UNIVERSITÉ DE GENÈVE
Départment de physique nucléaire et corpusculaire FACULTÉ DES SCIENCES
Professeur T. Golling

Identification of b-jets and c-jets using Deep Neural Networks with the ATLAS Detector

The Development and Performance of a Family of DL1 High-level Flavour Tagging Algorithms

THÈSE

Présentée à la Faculté des sciences de l'Université de Genève
Pour obtenir le grade de Docteur ès sciences, mention physique

Par

Marie Christine Lanfermann

d'Allemagne

Thèse N^{o} 5330

GENÈVE
Atelier d'impression ReproMail
2019

UNIVERSITÉ DE GENÈVE

FACULTÉ DES SCIENCES

DOCTORAT ÈS SCIENCES, MENTION PHYSIQUE

Thèse de Madame Marie Christine LANFERMANN

intitulée :

«Identification of *b*-jets and *c*-jets Using Deep Neural Networks with the ATLAS Detector

The Development and Performance of a Family of DL1 High-level Flavour Tagging Algorithms»

La Faculté des sciences, sur le préavis de Monsieur T. GOLLING, professeur associé et directeur de thèse (Département de physique nucléaire et corpusculaire), Madame A. SFYRLA, professeure assistante (Département de physique nucléaire et corpusculaire) et Monsieur D. ROUSSEAU, professeur (Laboratoire de l'Accélérateur Linéaire, Université Paris-Sud, ORSAY, France), autorise l'impression de la présente thèse, sans exprimer d'opinion sur les propositions qui y sont énoncées.

Genève, le 10 avril 2019

Thèse - 5330 -

Le Doyen

N.B. - La thèse doit porter la déclaration précédente et remplir les conditions énumérées dans les "Informations relatives aux thèses de doctorat à l'Université de Genève".

“Be less curious about people and more curious about ideas.”

- Marie Skłodowska Curie

Contents

Total Word Count: 36985

Abstract

In this thesis, a new family of high-level jet flavour tagging algorithms called DL1 [1, 2] is presented. It is now established within the ATLAS collaboration [3] at the Large Hadron Collider at CERN to be applied to Run 2 pp collision data at $\sqrt{s}$ = 13 TeV. DL1 represents the first use of Deep Learning for ATLAS physics object reconstruction as well as the first major application of advanced deep neural networks within the collaboration. The determination of jets originating from heavy flavour quarks is used to probe the particle identity of particles created in the pp collisions. These heavy flavour quarks play a major role in searches for new physics and precision measurements.

The potential of Deep Learning in flavour tagging using inputs from lower-level algorithms has been investigated. A systematic grid search over architectures and the training hyperparameter space is presented. In this neural network approach, the training is performed using multiple output nodes, which is a naturally suited method for the task of jet flavour tagging. This also provides highly flexible tagging algorithms. The DL1 studies presented show that the obtained b- and c-jet tagging algorithms provide good discrimination against jets of other flavours considered in flavour tagging. Their performance for arbitrary background mixtures can be adjusted after the training according to the needs of the physics analysis. The resulting development and structure of DL1 as well as the architectures of the neural networks used in the tagging algorithms are described and a detailed set of performance plots is presented, obtained from simulated $t\bar{t}$ events at $\sqrt{s}$ = 13 TeV and corresponding to the data taking conditions during Run 2 where these tagging algorithms will be applied. Performance comparison plots between predictions from simulation and collision data as well as the final b-jet tagging scale factors of the calibration for physics analyses usage are provided and show excellent agreement.

The algorithms are not only well optimised but also generalise the learned jet topologies well to other event topologies. A fully fledged family of robust b- and c-jet tagging algorithms with a reduced amount of required person power is established and recommended within the ATLAS collaboration. Now that DL1 has been

established, it is expected to improve a wide range of physics analyses throughout the collaboration.

Résumé

Dans cette thèse, une nouvelle famille d'algorithmes d'étiquetage de la saveur des jets de haut niveau appelée DL1 [1, 2] est présentée. Il est maintenant établi dans le cadre de la collaboration ATLAS [3] au Grand collisionneur de hadrons au CERN pour être appliqué aux données de collision Run 2 pp à $\sqrt{s}$ = 13 TeV. DL1 représente la première utilisation de Deep Learning pour la reconstruction d'objets physiques dans ATLAS ainsi que la première application majeure de réseaux neuronaux profonds avancés dans le cadre de la collaboration. La détermination des jets provenant de quarks lourds est utilisée pour sonder l'identité des particules créées dans les collisions pp. Ces quarks à saveur forte jouent un rôle majeur dans la recherche de nouvelles propriétés physiques et de mesures de précision.

Le potentiel de l'apprentissage en profondeur dans l'étiquetage des saveurs à l'aide d'entrées provenant d'algorithmes de bas niveau a été étudié. Une grille de recherche systématique sur les architectures et l'espace hyperparamétrique d'entraînement est présentée. Dans cette approche de réseau neuronal, l'apprentissage est effectué à l'aide de nœuds de sortie multiples, ce qui est une méthode naturellement adaptée à la tâche d'étiquetage des saveurs des jets. Cela permet également d'obtenir des algorithmes de étiquetage très flexibles. Les études DL1 présentées montrent que les algorithmes de d'étiquetage de jet b et c obtenus permettent une bonne discrimination par rapport aux jets d'autres saveurs. Leur performance pour des mélanges de bruit de fond arbitraires peut être ajustée après l'entrainement en fonction des besoins de l'analyse physique. Le développement et la structure de DL1 ainsi que les architectures des réseaux neuronaux utilisés dans les algorithmes de étiquetage sont décrits et un ensemble détaillé de diagrammes de performance est présenté, obtenu à partir d'événements simulés $t\bar{t}$ à $\sqrt{s}$ = 13 TeV et correspondant aux conditions de prise de données pendant le Run 2 où ces algorithmes seront appliqués. Des graphiques de comparaison des performances entre les prédictions issues de la simulation et les données de collision sont fournis, ainsi que les facteurs d'échelle de l'étalonnage de d'étiquetage des jets b pour l'utilisation dans les analyses physiques. Tous montrent un excellent accord.

Les algorithmes sont non seulement bien optimisés, mais ils permettent aussi de bien généraliser les topologies de jets apprises à d'autres topologies d'événements. Une famille complète d'algorithmes robustes de d'étiquetage des jets b- et c-jet avec une intervention humaine réduite est établie et recommandée dans le cadre de la collaboration ATLAS. Maintenant que DL1 a été établi, on s'attend à ce qu'il améliore un large éventail d'analyses physiques tout au long de la collaboration. [1]

[1] Traduit avec Ref. [4].

Acknowledgements

Many thanks to my Doktorvater Tobias for giving me the freedom and time to develop DL1 as I envisioned it. Your guidance helped me to learn how to articulate my arguments in an appropriate manner in suiting with the collaboration and being patient about expectations on feedback of hard work. Thanks to Andrea for encouraging me to consider ideas closer to the ATLAS work flow mentality. Thanks to John for not getting tired pointing out all the split infinitives in the first draft of this thesis and encouraging me to plough through it.

Steven, thank you for your consistent support. You took time to listen, tried to understand and were a grounded presence when times got tough and shared honest happiness in times of success. Thanks to Valerio for offering perspectives and advice on running marathons when needed. Frank, thanks for always making time for a few words when our paths crossed and thereby reminding me of the Radboud HEP group atmosphere. Andi and Gordon, thank you for seeing people and your inspiring presence in the collaboration. Your example means a lot and will continue to inspire me.

A large number of people made the past few years good fun and much more than just a professional endeavour. Special thanks to those who could be counted on for barbecues, walks, bike rides, coffees or drinks and good social non-work related conversations after work. Thanks also to Jelena, Lidia, Evelyn and Nan for being great office mates and shaping such an enjoyable atmosphere.

Special thanks to my mother Margret for your support and love throughout my life. Thank you for encouraging my studies and letting me choose my own path.

Finally, my biggest thanks go to my best friend and partner Johnny. Without you I would not be the same person I am now. Words cannot express how thankful I am for your believe in me, your love, understanding, patience and humour. Thank you for making me truly happy.

Preface

The author sought to incorporate many of the newer developments in the field of machine learning into High Energy Physics. As a result of this, the author single handedly developed DL1 and introduced the application of deep neural networks for use on collision data within ATLAS. This includes everything from the structure involving the preprocessing of the inputs, the software package configuration involving well maintained and developed open source software, its offline framework, which includes the streamlining of grid searches. The author's contribution also includes the technical implementation of DL1 within the ATLAS collaboration framework ATHENA, for which the author co-designed and co-developed the python package LWTNN [5], which is now widely used. The implementation within ATHENA performed by the author includes the validation of this implementation.

Furthermore, the author performed the initial test studies on defining good design approaches and performed the grid searches and optimisations as well as performance tunings for the final discriminants for all mentioned DL1 versions. This includes work presented at EPS in 2017 [1] and in Ref. [2], where dedicated DL1 variants are compared to the MV2 high-level tagging algorithm baselines. The author wrote this PUB note with the intent of providing public DL1 documentation and promoted it up until the first iterations with the first reader before having to step down in order to start working on a physics analysis. In addition, the author promoted DL1 and provided detailed technical documentation for use within the ATLAS collaboration.

In addition to this, the author provided knowledge transfer to other members of the collaboration like in the jet-$E_{\mathrm{T}}^{\mathrm{miss}}$ performance group and in the Geneva group, with the infrastructure built for DL1 used as foundations for further work employed in the ATLAS collaboration. Furthermore, this knowledge transfer was propagated to other experiments within the DPNC at Geneva, and the knowledge has been successfully adapted to a variety of different applications within particle physics.

The author of this thesis contributed to selected aspects of the content of this thesis with the full picture given for completeness. Chapter 5 provides the reader

with context regarding the supervised classification methods relevant to the main topic of this thesis from a Data Science perspective. Here, the author designed and created the schematic overviews of boosted decision trees as well as neural network related principal components, architectures and items of interest. Chapters 7 and 8 present the authors work on designing and optimising the presented DL1 family members, which represent the first use of Deep Learning within ATLAS object reconstruction. The data-MC comparisons and calibrations, with which the author was not involved, are presented to demonstrate the overall validity of the DL1 tagging performance for use in physics analyses on pp collision data recorded by the ATLAS experiment.

Finally, the author also worked on tuning the DL1 c-jet tagging algorithms for the $VH \rightarrow c\bar{c}$ search using the same data handling approach as for the construction of DL1. However, due to the timeline of the analysis these studies could not be drawn to a conclusion on time to be included in this thesis. Nonetheless, meaningful contributions regarding the tuning of DL1 were provided by the author.

1. Introduction

Searches for new physics in high-energy physics rely on our ability to know exactly what was created in collisions and how these elementary particles interact with each other. A large number of rare and interesting processes result in the production of heavy flavour quarks, namely b- and c-quarks. Top quarks, the heaviest known quarks decay before being able to form a bound state and therefore are probed through the presence of b-quarks. In addition, the Higgs boson primarily decays into a $b\bar{b}$ pair.

The ATLAS detector was designed as a general-purpose detector, which is required to allow many different precision measurements and searches to be performed to measure known processes and to cover a large spectrum of potential new physics. Due to realistic limitations mainly due to real physics constraints like increased material density close to the collisions resulting in more scattering and worse resolution in the calorimeter measurements as well as the material costs needed for detection and support structure for a large coverage of measurements, compromises had to be made when designing the ATLAS detector. The ATLAS detector is not designed to provide particle identification for these heavy flavour quarks similar to the way LHCb is designed to, which has dedicated particle identification systems and extremely high tracking resolution due to the VELO [6]. Due to this people within the ATLAS collaboration have to be creative about alternative ways to probe interesting high-energy events at the Large Hadron Collider.

One way to probe for the interesting events is by considering jets. Jets are reconstructed objects, which high energy particles form due to underlying physics principles and interaction with the detector material. By being able to determine the origin of a jet, one can probe the events containing heavy flavour quarks and by doing so identify collisions of interest in searches for new physics, which will help to broaden the general understanding of fundamental particle physics. This means that all top analyses heavily rely on the identification of jets originating from a b-quark. The ATLAS observation of $H \to b\bar{b}$ decays in 2018 [7] relied essentially on the tools to identify the origin of a jet, in particular if the originating particle

is a b-quark. This underlies the extreme importance of the ATLAS collaboration to provide its physics analyses with the best possible tools to identify the originating particle of a jet.

Naturally, the classification of the jet origin is a multi-class problem, as jets form from many different particles, mostly light-flavour, c- and b-quarks. Neural Networks are known to work exceptionally well for images on predicting the classification of a visualised object based on pixel information by adapting a multi-class output. Classifying the origin of jets is not that different from a conceptual point of view. Therefore, a Neural Network provides a perfect solution to the task of jet origin classification. The main topic of this thesis is the introduction and establishment of a new family of multi-class jet origin identification algorithms called DL1 [1, 2]. A DL1 algorithm provides a single trained Neural Network to identify the origin of jets as a tool to assist in probing collisions at the Large Hadron Collider for new physics and precision measurements. DL1 is to be used on pp collisions data collected during Run 2 recorded by the ATLAS collaboration at the Large Hadron Collider from 2015 to 2018 and beyond. It is constructed on simulated pp collision data using the official full ATLAS simulation chain using supervised learning and calibrated using recorded collision data.

This thesis is structured as follows. In Chapter 2 the particle physics theory is explained. This is followed by an overview of the experimental set-up of the Large Hadron Collider and the ATLAS detector in Chapter 3. From the recorded data from the detector measurements physics objects are reconstructed, which is described in Chapter 4. The principles for supervised learning which are relevant in the context of this thesis are outlined in Chapter 5. The context for jet flavour identification in the ATLAS collaboration is presented in Chapter 6, where the low- and high-level jet flavour tagging algorithms are introduced. In Chapter 7 the design of a new family of jet flavour identification algorithms is described in detail and it is described how the current variants of DL1 flavour tagging algorithms are optimised. The performance of the currently adapted variants to be used for physics analyses is discussed in Chapter 8. Finally, the conclusions from this body of work are presented in Chapter 9 and an outlook on ideas for possible future developments for the DL1 flavour tagging algorithm family is presented.

2. The Framework of Particle Physics

Contents

Particle physics aims to provide a mathematical description of nature. This includes constructing this description on the understanding of fundamental laws of nature and explaining phenomena in the universe from the largest to the smallest scales. Theoretical models are constructed to describe nature with the highest possible precision in order to describe and predict the behaviour of the smallest particles in our universe and their interactions among themselves. Particle physics is so far best described by the Standard Model (SM), a theoretical model which comprises the elementary particles observed in nature and their interactions in three of the four fundamental forces of nature. The scope of this chapter is to provide the foundation of particle physics theory for this thesis. First, the SM is described with particular emphasis on the decays of heavy elementary particles. The classification of the primary elementary particle of these decays and the resulting resulting decay cascades is the subject of this thesis. Alongside the introduction of the elementary particles and their properties, the generation of their mass is discussed in this chapter. A more detailed introduction can be found in Ref. [8, 9].

However, the full picture of physics phenomena is still incomplete when considering only the SM, motivating searches for physics beyond the SM. In order to test the predictions of the SM or look for deviations in comparisons to data, simulated predictions are required. For these simulations the Monte Carlo (MC) method is used to simulate elementary particle interactions. An overview of the technique used for these simulations is also provided. The simulations provide predictions on processes using the same physics objects as expected in reconstructed collision events and is as close as possible to the recorded data of known processes.

Natural units are used throughout this thesis, setting the reduced Planck constant ($\hbar$) and the speed of light in a vacuum (c) to unity, defining $\hbar = c = 1$, unless stated otherwise.

2.1 The Standard Model

The SM is a renormalisable, Lorentz invariant, non-abelian gauge theory. It postulates the existence of few a elementary particles as its building blocks. These particles are considered point-like without internal substructure or excited states.

An overview of all elementary particles in the SM is shown in Figure 2.1 with particles classified by their intrinsic properties, called quantum numbers. These quantum numbers also specify their properties and their potential interactions with other elementary particles. These quantum numbers and properties are the spin, the electrical charge, the colour charge, the rest mass and the weak isospin (I_3), the hypercharge (Y). In addition to the particles shown in Figure 2.1, for each of the elementary particles described above there exists an antiparticle which has equal values for its quantum numbers except for the electrical charge, which is of opposite sign.

Individual elementary particles are classified by their spin into two major groups. Elementary particles with 1/2-integer spin are the main constituents of matter and classified as (matter) fermions. Fermions are gauge eigenstates of Yukawa fields. The fermions of the SM are divided further into quarks and leptons based on their electrical charge. Leptons carry integer electrical charge (Q = 0, $\pm$1), while quarks carry a fractional value (Q = $\pm$1/3, $\pm$2/3). There are three generations of each quarks and leptons. The first generation consists of the lightest quarks, the up (u) and down (d) quarks, the second of the charm (c) and strange (s) quark and the third generation comprises the top (t) and bottom (b) quarks. Electrons, muons and taus together with their neutrino counterparts make up the leptons. The remaining elementary particles, which have integer spin, are classified as bosons, with those which act as

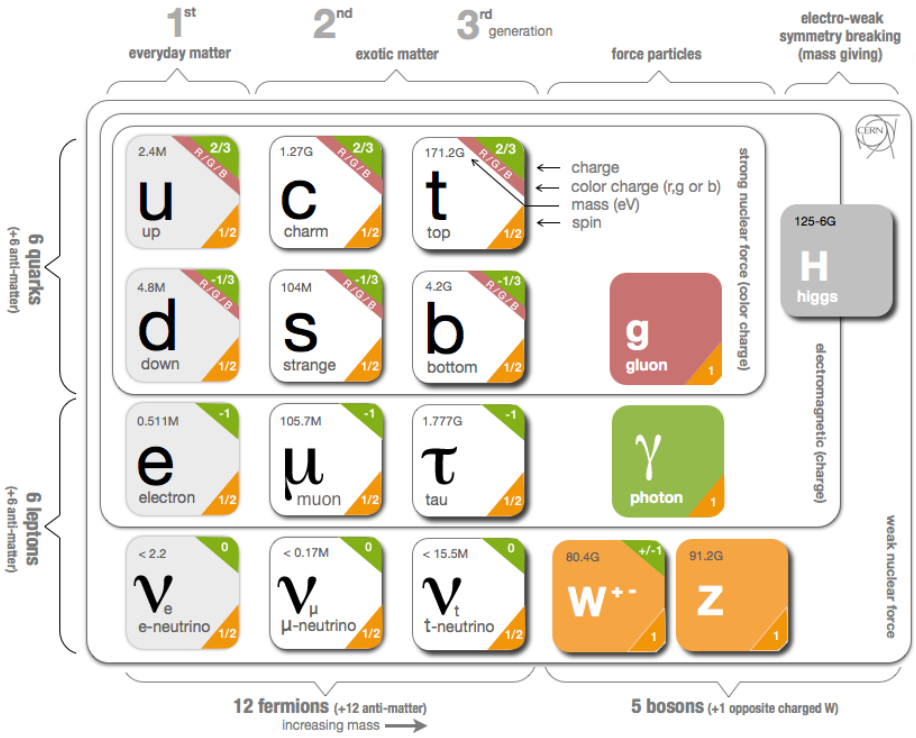


Figure 2.1: A schematic overview over the elementary particles of the SM. The particles are categorised into bosons and fermions according to their intrinsic spin. The fermions are further separated into quarks and leptons depending on their integer or non-integer charge. The lightest of either up- or down-type quarks and leptons are referred to as first generation and with increasing mass as second and then third generation. The overview has been adapted from Ref. [10].

mediators in particle interactions classed to as gauge bosons. Bosons are the gauge eigenstates of Dirac fields. The bosons of the SM are the gluons, the $W^{\pm}$ and Z^0, the photon (γ) and the Higgs boson (H). From them, the gluons, the $W^{\pm}$ and Z^0 and the photon (γ) are all gauge bosons and mediate one of three of the fundamental forces. All of them are of spin-1 which makes them vector bosons. However, in contrast to the other bosons, the Higgs boson carries spin-0 and is neither a gauge boson nor a vector boson. It is instead classified as a scalar boson.

From a Quantum Field Theory (QFT) perspective they are the gauge eigenstates of quantum fields. The theory unifies the strong, weak and electromagnetic forces. Interactions are associated with the exchange of elementary particles and the conservation of quantum numbers at interaction vertices. Gravity, the remaining fundamental force, is not included in this model.

Processes in particle physics are most precisely described using the respective Matrix Element which describes the state. Visualisations via Feynman diagrams represent the essential information content in elementary particle processes and aid faster perception for which they are widely used in particle physics. Each type of elementary particle is represented by a different line with arrows indicating the particle flow direction in time. Following common conventions, time progresses forward from left to right in the Feynman diagrams throughout this thesis. Fermions are represented by solid lines, scalars by dashed lines, abelian gauge bosons by wavy lines and non-abelian gauge bosons by curly lines. Dots indicate the interaction vertex of elementary particles where the electrical charge as well as energy and momentum of ingoing and outgoing particles are conserved.

2.1.1 Symmetries in the Standard Model

The SM of particle physics is an empirically driven theory which relies on the experimental determination of many of its particles properties. Still it remains based upon mathematical concepts assuming fundamental symmetries are the theoretical foundation of nature. The quantum numbers and properties of the elementary particles result from these underlying symmetries. Each of the interactions described encompassed in the SM is described using a fundamental symmetry group represented by Lie algebra terms and based on the underlying assumption of gauge invariance. Using Lie algebra, the SM is represented by the gauge symmetry group $SU(3)_C \otimes SU(2)_L \otimes U(1)_Y$ [11, 12]. The subscripts for each individual symmetry group indicate the conserved quantum numbers in the process described by this Lie group. According to the Emmy Noether theorem [13], each symmetry results in a conserved quantity in interactions. This too is also valid for these gauge symmetry

groups, with the nature of the symmetry related to the physical quantity.

2.1.2 Electroweak Interactions

The Electroweak (EW) force is one unified force at an energy of a unification of the weak and electromagnetic forces at high momentum transfers but manifests itself as two separate instances at lower momentum transfers. Therefore, before discussing the unification, both instances the electromagnetic and weak forces are described first before tying them together in the scheme of the Glashow-Weinberg-Salam (GWS) theory [14] which first proposed this unification.

Electromagnetic interactions are described by Quantumelectrodynamics (QED), a gauge field theory of phase transformations. These transformations correspond to the symmetry group $U(1)_Q$. Due to this symmetry, the total electrical charge of incoming and outgoing particles is conserved in electromagnetic interactions. In order for particles to interact via the electromagnetic force, the particles have to carry a non-zero quantum number electric charge (Q), which results in Q being a conserved quantum number in all interactions. This includes all quarks, charged leptons as well as the $W^{\pm}$ gauge bosons.

The mediator of the electromagnetic interaction is the photon (γ), which is massless. The photon itself does not carry electrical charge which prevents self-couplings. Since the photon has zero mass, it has infinite range. However, the strength of the interaction decreases with $1/r^2$, where r is the distance between the interacting particles. The interaction vertex of QED is shown by the Feynman diagrams in Figure 2.2. Note, however that this is not a physical process, as in this instance the total momentum is not conserved since the photon is massless.

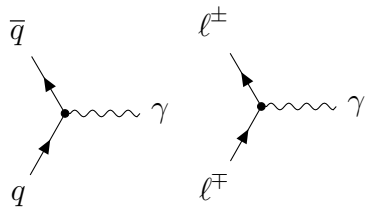
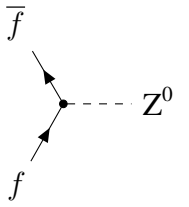


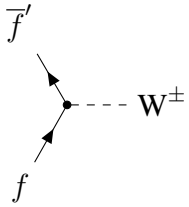
Figure 2.2: Feynman diagram for a leading order electromagnetic interaction. Fermion annihilation via incoming fermion and its anti-particle in the interaction vertex leads to the creation of a photon.

The weak interaction is a rotation in isospin space, represented by the chiral symmetry group $SU(2)_L$. The chirality of a particle is an intrinsic property of a particle and defined by the behaviour of the particle under Poincaré transformations. A particle can either have left- or right handed chirality. It is observed that the

weak interaction only acts on particles of left-handed chirality or antiparticles of right-handed chirality, resulting in the representation of the interaction by a chiral symmetry group. The intermediate vector bosons $W^{\pm}$ and Z^0 are predicted by the gauge theory description of the weak interaction, where they are the forces mediators with a conserved quantity of I_3. The basic interaction vertices are detailed in Figure 2.3 and the self-couplings of the mediators are shown in Figure 2.4.



(a) Neutral Current



(b) Charged Current

Figure 2.3: Feynman diagram for a weak interaction vertex. Here, fermion annihilation leads to the emission of a vector boson from the interaction vertex. Since the Z^0 boson does not carry charge, this is referred to as neutral weak interaction or weak neutral current (NC). An example is given in Figure 2.3a. The $W^{\pm}$ however carries positive/negative charge and therefore is referred to as weak charged (positive/negative) interaction or weak charged (positive/negative) current as shown in Figure 2.3b.

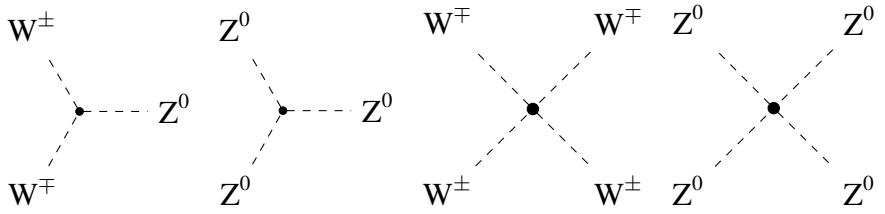


Figure 2.4: Feynman diagrams of self-couplings of vector bosons $W^{\pm}$ and Z^0.

The weak interaction only couples to left-handed particles and is divided into neutral current interaction or charged current interactions. The separation, which is shown by Figures 2.3a and 2.3b, is due to the force carriers charge with the Z^0

being charge neutral and the $W^{\pm}$ being electrically charged. The phrasing refers to the propagation of electrical charge in the gauge boson propagation.

Quark generation changing weak decays suggest the conversion of quark flavour. At least three generations are required for the theory to explain the observations of quark flavour changing currents in charged current interactions. A differentiation is required between observed quarks, referred to as weak eigenstates in weak interactions, and the physical quarks d, s and b, which are generally referred to as mass eigenstates. It has been experimentally proven that in weak interactions we observe the mixing of mass eigenstates via linear combinations. The linear mixing is described in Equation 2.1. The transition magnitudes $|V_{fi}|$, where i denotes the up-type quark flavour and f the final down-type quark flavour, are determined experimentally and make up the elements of the Cabibbo-Kobayashi-Maskawa matrix (V_{CKM}) [15] as shown in Equation 2.1. It is only due to the short range of the weak force, that the quantum number is generally assumed to be approximately conserved.

$$\begin{bmatrix} d' \\ s' \\ b' \end{bmatrix} \equiv V_{CKM} \begin{bmatrix} d \\ s \\ b \end{bmatrix} \equiv \begin{bmatrix} |V_{ud}| & |V_{us}| & |V_{ub}| \\ |V_{cd}| & |V_{cs}| & |V_{cb}| \\ |V_{td}| & |V_{ts}| & |V_{tb}| \end{bmatrix} \begin{bmatrix} d \\ s \\ b \end{bmatrix} \tag{2.1}$$

The GWS model proposes the unification of the electromagnetic and the weak forces by postulating that their gauge bosons are different manifestations of the same force, named the EW force. The weakness of the weak coupling in comparison to the electromagnetic coupling is attributed to the large masses of the mediating vector bosons.In the unified EW group the conserved quantity for $U(1)_Y$ is Y, which relates linearly to the weak isospin and the electrical charge as shown in

$$Q = I_3 + \frac{Y}{2}. \tag{2.2}$$

Due to this relation, the symmetry condenses down to $SU(2)_L \otimes U(1)_Y$ in the scheme of the EW unification, with $U(1)$ operating on the weak hypercharge (Y) and $SU(2)$ on the third component of the weak isospin (I_3).

2.1.3 Quantum Chromodynamics

In QFT, the strong interaction is described by Quantum Chromodynamics (QCD). It is represented by the non-Abelian colour symmetry group $SU(3)_C$, which reflects the symmetry of the quantum number colour (C). This quantum number is not referred to as a number but rather a property, expressed as being red (r), green (g) or

blue (b). The colour naming convention is based on optics arguments. Similar to optics where the mixing of specific colours leads to white, it is also the case regarding the quantum number colour that specific colour combinations lead to colourless particles. Analogous to the electric charge, the total colour is globally conserved in all elementary particle interactions. QCD is invariant under rotations in the isospin space, an internal symmetry, which is only concerned with the relation between elementary particles. It follows therefore that the conserved quantity in QCD interactions is the quantum number colour.

The associated interacting vector bosons which act as force mediator are the gluons. Gluons themselves carry colour, namely one unit of colour (r/g/b) and one anticolour ($\bar{\text{r}}/\bar{\text{g}}/\bar{\text{b}}$), and couple to any elementary particle which carries colour, quarks as well as other gluons. Since quarks and gluons are the constituents of protons, together they are referred to as partons. When interacting with colour carrying elementary particles, these interactions induce a colour change in the particles they interact with and can be visualised as a flow of colour in Feynman diagrams. The colour change depends on the colour state of the gluon. From the $SU(3)_C$ group it can be derived that there is a colour octet and one colour singlet state of which there exist eight linearly independent colour states. Each of these linearly independent colour states of a gluon can be expressed via the Gell-Mann matrices. Similar to the gauge bosons of the weak force, the gluons also exhibit self interactions. The QCD interaction vertices are shown by the Feynman diagrams in Figure 2.5.

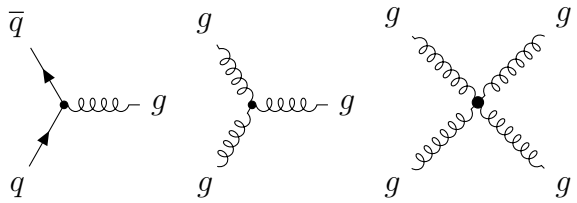


Figure 2.5: Feynman diagrams for the strong interaction (QCD).

Naturally occurring free particles are colourless. This principle is known as QCD confinement and as a logical consequence it follows that quarks are not observed in free states. It can be concluded from this that composite particles, called hadrons, consisting of bound states of multiple quarks, carry colours which in total are colour neutral. The most common examples of hadrons are mesons or baryons which are composed of two or three quarks respectively When separating the constituents of such a particle which is composed of quarks, an $q\bar{q}$ pair is created from the vacuum which create bound states with the original constituents. Those bound states each remain colourless.

The exhibition of self interaction introduces gluon loops. Due to this, the coupling strength of QCD vertices is not only influenced by positive contributions from quark polarisation but also negative contributions from gluon polarisation which decrease the coupling strength. In the determination of the coupling strength of QCD the parameter a weights both contributions against each other and provides a sign to the coupling strength, determining whether the force increases or decreases over short distances. The determination of this parameter is given by

$$a = 2f - 11n,$$

where the contributions from both quark polarisation and gluon polarisation are related to the number of available respective particles of interest. The value of f refers to the number of quark flavours which is relevant for contribution from quark polarisation. The contribution for gluon polarisation is accounted for by the value of n, which refers to the number of colours. With six quark flavours and three colours, the value of a is -21 and therefore the effective QCD couplings therefore increase over short distances and therefore for interactions of higher momentum transfer. This would meant that an infinite amount of energy is required to pull a quark out of a composite particle. However, this is in disagreement with observation, from which is clear that this task is achievable requiring finite energy, namely only about the energy to create the $q\bar{q}$ pair.

Various screening and anti-screening effects compete depending on distances. Anti-screening effects of higher order couplings involving gluons cause the coupling strength of the strong interaction to be dependent on the distance between the interacting particles. This behaviour is known as asymptotic freedom. As a result, the interaction coupling constant is better described to be a *running* coupling constant regarding distance and energy transfer scales. As a consequence of asymptotic freedom, gluons couple weakly at high energies and short distances but stronger at lower energies and larger distances. It is therefore possible to describe and treat coloured particles independently at high energies and small distances. Doing so makes it possible to use perturbation theory to describe interactions of individual quarks. This in turn simplifies the calculation of a quantitative cross section in hadronic interactions.

2.1.4 The Hadronic Model

Particles carrying the quantum number colour are not observed in their free state but only composite bound states after undergoing the process of hadronisation. Hadronisation occurs at all energies and effectively includes the creation of $q\bar{q}$ pairs from

the vacuum which then bound to the original unbound quarks to form colourless bound states. The quarks are then held together by the strong force in bound states as hadrons. In high energy particle collisions, hadronisation and radiation transform out of the original single quark a cone-shaped cascade of particles.

However, there is one quark that does not undergo hadronisation. Due to its high mass, the lifetime of the top quark is shorter than the time required for hadronisation to take place. This results in its decay before it can form a free isolated bound state. The decay probability of the top quark, which is referred to as its branching

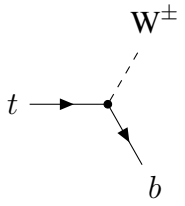


Figure 2.6: The Feynman diagram of the dominant decay mode of the top quark.

ratio (BR), to the b-quark is close to unity with the other BR being negligible and it can be assumed that in most cases, the top quark decays into a b-quark and $W^{\pm}$ vector boson as shown in Figure 2.6. The $W^{\pm}$ particle then subsequently decays either hadronically or leptonically resulting in different final states.

While top quarks decay before hadronisation can occur, this is not the case for the b-quarks. Although b-quarks are approximately 41 times lighter compared to the top quark, they are still relatively heavy compared to the other quarks - three times heavier then the next heaviest quark (c-quark) and about 43 times heavier than the next-to-next heaviest quark (s quark). This leads to a few very unique consequences which are used in the classification of the signatures originating from hadrons containing b-quarks. Compared to hadrons formed from the other quarks, b-hadrons have a relatively long lifetime ($\tau \simeq 455$ μm [15]) as well as harder fragmentation, which leads to a higher number of decay products and a substantial leptonic BR. Therefore, the identification of hadrons containing b-quarks plus the $W^{\pm}$ allows experimental physicists to spot top quarks and thus provide a window into the properties of a bare top quark.

2.1.5 Mass Generation and the Brout-Englert-Higgs Mechanism

The above description of the SM only allows for mediators for gauge invariance. Incorporating mass terms would break essential aspects of the theory. For the above theory to preserve gauge invariance, all gauge bosons should be massless.

However, the $W^{\pm}$ and Z^0 bosons are observed to be massive in nature, which would break gauge invariance, which is one of the underlying principles of the SM. Therefore, in order to preserve gauge invariance, the overall symmetry needs to be broken in the EW sector.

This means that the symmetry is not apparent in the ground state, i.e. the vacuum. Instead it can be chosen from a degenerate set of ground states. This ability to choose the physical vacuum breaks the symmetry. The gauge boson thereby acquires an an additional longitudinal polarisation, extending its previous two degrees to three, which allows it to be massive as only massive particles can have longitudinal polarisation.

2.1.6 Successes of the Standard Model

Over the past decades it has been found that the predictions by the SM match observations in data astonishingly well [15]. One example is given by the anomalous magnetic dipole moment of the muon, which is measured up to a precision of parts per million. The Muon g-2 experiment [16] at Fermilab is testing the precision of this value further to a precision of 0.14 part per million.

One of the major successes of the SM was the prediction of the third generation quarks before the discovery of the b-quark.Studies on meson mixing, the observation of suppressed K_L decays into either two neutral or opposite charged pions, lead to the conclusion that interactions are not symmetric under CP transformation, which is known as CP symmetry, and therefore that the $V_{CKM} \notin \Re$. It can therefore be concluded that the SM involves at least three generations of quarks. The b-quark was discovered via the discovery of the Upsilon (Υ) meson, a $b\bar{b}$ bound state. This hinted at the existence of the top quark, but its experimentally determined mass was much higher than anticipated, making its observation difficult. Perturbation theory and higher order corrections together with precise measurements of SM quantities allow theorists to predict the top quark mass precisely using the massive gauge bosons of the weak interaction. It was subsequently observed about twenty years after its postulation [17] at the Tevatron. Also the Higgs boson was fundamentally required in the SM and observed only in 2012 [18] at CERN.

2.1.7 Shortcomings of the Standard Model

Despite its successes there are however aspects in particle physics which the SM fails to address. For example, the SM does not provide a valid candidate to account for Dark Matter (DM) or dark energy and therefore misses to account for essential

aspects of nature, which are observed in cosmological observations of the cosmic microwave background and galaxy evolution.

One of the major failings of the SM is the fact that it is not able to incorporate gravity in the model. It can be considered a coherent theory of the forces up to a certain strength when dealing with elementary particles as gravity can be neglected at subatomic distances. However, this incompleteness does not fit with the image of nature following a coherent theory in the description of all elementary particle interactions.

Furthermore, it would be mathematically desirable to have an extension to the EW unification which unifies all three forces of the SM into one single force and one single common underlying symmetry. However, even when expanding the interaction energy further to higher energy scales, current predictions show that the coupling strengths do not converge towards the same value. There is, however, no necessary reason to assume that nature has to behave like this.Similarly, the observed mass of the SM Higgs boson is much smaller than expected. Only very careful fine-tuning of the parameters in the SM would lead to the loop corrections necessary to account for the observed mass. Including these complicated loop corrections is not straightforward. This issue is called the naturalness problem, and seeks to describe nature in a concise way without the need for complicated loop corrections. Again, this aspect is only motivated by the desire for a beautified theory.

In addition, similar to the observed mixing of flavours in the quark sector, neutrinos are also observed to oscillate between different flavours. This is currently not described by the SM, where they are often assumed massless. In order to accommodate this mixing, neutrinos are required to have mass.

2.2 Beyond the Standard Model

The failings of the SM suggest that there is more physics Beyond the Standard Model (BSM) theory required to describe nature. One of the theories is that the SM is just a low-energy manifestation of an underlying Grand Unified Theory (GUT), for which all coupling constants converge towards a common value at high enough energy. A common prediction for these GUT is that the proton may be unstable. However, so far there is no experimental evidence to support such theories.

Supersymmetry (SUSY) models build upon the underlying symmetries in the SM. In SUSY, additional underlying symmetries between bosons and fermions are assumed, which result in the prediction of one or more supersymmetric partners for each elementary particle of the SM. These are, for example, called squarks ($\widetilde{q}$) in the

case of the supersymmetric partner group of the quarks. In this schema the sbottom particle ($\widetilde{\mathrm{b}}$) is the supersymmetric elementary particle partner of the b in a minimal supersymmetric standard model theory. These theories are referred to as minimal because they only predict one supersymmetric partner per elementary particle. A large spectrum of SUSY models are possible. Individual SUSY theories vary in the number of predicted additional elementary particles, as well as their properties and parameters. Different SUSY models provide potential solutions to various shortcomings of the SM. For example, the naturalness problem could solved by introducing Feynman diagram loop corrections from the supersymmetric particles to the SM Higgs boson mass which would be reduced to a much lower scale than its true value. A much higher true value of the Higgs mass would then make the whole theory renormalisable. In addition, many of the SUSY models predict elementary particles which would be DM candidates. SUSY is also an important component of the fundamental assumptions of many string theory models, which attempt to unify gravity into particle physics.

