

# Simulation, Software and First ATLAS Physics.

Alexander John Richards

Thesis submitted for the degree of

**Doctor of Philosophy**

June 2, 2011

Department of Physics and Astronomy

University College London

## Declaration

I, Alexander J. Richards, 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 thesis.

Alexander Richards

Alexander Richards

## Abstract

Updates to the ATLAS fast simulation software are presented which improve the flexibility of its calorimeter and reconstructor objects allowing the easy implementation of new and unforeseen detector effects. Both hot and dead cell types are studied as initial examples and can be seen to be working as expected. A suite of jet finding algorithms known as ‘FastJet’ is introduced and linked into the ATLAS code framework to help unify the description of jets between fast and full simulations as well as in the reconstruction of data.

Preparations for an early-data supersymmetry search in the 0-lepton, jets and  $E_T^{miss}$  channel are presented which, in the absence of a study of the background systematics, show the validity of the channel in detecting the ‘SU3’ mSUGRA benchmark point above the combined standard model background. The most up-to-date publication observes good agreement between the simulated SM background and data up to values of  $E_T^{miss} \sim 100$  GeV and  $M_{eff} \sim 1500$  GeV showing a good understanding of both detector and physics simulation and that the real ATLAS detector is performing as expected.

The first ATLAS inclusive measurement of charged particle multiplicities in events with  $n_{ch} \geq 1$  within the kinematic range  $p_T > 500$  GeV and  $|\eta| < 2.5$  is discussed. With a measured charged particle multiplicity per event and per unit of pseudorapidity at  $\eta = 0$  of  $1.333 \pm 0.003(\text{stat.})$

$\pm 0.040$ (syst.) being some 5-15% higher than predicted, clear differences are evident between the Monte Carlo predictions and what is observed in the data. Contributions from the author including trigger efficiency studies, a ‘Rivet’ analysis routine, a simple simulation of the MBTS as well as the creation of a fast trigger simulation of the MBTS triggers L1\_MBTS\_1, L1\_MBTS\_2 and L1\_MBTS\_1\_1 are detailed.



## Acknowledgements

Firstly I would like to thank both Claire Gwenlan and Jon Butterworth for being excellent supervisors and looking after me well in their transition period. Secondly, the members of the ATLAS collaboration with whom I have interacted over the years who always readily provided guidance, stimulating conversation and debate. I would also like to thank the Science and Technology Facilities Council (STFC), formerly the Particle Physics and Astronomy Research Council (PPARC) for their financial support of my Ph.D.

Thanks to Peter Sherwood for his fantastic guidance through the world of C++ and to Brinick Simmons for giving me the responsibility of shift manager of the RTT. Thanks must also go to Emily Nurse for her help and guidance with the latter part of my Ph.D. work and special thanks must go particularly to my good friend Simon Dean who was never too busy to help me out (no matter how trivial the problem), who was always around for a chat and was the person with whom I worked closer than any other throughout my whole time at UCL.

Thanks to my UCL friends and office mates who made life at UCL so enjoyable, particularly Rob Flack, Ilija Bizjak, Matt Kauer and of course Simon Dean but more recently Sean Thomas as well.

Finally I must thank my family and girlfriend for their never-ending support and belief in me and without whom I would possibly not have

achieved half of what I have. They were, are now, and always will be my rock.

*“For the family.”*

# Contents

<b>1. The Standard Model</b>	<b>13</b>
1.1. Introduction . . . . .	13
1.2. Units . . . . .	16
1.3. Quantum Electrodynamics . . . . .	16
1.4. Quantum Chromodynamics . . . . .	17
1.5. Quantum Flavourdynamics . . . . .	18
1.6. Generating Simulated Events . . . . .	23
1.6.1. The Hard Process . . . . .	23
1.6.2. Parton Showers . . . . .	24
1.6.3. Hadronisation . . . . .	25
1.7. Minimum Bias . . . . .	25
1.8. Underlying Event . . . . .	28
<b>2. The Hardware</b>	<b>29</b>
2.1. The LHC . . . . .	29
2.2. The ATLAS Detector . . . . .	34
2.3. ATLAS Co-ordinate System . . . . .	36
2.4. The Inner Detector . . . . .	37
2.4.1. The Pixel Sub Detector . . . . .	40
2.4.2. The Semiconductor Tracker . . . . .	41
2.4.3. The Transition Radiation Tracker . . . . .	42

---

2.5. The Calorimeter System . . . . .	45
2.5.1. The Electromagnetic Calorimeter . . . . .	46
2.5.2. The Hadronic Calorimeter . . . . .	47
2.6. The Muon System . . . . .	50
2.7. Trigger . . . . .	52
2.8. The Minimum Bias Trigger Scintillator . . . . .	53
<b>3. ATLAS Code Framework</b>	<b>55</b>
<b>4. ATLAS Fast Simulation (Atlfast)</b>	<b>58</b>
4.1. Introduction . . . . .	58
4.2. Algorithms . . . . .	60
4.2.1. GlobalEventDataMaker . . . . .	61
4.2.2. CellMaker . . . . .	62
4.2.3. TrackMaker . . . . .	63
4.2.4. ClusterMaker . . . . .	63
4.2.5. ElectronMaker/PhotonMaker/MuonMaker . . . . .	63
4.2.6. ElectronIsolator/PhotonIsolator/MuonIsolator . . . . .	65
4.2.7. JetMaker . . . . .	66
4.2.8. AtlfastB . . . . .	69
4.2.9. EventHeaderMaker . . . . .	70
4.2.10. StandardHistogramMaker . . . . .	71
4.3. Jet Finding Algorithm Update . . . . .	71
4.3.1. Atlfast's Pre-Existing Jet Finders . . . . .	72
4.3.2. FastJet . . . . .	73
4.3.3. Updating Atlfast's Jet Finding Algorithms . . . . .	74
4.4. Custom Cells & Smearers . . . . .	75
4.4.1. Custom Calorimeter . . . . .	75
4.4.2. FileLoader . . . . .	79
4.4.3. Smearers and Reconstructors . . . . .	81

---

4.4.4. Dead Cells Example . . . . .	85
4.4.5. Hot Cell example . . . . .	86
4.5. Fast Trigger Simulation . . . . .	88
<b>5. Code Maintenance</b>	<b>91</b>
5.1. Motivation . . . . .	91
5.2. Introduction . . . . .	92
5.3. Shifters . . . . .	95
5.4. ELog . . . . .	96
5.5. Developer Control . . . . .	97
5.6. Web Interface . . . . .	99
5.7. DCube . . . . .	103
5.8. RTT Test Package Contributions . . . . .	106
5.8.1. TestAtlfast . . . . .	106
5.8.2. GeneratorsRTT . . . . .	107
<b>6. Supersymmetry Search</b>	<b>109</b>
6.1. Introduction . . . . .	109
6.2. R-Parity . . . . .	110
6.3. 0-Lepton Jets and Missing Transverse Energy $E_T^{miss}$ Channel	111
6.4. Simulated Samples . . . . .	113
6.4.1. SUSY Sample . . . . .	113
6.4.2. QCD Sample . . . . .	113
6.4.3. $t\bar{t}$ Samples . . . . .	115
6.4.4. W/Z Samples . . . . .	116
6.4.5. Di-Boson Samples . . . . .	116
6.5. Discriminating Variables . . . . .	117
6.5.1. Missing Transverse Energy, $E_T^{miss}$ . . . . .	118
6.5.2. $\delta\phi$ . . . . .	119
6.5.3. Transverse Sphericity, $S_T$ . . . . .	121

---

6.5.4. Effective Mass, $M_{eff}$ . . . . .	123
6.6. Trigger . . . . .	124
6.7. Pre-Selection and Overlap Removal . . . . .	125
6.7.1. Pre-Selection . . . . .	125
6.7.2. Overlap Removal . . . . .	127
6.8. Cuts . . . . .	128
6.9. Signal Sensitivity . . . . .	130
6.10. Summary and Conclusion . . . . .	133
<b>7. Minimum Bias Analysis</b>	<b>138</b>
7.1. Analysis Introduction . . . . .	138
7.2. Monte Carlo Simulation . . . . .	139
7.3. Event Selection . . . . .	140
7.4. Background Contribution . . . . .	141
7.5. Selection Efficiencies . . . . .	142
7.5.1. Trigger Efficiency . . . . .	142
7.5.2. Vertex Reconstruction Efficiency . . . . .	149
7.5.3. Track Reconstruction Efficiency . . . . .	149
7.6. Correction Procedure . . . . .	150
7.7. Systematic Uncertainties . . . . .	151
7.8. Results . . . . .	152
7.9. Rivet Toolkit . . . . .	155
7.10. Rivet Minimum Bias Analysis . . . . .	156
7.11. MBTS Simulation . . . . .	158
7.12. Summary and Conclusion . . . . .	162
<b>8. Conclusion</b>	<b>169</b>
<b>A. Atlfast Algorithm Flow Diagrams</b>	<b>172</b>
A.1. GlobalEventDataMaker . . . . .	172

---

A.2. CellMaker . . . . .	173
A.3. TrackMaker . . . . .	174
A.4. ClusterMaker . . . . .	175
A.5. DefaultReconstructedParticleMaker . . . . .	176
A.6. Isolator . . . . .	177
A.7. JetMaker . . . . .	178
A.8. AtlfastB . . . . .	179
A.9. EventHeaderMaker . . . . .	180
<b>B. ATLAS Minimum Bias Rivet Analysis Source Code</b>	<b>181</b>
<b>Bibliography</b>	<b>185</b>
<b>List of figures</b>	<b>193</b>
<b>List of tables</b>	<b>203</b>



# Chapter 1.

## The Standard Model

This chapter introduces the standard model of particle physics and details its component field theories. It will also introduce the idea of generating simulated events as well as the concepts of both minimum bias and underlying event.

### 1.1. Introduction

The standard model (**SM**) of particle physics represents our current best understanding of the interactions of matter at the fundamental level. It describes the electromagnetic, weak and strong forces (but not gravity) that can be seen in nature and is based on an  $SU(3) \times SU(2) \times U(1)$  symmetry group. Rather than a single theory, it is a collection of locally gauge invariant quantum field theories governing the interactions of different types of particles. These fundamental particles are accounted for within the standard model, which describes both the spin- $\frac{1}{2}$  fermionic fields that make up the matter and the integer spin bosonic fields for the force mediators.

(a) Lepton Classification

	$l$	$Q$	$L_e$	$L_\mu$	$L_\tau$
First Generation	$e$	-1	1	0	0
	$\nu_e$	0	1	0	0
Second Generation	$\mu$	-1	0	1	0
	$\nu_\mu$	0	0	1	0
Third Generation	$\tau$	-1	0	0	1
	$\nu_\tau$	0	0	0	1

(b) Quark Classification

	$q$	$Q$	$U$	$D$	$C$	$S$	$T$	$B$
First Generation	$u$	$\frac{2}{3}$	1	0	0	0	0	0
	$d$	$-\frac{1}{3}$	0	-1	0	0	0	0
Second Generation	$c$	$\frac{2}{3}$	0	0	1	0	0	0
	$s$	$-\frac{1}{3}$	0	0	0	-1	0	0
Third Generation	$t$	$\frac{2}{3}$	0	0	0	0	1	0
	$b$	$-\frac{1}{3}$	0	0	0	0	0	-1

**Table 1.1.:** Classification system showing quantum numbers for (a) Leptons and (b) quarks excluding spin [1]

All fundamental matter within the standard model has spin- $\frac{1}{2}$  and therefore obeys Fermi statistics. These **fermions** as they are known exist in one of two subsets depending of the nature of their charge (and colour which will be discussed later in Section 1.4). Those with integer charge are called **leptons** while those with fractional charge are called **quarks**. Irrespective of which subgroup they reside in, they can be further divided up into so-called *generations* where the particles in each successive generation are generally more massive than the last. Each of these generations contains a doublet of either leptons or quarks.

Table 1.1 shows the quantum numbers for the fermions of normal matter. The existence of anti-matter necessitates doubling this table to accommodate the anti-matter partners of each of these particles. These anti-matter particles have identical spin and mass to their matter counterparts but their quantum numbers, as detailed in Table 1.1, have opposite sign.

Fundamental Force	Boson	Q	Spin	Mass
Electromagnetism	photon ( $\gamma$ )	0	1	0
Weak Nuclear (charged)	$W^\pm$	$\pm 1$	1	$\sim 80$ GeV
Weak Nuclear (neutral)	$Z^0$	0	1	$\sim 91$ GeV
Strong Nuclear	gluon ( $g$ )	0	1	0
Gravity*	graviton	0	2	0

\* Gravity is not technically included in the Standard model.

**Table 1.2.:** Classification system for the gauge bosons showing the force which they mediate.

As stated in [1], the inclusion of the  $U$  and  $D$  quantum numbers is for completeness and is usually excluded since any quark with a value of zero for  $C, S, T$  or  $B$  must be either a  $u$  or  $d$ .

All of the leptons can exist as free particles but the quarks, owing to their colour (discussed later in Section 1.4) can only exist as a member of a bound state or **hadron**. There are theories about exotic bound states of quarks but those aside, quarks exist in one of two bound states. **Baryons** are the tri-quark bound state, bound in such a way as to have both integer charge and be colour neutral. The other bound state is the quark anti-quark pair called a **meson**. Mesons, like baryons, must have integer charge and be colour neutral.

The fundamental forces of nature are mediated by bosonic fields, that is to say, ones that obey Bose-Einstein statistics. They have integer spin of 1 (with the notable exceptions of gravity and the Higgs) and their quanta are therefore called vector bosons. Table 1.2 shows the quanta of these bosonic fields along with their charge, spin, and mass.

Gravity is not included in the standard model and is assumed to be so weak as to be negligible in any particle physics application. Strictly speaking Table 1.2 can be extended with the inclusion of the Higgs boson (or just Higgs). The Higgs, unlike the others, is not a vector boson but

instead has spin zero and is therefore a scalar boson. The Higgs however is still at this point a theoretical particle as no compelling evidence of its existence has been found. It is hoped that one of the things the new era of particle accelerators will accomplish is to determine whether the Higgs is real or not.

## 1.2. Units

The units in high energy physics are in terms of the electron volt ( $\text{eV}$ ) which is the energy in Joules divided by  $e$  the charge on the electron, which is  $1.6 \times 10^{-19}$  C. As such, momenta are given in  $\text{eV}/c$  and masses in  $\text{eV}/c^2$ . The speed of light in a vacuum,  $c$  is usually dropped however since it is usual to make the choice of setting it, along with  $\hbar$ , to unity as shown in equation (1.1). This choice of normalisation is often referred to as *natural units*.

$$\hbar = c = 1 \tag{1.1}$$

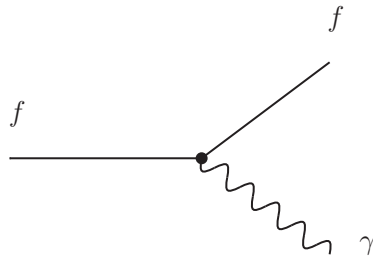
In this way, measurements of energy, momentum and mass are all in terms of eV.

## 1.3. Quantum Electrodynamics

Classically, electrodynamics follows the formalisms of Maxwell. The first quantum theory of electrodynamics was formulated by Tomonaga, Feynman and Schwinger in the 1940s [1] and was called Quantum Electrodynamics (or **QED**). In QED the electromagnetic force is mediated by the

exchange of a photon and affects all (but only) charged particles, i.e. those with a value of quantum number  $Q \neq 0$ .

The photon, as can be seen from Table 1.2 is a massless particle and as such the electromagnetic force has an infinite range. The fundamental QED interaction vertex can be seen in Figure 1.1.



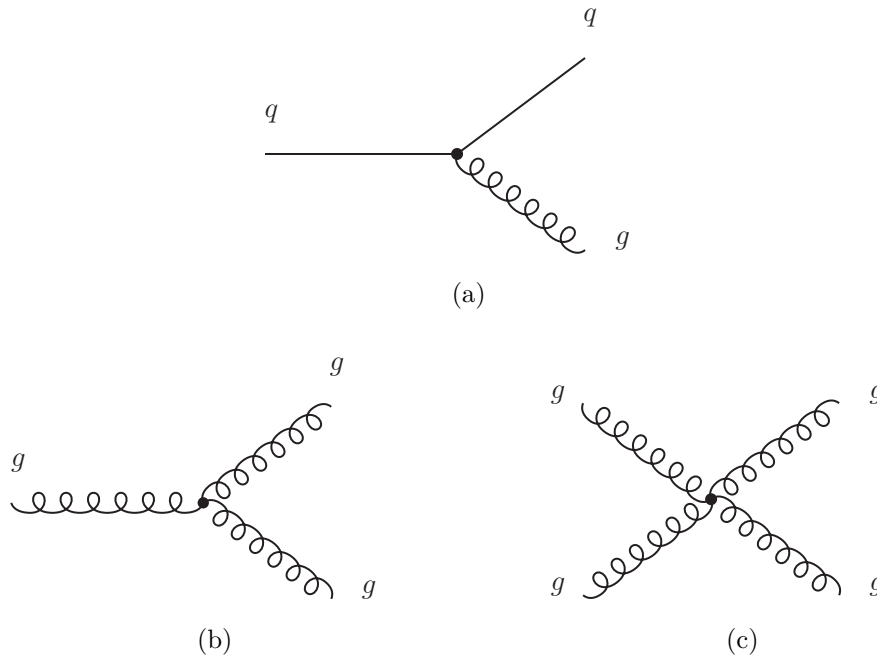
**Figure 1.1.:** Fundamental QED vertex

All electromagnetic interactions can be constructed by repetition of this fundamental vertex. The photon, although interacting solely with charged particles, carries no charge of its own away from the vertex and as such charge must be conserved amongst the other two particles.

## 1.4. Quantum Chromodynamics

Quantum Chromodynamics, (**QCD**) is the quantum field theory that governs the strong interaction. Its quantum is called the gluon. As the name of the theory implies, the gluon couples to coloured objects only (sometimes referred to as colour charge). This colour charge comes in three types labelled red, green and blue. Despite what the names might suggest, these do not represent *actual* colours but rather three additional quantum numbers possessed only by quarks and gluons. Therefore only quarks and gluons experience the strong interaction and one could very well extend Table 1.1

and Table 1.2 to include the  $r$ ,  $g$  and  $b$  quantum numbers. The strong force then affects all particles with  $r, g, b \neq 0$ . The fundamental QCD vertex is shown in Figure 1.2(a).



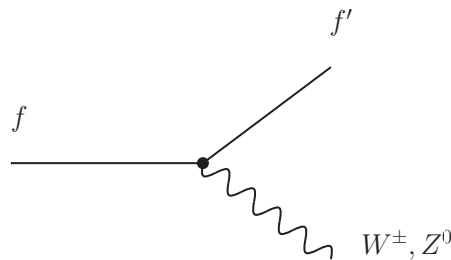
**Figure 1.2.:** Fundamental QCD vertices showing (a) the quark gluon vertex, (b) the tri-gluon vertex and (c) the quad-gluon vertex

Unlike in QED where the photon carried no charge itself, gluons do carry colour charge and as such can interact with themselves, via the vertices shown in Figure 1.2(b) and Figure 1.2(c).

## 1.5. Quantum Flavourdynamics

As with QCD, the theory that governs the weak interaction was developed as a quantum field theory from the beginning and so has no classical partner. Owing to the frequent change of flavour at the weak quark vertex,

it is often referred to as flavourdynamics [1]. The weak interaction is responsible for many types of particle decays including the radioactive  $\beta^\pm$  decay and as with QCD it is somewhat more complicated than QED. The mediators of the weak force are the massive  $W^\pm$  and  $Z^0$  bosons. In direct analogy to QED, where the photon couples to electric charge, the  $W^\pm$  and  $Z^0$  bosons can be said to couple to *weak charge* which is possessed by *all* quarks and leptons including neutrinos. Indeed it is the only type of interaction in which neutrinos interact at all since they possess neither electric charge nor colour charge. Another difference between the weak interaction and the previous two is that unlike the massless photon and gluon, the  $W^\pm$  and  $Z^0$  are very massive as seen in Table 1.2. This finite mass also means that the weak interaction is finite in range.



**Figure 1.3.:** Fundamental weak fermion W/Z interaction vertex

The fundamental vertex in the weak interaction can be seen in Figure 1.3. Traditionally, in the case of leptons, the weak interaction respects the generations below and there are no cross-generational vertices. In this way the weak interaction conserves individual lepton number.

$$\begin{pmatrix} e \\ \nu_e \end{pmatrix} \begin{pmatrix} \mu \\ \nu_\mu \end{pmatrix} \begin{pmatrix} \tau \\ \nu_\tau \end{pmatrix}$$

When it comes to quarks however the picture isn't quite as simple since the flavour eigenstates which participate in the weak interaction are not the

same as the mass eigenstates of the physical particles. One can construct quark generational doublets, similar to the leptonic ones, which are not violated as below.

$$\begin{pmatrix} u \\ d' \end{pmatrix} \begin{pmatrix} c \\ s' \end{pmatrix} \begin{pmatrix} t \\ b' \end{pmatrix}$$

These are now the weak eigenstates where  $d'$ ,  $s'$  and  $b'$  are linear combinations of the mass eigenstates  $d$ ,  $s$  and  $b$ . The relative mixtures of these mass eigenstates is described by the Cabbibo-Kobayashi-Maskawa (**CKM**) matrix equation (1.2).

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \quad (1.2)$$

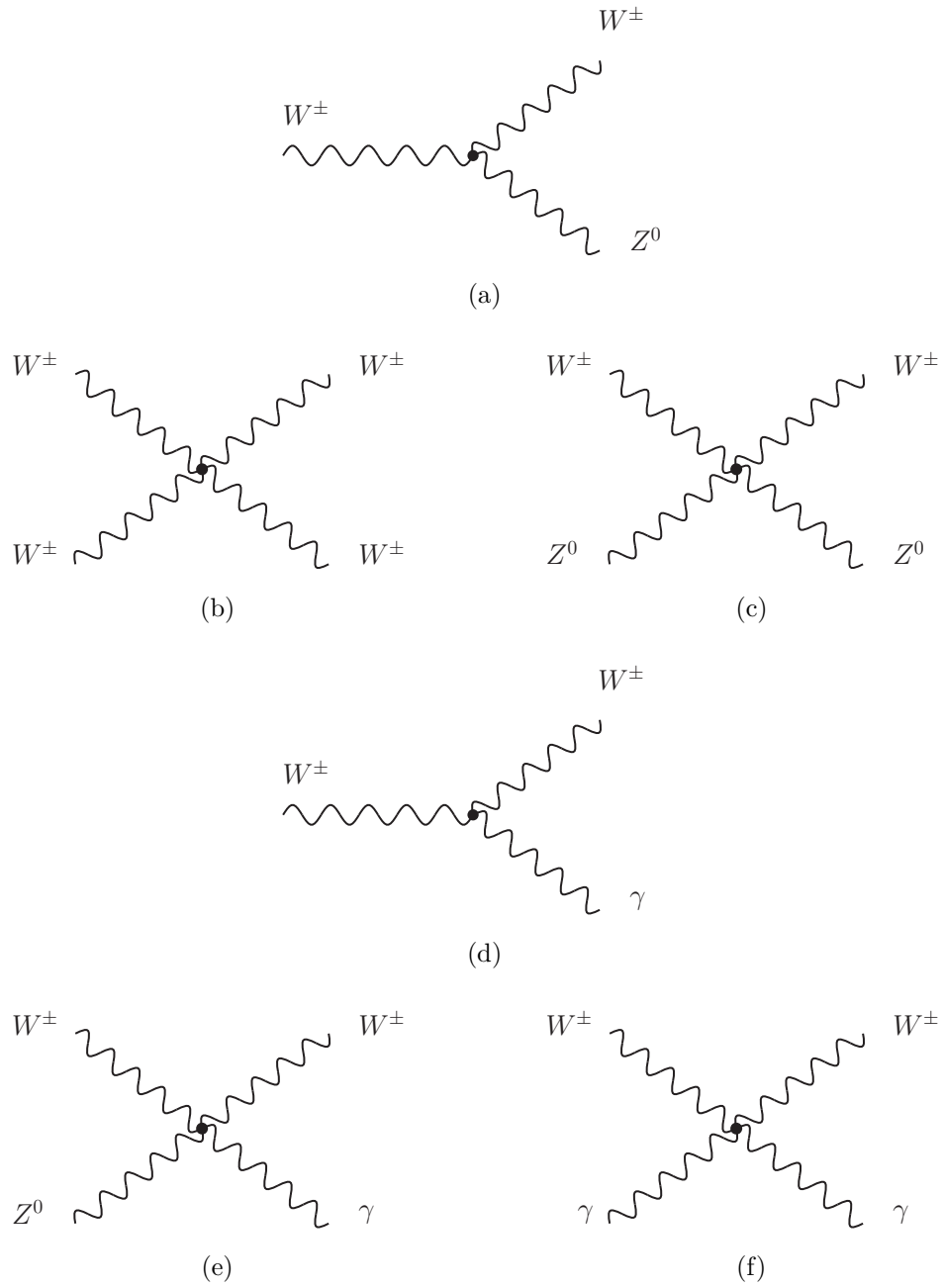
The elements of the CKM matrix are experimentally determined and seem to be close to (though crucially not exactly equal to) those of the unit matrix. If the CKM matrix were identical to the unit matrix then the flavour and mass eigenstates would be one and the same and the weak quark sector would mirror that of the traditional leptonic weak sector.

Verification of *neutrino oscillations* [2,3] increases the complexity of the ‘traditional’ leptonic weak sector. The existence of oscillations implies that the leptonic weak eigenstates  $\nu_e$ ,  $\nu_\mu$  and  $\nu_\tau$  shown above are in fact not the same as the mass eigenstates. Just as with the quark weak eigenstates  $d'$ ,  $s'$  and  $b'$ , they must be considered as linear combinations of the mass eigenstates of three neutrinos  $\nu_{1,2,3}$  with unknown (but crucially not zero) masses  $m_{1,2,3}$ .



Since the weak interaction has both charged and neutral mediators we say that all weak interactions fall into one of the two categories of either *charged current* if they involve the  $W^\pm$  or *neutral current* if they involve the  $Z^0$ . Neutral current processes where a  $Z^0$  is exchanged can also proceed by exchange of the electrically neutral photon under QED. Given the relative propagator masses of the weak and electromagnetic forces, at energies well below the  $W$  mass the weak neutral current processes tend to be masked by the much stronger QED ones. The exception of course is in processes involving the neutrino in which the weak force alone participates. At the high energy of the LHC however it is expected that weak and electromagnetic interactions take place with approximately equal rates.

Finally the field theory predicts couplings of the  $W^\pm$  and  $Z^0$  directly to one another as well as the  $W^\pm$  to the photon since the  $W^\pm$  is a charged particle. The fundamental vertices for such processes can be seen in Figure 1.4



**Figure 1.4.:** Fundamental weak vertices showing (a) tri-WZ vertex, (b) quad-W vertex, (c) quad-WZ vertex, (d) W photon vertex, (e) quad-WZ photon vertex and (f) quad-W photon vertex

## 1.6. Generating Simulated Events

The ability to simulate high energy collisions is useful because it allows us to compare data to our current best understanding of the physics, make estimates about quantities that are difficult/impossible to measure directly from the data and also to prepare analyses before the data are available. The first of these points is circular as by comparison of simulated events to data our understanding and thereby our simulation improves. This in turn gives more confidence in the predictive power of the simulation and thus the latter two points.

Programs called **generators** contain within their numerous algorithms, calculations to describe/predict everything that is currently known about the standard model as well as models to describe the less well understood non-perturbative aspects. These generators utilise Monte Carlo statistical techniques and so are sometimes simply referred to as Monte Carlo (**MC**) generators or just Monte Carlos for short.

There are many types of generators which are specialised to perform certain jobs particularly well. Rather than going through them all, an outline of a general purpose generator such as Pythia [4,5] or Herwig [6] is presented.

### 1.6.1. The Hard Process

At the heart of the generator is the simulation of the hard process. If more than one hard process is selected by the user, the generator dutifully picks which will be generated for any given event statistically based on their relative cross-sections.

Having decided on the type of hard process to be generated the generator must calculate the matrix element for it. This is not a trivial process since the theories are perturbative and one must decide to what order the calculation must be performed (typically to leading order) and the integrals involved are usually solved numerically. In addition if (as is the case with the LHC) the incoming particles are not fundamental but instead composite, the generator must further decide which of the constituents participate in the interaction. This again is performed statistically based on experimentally determined parton distribution functions (**pdf**).

Once calculated, the hard process becomes like a skeleton onto which the rest of the pieces which make up a typical event can be hung.

### 1.6.2. Parton Showers

All accelerated charges emit radiation and so the generator must now consider this for QED and QCD. In QED all electrically charged particles may emit a photon which in turn can split into electron-positron pairs. In QCD all coloured objects (i.e. quarks *and* gluons) may radiate a gluon. Gluons may split into quark pairs or even other gluons.

This means that on top of the hard process, any of the free particles may radiate a new particle which may cause a so-called cascade of emissions or parton shower. If the shower is initiated from the hard process's incident particles it is usually referred to as Initial State Radiation (**ISR**) whereas if it is initiated from the outgoing hard process particles then it is referred to as Final State Radiation (**FSR**).

### 1.6.3. Hadronisation

The parton shower usually continues down to some scale  $\Lambda_{QCD}$  at which point hadronisation takes over. Hadronisation is the process by which the coloured objects left after the parton shower form colourless final state composite particles. This is a non-perturbative step and relies on models.

The two basic types of models are:

- **Cluster Model.** This model takes the particles from the parton shower and proceeds to split the gluons into quark pairs as  $g \rightarrow q\bar{q}$ . Colour singlet combinations of quarks form clusters which decay into pairs of hadrons [7].
- **String Model.** This model is based on a string representing the colour flux between particles which breaks up into hadrons via  $q\bar{q}$  production [7].

## 1.7. Minimum Bias

The definition of minimum bias is one that is dependant on the definition of the triggers used. Given that most of the events at the LHC will either contain no collision or be uninteresting and since disk space is finite and limited, a trigger is used to select only those events of interest to store to disk. This inherently involves biasing our statistics. Minimum bias therefore is the description of the least biasing trigger that we choose to accept events with. It is important to understand this bias as it will be the basis for all other trigger biases and therefore affect all analyses.

The contents of a typical minimum bias event can be understood by looking at the total cross-section. The total cross-section can be broken

down into that for the elastic and inelastic processes.

$$\sigma_{tot} = \sigma_{elastic} + \sigma_{inelastic}$$

Typically in a detector and certainly in ATLAS at the LHC the elastic part of the cross-section will not be observed due to the finite angular acceptance of the detector and the low polar angles with which the particles in an elastic collision emerge. Such collisions usually end up with the products exiting down the beam pipe.

The inelastic part therefore will be what is focused on. It can further be broken down into its diffractive components; single, double and non-diffractive\*.

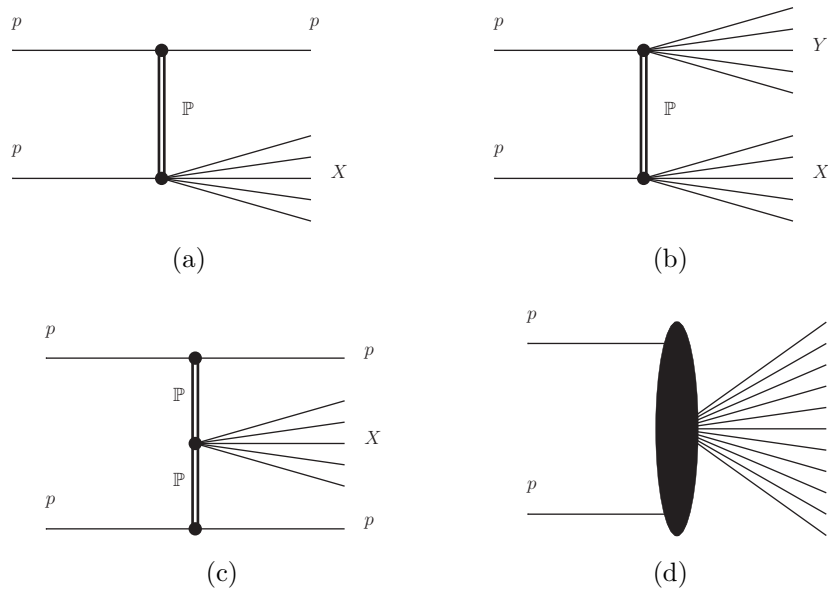
$$\sigma_{tot} = \sigma_{elastic} + \sigma_{sd} + \sigma_{dd} + \sigma_{nd}$$

In diffractive events the protons undergo exchange of a colour singlet object, usually referred to as the pomeron (or  $\mathbb{P}$ ), and then either one (in single diffractive) or both (in double diffractive) of the protons fragment. Non-diffractive events are those where the partons within the protons interact, usually with colour exchanged, leading to both protons fragmenting. The Feynman diagrams for the diffractive and non-diffractive processes can be seen in Figure 1.5.

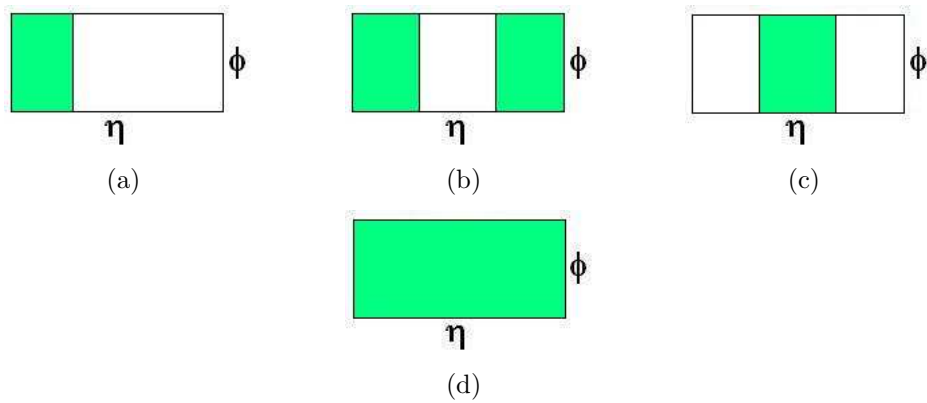
Diffractive events are typically signified by highly populated areas of the detector being interspersed with areas of low population density called *gaps*. Figure 1.6 shows the occupancy of a rolled out detector for both diffractive and non-diffractive events. For non-diffractive events this is not the case and the event will occupy the entirety of the detector.

---

\*There is also a central diffractive part but the cross-section is assumed to be so small as to be negligible here.



**Figure 1.5.:** Feynman diagrams for (a) single diffractive, (b) double diffractive, (c) central diffractive (double  $\mathbb{P}$ ) and (d) non-diffractive events



**Figure 1.6.:** Figure showing rolled out projection of a detector with azimuthal angle vertically and a representation of the polar angle horizontally. The shaded regions show the particle occupancy in (a) single diffractive, (b) double diffractive, (c) central diffractive (double  $\mathbb{P}$ ) and (d) non-diffractive events

These gap based definitions are not totally rigorous as it is possible for example to have double diffractive events in which the gap is so small as to look like a non-diffractive event [8].

## 1.8. Underlying Event

Hadron colliders are noisy environments and as such people's definitions of the underlying event (UE) can sometimes differ. The idea of the underlying event can depend on the definition of what parts of the event one is interested in. In the author's opinion the underlying event consists of everything coming from the same collision that is not related to the hard scattering process that one is studying. This then would include things like the proton remnants and multiple parton interactions (**MPI**) but not any cosmic ray particles, pile-up events or cavern background activity. Crucially the author does *not* consider ISR or FSR as part of the underlying event, rather part of the hard scattering process to be studied.

Regardless of the exact definition of underlying event employed, the task of recognising and as accurately as possible being able to pick out one's interesting hard scattering process on top of it can be a challenge for all analyses.



# Chapter 2.

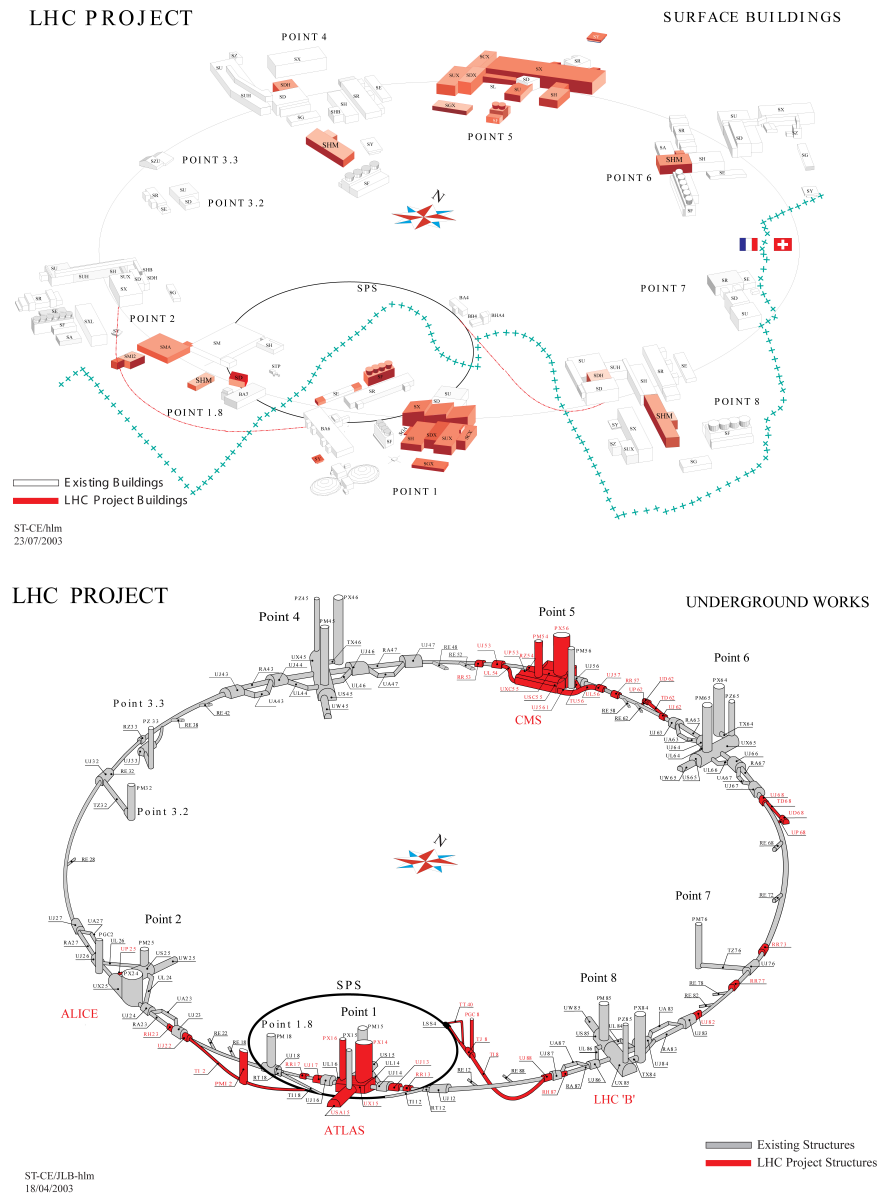
## The Hardware

This chapter focuses on the LHC itself as well the ATLAS detector and strives to give a more in-depth understanding of the working of both of these important physics tools.

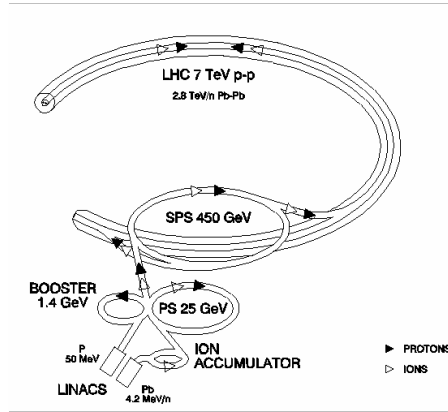
### 2.1. The LHC

The Large Hadron Collider (**LHC**) is (as of 2009) the worlds highest energy particle accelerator. One of the largest civil engineering projects of recent times, the LHC resides in the old LEP tunnel 27 km in circumference and 50 – 175 m underground beneath the Swiss - French border as shown in Figure 2.1 in which the original LEP buildings are in grey with LHC extensions in red.

The proton-proton synchrotron can collide protons with nominal centre of mass energy of 14 TeV and a design luminosity of  $10^{34} \text{ cm}^{-2}\text{s}^{-1}$ . At this luminosity beam crossings occur every 25 ns and there are  $\sim 23$  interactions per crossing [10]. Exploring such an energy frontier will mean that the attached experiments can explore the electroweak symmetry breaking, look



**Figure 2.1.:** The LHC surface and underground layouts showing the old LEP buildings in grey and the new LHC buildings in red [9].



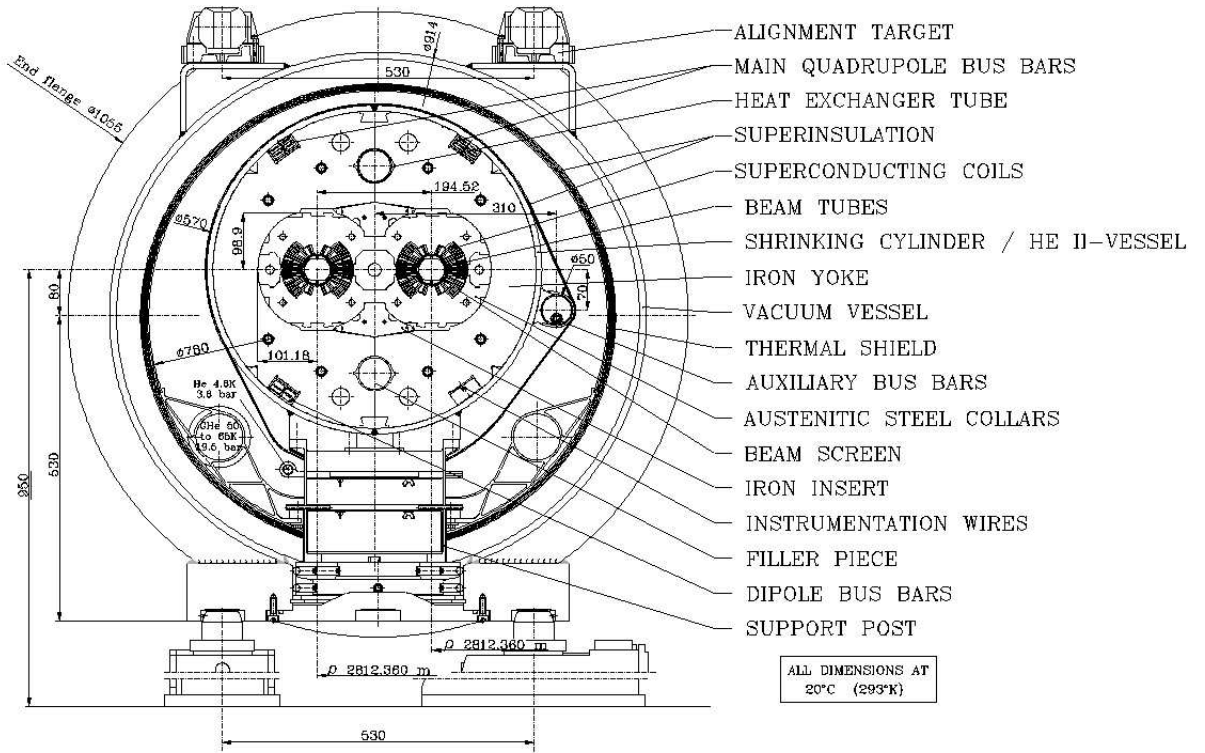
**Figure 2.2.:** The full LHC injector chain for both protons and heavy ions [11]

for extra dimensions, try to explain matter-antimatter asymmetry, and search for a dark matter candidate as they try to understand physics at the TeV scale.

