HLT2 could select events at the rate of 12.5 kHz.

At Run3, with a luminosity of $\mathcal{L}_{inst} = 2 \times 10^{33} cm^{-2} s^{-1}$, the hardware L0 trigger becomes inefficient. The trigger yield saturates on the hadronic modes as shown in Fig. 24, not being able to select efficiently the signals from b and c decays. Thus the approach of the hardware trigger system of LHCb Run2 had to be overhauled in order to meet the physics requirements of Run3.

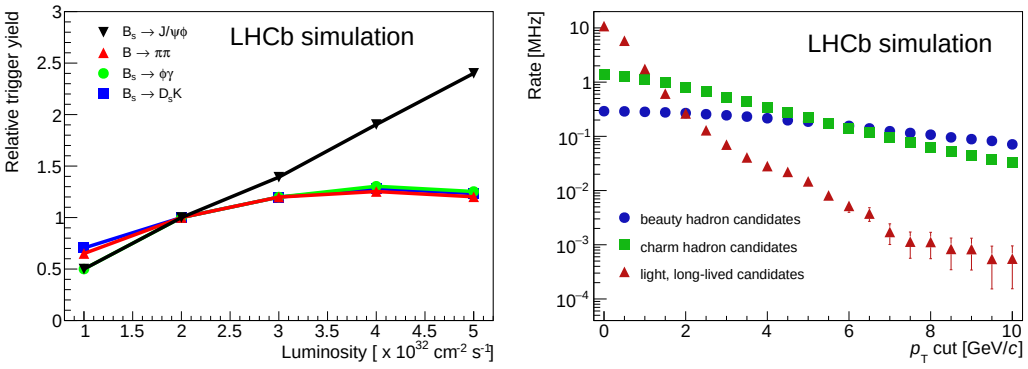


Figure 24: Left: relative trigger yields as a function of instantaneous luminosity, normalised to $\mathcal{L}_{inst} = 2 \times 10^{32} cm^{-2} s^{-1}$. Right: rate of decays reconstructed in the LHCb acceptance as a function of the cut in p_{T} of the decaying particle, for decay time $\tau > 0.2\,\mathrm{ps}$.

The new trigger system has been designed to be entirely software-based which enables real-time reconstruction of all events at a visible interaction rate of 30 MHz. It effectively reduces data volume from 4 TB/s,

to approximately 10 GB/s suitable for permanent offline storage. The complexity of this 400 times reduction, lies in the high rate of signals of potential interest that can be at least partially reconstructed within the detector acceptance.

To achieve this, the trigger must fully reconstruct and pinpoint specific signals of interest, retaining only a select portion of the event's information [55]. This necessitates that the trigger executes an offline-quality reconstruction, achieved by a near real-time alignment and calibration of the detector. LHCb real-time analysis approach, initiated in Run2 [56], underscores this demand [57].

The HLT1

The HLT1 serves as the first filter in the LHCb trigger system, as depicted in Fig. 25. Its design principle is rooted in leveraging track reconstruction signatures and selections from tracking subdetectors. The primary goal of HLT1 is to curtail event rates to manageable levels, allowing data to be buffered for subsequent HLT2 processing.

The event signal rates are predominantly driven by c-hadrons at nearly 1 MHz and b-hadrons at 300 kHz, both of which can be partially reconstructed within the LHCb's acceptance parameters. Additionally, The LHCb physics program, encompassing electroweak physics, quarkonia, semileptonic and rare heavy decays, among others, significantly influence the event rate. The HLT1 design ensures its output rate do not theoretically surpass 2 MHz. Yet, practical constraints, such as the disk buffer capacity and HLT2 processing speeds, dictate this rate, as visualised in Fig. 25. A more detailed exploration of this design is provided in Chapter 3.

The HLT2

The primary function of HLT2 is to carry out the full offline-quality reconstruction and selection of physics signatures. A sizable disk buffer is placed between HLT1 and HLT2 to serve as a temporary repository for data while real-time alignment and calibration are conducted. At HLT2, the fate of an event, whether it should be conserved, is determined

by roughly $\mathcal{O}(1000)$ selection algorithms each individually tuned for a distinct signal topology or physics analysis. These algorithms also decide which components of the full event are to be committed to permanent storage.

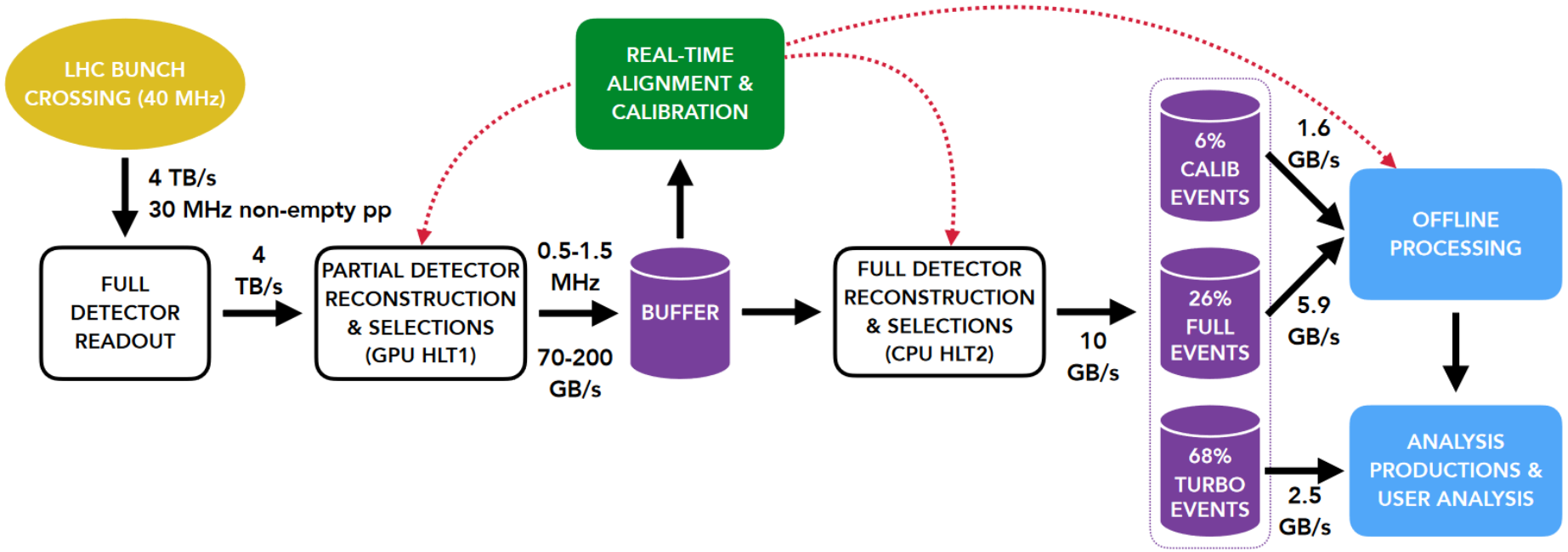


Figure 25: RTA Data flow in Run3, figure from Ref. [58]

The reconstruction pipeline comprises four core components: charged particle pattern recognition, calorimeter reconstruction, particle identification, and a Kalman fit for tracking. To achieve the required computing throughput, tracks, neutral objects, and particle identification data are grouped into Structure-of-Array (SoA) data structures, enabling efficient parallel processing. Selection algorithms employ both rectangular-cut-based methods and techniques rooted in multivariate analysis or artificial intelligence.

To manage and optimize the data volume, HLT2 selections are grouped into *streams*, namely, *Full*, *Turbo* and *Calib* as depicted in Fig. 25. Each stream is tailored to specific physics channels and can be adjusted as the experiment progresses. If multiple algorithms flag an event, the union of their requested data is saved. Within these streams, The *Full* data stream persists full event data along with the reconstructed objects when an event is triggered by any of the selection lines. This contains about 26% of the total events triggered while occupying about 60% of the total output bandwidth.

The *Turbo* stream, detailed in Refs. [55, 56], offers flexibility since it allows selective persistence of the data, i.e. it allows to choose which part of the event to keep along with the reconstructed objects. Depending on the physics channel under scrutiny, data storage can range from basic objects, like two tracks for a two-body decay, to comprehensive event details, as outlined in Ref. [59]. This consists of around 68% of the total events triggered while occupying 25% of the total output bandwidth.

The *Calib* stream, is used for processing the data for real-time alignment and calibration of the detector and accounts for about 6% of the total events processed occupying 15% of the bandwidth.

Figure 26 highlights the complexity of LHCb's trigger system in view of other similar particle physics experiments. It shows the distribution of hardware trigger rates vs the event size for different experiments.

2.5.2 DPA: Data Processing and Analysis

Software infrastructure for offline data processing and analysis is spread across different projects. These cover core software, conditions database,

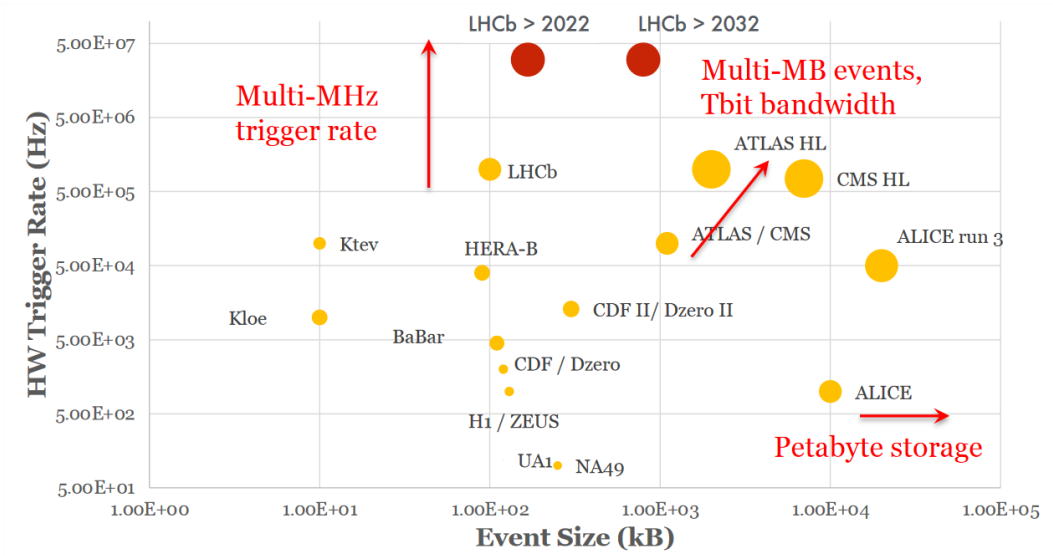


Figure 26: Trigger rates comparison of different particle physics experiments. Figure by A. Cerri (LHCP-2022).

detector geometry description, software infrastructure, simulation, offline data processing and analysis, distributed computing, and data management.

Core software

The Gaudi core software framework [60] is the base application for most of LHCb software applications. It offers a unified environment for the simulation, filtering, reconstruction, and analysis of data. Central to Gaudi design is its modularity, allowing components such as algorithms, services, and tools to be seamlessly integrated to build applications. It features a flexible event data model that structures the storage and retrieval of data objects processed during an event.

Conditions Database

The Conditions Database holds information essential for data processing, such as detector configurations, alignment or calibration constants, and environmental parameters. These conditions may be specific to individual detector elements and are defined in a three-dimensional structure based on geographical location, time evolution, and versioning. The Conditions Database supports various use cases including simulation and real

data processing.

Detector Geometry Description

The Detector Geometry Description is essential for various tasks ranging from simulation to alignment, visualisation, and material budget computation. It allows integration with the conditions database and ensure efficient navigation between detector elements. To meet these requirements, LHCb adopted the Detector Description for HEP (DD4HEP) toolkit [61].

Offline data processing and analysis

The LHCb Offline data processing flow, as shown in Fig. 27 has the following main aspects: sprucing, distributed analysis productions, and offline analysis.

Sprucing

The sprucing code base shares applications, algorithms, and tools with HLT2 and DaVinci. Sprucing serves three core functions:

1. **Data Skimming:** applies further selections to data saved in the *full* stream. Events selected by topological HLT2 trigger selections, for instance, undergo further processing and selections to reduce the volume of data saved to disk.
2. **Data Slimming:** enables tuning the amount of event information persisted in the output files through the Turbo mechanism. This helps reducing the size of events saved on disk for further physics analysis.
3. **Streaming and FSR Creation:** it streams data into various physics files and creates File Summary Records (FSR) that stores metadata about the file content in output Root files. This can be combined with skimming and slimming as needed.

Distributed Analysis Productions

The output of the Turbo stream and sprucing is split into multiple streams accessible to analysts. A new strategy for analysis data processing has been developed in run3 to deal with the large data volume.

This strategy is based on centralised analysis productions, which are an extension of the LHCbDirac 2.5.3 transformation system.

Offline Analysis

DaVinci is the software application used for physics analysis. It is used to further process the reconstructed data to produce particle candidates, decays, and other high level physics objects. It provides flexible environment to apply selection criteria, perform fits, and analyse the final state particles, providing integration with other software tools such as ROOT [62].

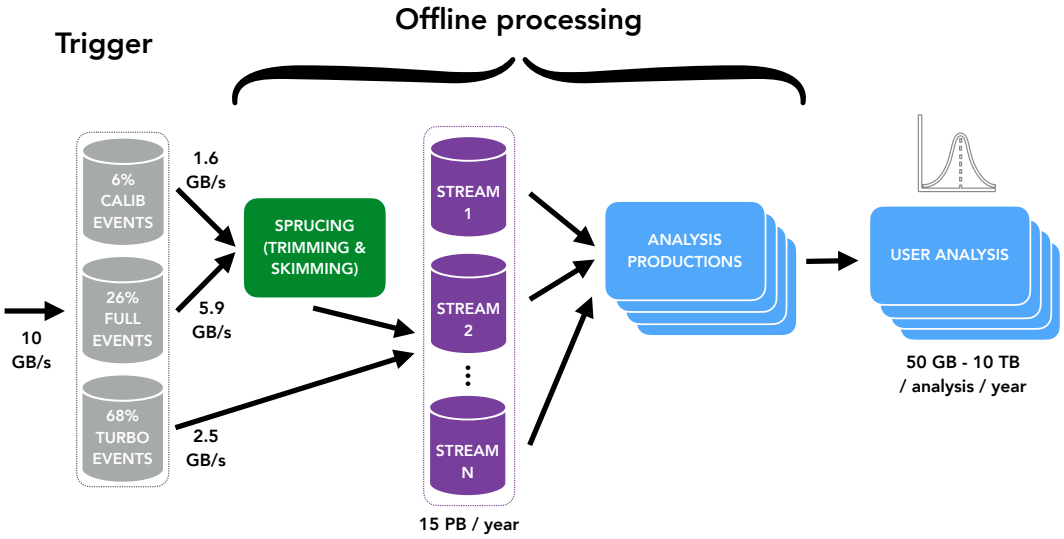


Figure 27: Offline data flow showing the output from different streams going through the sprucing step which utilises analysis Productions for tupling the data for offline user analyses. Figure from Ref. [58].

LHCb simulation framework

The LHCb simulation framework is designed to emulate the detector's response, as it is needed for all physics channels. Within the scope of this thesis, the author served as the MC simulation liaison for the Radiative Working Group. This role involved contributions to the preparation and management of the simulation requests, as well as conducting validation studies for various versions of the simulation software.

The Monte Carlo method is employed in two key phases. The first phase involves event generation, where various physical processes are

modeled using stochastic techniques. Specific generators like Pythia8 [63] and EvtGen [64] are used to describe particle decays and other high-energy phenomena. This phase emulates the underlying physics to produce realistic scenarios that might occur in the detector. The second phase is the detector simulation. Tools like Geant4 [65] are utilised to simulate how the generated particles traverse and interact with the LHCb detector's material, resulting in energy deposits or hits.

The resulting simulated data is then processed to recreate the detector's readout, undergoing digitisation to transform hits into recognizable signals. This includes modeling the imperfections in the detector, such as noise and dead channels. The final simulated data is consistent with the format produced by the real detector's Data Acquisition (DAQ) system.

Optimisation techniques are applied to balance the computational demands of the Monte Carlo method with the available resources. Options for fast and ultra-fast simulations are explored using methods like parametrisation, resampling, and machine learning models.

Figure 28 shows different components of simulation framework and their interplay with the data flow and other applications.

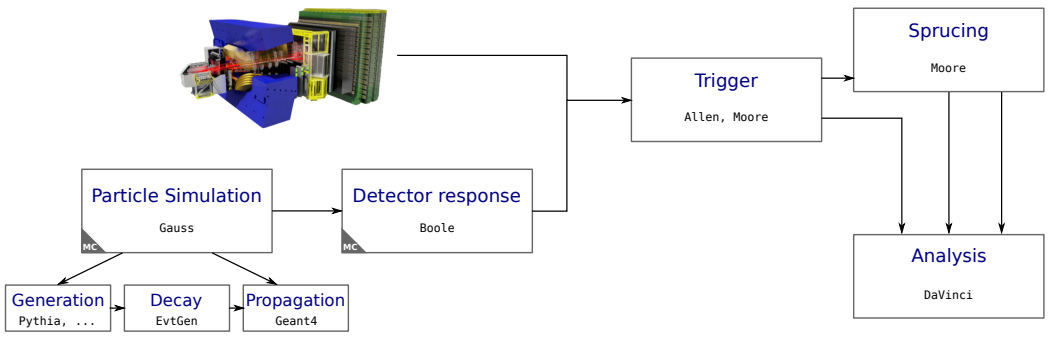


Figure 28: Schematic representation of the LHCb upgrade data flow and the related LHCb application, with an emphasis on simulation. Figure from Ref. [66].

Gauss: the Gauss package is responsible for the following functionalities.

- **Event generation & simulation**: simulating particle interactions, utilizing the Gaudi core software framework, supporting parallelisation and multithreading.

- **Phases**: includes a generation phase with various generators, and a simulation phase relying on Geant4 or custom parameterisations.
- **Gaussino**: an experiment-independent framework, encapsulating core features that Gauss builds upon, while providing interfaces to widely used packages like Geant4.

Boole: the Boole package is used for:

- **Detector and readout electronics modeling**: converting hits into specific subdetector signals.
- **Digitisation**: translating hits into a format compatible with Data Acquisition (DAQ).

Simulations types: different types of simulations are employed for various needs:

- **Filtered simulation**: selecting events based on specific criteria.
- **Fast simulation**: utilizing techniques like resampling for quicker execution.
- **Ultra-fast simulation (Lamarr)**: utilizes machine learning models to simulate the entire detector response, producing analysis-level variables with reduced computational time.

2.5.3 Distributed computing and Worldwide LHC Computing Grid (WLCG)

The WLCG [67] is an essential international computing infrastructure that accommodates the massive data requirements of the LHC experiments, including LHCb. It enables data processing, storage, and dissemination across global computing centres of collaborating institutes. The integration of international grid infrastructures and services, allows for resource sharing and collaborative science. The WLCG's tiered structure includes Tier-0 at CERN, which is responsible for initial data processing and archiving. Tier-1 centers, located in various countries, handle reprocessing, storage, and distribution. Tier-2 centers provide resources for simulation and user analysis, while Tier-3 centers support specific institutional or community needs.

Dirac interware

Dirac [68] is a middleware developed to build and operate distributed computing systems. It provides an array of services for both workload and data management tasks. Its core design allows for horizontal extension, facilitating the addition of independently versioned yet interdependent projects. It includes components such as WebAppDirac, RESTDirac, VMDirac, and COMDirac, each serving a distinct function within the overall system.

Additionally, Dirac offers vertical extensibility, enabling users and virtual organisations (VOs) to enhance the functionalities of the base projects, tailored to specific needs. For LHCb, this has led to the creation of new systems such as Bookkeeping and Production Management via LHCbDirac extensions. Dirac's compatibility with various batch systems, such as HTCondor, facilitates dynamic resource distribution and task scheduling. The workload management system (WMS) and the Dirac data management system (DMS) are responsible for managing Dirac jobs, complemented by a bookkeeping tool for efficient dataset retrieval.

3 LHCb HLT1 for Run 3

This chapter is devoted to a description of the HLT1 framework of LHCb, which is based on the Allen project. The Allen project serves as a GPU-based solution for the first stage of the upgraded LHCb detector High-Level Trigger. After an introduction, the design principles and features of the framework are detailed in Sec. 3.2.3. A pivotal aspect of any software project, its portability across different hardware platforms, is discussed in Sec. 3.3, where the contributions made in this thesis are also elaborated upon.

3.1 Introduction

In order to selectively process events with desired physics signals and exclude the rest, the HLT1 system must process input data, creating the following physical signatures that enable real-time decision-making:

1. Reconstruction of various types of tracks and vertices in different tracking subdetectors, covered in detail in Chapter 4.

2. Reconstruction of leptons, specifically muons and electrons.

To utilise these reconstructed signatures, a broad range of selection lines are defined to identify the desired physics signal. For instance, a total of 62 selection lines were defined at Run2 at the HLT1 level.

These selections, or trigger lines, are processed in parallel, with the output of each line, or group of lines, stored in a separate output stream. These output streams are then merged to produce the final output of the HLT1 system. Decision reports attached to each selected event detail

which trigger lines accepted the event. These reports are also saved alongside the triggered events. The integration of the HLT1 system within the framework of the online DAQ system is detailed in Chapter 2, Sec. 2.4 [69].

3.2 The Allen Framework

With requirements outlined in the previous section, Design principles and features of the Allen software framework which forms the core of the HLT1 system for Run3 are discussed below.

3.2.1 Heterogeneous architectures

Heterogeneous computing architectures have progressively became prevalent in computing systems. Use of GPUs for general purpose computing, commonly knows as GPGPU programming model, has accelerated the adoption of accelerators in today's computing and software ecosystem. The Allen framework is designed to exploit the parallel processing capabilities of GPUs to perform the computationally intensive tasks of the HLT1 system. The events generated at each proton-proton collision are independent and thus, can be processed in parallel. The full data stream from all detectors is read out, with each raw event data having a relatively small size of around 100 kB. This small size facilitates efficient on-the-fly transfer of data to GPU memory, thus making it suitable for GPU processing.

Introduction to GPUs

A modern GPU, or graphics processing unit, is a highly parallelised processor designed for efficient execution of many small, simple calculations simultaneously in single event multiple data fashion. Some key features of modern GPU which has to be kept in mind while designing the software framework are:

- Streaming multiprocessors (SMs): SMs are the building blocks of a modern GPU, each containing multiple processing cores, register

files, shared memory, and cache. A GPU can have multiple SMs, with each SM capable of executing multiple instructions in parallel.

- Memory subsystem: a GPU has a separate memory subsystem optimised for fast data access and high bandwidth. This includes registers, caches, shared memory as well as global memory.

3.2.2 Programming model

Allen is a software framework designed to run on GPUs and is developed using CUDA/C++ [70] programming model. CUDA is a parallel computing platform and programming model developed by Nvidia for general computing on GPUs. CUDA provides a C-like programming language, a compiler, libraries, and a runtime environment. The CUDA compiler translates the C-like code into a form that can be executed on the GPU. The CUDA runtime environment provides the necessary runtime support for the compiled code. The primary programming paradigm used for GPU programming is the Single Instruction, Multiple Threads (SIMT) model. Some key terminologies of the GPU programming model which are based on CUDA are:

- Host code: it runs on the CPU and is responsible for controlling the flow of the program, managing memory, and launching kernels on the GPU.
- Device code: it runs on the GPU and performs the actual computation. Device code is written in CUDA C or CUDA C++.
- Kernels: they are functions written in CUDA C or CUDA C++ that execute on the GPU. Kernels are launched from the host code and can execute thousands of *threads* in parallel.
- *Thread* hierarchy: *thread* is a small execution unit within a GPU. CUDA supports a hierarchical *thread* organisation, where *threads* are grouped into blocks and blocks are grouped into grids. This allows developers to express parallelism at multiple levels. This is shown in Fig. 29.
- Memory management: CUDA offers several types of memory to ensure efficient data access and storage. Additionally, it provides

explicit memory management functions for allocating, transferring, and deallocating memory on the GPU. The different memory types are: global, shared, local, register, constant, and texture memory. Each memory type possesses distinct properties and serves specific purposes.

- Global memory: it refers to the primary memory space on the GPU, accessible by all blocks and their *threads*. While it offers a substantial storage area, its latency is higher than other GPU memory types. Physically, this memory resides in the GDDR[1] of the graphics card.
- Shared memory: it is a low-latency memory type accessible by all *threads* within a block. It is designated for data that *threads* in the same *block* need to share, facilitating efficient communication and data exchange. However, its capacity is limited compared to global memory. Amount of shared memory available per *block* depends on the underlying GPU architecture.
- Register: registers are rapid, on-chip memory locations within the GPU SM. While the compiler handles their allocation for variables, each register is specific to a single *thread* and remains available throughout the thread's lifecycle.
- Local memory: it is an off-chip segment of the device's global memory, stores variables that exceed the register space. Contrary to its name, it is not genuinely "local" to the chip. It operates slower than registers, is exclusive to individual *threads*, and remains available for the thread's duration.
- Constant memory: it is a read-only, cached memory type accessible by all *threads* across blocks. It houses constant data that remains unchanged during kernel execution. Compared to global memory, constant memory offers lower latency and higher bandwidth.
- Texture memory: it is a read-only and cached memory type, available to all *threads* across blocks. Optimised for 2D spatial locality, it is ideal for storing and accessing data in graphics

[1]Graphics Double Data Rate

applications. Like constant memory, texture memory boasts lower latency and higher bandwidth than global memory. Different types of memory with grid and *block* hierarchy in GPU model are shown in Fig. 29.

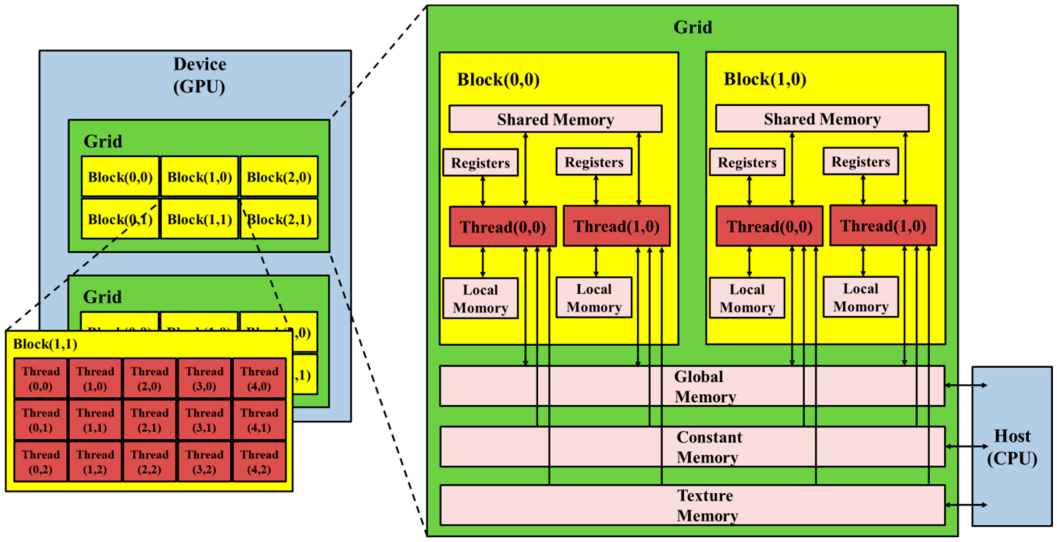


Figure 29: *Thread-Block* and memory hierarchy CUDA. Figure taken from [70].

- Synchronisation: it provides synchronisation primitives like barriers and locks for coordinating access to shared memory and ensuring that all *threads* in a *block* have completed their computations before proceeding.
- Streams: the CUDA programming model provides streams, which allow developers to execute multiple kernels in parallel. This allows developers to overlap the execution of kernels and data transfers, which can significantly improve performance.

3.2.3 The Allen framework

The Allen framework provides a compact, scalable, and modular solution designed for operating the LHCb HLT1 on GPUs. This section presents some key design principles of the Allen framework. The design of the framework mirrors the granular nature of the computing problem at hand. All Allen algorithms use Single Instruction Multiple Data (SIMD) operations. Traditional GPU algorithms leverage parallelism at the *Block*

and *Thread* levels. In LHCb, each *pp* collision event, is mapped to a *Block* on the GPU. Within this *block*, *Threads* with a shared cache and program counter process event segments concurrently. The *block* size is adjusted based on each algorithm's memory requirements. Key distinguishing characteristics of the framework include:

- **Scalability and modularity:** the Allen framework is designed to support a growing codebase. It requires minimal framework-specific knowledge for the integration of new algorithms and provides compile-time options for sequence configuration, debugging, testing, and deployment.
- **Memory management:** Allen avoids using dynamic memory allocation, opting instead for static memory allocation at the application's startup. A custom memory manager handles memory allocation and freeing upon demand, relieving developers of the need to invoke memory allocation routines. The framework design encourages algorithm to use GPU memory management techniques such as *Memory Coalescing*, where most favourable access pattern is used to access consecutive locations in the global memory, allowing for optional utilisation of global memory bandwidth.
- **Data types and access:** Allen's algorithms utilize the Structure of Arrays (SOA)[2] approach, promoting contiguous data access patterns and optimizing cache memory usage. Both temporal and spatial data locality are preserved on a per-algorithm basis. For certain algorithms, a K-Dimensional tree structure is implemented to optimize the spatial search window.
- **Integration with LHCb stack:** Allen provides integration with the broader LHCb software stack and can be invoked from `Gaudi` event loop (see Sec. 2.5.2). The framework is also harmonised with the LHCb software build system, enabling the generation of shared libraries and executables. For the Run3 commissioning phase, capabilities like online monitoring and data quality assessments have

[2]SOA is a data organisation method that improves memory efficiency and cache utilisation in SIMD (Single Instruction, Multiple Data) parallel operations. By storing each attribute type in a separate array, it facilitates rapid data processing in such environments, making it a preferred choice for high-performance computing applications.

been incorporated into the framework.

- **Platform support and portability:** the algorithmic design is not confined to a particular GPU architecture but is adaptable to any SIMD architecture. This adaptability is evidenced by the successful translation of algorithms for x86-64 processors and the Heterogeneous-compute Interface for Portability (HIP) platform. This translation has led to development of a custom performance portability layer, which is further discussed in the following section.

3.3 Performance portability layer of Allen

The work described in this section was carried out in collaboration with Carlos Sánchez Mayordomo and Daniel Hugo Campora Pérez.

3.3.1 Introduction

In the modern scientific computing landscape, heterogeneous computing architectures have become integral part of system design. Different processor types are uniquely suited to various computational tasks. For example, the LHCb DAQ and trigger systems for Run3 incorporate multiple architectures, including FPGAs, central to the DAQ board, and GPUs housed within the HLT1's event builder units. The key challenge is the development and maintenance of a codebase that remains efficient across an ever-widening array of architectures. The most common types of CPU architectures are x86, ARM, and PowerPC. Each of these has its own unique set of instructions.

Performance Portability Layers (PPLs) are software layers that promote easy portability across diverse architectures and platforms while preserving high performance. These layers are vital as they simplify software development, enabling code to be written and maintained across a wide range of hardware and software systems with relative ease.

3.3.2 Motivation for developing Allen support for AMD

Although Allen was initially developed using the CUDA framework specifically for Nvidia GPU platforms, we recognised the importance

of ensuring that the framework supports multiple architectures from various vendors. This was a critical step in avoiding vendor lock-in and ensuring that the framework is future-proof, especially considering the likely surge in competition in the GPU market. Consequently, developing support for x86-64 and AMD GPUs became essential. Before delving into the design and evolution of Allen's PPL, it is important to provide a brief overview of the different components of Allen's compilation infrastructure. As depicted in Fig. 30, it contains build tools such as CMake and Make and compilers responsible for translating source code into executable programs. Distribution of languages in the Allen's code-base is shown in Fig. 31.

As a benchmark of the capabilities of different hardware at our disposal, an example of RICH Quartic Solver SP Algorithm, which is a method for solving quartic equations using single-precision arithmetic, can be used. A comparison of the performance scaling this algorithm on Intel CPUs scalar vs Vector implementation and Nvidia vs AMD GPUs is shown in Fig. 32.

For a feasible HLT1 solution, the cost of the solution is a significant consideration alongside performance. The total cost stems from the hardware expenditure, software development costs, and maintenance overheads. Consequently, assessing the cost-effectiveness of different hardware platforms became crucial to the project. Considering our aforementioned example of the RICH Quartic Solver SP Algorithm, the throughput of the algorithm across various hardware platforms is illustrated in Fig. 33(a). In parallel, a comparative study of the cost associated with each platform is shown in Fig. 33(b).

As part of this thesis work, introducing cross-architecture capabilities in Allen commenced with the development of support for AMD's HIP architecture. At the time of this development, Allen comprised around 150 CUDA kernels with thousands of CUDA calls. Given that AMD's ROCm platform was a relatively new venture, the software support was minimal with limited features. Thus, the initial efforts were geared towards developing a HIP translation of the Allen framework that closely mirrored the original.

This process was undertaken in two phases. Initially, an ad-hoc auto-

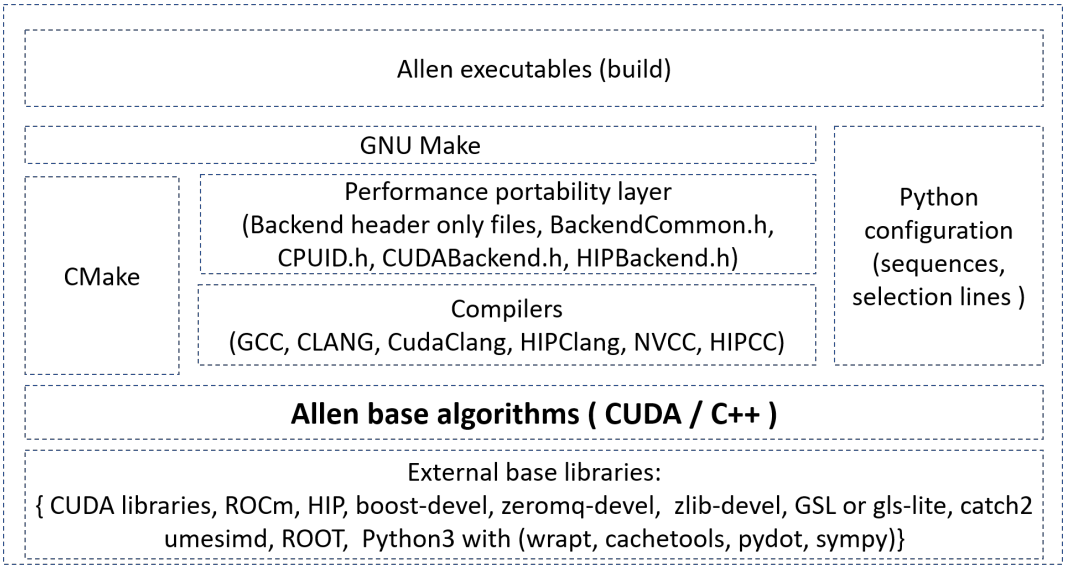


Figure 30: Allen software stack layout.

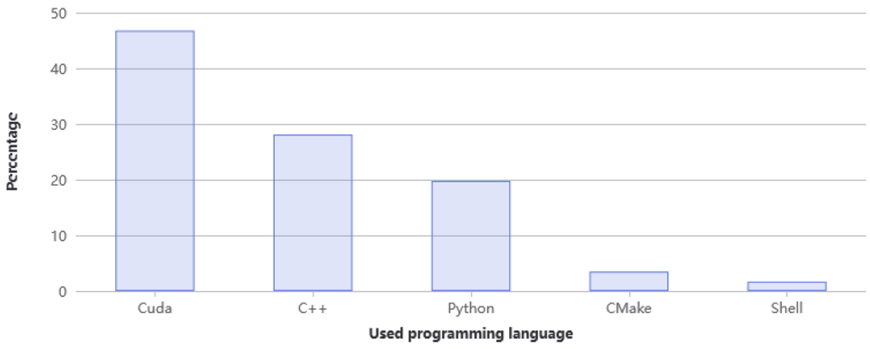
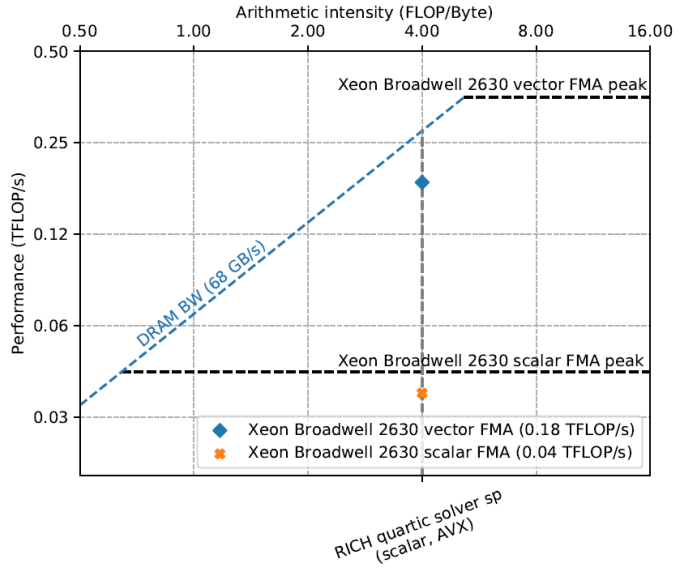
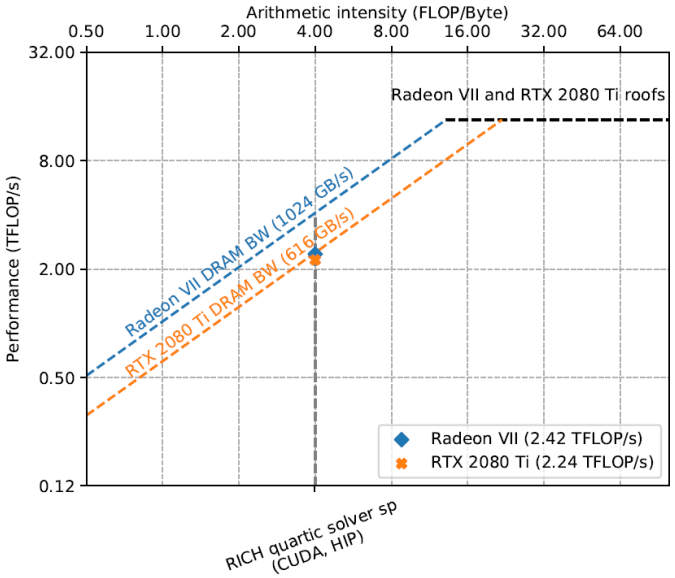


Figure 31: Allen codebase language percentage distribution.

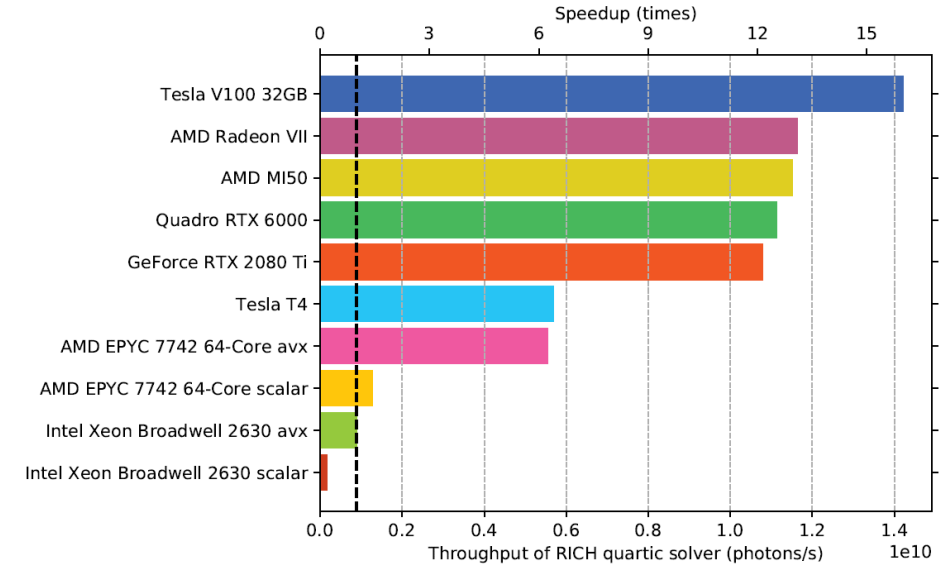


(a)

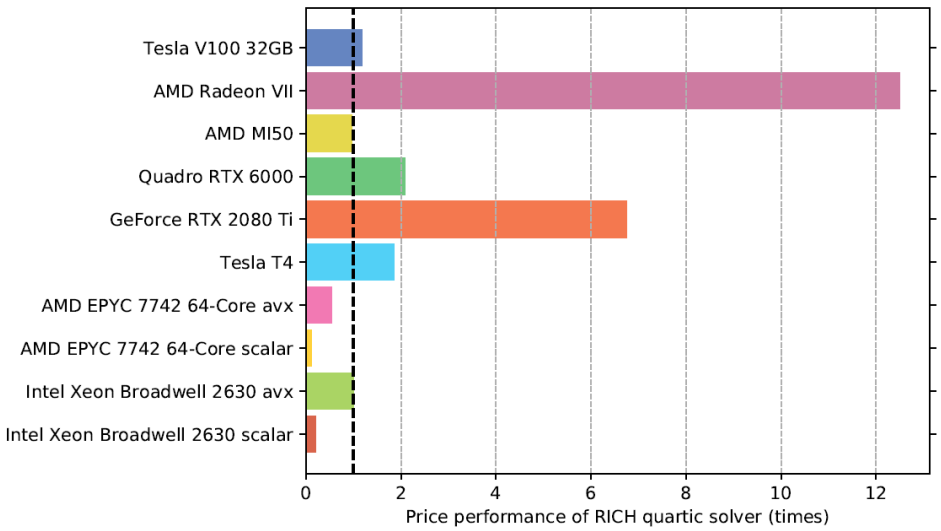


(b)

Figure 32: Scaling of RICH quadratic solver equation on (a) Intel CPUs for Scalar and Vector implementations and (b) Nvidia RTX 2080Ti vs AMD Radeon VII GPUs.



(a)



(b)

Figure 33: (a) Throughput of RICH Quartic Solver SP Algorithm on different hardware platforms and (b) Cost of the solution for respective hardware platforms.

translation script was tailored to match Allen's layout, translating CUDA code into HIP code. In the subsequent phase, certain sections of code, which used proprietary CUDA built-in functions, were re-implemented, utilizing C++ standard libraries, in order to circumvent CUDA features and libraries unsupported by HIP. A schematic representation of these procedures is illustrated in Fig. 34. After several iterations, the first fully functional version of Allen was developed, capable of running on AMD GPUs. This work utilised two AMD GPUs, the FIREPRO S9300 x2 (a passive card), and a newer active card variant known as the AMD Radeon FIREPRO WX7100.

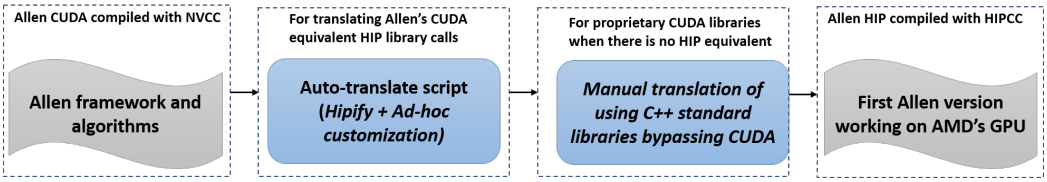


Figure 34: Schematic of the HIP translation process.

In light of the limited support from HIP in ROCm 2.0, a specific combination of build and compilation libraries was employed, namely, CUDA 9.0, LLVM+CLANG 6.0.1, cmake 3.12.3.9 (or above), Python 2.7, and ROCm 2.0. This phase provided critical insights into the capabilities and limitations of AMD's GPUs. This work was documented in the merge request MR!69[3]. However, the method involving an one-to-one mapping of CUDA kernels to HIP kernels and CUDA calls to HIP calls necessitated the maintenance of a separate codebase, either in a branch or an entirely different repository. This approach lacked scalability and was not deemed a viable long-term solution.

3.3.3 Single source compilation

The insights gained from these preliminary efforts showed that the heterogeneous compilers are more efficient when they can parse every segment of code associated with each compilation unit independently. This concept is often referred to as single-source compilation or inseparable compilation. To enhance the portability and scalability of the Allen

[3] https://gitlab.cern.ch/lhcb/Allen/-/merge_requests/69

framework, the subsequent step involved the development of a single-source compilation framework. This feature would facilitate architecture selection at compile time, thereby enabling the Allen framework to be compiled for different architectures without necessitating code alterations.

This was achieved by developing header-only translation libraries, which could permit architecture selection at compile time, as illustrated in Fig. 30. The primary header file for backend selection is included in all the other header files. This header file contains the backend selection code snippet as 3.1.

```
// Host / device compiler identification
#if defined(TARGET_DEVICE_CPU) || (defined(TARGET_DEVICE_CUDA) && defined(
    __CUDACC__)) ||
  (defined(TARGET_DEVICE_HIP) && (defined(__HCC__) || defined(__HIP__)))
------
// Dispatch to the right backend
#if defined(TARGET_DEVICE_CPU)
#include "CPUBackend.h"
#elif defined(TARGET_DEVICE_HIP)
#include "HIPBackend.h"
#elif defined(TARGET_DEVICE_CUDA)
#include "CUDABackend.h"
#endif
-----
```

Listing 3.1: Example of backend selection library from Allen.

The above code snippet shows the selection of the backend based on the preprocessor macros[4]. The choice is provided by the user at the compile time flag which is passed to the compiler. The below example shows the compilation command for the Allen framework on the AMD GPU.

```
cmake -DSTANDALONE=ON -DCMAKE_TOOLCHAIN_FILE=/cvmfs/lhcb.cern.ch/lib/lhcb/lcg-
    toolchains/LCG_101/x86_64-centos7-clang12+hip5-opt.cmake ..
```

Listing 3.2: Example of compilation command for AMD GPU.

If the user's choice of the backend is HIP then the preprocessor macro is defined and the corresponding backend header file is included in the code. The same applies for other backends as well. After adoption of this design, the first performance bench-marking over different cards

[4] `Allen/backend/include/BackendCommonInterface.h`.

was carried out. The performance comparison is shown in Fig. 35. The throughput measurement was carried out using MinBias sample with 10000 events with 1000 repetitions. Thereafter, using gitlab CI/CD auto-

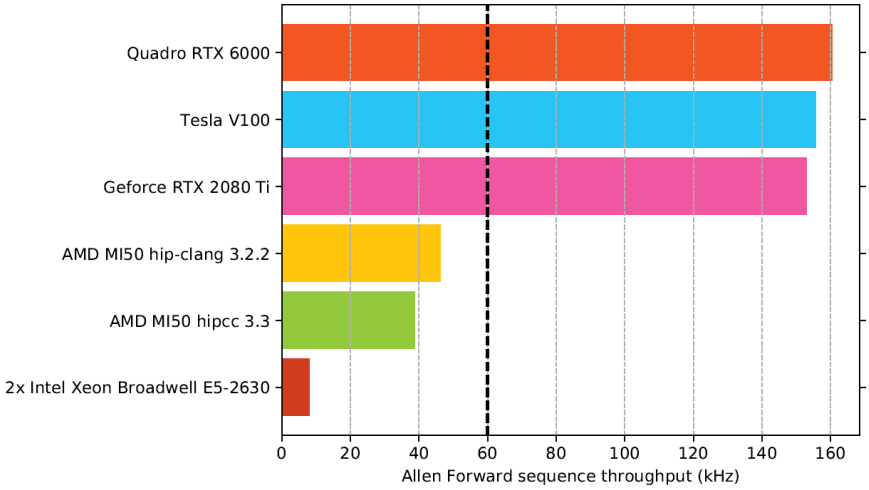


Figure 35: Performance comparison of the Allen framework on different cards.

mated continuous build and integration features, Allen throughput over different GPUs and CPUs evolved with more and more algorithms and features added over time. Various hardware specific optimisations also helped in improving the GPU occupancy and utilisation. The performance comparison of the Allen framework on different architectures after the hardware specific optimisations is shown in Fig. 36. The throughput performance of Nvidea Quadro RTX 6000 was almost 4 times higher than the AMD MI50. The default sequence used for this benchmarking

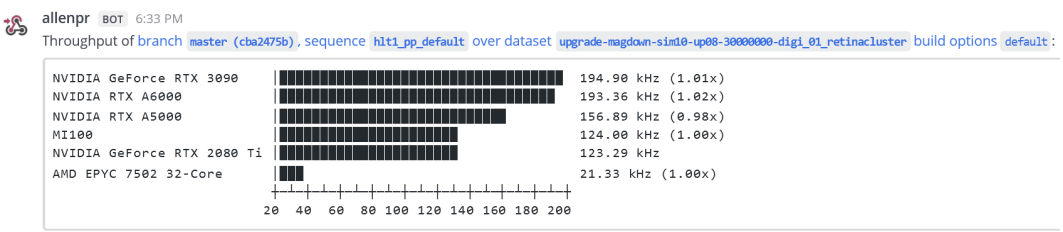


Figure 36: Performance comparison of the Allen on including hardware specific optimisations. MinBias samples with 10000 events and 1000 repetitions are used.

included the following algorithms as shown in Fig. 37.

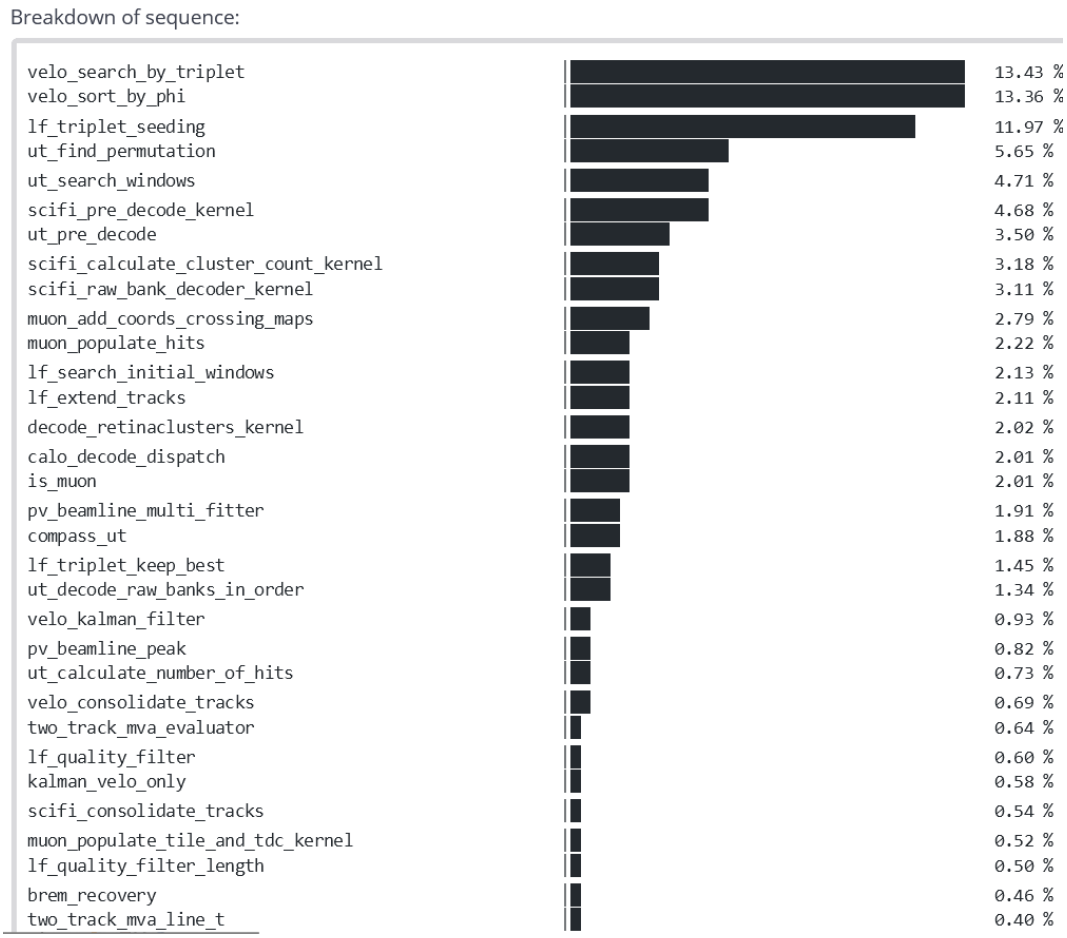


Figure 37: Breakdown of default sequence used for the performance benchmarking.

3.3.4 Impact on decision document

Towards the end of 2021, amidst concerted efforts to transition to a fully software-trigger system for Run3 data taking at the LHC bunch crossing rate, LHCb had two feasible options for HLT1: a completely CPU-based HLT1 and a Hybrid GPU-based variant. An exhaustive review process, supervised by external experts and non-collaboration reviewers, was initiated. The review entailed an in-depth technical assessment and forward-looking analysis for x86-based and GPU-based technologies. A thorough cost-effectiveness examination, considering significant parameters such as raw throughput, physics performance, design of the Online system and DAQ, constraints and choices in computing infrastructure design, future scalability and growth, the overhead of managing two distinct code-bases for HLT1 and HLT2, etc., was undertaken. The conclusion of the review indicated that the Hybrid GPU-based implementation was the optimal choice for HLT1. A detailed comparison of both the solutions is available at Ref [71].

During the review process, explicit queries regarding Allen's compatibility with architectures beyond CUDA were posed by reviewers. This consideration was crucial as it would enable LHCb to maintain a competitive tendering procedure for the HLT1 GPUs and avert exclusive dependence on a specific architecture and vendor. Insights gained during the thesis research provided answers to some of these queries and also helped in making the decision in favor of the Hybrid GPU-based implementation of HLT1.

3.3.5 Future outlook

For several decades, x86-based general-purpose computing architectures have served as the standard for software and computing framework development. Nevertheless, as computing demands rise and Moore's law approaches its limit, the emphasis is shifting towards accelerated computing on innovative parallel architectures.

Intel's latest scalable processor architectures incorporate dedicated on-chip accelerators and various specialised processors, each tailored to cer-

tain applications. In addition, newly introduced architectures from Intel (including GPUs and FPGA SoCs), ARM, AMD, and Nvidia bring along their respective development frameworks to facilitate heterogeneous computing, such as Intel's oneAPI, AMD's ROCm, and Nvidia's CUDA.

The PPL of LHCb's HLT1 stack offers a solid basis to harness these advancements in hardware and software. By introducing an abstraction layer for diverse compute platforms, the PPL allows developers to compose performance-optimised code that can be readily adapted to different architectures and frameworks. This versatility ensures that the HLT1 stack can effectively leverage the benefits of the most recent heterogeneous computing solutions.

4 Algorithms at HLT1

This chapter provides the description of the main algorithms in the current HLT1 sequence in the framework of the Allen project previously described. Section 4.1.2 explains the decoding process from the various subdetectors that provide inputs to the algorithms. The tracking and pattern recognition algorithms are then described in Sec. 4.2, where the definitions of LHCb track types and states are detailed. Two algorithms, `HybridSeeding` and `VELO-SciFi Matching`, are detailed in Secs. 4.3.1 and 4.3.2, respectively, as this thesis includes contributions to them.

4.1 The HLT1 sequence

A schematic of all the different processes of the HLT1 sequence is shown in Fig. 38. The HLT1 sequence can be broadly organised into three different categories of tasks:

- The decoding of the raw data from the subdetectors.
- The HLT1 reconstruction involving pattern recognition tasks such as clustering, tracking and vertexing.
- The HLT1 selection algorithms.

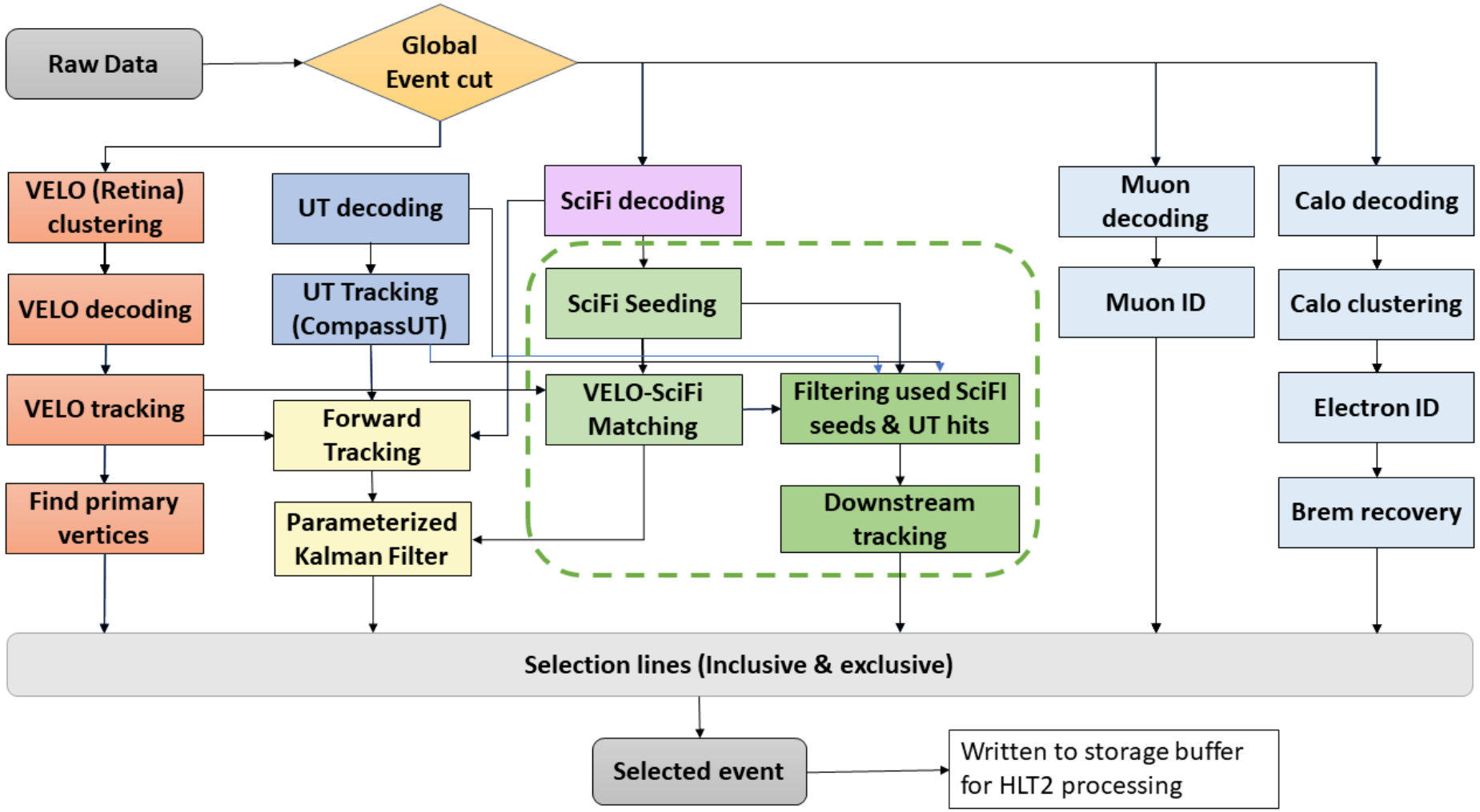


Figure 38: The algorithms participating in the LHCb HLT1 sequence, including decoding from raw data, reconstruction and selections. The green processes are the algorithms that have been developed in the context of this thesis.

4.1.1 General Event Cuts

The events with the highest occupancy, i.e. with the highest number of hits in tracking subsystems, require a disproportionate amount of resources for reconstruction in terms of calculation time. Additionally, since tracking performance typically decreases with occupancy, those same events are less useful for physics. For these reasons a Global Event Cut (GEC) is employed, specifically designed to eliminate the top 7% of Minimum Bias (MinBias) events with the highest occupancy prior to any processing. This cut is based on the size of the raw banks of the SciFi and UT in each event.

Future considerations might involve a cut based on the occupancy of a different subdetector, potentially the VELO. The main conclusion drawn is that the chosen baseline cut is adaptable, including both the cut value and the specific subdetector where the raw bank size cut is applied.

4.1.2 Decoding of the raw data from the subdetectors

A collection of unprocessed data from a specific subdetector or readout channel is referred to as a *raw bank*. During data acquisition (see Sec. 2.4), signals from subdetectors are digitised and organised in the detector readout electronics as a series of raw data words. These words encompass information regarding the amplitude and arrival time of each detected signal. The decoding process transforms the raw data from each subdetector into a format that aligns with the HLT1 tracking system.

VELO decoding

Starting with the tracking subdetector closest to the interaction point and traversing outwards, we begin with the VELO. As described in Sec. 2.3.1, when a particle passes through the VELO, it typically triggers multiple pixel hits per module. To accurately determine the position of the hit, the activated pixels that are adjacent to each other must be identified and grouped together to form clusters. To perform this task, FPGA-based clustering algorithm which runs on the existing FPGAs in the TELL40

readout boards, have been developed [72]. The output of the clustering algorithm is a list of clusters, each containing the position of the cluster, the number of pixels in the cluster, and the total charge.

UT Decoding

The UT decoding step involves reading the raw data from the UT sensors and organizing them in the parameters required for the track reconstruction step. This is performed in parallel on the GPUs in the HLT1 with each GPU given different chunks of raw data. The data from the UT detector is compressed and stored in raw banks, which include all the essential information needed to extract the UT hits. The parameters obtained through decoding include:

- *LHCbID:* it is a unique identifier for each hit in the LHCb detector. It is a 32-bit integer which is composed of 3 parts: the detector type, the station number, and the channel number. - *z at y = 0:* this is the z coordinate of the hit at the $y = 0$ plane. For each of the stations a and b (see Fig. 17), it is the centre of the panel in y-axis.

- *x at y = 0:* similarly, this is the x position at the center of the panel in the y axis. This is determined by the activated strip in a given sector.

- *yBegin and yEnd:* since the UT subdetector comprises vertically aligned strips, it is not possible to determine the precise y coordinate of a hit. Instead, the y coordinate is represented as a range that indicates the area where the hit is situated.

- *weight:* this is the weight of the hit. It is used to determine its quality. The weight is calculated by taking the ratio of the charge of the hit to the charge of the strip.

These parameters are stored in a SOA layout for coalesced memory access. In addition, a separate array is used to store the offsets between the hits for identifying the specific UT station. More details can be found in Ref. [73].

SciFi Decoding

The farthest tracking station from the interaction point after the dipole magnet is the SciFi (see Sec.2.3.1). Before entering the HLT1 sequence,

the raw data from the SciFi is preprocessed and sorted by x-coordinate. It is structured with headers and banks, as illustrated in Fig. 39. The data, encoded in 16-bit words, comprises clusters, which could be single or fragmented, differentiated by a specific bit in the word.

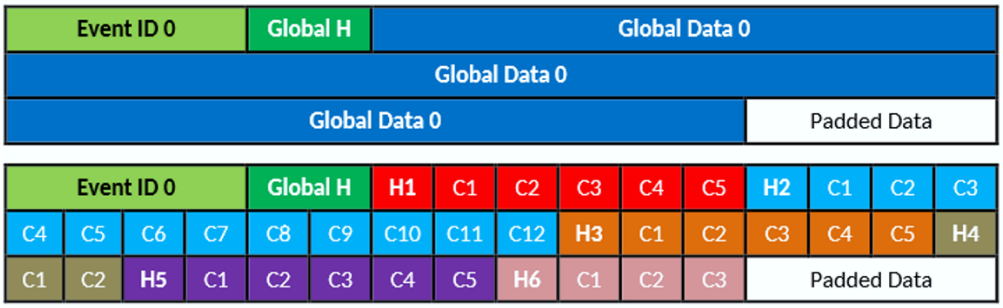


Figure 39: Data format from the TELL40 on a 256-bit-wide frame. The frame depicts the scenario where six data fibers are connected to the TELL40, utilising the FH/VD format. In this configuration, the local SiPM headers are also transmitted. Figure from Ref. [74].

