Development of an LHC model in BDSIM to study collimation cleaning and beam-induced backgrounds at ATLAS

Stuart Walker

Department of Physics

Royal Holloway, University of London

A thesis submitted to the University of London for the degree of Doctor of Philosophy

25th August 2020

DECLARATION

I confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the document.

Signed: 

Date: 25th August 2020

The struggle itself toward the heights is enough to fill a man's heart. One must imagine Sisyphus happy.

ALBERT CAMUS

ABSTRACT

The Large Hadron Collider (LHC) is at the frontier of high energy physics. At 27 km in circumference and operating at the highest achieved energy to date at 6.5 TeV, it is reliant on cold superconducting magnets throughout the machine to steer and control the beam. These cold superconducting magnets are extremely sensitive to heating from the beams that they steer. Even a billionth of a beam deposited on a single magnet can cause it to quench and lose its superconductivity, resulting in lengthy downtime or even component damage. For this reason the LHC features a number of systems to protect the machine from itself. For example, around 3600 beam loss monitors (BLMs) are placed around the ring, and if they detect losses above some threshold, the beam is dumped safely. Also aiding the machine's protection is its collimation system that is used to remove halo beam particles safely from circulation. Simulating the collimation system is vital to optimising its function, and in this thesis it is simulated using the novel accelerator code, Beam Delivery Simulation (BDSIM), and compared with existing comparable simulations. Additionally, individual BLMs are placed in the model, the BLM dose is simulated and compared one-to-one with measured BLM data from a dedicated qualification run.

The general purpose experiments are vulnerable to beam-induced backgrounds, which originate from upstream proton losses that generate secondary showers that can reach the detectors. This is particularly problematic due to the propensity for such backgrounds to mimic signals in the search for novel physics. They must be understood and mitigated in the physics analysis. In this thesis such background sources are simulated by building a detailed model of the beam line upstream of the ATLAS detector, which has been used to simulate various scenarios and the results are

compared with existing simulations at an interface plane with the detector. They are then passed into a dedicated ATLAS simulation where they are compared with real data from recent runs in which the beam-induced background rate was deliberately increased by raising the gas pressure in upstream sections of the beam pipe.

ACKNOWLEDGEMENTS

I have become heavily indebted to a large number of people whilst working on the PhD. The person to whom I am most indebted is my supervisor, Stephen, for his boundless support and guidance. I am grateful to him for taking me on as his student, and I am grateful to have had a supervisor who has such patience, and I am grateful for the endless supply of encouragement that he has offered. Your inhuman ability to work on a thousand different things at once is inspiring to me.

Little of this work would exist without Laurie. I really cannot understate just how much he has done for me and to help me, and I am heavily indebted to him for always trying so hard to make me a better physicist. I have developed a lot in the past few years and it's in no small part thanks to him.

Stewart has supported me in a number of ways for which I am grateful. In group meetings his sometimes harsh but always fair criticisms have helped me to become a better physicist. He is also a very kind man, and his kindness was of great help to me at the times I felt most demotivated.

I am grateful for having met my friend Helena. I don't think my third year would have been any fun at all without her. Besides being such a pleasant and fun person, her infectious enthusiasm has helped to make me a better physicist.

I must thank Andrey for our discussions on LHC collimation. I must thank Will for always listening to me and always being generous with his time above and beyond what I could reasonably expect. Thank you to Lewis, the full-time shopkeeper and part-time particle physicist for sustaining me with your delicious confectionery whilst I wrote. Thank you to everyone else in the department who has put up with me for the past few years.

Lastly, I thank my parents for everything.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER

ONE

INTRODUCTION

The Large Hadron Collider (LHC) at CERN is at the forefront of particle physics, with an unprecedented design centre-of-mass-energy and luminosity of 14 TeV and $1 \times 10^{34}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$, respectively. Capable of colliding both protons and ions, this energy enables it to probe new and rare physics, culminating most prominently to date with the discovery of the Higgs boson in 2012 [1, 2]. The design and continuing operation of such a machine introduces unique challenges, primarily due to the unprecedented stored energy in each beam. At 386 MJ per beam, this is an increase by a factor of around 200 [3] over the previous record holders, HERA [4] and the Tevatron [5]. With a circumference of 27 km, it is dependent on superconducting magnets to steer and control its high-energy beam. The superconducting magnets are cooled to 1.9 K, and extremely sensitive to heating from the high-energy circulating beams that are contained within. Protons lost within the machine may cause the superconducting magnets to quench and becoming normal-conducting. This may result in permanent machine damage, thus necessitating an automatic beam dump, and in either case will reduce machine availability due to slow cryogenic cooling. Protecting the machine from such losses requires a dedicated collimation system in which protons that would otherwise collide with these sensitive parts of the machine, are removed safely. Additionally, protons impacting with residual gas molecules in the beampipe vacuum may result in showers that can travel downstream into the experimental insertions, resulting in experimental backgrounds.

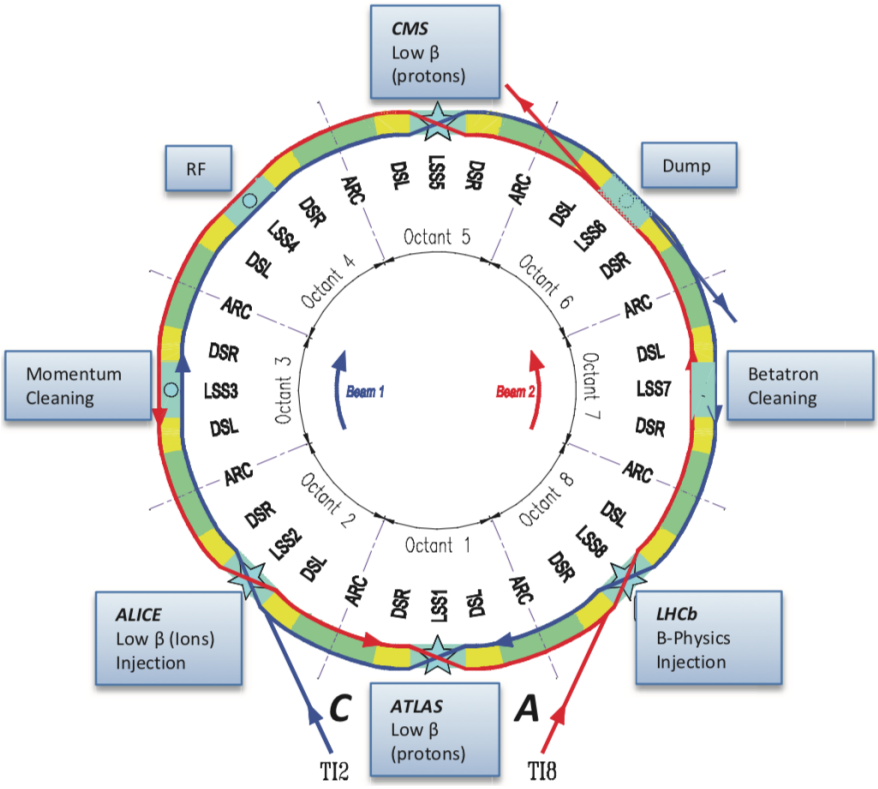


Figure 1.1: The LHC, with the insertions and sections labelled. Reproduced from [10] (CC BY 3.0).

1.1 The Large Hadron Collider

The LHC at CERN has to date reached a peak beam energy of 6.5 TeV, with a nominal design energy of 7 TeV. At 27 km in circumference it consists of eight arcs and eight straight insertion regions (IRs). Each of the insertion regions is dedicated to performing a different task, either vital to the operation of the machine, or a point in which the two counter-circulating beams are brought into collision, where a detector is housed. Figure 1.1 shows the location and functionality of each of the insertion regions. There are four insertion regions with detectors. The main, general purpose detectors, ATLAS [6] and CMS [7], are situated at IR1 and IR5, respectively. Each beam, labelled "1" and "2", are injected into the ring at IRs 2 and 8, respectively. The specialised experiments ALICE [8] and LHCb [9] are also contained within these insertions. The beam is accelerated in IR4, where the superconducting radio frequency (SRF) cavities are situated, and beam extraction is performed in IR6. Finally, the dedicated collimation system is housed in IRs 3 and 7. The three insertions particularly relevant to this thesis are IRs 1, 3 and 7.

Table 1.1: LHC machine parameters for 2018 [11] and nominal settings [12].

Parameter	2018	Nominal	Units
Beam energy	6.5	7	TeV
Stored energy per beam	312	362	MJ
Peak luminosity	2×10^{34}	10^{34}	$\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$
β^*	25	55	cm
Normalised transverse emittances	1.80	3.75	$\mathrm{mm} \cdot \mathrm{mrad}$
Protons per bunch	1.10	1.15	/
Total bunches	2556	2808	/
Bunch spacing	25	25	ns

The key operational parameters of the LHC are shown in Table 1.1. The 2018 parameters are quoted as this is the last year in which the LHC ran, it is currently shut down for upgrades as part of Long Shutdown 2 (LS2).

1.1.1 Collimation

The collimation system is vital to the LHC's operation. The nominal stored energy, at 362 MJ per beam [12], introduces unique challenges. The LHC features 1232 superconducting main bending magnets, and as little as $1\,\mathrm{mJ\,cm}^{-3}$ from the beam is sufficient to cause a superconducting magnet to quench and become normal conducting [13]. Compounding this is the 10 GJ stored within the superconducting magnets, which, if released in an uncontrolled manner, could cause severe damage to the accelerator.

Beam losses in any machine are unavoidable, and as the LHC is so sensitive to such losses, it features a dedicated protection system. Two key features are the beam loss monitor (BLM) system and the collimation system. Over 3600 BLMs are placed around the LHC to detect abnormal beam losses. If the beam losses exceed some threshold, then the beam is dumped. Two of the insertion regions (IR3 and IR7) house the collimation system in which beam protons outside the permitted spatial or momentum envelopes are physically intercepted and removed from circulation. The momentum cleaning insertion in IR3 is used to remove off-momentum protons from the beam. In IR7 is the betatron cleaning insertion where protons at large transverse distances from the beam are cleaned from the beam. Common to both is the concept of a hierarchy, consisting of sets of collimators of increasing aperture sizes, from the global

bottleneck in the primary collimator, which should be the first point of contact for any such proton, down to the secondary and tertiary collimators (TCTs). This multistage system is required because the individual collimators are not perfect absorbers, and protons will leak downstream.

Understanding the effectiveness of the collimation system is important to ensure the continued operation of the machine. This is especially true as the LHC's operation is driven to more and more adverse conditions, not least of which is the increase of the beam energy to the nominal value at 7 TeV. In the longer term, the LHC will be upgraded to the High Luminosity Large Hadron Collider (HL-LHC) [14], where beam-induced backgrounds (BIB) will be higher due to the larger bunch intensities [15], with a doubling in the stored beam energy and larger expected losses, the collimation system will be an even more vital component of the machine.

1.2 Beam-induced backgrounds

The detectors in the LHC are vulnerable to backgrounds originating from the beam upstream of the interaction point (IP), referred to as BIB. One source of these backgrounds is from beam protons impacting upon the collimators placed around 140 m upstream of the experimental IPs, which may then shower and travel downstream through the accelerator at large radii, and then are detected in the experiments. These collimators around the experimental IPs are referred to as TCTs, because they are primarily designed to intercept the tertiary beam halo. The tertiary beam halo consists of beam protons which have escaped the main collimation insertions, which could otherwise hit the superconducting final-focus quadrupoles if not for the TCTs.

Another source of BIB are from beam-gas interactions. These interactions with the residual gas molecules in the beampipe vacuum may be either elastic or inelastic. In the former case, protons at any point in the machine may scatter off of a gas molecule, sending the beam proton to a larger transverse amplitude. These protons may then be removed from the beam in one of the collimation insertions, contributing to the beam background rate in the IPs via any resulting tertiary halo. Alternatively, if the elastic beam-gas collision occurs between one of the collimation insertions and the experimental IPs, the scattered proton may instead impact directly on one of the TCTs,

contributing to the BIB rate [16, 17]. Alternatively, in the event of an inelastic beam-gas collision, a resulting particle shower may travel downstream parallel to the beampipe, before ultimately being detected in the detector. The contribution to the BIB rate is typically a local one, no more than 550 m [17] upstream of the IP. Elastic beam-gas, however, is a global effect, as the elastically scattered protons may travel for many turns before eventually contributing to the background rate in one of the IPs.

1.3 The ATLAS detector

The ATLAS detector [6] is one of the two general-purpose detectors at the LHC. It is situated in IR1 and is 44 m in length and 25 m in diameter, with almost full 4π sr coverage. The detector is shown in Figure 1.2. It consists of several subdetectors, including the inner detector which is situated in a 2 T solenoidal field, with an electromagnetic calorimeter beyond the inner detector, followed by a hadronic calorimeter, with a muon detector on the outermost layer. It is designed primarily for the study of proton-proton collisions at 14 TeV and an instantaneous luminosity of $1 \times 10^{34}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$, but can also be used to study heavy ion collisions [6]. The ATLAS detector is at the forefront in the search for novel physics.

1.3.1 Beam-induced backgrounds in the ATLAS detector

Beam-induced backgrounds in the ATLAS detector are relatively rare in absolute terms, but in the search for new and rare physics events, they can provide a significant source of background. As mentioned above, these backgrounds originate from the machine, upstream of the detector. They then travel towards the detector at large radii, and are detected in the ATLAS subdetectors, resulting in characteristic and problematic background signals in the detectors. The magnetic elements upstream of the detector give rise to an azimuthal asymmetry, which in ATLAS will be recorded as "missing energy"—a feature otherwise indicative of novel physics. Additionally, these backgrounds can be considerably out of time with the collisions, thus faking potentially long-lived particles decaying in the detector. Understanding beam-induced backgrounds is therefore extremely important if they are to be effectively removed from analyses searching for new physics.

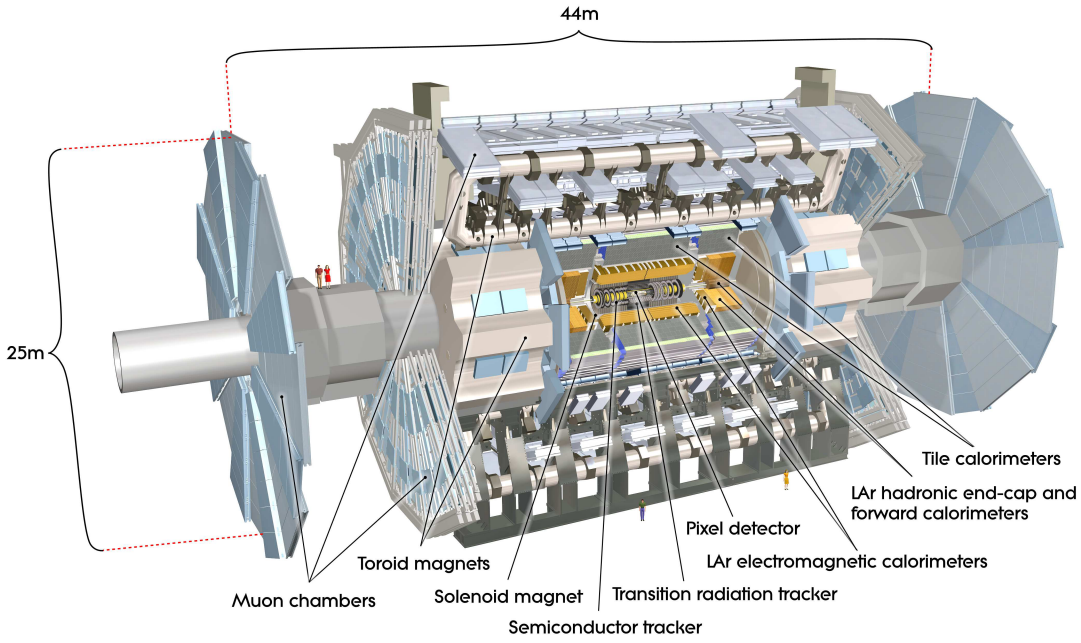


Figure 1.2: A cutaway view of the ATLAS detector [6].

1.4 Beam Delivery Simulation

Simulating the LHC collimation system and beam-induced backgrounds both have similar requirements. Particle-matter interactions are required in the collimators and to simulate the passage of particle showers through the accelerator lattice. Furthermore, accelerator tracking is needed for the particles to be accurately propagated through the lattice. Beam Delivery Simulation (BDSIM) [18] is a novel Monte Carlo accelerator simulation code that meets both of these requirements in that it is designed to bridge the interface between accelerator particle tracking simulations, and the particle-matter interactions of particle physics simulations. This is achieved by building BDSIM on top of the Monte Carlo particle physics library Geant4. In essence, it builds a particle accelerator in Geant4, giving it access to all of Geant4's particle physics routines. BDSIM's implementation Geant4 is then supplemented with accelerator physics tracking routines and other features to optimise its use for simulating particle accelerators. BDSIM has been used to generate many of the results presented throughout this thesis.

Whilst BDSIM has been around for nearly 20 years, with the earliest published results dating to 2002 [19], in recent years it has undergone a major redevelopment for new applications. One such development is its use for circular machines, a feature that was

added relatively recently, in 2014. Since that time BDSIM has been used to produce results for the LHC [20–23] and also demonstrate some of its capabilities with toy circular models [18, 24]. However there has not yet been a truly rigorous study of BDSIM's application to circular accelerators in general, and to the LHC in particular. Such a study should ideally feature comparisons to existing simulations, as well as to real data. This thesis involves all of these matters: a study of BDSIM's tracking for the LHC, a loss map comparison with SixTrack, as well as a direct comparison with BLM data from a recent dedicated loss map run.

BIB is an ideal application for BDSIM, as both the accelerator and particle-matter interactions must be included in any such simulation. Any study of BIB will necessarily require a combined simulation of particle-matter interactions with accelerator tracking. BDSIM is particularly useful in such scenarios as it has been designed from the ground up to combine particle-matter interactions with the automatic building of Geant4 accelerator models. In this thesis results from the application of BDSIM to the study of BIB in the LHC are presented and compared with existing simulations.

1.5 Thesis overview

In the next chapter, Chapter 2, the necessary accelerator theory is described. In Chapter 3 the variety of simulation tools used to produce the results in this thesis, and the importance of geometrical descriptions to such simulations is described. Finally, a novel Python package for translating geometry from the widely-used Monte Carlo particle physics code, FLUKA [25], to the geometry format used in BDSIM is introduced, which was used to generate the detailed IR1 model in Chapter 5. This package is particularly useful due to the widespread usage of FLUKA meaning that there is a lot of detailed geometries in existence described in FLUKA. Most of these geometries are now usable in BDSIM.

This thesis consists of three separate studies. In Chapter 4, the performance of the LHC collimation system in 2018 is studied with the novel accelerator code BDSIM, and benchmarked in detail against similar simulations using the standard LHC collimation code, SixTrack. The BLM response is simulated and directly compared with BLM data from a dedicated LHC run in 2018 used for studying the efficiency of the collimation

system. Finally, novel and unique results from BDSIM are presented showing some of the fuller capabilities of BDSIM.

Next, in Chapter 5, a highly detailed model of the region upstream of the ATLAS detector in IR1 is described in full. The model was used to simulate backgrounds from both beam-gas interactions as well as beam halo. This was achieved by simulating the initial inelastic beam-gas or proton-collimator interaction and then tracking the resulting secondary showers up to the interface plane—the plane where the detector is deemed to begin. To compare to ATLAS data, the results at the interface plane are then passed to a separate, dedicated ATLAS detector simulation. These data were compared with existing comparable simulations from other, widely-used codes.

In Chapter 6 BIB in the Pixel Detector is studied. This is achieved by examining pixel clusters associated with background triggers. In particular BIB from recent dedicated pressure bump runs, in which the rate of BIB was deliberately inflated in a region upstream of the ATLAS detector, are examined as these runs provide a particularly clean source of such backgrounds. Finally, the products at the interface plane from the BDSIM simulations of BIB were passed through the ATLAS Geant4 model. Geant4 model hits were compared with the distributions at the interface plane and discussed.

CHAPTER

TWO

ACCELERATOR PHYSICS

Beam dynamics is the study of the motion of charged particle beams. In this section the accelerator physics theory required for this thesis is stated.

2.1 Luminosity

For a particle collider like the Large Hadron Collider (LHC), the luminosity in the experimental insertion regions (IRs) is a key parameter. The luminosity $\mathcal{L}$ at the interaction points (IPs) determines the number of interactions per second, and is related to the interaction rate $\frac{\mathrm{d}N}{\mathrm{d}t}$ as

$$\frac{\mathrm{d}N}{\mathrm{d}t} = \mathcal{L}\sigma\,, \tag{2.1}$$

where σ is the cross section for a given interaction or set of interactions [26]. The luminosity can be related to beam parameters with the following proportionality

$$\mathcal{L} = f\frac{n_1 n_2}{4\pi\sigma_x\sigma_y}F\,, \tag{2.2}$$

where f is the number of bunch crossings per second, n_1 and n_2 the respective beam intensities, and σ_x, σ_y the transverse beam sizes at the IP. F is a geometric reduction factor (i.e. less than unity) stemming from reductions in the luminosity due to effects such as crossing angles and beam offsets.

The integrated luminosity is proportional to the total number of events of interest over

a period of time. Therefore, in the search for new, rare events, maximising the integrated luminosity is desirable. This means maximising the luminosity to record the greatest amount of data to detect novel physics, with the extra benefit of minimising the necessary running time of the accelerator.

2.2 Transverse motion

Electromagnetic fields are used to accelerate, steer, and control charged particle beams. The force on a particle in a combined electromagnetic field can be expressed as the Lorentz force,

$$\mathbf{F} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B}) \,, \tag{2.3}$$

where q is the particle charge, $\mathbf{E}$ the electric vector field, $\mathbf{v}$ the charged particle velocity vector and $\mathbf{B}$ the magnetic vector field [27]. This equation shows that for a pure magnetic field, an increase in velocity perpendicular to the magnetic field will result in an increased force perpendicular to the direction of motion. The particle speed saturates as it approaches the speed of light, and the Lorentz force saturates with it. This means that to maintain the bending of a charged particle over a given radius, the magnet field, B, must increase. The bending of a given charged particle beam in a magnetic field is expressed with the magnetic rigidity, $B\rho$, as

$$B\rho = \frac{p}{q} \,, \tag{2.4}$$

where B is the magnetic field required to bend the particle in a radius of ρ, p is the particle momentum, and q is the particle charge. This is an important equation in accelerator physics as it defines, for a given accelerator circumference, the magnetic field needed in the dipoles of a circular collider to steer a desired beam energy.
The coordinate system most commonly used in accelerator physics moves with the *reference* particle, which is called the Frenet-Serret coordinate system. The reference particle refers to an ideal particle that travels through the accelerator perfectly on-axis, with precisely the correct amount of energy, and in a machine without any imperfections such as misalignments or field errors. In reality, such an ideal machine does not exist, and a particle beam will consist of particles spread out transversely and

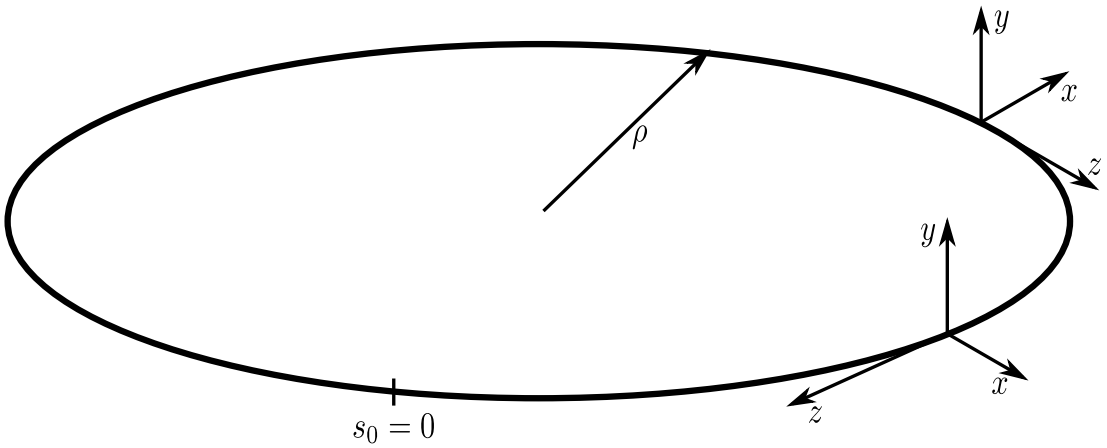


Figure 2.1: The Frenet-Serret coordinate system which is commonly used in accelerator physics. The transverse positions and momenta, (x, p_x, y, p_y) move along the path s with the reference particle.

longitudinally in space, as well as with a range of particle energies. The coordinate system is demonstrated in Figure 2.1

As the beam consists of many different positions and transverse momenta, in free space it will diverge. Quadrupoles are used in accelerators to constrain the beam, preventing it from diverging due to the spread in transverse momenta. Quadrupoles are similar to optical lenses, although they focus only in one plane and defocus in the other. Quadrupoles which focus in one plane are combined with ones which focus in the other such that the net effect is focussing in both planes. The combination of these focussing and defocussing quadrupoles in this way is referred to as a focussing-defocussing (FODO) cell. The strength of a quadrupole is expressed in terms of the normalised quadrupole coefficient as

$$k_1 = \frac{1}{B\rho}\frac{\mathrm{d}B_y}{\mathrm{d}x}\,. \tag{2.5}$$

Particles will tend to oscillate around the reference trajectory for two reasons. Firstly because of the dipoles, an initial offset in x (assuming the energy is the same as the reference) will result in a particle tracing out a circle with the same circumference as the reference particle, albeit slightly offset. The primary source of transverse oscillations will stem from the quadrupoles, where the alternating focussing and defocussing will drive oscillations around the reference trajectory (which never sees any quadrupolar field). Figure 2.2 shows these transverse oscillations around the reference

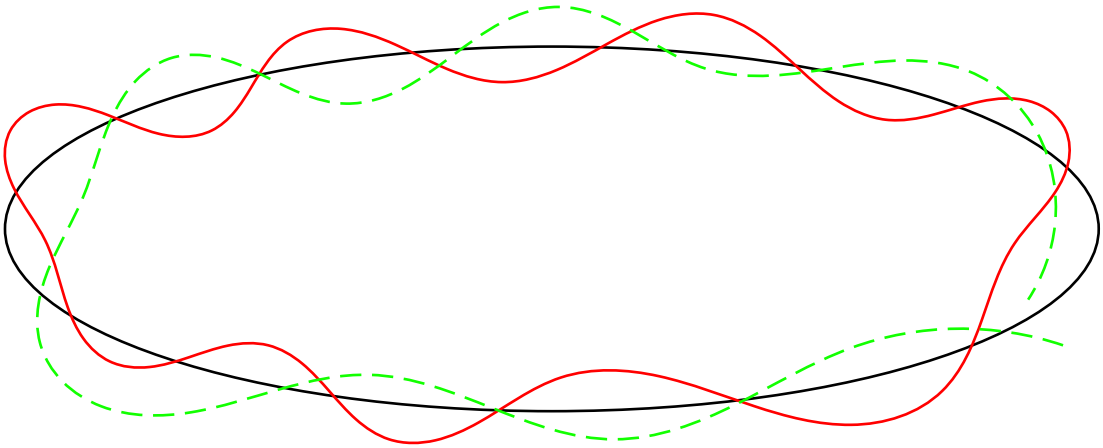


Figure 2.2: Betatron oscillations with integer (red) and non-integer period (green) around the reference trajectory (black).

trajectory, which are referred to as *betatron oscillations*.

Equation 2.6, called Hill's equation, describes the periodic transverse motion of a particle in a lattice of dipoles and quadrupoles,

$$\frac{\mathrm{d}^2x}{\mathrm{d}s^2} + K(s)x = 0\,, \tag{2.6}$$

where x is the transverse position of the particle with respect to the reference trajectory, s is the path length, and $K(s)$ is a path-dependent coefficient which determines the local restoring force for the oscillations. This equation has solutions of the form in Equation 2.7. The notable features are that the amplitudes of the oscillations are functions of beamline path length, s, as well as some phase μ.

$$x(s) = \sqrt{J_x}\sqrt{\beta(s)}\cos(\mu(s) + \mu_0) \tag{2.7}$$

where β is referred to as the beta function, and is a position-dependent amplitude, and J_x is a constant of motion, called the particle action, which the varying amplitude is scaled by.

The *phase advance*, μ, is related to $\beta(s)$ as,

$$\Delta\mu = \int_{s_0}^{s_1} \frac{1}{\beta} ds\,, \tag{2.8}$$

and the phase advance over the whole length of the ring C, divided by 2π, is the

tune, Q,

$$Q = \frac{1}{2\pi} \oint \frac{1}{\beta} ds \,, \tag{2.9}$$

which is the number of betatron oscillations per turn. This is generally chosen to not be an integer, as an integer tune will result in the particle arriving at the same point in x at each point in s for every turn. This means it will see the exact same imperfections at every point of every turn, which is fundamentally unstable.

In addition to β, two additional quantities are defined, which together are referred to as the Twiss parameters,

$$\alpha = -\frac{1}{2} \frac{\mathrm{d}\beta}{\mathrm{d}s} \,, \tag{2.10}$$

$$\gamma = \frac{1 + \alpha^2}{\beta} \,, \tag{2.11}$$

which are all related to one another, with J_x as an invariant emotion, x the horizontal offset and x' the horizontal divergence by the equation

$$J_x = \gamma x^2 + 2\alpha x x' + \beta x'^2 \,. \tag{2.12}$$

Equation 2.12 is the equation of an ellipse, and the motion of a particle governed by Hill's equation is an ellipse in phase space. The geometric relationship between the various Twiss parameters and the invariant of motion J_x, referred to as the particle action, is shown in Figure 2.3. In general, particles undergoing linear transverse motion in accelerators trace ellipses in phase space, as they travel through the lattice, which is shown in Figure 2.3. The tune, Q is the number of times the ellipse is traversed per turn of the ring.

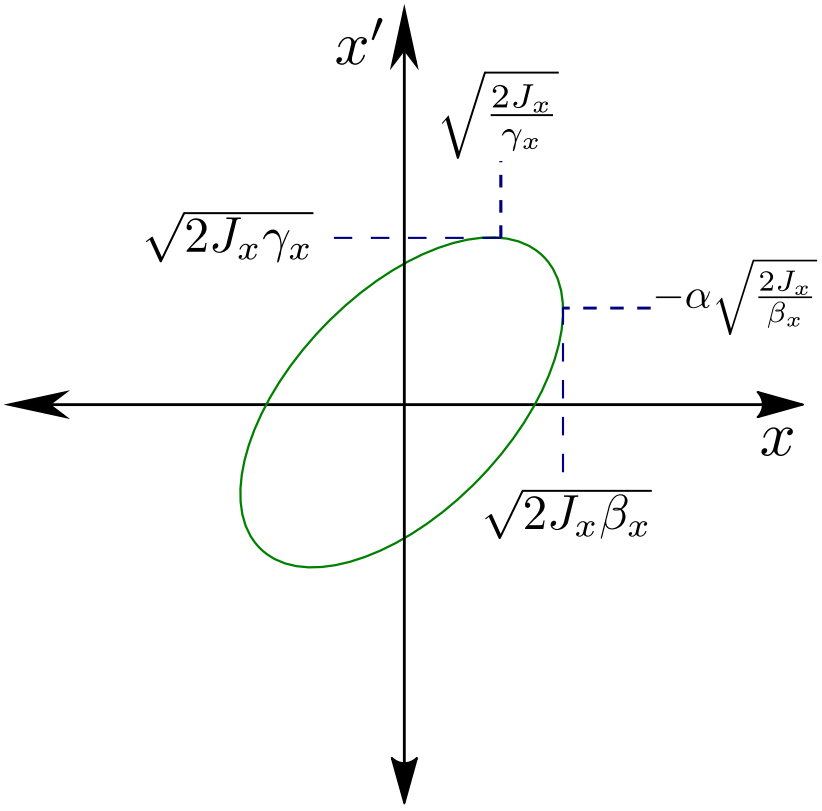


Figure 2.3: The geometric relationship between the horizontal Twiss parameters, α_x, β_x and γ_x, the horizontal particle offset x, the horizontal divergence x' and the invariant of motion J_x for a single particle.

2.3 Dispersion

In the previous section, the particle was assumed to be perfectly on-energy, however, this is not realistic, and a charged particle beam will consist of particles with a range of energies. In addition to the transverse coordinates (x, x', y, y') (horizontal and vertical offsets and divergences), it is useful to define δ, which is the normalised momentum deviation,

$$\delta = \frac{p - p_0}{p_0} = \frac{\Delta p}{p_0}\,, \tag{2.13}$$

where p_0 is the energy of the reference particle. These coordinates, (x, x', y, y', δ) form a 5D description of the particle dynamics.[1]

The energy deviation introduces a coupling between the transverse and longitudinal motion. Particles with $\delta > 0$, i.e. with energies greater than the reference particle, will orbit with larger amplitudes due to their increased momenta. Conversely, particles with $\delta < 0$ will tend to travel shorter paths around the ring as they are bent more by the magnets due to their lower momentum.[2] These off-momentum protons will undergo betatron oscillations centred around their new orbits.

The exact relation between the momentum deviation is described by introducing the dispersion function, $D(s)$,

$$x_D = D(s)\frac{\Delta p}{p_0}\,, \tag{2.14}$$

which defines the new orbit around which a particle will undergo betatron oscillations. The new position x, accounting for dispersion, is thus

$$x(s) = \sqrt{J_x}\sqrt{\beta(s)}\cos(\mu(s) + \mu_0) + D(s)\frac{\Delta p}{p_0}\,. \tag{2.15}$$

which is simply the sum of Equations 2.7 and 2.14.

[1] A full 6D treatment is omitted here as BDSIM is currently limited to 5 dimensions.

[2] This is only true for relativistic particles, at lower velocities, a decrease in energy will correspond to a longer length, not a shorter one, and vice versa. This is due to the saturation of the perpendicular magnetic field as the particle velocity approaches c.

2.4 The accelerator Hamiltonian

Tracking charged particles through accelerator components can be achieved in a number of ways. The most general approach might involve numerically integrating the motion of a particle in an arbitrary field, perhaps with the use of a field map. However, this can be slow and where possible it is preferable to use a closed-form solution to transport a particle from the beginning of a component through to the other side. This is particularly relevant in cases where there are tens of thousands of components and up to hundreds of thousands of turns. For such simulations, speed is absolutely necessary if rapid prototyping is to be feasible, or rare phenomena are to be studied in a reasonable amount of time.

Generating closed form solutions for stepping from one end of a component to the other will typically start with defining the accelerator Hamiltonian. The accelerator Hamiltonian in a straight coordinate system (it is necessary to express it differently in a curved component such as a dipole) is defined [28] as follows,

$$\mathcal{H} = \frac{\delta}{\beta_0} - \sqrt{\left(\delta^2 + \frac{1}{\beta_0} - \frac{q\phi}{cP_0}\right)^2 - (p_x - a_x)^2 - (p_y - a_y)^2 - \frac{1}{\beta_0^2\gamma_0^2}} - a_z \,, \tag{2.16}$$

where the subscripted zero refers to the variables corresponding to the reference particle with the usual meanings, the a_x, a_y, a_z are the magnetic vector potentials normalised with respect to the nominal magnetic rigidity (Equation 2.4), i.e. $\mathbf{a} = \frac{q}{p_0}\mathbf{A}$ [28]. The transverse dynamical variables are defined as (x, p_x), (y, p_y), where p is the canonical momentum in the respective plane, but normalised with respect to the reference momentum, P_0. The longitudinal dynamic variables (z, δ) are the longitudinal deviation from the reference trajectory and the energy deviation from the reference particle, respectively. It is also important to note that the independent variable here is path length s, not time, which are the Frenet-Serret coordinates shown in Figure 2.1. Stepping a particle through an accelerator component most generally involves combining the Hamiltonian in Equation 2.16 with Hamilton's equations,

$$\begin{aligned} \frac{\mathrm{d}p_i}{\mathrm{d}t} &= -\frac{\partial \mathcal{H}}{\partial q_i} \,, \\ \frac{\mathrm{d}q_i}{\mathrm{d}t} &= +\frac{\partial \mathcal{H}}{\partial p_i} \,, \end{aligned} \tag{2.17}$$

where q_i and p_i are the conjugate variables for the horizontal, vertical, and longitudinal dimensions, and then solving the resulting equations of motion. However, deriving a closed form solution to Hamilton's equations with the Hamiltonian defined in Equation 2.16 is generally not possible due to the presence of the square root, so typically this is expanded with a Taylor series and then truncated after the second term. This truncated approximation is valid only for small quantities in the variables and this is what necessitates normalising the momenta with respect to the reference momentum, thus ensuring these variables are small. This makes solving Hamilton's equations much simpler and enables closed form solutions for common accelerator components but means that the solutions are only valid for small quantities in the dynamical variables. This is called the paraxial approximation. It is valid where particles are at small angles and near the reference trajectory. This is generally a safe approximation to make as particles which are at very large angles (p_x, p_y), transverse positions (x, y), very far off-energy (δ), or situated far ahead or behind the reference particle (z) make up a very small number of the total particles in a beam. Also, such particles are likely to be lost within the machine very quickly. Where it's desirable to simulate such particles, one such solution is to fall back on numerical integration, which is much slower but will remain accurate.

2.5 Liouville's theorem

Particle dynamics which are governed by Hamilton's equations will obey Liouville's theorem, which states that particle density in phase space is conserved along all trajectories. In accelerator physics this conserved quantity is referred to as the *emittance*, which is proportional to the volume of phase space occupied by the beam, defined for each of the three pairs of dynamical variables. It is important to mention that the beam emittance is only conserved where there is no time dependence on the Hamiltonian, i.e. when the beam energy is constant. It would not be physically accurate for the emittance to be conserved when the beam is being accelerated, which is often the case in particle accelerators. However, emittance should be preserved when, for example, a particle is bent in a dipole (ignoring the effects of synchrotron radiation). Respecting Liouville's theorem is important in tracking codes, as if it is not, unphysical,

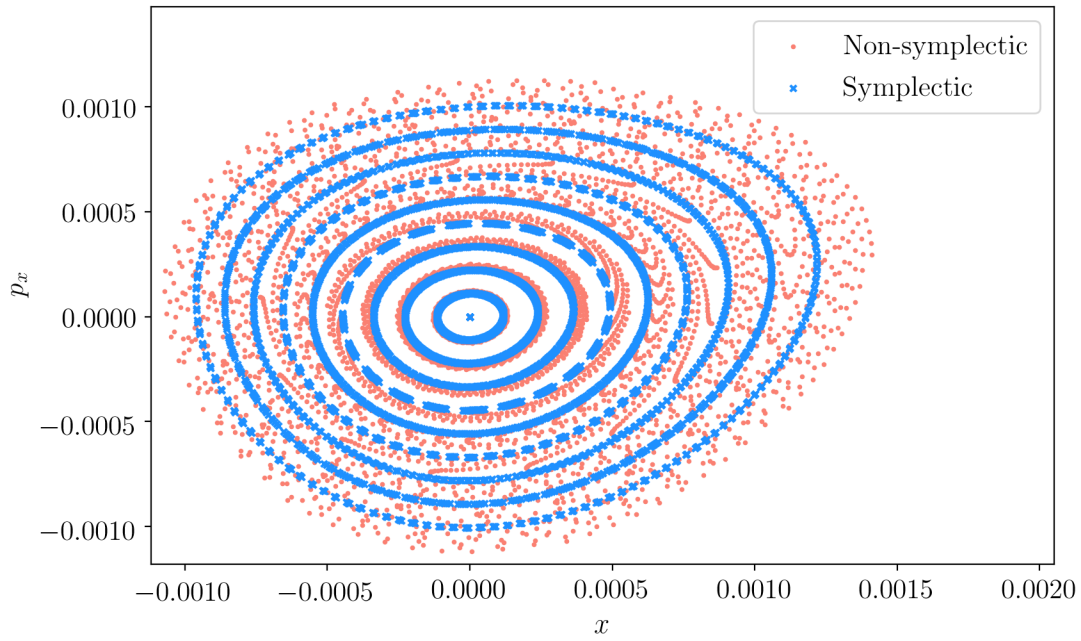


Figure 2.4: Poincaré sections for a lattice consisting of a phase advance of $\mu_x = 0.26\pi$ followed by a single sextupole with normalised strength $k_2 = -6 \times 10^3\,\mathrm{m}^{-3}$. The input particle distributions were tracked 2000 times through the lattice.

spurious beam growth or shrinking can occur. As all solutions to Hamilton's equations will respect Liouville's theorem, this motivates the Hamiltonian approach. Figure 2.4 shows the difference between a symplectic integration and non-symplectic integration for a sextupole.

2.6 Collimation

Beam particles far from the beam core are referred to as the beam halo. Particles in the core of the beam tend to travel along stable trajectories, whereas particles far outside of this region will undergo chaotic motion and be ejected from the machine.[3] The beam halo is populated continuously from the core via a number of processes, including intrabeam scattering, beam-gas interactions, and from collisions in the experimental IPs. The populating of the beam halo from the beam core is unavoidable. Collimation systems are necessary where the stored beam energy is large or the accelerator components are sensitive to heating from beam losses. These large-amplitude protons are safely intercepted by introducing a global aperture

[3]The region in phase space in which particles travel along stable trajectories is referred to as the dynamic aperture.

bottleneck in the machine, in the form of large blocks of material, called jaws. Pairs of these jaws (either side of the beam) placed close to the beam are referred to as primary collimators. As the diffusion of the beam halo is slow, the impact parameter, which is the transverse distance from the point of impact and the edge of the jaw, on the primary collimator is small. The small impact parameter means that a sizeable amount of the beam particles intercepted by the primary collimator, referred to as the primary halo may leak and re-enter the beampipe. These particles leaking from the primary collimator are referred to as the secondary halo and additional collimators downstream of the collimators to intercept the secondary halo may be necessary. This is referred to as a two-stage collimation system.

A simple 1D two-stage collimation system can explained by starting with a definition of the normalised coordinates X X'

$$\begin{pmatrix} X \\ X' \end{pmatrix} = \frac{1}{\sigma_x} \begin{pmatrix} 1 & 0 \\ \alpha_x & \beta_x \end{pmatrix} = \begin{pmatrix} x \\ x' \end{pmatrix}, \tag{2.18}$$

where σ_x is the transverse horizontal beam size, and α_x and β_x are two of the three transverse Twiss parameters. These normalised coordinates are defined as they reduce betatron oscillations to simple harmonic motion. That is to say, in this system advancing through the lattice is equivalent to a simple rotation in phase space. Also defined are the normalised collimator openings n_1 (primary) and n_2 (secondary). Figure 2.5 shows the operating principles of a 1D two-stage collimation system. In Figure 2.5 a primary beam halo particle will travel with normalised amplitude n_1 before impacting on the primary collimator with $X = n_1$. The particle is then given an instantaneous kick, k, which lies somewhere on the orange line, elevating it to a larger orbit in normalised phase space. If the angular kick is sufficient, it will be kicked onto an orbit with a normalised amplitude of n_2, sufficient for it to be intercepted by the secondary collimator after some amount of phase advance. This minimum kick, k_c, can be derived as

$$k_c = \sqrt{n_2^2 - n_1^2}\,. \tag{2.19}$$

Depending on whether the kick $k > k_c$ is either positive or negative, the phase advance, $\Delta\mu$ before that particle has a transverse amplitude of $X > n_2$ such that it can be

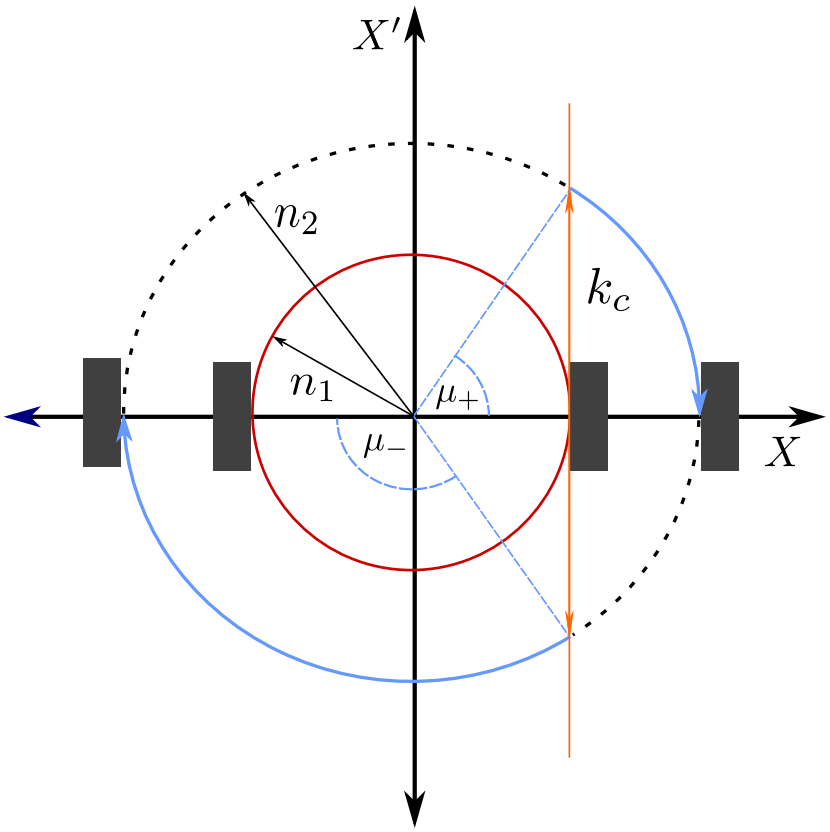


Figure 2.5: A two-stage, 1D collimation system in normalised phase space.

intercepted by the secondary collimator is different. For this reason there needs to be more than one secondary collimator following the primary collimator, one at a position corresponding to $\Delta\mu = \mu_+$ and another at $\Delta\mu = \mu_-$. This phase advance for the positive case is determined by the primary collimator opening and the kick, k_c,

$$\mu_+ = \arctan\left(\frac{k_c}{n_1}\right) = \arctan\left(\frac{\sqrt{n_2^2 - n_1^2}}{n_1}\right) , \tag{2.20}$$

or

$$\mu_+ = \arccos\left(\frac{n_1}{n_2}\right) , \tag{2.21}$$

with

$$\mu_- = \pi - \mu_+ , \tag{2.22}$$

which define the optimum positions to place the secondary collimator with respect to the primary. The positions of the collimators are linked to the optical parameters local to the collimation insertion, as well as the choice of the ratio of primary to secondary collimator opening, n_2/n_1. These equations allow one to define a two-stage collimation

system in which the secondary collimator will not intercept beam particles which have not first been impacted on the primary collimator. This is an important detail as otherwise the secondary collimator would effectively be a primary collimator, which is insufficient for high-energy, high-intensity machines such as the LHC due to resulting leakage.

In reality, the beam halo will be populated in both dimensions, as well as with coupling between the dimensions, necessitating additional primary collimators, including skew (at angles between the vertical and horizontal planes). The full treatment of the theory of two-stage collimation systems with these additional considerations can be found in [29, 30].

In addition to removing beam particles with large betatron amplitudes, it is also necessary to remove off-momentum particles, as these may otherwise be lost in highly dispersive regions of the machine. The operating principle of a momentum cleaning insertion is very similar to that of betatron cleaning. The main extra consideration is the orbit offset due to the dispersion, which is in addition to any betatron oscillations. Also, because the vertical dispersion is typically negligible, only a single primary collimator, in the plane of bending, is needed.

In addition to the two-stage system, there are a number of other methods, namely crystal collimation [31, 32] and hollow electron lens collimation [33]. In both cases, the aim is to provide a much larger transverse kick than in the system described above.

2.7 Conclusion

In this chapter, the theory of accelerator physics required to understand the studies presented in the rest of the thesis was stated and described. Understanding the use of maps to propagate particles around accelerators is important to understand how BDSIM and other accelerator codes work. The theory of two-stage collimation system has been described, which is pertinent to Chapter 4, where the LHC collimation system is studied, which has two multi-stage collimation systems. This is also relevant to Chapter 5, where IR1 is modelled to study beam-induced backgrounds (BIB) in ATLAS. One source of BIB is tertiary halo, which is the halo which leaks from the main cleaning insertions.

a

CHAPTER

THREE

ACCELERATOR PHYSICS CODES AND THEIR APPLICATIONS

Particle accelerators can be composed of thousands of different components, from bending magnets, to quadrupoles, radiofrequency cavities, collimators, higher-order multipoles, and more. In general all of these will be required for the acceleration and control of the particle beam. Furthermore, energy deposition and the subsequent radioactivation of the accelerator components from the beam requires detailed component geometrical models combined with particle physics simulations.

The accurate simulation of particles, their motion, and their interactions with matter is a challenge which often necessitates vast computing power and a careful balance between physical accuracy and computation time. Due to the level of complexity that must be managed, different scenarios and applications to be studied are dealt with using different, specialised, simulation software.

The scenarios treated in this thesis involve the simulation and transport of energetic particles in materials, which may be in electromagnetic fields; any subsequent secondary particles resulting from the decay or interaction with the material; and any secondary particles resulting from these interactions. The other simulation scenarios pertain to the accurate tracking of particles through particle accelerators.

In this section the simulation tools used at various points to acquire many of the results in this thesis are presented and discussed, and then the novel energy deposition and accelerator tracking code, BDSIM, is introduced. Finally, a Python package for

translating FLUKA geometries is described.

3.1 Particle tracking

MAD-X (Methodical Accelerator Design X) is the standard tool used at CERN for simulating beam dynamics and designing and optimising accelerator lattices [34]. With the interface to an implementation of Polymorphic Tracking Code (PTC) library, MAD-X/PTC, it is also capable of symplectic particle tracking and the array of other features offered by PTC including thick lens tracking [35, 36].

MAD-X is generally the starting point for other simulation codes at CERN, as it is an accessible format that the Large Hadron Collider (LHC), its magnetic elements, their strengths, and their apertures, are most completely described in. One will typically take a MAD-X job describing a given lattice and then use it to write this lattice to the Table File System (TFS) format in which all elements with their lengths, electromagnetic parameters, tilts and offsets, apertures, as well as the optical parameters at various points, and others, are written. The file additionally includes the Twiss parameters, dispersions, beam centroids and angles, and other optical parameters. In this regard the Twiss TFS file contains the ideal, full optical description of many lattices in accelerator physics.

SixTrack [37] is one such code which starts from a MAD-X description and its primary use is fast particle tracking for dynamic aperture studies in machines like the LHC, High Luminosity Large Hadron Collider (HL-LHC), Future Circular Collider (FCC) [38], and Relativistic Heavy Ion Collider (RHIC) [39]. It is a fully-symplectic 6D thin-lens tracking code capable of tracking many thousands of turns with the full treatment of nonlinear magnetic elements, machine imperfections, misalignments and more. Although its primary use was originally for dynamic aperture studies, in the years before the startup of the LHC, SixTrack was extended for use in collimation studies [40]. SixTrack is now the standard tool used at CERN for LHC collimation studies. Any other code used to study the LHC collimation system should therefore be compared with SixTrack. In addition to being used to design the LHC collimation system, as well as its ongoing optimisation, it has also been used to great effect to aid the design of collimation systems for future colliders, including FCC [41], HL-LHC [42],

and High Energy Large Hadron Collider (HE-LHC) [43].
Extending SixTrack to include collimation meant the accurate and symplectic tracking of the main code could be used in conjunction with collimation-specific routines of K2/COLLTRACK [44, 45]. This extension works by using the main SixTrack tracking routines, with the special treatment of apertures and collimators. If a particle reaches a collimator the tracker hands over to a set of particle physics routines capable of propagating the particle through the collimator, accounting correctly for any interactions with the collimator material, before then passing the particle back to the main tracking routines. If the particle undergoes an inelastic collision within the collimator, it is recorded as lost within that collimator, and that particle is no longer tracked. If, on the other hand, the particle makes it through the collimator jaws intact then it is passed back to the main tracking routines and continues on its way through the machine lattice. The aperture model, on the other hand, supplements the main tracking routines as an additional check. At each tracking step, the particle is checked against the local aperture and definition, and if its transverse position lays outside the aperture, then it is backtracked with an inverse drift and routine to the point of intersection and is recorded as lost at that point. Once the particle is recorded as lost, it is no longer tracked. It is key to this thesis to note that SixTrack does not treat the physics of secondaries at all.

3.2 Energy deposition

There exist a number of tools for the study of energy deposition and machine-induced backgrounds in particle accelerators. At CERN the code of choice is typically FLUKA, which is a general purpose Monte Carlo particle simulation code for propagating particles through matter. For the LHC it is used for shielding, activation, radiation protection, detector response studies, machine-induced background studies and others [46, 47]. For these applications at the LHC, FLUKA is effectively the benchmark to which other codes should be compared because it is the standard code and also because the FLUKA beamline component and shielding geometries are highly detailed, having been built up and improved over decades with contributions from a large number of people. The group at CERN have developed the FLUKA Element Database

(FEDB) that consists of geometries that have been iteratively improved upon over the many years that such simulations have been performed [48]. The FEDB is supplemented with the FLUKA LineBuilder that draws from the database to procedurally build the complex beamline models of various regions of interest in the LHC which can result in very large input files.

In an effort to combine the powerful accelerator tracking routines of SixTrack with the general purpose Monte Carlo simulation and elaborate component geometries of the FEDB, the SixTrack-FLUKA coupling was developed [49]. In this setup, a subsection of the whole ring, perhaps one of the insertion regions, is built in FLUKA, and the whole LHC ring is built in SixTrack. Once the simulation begins, when a proton reaches a certain point in SixTrack, the proton is passed over to the FLUKA model, and is then tracked through it and may then undergo particle-matter interactions that FLUKA provides, including the tracking of any secondaries within the FLUKA model. A sufficiently on-momentum proton that reaches some predefined plane where the FLUKA model stops is then passed back to SixTrack for tracking through the rest of the ring. This approach enables the study of the impact of multi-turn effects on energy deposition and activation in regions of interest.

