Analysis of the decay $B_c \to B_s\pi$ with LHC and ATLAS

Elina Berglund

Department of High Energy Physics
Lund University

A thesis submitted to the Faculty of Engineering (LTH),
Lund University,
for the degree of Master of Science
in the subject of Engineering Physics.

April 2008

Abstract

This thesis investigates the prospects for studying the $B_c \to B_s\pi$ decay with the ATLAS detector at the Large Hadron Collider. Due to the high luminosity and energy at the LHC, rare B hadrons such as the B_c meson will be produced in unprecedented quantities.

This work describes simulations of the B_c signal in the ATLAS detector in order to investigate whether the signal is observable or not and to what precision. The simulation results in an assessment of the lifetime and mass resolutions and the reconstruction efficiencies for signal and backgrounds. Due to high background contamination, the significance of the signal is determined to be less than 2σ at the integrated luminosity of 20 fb^{-1}. This, however, assumes a branching factor of $5 \cdot 10^{-2}$ for the $B_c \to B_s\pi$ signal. To reach the 5σ-limit a branching factor of $16 \cdot 10^{-2}$ is required, which could be the case since the branching factor for the $B_c \to B_s\pi$ decay has not yet been experimentally determined.

Acknowledgements

I would like to take the opportunity to thank the people who have supported me through my Master's thesis experience. A big thank you to my supervisor in Lund, Paula Eerola. I would also like to thank Maria Smizanska, Sergey Sivoklokov and Guennadi Borissov for help with various issues.

Most importantly, I would much like to thank James Catmore for all the unlimited and valuable assistance you have given me throughout my thesis. It would not have been possible without you.

I would also like to thank my family for always supporting me, as well as Raoul Scherwitzl and the boys at Chez les Suédois for making the stay at CERN a very enjoyable experience.

Contents

Chapter 1

Introduction

The Standard Model of Particle Physics, essentially unchanged since the 1970s, is about to undergo its most severe experimental test with the launch of the Large Hadron Collider later in the year of 2008. Will the few remaining uncertainties of this highly successful model then be answered or will a new world of questions result from the upcoming discoveries? If the Standard Model turns out to be the best that humanity can achieve in describing the fundamental Laws of Nature, our quest to understand the Universe will have a less satisfactory conclusion than is the dream of most physicists. The Standard Model is, if truth be told, lacking in the beauty of a simple Unified Theory of nature's building blocks, that humans have been searching for since the time of the ancient Greeks. With its multiple free parameters, failure to account for gravity and inability to explain the preponderance of matter over anti-matter in the modern Universe, many fundamental questions will remain forever unanswered if we cannot go beyond the Standard Model.

It is naive to think that human-kind with all its limitations, for the first time in history, has solved the fundamental questions of the Universe and has acquired complete knowledge of the physics of our surroundings. This belief has existed at various stages in the History of Science, but the truth is probably that we will never achieve complete understanding. There will always exist another theory, more fundamental, as long as we keep digging deeper into the constituents of our world. Then why even bother trying to further explore physics if the complete answers will never be known? Well, it is our natural curiosity and irrepressible desire to understand our world that gathers researchers from countries all over the globe to research institutes such as CERN, hoping to show that the Standard Model is simply an approximation of the truth and to gain more insight into the physics that lies beyond it.

This thesis merely reflects a minuscule portion of the preparation for the research that will take place at the LHC. The work behind the thesis lies within the field of B-physics, with the ambition of studying the feasibility of searching for the

$B_c \to B_s\pi$ decay in the ATLAS detector. Even at low luminosity, the $b\bar{b}$ quark pair production rate at the LHC is expected to reach as many as 10^{12} events per year. The available B-physics statistics will thus be limited only by the rate at which the data can be stored and processed. The possibilities for B-physics are therefore many, even for rare B decays involving the B_c meson.

The B_c meson displays unique features regarding both its production and decay properties, since it contains two heavy quarks of different flavor. The properties of the B_c particle can be used to investigate complex QCD models to a greater extent than has been possible before.

In the following chapter, the Standard Model of Particle Physics is discussed, with special emphasis on B-physics and physics of heavy quarks. An overview of the LHC and the ATLAS detector is then laid out. The third chapter provides a description of the methodology for accomplishing this work and of the software tools used. The fourth chapter lays out the resulting plots and calculations. Finally, the fifth chapter provides the reader with conclusions and a brief summary.

Chapter 2

Theory and Background

2.1 The Particle Physics Behind the Scenes

Although understanding the fundamental building blocks of our world is a demanding task, the curiosity of man has caused many models and theories to emerge over time. The state of the art is the Standard Model of Particle Physics and even though its predictions have been proven to be very accurate, the model has limitations. The shortcomings of the Standard Model have led physicists to believe that it is an incomplete theory; phenomena that cannot be explained within it (not yet observed but expected in the LHC) are referred to as Physics Beyond the Standard Model or simply New Physics.

The discovery of New Physics, as well as further studying properties of the Standard Model (such as the strong force), form the bulk of the LHC programme. Heavy quark dynamics are of great importance for such investigations.

2.1.1 The Standard Model of Particle Physics

The fundamental building blocks and forces of our world

The Standard Model (SM), which has gradually developed over the last 40 years, describes the elementary particles that comprise our world. The model is divided into particles of matter (quarks and leptons) and force mediating particles (gauge bosons). There are three generations of quarks and leptons, where the next generation is a more massive copy of the former (see figure 2.1). It is not known why nature is duplicating itself in this manner.

There are six leptons in the SM, which are classified according to electric charge, electron number, muon number and tau number. The electron, muon and the tau have charge -1 whilst their associated neutrinos are neutral. All leptons have corresponding antiparticles with opposite charge and leptonic numbers.

The *up*, *charm* and *top* quark have electric charge $+2/3$ whilst the *down*, *strange*

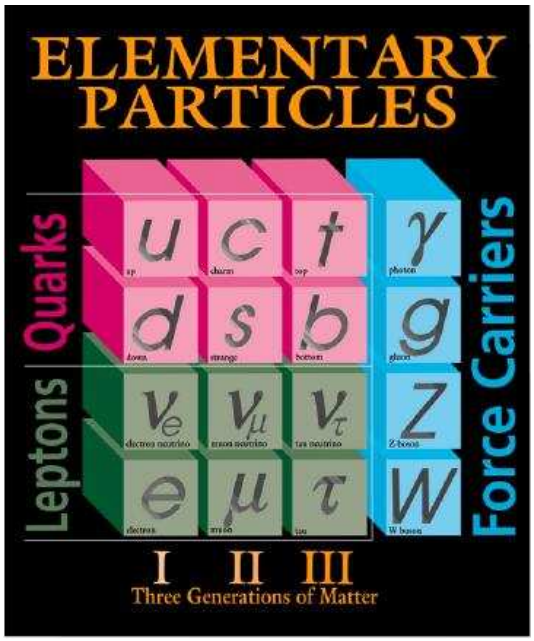

Figure 2.1: The elementary particles of the standard model [3].

and *bottom* quark have charge $-1/3$. Quarks are classified not only according to charge, but also flavor, such as strangeness (S), charm (C), beauty (B) and truth (T). All anti-quarks have opposite values of flavor and charge compared to the associated quark. Quarks and gluons posses another property that leptons are lacking: color (red (R), green (G) and blue (B)). Introducing these quantum numbers increases the number of quarks and antiquarks threefold, giving 36 in total. Quarks do not exist by themselves in nature, but exist as composites referred to as hadrons. There are two kinds of hadron; baryons and mesons. Baryons are bound states of three quarks such as the proton (udu) and the neutron (udd). Mesons consist of two quarks, one quark and one anti-quark, for instance the positive pion ($u\bar{d}$). All hadrons are colorless with the quarks adding up to a singlet representation of the color group. Color is a hidden quantum number since all detected particles are observed to be colorless.

Forces between particles are effected by mediators that are exchanged in the interaction. In the theory of quantum electrodynamics (QED) the massless and neutral photon is the carrier of the electromagnetic force, which affects all particles with electrical charge. The massive W^+, W^- and Z^0 bosons are the mediators of the weak force and the strong interactions are mediated by eight kinds of massless gluon which are described by the theory of quantum chromodynamics (QCD). The strong force affects all particles with color charge (quarks). Additionally the gluons themselves are bicolored with a combination of color and anti-color (for example

$R\bar{B}$). As a result gluons are able to interact with each other, making QCD a more complicated theory than QED, whose mediator, the photon is electrically neutral and therefore cannot self-interact.

The fourth fundamental force in our Universe is gravity and although it is the weakest, it dominates on an astronomical and cosmological scale, governing the motion and interaction of planets, stars and galaxies. Gravity is said to be mediated by the graviton, but this particle has not been observed and has not been integrated into the Standard Model.

The LHC accelerates two proton beams in opposite directions at very high energies. Although protons nominally consist of a bound state of two *up* quarks and one *down* quark (*udu*), reality is more complex. An unstable gluon 'soup' can be found surrounding the *up* and *down* quarks (also referred to as valence quarks), and the gluons continuously create quark-antiquark pairs from all of the generations. These quarks are called sea quarks and are rapidly annihilated to another gluon; however, under the extreme conditions of an LHC *pp* collision, these sea quarks can be ejected from the interaction region where they hadronize and decay into detectable particles. It is by this mechanism that exotic particles can be produced from 'mundane' protons.

Quantum Chromo-dynamics

As briefly mentioned above, Quantum Chromo Dynamics (QCD) is the part of the Standard Model describing the strong force. QCD is a more complicated theory than QED since the mediators of the force, the eight different gluons, carry color themselves and thus interact with each other. Additionally, there are three types of color while there is only one kind of electric charge.

Another special feature of the strong force is that the interaction strength actually increases with distance [1, 2]. This is described by the "running" coupling constant, α_s, which is large for long distances (characteristic of nuclear physics), where the energy is small (see figure 2.2). In the case of small distances between quarks, as for example within a proton, the coupling constant becomes relatively small, which makes the quarks and gluons behave almost like free particles. The phenomenon is called asymptotic freedom and allows perturbation theory to be applied at high energies. Perturbative QCD, which in several respects resembles QED, is a surprisingly simple model considering the complicated highly non-linear nature of QCD.

Unfortunately, there is no single model describing the strong force at all energies. Although perturbative QCD has been very successful it cannot always be applied. Bound states of quarks like the B_c meson must be described by non-perturbative models, since the coupling strength ($1/\alpha_s$) between the quarks is large [3]. One

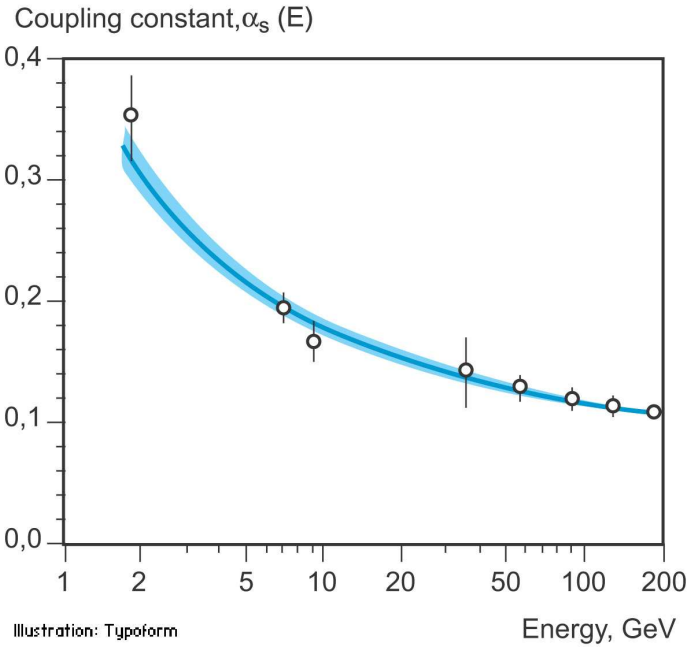

Figure 2.2: The evolution of the fundamental strong coupling constant, also called the "running" constant, as a function of energy [2].

of the more successful non-perturbative QCD approaches is called lattice QCD, a model based on a Monte Carlo simulation that uses discrete rather than continuous space-time. Lattice QCD predicted the mass of the B_c meson within a precise range of 0.3% before it was confirmed experimentally.

The inverted nature of α_s also gives rise to the color confinement property of the strong force, which forbids colored objects to exist on their own. If, for example, the distance between two quarks increases, then the strength of the force increases until conservation of energy allows a quark anti-quark pair to be created. This explains why quarks have never been observed separately, but solely in the form of bound states such as baryons and mesons. The Standard Model in general and QCD in particular are described in Ref. [4].

The Higgs Boson

The Higgs particle is a massive scalar boson predicted by the Standard Model. The Higgs mechanism permits the masses of the different elementary particles to be non-zero and different for different particles. A massive particle interacts more strongly with the Higgs field than a less massive particle. There are still several unknown factors concerning the Higgs mechanism, such as how many Higgs bosons there are and their different masses. Higgs searches are currently the subject of frenetic activity which will intensify when LHC data becomes available. Finding this particle is not an easy task due to the our lack of knowledge of the mass range

in which to search for it. According to [5], the mass is likely to lie within $(115-180)$ GeV. Hopefully the experiments at the LHC will be able to provide some answers to these questions.

Problems with the Standard Model

Even though most of the predicted values of the Standard Model have been experimentally confirmed with remarkably high precision, the theory has severe shortcomings. The undetected Higgs boson is one example together with the inability to describe gravity within the model. Another defect of the SM is the large number of "elementary" particles: 12 leptons and antileptons, 36 quarks and antiquarks and 12 mediators. Physicists often search for simple and beautiful theories containing as few unrelated parameters as possible. The SM is therefore very unsatisfactory and this has led to a broad consensus that there must be additional physics beyond the Standard Model.

There are also other questions remaining unanswered by the Standard Model. Charge-Parity (CP) violation in the SM, for instance, is not large enough to explain the matter/antimatter asymmetry in the Universe. The so-called Hierarchy Problem is left unexplained by the SM. In the Hierarchy Problem one of the important questions is why the weak scale and the Planck scale are so different, *i.e.* why gravity is so much weaker than the weak force when both forces involve constants of nature. Two of the more popular theories that could explain these issues are supersymmetry (SUSY) and string theory.

2.1.2 B-physics and the B_c meson

B-physics concerns the study of all subatomic particles which explicitly contain a b or $\bar{b}$ quark. B-hadrons display many interesting phenomena and the prodigious statistics expected from the LHC will allow further validation of the Standard Model as well as searching for New Physics. One of the key issues that can be investigated with b-flavored hadrons is the phenomenon of the breaking of the Charge-Parity symmetry [6]. Although CP violation was discovered as early as 1964 in the neutral kaon system, it still remains a high priority in experimental physics. Another interesting topic associated with B-physics are the rare b decays induced by flavor changing neutral current (FCNC) transitions $b \to d, s$. The decays are suppressed in the Standard Model and will therefore reveal New Physics effects if seen in larger-than-expected numbers.

Very recently (see [7]), the first tentative evidence (not at a good enough precision to claim a discovery) of physics beyond the Standard Model was found in the analysis of the $B_s \to J/\psi\phi$ channel made by the CDF and D0 collaborations at

Fermilab. Measurements from the LHC, with much greater statistics, will hopefully be able to confirm this evidence and claim a discovery of New Physics.

The B_c meson is a particularly interesting particle since it is the lowest bound state of two heavy quarks with different flavors, the b and the c quark [8, 9, 10]. Due to its explicit double-heavy-flavor content, it resembles the QCD dynamics of the quarkonium systems $b\bar{b}$ and $c\bar{c}$, which are approximately non-relativistic. An important difference between the B_c meson and the $b\bar{b}$ and $c\bar{c}$ states is due to the presence of both the b and the c quark in the B_c meson. Since the B_c particle carries open flavor, its ground state is stable against strong interactions and the particle can only decay through weak interactions. This causes the B_c meson to have a relatively long lifetime, enabling precise measurements on the particle. Therefore, the B_c meson provides a unique test of heavy-quark physics: it is possible to extract information about both QCD dynamics and weak interactions. Also non-perturbative QCD effects play an important role in the B_c production properties (see chapter 2.1.1).

The B_c meson was first observed in 1998 at the CDF detector at Fermilab through the semileptonic decay channel $B_c \rightarrow J/\psi l \nu_l$. With a signal significance of 4.8σ, the mass and lifetime measurements were found to be [11]:

$$\begin{aligned} M_{B_c} &= (6.04 \pm 0.39(\text{stat}) \pm 0.13(\text{syst}))\ \text{GeV/c}^2 \\ \tau_{B_c} &= (0.46^{+0.18}_{-0.16}(\text{stat}) \pm 0.036(\text{syst}))\ \text{ps} \end{aligned} \tag{2.1}$$

Updated lifetime results at Fermilab were published in 2006. The measurements were made on the decay channel $B_c^+ \rightarrow J/\psi e^+ \nu_e$ with results [12]:

$$\tau_{B_c} = (0.463^{+0.073}_{-0.065}(\text{stat}) \pm 0.036(\text{syst}))\ \text{ps} \tag{2.2}$$

More current mass measurements with the CDF experiment was published in mid 2007 through the channel $B_c \rightarrow J/\psi\pi$ with a significance greater than 8σ [9]:

$$M_{B_c} = (6274.1 \pm 3.2(\text{stat}) \pm 2.6(\text{syst}))\ \text{MeV/c}^2 \tag{2.3}$$

Decay processes and partial widths of the B_c meson

There are three dominating decay processes of the B_c^+ meson: c quark decay with the $\bar{b}$ quark as a spectator, $\bar{b}$ quark decay with the c quark as a spectator and $\bar{b}c$ annihilation decays through the channel $B_c^+ \rightarrow l^+ \nu_l (c\bar{s}, u\bar{s}), l = e, \mu, \tau$ (see figure 2.3). The approximate sharing of branching factors of the different channels are estimated to be 30%, 63% and 7% for the process in (a), (b) and (c), respectively [13]. There are several models predicting the mass and lifetime of the B_c meson and the results vary somewhat depending on the parameters used to calculate the decay contribution. More precise measurements of the B_c meson will therefore determine

the relative importance of these decay processes and decrease the uncertainties on the lifetime of the particle.

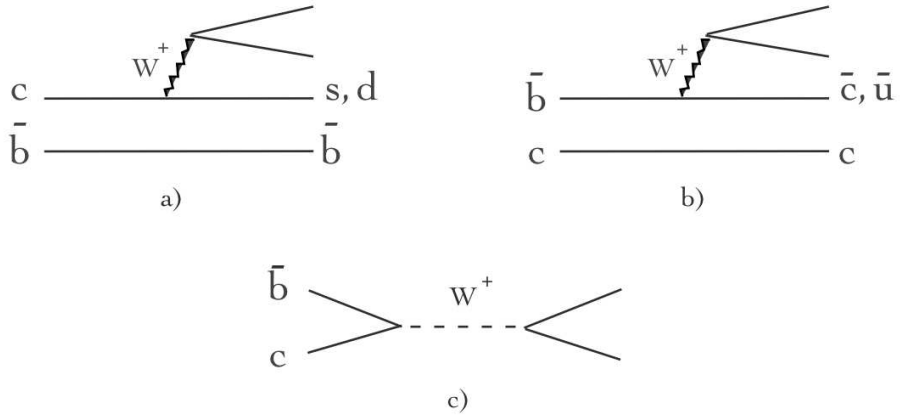

Figure 2.3: Diagrams of the three major decay processes of the B_c^+ meson where (a) is the b-spectator decay, (b) the c-spectator decay and (c) the annihilation process.

A simple nonrelativistic method of calculating the total decay width of the B_c^+ meson involves determining the sum of the three dominating decay processes of the particle [13, 14].

$$\Gamma(B_c \to X) = \Gamma(b \to X) + \Gamma(c \to X) + \Gamma(\text{annihilation}) \tag{2.4}$$

Through equation 2.4, a value of the total width can be estimated after separately calculating the partial decay widths. The expressions for the partial widths of the two spectator decay processes (Eq. 2.5 and 2.6) assumes that effects of quark binding inside the B_c meson can be neglected.

$$\Gamma(b \to X) = \frac{9G_F^2|V_{cb}|^2 m_b^5}{192\pi^3} = 8.75 \cdot 10^{-4}\,\text{eV} \tag{2.5}$$

$$\Gamma(c \to X) = \frac{5G_F^2|V_{cs}|^2 m_c^5}{192\pi^3} = 4.19 \cdot 10^{-4}\,\text{eV} \tag{2.6}$$

G_F^2 is the Fermi constant, V_{cb} and V_{cb} are the CKM matrix constants with values 0.975 and 0.044 according to [13]. The partial width of the annihilation decay channel is given by equation 2.7.

$$\begin{aligned} \Gamma(\text{annihilation}) &= \frac{G_F^2}{8\pi}|V_{bc}|^2 f_{B_c}^2 M_{bc}^5 \sum m_i^2 (1 - \frac{m_i^2}{M_{B_c}^2})^2 C_i, \\ &= 0.923 \cdot 10^{-4}\,\text{eV} \end{aligned} \tag{2.7}$$

where $C_i = 1$ for the $\tau\nu_\tau$ channel and $C_i = 3|V_{cs}|^2$ for the $\bar{c}s$ channel, m_i is the mass of the heaviest fermion. f_{B_c} is the pseudoscalar decay constant for the B_c meson (see [13] and [14] for more details).

The result of adding the three contributions gives $\Gamma(B_c \to X) = 13.863 \cdot 10^{-4}$ eV. The lifetime can then easily be calculated using $\tau = \hbar/\Gamma \approx 0.47$ ps, which agrees with the current measured value (see Eq. 2.2).

The production of B_c mesons

To produce a particle of mixed flavor at the LHC, such as the B_c^+ meson, a joint production of a $\bar{b}$ and a c quark is required in the proton-proton collision [14]. This is one reason for the relatively low production cross section of the particle (see Eq 3.2). However, almost all B_c mesons are immediately transformed into the lowest state through radiative transitions, and the lowest bound state is stable against all decays except for weak interactions, leading to a relatively long lifetime.

A B_c meson can be produced in several ways and the theory describing these processes is far from trivial. The hadronic B_c production is especially complex and requires a detailed study of a large number of Feynman diagrams. Experimental studies of the B_c meson can assist in understanding heavy quarks and QCD theories. At high energies, such as in the LHC, the gluon-gluon process, $gg \to B_c^+ + b\bar{c}$ is the dominating contribution to B_c production (see figure 2.4).

The cross section for B_c meson production at LHC energies is approximately 10^{-2} of the total cross section of producing a $b\bar{b}$ pair [14]. This can be compared to 10^{-4} of the total cross section for a hadron collider of center-of-mass energy 40 GeV, where B_c meson production is close to nonexistent. Fortunately, the prospect of $b\bar{b}$ production cross section at the LHC is quite large since approximately one percent of the collisions will produce a $b\bar{b}$ pair [10]. The LHC therefore provides a unique opportunity for studying the B_c meson with high precision. Decay modes including a J/ψ are particularly suitable since the mode $J/\psi \to \mu\mu$ can easily be triggered upon using a dimuon trigger scheme (see chapter 2.3.7 for more details).

The decay channel $B_c \to B_s(J/\psi(\mu^+\mu^-)\phi(K^+K^-))\pi$

The process of interest in this work concerns the decay of the B_c^+ meson (consisting of a $\bar{b}$ and a c quark) to a neutral B_s meson and a charged pion. The neutral B_s meson is composed of an s and a $\bar{b}$ quark. The quark content of the positive pion π^+ is $u\bar{d}$. The lowest order Feynman diagram representing this decay can be found in Figure 2.5.

In the channel studied in this work, the B_s meson then decays to the charmonium state J/ψ (consisting of a bound state of $c\bar{c}$) and a ϕ meson (with the quark contents $s\bar{s}$). These particles decay very rapidly via annihilation to a $\mu^+\mu^-$ pair

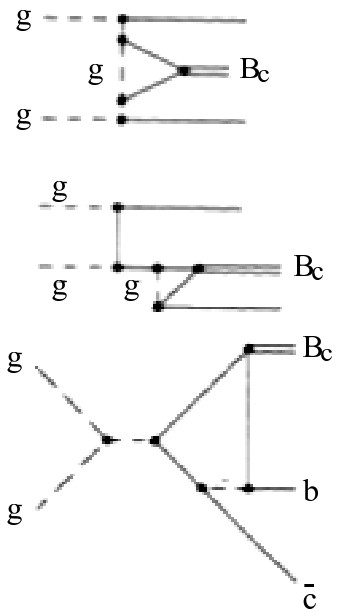

Figure 2.4: Diagrams of the single B_c meson production in gluon subprocesses [14].

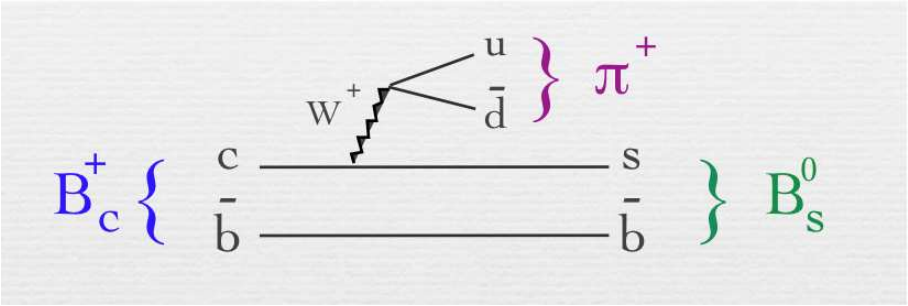

Figure 2.5: A Feynman diagram of the decay $B_c^+ \to B_s\pi^+$.

and a K^+K^- pair, respectively. The quark contents of the charged kaon is $u\bar{s}$ (for the positive particle) and $s\bar{u}$ (for the negative anti-kaon).

This particular decay channel has not been searched for in any major experiment so far. However, according to [6] the branching ratio is quite large at $[4-17]\cdot 10^{-2}$, which can be compared to the better known (and observed) channel $B_c^+ \to J/\psi\pi^+$ with a branching ratio of about $[1-2]\cdot 10^{-3}$. The latter decay process has the advantage of directly producing a J/ψ which is easily triggered upon in the detector. Cross sections and branching ratios are discussed in more detail in chapter 3.2.2.

2.2 An Overview of the Large Hadron Collider

Experimental particle physics today resembles a race where bigger is better. Bigger in this case implies larger accelerators with higher energy and beam intensities. When the LHC starts running in the summer of 2008 it will be the largest and most powerful synchrotron in the world. It will be the biggest piece of scientific apparatus ever constructed.