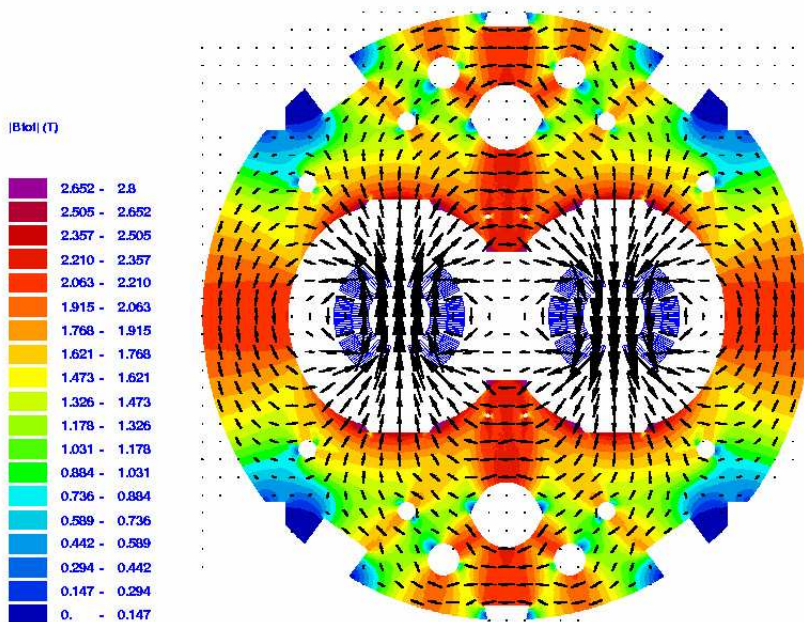
The LHC injection chain is a four stage process involving three other synchrotrons and a linac which had to be upgraded to meet the demands of the LHC. The chain begins with protons being accelerated in Linac2 up to 50 MeV, from here they pass into the Proton Synchrotron Booster (PSB) which further accelerates them up to 1.4 GeV. Next they are injected into the Proton Synchrotron (PS) in which they achieve an energy of 25 GeV before injection into the final part of the injection chain, the Super Proton Synchrotron (SPS). The SPS gives the protons the final boost before injection into the main LHC ring at 450 GeV as seen in Figure 2.2. For the heavy ion running program the chain is essentially the same with the exception that Linac3 rather than Linac2 is used and the PSB is not used as the required ion phase-space density could not be achieved by using it [11]. The particle beams are injected in two locations on the main LHC ring either side of point 1 (see Figure 2.1). The clockwise beam (beam 1) is injected at point 2 and the anti-clockwise beam (beam 2) at point 8. On entering the ring, the particles are accelerated using RF cavities and bent by the superconducting magnets. Many different variants of magnets are

used to perform different tasks but by far the majority are the cryodipole magnets (Figure 2.3) responsible for bending and accelerating the beams to such high energies by producing a magnetic field up to a nominal value of 8.33 T. In total there are 1232 of these curved dipole magnets each one about 16.5 m long with a diameter of 570 mm at room temperature and a mass of about 27.5 tonnes. The dipole is curved in the horizontal plane with a radius of curvature of  $\sim 2812$  m (at 293 K). They are cooled to 1.9 K by a vast cryogenic system utilising superfluid helium and machined to within very fine tolerances of hundredths of a millimetre [12].

Finally the counter rotating beams, having been accelerated to the desired energy up to the nominal 7 TeV are brought into collision at one of the experiment points, point 1 (ATLAS), point 2 (ALICE), point 5 (CMS) or point 8 (LHCb).

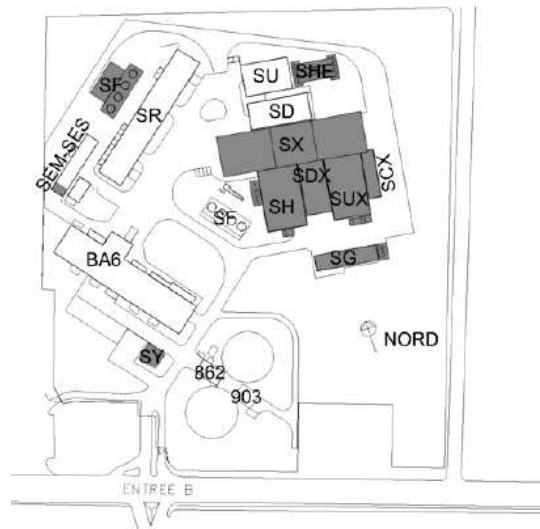


(a)



(b)

Figure 2.3.: A single LHC Dipole magnet (a) in cross-section and (b) showing the B-Field [12].



**Figure 2.4.:** Surface view of Point 1 showing the building covering the ATLAS shaft, SDX [14].

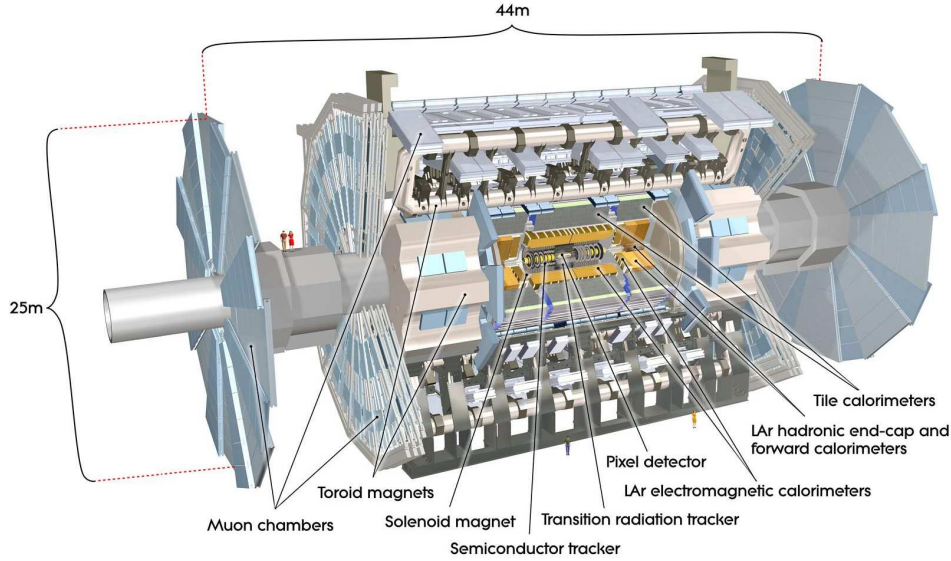
## 2.2. The ATLAS Detector

Located at point 1 (Figure 2.4), ATLAS (*A Toroidal Lhc ApparatuS*) is one of the four main experiments on the LHC ring and one of the two so-called general purpose detectors. It measures 44 m long by  $\sim 25$  m high and weighs  $\sim 7,000$  tonnes [13].

Its cylindrical design allows for a shell model of detector with the inner detector (**ID**) closest to the beam pipe surrounded by a solenoid which in turn is surrounded by the calorimetry systems and finally the muon systems as seen in Figure 2.5.

The ATLAS detector provides the following as specified in the basic design criteria [10]

- Very good electromagnetic calorimetry for electron and photon identification and measurements, complemented by full-coverage hadronic



**Figure 2.5.:** The ATLAS Detector (Photo. #:0803012\_01 ATLAS Experiment ©CERN 2011) [13]

calorimetry for accurate jet and missing transverse energy\* ( $E_T^{miss}$ ) measurements.

- High-precision muon momentum measurements, with the capability to guarantee accurate measurements at the highest luminosity using the external muon spectrometer alone.
- Efficient tracking at high luminosity for high- $p_T$  lepton-momentum measurements, electron and photon identification,  $\tau$ -lepton and heavy-flavour identification, and full event reconstruction capability at lower luminosity.
- Large acceptance in pseudorapidity ( $\eta$ ) with almost full azimuthal angle ( $\phi$ ) coverage everywhere. The azimuthal angle is measured around the beam axis, whereas pseudorapidity relates to the polar angle ( $\theta$ ) where  $\theta$  is the angle from the z direction see Section 2.3

---

\*  $E_T^{miss}$  is defined as  $-1 \times$  (vector sum of all the energy in the transverse x-y plane), see Section 6.5.1

- Triggering and measurements of particles at low- $p_T$  thresholds, providing high efficiencies for most physics processes of interest at the LHC

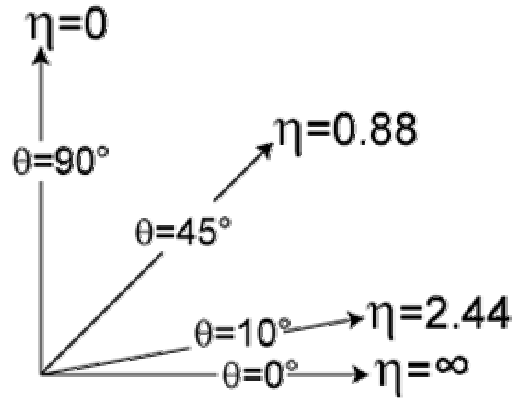
### 2.3. ATLAS Co-ordinate System

ATLAS uses a system with Cartesian basis vectors  $x$ ,  $y$  and  $z$  with their origin situated at the nominal beam interaction point. In this system  $z$  is parallel with the beam direction and the plane transverse to this is the  $x$ - $y$  plane. Positive  $x$  points to the centre of the LHC ring and positive  $y$  vertically upwards. The side of the detector at  $z > 0$  is referred to as side A whilst that at  $z < 0$  is side C. For completeness side B refers to the  $x$ - $y$  plane at  $z = 0$ .

Transverse quantities (usually denoted with subscript T) are defined in the transverse  $x$ - $y$  plane. The polar angle ( $\theta$ ) is measured from the positive  $z$  axis and is defined as  $\theta \in \mathbb{R} [0, \pi]$  while the azimuthal angle  $\phi$  is measured around the beam axis ( $z$ ) starting from the  $x$  axis and is defined thusly  $\phi \in \mathbb{R} [-\pi, \pi]$ . Since in the transverse plane the detector's shell structure resembles concentric circles, it also makes sense to talk about a transverse distance from the centre of these circles i.e. the  $z$  (or beam) axis. This transverse distance measure is often referred to as the radius  $R$ .

Lastly the pseudorapidity ( $\eta$ ) is related to  $\theta$  by the relation  $\eta = -\ln \left[ \tan \left( \frac{\theta}{2} \right) \right]$ . It is defined as  $\eta \in \mathbb{R} [-\infty, \infty]$  where  $\eta = \pm\infty$  is along the beam axis in the positive and negative  $z$  direction respectively.  $\eta = 0$  corresponds to  $\theta = \pi/2$  or lying in the transverse  $x$ - $y$  plane at side B as depicted in Figure 2.6.



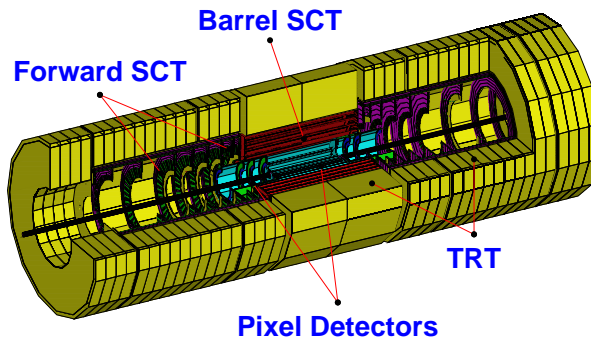


**Figure 2.6.:** Graphical representation of the relation between  $\eta$  and  $\theta$  [15]

## 2.4. The Inner Detector

The inner detector (**ID**) Figure 2.7 is designed to measure the properties of charged particles by recording series of hits which can be combined to form tracks showing their trajectories. The detector must be able to provide a high granularity for accurate distinction of tracks in the high track environment of pp collisions at LHC energies. It must also be able to produce a reasonable number of these hit points such that the track pattern recognition software is able to accurately reproduce the track. The requirement for very high granularity necessitates the use of semiconductor tracking technologies such as strip and pixel detectors. These cannot be used throughout the whole tracking volume however as they introduce a large amount of material and also a high cost. To complete the ID tracking volume and get the required high number of hit points per track  $\sim 36$  [16] for robust pattern recognition the use of straw tube tracking technology is utilised which introduces far less material and at a much lower cost than the semiconductors. This gives the complete ID a high resolution inner tracking volume with a continuous outer tracking volume.

The radius of the full ID is 115 cm and it has a total length of 7 m. In terms of construction the ID constitutes three parts; the barrel which



**Figure 2.7.:** Cut away of ID showing the individual ID subsystems; the pixel, SCT and TRT layers [16]

extends symmetrically from side B  $\pm 80$  cm [16] and two end-caps either side each of length 2.7 m. In the barrel section the detectors are arranged in concentric cylinders while in the end-caps they are arranged in discs perpendicular to the beam axis. In both cases it is the higher precision semiconductor pixels and strips that are innermost with the straw tracker around the outside. The radius of the combined high precision semiconductor detectors is 56 cm. [16] with the remaining 59 cm being the straw tracker. Figure 2.8 shows a schematic drawing of a cross-section in the  $y$ - $z$  plane of one quarter of the ID.

Surrounding the ID is a solenoid the purpose of which is to deflect the trajectories of charged particles in order to provide a measure of the momentum and to assist in particle identification. The solenoid has a length of 5.3 m which is less than that of the ID and is constrained by the length of the EM calorimeter cryostat. The solenoid has a nominal field of 2 T at side B; however, the longitudinal component of this field drops significantly as a function of distance along the beam axis while its radial component increases as a function of radius, see Figure 2.9.

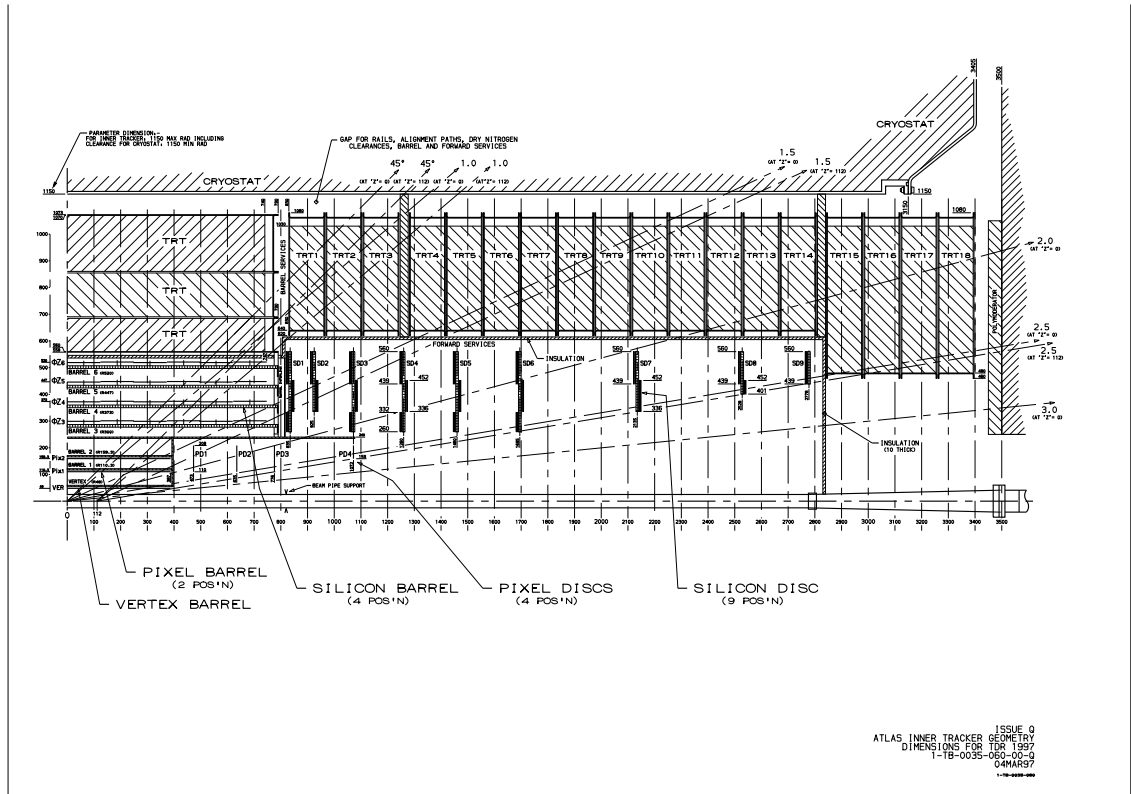


Figure 2.8.: schematic of ID showing the individual ID subsystems; the pixel, SCT and TRT layers [16]

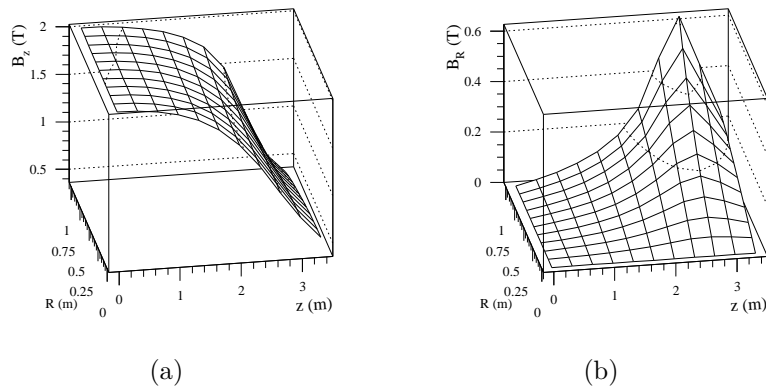
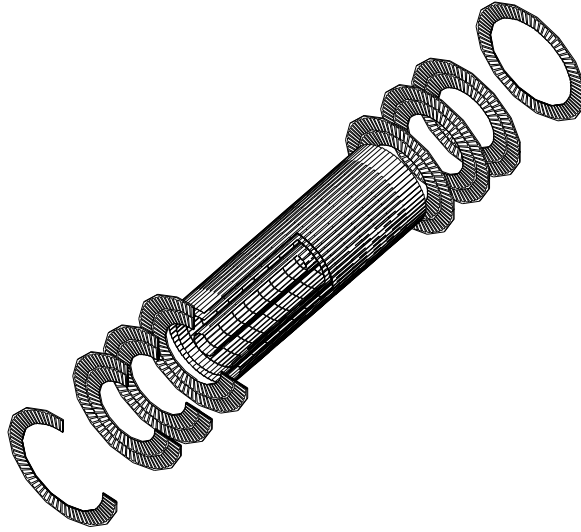


Figure 2.9.: Plots showing the ATLAS solenoid magnetic field intensity for components (a)  $B_z$  and (b)  $B_r$  as a function of longitudinal and radial displacement from the nominal interaction point,  $z$  and  $R$  respectively. [16]



**Figure 2.10.:** Three dimensional view of the complete pixel sub detector system. [16]

### 2.4.1. The Pixel Sub Detector

The pixel sub detector (Figure 2.10) is designed to provide the highest granularity and is situated closest to the beam pipe. Such high granularity proximal to the beam pipe is desirable for among other things obtaining the necessary resolution to identify secondary vertices which will help with things like  $b$  and  $\tau$  tagging. The system consists of three concentric barrels and five disks on each side. The barrels have nominal radii  $\sim 4$  cm, 11 cm and 14 cm and are each 62.4 mm long while the disks have radii between 11 and 20 cm [16]. The barrels together with the disks provide fully symmetric coverage for  $|\eta| < 2.5$  over which the pixel detector provides at least three space points which will help with the track pattern recognition.

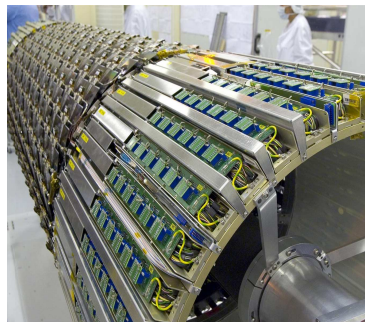
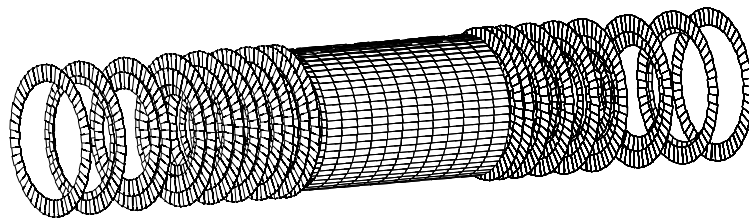
Being situated so close to the interaction point (**IP**) the pixel sub detector has to be extremely radiation hard. Even so the expected lifetime before a replacement must be sought is  $\sim 3$  years at  $10^{33} \text{ cm}^{-2}\text{s}^{-1}$  followed by about one year at design luminosity of  $10^{34} \text{ cm}^{-2}\text{s}^{-1}$  for the innermost

barrel layer. The other parts of the pixel detector are expected to last  $\sim 7$  years at  $10^{34} \text{ cm}^{-2}\text{s}^{-1}$  [17].

### 2.4.2. The Semiconductor Tracker

The Semiconductor Tracker (**SCT**) as seen in Figure 2.11 is the second layer of semiconductor sub detectors. It utilises silicon micro-strip technology rather than pixels to provide four space points per track from four concentric cylindrical barrels of radii 300 mm, 373 mm, 447 mm, and 520 mm.

The detector contains  $61 \text{ m}^2$  of silicon detectors each measuring  $6.36 \times 6.40 \text{ cm}$  with 768 readout strips. The detector itself has 6.2 million readout channels and a full coverage of  $|\eta| \leq 2.5$  with a spatial resolution of  $16 \mu\text{m}$  in  $R\phi$  and  $580 \mu\text{m}$  in  $z$  [10].



(b)

**Figure 2.11.:** Views of the ATLAS SCT in (a) perspective [18] and (b) Photo. of the SCT strip sub detector (Photo. #:0509006\_03 ATLAS Experiment ©CERN 2011) [13]

### 2.4.3. The Transition Radiation Tracker

The Transition Radiation Tracker (**TRT**) Figure 2.12 is the outermost sub detector of the ID and is based not on silicon semiconductor tracking technology but on the use of straw tubes. This approach can operate at the high rates expected at the LHC because of their small diameters and the isolation of the sense wires within individual gas volumes. This approach is intrinsically radiation hard as well as being relatively inexpensive. The ATLAS TRT typically provides  $\sim 36$  measurements per track [10].

Transition Radiation is produced by the traversing of an inhomogeneous medium by a relativistic particle. In particular this could be the transition between two materials of different electrical properties. The intensity of this radiation is approximately proportional to the particle's energy as in equation (2.1) [19],

$$I = \gamma m \tag{2.1}$$

where  $\gamma$  is the Lorentz factor. Crucially however the energy lost by (and hence the frequency of) this radiation is proportional to the Lorentz factor as shown in equation (2.2) and equation (2.3).

$$E \propto \gamma \tag{2.2}$$

$$\gamma = \frac{E}{mc^2} \tag{2.3}$$

This radiation therefore can provide a certain amount of particle identification. In ATLAS, electron identification is done by using xenon gas within the straws. Electrons will emit this transition radiation in the X-ray range in a radiator between the straws. The radiation will then ionise the xenon gas inside the straws which in turn is picked up by the isolated sense

wire that runs centrally throughout the length of the straw resulting in an above threshold signal from both the particle track and the transition radiation ionisations.

Each straw is 4 mm in diameter and has a 30  $\mu\text{m}$  diameter gold-plated W-Re wire running through it. The barrel contains  $\sim 50,000$  axial straws divided in two at the centre with a maximum straw length of 144 cm. The end-caps contain 320,000 radial straws. The spatial resolution of these straws is 170  $\mu\text{m}$  per straw [10].

The TRT is operated with a mixture of 70% Xe, 20%  $\text{CO}_2$  and 10%  $\text{CF}_4$  with a total volume of 3  $\text{m}^3$  [10].

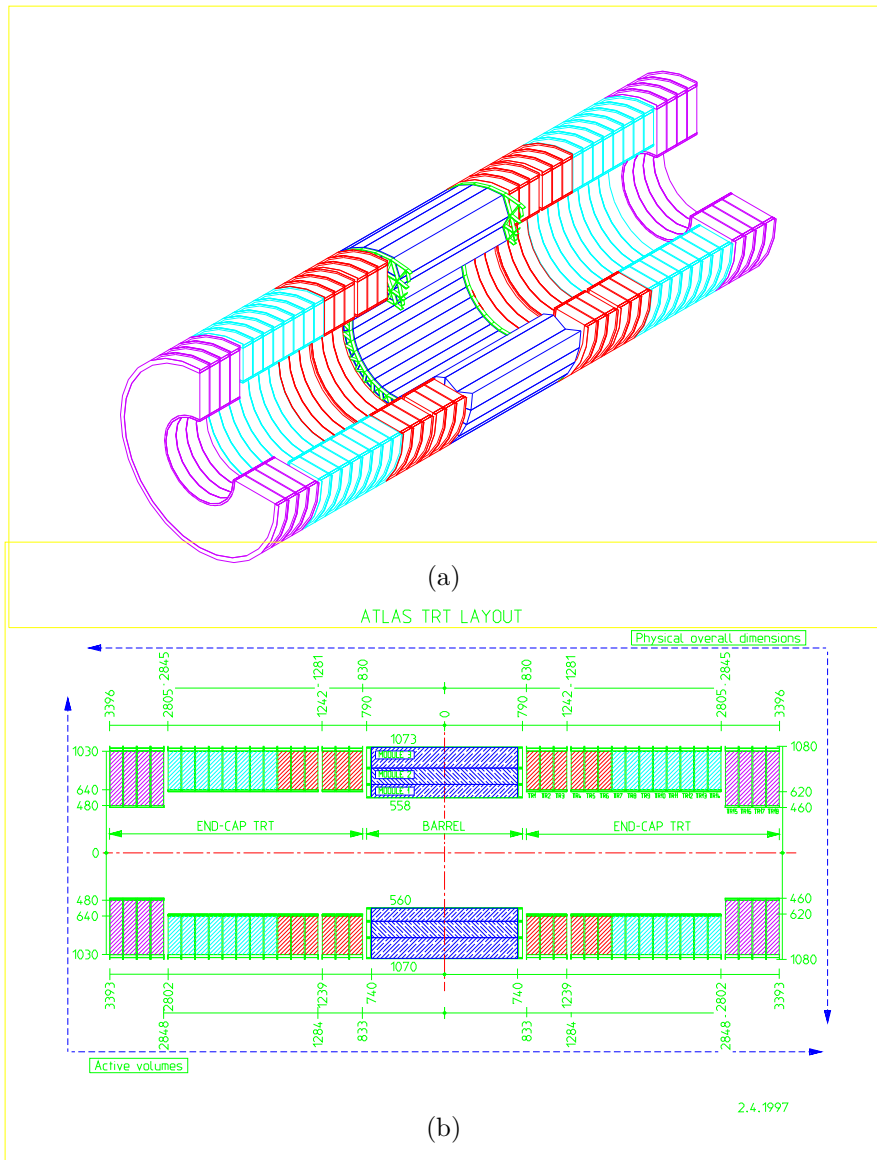
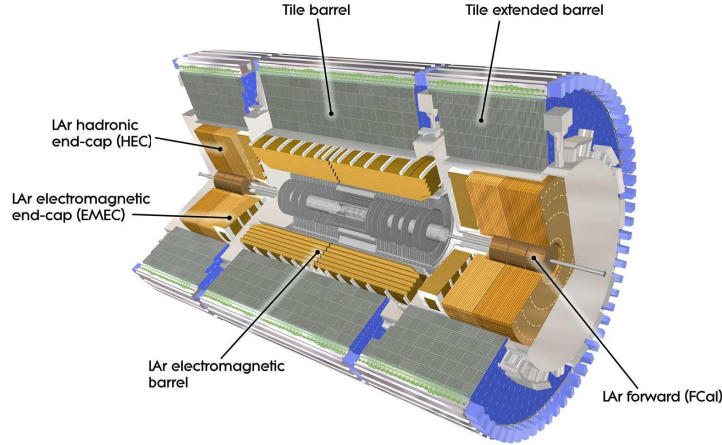


Figure 2.12.: Views of the ATLAS TRT in (a) perspective and (b) R-z profile [17].





**Figure 2.13.:** Perspective view of the ATLAS calorimeter systems

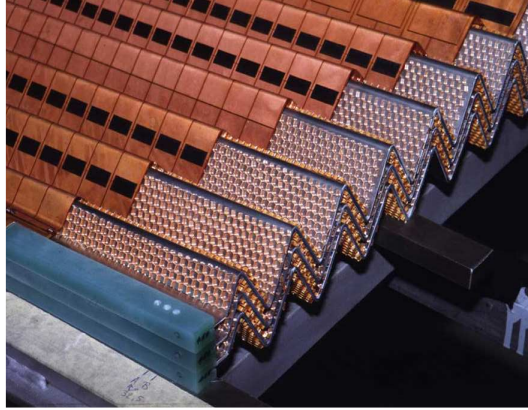
## 2.5. The Calorimeter System

The Calorimeter system (Figure 2.13) is designed to precisely measure electrons, photons, jets and missing transverse energy<sup>†</sup> ( $E_T^{miss}$ ) with complete  $\phi$  coverage. It comprises two subsystems, firstly the electromagnetic (**EM**) calorimeter (consisting of the barrel and end-caps) covering the range  $|\eta| < 3.2$ . Secondly the hadronic (**HAD**) calorimeter barrel (and extended barrel) covering  $|\eta| < 1.7$  and end-caps overlapping the extended barrel slightly with  $1.5 < |\eta| < 3.2$ . Finally the HAD calorimeter is completed with the forward calorimeters (**FCAL**) covering  $3.1 < |\eta| < 4.9$  [10].

There also exists a presampling calorimeter which can be used to correct for energy losses in the material of the ID and the coils etc. and covers the range  $|\eta| < 1.52$  in the barrel and  $1.5 < |\eta| < 1.8$  in the end-caps. In both cases the granularity in  $\Delta\eta \times \Delta\phi$  is  $0.025 \times 0.1$

---

<sup>†</sup>  $E_T^{miss}$  is defined as  $-1 \times$  (vector sum of all the energy in the transverse x-y plane), see Section 6.5.1



**Figure 2.14.:** Photo. showing the accordion shaped electrodes and absorber plates (Photo. #:9308048\_09\_general ATLAS Experiment ©CERN 2011). [13]

### 2.5.1. The Electromagnetic Calorimeter

The electromagnetic calorimeter is concerned with the detection of primarily (but not exclusively) electrons and photons which tend not to reach the hadronic calorimeter. In addition the measurements made by the EM calorimeter are used in the  $E_T^{miss}$  determination which requires knowledge of the total energy (EM and HAD) in the event. The ATLAS EM calorimeter uses a liquid argon (**LAr**) medium with accordion shaped lead absorber plates and Kapton electrodes (as in Figure 2.14) which makes it intrinsically more radiation hard than the tile parts of the HAD calorimeter. The ATLAS electromagnetic calorimeter consists of two components, the barrel and two end-caps. The barrel is divided into two identical half-barrels with a 6 mm gap at  $z = 0$  and covers a range of  $|\eta| < 1.475$ . The end-caps (one at each end) are divided into two coaxial wheels; an outer wheel covering the range  $1.375 < |\eta| < 2.5$  and an inner wheel covering the range  $2.5 < |\eta| < 3.2$  [10].

The EM calorimeter is (with the exception of the small *crack* region between barrel and end-caps at  $1.475 < |\eta| < 1.5$ ) split into three concentric measuring sections also called *samplings* right up to  $|\eta| < 2.5$  and two

EM CALORIMETER	Barrel	End-cap	
Coverage	$ \eta  < 1.475$	$1.375 <  \eta  < 3.2$	
Longitudinal segmentation	3 samplings	3 samplings	$1.5 <  \eta  < 2.5$
		2 samplings	$1.375 <  \eta  < 1.5$
			$2.5 <  \eta  < 3.2$
Granularity ( $\Delta\eta \times \Delta\phi$ )			
Sampling 1	$0.003 \times 0.1$	$0.025 \times 0.1$	$1.375 <  \eta  < 1.5$
		$0.003 \times 0.1$	$1.5 <  \eta  < 1.8$
		$0.004 \times 0.1$	$1.8 <  \eta  < 2.0$
		$0.006 \times 0.1$	$2.0 <  \eta  < 2.5$
		$0.1 \times 0.1$	$2.5 <  \eta  < 3.2$
Sampling 2	$0.025 \times 0.025$	$0.025 \times 0.025$	$1.375 <  \eta  < 2.5$
		$0.1 \times 0.1$	$2.5 <  \eta  < 3.2$
Sampling 3	$0.05 \times 0.025$	$0.05 \times 0.025$	$1.5 <  \eta  < 2.5$

**Table 2.1.:** Granularity and pseudorapidity coverage of the ATLAS EM calorimeter [10].

samplings after that. These are simply labelled numerically starting with the innermost (smallest R) sampling 1.

The relative granularities of these samplings varies and can be seen summarised in Table 2.1.

## 2.5.2. The Hadronic Calorimeter

The purpose of the hadronic calorimeter is to measure as accurately as possible the energy deposition of jets (clusters of hadronic energy as defined by a jet algorithm, see Section 4.3). As with the EM calorimeter the measurements made with the HAD calorimeter are also used in identifying any  $E_T^{miss}$  in the event. In order to get the best  $E_T^{miss}$  measurement possible the HAD calorimeter must be able to measure high energy jets with good resolution over large  $\eta$  ranges.

The ATLAS hadronic calorimeter does not use a single technique but instead uses differing techniques in the barrel than it does in the end-caps

<b>HADRONIC TILE</b>	<b>Barrel</b>	<b>Extended barrel</b>
Coverage	$ \eta  < 1.0$	$0.8 <  \eta  < 1.7$
Longitudinal segmentation	3 samplings	3 samplings
Granularity ( $\Delta\eta \times \Delta\phi$ )		
Samplings 1 and 2	$0.1 \times 0.1$	$0.1 \times 0.1$
Sampling 3	$0.2 \times 0.1$	$0.2 \times 0.1$
<b>HADRONIC LAr</b>	<b>End-cap</b>	
Coverage	$1.5 <  \eta  < 3.2$	
Longitudinal segmentation	4 samplings	
Granularity ( $\Delta\eta \times \Delta\phi$ )	$0.1 \times 0.1$	$1.5 <  \eta  < 2.5$
	$0.2 \times 0.2$	$2.5 <  \eta  < 3.2$
<b>FORWARD CALORIMETER</b>	<b>Forward</b>	
Coverage	$3.1 <  \eta  < 4.9$	
Longitudinal segmentation	3 samplings	
Granularity ( $\Delta\eta \times \Delta\phi$ )	$\sim 0.2 \times 0.2$	

**Table 2.2.:** Granularity and pseudorapidity coverage of the ATLAS HAD calorimeter [10].

and FCAL. The full layout of the HAD calorimeter and its relation to the EM calorimeter and ID can be seen in Figure 2.13. The combined HAD calorimeter coverage is  $|\eta| < 4.9$  and a breakdown of the granularity and longitudinal segmentation dependence on  $\eta$  can be seen in Table 2.2. The following sub-sections will look in more detail at the different components that make up the HAD calorimeter.

## HAD barrel

The structure of the HAD calorimeter barrel is cylindrical and consists of the main barrel with  $|\eta| < 1.0$  and an extended barrel on either side with  $0.8 < |\eta| < 1.7$ . Both the main and extended barrels utilise a sampling technique with plastic scintillator plates (tiles) 3 mm thick, inter spaced with iron absorbers [10]. The length of the main barrel is 5640 mm while that of each of the extended barrels is 2910 mm. The inner and outer radii of the barrels are 2280 mm and 4230 mm respectively [20].

Between the barrels and the extended barrels there is a necessary 600 mm gap for the ID and EM cables. In part of this gap is a small extension to the extended barrel called the Intermediate Tile Calorimeter (**ITC**). The purpose of this calorimeter is to maximise the active volume of the HAD calorimeter while still allowing space for the ID and EM cables. The ITC covers the range  $1.0 < |\eta| < 1.6$  and uses the same tile calorimetry technique as the rest of the barrel [10]. The ITC also extends down into the gap between the EM barrel and the end-caps and can be used to sample the energy lost in the cryostat walls.

### **HAD end-caps**

The hadronic end-cap calorimeters (**HEC**) cover  $1.5 < |\eta| < 3.2$ . For both the HEC and the FCAL, since high radiation resistance is required at these higher pseudorapidities, the intrinsically radiation hard LAr technology is used. Unlike the accordion geometry of the EM calorimeter however, the HEC utilises a copper parallel plate arrangement. Each end-cap is made of two independent wheels of outer radius 2.03 m, the first uses 25 mm copper plates while the second uses 50 mm plates for economic reasons. In both wheels the gap between consecutive copper plates is 8.5 mm.

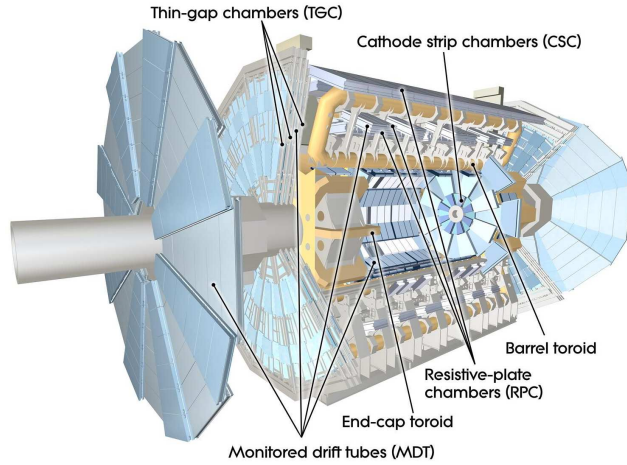
### **FCAL**

Like the HEC the FCAL has to cope with a high radiation environment and as such the LAr technology is also used for this detector. The FCAL is necessarily dense to avoid energy leakage out of the FCAL and into its neighbours. If the FCAL were too low a density then energy would leak into the other end-cap detectors significantly increasing the pile-up. The FCAL can be split into three sections with its front face  $\sim 4.7$  m from

the nominal IP. The first section (closest to the IP) has a copper matrix with longitudinal rod-shaped electrodes. The other two layers are of similar construction but use tungsten as opposed to copper [10,21].

## 2.6. The Muon System

Muons, which are more penetrating, will tend to ‘punch through’ the calorimeter and reach the muon system where they can be recorded.



**Figure 2.15.:** The ATLAS muon spectrometer system (Photo. #:0803017\_01 ATLAS Experiment ©CERN 2011). [13]

The Muon detectors (Figure 2.15) surround a second magnet system that creates a toroidal field of peak strength of 3.9-4.1 T [10]. In the barrel region of  $|\eta| < 1.0$  the field is provided by the barrel toroid (**BT**), a system of eight superconducting coils. Inserted into the either end of the BT are the end-cap toroids (**ECT**) which cover the range  $1.4 < |\eta| < 2.7$ .

Four chamber technologies are used in the muon spectrometer, these are summarised in [10] and condensed below.

**Monitored Drift-Tube Chambers (MDT)** Aluminium tubes 30 mm in diameter and 400 mm thick. They contain a 50 mm diameter central WRe wire. They are filled with a mixture of 93% Ar and 7% CO<sub>2</sub> at 3 bar pressure with a total volume of 800 m<sup>3</sup>.

**Cathode Strip Chambers (CSC)** Multi-wire proportional chambers with cathode strip readout providing a spatial resolution of 80 μm [22] filled with a mixture of 30% Ar, 50% CO<sub>2</sub>, and 20% CF<sub>4</sub>, with a total volume of 1.1 m<sup>3</sup>.

**Resistive Plate Chambers (RPC)** Narrow gas gap detectors formed by two parallel resistive Bakelite plates, separated by insulating spacers. The gas mixture is mainly C<sub>2</sub>H<sub>2</sub>F<sub>4</sub> with small amounts of SF<sub>6</sub>.

**Thin Gap Chambers (TGC)** Similar to the multi-wire proportional chambers, except the anode wire pitch is larger than the cathode-anode distance. The gas mixture is 55% CO<sub>2</sub> and 45% n-pentane (n-C<sub>5</sub>H<sub>12</sub>) with a total volume of 16 m<sup>3</sup>.

These chamber technologies are used in two separate sub-systems within the muon systems. Firstly there is the trigger system covering the range  $|\eta| \leq 2.4$  which consists of RPCs in the barrel and TGCs in the end-caps [10]. The triggering systems are used for bunch crossing identification as well as level 1 triggering. Secondly the precision measurement system consists of MDTs which cover  $\sim 99.5\%$  of the area with the remaining area covered by CSCs [22]. These chambers make up concentric cylindrical layers. With the exception of the range  $1.0 < |\eta| < 1.4$  where there is a fourth extra (**E**) layer present, there are three main cylindrical layers labelled intuitively in increasing radial order as; inner (**I**), middle (**M**), and outer (**O**). The breakdown of muon spectrometer technologies can be seen in Table 2.3.

Region $ \eta $	Barrel		End-caps		
	0-1.0	1.0-1.4	1.4-2.0	2.0-2.4	2.4-2.7
Station I	MDT	MDT/TGC	MDT/TGC	CSC	CSC
Station E	MDT*				
Station M	MDT/RPC	MDT/TGC	MDT/TGC	MDT/TGC	MDT/TGC
Station O	MDT/RPC		MDT	MDT	MDT

\* Special MDT chambers mounted onto the end-cap toroids extend into the field transition region giving coverage of  $|\eta| = 1.4 - 1.6$ .

**Table 2.3.:** Muon Spectrometer chamber technologies and their locations in  $\eta$  within the ATLAS detector [22].

## 2.7. Trigger

At design luminosity there will be a bunch crossing rate of  $\sim 40$  MHz with some 23 interactions per bunch crossing, giving an interaction rate of  $\sim 1$  GHz. This constitutes a huge amount of data and it simply would not be possible to store this quantity permanently. The actual storage rate would need to be  $\sim 100$  Hz. This represents a rejection factor of  $\sim 4 \times 10^5$  [10].

To achieve this task events stored permanently must be carefully selected so as to retain ‘interesting’ events and reject those that are less ‘interesting’<sup>‡</sup>. This is the job of the ATLAS trigger system. The trigger system in ATLAS is a three level system with each successive level refining the selection from the previous. The first level trigger, called the **Level 1** trigger, is the hardware based trigger and makes its decisions based on reduced granularity information. A variety of level 1 triggers are defined and may trigger on information from any of the hardware elements described earlier in this chapter and also on information from the minimum bias trigger scintillator (see next section). The level 1 trigger reduces the rate to  $\sim 75$  kHz [10].

<sup>‡</sup>The notion of an interesting event is a subjective one and will depend on an individual analysis. For this reason there are a variety of triggers available.

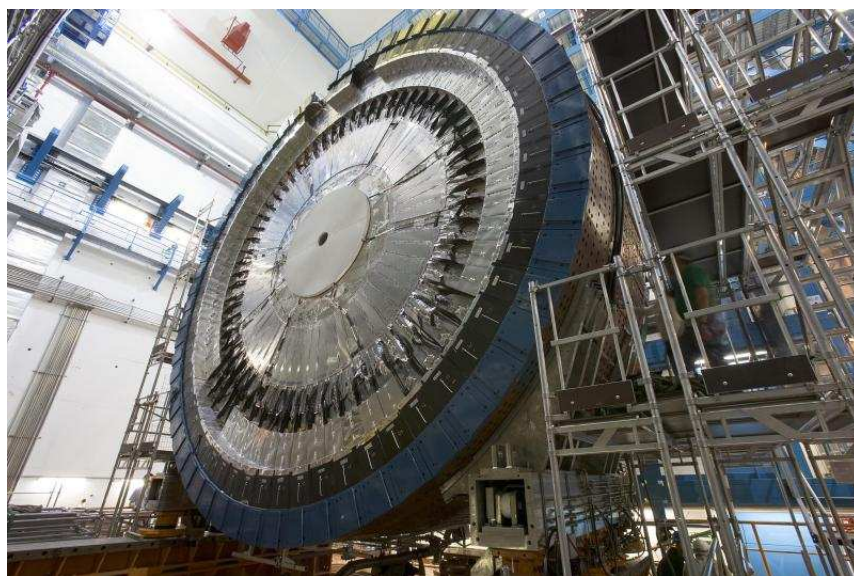


Unlike at level 1, the following two levels are software based triggers. Next in line is the **Level 2** trigger. The trigger at level 2 has access to the full detector granularity and full event information but will usually operate on regions-of-interest (**ROI**) picked out by the level 1 trigger for speed. After the level 2 trigger the rate has now been reduced to  $\sim 1\text{kHz}$  [10].