Geant4 is a C++ simulation toolkit used for simulating the traversal of particles through matter [50]. It supports an extensive set of physics processes, ranging from eV all the way up to the several TeV scale. The library is designed to allow simulations to be implemented in the most bespoke fashion possible, giving the user total control over the physics processes used, the geometry, the specific cuts to optimise the running time, the output written to file and more. Added to this, it is fully extensible, meaning that one can extend the physics processes and other features of Geant4 to an arbitrary degree, if for example a certain physics process is missing, one may add it. Geant4 is optimised for speed, meaning that with careful consideration of idiomatic C++ and Geant4, the resulting simulation will be both physically accurate and very fast. However, the user must write compiled C++ to describe their simulation, which in itself can be error-prone and labour intensive. Due to the aforementioned benefits, it has been applied successfully and widely to a range of scenarios, including medical, particle accelerator, and high energy particle physics applications, among others [51–53].

3.3 Machine-induced backgrounds

Particle physics simulations for the study of machine-induced backgrounds in the ATLAS detector of the LHC are of particular relevance to this thesis. A number of general purpose tools have been used to this end, the previously mentioned FLUKA has been used study to machine-induced background at the LHC in ATLAS and CMS [16], as well as LHCb [54]. MARS [55] is another tool which has been applied to machine-induced backgrounds in the LHC, showing good agreement when compared with FLUKA for backgrounds in ATLAS and CMS [16], as well as for the low-luminosity insertion regions [56], and CMS [57]. Studies in other experiments have used completely different tools altogether, for example at Belle II [58], where SAD [59] and TURTLE [60] have been used in combination with Geant4 [61]. Additionally, SixTrack has been used to simulate beam-gas scattering around the ring and its influence on machine-induced backgrounds [62].

These simulations will generally require a detailed model of the region upstream of the detector in which background particles can be tracked from their origin to the detector including its response. Each detector has its own highly specialised MC software to simulate its response, therefore a plane is chosen as an interface where particles can be recorded in one simulation and handed to another specialised detector simulation. The background source may be generated directly in the region upstream of the detector, for example with an event generator such as DPMJET-III, or an additional simulation may be used. For example, using SixTrack to generate hits on the tertiary collimators [15].

3.4 Beam Delivery Simulation

Beam Delivery Simulation (BDSIM) is a novel simulation code which features a wide array of accelerator tracking routines and has applications to both energy deposition and machine-induced backgrounds studies. BDSIM is built on top of a collection of widely used libraries, including CLHEP [63], ROOT [64] and Geant4 [50]. BDSIM automatically builds a Geant4 particle accelerator from generic, predefined accelerator components from a minimal, human-readable language. This is in contrast to FLUKA, which requires a full description of the geometry, and Geant4, where C++ must be

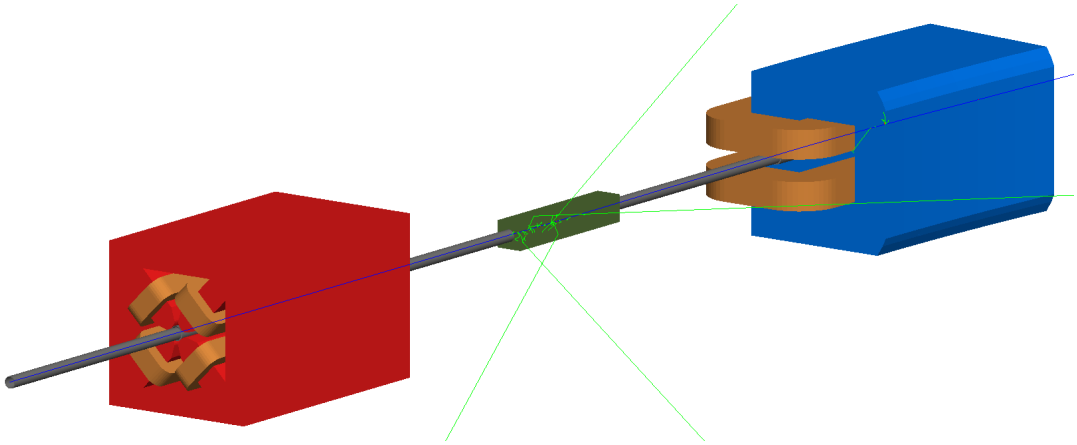
written and compiled before any simulation can be run. Both of these can require highly-specialist knowledge, and can be error prone and laborious, requiring 5-10 people years of work to created a validated model.

BDSIM's input file format, GMAD, is very similar to the widely-used MAD-X and MAD-8 formats, but in contrast to these two languages, it is implemented in terms of a grammar and parser generator which ultimately means the parsing is far more robust, and strongly eliminates errors permitting rapid correction and optimisation of a model. A simple, minimal GMAD example with resulting output in Geant4's QT visualiser, is shown in Figure 3.1. Whilst this example is very basic, describing more complicated machines is simply a matter of defining more components and adding them to the line sequentially.

Whilst BDSIM offers the means to define accelerators from scratch, in general a given accelerator will already have been defined in another code, such as MAD-X [34], MAD-8 [65], or Transport [66]. For this reason, BDSIM is supported with a utility, `pybdsim`, which is a Python package that can be used to quickly convert lattices between description formats. This allows BDSIM to easily simulate accelerators designed in other codes. This approach is particularly useful as it means that BDSIM can leverage the extensive set of existing optical descriptions in other languages. By just providing a few different converters most accelerators can be simulated in BDSIM. Converting from MAD-X to GMAD, as shown in Figure 3.2, involves a one-to-one mapping of elements between formats. However, MAD-X uses effective lengths for its magnets and a hard edge model, rather than the physical length, and so BDSIM's magnet geometries will tend to be shorter than they are in reality, but in the case of energy accelerators this difference can be considered negligible. It is useful to compare the optical functions of the original lattice with the converted GMAD one to validate the preparation of the model. To aid in the production of optical comparisons BDSIM provides a utility called `rebdsimOptics`. The conversion shown in Figure 3.2 is a very simple use of the conversion facilities, and results in a very simple description in GMAD. The resulting model features only the correct optical description of the lattice, with many features falling back on default values when loaded in BDSIM. Additional features such as apertures, outer magnet geometries, collimator openings, field maps

```
d1: drift, l=1*m, aper1=2.*cm;w
q1: quadrupole, l=1*m, k1=0.0001;
c1: rcol, l=0.6*m, xsize=5*mm, ysize=0*mm, material="Cu",
    outerDiameter=10*cm;
s1: sbend, l=1*m, angle=0.10;

l1: line = (d1, q1, d1, c1, d1, s1);

use, period=l1;

option, physicsList="em";

beam, particle="proton",
      energy=10.0*GeV;
```

(a) The GMAD input for a simple machine consisting of a quadrupole, a collimator and a sector bend, with 1 m drifts in between each element.

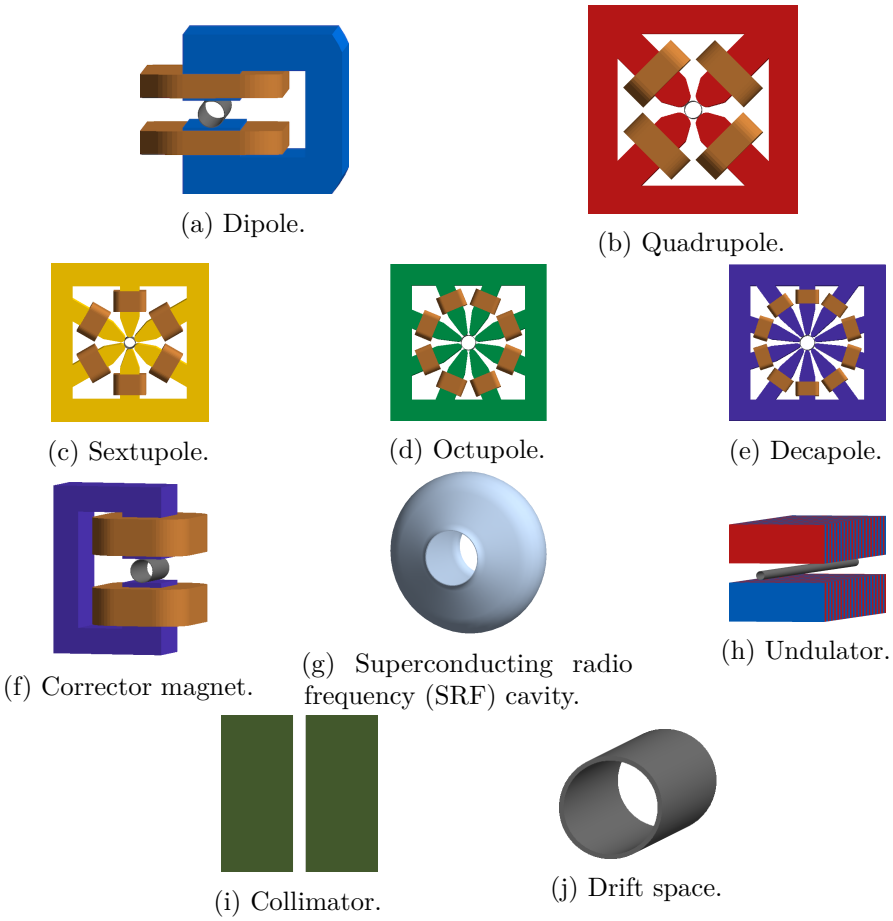


(b) The BDSIM rendering of the accelerator model as described in the GMAD shown above in a. A single proton has been fired into the machine (moving from left to right denoted with the blue line), where a number of interactions and secondary particles can be seen as it passes through the closed collimator. The trajectories are colour coded according to the particle's charge: blue is positive, green is neutral, and red is negative.

Figure 3.1: A simple GMAD lattice, (top) with the corresponding visualisation in BDSIM (bottom), with the propagation of a single 10 GeV proton shown.

```
import pybdsim
pybdsim.Convert.MadxTfs2Gmad("madx-lattice.tfs", "bdsim-lattice")
```

Figure 3.2: Minimal conversion from a MAD-X lattice (`"madx-lattice.tfs"`) to GMAD (`"bdsim-lattice.gmad"`) using the Python utility `pybdsim`. Additional information, such as apertures and collimator openings, may be supplied as optional key word arguments to `MadxTfsGmad`.



(a) Dipole. (b) Quadrupole. (c) Sextupole. (d) Octupole. (e) Decapole. (f) Corrector magnet. (g) Superconducting radio frequency (SRF) cavity. (h) Undulator. (i) Collimator. (j) Drift space.

Figure 3.3: A selection of predefined accelerator components in BDSIM.

and external shielding geometries can either be folded into the main `MadxTfs2Gmad` call or appended afterwards to the GMAD files.

Preparing geometry in Geant4 can be a laborious process due to the need to ensure the absence of overlaps between each piece of geometry. For this reason BDSIM provides a number of predefined Geant4 parameterised accelerator geometries that scale robustly in size and length without the risk of overlaps. A subset of these generic accelerator component geometries are shown in Figure 3.3.

BDSIM can also load geometry from external sources in three different formats, Mokka [67], ggmad and Geometry Description Markup Language (GDML) [68]. Whilst the other formats are supported, GDML is the recommended geometry format for use in BDSIM. With the use of the supporting python package `pyg4ometry`, a wide-range of other formats can be converted to GDML, as detailed further in section 3.5.

BDSIM merges familiar accelerator tracking routines, with the particle-matter interactions of Geant4. This is mostly achieved by combining the generic components

with integrators that map the beam particles from the beginning of the element to the end. As was presented in Chapter 2, these maps are expressed in the Frenet-Serret coordinate system. This is in contrast to the Geant4 coordinate system that is expressed purely in 3D Cartesian coordinates. A set of coordinate transforms between Cartesian and Frenet-Serret are used throughout BDSIM so that Frenet-Serret coordinates can be used with Geant4. With these transforms, the accelerator tracking routines can be used to step through many of the accelerator components for which closed-form solutions exist. For other components, BDSIM can use field maps combined with numerical integrators to step through the element. Furthermore, the accelerator maps for stepping directly through components are typically only valid in the paraxial approximation. In the non-paraxial case, the particle is stepped through with an RK4 integrator.

Geant4's set of physics processes is extremely broad, supporting energies up to 100 TeV [69] and down to below 1 keV [70]. This extremely broad energy range opens up a similarly broad range of applications for BDSIM. For a given application, a subset of all the possible physics processes or energy ranges are likely to be of interest. In high energy physics applications, one is unlikely to be concerned with physics on the eV-scale. Additionally, it may be desirable to disable certain physics processes of little interest that may otherwise dominate computation time, such as synchrotron radiation. Another issue is the question of when to stop an event. If a 6.5 TeV proton and all its secondaries were to be tracked all the way down to the eV scale, the total computation time could become prohibitively long.

The set of physics processes used in a Geant4 simulation is fully modular, allowing a bespoke subset of the physics processes to be selected for each application. Due to the difficulties in selecting the best set of physics processes for a given application, (referred to as the *physics list*), Geant4 provides a set of predefined *reference* physics lists that are routinely validated with specific use-cases in mind.

Also, as one is not necessarily interested in the full energy range, it is desirable to halt the propagation of a particle below a certain threshold. Geant4 provides two features for this purpose: kinetic energy cuts and range cuts. Kinetic energy cuts are fairly self-explanatory: when a particle's kinetic energy falls below a user-defined threshold, it

is killed and all remaining energy is deposited at that point. Range cuts are distances based on the expected path length of a particle within a given material. When deciding whether or not to create a secondary particle, the distance such a particle would travel were it to be created in the given material is queried. If it is below the range cut, then the secondary particle is not created and the energy of the proposed secondary is instead deposited at that point.

Range and kinetic energy cuts both provide means to fine tune the balance between computation time and accurate results. Variance reduction techniques offer a further means to control computation time. One such technique in particle physics is cross section biasing. Geant4 also supports cross section biasing [71], where the cross section is scaled up (or down) for a given, rare (or overly common) process, and then the weights of any products are scaled proportional to retain physical rates. This is demonstrated in Chapter 5 to force the relatively rare beam-gas interactions.

BDSIM can simulate circular machines and must therefore have a means by which it can count turns and terminate simulations beyond some desired number of turns. This is achieved with the use of a *terminator* that is placed at the end of a circular machine in BDSIM. Every primary has a turn counter associated with it that is incremented each time it passes through the terminator and is checked against the turn limit. When the turn counter for a particle within the terminator exceeds the turn limit, it is killed. Without the terminator, protons would simply orbit around the ring forever (since one often does not simulate synchrotron radiation so as to reduce simulation time), or until they were eventually lost on a limiting aperture. In the LHC for example, collimation simulations are typically limited to only 200 turns, meaning that the terminator is an important feature that enables such studies in BDSIM.

One of the features that distinguishes BDSIM is the full treatment of all secondaries and the recording of resulting energy deposition in the accelerator. A single 6.5 TeV proton can deposit energy before being lost to an inelastic process. However, the vast majority of this energy is carried by secondaries resulting from the inelastic collision. These secondaries may deposit the energy over a large distance, far from the initial proton loss point. For each event a large range of data are stored allowing for one to perform extremely detailed studies. This large amount of data results in large output

file sizes. For a typical LHC collimation study that will involve at least 6×10^6 protons, as losses in the TCT will often be on the order of 1 in 10^6. The total output file size for such a simulation will be on the order of several terabytes of ROOT output. Among BDSIM's supporting utilities there are the analysis tools *rebdsim* and *rebdsimCombine*, with which complex analyses can be defined and performed efficiently. These are important to make the analysis of such large quantities of data tractable. Unlike the simple integrals of lost particles provided by SixTrack, BDSIM has event-level structured data, permitting per-event statistical uncertainties to be calculated. This is done automatically by rebdsim and rebdsimCombine. However, more bespoke analyses may require Python or C++ to be written to fully leverage the possibilities offered by BDSIM, and BDSIM provides a library of ROOT classes to this end.
To conclude, BDSIM is a mature and well-developed Monte Carlo simulation suite that bridges the divide between accelerator and particle physics and is useful for a range of different applications. In this thesis it is used for the study of machine-induced backgrounds and collimation studies in the LHC.

3.5 Accelerator geometry

To properly calculate the fluences, activation, and heat loads in accelerator simulations, an accurate geometric model of the particle accelerator, shielding and surrounding regions is required. These details maybe necessary as particles propagating through a given model, and the interactions they undergo, will depend on the materials and shapes of the constituent regions they travel through. Asymmetries and details that may be omitted in cruder models can correspond to hot spots of energy deposition, or lead to characteristic fluence shapes. A model lacking the required level of detail may see incorrect rates and distributions altogether. In this section, geometry in the context of radiation transport simulations is introduced and discussed, and a library for building geometries, both from scratch and by converting from particle physics simulation codes is introduced.
There are a number of different ways of describing geometry in formats understandable by computers, however in radiation transport simulations the most common approach is the use of Constructive solid geometry (CSG). With CSG, complex geometries can be

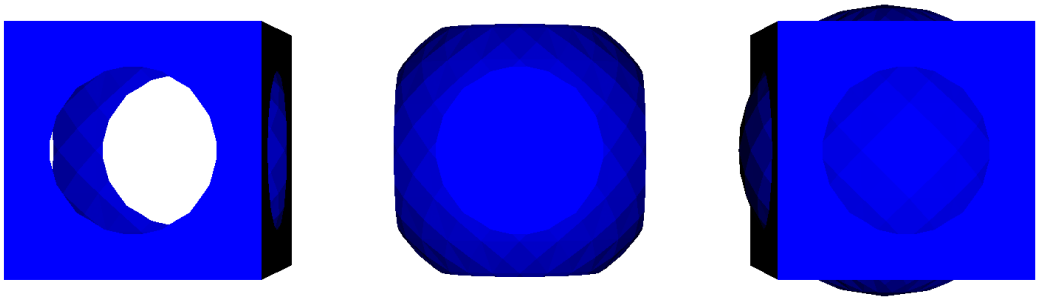
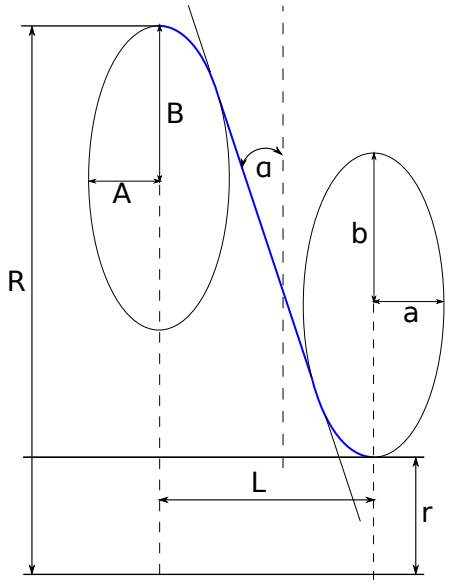


Figure 3.4: Example of Boolean constructive solid geometry renderings. The same box and sphere are used in all three cases, changing only the Boolean operation from subtraction (left), to intersection (middle) and then union (right).

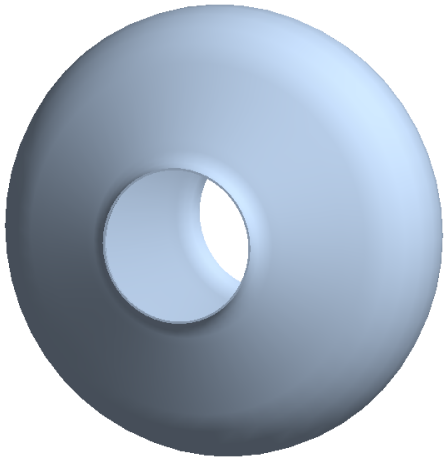
built up from simple primitives (spheres, boxes, cylinders, etc.) with the use of Boolean operators, which correspond to three common set operations: intersection, complement and union. The intersection of two solids consists of the regions of space shared by both of the operands, i.e. for two solids A and B, the intersection $A \cap B$ is defined as the set of all points common to both A and B, $A \cap B = \{\vec{v} : \vec{v} \in A \wedge \vec{v} \in B\}$. Similarly, the complement $A \setminus B$ is defined for two solids A and B, is defined as the set of points that are in A but not B: $A \setminus B = \{\vec{v} : \vec{v} \in A \wedge \vec{v} \notin B\}$. Finally, the union of two solids A and B is the set of points that are in either A or B: $A \cup B = \{\vec{v} : \vec{v} \in A \vee \vec{v} \in B\}$. Examples renderings of these combinations are shown in Figure 3.4.

It is important in tracking codes to be able to tell where a given particle is located within a model. This is the main reason why CSG is typically used in simulation codes, as the primitives can be defined in terms of simple equations. This means that the question of whether or not a particle lies inside or outside a given solid can be reduced to checking an inequality (e.g. the equation of a sphere). These primitives can then be combined with the aforementioned Boolean operations and determining whether a particle is contained within that Boolean reduces to evaluating a series of inequalities combined with logical NOTs (complement), ANDs (intersection) and ORs (union), which can be performed efficiently by computers. This optimisation is necessary as one single primary particle may result in millions of particles to be tracked through a given piece of geometry.

Another widely-used method for modelling solids is the *boundary representation*



(a) Parameterised boundary representation of the elliptical cavity as used in BDSIM.



(b) BDSIM elliptical cavity.

Figure 3.5: A boundary representation of a superconducting radiofrequency cavity and the resulting 3D solid. The RF cavity is defined in terms of its edges, a reflection about the larger radial axis, and 2π volume of revolution along the central longitudinal axis.

(B-rep). This describes solids in terms of faces, vertices, and edges, and has a wider set of operations (extrusion, chamfer, etc.) than CSG, and is not widely used in particle physics codes due to the difficulty in reducing them to simple and efficient equations. However a small subset of B-rep is implemented in Geant4 in terms of extrusion solids and solids of rotation, as these are very powerful and expressive ways to describe solids. An example solid of rotation as used in BDSIM for super-conducting RF cavities is shown in Figure 3.5.

Finally, the other solid modelling representation used in radiation transport is the mesh. A solid described in terms of an arbitrary number of facets is a meshed solid. This method allows for a solid with an arbitrary amount of detail, and particularly complex or continuously varying shapes (e.g. human phantom) can be represented using a solid where no simple combination of CSG primitives would achieve the same shape. A mesh can also often be exported from computer aided design (CAD) programs and therefore offers a means by which reproducing highly detailed components in CSG from scratch can be avoided. The disadvantage of this method is that for determining whether a point is inside or outside of the geometry, there must be a number of checks equal to

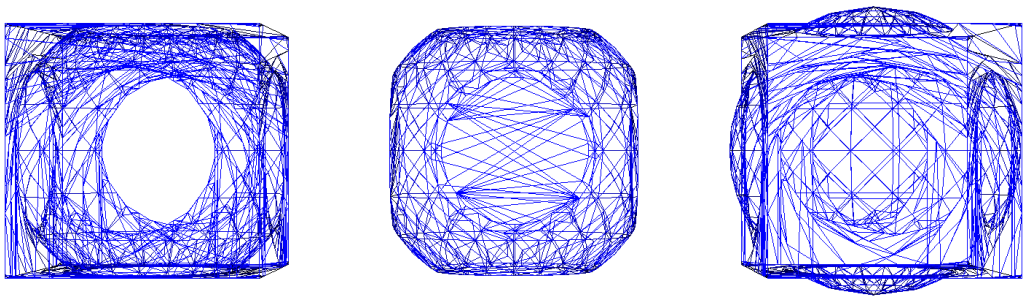


Figure 3.6: Example of Boolean solids with the underlying meshed representation. These are the same solids as shown in Figure 3.4.

the number of facets. This can be prohibitively computationally expensive for the most detailed geometries, although there exist a wide number of mesh simplification algorithms that can be used to reduce the number of facets whilst preserving detail [72]. Figure 3.6 shows the mesh representations of three Boolean solids.

3.6 Geometry conversion

BDSIM is built on top of Geant4, which uses CSG and a subset of boundary representation to describe geometry, and features a geometry loader for parsing externally provided geometry files for use in particle physics simulations. The native import and export geometry format for Geant4 is GDML [68]. The GDML format is capable of describing the full set of solids and other features provided in compiled Geant4, as well as meshed solids.

Another geometry format is that used in FLUKA. Geant4 and FLUKA are comparable in their approaches, both use CSG, however they are different in a number of fundamental ways. Geant4 makes exclusive use of finite solids, whereas FLUKA has access to a range of primitives that are infinite in size (e.g. half-spaces and infinite-length cylinders), but these must ultimately be reduced to finite volumes by making use of CSG Boolean operations. Geant4 has a wider range of fundamental primitives and with the use of *logical* and *physical* volumes, repetition is minimised and the memory footprint of the simulation can be reduced. These features stem from its original purpose for high energy physics detector simulation. FLUKA, in contrast, is able to reuse geometries to minimise repetition, but this capability (using the

LATTICE card) is not fundamental to its operation. FLUKA also has a useful and user-friendly graphical user interface (GUI), *flair*, which is a CAD-like program for designing CSG in FLUKA simulations [73]. This is in contrast to Geant4 which requires the geometry to be stated plainly, and rarely clearly, in either C++ or GDML. It is often the case that there exists detailed geometric descriptions of accelerator or detector components that have been implemented in CAD or a simulation code that is not directly usable in BDSIM. Geometry preparation can be extremely time consuming and error prone. For example it is completely possible to define unphysical geometry in which the tracking in Geant4 will fail, for example by introducing overlapping geometries. For this reason a lot of time can be saved whilst reducing simulation errors by using these geometries from other software in BDSIM.

Due to the difficulties in preparing geometry for radiation transport simulations and the desirability of translating other geometry description formats for use in BDSIM, the geometry Python package `pyg4ometry` has been developed [74]. This library allows one to write Geant4 geometries purely in terms of Python code and supports the full range of geometry features Geant4. Also included are a number of utility functions that ease the composition of the models by including a robust visualiser, fast overlap checking, coplanar faces and disjoint union checking, and others. This Python interface to GDML makes the translation of other geometry formats into GDML for use in BDSIM readily achievable. Conversion from FLUKA to GDML is particularly useful as it opens up the wide range of FLUKA geometries for possible use in BDSIM, of which there are a large number for the LHC [48]. An additional benefit is that FLUKA's GUI flair becomes usable with Geant4 and BDSIM. One would be able to design an accelerator component completely using flair's powerful GUI and then immediately convert it to GDML for use in BDSIM. A number of components used for accelerator models in this thesis were designed in just this way.

In Chapter 5 results from the simulation of beam-induced backgrounds (BIB) in the region of ATLAS are presented. Producing these results required a model of IR1 in BDSIM which itself requires an accurate model of the tunnel. A detailed model of the tunnel already exists in the format used by FLUKA, so to save building this geometry by hand in GDML, a Python package for converting FLUKA to GDML called `pyfluka`

```
1  GEOBEGIN                                                    COMBNAME
2      0    0
3
4  RPP Box       -50. 50. -50. 50. -50. 50.
5  ZCC InfCyl1   0.0 0.0 25.0
6  YZP HalfSp1   38.0
7  ZCC InfCyl2   -50.0 50.0 13.416407864999
8  XYP HalfSp2   0.0
9  TRC Cone      0.0 48.0 0.0 0.0 34.0 0.0 10.0 0.0
10 SPH BLCKSPH   0.0 -2.0384 0.0 5000.0
11 END
12 REGION      5 +Box -InfCyl1 +HalfSp1 -(+InfCyl2 +HalfSp2)
13                | +Cone
14 BLCKREG     5 +BLCKSPH -Box -Cone
15 END
16 GEOEND
17 ASSIGNMA    SILICON   REGION
18 ASSIGNMA   BLCKHOLE  BLCKREG
19 STOP
```

Figure 3.7: Example FLUKA input file describing a `REGION` consisting of a number of solids combined with all the possible Boolean operations.

has been written. The implementation of this conversion process for a single complex geometry, the aforementioned LHC IR1 tunnel, is described in detail in the following section.[1]

3.6.1 Conversion of FLUKA Geometry to GDML

The first step in translating FLUKA to GDML must involve reading the FLUKA input file. Much of FLUKA involves declaration in fixed-width format and are therefore relatively simple to parse, but in general difficult to interpret. However, reading the CSG is not trivial, as its grammar is sufficiently complex that it requires a dedicated parser to extract the syntactic components of the statements. To this end, the parser generator ANTLR 4 was used [75]. A grammar in the ANTLR 4 syntax was written and a parser capable of reading FLUKA's syntax was generated. An example FLUKA input file is shown in Figure 3.7, and Figure 3.8 shows the abstract syntax tree (AST) resulting from parsing the first region (defined on lines 12 and 13 of Figure 3.7).
The next step involves translating this AST into a corresponding GDML description. A FLUKA *region* consists of the Boolean combination of multiple *bodies* (primitives).

[1] Courtesy of the CERN FLUKA team, with contributions from the CERN Radiation Protection team.

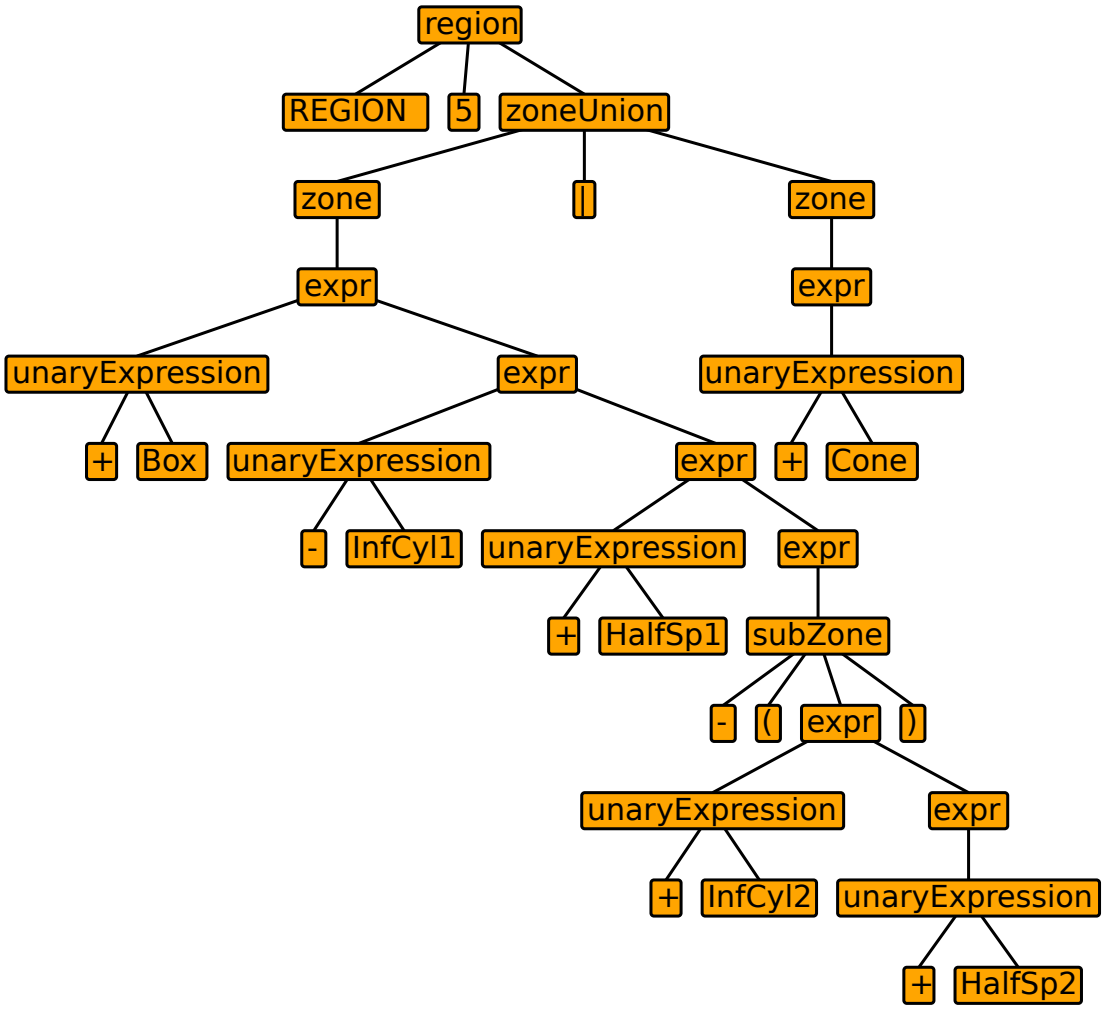
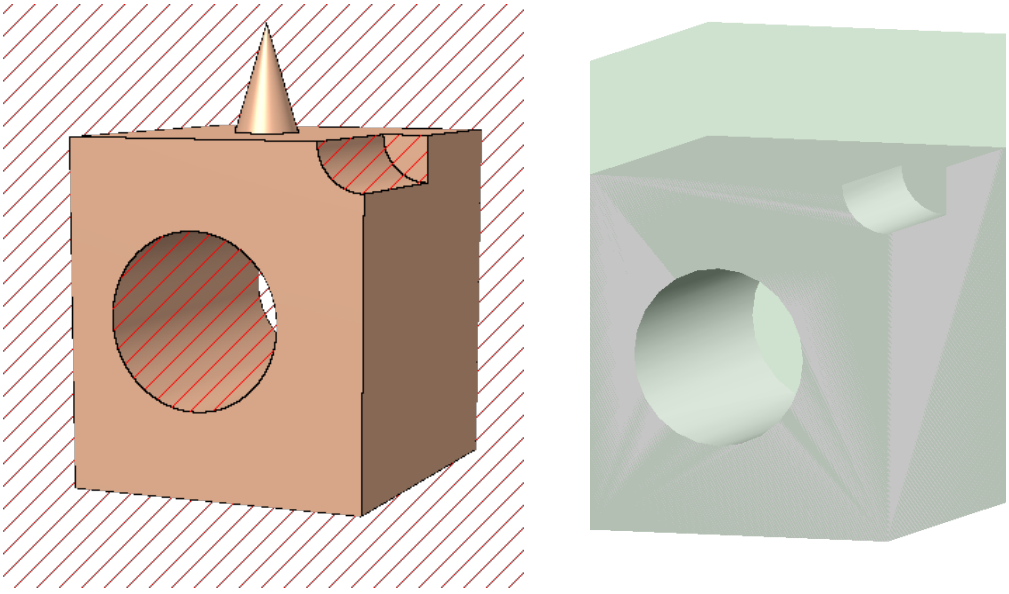


Figure 3.8: The abstract syntax tree for a FLUKA region showing the main syntactic features: intersection, subtraction, union and a nested Boolean operation. The plain text FLUKA is identical to line 12 and 13 of Figure 3.7.



(a) FLUKA geometry visualised in the flair geometry visualiser.

(b) Geometry viewed converted from FLUKA and viewed in BDSIM visualiser with the QT visualiser.

Figure 3.9: FLUKA geometry with converted GDML. The shapes can be seen to be identical, except for the missing cone on top of the GDML, which is simply a visual artefact from the deficient Geant4 Qt visualiser. The cone is in fact there, its presence can be inferred from the bounding box in green. This is the same geometry as shown in Figure 3.7 and Figure 3.8.

Each body is read from the input file and mapped to a Python class with methods for translating the body GDML solid with a rotation and position. The region is translated into a corresponding Boolean GDML solid whilst respecting the aforementioned operator precedence. The operator precedence in FLUKA is parentheses [`()`], then intersections [`+`], then subtractions [`-`] and finally unions [`|`]. An example conversion from GDML to FLUKA is show in Figure 3.9. The translation of this FLUKA region into the corresponding GDML description is not a simple one-to-one mapping of FLUKA bodies to GDML solids. Additional intermediate steps are required to reconcile the two fundamentally different approach to CSG. The key main considerations in the conversion process are as follows:

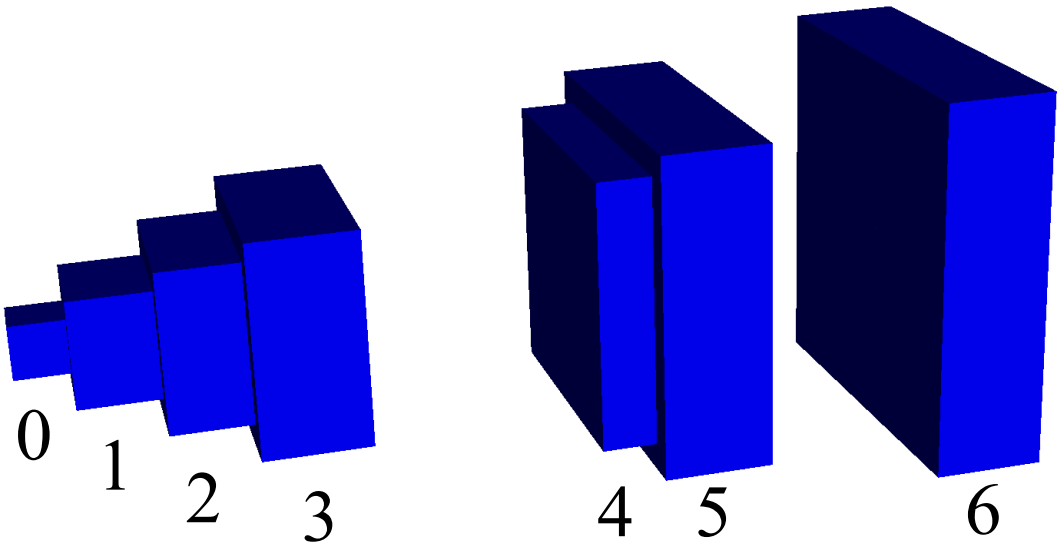
- FLUKA supports disjoint unions, that is, unions of solids which are not connected, whereas Geant4 does not. To guarantee correct tracking, these disjoint unions must be split up into their constituent parts.
- In FLUKA, faces of adjacent solids can be perfectly coplanar. In Geant4, there

must be a resolvable gap separating them. In BDSIM this gap is typically on the order of a nanometre, but if it is not present then errors will be introduced into the particle tracking.

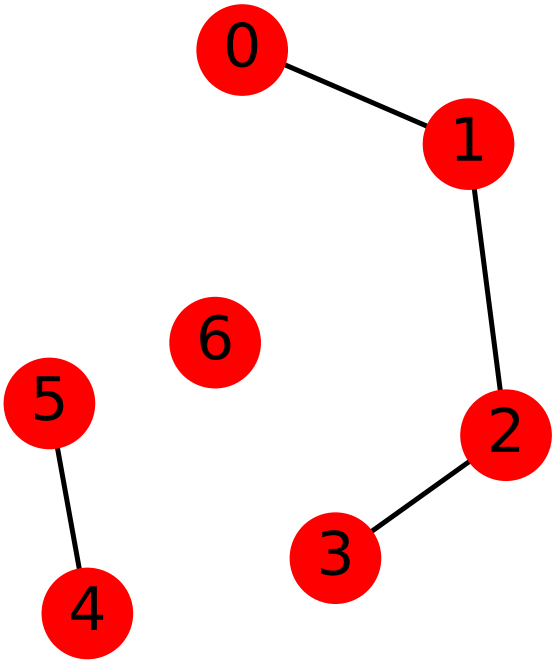
- FLUKA supports a range of infinite CSG primitives, whereas all CSG primitives in Geant4 are finite in extent. FLUKA's infinite primitives must be mapped to finite Geant4 ones in a way which preserves the final, finite Boolean.
- To speed up tracking and improve 3D rendering, redundant Boolean operations should be correctly omitted. A redundant Boolean operation is one which is equivalent to no operation at all. For example, intersect a solid with another which totally envelops it.
- Placing the geometry in BDSIM should be straight forward and making room for a beampipe should be simple, without the risk of overlaps or coplanar faces.

Firstly, disjoint solids can manifest themselves in two ways: the union of two completely unconnected solids, and a single solid cut in two with the appropriate subtraction. Neither of these two situations are obviously problematic, but as Geant4 states that these are improper constructions and can result in incorrect tracking, they must be treated (it may be the case that there are certain optimisations which Geant4 uses are only possible with these two constraints). FLUKA makes widespread use of the first case, that of unions of two completely unconnected solids. Therefore, for the correct translation to GDML, they must be handled correctly. This is achieved by meshing each individual union-component and then checking for mutual overlaps between them. The union components must be correctly sorted into sets that do overlap, and each set must then be joined and placed. A simple example union of 7 solids, only some of which are connected is shown in Figure 3.10a, with the graph resulting from the overlap checking between each union-component is in Figure 3.10b. The case where disjoint solids can arise from a subtraction (i.e. bisecting a union) has not been treated as of yet, however it seems to be considerably rarer than the aforementioned case.

Secondly, coplanar faces should be removed entirely when translating FLUKA to GDML. This is performed by adjusting the lengths, radii, etc., by the *length safety*—a



(a) A disjoint union consisting of four connected zones.



(b) The graph showing the connected regions of the zone.

Figure 3.10: a) A union of 7 boxes, numbered 0-6, only some of which are connected. In FLUKA this is a valid construct, but in Geant4 this is forbidden. b) The graph of connections between the various components of the union. Each node is a union-component, and each edge signifies a connection between two union-components. The generated graph of connections can then be used to split the union up into its connected parts, as mandated by Geant4.

small distance on the order of a nanometre in BDSIM that is sufficient to remove overlapping geometries and give correct tracking in Geant4. The important consideration here is that different Boolean solids should be adjusted differently: intersection and union operands should be shrunk slightly, and subtraction operands should be expanded slightly. The effect is that the resulting Boolean solid is smaller in all cases. The check on disjoint unions is performed after this step to correct for any resulting issues.

The next issue is that of the infinite solids, which are present in FLUKA but not in Geant4. Since all FLUKA infinite solids must ultimately be reduced to finite ones with the use of Boolean operations (consider for example truncating a cylinder of infinite length by subtracting from either end with an infinite half-space), there must also be a representation of that Boolean purely in terms of finite solids. This essentially involves translating infinite cylinders into finite cylinders, and half-spaces into finite boxes. Doing this in such a way that the result is identical requires a two stage process. First, the Boolean is evaluated with finite GDML components which are set to "effectively infinite", i.e. far bigger than they need to be. Then, this result is meshed, and the 3D extent, which consists of the three sides of a tight bounding box parallel with the axes of the Cartesian coordinate system, is calculated. The extent is then used to determine the minimum possible size of the constitute infinite solids and the parameters of the corresponding finite solids are set accordingly. This second stage may seem redundant, however this is required to robustly cover the many user-supplied possible input combinations. The solid minimisation algorithm is demonstrated in Figure 3.11, simplified to two dimensions.

Removing redundant Boolean operations involves checking to see whether or not the operation actually has any impact on the final geometry. For example, the intersection of a half-space with another solid which it completely envelops is redundant—it is completely equivalent to the enveloped solid, and the intersection with the half space can be omitted. This is achieved by comparing the extent before and after the operation, however the implementation of this feature remains in its early stages and it seems would offer only marginal improvements in tracking time and visualisation.

The complete geometry must have a single top-level volume that all geometry is

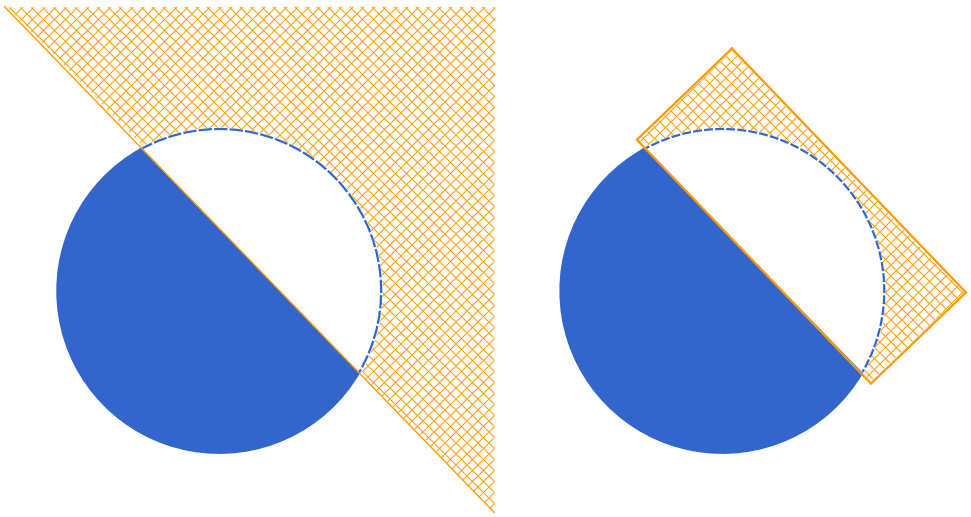


Figure 3.11: The subtraction of an infinite half space (orange) is replaced with the subtraction of a finite box, representing the same resultant geometry within the Geant4 geometry concepts.

contained inside. This is typically a simple box big enough to contain all the geometry. This is placed by BDSIM at the same level in the geometry hierarchy as the rest of the beamline. This presents a problem as the containing box will therefore overlap with BDSIM's beam line. To overcome this, the top-level volume can be made to be a Boolean solid with a hole cut through it in the inside of the tunnel. To this end an interface that allows for the placement of FLUKA geometry in terms of the original coordinate system and parallel to a component within BDSIM was written. Also, any hole required for a beamline that is otherwise purely a consideration of GDML, can be expressed completely in terms of the original FLUKA geometry. This makes a catalogue of highly detailed geometries readily available in Geant4 and BDSIM.
All of the above features and optimisations together result in a robust translator of FLUKA geometry to GDML that can be readily used in any Geant4 code. It should be apparent that these features are almost entirely dependent on advanced meshing algorithms featured in the underlying `pygdml`/`pyg4ometry`. The culmination of this work is shown in Figure 3.12, where the complicated and detailed IR1 tunnel geometry is shown both in FLUKA and in BDSIM, which is used in Chapter 5. The GDML geometry does still contain a number of overlaps that persist, therefore the offending subsections are removed to ensure correct tracking. However, by volume, these regions are on the order of 0.1 %, at large radii, and as such are not expected to have any

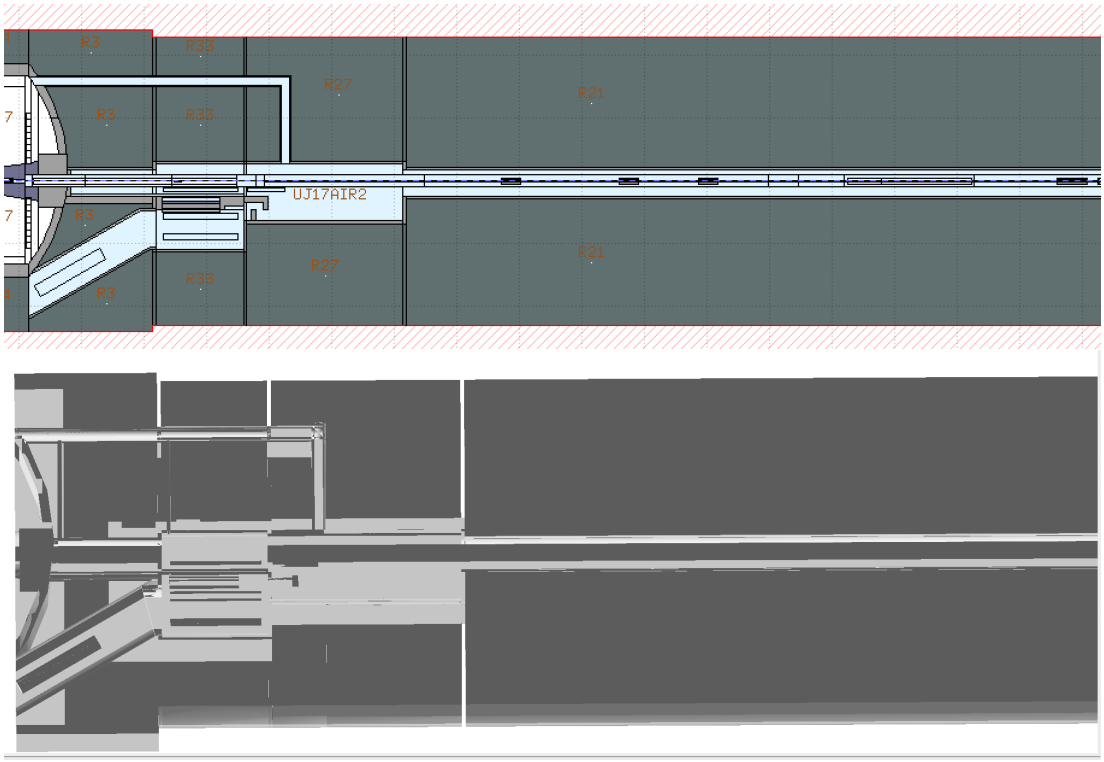


Figure 3.12: FLUKA IR1 C-side (incoming beam 2) tunnel model in flair (above) and converted and viewed in BDSIM (below). The characteristic features of the tunnel are visible in the BDSIM geometry. Any apparent differences are artefacts from the Qt visualiser, rather than differences in the geometry itself. The figures are not to scale.

sizeable impact on the results in BDSIM studies where this geometry is used.

3.7 Summary

In this chapter the range of Monte Carlo physics software used to generate results were described, and contrasted with BDSIM where appropriate. The importance of geometry to particle accelerator energy deposition studies was discussed, as well as the different ways that geometry can be described in a fashion usable in such simulations. The vast library of detailed FLUKA LHC geometries, the FEDB was described, and a novel Python package, `pyfluka` for translating FLUKA geometry to Geant4's GDML format was introduced. The techniques used in `pyfluka` to bridge the two fundamentally different geometry descriptions were outlined. The successful conversion of the highly detailed FLUKA IR1 tunnel to GMAD was shown as an example of the capabilities of the Python package. The algorithms used for the conversion are in the process of being improved to perfect the method. Its integration into the broader radiation transport geometry Python package, `pyg4ometry`, is ongoing, and its support for a broader base of the FLUKA features is being extended.

CHAPTER

FOUR

LHC COLLIMATION STUDIES

The Large Hadron Collider (LHC) at its design settings stores 362 MJ in each beam [12], which presents unique challenges in the operation of the LHC. The main issue is the need to protect the cold (1.9 K) superconducting magnets from heating from the beam, where as little as $1\,\mathrm{mJ\,cm^{-3}}$ [13], or 10^{-9} of the beam, can cause one of the superconducting magnets to quench, with the onset of damage to the machine occurring from as low as $1\,\mathrm{J\,cm^{-3}}$. Even without damage, quenching is a problem because the recovery process is lengthy and limits the time that physics data can be taken. These cold regions make up 90 % of the machine by length, so protecting the machine is a global problem requiring a comprehensive approach to intercept particles that would otherwise be lost in an uncontrolled fashion on some cold limiting aperture in the machine.

The primary means by which beam losses can occur is due to the growth of the beam halo. The beam halo consists of particles that lie outside of the beam core, either with large transverse or longitudinal amplitudes. Protons with large amplitudes may ultimately end up impacting on a limiting aperture. Similarly, off-momentum protons may leave their RF bucket and cease to undergo stable synchrotron motion, gradually losing energy due to synchrotron radiation, and impact somewhere in the machine [76]. The populating of the beam halo is a continuous process resulting from a number of effects, including intrabeam scattering, resonances, collisions at the interaction points (IPs), and others [77]. These effects can be reduced, but not eliminated.

For these reasons the LHC features a dedicated machine protection system [78, 79]. The two machine protection elements most relevant to this chapter are the LHC collimation and beam loss monitoring systems. The LHC collimation system is multi-staged and split over two insertion regions (IRs). Optimising the operation of the collimation system is important as the LHC parameters approach their nominal values, and in places, exceed them [80], making beam cleaning all the more important. The beam loss monitoring system consists of approximately 4000 beam loss monitors (BLMs) placed around the ring and is used to detect abnormal beam losses, and if they are detected above some threshold, a beam dump is initiated.

Simulating the LHC collimation system is important to understand its effectiveness, its limitations and its performance at all stages of operation including injection, acceleration, beam squeeze and during collisions, particularly in the drive toward higher beam intensities and smaller β^* in the IPs. Currently, this is typically achieved with SixTrack, the standard code used at CERN for such simulations and has been used to optimise the LHC collimation system as well as to aid in the design of future accelerators [41–43]. SixTrack, however, only tracks primary protons and only considers particle-matter interactions in the collimators. BDSIM, in contrast, will simulate both the proton, its interactions with matter anywhere in the accelerator (not just the collimators), and any secondary particles resulting from these interactions. To this end a BDSIM model of the LHC has been built for collimation studies. In this chapter the model is described in detail. BDSIM's use for LHC collimation studies is validated by comparing results to SixTrack. The BDSIM model was validated by assessing its tracking accuracy and then comparing simulated collimation performance with that of SixTrack. Next, the set of BLMs placed around the ring are simulated in the BDSIM model and the simulated dose is compared directly with the measured dose from a dedicated collimation qualification run in 2018. The simulated energy deposition in the LHC over multiple turns from the collimation system, a capability which is unique to BDSIM, is presented and discussed. Finally, the application of BDSIM to the LHC and the future work required to further optimise its use to this end is discussed.

