Simulation of the ATLAS electromagnetic calorimeter using generative adversarial networks and likelihood-free inference of the offshell Higgs boson couplings at the LHC

Aishik Ghosh

Thèse de doctorat

NNT: 2020UPASP058

université PARIS-SACLAY

Simulation of the ATLAS Electromagnetic Calorimeter using Generative Adversarial Networks and Likelihood-Free Inference of the Offshell Higgs Boson Couplings at the LHC

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 576, Particules, Hadrons, Énergie, Noyau, Instrumentation, Imagerie, Cosmos et Simulation (PHENIICS)
Spécialité de doctorat: Physique des Particules
Unité de recherche: Université Paris-Saclay, CNRS, IJCLab, 91405, Orsay, France
Référent: Faculté des sciences d'Orsay

Thèse présentée et soutenue en visioconférence totale, le 13 novembre 2020 par

Aishik GHOSH

Composition du jury:

Marie-Helene Schune Directrice de Recherche, CNRS, IJCLab, Université Paris Saclay	Présidente
Isabelle Wingerter-Seez Directrice de Recherche, CNRS, CPPM, Université Aix Marseille	Rapporteur & Examinateur
Maurizio Pierini Chercheur, CERN	Rapporteur & Examinateur
Tilman Plehn Professeur, Universität Heidelberg	Examinateur
Glen Cowan Professeur, Royal Holloway, University of London	Examinateur
Danilo J. Rezende Chercheur Senior, Google DeepMind	Examinateur
David Rousseau Directeur de Recherche, CNRS, IJCLab, Université Paris Saclay	Directeur de Thèse

Contents

Chapter 1

Introduction

Contents

1.1 The Broad Picture

Particle physics is the study of the smallest, fundamental building blocks of the universe, and the properties studied have consequences at the largest scale, cosmology. Our current understanding of the universe is incomplete, with no clear explanation for Dark Energy, Dark Matter, neutrino mass, the amount of Matter-Antimatter asymmetry seen in the universe and gravity from a quantum perspective. Before 2012 there was also no experimental confirmation for the Brout-Englert-Higgs (BEH) mechanism, which we now know to be the answer to how several of the fundamental particles obtain their masses.

To find answers to some of these questions, particle physicists endeavoured on a 20 year mission to build the Large Hadron Collider (LHC), the world's most powerful particle accelerator, at CERN on the border of Switzerland and France, near Geneva. As the world's largest machine, this circular collider accelerates proton beams in opposite directions and smashes them together at 'interaction plots' where large detectors are setup to record the new particles thrown out in all directions due to the high energy collision. Two sister experiments, ATLAS and CMS were built on either side of the circular ring, to independently measure properties of our quantum universe. The original purpose was to involve more particle physicists and create a healthy competition with the hope that it would improve results, however, in light of the recent crisis in reproducibility of research, this turns out be all the more necessary to reaffirm the validity of their results. The first proposals to search for experimental confirmation of the BEH mechanism began in the 1980s but this multi-decade endeavour remained unsuccessful until the LHC was built, culminating on 4$^{\text{th}}$ July 2012 with a joint announcement from the ATLAS and CMS experiments on the discovery[1, 2] of the Higgs boson particle, almost 50 years after its first proposals.

The LHCb and ALICE experiments were also setup on the LHC, to specialise in b quark physics and the study of strongly interacting matter at high energy densities respectively.

Since then, despite recording more data than originally planned, the ATLAS, CMS experiments have not found evidence for new physics that might give a hint to demystify any of the as yet

unsolved problems in physics. Running this machine is costly, therefore, more and more advanced algorithms are being used to squeeze out the maximum amount of information from the raw data. The "discovery machine" has been turned into a "precision machine" to allow for indirect detection of new physics through small deviations of measurements from the theoretically predicted values. One of the important measurements that particle physics graduate students learn about in their Higgs physics lectures is the "Off-shell Higgs couplings", which measures how strongly the BEH field interacts with other quantum fields when the mediator of the field (the Higgs boson particle) is highly virtual (when it has a very unusual mass).

The advantage of working at the fundamental physics level is that physicists can write down a mathematical model of the universe, "a Lagrangian", and simulate what they expect to see at the detectors if this model is correct.

Having conceptualised these experiments decades before the first actual data recording, physicists had to anticipate the advent of disruptive technology, and in the absence of it, build the technology themselves. Examples of now ubquitous technology developed or significantly advanced at CERN include the World Wide Web (for smooth communication between research institutions) and transparent capacitative touchscreen technology (for the control room of the Super Proton Synchrotron). Further disruptive innovation is required in simulation as well as physics inference to make the most of the available resources, and it is starting to come in the form of an Artificial Intelligence (AI) revolution.

The idea of autonomous machines can be found in some form even in ancient mythologies, but the computing research in Artificial Intelligence that started in the twentieth century, went through various periods of boom and bust in terms of funding, and finally saw an exponential growth in interest with the advent of data-driven Machine Learning (ML). This was made possible due to the advancement of computer technology, and availability of large amount of digitally stored data. ML algorithms quantify-ably improved the bottom line for companies, which resulting in enormous funding for such research from the private sector.

Particle physics research has had trysts with earlier versions of neural networks in the past and has used Boosted Decision Trees (BDTs) routinely, but since the recent machine learning revolution, it has started to use various new algorithms developed by the Computer Science community.

The Higgs Machine Learning challenge[3] in 2014 famously became the launching pad for a particular Boosted Decision Trees algorithm, `XGBoost` which quickly established itself as the gold standard in the ML community. Sometimes rather than the algorithms, its the hardware or differentiable programing software packages built for ML allow physicists to rework the interpretation of data from the LHC in ways that were otherwise computationally infeasible.

In physics research physicists have to be incredibly careful about biases and quantify-able uncertainties in their results. Although carefree in the early years of ML applications, risk assessment of AI models is becoming more of a concern in the wider society as well.

Recently, physicists have gone beyond using off-the-shelf ML models and are actively contributing to ML research and development. There have been efforts to develop models particularly for particle physics problems, or adapt typical statistics formalisms in the context of ML technology. As in previous cases, such innovations are likely to find use-cases far beyond High Energy Physics (HEP) research, particularly either for the removing biases, holistic modelling of a distributions, or estimating uncertainties.

1.2 Simulation Overview

The Standard Model (SM) of particle physics is being continuously tested at the LHC at the TeV scale. The scope for precision of measurements of deviations between the data and Monte-Carlo based simulations (MC) improves as more data is collected. Precise simulations of the deposition of energy in the calorimeter due to developing showers are slow because they require the modelling of interactions of particles with matter at the microscopic level, as implemented using the `Geant4` (GEometry ANd Tracking) toolkit [4]. The ATLAS detector has a complex calorimeter which proves to be the bottleneck in the simulation of events (in terms of CPU time), and the computational time scales as a function of the energy of the particles showering in the calorimeter. This would become a limiting factor in precision measurements, unless faster simulations are developed. ATLAS already relies on fast calorimeter simulation techniques based on thousands of individual parameterisations of the calorimeter response [5]. These allow significant gain in speed at the cost of accuracy.

In recent years, deep generative algorithms such as Variational Auto-Encoders (VAEs) [6, 7] and Generative Adversarial Networks (GANs) [8] have been demonstrated to accurately model the underlying distributions of data from various domains, including the response of an ATLAS-like calorimeter [9–11]. Crucially, deep learning based models have demonstrated the ability to interpolate on untrained parameter spaces, allowing to smarty curate training datasets that do not exhaustively encompass all possible input combinations.

This thesis summarises the first application of a GAN for fast simulation of the calorimeter response of the ATLAS detector for photons over a range of energies in the central region of the electromagnetic calorimeter. The integration of the model into the ATLAS simulation chain for the first time allows for a realistic validation and fair comparisons of deep generative models with other algorithms for fast simulation of the ATLAS calorimeter in terms of accuracy as well as speed and resource usage. The work has spurred further activities into this approach within the ATLAS community, and has paved the way for incorporating generative models into the ATLAS simulation framework.

1.3 Offshell H4L Overview

Although out of reach at the LHC in most decay channels, the offshell regime of the Higgs boson can be probed in the Higgs boson to four leptons decay channel, proving a unique opportunity. This is enabled by threshold effects coming from certain intermediate states (the top quarks and Z bosons) that go onshell in this regime. An offshell couplings measurement allows to break certain degeneracies such as between the Higgs couplings and the total Higgs width that cannot be disentangled by an on-shell measurement alone. Probing the total Higgs width is a very promising means of finding hints for any new particle that couples to the Higgs boson, such as an invisible particle that gains its mass through its interaction with the Higgs field.

An update to the previous ATLAS study [12] using the entire Run2 data provides an opportunity to develop innovative methodology to deal with quantum interference between the Higgs processes and other standard model processes. While the previous round used simple cuts to define the region of interest, we investigate a recently developed family of physics-aware machine learning techniques to improve the sensitivity of such an analysis, focusing on the VBF production mode. The study is performed using only the VBF process, which consists of both Higgs and non-Higgs processes. We show how quantum interference between the signal and background processes introduces non-linear effects in the yield as well as kinematic distributions because of which the analysis requires re-optimising for various values of the parameter of interest. A machine learning based inference model that is parameterised on the parameter of

interest is shown to considerably outperform estimations based on maximum likelihood fits on the distribution of a single observable or a few observables.

The study performed in this thesis provides a strong motivation to adapt the ATLAS simulation and inference framework to incorporate these new likelihood-free inference strategies that leverage the underlying physics and available machine learning technology to perform a neural network based statistical inference.

As a follow up to this work, other signal and background processes need to be included in future studies using the full ATLAS detector simulation. The work has generated interest in another group in the ATLAS community to join the effort in bringing such simulator-assisted learning models to an ATLAS analysis.

In brief, this thesis[1] is organised in the following way: a short introduction to the Standard Model of particle physics and the phenomenological overview of the offshell Higgs boson couplings measurement is presented in Chapter 2, the LHC and the ATLAS detector are introduced in Chapter 3, a review of certain concepts of Machine Learning relevant to this thesis are presented in Chapter 4, the study of fast simulation of the ATLAS electromagnetic calorimeter with a GAN is detailed in Chapter 5, optimisation studies using official ATLAS simulated datasets for the offshell analysis is presented in Chapter 6, the problems faced in this chapter lead to the investigation of physics-aware ML models for the same analysis, which is presented in Chapter 7, a new adversarial training algorithm, refereed to as the 'Aspiration Network', for mass decorrelation is presented in Chapter 8, and finally a summary of this thesis along with a discussion on the future outlook is presented in the Conclusion chapter.

[1] For the sake of clarity, all the work described in this thesis, unless explicitly stated otherwise, was performed by the author.

Chapter 2

Theoretical Overview

Contents

This chapter will provide an overview of the Standard Model of particle physics with a focus on the Higgs mechanism, followed by phenomenological aspects of it at the LHC and finally describe the motivations behind coupling measurements of the Higgs boson in the off-shell regime.

2.1 The Standard Model of Particle Physics

The Standard Model (SM) of particle physics is a mathematical model that attempts to describe three of the four known forces of the universe (electromagnetic force, weak force, strong force and excludes gravitational force). In this theory, every fundamental particle is either a *fermion* with a half integer spin, in which case it obeys Fermi-Dirac statistics, or a *boson* with integer spin, in which case it obeys Bose-Einstein statistics.

Photons (γ), Z, W^+ and W^- are gauge bosons with spin 1 that mediate the electro-weak interaction, eight other spin 1 bosons known as *gluons* mediate the strong interaction, and the scalar Higgs boson has spin 0 and is produced by the excitation of the Higgs field. Fermions form the building blocks of matter, and can be classified into *leptons* and *quarks*. They each exist in three 'generations', where each generation has greater masses than their counterpart in the previous generation (apart from neutrinos). Each generation of leptons consists of two

Name	Symbol	Charge	Spin	Mass $\left(\mathrm{GeV/c^2}\right)$	Force
Photon	γ	0	1	0	Electromagnetic
Z	Z	0	1	91.1876	Weak
$W^{\pm}$	$W^{\pm}$	± 1	1	80.399	Weak
Gluon	g	0	1	0	Strong

Table 2.1 – Properties of fundamental gauge bosons

Name	Symbol	Charge	Spin	Mass $\left(\mathrm{MeV/c^2}\right)$	Interactions
electron	e	-1	$\frac{1}{2}$	0.511	Electromagnetic, Weak*
electron neutrino	ν_e	0	$\frac{1}{2}$	$< 2.2 \times 10^{-6}$	Weak*
up quark	u	$+\frac{2}{3}$	$\frac{1}{2}$	2.3	Electromagnetic, Weak*, Strong
down quark	d	$-\frac{1}{3}$	$\frac{1}{2}$	4.8	Electromagnetic, Weak*, Strong
muon	μ	-1	$\frac{1}{2}$	105.6	Electromagnetic, Weak*
muon neutrino	ν_μ	0	$\frac{1}{2}$	< 0.19	Weak*
charm quark	c	$+\frac{2}{3}$	$\frac{1}{2}$	1.27×10^3	Electromagnetic, Weak*, Strong
strange quark	s	$-\frac{1}{3}$	$\frac{1}{2}$	95	Electromagnetic, Weak*, Strong
tau	τ	-1	$\frac{1}{2}$	1.777×10^3	Electromagnetic, Weak*
tau neutrino	ν_τ	0	$\frac{1}{2}$	< 18.2	Weak*
top quark	t	$+\frac{2}{3}$	$\frac{1}{2}$	173×10^3	Electromagnetic, Weak*, Strong
bottom quark	b	$-\frac{1}{3}$	$\frac{1}{2}$	4.18×10^3	Electromagnetic, Weak*, Strong

Table 2.2 – Properties of fundamental fermions grouped by generations.* only left handed fermions (and right handed anti-fermions) interact via the Weak Force.

leptons, the first with both electric charge and hypercharge (like the electron), and the second (known as *neutrinos*) with only a hypercharge. Each generation of quarks consists of two quarks with electric charge, hypercharge as well as colour charge. For each particle there is also an anti-particle with the opposite physical charge (unless the particle is its own anti-particle, such as photons, Z bosons, gluons and Higgs bosons)[1]. Another interesting property of a particle is its chirality (left-handed or right-handed), which describes the direction of its spin. Table 2.1 lists the properties of the bosons and Table 2.2 lists the properties of the fermions.

The different charges allow particles to interact through the different forces; the electric charge allows electromagnetic interaction, the hypercharge allows weak force interaction, and colour charge allows strong force interaction.

The strong force is described by Quantum Chromodynamics (QCD) and one of its properties is 'colour confinement', according to which any particle with colour charge (quarks and gluons) cannot exist alone and thus can never be observed in isolation. Multiple quarks can be bound together by gluons to form colour neutral *hadrons*, such as *mesons* (consisting of a quark-antiquark pair), and *baryons* (consisting of three quarks). Examples of such hadrons include protons (uud) and neutrons (udd). Apart from the three "valance" quarks, these hadrons also consist of "sea quarks" that come in and out of existence through gluon mediators but do not contribute to the overall charge of the hadron.

Predictions of the SM have been proven time and again with surprising precision, and the final piece of the SM, the Higgs boson particle, was discovered [1, 2] in 2012 by the ATLAS and CMS experiments at CERN. Despite its success, it is known that the SM is not a completely theory, and various particle physics experiments are looking for hints of new physics in the form

[1] The anti-neutrino has the opposite lepton number as its corresponding neutrino in the SM, but other experiments are studying whether it is a 'Majorana particle', in which case the neutrino would be its own anti-particle

of measurements that deviate from the SM prediction.

2.1.1 Notation

In the rest of this section,

- The reduced Plank constant $\hbar$ and speed of light in vacuum c are set to 1 unless otherwise specified.
- Greek indices span over spacetime coordinates {0,1,2,3}
- Latin indices span space coordinates {1,2,3}

2.1.2 Gauge Theories

Mathematical models in Quantum Field Theory (QFT) are described by a Lagrangian. A transformation that leaves the Lagrangian (and therefore the equations of motion) unchanged under its action represents a symmetry. If this symmetry is a function of the space-time coordinates, then it is a local symmetry, otherwise it is a global symmetry.

In a gauge theory, the Lagrangian remains invariant under local transformations from certain Lie groups. Consider a free Dirac Lagrangian for a fermion field $\psi(x)$ with mass m,

$$\mathcal{L}_{\text{Dirac}}^{\text{free}} = \bar{\psi}(i\not{\partial} - m)\psi, \tag{2.1}$$

where $\bar{\psi} = \psi^\dagger\gamma^0$, $\psi^\dagger$ the conjugate transpose of ψ, and

$$\not{\partial} = \gamma^\mu\partial_\mu. \tag{2.2}$$

The Pauli σ^k and Dirac matrices γ^μ are,

$$\sigma^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma^2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma^3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.3}$$

$$\gamma^0 = \begin{pmatrix} \mathbb{1} & \mathbb{0} \\ \mathbb{0} & -\mathbb{1} \end{pmatrix}, \quad \gamma^k = \begin{pmatrix} \mathbb{0} & \sigma^k \\ -\sigma^k & \mathbb{0} \end{pmatrix}, \tag{2.4}$$

A fifth matrix that becomes useful later is defined as

$$\gamma_5 = i\gamma_0\gamma_1\gamma_2\gamma_3 = \begin{pmatrix} \mathbb{0} & \mathbb{1} \\ \mathbb{1} & \mathbb{0} \end{pmatrix} \tag{2.5}$$

For a local transformation parameter α under the $U(1)$ group,

$$\psi(x) \to U(x)\psi(x) = e^{i\alpha(x)}\psi(x) \implies \bar{\psi}(x) \to e^{-i\alpha(x)}\bar{\psi}(x) \tag{2.6}$$

$$\begin{aligned}
\mathcal{L}_{\text{Dirac}}^{\text{free}} &\to e^{-i\alpha(x)}\bar{\psi}i\not{\partial}(e^{i\alpha(x)}\psi) - e^{-i\alpha(x)}\bar{\psi}\,m\,e^{i\alpha(x)}\psi \\
&= e^{-i\alpha(x)}\bar{\psi}\,i\left(i\not{\partial}(\alpha(x))\,e^{i\alpha(x)}\psi + e^{i\alpha(x)}\not{\partial}\psi\right) - \bar{\psi}\,m\,\psi \\
&= e^{-i\alpha(x)}\bar{\psi}\,ie^{i\alpha(x)}\not{\partial}\psi - e^{-i\alpha(x)}\bar{\psi}\,\not{\partial}(\alpha(x))\,e^{i\alpha(x)}\psi - \bar{\psi}\,m\,\psi \\
&= \bar{\psi}i\not{\partial}\psi - \bar{\psi}\not{\partial}(\alpha(x))\psi - \bar{\psi}\,m\,\psi \\
&= \mathcal{L}_{\text{Dirac}}^{\text{free}} - \bar{\psi}\not{\partial}(\alpha(x))\psi \qquad (2.7) \\
&\neq \mathcal{L}_{\text{Dirac}}^{\text{free}}, \qquad (2.8)
\end{aligned}$$

To make the Lagrangian gauge invariant, we introduce interactions. For mathematical convenience, it is done by defining a covariant derivative D_μ as,

$$D_\mu = \partial_\mu + iqA_\mu(x), \tag{2.9}$$

q is an arbitrary constant (for now) and A_μ transforms as,

$$A_\mu(x) \to A_\mu(x) - \frac{1}{q}\partial_\mu\alpha(x). \tag{2.10}$$

Now if $\mathcal{L}_{\text{Dirac}}^{\text{invariant}} = \bar{\psi}(i\not{D} - m)\psi$,

$$\begin{aligned}\mathcal{L}_{\text{Dirac}}^{\text{invariant}} &= \bar{\psi}i\gamma^\mu(\partial_\mu + iqA_\mu)\psi - \bar{\psi}m\psi \\ &= \bar{\psi}\left(i\gamma^\mu\partial_\mu - m\right)\psi - q\bar{\psi}\gamma^\mu A_\mu\psi \\ &= \mathcal{L}_{\text{Dirac}}^{\text{free}} - q\bar{\psi}\not{A}\psi.\end{aligned} \tag{2.11}$$

The second term describes the interaction between the a fermion and an anti-fermion with the field A_μ. To complete the Lagrangian we need to add a free field term for A_μ, and it can be shown that the only way to do this leads to a term in the Lagrangian involving,

$$F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu \tag{2.12}$$

which is a gauge invariant under $A_\mu(x) \to A_\mu(x) - \frac{1}{q}\partial_\mu\alpha(x)$,

$$\begin{aligned}F_{\mu\nu} &\to \partial_\mu\left(A_\nu - \frac{1}{q}\partial_\nu\alpha\right) - \partial_\nu\left(A_\mu - \frac{1}{q}\partial_\mu\alpha\right) \\ &= \partial_\mu A_\nu - \partial_\nu A_\mu - \frac{1}{q}\left(\partial_\mu\partial_\nu\alpha - \partial_\nu\partial_\mu\alpha\right) = F_{\mu\nu},\end{aligned} \tag{2.13}$$

since the order of the partial derivatives can be switched.

An important point to note is that adding a mass term to the gauge field $\frac{1}{2}m^2A_\mu A^\mu$ irrecoverably breaks the symmetry,

$$\frac{1}{2}m^2A_\mu A^\mu \to \frac{1}{2}m^2\left(A_\mu - \frac{1}{q}\partial_\mu\alpha\right)\left(A^\mu - \frac{1}{q}\partial^\mu\alpha\right) = \frac{1}{2}m^2A_\mu A^\mu + \ldots, \tag{2.14}$$

With this insight, we have the Lagrangian for Quantum Electrodynamics (QED), where A_μ is the photon field and q is the electric charge, and it is conserved in all interactions.

$$\mathcal{L}_{\text{QED}} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu} + \bar{\psi}(i\not{D} - m)\psi \tag{2.15}$$

$$. \tag{2.16}$$

In the SM, the weak force is described by $SU(2)$, and QCD is described by the $SU(3)$ group.

The unification of the weak interaction and electromagnetism leads to the electroweak (EW) interaction, described by the group $SU(2)_I \times U(1)_Y$, where Y is the weak hyper-charge and I the weak isospin. This theory has four gauge fields. Three come from $SU(2)_I$, and they are W_μ^a, while one comes from $U(1)_Y$, and it is B_μ. Further, the EW theory is a chiral theory (it lacks a mirror symmetry), the left and right handed spinors behave differently, and they can be written respectively as,

$$\psi_L = \frac{\mathbb{1} - \gamma_5}{2}\psi \quad \text{and} \quad \psi_R = \frac{\mathbb{1} + \gamma_5}{2}\psi, \tag{2.17}$$

with $\psi = \psi_L + \psi_R$. The gauge transformation of $SU(2)_I$ transforms only left handed doublets, the right handed singlets therefore do not interact through the weak interaction. The first generation left handed doublets are,

$$\psi_L \in \left\{ \begin{pmatrix} \nu_e \\ e^- \end{pmatrix}_L, \begin{pmatrix} u_\alpha \\ d_\alpha \end{pmatrix}_L \right\}, \tag{2.18}$$

and the first generation right handed singlets are,

$$\psi_R \in \left\{ e_R^-, u_{R,\alpha} \right\}, \tag{2.19}$$

where α runs over colour indices of the quark. Right handed neutrinos are not part of the SM, but can easily be included. The Lagrangian for the EW theory (for now without any Higgs related scalar field) reads,

$$\mathcal{L}_{\text{EW}} = -\frac{1}{4} W^a_{\mu\nu} W^{\mu\nu,a} - \frac{1}{4} B_{\mu\nu} B^{\mu\nu} + \sum_{\text{L fermions}} i\bar{\psi}_L \not{D} \psi_L + \sum_{\text{R fermions}} i\bar{\psi}_R \not{D} \psi_R, \tag{2.20}$$

with a covariant derivative defined as,

$$D_\mu = \partial_\mu - ig\frac{\sigma_a}{2} W^a_\mu - ig'\frac{Y}{2} B_\mu, \tag{2.21}$$

where g, g' are referred to as *coupling constants*. We have,

$$B_{\mu\nu} = \partial_\mu B_\nu - \partial_\nu B_\mu \tag{2.22}$$

now let ϵ^{abc} represent the fully antisymmetric Levi-Civita tensor,

$$W^a_{\mu\nu} = \partial_\mu W^a_\nu - \partial_\nu W^a_\mu - g\epsilon^{abc} W^b_\mu W^c_\nu. \tag{2.23}$$

If we wanted to consider inserting a mass term,

$$m\bar{\psi}\psi = m(\bar{\psi}_L + \bar{\psi}_R)(\psi_L + \psi_R) = m\left(\bar{\psi}_L\psi_L + \bar{\psi}_R\psi_R + \bar{\psi}_L\psi_R + \bar{\psi}_R\psi_L\right), \tag{2.24}$$

we see that,

$$\bar{\psi}_R\psi_R = \psi^\dagger \frac{1+\gamma^5}{2}\gamma^0 \frac{1+\gamma^5}{2}\psi = \psi^\dagger\gamma^0 \frac{1-\gamma^5}{2}\frac{1+\gamma^5}{2}\psi = 0, \tag{2.25}$$

$$\bar{\psi}_L\psi_L = \psi^\dagger \frac{1-\gamma^5}{2}\gamma^0 \frac{1-\gamma^5}{2}\psi = \psi^\dagger\gamma^0 \frac{1+\gamma^5}{2}\frac{1-\gamma^5}{2}\psi = 0,$$

leaving the terms $\bar{\psi}_L\psi_R + \bar{\psi}_R\psi_L$. These terms, and by implication the mass term (Equation 2.24) we considered cannot exist in an $SU(2)_I$ conserving Lagrangian. In experiments however, masses for EW gauge bosons (apart from the photon) and all fermions have been measured to be non-zero.

2.1.3 The BEH Mechanism

Spontaneous Symmetry Breaking (SSB) is an important concept in particle physics, as well as in statistical mechanics. A classical analogy is shown in Figure 2.1, where a nail, or a bamboo stick could have a perfect cylindrical symmetry, but if enough force is applied from the top, it eventually bends in one direction, breaking the symmetry. In Ginzburg–Landau theories, transitions of potential energy functions like the one shown often occur in second-order phase transitions.

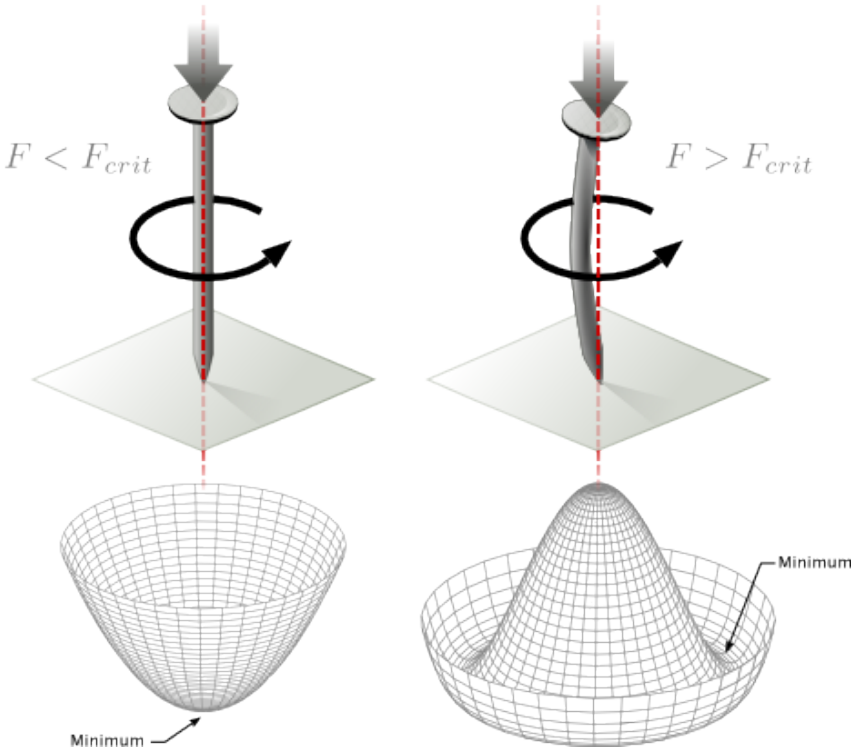


Figure 2.1 – Examples of Spontaneous Symmetry Breaking

Several physicists have contributed, sometimes independently of each other, in the development of a solution to how mass can be given the the fermions and gauge bosons without breaking gauge symmetry. This solution is commonly refereed to as the 'BEH mechanism' [13–18] (for Robert Brout, François Englert and Peter Higgs), or sometimes for simplicity the 'Higgs mechanism'.

In the simplest working version of this theory, an $SU(2)$ doublet a complex scalar field ϕ with weak hypercharge $Y = 1$ is introduced as,

$$\phi = \begin{pmatrix} \phi^+ \\ \phi^0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} \phi_1 + i\phi_2 \\ \phi_3 + i\phi_4 \end{pmatrix} \tag{2.26}$$

with a quadratic potential,

$$V(\phi^\dagger\phi) = -\mu^2\phi^\dagger\phi + \lambda|\phi^\dagger\phi|^2, \tag{2.27}$$

where $\{\mu, \lambda\} > 0$ are real constants. The negative sign of the first term destabilises the symmetric case of $\phi = 0$, the second term ensures stable minima, which are solutions of

$$\phi^\dagger\phi = \frac{1}{2}\left(\phi_1^2 + \phi_2^2 + \phi_3^2 + \phi_4^2\right) = \frac{\mu^2}{2\lambda}, \tag{2.28}$$

which has a 4 dimensional spherical symmetry. According to the Goldstone theorem, each continuous symmetry broken results in one Nambu–Goldstone (or simply Goldstone) boson.

Figure 2.2 illustrates the Higgs potential, with the unstable equilibrium at $|\phi| = 0$ and a degenerate ground state. In order to keep the photon massless (as we know to be the case), we choose the symmetry to break such that

$$\phi_1 = \phi_2 = \phi_4 = 0. \tag{2.29}$$

This choice leads to

$$\phi_3^2 = \frac{\mu^2}{\lambda} \equiv v^2 \tag{2.30}$$

which is referred to as the *vacuum expectation value* (VEV) of ϕ.

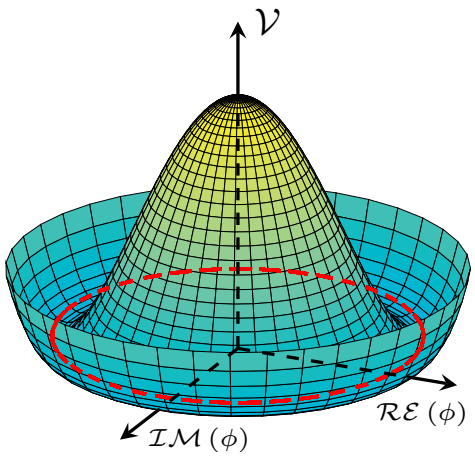


Figure 2.2 – Illustration of the Higgs potential in the SM. A highly unstable equilibrium exists at $|\phi| = 0$. The spontaneous symmetry breaking arise from the vacuum value at the minimum of the potential, which occurs for $|\phi| \neq 0$ along the red line.