As particles traverse the SciFi detector, they intersect several fibres, generating photons that trigger more than one SiPM channel. These channels are combined into clusters. The *Pacific* chip digitises the data from each channel into a 2-bit threshold value and transmits it to a clustering FPGA. The FPGA employs a clustering algorithm to generate clusters, which are then transmitted to the TELL40 boards via an optical fibre. The TELL40 processes the data from multiple optical fibres and outputs the data as raw banks.

The decoding process converts these raw banks into a format that can be used for further analysis. It decodes position and characteristics of each cluster. This involves reconstructing the *FTLiteCluster* object with the *FTChannelID*, the fraction bit, and the size flag.

Calorimeter decoding

After a Calo preprocessing, the data is decoded to extract the physical quantities such as the energy and position of the hit. The decoding process is carried out in two steps: unpacking the data and decoding the data for each channel.

- Unpacking the data: the raw data from the calorimeter detector is organised into a series of 32-bit words. The first step in the decoding process is to unpack the data from these words using bit-wise and bit-shifting operations.
- Decoding the data for each channel: once the data has been unpacked, the next step is to decode the data for each channel. This is done using parallel algorithms that use lookup tables to convert the raw data into physical quantities such as the energy and position. The lookup tables are precomputed and stored in GPU memory, allowing for fast access during the decoding process.

Muon decoding

The LHCb muon system is described in the Sec. 2.3.3. Similar to the other decoding algorithms, the muon decoding process involves several steps, including data unpacking, decoding of the readout elements, and conversion of the data into physical quantities such as position and angle. The muon detector consists of several layers of chambers, each of which contains several readout elements. The geometry of the muon detector is taken into account during the decoding process to ensure that the decoded data is accurate and reliable.

4.2 Tracking and pattern recognition

Following the decoding of the detectors, the subsequent phase for particle reconstruction is the pattern recognition. In this step, tracks are formed from individual hits and clusters. Various algorithms for pattern recognition are utilised across the different tracking systems.

4.2.1 Representation of *track states* in LHCb

In LHCb, the coordinate system adopted for the tracking system is right-handed with axes denoted as x, y, and z. Fig. 40 illustrates the layout of this coordinate system and can be summarised as follows:

- The z-axis is collinear with the beam pipe, originating at the collision point and extending through the detector.

- The x-axis, perpendicular to the z-axis, lies in the plane where charged particles deviate due to the magnetic field, termed as the bending plane or the x-z plane.
- The y-axis is orthogonal to both the x and z axes, pointing upwards, and defines the non-bending plane, referred to as the y-z plane.

Particle tracks are approximated as a series of linear segments, where each segment is termed a *track state*, defined at a given z. This representation is based on the premise that over short intervals, the track can be approximated as a straight line despite the curvature introduced by the magnetic field. Each track state is characterised by a state vector, $\vec{x}$, and an associated covariance matrix. The state vector is a five-dimensional vector defined as:

$$\vec{x} = \begin{pmatrix} x \\ y \\ t_x \\ t_y \\ \frac{q}{p} \end{pmatrix}, \tag{4.1}$$

where

- x and y denote positions along the horizontal (x) and vertical (y) directions.
- t_x and t_y represent the tangents of the direction angles with respect to the z coordinate, defined as $t_x = \frac{\partial x}{\partial z}$ and $t_y = \frac{\partial y}{\partial z}$.
- $\frac{q}{p}$ signifies the charge-to-momentum ratio, where q represents the charge and p denotes the momentum of the particle.

The covariance matrix, a 5×5 matrix, incorporates the uncertainties and correlations of the components of the state vector and plays a crucial role in track fitting and analysis. This basic common track model provides a unified representation of the track states across the different tracking systems of LHCb. Further details and adoption of this model according to the subdetector traversed by the particle and the track types are described in the following sections and subsequent chapters.

4.2.2 Track types

The main track types based on the subdetector traversed are shown in Fig. 40 defined as

- *VELO* tracks, are the tracks which pass only through the VELO subdetector.
- *Upstream* tracks, are tracks from the particles with short lifetime, which pass through VELO and UT but do not reach SciFi.
- *Long* tracks, which have information from at least the VELO and the SciFi, and possibly the UT. These are the main tracks used in physics analyses and used at all stages of the trigger.
- *Downstream* tracks, which have information from the UT and the SciFi, but not VELO. They typically correspond to decay products of K_S^0 and Λ^0 hadron decays.
- *SciFi* tracks, or *T* tracks, which only have hits from the SciFi.

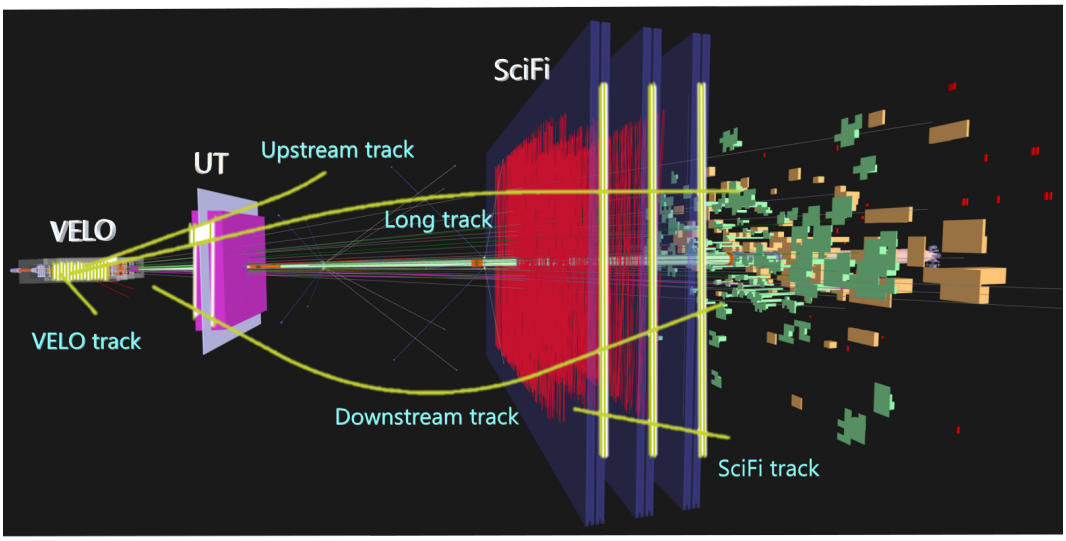


Figure 40: Different track types in LHCb based on the detectors traversed.

4.2.3 *VELO*-tracks reconstruction

As a first detector around the collision point, VELO has to deal with huge number of hits coming from the high collision rates[1]. Around 1 billion of particle tracks per second are expected in *pp* collisions. Composed of 52 planes of silicon pixel sensors encircling the interaction region, the primary function is the reconstruction of Primary Vertices (PVs) and the generation of initial track seeds. These seeds are then propagated through the other detectors of the LHCb for further processing.

The initial phase of reconstruction entails grouping measurements, which are generated when a particle traverses each silicon plane, into clusters. This technique is a specific implementation of the broader process referred to as *connected component labeling*[2]. To locate clusters within compact areas, Allen's clustering algorithm employs bit masks, thus allowing each region to be treated independently for parallel processing.

The reconstruction of straight-line tracks begins with the generation of three-hit seeds *triplets* from successive layers, using the algorithm called `Search by triplet` [75]. These triplets are then extended to other layers, considering that prompt particles from *pp* collisions that travel through the detector are straight lines with a constant polar ϕ angle from the PV. To facilitate rapid reference during the combination of hits into tracks, the hits on each layer are sorted by their ϕ angle. Lastly, these initial *VELO* tracks are refined using a simple Kalman filter [76].

4.2.4 PV reconstruction

The PV reconstruction at HLT1 (`TrackBeamLineVertexFinder` [69]) is based on performing a peak search of z-positions of *VELO* tracks at their point of closest approach to the beamline. Entries are stored in a histogram, where a cluster indicates a PV candidate. Peaks can be then isolated and fitted using Gaussian density distributions. The density distributions take the uncertainty of the track states into account. Instead of performing one-to-one mapping between a track and a vertex, every track is assigned

[1] about 3000 hits per event every 25 ns

[2] A technique in image processing to identify and label connected regions in binary images.

to every vertex based on a weight, allowing all candidates to be fitted in parallel.

4.2.5 `Compass-UT` reconstruction algorithm

For the reconstruction of tracks in the UT detector, the UT hits are decoded into regions based on their x-coordinate as discussed in Sec. 4.1.2. Every region is then sorted by the y-coordinate. This allows for a fast look-up of hits around the position of an extrapolated *VELO* track using `Compass-UT` algorithm [73]. The algorithm extrapolates *VELO* tracks to UT stations, taking the effect of the magnetic field into account. The `Compass-UT` algorithm is specifically designed for the multi-core parallelism offered by GPUs. It can be configured using two parameters: the number of sectors to search for hit candidates, and the number of valid candidates to form a track. UT window ranges are defined by two hit indexes with all the hits in between considered as potential candidates for creating a track. Fig. 41 shows an sketch of this searching. After the search for the window ranges is complete, the window ranges are stored in a pre-allocated memory space, as pairs composed of a hit pointers and the size of the window. Thereafter, a rigorous process is initiated to search for the best compatible hits in all the UT panels to form a tracklet. A tracklet is considered valid if it consists of at least three hits on different stations. The combination that best match the extrapolation from the *VELO* track is searched, considering the influence of the magnetic field that introduces a small kink in the particle trajectory. The order of the stations is inverted in the search: hits are searched in the window ranges first in forward direction and if no hits are found, the backward direction is tested to find a tracklet. The `Compass-UT` algorithm, is parallelised over the *VELO* tracks. In this setup, each thread is responsible for processing the tracklet search for each track. The best configuration of the algorithm is obtained when five sectors are used. The tracking reconstruction efficiency is close to 95% and the fake rate around 5%.

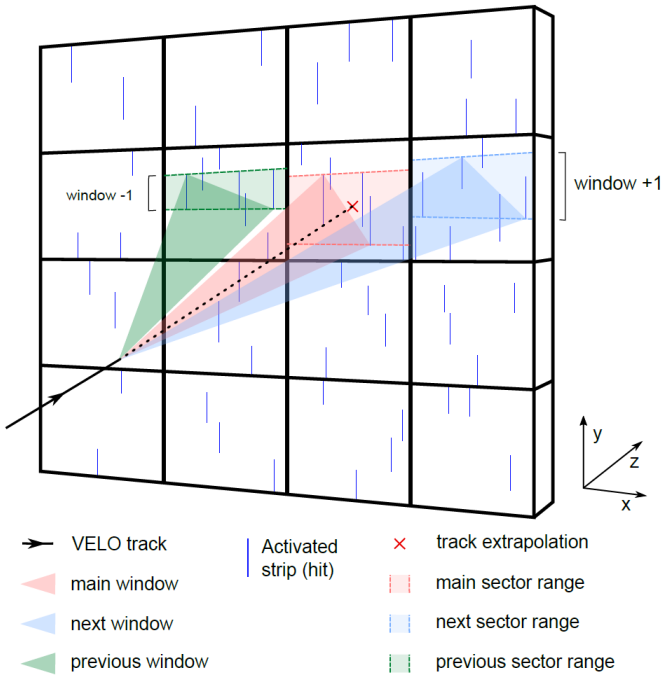


Figure 41: Representation of UT window ranges with a *VELO* track extrapolation to a sector. Several hits lie within the range of the windows, which are considered for UT tracking. Figure from Ref. [73].

4.2.6 SciFi reconstruction

The SciFi reconstruction is part of several algorithms inside the Allen framework, namely the `Forward` [69], the `HybridSeeding` [77] and `VELO-SciFi Matching` [78]. The `HybridSeeding` and `VELO-SciFi Matching` are described in detail in the coming sections. The standalone SciFi reconstruction algorithm (named `HybridSeeding`) uses the decoded SciFi hits described in Sec. 4.1.2 to create standalone SciFi tracks. The `VELO-SciFi Matching` uses these tracks for *long* tracks reconstruction. In addition, the `Forward` algorithm uses the SciFi hits to produce *long* tracks.

The `Forward` tracking algorithm

Tracks traversing both the VELO and UT detectors are projected onto the SciFi detector using a parameterisation derived from the track direction and the momentum estimate post-UT tracking. This avoids the need to load a sizeable magnetic field map into the GPU memory. A search window, dictated by the UT track properties and a maximum hit count,

is established for every UT track and each SciFi layer.

The scintillating fibres of the SciFi have a hit efficiency ranging from 98% to 99%. To ensure that track reconstruction efficiency is not compromised, multiple SciFi seeds per UT track are permitted. These seeds are produced by combining triplets of hits from the search windows of one x-layer in each of the three SciFi stations. Taking into account the track's curvature within the SciFi region due to tail of residual magnetic field from the LHCb dipole, seeds exhibiting the lowest χ^2 relative to a parameterised track description within the SciFi volume are selected.

The magnetic field within the SciFi detector is expected to be small, and is expressed by a linear function $B_y(z) = B_0 + B_1 \cdot z$, where the ratio B_1/B_0 is constant at the first order. Utilising this parameterisation, tracks are projected onto the remaining x and u/v layers. To form the *long*-tracks, hits minimally deviating from the reference trajectory are added. The u/v layers solely provide information on the track motion in the $y-z$ direction. Therefore, a track model accounting for slight curvature in the $y-z$ plane is incorporated once all hits have been included.

Finally, a least mean square fit is executed in both the x and y coordinates. Each track is assigned a weight based on the normalised $x-$fit χ^2, $y-$fit χ^2, and the number of hits in the track. To minimize the presence of fake tracks, only the best track per UT track is accepted. Fig. 42 shows an sketch of the `Forward` tracking algorithm.

4.2.7 Kalman Filter

In the HLT1 stage, the Kalman filter comes into play after the initial hit detection and track reconstruction stages. The filter takes the track output from reconstruction stage and tries to improve the precision of the measurements, specifically focusing on the impact parameter (IP).

The nominal Kalman filter setup in LHCb uses a method called Runge-Kutta extrapolation, which is a very precise way of predicting the state of a system (like the position and momentum of a particle) at a future point in time based on its current state. It also employs a detailed model of the detector to account for noise introduced by multiple scattering, which refers to the deflection of particles as they pass through the

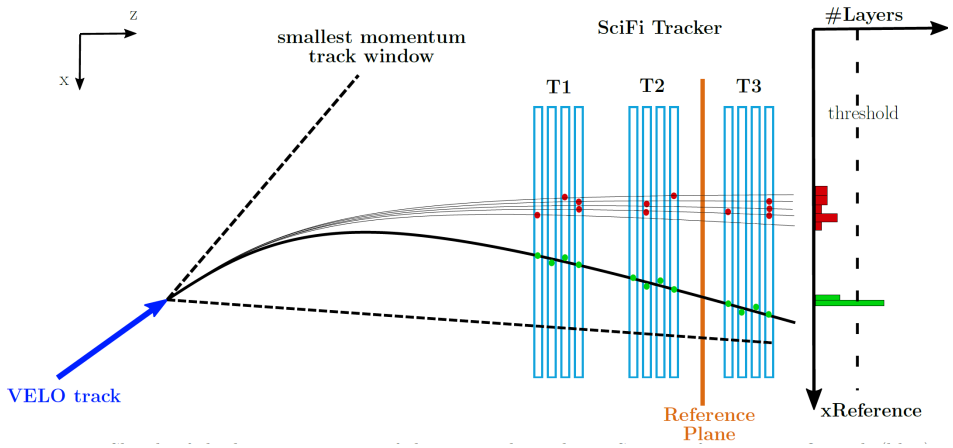


Figure 42: Sketch of the `Forward` tracking components. From a *VELO* track (blue), a search window (black dashed line) determines the smallest-momentum hit. SciFi station hits (light blue) project to the reference plane (orange) using a basic track model. The right side histogram counts unique SciFi layers based on projected x positions. Hits from the *VELO* track are green, others are red. Figure from Ref. [79].

detector material. However, at HLT1 level, this approach is incompatible due to throughput requirements. Therefore, these operations are replaced with parameterisations, or simplified mathematical models. Two versions of this parameterised Kalman filter are implemented: one considering the whole detector and one focused on the *VELO* track segment, but using the momentum estimate from the full track passing through the VELO, UT, and SciFi detectors.

The VELO-only Kalman filter is used in the HLT1 sequence because the impact parameter is primarily influenced by measurements closest to the interaction region, which is within the domain of the VELO detector. This approach achieves a significant computational speedup compared to applying the full Kalman filter [80].

4.2.8 Muon reconstruction

For Run3 a new algorithm to reconstruct and match *VELO* tracks with muon hits has been developed inside the Allen framework called `MatchVelo-Muon`. It consist of three parts:

- `FindMuonHits`

- `FitMuon`
- `MatchVeloMuon`

The first two steps are devoted to reconstructing muon tracks from the hits left at the muon stations. The first step looks for muon hits following some search windows, starting from the outer station, and the second fits these hits to line both in x-z and z-y planes. A linear extrapolation at each station is performed. The last algorithm matches *VELO* tracks to muon tracks using a χ^2 metric, following the same strategy of the `Matching` algorithm described below. Fake track removal is achieved by applying χ^2 thresholds. The momentum resolution for this types of tracks is 40%. The efficiency obtained for muons coming from b-hadron decays is about 54%.

4.2.9 Calorimeter reconstruction

LHCb calorimeters (ECAL and HCAL) as described in Sec. 2.3.3 are used for particle identification and energy measurements of photons, electrons and hadrons. The algorithms are responsible for reconstructing the energy and position of the hits in the calorimeter using following procedure.

- ***Calo Clustering***: utilising the output from Calo decoding (described in Sec. 4.1.2), clustering is implemented at HLT1. The procedure comprises identifying seed clusters, prefiltering clusters and determining clusters.
  - *Seed Clustering*: the process begins by identifying seed clusters in the calorimeter. This involves looping over all the calorimeter digits and checking if their ADC (Analog-to-Digital Converter) values are above a minimum threshold and if they represent local maxima. If so, they are identified as seed clusters and stored in an output array.
  - *Prefiltering clusters*: after identifying seed clusters, the process pre-filters the clusters. This involves looping over all the clusters and checking if their transverse energy and deposited energy in cells are above a defined thresholds.

  - *Finding clusters*: the next step is to define the clusters. This involves retrieving the ECAL geometry and then building simple 3x3 clusters from the prefiltered clusters. For each cluster, the total energy is calculated by summing the energy of neighboring cells that meet certain criteria. The position (x, y) of the cluster is updated, weighted by the energy fraction of each cell. Additionally, the transverse energy and deposited energy for each cluster are calculated.

  Each of these steps is performed independently for each event in parallel. In this algorithm the overlapping cells are not resolved and only the neighbors above a threshold are added to the cluster. A more advanced full calorimeter clustering is performed at HLT2 which uses a graph clustering algorithm [81]. In comparison to the HLT2 clustering, the efficiency of the HLT1 clustering is 10% lower.

- **Electron ID**: the process involves assigning a specific ID to tracks that are likely to be caused by electrons, this is achieved in following steps:

  1. *Iterating over detector positions*: the process begins by iterating over several z positions along the ECAL. For each z position, the `ecal_scan` function extrapolates input track in a straight line to the current z position using the track's current state.
  2. *Coordinate conversion to cell ID*: the extrapolated position of the track is then converted to a cell ID within the ECAL.
  3. *Checking validity and detector acceptance*: each cell ID obtained from the extrapolation process is checked for validity. Invalid IDs or those corresponding to cells outside the detector acceptance are discarded.
  4. *Energy estimation*: the ADC value for each valid cell is converted into an energy measurement. Noise or spurious signals are typically rejected at this stage based on energy thresholds.
  5. *Energy summation and digit index storage*: the energy contributions from all unique cells traversed by the track are summed

to provide a total energy measurement for the track within the ECAL. The indices of these cells are also stored for further reference.

By the end of this process, the algorithm has calculated the total energy of all the unique calorimeter digits that the track has traversed and stored the indices of these digits.

- **Brem recovery**: it is a technique used to correct for the energy loss by charged particles. The basic idea is to locate the photons emitted due to bremsstrahlung and add their energy back to the energy of the electron. For each event, the algorithm loops over the *VELO* tracks and calculates the intersection of the track with three different planes of the ECAL. After determining which ECAL cell the track passes through, it sums up the energy of these cells.

4.3 Standalone `HybridSeeding` and `Matching` algorithms

Work in this section has been carried out in collaboration with C. Agapopoulou, L. Calefice, A. Hennequin, L. Henry, L. Pica, V. Svintozelskyi and J. Zhuo. It consists of a CPU based HLT1 implementation inspired from the HLT2 `HybridSeeding` [77] and `PrMatchNN` [82] algorithms. The parallelisation scheme and GPU implementation of these algorithms has been significantly upgraded and optimised for meeting the throughput requirements of HLT1 as documented in merge request[3] and in Refs. [83] and [78].

4.3.1 `HybridSeeding` algorithm for HLT1

The SciFi standalone `HybridSeeding` algorithm for HLT1 on GPUs is an adaptation of the `HybridSeeding` algorithm implementation at HLT2 level based on CPUs. The algorithm is based on reconstructing tracklets in the *x*-*z* plane and then confirming the candidates in the *x*-*y*-*z* space by adding *u*-*v* information. It is organised into two cases, each with varying

[3] https://gitlab.cern.ch/lhcb/Allen/-/merge_requests/742

initial layers, this helps minimizing the tracking performance losses due to hit detection inefficiencies from any specific layer. Each case consists of two main stages of the algorithm: the first being `Seeding_XZ`, which performs pattern recognition in the x-z plane (also the bending plane for tracks), followed by the second stage, `Seeding_confirmTracks`, wherein the xz candidates are confirmed after adding the y information. These stages are explained below.

`Seeding_XZ`

In the first case, the first x-layers of the three SciFi stations are used while in the second case, the second x-layers are used. For every hit in the first x-layer of the T1, a small search window is opened in the first x-layer of T3 and all the hits in that window are matched to create two-hit combinations. The position and width of this search window are determined based on the assumption that the particle's origin vertex is (0,0,0) and is with a minimum track momentum of $p_{min} > 3$ GeV/c. At this stage the tracks are assumed to be straight lines in the bending plane and the size of the search window in the T3 is adjusted using parameters which are tuned using simulation samples and depend on the position of the layer of respective station and the momentum range. Figure 43 shows an sketch of the algorithm principle. For every two-hit candidate,

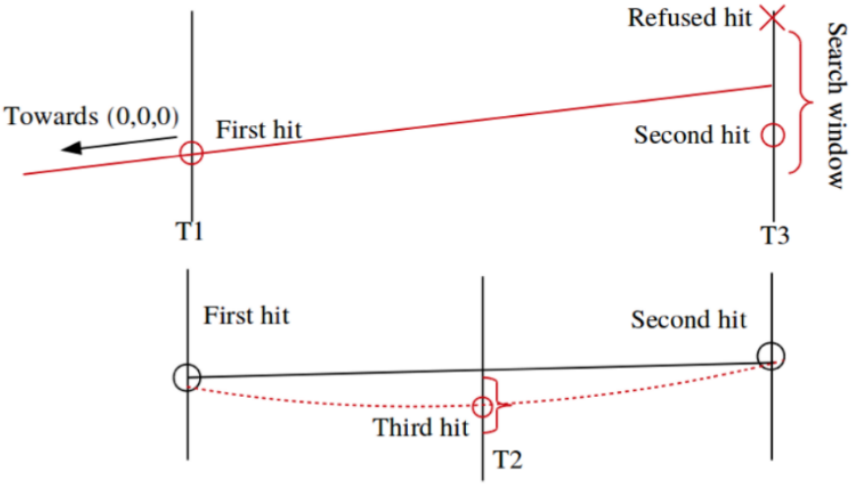


Figure 43: Illustration of the two- and three-hit searches in the x-z plane.

slope calculations are performed using the extrapolation of the doublets

to (x=y=z=0). An initial track momentum assumption of $p_{\text{min}} > 3\,\text{GeV/c}$ for which the seeding is tuned is used to determine the size of the window in the first x-layer in T2. Due to the remaining magnetic field in the SciFi detector, tracks in the x-z plane follow trajectories defined by the expression of the parabolic track model in Eq. 4.2.

$$x(z) = a_x + b_x\bar{z} + c_x\bar{z}^2(1 + \Delta B/B\bar{z}), \tag{4.2}$$

where $\bar{z} = z - z_{\text{ref}}$, with $z_{\text{ref}} = 8525\,\text{mm}$, and a_x, b_x, c_x are terms of the parabola that need to be determined. The variable $\Delta B/B$ is fixed using the simulated samples and represents the decreasing magnetic field along the z direction. This allows to define a tight search window for hits in the first x-layer in T2. Taking the bending into account, all the hits from the T2 first x-layer within the tolerance window are added to create three-hit combinations as depicted in Fig. 43. The three-hit combinations are fitted with a parabola track model with cubic correction term.

The three-hit combination's parabolic trajectory is subsequently projected onto the remaining three x-layers, and any detected hits within a permissible window of 1 mm are incorporated. The parabola is then extended to the remaining layers. The candidates with at-least two additional hits from remaining layers, i.e. at least five out of six possible hits, are selected, and a track fit is performed in the x-z plane. This same procedure is performed in the second case where the second x layers of all stations are used as the initial set of layers. Each track candidate is assigned a score based on the number of hits and χ^2 value. A GPU-optimised shared memory $O(N)$ voting algorithm is used for clone tracks removal before the next step of adding y-information.

Charge-momentum estimation and tolerance windows

The xz-seeding utilises tolerance windows, which are fine-tuned using simulated $B_s^0 \to \phi\phi$ decays. These windows are parameterized based on momentum and other topological variables. For efficient GPU seeding, an initial track momentum assumption of $p_{\text{min}} > 3\,\text{GeV/c}$ was adopted.

Tracks can be approximated as originating from the point $(0, 0, 0)$. The trajectory of a track can be predicted based on its first hit in the

detector. The relationship between this back-propagated x and q/p value for *long* tracks is modeled as

$$\frac{q}{p} = f(x_0) = a|x_0| + bx_0^2,$$

with specific parameters derived from simulations. The difference between the measured and predicted hit position in the T3 layer is proportional to x_0. Using the $q/p = f(x_0)$ model, a momentum cut can be translated into a tolerance on this difference. The tolerance window's limits are then calculated, and over 99% of hits from tracks with $p > p_{\text{min}}$ fall within this window.

The third hit's deviation from the expected line provides another measurement of the track's q/p. This deviation is proportional to the track's quadratic coefficient. This tolerance is equivalent to a momentum measurement using only the residual magnetic field inside the SciFi detector. The distribution of this deviation as a function of q/p shows that most tracks follow a linear relationship, but some deviate due to factors like energy loss or different integrated fields. An additional correction factor, α_{corr}, is introduced to account for charge asymmetry as a function of the initial hit position. This factor is modeled as a function of momentum, and its parameters are derived from fits to simulated sample. The established tolerances, including the α_{corr} correction, are designed to encompass 99% of the tracks.

Adding *uv* information and confirmation

In the case of a specific layer at a location $z = z_0$ and tilted at an angle θ, a measurement at $x = x_0$ can be converted into a y_0 measurement for a track with a given $x(z)$ equation, as expressed by:

$$y_0 = \frac{x(z_0) - x_0}{\tan(\theta)}. \tag{4.3}$$

Given that the magnetic field in the SciFi region primarily aligns with the y direction, trajectories are expected to follow straight lines in this direction. Based on the assumption that tracks originate from the origin, the y slope, denoted as t_y, should remain constant.

Consequently, for each parallel x-z track considered, every hit of an initial layer is examined, and the associated t_y is computed. This value

is then used to establish small tolerance windows in all other layers, and the hit nearest to the extrapolated position is collected. With each addition of a hit to the combination, the t_y value is updated.

Ultimately, a maximum of one combination per first hit considered is constructed. These combinations are then fitted, and the fit quality, along with the number of hits, is employed to select the candidate. Finally, a second iteration is performed, designating another layer as the initial layer, to compensate for hit inefficiencies. Tracks with a combined total of 10 hits or higher from x and uv layers are accepted and a linear model in y is used to calculate track parameters. The candidates with best χ^2 are confirmed as SciFi seeds. All these operations are carried out in parallel for all the xz candidates.

In the previously outlined process, executing the xy seeding twice tends to create a multitude of clones. The techniques employed in the HLT2 version of the seeding for eliminating these clones carry an $O(n^2)$ complexity, which is not ideal in this context as it would significantly affect performance. To address this, we introduce a voting algorithm. This algorithm calculates a score for each hit included in the tracks, based on the following formula:

$$s = 1000\chi^2 + c_x, \tag{4.4}$$

In this formula, χ^2 provides the fit quality of the track and c_x is the parabola term of the trajectory in x . If this calculated score is less than the current score attributed to a hit, the new score will be assigned to that hit.

The term c_x is included in our scoring equation due to the need for a deterministic way to break ties when two track χ^2 values are the same. The selection of c_x is specifically motivated by the intention to favor tracks with higher momentum over those with lower momentum, thereby prioritizing tracks that are more likely to come from b-hadrons decays.

Afterwards, we conduct a second round of evaluations on the tracks. A track will be preserved if a minimum of four hits are associated with this specific track score. This operation is designed to run in parallel over the tracks, optimizing the computational efficiency.

4.3.2 `VELO-SciFi Matching` algorithm for *long* tracks

VELO tracks are matched to the SciFi track seeds reconstructed by the standalone seeding algorithm described above to create *long* tracks. This is shown in Fig. 44. Before beginning with the matching step, some of the *VELO* tracks are filtered out if they do not meet some basic criteria. The tracks generated due to back-propagation of the particles are removed and only the tracks that fall within the geometrical acceptance of the SciFi are considered for matching. This algorithm is adapted from a CPU-based HLT2 matching algorithm called `PrMatchNN` [82].

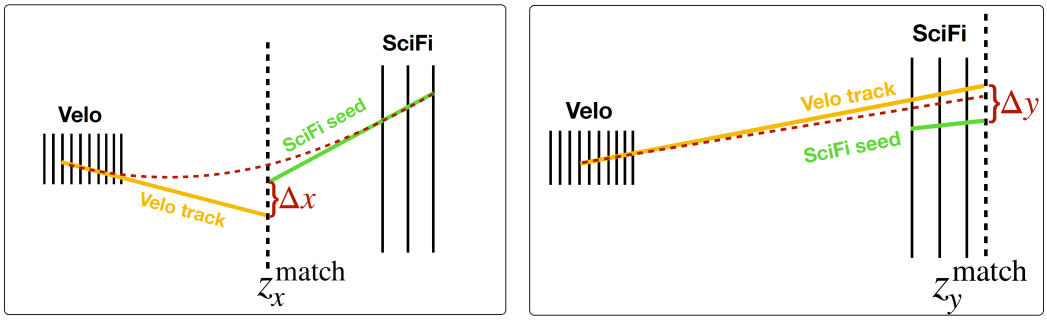


Figure 44: Illustration of the work principle of the `Matching` algorithm in the (left) x-z plane and (right) y-z plane. Figures from [83].

A fast q/p estimation of tracks called the "p_T *kink*" approximation method is used to predict the tolerances at the matching plane in z for a given momentum requirement, which is different for x_z and y_z planes as depicted in Fig. 44. The actual momentum kink depends on the integrated magnetic field along the path followed by the track and is calculated using MC simulations taking the magnetic field map of LHCb detector into account.

The ideal position to align the two track segments in a straight-line model fluctuates based on the momentum of the track. Hence, the z-position of the kink, termed z_x^{match}, is not static but rather includes the parameterisation of the magnetic field:

$$\begin{aligned} z_x^{\text{match}} &= A_x + B_x \times |\Delta t_x| + C_x \times |\Delta t_x|^2 \\ &+ D_x \times |x_{\text{SciFi}}^{z_{\text{ref}}}| + E_x \times (t_x^{\text{velo}})^2 \end{aligned} \tag{4.5}$$

In Eq. 4.5 the first-order parameterisation depends on the slope difference between the two track segments, $|\Delta t_x|$. In the last term, t_x^{velo} refers to the

slope of the *VELO* track. Additional terms take into account the leftover field beyond the magnet. The parameters A_x, B_x, C_x, D_x, and E_x are obtained by fitting simulated samples. The values are displayed in Table 2.

A_x	5287.6 mm
B_x	−7.988 78 mm
C_x	317.683 mm
D_x	0.011 937 9
E_x	−1418.42 mm

Table 2: Magnetic field parameterisation values obtained from the minimisation process.

Best candidates are selected using χ^2 minimisation in Δx, Δy and Δt_y. Further, a clone killing procedure is applied wherein the tracks which share a *VELO* track are compared, and only the ones with best χ^2 are kept. This produces *long* tracks as the output.

4.3.3 GPU implementation of the algorithm

Some key design considerations in the algorithm development are described below.

- GPU memory access: the choice of the GPU memory type for a given operation can make a significant difference in its performance. In the SciFi seeding algorithm, SciFi hits are read multiple times in a random (non-coalesced) order and therefore the shared memory storage is preferred as compared to global memory. Each hit can be stored in a single float of 4 bytes and in the SciFi seeding algorithm we only load hits from 6 (x or uv) layers at a time. Thus if we take 300 hits as nominal for each layer then we have a total of 1800 hits in total which in turn require 7.2 kB of shared memory. The binary search implementation for making pair candidates from hits is about 1.5 times faster using shared memory over global memory.

- Parallelisation: in the `Seeding_XZ` part, all the operations of creating two-hit, three-hit and multi-hit combinations are parallelised over the hits of the starting layer. In the `Seeding_confirmTracks` part, all the operations of adding *uv* hits are done in parallel for each starting *xz* seed from the previous step. The `VELO-SciFi Matching` algorithm is parallelised over the *VELO* tracks while matching with the SciFi track seeds.

- *Thread block* size optimisation: from the GPU hardware architecture perspective, a set of 32 threads that execute the same instructions are grouped into *warps*. A thread block is made up of several warps (maximum 64 for our nominal cards, the RTX A5000). A group of thread blocks is assigned to a SM. Optimal thread block size configuration is important to maximize the occupancy of the GPU and depends on the type of the GPU and its architecture. The `Seeding_XZ` and `Seeding_confirmTracks` parts require about 8 KB of shared memory per block to store the hits. This can be best met with 4 warps per block which gives a total of 128 threads. For `VELO-SciFi Matching` the block size is kept at 32.

4.3.4 Figures of merit

In this section, the performance of the `HybridSeeding` and `VELO-SciFi Matching` algorithms are presented. Several figures of merit are used to quantify the performance of the algorithms. In terms of physics results, the reconstruction efficiency and the ghost rates are key to optimize the output results. In terms of computing parameters, the throughput is the measurement of how fast the information can be processed by the algorithm. These performance indicators are described below.

Reconstruction efficiency

Reconstruction efficiency is defined as the ratio of the number of tracks accurately identified (*reconstructed*) by the algorithm to the total number of tracks that could have been identified (*reconstructible*). To quantify the reconstruction efficiency of the algorithm, simulated samples of various

decay channels of interest are used. When simulating collision data, tracks meeting certain thresholds are defined to be reconstructible and have an assigned type according to the subdetector reconstructibility. This is based on the existence of reconstructed detector digits or clusters in the emulated detector, which are matched to simulated particles if the detector hits they originated from are properly linked [84]. One can distinguish reconstructibility in the following subdetectors:

- VELO: at least 3 pixel sensors with at least 1 digit each.
- UT: at least 2 clusters where 1 cluster has to be in layer one or two and 1 in layer three or four. The clusters can be x or stereo clusters.
- SciFi: at least 1 x cluster and 1 stereo cluster in each of the 3 SciFi stations.

Requirements for reconstructible *long* tracks implies VELO and SciFi reconstructibility, *downstream* tracks must satisfy the UT and SciFi reconstructibility, and T-tracks only requires the SciFi one. The efficiency is then defined as:

$$\epsilon = \frac{N_{\text{reconstructed tracks}}}{N_{\text{reconstructible tracks}}} \tag{4.6}$$

Ghost Rate

A ghost track is defined as a track which, after the reconstruction process, cannot be associated with any simulated particle. This indicates that the track is likely a false positive. The prevalence of ghost tracks in the data is quantified using the *ghost fraction*. It is defined as:

$$\textit{Ghost fraction} = \frac{N_{\text{ghost tracks}}}{N_{\text{tracks}}}. \tag{4.7}$$

where $N_{\text{ghost tracks}}$ is the number of ghost tracks and N_{tracks} is the total number of tracks in the data sample.

Another term that is pertinent in the context of track reconstruction is a *clone*. A clone is a track that has been successfully matched to a simulated particle but is not unique in this association. Specifically, if the simulated particle is associated with at least one other reconstructed track, then the track is termed as a clone. This implies that there may be redundant or overlapping information associated with such tracks.

Usually physics efficiency and ghost rates are represented as function of physical variables such as momentum (p), transverse momentum (p_T), pseudorapidity (η), and the number of primary vertices nPV. A proper behaviour in all the range of the variables is key to evaluate if the algorithm works well.

The results shown in the following are produced with the standard LHCb physics performance checking tools. Tracks are matched with simulated particles if at least 70% of the track hits are common. Several samples of simulated sample are used to perform these studies, with have all the information of the detector. They are:

- $B_s^0 \to \phi\phi$ decays: 10000 event sample, where the final decay products are four kaons produced promptly ($\phi \to K^+K^-$). It is used as nominal sample for the evaluation of the physics performance of almost all Allen algorithms. They are usually *long* tracks with high momentum.

- $B^0 \to K^{*0}e^+e^-$ decays: 10000 event sample, where the final decay product are two electrons of opposite sign, one kaon and one pion ($K^{*0} \to K^+\pi^-$). It is the control channel to evaluate the performance on electrons.

- MinBias sample: 10000 MinBias events corresponding to the running conditions expected in Run3. These events are not expected to be triggered since they do not have events of interest, but they are useful to calculate the computing performance of the algorithm, as they are more representative of the average conditions the algorithm is operating on, together with the trigger rates.

The efficiency of the `HybridSeeding` and `Matching` track reconstruction is depicted in terms of p, p_T, and η in Fig. 45. The track reconstruction efficiency is displayed in relation to reconstructible non-electron tracks, which are produced from B decays and pass through the VELO, UT, and SciFi detectors. These tracks fall within the pseudorapidity coverage of the LHCb detector, which ranges from $2 < \eta < 5$. The global efficiency is more than 80% for both algorithms, and around 60% for low momentum tracks.

Ghost rates for HLT1 *long* tracks (Sec. 4.2.2) reconstructed with the `Matching` algorithm, which are derived from b-decays, are also presented as a function of p, p_T, η in Fig. 46. These plots illustrate tracks that have been reconstructed from the simulated $B_s^0 \to \phi\phi$ decays. The ghost rate provided by the algorithm is below 5%.

Momentum resolution

The momentum resolution for HLT1 *long* tracks, originating from b-decays and reconstructed by the `Matching` algorithm is presented as a function of momentum p in Fig. 47. Simulated $B_s^0 \to \phi\phi$ events are used. The resolution is below 1% for most of the tracks.

Throughput

The computational throughput is evaluated using simulated MinBias samples, as they offer the most accurate representation of the average occupancy conditions expected in Run3. Fig. 48 shows the comparison of the new `Matching` algorithm and the default `Forward` one, when running in the HLT1 sequence using different cards. Both algorithms have same performance. The throughput at this point, which includes the selection of different trigger lines, is around 125 kHz. This perfectly fits with the HLT1 requirements. Figure 49 shows the breakdown of the algorithms for a Nvidia RTX A5000 GPU card. As expected, the SciFi decoding and seeding is the most consuming part, due to the high occupancy of the SciFi. The `Matching` algorithm is fast and only consumes 3% of the resources.

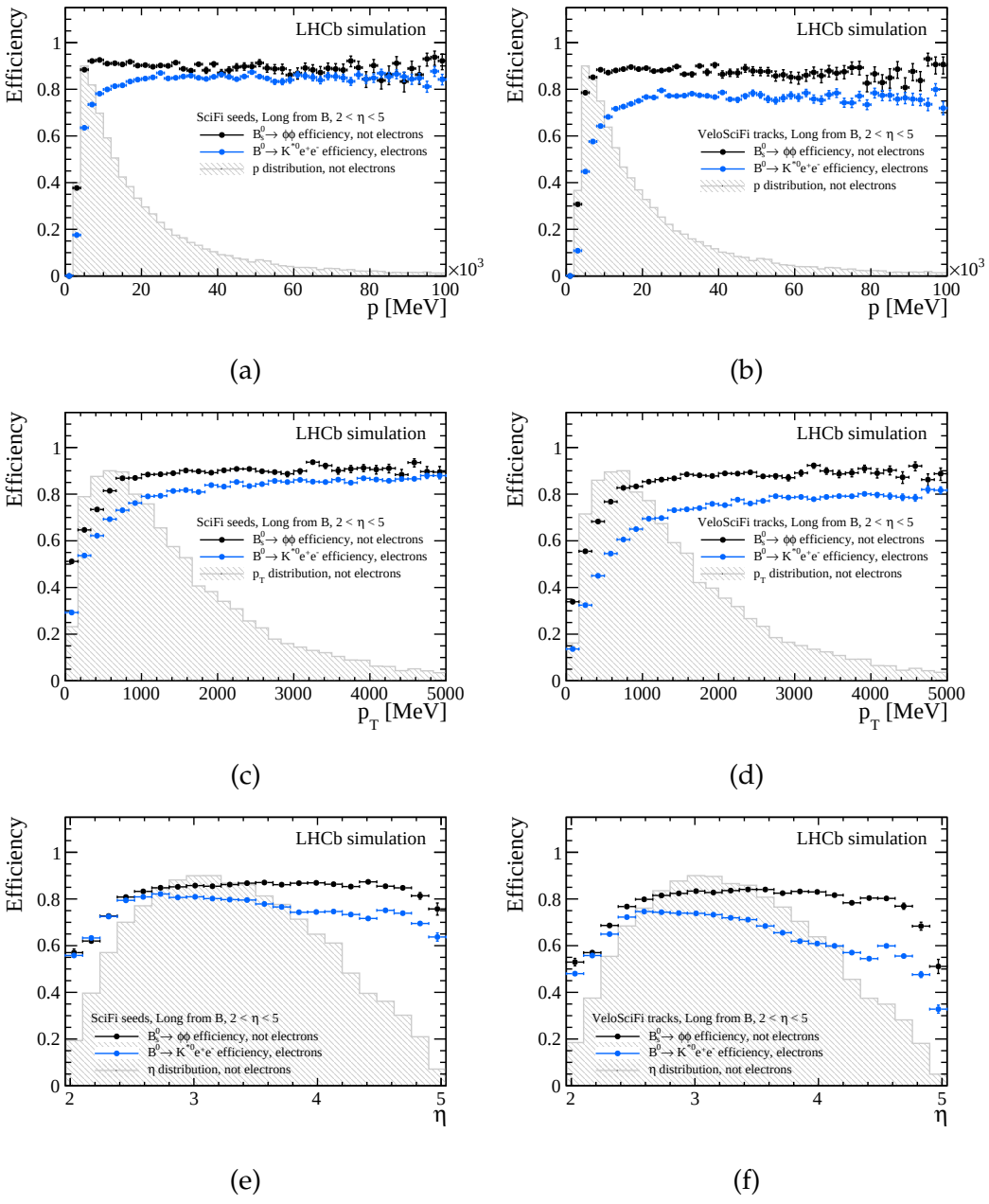
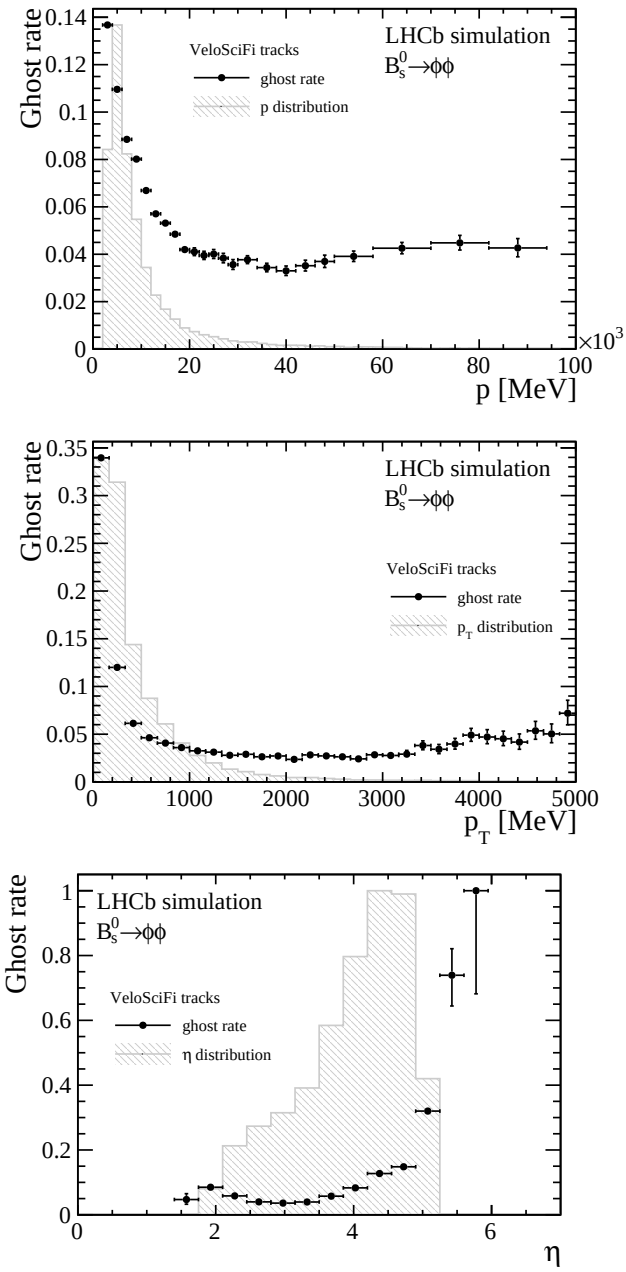


Figure 45: Efficiency for seeds reconstructed by the `HybridSeeding` algorithm (left), and *long* tracks reconstructed using the `Matching` algorithm (right).



b

Figure 46: Ghost rates for HLT1 *long* tracks provided by the `Matching` algorithm. $B_s^0 \to \phi\phi$ decays are analysed.

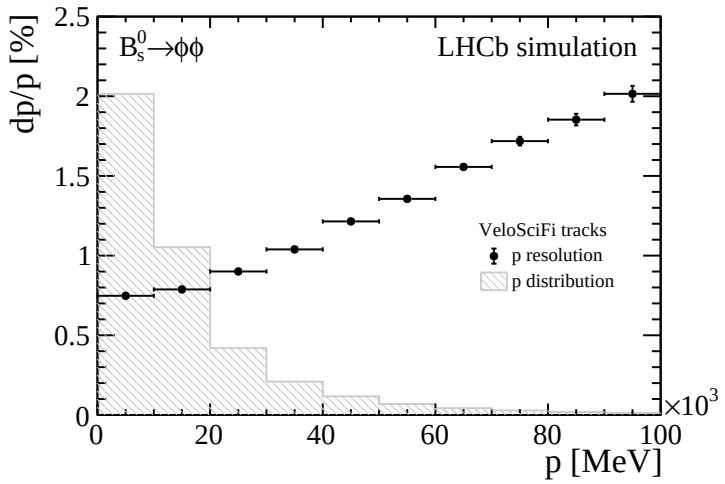


Figure 47: Momentum resolution for HLT1 *long* tracks of charged particles, originating from *b*-decays and reconstructed by the `Matching` algorithm.

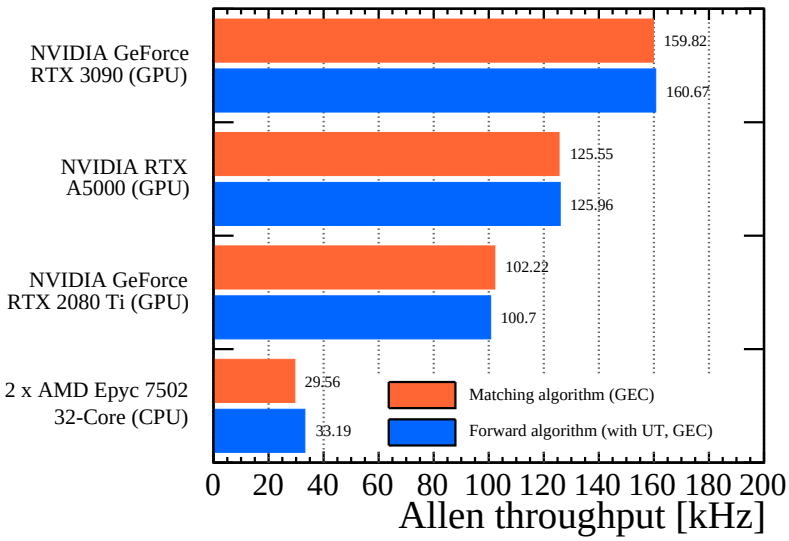


Figure 48: Throughput comparison between the `HybridSeeding` & `Matching` algorithms (highlighted in orange) and the default `Forward` (paired with UT, illustrated in blue) for the HLT1 sequence across different GPU cards.

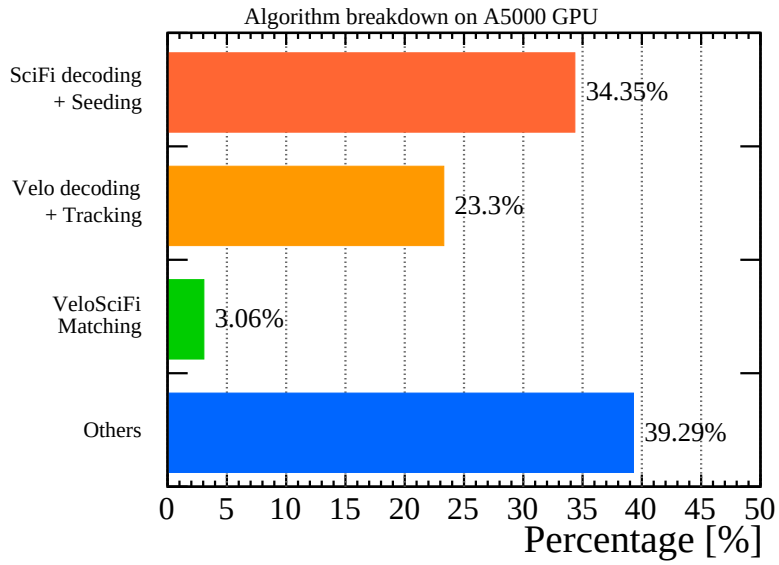


Figure 49: Breakdown of the `Matching` algorithm for the Nvidia RTX A5000 GPU card.

5 The Downstream track reconstruction algorithm at HLT1

This chapter delves into the development of the Downstream tracking algorithm for long-lived particles, implemented inside the Allen framework. Significantly, this marks the first comprehensive implementation of this algorithm at the HLT1 level[1]. The chapter examines the underlying physics motivation (Sec. 5.1.1), design considerations (Sec. 5.3), GPU implementation specifics (Sec. 5.5), and evaluates the algorithm robustness through performance assessments related to physics output and throughput (Sec. 5.6). A ghost rejection Neural Network (NN) developed as part of the Downstream algorithm is explained in Sec. 5.4. For an insight into the broader implications of this algorithm, the reader is referred to the ensuing Chapter 7. This work was carried out in collaboration with Jiahui Zhuo.

5.1 Introduction

If an event fails to meet the selection criteria set by the HLT1, it is immediately discarded. Owing to the real-time nature of the decision-making process and high throughput, events that do not get triggered at HLT1 level are neither stored nor buffered for later analysis. As such, these events are irretrievably lost.

As described in Sec. 4.2, tracks in LHCb are classified into various types based on the subdetectors that participate in their reconstruction.

[1] merge request https://gitlab.cern.ch/lhcb/Allen/-/merge_requests/1095

The introduction of this new Downstream algorithm represents a notable advancement in the LHCb HLT1 trigger fidelity and is poised to make substantial contributions to the study of LLPs.

5.1.1 Physics motivation and challenges

LLPs, which have been elaborated upon in Chapter 1, can be key in the search for new physics discoveries. The decay vertex of these LLPs is notably displaced from the primary vertex, and for large masses of the LLPs the decay products are tightly collimated. Consequently, these particles are not reconstructed as tracks in the VELO. The principal approach for investigating such particles has conventionally revolved around tracing displaced vertices and patterns in the inner detectors, and moving outwards while correlating these tracks with signatures in the outer trackers, muon and calorimeter systems, forming *long* tracks. In instances where LLPs does not give rise to *long* tracks, the trigger efficiencies experience a steep decline, plummeting to mere single-digit percentages. This reduction poses significant constraints on the scope of the physics investigations. This scenario underscores the need for the development of innovative strategies for efficient tracking of LLPs.

Algorithms from the HLT2 stage, namely PatSeeding and PatLongLivedTracking [85], were analyzed to understand the challenges of integrating the Downstream algorithm at the HLT1 stage. Compared to these CPU-based HLT2 algorithms, the throughput of the HLT1 algorithm needed to be two orders of magnitude higher.

5.2 The *downstream* track model

The track model takes inspiration from the HLT2 PatLongLivedTracking algorithm [85]. The algorithm's initial parameter values served as a foundation for the HLT1 Downstream tracking algorithm. Comprehensive studies were conducted with up-to-date simulations of $B_s^0 \to \phi\phi$ samples, reflecting the Run3 conditions, to optimize the track model parameterisation. The subsequent sections provide details of this optimisation process.

5.2.1 Particle movement through magnetic field

When analyzing the motion of particles moving through tracking stations positioned external to LHCb's magnetic field, one typically ignores multiple scattering[2]. Since this phenomenon does not have a significant effect on the trajectory of high-energy particles in this context, it is neglected to simplify calculations.

Particles follow straight-line paths outside the magnetic field, whereas within the LHCb dipole magnetic field, they curve in the bending plane of the LHCb magnetic field. This deviation can be modeled as a sharp change in direction, akin to a *kink*, at a specific position along the z-axis. This position is termed as the "magnet point" and is denoted as z_{Magnet}, as illustrated in Fig. 50. The layout of the coordinate system with track types, highlighting the *kink* hypothesis in *downstream* reconstruction is shown.

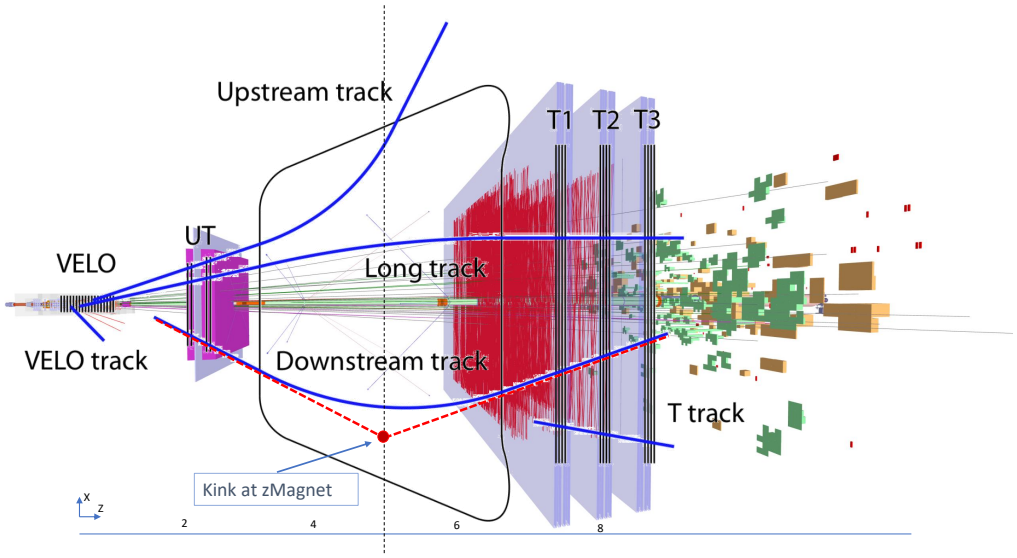


Figure 50: Schematic representation of the LHCb tracking system. A *downstream* track is shown as a blue line, while its linear approximation outside the magnetic field is depicted as a red dotted line along with the *kink* represented as red dot.

z_{Magnet} can be expressed as a combination of track parameters repre-

[2]Multiple scattering is a sequence of Coulomb scatterings that lead to a change of the flight direction of the particle.

sented in Eq. 5.1,

$$z_{\text{Magnet}} = \alpha_0 + \alpha_1 \cdot t_y^2 + \alpha_2 \cdot t_x^2 + \alpha_3 \cdot \frac{q}{p} + \alpha_4 \cdot |x_{\text{SciFi}}| + \alpha_5 \cdot |y_{\text{SciFi}}| + \alpha_6 \cdot |t_y| + \alpha_7 \cdot |t_x|. \quad (5.1)$$

Here, t_x and t_y are the slopes of the final state of the seeds in the 3rd (last) SciFi stations along the x and y, respectively (see Sec. 4.2 for the track states definition). The x_{SciFi} and y_{SciFi} denote the x and y coordinates of the last state within the SciFi stations, respectively. The q/p is the initial momentum estimate input from the SciFi seed.

Here, the α_i $(\alpha_0, \alpha_1, \ldots, \alpha_7)$ are coefficients that weight the contribution of each corresponding variables to the calculation of z_{Magnet}. These coefficients are obtained by performing a fit to simulated $B_s^0 \to \phi\phi$ sample. The set of coefficients (α_i) that minimizes the difference between the predicted z_{Magnet} and observed z_{Magnet} values for the given sample of tracks is obtained using the least squares method where the sum of the squared differences is minimised. The obtained values are quoted in Table 3.

Coefficient	Value
α_0	5367.359 mm
α_1	-2660.298 mm
α_2	325.387 mm
α_3	-4985.715 mm MeV/c
α_4	-0.026343
α_5	-0.066487
α_6	724.1572 mm
α_7	148.2586 mm

Table 3: Values of coefficients for the Eq. 5.1

These values of the coefficients are then fixed to calculate the z_{Magnet} in the Downstream algorithm.

A scatter plot of the predicted z_{Magnet} values versus the bias (the difference between the predicted and *true* values) is shown in Fig. 51 which provides an assessment of the model performance. On the x-axis

are the predicted z_{Magnet} values, and on the y-axis are the bias. Each point on the plot represents a particle track. The spread of the points around the $y = 0$ line indicates that there is no dependence of the bias with z_{Magnet}.

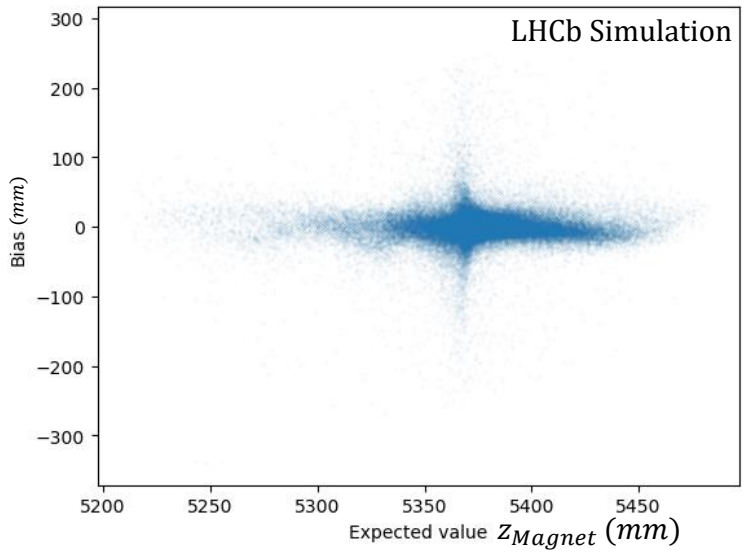


Figure 51: Distribution of predicted z_{Magnet} values vs the bias (the difference between the predicted and actual values)

After determining z_{Magnet}, the x and y positions of the magnet point, denoted as x_{Magnet} and y_{Magnet} respectively, are determined using following expressions:

$$x_{\text{Magnet}} = x_{\text{SciFi}} + t_{x_{\text{SciFi}}} \cdot (z_{\text{Magnet}} - z_{\text{SciFi}}). \tag{5.2}$$

$$y_{\text{Magnet}} = (y_{\text{SciFi}} + dy) + t_{y_{\text{Magnet}}} \cdot (z_{\text{Magnet}} - z_{\text{SciFi}}). \tag{5.3}$$

where

$$t_{y_{\text{Magnet}}} = t_{y_{\text{SciFi}}} + dt_y. \tag{5.4}$$

It is to be noted here that in the SciFi stations, the y information is extracted from the 5° tilt of stereo layers and is not as accurate as the x information. This can introduce bias in the y and t_y information. This bias can be corrected while calculated y_{Magnet}. In essence, the computation of y_{Magnet} involves extrapolating the y position of the particle in the SciFi stations to the magnet, where these bias corrections can be taken

into account. These corrections can be parameterised using simulation samples and are based on the differences between (a) *true* position in y vs the extrapolated position at y_{Magnet} denoted as dy and (b) the particle *true* slope and its extrapolated slopes at the magnet dt_y.

Here, dy and dt_y are the special extrapolation corrections in y_{Magnet} calculated using 5.5 and 5.6 where the coefficients are parameterised using the similar linear regression technique used for z_{Magnet}

$$dy = \beta_0 + \beta_1 \cdot y_{\text{SciFi}} + \beta_2 \cdot t_{y_{\text{SciFi}}} + \beta_3 \cdot q/p. \tag{5.5}$$

$$dt_y = \gamma_0 + \gamma_1 \cdot y_{\text{SciFi}} + \gamma_2 \cdot t_{y_{\text{SciFi}}} + \gamma_3 \cdot q/p. \tag{5.6}$$

The values of these coefficients are obtained by performing a fit to a simulated $B_s^0 \to \phi\phi$ sample and these values are summarised in the Table 4 and Table 5.

Function	β_0(mm)	β_1	β_2(mm)	β_3 mm MeV/c
dy	-0.5130062	0.1409223	-1470.980 mm	-231728.1

Table 4: Coefficients for the function dy.

Function	γ_0(mm)	γ_1	γ_2(mm)	γ_3(mm MeV/c)
dt_y	-0.00005468	-0.00006705	-0.63219	-34.076

Table 5: Coefficients for the function dt_y.

5.2.2 Momentum estimation

The momentum of a *downstream* track is predominantly influenced by the deviation it experiences within the magnetic field, commonly referred to as the kink. Moreover, the slopes in the x and y directions also have a substantial impact. A parameterised model is employed to estimate the momentum, and is represented by the following equation:

$$q/p = \frac{\Delta_{\text{slope}}}{\eta_0 + \eta_1 \cdot t_x^2 + \eta_2 \cdot t_y^2} \cdot \text{magnet\_polarity} \tag{5.7}$$

where:

- q/p: represents the estimated charge over momentum of a *downstream* track.
- η_0, η_1, η_2: serve as weighted factors that control the contribution of the square of the slope in each direction as well as the base term (η_0) to the estimated momentum. These are empirically determined by performing a fit to simulated $B_s^0 \to \phi\phi$ sample and the values are summarised in the Table 6.
- t_x, t_y: are slopes and t_x^2, t_y^2 terms ensure unit consistency and account for quadratic dependencies.
- Δ_{slope}: represents the variation in slope. It is indicative of the changes in the particle trajectory as it navigates through the magnetic field.