4.1 The LHC collimation system

The LHC collimation system is situated in IRs 7 and 3. In IR7 is the betatron collimation system, designed to clean the beam of protons with large betatron amplitudes. In IR3 is the momentum cleaning collimation system, where an aperture bottleneck is introduced at a position of large dispersion, intercepting the primary off-momentum halo with momentum deviations greater than 10^{-3}.
The first point of contact for beam halo particles is designed to be in the primary collimators of IR7 (betatron halo) or IR3 (off-momentum halo). The diffusion of the beam halo is relatively slow and as a result the impact parameter on the collimators has been found to be 0.02 µm to 0.3 µm [81]. This means that a halo particle will not travel very far in the primary collimator before reentering the beampipe, and travelling downstream having perhaps experienced a transverse kick in the primary collimator. This is called the secondary halo. If the deflection from the primary collimator is sufficiently large, then the secondary halo may impact on one of the strategically placed secondary collimators (TCSGs) with a larger impact parameter than the primary collimator. Acquiring the sufficient angular kick to impact upon the secondary collimator may require several passages through a primary collimator (TCP) over many turns. Between the secondary collimators and the beginning of the arc are the tungsten absorbers (TCLAs), which are in place to shield the cold aperture downstream from showering particles produced in the primary and secondary collimators. Lastly, protons may leak from secondary collimators, resulting in the tertiary halo, which leaves the cleaning insertion altogether. To protect the limiting aperture in the inner triplet of the experimental insertions, tertiary collimators (TCTs) are placed around 150 m upstream of the IPs. They also serve an additional function in reducing backgrounds in the experiments. The collimation system is displayed in Figure 4.1.
The materials of the collimators play an important role in their operation. The primary and secondary collimators are made of carbon fibre composite (CFC), which is extremely robust as these collimators will see a large number of the high energy impacts directly from the beam. The choice of CFC is also merited due to the fact that the primary collimators may receive several bunches impacting directly on them in the event of an extraction kicker misfire. The TCLAs are made of tungsten, which, with its

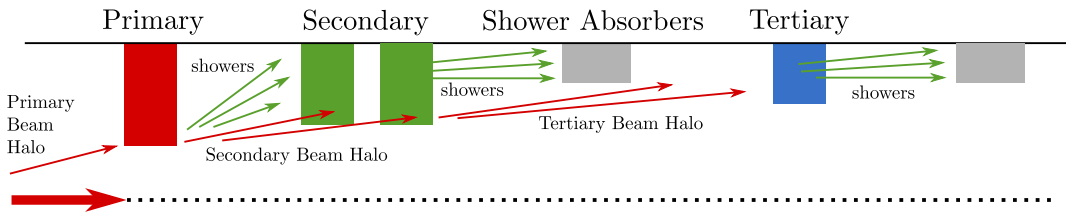


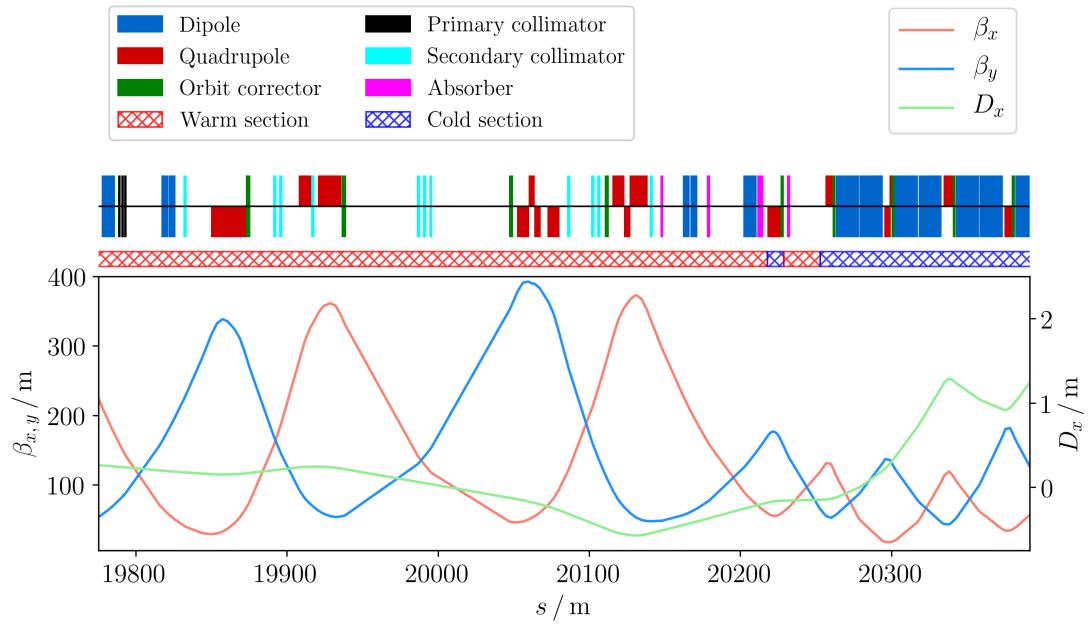
Figure 4.1: The main components of the multi-stage collimation system of the LHC.

combination of high Z and density, makes it very effective at absorbing particles. However, the TCLA are not designed to stop whole bunches.

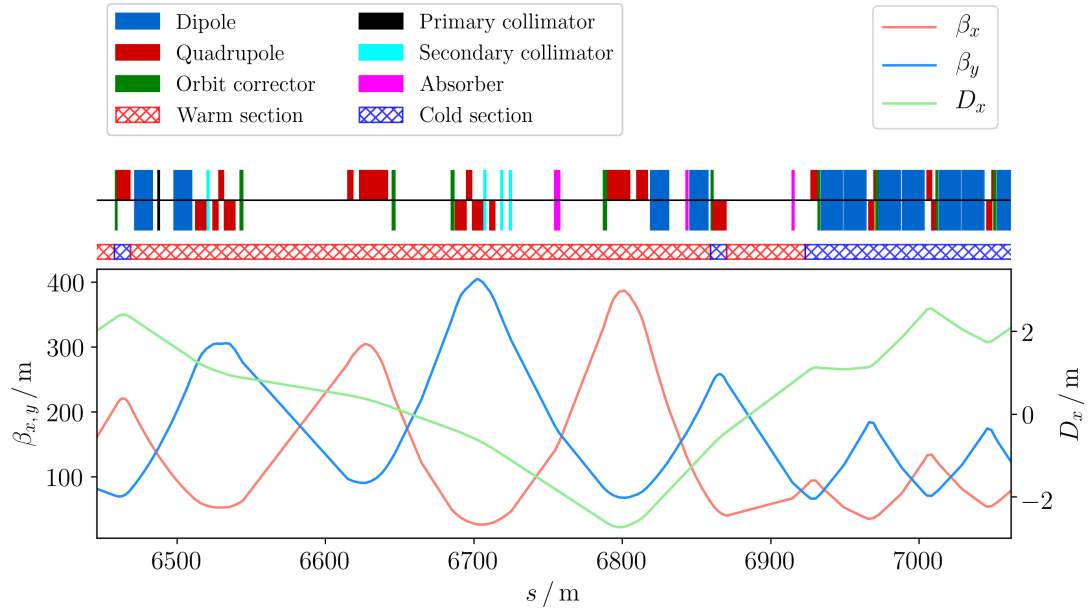
One benefit of this regime is that the secondary collimators can be placed relatively far from the beam in comparison to the primary collimators. This is important as the transverse impedance in the LHC is dominated by the placement of the primary collimators extremely close to the beam [82], so being able to keep the secondary and tertiary collimators and the absorbers at larger radii minimises this effect.

The layout of the two cleaning insertions is shown in more detail in Figure 4.2, and the positions of the collimators with respect to the betatron functions (β_x, β_y) and the dispersion (D_x) are shown. The primary collimator is placed in a position of high dispersion in IR3, and low in IR7. The betatron beam halo diffuses in every direction, which is why there are vertical and skew primary collimators in IR7. This is in contrast to IR3 where there is only one primary collimator, in the horizontal plane, because of the negligible vertical dispersion in the LHC. The red and blue hatched bands denote the warm and cold regions, respectively. The cold region immediately following IR7 is the dispersion suppressor, which is the beginning of the arc and is also entirely cold. The dispersion suppressor downstream of IR7 is very important to the functioning of the LHC system as during normal operation it is the region which is closest to quenching. For this reason the performance of the collimation system in IR7, and the resulting leakage to the dispersion suppressor (DS), puts an upper limit on the beam intensity. The DS immediately following the two IRs is vulnerable due to its proximity to the bulk of the upstream beam losses, the fact that it is superconducting (i.e. cold), and its high dispersion and small aperture.

In addition to the collimators associated with the main cleaning insertion, there are other collimators placed around the ring for more specialised purposes. For example,



(a) The betatron collimation system in IR7.



(b) The off-momentum collimation system in IR3.

Figure 4.2: The betatron and off-momentum collimation system in IRs 7 and 3, respectively. The beamline is displayed underneath the legends, above the red and blue hatching denotes the warm and cold regions. The variation of the horizontal betatron functions and the dispersion with the distance from IP1, β_x, β_y, D_x and s respectively are also shown.

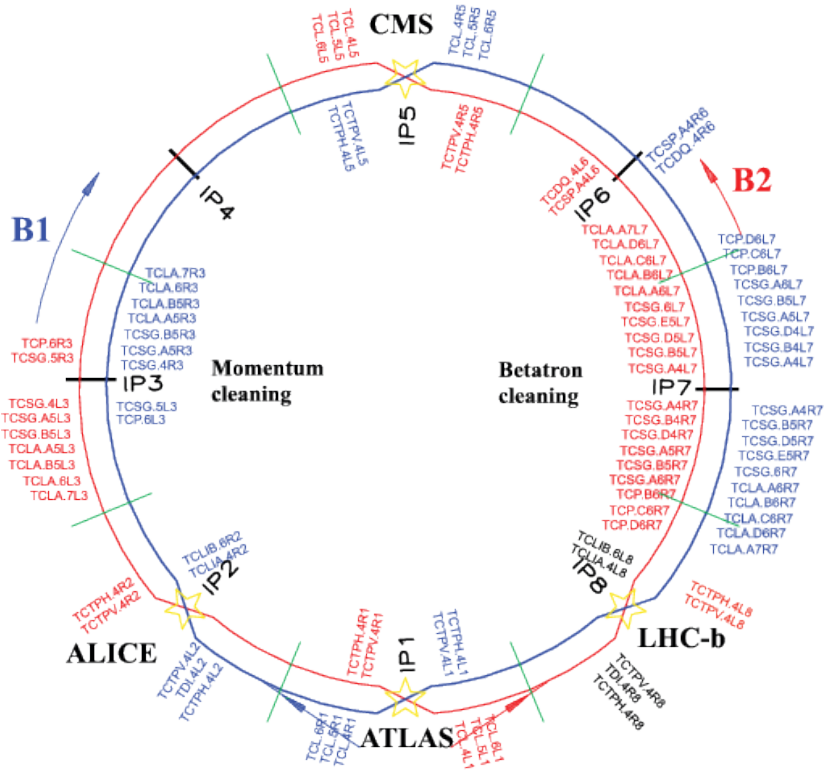


Figure 4.3: LHC insertions with Run II collimators and their locations. Reproduced from [84] (CC BY 4.0).

downstream of the experimental IPs are the physics debris absorbers (TCL) which are in place to protect the cold arc from physics debris originating from the experiments. The remaining collimators pertain to beam injection (IRs 2 and 8) and extraction (IR6). In the event of the failure of one of the injection kickers, collimators (TDI and TCLI) are placed downstream of injection in the LHC to protect the machine [83]. Finally, in the event of an asynchronous beam dump, the beam which would otherwise be sprayed downstream in IR6 onto superconducting components is intercepted with the TCDQ and TCSP collimators.[1] The complete set of collimators and their locations are shown in Figure 4.3.

The dump collimators also protect the downstream components against particles which are present in the abort gap and can cause quenches even during normal beam dumps [85, 86]. In a similar fashion, it is desirable to account for the possibility of faults during injection, again possibly due to a misfiring of one of the kickers.

[1]The filling pattern of the LHC features a 3 µs gap in the bunch train, which corresponds to the rise time of the extraction kickers. One possible error that can occur is a misfiring of the extraction kickers such that the beam dump occurs out of time with the abort gap. This is called an asynchronous beam dump, and can result in the beam being sprayed onto the aperture downstream of the kickers, if not for the specially positioned collimators.

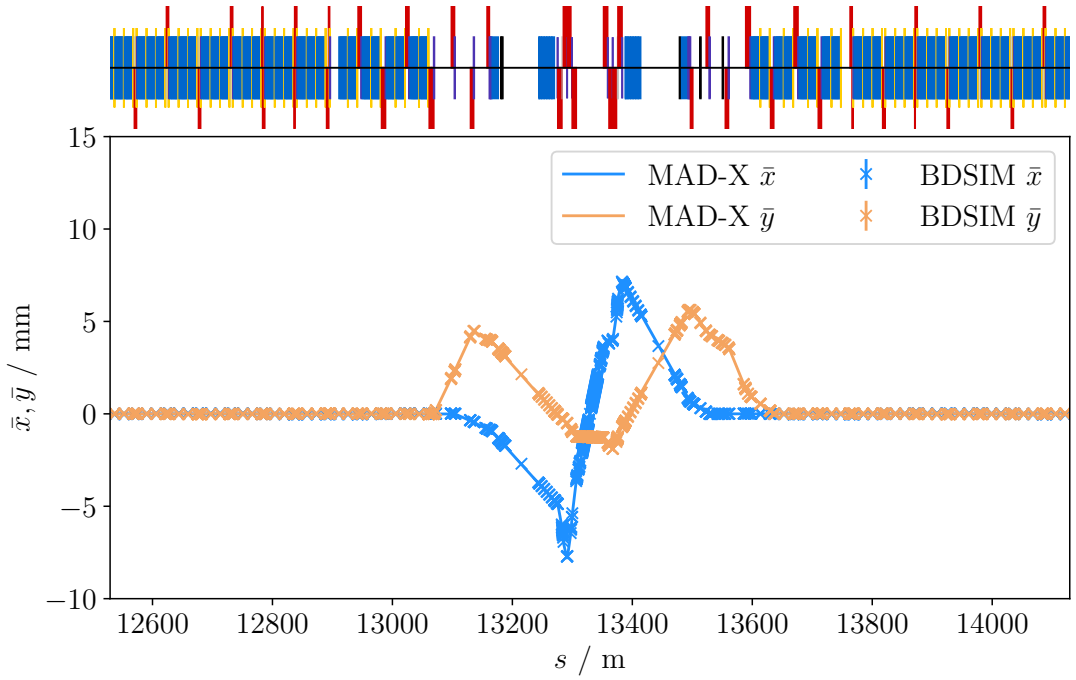
Whilst the main cleaning insertions removes the slowly growing halo from the beam, losses can also occur much more quickly. Examples include the injection and extraction failures mentioned above. In the event of such losses, the safest thing to do is initiate a beam dump. As mentioned previously, beam losses are detected with the use of around 4000 BLMs, which are designed to detect beam losses during normal operation, as well detect above-threshold beam losses. If they do exceed a threshold, a beam dump is triggered. This has proven vital as such fast losses have occurred numerous times [87, 88] during the LHC's operation, and are ultimately unavoidable.
The standard means by which the functioning of the collimation system can be evaluated is with the use of a loss map. A loss map, in the simplest terms, maps longitudinal positions along the beamline to some quantity which correlates with beam loss. For real data, this is typically the BLM integrated dose. For SixTrack, this is where a proton is considered "lost"—either it has undergone an inelastic collision in a collimator, or is outside the aperture definition. SixTrack does not at all treat the behaviour of the showers resulting from the lost beam proton. BDSIM can record losses in the way that SixTrack does, but also tracks the secondary showers of protons lost in the collimators, as well as continue to track protons when they have left the aperture.

4.2 The BDSIM model of the LHC

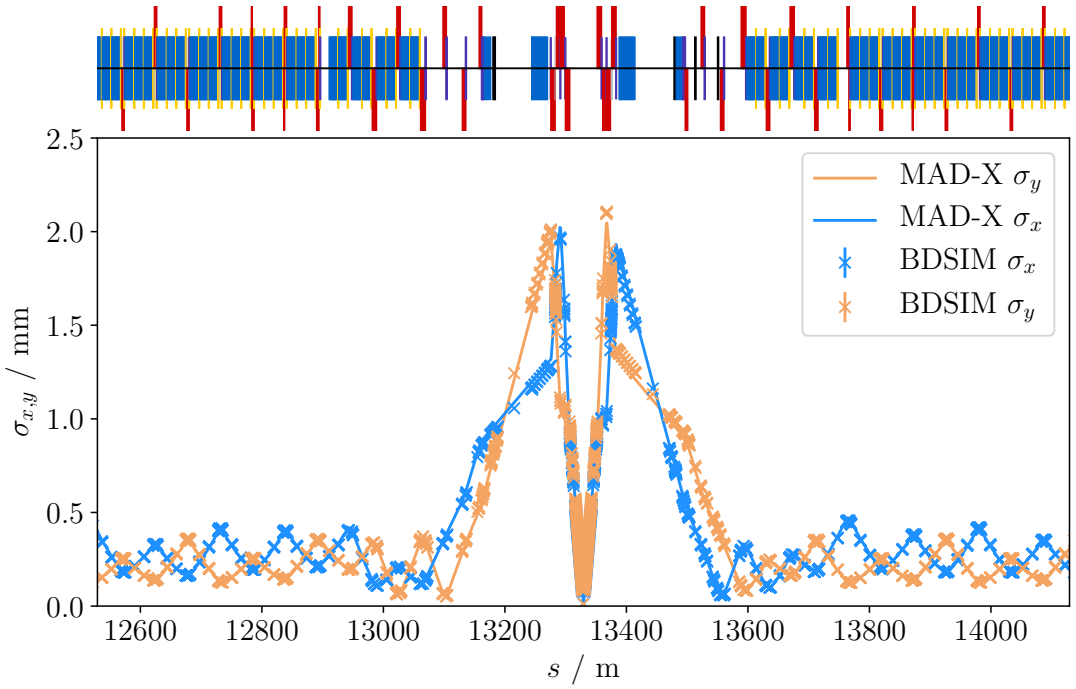
Building the BDSIM LHC model starts with an optical description from MAD-X. End-of-squeeze optics, i.e, the beam conditions at top energy but just before the two beams are brought into collisions, were sourced from the publicly available LHC Optics Web Home [89], and converted to GMAD using `pybdsim` [90]. These were chosen as they are the most adverse conditions in the LHC, and qualification loss maps have been performed in the LHC at this configuration, so comparisons with BLM data can be performed. Also, BLM data is not affected by cross talk from collisions, giving a clean image of the collimation hierarchy. Finally, another benefit is that set of standard externally prepared SixTrack input files for these optics are available, which is desirable as preparing SixTrack models by hand can be error-prone, and these input files are guaranteed to produce physical and meaningful results.
The MAD-X to BDSIM GMAD converter in `pybdsim` produces an optically accurate

model of the LHC in BDSIM. The pertinent optical parameters, the transverse beam sizes and centroids are shown in Figure 4.4, where excellent agreement can be seen. Whilst the correct optical description is extremely important, it is also necessary to get the correct model geometry for the resulting energy deposition in the simulation to be correct. Additional details can be folded into the MAD-X to GMAD conversion process in pybdsim. The most important geometric details are the collimator openings and aperture model, where the same collimator settings as used in SixTrack and the LHC loss map run were used in BDSIM. Secondly, the SixTrack aperture model was used in which the aperture is interpolated and sampled once every 10 cm along the beamline. This aperture model in SixTrack can be written to a text file, and an interface for mapping the aperture definitions to components was added to `pybdsim`, allowing the same aperture model to be used in BDSIM. In general, the aperture definitions will not line up in one-to-one correspondence with the component definitions, since a GMAD component definition is inseparable from its aperture definition. This means that the GMAD model has to be split into smaller components for each aperture point in the aperture model. This was done in such a way to guarantee the correct optical description, accounting for fringe fields and field strengths. The optical comparison of Figure 4.4 was generated with this model and it is clear that, for one turn at least, the agreement between MAD-X and BDSIM is unaffected. It is important to note that the aperture model was sourced from SixTrack version 4, but SixTrack version 5 [91] was used to generate the results presented throughout this chapter, and some subtle differences are present, as shown in Figure 4.5. However where the aperture is tightest, for example in the arcs, the apertures are in good agreement and this is the region it is most important to model. The agreement is worse near the collimators, but losses in this region will be dominated by collimator losses, not aperture losses, and so is less important.

Next, it is important to consider the geometries of the magnets themselves. As there are relatively few unique magnet designs used in the LHC, with most magnets belonging to one or few designs, an accurate model of the LHC can be built with a small number of definitions. However, whilst exact magnet geometries are not available for use in the BDSIM model, it is nevertheless important to get as good an approximation as possible.



(a) Beam 1 mean transverse positions.



(b) Beam 1 transverse beam sizes (standard deviation).

Figure 4.4: An optical comparison of BDSIM with MAD-X for beam 1 of the LHC, 800 m either side of IP5. The simulation was run for a single turn with 10,000 primary protons. In both cases, the error bars are smaller than the markers.

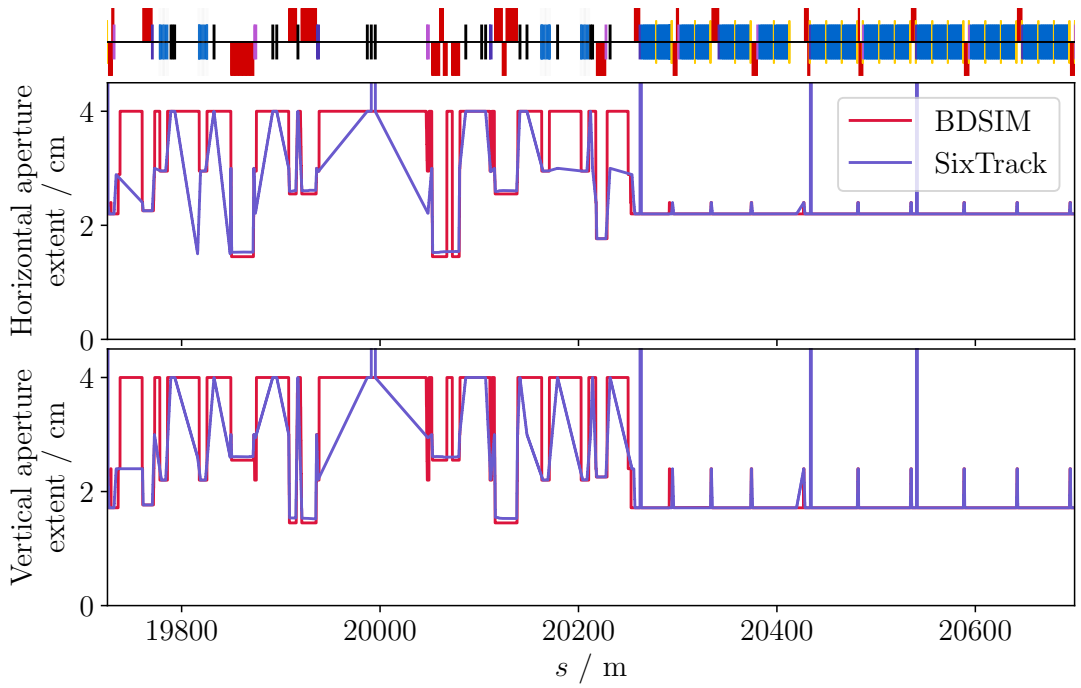


Figure 4.5: A comparison of LHC IR7 apertures between BDSIM, with apertures sourced from SixTrack version 4, and the aperture model used in SixTrack version 5, which was the version of SixTrack used to generate the results presented in this chapter.

This is important for two reasons, firstly that any secondaries will see approximately the correct material at a given radius from the beampipe. Erroneous cross talk between regions, either too much or too little, may manifest itself if the component material or size is incorrect. Secondly, with correct component geometry, the BLMs will be placed at the correct distances from the beampipe, and will see the correct dose. The BDSIM predefined magnet geometry to use for each magnet type is determined by considering its number of beampipes (one or two), the orientation of its beampipes (where there are two) for the given arc section and beam number, the most representative magnet geometry, and finally its width. Table 4.1 shows the complete set of parameters used for various LHC magnet types, and where these values have been sourced from.
The total length of the main arc dipoles (MB) is 17.6 km, which constitutes 66 % of the length of the LHC. Therefore a single accurate main arc dipole geometry greatly improves the accuracy of the LHC model. The main quadrupoles (MQ) make up a total length of 1215.2 m, providing an additional 4.5 % of the total LHC length. These two dual-piped outer magnet geometries are provided by BDSIM, and are shown in Figure 4.6. Additionally, they are set such that the correct beampipe (left or right) is

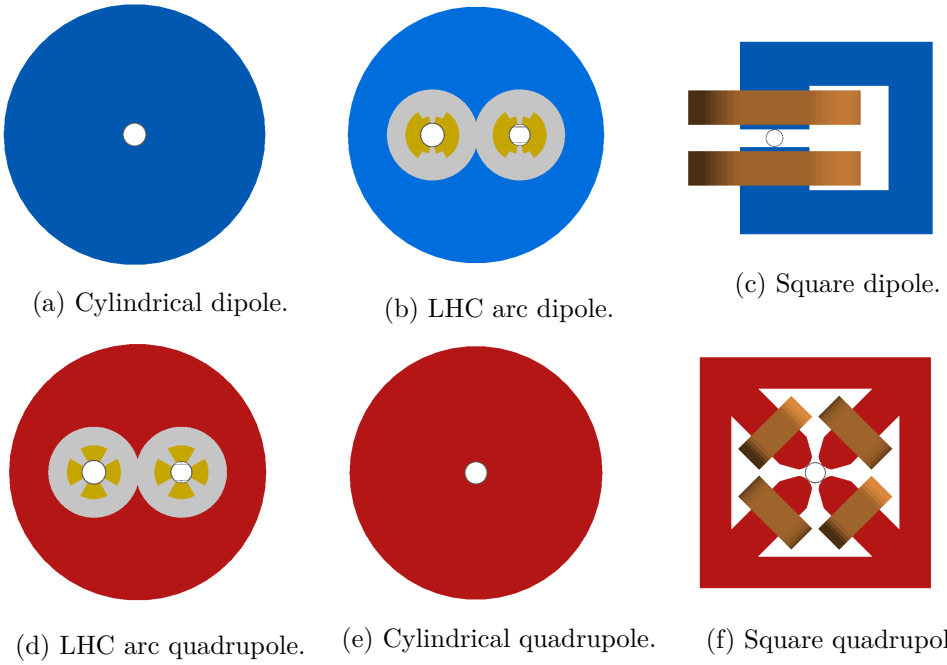
Table 4.1: LHC magnet geometries as used in the BDSIM LHC model. The descriptions and counts were sourced from [92] and the sources used to determine the number of beampipes, the magnet outer widths, and geometry types is also stated on an individual basis for each magnet. The default magnet geometry is in the final row, which is used if none of the others are applicable, which consists of mostly the octupoles, at 53 m in total.

Code	Description	Width / cm	BDSIM geometry	Total	Source
MB	Main dipoles	58	LHC	1232	[93]
MQ	Main quadrupoles	50	cylindrical	392	[92]
MS	Main sextupoles	50	cylindrical	688	[94]
MQX	Inner triplet quadrupoles	50	cylindrical	32	[95]
MCBX	Inner triplet dipole corrector	35	cylindrical	48	[96]
MBXW	IR1/5 separation dipole D1	80	cylindrical	24	[97]
MQML	Insertion quadrupole	100	LHC	32	[92]
MQY	Insertion quadrupole	50	LHC	24	[98]
MBRC	Separation dipole D2	60	LHC	8	[92]
MCBC	Orbit corrector	48	LHC	156	[93]
MCBY	Dipole orbit corrector	50	LHC	88	[92]
MQT	Tuning trim quadrupoles	50	cylindrical	320	[93]
MQM	insertion quadrupole	50	LHC	38	[12]
MQSX	Skew quadrupole (Q3)	18	cylindrical	8	[99]
MBW	Twin aperture warm dipole in IR3 and IR7	100	polessquare	20	[93]
MQW	Twin aperture warm quadrupole in IR3 and IR7	80	polessquare	58	[93]
MCS	sextupole corrector	12	cylindrical	2464	[94]
MCBH/V	Arc dipole corrector	50	LHC	752	[93]
MCBWH/V	Single aperture warm orbit dipole corrector	87	polessquare	8	[100]
Default	Default magnet geometry where not set as above	60	polessquare	166	N/A

used in the correct arc of the ring according to Figure 4.3. In these geometries, only the yoke is present, and the outer vacuum surrounding the yokes is missing, which is inaccurate. In BDSIM the default material of the world is air, so the difference should be minimal when considering energy deposition. However, it does have an impact when considering the placement of the BLMs, as they are placed flush against the side of the magnets. Excluding the vacuum reduces the radii at which some of BLMs are placed, thus increasing the dose. A previous study [77] has shown that simulated BLM dose is very sensitive to the transverse distance between the BLM and the vacuum chamber it is attached to.

A number of magnets do not have specific widths and geometries, instead falling back on the default magnet geometry stated in the final row of the table. For example, no octupole geometry could be sourced for the model. However, the vast majority of the magnets are covered, for example all the magnets in IR7 and the following dispersion suppressor have defined geometries and widths according to the aforementioned rules. Despite this, the geometries are of course generic, e.g., the MBW and MQW (warm collimation insertion magnets), should have two beampipes, but in this model have only one. This may result in more local energy deposition from secondaries simulated in the BDSIM model due to the relative increase in the material close to the aperture. Also, there are 33 thin magnetic elements in the BDSIM model, which, as they are thin, will not have any physical geometry. However, these are without exception corrector magnets in the experimental IRs, and therefore the absence of any geometry should have a minimal impact on energy deposition studies, as the majority of energy deposition occurs in the cleaning insertions.

A tunnel with a realistic shape and offset with respect to the beamline was added using BDSIM's automatic tunnel builder. The material was set to be a perfect black body, instantly absorbing any particle which impacts upon it. This was used to minimise any spurious cross-talk between distant sections of the ring and to minimise simulation time. Considerable effort was expended in removing geometry overlaps in the BDSIM model. This is important as tracking errors will result from erroneously overlapping sections. However a number of tracking bugs persist even without overlaps, resulting in stuck particles. These particles tend to get stuck in certain elements, e.g., a particular TCSG



(a) Cylindrical dipole. (b) LHC arc dipole. (c) Square dipole.

(d) LHC arc quadrupole. (e) Cylindrical quadrupole. (f) Square quadrupole.

Figure 4.6: The full set of dipole and quadrupole magnet geometries used in the BDSIM LHC model. The cylindrical and *polessquare* geometries scale correctly for higher order multipoles, with additional pole tips and busbars being added in the latter case. There are no special magnet geometries defined for any higher order multipoles in the arcs.

and a particular final focus quadrupole. It is not exactly understood to what extent this affects the final result, but as it appears to occur at random, it probably doesn't affect the results meaningfully.

Lastly, two errors during the preparation of the LHC model were identified but ultimately left uncorrected for the results presented in this chapter. Firstly, the primary and secondary collimator materials were incorrectly set to graphite, instead of CFC, and the tertiary collimators were misaligned. The impact of the incorrect primary and secondary collimator material is studied in Section 4.3.2.1, and the impact of the TCT misalignment is apparent in the generated BDSIM loss maps shown in this chapter.

4.2.1 Particle tracking for LHC collimation studies in BDSIM

Accurate tracking over many turns for collimation studies in the LHC is important because by its very design the collimation system will clean beam halo particles over many turns. Particles intercepted by the primary collimator may not acquire an adequately large transverse kick to impact on a secondary collimator until after many turns and passages through the TCP jaws. Accurate tracking may be the difference

between a proton impacting upon a collimator or passing the collimator only to be lost downstream in a sensitive region. Furthermore, the farther a proton hits from the edge of the jaw (i.e. a larger impact parameter), the more collimator material it will see. With a sufficiently large impact parameter a beam halo particle may travel the whole length of the collimator. For this reason, in proton collimation, a larger impact parameter tends to correspond to a more efficient collimation system. Therefore, a spurious drift to larger or smaller amplitudes may result in artificially high or low cleaning efficiency, which motivates the need for accurate tracking over many turns. Tracking codes which can track particles without introducing spurious amplitude growths are referred to as symplectic, which was discussed further in Section 2.5. Non-symplectic behaviour is akin to, for example, a swinging pendulum spontaneously increasing in its amplitude. This may be physical if the pendulum is driven, but if it's not driven, then it is unphysical. This could be especially wrong if the energy of the pendulum is left unchanged, such that the system is simultaneously being driven but without a change in energy, and ultimately amounts to a violation of the conversation of energy. It is important, however, to recognise that this defect is an artefact of the improper integration methods used to solve the differential equations, rather than the original differential equations themselves.

BDSIM was originally developed to study the Compact Linear Collider (CLIC) beam delivery system and laser wires [19, 101]. It was later further developed for studies of the International Linear Collider (ILC) collimation system [102]. Both the ILC and CLIC are linear accelerators. Extending BDSIM for use with circular colliders has introduced new challenges. Tracking inaccuracies which may be insignificant over the relatively small distances associated with linear colliders may manifest themselves over the much greater distances travelled by particles in circular colliders. Secondly, the simulation must be fast enough such that the many hundreds of turns traversed by millions of simulated particles will be done in a practical amount of time. Speed is particularly important when one wishes to compare different collimation configurations. A typical LHC collimation study will simulate the passage of beam halo around the machine for 200 turns, as this is sufficient to remove the vast majority of beam halo from the machine [103]. This turn limit sets a lower bound upon which BDSIM should

ideally accurately track particles for the LHC.

Two physical quantities in particular are important for assessing the accuracy of the tracking, the beam emittance ϵ, and the particle action J. These two quantities are related by the equations

$$\epsilon_x = \langle J_x \rangle \tag{4.1}$$

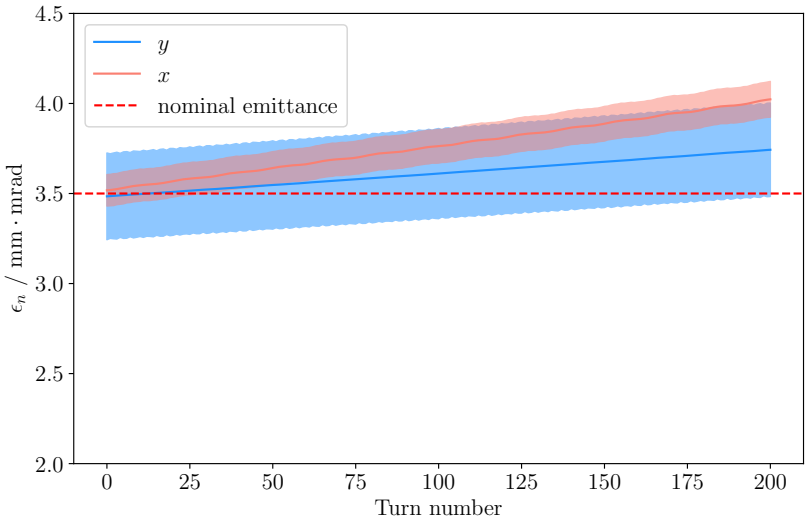
$$\epsilon_y = \langle J_y \rangle \ , \tag{4.2}$$

which is to say that the transverse emittances are simply the mean particle actions in each dimension. These are the invariant amplitudes in phase space, and give a measure of the symplecticity of BDSIM's tracking, and its relation to impact parameter growth. Ideally, these should be conserved outside of particle-collimation interactions (the BDSIM LHC model omits RF cavities) to accurately simulate the particle tracking through the LHC, and by extension, its collimation system. Otherwise, this is unphysical behaviour.

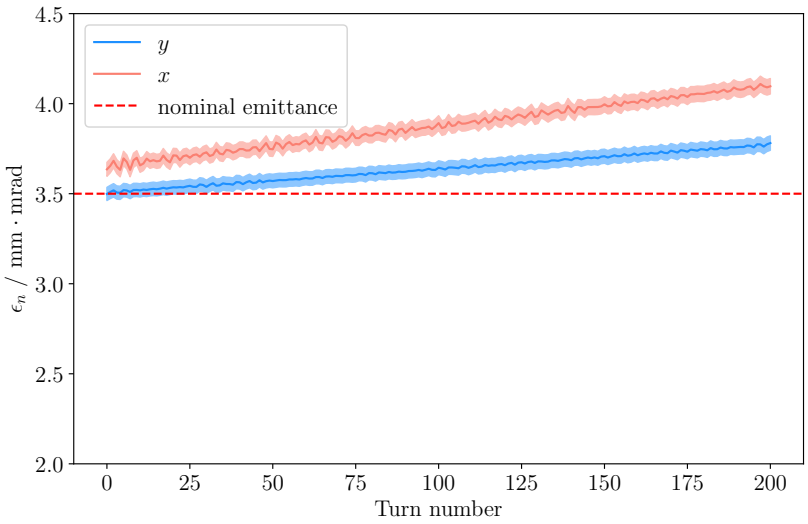
Figure 4.7a shows the normalised emittance for the beam core of protons traversing 200 turns of the BDSIM LHC model. In both transverse dimensions an emittance growth is clear, but it is more prominent in the horizontal plane. Over 200 turns the emittance grows from the nominal value by a factor of $(13 \pm 1)\,\%$ in x, and $(8.4 \pm 0.3)\,\%$ in y. Understanding the source of this non-symplectic behaviour is important if it is to be rectified and its impact on the generated loss maps understood.

All of BDSIM's integrators are nominally symplectic, using either symplectic matrices in the case of linear elements, or the second order semi-implicit Euler method for nonlinear magnets [104], so the choice of integrator for the nonlinear magnets is unlikely to be responsible. However, the integrators for the nonlinear elements are low order integrators, so it is useful to confirm with a model consisting just of linear components that the emittance growth persists. Figure 4.7b shows the emittance for the linear LHC model, and it can be seen to increase at a similar rate, growing by $(13 \pm 2)\,\%$ in x, and $(8 \pm 2)\,\%$ in y over 200 turns, meaning that the element maps themselves are not the cause of the emittance growth.

In both of the above models, the beam cores were tracked. However, in collimation studies, by definition, one is typically not interested in tracking the beam core. It is



(a) The emittances for 170,000 primary protons in the nominal LHC model. The emittance grows by a factor of $(13 \pm 1)\,\%$ in x, and $(8.4 \pm 0.3)\,\%$ in y.



(b) The emittances for 9000 primary protons in the linear LHC model. The emittance grows by a factor of $(13 \pm 2)\,\%$ in x, and $(8 \pm 2)\,\%$ in y.

Figure 4.7: The normalised emittances over 200 turns for the nominal LHC lattice and the linear LHC lattice in BDSIM. The shaded regions denote the statistical uncertainties. The emittance growth persists even in the linear model, where the maps are symplectic.

necessary to determine if this emittance growth is dependent on amplitude, as it may be absent or even worse at large amplitudes. To test this, a $5\,\sigma$ primary halo consisting of just the lobes overlapping with the collimator jaws (used to generate the loss maps presented later in this chapter) was tracked for 200 turns in the BDSIM model with all physics interactions disabled. Disabling all physics processes means that the collimators and geometries are completely transparent, allowing for any emittance growth due to the tracking to be determined in isolation.

The normalised mean horizontal action for the $5\,\sigma$ beam halo is shown in Figure 4.8. The growth is largely undiminished even at large amplitudes in relative terms, growing by $(10.8 \pm 0.1)\,\%$, and it is even worse in absolute terms, as the initial particle action was much larger. This growth in the action corresponds to an even larger growth in the impact parameter b. Typical impact parameters in the LHC collimation system during normal operation are between $0.02\,\mu\text{m}$ and $0.3\,\mu\text{m}$ [81], so this increase is considerable. In reality very few particles will reach 200 turns, but even over a relatively small number of turns, the impact parameter grows markedly, doubling by the tenth turn. For the LHC configuration presented here, the non-integer part of the tune is 0.31, meaning that impacts on the primary collimators should occur approximately once every 8 turns. Therefore, there are typically at least 8 turns of growth if the protons survive the first pass through the collimator jaws. Additionally, any growth will likely be larger than presented here, as the emittance is also increased by impacting on the jaws, and the growth appears to be exponential with offset, so these will compound upon each other. As in this simulation the proton-collimator physics was disabled, it is not immediately obvious to what extent this will impact BDSIM's use for collimation studies. A previous study [103] has shown the relationship between the impact parameter and a number of figures of merit and showed that in general the collimation system's effectiveness for proton beams is largely independent of the impact parameter up until around $10\,\mu\text{m}$, at which point it begins to improve. As this is on the order of the impact parameter growth seen, it is likely that the simulated collimation system in BDSIM should tend to be artificially more efficient.

One way in which BDSIM's unphysical emittance growth might be mitigated is with the use of a one turn map (OTM). Since particle tracking involves the continued

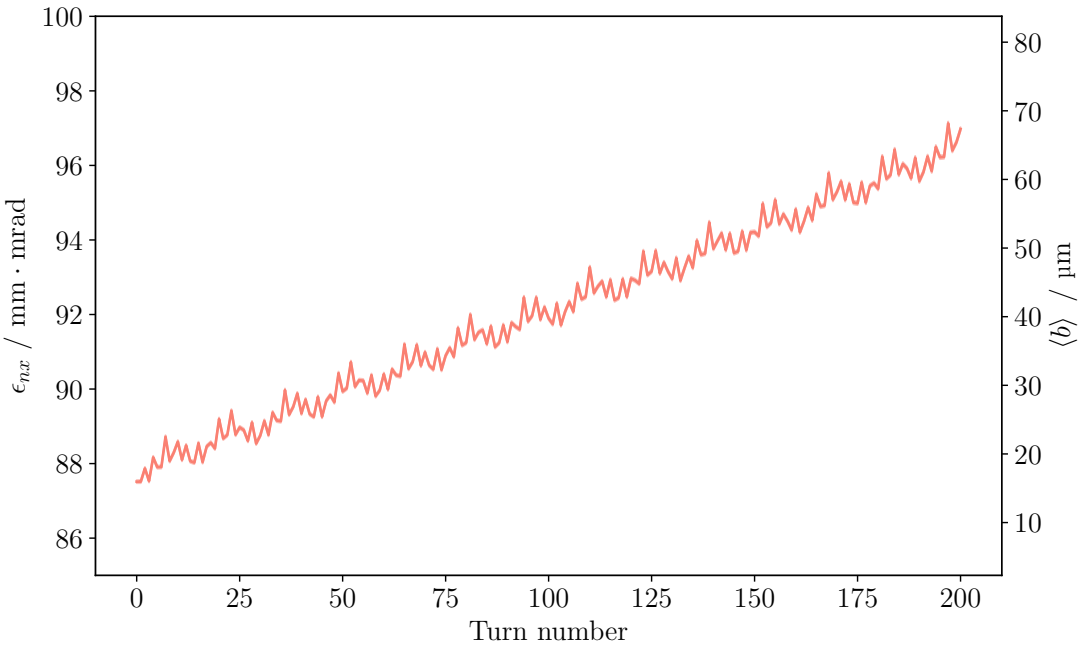


Figure 4.8: The normalised horizontal emittance, ϵ_x, over 200 turns for 13,000 primary protons. The primary distribution used was set at 5σ, just overlapping with the jaws of the primary collimator in IR7. The average impact parameter, $\langle b \rangle$, at the primary collimator for the given ϵ_x, is shown. The maximum normalised action growth over the 200 turns is $(10.8 \pm 0.1)\,\%$, and the mean impact parameter starts at 6 µm and grows by a factor of 13 over the 200 turns. The statistical uncertainties are not resolvable in this figure.

application of maps (corresponding to each component), the consecutive application of these individual component maps over a whole turn can be approximated with a Taylor series. This gives a polynomial that maps a particle from one point in the machine to the same point in the machine one turn later. This can be used in BDSIM by caching the particle coordinates at the end of ring, and then next time the particle reaches the end, the OTM can be evaluated with the cached coordinates from the previous pass, giving the corrected coordinates. These corrected coordinates can then be used to precisely kick the particle back on its correct trajectory, before finally updating the cached coordinates with these corrected coordinates for use if the proton reaches the end of the machine again. The OTM should only be used to correct the primary's coordinates if it did not interact on the previous turn. This is because it does not include any particle-matter interactions in its terms, including only the optics in its description. Instead, the particle coordinates at the end of the ring should be cached for possible use on the following turn.

MAD-X/PTC [36] can be used to generate a OTM and write it to file. The end-of-ring corrector was implemented according to the above description by loading the OTM into BDSIM. The emittance over 200 turns with the use of a 14th order OTM is shown in Figure 4.9, where any growth is limited to, at most one turn's worth, and is not statistically resolvable for the given number of primaries. It is worth noting that the OTM itself is not symplectic, but expanding to the 14th order results in a negligible non-symplectic error over 200 turns.

The source of the emittance growth can be explained in terms of Geant4's tracking internals. Even though BDSIM makes extensive use of component maps, which nominally take a particle from the front of an element to the end of it, it cannot be achieved outside of the Geant4 framework of chords and arcs used to determined where a particle has crossed a boundary. In general particles in fields will travel on curved trajectories, and chords are used to approximate these trajectories. The tolerance at which a chord is considered to be close enough at a boundary to the corresponding arc is referred to as the "delta intersection", or ΔI, which is a tuneable parameter set before the running of the simulation. This is demonstrated in Figure 4.10. The algorithm is biased in that the chosen, estimated point of intersection, is always on the

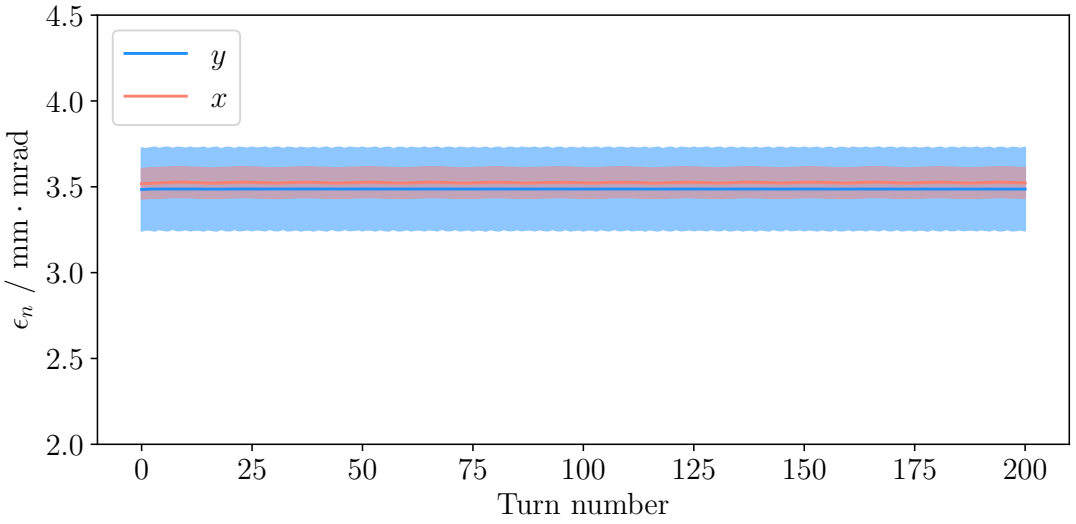


Figure 4.9: The emittance at the start of the LHC model every turn for 200 turns, for 9000 primary protons with the use of a one turn map at the end of the ring to reset the particle back onto a physical trajectory. The emittance growth in both dimensions is not resolvable over 200 turns. The shaded regions denote the statistical uncertainties.

inside of the arc. For this reason it is typically advised to keep ΔI as small as possible, but it cannot be equal to 0, and as a result this biased approximation of the true intersection with a boundary will always result in an unavoidable emittance growth. This approximation of the intersection at each element boundary is, in effect, an instantaneous transverse kick by an amount no greater than ΔI. This choice gives an upper bound on the emittance growth per element, and by extension, per turn. Therefore, a smaller ΔI is better, but this comes at the cost of increased computation time. This also explains the oscillatory behaviour in the observed emittance growths—depending on the particle's phase, the kick, which always occurs in the same direction, will over time tend to increase the particle action. However, as the kick is always in the same direction, this effect will be modulated by the phase advance. The growth is most clearly demonstrated with a reference particle. By definition, in a perfect machine the reference particle will stay on the reference orbit forever, which does not happen in BDSIM. Figure 4.11 shows the normalised particle action, γJ_x, over 10,000 turns for a primary which starts off on the reference trajectory, i.e. with $\gamma J_x = 0$. This is done for a range of different ΔI choices, where $<10\,\text{nm}$ is the default value used in BDSIM. Even though the particles start on reference, any slight offset will immediately, result in a transverse offset, which result in betatron oscillations. The

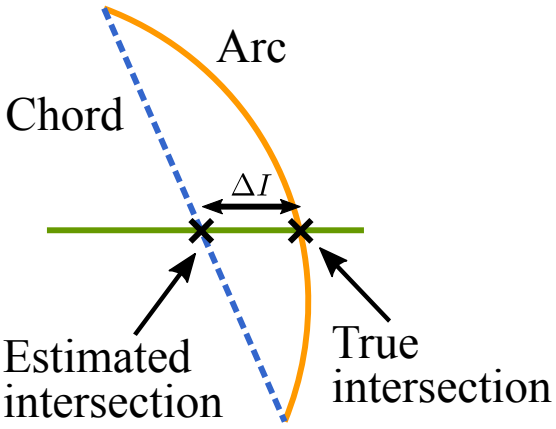


Figure 4.10: The cause of the non-symplectic tracking in BDSIM. At the boundary of each component in Geant4, the intersection of a particle's trajectory with that boundary is approximated with a chord. This will always underestimate the transverse position, resulting in what is effectively a transverse kick. This results in non-symplectic behaviour.

phase advance at each element will then determine whether the kick results in an instantaneous increase or decrease in J_x. It's also worth noting that not only does the emittance grow, but the rate of growth also increases. This is consistent with the effect shown in Figure 4.8, where the growth over 200 turns is far larger than the growth over 200 turns for the reference particle. As the betatron amplitudes become larger, the error per boundary crossing will approach the maximum, ΔI. By default in BDSIM this is 10 nm, which is small, but the BDSIM LHC model has several thousand of these boundaries in magnetic fields, resulting in several thousand of these 10 nm kicks per turn.

The effect of the emittance growth can be corrected in one of two ways, either with a OTM, as shown above, or with a separate tracker, detailed in Section 4.2.2 with which the initial aperture hits are generated. The difficulty with the OTM is that it should only be applied if the proton does not interact on the previous turn. In general this cannot be guaranteed in a collimation study, as hits will occur with the collimators. In BDSIM it's possible to record with a flag whether or not an interaction occurred on the previous turn, but this is not a robust process and it was observed on a number of occasions that a particle could interact without Geant4 recording it as such. For this reason, the OTM was not used to produce the results generated in this thesis, as the error associated with the emittance growth is more intuitive and predictable than that resulting from an erroneously applied one turn map at the end of each turn.

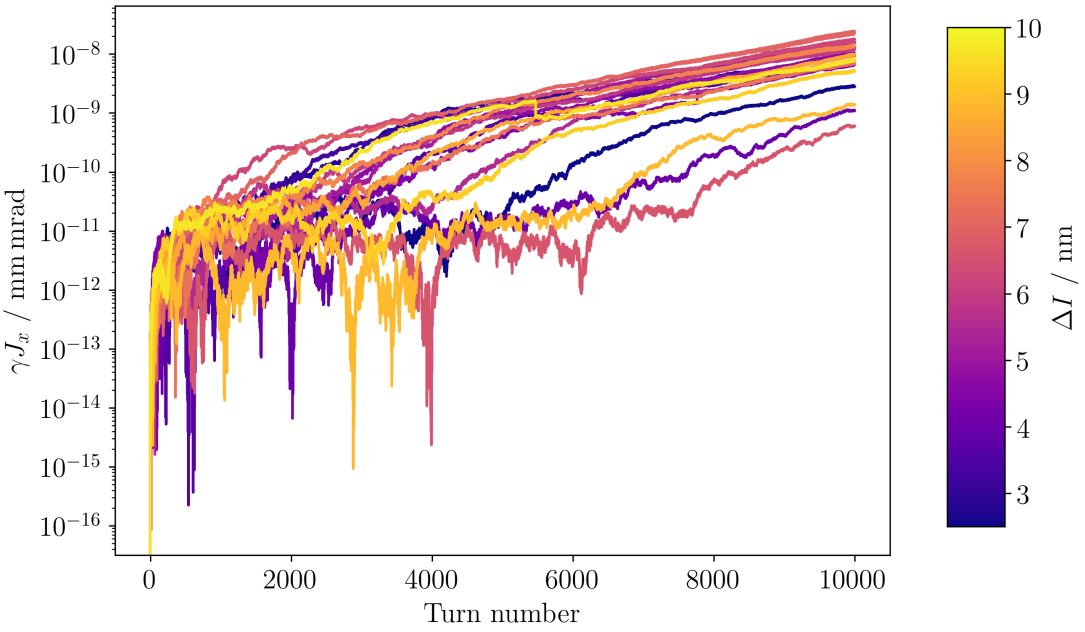


Figure 4.11: The normalised particle action for a reference particle (i.e. $J_x^0 = 0$) over 10,000 turns for 20 different values of ΔI. Whilst there is a degree of noise close to the reference trajectory as this is the region where the error will be smallest, in the long run as the particle leaves this region, the exponential nature of the emittance growth is apparent.

4.2.2 Proposed tracker design for BDSIM

As stated above, the use of a OTM is limited in its applicability due to the difficulty in determining whether or not a particle has undergone a physics process in a collimator. Therefore, overcoming the tracking accuracy limitations presented by the BDSIM Geant4 model will ultimately require a separate, dedicated tracker run in conjunction with the main BDSIM Geant4 model. This approach offers two additional benefits over a working OTM-based solution in that it will be faster and it will be possible to integrate physics interactions such as beam-collimator interactions, synchrotron radiation and beam-gas interactions into the tracking.

A separate tracker will be faster because it will completely exclude the boundary intersection algorithms, and generally completely sidestep the Geant4 framework which is heavily geared towards detector simulation over fast and accurate long term tracking for many hundreds of kilometres. A Geant4 simulation will typically spend the majority of the time in its geometry and particle tracking routines, so removing this will immediately offer a considerable speedup. For example, Geant4 will propose a step,

which may be revised multiple times, but in a tracker the particle can be stepped directly from the start of an element to its end at once, without need for further revision. Additionally, Geant4 tracks one particle at a time, but if instead whole bunches of particles are tracked together, then the tracking algorithms will benefit from locality of reference and see further speedup. In total SixTrack is able to track one primary around one turn of the LHC about one thousand times faster than in BDSIM. This factor of a thousand presents an approximate upper limit on the possible speedup. Lastly, the tuneable parameter δI described above is set as small as possible to preserve the LHC tracking for as long as possible, but this comes at the cost of increased tracking time. With an external tracker the parameter could be increased substantially as the main Geant4 model would no longer be used for long-term tracking, thus also speeding up the main BDSIM Geant4 simulation as well.

To integrate the tracker with the collimator physics-interactions would require the treatment of collimators separate from the main tracking algorithms. Whilst SixTrack has its own bespoke particle-matter physics routines, as BDSIM is fundamentally inseparable from Geant4, it makes sense to incorporate these validated and proven routines into the dedicated tracker for its collimation physics. To do this it would be necessary to build each collimator geometry (without the rest of the beamline) in Geant4 and then inject primary protons from the tracker into the Geant4 collimator world and track them through the collimator jaws. With the use of a sampler at the end, any on-energy surviving proton can be reinjected back into the tracker.[2] Any particles not injected back into the tracker after passing through a collimator could instead be injected into the main BDSIM model and their local energy deposition simulated. As sufficiently off-energy particles are unlikely to travel very far before being lost, the long term tracking accuracy provided by the external tracker is unnecessary, and BDSIM's will suffice.