The final stage of the trigger system is called the Event Filter (**EF**). The event filter uses modified versions of the full offline reconstruction software and has access to the most up to date calibration, alignment information, and magnetic field map [10]. The EF reduces the rate from that of the level 2 trigger by an order of magnitude to the required  $\sim 100\text{ Hz}$ .

## 2.8. The Minimum Bias Trigger Scintillator

The Minimum Bias Trigger Scintillators (**MBTS**), seen in Figure 2.16, are used to trigger at level 1 on minimum bias events. The system comprises two wheels which are located at  $|z| = 3560\text{ mm}$ . They are each segmented into two regions in  $\eta$ ; the inner with  $2.82 < \eta < 3.84$  and the outer with  $2.09 < \eta < 2.82$ . Each of these regions is further segmented into 8 equal units in  $\phi$  giving a total of 16 scintillating plates per wheel [23].



**Figure 2.16.:** One of the two MBTS wheels mounted on the LAr end-cap calorimeter (white circle) [24].

## Chapter 3.

# ATLAS Code Framework

The ATLAS code framework **Athena** is a Python based framework that sequentially runs tools. The sequence of tools run can be modified by the user by use of python input files referred to as job options (**JO**) files.

These tools are algorithms written generally in C++ by different members of the ATLAS collaboration specialising in particular areas of ATLAS physics (e.g. jets or muons etc.). These algorithms then all seamlessly plug into the Athena framework and run via a Python-C++ interface. In order to run within the Athena framework algorithms must inherit from the Algorithm base class and in doing so must define the member functions in Source 3.1.

```
StatusCode initialize();  
StatusCode execute();  
StatusCode finalize();
```

**Source 3.1:** The three member functions declared in the Algorithm base class that must be defined in any derived class. StatusCode is an ATLAS designed object defined in the following pages.

The initialize member function is called by the framework only once at the beginning of any job and is used primarily to set up the algorithm

by for example setting of member variables. While setting the member variables within the constructor may seem the logical choice, it is not possible within the Athena framework since the framework itself takes care of the construction of the algorithms. The job options files are the way in which the algorithms can take input from the outside world and the initialize function provides a convenient place from which to set the member variables from them.

The execute member function is essentially an event loop, that is to say it gets called exactly once *per event*. This is the place where the main workings of the algorithm are defined such that they can process the events sequentially.

Finally, the finalize member function, like initialize, is called only once, but this time at the very end of a job. It is usual for algorithms to make use of this function as a clean-up area. As such, code to clean up the memory allocated during execute or initialize is usually found here. In addition one may also find code to write to disk output files that have been filled during the execute method.

Each of these functions returns a StatusCode object which can take only one of the following values at any given time.

- SUCCESS
- FAILURE
- RECOVERABLE

While the way in which Athena responds to such StatusCodes can be configured by the user, the usual operation is that a StatusCode of FAILURE in the execute loop will cause Athena to terminate a job. A StatusCode of RECOVERABLE in the execution loop will cause Athena to skip

to the next event. StatusCode of SUCCESS obviously indicates normal operation conditions and so Athena takes no specific action except to run the next piece of code.

In addition to running the user specified tools, Athena has services that it can provide to the algorithms. These services include the transient data store which can be used by any algorithm in a job to store/edit data in memory and persistification services such as the histogram and ntuple services which facilitate the storage of data on disk. Also available is the messaging service. The messaging service allows algorithms to print messages to an output stream and assign a logging level to the output. The log level takes one of the following 5 values with increasing severity.

- VERBOSE
- DEBUG
- INFO
- WARNING
- FATAL

At run time the user can specify the logging level to which they wish to receive output and only those messages holding a level equal or lower (in the above list) than the desired level are reported.

# Chapter 4.

## ATLAS Fast Simulation (Atlfast)

This chapter introduces the idea of detector simulation and details the reasoning for the development of a fast simulation tool before an in-depth discussion of its inner workings.

### 4.1. Introduction

For comparison of the data coming from the ATLAS experiment we can use simulated events. These events can be created using a chain of software tools. Monte Carlo event generators as outlined in Section 1.6 contain our current best understanding of the underlying physics of any given process. They output a list of final state particles that can be stored in a file or used on-the-fly. These particles then go through a detector simulation created using a software toolkit describing the GEometry ANd Tracking (Geant4) [25]. Geant4 simulates the passage of particles such as those produced in high energy collisions as they penetrate matter. Within the

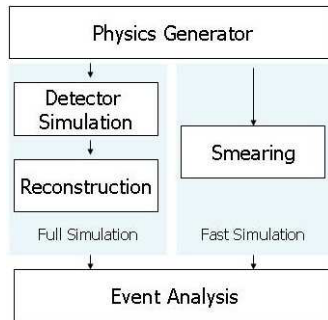
ATLAS simulation, Geant4 is used to simulate the various detector specific effects such as ionisations, Compton scatters, changes in the quantity of detecting material as in the crack regions etc.. Here extra particles may be added to the list to take account of said effects before their energy deposits/track hits are computed. Next is the digitisation where the detector hits and photomultiplier output are simulated from the list of simulated particles' hits and energy deposits. The simulated event is now in a similar condition to a real one and as such can undergo the same reconstruction code that a real event would.

The simulation stage of this full chain can be very computer intensive and as such take up to  $\sim 30$  mins per event. This is not a major problem for standard and general datasets as they are produced centrally in large production systems. Where this becomes a problem is for datasets that are required for a quick study or non standard datasets. In such cases, going through the central production system can be impractical. What is required is a faster route to simulate any event. This is the motivation behind the **Atlfast** software.

Atlfast\* is an abbreviation of ATLAS fast simulation. It aims to reproduce the physics distributions that one would see if using the full simulation as accurately as possible whilst keeping the speed of the simulation as fast as possible. The nature of the speed increase over the full simulation comes from the fact that Atlfast bypasses both the reconstruction and simulation stages as depicted in Figure 4.1. Atlfast instead takes the events straight from the Monte Carlo generator and smears the particles so as to reproduce the effects of a finite detector resolution.

---

\*Strictly should be Atlfast I since the emergence of Atlfast II but throughout this document referred to as just Atlfast



**Figure 4.1.:** Diagram showing routes between event generator and data analysis with the full simulation on the left and Atlfast fast simulation on the right. Atlfast replaces the need for separate detector simulation and reconstruction code by simply smearing the physics input.

Atlfast is formed of four packages (three main packages and one test package) as seen in Table 4.1. In Table 4.1 the dependency goes upwards.

Package	
<b>AtlfastEvent</b>	Code useful for describing a given event, e.g. cells, clusters & tracks.
<b>AtlfastUtils</b>	Utility code and function objects for use in Atlfasts algorithms
<b>AtlfastAlgs</b>	Top Level package defining the algorithms that make up the Atlfast program
<b>TestAtlfast</b>	Test package designed to monitor the Atlfast code using the ATLAS code testing framework, see Chapter 5

**Table 4.1.:** Table showing the four packages that make up the Atlfast software listed in dependency order and separated into the test package and the three main packages

## 4.2. Algorithms

Atlfast proceeds as a chain of C++ algorithms in sequential order. The sequence is shown in Table 4.2. A brief description of each algorithm is given in the following sections, while a full flow diagram of each can be seen in Appendix A.



Atlfast Sequence	
<b>GlobalEventDataMaker</b>	Global data i.e. B field strength
<b>CellMaker</b>	Make calorimeter cell map
<b>TrackMaker</b>	Make tracks
<b>ClusterMaker</b>	Make clusters based on cells
<b>ElectronMaker</b>	Make electrons
<b>PhotonMaker</b>	Make photons
<b>MuonMaker</b>	Make muons
<b>ElectronIsolator</b>	Sort the electrons into isolated and non-isolated
<b>PhotonIsolator</b>	Sort the photons into isolated and non-isolated
<b>MuonIsolator</b>	Sort the muons into isolated and non-isolated
<b>JetMaker</b>	Make Jets
<b>AtlfastB</b>	Correct Jets
<b>EventHeaderMaker</b>	Information like $E_T^{miss}$ gets calculated here
<b>StandardHistogramMaker</b>	Make standard histograms

**Table 4.2.:** The sequence of algorithms making up the standard Atlfast simulation program.

Luminosity	B-Field On/Off	Calo barrel/end-cap division eta
Smearer rand seed	List of invisibles	storegate key for MC data
Adjust $E_T^{miss}$ for isolation On/Off	PDGID for magnetic monopoles	Just consider hard scatter On/Off
File name for det effects file	File name for muon misalignment file	File name for muon misalignment linear fit
Muon misalignment value		

**Table 4.3.:** Contents of the GlobalEventData

### 4.2.1. GlobalEventDataMaker

The GlobalEventDataMaker algorithm runs before anything else and performs two tasks. It sets up two singleton<sup>†</sup> objects which are globally accessible throughout the rest of the Atlfast software. Predictably the first of these objects is the GlobalEventData object which exists to store information about the run that may be needed by any of the following algorithms. The contents of the GlobalEventData can be seen in Table 4.3.

<sup>†</sup>A singleton is a class that can only ever have one instance created from it. This is achieved by having a private constructor and an internal pointer to itself.

	Forward	Barrel	Forward
Acceptance:	$-5.0 < \eta \leq -3.2$	$-3.2 < \eta \leq 3.2$	$3.2 < \eta \leq 5.0$
	$-\pi < \phi \leq \pi$	$-\pi < \phi \leq \pi$	$-\pi < \phi \leq \pi$
Granularity: ( $\Delta\eta \times \Delta\phi$ )	$0.2 \times 0.2$	$0.1 \times 0.1$	$0.2 \times 0.2$

**Table 4.4.:** Acceptance and granularity of the three Atlfast calorimeter sections

In addition to this, the GlobalEventDataMaker algorithm also sets up the MagField singleton object, the sole purpose of which is to hold the strength of the B field in Tesla.

#### 4.2.2. CellMaker

CellMaker is responsible for setting up the calorimeter formed of a central barrel and two forward sections. The transition between the two is one of the variables stored in the GlobalEventData and is  $|\eta| = 3.2$ . Atlfast has a single layered calorimeter that is used solely for the reproduction of jet objects so as to provide a certain level of granularity. In Table 4.4 the acceptance of the three calorimeter sections can be seen in both pseudorapidity and azimuthal angle along with the respective granularities in  $\Delta\eta \times \Delta\phi$ . Using Table 4.4 it is possible to make a direct comparison between the Atlfast calorimeter model and that of the actual calorimeter (Table 2.1 & Table 2.2) which is simulated in the full simulation. Of note is that unlike the Atlfast calorimeter the actual calorimeter has more than one longitudinal sampling. The actual calorimeter is also generally more granular than the Atlfast calorimeter.

In addition to making the calorimeter, CellMaker is also responsible for transporting the particles through the solenoidal magnetic field and then populating the calorimeter with energy from the particle hits.

### 4.2.3. TrackMaker

The TrackMaker takes the list of particles from the generator/file and creates Atlfast track objects from those passing a minimum  $p_T$  cut of  $p_T > 500$  MeV and a maximum pseudorapidity cut of  $|\eta| < 2.5$ . The track objects then have a smearing applied to all track parameters ensuring that correlations between them are taken into account [26]. The nature of the smearing function depends on the type of particle that the track belongs to. Finally only those passing the impact parameter cuts  $z_0 < 40$  cm and  $d_0 < 5.05$  cm are kept as final tracks.

### 4.2.4. ClusterMaker

ClusterMaker is responsible for taking the populated cells and clustering them. This is done by use of a so-called jet finding algorithm<sup>‡</sup>. Atlfast has a few of these algorithms for the user to choose from, and it is the job of ClusterMaker to instantiate the correct one and run it over the collection of cells. The remaining unclustered cells are kept and stored separately and go into the  $E_T^{miss}$  calculation at the end, see Section 4.2.9.

### 4.2.5. ElectronMaker/PhotonMaker/MuonMaker

These algorithms are all instances of the DefaultReconstructedParticleMaker algorithm and as such the majority of their functionality is the same. They are called sequentially in the order given.

DefaultReconstructedParticleMaker starts by selecting stable particles of type electron/photon/muon (depending on which instance is running) that pass the criteria  $p_T \geq 0.0$  GeV and  $\eta \leq 100.0$  by default. For muons

---

<sup>‡</sup>see Section 4.3 below for more information about the jet finders.

$p_T \geq 0.5$  GeV is set by default in the job options. The kinematic cuts are purposefully loose by default to ensure as inclusive a sample as possible but are fully customisable via the Athena job options file.

The obtained list of particles is passed to a Reconstructor object specific to the type of particle selected (ElectronReconstructor, PhotonReconstructor or MuonReconstructor). It is the job of the reconstructor to create a reconstructed object from the generated truth particle. This is done by smearing the truth particle. The formula for the smearing depends on the type of particle and whether Atlfast is running in high or low luminosity mode, but generally takes the form of a parameterised resolution function as in the example shown in equation (4.1) (for central  $|\eta| < 1.4$  electrons in low luminosity mode). In this example the formula for the resolution is shown as the sum of the standard deviations of three Gaussian distributions. In order to get the overall  $\sigma/E$  a sample from each distribution is taken and then these are added together. This overall  $\sigma$  can then be multiplied by the four-momentum's components to give the smeared particle. The full list of resolution functions used in the electron and photon smearing algorithms can be seen in Table 4.5 and Table 4.6. For muons, the resolution is interpolated from a three dimensional (in  $p_T$ ,  $\eta$  and  $\phi$ ) 'look-up' table populated from the full simulation.

$$\frac{\sigma}{E} = \frac{12\%}{\sqrt{E}} \oplus \frac{0.245}{E_T} \oplus 0.7\% \quad (4.1)$$

One now has a list of reconstructed objects of type electron/photon/muon. These then optionally (disabled by default) pass through an efficiency function which tries to simulate the efficiency of the relevant detector. Finally the particles are subject to selection by a kinematic selector requiring that the  $p_T \geq 5.0$  GeV and  $|\eta| \leq 2.5$  by default. For muons  $p_T \geq 6.0$  GeV is

set by default in the job options. These cuts are all fully customisable via Athena job options.

Smearing Formula	
<b>Low Luminosity Mode:</b>	
$ \eta  < 1.4$	$\frac{\sigma}{E} = \frac{12\%}{\sqrt{E}} \oplus \frac{0.245}{E_T} \oplus 0.7\%$
$1.4 \leq  \eta  < 2.5$	$\frac{\sigma}{E} = \frac{12\%}{\sqrt{E}} \oplus \frac{0.306 \cdot (2.4 -  \eta ) + 0.228}{E} \oplus 0.7\%$
<b>High Luminosity Mode:</b>	
$ \eta  < 0.6$	$\frac{\sigma}{E} = \frac{12\%}{\sqrt{E}} \oplus \frac{0.245}{E_T} \oplus 0.7\% \oplus \frac{0.32}{E_T}$
$0.6 <  \eta  < 1.4$	$\frac{\sigma}{E} = \frac{12\%}{\sqrt{E}} \oplus \frac{0.245}{E_T} \oplus 0.7\% \oplus \frac{0.295}{E_T}$
$1.4 <  \eta  < 2.5$	$\frac{\sigma}{E} = \frac{12\%}{\sqrt{E}} \oplus \frac{0.306 \cdot (2.4 -  \eta ) + 0.228}{E} \oplus 0.7\% \oplus \frac{0.27}{E_T}$

**Table 4.5.:** Parameterised resolution functions for electron smearing in low and high luminosity modes.

Smearing Formula	
<b><math>\theta</math> smearing:</b>	
$ \eta  < 0.8$	$\sigma(\theta) = \frac{65 \text{ mrad}}{\sqrt{E}}$
$0.8 \leq  \eta  < 1.4$	$\sigma(\theta) = \frac{50 \text{ mrad}}{\sqrt{E}}$
$1.4 \leq  \eta  < 2.5$	$\sigma(\theta) = \frac{40 \text{ mrad}}{\sqrt{E}}$
<b>Low Luminosity Mode:</b>	
$ \eta  < 1.4$	$\frac{\sigma}{E} = \frac{10\%}{\sqrt{E}} \oplus \frac{0.245}{E_T} \oplus 0.7\%$
$1.4 \leq  \eta  < 2.5$	$\frac{\sigma}{E} = \frac{10\%}{\sqrt{E}} \oplus \frac{0.306 \cdot (2.4 -  \eta ) + 0.228}{E} \oplus 0.7\%$
<b>High Luminosity Mode:</b>	
$ \eta  < 0.6$	$\frac{\sigma}{E} = \frac{10\%}{\sqrt{E}} \oplus \frac{0.245}{E_T} \oplus 0.7\% \oplus \frac{0.32}{E_T}$
$0.6 <  \eta  < 1.4$	$\frac{\sigma}{E} = \frac{10\%}{\sqrt{E}} \oplus \frac{0.245}{E_T} \oplus 0.7\% \oplus \frac{0.295}{E_T}$
$1.4 <  \eta  < 2.5$	$\frac{\sigma}{E} = \frac{10\%}{\sqrt{E}} \oplus \frac{0.306 \cdot (2.4 -  \eta ) + 0.228}{E} \oplus 0.7\% \oplus \frac{0.27}{E_T}$

**Table 4.6.:** Parameterised resolution functions for photon smearing in low and high luminosity modes.

#### 4.2.6. ElectronIsolator/PhotonIsolator/MuonIsolator

As in the previous section, ElectronIsolator, PhotonIsolator and MuonIsolator are all specialised instances of the Isolator algorithm, and as such their functionality is very similar. They are called sequentially in the order given.

The Isolator algorithms create separate isolated and non-isolated collections of electrons/photons/muons in the transient data store. This is done by performing an isolation calculation on the reconstructed particles. Before this the reconstructed particles are first associated with clusters by searching for the closest unassociated cluster within  $\Delta R < 0.15$  of the particle.  $\Delta R$  is the distance measured in the  $\eta$ - $\phi$  plane, defined between two objects 1 and 2 as in equation (4.2), such that a line of constant  $\Delta R$  from an object forms a circle (in this plane) centred on the object.

$$\Delta R_{1-2} = \sqrt{(\eta_1 - \eta_2)^2 + (\phi_1 - \phi_2)^2} \quad (4.2)$$

Having done this the particle is then deemed to be isolated if the difference in energy between cells summed within a cone of  $\Delta R = 0.2$  of the smeared particle and the particle itself is below the threshold of 10 GeV. In addition no further reconstructed clusters must be found within a cone of  $\Delta R = 0.4$ . If both of these conditions are met then the particle is said to be isolated and put into the isolated collection, else it populates the non-isolated collection [26].

Also in the Isolator algorithms the summed energies in cones of various sizes ( $\Delta R = 0.1, 0.2, 0.3, 0.4$ ) around the particles are stored in the transient data store, as these are used by some analyses to calculate analysis level isolation cuts.

#### 4.2.7. JetMaker

Having made clusters and defined isolated electrons, photons and muons, JetMaker proceeds to make reconstructed Jet objects from the remaining clusters. JetMaker first checks to see whether a cluster is already associated

with an isolated electron, photon or muon and if so, does nothing and moves onto the next cluster. For unassociated clusters, JetMaker creates candidate jets by smearing the cluster four-momentum in accordance with the resolution functions in Table 4.7.

Smearing Formula	
<b>Low Luminosity Mode:</b>	
$ \eta  < 3.2$	$\frac{\sigma}{E} = \frac{50\%}{\sqrt{E}} \oplus 3\%$
$3.2 \leq  \eta  < 5.0$	$\frac{\sigma}{E} = \frac{100\%}{\sqrt{E}} \oplus 7\%$
<b>High Luminosity Mode:</b>	
$ \eta  < 3.2$	$\frac{\sigma}{E} = \frac{50\%}{\sqrt{E}} \oplus 3\% \oplus \frac{7.5}{E_T}$
$3.2 \leq  \eta  < 5.0$	$\frac{\sigma}{E} = \frac{100\%}{\sqrt{E}} \oplus 7\% \oplus \frac{7.5}{E_T}$

**Table 4.7.:** Parameterised resolution functions for jet smearing in low and high luminosity modes.

Non-isolated muons within a certain  $\Delta R$  of the candidate jet in the barrel and forward calorimeter are then added into the candidate jet. Both the  $\Delta R_{barrel}$  and  $\Delta R_{forward}$  are by default set to 0.400. Any added muons are then also removed from the non-isolated muons list so as to avoid double counting.

The candidate jet is now fully reconstructed but must pass certain acceptance criteria before it is stored as a final reconstructed jet. These acceptance criteria are  $p_T \geq 10$  GeV and  $|\eta| \leq 5.0$  by default but are configurable via the job options. If the jet passes these acceptance criteria then it is classed as a final reconstructed jet and is labelled before being added to the collection and stored.

If the reconstructed jet fails the acceptance criteria then it is not labelled and its four-momentum gets added to a running total of the rest of the unassociated clusters to which the unclustered cells are also added. This

will be used to calculate the  $E_T^{miss}$  in the EventHeaderMaker algorithm detailed in Section 4.2.9.

In Atlfast there are two distinct types of jet flavour label, described below.

1. **The Label** - The label reflects the flavour of the original parton and as such would label a b-jet as having originated from the b quark with 100% efficiency<sup>§</sup>.
2. **The Tag** - The tag is the result of simulating the b-tagging process (see next section) and depends on the efficiencies measured in the full simulation but will tag a b-jet as originating from a b quark some  $\sim 70\%$  of the time.

The labelling process modifies a specific quantity associated with the jet in order to reflect the type of the jet, i.e. those coming from an original b-quark, c-quark,  $\tau$ , or indeed the so-called light jet from the lighter quarks or gluons. This is important information as the different types of jets have different characteristics and analyses may require certain numbers of a particular type of jet.

In Atlfast the labelling is based on the truth information from the generator. The  $\Delta R$  between the jet and all the b-quarks and c-quarks with  $p_T \geq 5.0$  GeV, and  $\tau$ s with  $p_T \geq 10.0$  GeV in the event are calculated and the minimum of each is stored against the jet. If the  $\Delta R$  between jet and b-quark, c-quark or  $\tau$  is less than some maximum value (0.3 in all cases by default), then the jet's label quantity is set to that of the particle in the precedence order b-quark first followed by c-quark and then  $\tau$ . If no b, c or  $\tau$  particles are found within the required  $\Delta R$  limits then the jet's label

---

<sup>§</sup>In practise the  $\Delta R$  matching reduces this efficiency slightly from 100% as it is not exact.



quantity is set to zero and the jet is considered a light jet. After labelling, the jet is added to the jet collection and stored.

#### 4.2.8. AtlfastB

The AtlfastB algorithm is concerned with taking the jets from the JetMaker algorithm and producing objects that more accurately simulate the jets that would be reconstructed from the real detector. In this algorithm, jet tagging efficiencies are applied and the energy of the jets' has a correction applied based on the tag flavour. The efficiency and correction data for these operations are stored in files as part of the package.

Labelling the jets as described in the last section is useful to analyses performed on straight Monte Carlo simulated data, but maybe less so for analyses using the simulated data to get an idea of how real data might look. This is because in the full reconstruction code running on real data there is no way to get partonic level information about jets. In full reconstruction, b-tagging is carried out by tagging algorithms which attempt to predict if the b-quark was the original jet's parton based on detector level observables. Owing to a longer lifetime, this is usually done by looking for displaced vertices within the ID. Unlike the Atlfast labels they are never 100% efficient.

In Atlfast it is the job of the AtlfastB algorithm to simulate the outcome of the full reconstruction's b-tagging algorithms by assigning efficiencies for b-jet tagging and inefficiencies for non b-jet mis-tagging. These in/efficiencies are parameterised as a function of  $p_T$  separately for low and high luminosity modes [10]. Table 4.8 shows the nominal efficiencies averaged over  $p_T$  for b-jet tagging and the inefficiencies for tagging as a c-jet or other jet [10].

Efficiency	Low Luminosity	High Luminosity
$\epsilon_b$	60%	50%
$\epsilon_c$	10%	10%
$\epsilon_j$	1%	1%

**Table 4.8.:** b-jet tagging efficiencies and mis-tagging inefficiencies for low and high luminosity modes averaged over  $p_T$  [10].

Currently AtlfastB still runs tagging for legacy reasons, however the full simulation team wrote their own b-tagging simulator that runs upstream of AtlfastB and can also tag Atlfast jets. Both tags remain accessible to the user in the output file.

In addition to tagging, AtlfastB is also where the jet energy corrections are applied. Due to the finite cone size used to reconstruct them, some of a jet's energy can be lost outside and thus affect any measurements made of, say, invariant masses. The different flavoured jets have typically different lateral shower sizes and so the nature of this so-called out-of-cone correction depends on the tagged type of the jet (as well as its  $p_T$  and  $\eta$ ) [26].

#### 4.2.9. EventHeaderMaker

EventHeaderMaker creates an instance of an EventHeader object. An EventHeader object contains useful information about a given event. Such information includes event-wide variables such as the missing momentum, the scalar  $E_T$  sum (referred to as sumEt) as well as some event shape quantities. In addition to this it also stores the number of reconstructed particles of each type found in the event.

In order to store the missing momentum information it must first be calculated. In Atlfast the missing momentum is calculated summing the

four momenta as in equation (4.3).

$$\begin{aligned}
 E_T^{miss} = & -(\Sigma \text{UnusedCells} + \Sigma \text{IsolatedElectrons} + \Sigma \text{IsolatedPhotons} \\
 & + \Sigma \text{IsolatedMuons} + \Sigma \text{Jet} + \Sigma \text{NonIsolatedMuons} \\
 & - \Sigma \text{AssociatedJetNonIsolatedMuons})
 \end{aligned}
 \tag{4.3}$$

The SumEt is calculated in the same way by simple substitution of  $E_T^{miss}$  with SumEt in equation (4.3) and summing the transverse energies rather than using the four vector sum.

#### 4.2.10. StandardHistogramMaker

This is the final algorithm in the sequence and exists solely to create some standard histograms that can be used to debug Atlfast's output and use as regression points for testing changes to the physics. In most cases this means Multiplicity,  $P_x$ ,  $P_y$ ,  $P_z$  and  $E$  histograms for each of the standard Atlfast containers.

### 4.3. Jet Finding Algorithm Update

A jet finding algorithm can be somewhat of a misnomer since unlike leptons or quarks, jets are not pre-existing objects there to be *found* but rather their definition is dependent on the algorithm used to find them. One might therefore just as well refer to the collection of algorithms as jet defining algorithms.

There are many different types of such algorithms but broadly speaking they fall into two main categories.

- topological clustering
- kinematic clustering

The first type, commonly referred to as cone type algorithms attempt to maximise the energy contained within some geometric region (typically circular) defined in  $\eta - \phi$  space while the second type clusters particles in an event ordered in either  $p_T$  or  $\Delta R$ .

### 4.3.1. Atlfast's Pre-Existing Jet Finders

Atlfast previously utilized 3 different types of algorithm.

- Cone
- Kt
- Shared

The Kt algorithm was a kinematic clustering algorithm using the ‘KtJet’ [27] implementation of the Kt successive recombination algorithm. This was computationally slower than the newer implementation outlined in the next section running at a complexity of  $N^3$ . In the high multiplicity environment of the LHC this was considered to be a severe disadvantage. Cone was a simple topological clustering algorithm which clusters cells in circles of radius  $\Delta R$  in the  $\eta - \phi$  plane. Despite the similarity in name, it was *not* the same as the standard ATLAS Cone algorithm at the time but was instead an Atlfast proprietary algorithm. Shared was an attempt to share out the energy of cells which overlap with more than one cluster by  $\Delta R < 0.4$ . In this case the overlapping cells’ energy is shared out weighted by the clusters’ energy.

It is also worth noting that with the exception of the Kt algorithm the others are not infra-red safe. That is to say they are sensitive to the addition of soft inter-jet radiation. This includes the ATLAS Cone algorithm mentioned.

### 4.3.2. FastJet

The FastJet toolset [28] provides a fast ( $N \ln N$ ,  $N^2$ ) implementation of the longitudinally invariant Kt [29, 30], anti-Kt [31] and Cambridge/Aachen [32–34] jet algorithms as well as a collection of other common jet finding algorithms via its library of plug-ins. The faster approach comes from utilising methods from the computational geometry community [35]. Most noteworthy among the plug-in jet finders is the Seedless Infra-red Safe Cone (**SISCone**) algorithm [36]. This is an updated, more theoretically sound topological clustering algorithm which has no seed and is, as the name suggests, not sensitive to soft inter-jet radiation.

The generalised formulae for the longitudinally invariant kinematic clustering algorithms can be seen in equation (4.4), equation (4.5) and equation (4.6). In each case the algorithm defines a distance measure,  $d_{ij}$ , between all combinations of two particles. It also defines a beam distance,  $d_{iB}$ , and then finds the minimum of all the  $d_{ij}$  and  $d_{iB}$ . If the minimum is found to be  $d_{ij}$  then it merges the particles  $i$  and  $j$  according to the desired recombination scheme (typically the so-called E-scheme where the particles' four-momenta are summed). If however the minimum is  $d_{iB}$  then it declares particle  $i$  a final state jet and moves on to the next particle.  $R$  is a jet radius parameter usually taken to be of order 1. The formulae for the longitudinally invariant Kt, Cambridge/Aachen and anti-Kt algorithms can then be obtained from this generalised set of formulae by setting

$k = (1, 0, -1)$  respectively.

$$d_{ij} = \min(p_{Ti}^{2k}, p_{Tj}^{2k}) \frac{\Delta_{ij}^2}{R^2} \quad (4.4)$$

$$d_{iB} = p_{Ti}^{2k} \quad (4.5)$$

$$\Delta_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \quad (4.6)$$

### 4.3.3. Updating Atlfast's Jet Finding Algorithms

It was felt that in order for Atlfast to be generally more useful to the ATLAS community it should use the FastJet package to standardise the task of jet finding. FastJet provided theoretically superior and more choice of jet finder, via a common interface, with very little overhead. It was hoped that the migration of Atlfast to FastJet would help in the effort of convincing the wider ATLAS community to also make the switch. In this way the ultimate goal was to unify and modularise the jet finding between the full and fast simulations.

Implementing the switch to FastJet involved first linking the FastJet libraries into the Athena framework. This was done by use of a so-called 'glue' package written by the author. The glue package, called AtlasFastJet, when compiled links the external FastJet libraries into Athena's installation area by use of a symbolic link. Once in Athena's 'InstallArea' they are fully accessible not just to Atlfast but any ATLAS code that wishes to use them. This step then was necessary to help the migration of the full simulation as well.

Once FastJet was fully linked into Athena, its advantages were recognised by those within the ATLAS jet community and the use of its libraries via the author's glue package is now well established and common practice. This means that now the full simulation, Atlfast, as well as the full

reconstruction of data are indeed unified in the way in which they describe jets. This work also allowed the optimisation of several analyses using jets in both fast and fully-simulated samples by enabling easy (via standard interface) comparisons between a range of different jet finders and helped enable the move of the collaboration towards using the anti-Kt algorithm as the ATLAS default.

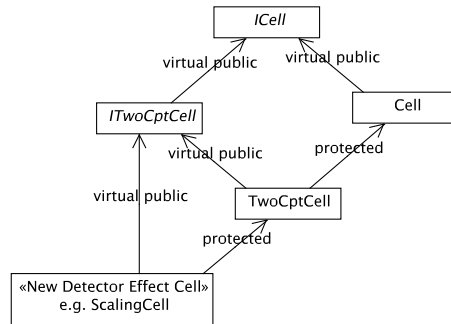
## 4.4. Custom Cells & Smearers

In an effort to compare the fast simulation with data as soon as possible it was decided that it would be advantageous to have more control of the calorimeter and smearers within Atlfast. ATLAS maintains a database of the condition of, for example, the cells in the calorimeter for any given run called the *conditions database*. The idea would be to automatically tap into this and be able to set up the Atlfast calorimeter and/or smearers accordingly.

### 4.4.1. Custom Calorimeter

In the real calorimeter, cells do not behave in a uniform way. Cells can occasionally register lots of activity irrespective of the energy distribution within the calorimeter. Such cells are referred to as ‘hot’. In addition, owing to whatever reason (e.g. broken readout) some cells are inactive and record no hits and are called dead.

To model these effects within Atlfast a new framework was implemented whereby all cells that populate the calorimeter sections have a common base class `Cell`, implementing the `ICell` interface, thereby making the cells more modular and easier to extend to either hot or dead. Figure 4.2 shows the



**Figure 4.2.:** UML style diagram showing the inheritance ladder for new custom cells

inheritance hierarchy of this design. Inheriting from the base Cell class is the TwoCptCell (or two compartment cell) which was included before the new framework to try and account for electromagnetic and hadronic energy separately within the cell. This functionality is used by the FastShower part of Atlfast which is currently not in use. This can nevertheless be inherited upon in a modular way with no adverse effect.

One upshot of this modular design is that there is built-in scalability. For example if it was decided later on that Atlfast was to model calorimeter noise (random low level pedestal activity registered in each cell on top of which the signal is situated) then the extension would require implementing the *NoisyCell*.

In addition to writing a NoisyCell class that inherits from ITwoCptCell, a few extra steps must be taken before the class is fully implemented. Since the framework is designed to be scalable, the construction of the actual NoisyCell at run time is the job of the CellFactory which has to be able to deal with any cell type that may be implemented in the future. For this reason, the CellFactory needs to be *aware* of the new cell type and so a mapping within the CellFactory also needs to be established as in Source 4.1. This maps the unique enum for the new cell type (defined in



header DetectorParameters.h) to a templated cellForge function seen below in Source 4.2.

```
m_cellTypeMap[DetectorParameters::Noisy] =
                                     &cellForge<NoisyCell>
```

**Source 4.1:** Mapping of problem type enum to templated cellForge function pointer.

Figure 4.3 shows the workings of the CellFactory. At its heart is the map of function pointers which point to different specialisations of the templated cellForge function as shown in the verbatim code (Source 4.2) below. This templated method is responsible for constructing the correct object.

```
typedef ITwoCptCell* (*cellForgePointer)(const CellDescriptor&,
                                       const std::vector<double>&);
template<typename T>
ITwoCptCell* cellForge(const CellDescriptor& CellID,
                      const std::vector<double>& params){
    return new T(CellID,params);
}
```

**Source 4.2:** Templated cellForge function which can return any class inheriting from ITwoCptCell.

This code, while initially seeming more convoluted, offers much more attractive scalability than the ever expanding list of ‘*if*’ statements common to factory programming. The job of adding a mapping to a new object type now simply becomes one of inserting a new element into a map.

The CellFactory contains a FileLoader object (see Section 4.4.2) which is responsible for reading in the user supplied (or conditions database) file and processing it into a mapping of regions where the custom cells/smearers need to be applied. In keeping with the modular nature of the framework the regions over which the desired effects are applied all implement the IRegion interface. The example of a TwoDRegion defines a region in a

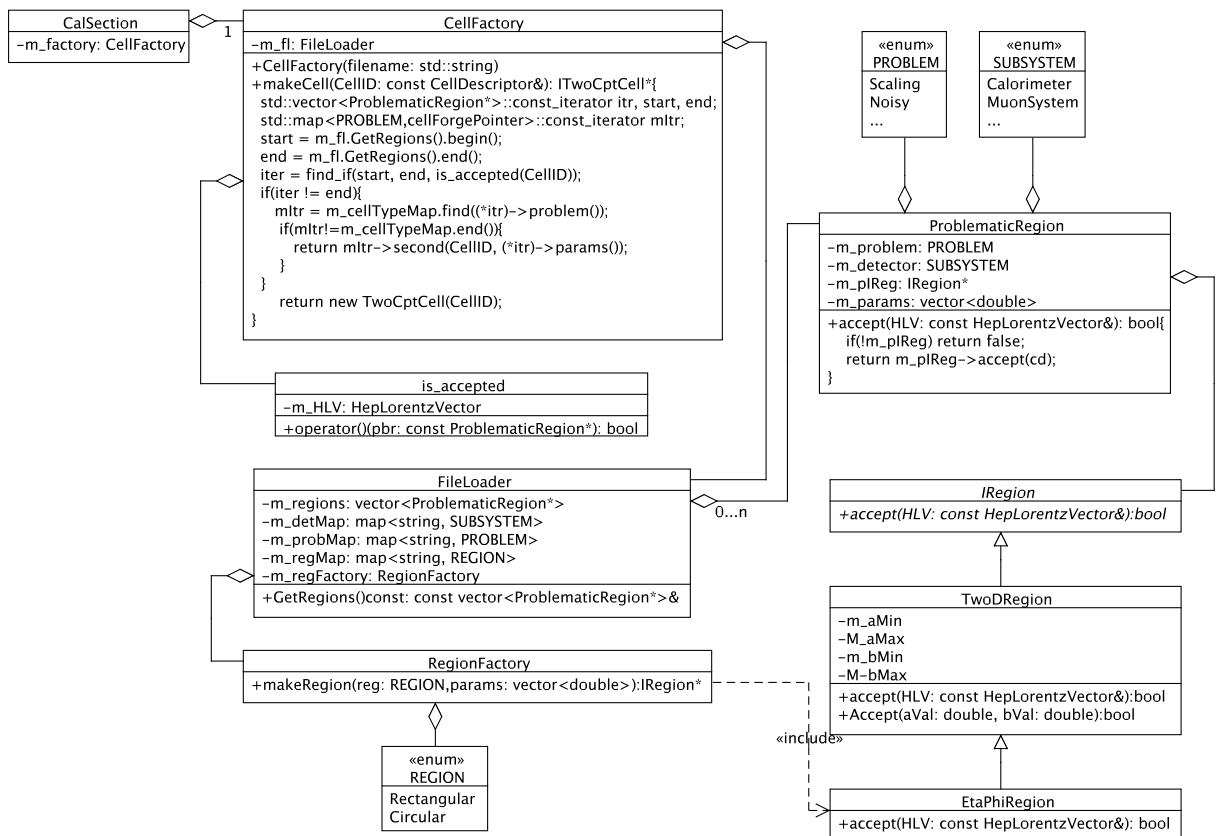


Figure 4.3.: UML style diagram showing the key components of the CellFactory update

general 2-parameter space and EtaPhiRegion inherits from this, extending the idea to the specific case of  $\eta - \phi$ .

As was the case when implementing a new cell type, if a new region type is required one must implement the region itself inheriting from IRegion and also insert a mapping into RegionFactory in the same manner as with CellFactory, since it's the RegionFactory that is responsible for the actual construction of the region.

#### 4.4.2. FileLoader

The FileLoader is the interface between the user and the factories and as stated above deals with the reading in of an input file (or conditions database)<sup>¶</sup> and using its content to control the factories behaviour. Whenever code relies on input from a user there is the possibility for unexpected behaviour. This is because one cannot predict *a priori* the full range of user inputs. The correct syntax for a problematic region can be seen in Source 4.3 with a hot cell example in Source 4.4.

```
DetectorSubSystem -p ProblemType ProblemParam1 ... -r RegionType RegionParam1 ...
```

**Source 4.3:** Syntax for the user input file to the FileLoader class.

```
Calorimeter -p Hot 2400.0 -r EtaPhi -2.5 2.5 -3.14 3.14
```

**Source 4.4:** Example entry for user input file to FileLoader setting hot cells of 2.4 GeV.

The customisation code needed to be robust enough to deal with any unforeseeable user input error. To this end the FileLoader will firstly ignore all blank lines. There is then a two level system in place in which the

<sup>¶</sup>The FileLoader was never integrated with the conditions database but a useful future study would be to write a converter to put the contents of the database in a form that the FileLoader could interpret.

FileLoader firstly checks to make sure it understands the type of detector subsystem, problem and region type that the user has asked for, and secondly it checks to make sure that the detector subsystem and the problem type are a valid combination. If any of the above tests are failed a clear and concise warning message is printed to the standard output, the offending line in the input file is ignored and FileLoader moves on to the next line.

The user can instruct Atlfast to use an input file by inclusion of the lines in Source 4.5 in their job options file.

```
from AtlfastAlgs.GlobalEventDataGetter import Atlfast__GlobalEventDataGetter
Atlfast__GlobalEventDataGetter().AlgHandle().DetEffectsFileName=<InsertFileNameHere>
```

**Source 4.5:** Lines that specify the user customisation file to be read into the FileLoader. These should be included in the job options file after the inclusion of the regular Atlfast lines.

The FileLoader, despite this complexity (for the sake of robustness), is very scalable and easy to update. It is merely a case of first updating the matrix that checks for valid combinations of detector subsystem and problem type in the initialise member function. An example of this can be seen in Source 4.6.

```
m_enumMat(DetectorParameters::Calorimeter,DetectorParameters::Hot)=1;
```

**Source 4.6:** Example of extending the Enum checking matrix allowing only combinations for which entry is unity.

Secondly, also in the initialise function one needs to add a mapping between the plain string that the user should enter in the input file and the unambiguous enum that will describe that string. A map exists for the subsystem type, the problem type and the region type, an example of which can be seen in Source 4.7.

```
m_probMap["Hot"] = DetectorParameters::Hot;
```

**Source 4.7:** Example of extending the problem type map which checks that the user inputted string matched the allowed value (HOT in this case).

### 4.4.3. Smearers and Reconstructors

For all Atlfast objects other than jets, the calorimeter doesn't get used. Instead the detector resolution effect is applied by the use of a smearer. In an effort to make the simulation more flexible a new version of the current smearer implementation was created. In keeping with the other updates to the code a modular design was favoured for the same reasons as before.

The new framework allowed for a more customisable approach. The first thing to note is that the name smearer was changed everywhere to reconstructor as it was felt this name better conveyed their role. In analogy with the CellFactory, there is a ReconstructorFactory which also contains a FileLoader object and is responsible for the construction of the reconstructors. Figure 4.4 shows a UML style diagram for the new framework. Under the new system the smearers of old were replaced by a CompoundReconstructor. The CompoundReconstructor is designed to capture all effects leading to the reconstruction of the particle. As the name suggests, this is a compound object containing a PrimaryReconstructor and a collection of so-called MarginalReconstructors. The MarginalReconstructors are reconstructors that can be run in addition to and sequentially following the PrimaryReconstructor. The user is free to define  $0 \dots n$  MarginalReconstructors.