Coefficient	η_0	η_1	η_2
Value	1217.77 MeV/c	454.598 MeV/c	3353.39 MeV/c

Table 6: Values of Coefficients for the Eq. 5.7

5.2.3 First slope estimation and corrections

The slope $t_{x,\text{UT}}$ is essentially the change in the x position of the particle with respect to the change in the z position from magnet point to the UT stations as shown in Fig. 52.

Sketch showing the first slope is calculated using Eq. 5.8

$$\text{First\_}t_{xUT} = \frac{x_{\text{Magnet}}}{z_{\text{Magnet}}} + dt_x, \tag{5.8}$$

wherein the correction to the first slope dt_x is determined by Eq. 5.9:

$$dt_x = \alpha_0 + \alpha_1 \cdot t_{y_{\text{SciFi}}} + \alpha_2 \cdot q/p. \tag{5.9}$$

Here, the coefficients are empirically determined in the similar way as explained earlier by performing a fit to simulated $B_s^0 \to \phi\phi$ sample and the values are summarised in Table 7.

A scatter plot of the First_t_{xUT} values versus the bias (the difference between the calculated first slope and *true* values) is shown in Fig. 53.

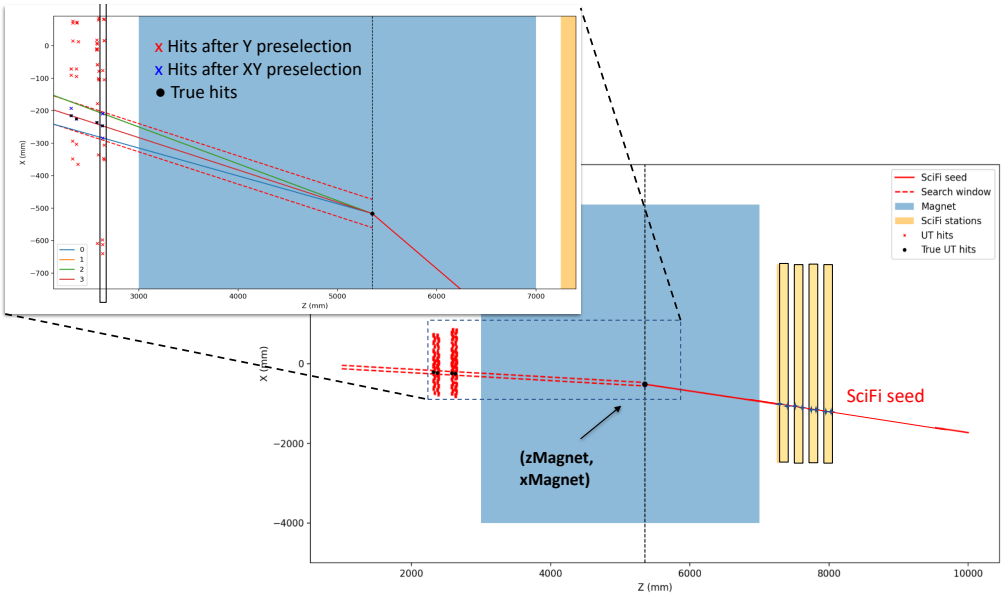


Figure 52: Illustration showing the used hits in the last UT layer (UTbX) and Magnet Points (x_{Magnet}, z_{Magnet}) to find the first slope and correction to the first slope $t_{x,\text{UT}}$.

Function	$\alpha_0(\text{mm})$	α_1	$\alpha_2(\text{mm MeV}/c)$
(dt_x)	0.00002247261	−0.0000868066	8.003155

Table 7: Coefficients for the functions dt_x in Eq. 5.9.

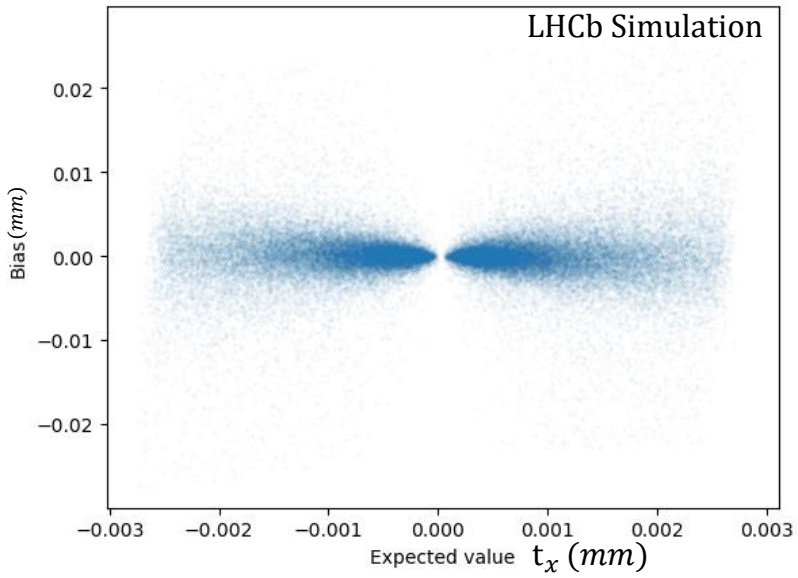


Figure 53: Distribution of estimated $First\_t_{xUT}$ values vs the bias (the difference between the estimated and *true* values).

5.2.4 Calculation of tolerance windows

The tolerance windows are used to account for how much the predicted position of the hit can deviate from its *true* position. Fig. 52 shows a sketch of the tolerance windows.

Tolerance Window calculation for each UT layers

The tolerance windows for the UT stations are calculated in the following way:

1. First slope $t_{x,\mathrm{UT}}$: the first slope $t_{x,\mathrm{UT}}$ is estimated as per Sec. 5.2.3.
2. Estimating y_{Magnet} and special correction in dy estimated from Eq. 5.3 and Eq. 5.5, respectively.
3. Estimating Expected Positions at $layer_i$: They are obtained from:

$$y_{\mathrm{layer}_i} = y_{\mathrm{Magnet}} + t_y \times (z_{\mathrm{layer}_i} - z_{\mathrm{Magnet}}), \tag{5.10}$$

$$x_{\mathrm{layer}_i} = x_{\mathrm{Magnet}} + t_x \times (z_{\mathrm{layer}_i} - z_{\mathrm{Magnet}}). \tag{5.11}$$

4. Threshold estimation: the tolerances are plotted against the absolute value of the expected q/p, and percentiles are computed to find the values of thresholds at which (98%) of the data lies below this threshold. See Fig. 54.
5. For X layers iė $UTbX$ and $UTaX$, a linear model (see Eq. 5.12) is used for fitting the thresholds as a function of the absolute value of the expected q/p.

$$T(layer_i) = \alpha_0 + \alpha_1 \cdot |q/p| \tag{5.12}$$

6. For UV layers iė $UTbV$ and $UTaU$, a quadratic model of the form Eq. 5.13 is fitted to the thresholds as a function of the absolute value of the expected q/p.

$$T(layer_i) = \alpha_0 + \alpha_1 \cdot |q/p| + \alpha_2 \cdot |(q/p)^2| \tag{5.13}$$

   Scatter plots of fitting these tolerance windows are shown in Fig. 54. The fitting curves can easily diverge for low p, therefore, we have to set a maximum value of tolerance which corresponds to 98% of the hits. The obtained values are summarised in Table 8.

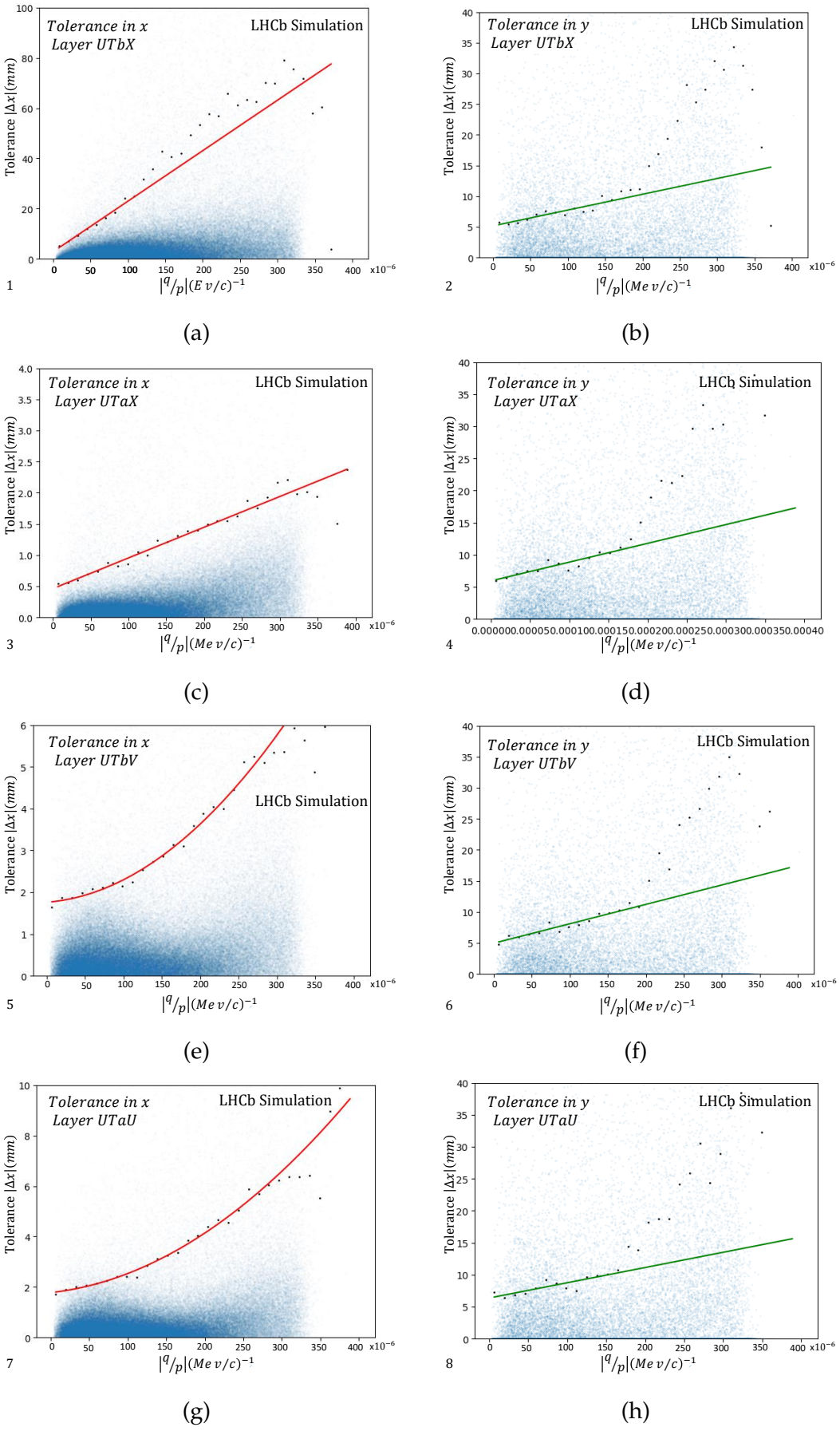


Figure 54: Distribution of tolerance window fits for different layers of UT for x and y.

Layer	Tolerance windows	Max Tol.(mm)
$UTbX_x$	$2.836915 + (2.011148 \times 10^5) \cdot \lvert q/p \rvert$	57.71
$UTbX_y$	$5.166317 + (2.575863 \times 10^4) \cdot \lvert q/p \rvert$	21.28
$UTaX_x$	$0.4622822 + (4.932216 \times 10^3) \cdot \lvert q/p \rvert$	1.92
$UTaX_y$	$5.932930 + (2.926494 \times 10^4) \cdot \lvert q/p \rvert$	23.72
$UTbV_x$	$1.7605 + (3.9739 \times 10^7) \cdot \lvert (q/p)^2 \rvert + (1400.553) \cdot \lvert q/p \rvert$	4.35
$UTbV_y$	$(3.1283 \times 10^4) \cdot \lvert q/p \rvert + 4.971600$	21.44
$UTbU_x$	$1.7754 + (4.2299 \times 10^7) \cdot \lvert (q/p)^2 \rvert + (3339.328) \cdot \lvert q/p \rvert$	4.93
$UTbU_y$	$6.367733 + (2.384459 \times 10^4) \cdot \lvert q/p \rvert$	22.86

Table 8: Maximum values of tolerance windows for different UT layers.

5.3 Algorithm design

A full skeleton of the algorithm is presented in the flow charts from Fig. 55 to Fig. 60. To take advantage of the underlying GPU architecture, the algorithm is implemented using multiple GPU kernel functions, parallelising different processes at various stages. The algorithm implementation can be divided into four major functional blocks:

1. Preparing inputs from the first UT layer and creating the output table, as shown in Fig. 55.

2. Filling the output table with candidate hits from the remaining three layers of the UT, as shown in Fig. 57.

3. Clone killing and ghost removal and confirmation of *downstream* tracks, as shown in Fig. 59.

4. Preparing the output of the *downstream* tracks for further processing as shown in Fig. 60.

5.3.1 Preparing inputs

The main inputs to the algorithm are the SciFi seeds from the `Hybrid-Seeding` algorithms (see Sec. 4.3.1) and UT information after the decoding stage described in Sec. 4.1.2. The first step is to filter out the used SciFi seeds which have been matched with the VELO tracks by `VELO-SciFi`

Matching (Sec. 4.3.2). In this way, only the *unmatched* SciFi seeds without any possible attribution to *VELO*, *Upstream* or *long* tracks are considered. The possibility to filter out used UT hits was also explored but found to make no difference in the throughout or performance, thus it has been kept as a optional feature and not part of the default algorithm.

Preparation of UT hits

UT hits, along with the sector information, are cached in the shared memory for faster access and operations. The UT data are organised in the standard LHCb format (see Sec. 4.1.2). The decoded data frame of the UT sub-detector (Sec. 4.1.2) contains LHCbID, xAtYEq0, zAtYEq0, yMax, yMin, and weight, for each hit.

Selection of SciFi tracks

The seeds which are not matched, called *unmatched*, are carried forward and cached in the shared memory which are typically around 70% of the total SciFi seeds. The quality of these leftover seeds, has a direct impact on the performance of the Downstream algorithm. These *unmatched* SciFi seeds contain more than 20% ghosts. To reduce this ghost rate, various filtering techniques and discrimination potential of different variables of seed quality and kinematics were checked. Distributions of some of the variables that were investigated using simulated MinBias samples are shown in Fig. 56.

In the end, a single ghost rejection neural network (NN) was developed which provided a unified control at the last stage of the algorithm. This has been explained in detail in Sec. 5.3.3 and Sec. 5.4. The SciFi minimal track state output for reconstructed SciFi seeds provides the following information: q/p, t_x, t_y, x, y, z, and χ^2.

First extrapolation to last UT layer

For each SciFi seed in parallel, the first extrapolation to the last UT layer $UTbX$ through the magnetic field is performed by approximating the bending of the trajectory inside the magnet using *kink* (see Sec. 5.2.3) in the flight path at a given z position called z_{Magnet}.

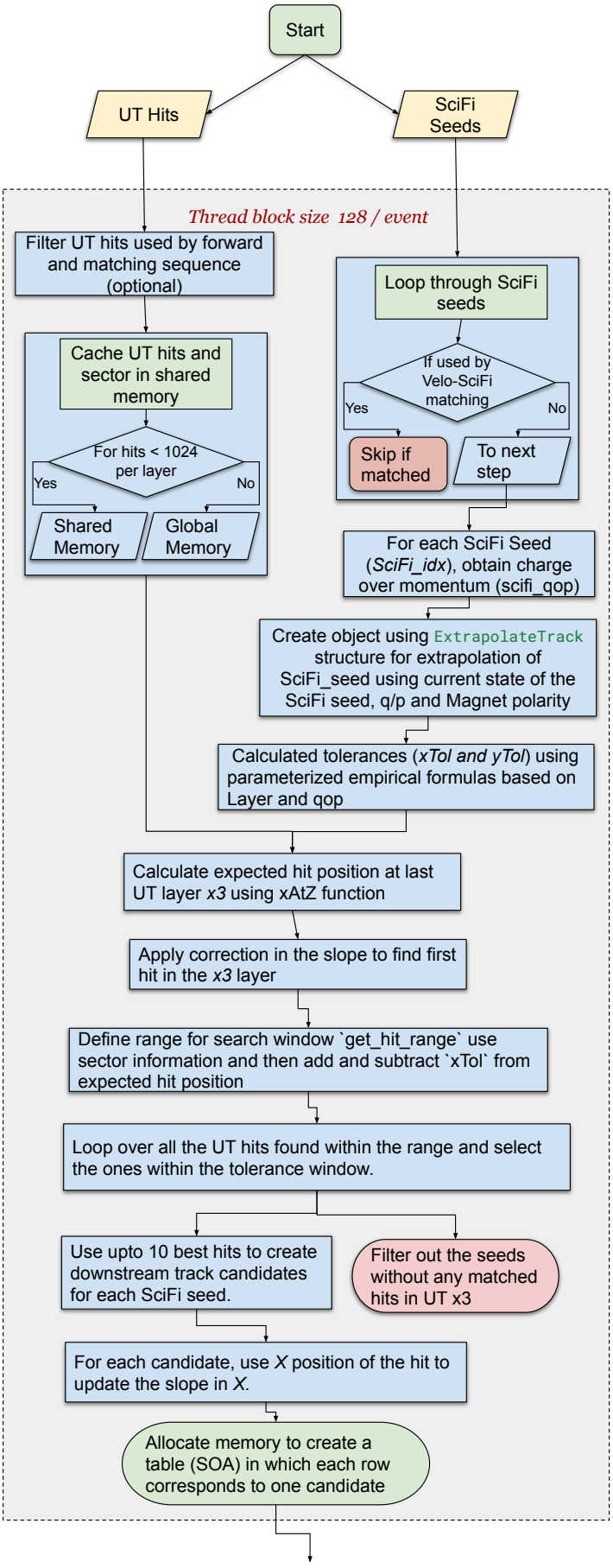
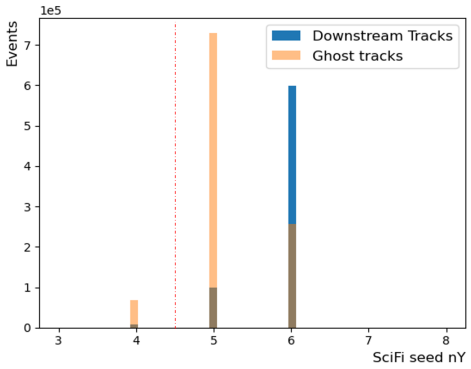
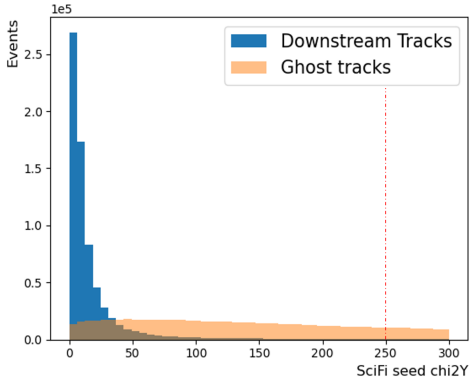


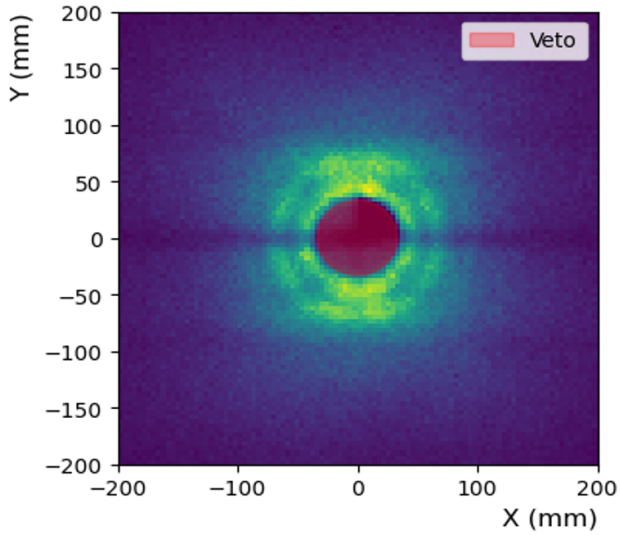
Figure 55: Flow chart of the `Downstream` algorithm (Part 1): Create output table.



(a) Distribution of *unmatched* SciFi tracks vs the number of hits in UV stations (nY) for true *downstream* tracks and ghost tracks.



(b) Distribution of *unmatched* SciFi tracks vs the track χ^2 distribution in Y, for true *downstream* tracks vs ghost tracks.



(c) Distribution of SciFi tracks around the beam pipe in the center which are removed by applying veto.

Figure 56: Distribution of SciFi *unmatched* tracks and ghost tracks for several variables, using simulated Minimum Bias samples.

The state of the current SciFi seed, its q/p, and magnet polarity are fed into the parameterised extrapolation function obtained in Eq. 5.1 to obtain the first estimate of z_{Magnet}. Using this z_{Magnet} position, the first slope of the seed in x, $t_{x,\text{UT}}$ and in y, t_{yUT} is determined as discussed in detail in Sec. 5.2.3.

After obtaining the expected hit position in the last UT layer $UTbX_x$ and $UTbX_y$, the tolerance windows are calculated as described in Sec. 5.2.4. These pre-calculated tolerances are then used to calculate search windows for the hits based on the position of a layer in x and y using the expression:

$$\text{search\_range}_x = [UTbX_x - x_{\text{Tol}}, UTbX_x + x_{\text{Tol}}], \tag{5.14}$$

and

$$\text{search\_range}_y = [UTbX_y - y_{\text{Tol}}, UTbX_y + y_{\text{Tol}}], \tag{5.15}$$

where x_{Tol} and y_{Tol} are the tolerances for the layer and $UTbX_x$ $UTbX_y$ are the expected positions of the layer in x and y based on the extrapolated state of the seed. The search windows are then used to filter out the hits which are not in the range.

Based on these criteria, up to 10 best hits are selected as candidates for *downstream* tracks, and the seeds without any matched hits in $UTbX$ are skipped. For each candidate, the x position of the hit in $UTbX$ is used to update the slope $t_{x(UT)}$. This updated $t_{x(UT)}$ is then used to update the q/p estimation of each candidate.

At the end of this kernel function, a SOA is used to store the selected caldidates, wherein each row corresponds to one candidate. This then enables the execution of memory-coalesced operations in parallel during subsequent steps

5.3.2 Searching hits in remaining UT layers

The SOA table created in the previous step is used as input to this kernel function. For each row of the table, the remaining UT layers $UTaX$, $UTaU$, and $UVbV$ are searched in parallel for each candidate.

Similar to the last x layer $UTbX$, the sizes of the tolerance windows for remaining UT layers are computed using the respective new slope

and q/p values of each candidate using the parameterised functions as described in Sec. 5.2.4. These pre-calculated tolerances are then used to calculate search windows for the hits based on the position of a layer in the same way as for the last layer.

Searching hits in *UTaX*

For each candidate, hits within the search_range are obtained using tolerance windows $UTaX_x$ and $UTaX_y$ as in Eq. 5.14 and Eq. 5.15. Score of the candidate is calculated based on their distance from expected hit position in X.

Searching hits in *UTaU* and *UTbV* layers

Similarly if the layer is UV *i.e.* *UTaU* or *UTbV*, the expected X and Y positions are calculated based on the existing slope estimate of the candidates and all the hits within the tolerance window are selected. For each candidate, score based on the distances from expected hit position in X and Y is calculated and in order to account for bias in t_y as shown in Fig. 58, up to two best hits are stored for each UV layer. Looping over all the candidates, a check is applied which requires each candidate to have at least one hit in each UT layer and the candidates which do not have matching hits in all the layers are assigned a score of infinity which is used to filter them out.

Find best combination of hits and compute score

At this stage, the algorithm SOA contains hit combinations from all the UT layers. For each candidate, the best combination of hits is selected based on the score of the candidate. At the end of this kernel function the best combination of hits for each candidate is selected using $score = |dist_u{}^2 + dist_v{}^2|$ and the score of the candidate is updated in the output table.

5.3.3 Creating the track `downstream_create_track`

The track candidates created in the last step contains a large number of clones and ghosts. To remove these clones and ghosts, the kernel function

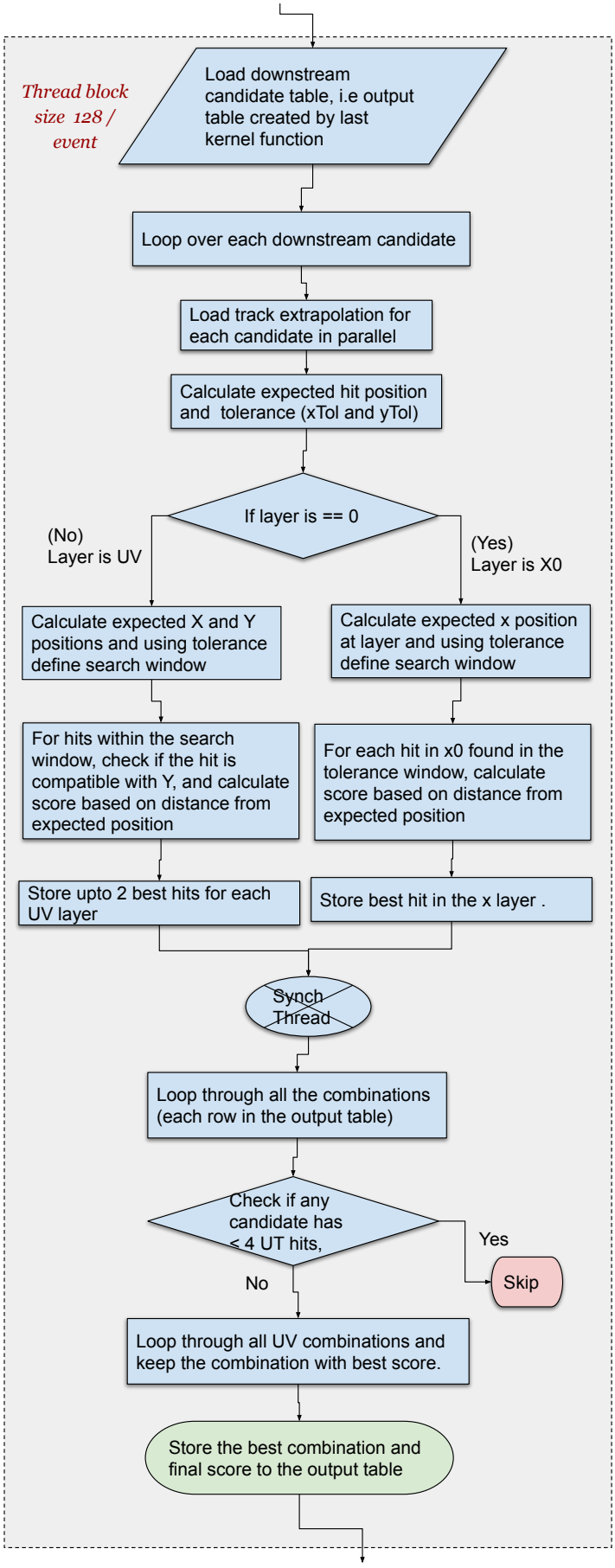


Figure 57: Flow chart of the `Downstream` algorithm (Part 2): Fill output table with candidates.

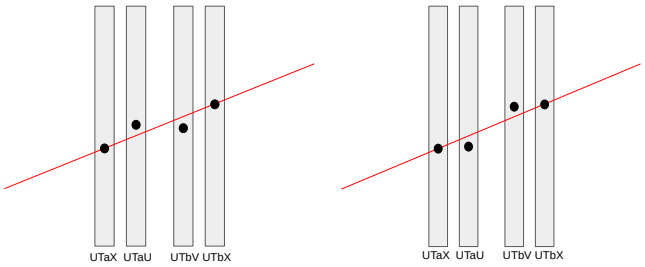


Figure 58: Bias in t_y for *downstream* tracks.

`downstream_create_track` is used. The input to this kernel function is the output table from the previous step.

Clone killing and ghost rejection

The *clone killing* step is necessary to remove duplicate tracks that may have been created due to multiple hits being associated with the same track. First the hits associated with each candidate track are cached in shared memory. Then iterating over the first and the last layer of the UT detector, a nested loop is used to compare each candidate track to all other candidate tracks in the same shared memory block. For each pair of candidate tracks, if they share a hit then their scores which were computed earlier are compared, and the candidate tracks with the lower score are killed. The clone killing step uses the number of shared memory arrays of each candidate track to store the hits, scores, and killed status. This allows for efficient parallel computation on the GPU, as each *thread* can operate on a different candidate track simultaneously. After the clone removal, which is typically about 1% of the total candidates, the remaining candidates are carried forward to the next step which employs a ghost rejection NN. The details are described in Sec. 5.4 as shown in the flowchart 59.

Output of the *ghost killer* provides a score for each track candidate which translates to the ghost probability. If the score of the candidate is less than or equal to a threshold then that candidate is killed, and

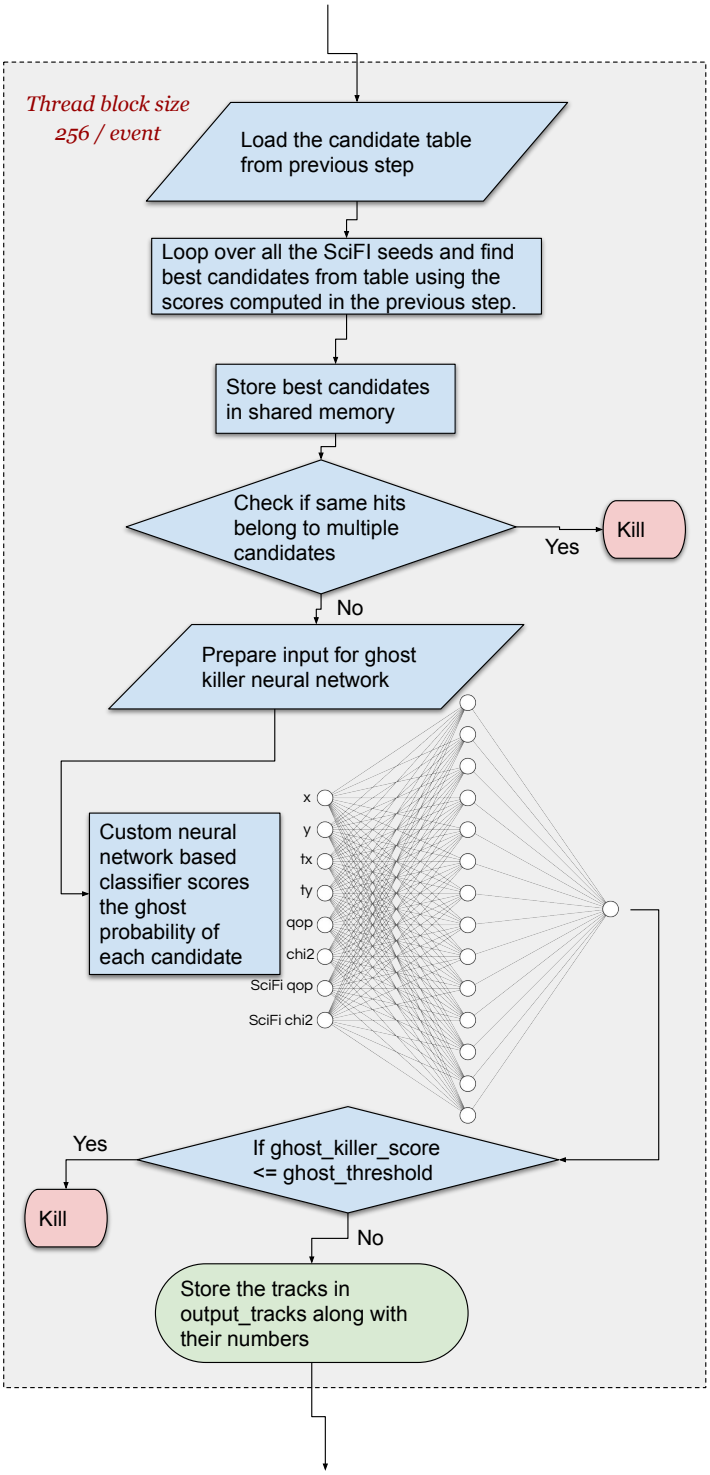


Figure 59: Flow chart of the `Downstream` algorithm (Part 3): Clone and Ghost removal.

the remaining candidates are stored along with their total numbers. The output of this function produces the candidates which have passed all the conditions. These good candidates are then confirmed, and are stored with their track parameters.

5.3.4 Preparation of the output

In this section, the confirmed track candidates are consolidated and prepared for the standard Allen output format which will be used for processing in the subsequent selection algorithms and monitoring. This is implemented in three small kernel functions as shown in Fig. 60, each of which is described below.

downstream_copy_track_hit_number At this stage, we have confirmed tracks with their track parameters from the previous step. In order to consolidate and store all these associated track hits in the next step, we need to determine the memory required. This is achieved by copying the hit numbers of the track candidates to an array. The size of this array is then used to allocate the memory for storing the hits in the next step. This is implemented using a GPU kernel with 512 *threads* per block. Each *thread* copies the hit numbers of a single track candidate to the output array.

downstream_consolidate This kernel function is implemented with 256 *threads* per block and consolidates the data for *downstream* tracks in a multi-particle container. First, a multi-event basic particle container is defined along with the sizes of various data structures used in the algorithm. It essentially allocates memory for the different components involved in the downstream consolidation process such as track states, hits, and others based on the number of events, tracks, and hits.

The kernel function is responsible for filling the consolidation memory with the data from the *downstream* tracks. It extracts data from the input arrays representing the *downstream* tracks and fills the output data structures with relevant information, including states and hits. This function operates in parallel over multiple tracks within each event, and copies and organizes data in the final output format which consists of a multi event basic particle container with the *downstream* tracks and their

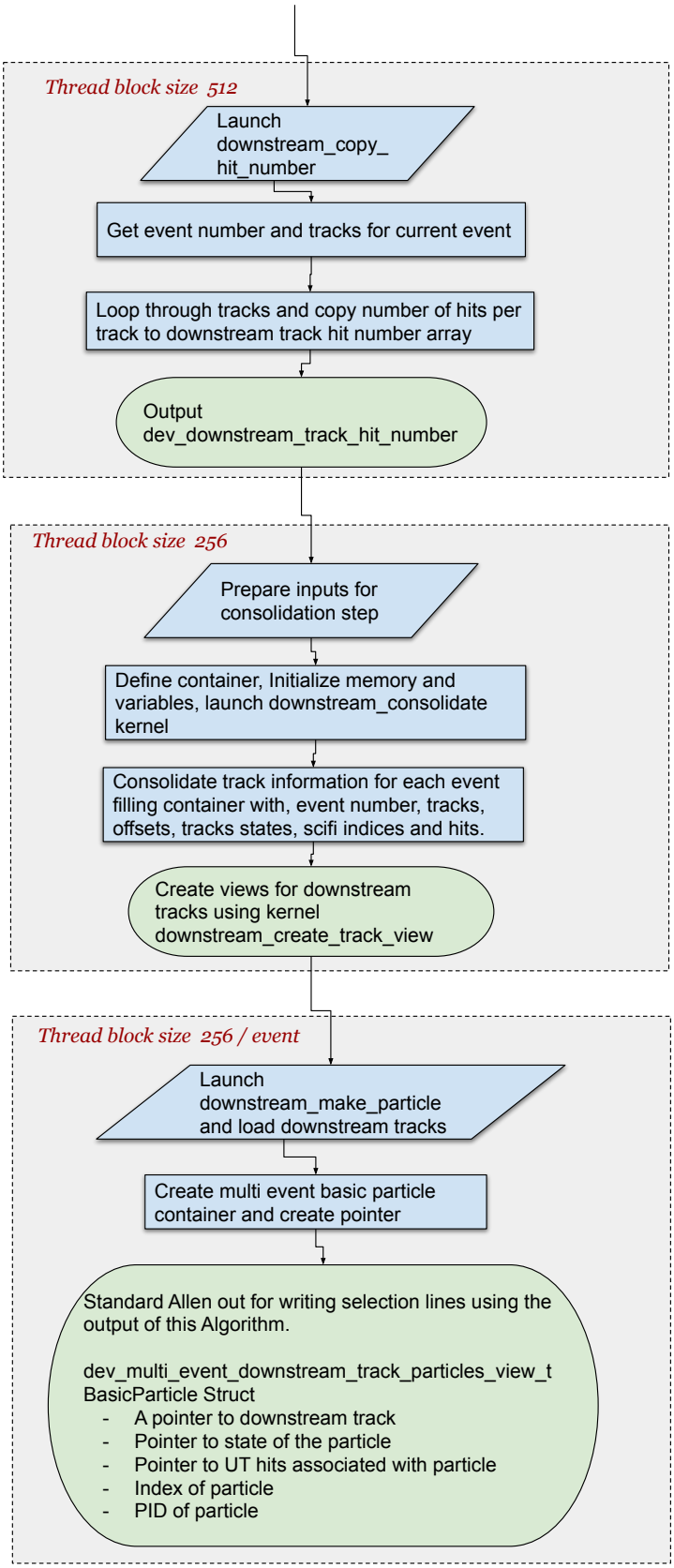


Figure 60: Flow chart of the `Downstream` algorithm (Part 4): Preparation of downstream output.

associated hits and states.

downstream_create_tracks_view This kernel function is also implemented with 256 *threads* per block and creates views[3] of the data for *downstream* tracks. It organizes the *downstream* tracks data for each event, and associates it with track hits and track states which contains kinematics variables of particle path. The function operates in a parallel manner across multiple events and tracks, and structures the data into specialised data structures for optimised access and analysis. This involves:

1. Associating each *downstream* track with its hits.
2. Creating specialised views for UT hits of *downstream* tracks.
3. Creating views for Kalman states, this is done to create the structure of *downstream* reconstructed track same as the standard *long* tracks in the HLT1.
4. Creating views that aggregate data across multiple events for processing.

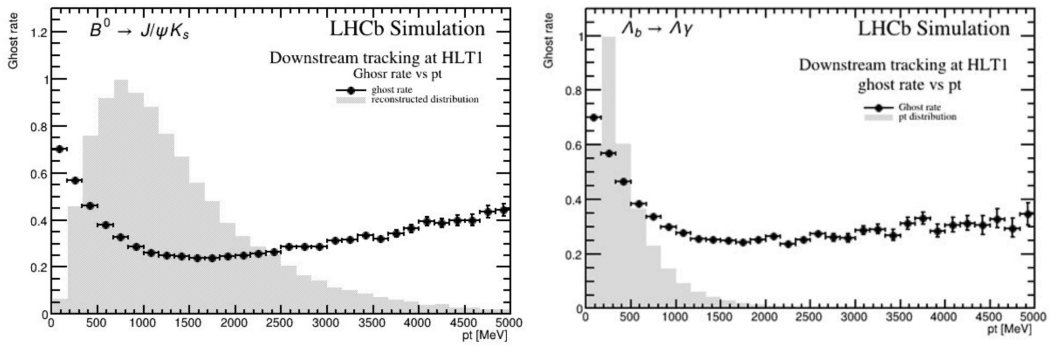
5.4 Neural Network based ghost rejection

Various techniques for reducing ghosts in the algorithm were explored. Some techniques explored such as tightening of the search windows, requiring hits in all the layers of UT and prefiltering of SciFi seeds helped reducing the ghost rates only marginally. In addition, other techniques such as fine-tuning the search windows sizes as function of q/p, applying empirical cuts on *downstream* track χ^2_{match} and re-matching the *downstream* candidates back with SciFi seeds after the reconstruction had negligible effect on the ghost reduction as shown in Fig. 61. At the end, the next avenue for improvements was to try out machine learning approaches.

Introduction

The spurious 'ghost' tracks that do not correspond to real particles can arise from various sources, such as detector noise or reconstruction am-

[3]In C++, "views" refer to lightweight, non-owning references to data. They provide a way to access or modify the data without taking ownership or duplicating the underlying data. This concept is especially useful for providing efficient access to sub-sequences or transformed sequences of data.



(a) (b)

Figure 61: Ghost rate before deploying NN as function of p, p_T for `Downstream` tracks using (a) $B^0 \to J/\psi K^{*0}$ samples and (b) $\Lambda_b^0 \to \Lambda\gamma$ samples.

biguities. A ghost rejection NN is developed and customised for GPU architecture. It consists of an interconnected group of artificial neurons which are organised in layers. The input to the NN is a set of features which are used to predict the output. It is trained using a set of training data samples which consists of input features and the corresponding output. The training data is used to adjust the weights of NN, being trained until the weights are optimised to give the desired output. It is then tested on another set of test data to evaluate its performance. The NN is then used to predict the output for the new set of input features.

NN architecture

The proposed architecture employs a feed-forward NN (FFNN) trained via back propagation, using the binary cross-entropy loss function. The architecture includes an input layer, a hidden layer with Rectified Linear Unit (*ReLU*) activation with 14 nodes, and an output layer with *sigmoid* activation. The input to the network consists of 8 variables related to the track, including hit coordinates and kinematic properties. The output of the network is a scalar between 0 and 1, representing the probability of input track being a ghost. The architecture of the NN, with the corresponding input variables, is shown in Fig. 62.

Mathematically, a NN can be defined as a function f that maps an

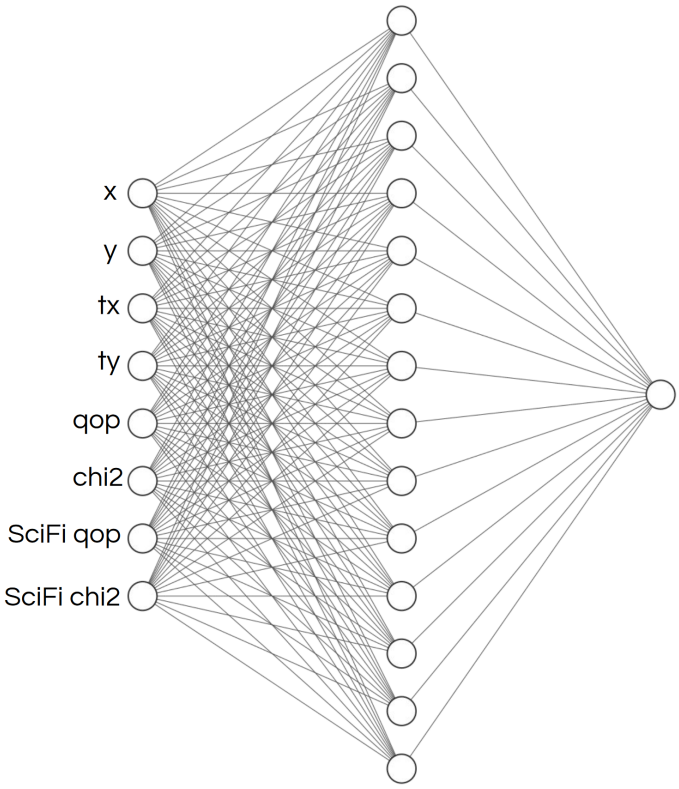


Figure 62: Architecture of the NN used for ghost rejection. Eight variables are used as input: six from the *downstream* track state and its χ^2, and two, the q/p and χ^2, from the corresponding SciFi seeds.

input vector $\mathbf{x}$ to an output scalar h:

$$f(\mathbf{x}) = \sigma(\mathbf{W}_2 \cdot \text{ReLU}(\mathbf{W}_1 \cdot \mathbf{x} + \mathbf{b}_1) + b_2) \tag{5.16}$$

Here, the function f is composed of several components: linear transformations, *ReLU* activation function, and the *sigmoid* activation function.

- $\mathbf{W}_1$ and $\mathbf{b}_1$ are the weight matrix and bias vector of the hidden layer.
- $\mathbf{W}_2$ and b_2 are the weight matrix and bias of the output layer.
- $\text{ReLU}(x) = \max(0, x)$ is the Rectified Linear Unit activation function for the hidden layer.
- $\sigma(x) = \frac{1}{1+e^{-x}}$ is the *sigmoid* function used as the activation function for the output layer.

Some details of these are discussed below. **Linear transformation:** The term $\mathbf{W}_1 \cdot \mathbf{x} + \mathbf{b}_1$ represents a linear transformation of the input vector $\mathbf{x}$. Here, $\mathbf{W}_1$ is a matrix that represents the weights connecting the input layer to the hidden layer, and $\mathbf{b}_1$ is a bias vector. The role of the weights is to control the strength of the influence of the input features on the hidden units, while the bias allows for shifting the activation function.

Rectified Linear Unit (ReLU) activation function:

The ReLU activation function is defined as $\text{ReLU}(x) = \max(0, x)$. In other words, this function returns x if x is greater than or equal to zero, and returns zero otherwise. Mathematically, this is represented as:

$$\text{ReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{5.17}$$

ReLU is commonly used due to its computational efficiency and its contribution in mitigating the vanishing gradient problem, a notable issue in deep NNs [86].

But, importantly, the primary role of the ReLU function is to introduce non-linearity into the network. Real-world data is often non-linear, and to capture the patterns within this data effectively, NNs need to account for this non-linearity. Without a non-linear activation function like ReLU, no matter how many layers the network has, it would behave similarly

to a single-layer network, as the composition of linear functions is still a linear function. ReLU provides this necessary non-linearity, enabling the network to learn more complex functions and better capture the intricacies within the data.

Sigmoid activation function: The sigmoid activation function, denoted by σ, is defined as $\sigma(x) = \frac{1}{1+e^{-x}}$. The sigmoid function takes any real-valued number and squashes it into the range between 0 and 1. This is useful, especially in the output layer of a binary classifier, where we want to interpret the output as a probability. The *sigmoid* function in mathematical form reads:

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{5.18}$$

In the context of the NN, this is used to convert the final linear transformation $(\mathbf{W}_2 \cdot \mathrm{ReLU}(\mathbf{W}_1 \cdot \mathbf{x} + \mathbf{b}_1) + b_2)$ into a probability score.

Pre-processing

The Input variables to the NN as depicted in Fig. 60 and Fig. 62 are positions in x, y, slopes t_x and t_y, q/p, χ^2_{match}, SciFi seeds q/p and χ^2_{match}.

Input features need to be standardised before being fed into the NN. Standardisation ensures that features are on a similar scale, which is beneficial for the convergence of the network's weights. For each feature x_i, the standardisation is performed as follows:

$$x'_i = \frac{x_i - \mu_i}{\sigma_i} \tag{5.19}$$

where μ_i is the mean of feature i and σ_i is the standard deviation of feature i.

Training

Training is performed using the Adam optimisation algorithm and the binary cross-entropy loss function[87]. The binary cross-entropy loss L is defined as:

$$L = -\frac{1}{N}\sum_{i=1}^{N} \left[y_i \log(f(\mathbf{x}_i)) + (1-y_i)\log(1-f(\mathbf{x}_i)) \right] \tag{5.20}$$

where y_i is the true label of the i-th example (1 for ghost, 0 for real track) and $f(\mathbf{x}_i)$ is the output of the NN for the i-th example.

The model was trained using the $B_s^0 \to \phi\phi$ sample, and after 18047 epochs, the model stopped improving on the test sample. The loss values in the final epoch for both the test sample and train samples were consistent with values as.

- *Train loss:* 0.3496.
- *Test loss:* 0.3522.

Postprocessing and evaluation

The output of the network, a scalar between 0 and 1, can be constrained to make binary classification decisions:

$$\text{prediction} = \begin{cases} 1 & \text{if } f(\mathbf{x}) > t \\ 0 & \text{otherwise} \end{cases} \tag{5.21}$$

where t is the threshold. The distribution of the classifier output for the test sample is shown in Fig. 63. The threshold can be tuned to

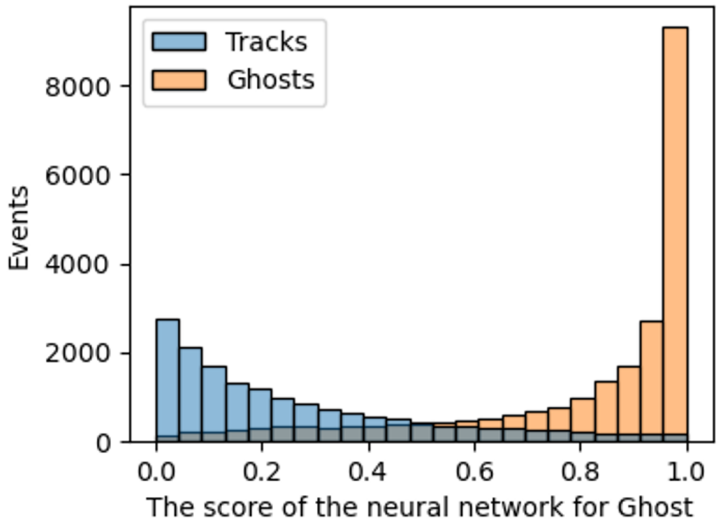


Figure 63: Distribution of the classifier output for the test sample.

achieve a desired trade-off between precision[4] and recall[5]. The effect of

[4]Precision: the ratio of correctly predicted positive observations to the total predicted positives.

[5]Recall: the ratio of correctly predicted positive observations to all the actual positives.

the *ghost killer* threshold on the physics efficiency was studied in detail for different physics channels. Efficiency distributions for different values of *thread* for different channels are shown in Sec. 5.6.1

Implementation challenges

Development and implementation of this NN within the HLT1 software framework inside the Downstream algorithm posed various implementation challenges. As this is the first reconstruction algorithm within the HLT1 framework to use a NN, one of the foremost challenge was to meet the strict throughput requirements of the HLT1. Different design and implementation considerations were taken into account to ensure that the algorithm meets the throughput requirements are discussed below.

- **Feature selection and preprocessing** Selecting the appropriate input variables, which contain discriminatory information regarding the ghost tracks, is crucial for optimizing the performance of the NN. These input variables were selected based on the performance of the classifier when evaluated with a test sample. To facilitate parallel SIMD operations, the selected input variables were organised into vectors, stored in an array. This array comprises the following input variables: x, y, t_x, t_y, q/p and χ^2.

- **Network architecture and hyperparameters**

  A simple FFNN with single hidden layer was constructed. The optimal number of neurons in the hidden layer was determined based on the performance of the classifier on the test sample. Distribution of the ghost rejection rate for different configurations of number of neurons (nodes) in the hidden layer is shown in Fig. 64. As shown, the model with the fewest number of neurons does not exhibit acceptable performance, then as we increase the number of neurons the model shows linear improvement, thereafter, we see the convergence wherein the rate of improvement significantly slows down. The optimal number of neurons in the hidden layer was determined to be 14.

  Additionally, other hyperparameters such as the learning rate, choice

of optimizer and number of epochs were tuned to ensure that the algorithm meets the throughput requirements.

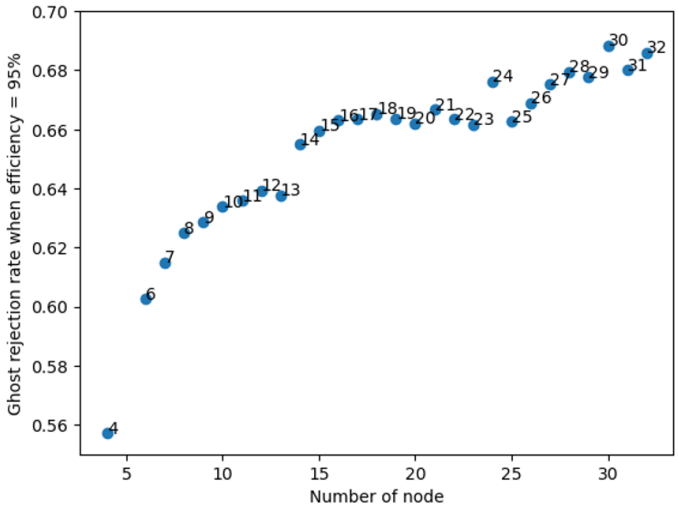


Figure 64: Ghost rejection rate for different configurations of number of neurons in the hidden layer.

- **Overfitting**

  NNs are highly flexible models and can easily overfit the training data, *i.e.*, they can perform well on the training data but poorly on unseen data. To combat overfitting, regularisation techniques and early stopping are used. The validation loss is monitored during training, and if it does not improve for a predefined number of epochs, the training is halted. This technique is known as early stopping. The number of epochs after which the training is stopped is determined based on the performance of the classifier on the test sample. This procedure employs the variables `early_stop_epochs`, `best_val_loss`, and `epochs_no_improve`. The algorithm can be summarised as follows:

  1. Initialize `best_val_loss` to ∞, and `epochs_no_improve` to 0.
  2. For each epoch, compute the validation loss.
  3. If the validation loss is lower than `best_val_loss`, update `best_val_loss` and reset `epochs_no_improve` to 0.
  4. If the validation loss is not lower, increment `epochs_no_improve`.
  5. If `epochs_no_improve` equals `early_stop_epochs` (which is fixed value of 1000), stop the training.

The number of epochs at which the improvement ceased was found to be 18,047.

- **Computation challenges** For each node in the hidden layer, a linear combination of all inputs must be computed, resulting in a high number of operations per track candidate.

$$\#\text{ of operations} = \#\text{ of inputs} \times \#\text{ of nodes in hidden layer} \quad (5.22)$$

This can be computationally expensive, especially when the number of nodes in the hidden layer is large. To overcome this, several computing techniques were employed which allowed efficient usage of the underlying GPU hardware. These techniques are described in detail in Sec. 5.5

Conclusion

The approach employs a simple FFNN, and by optimizing its weights, it learns to distinguish between real and ghost tracks. The NN is applying two linear transformations, the first is followed by a ReLU activation and the second by a sigmoid activation. This sequence of linear transformation, non-linearity, linear transformation, non-linearity, allows the network to learn complex relationships between the inputs and ghost classification. The final output is a value between 0 and 1, representing the probability that the input track is a ghost. This output can then be thresholded at a value (e.g default value used is 0.2) to make a binary classification.

The performance of the NN has been evaluated and crosschecked for different physics channels with different run conditions and is found to be stable for all the channels. Details of the performance are covered in the Sec. 5.6.

5.5 GPU implementation and optimisations

Achieving high-throughput in Downstream reconstruction at HLT1 necessitated leveraging advanced computational capabilities. To this

end, we utilised an array of cutting-edge features and optimisation techniques from the GPU and C++ programming paradigms.

5.5.1 Harnessing GPU capabilities

We employed various facets of GPU programming features for exploiting the computational power of the underlying hardware.

- **Kernels and parallelisation**: refining kernel functions to augment parallel processing performance.
- **Memory management**: fine-tuning memory hierarchy utilisation including global, shared, and texture memory.
- **Device-host communication**: optimizing data transfer between the GPU device and the host system using Memory coalescing techniques.

Kernel function design and parallelisation

The kernel functions form the crux of GPU computation. We dissected the entire algorithm into multiple kernel functions. Each kernel function was designed to perform a specific sub-task of the algorithm. This modular approach allowed for more granular optimisation and debugging. Block-level parallelisation within each kernel is employed, with *threads* collaborating to perform computations. To further enhance performance, each kernel function was carefully tuned with optimal block sizes, tailored for the underlying hardware architecture.

Memory optimisation

Memory bandwidth is often a bottleneck in GPU computations. To address this, memory optimisation strategies have been employed. Static compile-time memory allocation was favored over dynamic runtime allocation, as it reduces the overhead associated with memory management and increases the predictability and stability of

memory access patterns. Additionally, we made extensive use of shared memory spaces, which is faster than global memory.

Device-host communication

When executing parallel operations, GPUs read data from memory in a specific pattern. Ideally, if *threads* access contiguous memory locations, the memory access can be 'coalesced' into a single transaction, significantly improving the efficiency and speed of data retrieval. However, if the memory access pattern is non-contiguous or 'strided', it can lead to non-coalesced accesses, which can greatly increase memory latency and hamper the overall performance. Therefore, arranging data in a SOA format allowed coalesced memory access. This was achieved by storing each attribute type in a separate array, which facilitated rapid data processing in SIMD parallel operations.

GPU profiler-led optimisations

To systematically evaluate and optimize performance, we made extensive use of GPU profiling tools. These tools provided insights into the resource utilisation and execution behavior of our kernels on the GPU. Using this information, we were able to make informed decisions regarding memory access patterns, kernel launch configurations, and computation optimisations. This profiler-led optimisation process helped us in identifying bottlenecks and achieving performance improvements.

5.5.2 Leveraging C++ features

The versatility and efficiency of C++ as a programming language were crucial to achieving high performance. We integrated modern C++ features such as:

1. **Static structure**: The number of inputs to the NN and the number of nodes in the hidden layer is fixed and known at

compile time. This allows the compiler to optimize the code at instruction set level.

2. **Loop unwinding**: Implementing the number of operations in C++/CUDA required chaining two *for-loops*, which could introduce potential bottlenecks. To overcome this, the loop unwinding technique has been used to unroll the loops and vectorize the operations which is possible since we are using static structures. This technique is used to increase the performance of the algorithm by decreasing the overhead of loop control. This was achieved by using template C++ functions *unwind* to explicitly unroll the loops. Instead of having a traditional loop structure, each iteration of the loop is written out explicitly with the loop control code removed or minimised. This can provide performance improvements especially in cases where the number of iterations is small and known at compile time.
3. **Fast math functions**: The `fast_math` functions are a set of functions that can be used to speed up the execution of floating-point math operations. These functions are not IEEE 754 compliant and may not provide the same results as the standard math functions. However, they are faster than the standard math functions. The `fast_math` functions `__fdividef` and `__expf` are used in the implementation of the NN to speed up the execution of the algorithm.

The integration of these features and optimisation techniques was important to meet the stringent throughput requirements of `downstream` reconstruction at HLT1.

5.6 Figures of merit

In this section the performance of the `Downstream` algorithm is presented. It is evaluated for various figures of merits using multiple physics channels and compared to the performance of the HLT2 `PrLongLived` algorithm. A wide range of physics channels which are expected to be populated in *downstream* tracks have been studied.

5.6.1 Physics performance

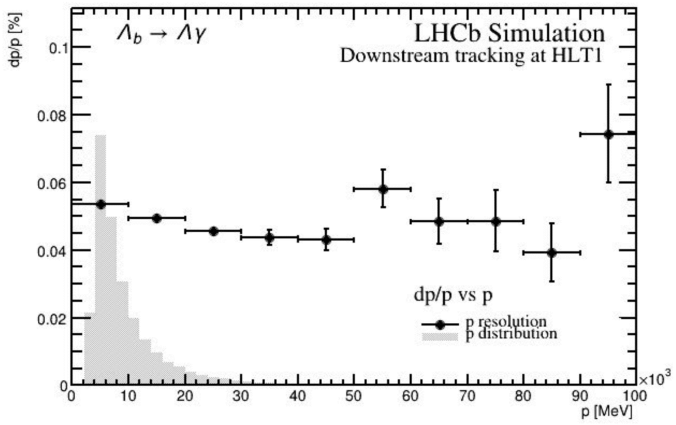
The samples of interest which were used for these performance studies cover wide range of physics decays. The samples are listed in Table 9.

Table 9: Summary of data samples used for Physics performance studies.

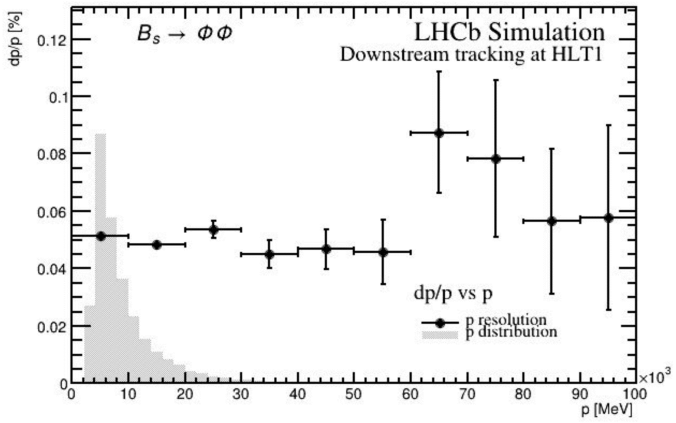
Sample Decay	Location of sample	Number of events
MinBias	/eos/lhcb/wg/rta/WP6/Allen /digi/input/RetinaCluster /samples/v1/upgrade-minbias-magdown-scifi-v5 /retinacluster/digi/upgrade-minbias-magdown-scifi-v5-retinacluster.digi	50k
$B_s \to \Phi\Phi$	/MC/Upgrade/Beam7000GeV-Upgrade-MagDown-Nu7.6-25ns-Pythia8/Sim10-Up02-OldP8Tuning/13104012/XDIGI	50k
$B^+ \to DK$	/MC/Upgrade/Beam7000GeV-Upgrade-MagDown-Nu7.6-25ns-Pythia8/Sim10-Up03-OldP8Tuning/12165102/XDIGI	50k
$\Lambda_b^0 \to \Lambda\gamma$	/MC/Upgrade/Beam7000GeV-Upgrade-MagDown-Nu7.6-25ns-Pythia8/Sim10aU1/15102320/XDIGI	50k
$K_S \to \mu^+\mu^-$	/MC/Upgrade/Beam7000GeV-Upgrade-MagDown-Nu7.6-25ns-Pythia8/Sim10aU1/34112100/XDIGI	50k
$J/\psi\Lambda\Lambda$	/MC/Upgrade/Beam7000GeV-Upgrade-MagDown-Nu7.6-25ns-Pythia8/Sim10aU1/24104101/XDIGI	50k

Momentum resolution

Figure 65 shows the momentum resolution of the Downstream algorithm to be under 6%, independent of any physics channel demonstrated using $\Lambda_b^0 \to \Lambda\gamma$ and $B_s^0 \to \phi\phi$ samples.



(a)



(b)

Figure 65: Momentum resolution of the `Downstream` algorithm. On the x-axis is the *true* momentum in MeV/c, and on the y-axis is the relative difference between the measured and *true* momentum (momentum resolution) $\frac{dp}{p}$% using (a) $\Lambda_b^0 \to \Lambda\gamma$ on left and (b) $B_s^0 \to \phi\phi$ on the right.

Comparison with the HLT2

The physics performance of the HLT2 (PrLongLived) algorithm is a good benchmark for comparing the performance of HLT1 Downstream algorithm. At the same time, it should be noted that for Run3, the data rates at the HLT1 level as shown in Fig. 25 are about 20 times higher than HLT2. Therefore, a key difference while developing the HLT1 algorithm was the throughput.

In terms of the physics performance, Table 10 shows the comparison of efficiencies and ghost rates for different physics channels of interest.

Table 10: HLT1 vs HLT2 performance comparison: Efficiencies and Ghost rates

MC Decay channel	Efficiency						Ghosts	
	$noVelo + UT + SciFi\_fromLambda\_P > 5GeV\_PT > 500MeV$		$noVelo + UT + SciFi\_fromKs0\_P > 5GeV\_PT > 500MeV$		$noVelo + UT + SciFi\_fromSignal\_P > 5GeV\_PT > 500MeV$			
	HLT1	HLT2	HLT1	HLT2	HLT1	HLT2	HLT1	HLT2
MinBias	73.25%	58.86%	71.45%	57.59%	-	-	18.27%	49.06%
$B_s \to \Phi\Phi$	74.98%	79.69%	72.71%	77.57%	48.85%	44.63%	16.99%	38.74%
$B^+ \to DK$	74.39%	78.31%	73.12%	77.67%	68.28%	70.09%	17.76%	40.31%
$\Lambda_b \to \gamma$	73.28%	55.30%	71.44%	54.54%	71.98%	52.71%	22.92%	43.48%
$K_S \to \mu^+\mu^-$	75.19%	57.24%	72.71%	58.01%	74.91%	59.67%	20.66%	40.82%
$J/\psi\Lambda\Lambda$	73.43%	55.17%	71.35%	55.72%	71.14%	52.77%	22.61%	43.89%

Reconstruction efficiency

In this section, the distribution of the efficiency vs p, p_T, η, and nPV for various physics channels listed in Table 9. Reconstructed *downstream* tracks passing through UT and SciFi detectors (*isDown* category) but not VELO (*noVELO*), and have $2 < \eta < 5$, $p > 5\,\mathrm{GeV/c}$, and $p_T > 0.5\,\mathrm{GeV/c}$ is shown from Fig. 67d to Fig. 74d.

