MSc Research Project Reading Notes

#Is it worth it/necessary to include a glossary of “unfamiliar” terms?

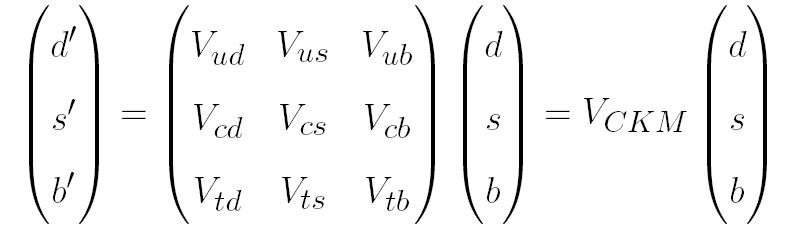
***Chapter 1: Background and Motivation***

**The Standard Model of Particle Physics #EMPHASISE WEAK DECAYS, AXIONS AND AXION LIKE PARTICLES (ALPS)**

* Gauge bosons (force mediators), and fermions (make up all matter)
* Fermions -> Quarks + Leptons
* Four fundamental forces: strong, weak, EM, and gravity, each of which are mediated by the bosons described above(except for gravity)

**Weak Interactions**

The weak interaction, unlike its strong and electromagnetic counterparts, does not possess an associated “charge” (where the strong and EM forces possess colour, and electric charge respectively). There are two types of interactions which are mediated by the charged W bosons, and the neutral Z boson, named charged, and neutral current interactions respectively. Flavour is not conserved in charged weak interactions involving quarks due to the principle of quark confinement (i.e. quarks are always found in bound states). Quarks are ‘skewed’ for weak interactions (i.e. the weak force couples the following pairs):



Here, $d’,s’ and b’$ are linear combinations of $d, s$, and $b$, and $V\_{CKM}$ represents the coupling strengths between the quarks ($V\_{ud}$ represents the coupling of $u$ to $d$ (i.e.$d\rightarrow u + W^{-}$). The elements of the CKM matrix have been experimentally measured and are given by:

**Electroweak Penguin Decays (Egede et al.)**

The most common decays of $b$ hadrons take place at the quark level through the decay of the $b$ quark via the emission of a virtual W boson. The $b$ quark is unable to decay directly to an $s$ quark as this would require a vertex with a neutral vector boson and a change of flavour. Such a process is known as a *Flavour Changing Neutral Current (FCNC)* process and is forbidden in the SM at tree level. However, at one-loop level, the FCNC quark level process like $b\rightarrow s\gamma$ is permitted. Examples of such decays include $B^{0}\rightarrow K^{\*0}\gamma$. Such decays which consist of either a lepton or a hard photon in the final state are referred to as **electroweak penguin decays.**

Electroweak penguin decays can act as a discovery mode for New Physics (NP). At either tree level or at loop level, there can be particles such as new vector bosons or leptoquarks that mediate the decays. The influence of these particles can be observed as differences with respect to the Standard Model for these decays. The theoretical calculation of the electroweak penguin decays employs an OPE (Operator Product Expansion). By analogy with the Fermi theory of weak decays, one exploits the fact that these decays are only sensitive to the spin, parity and CP properties of couplings involving particles at masses well below the b-hadron mass. For a radiative decay process, such as

Here, the operators $\mathcal{O}\_{i}$ encode the low energy behaviour and the complex valued Wilson coefficients $C\_{i}$ characterise how these different operators contribute to the overall process. Any new physics will manifest itself through Wilson coefficients that have different values form those expected in the SM or through Wilson coefficients that correspond to completely new operators such as scalar, pseudoscalar or tensor currents. The contributions to the Wilson coefficients from NP depend on both the coupling constants between the NP particles and the SM particles, as well as the masses of the NP particles. The study of electroweak penguin decays cannot therefore precisely determine the mass of any NP particles.

**Strong CP Problem**

To motivate the strong CP problem, one must understand the nature of transformations of systems under charge conjugation, C (which refers to the interchange of particles with their corresponding antiparticles), and parity, P, an inversion of spatial coordinates within a system. Parity is evidently violated in the weak theory is: the left-handed fields couple differently from the right-handed fields. It is manifest in nuclear $\beta$-decay which always produces left-handed electrons. It is also known that CP invariance (i.e. the invariance of a system under the combination of the aforementioned transformations) is violated by rare processes involving hadrons. This phenomenon is referred to as weak CP violation. There exists another form of CP violation known as **strong CP violation** which has been predicted by theory, but has not been experimentally observed. The absence of such an observation is known as the **strong CP problem.**

The QCD Lagrangian (subject to renormalisation) is given by:

Here, $q$ represents the quark fields, and F\_{\mu\nu} represents the QCD field strength. Here, $m$ represents the mass of the quarks involved, and $g$ represents the coupling strength of the strong interaction. At high energy, the short distance behaviour. On the other hand, long distance physics is hard for the strong force. For three light quarks (e.g. u, d, s), the aforementioned Lagrangian possesses a U(3) x U(3) symmetry at the classical level under which:

given that the masses of the quarks are neglected. The consideration of the mass would break this symmetry. SU(3) x SU(3) is spontaneously broken by

Here, the left hand side represents the **order parameter,** $c$ is a constant, $\Lambda$ represents a QCD scale and $\delta\_{ff’}$ is the Kronecker delta. The above relation leads to an unbroken symmetry in the case where $U = V$ (i.e. $SU(3)\_{V}$). This leads to 9 broken generators (and therefore 9 Goldstone bosons, namely the charged and neutral pions and kaons, the anti-kaon, and the eta. The ninth boson appears to be missing, the only candidate for which at this stage is the $\eta’$ (with a mass of 958 MeV). For a relativistic field theory for scalars, the relevant mass parameter is $m^{2}$ (i.e. the mass of the $\eta’$ is large).