With this choice, $\phi = \binom{0}{v}$, but it will fluctuate along the minimum, and we denote this as $h(x)$, a real valued function. Therefore,

$$\phi = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ v + h(x) \end{pmatrix}, \tag{2.31}$$

The EW Lagrangian with this complex scalar field doublet reads

$$\mathcal{L}_{\text{EW}} = -\frac{1}{4} W^a_{\mu\nu} W^{\mu\nu,a} - \frac{1}{4} B_{\mu\nu} B^{\mu\nu} + (D_\mu \phi)^\dagger (D^\mu \phi) - V(\phi^\dagger \phi), \tag{2.32}$$

To see the interaction of h with the other fields, we insert Equation 2.31 into Equation 2.32. D_μ is still defined as in Equation 2.21. The first terms that describe the propagation of ϕ with mass $m_H = \sqrt{2\lambda v^2}$ look like

$$\mathcal{L}_{\text{EW}} \supset \frac{1}{2} \partial_\mu h \partial^\mu h - \frac{v^2}{2} \lambda h^2 - \lambda v h^3 - \frac{\lambda}{4} h^4 + \dots. \tag{2.33}$$

However, the terms of most interest are,

$$\left| \frac{-i}{2} \begin{pmatrix} g W^3_\mu + g' B_\mu & g(W^1_\mu - i W^2_\mu) \\ g(W^1_\mu + i W^2_\mu) & -g W^3_\mu + g' \end{pmatrix} \begin{pmatrix} 0 \\ v + h \end{pmatrix} \right|^2. \tag{2.34}$$

We define,

$$W^\pm_\mu = \frac{W^1_\mu \mp i W^2_\mu}{\sqrt{2}}. \tag{2.35}$$

The component of Equation 2.34 corresponding to v leads to terms,

$$\frac{1}{4} v^2 g^2 W^+_\mu W^{\mu -} + \frac{v^2}{8} \begin{pmatrix} W^3_\mu & B_\mu \end{pmatrix} \begin{pmatrix} g^2 & -g g' \\ -g g' & g'^2 \end{pmatrix} \begin{pmatrix} W^{\mu,3} \\ B^\mu \end{pmatrix}. \tag{2.36}$$

Now also defining

$$\sin\theta_{\text{W}} = \frac{g'}{\sqrt{g^2 + g'^2}} \quad \text{and} \quad \cos\theta_{\text{W}} = \frac{g}{\sqrt{g^2 + g'^2}}, \tag{2.37}$$

gives us

$$\begin{pmatrix} Z_\mu \\ A_\mu \end{pmatrix} = \begin{pmatrix} \cos\theta_{\text{W}} & -\sin\theta_{\text{W}} \\ \sin\theta_{\text{W}} & \cos\theta_{\text{W}} \end{pmatrix} \begin{pmatrix} W^3_\mu \\ B_\mu \end{pmatrix}. \tag{2.38}$$

Putting Equation 2.38 back in Equation 2.32 ,

$$\mathcal{L}_{\text{EW}} \supset -\frac{1}{2}W^{+}_{\mu\nu}W^{-\mu\nu} - \frac{1}{4}Z_{\mu\nu}Z^{\mu\nu} - \frac{1}{4}A_{\mu\nu}A^{\mu\nu} + m_W^2 W^{+}_{\mu}W^{-\mu} + \frac{1}{2}m_Z^2 Z_\mu Z^\mu, \tag{2.39}$$

we finally get the required masses for the physical gauge bosons,

$$m_W^2 = \frac{1}{4}g^2v^2, \qquad m_Z^2 = \frac{1}{4}v^2(g^2+g'^2) \quad \text{and} \quad m_A = 0. \tag{2.40}$$

The Goldstone bosons provide gauge bosons with a third degree of polarisation, which is needed for massive gauge bosons.

Although not demonstrated, this additional scalar field also gives mass to all the fermions (apart from the neutrinos) once a Yukawa interactions between their Dirac fields and the scalar field are added to the theory. Other terms in the Lagrangian are responsible for coupling the Higgs to other gauge fields, and there are also terms responsible for triple and quartic self couplings of the Higgs field.

Apart from providing a possible explanation for the masses of the several fundamental particles of the SM, this theory predicts a relation,

$$\cos\theta_{\text{W}} = \frac{m_W}{m_Z}. \tag{2.41}$$

which has been verified by experiment. Experimental measurement of these self-couplings have yet to be made.

2.1.4 SM Lagrangian

The final Lagrangian for the Standard Model of Particle Physics is,

$$\begin{aligned}
\mathcal{L}_{\text{SM}} = & -\frac{1}{4}\sum_{a=1}^{8} G^a_{\mu\nu}G^{\mu\nu,a} - \frac{1}{4}\sum_{b=1}^{3} W^b_{\mu\nu}W^{\mu\nu,b} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} \\
& + (D_\mu\phi)^\dagger(D^\mu\phi) - V(\phi^\dagger\phi) \\
& + \sum_{i=1}^{3} -y^\ell_i(\bar{L}^i_L\phi\ell^i_R + h.c.) \\
& + \sum_{a=1}^{3}\sum_{i=1}^{3} -(y^u_{ij}\bar{u}^{a,i}_L u^{a,j}_R\tilde{\phi} + y^d_{ij}\bar{d}^{a,i}_L d^{a,j}_R\phi + h.c.) \\
& + \sum_{i=1}^{3} i\bar{L}^i_L \not{D} L^i_L + \bar{\ell}^i_R \not{D}\ell^i_R \\
& + \sum_{a=1}^{3}\sum_{i=1}^{3} i\bar{Q}^{a,i}_L \not{D} Q^{a,i}_L + \bar{u}^{a,i}_R \not{D} u^{a,i}_R + \bar{d}^{a,i}_R \not{D} d^{a,i}_R.
\end{aligned} \tag{2.42}$$

The first line describes the propagating and self interaction terms of the gauge fields. The second line describes the Higgs propagation, mass term, self couplings, and coupling to gauge bosons. The third and fourth lines describe the fermion interactions with the Higgs fields. The fifth and sixth lines describe the propagation of the fermions and their interaction with the gauge bosons. The couplings of the various fields to the Higgs field determines their masses, and therefore the photon and gluons are massless in the SM.

This Lagrangian does not describe the mass of neutrinos, the large matter-antimatter asymmetry seen in cosmology, does not provide an explanation for why the strong force preserves charge-parity (CP) symmetry and does not say anything about dark matter, dark energy, or gravity.

2.1.5 Effective Field Theory Framework

An Effective Field Theory (EFT) in particle physics is an approximation of an underlying theory that ignores substructures, additional degrees of freedom at higher energies. A famous example of an EFT is Fermi's theory of Beta decay which proposed a point-like interaction between four fermions,

$$n \to p + e^- + \bar{\nu}_e \tag{2.43}$$

while ignoring the detailed electro-weak interaction later discovered which would suggest that the interaction was mediated by W boson,

$$d \to u + W^-, \qquad W^- \to e^- + \bar{\nu}_e. \tag{2.44}$$

General relativity is also expected to be an effective field theory of full quantum theory of gravity that has yet to be formulated.

The SM can be considered a lower energy EFT of a full QFT that describes particle physics and in fact, a large class of BSM models can also be parameterised as EFTs at energies below a given energy scale Λ [19]. As explained in [20], one can calculate EFT without unnecessary reference to the high energy physics due to a 'decoupling theorem'.

An EFT Lagrangian is a systematic expansion of the SM Lagrangian and takes the form,

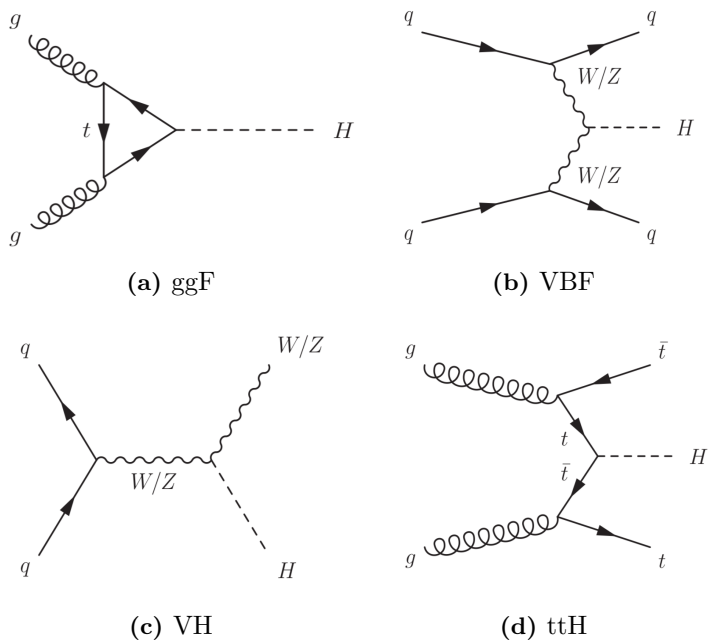
$$\mathcal{L}_{\text{EFT}} = \mathcal{L}_{\text{SM}} + \sum_i \frac{c_i^{d=5}}{\Lambda} \mathcal{O}_i^{d=5} + \sum_i \frac{c_i^{d=6}}{\Lambda^2} \mathcal{O}_i^{d=6} + \sum_i \frac{c_i^{d=7}}{\Lambda^3} \mathcal{O}_i^{d=7} + \sum_i \frac{c_i^{d=8}}{\Lambda^4} \mathcal{O}_i^{d=8} + \dots . \tag{2.45}$$

where each term $\mathcal{O}_i^{(D)}$ is an $SU(3) \times SU(2) \times U(1)$ invariant operator of dimension d and c_i are the *Wilson coefficients*, interpreted as the coupling constants for the new operators that are responsible for new effective interactions in the Lagrangian.

The idea is to define the most generic Lagrangian in each dimension, where operators of successive dimensions being suppressed by the previous one by the energy scale Λ. This way, EFT frameworks can be used in LHC physics for model-independent interpretation of experimental results. The translation of experimental data into a theoretical framework has to be done only once in the EFT context, rather than for each BSM model separately.

The reasoning is straightforward. Not all aspects of a full quantum field theory of particle physics can be tested at the LHC and therefore a lower energy approximation of the theory, its corresponding EFT would provide information on what new predictions of this theory are measurable at the LHC. The additional operators of the EFT would find their counterpart in the SM EFT expansion given above. If an experimental analysis tunes its sensitivity to operators in the SM EFT, it can therefore set limits to various theories in a model-independent way and old results may be used to set limits on any new BSM theory as well.

Often it is convenient to change the basis (set of complete non-redundant set of operators) of the EFT framework, which may help make the analysis more sensitive to certain kind of operators. The parametrisation of the space of $d = 6$ operators can be done using a subset of couplings in a mass eigenstate Lagrangian as well, which is the idea behind the *Higgs basis*, and it is often used in ATLAS because the parameters of the Higgs basis can be connected in a more intuitive way to LHC Higgs observables calculated at leading order in the EFT [19]. Certain irrelevant operators may also be removed to simplify the interpretation. Such steps have been taken in EFT measurements in ATLAS, but these details will not be summarised in this document.



(a) ggF **(b)** VBF

(c) VH **(d)** ttH

Figure 2.3 – Tree-level Feynman diagrams of the main Higgs boson production processes at the LHC.

2.2 Higgs Boson at the LHC

2.2.1 Production and Decay

At the LHC, the Higgs boson is produced via four leading production modes:

- Gluon-Gluon fusion (ggF) process, via a quark loop (dominated by the top quark). It accounts for 88% of the total production.
- Vector Boson Fusion (VBF) process, which leaves two forward jets coming from the two quarks in the final state in addition to the Higgs decay products. It accounts for 7% of the total production.
- Associated Production with Vector Bosons (VH). It accounts for 4% of the total production.
- Associated production with a top pair ($t\bar{t}H$). It accounts for 1% of the total production.

The leading Feynman diagrams for these processes are shown in Figure 2.3 and their cross sections as a function of the LHC centre of mass energy $\sqrt{s}$ is shown in Figure 2.4.

The Higgs boson is not a stable particle and has a lifetime of $\sim 10^{-22}$ s. Consequently, the detectors at the LHC cannot record its interaction to the detector directly. The Higgs rest mass and momentum are converted into the rest mass and momentum of the decay products. These particles may further decay before they can be detected, always conserving the four-momentum. The interaction of the final decay products with the detector are recorded. Since the couplings of the Higgs boson to fermions is a function of the mass of these particles, it prefers to decay to the heavier particles. The *Branching Ratio* (BR) is the fraction of the time a particle decays to one particular decay mode. The evolution of the BR for Higgs boson as a function of its mass is shown in in Figure 2.5. It is interesting to note that although the BR to two b quarks, and two W bosons is greater than to Z bosons, the latter has much cleaner final states. The leading decay mode $H \to b\bar{b}$ was observed by ATLAS only as recently as 2018 [24].

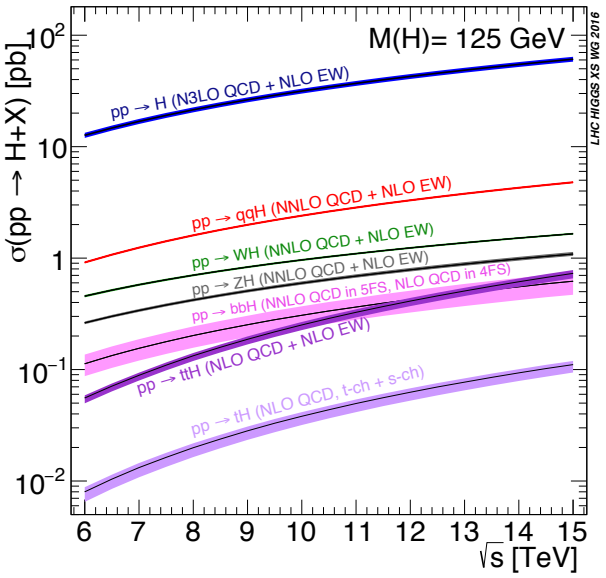
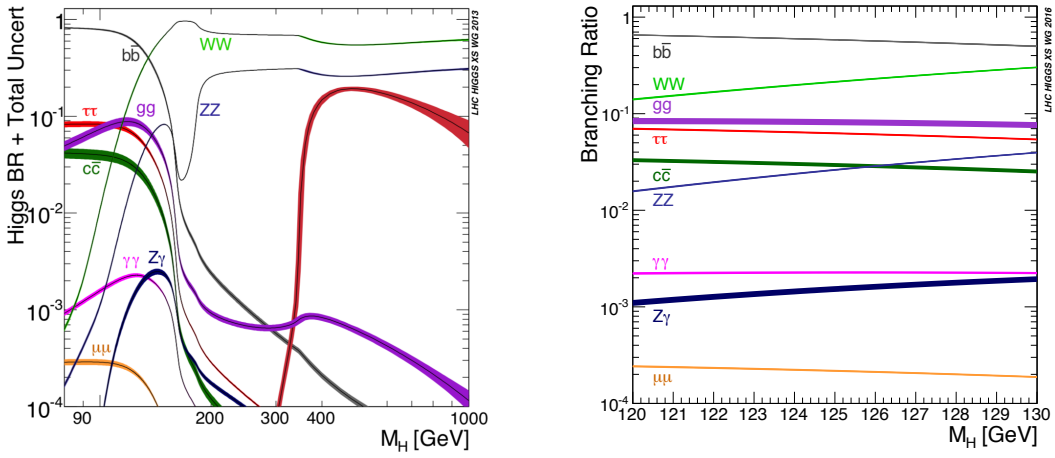
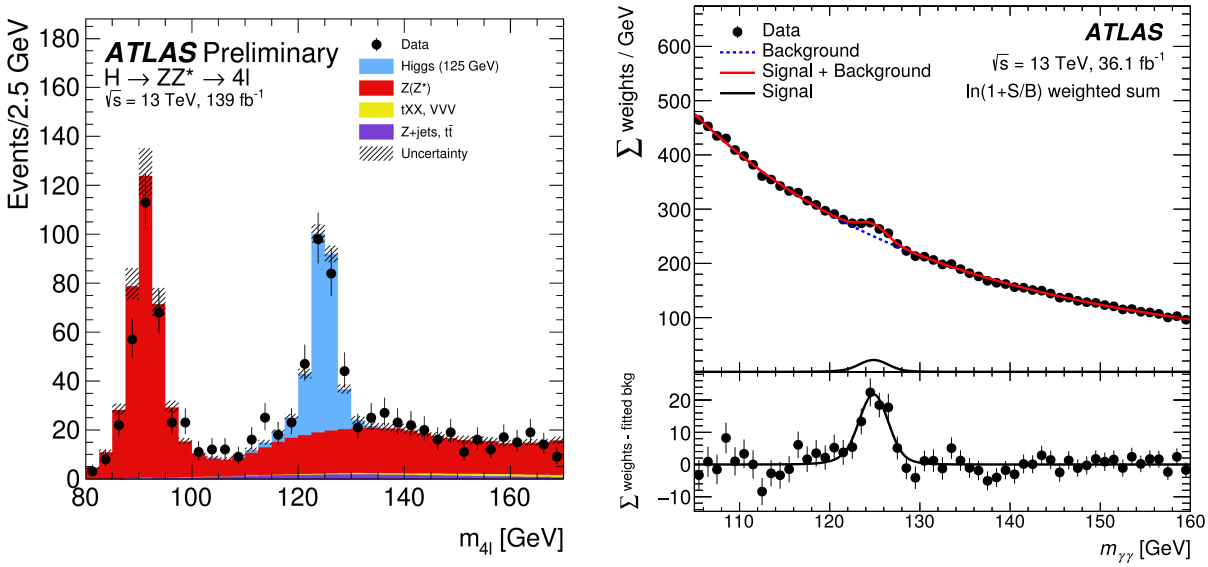


Figure 2.4 – The SM Higgs boson production cross sections as a function of the LHC centre of mass energy. [21]



(a) Large Range **(b)** Zoomed near 125 GeV

Figure 2.5 – Evolution of the SM Higgs branching ratio as a function of the mass of the Higgs. [21] The SM Higgs boson mass is around 125 GeV [22, 23].



(a) **(b)**

Figure 2.6 – The invariant mass distribution observed in the (a) $H \to ZZ^* \to 4l$ [26] and (b) $H \to \gamma\gamma$ channels in ATLAS. [22]

The invariant mass distribution of the four lepton and diphoton observed by ATLAS [25] is shown in Figure 2.6, with peaks near 125 GeV corresponding to Higgs boson in each case. These were the first two channels in which the Higgs boson was discovered in 2012.

2.2.2 Width of the Higgs Boson

The cross-section of a process is a function of the energy of its intermediate state (an unstable particle), and peaks at the resonance energy. The width Γ of such a resonance peak is related to the lifetime τ of the particle by,

$$\Gamma = \frac{\hbar}{\tau}. \tag{2.46}$$

The total width of the Higgs boson, Γ_H, is the sum of its partial widths, Γ_i, of the various decay modes,

$$\Gamma_H = \sum_i \Gamma_i \tag{2.47}$$

with

$$BR_i = \Gamma_i / \Gamma_H. \tag{2.48}$$

The total width of the Higgs in the SM varies as a function of its mass, as shown in Figure 2.7. For a mass, $m_H = 125$ GeV, the expected width, Γ_H is 4.07 MeV [27], and the partial widths are fractions of this number, which is smaller than the resolution of the ATLAS detector at about 1 GeV. The width therefore cannot be directly measured, unless the value is much larger than as expected from the SM.

The total Higgs width will provide information on the global Higgs decays (including to any as yet undiscovered new particles) and is therefore a very important measurement. Indeed, several Beyond Standard Model (BSM) theories expect a larger total width of the Higgs boson.

Indirect measurements of the Higgs width have been considered, such as by measuring the lifetime of the particle (due to their relation through Equation 2.48). CMS set a lower bound of $\Gamma_H > 3.5 \cdot 10^{-9}$ MeV at 95% confidence level by this method. [28]

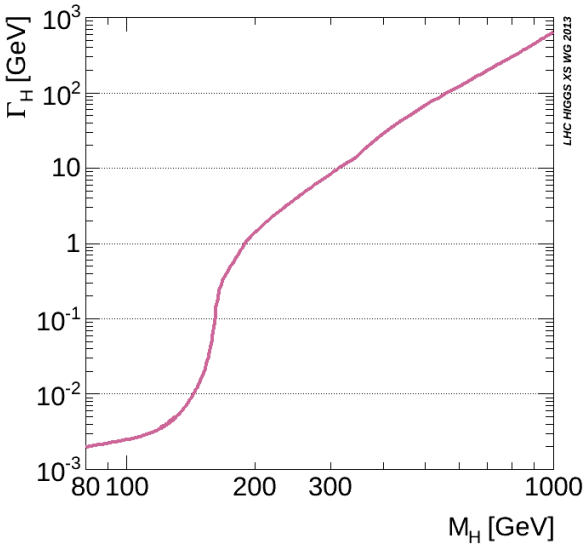


Figure 2.7 – The SM Higgs boson total width as a function of its mass. [27]

Another indirect approach involves the combination of the on-shell and off-shell Higgs boson cross-section measurements to set upper limits on the Higgs boson width. This would require certain assumptions, one of which is to assume the on-shell and off-shell Higgs boson couplings to remain the same. The context of the off-shell measurements of the Higgs boson is described in the following section.

2.3 Off-Shell Higgs Measurement in the Four Lepton Final State

This section will summarise the context of a measurement of the off-shell Higgs boson couplings at the LHC. It will start with a brief summary of the quantum mechanical aspects at the centre of the analysis, the concept of off-shell particles and quantum interference. This will be followed by a short discussion about the motivations and the connection of the off-shell couplings to the Higgs boson width, and the section will conclude with some comments about such a measurement in a dedicated VBF category.

2.3.1 Quantum Mechanical Considerations

The Heisenberg uncertainty principle of quantum mechanics ($\sigma_E \sigma_t \geq \frac{\hbar}{2}$, where E is energy, t is time) allows particles to become "virtual" for a short period of time, with a mass going far away from the one described by special relativity's mass-energy equivalence formula, $E^2 - |\vec{p}|^2 c^2 = m_0^2 c^4$ (where E is given in terms of the rest mass m_0 and momentum $\vec{p}$ of the particle). Since they are not restricted to the hyperboloid described by this formula, they are referred to as "off-shell" particles.

The probability of an interaction via a virtual particle falls off as the mass of this particle deviates from its pole mass. Given that an event with an on-shell Higgs boson is rare enough, being sensitive to the off-shell Higgs boson is usually beyond the reach of the LHC experiments at the current centre of mass energy (13 TeV). Certain situations, however, could enhance the cross section of the off-shell production and allow the study the off-shell Higgs boson. This is precisely the case in the $H^* \to ZZ$ decay channel (and will be described below).

Quantum mechanics also prescribes that when there are two paths from an initial state to a final state, they are both taken, and they may interfere with one another to produce a result

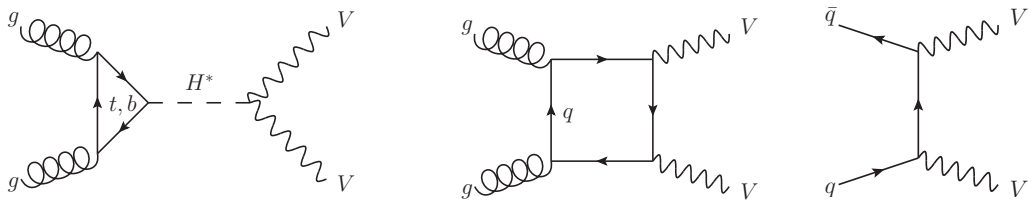


Figure 2.8 – Feynman diagrams of the main contributions to the ZZ production processes: (a) gg produced signal (Higgs-mediated), (b) gg produced background (interferes with the signal), (c) $q\bar{q}$ produced dominant background.

that is very different from a naive sum of the two. This is known as *quantum interference* and it carries through also to QFT.

Usually the signal and background processes either have different initial and/or final state particles, or come from disjoint phase spaces, and can therefore be simulated separately. As a simplified example, consider the probability of having one particular sample X, denoted $P(X)$ (with $0 \leq P(X) \leq 1$) is a function of the complex Matrix Elements, $M_s(X)$, $M_b(X)$ (with $M_s, M_b \in \mathbb{C}$), for the signal and background process respectively, is given by,

$$P(X) = |M_s(X) + M_b(X)|^2 = \underbrace{|M_s(X)|^2}_{P_s(X)} + \underbrace{|M_b(X)|^2}_{P_b(X)} + 2\underbrace{\texttt{Re}(\overline{M_s(X)}M_b(X))}_{P_i(X)}. \tag{2.49}$$

If the third term ($P_i(X)$, where 'i' stands for 'interference') is insignificant, the signal and background contributions can be simulated separately (with $P_b(X)$ and $P_s(X)$) and simply combined (because the combination is linear). However in the $gg \to (H^* \to)ZZ$ case, both the initial and final states of the signal ($gg \to H^* \to ZZ$, Figure 2.8a) and background ($gg \to ZZ$, Figure 2.8b) processes are identical, and the phase spaces overlap, therefore the contribution from the mixed term cannot be ignored. To produce physical samples, the two processes must be simulated together due to the non-linear contribution from $P_i(X)$. The interference component can have a negative contribution to $P(X)$. The individual components of the signal, background and the full process can be seen in Figure 2.9, and indeed the interference contribution is negative (explicitly shown in Figure 2.10).

A final interesting point to note is that if the couplings are scaled in such a way as to increase the signal contribution by a factor $\sqrt{\mu}$ then the corresponding matrix element needs to be scaled by $\sqrt{\mu}$ so that,

$$|M_s(X)|^2 \to |\sqrt{\mu} \cdot M_s(X)|^2, \tag{2.50}$$

then the interference component consequently is scaled by the square root of that factor (i.e $\sqrt{\mu}$) as,

$$\texttt{Re}(\overline{M_s(X)}M_b(X)) \to \texttt{Re}(\overline{\sqrt{\mu} \cdot M_s(X)}M_b(X)), \tag{2.51}$$

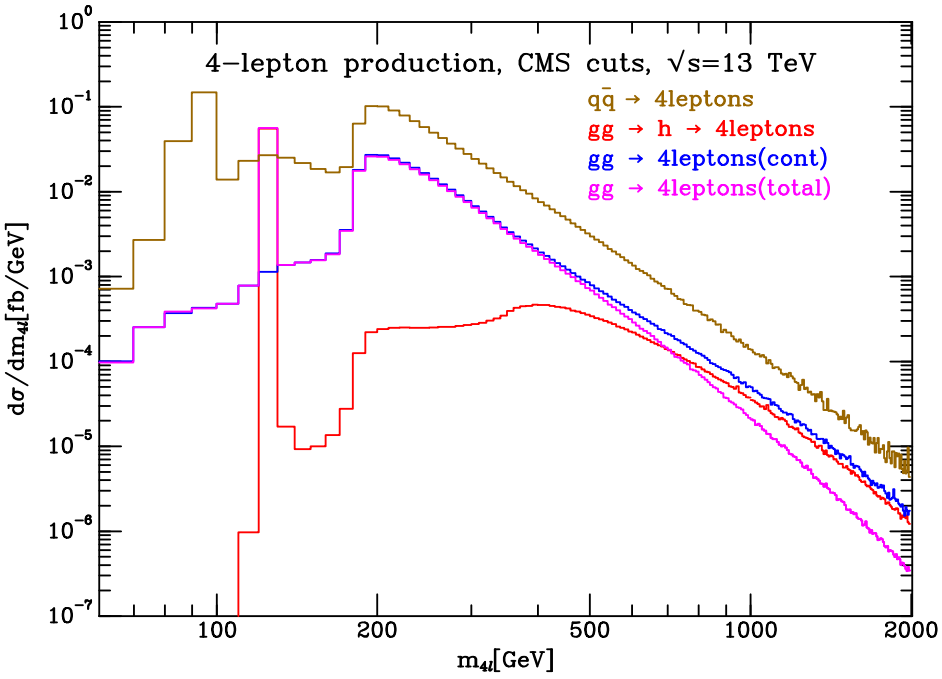
and therefore the full probability becomes

$$P_{\text{scaled}}(X) = \mu \cdot P_s(X) + P_b(X) + \sqrt{\mu} \cdot P_i(X). \tag{2.52}$$

This will play a crucial role in introducing non-linear effects in the yields in Chapter 6 and Chapter 7.

2.3.2 A unique opportunity for off-shell measurements

Since the mass of the Higgs is only around 125 GeV, the vector bosons ($2m_Z \approx 182$ GeV, $2m_W \approx 160$ GeV) and top quarks ($2m_t \approx 346$ GeV) that contribute to the on-shell Higgs



(a)

Figure 2.9 – Differential cross sections as a function of the invariant mass of the four leptons for various processes in the four lepton channel, $gg \to H^* \to ZZ$ signal (red line), $gg \to ZZ$ background (blue line), full process $gg \to (H^* \to) ZZ$ (pink line), and the dominant background $q\bar{q} \to ZZ$. [29]

processes do so through their low-mass off-shell tails (see the Feynman diagrams for the main contributors to the ZZ production in Figure 2.8). Near twice the Z mass, off-shell production of the SM Higgs boson has a substantial cross-section at the LHC [31, 32] (see Figure 2.9) because although the Higgs boson is off-shell, the intermediate Z bosons in the decay process can go on-shell. The threshold effect can be seen again near twice the top mass, corresponding to the top quarks in the production process going on-shell. This provides a unique opportunity to study the Higgs boson at higher energy scales. The destructive interference between certain SM signal and background processes further enhance the possibility to measure the presence of the signal.

The high mass off-shell study has received considerable attention because it is sensitive to various kinds of New Physics that might change the couplings of the Higgs to other fundamental particles in the high-mass region or change the ZZ background yield [33–35], and the measurement has interesting interpretations in the EFT framework [36]. Non-SM operators studied by [37] lead to enhanced yields in the off-shell regime coming from $gg \to X \to ZZ^* \to 4\ell$ where X indicates New Physics. The measurements can also help break degeneracies and compliment ttH measurements to constrain EFT parameters [38].

It is clear that at such high energies, the infinite top mass approximation often used to simplify the coupling of the Higgs to gluons breaks down, therefore it is essential to take finite top mass effects into account. New Physics could change the couplings to the top as well as introduce new heavy coloured states running in the loop and these effects might remain invisible for the on-shell Higgs [39]. The presence of any additional agent of symmetry breaking (such as a heavy neutral Higgs) is likely to affect this region of the distribution that is sensitive to interference effects. Finally, the off-shell measurement would help probe the total width of the Higgs boson, and the interest for doing so have been described in the previous section.

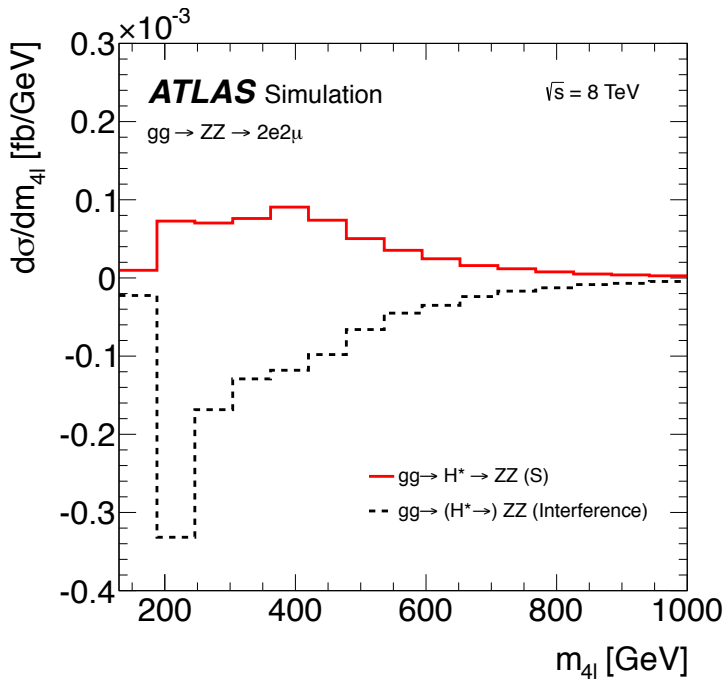


Figure 2.10 – Differential cross sections as a function of the invariant mass of the four leptons for the $gg \to H^* \to ZZ$ signal (solid red line), interference with $gg \to ZZ$ (dashed black line).[30]

2.3.3 Off-shell measurements and the Higgs Width

The width of the Higgs boson is so small that it is often approximated to zero in on-shell simulations. The authors of Ref. [31] noticed that such a zero width approximation for the off-shell Higgs simulations would provide inaccurate results. They made these conclusions based on the ggF production but it holds also for the VBF production. It begs the question of whether this relation between the Higgs boson width and off-shell effects could be inverted to measure the Higgs boson width. The authors of Ref. [32] proposed to constrain the width of the Higgs boson by combining its on-shell and off-shell coupling measurements. However, inverting relations that are not bijective can be tricky. Such a measurement of the width using ggF events would be model dependent, the reasons for which will be discussed at the end of this subsection. While keeping that in mind, we explore below the phenomenological aspects of the the Higgs width measurement strategy using the off-shell region.