Ghost Rate

Distribution of Ghost rate vs p, p_T, η, and nPV for various physics channels listed in Table 9 are shown from Fig. 75d to Fig. 80d.

Ghost killer threshold

To determine the optimal *ghost killer* threshold value from the NN output (as discussed in Sec. 5.4), the efficiency and ghost rate as a function of p, p_T, η, and nPV for various values of the *ghost killer* threshold are studied. The analysis considers different physics channels, as listed in Table 9. The tracks in consideration pass through the UT and SciFi detectors (*isDown*), but not through the VELO detector (*noVELO*). Moreover, they satisfy the criteria $2 < \eta < 5$, $p > 5\,\mathrm{GeV/c}$, and $p_T > 0.5\,\mathrm{GeV/c}$. Distribution of efficiencies and ghost rates for various values of *ghost killer* thresholds for different simulation samples are shown from Fig. 89 to Fig. 94.

5.6.2 Throughput

Throughput comparison of the base HLT1 sequence with Downstream sequences on a Nvidea A5000 GPU is shown in Fig. 66. This includes `Downstream_standalone`, `hlt1_pp_matching` and `hlt1_pp_forward_then_matching_downstream`.

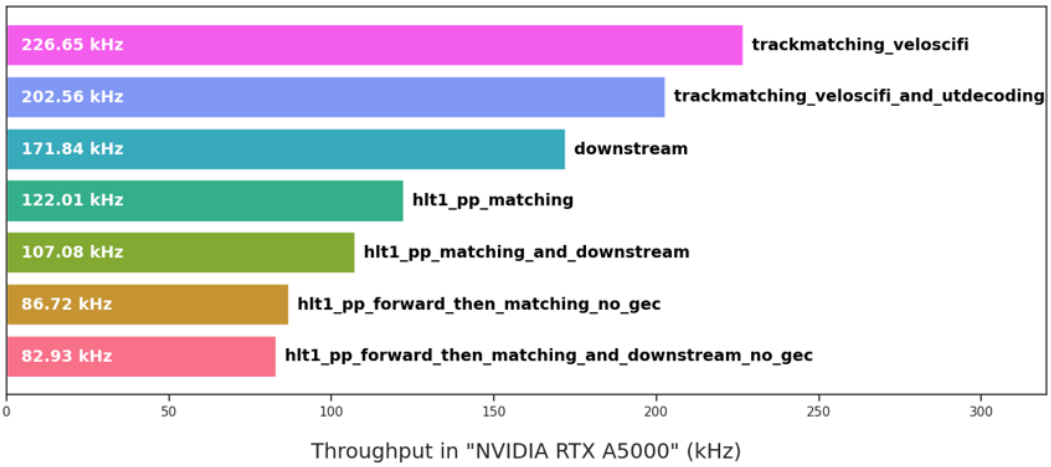


Figure 66: Throughput comparison of the base HLT1 sequence with Downstream sequences. This includes `Downstream_standalone`,`hlt1_pp_matching_and_downstream`, and `hlt1_pp_forward_then_matching_and_downstream_noGEC` on an NVIDIA A5000 GPU.

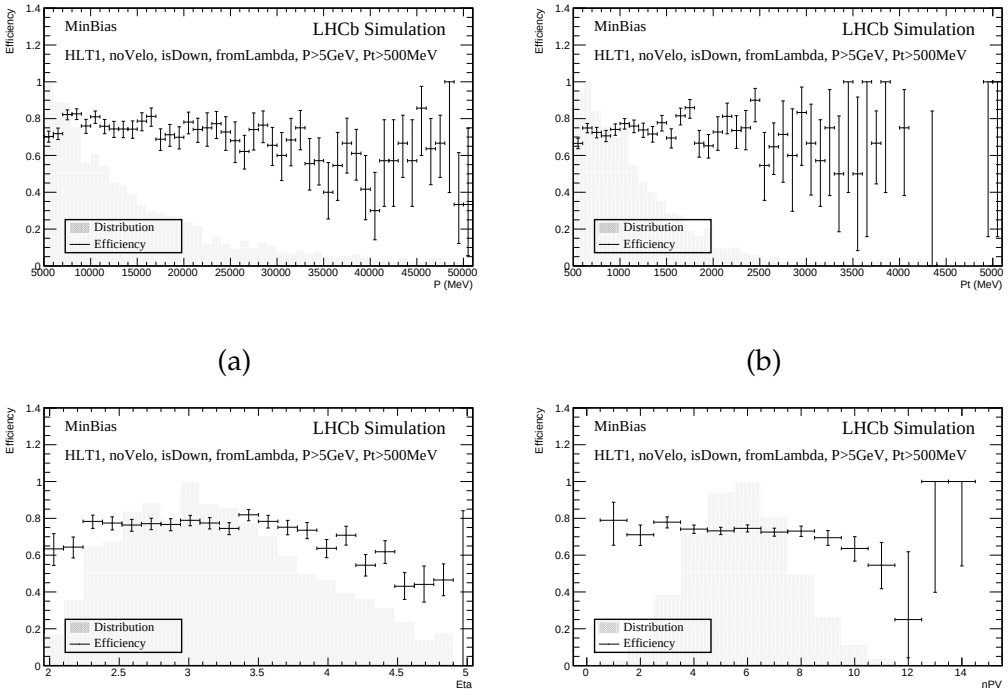


Figure 67: Efficiency distribution of the `Downstream` track reconstruction for non-electron tracks in *MinBias* samples (from Λ category) versus (a) p, (b) p_T, (c) η, and (d) nPV.

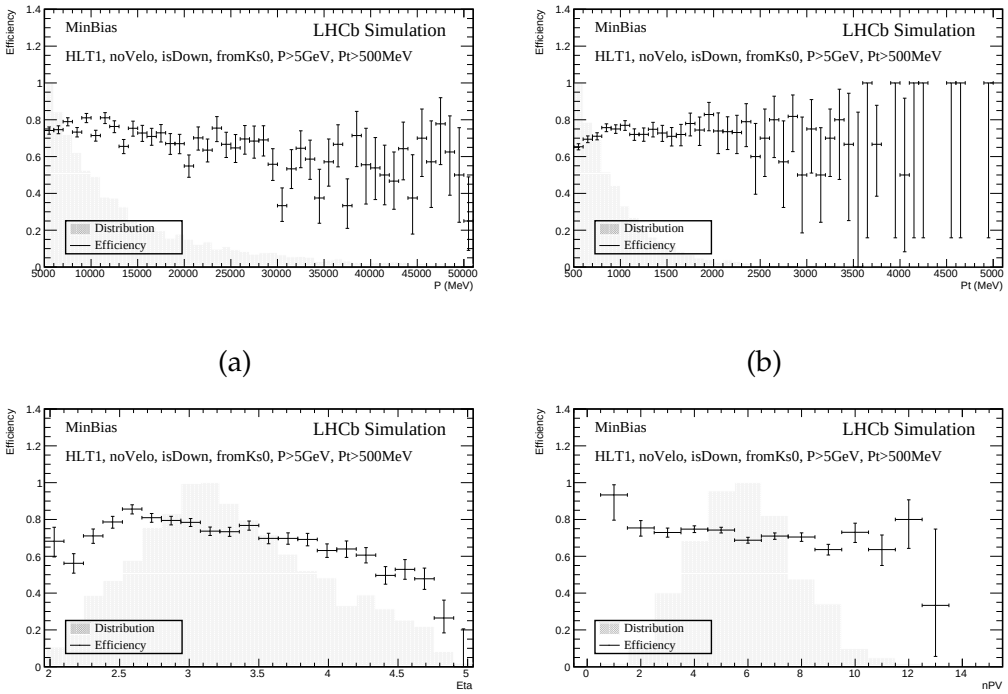


Figure 68: Efficiency distribution of Downstream track reconstruction for non-electron tracks in *MinBias* samples (from K_s^0 category) versus (a) p, (b) p_T, (c) η, and (d) nPV.

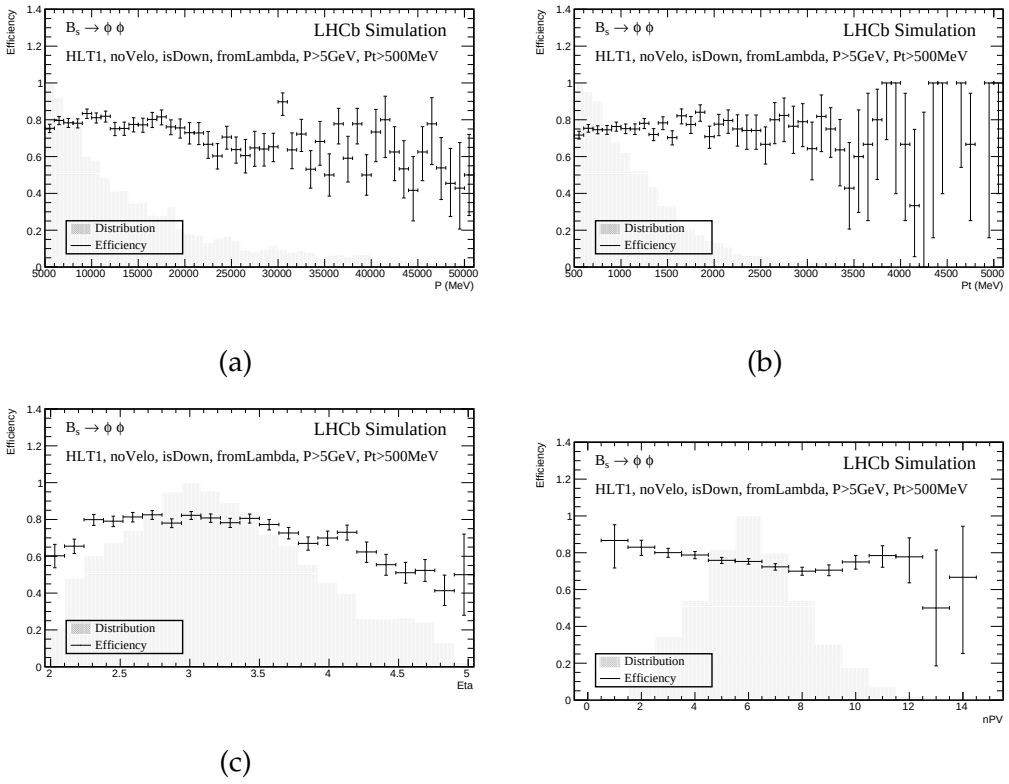


Figure 69: Efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $B_s^0 \to \phi\phi$ samples (from Λ category) versus (a) p, (b) p_T, (c) η, and (d) nPV.

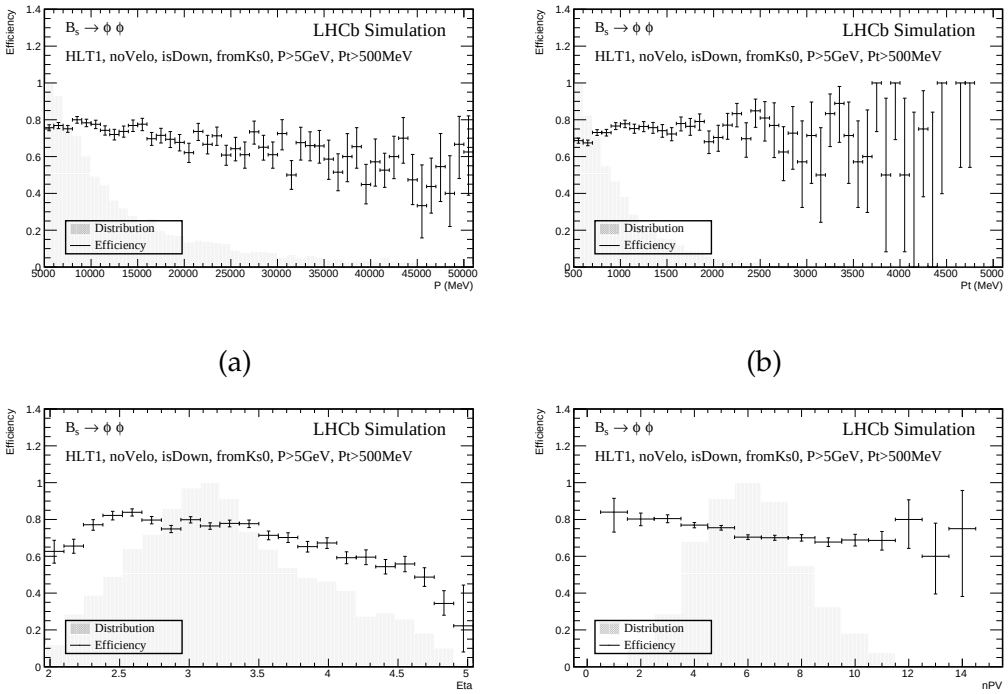
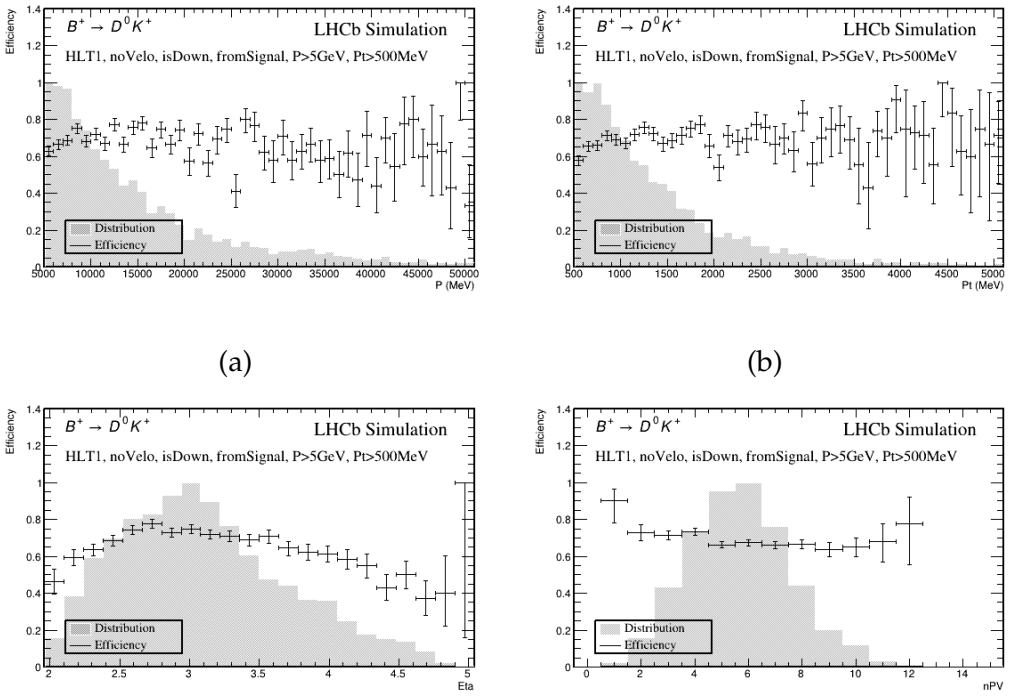


Figure 70: Efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $B_s^0 \to \phi\phi$ samples (signal K_S^0 category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



(a) (b)

(c) (d)

Figure 71: Efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $B^+ \to (D^0 \to K^0_S \pi^+ \pi^-) K^+$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.

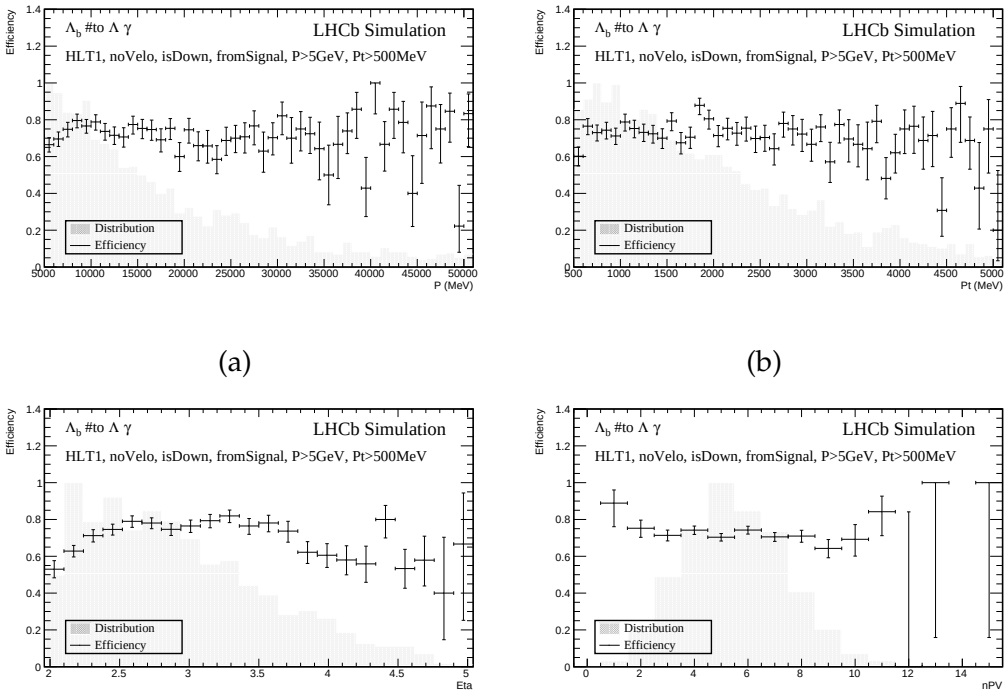
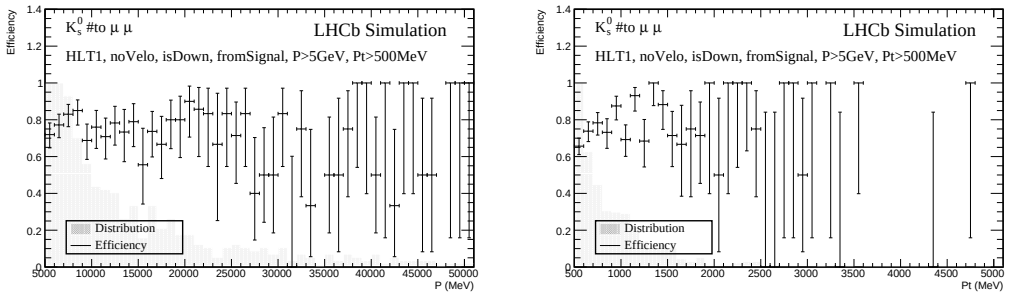
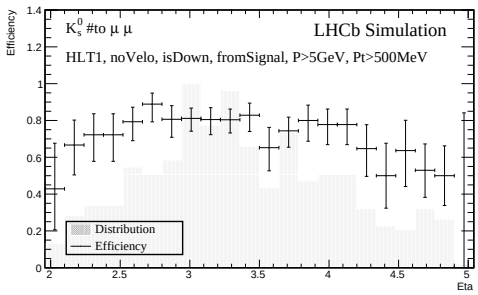


Figure 72: Efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $\Lambda_b^0 \to \Lambda\gamma$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.

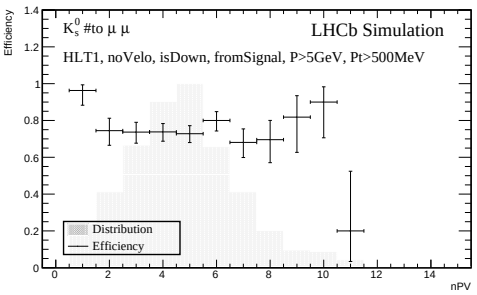


(a)

(b)

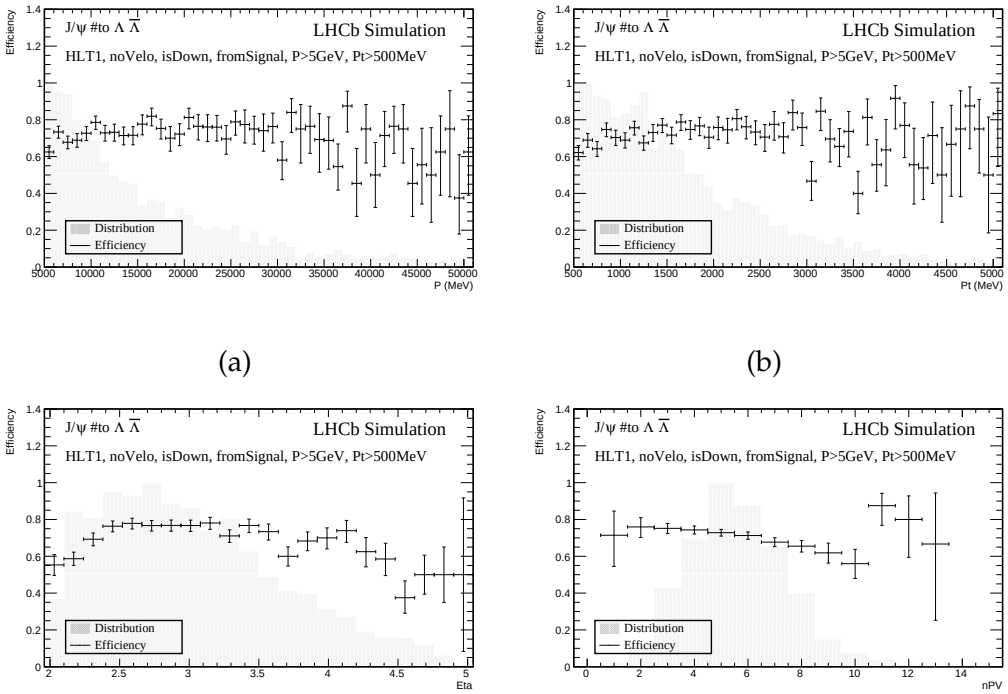


(c)



(d)

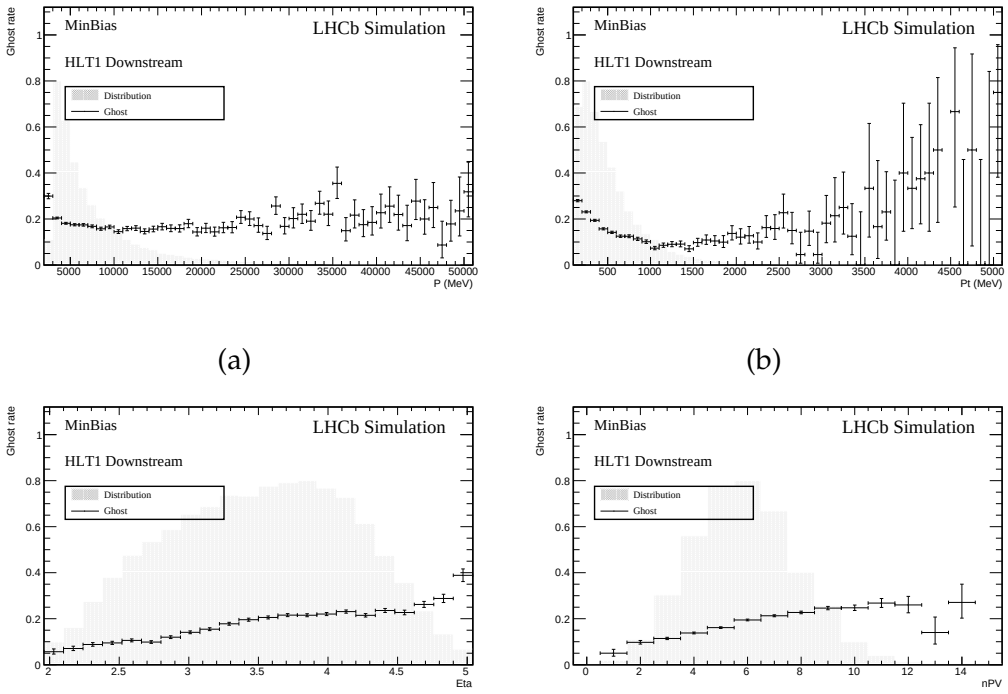
Figure 73: Efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $K_S^0 \to \mu^+\mu^-$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



(a) (b)

(c) (d)

Figure 74: Efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $J/\psi \to \Lambda\bar{\Lambda}$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



(a) (b)

(c) (d)

Figure 75: Ghost rate distribution of *Downstream* track reconstruction algorithm for non-electron tracks in *MinBias* samples versus (a) p, (b) p_T, (c) η, and (d) nPV.

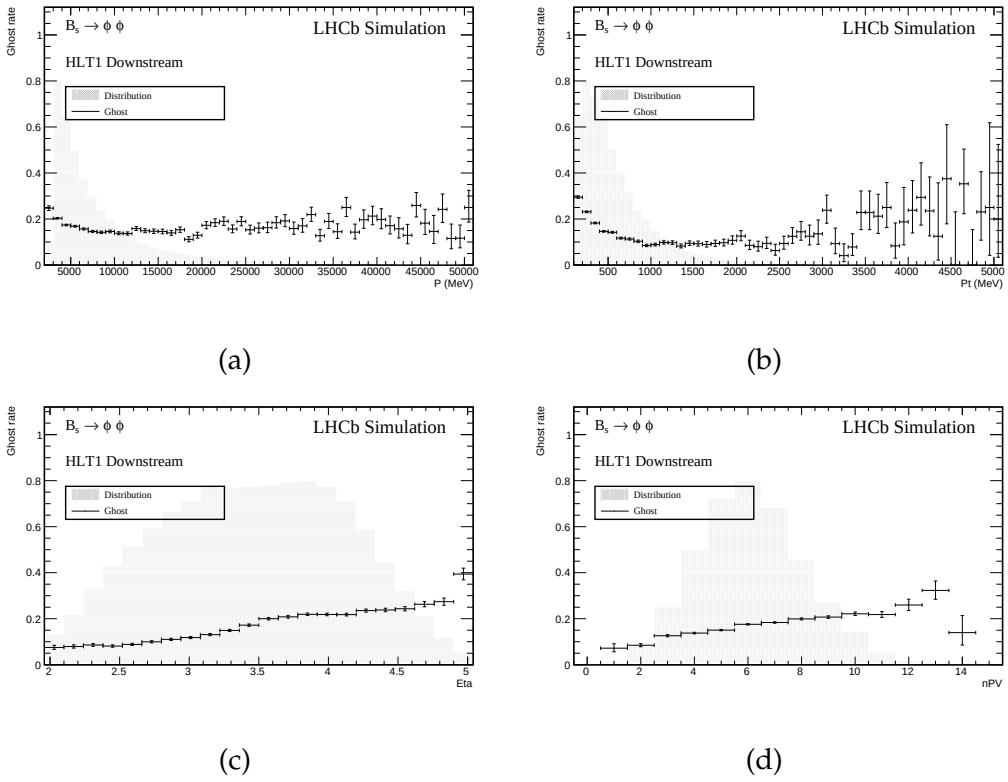
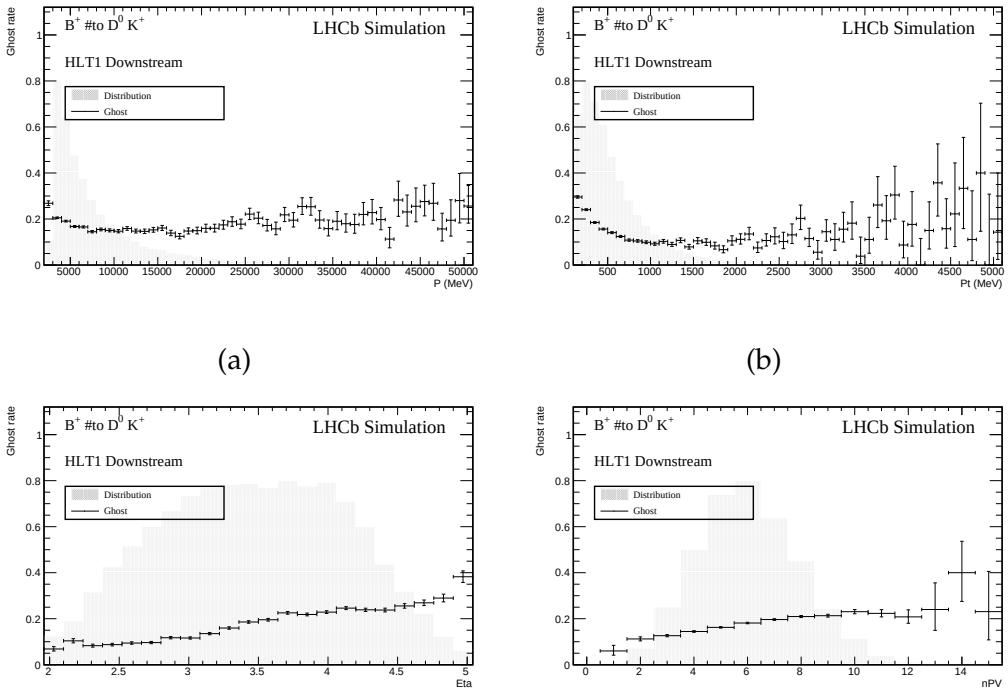


Figure 76: Ghost rate distribution of *Downstream* track reconstruction algorithm for non-electron tracks in $B_s^0 \to \phi\phi$ sample versus (a) p, (b) p_T, (c) η, and (d) nPV.



(a) (b)

(c) (d)

Figure 77: Ghost rate distribution of *Downstream* track reconstruction algorithm for non-electron tracks in $B^+ \to (D^0 \to K^0_S \pi^+ \pi^-) K^+$ sample versus (a) p, (b) p_T, (c) η, and (d) nPV.

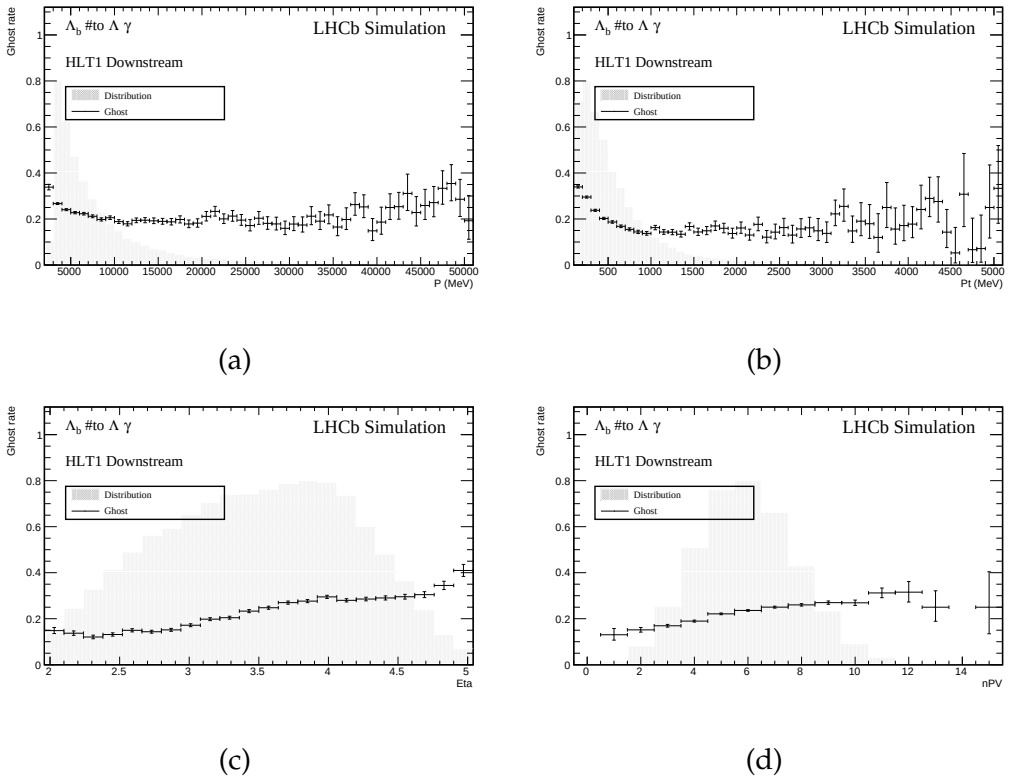
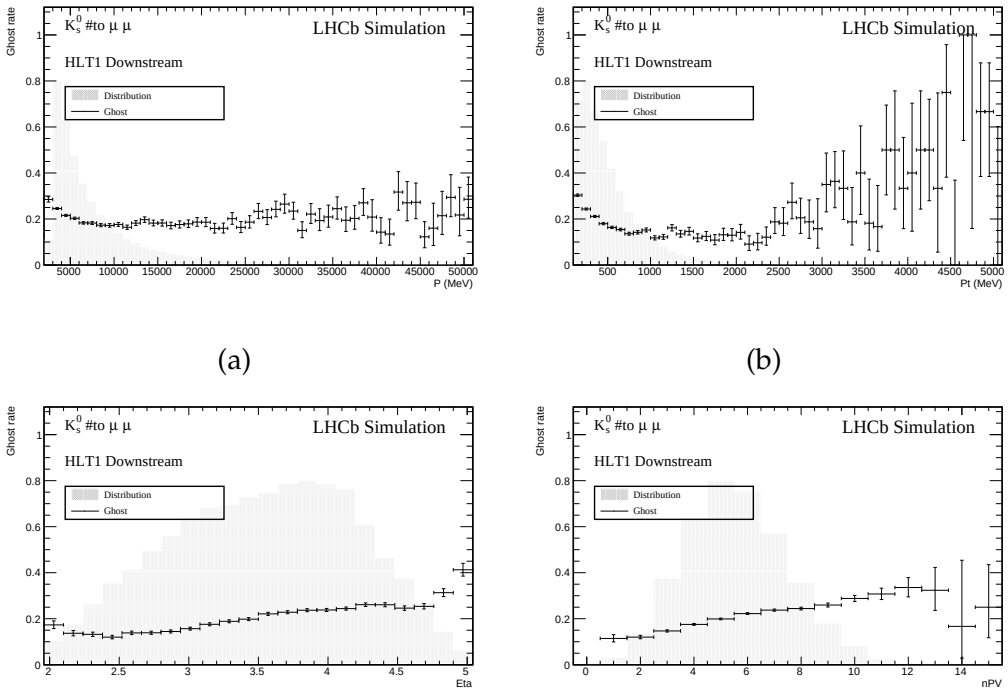


Figure 78: Ghost rate distribution of *Downstream* track reconstruction algorithm for non-electron tracks in $\Lambda_b^0 \to \Lambda\gamma$ sample versus (a) p, (b) p_T, (c) η, and (d) nPV.



(a) (b)

(c) (d)

Figure 79: Ghost rate distribution of *Downstream* track reconstruction algorithm for non-electron tracks in $K^0_S \to \mu^+\mu^-$ sample versus (a) p, (b) p_T, (c) η, and (d) nPV.

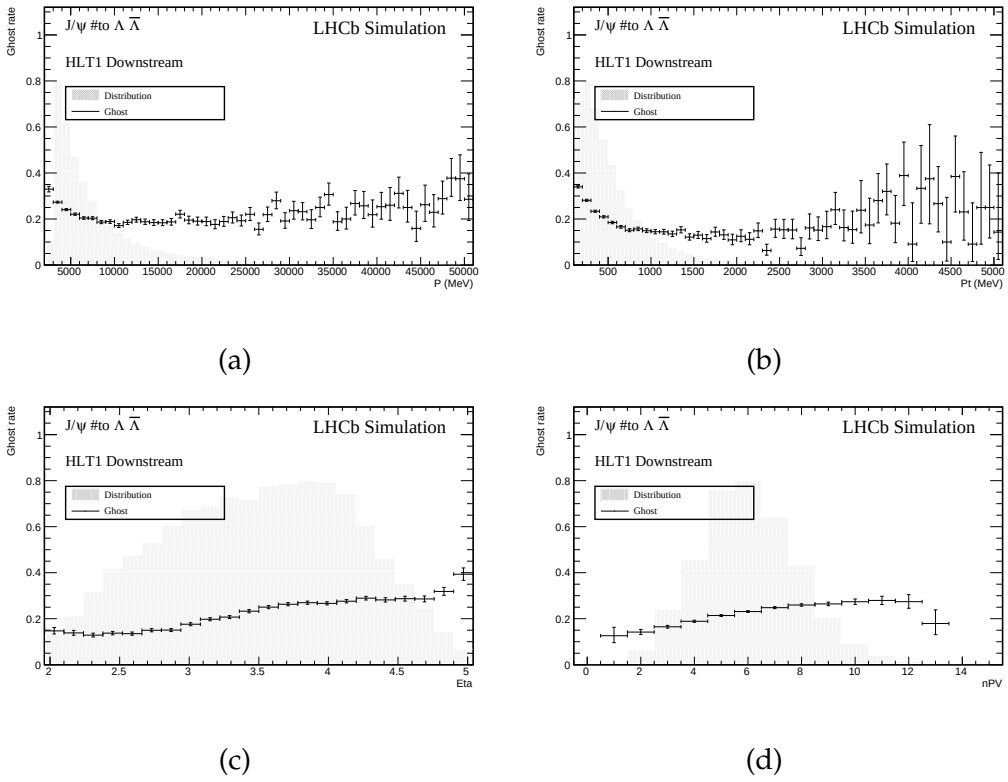
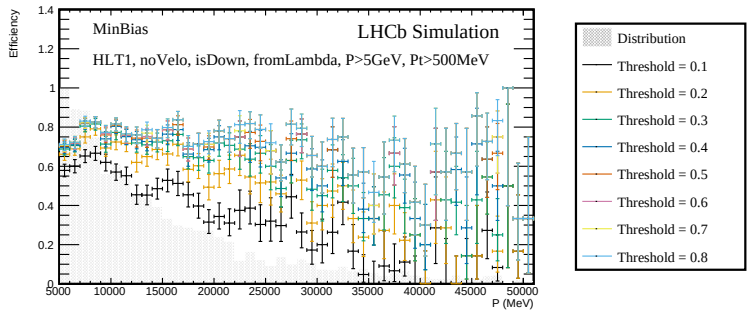
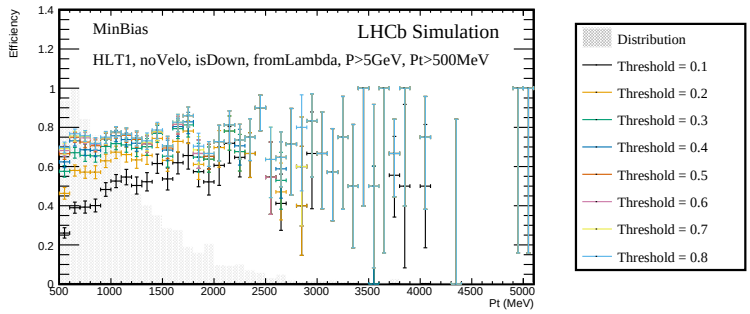


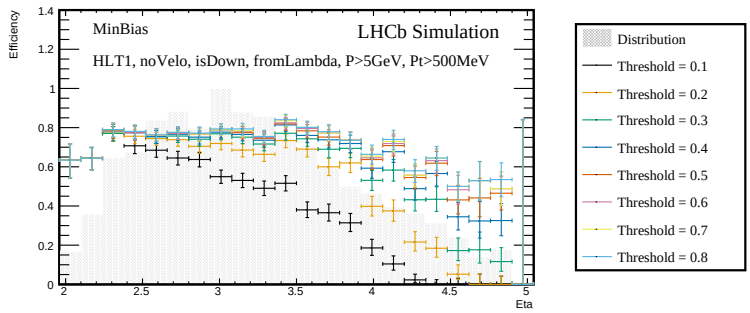
Figure 80: Ghost rate distribution of *Downstream* track reconstruction algorithm for non-electron tracks in $J/\psi \to \Lambda\bar{\Lambda}$ sample versus (a) p, (b) p_T, (c) η, and (d) nPV.



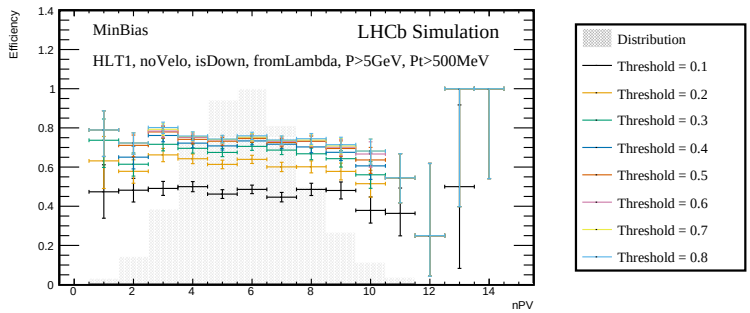
(a)



(b)

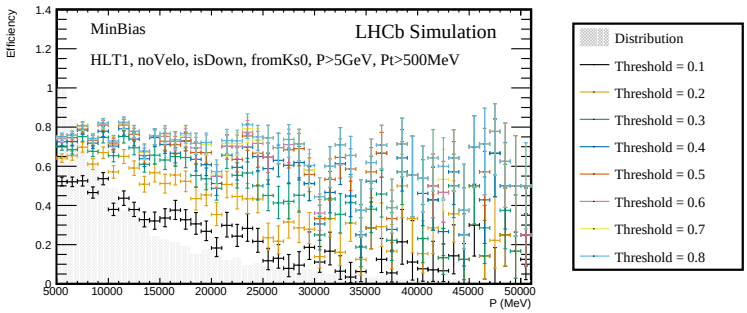


(c)

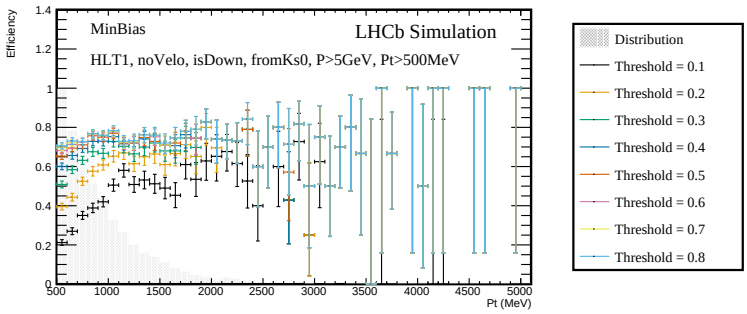


(d)

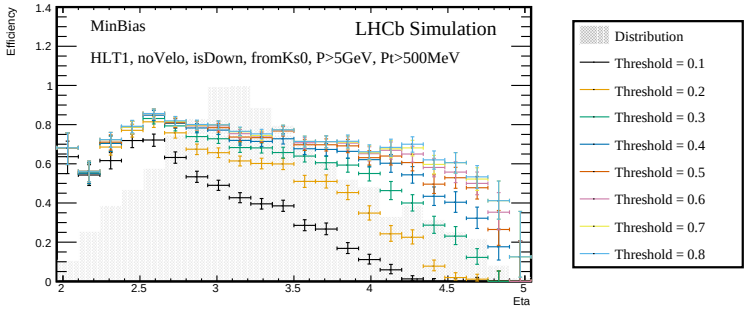
Figure 81: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in *MinBias* samples (from Λ category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



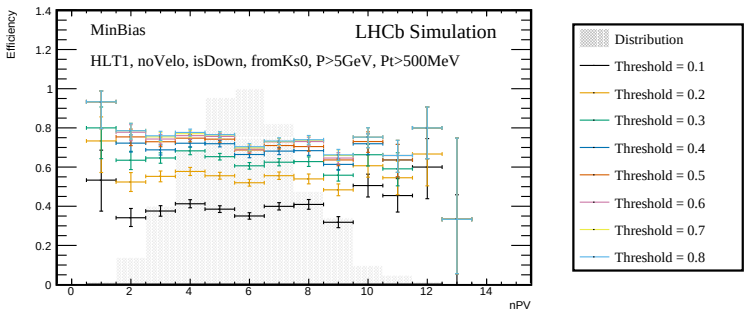
(a)



(b)

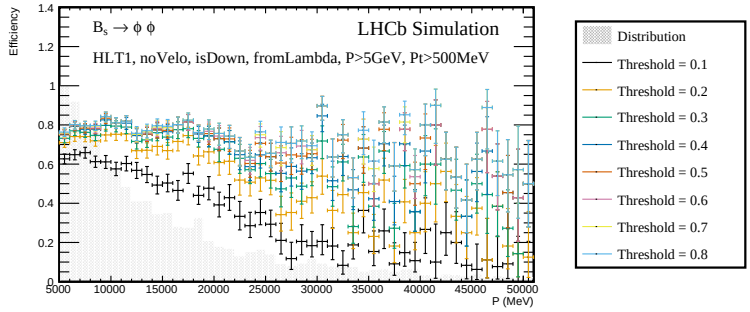


(c)

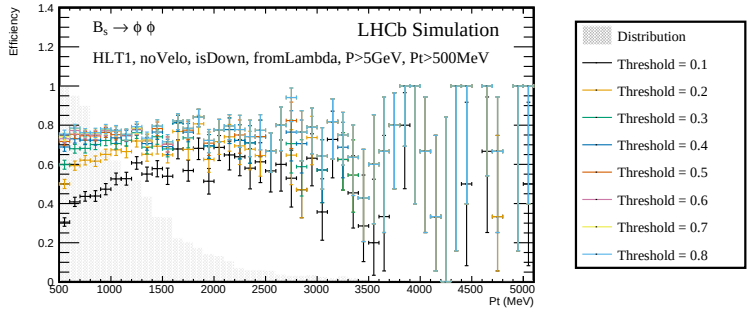


(d)

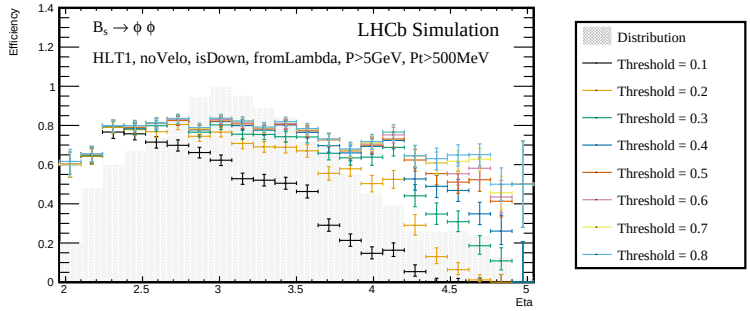
Figure 82: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in *MinBias* samples (from K_s^0 category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



(a)



(b)



(c)

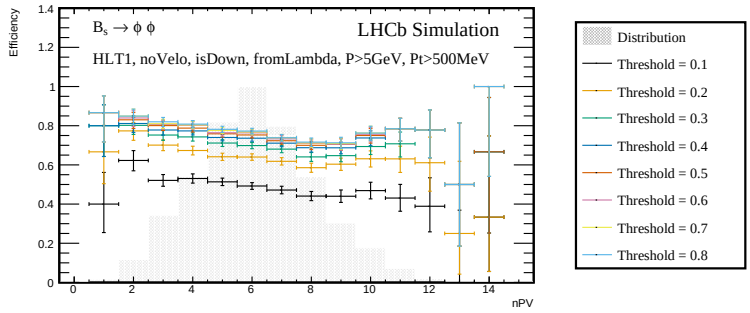
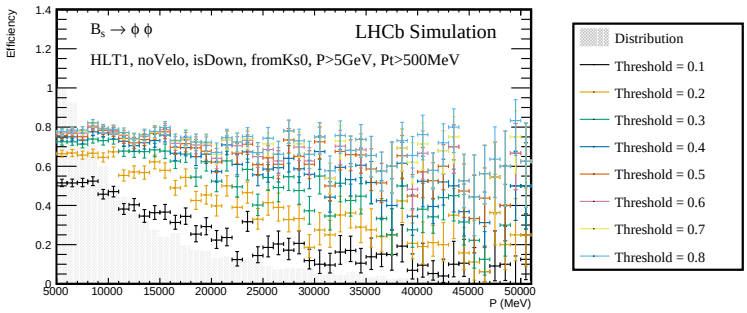
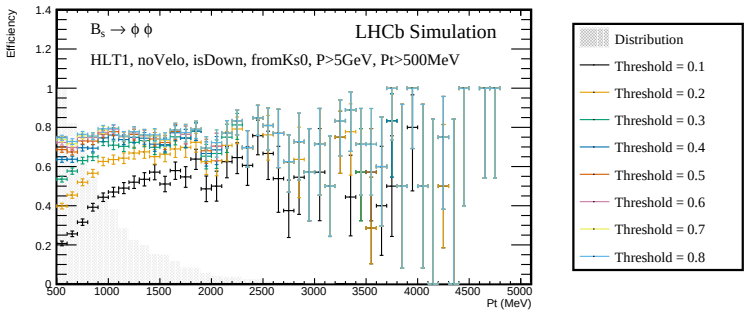


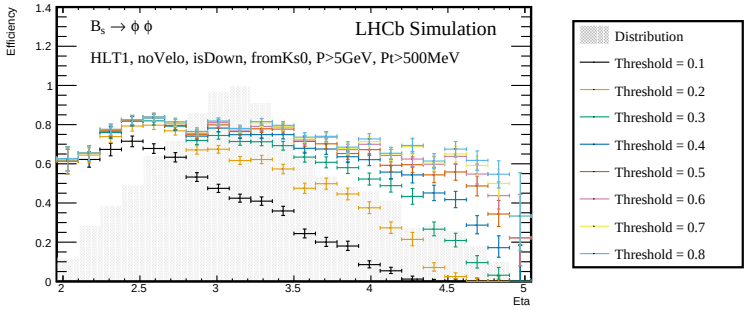
Figure 83: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $B_s^0 \to \phi\phi$ samples (from Λ category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



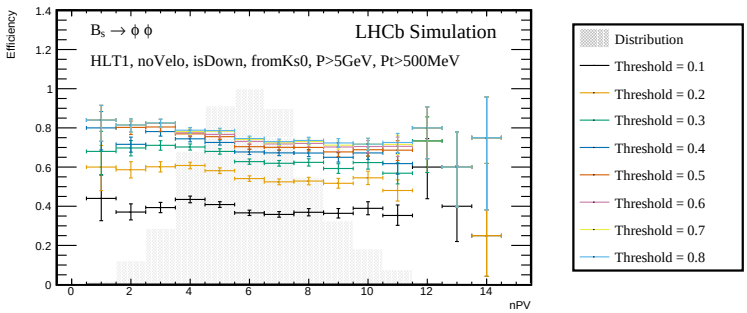
(a)



(b)

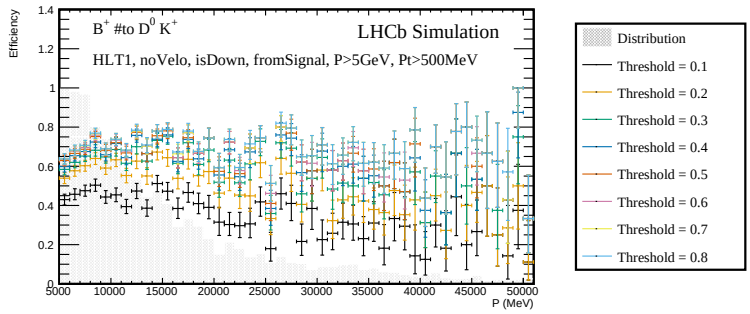


(c)

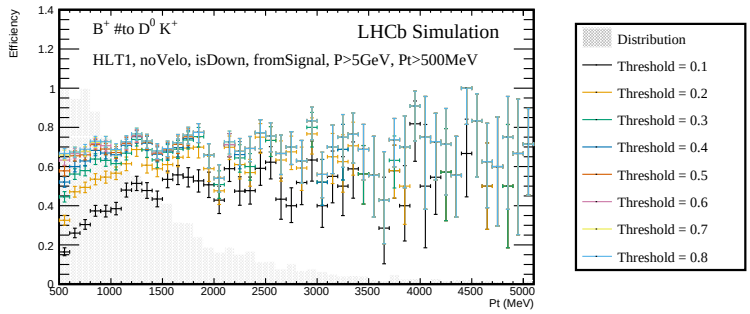


(d)

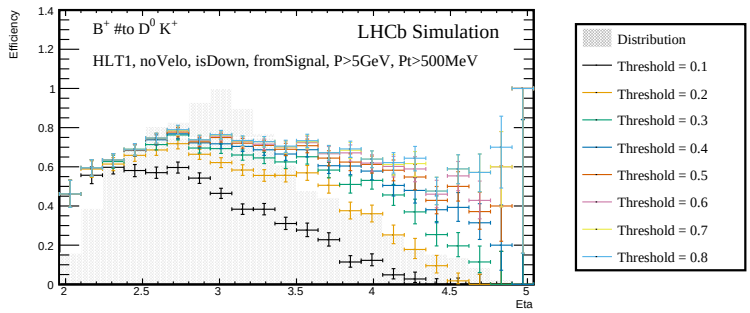
Figure 84: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $B_s^0 \to \phi\phi$ samples (signal K_S^0 category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



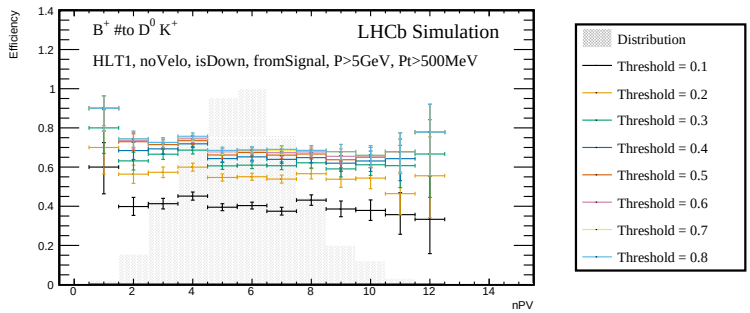
(a)



(b)

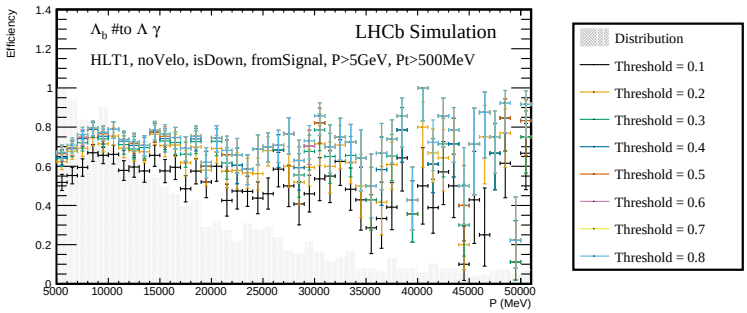


(c)

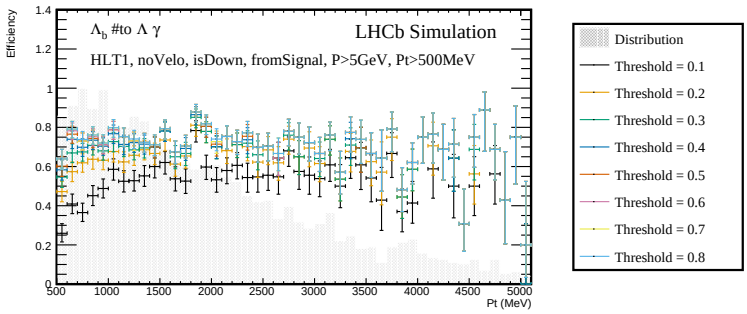


(d)

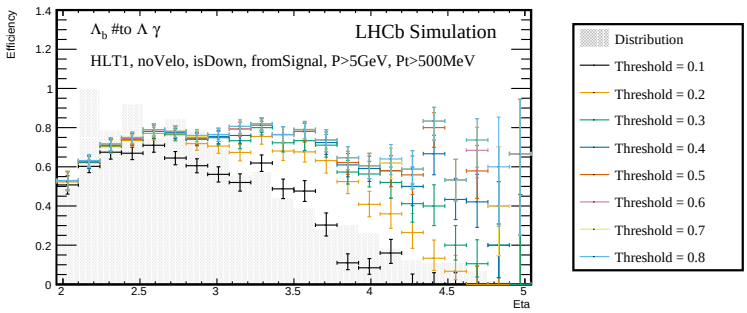
Figure 85: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $B^+ \to (D^0 \to K^0_S \pi^+ \pi^-) K^+$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



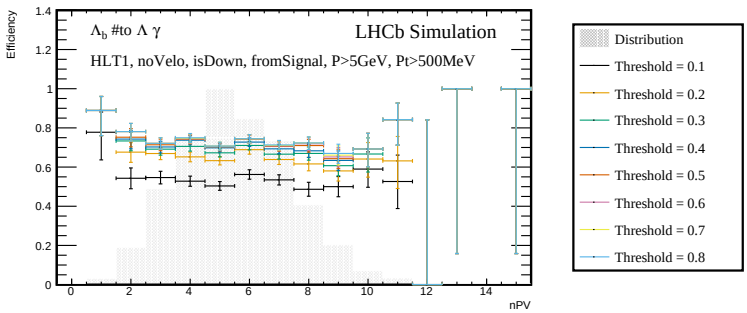
(a)



(b)

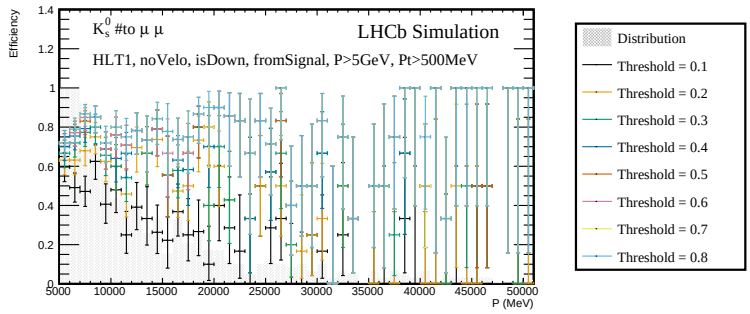


(c)

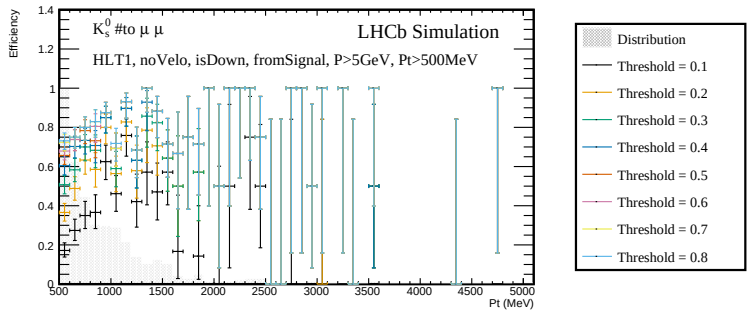


(d)

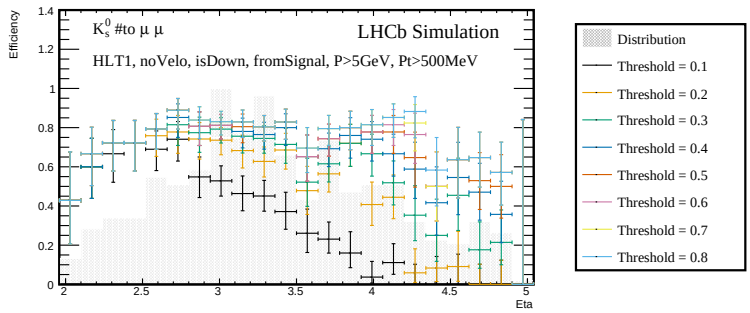
Figure 86: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $\Lambda_b^0 \to \Lambda\gamma$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



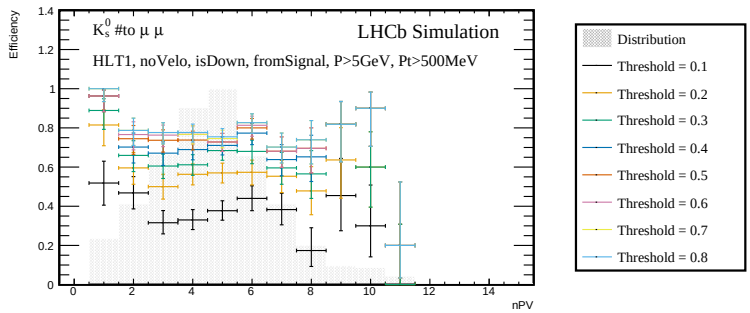
(a)



(b)

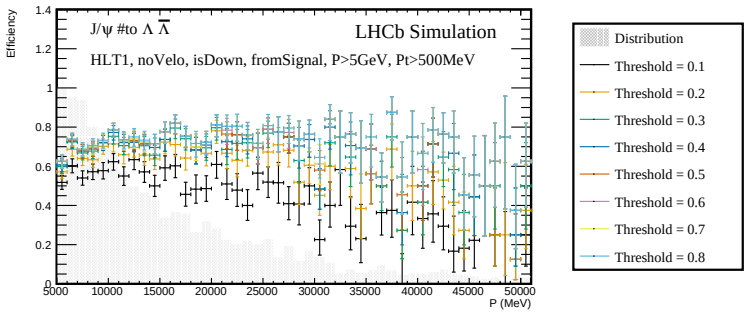


(c)

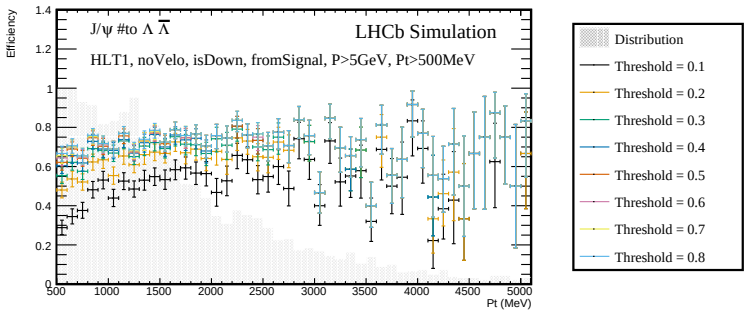


(d)

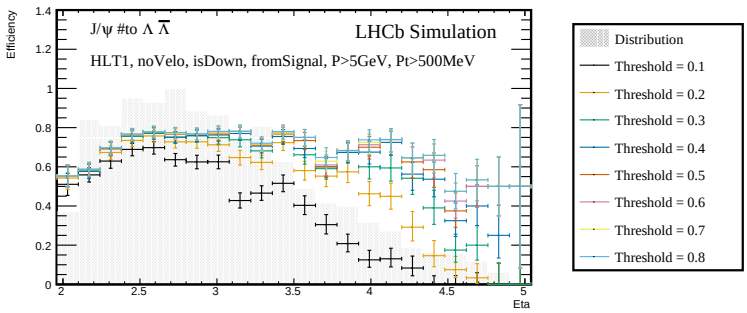
Figure 87: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $K^0_S \to \mu^+\mu^-$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV. Tracks pass through UT and SciFi detectors (isDown) but not VELO (noVELO),



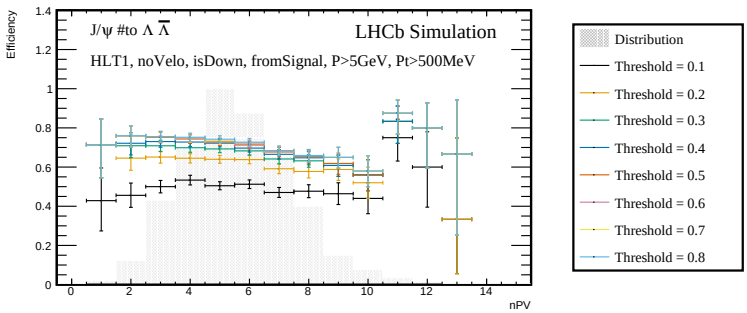
(a)