Something that could account for this is that there is a current which does not associate itself with a Goldstone boson. Mathematically:

Classically:

This term vanishes in the limit that $m\rightarrow 0$ (i.e the current is conserved in the limit that the quark masses tend to 0). However, there exists an extra term in the current, known as the **anomaly** such that

Here, $F\_{\mu\nu}$ represents the QCD field strength and $\tilde{F} = \frac{1}{2}\epsilon\_{\mu\nu\rho\sigma}F^{\rho\sigma}$. F and its dual are related by an exchange of $\vec{E}$ and $\vec{B}$ (electric and magnetic fields).

An additional term containing depending on an angle $\theta$ can be added to the QCD Lagrangian, thereby leading to:

**ELABORATE MORE ON THIS?**

**Axions**

Axions were postulated by Peccei and Quinn in 1977 as a solution to the Strong CP problem of quantum chromodynamics. They obey a well-defined relation between their mass and coupling to SM particles. Dropping the mass-coupling relation, any scalar or pseudoscalar particle with similar couplings to the SM particles may be referred to as an axion-like particle (ALP).

**Axion Like Particles (ALPs)**

Unlike axions, which are expected to be within the sub-eV regime, and hence very light, the masses of ALPs are not as constrained, meaning that these can be arbitrarily heavy. ALPs will always couple to photons and can hence be produced in photon-photon collisions. They could also couple to gluons, Z bosons, and Higgs bosons, as well as fermions (i.e. leptons and quarks). The interactions of ALPs with these SM particles preserves a global shift symmetry $a\rightarrow a+c$, where $a$ represents the ALP field, and $c$ is a constant. In general, ALPs arise as pseudo Nambu-Goldstone bosons in various predictions associated with theories with spontaneously broken symmetries. ALPs could resolve several outstanding issues relating to the naturalness of SM parameters, such as the aforementioned strong CP problem, as well as the hierarchy problem, and can also serve as mediators to dark sector. As such, there is growing interest in the search and phenomenological study of these particles.

At colliders, pseudoscalars can be produced in rare meson decays. Meson decays via flavour changing neutral currents (FCNC) are suppressed in the Standard Model but can be strongly enhanced if the ALP is resonantly produced. Additionally, constraints on the ALP masses due to their couplings to W +- bosons have not been explored in significant detail. Such constraints on the masses of ALPs only arise due to previous experimental searches, and not theoretical impositions. As such, this coupling can be exploited to obtain a pathway to more powerful discovery modes.

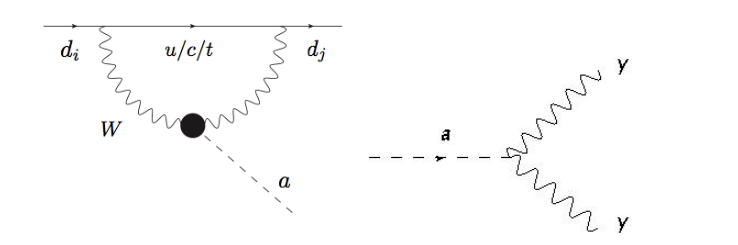
One can consider a minimal ALP (a) model with coupling $g\_{aW}$ to the SU(2)\_{W} gauge-boson field strengths, $W^{b}\_{\mu\nu}$, and Lagrangian:

$\tilde{W}^{b\mu\nu}$ is the dual field-strength tensor. This coupling leads to the production of ALPs at one-loop level in the process $B\rightarrow K^{\*}a$, where the ALP is emitted from an internal $W^{\pm}$ boson. Electroweak symmetry breaking and the resulting gauge boson mixing generates an ALP coupling to a pair of photons, and the branching fraction for $a\rightarrow\gamma\gamma$ is almost 100% in this model for $m\_{a} < m\_{W}$. The same ALP production and decay modes also occur in models with axion coupling to gluons.

**$B\rightarrow K\*A, A\rightarrow\gamma\gamma$ Decay Process**

For the purposes of this decay, one can consider the model described in **(Tongyan, Lin, Shuve paper)**, and thereby exploit the coupling of ALPs to W+- bosons, which can give rise to observable signatures. This ALP model possesses a zero coupling with gluons. The effective Lagrangian can be written as:

Here, the $g\_{aW}$ represents the coupling between the ALP field $a$ and the electroweak gauge boson field W ($\tilde{W}^{\mu\nu} = \epsilon^{\mu\nu\alpha\beta}W\_{\alpha\beta}/2$)



The above diagram represents a Flavour Changing Neutral Current (FCNC) process. Such a process is forbidden at tree level, but is permitted at loop level (as illustrated in the diagram on the left).

Considering the nature of the coupling of the ALPs to pairs of gauge bosons (specifically photons), the following decay channels were deemed to be promising for the detection of ALPs at the LHCb:

***Chapter 2: The LHCb Detector***

The LHCb experiment is dedicated to heavy-flavour physics at the LHC. It seeks indirect evidence of new physics in CP violation and rare decays of beauty and charm hadrons.

The extent of CP violation in the Standard Model weak interaction cannot explain the amount of matter in the universe. As a result, a new source of CP violation beyond the Standard Model is needed to solve the puzzle. The effect of such a source may be evident in heavy flavour physics. The LHCb intends to study the phenomenon of CP violation by analysing the higher statistics and various decay modes of the $B\_{d}$, $B\_{s}$ and $D$ mesons

The LHCb has a $b\bar{b}$ production cross-section of approximately 500 microbarn at an energy of 13 TeV, and is therefore the most abundant source of B mesons in the world. It must therefore be able to exploit the large number of $b$ hadrons. As a result, its trigger must be sensitive to a variety of final states. Displaced vertex and high transverse momentum signatures are exploited in order to separate the decays of interest from the background. The momentum and invariant mass resolution of the detector is also required to be high in order to mitigate effects from the combinatorial background. Additionally, the identification of charged particles is essential in flavour physics experiments, in order to, for instance, isolate suppressed decays and for b-quark flavour tagging. The ability to detect photons and charged particles permits the reconstruction of radiative decays, as well as more common ones such as those containing $\pi^{0}$ and $\eta$ mesons in the final state.