The cross-section of $gg \to H \to ZZ$ as a function of the invariant mass of the Z pair is:

$$\frac{\mathrm{d}\sigma_{\mathrm{gg\to H\to ZZ}}}{\mathrm{d}m^2_{\mathrm{ZZ}}} \sim \frac{g^2_{\mathrm{ggH}} \cdot g^2_{\mathrm{HZZ}}}{\left(m^2_{\mathrm{ZZ}} - m^2_{\mathrm{H}}\right)^2 + m^2_{\mathrm{H}}\Gamma^2_{\mathrm{H}}} \tag{2.53}$$

where g_{ggH} is the effective coupling of the gluons to the Higgs, g_{HZZ} is the coupling of the Higgs to Z bosons. On-shell analyses are restricted to the mass window close to the mass of the Higgs boson, where $(m^2_{\mathrm{ZZ}} - m^2_{\mathrm{H}}) \sim m_{\mathrm{H}}\Gamma_{\mathrm{H}}$ so the total cross section after integration is,

$$\sigma^{\text{on-shell}}_{\mathrm{gg\to H\to ZZ^*}} \sim \frac{g^2_{\mathrm{ggH}} \cdot g^2_{\mathrm{HZZ}}}{m_{\mathrm{H}}\Gamma_{\mathrm{H}}} \tag{2.54}$$

which is still function of Γ_{H}. However, for the off-resonance case, in a mass window starting above twice the mass of the Z boson, where $(m_{\mathrm{ZZ}} - m_{\mathrm{H}}) \gg \Gamma_{\mathrm{H}}$, the cross section can be approximated

as:

$$\sigma^{\text{off-shell}}_{\text{gg}\to\text{H}^*\to\text{ZZ}} \sim \frac{g^2_{\text{ggH}} \cdot g^2_{\text{HZZ}}}{(2m_{\text{Z}})^2} \tag{2.55}$$

which is independent of Γ_{H} [32] but still proportional to the square of the couplings g_{ggH} and g_{HZZ}. Now if the Higgs boson width is scaled by a factor ζ and the Higgs boson couplings are universally scaled by some factor $\sqrt[4]{\zeta}$,

$$\Gamma_{\text{H}} \to \zeta \cdot \Gamma_{\text{H}}, \qquad g^2_{\text{ggH}} \cdot g^2_{\text{HZZ}} \to \zeta \cdot g^2_{\text{ggH}} \cdot g^2_{\text{HZZ}} \tag{2.56}$$

then the on-shell cross-section remains unchanged in Equation 2.54 because the ζ drops out, but the off-shell cross section scales linearly. Thus the degeneracy between a coupling and width measurement is removed. Furthermore, the off-shell cross-section measurement allows to constrain the Higgs width under the assumption that the couplings are the same for on shell and off-shell.

Now the signal strength, μ, of any process is the ratio of its cross-section to the cross-section predicted by the SM. Since m_H and m_Z are well measured quantities, they can be considered a constant. Therefore,

$$\mu^{\text{on-shell}}_{ggH} = \frac{g^2_{\text{ggH}}}{g^2_{\text{ggH,SM}}} \cdot \frac{g^2_{\text{HZZ}}}{g^2_{\text{HZZ,SM}}} \cdot \frac{\Gamma^{\text{SM}}_{\text{H}}}{\Gamma_{\text{H}}}, \tag{2.57}$$

and

$$\mu^{\text{off-shell}}_{ggH} = \frac{g^2_{\text{ggH}}}{g^2_{\text{ggH,SM}}} \cdot \frac{g^2_{\text{HZZ}}}{g^2_{\text{HZZ,SM}}}. \tag{2.58}$$

The relation of the two signal strengths to the Higgs width is given by

$$\frac{\mu_{\text{off-shell}}}{\mu_{\text{on-shell}}} = \frac{\Gamma_H}{\Gamma^{SM}_H}. \tag{2.59}$$

This relation is the crux of the strategy by which the total width of the Higgs boson can be probed by combining on-shell and off-shell measurements of the couplings of the Higgs boson in the context of the ggF production process.

Over the years, the on-shell measurements have closely and consistently confirmed the SM predictions ($\mu^{\text{on-shell}} \simeq 1$). The off-shell measurements are not yet as precise. ATLAS has set an upper-limit on $\mu^{\text{off-shell}}$ at 3.8 in 2018 [12]. It appears that to measure a width greater than the SM expectation, $\Gamma_H > \Gamma^{\text{SM}}_H$, and to keep $\mu^{\text{on-shell}} = 1$ (the SM value), an enhanced cross-section of the Higgs boson diagrams, $\sigma_H > \sigma^{\text{SM}}_H$, would be required compared to the SM prediction.

However, this is a model-dependent assessment. The authors of [40] show that New Physics contributions can decorrelate on-shell and off-shell regions for ggF in such a way that although the Higgs boson width is greater than the SM expectation, $\Gamma_H > \Gamma^{\text{SM}}_H$, the cross-section in the off-shell regime remains the same.

2.3.4 The VBF Case

In the case of the VBF production mode, the arguments are quite similar. The corresponding equations for on-shell and off-shell cross sections for the VBF production mode are,

$$\sigma^{\text{on-shell}}_{\text{pp}\to\text{H}\to\text{ZZ}^*} \sim \frac{g^4_{\text{HZZ}}}{m_{\text{H}}\Gamma_{\text{H}}}, \tag{2.60}$$

and

$$\sigma^{\text{off-shell}}_{\text{pp}\to\text{H}^*\to\text{ZZ}} \sim \frac{g^4_{\text{HZZ}}}{(2m_{\text{Z}})^2}. \tag{2.61}$$

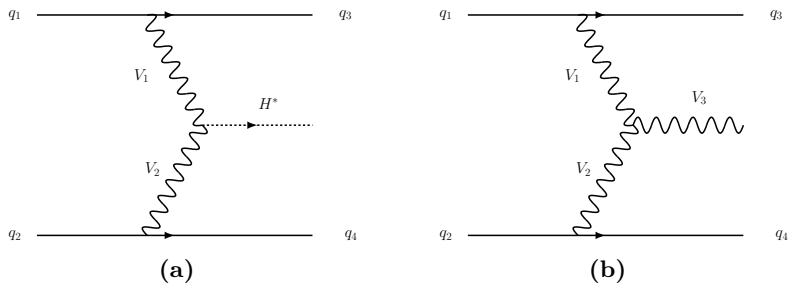


Figure 2.11 – (a) A Feynman diagrams of Vecot Boson Fusion (VBF) produced Higgs decaying to four leptons, (b) A Feynman diagram of Vector Boson Scattering (VBS)

The equations for the signal strength read,

$$\mu_{VBF}^{\text{on-shell}} = \frac{g_{\text{HZZ}}^4}{g_{\text{HZZ,SM}}^4} \cdot \frac{\Gamma_{\text{H}}^{\text{SM}}}{\Gamma_{\text{H}}}, \tag{2.62}$$

and

$$\mu_{VBF}^{\text{off-shell}} = \frac{g_{\text{HZZ}}^4}{g_{\text{HZZ,SM}}^4}. \tag{2.63}$$

The relation of VBF signal strengths to the Higgs width are therefore identical to the ggF case, and is representation by Equation 2.59.

The VBF production mode for the off-shell Higgs has different initial (qq) and final ($4l + 2j$) states as shown in Figure 2.11a, but nonetheless also faces significant quantum interference from the Vector Boson Scattering (VBS) background process shown in Figure 2.11b. A VBF analysis would further receive contamination from the $gg \to (H \to)ZZ$ process and from the $q\bar{q}$ continuum background.

Since this constraint on the width of the Higgs boson is model dependent using ggF events, the authors of [40] suggest that an off-shell measurement using the VBF production mode would allow a more model-independent interpretation. The major assumption here would be only that the Higgs boson couples similarly to the $W^{\pm}$ and Z bosons. Creating such VBF category is possible using jet information given the two-forward-jets signature of VBF. Such a measurement would allow to constrain certain kinds of BSM in a more model-independent way, although the measurement would be using far smaller statistics compared to an inclusive approach.

Chapter 6 will describe an effort to optimise an event selection strategy in such a way as to maximise the sensitivity to $\mu_{VBF}^{\text{off-shell}}$ with the use of BDTs. The premise being to make the final measurement using a maximum likelihood-fit using the learnt observable. In Chapter 7 a more sophisticated machine learning technique is studied where the maximum likelihood fit is replaced by a neural network. The advantage in the second case would be that the network can optimally account for interference effects.

Chapter 3

LHC and the ATLAS experiment

Contents

3.1 The Large Hadron Collider

The Large Hadron Collider (LHC) is the world's most powerful particle accelerator designed to study the elementary particle physics at the TeV scale with proton-proton collisions, and in addition, also study quark-gluon plasma with lead ion collisions as well as proton-lead collisions. Built at European Organisation for Nuclear Research (CERN) near Geneva, Switzerland, the circular collider is placed inside a 27 km tunnel below the surface of the earth. The first collisions were achieved in 2010 with a centre-of-mass-energy of 7 TeV, and after several upgrades, the centre-of-mass-energy was increased to 13 TeV for Run 2 (between 2015 and 2018) of the LHC. It is expected to reach its design centre-of-mass-energy energy of 14 TeV in Run3 (between 2021 and 2024). The motivation to go to higher energies is to allow for direct searches for heavier new unstable particles and also to probe the high energy tails of distributions, such as the off-shell tails of the Higgs boson, which might hide hints of new physics.

During the "High Luminosity LHC" (HL-LHC) [41] phase of the LHC, scheduled to start in 2026, the instantaneous luminosity will be increased dramatically with the goal to record up to 4000 fb^{-1} of data.

The four major particle detectors placed at the four interaction (collision) points of the LHC are:

- A Toroidal LHC Apparatus (ATLAS) [42]: A general purpose experiment with a wide range of physics objectives including the measurement of Higgs boson properties, precision measurements of the Standard Model and new physics searchers. The studies detailed in this document were performed in the context of ATLAS.
- Compact Muon Solenoid (CMS) [43]: An independent sister experiment using different technology but with similar objectives as ATLAS to ensure reproducibility of measurements made by the two experiments.
- LHCb [44]: An experiment dedicated to the study of physics related to the b quark, for example to investigate the asymmetry between matter and antimatter in the universe through CP violation.
- A Large Ion Collider Experiment (ALICE) [45]: A heavy ion experiment dedicated to study the physics of strongly interacting matter at extreme energy densities from lead ion collisions as well as proton-lead collisions.

3.1.1 The Accelerator Complex

Protons are accelerated through various stages and the LHC is the last step of the accelerator chain as shown in Figure 3.1. Once hydrogen gas is ionised to produce protons, they are accelerated to 50 MeV by Linac 2, then to 1.4 GeV by the Proton Synchrotron Booster (PSB), which allows improved injection rate into the Proton Synchrotron (PS). The PS brings the energy of the protons up to 25 GeV and is followed by the Super Proton Synchrotron (SPS), which is a 6 km accelerator brings the proton energies up to 450 GeV. After this stage the protons are finally injected into the even larger LHC in a series of bunches of protons separated in time by 25 ns. This accelerator system is also used to provide particles for experiments besides the LHC such as NA62, a fixed target experiment searching for rare decays of the K-meson, NA61, an experiment to measure the production of hadrons in different types of collisions, research and development projects such as AWAKE for high density plasma studies as well as to the test beam area.

The whole process of filling the LHC takes approximately two hours, because the smaller accelerators need to be filled multiple times to fill the LHC. To increase the probability of collisions at the interaction points, each bunch contains approximately 1.15×10^{11} protons, but this also results in additional collisions at the same time as the collision of interest, and this problem is known as *pile-up*.

The LHC uses a magnetic field to keep the particles in circular orbit, and an electric field to increase the speed. There are two beam-pipes for protons going in two opposite directions. Two-in-one superconductive dipole magnets are used to generate a magnetic field in each of the rings. The magnets are cooled to 1.9K and produce a magnetic field of 8.3 T. To increase the interaction rate, the beams are focused with quadrupoles before the collision. The electric fields are produced by the Radio Frequency (RF) Accelerating System (ACS). It takes approximately 20 minutes to accelerate the protons from 450 GeV to 6.5 TeV in the LHC.

3.1.2 Luminosity

The statistical power of any technique is ultimately limited by the size of the dataset. In the case of LHC experiments, the number of samples is a linear function of the number of events (collisions between bunches) that can can be observed at the LHC. This is directly proportional to the *luminosity*, which can be expressed as the number of events per unit of time per unit of area by the relation:

$$\mathcal{L} = \frac{1}{\sigma}\frac{dN}{dt} \tag{3.1}$$

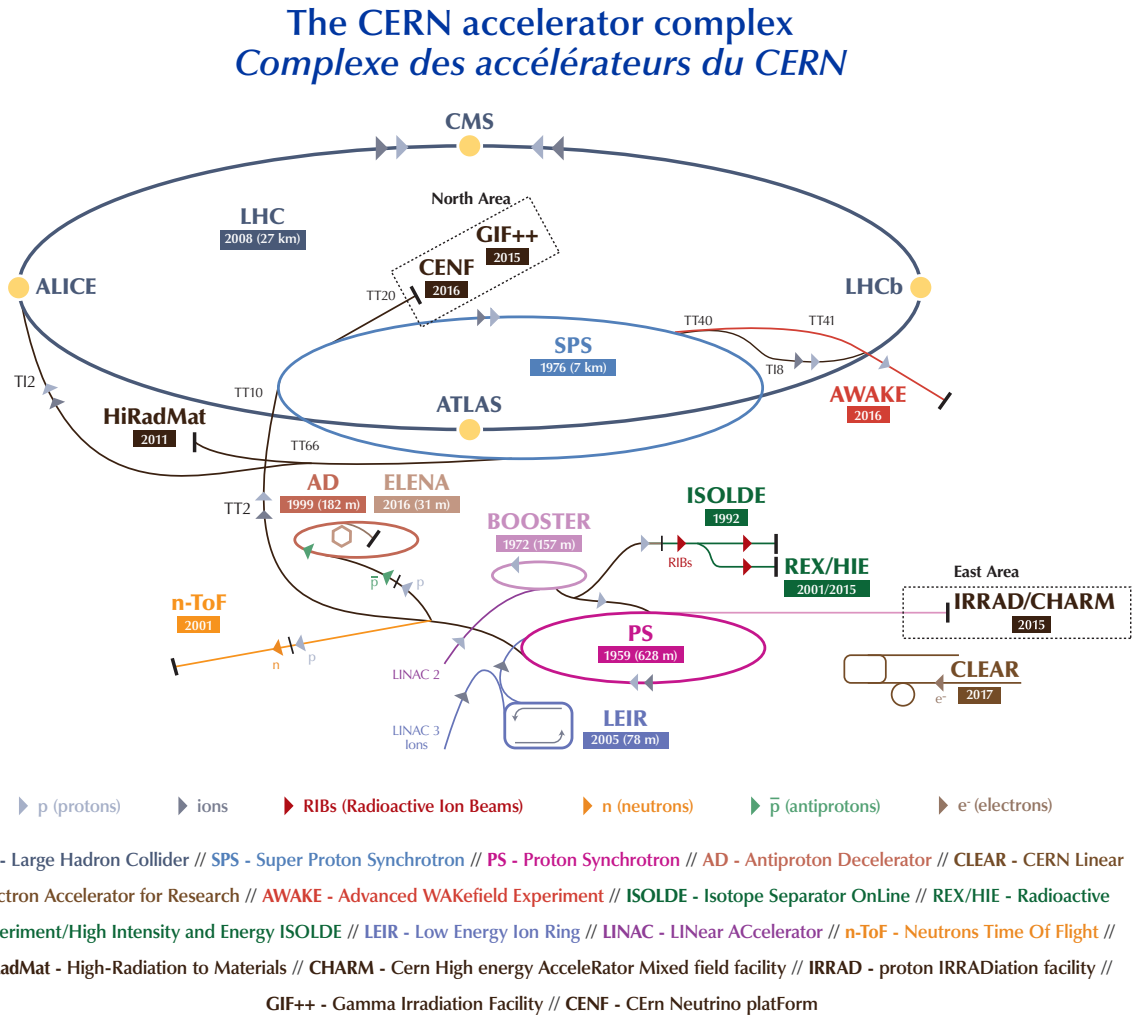


Figure 3.1 – The CERN accelerator complex and the LHC injection chain. [46]

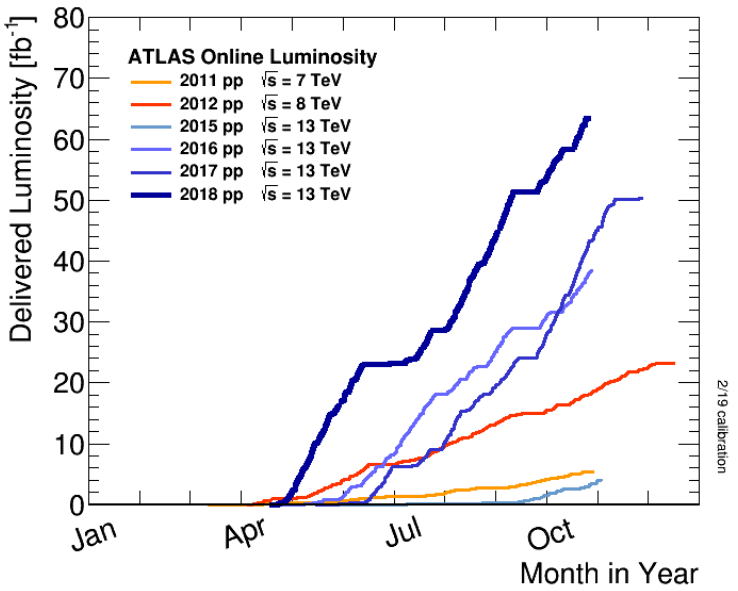


Figure 3.2 – Total integrated luminosity of pp collisions delivered to the ATLAS detector between 2011-2018 [47]

where σ is the cross section of a given process, and N is the number of times the process occurs. A great amount of effort goes into recording the maximum amount of total integrated luminosity for each run of the LHC to maximise the amount of data recorded. The luminosity depends on the quality and parameters of the beams according to the following relation:

$$\mathcal{L} = \frac{n_1 n_2 f_{rev} \mathrm{N}_b \mathrm{F}}{4\pi\sigma_x\sigma_y} \tag{3.2}$$

where N_b is the number of bunches in a beam, n_1 and n_2 are the number of protons in the two colliding bunches, f_{rev} is the revolution frequency, F is a geometrical factor to correct for the fact that the crossing angle is not exactly zero, and σ_x, σ_y are the transverse size of the beam.

The uncertainly on the measured cross section for a process is limited by the uncertainty on the luminosity, therefore, luminometers have been installed for each experiment to precisely measure their respective recorded luminosities.

The LHC has steadily improved its performance and delivered larger than expected luminosity to the experiments. The integrated luminosity delivered to the ATLAS experiment is given in Figure 3.2. The integrated luminosity used for the offshell studies reported in this document is about 36 fb^{-1} because the study was started in the middle of Run2 data taking and at the time the total integrated luminosity at the end on Run2 was unknown[1]. However, as can be seen in the LHC timeline illustration in Figure 3.3, ATLAS expects to record an order of magnitude more data in the future High Luminosity LHC phase, which would greatly improve the statistical power of most analyses if efficient algorithms are developed to handle additional pile-up. At this stage, the simulation data available may be eclipsed by the data recorded at the LHC, unless faster simulation algorithms are developed. This is important because reconstruction and analysis algorithms would likely be optimised using simulation data before being applied on the real data.

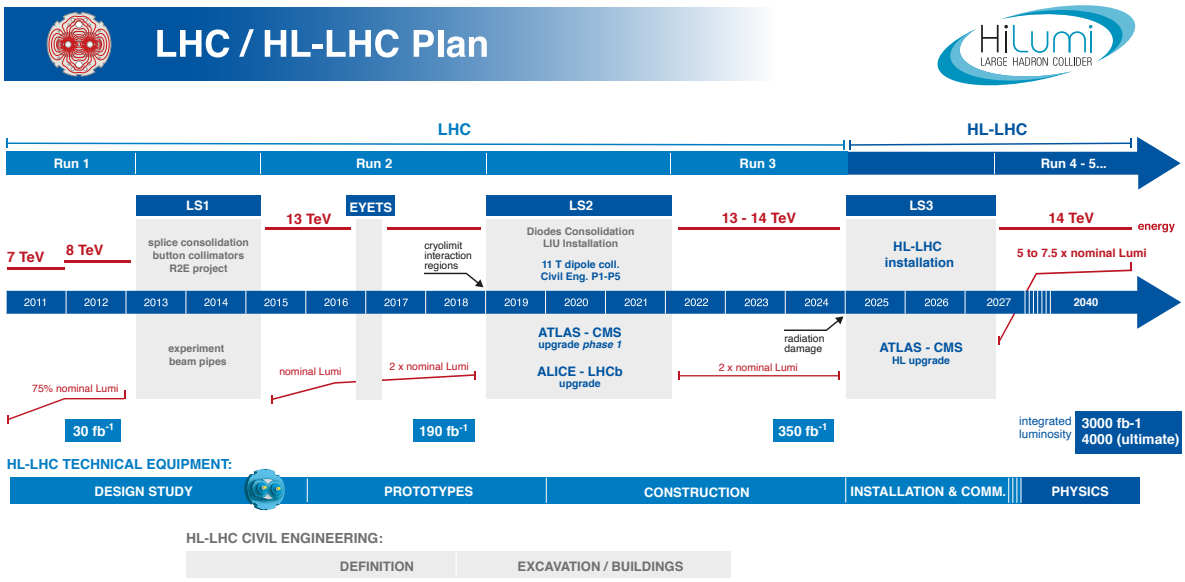


Figure 3.3 – LHC operation timeline [48]

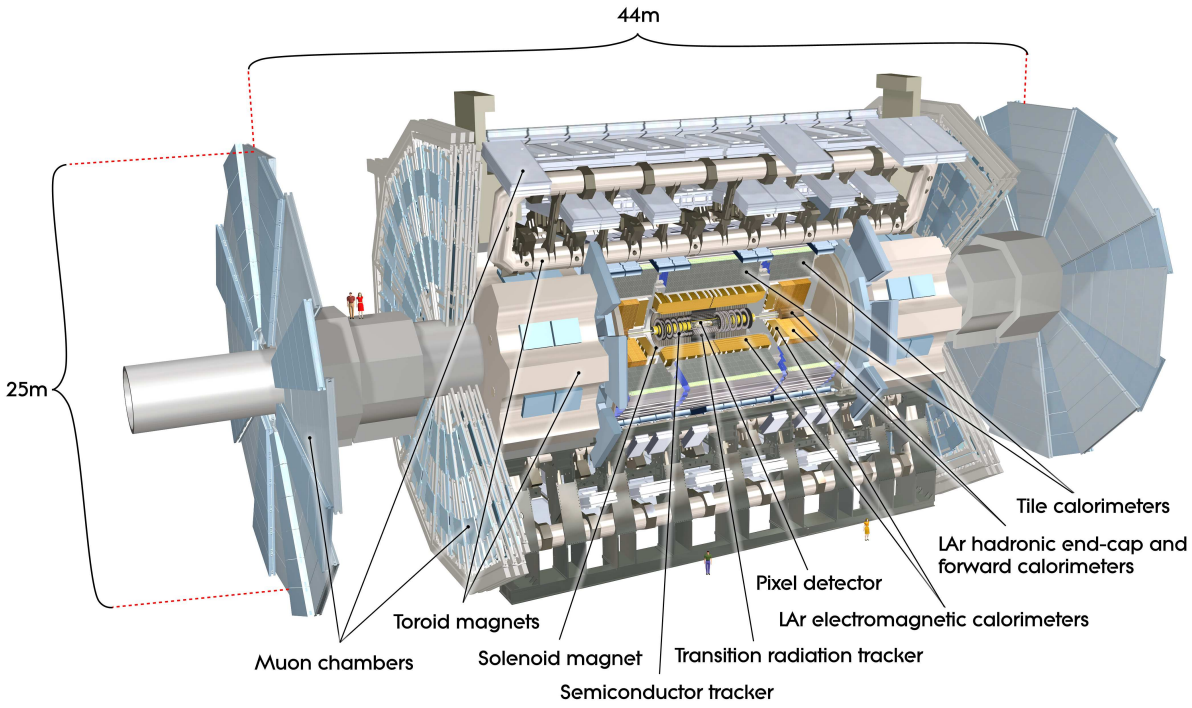


Figure 3.4 – Layout of the ATLAS detector. [49]

3.2 The ATLAS Detector

The ATLAS detector consists of a few sub-detectors as illustrated in Figure 3.4:

- The Inner Detector provides precise measurement of the trajectories and momentum of charged particles based on the curvature of the tracks in a magnetic field. It is also used to identity the primary and secondary vertices and differentiate pile-up vertices from hard-scatter vertices.
- The Electromagnetic and Hadronic Calorimeters measure the energies of electrons, photons and hadrons by absorbing their energies. They are used also to measure the Missing Transverse Energy (MET).
- The Muon Spectrometer is placed at the periphery, and provides measurements complemented by information from the Inner Detector for the identification and reconstruction of muons.

3.2.1 Coordinate System

The origin of the ATLAS coordinate system is at the nominal proton-proton interaction point, which corresponds to the centre of the detector. The x-axis points towards the centre of the LHC ring and the y-axis points upwards towards the surface of the earth. The z axis is along the beam pipe such that (x,y,z) forms a right-handed coordinate system.

A more convenient cylindrical coordinate system (θ,ϕ,z) is also defined where $\phi \in]-\pi, \pi]$ is the azimuthal angle around the beam axis, with positive values for the upper half of the detector, and $\theta \in [0, \pi]$ is the polar angle around the y-axis with $\theta = 0$ when pointing in the positive z direction.

$$\eta = -\ln\left(\tan\frac{\theta}{2}\right). \tag{3.3}$$

The pseudo-rapidity, η, defined in Equation 3.3 is more commonly used than θ, and is equivalent to rapidity, y, defined in Equation 3.4 in the ultra-relativistic limit (when a particle's mass is negligible compared to its momentum).

$$y = \frac{1}{2}\ln\left(\frac{E + p_z}{E - p_z}\right) \tag{3.4}$$

3.2.2 Inner Tracker

The tracking system is used to reconstruct the charged tracks for each collision at the LHC. A high granularity detector is needed for precise measurement, particularly in the inner layers, which becomes important because of the large number of tracks for each collision. The tracks are composed of 3D hits, the higher the number of hits, the better is the resolution of the track parameters (position and curvature, leading to vertices and momentum). The Inner Detector (ID) [50] is composed of a succession of silicon sensor layers followed by a gaseous layer and is enclosed in a 2T magnetic field generated by a solenoid. It extends up to $|\eta| = 2.5$. In a Higgs to four leptons analysis the performance of the ID impacts the muon resolution. A schematic diagram of the ID is presented in Figure 3.5.

The most central part of the ID consists of the Insertable B-Layer (IBL) which was installed in the long shutdown before Run2. It was inserted to compensate for the deterioration of the ID caused by irradiation and to improve track reconstruction. It is useful in *b-tagging*, an algorithm

[1] The optimisation performed in these studies however are expected to scale to the full Run2 data

that identifies jets originating from b quarks based on their slightly displaced vertices (which is caused by the relatively large lifetime of the b quark).

The Pixel detector surrounds the IBL and is composed of 3 layers of pixels up till $|\eta|$=2.0 and has complimentary disks to extend the coverage up to $|\eta|$=2.5. These layers consist of silicon pixel modules segmented in R and z which are highly granular, with a minimum activation size of 50μm$\times$ 400μm. This layer contributes to approximately 80 million readout channels.

After this comes the SemiConductor Tracker (SCT), which works similarly to the pixel detector but uses micro-strips instead of pixels to reduce costs. It also has additional disks in the forward region to increase coverage to $|\eta|$=2.5. The SCT contributes to approximately 6.3 million readout channels.

The final sub-detector of the tracking system is the Transition Radiation Tracker (TRT). It provides coverage up to $|\eta|$=2.0. It is a gaseous detector with a continuous active area. It has a much lower resolution but provides a high number of points along the track, which helps the reconstruction algorithm. The TRT relies on the transition radiation, that is, the emitting of a transition photon when a charged particle moves between two materials with different dielectric constants. Since the transition radiation depends on the mass of the charged particle in motion, measuring it allows to differentiate between electrons and pions from the track information.

3.2.3 Calorimeter

The ATLAS calorimeter measures the energy of particles by making them shower and absorbing the energy. The calorimeters covers the entire solid angle up to $\eta = 4.9$. A schematic view of the calorimeter is presented in Figure 3.6. It consists of three components:

- The Electromagnetic Calorimeter is a liquid argon-lead calorimeter that is used to precisely measure and identity electrons and photons.
- The Hadronic Calorimeter measures the energy of jets from hadrons and is also useful in preventing the showers from reaching the muon spectrometer.
- The Forward Calorimeter measures energies of particles in the forward region (covering 3.1 $<\eta<$ 4.9). It can measure both electromagnetic and hadronic particles.

These are sampling calorimeters (as opposed to homogeneous calorimeters as for the CMS electromagnetic calorimeter), which are built as a succession of active layers and absorber layers. The absorbers induce showering while the active materials are used to measure the signal. These calorimeters can be segmented, and therefore allow granular readout, and also contain all showers within a reasonable size, however, they have lower resolution compared to homogeneous calorimeters.

The energy resolution of a calorimeter can be written as:

$$\frac{\sigma(E)}{E} = \frac{a}{\sqrt{E}} \oplus \frac{b}{E} \oplus c \tag{3.5}$$

where $\oplus$ is a quadratic sum, $\sigma(E)$ is the energy resolution, E is the reconstructed energy, a represents the stochastic term related to the development of the shower in the absorbers, b is related to pile-up and the electronic noise from the devices used to read out the signal, and c is a constant term caused by for example inhomogeneities, or material effects. While c can be reduced with proper calibration of the detector, a and b are assumed to be the same for data and simulation. This means that any new fast simulation algorithm would have to model them correctly.

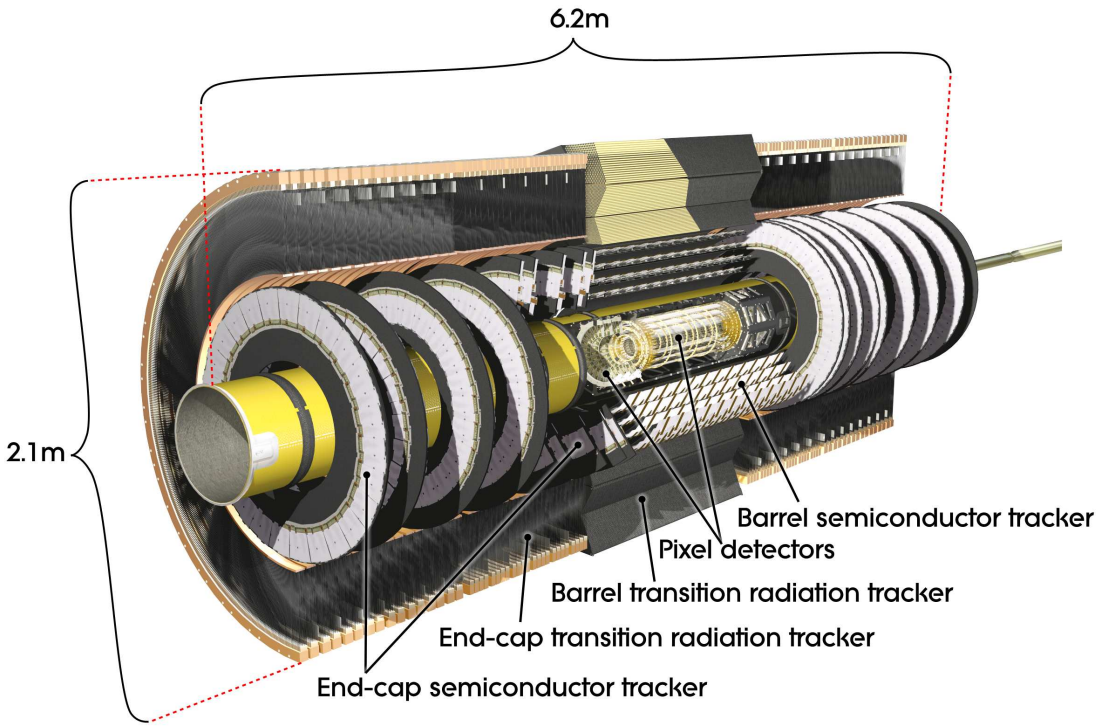


Figure 3.5 – Schematic diagram of the ATLAS Inner Detector. [49]

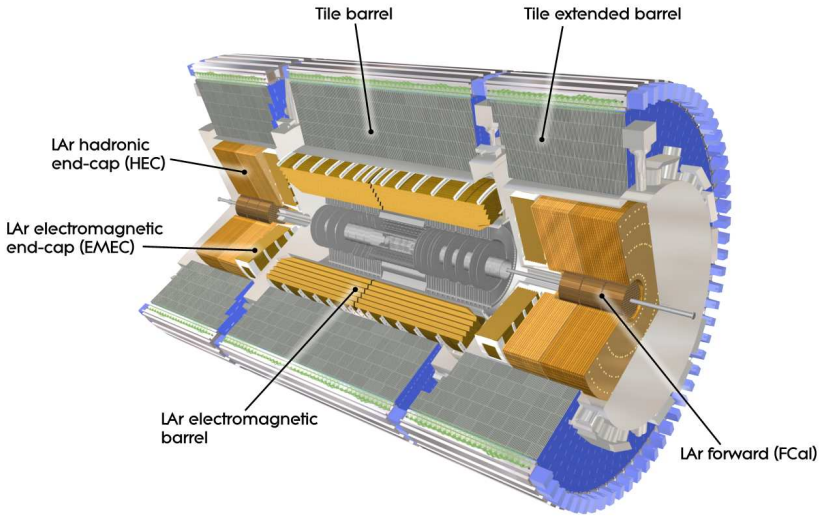


Figure 3.6 – Schematic view of the various calorimeter systems of ATLAS. The liquid argon components are shown in shades of orange, the tile calorimeter is shown in shades of green. The tracking system is shown at the centre in grey dark). [49]

3.2.3.1 Electromagnetic Calorimeter

The Electromagnetic Calorimeter (ECal) is a Liquid Argon Calorimeter (LAr) composed of two barrel components ($0 \leq |\eta| \leq 1.475$ in each direction) and two end-cap components ($1.375 \leq |\eta| \leq 3.2$).