(b)

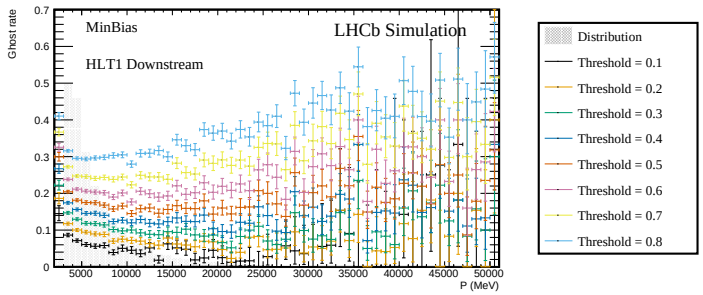


(c)

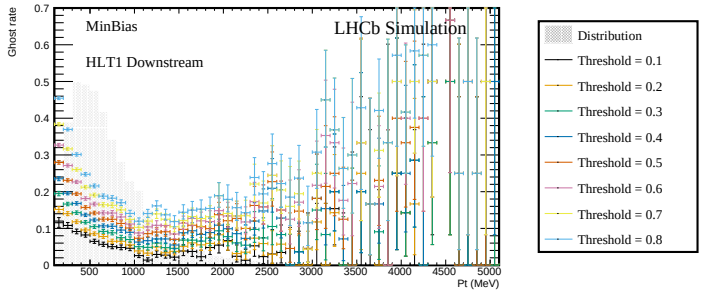


(d)

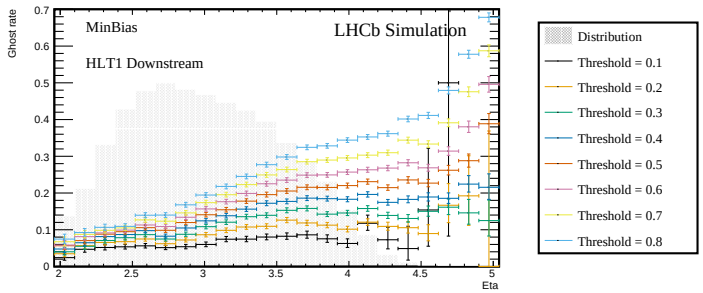
Figure 88: *Ghost Killer* threshold: for threshold values between 0 and 1, efficiency distribution of *Downstream* track reconstruction for non-electron tracks in $J/\psi \to \Lambda\bar{\Lambda}$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



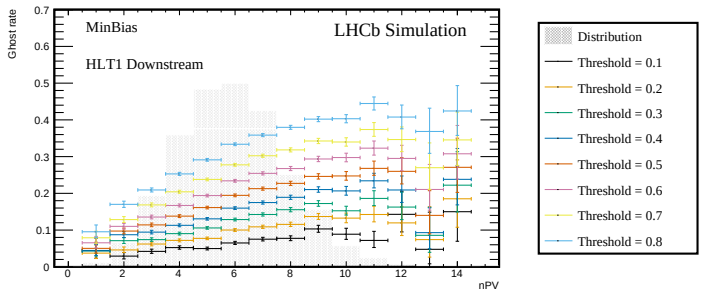
(a)



(b)

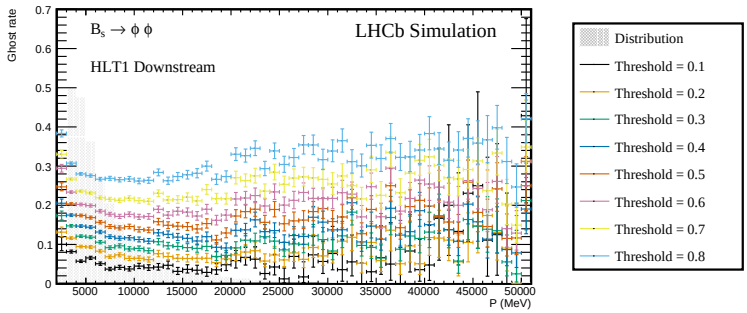


(c)

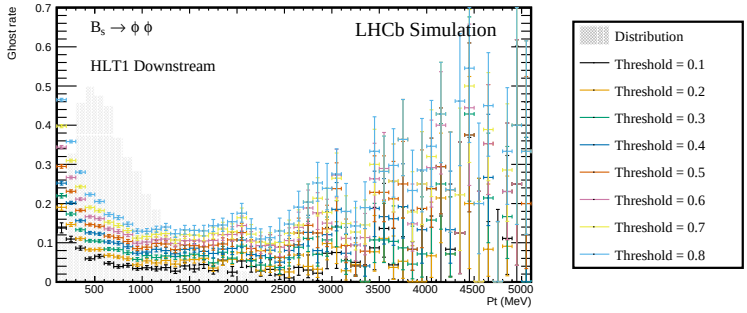


(d)

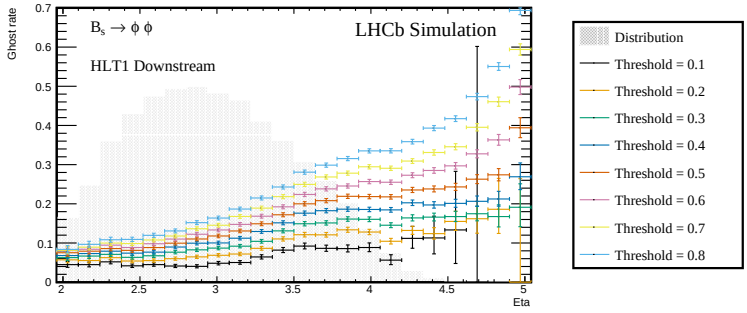
Figure 89: *Ghost Killer* threshold: for threshold values between 0 and 1, Ghost rate distribution of *Downstream* track reconstruction for non-electron tracks in *MinBias* samples (from Λ category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



(a)



(b)



(c)

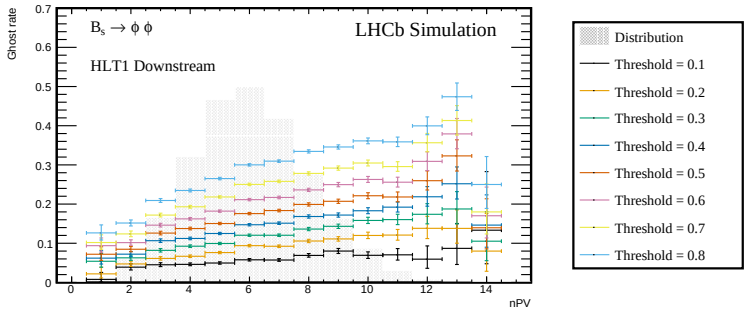
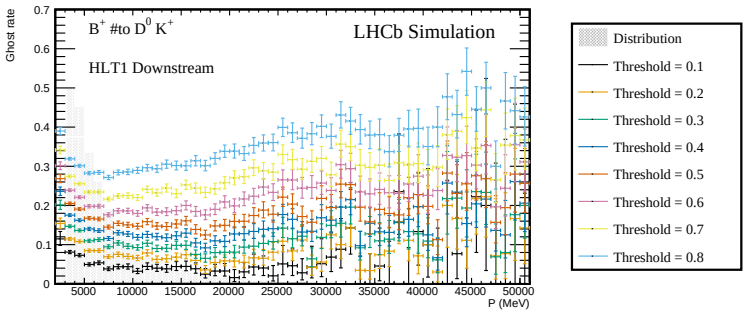
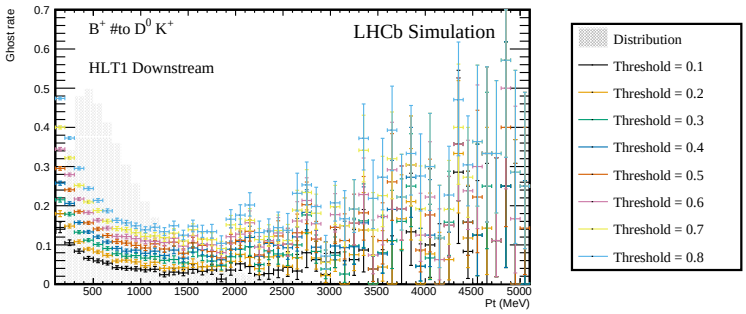


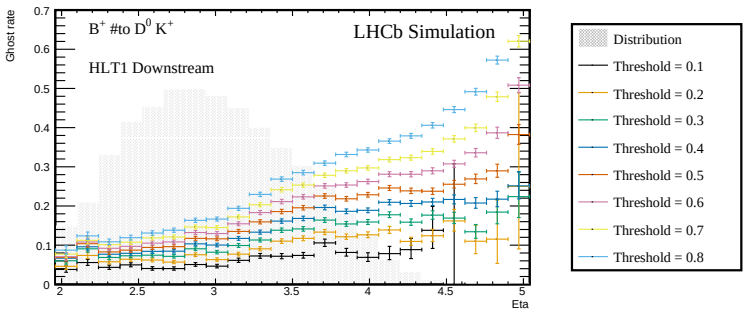
Figure 90: *Ghost Killer* threshold: for threshold values between 0 and 1, Ghost rate distribution of *Downstream* track reconstruction for non-electron tracks in $B_s^0 \to \phi\phi$ samples (from Λ category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



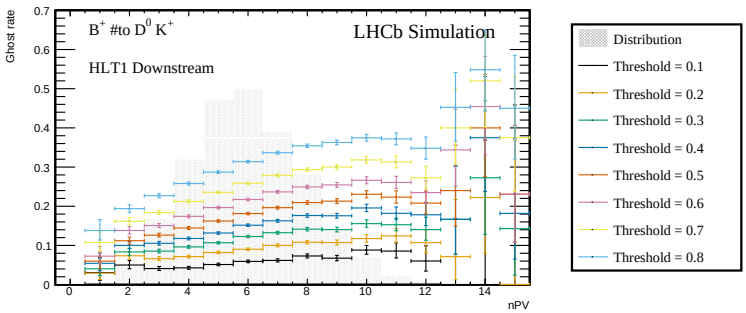
(a)



(b)

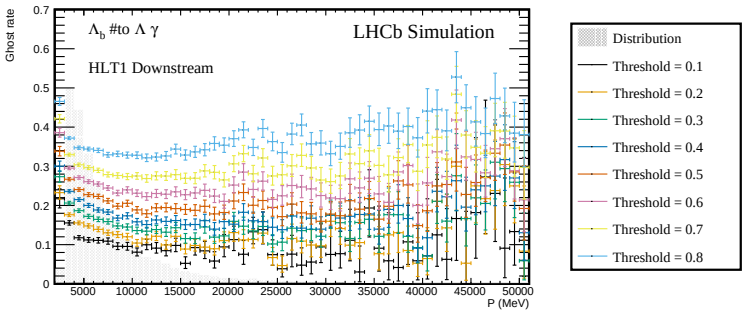


(c)

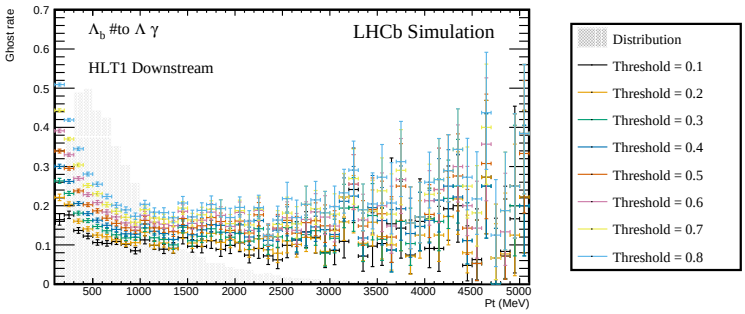


(d)

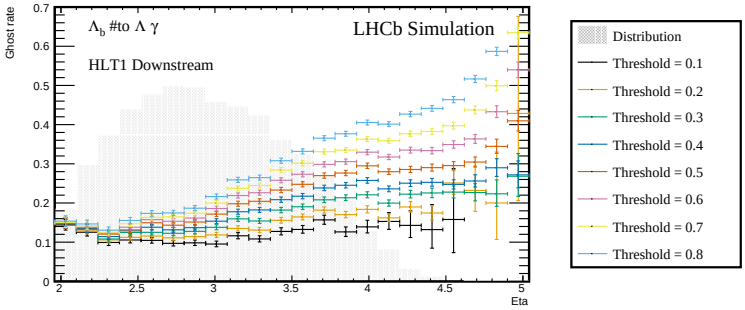
Figure 91: *Ghost Killer* threshold: for threshold values between 0 and 1, Ghost rate distribution of *Downstream* track reconstruction for non-electron tracks in $B^+ \to (D^0 \to K^0_S \pi^+ \pi^-) K^+$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



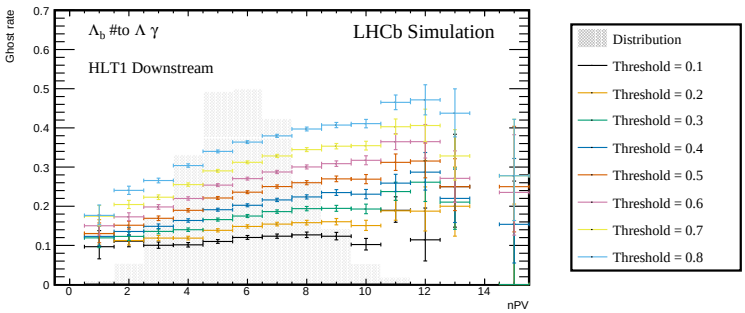
(a)



(b)

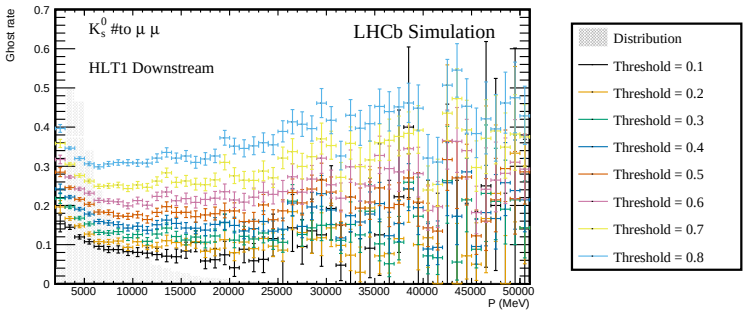


(c)

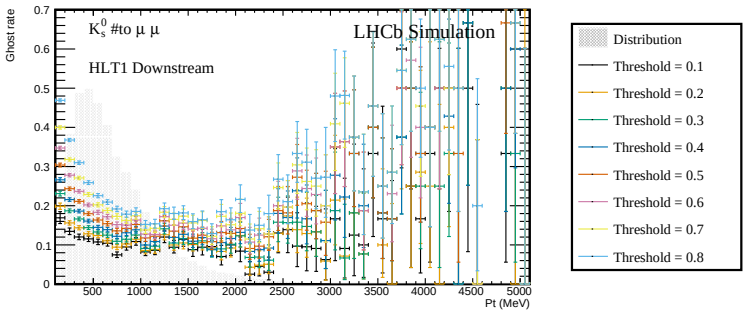


(d)

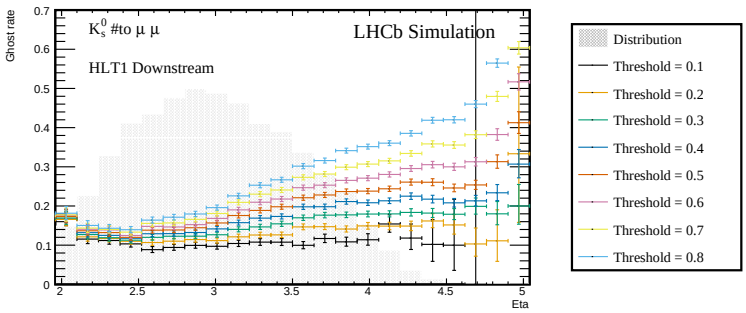
Figure 92: *Ghost Killer* threshold: for threshold values between 0 and 1, Ghost rate distribution of *Downstream* track reconstruction for non-electron tracks in $\Lambda_b^0 \to \Lambda\gamma$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.



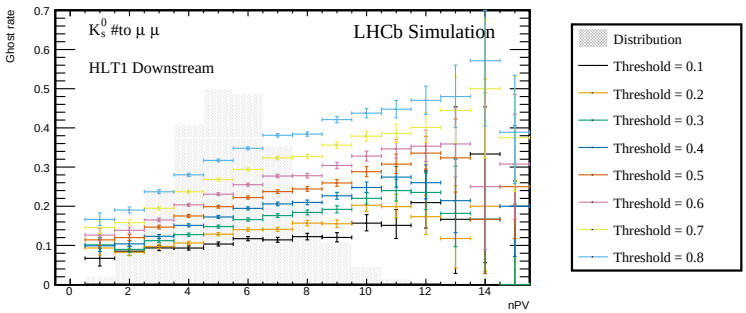
(a)



(b)

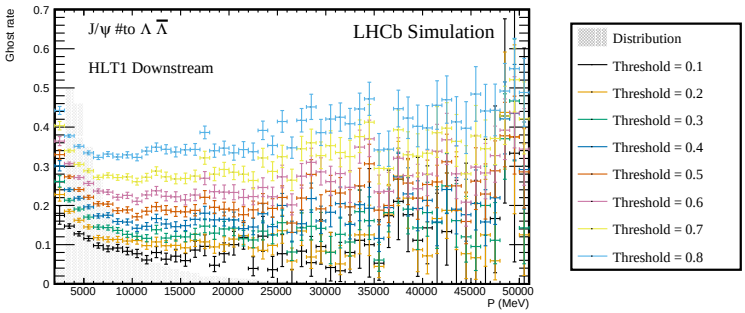


(c)

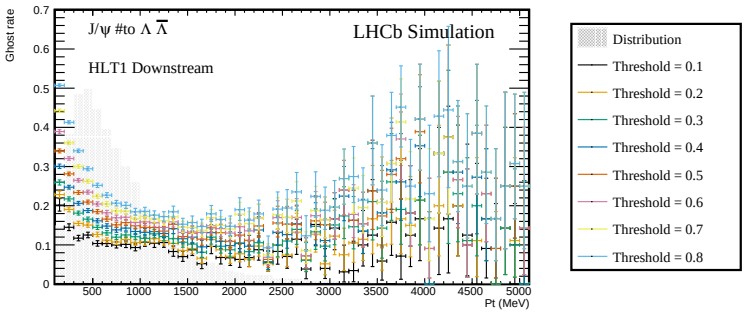


(d)

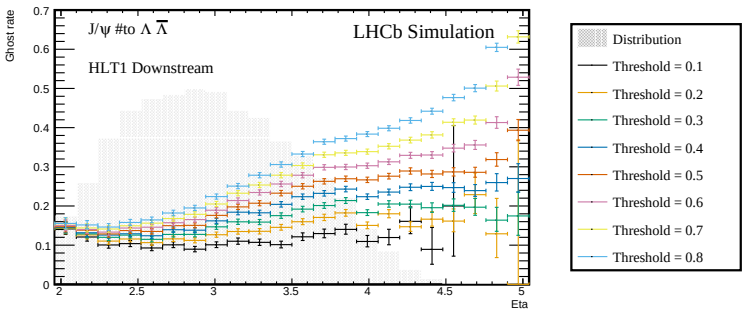
Figure 93: *Ghost Killer* threshold: for threshold values between 0 and 1, Ghost rate distribution of *Downstream* track reconstruction for non-electron tracks in $K^0_S \to \mu^+\mu^-$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV. Tracks pass through UT and SciFi detectors (isDown) but not VELO (noVELO),[163]



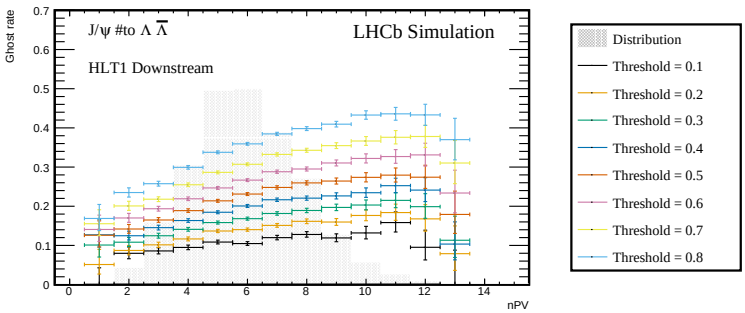
(a)



(b)



(c)



(d)

Figure 94: *Ghost Killer* threshold: for threshold values between 0 and 1, Ghost rate distribution of *downstream* track reconstruction for non-electron tracks in $J/\psi \to \Lambda\bar{\Lambda}$ samples (signal category) versus (a) p, (b) p_T, (c) η, and (d) nPV.

6

Commissioning and trigger lines using Downstream

This chapter presents the procedure to analyse the first data from Run3 which make use of the Downstream algorithm described in the previous chapter. The UT detector has been installed during the spring of 2023, and needs to be commissioned in order to allow data acquisition with the rest of the subdetectors. With this aim, a detailed commissioning procedure for the UT has been established, which is explained in Sec. 6.1.1. Dedicated trigger lines for selecting long-lived particles can be used to validate both the algorithm and the UT response. They are mainly devoted to selecting Λ and K^0_S hadrons from prompt production. The trigger lines developed with this aim, based on a newly developed fast track vertexing method and using a neural network based filtering are briefly described in Sec. 6.2.

Work described in this chapter has been carried out in collaboration with Jiahui Zhao and Volodymyr Svintozelskyi

6.1 Commissioning of LHCb during Run3

During the LHCb data taking, a small subset of the data selected by the trigger system is fully reconstructed on the LHCb online computing farm. The reconstruction produces sets of histograms that allow the subdetector performance to be assessed in real-time. These histograms are presented by the Data Quality Monitoring (DQM) software to the DQM shifter who decides whether the run is suitable for physics analysis or not, by comparing it to a reference run previously benchmarked by experts.

The same software allows the histograms to be posted with comments to an electronic logbook for discussion and further clarification. Given the potential of the Downstream algorithm to reconstruct Λ and K^0_S, these particles can be used as SM candles to control the operation of the detector, and in particular of the UT detector that has been recently installed.

Since the commissioning of the UT with collisions will start around Autumn 2023, in this thesis we present the work which will be helpful in commissioning the UT.

6.1.1 UT commissioning

As discussed in Sec. 2.3.1, the UT detector design is based on silicon sensors, $10\,\text{cm} \times 10\,\text{cm} \times 0.25\,\text{mm}$, mounted on a lightweight carbon fiber support structure, called a *stave*. The staves are used to form four large planes of silicon. Difficulties in the delivery of components and challenges in testing during the pandemic (2020-2022) led to a postponed schedule for assembly.

The UT was successfully installed on a very tight schedule during the 2022 year end technical stop. The final connections and closing around the beam-pipe were performed in time for the start of the 2023 run. Figure 95 shows the installed UT in the LHCb cavern. Fig. 96 shows how the system is operational and working from the monitors in the LHCb control room.

The commissioning stage aims to have the UT ready for the data taking during the ion run in October 2023. Starting from that month, Pb-Pb collisions are scheduled at LHCb for 27 days at 5.36 TeV, with an aim to acquire an integrated luminosity $\mathcal{L} = 0.4\,\text{nb}^{-1}$. The requirements for this are:

1. A working firmware, to be able to read hardware.
2. A working ECS (experimental control signal) to have control on the hardware.
3. A working system for detector safety.
4. Correct timing.
5. Correct pedestals and thresholds.

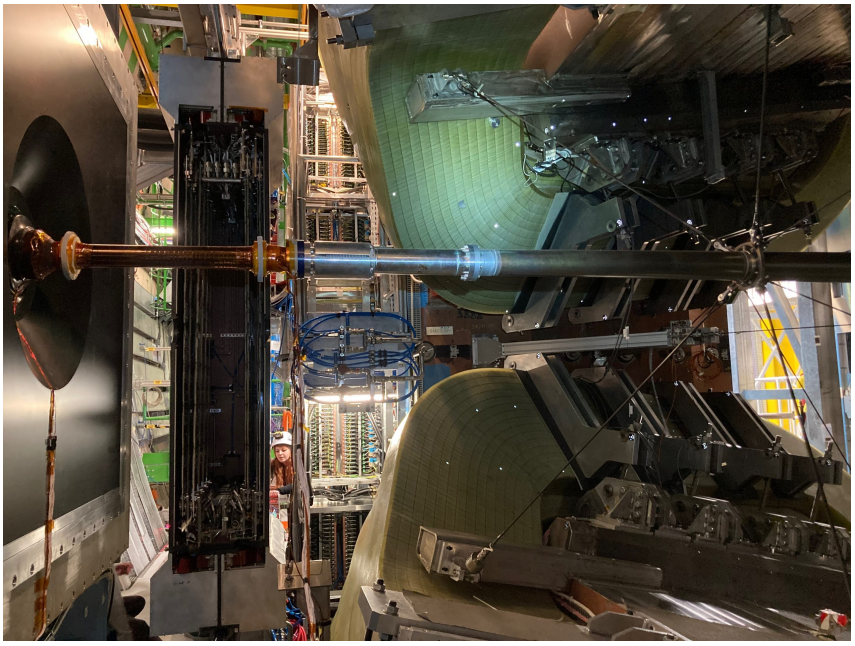


Figure 95: UT installation at the beginning of 2023.

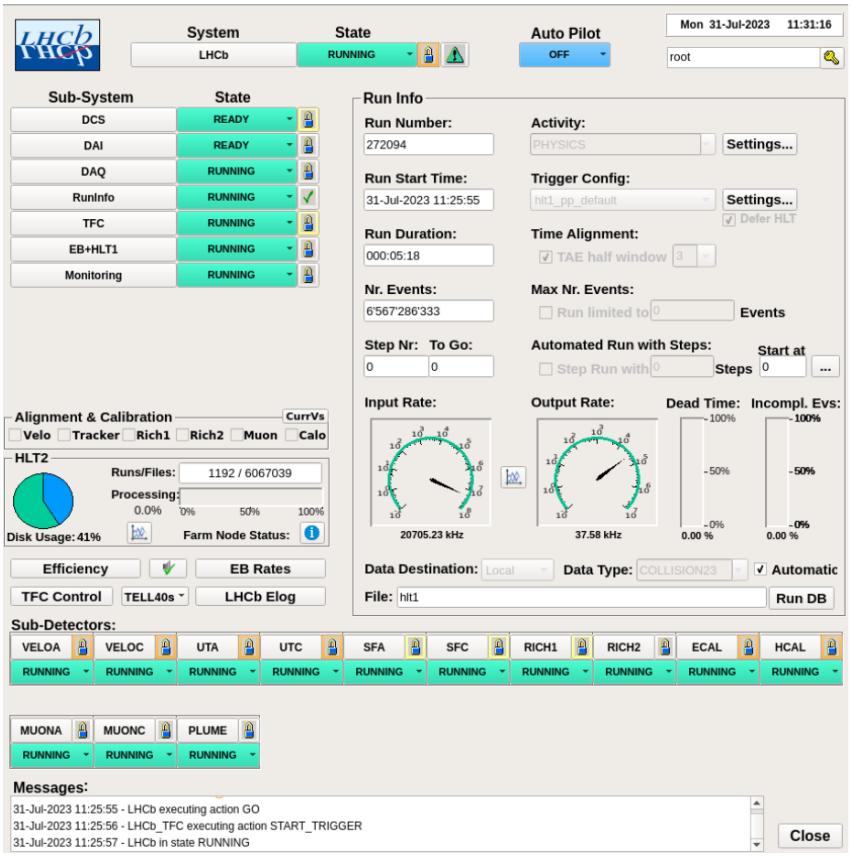


Figure 96: UT in global operation and control monitoring in the LHCb control room.

6. A working HLT1 implementation.
7. Online data monitoring for data quality and validation.

Concerning the work of this thesis, contributions towards item 6 (a working HLT1 implementation), and item 7 (online data monitoring for data quality and validation) are discussed below. The experience gained in Allen will be crucial for the HLT1 commissioning of the Downstream algorithm and selection lines. The K^0_S and Λ lines developed using *downstream* tracks will be integrated within the Monet system for Real-time monitoring during the commissioning and data-taking. The Monet system is described in Sec. 6.1.2.

6.1.2 Monet Monitoring

The data quality monitoring (DQM) at LHCb is carried out by a web-based monitoring system that uses Flask Python[1] and is executed in real-time and visualised directly in the screens of the LHCb control room. It makes use of different sets of input data:

- Control signals produced by detector electronics, HV, readout (e.g.: ADC signals).
- Unprocessed (raw) data directly from the LHCb detector.
- Higher-level quantities: mainly the output of the HLT reconstruction algorithms and selection lines (e.g.: track multiplicity).
- Fast analysis data (e.g. combined histograms, fits to shapes)

After reducing, filtering, moving and storing these data sets, the *Monitoring Data Hub* is responsible for the analysis and object visualisation. This hub is a common interface for all monitoring sources to the Monet framework [88]. It offers a uniform approach to data handling, outputting structured information in formats such as JSON, XML, or ROOT files for interpretation and display. This information is also translated to create log messages, useful in logbooks. The structure of the monitoring framework at LHCb is shown in Fig. 97.

Monet, as part of the Monitoring Data Hub, is responsible for data visualisation. It is presented and organised in configurable web pages,

[1] https://flask.palletsprojects.com/en/2.3.x/

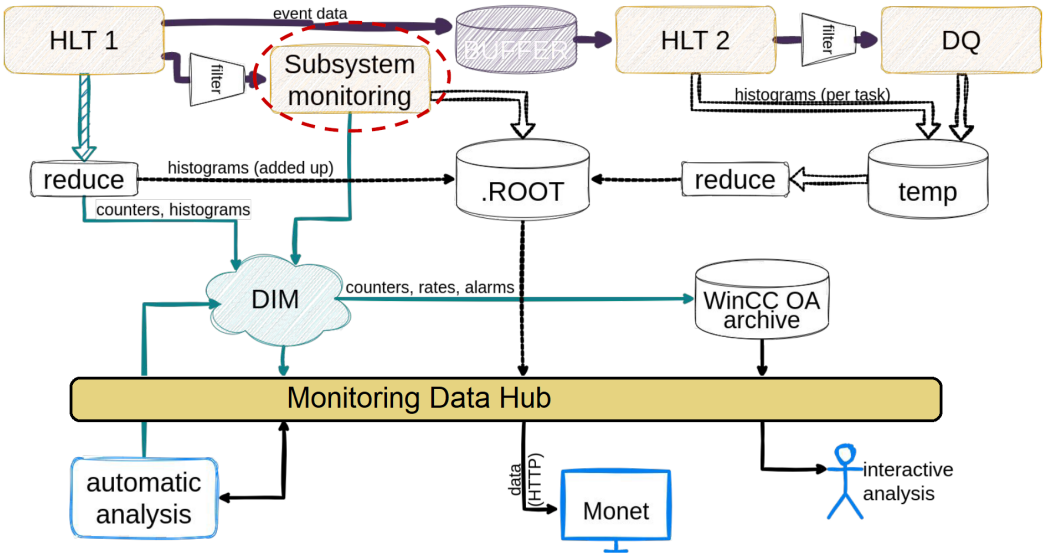


Figure 97: The structure of the monitoring framework at LHCb after the HLT1 processing. The Monitoring Data Hub allows the data visualisation, via Monet, and analysis, via the Automatic Analyses package.

with features that include:

- overlay several histograms in one plot,
- customise drawing attributes,
- overlay references,
- annotate plots,
- draw profile histograms,
- send plots to ELOG,
- display alarms.

Fig. 98 shows an example of some objects reconstructed from the calorimeter systems and displayed by the Monet framework. This interface will be used for the visualisation of output from `Downstream` lines in real-time.

6.1.3 Pre-alignment and calibration

Before using actual proton-proton beam for alignment, various subdetectors are pre-aligned and calibrated using several methods. Some of the methods which were also used during previous runs are:

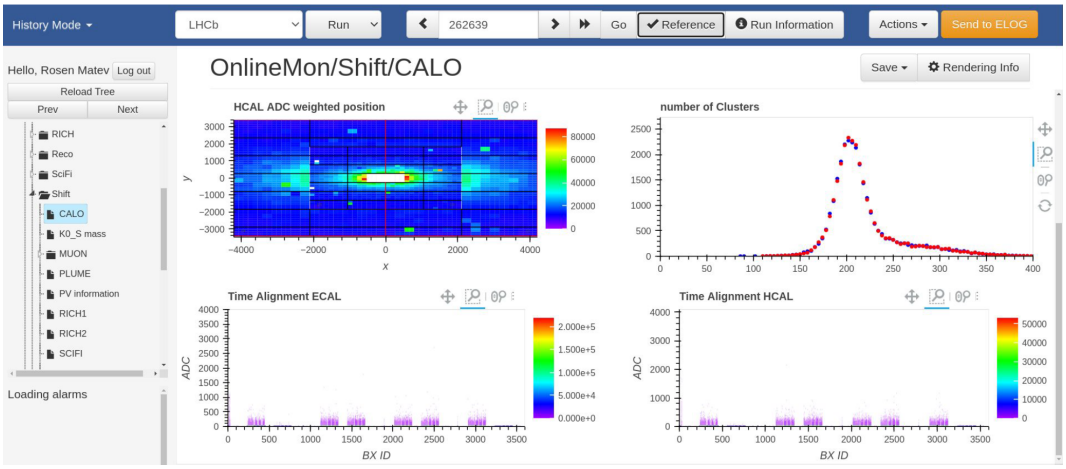


Figure 98: Display of calorimeter objects from Monet.

- Cosmic rays: cosmic triggers are excellent particles to be used for detector alignment purposes. Despite of the horizontal orientation of LHCb, part of the hundreds of cosmic events are triggered and few events contain track segments reconstructed in several tracking stations.

- TED (Three beam dumps) shots: commissioning data tracks can be collected during injection tests where the protons of the beam are dumped on an absorber at the end of the injection line, 350m downstream of LHCb, producing a lot of secondary particles. Before Run1 these tests were done with the 450 GeV SPS beams, intercepted concentrated in very short pulses (7.8-10.5 μs), and at intensities up to $\approx 5 \times 10^{13}$ protons, every 16.8 seconds. These dense particle showers allowed for an initial time and space alignment of LHCb. This was especially beneficial for both the VELO detector and the TT, as their small size resulted in very few cosmic events crossing the detectors.

- A new alignment system based on the Brandeis CCD Angle Monitor (BCAM) [89], was installed before Run2 to provide real-time monitoring of the movement of the IT stations. Two BCAMs were needed for each station and passive reactive targets were installed on each half station. The system was calibrated during the closure of the detectors. Relative movements in the z direction of up to $1 \pm 0.1\,\text{cm}$ were seen. The positioning and the alignment of the IT

stations were improved thanks to this technique.

In the following, the proposal to use trigger lines based on the Downstream algorithm with LHC collisions is explained.

6.2 Trigger lines using Downstream

This section discusses an extended Kalman filter [76] developed in HLT1 to reconstruct a mother particle from two daughter tracks. The filter aids in vertex fitting of two *downstream* tracks. The study comprises two primary components: *downstream* track extrapolation from UT to the corresponding origin vertex, and the implementation of the Kalman filter for vertex fitting with two *downstream* tracks. Thereafter, using these tools, trigger lines for selecting Λ^0 and K_s particles from prompt production are developed which are described in the subsequent subsections.

6.2.1 *Downstream* track extrapolation

The simulation study showed that the track state in the middle of the UT station, reconstructed using true UT hits information, exhibits a different t_x compared to the t_x of the state in the origin vertex, as shown in Fig. 99. This discrepancy is due to the nonzero magnet field in the y direction between the UT and the origin vertex position, as can be seen in Fig. 100. To correctly extrapolate the *downstream* track across these distances, the magnetic field vector can be employed to solve the partial differential equation using techniques like the Runge-Kutta method. However, considering the computational cost for an algorithm at HLT1 level, an approximation using a second-order polynomial function is more suitable. This is expressed in Eq 6.1 and Eq. 6.2.

$$x(z) = x_0 + t_x \cdot (z - z_0) + \gamma \cdot (z - z_0)^2 \tag{6.1}$$

$$y(z) = y_0 + t_y \cdot (z - z_0) \tag{6.2}$$

Here, z_0 is set to the z position of the midpoint in the UT station, which is the state_z obtained from the Downstream tracking algorithm. Similarly, x_0, y_0, t_x, and t_y correspond to state_x, state_y, state_{t_x}, and state_{t_y} respectively.

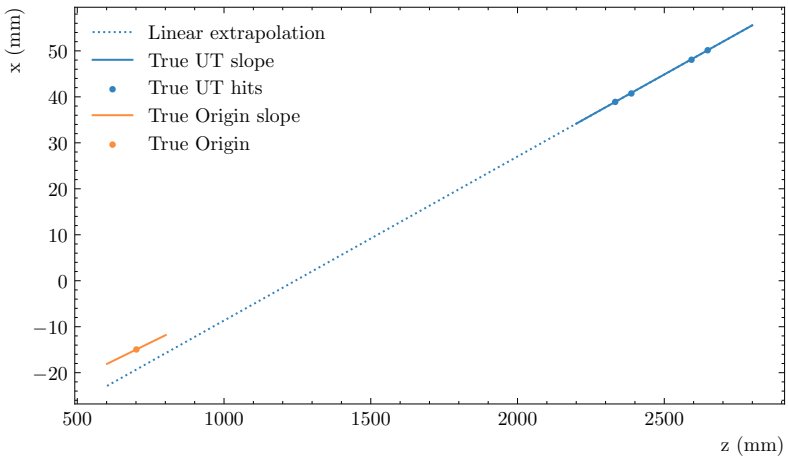


Figure 99: Linear extrapolation of True UT hits of a *downstream* track using the true slope and its displacement from the true origin vertex and slope.

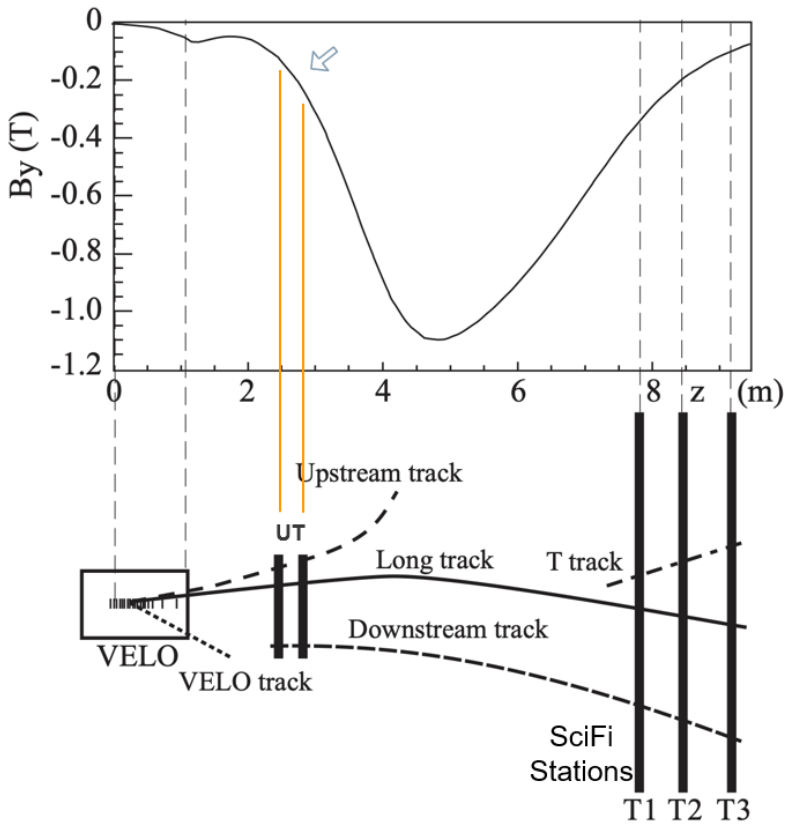


Figure 100: Magnetic field distribution B_y for LHCb detector along z.

The γ parameter serves as the only degree of freedom to account for the magnetic effect. Two different constraints can be theoretically applied:

- $x(\text{ovtx}_z) = \text{ovtx}_x$,
- $t_x(\text{ovtx}_z) = \left.\frac{dx}{dz}\right|_{z=\text{ovtx}_z} = \text{ovtx}_{t_x}$,

related to the position and the slope on the origin vertex, respectively.

Despite the validity of both constraints, since we are using this polynomial trajectory as an approximation with only one degree of freedom, we must choose one of them. For this study, we opted for the second constraint and checked the bias of the $x(\text{ovtx}_z)$ at the end.

In the simulation study, the expected γ was computed using the second constraint in Eq 6.3 as:

$$\gamma = \frac{\text{ovtx}_{t_x} - \text{state}_{t_x}}{2(\text{ovtx}_z - \text{state}_z)} \tag{6.3}$$

The expected γ demonstrates a linear dependence on the true q/p of the track, as shown in Fig. 101, allowing the estimation of this parameter from the q/p estimation in the tracking state. A comparison of the bias in the origin vertex using a second-order polynomial and a linear trajectory, as shown in Fig. 102, shows that the approximation is effective.

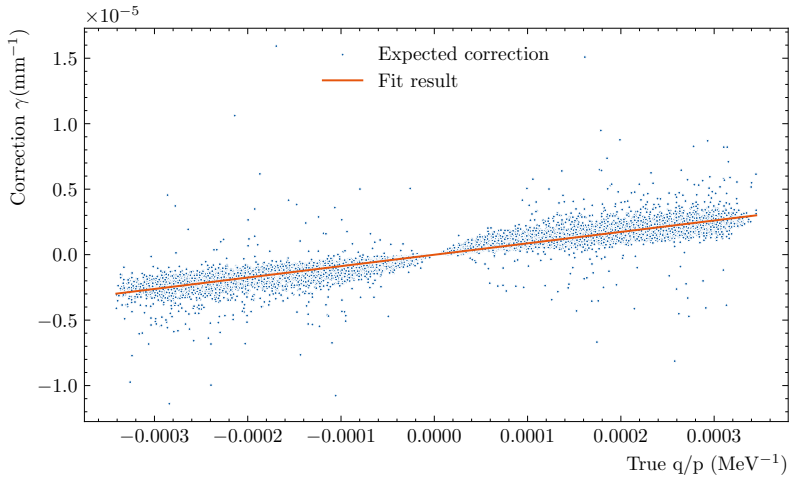


Figure 101: Comparison of the γ parameter obtained from the second-order polynomial trajectory and the true q/p.

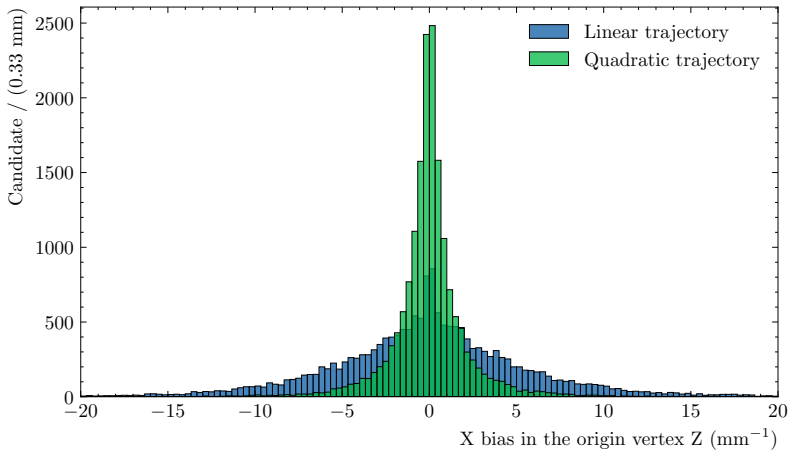


Figure 102: Distribution of the x bias in the origin vertex for second-order polynomial trajectory in (green) and the linear trajectory (blue).

Given the linear dependency, the γ correction can be parameterised, yielding the following result:

$$\begin{aligned}\gamma_(q/p) = \{ &-1.048781 \times 10^{-8} + \\ &9.347830 \times 10^{-3} \cdot q/p\} \cdot (-\text{MagnetPolarity})\end{aligned} \tag{6.4}$$

It is expected that this correction term will flip with different magnet polarities. The correction term is applied to the extrapolated track state in the UT station, and the extrapolated track state is then used for the vertex fitting.

6.2.2 Vertexing with *downstream* tracks

Before proper vertexing, the point of closest approach (POCA) can be considered as the first estimation of the vertex position of two *downstream* tracks. Considering we have two tracks A and B, the corresponding states are $\text{state}_A = (x_A, y_A, z_{UT}, t_{xA}, t_{yA}, (q/p)_A)$ and $\text{state}_B = (x_B, y_B, z_{UT}, t_{xB}, t_{yB}, (q/p)_B)$, the magnet correction in the track extrapolation is γ_A and γ_B. Then the two-track trajectories $\text{Traj}_A(z)$ and $\text{Traj}_B(z)$

are:

$$\text{Traj}_A(z) = \begin{bmatrix} x_A + t_{xA}(z - z_{UT}) + \gamma_A(z - z_{UT})^2 \\ y_A + t_{yA}(z - z_{UT}) \\ z \end{bmatrix} \tag{6.5}$$

$$\text{Traj}_B(z) = \begin{bmatrix} x_B + t_{xB}(z - z_{UT}) + \gamma_B(z - z_{UT})^2 \\ y_B + t_{yB}(z - z_{UT}) \\ z \end{bmatrix} \tag{6.6}$$

where z_{UT} is the z position in the middle of the UT, in which the track state is computed in the tracking stage.

In principle, the POCA of each track may have a different z position (POCA_z), but it is a good approximation to consider our first estimation of the vertex position z corresponds to the common z that minimizes the distance of both tracks. Then the distance $L(z)$ is:

$$L(z) = \|\text{Traj}_A(z) - \text{Traj}_B(z)\| \tag{6.7}$$

Minimizing this distance analytically involves solving a third-order linear equation, which will have three different solutions and may not be very numerically stable. An efficient numerical method considered to find this POCA_z is the Newton-Raphson method [90]. To convert the problem into something that the Newton-Raphson method can be used for, one can simplify the problem to the root-finding problem of the equation:

$$\left.\frac{\mathrm{d}L(z)}{\mathrm{d}z}\right|_{z=\text{POCA}_z} = A_0 + A_1 dz + A_2 dz^2 + A_3 dz^3 = 0 \tag{6.8}$$

where

$$A_0 = t_{xA} \cdot x_A - t_{xB} \cdot x_A - t_{xA} \cdot x_B + t_{xB} \cdot x_B + \tag{6.9}$$

$$t_{yA} \cdot y_A - t_{yB} \cdot y_A - t_{yA} \cdot y_B + t_{yB} \cdot y_B \tag{6.10}$$

$$A_1 = t_{xA}^2 - 2 \cdot t_{xA} \cdot t_{xB} + t_{xB}^2 + t_{yA}^2 - 2 \cdot t_{yA} \cdot t_{yB} + \tag{6.11}$$

$$t_{yB}^2 + 2\gamma_A \cdot x_A - 2\gamma_B \cdot x_A - 2\gamma_A \cdot x_B + 2\gamma_B \cdot x_B \tag{6.12}$$

$$A_2 = 3(\gamma_A - \gamma_B)(t_{xA} - t_{xB}) \tag{6.13}$$

$$A_3 = 2(\gamma_A - \gamma_B)^2 \tag{6.14}$$

$$dz = \mathrm{POCA}_z - z_{UT} \tag{6.15}$$

$$\tag{6.16}$$

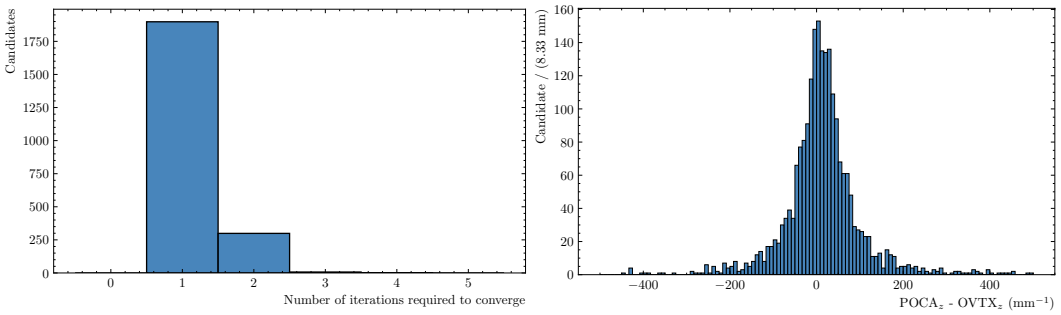
The Newton-Raphson method can converge very quickly to the solution, especially if the initial value is good. The initial value $\mathrm{POCA}_{z,0}$ can be considered as for the case in which γ_A and γ_B tend to be zero:

$$\mathrm{POCA}_{z,0} - z_{UT} = \frac{t_{xB}(x_A - x_B) + t_{xA}(x_B - x_A) + (t_{yA} - t_{yB})(y_B - y_A)}{t_{xA}^2 - 2t_{xA}t_{xB} + t_{xB}^2 + (t_{yA} - t_{yB})^2} \tag{6.17}$$

As one can observe in the Fig. 103a, the Newton-Raphson method can find the solution mostly in 3 iterations, and the bias between the found POCA_z and the true origin vertex position z is shown in the Fig. 103b. The bias has a standard deviation of around 140 mm, since the magnet correction γ is applied to Δz^2, and γ has an order of magnitude equal to 0.5×10^{-5} mm^{-2}, the expected bias in the x position would be less than 0.1 mm. This allows us to consider that the downstream tracks are extrapolated linearly in a small region around this POCA_z, which means we can first extrapolate our downstream tracks to this POCA_z and then use the vertexing algorithm of two tracks that extrapolate linearly to find the vertex.

To further optimize the fitting process, we use the extended Kalman filter to iteratively approximate the mother particle's vertex. With the initial state vector comprising the prior information about the vertex position, we progressively augment it with the track state at the vertex position, for each track, from the first to the n^{th}.

We represent the relationship between the track state and the vertex state by the projection matrix **A**:



(a) (b)

Figure 103: (a) Number of iterations for the Newton-Raphson method to converge. (b) Bias of the POCA$_z$ estimation.

$$A = \begin{bmatrix} 1 & 0 & -t_x \\ 0 & 1 & -t_y \end{bmatrix} \tag{6.18}$$

and $\mathbf{v}_i = A\mathbf{x}_i$, where $\mathbf{x}_i = \begin{bmatrix} \text{state}_x \\ \text{state}_y \\ (v_z - \text{state}_z) \end{bmatrix}_i$.

By having a diagonal covariance matrix for the track state positions $\mathbf{C}$, it can be simply inverted to $\mathbf{G}$ by using:

$$\mathbf{G} = \mathbf{C}^{-1} = \begin{bmatrix} \omega_x & & \\ & \omega_y & \end{bmatrix} \tag{6.19}$$

where $\omega_x = 1/c_{xx}$ and $\omega_y = 1/c_{yy}$.

The covariance matrix is updated iteratively as:

$$\mathbf{C}_k^{-1} = \mathbf{C}_{k-1}^{-1} + \mathbf{A}_{k-1}^T \mathbf{G}_{k-1} \mathbf{A}_{k-1} \tag{6.20}$$

and the vertexing state as:

$$\mathbf{v}_k = \mathbf{v}_{k-1} + \mathbf{C}_k \mathbf{A}_{k-1}^T \mathbf{G}_{k-1} \mathbf{r}_{k-1} \tag{6.21}$$

where $\mathbf{r}_k = \begin{bmatrix} \text{state}_{kx}(z = v_{kz}) - v_{kx} \\ \text{state}_{ky}(z = v_{kz}) - v_{ky} \end{bmatrix}$ is the residual vector.

The χ^2 is updated as follows:

$$\chi^2_k = \chi^2_{k-1} + \tag{6.22}$$

$$(\mathbf{v}_k - \mathbf{v}_{k-1})\mathbf{A}^T_{k-1}\mathbf{G}_{k-1}\mathbf{r}_{k-1} + \tag{6.23}$$

$$\mathbf{r}_{k-1}\mathbf{G}_{k-1}\mathbf{r}^T_{k-1} \tag{6.24}$$

The decrease in χ^2 between iterations indicates the convergence of the fitting process. We consider a change smaller than 0.01 as the convergence condition for the vertexing.

We limit the number of iterations to three for each track pair due to computational reasons. This approach is sufficient for most real cases, as illustrated by the simulation studies and shown in Fig. 104. After three iterations, all combinations that do not converge are discarded.
If the vertexing is not converging at the end of the iteration, we have to update the track slope $m = (t_x, t_y)$ as follows:

$$\mathbf{m}_k = \mathbf{m}_{k-1} + \mathbf{B}^T_{k-1}\mathbf{G}_{k-1}\mathbf{r}_{k-1} \tag{6.25}$$

where, $\mathbf{B}$ is the matrix describing the correlation between position and slope.

$$\mathbf{B} = \begin{bmatrix} c_{x,t_x} & c_{x,t_y} \\ c_{y,t_x} & c_{y,t_y} \end{bmatrix} = \begin{bmatrix} c_{x,t_x} & 0 \\ 0 & c_{y,t_y} \end{bmatrix}. \tag{6.26}$$

The final comparison between this vertexing algorithm and the direct sum of the track four-momenta for Λ and K^0_s masses demonstrates the effectiveness of this approach. This is shown in Fig. 105. The χ^2 of the vertexing is also used for the development of *downstream* track lines.

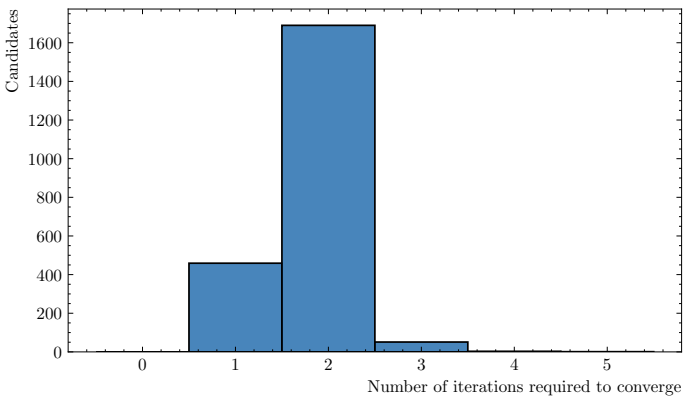
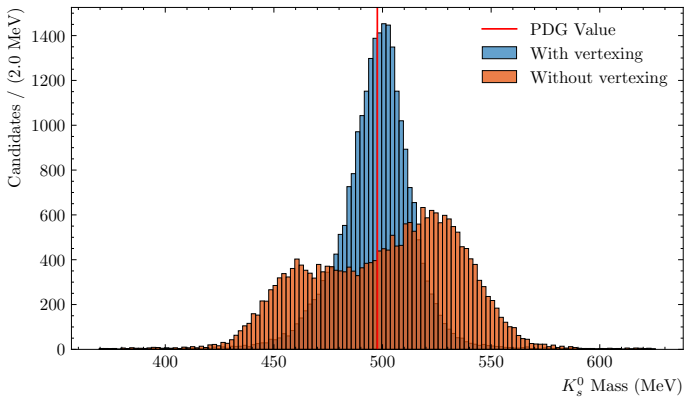
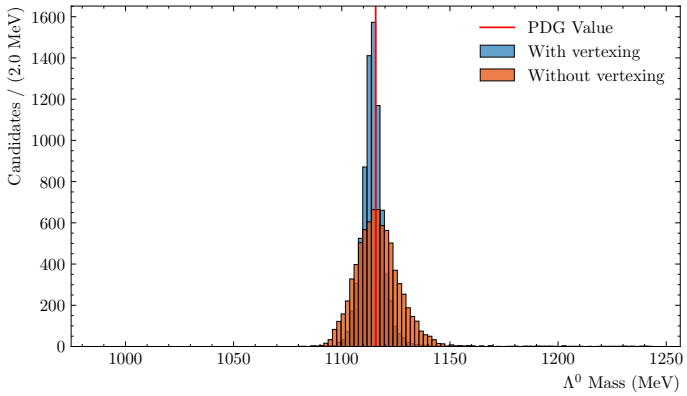


Figure 104: Number of iterations required to converge all the candidates



(a)



(b)

Figure 105: Comparison of the vertexing algorithm with the direct sum of the track four-momenta for Λ and K^0_S masses.

6.2.3 HLT1 Selection lines for K^0_S and Λ

Taking the output from the Downstream algorithm and after performing the vertex fitting as described in the previous step, various cut-based selection line approaches were explored for persisting the events with K^0_S and Λ candidates using the traditional HLT1 line approach. However, having developed the Neural Network (NN) framework discussed in the previous chapter (see Sec. 5.4), it was now possible to use the same framework for developing the selection lines. Two separate NNs were developed, each for K^0_S and Λ candidates. The NN used for both cases was a simple feed-forward NN with an architecture similar to the one discussed in Sec. 5.4. As shown in Fig. 106, the input layer consisted of 12 features from the track parameters of the two daughter tracks, the vertex parameters, and the mother track parameters. A single hidden layer consisted of 7 nodes, and there was a single-node output layer.

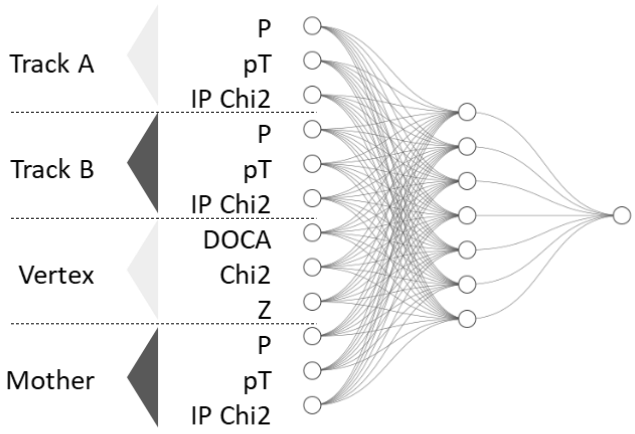


Figure 106: NN architecture for selection lines.

The daughter tracks of $\Lambda(\to p\pi^-)$ and for $K^0_S(\to \pi^+\pi^-)$ were required to have opposite charges. The models were trained using upgrade condition MinBias samples[2] and were split into 50% for training and 50% for testing.

Fig. 107 shows the distribution of K^0_S selection efficiency on the y-axis vs the number of nodes in the hidden layer on x-axis. To meet the

[2] /MC/Upgrade/Beam7000GeV-Upgrade-MagDown-Nu7.6-25ns-Pythia8/Sim10aU1/13104012/XDIGI

HLT1 throughput requirements, it is important to keep the size of the NN small. The efficiency of the selection line was found to be stable for hidden layer size of 7 nodes. The same was used for Λ selection line.

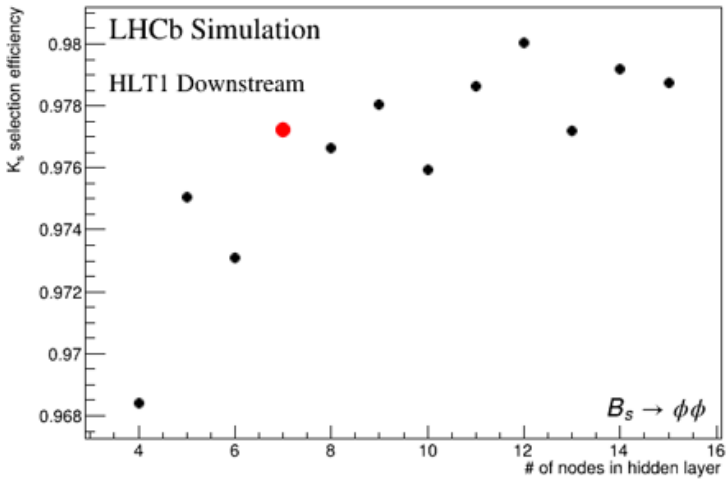
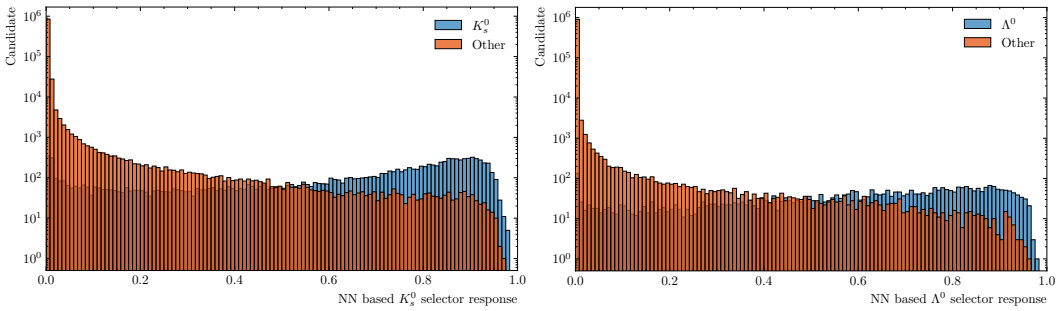


Figure 107: NN size vs efficiency for K^0_S selection line.

The training was stopped after 21397 and 27832 epochs for K^0_S and Λ, respectively. The training and testing loss for K^0_S are 0.0150 and 0.0153 and for Λ are 0.00454 and 0.00480. The training and testing loss for both the cases are similar and the model is not overfitting. Fig. 108a and Fig. 108b shows the classifier score for K^0_S and Λ, respectively. The classifier score is defined as the output of the NN with values between 0 and 1. The threshold for the selection line was set to 0.5. The efficiency of the selection line was found to be 70% and 60% for K^0_S and Λ, respectively, and the ghost rejection was found to be 99.8% for both the cases.

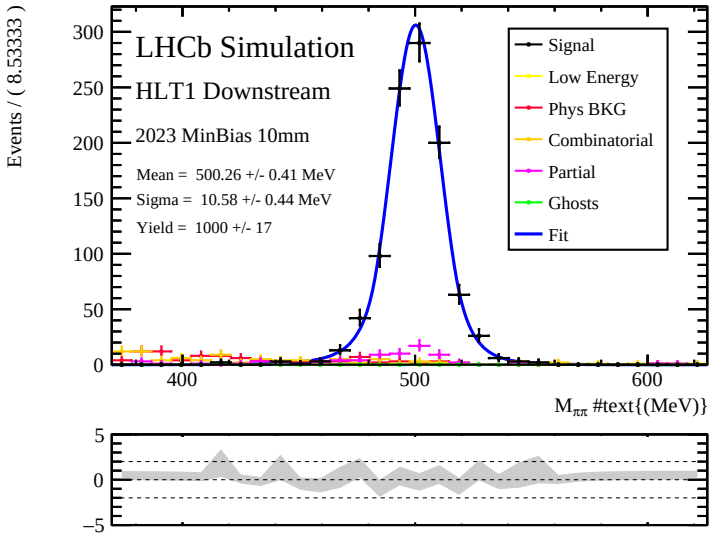
Using Run3 nominal conditions, the mass fit performed for K^0_S and Λ candidates is shown in Fig. 109a and Fig. 109b respectively.

At HLT1 level the throughput is one of the key metrics to assess the performance. Fig. 110 shows the throughput of the selection lines for K^0_S and Λ candidates. The throughput of the selection line sequence, including these two lines along with the vertexing, is compared with the baseline reconstruction sequence which includes Downstream sequence. The difference of throughput on the production GPU card A5000 is only 4% of the total HLT1 budget.