One of the main goals of the LHC is to find the Higgs Boson, which is the only remaining undetected particle predicted by the Standard Model. This missing piece of the puzzle is essential since the Higgs field is needed to permit massive particles in the Standard Model. Another important goal of the LHC is to create an environment that resembles the moments just after the Big Bang. The high energy of the accelerator will induce interactions that are believed to have ruled the very early Universe. The LHC will perhaps also provide some answers to the mystery of why the Universe is so heavily dominated by matter, where current theories suggest that antimatter must have been produced at the same rate as matter at the instant of the Big Bang.

Figure 2.6: Geneva area with the French Alps in the background. The (red) ring represents the location of the underground LHC tunnel [15].

2.2.1 Some facts and figures

The LHC is indeed a large hadron collider. It contains two counter-rotating proton beams, each having an average energy per proton of 7 TeV, leading to a mean center-of-mass energy of 14 TeV per collision. This energy can be compared to the particle accelerator with the current highest energy in the world - the Tevatron at Fermilab. The Tevatron collides protons with antiprotons at a maximum energy of approximately 1 TeV (hence the name). To achieve the impressive high energies of the Large Hadron Collider, the accelerator ring has a substantial circumference of 27 km (see figure 2.6); the Tevatron by comparison has a circumference of just under 6.5 km.

The original design for the LHC was conceived in the early 80s, which was before the Large Electron Positron Collider (LEP) was built [15]. LEP ran from 1989 until the year 2000, when it was decommissioned to make way for the much more powerful LHC which resides in the old LEP tunnel.

Before the protons are to be accelerated in the LHC ring, the beams will first travel through the other accelerators at CERN. After being gradually accelerated they will be injected into the LHC from the Super Proton Synchrotron (SPS) at an energy of 450 GeV. The LHC itself accelerates the protons to 7 TeV, which is equivalent to a speed of $0.999999991c$ and at this speed the protons perform 11 000 laps of the LHC per second.

To minimize interactions between the protons and air molecules and thereby sustain the high energy of the beam, a high quality vacuum is maintained in the storage ring. In figure 2.7 an overview of the complex of accelerators, through which the proton beams travel, is shown.

2.2.2 The accelerator properties

Bunches of particles

The proton beams are not continuous, but are deliberately grouped into bunches. This is partially to facilitate the acceleration (discussed later in this section) but also to provide a regular 'clock' in the detectors, which is essential for triggering and defining the events. Each of the two beams in LHC will contain 2808 bunches with a minimum distance of approximately 7 m between them [15]. Each bunch contains about 1.15×10^{11} protons and has a length of about 11 centimeters and a diameter of about 400 μm. To increase the probability of collision, the bunches are compressed close to the interaction points such that their length is reduced to just over 7.5 cm and their diameter down to about 16 μm. Despite the large number of protons in each bunch, there will only be about 20 proton-proton collisions per bunch crossing during a high luminosity run. Although this number seems low,

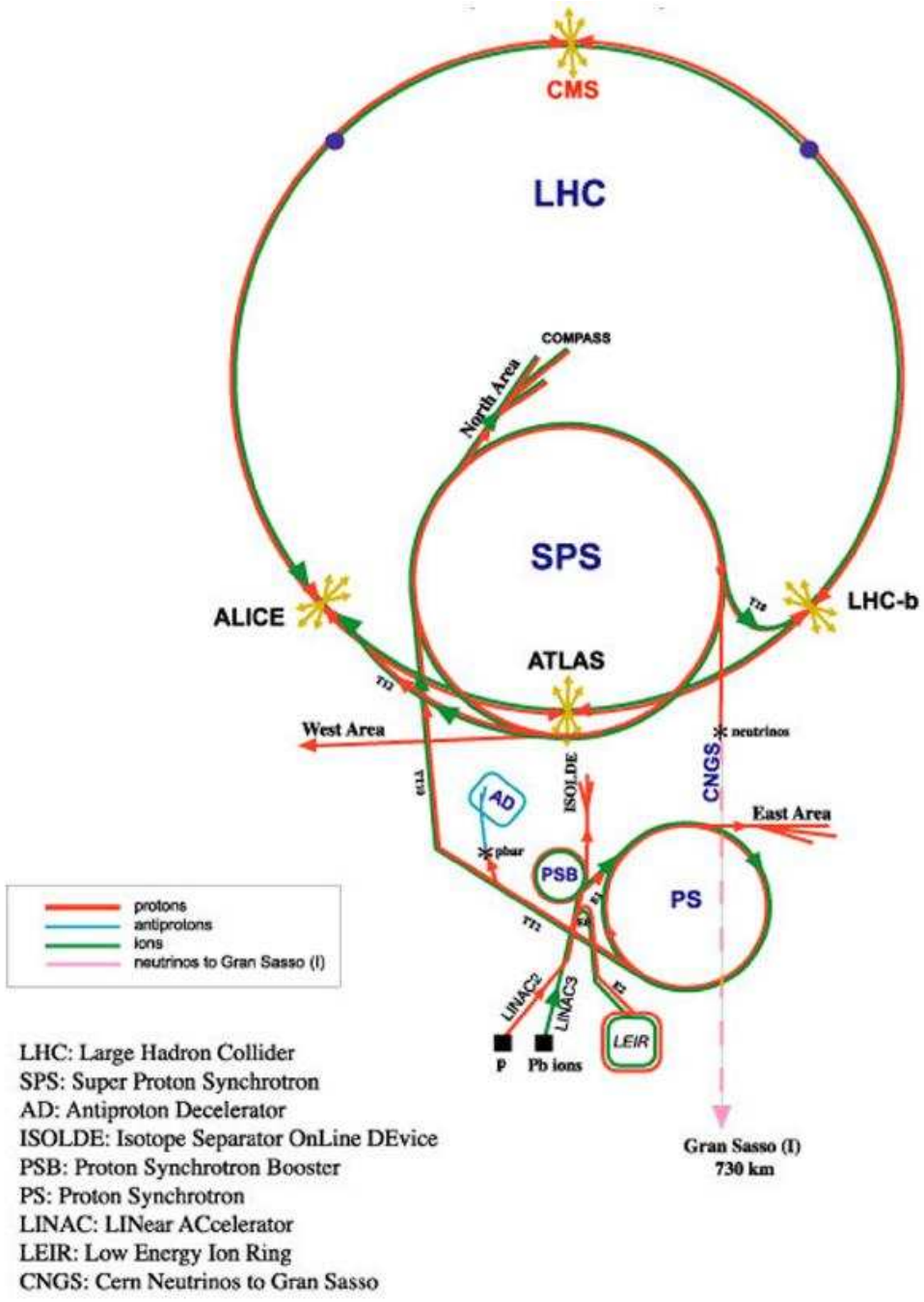

Figure 2.7: The accelerator complex at CERN, which gradually accelerates the proton beams [20].

given that the bunches cross as often as 40 million times per second in the LHC (ca 40 MHz), the total number of collisions is over 800 million per second. According to [10], 1% of the proton collisions result in a b or $\bar{b}$ quark.

Acceleration and deflection in the LHC

The proton beams are deflected along the LHC ring using 1232 superconducting dipole magnets, each having length of 14.3 m [15, 16]. The beam is focused by 858 smaller quadrupole magnets. Another 6200 correction magnets are required to suppress unwanted resonances in the accelerator. Liquid helium is used to realize the low temperatures needed to induce superconductivity in the magnets; the substance also has a high efficiency of heat transportation. Each magnetic core is bathed in helium at a temperature of 1.9 K. The cryogenics used to supply the helium make up the largest refrigerated system in the world.

The proton beams are accelerated in the storage ring using 8 superconducting radio frequency oscillators per beam. These provide a resonant electric field that can either accelerate or decelerate the particles depending on when they arrive at the resonator cavity. The beam continuously looses energy as synchrotron radiation due to the deflection of the protons (to keep them within the curved sections of the beam pipe), so the RF oscillators must compensate for these effects as well as providing the acceleration from 450 GeV to 7 TeV. The design of the cavities explains why bunches of particles, as opposed to a continuous beam, are required - it is important to match the phase of the bunch frequency with the RF oscillators so that the acceleration of the protons is optimal. In the case of the LHC, the oscillation frequency of the RF cavities is around 400 MHz, increasing somewhat to maintain resonance as the protons are accelerated. The magnetic field must simultaneously increase to avoid dispersion and thereby keep the radius constant.

The RF oscillator also assists in keeping the particles within the bunches. Consider for instance a particle at the end of a bunch, arriving somewhat later in the oscillation than average. The proton will then feel a stronger electric field than the protons in the middle of the bunch and will be accelerated to a greater extent. Protons that arrive early in the RF oscillator will be slightly delayed and will therefore become closer to the center of the bunch. The oscillations of particles around the optimal phase in the center of the bunch are called synchrotron oscillations, hence the name of the accelerator type. An LHC beam is sustained for about ten hours in the storage ring, traveling a great distance during this period of time. Further details about accelerators may be found in [17].

Luminosity

To maximize the event rate from the accelerator, the luminosity must be high. The luminosity describes the intensity of the beam and equals the number of particles passing through a cross section of the beam per unit time. If two bunches cross with frequency f, containing n_1 and n_2 number of particles each, the luminosity is given by [16]:

$$L = \frac{f n_1 n_2}{4\pi\sigma_x\sigma_y} \tag{2.8}$$

where σ_x is the beam cross section in the direction of the x-axis and σ_y in the y-axis.

At the LHC, the bunch cross frequency is around 40 MHz. The LHC will go through three different luminosity phases; initially a low phase ($5.0 \cdot 10^{32}$ cm^{-2}s^{-1}), followed by a high luminosity phase ($2 \cdot 10^{33}$ cm^{-2}s^{-1}) before it finally reaches its peak value of 10^{34} cm^{-2}s^{-1}.

2.2.3 The four different experiments

The collision points at the LHC correspond to four of the CERN experiments (see figure 2.7). These are ATLAS (A Toroidal LHC ApparatuS), CMS (Compact Muon Solenoid), ALICE (An LHC Ion Collision Experiment) and LHCb (LHC beauty). Each of the detectors are of radically different design to ensure the full range of possibilities are explored in the LHC, and also to cross-check measurements (*e.g.* if the Higgs is found in ATLAS with a certain mass, failure to find it in CMS will be a serious concern). CMS and ATLAS are general-purpose experiments designed to capture as broad variety of physics as possible. The ALICE detector is designed to study heavy ion collisions, which will be provided by Pb^{82+} ion beams later on in the LHC programme. LHCb is a smaller experiment dedicated entirely to B-physics; this machine has several special features including a defocussed beam to reduce luminosity and a series of Ring Imaging Cherenkov (RICH) detectors for the identification of protons, kaons and pions. This device resembles a fixed target single-arm experiment and is conical in shape, whereas the other detectors are broadly cylindrical with the interaction point in the centre.

2.2.4 b-quark production in the LHC

One of the first issues to deal with when the LHC starts running is to measure the total $b\bar{b}$ pair production cross section $\sigma_{b\bar{b}}$. This is a non-trivial task, especially at LHC energies which are not fully understood at the quark level; extrapolation from known regimes is needed and many uncertainties therefore exist. According

to [18] all estimates seem to indicate a quite large value due to the high luminosity. According to [10], the cross section $\sigma_{b\bar{b}}$ is expected to be around 500 μb. Figure 2.8 shows the simulated $b\bar{b}$ production cross section as a function of the minimum of the transverse momentum p_T^{min} of the b-quark, for the Tevatron and the LHC. The continuous, dashed and dotted curves in the plot represent different methods of measuring the cross section. The constraint of $\mid \eta \mid < 0.9$ corresponds to the acceptance of the ATLAS detector. As can be seen in the figure, the high energy and luminosity at the LHC increases the $b\bar{b}$ production cross section.

The high flux of quarks at the LHC could also cause the production of two or more $b\bar{b}$ pairs in one pp collision. Multiple $b\bar{b}$ pairs are then produced simultaneously by different quark collisions within the single pp collision event. Due to this large $b\bar{b}$ cross section, a huge amount of B-physics data will be available soon after start-up. This data will assist in examining and comprehending the detector performance and alignment, as well as testing QCD by studying the production mechanism.

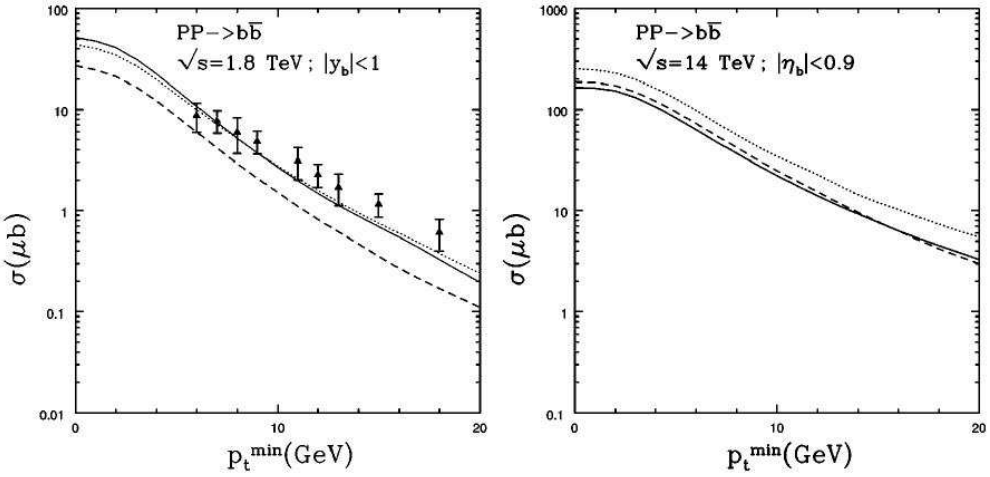

Figure 2.8: The $b\bar{b}$ production crossection as a function of p_T^{min} at the center of mass energy of $1.8TeV$ on the left and $14TeV$ on the right [18].

The need to trigger

When two bunches cross at the collision points in the accelerator, the probability of having more than one proton-proton collision is high. Such multiple events are referred to as pile-up events, and the average number varies from 4 (during low luminosity runs) to 22 (high luminosity) [16]. Additionally, even a single collision can produce many fragments which leave signatures in the detector, and furthermore most events do not lead to interesting physics. Most of these events will be

so called minimum bias and mainly consist of low energy hadrons. The average number of these low momentum hadrons in a minimum bias event is 7.5, which indicates that during each bunch crossing, there will be hundreds of particle tracks to record.

As a result a reliable triggering system, which only records events containing signatures of interesting physics, is needed. Events that do not pass the trigger are immediately discarded and are not committed to disk for detailed reconstruction and analysis. The triggers must be highly flexible to enable different physics regimes to be studied, and this implies that they must be programmable. Objects that can be used to activate the triggers include high energy photons and leptons as well as jets. For further information about triggering see chapter 2.3.6 and later sections of this work.

2.3 The ATLAS Experiment

ATLAS is a general-purpose detector, implying that it collects data using as many different physical entities as possible, and the project is not designed with any specific type of physics in mind. Some of the main goals of the ATLAS experiment are:

- Validation of the Standard Model by searching for particles that have been studied in other experiments (such measurements will also be used to tune the detector and understand its performance).
- Finding the Higgs boson and hence the origin of mass.
- Searching for additional CP violation in B meson decays
- Thorough study of poorly understood QCD and top-quark processes
- Searching for effects of New Physics, especially SUSY (supersymmetry), and new massive vector bosons.
- Searches for exotic particles and effects such as internal degrees of freedom within quarks and leptons, and massive weakly interacting particles that may be the constituents of Dark Matter

2.3.1 The concept of ATLAS

The ATLAS detector is the largest accelerator-based particle detector in the world. It is a massive device weighing approximately 7000 tons, and having a length of some 46 metres and a diameter of about 11 metres. The detector is described in detail in the Technical Design Report (TDR) [10, 19], but an overview is layed out here, focusing on issues relevant to this report such as the inner detector, the muon systems and the trigger.

The basic design criteria for ATLAS include the following [19]:

- Efficient tracking, enabling full event reconstruction at low luminosity and identification of leptons, photons and heavy avour jets at high luminosity.
- High-precision muon momentum measurements, using the muon spectrometer alone at high luminosities.
- Electro-magnetic calorimetry for photon and electron identication and energy measurements, and hadronic calorimetry for jet and missing energy measurements.

- Triggering on low p_T particles, enabling selection of physics events with high efficiency.
- Large acceptance in pseudorapidity η and almost full azimuthal angle coverage ϕ.

The detector is made up from several sub-detectors, each of which fulfills different criteria (see figure 2.9). These components are generally designed as concentric cylinders. The Inner Detector is a high-precision tracking detector, located at the centre of the machine close to the interaction point. It is equipped with a central solenoid magnet that provides a magnetic field with a strength of 2 T. The Inner Detector is surrounded by an electromagnetic and a hadronic calorimeter, which measures the energy of electrons, photons and hadrons with high resolution. The outermost sub-detector is the muon spectrometer which detects muons as they leave the machine. Since only muons are sufficiently penetrative to punch through the calorimetry, the muon spectrometer acts as a muon identifier. There are three large air-cooled toroid magnets surrounding the muon spectrometer providing a field of peak strength at 4 T. ATLAS also has a three-level selective trigger system that operates at very high speed.

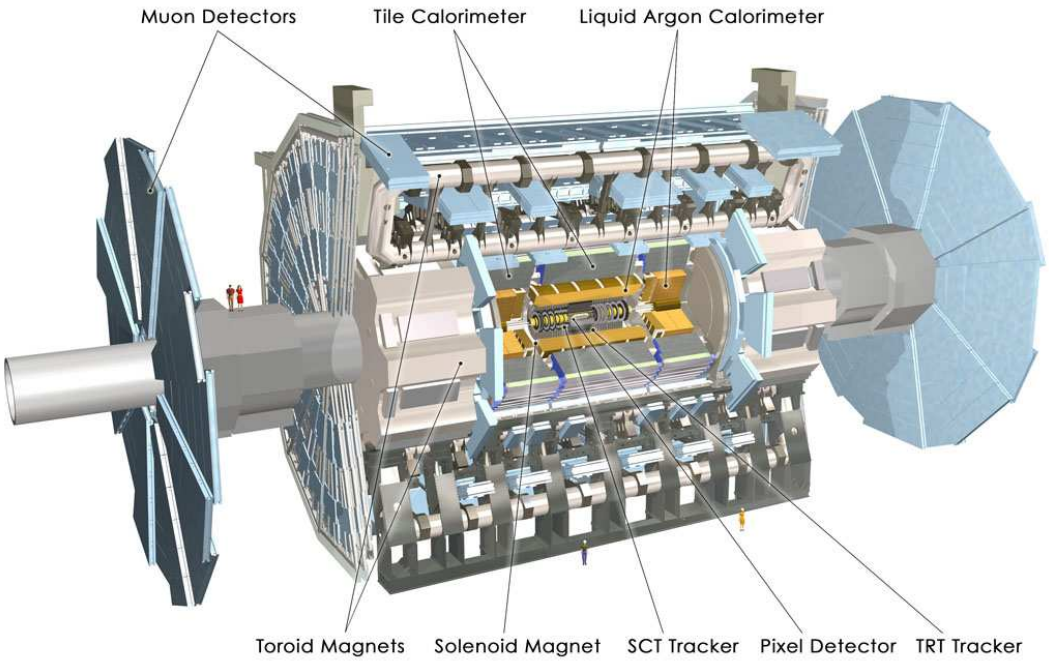

Figure 2.9: The ATLAS detector [20].

2.3.2 Orientation

Since ATLAS has a cylindrical shape with the beam-axis in the middle, it is a natural choice to define the beam direction as the z-axis. The xy plane is orthogonal to the beam axis (see figure 2.9). The transverse variables, such as transverse momentum p_T, are in the xy plane unless stated otherwise. The positive x-axis is defined to point from the detector to the origin of the LHC ring and the y-axis must therefore be pointing upwards towards the surface of the earth.

In cylindrical coordinates this translates into an azimuthal angle, which is measured around the beam axis with the x-axis defining $\phi = 0$. The polar angle θ is measured from the positive beam direction towards the y-axis and the z-axis is the same as for the Cartesian coordinates.

The pseudorapidity η is defined as $\eta = -\ln(\tan(\frac{\theta}{2}))$ and is a convenient way of describing the polar angle, due to the fact that the quantity is Lorentz invariant. The pseudorapidity is zero at $\theta = 90°$ and ∞ and $-\infty$ at $\theta = 0°$ and $180°$ respectively. Pseudorapidity is a good measure of how precisely a particle's track can be reconstructed, since it is harder for the detector to make valid measurements on trajectories close to the beam axis. The distance ΔR in the $\eta\theta$ space is given by $\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$.

2.3.3 The inner detector

The Inner Detector has the shape of a barrel and contains cylindrical layers of detectors with rings of subdetectors in the end-cap region (see figures 2.10 and 2.11). The inner detector is 7 m long and has a diameter of 2.2 m [19, 21]. The layout provides a full tracking coverage over an angular range of $9° < \theta < 171°$, equivalent to the pseudorapidity $\mid \eta \mid < 2.5$. The innermost layers of the barrel consist of a pixel system since it provides high tolerance for radiation and good resolution in three dimensions. A semiconductor tracker (SCT) surrounds the pixel detector and at the outer layers a transition radiation tracker (TRT) provides continuous tracking.

The inner detector has a number of roles to play. Firstly, it measures the particle momentum. It also provides high-granularity tracking measurements with up to 40 points per track, which enables the reconstruction software to build the tracks with excellent precision, and enables them to be distinguished from one another close to the interaction point, where the density of tracks is very high.

Additionally, the inner detector is used to distinguish between electrons and pions and measuring the sign of the charge of electrons with high transverse momentum. The energy of the electrons is measured further out in the electromagnetic calorimeter. Other requirements of the detector are to participate in the Level-2

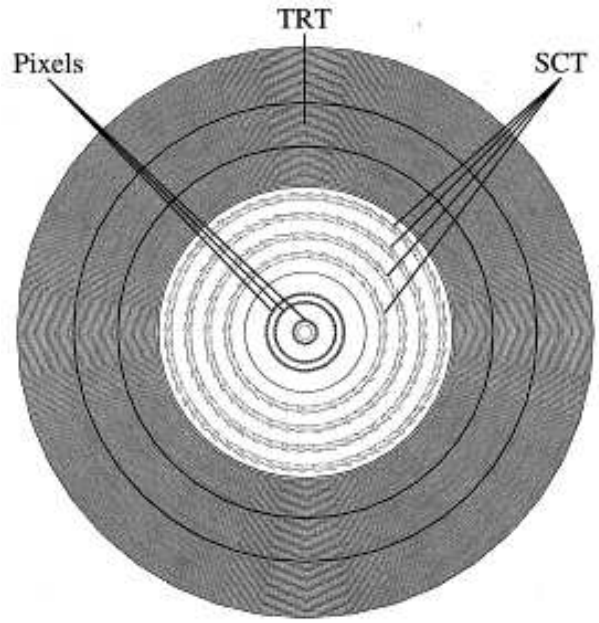

Figure 2.10: Cross section of the ATLAS inner detector [21].

Trigger (for details see section 2.3.6) and to provide lifetime tagging of tau and b-jet decays.

To fulfill these requirements, ATLAS combines high precision tracking via the pixel and SCT systems with continuous tracking through the TRT, which provides good separation and pattern resolution. The pixel detector together with the SCT are both fine-granularity detectors satisfying the requirement of high momentum, despite the large concentration of tracks from the collision.

The pixel system

The pixel system has to be both resistant to radiation and finely constructed since it is the innermost part of the detector. It is constructed to accomplish high-precision measurements as close to the primary interaction point as possible [19, 22]. The detector contains 80 million pixel elements with dimensions of 50 μm in the $R\phi$ direction and 400 μm in the z direction. Tracking errors are therefore very small. There is therefore not much room for precision errors. The pixels are contained in detector modules and each module includes 16 chips that are read out with electronics mounted on the reverse side.

The detector consists of three barrels of radii 5, 9 and 12 cm [19, 21]. The innermost barrel is called the B-layer, since it is constructed to facilitate the reconstruction of secondary vertices with high precision in order to measure lifetimes of short-lived particles such as B-hadrons (see later). The impact parameter resolution of the inner detector is dominated by the thickness of the B-layer pixels. The

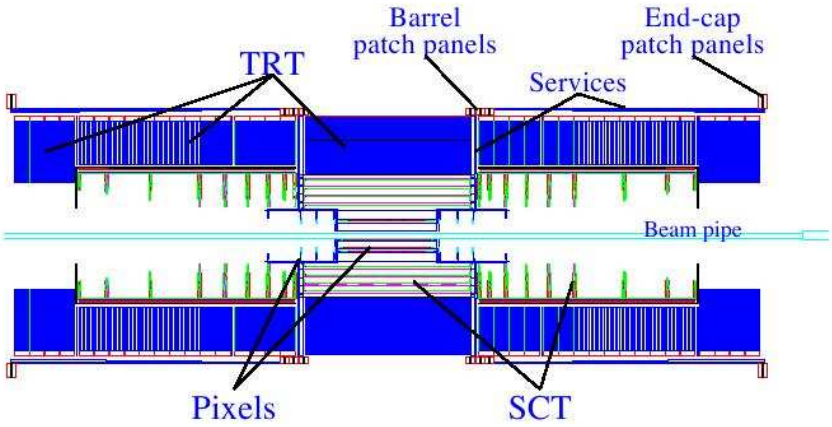

Figure 2.11: A longitudinal view of the ATLAS inner detector [19].

B-layer will have to be replaced after a couple of years due to radiation damages. The pixel system also contains pixel elements on five disks on each side of the barrels in order to complete the angular coverage.

The semiconductor tracker

The SCT detector is constructed to make eight high-granularity measurements per track to assist in the measurements of momentum, vertex position and impact parameter [19, 22]. The SCT consists of four barrel layers at radii of 30, 37, 44 and 51 cm together with 9 end-cap wheels on each side of the barrel. Containing 61 m^2 of silicon detectors, there are 6.3 million read-out strips. To prevent the detectors from as much radiation damage as possible, both the SCT and the pixel detector are cooled down to be operated at around $-7°C$ [21].

The transition radiation tracker

The main goals of the transition radiation tracker are to provide additional tracking capability and to distinguish between the tracks of electrons and pions. The TRT uses 351 000 straw detectors located axially in the barrel and radially in the end-caps [19, 22]. The detectors are able to operate at a high rate, which is necessary such that they can respond to and read out any hits before the next bunch crossing. This is achieved by the small diameter of the straws, as well as the fact that the sense wires are isolated within separated gas volumes. The straw tube technology is also intrinsically radiation-hard, unlike the silicon components.

The identification of the electrons and pions is made possible by using a gas containing 70 % Xenon to detect the transition-radiation photons, which are created in between the straws [21]. In this manner, a large number of measurements can be made; at least 36 points on each track. The pions can be distinguished from

the electrons through their transition radiation characteristics. At a momentum of a few GeV, electrons will deposit approximately 7 keV in more than 7 different straws, while pions will deposit the same amount of energy in only one or two straws.