Other exotic BSM theories predict additional versions of the SM particles with the same quantum numbers as their SM counterpart but larger masses. One example would be the possible existence of the Z' and W' heavy gauge bosons. Their decays would result in charge asymmetries which can be detected via determination of the lepton charge at high-p_{T}.

However, so far there has been little evidence from either direct or indirect searches for any of the fundamental particle physics theories beyond the SM. Similarly, there are also no observations for particles which are not predicted by the SM. It can therefore be concluded that no extension of the SM provides better predictions than the SM on its own.

2.3 Simulation of Interactions

In a pp collision event the partons interact with each other. If two partons interact and it involves a high p_{T} transfer, it is known as a hard scatter. The hard scatter is the kind of interaction where interesting physics happens. While it is more likely for two partons of the proton constituents to interact, it can happen that more than two partons interact as the protons collide, which is referred to as Multi-Parton Interactions (MPI). The partonic final states of pp collisions use Matrix Element (ME) for the calculation of the hard scatter in the MC method. Proton Synchrotron (PS) modelling is used for the simulation of the ingoing and outgoing partons of the hard scatter. The interactions of the partons from the beam particles is accompanied by the emission of low-momentum gluons, referred to as soft radiation, in the initial and final state of the hard scatter, known as Initial-State (QCD) Radiation (ISR) and Final-State (QCD) Radiation (FSR). This radiation and the presence of MPI lead to additional activity in the event, which is referred to as the Underlying Event (UE). The outgoing partons of the hard scatter create a Parton Shower (PS). The outgoing partons as well as the PS constituents both hadronise to build colourless bound states due to colour confinement. Of primary interest are the initial hard scatter and the associated PS.

Simulated high-energy pp events are generated to allow the comparison of the collision data with physics knowledge and theories. The simulation is done using general-purpose MC event generators [20] in the QCD factorisation model. The factorisation model divides the simulation into steps with energy levels of the momentum transfer of the particle interaction interaction separating them at the factorisation scale μ_F. Individual steps in the event simulation using the factorisation model are either based on first principles and can be calculated exactly or are based on models whose parameters are empirically determined. The components in the simulation of an event are shown in Figure 2.7, where the different steps involving particle interactions using the particle notation convention from Feynman diagrams are each represented by different colours. In addition, for a realistic representation of known physics knowledge, the simulation is also interfaced with detector specific simulation software. This is done to include the interactions with the detector material and signal digitisation as accurately as possible and how the physics objects are recorded. This is necessary for the recorded signals to be as close as possible to the recorded collision data.

The processes of interest are the high-momentum interactions like the initial parton interaction in an collision event, where the highest energy scales of the event

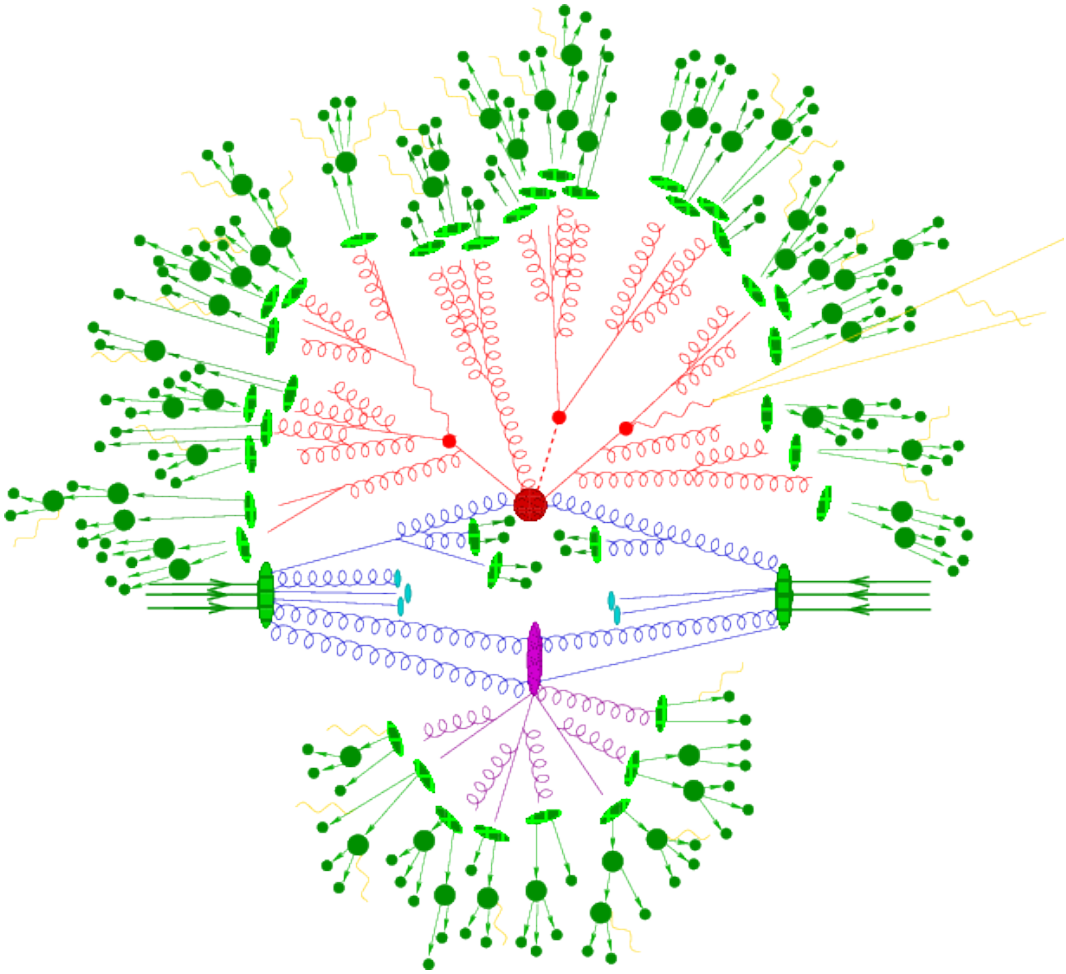


Figure 2.7: Schematic overview of the processes involved in simulating a pp scattering event at high energy [19]. The hard scatter is shown in red and multi-parton interactions are shown in purple. Initial- and final-state radiation as well as PS simulation processes are shown in blue. Hadronisation processes are shown in green. Gluon radiation is shown in yellow.

are involved, is referred to as the hard scatter. The process is calculated up to a fixed order in QCD and QED, which requires the ME, Parton Density Function (PDF) and the phase space of the interaction. Higher order QCD calculations of the ME of the process increase the precision but also result in higher computational costs.

Of less interest but an important aspect to consider as a source of noise are additional partons, which are not produces in the hard scatter and lead to additional activity in the event. They originate from scattered, annihilated and created partons in the collision event which didn't partake in the hard scatter of interest, the MPI, ISR, FSR as well as any soft radiation along the decay and propagation chains of the partons. They are known as the UE and can lead to a higher jet multiplicity in the event.

Parton Distribution Functions

As protons are composite particles of individual parton constituents, only individual partons interact with each other in collisions, an experimentally determined model is required to describe the probability for any given parton to carry a certain fraction of the total momentum of the proton. A model, which provides these probabilities, is called a PDF. These models are independent of the colliding particles and are determined through Deep Inelastic (lepton) Scattering (DIS). As well as being used to determine the energy distributions of the incoming partons in the hard scatter, the PDF is used in the PS as well as for the simulation of the MPI.

An essential choice for the representation of physics objects in the simulated data is the choice of flavour scheme for the PDF, which can either be a three flavour (3F), four flavour (4F) or five flavour (5F) scheme. The 5F scheme takes b-quarks into account as partons which can be found in the initial high-energetic protons. However, they are treated as being massless and therefore mass effects are not included. This aspect is included in the 4F scheme but there b-quarks are not treated as constituents which can be found in the high-energetic protons and therefore will not participate in the initial hard scatter. When using the PDF with the 4F scheme for the proton, b-quarks can only occur from decays or gluon splitting, but they can have mass. The 3F scheme also excludes c-quarks as constituents and particles involved in the hard scatter.

Parton Shower and Hadronisation

The partons from the hard scatter evolve in a succession of emissions of ingoing and outgoing partons. The momentum is conserved in this evolution and transferred from the initial momentum of partons after the hard scatter to final-state particles

with energies of about 1 GeV, which matches confinement of the partons. The final states of the PS consist of a bunch of colinear partons, which are naturally arranged in a cone-like shape, and as such are referred to as jets. The final states of the PS can overlap with the same final states from the ME from the hard scatter if the calculation is done at Next-to-Leading Order (NLO). In order to achieve the best simulation for multi-parton states, this overlap needs to be taken into account when interfacing PS and ME calculations. This is done either via the overlap removal, where simulated events are removed, or overlap subtraction, where MC weights are assigned to the simulated events which can result in negative MC event weights.

A dipole approach is used by simulation generators like the general-purpose generator Pythia 8 [21], where the PS is modelled using the emission of colour dipoles, which automatically preserves colour flow. QED radiation is simulated using the same approach as for QCD radiation as both can occur in pp collisions. However, in comparison to QCD radiation, where colour flow is crucial in the evolution, charge conservation determines the basic dynamics of QED radiation. Only Leading Order (LO) calculations are considered due to the strength of the coupling constant, which is much smaller than the strong coupling constant.

The simulation of the hadronisation process transforms the final state coloured partons into colour neutral hadrons. Hadronisation requires modelling from experimental data as it cannot be calculated from first principles. The two methods available to model hadronisation are string fragmentation, of which a variant is applied in Pythia 8 simulations, and cluster fragmentation. String fragmentation is built on the QCD concept of linear confinement at large distances. Its theory predicts breaks in the colour flux tube between a $q\bar{q}$ pair, which defines a string, at large enough invariant masses. The cluster fragmentation method is based on a pre-confinement property of PS and results in a uniform mass distribution at low energy scales. This leads to limited p_{T} spectra and, among other effects, to a suppression of hadrons including heavy flavour quarks. Using data from the Particle Data Group (PDG) database [22] in these models is not enough to simulate hadronisation as there are a lot of free choices required in the modelling or tunable parameters. Therefore, dedicated simulation software are used, which specialises in different aspects of the simulation of physics processes. EvtGen [23] is a software packages, which specialises in the simulation of hadronic decays, notably for decays of the b and c-mesons. Other hadronisation simulation generators like Herwig++ [24] or Sherpa [25] are more suited to model τ-decays as they include a more suited description of spin effects involved in the decay of the τ lepton. The spin of the t quark is modelled with MadSpin to preserve the its spin information in the decays.

3. LHC and the ATLAS Experiment

Contents

Consistency with experimental observations is essential for the validity and acceptance of physics theories like the Standard Model or Beyond the Standard Model Theories. The Large Hadron Collider, located at European Organisation for Nuclear Research (CERN) near Geneva, is the largest and most powerful human engineered particle collider. It is therefore a powerful tool to test the Standard Model and theories which go beyond it. Large general-purpose detectors which are designed to operate in the environment around the particle collisions of the Large Hadron Collider are of special importance to these tests. With the ATLAS detector being the largest general-purpose detector at the Large Hadron Collider (LHC), its design and performance is of special interest. As is the Compact Muon Solenoid (CMS) detector, the second largest general-purpose detector at the LHC with a different construction set-up but the same scientific aims.

In the following sections of this chapter, a broad overview of both the accelerator and storage systems of the LHC are presented. This includes the CERN accelerator systems which produce beams of high energy particles. These beams are injected into the LHC which then subsequently accelerates the beam particles even further.

After the quick overview over the supplying beam facilities, the main focus of this chapter is the design of the ATLAS detector and its subsystems relevant to work presented in this thesis.

3.1 The Large Hadron Collider

The LHC [26] is a circular hadron accelerator with a 27 km circumference. Its main task is to accelerate pre-accelerated particles even further to nearly the speed of light in vacuum and subsequently systematically collide those beams. For this the accelerator relies on a series of other accelerating units, referred to as the injector chain. Each subsequent accelerator in this injector chain is dedicated to accelerating particles to a certain energy. The beam of particles is then injected into the LHC. A schematic overview of connected facilities is shown in Figure 3.1.

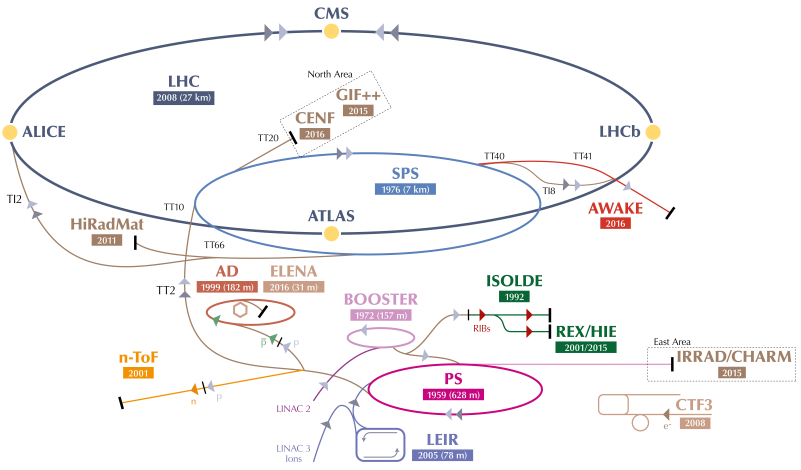


Figure 3.1: The LHC and its surrounding accelerator and storage units as well as detector systems [27].

The injector chain [26] consists of a number of subsequent accelerator systems and storage rings. For protons, it includes a linear accelerator (Linac 2), the Proton Synchrotron Booster (PSB), the PS and the Super Proton Synchrotron (SPS). The delivery of ultra high energy protons starts with a bottle of hydrogen gas. Upon release of the hydrogen gas its electrons are removed using an electric field to retrieve pure protons. These protons are then accelerated using a linear accelerator. There, electric fields of alternately charged conductors accelerate the protons to energies of 50 MeV. The next acceleration steps happens in the PSB, where four synchrotron rings which are stacked upon each other increase the energies of the protons fur-

ther to 1.4 GeV. The beams are then passed on into the PS using a two-batch filling scheme. The beam batches referred to as bunches from there on. The bunch structure is then further changed by the subsequent associated accelerator systems. Like the PSB, the PS is also a synchrotron but it increases the energy of the proton beams to 25 GeV. After this, the proton beams are further accelerated in the SPS, another synchrotron, up to energies of 450 GeV. The beams then progress in bunches via two injection points into the LHC.

Ion beams are accelerated similarly but in a different accelerators up until injection into the PS where they replace the protons in case of dedicated runs. The heavy ions (A) are first accelerated by a dedicated heavy ion linear accelerator (Linac 3). The heavy ion beam energy which can be delivered is referred to as energy per nucleon (u), as the element chosen for heavy ion runs might vary. At the Linac 3 the heavy ions are accelerated to 4.2 MeV/u before being further accelerated in the Low Energy Ion Ring (LEIR) facilities to 14.8 MeV/u. From here they are injected into the PS, which accelerates them further to 4.25 GeV/u and then into the SPS which subsequently accelerates them to 177 GeV/u before injection into the LHC. Heavy ions like Lead, used in $p-Pb$ or $Pb-Pb$ collisions, or Xenon, selected for $Xe-Xe$ collisions, are accelerated in the LHC to energies resulting in a centre of mass of 5.02 TeV/u during Run 2 data taking, exceeding a centre of mass energy of 1 PeV.

In the LHC, the beams of particles are then further accelerated, travelling next to each other in opposite directions around the ring in two separate beam pipes. In addition, they are also arranged into bunches of particles to provide cleaner timestamps and have better control over the collision parameters. While circulating the beams in the LHC, superconducting dipole electromagnets provide strong magnetic fields to bend the beams to keep them within the beam pipes and re-circulating the beams within the LHC storage ring to bring the beams into the desired final form and up to the desired collision energy. As this takes time and slight variations in particle velocity directions within the beam might cause the beam to fizzle out, quadrupole magnets are used to focus the beams to prevent losses along the travel and increase the particle density within the beams. Along the LHC exist four beam crossing points for collisions of the beams to take place. These four points coincide with the nominal interaction points for the four major LHC experiements. Next to ATLAS and CMS this includes LHCb and A Large Ion Experiment (ALICE). The Run 2 beam centre of mass energy of pp collisions is 13 TeV.

3.1.1 Luminosity Measurement at the Large Hadron Collider

Alongside the beam energy, another important design parameter of the LHC is the instantaneous luminosity (L), given by

$$\mathrm{R} = \mathrm{L} \times \sigma_{int}. \tag{3.1}$$

It represents the proportionality factor between the event rate R and the interaction cross section σ_{int} of the beam particles. The higher the instantaneous luminosity, the higher the number of collisions per time interval and the denser the environment around the beam collision point. This results most importantly in an increase in a higher number of interesting physics events but also in the downside of a higher radiation for the entire detector environment. The design luminosity of the LHC is L = 10^{34} cm^{-2}s^{-1} with a peak luminosity during Run 2 of about 2.14 $\times$ 10^{34} cm^{-2}s^{-1}. However, since the beams are delivered in bunches by the LHC, the luminosity per bunch crossings, the bunch luminosity ($\mathcal{L}_b$) [28], is defined as

$$\mathcal{L}_b = \frac{\mu f_{\mathrm{r}}}{\sigma_{\mathrm{inel}}}. \tag{3.2}$$

$\mathcal{L}_b$ is used to provide a measure of the intensity of the beams per bunch and is calculated using direct measurements of the beam parameters [29, 30]. The pile-up parameter μ represents the average number of inelastic interactions per bunch crossing, f_{r} is the bunch revolution frequency and σ_{inel} is the inelastic cross section of the colliding beam particles. The uncertainty on $\mathcal{L}_b$ contributes a major systematic uncertainty in the cross section measurement of SM precision measurements and it also influences background levels in BSM searches. The quantity $\mathcal{L}_b$ therefore is a measure to quantify the number of expected inelastic interactions.

However, since data taking over pp physics runs with Run 2 conditions involves bunch separations of 25 ns, the total luminosity is of more practical value. The calculation of the total luminosity via integration of L over time, taking into account multiple bunch crossings, is given by

$$\mathcal{L} = \int \mathrm{L}\ \mathrm{dt} = \Sigma_{i=1}^{n_b} \mathcal{L}_{b_i} = n_b \langle \mathcal{L}_b \rangle = n_b \frac{\langle \mu \rangle f_r}{\sigma_{inel}}. \tag{3.3}$$

Due to statistical variations in the distribution of particles in bunches, the pile-up parameter μ is measured across multiple bunch crossings to get the total number of collisions. Over the entire LHC Run 2, an integrated luminosity of 158 fb^{-1} [31] was delivered.

3.2 The ATLAS Detector

The ATLAS detector [32] is the a forward-backward symmetrical general purpose particle detector for measuring pp, pA and AA collisions and probe the contained physics. With a weight of 7 tonnes, it weighs about as much as the Eiffel tower. It is located at Interaction Point 1, one of the LHC collision points which for ATLAS defines the nominal interaction point (IP) where collisions are expected to take place.

To provide the largest possible coverage and efficiency for recording the kinematics and trajectory of particles created in collisions, the detector, which is centred around the IP, consists of seven different subdetectors which are layered concentrically around the detectors center. These seven subdetectors are the Pixel, the SemiConductor Tracker (SCT), Transition Radiation Tracker (TRT), Magnet System, electromagnetic calorimeter (ECAL), hadronic calorimeter (HCAL) and the Muon Spectrometer (MS). Their layering is closely related to their detection technique to target the measurement of different physics objects and particles produced in the collisions. After introducing the coordinate system used by the ATLAS Collaboration, the main performance goals and the overall structure of ATLAS will be discussed. After this the focus of this section will be on discussing the individual layers from the IP outwards.

ATLAS Coordinate System

The IP is the origin of the right-handed coordinate system used by ATLAS. Its individual axes are defined with the z-axis pointing along the beam line with the x-y plane transverse to it. The x-axis points towards the centre of the LHC ring and the y-axis points upwards. For convenience spherical coordinates (r,θ,ϕ) are used superimposed on the right-handed cartesian coordinate system along the z-axis in the transverse plane. In this superimposed coordinate system r is the radial distance from the IP and ϕ is the azimuthal angle around the beam pipe.

The polar angle θ, the inclination with origin in the IP as calculated from the positive z-axis, is expressed in terms of the pseudorapidity η , an approximation of the rapidity y, which is defined via the energy E of the particle and its longitudinal momentum component p_z along the positive beam axis as

$$y = \frac{1}{2} \ln \left(\frac{E + p_z}{E - p_z} \right),$$

for which particle production is constant per unit. Unfortunately, the rapidity is not necessarily Lorentz invariant under longitudinal boosts for all particle masses or

speeds. The pseudorapidity on the other hand is an approximation in the relativistic limit in a frame where objects only have a velocity perpendicular to the beam axis. It is given by

$$\eta = -\ln\left[\tan\left(\frac{\theta}{2}\right)\right],$$

which is Lorentz invariant under longitudinal boosts along the beam direction. Therefore, the pseudorapidity η is chosen and preferred compared to the rapidity y.

Performance Goals and Detector Overview

To perform high precision measurements of SM processes and searches for BSM physics, it is essential to be able to measure collimated cascades of particles from hadronic or electromagnetic processes ($jets$) and leptons as well as determine missing transverse energy ($E_{\mathrm{T}}^{\mathrm{miss}}$). Measurements of the kinematic properties of particles over a wide range of energies, from the GeV scale to a few TeV, are fundamental for physics analyses and need to be combined with high granularity for good pattern recognition capability. Specially as new physics might be accessible via high-p_{T} jet measurements where the jets originate from a b-hadron, which creates challenges especially for trajectory pattern recognition in densely populated environments. Therefore, the detector which provides the data for any of the following steps of object reconstruction, is required to have good particle triggering in place and record data with very high precision and granularity, especially around the IP. The detector especially needs to provide very good particle identification and precise momentum measurement capabilities. The same holds true for the containment and precise measurement of particle cascades created by hadronic or electromagnetic particles from collisions. The detector components are required to be highly resistant to radiation in order to operate in the harsh environment surrounding the beam collisions. Support structure, cables and different operating temperatures of subdetector systems also need to be taken into account.

To match its performance goals, ATLAS utilises multiple different subdetectors. Moving from the interaction point outwards, ATLAS consists of a tracking detector system for interaction origin and particle travel path determination, surrounded by a strong solenoidal magnet to bend charged particles. Moving outwards radially, this is followed by both an electromagnetic and a hadronic calorimetry system for particle cascade generation and measurement. The final outermost subsystem is the MS which is used to detect muons, resolve multiple trajectories ambiguities. Additionally, the MS is encompassed by a toroidal magnetic field. A schematic overview of

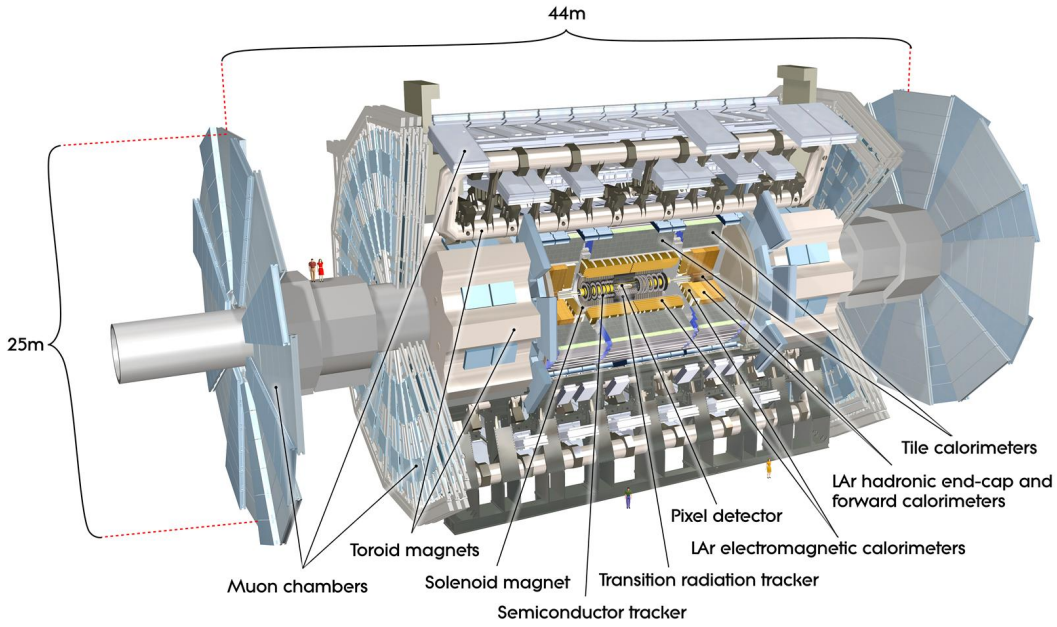


Figure 3.2: Schematic cut-away overview of the ATLAS detector with its individual subdetectors [33].

the detector with its individual subsystems is shown in Figure 3.2. The individual subsystems of the detector are described in the following sections of this chapter.

3.2.1 Magnet System

Magnetic fields bend the trajectories of charged particles due to the electromagnetic interaction. This provides an easy tool for some subdetector parts to distinguish charged from neutral particles and measure the momenta of charged particles more precisely. In combination with momentum measurements this makes it possible to quantify the charge of the particles. However, in other subdetectors a deflection in the travel direction for charged particles might be undesired and distort the measurements of the energy deposits. The magnet system of ATLAS consists of a hybrid solution with two different magnet arrangements. Each magnet arrangement provides a uniform field for certain subdetector systems, whilst not introducing undesired effects in the other detector subsystems.

The magnet closest to the IP consists of a thin superconducting solenoid magnet which surrounds the Inner Detector cavity. It provides an axial magnetic field of 2 T to enable trajectory signatures from charged particles to be distinguished from those from neutral particles and momentum measurement.

The second magnet arrangement consists of several large superconducting air-

core toroidal magnets encompassing the hadronic calorimeters. The air-core construction was chosen to minimise material density. Here, multiple toroids are arranged symmetrically around the interaction point in the central (*barrel*) and forward (*end-cap*) regions. A central barrel toroid, which produces a magnetic field of strength 0.5 T, covers the range $|\eta| < 1.4$. End-cap toroids, one on either side of the barrel, cover $1.6 < |\eta| < 2.7$ and produce a magnetic field of the strength of 1.0 T in the end-cap region. In the transition region between these two toroidal magnet arrangements $1.4 < |\eta| < 1.6$ the overlapping fields of the barrel and end-cap toroidal magnets are used to provide the magnetic field. These toroidal magnets provide a weaker magnetic field in comparison to the solenoid but over a much larger volume to bend the muons as they are traversing through the muon spectrometer.

This hybrid solution provides suitable magnetic fields for the individual subdetector systems while not interfering with the particle cascades in the calorimeters, where deflected particle trajectories would degrade the performance.

3.2.2 Inner Detector

The Inner Detector (ID) is the closest subdetector to the IP, operating in an environment with very high concentration of highly energetic particles where high radiation resistance of the detector is essential. The system consists of three independent but complementary systems, operating within an axial magnetic field, provided by the ATLAS solenoid, to bend the trajectories of charged particles produced in collisions. Each of these systems is composed of a barrel structure, with the detector elements arranged in concentric cylinders around the beam axis and two end-caps, one on each side of the barrel. In the end-caps the detector elements are either arranged in disks or wheels, positioned perpendicular to the beam axis. Its design is a trade-off between high performance and material density. More sensors would increase the number of measurements but also increase the multiple scattering pollution via the increasing interaction cross section with the detector material. This results in an increased deflection in the particles travel direction due to multiple Coulomb or hadronic interactions with the nuclei they pass through. This influences the paths taken by the particles as they travel through the detector.

Following the flight path of scattered particles from the interaction point, layers of semiconductor technology, first pixel detectors then silicon microstrip SCT, cover measurements in the region $|\eta| < 2.5$. This is then followed by the straw-tube TRT which covers the region $\eta \leq 2.0$. The layout of the ID is illustrated in Figure 3.3, where the different subsystems are pointed out.

The semiconductor sensors which are both used in the Pixel and SCT systems

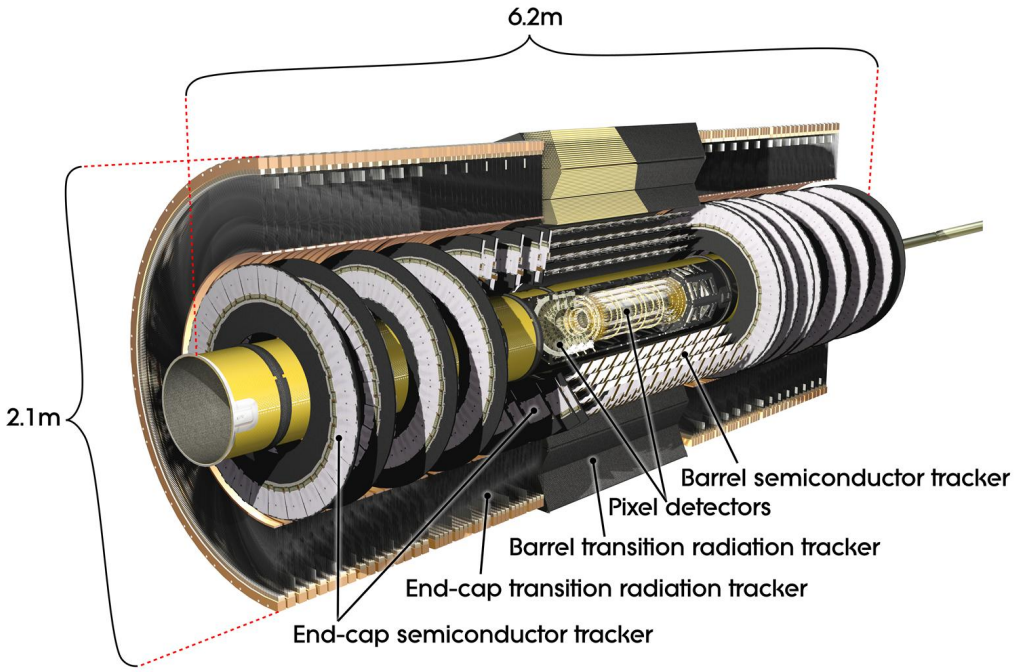


Figure 3.3: Schematic cut-away overview of the ATLAS ID and with its systems labelled accordingly [34].

operate based on electron-hole creation, which creates a current which is then propagated to readout electronics. Scattered particles deposit energy in these pixel detectors from which particle trajectories, interaction origins and momentum measurements can be reconstructed.

Pixel Detector

The pixel detector subsystem is closest to the IP. It consists of four layers of silicon wafer readout pixel detectors which are organised in a matrix arrangement. The layers are arranged in concentric cylinders in the barrel region and as disks in the end-caps. These pixel sensors make use of semiconductor sensors. Therefore, when a charged particle is incident on a pixel, these detector layers provide discrete two-dimensional space-point measurements. Their positioning provides the dimensional expansion to a three-dimensional hit measurement with high granularity for good pattern recognition performance. This is of special importance for the detection of trajectories associated to b-hadron decays and being able to distinguish their signatures from other decays in data as a b-hadron is exhibiting a secondary interaction point upon its decay, which is displaced only by a few mm from the IP, depending on its energy. The innermost layer of Pixel is the Inner Barrel Layer (IBL) [35, 36], an upgrade for Run 2 of the LHC which increased the number of Pixel layers

from three to four with the IBL layer even closer to the IP. This additional layer is increasing tracking resolution near the IP and increases the resolution for particle interaction points, which helps in the detection of b-hadron decays.

SemiConductor Tracker

The SCT system works similar to the Pixel detector described above with the change to stereo strip detectors while maintaining high granularity. It consists of four coaxial cylindrical layers in the barrel region and nine disk layers at each end-cap side. Here two strips measure back-to-back nearly simultaneously per layer of SCT with one strip in each layer placed parallel to the beam direction and the other placed back-to-back at angle of 40 mrad. This provides a measure for both θ and ϕ, which allows a reconstruction of particle trajectories in three dimensions instead of only a two dimensional plane.

Transition Radiation Tracker

The TRT is the outermost component of the ID. It consists of polyamide drift (straw) tube technology segmented into a barrel component and 192 disks on each end-cap side of the barrel. The straw tubes are filled with a $Ar/CO_2/O_2$ gas mixture and contain a gold plated tungsten anode wire which collects the photon radiation of transition radiation, which is generated with varying signatures by different charged particles. This property enables the TRT with particle identification capability as it is able to distinguish charged particles like electrons, pions, muons or kaons from each other based on their energy deposition into transition radiation. This property is largely used for electron identification against other signatures. The tubes are stacked into many layers and are interleaved with dielectric transition radiation generating material (polyethylene). By measuring the drift time, the tube segmentation together with their measurements provide a three-dimensional hit measurement. These tubes provide many hits, which contribute to the determination of the p_{T} resolution in the reconstruction of particle trajectories. The technology is well suited for the high radiation environments expected in Phase 1 of the LHC.

3.2.3 Calorimetry System

The ATLAS calorimetry system has a fully ϕ-symmetric design and aims to measure the energy, position and momentum of electromagnetic as well as hadronic particles. The system comprises multiple calorimetry technologies organised in barrel and end-cap structures, as shown in Figure 3.4, with coverage up to $|\eta| < 5$. The

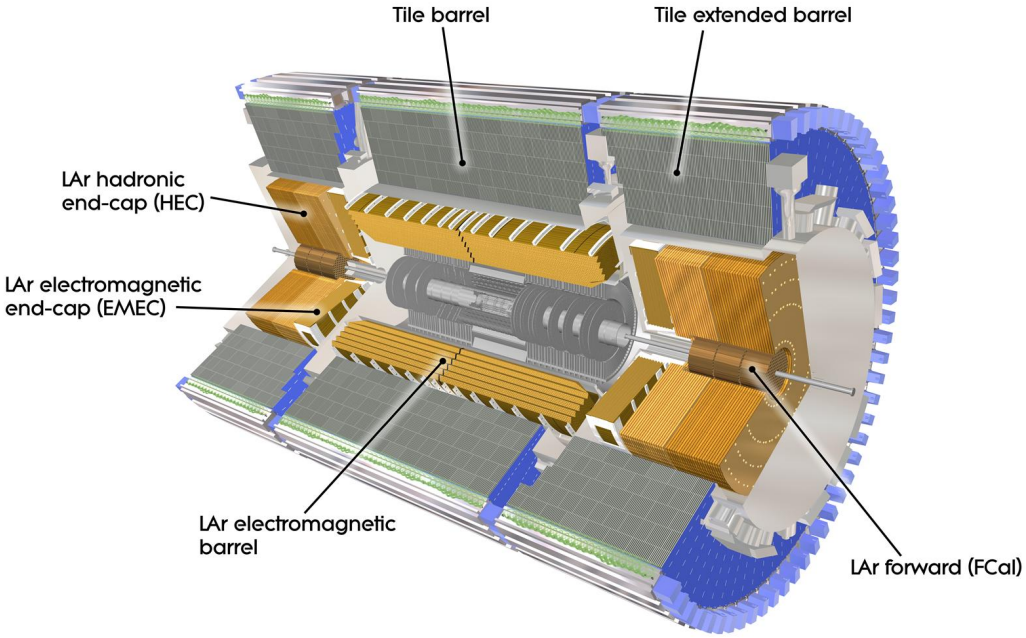


Figure 3.4: Schematic cut-away overview of the ATLAS calorimetry systems with its individual systems labelled accordingly [37].

ECAL, together with the ID, is enclosed within a vacuum chamber. It focusses on the detection and absorption of electrons and photons created in collisions and their associated particle cascades. However, only a fraction of the energy of hadronic particles is measured by this detector subsystem. Therefore, a complimentary outer layer of calorimeters, the HCAL, surrounds the ECAL. The HCAL absorbs and measures hadrons as well as particles of the resulting interactions with the detector material.

Both the electromagnetic and hadronic calorimeter make use of an absorber and active material. The absorber is a dense material, which increases the interaction probability with the incoming particles from the IP, creating collimated cascades of particles. The active material produces the signal for the measurement of the cascade constituents. Each calorimeter module comprises three complimentary layers. For the ECAL, the first layer provides a high precision position measurement with a fine segmentation in η, whereas the segemtation in the first layer of the HCAL is much coarser. The middle layer comprises more material to absorb most of the energy of the cascade. The outer layer serves the purpose to estimate the amount of energy escaping the module and has a coarser segmentation in η.

The ECAL measures the energy and location of electrons and photons via energy clusters and is built to contain the electromagnetic cascades of such particles

within the ECAL volume. Lead plates are used as the absorber material. The electromagnetic cascades originating from electrons or photons are initiated and propagated via bremsstrahlung and e^+e^- production. This chain of particle production in the cascade continues until the energy is below the thresholds for these processes to take place due to preceded energy depositions in the material. The ECAL collects the signals from the electrons and photons. Liquid argon (LAr) constitutes the active material between the absorber plates. A high voltage is applied across the LAr to collect the ionisation electrons and amplify the signal. In the region which is covered by the ID, the ECAL has a higher granularity for more precise measurements which also helps to match electron clusters to tracks in the ID.

The HCAL is a collection of sampling calorimeters which focus on measuring the energy deposited by hadron-gernerated cascades of particles. These cascades have a different lateral development compared to pure electromagnetic cascades as the hadrons interact mainly via inelastic interactions with the absorber nuclei, producing non-electromagnetic energy along further interactions. During the cascade development, large amounts of pions are produced via inelastic interactions. Of these pions, the charged pions decay leptonically into mostly a μ and $\overline{\nu}_\mu$ pair. However, the π^0 mainly decays into two photons, resulting into an subcascade of electromagnetic particles. Within the hadronic cascades neutron capture leads to fission, a release of binding energy, which escapes detection. The subdetector system is surrounding the ECAL and has various components arranged in the barrel and end-cap regions. These are the tile calorimeter, the LAr hadronic end-cap calorimeter (HEC) and the LAr forward calorimeter (FCal). The tile calorimeter, which uses scintillating tiles as the active material, is split into a central barrel covering $|\eta| < 1.0$ and two extended barrel partitions covering $0.8 < |\eta| < 1.7$. The tiles are producing the signal which is then propagated to photomultipler tubes using wavelength shifting fibers. This technology provides the best cost-performance ratio as it provides maximum radial depth at comparable low cost. Given the technologies sensitivity to radiation when compared to other technologies, the system is protected against radiation damage by the LAr electromagnetic calorimeter, which exposes it to much lower radiation levels compared to other HCAL systems. The HEC consists of independent calorimeter wheels which use copper as absorber material and LAr as active material. It is located behind the ECAL end-cap wheels extending over the range $1.5 < |\eta| < 3.2$. The FCal covers the highest pesudorapidity range of the HCAL as it covers $3.1 < |\eta| < 4.9$. Similar to the HCAL, the FCal also uses LAr as active material but in addition to copper also uses tungsten absorber plates. While the segmentation for the HCAL is coarser than for the ECAL, it can still precisely

measure clusters and energy deposits.