The PrimaryReconstructor is the reconstructor that gets run before any of the marginal ones. It is itself a composite object consisting of a default reconstructor and a collection of valid ReplacementReconstructors. The

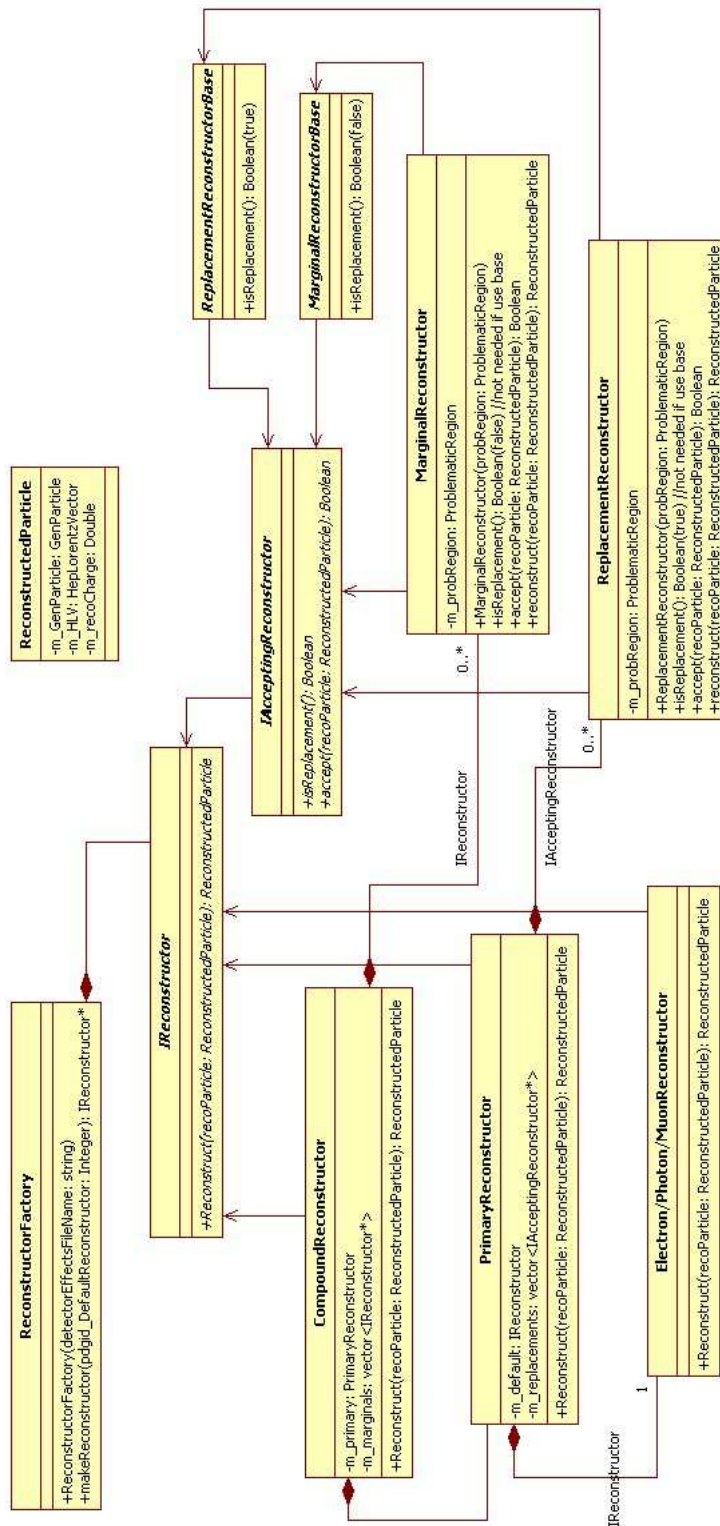


Figure 4.4.: UML style diagram showing the key components of the smearers update

default reconstructor depends on the type of particle that is being reconstructed but will typically be of type `ElectronReconstructor`, `PhotonReconstructor` or `MuonReconstructor`. These are nothing but the previously validated Electron, Photon and Muon Smearers from the old framework merely renamed as reconstructors. A valid `ReplacementReconstructor` is one that applies to the particle to which the default reconstructor applies and is designed to override the `DefaultReconstructor` in areas where it does not perform as well (such as the forward region). The `PrimaryReconstructor` will have a complete list of all user defined valid `ReplacementReconstructors`, each of which will have an acceptance defined in which it acts. The user is free to define  $0 \dots n$  `ReplacementReconstructors`.

When the `CompoundReconstructor` tries to use the `PrimaryReconstructor`, the `PrimaryReconstructor` will first check that there is no `ReplacementReconstructor` defined with an acceptance that means it would act on the particle. If this is the case then the `DefaultReconstructor` is used otherwise the first `ReplacementReconstructor` for which the particle is accepted is used.

As with the `CellFactory`, `ReconstructorFactory` can be made to recognise new reconstructors by addition of a map entry mapping the reconstructor/problem type enum to a templated `reconstructorForge` function pointer as in Source 4.8.

```
m_reconstructorTypeMap[DetectorParameters::MuonMisalignment] =  
    &reconstructorForge<MuonMisalignmentReconstructor>;
```

**Source 4.8:** Inserting an element mapping reconstructor/problem type enum to templated `reconstructorForge` function pointer.

The workings of the templated `reconstructorForge` are identical in construction to that of the `cellForge` seen in Source 4.2 and simply return an

instantiated object of the same type as the template parameter. The code can be seen in Source 4.9.

```

typedef IAcceptingReconstructor* (*reconstructorForgePointer)(
                                const ProblematicRegion&);

template<typename T>
IAcceptingReconstructor* reconstructorForge(
                                const ProblematicRegion &pbr){
    return new T(pbr);
}

```

**Source 4.9:** Templated reconstructorForge function which returns an instance of the class type passed as the template parameter so long as it inherits from IAcceptingReconstructor

Finally, the ReconstructorFactory contains a multimap which maps the particle PDG id number to the detector subsystems in which it can be found. This is to ensure that for any given default reconstructor type (electron, muon etc.) only those user defined reconstructors that affect the particular subsystem and hence particle are applied. This, like the map above, is filled during construction of the ReconstructorFactory. To extend the multimap, lines like the examples in Source 4.10 can be added as they are needed.

```

m_particleSubsystemMap.insert(
    std::pair<int,DetectorParameters::SUBSYSTEM>(13,
                                                  DetectorParameters::Calorimeter)
);
m_particleSubsystemMap.insert(
    std::pair<int,DetectorParameters::SUBSYSTEM>(13,
                                                  DetectorParameters::MuonSystem)
);

```

**Source 4.10:** Extending the multimap to create a mapping between muon (pdg id number 13 and the calorimeter and muon system. Therefore any additional user defined reconstructors that describe additional effects of the calorimeter or muon system will be included in the muon reconstructor.



#### 4.4.4. Dead Cells Example

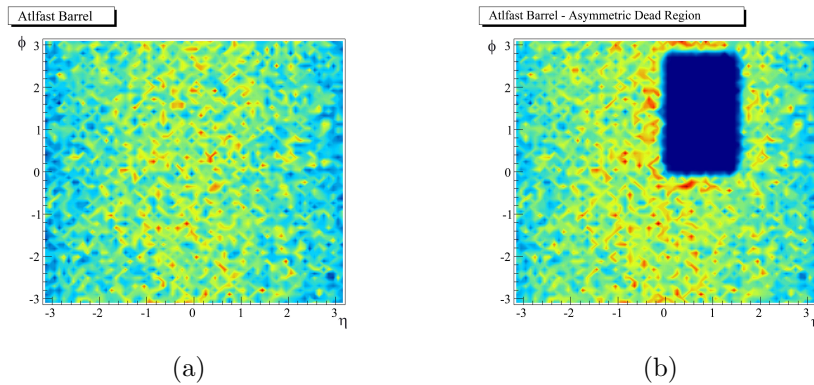
By way of example, consider an arbitrarily large section of the calorimeter dead for whatever reason. To simulate this, the custom cell ‘ScalingCell’, which scales the energy of a cell by the given scaling parameter can be used with a parameter of zero defined in the FileLoader configuration file. To create an asymmetric dead region, the entry in the file was as in Source 4.11.

```
Calorimeter -p Scaling 0.0 -r EtaPhi 0.0 1.5 0.0 2.7
```

**Source 4.11:** Entry in user input file to FileLoader for scaling part of the calorimeter to zero.

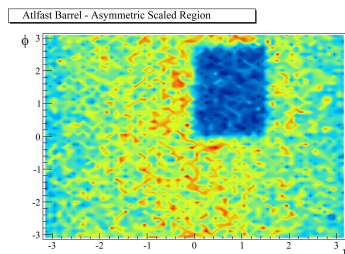
A pythia6 di-jet J1 (17 - 35 GeV) sample (see Table 6.3) was run through Atlfast with the custom calorimeter in place. Figure 4.5 shows rolled out  $\eta - \phi$  projections of the barrel calorimeter’s jet population density as determined by the SISCone algorithm with cone radius of 0.4. In Figure 4.5(a) the jet population density for the normal Atlfast barrel calorimeter can be compared to that with the dead region. In Figure 4.5(b) the simulated dead region is clearly visible.

In general Atlfast’s cells are larger than the cells in the real calorimeter. In addition the real calorimeter cells will typically have multiple samplings whereas Atlfast cells have just the one. The new cell framework can be used to try and approximate not just an entirely dead region as shown above but also the loss of a single real calorimeter cell, multiple real calorimeter cells (but not the entire Atlfast cell) or a sampling layer in the real calorimeter. The idea behind the approximation would be to utilise the ‘ScalingCell’ as before but this time changing the scaling parameter from zero in the case of the totally dead Atlfast cell to some value which could represent the fraction of energy lost in the dead part. This requires very little change



**Figure 4.5.:** Rolled out Atlfast barrel calorimeter jet population density plots for (a) normal Atlfast calorimeter and (b) custom dead calorimeter.

over the original dead cell example as it is merely a case of changing the scaling parameter in Source 4.11 from 0.0 to whatever is desired. Figure 4.6 shows the same asymmetric region as the previous example but this time scaled to half the activity (i.e. change of scaling parameter from 0.0 to 0.5).



**Figure 4.6.:** Rolled out Atlfast barrel calorimeter jet population density plot for scaled asymmetric region

#### 4.4.5. Hot Cell example

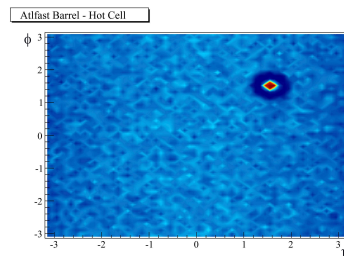
By way of another example one can simulate the effects of a hot cell in the calorimeter by using the custom defined ‘HotCell’ type. This allows the

user to specify the energy of a given calorimeter cell which it will keep for every event. In this example we consider the region  $1.5 < (\eta, \phi) \leq 1.6$  as hot with an energy of 50 GeV. This is achieved by using Source 4.12 in the input file. Note that the hot cell's energy parameter (fourth term) is in MeV in accordance with other ATLAS code.

```
Calorimeter -p Hot 50000.0 -r EtaPhi 1.5 1.6 1.5 1.6
```

**Source 4.12:** Entry in user input file to FileLoader for including a 50 GeV hot cell.

As before, a Pythia6 di-jet J1 (17 - 35 GeV) sample was generated and run through Atlfast on the fly with the custom calorimeter defined as above. Also as before, the SIScone jet finding algorithm was used with a cone radius of 0.4. Looking at the jet population density of the rolled out barrel calorimeter in Figure 4.7 a couple of interesting features can be seen.



**Figure 4.7.:** Rolled out Atlfast barrel calorimeter jet population density plot showing a single hot cell of 50 GeV.

Firstly the hot cell is clearly visible in the specified location far in excess of energy than any surrounding cell. Secondly there is a depleted region around the hot cell. The fact that this is a relatively low energy di-jet sample means that the hot cell energy specified is more energetic than the actual jets. The upshot of this is that the jet finder will find a jet more or less centred on that cell in every event. This will have the effect of sucking

in neighbouring particles which in the absence of the hot cell may have themselves been the jet centre.

## 4.5. Fast Trigger Simulation

Following on from the last section, in order to compare first data to the fast simulation on a competitive time-scale it was deemed important to be able to simulate the response of a given trigger quickly and be able to update the simulation to cope with any change in the trigger.

Given one of the first analyses to be performed with ATLAS data was the minimum bias analysis which the author had worked on, a fast trigger simulation was designed by measuring the single particle efficiency to fire the fully simulated minimum bias trigger using a single  $\pi$  Monte Carlo sample. This efficiency could then be applied sequentially to every particle in an event that falls within the MBTS acceptance.

The trigger used for minimum bias studies (see Chapter 7) was the L1\_MBTS\_1. This trigger merely requires a single hit anywhere within the MBTS. The efficiency of this trigger,  $\epsilon$ , was measured *per particle* binned in both  $p_T$  and  $\eta$  using equation (4.7). To achieve this a single  $\pi$  sample was used to ensure that the efficiency measured was indeed the per particle efficiency.

$$\epsilon = \frac{\text{No. passing L1\_MBTS\_1}}{\text{No. passing NoTrigReq}} \quad (4.7)$$

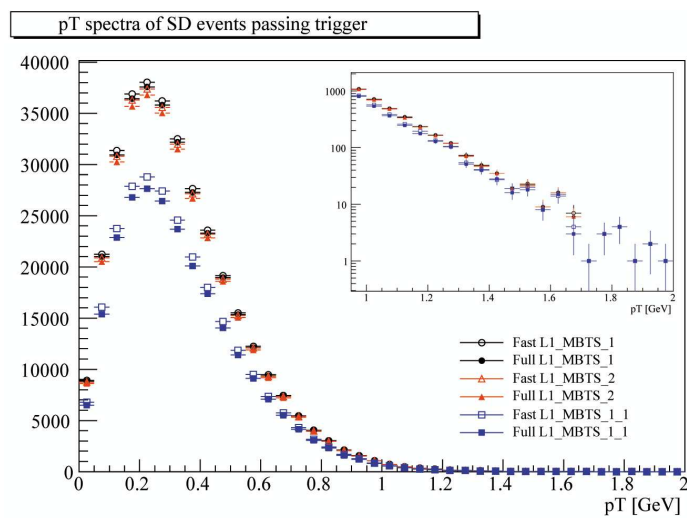
Where the NoTrigReq is a trigger that essentially passes every event. The fast trigger simulation could then be set up to use this two dimensional histogram as a look up table to apply the relevant ‘efficiency to trigger’

sequentially to any particle falling within the MBTS acceptance. This method was used with a full (*not single particle*) single diffractive sample to firstly recreate the L1\_MBTS\_1 trigger response and later to try and infer that of other MBTS triggers (L1\_MBTS\_2 and L1\_MBTS\_1\_1). The latter was performed by counting the number of particles that trigger L1\_MBTS\_1 in two bins, those with positive  $\eta$  and those with negative  $\eta$ . Since the requirements to fire L1\_MBTS\_2 and L1\_MBTS\_1\_1 are having 2 MBTS hits or having a single hit in both sides respectively, the responses could be inferred using equation (4.8) & equation (4.9). These efficiencies would be impossible to measure using a single particle sample.

$$\text{L1\_MBTS\_2} = \text{No. +ve } \eta > 1 \ || \ \text{No. -ve } \eta > 1 \quad (4.8)$$

$$\text{L1\_MBTS\_1\_1} = \text{No. +ve } \eta > 0 \ \&\& \ \text{No. -ve } \eta > 0 \quad (4.9)$$

Figure 4.8 shows the  $p_T$  spectra of pythia6 generated single diffractive events passing both the fully simulated and the fast simulated L1\_MBTS\_1, L1\_MBTS\_2 and L1\_MBTS\_1\_1 triggers. It can be seen that across the spectrum the Atlfast fast trigger simulation is overestimating that of the full simulation. As a first approximation and given the relatively small fully simulated trigger sample used in the parameterisation, the fact that the shapes are so similar is very encouraging. The parameterisation used was the one created during the initial testing phase and as a future study one could perform the same test with a more detailed parameterisation or further investigate the nature of the overestimation.



**Figure 4.8.:** Plot showing the pT spectra of single diffractive events passing the triggers; L1\_MBTS\_1 (black), L1\_MBTS\_2 (red) and L1\_MBTS\_1.1 (blue) for fully simulated trigger (closed) and the new fast trigger implementation (open). The inner box shows the tails on a logarithmic scale.

# Chapter 5.

## Code Maintenance

This chapter will introduce the ATLAS code maintenance system.

### 5.1. Motivation

The ATLAS code base is *very* large, circa 2023\* packages. In addition many of the packages exhibit dependencies on others. This means that even the slightest change in the code of one package may affect any number of packages that depend on it in ways that may not be instantly apparent but may manifest in *slight* changes in the physics output. This is clearly an undesirable position to be in and is the very reason that interfaces exist. The ATLAS code is no exception and also relies heavily on interfaces to make the code more stable. Interfaces however don't protect from changes to the underlying physics of the code, merely from changes to function names. It is also a fact of life that from time-to-time (ideally as infrequently as possible) even interfaces will change.

---

\*on 31<sup>st</sup> August 2010

Maintaining such a vast amount of code then, especially when each package may be the responsibility of a different author or set of authors, is clearly a very complex but very necessary task. The ATLAS solution is a simple framework that tests at run time<sup>†</sup> all the code together (i.e. with all dependencies) and regularly enough to allow any newly incorporated errors to be picked up. Such a framework exists for the ATLAS code and is called the Run Time Tester.

## 5.2. Introduction

The ATLAS Run Time Tester (**RTT**) is a Python coded framework for developers of ATLAS software which provides them with a convenient tool to test, on a daily basis, if changes to their code have affected the integrity of the package or the physics of the output. Test jobs can be run interactively or submitted to a batch system.

The RTT is run by the RTT team in batch mode over the ATLAS nightly software releases. These nightly releases (nightlies) are full builds of the entire ATLAS software overnight, reflecting the current status of the trunk (head) of the ATLAS code base. Nightlies are labelled rel\_0, ..., rel\_6, where rel\_0 roughly corresponds to the nightly built over Sunday evening - Monday morning etc..

The framework automates the following three step process:

- The setting up and running of a job<sup>‡</sup> or a chain of jobs be they parallel, sequential or combination thereof.

---

<sup>†</sup>Compile time checks are also made but are insufficient by themselves and not part of the run time tester discussed here.

<sup>‡</sup>These jobs are usually but not exclusively Athena jobs, where Athena is the ATLAS software Python run time environment, as discussed in Chapter 3



- The execution of post-job actions/tests<sup>§</sup> such as regression tests between the job's results and a benchmark, or just running a custom script to process the job's output in a specific way.
- The publishing of the job/test results to a globally accessible webpage for ease of viewing.

In the case where any of a developer's jobs should fail, the RTT can be configured to send an email notification of the failure to any number of designated email addresses, thereby providing a convenient and more persistent reminder about package shortcomings.

The RTT provides a library of post-job tools which perform commonly requested actions/tests. These can be seen in Table 5.1. In addition to these standard tools, the user is free to define their own as long as they conform to certain constraints such as being a Python module and not binding to any non-shelveable objects. For the full list of constraints for user tools, see the RTT manual [37].

CheckFileRunner <sup>  </sup>	Checks every pool file that your job outputs
DCubeRunner	Runs the DCube Framework for regression tests
DiffPoolFilesRunner	Does a diff on a pool file and a reference pool file
ExeRunner	Runs an executable
FileGrepper	Grep one or more input files for strings to find and/or veto strings not to find
FileComparator	Compares two text based files
PerfMonRunner	Runs the pefmon.py script
ROOTMacroRunner	Runs a ROOT macro

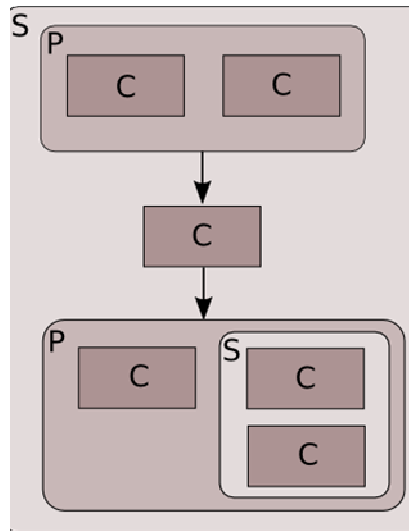
**Table 5.1.:** The current list of available RTT library tools

Stand alone jobs are themselves useful, however some situations call for a more complex arrangement of jobs. This is achieved within the RTT by

---

<sup>§</sup>The only distinction between an action and a test is that tests return a boolean pass/fail

<sup>||</sup>Included for completeness since it is run by default for all non-interactive RTT jobs



**Figure 5.1.:** Job Chain showing a complicated job structure with parallel (**P**) and sequential (**S**) chain elements/jobs (**C**) [37]

the use of *job chains*. A job chain implies a specific relationship between different jobs which the user is able to fully control and customise. The RTT understands and can nest any combination of *parallel* or *sequential* jobs allowing for complicated structures such as that shown in Figure 5.1.

With such a complicated job structure it is frequently required that the output from one job be shared with a subsequent one. This functionality is covered by the *chain file store*. Any of the jobs in the chain can store files or retrieve files from the chain file store. The following constraints apply:

- The chain name must be unique within the package.
- The file to be stored in the chain file store must have a unique name within the chain.

Finally, all jobs *must* belong to a **job group**. Job groups can contain any number of jobs and provide a way for developers to treat the jobs in a common way. For example while the specification for several jobs may be different, it might be the case that a certain macro should be run on the

output from each of them. This behaviour can be coded at the job group level using the action/test<sup>¶</sup> tools defined earlier in Table 5.1.

### 5.3. Shifters

To assist in keeping the RTT running smoothly, many of its individual components output log files documenting their status along with any problems, however minor, that they may have encountered. Part of the job of maintaining the RTT is to regularly check these logs in order to catch problems as soon as they develop. In addition to the log files, the RTT reports in real-time to a diagnostic webpage about things like its current available disk space, batch load and run status. These too need checking regularly. Such a task would dominate the time of the core developers (of which the author was a member) leaving them unable to respond to support issues or perform necessary development, and as such it is outsourced to a team of trained shifters. It was the job of the author to assume the newly created role of shift manager. The responsibilities of the shift manager included scheduling the shifts and assigning them to a particular shifter on a clearly visible online timetable, being available throughout each day in order to respond to and assist the shifters in any aspect of their shift work, to keep an eye on the RTT's daily running in an overview role to ensure that things were proceeding smoothly, to take preventative action to stop any problems arising, and to filter out and fix as many problems as possible before they got to the lead developer.

---

<sup>¶</sup>Note that actions/tests can also be defined at the job level if they are unique to that job. A test common to all jobs in a job group could be defined separately in each job's definition but it is obviously less obfuscated to define it once at the job group level

ID	Manager	Shift Report ID (Optional)	Text
165	mmash	Collated runs for week beginning Mon 18 Oct 2010	exceptionText Non-RTT exception:
164	Brinick Simmons		Started up new run: PCT devval. First run on rel_2.
163	mmash	Collated runs for week beginning Mon 11 Oct 2010	exceptionText Non-RTT exception:
162	mmash	Collated runs for week beginning Mon 4 Oct 2010	exceptionText CustomDriver: exception thrown while
161	mmash	Collated runs for week beginning Mon 27 Sep 2010	exceptionText Non-RTT exception:
160	mmash	Collated runs for week beginning Mon 20 Sept 2010	exceptionText Non-RTT exception:
159	Brinick Simmons		4 new runs set up. First running on the 17th: 16.0.X(-VAL) RTT/TCT

Figure 5.2.: Example of ELog web interface

## 5.4. ELog

The shifters report any problems they have found back to the developers for bug fixing and record them, along with run statistics and any other useful information, in an RTT ELog [38]. This was initially set-up (by the author) to speed up the feedback loop between developer and shifter and to act as a permanent record of the completion and results of the many tasks which the shifters were doing at the time.

Figure 5.2 shows the web interface for the ELog system. Owing to the nature of the ELog (as can be seen in its documentation [38]) this web interface is fully customisable to suit almost any need. In its original form, several fields were present representing the fact that the shifters had many more tasks to perform. Due to the streamlining efforts of the development team these tasks have been reduced to a minimum and as such at the present time each ELog entry merely has a large blank field for text entry.

As the information recorded and tasks performed by the shifters have been streamlined, the need for large shift teams has been somewhat reduced. The ELog is currently still in use as a means of documenting key tasks that need to be/are being performed by the development team in addition to being a general purpose log for the couple of remaining shifters.

## 5.5. Developer Control

Control of the RTT is maintained at the top level by the RTT development team who are responsible for ensuring that the framework is stable, performing any updates/modifications necessary and initiating the daily runs automatically via the scheduling tool ‘acron’. The RTT processes only those packages within the ATLAS code base which define that they have need of its services. The developer includes within their package a control XML file that defines the actions which the RTT must perform. In this way the nature of the testing of individual packages is left up to the package developers who it is assumed have a greater knowledge of their own code and therefore what should be tested. This modularity leaves the RTT development team free to concentrate on providing a stable and reliable service, free from worrying about the specifics of each of the  $\sim 2000$  packages.

Figure 5.3 shows an example piece of XML code for defining an Athena job. The RTT manual [37] details all the necessary and optional tags that one can supply in the configuration XML file along with examples. In addition the site offers an XML validation service which developers can use to check the validity of their configuration XML file prior to committing it into their package.

```

<jobList>
  <athena trendId="banana">
    <doc>This job does wonderful things.</doc>
    <doclink>http://www.more-info-on-my-job-here.org</doc>
    <numberedOnly/>
    <limitedNightlies/>
      <classification>
        <displayClass>.....</displayClass>
        <displayProcess>.....</displayProcess>
        <displayComponent>.....</displayComponent>
      </classification>

    <displayName>[NameToDisplayOnWebPage]</displayName>
    <commandLineFlags>myFlag1=True;myFlag2="hello"</commandLineFlags>
    <commandLineString>-s -c "myflag=2"</commandLineString>

    <preconfigOptions>[NameOfThePreJobOptionsToRun]</preconfigOptions>
    <options>[NameOfTheJobOptionsToRun]</options>
    <group>[NameOfYourJobGroup]</group>

    <queue>short</queue>

    <batchWallTime>10</batchWallTime>

    <dataset_info>
    <bytestream />
    <datasetName>/castor/cern.ch/user/n/nobody/myWonderfulDataset.root</datasetName>
    <datasetRepeat>3</datasetRepeat>
    </dataset_info>

    <dq2>
    <dq2site>CERN</dq2site>
    <dq2dataset>csc11.007502.singlepart_K3.recon.CBNT.v11004103</dq2dataset>
    <datatype>dc2</datatype>

    <dq2file>csc11.007502.singlepart_K3.recon.CBNT.v11004103._00001.root.5</dq2file>
    <dq2file>csc11.007502.singlepart_K3.recon.CBNT.v11004103._00002.root.1</dq2file>
    </dq2>

    <action position="5">[action_child_tags]</action>
    <test position="abc">[test_child_tags]</test>

    <keepFilePattern>ExactFileToMatch.txt</keepFilePattern>
    <auxFilePattern>MyWildcard*.txt</auxFilePattern>

    <athenaFlavour>athena.py</athenaFlavour>
    <castorSvcClass>BananaApple</castorSvcClass>
    <athenaCommonFlags />
    <alwaysRunPostProc />

  </athena>
</jobList>

```

**Figure 5.3.:** Example XML code for an Athena job showing all necessary and optional tags [37]

## 5.6. Web Interface

As mentioned earlier the results of the RTT tests are published to a webpage, the main page of this can be seen in Figure 5.4.



**Figure 5.4.:** ATLAS RTT mainpage showing buttons for results from RTT FCT and TCT.

This webpage (<https://atlasrftt.cern.ch>) requires the user to log in using the standard CERN Single Sign On (SSO) and as such is only accessible to ATLAS personnel. By signing into the website the user has his/her own unique view of the RTT results. The user may select which packages s/he wishes to see the test information about from the list on the left and is free to choose more than one. In this way the webpage is personalised to the individual. Selected packages are displayed in the main workspace window and will remain there from one sign in to the next. Before any are selected the user is presented with a blank workspace like that seen in Figure 5.5.

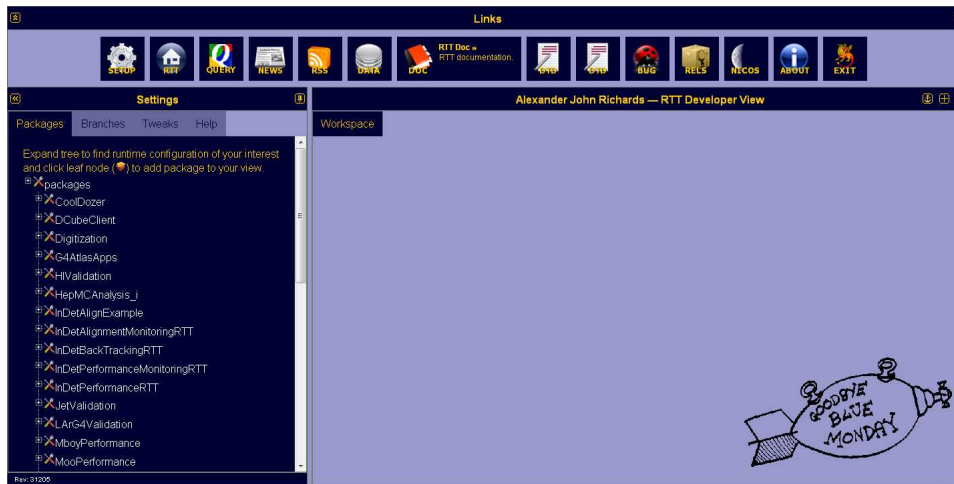
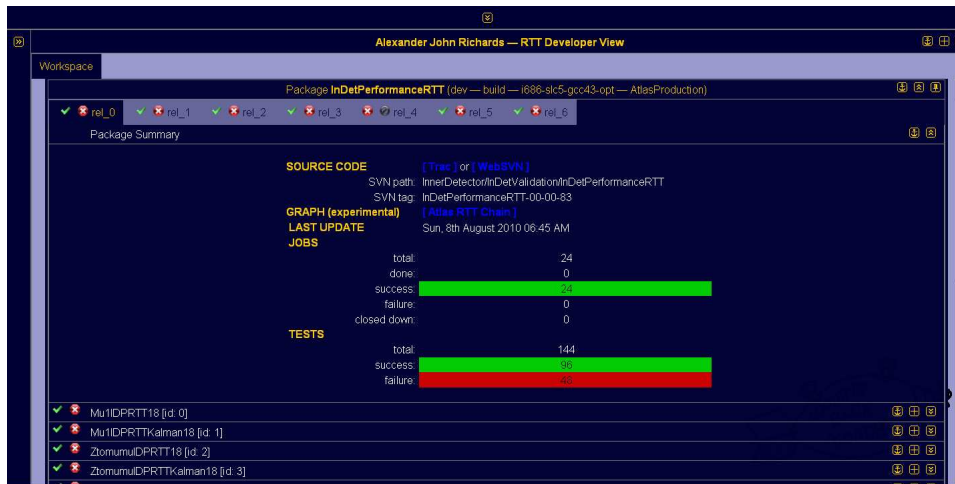


Figure 5.5.: RTT results page showing a blank workspace.

The RTT results page gives users a quick and convenient view of all the packages that they have selected along with their respective status for each nightly release (rel\_0, ..., rel\_6). The presence of two icons next to the nightly label give a quick view of the status of the package's jobs in that nightly; where the first refers to the overall number of jobs completed and the second, the overall number of tests returning pass. In both cases a green tick signifies that all jobs were completed or all tests passed. If any should fail then the green tick is replaced by a white cross in a red circle.

By selecting a given nightly release for a given package, one is presented with a view like that depicted in Figure 5.6. In this view a summary of the jobs run for this nightly is visible, in which one can see the number of jobs that were successfully completed or failed along with the number, if any, that were closed down by the RTT. Jobs would be closed down by the RTT if they had exceeded their allocated run time which since the RTT operates on a daily cycle is  $\sim 24$  hours. In addition to this job level summary, a test summary is also presented. It details, along with the total number of requested tests for all jobs in the package, the number that have been passed and the number that failed.

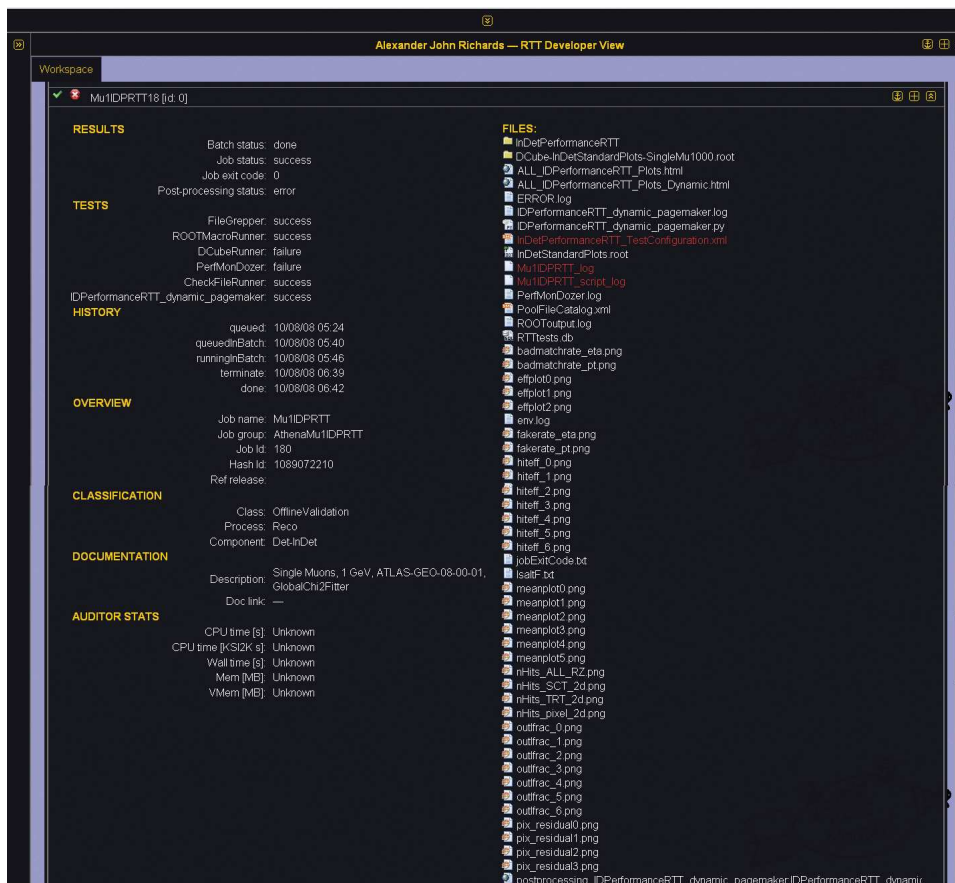




**Figure 5.6.:** RTT results page showing the results from the InDetPerformanceRTT package.

Finally in this view, as with the package view, two icons act as a quick summary of each of the individual job's performance. These icons with the individual job names appear listed below the summary information. The meaning of the icons is identical to that at the package level.

Clicking on one of these jobs will extend the information about the job itself, an example of which is given in Figure 5.7. The job view contains two columns of information. The column on the left contains useful statistics about the job such as cpu time etc.. It also contains information about which tests in particular failed, the job's unique id number and also a description of the job. The second column contains a list of all the files that after having been produced by the job, were requested to be saved by the package developer within the XML control file. This list is linked such that by clicking on any given file, the user can download that file straight from the results store and onto their computer.



**RESULTS**

Batch status	done
Job status	success
Job exit code	0
Post-processing status	error

**TESTS**

FileGrepper	success
ROOTMacroRunner	success
DCubeRunner	failure
PerfMonDozer	failure
CheckFileRunner	success
IDPerformanceRTT_dynamic_pagemaker	success

**HISTORY**

queued	10/08/08 05:24
queuedInBatch	10/08/08 05:40
runningInBatch	10/08/08 05:46
terminate	10/08/08 06:39
done	10/08/08 06:42

**OVERVIEW**

Job name	MuIDPRTT
Job group	AthenaMu1IDPRTT
Job id	180
Hash id	1088072210
Ref release	

**CLASSIFICATION**

Class	OfflineValidation
Process	Reco
Component	DetInDet

**DOCUMENTATION**

Description	Single Muons, 1 GeV, ATLAS-GEO-08-00-01, GlobalChi2Fitter
Doc link	—

**AUDITOR STATS**

CPU time [s]	Unknown
CPU time [kSI2K s]	Unknown
Wall time [s]	Unknown
Mem [MB]	Unknown
VMem [MB]	Unknown

**FILES:**

- InDetPerformanceRTT
- DCube-InDetStandardPlots-SingleMu1000.root
- ALL\_IDPerformanceRTT\_Plots.html
- ALL\_IDPerformanceRTT\_Plots\_Dynamic.html
- ERROR.log
- IDPerformanceRTT\_dynamic\_pagemaker.log
- IDPerformanceRTT\_dynamic\_pagemaker.py
- IDPerformanceRTT\_resConfiguration.xml
- InDetStandardPlots.root
- MuIDPRTT\_log
- MuIDPRTT\_script\_log
- PerfMonDozer.log
- PoolFileCatalog.xml
- ROOT output.log
- RTTtests.do
- badmachrate\_eta.png
- badmachrate\_pt.png
- efplot0.png
- efplot1.png
- efplot2.png
- env.log
- fakereate\_eta.png
- fakereate\_pt.png
- hit\_eff\_0.png
- hit\_eff\_1.png
- hit\_eff\_2.png
- hit\_eff\_3.png
- hit\_eff\_4.png
- hit\_eff\_5.png
- hit\_eff\_6.png
- hitEffCode.txt
- ksatF.txt
- meanplot0.png
- meanplot1.png
- meanplot2.png
- meanplot3.png
- meanplot4.png
- meanplot5.png
- nHits\_ALL\_R2.png
- nHits\_SCT\_2d.png
- nHits\_TRT\_2d.png
- nHits\_pixel\_2d.png
- outtrac\_0.png
- outtrac\_1.png
- outtrac\_2.png
- outtrac\_3.png
- outtrac\_4.png
- outtrac\_5.png
- outtrac\_6.png
- pix\_residual0.png
- pix\_residual1.png
- pix\_residual2.png
- pix\_residual3.png
- postprocessing\_IDPerformanceRTT\_dynamic\_pagemaker\_IDPerformanceRTT\_dynamic

**Figure 5.7.:** RTT results page showing results from a job within the InDetPerformanceRTT package.

## 5.7. DCube

One of the RTT library tools worthy of special mention is the DCubeRunner. This tool allows the user to schedule regression tests using an XML control file similar to that for the RTT itself. As with all RTT tests DCube is run after the job itself, usually comparing the output of the job to a reference file.

DCube allows a user to compare any root file (called the monitored file) with a reference file by plotting them both on the same axes. In addition the difference histogram can also be plotted. As well as simply plotting them side by side, a quantitative comparison is performed via calculation of p-values using the Kolmogorov-Smirnov, Pearson's  $\chi^2$  and a bin-by-bin method. Both the Kolmogorov-Smirnov and Pearson's  $\chi^2$  are well defined [39, 40]. For the DCube bin-by-bin test, the p-value is calculated according to equation (5.1) [41].

$$\text{p-value}_{bbb} = \frac{\text{No. Bins}_{\text{non-empty,equal}}}{\text{No. Bins}_{\text{non-empty,all}}} \quad (5.1)$$

Any combination of the above tests may be scheduled for any given histogram using the keywords 'KS', 'chi2', 'bbb' or simply 'all'. The user is free to set within the DCube configuration XML file the levels of the p-value for which the test is considered failed and for which it counts not as a fail but alerts the user with a warning. An example of the syntax for a DCube Reference file can be seen in Source 5.1. The XML file may be written from scratch or alternatively may be generated automatically using the DCubeClient package [41].

```
<?xml version="1.0" ?>
<dcube_client_config>
  <dcube branch="*" cmtconfig="*" install="*" jobId="1" project="*">
    <reference path="your_DCube_refFile_path"/>
    <ref_desc>
      PUT YOUR TEST DESCRIPTION HERE
    </ref_desc>
    <test_desc>
      PUT YOUR MONITORED FILE DESCRIPTION HERE
    </test_desc>
    <plimit fail="0.7" warn="0.9"/>
    <TDirectory name="/">
      <hist1D name="SimpleTest" tests="KS,chi2,bbb" type="TH1D"/>
    </TDirectory>
  </dcube>
</dcube_client_config>
```

**Source 5.1:** Example syntax for a DCube XML configuration file.

DCube will produce an output webpage that can be viewed directly from the RTT results webpage for the corresponding job. An example of the DCube output can be seen in Figure 5.8. The output page shows a summary of the calculated p-values for each of the three statistical tests requested between monitored and reference data. The breakdown is conveniently colour coded in order to make it clear how many histograms had p-values consistent with the user's definition of a pass and likewise for warning and failure. Summaries are made for directories contained within the data files and these can be expanded by simply clicking on them. Clicking on any given histogram name on the webpage will show both the monitored and reference histograms with the same name overlaid on the same axes for a visual comparison as in Figure 5.9. The user may, by simply clicking the 'diff' button, also request the difference between the two to be plotted.

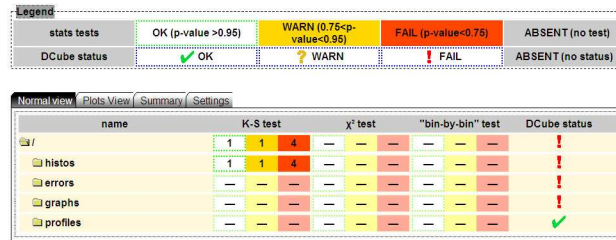


Figure 5.8.: Example of main DCube output webpage.

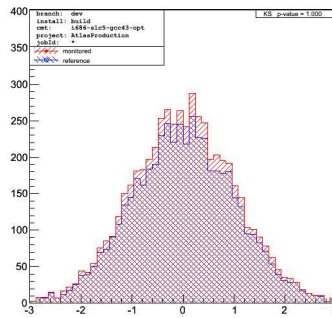


Figure 5.9.: Example of regression test histogram.

As with any of the tools within the RTT framework, the DCubeRunner tool is setup by the user within the RTT XML control file. The syntax for initiating the running of the DCubeRunner tool as a test after a job can be seen in Source 5.2.

```
<test>
  <modulename>RttLibraryTools</modulename>
  <testname>DCubeRunner</testname>
  <arg>
    <argname>DCubeCfg</argname>
    <argvalue>your_DCube_configFile.xml</argvalue>
  </arg>
  <arg>
    <argname>DCubeRef</argname>
    <argvalue>your_DCube_refFile</argvalue>
  </arg>
  <arg>
    <argname>DCubeMon</argname>
    <argvalue>your_DCube_monitoredFile</argvalue>
  </arg>
  <arg>
    <argname>DCubeJobId</argname>
    <argvalue>what_ever_you_want</argvalue>
  </arg>
</test>
```

**Source 5.2:** Example syntax for a DCube test within an RTT XML configuration file.

## 5.8. RTT Test Package Contributions

In this section direct contributions and contact with RTT test packages by the author are laid out.

### 5.8.1. TestAtlfast

As mentioned briefly in Table 4.1 there exists a package within the Atlfast code called TestAtlfast. This is a test package (in which the author made significant contributions) that is picked up by the RTT and used to run tests on the Atlfast code.

The tests include; running on both POOL data files as well as running on the Pythia6 generator output on the fly, validating the Atlfast jet distributions against an Atlfast reference using DCube, a comparison of Atlfast's jet distributions to those of full simulation, a check for memory leaks as well as other undesirable coding artefacts at runtime using valgrind [42] software, and regression checks using various data sets and various configurations of Atlfast again using DCube. These tests were designed to give excellent code coverage by utilising different datasets and thereby exercising lots of areas of the code (jet finding, tracks, muons etc.) as they would be in a real physics analysis.

The setting up of all these tests required a good understanding of the RTT and DCube, something with which the author gained a lot of experience and about which frequently advised others. An XML configuration file was of course needed for both the RTT and the DCube jobs run therein. In this area the author was a major contributor and indeed the original author of the DCube XML file. One of the author's responsibilities was in maintaining the TestAtlfast package and checking the output of the jobs. This was made easier due to the overlap of this work with the author's work as an RTT developer and shift manager.

### 5.8.2. GeneratorsRTT

In addition to significant contribution to the TestAtlfast package, the author was the package manager of a second RTT test package called GeneratorsRTT. As the name suggests, this package was designed as a common test bed for all generator testing within the RTT. Unlike the TestAtlfast package which was more or less an extension of the other Atlfast packages, having the same developers, GeneratorsRTT was more complicated as the

author was not an author of any generator software. In addition, generally the authors of one generator do not work on the code for another. Bringing all the generators together into a single common testing environment was therefore author's main responsibility.

In practical terms, the teams from a particular generator would provide the tests that they wanted performed and it was down to the author to implement the tests within the RTT framework and inform the generator teams where the results could be found. Owing to the complexity and number of individual generators and the need for relatively specialised knowledge, it was left as the responsibility of the generator teams to check the output of the tests and obviously fix any bugs highlighted by them. Any change to the tests themselves would then be passed on to the author who would implement them accordingly.



# Chapter 6.

## Supersymmetry Search

This chapter introduces an early data search for physics beyond the SM, specifically an extension known as supersymmetry. It introduces the idea of supersymmetry, its motivation as well as consequences and finally shows the validity of a search channel in which the author participated for a year.

### 6.1. Introduction

Supersymmetry is the as yet unproven extension to the standard model that predicts a symmetry between fermions and bosons. Since none of the current standard model particles can be their own super-partner this forces the introduction of at least as many particles again. In the case of the Higgs we add two fermionic super-partners and therefore have two scalar Higgs doublets. This enables us to avoid the anomaly that arises with just a single doublet whilst at the same time having a Yukawa coupling to both up-type and down-type quarks.