Data taking at the LHCb detector is divided into fills and runs. A fill is a single period of collisions delimited by the announcement of stable beam conditions and the dumping of the beam by the LHCb; a phase which typically lasts approximately 12 hours. A fill is further divided into runs, each of hich lasts a maximum of one hour.In order to benefit from the high event rate at the LHCb, a high-bandwidth data acquisition system and a robust, and selective trigger system are essential. This chapter describes the key components of the experimental setup, as well as the processes relating to the acquisition and analysis of data from this experiment.

**Structure of the LHCb Detector**

The LHCb is a single arm spectrometer with a forward angular coverage from approximately 10 mrad to 300 mrad in the bending (non-bending) plane and covering the pseudorapidity range between 2 and 5. The detector is designed such that the b and $\bar{b}$ hadrons are produced in the same forward or backward cone at high energies, and consists of a high-precision tracking system consisting of the following components:

* **Silicon-strip vertex detector (VELO)**
* **Silicon strip detector (TT) detector**
* **Dipole magnet with a bending power of 4 Tm, located downstream from TT detectors**
* **Straw drift tubes**

These components are collectively referred to as T-stations. The tracking system provides a measurement of the momentum of charged particles with a relative uncertainty ranging between 0.5% to 1.0% at 200 GeV/c. The other key components that are responsible for measuring the properties of the particles that traverse the detector include the two ring imaging Cherenkov (RICH) detectors, and a calorimeter system comprising of a scintillating-pad (SPD) and preshower detectors (PS), as well as electromagnetic and hadronic calorimeters (ECAL and HCAL respectively). A system responsible for detecting muons is also present as part of the experimental setup. Each of the components are described in further detail below

**Vertex Locator (VELO)**

The VELO is a silicon microstrip detector that surrounds the proton-proton interaction region in the LHCb experiment. It provides measurements of track coordinates that enable the identification of the primary interaction and secondary vertices, which are characteristic of b and c-hadron decays. The VELO was designed to optimise the following aspects of the LHCb experiment:

* **Angular coverage:** The VELO is designed to cover the forward region such that all tracks inside the nominal LHCb acceptance of 15-300 mrad cross at least three VELO stations. In this way, the detector fully reconstructs roughly 27% of $b\bar{b}$ production for 7 TeV proton-proton centre of mass collisions, while covering just 1.8% of the solid angle. The VELO also reconstructs tracks in the forward and backward directions hich do not have momentum information, but are useful to improve the primary vertex reconstruction
* **Triggering:** The reconstruction of the primary vertex and displaced secondary decay vertex of a heavy flavour hadron in the VELO is a key ingredient of the high level trigger which reduces the event rate from a 1 MHz event rate to a few kHz
* **Efficient reconstruction:** LHCb has studied decay modes with up to six charged tracks in the final state. This relies on the highly efficient cluster reconstruction in the VELO since the track reconstruction efficiency losses are transmitted as the sixth power. The cluster reconstruction efficiency in the VELO is paramount, both for the selection of those tracks, as six measurements per track are required, and for efficient pattern recognition and fake track rejection.
* **Displaced tracks and vertices:** Excellent vertex resolution is essential to the LHCb physics programme. Most analyses rely heavily on selection cuts on the distance with which tracks approach the primary vertex (i.e. the impact parameter). The impact parameter resolution was optimised by positioning the VELO sensors as close to the LHC beam as permitted by safety consideration, having a small inter-strip pitch at the inside of the sensors, and minimising the amount of material traversed by a particle before the first measured hits in the VELO
* **Decay time:** The decay time of a particle is obtained from the measurement of its flight distance in the VELO. This is necessary for lifetime requirements and, critically, for time-dependent measurements in the rapidly oscillating $B\_{s}^{0}-B\_{s}^{0}$ bar meson system

**RICH Detectors**

The two Ring Image Cherenkov (RICH) detectors, named RICH1 and RICH2 are responsible for the identification of charged hadrons (e.g. pions, kaons, and protons). RICH1 covers the low and intermediate momentum region (i.e. 2-40 GeV/c) over the full spectrometer angular acceptance of 25-300 mrad, while RICH2 covers the high-momentum region 15-100 GeV/c over the angular range 15-120 mrad.

In order to limit its overall volume, RICH 1 is placed as close as possible to the interaction region, and is located immediately downstream of the VELO described above. RICH 2 on the other hand, is placed downstream of the magnet, since the high momentum tracks that it measures are less influenced by the magnetic field. Both detectors consist of entrance and exit windows that serve as links to the VELO and magnet respectively. These windows form a foam sandwich construction, and are skinned with carbon fibre and aluminium respectively. **(LINK TO RICH PAPER FOR MORE INFORMATION)**

**Calorimeters**

The calorimeters are assembled in to two halves (A and C sides) which can be moved out horizontally for assembly and maintenance purposes, as well as to provide access to the beam pipe. The SPD, PS, ECAL and HCAL are segmented in the plane perpendicular to the beam axis into square cells.

At the LHC, the density of the particles incident on the calorimeters varies by two orders of magnitude over the calorimeter surface, the inner part being subject to the highest density. The calorimeters are divided into regions with cells of different sizes in accordance with these variations. There are three regions for the SPD, PS, and ECAL, and two regions for the HCAL

All four sub-detectors (SPD, PS, ECAL and HCAL) consist of successions of lead, or iron absorbers, and scintillator plates. The scintillation light resulting from the showers of the particles going through the detector is collected by the wavelength-shifting (WLS) fibres. The fibres transport the light to photomultiplier tubes (PMT) or multi-anode photomultiplier tubes (MA-PMT) that convert the light into electrical signals, which are readout by electronic Front-End-Boards (FEB).