3.2.4 Muon Spectrometer

The ATLAS MS detects charged particles and measures their momentum within $|\eta| < 2.7$. As the name suggests, the focus is mainly on muons from collisions but the system can also provide information to detect punch-through when a cascade is not fully contained by the hadronic calorimeter. Muons might only leave a small amount of energy deposit in the other subdetectors but the majority of their energy would escape undetected. To prevent this and reduce material density in front of the other subsystems, the MS is the outermost measurement system. It defines together with its associated toroidal magnet system the detectors overall dimensions. As an independent system, the MS is able to detect muons with momenta ranging from $\sim$ 3 GeV to about 3 TeV with the limiting factor arising from the bending force of the magnet.

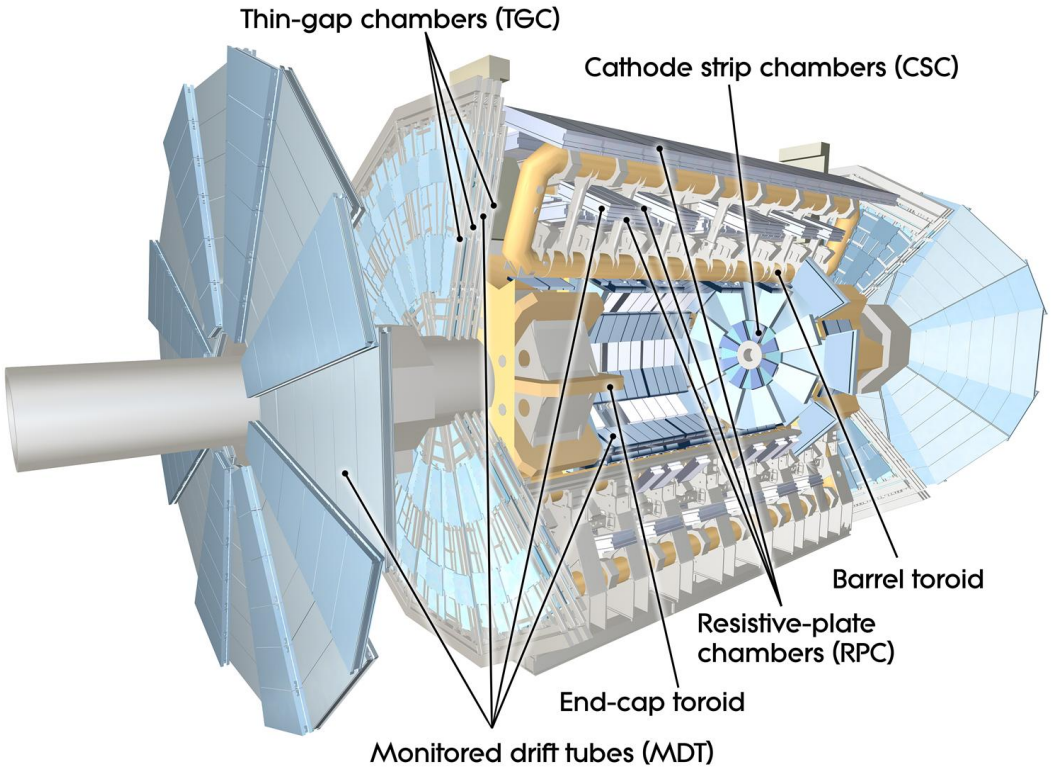


Figure 3.5: Schematic cut-away overview of the ATLAS MS with its individual systems labelled accordingly, including the assisting air-core toroidal magnets [38].

As shown in Figure 3.5, the MS comprises complimentary tracking as well as triggering chambers: Monitored Drift Tubes (MDTs), Cathode Strip Chambers

(CSCs), Thin Gap Chambers (TGCs) and Resistive Plate Chambers (RPCs). The magnetic fields enable momentum measurement for tracks with at least three hits in the tracking chambers, the MDTs and the CSCs. Ionisation charge drift times within the tracking chambers of up to 700 ns are possible. Therefore, the time stamp of an incident charged particle is determined by the trigger chambers, the TGCs or RPCs, and can then be compared with the collision time for possible matching. The trigger chambers also provide information to systems which make quick decisions based on predefined criteria on whether information is relevant enough to be kept or not. These decision making systems are referred to as triggers.

The MDTs are pressurised drift tubes, which are monitored optically on their positions and internal deformations. Within these tubes the pressurised gas gets ionised by charged particles passing through. The ionisation current is travelling via an applied electric field towards a wire in the middle of the tube where this signal is collected. These drift tubes are arranged in chambers of three to eight layers, which are stacked upon each other in parallel, and perform the precision momentum measurement of charged particles. Each tube provides an independent measurement in the bending direction η. The sizes of these chambers vary increase the further away from the IP they are located. Their coverage is $|\eta| < 2.7$ with the exception of $|\eta| < 2.0$ for the innermost end-cap layer. The CSCs measure both track coordinates, one in the bending plane η and the other in the non-bending plane ϕ, simultaneously are used in the innermost layer in the forward region. The system encompasses the range $2.0 < |\eta| < 2.7$, where the safe operation counting rates for the application of MDTs are exceeded. For both systems the radial coordinate is given by the modules location. Its resolution is of less importance given the distance to the IP and the requirement for muons from collisions to leave matching hits in the subdetectors which are closer to the IP. In comparison to the MDTs, CSCs have a higher granularity as well as higher readout capability as well as a higher time resolution and response. This makes them better suited for the region encompassing $2.0 < |\eta| < 2.7$ where the particle flux and thus the muon track density is highest and the background conditions more challenging. They also help to resolve multi-trajectory ambiguities, where due to high densities the wrong trajectory might be reconstructed because of wrongly associated hits.

The trigger chambers, TGCs and RPCs, are used to allow to make nearly instantaneous decisions on the presence of muons and their basic properties. They measure and allow to trigger on muons over the region $|\eta| < 2.4$ and measure two coordinates of the charged particle trajectory hits. One is measured in η and the other one in ϕ. In combination with MDTs these systems are providing the ϕ co-

ordinate to the MDTs measurements. The precision-tracking RPCs cover the barrel region of $|\eta| < 1.05$. This technology provides good spatial and time resolution and its signals are therefore more suited for precise trigger operation. The TGCs, which are multi-wire proportional chambers, are used in the end-caps, encompassing $1.05 < |\eta| < 2.4$. They provide high rate capability, good time resolution and granularity to allow the calculation of an estimate of the momentum of a triggered muon for triggering given these triggers are hardware triggers which fully rely on the input by hardware which allows to make a faster decision using an algorithm.

3.2.5 ATLAS Luminosity Measurements

Since the luminosity delivered by the LHC is divided between different experiments, it is of high importance for ATLAS to be able to measure luminosity. This is necessary as trigger related decisions depend on the instantaneous luminosity. Long term, this is important to keep track of the measured luminosity of the recorded data in order not to over- or underestimate the measured processes with MC simulated data in physics analyses and to measure the absolute cross sections of processes more precisely.

Two independent detector systems LUminosity measurement using Cherenkov Integrating Detecor (LUCID) and Absolute Luminosity For ATLAS (ALFA) operate in the ATLAS forward region outside the vacuum enclosure at different distances from the IP and monitor the luminosity for ATLAS. They concentrate on measuring $\mathcal{L}_b$ via the following relation

$$\mathcal{L}_b = \frac{\mu_{\mathrm{vis}} f_{\mathrm{r}}}{\sigma_{\mathrm{vis}}}. \tag{3.4}$$

This calculation is derived from Equation 3.2 and uses the visible interaction rate per bunch crossing μ_{vis} as well as the visible cross-section σ_{vis}. Both of these quantities are measured using independent detector systems [28, 39] as well as different algorithms and take detector specific efficiencies into account. Those efficiencies are calibrated using beam-separation scans, relating them to the beam parameters and coordinates.

3.2.6 Trigger and Data Acquisition

With an average of 36 events per bunch crossing in 2018 and approximately 40 million bunch crossings per second, ATLAS saw about 1.44 billion events per second. Peak number of events per bunch crossings of 90 were recorded, bringing this number up to 3.6 billion events every second. Since each event requires approximately

1.5 MB of disk space, this means that ATLAS would be required required to store 2.2 TB/s on average, with peaks up to 5.4 TB/s. In addition, each event requires handling of $\approx$ 1600 point-to-point readout links. The Trigger/Data Acquisition (TDAQ) system provides ATLAS with highly efficient data-taking and event-selection capabilities in order to reduce the event rate in a feasible and manageable way. A quick overview of it is given in this section.

At design luminosity of 10 nb/s, the data rate of pp collisions is about 40 MHz whereas event data processing and storage requires a reduction by about seven orders of magnitude to be manageable. Fortunately, the majority of pp collisions are not of interest in the searches for new physics or SM precision measurements and can be dropped at various stages of the TDAQ chain. If an event passes this first filtering, the subsequent Level 1 (L1) hardware trigger system which uses a subset of the total detector information from the calorimetry and muon systems is used to make a rejection decision on the event. This reduces the data rate down to 100 kHz. Then, before being stored, the event is required to pass the last high-level trigger system, which has access to the full event information including the data from the front-end readout electronics systems as well as regions of interest selected by the L1 hardware trigger. During this stage, trajectories are reconstructed from hits in the ID and further physics objects are reconstructed as well. An event is only stored if it passes both on these L1 and HLT trigger requirements. The data rate at this point is reduced to about 1.5 kHz. Each event requires approximately 1.5 MB of disk space. Therefore, only about 2.3 GB are required to be stored every second.

4. Object Reconstruction

Contents

The detector measurements only record the interactions of particles with the active measuring elements of the various detector subsystems. Therefore, in order to relate these signatures to physics objects, both online and offline algorithms are used to reconstruct the particles. The reconstruction algorithms rely largely on algorithms which range from simple cuts to expensive pattern recognition techniques. Therefore, some reconstruction algorithms are applied offline as they may be very computationally intensive. Dedicated Combined Performance groups provide recommendations for the reconstruction of physics objects. These recommendations might change over time, which supports applying these algorithms which include these recommendations offline. The precision of reconstruction algorithms and systematic uncertainties originating from them are crucial to all physics analyses.

In this chapter, an overview is given of the reconstruction methods for the most commonly used physics objects employed in physics analyses using data recorded by the ATLAS detector. These objects comprise leptons, photons, collimated particle showers and the traverse energy which escapes detection. The collimated particle showers originate from one quark or gluon produced in collisions, which undergoes hadronisation, parton scattering and decays, and are known as jets. The transverse energy which escapes detection is referred to as missing transverse en-

ergy. Reconstructed trajectories of particles, known as tracks, and particle interaction points, referred to as vertices, as well as energy deposits in the calorimeters are the basic building blocks of these physics objects.

4.1 Tracks and Vertices

As charged particles pass through the ID they record hits upon traversing the active measuring elements. From these hits, the trajectories of charged particles and their origin can be reconstructed as tracks and vertices. As they are dependent on the ID hits, their reconstruction is limited to the ID coverage of $|\eta| < 2.5$. The hits are used to reconstruct the particle's tracks, which are subsequently used to determine the interaction vertices. The accurate determination of the association of tracks with the correct vertex and the precise location of the vertex is crucial for correct physics object reconstruction. Ambiguities in track and vertex reconstruction can mislead further pattern recognition if the association to the physics object is incorrect and even lead to misinterpreting the underlying physics.

Primary vertices are the first interaction points in the collisions of the beam particles. They can be slightly displaced from the IP due to the nature of the bunch crossings, though it is intended for the primary interaction to take place at the IP. Therefore, due to the bunches being three-dimensional objects, the beam spot is defined as the area where the majority of primary interactions during a single bunch crossing occur. It is determined using the global maximum of the z-coordinate distribution of all tracks and is not expected to be exactly at the IP at the xyz-coordinate (0, 0, 0). Since this area is indicating the highest track density, the highest population of primary vertices coinciding with the beam spot is expected in the same area.

Track Reconstruction

The preferred charged particle track reconstruction technique is the inside-out track reconstruction [40, 41]. This technique represents a sequence of reconstruction steps, which involve global as well as consecutive local pattern recognition within the ID. The reconstruction starts with three-dimensional hits recorded in the pixel detector and SCT. They are processed to identify possible candidates for starting points of tracks from which their trajectories are reconstructed, known as track seeds. Track seed finding is performed without requiring the position along the z-axis to be close to the beam spot. It is followed by matching other hits to the track seeds to form a track, proposing likely trajectories as track candidates. A Kalman filter method is used to predict the natural progression of the trajectory of the initial

charged particle by fitting the trajectory and including matching hits in the track candidate fit. This formalism also detects outliers, whose hits have larger distances to the calculated fit trajectory, and neglects them in the subsequent track finding process. Due to the high combinatorics for hits to be associated to a track candidates, which increases with denser environments as well as higher pile-up conditions, it is possible that track seeds can be associated to multiple track candidates. In order to resolve ambiguities, track candidates are ranked and the hit is assigned to the highest ranking candidate. They are ranked by the likelihood according to which they reflect the original trajectories of the charged particles. The score of a track also depends on multiple factors and attempts to incorporate physics knowledge by use of structural parameters and importance weighting of hits to an otherwise pure pattern recognition task. If multiple track candidates share the same hit, a multi-layer perceptron algorithm [42] is used to resolve these ambiguities. Preferential treatment of hits close to the IP is included using weights. This gives more importance to the pixel layers closest to the beam spot as they provide a measurements of higher precision for vertex location compared to those with larger distance to the beam spot. Information from the TRT is also used for ambiguity solving if including TRT hits results in an overall improvement in the reconstruction of the track candidate. This includes testing whether the hits in the TRT follow the natural progression of the track reconstructed using only the pixel detector and SCT, referred to as silicon tracks.

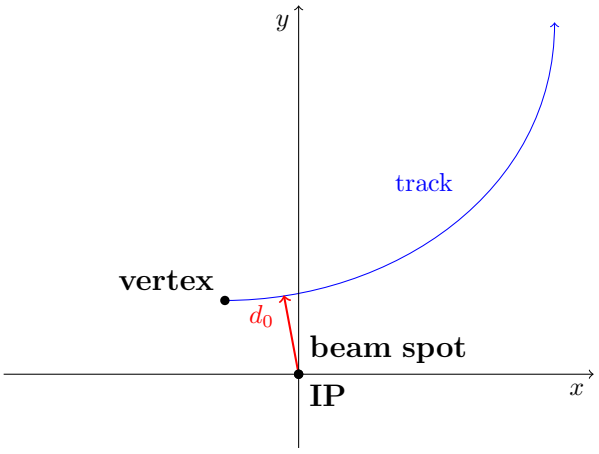


Figure 4.1: Schematic representation of the definition of the displacement parameter d_0 in the transverse plane.

A reconstructed track is described by its direction in ϕ and θ, its charge-to-momentum ratio q/p as well as by its closest approaches to the beam spot in the transverse and longitudinal direction. these distances of closest approach in differ-

ent planes are known as transverse impact parameter d_0 and longitudinal impact parameter (z_0). The parameter d_0 is illustrated in Figure 4.1, which shows the projection of the track in the transverse plane. The longitudinal impact parameter z_0 represents the closest approach in radial distance to the beam spot along the beam axis. The uncertainties on the impact parameters given the reconstructed track fit are known as the d_0 uncertainty ($\sigma(d_0)$) and z_0 uncertainty ($\sigma(z_0)$).

Depending on their use, different quality requirements are applied to the tracks. These quality requirements typically include constraints on p_{T}, the minimum number of silicon hits or the maximum number of holes allowed per track. A hole refers to a missing signal hit in an active silicon module along the fit of the reconstructed track. Upper limits on d_0, z_0, $\sigma(d_0)$ and $\sigma(z_0)$ may also be applied.

Vertex reconstruction

The identification of interaction points as reconstructed vertices [43, 44] relies heavily on the precise reconstruction of charged particle tracks. Vertices are reconstructed using two algorithms, the primary iterative vertex finding algorithm and the adaptive vertex fitting algorithm [43], where the finding algorithm uses the fitting algorithm.

The iterative vertex finding algorithm selects reconstructed tracks which pass quality selection criteria and include tracks with transverse momentum down to 400 MeV. These potential vertex candidates are constrained to be close to the beam spot. In the next step, their position is determined using the adaptive vertex fitting algorithm. This vertex fitting algorithm is based on a χ^2 fitting algorithm and uses the vertex candidates and the tracks around them as inputs. It uses a soft cut-off value for the χ^2 value in the association of tracks to vertex candidates. Each reconstructed vertex candidate is required to have at least two associated tracks. Outlying hits are assigned a lower weight in the overall χ^2 value per vertex fit in this procedure. The vertex finding algorithm is repeated until all tracks are associated to a vertex or no additional vertex candidates can be found.

The main primary vertex is determined as the vertex with the largest sum of p_{T}^2 of the associated tracks. The total number of primary vertices is used to determine the instantaneous pile-up parameter, which is important for trigger operations and calibrations.

After the vertices are determined, the parameters d_0 and z_0 are redefined with respect to the primary vertex in the event. In addition, both $\sigma(d_0)$ and $\sigma(z_0)$ are recalculated in order to reflect the change of reference point. Due to the proximity of the pixel modules to the beam spot, both impact parameter uncertainties strongly

depend on the precision of individual measurements in the four pixel layers.

4.2 Jets

Collimated hadronic particle showers, which can originate from quarks or gluons produced in collisions, are referred to as jets. These partons accumulate to bound states via hadronisation and interact mainly via parton scattering which creates showers of particles which themselves interact and eventually decay or are slowed down by the detector material in their path. This creates a shower of partons in the direction of the travel direction of the parton the shower originated from. Due to predominantly hard initial interactions the shower development proceeds differently compared to purely electromagnetic showers. This results in a different shower profile depending on the amount of material traversed by the shower, known as lateral shower development. The lateral shower development is also strongly dependent on the kinematics of the quark or gluon from which the shower originates. To identify jets based on their shower development, determine the origin of their initialising particle, as well as jet kinematics, jets are reconstructed from topologically connected energy deposits in ECAL and HCAL calorimeter cells. Jets are expected to deposit the majority of their energy in the HCAL with a small amount of energy deposited in the ECAL. The signature in the ECAL is due to the production of π^0, which upon production decay into two photons, with hadrons depositing smaller percentages of their energy in the ECAL. Individual deposits are required to be above a noise threshold. These connected energy deposits are referred to as topo-clusters.

The sequential recombination anti-k_T algorithm [45, 46] is used for the clustering of topo-clusters to define jets. The algorithm uses a radius parameter of ΔR = 0.4, which defines the cone size in η and ϕ space given by

$$\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}. \tag{4.1}$$

The anti-k_T algorithm is used as it favours energy clusters of hadronic origin rather than clusters originating from soft radiation or energy-independent clusters in its jet reconstruction. In addition, the algorithm is particularly robust against the underlying event and pile-up, while still producing cone-shaped jets with circular base in the η - ϕ plane. The jets are also infrared and collinear safe by construction [46]. The algorithm also defines the axis of the cone, which defines the jet axis as well as the jet direction. The jet direction specifies the propagation of the jet as either upstream or downstream along the jet axis.

Calibration

Due to detector response and the dependence on event topologies, several systematic uncertainties can result in the recorded data not to reflect the true energy of the jet. Therefore, the reconstructed jets are calibrated to correct for these systematic uncertainties using the best understanding of physics. To do this, calibrations are determined using comparisons to MC simulations and in-situ corrections are applied. Not taking into account the originating vertex or residual signals from pile-up events or processes which lead to undetected energy losses, are all systematic uncertainties in jet reconstruction. These aspects are unaccounted for and therefore each require calibration. The jet calibration [47] relies on reconstructed charged particle tracks from the ID, which reduces the range to $|\eta| < 2.5$. The most important part of the jet calibration includes the application of an absolute jet energy scale (JES) factor to the jet.

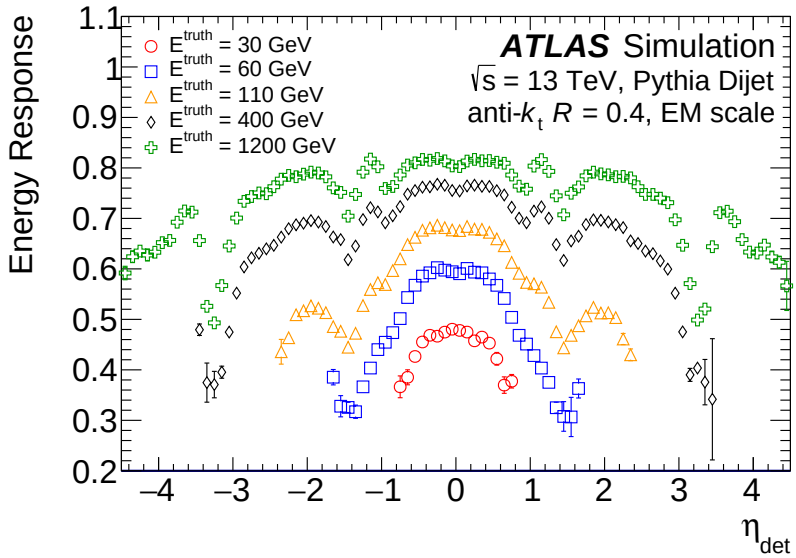


Figure 4.2: The average energy response E^{reco}/E^{truth}, which refers to the ratio between deposited E^{reco} and truth jet (E^{truth}) energy. The displayed data was derived using jets from MC simulations of di-jet events for particles of different truth energy [47]. The origin and pile-up corrections are applied. The average energy response is shown as a function of η_{det}, the η value of the jet's cone axis.

Few systematic uncertainties of the jet regarding the reference point of origin and pile-up can be addressed individually. Since the primary vertex of the event from which a jet originates is not necessarily the IP, the four-momentum of the jet might be miscalculated. To prevent this, an origin correction is applied to each jet where the primary vertex of the event is assumed to be the origin of the jet and

the four-momentum is re-calculated. The pile-up contribution is accounted for by including two corrections. A data-driven area-based p_{T} density subtraction [48] is applied alongside corrections for residual pile-up. The contribution from residual pile-up is accounted for by additional pile-up contributions to the JES.

Since the detector is not operating at an idealistic efficiency of 100%, inefficiencies in measuring the energy deposits are expected. In addition, some processes within a jet, like fission or ν production, lead to energy which escapes the measurement of the calorimeters. Both of these result in the measured deposited energy being less than its true value. Therefore, the deposited energy of jets is also calibrated to make up for this loss by using their energy deposits in the ECAL. Using this information, the jets are calibrated at the electromagnetic scale [47]. This corrects the overall recorded jet energy to the true value. As can be seen in Figure 4.2, MC simulations show that the calibration depends on the jet kinematics. Biases in the jet η reconstruction are also taken into account in the calibration to the electromagnetic scale. A global sequential calibration [47], which uses measurements from the calorimetry system and MS, as well as reconstructed tracks, removes residual dependencies from the JES. As a final calibration step, in-situ calibrations are sequentially applied, each tailored for individual jet observables, using well-known reference objects.

Quality Requirements

Quality requirements applied to the calibrated jets used in this thesis include that they are located within $|\eta| < 2.5$ and have $p_{\mathrm{T}} > 20$ GeV. To reduce the amount of pile-up jets in the event, a k-nearest neighbour finding classification algorithm is trained on hard-scatter and pile-up jets from MC simulated di-jet events in the effort to construct a discriminant called the Jet Vertex Tagger (JVT) [49] to reduce pile-up contamination.

Both the number of primary vertices in an event and track-based variable information are combined in two parameters where pile-up is taken into account as well as the fraction of tracks originating from hard scatterings processes. The algorithm is run within the two-dimensional plane of these two variables to provide a separation which is robust to changing pile-up conditions and suited to hadronic showers. The two emerging variables, which make up this plane, are used in the construction of a two-dimensional likelihood which provides the relative probability for a jet to be a hard-scatter jet at each point in the plane. This two-dimensional likelihood, called the JVT, is used to filter out the pile-up contribution by applying a veto on low p_{T} jets. Jets with $p_{\mathrm{T}} < 60$ GeV and $|\eta| < 2.4$ are required to have a value

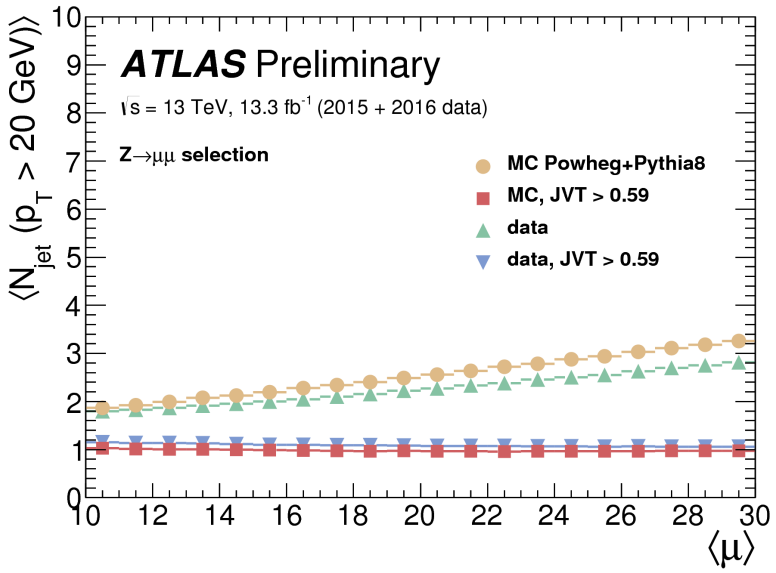


Figure 4.3: The average number of jets with $p_{\mathrm{T}} < 20$GeV as a function of different pile-up conditions for both MC simulations and data [50]. Events with exactly two muons were selected. The yields for both simulations and data where the JVT quality requirement is included show no dependence on the pile-up condition.

greater than 0.59 to pass the quality requirement. Figure 4.3 shows the impact of the JVT quality requirements on the average jet yield in simulations and data for jets with $p_{\mathrm{T}} > 20$ GeV for different pile-up conditions. As can be seen, the pile-up dependence is removed and the yields in both simulation and data are constant with varying pile-up scenarios.

4.3 Leptons and Photons

4.3.1 Electrons and Photons

Electrons and photons are reconstructed from energy depositions in the ECAL. There they are expected to deposit almost all of their energy through the creation of electromagnetic particle showers in the ECAL.

Electron reconstruction requires an electromagnetic shower in the ECAL whose axis is matched using restrictions on $\Delta\phi$ and $\Delta\eta$ to a track in the ID. The TRT provides additional particle identification information to distinguish between electrons and charged hadrons. Electromagnetic showers, where the constituents only interact electromagnetically through bremsstrahlung and pair creation, develop differently compared to hadronic showers. However, as for jets, the shower develop-

ment also depends on the kinematics of the originating particle which created the shower. Because of this, electron reconstruction starts with a selection of topologically connected clusters in the ECAL, which pass loose shape requirements [51]. These include energy distribution in η as well as restrictions on hadronic leakage in the ECAL in case of an early shower development of jets which starts already in the ECAL. This restriction is included to separate the bulk of ECAL clusters of hadronic origin from those of purely electromagnetic showers but also to correct the energy deposits for leaking jets. Clusters which pass these requirements are grouped into Regions of Interest (ROIs). Each ROI is matched to a track from the ID. If this fails, different methods using Kalman filters can be applied to recover track candidates for the clusters. Electron reconstruction then requires tighter requirements on $|\Delta\phi|$ between the seed cluster and the matched reconstructed track, which tightens from an upper limit of 0.2 to 0.1 [52]. Ambiguities arising when matching tracks to clusters are solved by a track categorisation, which ranks suited tracks higher than less suited candidates based on structure arguments. The score used in the rankingădepends on which semiconductor layer the track first recorded a hit and which track has a better ΔR matching to the clusters.

Within the photon reconstruction a distinction is made between converted and unconverted photons. In case no track can be associated with the electromagnetic shower recorded in the ECAL, this shower is assumed to originate from an unconverted photon. The TRT again also provides information to help with solving of ambiguities.

Converted photons are referred to as photons which created an e^+ e^- pair within the volume of the ID. This leads to a signature of two electrons with their trajectories close to each other, which might be misinterpreted with the signature of electrons if the tracks are not both individually recognised. A conversion reconstruction is performed using the ID tracks to find a conversion vertex which has the signature of a massless particle decaying into two charged electrons. Information on the particle identification is provided by the TRT and is applied as an additional quality requirement to achieve higher purity for converted photons.

4.3.2 Muons

Muons are identified within the range $|\eta| < 2.7$. Their reconstruction is performed using measurements from the MS, ID and the calorimetry system [53]. The main difficulty in their reconstruction is the material density between the IP and the MS, cavern backgrounds and local variations in the magnetic field strength across the MS. Due to the large range of subdetectors and their respective spatial separation, as

well as large material density in between, the reconstructed muons are categorised based on which subdetector systems contributed to their reconstruction. Dependent on their categorisation, the reconstructed muons have varying degrees of purity, momentum resolution and systematic uncertainties. The categories they are grouped into are stand-alone (MS), combined (CB), segment-tagged and calorimeter-tagged muons. In case matching information is available in both the ID and MS, CB muons are reconstructed using the combined measurements of these subsystems. Tracks in the MS are matched to tracks in the ID in η and ϕ and subsequently refit using all hit information. This provides the best momentum resolution and muon purity for the reconstructed muons. In case no track in the ID can be matched to the muon candidate measured in the MS, these muon candidates are referred to as reconstructed MS muons. Their origin is only extrapolated to the beam line. Segment-tagged and calorimeter-tagged muons both rely on ID tracks. For ID segment-tagged muons, which have the lowest purity in comparison to the other types of reconstructed muons, the ID tracks are extrapolated to the MS. Due to the gap in the MS encompassing $|\eta| < 0.1$, previously described muon reconstruction fails to work. An alternative for this region is provided by the calorimeter-tagged muon reconstruction. This reconstruction requires an ID track to be matched to a minimum ionising particle energy deposit in the calorimetry system.

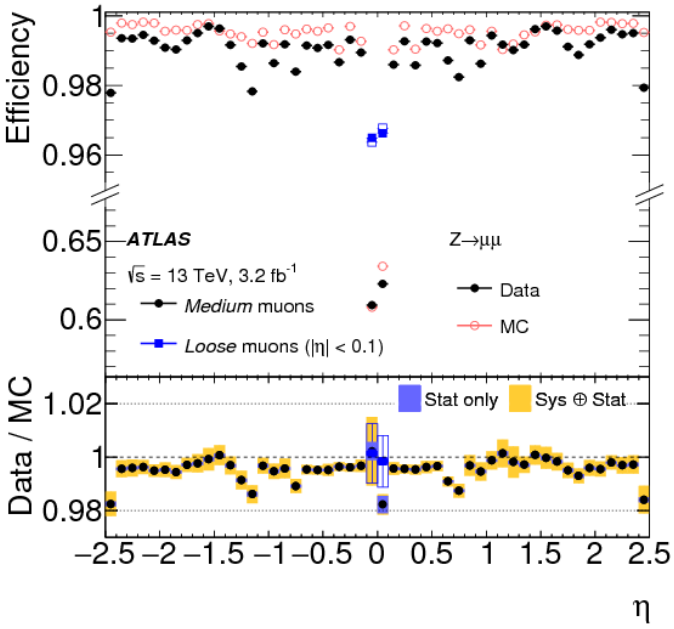


Figure 4.4: The muon reconstruction efficiency as a function of η as derived from real and simulated $Z^0 \rightarrow \mu\ \mu^+$ events [54].

All reconstructed muons are labelled with a quality depending on which kind of

reconstruction they originate from as well as their kinematic properties. Four quality requirements are defined. Muons which pass the loose quality requirements have good quality tracks and maximise the reconstruction efficiency. Muons of medium quality are selected only from MS and CB muons and include quality criteria to provide a reconstructed muon selection which minimises the systematic uncertainties associated with muon reconstruction and calibration. The reconstruction efficiency for muons with loose (only for $|\eta| < 0.1$) and medium quality requirements is shown in comparison to predictions from MC simulations in Figure 4.4. A varying η dependence can be seen, which exhibits a large reduction in reconstruction efficiency of medium quality muons due to services requiring the gap in the MS. The acceptance of the reconstruction category is even further narrowed in the tight muon selection, which only accepts reconstructed muons from CB muons, whose tracks include hits in at least two chambers of the MS and pass the medium selection criteria. This leads to a loss in reconstruction efficiency but maximises the purity of the reconstructed muons. The final quality requirements are specialised on high-p_{T} muons. For the high-p_{T} muon selection, an additional constraint of $p_{\mathrm{T}} > 100$ GeV is added on top of the tight selection requirements but instead of hits in two chambers, the tracks are required to have at least three hits in three MS chambers. This makes the high-p_{T} muon selection most suited for searches for the W$'$ and Z' heavy gauge bosons as the selection maximises the momentum resolution for high-p_{T} muons.

4.3.3 Taus

Tau leptons can either decay leptonically or hadronically. However, since leptonically decaying taus would decay into either a muon or electron and associated neutrinos, with the neutrinos escaping detection, these are misidentified as muons or electrons. Therefore, attempts are only made to reconstruct hadronically decaying τleptons (τ_{had}) and their fake rate is very high. Their reconstruction method is based on a Boosted Decision Tree (BDT) classifier. The BDT uses track and topo-cluster information as input and provides a classification discriminant which separates true τ_{had} from candidates which do not originate from τ_{had}. Only jets which contain either one or three tracks within a cone of $\Delta R < 0.2$ with $|\sum_{i_{track}} q_{i_{track}}| = 1$ are considered in this classification. This restriction is due to the number of charged pions in the decay products as most τ_{had} decays are either one or three pronged processes. Jets which are classified as τ_{had} candidates are required to have $p_{\mathrm{T}} > 10$ GeV and be within $|\eta| < 1.37$ or $1.52 < |\eta| < 2.5$, vetoing the barrel-end-cap transition region [55].

4.4 Missing Transverse Energy

All reconstructed objects described above are taken into account when calculating the missing transverse energy ($E_{\mathrm{T}}^{\mathrm{miss}}$) in an event, given by

$$E_{\mathrm{T}}^{\mathrm{miss}} = -\sum_{n=1}^{\mathrm{N_{objects}}} p_i \sin(\theta_i).$$

An additional soft term is added, which reflects deposited energy which is not accounted for by other objects in the event. This term is only calculated from ID tracks with $p_{\mathrm{T}} > 400$ MeV which are associated to the reconstructed event vertex, in order to be more robust to pile-up. The chosen tracks are additionally required to pass vertex association cuts of $d_0/\sigma(d_0) < 2$ and $|z_0 \sin(\theta)| < 3$ mm [56].

5. Machine Learning for Supervised Classification

Contents

Since this thesis focusses on the application of a classification algorithm in particle physics, this chapter provides the concepts which these applications rely on. By starting with the introduction to the general standard procedure in the development of a supervised algorithm for classification using machine learning and defining classification, the foundation for the presented approaches is build.

In the following sections of this chapter, only the concepts of supervised learning of classification algorithms on objects, where the class association is well known, will be discussed. Details of specific applications are addressed at a later point. When considering potential solutions for classification problems, an introduction to decision trees for classification will be given first. Then, the main focus of this chapter is building up the understanding of a few of the concepts to build and train a deep neural networks. This is done by starting from simple structures and moving on to more advanced and complex networks as well as optimisation strategies and algorithms. It should be noted that this chapter only aims at a brief overview of the concepts used in this thesis and for more in-depth explanation the author refers to the relevant literature which this chapter is based on [57, 58].

5.1 Introduction to Classification

A classifier provides predictions for an object to belong to pre-defined classes of interest. In mathematical terms, a classifier is supposed to predict an estimate $p(y|\mathbf{x})$ for a previously unseen object to be of class y based on a set of n-dimensional input data $\mathbf{x} = (x_1, x_2, ..., x_n)$. In case of multiclass problems, y is of the form (y_1,..,y_{N_c}), with N_c denoting the number of classes under consideration.

The theoretical construct to perform the classification task can be represented as a graphical model. A graphical model is a visualisation which represents the structure and information flow of a machine learning algorithm. The structure is given by individual elements, which are referred to as nodes and the information flow is indicated using arrows connecting the nodes. A node is a unit where the ingoing information flow is redirected using mathematical functions. The arrangement of individual nodes is referred to as the architecture of the model. The chosen model itself determines the nature of the mathematical operations performed on a node. This can in the cases discussed in this chapter be a decision tree, where the node is a decision boundary on attribute values to determine the sample propagation, or a node in a neural network which propagates a modified value. During the training of the model, the mathematical formulation is tuned using an optimisation algorithm to optimise a predefined function for the model, which is referred to as the objective function. Since the arrows indicate the information flow, these models are to be read following the arrows rather than following a strict left to right or top to bottom convention and for different models it is chosen what seems more intuitive and better suited for the intended representation according to personal preference.

For a reliable classification performance of the model, it needs to be trained on a representative dataset. It is therefore important to have a training set which captures the features which are representative for the objects of interest. In addition, the model is required to be able to capture the important features of the training dataset and generalise to unseen data while being robust enough to minor variations due to noise in the training dataset. Those features can include direct correlations between class and individual input variables but also underlying complex correlations which feature subtly in the input variables. If the chosen training set is not representative of the full spectrum of data, which is to be tested on, the performance or the model does not generalise enough or simply learns features, which are very specific to only objects in the training set, the performance will be worse on unseen data and the model is referred to as being overtrained as the model is considered to overfit on the training data.

User defined parameters which define the model but are not learned by algorithms are referred to as hyperparameters θ of the model. Their parametrisation of the model turn the previously mentioned estimate $p(y|\mathbf{x})$ into $p(y|\mathbf{x};\theta)$. The more largely uncorrelated attributes a set of samples has, the higher is the dimensionality of the problem. With increased dimensionality, the algorithm can become more complex to determine a class separation in a higher dimensional space. Therefore the number of considerations for the architecture design of the model and the amount of related free parameters increase. This includes both learnable parameters, which the model adapts during optimisation such as the connection weights between nodes, as well as hyperparameters fixed by the choice of architecture and optimisation settings. This large increase in possible hyperparameter space is also known as the curse of dimensionality. Its challenges include the restriction of the model dimensions with respect to the available statistics, as well as the the potential impossibility to cover the entire hyperparameter space in the optimisation process due to time or resource constraints.

The optimisation of the objective function can either mean to minimise or maximise its value. When minimised, it is referred to as loss or error function. The set of objects used for training and performance evaluation are required to be statistically independent and mutually exclusive. Next to the objective function, another quantity in monitoring the prediction optimisation is the accuracy of the predictions made by the model. The accuracy of a model reflects the numbers of mistakes in the form of boolean losses of the predicted class value in comparison to the original label value and provides the average misclassification over all classes based on those boolean values. It is dependent on the task whether this quantity is considered to be meaningful alongside the objective function and other figures of merit.

An important aspect in finding an optimal classifier is the ability of the model to generalise well to unseen data which might be slightly different. It can be seen as a robustness to small variations where the model is expected to still provide meaningful predictions. These small variations can include noise in data generation, reduced impact of missing data or variations between simulated and recorded physics data. Methods which perform modifications to the optimisation of the learnable parameters of the model are referred to as regularisers. Those regularisers aim at increasing the models capability to result in less drastic output changes given small variations from the expected node inputs or to generalise. This in its correct application results in less over-fitting on the training data.