2.3.4 Calorimeters

There are two calorimeters to measure position and most importantly energy; the electromagnetic and the hadronic. The electromagnetic calorimeter measures electrons and photons, while the hadronic calorimeter concerns hadrons and jets. Both calorimeters use a highly granular liquid argon technique (LAr) and the overall pseudorapidity is $\mid \eta \mid < 4.9$. The most important feature of the calorimetry system is that it provides very precise measurements of jets and missing transverse energy. Another important application is to filter out all remaining particles except for muons, to prevent the muon spectrometer from being polluted by hadrons and electrons. For an overview of the ATLAS calorimetry system see figure 2.12.

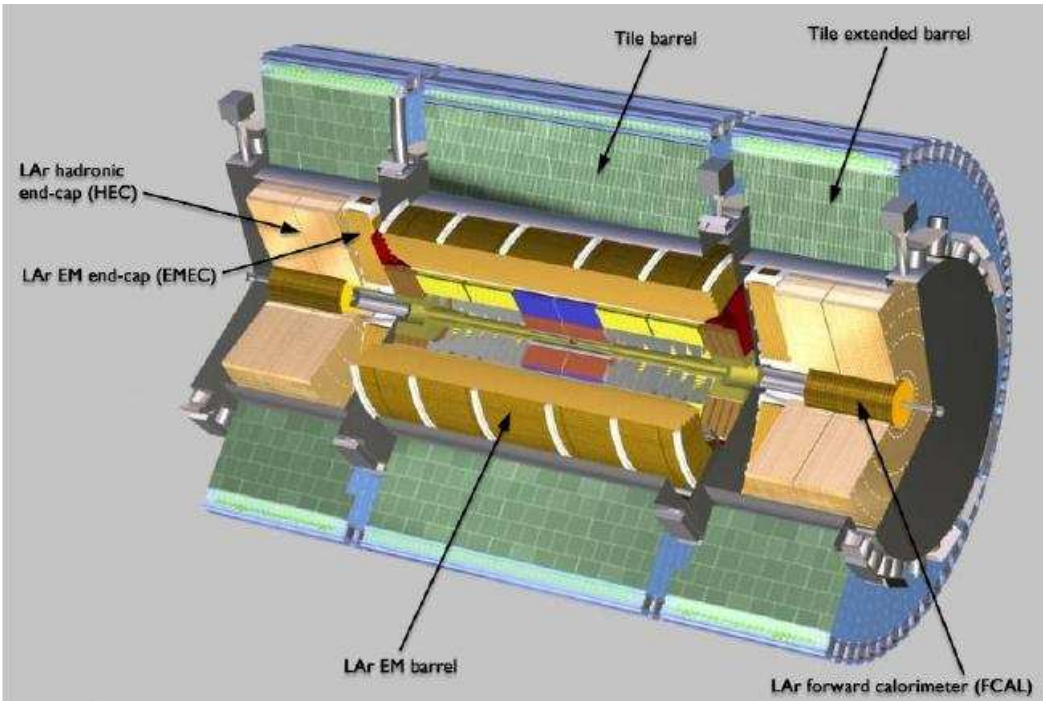

Figure 2.12: The components of the ATLAS calorimetry system [20].

The electromagnetic calorimeter

The EM calorimeter consists of two end-caps and a barrel, where the barrel is divided into two identical parts of length 3.2 m, with a 6 mm gap separating them at $z = 0$ [19]. The calorimeter is a LAr detector with lead absorber plates. It also

contains electrodes shaped as an accordion providing a complete coverage in the azimuthal symmetry.

The hadronic calorimeter

The hadronic calorimeter uses the radiation-hard liquid argon technology for higher pseudorapidities (where radiation levels will be higher) and plastic scintillator tiles embedded in iron absorbers for lower values ($|\eta| < 1.7$). These scintillator tiles are arranged radially in a periodic pattern, with an iron layer between each period. The emissions from each tile are read into photomultipliers via fiber optics.

2.3.5 The muon spectrometer

High p_T muons are the only charged particles that are able to punch through the calorimetry. Muons are often the result of interesting physics and are therefore of vital importance as trigger signatures. To measure the properties of the muons, the muon spectrometer was constructed to surround the rest of the ATLAS detector (see in figure 2.9). It is the muon spectrometer that defines the large dimensions of the ATLAS detector and it is by far the biggest subdetector.

The concept of the spectrometer is to deflect the path of the muons with the superconducting toroid magnet systems, consisting of eight long barrel toriods and two inserted end-cap magnets [19]. In this way, high-resolution measurements of muon momentum can be obtained. The precision measurements of the muon tracks are made in the $R - z$ projection, which is parallel to the bending direction of the magnetic field.

The trigger chambers in the muon spectrometer have three main purposes. First of all they have to be able to separate and identify the bunch crossings, requiring a good time resolution. The trigger must also have well-defined p_T cut-offs in moderate magnetic fields, which requires a granularity of order 1 cm. It must be able to measure the second coordinate orthogonal to the coordinate measured in the precision chambers with a resolution of the order of 5-10 mm.

2.3.6 The trigger system

As already mentioned in chapter 2.2.4, the trigger system is essential if ATLAS is not to be overwhelmed by data from the enormous number of particle tracks in the detector. As an example, consider the 40 MHz bunch crossing frequency. The rate of selected events must be reduced to approximately 100 Hz to be able to store the data permanently. In the ATLAS experiment, the trigger system provides a solution to this problem by filtering the data recorded by the subdetectors. The ATLAS trigger system contains three different levels of triggering, where each level

reduces the number of accepted events. For a schematic overview of the ATLAS trigger system see figure 2.13.

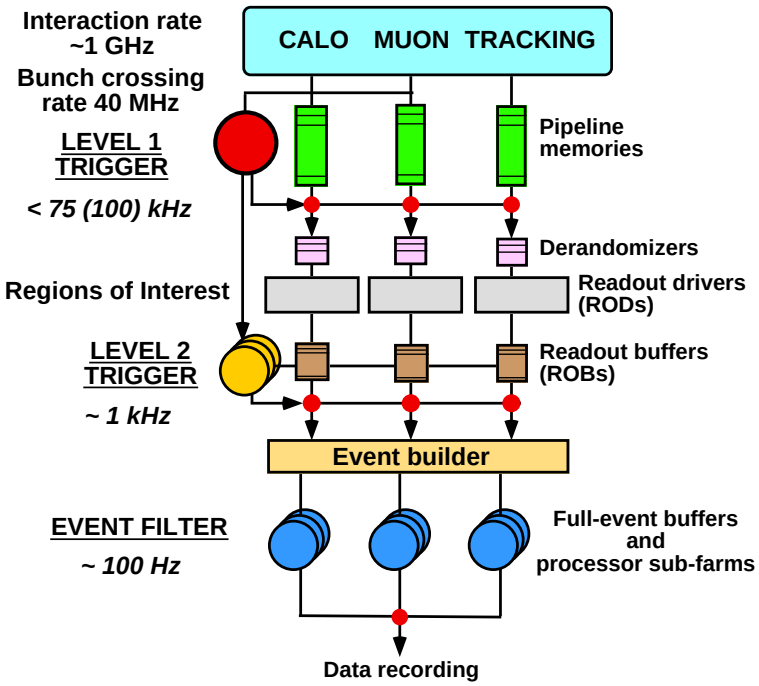

Figure 2.13: An overview of the ATLAS trigger system [19].

First level trigger

The Level-1 (LVL1) Trigger makes the initial decision as to whether or not to save the detector data from a given collision [19]. This data is initially stored in the so-called pipeline memories, which are contained within the detector itself. These have a limited capacity and, given that a muon from one collision will reach the muon system after the next bunch crossing has occurred, the tracking pipeline memories will rapidly accumulate data. The decision on whether to save or dump a given event must therefore be made very quickly. For this reason the selection of events is therefore based only on reduced-resolution information from the trigger systems in the muon spectrometer and the calorimeters. LVL1 signatures include jets, high p_T electrons and photons, missing energy as well as high p_T muons in the muon spectrometer. The locations where these objects are found are then defined as Regions of Interest (RoI) for potentially interesting events. The RoIs extend as a cone from the interaction point to the outer parts of the ATLAS detector.

The LVL1 trigger is integrated in the detector and has programmable thresholds that can be adjusted according to luminosity and physics requirements in the particular run. Most of the physics requirements of ATLAS can be met by using

relatively simple criteria already at the first trigger level. For B-physics, the LVL1 trigger implies an inclusive muon trigger with a p_T-threshold of 6 GeV [21].

Second level trigger

Events selected by LVL1 are read out from electronic systems at the front-end of the detector into readout drivers (RODs) and further on into readout buffers (ROBs). The Level-2 Trigger then scans the events from the ROBs, now reading the data with full-resolution with all subsystems. The aim of the LVL2 trigger is then to reduce the event rate from LVL1 by two orders of magnitude [21], which can be achieved by applying the restrictions of the regions of interests (RoIs). The trigger is implemented in software and runs on a cluster of processors separated from the detector but very near to it. The decision making time for this step of the trigger system is in the range of (1-10) ms, which is significantly longer than the LVL1 trigger. For B-physics, the importance of the LVL2 trigger will be as great for low luminosity as for high, since the effort to select events of interest will be larger.

The event filter

The data accepted by the LVL2 trigger system is passed onto the event builder [16]. This is again implemented in software and runs on another set of nearby processors. The event builder does a full reconstruction on the event and sends it on to the final stage of the ATLAS trigger system - the event filter. The event filter can make far more advanced selections including topological and vertexing cuts. The total rejection factor for the whole ATLAS trigger system is $5 \cdot 10^6$. For more details about the trigger or the data-acquisition system see [19].

2.3.7 B-physics in ATLAS

As discussed in chapter 2.2.4, the $b\bar{b}$ production rate at the LHC will be very high. This presents both huge opportunities and huge difficulties for those physicists wishing to study B-physics in the LHC environment. The opportunity is in the enormous B-event statistics, and the difficulties are in the severe backgrounds which must be contended with.

B-physics analyses typically involve detailed measurements of specific decay channels. Such measurements require a clean experimental environment, which obviously is not the case at the LHC due to pile-up and the large number of additional tracks in a single event which arise from entities other than the $b\bar{b}$ pair. The main issue for any B-physics measurement at the LHC (as well as the work

of this thesis) is obtaining accurate identification of the decay channel of interest among the large amounts of background data.

The LHCb detector, which is an experiment entirely dedicated to B-physics, simplifies this problem to some extent by decreasing the luminosity at the interaction point by using defocusing magnets. LHCb also facilitates the identifications of many decay channels by distinguishing between pions and kaons in two of its sub-detectors (RICH-1 and RICH-2). ATLAS, on the other hand, lacks the ability to distinguish between different kinds of hadron directly. ATLAS B-physics studies must also contend with the full LHC luminosity, since most of the activities of ATLAS are oriented to the discovery of new entities which clearly requires as high a luminosity as possible. B-physics measurements in ATLAS are therefore non-trivial.

This, however, will not prevent B-physics research from taking place in ATLAS. It only puts a stronger demand on the trigger system and the offline analysis procedure. The ATLAS detector is in many ways particularly suitable for B-physics. The tracking instruments in the inner detector will, for instance, provide high quality tracking information. This is essential for B-physics measurements since the B-hadrons decay in the beam pipe and in order to measure the lifetime of a given particle (which is at the heart of, for example, CP-violation studies) it is necessary to accurately calculate the position of the particle's decay vertex (referred to as the secondary vertex). The covariance matrix of the track parameters determines the uncertainty on the position of a vertex constructed with those tracks, so the better the quality of the tracks, the better the resolution of the vertex position (and hence the lifetime) measurement. The high-quality tracking of ATLAS provides excellent secondary-vertex resolution. Additionally, since most B-physics measurements begin with the detection of muons as trigger signatures, the muon spectrometer in ATLAS, which measures muon momentum with a very high resolution, is the other main strength of the detector as far as B-physics is concerned.

In ATLAS the B-physics triggering depends mainly on detection of di-muons in the LVL1 trigger with $p_T(\mu) > 6$ GeV in the barrel and $p_T(\mu) > 3$ GeV in the end-caps [23]. The muons are then reconfirmed in the muon tracking chambers. The cross section for a $b\bar{b}$ event to pass the ATLAS trigger is 110 nb according to [24].

The main background to the exclusive decay channels arises from single muons coming from pions and kaons in flight. The majority of these non B events are removed by the LVL2 trigger by comparing the muons to the inner detector tracks. The decay channels where the B-meson decays through a $J/\psi \rightarrow \mu\mu$ are especially appealing since they are self-triggering according to the di-muon scheme. These decay channels also imply several interesting physics processes.

The problem of hadron identification remains, however. By necessity ATLAS B-physics offline analyses must treat all tracks not identified as leptons equally. The usual procedure is to form up combinations of tracks for a given decay, testing whether they are consistent with having arisen from a common vertex, and calculating quantities such as their invariant mass, applying a hypothesis on the mass of each track according to the decay in question. The hope is that incorrect combinations will either fail to fit to a single vertex or will have a self-evidently wrong invariant mass. This is not always the case however, and combinatorial backgrounds are always present for these kinds of analyses, particularly for long-chain decays with many stages. In these cases the analysis code will often find genuine hadronic decays of the type sought in the analysis, but which are not actually part of the decay chain of the B-hadron. For instance consider the code for searching for the decay $B_s^0 \to J/\psi\,(\mu^+\mu^-)\,\phi\,(K^+K^-)$. The algorithm will often fall victim to real J/ψs and ϕs that are not from the B_s chain. Usually other more sophisticated cuts can eliminate these, but not always.

Implicit in all this is the existence of secondary vertex finding in the offline analysis. This is somewhat unusual in ATLAS, which normally uses vertex-finding code at the reconstruction stage for finding the primary vertex position.

Chapter 3

The Analysis Procedure

Since the LHC is not yet in use, simulations of proton-proton collision events are made in order to prepare for the real physics data. The simulations require several software packages, which are briefly discussed in this chapter. Simulated data is then passed through reconstruction - the same code that will be used on real ATLAS data. This is of considerable importance since, knowing the entire truth about the original events from the Monte Carlo generator, the reconstruction code performance can be assessed and it can be optimized in advance of the real data arriving. Furthermore, using the output of the reconstruction, full physics analyses can be developed and tested, and the results of these exercises will guide the activities of the experiment once the data becomes available. The work presented here is an example of this kind of exercise.

3.1 Software analysis tools

There is an important distinction between the online and offline computing concerning the software and infrastructure at the LHC experiments. Online computing deals with the taking of data, whilst offline computing concerns the further management of the data once it has been recorded and written to disk. A few examples of tasks for the online computing are: monitoring the detector and implementation of the trigger system. Offline software includes reconstruction of raw data, physics analysis and data simulation.

ATLAS mainly uses an offline software framework called Athena. The physics analysis programming for the work of this thesis, however, uses a newly developed software package called AAna, which has been developed with B-physics especially in mind [25].

Large sets of simulated events must be produced to allow preparation of software and analysis techniques. According to [19], the data to be archived from the ATLAS experiment alone is estimated to about 1 PByte per year and therefore storing this

massive amount of data presents a challenge. For instance, storing the data at one single location would be impractical. Instead, the LHC experiments will use a number of computing grids (known collectively as "The Grid") [26]. Computing grids consisting of a large number of processors and storage distributed all over the world. To the user, the Grid should appear to be a single cluster of computers and not an entire network. There are many computing grids in the world but the high energy physics community mainly uses three. The LHC Computing Grid (LCG) is based in Western Europe, Canda and Asia. OpenScience Grid (OSG) is located in the United States. NorduGrid is primarily situated in the Nordic European countries. The computers which constitute these grids may also contribute to other scientific research grids as well - for instance, machines contributing to the LCG may also be used for medical science applications.

3.1.1 Athena

Athena is an overarching framework that ties together the different packages of the offline software. It is based on the Gaudi framework that was initially developed by LHCb [27]. Athena is designed to steer all offline software processes and contains many components (packages). Applications which actually process the data are known as Algorithms; these can draw upon Services which are globally available. Athena is written in C++ but jobs are controlled through the use of Python scripts known as Job Options which are read in at run-time. For an overview of the Athena components see figure 3.1.

3.1.2 Event generation

The purpose of the event generator is to simulate the proton-proton collisions and the subsequent production of particles, up to the point at which the particles pass into the detector. The output data is in the form of lists of information concerning particles such as four-momentum, time of production and decay, together with the parent particle and decay products [16]. Because of the complexity of the underlying physics of the processes behind the particle interactions, and our very incomplete understanding of these processes, the data produced by the generation is merely a "best guess" as to what will really happen in the LHC.

Pythia, PythiaB and BSignalFilter

There are several software packages used for event generation. These are interfaced to Athena. The standard event generator package for proton-proton collisions is called Pythia [28]. This is a general-purpose generator, which simulates the entire event production chain using Monte Carlo techniques. Since only about 1% of the

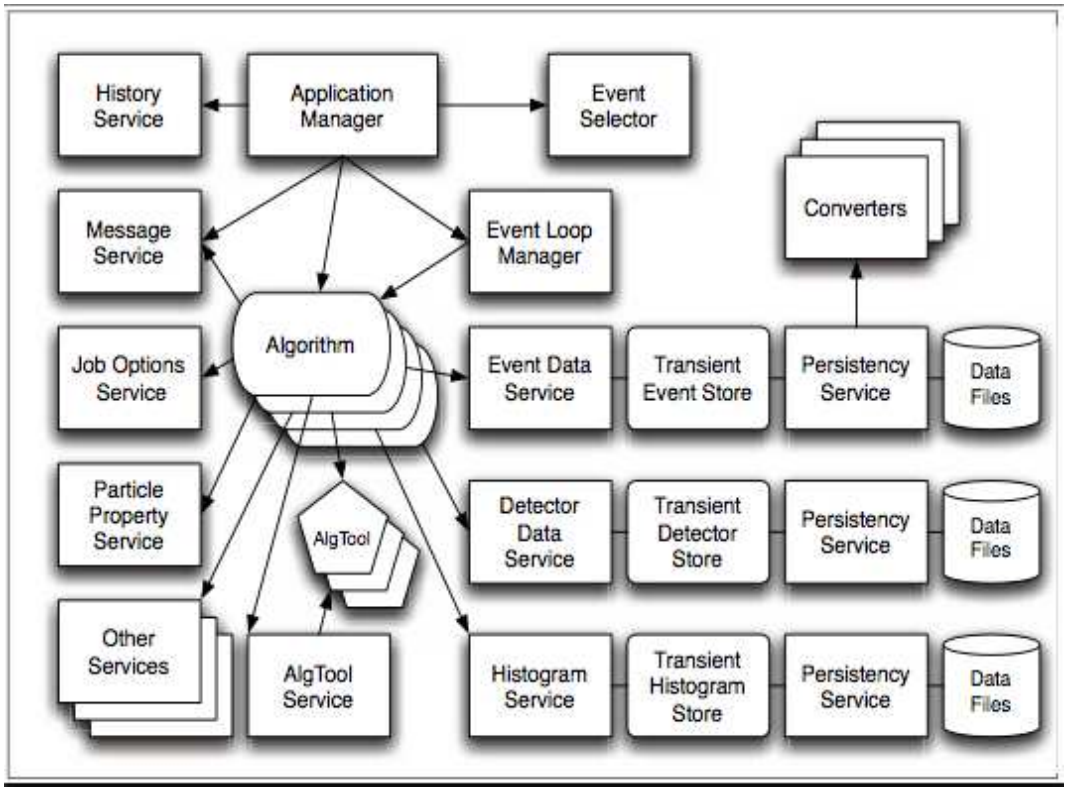

Figure 3.1: A model of the components of the Athena framework [27].

proton collision events produces a b or $\bar{b}$ quark, the efficiency of Pythia for B-physics production becomes quite low. An Athena package called PythiaB was developed in order to mitigate this problem [29]. When PythiaB detects that Pythia has made a $b\bar{b}$ quark pair, it interrupts the action of Pythia to prevent it from turning these quarks into hadrons. PythiaB clones the event a number of times before passing it back to Pythia for hadronization and decay. The efficiency of the generation is then increased considerably.

Once hadronization has occurred it is possible to either force the hadrons to decay to a certain channel via Pythia control strings passed from the Job Options file, or to let Pythia go ahead and decay the hadrons according to its internal tables. Events containing a certain decay can also be selected rather than forced (such that events not containing a given decay are thrown away and Pythia is ordered to try again). The difference is subtle but important, for in the former case it is necessary to correct the final Pythia cross section according to branching ratio of the process which was forced. In the latter case, however, since Pythia "knows" that the event has been rejected and that it has to throw the dice again, it accounts for the rejected events and includes them in the cross section calculation. There is therefore no need to apply any corrections for selected decays. For more details concerning Pythia consult Ref. [28].

The output produced by Pythia (in the HepMC format) can be further processed by a selection algorithm called the BSignalFilter [16]. The purpose of the filter is to remove all events that would not pass the ATLAS triggers, since the Monte Carlo data should not contain any events that are undetectable to ATLAS. BSignalFilter acts after the Pythia cycle has finished, so the cross section from Pythia must again be corrected according to the rejection factor from BSignalFilter.

3.1.3 Simulation software

Once generated, the events must be simulated by the detector simulation software [16, 27]. The purpose of the simulator is to mimic effects of the detector materials on the particles passing through it, and to represent the response of the active components of the detector that are interfaced to read-out electronics. The simulation software used is the gigantic C++ package Geant4, containing over one million lines of code. Geant4 is a general package providing tools for modelling the detector geometry, materials and magnetic field, and the wide range of effects that these have on the particles. The main algorithms of the package are based on the Monte Carlo technique and account for numerous physical processes. Geant4 is widely used for very different applications, including space and medical physics as well as for high energy physics. The particular implementation for ATLAS uses Athena's geometry and magnetic field services. In this way a full simulation of the behavior of the particles emerging from the interaction point, in the different parts of the detector, can be achieved.

The output of the detector simulation software is referred to as SDOs (Simulated Data Objects). These are then digitized such that they have the same format as ATLAS will provide when data is taken. The digitization step is separate from the simulation process. The simulated response of the active regions of the different sub-detectors are converted into a stream of electronic hits to match the detector output. The output of this stage is referred to as RDO (Raw Data Objects).

3.1.4 Reconstruction software

The process of reconstructing the physics event from the digital hits in the detector is a crucial step. The whole purpose of the generation and simulation of events is to optimize the reconstruction and the analysis, such that when the authentic data is produced, the reconstruction performance is as well understood as possible. The advantage of the simulated events is that they can be compared with the Monte Carlo truth, in order to investigate how reliably the reconstruction code can interpret the RDOs from the detector. To get an overview of the software processes

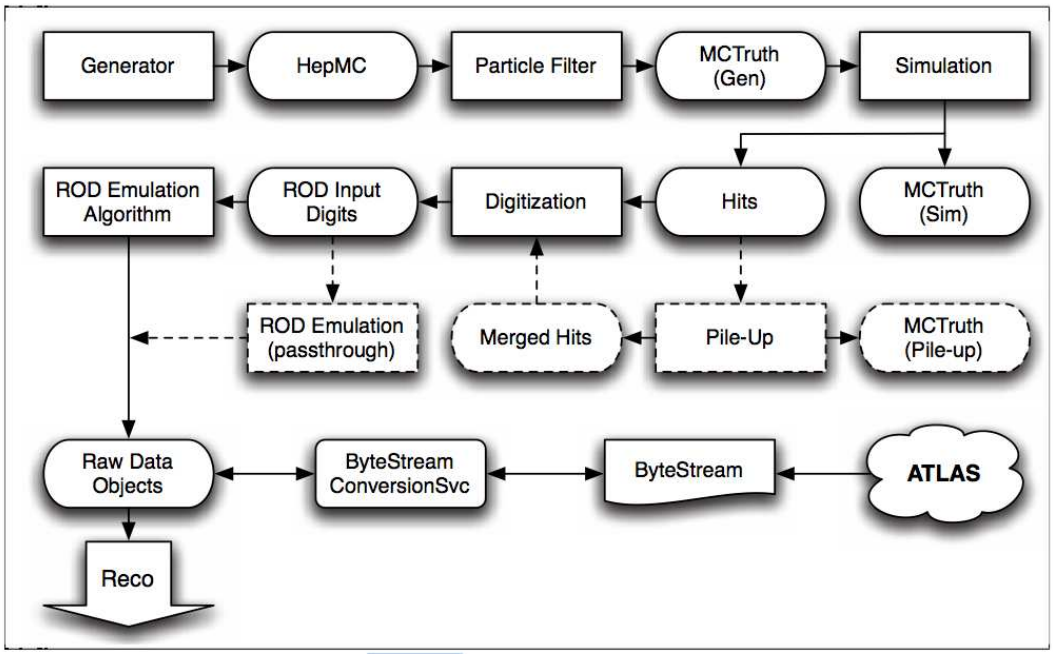

Figure 3.2: The software process from event generation to reconstruction. The data can also come directly from ATLAS in the form of real data, instead of being simulated. The rectangles represent the processing stages and the rounded rectangles represent the data objects that follows from the processes. The crosshatched rectangles are optional in that they are not essential to produce workable events. [27].

leading up to the reconstruction see figure 3.2.

To obtain a realistic image of the full event, the output from the reconstruction of the processes occurring in the different sub-detectors are merged into one with the *combined reconstruction* software [16, 27]. The final output produced by the combined reconstruction is then in the form of stored C++ data objects in the Event Summary Data (ESD). The data objects, which are produced by the pattern recognition software, represent geometric parameters such as tracks, vertices and energy clusters. These can facilitate the identification of particles such as photons, electrons, muons and taus, as well as the building of jets. Especially important for B-physics analysis in ATLAS are the tracking data and the muons.

3.1.5 Physics analysis software

Physics analysis is the final step of the data processing as well as the main part of the work leading up to this thesis. In this step, analysis code is used to carefully analyze the reconstructed objects in the events. The goal of the physics analysis varies from study to study, but in the case of this thesis, the purpose is to identify instances of a particular decay channel.

Analysis Object Data (AOD)

The Event Summary Data generally contains more information than is necessary for physics analysis and it is therefore inefficient to ship ESD around the world [16]. Instead, the ESD file is condensed into a more simple form known as AOD (Analysis Object Data). The AOD objects are designed to derive from properties of the particles they represent and inherit from the same *particle* class, which in turn inherits from a *four momentum* class. Examples of resulting AOD objects are *track particles, photons, electrons, muons* and *taus*.

For generated data, an important property of the AOD objects is the ability to associate the reconstructed objects with the 'true' Monte Carlo particles from the event generation or the Geant4 simulation. This enables the analyst to obtain the mass and lifetime resolution, as well as the reconstruction efficiency, for the particle or process in question. It also enables detailed investigation on exact composition of backgrounds.