Each cell of the ECAL and HCAL is equipped with a PMT and the corresponding FEB converts the signal to an energy value E, in order to obtain a constant dynamic range in transverse energy over the calorimeter plane. The transverse energy, $E\_{T}$ is defined as $E\_{T} = E\sin(\theta)$ where $\theta$ is the angle between the beam axis and a line from the interaction point to the centre of the cell.

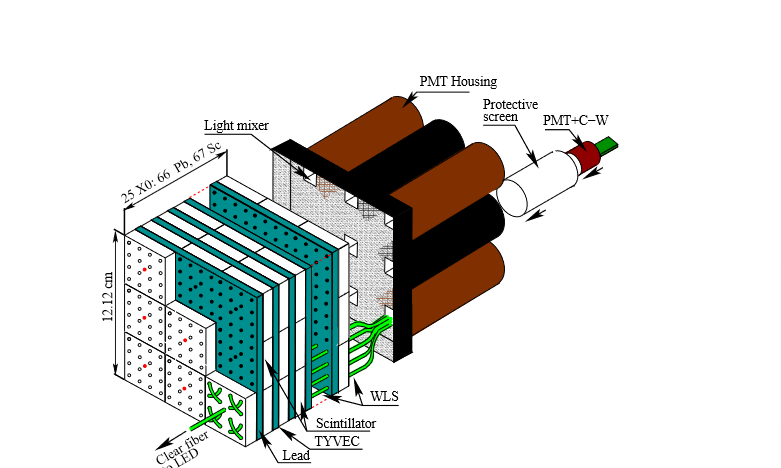
The SPD and PS provide information for particle identification of electrons and photons, which are used by the L0 trigger. The PS information is used to separate electrons, photons, and pions, while the SPD measurements contribute to the separation of neutral particles from charged ones. Additionally, the overall SPD hit multiplicity provides an estimation of the charged particles multiplicity in each beam crossing. This information can be used to veto complex events that would be difficult to analyse offline

The ECAL measures the transverse energy of electrons, photons and neutral pions for the L0 trigger, which is relevant to select B decays with an electron or photon in the final state The offline reconstruction and energy computation of neutral pions, electrons and photons also make use of the information from the PS detector. The HCAL provides a measurement of the transverse energy of hadrons for the L0 trigger in order to select a large variety of D and B decays with a charged hadron (e.g. a kaon, pion, or proton) in the final state. The SPD, PS, ECAL and HCAL are all used for offline particle identification

**Calorimeter L0 Trigger**

The calorimeter part of the L0-trigger computes the transverse energy deposited in clusters of 2 x 2 cells of the same size (or equivalently of the same region). Three different types of clusters or “trigger candidates” are built up by the L0-Calorimeter. The information from the different calorimeter sub-detectors allows to build through a trigger validation system. The first type is a “hadron” candidate, which is a HCAL cluster, with the transverse energy equal to:

The second type is the “photon candidate” which is an ECAL cluster with 1 or 2 PS cells hit in front of the ECAL cluster with no hit in the SPD cells corresponding to the PS ones. In the central zone of the ECAL, due to the higher multiplicity, an ECAL cluster with 3 or 4 PS cells hit in front of it is also defined as a photon, to increase the trigger efficiency.

The third type of candidate is the “electron candidate” which is built with the same requirements concerning the PS hits as the photon candidate requiring in addition at least one SPD cell hit in front of the PS ones. The transverse energy of the photon and electron candidates is the transverse energy of the ECAL cluster. The transverse energy of these candidates is compared to a fixed threshold (3.5 GeV for hadrons, and 2.5 GeV for electrons and photons, in Run 1). Only events containing at least one candidate above these thresholds are sent to the software trigger for further processing. 

**SPD and PS**

The SPD and PS are walls of scintillator pads (cells) with a WLS fibre coil grooved inside for better light collection. The two walls are separated by a lead curtain with a thickness corresponding to 2.5 radiation lengths. The SPD and PS have in total 6016 cells each. Their readout is multi-anode photomultipliers (MA-PMT) of type “Hamamatsu 5900 M64”.

The SPD delivers a binary information per cell, depending on the comparison of the energy deposited in the cell with a threshold. This is used to distinguish charged particles from neutral ones and, in association with the energy measured in the corresponding PS cells, helps photon and electron identification

**ECAL**

The ECAL thickness is of 25 radiation lengths to ensure the full containment of the high energy electromagnetic showers and to get an optimal energy resolution. The ECAL cells have a ‘shashlik’ structure, with scintillator (4 mm), and lead (2 mm) layers. The scintillation light readout is performed by Hamamatsu R7899-20 photomultipliers. The total number of cells is 6016

Three regions have been labelled in the diagram below (inner, middle and outer), depending on the distance of the cells to the beam-pipe. The cell size is such that the SPD-PS-ECAL system is projective, as seen from the interaction point. In addition, the outer dimensions of the ECAL match projectively with those of the tracking system $\theta\_{x} < 300 mrad$ and $\theta\_{y} < 250 mrad$ while the inner angular acceptance of ECAL is limited to $\theta\_{x,y} > 25 mrad$ around the beam pipe, where $\theta\_{x}$ and $\theta\_{y}$ are the polar angles in the LHCb frame in which (x,y) is the plane perpendicular to the beam axis. The ECAL front surface is located approximately 12.5 m from the interaction point.  The energy resolution of the ECAL for a given cell, measured with test-beam electrons is parameterised as:

Here, $E$ is the particle energy in GeV, $\theta$ is the angle between the beam axis and a line from the LHCb interaction point and the centre of the ECAL cell. The second contribution is the constant term (corresponding to mis-calibration, non-linearities, leakage, etc.) while the third one is due to the noise of the electronics, which is evaluated on average to 1.2 **ADC counts**