The performance is commonly investigated using Receiver Operating Characteristics (ROC) curves where the background rejection is shown against the signal

efficiency. The area under the ROC integrated from a specified signal efficiency up to 100% signal efficiency is often used as simple one-value measure of the performance and is referred to as the Area Under the Curve (AUC) given the specified signal efficiency of interest. Overtraining therefore manifests itself in a larger performance degradation between the performance on the test set compared to the performance on the training set and can be picked up on when comparing the average value of the objective function or ROC curves and AUC values each on both sets. Depending on the model, overtraining can be prevented using different techniques but it is important to start with a balanced representation of the expected data population and train with enough statistics to fit the model and its optimisation strategy. An unbalanced dataset with respect to class representation would lead to an effect, known as the class prior problem, where the model is biased towards classes differently, which results also in the posterior of the output predictions being affected by this bias.

5.2 Decision Trees

A simple and computationally cheap way to solve a classification problem is to use a decision tree, which is especially suited for cases with statistical limitation. Decision trees [57, 59, 60], as shown in a schematic overview in Figure 5.1, are graphical models whose nodes are arranged in a directed tree structure. The tree starts with a node which has no parent and is referred to as root node. Each node element is a test on a given attribute, which results in a boolean outcome, which results in the sample being propagated in one of two directions. This can be described as cutting on a variable of choice. Therefore, decision trees are binary decision making tests, which split the results into two branches which each connect to a child node. This means that, except the root node, all other nodes of the tree have exactly one parent. Nodes which have no children are called leafs and their total number in the tree determines the complexity of the model. Each leaf can have a different population of signal samples versus background samples, which is determined by the hyperparameter optimisation strategy. The training of the algorithm divides the total input space into regions. Each region is associated with a class. The decision tree classifier can therefore be considered to provide an association of classes by defining collections of multidimensional rectangular cuts belonging to predefined classes.

The determination of the width and depth of a decision tree as well as which variable to test on at each node and which value to choose is part of the model optimisation. Specialised algorithms train these models. For these models regularisation

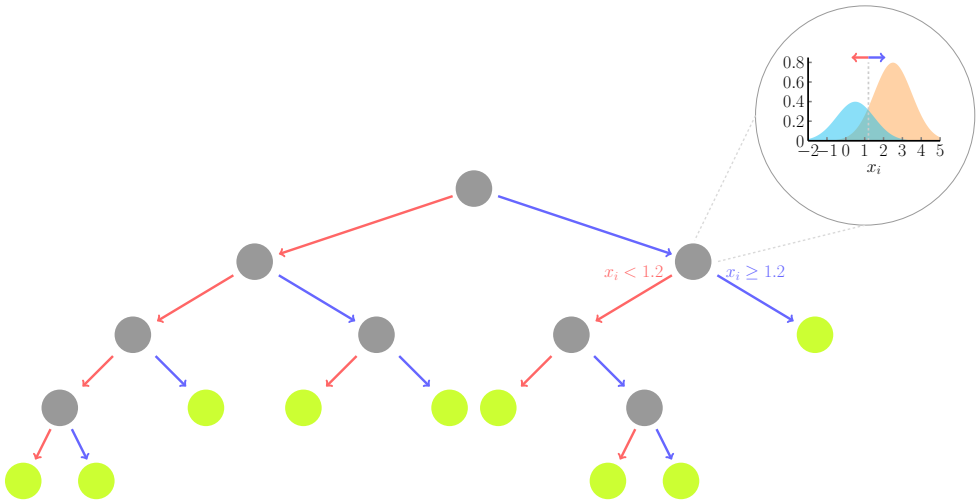


Figure 5.1: Schematic overview of a decision tree with a zoomed-in visualisation of a decision boundary at a single node, where a simple cut on an individual attribute of a sample determines the further propagation of the sample within the tree structure along the arrows. Grey circles represent individual decision boundaries and green circles represent leafs. The arrows indicate the possible categorisation paths a sample could be propagated depending on the individual decision boundaries along the path of propagation and the values of the attributes of the sample. The leaves do not include further decision boundaries but represent ends of the chain of cutting on individual attribute of a sample.

techniques as mentioned in the previous section include constraints on the tree size, which parametrises the model. Criteria can be defined as to when a node becomes a leaf. Those criteria can be based on the purity of the test outcome or minimum percentage of data points it handles. The complexity of a decision tree can be given by the number of leave nodes it contains. It is common practice to first grow the tree and prune it afterwards to reduce the complexity of the model.

Since the χ^2 performance measure has several pitfalls when applied in classification tasks, other performance measures are preferred for the use of decision trees. These measures of performance for each node τ for growing the tree are the cross-entropy given by

$$Q_\tau(T) = \sum_{k=1}^{K} p_{\tau k} \ln\left(p_{\tau k}\right) \tag{5.1}$$

or the Gini index, which is a measure of purity, given by

$$Q_\tau(T) = \sum_{k=1}^{K} p_{\tau k}\left(1 - p_{\tau k}\right), \tag{5.2}$$

where $p_{\tau k}$ denotes the proportion of samples, which node τ assigns to class k, which is either signal or background [57]. Both of these performance measures are differentiable, which means are coped better with in the application of gradient based optimisation algorithms.

To achieve generalisation of the model, overtraining can be prevented to some extent by applying constraints to the complexity of the tree as not to exceed the available statistics. One way this can be implemented is by setting a maximum depth or width of the tree.

The advantage of these models is that they are easily interpretable as they represent a sequence of binary decisions for each leaf. However, the separation of the feature space into categories with boundaries which are aligned with the axes of the feature space, which is referred to as hard cuts on the attributes, is one of the models weaknesses. The separation only happens via linear decision surfaces rather than a more flexible function, even if it was just a linear fit as in linear regression. Such problems can be captured by a decision tree, but only at the cost of increased complexity by modelling steps via series of linear cuts.

5.2.1 Boosted Decision Trees

Another way to improve the performance of decision trees is to apply boosting, which leads to the transformation of the decision tree into a BDT. Boosting is a

technique, which implies sequential optimisation, which boosts the performance. This means that boosted decision trees expand the formulation of individual decision trees into a cascade of decision trees, which are referred to as forests. Trees are stacked upon each other, each profiting of the history of previously trained trees, which results in the performance boost. In each of the sequential iteration, each sample is associated a relative importance in a given round of the training.

A weak learner is referring to an algorithm which produces a hypothesis which is only slightly better than a random guess. It is distinguished from a strong learner, which represents an algorithm whose output hypothesis is of arbitrary accuracy, which is more desirable as it reduces complexity of the model. In the context of decision tree classification, a decision tree is a weak learner until its test is adjusted enough for the resulting hypothesis to provide a good separation.

Algorithms which turn a weak learner quickly into a strong learner are referred to as boosting algorithms and a decision tree using them is known as a BDT. Weak learners are iteratively trained and then added to a strong classifier using a weighing scheme for training data or hypotheses. Boosting algorithms only vary in their weighing implementation.

The Adaptive Boosting (AdaBoost) algorithm [61] aims to minimise the error function in a BDT by adjusting adaptively to the outcome of the objective function between iterations of training. Each subsequent training starts using the weights from the end of the previous training but uses weighted events based on their classification accuracy in the previous training step. By weighting the inputs, the hard cuts are softened. This method does not require prior knowledge on the accuracy of the weak hypothesis. In the end, an ensemble of weak hypotheses is retained. Each hypothesis is weighted according to its accuracy. The final output is provided by summing their individual probabilistic predictions in a weighted majority vote on the weighted outputs of the trained decision tress. The AdaBoost algorithm is a popular choice for training decision tree models. Its success is largely based on empirical results and its formulation provided a theoretical foundation for previous empirical results.

Checks for overtraining are performed based on the figures of merit. By comparing the ROC curves for the performance on the training with the curve calculated on the test set, the state of overtraining can be inferred. Subsequently, the AUC is calculated for both. If the AUC and the ROC curves do not match and the performance based on the figure of merit on the training set largely exceeds the test set, the BDT is likely overtrained.

5.3 Neural Networks

A neural network is similar to a decision tree also a graphical model with a directed tree structure. In this instance however, the nodes it contains are fundamentally different from those used in decision trees. This difference is that they propagate individual values instead of making a binary decision on the categorisation of a sample. This results in the nodes being able to in principle have an infinite number of inputs but only one output, which can connect to an infinite number of nodes by infinite connections.

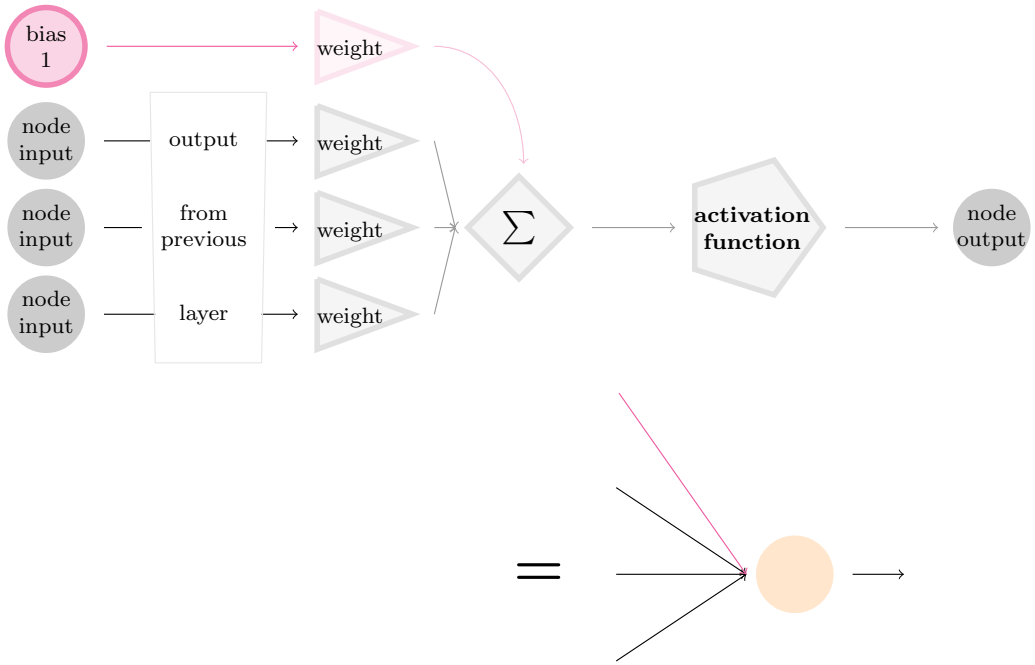


Figure 5.2: Schematic overview of an individual neural network node [2], which makes up the basic node element in a Neural Network. Output values from the ingoing nodes, denoted as inputs to the node of interest, as well as the contribution from the bias node of the previous layer are each multiplied with a connection weight and then summed up. An activation function is applied to this sum value. The output after applying this activation is the output of the node and its value is propagated further.

The architecture of a neural network consists of nodes which are arranged in a three-fold layer structure of a single input, potentially multiple hidden and a single output layer. Each visible and hidden layer includes a bias node of constant value, which is typically one. This contribution in combination with the individual

connection weights to immediately connected nodes introduces a way to shift the activation function individually in each connected nodes. Every node of a hidden layer is connected to all nodes of the previous layer, including the corresponding bias node. Input nodes, which together with its bias node compose the first layer of a neural network, are referred to as visible nodes. All nodes between the input and output layer are known as hidden nodes. If a network only consists of one single layer, it is referred to as being shallow. The condition for a neural network to be deep only requires it to contain more than one hidden layer.

At a typical simple neural network node, the values provided by the connections to all nodes of the previous layer are multiplied by associated connection values, which are referred to as weights. The resulting values are subsequently summarised to provide a linear transformation of all previously available data. Figure 5.2 shows the individual steps performed at such a typical node. After summation of the inputs to the node, an activation function is applied to the resulting value in order to prevent excessive saturation which would result in a vanishing gradient during backpropagation which would reduce the learning effect. For this, different properties of various activation functions can be exploited for different tasks and purposes. However, if this function were linear, the network would only be able to represent linear functions. Therefore, these activation functions are preferably non-linear to provide a network the ability to represent any possible function with minimal additional complexity, meaning limited additional number of learnable parameters for an approximately similar result. The value retained after applying the activation function is referred to as activation and is the output value of the node.

The inputs are desired to be as uncorrelated as possible. It is assumed that the optimiser of the network will be able to learn the best correlations for the classification during training when optimised using a suitable set of hyperparameters. For individual input variables not to dominate these processes from the beginning, each distribution for each input variable is normalised. By assuming a gaussian distribution of the input variables, the distribution is shifted and scaled to have zero mean and a variance of one standard deviation. In addition, the values of the weights, which are used in the connection of nodes, need to be initialised to small but non-zero randomly chosen numbers.

Next to the design of the Neural Net (NN), the optimisation strategy as well as input preparation determines the model and its success. Hyperparameters are settings which can be varied to alter the models performance. The architecture of a model refers to the bare bones building blocks whereas the training is defined by the optimisation algorithms as well as the assisting algorithms like regularisers.

The number of learnable parameters of a model is used to specify its complexity and it is common practice to compare a models complexity to the number of available statistics when building a model not to construct a model which is likely to overtrain quickly as not enough samples are provided to learn all learnable parameters of the model, meaning if the number of learnable parameters of the model exceeds the number of training samples. It comes down to an intricate problem with multiple aims which depends on a large number of possible hyperparameter values. The collectivity of this is referred to as the hyperparameter space, which represents the space of all possible settings one could choose from.

A validation dataset, which is lower in statistics and typically of the order of about 5%-20% of the training statistics, is split from the training set, depending on the classification task and available statistics. When using the validation set to calculate the loss, this disjoint split provides an estimator on generalisation capabilities and overtraining of the model on the training set.

5.3.1 Multi Layer Perceptron

A Multi Layer Perceptron (MLP) is a neural network with at least one hidden layer, where the layers only consists of simple fully connected nodes and are known as dense layers. A schematic overview of the architecture of a typical MLP is shown in Figure 5.3, where only dense layers are following upon each other and form a shallow neural network, which consists of only one single hidden layer.
Non-linear activations are applied to the summed node inputs but the activation functions themselves are typically just sigmoid activation functions given by

$$h(z) = \frac{1}{1 + e^{-z}}, \tag{5.3}$$

which is also known as the standard logistic function. The distribution of typical sigmoid activation functions is shown in Figure 5.4a. Output layers typically use the same sigmoid activation function or a employ linear activation functions of the form $h(z) = az + b$ with a and b being constant parameters.

A drawback regarding the sigmoid function is that it can be approximated at the origin by a linear function. It might be converging slow compared to other function choices.

The output value of a node at position j is computed as weighted sum of its inputs from the previous layer at position i given by

$$a_j = \sum_i w_{ij} z_i, \tag{5.4}$$

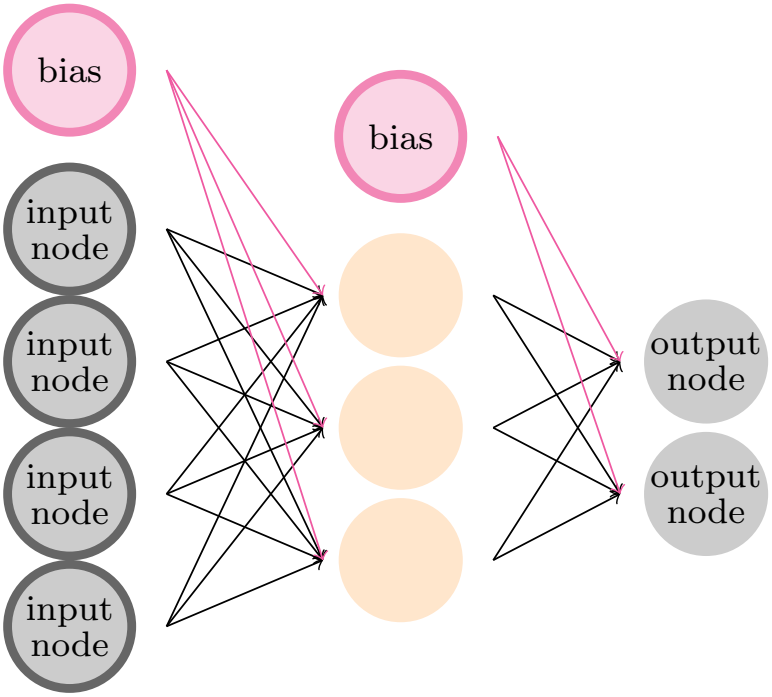
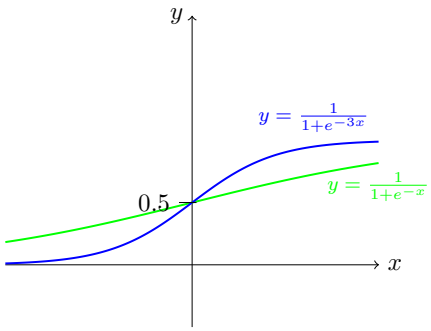
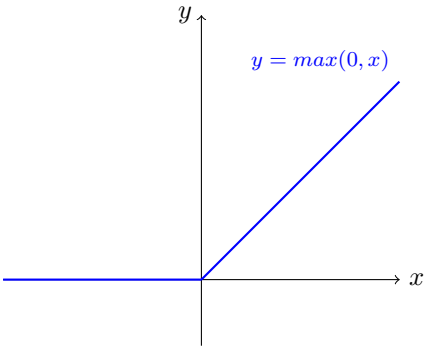


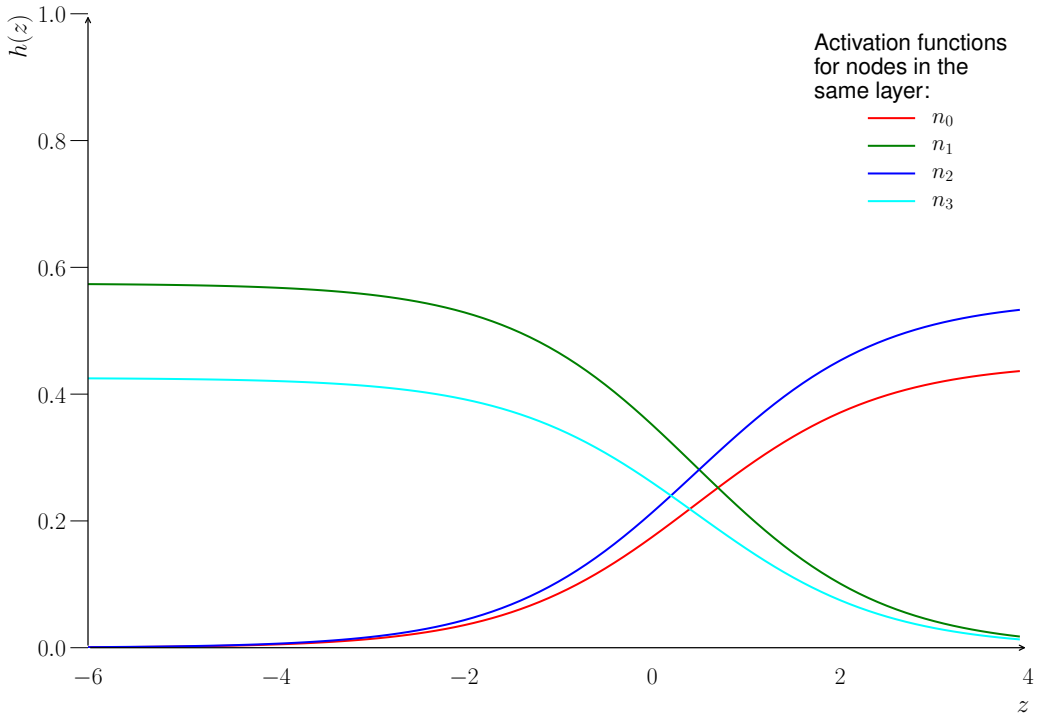
Figure 5.3: Schematic overview of a MLP. Since it only consists of one single hidden layer in this representation, it is referred to as being shallow. Framed nodes provide a constant value to the nodes they are connected to.



(a) Sigmoid



(b) ReLU



(c) Softmax

Figure 5.4: Visualisation of examples of the sigmoid (top left), ReLU (top right) and softmax (bottom) activation functions. For the softmax function, the contributions from four nodes (n_0, ..., n_3) are summing up to unity at any point of the distribution, which transforms the outputs of the nodes into probability values with feasible lower and upper bounds.

using factors w_{ij}, which are referred to as weights. The final output of the node, referred to as z_j, is given by

$$z_j = h\left(a_j\right), \tag{5.5}$$

which provides the activation of a node in the previous layer for a given activation function h. Bias nodes have a fixed activation of +1. Following these two equations, information is propagated forwards through a simple neural network, a process which is known as forward propagation and shown in Figure 5.2. However, with the weights w_{ij} initially defined, no learning takes place and the predicted values do not change.

A neural network learns by the means of the updates of its internal weights w_{ij} which connect individual nodes. The weight update ∂w_{ij} is calculated according to the weight updating formula given by

$$\frac{\partial E_n}{\partial w_{ji}} = \delta_j z_i, \tag{5.6}$$

with δ_j representing the update of a node, referred to as error, which represents the changes to a given node given its weighted connections to the layers closer to the output based on the estimated error E on the label values. The value of the error δ_j is calculated using the backpropagation algorithm, which is given by the backpropagation formula defined as

$$\delta_j = h'\left(a_j\right) \sum_{k=1}^{K} w_{kj}\delta_k \tag{5.7}$$

with k the labelling referring to the subsequently connected nodes and $h'(a_j)$ the gradient of the activation function of a_j. These updates ∂w_{ij} are performed backwards, starting with the calculated error with respect to the label on an output node and subsequently calculating the update for each node in the layers closer to the inputs layer per layer as shown in Figure 5.5. This process is repeated until all weights are updated.

When training in batches, the weight update is only performed for the summed error per batch. This reduces the otherwise hard impact of outlier which might disrupt the learning process.

5.3.2 Deep Neural Networks

A simple deep neural network consists of multiple hidden layers. Due to the depth of these networks the main complication is in training them as their depth results in vanishing gradients during backpropagation.

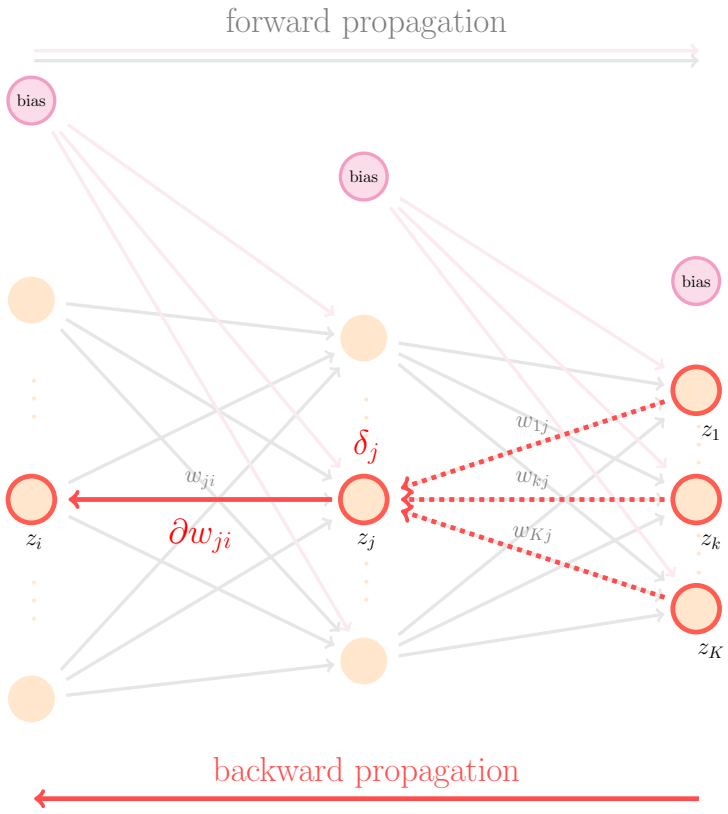


Figure 5.5: Schematic overview of the application of the backpropagation algorithm in a zoomed-in and cut-away view of a deep NN showing three subsequent hidden layers. The backpropagation formula 5.7 is applied on the nodes to calculate the error on node z_j, which is denoted by δ_j, based on all nodes which are directly connected to this node in the forward propagation step, denoted by the dashed red arrows. Then the weight update ∂w_{ji} can be calculated using Equation 5.6 to be applied to the forward propagation weight w_{ji}, which connects the node z_i to the node z_j in the following layer.

Activation Functions

The activation function which is applied can vary per layer.

The Rectified Linear Unit (ReLU) function given by

$$h\left(a\right) = \max\left(0, a\right) \tag{5.8}$$

features a piecewise linear function of two connected linear pieces. As shown in Figure 5.4b, its non-linearity is expressed by a discontinuity between its two linear pieces at the origin. What makes a ReLU function a widely used function is its speed due to its simplicity of its gradient being a stepfunction, which speeds up computations, and its compatibility to gradient based optimisers.

In order to protect against non-physical under- and overflow of the output values, an output probability distribution is chosen as output layer activation function. This choice enforces feasibility, meaning the output will be within the range of zero to one and the sum over all output nodes per sample sums up to one. The activation function for the output layer of K output nodes is chosen to be a softmax transformation function given by

$$h\left(a\right) = softmax(a) = \frac{e^{a}}{\sum_{k=0}^{K} e^{a_k}}. \tag{5.9}$$

As is shown for an example of $K = 4$ output nodes in Figure 5.4c, the summed contributions of all output nodes equals unity across the full spectrum of inputs.

The nature of the loss function $J(\theta)$ is chosen based on the classification problem. For multi-class classification the categorical cross-entropy given by

$$J\left(\theta\right) = \frac{1}{N}\sum_{n=1}^{N} H(p_n, q_n) = \frac{1}{N}\sum_{n=1}^{N} \left[y_n \log(\hat{y}_n) + (1 - y_n)\log(1 - \hat{y}_n)\right] \tag{5.10}$$

with p the predicted values and q the provided original label values is the most obvious choice as it includes the relative entropy $H(p_n, q_n)$ between the predictions per individual output nodes denoted by n. The predictions per node are given by y_n and the original label values per node by $\hat{y}_n$. It depends on the hyperparameters of the model and the aim is to find the set of hyperparameter which minimises the loss. The optimiser algorithm minimises the loss for a given set of hyperparameters.

One approach to minimise the loss is using gradient descent algorithms. The loss is estimated for a small change around the current value by calculating the derivative of the loss. Using the directional derivative, the highest gradient is determined and the update is performed in the direction of the negative gradient. During

backpropagation the output nodes are therefore adjusted by to x', which replaces its previous value x. The updated value x' is given by

$$x' = x - \epsilon \frac{\partial f(x)}{\partial x}, \tag{5.11}$$

where ϵ is the learning rate, a hyperparameter which defines the overall size of the update.

To accelerate the learning progress, the number of floating point precision calculations can be reduced by applying this update after calculating the loss on randomly chosen minibatches. This is then referred to as the Stochastic Gradient Descent (SGD) algorithm. Since the minibatches are composed randomly, no gradient bias is introduced on average and outlier are less likely to direct the updates into a corner of the hyperparameter space by introducing large updates for individual samples. Since the gradient decreases upon approaching a minimum, the learning rate ϵ can be chosen as a constant.

To speed up the learning process even further, optimisation algorithms adapted the concept of momentum $\vec{p} = m\vec{v}$ from Newtons second law of motion from classical mechanics. The concept obtains its analogy by treating the negative gradient in the hyperparameter space upon which the change in hyperparameter space is based as the force. The current position in hyperparameter space is thereby pictured as a ball of unity mass whose future motion is not only influenced by the force at any location but also the velocity at those points, which lets the algorithm take into account the history of the gradient change. A hyperparameter α, whose values range from zero to one, acts thereby as a sort of friction coefficient which dampens the initial velocity at each update. It should be noted though that the dampening effect is higher for lower values of α and gets smaller as it approaches one. The hyperparameter α refers to how fast the contribution from previous gradients vanishes. By giving weight to the alignment of gradients in the update, a minima can be found quicker.

A variation of the SGD algorithms which use momentum is the ADAptive Moments (ADAM) algorithm [62]. It uses first and second order momenta which are weighted by the gradient to find a minimum. The algorithm also includes bias corrections with respect to their initialisation for those momenta.

A default weight initialisation choice is to assign weights to a random small value. A method which is often used for this is to draw values from a Glorot uniform distribution [63]. This distribution is a uniform distribution within the range

$$\left[-\frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}}, \frac{\sqrt{6}}{\sqrt{n_j + n_{j+1}}} \right], \tag{5.12}$$

where n_j refers to the number of nodes in the previous layer and n_{j+1} to the nu mber os nodes in the current layer.

Maxout layers are layers where copies of a layer with varying initialisation weights are simultaneously trained in parallel. The activation per node of this 'single' layer is calculated as the element-wise maximum of the nodes which were trained in parallel. A schematic overview is shown in Figure 5.6, where

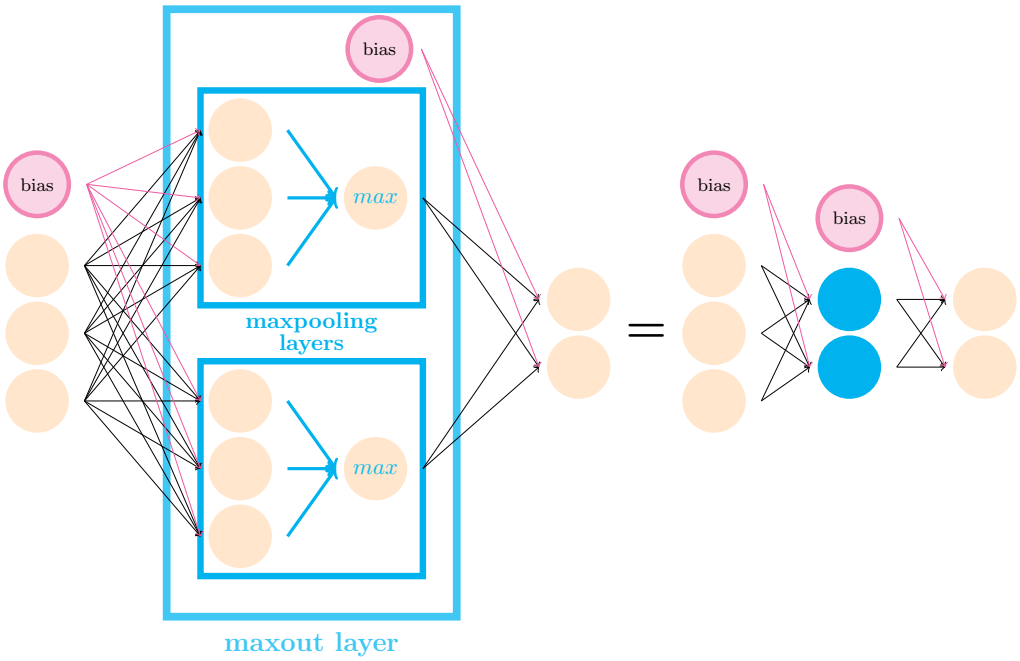


Figure 5.6: Schematic overview of a maxout layer and its definition for simplification purpose [2] for further reference throughout this thesis. Within the maxout layer, layers are trained in parallel and only the element-wise maximum output value, which is calculated within the element-wise maxpooling layers, is propagated to the next layer. For simplicity in the visualisations, individual maxpooling layers are denoted as blue nodes throughout this thesis.

Dropout [64] is a regularisation technique which prevents overfitting of the NN and effectively samples over reduced versions of the model. During the training on each mini-batch, hidden and visible nodes of the NN are randomly chosen to be temporarily dropped. The amount of total nodes to be dropped at each iteration is specified as by a hyperparameter and the dropping of the node is often effectively done by effectively masking the node and setting the output of it after application of the activation function to zero. This results in each node being trained more independent on the other nodes. Since this results in fewer floating point operations during training, it speeds up the training as well. However, this technique only affects the

training phase. During testing only the complete NN with all nodes included is used. This can improve generalisation capabilities of the NN as it would not always pick up on all possible correlations.

During the training each layers input changes after each update. Since the nodes input expectations keep changing whereas the performed update was performed on the assumption of an outdated input, this makes it difficult for the weights to adapt to a value suited for its inputs. This unstable behaviour requires fine tuning of the learning rate and other parameters of the NN as not to shift the weights too much at once. This problem is known as the internal covariate shift.

Batch Normalisation [65] applies a Batch Normalising Transform to each node activation input where a scale and shift parameter is learned from each mini-batch. The means and variances of the inputs are shifted such that the input distribution appears more stable with an expected value of 0 and a variance of one standard deviation, which effectively reduces the internal covariate shift. By having the activations in more defined ranges, the learning process is taking place on a stabler foundation, which can result in faster converging of the network training. It also reduces the dependence of the training outcome on the initial weight initialisation and acts as a regulariser while maintaining the capacity of the NN.

5.3.3 Performance Optimisation

As stated by the curse of dimensionality, the complexity of having a large hyperparameter space from which to optimise the set of hyperparameters results in multiple challenges. It is therefore a common approach to rather perform an approximate optimisation than finding the absolute best solution possible, as shown in Figure 5.7.

For the process of optimisation, it is possible to optimise either all or a specific subset of hyperparameters by performing a bayesian optimisation [66] based on a predefined range for those hyperparameters. Each bayesian optimisation for a parameter is then based on the settings of the others at that moment and the optimisation is performed in the initially specified sequence. Another approach which allows parallelisation of the optimisation of these hyperparameters is a grid search. This entails a systematic performance investigation of networks which were trained using a limited number of values within a limited range for each of their hyperparameters. This brute force approach allows to investigate the performance at any training state and compare the best results for each hyperparameter combination to see the impact of hyperparameter changes on both the objective function as well as the figures of merit.

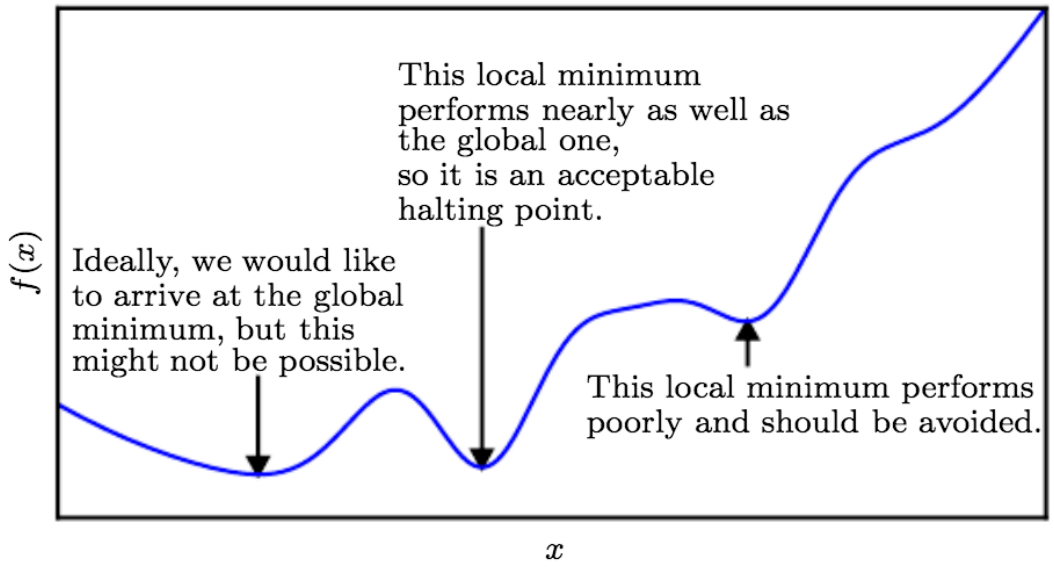


Figure 5.7: Next to a global minimum, multiple local minima in the objective function $f(x)$ are possible in the space of hyperparameters, denoted here for simplified visualisation by a single dimension variable x. Finding a well performing minimum is preferable in some cases as covering the full hyperparameter space might not always be realistic given real-life constraints. Accepting a well-performing minimum in the hyperparameterspace is known as approximate minimisation [58].

The optimisation of a Deep Neural Net (DNN) is time intensive. One not only has to optimise the architecture to best fit the problem and optimise the cost function. A DNN is represented by an intricate structure of parameter choices, which involve the placement of regularisers and their parameters as well as choices of which layer and respective activation function or which algorithm to choose. Not only is the aim to minimise the loss, but also to generalise to unseen data. Even the definition of a figure of merit can pose a complex optimisation criteria on its own. Intuition can be a guiding factor as choices are based on empirical results in the field. However, understanding the principles of what individual techniques do provides a basis to help with intuition. In the end, it is impossible to check every possible combination without infinite time and computing resources. It is not just about running the trainings but also about evaluating all relevant figures of merit. With the DNN depending a lot on the inputs and the underlying correlations, a variation in the inputs might require a new optimisation of parameters or even different architecture to perform better. With time being of the essence in a large collaboration, pragmatism regarding getting the best possible result given the time and resources available is imperative to progress. Bayesian optimisation might speed the optimisation of individual parameters up but that is not to say that it will optimise these parameters in exactly the way which is best for physics results and some aspects might be left out or averaged over. A grid search of the most important parameters based on a set of reasonable parameters allows to get into results at any step of the training and compare the loss, performance and generalisation capabilities. The decision can then be made to match the desired performance features over a large spectrum of criteria.

6. Flavour Tagging

Contents

Flavour Tagging refers to the association of jets with their originating elementary particle and recommendations are provided by the dedicated Flavour Tagging combined performance group. This chapter provides a general overview over the sample and information content considerations as well as the algorithm structure towards the final association of a jet to a certain origin.

6.1 Motivation

A jet can originate either from a gluon or a hadronic particle, which in combination with its kinematic properties determine the shower development of the jet. Due to differences in the shower development based on the originating particle, the originating particle of the jet is referred to as its flavour with the distinction between b-, c-, light-flavour and τ-jets. Of special interest for physics analyses is the association of jets containing b or c hadrons against those which are missing these constituents in their early shower development. The association is done based on morpholog-

ical properties like the internal structure of the jet. The morphological properties strongly depend on the hadronisation of the initial elementary particle as well as decay and kinematic properties of the hadrons. The signature of the jet development close to the primary vertex is of large interest and the lifetime is a good indicator for the originating particle, which created the jet formation. Due to the long lifetime of b hadrons, a secondary vertex is expected to be found in the jet. However, the lifetime of the next heaviest particle the c-quark is much reduced and therefore, a detectable secondary vertex is expected at much shorter distances from the primary vertex. This characteristic behaviour is also used in the search for top quarks decaying into b-quarks. All other lighter particles as well as gluons are expected to exhibit similar behaviour with respect to each other as their shower development does not include a secondary vertex displaced from the IP. Therefore, Flavour Tagging is considering the three distinct classes of jets originating from b, c and light-flavour quarks, referred to as b-, c- and light-flavour jets. The primary interest in Flavour Tagging is on the identification of b-jets against those of other flavours. In addition, the correct assignment of c-jets against both b- and light-flavour jets is also an important instrument provided by the Flavour Tagging combined performance group to analyses. Since more analyses rely on b-jet tagging, good b-jet tagging performance is the primary focus.