AAna

The standard framework for physics analysis (as with all offline activities) in ATLAS, is Athena. However, Athena is a large package and not always the most efficient way of performing analysis due to very slow compilation times and execution times and the complexity of the code. To address these issues a prototype ATLAS code, AAna, has been developed by Guennadi Borissov and James Catmore from Lancaster University. AAna is written purely in C++ code and is smaller, faster and more flexible than Athena since it only contains classes concerned with physics analysis. It is also platform independent, running on any Linux or Mac platform. The code also contains an internal vertex-finding algorithm (which is essential for B-physics studies). AAna is based on a similar package from the D0 experiment at Fermilab, where it was successfully deployed in several published B-physics measurements.

AAna is entirely compatible with Athena but can also run in stand-alone mode. AAna data is built from a single algorithm in Athena, which converts data from the AOD file into a binary file containing simplified analysis objects. It does not modify the original data in any way, but simply copies the numerical information and stores it differently.

The analysis code written for this thesis used the AAna framework and the author of this thesis has consequently contributed to its testing. At the time of writing it has however not been determined if AAna will be accepted as official code for ATLAS or not. For more information concerning the AAna framework see Ref. [25].

ROOT

To be able to plot histograms, a data analysis package is required to read the *nTuples* (arrays) produced by the analysis code [30]. The most commonly used package is ROOT which was developed by CERN in 1994 (the logo is shown i figure 3.3). The name *ROOT* comes from the fact that data is stored in a *tree*, with substructures of *branches* and *leaves*. ROOT operates either from compiled C++ programs or directly via a command line interpreter. It has a huge range of features but the most frequently used are those associated with histogram drawing and statistical fitting. All histograms shown in this thesis are produced with ROOT.

Figure 3.3: The ROOT logo [30].

A brief summary of the different software steps of producing and analyzing events can be found in table 3.1.

Simulation	**Software package**	**Assignment**
Generation	Pythia/PythiaB	A general-purpose event generator. PythiaB increases the efficiency for B-physics events.
	BSignalFilter	Filters out the events that would not activate the ATLAS trigger system.
Simulation	Geant4	Geant4 models the design and performance in the detector and simulates the behavior of the interacting particles. Athena then digitizes the output.
Reconstruction	Various	Reconstructs the tracks and interactions of the particles in the different parts of the detector and then combines the information into the Event Summary Data, ESD.
Analysis	Athena algs / AAna	The ESD is compressed into AODs, carrying the Monte Carlo truth. Physics analysis of the events can then be made either directly on AOD using Athena or using AAna on pre-prepared binaries.
	ROOT	Framework for final analysis, tuning cuts, statistical fits and histogram drawing

Table 3.1: A summary of the different stages and software used to simulate and analyze proton proton collision events with ATLAS.

3.2 Analysis of the decay $B_c \to B_s\pi$ – Decay modeling and selections

With the software tools described in the first part of this chapter, proton proton collision events can be simulated and analyzed. In this thesis, the signal to be analyzed is the decay $B_c^+ \to B_s(J/\psi\phi)\pi^+$, and three different background processes are also included.

In the remaining part of this chapter, the procedure of the signal and background analysis is described together with how the selection cuts are chosen in order to optimize the significance of the signal. The signal and background efficiencies are calculated and can be found in tables 3.2 to 3.5. The normalisation of the number of events to a certain integrated luminosity and scaling up according to cross sections of the different backgrounds is shown in the *Results* chapter.

3.2.1 Overview of signal and background event samples

45000 signal events were generated using Pythia[1]. These were then passed through full detector simulation and reconstruction[2] using the standard ATLAS settings agreed for the Computing Systems Commissioning (CSC) exercises (and therefore accepted by the ATLAS offline community as being "good" for physics analysis).

For the background, a fully simulated sample of 50 000 $B_s \to J/\psi(\mu^+\mu^-)\phi(K^+K^-)$ events (events produced as part of the CSC exercises) were used. This exclusive decay channel is particularly likely to contaminate the B_c signal since it is quite probable for an accidental track in the same event to fit with the B_s to make a candidate particle that is easily mistaken for a real B_c. Two smaller background samples, of the inclusive decays $pp \to J/\psi$ and $b\bar{b} \to J/\psi X$, were also included in the analysis. The samples contain 10000 and 7000 events, respectively, and were not expected to contaminate the signal to any greater extent. These events also originated in the CSC exercises.

The reconstructed data (AOD) was then converted into AAna binaries for analysis.

3.2.2 Generation results and cross sections

To be able to normalize the results to a given integrated luminosity of the LHC, the cross sections of the different event samples are required. The production cross section can be taken from the outcome of the generation in PythiaB, but since the

[1]Generation carried out by Sergey Sivoklokov from Moscow State University

[2]Simulation and reconstruction carried out by James Catmore (Lancaster University) on the Lancaster HEP computing cluster and Björn H. Samset (University of Oslo) using NorduGrid

final sample must contain the decay channel in question, some adjustments must be made.

The $B_s \to J/\psi\phi$ background events

According to the PythiaB file from the generation of the $B_s \to J/\psi(\mu^+\mu^-)\phi(K^+K^-)$ background sample, the cross section for producing the B_s decays from the proton proton collision is 0.34 μb. The cross section includes the kinematic cuts $p_T(\mu_1) > 6$ GeV, $p_T(\mu_2) > 4$ GeV and $|\eta| < 2.5$ which in this case were selected by Pythia itself. Since the B_s is forced to decay according to a specific channel, the following branching factors must be factored in to obtain the cross section for the entire decay process. The values below are taken from [31].

- $B[B_s \to J/\psi\phi] = 9.3 \cdot 10^{-4}$
- $B[J/\psi \to \mu\mu] = 0.059$
- A factor of 2 for the symmetry correction, due to the fact that the opposite side quark is allowed to decay freely.

The final cross section for then becomes:

$$\sigma(B_s \to J/\psi(\mu^+\mu^-)\phi(K^+K^-)) = 3.73 \cdot 10^{-5}\mu b = 37.3 \text{ pb} \tag{3.1}$$

Notice that the branching ratio of the subdecay $\phi \to KK$ was not included in the calculation. This is because it was selected rather than forced, for the obvious reason that there will be many ϕ mesons not associated with the B-decay in the event, and these should not all be forced to decay to KK. The selection ensured that only the ϕs from the B_s tree were decaying to KK, and the rest were left to decay according to the Pythia internal tables.

The total number of events passing through the simulation and written to disk was 47 967.

Signal events

The final cross section for producing a B_c meson from a proton proton collision reported by Pythia was 180 nb (including the kinematic cuts of $p_T(\mu) > 6,4$ GeV and $|\eta| < 2.5$) [32]. This must be multiplied by 0.0277 to account for the BSignalFilter, which yields the B_c production cross section:

$$\sigma(pp \to B_c) = 180 \text{ nb} \cdot 0.0277 = 4.986 \text{ nb} \tag{3.2}$$

To obtain the cross section for the entire decay process, equation 3.2 is multiplied by the branching factor for the decay $B_c^+ \to B_s\pi^+$ together with the factors

for the B_s subdecay in the list above. The branching factor for $B_c^+ \to B_s\pi^+$ has not yet been experimentally examined, but according to [6] it should lie in between $[4-17]\cdot 10^{-2}$. In another article [14] the branching ratio is estimated to approximately $5\cdot 10^{-2}$, which is the number used for the calculations in the main part of this thesis (for results with other values of the branching ratio see chapter 4.2.3). The final result of the cross section is shown in equation 3.3.

$$\sigma(B_c^+ \to B_s(J/\psi(\mu^+\mu^-)\phi(K^+K^-))\pi^+) = 0.027\,\text{pb} \tag{3.3}$$

The final number of signal events written to disk was 43 847.

The $pp \to J/\psi$ and $b\bar{b} \to J/\psi X$ background events

The cross section from PythiaB for the smaller background samples are (according to [24]) 22 nb for $pp \to J/\psi X$ and 11 nb for $b\bar{b} \to J/\psi X$. The cross sections do not need to be corrected with any branching factors since the decays are inclusive and do not involve any forcings of decay processes.

3.2.3 Searching for the decay channel $B_c^+ \to B_s(J/\psi(\mu^+\mu^-)\phi(K^+K^-))\pi^+$: general description

Once the events have been simulated and digitized the data appears as if it were real hits from the ATLAS detector. The most interesting particles decay in the beam pipe and cannot be directly observed. One must therefore study their decay products, which can be seen in the detector, and try to deduce their parentage from their kinematic properties. In the case of the $B_c^+ \to B_s(J/\psi(\mu^+\mu^-)\phi(K^+K^-))\pi^+$ channel, the detectable particles are the kaons, the muons and the pion. Each part of the decay must be found by attempting to fit the tracks to common vertices, ultimately leading to the reconstruction of the parent particle.

Since ATLAS lacks charged hadron identification, the initial selection of B_c^+ candidates must be made based on the momenta and charge of the particle tracks. Also the use of vertex position and invariant mass is essential in rejecting background events. The high quality muon spectrometer in ATLAS enables the reconstruction software to match the muon identification with the hits in the inner detector. Decays including a muon pair are therefore especially convenient in the reconstruction procedure and for finding the $J/\psi \to \mu\mu$ part of the decay only those tracks matched with muon chamber hits are used.

In general terms the search proceeds per event as follows:

– Two oppositely charged muons are searched for, which are consistent with having arisen from the same decay point and having an invariant mass close to that of the J/ψ meson.

– The two tracks from these muons are then combined with two other oppositely charged tracks which are taken to be kaons. These are fitted to a common vertex (constraining the two muon tracks to the mass of the J/ψ) and for successful fits, if the invariant mass is consistent with that of the B_s meson, the four tracks are taken to be from the decay of the B_s via the J/ψ and ϕ (due to the very short lifetime of the J/ψ and ϕ the assumption is made that the tracks come from a single point).

– The B_s candidate is fitted with another positive track, with the resulting vertex constrained to point at the primary vertex. If the two are consistent with having arisen from the same decay point, and the invariant mass is close to the B_c^+ table mass, the five tracks are taken to be from the decay of a B_c^+.

A schematic sketch of the the entire decay channel, emerging from the primary vertex, is shown in figure 3.4.

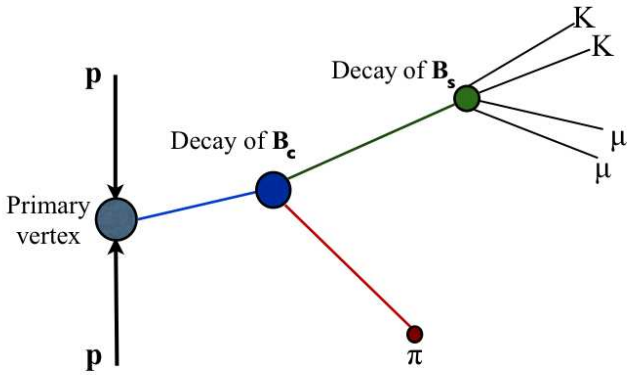

Figure 3.4: The decay chain from the initial proton proton collision to the detectable signal particles.

The following essential quantities need to be acquired from the physics analysis, such as to calculate the number of signal events that can be expected from ATLAS once data taking has begun.

– **Resolution** σ for the mass and lifetime of the reconstructed particles.

– **Event reconstruction efficiency** ε_{reco}, representing the proportion of the number of successfully reconstructed signal or background events identified by the comparison with the associated Monte Carlo truth.

– **Background rejection**, which is the ratio of the background events successfully rejected by the analysis algorithm. There are two types of background events:

– **Real background** implies actual events consisting of a decay process other than the signal decay. The real background used for this thesis are the three decays: $B_s \rightarrow J/\psi(\mu^+\mu^-)\psi(K^+K^-)$, $pp \rightarrow J/\psi X$ and $b\bar{b} \rightarrow J/\psi X$. These events can accidentally be identified as signal events, due to similarities in for instance vertex location and mass distributions.

– **Combinatorial background** is an inevitable feature in the analysis program. When the signal decay is reconstructed, certain combinations of tracks separate from the signal process can return physical similarities to the true signal decays. In the case of the B_c decay channel, it is probable that another track that is not a pion, is fitted with the B_s meson, since the decay of the B_c occurs at such short distances from the primary vertex. It is then important to try to minimize the number of these "fake" events.

3.2.4 Analysis procedure

The analysis procedure takes place in two large steps. Firstly the events are processed using AAna. At this stage only the lightest of cuts are applied, leading to a very large number of B_c candidates. In the second stage the resulting n-tuples are inspected in ROOT and the cuts are optimized to obtain the best significance. The second stage is deferred until later; this part concerns the bulk processing.

AAna analyses are designed such that the work of finding specific parts of the decay is factored out into algorithms. These are essentially C++ methods which return candidates for a specific decay, given the input data and user parameters such as mass hypotheses and allowed ranges for various values. The analysis program, which is compiled and run, consists of a list of "calls" to these algorithms, and then a set of loops which construct the analyst's n-tuple.

In this case three analysis algorithms are used which find the J/ψ, B_s^0 and B_c^+. These are now described in detail, along with the cuts applied at this stage.

The $B_s \rightarrow J/\psi\phi$ analysis algorithm

Before embarking on the main topic of this thesis, an exercise was carried out to reconstruct $B_s \rightarrow J/\psi\phi$ candidates, which make up the secondary decay after the B_c itself has decayed. The AAna algorithm for finding $B_s \rightarrow J/\psi(\mu^+\mu^-)\phi(K^+K^-)$ decays is called *Bs2Jpsiphi*. It requires the *JpsiFinder*, which searches for $J/\psi \rightarrow \mu\mu$ events, to have been called first. The program *analyzeBs2JpsiPhi* (constructed by the author of this thesis) makes use of these two algorithms and creates the n-tuples containing the final B_s^0 candidates - see Appendix A.1 for the C++ code itself.

The analysis program proceeds as follows:

1. The *analyzeBs2JpsiPhi* program calls the the *JpsiFinder* to find the $J/\psi \rightarrow \mu\mu$ candidates. In the JpsiFinder, all oppositely charged tracks are assumed to be muon pairs. A common vertex is built from the two tracks with the constraint that the χ^2 per degree of freedom ($\chi^2/d.o.f$) of the vertex must be less than 25, otherwise the pair is rejected.

2. The *analyzeBs2JpsiPhi* program is then able to plot the invariant mass, transverse momentum and χ^2 of the J/ψ candidates ($M_{J/\psi} = 3096.9MeV$ and $M_\mu = 105.6MeV$ according to [31]). The histograms can be seen in figure 3.5 and 3.6. A comparison between the decay products of the J/ψ candidates and the muons in the associated Monte Carlo truth is made. The results are shown in table 3.2.

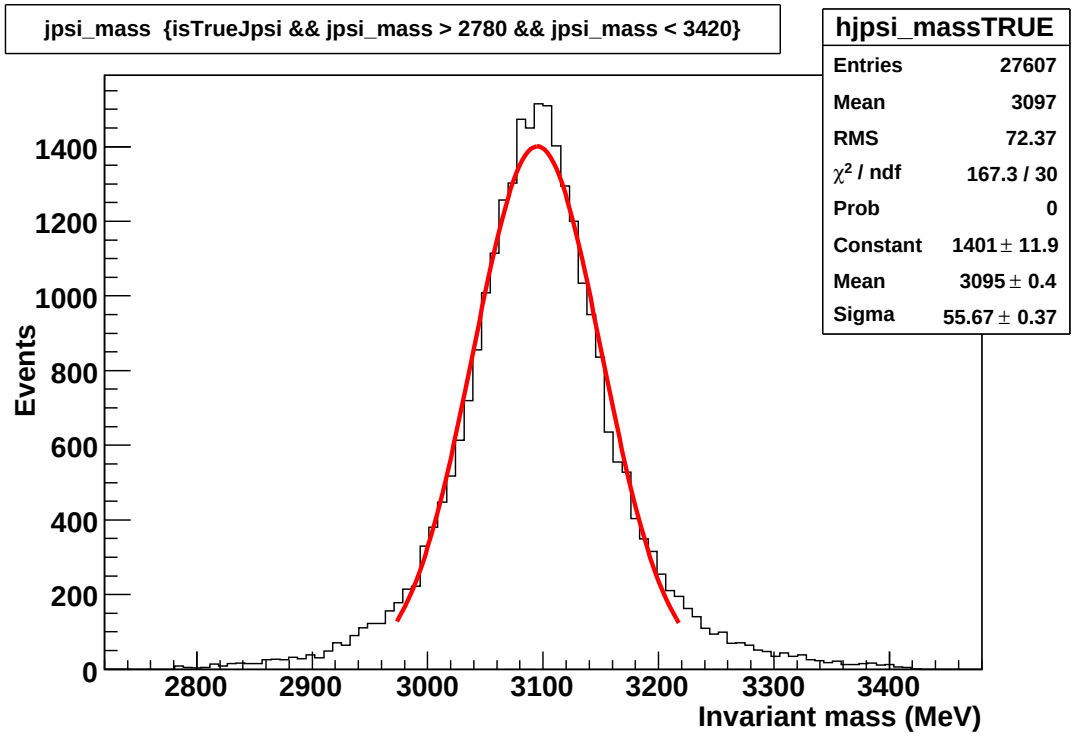

Figure 3.5: A histogram, made in ROOT, illustrating the invariant mass of the reconstructed J/ψ candidates, defined true by the Monte Carlo association. The fit is a Gaussian truncated at 10% of the peak value, providing a mass resolution of $\sigma = 55.7$ MeV.

3. Using the algorithm *Bs2Jpsiphi*, the B_s candidates are reconstructed. The algorithm extracts the J/ψ candidates from the *JpsiFinder* and applies several cuts. Initially, the invariant mass of the muon pairs is calculated and those with mass outside the range of $M_{J/\psi} \pm 150.0$ MeV are discarded.

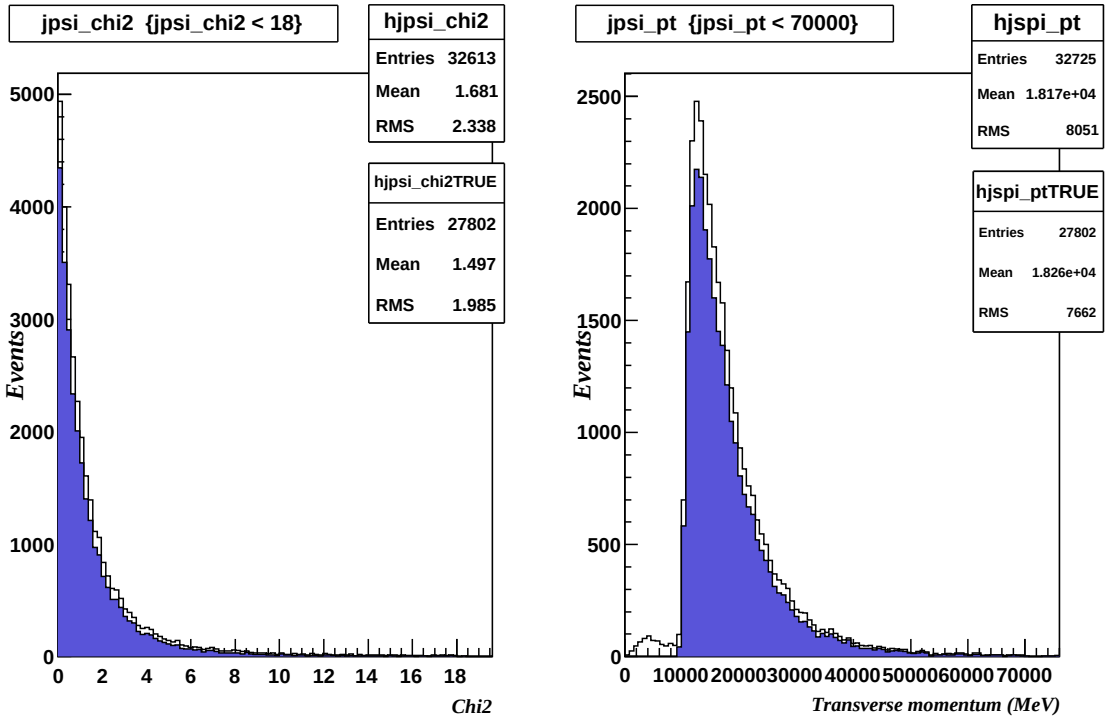

Figure 3.6: The χ^2 to the left and p_T to the right, of the J/ψ candidates. The filled plots illustrates the true J/ψ candidates, while the unfilled plots show all reconstructed J/ψ candidates, also including the combinatorial background.

4. Still in the *Bs2Jpsiphi* algorithm, the muon pair selected is then fitted with a kaon pair. All tracks are looped over once again to find an oppositely charged pair of particles. Tracks that were previously identified as muons are rejected. The invariant mass of the kaons is calculated and only accepted if it lies within the range of $M_\phi \pm 3.0\sigma_\phi$ ($M_\phi = 1019.5$ MeV, $M_{K^\pm} = 493.7$ MeV [31] and $\sigma_\phi = 7.0$ MeV).

5. Due to the very short decay length of both the J/ψ and the ϕ particle, all four signal tracks found by the algorithm are fitted to the common decay vertex of the B_s meson. Requirements are imposed that the vertex must have a $\chi^2/d.o.f < 25$ and that the momentum vector of the reconstructed B_s must point in the direction of the primary vertex.

6. Using the B_s candidates from *Bs2Jpsiphi*, the analysis program *analyzeBs2JpsiPhi* is then able to calculate different properties of the reconstructed B_s signal to create the following histograms: mass, lifetime, transverse momentum and χ^2 (see figures 3.7, 3.8 and 3.9). The associated Monte Carlo truth helps to identify which B_s candidates are really from the signal decay (which is necessary in order to be able to calculate different efficiencies - see table 3.2).

The outcome of the $B_s \to J/\psi\phi$ analysis algorithm can be found in table 3.2.

Analysis of $B_s \to J/\psi\phi$	**Number of events**
Total number of Monte Carlo events	47 967
Number of $J/\psi \to \mu\mu$ reconstructed	32 806
Number of true signal $J/\psi \to \mu\mu$ reconstructed	27 802 (58.0%)
Number of $B_s \to J/\psi\phi$ reconstructed	62 620
Number of true signal $B_s \to J/\psi\phi$ reconstructed	19 453 (40.6%)

Table 3.2: Track reconstruction efficiencies of the analysis program for the signal process $B_s \to J/\psi(\mu^+\mu^-)\phi(K^+K^-)$

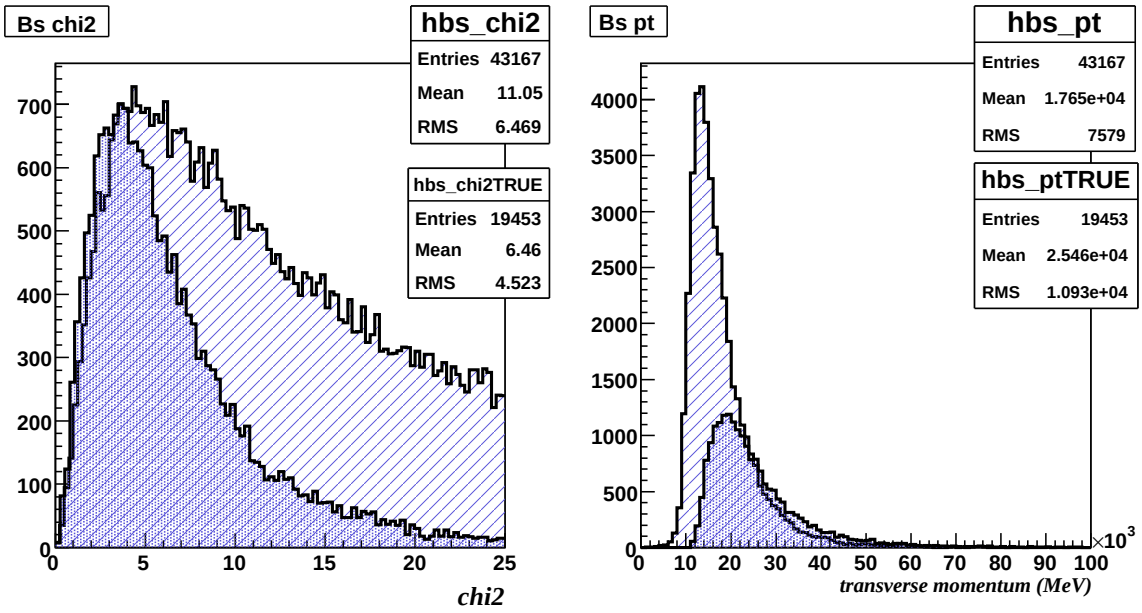

Figure 3.7: The χ^2 to the left and p_T to the right, of the B_s candidates. The more filled plots illustrates the true B_s candidates, while the striped plots show the combinatorial background.

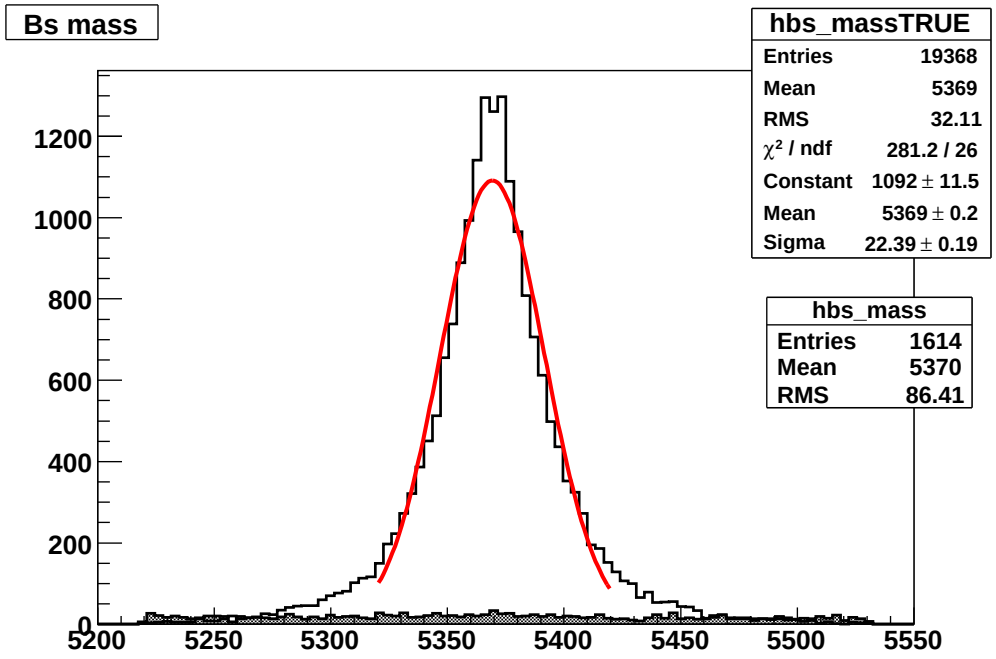

Figure 3.8: The mass distribution of the true B_s candidates, with the 1614 events from the combinatorial background overlaid. The fit of the true B_s events is a gaussian with resolution $\sigma = 22.39$ MeV.