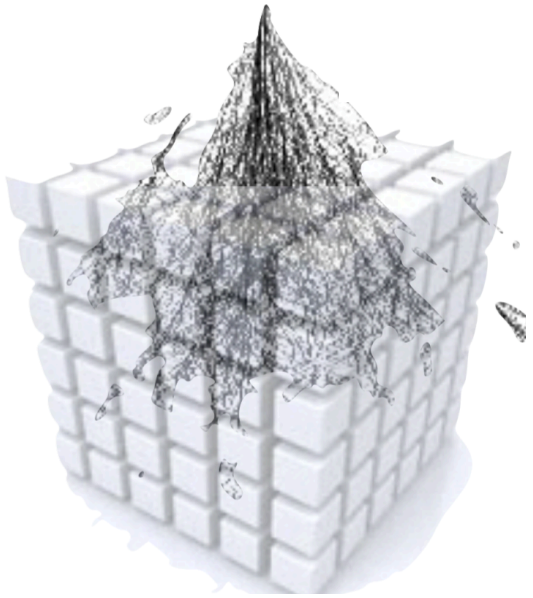


Figure 3.7 – Illustration of a particle shower in a calorimeter where the particle has already gone through material that induces the shower.

An electron that passes through the absorption material will undergo several bremsstrahlung interactions, losing a part of its energy to create a high energy photon each time. The photons will convert to pairs of high energy electrons giving rise to a cascading effect, until the subsequent particles reach the critical energy, E_c, at which the electron looses as much energy through bremsstrahlung as by ionisation. Eventually the longitudinal shower development will cease and all the energy will have been absorbed by the ECal. Figure 3.7 illustrates such a showering.

The number of particles created is proportional to E/E_c and the energy measured is proportional to the final number of low energy electrons, therefore the resolution of the calorimeter depends on the number of particles, and a lower E_c will allow a better energy resolution. The critical energy in the lead (which make up the absorbers) is $E_c = 7.4$ MeV.

The first layer, often referred to as the *Strips* or the *Front*, is finely segmented in the η direction, which allows for a precise measurement of the lateral shower shape, and it helps identification of elections and photons. Its main objective is to measure the energy of electrons and photons but since it is segmented into three layers, it allows for pointing, i.e. determining the straight line trajectory, for unconverted photons that do not leave a trance in the tracker. The next layer along the trajectory of the particle is the *Middle*, which is the thickest layer and absorbs the majority of the energy. It is less granular in η but more granular in ϕ (in most regions of the calorimeter). The final layer is known as the *Back*, and absorbs only a small amount of the residual energy. It is less granular than the Middle layer. An additional layer, referred to as the *Pre-Sampler*, is placed in front of the first layer. It is made of only liquid argon, and is used to measure the energy lost in the material in front of the calorimeter. The detailed dimensions of the cells in each of the layers is given in Table 3.1[2] and a schematic view of a section in the

[2]The reader is not expected to study the table in detail. The relevant segments will be described in Chapter 5. The table is presented to demonstrate the significant variation in cell granularity in the calorimeter, which will become relevant in the concluding discussions in Chapter 5.

barrel is shown is Figure 3.8. The varying granularity of the calorimeter will become relevant in discussions on how to extend the work presented in Chapter 5 to the full detector. While the varying granularity does pose problems to a simple generative network approach, possible solutions are discussed at the end of Chapter 5.

The resolution for the ECal is [51],

$$\frac{\sigma_E}{E} = \frac{10.7\%}{\sqrt{E}} \oplus 1\% \tag{3.6}$$

and the noise term is neglected in this equation.

Table 3.1 – Readout segmentation of the liquid argon calorimeters. The total amounts to more than 180 000 channels including the ECal, the HEC and the FCal [49].

	Barrel		**End-cap**	
EM calorimeter				
Number of layers and $\lvert\eta\rvert$ coverage				
Presampler	1	$\lvert\eta\rvert < 1.52$	1	$1.5 < \lvert\eta\rvert < 1.8$
Calorimeter	3	$\lvert\eta\rvert < 1.35$	2	$1.375 < \lvert\eta\rvert < 1.5$
	2	$1.35 < \lvert\eta\rvert < 1.475$	3	$1.5 < \lvert\eta\rvert < 2.5$
			2	$2.5 < \lvert\eta\rvert < 3.2$
Granularity $\Delta\eta \times \Delta\phi$ versus $\lvert\eta\rvert$				
Presampler	0.025×0.1	$\lvert\eta\rvert < 1.52$	0.025×0.1	$1.5 < \lvert\eta\rvert < 1.8$
Calorimeter 1st layer	$0.025/8 \times 0.1$	$\lvert\eta\rvert < 1.40$	0.050×0.1	$1.375 < \lvert\eta\rvert < 1.425$
	0.025×0.025	$1.40 < \lvert\eta\rvert < 1.475$	0.025×0.1	$1.425 < \lvert\eta\rvert < 1.5$
			$0.025/8 \times 0.1$	$1.5 < \lvert\eta\rvert < 1.8$
			$0.025/6 \times 0.1$	$1.8 < \lvert\eta\rvert < 2.0$
			$0.025/4 \times 0.1$	$2.0 < \lvert\eta\rvert < 2.4$
			0.025×0.1	$2.4 < \lvert\eta\rvert < 2.5$
			0.1×0.1	$2.5 < \lvert\eta\rvert < 3.2$
Calorimeter 2nd layer	0.025×0.025	$\lvert\eta\rvert < 1.40$	0.050×0.025	$1.375 < \lvert\eta\rvert < 1.425$
	0.075×0.025	$1.40 < \lvert\eta\rvert < 1.475$	0.025×0.025	$1.425 < \lvert\eta\rvert < 2.5$
			0.1×0.1	$2.5 < \lvert\eta\rvert < 3.2$
Calorimeter 3rd layer	0.050×0.025	$\lvert\eta\rvert < 1.35$	0.050×0.025	$1.5 < \lvert\eta\rvert < 2.5$
Number of readout channels				
Presampler	7808		1536 (both sides)	
Calorimeter	101760		62208 (both sides)	
LAr hadronic end-cap				
$\lvert\eta\rvert$ coverage			$1.5 < \lvert\eta\rvert < 3.2$	
Number of layers			4	
Granularity $\Delta\eta \times \Delta\phi$			0.1×0.1	$1.5 < \lvert\eta\rvert < 2.5$
			0.2×0.2	$2.5 < \lvert\eta\rvert < 3.2$
Readout channels			5632 (both sides)	
LAr forward calorimeter				
$\lvert\eta\rvert$ coverage			$3.1 < \lvert\eta\rvert < 4.9$	
Number of layers			3	
Granularity $\Delta x \times \Delta y$ (cm)			FCal1: 3.0×2.6	$3.15 < \lvert\eta\rvert < 4.30$
			FCal1: $\sim$ four times finer	$3.10 < \lvert\eta\rvert < 3.15$, $4.30 < \lvert\eta\rvert < 4.83$
			FCal2: 3.3×4.2	$3.24 < \lvert\eta\rvert < 4.50$
			FCal2: $\sim$ four times finer	$3.20 < \lvert\eta\rvert < 3.24$, $4.50 < \lvert\eta\rvert < 4.81$
			FCal3: 5.4×4.7	$3.32 < \lvert\eta\rvert < 4.60$
			FCal3: $\sim$ four times finer	$3.29 < \lvert\eta\rvert < 3.32$, $4.60 < \lvert\eta\rvert < 4.75$
Readout channels			3524 (both sides)	

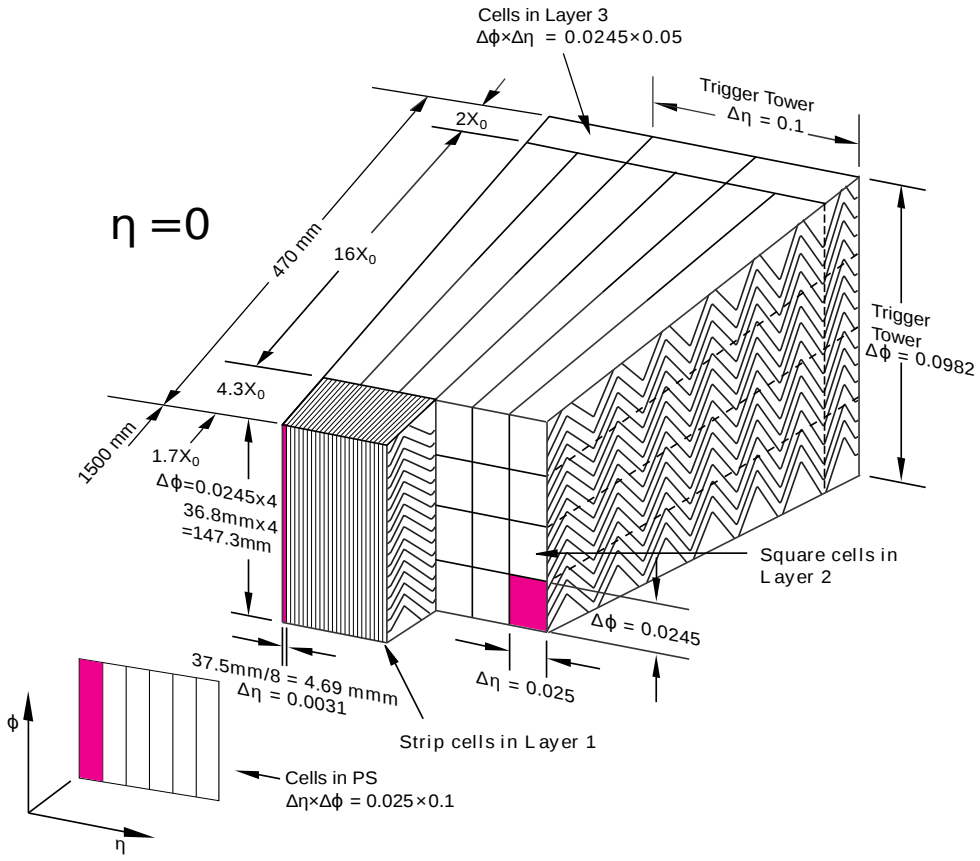


Figure 3.8 – Schematic view of a section of the electromagnetic calorimeter in the region $0 < \eta < 0.3$. The presampler in the front as well as the three layers are visible. [52]

The transition region between the barrel and the endcap ($1.4 \leq |\eta| < 1.5$) has a large amount of passive material such as readout cables and therefore the measurement is degraded. There is also a gap between the two barrels at $\eta = 0$ to pass cables. Since the radiation length for a particle originating at the centre of the detector increases with η, the thickness of the calorimeter layers is decreased once at $\eta = 0.8$. The LAr has an accordion structure (shown in the top panel of Figure 3.9) to allow for shorter readout cables while maintaining complete coverage and symmetry in ϕ. All these aspects have to be considered in simulations of the calorimeter.

3.2.3.2 Hadronic Calorimeter

The hadronic calorimeter is composed of three parts, the barrel ($|\eta|<1$), the extended barrel ($0.8<\eta<1.7$) and the end-cap ($1.7<|\eta|<3.2$). The barrel and extended barrel parts are a sampling calorimeters which use steel as an absorber and scintillating plastic tiles as an active medium. They surround the ECal and in general have a more coarse granularity compared to the ECal. In the end-cap, the expected luminosity does not allow the use of scintillating tiles, therefore radiation-hard granular liquid argon calorimeter is used with copper absorbers.

The hadronic calorimeter measures hadrons which interact via the strong force in addition to the electromagnetism, and their energy distributions are more complex. For example, hadronic shower shapes have much larger fluctuations compared to photons. This makes fast simulation of the hadronic showers a difficult task.

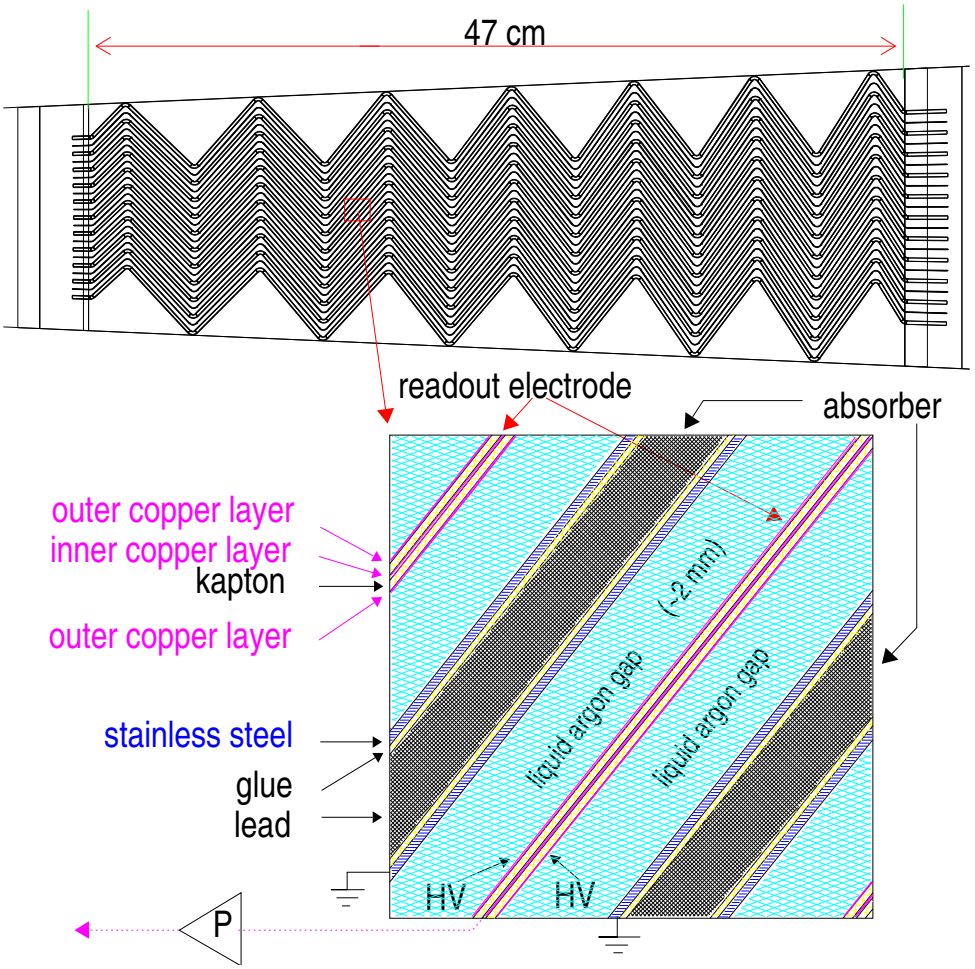


Figure 3.9 – Arrangement of the active and passive material in the LAr electromagnetic calorimeter. [53]

3.2.3.3 Forward Calorimeter

The forward calorimeter (FCal) covers the region $3.1 < |\eta| < 4.9$ and uses liquid argon as the active material. It is segmented into three disk layers, the first uses copper as the absorber which allows better reconstruction of electromagnetic showers and the other two use tungsten as absorbers which allow for better reconstruction of hadronic showers.

This detector is useful to measure forward jets and provides a better η coverage. This enables a more complete measurement of the energy for an event, thereby allowing for a more precise measurement of the missing transverse energy for that event.

3.2.4 Muon Spectrometer

Muons pass through the ID and the Calorimeters leaving a signal but without being completely absorbed. They are measured by the outer most detector known as the Muon Spectrometer (MS). The MS covers up to η=2.7 and contains four sub-detectors, two types of gas chambers to precisely measure the position, and two dedicated chambers for fast triggering:

- The Monitored Drift Tubes (MDTs) are used for precise measurement of the track position in the barrel and end-caps. They provide a good resolution in η but no information in the ϕ coordinate.
- The Cathode Strip Chambers (CSCs) are multi-wire proportional chambers installed in the in the end-caps that provide excellent spatial resolution and high counting rate capability in the forward region ($2 < \eta < 2.7$).
- The Resistive Plate Chambers (RPCs) are installed in the barrel (η<1.05) and they compromise on precision to allow fast triggering. They also provide ϕ information missing from the MDT in this region. It has no wires and is made from two resistive plates separated by a gaseous insulator, allowing an avalanche to form when a muon passes. The timing resolution is approximately 5 ns.
- The Thin Gap Chambers (TGCs) are installed in the end-caps (1.1<η<2.4) with the same purpose as the RPCs, to provide trigger capability and missing ϕ information.

A strong magnetic field is applied in the MS (1T in the barrel and 0.5T in the end-caps) to allow measurement of the momentum of the muons.

3.2.5 Trigger

ATLAS stores a tiny fraction (roughly one in 4×10^4) of the data produced. The LHC produces collisions for ATLAS as a rate of 40 MHz and given the size of an ATLAS events (roughly 1 MB), writing every event to disk would require recording at 40 TB/s. To avoid this, a two stage trigger system is used to filter out uninteresting events in real time.

The first stage is a hardware trigger known as *L1* (Level-1) made out of logic circuits. It uses only coarse calorimeter and MS information to increase the speed of the trigger and identifies Regions of Interest (RoI). This drops the event rate from 40 MHz to 100 KHz. The RoI of the events that passed the L1 are sent to the High Level Trigger (HLT).

The HLT is a software trigger that reconstructs certain objects (such as tracks, electrons, muons, jets) in the different RoIs using information from the entire ATLAS detector. It first uses fast but imprecise algorithms to perform a quick pre-selection before using more precise reconstruction algorithms to perform a second round of selection. The HLT reduces the event rate from 100

kHz to nearly 1kHz. Outlier events that take very long to process are stored in a debug stream to be processed at the end of the run.

While most of the filtered out events are uninteresting, the trigger system does not have a zero false negative rate, and sometimes throws out interesting events. The instantaneous luminosity decreases with time, so the trigger thresholds can be adjusted as the run progresses to utilise the trigger capability to its maximum bandwidth. The triggers are organised into *menus* for each stream (electron, photon, missing transverse energy, etc) with different energy thresholds for this purpose. For the low energy threshold triggers that have a bandwidth that is too high, there is an additional randomness in the selection, where 1/N events that pass the requirements are selected, for some factor N.

ATLAS has recently [54] started storing partial information already calculated by the trigger reconstruction for events that do not pass the HLT, thereby reducing the disk space required and allowing for a high event rate. There is further scope for hardware and software based machine learning approaches to improve various aspects of the trigger system in the future, such as event selection and trigger threshold control.

The rejection of large quantities of data, the option to store only partial information, limitation of computational resources which lead to the need to make decisions of whether or not to run reconstruction on an event and the choice of trigger *menus* all indicate interesting mechanisms that could be optimised and automised with novel machine learning techniques in the near future.

Chapter 4

Machine Learning

Contents

Machine Learning (ML) can be considered a sub-field of artificial intelligence that has recently been developing at an extreme pace, with growing applications in various domains including particle physics. It deals with algorithms that can optimise themselves for a particular task or set of tasks (which are usually explicitly defined) given some training environment, and most often, this environment is provided in the form of a training dataset. An ML algorithm represents

a mathematical model, and the training usually involves tuning a set of free parameters of that model. The choice of the ML model then represents an inductive bias, that can be helpful in restricting the class of mathematical functions to search through during the training phase. In this sense, these algorithms similar to a polynomial fit (such as a least squares regression), however, it would be naive to consider them as "merely an incremental step in improving fitting tools" in the context of high energy physics.

In this field, success in application has often preceded a deeper mathematical understanding of a given technique by several years, and in the case for Deep Learning (DL), a category of ML algorithms described further below, it has lead to the creation of a new sub-field that attempts to play catch up, to understand the "unreasonable" success of DL.

Advances in ML have had a two pronged effect on particle physics, the first is the availability of off-the-shelf ML models that can be used to improve data analysis, and the second is the development of hardware (like GPUs, TPUs) as well as free and open source software (like Tensorflow [55], PyTorch [56], JAX [57]) that allow physicists to either re-purpose them or build upon them for tasks that were computationally infeasible before. On the other hand, the robust statistical analysis strategies, mathematically defined objectives, and a strong theoretical grasp over mathematical modelling (such as complex symmetries in the data) that are ubiquitous in the physics community allow the physics community to contribute to ML research.

This chapter gives a brief description of some relevant tools and concepts of ML and statistics that will help the reader follow discussions in this thesis. It is by no means an exhaustive summary.

4.1 General Overview

A general overview of the (ever growing) categories of ML algorithms and the typical ML practices are presented below.

4.1.1 Categories of Algorithms

Currently, ML algorithms are broadly categorised into:

- Supervised Learning: The model is given a set of inputs and a set of correct outputs (i.e. a labelled dataset), and it must learn the best mapping. Examples include least squares polynomial regression, deep neural networks for classification, decision trees.
- Unsupervised Learning: The model is required to find structure in the dataset. Usually these algorithms are far more sensitive to inductive bias. Examples include Principle Component Analysis, Generative Adversarial Networks.
- Semi-Supervised Learning: The model is training on a small labelled dataset and a large unlabelled dataset for some task.
- Weakly-Supervised Learning: The model is trained on a dataset with very noisy labels.
- Self-Supervised Learning: These models are used when there are no human annotated labels for a given task but a labeled dataset can be created from the original dataset or the model can be trained for an auxiliary task which requires learning a useful representation.
- Reinforcement Learning (RL): An agent set inside an environment takes actions according to a policy, and the final reward is only realised after many successive steps, so the model must try to maximise the cumulative reward. The most famous example is AlphaZero [58] which can teach itself to play games like go and chess better than any human.

- Active Learning: When labeled data points are expensive to acquire, active learning models can suggest what datapoint to acquire next in order to improve the performance of the model on a given task. Like RL, these models have to balance the trade-off between exploration (of the unknown) and exploitation (of the known).

Not all algorithms neatly fall under just one or even any of these categories. Certain algorithms do not optimise any model parameters during the training, but rather memorise the training dataset, such as the k-Nearest Neighbours algorithm.

The vast majority of applications of ML involve classification of samples into two or more classes, or the regression of a target value. The training algorithm usually minimises the 'loss', a mathematical function that defines how far the output of a model is from the desired behaviour, for example a squared error for the regression of some θ_{True} and where the model predicts $\hat{\theta}$,

$$L(\theta_{\text{True}}, \hat{\theta}) = (\theta_{\text{True}} - \hat{\theta})^2.$$

4.1.2 General Machine Learning Practices

These powerful algorithms are often able to fit vastly expressive mathematical models to the training data a bit too well, which is why a robust test of over-fitting, over-training and generalisability is important. The standard practice is to randomly split the entire dataset into three parts:

- Training Dataset: The ML model is trained (or made to fit) on this dataset. Usually at least 50% of the full dataset if reserved for training.
- Validation Dataset: The performance of a trained model is evaluated on this independent dataset. The model architecture is optimised to minimise the mean of the loss function evaluated on this dataset. If the model is bring over-trained then its performance will deteriorate on the validation dataset even as it improves on the training dataset.
- Test Dataset: An independent dataset to measure the performance of the final model chosen for deployment.

Often a more computationally heavy strategy of k-fold Cross-Validation (CV) is employed to be able train on a larger fraction of the dataset, and also get a robust estimate of the performance. In this case the dataset may be split into k equal parts, trained on $k-1$ parts and evaluated on the remaining part. This can be done k times to get a mean and variance of the performance. For model selection, a nested k-fold CV may be used. The generalisability of the model may be tested depending on the use-case, for example the ability to interpolate well to points in the input space where the model was not trained.

In ATLAS, samples used for training are re-used for downstream analysis, which may involve using the output of the ML model as an observable. For this reason, a standard practice in ATLAS is to train two identical models on 50% of the dataset each, and record their outputs on the other 50%.

A list of useful machine learning based terminology is provided in subsection 4.4, which may help the reader follow the discussions in the following chapters.

4.2 Boosted Decision Trees

Decision Trees (DT) for classification recursively perform cuts on input features in order to split the dataset into 'leaves' with high purity of a single class of samples. The decision for which

feature is to be used for the split, and the value of the feature at which the split is to be made is determined based on the optimal value found from the data. This evaluation is based on a statistical criteria such as purity or combined entropy of the two new nodes.

'Boosting' comprises of ensembling a set of weak learners to make one strong learner. Two popular algorithms are AdaBoost, and Gradient Boosting. Given an objective function that is being optimised, gradient boosting tries to iteratively improve performance on the objective function by adding a new tree (trained on the full training data-set) to the ensemble while freezing everything learnt by the previous trees. If x is the training dataset and y are the class labels for this dataset,

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x), \quad \gamma_m = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i)), \tag{4.1}$$

where $F(x)$ is the classifier output for input x and the subscript m indicates the training iteration, h_m is the tree added at iteration m. L is any loss function. Often a 'learning rate' $0 < \nu \leq 1$, is used as,

$$F_m(x) = F_{m-1}(x) + \nu \cdot \gamma_m h_m(x), \quad 0 < \nu \leq 1 \tag{4.2}$$

to prevent over-training. Modern BDT algorithms such as XGBoost [59], LightGBM [60] and CatBoost [61] contain several further optimisations to improve speed, performance and reduce over-fitting, and they have several tunable hyper-parameters.

The key idea behind AdaBoost is that after each iteration, the misclassified samples in the training dataset are given a higher weight, encouraging the next tree to correctly classify them, thereby providing complimentary information to the previous trees.

The advantage of BDTs is that they are much faster to train than neural networks, work out-of-the-box, without need for much hyper-parameter optimisation, and they are well suited to structured data. However they are far less flexible compared to neural networks, in terms of architecture as well as objectives. They are not well suited for unstructured data such as images, and they also do not interpolate well. These algorithms are widely used for typical classification problems.

Chapter 6 will describe studies using BDTs to optimise the sensitivity of the off-shell Higgs to four leptons analysis.

4.3 Deep Neural Networks

Artificial Neural Networks have had a long history of waxed and waned excitement in the computer science community over the years but following a series of incremental breakthroughs in training strategy, they are today the most popular form of machine learning.

4.3.1 A simple example

A simple densely connected feed-forward deep neural network (DNN) is shown in Figure 4.1[1], where information flows from left to right. The network has a hidden layer with two nodes each (in green) and one output layer with a single node (in blue). Each node in the network performs a linear combination of its inputs followed by an 'activation function', a non-linear transformation.

If the network takes two inputs (in red), then each node in the first layer of the model has two coefficients (known as 'weights') corresponding to the two inputs, and a 'bias' term (usually the

[1] Inspired by a similar simple explanation by Stefan Wunsch

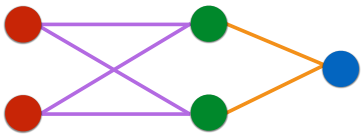


Figure 4.1 – A simple two layered neural network described mathematically by Equation 4.3

term 'weights' is used to refer to both the coefficients and the biases together). The output node similarly performs a linear combination of its inputs (which are now the output of the nodes in the previous layer) followed by an activation function. These terms can be written in matrix form as,

$$\text{Input} : x = \begin{bmatrix} x_{1,1} \\ x_{2,1} \end{bmatrix}$$

$$\text{Weight} : W_1 = \begin{bmatrix} W^1_{1,1} & W^1_{1,2} \\ W^1_{2,1} & W^1_{2,2} \end{bmatrix}, \quad W_2 = \begin{bmatrix} W^2_{1,1} \\ W^2_{2,1} \end{bmatrix}$$

$$\text{Bias} : b_1 = \begin{bmatrix} b^1_{1,1} \\ b^1_{2,1} \end{bmatrix}, \quad b_2 = \begin{bmatrix} b^2_{1,1} \end{bmatrix}$$

Activation: $\sigma(z) = \tanh(z)$ (as an example) applied elementwise.

The output of the network can then be mathematically expressed as in Equation 4.3.

$$\mathrm{f_{NN}} = \sigma\left(\mathrm{b_2} + \mathrm{W_2}\sigma\left(\mathrm{b_1} + \mathrm{W_1 x}\right)\right) \tag{4.3}$$

4.3.2 Backpropagation with AutoDiff

There are several possible algorithms to update the free parameters of a model (the weights of a neural network), such as genetic algorithms inspired by natural selection. The most successful one for deep learning is called 'backpropagation' [62]. It is the simple idea of correcting the weights of a network based on the gradient of the loss function with respect to the weights.

There are several different prescriptions for how to use these gradients to make the updates to the network weights, such as gradient descent (where the network weights are updated after the evaluation of the loss for each individual training sample) and stochastic gradient descent (where the updates are made together for a 'batch' of training samples). An entire list of 'optimizers' have been built to optimise this update process, which include a dampening factor for the update (known as the 'learning rate'), taking into account previous updates (known as 'momentum'), different learning rates for each parameter, and so on.

The entire training dataset may be used many times (each known as an 'epoch') to update the weights until the best possible performance is attained. These algorithms are only feasible if the entire forward pass as well as backward propagation of the gradients can be performed extremely fast, therefore symbolic differentiation would be too inefficient and numerical differentiation would be inexact and inefficient.

The advent of efficient automatic differentiation packages made these algorithms feasible to implement on ever growing size of neural networks and datasets. These packages build a shadow program expression-by-expression, are able to compute the gradients on-the-fly at a given fixed point. Instead of computing the gradient for a big expression as in symbolic differentiation, these packages rely on the fact that any expression is constructed from a small set core mathematical

```
                    y = x1*x2 + sin(x1)


def exp(x1,x2):                          def shadow_exp(point,diff):
    a = x1*x2                                x1,x2 = point
    b = sin(x1)                              dx1, dx2 = diff
    y = a+b                                  da = x1 * dx2 + x2 * dx1
    return y                                 db = cos(x1) * dx1
                                             dy = da + db
                                             return dy


print ("Value for (x1,x2)=(1,2): ", exp(1,2))
print ("Differentiation w.r.t. x1 at (x1,x2)=(1,2): ", shadow_exp((1,2),(1,0)))
print ("Differentiation w.r.t. x2 at (x1,x2)=(1,2): ",shadow_exp((1,2),(0,1)))
```

```
Value for (x1,x2)=(1,2):  2.8414709848078967
Differentiation w.r.t. x1 at (x1,x2)=(1,2):  2.5403023058681398
Differentiation w.r.t. x2 at (x1,x2)=(1,2):  1.0
```

Figure 4.2 – Pseudo-code to illustrate the concept of shadow programs to compute exact gradients on-the-fly. Automatic differentiation packages internally build such a map based on differentiation rules for core mathematical functions.

functions, and knowing the differential for this smaller set is sufficient to compute the gradient of the entire expression with the help of the chain rule, without the need to ever evaluate or store the full derivative of the expression. An example shadow program is shown in Figure 4.2.

4.3.3 What sets DNNs apart from other ML models

Neural Networks (NN) offer far more flexibility in terms of architecture compared to other ML algorithms. They can be trained with multiple objectives, pre-trained on alternative datasets, and different constrains can be applied to different parts of the architecture. An apparent paradox of deeper learning is that over-parameterised networks and very deep networks (networks with many hidden layers) can often leads to better performance. They also interpolate well to untrained points in the input space. The reason why over-parameterised networks do not over-fit, appears to be related to the self-regularised nature of deep neural networks, although it is still an active research question .