To classify the flavour of a jet, multiple variables are constructed to provide a discriminant which is derived from the relevant physics of the jet. In addition, dedicated algorithms, which are called low-level taggers, are constructed based on these variables to provide physics motivated discriminants. These are subsequently propagated to a classification algorithm, which is referred to as a high-level tagging algorithm. The low-level algorithms are based on variables which are related to either the charged particle tracks or reconstruction information of the vertices associated with the jet. The baseline high-level tagging algorithms are a family of BDTs, referred to as MultiVariate 2 (MV2) flavour tagging algorithms. One instance of MV2 is trained for the purpose of b-jet tagging and another instance for c-jet tagging, which is called MV2c(l)100. Both the low-level and high-level algorithms are described in this chapter.

Samples of simulated collision events generated with the MC method are used as labelled data to train the flavour tagging classifiers and are discussed in the following section of this chapter. Discrepancies to collision data are studied for each individual low- and high-level tagging algorithm using predictions from MC simulations. However, only the final output of the high-level tagging algorithm requires calibration to provide η and p_{T} dependent scale factors which are used to account for

these variations. The comparisons of data to predictions from MC samples as well as the calibration technique and results for the baseline algorithms are discussed in the final sections of this chapter.

6.2 Simulated Samples

A mixed sample of simulated events, which is referred to as the hybrid sample [67], is used for training and testing of the flavour tagging algorithms. The hybrid sample contains simulated events from proton-proton collisions resulting in direct $t\bar{t}$ and Z' production. The $t\bar{t}$ and Z' events are simulated separately in dedicated samples before being merged.

	$t\bar{t}$	Z'
ME generator	Powheg	Pythia 8
PS/UE generator	Pythia 8	Pythia 8
ME precision	NLO	LO
ME (PS/UE) PDF	CT10	NNPDF2.3
Additional information	at least one leptonic W decay	BR($Z' \to b\bar{b}$) = BR($Z' \to c\bar{c}$) = BR($Z' \to u\bar{u}$) = 1/3

Table 6.1: Simulation details for samples which make up the training sample. Here, u refers to any quark lighter than the c-quark.

The main sample consists of simulated $t\bar{t}$ events. This process has a high production ratio at the LHC and since the top decay has a branching ratio of $t \to Wb$ close to unity, this production process provides a b-jet sample with low background contamination. In addition, these events contain c- and light-flavour jets from $W \to q\bar{q}$ decays. These events are combined with events of an artificial Z' sample, which was produced to broaden the p_{T} range accessible to the Flavour Tagging algorithms.

The generator of the ME for the initial hard scattering process modelling as well as PS and UE generators used in the simulation of both samples are listed in Table 6.1. For both samples the A14 tuned parameter set [68] is used with Pythia 8 [21]. Both samples are overlaid with an average simulated $< \mu >$ = 24.4 pile-up events using Pythia 8 [69].

For the simulation of the $t\bar{t}$ events the ME is generated at NLO by Powheg [70]. The ME generator is interfaced with Pythia 8 to model the PS and UE, where the UE generation is restricted to those which contain at least one subsequent leptonic W decay. For the ME, PS and UE simulation for the $t\bar{t}$ events, the CT10 PDF is used.

The $Z' \rightarrow b\bar{b}\backslash c\bar{c}\backslash u\bar{u}$ events is simulated using the ME calculated at LO by Pythia 8 using the NNPDF2.3 PDF. The same generator in combination with the same PDF is also used to model the ME and UE. The cross section of the hard scattering process of the Z' events has been modified in-situ to produce a sample with a large plateau for higher p_{T} values compared to the majority of the p_{T} spectrum of the constituent jets in $t\bar{t}$ events. Therefore, by using this artificial hybrid sample, a higher number of jets across the p_{T} range for all relevant jet flavours is available for the training of the high level tagging algorithms. Within the framework of PS simulation the BR of the Z' events are artificially set to equal fractions of one third for each decay into $b\bar{b}$, $c\bar{c}$ and $u\bar{u}$ in the simulation, where u refers to light-flavour quarks u, d and s quarks. This provides a sample which is rich in jets of all flavours and therefore extends the p_{T} range for them all evenly.

Through subsequent decays of these produced particles the simulated events contain jets, which are labelled according to their particle content within a cone of $\Delta R < 0.3$ for hadrons and τ leptons with $p_{\mathrm{T}} > 5$ GeV. If a jet contains a b hadron, it is labelled as a b-jet. If it does not contain a b hadron but the jet contains a c hadron, it is labelled as a c-jet and subsequently if only a τ lepton is found within the cone, it is labelled as a τ-jet. In case neither a b hadron, c hadron or τ lepton is found, it is always labelled as a light-flavour jet.

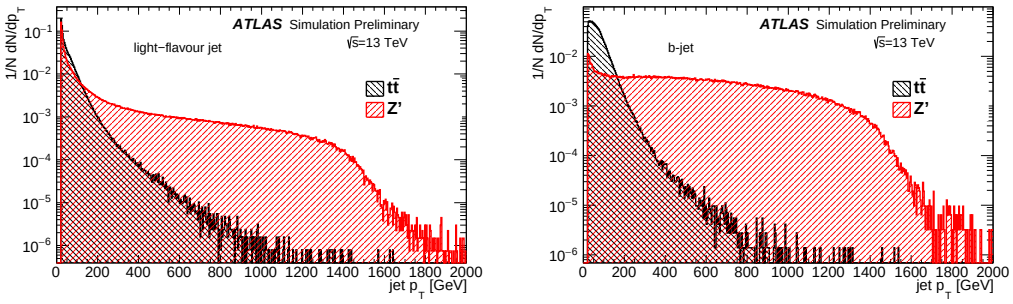


Figure 6.1: The jet p_{T} population for light-flavour (left) and b-jets (right) of the $t\bar{t}$ and Z' samples [67]. The Z' sample exhibits a plateau of enriched population up to 1 TeV for both b- and light-flavour jets.

For both samples the EvtGen simulation [23] is applied to model the subse-

quent decays of b and c hadrons. Events from both samples are also processed using GEANT [71] to include interaction effects with detector material. The physics objects in each event are reconstructed as described in Chapter 4. The distribution of the reconstructed jet p_{T} of b-, c- and light-flavour jets contained in the the $t\bar{t}$ and Z' samples are shown in Figure 6.1.

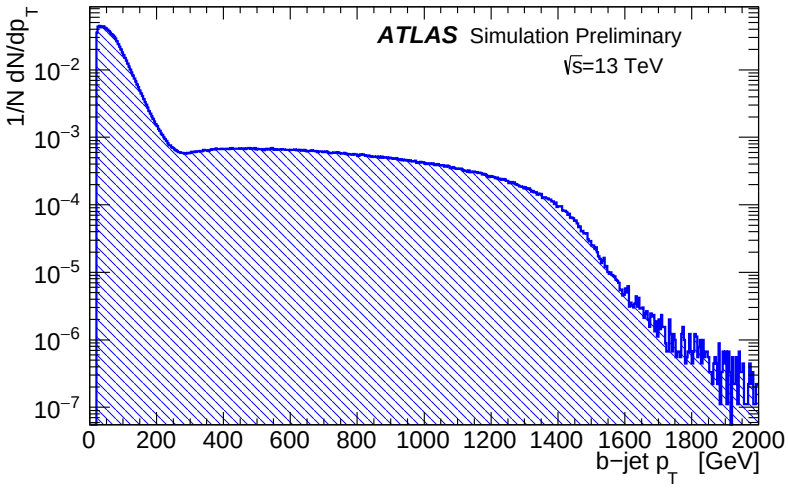


Figure 6.2: The jet p_{T} population of the hybrid sample for b-jets [67].

The hybrid sample is created using jets from the $t\bar{t}$ and Z' samples by selecting b-jets according to their b hadron p_{T} with a boundary value of 250 GeV. The b-jets whose b hadron p_{T} is below the threshold are only included from the $t\bar{t}$ sample and above the threshold, only b-jets from the Z' sample are included. The resulting population as a function of jet p_{T} is shown in Figure 6.2. For c-, light-flavour flavour and τ-jets a similar procedure is applied for the merging process, however here the p_{T} of the total jet is used instead, with a threshold of 250 GeV. Jets with lower jet p_{T} are merged into the hybrid when originating from the simulated $t\bar{t}$ process, whereas jets with higher p_{T} are chosen if they are from a simulated Z' process. The hybrid sample contains in total 5 M $t\bar{t}$ and 3 M Z' simulated events with the jet flavour population varying per jet flavour.

The hybrid sample is used primarily in training the high level tagging algorithms. The low-level tagging algorithms are either using the hybrid sample or only the $t\bar{t}$ sample as indicated in Ref. [72]. In addition to the samples used to create the hybrid training and validation sample, additional samples were generated to perform checks on the performance of the trained algorithms for other event topologies and for the predictions for all background processes used in the calibrations.

6.3 Low-Level Algorithms

The low-level tagging algorithms aim to provide variables with a well understood physics content which are robust against differences in simulated events compared to collision data. Each of these algorithms is dedicated to a specific aspect of key components to differentiate the flavours of a jet. The final list of variables provided by the low-level algorithms which are used in the high-level tagging algorithms is provided in Tables 6.2, 6.3 and 6.4.

6.3.1 Track Based Algorithms

Several low-level tagging algorithms use information of tracks reconstructed from hits in the ID. These are the IP2D, IP3D and Recurrent Nueral Network Impact Parameter (RNNIP) algorithms which largely concentrate on impact parameter information of tracks within the jet. In addition, information on reconstructed muon tracks, which are matched to the jet, provide variables of jet flavour discriminating power, which are propagated to an additional low-level tagging algorithm, the Soft Muon Tagger (SMT) algorithm.

For ID tracks to be considered to be within the jet, they need to pass a p_{T} dependent ΔR requirement, which starts with $\Delta R < 0.45$ for tracks with a p_{T} of 20 GeV. The maximum separation in ΔR becomes tighter with increasing p_{T} of the track to take into account the more collimated behaviour of constituents. These algorithms exploit in particular the long b hadron lifetime. As their charged decays are characterised by the distances of displaced vertices, which are further away from the primary vertex, larger absolute values can be expected for the impact parameters. By convention, the impact parameters have a positive sign when the primary vertex is before the secondary vertex in the jet direction as is expected for jets originating from a b hadron decay. A negative sign is assigned if the secondary vertex is located before the primary vertex in the direction of the jet. Therefore, important inputs for these algorithms are the ratios of the transverse and longitudinal impact parameter to their uncertainty, known as the transverse and longitudinal impact parameter significances S_{d_0} and S_{z_0}, which are given by d_0/σ_{d_0} and z_0/σ_{z_0} respectively.

The IP2D and IP3D algorithms [73], universally referred to as IPxD, are log-likelihood ratio based methods, which involve the lifetime information of the hadron contained within the jet. Each provide a set of IPxD discriminants, which for a jet associated to N tracks is given by $\sum_{i=1}^{N} p_b/p_{\mathrm{light-flavour}}$, $\sum_{i=1}^{N} p_b/p_c$, and $\sum_{i=1}^{N} p_c/p_{\mathrm{light-flavour}}$. A typical distribution of the variable $\sum_{i=1}^{N} p_b/p_{\mathrm{light-flavour}}$ as provided by IP3D, which is referred to as ip3, is shown in Figure 6.3. The Poisson-like distributions

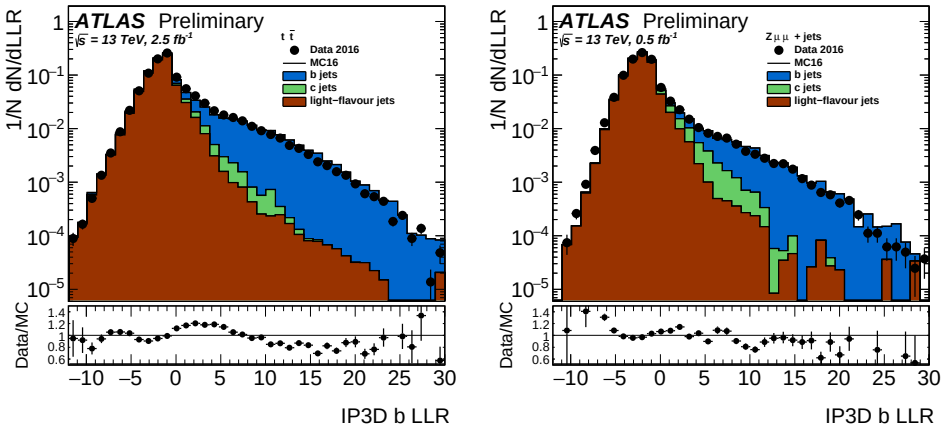


Figure 6.3: The distribution of the log-likelihood ratio variable to separate b- from light-flavour jets, which is calculated in the IP3D low-level algorithm. The variable distribution from simulated MC events is compared to pp collision data at $\sqrt{s}$ = 13 TeV. The distributions and comparisons are shown based on the calculation on the $t\bar{t}$-dominated $e\mu$ sample (left) and the muons and jet dominated Z sample (right) [67].

for different flavours show a peak between -5 and 0 within an overall range of -10 $<$ ip3 $<$30 with varying tail shapes per flavour, which reflects the capability of the algorithm to discriminate between the different jet flavours. In the definitions of the variables these algorithms provide, p_b, p_c and $p_{\text{light}-\text{flavour}}$ are defined as probability functions of the track being from a b, c or light-flavour hadron decay. These probability functions are derived template reference histograms obtained from MC simulation whose construction differs between IP2D and IP3D. In addition, each flavour probability differs depending on the hit pattern of the track, which is categorised into pre-defined hit pattern categories, which are referred to as track grade. The categorisation of the track is based on individual histograms derived from the simulation according to the hit pattern. For each flavour the probability function value for the track grade is used in the calculation of the IPxD discriminants. In the case of IP2D, the transverse impact parameter S_{d_0} is used in a one-dimensional template. For IP3D, S_{d_0} and the significance of the projection of the longitudinal impact parameter are combined in a two-dimensional template. The projection of the longitudinal impact parameter is performed on the azimuthal angle θ as seen from the primary vertex and is used to add complimentary information from the orientation of the track. By combining this information with the longitudinal impact parameter, the template dimensionality is kept low and the concept relatively simple in order to reduce the overall calculations required while adding complementary in-

formation compared to IP2D. However, by adding up the individual contributions per each track associated to the jet, the IPxD algorithms do not take into account correlations between tracks, which could provide additional discrimination power. The complete list of variables provided by the IPxD algorithms, which are used in a high-level tagger, is shown in Table 6.2.

The RNNIP algorithm [72] uses the same track based inputs as the IPxD algorithms as well as the relative momentum contribution of a track compared to the entire jet and its distance in ΔR relative to the jet axis. However, where it differs is its underlying method is based on a recurrent neural network model. It uses the associated tracks as a sequence of inputs, with the tracks arranged according to their S_{d_0} value in decreasing order. This method therefore includes the correlations between tracks associated to the jet, which makes it complementary to the information provided by the IPxD algorithms. The algorithm is trained on b-, c-, light-flavour and τ-jets and includes four respective output nodes, each corresponding to one of the jet flavours. These variables are propagated to a high-level tagging algorithm and are listed in Table 6.2.

Variables related to the presence of reconstructed muons, relying on muon tracks recorded either in the MS or ID, within the jet can be used in the jet flavour association. Muons are rarely found within typical light-flavour jets whereas b and c hadron decays may include them. Variables dedicated to the location of a muon within the jet or its relative contribution to it are constructed to allow the higher level flavour tagging algorithms to take the advantage of additional information of muons in jets. This can be performed by direct use of these constructed muon related variables, which are listed under the SMu category in Table 6.2, or by means of the output variables of a BDT trained using them as inputs, called SMT [67], which are listed under SMT in Table 6.2.

6.3.2 Vertex Based Algorithms

Another approach is to use information based on vertex reconstruction in the jet flavour identification. The vertex based low-level tagging algorithms are the Secondary Vertex 1 (SV1) and the JetFitter algorithms.

The SV1 algorithm [74] aims to reconstruct a single displaced vertex within the jet using a Jet Vertex Finding (JVF) algorithm. The JVF algorithm is based on a χ^2 fit method and reconstructs vertices within the jet from the set of all possible two-track vertices using the track constituents. The SV1 algorithm includes a likelihood test between b- and light-flavour jets using template histograms obtained from the JVF algorithm results from MC simulations. The algorithm provides kinematic fea-

Track based low-level algorithms		
Category	**Variable**	**Description**
IPxD (IP2D, IP3D)	ipX	Log-likelihood ratio (LLR) to separate b from light-flavour jets using the lifetime signed impact parameter significance. It is a sum of the per-track contributions. For N tracks in a jet, this is given by $\Sigma_{i=1}^{N} \log \frac{p_b}{p_{\text{light-flavour}}}$.
	ipX_c	LLR to separate b- from c-jets using the lifetime signed impact parameter significance
	ipX_{cu}	LLR to separate c- from light-flavour jets using the lifetime signed impact parameter significance
RNNIP	$\text{p}_{\text{b, c, light, tau}}$	RNNIP output
SMu	$\Delta\text{R}^{\text{SMu}}$	Distance between muon and closest jet
	d_0	Distance of closest approach to the primary vertex in the r-ϕ plane
	$\text{p}_{\text{T}}^{\text{rel}}$	Orthogonal projection of the momentum on the jet axis
	$\mathcal{S}$	Scattering neighbour significance
	$\mathcal{M}$	Momentum imbalance significance
	$\mathcal{R}$	Double ratio of Inner Detector and Muon Spectrometer q over p ratios
SMT	SMT_{BDT}	Output of a BDT [67], which is trained on SMu variables.

Table 6.2: Track based variables provided by the low-level algorithms.

tures associated to the secondary vertices which are reconstructed within the jets, which are listed in Table 6.3. An example of one of them, the reconstructed mass as calculated by SV1, is shown in Figure 6.4 and it can be seen that its population based on a $t\bar{t}$-dominated $e\mu$ sample can be approximated by a Poisson distribution with different maxima locations as well as tail shapes per jet flavour.

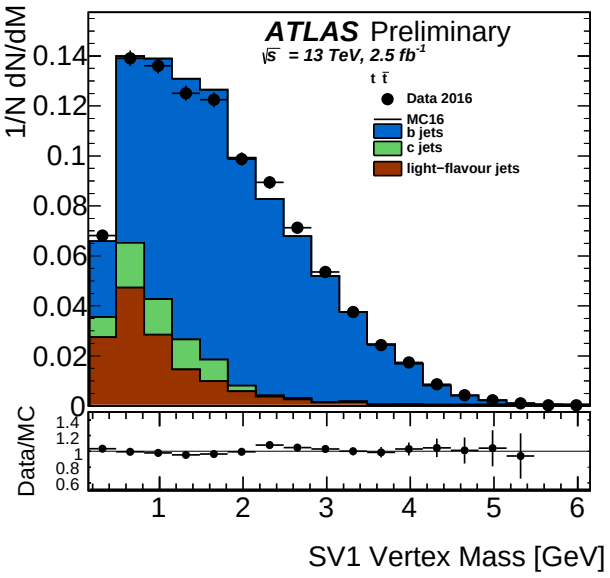


Figure 6.4: Distribution of the SV1 mass m_{inv}^{SV1} for the $t\bar{t}$-dominated $e\mu$ sample and its comparison to pp collision data at $\sqrt{s}$ = 13 TeV [67].

The JetFitter algorithm [75] performs a topological decay reconstruction of the jet, which focusses on the topological decay structure of weak interaction decays within a jet. The JetFitter algorithm includes a reconstruction of the vertices within the jet as well as track association to these vertices. This reconstruction is fundamentally constrained by the underlying assumption that the primary vertex, the b hadron and the c hadron flight paths are aligned on one line. This line is constrained to be able to be approximated by the direction of the jet axis within the uncertainties on the distances of the axis to the b hadron flight path. The uncertainties are derived from simulations and reconstruction or precision measurements. The algorithm considers the line of the reconstructed hadron flight path as the only potential origin of any subsequent vertex and tracks of the detectable decay particles. The vertices as well as the track association is sped up by these physics based assumptions and also further by the application of a Kalman Filter. The Kalman Filter method is applied for the consideration of the contribution of the addition of each track to the full topology reconstruction on a step by step basis as well as in the cross check of adding or reconstructing of a vertex. The reconstructed topology is considered

to simplify the problem as it reduces background tracks and vertices. The variables provided by this low-level algorithm are shown in Table 6.3 regarding those which were designed for b-jet tagging and in Table 6.4, which lists the variables designed for c-jet discrimination towards the other flavours and therefore are categorised as JetFitter$_c$.

Vertex based low-level algorithms (I / II)		
Category	**Variable**	**Description**
SV1	$N_{TrkAtVtx}$	Number of tracks associated with the secondary vertex
	m_{inv}^{SV1}	Invariant mass of the tracks associated with the secondary vertex assuming pion mass
	$N_{2TrkVtx}$	Number of two-track vertices candidates reconstructed within the jet
	f_E^{SV1}	Energy fraction $\Sigma_{track_i=1}^{N_{track,\ secondary\ vtx}} E_{track_i}$ / $\Sigma_{track_j=1}^{N_{track,\ jet}} E_{track_j}$ with the numerator being the sum over the energy of all tracks $N_{track,\ secondary\ vtx}$ associated to the secondary vertex and the denumerator being the sum over the energy of all tracks $N_{track,\ jet}$ of the reconstructed jet
	$\Delta R^{jet,\ SV1}$	ΔR between the reconstructed jet axis and the direction of the secondary vertex relative to the primary vertex
	L_{xy}	Transverse decay length of the reconstructed secondary vertex
	L_{xyz}	Decay length of the reconstructed secondary vertex
	S_{xyz}	3D decay length significance, i.e. the decay length of the secondary vertex divided by its uncertainty
JetFitter	$m_{inv}^{JetFitter}$	Invariant mass of tracks associated to one or more displaced vertices
	$f_E^{JetFitter}$	Charged jet energy fraction in the secondary vertices
	S_{xyz}	Decay length significance of the displaced vertex
	$N_{1\text{-trk vertices}}$	Number of 1-track displaced vertices
	$N_{\geq 2\text{-trk vertices}}$	Number of vertices with more than one track
	$\Delta R^{\vec{p_{jet}},\vec{p_{vtx}}}$	ΔR between the jet axis and the vectorial sum of all track momenta associated to displaced vertices

Table 6.3: Vertex based variables provided by the low-level algorithms.

Vertex based low-level algorithms (II / II)		
Category	**Variable**	**Description**
JetFitter$_c$	L_{xyz}	Distance of the secondary vertex from the primary vertex
	L_{xy}	Transverse displacement of the secondary vertex
	$Y_{trk}^{min,max,avg}$	Minimum, Maximum and Average track rapidity for all tracks in the jet
	$Y_{trk}^{min,max,avg}$ (2^{nd} vtx.)	Minimum, Maximum and Average track rapidity for tracks associated with the secondary vertex
	m	Invariant mass of tracks associated to secondary vertex
	E	Energy of charged tracks associated to secondary vertex
	f_E	Energy fraction of charged tracks (from all tracks in the jet) associated to secondary vertex
	N_{trk}	Number of tracks associated to the secondary vertex

Table 6.4: Additional vertex based variables which are specifically designed to improve c-jet tagging, as provided by the low-level algorithm JetFitter.

6.4 High-Level Algorithms

High-level tagging algorithms combine the discrimination power of the low-level algorithms into one single discriminant for the jet flavour tagging of interest being either b- or c-jet tagging. This reduces the required amount of calibrations and simplifies the inclusion of flavour tagging recommendations in analyses.

The baseline algorithms for b- and c-jet tagging are BDT based methods using the implementation of BDTs as given by the ROOT Toolkit for Multivariate Data Analysis (TMVA) software package [76]. This is the MV2 approach, where b- and c-jet tagging involves separate optimisations of multiple BDTs to contribute a high-level tagging algorithm of the MV2 family. The training is performed using the hybrid sample, whereas the performance is studied on the $t\bar{t}$ and Z' samples individually. The kinematic variables η and p_{T} of the jet are included in the training of a MV2 high-level tagging algorithm by default. Due to the kinematic dependence of the jet topology, the kinematics per flavour need to be represented equally per jet flavour as not to introduce a prior bias originating from the kinematics population of the training set. The b- and c-jets of the training training set are therefore modified in η and p_{T} to have their (η,p_{T}) distributions match the two dimensional (η,p_{T}) light-flavour jet distribution, which is referred to as reweighting.

The figures of merit are ROC curves, which show the respective background rejection power of the algorithm against the efficiency of the signal of interest. In addition, to provide a tagging algorithm which is performing well not just in a few small isolated p_{T} regions, the background rejection factors for a given signal efficiency as a function of the p_{T} of the jet is an additional figure of merit. Here, the rejection can either be calculated based on a single restriction for the entire p_{T} region of interest, which is referred to as a flat cut, or by defining cuts per p_{T} bin to keep the signal tagging efficiency constant within each p_{T} bin, which is known as a flat efficiency cut.

Overtraining of the MV2 algorithms is checked by calculating the predictions on the training set and comparing the resulting ROC curves to those calculated on the test set. The algorithm hyperparameter tuning was based on the performance on both of these sets.

A short overview of the MV2 b- and c-jet tagging algorithms is given below. Further information can be found in Ref. [67].

6.4.1 MV2 *b*-Jet Tagging Algorithm

The baseline high level algorithm for b-jet tagging is of the MV2 family and is known as *MV2*. The *MV2* tagging algorithm utilises a background versus signal labelling of the training jets where the background is constructed from a pre-defined mixture of light-flavour and c-jets. The exact mixture of these two components in the background composition is therefore considered a *MV2* hyperparameter which is tuned in dedicated studies. The procedure and choice is specific to the tagger optimisation and is not part of a general procedure specific to high level algorithms. The fraction of c-jets in the background composition of the *MV2* algorithm training set is set to 7%.

By defining a restriction value on the BDT output of *MV2*, a jet is defined to be tagged or not depending on whether the corresponding *MV2* value of the jet is above or below the restriction value respectively. Each restriction value is associated to a signal tagging efficiency, which is determined by the test set. The most common signal efficiency which is used is 70%, but 60%, 77% and 85% are also investigated as different analyses require different operating points.

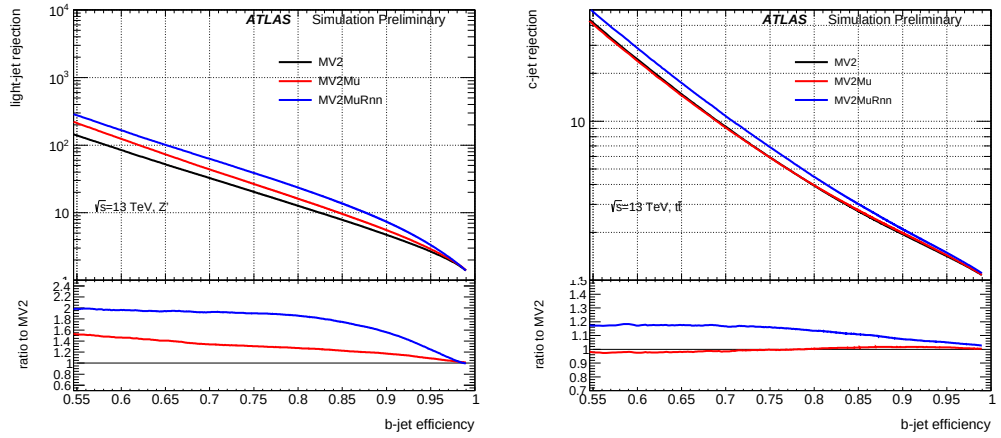


Figure 6.5: ROC curve showing the light-flavour jet (left) and c-jet rejection (right) performance of the MV2 b-jet tagging algorithms *MV2*, *MV2mu* and *MV2rnn* on the simulated $t\bar{t}$ sample [67].

Since additional low-level algorithms have been developed during Run 2, three different variations of MV2 are provided for b-jet tagging. These variations each use a different set of low-level algorithm variables as inputs to the higher level algorithm construction. The three MV2 variations are known as *MV2*, *MV2mu* and *MV2rnn*, where each subsequent algorithm includes more input variables compared to the previous variant. The basic set of input variables used for *MV2* includes the IPxD,

SV1 and JetFitter variables, which were designed for b-jet identification. *MV2mu* also includes the SMT variables as BDT inputs. In addition to these variables, the *MV2rnn* algorithm also includes the RNNIP variables and therefore includes the largest set of low-level algorithm output variables as inputs compared to the other instances of MV2 b-jet tagging algorithms. Their performance on $t\bar{t}$ events is shown in Figure 6.5 and shows the improvements in background rejections when including additional information from low-level tagging algorithms.

6.4.2 MV2 c-Jet Tagging Algorithm

Two BDTs are separately trained in the effort to provide a MV2-based c-jet tagging algorithm, referred to as MV2c(l)100. The inputs to these two BDTs are based on the input set used for *MV2rnn* but in addition also include additional JetFitter variables designed for c-jet identification, described previously in Table 6.4. One BDT, which is referred to as MV2c100, is trained to separate c- from b-jets and the other BDT, known as MV2cl100, is trained to separate c- from light-flavour jets. After training, these two BDTs are combined into one single c-jet tagging algorithm, referred to as MV2c(l)100, by combining their respective outputs in a two-dimensional discriminant and performing a rectangular cut on this plane.

MV2c(l)100 Operating Points

Operating Point Name	***c*-jet tagging efficiency**	**light-flavour jet rejection**	***b*-jet rejection**
Loose	41.5%	19.9	4.0
Tight	17.5%	190.9	18.3

Table 6.5: Chosen operating point of the MV2c(l)100 flavour tagging algorithm [77].

The figure of merit is an iso-efficiency curve, which visualises the background rejection performances by shifting the rectangular cuts while keeping a constant c-jet tagging efficiency. The performance is preferably investigated using the performance of the algorithm on the $t\bar{t}$ events, as shown by the iso-efficiency curve in Figure 6.6. Based on this figure of merit, two fixed performance points along given c-jet tagging efficiency lines, are chosen. They are referred to as the loose and tight operating points and refer to a given predicted light-flavour and b-jet rejection, which are provided in Table 6.5. The loose operating point corresponds to a c-jet tagging efficiency of 40%, and has a light-flavour jet rejection factor of 19.9 and a

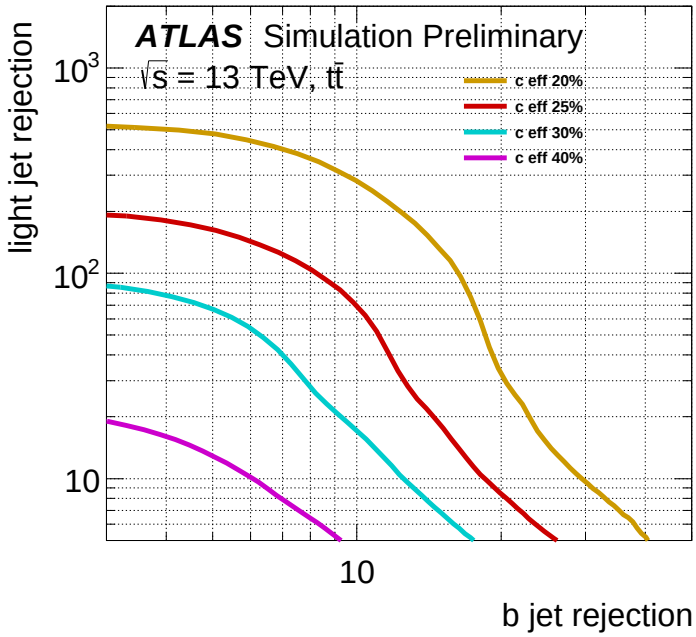


Figure 6.6: Iso-efficiency curve showing the performance for selected c-jet tagging efficiencies of the MV2c(l)100 c-jet tagging algorithm using the simulated $t\bar{t}$ sample [67].

b-jet rejection factor of 4.0. The tight operating point corresponds to a c-jet tagging efficiency of 17% and has a light-flavour jet rejection factor of 190.9 and a b-jet rejection factor of 18.3 .

6.5 Calibration

In order to use the outputs of the high-level tagging algorithms in physics analyses, a full calibration is required for their outputs. This is necessary in order to accommodate out differences between simulated events and reconstructed collision data using scale factors, which are defined as the ratio of the selection efficiency observed in data compared to the prediction from MC simulation. These scale factors are calculated as a function of jet kinematics. For the low-level algorithms a calibration is unnecessary as they are only used to provide intermediate variables within the generation of the high-level tagging algorithms.

The calibration procedure for the b-jet tagging efficiency is based on $t\bar{t}$ based calibration methods. A distinction is made between tag counting, kinematic selection, kinematic fit and combinatorial likelihood methods. Each method focusses on the fitting of either kinematic distributions or object, jet or event multiplicities of

	$t\bar{t}$ based b-jet tagging efficiency calibration methods			
Decay Channel	tag counting	kinematic selection	kinematic fit	combinatorial likelihood
single lepton	✓	✓	✓	—
dilepton	—	✓	—	✓

Table 6.6: Application of $t\bar{t}$ based b-jet tagging efficiency calibration methods depending on $t\bar{t}$ decay channels.

the collision data and the MC predictions. From these fits the predictions using the scale factors given by SF=$\epsilon_{data}/\epsilon_{MC}$ are extracted and applied to the output of the high-level tagger for use in data analysis. Different methods are applied depending on the $t\bar{t}$ decay channel due to their suitability. The decay channels include the single lepton decay or the dilepton decay channels, which are defined using the decays of the two $W^{\pm}$ bosons from the $t\bar{t}$ pair, and the methods that can be used for each respective decay channel are listed in Table 6.6. A detailed overview of the available procedures is presented in Ref. [78].

6.6 Monte Carlo Comparisons to Data

Before calibration, the uncalibrated high-level tagging algorithms are compared to their predictions on recorded pp collision data at $\sqrt{s}$ = 13 TeV. This comparison is meant to indicate the undesirable property whether the scale factors would end up making up for larger differences of how the high-level tagging algorithm interpreted the inputs and assigns class predictions. The recorded events were processed using the reconstruction settings from 2017 and therefore synchronise this aspect with the simulated data. This serves to review the agreement of the performance of the calibrated tagging algorithms compared to its performance on data. These studies use a $t\bar{t}$-dominated sample selecting opposite sign $e\mu$ events as well as a Z $\rightarrow \mu^{+}\mu^{-}$+jets-dominated sample. However, other samples are considered as well in order to study the universality of the high-level tagging algorithms.

The pre-calibration matching of the MV2 b-jet tagging algorithm to the predictions from simulation shows general good agreement between the predictions on MC simulation and the predictions on data as is shown in Figure 6.7.

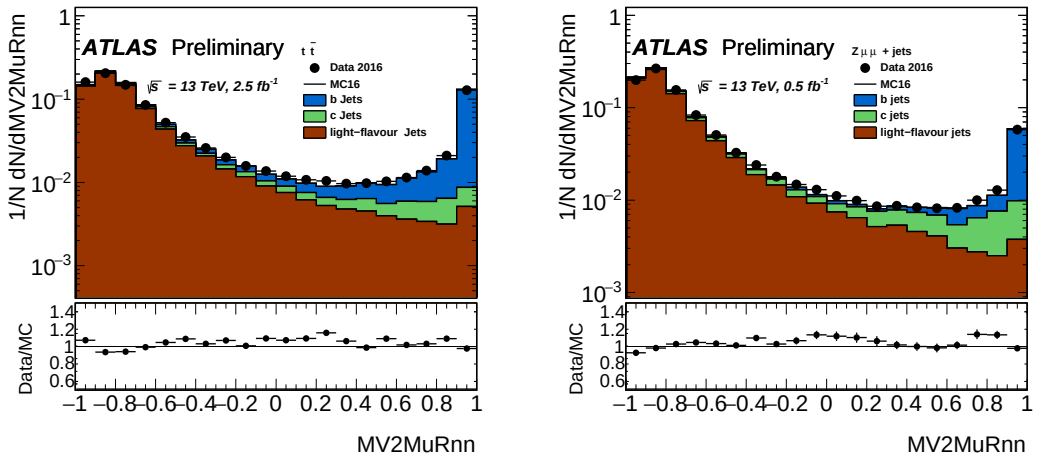


Figure 6.7: Data-MC comparison plots for *MV2rnn* on the predictions for b-, c- and light-flavour jets calculated for a $t\bar{t}$-dominated selection (left) and the Z $\rightarrow$ $\mu^+\mu^-$+jets-dominated selection (right) [67].

7. DL1 Design

Contents

DL1 is a high-level flavour tagging algorithm. The model of the algorithm is based on a deep NN with three output nodes to match the commonly well modelled and statistically well represented flavours in flavour tagging. Each output node corresponds to the predicted probability of the jet being a b-, c- or light-flavour jet. The output predictions of the NN simultaneously provide a b- as well as c-jet tagging algorithm, depending on the formulation of the final discriminant using the three output nodes. Following the Neyman-Pearson lemma [79], which states that the highest discrimination power is achieved by a log-likelihood combination, the final discriminant is defined as log-likelihood ratios constructed from the predicted probabilities to be a b-, c- or light-flavour jet.

This chapter describes the individual steps used to create instances of the DL1 tagging algorithm and making these available within the framework of the ATLAS collaboration. It covers the preprocessing required and performed before training, the tagging algorithm general overview of the overall software set up, how the discriminants to use the algorithms for both b- and c-jet tagging are constructed as well as the training approach. Different variants of the tagging algorithms are described and their set up is described at the end of this chapter and monitored quantities to decide on the final training hyper parameters.

7.1 Input Preprocessing

A few steps are required to prepare the input data for training in order to protect the NN from an a priori bias as well as destabilising effects, which disturb or alter the learning process. This is handled in three individual preprocessing steps before the training.

The first aspect of the preprocessing involves setting feasible default values for each attribute, which suit the usage within a NN model. Should the value of an attribute be undefined, default values are provided, which are usually far displaced from the physics values. However, using them as input variables to a NN, which works with the values of the attributes, is questionable as they would disturb the learning process as they largely extend the range of possible values even after applying an offset and scaling. Therefore, instead of using values far away from the valid physics distribution, the mean values of the distributions are used in combination with an additional binary check variable, which are shown in Table 7.2. These binary check variables indicate whether an input variable category contained a variable which includes a default value. This is to distinguish these jets from those which have the same value coming from underlying physics. As jets with a non-zero check variable have values for the given category, which are not determined by physics, they require special handling in the DNN. The binary check variable propagates this additional information. This procedure is followed except for few attributes where default values, which are motivated by the physics that results in them being undefined, are chosen in combination with the additional binary check variable. This is the case for the energy fractions associated to the reconstructed secondary vertex of a jet as well as the number of vertices with more than one track in a reconstructed jet. These variables are provided by vertex based low-level algorithms shown in Tables 6.3 and 6.4, which are set to a default value of zero as this value is undefined in the case no secondary vertex could be reconstructed in the event as it mostly is the case for light-flavour jets. The number of tracks associated with the secondary vertex of the SV1 vertex based low-level algorithm is set to two as this is the minimum track requirement for a vertex reconstruction within a jet.