Supersymmetry was first put forward as a way of solving the hierarchy problem of the standard model in which the Higgs boson mass is hugely

increased by the divergence of the radiative corrections. With Supersymmetry, an exact cancellation can occur between the fermionic and bosonic loops. It has however, in addition to this, many nice features such as allowing for the unification of the couplings at about the GUT scale  $\sim 10^{15}$  GeV as well as providing a good dark matter candidate.

Supersymmetry however is not an exact symmetry of nature. If it were then we should have seen these supersymmetric particles (**sparticles**) already. Supersymmetry if it exists in nature then must be a broken symmetry allowing the sparticles to be much heavier than the regular particles of the standard model.

## 6.2. R-Parity

The most general form of the superpotential contains terms that would allow for violation of baryon or lepton number. The outcome of such terms would be that the proton decay would not only be allowed, but if these terms are unsuppressed then the proton lifetime could be very short. Since the lifetime of the proton has been experimentally determined to be in excess of  $10^{32}$  years, such terms must be either eliminated or highly suppressed. In order to eliminate these terms in the superpotential one can (but doesn't have to) introduce a new symmetry, the so-called 'R-parity' which unlike baryon number etc. is a multiplicatively conserved quantum number defined as equation (6.1),

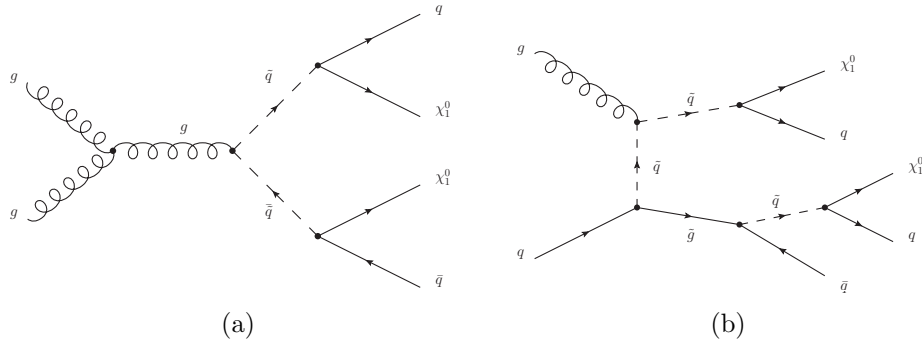
$$P_R = (-1)^{3(B-L)+2S} \tag{6.1}$$

where B, L and S are the baryon, lepton and spin quantum numbers respectively. In essence then it can be seen that all SM particles including the Higgs boson have R-parity of 1 ( $P_R = +1$ ) while all SUSY particles have R-parity of -1 ( $P_R = -1$ ). With the requirement that R-parity be conserved therefore, the proton can retain its stability. In addition, due to the multiplicative nature of the R-parity quantum number any SUSY interaction vertex must have pairs of sparticles. This leads to the observation that the decay of any sparticle must itself produce another sparticle. With this simple observation it can be seen that the lightest of the sparticles must be stable since it has no lighter sparticle to decay into. This stable lightest supersymmetric particle (**LSP**) is expected to be weakly interacting and escape the detector, like the neutrino, albeit much more massive. If Supersymmetry really exists and R-parity is truly a conserved quantum number then because of the stability of the LSP there should be LSPs left over from the big bang. So-called 'dark matter' is expected to consist of weakly interacting massive particles (**WIMPs**) and the LSP is considered a prime candidate.

### 6.3. 0-Lepton Jets and Missing Transverse

#### Energy $E_T^{miss}$ Channel

If R-parity is conserved then any SUSY interaction vertex must contain two sparticles. This means that when producing squarks or gluinos at the LHC they must be pair produced. The decay of gluinos goes via a squark  $\tilde{g} \rightarrow q\tilde{q}$  whereas, the decay of the squark will proceed via a neutralino or chargino  $\tilde{q} \rightarrow q\tilde{N}_i$  or  $q'\tilde{C}_i$  assuming a mass hierarchy of  $M_{gluino} > M_{squark}$  [43]. Eventually of course the sparticle's decay chain will end with the LSP. This will escape the detector in a similar way to neutrinos, meaning that



**Figure 6.1.:** Example signal Feynman diagrams for (a) the 2-jet +  $E_T^{miss}$  final state and (b) the 3-jet +  $E_T^{miss}$  final state.

there will be real  $E_T^{miss}$  in the event. Due to the relative mass scales of the LSP and neutrino it is expected that a sparticle producing event will have a much larger  $E_T^{miss}$  than a standard model event. Thus the typical signature of a SUSY event is jets + large  $E_T^{miss}$  + 0 or more leptons.

Most SUSY discovery searches concern themselves with finding the leptonic products of SUSY decay chains. The reasons for this are twofold: Firstly, leptonic final states are easier to trigger on. Secondly, the relatively low SM background for these events. There is a noticeable absence at this time of people studying the 0-lepton channels. This leaves a niche and provides reason in itself to study these processes. In addition, most 0-lepton SUSY searches concentrate on topologies with high jet multiplicity ( $\geq 4$ ) in order to reduce the backgrounds from vector boson + jet and QCD. In this study the possibility of utilising the lower 2 or 3 jet final states (Figure 6.1) is considered.

Many standard model processes can create final states with jets and  $E_T^{miss}$ . However due to the high squark and gluino masses, jets from SUSY events tend to have higher  $p_T$ . In addition, real  $E_T^{miss}$  (in standard model events) coming from neutrinos, will normally be accompanied by a lepton due to the weak production of the neutrino.

For this study, the backgrounds from QCD jets, top and vector boson + jets has been considered.

## 6.4. Simulated Samples

This section briefly describes the types of samples used in these analyses and at the end there is a table giving cross-section information about them labelled Table 6.4.

### 6.4.1. SUSY Sample

Since SUSY is as yet unproven and the parameter space for differing models is so large, most analyses concern themselves with only the mSUGRA benchmark points SU1-9 [44] which only require four parameters and a sign. These parameters are the common boson mass at the Grand Unification scale  $m_0$ , the common fermion mass  $m_{1/2}$ , the common tri-linear coupling  $A_0$ , the ratio of the Higgs vacuum expectation values  $\tan\beta$  and the sign of the Higgs potential parameter  $\mu$  [45].

For this analysis only SU3 was considered which has mSUGRA parameters shown in Table 6.1 and a leading order inclusive SUSY production cross-section of 18.59 pb. The mass spectra for the SU3 benchmark point can also be seen in Table 6.2 [44].

### 6.4.2. QCD Sample

Owing to the large  $p_T$  range covered by the background QCD di-jets and the relative cross-section of the lower  $p_T$  di-jet events, it is easier to sim-

	SU3
$m_0$	100 GeV
$m_{1/2}$	300 GeV
$A_0$	-300 GeV
$\tan\beta$	6
$\mu$	> 0

**Table 6.1.:** mSUGRA SU3 benchmark point parameters

Particle	SU3 Mass (GeV)
$\tilde{d}_L$	636.27
$\tilde{u}_L$	631.51
$\tilde{b}_1$	575.23
$\tilde{t}_1$	424.12
$\tilde{d}_R$	610.69
$\tilde{u}_R$	611.81
$\tilde{b}_2$	610.73
$\tilde{t}_2$	650.50
$\tilde{e}_L$	230.45
$\tilde{\nu}_e$	216.96
$\tilde{\tau}_1$	149.99
$\tilde{\nu}_\tau$	216.29
$\tilde{e}_R$	155.45
$\tilde{\tau}_2$	232.17
$\tilde{g}$	717.46
$\tilde{\chi}_1^0$	117.91
$\tilde{\chi}_2^0$	218.60
$\tilde{\chi}_3^0$	463.99
$\tilde{\chi}_4^0$	480.59
$\tilde{\chi}_1^+$	218.33
$\tilde{\chi}_2^+$	480.16
$h^0$	114.83
$H^0$	512.86
$A^0$	511.53
$H^+$	518.15
$t$	175.00

**Table 6.2.:** Sparticle mass spectra for the SU3 mSUGRA benchmark point

ulate the QCD background in a series of samples divided in  $p_T$  of the leading jet. These samples are arbitrarily labelled J0 - J8, the details of which can be seen in Table 6.3. The samples for this analysis are all generated with Pythia6 and combined by weighting them all to a luminosity

of  $100 \text{ pb}^{-1}$ . Since most SUSY searches require high jet multiplicities it is usually desirable to use a generator that can correctly predict the kinematics of these multi-parton final states such as ALPGEN [46]. The analyses described here require only a low jet-multiplicity, as such only Pythia was used which has a  $2 \rightarrow 2$  matrix element calculation with additional initial and final state radiation generated by the parton shower algorithms.

Top quark production was not included, instead it was confined to a separate sample which is discussed below.

Jet Sample	$p_T$ Range (GeV)
J0	8-17
J1	17-35
J2	35-70
J3	70-140
J4	140-280
J5	280-560
J6	560-1120
J7	1120-2240
J8	2240+

**Table 6.3.:** QCD di-jet samples and their respective  $p_T$  ranges.

### 6.4.3. $t\bar{t}$ Samples

The  $t\bar{t}$  processes are very important and form the dominant part of the SM background. The  $t\bar{t}$  samples used in these analyses were generated at next to leading order in the strong coupling using the MC@NLO generator [47, 48].

The  $t\bar{t}$  sample is further subdivided into 2 sub-samples containing those events which have fully hadronic decay products and those that don't.

#### 6.4.4. W/Z Samples

The W/Z samples are all generated with Pythia6 and consist separately of  $W \rightarrow$  all leptonic combinations and  $Z \rightarrow$  all leptonic combinations totalling seven separate samples as outlined in Table 6.4.  $W$  and  $Z$  going to hadrons is not considered as it looks the same as the QCD background and is completely swamped by it.

#### 6.4.5. Di-Boson Samples

Di-boson samples have smaller cross-sections and mainly contribute to the multi-lepton search channels but nevertheless are included here for completeness. In these analyses the  $WW$ ,  $ZZ$  and  $WZ$  processes (all generated in Herwig) are considered. For cross-section information refer to Table 6.4.



Sample	$\sigma$ (pb)
J0	$1.76 \times 10^{10}$
J1	$1.38 \times 10^9$
J2	$9.33 \times 10^7$
J3	$5.88 \times 10^6$
J4	$3.08 \times 10^5$
J5	$1.25 \times 10^4$
J6	$3.60 \times 10^2$
J7	$5.71 \times 10^0$
J8	$2.40 \times 10^{-2}$
$W \rightarrow e\nu$	$4.90 \times 10^1$
$W \rightarrow \mu\nu$	$2.90 \times 10^1$
$W \rightarrow \tau\nu$	$5.60 \times 10^1$
$Z \rightarrow e^+e^-$	$4.62 \times 10^1$
$Z \rightarrow \mu^+\mu^-$	$9.60 \times 10^0$
$Z \rightarrow \tau^+\tau^-$	$4.50 \times 10^0$
$Z \rightarrow \nu\bar{\nu}$	$4.10 \times 10^1$
$WW$	$2.45 \times 10^1$
$ZZ$	$2.10 \times 10^0$
$WZ$	$7.80 \times 10^0$
$t\bar{t}$ (Fully Hadronic)	$3.69 \times 10^2$
$t\bar{t}$ (Not Fully Hadronic)	$4.61 \times 10^2$
SUSY SU3	$1.859 \times 10^1$

**Table 6.4.:** Analyses samples and their respective cross-sections\*.

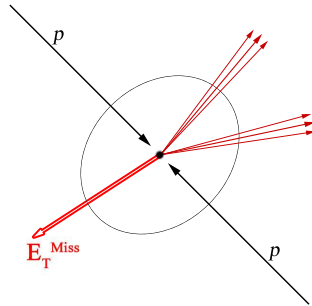
\* The cross-sections  $\sigma(W \rightarrow e\nu) \neq \sigma(W \rightarrow \mu\nu)$  etc. owing to the use of generator level filters applied when generating the samples. The filters require  $\geq 80$  GeV true  $E_T^{miss}$  as well as  $\geq 2$  cone 0.4 jets with  $p_T > 80, 40$  GeV.

## 6.5. Discriminating Variables

As with any discovery search at a particle collider, the aim is to try and detect the signal, however small, above the known backgrounds. While there are many ways of achieving this goal, some very complicated such as certain multivariate techniques, this study concerns itself with the simplest, namely the imposing of discriminating cuts. This section introduces the key physical quantities on which cuts will be placed in an attempt to enhance the SUSY signal and suppress the SM backgrounds.

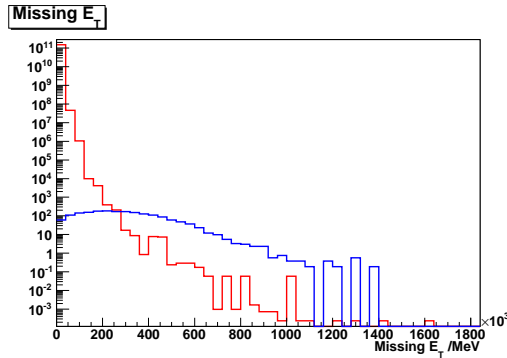
### 6.5.1. Missing Transverse Energy, $E_T^{miss}$

$E_T^{miss}$  is a vector in the transverse plane to the beam axis (the x–y plane) which is equal in magnitude but opposite in direction to the net (of all detected particles in the event) energy-momentum vector in that plane (Figure 6.2). While the initial momentum of the colliding partons in the direction of their propagation (z-axis) cannot be precisely known at a hadron–hadron collider, it is true that the initial momentum perpendicular to the beam axis is zero. As a direct consequence of conservation of energy-momentum one expects that in the final state this should also be the case; therefore, a net energy-momentum vector in the transverse plane can signify among other things the loss of a particle from the detector owing to it not interacting (e.g. neutrinos or the LSP etc.) and is referred to as missing transverse energy or  $E_T^{miss}$ .



**Figure 6.2.:** Diagram showing the  $E_T^{miss}$  vector balancing in the transverse plane the detectable matter in the event

In SM events  $E_T^{miss}$  can arise from neutrinos in the final state escaping the detector, from particles hitting dead detector material or being lost down the cracks or beam pipe or from energy mis-measurements. In SUSY events the same factors can contribute but in addition there is a large  $E_T^{miss}$  arising from the existence of the two LSPs in the final state. Owing to the heavy LSPs in the final state, one expects SUSY events to have a



**Figure 6.3.:**  $E_T^{miss}$  plot for J6 QCD (red) and SUSY (blue) jets.

relatively high  $E_T^{miss}$  when compared to SM events.  $E_T^{miss}$  then provides powerful discrimination between a SUSY and a SM event (Figure 6.3).

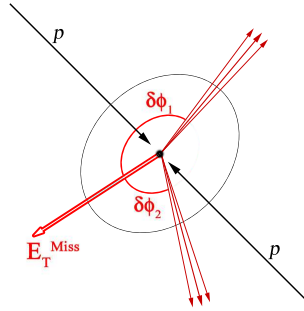
### 6.5.2. $\delta\phi$

One of the key backgrounds in any (but particularly a 0-lepton) SUSY search is the QCD jet background. In this background  $E_T^{miss}$  can be real, from the decay products of the hadronised partons, or fake from the jet energy mis-measurement as discussed earlier. In either case it is clear that there is a strong correlation between the direction of the jets and the direction of the  $E_T^{miss}$ . This however is not expected to hold true for the SUSY decay chains.

We can define therefore a useful quantity,  $\delta\phi$ , as in equation (6.2) that helps differentiate between the QCD jet background and our SUSY signal.

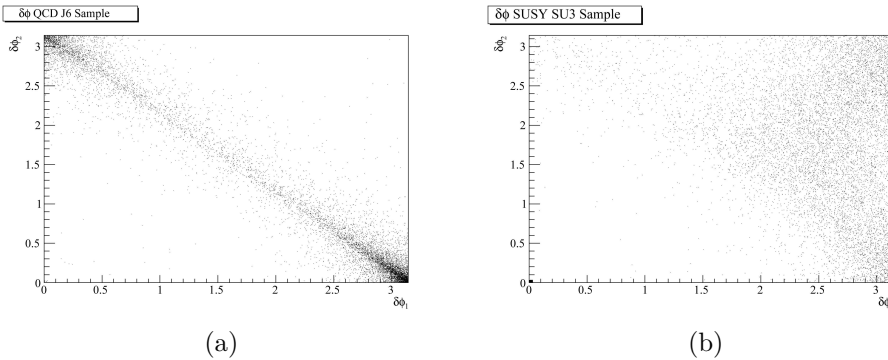
$$\delta\phi_n = |\phi_{jet\ n} - \phi(E_T^{miss})| \quad (6.2)$$

This quantity can be equally applied to any jet in the event and represents the difference in azimuthal angle  $\phi$  between the jet and the  $E_T^{miss}$  vector as shown in Figure 6.4.



**Figure 6.4.:** Diagram showing two jets and their azimuthal distance from the  $E_T^{miss}$  vector labelled  $\delta\phi_{1,2}$

In the  $\delta\phi_1 - \delta\phi_2$  plane the strong correlation of the  $E_T^{miss}$  with the QCD jet direction can be seen as in Figure 6.5(a). For the SUSY events (Figure 6.5(b)), this correlation is not as strong owing to the many possible decay chains of the produced sparticles.

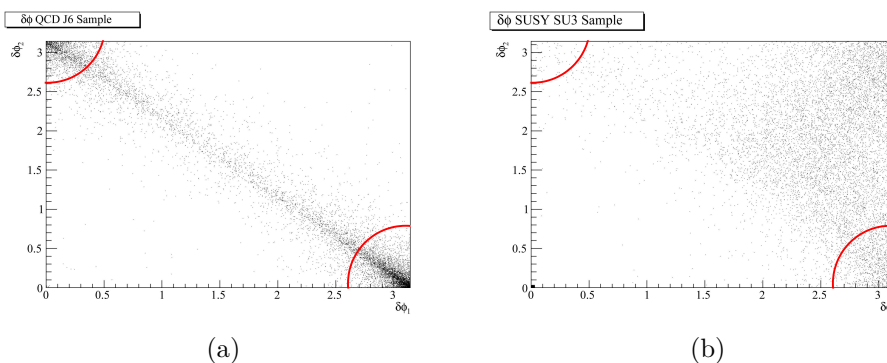


**Figure 6.5.:**  $\delta\phi_1 - \delta\phi_2$  plane showing (a) J6 QCD jets and (b) SUSY jets.

Two additional variables that can aid in the differentiation between QCD and SUSY events are shown in equation (6.3). These variables,  $R_1$  and  $R_2$ , define a radius from the bottom right, and from the top left corners of Figure 6.5 respectively and can be seen illustrated in Figure 6.6. As can be seen from Figure 6.6(a), these areas constitute the areas of highest jet

population density in the QCD samples.

$$\begin{aligned}
 R_1 &= \sqrt{\delta\phi_2^2 + (\pi - \delta\phi_1)^2} \\
 R_2 &= \sqrt{\delta\phi_1^2 + (\pi - \delta\phi_2)^2}
 \end{aligned}
 \tag{6.3}$$

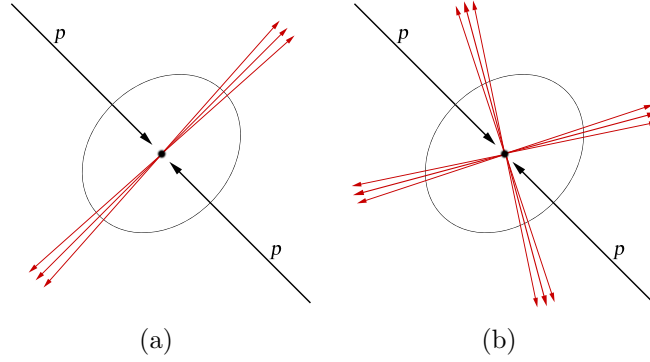


**Figure 6.6.:**  $\delta\phi_1$ - $\delta\phi_2$  plane showing (a) J6 QCD jets and (b) SUSY jets. The red lines show the effect of the  $R_{1,2}$  cuts

### 6.5.3. Transverse Sphericity, $S_T$

Sphericity  $S$  is a measure of the isotropy of an event in three-dimensional space. It is defined between 0 and 1 inclusive  $0 \leq S \leq 1$ , where  $S = 0$  would correspond to a perfectly back-to-back (dijet like) event and  $S = 1$  a completely isotropic one. ‘Transverse Sphericity’  $S_T$ , also known as circularity, is the application of the same concept to the two-dimensional plane perpendicular to the beam axis as shown in Figure 6.7.

$S_T$  then is a useful variable for helping remove backgrounds that are very back-to-back in nature.



**Figure 6.7.:** Diagrams showing the orientation of jets in an event for (a) transverse sphericity of 0 and (b) transverse sphericity of 1.

Transverse sphericity is defined as in equation (6.4),

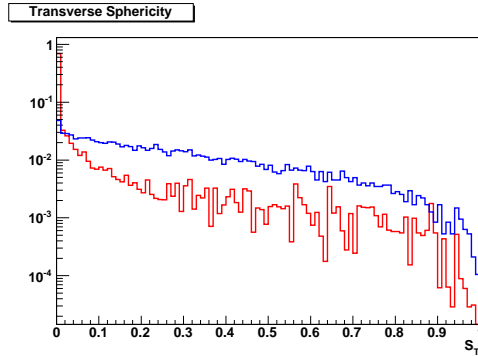
$$S_T = \frac{2\lambda_2}{(\lambda_1 + \lambda_2)} \quad (6.4)$$

where  $\lambda_1$  and  $\lambda_2$  are the eigenvalues of the  $2 \times 2$  sphericity tensor  $\mathbf{S}$  defined in equation (6.5) and equation (6.6) with the summations running over all jets,  $k$ .

$$\mathbf{S} = \begin{pmatrix} \sum_k p_x^2 & \sum_k p_y p_x \\ \sum_k p_x p_y & \sum_k p_y^2 \end{pmatrix} \quad (6.5)$$

$$\text{such that } S_{ij} = \sum_k p_{ki} p_{kj} \quad (6.6)$$

An area normalised shape comparison of the  $S_T$  variable for both QCD jet and SUSY events (Figure 6.8) shows the expected sharp increase for QCD jets as  $S_T \rightarrow 0$ . The nature of this increase comes from the dominance of the QCD 2-to-2 cross-section.



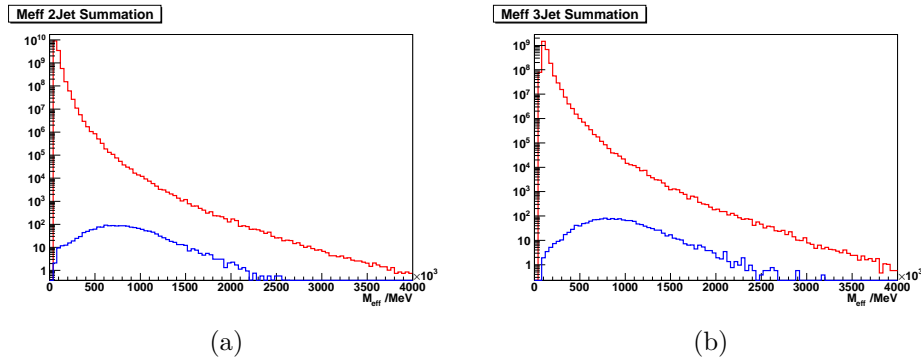
**Figure 6.8.:** Area normalised plot of  $S_T$  for QCD jets (red) and SUSY jets (blue).

#### 6.5.4. Effective Mass, $M_{eff}$

The last variable to introduce is the so-called ‘effective mass’  $M_{eff}$ . This is an event level variable and can be thought of as the activity of the event. It is the scalar sum of the  $E_T^{miss}$  and the  $n$  hardest jets and is defined as in equation (6.7).

$$M_{eff}^{njet} = E_T^{miss} + \sum_n p_T^{njet} \quad (6.7)$$

Where the ‘ $njet$ ’ in both  $M_{eff}^{njet}$  and  $p_T^{njet}$  merely label the variables indicating the extent of the summation and should not be confused with exponents. Figure 6.9 shows a comparison between the  $M_{eff}$  variable for the QCD di-jet and SUSY SU3 events using the hardest two and three jets in the summation respectively. The effect of the SUSY events having generally more  $E_T^{miss}$  can be seen in the skew of the mean above that of the QCD di-jet sample.



**Figure 6.9.:**  $M_{eff}$  plots for (a) two hardest and (b) three hardest jets in QCD (red) and SUSY (blue) events

## 6.6. Trigger

For the SUSY analyses, the calorimeter jet triggers are used. Additionally, combined jet and  $E_T^{miss}$  triggers have been studied but for early data running it is assumed that only the jet triggers will be used as the  $E_T^{miss}$  will not be fully understood straight away.

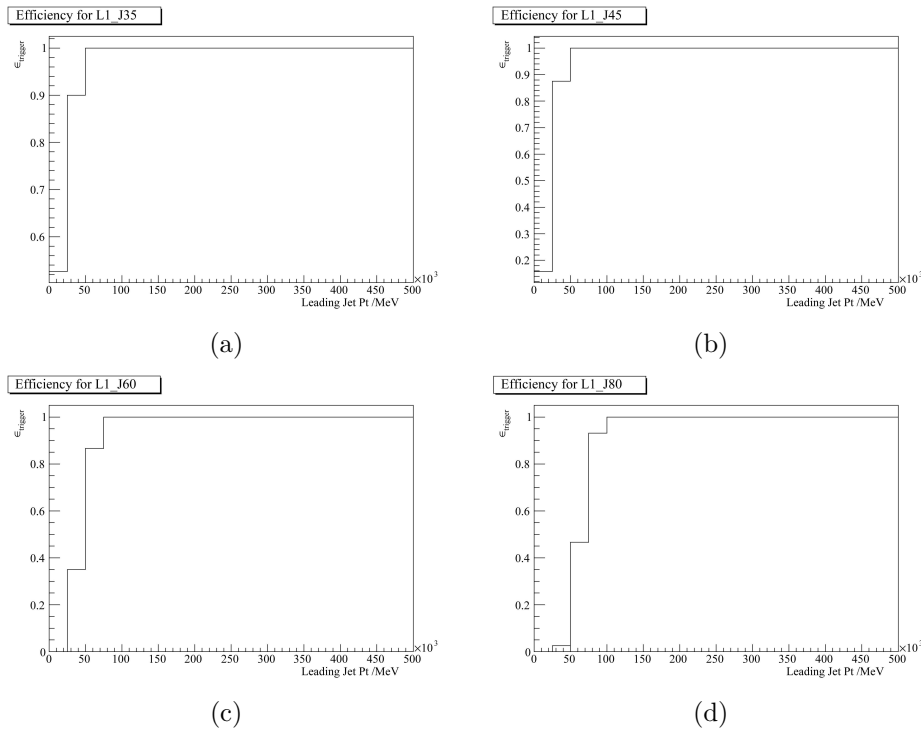
In early data, the higher level triggers will be in a *pass-through* mode whereby they simply accept anything passing the level 1 trigger.

It is assumed that the level 1 jet calorimeter triggers will have a high efficiency for these analyses but for simplicity they are not considered here and merely assumed to be 100% efficient. By way of a quick check, the leading jet's  $p_T$  for all events passing a variety of level one triggers was plotted divided by the total of all the leading jets'  $p_T$ , giving a trigger efficiency based on leading jet  $p_T$  as in equation (6.8).

$$\epsilon = \frac{\text{Leading Jet } p_T \ \&\& \ \text{TRIGGER}}{\text{Leading Jet } p_T} \quad (6.8)$$



These plots can be seen in Figure 6.10. Given the  $p_T$  cut to be imposed on the leading jet as outlined in Section 6.8 is 150 GeV it can be seen that each of these triggers is 100% efficient.



**Figure 6.10.:** Plots showing the trigger efficiency ratio of (leading jet  $p_T$  of events passing the trigger/jet  $p_T$ ) for level one triggers with threshold of (a) 35 GeV, (b) 45 GeV, (c) 60 GeV and (d) 80 GeV.

## 6.7. Pre-Selection and Overlap Removal

### 6.7.1. Pre-Selection

A series of pre-selection cuts were placed on the the reconstructed particles, the point of which was to ensure that only well defined objects (i.e. those whose kinematic variables pass the pre-selection cuts) populate the sample *before* any analysis takes place. These cuts perform various tasks such as

requiring a minimum  $p_T$  or restricting the  $\eta$  range in which objects are considered. These pre-selection cuts, rather than reducing the number of events, act as a filter reducing the number of reconstructed objects in each event to some lower number of well defined objects. The full list of pre-selection cuts used in this study (2/3 jet analysis independent) can be seen in Table 6.5.

(a)	(b)	(c)	(d)
<b>Jet</b>	<b>Electron</b>	<b>Photon</b>	<b>Muon</b>
$p_T > 20$ GeV	author & 1 = 1*	$E_T > 10$ GeV	isHighPt = 1
$ \eta  < 2.5$	$p_T > 20$ GeV	etcone < 10 GeV <sup>†</sup>	bestMatch = 1
	$ \eta  < 2.5$	Shower shape = 15	isCombinerMuon = 1
	isem & 0x3ff = 0*		$p_T > 20$ GeV
	etcone20 < 10 GeV		$ \eta  < 2.5$
			etcone20 < 10 GeV
			$0 < \text{fitChi2OverDoF} < 5$
			$0 < \text{matchChi2} < 20$

\* The & symbol here represents the bit-wise AND operation.

† In the software the variable etcone45 is known just as etcone.

**Table 6.5.:** Preselection cuts for (a) jets, (b) electrons, (c) photons and (d) muons

Jets were reconstructed using the infra-red unsafe ATLAS cone jet finding algorithm as it was in common use at the time. Owing to the high multiplicities involved in SUSY events the smaller standard cone size parameter was used of  $R_{cone} = 0.4$ . Jets found with this algorithm then have loose kinematic pre-selection cuts applied to them before being considered for the final analysis.

Electrons are required to be isolated, with less than 10 GeV being in the calorimeter within a radius of  $\Delta R = 0.2$ . In addition they must kinematically have a  $p_T > 20$  GeV and be within  $|\eta| < 2.5$ . The *author* of the electron relates to the type of algorithm that the electron was recon-

structed with. Here, with the requirement that `author = 1`, only electrons reconstructed with the standard cluster based algorithm are considered. The `isEM` flag is a bit pattern that contains the results of various calorimeter and tracking criteria. The `0x3FF` mask selects only those electrons for which the calorimeter only criteria are applied in the reconstruction.

Muons are required to be reconstructed by the matching of inner detector and muon spectrometer tracks, so-called *combined* muons. The isolation and kinematic requirements are otherwise as with the electrons. The requirement of ‘`isHighPt`’ selects muons reconstructed with the high  $p_T$  algorithm while ‘`bestMatch`’ is true for muons whose combined track is the best match for the muon spectrometer track\*. The *fit* and *match* Chi2 quantities relate to the quality of the track fitting and matching respectively.

For photon isolation, a cone of  $\Delta R = 0.45$  is required to have less than 10 GeV of calorimeter energy within it and the photon is required to have  $E_T > 10$  GeV. The shower shape variable encode the properties of the EM shower within the LAr calorimeter taking into account the shapes in the first and second sampling as well as the hadronic leakage. These photons are then kept and counted as jets in the main analysis.

### 6.7.2. Overlap Removal

In addition to achieving a well defined set of reconstructed objects in each event, it is also desirable to reduce the double counting that might occur. To this end we define a so called ‘overlap criterion’ which specifies that a pre-selected jet is considered overlapped if it is closer in the  $\eta$ - $\phi$  plane to

---

\*Owing to the high track multiplicity of the inner detector there could be several combined tracks possible for any given muon spectrometer track.

a pre-selected electron than 0.2, as in equation (6.9). If one is found then the electron is kept in preference. It is clear to see that this step must be performed post pre-selection and that it has the effect of reducing the number of fake jets coming from electrons,

$$\Delta R_{jet-electron} < 0.2 \quad (6.9)$$

where  $\Delta R$  is the distance measure in the  $\eta$ - $\phi$  plane defined between two objects 1 and 2 as in equation (4.2) (and shown again below).

$$\Delta R_{1-2} = \sqrt{(\eta_1 - \eta_2)^2 + (\phi_1 - \phi_2)^2}$$

## 6.8. Cuts

The cuts for the SUSY 2-jet and 3-jet analyses are based on the discriminating quantities introduced earlier.

The first series of cuts are placed on the  $p_T$  of the jets, with the requirement that there be at least 2 jets with  $p_T$  greater than 100 GeV in the 2-jet analysis and 3 jets greater than 100 GeV in the 3-jet analysis. In addition, the hardest jet is required to be harder in  $p_T$  than 150 GeV.

The second set of cuts relate to the  $E_T^{miss}$  quantity. The requirement is that the  $E_T^{miss}$  be at least 100 GeV and that in the 2-jet analysis it be greater than  $0.3M_{eff}^{2jet}$ , or  $0.25M_{eff}^{3jet}$  in the 3-jet analysis.

There then follows a series of cuts on  $\delta\phi$ , requiring an angular separation between the leading  $n$  jets and the  $E_T^{miss}$  greater than 0.2 radians for the  $n$ -jet analysis. In addition there are also cuts on the  $R_{1,2}$  quantities introduced in Section 6.5.2 and defined in equation (6.3). This has the effect of cutting out the activity in the highly populated corner regions of

the  $\delta\phi_1 - \delta\phi_2$  plot with a radius of 0.5 radians. These  $R_{1,2}$  cuts were under investigation but proved to be redundant in combination with the  $\delta\phi$  cuts and so were removed for later work.

The final cut is an isolation cut for leptons. Since this analysis concerns itself with the 0-lepton channel, the requirement of *no* isolated leptons is placed on the data. The definition of an isolated lepton for the purposes of this analysis is given in equation (6.10),

$$\Delta R_{lepton-jet} > 0.4 \quad \forall \text{ jets} \quad (6.10)$$

where here lepton refers to the charged leptons of the first two generations, namely the electron and the muon. The full cuts list for the two and three jet analyses are summarised in Table 6.6.

(a)		(b)	
Cut No.	Definition	Cut No.	Definition
Cut 1	$N_{jets} \geq 2$	Cut 1	$N_{jets} \geq 3$
Cut 2	$p_T^{jet1} > 150 \text{ GeV}$	Cut 2	$p_T^{jet1} > 150 \text{ GeV}$
Cut 3	$p_T^{jet2} > 100 \text{ GeV}$	Cut 3	$p_T^{jet2} > 100 \text{ GeV}$
Cut 4	$E_T^{miss} > 100 \text{ GeV}$	Cut 4	$p_T^{jet3} > 100 \text{ GeV}$
Cut 5	$E_T^{miss} > 0.3M_{eff}^{2jet}$	Cut 5	$E_T^{miss} > 100 \text{ GeV}$
Cut 6	$\delta\phi_1 > 0.2^c$	Cut 6	$E_T^{miss} > 0.25M_{eff}^{3jet}$
Cut 7	$\delta\phi_2 > 0.2^c$	Cut 7	$\delta\phi_1 > 0.2^c$
Cut 8*	$R_1 > 0.5^c$	Cut 8	$\delta\phi_2 > 0.2^c$
Cut 9*	$R_2 > 0.5^c$	Cut 9	$\delta\phi_3 > 0.2^c$
Cut 10	<b>NO</b> isolated lepton	Cut 10*	$R_1 > 0.5^c$
		Cut 11*	$R_2 > 0.5^c$
		Cut 12	<b>NO</b> isolated lepton

\*  $R_n$  cuts not standardised within the SUSY community and dropped for subsequent papers owing to similarity with  $\delta\phi_n$  cuts.

**Table 6.6.:** Event selection cuts for (a) two and (b) three jet analysis

## 6.9. Signal Sensitivity

It can be shown that for a counting experiment with known background  $\mathbf{B}$ , an estimate of the expected discovery significance (significance with which one would reject the background-only hypothesis) is equation (6.11) [49], where the expectation value,  $\mathbf{Z}$ , assumes a signal strength  $\mathbf{S}$ . This becomes the more familiar  $S/\sqrt{B}$  in the limit  $S \ll B$ .

$$Z = \sqrt{2((S + B) \ln(1 + S/B) - S)} \quad (6.11)$$

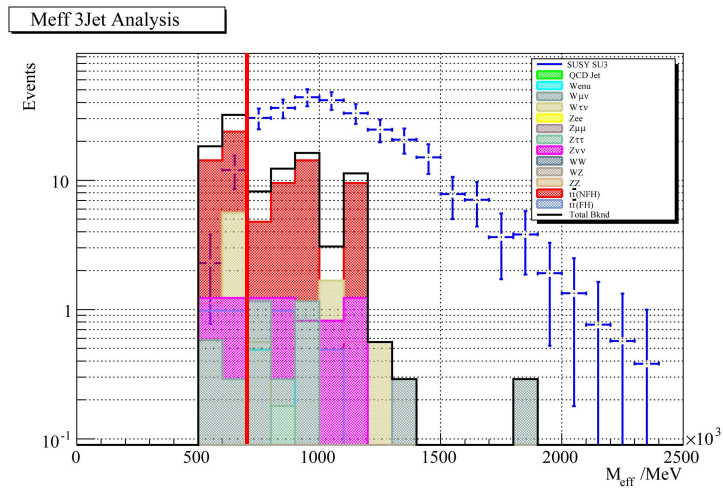
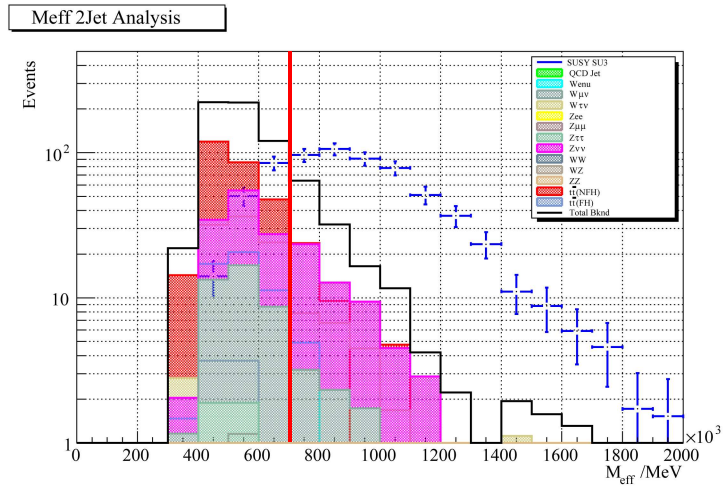
Both the  $M_{eff}$  and  $E_T^{miss}$  were plotted for the SUSY signal and the SM background samples normalised to  $100 \text{ pb}^{-1}$ . The total of the SM background was also plotted on the same axes. These plots can be seen in Figure 6.11 and Figure 6.12.

Following the analysis cuts, the signal can be seen to extend to both higher  $M_{eff}$  and  $E_T^{miss}$  with the lower values dominated by the SM background. A cut can therefore be placed on these values in order to maximise our signal. A sensible value for these cuts is chosen and can be seen in Table 6.7 and shown on Figure 6.11 and Figure 6.12 as a red line.

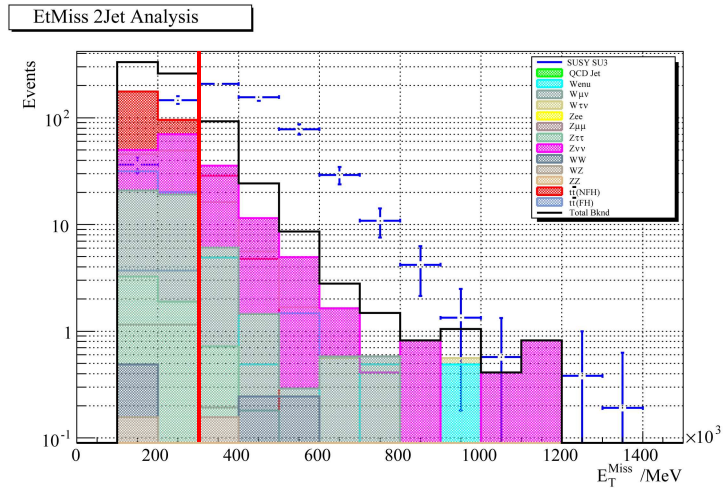
	Cuts
$M_{eff}$	700 GeV*
$E_T^{miss}$	300 GeV

\* These cuts are initial exploratory values. Current SUSY values for  $M_{eff}$  are 500 GeV for 2-jet analysis and both 500 GeV and 1 TeV for the 3-jet analysis.

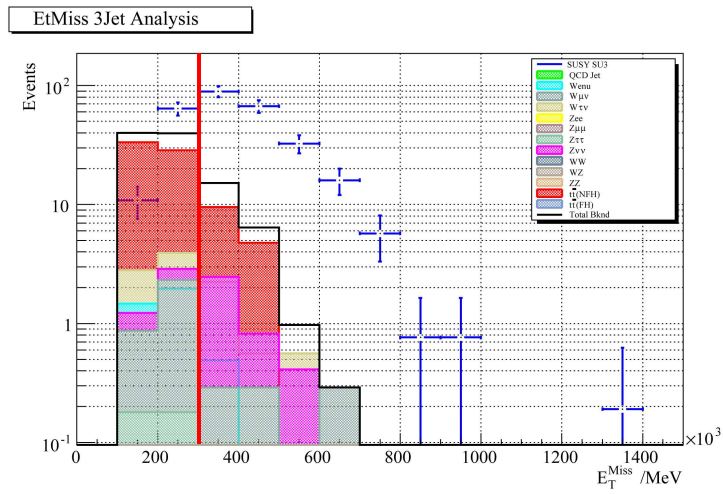
**Table 6.7.:** Table showing the cuts used to separate the signal from the background.



**Figure 6.11.:**  $M_{eff}$  plots showing SUSY signal and combined SM background for (a) the 2-jet analysis and (b) the 3-jet analysis. The red line shows the signal separation cut.



(a)



(b)

**Figure 6.12.:**  $E_T^{\text{miss}}$  plots showing SUSY signal and combined SM background for (a) the 2-jet analysis and (b) the 3-jet analysis. The red line shows the signal separation cut.



In order to work out the signal significance  $Z$ , both the SUSY and the combined SM plots were integrated to get an event count from the first bin above this cut value up to last bin. The event count from the SUSY plot was then used as the signal,  $S$  value and that from the combined SM plot was the background,  $B$ . The expected discovery significances were then computed using equation (6.11), the results of which can be seen in Table 6.8.

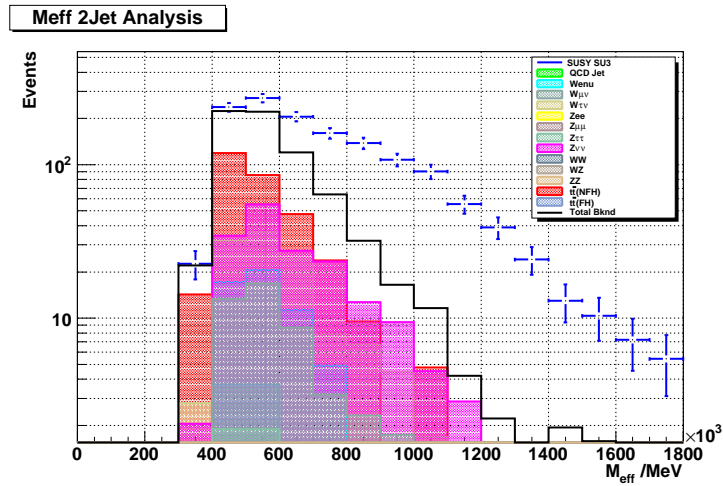
Discovery Significance, $Z$	
<b>2 Jet Analysis</b>	
$M_{eff}$	38
$E_T^{miss}$	37
<b>3 Jet Analysis</b>	
$M_{eff}$	29
$E_T^{miss}$	28

**Table 6.8.:** Table of significances from the plots in Figure 6.11 and Figure 6.12.

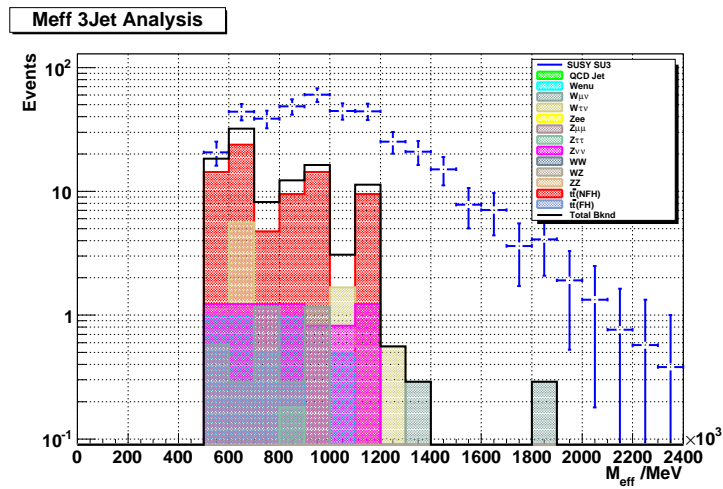
## 6.10. Summary and Conclusion

When looking at the significance values in Table 6.8 it is important to note the caveat that these values are higher than expected in data since no systematics were considered in the background determination. Nevertheless, the validity of studying this channel as an early data SUSY discovery search can clearly be seen. Figure 6.13 and Figure 6.14 show the final  $M_{eff}$  and  $E_T^{miss}$  plots, for both the 2-jet and 3-jet analyses, with the SUSY SU3 signal stacked on top of the SM background as would be expected in data.