Neural networks could be viewed as an emergent phenomena which requires far more sophisticated tools for interpretability compared to polynomial fits. Nevertheless, introducing inductive biases in the architecture often lead to better performance, and therefore domain knowledge remains key to improving performance.

Beyond classification, neural networks can be considered function approximators, or tools for differentiable programming, which opens the door to new approaches to solving experimental physics problems [63–66].

4.4 Terminology

Listed below are certain useful machine learning terminology, many of which have been used in this thesis.

General terminology:

- Feature: Measurable property ('observable' in HEP context), which is usually given as an input to an ML model.
- Preprocessing: Transformations made to the dataset that make the training easier. This may also include handling of missing values. A typical example of preprocessing is to standard-normalise[2] the features.
- Hyper-Parameters: Parameters that define the model, beyond the free parameters that are optimised during the training. For neural networks, typical hyper-parameters include depth, width of the layers.
- One-Hot Vector Encoding: Is is a type of preprocessing in which a categorical feature is converted into a format more conducive to neural networks, using only 1,0. For example, if the feature indicates the preferred type of operating system of an organisation, [1,0,0] may represent Linux, [0,1,0] Macintosh and [0,0,1] Windows. The idea can easily be extended to multi-hot encoding when there are overlaps. In the case of jets in HEP, one may reserve a single number per jet, so [0,0,1] for 1 jet, [0,1,1] for 2 jets and [1,1,1] for 3 jets.
- Transfer Learning: A technique in machine learning where information that is learnt to solve one problem can be re-purposed to learn to solve a related problem. A typical example in image recognition is to pre-train convolutional layers on a large but unrelated image dataset before re-training the network on a smaller target dataset.
- Domain Adaptation: A subset of transfer learning where the task remains the same on the source (distribution used to train the model) and target (distribution on which the model will be applied) domains, although their distributions are not identical.
- Multi-label Classification: Classification where each sample may have multiple labels.
- Multi-Task Learning: In this scenario a model has multiple objectives to simultaneously optimise. It can get difficult to balance the trade-off between performance at the two tasks and there is often a hyper-parameter used to indicate the relative importance of each task.
- Anomaly Detection: A model is used to detect unexpected, rare samples compared to a 'normal' dataset.
- Symbolic Regression: A regression within a space of mathematical expressions. It is useful for interpretability, and in contexts where an analytical solution is required. Although still a small subfield within the deep learning context, it could offer useful solutions to physics problems in the future.
- Receiver Operating Characteristic (ROC) curve: A plot of the True Positive Rate vs the False Positive Rate of a binary classifier as the discrimination threshold is varied. A $\frac{\pi}{4}$ rad diagonal line indicates performance equivalent to random guesses.
- Area Under the ROC Curve (AUC): A metric often used to evaluate the performance of a binary classifier, with the value in the range [0,1] (the higher the better). An AUC of 0.5 indicates performance equivalent to a random chance, a classifier with a lower AUC could be inverted to attain a better than chance performance.
- Early Stopping: For an iterative training algorithm the performance may be evaluated on a validation dataset. If the validation performance starts to deteriorate due to over-training, the training may be stopped earlier than the number of training iterations planned. There may be some leniency as to how much or how consistently the performance must deteriorate over the iterations for early stopping to be triggered.

Deep Learning related terminology:

[2] the mean of each variable is moved to zero and the standard deviation to one

- Dense: In a dense layer (fully connected layer), all the nodes from the previous layer are connected to all the nodes in the next layer.
- Regularisation: Similar to its meaning in statistics, regularisation is used in deep learning to add information (such as requirement for smoothness, sparsity, an invariance), often in the form of an additional objective to simultaneously optimise. It is often used to prevent over-fitting.
- Convolutional Layer: These layers are often used in computer vision problems to add inductive bias into the architecture of the network. This layer explicitly adds space invariance or shift invariance into the architecture.
- Graph Nets: Graph based layers are an active field of development aimed at generalising the concept of convolutional layers to much more abstract graphs. They take graphs, with edges, nodes and general attributes as input and produce graphs as output. They can more naturally deal with images on curved surfaces, non-traditional pixel sizes or even particle physics inspired connections between objects.
- ONNX: Open Neural Network Exchange, is an open format built to represent machine learning models. The neural networks are represented using the underlying set of base mathematical functions, which allows even custom layers to be saved and loaded on different neural network platforms.

4.5 Generative Models

Since neural networks can be represented as a series of matrix multiplications (and non-linear, often elementwise transformations), there is no inherent restriction to the number of outputs of a network, as long as an appropriate loss function can be determined.

Generative networks can learn to reproduce entire probability distribution functions. If a discriminative network is trained to learn a conditional probability of target Y for some observed x as $P(Y|X = x)$, generative networks instead learn to reproduce the conditional probability of observable X for a given target y as $P(X|Y = y)$. The two most popular Deep Generative Models (DGM), that have been used extensively to generate photo-realistic images of humans, houses, cars and so on are:

- Generative Adversarial Network (GAN) [8]: It is an unsupervised learning algorithm consisting of two networks, the Generator (G) and the Discriminator (D), where the latter assists the former in learning a high dimensional target distribution. Since a neural network is deterministic, to induce stochastic behaviour, G is designed to act as a function that takes a vector of random noise (numbers sampled from a random Gaussian distribution, for example) as input. This way a range of different outputs can be generated from the same network G, each time with a different random input vector[3]. The training algorithm attempts to solve a minimax game where D is trained to differentiate 'fake' samples generated by G from real data (X) while G is trained to produce samples that are missclassified by D. The two networks are trained alternately, until the desired convergence has been reached. The Nash Equilibrium for this two player game is where G produces samples so similar to the target distribution X that D always outputs $\frac{1}{2}$, however, there are no mathematical guarantees for convergence of this algorithm. A schematic diagram of a GAN is shown in Figure 4.3. A variant of this model is developed for calorimeter simulation in this thesis.

[3]Two identical photons entering the calorimeter may produce very different looking showers due to the quantum randomness, the generator must be able to reproduce the full probability distribution of shower images, not just a single image of a shower

- Variational AutoEncoder (VAE) [6, 7]: It is an unsupervised learning algorithm that combines deep learning with variational inference. It is a latent variable models that introduce a set of random variables that are not directly observed but are responsible for the underlying structures in the data. The model is composed of two stacked neural networks, the Encoder (E) and the Decoder (D), where the latter can be used as a generative model after training. E compresses the input data x as $q_\theta(z|x)$ into a lower dimensional latent space $q_\theta(z)$, while D learns the inverse mapping, reconstructing the original input from this latent representation as $p_\phi(x|z)$. A crucial point to note is that this latent representation is stochastic, that is $q_\theta(z|x)$ maps x to a full distribution rather than being a function $x \mapsto z$ Once the composite model is trained, the decoder can be used independently to generate new data $\tilde{x}$, where new samples are synthesised by sampling z according to the prior probability density function $p(z)$ thus sampling $\tilde{x}$ from $p_\phi(x|z)$. The restriction of the distribution $p(z)$ is enforced during the training. Often times, an adversarial constrain, or normalising flows are used to improve the distribution of z. A schematic diagram of a VAE is shown in Figure 4.4.

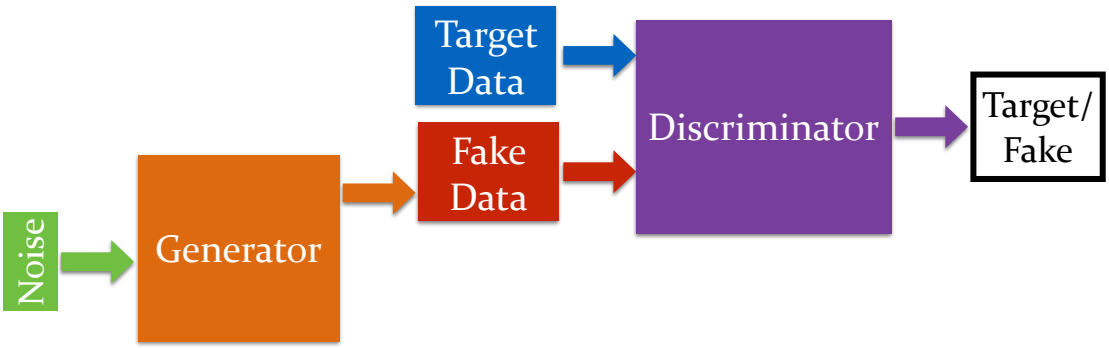


Figure 4.3 – Schematic Diagram of a GAN

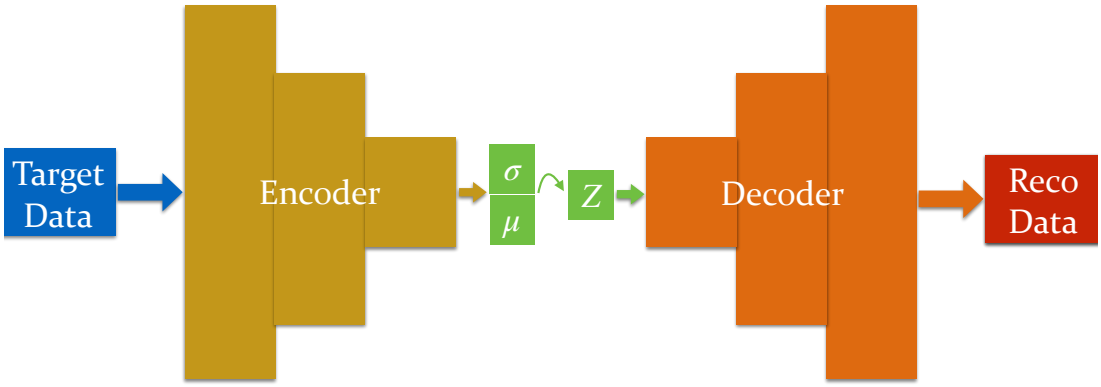


Figure 4.4 – Schematic Diagram of a VAE, where $Z = \mu + \epsilon \cdot \sigma$, ϵ is a random noise drawn from a multidimensional standard normal distribution, $\mathcal{N}(0, \mathbb{I})$.

These models do not simply reproduce the data seen in the training, they can produce new samples from the approximated underlying distribution and can interpolate continuously though the input space (as will be described in Subsection 4.5.2).

4.5.1 Wasserstein GANs with Gradient Penalty

The optimisation the original 'vanilla' GAN is performing can be mathematically represented as,

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))], \tag{4.4}$$

where G and D are the generator and discriminator networks respectively, and $p_{\boldsymbol{z}}$ represents the distribution of the latent space of random noise that the generator takes as input. This algorithm is difficult to train, and suffers from problems such as:

- Training Instability: A cocktail of heuristic tricks [67] are needed to train a GAN, in addition to very fine tuning of certain hyper-parameters such as the Training Ratio (number of training iterations of the discriminator for every iteration of the generator). There is still no guarantee of convergence.
- Modal Collapse: The generator produces only a few modes of a multi-modal target distribution at any given point in the training process, illustrated with an example in Figure 4.6.
- Vanishing Gradients: When the discriminator is fully trained it may have a loss of zero, particularly at the very beginning of the training algorithm when the support of the real and fake distributions are disjoint, illustrated in Figure4.5. The generator cannot learn because the discriminator does not provide useful gradients.

The instability of the algorithm leads to problems with reproducing even similar performances for exactly the same architecture. Randomness comes from the train-test split of the dataset, the random noise taken by the generator, the random initialisations of the network weights and the randomness coming from parallelisation of the training on multiple GPUs.

Authors of the Wasserstein GAN (WGAN) [68] proposed to replace the discriminator with a 'critic' network which estimates the Wasserstein-1 or the Earth-Mover (EM) distance[4], between the real and generated distributions. Such a loss function would not suffer from vanishing gradients, and it would also take into account all modes of the target distribution. They show that it can be implemented by enforcing the estimating function (here the critic network) to be within a k-Lipschitz space[5] (where k is some arbitrary natural number). In practice this is enforced by clipping the weights of the critic network to within $[-c, c]$, where c is a tunable hyper-parameter. These GANs are more stable to train. The authors suggest interpreting the critic loss as the Wasserstein-1 distance between the real and generator distributions. The authors of the Gradient Penalty based Wasserstein GAN (WGAN-GP) [69] further improved upon this idea by softly enforcing a 1-Lipschitz constrain on the critic. They replaced the weight clipping with a gradient penalty term in the loss of the critic. The WGAN-GP solved certain pathological problems of the WGAN such as estimating higher order moments, as demonstrated in Figure 4.7.

The loss function for the critic reads,

$$L_{\text{Critic}} = \underbrace{E_{\tilde{x}\sim p_{\text{gen}}}\left[D(\tilde{x})\right] - E_{x\sim p_{\text{real}}}\left[D(x)\right]}_{\text{Wasserstein Distance}} + \underbrace{\lambda E_{\hat{x}\sim p_{\hat{x}}}\left[\left(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1\right)^2\right]}_{\text{Gradient Penalty}}. \tag{4.6}$$

Here p_{real} is the target probability distribution, and p_{gen} is the probability distribution of outputs of the generator network (which takes random numbers drawn from some distribution as the input latent space) and $D(x)$ is the output of the critic network for a given input x. The term $E_{\tilde{x}\sim p_{\text{gen}}}[D(\tilde{x})]$ represents the critic's ability to correctly identify synthesised showers, while the

[4]

$$W(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma\in\Pi(\mathbb{P}_r,\mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}\left[\, \|x-y\| \,\right], \tag{4.5}$$

where $\Pi(\mathbb{P}_r, \mathbb{P}_g)$ denotes the set of all joint distributions $\gamma(x, y)$ whose marginals are $\mathbb{P}_r$ and $\mathbb{P}_g$ respectively. $\gamma(x, y)$ indicates how much "earth" must be transported from x to y in the 'soil distribution' $\mathbb{P}_r$ into order to transform it into the distribution $\mathbb{P}_g$, and the EM distance is minimum total earth one would have to move, a famous solution to an optimal transport problem.

[5] Lipschitz constrain intuitively is an upper limit to the norm of the gradient of the function.

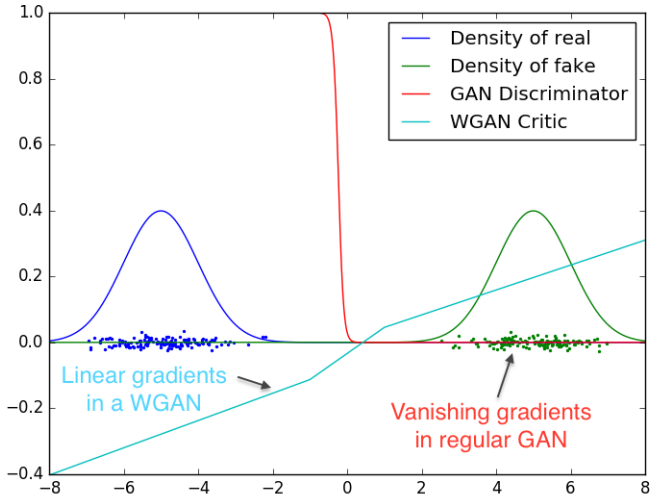


Figure 4.5 – Optimal discriminator and (weighted clipped) critic when learning to differentiate two Gaussians. The discriminator of a vanilla GAN saturates and results in vanishing gradients, whereas the WGAN critic provides clean gradients on all parts of the space. [68]

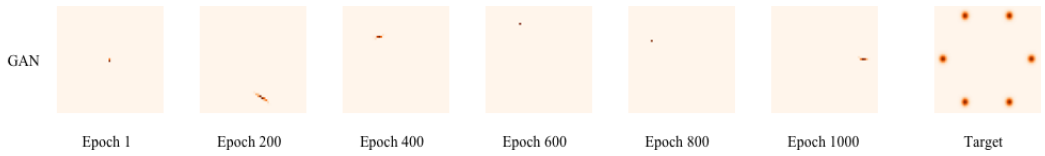


Figure 4.6 – Example of a GAN with mode collapse. At every stage the discriminator only learns to differentiate between the real images and the generator images, and the generator learns only to produces images from one or a few modes that fool the discriminator. [70]

term $E_{x\sim p_{\text{real}}}[D(x)]$ represents the critic's ability to correctly identify showers from target distribution. Together they estimate the Wasserstein-1 distance between the real distribution and the distribution from the generator. The last term in the loss function, $\lambda E_{\hat{x}\sim p_{\hat{x}}}[(||\nabla_{\hat{x}} D(\hat{x})||_2 - 1)^2]$, is the two-sided gradient penalty (which leads to the "GP" in WGAN-GP), where $\hat{x}$ is a random point along the straight line connecting a point from the real distribution p_{real} and generator distribution p_{gen}, and λ is a hyper-parameter that indicates the relative importance of the final term in the loss and is known as the Gradient Penalty Weight (GPW). This additional loss term is a way of softly enforcing the 1-Lipschitz constrain mentioned above, while also penalising very low gradients (demonstrated mathematically in [69]) Penalising low gradients is not mathematically required but is often found to help the training.

Unlike the vanilla GANs, WGANs do not require deliberate under-training of the discriminator, on the contrary, it is advised to use a high training ratio (number of times the discriminator is trained for each time the generator is trained) to ensure that the critic is fully trained after each iteration of the generator. The default recommendation is a training ratio of 10. WGAN-GPs also have certain problems:

- Slow training: WGAN-GPs take longer to train, requiring many more epochs than vanilla GANs. The WGAN-GP developed in this thesis was trained for 25000 epochs and a related project[6] has reported having trained for over two million epochs.
- Oversold Loss Interpretability: Although the authors encourage interpreting the loss as a

[6] A WGAN used in ATLAS to train on voxelised calorimeter images, a technical note is in preparation at the time of writing.

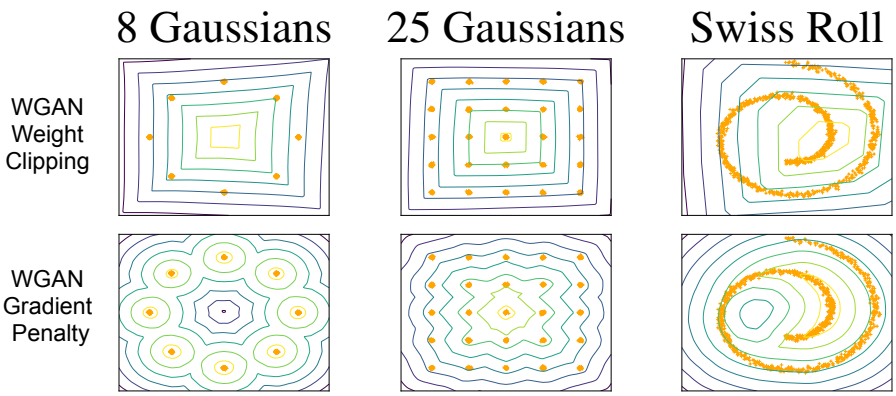


Figure 4.7 – Value surfaces of WGAN critics trained on toy datasets using weight clipping and gradient penalty, with real data in orange. The former fails to capture higher moments of the distributions. The fake data was fixed at real data plus unit-variance Gaussian noise. [68]

measure of the Wasserstein-1 distance between the two distributions, they also note that the critic training loss increases after a while even as the validation loss continues to fall. In practice, conditioned WGAN-GPs often continue to improve well after the loss appears to be only randomly fluctuating. The loss is not a sufficiently reliable metric to rank various versions of generators or pick the best one.

- Loss of performance: The gradient penalty can inherently cripple the critic from learning certain patterns in the data. This will demonstrated in Chapter 5 and a solution will also be suggested in the context of HEP.
- Gradient penalty not applied in entire domain: In theory the gradient penalty on the critic must be applied everywhere, but in practice it is only applied at random points on a straight line between real and fake data points according to $p_{\hat{x}}$. Authors of Ref. [71] have attempted to improved upon it with an additional loss term.

Despite these shortcomings, in this thesis, WGAN-GPs were found to be most useful, since usually there were no problems with the availability of computational resources (important due to the long training time of WGAN-GPs, typically taking between 1 and 6 days depending on specifications of the training and hyper-parameters). Recently a large scale study of various flavour of GANs [72] also concluded that WGAN-GPs are the recommended flavour of GANs if computing resources are not very limited, and recommended a different kind of regularisation if resources are limited. Although the vanilla GANs are unstable, with the help of heuristic tricks, once in many tries they will outperform regularised GANs [73] and they therefore remain useful for organisations with extremely large resources, such as at Google.

4.5.2 Key aspects for application in HEP

Figure 4.8 – Interpolation of the latent space for BEGAN. [74]

DGMs have demonstrated the ability to interpolate smoothly through the input space, for example in Figure 4.8. Most of the research effort has however been on producing more and more photo-realistic images that impress the human eye, which is not necessarily the same as objective as correctly modelling a probability distribution. Some papers acknowledging that the entire probability distribution is not reproduced, for example the authors of Ref. [74] note "we see few older people and there are more women than men" in the generated images compared to the training dataset. This implies that care needs to be taken in the choice of DGM architecture in order to correctly model the full probability distribution.

It is not straightforward to make systematic assessments of probability distributions generated by DGM because marginal distributions do not give the full picture. Over-training is also tricky to spot. In the context of HEP, such models can in fact be evaluated by looking at reconstructed physics observables which catch the correlations that are of interest. In fact even relevant over-training can be spotted at the tails of distributions, because statistical fluctuations in the training set may result in systematic effects in the DGM distributions. Nevertheless, there is no single metric that can be used to confidently quantify the performance of the models, and manual comparisons by eye are prone to confirmation biases.

4.6 Likelihood-Ratio Trick

Machine Learning is traditionally used in particle physics for classification, trained on simulated data of signal and background events and then applied on unlabelled data recorded from collisions the LHC. The output of the model is either used to filter out background events, or used as an observable for the final fit.

In HEP the final sensitivity of the analysis is highest when the selection criteria has a high background rejection (fraction of background samples that are rejected) while maintaining high signal efficiency (fraction of signal events retained). Signals in HEP tend to be orders of magnitude smaller than the background, but it is customary to train a classifier for a signal vs background problem after equalising the weights of the samples, and then optimising the decision threshold to use based on the appropriate metric of performance evaluation (some of them are discussed in section 4.10).

The most powerful test statistic according the Neyman-Pearson lemma is the likelihood ratio,

$$\lambda\left(\mathcal{D};\theta_0,\theta_1\right) = \prod_{x\in\mathcal{D}} \frac{p\left(x \mid \theta_0\right)}{p\left(x \mid \theta_1\right)}, \tag{4.7}$$

where x are individual observations, $\mathcal{D} = \{x_1, \ldots, x_n\}$ is the observed data, $p\left(x \mid \theta_0\right)$, $p\left(x \mid \theta_1\right)$ are the conditional probability distributions of x for null hypothesis θ_0, and alternate hypothesis θ_1 respectively. These conditional probability distributions quickly become intractable, however, they can be sampled from using simulators. The simulations are usually done separately for the signal and background processes providing an estimate of the probability distributions $p\left(x \mid S\right)$ and $p\left(x \mid B\right)$. While $p\left(x \mid \theta_0\right) = p\left(x \mid B\right)$, $p\left(x \mid \theta_1\right) = p\left(x \mid S\right) + p\left(x \mid B\right)$.

It can be shown [75] that

$$\lambda'\left(\mathcal{D};\theta_0,\theta_1\right) = \prod_{x\in\mathcal{D}} \frac{p_{\mathbf{U}}\left(u = s(x) \mid \theta_0\right)}{p_{\mathbf{U}}\left(u = s(x) \mid \theta_1\right)} \tag{4.8}$$

is equivalent to Equation 4.7 for $\mathbf{U} = s(x)$ where s is monotonic with the density ratio

$$r\left(x;\theta_0,\theta_1\right) = \frac{p\left(x \mid \theta_0\right)}{p\left(x \mid \theta_1\right)}. \tag{4.9}$$

If a probabilistic classification model (which is approximating a Bayes optimal classifier) learns the decision function,

$$s^*(x) = \frac{p\left(x \mid \theta_1\right)}{p\left(x \mid \theta_0\right) + p\left(x \mid \theta_1\right)}, \tag{4.10}$$

and if it were perfectly trained, then its output would be sufficient to obtain

$$r(x|\theta_0, \theta_1) = \frac{1 - s(x)}{s(x)}\,. \tag{4.11}$$

Considering a signal strength (μ) estimation analysis, a classifier trained on equal weighted number of signal and background events to differentiate them learns the decision function,

$$c^*(x) = \frac{p\left(x \mid S\right)}{p\left(x \mid B\right) + p\left(x \mid S\right)}, \tag{4.12}$$

so for a parameter estimation of μ, we see from the relation,

$$\frac{p\left(x_i \mid \mu = 1\right)}{p\left(x_i \mid \mu = 0\right)} = \frac{p\left(x_i \mid S\right) + p\left(x_i \mid B\right)}{p\left(x_i \mid B\right)} = \frac{c(x)}{(1 - c(x))} + 1, \tag{4.13}$$

that indeed the output of the classifier does help arrive at the likelihood ratio.

In practice the classifier is never optimal but performs a useful dimensionality reduction. The output of the classifier is used an an observable for the final fit of the analysis. The last bin of the observable usually contains most of the useful information.

4.7 Physics Aware Models

Although it has been demonstrated that deep neural networks can learn high level features in HEP datasets [76], limited training statistics necessitates feature engineering by hand, or physics inspired inductive biases in the architecture such as Lorentz Boost Networks [77].

Going beyond classification, a new approach in HEP has been to re-imagine the inference framework given the new tools available, whether it be optimising the final objective directly [64], bump hunting [63], or detector unfolding [65]. The next section describes one such family of strategies [66] for likelihood-free inference which will be used in an off-shell Higgs to four leptons study in Chapter 7.

4.8 Likelihood-Free Inference with MadMiner

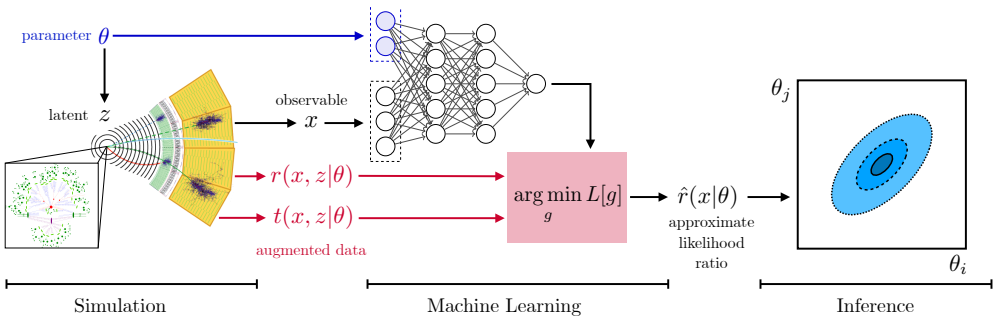


Figure 4.9 – Schematic overview of the family of techniques investigated [78]

In several fields and particularly in particle physics, the best description of a complicated process is often given by a simulation, rather than an analytical equation. These simulations usually can only run in forward-mode; given initial conditions and the laws of physics it can generate a possible outcome. Given the final outcome, the simulation framework cannot describe the possible history. Even though the full probability distribution of the outcomes becomes intractable, it can be sampled from using, for example a Monte-Carlo forward simulation. Particle physics is interested in the inverse problem.

It has already been described above how these simulated events in particle physics can be used to train models from which the likelihood ratio can be extracted, however, such a strategy loses out on the opportunity to fully take advantage of the simulation framework. Additional information about the hidden intermediate states, which would be completely inaccessible for real events, can be extracted from the simulator for simulated events. Since the final goal is to use the trained model to apply the model in real data, care has to be taken in designing a training algorithm that only requires this additional information at training time, but not at inference time.

In the HEP case, a family of ML based inference strategies have been recently introduced [66, 78–80] that could appreciably improve constrains on EFT (see section 2.1.5) parameters compared to traditional methods, some of these models can scale to multiple parameter estimation, while some others significantly improve sample efficiency (require less simulated data) compared to both traditional methods and generic machine learning methods that do not make use of the particle physics structure.

The use of such techniques is studied for the off-shell Higgs to four leptons analysis is studied in Chapter 7.

4.8.1 Key Ideas

The main objective in EFT studies at the LHC is often to measure a parameter, say θ, of the EFT Lagrangian using data from the collisions, x, i.e. measure $p(\theta|x)$. The authors of these techniques note that the probability of measuring x given some θ factorises into the parton-level process, which depends on the theory parameters, followed by the parton shower and detector interactions, which usually do not depend on the theory parameter,

$$p(x \mid \theta) = \int \mathrm{d}z_{\text{detector}} \int \mathrm{d}z_{\text{shower}} \int \mathrm{d}z \underbrace{p\left(x \mid z_{\text{detector}}\right) p\left(z_{\text{detector}} \mid z_{\text{shower}}\right) p\left(z_{\text{shower}} \mid z\right) p(z \mid \theta)}_{=p(x, z_{\text{detector}}, z_{\text{shower}}, z|\theta)}. \tag{4.14}$$

Here $z \equiv z_{\text{parton}}$ is the parton level momenta, z_{shower} is the state after parton showering, and z_{detector} represents the state after detector interactions. These are latent variables that cannot be observed in real life.

The *joint likelihood ratio* (imagine if the parton level momenta could be observed),

$$\begin{aligned} r\left(x, z \mid \theta_0, \theta_1\right) &\equiv \frac{p\left(x, z_{\text{detector}}, z_{\text{shower}}, z \mid \theta_0\right)}{p\left(x, z_{\text{detector}}, z_{\text{shower}}, z \mid \theta_1\right)} \\ &= \frac{p\left(x \mid z_{\text{detector}}\right)}{p\left(x \mid z_{\text{detector}}\right)} \frac{p\left(z_{\text{detector}} \mid z_{\text{shower}}\right)}{p\left(z_{\text{detector}} \mid z_{\text{shower}}\right)} \frac{p\left(z_{\text{shower}} \mid z\right)}{p\left(z_{\text{shower}} \mid z\right)} \frac{p\left(z \mid \theta_0\right)}{p\left(z \mid \theta_1\right)} = \frac{p\left(z \mid \theta_0\right)}{p\left(z \mid \theta_1\right)}, \end{aligned} \tag{4.15}$$

is independent of the showering and detector interactions, and it can therefore be extracted right at the parton level, where it is still tractable. Similarly, the *joint score*,

$$t\left(x, z \mid \theta_0\right) \equiv \nabla_\theta \log p\left(x, z_{\text{detector}}, z_{\text{shower}}, z \mid \theta\right)|_{\theta_0} = \left.\frac{\nabla_\theta p(z \mid \theta)}{p(z \mid \theta)}\right|_{\theta_0}, \tag{4.16}$$

which describes the gradient of the likelihood with respect to the θ can also be obtained for any desired value of θ.

These two quantities in themselves appear to be worthless, because in practice, z is unobserved. If only there were a computational tool that could learn to marginalise the dependence on z. In [79] the authors make the crucial connection to the useful quantities,

$$r\left(x \mid \theta_0, \theta_1\right) \equiv \frac{p\left(x \mid \theta_0\right)}{p\left(x \mid \theta_1\right)}, \tag{4.17}$$

$$t\left(x \mid \theta_0\right) \equiv \nabla_\theta \log p(x \mid \theta)|_{\theta_0}, \tag{4.18}$$

by defining functionals,

$$L_r = \mathbb{E}_{p(x,z|\theta_1)}\left[\left(r\left(x, z \mid \theta_0, \theta_1\right) - \hat{r}(x)\right)^2\right], \tag{4.19}$$

$$L_t = \mathbb{E}_{p(x,z|\theta_0)}\left[\left(t\left(x, z \mid \theta_0\right) - \hat{t}\left(x \mid \theta_0\right)\right)^2\right] \tag{4.20}$$

that are minimised by $r^*(x) = \arg\min_{\hat{r}} L_r = \mathbb{E}_{p(z|x,\theta_1)}\left[r\left(x, z \mid \theta_0, \theta_1\right)\right] = r\left(x \mid \theta_0, \theta_1\right)$, and $t^*(x) = \mathbb{E}_{p(z|x,\theta_0)}\left[t\left(x, z \mid \theta_0\right)\right] = t\left(x \mid \theta_0\right)$ respectively.

This minimisation is, of course, approximated by training neural networks on suitable loss functions, which are known to be able to learn the underlying decision function even with noisy labels. A schematic diagram of this approach is shown in Figure 4.9.