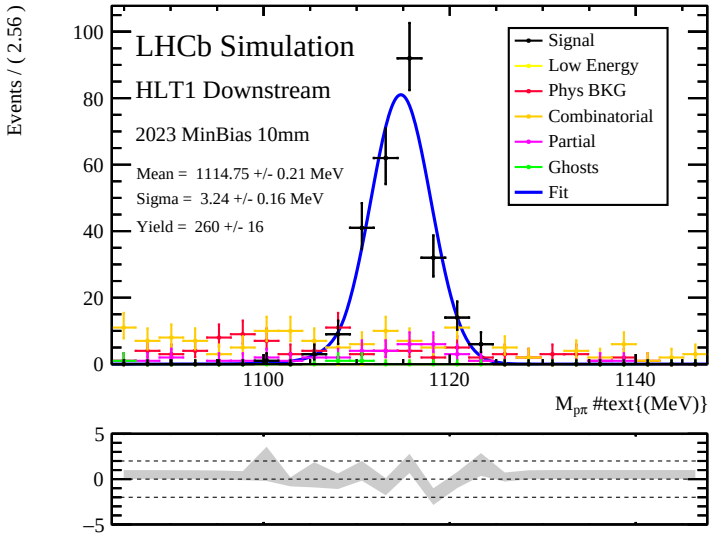


(a) (b)

Figure 108: Classifier score for (a) K_S^0 and (b) Λ Ghost killer NN.



(a)



(b)

Figure 109: Mass fit for (a) K_S^0 and (b) Λ candidates. A Crystal-Ball function [91] has been used to describe the signal.

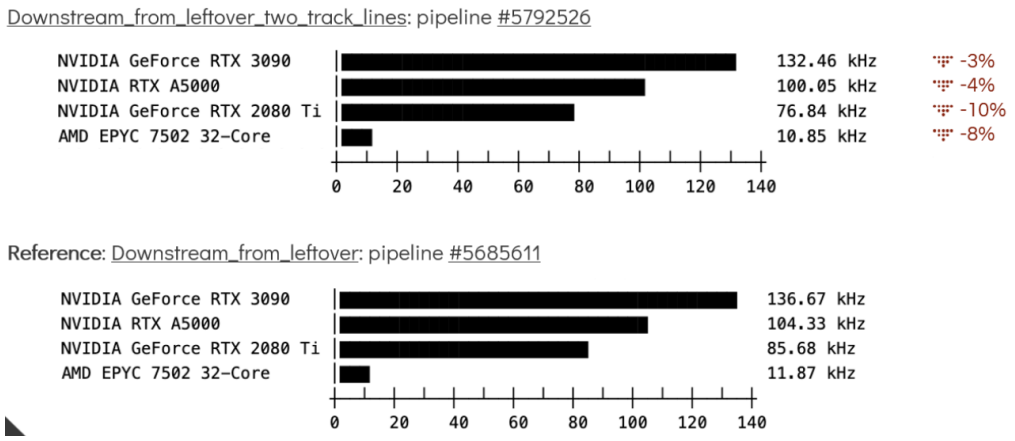


Figure 110: Throughput of the selection lines for K_S^0 and Λ candidates.

6.3 UT alignment and calibration using *downstream* tracks

The alignment and calibration of the detector are essential for achieving the best possible physics performance of the experiment. During Run2, a fully automated real-time alignment and calibration procedure was pioneered by the LHCb. This approach is critical for Run3, where the trigger is purely software-based. The alignment and calibration of VELO and SciFi [92], used *long* tracks which traverse all the tracking subdetectors. Similarly, in addition to the *long* tracks, the alignment and calibration of the UT can be performed using the output of *downstream* tracks and the *downstream* selection lines detailed in Chapter 5 and Sec. 6.2 using the procedure described in the following section.

6.3.1 Alignment Model and Error Handling

- **Alignment model**: using *downstream* tracks, the alignment of the UT is performed by minimizing the χ^2 of all tracks with respect to the alignment parameters α, where α describes the translation and rotation degrees of freedom of each alignable detector element.
- **Residuals**: the residuals are defined as the discrepancy between the measured positions (m) and the expected positions derived from the track model (h), considering both track parameters (x) and align-

ment parameters (α):

$$r = m - h(x, \alpha).$$

- **Alignment parameters**: the alignment parameters, representing translations and rotations of the detector elements, are specified, along with any applicable physical constraints or priors.
- **Covariance Matrices**: the measurement covariance matrix (V) and the residuals' covariance matrix (R) are defined, encapsulating the uncertainties in the measurements and the correlations between different measurements.

Minimisation of χ^2 and Iterative Procedure

- **Derivatives**: the first and second derivatives of χ^2 with respect to α are computed as follows:

$$\frac{d\chi^2}{d\alpha} = 2 \sum_{\text{tracks}} \frac{dr}{d\alpha}^T V^{-1} r, \quad \frac{d^2\chi^2}{d\alpha^2} = 2 \sum_{\text{tracks}} \frac{dr}{d\alpha}^T V^{-1} R V^{-1} \frac{dr}{d\alpha}.$$

- **Iterative update**: the alignment parameters are iteratively updated using the Newton–Raphson method [90], with numerical techniques applied for stability:

$$\alpha_1 = \alpha_0 - \left(\frac{d^2\chi^2}{d\alpha^2} \right)^{-1} \frac{d\chi^2}{d\alpha}\bigg|_{\alpha_0}.$$

- **Convergence criteria**: specific convergence criteria are established, including thresholds for changes in χ^2 or alignment parameters, and a maximum number of iterations to ensure a meaningful convergence.

6.3.2 Real-time alignment and calibration tasks

The real-time alignment and calibration of different subdetectors as shown in Fig. 111 are performed using the following general procedure:

- **Analyzer and Iterator Processes**: the alignment procedure is bifurcated into two components: the *analyzer*, which reads current alignment constants, reconstructs tracks, calculates derivatives, and saves them to a binary file; and the iterator, which aggregates the

files, performs minimisation, assesses convergence, and instigates updates if required.

- **Execution Timing**: the alignment is executed using a subset of the event-filter farm nodes, leveraging multi-threading. The alignment of the tracking system is anticipated to be completed within minutes, whereas the RICH mirror alignment may necessitate hours.
- **Real-Time Calibration**: This includes the evaluation of the refractive index for the RICH and the calibration of the ECAL high voltage, thereby optimizing the calorimeter performance.

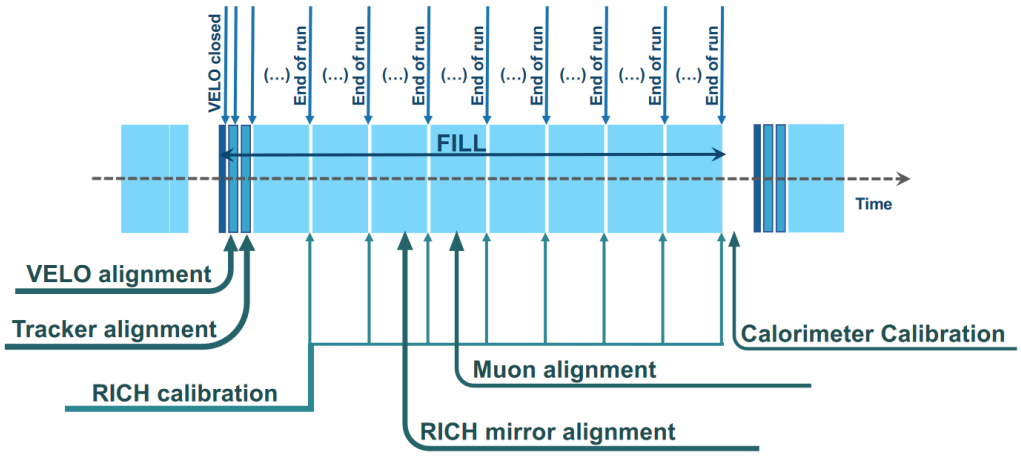


Figure 111: Real-time alignment and calibration tasks.

7 Physics impact of the Downstream algorithm

In this chapter, the expected impact on physics due to the inclusion of the new Downstream algorithm inside the LHCb framework is shown. The increase of the physics potential to detect LLPs beyond the SM is explained, making use of a Higgs dark boson model in Sec. 7.1, and of a composite Higgs model in Sec. 7.2. In Sec. 7.3 the channels which could also benefit from the Downstream algorithm in the SM are described in the *beauty*, *charm* and *strange* sectors. The work in this chapter has been performed in collaboration with Diego Mendoza, Louis Henry, Jiahui Zhuo, Valerii Kholoimov and Volodymyr Svintozelskyi.

7.1 Impact of the Downstream algorithm to detect new particles in the hidden sector

As it was explained in Chapter 1, many models beyond the SM predict particles which could have long lifetimes. One example is a model with a Higgs field serving as the portal to a dark sector, which could accommodate dark matter candidates [93]. It predicts the existence of a mixed state between a new scalar low-mass boson (H') and the SM Higgs (H), regulated by the mixing strength θ.

In this model, the new H' can be interpreted as a mediator to a dark sector, of unknown mass and lifetime. The model could be validated through the experimental signature of the decay $B \to \mathrm{H}'K$, with the H' decaying into $\pi^+\pi^-, K^+K^-, \mu^+\mu^-$, or $\tau^+\tau^-$, depending on its mass. A displaced vertex could be determined, allowing to reconstruct the H'

mass from the kinematics and identification of the two decay particles. The sensitivity to this model depends nevertheless on the H′ mass and lifetime, which could lead the new scalar to decay outside the VELO. In particular, if the H′ has long lifetime, the two final decay particles could only be selected if they are reconstructed by the Downstream algorithm.

Figure 112 shows the decay probabilities of this dark boson into different decay channels, including two leptons, as function of the H′ mass and normalised to unity.

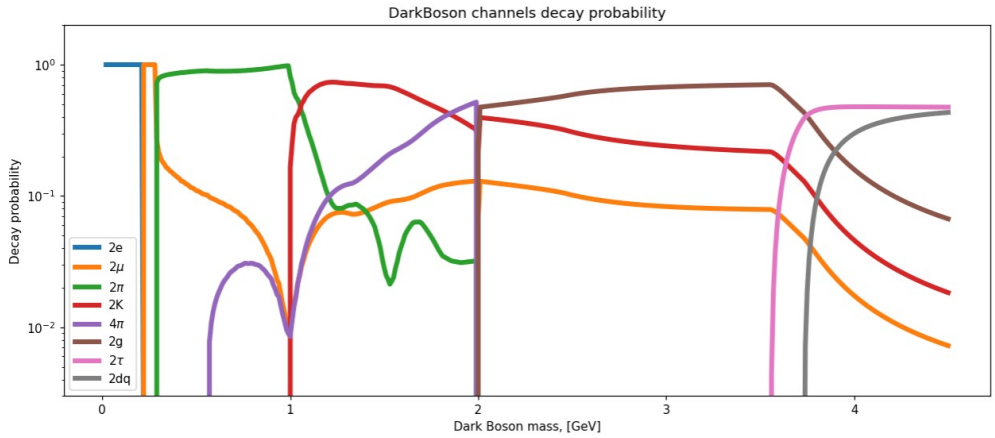


Figure 112: H′ decay probabilities as function of its mass. The decay into two muons corresponds to the orange line. Calculations by Maksym Ovchynnikov, private communication.

Considering only the leptonic decay mode H′$\to \mu^+\mu^-$, the decay rate can be expressed by:

$$\Gamma\left(\mathrm{H}' \to \ell\ell\right) = \sin^2\theta \frac{G_F m_{\mathrm{H}'} m_\ell^2}{4\sqrt{2}\pi}\left(1 - \frac{4m_\ell^2}{m_{\mathrm{H}'}^2}\right)^{3/2}, \tag{7.1}$$

where G_F is the Fermi constant and m_l the lepton mass. The H′ lifetime can thus be computed:

$$\tau_{H'} = \frac{1}{\Gamma\left(H' \to \mu^+\mu^-\right)}. \tag{7.2}$$

We can study the sensitivity of the LHCb experiment to the H′ decay into the $\mu^+\mu^-$ final state, and the expected effect of the Downstream algorithm [94]. For this decay channel the Higgs mass is bounded to be $m_H > 2m_\mu \approx 212\,\mathrm{MeV/c^2}$. The H′ mass is also constrained to

$m_{H'} < m_{B^+} - m_{K^+} \approx 4700\,\mathrm{MeV/c^2}$. Using the upgraded LHCb simulation and Pythia8 assuming Run3 beam conditions, 77 Monte Carlo (MC) samples of 7000 events each have been simulated. The $B \to H'(\to \mu^+\mu^-)K$ decay channel has been generated considering H' masses in the range of 500 - 4500 MeV and lifetimes from 1 to 2000 ps. The decay vertex of the H' is expected to be displaced with a dependence on these variables, and they will be labelled in the following according to the track type of the two muons (two *long* tracks = *LL*, two *downstream* tracks = *DD* and two *T*-tracks = *TT*). *LL* vertices are thus expected to be produced in the VELO detector, *DD* are produced between the VELO and the UT, and *TT* vertices are produced between the UT and SciFi. The reconstructibility of these vertices is defined according to the track reconstructibility already explained, and imposing that the two muons are coming from the same vertex, the decay vertex of the H'. Figure 113 shows the reconstructibility of the decay vertex of the H' particle as a function of its mass and lifetime. For lifetimes below 10 ps, a large proportion of *LL* vertex topologies is found, as expected, where the H' decays in the VELO acceptance and both muons can be reconstructed as *long* tracks. Nevertheless for H' lifetimes larger than 100 ps (and small mixing angle), most of the decays are produced downstream from the VELO, resulting in a large proportion of the *DD* and *TT* topologies. These fractions are very similar in the case that the H' decays into two hadrons[1]. Figure 114 shows the present LHCb HLT1 effect when triggering on the H' decay products (Trigger on Signal (TOS)). Since until now only long tracks are reconstructed at the HLT1 level, a high inefficiency can be observed for large H' lifetimes, going down to 10% for lifetimes larger than 500 ps. A loss in sensitivity for small H' masses is also observed, since the H' is experiencing larger boosts, muons are escaping from detection in the VELO.

[1]One should note that the tracking reconstruction at this level does not include particle identification.

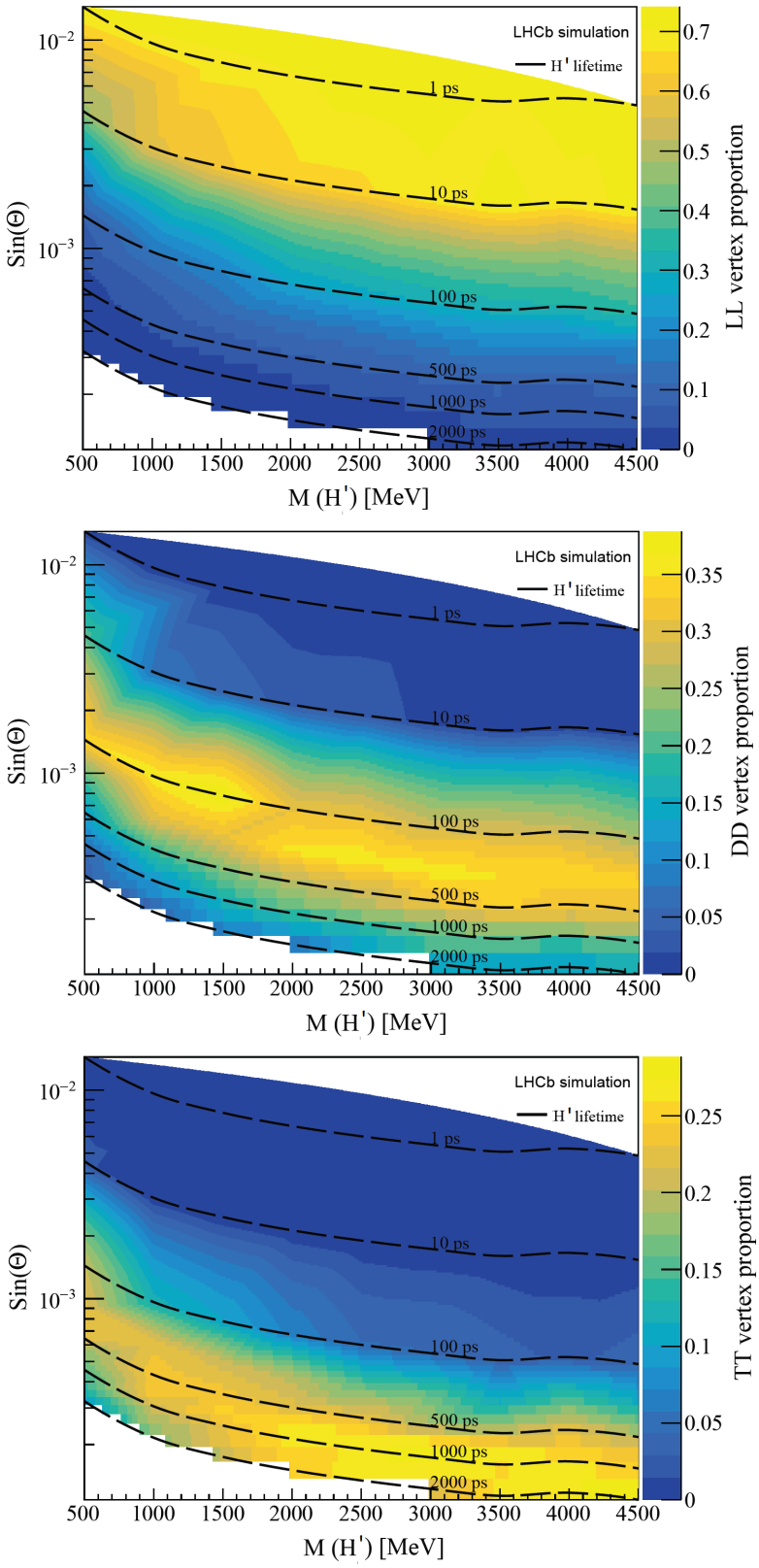


Figure 113: Reconstructibility of the decay vertex of the H′ particle as a function of its mass and lifetime. Decay topologies are shown, from top to bottom, in the order: *LL*, *DD*, *TT*, corresponding to the track types of the two muons.

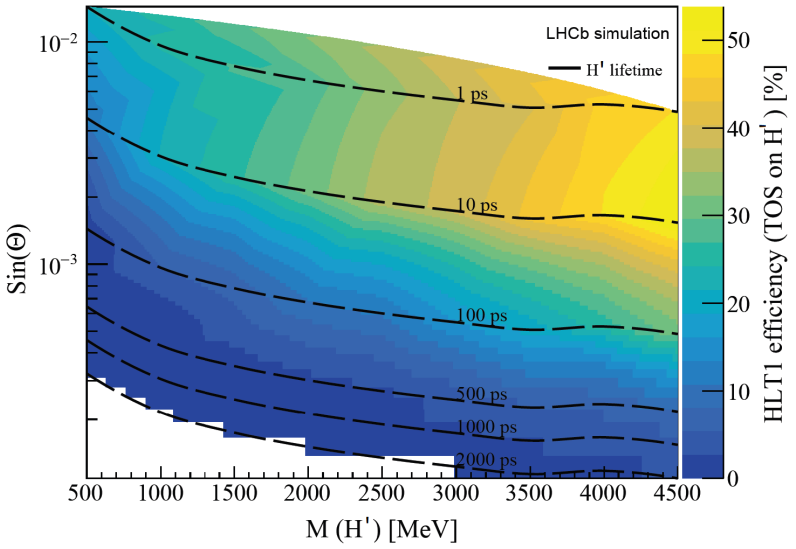


Figure 114: Proportion of events triggered by the HLT1 decision on the H′ decay products (Trigger on Signal (TOS))[94].

7.2 Impact of the Downstream algorithm to detect new particles in a composite Higgs model

Composite Higgs models are also proposed to solve the problem of naturalness in the SM. The idea is that the Higgs is not anymore a point-like particle, but rather a composite system with an specific geometrical size L_H. A new strong force is proposed which characterizes this Higgs bound state at a confinement scale of order $m_* = 1/L_H$. In this way, since the low energy quanta have very large wavelength, they are not able to resolve the Higgs size, and its mass is not longer sensitive to high energy quadratic corrections. In some of these models, in addition to a new heavy vector bosons (V), new light scalars (a_1, a_2) can appear, which could have long lifetimes. Some of the proposed channels to verify the model are the $B_s^0 \to a_1 a_2$ and $B_s^0 \to K^+ a_1 a_2$, where the a_1 and a_2 decay into two muons [95].

Since the light scalars couple to leptons, the decay width for such processes can be written as

$$\Gamma(a_1 \longrightarrow l^+ l^-) = \frac{g_1^2 y_l^2}{8\pi} m_{a_1} \left(1 - \frac{4m_l^2}{m_{a_1}^2}\right)^{3/2}, \tag{7.3}$$

with m_l and m_{a_1} the masses of the lepton and the a_1 particle, y_l the

Yukawa coupling, and g_1 a free dimensionless parameter. The decay width is:

$$\Gamma(a_1 \longrightarrow \mu^+\mu^-) = \frac{1}{\tau_{a_1}}, \tag{7.4}$$

which links the coupling constant g_1 with the lifetime.

The effect of the Downstream algorithm can be tested for a specific case, using the $B^+ \to K^+ a_1(\to \mu^+\mu^-)a_2(\mu^+\mu^-)$ decay channel. One of the light scalar (a_1) could be long-lived, reaching lifetimes of the order of ps or ns. Several simulations are performed, consisting of 44 samples of 1000 events each, and with masses and lifetimes ranging between 500 and 2000 MeV/c^2 in steps of 500 MeV/c^2 and τ=1, 10, 100, 250, 500, 750, 1000, 1250, 1500, 1750 and 2000 ps. As in the previous section, one can study the track reconstructibility of the two muons coming from the a_1 as function of its mass and lifetime (or coupling constant g_1). Fig. 115 shows the expected distribution of *LL*, *DD* and *TT* tracks distributions. As it can be seen, these plots present a similar pattern to the ones in the previous section, and emphasize the importance of the downstream and *T-track* reconstruction for lifetimes of 100 ps or larger.

Figure 116 shows the effect of the HLT1 trigger during the Run2 for this decay, when the Trigger On signal (TOS) is considered for the a_1 particle. As it can be seen, the sensitivity is very limited to regions of low lifetimes and large masses, as compared to what it could be reconstructed as shown in the previous plots (see Fig. 115), due to the fact that only *long* tracks have been included until now in HLT1.

7.3 Impact of the Downstream algorithm to detect long lived particles in the SM

As explained in Sec. 1.1, some particles in the SM have lifetimes of order 100 ps, such as the Λ and K^0_S hadrons, and a large amount of their decays occur after the VELO detector. These particles are reconstructed in many physics analyses to measure observables which can be sensitive to new physics models.

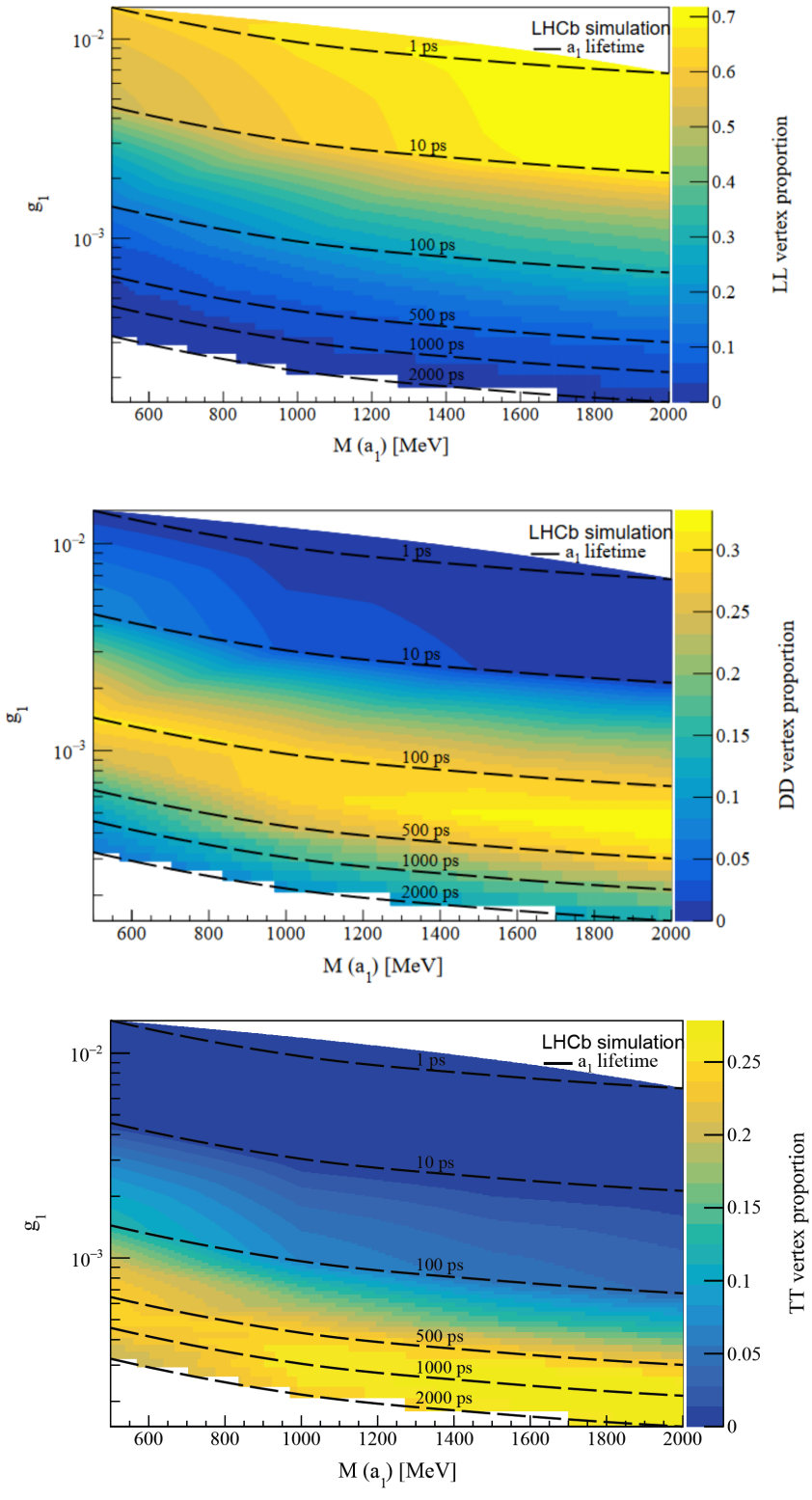


Figure 115: Reconstructibility of the decay vertex for the a_1 into two muons as function of its mas and coupling constant g_1. Corresponding lifetime curves are also drawn. Topologies are shown for *LL* (top), *DD* (medium) and *TT* (bottom) tracks.

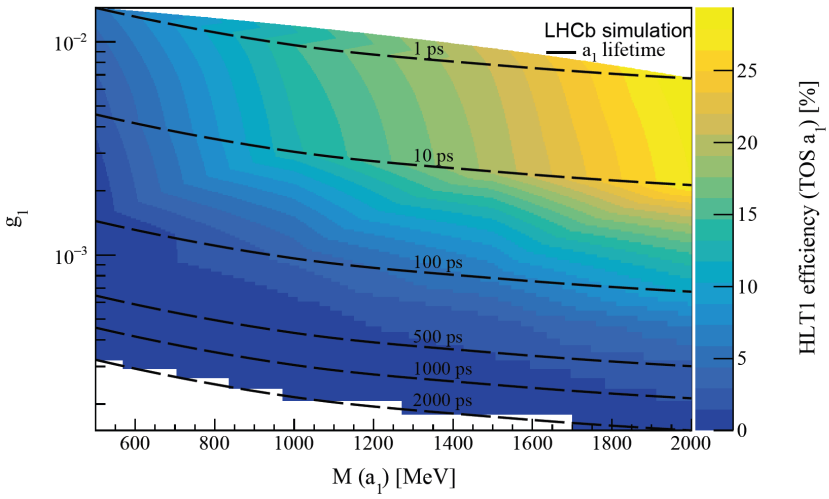


Figure 116: Proportion of events triggered by the HLT1 decision on the a_1 decay products (Trigger on Signal (TOS)).

7.3.1 Impact for the $\Lambda_b^0 \to \Lambda\gamma$ and $K_S^0 \to \mu^+\mu^-$ decay channels

The rare $\Lambda_b^0 \to \Lambda\gamma$ decay, where the Λ baryon decays into a proton and a pion, is one example of a decay to probe new physics. Measurements of the branching fraction and the angular distribution of the decay particles are sensitive to models with non-standard right-handed currents [**LHCb:2019wwi**, 48, 96]. These measurements are addressed in detail in Chapter 8. Another example is the very rare $K_S^0 \to \mu^+\mu^-$ decay, which is very suppressed in the SM, and has not been observed experimentally yet. This decay is very sensitive to different BSM scenarios such as SUSY [97] or the presence of leptoquarks [98]. Figure 117 shows the sensitivity to the $K_S^0 \to \mu^+\mu^-$ decay channel (in terms of branching fraction limit) as a function of the product of the trigger efficiency and luminosity.

Using 10000 simulated events, the HLT1 trigger effect (prior to the implementation of the algorithms developed in this thesis) has been studied on these two decay channels involving Λ and K_S^0 particles [94]. Figure 118 shows the normalised number of reconstructible events (LL+DD+TT) as function of the end decay vertex of the Λ and of the K_S^0.

The relative proportions of track types are 12% (LL), 51% (DD) and

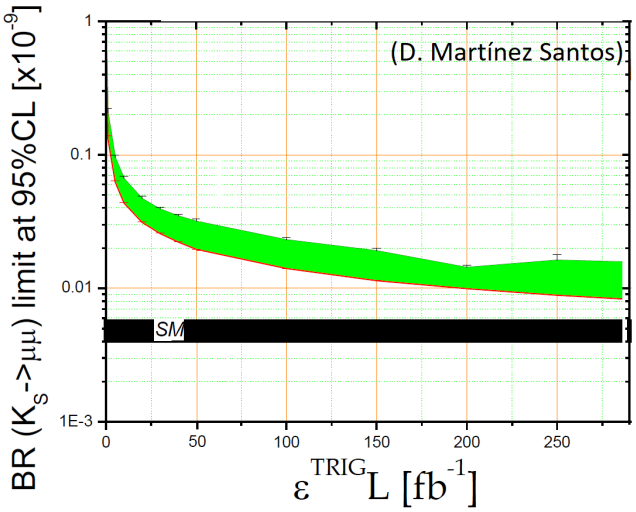


Figure 117: Sensitivity (green band) to the $K^0_S \to \mu^+\mu^-$ branching fraction as function of the trigger efficiency $\times$ luminosity. The SM prediction is shown in black [99].

37% (TT) for the $\Lambda^0_b \to \Lambda\gamma$ decay channel. For the case of prompt K^0_S the amount of reconstructible tracks is 46% (LL), 38% (DD) and 16% (TT) A large amount of decays occurs at the end of the VELO, and they do not let enough hits to be reconstructed as *long* tracks, being reconstructed thus in the DD category. That is, the amount of DD+TT tracks is about 88% for the Λ decay channel and 54% for prompt K^0_S.

One can apply the HLT1 conditions on the reconstructible events, and in particular to check how many events are selected by inclusive trigger lines such as the OneTrackMVA or TwoTrackMVA. They require tracks with minimum transverse momentum and minimum impact parameter significance with respect to the primary vertex, and for the latter, that they form a vertex with minimum requirements. In the case of $\Lambda^0_b \to \Lambda\gamma$, the HLT1 signal efficiency for the proton and pion coming from the Λ is found to be less than 10%. In the case of the K^0_S the HLT1 efficiency on the muons, adding some inclusive muon lines and dedicated lines for K^0_S selection is less than 25%. Note that we normalize to the sum of LL+DD+TT reconstructible events. The K^0_S candidates in this work are *prompt*, produced at the interaction point. In the case that K^0_S candidates are coming from the decays of b or c-hadrons the amount of reconstructible LL candidates are expected to decrease, increasing the

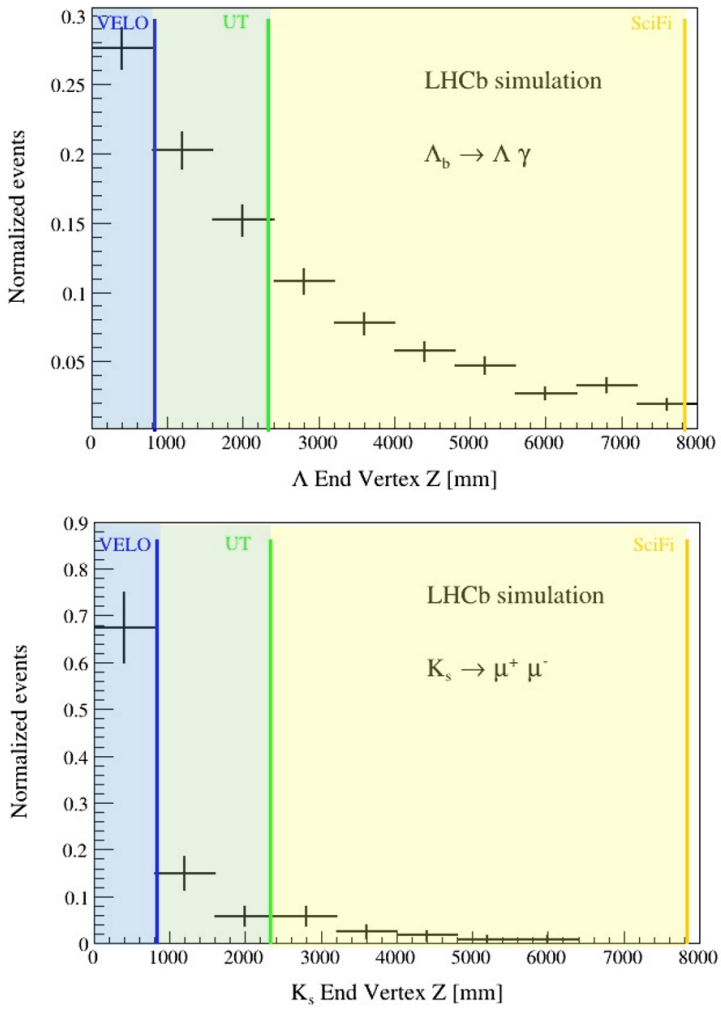


Figure 118: Λ (top) and K^0_S (bottom) reconstructible candidates as function of the decay vertex position. The Λ candidates decay into a pion and a proton. The K^0_S candidates decay into two muons of opposite charges. Vertical colour lines indicate the positions of the VELO, UT and SciFi detectors in the z-axis.

HLT1 inefficiency.

7.3.2 Impact on other exclusive decay channels

The Downstream algorithm is expected to improve many other physics analyses in the SM which involve Λ and K^0_S particles. Some of them are listed below:

- **B-decays**:
  - *Radiative decays of b-baryons*: in addition to the $\Lambda_b^0 \to \Lambda\gamma$ decay channel, other b-baryon decays will largely benefit of the inclusion of the Downstream algorithm in HLT1. The $\Xi_b^- \to \Xi^-\gamma$ decay channel has been searched for at LHCb using Run2 data and a limit has been set for its branching fraction [100]. Since the Ξ^- baryon has a large lifetime and decays into a Λ and a pion, the final decay products mostly originate after the VELO. Something similar happens for the $\Omega_b^- \to \Omega^-\gamma$ decay channel, where the Ω^- decays into a Λ^0 and a kaon. Many other excited states of the these b-baryons decay to Λ and kaons [101], and all of them will benefit of the work in this thesis. The measurement of observables in these decays such as the photon polarisation, CP asymmetries or branching fractions, are very sensitive probes to new physics scenarios.
  - *Charmless B decays:* Decays of B mesons to final states with even numbers of strange quarks or antiquarks, such as the $B^+ \to K_S^0 K_S^0 \pi^+$ or $B^+ \to K_S^0 K_S^0 K^+$ decay modes are suppressed in the Standard Model. Such decays proceed mainly via $b \to d$ or $b \to s$ loop transitions and are also very sensitive to new physics. Decays of B mesons via intermediate resonant states (f_0, f_2, etc...) can also lead to $K_S^0 K_S^0$ pairs in the final state. Measurements of branching fractions and time dependent asymmetries in these decays will largely improve since these decays involve two LLPs and just one pion or kaon as long track to fire the HLT1.

- **Charm decays:**
  - *Decays of charm hyperons*: decays of the type $\Xi_c^- \to \Xi^-$ + nπ, with n being 1, 2 or 3 pions can also benefit of the work of this thesis. The Ξ^- baryon is long-lived and decays into a Λ and a pion, meaning that at the end one has to deal with three tracks which can be reconstructed as LLL, DDL or DDD combinations, L referring to a long track and D to a *downstream* track.

Even if the Ξ companions pions are reconstructed as *long* tracks and can fire the HLT1 trigger, the small transverse momentum (typically less than 1 GeV/c) makes the efficiency be low. The reduction observed for the *DDL* and *DDD* combinations at the HLT1 level is 25% lower as compared to the *LLL* case, so we expect that the inclusion of the Downstream in HLT1 makes the decay particles from the Ξ and its Λ^0 daughter contributed in the trigger decision and increase the efficiency.

- $D^0 \to K^0_S K^0_S$*:* Measurements of CP violation in the charm sector offer a unique opportunity to search for new physics. In the SM, CP violation in charm decays is expected to be O(0.1%) or below, so any enhancement would indicate physics beyond the SM. The $D^0 \to K^0_S K^0_S$ is specially relevant because the expected size of the CPV effects is large [102], up to the percent level in the SM. The measurement of the CP asymmetry in this channel is sensitive to a different mix of amplitudes compared to other charm decay channels, and can help to elucidate the mechanisms of CPV in charm hadron decays. Other channels used to measure CP-violating asymmetries, such as $D^{\pm}_{(s)} \to K^0_S h^{\pm}$, where h can be a kaon or a pion, will also benefit from Downstream, allowing HLT1 not to be only fired by the long h.

- *Decays from charmonium states:* Decays of $c\bar{c}$ states into strange baryons, of the type of $J/\psi \to \Lambda\bar{\Lambda}$, have relatively large branching fractions, of the order of 10^{-3}. These decays are important for performing polarisation studies, and in particular they have been proposed, together with decays of charm hyperons, for measurements of electric and magnetic dipole moments at LHCb [103]. The possibility of selecting two downstream particles and perform vertexing with them will largely benefit measurements related to spin physics and in general to spectroscopy.

- **Strange decays**

  - *Rare decays of* K^0_S *mesons*: Even if the branching fractions of

these decays are below 10^{-9}, they offer a unique laboratory to probe the SM physics in the strange sector [99]. At LHCb the geometrical acceptance of K^0_S is 1% and the cross section is around 0.3 barns. With the inclusion of the Downstream algorithm, and considering that the momentum resolution is about 5%, the search for a vast amount of decays can be improved. In particular $K^0_S \to \mu^+\mu^-$, $K^0_S \to \mu^+\mu^-\mu^+\mu^-$, $K^0_S \to \pi^0\mu^+\mu^-$, $K^0_S \to \pi^0 e^+e^-$, $K^0_S \to \gamma\mu^+\mu^-$, and $K^0_S \to \pi^+\pi^-e^+e^-$ decay channels. Until now, selection criteria for the study of these channels have been restricting the search to decays inside the VELO detector. With the new algorithm the LHCb sesnitivity to these channels will improve considerably.

– *Semileptonic and rare decays of hyperons*: Strange hyperons, Λ, Σ, Ξ, Ω, are copiously produced at LHCb, directly from proton-proton collisions (prompt) or in the decays of beauty and charm hadrons. Transitions with $|\Delta S| = 2$ are practically forbidden in the SM, with branching fractions of order of 10^{-17}. New physics transitions could enhance these kind of decays. The Downstream algorithm will benefit all these decays allowing to increase the sensitivity over the current limits [99].

Table 11 summarizes the expected impact of the Downstream algorithm in different SM decays. Another important improvement concerns the reconstructibility of converted photons ($\gamma \to e^+e^-$). This is still under study but preliminary tests show a 50% gain in statistics when including *downstream* tracks in HLT1.

Channel	DD/LL proportion	Interest
b-hadron decays		
$\Lambda_b^0 \to \Lambda\gamma$	3.4	γ polarisation, BR
$\Xi_b^- \to \Xi^-\gamma$	25	γ polarisation, BR
$\Omega_b^- \to \Omega^-\gamma$	13	γ polarisation, BR
$B^+ \to K_S^0 K_S^0 \pi^+$	2.8	CPV, BR
$B^+ \to K_S^0 K_S^0 K^+$	2.7	CPV, BR
$B_s^0 \to K_S^0 K_S^0$	3.6	CPV, BR
Charm physics		
$\Lambda c^+ \to \Lambda K^+$	4.4	Polarisation studies
$\Xi_c^- \to \Xi^-\pi^-$	8.4	Polarisation studies
$\mathrm{D}^0 \to K_S^0 K_S^0$	1.8	CPV
$J/\psi \to \Lambda\bar{\Lambda}$	4.8	Polarisation studies, BR
Strange physics		
$K_S^0 \to \mu^+\mu^-$	0.6	BR
$K_S^0 \to \mu^+\mu^-\mu^+\mu^-$	0.8	BR
$K_S^0 \to \gamma\mu^+\mu^-$	0.8	BR

Table 11: Decay channels in the SM which can benefit of the Downstream reconstruction at HLT1 level. Second column represents the proportion of DD over LL tracks.

8 Study of $\Lambda_b^0 \to \Lambda\gamma$ decays

In this chapter the main procedure to reconstruct and select $\Lambda_b^0 \to \Lambda\gamma$ decays are explained, with the aim of measuring its branching fraction. In Sec. 8.1 the importance of the measurement is outlined, following the discussion from Chapter 1. The motivation to use the $B^0 \to K^{*0}\gamma$ decay channel as normalisation channel over other decay modes is explained, even if it is not a b-baryon decay and the K^* is not a long-lived particle. Reconstruction and selection procedures for the two channels are explained in Sec. 8.3. After the selection, there are still an important amount of background events under the signal peak. In Sec. 8.4 the sources of backgrounds expected in both decay channels are detailed, and the procedure to extract the signal outlined. Sec. 8.5 focuses on the expected improvement with the inclusion of the `Downstream` algorithm at HLT1. The analysis of the photon polarisation is also a key measurement and the prospects are outlined in Sec. 8.6.

8.1 Measurement of the $\Lambda_b^0 \to \Lambda\gamma$ decay channel

The $\Lambda_b^0 \to \Lambda\gamma$ decay channel involves a $b \to s$ flavour-changing neutral-current (FCNC) transition and it is forbidden at tree level in the SM. It has to proceed through *loop* diagrams, as shown in Fig. 119, and thus is very sensitive to new particles which can modify observables such as the decay branching fraction or the photon polarisation. Precise measurements of branching fractions and CP observables have been performed in the B-meson system at BaBar, Belle and LHCb experiments [33–37]. They are in agreement with the SM predictions. The $\Lambda_b^0 \to \Lambda\gamma$ decay

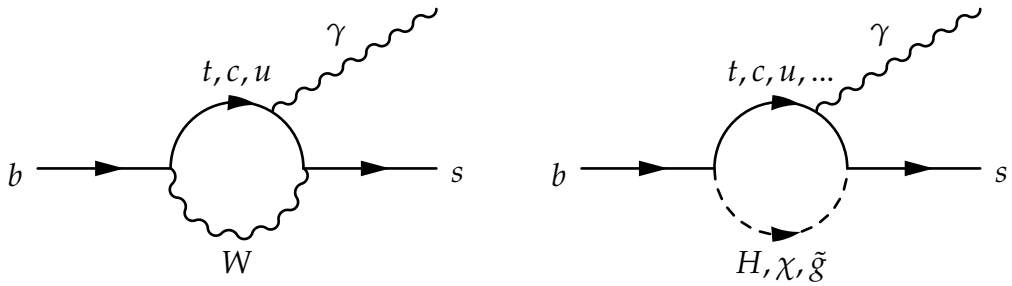


Figure 119: The $b \to s\gamma$ penguin diagram, mediated by SM particles (left) and BSM particles (right).

channel has the peculiarity that it is a b-baryon decay and it offers a much more rich spin structure, suitable to probe the helicity anatomy in $b \to s\gamma$ transitions. It has been observed for first time at LHCb [42, 43], and its branching fraction has been measured using a limited data sample of 1.7 fb^{-1}. The value is $\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma) = (7.1 \pm 1.5 \pm 0.6 \pm 0.7) \times 10^{-6}$, where the first uncertainty is statistical, the second systematic and the third is the systematic from external measurements, is compatible with theoretical predictions [44–46]. At present the precision is limited by the statistical uncertainty. In addition to the possibility of being enhanced or suppressed by new physics mechanisms, in the framework of the SM the measurement of the branching fraction is important to validate the calculations of heavy-to-light baryonic form factors, which enter in other transitions such as $\Lambda_b \to \Lambda\ell^-\ell^+$, which is measured to probe lepton flavour universality. A more realistic representation of the $\Lambda_b^0 \to \Lambda\gamma$ decay is depicted in Fig. 120. The precise measurement of the $\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)$ is important to constraint QCD models such as SU(3) Flavour Symmetry [104], Light Cone Sum Rules (LCSR) [105], Quark Models (QM) [106] or Heavy Quark Symmetry (HQS) [107].

The $\Lambda_b^0 \to \Lambda\gamma$ decay is experimentally challenging to reconstruct since the Λ_b^0 decay vertex cannot be determined directly due to the long lifetime of the Λ baryon and the unknown photon direction, when reconstructed as a cluster in the electromagnetic calorimeter. The basic ingredients for reconstructing $\Lambda_b^0 \to \Lambda\gamma$ events is explained in the following section, with the variables and selection criteria that can be used in order to obtain the maximum signal yield. The $\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)$ measurement can be performed using different decay channels for normalisation. Several

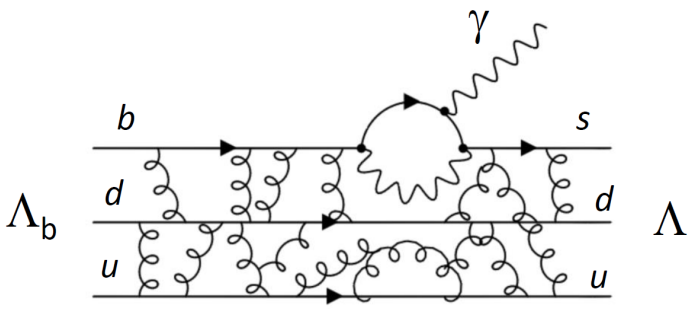


Figure 120: Representation of the $\Lambda_b^0 \to \Lambda\gamma$ decay including QCD effects.

possibilities have been explored in the context of this thesis, which have different advantages and disadvantages. The number of events of a specific decay channel X_a into generic X_b and X_c particles, can be obtained by using Eq 8.1:

$$N_{X_a \to X_b\, X_c} = 2\,\mathcal{L} \times \sigma_{pp\to b\bar{b}} \times f_{b\to X_a} \times \mathcal{B}(X_a \to X_b\, X_c) \times \mathcal{B}(X_{b,c}) \times \epsilon_{\text{sel}}^{X_a} \quad (8.1)$$

where $\mathcal{L}$ is the integrated luminosity, $\sigma_{pp\to b\bar{b}}$ is the proton-proton cross section into heavy-quark pairs, $f_{b,\bar{b}\to X_a}$ is the fragmentation fraction of a b-quark or anti b-quark hadronising to a specific X_a specie, $\mathcal{B}$ refers to the different branching fractions of the particle decays involved in the process, and ϵ_{sel} is the selection and reconstruction efficiency of the full decay channel in study. Using a normalisation channel, and the ratio of branching fractions as observable, implies that many of these quantities cancel, in particular the luminosity, cross section, or the fragmentation fractions, which are quantities measured with poor precision. Part of the selection and reconstruction efficiencies also cancels if the channels involved are similar and close selection criteria can be used, providing a reduction in the systematic uncertainties. In the following the different normalisation channels that have been considered in the context of this thesis are explained.

- $\Lambda_b^0 \to J/\psi\Lambda$ decay mode:

  This normalisation mode benefits from the fact that both signal and normalisation channels are Λ_b^0 decays with the same fragmentation fractions. Also both channels, the signal and the normalisation, include the decay of a Λ decaying into a proton and a pion, thus

the branching fraction, and partial efficiencies (PID and tracking for these particles) cancel in the ratio. However, the main disadvantage of this decay mode is that the branching ratio is measured together with the hadronisation probability $\mathcal{B}(\Lambda_b^0 \to J/\psi\Lambda) \times f(b \to B) = (5.8 \pm 0.8) \cdot 10^{-5}$ [101], with a relatively large uncertainty. Also, since the probability of hadronisation to a Λ_b^0 baryon is smaller as compared to a B meson, it might not provide a large number of events, limiting the statistical power of the measurement. The ratio of branching fractions can be written as follows

$$\frac{\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)}{\mathcal{B}(\Lambda_b^0 \to J/\psi\Lambda)} = \frac{N_{\Lambda_b^0 \to \Lambda\gamma}}{N_{\Lambda_b^0 \to J/\psi\Lambda}} \times \mathcal{B}(J/\psi \to \mu^+\mu^-) \times \frac{\epsilon_{\text{sel}}^{\Lambda_b^0 \to J/\psi\Lambda}}{\epsilon_{\text{sel}}^{\Lambda_b^0 \to \Lambda\gamma}}. \quad (8.2)$$

The branching ratio of the $\mathcal{B}(J/\psi \to \mu^+\mu^-) = (5.961 \pm 0.033)\%$ [101].

- $B^0 \to K^{*0}\gamma$ decay mode:

  This decay mode is advantageous due to the large number of events it provides, which improves the statistical significance of the results. This large number of events comes from the relatively high branching fraction of $B^0 \to K^{*0}\gamma$ decays, $\mathcal{B}(B^0 \to K^{*0}\gamma) = (4.18 \pm 0.25) \times 10^{-5}$. However, it has the disadvantage of involving different b-species (B^0 and Λ_b^0), and different final states (K^* and Λ), which means that many factors (e.g., fragmentation fractions, branching fractions, efficiencies) do not cancel in the ratio of the branching fractions. Using this normalisation channel the ratio is

$$\frac{\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)}{\mathcal{B}(B^0 \to K^{*0}\gamma)} = \frac{N_{\Lambda_b^0 \to \Lambda\gamma}}{N_{B^0 \to K^{*0}\gamma}} \times \frac{f_{B^0}}{f_{\Lambda_b^0}} \times \frac{\mathcal{B}(K^* \to K^+\pi^-)}{\mathcal{B}(\Lambda \to p\pi^-)} \times \frac{\epsilon_{\text{sel}}^{B^0 \to K^{*0}\gamma}}{\epsilon_{\text{sel}}^{\Lambda_b^0 \to \Lambda\gamma}}, \quad (8.3)$$

with $\mathcal{B}(K^* \to K^+\pi^-) = (66.503 \pm 0.014)\%$, and the $\mathcal{B}(\Lambda \to p\pi^-) = (64.1 \pm 0.5)\%$ [101]. Here the value of $\mathcal{B}(K^* \to K^+\pi^-)$ has been obtained multiplying the value in Ref. [101] of $(99.754 \pm 0.021)\%$ by 2/3, considering isospin rules. Using $\frac{f_{\Lambda_b}}{f_u + f_d} = 0.259 \pm 0.018$ and the assumption that $f_u = f_d = 0.340 \pm 0.021$ CDF [108], the value $\frac{f_{\Lambda_b}}{f_B} = 0.518 \pm 0.036$ is obtained.

- $B_s^0 \to \phi\gamma$ decay mode:

  This decay mode also involves different particles in the final state, and different fragmentation fractions. The ratio reads

$$\frac{\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)}{\mathcal{B}(B_s^0 \to \phi\gamma)} = \frac{N_{\Lambda_b^0 \to \Lambda\gamma}}{N_{B_s^0 \to \phi\gamma}} \times \frac{f_{B_s^0}}{f_{\Lambda_b^0}} \times \frac{\mathcal{B}(\phi \to K^+K^-)}{\mathcal{B}(\Lambda \to p\pi^-)} \times \frac{\epsilon_{\text{sel}}^{B_s^0 \to \phi\gamma}}{\epsilon_{\text{sel}}^{\Lambda_b^0 \to \Lambda\gamma}}, \quad (8.4)$$

  The branching ratio has been measured by LHCb with 1 fb^{-1} [38] and the value is $\mathcal{B}(B_s^0 \to \phi\gamma) = (3.4 \pm 0.4) \cdot 10^{-5}$. The branching ratio of the ϕ mesons into two kaons of opposite charge is $\mathcal{B}(\phi \to K^+K^-) = (49.1 \pm 0.5)\%$ [101]. It has a larger error as compared to the $B^0 \to K^{*0}\gamma$ decay channel, because it has less statistics and is largely affected by the uncertainties of the fragmentation function. Using the value measured by LHCb, $\frac{f_s}{f_d} = 0.2539 \pm 0.0079$ obtained at 13 TeV [109], and using the previous results for b-baryons, we get $\frac{f_{\Lambda_b}}{f_{B_s}} = 2.04 \pm 0.16$.

- $\Lambda_b^0 \to J/\psi p K^-$ decay mode:

  This mode involves the same Λ_b^0 baryon as the signal mode, which is an advantage because the fragmentation function, f_Λ, cancels in the ratio of the branching ratios. However, the final states are different (J/ψ and γ), which introduces additional uncertainties in terms of the trigger and reconstruction procedures. The expression for the ratio is

$$\frac{\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)}{\mathcal{B}(\Lambda_b^0 \to J/\psi p K^-)} = \frac{N_{\Lambda_b^0 \to \Lambda\gamma}}{N_{\Lambda_b^0 \to J/\psi p K^-}} \times \frac{\mathcal{B}(J/\psi \to \mu^+\mu^-)}{\mathcal{B}(\Lambda \to p\pi^-)} \times \frac{\epsilon_{\text{sel}}^{\Lambda_b^0 \to J/\psi p K^-}}{\epsilon_{\text{sel}}^{\Lambda_b^0 \to \Lambda\gamma}}. \quad (8.5)$$

  The main disadvantage is that the branching ratio is measured with large uncertainty: $\mathcal{B}(\Lambda_b^0 \to J/\psi p K^-) = (2.6^{+0.5}_{-0.4}) \times 10^{-5}$. As quoted before, $\mathcal{B}(J/\psi \to \mu^+\mu^-) = (5.961 \pm 0.033)\%$ and $\mathcal{B}(\Lambda \to p\pi^-) = (64.1 \pm 0.5)\%$ [101].

All these channels are studied in detail to determine the best choice for the normalisation mode. They are described below.

- **Statistical uncertainly:** during its Run2 the LHCb experiment has collected $6\,\text{fb}^{-1}$ at 13 TeV. In the context of this thesis, the author has participated in the analysis of the $\Lambda_b^0 \to \Lambda\gamma$ events, and two selections were studied: a tight selection with a signal over background (S/B) rate of 0.5, and a loose selection with $S/B = 0.3$. The number of signal events in the loose selection was found to be 440 ± 40, a factor 2.4 larger than for the tight selection. Since this measurement is expected to be statistically limited at present, the loose selection is chosen for the measurement of the branching ratio. In the following, an estimation of the number of events for the several normalisation channels is performed. For Run3 it is expected that an integrated luminosity of $23\,\text{fb}^{-1}$ will be acquired. Scaling by luminosity the number of $\Lambda_b^0 \to \Lambda\gamma$ decays, we would expect about 1700 events in Run3, using only *long* tracks for performing the analysis. If we take into account the new `Downstream` algorithm developed in this thesis, and considering the *long/downstream* tracks relation in Table 11, we are expecting around 7500 signal $\Lambda_b^0 \to \Lambda\gamma$ events.

  The number of events for different normalisation decay channels in Run2 has been obtained following the analysis results in Refs [42, 43, 48]. Table 12 shows the statistical uncertainty coming from the term $\frac{N_{sig}}{N_{norm}}$ for the different decay modes. The statistical uncertainty, including the background suppression procedure, is dominated by the number of events in the signal $\Lambda_b^0 \to \Lambda\gamma$ decay channel. The large improvement in Run3 is due to the inclusion of *downstream* tracks in the analysis, thanks to the new algorithm in HLT1.

- **Fragmentation functions:** the uncertainties introduced by the fragmentation fractions are quoted in Table 13.

- **External branching fractions:** the uncertainties introduced by the external branching fractions are quoted in Table 14. As it can be noticed in this table the uncertainty is entirely dominated by the values of the branching fractions of the b-hadron decay chosen as normalisation channel.

Channel	$\Lambda_b^0 \to \Lambda\gamma$	$\Lambda_b^0 \to J/\psi\Lambda$	$B^0 \to K^{*0}\gamma$	$B_s^0 \to \phi\gamma$	$\Lambda_b^0 \to J/\psi pK^-$
Run2 ($6\,\text{fb}^{-1}$)	440	11465	170549	30340	41000
Run3 ($23\,\text{fb}^{-1}$)	7500	43950	653771	116303	157167
$\kappa_{bkg.}$	1.9	1.5	1.7	1.6	1.5
Run2 σ_{stat}		9.2%	9.1%	9.1%	9.1%
Run3 σ_{stat}		2.3%	2.2%	2.2%	2.2%
Run2+3 σ_{stat}		2.2%	2.1%	2.2%	2.1%

Table 12: Number of events for the signal and normalisation channels in Run2 and expected for Run3. The $\kappa_{bkg.}$ factor is considered to take into account the effect of the background contribution in the statistical uncertainty of the number of events in each channel, being $\sqrt{N} \cdot \kappa_{bkg.}$[1]. The relative statistical uncertainty of the ratio of the number of signal to normalisation events, σ_{stat}, is computed for Run2 and Run3.

Norm. channel	$\Lambda_b^0 \to J/\psi\Lambda$	$B^0 \to K^{*0}\gamma$	$B_s^0 \to \phi\gamma$	$\Lambda_b^0 \to J/\psi pK^-$
$\frac{f_{\Lambda_b^0}}{f_{B_s^0,B^0}}$	-	0.518 ± 0.036	2.04 ± 0.16	-
$\sigma_{frag.}$	-	6.9%	7.8%	-

Table 13: Uncertainties introduced by the knowledge of the fragmentation fractions.

Norm. channel	$\Lambda_b^0 \to J/\psi\Lambda$	$B^0 \to K^{*0}\gamma$	$B_s^0 \to \phi\gamma$	$\Lambda_b^0 \to J/\psi pK^-$
$\sigma_{\mathcal{B}(Norm.\,channel)}$	14%	6%	12%	19%
$\sigma_{\mathcal{B}(\Lambda\to p\pi^-)}$	-	0.8%	0.8%	0.8%
$\sigma_{\mathcal{B}(J/\psi\to\mu^+\mu^-)}$	0.6%	-	-	0.6%
$\sigma_{\mathcal{B}(K^*\to K^+\pi^-)}$	-	0.02%	-	-
$\sigma_{\mathcal{B}(\phi\to K^+K^-)}$	-	-	1.0%	-
$\sigma_{ext.}$	14%	6%	12%	19%

Table 14: Uncertainties introduced by the knowledge of the branching fractions entering in the ratio.

- **Systematic uncertainties:** potential systematic uncertainties, in addition to the values obtained from external inputs, have also to be considered to choose the proper normalisation channel. They can be grouped into different sources:

  - *Signal and background modelling:* mass fits have to be performed to extract the signal yields, as it is explained in Sec. 8.4. The systematic uncertainty is expected to be dominated by the background modelling, so channels with larger background, as $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$, are expected to have the larger effects. Recent studies on $B^0 \to K^{*0}\gamma$ and $B_s^0 \to \phi\gamma$ decay events with Run2 present a good signal and background modelling and introduce less than 1% systematic uncertainty [110]. $\Lambda_b^0 \to J/\psi\Lambda$ and $\Lambda_b^0 \to J/\psi p K^-$ are very clean modes and the proportion of background is very small. The systematic uncertainty is then assumed to be dominated by the background subtraction procedure of the $\Lambda_b^0 \to \Lambda\gamma$ decay channel, and it is evaluated to be below 9%[43, 48]. This is anticipated to be the dominant systematic uncertainty of the analysis of the branching ratio, and it is expected to be reduced with more data to control the background.

  - *Efficiency calculation:* acceptance, trigger, reconstruction, and selection efficiencies, as explained in Sec. 8.3, are obtained using large simulated samples, apart from the particle identification efficiency (PID) which is obtained from large $\Lambda \to p\pi^-$ and $D^0 \to K^-\pi^+$ data control samples. The photon identification is usually one of the major sources of uncertainty. Several contribution have to be evaluated separately, but for any of the channels used as normalisation, they are expected not to be larger than 1 or 2%.

  - *Data-MC differences:* the variables used to select the signal and normalisation modes could be different in data as compared to simulation samples. Following Refs. [43, 48], possible differences can be controlled using some of the b-baryon modes ($\Lambda_b^0 \to J/\psi\Lambda$ or $\Lambda_b^0 \to J/\psi p K^-$), and the systematic uncertainty

associated to this source is expected to be below 4%.

As conclusion, and following the previous discussion, we have decided to use the $B^0 \to K^{*0}\gamma$ decay channel as normalisation decay mode since it provides the smallest uncertainty to measure the $\Lambda_b^0 \to \Lambda\gamma$ branching fraction. For Run3 the uncertainty of the ratio $\mathcal{R} = \frac{\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)}{\mathcal{B}(B^0 \to K^{*0}\gamma)}$ will be dominated by the knowledge of the fragmentation functions (6.9%), the $\mathcal{B}(B^0 \to K^{*0}\gamma)$ (6%), and the systematic uncertainties (expected below 9%).

8.2 Data samples

Data: the analysis may exploit the data recorded by LHCb during Run2 which corresponds to a total integrated luminosity of $\mathcal{L} = 5.80\,\text{fb}^{-1}$. Table 15 shows the integrated luminosity per year of data taking. The signal candidates are built by the stripping line `Lb2L0Gamma`, as explained below. As the running conditions, trigger, calorimeter resolution, etc. are different, the events are reconstructed using dedicated versions of the reconstruction software for each year of data taking.

Year	$\mathcal{L}$ (fb^{-1})
2015	0.33
2016	1.67
2017	1.67
2018	2.19

Table 15: Run2 data samples.

Simulation: the simulation samples for each year are listed in Table 16 for the $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$ decay channels.

8.3 Reconstruction and selection of signal and normalisation candidates

The reconstruction of a $\Lambda_b^0 \to \Lambda\gamma$ event candidate requires a Λ baryon reconstructed from two *long* or *downstream* tracks, compatible with a

Decay	Event type	Year	Sim version	Events	Stripping
$\Lambda_b^0 \to \Lambda\gamma$	15102307	2016	Sim09b	37M	v41r4p4
	15102320	2017	Sim09h-ReDecay01	0.84M	S29r2p1
	15102320	2018	Sim09h-ReDecay01	0.84M	S34r0p1
$B^0 \to K^{*0}\gamma$	11102204	2016	Sim09j	8M	S28r2
	11102204	2017	Sim09j	11.6M	S29r2p1
	11102204	2018	Sim09j	11.6M	S34r0p1

Table 16: Simulation samples available for the signal and normalisation channels, with event type, year, simulation version, number of simulated events and stripping version used to build the samples.

proton and a pion hypotheses, pointing to a common displaced vertex. The Λ is later combined with an energetic photon to produce the $\Lambda_b^0 \to \Lambda\gamma$ initial candidate. This event topology can be seen in Fig. 121. The selection strategy explored in this thesis is based on the one in Ref. [42]. The main variables used to select signal candidates and reject background events are explained in the following. They exploit the fact that particles produced in heavy hadron decays have a higher transverse momentum and are displaced from the PV.

- **Transverse momentum p_T**: the transverse momentum is the momentum of a particle in the plane perpendicular to the beam axis, and is defined as $p_T = \sqrt{p_x^2 + p_y^2}$.
- **Pseudorapidity η**: is the angle of a particle with respect to the beam axis and is defined as $\eta = -\ln[\tan(\theta/2)]$.