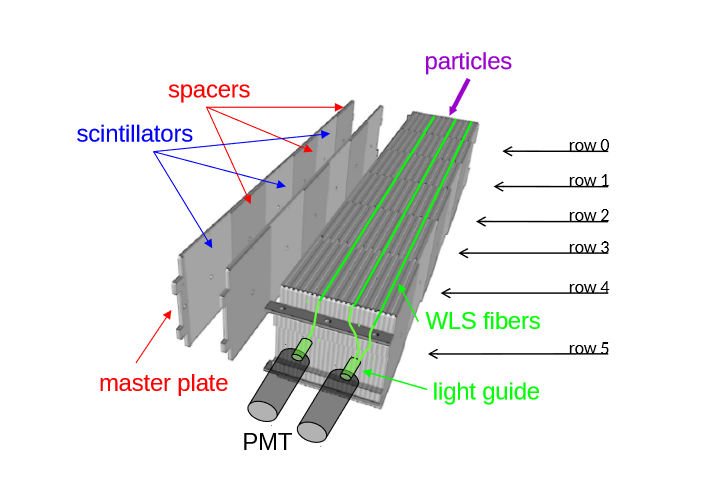
**HCAL**

The HCAL thickness is 5.6 interaction lengths due to space limitations. A sampling structure was chosen, made from iron and scintillating tiles, as absorber and active material, respectively. The special feature of this sampling structure is the orientation of the scintillating tiles that are placed parallel to the beam axis, thereby enhancing the light collection compared to a perpendicular orientation of the scintillating tiles. The same photomultiplier type as in ECAL (Hamamatsu R7899-20) is used for the readout. The HCAL has in total 1488 cells all of the same dimension located in two regions (inner and outer), depending on the distance to the beam pipe.

The energy resolution, measured in test beams, with pions is:

EE=675%E92%

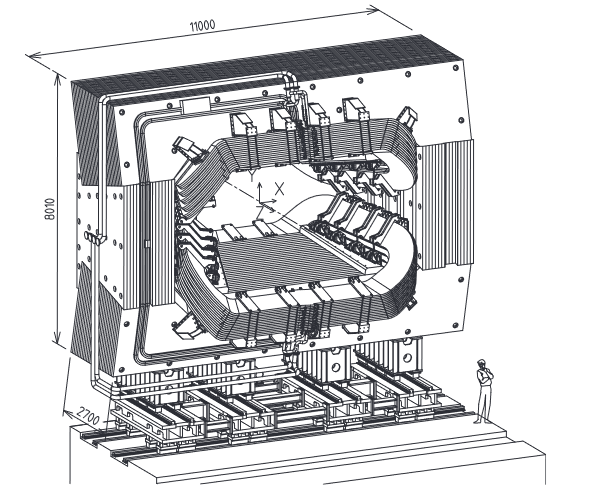
Here, $E$ is the deposited energy in GeV



**Magnet**

The LHCb dipole magnet is a **warm magnet** (i.e. it does not require cryogenic cooling) that consists of two trapezoidal coils bent at 45 degrees on the two transverse sides, arranged inside an iron yoke of window-frame configuration. It is responsible for the production of magnetic fields for momentum measurements of charged particles. The measurements cover the forward acceptance of $\pm 250$ mrad vertically and of $\pm 300 $ mrad horizontally. The magnet consists of saddle-shaped coils in a window-frame yoke with sloping poles in order to match the required detector acceptance, and has an integrated field of 4 Tm for tracks of 10 m length. This choice has been made to accommodate for the contrasting needs for a field level inside the RICHs envelope less than 2mT and a field as high as possible in the regions between the vertex locator and the Trigger Tracker tracking station. The external (top and bottom) parts of the magnet yoke were made up of laminated low carbon steel plates, of thickness 100 mm, and a maximum weight of 25 tons The magnet yoke weighs 1500 tons and contains two identical coils of conical saddle shape, weighing 54 tons, that are placed symmetrically to each other. Each coil consists of 15 pancakes arranged in five triplets and produced of pure Al-99.7 hollow conductor in an annealed state with a central cooling tunnel of 25 mm diameter. Cast aluminium clamps are used to hold together the triplets making up the coils, and to support and centre the coils with respect to the measured mechanical axis of the iron poles with tolerances of several millimeters.

The magnet is operated via the Magnet Control System, which controls the power supply and monitors a number of operational parameters (e.g. temperatures, voltages, water flow, mechanical movements, etc.). A second, fully independent system, the Magnet Safety System (MSS) ensures the safe operation and acts autonomously by enforcing a discharge of the magnet if critical parameters are outside the operating range



***Chapter 3”: Overview of Software and Modules Used in Analysis***

**The LHCb Data Flow**

The LHCb is provided with approximately 40 million proton-proton collisions by the LHC every second. If all of this data were to be stored, one would be recording approximately 1 TB of data every second (a large amount of data). The data needs to be filtered in order to keep only the events that contain something interesting. The LHCb data flow proceeds as follows:

1. Data from the detector are filtered through the *trigger*, which consists of the L0 (implemented in hardware), and the high-level trigger (HLT) implemented in software, using an application known as Moore.
2. Triggered, raw data are reconstructed to transform the detector hits into objects such as tracks and clusters, which is achieved by the Brunel application. The objects are stored in an output file in a ‘DST’ format
3. The reconstructed DST files are suitable for analysis, but are not accessible to users due to computing restrictions. Data are filtered further through a set of selections called the *stripping*, which is controlled by an application called DaVinci, which output data either in DST or micro-DST format.