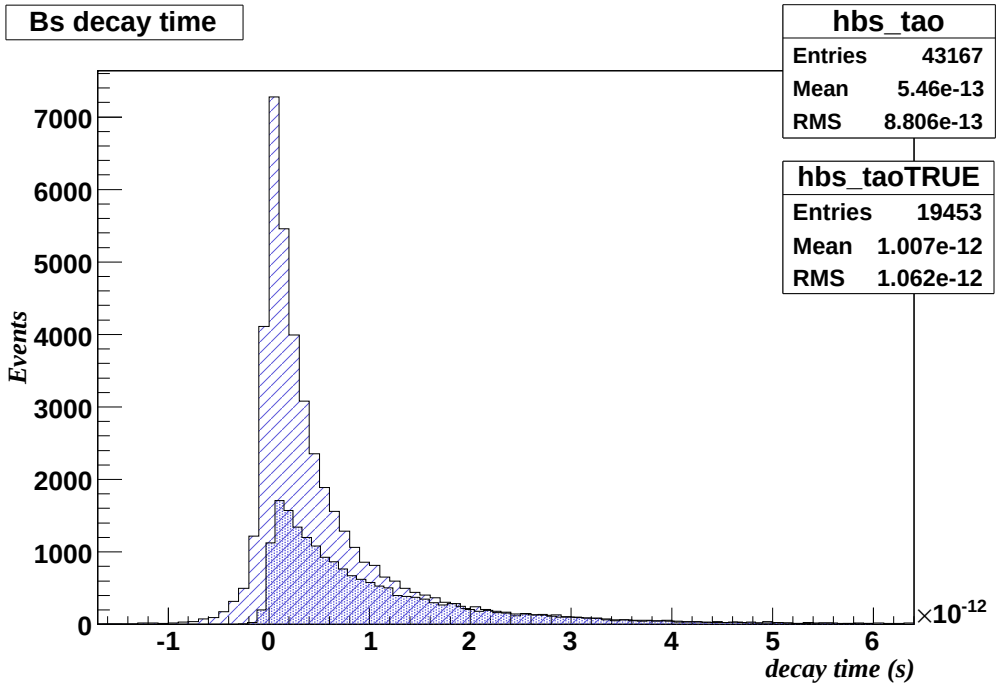

Figure 3.9: The proper decay time of the B_s candidates. The more filled plot illustrates the true B_s candidates, while the striped plot show the combinatorial background.

The $B_c \to B_s\pi$ analysis algorithm

As the main part of the thesis, the analysis algorithms for the signal $B_c^+ \to B_s(J/\psi(\mu^+\mu^-)\phi(K^+K^-))\pi^+$ was constructed. This was made by using the algorithm *Bs2JpsiPhi* to extract the $B_s \to J/\psi\phi$ decay and then further fit the B_s meson with a detected pion.

1. A few modifications are made in the algorithm *Bs2JpsiPhi* to adjust it according to the $B_c \to B_s\pi$ signal process. Most importantly, the constraint that the B_s track must point towards the primary vertex, is removed. A mass cut on the B_s meson itself, 4500 MeV $< M_{B_s} <$ 6000 MeV, is added to the algorithm. In the analysis program *analyzeBc2BsPi*, the maximum χ^2 of the decay vertex of the B_s is set to 100.

2. The algorithm *Bc2BsPi*, which can be found in Appendix A.2, takes the B_s candidates from the *Bs2JpsiPhi* and applies additional cuts. B_s candidates with mass outside the region $M_{B_s}\pm$100 MeV are rejected ($M_{B_s} = 5367.5$ according to [31]). Also the selections $\chi^2/d.o.f. < 25.0$ and $p_T > 8.0$ GeV are carried out.

3. All tracks in the events are then scanned to find a potential pion. The tracks descending from the B_s, already identified as the kaons and muons, are excluded. Also pions with a transverse momentum less than 1.5 GeV are rejected ($p_T(\pi) > 1500$ MeV).

4. A common vertex is fitted between the B_s candidate and the pion, with the restriction that the χ^2 of the vertex must be less than 25.0. The new vertex is constrained to point at the primary vertex. Successful fits yield a new particle candidate, *i.e.* the B_c meson.

5. It is then possible for the program *analyzeBc2BsPi* (see Appendix A.3) to use the B_c candidates passing the *Bc2BsPi* algorithm, to calculate the mass, χ^2, impact parameter and proper decay time of the B_c meson. Also the open angle between the B_s and pion is calculated and plotted.

The resulting outcome of the $B_c \to B_s\pi$ analysis algorithms can be found in table 3.3.

Due to the short decay length of the B_c meson, the decay vertex is situated relatively close to the primary vertex. This causes a problem in that the B_s is fitted with a large number of tracks in *Bc2BsPi*, as a result of the large density of tracks close to the primary collision. This explains the very large number of combinatorial background events in table 3.3. Making appropriate cuts to the B_c events is therefore of great importance.

Analysis of $B_c \to B_s\pi$	**Number of events**
Total number of Monte Carlo events	43 847
Number of $B_s \to J/\psi\phi$ reconstructed	47 645
Number of true signal $B_s \to J/\psi\phi$ reconstructed	29 103 (66.4%)
Number of $B_c \to B_s\pi$ reconstructed	226 717
Number of true signal $B_c \to B_s\pi$ reconstructed	15 520 (35.4%)

Table 3.3: Track reconstruction efficiencies of the analysis program for the signal process $B_c \to B_s(J/\psi(\mu^+\mu^-)\phi(K^+K^-)\pi)$

3.2.5 Optimization through selection cuts

To improve the signal reconstruction efficiency and also minimize the number of real and combinatorial background, further selection cuts are made on the events passing the analysis programs. Exactly which cuts are chosen is determined by optimizing the ratio between the number of signal events and total number of reconstructed events, including both signal and combinatorial background (see equation 3.4). In other words, the purpose of the cuts is to reject as much background as possible, without loosing too many signal events.

$$\frac{N_{signal}}{N_{total}} = \frac{N_{signal}}{N_{signal} + N_{background}} \tag{3.4}$$

Plots of different parameters of the B_c signal and background were made to assist in determining which cuts should be applied. The best cut is on the opening angle between the B_s and the pion, since it is expected that they will be moving in roughly the same direction as they originated from the same high-momentum body (see figure 3.10). The other selection cuts are not as effective since the signal and background distributions are much more similar (see figure 3.11, 3.12, 3.13 and 3.14). Cuts on the transverse momentum of the signal pion are also highly effective (see figure 3.15).

In conclusion, the final selection cuts in the analysis are:

1. A selection on the B_s mesons identified as decay products of the B_c is made by the mass requirement: 5300 MeV $< M_{B_s} <$ 5440 MeV (see figure 3.8).

2. The B_c mesons with a transverse momentum less than 8550 MeV are rejected: $p_T > 8.55$ GeV (see figure 3.11).

3. The χ^2 of the B_c must be less than 12.0: $\chi^2/d.o.f < 12.0$ (see figure 3.12).

4. The lifetime of the B_c meson must lie in the range: -4 ps $< \tau_{B_c} < 11.4$ ps (see figure 3.13).

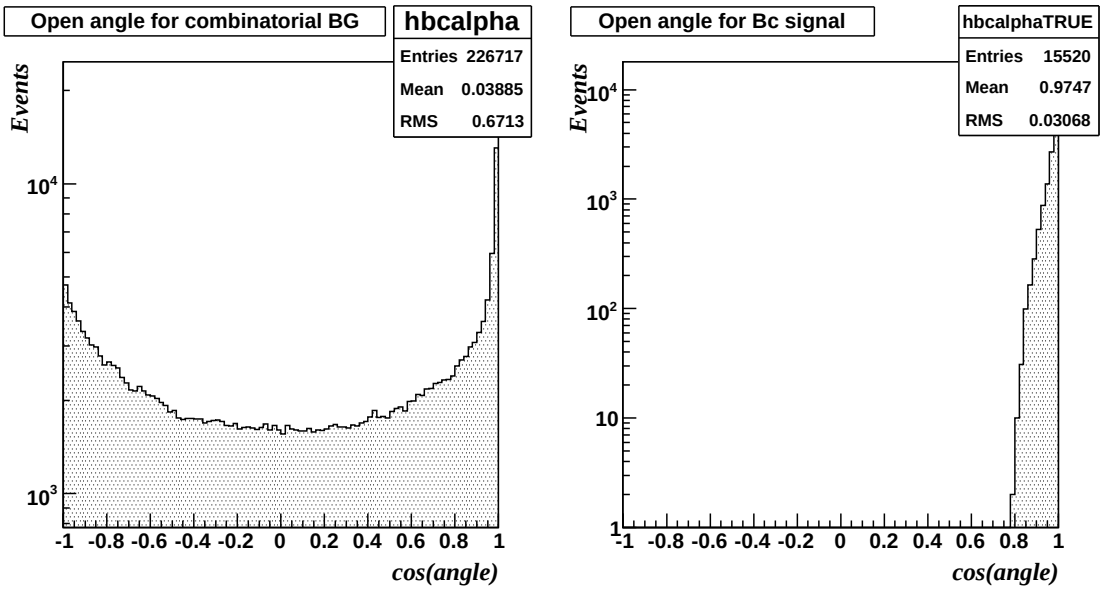

Figure 3.10: The opening angle between the B_s meson and the pion for the combinatorial background (left) and the signal (right) in logarithmic scale. It is evident that a cut around 0.8 will remove the majority of the background events without damaging the signal to any great extent.

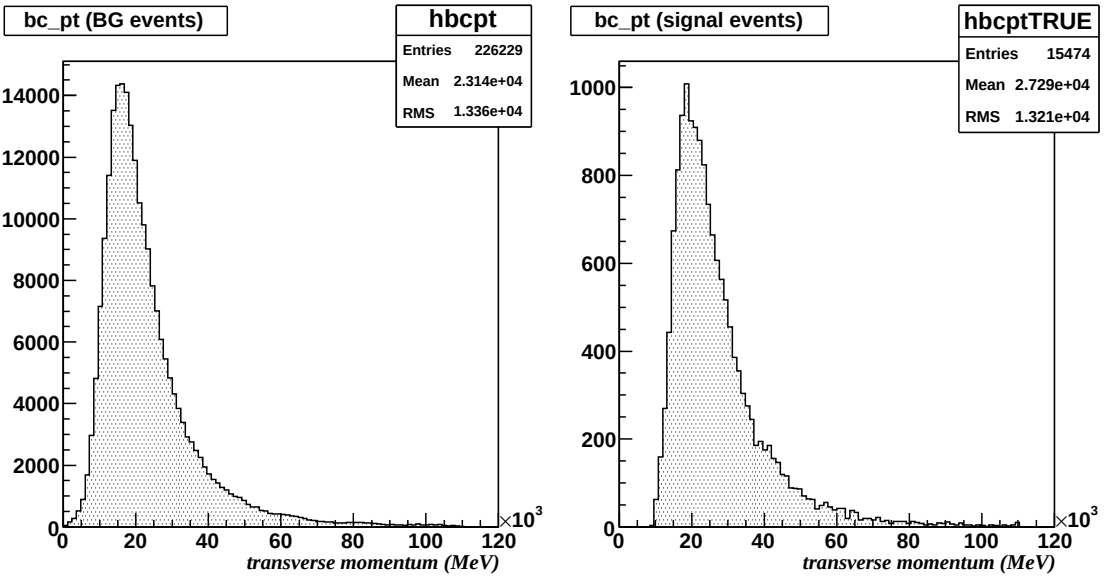

Figure 3.11: Histograms of the transverse momentum of the combinatorial background (left) and signal events (right).

5. The cosine value of the open angle between the B_s meson and the pion is cut off at 0.84: $\cos(\alpha) > 0.84$ (see figure 3.10).

6. The impact parameter significance of the B_c (the impact parameter divided

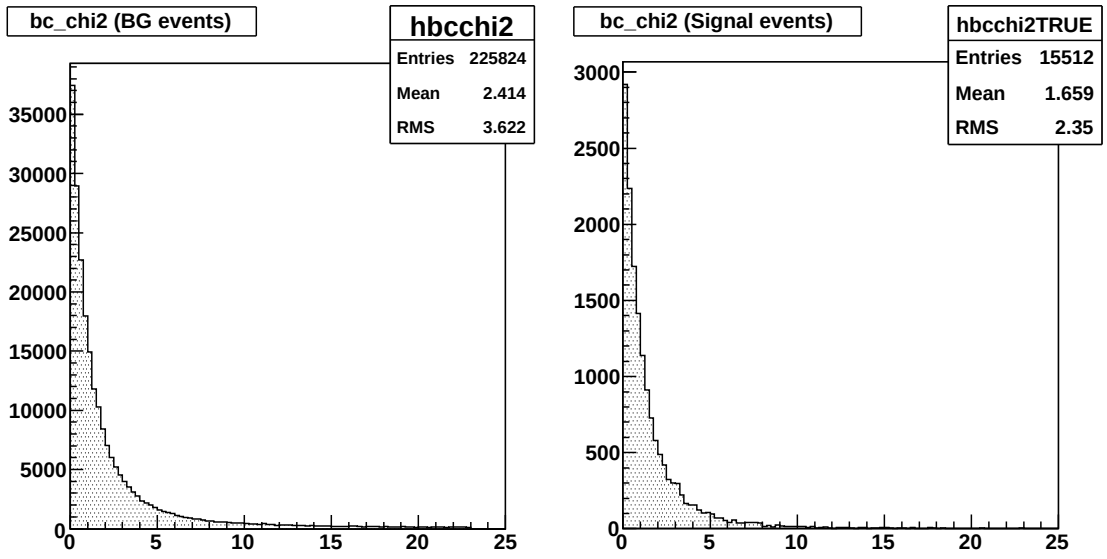

Figure 3.12: Histograms of the χ^2 of the combinatorial background (left) and signal events (right).

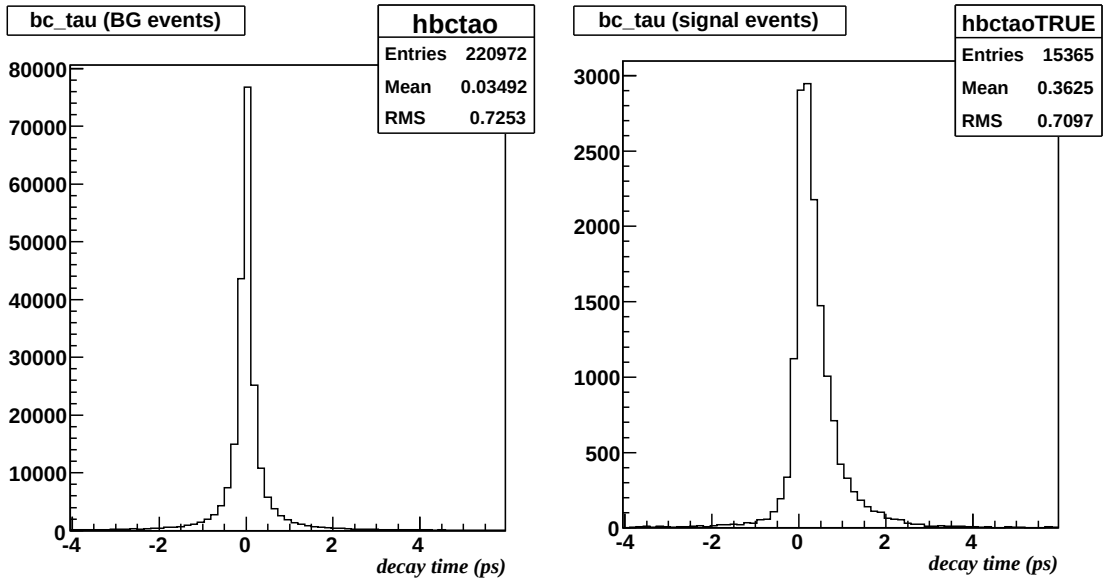

Figure 3.13: Histograms of the decay time of the combinatorial background (left) and signal events (right).

by the uncertainty) is restricted according to:
impact parameter significance < 5.0 (see figure 3.14).

7. The transverse momentum of the pion is required to be above 2.0 GeV: $p_T(\pi)$ >2.0 GeV (see figure 3.15).

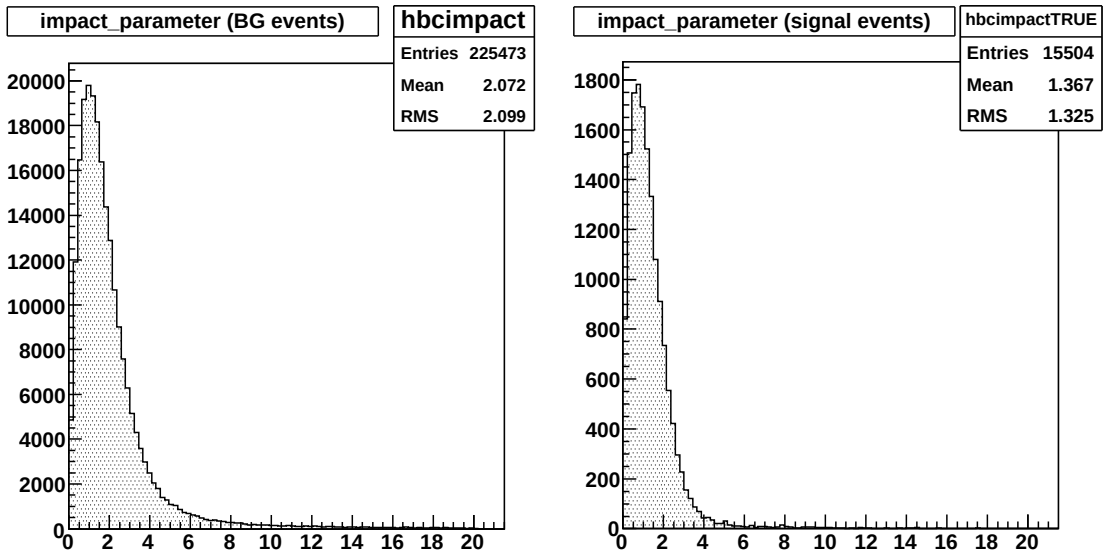

Figure 3.14: Histograms of impact parameter significance (impact parameter/uncertainty) of the combinatorial background (left) and signal events (right).

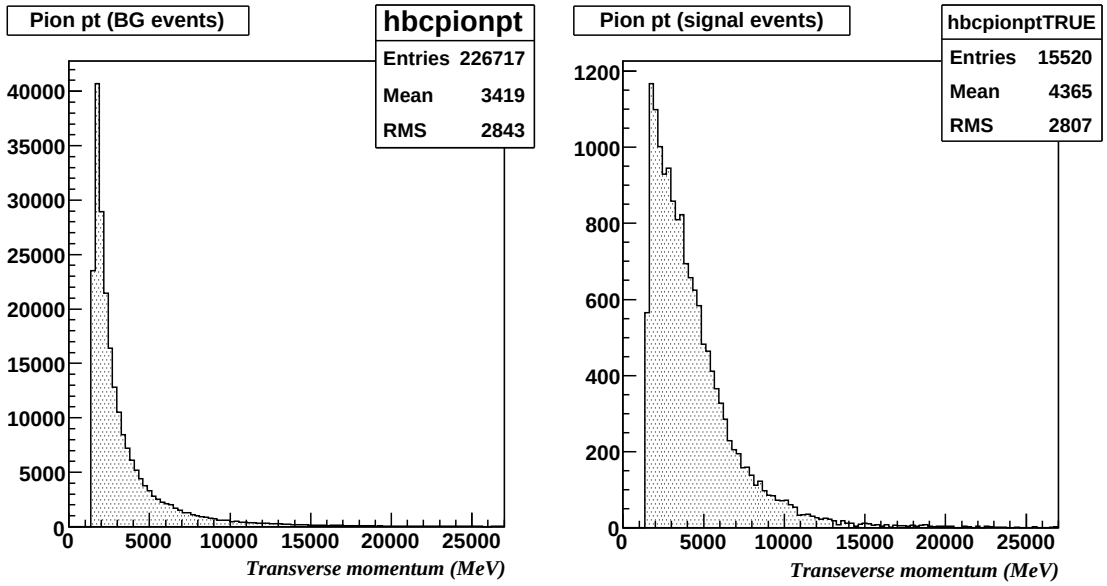

Figure 3.15: Histograms of the transverse momentum of the signal pion for the combinatorial background (left) and signal events (right).

8. Finally, the mass of the B_c meson itself is restricted to:
   6225 MeV $< M_{B_c} <$ 6375 MeV.

The results for the selection cuts and the effect on the number of events of both background and signal can be found in table 3.4.

Selection cut	B_c **signal accepted**	**Combinatorial** B_c **accepted**
None	15 520	226 717
5300 MeV $< M_{B_s} <$ 5440 MeV	15 021	215 526
$p_T > 8.55$ GeV	15 021	208 964
-4 ps $< \tau_{B_c} <$ 11.4 ps	14 872	203 755
$\chi^2 < 12.0$	14 729	196 041
$\cos(\alpha) > 0.84$	14 688	33 456
impact parameter significance < 5.0	14 511	31 866
$p_T(\pi) > 2.0$ GeV	12 497	21 835
6225 MeV $< M_{B_c} <$ 6375 MeV	12 181	1534

Table 3.4: A summary of the selection cuts and their effect on reducing the number of signal and combinatorial background events.

To determine the number of mistakenly reconstructed real background events that pass through the analysis algorithms, the event samples containing the three background processes are run through the same code that was used for the B_c signal analysis. The results, after gradually applying identical selection cuts that were implemented on the signal events, are shown in table 3.5.

Selection cut	$B_s \to J/\psi\phi$	$pp \to J/\psi X$	$bb \to J/\psi X$
None	236 675	770	908
5300 MeV $< M_{B_s} <$ 5440 MeV	224 901	459	610
$p_T > 8.55$ GeV	223 218	452	586
-4 ps $< \tau_{B_c} <$ 11.4 ps	217 746	441	575
$\chi^2 < 12.0$	210 382	433	562
$cos(\alpha) > 0.84$	28 813	68	139
impact parameter significance < 5.0	27 912	67	123
$p_T(\pi) > 2.0$ GeV	16 856	37	93
6225 MeV $< M_{B_c} <$ 6375 MeV	991	2	7

Table 3.5: The table reflects the outcome of the three different real background samples that were run through the $B_c \to B_s\pi$ analysis algorithm. Also the identical selection cuts as for the B_c signal, are applied to the background events.

Identifying the signal events through the Monte Carlo association is clearly not possible in the case of real data. Consequently, the procedure of determining cuts must be done sing simulated events. Only then, after the cuts have been determined, can the signal resolution and the final reconstruction efficiencies be calculated.

Chapter 4

Results

4.1 Event reconstruction

4.1.1 Mass resolution of the B_c signal

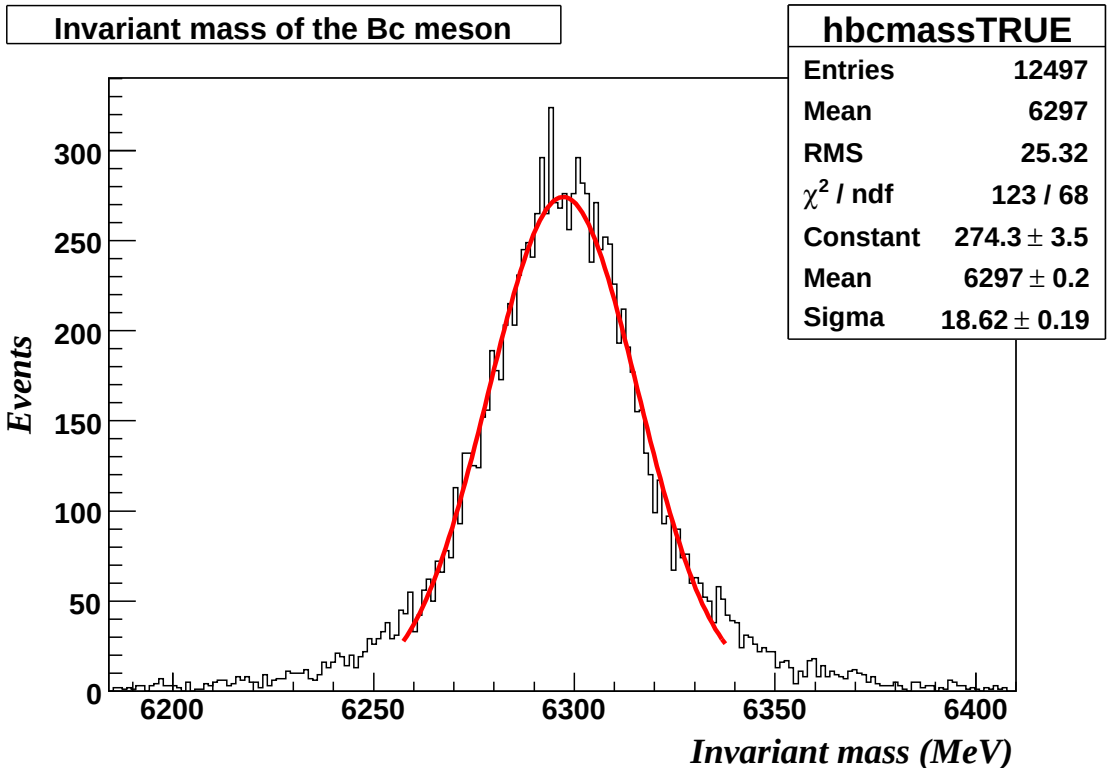

Figure 4.1: The mass distribution of the B_c candidates confirmed to be Monte Carlo signal processes and passing the selection cuts. The fit is a gaussian with resolution $\sigma = 18.6$ MeV.

The results of the reconstructed invariant mass of the 12 497 signal B_c mesons accepted by the cuts described in the previous chapter (except for the mass cuts on the B_c mass itself) is shown in figure 4.1. A single gaussian function, truncated at

10% of the peak, is fitted to this histogram yielding a resolution of 18.62 MeV. The peak value equals 6297 MeV, which is well consistent with the simulation input of 6300 MeV.

4.1.2 Decay time resolution of the B_c signal

Obtaining a resolution for the decay time distribution of the B_c mesons is not as straightforward as for the invariant mass distribution. The decay time distribution is a convolution of an exponential (the lifetime) and a gaussian (the detector resolution). The gaussian contribution also explains why some B_c candidates have a negative lifetime, which is due to the broadening of the signal (see figure 4.2).