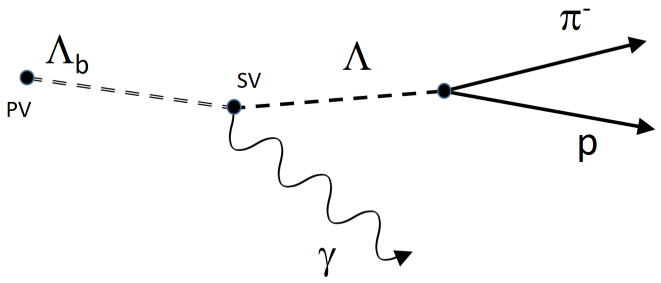


Figure 121: Decay topology of signal $\Lambda_b^0 \to \Lambda\gamma$ decays.

- **DOCA**: the Distance Of Closest Approach is defined as the smallest distance between two particle tracks. A small DOCA implies that the two tracks have been originated at the same vertex.

- **MT-DOCA**: the Mother DOCA is the DOCA of a particle with respect to its mother. A small MT-DOCA ensures the correct reconstruction of the mother particle.

- **Impact Parameter (IP)**: is the equivalent of the DOCA for a track and a vertex. It is defined as the smallest distance between a vertex and the extrapolation of the track, as pictured in Fig. 122. A small IP implies that the evaluated track has been originated in the vertex. The IP is commonly used to select those tracks that do not come from the PV and, thus, have a high IP with respect to the PV.

- ΔM: is the difference between the reconstructed mass and the expected nominal mass of a certain particle.

- χ^2: the quality of the track or vertex reconstruction from the track or vertex fit allows to select real tracks and vertices which position have been correctly computed. This quantity is usually normalised by the number of degrees of freedom (ndf), and expressed as χ^2/ndf.

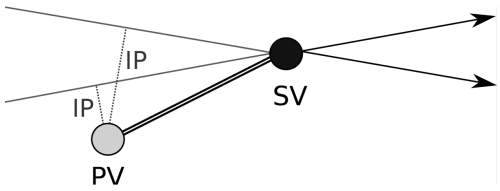


Figure 122: Definition of the IP with respect to the PV.

- **DIRA**: the DIRection Angle is the angle between the path defined by the position of the origin and decay vertices of the particle, and its momentum direction reconstructed from its decay products. The DIRA is nearly zero if the decay is properly reconstructed. The DIRA can also be defined using the PV instead of the origin vertex. A large DIRA_{PV} implies that the particle is not originated at the PV. The DIRA definition is illustrated in Fig. 123.

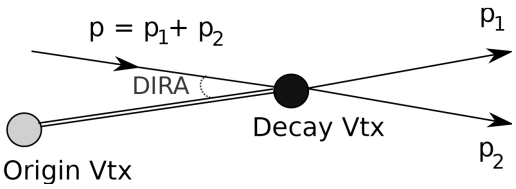


Figure 123: Definition of the DIRA angle.

- **Ghost probability**: it is the output of a multivariate classifier (MVA) that combines information from the track fit quality using different tracking systems. It grants rejection against ghost tracks, for which a significant fraction of hits associated to the track does not belong to it.

- **FD**: the Flight Distance is the length traveled by a decaying particle. It is computed as the distance between the production and the decay vertex. The FD is specially useful taking the assumption that the particle is originated at the PV. Hereby, a FD larger than expected implies that the hypothesis of PV as origin point for the particle is false and, thus, that the particle has been produced in a secondary vertex.

- $\chi^2_{\mathbf{FD}}$: the flight distance significance of a decaying particle. It express the level of certainty on the FD measured from the PV.

- $\mathcal{A}_p$: the momentum asymmetry is defined as the normalised difference between the momentum of a given particle (p) and the total momentum of the tracks in a cone around the particle (p_{Cone}):

$$\mathcal{A}_p = \frac{p - p_{\text{Cone}}}{p + p_{\text{Cone}}}. \tag{8.6}$$

  The transverse momentum asymmetry is defined with an equivalent expression, replacing p with p_T. They are often referred as *isolation* variables.

- γ **CL**: The Confidence Level of the photon helps to the discriminate photons from hadrons using PS, ECAL, HCAL and cluster-track matching information[2].

[2]In Run3 the PS has been removed and this information is not included.

- **PID**: The Particle IDentificacion requirements for a particle combine the information from the PID system to compute the likelihood of a certain mass hypothesis. Some of these variables are `ProbNNk`, `ProbNNpi` and `ProbNNp`, for kaon, pion and proton identification, respectively.

The same variables are used in the reconstruction of the $B^0 \to K^{*0}\gamma$ candidates, apart from the momentum asymmetry, which is a signal isolation variable that helps to reduce the background. The $B^0 \to K^{*0}\gamma$ decay topology is shown in Fig. 124. The main differences comes from the fact that the K^* decays promptly and the secondary vertex (SV) is easier to reconstruct from the pion and kaon information. In addition, the helicity angle, $\cos\theta_H$, defined as the angle between the direction of the kaon in K^* rest frame and the direction of the K^* in the B^0 rest frame, is used to suppress backgrounds.

8.3.1 Reconstruction and selection efficiencies

The ratio of the selection efficiencies in Eq. 8.7 can be obtained by using individual efficiencies, which are defined as a product of acceptance, reco+stripping, offline, photon PID, track PID and trigger efficiencies

$$\frac{\epsilon_{\text{sel}}^{\Lambda_b^0 \to \Lambda\gamma}}{\epsilon_{\text{sel}}^{B^0 \to K^{*0}\gamma}}, \tag{8.7}$$

$$\epsilon_{\text{sel}} = \epsilon_{\text{acc}} \times \epsilon_{\text{reco+stripping}} \times \epsilon_{\text{offline}} \times \epsilon_{\gamma\text{PID}} \times \epsilon_{tr\text{PID}} \times \epsilon_{\text{trigger}} \tag{8.8}$$

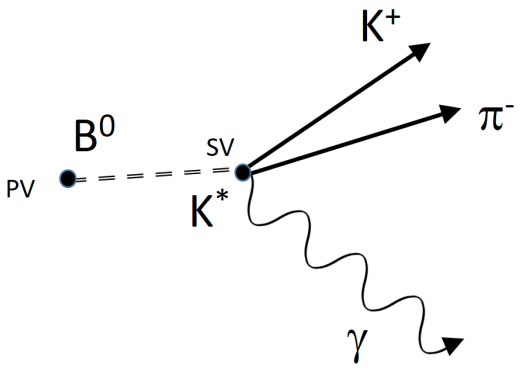


Figure 124: Decay topology of normalisation $B^0 \to K^{*0}\gamma$ decays.

Geometrical acceptance ϵ_{acc}

Since the detector simulation is computationally expensive, and the event generation is comparatively fast, we usually apply some cuts at generator level to remove events where the decay products are far outside of LHCb acceptance. The efficiencies of those cuts are studied using `MCStatTools`, as it is shown in Table 17.

	Year	Magnet Up (%)	Magnet Down (%)
$\Lambda_b^0 \to \Lambda\gamma$	2016	50.19 ± 0.14	50.04 ± 0.14
	2017	24.2 ± 0.053	33.69 ± 0.025
	2018	33.24 ± 0.025	33.70 ± 0.025
	2024	33.045 ± 0.035	33.731 ± 0.034
$B^0 \to K^{*0}\gamma$	2016	13.241 ± 0.033	13.204 ± 0.033
	2017	13.131 ± 0.034	13.125 ± 0.034
	2018	13.187 ± 0.033	13.145 ± 0.033
	2024	25.223 ± 0.010	25.139 ± 0.013

Table 17: Acceptance efficiency evaluated at generation phase for $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$, for the two different magnet polarities.

Trigger selection and efficiency $\epsilon_{\text{trigger}}$

The trigger strategy exploits the presence of a high-energetic photon and a high-p_{T} proton in the final state. At the hardware level[3], the specific requirement `L0Electron TOS` or `L0Photon TOS` is applied, while at HLT1, `Hlt1TrackMVA TOS` should be satisfied. Due to the highly asymmetric decay of the Λ baryon, the proton carries most of the momentum and the two-track line `Hlt1TwoTrackMVA` does not help to select these decays, so it is not used. The specific criteria are listed in Table 19[4]. Since both, the $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$ decay channels are triggered by the same lines, the corresponding efficiencies cancel in the ratio.

A dedicated HLT2 trigger, `Hlt2RadiativeLb2L0GammaLL`, is used to reconstruct and select signal candidates. The specific requirements included in the HLT2 selection are detailed in Table 20. In the offline

[3]Only during the Run2.

[4]The Velo Qcut variable means the number of VELO layers with no hits.

selection, candidates are required to be selected as TOS by this trigger line. For the control mode $B^0 \to K^{*0}\gamma$ the L0 and HLT1 follow the same trigger strategy, while for HLT2 trigger, a exclusive HLT2 line `Hlt2RadiativeBd2KstGamma_TOS` is used. The trigger strategies are summarised in the Table. 18.

Table 18: Run 2 trigger strategies.

	$\Lambda_b^0 \to \Lambda\gamma$	$B^0 \to K^{*0}\gamma$
L0	L0Photon_TOS or L0Electron_TOS	
HLT1	Hlt1TrackMVA_TOS	
HLT2	Hlt2RadiativeLb2L0GammaLL_TOS	Hlt2RadiativeBd2KstGamma_TOS

Table 19: HLT1 trigger selection for Run2. A logical OR of the standard and high-E_T paths is applied

Variable	Hlt1TrackAllL0	Hlt1TrackPhoton
p_T (MeV/c)	> 1300	> 1200
p (MeV/c)		> 6000
χ^2/ν		< 2
χ^2_{IP}		> 13
Number of T-Hits	> 16	-
Number of Velo Hits	> 9	-
Velo Qcut	< 3	-
Trigger	L0All	L0Photon or L0Electron

Reconstruction and stripping $\epsilon_{\text{reco+strip}}$

The stripping refers to a process of preselecting specific events of interest. This process is performed to reduce the amount of data and to enhance the signals of the interesting physics processes. The stripped data is then used for further analysis and reconstruction of particles. The efficiency of reconstruction and stripping can be evaluated using

Variable	Units	Requirement
Track p	MeV/c	> 2000
Track p_{T}	MeV/c	> 250
Track χ^2_{IP}		> 36
Track χ^2		< 3
p DLLp		> 0
Tracks DOCA	mm	< 0.2
γ p	MeV/c	> 5000
γ p_{T}	MeV/c	> 2000
Λ p_{T}	MeV/c	> 1500
Λ IP	mm	> 0.1
Λ χ^2_{Vtx}/ndof		< 15
Λ χ^2_{FD}		> 0
Λ τ	ps	> 2
Λ ΔM	MeV/c^2	< 20
γ p_{T} + Λ p_{T}	MeV/c	> 5000
Λ_b^0 χ^2_{MTDOCA}		< 9
Λ_b^0 p_{T}	MeV/c	> 1000
Λ_b^0 ΔM	MeV/c^2	< 1000

Table 20: Requirements included in the `Hlt2RadiativeLb2L0GammaLL` selection.

Monte Carlo samples. The uncertainties are estimated by the binomial uncertainty calculation, $\sigma_\epsilon = \sqrt{\epsilon(1-\epsilon)/N}$.

The stripping versions used for Run 2 are listed in Table 21. For the signal mode $\Lambda_b^0 \to \Lambda\gamma$, the dedicated stripping line is `StrippingLb2L0Gamma` and for the normalisation channel, the `Beauty2XGammaExclTDCPVBd2KstGamma` from Ref [110] can be used. Nevertheless, this line has to be aligned with the signal selection criteria. The selection criteria for signal $\Lambda_b^0 \to \Lambda\gamma$ decays in Run2 is summarised in Table 22.

Table 21: Run 2 stripping lines

	$\Lambda_b^0 \to \Lambda\gamma$	$B^0 \to K^{*0}\gamma$
`Stripping line`	`StrippingLb2L0Gamma`	`Beauty2XGammaExclTDCPVBd2KstGamma`
`Stripping v2016`	`S28r2`	`s28r2`
`Stripping v2017`	`S29r2p1`	`S34r0p1`
`Stripping v2018`	`S29r2p1`	`S34r0p1`

Offline efficiency $\epsilon_{\text{offline}}$

The selection consists of two parts. First, a preselection with particle identification (PID) selection criteria and some fiducial requirements. Second, a multivariate classifier based on a Gradient Boosted Decision Trees (GBDT) is chosen, using as input variables the most discriminant to distinguish between signal and combinatorial background.

Preselection

After the stripping, a loose preselection is applied to reduce the size of the data samples while keeping as much signal as possible. It includes tighter cuts in the quantities previously used in the stripping and HLT2 selections. Loose PID cuts are also imposed on the proton and the pion. The selection criteria, taken from Ref. [111]. is detailed in Table 23.

Multivariate classifier

After the preselection, a Boosted Decision Tree (BDT) [112] can be used to further discriminate the signal from the background. In particu-

heightVariable	`Lb2L0Gamma`	Units
Track χ^2_{IP}	> 16	
max(p, π) Track χ^2/ndof	< 3	
min(p, π) Track χ^2/ndof	< 2	
Track Ghost Prob.	< 0.4	
π p_T	> 300	MeV/c
π p	> 2000	MeV/c
p p_T	> 800	MeV/c
p p	> 7000	MeV/c
p DLLp	> -5	
Tracks χ^2_{DOCA}	< 30	
Λ p_T	> 1000	MeV/c
Λ ΔM	< 20	MeV/c^2
Λ IP	> 0.05	mm
Λ χ^2_{Vtx}/ndof	< 9	
γ $p_T + \Lambda$ p_T	> 5000	MeV/c
Λ_b^0 p_T	> 1000	MeV/c
Λ_b^0 ΔM	< 1100	MeV/c^2
Λ_b^0 χ^2_{MTDOCA}	< 7	
γ CL	> 0.2	
γ p_T	> 2500	MeV/c

Table 22: Stripping selection for the $\Lambda_b^0 \to \Lambda\gamma$ signal channel.

lar, the XGBoost algorithm [113] implemented through the scikit-learn package [114] is used in Ref. [48]. The algorithm uses the simulation samples as proxy for signal and the high mass side band (Λ_b^0 Mass $>$ 6100 MeV/c^2) of the data samples as proxy for background.

The variables used in the training model for the hadronic part of the decay include transverse momenta, impact parameters and other geometric and kinematic variables such as the DOCA of the proton and the pion and the flight distance of the Λ. For the photon, the transverse momentum and the pseudorapidity are used. Isolation variables are also used for both the photon and Λ. The variables entering in the BDT are listed in Table 24.

Variable	Units	Requirement
Max Track Ghost Prob		< 0.2
Track p	GeV/c	$\in (3, 100)$
$\pi^{\pm}$ first hit Z	mm	< 270
p ProbNNp		> 0.2
π ProbNNpi		> 0.2
γ p_{T}	MeV/c	> 3000
Λ IP	mm	> 0.15
Λ χ^2_{IP}		> 16
Λ χ^2_{FD}		> 225
Λ M	MeV/c^2	$\in (1110, 1122)$
Λ_b^0 MTDOCA	mm	< 0.05
Λ_b^0 χ^2_{MTDOCA}		< 5
Λ_b^0 p_{T}	MeV/c	> 4000
Λ_b^0 ΔM	MeV/c^2	< 1000

Table 23: Preselection requirements applied on $\Lambda_b^0 \to \Lambda\gamma$ candidates.

The performance of the BDT is studied with testing samples and small effects from overtraining are observed [48]. The output is also measured in bins of Λ_b^0 and no bias is found, assessing a good behaviour of the BDT. A BDT cut of about 0.96 is choosen[5] which retains more than 50% of the signal and rejects 99% of the combinatorial background. In Fig. 128 (right) the mass distribution of the $\Lambda_b^0 \to \Lambda\gamma$ candidates after applying the BDT selection (and including PID criteria described below) is shown.

Similar variables are used in Ref. [110] to train a Gradient Boosted Decision Tree (GBDT) and select $B^0 \to K^{*0}\gamma$ candidates. Nevertheless, several differences can be found. In particular no isolation variables are used for the $B^0 \to K^{*0}\gamma$ selection, and the DIRA variable is utilised instead of the MTDOCA, providing equivalent information. The helicity angle of the decay is included in this case, and its importance in the case of the $\Lambda_b^0 \to \Lambda\gamma$ has to be evaluated.

[5] In practice different BDTs are trained for different run periods.

Variables
p $p_T + \pi^\pm$ $p_T + \gamma$ p_T
$\pi^\pm$ p_T
$\pi^\pm$ IP
p IP χ^2
Tracks DOCA
γ p_T
γ η
Λ p_T
Λ IP
Λ IP χ^2
Λ FD
Λ_b^0 p_T
Λ_b^0 MTDOCA
Λ Cone(1.0) $\mathcal{A}_p$
Λ Cone(1.0) $\mathcal{A}_{p_T}$
γ Cone(1.0) $\mathcal{A}_{p_T}$

Table 24: Input variables to train the BDT model for the $\Lambda_b^0 \to \Lambda\gamma$ decay.

PID efficiencies $\epsilon_{\gamma\text{PID}}$ and $\epsilon_{tr\text{PID}}$

The PID algorithms associate the reconstructed hadronic tracks to protons, pions and kaons. The PID variables are constructed using a neural network approach that combines the information of every subsystem into a single probability for each particle hypothesis [115]. For $\Lambda_b^0 \to \Lambda\gamma$ candidates, the requirements `ProbNNp>0.2` for protons and `ProbNNpi>0.2` for pions are chosen. This preselection removes most of the physical background that comes from the misidentification of the final state hadrons.

Similar requirements are applied to $B^0 \to K^{*0}\gamma$ candidates with loose PID criteria for the pion and the kaon: `ProbNNk`>0.2 and `ProbNNpi`>0.2. For pions, `ProbNNk`<0.2 is imposed to reject physic backgrounds.

High energetic photons coming from decays of π^0s ($\pi^0 \to \gamma\gamma$) merge in the same cell of the calorimeter and are reconstructed as a single cluster. To separate between π^0 and photons, a multivariate clasifier based on the cascade shape and the energy of the cluster is used. The tool is called `IsPhoton` [116]. The criterion for both $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$ candidates imposes this variable exceeds 0.6 for Run2 data.

8.4 Background subtraction

Once the candidates are selected, one needs to remove the background coming from different sources to extract the number of signal events. Background events faking the signal are contributing to bias measurements and to larger systematic uncertainties. The signal can be isolated from the remaining background by a fit to the mass distribution. To characterize the mass distribution of the signal and of each background component simulated samples can be used. Then, a simultaneous fit of the $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$ is the preferred strategy to perform the analysis and determine the signal yield and branching ratio.

8.4.1 Signal events

The invariant mass distribution of both, the $\Lambda_b^0 \to \Lambda\gamma$ and the $B^0 \to K^{*0}\gamma$ modes, can be modeled by a double-tail Crystal Ball [91]:

$$\mathrm{CB}(m;\mu,\sigma,\alpha_{R,L},n_{R,L}) = N\cdot\begin{cases} A_L\left(B_L - \frac{m-\mu}{\sigma}\right)^{-n_L}, & \text{for } \frac{m-\mu}{\sigma} \leq -|\alpha_L| \\ \exp\left(-\frac{(m-\mu)^2}{2\sigma^2}\right), & \text{for } -|\alpha_L| < \frac{m-\mu}{\sigma} < |\alpha_R| \\ A_R\left(B_R + \frac{m-\mu}{\sigma}\right)^{-n_R}, & \text{for } \frac{m-\mu}{\sigma} \geq |\alpha_R| \end{cases} \tag{8.9}$$

where N stands for the normalisation, and $A_{L(R)}$ and $B_{L(R)}$ are defined as:

$$\begin{aligned} A_i &= \left(\frac{n_i}{|\alpha_i|}\right)^{n_i}\exp\left(-\frac{\alpha_i^2}{2}\right), \\ B_i &= \frac{n_i}{|\alpha_i|} - |\alpha_i|. \end{aligned} \tag{8.10}$$

The shape of the b-hadron reconstructed mass is leading by the photon resolution, and this function allows to characterize different calorimeter effects. The left side tail, determined by n_L and α_L parameters, account for energy losses due to the finite volume of the detector, while pile-up effects are taken into account by the right side tail (n_R and α_R). For Run3 *pileup* effects are expected to increase. The main function of the distribution is a Gaussian core determined by two parameters, μ and σ. Two σ parameters can be also considered, for taking into account events with poorer resolution. Fig. 125 shows the fits to simulated events for the signal $\Lambda_b^0 \to \Lambda\gamma$ and normalisation $B^0 \to K^{*0}\gamma$ events for Run2. The mass resolution is about $95\,\mathrm{MeV/c^2}$ and no bias is observed. Tail parameters are found to be very similar for both decay channels [48, 110].

8.4.2 Background events

Combinatorial background

The most important background contribution for both the signal and normalisation channels is the combinatorial background. It originates from random combinations of a photon with two tracks, which largely comes from real Λ or K^*. This background can be described by an exponential distribution for the signal $\Lambda_b^0 \to \Lambda\gamma$ channel:

$$\mathrm{Exp}(m;\tau) = \frac{1}{\tau}\cdot\exp(m\tau), \tag{8.11}$$

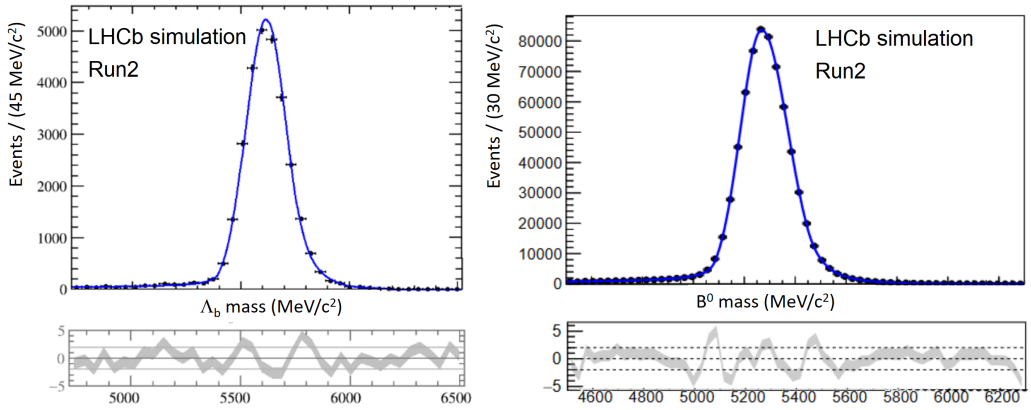


Figure 125: Fit to simulated (left) $\Lambda_b^0 \to \Lambda\gamma$ ([48]) and (right) $B^0 \to K^{*0}\gamma$ ([110]).

and by a first-order polynomial for the normalisation channel

$$P_{\text{comb}}(m; p_0) = 1 + m\, p_0. \tag{8.12}$$

where yields and the parameters p_0 and τ can be obtained directly from data.

Partially reconstructed background

This kind of events correspond to real b-hadron decays, reconstructed as a signal candidate, for which one or more final state particles have been missed. They present at least two tracks in the final state and a neutral particle. It could also happened that some of the reconstructed particles are misidentificated. Due to the missing particles, the mass distribution peaks in a lower mass region as compared to the signal candidate, but it can even extend to the signal peak, being counted as a signal event. The knowledge of these background contributions is expected to be the larger source of systematic uncertainties in the measurement of the ratio of branching fractions. The partially reconstructed background is usually modeled by an Argus distribution convoluted with a Gaussian distribution accounting for resolution effects. The Argus distribution is defined as:

$$\text{Arg}(m; m_0, c, p) = \mathcal{C} \cdot m \cdot \left[1 - \left(\frac{m}{m_0}\right)^2\right]^p \cdot \exp\left\{c \cdot \left(1 - \left(\frac{m}{m_0}\right)^2\right)\right\} \tag{8.13}$$

where $\mathcal{C}$ is a normalisation term that depends on all three parameter, namely m_0, c, p. The parameters can be obtained by fitting simulated b-hadron decay channels that could be sneaked in the signal or normalisation mass regions. The expected contributions are

- For the signal $\Lambda_b^0 \to \Lambda\gamma$ decay channel:
  $\Lambda_b^0 \to \Lambda\eta$ candidates where $\eta \to \gamma\gamma$ and one of the photons is missing or reconstructed as only one photon in the calorimeter.

- For the normalisation $B^0 \to K^{*0}\gamma$ decay channel:
  Due to the large width of the K^{*0} several several contributions are expected. They can be grouped in several types:

  - Decays of the type $B^0 \to K^{*0}\eta(\to \gamma\gamma)$ where a γ is missing.
  - Decays of the type $B \to (K\pi\pi)\gamma$ with or without an intermediate resonance ($K_1(1270), K_1(1400), K_2^*(1430)$ *etc.*), where one of the pions is not reconstructed.
  - Decays of the type $B \to D\rho$, where two or three pions (neutral or charged) are missing, and a π^0 is reconstructed as a photon. The main contributions are expected to come from the decays $B^+ \to D^0(K\pi^0)\rho^+(\pi\pi^0)$ and $B^+ \to D^0(K\pi\pi^0)\rho^+(\pi^+\pi^0)$.

These sources of backgrounds have been studied in detail in Refs. [48] and [110] using dedicated simulated samples. They account for less than 1% and 5% for the signal and normalisation channels, respectively. A similar procedure will be used for the analyses of Run3 data, and several simulated samples are being prepared for its generation.

Peaking background

This indistinguishable source of background could arise from the misidentification of one or more particles in the final state. Peaking backgrounds are decays that are fully reconstructed and survive the event selection criteria. These decays have typically two charged tracks and a high-energy neutral particle (π^0 or photon) in the final state. They usually can be represented by single or double-tail Crystal Ball functions, similar to the signal shapes.

- For the signal channel this background can be suppressed by the PID requirements, and specially by a tight cut around the Λ candidate mass. Misidentification of a neutral pion as a photon is also a typical source of peaking background for radiative B decays. However, the $\Lambda_b^0 \to \Lambda\pi^0$ decay is colour suppressed in the SM [117] and its contamination is expected to be negligible (see Ref. [42]).

- For the normalisation channel those with significant relative contaminations are $\Lambda_b \to \Lambda^*(pK)\gamma$ (with the proton reconstructed as a pion), $B^0 \to K^{*0}(K\pi)\pi^0$ (with the π^0 reconstructed as a photon), $B^0 \to \rho^0(\pi\pi)\gamma$ (where a pion is reconstructed as a kaon), and $B_s^0 \to \phi(KK)\gamma$ (where a kaon is reconstructed as a pion). Those backgrounds are studied in simulation and its contribution can be fixed when performing the fit to the mass distribution.

8.5 Branching fraction: improvement using *downstream* tracks

To determine the number of signal and normalisation events in Eq. 8.3, an extended unbinned maximum-likelihood fit has to be performed to the $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$ data samples. For this an extended probability density function (PDF) is used to fit the data:

$$PDF(m) = N_{sig} \cdot \left[PDF_{\text{signal}} + C_i \cdot PDF_{\text{peak.}}\right] + \sum_{\text{bkg}} N_{\text{bkg}} PDF_{\text{bkg}} \qquad (8.14)$$

where N_{sig} stands for the number of the signal events and N_{bkg} corresponds to each contribution of background (combinatorial and partially reconstructed), respectively, described by the corresponding functions above. C_i accounts for the coefficients of the peaking background relative to the signal.

The free parameters of the fit are:

- μ and σ: The mean and width of the signal Gaussian core.

- p_0: The slope of the combinatorial background.

- Signal and combinatorial background yields, N_{sig} and N_{bkg}. The yields of the partially reconstructed backgrounds are also let free to vary.

The fixed parameters of the fit are the peaking background contributions, C_i. They are determined based on the relative contributions studied in Ref. [110]. The mass range for the fit is choosen to be $[4600, 6000]$ MeV$/c^2$ for both the signal and normalisation channels. Since in radiative b-hadron decays the mass resolution is dominated by the photon momentum resolution, nor resolution degradation neither an increase of the partially reconstructed background is expected when using *downstream* tracks in Run3. The proper calibration of the CALO is crucial when using Run3 data, which is specially challenging due to the high *pileup* conditions. Preliminary studies show a mass resolution of $\sigma \approx 110\,\text{MeV}/c^2$ at ν=7.6, corresponding to Run3 conditions.

Figure 126 shows a toy simulation of the $B^0 \to K^{*0}\gamma$ mass distribution including the sources of combinatorial, partially reconstructed and peaking backgrounds, using the shapes described above. The generation has been performed scaling the expected yields as in data [110]. Run2 + Run3 data is considered. The fit to the signal and background components is superimposed.

As it was already argued, for Run2 the expected statistical uncertainty is considerable because only *long* tracks can be considered for the analysis. The mass distribution using Run2 data can be shown in Fig. 128 (left). However, with the inclusion of all full Run2 + Run3 data and the ability to trigger *downstream* tracks at HLT1, the measurement will improve significantly, as it can be seen in Fig. 129 (left).

To complete the measurement several actions will be undertaken:

- An alignment of the selection criteria for the signal and normalisation channels. This includes for instance removing the helicity cut for the $B^0 \to K^{*0}\gamma$ decay channel, and to understand the effect of the isolation variables. Differences of the effect of introducing the MTDOCA or the DIRA variables need to be studied.

- The detailed efficiency computation for all the selection steps after the alignment procedure.

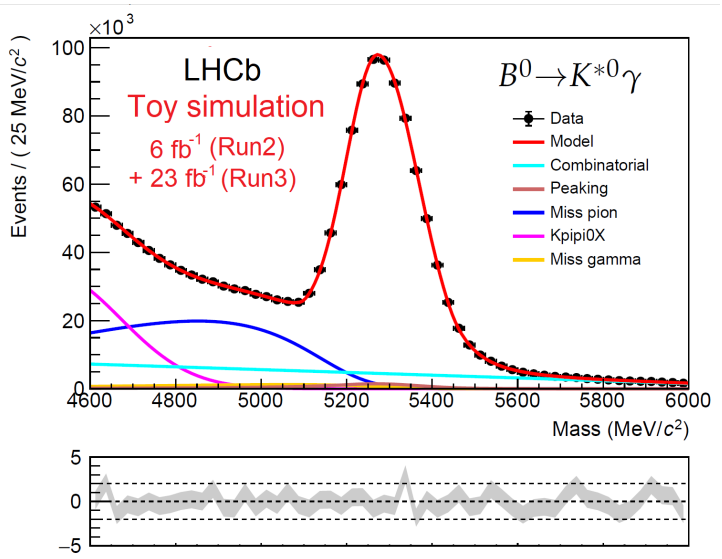


Figure 126: A toy simulation of the $B^0 \to K^{*0}\gamma$ mass distribution including several sources of combinatorial, partially reconstructed and peaking backgrounds. The statistics corresponding to Run2 + Run3 has been generated.

- The adoption of a simultaneous mass fit strategy to the signal and normalisation modes. This will allow to extract the $\Lambda_b^0 \to \Lambda\gamma$ and $B^0 \to K^{*0}\gamma$ number of events from a single fit, ensuring the correlations between both values are properly accounted for. Some of the parameters of the fit, such as the signal peak position and width, can be shared between the signal and normalisation modes.

- A detailed evaluation of the systematic uncertainties, mainly the ones concerning to the trigger and PID sources.

8.6 Photon polarisation: improvement using *downstream* tracks

The photon polarisation asymmetry, α_γ is the difference between the observed number of left-handed (L) and right-handed (R) polarised photons, normalised to the total number of detected photons:

$$\alpha_\gamma = \frac{\gamma_L - \gamma_R}{\gamma_L + \gamma_R}. \tag{8.15}$$

In the SM this quantity is predicted to be $\alpha_\gamma^{SM} = 1$ with corrections of the order $(m_s/m_b)^2 = \mathcal{O}(10^{-4})$.

The angular distribution of the $\Lambda_b^0 \to \Lambda\gamma$ decays can be expressed by

$$W(\theta_\Lambda, \theta_p) \propto 1 - \alpha_\Lambda P_{\Lambda_b} \cos\theta_p \cos\theta_\Lambda - \alpha_\gamma(\alpha_\Lambda \cos\theta_p - P_{\Lambda_b} \cos\theta_\Lambda), \quad (8.16)$$

where P_{Λ_b} is the initial Λ_b^0 polarisation and α_Λ is the Λ weak decay parameter. The angles involved in the decay are defined in Fig. 127. The angle θ_Λ is the angle between the Λ momentum in the Λ_b^0 rest frame and a vector $\hat{n}$ normal to the plane defined by the beam axis and the Λ_b^0 momentum in the laboratory frame (also drawn in the Λ_b^0 rest frame for convenience), and θ_p is the angle between the proton momentum in the Λ rest frame and the Λ momentum in the Λ_b^0 rest frame.

The dependence on the initial b-baryon polarisation can be eliminated by integrating out $\cos\theta_\Lambda$ in Eq. 8.16, obtaining a simpler expression

$$W(\theta_p) \propto 1 - \alpha_\gamma \alpha_\Lambda \cos\theta_p. \quad (8.17)$$

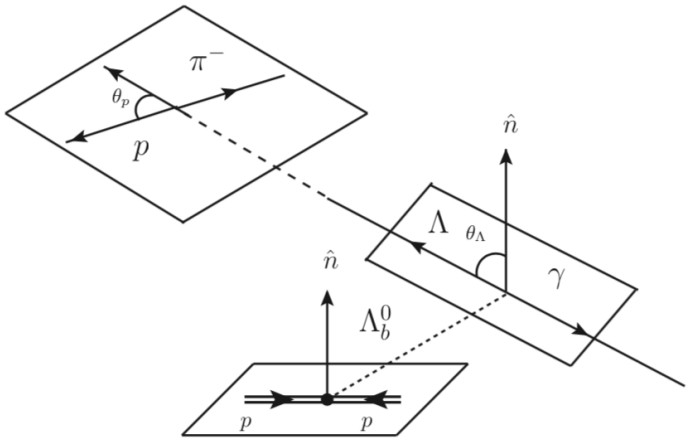


Figure 127: Schematic view of the $\Lambda_b^0 \to \Lambda\gamma$ decay (Ref [96]).

By measuring the angular distribution of $\Lambda_b^0 \to \Lambda\gamma$ candidates in $\cos\theta_p$, which is the helicity angle of the proton, and using the precisely determined Λ weak decay parameter, $\alpha_\Lambda = 0.754 \pm 0.004$ [118], one can access the photon polarisation in $\Lambda_b^0 \to \Lambda\gamma$ decays.

Using the reconstruction, selection and background characterisation procedures described above, the photon polarisation parameter has been determined using Run2 data [48]. An extended unbinned maximum likelihood to the mass distribution is performed, as shown in Fig. 128

(left). In the signal region, from 5387 to 5852 MeV/c^2, the signal yield is 440 $\pm$ 40, 1460 $\pm$ 23 combinatorial events and 10 $\pm$ 4 $\Lambda_b^0 \to \Lambda\eta$ events. The photon polarisation is then extracted with an unbinned maximum likelihood to the $\cos\theta$ distribution in Eq. 8.17 multiplied by an acceptance function, which accounts for the effect of the detector geometry, reconstruction and selection requirements [43, 48]. This acceptance function, parameterised as a fourth order polynomial, is obtained from simulated events after validating its behaviour using $\Lambda_b^0 \to J/\psi\Lambda$ decays[6]. The background is also accounted in the fit, and it is also described by a fourth-order polynomial, which accounts for both the combinatorial and partially reconstructed $\Lambda_b^0 \to \Lambda\eta$ decays. The result of the fit gives $\alpha_\gamma = 0.82 \pm 0.23 \pm 0.13$ and it is shown in Fig. 128 (right). The result is compatible with the Standard Model prediction but the precision is limited by the small statistical sample available, even if a loose selection has been used. Using Run2 + Run3 data, and including the

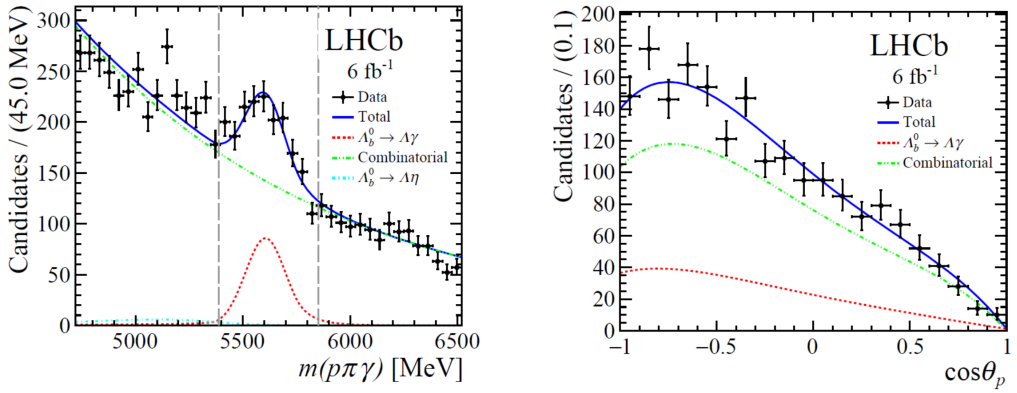


Figure 128: Results of the mass fit (left) and photon polarisation measurement (right) using 5.8 fb^{-1} of Run2 data [48].

downstream tracks in the analysis, the uncertainty of the α_γ parameter will be drastically reduced, as it is shown in Fig. 129. In Fig. 130 (left) a toy simulation example of a mass fit and photon polarisation measurement with 7500 $\Lambda_b^0 \to \Lambda\gamma$ signal events corresponding to Run3 is presented. The SM prediction has been used as input, and the value $\alpha_\gamma = 0.998 \pm 0.076$ is fitted in this particular example, where acceptance

[6]The J/ψ here is treated as a fake photon, the two-muon vertex no contributing in the reconstruction.

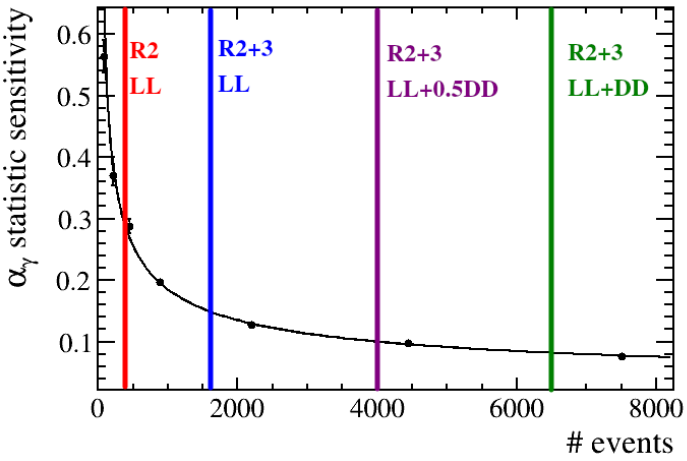


Figure 129: Expected sensitivity to the photon polarisation parameter, α_γ, using Run2 and Run3 data and including downstream tracks coming from Λ decays. The last two fringes corresponds to the present tracking efficiency of downstream tracks of 70%, and ideal 100%.

and resolution effects have been included, and the background has been scaled according to the luminosity. The θ angle resolution obtained for

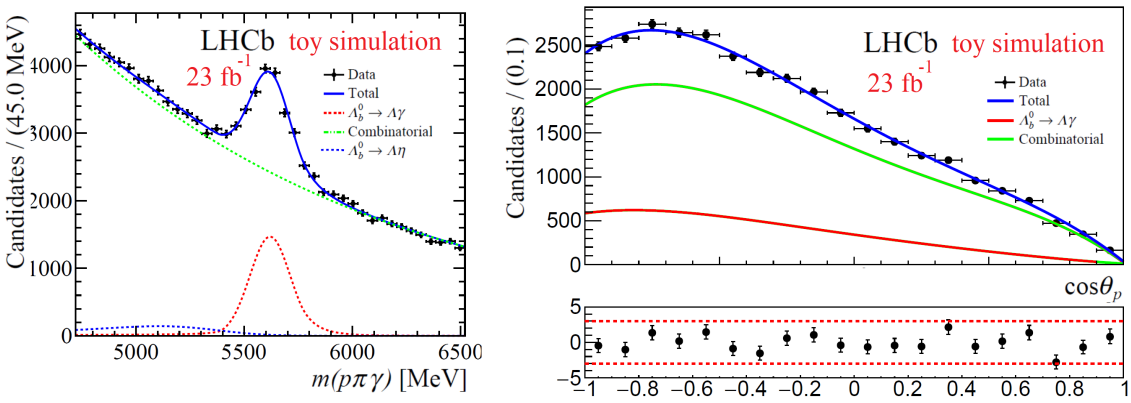


Figure 130: Example of a toy mass fit (left) and corresponding photon polarisation measurement (right) assuming 23 fb^{-1} of Run3 data and including *downstream* tracks coming from Λ decays.

downstream tracks is $\sigma_{\theta_{DD}} = 36 \pm 3\,\text{mrad}$, as compared to *long* tracks, which is $\sigma_{\theta_{LL}} = 22 \pm 5\,\text{mrad}$. Since the $\cos\theta$ distribution is quite smooth, the sensitivity to the photon polarisation parameter is not affected by this small difference.

A large amount of signal events is even more important for the study

of the CP asymmetry introduced in Sec. 1.4, since about half of the data correspond to each flavour, Λ_b^0 and $\overline{\Lambda_b^0}$. In the SM, CP asymmetries in $\Lambda_b^0 \to \Lambda\gamma$ decays are estimated to be less than O(1%) [119], but some BSM models could produce asymmetries as large as O(10%). Measurements of angular asymmetries and CP-conjugated decay branching ratios are thus of high interest. The inclusion of the `Downstream` algorithm in HLT1 will be decisive to come up with meaningful results.

9 Summary and conclusions

The work presented in this thesis constitutes a significant contribution to the first high level trigger (HLT1) of the LHCb experiment, based on the Allen project. In Allen, the entire HLT1 sequence of reconstruction algorithms have been designed to be executed on GPU cards. The work in this thesis has contributed to propel the project forward, enabling the LHCb trigger during the Run3, to successfully select real-time events at a frequency of 30 MHz. An extensive effort has been performed during the Allen development program, leading to the creation of Allen performance portability layer which enables framework to be executed in several architectures. Furthermore, inside this framework several key algorithms have been developed.

One of these algorithms, termed `HybridSeeding`, efficiently reconstructs the tracks produced in the SciFi detector (*T*-tracks). Another algorithm, named `VELO-SciFi Matching`, building upon the former, allows the reconstruction of *long* tracks with a momentum precision better than 1%. Additionally, a new algorithm named `Downstream` has been conceived, developed and incorporated into HLT1 for first time. A fast and efficient search of hits in the UT detector is performed, and a fast neural network (NN) is applied to reject ghost tracks. It allows to reconstruct *downstream* tracks with an efficiency of 70% and a ghost rate below 20%. This is the first time that a NN is developed for GPUs inside Allen. This new algorithm will allow the selection of long-lived particles at HLT1 level, opening up an unprecedented realm within both the Standard Model and its extensions. Of particular note is its implication in expanding the search scope for exotic long-lived particles, spanning

from 100 ps to several nanoseconds, a domain unexplored until now by the LHCb experiment. This, in turn, enhances the sensitivity to new particles predicted by theories that include a dark sector, heavy neutral leptons, supersymmetry, or axion-like particles.

In addition, the LHCb's ability to detect particles from the Standard Model, such as Λ and K^0_S, is greatly augmented, thereby enhancing the precision of analyses involving b and c hadron decays.

The integration of the HLT1 selection lines derived from the `Downstream` algorithm into the LHCb's real-time monitoring infrastructure will be important for the data taking during Run3 and beyond, and notably for the present alignment and calibration of the UT detector.

The precision in measuring observables which are sensitive to physics beyond the Standard Model, such as the rare $\Lambda^0_b \to \Lambda\gamma$ decay channel, will be greatly augmented. In this thesis a study of the measurement of the branching fraction of the $\Lambda^0_b \to \Lambda\gamma$ decay relative to the $B^0 \to K^{*0}\gamma$ channel has been performed. The analysis procedure, including selection, reconstruction and background rejection, has been described. A evaluation of the main systematic uncertainties affecting the measurement has been included. It has been concluded that the statistical precision for Run3 will be below 2% as a result of the inclusion of *downstream* tracks. The measurement of the photon polarisation in these transitions will also benefit from the increase in the yield, reaching a 10% precision in the α_γ parameter. Measurements of the CP asymmetry in $\Lambda^0_b \to \Lambda\gamma$ decays will also reach higher precision.

Resumen

9.1 Introducción

Las oportunidades de física que ofrece la próxima generación de experimentos basados en el Gran Colisionador de Hadrones (LHC) implica diferentes desafíos. La gran cantidad de colisiones protón-protón producidas debido a la alta luminosidad significa tener que lidiar con un mayor apilamiento (*pileup*) y una alta tasa de datos. Para hacer frente a estos retos el experimento LHCb ha desarrollado sofisticados sistemas de *trigger* que funcionan en dos etapas, y que seleccionan eventos de interés para hacer análisis de física. Para el período de operación actual del experimento LHCb, el Run3, y para períodos posteriores, la primera etapa del trigger llamada High Level Trigger 1 (HLT1), se ha implementado en tarjetas de procesamiento gráficas (GPUs) capaces de reducir la tasa de colisión visible de 30 MHz a 1 MHz. Estos sistemas de selección están diseñados para identificar eventos que pueden ser de interés científico y descartar eventos que no son relevantes.

Un tema de gran interés en la física de partículas es el estudio de las partículas de larga vida media (LLP), dentro del Modelo Estándar (SM), así como más allá de este modelo (BSM). Muchos modos de desintegración interesantes involucran partículas extrañas con vidas medias largas, tales como son los K^0_S o Λ. Muchos modelos teóricos nuevos también predicen LLP exóticas. La selección y reconstrucción de las LLP producidas es un desafío experimental. Estas partículas pueden desintegrarse lejos del vértice de interacción primario y son difíciles de

seleccionar mediante los sistemas de trigger de los experimentos actuales, y difíciles de aislar de los fondos del SM.

Esta tesis se enmarca dentro de la nueva infraestructura de computación en tiempo real y el sistema de trigger de LHCb. El principal objetivo ha sido el desarrollo de algoritmos clave de reconstrucción y de selección de sucesos, utilizando arquitecturas altamente paralelas para el HLT1, en particular GPUs. En la tesis se detalla como funciona el trigger y en particular este primer nivel, y como se ha llegado a la elección de utilizar GPUs frente a las convecionales CPUs. Se explica en detalle los algoritmos que se ocupan de la reconstrucción de trazas, haciendo énfasis en los nuevos algoritmos `HybridSeeding`, `Matching` y `Downstream`. Este último ha consituido uno de los trabajos principales dentro de la tesis y se detalla como se ha diseñado y cuáles son sus carácteristicas. Se discute también cómo estos algoritmos serán fundamentales en los estudios y búsquedas de LLP dentro del SM y más allá de él, para encontrar nueva física. También se presenta un proyecto y plan de implementación y verificación (*comissioning*) para la validación del algoritmo `Downstream` que utiliza uno de los detectores de trazas más relevantes, el UT, que estará instalado y totalmente operativo a partir de noviembre de 2023. Por último se discute la relevancia de este algoritmo y las perspectivas en un análisis de física concreto, el estudio del canal $\Lambda_b^0 \to \Lambda\gamma$ para medir dos observables clave: la fracción de desintegración y la polarización del fotón, predicha levógira en el Modelo Estándar.

9.1.1 El Modelo Estándar de las partículas elementales

El Modelo Estándar de la física de partículas (SM) es un marco teórico que describe las partículas y las fuerzas fundamentales que constituyen nuestro Universo. Es un modelo de gran éxito que ha sido rigurosamente probado y verificado por numerosos resultados experimentales. El SM describe tres de las cuatro fuerzas fundamentales de la naturaleza: la fuerza electromagnética, la fuerza débil y la fuerza fuerte. No incluye la gravedad, que está descrita por la teoría de la relatividad general. Según el Modelo Estándar la materia está formada por partículas llamadas fermiones, con espín 1/2, que se clasifican en quarks y leptones.

Los quarks son los componentes básicos de los protones y neutrones, que forman los núcleos atómicos, mientras que los leptones incluyen partículas como electrones y neutrinos.

El SM también describe las interacciones a través del intercambio de otras partículas llamadas bosones, con espín entero. La fuerza electromagnética está mediada por el fotón, mientras que la fuerza débil está mediada por los bosones $W^{\pm}$ y Z. Estas dos fuerzas se unen y son conocida como la fuerza Eletrodébil (EW). La fuerza fuerte, que mantiene unidos a los quarks dentro de protones y neutrones, está mediada por partículas llamados gluones. La teoría que explica estas interacciones se llama Cromodinámica Cuántica (QCD). Además de los fermiones y bosones, el Modelo Estándar predice la existencia del bosón de Higgs, que es responsable de otorgar masa a las partículas. Este fue uno de los descubrimientos más importantes en 2012 del Gran Colisionador de Hadrones LHC.

A pesar de sus éxitos, el Modelo Estándar no explica todos los fenómenos del Universo, como son la materia oscura o la energía oscura. Tampoco es consistente con la teoría de la relatividad general que describe la gravedad, ya que no se ha observado el *gravitón*, la partícula mediadora esperada. La asimetría materia-antimateria en nuestro Universo sigue siendo un misterio que no puede ser resuelto por el SM. Por lo tanto, los físicos todavía están buscando una teoría más completa que pueda unificar todas las fuerzas de la naturaleza y proporcionar una comprensión más íntegra del Universo. Muchas de las teorías propuestas involucran lo que se llama "partículas de vida media larga".

Partículas de vida media larga en el Modelo Estándar y más allá de él

La vida media, τ, de una partícula es una medida de cuanto tiempo una partícula inestable existe antes de desintegrarse en otras partículas. Es inversamente proporcional a su anchura de desintegración, Γ, que representa la probabilidad por unidad de tiempo de desintegrarse en estados finales específicos. La anchura de desintegración depende de factores como las masas de las partículas involucradas en el proceso, el espacio de fase, o la magnitud de la fuerza involucrada. En el Modelo

Estándar la gran mayoría de partículas son inestables y se desintegran en partículas más ligeras en un tiempo de vida muy corto. No obstante, el rango de vidas medias es muy grande, y abarca desde 10^{-25} s hasta 10^{35} años. En la Fig. 131 se pueden observar las vidas medias de las partículas del Modelo Estándar medidas en los detectores. En

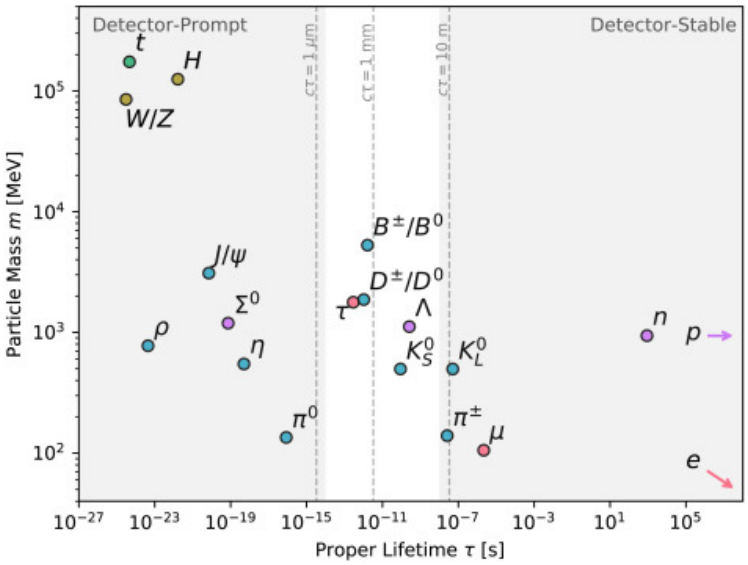


Figure 131: Vidas medias de las particlas en el SM e función de su masa. Las áreas sombreadas indican la región donde las partículas son estables o se desintegran muy rápidamente, desde el punto de vista de la reconstrucción en los detectores (figura obtenida en la Ref. [6]).

esta tesis consideramos las partículas de vida media larga aquellas que se encuentran en el rango desde aproximadamente 100 ps hasta varios nanosegundos.

Partículas de vida media larga más allá del Modelo Estándar

Muchos modelos de nueva física predicen partículas de vida media larga. Su presencia está motivada por el hecho de que se espera que interaccionen muy débilmente y que sean díficiles de detectar experimentalmente. Estos modelos son muy relevantes ya que podrían ayudar a entender la materia oscura y la bariogénesis en nuestro Universo. Estos modelos se pueden englobar en varios tipos:

- Partículas en el sector oculto (DB).
- Leptones neutros pesados (HNL).

- Supersimetría (SUSY).
- Partículas del tipo Axión (ALPs).

Para corroborar estos modelos se han realizado búsquedas extensivas en LHC, sin éxito, normalmente utilizando como signatura partículas con vértices desplazados. Algunas de las señales analizadas han sido eventos con un leptón proveniente de un vértice desplazado de alta multiplicidad, eventos con dos vértices desplazados de alta multiplicidad, desintegraciones de mesones B (mediados por un neutrino de Majorana) a un estado final con dos leptones del mismo signo asociados a vértices diferentes, desintegraciones del mesón B a un estado final con dos leptones de signo opuesto que forman un vértice desplazado, trazas de desintegraciones rápidas (*prompt*) (partículas estables masivas cargadas) que se extienden hasta las estaciones de muones con velocidad por debajo del umbral para producir luz Cherenkov en el detector RICH, y muones rápidos que forman un vértice desplazado de alta calidad. El principal problema es que para el primer nivel de trigger del experimento LHCb estos desplazamientos de vértices han estado limitados a un metro como máximo, al estar determinados por el detector de vértices más interno, el VELO.

Aparte de los detectores de propósito general como ATLAS y CMS que también contribuyen a este tipo de búsquedas, se han diseñado otros experimentos para búsquedas de partículas de vida media larga, como son MoEDAL-MAPP [19], MATHUSLA [20], CODEX-b [21], FASER [22] y SHIP [23]. Un repertorio sobre este tema puede encontrarse en la Ref. [24].

Partículas de vida media larga en el Modelo Estándar

En el contexto de esta tesis, las partículas que se consideran de vida media larga son principalmente K^0_S y Λ, con vidas medias de alrededor de 100 ps. Estas partículas son muy relevantes porque permiten estudiar una simetría fundamental de la naturaleza, la simetría CP, que abarca el concepto de Paridad (P) y Conjugación de carga (C), e implica que las partículas y anti-partículas se ven afectadas igualmente por las leyes de la naturaleza. En el Modelo Estándar la interacción electrodébil viola esta

simetría, y su estudio puede ayudar a entender la asimetría de materia-antimateria en el Universo. En particular los canales de desintegración que involucran hadrons pesados, tales como $B^0 \to K^0_S K^0_S$, $B^0 \to K^0_S \pi^+ \pi^-$, $D^0 \to K^0_S K^0_S$, $D^0 \to K^0_S \pi^+ \pi^-$, etc... son un laboratorio especial para estudiar esta simetría. La violación de CP en el Modelo Estándar está contemplada por una fase en la matriz compleja de Cabibbo-Kobayashi-Maskawa (CKM) que proporciona las probabilidades de transición entre quarks de diferente sabor y diferente carga. Las desintegraciones raras de hadrones pesados son también una herramienta muy poderosa para investigar si existe nueva física más allá del SM. La matriz CKM describe transiciones entre quarks de diferente carga, pero las transiciones de quarks sin cambio de carga (*Flavor Neutral Changing Currents*, FNCC), del tipo $b \to s$ o $b \to d$, no existen a nível árbol en el SM, y tienen que proceder via mecanismos más complejos que son muy sensibles a la existencia de nuevas partículas masivas. Estos mecanismos (*loops*) hacen que este tipo de desintegrciones estén muy supridas, y sean un reto experimental. Las desintegraciones radiativas de hadrones con un quark b, y en particular la desintegración $\Lambda^0_b \to \Lambda\gamma$ estudiada en el contexto de esta tesis, son muy interesantes para la búsqueda de nueva física, en observables relacionados con la polarización del fotón, y con la tasa de desintegración en estos decaimientos. En el SM la transición $b \to s\gamma$ procede como se muestra en la Fig. 132 y la anchura de desintegración se expresa en términos del coeficiente de Wilson C_7, relacionado con el operador pingüino electromagnético, como muestra la Eq. 9.1.

$$\Gamma(b \to s\gamma) = \frac{G_F^2\, \alpha_{EM}\, m_b^5}{32\, \pi^4} |V_{ts}^* V_{tb}|^2\, |C_7|^2 + \text{correcciones.} \tag{9.1}$$

Aquí G_F es la constante de Fermi, α_{EM} es la constante electromagnética, m_b es la masa del quark b, y V_{ij} son los correspondientes elementos de la matriz CKM. En el Modelo Estándar la transición $b \to s\gamma$ solo puede proceder con el fóton con polarización levógira. Una polarización distinta del fotón daría lugar a un nuevo coeficiente de Wilson, C_7', e indicaría la presencia de nueva física.

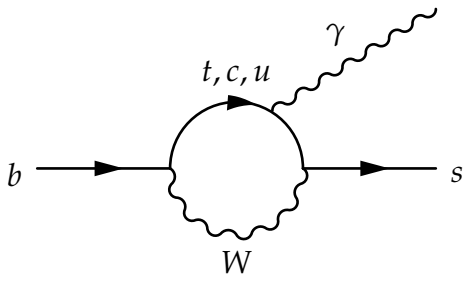


Figure 132: Diagrama de Feynman de la transición $b \to s\gamma$.

9.1.2 El experimento LHCb del colisionador LHC

El detector LHCb es un espectrómetro frontal y uno de los principales detectores del acelerador del Gran Colisionador de Hadrones (LHC) en el CERN (Ginebra, Suiza), cuyo objetivo principal es la búsqueda de nueva física a través de estudios de violación de CP y desintegraciones de hadrones pesados (que contienen quarks c o b). El experimento LHCb ha estado funcionando desde el año 2011 en los períodos Run1 y Run2, a energías de 7 TeV y 8 TeV en el primero, y 13 TeV en el segundo, con muy alto rendimiento, proporcionando una plétora de resultados físicos de precisión y descubriendo nuevas partículas.

Actualmente el LHCb se ha mejorado para el Run3, que comenzó en el 2022. En comparación con el detector anterior una de las mejoras más importantes está reacionada con el nuevo sistema de reconstrucción de trazas. El LHCb está compuesto por un sistema de tres subdetectores para la reconstrucción de trazas de partículas cargadas (VErtex LOcator (VELO), Upstream Tracker (UT) y SciFi tracker). El sistema de identificación de partículas está formado por dos detectores Ring Imaging Cherenkov (RICH), un calorímetros hadrónico (HCAL) y otro electromagnético (ECAL) y cámaras de muones.

El VELO está formado por sensores de píxeles de silicio y es fundamental para determinar los vértices de desintegración de los hadrones pesados. El UT es el siguiente sistema detrás del VELO, y antes del imán de 4 T, y está formado por capas de silicio segmentadas verticalmente. También se usa para determinar el momento de las partículas cargadas y poder definir sus trayectorias. Un tercer sistema de trazas, novedoso, es el detector de trazas de fibras de centelleo SciFi. Se encuentra detrás

del imán y está constituido por fibras centelleadoras de 2 m de longitud. Este detector es uno de los más relevantes en este trabajo de tesis ya que participa en todos los algoritmos desarrollados. Los subdetectores de radiación Cherenkov RICH1 y RICH2 están situados antes y después del imán, respectivamente. Proporcionan información de la velocidad de las partículas cargadas, que junto al momento obtenido por los sistemas de trazas, permiten su identificación. El gas radiador del RICH1 es tal que permite la identificación de partículas de bajo momento, mientras que el RICH2 está focalizado en partículas de alto momento. Los calorE-CAL y HCAL se encuentran a continuación y son los responsables de la reconstrucción de fotones y electrones, en el primer caso, y de hadrones (piones, protones, neutrones y kaones) en el segundo. Por último, el sistema más lejano de la colisión protón-protón consiste en cuatro cámaras de muones, M2-M5, que permiten la recontrucción precisa de muones y ayudan a mejorar la resolución en momento obtenida por el sistema de trazas. En la Fig. 133 se puede observar un esquema del detector LHCb con sus correspondientes subdetectores.

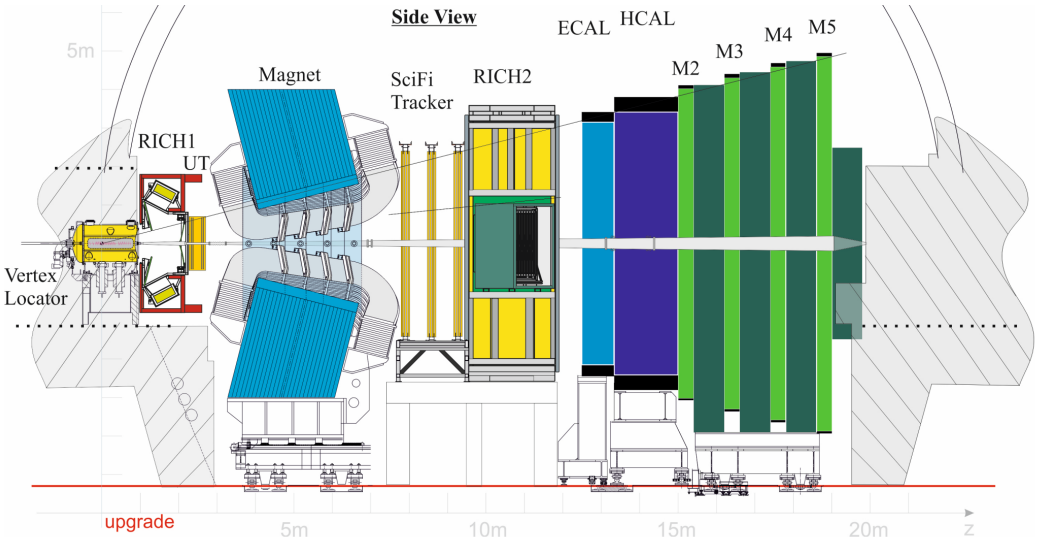


Figure 133: El detector LHCb.

9.1.3 El primer nivel de trigger de LHCb y el proyecto Allen

El experimento LHCb está utilizando desde 2022 un sistema de trigger completamente pionero, totalmente basado en software para recopilar los datos de las colisiones protón-protón a una frecuencia de eventos de 30 MHz.