**The Simulation Framework**

A large number of simulated events, known as MC data, are often produced. These are processed in a similar way to real data. There are two simulation steps which replace the proton-proton collisions and the detector response:

1. The simulation of proton-proton collisions, and the hadronization and decay of the resulting particles, are controlled by the Gauss application, which is responsible for calling the various Monte Carlo generators that are supported such as Pythia (LHCb default) and POWHEG, and for controlling EvtGen and Geant4. EvtGen describes the decay of simulated particles, whilst Geant4 is used to simulate the propagation and interaction of particles through and with the detector
2. The simulated hits made in the virtual detector are converted to signals that mimic the real detector by the Boole application. The output of this is designed to closely to match the output of the real detector, and so the simulated data can be passed through the usual data processing flow as described

In a majority of studies, only the DaVinci application is run by users, with the other applications being run centrally either on the computing farm next to the detector or on the Grid. The dependencies in the simulation framework described above are represented diagrammatically below

Diagram

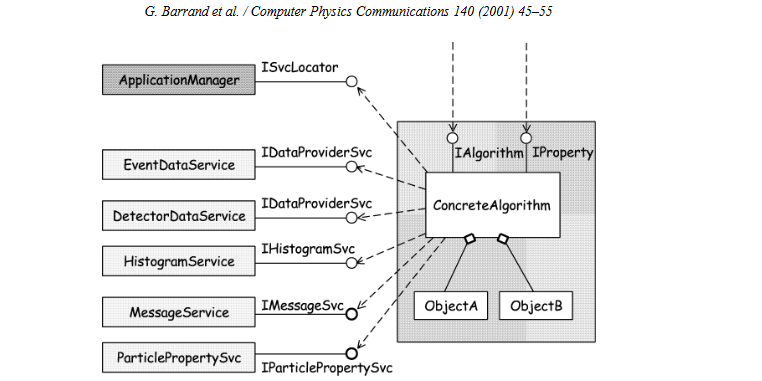
Description automatically generated

**Moore #DO I NEED TO DESCRIBE THIS?**

**Boole #DO I NEED TO DESCRIBE THIS?**

**Gaudi**

The main components of the Gaudi software architecture are shown in the below *object diagrams*, which provide a visual description of the decomposition of a system into its constituents and represent a hypothetical snapshot of the state of the system, in addition to indicating relationships between the objects and their relationships in terms of ownership and usage. They **do not** illustrate the structure (i.e. class hierarchy) of the software:



Diagram

Description automatically generated

**Algorithms and Application Manager**

The event data processing applications are predominantly based around the physics algorithms, which are encapsulated into a set of components known as algorithms. These implement a standard set of generic interfaces and can be called without knowing their functionality. The application manager is placed atop the hierarchy of algorithms, as it knows when the algorithms that are to be instantiated and called.

Algorithms are scheduled to be executed explicitly. A complex algorithm schedules the execution of the sub algorithms in the proper order to produce the desired results.

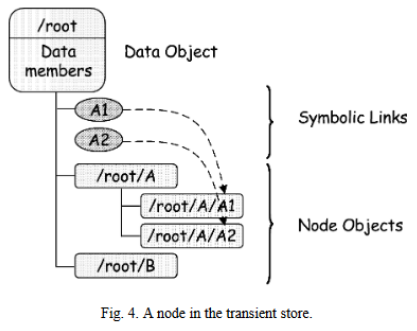
**Transient Data Stores**

The data objects needed by the abovementioned algorithms are organised in numerous transient data stores, depending on the nature of the data itself and its lifetime. The TES contains data that are valid only for the time that it takes to process one event. The Transient Detector Store contains data that describe various aspects of the behaviour of the detector (i.e. alignment) and generally have a lifetime that corresponds to the processing of many events. The Transient Histogram Store contains statistical data, which typically have a lifetime corresponding to the data processed in a complete job. Although the stores behave slightly differently, particularly with respect to the data lifetime (e.g. the event data store is cleared for each event), their implementations have many things in common and are based on a common component

A transient store enables the minimisation of the coupling between algorithm objects and data objects. An algorithm can deposit some piece of data into the transient store, and these data can be picked up later by other algorithms for further processing, without knowing how they were created.

The transient data store also serves as an intermediate buffer for any type of data conversion to another representation of the data, particularly the conversion into persistent objects or graphical objects. Hence, data can have more than one transient representation, and zero or more graphical representations.

The organisation of the data within the transient data stores is “tree-like”, similar to a Unix file system, thereby allowing data items that are logically related (e.g. Monte Carlo “truth” information), to be structured and grouped at run-time. Each node in the tree may contain either data members, or other nodes containing further groups of data members as shown in the figure below. As in a directory structure, each node is the *owner* of everything below it, and will delete all of these items when it gets deleted. In general, object-oriented data models do not map onto a tree structure, and as a result, mesh-like object associations have been implemented using symbolic links wherein the node does not acquire ownership of the referenced item



**Services**

Services are a category of components that should offer all the services directly or indirectly needed by the algorithms. This approach releases the algorithm builder from having to develop the routine software tasks that are typically needed for a physics data processing application. Examples of services are illustrated in the diagram above. These include:

* The services for managing the different transient stores (event data service, detector data service…) which should offer simplified data access to the algorithms
* The different persistency services provide the functionality needed to populate the transient data stores from persistent data or vice versa. These services require the help of specific converters which know how to convert a specific data object from its persistent representation into its transient one, or vice versa
* The job options service, message service and particle properties service

Other services, such as visualisation and event selectors are also part of the architecture. It is planned to implement many of the services using third party components.

**Data Access and Data Conversion**

There exist several options for maintaining transient and persistent data representations. One is to describe the user-data types within the persistent storage framework (meta-data) and have utilities able to automatically create both representations using this meta-data. This approach is concise and relatively easy for basic data types but is complicated when converting objects with many relationships. One can also choose to code the conversion specifically for each data type, as is done in Gaudi. A Converter, with a common interface is called whenever an object needs to be created in another representation. The converter can perform operations such as the combination of many small transient objects into a single object to minimise overhead in storage space and I/O.

Upon converting to the transient representation, the persistent representation is expanded to the individual objects. This flexibility is only possible if the code is specifically written.

The data service searches the data store for the requested objects. If the object exists, a reference is returned and the sequence ends. Otherwise, the request is forwarded to the persistency service. This service dispatches the request to the appropriate conversion service capable of handling the specified storage technology. The selected conversion service uses a set of data converters and forwards the request to the proper converter

**#ELABORATE?**

**Geant4**

Geant4 is an object-oriented simulation toolkit that has been developed to address the demand for large scale, accurate and comprehensive simulations of the particle detectors used in the experiments, owing to the escalating size, complexity and sensitivity of the detectors and fueled by the availability of moderate-cost, high-capacity computer systems on which larger and more complex simulations become possible. The toolkit provides a diverse, wide-ranging, yet cohesive set of software components which can be employed in a variety of settings.