Owing to the author's supervisor leaving, this was as far as the author pursued this analysis. The analysis itself has continued and improved, however the basics of the analysis are as presented in this thesis albeit with

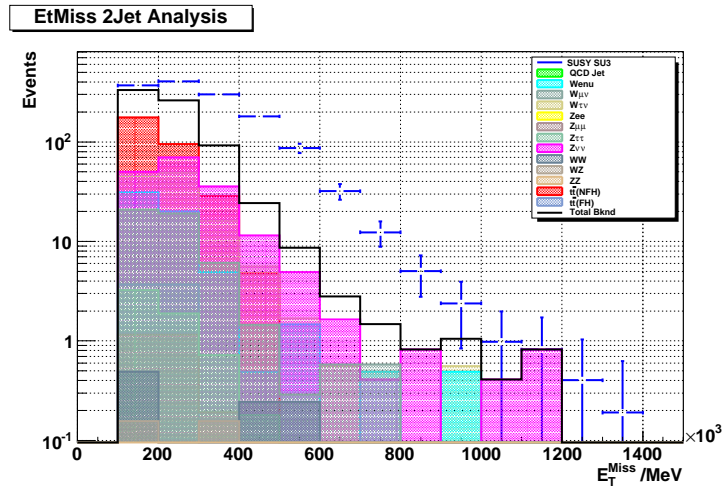


(a)

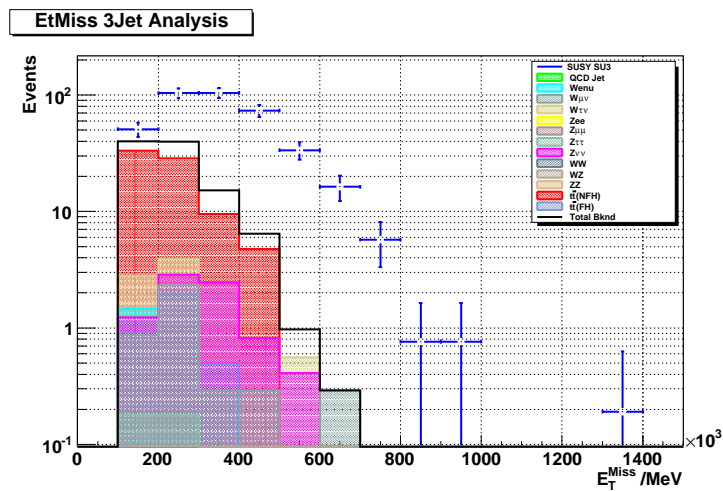


(b)

**Figure 6.13.:**  $M_{eff}$  plots showing SUSY signal stacked on top of the combined SM background for the (a) 2-jet and (b) 3-jet analysis. The individual components of the SM background are also superimposed for completeness.



(a)



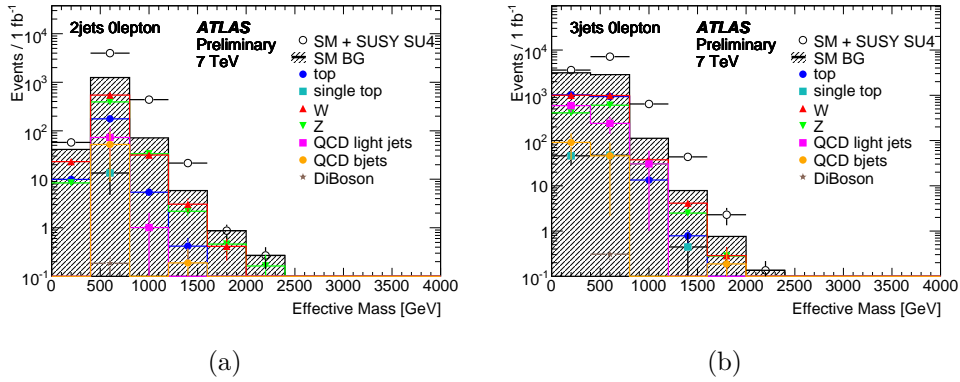
(b)

**Figure 6.14.:**  $E_T^{\text{miss}}$  plots showing SUSY signal stacked on top of the combined SM background for the (a) 2-jet and (b) 3-jet analysis. The individual components of the SM background are also superimposed for completeness.

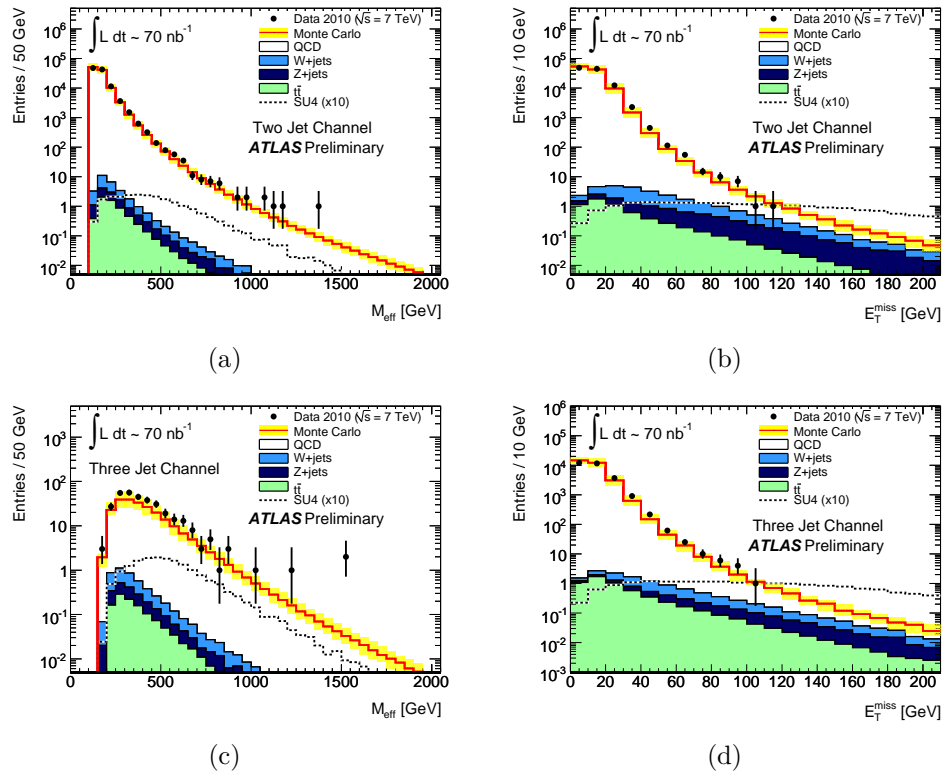
more optimised cuts etc. and the use of a couple of additional discriminatory variables. The latest published simulation plots [45] can be seen in Figure 6.15. These plots show (normalised for  $1 \text{ fb}^{-1}$ ) the sensitivity of the SUSY  $M_{eff}$  variable, in both the 2-jet and 3-jet channels, this time to the SU4 mSUGRA benchmark point. In both cases, the excess of events from the SU4 SUSY sample can be seen clearly above the SM background.

Since ATLAS has started taking data, attempts have been made to look again at the SUSY variables in light of the data [50]. With the small amount of collected data used in this paper a good agreement between the simulated SM background and data is observed up to values of  $E_T^{miss} \sim 100 \text{ GeV}$  and  $M_{eff} \sim 1500 \text{ GeV}$  showing a good understanding of both detector and physics simulation and that the real ATLAS detector is performing as expected. Figure 6.16 shows the plots from this paper for the two and three jet analyses with only the jet cuts applied.

Finally, in the latest paper [51] submitted to Physics Letters B and utilising  $\sim 35 \text{ pb}^{-1}$  of data at  $\sqrt{s} = 7 \text{ TeV}$ , the current world's best limits have been set using the 0-lepton channel. They find that in simplified models containing only squarks of the first two generations, a gluino octet and a massless neutralino, that with 95% confidence gluinos are excluded from having masses below 500 GeV. For equal mass squarks and gluinos this limit increases to 870 GeV. In mSUGRA/CMSSM models with  $\tan \beta = 3$ ,  $A_0 = 0$  and  $\mu > 0$ , equal massed gluinos and squarks are excluded below 775 GeV [51].



**Figure 6.15.:**  $M_{eff}$  plots from the latest Monte Carlo simulation paper [45], showing SUSY signal and combined SM background for both the (a) 2-jet and (b) 3-jet analysis.



**Figure 6.16.:** SUSY signal and combined SM background plots for real ATLAS data [50] showing (a)  $M_{eff}$  plot with only the 2-jet analysis jet cuts, (b)  $E_T^{miss}$  plot with only the 2-jet analysis jet cuts, (c)  $M_{eff}$  plot with only the 3-jet analysis jet cuts and (d)  $E_T^{miss}$  plot with only the 3-jet analysis jet cuts.

# Chapter 7.

## Minimum Bias Analysis

This chapter will focus on summarising the first ATLAS minimum bias analysis paper [52] in which the author participated and it will highlight the author's contributions. There is a brief introduction to the Rivet toolkit on which the author did some analysis coding work followed by Monte Carlo simulated analysis work related to the minimum bias analysis presented. Finally it will show the development of the analysis since the author ceased active participation to write up this thesis.

### 7.1. Analysis Introduction

The aim of the analysis was to measure the following quantities for charged particles with  $p_T > 500$  MeV,  $|\eta| < 2.5$  and with mean lifetime  $\tau > 0.3 \times 10^{-10}$  s. With this lifetime,  $K_S^0$ ,  $\Lambda$ ,  $\Sigma^-$ ,  $\Sigma^+$ ,  $\Xi^-$ ,  $\Xi^0$  and  $\Omega^-$  are taken to be stable [5]. Comparisons are then made to current Monte Carlo simulation models as well as previous measurements at  $\sqrt{s} = 900$  GeV.

1.  $\frac{1}{N_{ev}} \cdot \frac{dN_{ch}}{d\eta}$  The pseudorapidity distribution of charged primary particles.

2.  $\frac{1}{N_{ev}} \cdot \frac{1}{2\pi p_T} \cdot \frac{d^2 N_{ch}}{d\eta d p_T}$  The  $p_T$  spectra of primary charged particles.
3.  $\frac{1}{N_{ev}} \cdot \frac{d N_{ev}}{d n_{ch}}$  The primary charged particle multiplicity.
4.  $\langle p_T \rangle$  vs.  $n_{ch}$  The mean primary charged particle  $p_T$  against multiplicity.

$N_{ev}$  is the number of events passing the selection criteria and  $N_{ch}$  is the number of charged particles in all events passing selection and  $n_{ch}$  is the number of charged particles per event.

Unlike previous analyses, the aim of this analysis was to be as model independent as possible. Whereas in the past certain sections of diffractive events were removed\* either by use of a double-arm coincidence trigger or model dependent corrections, the paper described here requires only one charged particle within the acceptance and therefore presents an inclusive inelastic measurement with minimal model dependence.

## 7.2. Monte Carlo Simulation

For the simulated events, individual samples some  $10^7$  large of single, double and non-diffractive events were produced by the Pythia 6.4.21 generator and mixed to form the inclusive inelastic sample, weighted according to the relative generator cross-sections. For comparison with the alternate ‘dual parton model’ [53,54] the Phojet [55] generator version 1.12 was also used. Unlike Pythia which has at least one hard interaction per event, particle production in Phojet is predominantly soft [56]. A pomeron exchange in the dual parton model gives rise to two strings stretched between valence/sea partons. Low energy collisions are characterised by single pomeron ex-

---

\*Some remove the single diffractive component leaving what is labelled non-single diffractive (NSD). Others apply corrections to remove the double diffractive as well, leaving only the inelastic non-diffractive component.

change, with more pomerons and hence more strings added for the higher energy collisions.

The reference tune of the Pythia generator used throughout the paper was the ATLAS MC09 [57] tune which employed the ‘MRST LO\*’ parton density functions [58]. This tune’s parameters were derived from tuning to underlying event and minimum bias data from the Tevatron at 630 GeV and 1.8 TeV using the Rivet Toolkit [59] described in Section 7.9.

In addition to the reference tune, for comparison purposes three further Pythia tunes were used. The first of these was the ATLAS MC09c tune, an extension of the reference tune optimising the strength of the colour reconnection to describe the  $\langle p_T \rangle$  as a function of  $n_{ch}$  measured by CDF [60]. Secondly the Perugia0 [61] tune which has had the soft-QCD tuned using minimum bias data only from the Tevatron and CERN. Finally there was the DW tune using virtually ordered showers to describe Run II underlying event and Drell-Yan data from CDF [62].

### 7.3. Event Selection

The event selection in this analysis utilises only the MBTS trigger and inner detector information. Events are selected as ones having the following criteria.

1. A *single* reconstructed primary vertex using at least three tracks satisfying:
  - $p_T > 150$  MeV
  - $|d_0^{BS}| < 4$  mm
2. At least one track with:



- $p_T > 500$  MeV
- $\geq 1$  pixel hit and  $\geq 6$  SCT hits
- $|d_0| < 1.5$  mm and  $|z_0| < 1.5$  mm

Where  $|d_0^{BS}|$  is the transverse distance of closest approach (impact parameter) with respect to the beam spot position,  $|d_0|$  and  $|z_0|$  being the transverse and longitudinal impact parameters with respect to the primary vertex respectively.

Tracks used to produce the plots in the analysis were those selected as in 2 above and are labelled as selected tracks, the multiplicity of which were denoted by  $n_{Sel}$ .

## 7.4. Background Contribution

Background contributions come from either cosmic-rays or so-called *beam induced* sources. The background from cosmic-ray events selected with the minimum bias trigger, described in the Section 7.5.1, was given a limit from data studies and was found to be smaller than  $10^{-6}$  (i.e. 1 in  $10^6$  events).

Beam induced backgrounds come from either proton collisions with a collimator upstream of the detector or residual gas particles inside the beam pipe. These backgrounds were studied using unpaired bunch crossings and were found to be less than  $10^{-4}$  events of those selected.

Both the potential backgrounds from fakes and secondaries were measured from the Monte Carlo Simulation and found to be less than 0.1% and  $2.20 \pm 0.05(\text{stat.}) \pm 0.11(\text{syst.})$  respectively.

## 7.5. Selection Efficiencies

Both the trigger and vertex reconstruction efficiencies were parameterised not as a function of  $n_{Sel}$ , but rather  $n_{Sel}^{BS}$ . The difference between them is that for  $n_{Sel}^{BS}$  the constraints on the longitudinal and transverse impact parameters in relation to the primary vertex are replaced with just one constraint on the transverse impact parameter with respect to the beam spot as  $|d_0^{BS}| < 4$  mm identical to the primary vertex reconstruction condition. This was done in order to avoid any possible correlation between the two efficiencies.

### 7.5.1. Trigger Efficiency

The MBTS described in Section 2.8 was used as the triggering device for this analysis. A single-arm trigger was used, configured to require one hit only above threshold on either side of the detector so as to avoid introducing any unnecessary model dependence. This trigger was denoted L1\_MBTS\_1.

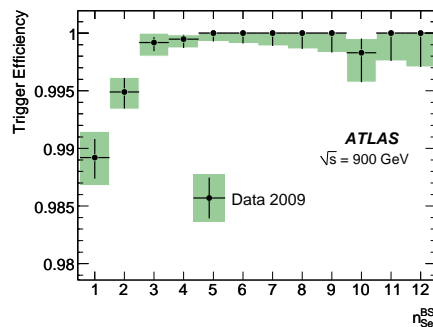
To measure the efficiency of the L1\_MBTS\_1 trigger, a second pre-scaled L1 trigger was used, utilising the signals from the beam pick-up timing devices (**BPTX**) used to allow ATLAS to ‘see’ bunches on their way into the experiment. The BPTX are attached to the beam pipe at  $z = \pm 175$  m. While the L1\_BPTX trigger is a zero-bias trigger and thus a desirable comparison, owing to the level of the pre-scaling its use was not feasible as the number of collected events whilst using it was too low. Instead a new trigger was formed using the L1\_BPTX signal at L1, filtered to obtain inelastic interactions by requiring more than 6 pixel clusters and 6 SCT

hits at L2 and one or more reconstructed tracks<sup>†</sup> at the EF level with  $p_T > 200$  MeV. Together the BPTX, L2 and EF filters formed a control trigger.

Using an independent data sample selected with the control trigger, the efficiency was calculated as the ratio of events selected by the control trigger as well as the L1\_MBTS\_1 trigger, over the total of those selected by the control trigger as shown in equation (7.1).

$$\epsilon_{trig}(n_{Sel}^{BS}) = \frac{N_{ev}^{CONTROL \ \&\& \ L1\_MBTS\_1}(n_{Sel}^{BS})}{N_{ev}^{CONTROL}(n_{Sel}^{BS})} \quad (7.1)$$

The Trigger efficiency as a function of  $n_{Sel}^{BS}$  was found to be  $\sim 100\%$  everywhere, only marginally lower for the first two bins, i.e.  $n_{Sel}^{BS} \leq 2$ . The trigger efficiency plot can be seen in Figure 7.1.



**Figure 7.1.:** Plot showing the L1\_MBTS\_1 trigger efficiency as a function of  $n_{Sel}^{BS}$  [52].

The section of trigger efficiencies was where the author contributed and is also named as an author on the support paper [63]. A digression is now made to introduce some of the background work carried out by the author.

<sup>†</sup>Using  $n_{Sel}^{BS}$  as described in Section 7.5, i.e. the vertex requirement for the track reconstruction was neglected, instead utilising the impact parameter from the beam spot.

### MBTS Trigger Simulation Efficiencies

In addition to the L1\_MBTS\_1 trigger used in the minimum bias analysis and already mentioned, several other MBTS triggers were initially considered, most notably the L1\_MBTS\_2 and L1\_MBTS\_1\_1 first introduced in Section 4.5 and outlined again below in Table 7.1.

Name	Requirement
L1_MBTS_1	One MBTS hit above threshold in either side
L1_MBTS_2	Two MBTS hits above threshold in either side
L1_MBTS_1_1	One MBTS hit above threshold in each side

**Table 7.1.:** Requirements for the three main MBTS related triggers.

The main focus of the author's contribution was to investigate using the Monte Carlo simulation the response of the MBTS trigger simulations to the different diffractive sub-samples. All three of the triggers listed in Table 7.1 were investigated using the Pythia6 Monte Carlo simulation. Trigger efficiency plots were made as a function of  $\eta$ ,  $p_T$  and number of charged particles  $N_{ch}$  with the efficiency per bin calculated as in equation (7.2),

$$\epsilon_{trig} = \frac{\text{No. passing L1\_MBTS\_X}}{\text{No. passing NoTrigReq}} \quad (7.2)$$

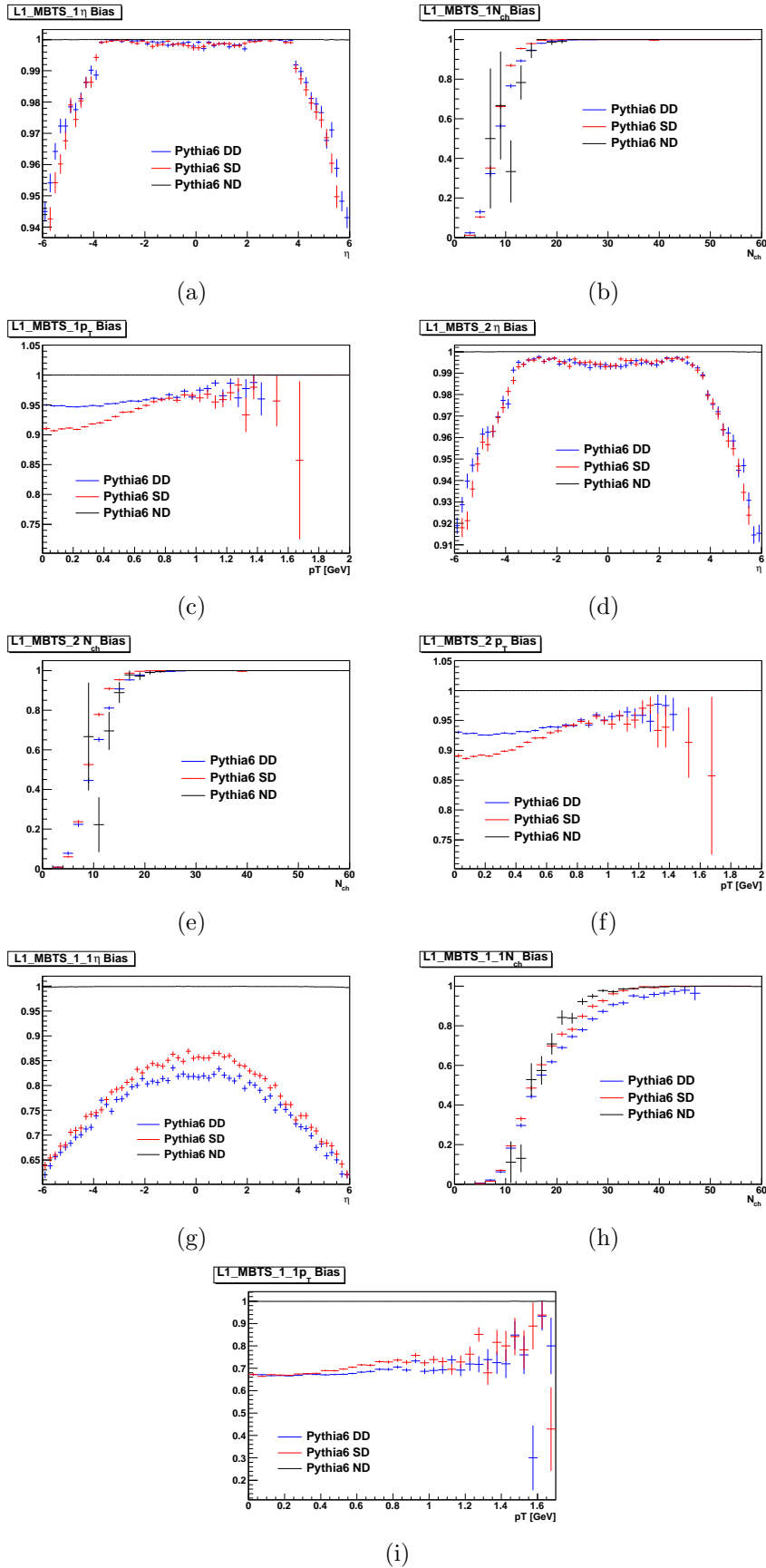
where 'NoTrigReq' is, as first introduced in Section 4.5, essentially a pass-through trigger accepting all events. The different diffractive samples (double, single and non-diffractive) are all plotted on the same axes for comparison.

From Figure 7.2 it can be seen that the efficiency for non-diffractive events is  $\sim 100\%$  across the board for  $N_{ch} > 20$  or  $N_{ch} \gtrsim 35$  for the coincidence trigger. It can also be seen that the performance of the triggers

is comparable between the different diffractive samples as a function of  $N_{ch}$ . Noticeable differences however are apparant when viewed as a function of either  $\eta$  or  $p_T$ .

Trigger efficiency as a function of  $\eta$  is approximately flat for the non-diffractive sample, but tails off for the diffractive samples above  $|\eta| \gtrsim 3$  for the non-coincidence triggers. The coincidence trigger shows no real sign of flattening for central pseudorapidities. In all cases the peak efficiency of the diffractive samples lowers going down the trigger table in Table 7.1.

As a function of  $p_T$  the trigger efficiency for the non-diffractive sample is flat. For the diffractive samples there can be seen to be a gradual increasing trend in the efficiency with  $p_T$ . It can also be seen that the efficiency for the single diffractive sample increases faster with  $p_T$  than it does for the double diffractive. As above the trigger efficiency is generally lower for triggers further down Table 7.1.

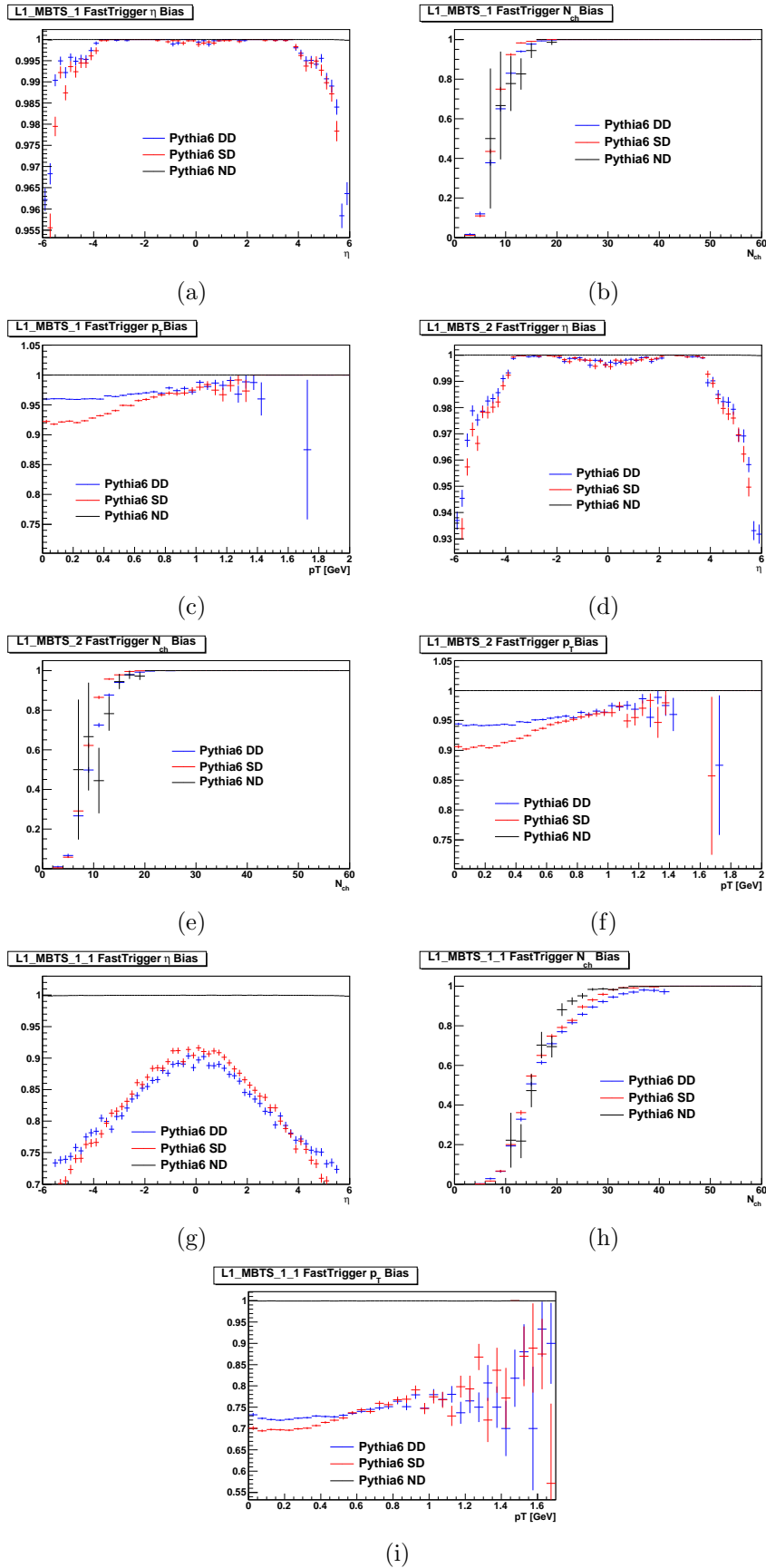


**Figure 7.2.:** Trigger efficiency plots showing DD (blue), SD (red) and ND (black) samples on the same axes for comparison. The plots show the efficiency for the (a,b,c) L1\_MBTS\_1, (d,e,f) L1\_MBTS\_2 and (g,h,i) L1\_MBTS\_1.1 as a function of (a,d,g)  $\eta$ , (b,e,h)  $N_{ch}$  and (c,f,i)  $p_T$ .

In Section 4.5 it was shown that the fast trigger simulation derived by the author within the Atlfast framework could reproduce the trigger response of these triggers, albeit slightly overestimating the response. The fast trigger itself had no real dependencies on the Atlfast framework and was merely created with it in mind since the author is an Atlfast developer. In fact the fast trigger simulation was designed from the ground up to be general enough to be of use standalone by any code that required it.

It was therefore of interest to the author to attempt to reproduce the plots in Figure 7.2 using the fast trigger simulation. It was a very simple task to do this and the only required modification to the code was to include the fast trigger headers and then switch the official trigger decision line for the fast trigger decision. Having done this the bias plots were re-generated.

As can be seen by comparing Figure 7.3 with Figure 7.2 the fast trigger simulation generally agrees well with the shape of the official trigger simulation. This is impressive given the low statistics used in the parameterisation, which was the same one used while testing the trigger development. The most notable discrepancies in shape are for the coincidence trigger and in the  $\eta$  and  $p_T$  plots. There is, as already mentioned, a tendency for the fast trigger to overestimate the efficiency, however this effect appears less so in the  $N_{ch}$  plots. This overestimation appears to have affected the double diffractive sample more than the single diffractive sample most notably for the coincidence trigger.



**Figure 7.3.:** Trigger efficiency plots showing DD (blue), SD (red) and ND (black) samples on the same axes for comparison. The plots show the efficiency for the fast simulation of (a,b,c) L1\_MBTS\_1, (d,e,f) L1\_MBTS\_2 and (g,h,i) L1\_MBTS\_1.1 as a function of (a,d,g)  $\eta$ , (b,e,h)  $N_{ch}$  and (c,f,i)  $p_T$ .

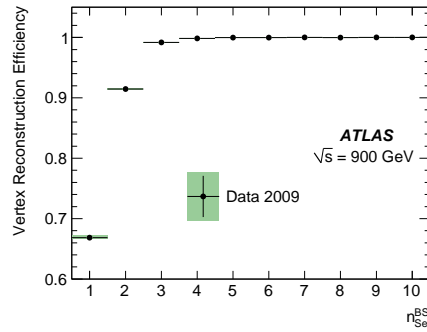


### 7.5.2. Vertex Reconstruction Efficiency

The vertex reconstruction efficiency was determined by taking the ratio of triggered events with a reconstructed vertex to all triggered events. This was done from the data and to avoid any correlations was again done as a function of  $n_{Sel}^{BS}$ . The efficiency was defined as in equation (7.3)

$$\epsilon_{vertexrec}(n_{Sel}^{BS}) = \frac{N_{ev}^{triggered \&\& \text{ vertex}}(n_{Sel}^{BS})}{N_{ev}^{triggered}(n_{Sel}^{BS})} \quad (7.3)$$

The vertex reconstruction efficiency can be seen in Figure 7.4.



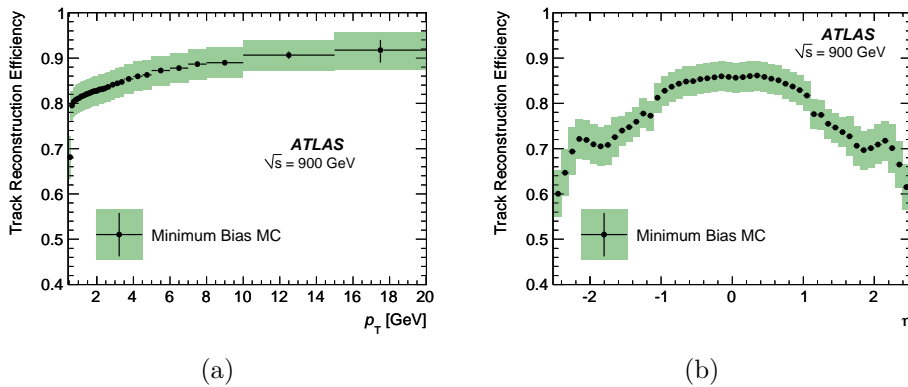
**Figure 7.4.:** Vertex reconstruction efficiency as a function of  $n_{Sel}^{BS}$ . [52]

### 7.5.3. Track Reconstruction Efficiency

The track reconstruction efficiency was calculated from the Monte Carlo simulated events and as a function of both  $p_T$  and  $\eta$ . It was taken as the ratio of the number of reconstructed tracks matched to a generated charged particle in a given bin over the total number of charged particles in that same bin as seen in equation (7.4)

$$\epsilon_{trkrec}(p_T, \eta) = \frac{N_{rec}^{matched}(p_T, \eta)}{N_{gen}(p_T, \eta)} \quad (7.4)$$

The matching of reconstructed tracks to generated charged particles was done using a cone based algorithm in the  $\eta$ - $\phi$  plane matching the track with the smallest  $\Delta R$  to generated charged particles within a cone of radius 0.05. Figure 7.5 show the track reconstruction efficiency as a function of  $p_T$  and  $\eta$ . The reduction in track reconstruction efficiency for  $|\eta| > 1$  in Figure 7.5(b) is due to the presence of more material in this region.



**Figure 7.5.:** Plots showing the track reconstruction efficiency as a function of (a)  $p_T$  and (b)  $\eta$ .

## 7.6. Correction Procedure

The charged particle spectra are obtained by applying corrections to the data to account for inefficiencies from the trigger selection, vertex and track reconstruction as well as backgrounds from secondary tracks.

Events lost due to trigger selection and vertex reconstruction inefficiencies are corrected for using an event-by-event weight as defined in equation (7.5) below.

$$w_{ev}(n_{Sel}^{BS}) = \frac{1}{\epsilon_{trig}(n_{Sel}^{BS})} \cdot \frac{1}{\epsilon_{vtxrec}(n_{Sel}^{BS})} \quad (7.5)$$

$p_T$  and  $\eta$  distributions were corrected on a track-by-track basis for the effects of the track reconstruction inefficiencies using the weight in equation (7.6).

$$w_{trk}(p_T, \eta) = \frac{1}{\epsilon_{trkrec}(p_T, \eta)} \cdot (1 - f_{sec}(p_T)) \cdot (1 - f_{okr}(p_T, \eta)) \quad (7.6)$$

where  $f_{sec}(p_T)$  and  $f_{okr}(p_T, \eta)$  are both estimated from the Monte Carlo simulation and correspond to the fractions of secondary tracks, and selected tracks for which the charged particle are outside the kinematic range respectively.

For the plots versus  $n_{ch}$ , a track level Bayesian unfolding [64] was used to correct back from the number of selected tracks to the corresponding number of charged particles.

## 7.7. Systematic Uncertainties

Numerous studies were carried out to determine the sources of systematic uncertainty, details of which can be seen in the paper [52].

Essentially the normalisation factor of ‘number of events’,  $N_{ev}$  and the number of charged particles were treated separately, with each having their different contributing sources added in quadrature to get their respective totals. The totals were then combined using equation (7.7) [65] below to account for their anti-correlation.

$$u_{tot} = |u_a - u_b| \quad (7.7)$$

Once combined the uncertainty was propagated to the final distributions. Table 7.2 summarises the systematic uncertainty contributions on the charged particle density at  $\eta = 0$ .

Systematic uncertainty on the number of Events, $N_{ev}$	
Trigger efficiency	< 0.1%
Vertex reconstruction efficiency	< 0.1%
Track reconstruction efficiency	1.1%
Different MC tunes	0.4%
Total uncertainty on $N_{ev}$	1.2%
Systematic uncertainty on $(1/N_{ev}) \cdot (d N_{ch}/d \eta)$ at $\eta = 0$	
Track reconstruction efficiency	4.0%
Trigger and Vertex efficiency	< 0.1%
Secondary fraction	0.1%
Total uncertainty on $N_{ev}$	-1.2%
Total uncertainty on $(1/N_{ev}) \cdot (d N_{ch}/d \eta)$ at $\eta = 0$	2.8%

**Table 7.2.:** Systematic uncertainties on the number of events,  $N_{ev}$  and on the charged particle density  $(1/N_{ev}) \cdot (d N_{ch}/d \eta)$  at  $\eta = 0$  which are anti-correlated. All other sources are assumed to be uncorrelated.

## 7.8. Results

The final corrected charged particle distributions can be seen in Figure 7.6 where they are compared to the aforementioned different generator tunes/models.

Figure 7.6(a) shows the multiplicity distribution as a function of  $\eta$  and is approximately flat in the region  $|\eta| < 1.5$  with an average quoted value of  $1.333 \pm 0.003$  (stat.)  $\pm 0.040$  (syst.) over the range  $|\eta| < 0.2$ . With the exception of the Pythia DW tune the others seem to agree *in shape* with the data albeit significantly lower.

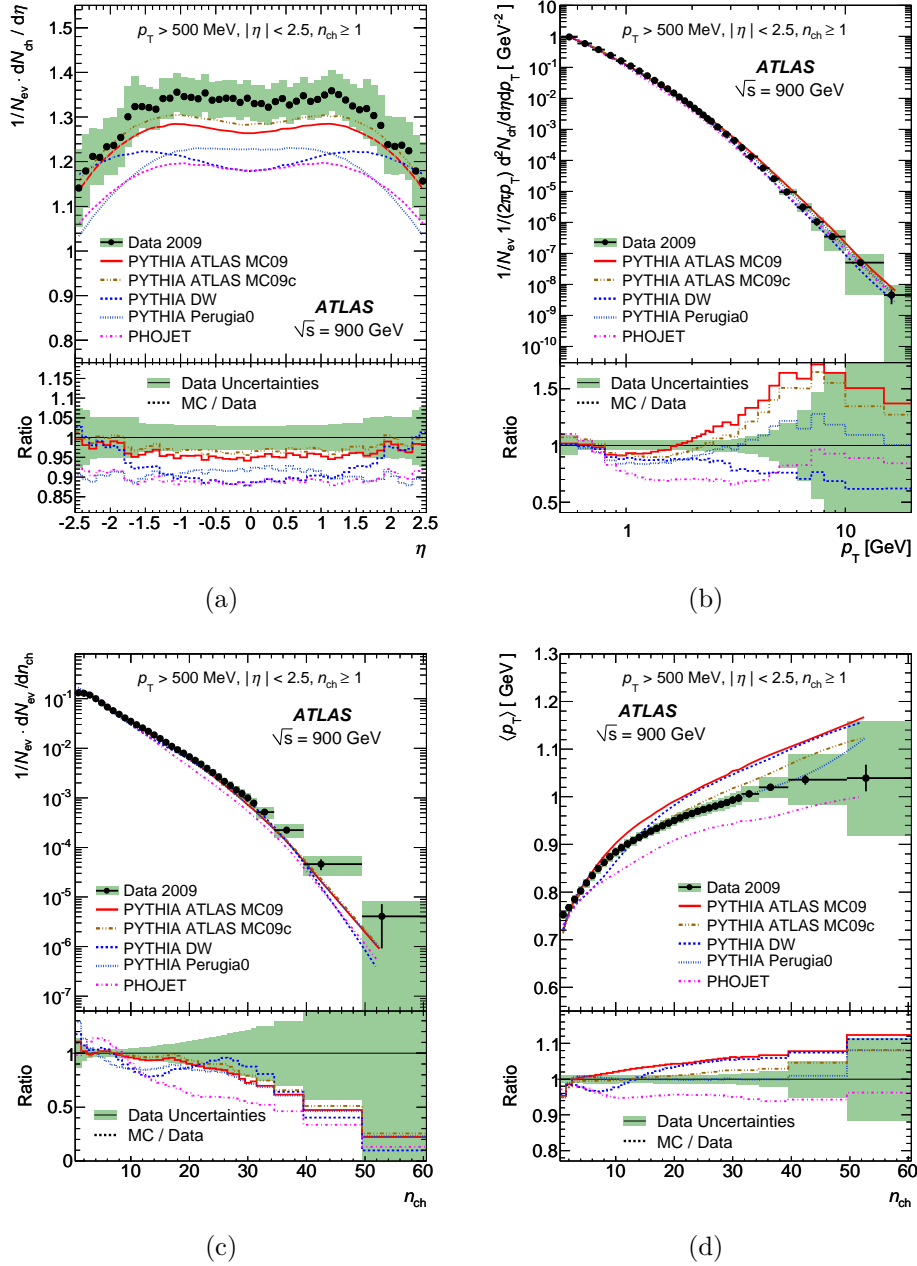
It can be seen that the current models do a poor job of describing the  $p_T$  distribution of charged particles seen in Figure 7.6(b) above  $p_T > 0.7$  GeV. The multiplicity distribution in Figure 7.6(c) clearly shows that above  $n_{ch} \gtrsim 10$  all Pythia models consistently undershoot the data. Only Phojet reproduces the  $n_{ch} = 1$  bin but its departure from the data at higher values is even more dramatic.

Figure 7.6(d) shows the average  $p_T$  as a function of  $n_{ch}$ . It shows a change of slope at  $n_{ch} = 10$ . The Perugia0 parameterisation best describes the data for this distribution, presumably since this parameterisation was tuned to the CDF minimum bias data at 1.96 TeV.

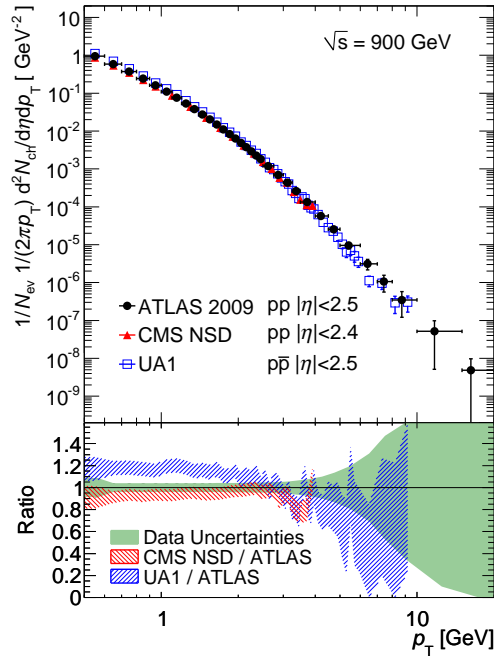
In addition to the Monte Carlo simulation models, the ATLAS data was also compared to previous measurements. Specifically the  $p_T$  distribution was compared to both the CMS [66] and UA1 [67] results.

Figure 7.7 shows that the CMS results are consistently lower than the ATLAS data. This is expected from the CMS definition of NSD events. Conversely the UA1 results are higher than the ATLAS data arising from the use of a double-arm trigger biasing the data against the lower charged particle multiplicity events.

At this point it is useful to take a brief digression to introduce the Rivet toolkit and the work that the author carried out relating to both it and the analysis presented up until this point.



**Figure 7.6.:** Comparison between data and MC models of charged particle multiplicities as a function of (a) pseudorapidity and (b)  $p_T$ . Also showing (c) the charged particle multiplicity per event and (d) the average  $p_T$  as a function of the number of charged particles in an event. Each of these plots are for events with  $n_{ch} \geq 1$  and charged particles within the kinematic range  $p_T > 500$  MeV and  $|\eta| < 2.5$ .



**Figure 7.7.:** Comparison of the charged particle multiplicity's  $p_T$  spectrum between the ATLAS , CMS and UA1 data.