When a classifier is trained for a θ_0 vs θ_1 problem (two datasets generated at different values of a theory parameter, or even a signal vs background problem) with only {0,1} labels, the decision function it has to learn is very different from the labels provided. Training a model to regress the join likelihood ratio could provide labels that are closer to the target function. This is illustrated with a toy example in Figure 4.10. A visual representation of the behaviour of the score for a one dimensional toy example is also shown in Figure 4.11.

It is worth noting that the joint likelihood ratio for background events with a different final state (different composition of particles) will be a constant number (there is absolutely zero probability that a different set of particles at parton level can come from the process of interest, this is computed at the parton-level). While the additional information from the simulator is useful in training networks for theory parameter estimations, for simple signal strength measurement problems, a simple signal vs background classifier is likely the best choice.

4.8.2 MadMiner Package

`MadMiner` [81] is a python package developed to,

- Automatise the extraction of the additional information from `MadGraph5_aMC` [82].
- Perform *morphing*, a technique that allows to cheaply re-weight samples from one theory parameter point to another without the use of an event generator.
- Unweight events by sampling them based on their weights.
- Compute the additional labels required for training using the augmented data.
- Train various models using one of the pre-defined loss functions as specified.
- Perform simple inference.

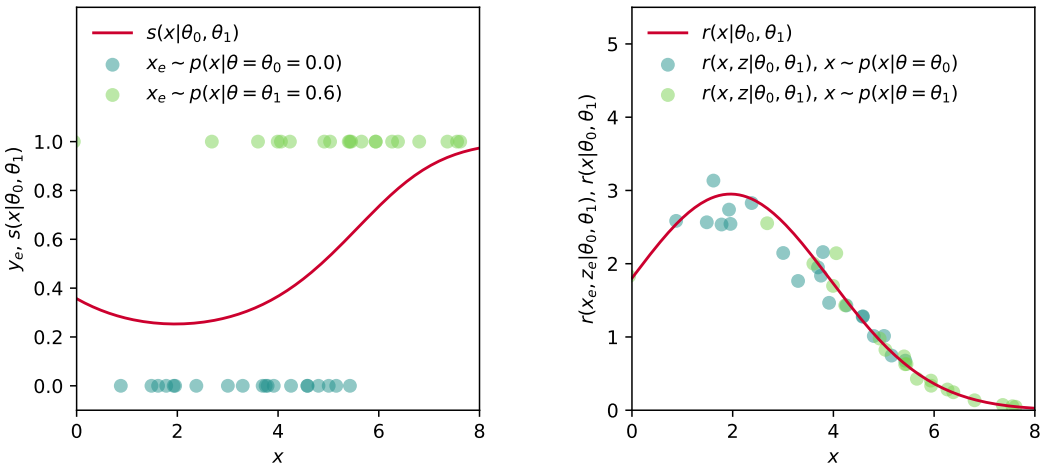


Figure 4.10 – Illustration of decision functions with a one-dimensional Gaussian toy example. Left: classifiers trained to distinguish two sets of events generated from different hypotheses θ_0, θ_1 (green dots) converge to an optimal decision function $s(x|\theta_0, \theta_1)$ (in red). Right: regression on the joint likelihood ratios $r(x_e, z_e|\theta_0, \theta_1)$ of the simulated events (green dots) converges to the likelihood ratio $r(x|\theta_0, \theta_1)$ (red line) [66].

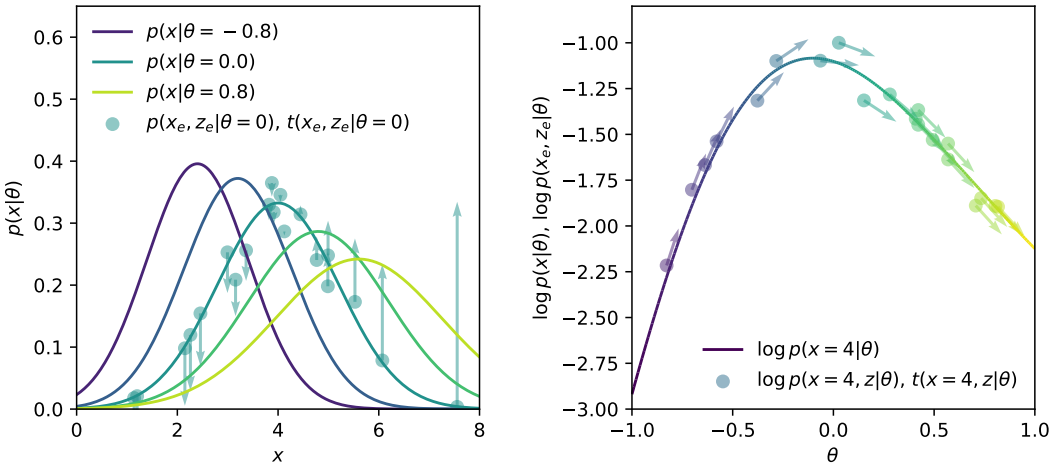


Figure 4.11 – Illustration of score with a one-dimensional Gaussian toy example. Left: probability density functions for different values of θ and the scores $t(x_e, z_e|\theta)$ at generated events (x_e, z_e). These tangent vectors measure the relative change of the density under infinitesimal changes of θ. Right: dependence of $\log p(x|\theta)$ on θ for fixed $x = 4$. The arrows show the (tractable) scores $t(x_e, z_e|\theta)$. [66]

The package can interact with `MadGraph5_aMC`, `Pythia 8` [83] and `Delphes` [84] to allow consolidated phenomenological studies.

The inference capabilities are likely to be extended with an upcoming version of `pyhf` [85]. To be able to use `MadMiner`, an LHC experiment will need to adapt its software framework to pass through the augmented data through the simulation chain, and write the data in a `MadMiner` readable format.

4.8.3 Mining Gold: The Additional Information

The "gold" that is "mined" from the simulation framework is a set of new weights for each event under different physics scenarios. The event generator can be given a parton-level distribution and asked what the probability is of such an event occurring based on some theory model. If θ is the parameter to be measured, an even can be generated at say, $\theta = \theta_0$ using an event generator. The value of θ can then be changed in the event generator and the probability of obtaining this same event can be computed for a new value of the parameter. At this stage the simulation is indeed able to run "backwards".

In practice these events are re-weighted using `MadGraph5_aMC` to several other benchmark points, $\theta = \{\theta_1, \theta_2, \theta_3, \ldots\}$. The number of minimum benchmark points needed depends on the physics process (as is shall be seen when morphing is discussed in Chapter 7).

The weight of an event is directly proportional to the squared matrix element. The different weights of an event indicate the probability of observing the event under the different possible values of θ, and can therefore be used to compute the joint likelihood ratio, $r\left(x \mid \theta_0, \theta_1\right)$.

4.8.4 Models that learn on augmented data

Several prescription of ML models are listed in [66] that take advantage of the additional the information from the simulator in different ways. The two that were found to be useful in Chapter 7 are described below.

4.8.4.1 ALICES

The first is a parameterised model called ALICES (**A**pproximate **L**ikelihood with **I**mproved **C**ross-entropy **E**stimator and **S**core)[86]. It takes as input the two values of the theory parameter, θ_0 and θ_1, that are to be compared as input, in additional to the features of the samples, x. The output is denoted $\hat{s}(x|\theta_0, \theta_1)$.

Starting from the standard binary-cross entropy loss,

$$L[\hat{s}(x)] = -\frac{1}{N} \sum_{(x_i, y_i)} \Big[y_i \log(\hat{s}(x_i)) + (1 - y_i) \log(1 - \hat{s}(x_i)) \Big] , \tag{4.21}$$

where $\hat{s}(x)$ is the output of the network, and y_i is a binary label $\{0,1\}$) that indicates whether or not the sample x_i is from θ_1. y_i is an unbiased but very high variance estimator for $s(x_i|\theta_0, \theta_1)$ (see Figure 4.10), but given that the joint likelihood ratio is available from the simulator, and the fact that it can be used to compute a much lower variance estimator $s(x_i, z_i|\theta_0, \theta_1)$ (see Equation 4.11), in this case an 'improved cross-entropy' loss can be obtained by replacing y_i with $s(x_i, z_i|\theta_0, \theta_1)$. This change also brings both terms of the cross-entropy loss into play for the same sample.

In addition, given that the neural network output is differentiable with respect to its inputs, the estimated score can be calculated as,

$$\hat{t}(x|\theta_0,\theta_1) = \nabla_\theta \log \hat{r}(x|\theta_0,\theta_1) = \nabla_\theta \log\left(\frac{1-\hat{s}(x_i|\theta,\theta_1)}{\hat{s}(x_i|\theta,\theta_1)}\right). \tag{4.22}$$

The joint score can therefore also be used to guide the training with an additional regression task. Note that this is different from the usual auxiliary task training where the network is made to regress an additional quantity at an additional output node in the hope that this task helps the network better solve the original problem. Here there is no additional output node, because the additional loss penalises the gradient of the same output node. The final loss of ALICES reads,

$$\begin{aligned} L_{\text{ALICES}}[\hat{s}(x|\theta_0,\theta_1)] = -\frac{1}{N}\sum_{(x_i,z_i)\sim p(x_i,z_i)} \Bigg[& s(x_i,z_i|\theta_0,\theta_1)\log(\hat{s}(x_i)) \\ & + (1-s(x_i,z_i|\theta_0,\theta_1)\log(1-\hat{s}(x_i)) \\ & + \alpha\,(1-y_i)\left| t(x_i,z_i|\theta_0,\theta_1) - \nabla_\theta \log\left(\frac{1-\hat{s}(x_i|\theta,\theta_1)}{\hat{s}(x_i|\theta,\theta_1)}\right)\Bigg|_{\theta_0}\right|^2 \Bigg]. \end{aligned} \tag{4.23}$$

Here α is a tunable hyper-parameter that balances the two loss terms. The factor $(1-y_i)$ is necessary to guarantee the correct minimum of the squared error on the score.

4.8.4.2 SALLY

The second model, SALLY (**S**core **A**pproximates **L**ikelihood **L**ocall**y**), is a neural network trained to regress the joint score $t(x,z|\theta_0)$ at the reference point θ_0, with a simple loss,

$$L_{SALLY}[\hat{t}(x)] = \mathbf{MSE}\left[t(x_i,z_i|\theta_0),\hat{t}(x_i)\right] \tag{4.24}$$

Since it is not parameterised on θ, it is only locally optimal [66] near the reference point, but requires far less training data compared to ALICES. The output of SALLY can be treated like any other observable, binned as a histogram for maximum likelihood fit.

4.9 Permutation Importance

Permutation Importance (PI) is a technique for evaluating the importance of a particular input feature to a model, based on an appropriate metric. Unlike feature importance provided by certain BDT packages (based on the internal parameters of the trained BDT and the training data), PI is computed on a given dataset, which may or may not even come from the same distribution as the training dataset.

Arguably, the best way to evaluate the importance of a features is with the iterative removal method. To start with, train one model with all features and another model with all-but-one feature, and use the drop in performance (based on an appropriate metric) to quantify the importance of the dropped feature. This assumes that the model is perfectly optimised for each case, which is rarely done in practice. But even if the models were perfectly optimal, this step alone would not perfectly account for correlations between features. For example, if two features, f_1 and f_2, are related as $f_2 = 2f_1$, then each may individually appear to have zero importance (if the model learns to get the same information from both features), although dropping both together might significantly reduce performance. To be more careful in feature

selection, one might choose the new model with one less feature than the original as the baseline, and iteratively find the next least important feature to drop.

The iterative addition strategy which is often used in ATLAS is the opposite. It starts with one feature and adds a new feature one at a time based on which one best improves performance. This suffers from even worse pathological problems. Consider two features often found in ATLAS datasets, the ϕ of the first and second jet. Individually, $\phi_{\text{jet1}}, \phi_{\text{jet2}}$ are totally symmetric in ϕ and give no useful information. They would never be selected in this strategy. However, as soon as both of them are included, the model might learn that $\Delta\phi_{\text{jj}} = \phi_{\text{jet1}} - \phi_{\text{jet2}}$ is an incredibly useful quantity.

Both these methods are computationally very expensive. In HEP, the search is made easier by first comparing one dimensional distributions of signal and background events for each feature and only investigating subset of these features that appear to have discriminating power. This strategy disregards all correlations and in fact even with domain knowledge, it is not obvious what higher order correlations might be useful for the model.

PI is a technique that attempts to simulate the removal of a feature for a trained model to assess feature importance. If the values of this feature in the dataset are simply replaced with zeros, the mean or even random numbers dawn from a uniform distribution, it may introduce biased behaviour in the model. The values must be replaced from a random distribution of the same shape as the distribution of this feature. This can be done in practice simply by shuffling the values of the feature between the samples. The overall distribution of the feature remains the same, but its correlation to the label of the sample is completely broken[7].

The drop in performance of the model due to the shuffling indicates the importance of the shuffled feature. The exact value of the importance will vary from one shuffle to another. Therefore the shuffling and evaluation for the same feature is performed multiple times (with a new shuffle permutation), and the mean and variance (or standard error) may be reported (which works under the assumption of a large number of samples).

Just like iterative removal, PI can provide unrealistic feature importance in cases where two or more features have high correlation, especially for random forests or neural networks with large drop out rates. These are models that are forced to learn multiple ways of reconstructing the same information, and therefore more likely to separately extract the same information from two correlated features, which makes each of them appear less important with PI.

The metric used to evaluate PI plays a large role in determining which features are important and it must be chosen with consideration for the final objective (which will be demonstrated in Chapter 6). For classification, the AUC is usually a good option, although in particle physics, there is often a better objective based metric.

The benefits of PI are:

- PI provides an error on feature importance.
- Applicable to most ML models, including neural networks. In principle applicable also to decision making algorithms that do not use any learning.
- Objective driven evaluation metrics (such as discovery significance) can be used instead of generic metrics such as cross-entropy gain.
- The value of the importance is interpretable (for example, the expected drop in discovery significance by removing a given feature).
- Can be evaluated on new datasets that come from a different distribution compared to the training dataset.

[7] This trick will be revisited in Chapter 8 for the Aspiration Network.

- In practice, PI rankings where found to be more stable compared to feature importance that come from trained BDT models for multiple identical trainings (with different random seed) in the work described in this thesis.

The idea of permutation importance can be extended to generative models, for example by evaluating how important each output cell of a generative network for a critic's evaluation of the Wasserstein distance.

There are several other tools to enhance interpretability of models with their own advantages and disadvantages (in terms of computational cost as well as interpretability), such as SHAP (SHapley Additive exPlanations)) [87] a strategy inspired from coolition game theory, but these will not be discussed here.

The large majority of ML models used in ATLAS are at the final analysis stage, to improve event selection or create a powerful observable. The ATLAS community prefers simpler models, trained with fewer input variables if it can produce similar results to a more complex one. It is partly because using fewer input variables demands fewer studies to entire that these features are well described by the simulation, and not too sensitive to systematic uncertainties. Neural networks are sometimes perceived to be less interpretable than BDTs. Considering a large amount of effort goes into feature selection in numerous analyses, PI is a very valuable tool to the ATLAS community.

Due to several drawbacks of existing implementations of PI, an open source package was developed that supports functionalities like sample weights, physics based metrics and AUC metric that can handle negative weights.
`PermutationImportancePhysics`[8].

4.10 Sensitivity Metrics

The particle physics community uses, and in certain cases has developed, a vast range of statistical tools for analysis. This section will only describe the essential ideas behind certain metrics used to evaluate the performance of various ML approaches in the following chapters.

The ideal way to evaluate a model that is meant to improve the sensitivity of a parameter measurement is to look at the negative log likelihood curve for the final fit. This is computationally expensive when performed repeatedly for a hyper-parameter scan. Some typical metrics that are used to estimate the sensitivity of an analysis strategy are briefly discussed below. These metrics however do not take into account quantum interference between signal and background events, and therefore their counterparts taking interference into account has also been obtained.

4.10.1 Simple case: Counting experiment without interference

Consider a counting experiment where an event might come from the signal process or some background process, and the objective of the experiment is to measure the signal strength μ. The total expected number of events is,

$$N_{exp} = \mu S + B, \tag{4.25}$$

where S, B are the expected number of signal and background events for the Standard Model, estimated with Monte-Carlo simulations. The Poisson likelihood of observing N events is then,

$$\mathcal{L} = \text{Poisson}(\mu S + B, N) = \frac{(\mu S + B)^N}{N!} e^{-(\mu S + B)}. \tag{4.26}$$

[8] `pip install PermutationImportancePhysics`

We can also compute the negative log-likelihood and its derivative,

$$-\ln \mathcal{L} = -N \ln (\mu S + B) + \mu S + B, \tag{4.27}$$

$$\frac{\partial - \ln \mathcal{L}}{\partial \mu} = -N \frac{S}{\mu S + B} + S.$$

The maximum likelihood estimator $\hat{\mu}$ is at the minimum of the negative log-likelihood,

$$\left.\frac{\partial - \ln \mathcal{L}}{\partial \mu}\right|_{\hat{\mu}} = 0 \implies N = \hat{\mu} S + B.$$

It can be verified that it is indeed a minimum with the second derivative. Next we assume a Gaussian likelihood curve, which is usually a reasonable approximation near $\hat{\mu}$. We can thus compute the width [9],

$$\sigma_\mu^2 = \frac{1}{\left.\frac{\partial^2 \ln L}{\partial^2 \mu}\right|_{\hat{\mu}}}, \tag{4.28}$$

$$\frac{1}{\sigma_\mu^2} = \frac{N S^2}{(\hat{\mu} S + B)^2} = \frac{S^2}{\hat{\mu} S + B} \tag{4.29}$$

and considering

$$< \hat{\mu} >= 1, \tag{4.30}$$

$$< \sigma_\mu >= \frac{\sqrt{S + B}}{S}. \tag{4.31}$$

This is the classical result. The inverse now gives us an approximate "significance" formula for measurement,

$$Z = \frac{S}{\sqrt{S + B}}. \tag{4.32}$$

4.10.2 Counting experiment with interference

Based on the discussion in Chapter 2 that lead to Equation 2.52 it is clear that in the case of interference, the total expected number of events no longer scales linearly with the signal strength parameter μ, contrary to Equation 4.25. It follows straightforwardly from Equation 2.52 that we can simply replace Equation 4.25 with

$$N_{exp} = \mu S + \sqrt{\mu} I + B, \tag{4.33}$$

where I is the interference component. As discussed in Chapter 2, the interference component can be (and often is) negative, which means that N_{exp} can sometimes decrease as μ increases.

[9] The second derivative of the log of a Gaussian function $g(x)$ provides the width σ:

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right), \quad \ln g(x) = \ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) - \frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}, \quad \frac{d}{dx}\ln g(x) = -\frac{(x-\mu)}{\sigma^2},$$

$$\frac{d^2}{dx^2}\ln g(x) = -\frac{1}{\sigma^2}.$$

In practice, I can be computed from Monte-Carlo samples[10] as,

$$I = SVI - S - V, \tag{4.34}$$

where S is the signal-only simulation, V is the simulation of only the background processes that interfere with the signal and SVI is a simulation of the signal and interfering background processes together, taking into account interference effects.

If we also include non-interfering background processes (denoted as $B2$) then the Equation 4.33 becomes:

$$N_{exp} = \mu S + \sqrt{\mu} I + V + B2 \tag{4.35}$$

but for the following derivation we will use

$$B = V + B2. \tag{4.36}$$

Following the same steps as the previous case,

$$\mathcal{L} = \text{Poisson}(\mu S + \sqrt{\mu} I + B, N) = \frac{(\mu S + \sqrt{\mu} I + B)^N}{N!} e^{-(\mu S + \sqrt{\mu} I + B)}, \tag{4.37}$$

$$-\ln \mathcal{L} = -N \ln (\mu S + \sqrt{\mu} I + B) + \mu S + \sqrt{\mu} I + B, \tag{4.38}$$

$$\frac{\partial - \ln \mathcal{L}}{\partial \mu} = -N \frac{S + \frac{I}{2\sqrt{\mu}}}{\mu S + \sqrt{\mu} I + B} + S + \frac{I}{2\sqrt{\mu}}, \tag{4.39}$$

$$\frac{\partial^2 - \ln \mathcal{L}}{\partial^2 \mu} = -N \left[\frac{\frac{-I}{4\mu^{\frac{3}{2}}}}{N} - \frac{-\left(S + \frac{I}{2\mu^{\frac{1}{2}}}\right)^2}{N^2} \right] - \frac{I}{4\sqrt{\mu}}. \tag{4.40}$$

Replacing Equation 4.40 in Equation 4.28 and using Equation 4.30 we get

$$\frac{1}{< \sigma_\mu >^2} = \left[\frac{I}{4} + \frac{(S + \frac{I}{2})^2}{N} \right] - \frac{I}{4} = \frac{(S + \frac{I}{2})^2}{N}, \tag{4.41}$$

so

$$< \sigma_\mu > = \frac{\sqrt{S + I + B}}{S + \frac{I}{2}}. \tag{4.42}$$

In analogy to equation Equation 4.32, a "significance" like formula will be just the inverse,

$$iZ = \frac{S + \frac{I}{2}}{\sqrt{S + I + B}}. \tag{4.43}$$

To use the formula in practice with MC simulated samples, the interference component will need to be replaced. Using Equation 4.34 and Equation 4.36 in Equation 4.43,

$$iZ = \frac{S + SVI - V}{2\sqrt{SVI + B2}}. \tag{4.44}$$

[10] In fact there is sometimes a trick by which interference "events" can directly be simulated from the event generator `MadGraph5_aMC`. The trick is to only allow first order terms of a parameter in the ME computation. Such samples were very briefly studied for work related to Chapter 7, but this additional study is not described in this document.

4.10.3 Asymptotic Formula

The authors of Ref. [88] show that the more precise metric for median expected significance is,

$$Z' = \sqrt{2((S+B)\ln(1+S/B) - S)}, \tag{4.45}$$

and it can be approximated to $S/\sqrt{(B)}$ when $S \ll B$. Following the same logic, the formula with interference reads,

$$iZ' = \sqrt{2\left[(SVI + B2)\ln\left(1 + \frac{SVI - V}{V + B2}\right) - (SVI - V)\right]} \tag{4.46}$$

Equation 4.45 and Equation 4.44 will be used in Chapter 6 for optimisation studies.

Chapter 5

Simulation of the Electromagnetic Calorimeter

Contents

The Standard Model (SM) of particle physics is being continuously tested at the LHC at the TeV scale. The scope for precision of measurements of deviations between the data and Mote-Carlo based simulations (MC) improves as more data is collected. Simulations of both SM and Beyond Standard Model (BSM) physics is required to tune analyses strategies so that precision measurements can be made using the data collected at the LHC. The simulation of particle showers in the calorimeter is important because this information is used for identification of particles and has an impact in many analyses, ones that select photons, electrons or pions and it is also important for jet calibration.

Precise simulations of the deposition of energy in the calorimeter due to developing showers are slow because they require the modelling of interactions of particles with matter at the microscopic level, as implemented using the `Geant4` toolkit [4]. In particular, `Geant4` models the chronological evolution of the cascade of particle showering in various materials with detailed spacial resolution, even if the only data recorded by the detector is the final stage of the cascade and with a coarse granularity. A particle shower produces an exponential growth in the number of particles to simulate for `Geant4`, increasing as a function of the energy of the incident particle. The physics processes that need to be modelled include but are not limited to Compton scattering, Reyleigh scattering, Coulomb scattering, pair production, annihilation, ionisation, photo effect, Bremsstrahlung, Cherenkov effect, transition radiation, scintillation, reflection and refraction. Even though certain simplifying assumptions are made in `Geant4`, modelling all these effects is computationally expensive. This is the reason particle showers usually take up the largest fraction of detector simulation time when the simulation is performed chronologically based on first principles.

The ATLAS detector has a complex calorimeter which proves to be the bottleneck in the simulation of events (in terms of CPU time), and the computational time scales with the energy of a particle showering in the calorimeter. In 2016 ATLAS spent 34% of computing wall clock time on simulation, and about 75% of simulation time is taken up by particle shower simulation [89]. Significant research and development is required to bring down the CPU consumption of these detector simulations to stay within the expected research budget shown in Figure 5.1. This would become a limiting factor in precision measurements, unless faster simulations are developed. ATLAS already relies on fast calorimeter simulation techniques based on thousands of individual parameterisations of the calorimeter response [5]. Such fast simulations are routinely used for exotics and BSM searches because the use of `Geant4` simulated MC datasets is computationally expensive.

The current fast simulator, `ATLAS Fast II` will be overhauled with the upcoming `FastCaloSimV2` [90], and still further upgrades are expected in the future. These allow significant gain in speed at the cost of accuracy. Such techniques rely on storing several parameterisation files and histograms in memory, and need to also compromise on accuracy to maintain a reasonable memory footprint.

Following a preliminary study on an ATLAS single photon dataset which included electronic noise and a continuous spread of true photon energy, an detailed study is performed in a small region $0.2 < |\eta| < 0.25$ in the central region of the electromagnetic (EM) calorimeter using photons. The intention behind this choice was to avoid regions of the calorimeter with discontinuities such as at $\eta = 0, 0.8$ in the barrel as well as barrel to end-cap transition regions. The chosen region of the calorimeter is segmented into cells in the r/z, η, ϕ space, with the EM calorimeter layers, namely presampler, strips (or front), middle and back, arranged one behind the other in the r direction, as seen in Figure 3.8. Each layer is made up of discrete cells with a particular width in η and ϕ (detailed in Table 3.1), they measure a total energy deposited and do not capture finer structures of the shower within the cell. Different cells sizes between the layers introduce a complication of periodic changes in alignment of different layers as they overlay the same η, ϕ region. The structure is presented in Figure 5.2, which shows the small cropped images which are part of the full cylindrical EM calorimeter. Further details about the ATLAS LAr calorimeter are presented in Chapter 3 subsection 3.2.3.

This chapter starts with a brief summary of the `FastCaloSimV2` strategy, followed by the study performed with a deep generative model. We study the viability of training a GAN to learn the probability distribution of the final image of the calorimeter directly for a given incident particle, bypassing the need for a first principles based physics simulation on-the-fly for full event simulations, and avoiding the need for memory-heavy, hand designed parameterisations. The first principles based simulation is still required to train the generative model but the cost can be quickly amortised in the course of millions of full event simulations.

5.1 Traditional Fast Calorimeter Simulation in ATLAS

The general idea in fast simulation is to simplify the geometry of the detector (but still more complex compared to `Delphes`, which is very briefly described in section 7.4), replace material interactions with analytical or parameterised interaction models. The reconstruction may also be simplified to gain speed but this aspect is not studied in this thesis. The ATLAS calorimeter in particular is complex (for example compared to the one in CMS) and required significant research and development for fast simulation. The calorimeter simulation proves to be the bottleneck in terms of simulation time if fast calorimeter simulation is not used.

The goal is to let `Geant4` simulate the particles till the calorimeter surface, and then handover the simulation to the fast simulation software for the shower. Meaning that the tracks will be correlated to the showers. Fast tracker simulation may be used in addition but the fast

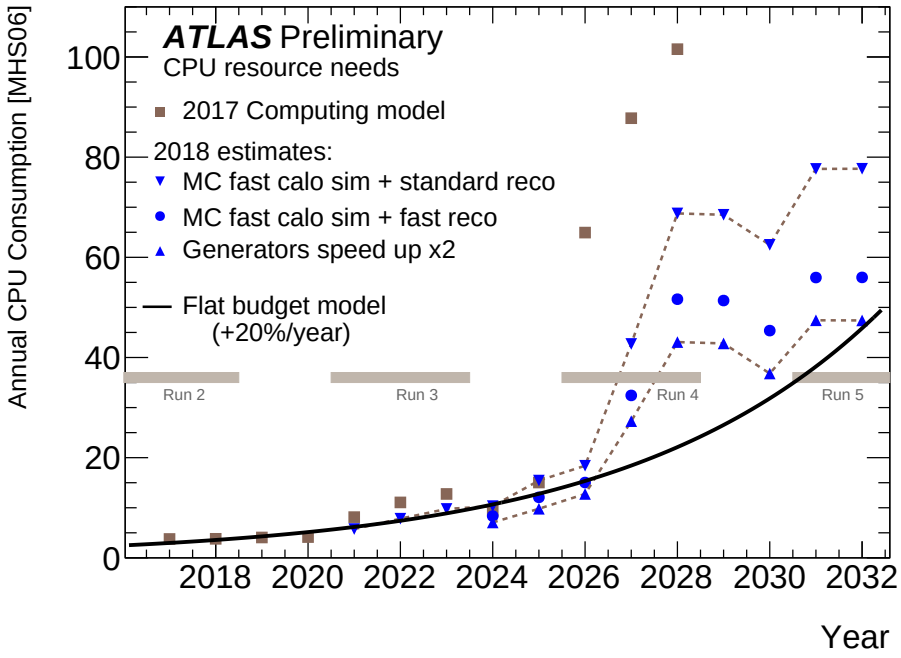


Figure 5.1 – The estimated CPU resources needed by the ATLAS experiment for data and simulation processing. The brown points are estimates made in 2017, based on existing software performance estimates and using the ATLAS computing model parameters from 2017. The blue points show the improvements possible in three different scenarios: (1) top curve with the fast calorimeter simulation used for 75% of the Monte Carlo simulation; (2) middle curve using in addition a faster version of reconstruction, which is seeded by the event generator information for the tracks; (3) bottom curve, where the time spent in event generation is halved, either by software improvements or by re-using some of the events. The solid line shows the amount of resources expected to be available if a flat funding scenario is assumed, which implies an increase of 20% per year, based on the current technology trends. [91]

simulation components remain modular. The calorimeter response can be factorised into showers from individual incident particles (such as multiple photons, electrons, pions). The calorimeter is fundamentally linear, so simulating the raw energy recorded from the shower of two incident particles is the same as simulating the raw energy recorded from the showering of each of them individually and then adding up the total energy deposited in each cell. This allows to parameterise individual particle showers and simply compose them to simulate full events.

5.1.1 FastCaloSimV2

The `FastCaloSimV2` parameterisation is based on a set of `Geant4` simulated single particle showers. The (longitudinal) parameterisation is performed in η bins of size 0.05 a succession of Principle Component Analysis (PCA) rotations (detailed below) for a fixed energy point (fixed particle true energy) and then an interpolation mechanism is used for particles with intermediate energies. For this reason, up to 10000 particles are generated on the calorimeter surface and the showering is simulated with `Geant4`. The lateral parameterisation is based on storing two-dimensional probability density histograms (also described below). This entire process is repeated for each energy point (17 in total), each particle type (3 in total) and each bin in η (100 bins ranging from -5 to 5), totalling 5100 sets of parameterisations without taking into account z-vertex spread and additional interpolation and correct models.

`Geant4` simulates hits with x, y, z coordinates which are cast into cells. `FastCaloSimV2` can therefore take advantage of this granular information to fit its parameterisation.