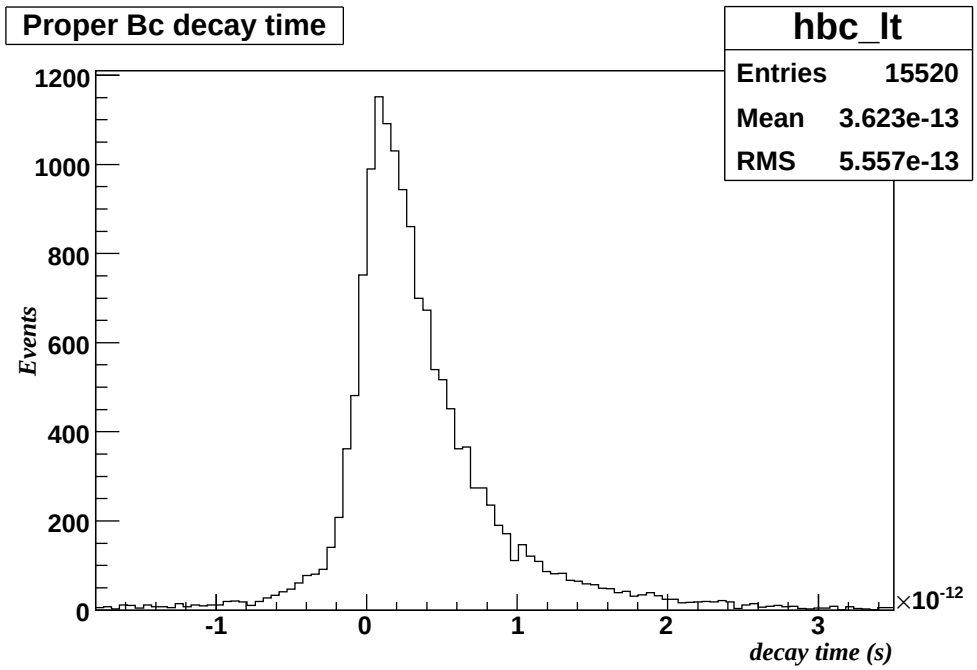

Figure 4.2: The proper decay time of the signal B_c mesons, passing the analysis algorithm.

The resolution of the decay time measurement can be obtained by taking a simple mean of the calculated uncertainty on the proper decay time of each candidate (calculated by AAna from the vertex position covariance matrix). The result acquired is shown in figure 4.3, with the lifetime resolution of $\sigma = 0.187$ ps.

Finally, to calculate the mean lifetime of the B_c meson, an exponential distribution is fitted to the events with positive decay time values (see figure 4.4). The lifetime is then obtained by calculating the inverse of the slope provided by the exponential fit (see box in figure 4.4), which returns a lifetime of $\tau = (2.13 \cdot 10^{12})^{-1} = 0.469 \cdot 10^{-12} s$. The value is consistent with the simulation input of 0.46 ps.

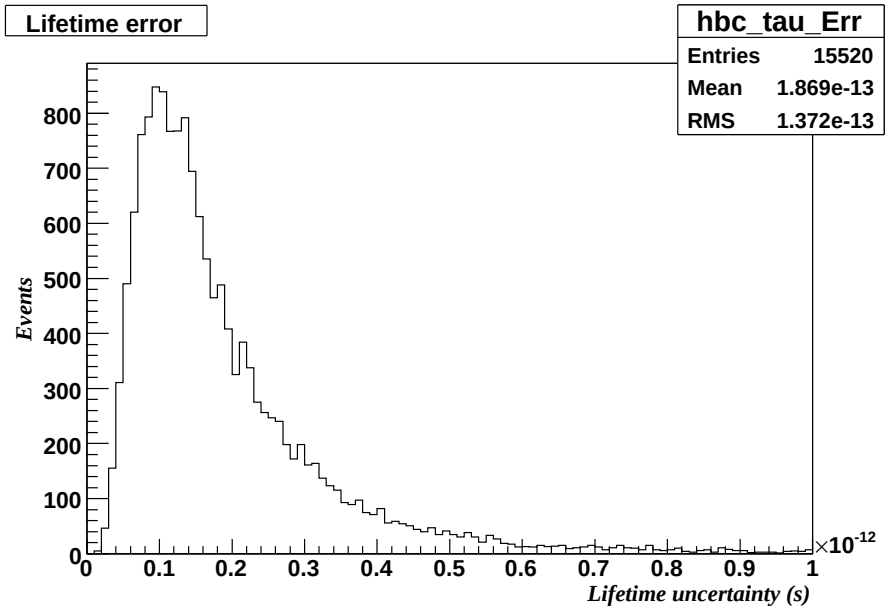

Figure 4.3: The histogram shows the uncertainty of the decay time calculations, which provides a lifetime resolution of $\sigma = 0.187$ ps.

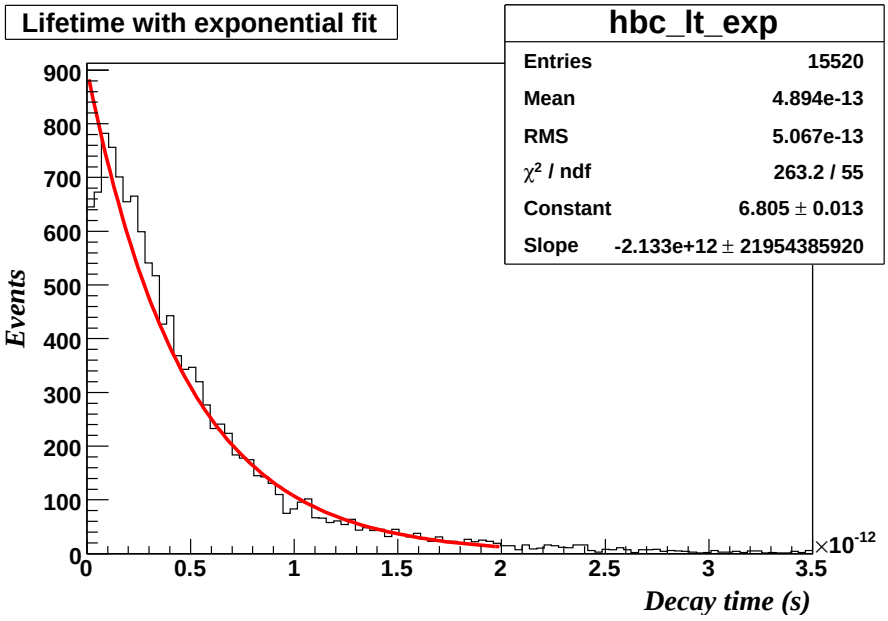

Figure 4.4: The B_c candidates with a positive decay time are included in this plot in order to fit an exponential distribution. The mean lifetime is obtained by inverting the slope of the exponential fit, which returns the value $\tau_{B_c} = 0.469$ ps.

4.1.3 Results including background

In order to be able to compare the signal and the background events that pass through the physics analysis and selection cuts, the signal and background are

both plotted in the same histogram. In figure 4.5, the signal and the backgrounds are superimposed (overlaid) and the resulting number of events for each sample that passes the cuts can be found in the boxes. In figure 4.6 the signal and the different backgrounds can be piled on top of each other (stacked), which is the case for the histogram in figure .

Notice that the resulting number of events from the analysis reflects on the performance but it does not give any consideration to what the results from real ATLAS data would correspond to. The number of events in figure 4.5 and 4.6 are directly related to the original sample sizes of the simulated events. For the histogram to provide a more realistic view of the analysis, the background plots must be scaled up according to cross section and properly normalized to an integrated luminosity (see chapter 4.2.1).

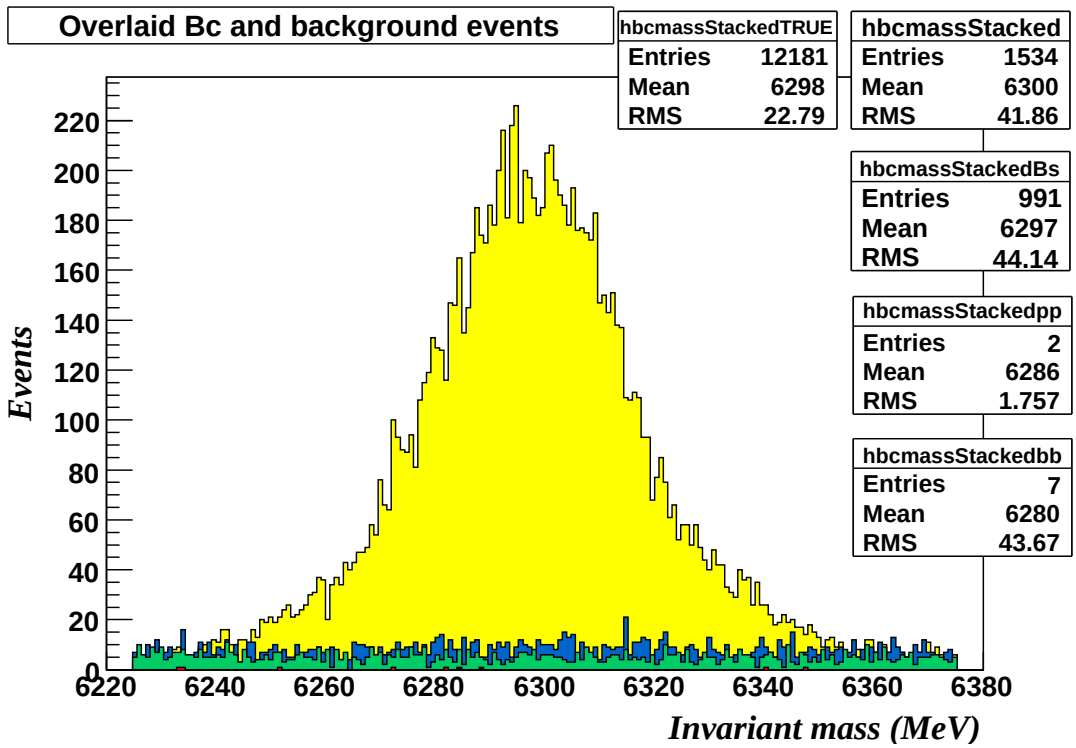

Figure 4.5: The invariant B_c mass including overlaid background plots. The yellow plot represents the signal, the blue plot the combinatorial background and the green plot represents the real $B_s \rightarrow J/\psi\phi$. The two inclusive backgrounds are colored in red and black but contain too few events to be seen clearly.

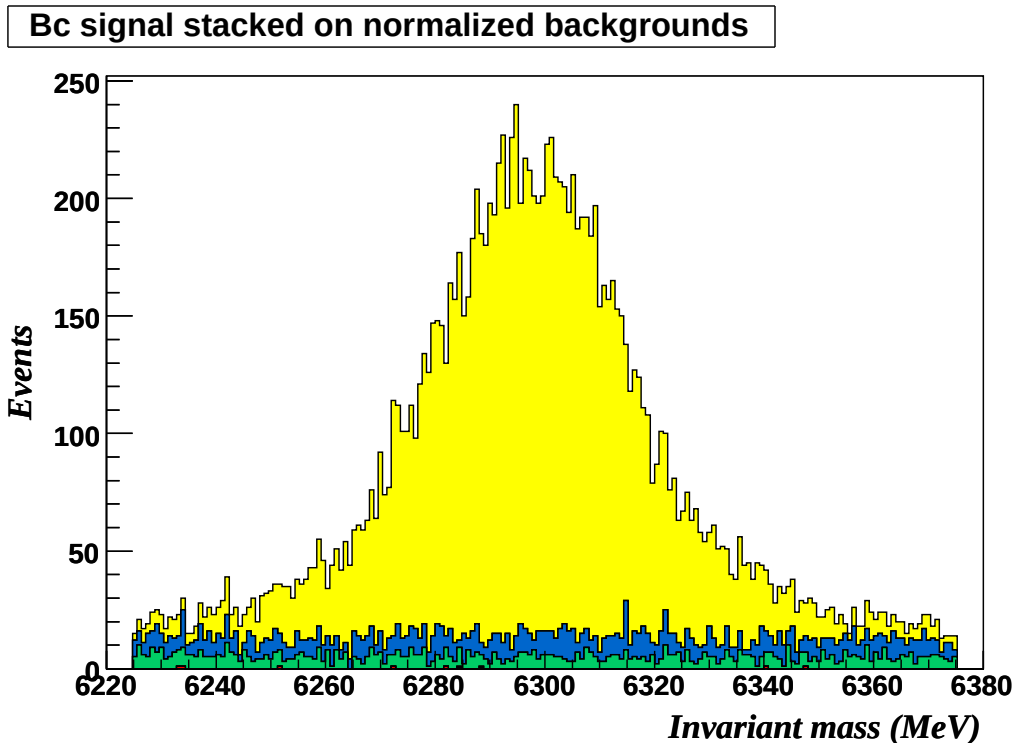

Figure 4.6: The figure is equivalent to figure 4.5 except for that the invariant B_c mass is stacked on top of the combinatorial and real background events in order to more appropriately reflect the total number of processed events.

4.2 Normalized results

4.2.1 Efficiencies and normalization

For the results to be realistic, the number of events must be scaled up to show how many events can be expected for an LHC run of a certain luminosity. The resulting numbers of events of the signal and background processes, which pass the analysis and selection cuts, are used to calculate the different reconstruction efficiencies, ε_{reco}. This value is then applied, together with the cross sections of the decay channels, to normalize the results in order to calculate the expected number of events for a certain integrated luminosity (during one year at the LHC). The equation for normalizing the results is:

$$N_{events} = \int Ldt \cdot \sigma\varepsilon_{reco} \tag{4.1}$$

The reconstruction efficiencies for the signal and background processes are easily obtained by taking the number of accepted events from table 3.4 and 3.5 and dividing the values by the original numbers of simulated events. See table 4.1 for an overview.

Physical process	**Simulated events**	**Resulting nbr of events**	**Reconstruction efficiency**
Signal $B_c \to B_s\pi$	43 847	12 181	27.8%
Combinatorial $B_c \to B_s\pi$	43 847	1 534	3.50%
$B_s \to J/\psi\phi$	47 967	991	2.07%
$pp \to J/\psi X$	10 000	2	0.02%
$bb \to J/\psi X$	7 000	7	0.1%

Table 4.1: The resulting number of events passing the analysis algorithm and selection cuts divided by the number of simulated events yields the reconstruction efficiencies for the signal and background processes.

The cross sections calculated in chapter 3.2.2 for the signal and background decays (equations 3.2 and 3.3) can be inserted into equation 4.1, together with the reconstruction efficiencies calculated in the table above. The accepted $b\bar{b} \to J/\psi$ were all found to be $B_s \to J/\psi\phi$ events and they are therefore omitted to avoid double counting. The other inclusive background decay is not included since there are too few events passing the analysis to be able to make valid estimations.

A low luminosity run ($2 \cdot 10^{33}$ cm$^{-2}s^{-1}$) integrated during the time span of one effective year gives a value of approximately 20 fb^{-1}. The resulting normalized number of events expected for a low phase luminosity integrated over one year, are thus shown in table 4.2.

Physical process	**Reconstruction efficiency** (ε_{reco})	**Cross section** (σ)	**Normalized nbr of events**
Signal $B_c \to B_s\pi$	0.278	0.027pb	150
Combinatorial $B_c \to B_s\pi$	0.0350	0.027pb	19
$B_s \to J/\psi\phi$	0.0207	37.3 pb	15 442

Table 4.2: The right column shows the resulting number of events normalized to the integrated luminosity of $\int Ldt = 20$ fb^{-1}, according to equation 4.1.

4.2.2 Mass results with normalized number of events

To get a more realistic view of what real data produced by the ATLAS detector would appear as, the invariant mass histogram is replotted with the background scaled according to the normalized number of events in table 4.2. The large difference between the cross section for the signal channel and for the real background channel results in a scaling which is not proportional to the number of events generated in the simulation. In order to scale the background events without simulating an enormous sample of $B_s \to J/\psi\phi$ events, an exponential is fitted to the unscaled real background and then masses can be generated according to the parameters

of the exponential fit, using an simple accept-reject Monte Carlo program. The masses thus generated will then maintain the same distribution as the invariant mass of the original background sample. The results are shown in figure 4.7 and simulates the outcome of real data from the ATLAS detector, passed through the analysis algorithm made in this thesis.

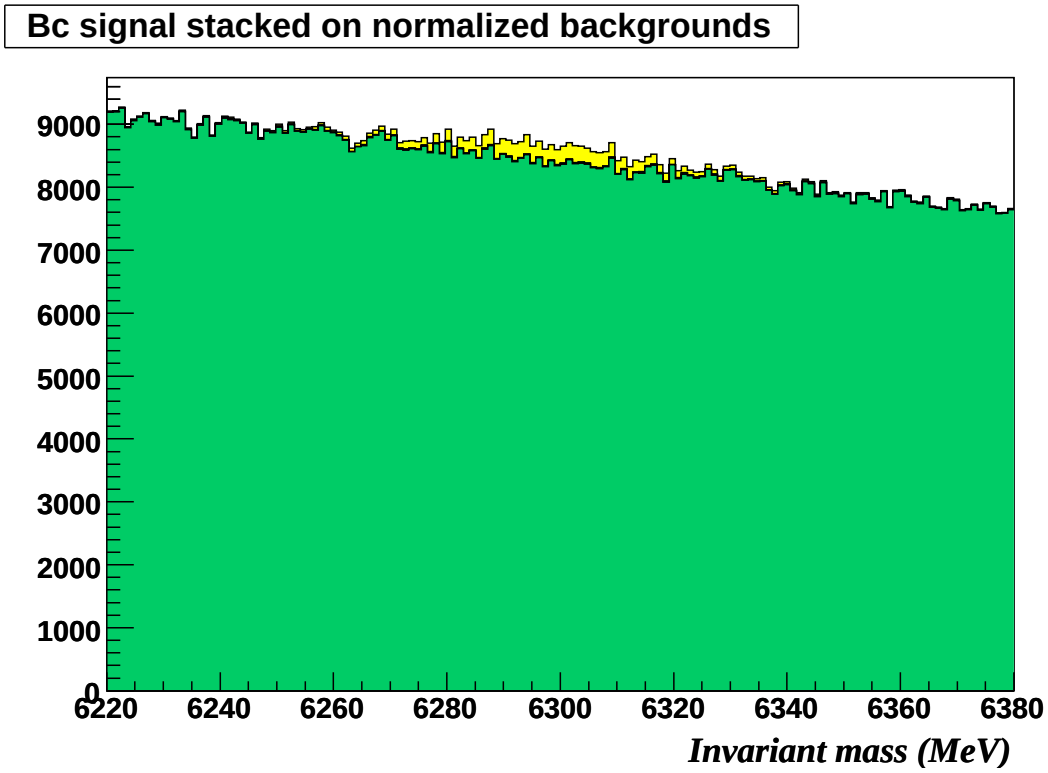

Figure 4.7: The histogram reflects the outcome of the analysis, after the selection cuts, with the $B_s \rightarrow J/\psi\phi$ background scaled according to cross section (green plot). The yellow distribution shows the signal stacked on top of the background. The combinatorial background events are too few to be seen clearly, but merely resemble a dark layer on top of the background.

4.2.3 Significance of the signal

For a signal to be considered statistically significant, the signal significance must be above the five-sigma (5σ) limit according to equation 4.2.

$$\frac{N_{signal}}{\sqrt{N_{Background}}} > 5 \tag{4.2}$$

The normalized number of signal and background events, within the 4σ mass region (σ = 18.6 MeV), reaches 150 for the signal and 19 + 15442 = 15461 for both the combinatorial and the $B_s \rightarrow J/\psi\phi$ background (see table 4.2). Inserting the

numbers into equation 4.2 yields a signal significance of 1.2σ. To improve the significance, the mass acceptance region can be decreased. Table 4.3 shows an overview of the calculated significance for four different mass cuts. It is evident that the significance of the $B_c \rightarrow B_s\pi$ signal does not reach the five-sigma limit.

Mass cut	N_{signal}	$N_{Background}$	**Significance**
$M_{B_c} \pm 4\sigma$	150	15 461	1.2
$M_{B_c} \pm 3\sigma$	146	11 524	1.4
$M_{B_c} \pm 2\sigma$	134	7 491	1.5
$M_{B_c} \pm 1\sigma$	98	3734	1.6

Table 4.3: The resulting signal significance for different mass acceptance regions ($\sigma = 18.6$ MeV).

The question is then - which luminosity is required to reach the 5σ limit with the $B_c \rightarrow B_s\pi$ signal? A simple calculation can be made to establish with which factor the statistics must increase to acquire a signal significance of 5σ (for the 1σ mass region).

$$\frac{x \cdot N_{signal}}{\sqrt{x \cdot N_{Background}}} = \frac{x \cdot 98}{\sqrt{x \cdot 3734}} > 5 \quad \Rightarrow \quad x > 9.7 \tag{4.3}$$

The statistics must therefore increase by a factor of 9.7 for the signal to be scientifically observable, which implies an integrated luminosity of approximately 200 fb^{-1}.

In chapter 3.2.2, the branching factor for the subdecay $B_c \rightarrow B_s\pi$ was assumed to be $5 \cdot 10^{-2}$, but this has not yet been thoroughly investigated. According to [6], the upper limit of the branching factor is $17 \cdot 10^{-2}$ and using this value in the same calculations as before returns a new estimate of the signal significance. Since the number of signal events depends linearly on the branching factor of the subdecay, the new number is trivially obtained by $N_{signal} \cdot 17 \cdot 10^{-2}/5 \cdot 10^{-2} = 98 \cdot 17/5 = 333.2$. The signal significance can then be recalculated according to equation 4.2: $333.2/\sqrt{3734} = 5.45\sigma$, yielding a result which is indeed above the 5σ limit.

In figure 4.8, some values of the branching factor for the subdecay $B_c \rightarrow B_s\pi$ are plotted as a function of integrated luminosity, under the assumption of a 5σ significance. The calculated values, required to reach the 5σ limit, ranges from $\int Ldt = 20$ fb^{-1} presuming a branching factor of $15.6 \cdot 10^{-2}$, to $\int Ldt = 200$ fb^{-1} with the corresponding branching factor of $5 \cdot 10^{-2}$. All values situated above the curve in the figure then returns a signal significance above 5σ.

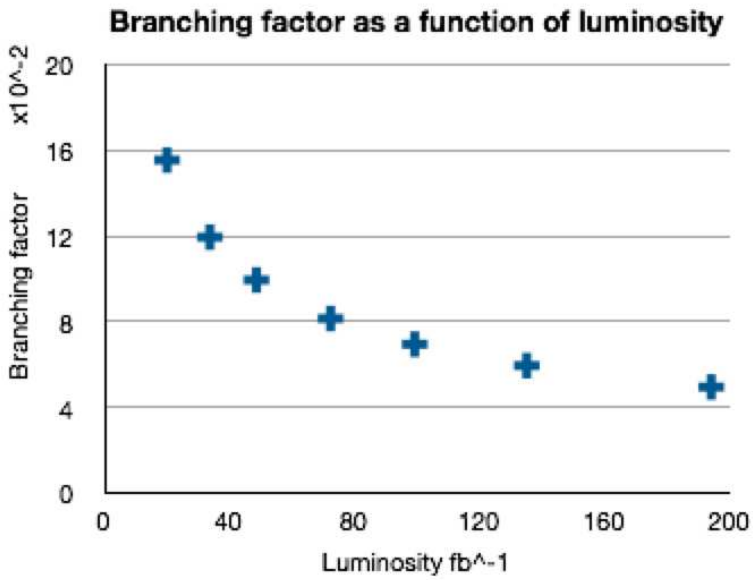

Figure 4.8: The minimum value of BR$[B_c \to B_s\pi]$ required to obtain a signal with 5σ significance, as a function of the integrated luminosity.

Chapter 5

Conclusion

The purpose of this thesis has been to analyze the decay process of $B_c^+ \rightarrow B_s(J/\psi(\mu\mu)\phi(KK))\pi^+$ and investigate the performance of the ATLAS detector through analyzing fully simulated and reconstructed Monte Carlo signal and background events. The initial work concerned the reconstruction of the more well-known decay channel $B_s \rightarrow J/\psi(\mu\mu)\phi(KK)$, to then further revise the analysis algorithm in order to fit the reconstructed B_s meson with a positive pion to obtain the secondary decay vertex of the B_c meson.

The analysis algorithm searching for fully reconstructed B_c^+ mesons succeeded in collecting approximately 35% of the simulated events, identified as signal by a comparison with the associated Monte Carlo truth. A large number of combinatorial and real background events are contaminating the signal and several selection requirements are therefore carefully chosen in order to improve the signal/background ratio. The final results for the reconstruction efficiencies after cuts are applied are: 28% for the B_c signal, 3.5% for the combinatorial background and 2.1% for the real exclusive $B_s \rightarrow J/\psi\phi$ background. Only a negligible small fraction of the inclusive background samples, $pp \rightarrow J/\psi X$ and $b\bar{b} \rightarrow J/\psi X$, succeeds in passing through the analysis and cuts.

The invariant mass and proper decay time distribution of the true B_c^+ mesons were reconstructed as follows:

$B_c \rightarrow B_s\pi$ **property**	**Resulting value**	**Resolution** (σ)
Invariant mass	(6297 ± 0.2) MeV	(18.62 ± 0.19) MeV
Average proper decay time	0.469 ps	0.187 ps

Table 5.1: Resulting properties of the reconstructed B_c^+ passing all requirement selections.

The obtained mass resolution can be compared to the results of the $B_c^- \rightarrow J/\psi\pi$ decay analysis at the CDF experiment (see [9]), where the best fit of the signal was

a Gaussian with $\sigma = 15.5$ MeV. Both the resulting mass and lifetime value in table 5.1 are consistent with the simulation input values (6300 MeV and 0.46 ps).

Normalizing the final number of signal and background events to an integrated luminosity of 20 fb^{-1}, gives rise to an overwhelming amount of $B_s \to J/\psi\phi$ background, due to the large difference in cross section for the signal and background decay process ($\sigma_{B_c \to B_s\pi} = 0.027$ pb and $\sigma_{B_s \to J/\psi\phi} = 37.3$ pb). The resulting normalized histogram displaying the invariant mass of the signal and B_s background can be found in figure 4.7, which shows that the small cross section for the B_c channel will cause difficulties in searching for this particular decay with the large amount of background that is expected for a hadronic collider. One way of dealing with this issue is to optimize the selection requirements with great precaution.

The contamination of the B_c signal by the large amount of real background events gives rise to the small signal significance estimated to lie between 1.2σ and 1.6σ for different mass acceptance regions. The values are then significantly smaller than the 5σ required to claim the signal to be detectable. The main explanation for the small significance is the exceptionally small cross section of the decay channel of the signal decay as mentioned above. Even though the probability of the B_c meson decaying to a B_s meson and a pion is reasonably large, the decay channel studied in this work is very specific with smaller branching factors for several subdecays that has to be included in the estimations. The branching factor for the subdecay $B_c^- \to B_s\pi$ has, however, not yet been precisely determined and the assumption of 5% made in this thesis might be too low. The signal significance was therefore reevaluated with a branching factor of 17% which returns a significance of approximately 5.5σ in the $\pm 1\sigma$ mass region. If this is indeed the case, then the signal would actually be observable in the ATLAS detector.