In the second step, the values of the input variables are modified to work well in a NN. The individual input variables are scaled and shifted into distributions with a mean of zero and standard deviation equal to unity. This is achieved by applying an offset value to shift and a scale value to scale the distribution. The procedure is motivated by the desire to limit all the input ranges to have approximately the same ranges and to prevent individual variables from dominating the NN solely as

a result of their range of possible values when compared to others. Therefore, this step results in a more balanced set of input variables.

Finally, in order to provide a single training which is unbiased to favour the classification of any jet flavour against another, the values of the input variables are weighted to match the two-dimensional jet kinematics distribution of b-jets in $|\eta|$ and p_{T} as the distribution is smoother than the light-flavour jet distribution. This is in contrast to the MV2 procedure, which will adapt to also use the b-jet distribution. The absolute value of jet η is used as the particle population, and therefore the jet population, is symmetric in η around the IP for pp collisions.

The distributions of both attributes are shown per jet flavour in Figure 7.1, which highlights the jet kinematics dependence on the jet flavour. The weights used to modify the distributions are calculated from the training and validation set. They are used during training when updating the NN connection weights as well as when validating the performance of the NN based upon the objective function on the validation set after each training epoch.

The distribution of the weights applied to the samples are shown in Figure 7.2. The range of values does not exceed many orders of magnitude, which prevents updates using them to dominate against other contributions and disturb the learning process.

As is shown in Figure 7.3, after applying these weights, there are negligible differences in both kinematic variables between the three different jet flavours, as expected. These weights also affect the effective representation of the attributes. Therefore, a jet flavour class prior in the training data is prevented from entering and influencing the training.

It is shown in Figure 7.4 how the distributions of a variable change for c- and light-flavour jets change when applying the weights to match the c- and light-flavour jets to the two-dimensional (η, ϕ) b-jet distribution. It is also shown in the same Figure how the distributions of all flavours change when applying the scale and offset corresponding to the distribution of the attribute. However, it should be noted that the weights are only entering the network training in determining the loss contribution of the NN outputs. The NN is then optimised on the weighted sum of the of the losses for all individual three nodes. This way the network weights are optimised in the connection weight update step towards the weighted distributions and effectively see these weighted distribution as per attribute only one value can be propagated, not a weighted value.

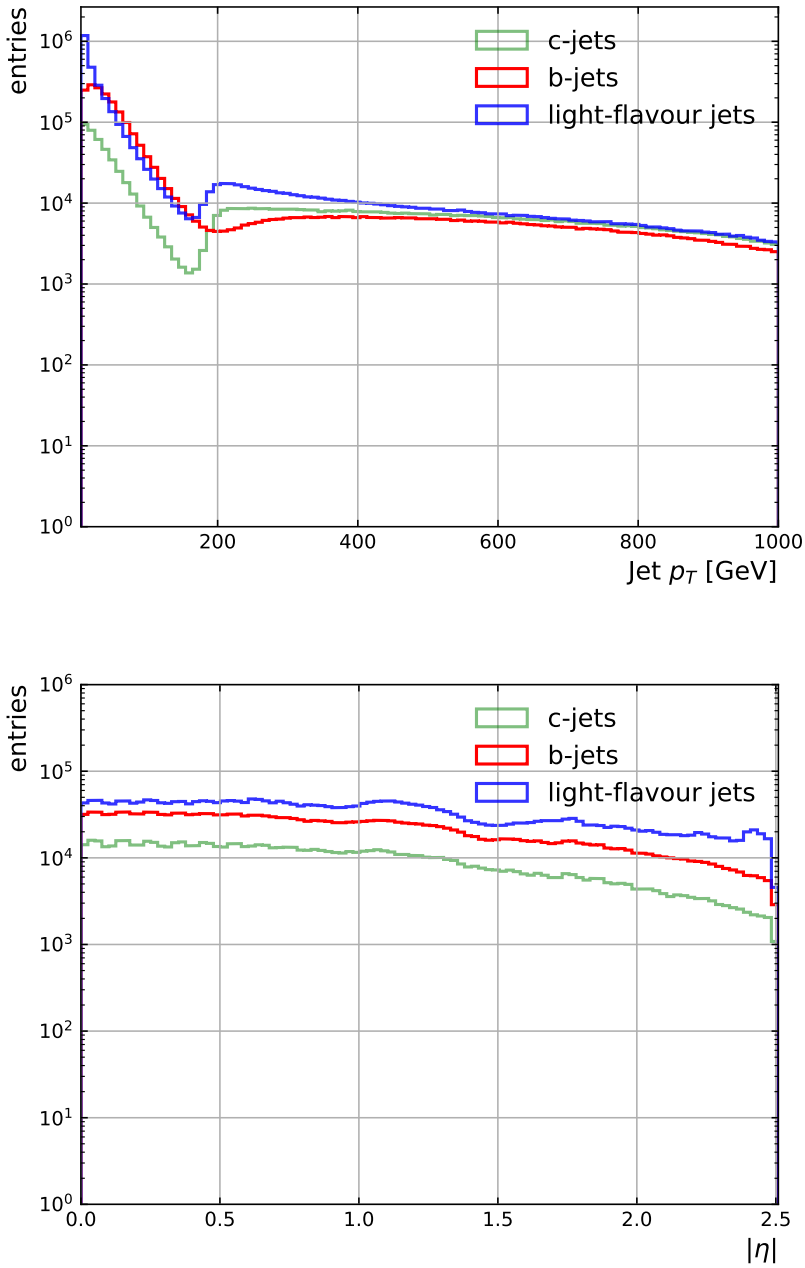


Figure 7.1: The jet p_{T} (top) and jet $|\eta|$ (bottom) distributions for b-, c- and light-flavour jets in the training and validation sets from the hybrid sample before reweighting.

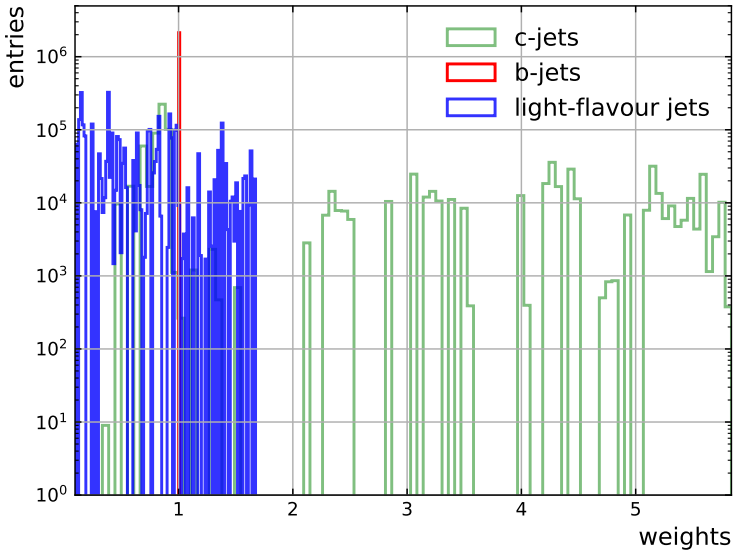


Figure 7.2: The weights used for b-, c- and light-flavour jets in the training and validation sets from the hybrid sample.

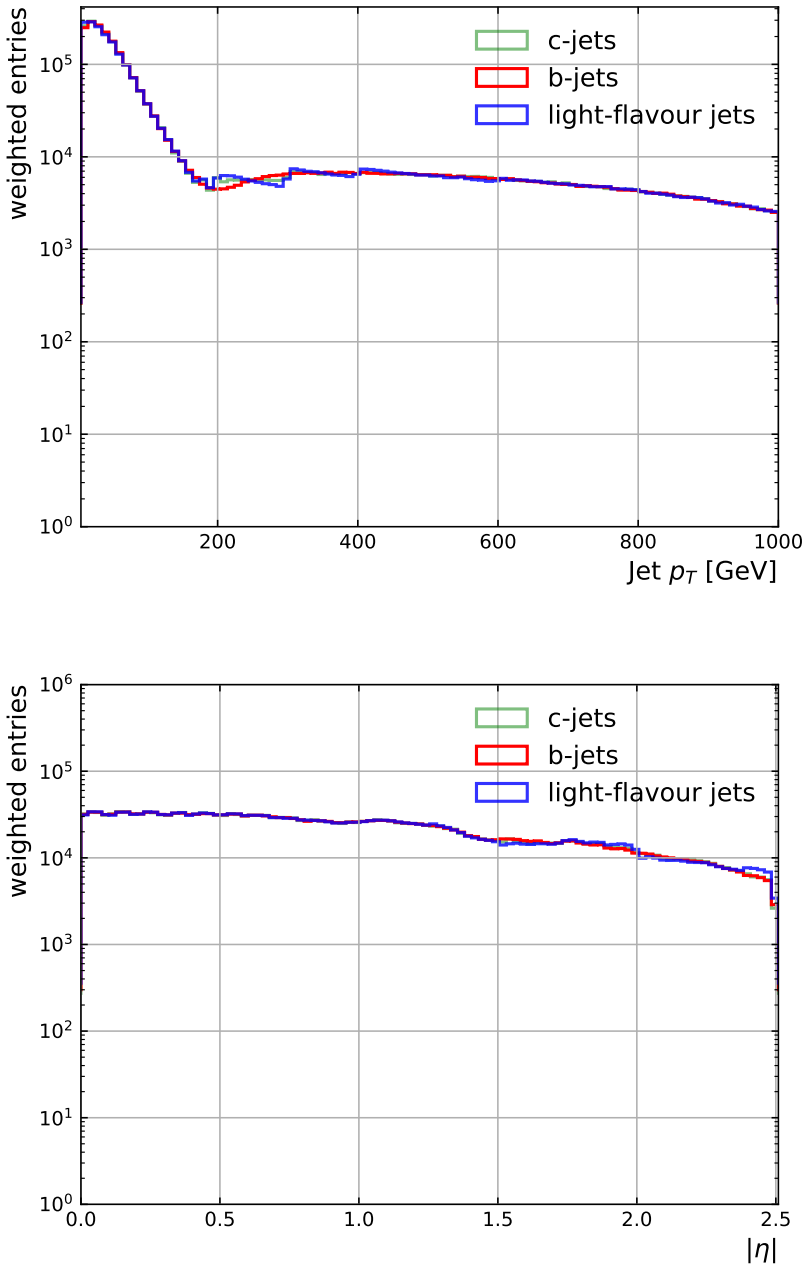


Figure 7.3: The jet p_{T} (top) and jet $|\eta|$ (bottom) distributions for b-, c- and light-flavour jets in the training and validation sets from the hybrid sample after applying the weights.

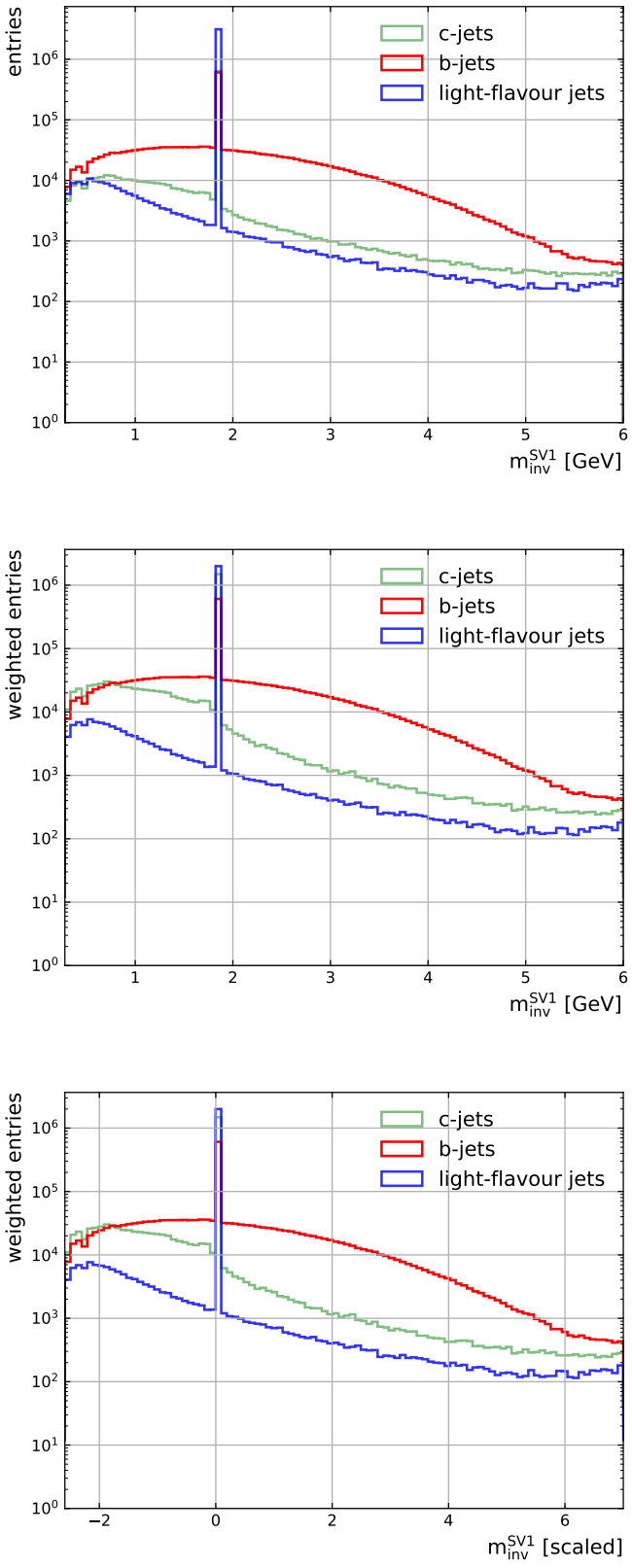


Figure 7.4: The SV1 mass distribution for b-, c- and light-flavour jets in the training and validation sets from the hybrid sample before (top) and after (middle) reweighting as well as with the corresponding scale and offset applied (bottom).

7.2 General Arrangement

Besides the preprocessing, architecture design and optimisation during the training it is a desirable feature of DL1 to provide a highly flexible high-level tagging algorithm. Since DL1 requires only one training to deliver multi-purpose tagging algorithms, it reduces the overall person power, computing power as well as required storage space in space-limited configuration databases and reduces the number of variables to be stored in simulated MC and collision data sets after including the object reconstruction recommendations. In addition to this, a portable implementation is required for the algorithm to be applied within the framework of a large collaboration with a bare minimum of library dependences and computational overhead.

7.2.1 Definition of the Final Discriminant

One of the biggest assets of DL1 is to be able to provide a general-purpose flavour tagging algorithm which can still be tuned after the training. This is provided by DL1 being a NN with multiple output nodes, which effectively is trained on an equal flavour representation. The outputs of the DL1 DNN can be combined to provide either a b- or c-jet tagging algorithm, where the extend of taking into account the different background output nodes can be tuned towards a desired performance within the discrimination capabilities of the DNN.

To provide a b-jet tagging algorithm, the outputs are combined into $\mathrm{DL1cf}_{c-\mathrm{jets}}$ given by

$$\mathrm{DL1cf}_{c-\mathrm{jets}} = \ln\left(\frac{\mathrm{p}_b}{\mathrm{f}_{c-\mathrm{jets}} \cdot \mathrm{p}_c + (1-\mathrm{f}_{c-\mathrm{jets}}) \cdot \mathrm{p}_{\mathrm{light-flavour}}}\right), \tag{7.1}$$

and the log-likelihood ratio $\mathrm{DL1bf}_{b-\mathrm{jets}}$ to provide a c-jet tagging discriminant given by

$$\mathrm{DL1bf}_{b-\mathrm{jets}} = \ln\left(\frac{\mathrm{p}_c}{\mathrm{f}_{b-\mathrm{jets}} \cdot \mathrm{p}_b + (1-\mathrm{f}_{b-\mathrm{jets}}) \cdot \mathrm{p}_{\mathrm{light-flavour}}}\right), \tag{7.2}$$

where p_b, p_c and $\mathrm{p}_{\mathrm{light-flavour}}$ are the outputs of the respective nodes from the same trained DL1 NN. This provides a high flexibility in tuning the fractions $\mathrm{f}_{b-\mathrm{jets}}$ and $\mathrm{f}_{c-\mathrm{jets}}$ after the training to tune the performance of the b- or c-jet tagging algorithms to match analysis preferences. In each case, the chosen value of the fraction is a hyperparameter in the optimisation of the final flavour tagging algorithm.

7.2.2 Software Implementation

The DL1 framework is an offline Python 2.7 based framework, which makes extensive use of open source libraries, especially NumPy [80]. The training of the DNN is performed on a graphical processing unit (GPU) cluster using Keras [81] with a Theano [82] back-end using 32-bit floating point precision. Next to NumPy, Pandas [83] is used for data handling and data is stored in the HDF5 [84] and JSON file formats. The hyperparameter space is probed using a grid search over feasible combinations of architecture constructions and training parameters. An optimally trained NN is chosen based on the performance of the NN. The performance is judged by the development of the objective function result as well as dedicated figures of merit calculated from the predictions on MC simulated events as well as comparisons using data recorded from pp collisions. Alongside the optimisation of the trained NN, the figures of merit are also analysed for different $\mathrm{f}_{c-\mathrm{jets}}$ as well as $\mathrm{f}_{b-\mathrm{jets}}$. These fractions are specifically optimised for each single training, however, they remain variable until set by calibration and can be optimised for individual use cases.

The final trainings are provided to the ATLAS software framework in ATHENA to apply the algorithm to jets of any sample of interest and include it in the general MC simulation productions as well as data processing. Within ATHENA, the NN model and connection weight information is combined within one configuration file to rebuild the DNN and the required calculations are performed using C++ libraries. For this, a C++ client called LightWeight Tagger Neural Network (LWTNN) [5] was developed to construct the NN within C++. The predictions p_b, p_c and $\mathrm{p}_{\mathrm{light-flavour}}$ of the NN are stored for each jet from which a final tagging discriminant can be calculated. This is done to preserve the flexibility of the algorithm. By changing the fractions $\mathrm{f}_{c-\mathrm{jets}}$ and $\mathrm{f}_{b-\mathrm{jets}}$, which let the background prediction values contribute respectively to the value of the final discriminant to a different extent, the freedom to vary the background rejections is kept. This enables people performing an ATLAS physics analysis the optional tuning of the flavour tagging algorithm to individual use cases. However, only a few selected fractions are chosen for calibration by the flavour tagging combined performance group.

7.3 Architecture and Training Considerations

The DL1 architecture is constructed from advanced layers featuring both dense and maxout layers described in Chapter 5 and a schematic overview of a typical DL1

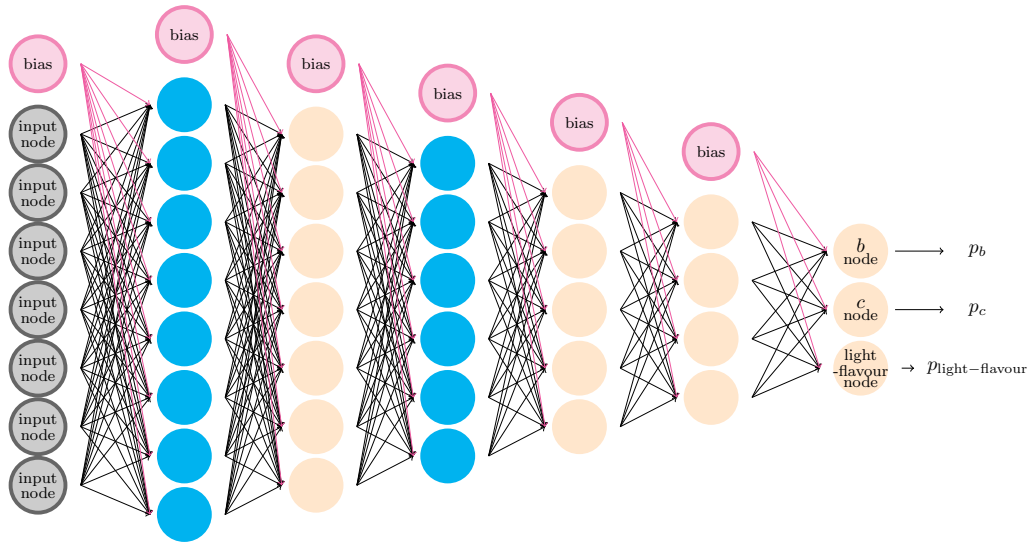


Figure 7.5: A schematic overview of a typical DNNarchitecture as used for DL1 [2].

architecture is shown in Figure 7.5. The ReLU activation functions applied to all layers except the output layer, where the softmax function is applied for the NN to provide classification probabilities per jet to each of the considered classes. Batch normalisation is included for theoretical reasons as well as empirical results as it leads to a faster training and improved performance. During training dropout is employed to provide a robust DNN.

Hyperparameter	**Range**
$N_{\text{hidden layers}}$	5 to 14
$N_{\text{maxout layers}}$	1 to 3 at different positions
$N_{\text{parallel layers}}$ per maxout layer	5 to 30
$N_{\text{nodes/layer}}$	Up to 78
$N_{\text{training epochs}}$	100 [50 (10)] [1]
Learning rate	0.0001, 0.0005, 0.001
Training minibatch batch size	50 to 500

Table 7.1: Table of hyperparameters used in the DL1 grid searches.

The optimisation process relies on a grid search over multiple hyperparameter, as seen in Table 7.1, which is combined with manual quality checks. The construction of the grid search is guided by the principle of keeping the number of learnable

[1] The best performing configurations are trained further and considered from 50 to 500 in steps of 50, then those which perform best are considered in steps of ten.

parameters about an order of magnitude lower than the available number of jets in the training sample to prevent larger risks of immediate overtraining. The training set consists of 5.1 million jets, the validation set of 1.3 million jets and the test set of 6.6 million jets. The quality checks include the monitoring of the development of the objective function as well as performance measures based on multiple figures of merit. Keras is performing the training as well as calculating the predictions on the test set. During the training of each NN the NN connection weights and the loss calculated on the training and validation sets are saved after each single training epoch. This allows monitoring checks and a quick post-training optimisation of the number of training epochs, which is an important hyperparameter of the training. For the DL1 trainings with the best overall performance the training is extended to 500 epochs and the performance is further monitored in training epoch steps of 50. The final number of epochs is determined by investigating the loss develop monitoring and performance figures of merit in steps of 10 for the previously found best performance.

7.4 DL1 Variants

Within a flavour tagging configuration database is the infrastructure for three variants of DL1 NN configurations to make the outputs available for analyses. One flavour tagging configurations database is created for internal performance comparisons to reflect the possibilities of the methods using the same information content as used by the MV2 variants with the limitations to three variants within the ATHENA framework.

The versions of DL1 in this configuration are referred to as *DL1baseline*, *DL1mu* and *DL1rnn* and described in detail in Ref. [2]. A ROC curve performance comparison is comparing the rejection factors of the b-jet tagging algorithm *DL1rnn* to the flavour tagging baseline is shown in Figure 7.6.

Another flavour tagging configurations database, which is used for physics analysis and is the main focus of this thesis as it represents the full spectrum of tagging algorithms which end up available to analyses. The DL1 variants which are available to analyses always include the additional JetFitter variables designed for c-jet tagging as DL1 is designed to be both, a b- and c-jet tagging algorithm simultaneously. These variants are referred to as *DL1baseline*, *DL1mu* and *DL1* and written in italic when referred to throughout this thesis. The number of learnable parameters of the DNN models of these DL1 variants range from approximately 110,000 to 135,000.

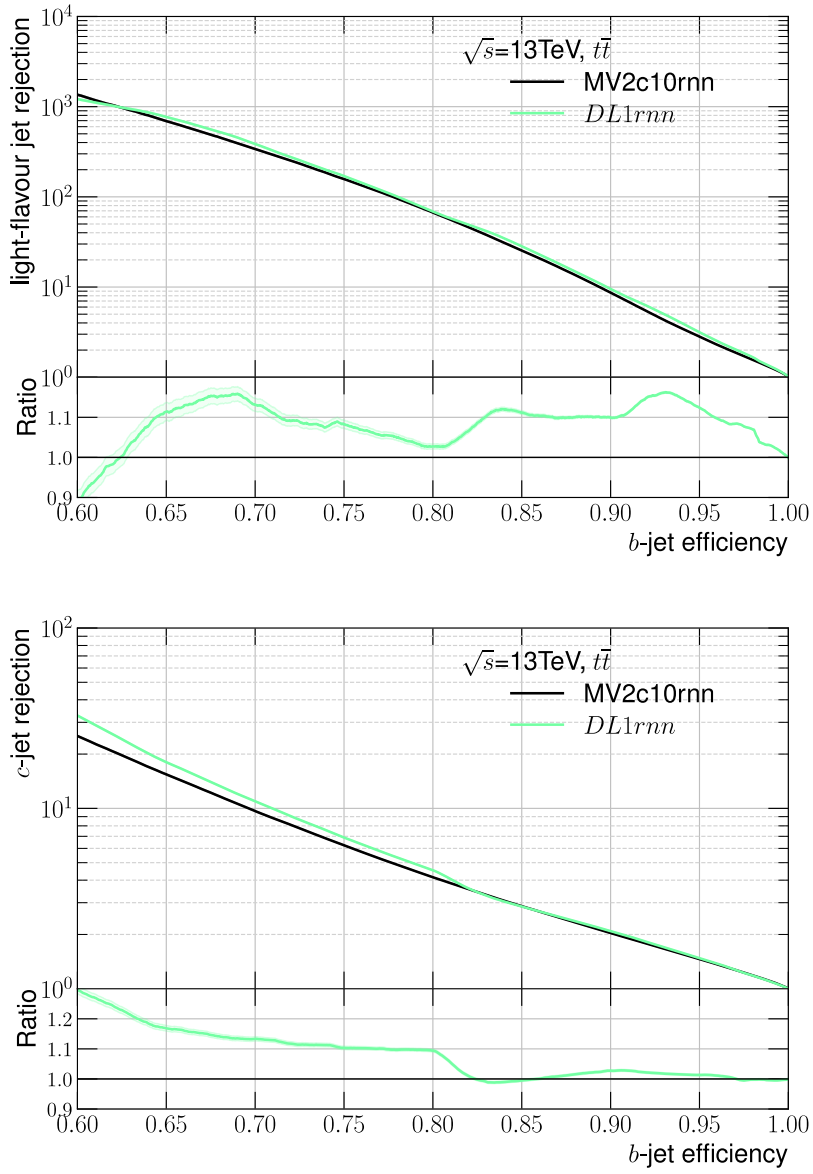


Figure 7.6: The ROC curves showing the light-flavour (top) and c-jet (bottom) rejection factors of the b-jet tagging algorithm *DL1rnn* compared to the performance of the MV2 baseline high-level flavour tagging algorithm *MV2rnn* as shown in Ref. [2].

Binary default check variables		
Category	**Variable**	**Description**
Binary	(IP2, IP3, SV1, JetFitter, JetFitter$_c$, SMu, RNNIP)$_{\text{check}}$	To keep track of jets with default values in each category. Variables equal one if category contains a default value, else zero.

Table 7.2: Binary input variables accompanying the different sets of input variables in the different DL1 variants.

	DL1 Variants		
Low-level algorithm	*DL1baseline*	*DL1mu*	*DL1*
IPxD	✓	✓	✓
SV1	✓	✓	✓
JetFitter	✓	✓	✓
JetFitter$_c$	✓$^{(*)}$	✓$^{(*)}$	✓$^{(*)}$
SMu	–	✓	✓
SMT	–	–	–
RNNIP	–	–	✓

Table 7.3: Overview of the low-level algorithm categories of input variables for the instances of DL1 high-level tagging algorithms. The kinematic variables $|\eta|$ and ϕ are included by default, which results in a total number of 35, 42 and 46 attributes for *DL1baseline*, *DL1mu* and *DL1* respectively. Ref. [2] compares independently optimised variants of DL1 tagging algorithms where categories denoted with * are excluded. The reference also includes *DL1* in order to have a presentation of its c-jet tagging performance, which is a fair comparison to MV2c(l)100 as both algorithms use the JetFitter$_c$ variables shown in Table 6.4.

The different variables used for each DL1 variant are given in an overview per category in Table 7.3. The binary check variables mentioned in Section 7.1 are added per input category shown in Table 7.3 as listed in Table 7.2. It is chosen to add them per category in order to prevent copies of the same information content and keep the number of input variables at a minimum.

7.5 Monitoring and Quality Checks

While the Grid Search described in Section 7.3 was performed for each of the three DL1 variants described in Section 7.4, this section focusses on determining the best suited NN hyperparameters to provide high-level flavour tagging algorithms for the different sets of attributes presented in Table 7.3. The decision on the final hyperparameter values is based on a series of cross checks involving monitoring of the training on the hybrid sample, containing $t\bar{t}$ and Z' events, as well as a systematic sequence of multiple figures of merit, each reflecting the performance of the DNN at a given stage in the training using $t\bar{t}$ events.

Due to the large number of trainings, the first sets of hyperparameters are filtered out by monitoring the loss development on the training and validation dataset. The loss on each dataset should not be separated by a large amount and the number of training epochs should preferably be chosen so as not to coincide with a spike in the loss calculation on the validation set.

Alongside the relative difference between the loss on the training and validation set, the overall decrease of the values is to be minimised. The loss development on the training and validation sets for the *DL1* training is shown in Figure 7.7. The calculated losses on both sets are decreasing up to about ten epochs from where the validation loss starts to slowly increase whereas the loss on the training set continues to decrease from values which are larger than the loss calculated on the validation set. Both loss developments have overlapping values from 50 to 80 training epochs above which the validation set starts to continuously show higher values. While the loss calculate on the training set continues to drop slightly above 100 training epochs, the validation set loss begins to rise to higher loss values which clearly separate these two developments and indicate overtraining, which also reflects itself in the decreasing performance in the figures of merit on the test set. During the entire training large spikes in the validation set for individual training epochs can be observed. These spikes are also an indication for overtraining on the training dataset and the corresponding epochs are best to be avoided as the final choice but the NN might be able to recover from it in the next training epoch.

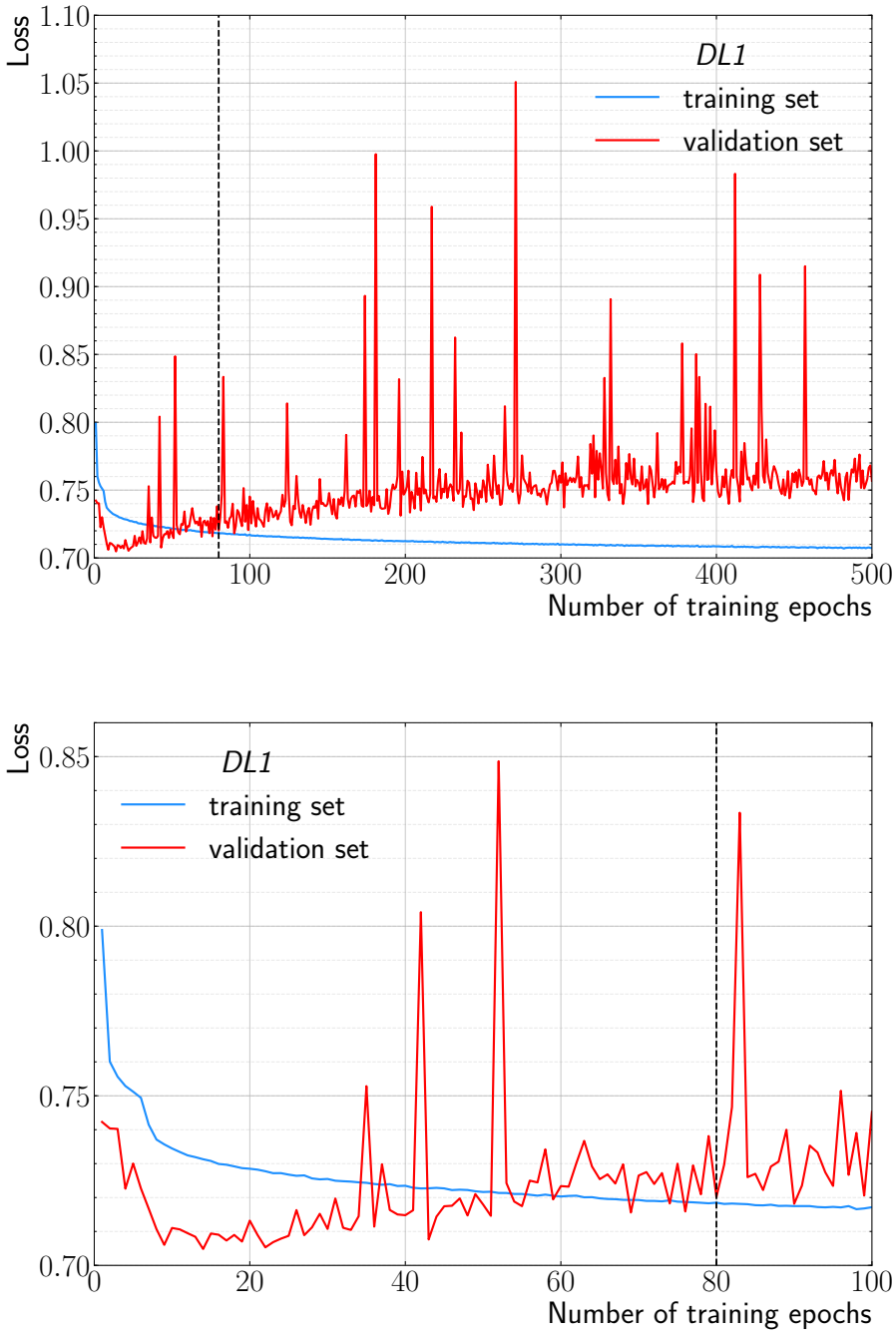


Figure 7.7: The loss development on the training and validation set for *DL1* for the extended number of training epochs (top) and a cut-away view over the first 100 training epochs. The dashed line represents the number of training epochs chosen as the final hyperparameter value of *DL1*.

Trainings with the lowest overall loss values are determined from the grid search runs. In a next step these trainings are evaluated on the test set consisting of $t\bar{t}$ events, which are a disjoint selection from the training and validation set. The figures of merit are compared to a baseline. At first the background rejection factors provided by iso-efficiency curves are analysed. From this figure of merit general performance trends are deduced and individual fractions $\mathrm{f}_{c-\mathrm{jets}}$ are selected to be used for further performance investigation. This further investigation of the performance includes ROC curves for selected signal tagging efficiencies. The main focus of the optimisation was on the performance of b-jet tagging at 77% b-jet tagging efficiency. Promising NN trainings are further analysed using the rejection factors as a function of jet p_{T}. Both the ROC curve and the jet p_{T} dependence uses a flat cut calculation for the performance as described in Section 6.4 on the $t\bar{t}$ jet p_{T} spectrum up to 300 GeV, selecting only jets below this threshold. At this stage, the most promising DNN trainings are continued to 500 training epochs and their performance is re-evaluated in steps of 50 and subsequently 10 for regions of interest to optimise the final total number of training epochs after which the network training is determined. Only DL1 trainings with a consistent improvement are considered as DL1 high-level tagging algorithm candidates and have their c-jet tagging performance evaluated using iso-efficiency curves. After all these stages for *DL1baseline*, *DL1mu* and *DL1baseline*, a final candidate is chosen for each of them from their individual grid searches. The *DL1* variant is again chosen in the remaining part of this chapter to illustrate further quality checks which are performed on all final candidates for DL1.

The classification predictions are shown per *DL1* output node in Figure 7.8, where the distributions are shown for each jet flavour label. As can be seen, the DNN predicts the labels, which it was trained with, with high probability as the distributions of the matching labels are peaking at higher values for these nodes. Nonetheless, jet overlap of different labels remains, though for high values of probability in each node, the other jet flavour labels, which a node was not trained on, are around an order of magnitude smaller. Additionally, these distributions reflect the expected trend that b- and light-flavour jets are more different from each other and easier to separate than c- and light-flavour jets.

The tunings of the final discriminants when using the same p_b, p_c and $\mathrm{p}_{\mathrm{light-flavour}}$ shown in Figure 7.8 using Equations 7.1 and 7.2 lead to different distributions of the final discriminants. The final discriminants for the b- and c-jet tagging algorithms of *DL1* with the tuned $\mathrm{f}_{c-\mathrm{jets}}$ or $\mathrm{f}_{b-\mathrm{jets}}$ are shown in Figure 7.9. It is desirable to avoid extreme peaks of multiple orders of magnitude difference for a small range

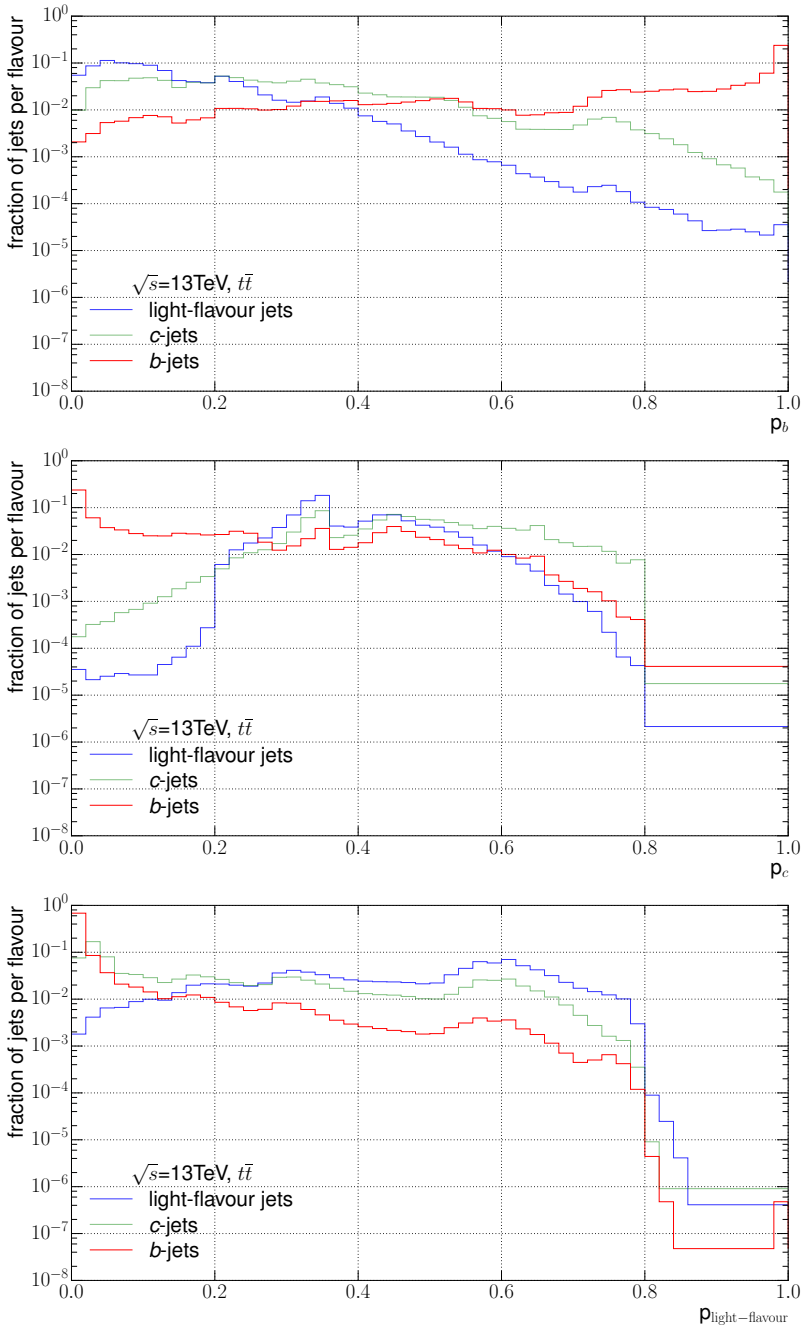


Figure 7.8: The per flavour normalised distributions of the predictions provided by the outputs of the output layer of the *DL1* DNN [2].