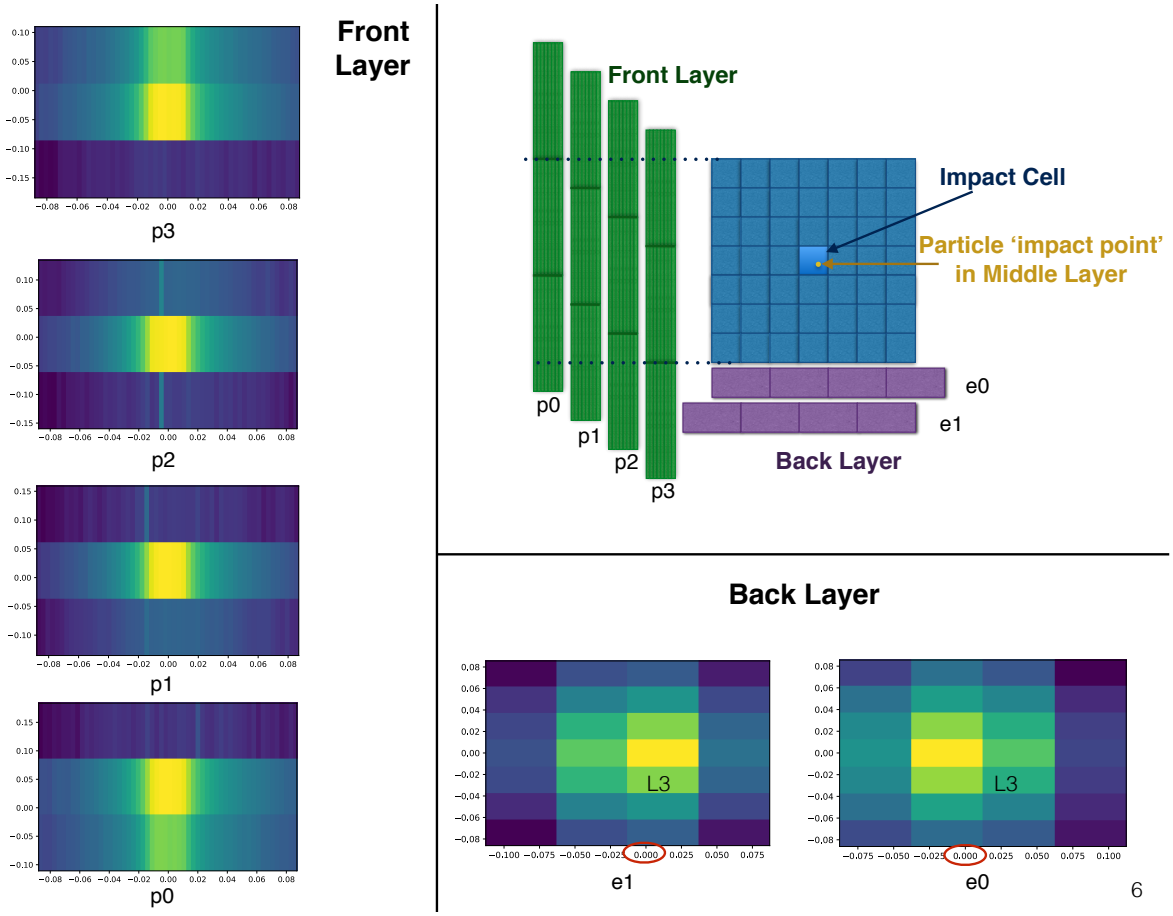


Figure 5.2 – Right Top: Different possible calorimeter layer alignments for the front layer (left, green) and back layer (bottom, purple) when the 3D calorimeter image is cropped with respect to the 'impact cell'. Each layer has the same number of cells regardless of the alignment. p0 through p3 are four possible alignments in ϕ for the front layer, (left, showing a 8×3 portion of the 56×3 cell image), and e0 & e1 are the two possible alignments in η for the back layer, (bottom, showing a 4×1 portion of the 4×7 cell image) with respect to the middle layer (centre, showing the full 7×7 image). The calorimeter layers are actually one behind another in the third dimension. [92]. Bottom: Average energy recorded in the Back Layer for the two possible alignments. Red circles highlight $\eta = 0$ (with the Impact Cell as the origin) falls in the third cell for e1 but second cell for e0. Left: Average energy recorded in the Front Layer for the four possible alignments. Although represented in a flat geometry, these images are small sections of the cylindrical calorimeter seen in Figure 3.8.

5.1.1.1 Longitudinal Parameterisation

The steps of the PCA-chain, which determine the longitudinal development of the shower are as follows:

- Inputs: The inputs to the PCA are the total energy (sum over all layers) and the fraction of this energy in each layer.
- Transformation: The input distributions are transformed into cumulative distributions (integrating over the bins) and then converted into Gaussian distributions using the inverse error function[1], shown in Figure 5.3. The x-axis units of these Gaussian distributions can be interpreted as probability quantiles.

[1] When z is real,

$$\operatorname{erf} z = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt,$$

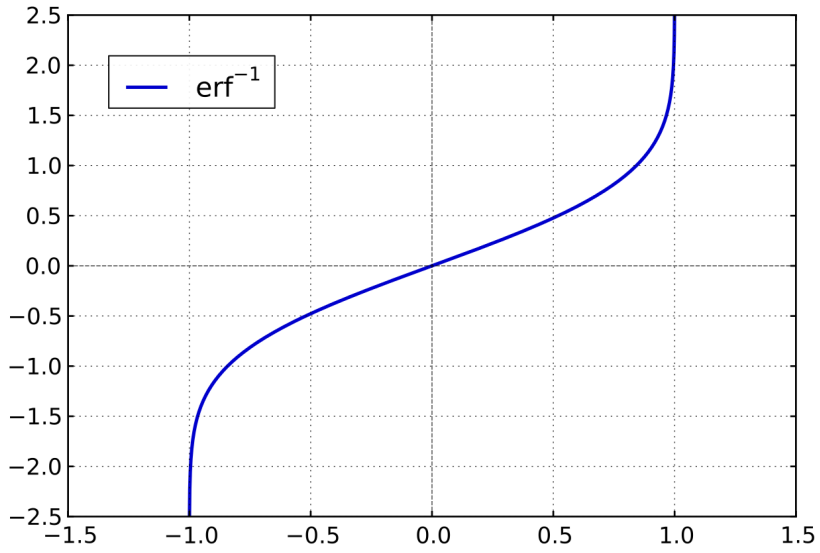


Figure 5.3 – Response of inverse error function.

- First PCA: The first PCA is applied to these Gaussian distributions without any dimensionality reduction (number of input and output dimensions remain the same) to covert the dataset into linearly uncorrelated components. These new axes are called Principle Components (PC), and they are sorted in order of their variance.
- PCA Bins: The leading PC of the first PCA is used to divide the dataset into quantiles (referred to as "PCA bins").
- Second PCA: For each PCA bin, a second PCA is applied to better decorrelate the data.

Figure 5.4 illustrates an example of this procedure carried out on showers with 65.5 GeV photons in $0.2 \leq |\eta| \leq 0.25$. The information that needs to be stored in a parameterisation file includes the cumulative energy distributions, the second PCA matrices, and the means and variance of the Gaussian distribution after PCA rotation. For fast simulation, this process is run in reverse:

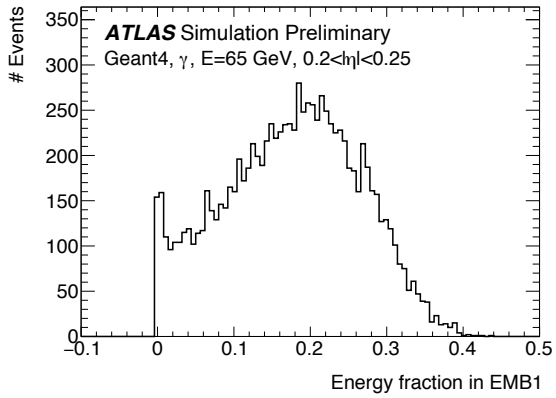
1. Choose a PCA bin at random using a uniformly distributed random number.
2. Sample random numbers from a Gaussian distribution (one for each PCA output component).
3. Rotate these numbers using the inverse PCA.
4. The resulting Gaussians are transformed into correlated uniform random numbers using the error function.
5. From the stored cumulative distributions, determine the energy distribution in each layer.

5.1.1.2 Lateral Parameterisation

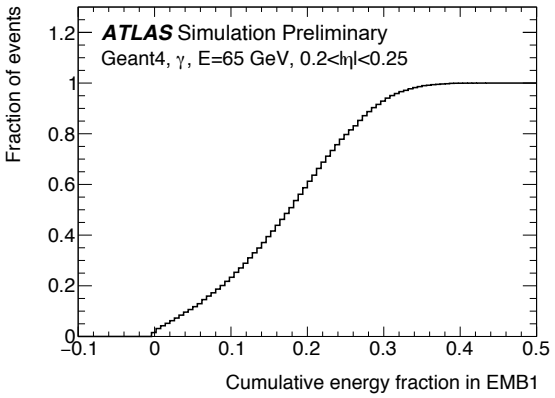
The lateral shower parameterisation is derived separately in each relevant layer (layers with more than 1% of total energy) for each PCA bin. This is done in two dimensions defined with

For $z \in [-1, 1]$,

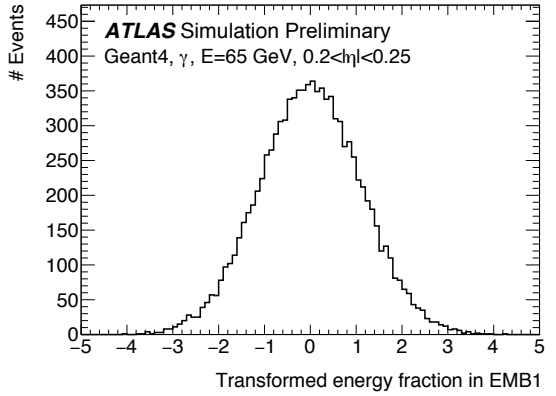
$$\mathrm{erf}^{-1}(z) = \sum_{k=0}^{\infty} \frac{c_k}{2k+1} \left(\frac{\sqrt{\pi}}{2} z \right)^{2k+1}, \qquad c_k = \sum_{m=0}^{k-1} \frac{c_m c_{k-1-m}}{(m+1)(2m+1)} = \left\{ 1, 1, \frac{7}{6} \ldots \right\}$$



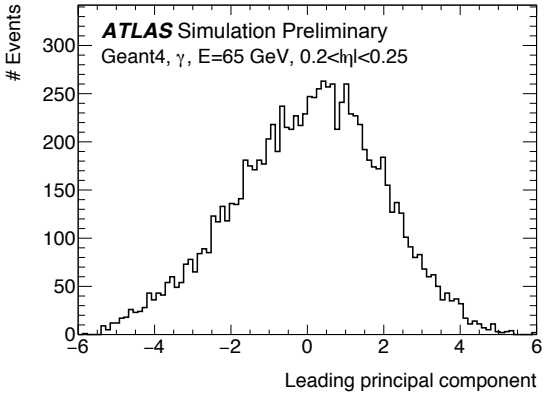
(a) Energy distribution



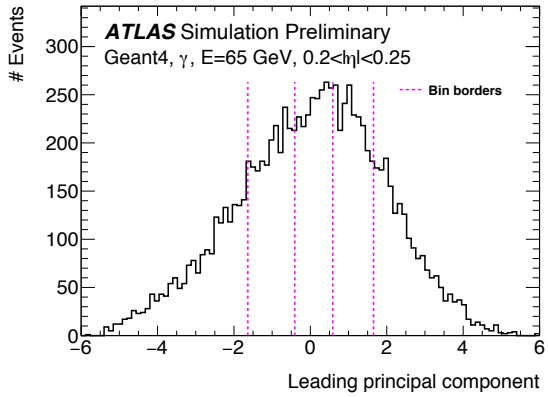
(b) Cumulative distribution



(c) Gaussian distribution

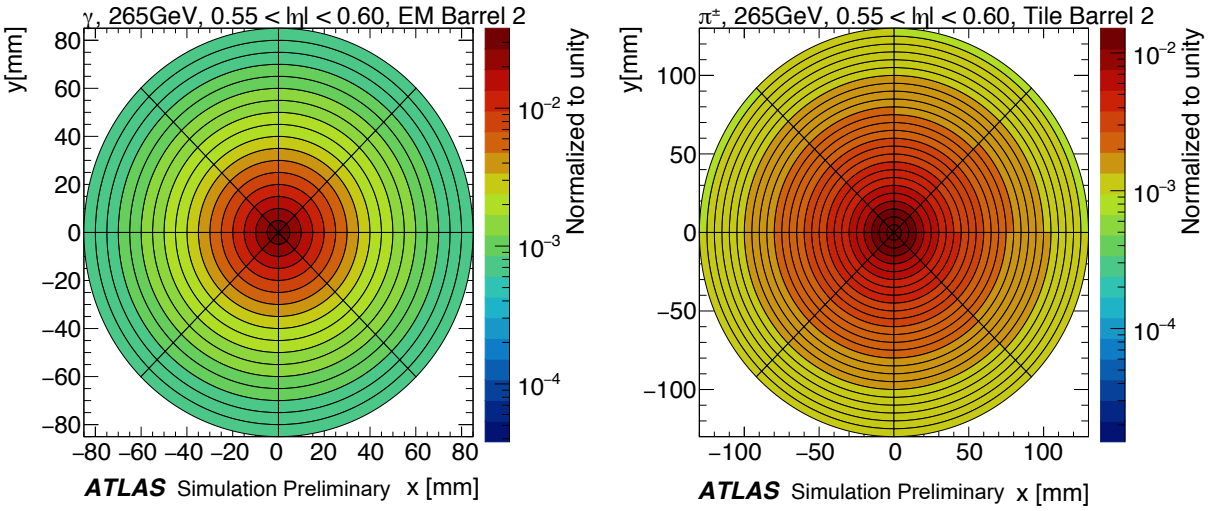


(d) Leading principal component after PCA rotation



(e) Leading principal component after PCA rotation and the cut values (in pink) that define the PCA bins.

Figure 5.4 – The steps of the PCA chain for one input sample of central photons with 65.5 GeV energy. (a) the distribution of the fractional energy in EM barrel 1, (b) the cumulative distribution, (c) transformed into a Gaussian distribution, (d) the leading principal component after PCA transformation and (e) the "PCA bins" in pink. [90]



(a) photon (b) pion

Figure 5.5 – The lateral shower development of (a) photons and (b) pions of energy 265.5 GeV in the range $0.55 \leq |\eta| \leq 0.60$ parametrised in the second layer of EM barrel and Tile barrel respectively. Th [90]

respect to the extrapolated position of the particle. For this reason, an extrapolation algorithm also exists to calculate the position of the particle as it traverses through the calorimeter if it had not showered.

The coordinate transformation is defined,

$$\begin{aligned} \Delta\eta &= \eta^{\text{hit}} - \eta^{extr} \\ \Delta\phi &= \phi^{\text{hit}} - \phi^{extr} \\ \Delta\eta^{\text{mm}} &= \Delta\eta \times \eta_{\text{Jacobi, hit}} \times \sqrt{r_{\text{cell}}^2 + z_{\text{cell}}^2} \\ \Delta\phi^{\text{mm}} &= \Delta\phi \times r_{\text{cell}} \end{aligned}$$

where

$$\eta_{\text{Jacobi}} = |2 \times \exp(-\eta_{\text{cell}}) / (1 + \exp(-2\eta_{\text{cell}}))|$$

and hit refers to to the energy distribution inside the calorimeter cell. The symmetry of the shower around the centre is exploited by transforming to a new coordinate set,

$$\begin{aligned} r^{\text{mm}} &= \sqrt{(\Delta\eta^{\text{mm}})^2 + (\Delta\phi^{\text{mm}})^2}, \\ \alpha &= \arctan 2\,(\Delta\phi^{\text{mm}}, \Delta\eta^{\text{mm}})\,. \end{aligned}$$

The two-dimensional histogram of the showers in these coordinates is also stored in the parameterisation file. Two examples of such histograms are shown in Figure 5.5. To reduce the memory footprint, only $0 \leq \alpha \leq \pi$ is stored with the assumption of a ϕ symmetry of the shower, which corresponds to storing only the top half ($y \geq 0$) of the histograms in Figure 5.5.

For shower simulation, hit coordinates are randomly sampled from these histograms. A simplified geometry is used to cast the hits to cells, which neglects the accordion shape of the detector, seen in Figure 3.9. A correction function is used to displace hits to neighbouring cells to mitigate this difference.

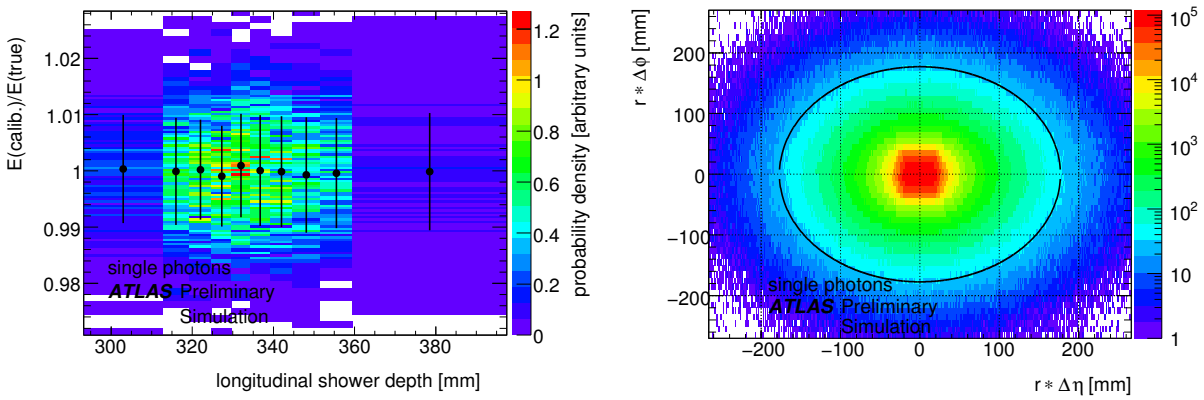


Figure 5.6 – 2-D Histograms of 200 GeV photons in the range $0.20 < \eta < 0.25$ used in the current `FastCaloSim` (`ATLAS Fast II`. (Left) Calibrated energy response and the longitudinal shower depth (Right) average simulated cell energy as function of the distance in $(\Delta\eta, \Delta\phi)$ of the cell from the expected photon impact point into EMB2. [5]

5.1.1.3 Additional Refinement and Casting

A spline interpolation is used between energy points. Further corrections such as for energy fluctuations and wiggles are also applied, but will not be discussed in this text.

Finally the simulated energy deposits are cast into cells with the coarse and irregular shape of the ATLAS calorimeter.

5.1.2 ATLAS Fast II (FastCaloSimV1)

The strategy used in the current `ATLAS Fast II` is similar in principle[5]. The parameterisation is done separately for each particle, η bin and energy point. For the longitudinal development, the depth is divided in 10 bins of equal number of showers. Two-dimensional histograms of energy vs shower depth (distance of the deposit from the calorimeter surface) are stored for total energy and energy fraction per layer. Correlations between the deposits in each layer is stored in correlation matrices. For simulation, random values are drawn from the histograms.

For lateral development, a radially symmetric third order polynomial spline function centred around the impact point of a particle in the calorimeter layer is used. Corrections are applied for asymmetries when particles are not perpendicular to the calorimeter surface. These parameters are obtained from a fit to the `Geant4` single particle shower data. Examples of the 2-D histograms used for the longitudinal and lateral parameterisation is shown in Figure 5.6.

`ATLAS Fast II` is also tuned to data on top of parameterisations based on `Geant4`, and this aspect will not be detailed here.

The expertise gained while building `ATLAS Fast II` is used to build `FastCaloSimV2` and therefore the first iteration does not use PCAs and several other refinements that are implemented in the upcoming new version. `FastCaloSimV2` has demonstrated that it outperforms `ATLAS Fast II` for certain cluster level variables [90] in the calorimeter. `FastCaloSimV2` is also faster and consumes less memory compared to `ATLAS Fast II`, although they both consume far more memory than ideally desirable (a comparison with the GAN is briefly discussed in section 5.5.5).

5.2 GAN for Fast Simulation

The idea here is to replace most of the components of the fast calorimeter simulation chain with one or a few generative networks (generative networks and GANs in particular are described in Chapter 4). Considering that the final images are at the granularity of cell images, a GAN could be trained directly on cell images, bypassing the need to cast hits to cells (drawbacks of this approach and proposed solutions will be mentioned in the end of this Chapter).

Such a GAN would need to be parameterised on the properties of the particle such as its energy, position, incident angle. It may be conditioned on the particle type as well, but given enough training data, it is simpler to train separate models for each of the three particle types. It will be shown in the next few sections that the GAN must also be conditioned on the geometry of the detector in the local region where the shower is being simulated. The incident angle was not used for this study, and left for follow up studies for which a dataset with various incident angles needs to be generated.

In this study, during model optimisation, the performance of the models were compared to `Geant4` samples standalone with the use of physics observables that can be computed outside the ATLAS software `Athena` and also using distributions that are more commonly used in the ML community. A suitable network was then integrated into `Athena` as a new simulation service for a realistic performance evaluation. The service picked up information about the incident photon, mimicked both the pre-processing and post-processing for the GAN, found the correct calorimeter cells and filled them with the generated energy.

This service can be used to simulate showers, use the full reconstruction of `Athena` (including the clustering algorithms and calibration) and compute complex observables for validation using the existing validation framework of `FastCaloSimV2`. This project therefore benefits greatly from the years of effort in the ATLAS collaboration to develop a list of distributions that can be used to assess various aspects of the performance of a fast simulation strategy. Many of these distributions in the validation framework were totally untracked during model optimisation.

The final step of this study is to consolidate and transmit the expertise gained on this approach for a full scale effort to simulate the entire calorimeter using a deep generative model. For this reason, a decision was taken to refrain from the use of inelegant, non-generalisable 'hacks' to refine the final performance of the GAN.

5.3 Dataset for the GAN

No dedicated `Geant4` showers were generated specifically for the GAN. A subset of the dataset generated for the `FastCaloSimV2` parameterisation was re-used to build the training dataset. The dataset built in this work was also used for a twin project to train a Variational Auto-Encoder (VAE) [92] for fast calorimeter simulation. The common input and output structures enabled fair comparisons between the two approaches [92]. These samples are described below.

5.3.1 Monte Carlo Samples

Samples of single unconverted photons are simulated using `Geant4` 10.1.patch03.atlas02, the standard MC16 RUN2 ATLAS geometry (`ATLAS-R2-2016-01-00-01`) with the conditions tag `OFLCOND-MC16-SDR-14`. The samples are generated for nine discrete particle energies logarithmically spaced in the range between approximately 1 and 260 GeV and uniformly distributed in $0.20 < |\eta| < 0.25$. 10000 showers each are generated for the lower and middle energies while 9000 showers are generated for last two energy points, totalling 88000 showers.

The truth particles are generated on the calorimeter surface, rather than at the interaction point. This is to avoid conversion of the particle before it arrives at the calorimeter, such as showering in the inner detector. The generated samples do not include effects corresponding to the expected beam spread or electronic noise. However, a small fraction of cells contained small negative energies (possibly from cross-talk between neighbouring cells).

The samples do include dead cells (due to an oversight) which breaks translation invariance (and can easily be simulated a posteriori), but their effect is expected to be minimal as the particle is never directly incident on a dead cell and it was estimated that only in $\sim 0.5\%$ of the samples does a dead cell exist within one cell distance of the extrapolated position of the particle in the middle layer. For the other layers also the impact of dead cells was estimated to very small.

5.3.2 Preparation of Training Dataset

The simplified geometry of `ATLAS Fast II` was used to extract the η, ϕ values for the cells and `FastCaloSimV2` extrapolation algorithm provided the extrapolated position of the particle in each layer of the calorimeter. Access to the geometry file allowed creating a dataset with more calorimeter geometry information, and this was crucial to improving GAN performance. Since the simplified geometry was used, the effects of the accordion shape of the calorimeter will not be reproduced by the GAN.

5.3.2.1 Cropping

The impact cell was defined as the cell in the middle layer closest to the extrapolated position of the photon (see Fig. 5.2), where the extrapolation is done under the assumption that the photon did not shower at all. For each layer of the calorimeter, the energy deposits within a rectangular region are selected with respect to the impact cell. The dimensions for the cropped layers are,

- Presampler: 7×3,
- Front (Strips): 56×3,
- Middle: 7×7,
- Back: 4×7,

totalling 266 cells. 99% of the total energy deposited is within this selection. The cells in each calorimeter layer are uniformly shaped, however their widths differs as a function of the calorimeter layer, both in η and ϕ directions. Therefore, the alignment between the layers is different from shower to shower.

5.3.2.2 Alignment Configuration

After cropping consistently with respect to the impact cell for each shower, the central cell in each layer is not necessarily at the centre of the cropped image. The possible alignment configurations for the Front layer (p0, p1, p2 and p3) as well as the Back layer (e0 and e1) are represented in Figure 5.2. Two of these configurations are illustrated using the cross-sectional view of the ATLAS detector in Figure 5.7. An individual shower will be in one such configuration, for example {p0, e0}. A simplified illustration of how the same shower can be recorded as different images due to the alignments is presented in Figure 5.8. The configurations for the Presampler in the ϕ directly follow that of the strips. In the η direction it has the same widths as the Middle layer and therefore there is never any misalignment.

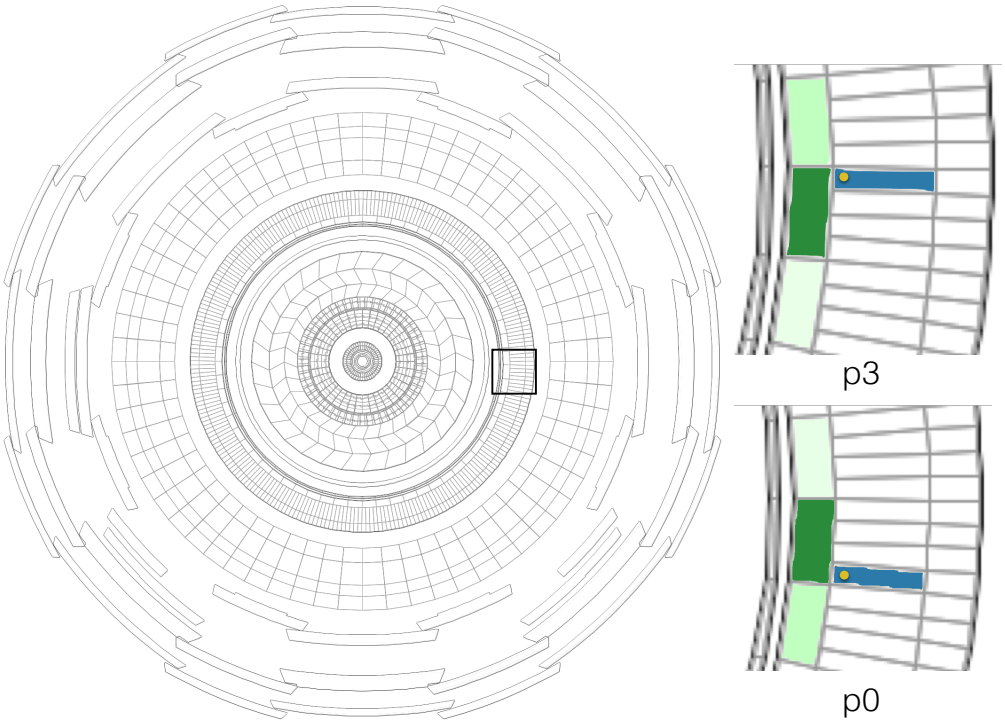


Figure 5.7 – Cross-sectional mesh-grid view of the ATLAS detector (a) in full, the black boxed region is zoomed and coloured in (b) under the p0 and p3 alignments. The yellow dot represents the extrapolated position of the particle in the middle layer, in blue is the impact cell, and in three shades of green the strip cells with varying amount of energy deposits (darker indicates more energy).

These alignments are a periodic function of the cell indices counting in the η (ϕ) direction for the Black layer (Strip/Presampler layer). The same alignment of the Middle Layer with respect to the Back layer (Strip/Presampler layer) occurs after moving every two (four) Middle cells in η (ϕ). This η index was calculated using the formula,

$$\text{index}_{\eta} = \frac{\eta_{\text{Impact Cell}} - c}{\delta\eta_{\text{Middle}}} \tag{5.1}$$

where $\delta\eta_{\text{Middle}} = 0.025$ is the width of cells in the Middle layer and c is some offset which is different for the two halves of the detector. The ϕ index was similarly calculated using,

$$\text{index}_{\phi} = \frac{\phi_{\text{Impact Cell}} - \phi_{\text{Reference Cell}}}{\delta\phi_{\text{Middle}}} \tag{5.2}$$

where $\delta\phi_{\text{Middle}} = \frac{2\pi}{2^8}$ is the width of cells in the Middle layer and $\phi_{\text{Reference Cell}}$ is some reference middle cell to start counting from. $\text{index}_{\eta} \mod 2$ and $\text{index}_{\phi} \mod 4$ are then useful quantities corresponding to the {e0,e1} and {p0,p1,p2,p3} configurations. An offset by 1 was required to make it match with the index computed using the inbuilt function in `Athena`.

5.3.2.3 Raw vs Real Coordinates

In reality, the ATLAS detector is not perfectly cylindrical, it sags from its own weight and it is shifted by a few millimetres from its nominal position. The real coordinates take into account such imperfections. The calorimeter cells' η and ϕ referred to in this document are the raw values, which assume a perfectly cylindrical shape.

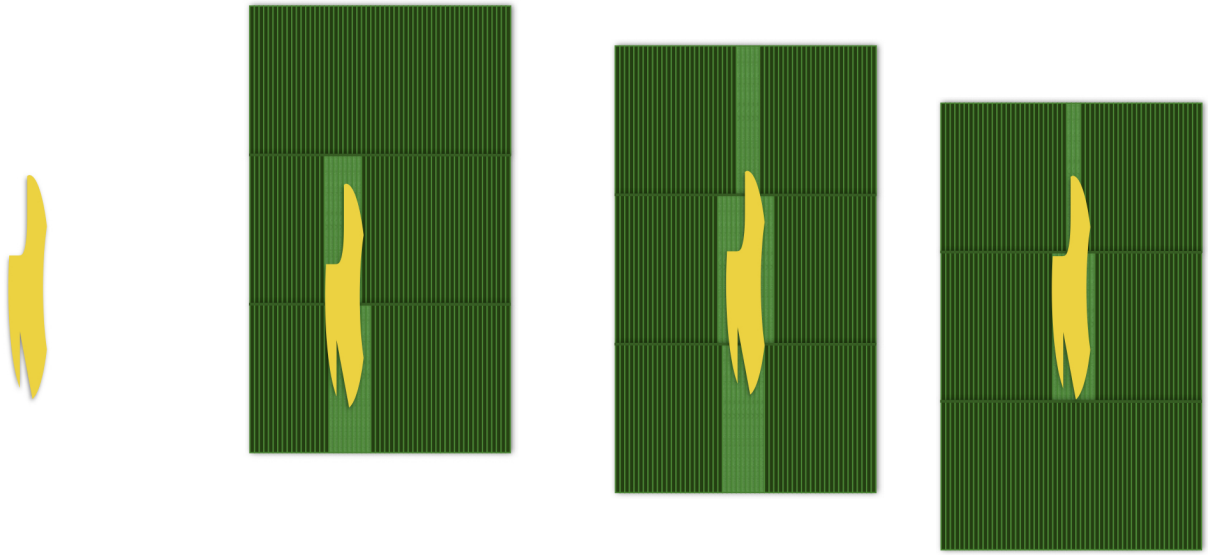


Figure 5.8 – Illustration of how a hypothetical shower (yellow) would look as an image on the Front layer cells for three different alignments of the Front layer with respect the Middle layer. Dark green cells do not record any energy, light green cells record energy from the shower

The extrapolation of the particle position however is performed using the real coordinates and this inconsistency[2] gives rise to certain problems that will be mentioned in Section 5.6.

5.3.2.4 Symmetry in the two halves

The two halves of the detector are identical but one is rotated to face with the other. The symmetry around the $\eta = 0$ plane is exploited to boost training statistics, by mirroring all shower in the $\eta < 0$ region. It is possible because there are no showers covering both halves of the calorimeter in this dataset. A shift of ~ 0.012 in ϕ between the two halves was compensated for in the cropping and calculations of alignment related variables.

5.3.2.5 Remove Negative Energies

Cell energies are required to be positive, therefore the small amount of negative energy found in a small fraction of the cells were set to zero in the dataset.

5.3.3 Advantages and Disadvantages of the Dataset

The advantages of using this dataset are:

- This dataset allowed training the network without electronic noise (which can be added in quickly and accurately in the ATLAS reconstruction software), and this also helped allow closer inspection of correlations.
- Building this dataset facilitated the addition calorimeter geometry information that was unavailable in the previous dataset used in the preliminary studies.
- With a common training dataset and output structure, fair comparisons could be made between the GAN and VAE approach without integration into `Athena`.

[2] This inconsistency was caused due to an oversight and could be fixed in the next stage of this project.

The disadvantages of this dataset are:

- The dataset comprises of up to 10000 showers for exactly the same particle energy but only nine unique energy points. The energy points are log-spaced with larger gaps at higher energies.
- The dataset does not simulate any beam spread. All the photons are perpendicular to the calorimeter surface, therefore the GAN cannot be conditioned in the angle of the incident particle.
- The dataset includes dead cells (a newer version of the parent dataset does not have this problem).
- The parent dataset contains small amount of negative energies in a few cells (which may be due to cross-talk between neighbouring cells).
- The dataset uses an inconsistent definition for η, ϕ coordinates for the position of the particle and the position of the cells. This could be corrected in future studies.

5.4 The GAN Model

A gradient penalty based Wasserstein GAN (described in Chapter 4 subsection 4.5.1) was trained for the given task. The architecture was modified slightly to include two critic networks, referred to as the 'critic' and the 'energy critic', and the logic behind the nomenclature will become evident in the following sections.