Another issue which concerns the B_c meson in general, is the high contamination of the signal, due to the small distance between the primary and secondary vertex. This problem is, however, primarily solved by carefully choosing appropriate selection cuts.

It is important to remember that this is a first study of this particular decay channel and there are probably other possibilities for optimizing the signal significance further, which lie beyond the scope of a Master's thesis. One possibility could be to study additional selection cuts. Furthermore, the constraints for fitting the B_s meson with the pion in the algorithm are perhaps not sufficiently strict, since the majority of the B_s candidates are fitted with particles other than pions (the lack of hadron identification in ATLAS is a major weakness in this respect). A combined analysis over many decay modes of the B_c will clearly be necessary.

In conclusion, the LHC and the ATLAS experiment will provide unparalleled opportunities for particle physics research. The expected B-physics performance is

particularly prominent at the LHC compared to other running hadron machines, due to the high energy and high luminosity. The large b-quark production cross section will provide a better signal-to-noise ratio than before, making rare B decays accessible as well. It might therefore be possible for research to be made on the $B_c \to B_s\pi$ decay process with real ATLAS data.

Appendix A

C++ code

A.1 Analyze Bs2JpsiPhi

```
// analyzeBs2Jpsiphi.cxx
// This program provides analysis of Bs->Jpsi(mumu)phi(KK) using the algorithms
// JpsiFinder and Bs2Jpsiphi.


#ifdef VCPP
#pragma warning(disable:4786)
#pragma warning(disable:4503)
#endif

#include <iostream>
#include <fstream>
#include <iomanip>
#include <string>
#include <vector>
#include <map>
#include <cmath>
#include <math.h>
#include <algorithm>

//core classes
#include "AAna/core/AAna.h"
#include "AAna/core/Experiment.h"
#include "AAna/core/Event.h"
#include "AAna/core/Particle.h"
#include "AAna/core/MCParticle.h"
#include "AAna/core/Muon.h"


//ATLAS stand-alone interface
#include "AAna/stream/StreamIn.h"
#include "AAna/stream/StreamOut.h"
#include "AAna/stream/BlockDefinitions.h"

// // ROOT stuff
#include <TROOT.h>
#include <TChain.h>
#include <TCanvas.h>
#include <TArc.h>
```

```
#include <TText.h>
#include <TMarker.h>
#include <TObjArray.h>
#include <TFile.h>
#include "TH2.h"
#include "TNtuple.h"

//Algorithms
#include "AAna/algorithms/PrimaryVertex.h"
#include "AAna/algorithms/JpsiFinder.h"
#include "AAna/algorithms/Bs2Jpsiphi.h"
#include "AAna/algorithms/ToolKit.h"

using namespace std;
using namespace AAna;

namespace AAna{
  //Global objects
  Hive events;      //container for all event-related information
  Hive experiment; //container for all experiment-related information
}

Double_t bsmass;
Double_t bstao;
Double_t bschi2;
Double_t bspt;
Bool_t isTrueBs;
Double_t trkpt;
Double_t trketa;
Double_t muonpt;
Double_t truemuonpt;
Double_t muoneta;
Double_t truemuoneta;
Double_t muonpairmass;
Double_t jpsimass;
Double_t jpsitao;
Double_t jpsichi2;
Double_t jpsipt;
Bool_t isTrueJpsi;
Double_t bsJpsimass;
Double_t bsJpsichi2;
Double_t bsJpsipt;
Double_t bswbgMass;



void analyse(Event* pevt);

// Functions/variables for ROOT
TFile *bFile;
TTree *jpsiTree;
TTree *bsTree;
TTree *trkTree;
TTree *muonTree;
TTree *pairTree;
TTree *bswBgTree;

void makeBranch() {
  bFile = new TFile("bstest.root", "recreate");
  bsTree = new TTree("bsTree","BS_TREE");
```

```
  jpsiTree = new TTree("jpsiTree","JPSI_TREE");
  trkTree = new TTree("trkTree","TRK_TREE");
  muonTree = new TTree("muonTree","MUON_TREE");
  pairTree = new TTree("pairTree","PAIR_TREE");
  bswBgTree = new TTree("bswBgTree","BS_TREE_W_BG");

  bsTree->Branch("bs_mass",&bsmass,"bs_mass/D");
  bsTree->Branch("bs_tao",&bstao,"bs_tao/D");
  bsTree->Branch("bs_chi2",&bschi2,"bs_chi2/D");
  bsTree->Branch("bs_pt",&bspt,"bs_pt/D");
  bsTree->Branch("isTrueBs",&isTrueBs,"isTrueBs/O");
  trkTree->Branch("trk_pt",&trkpt,"trk_pt/D");
  trkTree->Branch("trk_eta",&trketa,"trk_eta/D");
  muonTree->Branch("muon_pt",&muonpt,"muon_pt/D");
  muonTree->Branch("true_muon_pt",&truemuonpt,"true_muon_pt/D");
  muonTree->Branch("muon_eta",&muoneta,"muon_eta/D");
  muonTree->Branch("true_muon_eta",&truemuoneta,"true_muon_eta/D");
  pairTree->Branch("muonpair_mass",&muonpairmass,"muonpair_mass/D");
  jpsiTree->Branch("jpsi_mass",&jpsimass,"jpsi_mass/D");
  jpsiTree->Branch("jpsi_tao",&jpsitao,"jspi_tao/D");
  jpsiTree->Branch("jpsi_chi2",&jpsichi2,"jpsi_chi2/D");
  jpsiTree->Branch("jpsi_pt",&jpsipt,"jpsi_pt/D");
  jpsiTree->Branch("isTrueJpsi",&isTrueJpsi,"isTrueJpsi/O");
  bsTree->Branch("bsJpsimass",&bsJpsimass,"bsJpsimass/D");
  bsTree->Branch("bsJpsichi2",&bsJpsichi2,"bsJpsichi2/D");
  bsTree->Branch("bsJpsipt",&bsJpsipt,"bsJpsipt/D");
  bswBgTree->Branch("bs_w_bg_mass",&bswbgMass,"bs_w_bg_mass/D");
}
void saveBranch(){
  bFile->cd();

  bsTree->Write();
  jpsiTree->Write();
  pairTree->Write();
  trkTree->Write();
  muonTree->Write();
  bswBgTree->Write();
  bsTree->Print();
  jpsiTree->Print();
  pairTree->Print();
  trkTree->Print();
  muonTree->Print();
  bswBgTree->Print();

  bFile->Close();
  }
void fillRoot(Event* pevt);


//Numbers of different kinds of particles to calculate
int nbrOfTrueRecoJpsi = 0;
int nbrOfTotJpsiTracks = 0;
int nbrOfFakeJpsiTracks = 0;
int nbrOfTrueRecoBs = 0;
int nbrOfTotBsTracks = 0;
int nbrOfFakeBsTracks = 0;
int nbrOfFakeBsWithTrueJpsi = 0;
int nbrOfBackgroundEvents = 0;

int main(int argc, char** argv){
```

```
//AA initilisation
Global::newJob();

//Define input interface
StreamIn streamIn;
streamIn.logLevel(INFO);
// streamIn.openFile("Bs_DPD"); // Input file
streamIn.openFileList("BsJpsiPhi"); // Input file


//--There is a default collection of I/O blocks which is defined
//--automatically when the Stream object is created.
//--Instead of setting the default collection of input/output blocks,
//--user can define the required collection of blocks himself using
//--these commands:
//
//     stream.clearBlockCollection();
//     ParticleDPD::addBlock(&stream);
//     MuonDPD::addBlock(&stream);
//     etc...
//
//--In this case, only the defined blocks will be input or output.
//--Teh same method addBlock(Stream*) can be used both for input and output.

//Create Experiment object
Experiment* pexp = Experiment::newObject(&experiment);

//optional printout of experiment-related information
pexp->print(INFO);

//pointer for event
Event* pevt;
int count = 0;

// Book ROOT tree branch
makeBranch();

//Get the event from the input stream and
//loop over all events in the stream

while(streamIn.getEvent(&events,pexp,pevt)) {
  ++count;

  // Get run and event numbers and print to screen
  if (count<2) cout << "RUN NUMBER: " << pevt->runNumber() << std::endl;
  if (count<10) cout << "EVENT: " << pevt->eventNumber() << std::endl;
  if (count==11) cout << "Now printing every 1000th event" << std::endl;
  if ((count % 1000)==0) cout << "EVENT: " << pevt->eventNumber() << std::endl;
  //if (count = 1000) break;

  //Analyse event with the standard algorithms
  analyse(pevt);

  // Fill the root branch with data
  fillRoot(pevt);

  //Clear container in the end of cycle
  events.clear();
```

```
  }

  saveBranch();
  streamIn.close();

  std::cout << "Number of total reconstructed Bs->Jpsi(mumu)Phi(KK) tracks: "
            << nbrOfTotBsTracks << std::endl;
  std::cout << "Number of true reconstructed Bs->Jpsi(mumu)Phi(KK) events: "
            << nbrOfTrueRecoBs << std::endl;
  std::cout << "Number of fake reconstructed Bs->Jpsi(mumu)Phi(KK) tracks: "
            << nbrOfFakeBsTracks << std::endl;
  std::cout << "Number of fake reconstructed Bs->Jpsi(mumu)Phi(KK) with true
reconstucted Jpsi->mumu as descendant: " << nbrOfFakeBsWithTrueJpsi << std::endl;
  std::cout << "Number of total reconstructed Jpsi->mumu tracks: "
            << nbrOfTotJpsiTracks << std::endl;
  std::cout << "Number of true reconstructed Jpsi->mumu events: "
            << nbrOfTrueRecoJpsi << std::endl;
  std::cout << "Number of fake reconstructed Jpsi->mumu tracks: "
            << nbrOfFakeJpsiTracks << std::endl;
}

void analyse(Event* pevt){

  // Check
  if(pevt->particleCollection() == 0) return;

  //Definition of Primary Vertex Finder
  PrimaryVertex* papv = PrimaryVertex::includeIn(pevt);

  // Definition of J/psi finder
  JpsiFinder* jpsi0 = JpsiFinder::includeIn(pevt);

  // Definition of Bs->Jpsi phi algorithm
  Bs2Jpsiphi* bs2jpsiphi = Bs2Jpsiphi::includeIn(pevt);
  bs2jpsiphi->useConstraints(true);

  //Primary vertex and association of particles to primary vertex
  //This only runs if the primary vertex from ATLAS is NULL
  if(!papv->valid()) papv->run();

  // Run Jpsi search
  if(!jpsi0->valid()) jpsi0->run();

  // Run Bs->Jpsi phi search
  if(!bs2jpsiphi->valid()) bs2jpsiphi->run();
}

void fillRoot(Event* pevt){
  ToolKit myToolKit;
  int i=0;
  double c = 299792458; //in m/s

  ParticleCollection* partColl = pevt->particleCollection();
  if (partColl == 0) return;

  std::vector<Particle::Type> muonmasses;
  muonmasses.push_back(Particle::MU_PLUS);
  muonmasses.push_back(Particle::MU_MINUS);

  Class* bsfinder = pevt->algorithm("Bs2Jpsiphi");
```

```
// Get the Bs candidates from the finder
ParticleCollection* theBsCandidates = bsfinder->particleCollection();
// Get the Jpsi candidates from the finder
Class* jpsifinder = pevt->algorithm("JpsiFinder");
ParticleCollection* theJpsiCandidates = jpsifinder->particleCollection();
// Get the MCParticleCollection from the event
MCParticleCollection* theMCParticles = pevt->mcParticleCollection();

// Plot the transverse momentum and pseudorapidity of all the
// reconstructed tracks in the event
ParticleCollectionIt partItr;
for (partItr=partColl->begin(); partItr!=partColl->end(); ++partItr) {
  trkpt = (*partItr)->momentumTransverse();
  trketa = (*partItr)->pseudoRapidity();
  trkTree->Fill();
}


// Muons
MuonCollection* muonColl = pevt->muonCollection();
ParticleCollection theMuonPtcles;
if (muonColl != 0) {
  //Plot the transverse momentum and the pseudorapidity of the reconstructed
  //muons in the event
  MuonCollectionIt muonItr;
  Particle* m;
  for (muonItr=muonColl->begin(); muonItr!=muonColl->end(); ++muonItr) {
    m = (*muonItr)->particle();
    muonpt = m->momentumTransverse();
    muoneta = m->pseudoRapidity();
    theMuonPtcles.push_back(m);
    muonTree->Fill();
  }
}

//Plot the transverse momentum and the pseudorapidity of the true
//reconstructed muons in the event also saving the true reconstructed
//muons in trueMuonPtcles, for later use.
ParticleCollection trueMuonPtcles;
std::vector<int> topology;
topology.push_back(443);
topology.push_back(-13);
topology.push_back(13);
std::vector<ParticleCollection> theTrueRecoJpsi =
      myToolKit.findSignalParticles(topology,*theMCParticles,*partColl);

//Plot the invariant mass of all the muon pairs in this event
std::vector<ParticleCollection> muonPairs =
  myToolKit.makePairs(theMuonPtcles, true);
std::vector<ParticleCollection>::iterator pairItr;
Particle* muPlus;
Particle* muMinus;
for (pairItr=muonPairs.begin(); pairItr!=muonPairs.end(); ++pairItr) {
  ParticleCollectionIt muItr;
  muItr = pairItr->begin();
  muPlus = (*muItr);
  muItr++;
  muMinus = (*muItr);
  muonpairmass = myToolKit.invariantMass
    (muPlus, Particle::MU_PLUS, muMinus, Particle::MU_MINUS);
```

```
  muonTree->Fill();
}

//Get the true reconstructed muons and kaons and put them into trueBsSignalPtcles
ParticleCollectionIt bsItr;
ParticleCollection trueBsSignalPtcles;
std::vector<int> topology2;
topology2.push_back(531);
topology2.push_back(443);
topology2.push_back(-13);
topology2.push_back(13);
topology2.push_back(333);
topology2.push_back(-321);
topology2.push_back(321);
std::vector<ParticleCollection> theTrueRecoBs =
   myToolKit.findSignalParticles(topology2,*theMCParticles,*partColl);
std::vector<ParticleCollection>::iterator muKaCollItr;
for (muKaCollItr=theTrueRecoBs.begin(); muKaCollItr!=theTrueRecoBs.end();
     ++muKaCollItr) {
  if ((*muKaCollItr).size() == 4) {
    ParticleCollectionIt trueMuKaItr;
    for (trueMuKaItr=muKaCollItr->begin(); trueMuKaItr!=muKaCollItr->end();
         ++trueMuKaItr) {
      trueBsSignalPtcles.push_back(*trueMuKaItr);
    }
  }
}
// Loop over the Bs candidates, get the invariant masses, proper lifetime, chi2
// and transverse momentum and put them into an n-tuple, and plot corresponding
// Jpsis at the same time
std::vector<Particle::Type> masses;
masses.push_back(Particle::MU_PLUS);
masses.push_back(Particle::MU_MINUS);
masses.push_back(Particle::K_MINUS);
masses.push_back(Particle::K_PLUS);

for (bsItr = theBsCandidates->begin(); bsItr!=theBsCandidates->end(); ++bsItr) {
  nbrOfTotBsTracks++;
  bsmass = (*bsItr)->mass(masses);
  bschi2 = (*bsItr)->decayVertex()->chi2();
  bspt = (*bsItr)->momentumTransverse();

  //Get the proper lifetime
  double x1 = (*bsItr)->primaryVertex()->position(1);
  double y1 = (*bsItr)->primaryVertex()->position(2);
  double z1 = (*bsItr)->primaryVertex()->position(3);
  double x2 = (*bsItr)->decayVertex()->position(1);
  double y2 = (*bsItr)->decayVertex()->position(2);
  double z2 = (*bsItr)->decayVertex()->position(3);
  double distance = sqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1)+(z2-z1)*(z2-z1))/1000;
  //Dividing the distance it with 1000 to get it in meters

  double lxy; //the decay length in the xy plane
  double slxy; // the standard deviation of the decay length
  (*bsItr)->decayLengthXY(lxy,slxy);
  double p = (*bsItr)->momentumTotal(); //in dimensions of MeV
  double m = (*bsItr)->mass(masses); //in dimensions of MeV
  double E = sqrt(p*p + m*m); //in dimensions of MeV
  double v = p*c/E;
  double gamma = 1/(sqrt(1-v*v/(c*c)));
```

```
bstao = lxy/1000/(v*gamma);

ParticleCollection* children = (*bsItr)->children();

// Get associated Jpsi and replot
for (ParticleCollectionIt jpsiItr=theJpsiCandidates->begin();
     jpsiItr!=theJpsiCandidates->end(); ++jpsiItr) {
  ParticleCollection* jpsimuons = (*jpsiItr)->children();
  ParticleCollection bmuons;
  for (ParticleCollectionIt bsChdrIt = children->begin();
           bsChdrIt!=children->end(); ++bsChdrIt) {
    if(((*bsChdrIt)->type() == Particle::MU_PLUS) ||
           ((*bsChdrIt)->type() == Particle::MU_MINUS)) {
      bmuons.push_back(*bsChdrIt);
    }
  }
  if(myToolKit.isIdenticalCollection((*jpsimuons),(bmuons))) {
    bsJpsimass = (*jpsiItr)->mass(muonmasses);
    bsJpsichi2 = (*jpsiItr)->decayVertex()->chi2();
    bsJpsipt = (*jpsiItr)->momentumTransverse();
  }
}

//Plotting the true reconstructed Bs, after checking that they are indeed true
int cnt = 0;
isTrueBs = false;
bool foundAMatch = false;
std::vector<ParticleCollection>::iterator muKaCollItr2;
for (muKaCollItr2=theTrueRecoBs.begin(); muKaCollItr2!=theTrueRecoBs.end();
     ++muKaCollItr2) {
  if (myToolKit.isIdenticalCollection((*children),(*muKaCollItr2))) {
    nbrOfTrueRecoBs++;
    foundAMatch = true;
    isTrueBs = true;
    break;
  }
}

if(!foundAMatch) {
    nbrOfFakeBsTracks++;

    //to check if this fake Bs has a true Jpsi as a descendant
    ParticleCollectionIt childItr;
    for (childItr = children->begin(); childItr!=children->end(); ++childItr) {
      ParticleCollectionIt trueMuKaItr;
      //iterate over the true reconstructed muons and kaons
      for(trueMuKaItr = trueBsSignalPtcles.begin();
            trueMuKaItr!=trueBsSignalPtcles.end(); ++trueMuKaItr) {
        if ((*childItr) == (*trueMuKaItr)) {
          cnt++;
        }
      }
    }
    if(cnt == 2) {
      MCParticle* muon;
      ParticleCollectionIt itr = children->begin();
      if ((*itr)->mcParticle() != NULL) muon =(*itr)->mcParticle();
      if(abs(muon->pdgID()) == 13) nbrOfFakeBsWithTrueJpsi++;
    }
}
```

```
    bsTree->Fill();
  }

  // Plot the invariant mass, chi2 and transverse momentum for all formed Jpsi in
  // this event
  ParticleCollectionIt JpsiItr;
  for (JpsiItr = theJpsiCandidates->begin();
       JpsiItr!=theJpsiCandidates->end(); ++JpsiItr) {
    isTrueJpsi = false;
    nbrOfTotJpsiTracks++;
    jpsimass = (*JpsiItr)->mass(muonmasses);
    jpsichi2 = (*JpsiItr)->decayVertex()->chi2();
    jpsipt = (*JpsiItr)->momentumTransverse();

    //same for the true reconstructed Jpsi, after checking that they are
    //indeed true
    bool foundAMatch = false;
    ParticleCollection* children = (*JpsiItr)->children();
    std::vector<ParticleCollection>::iterator muCollItr2;
    for (muCollItr2=theTrueRecoJpsi.begin(); muCollItr2!=theTrueRecoJpsi.end();
         ++muCollItr2) {
      if (myToolKit.isIdenticalCollection((*children),(*muCollItr2))) {
        nbrOfTrueRecoJpsi++;
        foundAMatch = true;
        isTrueJpsi = true;
        break;
      }
    }

    if(!foundAMatch) {
      nbrOfFakeJpsiTracks++;
    }
    jpsiTree->Fill();
  }
}
```

A.2 Bc2BsPi finder

```
// BC2BSPI - algorithm for searching for decays of the type
// Bc->Bs(Jpsi(mu+mu-)phi(K+K-))


#include <iostream>

#include "AAna/core/Event.h"
#include "AAna/core/Particle.h"
#include "AAna/core/Beam.h"
#include "AAna/core/Vertex.h"
#include "AAna/core/Muon.h"
#include "AAna/core/Constraint.h"
#include "AAna/core/ConstraintMass.h"
#include "AAna/core/ConstraintOrigin.h"
#include "AAna/core/ConstraintProcessor.h"
#include "AAna/core/ConstraintVertex.h"
#include "AAna/algorithms/PrimaryVertex.h"
#include "AAna/algorithms/JpsiFinder.h"
#include "AAna/algorithms/V0Finder.h"
#include "AAna/algorithms/ToolKit.h"
#include "AAna/algorithms/Bs2Jpsiphi2.h"
```

```
#include "AAna/algorithms/Bc2BsPi.h"

using namespace std;
using namespace AAna;

// FRAMEWORK-RELATED CODE. DO NOT MODIFY BELOW THIS LINE.
// ALL NEW ALGORITHMS MUST CONTAIN THESE METHODS

AAna::Bc2BsPi* AAna::Bc2BsPi::newObject(Hive* pbox,
                                        const Event* pevt){
  if(pevt->local()) {
    if(pevt->logLevel() > AAna::SILENT) {
      cout << "Vertex: Attempt to store the link to local event." << endl;
      pevt->print(pevt->logLevel());
    }
    abort();

  }

  Bc2BsPi* p = new Bc2BsPi(pevt);
  if(p != 0) {
    pbox->addObject(p);
  }
  return p;
}

bool AAna::Bc2BsPi::verify() const {
  //verify all links within an object
  for(ParticleCollectionCIt p = _lptl.begin(); p != _lptl.end(); ++p){
    if(*p != 0) if((*p)->local())   return false;
  }
  if(_pevt  != 0) if(_pevt->local())  return false;
  return true;
}

AAna::Class* AAna::Bc2BsPi::store(Hive* pbox){

  { //Check if the object is stored already in the Hive
    //Namely here we break potential closed loops in requests to store objects
    //Therefore, the loop of links will be processed correctly.
    Class* p = getGlobalLink(pbox);
    if(p != 0) return p;
  }

  LogLevel l = logLevel();
  if(l == AAna::UNKNOWN) {
    //This is the indicator that the requests to store objects make a closed loop.
    //It is a severe error in the structure. Normally it should never happen,
    //since the previous check should break the loop and return the pointer of the
    //new stored object. We keep this check just to be safe.

    cout << name() << ": Closed loop in request to store object." << endl;
    print(AAna::ERROR);
    abort();
  }
  logLevel(AAna::UNKNOWN);

  Bc2BsPi* p = Bc2BsPi::newObject(pbox,event());
  *p =  *this;
  setGlobalLink(p);
```

```
  //Store all linked objects in Hive
  p->storeLinks(pbox);

  //Return the logLevel into the initial state
  logLevel(l);
  p->logLevel(l);

  if(!p->verify()) {
    //Severe error: links to local objects detected.
    //Normally it should never happen
    if(logLevel() > AAna::SILENT) {
      cout << name() << ": Attempt to store object"
           << " with links to local objects." << endl;
      print(logLevel());
    }
    abort();
  }

  return p;
}

void AAna::Bc2BsPi::storeLinks(Hive* pbox){
  for(ParticleCollectionIt p = _lptl.begin(); p != _lptl.end(); ++p){
    if(*p != 0) (*p) = Particle::convert((*p)->store(pbox));
  }
}

void AAna::Bc2BsPi::clear(){
  valid(false);
  _lptl.clear();
}

AAna::Bc2BsPi* AAna::Bc2BsPi::includeIn(Event* pevt, const string& str){
  //Include algorithm in the event. The new object is created only if the event
  //does not contain already such an algorithm
  Bc2BsPi* pv = Bc2BsPi::convert(pevt->algorithm(str));
  if(pv == 0) {
    pv = Bc2BsPi::newObject(pevt->hive(),pevt);
    pv->name(str);
    pevt->addAlgorithm(pv);
  }
  return pv;
}

void AAna::Bc2BsPi::print(LogLevel log) const {
  if(log > AAna::WARNING) {
    if(!valid()) {
      cout << "Bc2BsPi index: " << index() << " not valid" << endl;
      return;
    }
    cout << "Bc2BsPi index: " << index() << endl;
  }

}

AAna::Bc2BsPi::~Bc2BsPi(){
  if(logLevel() > AAna::DETAILS) {
    cout << "Bc2BsPi: Removing object " << name() << " with index " << index()
         << endl;
```

```
  }
}

// END OF GENERAL FRAMEWORK CODE
////////////

bool AAna::Bc2BsPi::run(){

//Search for all Bc mesons decaying to a Pi and a Bs where the latter is decaying
//to two muons and two charged kaons via the intermediate state J/psi phi
  valid(false);
  bool success = false;

  //Find Primary vertex algorithm
  const PrimaryVertex* papv = PrimaryVertex::convert
    (_pevt->algorithm("PrimaryVertex"));
  if(papv == 0) return false;

  // Retrieve the Bs2JpsiPhi
  const Bs2Jpsiphi2* bs0 = Bs2Jpsiphi2::convert(_pevt->algorithm("Bs2Jpsiphi2"));
  if(bs0 == 0) return false;

  // Select those Bs candidates which are within an appropriate mass window
  const ParticleCollection* theBsCandidates = bs0->particleCollection();
  if(theBsCandidates == 0) return false;

  // Get collection of all particles from Event
  const ParticleCollection* eventParticles = _pevt->particleCollection();
  if(eventParticles == 0) return false;

  double bsSigma = 100.0;
// Permissible width in MeV - check this number while running
  double bsPDGMass = 5367.5;
  double piPDGMass = Particle::mass(Particle::PI_PLUS);
// PDG mass of the pi
  double bsChi2Max = 25.0;
  double bsPtMin = 8000.0;

  std::vector<double> masses;
  masses.push_back(5367.5);
  masses.push_back(1395.7);

  std::vector<double> bcPtcleMasses;
  std::vector<AAna::Particle::Type> bsSignalPtcleTypes;
  bcPtcleMasses.push_back(bsPDGMass);
  bcPtcleMasses.push_back(Particle::mass(Particle::PI_PLUS));
  bsSignalPtcleTypes.push_back(Particle::MU_PLUS);
  bsSignalPtcleTypes.push_back(Particle::MU_MINUS);
  bsSignalPtcleTypes.push_back(Particle::K_PLUS);
  bsSignalPtcleTypes.push_back(Particle::K_MINUS);


  //Loop over Bs candidates (there should not be more than one Bc
  //per Bs)
  for(ParticleCollectionCIt pB = theBsCandidates->begin(); pB !=
        theBsCandidates->end(); ++pB){
    double bsMass = (*pB)->mass(bsSignalPtcleTypes);
    double bsChi2 = (*pB)->decayVertex()->chi2();
    double bsPt = (*pB)->momentumTransverse();
    if ((bsMass < (bsPDGMass - bsSigma)) || (bsMass > (bsPDGMass + bsSigma))) {
```

```
    continue;}
  if (bsChi2 > bsChi2Max) {
    continue;}
  if (bsPt < bsPtMin){
    continue;}

  Particle* muon1 = (*pB)->children()->operator[](0);
  Particle* muon2 = (*pB)->children()->operator[](1);
  Particle* kaon1 = (*pB)->children()->operator[](2);
  Particle* kaon2 = (*pB)->children()->operator[](3);

  Vertex bcVrt(_pevt); //construct bc vertex
  Particle bcMeson(_pevt); //create Bc meson
  Particle* pion;

  //Loop over all particles in the event to find pion
  for(ParticleCollectionCIt p1 = eventParticles->begin(); p1 !=
      eventParticles->end(); ++p1){
    if(*p1 == muon1) continue;
    if(*p1 == muon2) continue;
    if(*p1 == kaon1) continue;
    if(*p1 == kaon2) continue;
    pion = *p1;

    if(!bcVrt.find((*p1),(*pB))) {
      continue;}
    //  Build the vertex from the Bs and pion (no constraints yet)
    if(bcVrt.chi2() > _chi2Max) {
      continue;} // Chi2 check (chi2Max = 25)
    double q = (*pB)->charge() + (*p1)->charge();
    if(!bcMeson.combine(&bcVrt,q)) {
      continue;} // Build combined particle
    if(!papv->associateParticle(&bcMeson)) {
      continue;} // Associate with the primary vertex


    Particle* pp;
    if(_doConstraint) { // Now do the constrained fit
      Vertex* pv = bcMeson.decayVertex(); // Unconstrained vertex from above
      Vertex* po = bcMeson.primaryVertex(); // Assoc. PV
      ConstraintVertex cv(_pevt); // Create new constrained vertex
      cv.fill(pv); // Define constrained vertex
      ConstraintOrigin co(_pevt); // Create pointing constraint object
      co.fill(po,pv); // Define pointing constraint object

      ConstraintProcessor cpc(_pevt); // Create consraint processor object
      cpc.addConstraint(&cv);
      cpc.addConstraint(&co);
      bcMeson.setConstraints(&cpc); // Do the work
      if(!cpc.valid() ) {
        continue;} // Accept/reject?
      pp = Particle::convert(bcMeson.store(hive()));
    }
    if (pp == 0) pp = Particle::convert(bcMeson.store(hive()));
    _lptl.push_back(pp);
    success = true; // Set to true for any successful fit (we only need one
    //accepted candidate for a success)
  } // end of loop over pions
  valid(success);
} // End of loop over Bs
```

```
  return success;
}
```