## 7.9. Rivet Toolkit

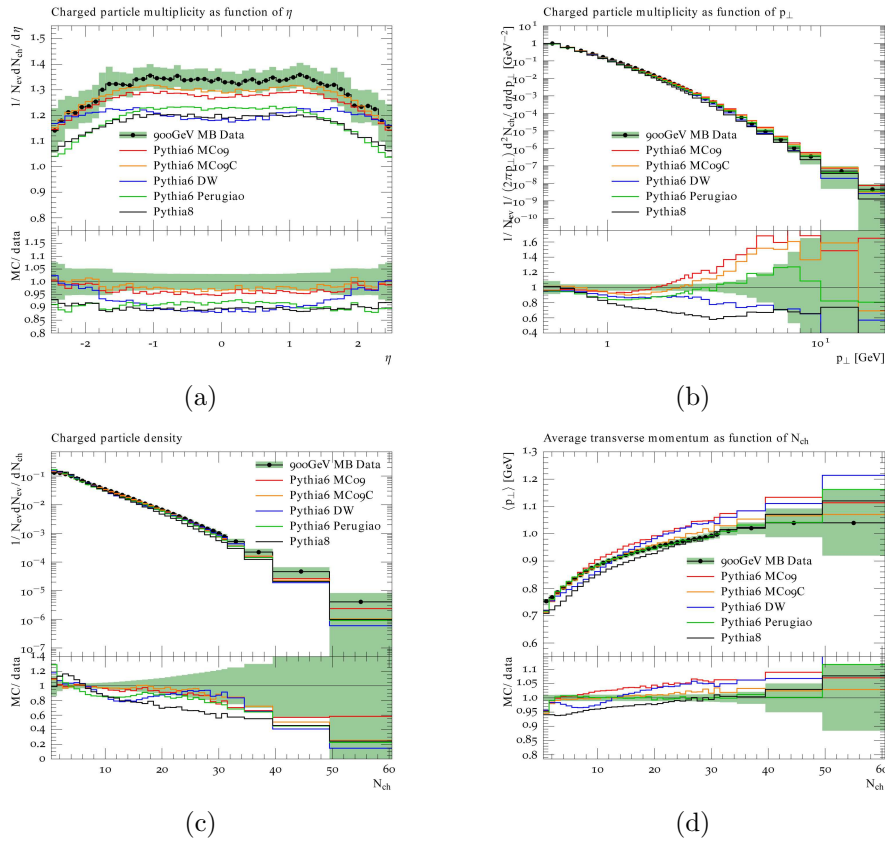
Rivet [59] (Robust Independent Validation of Experiment and Theory) is a generator validation toolkit. It provides a set of validated data analyses, along with their corresponding reference data, that can be run on the output ‘HepMC’ data file from a generator and compared to published results. It utilises the fact that many different generators can write events to the HepMC data format to provide a truly cross-generator validation platform.

Since it is relatively easy to use Rivet to run over a variety of different data, it can also be of assistance to the tuning community as well as a straightforward validation tool. In just such an application it is used as part of the automated tuning package Professor<sup>‡</sup> [68,69].

<sup>‡</sup>Professor is mentioned here for completeness but will not be discussed further.

## 7.10. Rivet Minimum Bias Analysis

In addition to being involved in the ATLAS minimum bias analysis, the author wrote, in parallel with another, a Rivet analysis routine which can be used to compare generator output from any generator with the data from this analysis. The source code for this Rivet analysis routine can be seen in Appendix B. Figure 7.8 shows plots made using this Rivet analysis and some of the first to compare the newer diffractive model of Pythia8 [4] with the data.

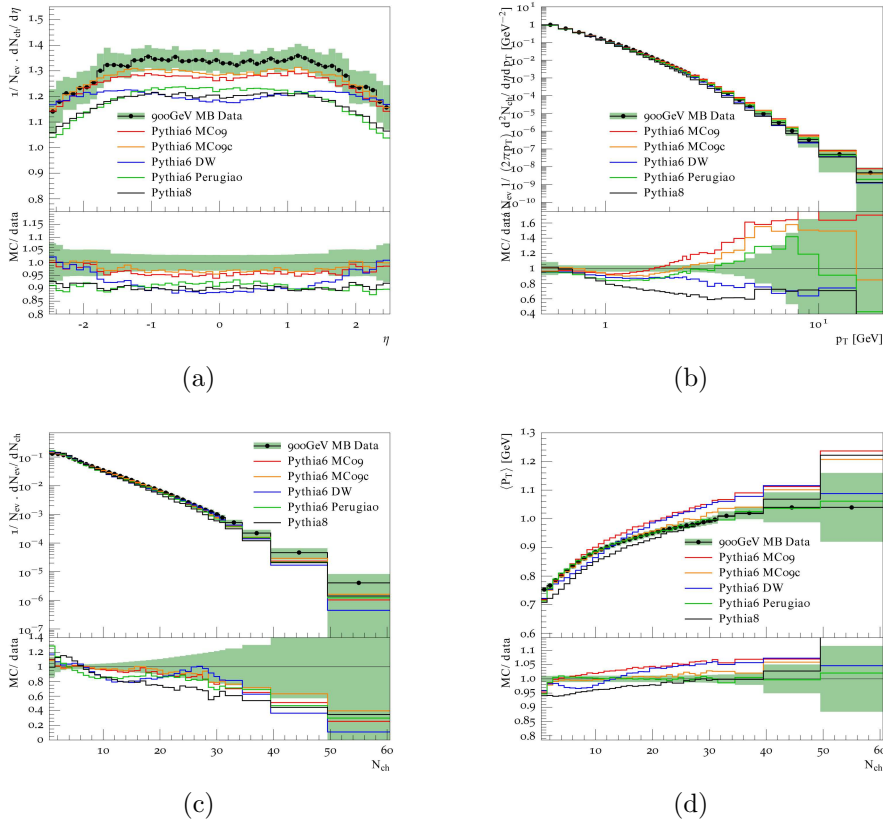


**Figure 7.8.:** Plots from the Rivet MinBias analysis performed independently of Athena, showing (a)  $\eta$  distribution, (b)  $p_T$  distribution and (c) multiplicity distribution for primary charged particles. Also shown is a plot of (d) the mean  $p_T$  vs. charged particle multiplicity.



Since Rivet has been linked into the ATLAS Athena framework, it is possible also to run this analysis from within the framework and compare the differences to standalone. When doing this it was important to note that when running from within Athena, the standard Pythia6 tune job options files contain additional flags to set the top,  $W$  and  $Z$  masses to 172.5 GeV, 80.403 GeV and 91.1876 GeV respectively. To get a fair comparison, when performed standalone, the identical values were used. Additionally, in the standard job options within Athena, the mean lifetime cut  $\tau > 0.3 \times 10^{-10}$  s is included. This too needed to be explicitly turned on in the standalone run. Figure 7.9 then shows the plots made from the same analysis and generator set-up/tunes run from within Athena.

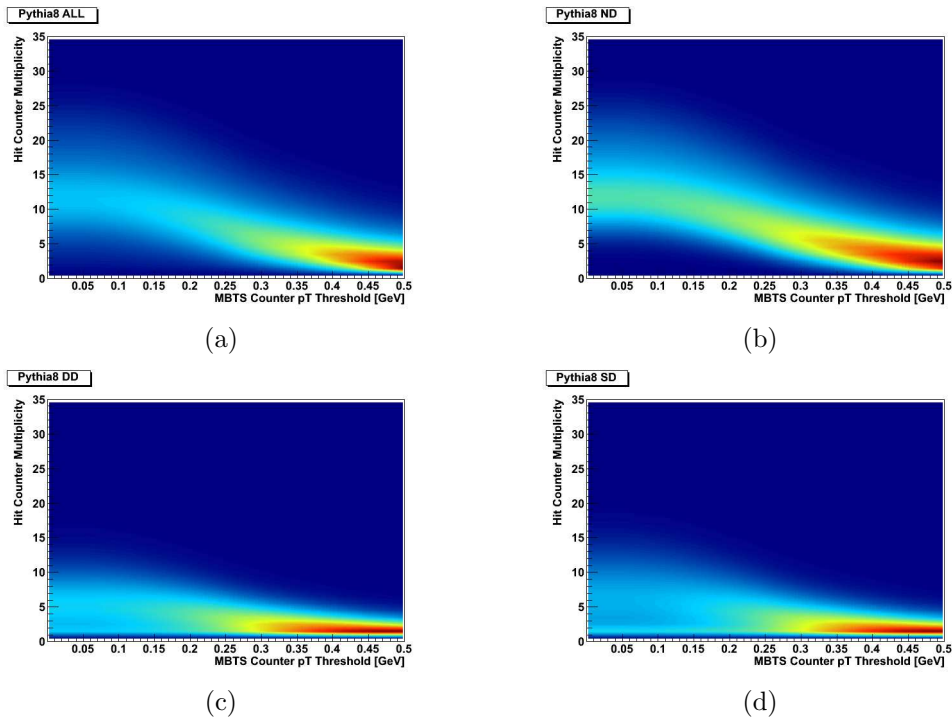
By comparison between the two figures one can see that, after ensuring that the masses used were the same, as was the lifetime cut, running standalone and within Athena produce comparable results. It is encouraging to know that nothing untoward has been introduced by merely changing the operating framework. In addition one can see that the Rivet minimum bias results are also comparable to those from the full analysis (Figure 7.6), thereby validating the Rivet analysis for future use.



**Figure 7.9.:** Plots from the Rivet MinBias analysis performed within Athena, showing (a)  $\eta$  distribution, (b)  $p_T$  distribution and (c) multiplicity distribution for primary charged particles. Also shown is a plot of (d) the mean  $p_T$  vs. charged particle multiplicity.

## 7.11. MBTS Simulation

Since the MBTS is essentially a collection of scintillator counters, the extent to which they could be used to help differentiate between the different diffractive models was investigated. A simple collection of C++ counting objects was constructed to represent the MBTS. Each of the counters contained therein knew the  $\eta$ - $\phi$  range of their own acceptance. In addition, the ability to specify a  $p_T$  threshold for each counter was included, below which an incident particle would register no hit.



**Figure 7.10.:** MBTS Threshold diagrams for Pythia 8 samples containing (a) standard mixture of elastic and in-elastic events, (b) just the non-diffractive events, (c) just the double diffractive events and (d) just the single diffractive events.

Using these counter objects it was possible to investigate how changing the threshold for the individual MBTS counters to fire affected the multiplicity of fired counters within an event. This was first done using only one generator/model in order to see the contributions from the different diffractive event types. Only events with at least one counter firing were considered as part of this study. Figure 7.10 shows a 2D histogram binned in MBTS counter threshold and hit counter multiplicity for the different diffractive event types from the Pythia8 Monte Carlo simulation. In these figures, a ‘hot/cold’ colour scheme is used where red indicates the highest density and blue the lowest. Integrating over any vertical slice (counter  $p_T$

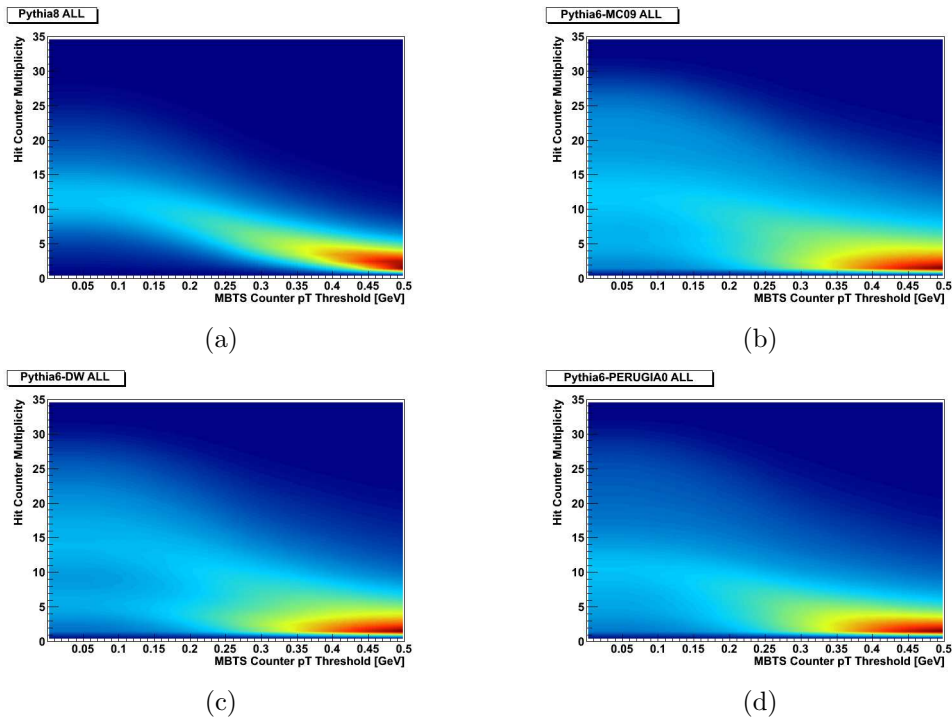
threshold) gives the same number of events<sup>§</sup>. As is expected, lowering the MBTS threshold has the effect of increasing the hit counter multiplicity in all cases. This can be seen in the figures by the broadening of the band and the change of colour (density) from red to blue as one goes from right to left. One can also see, as is expected from the physics/topology, that there is a higher hit counter multiplicity on average for the non-diffractive sample than for either of the diffractive ones. Since the non-diffractive events have the largest cross-section it can be seen that the standard mixture in Figure 7.10(a) is dominated by the non-diffractive contribution.

Having set up and tested the code to do this, it was now trivial to repeat the procedure with the standard Pythia cross-section mix of diffractive and non-diffractive samples for both Pythia6 and Pythia8. Plots were made comparing the various tunings of Pythia6; ATLAS MC09, DW, and Perugia0 (defined above in Section 7.2) as well as Pythia8. Figure 7.11 shows these plots side-by-side for ease of comparison. Clear differences can be seen between the older Pythia6 model and the newer one in Pythia8. These differences become even more noticeable as the  $p_T$  threshold of the MBTS counters is reduced. This was as far as this investigation was taken before the author began writing up. It is worth noting however that this is done at the generator level with the effect of the magnetic field taken into account. The code for the spiralling of particles within the magnetic field was never fully tested so these plots in their current state are merely meant as a guide to where possible future work could take place.

Finally since the fast trigger simulation had been created these plots provided another good test. The threshold plots already mentioned were

---

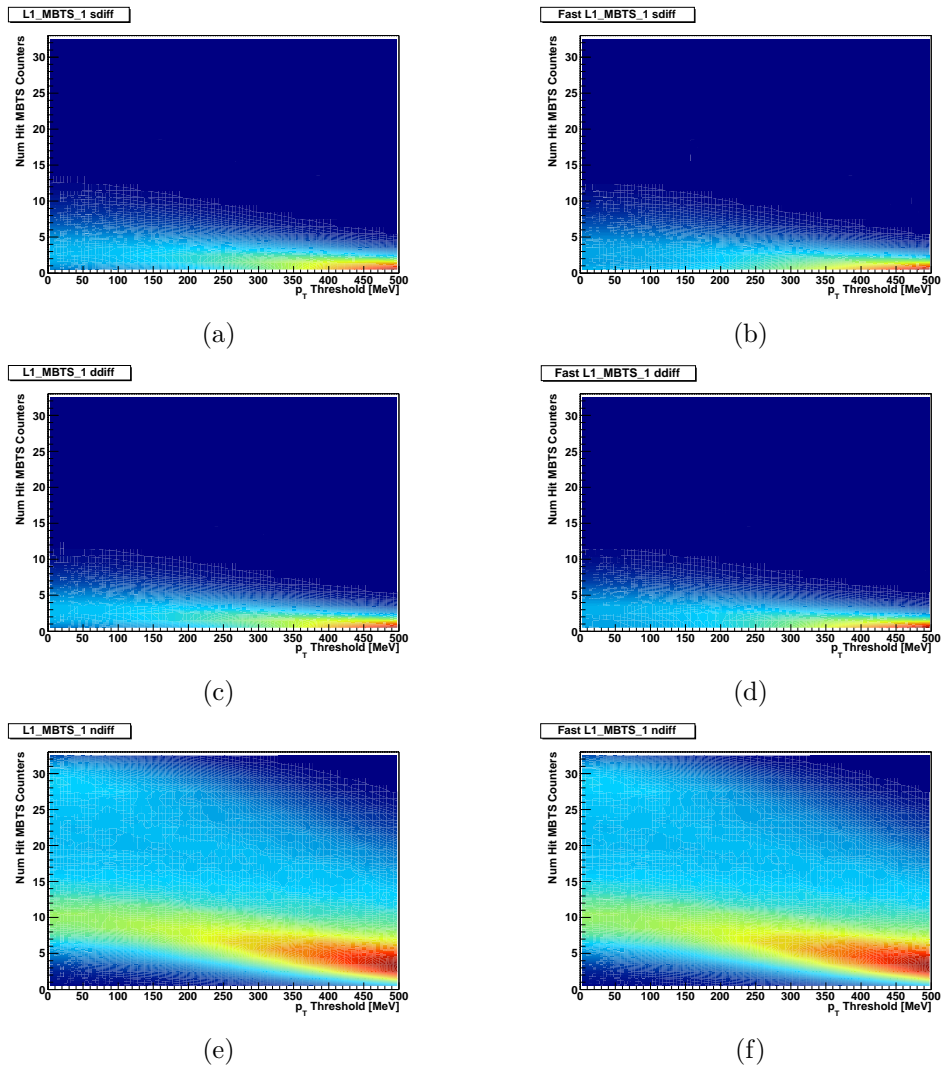
<sup>§</sup>In practice, while identical events went into each slice, those slices with the highest  $p_T$  threshold actually have slightly lower integrated number of events. This is because of the requirement that at least one counter fire per event. Thus by raising all the counter's threshold one increases the proportion of events in which no counters fired and as such the proportion that are discounted.



**Figure 7.11.:** MBTS Threshold diagrams for (a) Pythia 8 and (b)(c)(d) Pythia 6 samples containing standard mixture of elastic and in-elastic events, with (b) ATLAS MC09 tuning, (c) DW tuning and (d) Perugia0 tuning.

again created for each of the non/diffractive samples this time using the Pythia6 generator. In this instance Figure 7.12 shows the hit MBTS counter multiplicity as a function of the MBTS threshold only for events in which the full L1\_MBTS\_1 minimum bias trigger simulation was passed. In the same figure and next to each of these are the same plots for those events passing the fast L1\_MBTS\_1 trigger.

As can be seen by looking at Figure 7.12 there is very little *if any* difference between the two trigger responses visible which is not too surprising since it was the L1\_MBTS\_1 trigger in precisely this acceptance from which the parameterisation for the fast trigger was acquired.



**Figure 7.12.:** Comparison of MBTS threshold plots between the (a,c,e) full trigger simulation and (b,d,f) fast trigger simulation. Shown is the hit MBTS counter multiplicity against counter  $p_T$  threshold for a (a,b) single, (c,d) double and (e,f) non-diffractive event sample.

## 7.12. Summary and Conclusion

The first ATLAS inclusive measurement of charged particle multiplicities in events with  $n_{ch} \geq 1$  within the kinematic range  $p_T > 500$  GeV and  $|\eta| < 2.5$  was presented. With a measured charged particle multiplicity per event and

per unit of pseudorapidity at  $\eta = 0$  of  $1.333 \pm 0.003(\text{stat.}) \pm 0.040(\text{syst.})$  being some 5-15% higher than predicted, clear differences are evident between the Monte Carlo predictions and what is observed in the data.

The author's work on the trigger efficiencies for other level 1 MBTS triggers was presented for single, double and non-diffractive event samples. This was then compared to the efficiency measured using the author's fast trigger simulation and found to be comparable in shape in most cases but that the fast trigger tends to overestimate the fully simulated trigger by  $\sim 5\%$  in the worst case for the single diffractive and a bit more for the double diffractive case,  $\sim 10\%$  in worst case. Given the low statistics put into the parameterisation of the fast trigger which was at that point in its infancy and still undergoing testing, this represents a good level of agreement and clearly would benefit from being pursued.

The Rivet toolkit was introduced and the author's contribution of a Rivet version of the ATLAS minimum bias analysis was shown. This analysis validates well against the ATLAS analysis when run both standalone as well as from within the ATLAS code framework Athena.

A simple way of simulating the MBTS as a collection of individual counters was presented and used to illustrate the difference in hit counter multiplicity between non/diffractive samples. It was also used to highlight the difference in the low  $p_T$  models between different tunes of the Pythia6 and Pythia8 Monte Carlo simulations. These MBTS threshold plots also served as another way of comparing the full and fast simulated triggers which showed little difference as expected in this acceptance.

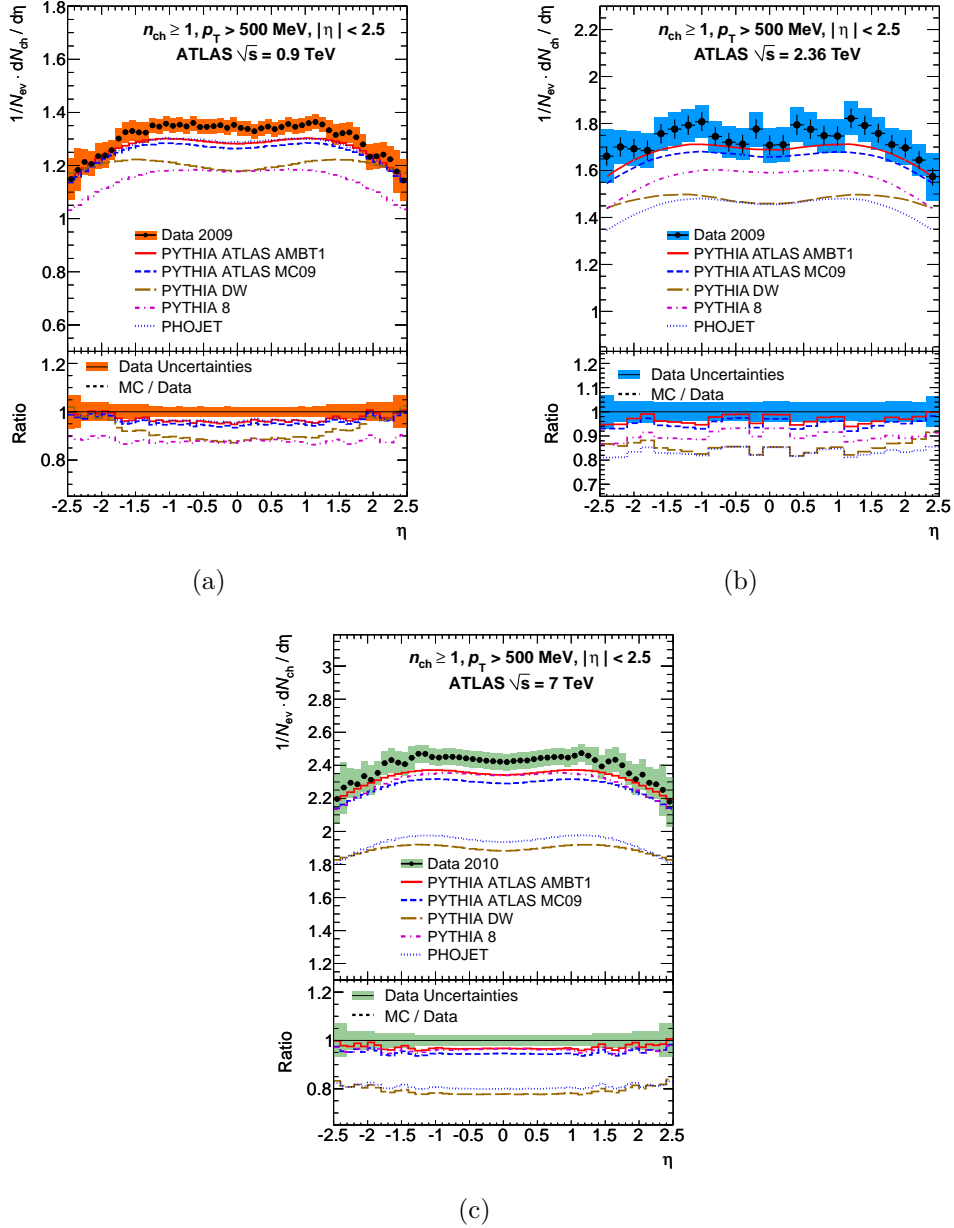
Since the paper discussed in this chapter was published, the author ceased active participation within the minimum bias group in order to write up this thesis. The group itself however has remained active through-

out and has submitted a new paper [70] for publication in ‘New J. Phys.’ which builds on the presented analysis with data obtained from the ATLAS detector at  $\sqrt{s} = 2.36$  TeV and also  $\sqrt{s} = 7$  TeV.

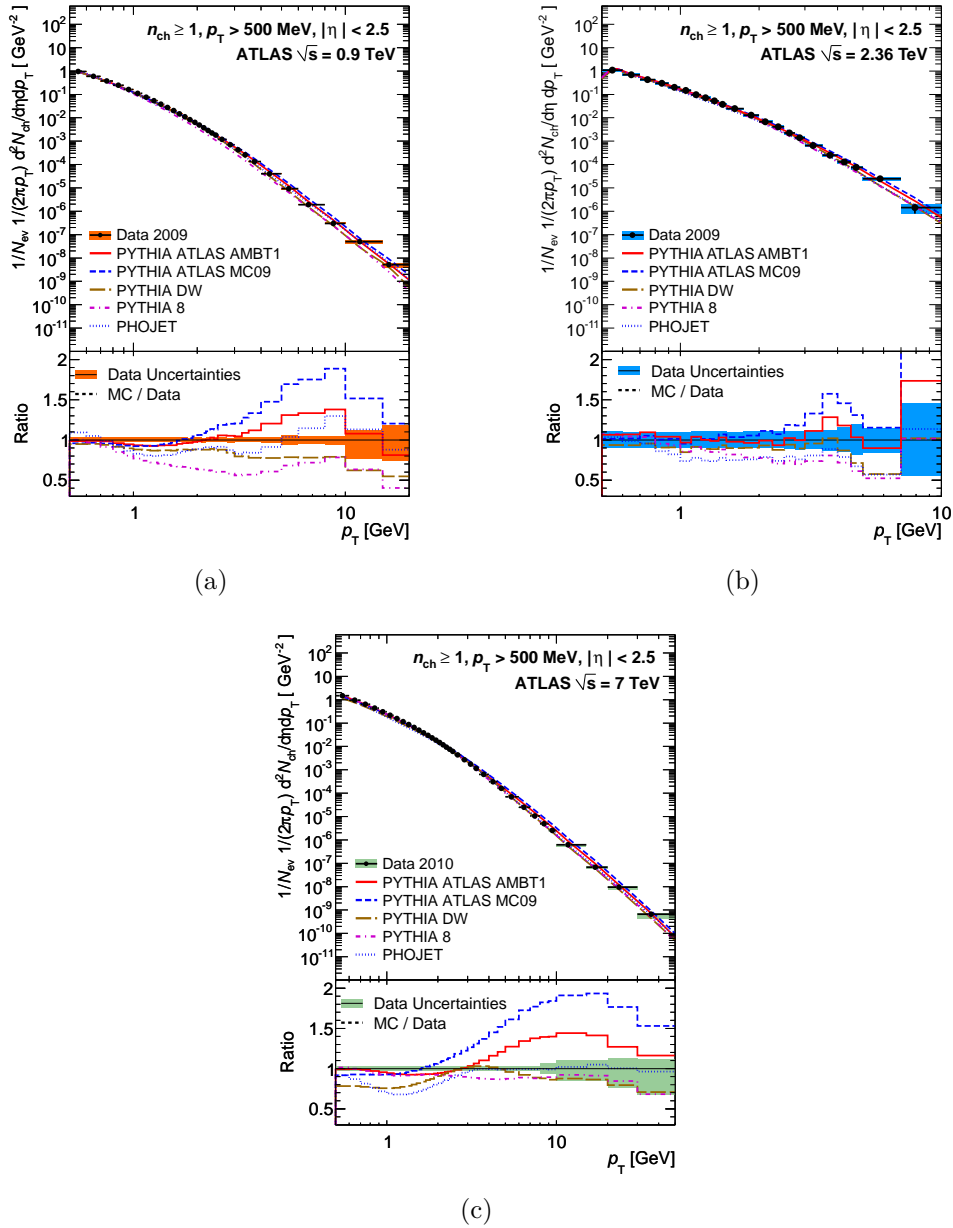
Figure 7.13 shows the charged particle multiplicity as a function of pseudorapidity for the original analysis at  $\sqrt{s} = 0.9$  TeV as well as at the two new centre-of-mass energies above. Figure 7.14 shows the multiplicity as a function of  $p_T$  and Figure 7.15 show the multiplicity of charged particles per event as in the original analysis described in this chapter. For the distribution of the mean  $p_T$  as a function of charged particle multiplicity, Figure 7.16 shows only distributions from data collected at (a)  $\sqrt{s} = 0.9$  TeV and (b)  $\sqrt{s} = 7$  TeV. The omission of a distribution from data collected at  $\sqrt{s} = 2.36$  TeV is due to the fact that different track reconstruction methods were used for determining the  $p_T$  and multiplicity distributions as described in the new paper [70].

Finally it is worth noting that in this new paper the analysis was further extended to consider events with  $n_{ch} \geq 2$  with  $p_T > 100$  MeV and also events with  $n_{ch} \geq 6$  with  $p_T > 500$  MeV. In both cases the acceptance was kept the same as for the original analysis described in this chapter  $|\eta| < 2.5$ .

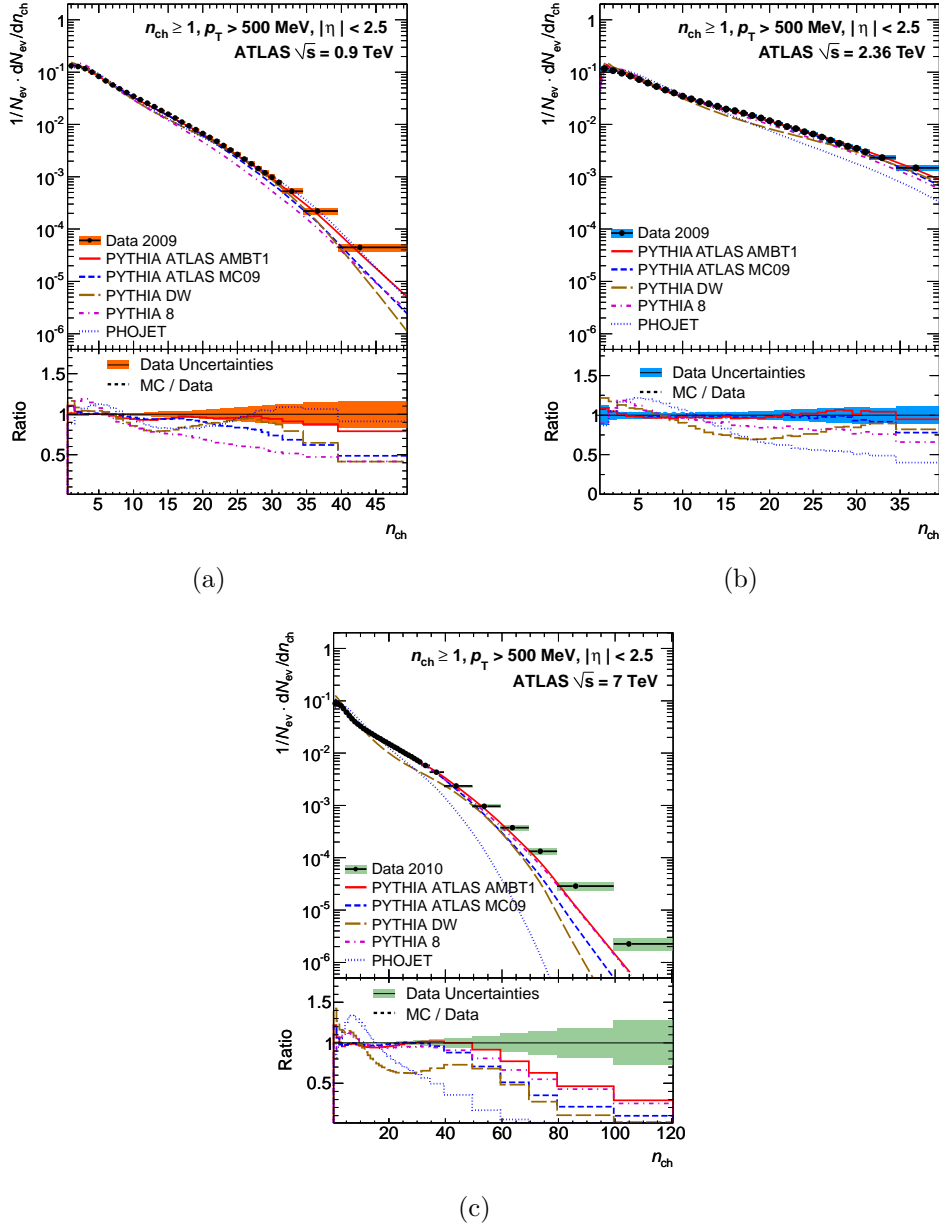




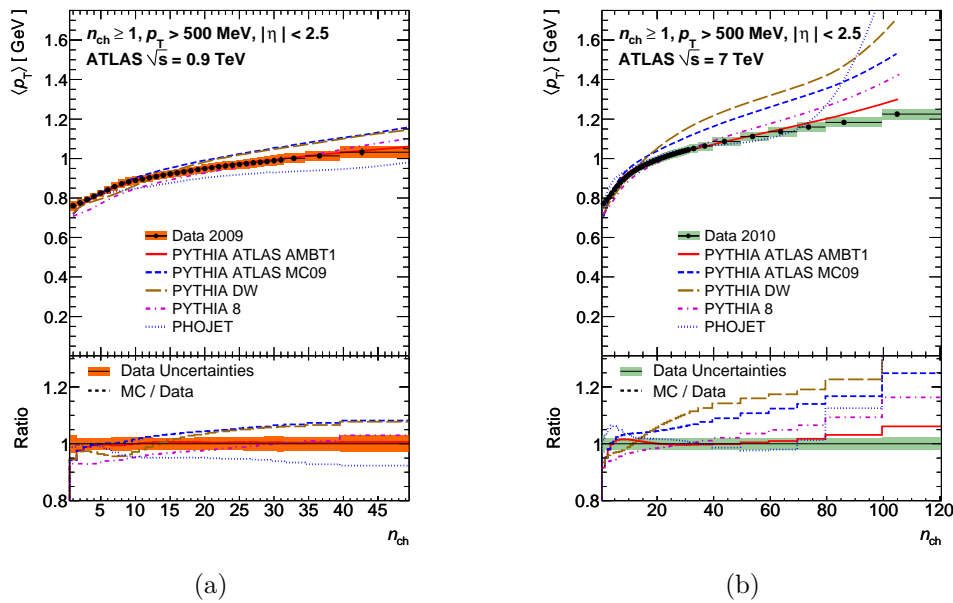
**Figure 7.13.:** Charged-particle multiplicities as a function of the pseudorapidity for events with  $n_{ch} \geq 1$ ,  $p_T > 500$  MeV and  $|\eta| < 2.5$  at  $\sqrt{s} =$  (a) 0.9 TeV, (b) 2.36 TeV and (c) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70].



**Figure 7.14.:** Charged-particle multiplicities as a function of the transverse momentum for events with  $n_{ch} \geq 1$ ,  $p_T > 500$  MeV and  $|\eta| < 2.5$  at  $\sqrt{s} =$  (a) 0.9 TeV, (b) 2.36 TeV and (c) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70].



**Figure 7.15.:** Charged-particle multiplicity distributions for events with  $n_{ch} \geq 1$ ,  $p_T > 500 \text{ MeV}$  and  $|\eta| < 2.5$  at  $\sqrt{s} =$ (a) 0.9 TeV, (b) 2.36 TeV and (c) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70].



**Figure 7.16.:** Average transverse momentum as a function of the number of charged particles in the event for events with  $n_{ch} \geq 1$ ,  $p_T > 500$  MeV and  $|\eta| < 2.5$  at  $\sqrt{s} =$  (a) 0.9 TeV and (b) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70].

## Chapter 8.

### Conclusion

In conclusion, a detailed introduction to ATLAS code has been presented, including the Athena framework itself, the Atlfast fast simulation software and the RTT code maintenance/validation framework. Within the RTT the author had a developer role and was also responsible for running the daily shifts. Within Atlfast the author was a key developer who was entirely responsible for a brand new framework for allowing easy modification of the functionality of Atlfast's calorimeters and smearers/reconstructors in as user-safe, scalable way as possible. This framework was then tested and by way of example was used to investigate the effects of hot and dead cells within the calorimeter. The new framework performed as expected and provided insight into the clustering of jets around hot cells.

A 'fast' trigger simulation was presented. While this was initially intended to be useful within Atlfast for fast comparison with new data (thereby guiding the more detailed studies), it was developed from the ground up to be as independent as possible, meaning it could easily be used standalone by any other code. The trigger was a two dimensional look-up table stored in the popular and convenient ROOT [71] 'TH2D' his-

togram format. The advantages of this format were its prevalence within the community and that any user could open the parameterisation file and visualise it using the ROOT GUI. This fast trigger parameterised the response of the official fully simulated L1\_MBTS\_1 trigger in both  $p_T$  as well as  $\eta$ . This was done for a single pion sample to get the *per particle* efficiency which could then be applied sequentially to all particles within the acceptance. From this one parameterisation the decisions of the L1\_MBTS\_2 and L1\_MBTS\_1.1 triggers were inferred. This fast trigger simulation was tested by comparing the efficiencies binned in  $\eta$ ,  $p_T$  and  $N_{ch}$  with those from the fully simulated versions and was found to be in good agreement albeit slightly overestimating the response. This was also the case when plotting the  $p_T$  spectra of a single diffractive sample with all the aforementioned triggers. The reason for this overestimation warrants further investigation, however due to the small size of the sample used to obtain the parameterisation, this could be a statistical effect that would disappear were a larger sample used.

A somewhat crude MBTS simulation was presented consisting of essentially a collection of counter objects. These counters knew their own acceptances and in addition had a  $p_T$  threshold below which they would register no hit. This crude simulation was used to show how the hit counter multiplicity varies with this  $p_T$  threshold, with the unsurprising result that the lower the threshold the higher the multiplicity. It also clearly showed the model differences between the Pythia6 and Pythia8 generators. Finally it acted as yet another test of the fast trigger simulation which performed very well in this kinematic range as was expected.

An early data supersymmetry search in the 0-lepton + jets +  $E_T^{miss}$  channel was outlined showing the validity of looking for SUSY mSUGRA SU3 benchmark events with low ( $< 4$ ) jet multiplicities. Excluding the sys-

tematics on the background determination, significances well above the  $5\sigma$  level are seen in both the 2-jet and 3-jet analyses and in both the  $M_{eff}$  and  $E_T^{miss}$  plots. The channel's validity can also be seen for the SUSY SU4 mSUGRA benchmark point in more recent publications of the SUSY group. With the limited initial start-up data, no deviation from the SM was detected up to values of  $E_T^{miss} \sim 100$  GeV and  $M_{eff} \sim 1500$  GeV.

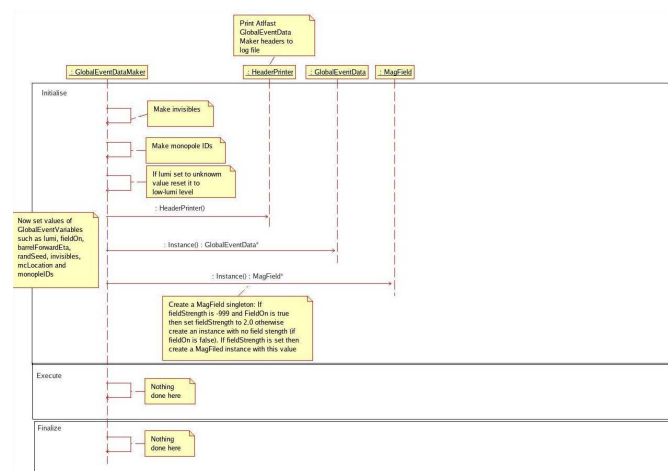
Finally the first published ATLAS minimum bias analysis at  $\sqrt{s} = 900$  GeV was presented, highlighting the author's work on trigger efficiencies which were also compared to the fast trigger simulation. The author coded a version of the analysis for the Rivet validation toolkit which was run standalone and from within the ATLAS Athena framework. The results show good agreement with each other as well as with the published results of the official analysis. This both validates the Rivet analysis routine as well as confirming that the use of the ATLAS Athena framework does not influence the results. The official results of this analysis as well as the more recent update to this analysis at  $\sqrt{s} = 2.36$  TeV and  $\sqrt{s} = 7$  TeV show poor agreement between all tested Monte Carlo event generators and the data indicating an area warranting further study/tuning.

# Appendix A.

## Atlfast Algorithm Flow Diagrams

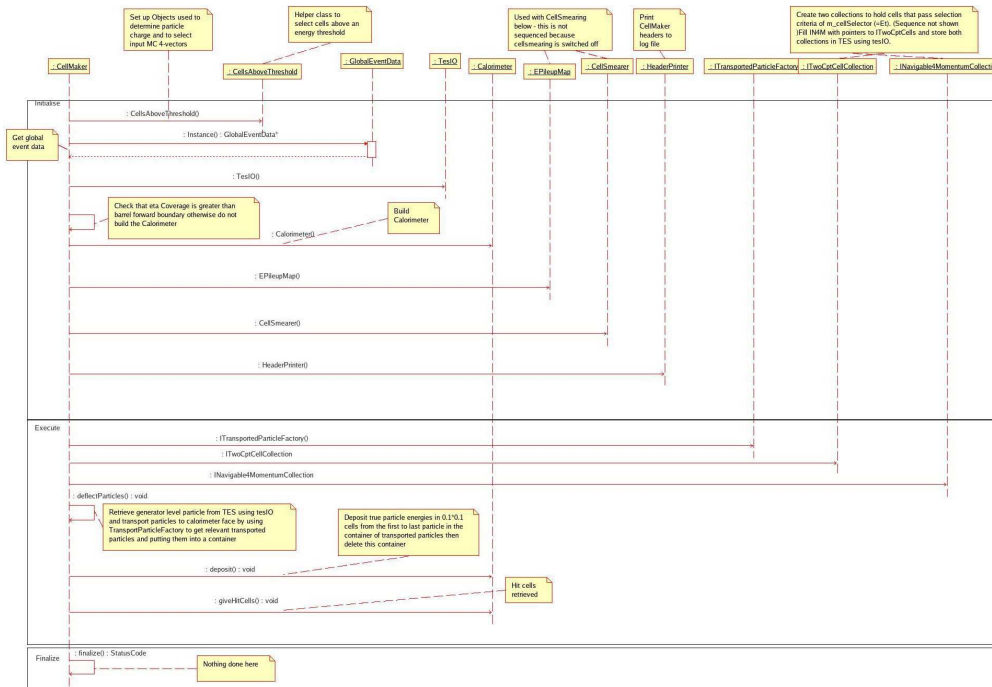
This appendix shows (for completeness) flow diagrams for the various Atlfast algorithms. They were made by Thomas Doherty of the University of Glasgow.