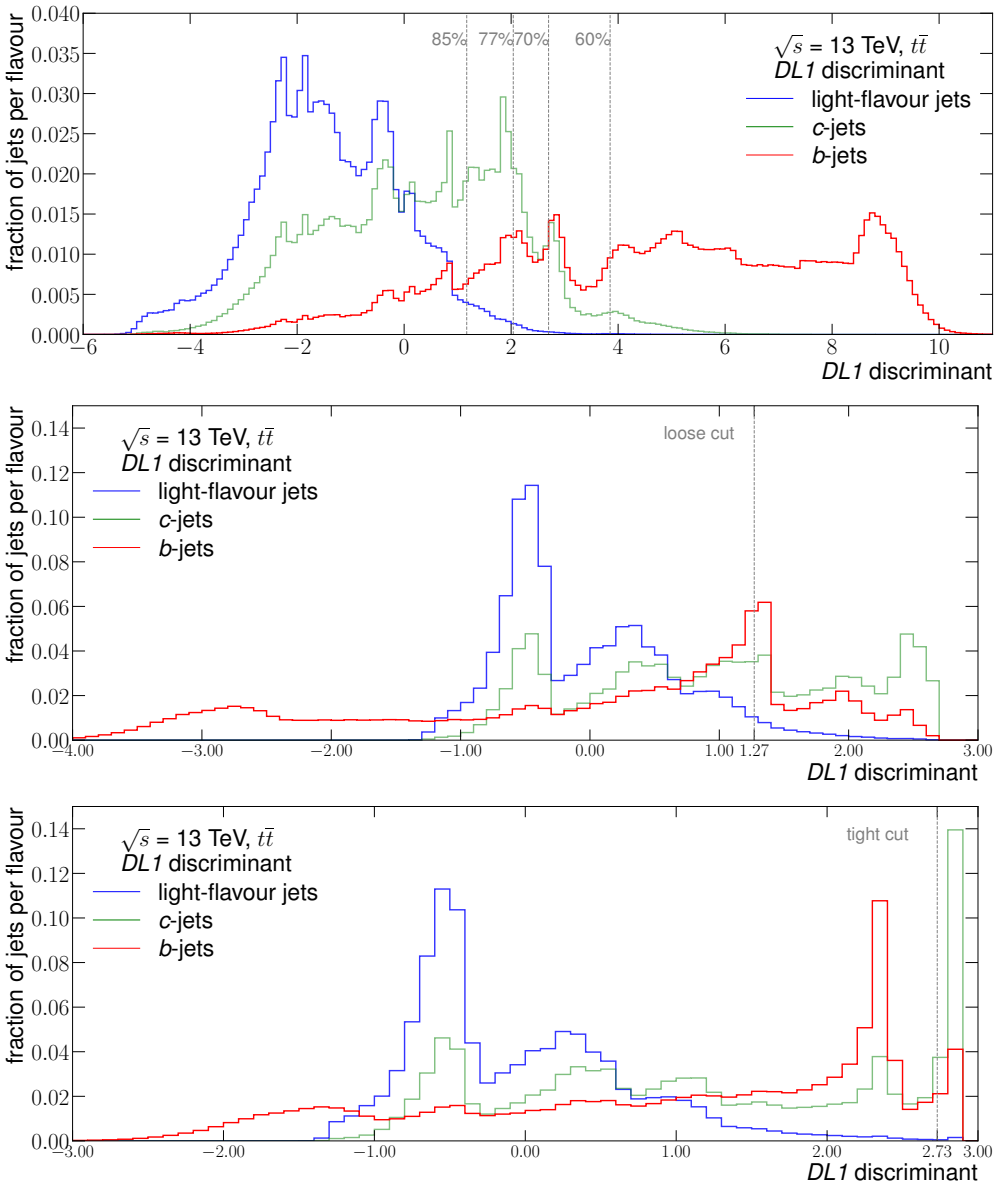


Figure 7.9: The final discriminant distributions of *DL1* for the different background node tunings for b- (top) and c-jet (middle and bottom) tagging normalised for all individual jet flavours. The middle plot shows the final discriminant distribution for the loose c-jet tagging operating point and the bottom plot for the tight c-jet tagging operating point, where both have individually tuned background node fractions.

of final discriminant values in regions of signal efficiencies of interest. If present in the distribution of the final discriminant, they can have a large impact on the performance when considering small shifts of the cut value if the cut values are located close to the spike. All three presented variants of DL1 have been checked and no such spikes are present. It is found that the addition of the binary check variables as input variables help broadening peaks originating from default variable spikes.

The *DL1baseline* b-jet tagging algorithm uses a $\mathrm{f}_{b-\mathrm{jets}}$ of 3%, which is chosen based on an optimisation comparing the light-flavour and c-jet rejection factors at 77% b-jet efficiency to the same figures of merit of the MV2 high-level b-jet tagging algorithm *MV2rnn*. The chosen operating point of 77% b-jet tagging efficiency is used for the optimisation of high-level flavour tagging algorithms as it is considered the most important one of the four operating points used in flavour tagging, which correspond each to a b-jet tagging efficiency of 60%, 70%, 77% or 85%.

The c-jet tagging algorithm performance is separated into a loose and tight operating point and uses a c-jet tagging efficiency of 40% at a $\mathrm{f}_{b-\mathrm{jets}}$ of 8% for the loose operating point and 17% at a $\mathrm{f}_{b-\mathrm{jets}}$ of 2% for the tight operating point. These operating points were primarily tuned to match the b-jet rejection factors of the operating points chosen for MV2c(l)100. Both are calculated using the $t\bar{t}$ sample while keeping the c-jet tagging efficiency approximately fixed as it was chosen to reduce the tagging algorithm complexity for DL1 algorithms to only allow for integer percentages of c-jet tagging efficiencies. A variation of single percentages is only allowed if the light-flavour jet rejection factors are largely improved by doing so.

8. DL1 Performance

Contents

Following on from the setup and optimisation strategy discussed in Chapter 7, this chapter presents the b- and c-jet tagging performance of the three DL1 variants *DL1baseline*, *DL1mu* and *DL1* evaluated on simulated $t\bar{t}$ events with a jet p_{T} of up to 300 GeV. The performance is measured using different figures of merit for the b- and c-jet tagging algorithms. Additionally, the calibration is presented to show that the MC simulation based performance of DL1 is in very good agreement with collision data and therefore the shown algorithm performance can be assumed to hold true on collision data as well as simulation.

8.1 *b*-Jet Tagging Performance

For b-jet tagging, figures of merit used to evaluate the performance are ROC curves, calculated using a flat cut efficiency and investigating the background rejection factors as a function of jet p_{T} using a flat signal efficiency efficiency calculation across individual jet p_{T} bins. Both flat efficiencies and flat cuts are calibrated and used for b-jet tagging in physics analyses.

The b-jet tagging performance of all three DL1 variants for a b-jet tagging efficiency calculated using all jets with $p_{\mathrm{T}} > 20\ GeV$ ranging from 60% to 100% is shown in Figure 8.1. Since the tagging algorithm were optimised especially for a b-jet tagging efficiency of 77%, with the intention to prioritise on increase in the light-flavour jet rejection factors rather than the c-jet rejection factors, it can be seen

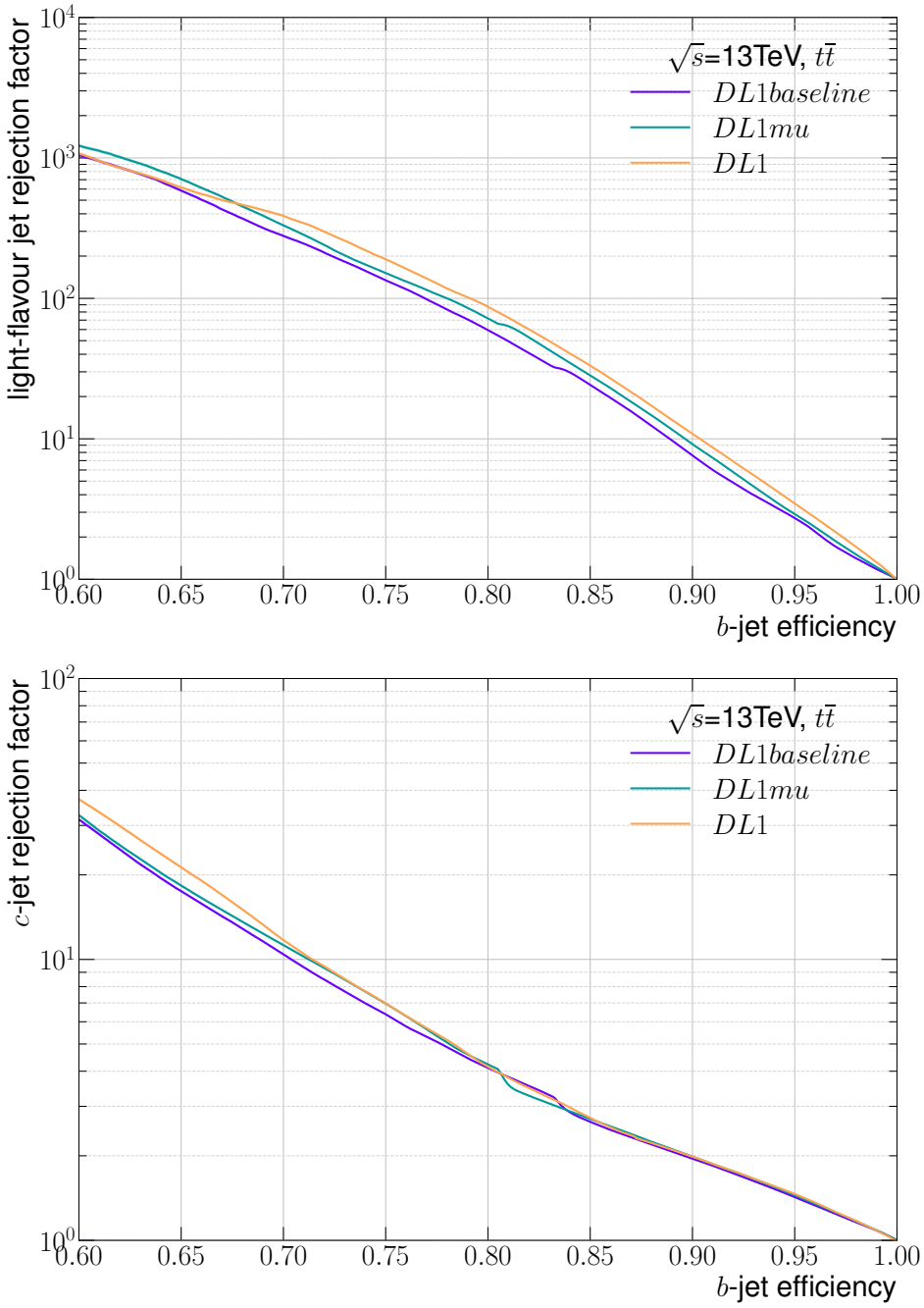


Figure 8.1: The ROC curve b-jet tagging performance plots showing the light-flavour jet (top) and c-jet (bottom) rejection factors on jets reconstructed in simulated $t\bar{t}$ events from pp collisions at $\sqrt{s}$ = 13 TeV as a function of the b-jet tagging efficiency of the DL1 variants *DL1baseline*, *DL1mu* and *DL1*.

that the c-jet rejection factors at 77% are quite close across the DL1 variants, with *DL1mu* and *DL1* performing slightly better. However, the light-flavour jet rejection factors increase when increasing the information content used as an input to the individually optimised DNNs. This behaviour for light-flavour jet rejection factors

is quite constant for b-jet tagging efficiencies down to 68%, where the light-flavour jet rejection factors of *DL1mu* outperform those of *DL1* and below which exhibits a performance closer to *DL1baseline*. For c-jet rejection however, *DL1* outperforms the other two DL1 variants for b-jet tagging efficiencies below 72%, where *DL1mu* and *DL1baseline* indicate similar rejection power. The overall performance is in agreement with the intended optimisation, favouring the light-flavour jet rejection power over the c-jet rejection power and focussing especially on a b-jet tagging efficiency of 77%.

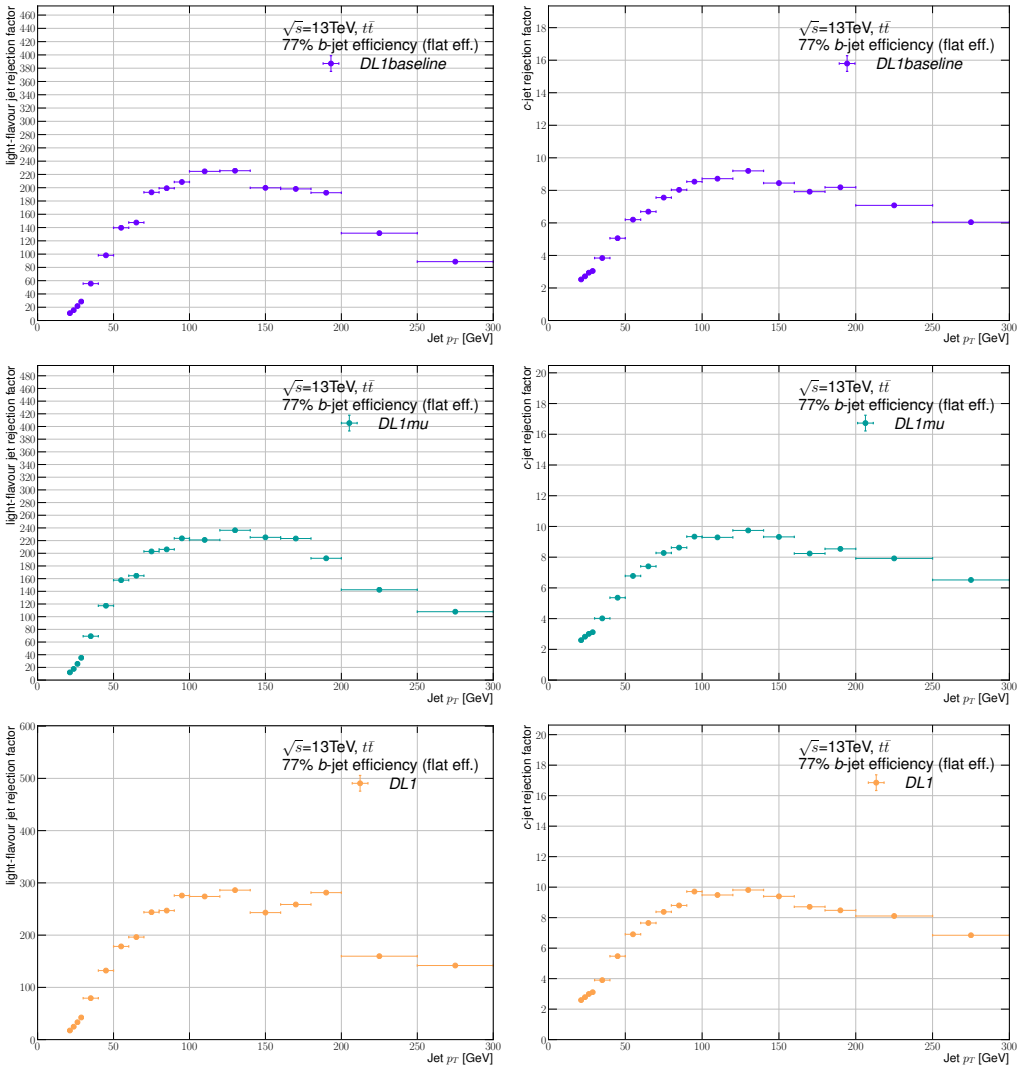


Figure 8.2: The b-jet tagging performance plots showing the light-flavour (left) and c-jet rejection factors (right) on jets reconstructed in simulated $t\bar{t}$ events from pp collisions at $\sqrt{s}$ = 13 TeV as a function of jet p_{T} for a b-jet tagging efficiency of 77% for the DL1 variants *DL1baseline* (top), *DL1mu* (middle) and *DL1* (bottom).

However, ROC curves on their own do not provide sufficient enough informa-

tion concerning the quality of the performance. The overall performance calculated using a flat cut efficiency masks occurrences where the DNN might be very well optimised for classifying jets within a small region of high population in the $t\bar{t}$ sample, which then counteracts less well performing classification in the remaining jet p_{T} spectrum.

To study the consistent performance of a b-jet tagging algorithm, flat efficiency cuts are applied and the performance in terms of background rejection factors is analysed per bin as a function of jet p_{T}. The light-flavour and c-jet rejection factors for a b-jet tagging efficiency of 77% are shown for all three DL1 variants as a function of jet p_{T} in Figure 8.2. The rejection factors for both light-flavour jet and c-jet rejection follow the generally expected shape which is also observed for the MV2 b-jet tagging algorithms. This expected shape consists of high rejection factors for jets in the mid p_{T} range from 60 GeV to 200 GeV, where enough statistics is present and the b hadron p_{T} is high enough to result in a distinguishable travel path, whereas in the lower jet p_{T} region the travel path is shorter and therefore leads to a jet topology, which is more difficult to distinguish from c- and light-flavour jets. For more boosted jets in the higher jet p_{T} region the rejection factors are expected to decrease a bit due to the modelling of the boosting of the jet and lower statistics.

From these plots, the overall good performance improvement is seen throughout the full jet p_{T} spectrum and not due to a localised performance improvement for all DL1 b-jet tagging algorithms. While there are only slight improvements observed when moving from *DL1baseline* to *DL1mu*, they become more visible when comparing to the b-jet tagging performance of *DL1*. This is especially the case in the light-flavour jet rejection factors of *DL1*, where the improvements with respect to the other DL1 b-jet tagging algorithms are global improvements. In the mid p_{T} range from 60 GeV to 200 GeV, which covers the majority of the region of interest in jet p_{T} for b-jet tagging performance on $t\bar{t}$ events, the improvements are most visible to the extent of a less curved but more plateau-like behaviour in the rejection factors. However, improvements can also be seen in the rejection factors in the lower and higher jet p_{T} regions. As expected from the rejection factor values observed in the ROC curves at 77% b-jet tagging efficiency, the performance increase, though present, is less obvious in the c-jet rejection factors.

8.2 *c*-Jet Tagging Performance

In contrast to b-jet tagging, the main figures of merit for the overall tagging performance in c-jet tagging are iso-efficiency curves, where the rejection factors for

both b- and light-flavour jets are calculated at a given c-jet tagging efficiency. In this method a single $\mathrm{f}_{b-\mathrm{jets}}$ value can be represented as a dot and a fine array of them as an iso-efficiency line, whose values correspond to the same c-jet tagging efficiency. The overall rejection power is shown across the jet p_{T} spectrum up to 300 GeV. The *DL1* c-jet tagging algorithm allows for a direct comparison to the MV2 c-jet tagging variant, which is used to define the loose and tight c-jet tagging operating points.

For *DL1baseline* and *DL1mu* similar tunings of the final discriminants are shown in Figure 8.3 without showing their individual tunings, which vary due to optimisation in the choice of $\mathrm{f}_{b-\mathrm{jets}}$ in each of their final discriminants and slightly in the choice of c-jet tagging efficiency. For all three DL1 variants the same c-jet tagging efficiencies as well as position of the performance given the same fraction $\mathrm{f}_{b-\mathrm{jets}}$ in the final discriminants are shown, as long as they represent a valid choice for the tagging algorithm. A valid choice is defined as being in a region of performance, which would be considered for use in an analysis. As already observed in the performance of the b-jet tagging algorithm, additional variables improve the rejection factors for both b- and light-flavour jets. This can be seen in the iso-efficiency lines, which given the same b-jet rejection factor only showing higher light-flavour jet rejection factors as the number of input variables increases. It should be noted that the position of the performance corresponding to the same $\mathrm{f}_{b-\mathrm{jets}}$ in the definition of the final discriminant for every iso-efficiency line does not lie at the same distance along the line with respect to the starting point of the iso-efficiency line, and that respective distances can vary individually per line and per line segment. This reflects the fact that the fractions $\mathrm{f}_{b-\mathrm{jets}}$, as well as $\mathrm{f}_{c-\mathrm{jets}}$, are performance parameters which need to be optimised for each individual flavour tagging algorithm.

The rejection factors for the chosen operating points for *DL1* are listed in Table 8.1, where the values are compared to the operating points defined for MV2c(l)100. While both c-jet tagging algorithms perform quite similarly for the loose operating points, the good flavour separation power of DL1 tagging algorithms becomes more visible when moving to a tighter operating point, which involves a lower c-jet tagging efficiency. There the smaller overlap in the final discriminant between c- and light-flavour jets becomes apparent and is reflected by large light-flavour jet rejection factors. It is possible for analyses, which use c-jet tagging, that the performance provided by these MV2c(l)100 information based definitions of *DL1baseline* operating points can be improved further by tuning the definitions of the operating points purely for *DL1*. This can also include a new tuning of $\mathrm{f}_{b-\mathrm{jets}}$ for their final discriminants.

An additional figure of merit is provided by the rejection factors as a function

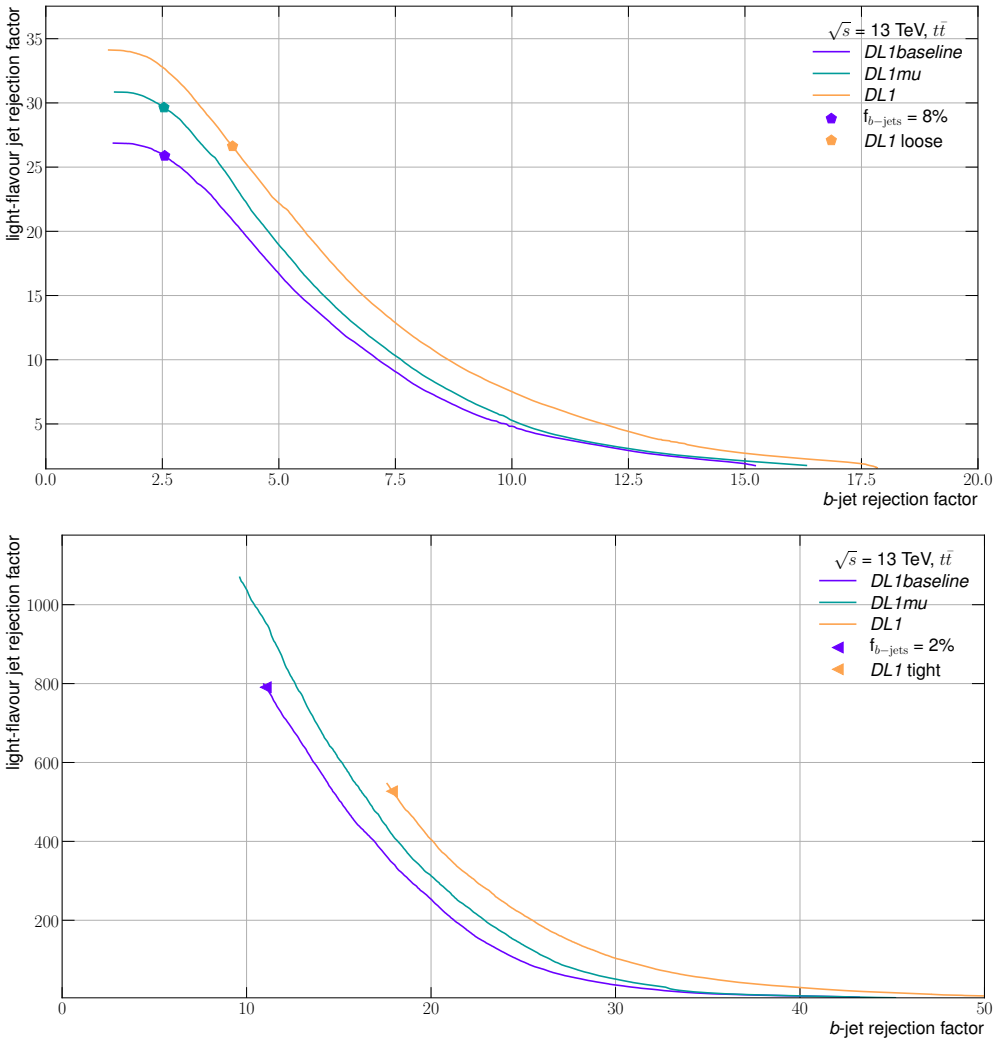


Figure 8.3: The iso-efficiency curves showing the performance of each chosen final discriminant tunes of *DL1* for c-jet tagging efficiencies matching the loose (top) and tight (bottom) operating points. Only valid b-jet fractions $f_{b-\text{jets}}$ are used to define the final discriminants of the individual algorithms. The rejection factors are calculated using jets reconstructed in simulated $t\bar{t}$ events from pp collisions at $\sqrt{s}$ = 13 TeV. Each iso-efficiency line corresponds to the c-jet tagging efficiency of either the loose (40% c-jet tagging efficiency) or tight (17% c-jet tagging efficiency) operating points of *DL1* and the markers indicate the same fractions $f_{b-\text{jets}}$ as tuned for those operating points.

of jet p_{T} using the fixed cut calculation. The calculation method is preferred in c-jet tagging for its simplicity and therefore used in the determination of the loose and tight c-jet tagging operating points. The rejection factors for the loose and tight operating points of *DL1* are shown as a function of jet p_{T} in Figures 8.4 and 8.5. As with b-jet tagging, the jet p_{T} dependence is again in agreement with the expected

Operating Point	MV2c(l)100	*DL1*
loose		
c-jet tagging efficiency	41.5%	40%
light-flavour jet rejection factor	19.9	26.6
b-jet rejection factor	4.0	4.0
tight		
c-jet tagging efficiency	17.5%	17%
light-flavour jet rejection factor	190.9	527.1
b-jet rejection factor	18.3	17.9

Table 8.1: Rejection factors for the chosen loose and tight c-jet tagging operating points of *DL1* and the comparison to the rejection factors of the loose and tight operating points defined for MV2c(l)100 [77].

shape of the distribution. For jets with higher p_{T}, statistical uncertainties which are especially visible for the tight operating point are a result of limited statistics in the sample used to evaluate the performance. As before in the case of b-jet tagging, the c-jet tagging performance shows a plateau structure, which is more pronounced for the tight operating point in the mid p_{T} range, from where it also extends into the lower and higher jet p_{T} regions.

From the overall performance it can be concluded that *DL1* is not only the tagging algorithm with the best b-jet tagging algorithm performance but also the best c-jet tagging algorithm performance. This is expected from a larger information input. This performance strongly indicates the simplicity of the multiple purpose flavour tagging algorithm DL1, which not only results in great performance in flavour tagging but at the same time reduces the amount of person power required for tagging algorithm maintenance or re-optimisation in case of updates on object reconstruction or recommendations from relevant combined performance groups which impact the input variables. This is due to the fact that only one DNN training is required to be trained and stored.

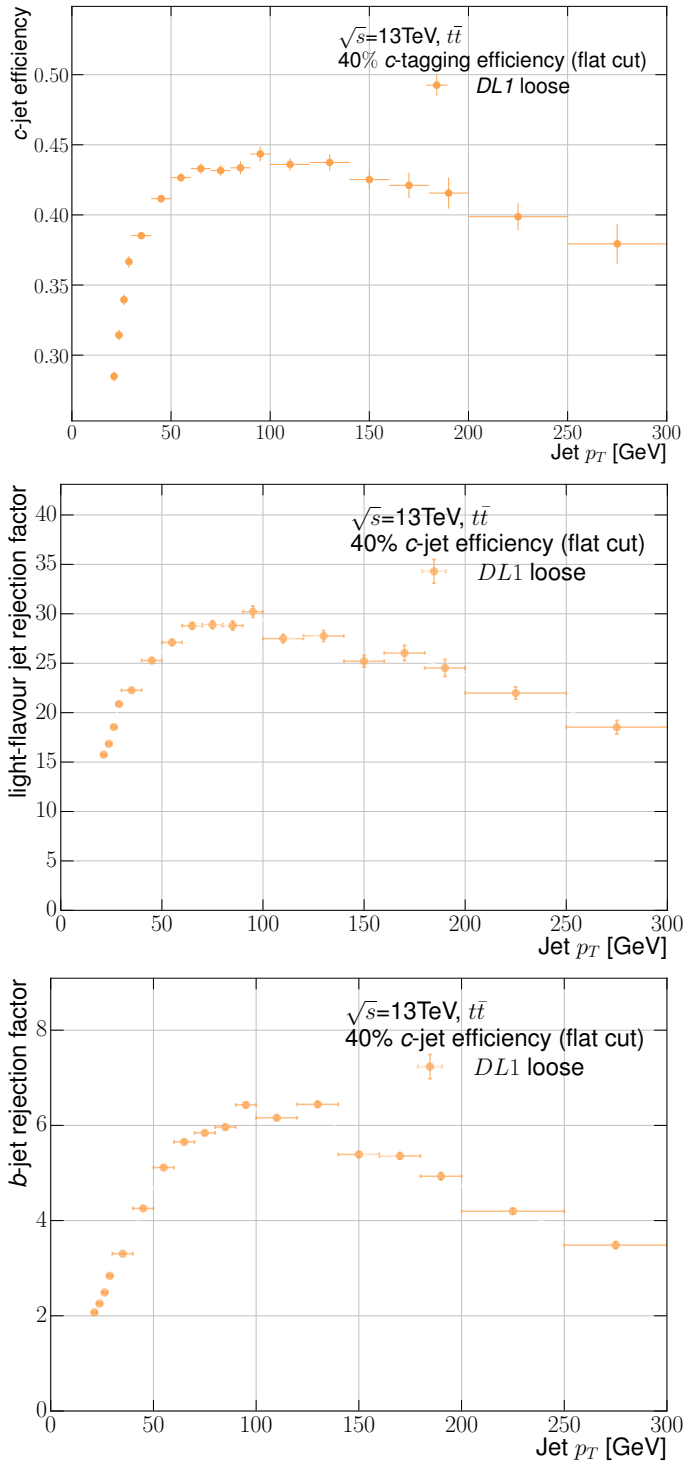


Figure 8.4: The c-jet tagging performance in terms of the jet p_{T} dependence of the c-jet tagging (top), light-flavour jet rejection factors (middle) and b-jet rejection factors (bottom) of *DL1* as calculated on jets reconstructed in simulated $t\bar{t}$ events from pp collisions at $\sqrt{s}$ = 13 TeV using a flat cut calculation for the c-jet tagging efficiency across the indicated jet p_{T} range for the loose operating point. Only statistical and no systematic uncertainties are included in the error calculation.

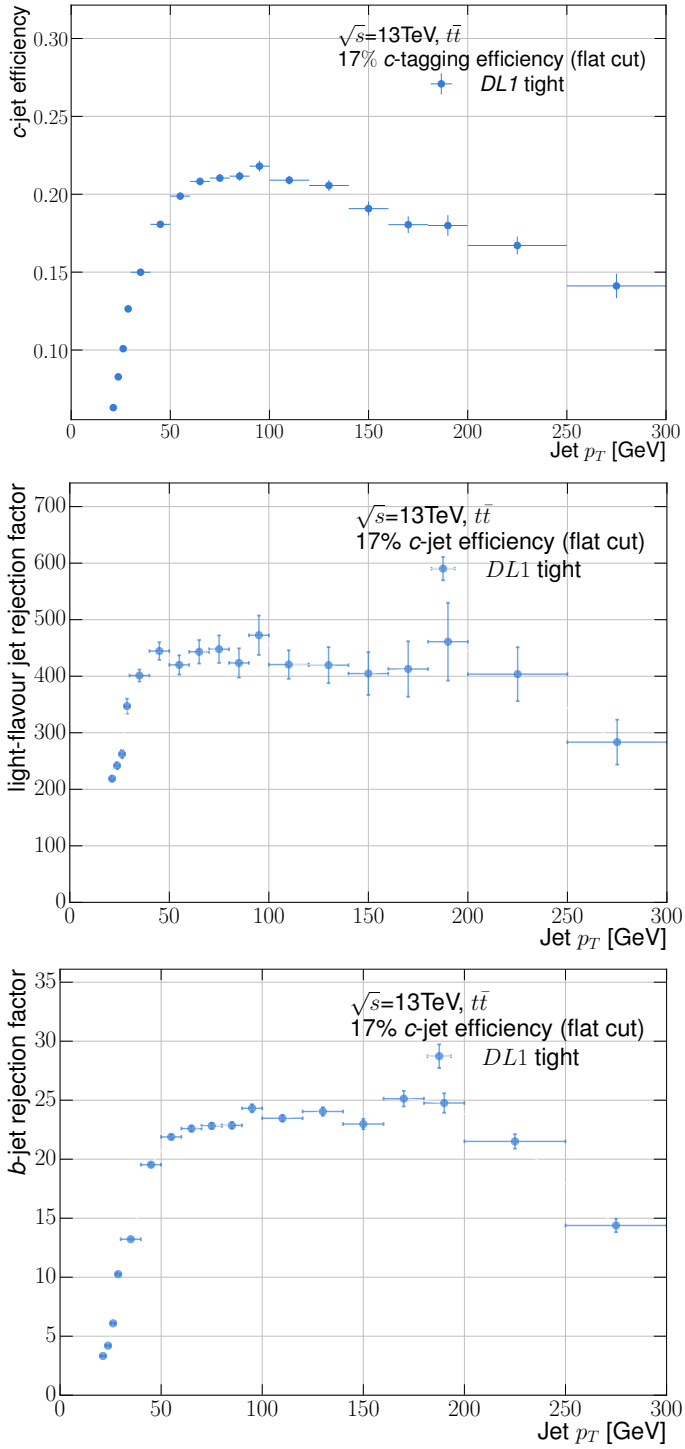


Figure 8.5: The c-jet tagging performance in terms of the jet p_{T} dependence of the c-jet tagging efficiency (top), light-flavour jet rejection factors (middle) and b-jet rejection factors (bottom) of *DL1* for a flat cut across the indicated jet p_{T} range for the tight operating point. The performance is calculated on jets reconstructed in simulated $t\bar{t}$ events from pp collisions at $\sqrt{s}$ = 13 TeV. Only statistical and no systematic uncertainties are included in the error calculation.

8.3 Calibration

In order to qualify the performance and demonstrate that the algorithm has learned the underlying physics, comparisons of the per jet predictions in MC simulations need to be compared to real life collision data. The data-MC agreement of predictions of the DL1 variants using a $t\bar{t}$-dominated selection (left) and a Z $\rightarrow \mu^+\mu^-$+jets-dominated selection of events from pp collision data collected at $\sqrt{s}$ = 13 TeV is very good. This is shown for *DL1* in Figure 8.6 for both the $t\bar{t}$ dominated selection

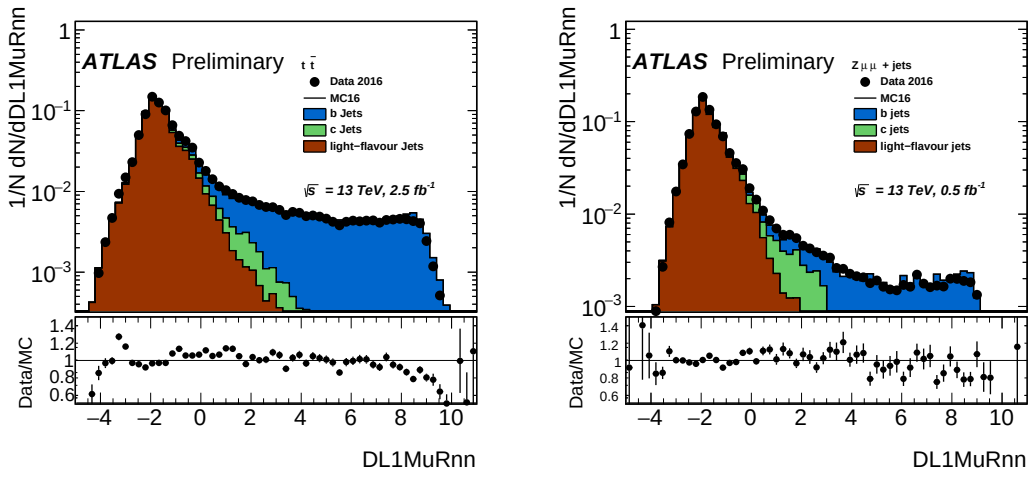


Figure 8.6: Data-MC comparison plots for *DL1* on the predictions for b-, c- and light-flavour jets calculated for a $t\bar{t}$-dominated selection (left) and the Z $\rightarrow \mu^+\mu^-$+jets-dominated selection (right) [67].

as well for the jets of higher jet p_{T} in the Z $\rightarrow \mu^+\mu^-$+jets-dominated selection. Besides larger uncertainties for bins with lower-statistics, only slight disagreement is observed at the starts and tails of the prediction distributions. However, the origins of these can be understood in part to be a result of the MC subtraction methods which result in an imprecise MC prediction estimate. This contributes additional uncertainty, but it is a requirement as it is impossible to select a sample of collision data which only contains one interaction process of the colliding protons. The slight disagreement can also originate from a slight generator mismodelling of the data, which is also expected. Due to this generally good agreement the jet p_{T} dependent b- and c-jet tagging efficiency scale factors, which are used to bring the predictions in line with real collision data, are expected to be close to one. The fact that the agreement is this good when considering not only $t\bar{t}$ but also different topologies such as Z + jets events shows that the DL1 tagging algorithms generalise jet topologies well independent of the event topology. This means that the DL1 tagging

algorithms picked up on the underlying physics of the jets and learned jet information as intended. It is also an indicator that the algorithms generalise well enough not to display exceedingly worse behaviour where there is generator mismodelling. Otherwise this would indicate that instead of being robust to small variations in the values of the attributes, which can be expected when moving to predicting on collision data, the algorithm picked up too much on generator specific features in the attributes or even statistical fluctuations in the training data, which is part of why the monitoring and the initial setup of a well balanced training set is of such high importance. This undesired behaviour would not necessarily be visible in the loss development on the training and validation set or the figures of merit in the simulated test set. In that case the algorithm would be an unreliable toy model, which is of little use for physics analyses, both precision measurements and searches, which require generalisation beyond the phase space of the training regime.

Unfortunately, only flavour tagging algorithms without SMu or SMT information are currently able to be calibrated. Including the muon information in the calibration is ongoing work. Therefore, only *DL1baseline* is currently calibrated for use in physics analyses.

The scale factors for *DL1baseline* at the 60%, 70%, 77% and 85% b-jet tagging efficiencies are shown as a function of jet p_{T} in Figure 8.7 and 8.8 [85]. Overall the b-jet tagging efficiency scale factors are very close to one with excellent proximity to one for the higher jet p_{T} regions across all relevant b-jet tagging efficiencies.

Due to the usage of muon information in MV2c(l)100, *DL1baseline* is currently the only c-jet tagging algorithm in ATLAS flavour tagging, which is able to be calibrated. However, the full calculation of the calibrations has not yet been performed.