The toolkit offers the user the ability to create a geometrical model with a large number of components of different shapes and materials, and to define “sensitive” elements that record information (hits) needed to simulate detector responses (digitisation). It also provides a comprehensive set of physics processes to model the behaviour of particles. The user is able to choose from different approaches and implementations and to modify or add to the set provided. The toolkit was required

* To have well-defined interfaces to other components
* To provide parts to be used by the other components

Other design requirements are that it is modular and flexible, and that its implementation of physics is transparent, and open to user validation. It should enable the user to comprehend, customise, and extend it in all domains. Its modular architecture should also allow the user to choose only the desired components. This leads to a hierarchical and modular structure for the toolkit as depicted in the diagram below (sub-domains are linked by a uni-directional flow of dependencies). The key domains of the simulation of the passage of particles through matter are:

* Geometry and materials
* Particle interaction in matter
* Tracking management
* Digitisation and hit management
* Event and track management
* Visualisation and visualisation framework
* User interface

These domains naturally led to the creation of class categories with coherent interfaces, and, for each category, a corresponding working group with a well-defined responsibility. It also led to the concept of a “toolkit”, which implies that a user may assemble their program at a compile time from components chosen from the toolkit or self-supplied.

Diagram

Description automatically generated

**Pythia**

The Pythia program is a standard tool for the generation of high-energy collisions, comprising a coherent set of physics models for the evolution from a few-body hard process to a complex multihadronic final state. It contains a library of hard processes and models for initial and final-state parton showers, multiple parton-parton interactions, beam remnants, string fragmentation and particle decays. It also has a set of utilities and interfaces to external fragmentation and particle decays. It also has a set of utilities and interfaces to external programs. Pythia 8 represents a complete rewrite in C++, while its predecessors were written in Fortran

Currently, the program only works with proton-protons, antiprotons-protons, electron-positrons, and antimuon-muon

The hard processes, parton showers, multiple interactions, beam remnants, and hadronization (i.e. the physics topics) that are to come together in a complete event generator can crudely be subdivided into three stages:

1. The generation of a “process” that decides the nature of the event. Often, it would be a “hard process” (gg -> h0, Z0Z0 -> two charged muons and a quark-antiquark pair) that is calculated in perturbation theory. Only a very small set of partons/particles is defined at this level, such that the main aspects of the event structure are covered
2. The generation of all subsequent activity on the partonic level, involving initial and final-state radiation, multiple parton-parton interactions and the structure of beam remnants. Much of the phenomena are under an (approximate) perturbative control., but nonperturbative physics aspects are also important. At the conclusion of this step, a realistic partonic structure has been obtained (e.g. with broadened jets and an underlying event activity)
3. The hadronization of this parton configuration by string fragmentation, followed by the decays of unstable particles. This part is almost completely nonperturbative, and so requires extensive modelling and tuning or especially for decays, parametrisations of existing data. It is only at the end of this step that realistic events are available, as they can be observed by a detector

The three subdivisions of tasks are not rigid- parton distributions span and connect the first two steps. The structure of the Pythia 8 generator is based on this subdivision. This structure is shown below

Diagram

Description automatically generated

Here, the thick arrows show the flow of commands to carry out different physics tasks, while the thinner arrows show the flow of information between the tasks. The bottom box contains common utilities that may be used anywhere

The most important piece of information is that of the event record, which is represented by the ‘Event’ class, of which there are two objects, namely process (which only covers the few partons of the “hard” process of point 1), and another called event, which covers the full story from the incoming beams to the final hadrons. A small Info class keeps track of useful, unique information, such as the kinematic variables fo the hard process

There are also two incoming BeamParticles that keep track of the partonic content left in the beams after several interactions and initial-state radiations, and rescales parton distributions accordingly

The process library, as well as parametrisations of total, elastic, and diffractive cross sections, are used both by the hard-process selection machinery and the multiple interactions one.

The Settings database keeps track of all integer, double, boolean, and string variabes that can be changed by the user to steer the performance of Pythia, except that ParticleDataTable is its own separate database.

Finally, a number of utilities can be used just about anywhere, for Lorentz four vectors, random numbers, jet finding and simple histograms, and for a number of other “minor” tasks.

Orthogonal to the subdivisions described above, there is another, more technical classification, wherein the user interaction with the generator occurs in three phases:

* Initialisation, where the tasks to be performed are specified
* Generation of individual events (the “event loop”)
* Finishing, where final statistics is made available

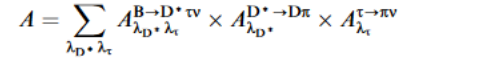
Again, the subdivision (and orthogonality) is not strict, with many utilities and tasks stretching across the borders, and with no finishing step required for many aspects. Nevertheless, as a rule, these three phases are represented by different methods inside the class of a specific physics task

**EvtGen**

The EvtGen package is an event generator designed for the simulation of the physics of B decays. This package provides a framework to handle complex sequential decays and CP violating decays. The simulation of decays proceeds using decay amplitudes, rather than probabilities. The amplitude of each node in a decay tree is used to simulate the entire decay chain, including all angular, and time-dependent correlations. The framework is designed in such a way that additional decay models can easily be implemented.

**Event Selection Algorithm**

Consider the decay and . In this case, the decay amplitude can be written as:



Where and label the states of spin degrees of freedom of the D\* and the respectively. Thus, the represents the decay amplitude for the superscripted decay mode. One can implement the above equation to generate kinematics according to phase space for the entire decay and to calculate the probability, the amplitude squared, which is used in an accept-reject algorithm. This approach has two serious limitations:

* The maximum probability of the decay chain must be known (difficult since there are numerous potential decay chains in B decays)
* For long decay chains, the accept-reject algorithm can be very inefficient, since the entire chain would need to be regenerated if the event is rejected.