One of BDSIM's most powerful features is that for a given primary event it is possible to connect all energy deposition, sampler hits, aperture impacts, etc., from that primary particle from and any of its resulting secondaries. This enables one to study, for example, correlations between primary losses in a given collimator and subsequent

[2] SixTrack optionally supports using Geant4 for its collimator-physics routines instead of its own, and works in much the same way as described here.

downstream energy loss. This is achieved by filling event-level structures in memory before writing them to a ROOT file at the end of the event. It would be preferable in the implementation of the tracker to preserve this capability, however in tracking many bunches at once one must therefore fill the event-level structures in memory before writing at the end. For a sufficiently large number of bunch particles this will result in a drastic increase in the memory footprint of the simulation. It would be necessary therefore to determine a trade-off between the memory footprint and particles tracked per bunch, and in the limit of the highest energy machines such as the LHC, it may be necessary to track one particle at a time in the tracker and forgo the concept of bunches altogether.[3] In doing this it would be possible to achieve the ideal scenario in which the same ROOT output is acquired without the tracker, except with more accurate tracking and in less time.

Finally, whilst not necessary for LHC collimation studies, beam-induced backgrounds (BIB) studies described in Chapter 5 could also benefit from a dedicated tracker, albeit with one further extension to the design described so far. Beam halo impacting on the TCTs just upstream of the experiments will result in particle showers, some of which will be detected in the detectors. These signals in the detectors from upstream can present issues for a variety of reasons, outlined in detail in Chapter 5. For the purpose here of outlining a tracker design, it is only necessary to state that particles may scatter elastically off residual gas molecules and impact either directly on the TCTs or even closer to the detector, contributing to the background rate. The simulation of elastic beam-gas events in conjunction with particle tracking around the LHC has been studied in detail using SixTrack [62, 105]. However, it should also be possible to do similar studies in BDSIM with the tracker. This could be achieved by leveraging the Geant4 `GetMeanFreePath` and `PostStepDoIt` methods common to all physics processes. The method `GetMeanFreePath` proposes a step length for that process and a given particle in a given medium, and `PostStepDoIt` applies the physics of the process to the particle. The step length proposed by `GetMeanFreePath` may be less than the length of the component to be transported through. This possibility of step lengths less than the component length alters the design from that of a conventional particle tracker

[3] One possible solution in principle might be to write the data to ROOT output on the fly, but ROOT is explicitly designed for write-once in contrast to incremental writing, so this is not an option.

where the step length is always equal to the component length. In conventional particle trackers the algorithms will simply transport a particle from the start of an element directly to its end, and so on for every subsequent element in the lattice. With the necessity of `GetMeanFreePath` for the particle tracker, it must be possible to determine precisely where in the lattice the particle is. In general a given particle will be inside an element, not between elements as is the case with a conventional tracker. By querying Geant4's physics processes in this way, with a variable step length tracker, it would be possible to study the effects of elastic beam-gas in BDSIM. Other applications, such as synchrotron radiation, are a direct extension of its application to beam-gas, simply necessitating calling `GetMeanFreePath` and `PostStepDoIt` on a different process. Whilst synchrotron radiation is not a concern for the conventional LHC, runs involving partially stripped ions [106] may benefit from this feature in particular.

4.3 BDSIM comparisons with SixTrack

In this section comparisons between BDSIM and SixTrack are presented.
SixTrack version 5 [91] was used to produce the results presented in this chapter. The cleaning inefficiency is used in this section to define the local losses around the ring, and is defined as

$$\eta = \frac{N_{\text{local}}}{\Delta s N_{\text{coll}}}\,, \tag{4.3}$$

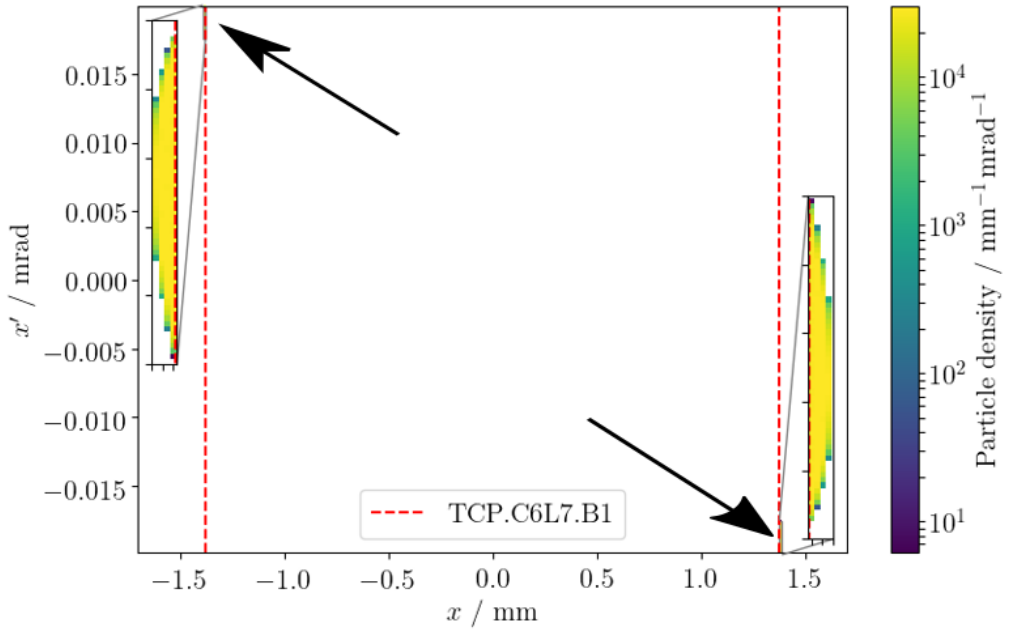
where N_{local} is the local loss, N_{coll} the total number of losses in the collimation system and Δs is the length over which the local losses are binned. Another quantity, the *global* cleaning inefficiency is also used and is defined as

$$\eta_{\text{global}} = \frac{\Sigma N_{\text{aper}}}{\Sigma N_{\text{coll}}}\,, \tag{4.4}$$

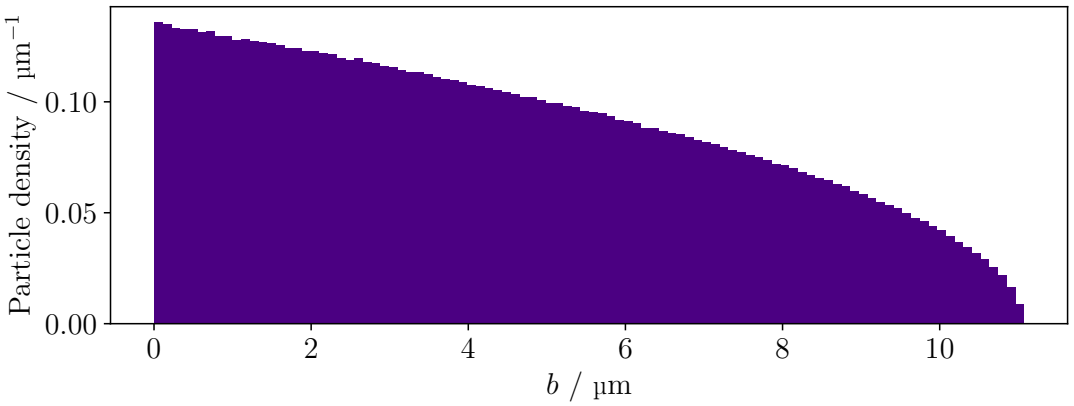
which is defined as the ratio of the total aperture losses (i.e. not inside a collimator), ΣN_{aper}, to losses within all collimators, ΣN_{coll}. In general a better performing collimation system will have a lower global cleaning inefficiency and is a useful figure of merit for evaluating different configurations.

4.3.1 Primary distribution

The SixTrack primary distribution was generated using the undocumented `IPENCIL=3` distribution, in which only the section of phase space directly overlapping both the collimator jaws is populated. This is used to generate a reasonable mean impact parameter, $\langle b \rangle$, with all the protons impacting immediately as soon as particle tracking begins. Alternatively, a thin annulus around $5\,\sigma$ (the normalised primary collimator opening) may be used, and tracked around the ring, depopulating the primary halo distribution over many more turns. One advantage of this is that any deformation of the halo due to nonlinearities, which will be much more apparent at larger radii, as well as machine imperfections, can be included. However, starting the distribution directly in front of the collimator results in a faster simulation time and is much simpler as all the protons will immediately impact on the jaws, requiring less fine-tuning.
Additionally, as was shown in the previous section, BDSIM's multi-turn tracking is inaccurate, so starting the beam directly in front of the collimator will minimise this effect. SixTrack uses a thin representation of the lattice, with the collimators also represented with thin markers. These markers are situated at the centre of the corresponding thick collimator, and as a result, the primary distribution that is generated at the collimator in SixTrack is also generated at the centre. These markers denote where SixTrack should hand over to the collimation routines, and the backtracking both before and after occurs as part of these routines. The collimation routines have no output functionality, so the output distribution must be written by SixTrack, which is the one at the centre of the collimator. This means that for the primary distribution to be used in BDSIM the distribution must be backtracked with an inverse drift as a preprocessing step.
The 2D primary horizontal phase space is shown in Figure 4.12a, with the zoomed lobes shown inset. A small amount of phase space is populated, giving a realistic mean impact parameter of 4.4 µm. The distribution begins from the edge of the collimator so that only particles that will immediately hit the collimator are generated. The impact parameter for this distribution is shown in Figure 4.12b.
The phase space probability density functions in y and y' are Gaussian such that there is no halo in the vertical phase space. By using exactly the same distribution in



(a) The primary horizontal phase space. The lobes are shown zoomed inset, with the collimator jaws marked. The halo lines up perfectly with either side of the collimator jaws.



(b) The impact parameter for the primary distribution situated in front of the horizontal collimator in IR7. The mean impact parameter is 4.4 µm.

Figure 4.12: The primary distribution in horizontal phase space and their corresponding impact parameters on the primary collimator jaws.

BDSIM as was used in SixTrack, this removes an additional source of potential disagreement between the two codes.

4.3.2 Proton-collimator physics

One potential source of disparity between BDSIM and SixTrack is that of the proton-collimator interactions. As BDSIM simply wraps Geant4, it has access to its full set of validated particle physics processes. SixTrack, on the other hand, implements its interactions with the dedicated routines of K2/COLLTRACK [40, 44]. Differences between the proton-matter physics libraries may have a significant effect on the simulated collimation system performance. Therefore, it is necessary to directly compare the products resulting from protons impacting on the primary collimator in BDSIM and in SixTrack.

In the BDSIM model a sampler was placed directly following the collimator, and in SixTrack the products were written to files using the `DUMP` command. The primary horizontal collimator in IR7, TCP.C6L7.B1, was used for the comparison. Examining the products from this collimator is useful as it is the first point of impact for loss maps presented in this chapter, and may explain any subsequent differences between BDSIM, SixTrack, and BLM data. The primary distribution detailed in Section 4.3.1 was used for both simulations, with 6.4×10^6 protons simulated in both codes, and whilst BDSIM can track all secondary particles, only protons are shown here. Furthermore, as BDSIM will otherwise produce protons all the way down to the rest mass scale, a kinetic energy cut of 650 GeV (10 %) was also applied. As SixTrack will only produce relatively on-energy protons, this cut is used to produce a comparable distribution, which is a subset of the full set of particles exiting the collimators (both primaries and secondaries) in BDSIM.

Figure 4.13 shows the energy deviation normalised with respect to the nominal beam energy, $\delta = \Delta E/E_0$. The energy deviation of the proton will in part determine its lifetime within the machine. Beam protons very far off-energy which escape IR7 will likely impact on the aperture in the dispersion suppressor immediately downstream, something which must be minimised for the safe operation of the machine. Alternatively, they may be lost in the momentum cleaning insertion which is designed

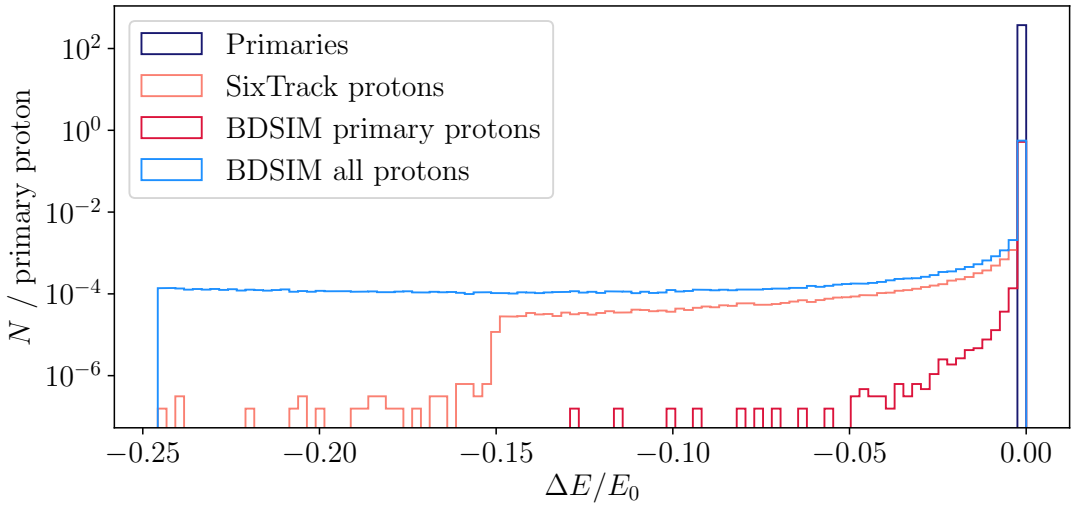
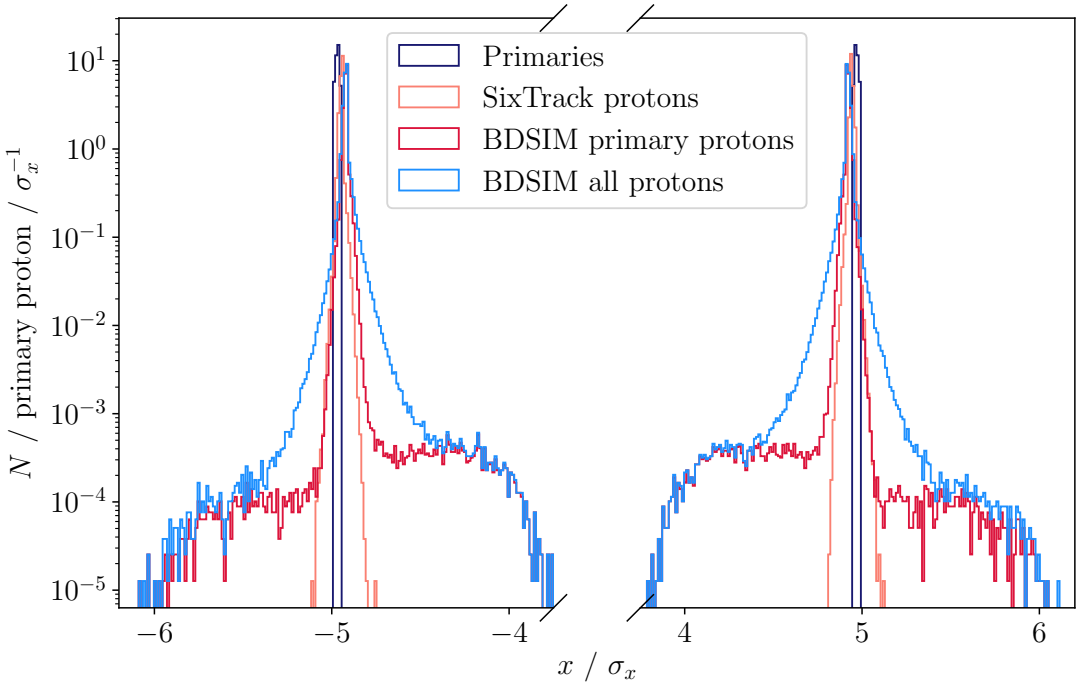


Figure 4.13: The energy offset, $\delta = (E - E_0)/E_0$ for protons immediately following the horizontal primary collimator in IR7, TCP.C7L7.B1. The impacting primary protons all had $\delta = 0$, i.e. they were all perfectly on-energy.

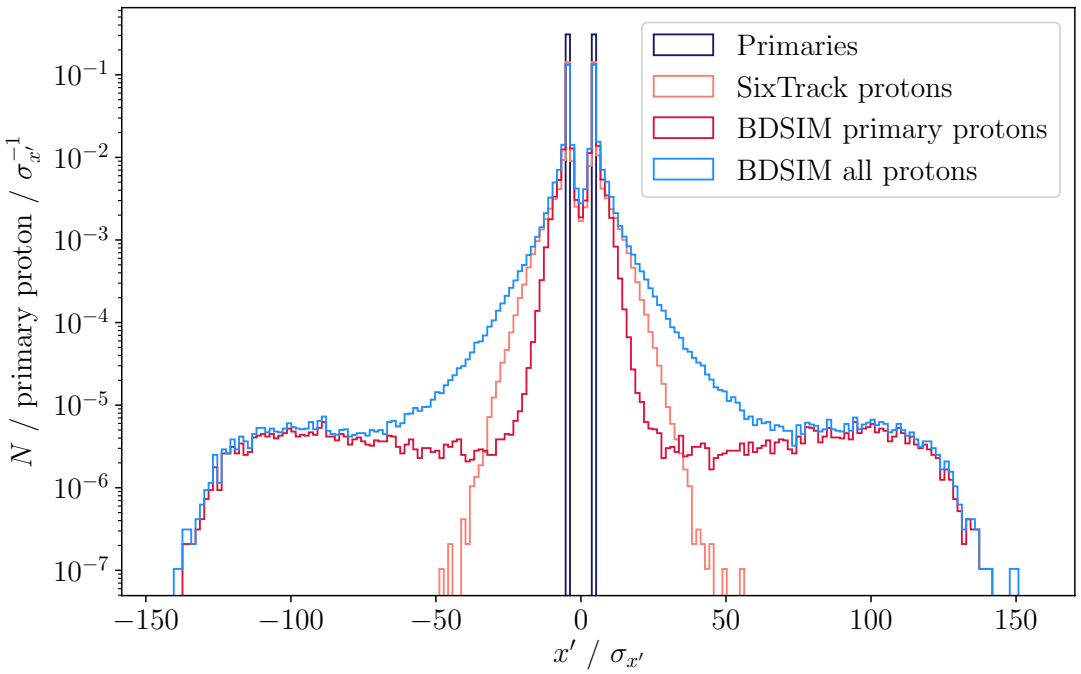
to clean protons with $|\delta| > 10^{-3}$. The total flux of protons escaping with $|\delta| < 0.1$ is only slightly larger in BDSIM than in SixTrack, at 0.57 and 0.54 per impacting proton, respectively. This 5 % increase in products will likely result in energy deposition and losses spread over a greater range in s in IR7 and the dispersion suppressor.

The transverse phase space projections are shown in Figure 4.14, where the the effect of the material on the transverse phase spaces is clear, with the primary distribution in all four cases appearing convolved with a Gaussian distribution. The key feature is that the tails in the horizontal phase space are up to 5 times broader in BDSIM than in SixTrack. The agreement in all four cases is very good, but the long tails in the horizontal distributions may result in certain features present in the BDSIM loss map which are absent in SixTrack. This kicking of protons to much larger amplitudes, which appear to otherwise be absorbed in the SixTrack simulation, are at such large amplitudes that they are extremely likely to be lost immediately on the aperture, or one of the secondary collimators.

As the aperture model is very similar and BDSIM can record losses in the same way that SixTrack does, any disparities can most likely be explained in terms of the particle-matter physics routines, BDSIM's tracking (particularly that it is non-symplectic), as well as the model preparation errors pertaining to the incorrect collimator materials and TCT misalignments.



(a) Horizontal position x. The x-axis in this figure is discontinuous so that only the relevant regions around the jaw edges is displayed.



(b) Horizontal angle x'.

Figure 4.14: The transverse phase space coordinates before and after impacting on the 0.6 m-long horizontal primary collimator in IR7, TCP.C6L7.B1. The primary distribution is recorded immediately prior to the collimator, and the products are recorded immediately after it.

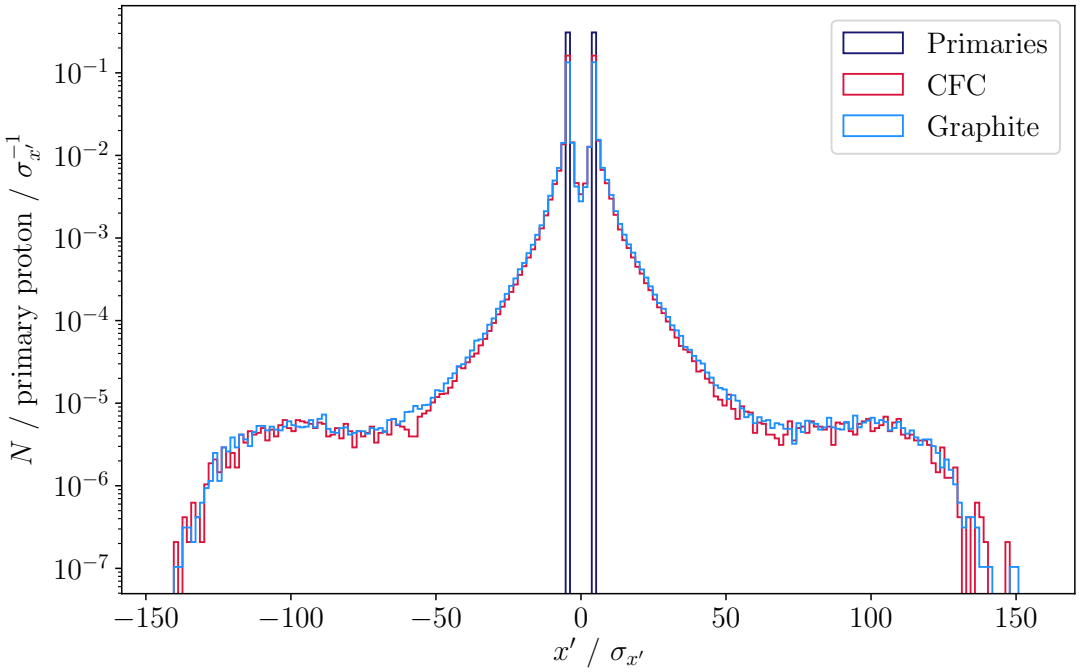


Figure 4.15: The angular distributions after a single pass through the TCP with the beam halo.

4.3.2.1 Primary collimator material density effects

Due to an error during the preparation of the model, the energy deposition and BLM dose simulations had the wrong material set for the primary and secondary collimators. They are supposed to be CFC, but were erroneously set to graphite. The only difference between CFC and graphite is in the density, where CFC has a density of $1.67\,\mathrm{g\,cm^{-3}}$ [107], and graphite has a density of $2.27\,\mathrm{g\,cm^{-3}}$. The relationship between the density, the scattering angle and the stopping power is nonlinear, meaning that the effect is not trivially identifiable. The most appropriate thing to do is simply to simulate the effect directly in BDSIM to compare the different materials. As the first point of impact in the machine is the primary collimator, it is important to determine any effect this error may have. Figure 4.15 shows the resulting scattering angles for a CFC and a graphite collimator jaw, with the same primary distributions as used above. Whilst any differences in the tails of the distributions are negligible, the overall rate of products escaping the lower density, CFC collimator, is $(13.2 \pm 0.1)\,\%$ larger than in comparison with graphite.

The difference between the CFC and graphite collimators is large and would likely

positively affect the simulated cleaning efficiency. To give a best possible comparison between SixTrack and BDSIM, the comparable BDSIM loss maps were regenerated with the correct collimator material, CFC. However due to the considerable computation time that the energy deposition and BLM studies required, these were not regenerated, and this represents a source of systematic error. This increased density is likely to result in more losses confined to the primary and secondary collimators, and IR7 in general.

4.3.3 BDSIM LHC loss maps and SixTrack

In SixTrack, a loss is recorded when a primary particle exceeds a locally defined aperture or undergoes an inelastic collision inside of a collimator. The simulation of a particle ceases if it is lost in either one of these two ways. BDSIM will also track primaries up until they are lost due to particle-matter interactions, but in contrast to SixTrack, will also simulate the passage of resulting secondaries through the model. However, for the sake of making comparisons between the two codes, it is useful for BDSIM to be able to record losses in the same way as SixTrack. BDSIM can do this by recording, in addition to all other data, such as energy deposition, the *collimator hits* and *aperture impacts*. These allow one to record all interactions in collimators and all points in the lattice where particles exceed the locally defined aperture. By default, for the sake of minimising required storage space, only the primaries are recorded, but as was shown in Figure 4.13 in the previous section, the definition of "primary" in BDSIM is stricter than in SixTrack. Where appropriate the impacts and losses will be recorded for all protons with kinetic energies above 5.5 TeV, which will give a more meaningful comparison between the two codes.

As has been mentioned, both simulations used the exact same primary distribution described in Section 4.3.1. Furthermore, the same collimator openings, stated in Table 4.2, were used in both simulations. Lastly, a very similar aperture model was used. SixTrack version 5 [91] introduces a new aperture model, whereas the BDSIM aperture model is sourced from SixTrack version 4 [108]. As the real LHC aperture has not changed between the time that these two versions were released, differences between the two aperture models is minimal, as was shown in Figure 4.5. Lastly, the

Table 4.2: The machine parameters for the LHC loss map simulations in this chapter. The nominal normalised emittance is 3.5 mm · mrad, but it varies from fill to fill and is typically closer to 2.5 mm · mrad in Run II. A larger emittance of course means a larger beam and the convention is to simulate the nominal parameters as they are more unfavourable for the operation of the machine.

Parameter	Simulation (2018)	Nominal	Units
E	6.5	7	TeV
β^*	30	55	cm
ϵ_n	3.5	3.5	mm · mrad
TCP IR7 opening	5.0	6.0	σ
TCSG IR7 opening	6.5	7.0	σ
TCLA IR7 opening	10.0	10.0	σ
TCP IR3 opening	15.0	15.0	σ
TCSG IR3 opening	18.0	18.0	σ
TCLA IR3 opening	20.0	20.0	σ
TCT IR1 and IR5 opening	8.5	8.3	σ

primary protons were tracked for a maximum of 200 turns, by which point very few protons remain.

The fractional survival per turn is shown in Figure 4.16. Two features are notable, firstly, the impact of the betatron tune is clearly visible, with the fractional survival dropping periodically, consistent with the non-integer part of the betatron tune, $q = 0.31$. For example, there are sudden increases in the losses at turn 8 ($0.31 \times 8 = 2.48$) and turn 16 ($0.31 \times 16 = 4.96$), and so on. Secondly, the collimation system in the BDSIM model seems to be far more efficient at removing the halo from circulation. In SixTrack, $(0.938 \pm 0.003)\,\%$ of halo particles survive after 200 turns, but in BDSIM, only 2.77 ± 0.05 in 10,000 primaries survive, a factor 100 difference. This can be explained in terms of the non-symplectic tracking, manifesting itself as a growth in the particle action, shown previously in Section 4.2.1, as well as the TCT misalignments. A larger impact parameter means the primary will see more of the primary collimator material each turn, either undergoing a kick sufficient to impact upon one of the down stream secondary collimators, or an inelastic collision in the collimator, resulting in a decreased chance of survival.

The simulated cleaning inefficiencies across the whole LHC in SixTrack and BDSIM are shown in Figure 4.20. A number of features stand out in these loss maps. The spikes situated around the ring largely correspond to the different IPs, with few scattered

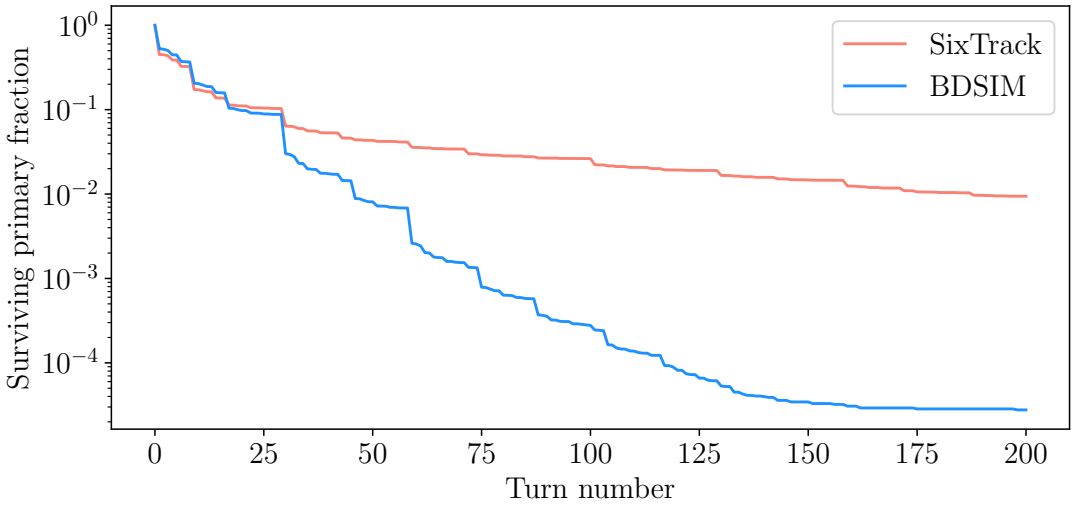


Figure 4.16: Fractional surviving primaries at the end of each turn of the LHC. 7.4×10^6 primaries were tracked BDSIM and 1.7×10^6 primaries tracked in SixTrack.

losses around the ring in the arcs. The collimation hierarchy in IR3 is in agreement in both codes. Good agreement at the macroscopic level in IR7 and the following dispersion suppressor is apparent. Lastly, there are a large number of cold losses distributed across IR8 and IR1, both of which are shown in greater detail in Figure 4.18, with similar deposition in SixTrack totally absent. This is possibly a result of the tertiary collimators that are misaligned in the BDSIM model.

The cleaning inefficiencies in IR7 are shown in Figure 4.17, where the collimator hierarchy can be seen to be faithfully reproduced in BDSIM, with decreasing losses further from the primary collimator as the collimator openings increase. The excess of warm losses immediately following the primary collimators in BDSIM may be explained in terms of the differences in the interactions with the primary collimator, shown in Figure 4.14. The BDSIM collimator products extend to much larger amplitudes, with long tails extending several σ out from the primary. SixTrack, on the other hand, has no such tails. These large-amplitude protons in BDSIM may then register as aperture impacts almost immediately after leaving the collimator. The extra warm depositions downstream of the secondary collimators may be explained in a similar fashion.

The integrated losses per primary over a range of regions comparing SixTrack and BDSIM are shown in Figure 4.21. Broadly similar agreement is shown except for a a clear excess in the losses in the TCTs. As has already been mentioned, the TCTs were incorrectly aligned. This was later discovered to be due to a misalignment in the lattice

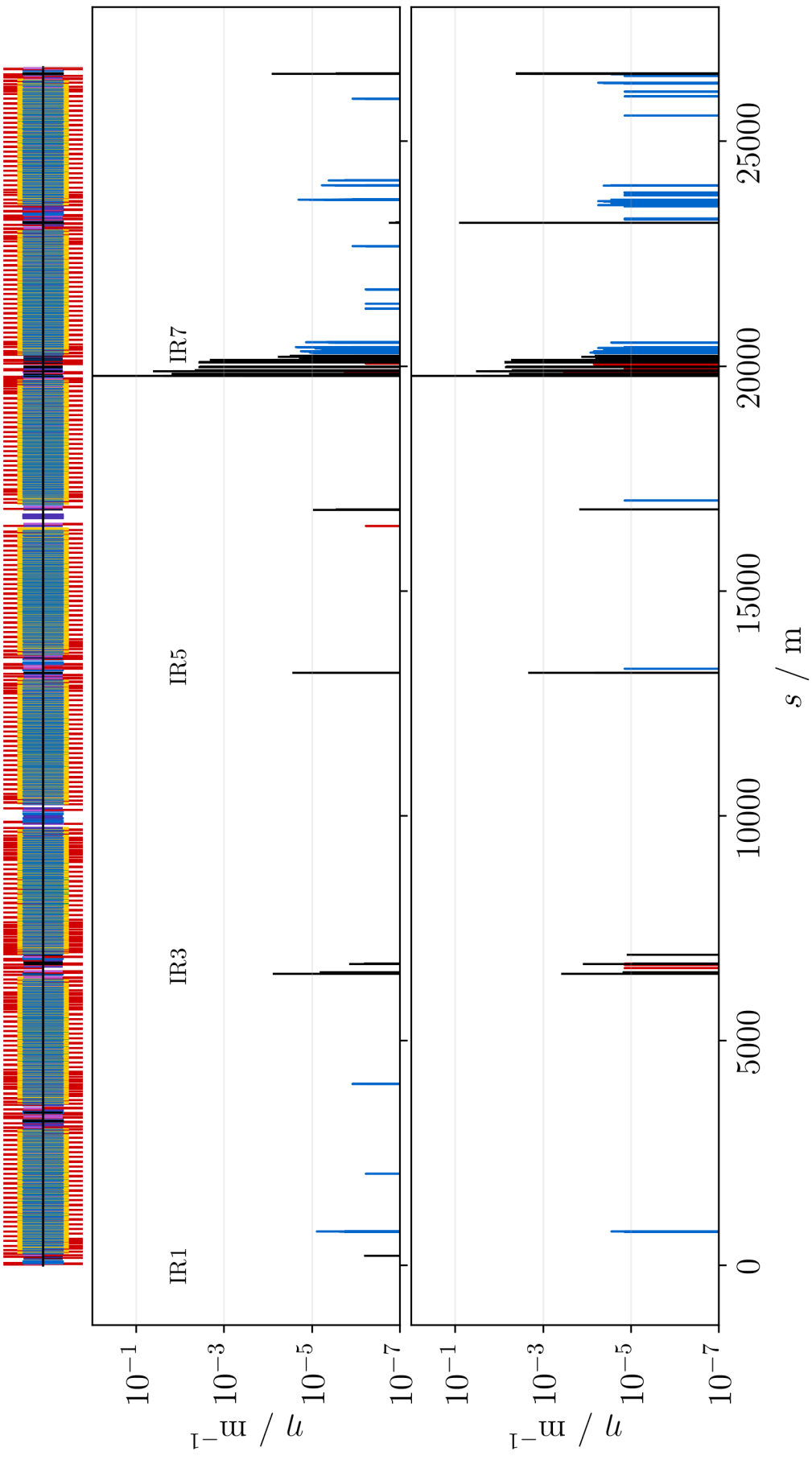


Figure 4.17: Cleaning inefficiency for SixTrack (top) and BDSIM (bottom).

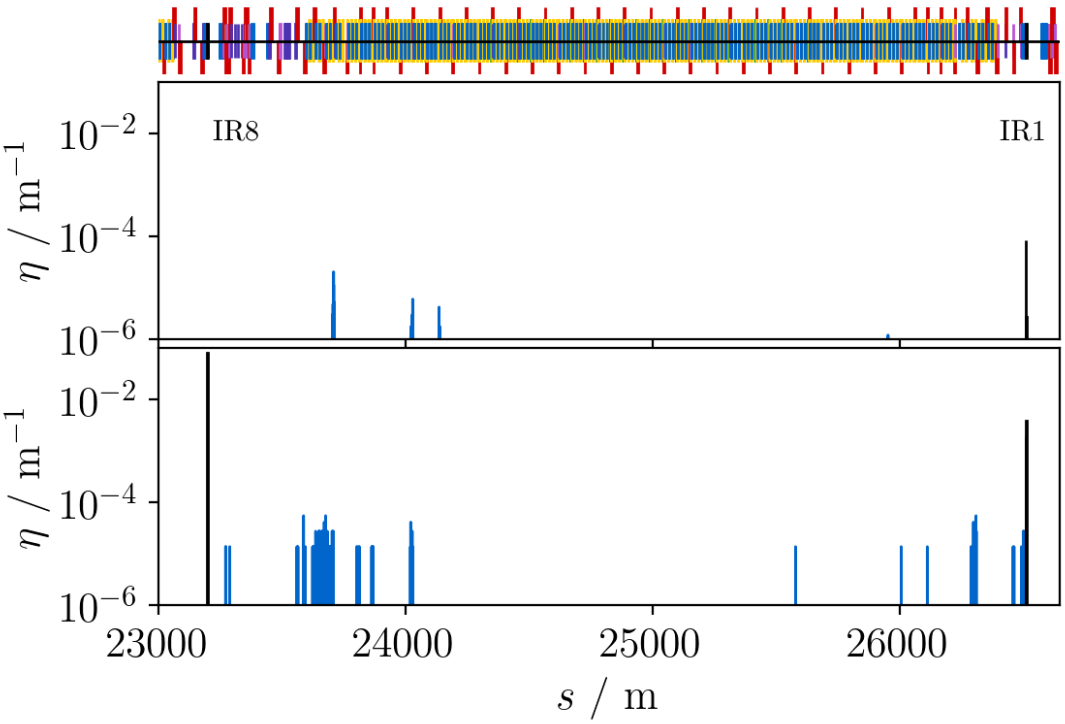


Figure 4.18: Cleaning inefficiencies in SixTrack (top) and BDSIM (bottom) across IR8 and IR1.

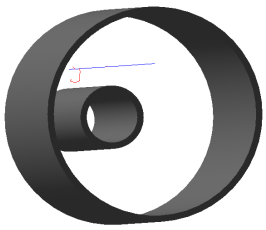


Figure 4.19: Visualisation in BDSIM of a proton travelling from a beampipe with a large radius into a beampipe with a much smaller one. The transition between the two is not smooth and there is a radial gap between the two.

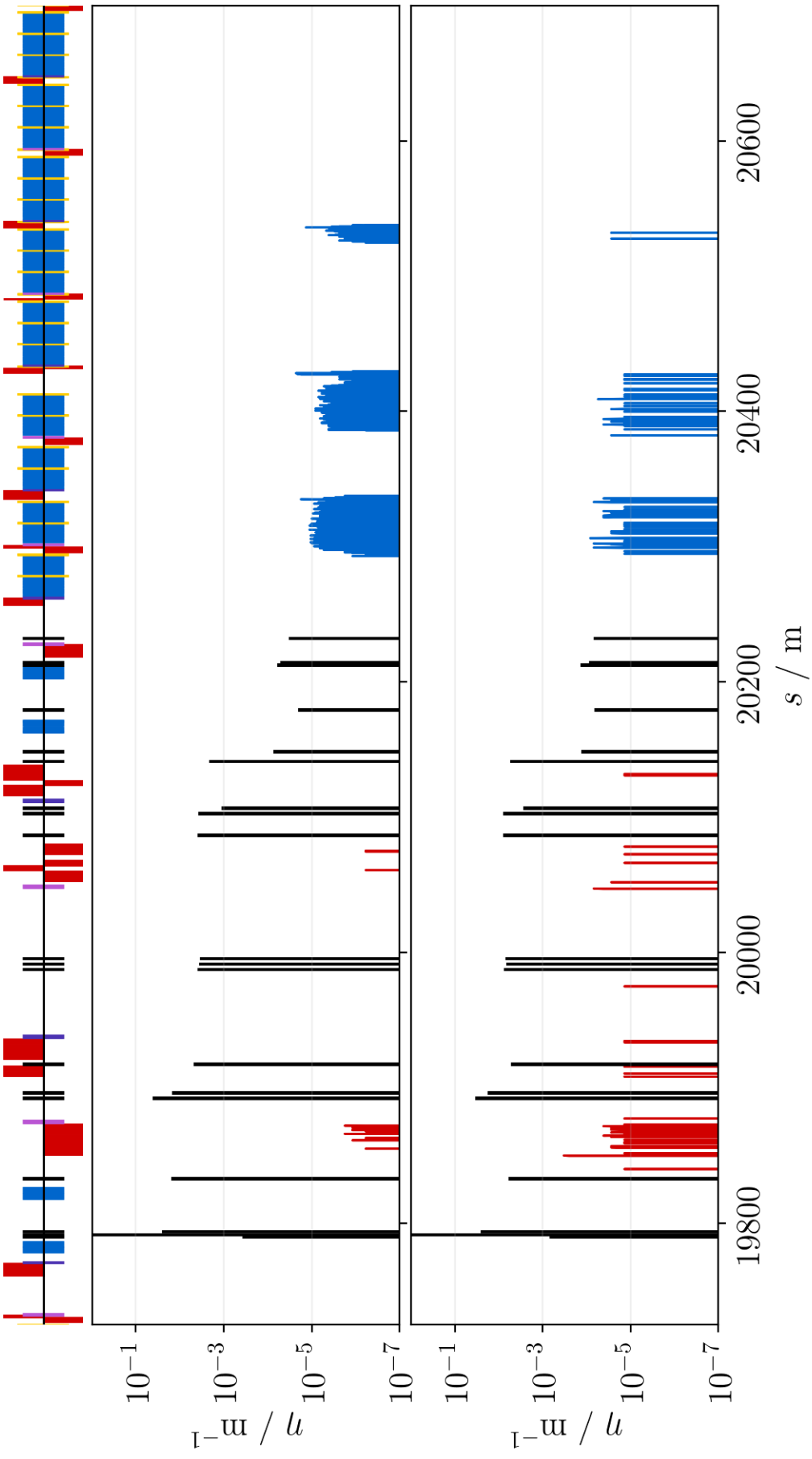


Figure 4.20: Cleaning inefficiencies of SixTrack (top) and BDSIM (bottom) in IR7.

description of the collimators. None of the collimators were aligned with the beam centroids, which is not problematic for most of the collimators, as they are generally centred on $(0, 0)$. The TCTs are the exception as the beam is off-reference near the experimental insertions, which includes the TCTs. Therefore, the TCTs in the BDSIM model were consistently effectively much smaller in their openings on one side. For example, the horizontal TCT in IR8, which is the first TCT after IR7 in the direction of beam 1, has an opening of $15\,\sigma$, which is reduced to $5\,\sigma$ on one side with the missing alignment, which is only slightly more open than the primary collimator at $5\,\sigma$. This is the likeliest source of the large excess in the tertiary collimators, and may also explain the excess aperture losses. Indeed the global cleaning inefficiency (Equation 4.4) in BDSIM is $(7.8 \pm 0.4) \times 10^{-4}$, compared with $(3.01 \pm 0.07) \times 10^{-4}$ in SixTrack, meaning that the collimation system as simulated with BDSIM performed worse than as simulated with SixTrack. Excess aperture losses could simply result from the tertiary collimators in IR8 and IR1 acting almost like primary collimators, only without the subsequent secondary and tertiary collimators to intercept any leakage. Instead, the apertures intercept the leakage and the spikes in the cold losses in IR8 and IR1 shown in Figure 4.17 are the result.

Good agreement is shown between the two codes in the losses in the dispersion suppressor. In SixTrack, one can see loss spikes immediately preceding the end of each stack, corresponding the locations of beam position monitors, where the aperture is much smaller. BDSIM struggles to detect losses where the aperture changes suddenly, because in BDSIM the aperture is simply a sequence of consecutive concentric extruded hollow aperture cross sections, resulting in vertical gaps in the aperture definition. This is demonstrated in Figure 4.19, where an aperture loss can be seen that would fail to be recognised by BDSIM. In SixTrack, however, this would be correctly recorded as a loss, and may explain why loss spikes in regions where the aperture model suddenly tightens, such as in the DS, are absent from the BDSIM loss map, but present in SixTrack. This is unlikely to cause problems in the simulated BLM response or energy deposition, as any proton leaving the aperture in this way will immediately hit a magnet, and result in particle showers.

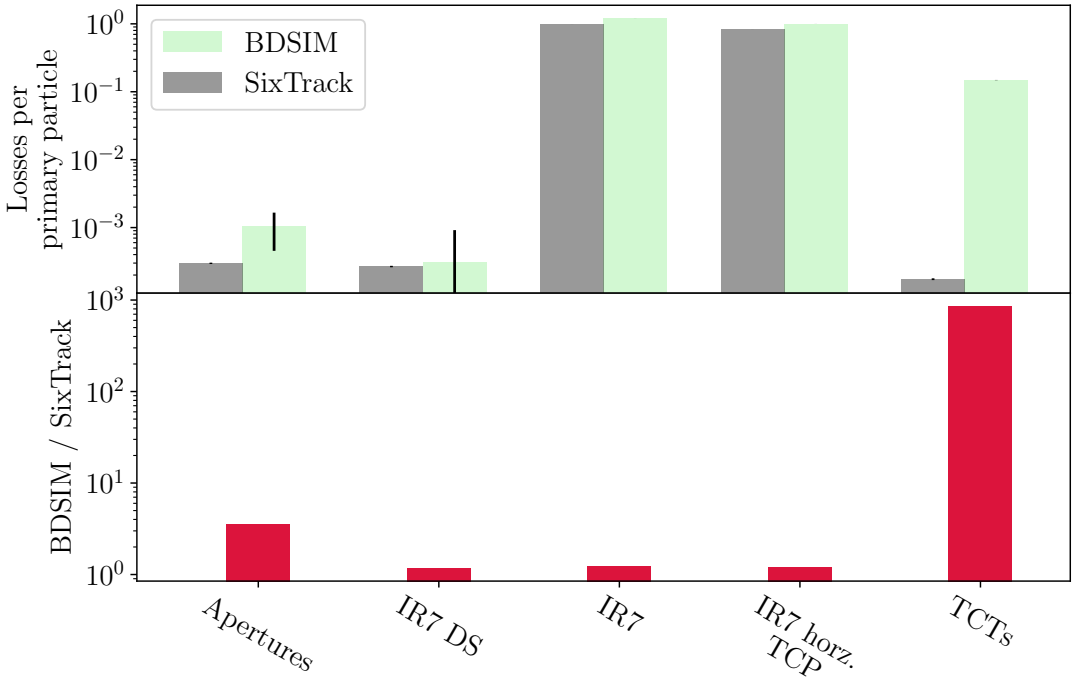


Figure 4.21: BDSIM and SixTrack integrated losses for a range of locations, normalised by total number of primaries simulated.

4.4 BDSIM loss map comparisons with beam loss monitor data

In this section the description, placement and subsequent simulation of BLMs in the BDSIM LHC model is described. The simulated BLM dose is recorded and compared to BLM data from a recent qualification loss map run. The simulation setup details not mentioned here are identical to those of the previous simulations unless stated otherwise.

4.4.1 Beam loss monitors in the BDSIM LHC model

BLMs were added to the LHC model in one-to-one correspondence with those found in the LHC for beam 1. The interior of an LHC BLM is shown in Figure 4.22. Due to a lack of available sources describing the BLM geometry, the BLMs in the BDSIM model are extremely simple. Each BLM is modelled as a cylinder of aluminium, with a length of 25 cm and a radius of 4.5 cm, giving a total active volume of around 1.5 dm^3 [109]. As can be seen in Figure 4.22, the real detectors are not solid aluminium. In addition to the aluminium, they also consist of a volume of nitrogen around 100 mbar

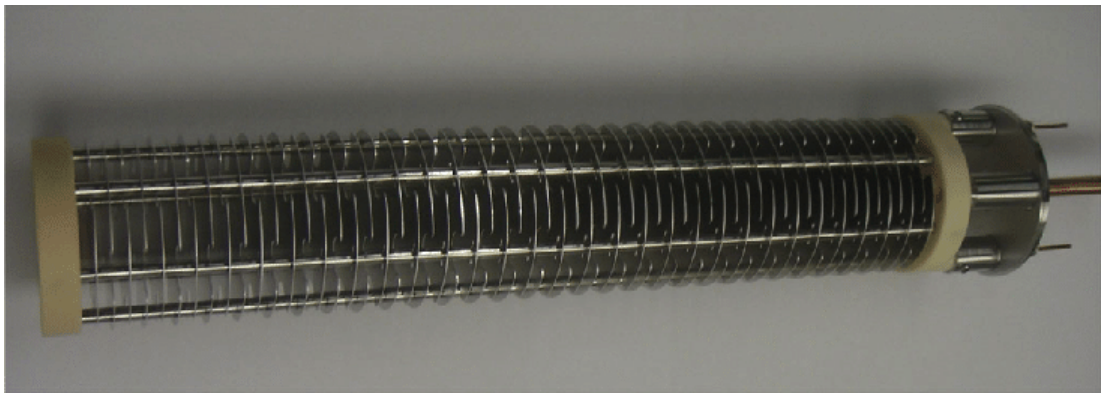


Figure 4.22: The interior of an LHC BLM. Adapted from [77] (CC BY 4.0).

overpressure. Whilst this introduces a source of systematic error, it does at least guarantee that there will be energy deposition in the region. Furthermore, as the BLMs are quite spaced out, cross-talk between BLMs is unlikely and therefore the choice of aluminium is unlikely to make much difference in this regard.

The BLMs were placed according to their expert names, which encodes their positions, their detector type, and other details such as their transverse location [110]. The BLM expert names encode many of the important features pertaining to its type and placement, but the position along the beamline is considered inaccurate and has to be determined with an external database. The BLMs were nominally only attached to quadrupoles and collimators due to their similar sensitivities, as this is standard practice for loss maps, although in reality the external database used for the placements often disagreed sufficiently that BLMs which are nominally attached to quadrupoles may be attached to other components entirely, such as dipoles. BDSIM has a user-friendly interface for placing BLMs, allowing one to place them flush against the side of components, including the capability to introduce a gap between the sides of the component and the BLM—often necessary to remove latent overlaps between the components and the BLM. This naming convention which was inaccurate for the longitudinal placements is perhaps also inaccurate for the transverse positions. It is possible that the collimators are positioned incorrectly as in a previous study [103] the BLMs have been displayed situated beneath the collimators, not at their sides. However, that was a simulation in Run I, and for consistency, the expert names were used throughout for the transverse positions. Using these transverse positions, BLMs were attached flush against the side of the element found at the corrected longitudinal

position. This is one of the reasons why the horizontal extents of the magnets stated in Table 4.1 are important. Furthermore, the BLMs were filtered on their beam number so that, for simulating beam 1, only BLMs associated with beam 1 (B1 in the expert name) were added to the model. The total deposited dose and energies were scored for the whole volume, and these quantities were used to compare with real BLM data, the results of which are presented in Section 4.4.

4.4.2 Qualification loss maps

The simulated BLM data are compared with data from a loss map qualification run which took place on 23rd September 2018. In the qualification loss map run the transverse emittances were deliberately blown up with the use of the transverse damper (ADT) [111] to add white noise to the beam [112]. In effect at each pass of the transverse dampers, the beam particles will receive a random transverse kick. The transverse damper provides fine control of the excitation and can be applied to individual bunches over many turns. This excitation of an individual bunch onto the primary collimator in IR7 over many turns produces a clean loss map with a reasonable impact parameter. This method is in contrast to a previous approach which involved crossing the betatron tune across a third order resonance, in which the whole beam was excited at once.

Dedicated simulations [113] have shown that the use of the ADT to induce beam losses results in impact parameters of up to 10 µm to 20 µm at 7 TeV, which is consistent with the primary distribution used throughout this chapter shown in Figure 4.12b, with an average impact parameter of 4 µm and a maximum of 11 µm. However, considerable variance in the impact parameter was reported in [113] depending on the ADT gain, so this introduced a potentially considerable source of systematic error.

The BLM data were acquired over the few seconds that it took for the ADT to dump the bunch onto the collimator. To subtract any background noise, the BLM signal was recorded for 10 s to 15 s beforehand whilst the ADT was disabled. This background sample was then subtracted from the BLM loss map signal to produce a clean sample of the BLM signal.

4.4.3 Optimal kinetic energy cuts for LHC BLM simulations

Simulating deposition from beam losses in BLMs in the LHC is computationally expensive due to the broad, keV–TeV, energy range involved [109]. A balance must be struck between accuracy and computational efficiency because of the very large number of protons involved (a typical loss map simulation will involve 6.4×10^6 primaries), typically simulated over no less than 200 turns. Furthermore, if one wishes to compare different collimation system configurations or phenomena, the total required number of simulated particles will multiply further, increasing the necessity of fast and efficient simulations.

BDSIM offers two means by which simulation run time and accuracy can be fine-tuned, range cuts and kinetic energy cuts. The former were set to 0.5 m, which is fine enough to prevent any discontinuities in the energy deposition, where convention is that the range cut should be on the order of the length of an average single component. Selecting the optimum kinetic cut is a matter of picking the largest cut which produces the same BLM dose. A smaller model consisting of just IR7 and the following dispersion suppressor was created to test the effect of different kinetic energy cuts on the scored energy deposition in the BLMs. Figure 4.23 shows the BLM scored dose in IR7 for various kinetic energy cuts. Both 1 MeV and 10 MeV appear to be indistinguishable from the scenario without any cut. The 10 MeV cut was chosen so as to speed up simulation time.

4.4.4 Simulated BLM loss maps

The simulated BLM energy deposition across the whole LHC is shown in Figure 4.24, and shown in comparison with BLM data from the qualification loss map described previously. There is a small degree of noise which is present around the ring, but across the various IPs, the BLM dose is comparable in both simulations. However, even at this level it is already clear that the simulated losses in the DS immediately following IR7 are much lower. This can probably be explained firstly in terms of the chosen collimator material, which was erroneously selected as graphite, when it should be CFC, resulting in a density increase of around 50 %. As an excess of the protons are absorbed in primary and secondary collimators, less particle flux should reach downstream to the

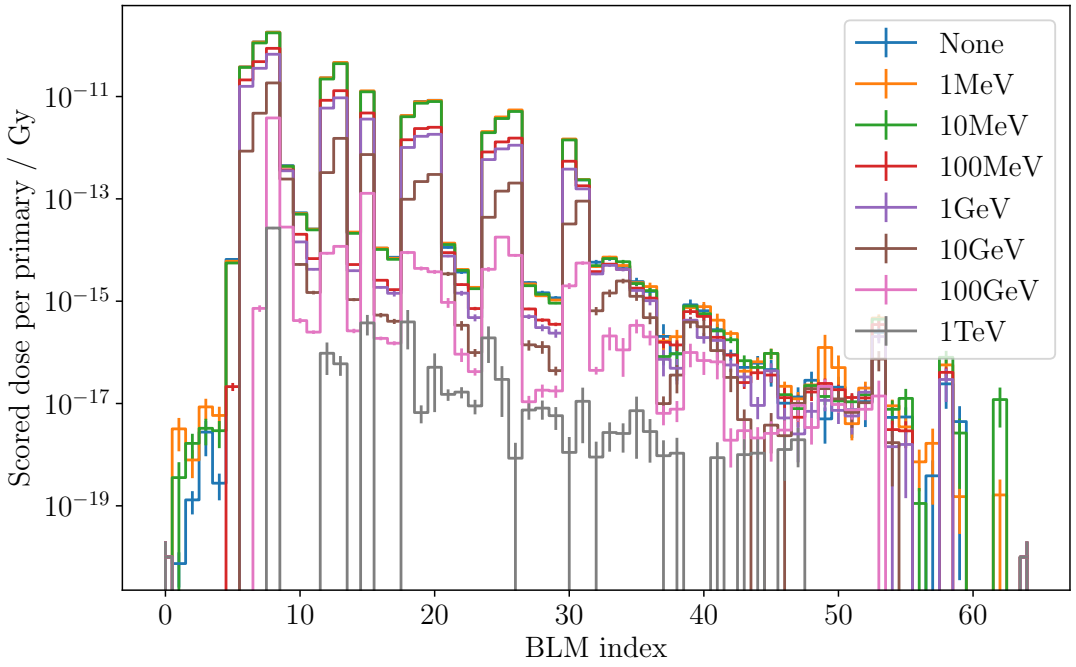


Figure 4.23: Scored dose per event in the BLMs in IR7 and subsequent dispersion suppressor for various kinetic energy cuts.