However, based on the available b-jet tagging efficiency scale factors, it can be concluded that the DL1 tagging algorithms provide a robust, reliable and therefore trustworthy performance. This makes DL1 a fully-fledged and trustworthy high-level flavour tagging algorithm family for use in all physics analyses performed within the ATLAS collaboration.

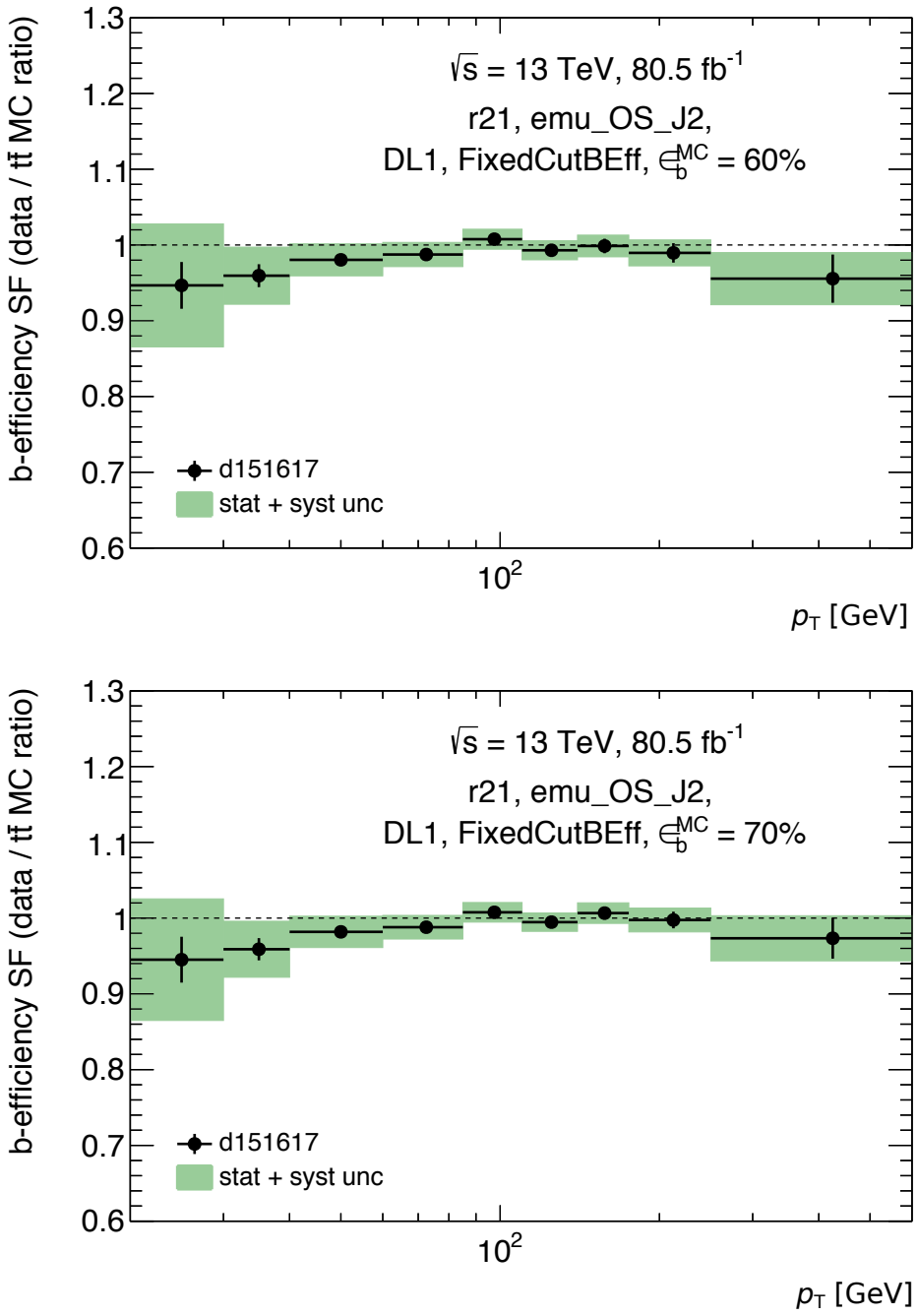


Figure 8.7: The b-jet tagging efficiency scale factors (SF) for *DL1baseline* as a function of jet p_{T} for of 60% (top) and 70% (bottom) b-jet tagging efficiencies [85]. The SF are derived using the combined full collision datasets collected in the years 2015 to 2017.

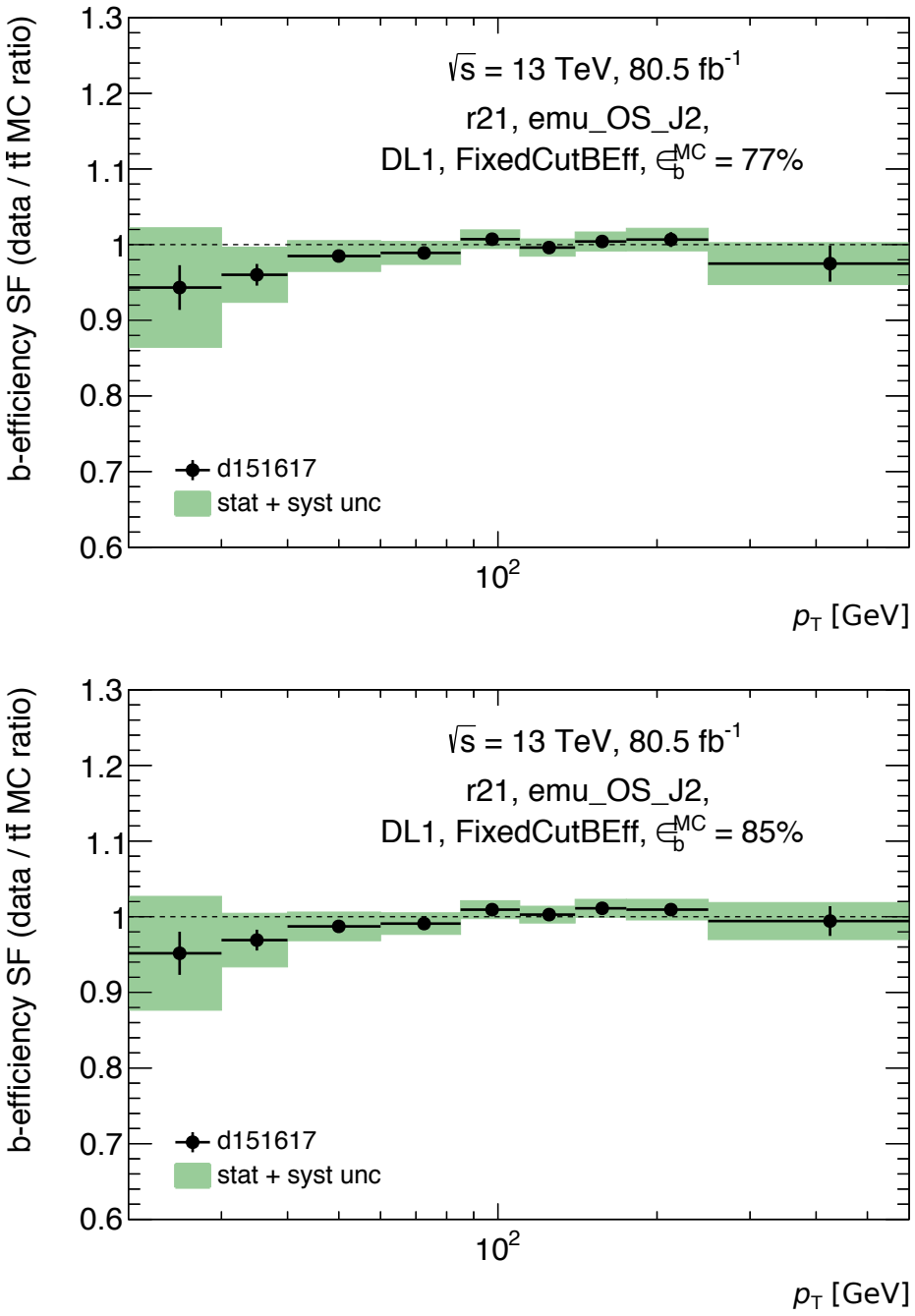


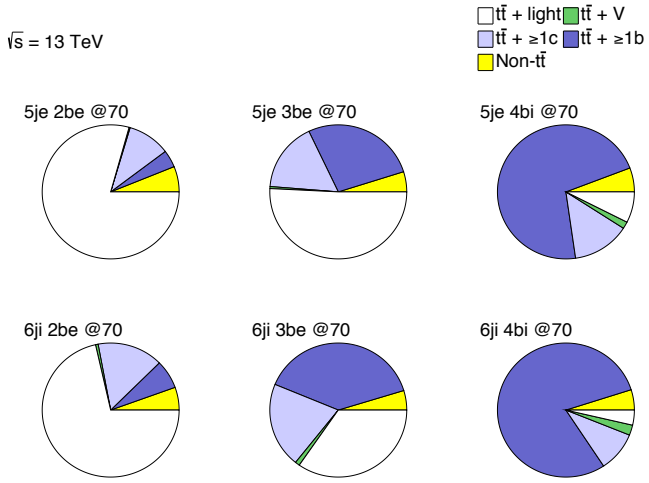
Figure 8.8: The b-jet tagging efficiency scale factors (SF) for *DL1baseline* as a function of jet p_{T} for of 77% (top) and 85% (bottom) b-jet tagging efficiencies [85]. The SF are derived using the combined full collision datasets collected in the years 2015 to 2017.

9. Conclusion and Outlook

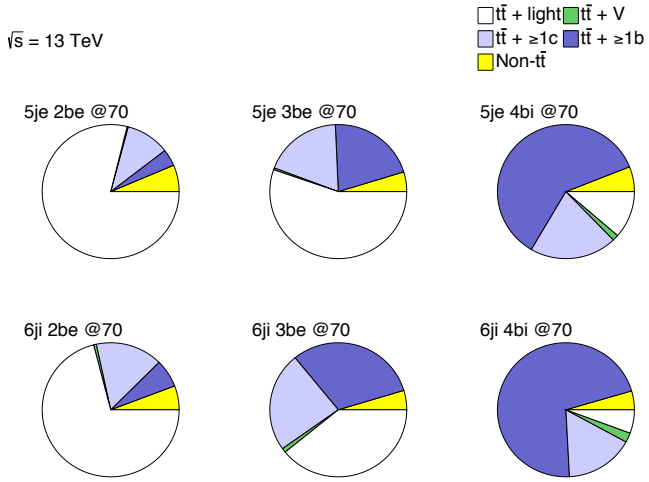
Contents

A new family of flavour tagging algorithms has been presented. Three different variants to provide b- and c-jet tagging algorithms for the ATLAS collaboration are employed and available for use in physics analyses with recommendations provided by the flavour tagging performance group. DL1 is recommended on par with the previous high-level b-jet tagging algorithms baseline MV2. *DL1baseline* currently provides the only c-jet tagging algorithm which is able to be calibrated and used in analyses. The calibration of the three DL1 variants for b- and c-jet tagging is available to be used in ATLAS analyses on pp collision data at $\sqrt{s}$ = 13 TeV. The significance of the impact of the performance improvements provided by the DL1 algorithms and is investigated by a large fraction of physics analyses within the ATLAS collaboration.

An example is the application of the *DL1baseline* b-jet tagging algorithm using an flat cut operating point corresponding to 70% b-jet tagging efficiency in the $t\bar{t}H\left(H \rightarrow b\bar{b}\right)$ analysis. The choice of 70% b-jet tagging efficiency reflects the nominal flavour tagging operating point of the analysis. This high jet multiplicity analysis is highly reliant on flavour tagging as in the single lepton channel a total of six jets of which four should be b-tagged are expected. The $t\bar{t}H\left(H \rightarrow b\bar{b}\right)$ process is expected to result in a reconstructed physics object signature containing one b-jet from each semileptonic top-quark decay, and a $b\bar{b}$ pair coming from the decay of the Higgs boson. The analysis benefits in important areas of the high jet multiplicity phase space from the separation power provided by *DL1baseline*, which enables the analysis team to reduce signal region contamination from $t\bar{t}$ production in association with additional light-flavour or c-jets in particular. This can be seen in



(a) Application of the *DL1baseline* b-jet tagging algorithm



(b) Application of an alternative high-level b-jet tagging algorithm.

Figure 9.1: Pie charts showing the background contributions from different physics processes in regions defined by jet and b-jet tagging multiplicities. Either the *DL1baseline* b-jet tagging algorithm (a) or a corresponding alternative high-level b-jet tagging algorithm (b) is used, both employing an operating point corresponding to a b-jet tagging efficiency of 70% [86]. The top rows show regions containing exactly five jets (5je) and the bottom rows show regions with at least six jets (6ji). The columns, moving from left to right, show the regions which have exactly two (2be), exactly three (3be) or at least four (4bi) b-tagged jets. The most signal enriched regions are those containing at least four b-tagged jets matching the expected reconstructed objects in the $t\bar{t}H\left(H \to b\bar{b}\right)$ process [87].

Figure 9.1, which shows the background composition in regions defined by jet and b-jet tagging multiplicity. Of special interest are the pie charts corresponding to a selection closest to the signal, which encompasses regions including three exclusive b-tagged jets (3be) as well as four inclusive b-tagged jets (4bi). In these regions the dominant background is expected to come from $t\bar{t}+ \geq$ 1b, which matches the final state of $t\bar{t}H\left(H \rightarrow b\bar{b}\right)$. However, in addition there are contributions from $t\bar{t}+$ light and $t\bar{t}+ \geq$ 1c due to c- and light-flavour jets passing the operating point requirements and therefore are labelled as b-tagged jets. In Figure 9.1 it can be seen that the $t\bar{t}+$ light and especially $t\bar{t}+ \geq$ 1c backgrounds are largely reduced when using DL1 in comparison to the corresponding alternative high-level b-jet tagging algorithm. This result is a direct consequence of the improved light-flavour and c-jet rejection factors of *DL1baseline* at a 70% b-jet tagging efficiency. The reduction of the $t\bar{t}+ \geq$ 1c contribution in the signal regions was one of the main experimental challenges of the previous iteration of the analysis, as shown in Ref. [87].

The careful preprocessing procedure including the two-dimensional weighting to the b-jet distribution, whose population is smoother in the hybrid sample for jets with jet p_{T} of $\sim$ 200 GeV than for c- or light-flavour jets, results in a more attractive performance as a function of jet p_{T} for analyses in comparison to algorithms in the MV2 family. In addition, as expected from the careful preprocessing and inclusion of generalisation techniques in the training of each DNN, the DL1 algorithms extrapolate well to higher jet p_{T}. This is apparent in the high jet p_{T} regime above 1 TeV, where the light-flavour jet rejection factors of the MV2 b-jet tagging algorithms display non-continuous behaviour and decrease with increasing jet p_{T} in contrast to the corresponding DL1 variants presented in this thesis. This regime is of high interest for heavy resonance searches such as $Z' \rightarrow\ b\bar{b}$, where the robustness and understanding of the tagging performance is critical.

For these reasons it is crucial to point out the importance of the DL1 preprocessing as well as using multiple figures of merit to evaluate the overall performance. Both are extremely valuable in determining the final optimised DL1 version to be applied to collision data. One of the most important aspects of the preprocessing is that it counteracts the class prior problem and prepares the MC simulation data to match the applied classification algorithm. In addition to the preprocessing, as has been shown, a single figure of merit is not able to encompass the complexity of the task and evaluate the desired performance. Determining the full set of important figures of merit is one of the crucial first steps in the development of DL1. It is therefore important to keep in mind that it is this carefully designed sequence of steps that provides the foundation of DL1 toward the final family of algorithms. The

success of this family of algorithms is to a large extent based on this foundation.

As DL1 is based on supervised learning, potential performance improvements are possible using more precise labelling of the MC training data. Improvements in the modelling of processes as well as a better physics understanding in the simulation of the MC data and more precise generation of processes are expected to provide a simulation, which describes collision data with higher accuracy. This improved picture of nature would enable a more precise labelling in the simulated data and therefore enables the DL1 algorithms to provide predictions on collision data, which closer reflect nature due to less association errors in the MC training data.

Algorithms in the DL1 family are designed carefully and their performance is monitored towards the aim of providing a robust and reliable flavour tagging algorithm which is well optimised, robust towards deviations in the expected MC simulated training data and capable of generalisation of the underlying physics. The studies shown in this thesis show that this aim has been achieved and the high-level flavour tagging algorithms are now established within the collaboration with the added success of having gained equal recommendation to analyses by the flavour tagging performance group. The presented data-MC comparisons show very good agreement and the MC scale factors derived using real pp collision data recorded by the ATLAS detector during Run 2 at $\sqrt{s}$ = 13 TeV are consistent with one even up to high jet p_{T} for all operating points of interest. The presented performance of DL1 on jets from simulated $t\bar{t}$ events can therefore be considered to be the same on ATLAS pp collision data at $\sqrt{s}$ = 13 TeV during the Run 2 data taking period.

9.1 Outlook: Ideas on Developing DL1 Further

Several aspects of DL1 can be investigated for potential performance improvements. They go partially beyond the ramifications of the first proof of principle of the method where restrictions are imposed in order to treat all high-level flavour tagging algorithms the same and provide them with the same information content. Now that DL1 has proven that it can hold itself up to the previously existing methods, the full potential of the DL1 method can be explored in more depth.

Before discussing the potential changes to DL1, it should be pointed out that it would be largely beneficial to calibrate the DL1 outputs directly rather than the defined final discriminant which requires tuning and re-calibration already for b-jet tagging and two operating points for c-jet tagging, which results in three different final discriminants to calibrate. For the same amount of calibrated variables, the outputs of the DL1 output nodes p_b, p_c and $\mathrm{p}_{\mathrm{light-flavour}}$ could be calibrated. This

would lead to highly improved flexibility for adaptation within physics analyses, as the values of $f_{c-\text{jets}}$ and $f_{b-\text{jets}}$ in the final discriminants of the b- and c-jet tagging algorithms could be optimised per use case. Any potential configuration requested by analyses which would benefit by a change would have access to a calibration and be able to tune their flavour tagging for their own use case. This could improve sensitivity in all areas of physics using flavour tagging information and especially help pushing the significances of observations like the $H \to b\bar{b}$ decay observation, which are heavily reliant upon flavour tagging.

An obvious area for improvement would be to include the inputs to the low-level algorithms of flavour tagging directly. However, in order to have a direct comparison to the MV2 method, the same variables were used to provide an apples to apples comparison of first principles of the two methods. It would have to be carefully checked whether the physics content communicated by a DL1 algorithm trained on these variables would still show the same robustness, generalisation and agreement with experimental collision data as seen for the current implementation. The additional variables would include hit-based variables and the performance under addition of jet substructure variables could be investigated as well. Additionally, the training of the RNN of the RNNIP low-level algorithm could be directly included in the DL1 DNN by merging the architectures.

Even without adding new variables or variables which were used by low-level algorithms, an optimisation of the current set of input variables could also be considered. This could be done in order to achieve a set of more orthogonal information content and higher information density.

Furthermore, the construction of a well modelled MC simulated training sample, which includes for example gluon or double-b hadron labels, might be possible. Then DL1 could be extended to more than three outputs but the same MC generator would be needed as not to learn generator differences between jet flavours. While additional labels would extend the use case of DL1 and reduce overall required person power, the b- and c-jet tagging algorithm performances should not be degraded in the process. Therefore, additional studies would be needed if it is beneficial for b- and c-jet tagging which remains of primary interest. In case of a performance degradation, it might be a better approach to train individual DNNs for different sets of labels. It should be kept in mind when thinking about this addition of new labels that different labels might be best modelled with different generators or generator setting. For example, this is the case for τ-jets whose decays are known not to be well modelled with EvtGen, which specialises on modelling hadronic decays, but better with different hadronisation generators. Due to MC-MC scale factors, which

point out disagreements between different generators, it is known that these generators introduce features, which when mixed in a DNN training might be picked up upon in combination with the jet topology of different flavours. This would be highly undesirable as it would lead to increased MC-MC scale factors, worse agreement to collision data and therefore a reduced sample statistics in physics analysis by increasing the flavour tagging related MC-MC scale factor weights. Therefore, this requires a careful approach, which is not to be taken lightly.

In a few cases the distributions of the discriminants provided by the low-level algorithms of flavour tagging are exhibiting tails, which for few cases results in jets still having a value for this attribute which is of a higher value compared to the majority of the attribute values. This could be shortened by applying a log transform before the application of a scale and offset. Doing so might potentially improve the performance.

The pile-up profile of the training set used for DL1 is currently flat. Instead, one could train on a pile-up profile which matches the profile expected of events over the intended data taking period so that the DL1 NN becomes aware of the changed pile-up situation and is aware of small changes when moving to higher pile-up. This adds additional complications of its own. Besides unforseen conditions and the wish for a general well working tagging algorithm, including this is a balancing act. In order to make the DL1 DNN aware of the pile-up situation, the pile-up parameter can be parametrised and taken into account for example in the weighing procedure during preprocessing in order to not prioritise the jets of a certain pile-up against others. Following another approach, the pile-up dependence could also be mitigated using an adversary discriminator to train out the dependence of the performance on the pile-up situation. The final approach will, however, heavily depend on empirical results.

Training of DL1 on other jet collections is currently in progress by the flavour tagging performance group. This includes Particle Flow jets [88] as well as jets constructed using ID tracks instead of topo-clusters as inputs to the anti-k_t clustering algorithm.

In addition, the development and support of the Theano libraries has been terminated. However, TensorFlow [89] libraries as well as NN architectures and optimisation tools remain under active development. Their availability within Keras or by direct access using TensorFlow is constantly being updated. New architectures and optimisation strategies could be investigated in their use for DL1 performance optimisation and generalisation.

Considering the wide spectrum of potential developments, it is important to ap-

proach the next steps in the DL1 development wisely. The proper preparation of a solid training set, the monitoring and cross checks of figures of merit as well as good generalisation capabilities are crucial for a reliable performance. For all changes the basic principle of the added steps need to be questioned for feasibility, theoretical sense and potential impact. It is very important to keep the full picture in mind. While some of the proposed changes seem trivial, they might turn out more complicated in praxis and care must be taken in further developments to deliver a robust and reliable tagging algorithm. Also the person power, estimated development time and complexity need to be taken into consideration in relation to the potential overall gain when prioritising one over the other. Long term research and development is needed and should be encouraged. However, solid short-term improvements are also a priority. These improvements leading to an increase in performance of the DL1 family then directly feed into potential improvements within analyses.

Overall, DL1 has demonstrated excellent performance as well as agreement with recorded collision data. With more physics analyses investigating the use of DL1, its benefits and the potential improvements to sensitivity and precision of both measurements and searches will be ascertained as more results become available. Going forwards, keeping an open mind and sensibly incorporating new ideas with the full picture in mind will continue to aid and improve analyses and push onwards in the hunt for new physics.

References

[1] Lanfermann, M. C. *Deep Learning in Flavour Tagging at the ATLAS experiment*. Tech. rep. ATL-PHYS-PROC-2017-191. Geneva: CERN, Oct. 2017. URL: https://cds.cern.ch/record/2287551.

[2] ATLAS Collaboration. *Deep Neural Network based higher level flavour tagging algorithm at the ATLAS experiment*. PUB Note in Preparation. 2019. URL: https://cds.cern.ch/record/2290144.

[3] ATLAS Collaboration. *The ATLAS Experiment at the CERN Large Hadron Collider*. In: *JINST* 3 (2008), S08003.

[4] *DeepL*. URL: https://www.deepl.com/en/translator (visited on 02/27/2019).

[5] Guest, D., Lanfermann, M., makagan, Paganini, M. and Smith, J. *Lwtnn/Lwtnn: Version 2.2*. 2017. URL: https://zenodo.org/record/820616.

[6] Collaboration, L. *LHCb VELO Upgrade Technical Design Report*. Tech. rep. CERN-LHCC-2013-021. LHCB-TDR-013. Nov. 2013. URL: https://cds.cern.ch/record/1624070.

[7] Aaboud, M. et al. *Observation of $H \to b\bar{b}$ decays and VH production with the ATLAS detector*. In: *Phys. Lett.* B786 (2018), pp. 59–86. arXiv: 1808.08238 [hep-ex].

[8] Griffiths, D. *Introduction to Elementary Particles*. 2nd edition. Wiley-VCH, 2008.

[9] Schwartz, M. D. *Quantum Field Theory and the Standard Model*. Cambridge University Press, 2014. ISBN: 1107034736, 9781107034730.

[10] A. Purcell, *Go on a particle quest at the first CERN webfest*. URL: https://cds.cern.ch/journal/CERNBulletin/2012/36/News%5C%20Articles/1473657 (visited on 02/04/2019).

[11] Wigner, E. P. *Gruppentheorie und ihre Anwendung auf die Quantenmechanik der Atomspektren*. Springer, 1931. ISBN: 978-3-663-02555-9.

[12] Weinberg, S. *The Quantum theory of fields. Vol. 1: Foundations*. Cambridge University Press, 2005. ISBN: 9780521670531, 9780511252044.

[13] Noether, E. *Invariant Variation Problems*. In: *Gott. Nachr.* 1918 (1918). [Transp. Theory Statist. Phys.1,186(1971)], pp. 235–257. arXiv: physics/0503066 [physics].

[14] Weinberg, S. *A model of leptons*. In: *Phys. Rev. Lett.* 19 (1967), p. 1264.

[15] Tanabashi, M. et al. *Review of Particle Physics*. In: *Phys. Rev. D* 98 (3 Aug. 2018), p. 030001. URL: https://link.aps.org/doi/10.1103/PhysRevD.98.030001.

[16] Grange, J. et al. *Muon (g-2) Technical Design Report*. In: (2015). arXiv: 1501.06858 [physics.ins-det].

[17] Kobayashi, M. and Maskawa, T. *CP-Violation in the Renormalizable Theory of Weak Interaction*. In: *Progress of Theoretical Physics* 49.2 (Feb. 1973), pp. 652–657. ISSN: 0033-068X. eprint: http://oup.prod.sis.lan/ptp/article-pdf/49/2/652/5257692/49-2-652.pdf. URL: https://dx.doi.org/10.1143/PTP.49.652.

[18] ATLAS Collaboration. *Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC*. In: *Phys. Lett. B* 716 (2012), p. 1. arXiv: 1207.7214 [hep-ex].

[19] Krauss, Frank. *QCD & Monte Carlo Event Generators*. URL: https://www.ippp.dur.ac.uk/~krauss/Lectures/MonteCarlos/Heidelberg14.pdf (visited on 02/04/2019).

[20] Buckley, A. et al. *General-purpose event generators for LHC physics*. In: *Phys. Rept.* 504 (2011), pp. 145–233. arXiv: 1101.2599 [hep-ph].

[21] Sjöstrand, T. et al. *An Introduction to PYTHIA 8.2*. In: *Comput. Phys. Commun.* 191 (2015), pp. 159–177. arXiv: 1410.3012 [hep-ph].

[22] Patrignani, C. et al. *Review of Particle Physics*. In: *Chin. Phys.* C40.10 (2016), p. 100001.

[23] Lange, D. J. *The EvtGen particle decay simulation package*. In: *Nucl. Instrum. Meth.* A462 (2001), pp. 152–155.

[24] Bahr, M. et al. *Herwig++ Physics and Manual*. In: *Eur. Phys. J.* C58 (2008), pp. 639–707. arXiv: 0803.0883 [hep-ph].

[25] Gleisberg, T. et al. *Event generation with SHERPA 1.1*. In: *JHEP* 0902 (2009), p. 007. arXiv: 0811.4622 [hep-ph].

[26] Brüning, O. S. et al. *LHC Design Report*. CERN Yellow Reports: Monographs. Geneva: CERN, 2004.

[27] Mobs, E. *The CERN accelerator complex. Complexe des accélérateurs du CERN*. General Photo. July 2016. URL: https://cds.cern.ch/record/2197559.

[28] ATLAS Collaboration. *Luminosity determination in pp collisions at $\sqrt{s} = 8$ TeV using the ATLAS detector at the LHC*. In: (2016). arXiv: 1608.03953 [hep-ex].

[29] Meer, S. van der. *Calibration of the effective beam height in the ISR*. Tech. rep. CERN-ISR-PO-68-31. ISR-PO-68-31. Geneva: CERN, 1968. URL: http://cds.cern.ch/record/296752.

[30] Rubbia, C. *Measurement of the luminosity of $p-overlinep$ collider with a (generalized) Van der Meer Method*. Tech. rep. CERN-p$\overline{p}$-Note-38. Geneva: CERN, Nov. 1977. URL: http://cds.cern.ch/record/1025746.

[31] CERN. *LHC Overview Panel*. URL: http://acc-stats.web.cern.ch/acc-stats/#lhc/overview-panel (visited on 02/27/2018).

[32] ATLAS Collaboration. *The ATLAS Experiment at the CERN Large Hadron Collider*. In: *JINST* 3 (2008), S08003.

[33] Pequenao, J. *Computer generated image of the whole ATLAS detector*. Mar. 2008. URL: http://cds.cern.ch/record/1095924.

[34] Pequenao, J. *Computer generated image of the ATLAS inner detector*. Mar. 2008. URL: https://cds.cern.ch/record/1095926.

[35] ATLAS Collaboration. *ATLAS Insertable B-Layer Technical Design Report*. In: *ATLAS-TDR-19* (2010). https://cds.cern.ch/record/1291633.

[36] *ATLAS Insertable B-Layer Technical Design Report Addendum*. Tech. rep. CERN-LHCC-2012-009. ATLAS-TDR-19-ADD-1. Addendum to CERN-LHCC-2010-013, ATLAS-TDR-019. May 2012. URL: https://cds.cern.ch/record/1451888.

[37] Pequenao, J. *Computer Generated image of the ATLAS calorimeter*. Mar. 2008. URL: https://cds.cern.ch/record/1095927.

[38] Pequenao, J. *Computer generated image of the ATLAS Muons subsystem*. Mar. 2008. URL: https://cds.cern.ch/record/1095929.

[39] ATLAS Collaboration. *Luminosity Monitoring in ATLAS with MPX Detectors*. ATLAS-CONF-2013-103. 2013. URL: https://cds.cern.ch/record/1604286.

[40] Cornelissen, T. et al. *Concepts, Design and Implementation of the ATLAS New Tracking (NEWT)*. In: (2007). Ed. by A. Salzburger.

[41] ATLAS Collaboration. *Track Reconstruction Performance of the ATLAS Inner Detector at $\sqrt{s} = 13$ TeV*. ATL-PHYS-PUB-2015-018. 2015. URL: https://cds.cern.ch/record/2037683.

[42] ATLAS Collaboration. *A neural network clustering algorithm for the ATLAS silicon pixel detector*. In: *JINST* 9 (2014), P09009. arXiv: 1406.7690 [hep-ex].

[43] ATLAS Collaboration. *Performance of primary vertex reconstruction in proton–proton collisions at $\sqrt{s} = 7$ TeV in the ATLAS experiment*. ATLAS-CONF-2010-069. 2010. URL: https://cds.cern.ch/record/1281344.

[44] ATLAS Collaboration. *Vertex Reconstruction Performance of the ATLAS Detector at $\sqrt{s} = 13$ TeV*. ATL-PHYS-PUB-2015-026. 2015. URL: https://cds.cern.ch/record/2037717.

[45] Cacciari, M., Salam, G., and Soyez, G. *The anti-k_t jet clustering algorithm*. In: *JHEP* 04 (2008), p. 063. arXiv: 0802.1189 [hep-ph].

[46] Salam, G. P. *Towards Jetography*. In: *Eur. Phys. J.* C67 (2010), pp. 637–686. arXiv: 0906.1833 [hep-ph].

[47] ATLAS Collaboration. *Jet energy scale measurements and their systematic uncertainties in proton–proton collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector*. In: *Phys. Rev. D* 96 (2017), p. 072002. arXiv: 1703.09665 [hep-ex].

[48] Cacciari, M. and Salam, G. P. *Pileup subtraction using jet areas*. In: *Phys. Lett.* B659 (2008), pp. 119–126. arXiv: 0707.1378 [hep-ph].

[49] ATLAS Collaboration. *Performance of pile-up mitigation techniques for jets in pp collisions at $\sqrt{s} = 8$ TeV using the ATLAS detector*. In: (2015). arXiv: 1510.03823 [hep-ex].

[50] ATLAS Collaboration. *JVT 2016 performance plots*. 2016. URL: https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/JETM-2016-011/ (visited on 02/18/2019).

[51] *Electron and photon reconstruction and performance in ATLAS using a dynamical, topological cell clustering-based approach*. Tech. rep. ATL-PHYS-PUB-2017-022. Geneva: CERN, Dec. 2017. URL: https://cds.cern.ch/record/2298955.

[52] Aaboud, M. et al. *Electron efficiency measurements with the ATLAS detector using 2012 LHC protonproton collision data*. In: *Eur. Phys. J.* C77.3 (2017), p. 195. arXiv: 1612.01456 [hep-ex].

[53] Sampsonidou, D. *Precise determination of the muon reconstruction efficiency in ATLAS at Run-II*. In: *EPJ Web Conf.* 164 (2017), p. 08006.

[54] Aad, G. et al. *Muon reconstruction performance of the ATLAS detector in protonproton collision data at $\sqrt{s}$ =13 TeV*. In: *Eur. Phys. J.* C76.5 (2016), p. 292. arXiv: 1603.05598 [hep-ex].

[55] collaboration, T. A. *Measurement of the tau lepton reconstruction and identification performance in the ATLAS experiment using pp collisions at $\sqrt{s} = 13$ TeV*. In: (2017).

[56] collaboration, T. A. *E_T^{miss} performance in the ATLAS detector using 2015-2016 LHC p-p collisions*. In: (2018).

[57] Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, 2006.

[58] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[59] Breiman, L. et al. *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis, 1984.

[60] Quinlan, J. R. *Induction of Decision Trees*. In: *MACH. LEARN* 1 (1986), pp. 81–106.

[61] Freund, Y. and Schapire, R. E. *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*. In: *Journal of Computer and System Sciences* 55.1 (1997), pp. 119–139. ISSN: 0022-0000.

[62] Kingma, D. P. and Ba, J. *Adam: A Method for Stochastic Optimization*. In: *CoRR* abs/1412.6980 (2014). arXiv: 1412.6980. URL: http://arxiv.org/abs/1412.6980.

[63] Glorot, X. and Bengio, Y. *Understanding the difficulty of training deep feedforward neural networks*. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Y. W. Teh and M. Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. URL: http://proceedings.mlr.press/v9/glorot10a.html.

[64] Srivastava, N. et al. *Dropout: a simple way to prevent neural networks from overfitting*. In: *Journal of Machine Learning Research* 15.1 (2014), pp. 1929–1958.

[65] Ioffe, S. and Szegedy, C. *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. In: *CoRR* abs/1502.03167 (2015). arXiv: 1502.03167. URL: http://arxiv.org/abs/1502.03167.

[66] Bergstra, J., Yamins, D., and Cox, D. D. *Making a Science of Model Search*. In: *arXiv e-prints*, arXiv:1209.5111 (Sept. 2012), arXiv:1209.5111. arXiv: 1209.5111 [cs.CV].

[67] ATLAS Collaboration. *Optimisation and performance studies of the ATLAS b-tagging algorithms for the 2017-18 LHC run*. ATL-PHYS-PUB-2017-013. 2017. URL: https://cds.cern.ch/record/2273281.

[68] ATLAS Collaboration. *ATLAS Run 1 Pythia8 tunes*. ATL-PHYS-PUB-2014-021. 2011. URL: http://cdsweb.cern.ch/record/1966419.

[69] Sjostrand, T., Mrenna, S., and Skands, P. Z. *A Brief Introduction to PYTHIA 8.1*. In: *Comput. Phys. Commun.* 178 (2008), p. 852. arXiv: 0710.3820 [hep-ph].

[70] Alioli, S. et al. *A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX*. In: *JHEP* 1006 (2010), p. 043. arXiv: 1002.2581 [hep-ph].

[71] GEANT4 Collaboration, S. Agostinelli et al. *GEANT4: A Simulation toolkit*. In: *Nucl. Instrum. Meth. A* 506 (2003), p. 250.

[72] ATLAS Collaboration. *Identification of Jets Containing b-Hadrons with Recurrent Neural Networks at the ATLAS Experiment*. ATL-PHYS-PUB-2017-003. 2017. URL: https://cds.cern.ch/record/2255226.

[73] ATLAS Collaboration. *Optimisation of the ATLAS b-tagging performance for the 2016 LHC Run*. ATL-PHYS-PUB-2016-012. 2016. URL: https://cds.cern.ch/record/2160731.

[74] ATLAS Collaboration. *Secondary vertex finding for jet flavour identification with the ATLAS detector*. ATL-PHYS-PUB-2017-011. 2017. URL: https://cds.cern.ch/record/2270366.

[75] *Topological b-hadron decay reconstruction and identification of b-jets with the JetFitter package in the ATLAS experiment at the LHC*. Tech. rep. ATL-PHYS-PUB-2018-025. Geneva: CERN, Oct. 2018. URL: https://cds.cern.ch/record/2645405.

[76] Hocker, A. et al. *TMVA - Toolkit for Multivariate Data Analysis*. In: *PoS* ACAT (2007), p. 040. arXiv: physics/0703039 [PHYSICS].

[77] Shlomi, J. Private communication. July 11, 2017.

[78] Aad, G. et al. *Performance of b-Jet Identification in the ATLAS Experiment*. In: *JINST* 11.04 (2016), P04008. arXiv: 1512.01094 [hep-ex].

[79] Neyman, J. and Pearson, E. S. *On the Problem of the Most Efficient Tests of Statistical Hypotheses*. In: *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character* 231 (1933), pp. 289–337. URL: http://www.jstor.org/stable/91247.

[80] *NumPy*. URL: http://www.numpy.org/.

[81] Chollet, F. et al. *Keras*. https://keras.io. 2015.

[82] Theano Development Team. *Theano: A Python framework for fast computation of mathematical expressions*. In: *arXiv e-prints* abs/1605.02688 (May 2016). URL: http://arxiv.org/abs/1605.02688.

[83] *Python Data Analysis Library - pandas*. URL: https://pandas.pydata.org/index.html.

[84] The HDF Group. *Hierarchical Data Format, version 5*. 1997-2019. URL: http://www.hdfgroup.org/HDF5/.

[85] *Measurement of the b-jet identification efficiency with $t\bar{t}$ events using an improved likelihood method*. Paper in Preparation. 2019.

[86] Held, A. Private communication. Feb. 27, 2019.

[87] ATLAS Collaboration. *Search for the Standard Model Higgs boson produced in association with top quarks and decaying into a $b\bar{b}$ pair in pp collisions at $\sqrt{s}$ = 13 TeV with the ATLAS detector*. In: *Phys. Rev. D* (2017). arXiv: 1712.08895 [hep-ex].

[88] Aaboud, M. et al. *Jet reconstruction and performance using particle flow with the ATLAS Detector*. In: *Eur. Phys. J.* C77.7 (2017), p. 466. arXiv: 1703.10485 [hep-ex].

[89] Martn Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.