Therefore, an algorithm that generates a decay chain as a sequence of sub-decays is introduced in order to avoid the abovementioned limitations. First, one can consider the decay of the B. Kinematics are generated according to phase space and the probability is computed:



The kinematics are regenerated until the event passes an accept-reject based on . After decaying the B, one forms the spin density matrix\*

Text

Description automatically generated

This describes a D\* from the decay after summing over the degrees of freedom for the . To generate the decay, proceed as with the B, including the :

Text, letter

Description automatically generated

Here, the scale factor (the first term) is proportional to the decay rate, and does not influence the angular distributions. The scale factor makes the maximum decay probability of each sub-decay independent of the full decay chain.

Finally, one can decay the , forming the density matrix:



Which encapsulates the information about the decay needed to properly decay the with the full correlations between all kinematic variables in the decay. Using the above matrix, one can compute the spin density matrix of the :



As with the other decays, kinematics are generated according to phase space and the accept-reject is based on the probability calculated as in the equation for , replacing with

**Decay Models**

The computation of the spin density matrices and the decay probability in the above example are performed by the EvtGen framework. Decay models, which implement a single node in a decay tree, must only specify the decay amplitude for each combination of mother and daughter spin states. Consider the decay:

Here, the amplitude is , where represents the polarisation of the D\* and p\_\pi is the momentum of the \pi. In this case. Three amplitudes must be provided, one for each basis state of the

Various decay models are implemented in the EvtGen package, including those for specific CP violating channels, Dalitz decay models, mixing and semileptonic form factor models. Generic models handle either decays to specific sets of spin states, such as a pseudoscalar to a vector plus pseudoscalar, or other such models. Two body CP violating decays, including scalar-scalar, scalar-vector, scalar-tensor, and vector-vector final states have been implemented. Model parameters include the value of the CKM angle and the amplitude for B or anti-B mesons to decay to the specified final state. For vector-vector final states, such as , transversity amplitudes are specified

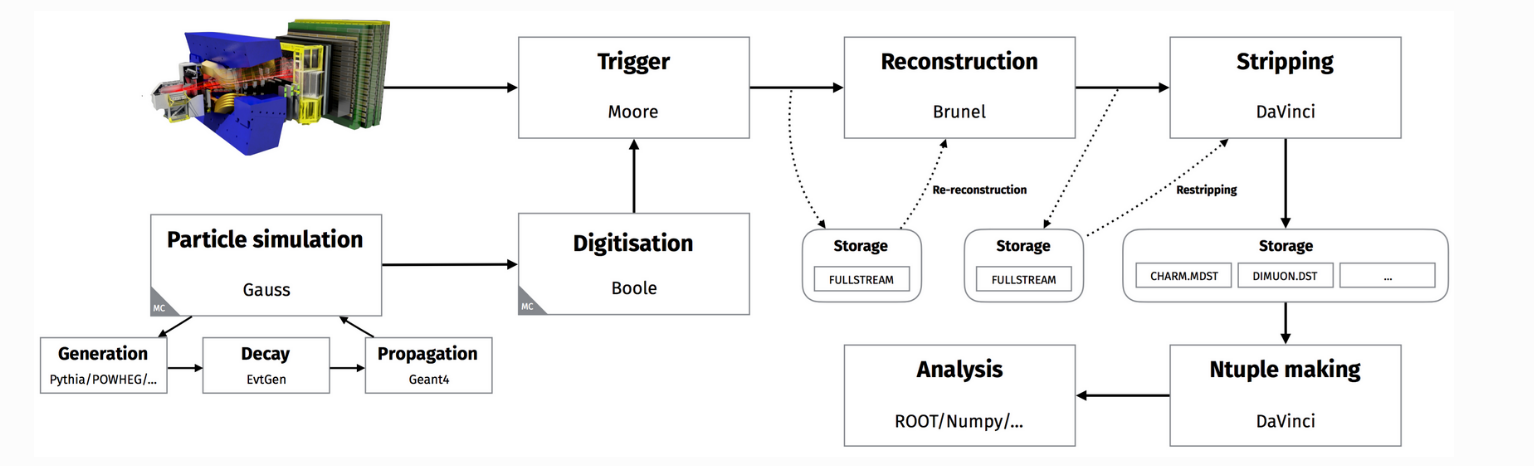
Semileptonic models include form-factor implementations for specific theoretical models, or more generic models, where the user specifies parameters for the form factors. For example, there is a model for based on HQET with parameters for and

The EvtGen generator has been designed to correctly simulate the angular and time-dependent distribution in sequential B decays. In order to implement a decay chain, the only the decay amplitude for each step in the chain is needed. This design is efficient and leads to reusable decay models.

**Gauss**

The LHCb simulation framework Gauss is built, as all the other LHCb software applications, using the general GAUDI data-processing framework. Gaudi helps in the configuration of algorithms and tools in the application. It also controls the flow of data in the event loop. The two key purposes of the Gauss framework are:

* To control the generation of collisions (in most cases, proton-proton collisions) with Pythia, where a specific LHCb configuration is used. Gauss then makes use of EvtGen in order to model the decays of unstable particles.
* Propagate generated particles through the experimental apparatus, and to simulate the physics processes within the sub-detectors using the Geant4 toolkit. The information that mimics the response of the given sub-detectors when a particle traverses through them (MC hits) is written to a file for further processing. This enables the comprehension of trigger and reconstruction performance (MC truth)



***Experimental Methods***

1. Simulated B to K\* gamma events. Plotted signal selection efficiency vs photon transverse energy for both decay modes
2. Deduced upper bound on the branching ratio of the two-photon decay mode using the relationship:

1. Determined the single event sensitivity of the single photon decay mode using the literature
2. Plotted signal selection efficiency curves and estimated upper limit on branching fraction of

**References**

* <https://www.youtube.com/watch?v=UlyGazWkeuA>