BLMs in the DS. Also, the large impact parameters due to the tracking will result in greater energy absorbed within the collimators.

The normalised BLMs signals in IR7 and the following DS are shown in Figure 4.25, where one can see the familiar hierarchy in both the data and the simulated BLMs. However, it is also clear that the decrease in the BLM signal falls much faster in BDSIM. This may be occurring for a number of reasons. The most obvious cause is the collimator material, as it is high density, far more secondaries are being absorbed by the primary and secondary collimators. However, even close to the TCPs, one can see a far smaller signal in the BLMs attached to warm components in BDSIM. This is resulting from an incorrect quadrupole geometry, perhaps too large. If it is too large then the BLMs will be further away from the beamline and the signal will be reduced. Alternatively, they could be placed incorrectly altogether, or with the wrong orientation. It is possible the quadrupole geometries are consistently overestimated, as the warm losses in this region are consistently underestimated.

The magnet geometries cannot explain the relative deficit in the DS BLMs, as these magnets are smaller than in reality, because only the yoke is modelled, without the surrounding heat shield and cryostat. The most obvious possible source of difference is

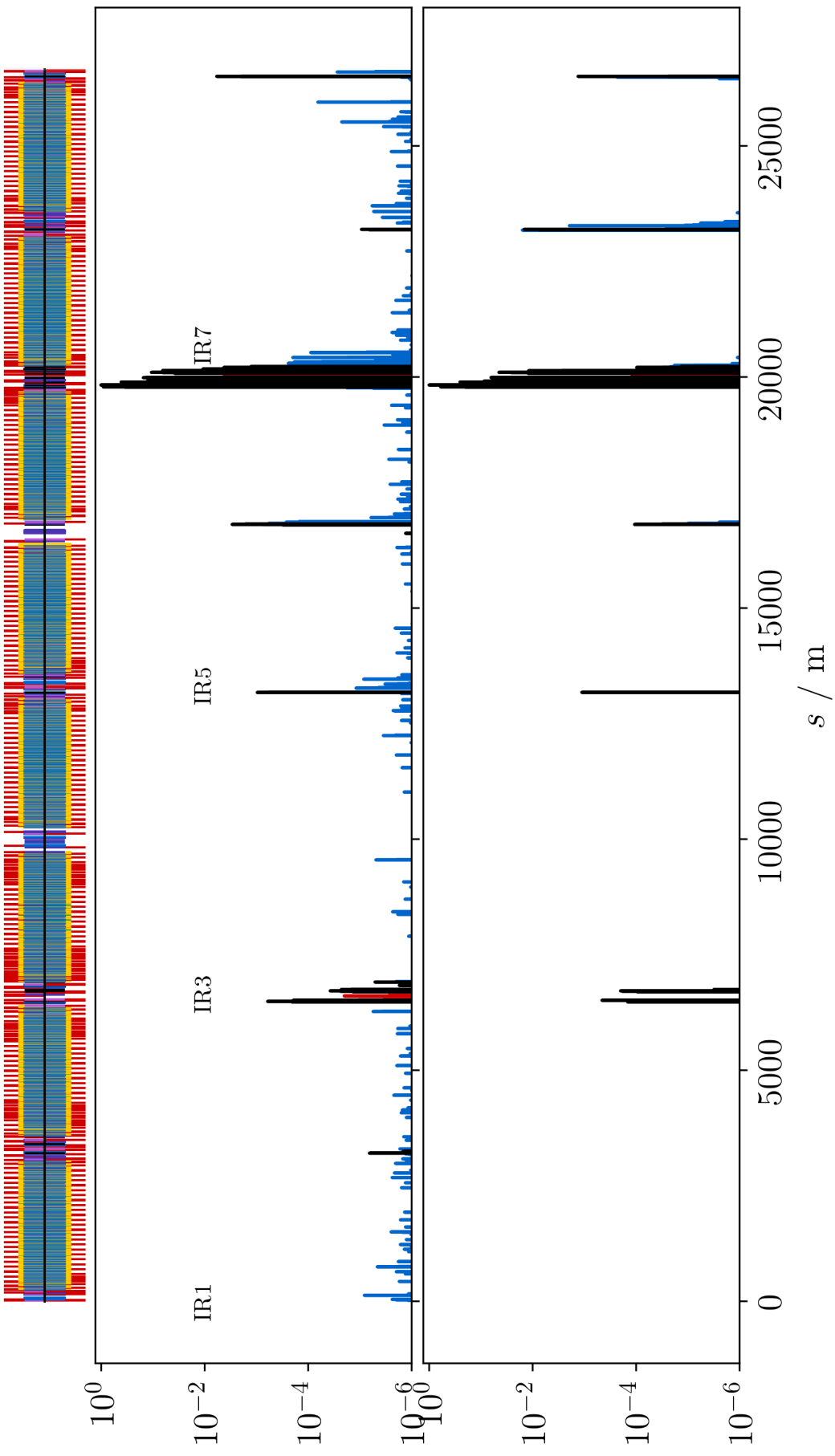


Figure 4.24: Normalised BLM signal from a qualification loss map (top) and simulated BLM energy deposition from BDSIM (bottom). The BLM signal in both loss maps is normalised with respect to the peak in IR7.

the incorrect choice of collimator material, resulting in more absorptions closer to the TCPs and farther away from DS.

Furthermore, the crude choice of the BLM model will undoubtedly affect the simulated signal. Figure 4.23 showed that the BLM signal in IR7 was not reduced when introducing a kinetic energy cut of up to 10 MeV, but full detector response simulations have shown contributions to the BLM signal from all the way down to the 1 MeV scale and below. A similar study to the one presented here simulated particles down to the 1 MeV level [77]. The fact that this sensitivity has not been reproduced in the BDSIM model suggests that the BLM model is likely to be too simple.

The relationship between the energy deposition in the nearest 0.3 m in BDSIM and the BLM signal is shown in Figure 4.26. There is a clear correlation between the BLM signal and adjacent energy deposition, as expected.

4.4.5 Secondary energy deposition in the LHC

One of the unique features of BDSIM is that, in addition to the familiar tracking routines of accelerator tracking codes, it can also track the full range of secondary particles resulting from losing a beam particle anywhere in the machine. In contrast, SixTrack only track primaries. In this section, the energy deposition resulting from primaries is compared with that of the subsequent secondaries, and is used to motivate BDSIM's approach to collimation studies.

The total energy spectrum resulting from all secondary products over many turns exiting the horizontal TCP in IR7 are shown in Figure 4.27. Whilst the total energy is dominated at large energies by the single-diffractive peak in the protons, the pions, photons, electrons, and positrons also contribute a significant amount of energy flux at high energies. This is especially true when considering that many protons at the single diffractive peak will likely either end up impacting on the TCP on later turns, or on the secondary collimators or elsewhere in the machine, thus contributing to the particle flux in the non-proton secondaries. Another interesting place to examine the flux would be at the final TCLA before the DS begins, as this will contribute to the heating on the cold aperture in this region, contributing to energy deposition from the primaries in this region.

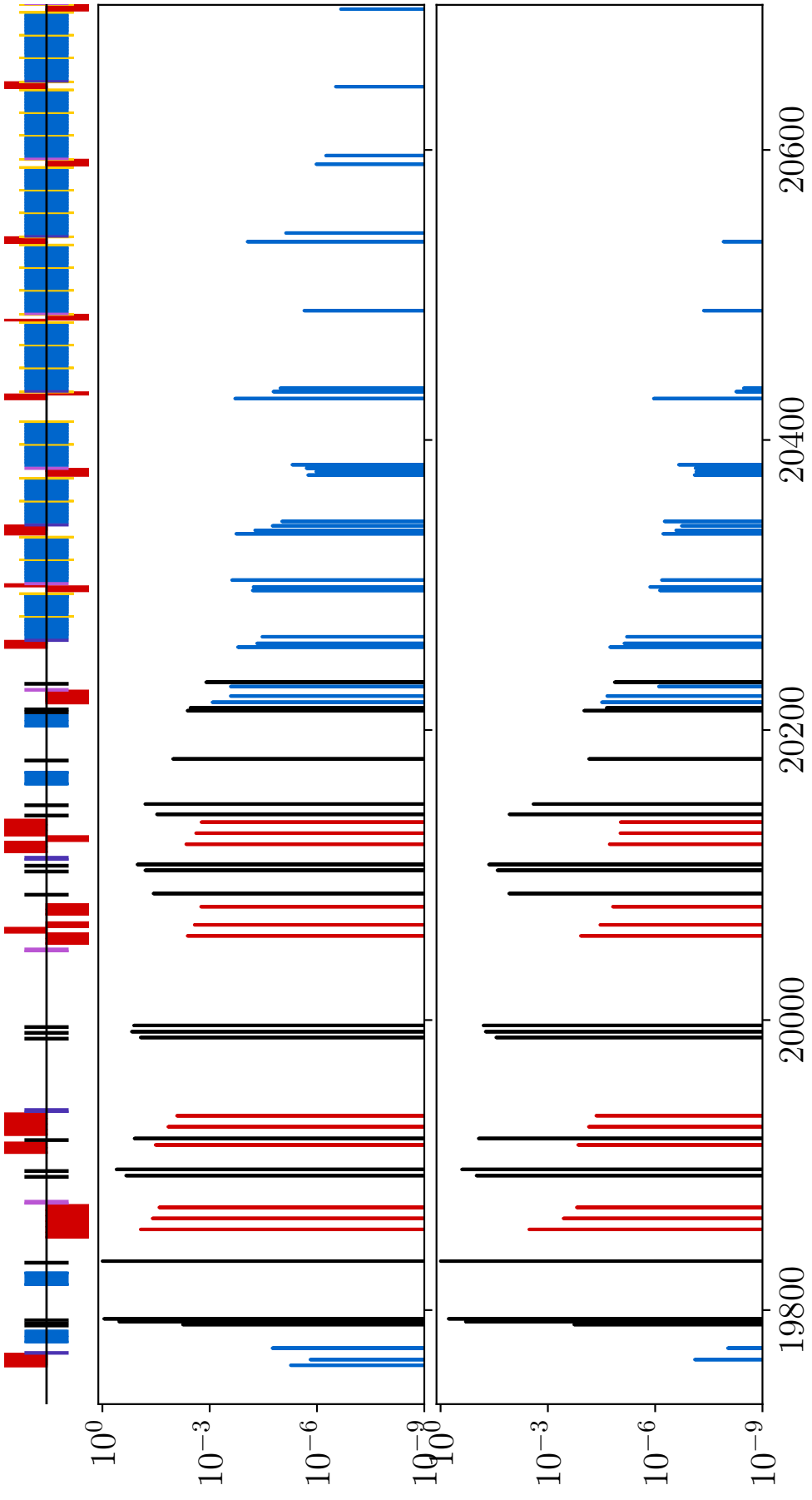


Figure 4.25: Normalised BLM signals across IR7 from a qualification loss map (top) and simulated BLM energy deposition from BDSIM (bottom). The BLM signal in both loss maps is normalised with respect to the peak in IR7.

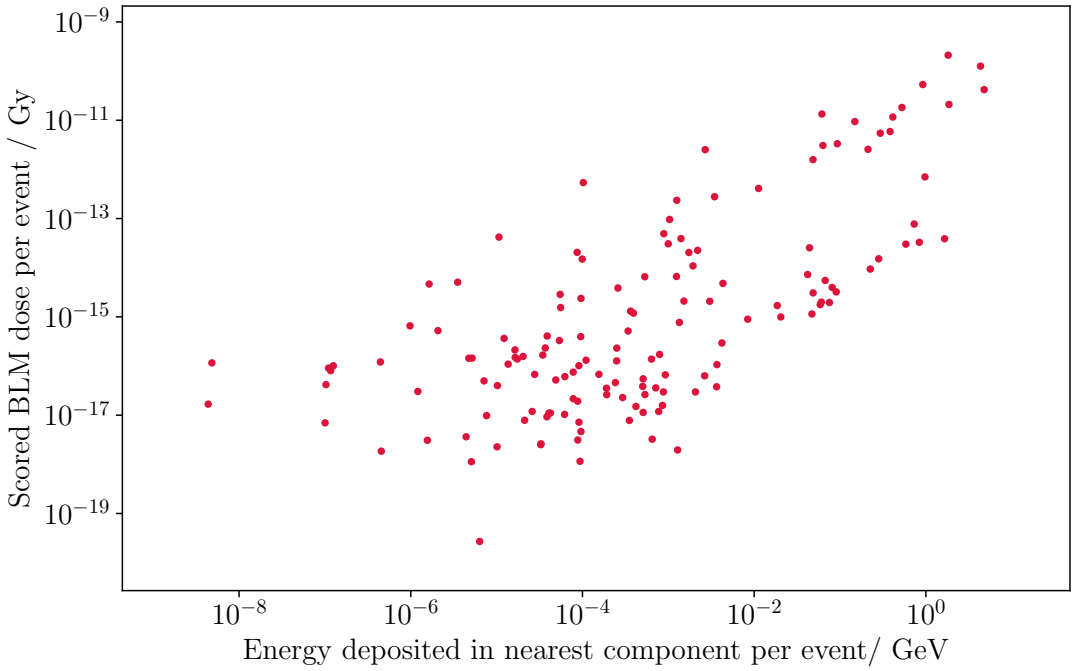


Figure 4.26: Simulated BLM energy deposition with energy deposition of nearest component. The simulated BLM and energy deposition slice in 0.3 m are normalised with respect to the peak loss in the ring.

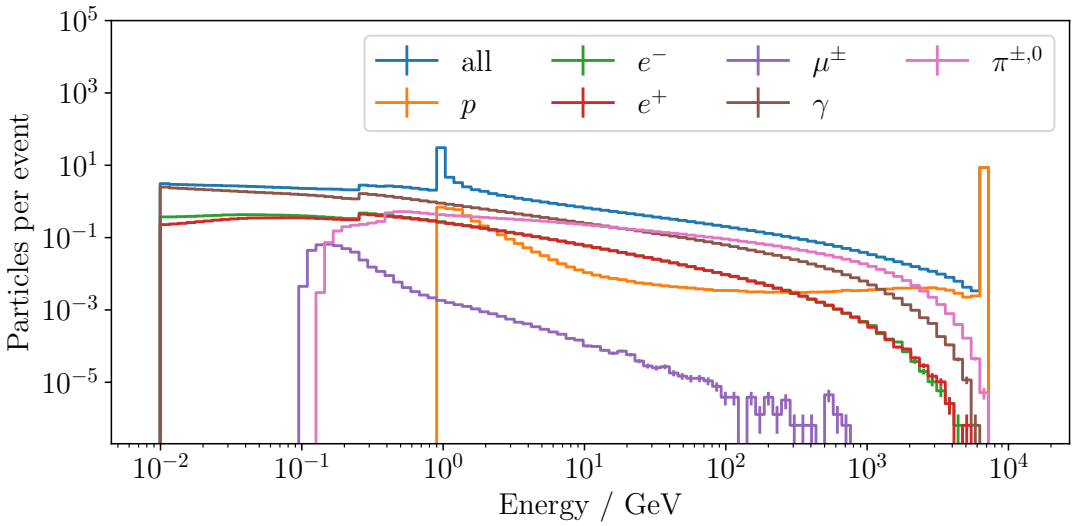


Figure 4.27: Particle energy spectrum out of the horizontal primary collimator in IR7. The particles energies are scored over multiple turns.

The influence of secondaries on the energy deposition can be assessed by comparing aperture hits with energy deposition from all secondary particles. This comparison is shown in Figure 4.28. The energy deposition distribution is clearly much more feature-rich. For example, there are hot spots in the energy deposition in the warm regions throughout, which is a feature which is purely a product of secondary energy deposition. Vitally, a loss spike can be seen in the cold quadrupole just before the last TCLA, which, again, is purely due to secondary energy deposition. Furthermore, the cold losses in the DS are far broader with the inclusion of secondary energy deposition, both due to local losses in the DS, and also from secondary particle showers from upstream in IR7. Furthermore the range of losses is far greater, spanning six orders of magnitude in the impacts, but nine in the energy deposition. This is due to both the increased smearing, and hot spots in the energy deposition.

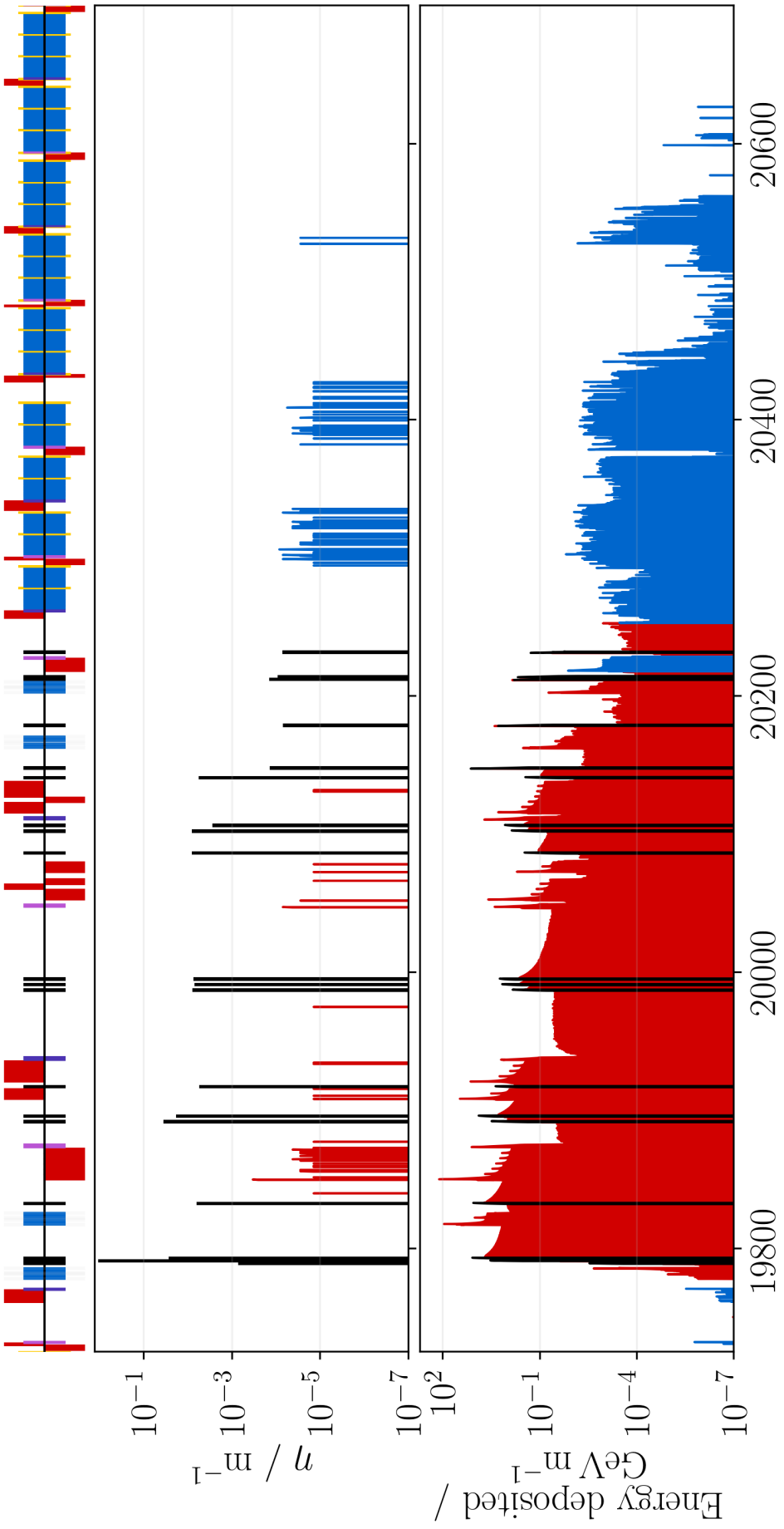


Figure 4.28: Primary (top) and secondary (bottom) energy deposition in IR7 using BDSIM.

4.5 Discussion and future work

In this chapter, the application of BDSIM to LHC collimation simulations has been studied. BDSIM's tracking was evaluated and it was shown to be non-symplectic. Over 200 turns for an LHC collimation study, this was shown to result in an increase in the mean impact parameter from around 4 µm all the way up to 80 µm, which is a considerable source of growth. This was mitigated in two ways, firstly, the primary distribution was started directly in front of the collimator jaw, so as to minimise the total amount of distance traversed, and therefore the accrued error. Secondly, since most of the particles are lost on the first few turns, the error is reduced in this way. The source of the error in the tracking was discussed and explained. It is apparent that this error cannot be simply fixed, as it is fundamental to the way in which Geant4 performs its tracking. The use of a OTM to correct the particle back onto its symplectic trajectory was demonstrated to great effect, completely removing the emittance growth. However, this was not applied to the subsequent loss map simulations, due to the difficulty in BDSIM in determining whether or not the particle underwent an interaction on the previous turn (in which case the OTM should not be applied). As the impact of non-symplectic tracking is more intuitively understood, it was deemed preferable in this case to a less intuitive and less predictable error.
Loss maps were generated in SixTrack and BDSIM and compared. Good agreement was shown between the two codes, with similar losses in the downstream dispersion suppressor. The good agreement suggests that the non-symplectic tracking is not a major problem, at least for a simulation of this type where the beam protons impact directly on the collimator jaws in the LHC. This is consistent with the ideas that whilst non-symplectic tracking will result in an increase in the impact parameter, the impact parameter also has little effect on the simulated cleaning efficiencies, which has been shown in [103]. Regardless, as the emittance growth in y is about half of that in x, simulating the vertical halo should be more accurate, and is one of the next steps to be taken. The global cleaning inefficiency was simulated in BDSIM to be $(7.8 \pm 0.4) \times 10^{-4}$, around 2.5 times greater (i.e. worse performance) than SixTrack at $(3.01 \pm 0.07) \times 10^{-4}$. This particularly manifested itself as excess losses in the IR8 and IR1 and likely caused by misaligned TCTs. This has been corrected for future studies.

Whilst the non-symplectic tracking does not appear to be an issue in the scenario simulated and presented in this chapter, that does not mean it will not be a problem in general. As already stated, the main reason the inaccurate tracking does not seem to matter too much is that the collimation system performance is largely independent of the impact parameter. This is not true of all accelerators, and not even always true of the LHC. Ion collimation has shown a sensitive dependence on the impact parameter [114], and this is also true of ion collimation in the LHC due to ion fragmentation. The choice of impact parameter for LHC ion collimation studies was shown in [115] to have a large effect, when comparing parameters of 1 µm, 3 µm and 10 µm. Each of the three scenarios shown both have unique qualitative and quantitative profiles across the whole of the collider and particularly in the DS. This range is comparable to the emittance growth between the first impacts and subsequent impacts (around once every 8 tunes for a non-integer tune of 0.31) in the LHC model shown above. Whilst this is clearly an issue for heavy-ion colliders, it has also been shown to be true of proton colliders elsewhere. The Rapid Cycling Synchrotron at J-PARC is a proton accelerator that has also been shown to exhibit a nonlinear relationship between the collimation efficiency and the impact parameter [116]. Furthermore, the impact of the tracking was mitigated by starting the proton directly in front of the collimator, but if this is not possible then the tracking will become much more relevant.

For the above reasons it will likely be necessary in the future to implement a separate symplectic tracker. The key features necessary for such a tracker are symplecticity and its speed whilst otherwise retaining feature-parity with the conventional pure Geant4 BDSIM. As it takes around 1 s to track a proton around one turn of the LHC, this could be sped up by a factor of around a thousand if it were to be comparable in speed to SixTrack. The beam-collimator physics could be applied by stepping out of the tracker and passing particles through isolated collimators, before possible reentry into the tracker, or permanent insertion into the main BDSIM model, if the resulting particles are sufficiently off-momentum. A possible design was described and outlined, and work towards implementing this design is ongoing.

Whilst the use of a tracker would speed BDSIM up for LHC collimation studies, the bulk of the time is still spent tracking the secondary shower through the Geant4 model.

This could be further improved by tracking through the BDSIM model and any resulting secondaries only in certain regions of interest, e.g., IR7 and the dispersion suppressor, and then passing back to the tracker. Furthermore, proton losses outside of this region could simply not be inserted into the main Geant4 model to begin with. Further fine-tuning of the kinetic energy and range cut on a per-particle basis would then be necessary to increase the speed of the simulation whilst retaining physical accuracy (e.g. a single beam halo event takes 50 s to simulate with a kinetic energy cut of 1 MeV, but only 30 s with a cut of 10 MeV). BDSIM will need to be able to simulate the LHC model faster if different scenarios are to be compared, and this would open up the possibility of additional studies. The production of a single 6.4×10^6 sample takes around 2 weeks, which is sufficiently long that time saved on the level of a few tens of percent will translate to several days saved. Another benefit of implementing a tracker is that it would allow one to study the impact of nonlinearities on the beam halo, an effect which would otherwise be difficult to model due to the emittance growth. Furthermore, off-momentum studies would be possible with a tracker, as implementing 6D beam dynamics is difficult in Geant4. The implementation of this tracker would be one of the next steps required for more extensive applications of BDSIM to LHC collimation studies.

Furthermore, the particle-collimator physics of BDSIM was shown to produce tails much broader than in SixTrack. As SixTrack has the capability to use Geant4 for its collimator physics, it may be useful in the future to compare BDSIM with this version of SixTrack.

The detailed BDSIM model of the LHC and how it was built was described in detail. This included using the SixTrack aperture model, as well as a set of magnet geometries with at the very least a representative transverse size. BLMs were attached to the components in one-to-one correspondence with those found in the real machine, and the energy deposited was compared with the recorded dose from a recent qualification loss map. Good agreement with the loss map in the collimator hierarchy of IR7 was seen, but the BLMs in the warm and cold sections were systematically underestimated. In the warm sections this may be due to an overestimation of the magnet sizes, resulting in a larger distance from the beampipe and a smaller signal. However, the

signals are also underestimated in the DS, where the magnets are definitely too small, which should result in a larger signal. As the signal remains too small even in the DS, the problem likely lies elsewhere. One possibility is that as a slightly too-dense primary and secondary collimator material was used, far more of the showers are absorbed far away from the DS, resulting in smaller signals downstream. Lastly, the BLM model used is highly simplified but already demonstrates a strong correlation with measured data and simulated energy deposition. Improving this model would be a natural route to reducing the differences observed between simulation and measurements.

Future work on improving the model could involve using the FLUKA-GDML converter introduced in Chapter 3 to convert the existing highly detailed IR7 components, to GDML for use in BDSIM. These components have been built up over many years by tens of individuals, meaning that it is unlikely otherwise for the BDSIM model to have a comparable level of detail, however the simulations compare well to measured BLM signals and the differences were understood. As Geant4 is first and foremost a software library for simulating particle detectors, leveraging the full capabilities of Geant4 to this end would be a natural extension to the work presented here.

In summary this chapter has laid the foundation for further LHC collimation studies using BDSIM. A number of pitfalls in preparing a model of the LHC in BDSIM and in its subsequent simulation have been identified and will be entirely absent in future studies. Even though as it stands the non-symplectic tracking was not found to be detrimental, the proposed development of a tracking library outside the Geant4 model would greatly improve computational efficiency whilst maintaining the unique energy deposition and radiation transport aspects of the BDSIM simulation. In the short term the use of a OTM for collimation studies should be further explored as a means to reduce the accrued error per turn. The range and kinetic energy cuts will need to be fine-tuned to increase the simulation speed without effecting the simulated BLM dose. An approximate BLM model was used, however and strong correlation with the simulated was with the measured signals was shown. It is expected that a more detailed BLM model including improved geometry and a field map would further improve the agreement.

CHAPTER

FIVE

SIMULATION OF BEAM-INDUCED BACKGROUNDS IN ATLAS

5.1 Introduction

Beam-induced backgrounds (BIB) in the Large Hadron Collider (LHC) are backgrounds which result from the loss of protons within the ring. Beam-induced backgrounds offer a unique challenge in the ATLAS detector in that whilst their rates are very low in comparison with backgrounds from *pp* collisions, their rates are comparable to those of rarer events, and their unique characteristics can cause difficulties during the analysis of collected data. The main consideration regarding BIB is that as the protons are lost upstream of the interaction point (IP), particles originating from these events will enter the detectors and travel longitudinally through them. In general the BIB particles will not be uniform in distribution either azimuthally or radially and high-energy muons with this asymmetry are capable of penetrating the shielding before spontaneously decaying and depositing large amounts of energy within the calorimeters. These energy deposits that are not azimuthally uniform result in measurements of missing transverse energy, which otherwise might point to novel physics, are in fact spurious signals originating from BIB. There have been a number of physics searches to date reliant on missing transverse energy which were reliant on identifying and eliminating BIB [117]. The inner detector is also adversely affected by BIB. Charged particles travelling from upstream of the IP in the inner detector will increase the occupancy, and give rise to

spurious clusters, both of which negatively impact the ability to accurately reconstruct tracks [118].

With the advent of the High Luminosity Large Hadron Collider (HL-LHC), conditions will become more adverse: greater beam intensity, higher stored energies and the push for greater luminosity all further motivate the study of beam-induced backgrounds. Understanding BIB in the LHC now may be of great use for the operation of the HL-LHC.

For these reasons it is therefore necessary to understand beam-induced backgrounds, their sources, and how they can be mitigated. Three main sources of BIB have been identified: beam-halo, local beam-gas scattering and global beam-gas scattering [10]. Primary protons which leak from the betatron collimation insertion in IR7 may end up scattering through the collimation system before impacting on the tertiary collimators (TCTs) upstream of the experimental IPs. These protons impacting on the TCTs can lead to secondary particle showers which may ultimately reach the detectors downstream, giving rise to BIB. This source is referred to as beam-halo. Another potential source of hits on the tertiary collimators is from elastic beam-gas interactions with residual gas molecules in the beampipe vacuum; this source is referred to as elastic beam-gas. These protons contribute to the background rate by impacting on the TCTs, either by leaking from the main cleaning insertions or even impacting directly on the TCTs [16]. Finally, protons may undergo inelastic interactions with the residual gas molecules within the beampipe, where the resulting secondary particle showers may reach the detector and contribute to the BIB rate. Inelastic beam-gas is a relatively local effect; studies have shown that inelastic beam-gas from only 550 m upstream of the IP contributes to the background rate in ATLAS [17]. Elastic beam-gas, in contrast, is more global; the elastic interactions between beam protons and gas molecules can occur anywhere, although ultimately contribute to hits on the TCT. Dedicated studies have shown that the rate of BIB from the TCTs [119, 120] is on the order of a few percent, and therefore that BIB from local inelastic beam-gas interactions dominate. To study the BIB originating from inelastic beam-gas collisions in particular, there have been a number of so-called pressure bump runs in recent years in the LHC. These are special runs in which the pressure is deliberately raised in regions local to the detectors

to artificially increase the interaction rate and subsequent background rate in the downstream detectors.

In this chapter both inelastic beam-gas and beam-halo events in IR1 are studied using BDSIM, introduced in full in Section 3.4, and the distributions at the interface plane are presented and discussed. The interface plane is an imaginary plane situated at 22.6 m upstream of the IP deemed to be the longitudinal extent of the detector. Additionally, the physics models of BDSIM are compared with those of FLUKA [25, 121] by comparing distributions at the interface plane. There are a number of comparable existing FLUKA simulations with publicly available results [122] which are used to make the comparisons. These simulations are performed for inelastic beam-gas events from the entire region upstream of the IP, the dedicated pressure bump runs, and also beam-halo. First, the BDSIM model of IR1 is introduced and described, then the simulation method is detailed, and finally the particle spectra and spacial distributions at the interface plane are shown for the various simulation scenarios.

5.2 IR1 Model

Previous studies have shown that the contribution to the background rate in ATLAS from inelastic beam-gas interactions does not extend beyond 550 m upstream of the interaction point [17], and as a result the model here consists only of this region. For this reason the BDSIM model is 550 m long, stretching from the IP up until the beginning of the arc. Building the BDSIM model of IR1 starts with building a model with the correct optical description. This is achieved using the suite of supporting software which exists in addition to the main simulation program. One such example is `pybdsim`, which is a Python package used for, among other things, converting lattices from formats used by other accelerator software tools to BDSIM's own GMAD (Geant4 + MAD) language.

MAD-X [34] is the most widely used program to describe LHC accelerator lattices and optical functions, and a wide variety of MAD-X LHC optical configurations are publicly available on the internet [89]. There are a range of different optical schemes for the various parts of the LHC operation cycle, and here collision optics were used. To improve the comparison between the FLUKA and BDSIM models, the same

Table 5.1: Beam parameters at the IP for the IR1 beam-gas simulations for both scenarios, IR1 beam-gas simulations (2015), and pressure bump studies (2016). Note that the actual emittance, rather than the nominal, was used.

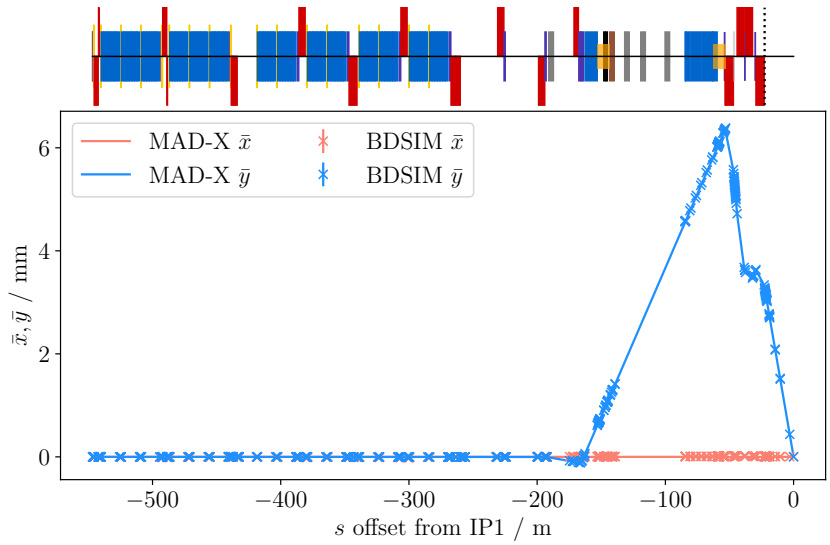
Parameter	2015	2016	Unit
E	6.5	6.5	TeV
$\beta_{x,y}$	0.8	0.4	m
x'	0	0	µrad
y'	145	185	µrad
x, y	0	0	mm
TCT opening	13.7	9.0	σ
ε_N	2.5	2.5	mm · mrad

configurations were used in BDSIM as in FLUKA for the different simulation scenarios. The 2015 collision optics were used for IR1 beam-gas and beam-halo, and 2016 collision optics were used for the pressure bump studies. The pertinent parameters for the two scenarios are shown in Table 5.1.

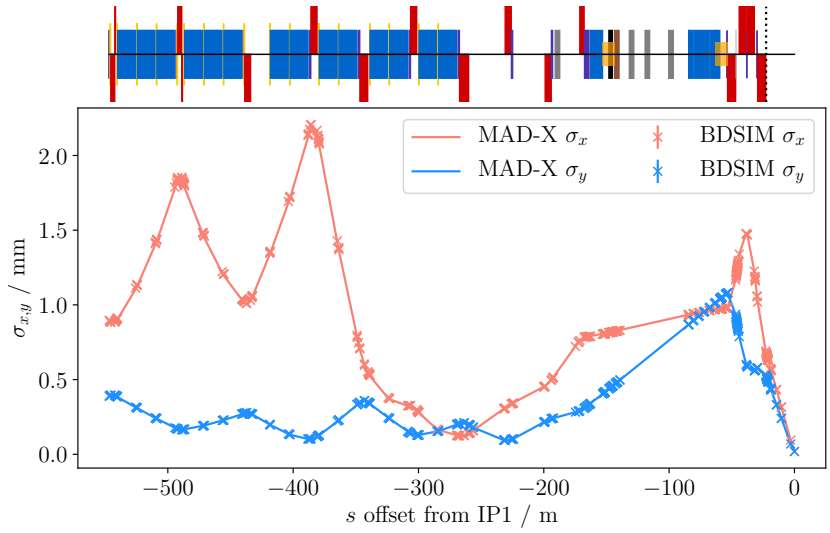
Whilst both BDSIM and `pybdsim` are mature and robust tools, it is worthwhile to check the optical agreement of the converted BDSIM lattice with the source MAD-X description to ensure the preparation of the model is correct. The two most relevant values for this study are the horizontal and vertical beam centroids $\bar{x}, \bar{y}$, and the horizontal and vertical beam sizes, σ_x, σ_y, which are shown in Figures 5.1 and 5.2 for the 2015 and 2016 collision optics used in the simulations.

A correct optical description is necessary for a realistic simulation, but to ensure the rate of flux seen at the interface plane is accurate, it is also necessary to consider the physical geometries and materials of the accelerator components, shielding, and other components. As the description from MAD-X includes no geometric information besides component lengths and angles, `pybdsim` will generate a BDSIM model without any of this information, and fall back on default apertures, magnet geometries and collimator openings throughout. These defaults will in general be inaccurate for the purpose of the IR1 simulation because in the default conversion the relevant information is completely absent. This extra information must be folded in during the conversion process: an accurate aperture model was added, magnet component geometries were substituted and additional pieces of geometry were placed around the beamline.

A BDSIM model's geometry may be improved in two key ways: firstly, there are a

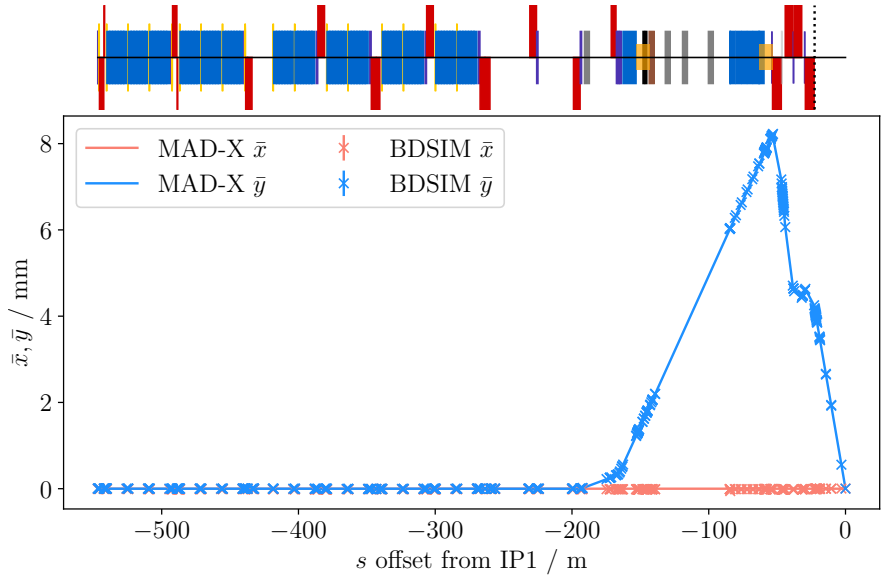


(a) Beam centroids.

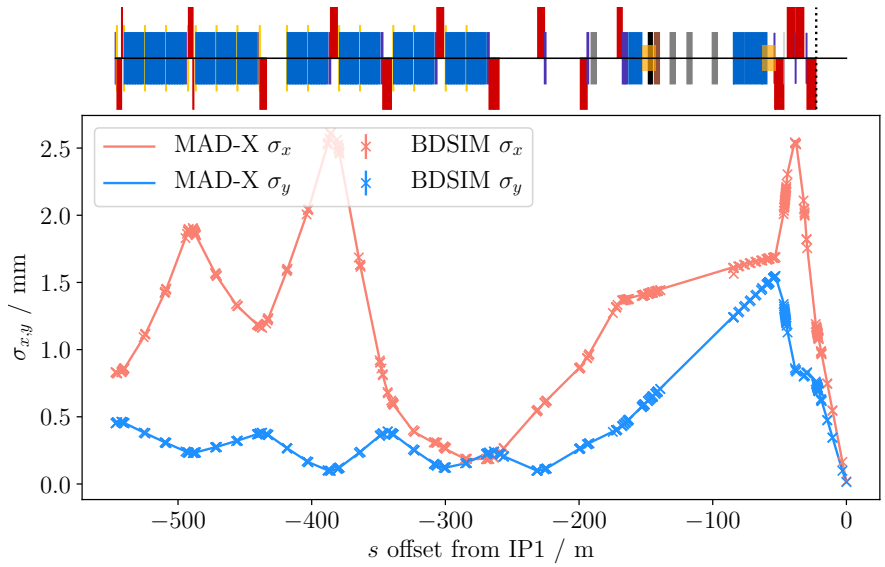


(b) Beam sizes.

Figure 5.1: Optical comparison between MAD-X and BDSIM (ten thousand primaries) for the 2015 collision optics IR1 beam-gas model.



(a) Beam centroids.



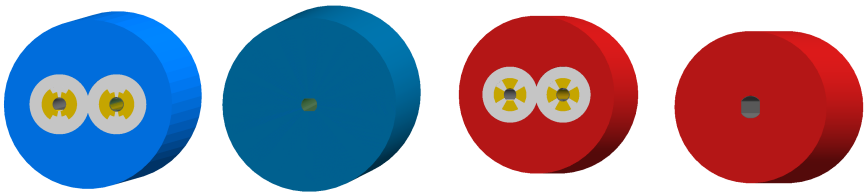
(b) Beam sizes.

Figure 5.2: Optical comparison between MAD-X and BDSIM (ten thousand primaries) for the 2016 collision optics pressure bump model.

number of predefined BDSIM magnet and beampipe geometries from which the more detailed LHC model may be built. Secondly, the user can provide arbitrary external geometries in the Geometry Description Markup Language (GDML) format [68]. Improving the IR1 model in these ways consists of picking many of what are perceived to be high-yield geometric features combined with the Python geometry package `pyg4ometry`, discussed in detail in Section 3.5.

The IR1 magnet geometries were made more realistic as follows. BDSIM includes the cross section of an LHC main bend dipole, and since BDSIM builds the beamline model following only one beam, BDSIM offers two variations of the dual beam pipe geometry. The magnet geometry *lhcright* has an inactive beampipe to the right of the active one, and vice versa for *lhcleft*. Since all the IR1 simulations were of beam two to the right of IR1, the correct choice for the geometry was lhcright. Where the magnets contain single beampipes, the predefined *cylindrical* geometry was used. Both dipole geometries are shown in Figure 5.3a. The blue sections shown in these renderings (referred to as the *outer-geometries*) can also be fine-tuned by setting their sizes and materials. Basic sets of geometry parameters can be applied to families of magnets with the aid of the naming convention used through the LHC. This includes both the outer diameters and materials of the magnets. In reality the magnets are not as homogeneous as shown in Figure 5.3, but setting all of the outer geometry materials to `G4_STAINLESS_STEEL` is a good compromise as it is makes up the bulk of the material by volume in the magnets at the relevant radii.

Many of the magnets in the lattice, particularly the corrector magnets near to the interface plane, are represented as *thin kicks*. Nearer the interface plane, these are mostly used to introduce the crossing angle at the IP, but more generally across the whole LHC are used to account for magnet imperfections. The treatment of the kickers works perfectly for the correct optical transmission of the beam, but means that the thick magnet geometry which would otherwise impede the traversal of secondary particles is completely absent from the model. Nearer to the interface plane these are particularly likely to absorb secondary particles from BIB, meaning it's important to include the geometries of these magnets. One possible approach is to manually edit the lattice to thicken these kickers by absorbing drifts on either side, as BDSIM will then



(a) Dipole geometries used in the IR1 model.

(b) Quadrupole geometries used in the IR1 model.

Figure 5.3: The dipole (quadrupole) geometries used in the IR1 model. The double-aperture *lhcright* dipole (quadrupole) magnet geometry, left, was used for the main bend dipoles (quadrupoles), and the *cylindrical* magnet geometry, right, was used where the beampipes were combined.

automatically include external magnet geometries. A simpler solution is to place cylinders around thin drifts the same length as if they were to be thickened. The advantage of this approach is purely in its simplicity, as BDSIM steps particles through the thin kicker with a transfer matrix. However, if the incoming particle is sufficiently non-paraxial, then the particle will see no kick at all. This is in contrast to the treatment of thick elements, where if the particle is sufficiently non-paraxial then it will be stepped through the component with a numerical integrator. However in this case, the effect is expected to be negligible as the strongest kickers are thick, and many of the others aren't powered at all. Finally, their geometries are `cylindrical` where there is only one beampipe, and the dual aperture `lhcright` where there are two.

Accurately representing the tertiary collimators is important as these are explicitly used to shield the final focus quadrupoles and the experiment from beam-halo particles. To this end the collimators were opened to the correct extent, offset such that the beam centroids are situated in the middle of the opening, and the material was set to tungsten. The outer widths used for both tertiary collimators are 0.25 m and the length of the opening was set to 5 cm, both of which are sufficiently accurate for these simulations.

The aperture model is important as, outside of the collimators, it defines the first point of impact between a particle from the beampipe and the surrounding accelerator geometry. The most detailed LHC aperture model is used in SixTrack collimation simulations, which is interpolated to a precision of 10 cm. In this model all apertures are of type *rectellipse*, which is the intersection of a rectangle with a concentric ellipse. The apertures from SixTrack were written to a text file and used with `pybdsim` to build

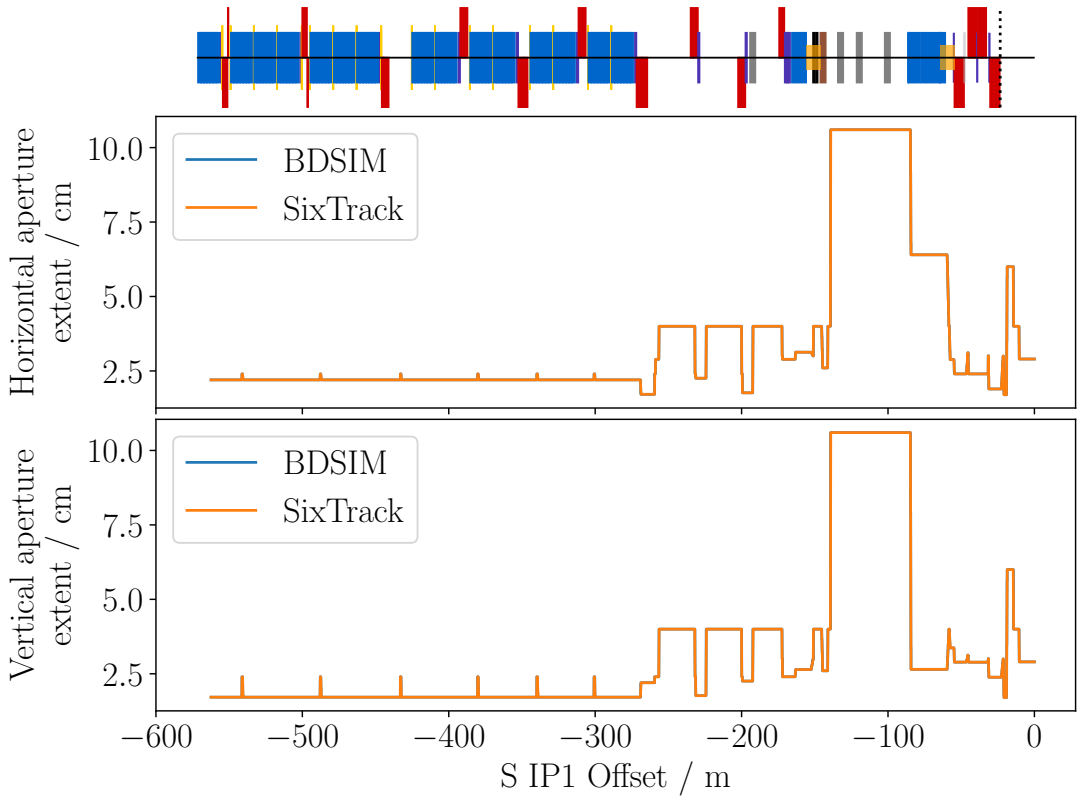


Figure 5.4: IR1 beam two aperture extents compared between SixTrack and BDSIM. The tertiary collimator openings are not included in this figure, but would be the bottleneck if they were.

a BDSIM model, such that BDSIM has the identical 10 cm resolution aperture description, which is shown in Figure 5.4.

Four concrete shielding blocks were placed in the region 200 m upstream of the IP: JSCAB_CN is placed before the tertiary collimators, with the JSCAA1, JSCAA2 and JSCAA3 each placed in between the separation dipoles. One of the JSCAA blocks is shown in Figure 5.5, the JSCAB_CN shielding block is about two metres longer and has a narrower vertical aperture, but is otherwise very similar.

The most important piece of shielding that is otherwise missing, is the target neutral absorber (TAN). The TAN's primary purpose is to protect the superconducting D2 magnet from neutral particles originating from collisions at the IP, however its secondary purpose is to provide forward shielding of the experiments to reduce experimental background [123]. For this reason it is important to include it in the model to generate the correct distributions at the interface plane. The TAN is a complex and critical component in the insertion region and as a result, ideally a

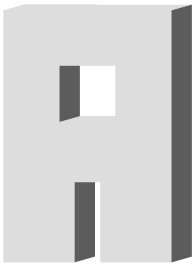
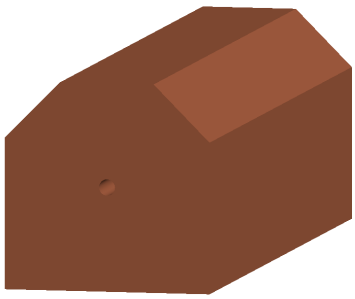
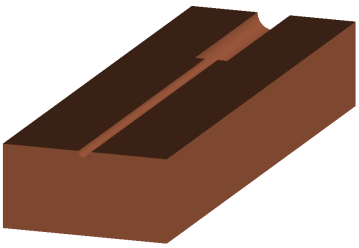


Figure 5.5: JSCA concrete shielding block geometry as rendered in BDSIM. The locations of these blocks is shown in Figure 5.9, with 3D renderings of the beamline in Figure 5.7.



(a) TAN as rendered in BDSIM



(b) TAN as rendered in BDSIM with cutaway plane to demonstrate internal beampipe geometry.

Figure 5.6: The TAN used in the BDSIM model. The TAN has length 3.5 m and cross section 1 m.

detailed model of it would be built. However, acquiring a detailed description of the layout of TAN has proved to be unsuccessful. For the sake of simplicity, a basic TAN model was designed using FLUKA's graphical interface, FLAIR [73], and then translated to GDML with the use of `pyg4ometry`. The TAN geometry is shown in Figure 5.6, where the beampipe radius changes represents the point where the two beampipes combine into one immediately following the separation dipole D2. The BDSIM beamline with the full set of geometric modification, including the external beamline geometries is shown in Figure 5.7.

A detailed FLUKA model of the IR1 tunnel was converted to GDML and placed around the beamline[1]. The tunnel model extends 249.9 m from the interface plane $s_{\mathrm{IP1}} - 22.6\,\mathrm{m}$) which is the end of the long straight section (LSS) and where the arc begins. The diameter of the tunnel is on the order of 4 m, much larger than the enclosed accelerator components. However in the final 5.65 m before the interface plane, in the last quadrupole of the final focus system, Q1, the tunnel closes in on the quadrupoles

[1]Courtesy of the CERN FLUKA team, with contributions from the CERN Radiation Protection team.

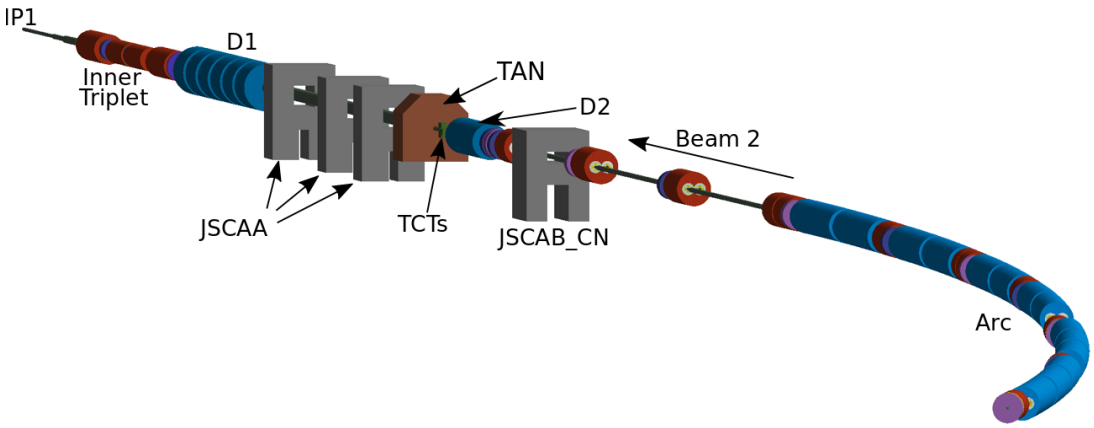
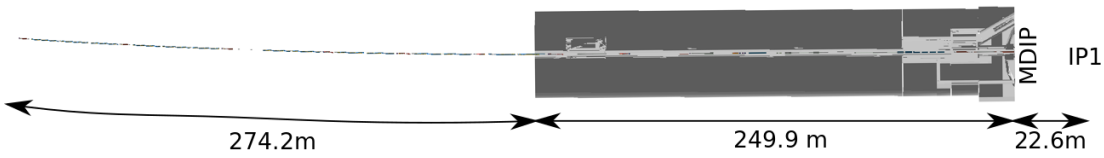
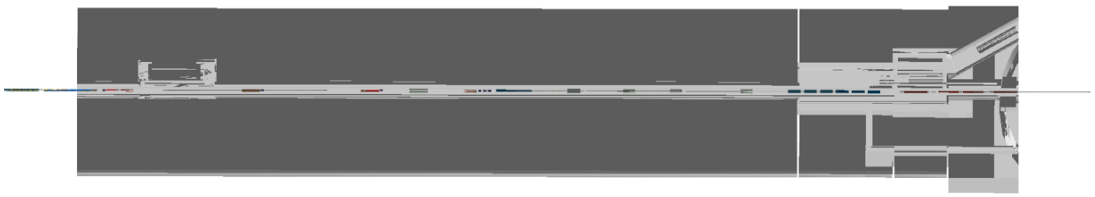


Figure 5.7: Annotated Rendering of the beamline in BDSIM as used for the following simulations. Note that the tunnel is not visible here. IP1 is in the top left hand corner, with the region upstream of the detector extending toward the lower right hand corner.