A.3 Analyze Bc2BsPi

```
// analyzeBc2BsPi.cxx
// This program provides analysis of Bc->Bs(Jpsi(mumu)phi(KK))Pi using the
// algorithms JpsiFinder, Bs2Jpsiphi and Bc2BsPi.


#ifdef VCPP
#pragma warning(disable:4786)
#pragma warning(disable:4503)
#endif

#include <iostream>
#include <fstream>
#include <iomanip>
#include <string>
#include <vector>
#include <map>
#include <cmath>
#include <math.h>
#include <algorithm>

//core classes
#include "AAna/core/AAna.h"
#include "AAna/core/Experiment.h"
#include "AAna/core/Event.h"
#include "AAna/core/Particle.h"
#include "AAna/core/MCParticle.h"
#include "AAna/core/Muon.h"


//ATLAS stand-alone interface
#include "AAna/stream/StreamIn.h"
#include "AAna/stream/StreamOut.h"
#include "AAna/stream/BlockDefinitions.h"

// // ROOT stuff
#include <TROOT.h>
#include <TChain.h>
#include <TCanvas.h>
#include <TArc.h>
#include <TText.h>
#include <TMarker.h>
#include <TObjArray.h>
#include <TFile.h>
#include "TH2.h"
#include "TNtuple.h"

//Algorithms
#include "AAna/algorithms/PrimaryVertex.h"
#include "AAna/algorithms/JpsiFinder.h"
#include "AAna/algorithms/Bs2Jpsiphi2.h"
#include "AAna/algorithms/Bc2BsPi.h"
#include "AAna/algorithms/ToolKit.h"
```

```
using namespace std;
using namespace AAna;

namespace AAna{
  //Global objects
  Hive events;      //container for all event-related information
  Hive experiment; //container for all experiment-related information
}

Double_t bcmass;
Double_t bctao;
Double_t bctaoErr;
Double_t bcchi2;
Double_t bcpt;
Bool_t isTrueBc;
Double_t bcBsmass;
Double_t bcBschi2;
Double_t bcBspt;
Double_t bsmass;
Double_t bstao;
Double_t bschi2;
Double_t bspt;
Bool_t isTrueBs;
Double_t bcwbgMass;
Double_t cosalpha;
Double_t impactparameter;
Double_t pionpt;
Double_t pionimpact;

std::vector<ParticleCollection> theGlobalBsDecayProds;
std::vector<ParticleCollection> theGlobalBcDecayProds;
ParticleCollection theGlobalTrks;

void analyse(Event* pevt);

// Functions/variables for ROOT
TFile *bFile;
TTree *bcTree;
TTree *bsTree;
TTree *bcwBgTree;

void makeBranch() {
  bFile = new TFile("bctest.root", "recreate");
  bcTree = new TTree("bcTree","BC_TREE");
  bsTree = new TTree("bsTree","BS_TREE");
  bcwBgTree = new TTree("bcwBgTree","BC_TREE_W_BG");

  bcTree->Branch("bc_mass",&bcmass,"bc_mass/D");
  bcTree->Branch("bc_tao",&bctao,"bc_tao/D");
  bcTree->Branch("bc_tao_Err",&bctaoErr,"bc_tao_Err/D");
  bcTree->Branch("bc_chi2",&bcchi2,"bc_chi2/D");
  bcTree->Branch("bc_pt",&bcpt,"bc_pt/D");
  bcTree->Branch("isTrueBc",&isTrueBc,"isTrueBc/O");
  bcTree->Branch("bcBsmass",&bcBsmass,"bcBsmass/D");
  bcTree->Branch("bcBschi2",&bcBschi2,"bcBschi2/D");
  bcTree->Branch("bcBspt",&bcBspt,"bcBspt/D");
  bcTree->Branch("cos_alpha",&cosalpha,"cos_alpha/D");
  bcTree->Branch("impact_parameter",&impactparameter,"impact_parameter/D");
  bcTree->Branch("pion_pt",&pionpt,"pion_pt/D");
  bcTree->Branch("pion_impact",&pionimpact,"pion_impact/D");
```

```
  bsTree->Branch("bs_mass",&bsmass,"bs_mass/D");
  bsTree->Branch("bs_tao",&bstao,"bs_tao/D");
  bsTree->Branch("bs_chi2",&bschi2,"bs_chi2/D");
  bsTree->Branch("bs_pt",&bspt,"bs_pt/D");
  bsTree->Branch("isTrueBs",&isTrueBs,"isTrueBs/O");

  bcwBgTree->Branch("bc_w_bg_mass",&bcwbgMass,"bc_w_bg_mass/D");
}

void saveBranch(){
  bFile->cd();

  bcTree->Write();
  bsTree->Write();
  bcwBgTree->Write();
  bcTree->Print();
  bsTree->Print();
  bcwBgTree->Print();

  bFile->Close();
  }

void fillRoot(Event* pevt);

//Numbers of different kinds of particles to calculate
int nbrOfTrueRecoBc = 0;
int nbrOfTotBcTracks = 0;
int nbrOfFakeBcTracks = 0;
int nbrOfTotBsTracks = 0;
int nbrOfFakeBsTracks = 0;
int nbrOfTrueRecoBs = 0;
int nbrOfBackgroundEvents = 0;
int trueRecoBsCntr = 0;
int trueRecoBcCntr = 0;
int trueBcDecayCntr = 0;
int trueBsDecayCntr = 0;

void findTruthTracks(Event* pevt) {

  //Get the true reconstructed muons and kaons and put them into
  //theGlobalBsChildren
  AAna::ToolKit myToolKit;
  ParticleCollection* partColl = pevt->particleCollection();
  MCParticleCollection* MCParticles = pevt->mcParticleCollection();
  MCParticleCollection theMCParticles;
  for (MCParticleCollectionIt mcItr = MCParticles->begin(); mcItr !=
       MCParticles->end(); ++mcItr) {
    if ( !( ((*mcItr)->pdgID() == 541) && ((*mcItr)->decayVertex() == NULL)))
      theMCParticles.push_back(*mcItr);
    if ( ((*mcItr)->pdgID() == 541) && ((*mcItr)->decayVertex() != NULL) )
      ++trueBcDecayCntr;
    if ( ((*mcItr)->pdgID() == 531) && ((*mcItr)->decayVertex() != NULL) )
      ++trueBsDecayCntr;
  }

  theGlobalBsDecayProds.clear();
  theGlobalBcDecayProds.clear();

  std::vector<int> topology;
```

```
  topology.push_back(531);
  topology.push_back(443);
  topology.push_back(-13);
  topology.push_back(13);
  topology.push_back(333);
  topology.push_back(-321);
  topology.push_back(321);
  theGlobalBsDecayProds = myToolKit.findSignalParticles(topology,
                          theMCParticles,*partColl);
  trueRecoBsCntr += theGlobalBsDecayProds.size();

  if(theGlobalBsDecayProds.size()==1) theGlobalTrks =
                                        *(theGlobalBsDecayProds.begin());

  //Get the true reconstructed pion, muons and kaons and put them
  //into trueBcSignalPtcles
  std::vector<int> topology2;
  topology2.push_back(541);
  topology2.push_back(531);
  topology2.push_back(443);
  topology2.push_back(-13);
  topology2.push_back(13);
  topology2.push_back(333);
  topology2.push_back(-321);
  topology2.push_back(321);
  topology2.push_back(211);
  std::vector<ParticleCollection> theBcDecayProds =
    myToolKit.findSignalParticles(topology2,theMCParticles,*partColl);
  std::vector<ParticleCollection>::iterator muKaPiCollItr;
  for (muKaPiCollItr=theBcDecayProds.begin();
       muKaPiCollItr!=theBcDecayProds.end(); ++muKaPiCollItr) {
    if ((*muKaPiCollItr).size() == 5) {
      theGlobalBcDecayProds.push_back(*muKaPiCollItr);
    }
  }
  trueRecoBcCntr += theGlobalBcDecayProds.size();
}


int main(int argc, char** argv){

  //AA initilisation
  Global::newJob();

  //Define input interface
  StreamIn streamIn;
  streamIn.logLevel(INFO);
  streamIn.openFileList("BcBsPiList"); // Input file


  //--There is a default collection of I/O blocks which is defined
  //--automatically when the Stream object is created. Instead of setting
  //--the default collection of input/output blocks, user can define the
  //--required collection of blocks himself using these commands:
  //
  //     stream.clearBlockCollection();
  //     ParticleDPD::addBlock(&stream);
  //     MuonDPD::addBlock(&stream);
  //     etc...
  //
```

```
  //--In this case, only the defined blocks will be input or output.
  //--The same method addBlock(Stream*) can be used both for input and output

  //Create Experiment object
  Experiment* pexp = Experiment::newObject(&experiment);

  //optional printout of experiment-related information
  pexp->print(INFO);

  //pointer for event
  Event* pevt;
  int count = 0;

  // Book ROOT tree branch
  makeBranch();

  while(streamIn.getEvent(&events,pexp,pevt)) {
    ++count;

    // Get run and event numbers and print to screen
    if (count<2) cout << "RUN NUMBER: " << pevt->runNumber() << std::endl;
    if (count<10) cout << "EVENT: " << pevt->eventNumber() << std::endl;
    if (count==11) cout << "Now printing every 1000th event" << std::endl;
    if ((count % 1000)==0) cout << "EVENT: " << pevt->eventNumber()
                                << std::endl;

    //Analyse event with the standard algorithms
    analyse(pevt);

    // Fill the root branch with data
    fillRoot(pevt);

    //Clear container in the end of cycle
    events.clear();

    //if (count>10000) break;
  }

  saveBranch();
  streamIn.close();

  std::cout << "Total number of events in this file: " << count << std::endl;
  std::cout << "Number of Bc mesons in truth: "<<trueBcDecayCntr <<std::endl;
  std::cout << "Number of Bs mesons in truth: " <<trueBsDecayCntr <<std::endl;
  std::cout << "Number of Bc->Bs(Jpsi(mumu)Phi(KK))Pi fully reconstructed by
ATLAS: " << trueRecoBcCntr << std::endl;
  std::cout << "Number of Bc->Bs(Jpsi(mumu)Phi(KK))Pi found by code: "
            << nbrOfTotBcTracks << std::endl;
  std::cout << "... of which the following were correct: "<<nbrOfTrueRecoBc
            << std::endl;
  std::cout << "... and the following were fake: " << nbrOfFakeBcTracks
            << std::endl;
  std::cout << "Number of Bs(Jpsi(mumu)Phi(KK)) fully reconstructed by ATLAS:
" << trueRecoBsCntr << std::endl;
  std::cout << "Number of Bs->Jpsi(mumu)Phi(KK) found by code: "
            << nbrOfTotBsTracks << std::endl;
  std::cout << "... of which the following were correct: "<< nbrOfTrueRecoBs
            << std::endl;
  std::cout << "... and the following were fake: " << nbrOfFakeBsTracks
            << std::endl;
```

```
}

void analyse(Event* pevt){

  // Check
  if(pevt->particleCollection() == 0) return;

  // Find the tracks identified as being from the signal processes
  // by the MC truth
  findTruthTracks(pevt);

  //Definition of Primary Vertex Finder
  PrimaryVertex* papv = PrimaryVertex::includeIn(pevt);

  //Definition of J/psi finder
  JpsiFinder* jpsi0 = JpsiFinder::includeIn(pevt);

  //Definition of Bs->Jpsi phi algorithm
  Bs2Jpsiphi2* bs2jpsiphi = Bs2Jpsiphi2::includeIn(pevt);
  bs2jpsiphi->useConstraints(true);
  bs2jpsiphi->setMaxChi2(100.0);
  bs2jpsiphi->signalTrks = theGlobalTrks;

  //Definition of Bc->BsPi algorithm
  Bc2BsPi* bc2bspi = Bc2BsPi::includeIn(pevt);
  bc2bspi->useConstraints(true);

  //Primary vertex and association of particles to primary vertex
  //This only runs if the primary vertex from ATLAS is NULL
  if(!papv->valid()) papv->run();

  // Run Jpsi search
  if(!jpsi0->valid()) jpsi0->run();

  // Run Bs->Jpsi phi search
  if(!bs2jpsiphi->valid()) bs2jpsiphi->run();

  // Run Bc->BsPi search
  if(!bc2bspi->valid()) bc2bspi->run();

}

void fillRoot(Event* pevt){

  ToolKit myToolKit;
  double c = 299792458; //in m/s

  ParticleCollection* partColl = pevt->particleCollection();
  if (partColl == 0) return;

  std::vector<Particle::Type> muKaMasses;
  muKaMasses.push_back(Particle::MU_PLUS);
  muKaMasses.push_back(Particle::MU_MINUS);
  muKaMasses.push_back(Particle::K_PLUS);
  muKaMasses.push_back(Particle::K_MINUS);

  //Get the Bc candidates from the finder
  Class* bcfinder = pevt->algorithm("Bc2BsPi");
  ParticleCollection* theBcCandidates = bcfinder->particleCollection();
```

```
// Get the Bs candidates from the finder
Class* bsfinder = pevt->algorithm("Bs2Jpsiphi2");
ParticleCollection* theBsCandidates = bsfinder->particleCollection();
// Get the Jpsi candidates from the finder
Class* jpsifinder = pevt->algorithm("JpsiFinder");
ParticleCollection* theJpsiCandidates = jpsifinder->particleCollection();

//Get the muoncollection from the event
MuonCollection* muonColl = pevt->muonCollection();

//Loop over the Bc candidates, get the invariant masses, lifetime, chi2
//and transverse momentum and put them into an n-tuple and plot
//corresponding Bs at the same time
std::vector<double> masses;
masses.push_back(139.6);
masses.push_back(5367.5);

for (ParticleCollectionIt bcItr = theBcCandidates->begin();
     bcItr!=theBcCandidates->end(); ++bcItr) {
  nbrOfTotBcTracks++;
  bcmass = (*bcItr)->mass(masses);
  bcchi2 = (*bcItr)->decayVertex()->chi2();
  bcpt = (*bcItr)->momentumTransverse();
  impactparameter = (*bcItr)-> significance();

  double lxy; //the decay length in the xy plane
  double slxy; // the standard deviation of the decay length
  (*bcItr)->decayLengthXY(lxy,slxy);
  double p = (*bcItr)->momentumTotal(); //in dimensions of MeV
  double m = (*bcItr)->mass(masses); //in dimensions of MeV
  double E = sqrt(p*p + m*m); //in dimensions of MeV
  double v = p*c/E;
  double gamma = 1/(sqrt(1-v*v/(c*c)));
  bctao = lxy/1000/(v*gamma);
  bctaoErr = slxy/1000/(v*gamma);

  //Get the detected particles (muons, kaons and pion) descended
  //from the Bc and also the open angle between the Bs and Pi
  double pi_momTot;
  double bs_momTot;
  HepVector pi_mom;
  HepVector bs_mom;
  ParticleCollection bcDescendants;
  ParticleCollection* children = (*bcItr)->children();
  for (ParticleCollectionIt childItr = children->begin();
       childItr!=children->end(); ++childItr) {
    if ((*childItr)->decayVertex() == NULL) { // then it is a pion
      pi_momTot = (*childItr)->momentumTotal();
      pi_mom = (*childItr)->momentum();
      bcDescendants.push_back(*childItr);
      pionpt = (*childItr)->momentumTransverse();
      pionimpact = (*childItr)->significance();
    }else{ // then it is a Bs
      bs_momTot = (*childItr)->momentumTotal();
      bs_mom = (*childItr)->momentum();
      ParticleCollection* grandChildren = (*childItr)->children();
      for (ParticleCollectionIt grandChildItr = grandChildren->begin();
           grandChildItr!=grandChildren->end(); ++grandChildItr) {
        bcDescendants.push_back(*grandChildItr);
      }
```

```
    }
  }
  cosalpha = (pi_mom(1)*bs_mom(1) + pi_mom(2)*bs_mom(2)
              + pi_mom(3)*bs_mom(3))/(pi_momTot*bs_momTot);

  //Get associated Bs and replot
  for (ParticleCollectionIt bsItr=theBsCandidates->begin();
       bsItr!=theBsCandidates->end(); ++bsItr) {
    for (ParticleCollectionIt childItr = children->begin();
         childItr!=children->end(); ++childItr) {
      if ((*childItr)->decayVertex() == NULL) continue;
      if ((*childItr) == (*bsItr)) {
        bcBsmass = (*bsItr)->mass(muKaMasses);
        bcBschi2 = (*bsItr)->decayVertex()->chi2();
        bcBspt = (*bsItr)->momentumTransverse();
      }
    }
  }
  //Plotting the true reconstructed Bc, after checking that they are
  //indeed true
  int cnt = 0;
  isTrueBc = false;
  bool foundAMatch = false;
  std::vector<ParticleCollection>::iterator muKaPiCollItr2;
  for (muKaPiCollItr2=theGlobalBcDecayProds.begin();
  muKaPiCollItr2!=theGlobalBcDecayProds.end(); ++muKaPiCollItr2) {
    if (myToolKit.isIdenticalCollection((bcDescendants),(*muKaPiCollItr2))){
      nbrOfTrueRecoBc++;
      foundAMatch = true;
      isTrueBc = true;
      break;
    }
  }
  if(!foundAMatch) {
      nbrOfFakeBcTracks++;
  }
  bcTree->Fill();
}
// Plot the invariant mass, lifetime, chi2 and transverse momentum for
// all formed Bs in this event
for (ParticleCollectionIt bsItr = theBsCandidates->begin();
     bsItr!=theBsCandidates->end(); ++bsItr) {
  isTrueBs = false;
  nbrOfTotBsTracks++;
  bsmass = (*bsItr)->mass(muKaMasses);
  bschi2 = (*bsItr)->decayVertex()->chi2();
  bspt = (*bsItr)->momentumTransverse();

  //Get the proper lifetime
  double lxy; //the decay length in the xy plane
  double slxy; // the standard deviation of the decay length
  (*bsItr)->decayLengthXY(lxy,slxy);
  double p = (*bsItr)->momentumTotal(); //in dimensions of MeV
  double m = (*bsItr)->mass(masses); //in dimensions of MeV
  double E = sqrt(p*p + m*m); //in dimensions of MeV
  double v = p*c/E;
  double gamma = 1/(sqrt(1-v*v/(c*c)));
  bstao = lxy/(v*gamma);

  //same for the true reconstructed Bs, after checking that they
```

```
    //are indeed true
    bool foundAMatch = false;
    ParticleCollection* bsChildren = (*bsItr)->children();
    std::vector<ParticleCollection>::iterator muKaItr;
    for (muKaItr=theGlobalBsDecayProds.begin();
         muKaItr!=theGlobalBsDecayProds.end(); ++muKaItr) {
      if (myToolKit.isIdenticalCollection((*bsChildren),(*muKaItr))) {
        nbrOfTrueRecoBs++;
        foundAMatch = true;
        isTrueBs = true;
        break;
      }
    }

    if(!foundAMatch) {
      nbrOfFakeBsTracks++;
    }
    bsTree->Fill();
  }
}
```

Bibliography

[1] David Griffiths, *Introduction to Elementary Particle Physics*, John Wiley & Sons, 1987

[2] The Royal Swedish Acadamy of Science website for the Nobel Prize in physics 2004: http://nobelprize.org/nobel_prizes/physics/laureates/2004/public.html

[3] M. Wingate, *Lattice QCD and Fundamental Parameters of the Standard-Model*, the HPQCD Collaboration, Institute for Nuclear Theory, University of Washington, J.Phys.Conf.Ser.16:179-183, 2005

[4] Francis Halzen and Alan D. Martin, *QUARKS & LEPTONS: An Introductory Course in Modern Particle Physics*, John Wiley & Sons 1984

[5] G. Davatz, M. Dittmar and F. Pauss, *Simulation of a cross section and mass measurement of a standard model Higgs boson in the* $gg \to H \to WW \to l\nu l\nu$ *channel at the CERN LHC*, Institute for Particle Physics (IPP), ETH Zürich, Phys. Rev. D **76**, 032001, 2007

[6] P. Ball et al., *B decays at the LHC*, CERN-TH/2000-101, 2000

[7] UTfit Collaboration, *First evidence of New Physics in* $b \leftrightarrow s$ *Transitions*, arXiv:hep-ph/0803.0659v1, 2008

[8] Chao-Hsi Chang and Yu-Qi Chen, *Decays of the* B_c *meson*, China Center of Advanced Science and Technology and Institute of Theoretical Physics, Academia, Sinica, Phys. Rev. D **49**, 3399, 1993

[9] The CDF Collaboration, $B_c^- \to J/\psi\pi^-$ *at CDF with* $2.2fb^{-1}$, The CDF Collaboration, CDF-Note 8004, 2007

[10] ATLAS Collaboration, *Detector and Physics Performance, Technical Design Report Volume II*, CERN/LHCC/99-15, 1999

[11] F. Abe et al., *Observation of B mesons in pp-collisions at* $\sqrt{s} = 1.8TeV$, Phys. Rev. D **58** 112004, 1998

[12] The CDF collaboration, *Measurements of the B_c^+ Meson Lifetime Using the Decay Mode $B_c^+ \to J/\psi e^+ \nu_e$*, The CDF Collaboration, Phys. Rev. Lett. **97**, 012002, 2006

[13] Ajay Kumar Rai adn P C Vinodkumar, *Properties of the B_c meson*, arXiv:hep-ph/0606194v1, 19 June 2006

[14] A K Likhoded et al, *Physics of B_c-mesons*, Institute for High Energy Physics, Protvino, Moscow Region, Physics-Uspekhi **38** (1), 1995

[15] The LHC homepage: http://lhc.web.cern.ch/lhc/

[16] J. R. Catmore *$B_s^0 \to J/\psi\phi$ with LHC-ATLAS: simulations and sensitivity studies*, PhD Thesis, Department of Physics, Lancaster University, 000615919, 2007

[17] Kenneth S. Krane, *Introductory Nuclear Physics*, John Wiley & Sons, 1988

[18] A. Del Fabbro and D. Treleani, *Double parton scatterings in b-quark pair production at the CERN LHC*, 10.1103/PhysRevD.**66**.074012, 2002

[19] ATLAS Collaboration, *Detector and Physics Performance, Technical Design Report Volume I*, CERN/LHCC/99-14, 1999

[20] The ATLAS homepage: http://www.cern.ch/atlas/

[21] S. Haywood, *The ATLAS Inner Detector*, Nuclear Instruments and Methods in Physics Research A 408 (1998) 242-250, 1998

[22] ATLAS Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST preprint July 27, 2007

[23] A. Sherstnev, *Prospects for b-quark production cross section in pp collisions at the LHC*, arXiv:hep-ph/0609143v1, 2006

[24] The twiki website for B physics cross sections: https://twiki.cern.ch/twiki/bin/view/Atlas/BPhysWorkingGroupCrossSections

[25] The AAna homepage: http://jcatmore.web.cern.ch/jcatmore/AAna/AAnaMain.html

[26] The LCG Grid homepage: http://lcg.web.cern.ch/LCG/

[27] ATLAS Collaboration, *ATLAS Computing Technical Design Report*, ATLAS-TDR-017, CERN-LHCC-2005-022, 2005

[28] T. Sjöstrand et al., *PYTHIA 6.4 Physics and Manual*, arXiv:hep-ph/0603175v2, 2006

[29] The PythiaB manual homepage: http://www-theory.lbl.gov/ ianh/monte/-Generators/PythiaBModule/

[30] The ROOT homepage: http://root.cern.ch/

[31] The Particle Data Group, *Particle Physics Booklet*, Lawrence Berkeley National Laboratory, Journal of Physics G **33** 1, 2006.

[32] Private dialogue with Sergey Siviklokov.