### A.1. GlobalEventDataMaker

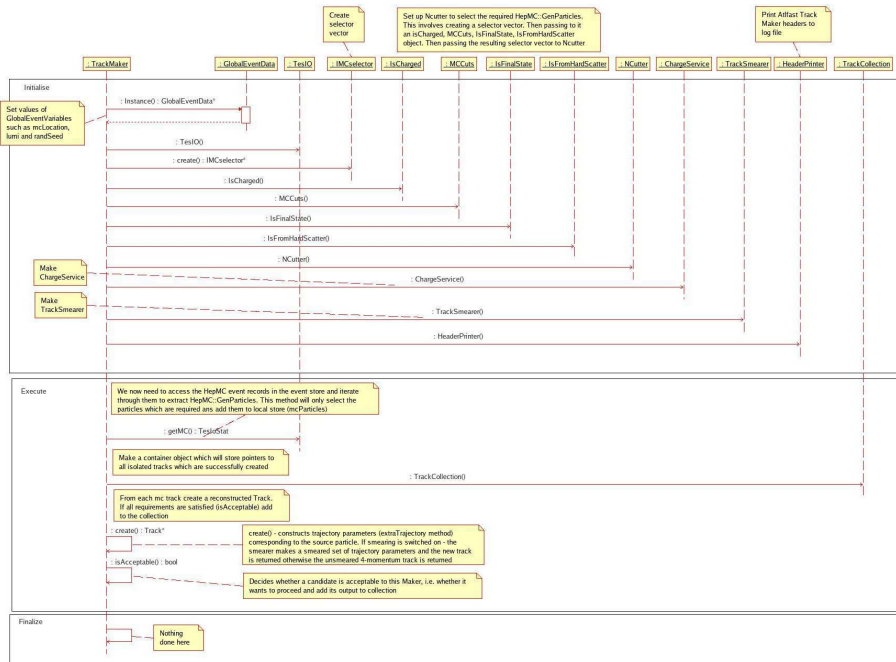




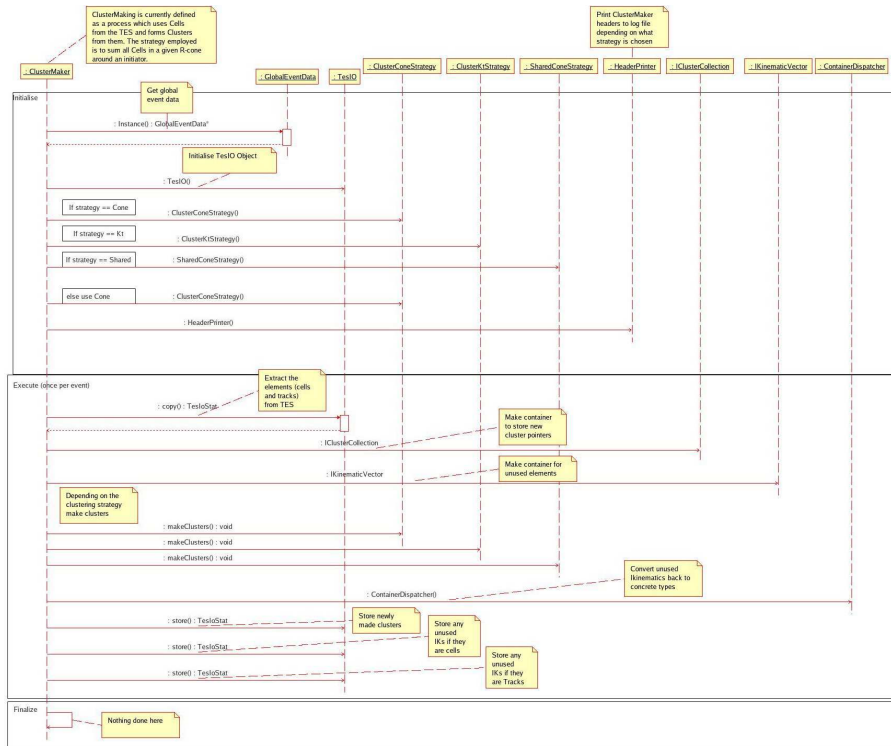
## A.2. CellMaker



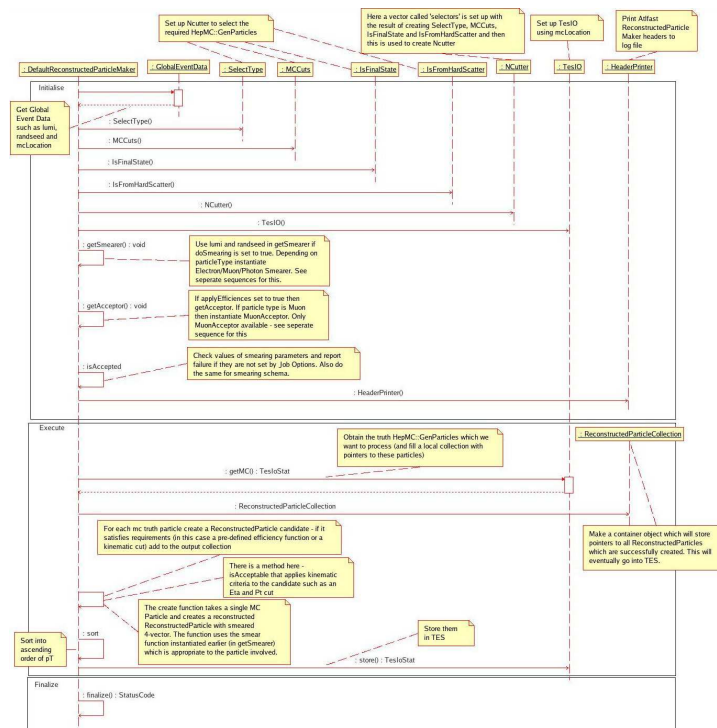
### A.3. TrackMaker



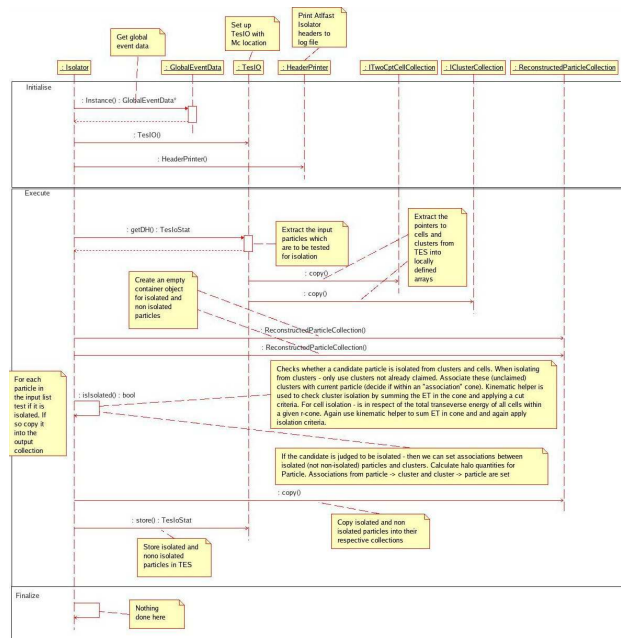
## A.4. ClusterMaker



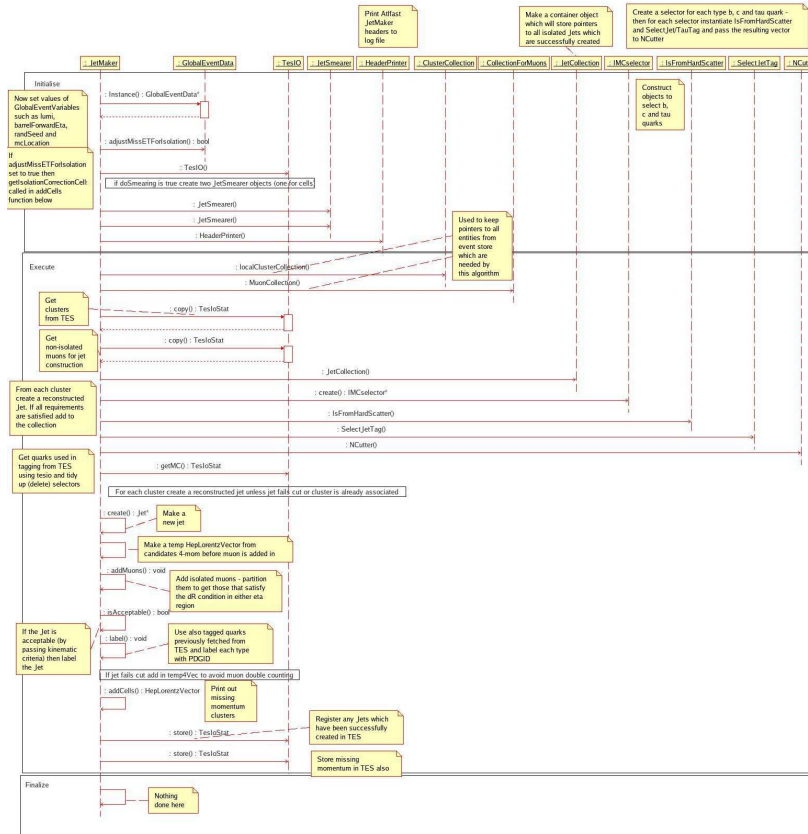
## A.5. DefaultReconstructedParticleMaker



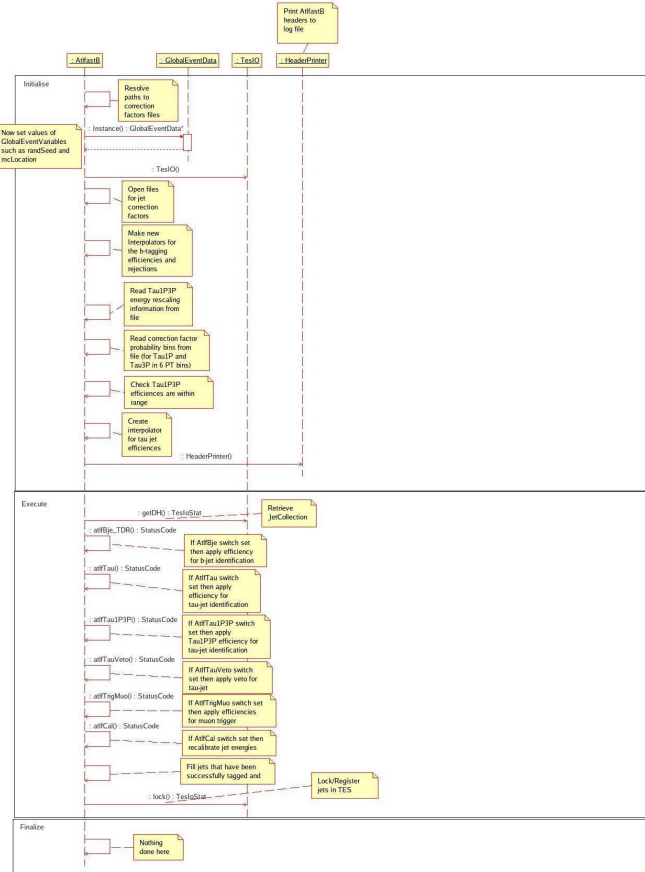
## A.6. Isolator



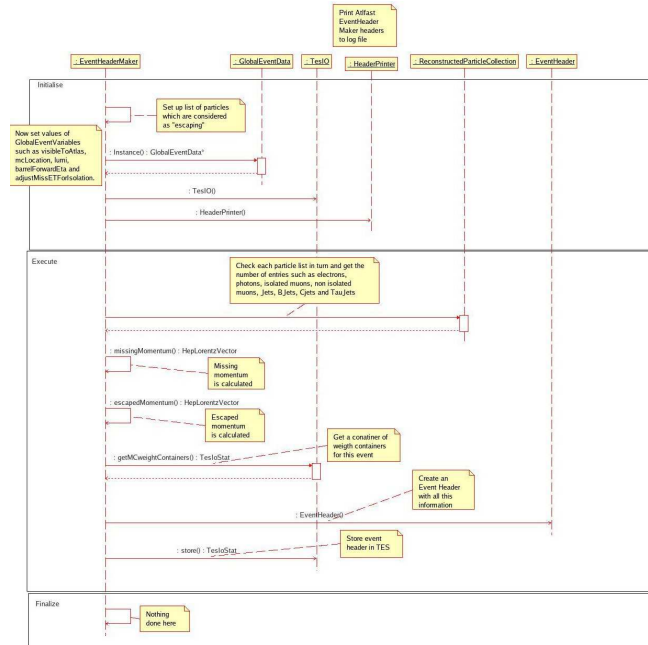
### A.7. JetMaker



## A.8. AtlfastB



## A.9. EventHeaderMaker





## Appendix B.

# ATLAS Minimum Bias Rivet Analysis Source Code

This appendix shows the source code for the the ATLAS minimum bias Rivet analysis which the author wrote in parallel with another. Though algorithmically identical, it was the other author's version that was put into official Rivet production as the other author had more of a connection with the Rivet project.

```
// -*- C++ -*-
// Rivet Minimum Bias Analysis
// author: Alexander Richards - University College London
#include "Rivet/Analysis.hh"
#include "Rivet/Tools/Logging.hh"
#include "Rivet/Projections/ChargedFinalState.hh"
#include "Rivet/RivetAIDA.hh"
#include "Rivet/Tools/BinnedHistogram.hh"

namespace Rivet{

class ATLASMinBias : public Analysis{

public:

  ATLASMinBias(): Analysis("ATLASMINBIAS"), m_NevTot(0.0){}

  void init(){

    //////////////////////////////////////
    /// Charged Particle: ///
    /// ----- ///
    ///          ///
    /// 1) Pt > 500 MeV  ///
    /// 2) |eta| < 2.5   ///
    //////////////////////////////////////

    /// Cut out all but our definition of a Charged Particle
    /// above
    const ChargedFinalState cfs(-2.5, 2.5, 0.5*GeV);
    addProjection(cfs, "MinBiasProjection");

    /// Setup the histograms taking the binning from the
    /// reference data
    m_EtaNormalised = bookHistogram1D(1, 1, 1);
    m_PtNormalised = bookHistogram1D(1, 1, 2);
    m_NchNormalised = bookHistogram1D(1, 1, 3);
    m_meanPt_vs_Nch = bookProfile1D(1, 1, 4);

    return;
  }
}
```

```
void analyze(const Event &evt){

    double weight = evt.weight();

    const ChargedFinalState &cfs =
        applyProjection<ChargedFinalState>(evt,
                                           "MinBiasProjection");

    if(cfs.size() < 1) vetoEvent;

    ParticleVector tracks = cfs.particles();

    // Keep track of number of events for normalising later
    m_NevTot+=weight;

    /// Filling the per particle plots Eta and Pt
    //////////////////////////////////////

    ParticleVector::const_iterator track;
    for(track = tracks.begin();
        track != tracks.end();
        ++track){

        /// Non-Normalised plots
        m_meanPt_vs_Nch->fill(cfs.size(),
                             track->momentum().pT(),
                             weight);

        /// Plots to be normalised
        m_EtaNormalised->fill(track->momentum().eta(),
                              weight);
        m_PtNormalised->fill(track->momentum().pT(),
                              weight/track->momentum().pT() );
    }

    /// Filling the per Event plots Nch
    //////////////////////////////////////

    /// Plots to be normalised
    m_NchNormalised->fill(cfs.size(),weight);

    return;
}
```

```
void finalize(){

    if(!m_NevTot) return;

    double PI=3.141592654;

    /// The acceptance in eta is one bin -2.5 -> 2.5
    /// i.e. 5.0 units
    double deta = 5.0;

    /// Normalising plots
    m_NchNormalised->scale( 1/m_NevTot );
    m_EtaNormalised->scale( 1/m_NevTot );
    m_PtNormalised->scale( 1/(2*PI*deta*m_NevTot) );
    return;
}

private:

    /// Double keeping track of the sum of all bins in the
    /// Nch Histo to use for normalisation
    /// i.e. Nev (Number of events)
    double m_NevTot;

    /// The Number of Charged Particles / Nch(Tot)
    AIDA::IHistogram1D *m_NchNormalised;

    /// The Eta of Charged Particles / Nch(Tot)
    AIDA::IHistogram1D *m_EtaNormalised;

    /// The Pt of Charged Particles / Nch(Tot)
    AIDA::IHistogram1D *m_PtNormalised;

    /// The <Pt> of Charged Particles vs Nch
    AIDA::IProfile1D *m_meanPt_vs_Nch;

};

AnalysisBuilder<ATLASMinBias> plugin_ATLASMinBias;
}
```

# Bibliography

- [1] David Griffiths. *Introduction to Elementary Particles*. WILEY-VCH, 2004.
- [2] Y. Fukuda et al. Evidence for Oscillation of Atmospheric Neutrinos. *Phys. Rev. Lett.*, 81(8):1562–1567, Aug 1998.
- [3] CERN Bulletin. Neutrino oscillations make their first appearance in OPERA. 23/2010(BUL-NA-2010-134. 23/2010):6, Jun 2010.
- [4] Torbjorn Sjostrand, Stephen Mrenna, and Peter Z. Skands. A Brief Introduction to PYTHIA 8.1. *Comput. Phys. Commun.*, 178:852–867, 2008. CERN-LCGAPP-2007-04, LU TP 07-28, FERMILAB-PUB-07-512-CD-T.
- [5] Torbjorn Sjostrand, Stephen Mrenna, and Peter Z. Skands. PYTHIA 6.4 Physics and Manual. *JHEP*, 05:026, 2006.
- [6] G. Corcella et al. HERWIG 6.5: an event generator for Hadron Emission Reactions With Interfering Gluons (including supersymmetric processes). *JHEP*, 01:010, 2001.
- [7] B. R. Webber. Fragmentation and hadronization. *Int. J. Mod. Phys. A*, 15S1:577–606, 2000. CERN-TH/99-387, Cavendish-HEP-99/16.
- [8] P. Skands. Min-Bias Models - The modeling of SD, DD, and

- ND. <http://indico.cern.ch/conferenceOtherViews.py?view=standard&confId=83609>, March 2010. presented at LHC wide minimum bias and underlying event working group meeting.
- [9] LHC Webmaster. LHC Homepage. <http://lhc.web.cern.ch/lhc/>.
- [10] The ATLAS collaboration. *ATLAS detector and physics performance: Technical Design Report, 1*. Technical Design Report ATLAS. CERN, Geneva, 1999. ATLAS-TDR-014, CERN-LHCC-99-14.
- [11] The ATLAS collaboration. LHC Design Report Volume III: The LHC Injector Chain. <http://lhc.web.cern.ch/lhc/LHC-DesignReport.html>.
- [12] The ATLAS collaboration. LHC Design Report Volume I: The LHC Main Ring. <http://lhc.web.cern.ch/lhc/LHC-DesignReport.html>.
- [13] The ATLAS collaboration. ATLAS Experiment Homepage. <http://www.atlas.ch/>.
- [14] The ATLAS collaboration. LHC Design Report Volume II: The LHC Infrastructure and General Services. <http://lhc.web.cern.ch/lhc/LHC-DesignReport.html>.
- [15] JabberWok. A graphic showing the relationship between angle and pseudorapidity. (Pseudorapidity2.png). <http://en.wikipedia.org/wiki/Pseudorapidity>.
- [16] The ATLAS collaboration. ATLAS inner detector: Technical design report. Vol. 1. 1997. CERN-LHCC-97-16, ATLAS-TDR-4.
- [17] The ATLAS collaboration. ATLAS inner detector: Technical design report. Vol. 2. 1997. CERN-LHCC-97-17.

- 
- [18] The ATLAS collaboration. ATLAS SCT Gallery. [http://atlas.web.cern.ch/Atlas/GROUPS/INNER\\_DETECTOR/SCT/gallery/](http://atlas.web.cern.ch/Atlas/GROUPS/INNER_DETECTOR/SCT/gallery/).
- [19] R. K. Bock and A. Vasilescu. The Particle Detector BriefBook. <http://rkb.home.cern.ch/rkb/titleD.html>.
- [20] The ATLAS collaboration. ATLAS tile calorimeter: Technical design report. 1996. CERN-LHCC-96-42.
- [21] The ATLAS collaboration. ATLAS liquid argon calorimeter: Technical design report. 1996. CERN-LHCC-96-41.
- [22] The ATLAS collaboration. ATLAS muon spectrometer: Technical design report. 1997. CERN-LHCC-97-22, ATLAS-TDR-10.
- [23] William Bell et al. MbtsTrigger. <https://twiki.cern.ch/twiki/bin/view/Atlas/MbtsTrigger>, Aug 2009.
- [24] Eric Feng et al. MinimumBiasTriggerScintillatorInfo. <https://twiki.cern.ch/twiki/bin/view/Atlas/MinimumBiasTriggerScintillatorInfo>, Mar 2010.
- [25] S. Agostinelli et al. G4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250 – 303, 2003.
- [26] D Cavalli, D Costanzo, S Dean, M Dhrssen, S Hassani, M Heldmann, K Jakobs, A Nairz, A Phillips, S Resconi, E Richter-Was, P Sherwood, L Vacavant, I Vivarelli, J B De Vivie de Rgie, and I Wingerter-Seez. Performance of the ATLAS fast simulation ATLFAST. Technical Report ATL-PHYS-INT-2007-005. ATL-COM-PHYS-2007-012, CERN, Geneva, Jan 2007.
- [27] J. M. Butterworth, J. P. Couchman, B. E. Cox, and B. M. Waugh. Kt-

- Jet: A C++ implementation of the K(T) clustering algorithm. *Comput. Phys. Commun.*, 153:85–96, 2003.
- [28] Gavin Salam Matteo Cacciari and Gregory Soyez. FastJet. <http://www.lpthe.jussieu.fr/~salam/fastjet/>.
- [29] S. Catani, Yuri L. Dokshitzer, M. H. Seymour, and B. R. Webber. Longitudinally invariant  $K_t$  clustering algorithms for hadron hadron collisions. *Nucl. Phys. B*, 406:187–224, 1993.
- [30] Stephen D. Ellis and Davison E. Soper. Successive combination jet algorithm for hadron collisions. *Phys. Rev. D*, 48:3160–3166, 1993.
- [31] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. The anti- $k_t$  jet clustering algorithm. *JHEP*, 04:063, 2008.
- [32] Yuri L. Dokshitzer, G. D. Leder, S. Moretti, and B. R. Webber. Better Jet Clustering Algorithms. *JHEP*, 08:001, 1997.
- [33] M. Wobisch and T. Wengler. Hadronization corrections to jet cross sections in deep- inelastic scattering. *arXiv:hep-ph/9907280v1*, 1998. PITHA 99/16, Proceedings of the Workshop on Monte Carlo Generators for HERA Physics.
- [34] M. Wobisch. *Measurement and QCD analysis of jet cross sections in deep-inelastic positron proton collisions at  $s^{**}(1/2) = 300\text{-GeV}$* . PhD thesis, Rheinisch-Westfal. TU, Aachen, 2000. DESY-THESIS-2000-049.
- [35] Matteo Cacciari and Gavin P. Salam. Dispelling the  $N^3$  myth for the  $k_t$  jet-finder. *Phys. Lett. B*, 641:57–61, 2006.
- [36] Gavin P. Salam and Gregory Soyez. A practical Seedless Infrared-Safe Cone jet algorithm. *JHEP*, 05:086, 2007.



- [37] B. Simmons. The RunTimeTester User Guide. <https://atlasrtdt.cern.ch/docs/site/guide>.
- [38] Stefan Ritt. The ELOG Home Page. <https://midas.psi.ch/elog>.
- [39] I. M. Chakravarti, R. G. Laha, and J. Roy. *Handbook of Methods of Applied Statistics*, volume I. John Wiley and Sons, USE, 1967.
- [40] N. D. Gagunashvili. Comparison of weighted and unweighted histograms. *arXiv:physics/0605123v1 [physics.data-an]*, May 2006.
- [41] Krzysztof Ciba. DCube Documentation. <https://twiki.cern.ch/twiki/bin/view/Sandbox/DCubeDoc>.
- [42] Valgrind Developers. Valgrind. <http://valgrind.org/>.
- [43] Stephen P. Martin. A Supersymmetry Primer. *arXiv:hep-ph/9709356v5*, 1997.
- [44] G. Aad et al. Expected Performance of the ATLAS Experiment - Detector, Trigger and Physics. *arXiv:0901.0512v4 [hep-ex]*, 2009.
- [45] The ATLAS collaboration. Prospects for Supersymmetry discovery based on inclusive searches at a 7 TeV centre-of-mass energy with the ATLAS detector. Technical Report ATL-PHYS-PUB-2010-010, CERN, Geneva, Jul 2010.
- [46] Michelangelo L. Mangano, Fulvio Piccinini, Antonio D. Polosa, Mauro Moretti, and Roberto Pittau. ALPGEN, a generator for hard multiparton processes in hadronic collisions. *Journal of High Energy Physics*, 2003(07):001, 2003.
- [47] Stefano Frixione and Bryan R. Webber. The MC@NLO 3.2 event generator. *arXiv:hep-ph/0601192v1*, 2006.
- [48] Stefano Frixione and Bryan R. Webber. Matching NLO QCD compu-

- tations and parton shower simulations. *JHEP*, 06:029, 2002.
- [49] Glen Cowan, Kyle Cranmer, Eilam Gross, and Ofer Vitells. Asymptotic formulae for likelihood-based tests of new physics. *Eur. Phys. J. C*, 71:1554, 2011.
- [50] The ATLAS collaboration. Early supersymmetry searches in channels with jets and missing transverse momentum with the ATLAS detector. Technical Report ATLAS-CONF-2010-065, CERN, Geneva, Jul 2010.
- [51] G. Aad et al. Search for squarks and gluinos using final states with jets and missing transverse momentum with the ATLAS detector in  $\sqrt{s} = 7$  TeV proton-proton collisions. *Submitted to Phys. Lett. B*, arXiv:1102.5290v1 [hep-ex], 2011. CERN-PH-EP-2011-022.
- [52] G. Aad et al. Charged-particle multiplicities in pp interactions at  $\sqrt{s} = 900$  GeV measured with the ATLAS detector at the LHC. *Physics Letters B*, 688(1):21 – 42, 2010.
- [53] A. Capella, U. Sukhatme, Chung-I Tan, and J. Tran Thanh Van. Jets in small-pT hadronic collisions, universality of quark fragmentation, and rising rapidity plateaus. *Physics Letters B*, 81(1):68 – 74, 1979.
- [54] A. Capella, U. Sukhatme, C. I. Tan, and J. Tran Thanh Van. Dual parton model. *Physics Reports*, 236(4-5):225 – 329, 1994.
- [55] R. Engel. Photoproduction within the two-component Dual Parton Model: Amplitudes and cross sections. *Zeitschrift für Physik C Particles and Fields*, 66:203–214, 1995. 10.1007/BF01496594.
- [56] Vardan Khachatryan et al. Charged particle multiplicities in pp interactions at  $\sqrt{s} = 0.9, 2.36,$  and 7 TeV. *JHEP*, 01:079, 2011.
- [57] The ATLAS collaboration. ATLAS Monte Carlo tunes for MC09.

- Technical Report ATL-PHYS-PUB-2010-002, CERN, Geneva, Mar 2010.
- [58] A. Sherstnev and R. S. Thorne. Parton Distributions for LO Generators. *Eur. Phys. J. C*, 55:553–575, 2008.
- [59] Rivet Development Team. Rivet. <http://projects.hepforge.org/rivet>.
- [60] T. Aaltonen et al. Measurement of particle production and inclusive differential cross sections in  $p\bar{p}$  collisions at  $\sqrt{s} = 1.96$  TeV. *Phys. Rev. D*, 79(11):112005, Jun 2009.
- [61] Peter Z. Skands. The Perugia Tunes. *arXiv:0905.3418v1 [hep-ph]*, 2009. Fermilab-CONF-09-113-T, MCNET-09-11, Presented at 1st International Workshop on Multiple Partonic Interactions at the LHC, Perugia, Italy, Oct 2008.
- [62] Michael G. Albrow et al. Tevatron-for-LHC Report of the QCD Working Group. *arXiv:hep-ph/0610012v1*, 2006. Fermilab-Conf-06-359.
- [63] W. H. Bell, D. Berge, J. A. Gray, R. Kwee, B. T. Martin, A. Messina, E. L. Nurse, A. J. Richards, and P. A. Steinberg. MBTS trigger efficiency for the minimum bias analysis using Inner Detector tracks from pp interactions at  $\sqrt{s} = 900$  GeV. Technical Report ATL-DAQ-INT-2010-009, CERN, Geneva, Sep 2010.
- [64] G. D’Agostini. A multidimensional unfolding method based on Bayes’ theorem. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 362(2-3):487 – 498, 1995.
- [65] H. Castrup. Estimating and Combining Uncertainties, May 2004. presented at the 8th Annual ITEA Instrumentation Workshop, Lancaster,

- CA, May 2004.
- [66] Vardan Khachatryan et al. Transverse momentum and pseudorapidity distributions of charged hadrons in pp collisions at  $\sqrt{s} = 0.9$  and 2.36 TeV. *JHEP*, 02:041, 2010.
- [67] C. Albajar et al. A study of the general characteristics of proton-antiproton collisions at  $\sqrt{s} = 0.2$  to 0.9 TeV. *Nuclear Physics B*, 335(2):261 – 287, 1990.
- [68] Professor Development Team. Professor. <http://projects.hepforge.org/professor>.
- [69] Andy Buckley, Hendrik Hoeth, Heiko Lacker, Holger Schulz, and Jan Eike von Seggern. Systematic event generator tuning for the LHC. *Eur. Phys. J. C*, 65:331–357, 2010.
- [70] The ATLAS collaboration. Charged-particle multiplicities in pp interactions measured with the ATLAS detector at the LHC. *Submitted to New J.Phys*, arXiv:1012.5104v2 [hep-ex], 2010.
- [71] The ROOT Team. ROOT. <http://root.cern.ch/drupal/>.

# List of figures

1.1. Fundamental QED vertex . . . . .	17
1.2. Fundamental QCD vertices showing (a) the quark gluon vertex, (b) the tri-gluon vertex and (c) the quad-gluon vertex . . . . .	18
1.3. Fundamental weak fermion W/Z interaction vertex . . . . .	19
1.4. Fundamental weak vertices showing (a) tri-WZ vertex, (b) quad-W vertex, (c) quad-WZ vertex, (d) W photon vertex, (e) quad-WZ photon vertex and (f) quad-W photon vertex . . . . .	22
1.5. Feynman diagrams for (a) single diffractive, (b) double diffractive, (c) central diffractive (double $\mathbb{P}$ ) and (d) non-diffractive events . . . . .	27
1.6. Figure showing rolled out projection of a detector with azimuthal angle vertically and a representation of the polar angle horizontally. The shaded regions show the particle occupancy in (a) single diffractive, (b) double diffractive, (c) central diffractive (double $\mathbb{P}$ ) and (d) non-diffractive events . . . . .	27
2.1. The LHC surface and underground layouts showing the old LEP buildings in grey and the new LHC buildings in red [9]. . . . .	30

---

2.2. The full LHC injector chain for both protons and heavy ions [11] . . . . .	31
2.3. A single LHC Dipole magnet (a) in cross-section and (b) showing the B-Field [12]. . . . .	33
2.4. Surface view of Point 1 showing the building covering the ATLAS shaft, SDX [14]. . . . .	34
2.5. The ATLAS Detector (Photo. #:0803012_01 ATLAS Experiment ©CERN 2011) [13] . . . . .	35
2.6. Graphical representation of the relation between $\eta$ and $\theta$ [15]	37
2.7. Cut away of ID showing the individual ID subsystems; the pixel, SCT and TRT layers [16] . . . . .	38
2.8. schematic of ID showing the individual ID subsystems; the pixel, SCT and TRT layers [16] . . . . .	39
2.9. Plots showing the ATLAS solenoid magnetic field intensity for components (a) $B_z$ and (b) $B_r$ as a function of longitu- dinal and radial displacement from the nominal interaction point, $z$ and $R$ respectively. [16] . . . . .	39
2.10. Three dimensional view of the complete pixel sub detector system. [16] . . . . .	40
2.11. Views of the ATLAS SCT in (a) perspective [18] and (b) Photo. of the SCT strip sub detector (Photo. #:0509006_03 ATLAS Experiment ©CERN 2011) [13] . . . . .	41
2.12. Views of the ATLAS TRT in (a) perspective and (b) R-z profile [17]. . . . .	44
2.13. Perspective view of the ATLAS calorimeter systems . . . . .	45

---

2.14. Photo. showing the accordion shaped electrodes and absorber plates (Photo. #:9308048_09_general ATLAS Experiment ©CERN 2011). [13] . . . . .	46
2.15. The ATLAS muon spectrometer system (Photo. #:0803017_01 ATLAS Experiment ©CERN 2011). [13] . . . . .	50
2.16. One of the two MBTS wheels mounted on the LAr end-cap calorimeter (white circle) [24]. . . . .	54
4.1. Diagram showing routes between event generator and data analysis with the full simulation on the left and Atlfast fast simulation on the right. Atlfast replaces the need for separate detector simulation and reconstruction code by simply smearing the physics input. . . . .	60
4.2. UML style diagram showing the inheritance ladder for new custom cells . . . . .	76
4.3. UML style diagram showing the key components of the Cell-Factory update . . . . .	78
4.4. UML style diagram showing the key components of the smearers update . . . . .	82
4.5. Rolled out Atlfast barrel calorimeter jet population density plots for (a) normal Atlfast calorimeter and (b) custom dead calorimeter. . . . .	86
4.6. Rolled out Atlfast barrel calorimeter jet population density plot for scaled asymmetric region . . . . .	86
4.7. Rolled out Atlfast barrel calorimeter jet population density plot showing a single hot cell of 50 GeV. . . . .	87

- 
- 4.8. Plot showing the  $p_T$  spectra of single diffractive events passing the triggers; L1\_MBTS-1 (black), L1\_MBTS-2 (red) and L1\_MBTS-1.1 (blue) for fully simulated trigger (closed) and the new fast trigger implementation (open). The inner box shows the tails on a logarithmic scale. . . . . 90
  
  - 5.1. Job Chain showing a complicated job structure with parallel (**P**) and sequential (**S**) chain elements/jobs (**C**) [37] . . . . . 94
  
  - 5.2. Example of ELog web interface . . . . . 96
  
  - 5.3. Example XML code for an Athena job showing all necessary and optional tags [37] . . . . . 98
  
  - 5.4. ATLAS RTT mainpage showing buttons for results from RTT FCT and TCT. . . . . 99
  
  - 5.5. RTT results page showing a blank workspace. . . . . 100
  
  - 5.6. RTT results page showing the results from the InDetPerformanceRTT package. . . . . 101
  
  - 5.7. RTT results page showing results from a job within the InDetPerformanceRTT package. . . . . 102
  
  - 5.8. Example of main DCube output webpage. . . . . 105
  
  - 5.9. Example of regression test histogram. . . . . 105
  
  
  - 6.1. Example signal Feynman diagrams for (a) the 2-jet +  $E_T^{miss}$  final state and (b) the 3-jet +  $E_T^{miss}$  final state. . . . . 112
  
  - 6.2. Diagram showing the  $E_T^{miss}$  vector balancing in the transverse plane the detectable matter in the event . . . . . 118



- 
- 6.3.  $E_T^{miss}$  plot for J6 QCD (red) and SUSY (blue) jets. . . . . 119
- 6.4. Diagram showing two jets and their azimuthal distance from the  $E_T^{miss}$  vector labelled  $\delta\phi_{1,2}$  . . . . . 120
- 6.5.  $\delta\phi_1$ - $\delta\phi_2$  plane showing (a) J6 QCD jets and (b) SUSY jets. . 120
- 6.6.  $\delta\phi_1$ - $\delta\phi_2$  plane showing (a) J6 QCD jets and (b) SUSY jets. The red lines show the effect of the  $R_{1,2}$  cuts . . . . . 121
- 6.7. Diagrams showing the orientation of jets in an event for (a) transverse sphericity of 0 and (b) transverse sphericity of 1. . 122
- 6.8. Area normalised plot of  $S_T$  for QCD jets (red) and SUSY jets (blue). . . . . 123
- 6.9.  $M_{eff}$  plots for (a) two hardest and (b) three hardest jets in QCD (red) and SUSY (blue) events . . . . . 124
- 6.10. Plots showing the trigger efficiency ratio of (leading jet  $p_T$  of events passing the trigger/jet  $p_T$ ) for level one triggers with threshold of (a) 35 GeV, (b) 45 GeV, (c) 60 GeV and (d) 80 GeV. . . . . 125
- 6.11.  $M_{eff}$  plots showing SUSY signal and combined SM background for (a) the 2-jet analysis and (b) the 3-jet analysis. The red line shows the signal separation cut. . . . . 131
- 6.12.  $E_T^{miss}$  plots showing SUSY signal and combined SM background for (a) the 2-jet analysis and (b) the 3-jet analysis. The red line shows the signal separation cut. . . . . 132

- 
- 6.13.  $M_{eff}$  plots showing SUSY signal stacked on top of the combined SM background for the (a) 2-jet and (b) 3-jet analysis. The individual components of the SM background are also superimposed for completeness. . . . . 134
- 6.14.  $E_T^{miss}$  plots showing SUSY signal stacked on top of the combined SM background for the (a) 2-jet and (b) 3-jet analysis. The individual components of the SM background are also superimposed for completeness. . . . . 135
- 6.15.  $M_{eff}$  plots from the latest Monte Carlo simulation paper [45], showing SUSY signal and combined SM background for both the (a) 2-jet and (b) 3-jet analysis. . . . . 137
- 6.16. SUSY signal and combined SM background plots for real ATLAS data [50] showing (a)  $M_{eff}$  plot with only the 2-jet analysis jet cuts, (b)  $E_T^{miss}$  plot with only the 2-jet analysis jet cuts, (c)  $M_{eff}$  plot with only the 3-jet analysis jet cuts and (b)  $E_T^{miss}$  plot with only the 3-jet analysis jet cuts. . . . 137
- 7.1. Plot showing the L1\_MBTS\_1 trigger efficiency as a function of  $n_{Sel}^{BS}$  [52]. . . . . 143
- 7.2. Trigger efficiency plots showing DD (blue), SD (red) and ND (black) samples on the same axes for comparison. The plots show the efficiency for the (a,b,c) L1\_MBTS\_1, (d,e,f) L1\_MBTS\_2 and (g,h,i) L1\_MBTS\_1\_1 as a function of (a,d,g)  $\eta$ , (b,e,h)  $N_{ch}$  and (c,f,i)  $p_T$ . . . . . 146

- 
- 7.3. Trigger efficiency plots showing DD (blue), SD (red) and ND (black) samples on the same axes for comparison. The plots show the efficiency for the fast simulation of (a,b,c) L1\_MBTS\_1, (d,e,f) L1\_MBTS\_2 and (g,h,i) L1\_MBTS\_1\_1 as a function of (a,d,g)  $\eta$ , (b,e,h)  $N_{ch}$  and (c,f,i)  $p_T$ . . . . . 148
- 7.4. Vertex reconstruction efficiency as a function of  $n_{Sel}^{BS}$ . [52] . . 149
- 7.5. Plots showing the track reconstruction efficiency as a function of (a)  $p_T$  and (b)  $\eta$ . . . . . 150
- 7.6. Comparison between data and MC models of charged particle multiplicities as a function of (a) pseudorapidity and (b)  $p_T$ . Also showing (c) the charged particle multiplicity per event and (d) the average  $p_T$  as a function of the number of charged particles in an event. Each of these plots are for events with  $n_{ch} \geq 1$  and charged particles within the kinematic range  $p_T > 500$  MeV and  $|\eta| < 2.5$ . . . . . 154
- 7.7. Comparison of the charged particle multiplicity's  $p_T$  spectrum between the ATLAS, CMS and UA1 data. . . . . 155
- 7.8. Plots from the Rivet MinBias analysis performed independently of Athena, showing (a)  $\eta$  distribution, (b)  $p_T$  distribution and (c) multiplicity distribution for primary charged particles. Also shown is a plot of (d) the mean  $p_T$  vs. charged particle multiplicity. . . . . 156

- 7.9. Plots from the Rivet MinBias analysis performed within Athena, showing (a)  $\eta$  distribution, (b)  $p_T$  distribution and (c) multiplicity distribution for primary charged particles. Also shown is a plot of (d) the mean  $p_T$  vs. charged particle multiplicity. . . . . 158
- 7.10. MBTS Threshold diagrams for Pythia 8 samples containing (a) standard mixture of elastic and in-elastic events, (b) just the non-diffractive events, (c) just the double diffractive events and (d) just the single diffractive events. . . . . 159
- 7.11. MBTS Threshold diagrams for (a) Pythia 8 and (b)(c)(d) Pythia 6 samples containing standard mixture of elastic and in-elastic events, with (b) ATLAS MC09 tuning, (c) DW tuning and (d) Perugia0 tuning. . . . . 161
- 7.12. Comparison of MBTS threshold plots between the (a,c,e) full trigger simulation and (b,d,f) fast trigger simulation. Shown is the hit MBTS counter multiplicity against counter  $p_T$  threshold for a (a,b) single, (c,d) double and (e,f) non-diffractive event sample. . . . . 162
- 7.13. Charged-particle multiplicities as a function of the pseudorapidity for events with  $n_{ch} \geq 1$ ,  $p_T > 500$  MeV and  $|\eta| < 2.5$  at  $\sqrt{s} =$  (a) 0.9 TeV, (b) 2.36 TeV and (c) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70]. 165

- 7.14. Charged-particle multiplicities as a function of the transverse momentum for events with  $n_{ch} \geq 1$ ,  $p_T > 500$  MeV and  $|\eta| < 2.5$  at  $\sqrt{s} =$  (a) 0.9 TeV, (b) 2.36 TeV and (c) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70]. . . . . 166
- 7.15. Charged-particle multiplicity distributions for events with  $n_{ch} \geq 1$ ,  $p_T > 500$  MeV and  $|\eta| < 2.5$  at  $\sqrt{s} =$ (a) 0.9 TeV, (b) 2.36 TeV and (c) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70]. . . . . 167

- 7.16. Average transverse momentum as a function of the number of charged particles in the event for events with  $n_{ch} \geq 1$ ,  $p_T > 500$  MeV and  $|\eta| < 2.5$  at  $\sqrt{s} =$  (a) 0.9 TeV and (b) 7 TeV. The dots represent the data and the curves the predictions from different MC models. The vertical bars represent the statistical uncertainties, while the shaded areas show statistical and systematic uncertainties added in quadrature. The bottom inserts show the ratio of the MC over the data. The values of the ratio histograms refer to the bin centroids [70]. . . . . 168

# List of tables

1.1. Classification system showing quantum numbers for (a) Leptons and (b) quarks excluding spin [1] . . . . .	14
1.2. Classification system for the gauge bosons showing the force which they mediate. . . . .	15
2.1. Granularity and pseudorapidity coverage of the ATLAS EM calorimeter [10]. . . . .	47
2.2. Granularity and pseudorapidity coverage of the ATLAS HAD calorimeter [10]. . . . .	48
2.3. Muon Spectrometer chamber technologies and their locations in $\eta$ within the ATLAS detector [22]. . . . .	52
4.1. Table showing the four packages that make up the Atlfast software listed in dependency order and separated into the test package and the three main packages . . . . .	60
4.2. The sequence of algorithms making up the standard Atlfast simulation program. . . . .	61
4.3. Contents of the GlobalEventData . . . . .	61

---

4.4. Acceptance and granularity of the three Atlfast calorimeter sections . . . . .	62
4.5. Parameterised resolution functions for electron smearing in low and high luminosity modes. . . . .	65
4.6. Parameterised resolution functions for photon smearing in low and high luminosity modes. . . . .	65
4.7. Parameterised resolution functions for jet smearing in low and high luminosity modes. . . . .	67
4.8. b-jet tagging efficiencies and mis-tagging inefficiencies for low and high luminosity modes averaged over $p_T$ [10]. . . . .	70
5.1. The current list of available RTT library tools . . . . .	93
6.1. mSUGRA SU3 benchmark point parameters . . . . .	114
6.2. Sparticle mass spectra for the SU3 mSUGRA benchmark point	114
6.3. QCD di-jet samples and their respective $p_T$ ranges. . . . .	115
6.4. Analyses samples and their respective cross-sections. . . . .	117
6.5. Preselection cuts for (a) jets, (b) electrons, (c) photons and (d) muons . . . . .	126
6.6. Event selection cuts for (a) two and (b) three jet analysis . .	129
6.7. Table showing the cuts used to separate the signal from the background. . . . .	130
6.8. Table of significances from the plots in Figure 6.11 and Figure 6.12. . . . .	133



---

7.1. Requirements for the three main MBTS related triggers. . .	144
7.2. Systematic uncertainties on the number of events, $N_{ev}$ and on the charged particle density $(1/N_{ev}) \cdot (dN_{ch}/d\eta)$ at $\eta =$ 0 which are anti-correlated. All other sources are assumed to be uncorrelated. . . . .	152