(which are themselves 50 cm in diameter) to a diameter of 2 m. This means the tunnel geometry is likely to have a sizeable impact on the BIB seen in the inner detector ($r < 2.1$ m) and thus particularly important for getting the correct particle fluxes at the interface plane. The tunnel geometry is converted directly from FLUKA input to GDML using the conversion utility `pyg4ometry` as described in Section 3.6. Whereas some of the other geometries used in the model are generic, this is a very accurate rendering of the tunnel, and by converting it directly these details are preserved without much effort. A cutaway plan view of the tunnel is shown in Figure 5.8.
The existing FLUKA models of IR1 are much more detailed than the BDSIM model presented, making use of an extensive library of components called FLUKA Element Database (FEDB) and LineBuilder to combine them into functioning beamline models [48]. The FLUKA model also uses bespoke field maps for the components, where as BDSIM uses the ideal field in the beampipe and generic fields in the yokes. However, it is hoped that the salient features have been captured and therefore the simulations are an acceptable representation of IR1, giving accurate particle distributions scored at the interface plane for use with beam-induced backgrounds in the ATLAS detector.



(a) Whole BDSIM IR1 beamline with tunnel including annotated key dimensions.



(b) Zoom of IR1 long straight section tunnel.

Figure 5.8: IR1 BDSIM model, featuring both the full beamline long straight section tunnel. Both images feature a cutaway in the y-plane along the beamline axis.

5.3 Simulation Method

The approach for two main simulation scenarios, beam-gas and beam-halo, are presented here. Associated with the beam-gas simulations are two further sub-scenarios: beam-gas collisions along the whole beampipe upstream of the interface plane, and the simulation of beam-gas collisions in two 10 m long regions centred at 58 m and 148 m corresponding to each of the two pressure bumps. Additionally, inelastic beam-gas collisions were simulated from the interface place up to 546.6 m upstream of the IP. Due to the extremely low vacuum pressure and subsequent small rate of beam-gas interactions, directly simulating these collisions requires additional steps to achieve statistically significant results. In the FLUKA simulations, this is achieved by producing daughter products from inelastic collisions using the generator Peanut and inserting these products directly into the FLUKA model uniformly along the beam trajectory [124]. BDSIM on the other hand, does not support such an approach, but does support physics cross section biasing, and this was used to compose a comparable simulation by biasing only the primary proton physics. The scale factor for the cross section of the inelastic proton physics (`protonInelastic` in Geant4) process was chosen such that for a proton traversing a single metre within a biased volume, it would

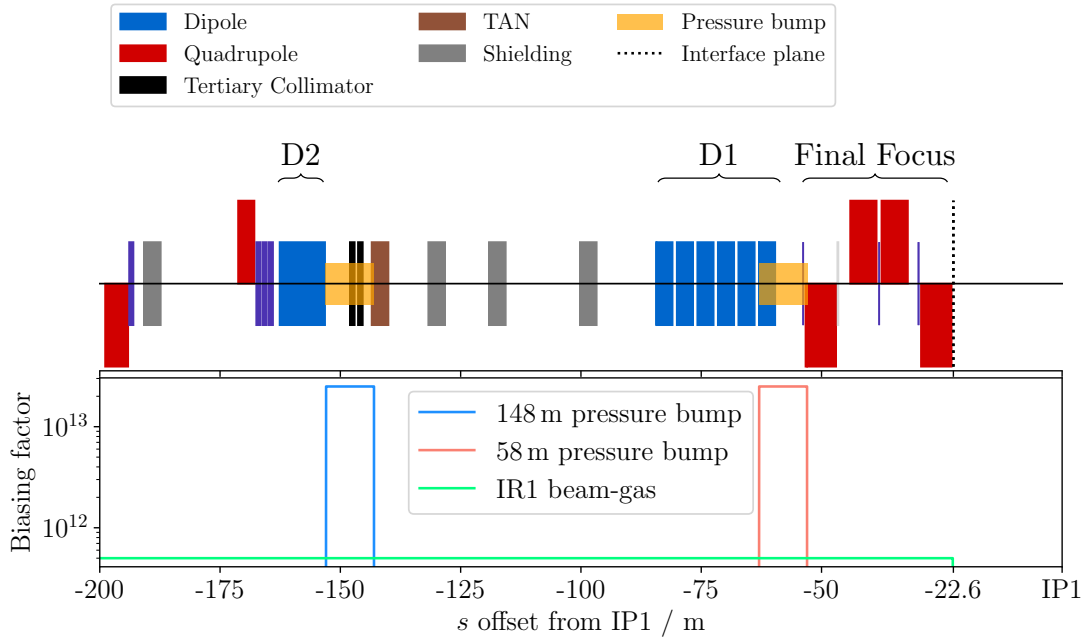


Figure 5.9: The region extending 200 m upstream of IP1, with key features, including the two pressure bumps at 148 m and 58 m noted. The full model extends 562 m upstream of the IP. The `protinInelastic` biases used in BDSIM for the three different scenarios are shown on the lower plot.

undergo an inelastic event approximately 50 % of the time. For a single metre, this value is 2.5×10^{12}, which was then further adjusted based on the total length of the region of interest to be biased to ensure the 50 % reaction rate over that region. This meant that around half of all primaries did not interact, however the simulation time for these events was negligible. The benefit of this approach was that it meant the region sampled in s was relatively flat and therefore was adequately sampled across the entire region in s.

To replicate the uniform sampling along the length of the model of the inelastic collision location, the per-event weights stored by BDSIM are re-scaled such that the inelastic event location in s is uniform across the region of interest and normalised to the total number of inelastic events, thus giving a uniform sample in s as shown in Figure 5.10b. The flat analogue weights, w_i^{analogue}, are given by

$$w_i^{\text{analogue}} = w_i^{\text{biased}} \frac{N_{\text{inelastic}}}{\sum_{w_j^{\text{biased}}} w_j^{\text{biased}}}, \tag{5.1}$$

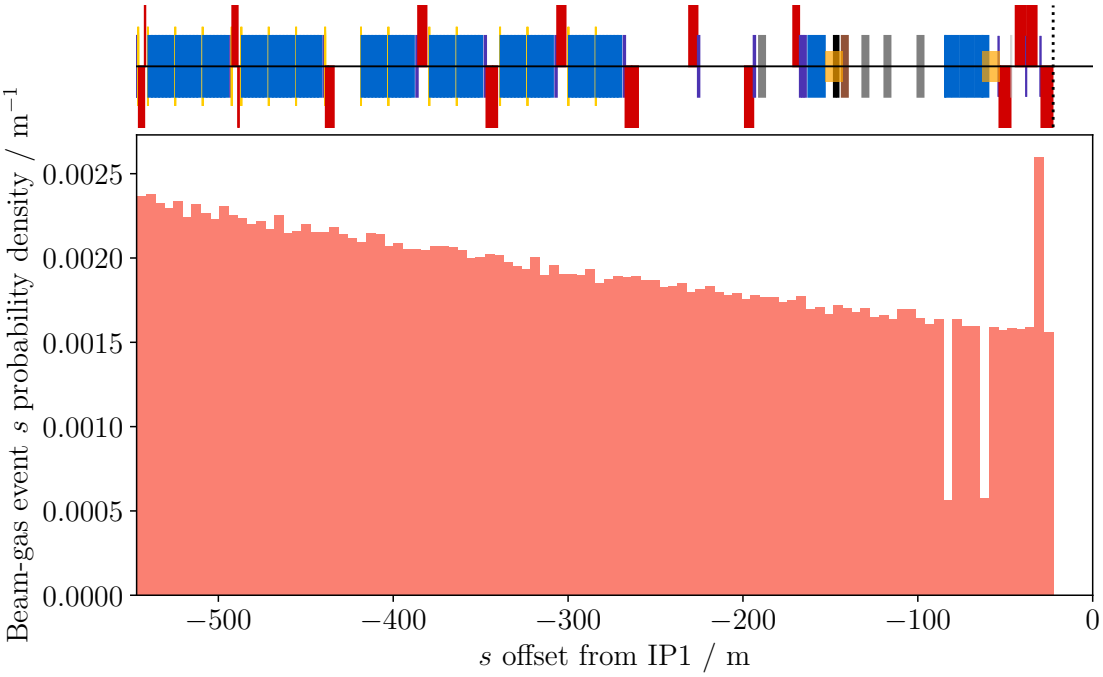
where w^{biased} is the Geant4-provided weight and $N_{\text{inelastic}}$ is the total number of

inelastic events sampled. This equation gives the means by which each Geant4 biased weight can be converted to the analogue (but still flat) weight w^{analogue}. The sum of the analogue flat weights is equal to the total number of particles, which means that this is equivalent to sampling uniformly in s. The selection of weights used is shown in Figure 5.9. The greatest benefit of this approach over inserting daughter products from a generator into the model is that the events are guaranteed to be at realistic positions, amplitudes and angles, i.e. the initial conditions are optically correct.

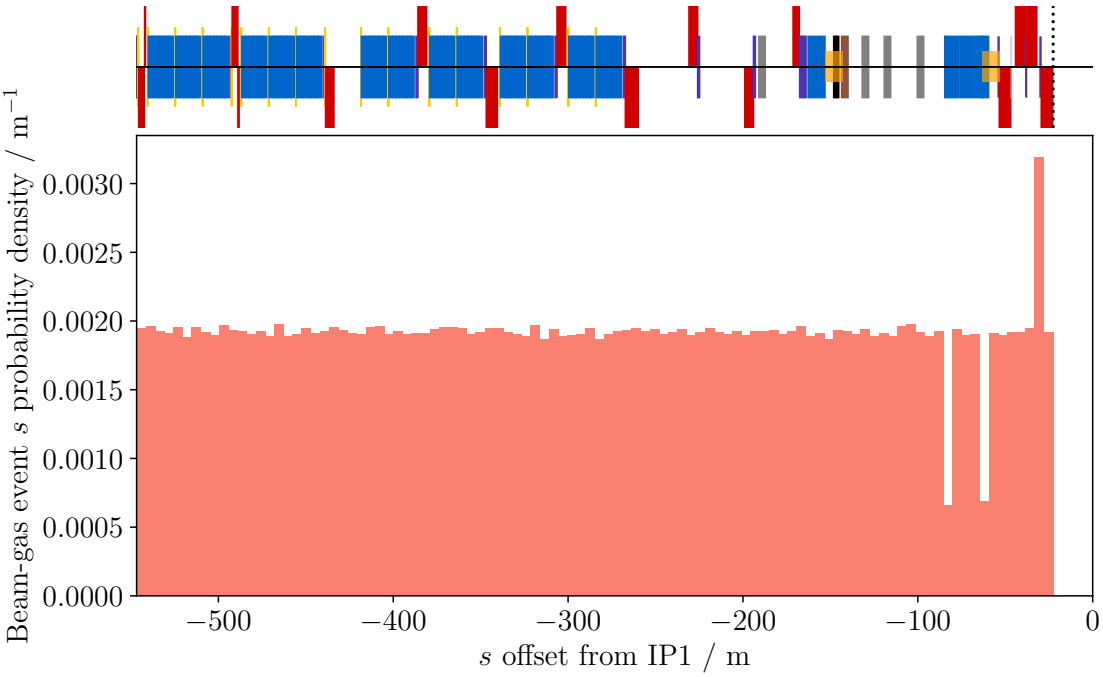
It is important to note two features in Figure 5.10: the two reduced bins in D2 and the spike in the inner triplet. Both features are due to bugs in BDSIM that were only exposed after the lengthy simulations were finished. The empty D2 bins were a result of the biasing not being set correctly and must be accepted as a source of systematic error. The spike in the inner triplet is as yet unexplained but as the event excess occurs at a single point in s all the way down to machine precision, it is most certainly a bug as well. This is corrected by simply removing the events at this point s from the sample.

The simulations were further divided into two sub-scenarios with different kinetic energy cuts, either 20 MeV or 20 GeV. These cuts were used to optimise the study of the respective energy ranges. The 20 MeV cut is used because particles below this are too low energy to reach the subdetectors in ATLAS and the 20 GeV cut is used to optimise the production of high energy muons.

In all scenarios the protons are tracked through the vacuum where the `protonInelastic` cross section is biased and an inelastic collision may occur. Any particles resulting from the collision are propagated through the model until their kinetic energy drops below the kinetic energy cut, or they escape the model at which point they are likely to be killed according to range cuts, which is another, Geant4-specific means of controlling simulation time. Range cuts are discussed in more detail in Section 3.4, but in short they determine whether a secondary particle should be produced in an event, or its energy should simply be considered deposited at that point. If the proposed secondary particle would not travel beyond the distance specified by the range cut within the volume it finds itself in, then it is not created, otherwise it is. In these simulations, however, the range cuts were left at 1 mm, their default values. These are far lower than the minimum kinetic energy cuts, and therefore have no



(a) Prior to re-weighting.



(b) After re-weighting.

Figure 5.10: Distribution in beam-gas event position s. The events are weighted such that the distribution is flat to replicate FLUKA's uniform sampling in s.

Table 5.2: Total inelastic collisions simulated for each scenario in comparison with the corresponding FLUKA simulations.

Scenario	E_K cut	FLUKA events ($\times 10^6$)	BDSIM events ($\times 10^6$)
148 m pressure bump	20 GeV	195.9	267.0
148 m pressure bump	20 MeV	5.0	3.5
58 m pressure bump	20 GeV	99.0	168.9
58 m pressure bump	20 MeV	1.3	1.9
IR1 beam-gas	20 GeV	181.7	167.4
IR1 beam-gas	20 MeV	3.2	2.1
IR1 beam-gas, no crossing angle	20 GeV	N/A	125.0
IR1 beam-halo	20 GeV	295.3)	320.5

impact on the running of the simulation.

Either way, a square scoring plane (`sampler` in BDSIM terms) with sides of 48 m, which is large enough to cover the entire detector, is placed at the interface plane and all particles are scored. Each particle can be associated with the parent primary proton, which if it interacted with the vacuum, has a corresponding position in s.

The complete set of simulations including beam-gas, pressure bump and beam-halo with associated kinetic energy cuts are shown in Table 5.2, where it can be seen that comparable statistics were generated in comparison with the existing FLUKA simulations.

5.4 Results from beam-gas simulations

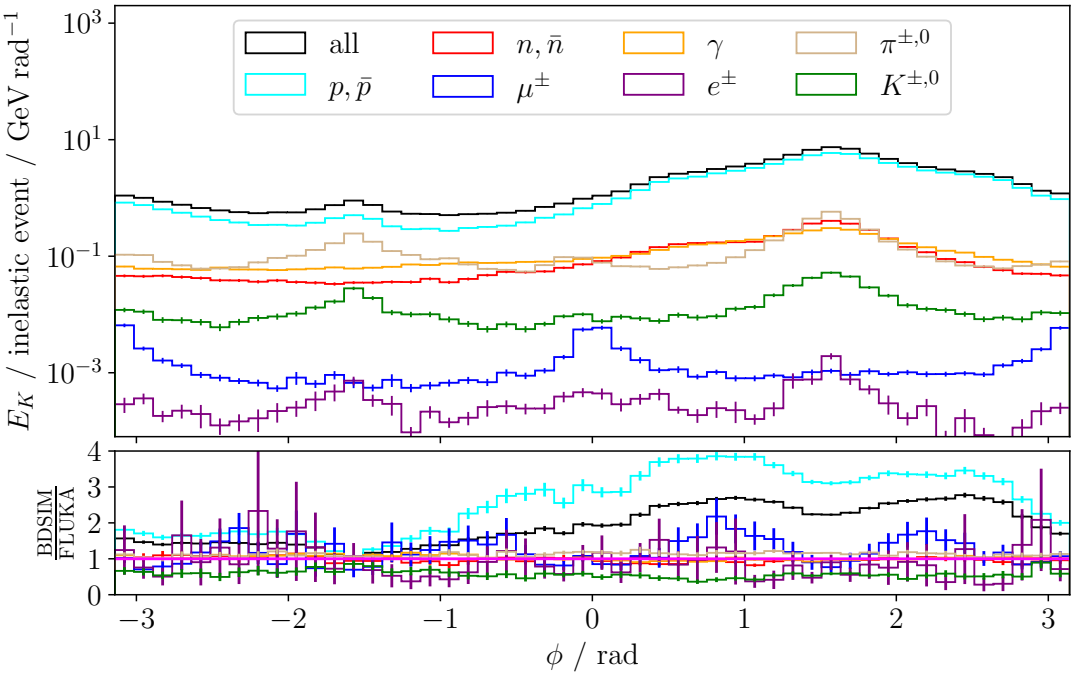
Results from the IR1 beam-gas simulations, with both kinetic energy cuts at 20 GeV and 20 MeV, are presented here. These show the range of features of the distributions of the secondaries originating from inelastic beam-gas events at the interface plane. Additionally, the high-statistic pressure bump simulations are compared alongside the IR1 beam-gas and the features discussed. These are compared and contrasted with existing comparable FLUKA simulations, and the BDSIM to FLUKA ratios in the relevant quantities are presented. The azimuthal distribution at the interface plane is important because as it's transmitted to the sub-detectors and isn't flat, it will give rise to so-called "missing momentum" from which novel physics may be inferred if not

otherwise flagged as a BIB event. Good, broad agreement between FLUKA and BDSIM is shown in Figure 5.11, particularly in the 20 MeV case. The characteristic shape in the distributions is recreated in BDSIM, which is primarily due to the vertical crossing angle at the IP. There are clear excesses in the flux at $\phi = \frac{\pi}{2}$ and $\phi = -\frac{\pi}{2}$. Both are due to the crossing angle at IP1, where the one at $\phi = \frac{\pi}{2}$ is larger because this is where the nominal proton beam is located, and at $\phi = \frac{-\pi}{2}$ is as a result of lower rigidity secondary particles. Indeed it is clear also that the difference between $\phi = \frac{\pi}{2}$ and $\phi = \frac{-\pi}{2}$ is most pronounced for the protons, suggesting that nearly on-momentum secondary protons are particularly relevant to the overall flux observed at the interface plane.

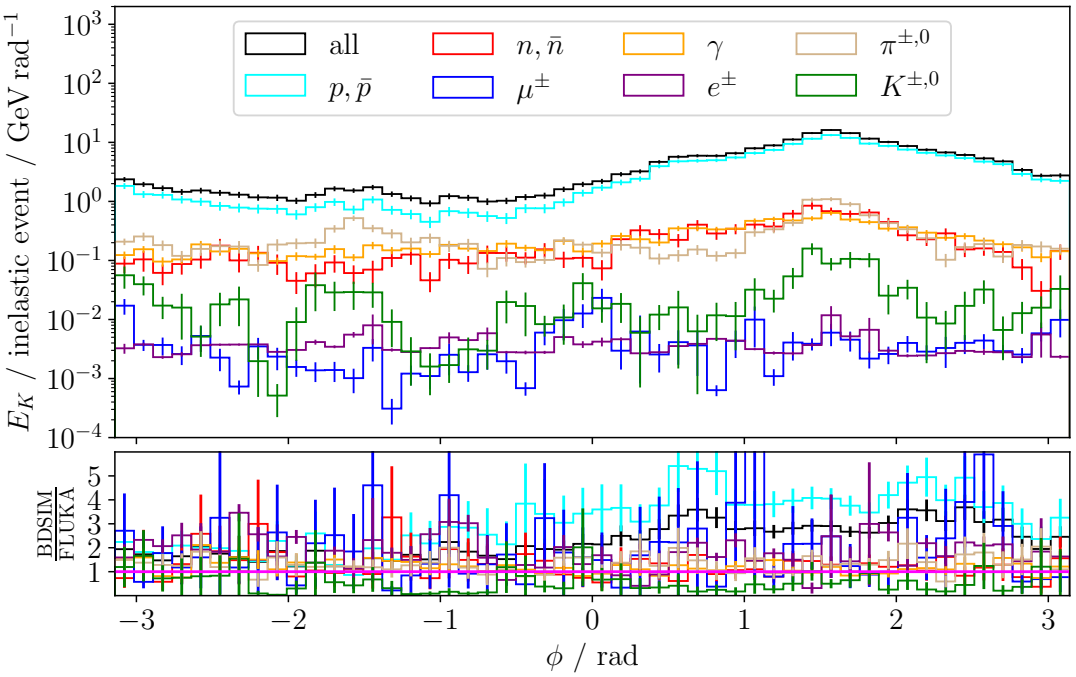
The other notable feature are the excesses at $\phi = 0$ and $\phi = \pi$ which are most pronounced in the muon distribution, but present throughout the charged particle distributions, which suggests magnetic fields are responsible. The aforementioned angles correspond to $\pm x$, i.e. a horizontal smear. The two separation dipoles, D1 and D2, which bring the nominal 194 mm separation between the two beams in the arcs to collision in the experimental insertions are the likely source of this effect. Figure 5.12 clearly demonstrates this relationship, the 148 m pressure bump (Figure 5.12a) is upstream of D1 and the peaks are clear, whereas at 58 m pressure bump (Figure 5.12b) is situated almost entirely after D1, and these peaks are completely absent.

Figure 5.13 shows the total contribution to the flux observed at the interface plane for inelastic beam-gas events at the different positions upstream of the interface plane. Further upstream, secondary protons dominate in transporting kinetic energy to the interface plane, and the only other type of secondary particle seen in any abundance are muons. Protons dominate over other types of secondaries because the machine itself is designed to transport protons and thus there will be a preponderance of protons which are captured and transported far downstream, particularly if these protons are on-momentum. The presence of the muons can simply be explained by their highly penetrating nature. Other particles interact within the machine components and are lost.

A number of other interesting features can be observed in Figure 5.13, firstly the jump at around 272.5 m in the muons and particularly the other non-proton secondaries

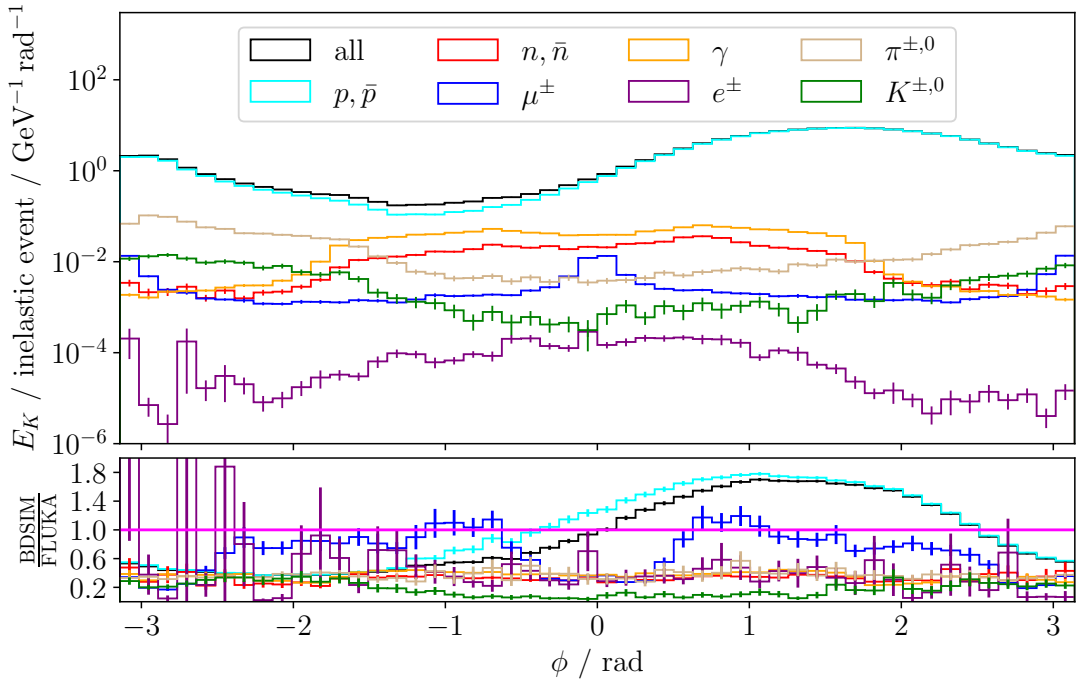


(a) Kinetic energy cut at 20 GeV.

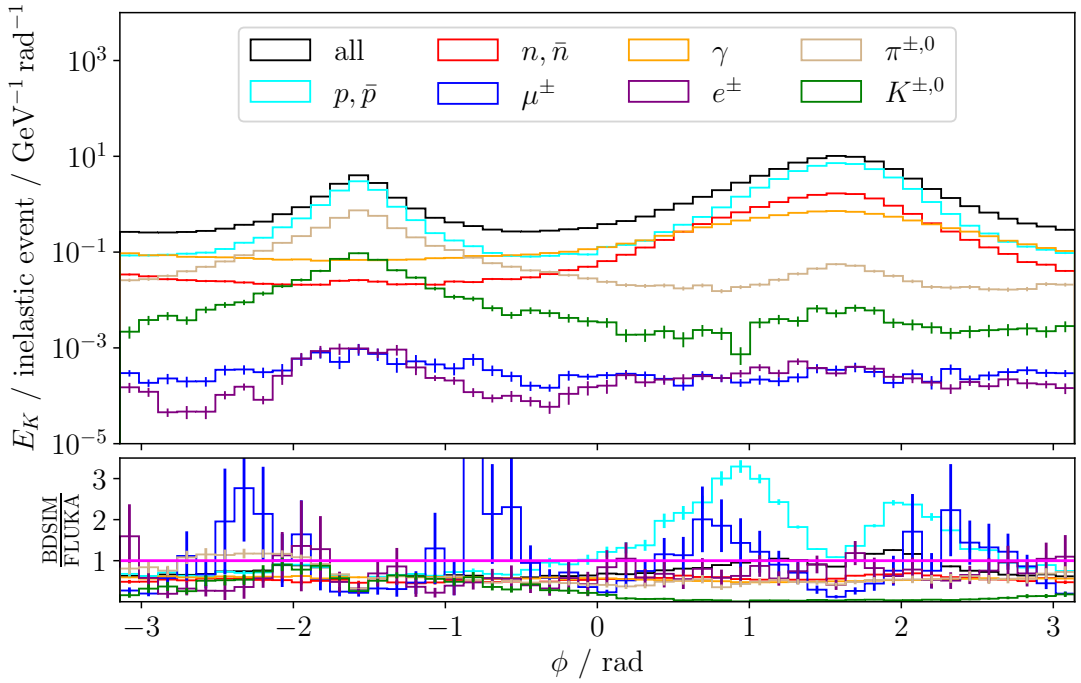


(b) Kinetic energy cut at 20 MeV.

Figure 5.11: BDSIM IR1 azimuthal ϕ at the interface plane with BDSIM/FLUKA ratio below, for two kinetic energy cuts, 20 GeV and 20 MeV. The solid magenta line in the ratio plot corresponds to BDSIM/FLUKA = 1.



(a) Pressure bump at 58 m.



(b) Pressure bump at 148 m.

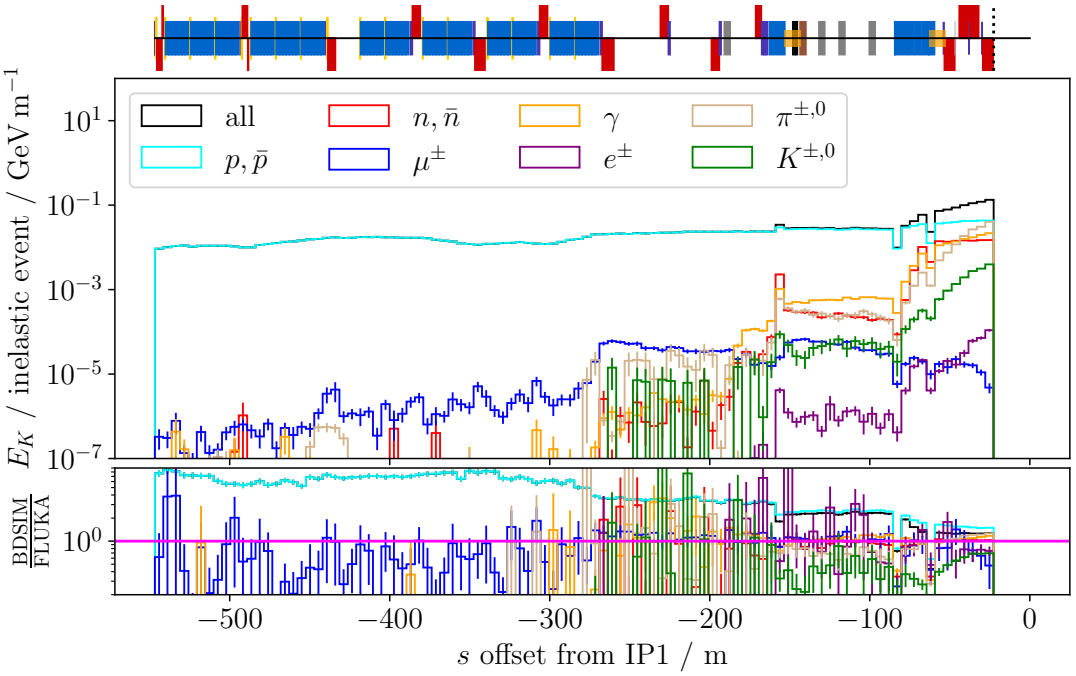
Figure 5.12: BDSIM azimuthal ϕ at the interface plane for the two pressure bumps, 148 m and 58 m with BDSIM/FLUKA ratio on the lower axes, with identical, 20 GeV, kinetic energy cuts. The solid magenta line in the ratio plot corresponds to BDSIM/FLUKA = 1.

corresponds to the end of the arc and the beginning of the long straight section. This demonstrates the role of the arc in acting as a spectrometer and preventing secondaries from reaching the interface plane and simply hitting the apertures. The fact that this bump is least prominent in the protons and the proton distribution is relatively flat along the entire length of the machine suggests there may be a dominance in the production of relatively on-momentum protons in the initial inelastic collision.

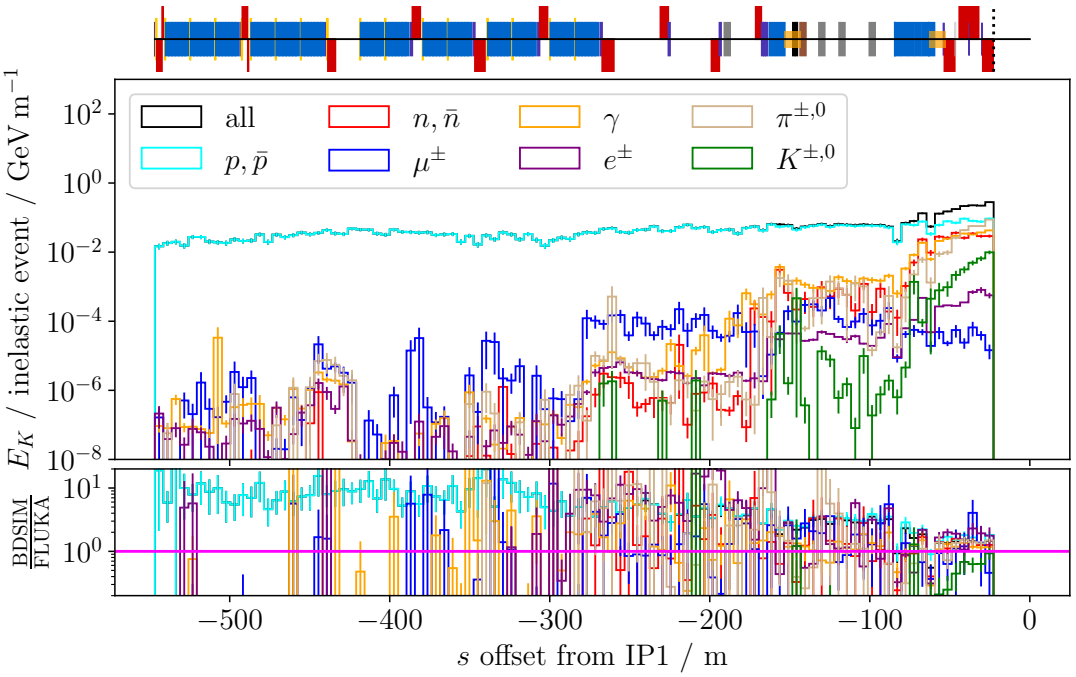
The next notable feature is around D2, situated around 160 m upstream of the IP where there is a spike in the transmission of non-proton and non-muon secondaries. This increase is most apparent for neutral particles. This is perhaps explained by considering the effect of beam-gas interactions in this magnet. Charged particles should be swept into the apertures, leaving only neutrals remaining. From the separation dipole closest to the interface plane, D1, onward, the transmission increases dramatically and so this region is likely to be of most importance to the BIB rate, excluding muons. This is further evidence that the dipoles play a dominant role in screening secondaries from reaching the interface plane.

The impact of the component geometry must be considered: the FLUKA model has incredibly detailed, bespoke component geometries, whereas BDSIM makes use of generic ones. Whilst BDSIM has an included TAN, shown in Figure 5.6, it is very basic and not nearly as detailed as the one used in the FLUKA simulation. The lack of apparent discontinuity in the ratios in Figures 5.13a and 5.13b suggest that precision modelling of the TAN is not critical for studying the impact of beam-gas events in the ATLAS detector.

Regarding a comparison between BDSIM and FLUKA, it is apparent that agreement between the two simulations improves monotonically with proximity to the interface plane. The good agreement in the muons even very far away from the interface plane suggests the secondary physics is comparable, albeit with one exception: there are too many on-momentum protons being created in the beam-gas event. Clearly, closer to the interface plane it is likely that geometry will have less influence as the secondaries are going to see less material. However, the impact of the geometry closer to the interface plane is not completely irrelevant, and the agreement here suggest that at the very least, the BDSIM model's geometry is quite sufficiently accurate, at least at lower radii.



(a) Kinetic energy cut at 20 GeV.

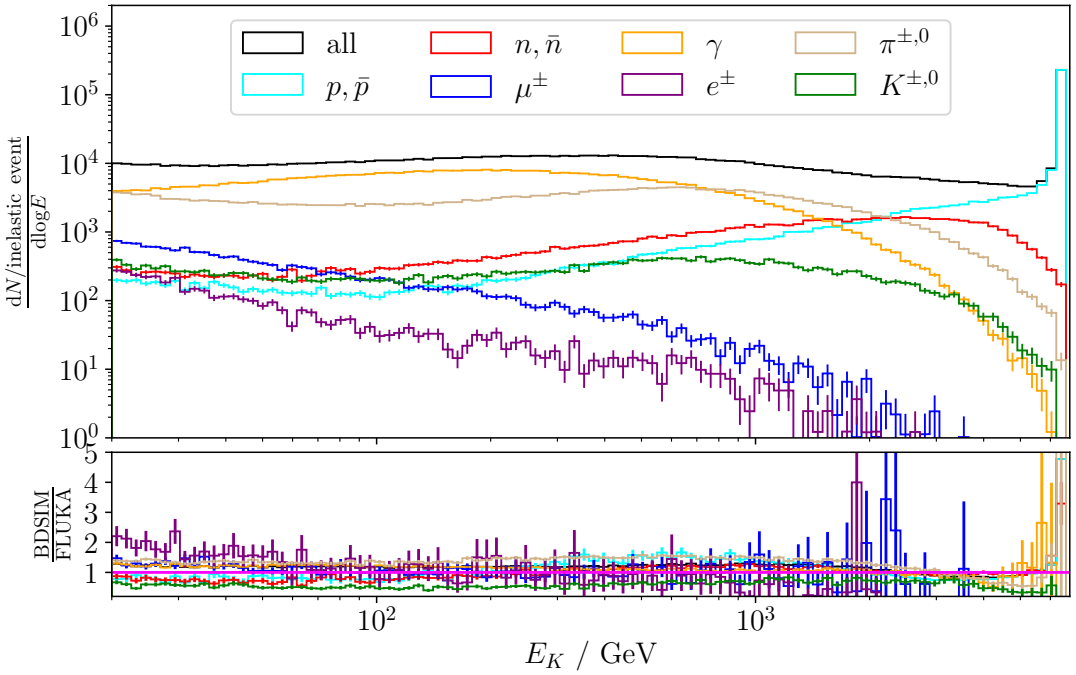


(b) Kinetic energy cut at 20 MeV.

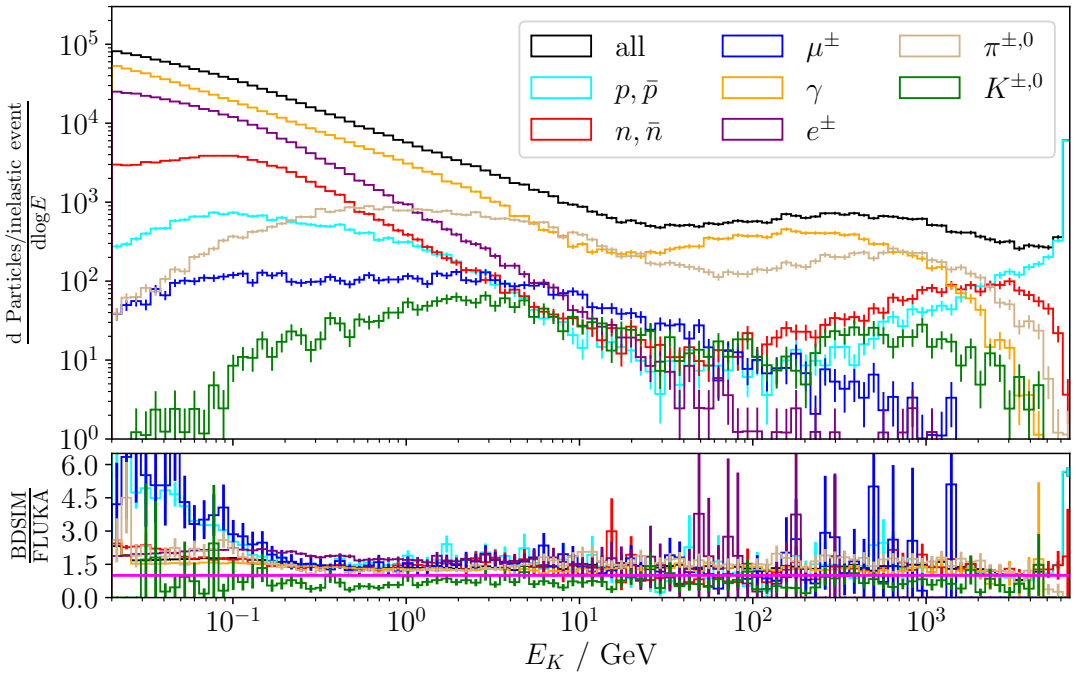
Figure 5.13: BDSIM IR1 beam-gas event z, weighted by kinetic energy for particles at the interface plane with BDSIM/FLUKA ratio below, for two kinetic energy cuts, 20 GeV and 20 MeV.

A number of other interesting features are apparent in Figure 5.13. It is clear that far from the interface plane, and particularly in the arc, very few particles besides protons reach the interface plane. This is because the protons are more likely to be transported by the machine further, as they'll tend to be closer to the design rigidity in comparison with other secondaries, either because they have the wrong charge or because they are very off-momentum. For secondaries originating from beam-gas events closer to the interface plane, the rates are higher because they are able to travel the shorter distance before being lost. The one exception is the muon, which persists at large s due to their highly-penetrating nature and their relatively long half life. The large number of high-energy protons can be seen in the kinetic energy distribution of protons in Figure 5.14. Excellent agreement is clear across the entire spectrum, except for the clear excess around the 6.5 TeV.

The larger transmission far from the interface plane of protons shown in Figure 5.13 and the larger number of protons at the nominal beam energy shown in Figure 5.14 together suggest the main source of disparity between the two simulations is the physics of the initial physics interaction. The inelastic proton-nucleus physics of BDSIM seems to be producing far more on-momentum protons which are transported directly to the interface plane. One way to examine the initial inelastic beam-gas physics in this instance is to compare the spectra at the interface plane from inelastic collisions originating very close to the interface plane. At such distances, the particle showers will either stay within the beampipe, or see very similar geometries where it is expected that the FLUKA and BDSIM models agree most. Additionally, the quadrupole field of the inner triplet will minimally affect the shower, and any bending that does occur should be very similar in both cases as the secondaries will remain at low radii, nearest to the good field region where the fields in both simulations will be most similar.



(a) Kinetic energy cut at 20 GeV.



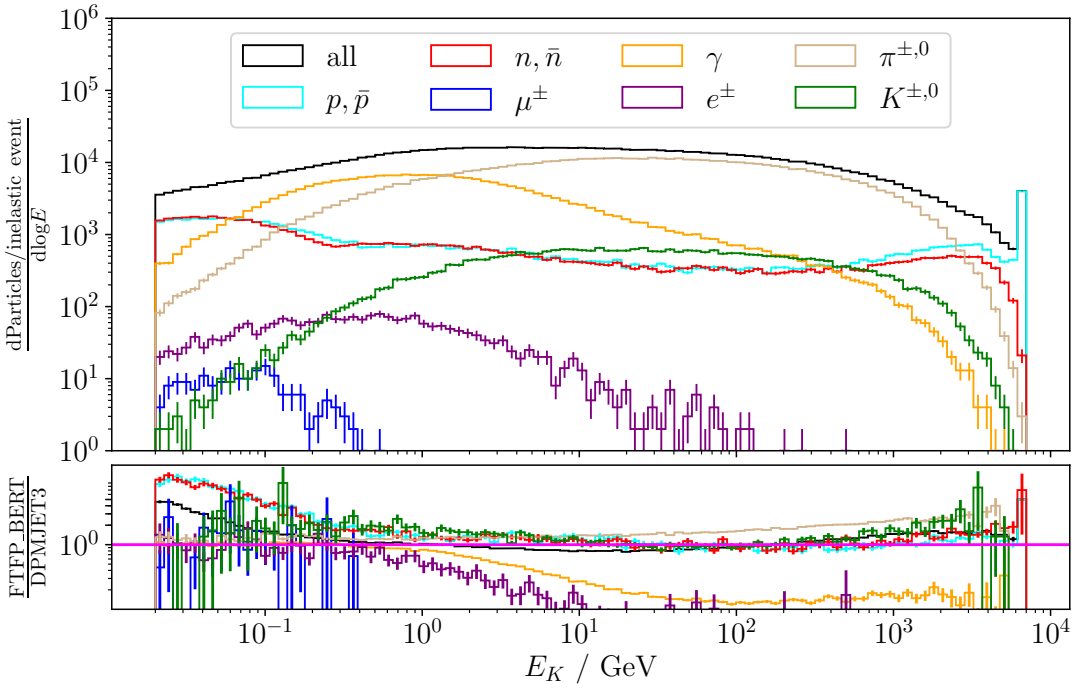
(b) Kinetic energy cut at 20 MeV.

Figure 5.14: BDSIM IR1 beam-gas event z, weighted by kinetic energy for particles at the interface plane with BDSIM/FLUKA ratio below, for two kinetic energy cuts, 20 GeV and 20 MeV.

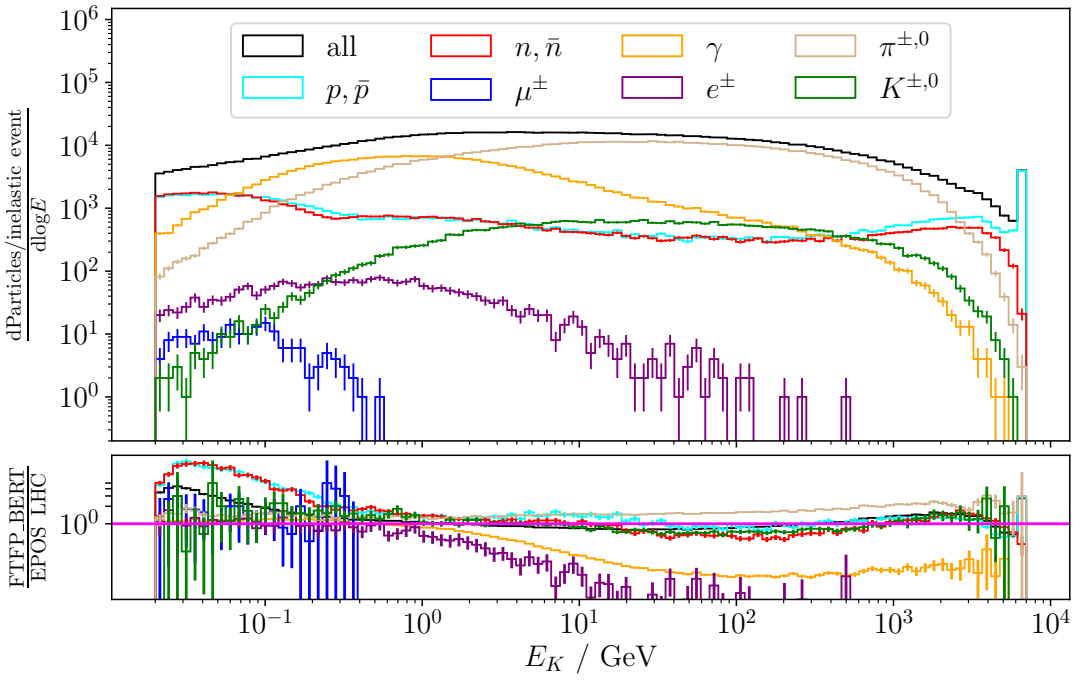
5.4.0.1 Beam-gas interaction physics

The physics of the initial beam-gas interaction is important as it ultimately determines what is seen downstream at the interface plane. In this section the inelastic proton-nucleus physics of Geant4 as used in the BDSIM IR1 model is compared with two dedicated Monte Carlo event generators and used to explain the features of the distributions seen at the interface plane when comparing FLUKA and BDSIM.
Through Geant4, BDSIM has access to a range of physics processes suitable for a range of applications, described in more detail in Section 3.4. In these simulations, the reference physics list `FTFP_BERT` was used, which should be accurate at the energy ranges in the simulations used. This physics list is also used in both ATLAS and CMS for their high energy physics applications.
The physics of the initial beam-gas interaction was examined with a similar approach as used in the beam-gas simulations. In BDSIM a model consisting of a single drift tube of length 1 µm with a vacuum material of nitrogen-14, the same as used in the IR1 beam-gas simulations, was used to study the physics of the proton-nitrogen interaction. To force an interaction in this thin slice of nitrogen the `protonInelastic` process was biased by a factor of 10^{100}, which is large enough to effectively guarantee an event within the 1 µm slice of gas. Finally, a sampler was placed immediately afterwards to record all the secondary particles produced.
Two Monte Carlo event generators were used to compare with the `protonInelastic` process of Geant4: DPMJET-III [125] and EPOS LHC [126], both of which were accessed through the command line interface Cosmic Ray Monte Carlo (CRMC) event generator [127]. The generator used for the FLUKA simulations was neither of these, instead PEANUT was used [121, 124]. However, PEANUT is not separable from FLUKA, so a direct comparison was not readily achievable. However, like PEANUT, DPMJET-III (which has an interface to FLUKA) and EPOS LHC have all been validated with LHC test beam data, so should be broadly similar at the energy range of the beam-gas simulations. The generated sample was in the HepMC format [128] and to translate it into the BDSIM format for analysis required loading the HepMC into BDSIM and then immediately writing it to file. This introduced another source of systematic error, as many particles were not supported in the FTFP_BERT physics list,

particularly for the DPMJET-III sampler, and were excluded entirely.
In total, separate samples of one million proton-nitrogen inelastic collisions each were generated with the methods described above, each with a kinetic energy cut of 20 MeV. Figure 5.15 shows the spectrum in the kinetic energies from DPMJET-III and EPOS LHC and are compared with that of the inelastic proton physics of FTFP_BERT. Excess proton production in BDSIM can clearly be seen. The relative proton fluxes for the two highest-energy bins in Figure 5.15a (FTFP_BERT / DPMJET-III) and Figure 5.15b (FTFP_BERT / EPOS LHC) are 5.0 ± 0.3 and 2.8 ± 0.1, respectively. Other than this, both DPMJET-III and EPOS LHC appear to be broadly in agreement with each other and since both are validated against LHC beam data, it may be appropriate for future BIB studies to use a generator for the initial input distribution. The FTFP_BERT reference physics list is designed for high energy physics, and whilst it is used in the simulation of both ATLAS and CMS, the typical workflow is to use a Monte Carlo event generator for the initial collision. This is another reason to consider the use of a generator in the future.



(a) DPMJET-III.



(b) EPOS LHC.

Figure 5.15: Kinetic energy spectrum for the secondary products from proton-nitrogen inelastic events at 6.5 TeV, showing FTFP_BERT in ratio to output from two Monte Carlo event generators, DPMJET-III, and EPOS LHC.

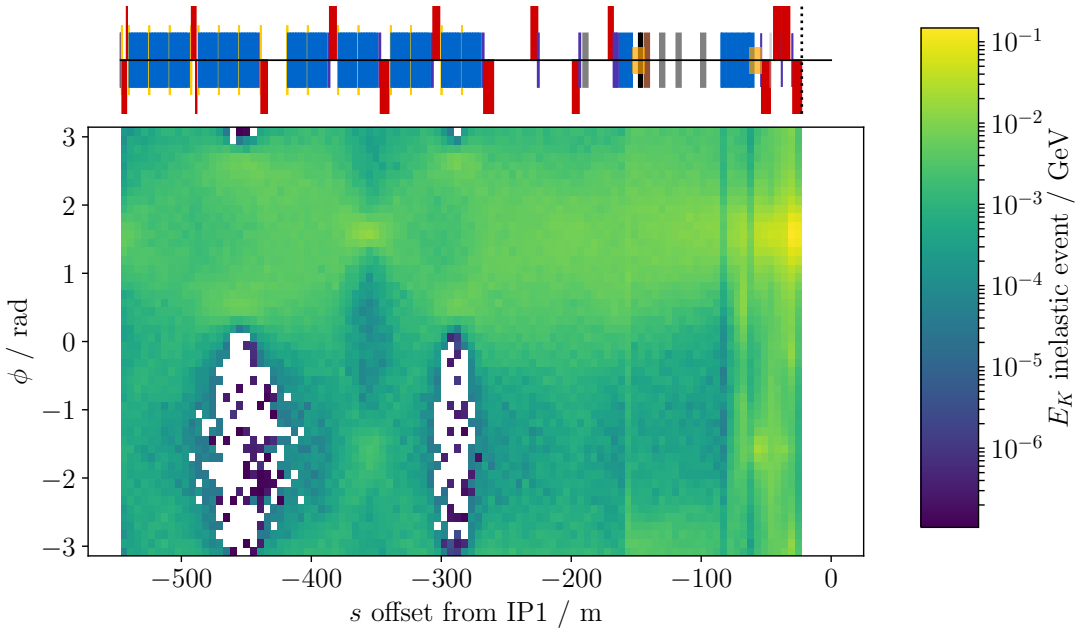
5.4.0.2 Influence of pressure on beam-induced backgrounds

In all plots presented thus far, the events have been weighted such that weighting is uniform in s for the beam-gas events. This is useful for analysing the accuracy and efficacy of the BDSIM model, however, to convert these rates into physical ones that can be compared directly with ATLAS data requires further re-weighting according to a longitudinal profile of the pressure found in IR1.

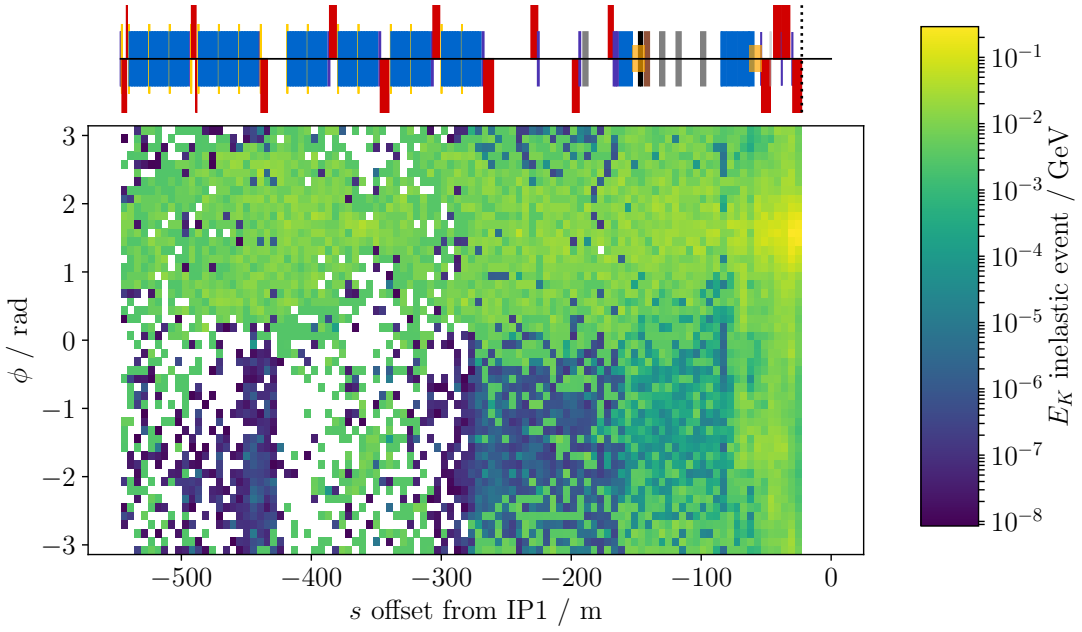
The vacuum pressure found in the LHC is dynamic, depending on both beam energy and beam intensity [129]. At the time of writing, pressure profiles for the beam energies and intensities seen in Run II are not available. However, whilst it is not yet possible to get the true physical rates, it is worthwhile examining the potential impact of any such pressure map on the sorts of distributions seen at the interface plane and in the ATLAS detector.