Durante la primera etapa del Trigger (HLT1), el proyecto Allen realiza una reconstrucción parcial de las trazas de las partículas en tiempo real, utilizando nuevas técnicas y una paralelización eficiente con tarjetas de procesamiento gráfico GPUs. En este nivel del trigger se reduce la tasa de eventos alrededor de un factor 30. Reconstruir trazas a 30 MHz representa un desafío que debe enfrentarse con algoritmos muy eficaces, y es el trabajo fundamental de esta tesis. En la Fig. 134 se muetra el funcionamiento del procesamiento de datos durante el Run3 de LHCb, incluyendo la parte correspondiente al primer nivel de trigger HLT1. En

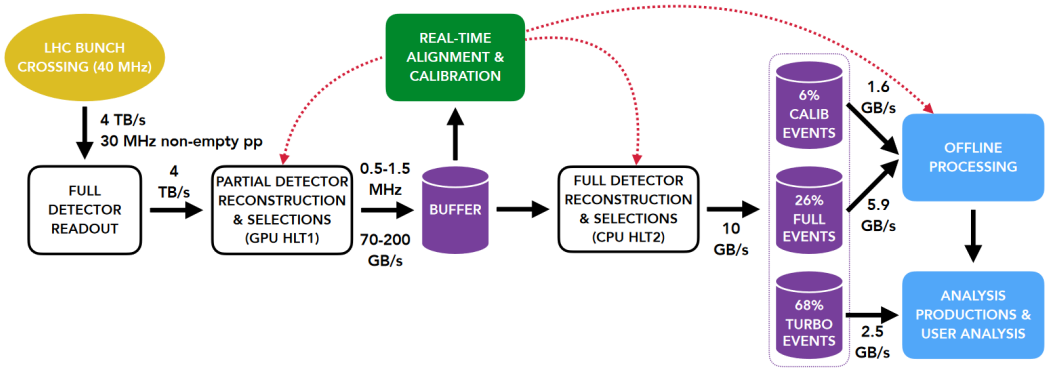


Figure 134: Funcionamiento del procesamiento de datos durante el Run 3 de LHCb. La parte involucrada en el primer nivel de trigger (HLT1) está marcada en la figura.

la Fig. 135 se esquematizan los diferentes algoritmos que se utilizan en el marco del proyecto Allen, en verde se marcan los que involucran al trabajo llevado a cabo en esta tesis.

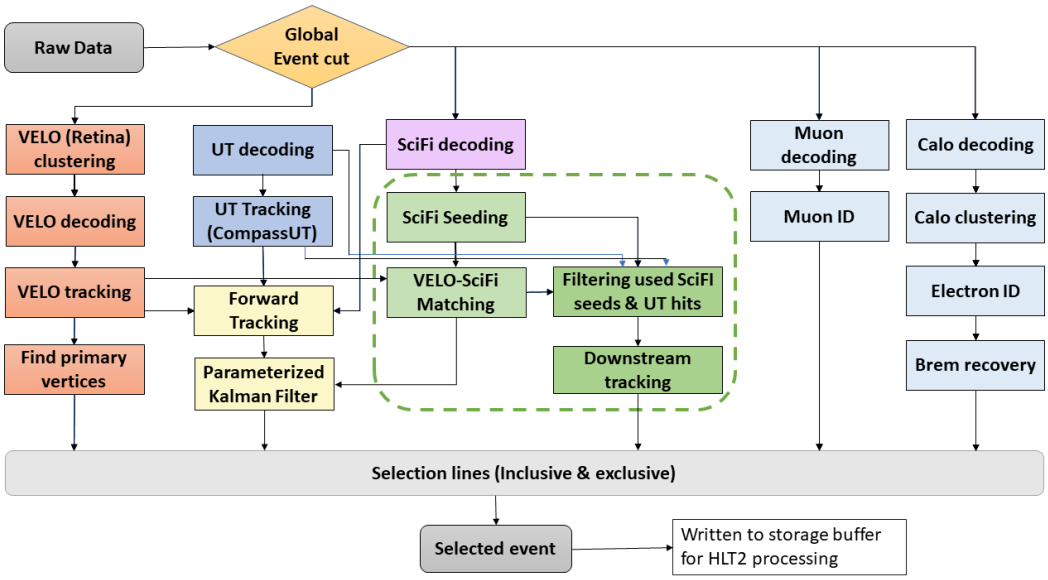


Figure 135: La secuencia de algoritmos ejecutados en HLT1 en el marco de trabajo del proyecto Allen.

9.2 Objetivos

Los objetivos llevados a cabo durante la ejecución de esta tesis, en el marco de trabajo del experimento LHCb, han sido los siguientes:

- Contribución al desarrollo del proyecto Allen para el primer nivel del trigger HLT1 de LHCb.
- Contribución a los algoritmos de `Seeding` y `Matching` en HLT1.
- Desarrollo del nuevo algoritmo de reconstrucción para la detección de partículas de vida media larga en LHCb: el algoritmo `Downstream`.
- Desarrollo de líneas de trigger y propuesta de un programa de trabajo y verificación (*commisioning*) para la puesta en marcha del nuevo algoritmo de `Downstream`.
- Estudio del impacto esperado en la física del Modelo Estándar y más allá de él.
- Estudio del canal $\Lambda_b^0 \to \Lambda\gamma$ y las perspectivas debido al desarrollo de los nuevos algoritmos de reconstrucción.

9.3 Metodología y resultados

9.3.1 Contribución a la portabilidad del proyecto Allen

Las tarjetas gráficas de procesamiento (GPUs) consisten en procesadores masivamente paralelos que contienen miles de núcleos. Cada núcleo puede procesar múltiples datos simultáneamente de forma que estas arquitecturas son ideales si se quieren realizar tareas con un alto grado de repetición, como ocurre al procesar los datos de los subdetectores de LHCb para su selección. Es por ello que el esfuerzo se ha enfocado en el proyecto Allen, donde el principal objetivo ha sido desarollar el primer nivel de trigger de LHCb utilizando estas tarjetas. Además se ha apostado por un sistema compacto, escalable y modular, que pueda desarrollarse y mejorarse con el tiempo. La gestión de la memoria también ha sido un elemento clave en este marco de trabajo, optándose por una memoria estática, y utilizando técnicas concretas como la *memoria coalesciendo* donde se comparten instrucciones entre diferentes *threads*.

Además, Allen utiliza datos en forma de SOAs (estructura de matrices), que permite patrones de acceso a datos contiguos y optimiza la memoria caché. Este proyecto se ha integrado dentro del resto de software de LHCb, de forma que es compatible y puede ser ejecutado junto a otros marcos de trabajo como lo es *Moore*. En esta tesis, se ha hecho especial hincapié en el soporte de plataforma y la portabilidad, para que el diseño de los algoritmos no esté restringidos a un solo tipo de GPUs, sino que pueda abarcar cualquier tipo de arquitectura SIMD (una instrucción, múltiples datos). El método utilzado con este fin ha sido la conversión de todos los algoritmos involucrados, por defecto escritos en CUDA para plataformas de GPUs de Nvidia, para ejecutarse en CPUs x86-64 y plataformas ROCm, proporcionadas por la compañía AMD. En la Fig. 136 se muestra el resultado del rendimiento de la secuencia de Allen en diferentes plataformas, incluidas varias tarjetas proporcionadas por Nvidia (Quaddro RTX 6000, Tesla V100 y Geforce RTX 2080 Ti) así como AMD (AMD M100) e Intel Xeon Broadwell ES-2630.

Como se puede observar, el mejor rendimiento (*throughpout*) lo proporcionan las tarjetas GPUs de Nvidia. Además, durante esta tesis se

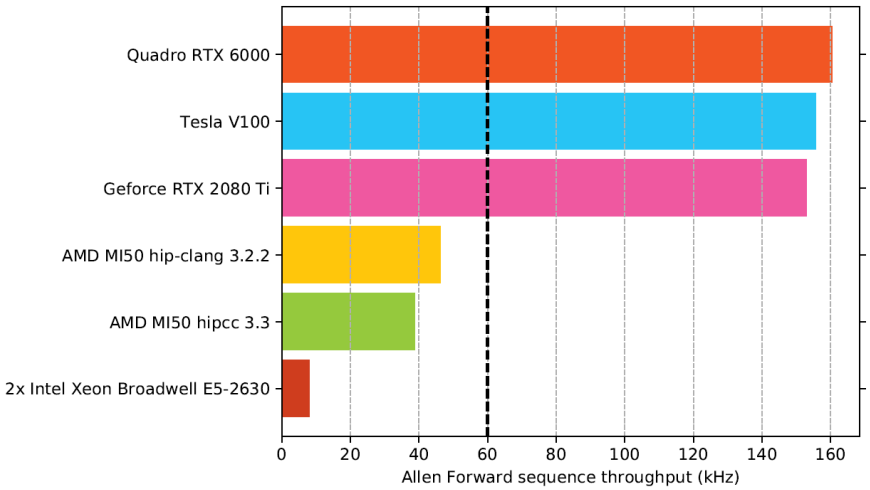


Figure 136: Rendimiento de la secuencia HLT1 de Allen ejecutándose en diferentes arquitecturas .

observó que AMD no proporciona la documentación y soporte técnico necesario para un proyecto de esta embergadura. Este trabajo ha sido decisivo para la elección de la arquitectura utilizada en el primer nivel de trigger de LHCb, y se detallan en el documento Ref. [71].

9.3.2 Contribución a los algoritmos `HybridSeeding` y `VELO-SciFi Matching`

Dentro del proyecto Allen, y una vez elegida la tecnología utilizada (GPUs), en esta tesis se ha contribuido al desarrollo de dos algoritmos clave en HLT1:

- El algoritmo `HybridSeeding` se encarga de la reconstrucción de trazas en el detector SciFi. Consiste en dos partes: un primer algoritmo busca la unión de *h*its para formar *seeds* en el plano $x-z$, incluyendo la información del campo magnético que curva la trayectoria. Un segundo apartado se encarga de incluir la información en el plano $y-z$ proporcionada por las capas u y v del detector. Se buscan combinaciones de dos *h*its en la primera y última capa de la primera y última estación del SciFi, respectivamente. Con esa primera medida se abre una ventana de búsqueda en las otras capas del SciFi, considerando una parábola para la trayectoria. Las trazas de mejor calidad se guardan para el siguiente paso. Se aplica un al-

goritmo de *clone killing* para limpiar trazas espúreas, y se añade por último la información de las capas $u - v$. En la Fig. 137 (izquierda) se muestra la eficiencia de reconstrucción de este algoritmo.

- El algoritmo `VELO-SciFi Matching` se encarga de la reconstrucción de trazas *long*, haciendo uso de los detectores VELO y SciFi. Este algoritmo hace uso de la información proporcionada por el `Hybrid-Seeding`, y de las trazas creadas en el detector VELO. Se buscan las trazas que hacen coincidir los dos detectores en ambos planos, $x - z$ y $x - y$, habiendo hecho un prefiltrado antes, basado en información angular, de las trazas del VELO para evitar trazas falsas. En la Fig. 137 (izquierda) se muestra la eficiencia de reconstrucción de este algoritmo.

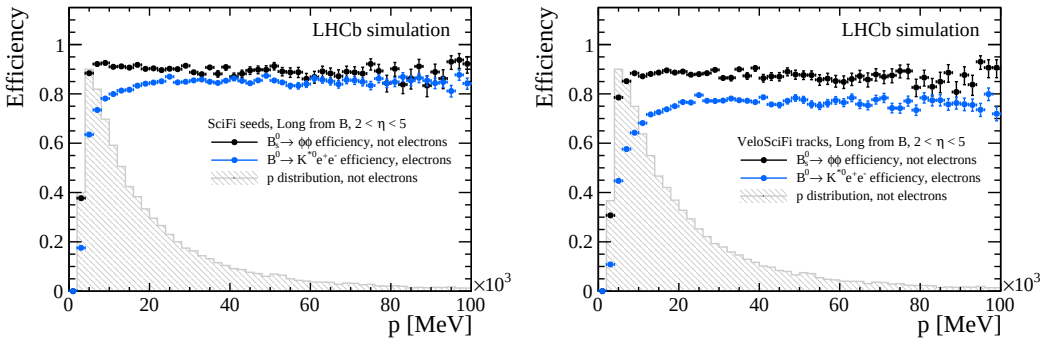


Figure 137: Eficiencia de reconstrucción de los algoritmos `HybridSeeding` (izquierda) y `VELO-SciFi Matching` (derecha).

9.3.3 Desarrollo del nuevo algoritmo `Downstream` para la reconstrucción de partículas de vida media larga

Este nuevo algoritmo es de especial relevancia ya que permite la reconstrucción de partículas de vida media larga. Constituye un trabajo pionero y primordial de esta tesis. Está diseñado en cuatro bloques con las siguientes funciones:

1) Creación de una tabla (SOA) de posibles candidatos *downstream*, con la información proporcionada por la última capa del UT.
2) Llenado de la tabla con las trazas candidatas del UT, utilizando las demás capas del UT.

3) Rechazo de trazas clones y trazas falsas.
4) Preparación de la información de salida del algoritmo.

En el primer bloque el agoritmo filtra las trazas del SciFi, reconstruidas con el `HybridSeeding` que han sido utilizadas para reconstruir trazas *long* mediante los algoritmos `VELO-SciFi Matching` o `Forward`. Cada traza del SciFi remanente se extrapola en la dirección del UT, incluyendo el efecto del campo magnético en la coordenada x, y se abre una ventana de búsqueda en la última capa del UT. Se definen regiones de tolerancia para cada traza del SciFi. Para cada hit del UT se crea un candidato. En un segundo paso se añade la información de las demás capas del UT, abriendo regiones de tolerancia y buscando los *hits* que coincidan con la traza esperada. Una de las características más importantes de este algoritmo es el método de rechazo de trazas falsas. Para ello se utiliza una red neuronal de una sola capa oculta, que tiene una respuesta muy rápida y cumple los requisitos de tiempo del primer nivel de trigger. Esta es la primera vez que se utiliza una red neuronal dentro del proyecto Allen. En el último bloque se prepara la información de salida del algoritmo. Se consolida la salida en estructuras compactas de forma SOA, que incluye las trazas *downstream* creadas, con su correspondiente estado, los *hits* del UT asociados, el índice de la partícula y el tipo de partícula asociado. En la Fig. 138 se muestra la eficiencia de reconstrucción y la tasa de rechazo de trazas falsas de este algoritmo para el canal $\Lambda_b^0 \to \Lambda\gamma$.

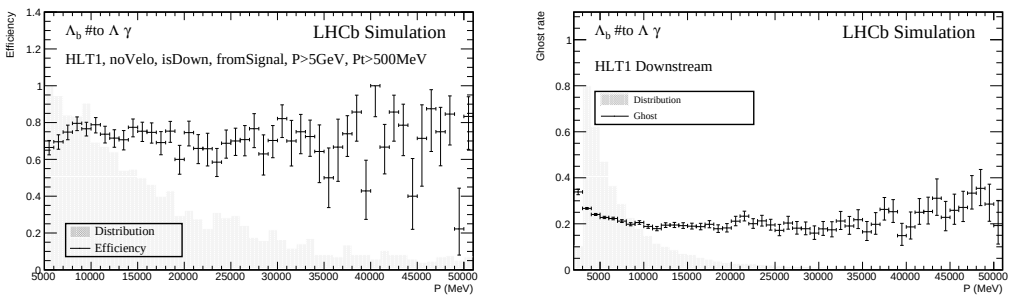


Figure 138: Eficiencia (izquierda) y tasa de trazas falsas (derecha) obtenidas por el algoritmos `Downstream` utilizando el canal $\Lambda_b^0 \to \Lambda\gamma$.

9.3.4 Desarrollo de líneas de trigger y validación del algoritmo `Downstream`

Para poder validar el algortimo de `Downstream` es necesario crear líneas de trigger que seleccionen partículas de vida media larga que no dejan señal en el VELO. Para ello se han desarrollado dos líneas de selección optimizadas para los canales $\Lambda \to p\pi^-$ y $K^0_S \to \pi^+\pi^-$. Estas líneas servirán además de control para la puesta en marcha del detector UT, sirviendo para corroborar su correcto alineamiento y calibración. La metodología utilizada consiste en la extrapolación de las trazas *downstream* a su correpondiente vértice de origen, y la aplicación de un filtro de Kalman dedicado en HLT1 para determinar la intersección (vértice) de dos trazas que provienen de la misma partícula progenitora. En la extrapolación se considera el campo magnético residual en el UT, pero en lugar de utilizar el método de Runge-Kutta, que requiere muchos recursos computacionales, se parametriza con un polinomio de segundo orden. Se determina así una corrección para las trazas en la coordenada x, que depende de q/p. Para la determinación del vértice entre dos trazas se utiliza en un primer momento un método de Newton-Raphson para determinar el punto más cercano (POCA) y a partir de ahí se ajusta el vértice haciendo uso de un filtro de Kalman extendido. Las líneas incluyen además una red neuronal, con información de los parámetros de las trazas finales, los parámetros del vértice, y los paramétros de la traza de la partícula que se desintegra. La red consiste en una sola capa con 7 nodos, y permite reducir el fondo debido a combinaciones aleatorias de una forma muy efectiva. Se han diseñado y entrenado dos redes neuronales, una para cada línea. En la Fig. 139 se muestra el resultado de las combinaciones de trazas seleccionadas para el K^0_S y la Λ. Se observa claramente la distribución de masa, utilizando únicamente trazas *dowsntream* con una muy buena resolución y muy poco fondo. Estas distribuciones de masa pueden servir para alinear y calibrar el detector UT, recientemente instalado, una vez se encuentre en situación *global* junto a los demás detectores. Para ello se hace uso de la herramienta Monet, que propociorna histogramas de las variables deseadas para poder observar *online* el funcionamiento del detector. El alineamiento del detector

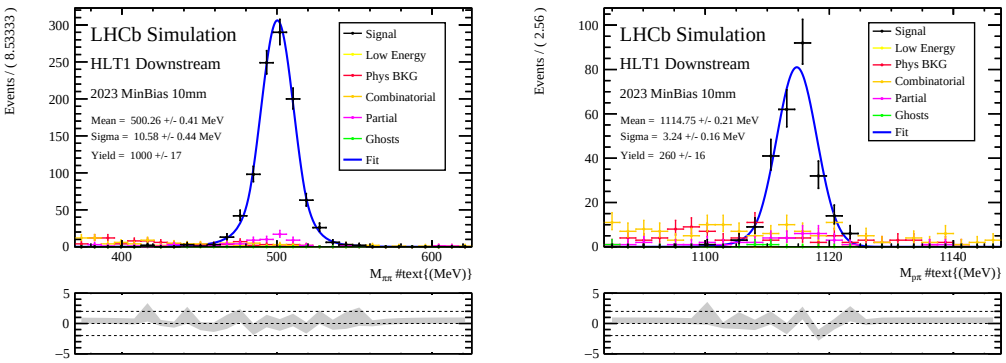


Figure 139: (a) Candidatos K^0_S y (b) Λ que pasan las lineas de trigger HLT1.

se puede realizar utlizando las trazas reconstruidas del K^0_S y Λ, facilitando la información de las correcciones necesarias en los parámetros de alineamiento.

9.3.5 Impacto esperado del algoritmo `Downstream` en el SM y más allá de él

Para estudiar el impacto del algoritmo `Downstream` en los canales de física se ha hecho uso de simulaciones MC. Se han estudiado en detalle dos modelos más allá del SM que presentan partículas de vida media larga: un bosón oscuro (H') en el sector escondido, y un modelo de Higgs compuesto (a_1, a_2) donde tanto el H' como el a_1 pueden tener vidas medias considerables. Estas nuevas partículas se han estudiado a través de desintegraciones de mesones B, adecuadas para su detección en LHCb. Usando una simulación del detector LHCb actualizado, y el generador Phythia8 con las condiciones esperadas para el Run3, se han generado 77 muestras de Monte Carlo (MC) cada una de 7000 sucesos. El canal $B \to H'(\to \mu^+\mu^-)K$ se ha generado considerando la masa del H' en el rango 500 - 4500 MeV/c^2 y vidas medias entre 1 y 2000 ps. Los resultados se muestran en la Fig. 140 (izquierda), en función de un parámetro del modelo que determina el acoplamiento del H' al Higgs del SM, $sin\theta$. La reconstructibilidad en función del tipo de traza muestra claramente una fuerte contribución de trazas *downstream* y T para vidas medias largas. También se ha estudiado otro modelo específico, con un Higgs compuesto, en el que el canal de desintegración

es $B^+ \to K^+ a_1(\to \mu^+\mu^-)a_2(\mu^+\mu^-)$, siendo el a_1 un escalar ligero de larga vida media. Se han generado 44 muestras de 1000 eventos cada una, con masas y vidas medias en el rango de 500-2000 MeV/c^2, y 1-2000 ps. Se ha estudiado la reconstructibilidad en función del tipo de traza, obteniéndose resultados similares al caso del bosón oscuro. Estos resultados se muetsran en la Fig. 140 (derecha), en función de la contante de acoplamiento característica del modelo (g_1).

También se han generado diferentes canales de desintegración del Modelo Estándar que incluyen K^0_S y Λ, y se ha estudiado el impacto esperado del algoritmo `Downstream` en la reconstructibilidad de dichos canales, para trazas *downstream* (DD) y trazas *long* (LL). En la Tabla 25 se resume la proporción esperada.

Canal	Proporción DD/LL	Interés
Hadrones b		
$\Lambda^0_b \to \Lambda\gamma$	3.4	γ polarización, BR
$\Xi^-_b \to \Xi^-\gamma$	25	γ polarización, BR
$\Omega^-_b \to \Omega^-\gamma$	13	γ polarización, BR
$B^+ \to K^0_S K^0_S \pi^+$	2.8	CPV, BR
$B^+ \to K^0_S K^0_S K^+$	2.7	CPV, BR
$B^0_s \to K^0_S K^0_S$	3.6	CPV, BR
Hadrones c		
$\Lambda c^+ \to \Lambda K^+$	4.4	Estudios de polarización
$\Xi^-_c \to \Xi^-\pi^-$	8.4	Estudios de polarización
$D^0 \to K^0_S K^0_S$	1.8	CPV
$J/\psi \to \Lambda\bar{\Lambda}$	4.8	Estudios de polarización, BR
Hadrones s		
$K^0_S \to \mu^+\mu^-$	0.6	BR
$K^0_S \to \mu^+\mu^-\mu^+\mu^-$	0.8	BR
$K^0_S \to \gamma\mu^+\mu^-$	0.8	BR

Table 25: Canales de desintegración en el Modelo Estándar que pueden beneficiarse de la reconstrucción del algoritmo `Downstream` en el nivel HLT1. La segunda columna representa la proporción de trazas *downstream* sobre las *long*.

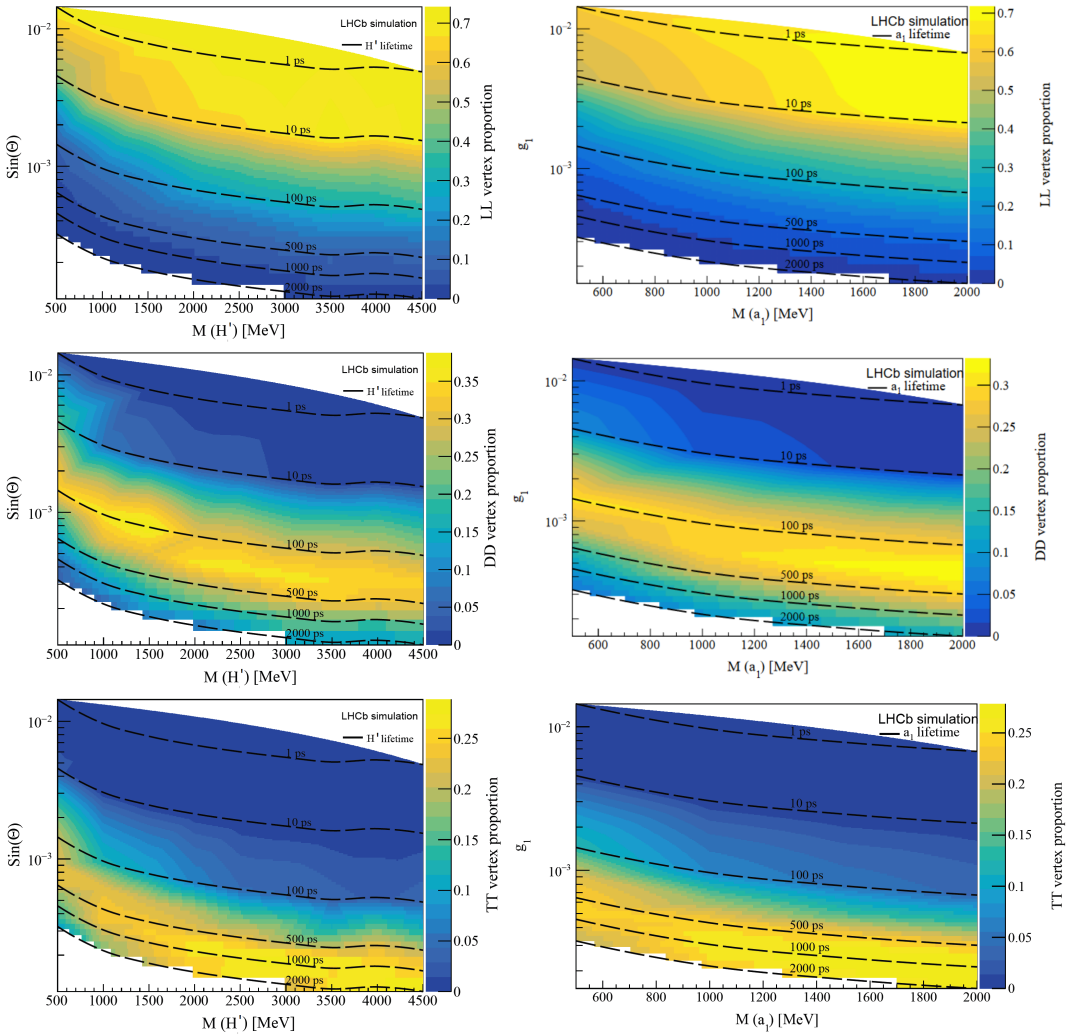


Figure 140: Reconstructibilidad del vértice de desintegración de las partículas H′ (iquierda) y a_1 (derecha) en función de su masa y vida vedia. Se muestran diferentes topologías, de arriba a abajo: dos trazas *long* (LL), dos trazas *downstream* (DD) y dos trazas *T* (TT), para la reconstrucción de los dos muones.

Cabe destacar la importancia de los canales bariónicos, tanto en el sector b como en el *charm*.

9.3.6 Estudio del canal $\Lambda_b^0 \to \Lambda\gamma$

La reconstrucción del canal de desintegración $\Lambda_b^0 \to \Lambda\gamma$ es un reto ya que la Λ es una partícula neutra que tiene además un tiempo de vida largo. Este canal es de especial interés porque al ser una transición $b \to s\gamma$ se tiene que producir a través de *loops* y es muy sensible a nueva física. Se han estudiado dos observables de interés en este canal: la fracción de desintegración, $\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)$, y la polarización del fotón, α_γ, que el Modelo Estándar predice que tiene que ser levógira. Para la reconstrucción de los sucesos se combina un protón y un pión negativo para reconstruir el vértice de la Λ, además de seleccionar un fotón de alto momento. Se utilizan líneas de trigger y filtrado (*stripping*) específicas optimizadas para la selección de este canal, así como una selección mediante una técnica de aprendizaje automático basada en árboles de regresión (BDT). Muchas de las variables que se utilizan tienen que ver con la topología de las desintegraciones en estudio. La fracción de desintegración se determina a partir de la relación

$$\frac{\mathcal{B}(\Lambda_b^0 \to \Lambda\gamma)}{\mathcal{B}(B^0 \to K^{*0}\gamma)} = \frac{N_{\Lambda_b^0 \to \Lambda\gamma}}{N_{B^0 \to K^{*0}\gamma}} \times \frac{f_{B^0}}{f_{\Lambda_b^0}} \times \frac{\mathcal{B}(K^* \to K^+\pi^-)}{\mathcal{B}(\Lambda \to p\pi^-)} \times \frac{\epsilon_{\text{sel}}^{B^0 \to K^{*0}\gamma}}{\epsilon_{\text{sel}}^{\Lambda_b^0 \to \Lambda\gamma}}, \quad (9.2)$$

donde $\mathcal{B}(K^* \to K^+\pi^-) = (66.503 \pm 0.014)\%$, y $\mathcal{B}(\Lambda \to p\pi^-) = (64.1 \pm 0.5)\%$ [101]. El valor de la $\mathcal{B}(K^* \to K^+\pi^-)$ se obtiene utilizando reglas de isospín. Se considera que $\frac{f_{\Lambda_b}}{f_u + f_d} = 0.259 \pm 0.018$, y que $f_u = f_d = 0.340 \pm 0.021$ (CDF [108]), de forma que se obtiene $\frac{f_{\Lambda_b}}{f_B} = 0.518 \pm 0.036$. Las eficiencias ϵ_i se corresponden a la fracción de sucesos que sobrevive en cada uno de los pasos del trigger, reconstrucción y selección. Se utilizan simulaciones Monte Carlo para el entrenamiento de la señal y datos reales para el fondo. Para la medida de la fracción de desintegración se han estudiado diferentes canales de normalización: $\Lambda_b^0 \to J/\psi\Lambda$, $B^0 \to K^{*0}\gamma$, $B_s^0 \to \phi\gamma$ y $\Lambda_b^0 \to J/\psi p K^-$. Se ha concluido que el canal que proporciona mejor precisión en la medida es el canal $B^0 \to K^{*0}\gamma$, ya que se conoce con relativa precisión su fracción de desintegración. Se

han evaluado los diferentes efectos que se espera que intervengan en la precisión de la medida, como son las fracciones de desintegración, la sustracción del fondo, el cálculo de las eficiencias y otras incertidumbres sistemáticas. Uno de los principales estudios conlleva a la necesidad de la inclusión del Run3 y la utilización de trazas *downstream* para conseguir una buena precisión de la medida. Para obtener el número de sucesos de los canales señal y de normalización se utiliza un ajuste de masa. La forma de la señal se parametriza utilizando una función asimétrica DSCB (*double sided Crystal-Ball*). Los principales fondos que afectan la medida consisten en combinaciones aleatorias de trazas de piones y protones (o kaones) con un fotón enegético, cuya distribución de masa se puede aproximar a una exponencial o a un polinomio de primer grado. Además varios procesos físicos que no se reconstruyen completamente pueden contribuir al fondo. El canal de desintegración $\Lambda_b^0 \to \Lambda\eta$, donde la η se desintegra en dos fotones, es el más relevante, aunque su contribución es pequeña. El canal $B^0 \to K^{*0}\gamma$ está afectado por una variedad de fondos debido a procesos reconstruidos parcialmente. Para determinar su distribución de masa se utilizan datos simulados. La resolución de masa de los canales señal y de normalización es de aproximadamente $95\,\mathrm{MeV/c^2}$. Una vez determinada la estrategia, para completar el análisis se necesita alinear los criterios de selección de los canales $\Lambda_b^0 \to \Lambda\gamma$ y $B^0 \to K^{*0}\gamma$, realizar el cálculo detallado de las eficiencias que intervienen en la Eq. 9.2, adoptar un sistema de ajuste simultáneo a las distribuciones de masa de los canales señal y de normalización, y realizar un estudio detallado de las incertidumbres sistemáticas que afectan a la medida. Además la inclusión de los datos del Run3 con trazas *downstream* es decisiva para obtener una medida precisa. Se esperan 7500 sucesos $\Lambda_b^0 \to \Lambda\gamma$ y 653771 $B^0 \to K^{*0}\gamma$ considerando una luminosidad integrada de $23\,\mathrm{fb}^{-1}$. Con los sucesos señal también se puede medir otro observable: la polarización del fotón. Para ello se puede utilizar la distribución angular

$$W(\theta_p) \propto 1 - \alpha_\gamma \alpha_\Lambda \cos\theta_p. \tag{9.3}$$

donde el ángulo θ_p es la helicidad del fotón en el sistema de la partícula Λ. El análisis se realiza en la región de masa de 5387 a 5852 MeV/c^2.

Con los datos del Run3 que incluyen trazas *downstream*, y teniendo en cuenta que la resolución angular es de $\sigma_{\theta DD} = 36 \pm 3\,\text{mrad}$, en comparación con $\sigma_{\theta LL} = 22 \pm 5\,\text{mrad}$, para trazas *long*, la precisión esperada es menor al 10%, que puede proporcionar claros indicios de nueva física. En la Fig. 141 se muestra un ejemplo de una simulación del canal $\Lambda_b^0 \to \Lambda\gamma$, su distribución de masa (izquierda) y el correspondiente ajuste a la polarización del fotón (derecha). Para describir el fondo combinatorial se utiliza un polinomio de cuarto grado. También se incluye el efecto de la aceptancia debido a la geometría y a efectos de reconstrucción en el detector, parametrizado con un polinomio del mismo grado.

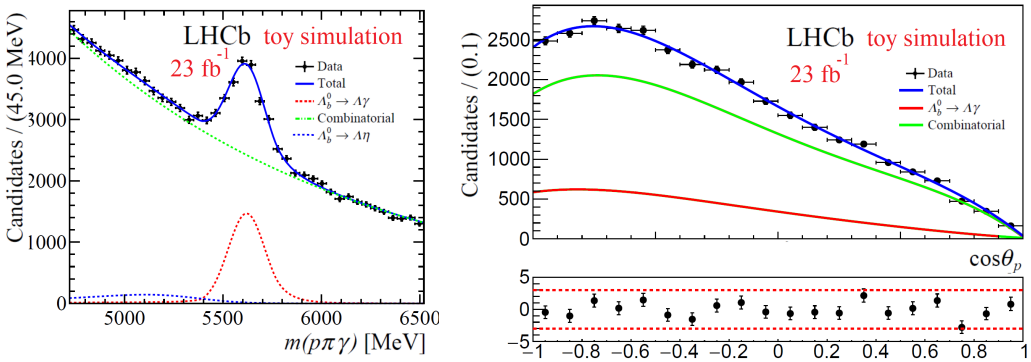


Figure 141: Ejemplo de un ajuste de masa y de la medida de la polarización del fotón asumiendo una luminosidad integrada en LHCb de 23 fb^{-1} para el Run3 e incluyendo trazas *downstream* provenientes de las desintegraciones de partículasΛ.

9.4 Conclusiones

El trabajo de esta tesis es una contribución capital al primer nivel de trigger del experimento LHCb, en el marco del proyecto Allen, donde toda la secuencia de algoritmos de reconstrucción se ejecuta en tarjetas GPUs. Se ha contribuido al desarrollo del proyecto, haciendo posible que el trigger de LHCb sleccione con éxito los datos de interés en tiempo real a una frecuencia de 30 MHz. Además, se ha contribuido al diseño y creación de varios algoritmos fundamentales. Por una parte el algoritmo `HybridSeeding` proporciona las trazas detectadas en el detector SciFi con

gran eficiencia (trazas T). El algoritmo `VELO-SciFi Matching`, que hace uso del algoritmo anterior, proporciona a su vez la reconstrucción de trazas *long* con una precisión en momento mayor al 1%. Además, se ha creado e incluido por primera vez en HLT1 un algoritmo, `Downstream`, para la selección de partículas de vida media larga. Este algoritmo abre un área de física tanto en el Modelo Estándar como más allá de él sin precedentes. En particular implica la ampliación del rango de búsquedas de partículas exóticas de vida media larga en el rango de 100 ps a varios nanosegundos, inexplorado hasta la fecha en LHCb. La sensibilidad a nuevas partículas predichas en teorías que incluyen un sector oscuro, leptones neutros pesados, supersimetría o partículas tipo axión se verá incrementada en gran medida. Asimismo se incrementará la eficiencia de detección para partículas del Modelo Estándar como Λ y K^0_S, mejorándose la precisión de muchos análisis que involucran desintegraciones de hadrones con quarks b y c. En especial, el aumento de la precisión en la medida de observables en el canal de desintegración $\Lambda^0_b \to \Lambda\gamma$ como son la fracción de desintegración o la polarización del fotón será decisivo para revelar nueva física.

Bibliography

[1] G. Aad et al. "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC". In: *Phys. Lett.* B716 (2012), pp. 1–29. arXiv: 1207.7214 [hep-ex] (cit. on p. 2).

[2] S. Chatrchyan et al. "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC". In: *Phys. Lett.* B716 (2012), pp. 30–61. arXiv: 1207.7235 [hep-ex] (cit. on p. 2).

[3] M. E. Peskin and D. V. Schroeder. *An introduction to quantum field theory*. Boulder, CO: Westview, 1995. URL: https://cds.cern.ch/record/257493 (cit. on p. 2).

[4] P. W. Higgs. "Broken Symmetries and the Masses of Gauge Bosons". In: *Phys. Rev. Lett.* 13 (1964), pp. 508–509 (cit. on p. 3).

[5] F. Englert and R. Brout. "Broken Symmetry and the Mass of Gauge Vector Mesons". In: *Phys. Rev. Lett.* 13 (1964), pp. 321–323 (cit. on p. 3).

[6] L. Lee et al. "Collider searches for long-lived particles beyond the Standard Model". In: *Progress in Particle and Nuclear Physics* 106 (May 2019), pp. 210–255. URL: https://doi.org/10.1016%2Fj.ppnp.2019.02.006 (cit. on pp. 5, 238).

[7] C. Kolda. "Gauge-mediated supersymmetry breaking: Introduction, review and update". In: *Nuclear Physics B - Proceedings Supplements* 62.1-3 (Mar. 1998), pp. 266–275. URL: https://doi.org/10.1016%2Fs0920-5632%2897%2900667-1 (cit. on p. 8).

[8] S. M. Barr. "Solving the Strong CP Problem without the Peccei-Quinn Symmetry". In: *Phys. Rev. Lett.* 53 (4 July 1984), pp. 329–332. URL: https://link.aps.org/doi/10.1103/PhysRevLett.53.329 (cit. on p. 9).

[9] R. Aaij et al. "Search for massive long-lived particles decaying semileptonically in the LHCb detector". In: *Eur. Phys. J.* C77 (2017), p. 224. arXiv: 1612.00945 [hep-ex] (cit. on p. 9).

[10] R. Aaij et al. "Updated search for long-lived particles decaying to jet pairs". In: *Eur. Phys. J.* C77 (2017), p. 812. arXiv: 1705.07332 [hep-ex] (cit. on p. 9).

[11] R. Aaij et al. "Search for Higgs-like boson decaying into pair of long-lived particles". In: *Eur. Phys. J.* C76 (2016), p. 664. arXiv: 1609.03124 [hep-ex] (cit. on p. 9).

[12] R. Aaij et al. "Search for Majorana neutrinos in $B^- \to \pi^+\mu^-\mu^-$ decays". In: *Phys. Rev. Lett.* 112 (2014), p. 131802. arXiv: 1401.5361 [hep-ex] (cit. on p. 9).

[13] R. Aaij et al. "Searches for Majorana neutrinos in B^- decays". In: *Phys. Rev.* D85 (2012), p. 112004. arXiv: 1201.5600 [hep-ex] (cit. on p. 9).

[14] R. Aaij et al. "Search for hidden-sector bosons in $B^0 \to K^{*0}\mu^+\mu^-$ decays". In: *Phys. Rev. Lett.* 115 (2015), p. 161802. arXiv: 1508.04094 [hep-ex] (cit. on p. 9).

[15] R. Aaij et al. "Search for long-lived scalar particles in $B^+ \to K^+\chi(\mu^+\mu^-)$ decays". In: *Phys. Rev.* D95 (2017), p. 071101. arXiv: 1612.07818 [hep-ex] (cit. on p. 9).

[16] R. Aaij et al. "Search for long-lived heavy charged particles using a ring-imaging Cherenkov technique at LHCb". In: *Eur. Phys. J.* C75 (2015), p. 595. arXiv: 1506.09173 [hep-ex] (cit. on p. 10).

[17] R. Aaij et al. "Search for $A' \to \mu^+\mu^-$ Decays". In: *Phys. Rev. Lett.* 124.4 (2020), p. 041801. arXiv: 1910.06926 [hep-ex] (cit. on p. 10).

[18] ATLAS Collaboration. In: *ATLAS public web page* (2023). URL: https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2023-008/fig_01.png (cit. on p. 11).

[19] B. Acharya et al. "MoEDAL-MAPP, an LHC Dedicated Detector Search Facility". In: *Snowmass 2021*. Sept. 2022. arXiv: 2209.03988 [hep-ph] (cit. on pp. 11, 239).

[20] D. Curtin et al. "Long-lived particles at the energy frontier: the MATHUSLA physics case". In: *Reports on Progress in Physics* 82.11 (Oct. 2019), p. 116201. URL: https://doi.org/10.1088%2F1361-6633%2Fab28d6 (cit. on pp. 11, 239).

[21] G. Aielli et al. *The Road Ahead for CODEX-b*. 2022. arXiv: 2203.07316 [hep-ex] (cit. on pp. 11, 239).

[22] J. L. Feng et al. "ForwArd Search ExpeRiment at the LHC". In: *Physical Review D* 97.3 (Feb. 2018). URL: https://doi.org/10.1103%2Fphysrevd.97.035001 (cit. on pp. 11, 239).

[23] S. Collaboration. *The SHiP experiment at the proposed CERN SPS Beam Dump Facility*. 2022. arXiv: 2112.01487 [physics.ins-det] (cit. on pp. 11, 239).

[24] J. Alimena et al. "Searching for long-lived particles beyond the Standard Model at the Large Hadron Collider". In: *Journal of Physics G: Nuclear and Particle Physics* 47.9 (Sept. 2020), p. 090501. URL: https://dx.doi.org/10.1088/1361-6471/ab4574 (cit. on pp. 11, 239).

[25] A. D. Sakharov. "Violation of CP Invariance, C Asymmetry, and Baryon Asymmetry of the Universe". Trans. by J. Letters. In: *Pisma Zh. Eksp. Teor. Fiz.* 5 (1967). Reprinted in Usp. Fiz. Nauk 161, 61 (1991) [Sov. Phys. Usp. 34, 392 (1991)], pp. 32–35 (cit. on p. 12).

[26] J. H. Christenson et al. "Evidence for the 2π Decay of the K_2^0 Meson". In: *Phys. Rev. Lett.* 13 (1964), pp. 138–140 (cit. on p. 13).

[27] B. Aubert et al. "Observation of CP violation in the B^0 meson system". In: *Phys. Rev. Lett.* 87 (2001), p. 091801. arXiv: hep-ex/0107013 (cit. on p. 13).

[28] K. Abe et al. "Observation of large CP violation in the neutral B meson system". In: *Phys. Rev. Lett.* 87 (2001), p. 091802. arXiv: hep-ex/0107061 (cit. on p. 13).

[29] R. Aaij et al. "Observation of CP Violation in Charm Decays". In: *Phys. Rev. Lett.* 122.21 (2019), p. 211803. arXiv: 1903.08726 [hep-ex] (cit. on p. 13).

[30] L. Wolfenstein. "Parametrization of the Kobayashi-Maskawa Matrix". In: *Phys. Rev. Lett.* 51 (1983), p. 1945 (cit. on p. 14).

[31] W. Qian. "Recent CKMfitter updates on global fits of the CKM matrix". In: *PoS* CKM2021 (2023), p. 074 (cit. on pp. 14, 15).

[32] R. Ammar et al. "Evidence for penguin-diagram decays: First observation of B→$K^*(892)\gamma$". In: *Phys. Rev. Lett.* 71 (5 Aug. 1993), pp. 674–678. URL: https://link.aps.org/doi/10.1103/PhysRevLett.71.674 (cit. on p. 18).

[33] J. P. Lees et al. "Precision measurement of the $B \to X_s\gamma$ photon energy spectrum, branching fraction, and direct CP asymmetry $A_{CP}(B \to X_{s+d}\gamma)$". In: *Phys. Rev. Lett.* 109 (2012), p. 191801. arXiv: 1207.2690 [hep-ex] (cit. on pp. 18, 201).

[34] T. Saito et al. "Measurement of the $\bar{B} \to X_s\gamma$ branching fraction with a sum of exclusive decays". In: *Phys. Rev.* D91.5 (2015), p. 052004. arXiv: 1411.7198 [hep-ex] (cit. on pp. 18, 201).

[35] T. Horiguchi et al. "Evidence for isospin violation and measurement of CP asymmetries in $B \to K^*(892)\gamma$". In: *Phys. Rev. Lett.* 119.19 (2017), p. 191802. arXiv: 1707.00394 [hep-ex] (cit. on pp. 18, 201).

[36] Y. Ushiroda et al. "Time-dependent CP asymmetries in $B^0 \to K^0_S\pi^0\gamma$ transitions". In: *Phys. Rev.* D74 (2006), p. 111104. arXiv: hep-ex/0608017 [hep-ex] (cit. on pp. 18, 201).

[37] B. Aubert et al. "Measurement of time-dependent CP asymmetry in $B^0 \to K^0_S\pi^0\gamma$ decays". In: *Phys. Rev.* D78 (2008), p. 071102. arXiv: 0807.3103 [hep-ex] (cit. on pp. 18, 201).

[38] R. Aaij et al. "Measurement of the ratio of branching fractions $BR(B_0 \to K^{\star 0}\gamma)/BR(B_{s0} \to \phi\gamma)$ and the direct CP asymmetry in $B_0 \to K^{\star 0}\gamma$". In: *Nucl. Phys. B* 867 (2013), pp. 1–18. arXiv: 1209.0313 [hep-ex] (cit. on pp. 18, 205).

[39] R. Aaij et al. "Observation of Photon Polarization in the b→sγ Transition". In: *Phys. Rev. Lett.* 112.16 (2014), p. 161801. arXiv: 1402.6852 [hep-ex] (cit. on p. 18).

[40] R. Aaij et al. "First experimental study of photon polarization in radiative B_s^0 decays". In: *Phys. Rev. Lett.* 118.2 (2017). [Addendum: Phys.Rev.Lett. 118, 109901 (2017)], p. 021801. arXiv: 1609.02032 [hep-ex] (cit. on p. 18).

[41] R. Aaij et al. "Measurement of *CP*-violating and mixing-induced observables in $B_s^0 \to \phi\gamma$ decays". In: *Phys. Rev. Lett.* 123.8 (2019), p. 081802. arXiv: 1905.06284 [hep-ex] (cit. on p. 18).

[42] R. Aaij et al. "First Observation of the Radiative Decay $\Lambda_b^0 \to \Lambda\gamma$". In: *Phys. Rev. Lett.* 123 (3 July 2019), p. 031801. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.031801 (cit. on pp. 18, 202, 206, 210, 225).

[43] L. M. Garcia Martin. "Search for radiative *b*-baryon decays and study of their anomalous photon polarization at LHCb". Presented 29 May 2020. URL: http://cds.cern.ch/record/2799936 (cit. on pp. 18, 202, 206, 208, 229).

[44] Y.-M. Wang, Y. Li, and C.-D. Lü. "Rare decays of $\Lambda_b^0 \to \Lambda\gamma$ and $\Lambda_b^0 \to \Lambda\ell^+\ell^-$ in the light-cone sum rules". In: *Eur. Phys. J.* C59 (2009), pp. 861–882. arXiv: 0804.0648 [hep-ph] (cit. on pp. 18, 202).

[45] T. Mannel and Y.-M. Wang. "Heavy-to-light baryonic form factors at large recoil". In: *JHEP* 12 (2011), p. 067. arXiv: 1111.1849 [hep-ph] (cit. on pp. 18, 202).

[46] R.-M. Wang et al. "Studying radiative baryon decays with the SU(3) flavor symmetry". In: *Journal of Physics G: Nuclear and Particle Physics* 48.8 (June 2021), p. 085001. URL: https://dx.doi.org/10.1088/1361-6471/abffdd (cit. on pp. 18, 202).

[47] R. Aaij et al. "Strong constraints on the $b \to s\gamma$ photon polarisation from $B^0 \to K^{*0}e^+e^-$ decays". In: *JHEP* 12 (2020), p. 081. arXiv: 2010.06011 [hep-ex] (cit. on p. 19).

[48] R. Aaij et al. "Measurement of the photon polarization in $\Lambda_b^0 \to \Lambda\ \gamma$ decays". In: *Phys. Rev. D* 105.5 (2022), p. L051104. arXiv: 2111.10194 [hep-ex] (cit. on pp. 20, 194, 206, 208, 218, 219, 222–224, 228, 229).

[49] C. Elsasser. *$\bar{b}b$ production angle plots*. 2023. URL: %7B%5Chref%7Bhttps://lhcb.web.cern.ch/lhcb/speakersbureau/html/bb%5C_ProductionAngles.html%7D%7D (cit. on p. 25).

[50] *LHCb Trigger and Online Technical Design Report*. LHCb-TDR-016. Geneva, 2014 (cit. on p. 28).

[51] L. Collaboration. *LHCb Tracker Upgrade Technical Design Report*. Tech. rep. 2014. URL: https://cds.cern.ch/record/1647400 (cit. on p. 29).

[52] A. A. Alves et al. "The LHCb Detector at the LHC". In: *JINST* 3 (2008). Also published by CERN Geneva in 2010, S08005. URL: https://cds.cern.ch/record/1129809 (cit. on p. 32).

[53] L. Collaboration. *LHCb PID Upgrade Technical Design Report*. Tech. rep. 2013. URL: https://cds.cern.ch/record/1624074 (cit. on p. 32).

[54] L. Anderlini et al. "Muon identification for LHCb Run 3". In: *JINST* 15.12 (2020), T12005. arXiv: 2008.01579. URL: https://cds.cern.ch/record/2727496 (cit. on p. 36).

[55] R. Aaij et al. "A comprehensive real-time analysis model at the LHCb experiment". In: *JINST* 14.04 (2019). 15 pages, 4 figures, 1 table, P04006. arXiv: 1903.01360. URL: https://cds.cern.ch/record/2665946 (cit. on pp. 41, 44).

[56] R. Aaij et al. "Tesla: an application for real-time data analysis in High Energy Physics". In: *Comput. Phys. Commun.* 208 (2016), pp. 35–42. arXiv: 1604.05596 [physics.ins-det] (cit. on pp. 41, 44).

[57] C. Fitzpatrick and V. V. Gligorov. *Anatomy of an upgrade event in the upgrade era, and implications for the LHCb trigger*. Tech. rep. Geneva: CERN, 2014. URL: https://cds.cern.ch/record/1670985 (cit. on p. 41).

[58] LHCb collaboration. "RTA and DPA dataflow diagrams for Run 1, Run 2, and the upgraded LHCb detector". In: (2020). URL: https://cds.cern.ch/record/2730181 (cit. on pp. 43, 47).

[59] I. Bediaga et al. *Computing Model of the Upgrade LHCb experiment*. Tech. rep. Geneva: CERN, 2018. URL: https://cds.cern.ch/record/2319756 (cit. on p. 44).

[60] G. Barrand et al. "GAUDI — A software architecture and framework for building HEP data processing applications". In: *Computer Physics Communications* 140.1 (2001). CHEP2000, pp. 45–55 (cit. on p. 45).

[61] F. Gaede et al. "DD4hep a community driven detector description for HEP". In: *EPJ Web Conf.* 245 (2020), p. 02004. URL: https://cds.cern.ch/record/2719129 (cit. on p. 46).

[62] R. Brun and F. Rademakers. "ROOT: An object oriented data analysis framework". In: *Nucl. Instrum. Meth.* A389 (1997), pp. 81–86 (cit. on p. 47).

[63] T. Sjöstrand, S. Mrenna, and P. Z. Skands. "A Brief Introduction to PYTHIA 8.1". In: *Comput. Phys. Commun.* 178 (2008), pp. 852–867. arXiv: 0710.3820 [hep-ph] (cit. on p. 48).

[64] D. J. Lange. "The EvtGen particle decay simulation package". In: *Nucl. Instrum. Meth.* A462 (2001), pp. 152–155 (cit. on p. 48).

[65] S. Agostinelli et al. "GEANT4: A Simulation toolkit". In: *Nucl. Instrum. Meth.* A506 (2003), pp. 250–303 (cit. on p. 48).

[66] R. Aaij et al. *The LHCb upgrade I*. Tech. rep. 2023. arXiv: 2305.10515 (cit. on p. 48).

[67] I. Bird et al. "Computing for the Large Hadron Collider". In: *Annual Review of Nuclear and Particle Science* 59 (2009), pp. 99–118 (cit. on p. 49).

[68] A. Tsaregorodtsev et al. "DIRAC: a community grid solution". In: *J. Phys. Conf. Ser.* 219 (2010), p. 062029 (cit. on p. 50).

[69] R. Aaij et al. "Allen: A high level trigger on GPUs for LHCb. Allen: A high level trigger on GPUs for LHCb". In: *Comput. Softw. Big Sci.* 4.1 (2020). 12 pages, 12 figures, submitted to Computing and Software for Big Science, p. 7. arXiv: 1912.09161. URL: https://cds.cern.ch/record/2704717 (cit. on pp. 52, 77, 79).

[70] NVIDIA, P. Vingelmann, and F. H. Fitzek. *CUDA, release: 12.2*. 2022. URL: https://developer.nvidia.com/cuda-toolkit (cit. on pp. 53, 55).

[71] R. Aaij et al. "A Comparison of CPU and GPU Implementations for the LHCb Experiment Run 3 Trigger". In: *Computing and Software for Big Science* 6.1 (Dec. 2021). URL: https://doi.org/10.1007%2Fs41781-021-00070-2 (cit. on pp. 66, 246).

[72] F. Lazzari et al. "Real-time cluster finding for LHCb silicon pixel VELO detector using FPGA". In: *J. Phys.: Conf. Ser.* 1525.1 (2020), p. 012044. URL: https://cds.cern.ch/record/2744309 (cit. on p. 72).

[73] P. F. Declara et al. "A Parallel-Computing Algorithm for High-Energy Physics Particle Tracking and Decoding Using GPU Architectures". In: *IEEE Access* 7 (2019), pp. 91612–91626. URL: https://doi.org/10.1109%2Faccess.2019.2927261 (cit. on pp. 72, 78, 79).

[74] S. Esen et al. *Clustering and rawbank decoding for the SciFi detector*. Tech. rep. Geneva: CERN, 2018. URL: https://cds.cern.ch/record/2630154 (cit. on p. 73).

[75] D. H. Cámpora Pérez, N. Neufeld, and A. Riscos Núñez. "Search by triplet: An efficient local track reconstruction algorithm for parallel architectures". In: *Journal of Computational Science* 54 (2021), p. 101422. URL: https://www.sciencedirect.com/science/article/pii/S1877750321001071 (cit. on p. 77).

[76] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1 (Mar. 1960), pp. 35–45. eprint: https://asmedigitalcollection.asme.org/

fluidsengineering/article-pdf/82/1/35/5518977/35_1.pdf. URL: https://doi.org/10.1115/1.3662552 (cit. on pp. 77, 171).

[77] S. Aiola et al. "Hybrid seeding: A standalone track reconstruction algorithm for scintillating fibre tracker at LHCb". In: *Computer Physics Communications* 260 (Mar. 2021), p. 107713. URL: https://doi.org/10.1016/j.cpc.2020.107713 (cit. on pp. 79, 84).

[78] B. K. Jashal. "Standalone track reconstruction and matching algorithms for GPU-based High level trigger at LHCb". In: (2022). URL: https://cds.cern.ch/record/2826068 (cit. on pp. 79, 84).

[79] P. A. Günther. "LHCb's Forward Tracking algorithm for the Run 3 CPU-based online track reconstruction sequence". In: (2022). URL: https://cds.cern.ch/record/2819858 (cit. on p. 81).

[80] P. Billoir et al. "A parametrized Kalman filter for fast track fitting at LHCb". In: *Computer Physics Communications* 265 (Aug. 2021), p. 108026. URL: https://doi.org/10.1016%2Fj.cpc.2021.108026 (cit. on p. 81).

[81] N. V. Canudas et al. "Graph Clustering: a graph-based clustering algorithm for the electromagnetic calorimeter in LHCb". In: *Eur. Phys. J. C* 83.2 (2023). 12 pages, 9 figures, submitted to EPJ C, p. 179. arXiv: 2212.11061. URL: https://cds.cern.ch/record/2846012 (cit. on p. 83).

[82] S. Esen and M. De Cian. *A Track Matching Algorithm for the LHCb upgrade*. Tech. rep. Geneva: CERN, 2016. URL: https://cds.cern.ch/record/2238266 (cit. on pp. 84, 89).

[83] L. Calefice. "Standalone track reconstruction on GPUs in the first stage of the upgraded LHCb trigger system and Preparations for measurements with strange hadrons in Run3". Presented 13 Dec 2022. 2022. URL: https://cds.cern.ch/record/2856339 (cit. on pp. 84, 89).

[84] P. Li, E. Rodrigues, and S. Stahl. "Tracking Definitions and Conventions for Run 3 and Beyond, *LHCb-PUB-2021-005*." In: (Feb. 2021). URL: https://cds.cern.ch/record/2752971 (cit. on p. 92).

[85] A. Davis et al. *PatLongLivedTracking: a tracking algorithm for the reconstruction of the daughters of long-lived particles in LHCb*. Tech. rep. Geneva: CERN, 2017. URL: https://cds.cern.ch/record/2240723 (cit. on p. 100).

[86] A. F. Agarap. "Deep learning using rectified linear units (relu)". In: *arXiv preprint arXiv:1803.08375* (2018) (cit. on p. 123).

[87] D. P. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG] (cit. on p. 124).

[88] M. Adinolfi et al. "LHCb data quality monitoring". In: *Journal of Physics: Conference Series* 898.9 (Oct. 2017), p. 092027. URL: https://dx.doi.org/10.1088/1742-6596/898/9/092027 (cit. on p. 168).

[89] K. S. Hashemi and J. Bensinger. *The BCAM Camera*. Tech. rep. Geneva: CERN, 2000. URL: https://cds.cern.ch/record/684119 (cit. on p. 170).

[90] J.-P. Dedieu. "Newton-Raphson Method". In: *Encyclopedia of Applied and Computational Mathematics*. Ed. by B. Engquist. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 1023–1028. URL: https://doi.org/10.1007/978-3-540-70529-1_374 (cit. on pp. 175, 184).

[91] T. Skwarnicki. "A study of the radiative CASCADE transitions between the Upsilon-Prime and Upsilon resonances". PhD thesis. Cracow, INP, 1986 (cit. on pp. 182, 221).

[92] F. Reiss and L. Collaboration. "Real-time alignment procedure at the LHCb experiment for Run 3". In: (2023). URL: https://cds.cern.ch/record/2846414 (cit. on p. 183).

[93] A. Kachanovich, U. Nierste, and I. Nišandžić. "Higgs portal to dark matter and $B \to K^{(*)}$ decays". In: *The European Physical Journal C* 80.7 (2020), p. 669. URL: https://doi.org/10.1140/epjc/s10052-020-8240-z (cit. on p. 187).

[94] L. Calefice et al. "Effect of the high-level trigger for detecting long-lived particles at LHCb". In: *Frontiers in Big Data* 5 (2022). URL: https://www.frontiersin.org/articles/10.3389/fdata.2022.1008737 (cit. on pp. 188, 191, 194).

[95] A. Blance et al. "Novel B-decay signatures of light scalars at high energy facilities". In: *Phys. Rev. D* 100 (11 Dec. 2019), p. 115015. URL: https://link.aps.org/doi/10.1103/PhysRevD.100.115015 (cit. on p. 191).

[96] L. M. Garcıa Martın et al. "Radiative b-baryon decays to measure the photon and b-baryon polarization". In: *Eur. Phys. J. C* 79.7 (2019), p. 634. arXiv: 1902.04870 [hep-ph] (cit. on pp. 194, 228).

[97] X. Zhu. "K^0_S, Λ and Ξ production at intermediate to high p_T from Au+Au collisions at $\sqrt{s_{NN}}$= 39, 11.5 and 7.7 GeV". In: *Open Physics* 10.6 (2012), pp. 1345–1348. URL: https://doi.org/10.2478/s11534-012-0102-3 (cit. on p. 194).

[98] C. Bobeth and A. J. Buras. "Leptoquarks meet ε'/ε and rare Kaon processes". In: *JHEP* 02 (2018), p. 101. arXiv: 1712.01295 [hep-ph] (cit. on p. 194).

[99] A. A. Alves Junior et al. "Prospects for measurements with strange hadrons at LHCb". In: *Journal of High Energy Physics* 2019.5 (May 2019). URL: https://doi.org/10.1007%2Fjhep05%282019%29048 (cit. on pp. 195, 199).

[100] R. Aaij et al. "Search for the radiative $\Xi_b^- \to \Xi\gamma$ decay". In: *Journal of High Energy Physics* 2022.1 (2022), p. 69. URL: https://doi.org/10.1007/JHEP01(2022)069 (cit. on p. 197).

[101] R. L. Workman et al. "Review of Particle Physics". In: *PTEP* 2022 (2022), p. 083C01 (cit. on pp. 197, 204, 205, 253).

[102] U. Nierste and S. Schacht. "CP violation in $D^0 \to K_S K_S$". In: *Phys. Rev. D* 92 (5 Sept. 2015), p. 054036 (cit. on p. 198).

[103] F. J. Botella et al. "On the search for the electric dipole moment of strange and charm baryons at LHC". In: *Eur. Phys. J. C* 77.3 (2017), p. 181. arXiv: 1612.06769 [hep-ex] (cit. on p. 198).

[104] T. A. Heim and G. Morpurgo. "Gell-Mann's Eightfold Way - A History of a Failure". In: *ArXiv preprint arXiv:0708.4044* (2007) (cit. on p. 202).

[105] V. Chernyak and A. Zhitnitsky. "Exclusive processes in quantum chromodynamics: Evolution equations for hadronic wave functions and the form factors of mesons". In: *Nuclear Physics B* 345.1 (1990), pp. 137–174 (cit. on p. 202).

[106] S. Godfrey and N. Isgur. "Mesons in a relativized quark model with chromodynamics". In: *Physical Review D* 32.1 (1985), p. 189 (cit. on p. 202).

[107] H. Georgi. "Heavy quark effective field theory". In: *Physics Letters B* 240.3-4 (1990), pp. 447–450 (cit. on p. 202).

[108] Y. S. Amhis et al. "Averages of b-hadron, c-hadron, and τ-lepton properties as of 2018". In: (2019). arXiv: 1909.12524 [hep-ex] (cit. on pp. 204, 253).

[109] R. Aaij et al. "Precise measurement of the f_s/f_d ratio of fragmentation fractions and of B^0_s decay branching fractions". In: *Phys. Rev. D* 104.3 (2021), p. 032005. arXiv: 2103.06810 [hep-ex] (cit. on p. 205).

[110] M. Calvo Gomez, M. Chefdeville, Y. Hou, A. Oyanguren Campos, I. Sanderswood, J. Zhuo. "Measurement of the branching fraction ratio between $B^0 \to K^{*0}\gamma$ and $B^0_s \to \phi\gamma$ decays using the full LHCb dataset." In: (2023) (cit. on pp. 208, 217, 219, 222–224, 226).

[111] R. Aaij et al. "First Observation of the Radiative Decay $\Lambda^0_b \to \Lambda\gamma$". In: *Phys. Rev. Lett.* 123.3 (2019), p. 031801. arXiv: 1904.06697 [hep-ex] (cit. on p. 217).

[112] L. Breiman et al. *Classification and regression trees*. Belmont, California, USA: Wadsworth international group, 1984 (cit. on p. 217).

[113] T. Chen and C. Guestrin. "Xgboost: A scalable tree boosting system". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), pp. 785–794 (cit. on p. 218).

[114] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (cit. on p. 218).

[115] A. Powell et al. "Particle identification at LHCb". In: *PoS* ICHEP2010 (2010). LHCb-PROC-2011-008, p. 020 (cit. on p. 221).

[116] M. Calvo Gomez et al. *A tool for γ/π^0 separation at high energies*. Geneva, Aug. 2015. URL: https://cds.cern.ch/record/2042173 (cit. on p. 221).

[117] J. Zhu, Z.-T. Wei, and H.-W. Ke. "Semileptonic and nonleptonic weak decays of Λ_b^0". In: *Phys. Rev.* D99.5 (2019), p. 054020. arXiv: 1803.01297 [hep-ph] (cit. on p. 225).

[118] M. Ablikim et al. "Polarization and Entanglement in Baryon-Antibaryon Pair Production in Electron-Positron Annihilation". In: *Nature Phys.* 15 (2019), pp. 631–634. arXiv: 1808.08917 [hep-ex] (cit. on p. 228).

[119] G. Hiller and A. Kagan. "Probing for new physics in polarized Λ_b^0 decays at the Z". In: *Phys. Rev.* D65 (2002), p. 074038. arXiv: hep-ph/0108074 [hep-ph] (cit. on p. 231).