By considering the contribution to distributions at the interface plane from different beam-gas interaction points in s, one sees how the shapes and rates may be modulated at the interface plane, and to what extent the pressure affects the beam-gas interactions detected in ATLAS beyond just the rate. In Figure 5.16, the correlation between the initial beam-gas event location s, and the kinetic energy flux in azimuthal ϕ is shown. For the kinetic energy cut at 20 GeV (Figure 5.16a), the impact of a pressure distribution is most stark as the transmitted kinetic energy in ϕ is clearly modulated over the position of the beam-gas event in s. The features in the arc are not as easily explained by referring to the optical parameters in this region as no clear correlation exists. However, in the arcs, which tend to be the coldest sections of the accelerator, the pressure is going to be very low and the contribution to the flux at the interface plane is going to be low, also because the arc effectively screens off-momentum particles from reaching the interface plane. A fine structure is not so clear with the 20 MeV kinetic energy cut (Figure 5.16b), this may be due to poor statistics, but it appears to be the case that with the lowest kinetic energy cut, the peak at $\phi = \frac{\pi}{2}$ due to the crossing angle is particularly well-defined. However this is likely because the lower kinetic energy particles have very small chance of reaching the interface plane, leaving mostly on-momentum particles to dominate.

The transmission in the radial kinetic energy to the interface plane in s is shown in



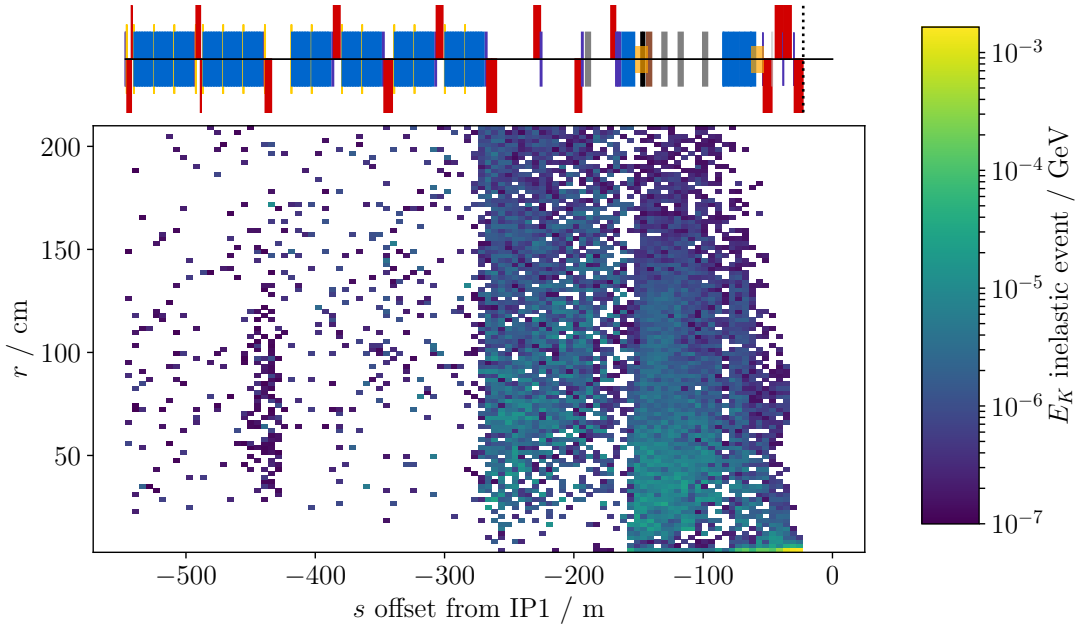
(a) Kinetic energy cut at 20 GeV.



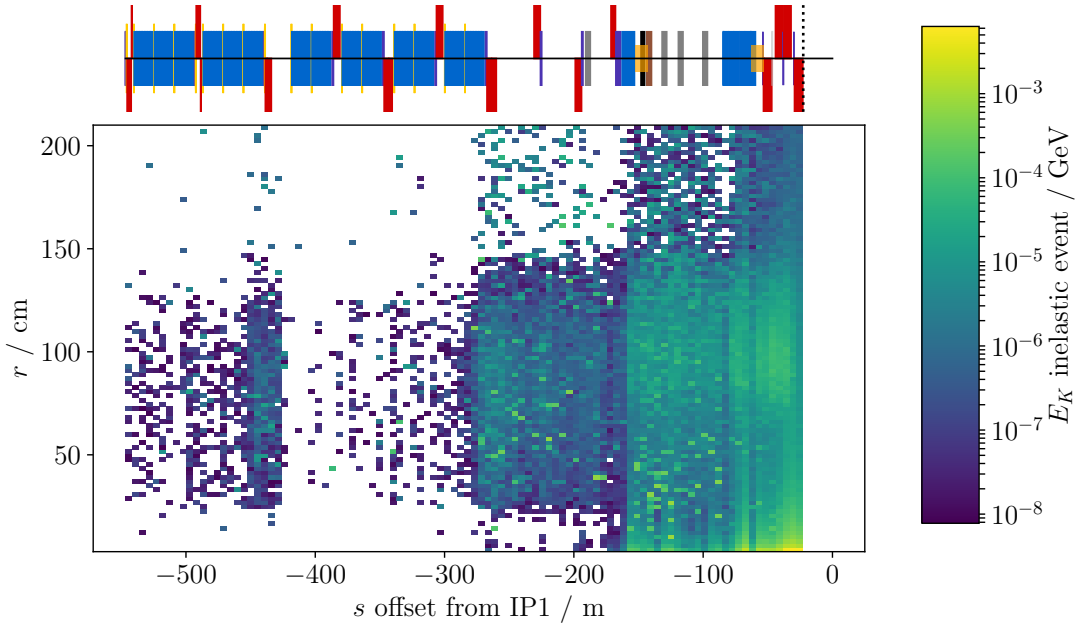
(b) Kinetic energy cut at 20 MeV.

Figure 5.16: The contribution to the kinetic energy weighted distribution in ϕ for the different beam-gas s locations seen at the interface plane.

Figure 5.17. Only the radii corresponding to the inner detector are shown (3 cm to 2100 cm) to account for the dominance of the on-momentum protons travelling in the beampipe. At the 20 GeV kinetic energy cut (Figure 5.17a), one can see that nothing from the arc reaches the radius of the pixel detector (3 cm to 15 cm), and the influence of the arc on screening BIB in the inner detector is abundantly clear from the sudden increase at the end of the arc ($s = -172.5$ m). The pixel detector is most affected by beam-gas in the inner triplet, but the outermost layers of the inner detector see no BIB from the inner triplet at all. Much more fine structure is visible at the 20 MeV kinetic energy cut, potentially due to the various changes in the tunnel radius. The beamline is offset in the tunnel, so the tunnel walls surrounding the beamline are not all at the same distance from it, but the tunnel wall nearest to the beamline is on the order of 1.5 m in the region upstream of the collimators. Closer to the interface plane the tunnel opens to a much larger radius, before reducing down again to wrapping the beamline tightly. The absence of secondaries at the pixel radius from upstream of D2 to the beginning of the arc is likely as a result of the TCT immediately downstream, as well as D2 sweeping the particles into the sides of the apertures. In the arcs, there is also a structure where in the focussing-defocussing (FODO) cells there seems to be a gradual increase in the transmission towards the end of the cell, before dropping again. This is likely due to an optical effect, but an explanation is not immediately forthcoming, as there is no clear correlation between this structure and the various optical functions. In both Figure 5.16 and Figure 5.17 it is apparent in the plots shown above that beyond just affecting the rate, any such pressure profile could drastically impact the shapes and radii of the distributions seen at the interface plane depending on just how the pressure varies along IR1. In the cold regions of the arc the pressure is going to tend to be lower, but clearly for the inner detector for any reasonable pressure the contribution to the rate is going to be negligible. The characteristic muon shape will be most impacted. However, across the lengths of the pressure bumps, their distributions change very little and in this case the pressures will mostly just determine the absolute rates observed.



(a) Kinetic energy cut at 20 GeV.



(b) Kinetic energy cut at 20 MeV.

Figure 5.17: The contribution to the kinetic energy weighted distribution in r for the different beam-gas s locations seen at the interface plane.

5.4.0.3 Influence of the crossing angle on beam-induced backgrounds

Throughout Section 5.4 the crossing angle in IP1 has been used to explain the features at the interface plane a number of times, the most prominent of which is the peak at $\phi = +\frac{\pi}{2}$. This angle corresponds to the vertical offset in the beam at 22.6 m, before it is brought to the interaction point, where there is no offset. Here the influence of crossing angle is shown explicitly with a dedicated simulation. Previous studies [15] have investigated the impact of the crossing angle on the particle distributions at the interface plane.

For this simulation the 2015 collision optics which were used for the IR1 beam-gas simulations presented in Section 5.4 were used as a starting point. Removing the cross angle from this lattice simply required disabling every kicker in the lattice.

Additionally, a 20 GeV kinetic energy cut was used.

Figure 5.18 shows the azimuthal ϕ distribution weighted by kinetic energy for the model without a crossing angle. The impact of the crossing angle is plain to see, the bump which is usually at $\frac{\pi}{2}$ is totally absent. Instead, there are four much smaller bumps distributed at $\phi = 0$, $\phi = \pm\frac{\pi}{2}$ and $\phi = \pm\pi$. The peaks at $\phi = \pm\frac{\pi}{2}$ are less easily explained without the crossing angle; as these are completely absent in the neutrals, they are most likely due to magnetic fields.

Figure 5.19b is useful for explaining the features in Figure 5.18, as the link between ϕ and the original beam-gas location can be seen. Two features are prominent here. First, the impact of the inner triplet is the most likely explanation for the $\phi = \pm\frac{\pi}{2}$ peaks in Figure 5.18; as the quadrupoles bend in both the horizontal and vertical planes. Second, in the arc there is a clear modulation in the transmission in the kinetic energy in ϕ. There is no clear relation between the optics in the arc and this phenomenon remains unexplained. It is possible that losses in the arc in these areas will tend to hit the aperture further downstream at particular points, but it is not immediately clear. To understand this, a full trajectory analysis may be necessary. However, as the vacuum pressure in the arc tends to be much lower than further downstream in the IR, the impact on BIB should be minor compared with the losses closer to the interface plane. Additionally, it's clear that most of the particles in this region are protons, which if they made it to the interface plane are most likely on-momentum and within

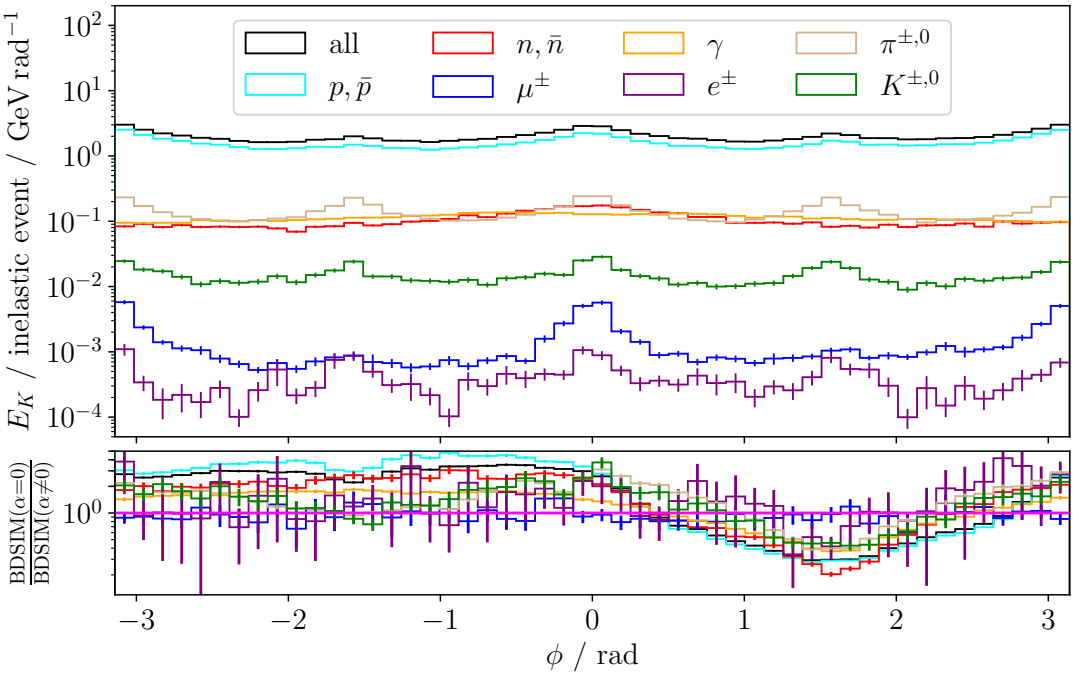
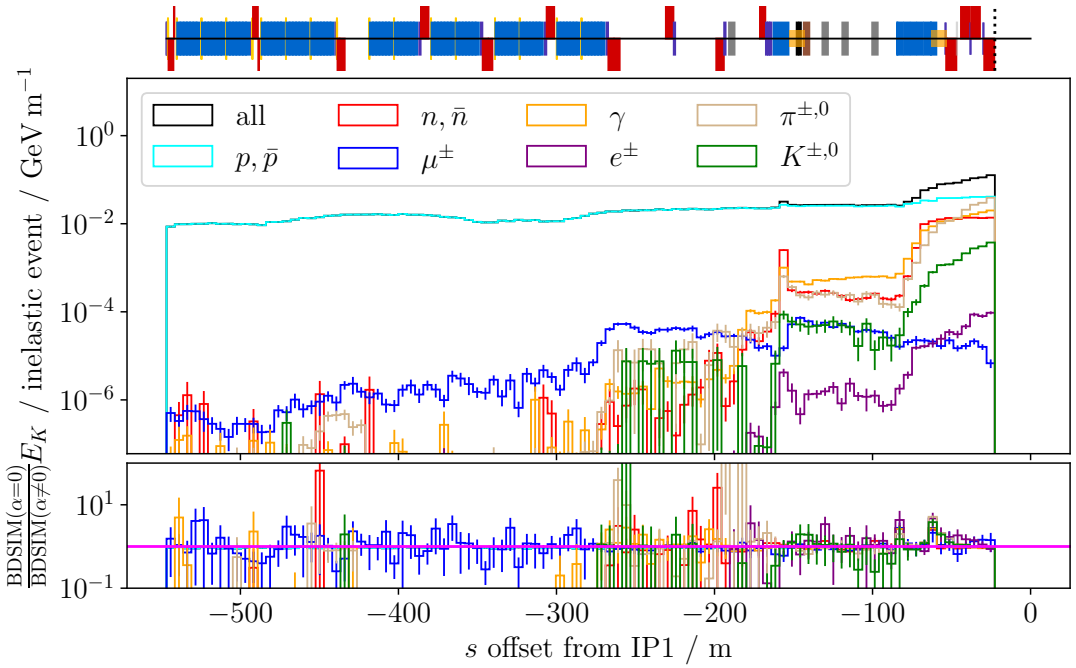
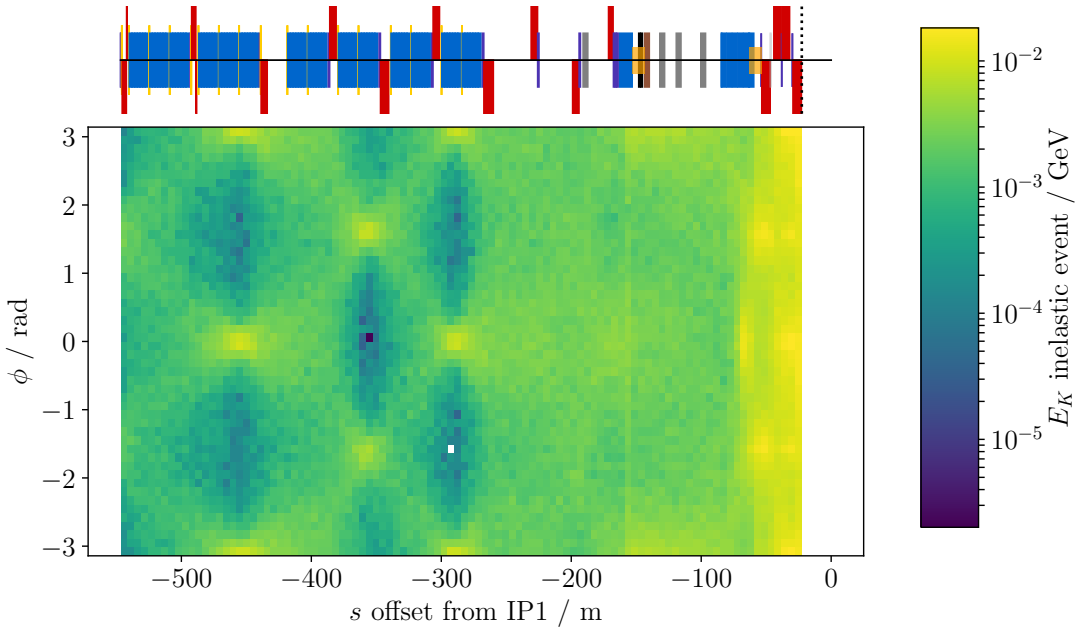


Figure 5.18: The azimuthal ϕ distribution at the interface plane for the IR1 model without any crossing angle with a kinetic energy cut of 20 GeV. The ratio of the distributions from this simulation to the comparable IR1 beam-gas simulation with the crossing angle is shown in the lower plot.

the geometric aperture meaning again these are unlikely to affect the BIB rate in the subdetectors in ATLAS.



(a) Total kinetic energy transmitted to the interface plane in the original beam-gas s location.

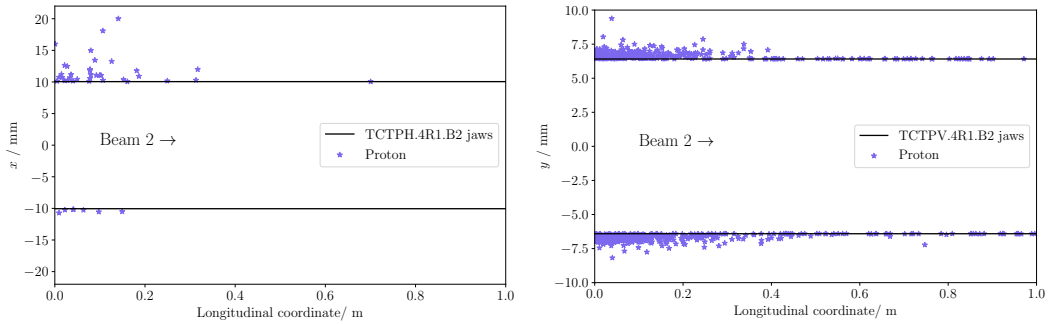


(b) The transmission in kinetic energy in ϕ upstream of the interface plane.

Figure 5.19: The transmission of kinetic energy to the interface plane from upstream of the interface plane, without a crossing angle, with a kinetic energy cut of 20 GeV.

5.5 Beam-halo simulations

A brief study of the BDSIM IR1 model applied to beam-halo is presented here. Whilst there are two possible sources of TCT hits, global beam-gas and collimator leakage, only results from collimator leakage are presented here. Realistic hits on the TCTs from collimator leakage were generated in SixTrack, treating the leakage from the vertical and horizontal collimators in IR7 as separate scenarios. This was achieved by impacting 6.5×10^7 protons on the primary collimators. Leakage from the collimators results in hits on both the vertical and horizontal collimators in the insertion regions (IRs). These hits are then passed into a dedicated Monte Carlo particle transport code, such as FLUKA or BDSIM. The input distribution from the SixTrack collimator leakage studies as well as the following FLUKA simulation were both sourced from [122].
The same 2015 optical configuration and model was used as for the IR1 beam-gas simulations described above in BDSIM to ensure a meaningful comparison with FLUKA. However, the exact collimator settings for the BDSIM model were reconstructed from the input distributions such that the collimator jaws perfectly enveloped the input distributions. This was necessary as the actual collimator openings used in the FLUKA simulations were not available. Reconstructing the openings in this way most likely introduces a source of systematic error. However, as the total number of primary losses is quite large, the error is likely not too great, because for a sufficient number of proton losses it is very likely that the losses will be very near the jaw boundary. Another potential source of discrepancy is that in the BDSIM simulations the protons are directly inserted in the models at the SixTrack primary loss points in the collimators, whereas in FLUKA an interaction was forced at that point. In BDSIM the protons will not necessarily instantly shower at their starting point, and will instead travel some distance before interacting or even leave the collimator without undergoing an inelastic collision at all. The combined input distribution, spread over both TCTs, is shown in Figure 5.20.
Figure 5.20 shows that the vast majority of the losses are at the front corner of the collimator, and it can reasonably be assumed that most of the flux at the interface plane comes from interaction at these points. Figure 5.21 shows the relationship between the location of the inelastic event and the resulting total kinetic energy seen downstream at



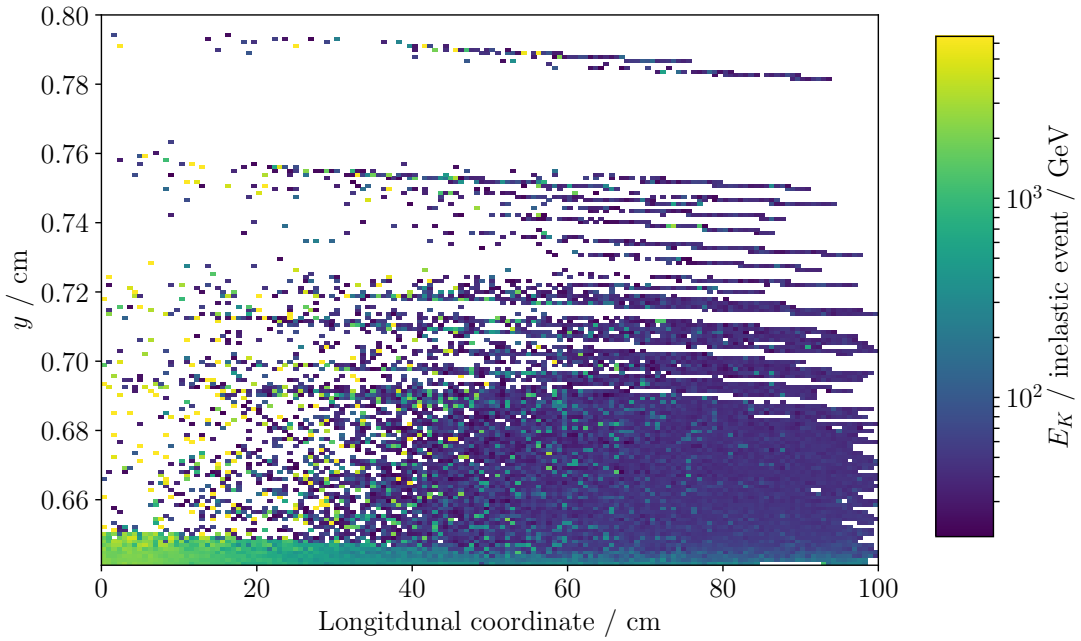
(a) The horizontal TCT primary losses from SixTrack.

(b) The vertical TCT primary losses from SixTrack.

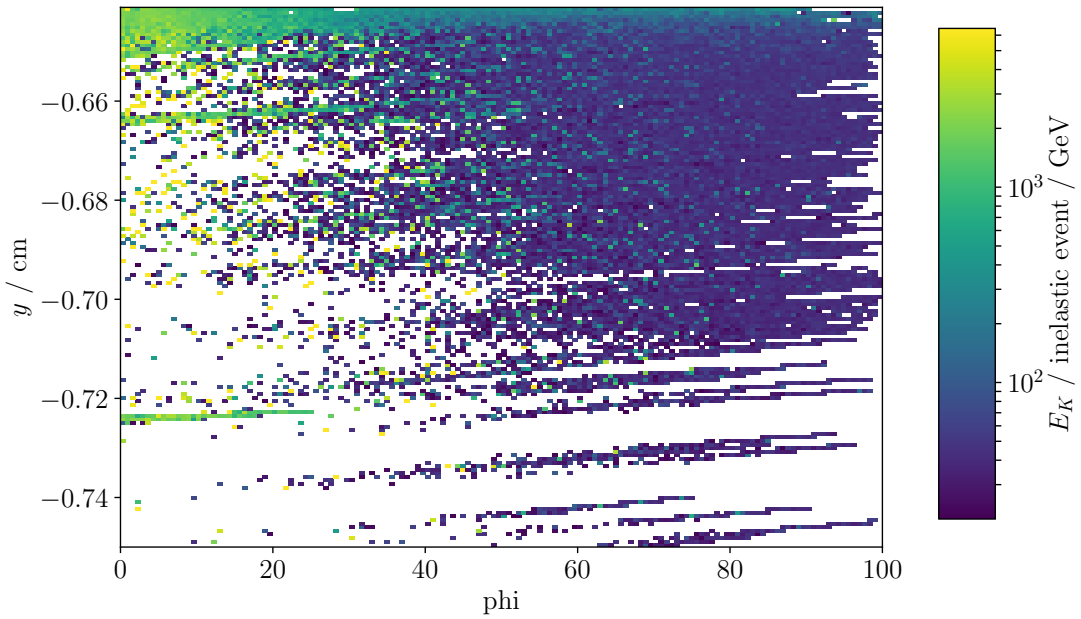
Figure 5.20: Primary loss points in SixTrack sourced from [122] used as the primary input distribution for the BDSIM beam-halo distributions. The exact collimator openings were not available and were reconstructed from these distributions. The end of the horizontal tertiary collimator precedes the start of the vertical one by 1 m. Both collimators are 1 m in length. The jaws are not centred on zero, as is standard for collision optics .

the interface plane. The energy deposition in each bin is normalised by the number of inelastic events in that bin. This then shows the influence of the longitudinal and transverse coordinates of the inelastic event on the total flux at the interface plane. A number of features are apparent in Figure 5.21, firstly the "tracks" which are clearest furthest from the collimator edge are from the primary protons which are fired in BDSIM deep in the collimators (c.f. Figure 5.20b). These will generally travel some distance before undergoing the inelastic event with the collimator material, resulting in longitudinal tracks from the primary origin. Secondly, even with the aforementioned normalisation, the front edge still dominates the kinetic energy transmission to the interface plane. This is likely because many of the showering secondaries from this region will only see a small amount of collimator material before reentering the beampipe and travelling relatively unimpeded downstream. Impacts further down stream along the collimator are clearly not as penetrating to the interface plane. The beam at this point is diverging, meaning that any resulting secondaries which are directed towards the centre of the beampipe will likely be at very shallow angles. At these angles they are much more likely to be intercepted by the neutral absorber (TAN) immediately following the TCT. This could be further studied by linking the location of the primary hit to energy deposited further downstream, which is readily achievable in BDSIM. Also readily apparent is the negative correlation between impact parameter and the flux at the interface plane; the deeper the penetration the more collimator

material the showering secondaries will see. However, even deep within the collimator jaws, there are clear spikes, likely from the production of one or more high energy muons.



(a) Upper collimator jaw.



(b) Lower collimator jaw.

Figure 5.21: Primary loss location in the two jaws of the collimator, weighted by the resulting kinetic energy seen at the interface plane. The bins are normalised by the number of particles lost in that bin.

5.6 Conclusion

In this chapter a detailed BDSIM model of IR1 was described in detail and applied to the study of BIB and compared with a similar FLUKA model for various scenarios. Good agreement at the interface plane was clear across the many simulations, showing that model is suitable for further, novel simulations of BIB in IR1. The clearest cause of disparity between BDSIM and FLUKA was the difference in the physics of the initial beam-gas interaction. For future simulations it may be advisable to use a Monte Carlo event generator for the initial interaction, as BDSIM sees far more high energy proton production than FLUKA, DPMJET-III and EPOS LHC. BDSIM mostly does support such a workflow as it can load particles in the High Energy Physics Monte Carlo (HEPMC) format [128], the common output format used by Monte Carlo event generators. It does not however support inserting the products at many different positions as would be required for BIB simulations, but would be a worthy addition to the code for future BIB simulations. The benefit of the existing approach is that the positions and coordinates of the beam-gas locations are guaranteed to be optically correct. A possible implementation with an event generator would be to sample the interaction points as usual, but instead of tracking the secondary shower, immediately kill the event and record the position and angle. The positions and angles of these locations could then be fed back into BDSIM as starting coordinates for events from a generator.

Further studies will be necessary for the future HL-LHC, and these simulations provide a firm foundation for this work. As the difference in the initial beam-gas interaction currently dominates the distribution at the interface plan, to further evaluate the accuracy of the model it may be necessary to extract the generated distributions from FLUKA for use in BDSIM. Previous comparisons [16] of two independent models, FLUKA and MARS, have shown excellent agreement with each other—much better than the comparison here with FLUKA. One key difference is in that study, identical inputs were used, further motivating a similar approach in BDSIM. More subtle differences in the model may only become apparent when the physics of the beam-gas event are in agreement. For example, the BDSIM IR1 model has no tunnel in the arc, where as the FLUKA model does, so to get the most accurate muon rates at the

interface plane this may be necessary. Additionally, the FLUKA model is extremely detailed, using detailed component geometries which have been iterated on over many years. The BDSIM model is currently dependent on generic components and generic fields both in the beampipe and in the yoke. The FLUKA model on the other hand has bespoke field maps for the entire component. This may be particularly relevant in the inner triplet. Another example is the offset in TAN, which is totally absent, and as it is a shielding component and one of its purposes is to shield the detector from BIB, a more accurate model may provide much more accurate rates at the interface plane.
In Chapter 3, the conversion of FLUKA geometry to GDML is described. One possibility is to convert the entire library of components from the FEDB and use all of these for the IR1 model. Regardless, as the HL-LHC will be around well into the 2030s, building these components for use in BDSIM will very likely be worthwhile in the long term for the wide range of studies which will be required, and BDSIM could be used for a wide range of applications in the study and optimisation of the machine. This is not least because conditions will in general be more adverse in the HL-LHC.
Further studies leveraging the full capabilities of BDSIM would be particularly interesting. For example it is possible to record the full trajectories of the particles, allowing for a more detailed analysis. Additionally, events at the interface plane could be correlated with energy deposition upstream, showing where shielding is most effective or might be useful to reduce BIB rates.
A further step would involve the use of a pressure profile to more accurately ascertain the importance of beam-gas events in certain locations. Section 5.4.0.3 shows that a pressure profile would likely have a meaningful impact not just on the absolute rates, but also the shape of the distributions seen at the interface plane. With a pressure map it would then be possible to pass the distributions at the interface plane into the ATLAS detector for a full analysis of BIB. For example it would be possible to correlate the distributions at the interface plane with what is seen in the pixel detector, and determine to what extent the azimuthal asymmetry at the interface plane is passed onto the subdetectors. Even without a pressure map, however, it would be possible to perform a meaningful analysis of the distributions from the pressure bump run. This is because, as seen in Section 5.4.0.2, the location of the beam-gas event over the 10 m of

the pressure bump has less impact on the resulting flux at the interface plane.

CHAPTER

SIX

BEAM-INDUCED BACKGROUNDS IN THE ATLAS PIXEL DETECTOR

In the previous chapter the simulation of beam-gas and beam halo in the region upstream of the ATLAS using BDSIM was described. The resulting particle showers from these sources were tracked through the lattice up until the interface plane, at which point the simulation was terminated. The particle distributions at the interface plane were shown, and their characteristic features were described. The next stage is to then pass these particles at the interface plane through to the official ATLAS Geant4 simulation. Preliminary results are presented in this chapter from these Monte Carlo simulations. Furthermore, these results are qualitatively compared with real data from dedicated ATLAS pressure bump runs and particularly with reference to the results from the previous chapter. The comparisons here are focused on the Pixel Detector, which is the subdetector closest to the collision point. Finally, proposed future work is discussed.

6.1 Beam-induced backgrounds in the Pixel Detector

The ATLAS detector situated in IP1 of the LHC features numerous subdetectors, and the Pixel Detector is the one closest to the interaction point. As originally built, it consisted of three layers of pixel detectors, but was upgraded before the start of Run II with an additional fourth layer, the Insertable B-Layer (IBL), at 3.3 cm, which is closest to the beampipe. The pixel size from the original 3 layers was $50 \times 400\,\mu m$, but

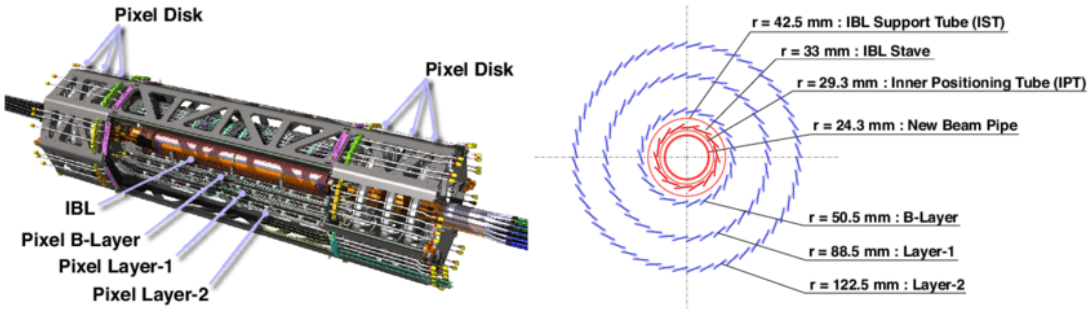
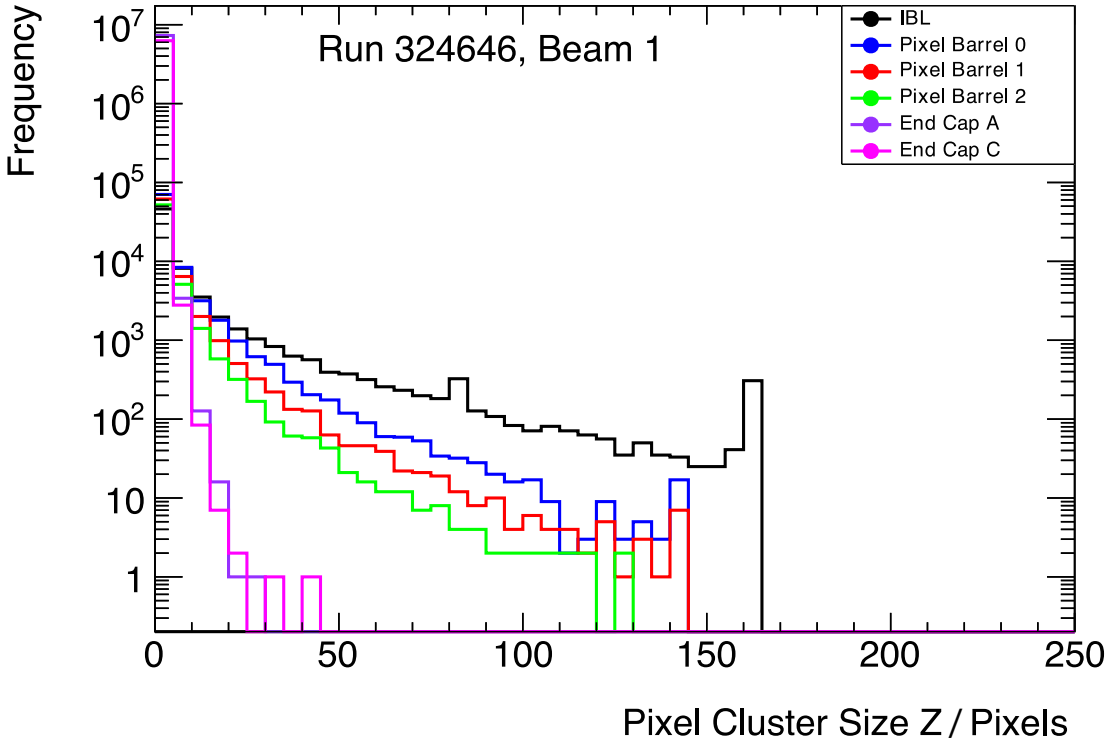


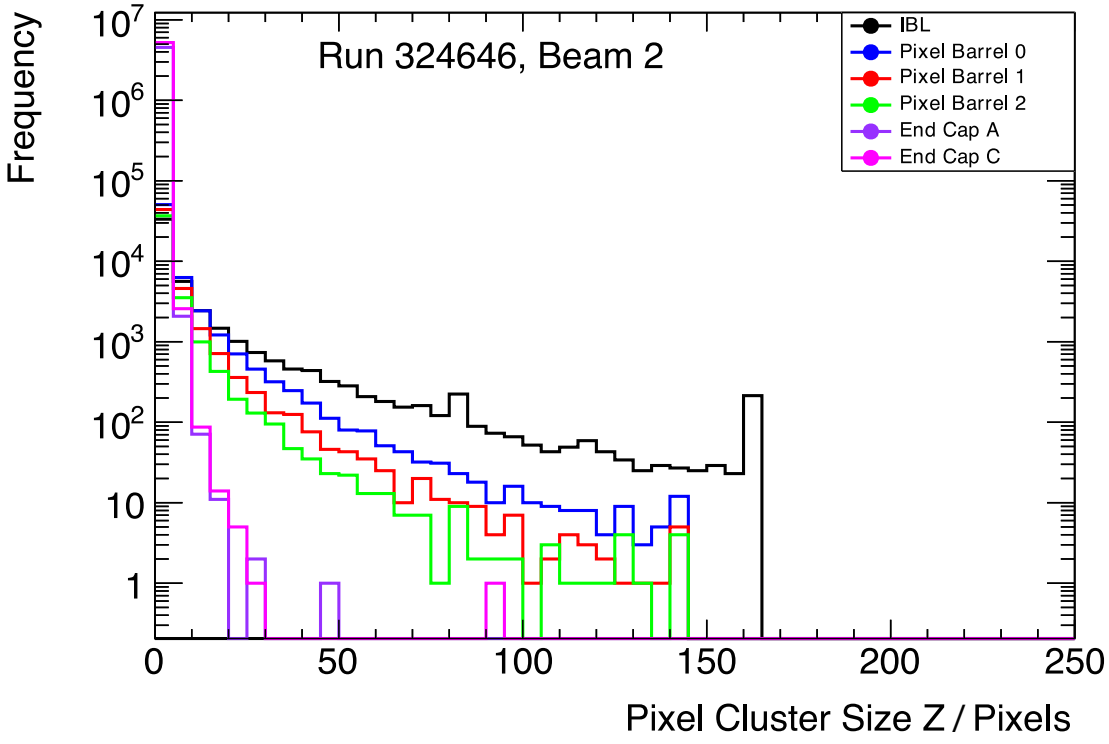
Figure 6.1: Schematic of the four-layer Pixel Detector in ATLAS. Reproduced from [130] (CC BY 3.0).

reduced to $50 \times 250\,\mu\text{m}$ in the IBL. A diagram of the Pixel Detector including a schematic showing the different radii of the detector layers is shown in Figure 6.1. The purpose of the Pixel Detector is to track charged particles originating from collisions at the IP. The Pixel Detector also serves to identify displaced vertices due to long lived particles, which do not originate from the original interaction point. Due to the geometry of the detector, particles originating from the IP will leave hits in four layers, that can be reconstructed as a track. As particles from the IP will tend to pass through the detector layers at steep angles, the pixel *clusters* (groups of pixel hits considered to be originating from the same particle) will tend to consist of a compact, small number of pixels. In contrast, particles associated with beam-induced backgrounds, typically travelling parallel to the axis of the beampipe, will leave long clusters in the barrel pixel layers (in the end-cap disks, the pixel modules are perpendicular to the incident beam background, so the clusters are much smaller) [10, 118]. These long pixel clusters are shown in Figure 6.2. The difference in length between clusters in the barrels and end-caps from beam-induced backgrounds (BIB) events is clearly visible.

To help detect BIB, bunch crossings are split into unpaired and paired bunches. Collisions at the IP will occur in paired bunches, whereas in unpaired bunches there is no corresponding bunch in the other beam, meaning that there will be no collisions at the IP. The unpaired bunch crossings therefore give a much cleaner sample of beam-induced backgrounds. Furthermore, there's an additional qualifier for unpaired bunches, that of being *isolated*. Unpaired isolated bunches are those in which there is a bunch in only one LHC beam with no bunch in the other beam within ± 3 bunch



(a) Pixel cluster length in z.



(b) Pixel cluster lengths in the z direction for beam 2.

Figure 6.2: Pixel cluster lengths in z (parallel to the barrel) for beam 1 and beam 2 unpaired isolated bunches. The clusters are much longer in length for events from BIB than collisions.

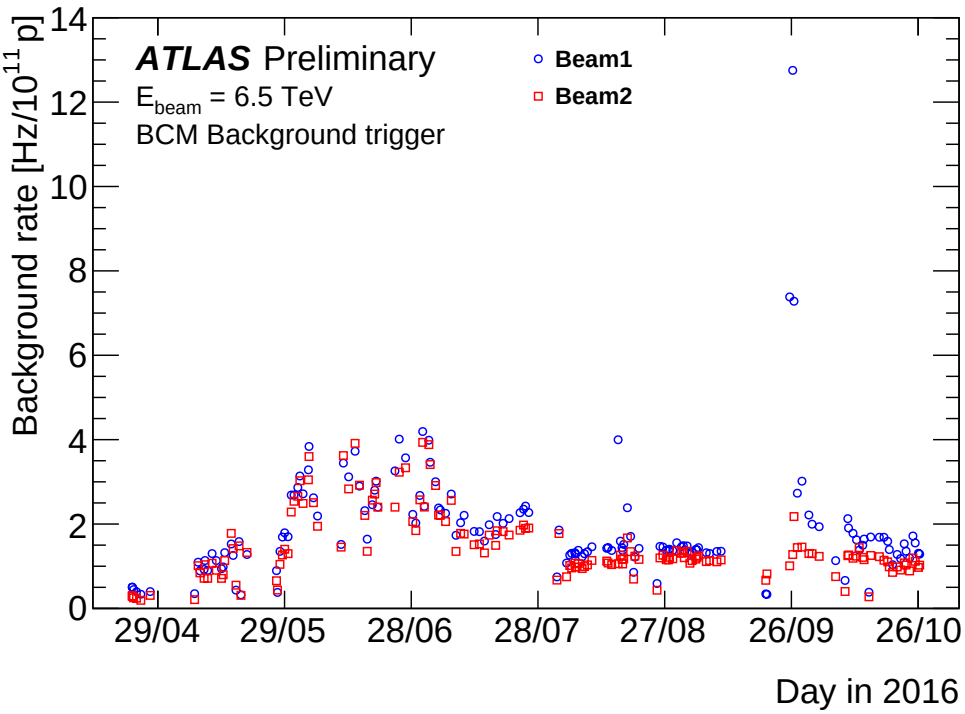


Figure 6.3: BCM background trigger rates for ATLAS runs in 2016 [133]. The rates are normalised by the number of bunches, so that with a number of bunches totalling around 2,000, the triggered background rate in 2016 was on the order of 2 kHz.

crossing identifierss (BCIDs). There are a number of triggers used to detect BIB in the ATLAS detector, a number of which rely on the BCM. The BCM consists of two sets of four modules located around the IP at ±184 cm and 5.5 cm [131, 132]. Due to their proximity to the IP and the Pixel Detector, as well as their high time resolution, they are well-situated to detect BIB in this region. The AC CA triggers are the most pertinent relating to the Pixel Detector, and rely on timing information to detect BIB. If a Beam Conditions Monitor (BCM) on one side records a hit which is out of time with hits from collisions, and a hit is subsequently recorded in the farther BCM in time with collisions, then this is characteristic of a BIB event and is recorded as such. The AC and CA suffixes refer to which side the early and in time hits are on.[1] AC means that the early BCM hit is in the incoming beam 1 side, with the in time hit on the incoming beam 2 side, and vice versa. The BCM rates for 2016 are shown in Figure 6.3.

[1]The ATLAS convention is that "A"-side refers to incoming beam 1, and "C"-side refers to incoming beam 2.

6.2 ATLAS Pressure bump runs and simulation

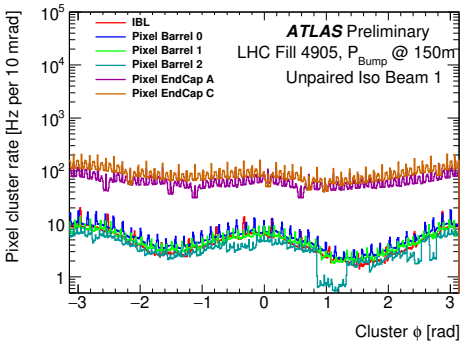
To further study beam-induced backgrounds in ATLAS, in recent years there have been a number of dedicated pressure bump runs. In such runs the pressure is deliberately increased by heating the non-evaporable getters in regions of interest. The operating principle behind these pressure bump runs is to elevate the pressure such that the rate of beam-gas interactions is drastically increased so that beam-gas sources from different regions and the influence of the fields and material in between those regions and the detector can be studied.

During LHC Fill 4905 the pressure in three regions, at 19 m, 58 m and 150 m was elevated each in turn whilst data was continued to be recorded. Figure 6.4 shows the pixel cluster azimuthal positions, ϕ, for beam 1 and beam 2 resulting from each of these pressure bumps. The distributions are the same for both beams, and the dependency on beam-gas interaction location is clear, and very similar to the distributions seen at the interface plane in the previous chapter, as expected. The fine structure is a result of overlapping modules and ganged pixels, with the broader dips due to a few inoperable modules in some of the layers.

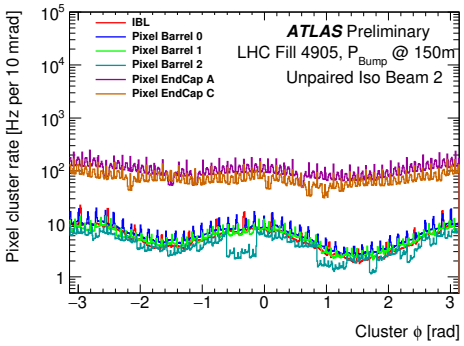
Figures 6.4a and 6.4b show the ϕ distributions just upstream of the separation dipole around the TCTs. Also, the effect of the separation dipoles in smearing the clusters transversely, resulting in peaks at $\phi = 0$, $\phi = 2\pi$, is clear. The 58 m pressure bump is situated just at the start of the inner triplet. Figures 6.4c and 6.4d show an up down asymmetry ($\phi = \pm\frac{\pi}{2}$) which is thought to originate from the crossing angle. At the start of the inner triplet, the closed orbit is still increasing in vertical distance from the centre of the beampipe and quadrupoles, peaking at around 6 mm. The final focus quadrupoles are then effectively used as combined function quadrupoles to bend the beam back down towards the centre, at the IP. Products from beam-gas interactions originating from this region will be bent more by the fields due to their decreased magnetic rigidities, perhaps explaining the excess at $-y$, and deficit at $+y$.

The last pressure bump was at 19 m, which is beyond the interface plane, and is shown in Figures 6.4e and 6.4f. Due to the absence of magnetic field transverse to the beamline from this region onward, the pixel cluster ϕ distributions are totally flat.

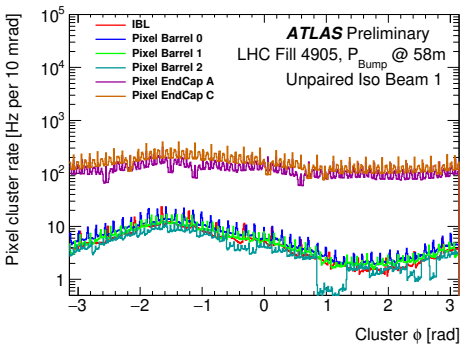
The pixel cluster multiplicity for a given event refers to the number of pixel clusters in



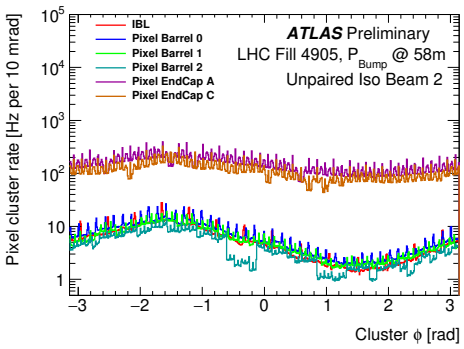
(a) Beam 1, pressure bump at 150 m.



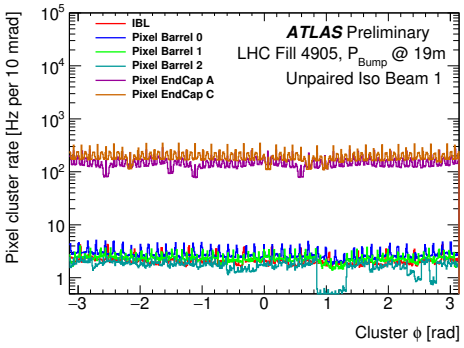
(b) Beam 2, pressure bump at 150 m.



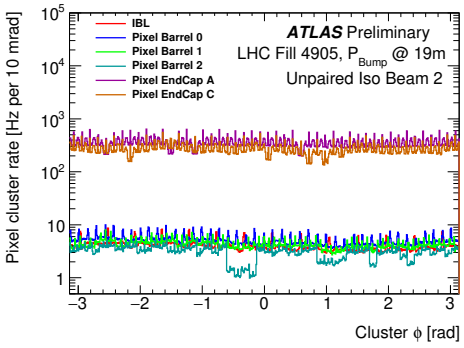
(c) Beam 1, pressure bump at 58 m.



(d) Beam 2, pressure bump at 58 m.



(e) Beam 1, pressure bump at 19 m.



(f) Beam 2, pressure bump at 19 m.

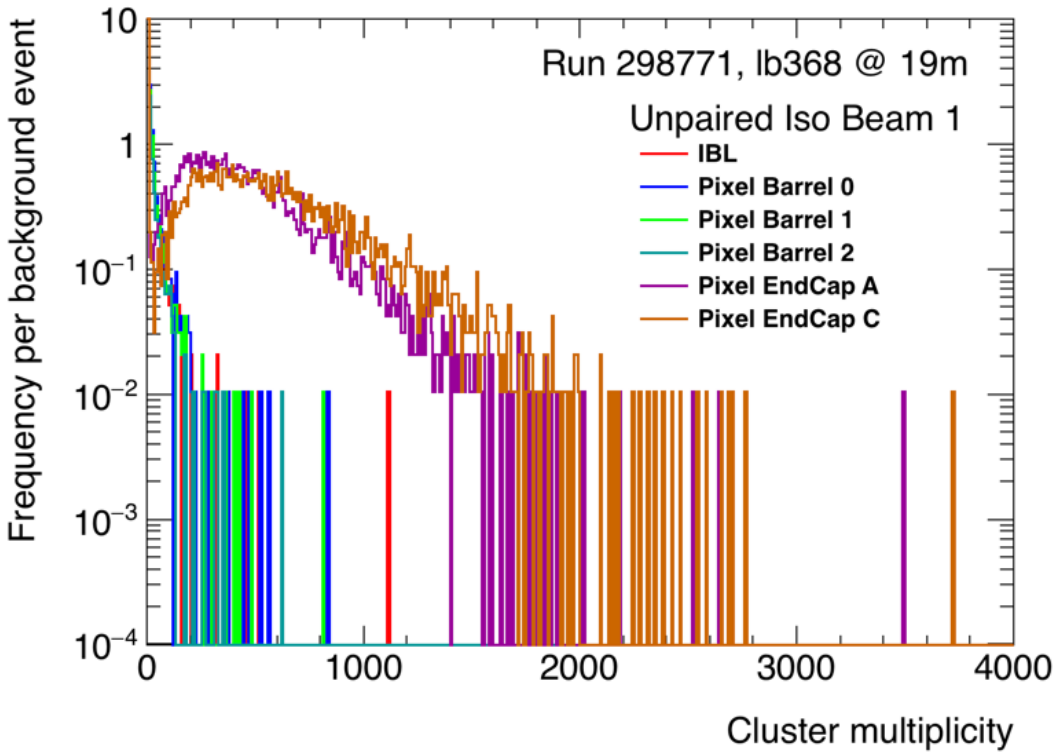
Figure 6.4: Pixel cluster distributions from measured data taken during the pressure bump run for LHC Fill 4905 (ATLAS run 298771) in 2016. These distributions are only for unpaired isolated bunches and are normalised with respect to the luminosity.

the Pixel Detector that are recorded for that event. Figure 6.5 shows the multiplicities for two pressure bumps, again from LHC Fill 4905 (the same data as used for Figure 6.4), at 19 m and 150 m. A number of features and differences between the pair of distributions are apparent. The events resulting from the 150 m pressure bump have a broadly larger cluster multiplicity in the end caps. This is most likely explained by the fact that products from events further upstream are more likely to hit both end caps. Furthermore, it could be an artefact of plotting on-track clusters and is related to the likelihood fitting of BIB end-cap clusters with a fake track that is forced to originate from the ATLAS interaction point (IP) by the track reconstruction. That is to say that end cap clusters from BIB are more likely to be mistaken for clusters caused by products from the IP. Furthermore, a high momentum proton hitting a low energy gas molecule will result in products tending to go downstream along the barrel at very small angles. Far from the IP at 150 m, this is conducive to hits in the Pixel Detector, but much closer at 19 m, more particles will tend to travel along the beamline and miss the Pixel Detector barrels entirely.

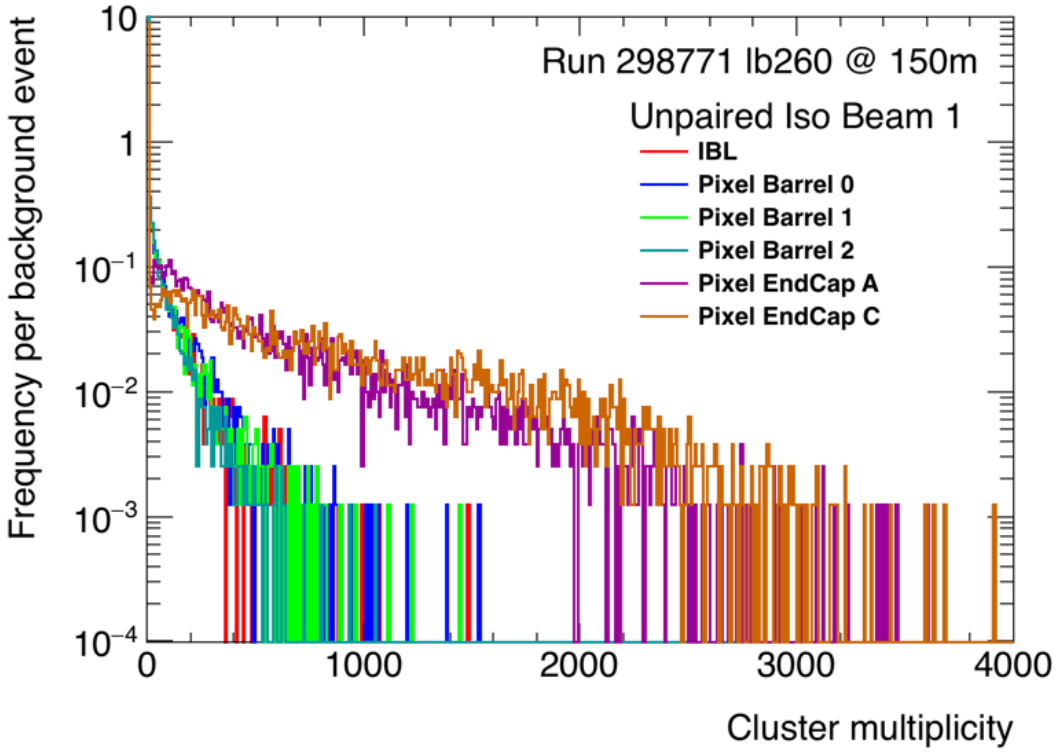
6.2.1 Simulation of beam-induced backgrounds in the ATLAS detector

In Chapter 5 the simulation of beam-gas interactions upstream of the ATLAS detector in pressure bumps using BDSIM was described and particle distributions at the interface plane 22.6 m upstream of the IP were shown. These particle distributions generated with BDSIM were then passed through the official G4 ATLAS model. Preliminary results from this effort are presented here.

Generating simulated Geant4 ATLAS detector response from BDSIM results is a multi-stage process, first translating ASCII input files at the interface plane to HepMC [128] using BeamHaloGenerator [122], and then passing these HepMC files to the ATLAS Geant4 simulation. The ATLAS Geant4 simulation then produces hits that can then be translated using digitisation into simulated detector signals. Finally these simulated signals are reconstructed into particle tracks and identified with particle species for the physics analyses. An official ATLAS production of the Monte Carlo simulation has so far run no further than the Geant4 simulation, resulting in hits in the



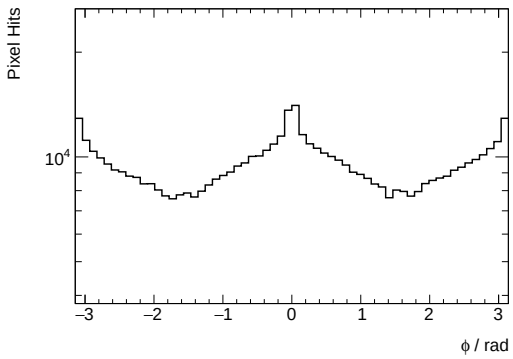
(a) Pixel cluster multiplicity for BIB events in the Pixel Detector with a pressure bump at 19 m.



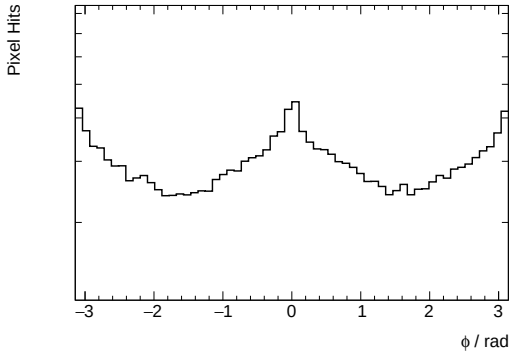
(b) Pixel cluster multiplicity for BIB events in the Pixel Detector with a pressure bump at 150 m.

Figure 6.5: Pixel cluster multiplicity distributions for two different pressure bumps, one at 19 m and the other at 150 m.

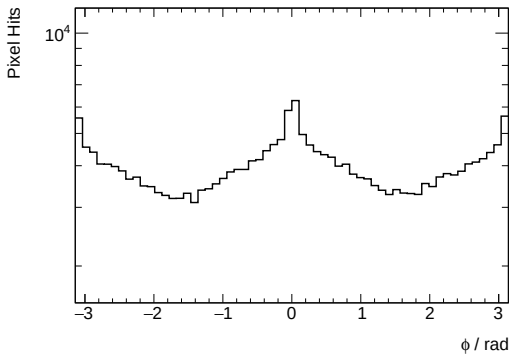
Pixel Detector model. The distribution of hits in ϕ shown with 20 MeV kinetic energy cuts for the three different sources are shown in Figure 6.6. The same smearing in the x plane is visible here as was shown in Figures 6.4a and 6.4b. However, the spikes in all three cases are all at $0, \pm\pi$, features which are more common in the muon distributions, as seen for example in Figure 5.11. This suggests that the muons dominate the simulated hits in the Pixel Detector. However, normalising with a pressure map may further drastically affect these results.



(a) Azimuthal ϕ for simulated ATLAS pixel hits for products originating from BDSIM beam-gas interactions in a pressure bump 10 m in length centred at 58 m.



(b) Azimuthal ϕ for simulated ATLAS pixel hits for products originating from BDSIM beam-gas interactions in a pressure bump 10 m in length centred at 148 m.



(c) Azimuthal ϕ for simulated ATLAS pixel hits for products originating from BDSIM beam-gas interactions 550 m region upstream of the interface plane.

Figure 6.6: Simulated pixel hit azimuthal ϕ in the ATLAS Geant4 model for secondary particles originating from beam-gas events in three different regions of the BDSIM IR1 model.

6.3 Conclusion

In this chapter data from a recent pressure bump run were qualitatively compared with early results from ATLAS Monte Carlo simulations of the BIB simulations presented in the earlier chapter. The influence of the accelerator optics that was suggested in the previous chapter was clearly demonstrated here with the use of data from the pressure bump run.

Work is ongoing in generating a full set of Monte Carlo data of ATLAS Pixel Detector hits at the reconstruction level for the six total scenarios (kinetic energy cuts of 20 MeV and 20 GeV, and beam-gas sources from 58 m, 148 m and the whole region upstream) for further analysis and comparison with pressure bump data. Furthermore, a Run II pressure map for IR1 should be applied to the ATLAS Monte Carlo data, although at the time of writing such a pressure map does not exist. Also, a tagger for robustly identifying BIB in the Pixel Detector has been written, with the ultimate aim of using it for the online detection of such backgrounds.

CHAPTER

SEVEN

CONCLUSION

In this thesis, the LHC, its collimation system, and beam-induced backgrounds in the ATLAS detector have been studied with the novel accelerator code BDSIM, and compared with real data. Whilst BDSIM has been used for LHC collimation studies in the past, these applications have not been examined in detail. Furthermore, BDSIM has never before been used to simulated beam-induced backgrounds (BIB).

In Chapter 3, a new Python package, pyfluka, for robustly translating geometries from FLUKA to GDML for use in BDSIM was described and demonstrated. This powerful package is useful because of the large array of existing detailed geometries which have been built over many years, particularly for the LHC, HL-LHC, and other accelerators at CERN. This was shown to great effect with the translation of the IR1 tunnel geometry from FLUKA to GDML, where an extremely complex design was translated. With features spanning from the km-range all the way down to the mm-range, accruing only a small number of errors on the level of 0.01 % by volume is a resounding success. In principle, pyfluka allows one to take an existing FLUKA simulation and then perform it again in BDSIM or Geant4. Furthermore, the FLUKA graphical interface, flair, allows one to easily design and build CSG with very similar approach as used in typical CAD packages. However, typical CAD software used in engineering will produce geometry, if loaded into a physics code, which consists of a mesh of many thousands of facets. In contrast, as flair writes CSG, it will tend to produce geometry much more optimised for particle physics simulations.

Additionally, it is worth noting that expressing CSG purely graphically in flair is in stark contrast to BDSIM where one must express it purely in code. Now, with the advent of pyfluka, one can use flair to graphically build efficient geometries and then use them in BDSIM. This newly available workflow was used to design several geometries, including the TAN, in the IR1 simulations of Chapter 5.

Work is ongoing in integrating pyfluka into the broader Python-GDML interface, pyg4ometry. The feature set of pyfluka is extensive, but missing one last capability, and that is translating LATTICE commands to GDML. In short, these are arbitrary combined rotations and translations, which are used widely in FLUKA codes to optimise their geometries. This will make the translation of some repetitive geometries easier. Lastly, a translator from GDML to FLUKA is being written as part of pyg4ometry.

In Chapter 4 BDSIM was used to study the LHC collimation system. A detailed model of the LHC including an accurate aperture model, magnet geometries, and placed BLMs was built in BDSIM. BDSIM's tracking was shown to not be symplectic and the corresponding increase in the impact parameter over the duration of a typical loss map study was shown. A mitigation strategy in the form of a one turn map was introduced and demonstrated as a viable method for accounting for the errors in BDSIM's tracking, albeit with further work needed to account for possible particle-matter interactions. Furthermore, the source of the error in the tracking was shown to be fundamental to the Geant4 tracking algorithms themselves and therefore unrelated to the choice of component maps. Whilst the impact parameter growth could plausibly result in an artificially increased collimation performance, previous studies have shown that [103] the impact parameter in LHC proton collimation has a negligible effect on the resulting collimation performance.

The BDSIM LHC model was validated and compared with both SixTrack and beam loss monitor (BLM) data from a recent qualification loss map. Good agreement with SixTrack was shown by reproducing both the cleaning hierarchy and the losses in the downstream dispersion suppressor. However, excess losses in IR8 and IR1 due to substantial tertiary collimator misalignments were also clearly apparent and has since been corrected for future studies. Additionally, the primary and secondary collimator

materials were erroneously set to graphite instead of CFC, presenting a further source of systematic error, which has also been corrected for future studies of the LHC. The simulated BLM energy deposition was compared with the measured dose and strong correlation was clear. As the BLM was simply modelled with a solid aluminium cylinder, to achieve greater agreement with the loss map data will likely involve building a more accurate model of the BLMs.

Whilst the impact parameter does not affect the collimation system performance for protons, it does in ion collimation. For this reason if BDSIM is to be applied to ion collimation it is likely that a separate, fully symplectic tracker will need to be written. As the rate of emittance growth increases with the number of boundaries, it will be worse when simulating the Future Circular Collider (FCC), perhaps prohibitively so. Implementing a tracker therefore will allow one to robustly simulate the FCC. Lastly, a single LHC collimation simulation can take up to two weeks to generate, meaning that increasing the speed of simulation will be extremely useful.

In Chapter 5, the design of a detailed model of IR1 for use in BDSIM was described and then applied to the study of BIB. Two of the modelled scenarios focused on beam-gas interactions, one from the whole region upstream of interface plane, and another situated upstream in two 10 m regions corresponding to pressure bumps. Simulating the pressure bumps can be particularly insightful as there have been a number of runs in recent years in which the pressure was deliberately inflated in these regions to increase the rate of beam-gas interactions. The third scenario involved simulating the passage of the debris from the beam halo impacting on the tertiary collimators positioned 150 m upstream of the IP. The means by which beam-gas events could be sampled uniformly in s in the region upstream of the interface plane using cross section biasing was described. The effects of the crossing angle, final focus quadrupoles, and the separation dipoles on the azimuthal asymmetry seen at the interface plane, were demonstrated. Additionally, the impact of the arc screening particles such that only muons and on-energy protons reach the interface plane was clearly demonstrated. Two different kinetic energy cuts were used, one to exclude particles which aren't energetic enough to pass through the TAS into the detector, and a higher cut for specifically targeting high-energy muons, which are particularly

important due to the long tracks and energy deposits left in the detector.
All of the BDSIM simulations had existing publicly available FLUKA simulations of the same scenarios with which to compare. Good agreement between two codes was apparent, with the most apparent feature being a relative excess in the number of on-energy protons, which came at the cost of a smaller rate in the secondary particles of all other types. The physics of the beam-gas interaction was compared with that of two Monte Carlo event generators, EPOS LHC and DPMJET-III, and the kinetic energy distributions from the event generators seemed to be more consistent with the results from FLUKA than BDSIM. In future simulations of BIB it would likely be best to use one of these event generators and then insert the products directly into the machine, a feature which BDSIM does support. This will give better agreement with existing codes, and allow greater fine tuning of the interaction itself.
Another important step that should be taken is to use a pressure map to reweight the particle distributions. This is particularly important for the understanding of the sources of different features at the interface plane from within the machine upstream. At the time of writing there are no existing pressure maps for Run II, so such a comparison was not possible. In place of this the relative contributions from various points assuming a flat profile demonstrated that the varying pressure will affect the distribution in the particle flux at the interface plane considerably.
The model accuracy could also be further improved. The tunnel model was extremely detailed, but apart from this, the model was dependent on generic model components for all its magnets. Additionally, generic, ideal magnetic fields were used for all of the magnetic elements. Due to the substantial effect the magnetic fields were shown to have on the BIB flux distributions, using exact field maps may be necessary to attain greater accuracy. Investing time in acquiring both more accurate component geometries and field maps will likely be worthwhile due to the fact the LHC, in some form, will run until 2038. Whilst the components and fields will change over this time (most notably for the High Luminosity Large Hadron Collider (HL-LHC)), they will nevertheless be much more representative of reality than BDSIM's generic components and fields. These would not only be usable for studies of BIB in CMS and ATLAS, but other experiments located in these regions, such as FASER [134]. One possibility would

be to use pyfluka to convert the set of existing highly-detailed geometries from FLUKA for use in BDSIM, although these are not publicly available and would be dependent on collaboration with the FLUKA and Radiation Protection teams at CERN, who built and are responsible for the models, which may not be realisable.

In Chapter 6 BIB in the Pixel Detector were introduced and described. The characteristic features of such backgrounds, particularly that they tend to leave far longer tracks in the barrel layers than in comparison with collision products was demonstrated. Furthermore, the azimuthal distribution and its relationship with the magnetic fields situated between the beam-gas interaction point and the detector was demonstrated. This was done with particular reference to recent pressure bump runs in which the rate of BIB was deliberately inflated in a region upstream of the ATLAS detector, as this provides a particularly clean source of such backgrounds. The distributions at the interface plane in Chapter 5 were passed through a Geant4 simulation to generate hits in the Pixel Detector, and their distributions were discussed. Proposed future work includes a full comparison of the reconstructed hits with pressure bump data.

In summary, BDSIM has been shown to be highly effective in simulating LHC collimation and backgrounds. Detailed models and tools have been developed to enable such studies. The LHC will be running for at least another 15 years and the work presented here provides a foundation which can be built on to provide further, more developed studies.

ACRONYMS AND INITIALISMS

AST abstract syntax tree. 56

BCID bunch crossing identifiers. 160

BCM Beam Conditions Monitor. 17, 162

BDSIM Beam Delivery Simulation. 24

BIB beam-induced backgrounds. 16, 18, 22, 23, 25, 26, 39, 55, 89, 119, 120, 125, 129, 135, 138, 143, 147, 149, 150, 156, 157, 160–162, 165, 166, 169, 170, 172–174

BLM beam loss monitor. 21, 25, 67, 72, 75, 77, 104–106, 108, 110, 171, 172

CAD computer aided design. 53, 55

CFC carbon fibre composite. 68, 96, 97, 107

CLIC Compact Linear Collider. 79

CRMC Cosmic Ray Monte Carlo. 142

CSG Constructive solid geometry. 51–56, 58, 59

DS dispersion suppressor. 69, 103, 107, 108, 110, 116, 118

FCC Future Circular Collider. 42, 172

FEDB FLUKA Element Database. 43, 44, 65, 129, 157

FODO focussing-defocussing. 29, 147

GDML Geometry Description Markup Language. 12, 48, 54–56, 58, 59, 61, 62, 65, 125, 128, 129, 157

GUI graphical user interface. 55

HE-LHC High Energy Large Hadron Collider. 43

HEPMC High Energy Physics Monte Carlo. 156

HL-LHC High Luminosity Large Hadron Collider. 22, 42, 120, 156, 157, 173

IBL Insertable B-Layer. 159, 160

ILC International Linear Collider. 79

IP interaction point. 15, 22, 23, 27, 36, 66, 119–122, 125, 127, 130, 131, 135, 138, 165

IR insertion region. 20, 27, 67, 152

LHC Large Hadron Collider. 11, 13, 19–22, 25, 27, 39, 42–45, 50, 51, 55, 56, 65–69, 71–75, 79, 80, 85, 98, 104, 119–121, 125, 126, 142, 143, 145

LS2 Long Shutdown 2. 21

OTM one turn map. 82, 84, 86, 87, 115, 118

PTC Polymorphic Tracking Code. 42

RHIC Relativistic Heavy Ion Collider. 42

SRF superconducting radio frequency. 20, 48

TAN neutral absorber. 153

TCLA tungsten absorber. 68, 69

TCP primary collimator. 14, 68, 78, 96, 108, 110

TCSG secondary collimator. 68, 77

TCT tertiary collimator. 22, 68, 78, 89, 98, 99, 103, 115, 120, 122, 152, 153

TFS Table File System. 42

BIBLIOGRAPHY

[1] ATLAS Collaboration, Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC, Physics Letters B 716 (1) (2012) 1–29. arXiv:1207.7214, doi:10.1016/j.physletb.2012.08.020.

[2] S. Chatrchyan, V. Khachatryan, A. Sirunyan, A. Tumasyan, W. Adam, E. Aguilo, T. Bergauer, M. Dragicevic, J. Erö, C. Fabjan, et al., Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC, Physics Letters B 716 (1) (2012) 30–61. doi:10.1016/j.physletb.2012.08.021.
URL http://dx.doi.org/10.1016/j.physletb.2012.08.021

[3] R. Schmidt, R. Assmann, E. Carlier, B. Dehning, R. Denz, B. Goddard, E. B. Holzer, V. Kain, B. Puccio, B. Todd, J. Uythoven, J. Wenninger, M. Zerlauth, Protection of the CERN Large Hadron Collider, New Journal of Physics 8 (11) (2006) 290. doi:10.1088/1367-2630/8/11/290.

[4] R. Brinkmann, HERA status and plans, Conf. Proc. C950501 (1996) 406–410.

[5] R. S. Moore, Tevatron Collider Status and Prospects, in: Particles and fields. Proceedings, Meeting of the Division of the American Physical Society, DPF 2009, Detroit, USA, July 26-31, 2009, 2009. arXiv:0910.3612.
URL http://lss.fnal.gov/cgi-bin/find_paper.pl?conf-09-492

[6] ATLAS Collaboration, The ATLAS Experiment at the CERN Large Hadron Collider, JINST 3 (2008) S08003. doi:10.1088/1748-0221/3/08/S08003.

[7] S. Chatrchyan, et al., The CMS Experiment at the CERN LHC, JINST 3 (2008) S08004. doi:10.1088/1748-0221/3/08/S08004.

[8] K. Aamodt, et al., The ALICE experiment at the CERN LHC, JINST 3 (2008) S08002. doi:10.1088/1748-0221/3/08/S08002.

[9] A. A. Alves, Jr., et al., The LHCb Detector at the LHC, JINST 3 (2008) S08005. doi:10.1088/1748-0221/3/08/S08005.

[10] ATLAS Collaboration, Beam-induced and cosmic-ray backgrounds observed in the ATLAS detector during the LHC 2012 proton-proton running period, Journal of Instrumentation 11 (5) (2016) P05013. arXiv:1603.09202, doi:10.1088/1748-0221/11/05/P05013.

[11] R. Bruce, LHC machine status report, https://indico.cern.ch/event/744723/contributions/3077440/attachments/1714182/2764924/2018.09.12_LHCC_LHC_machine_status.pdf, [Online; accessed 16-October-2019] (2018).

[12] L. Evans, P. Bryant, LHC Machine, JINST 3 (2008) S08001. doi:10.1088/1748-0221/3/08/S08001.

[13] E. B. Holzer, B. Dehning, E. Effinger, J. Emery, G. Ferioli, J. L. González, E. Gschwendtner, G. Guaglio, M. Hodgson, D. Kramer, R. Leitner, L. Ponce, V. Prieto, M. Stockner, C. Zamantzas, Beam Loss Monitoring System for the LHC, Tech. Rep. CERN-AB-2006-009, CERN, Geneva (Sep 2005). URL http://cds.cern.ch/record/930275

[14] G. Apollinari, I. Béjar Alonso, O. Brüning, M. Lamont, L. Rossi, High-Luminosity Large Hadron Collider (HL-LHC): Preliminary Design Report, CERN Yellow Reports: Monographs, CERN, Geneva, 2015. doi:10.5170/CERN-2015-005. URL https://cds.cern.ch/record/2116337

[15] R. Kwee-Hinzmann, G. Bregliozzi, F. Cerutti, S. Gibson, C. Yin Vallgren, L. S. Esposito, R. Bruce, H. Garcia Morales, A. Lechner, Machine-induced background simulation studies for LHC Run 1, Run 2 and HL-LHC, Tech. rep., CERN (2017).

[16] R. Bruce, R. Assmann, V. Boccone, G. Bregliozzi, H. Burkhardt, F. Cerutti, A. Ferrari, M. Huhtinen, A. Lechner, Y. Levinsen, et al., Sources of machine-induced background in the ATLAS and CMS detectors at the CERN Large Hadron Collider, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 729 (2013) 825–840.

[17] A. Drozhdin, N. Mokhov, S. Striganov, Beam losses and background loads on collider detectors due to beam-gas interactions in the LHC, Tech. rep., Fermilab (2010).

[18] L. Nevay, J. Snuverink, A. Abramov, L. Deacon, H. Garcia-Morales, S. Gibson, R. Kwee-Hinzmann, H. Pikhartova, W. Shields, S. Walker, et al., BDSIM: An accelerator tracking code with particle-matter interactions, arXiv preprint arXiv:1808.10745.

[19] G. A. Blair, Simulation of the CLIC beam delivery system using BDSIM, Tech. Rep. CERN-OPEN-2002-057. CERN-PS-RF-Note-2002-072. CLIC-Note-509, CERN, Geneva (Sep 2002).
URL http://cds.cern.ch/record/582181

[20] L. Nevay, S. Boogert, R. Bruce, H. Garcia, S. Gibson, R. Kwee-Hinzmann, S. Redaelli, LHC Collimation and Energy Deposition Studies Using Beam Delivery Simulation (BDSIM) (CERN-ACC-2016-190) (2016) WEPOY047. 3 p. doi:10.18429/JACoW-IPAC2016-WEPOY047.
URL http://cds.cern.ch/record/2207433

[21] A. Abramov, S. Boogert, L. Nevay, S. Walker, First Studies of Ion Collimation for the LHC Using BDSIM, in: Proceedings, 9th International Particle Accelerator Conference (IPAC 2018): Vancouver, BC Canada, April 29-May 4, 2018, 2018, p. MOPMF090. doi:10.18429/JACoW-IPAC2018-MOPMF090.

[22] S. Walker, A. Abramov, S. Boogert, H. Garcia Morales, S. Gibson, L. Nevay, H. Pikhartova, W. Shields, A Holistic Approach to Simulating Beam Losses in the Large Hadron Collider Using BDSIM, in: Proceedings, 13th International

Computational Accelerator Physics Conference, ICAP2018: Key West, FL, USA, 20-24 October 2018, 2019, p. TUPAF15.
doi:10.18429/JACoW-ICAP2018-TUPAF15.

[23] S. Walker, A. Abramov, S. Boogert, S. Gibson, L. Nevay, H. Pikhartova, Precision Modelling of Energy Deposition in the LHC using BDSIM, in: Proceedings, 10th International Particle Accelerator Conference (IPAC2019): Melbourne, Australia, May 19-24, 2019, 2019, p. MOPRB064.
doi:10.18429/JACoW-IPAC2019-MOPRB064.

[24] S. Gibson, S. Alden, L. Nevay, Laser Sculpted Cool Proton Beams, in: Proceedings, 10th International Particle Accelerator Conference (IPAC2019): Melbourne, Australia, May 19-24, 2019, 2019, p. MOPRB116.
doi:10.18429/JACoW-IPAC2019-MOPRB116.

[25] T. Böhlen, F. Cerutti, M. Chin, A. Fassò, A. Ferrari, P. Ortega, A. Mairani, P. R. Sala, G. Smirnov, V. Vlachoudis, The FLUKA code: developments and challenges for high energy and medical applications, Nuclear data sheets 120 (2014) 211–214.

[26] W. Herr, B. Muratori, Concept of luminositydoi:10.5170/CERN-2006-002.361.
URL https://cds.cern.ch/record/941318

[27] A. W. Chao, K. H. Mess, M. Tigner, F. Zimmermann (Eds.), Handbook of Accelerator Physics and Engineering, 2nd Edition, World Scientific, 2013, Ch. 1, pp. 6–7.

[28] A. Wolski, Beam Dynamics in High Energy Particle Accelerators, 2014.
doi:10.1142/p899.

[29] J. B. Jeanneret, Optics of a two-stage collimation system, Phys. Rev. Spec. Top. Accel. Beams 1 (LHC-Project-Report-243. CERN-LHC-Project-Report-243) (1998) 081001. 13 p. doi:10.1103/PhysRevSTAB.1.081001.
URL https://cds.cern.ch/record/368725

[30] P. J. Bryant, E. Klein, The design of betatron and momentum collimation systems, Tech. Rep. CERN-SL-92-40-AP, CERN, Geneva (Aug 1992).
URL https://cds.cern.ch/record/241140

[31] V. Biryukov, V. Chepegin, Y. Chesnokov, V. Guidi, W. Scandale, Crystal collimation as an option for the large hadron colliders, Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms 234 (1) (2005) 23 – 30, relativistic Channeling and Related Coherent Phenomena in Strong Fields.
doi:https://doi.org/10.1016/j.nimb.2005.03.003.
URL
http://www.sciencedirect.com/science/article/pii/S0168583X05002284

[32] A. G. Afonin, V. T. Baranov, V. M. Biryukov, M. B. H. Breese, V. N. Chepegin, Y. A. Chesnokov, V. Guidi, Y. M. Ivanov, V. I. Kotov, G. Martinelli, W. Scandale, M. Stefancich, V. I. Terekhov, D. Trbojevic, E. F. Troyanov, D. Vincenzi, High-efficiency beam extraction and collimation using channeling in very short bent crystals, Phys. Rev. Lett. 87 (2001) 094802.
doi:10.1103/PhysRevLett.87.094802.
URL https://link.aps.org/doi/10.1103/PhysRevLett.87.094802

[33] G. Stancari, A. Valishev, G. Annala, G. Kuznetsov, V. Shiltsev, D. A. Still, L. G. Vorobiev, Collimation with hollow electron beams, Phys. Rev. Lett. 107 (2011) 084802. doi:10.1103/PhysRevLett.107.084802.
URL https://link.aps.org/doi/10.1103/PhysRevLett.107.084802

[34] H. Grote, F. Schmidt, MAD-X—an upgrade from MAD8, in: Proceedings of the 2003 Particle Accelerator Conference, Vol. 5, IEEE, 2003, pp. 3497–3499.

[35] E. Forest, F. Schmidt, E. McIntosh, Introduction to the Polymorphic Tracking Code, KEK report 3 (2002) 2002.

[36] F. Schmidt, MAD-X PTC integration, in: Proceedings of the 2005 Particle Accelerator Conference, IEEE, 2005, pp. 1272–1274.

[37] F. Schmidt, SIXTRACK: version 1, single particle tracking code treating transverse motion with synchrotron oscillations in a symplectic manner, Tech. rep., CM-P00049314 (1990).

[38] R. De Maria, et al., SixTrack Project: Status, Runtime Environment, and New Developments, in: Proceedings, 13th International Computational Accelerator Physics Conference, ICAP2018: Key West, FL, USA, 20-24 October 2018, 2019, p. TUPAF02. doi:10.18429/JACoW-ICAP2018-TUPAF02.

[39] G. Robert-Démolaize, A. Drees, Loss maps of RHICdoi:10.5170/CERN-2008-005.94.
URL https://cds.cern.ch/record/1133364

[40] G. Robert-Demolaize, R. Assmann, S. Redaelli, F. Schmidt, A new version of SixTrack with collimation and aperture interface, in: Proceedings of the 2005 Particle Accelerator Conference, IEEE, 2005, pp. 4084–4086.

[41] J. Molson, P. Bambade, S. Chancé, A. Faus-Golfe, Simulation of the FCC-hh collimation system (2016) TUPMR054. 3
pdoi:10.18429/JACoW-IPAC2016-TUPMR054.
URL https://cds.cern.ch/record/2304772

[42] A. Mereghetti, et al., SixTrack for Cleaning Studies: 2017 Updates, in: Proceedings, 8th International Particle Accelerator Conference (IPAC 2017): Copenhagen, Denmark, May 14-19, 2017, 2017, p. THPAB046. doi:10.18429/JACoW-IPAC2017-THPAB046.

[43] A. Abramov, R. Bruce, M. Crouch, N. Fuster-Martínez, A. Mereghetti, J. Molson, L. Nevay, S. Redaelli, Collimation of Heavy-Ion Beams in the HE-LHC, in: Proceedings, 10th International Particle Accelerator Conference (IPAC2019): Melbourne, Australia, May 19-24, 2019, 2019, p. MOPRB059. doi:10.18429/JACoW-IPAC2019-MOPRB059.

[44] T. Trenkler, J. B. Jeanneret, K2, A software package evaluating collimation systems in circular colliders (manual), Tech. Rep. SL-Note-94-105-AP, CERN, Geneva (Dec 1994).
URL https://cds.cern.ch/record/703539

[45] R. W. Assmann, I. S. Baishev, M. Brugger, M. Hayes, J. B. Jeanneret, V. Kain, D. I. Kaltchev, F. Schmidt, Tools for Predicting Cleaning Efficiency in the

LHC (LHC-Project-Report-639. CERN-LHC-Project-Report-639) (2003) 4 p. URL https://cds.cern.ch/record/619608

[46] G. Battistoni, E. Gadioli, M. Brugger, A. Ferrari, G. Smirnov, V. Vlachoudis, S. Muraro, A. Empl, R. Villari, M. Carboni, et al., The application of the Monte Carlo code FLUKA in radiation protection studies for the large hadron collider, Prog. Nucl. Sci. Tech. 2 (2011) 358–364.

[47] A. Drozhdin, M. Huhtinen, N. Mokhov, Accelerator related background in the CMS detector at LHC, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 381 (2-3) (1996) 531–544.

[48] A. Mereghetti, V. Boccone, F. Cerutti, R. Versaci, V. Vlachoudis, The FLUKA LineBuilder and element database: tools for building complex models of accelerator beam lines, IPAC2012, WEPPD071 (2012) 2687.

[49] A. Mereghetti, M. Meddahi, S. D Pastor, Y. Papaphilippou, R. De Maria, R. Appleby, F. Cerutti, O. Mete, B. Goddard, V. Kain, et al., SixTrack-FLUKA active coupling for the upgrade of the SPS scrapers, in: Conf. Proc., Vol. 130512, 2013, p. WEPEA064.

[50] S. Agostinelli, et al., GEANT4: A Simulation toolkit, Nucl. Instrum. Meth. A506 (2003) 250–303. doi:10.1016/S0168-9002(03)01368-8.

[51] L. Archambault, L. Beaulieu, J. Carrier, F. Castrovillari, S. Chauvie, F. Foppiano, G. Ghiso, S. Guatelli, S. Incerti, E. Lamanna, et al., Overview of geant4 applications in medical physics, in: 2003 IEEE Nuclear Science Symposium. Conference Record (IEEE Cat. No. 03CH37515), Vol. 3, IEEE, 2003, pp. 1743–1745.

[52] D. Sardari, R. Maleki, H. Samavat, A. Esmaeeli, Measurement of depth-dose of linear accelerator and simulation by use of Geant4 computer code, Reports of Practical Oncology & Radiotherapy 15 (3) (2010) 64–68.

[53] G. Borissov, E. Bouhova-Thacker, J. Catmore, S. Cheatham, A. Chilingarov, R. Davidson, L. de Mora, H. Fox, R. Henderson, G. Hughes, et al., The ATLAS

simulation infrastructure, European Physical Journal C: Particles and Fields 70 (3) (2010) 823–874.

[54] R. B. Appleby, H. Burkhardt, G. Corti, Y. I. Levinsen, M. H. Lieng, V. Talanov, Simulation of machine induced background in the LHCb experiment: methodology and implementation, IEEE Transactions on Nuclear Science 59 (4) (2012) 1681–1694.

[55] N. Mokhov, S. Striganov, MARS15 overview, in: AIP Conference Proceedings, Vol. 896, AIP, 2007, pp. 50–60.

[56] L. Azhgirey, I. Baishev, V. Talanov, K. Potter, Machine induced background in the low luminosity insertions of the LHC, Tech. rep. (2002).

[57] A. I. Drozhdin, M. Huhtinen, N. V. Mokhov, Accelerator related background in the CMS detector at LHC, Nucl. Instrum. Methods Phys. Res., A 381 (CERN-TIS-96-008-RP-PP. LHC-Project-Report-8. CERN-TIS-RP-96-08-PP. CERN-LHC-Project-Report-8. 2-3) (1996) 531–544. 20 p, revised version number 1 submitted on 2004-02-16 16:26:14. doi:10.1016/S0168-9002(96)00807-8. URL https://cds.cern.ch/record/304958

[58] T. Abe, I. Adachi, K. Adamczyk, S. Ahn, H. Aihara, K. Akai, M. Aloi, L. Andricek, K. Aoki, Y. Arai, et al., Belle II technical design report, arXiv preprint arXiv:1011.0352.

[59] KEK, Strategic Accelerator Design, http://acc-physics.kek.jp/SAD/index.html, [Online; accessed 8-October-2019] (2019).

[60] D. C. Carey, K. L. Brown, F. C. Iselin, Decay TURTLE (Trace Unlimited Rays Through Lumped Elements): A Computer Program for Simulating Charged Particle Beam Transport Systems, Including Decay Calculations.

[61] H. Nakano, H. Yamamoto, K. Kanazawa, H. Nakayama, Y. Ohnishi, C. Kiesling, S. Koblitz, A. Moll, M. Ritter, Beam background simulation for SuperKEKB/Belle-II, Energy [GeV] 3 (2011) 8–0.

[62] Y. I. Levinsen, Machine Induced Experimental Background Conditions in the LHC, Ph.D. thesis, Oslo U. (2012-02-20).

[63] L. Lönnblad, CLHEP—a project for designing a C++ class library for high energy physics, Computer Physics Communications 84 (1-3) (1994) 307–316. doi:10.1016/0010-4655(94)90217-8.

[64] R. Brun, F. Rademakers, ROOT — an object oriented data analysis framework, Nuclear Instruments and Methods in Physics Research A 389 (1) (1997) 81–86. doi:10.1016/S0168-9002(97)00048-X.

[65] CERN, MAD8 website, http://mad8.web.cern.ch, [Online; accessed 8-October-2019] (2019).

[66] K. L. Brown, F. Rothacker, D. C. Carey, F. C. Iselin, Transport: A Computer Program for Designing Charged Particle Beam Transport Systems.

[67] P. Mora de Freitas, H. Videau, Detector simulation with MOKKA/GEANT4: Present and future, Tech. rep. (2002).

[68] R. Chytracek, J. McCormick, W. Pokorski, G. Santin, Geometry Description Markup Language for physics simulation and analysis applications, IEEE Transactions on Nuclear Science 53 (5) (2006) 2892–2896.

[69] A. Bagulya, et al., Recent progress of GEANT4 electromagnetic physics for LHC and other applications, J. Phys. Conf. Ser. 898 (4) (2017) 042032. doi:10.1088/1742-6596/898/4/042032.

[70] S. Chauvie, S. Guatelli, V. Ivanchenko, F. Longo, A. Mantero, B. Mascialino, P. Nieminen, L. Pandola, S. Parlati, L. Peralta, M. G. Pia, M. Piergentili, P. Rodrigues, S. Saliceti, A. Tnndade, Geant4 low energy electromagnetic physics, in: IEEE Symposium Conference Record Nuclear Science 2004., Vol. 3, 2004, pp. 1881–1885 Vol. 3. doi:10.1109/NSSMIC.2004.1462612.

[71] M. H. Mendenhall, R. A. Weller, A probability-conserving cross-section biasing mechanism for variance reduction in monte carlo particle transport calculations, Nuclear Instruments and Methods in Physics Research Section A: Accelerators,

Spectrometers, Detectors and Associated Equipment 667 (2012) 38–43.
doi:10.1016/j.nima.2011.11.084.
URL http://dx.doi.org/10.1016/j.nima.2011.11.084

[72] P. Cignoni, C. Montani, R. Scopigno, A comparison of mesh simplification algorithms, Computers & Graphics 22 (1) (1998) 37–54.

[73] V. Vlachoudis, et al., FLAIR: a powerful but user friendly graphical interface for FLUKA, in: Proc. Int. Conf. on Mathematics, Computational Methods & Reactor Physics (M&C 2009), Saratoga Springs, New York, Vol. 1, 2009, p. 3.

[74] S. Boogert, A. Abramov, J. Albrecht, G. L. D'Alessandro, L. Nevay, W. Shields, S. Walker, Pyg4ometry: A tool to create geometries for Geant4, BDSIM, G4Beamline and FLUKA for particle loss and energy deposit studies, in: 10th Int. Partile Accelerator Conf.(IPAC'19), Melbourne, Australia, 19-24 May 2019, JACOW Publishing, Geneva, Switzerland, 2019, pp. 3244–3247.

[75] T. Parr, The definitive ANTLR 4 reference, Pragmatic Bookshelf, 2013.

[76] D. I. Kaltchev, M. K. Craddock, R. V. Servranckx, T. Risselada, Momentum Cleaning in the CERN LHC (LHC-Project-Report-194. CERN-LHC-Project-Report-194) (1998) 4 p, revised version submitted on 2004-08-20 17:03:58.
URL https://cds.cern.ch/record/359256

[77] A. Lechner, B. Auchmann, T. Baer, C. Bahamonde Castro, R. Bruce, F. Cerutti, L. S. Esposito, A. Ferrari, J. M. Jowett, A. Mereghetti, F. Pietropaolo, S. Redaelli, B. Salvachua, M. Sapinski, M. Schaumann, N. V. Shetty, V. Vlachoudis, E. Skordis, Validation of energy deposition simulations for proton and heavy ion losses in the CERN Large Hadron Collider, Physical Review Accelerators and Beams 22 (7) (2019) 071003.
doi:10.1103/PhysRevAccelBeams.22.071003.

[78] J. Wenninger, Machine Protection and Operation for LHC, in: Proceedings, 2014 Joint International Accelerator School: Beam Loss and Accelerator Protection:

Newport Beach, CA, USA, November 5-14, 2014, 2016, pp. 377–401. arXiv:1608.03113, doi:10.5170/CERN-2016-002.377.

[79] S. Wagner, LHC Machine Protection System: Method for Balancing Machine Safety and Beam Availability, presented 2010 (2010).
URL http://cds.cern.ch/record/1349317

[80] R. Bruce, C. Bracco, R. D. Maria, M. Giovannozzi, A. Mereghetti, D. Mirarchi, S. Redaelli, E. Quaranta, B. Salvachua, Reaching record-low β^* at the CERN Large Hadron Collider using a novel scheme of collimator settings and optics, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment 848 (2017) 19 – 30. doi:https://doi.org/10.1016/j.nima.2016.12.039.
URL
http://www.sciencedirect.com/science/article/pii/S0168900216313092

[81] G. Valentino, R. Aßmann, R. Bruce, F. Burkart, V. Previtali, S. Redaelli, B. Salvachua, G. Stancari, A. Valishev, Beam diffusion measurements using collimator scans in the LHC, Phys. Rev. Spec. Top. Accel. Beams 16 (2013) 021003. 10 p. doi:10.1103/PhysRevSTAB.16.021003.
URL https://cds.cern.ch/record/1517939

[82] E. Métral, G. Arduini, R. W. Assmann, A. Boccardi, C. Bracco, T. Bohl, F. Caspers, M. Gasior, O. R. Jones, K. Kasinski, T. Kroyer, S. Redaelli, R. Robert-Demolaize, F. Roncarolo, G. Rumolo, B. Salvant, R. Steinhagen, T. Weiler, F. Zimmermann, Transverse Impedance of LHC Collimators (LHC-PROJECT-Report-1015. CERN-LHC-PROJECT-Report-1015) (2007) 4 p. doi:10.1109/PAC.2007.4441125.
URL https://cds.cern.ch/record/1056724

[83] L. Drosdal, LHC injection beam quality during LHC Run I, Ph.D. thesis, CERN (2015-03-03).

[84] S. Redaelli, Beam Cleaning and Collimation Systems, arXiv e-prints (2016) arXiv:1608.03159arXiv:1608.03159.

[85] M. Meddahi, et al., LHC abort gap monitoring and cleaning, Conf. Proc. C100523 (2010) MOPEC009.

[86] S. Bart Pedersen, T. Lefevre, E. Bravin, A. Boccardi, A. Goldblatt, A. Jeff, F. Roncarolo, A. Fisher, First operation of the abort gap monitors for LHC (CERN-BE-2010-026. CERN-ATS-2010-111) (2010) 3 p.
URL https://cds.cern.ch/record/1272172

[87] J. Jimenez, et al., Observations, analysis and mitigation of recurrent LHC beam dumps caused by fast losses in arc half-cell 16L2, in: Proceedings, 9th International Particle Accelerator Conference (IPAC 2018): Vancouver, BC Canada, April 29-May 4, 2018, 2018, p. MOPMF053.
doi:10.18429/JACoW-IPAC2018-MOPMF053.

[88] R. Steerenberg, J. Wenninger, Operational challenges and performance of the LHC during Run II, in: Proceedings, 61st ICFA Advanced Beam Dynamics Workshop on High-Intensity and High-Brightness Hadron Beams (HB2018): Daejeon, Korea, June 17-22, 2018, 2018, p. THA2WD01.
doi:10.18429/JACoW-HB2018-THA2WD01.

[89] CERN, LHC optics web home, http://abpdata.web.cern.ch/abpdata/lhc_optics_web/www/index.html, [Online; accessed 25-April-2019] (2019).

[90] J. . RHUL, pybdsim, http://www.pp.rhul.ac.uk/bdsim/pybdsim/, [Online; accessed 19-October-2019] (2019).

[91] R. De Maria, J. Andersson, B. Dalena, L. Field, M. Giovannozzi, P. Hermes, N. Hoimyr, G. Iadarola, S. Kostoglou, E. Maclean, E. McIntosh, A. Mereghetti, J. Molson, V. Bergyld Olsen, D. Pellegrini, T. Persson, T. Pugnat, M. Schwinzerl, K. Sjobak, I. Zacharov, SixTrack Version 5: Status and new developments (2019) WEPTS043. 4 pdoi:10.18429/JACoW-IPAC2019-WEPTS043.
URL https://cds.cern.ch/record/2693253

[92] O. S. Brüning, P. Collier, P. Lebrun, S. Myers, R. Ostojic, J. Poole, P. Proudlock, LHC Design Report, CERN Yellow Reports: Monographs, CERN, Geneva, 2004.

doi:10.5170/CERN-2004-003-V-1.
URL http://cds.cern.ch/record/782076

[93] M. Brugger, F. Cerutti, A. Ferrari, M. Magistris, M. Santana-Leitner, A. Tsoulou, V. Vlachoudis, Technical description of the implementation of IR7 section at LHC with the FLUKA transport code., Tech. Rep. CERN-AB-Note-2008-031, CERN, Geneva (Dec 2006).
URL https://cds.cern.ch/record/1110292

[94] CERN, Technical specification for the supply of sextupole spool corrector magnets MCS for the LHC superconducting Dipole Magnets (Feb 1999).

[95] C. Estevan, EDMS MQX 2D pictures,
https://edms.cern.ch/document/320881/1,
[Online; accessed 11-September-2019] (2001).

[96] M. Karppinen, A. Ijspeert, N. Hauge, B. R. Nielsen, The Development of the Inner Triplet Dipole Corrector (MCBX) for LHC, Tech. Rep. LHC-Project-Report-265. CERN-LHC-Project-Report-265, CERN, Geneva (Feb 1999).
URL https://cds.cern.ch/record/382868

[97] S. Ramberger, S. Bidon, D. Cornuet, D. Gérard, D. Gurov, O. Hans, W. Kalbreier, O. Kiselev, I. Morozov, A. Ogurtsov, V. Petrov, G. de Rijk, E. Ruvinsky, A. Sukhanov, K. Zhilayev, Normal-Conducting Separation and Compensation Dipoles for the LHC Experimental Insertions, IEEE Trans. Appl. Supercond. 16 (LHC-PROJECT-Report-878. CERN-LHC-Project-Report-878. 2) (2006) 1642837. 5 p. doi:10.1109/TASC.2005.869553.
URL https://cds.cern.ch/record/970361

[98] M. Siegel, Overview of the LHC magnets other than the main dipoles, in: Proceedings of the 2000 European Particle Acceleator Conference (EPAC 2000), Vienna, Austria, 2000, pp. 23–27.

[99] M. Karppinen, J. C. Pérez, R. Senis, Inner Triplet Corrector Package MQSXA for the LHC, IEEE Trans. Appl. Supercond. 12 (LHC-Project-Report-529.

CERN-LHC-Project-Report-529. 1) (2002) 102–106. 5 p.
doi:10.1109/TASC.2002.1018361.
URL https://cds.cern.ch/record/544634

[100] I. S. Baishev, J. B. Jeanneret, I. A. Kourotchkine, G. R. Stevenson, I. L. Azhgirey, Cascade simulation studies for the momentum cleaning insertion of LHC, Tech. Rep. LHC-PROJECT-NOTE-263, CERN, Geneva (Jul 2001).
URL http://cds.cern.ch/record/691855

[101] G. A. Blair, Simulation of laser-wires at CLIC using BDSIM.
URL https://cds.cern.ch/record/725980

[102] J. Carter, I. Agapov, G. A. Blair, L. Deacon, A. I. Drozhdin, N. V. Mokhov, Y. Nosochkov, A. Seryi, Simulation of the ILC collimation system using BDSIM, MARS15 and STRUCT, in: Proceedings of the 2006 European Particle Acceleator Conference (EPAC 2006), Edinburgh, Scotland, 2006, pp. 744–746.

[103] R. Bruce, R. W. Assmann, V. Boccone, C. Bracco, M. Brugger, M. Cauchi, F. Cerutti, D. Deboy, A. Ferrari, L. Lari, A. Marsili, A. Mereghetti, D. Mirarchi, E. Quaranta, S. Redaelli, G. Robert-Demolaize, A. Rossi, B. Salvachua, E. Skordis, C. Tambasco, G. Valentino, T. Weiler, V. Vlachoudis, D. Wollmann, Simulations and measurements of beam loss patterns at the CERN Large Hadron Collider, Physical Review Accelerators and Beams 17 (8) (2014) 081004.
arXiv:1409.3123, doi:10.1103/PhysRevSTAB.17.081004.

[104] BDSIM documentation,
http://www.pp.rhul.ac.uk/bdsim/manual-develop/index.html,
[Online; accessed 20-October-2019] (2019).

[105] Y. Levinsen, H. Burkhardt, V. Talanov, Simulation of Beam-Gas Scattering in the LHC (CERN-ATS-2009-141) (2009) 4 p.
URL https://cds.cern.ch/record/1227586

[106] A. Gorzawski, A. Abramov, R. Bruce, N. Fuster-Martínez, M. Krasny, J. Molson, S. Redaelli, M. Schaumann, Collimation of partially stripped ions in the CERN

Large Hadron Collider, arXiv e-prints (2020) arXiv:2007.12507arXiv:2007.12507.

[107] E. Quaranta, A. Valloni, R. Bruce, A. Mereghetti, S. Redaelli, Updated implementation of collimator materials in SixTrack and MERLIN codes (2018) 10 p.
URL http://cds.cern.ch/record/2694337

[108] CERN, SixTrack Version 4.7.16 (Aug 2018).

[109] M. Stockner, B. Dehning, C. W. Fabjan, E. B. Holzer, D. Kramer, Classification of the LHC BLM Ionization Chamber, no. CERN-AB-2007-029, 2007, p. 3 p.
URL https://cds.cern.ch/record/1045385

[110] C. Zamantzas, G. S, LHC BLM system: Expert name convention additions, https://indico.cern.ch/event/318640/contributions/1696055/attachments/613074/843440/MPP_LHCBLM_LS1_ExpNameChanges_20140516.pdf, [Online; accessed 12-September-2019] (2014).

[111] W. Höfle, Transverse damper (2012) 6 pdoi:10.5170/CERN-2012-006.157.
URL https://cds.cern.ch/record/1492958

[112] V. Moens, R. Bruce, S. Redaelli, B. Salvachua, G. Valentino, Comparison of LHC Beam Loss Maps Using the Transverse Damper Blow Up and Tune Resonance Crossing Methods (CERN-ACC-2013-0074) (2013) 3 p.
URL https://cds.cern.ch/record/1574590

[113] C. Zamantzas, G. S, Impact parameters on LHC collimator for different ADT settings, http://lhc-collimation-upgrade-spec.web.cern.ch/LHC-Collimation-Upgrade-Spec/Files/meetings/23/colusm23-07062013-Valentina.pdf, [Online; accessed 24-September-2019] (2013).

[114] I. Strašík, I. Prokhorov, O. Boine-Frankenheim, Beam halo collimation in heavy ion synchrotrons, Phys. Rev. ST Accel. Beams 18 (2015) 081001.

doi:10.1103/PhysRevSTAB.18.081001.
URL https://link.aps.org/doi/10.1103/PhysRevSTAB.18.081001

[115] P. D. Hermes, Heavy-Ion Collimation at the Large Hadron Collider: Simulations and Measurements, presented 19 Dec 2016 (Sep 2016).
URL https://cds.cern.ch/record/2241364

[116] K. Yamamoto, S. Kato, H. Harada, M. Yoshimoto, Y. Yamazaki, N. Hayashi, P. K. Saha, H. Hotchi, M. Kinsho, Performance of the beam collimation system in the J-PARC RCS, Prog. Nucl. Sci. Tech. 4 (2014) 243–246.
doi:10.15669/pnst.4.243.

[117] ATLAS Collaboration, Search for new phenomena with the monojet and missing transverse momentum signature using the ATLAS detector in $\sqrt{s} = 7\,\text{TeV}$ proton-proton collisions, Physics Letters B 705 (4) (2011) 294–312.
arXiv:1106.5327, doi:10.1016/j.physletb.2011.10.006.

[118] ATLAS Collaboration, Characterisation and mitigation of beam-induced backgrounds observed in the ATLAS detector during the 2011 proton-proton run, Journal of Instrumentation 8 (7) (2013) P07004. arXiv:1303.0223, doi:10.1088/1748-0221/8/07/P07004.

[119] Beam backgrounds in the ATLAS detector during LHC loss map tests at $\beta^* = 40\,\text{cm}$ and $\beta^* = 80\,\text{cm}$ at $E_{\text{beam}} = 6.5\,\text{TeV}$, Tech. Rep. ATL-DAPR-PUB-2017-001, CERN, Geneva (Oct 2017).
URL https://cds.cern.ch/record/2286836

[120] R. Bruce, M. Huhtinen, A. Manousos, F. Cerutti, L. Esposito, R. Kwee-Hinzmann, A. Lechner, A. Mereghetti, D. Mirarchi, S. Redaelli, B. Salvachua, Collimation-induced experimental background studies at the CERN Large Hadron Collider, Physical Review Accelerators and Beams 22 (2) (2019) 021004. doi:10.1103/PhysRevAccelBeams.22.021004.

[121] A. Ferrari, P. R. Sala, A. Fasso, J. Ranft, FLUKA: A multi-particle transport code (Program version 2005).

[122] CERN, Beam background generator, http://bbgen.web.cern.ch/bbgen/, [Online; accessed 15-July-2019] (2019).

[123] H. Burkhardt, I. Efthymiopoulos, Interface with Experiments, arXiv e-prints (2017) arXiv:1705.09492arXiv:1705.09492.

[124] A. Fassò, A. Ferrari, P. R. Sala, J. Ranft, FLUKA: Status and prospects for hadronic applications, in: A. Kling, F. J. C. Baräo, M. Nakagawa, L. Távora, P. Vaz (Eds.), Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2001, pp. 955–960.

[125] S. Roesler, R. Engel, J. Ranft, The Monte Carlo Event Generator DPMJET-III, 2001, p. 1033. doi:10.1007/978-3-642-18211-2_166.

[126] T. Pierog, I. Karpenko, J. M. Katzy, E. Yatsenko, K. Werner, EPOS LHC: test of collective hadronization with LHC data, arXiv e-prints (2013) arXiv:1306.0121arXiv:1306.0121.

[127] C. Baus, T. Pierog, R. Ulrich, Cosmic Ray Monte Carlo (CRMC).

[128] M. Dobbs, J. B. Hansen, The HepMC C++ Monte Carlo event record for high energy physics, Computer Physics Communications 134 (1) (2001) 41–46. doi:10.1016/S0010-4655(00)00189-2.

[129] ATLAS Collaboration, Comparison between simulated and observed LHC beam backgrounds in the ATLAS experiment at $E_{\text{beam}} = 4\,\text{TeV}$, Journal of Instrumentation 13 (12) (2018) P12006. arXiv:1810.04450, doi:10.1088/1748-0221/13/12/P12006.

[130] H. Pernegger, The Pixel Detector of the ATLAS experiment for LHC Run-2, JINST 10 (06) (2015) C06012. doi:10.1088/1748-0221/10/06/C06012.

[131] A. Gorisek, The ATLAS Beam Condition Monitor Commissioningdoi:10.5170/CERN-2008-008.264.
URL https://cds.cern.ch/record/1158638

[132] M. Aaboud, et al., Comparison between simulated and observed LHC beam backgrounds in the ATLAS experiment at $E_{\text{beam}} = 4$ TeV, JINST 13 (12) (2018) P12006. arXiv:1810.04450, doi:10.1088/1748-0221/13/12/P12006.

[133] A. Collaboration, Noncollision background public results, https://twiki.cern.ch/twiki/bin/view/AtlasPublic/NonCollisionBackgroundPublicResults, [Online; accessed 8-November-2019] (2019).

[134] J. L. Feng, I. Galon, F. Kling, S. Trojanowski, Forward search experiment at the lhc, Phys. Rev. D 97 (2018) 035001. doi:10.1103/PhysRevD.97.035001. URL https://link.aps.org/doi/10.1103/PhysRevD.97.035001