5.4.1 Pre-processing

Apart from the data processing already mentioned above particular transformations are performed in order to make the learning easier for the GAN. These steps are listed below.

- The cell energies are normalised by the true energy of the particle. This is to help the GAN learn the shower shapes despite the widely varying energies from shower to shower. Otherwise the cell energies would have extreme variations from a 1 GeV particle shower to a 262 GeV particle shower. The true energy is given as an additional input, so the GAN does not lose the total energy information due to this normalisation.
- The 3D image is converted into a flat vector. Although there are expected to be correlations between neighbouring cells in the η, ϕ, z directions, the dense network is expected to learn the spacial correlations on its own.
- The (natural) logarithm is applied on the true energy of the particle followed by a standard-normalisation before giving it as an input to the GAN. Since the energy points are log-spaced and the general variation of the shower shapes is also expected to be a function of the logarithm of the true energy.
- The alignment of the Strip/Presampler to the Middle layer is converted into a one-hot vector (see the terminology section 4.4 of chapter 4). Neural networks learn better when categorical variables such as this one are one-hot vector encoded.
- By the same logic the alignment of the Back to the Middle layer is also converted to a one-hot vector.
- The position of the particle in the impact cell is represented as the relative to the centre of the impact cell. These two values are further standard-normalised.

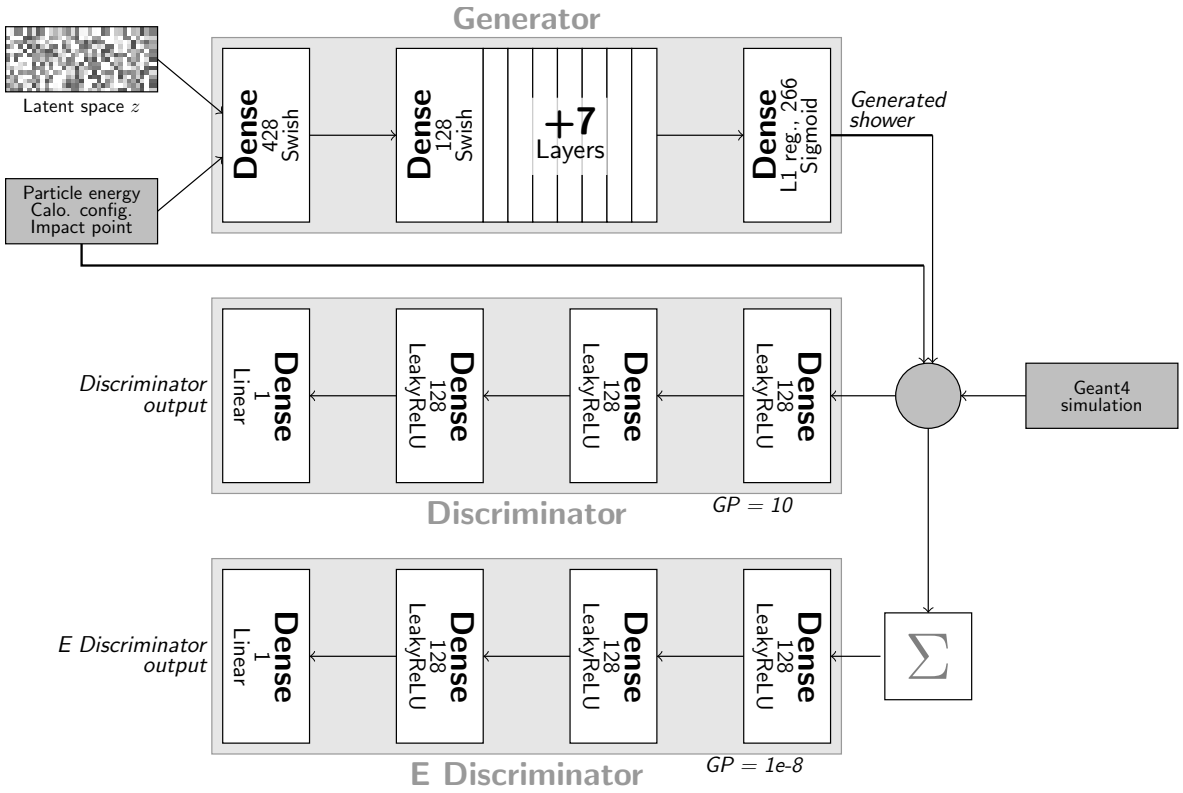


Figure 5.9 – Schematic diagram of the double critic GAN architecture (with a trainable swish[93] activation) used in this study.

5.4.2 Inputs and Outputs

The inputs of the Critic are:

A 266 Inputs: The cropped 3D calorimeter image consisting of 266 cells from the four electromagnetic layers, given as a flat one-dimensional vector.

B 1 Input: The true energy of the incident photon.

C 4 Inputs: The configuration of the alignment of Strip/Presampler layers given as a one-hot vector.

D 2 Inputs: The configuration of the alignment of Back layer given as a one-hot vector.

E 2 Inputs: The position of the extrapolated position of the particle in the impact cell (in η, ϕ coordinates).

Therefore the total number of inputs is 275. It has a single output, the estimated Wasserstein score.

For the energy critic apart A, the rest of the inputs (B through E) are the same from the first one. Instead of the 266 cells, it only takes the sum of these 266 cells as an input. The total number of inputs is therefore only 10. It also has a single output, the estimated Wasserstein score.

The generator also takes B through E as conditional inputs. In addition it takes 300 latent variables as input, which are random numbers sampled from a standard normal distribution. The total number of inputs is therefore 309. The output of the generator is a single vector of 266 numbers which represent the cropped 3D electromagnetic calorimeter image.

5.4.3 The Architecture

The GAN architecture developed in this work is shown in Figure 5.9. It consists of three neural networks, a generator and two critics (often referred to as a 'discriminators'), trained with a Wasserstein loss function[68] as proposed in Ref. [69]. The additional critic was required in order to overcome a typical limitation of gradient penalty based Wasserstein GANs, and will be elaborated upon later (section 5.4.6). All networks are conditioned on the energy of the incident particle, the alignments of the different calorimeter layers with the middle layer, and the extrapolated position of the particle inside the impact cell (see Fig. 5.2). The hyper-parameter optimisation is described in subsection 5.4.7.

The architecture of the critic consisted of:

- Three Dense hidden layers of 128 nodes.
- $LeakyReLu$[3] activation for the three hidden layers.
- Dense output layer with one node.
- Linear activation function for the output layer.

With 275 input features, the critic network therefore has 68,481 trainable parameters.

The architecture of the energy critic consisted of:

- Three Dense hidden layers of 128 nodes.
- $LeakyReLu$ activation for the three hidden layers.
- Dense output layer with one node.
- Linear activation function for the output layer.

With 10 input features, the energy critic network therefore has 34,561 trainable parameters.

The generative network is more complex and was found to be much more sensitive to hyper-parameter optimisation (as described in subsection 5.4.7. The architecture consists of:

- One Dense layer with 428 nodes.
- A trainable swish. activation [93] for the layer (one trainable parameter for the entire layer).
- Eight Dense layers with 128 nodes.
- A trainable swish activation for each of the eight layers (one trainable parameter per layer).
- One Dense output layer with 266 nodes (corresponding to the number of cells that are being simulated).
- A Sigmoid activation on the output nodes.
- An L1 activity regulariser with a weight of 10^{-5} to enforce sparsity in the output.

With 309 input features, the generator network therefore has 337,499 trainable parameters.

A gradient penalty (GP) can be interpreted as a penalty on how sharply the network changes its output as a function of the inputs. The GPs on the two critics were applied only on with respect to the image input space (item A for the critic, the total energy for the energy critic), and not on the conditional input space (items B through E), to allow the critics to make arbitrarily sharp

[3] $f(x) = \alpha \cdot x$ if $x < 0$, and $f(x) = x$ otherwise, with $\alpha = 0.3$.

decisions based on the conditional input. The critics can then be considered to be estimating the conditional Wasserstein distance between the real and fake images for a given set of conditional inputs.

The additional term in the loss of the critic due to the GP, as described in Chapter 4, is the Gradient Penalty Weight (GPW). The values for the GPW are 10 for the critic network and 10^{-8} for the energy critic (more on this apparently bizarre number below). The loss functions for the two critics can still be described by Equation 4.6, although x, $\tilde{x}$ and $\hat{x}$ are now single numbers, $\sum_{\text{cells}} x$, $\sum_{\text{cells}} \tilde{x}$ and $\sum_{\text{cells}} \hat{x}$ respectively, and the conditional nature of the critics is now implicit.

The generator is trained again the two critics and therefore also has two loss terms, the weights associated with the loss from the critic and energy critic are 1 and 10^{-6} respectively. The combined loss for the generator reads,

$$L_{\text{Generator}} = E_{\tilde{x}\sim p_{\text{gen}}}\left[D(\tilde{x})\right] + 10^{-6} \cdot E_{\tilde{x}\sim p_{\text{gen}}}\left[D_E\left(\sum_{\text{cells}} \tilde{x}\right)\right], \tag{5.3}$$

where $D(\tilde{x})$ and $D_E(\sum_{\text{cells}} \tilde{x})$ are the outputs of the first critic and the energy critic for a generated image $\tilde{x}$, and again the conditional nature of the critics is implicit.

5.4.4 The Training

The hyper-parameters used to train the GAN are listed below.

- Batch Size: 64 for all three networks
- Training Ratio (Number of critic updates for each generator update): 5 for both critics
- Gradient Penalty Weight: 10 for the critic, 10^{-8} for the energy critic
- Generator Loss Weights: $1 : 10^{-6}$ for the terms related critic and energy critic respectively
- Number of Epochs: 2500 epochs trained but best generator found at epoch 7500.
- Optimizer: *RMSProp* with lr=10^{-5}, ϵ = None, decay=0 and all other parameters left at their default values.

The training was performed on 50% of the available samples[4] (roughly 44000 showers). A comment about the learning curve is made in subsection 5.4.7. A training of 25000 epochs was completed in 79 h on a NVIDIA® Kepler™ GK210 GPU with a processing power of 2496 cores, each clocked at 562 MHz. The card has a video RAM size of 12 GB with a clock speed of 5 GHz. The training data size is 1 GB and is read from memory. Trainings for the hyperparameter optimisation are performed in parallel on multiple GPUs. These resources were provided through a batch system setup by the IN2P3 Computing Centre in Lyon.

The model is implemented and trained in `Keras 2.0.8` [94] using `TensorFlow 1.3.0` [55] as the backend.

5.4.5 Epoch Picking

In typical statistical fits and while training neural networks for classification, there are usually some mathematical guarantees for convergence of the fit, and such convergence is also empirically observed in practice. In the case for GANs, however, there are no convergence guarantees and they often fail to converge even in practice. The objective for the generator and the critic are

[4] No more official samples were available at these η regions at the time.

constantly changing in this minimax game, and the loss fluctuates. Furthermore, the version of the generator corresponding to the training iteration at which the Wasserstein component of the loss (which is the loss of the critic without accounting for the GP) appears to be the lowest may not be the best version of the generator in terms of physics objectives, since the critic knows nothing about the physics goals.

For these two reasons, a manual inspection of the generator networks is performed with the help of several physics observables (such as the energy response and the average η in the Back layer) as well as the Mean Squared Error (MSE) between the covariance matrix of the 266 cells for the network generated samples and the `Geant4` samples (see Figure 5.20). Neither Kolmogorov-Smirnov test nor Anderson-Darling test based rankings consistently corresponded well with rankings based on visual inspection of the distributions. Apart from the MSE of the covariance matrices, the energy resolution distribution, the distributions of the average η in the Back layer, the average ϕ in the Strips layer and the the correlations between the Middle and Strip layer were used to select the best version of the generator. These distributions will be shown in the validation section (Section 5.5) of this chapter. This methodology is highly susceptible to confirmation biases and human errors.

5.4.6 A peculiar problem and its solution: The Second Critic

This subsection will discuss a peculiar drawback of WGAN-GPs discovered in this work from the perspective of a detective and then propose a solution. It will start by showing hints of a drawback in the form of a mis-modelling of a physics distribution of interest, proceed to discuss how the usual solutions to fixing GANs failed, then show the discovery of the origin of the problem and finally demonstrate the solution.

5.4.6.1 Mis-modelling of the Total Energy

The first round of results were made public in Ref. [92] where only a traditional critic was used with a GPW of 10. Some of these comparisons will be presented in the beginning section 5.5.

The GAN failed to learn the total energy of the showers (which is simply the sum of energies in the 266 cells), even though it was able to learn the distribution of more complex physics observables, such as the width of the shower in η, ϕ. Figure. 5.10 demonstrates that this is the case both for the GAN (developed in this work) and the VAE (from the sister project). The means of the GAN match `Geant4` but not the widths (shown as error bars). An example of the total energy distribution for a single energy point is shown in Figure 5.11.

5.4.6.2 Failure of obvious solutions

Hyperparameter optimisation, many of the usual solutions used to improve GAN performances [67] and some intuitive ideas failed to make even the slightest improvement in the energy response modelling. These ideas included MiniBatch discrimination (where the discriminator/critic is effectively able to look at multiple data samples at once, helping it learn the distribution), different ways of conditioning a GAN, auxiliary task to regress the total energy, training at only a single energy point, an additional discriminator (of the traditional variety, trained with a binary-cross entropy loss).

Explicitly writing a custom layer to help the critic calculate the total energy (Figure. 5.12) also did not improve performance.

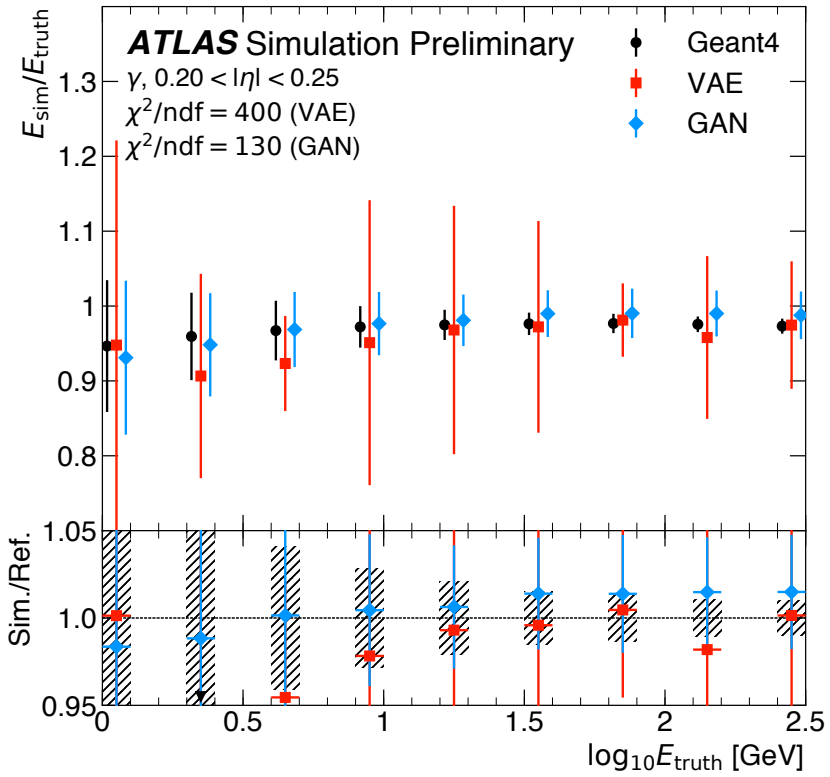


Figure 5.10 – Energy response of the calorimeter as function of the true photon energy for particles in the range $0.20 < |\eta| < 0.25$. The calorimeter response for the full detector simulation (black markers) is shown as reference and compared to the ones of a VAE (red markers) and a single critic GAN (blue markers). The shown error bars indicate the standard deviation of the simulated energy deposits at each energy point. The GAN is able to reproduce the means but not the widths of the distributions. [92]

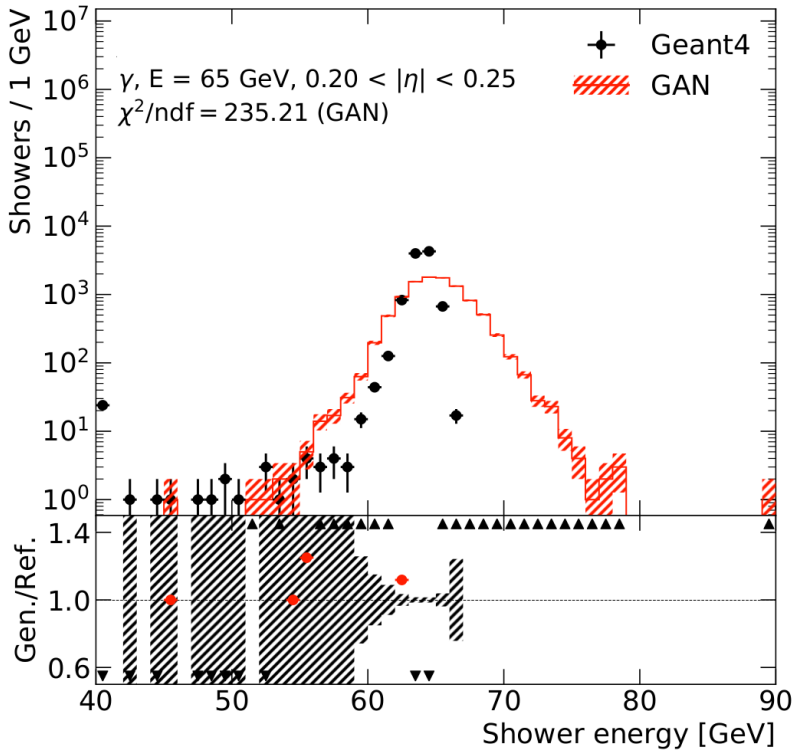


Figure 5.11 – Energy response of single critic GAN (red) in comparison to `Geant4` (Black) for 65.5 GeV incident photons. The width of the distribution is far larger for the single critic GAN.

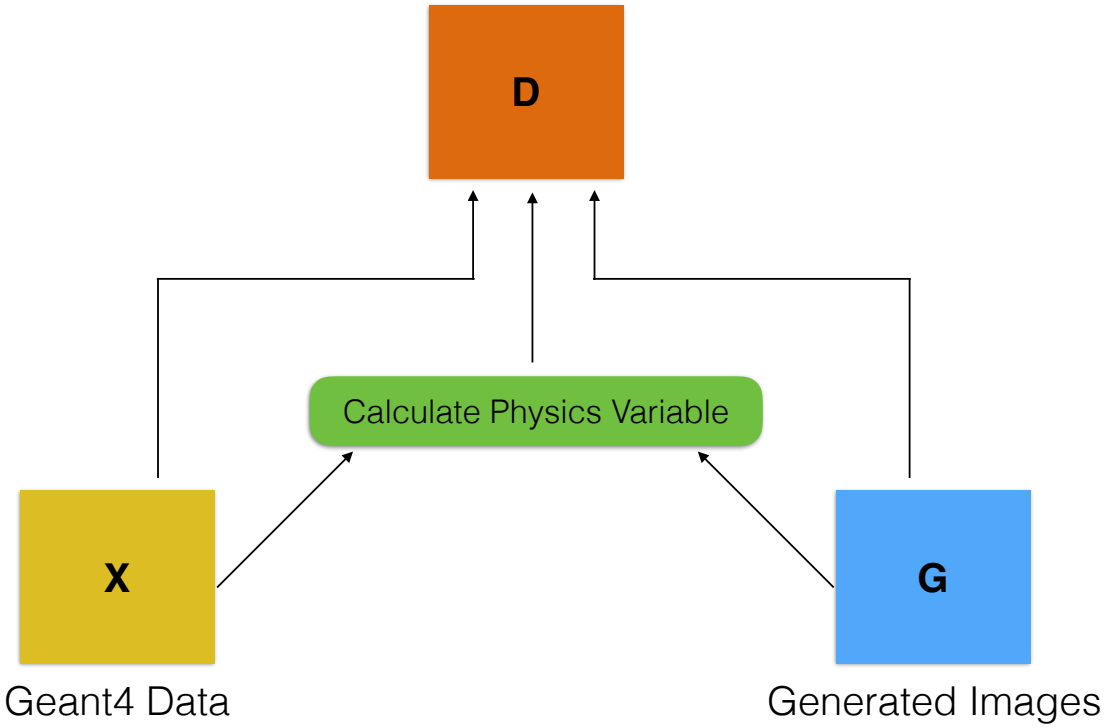


Figure 5.12 – Illustration of an idea to explicitly calculate physics variables from shower images to help the critic. It was not used in the final model.

5.4.6.3 Source of the problem: The Gradient Penalty

An inspection into the critic revealed that the network was flexible enough to learn to construct the total energy feature from the raw images but was prevented from using this information due to the gradient penalty. At the time, most WGAN-GP related literature on natural images advised that the GPW hyper-parameter does not require much tuning, and a typical scan between 1 and 500 could be performed during hyper-parameter optimisation. This was consistent with these preliminary findings in this study because the default value of 10 produced similar performance to 1 or 100. Interestingly, a GPW of 0 resulted in badly modelled distributions as well as crashes after a few epochs of training, as would be expected.

Dropping the GPW below $1e{-}13$ showed significantly improved energy resolution plots, although at the expense of all other distributions and training stability. If the total energy was given as an additional input feature but with no gradient penalty applied to it, again the total energy improved at the cost of other shower shape distributions. It is to be noted that to remove the gradient penalty on the total energy, it must be treated as an independent feature while training the critic (otherwise the gradients will pass through the total energy feature to 266 cells of the calorimeter image and the gradient penalty will indirectly restrict the critic from using this additional input feature) but a (sum) function of the calorimeter image while training the generator (so that feedback from the critic with regard to the total energy of the image simulated by the generator passes back to the generator in the form of gradients with respect to each of the cells of the output image).

5.4.6.4 The solution

Disentangling the two tasks into separate critic networks[5] allowed to inject domain priorities into the training algorithm. In addition to the original critic which focuses on the shower shapes, an additional energy critic with a GPW of 10^{-8} was trained to be blind to all aspects of the calorimeter image apart from the total energy. The energy critic does not provide unhelpful feedback about the shower shape to the generator because it only has access to the total energy.

The generator network is trained against the critic and energy critic simultaneously with a loss ratio of $1 : 18^{-6}$, forcing it to learn the shower shapes as well as total energy distributions simultaneously. The number of layers for the generator had to be increased for four to ten to keep up with the two tasks.

The gradient penalty on the critic is applied to restrict the class of functions it can approximate. Since the gradient is calculated with respect to the input features (albeit a random average of real and fake images), the critic is restricted from making very sharp decisions based on the total energy of the shower (sum of the 266 cells). This is not usually a problem for natural images where the sum of intensity of each pixel does not hold a lot of information, but for a calorimeter data it has a physical meaning. Thus, a lower GPW helped the energy critic force the generator to reproduce a sharp total energy distribution. The improved energy resolution using the new architecture is shown in Figure 5.13.

Since the energy response of the calorimeter is known very well, an alternate solution might have been to inject the known energy resolution function through post-processing, simply by multiplying the cell energies by an appropriate factor. The authors of the WGAN-GP also proposed two separate GANs, one only for the total energy and a second one for the shower shape (this could be implemented by either conditioning the second on the output of the first or by producing showers normalised to 1 with a *softmax* final activation layer and multiplying

[5] Idea suggested by Gilles Louppe, University of Liège, an ATLAS Analysis Consultant and Expert (ACE) and collaborator on this project, following discussions based on the gradient penalty based studies.

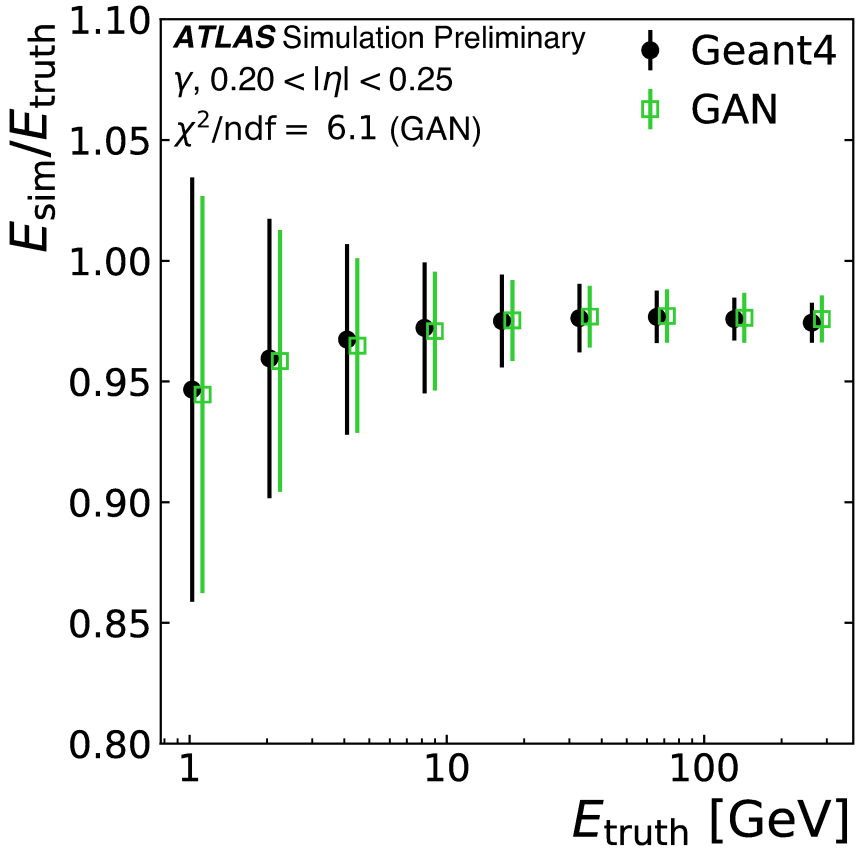


Figure 5.13 – Energy response of the calorimeter as a function of the true photon energy for particles in the range $0.20 < |\eta| < 0.25$. The calorimeter response for the full detector simulation (Geant4), is shown in black full markers used as reference and is compared to the one from the generative adversarial network (GAN), shown in green open markers. The GAN is shown with a small artificial shift towards the right for better visibility. The shown error bars indicate the resolution of the simulated energy deposits. The y-axis range is made smaller compared to Figure 5.10.

the total energy to it). Fortunately such inelegant solutions were not required.

Although such sensitivity to the GPW was not mentioned in any literature on gradient penalty based WGANs at the time, other projects that tried to use WGANs to model physics distributions have also encountered this problem in different forms. The WGAN trained for the CMS prototype High Granularity Timing Detector had trouble modelling the hit energy spectrum [95], a WGAN trained to generate full events used an additional Maximum Mean Discrepancy (MMD) loss to model the mass of the particle [96] and recently a new generative project used a post processing network to correct the energy spectrum [97]. In each of these cases, the other distributions were usually well modelled but an additional trick was required to fix the energy/mass distribution.

5.4.7 Hyper-Parameter Optimisation (HPO)

Many kinds of hyperparameter searches and various architectures were studied for this project and not all of them will be listed in this section. Instead, only some of the aspects will be mentioned below.

The performance of the generator was found to be much more sensitive to HPO compared to the critics, improving significantly with increased depth, increased width of the first layer and also improving significantly with the right choice of activation functions for the hidden and output layers. Although using a *ReLU* activation in the output layer allowed generate cells with exactly zero energy deposits, the lateral shower shapes were better reproduced with a *sigmoid* activation. The activity regularizer too significantly improved the performance of the GAN.

The performance of the GAN improved significantly when trained on more data, going from 4% to 40% and saturated by 75% of the available dataset. Note that the best version of the GAN happened to be trained with 50% of the entire dataset, possibly because the GAN was trained more times with 50% of the dataset compared to 75% (due to shorter training times) and by chance the best version happened to be a training on 50% of the dataset.

5.4.7.1 Switching optimizers between quick experiments and experiments on the full training dataset

Depending on the architecture and hyper-parameters training the GAN took between 1 and 6 days on a single GPU. The *Adam* optimizer produced better results when the GAN was trained on a small fraction (4%) of the dataset but did not scale well to more data. This is because the *momentum* in *Adam* pushes network updates in the same direction as previous updates, and this behaviour is not well suited for adversarial training where the dynamics change quickly. For this reason, trainings on a small fraction of the dataset (such as for quick tests of new architectures) were often performed with *Adam*, while trainings on a larger fraction of the dataset were performed using $RMSProp$.

5.4.7.2 Statistical Framework for HPO

At the initial stage of development, a statistical strategy was developed to tune hyper-parameters. GANs with exactly the same architecture and trained with exactly the same algorithm still produces very different results at the end of the training, shown in Figure 5.14b. This is the case when the random seeds are not fixed. The differences are much larger than what are usually seen in simple classification models because of the dynamical nature of the training and the lack of convergence guarantees. It was verified that an individual trained generative network produced reproducible results. Three sets of showers produced by the network were consistent with each

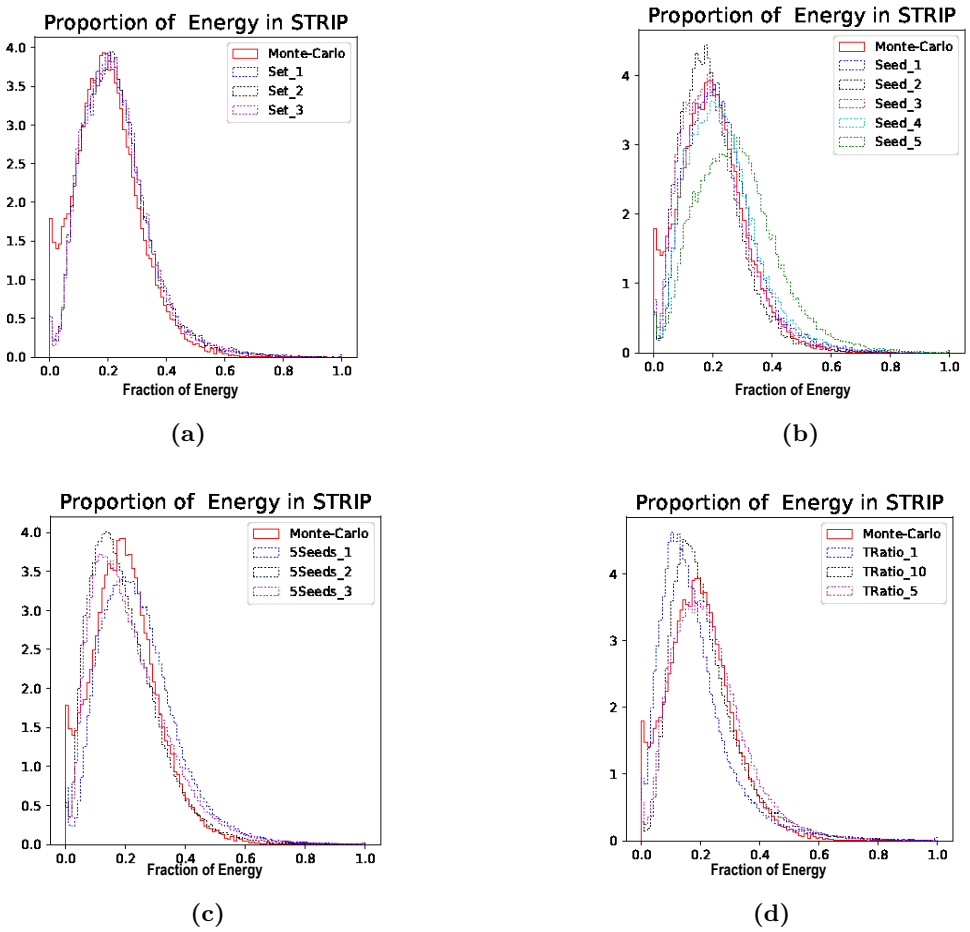


Figure 5.14 – Statistical approach to Hyper-Parameter Optimisation: (a) the same model consistently produces similar performances for different random seed during evaluation (b) models with identical architectures trained with a different random seeds exhibit very different performance, (c) three sets of average performance (data from 5 identically trained GANs) where all three sets of GANs have identical hyper-parameters reduce the variations, (d) three sets of average performance (5 identically trained GANs) where each set is trained with a different value of the hyper-parameter 'training ratio'.

other, as seen in Figure 5.14a. In fact the GAN also reproduced the statistical fluctuations at the tails of the training set consistently if over-trained. This can be seen in Figure 5.19b, where the GAN was trained only on 4% of the dataset.

Five identical GANs were trained with different random seeds and their average performance was compared to find hyper-parameters that consistently improve performance, beyond random chance. As a control, three sets of average performances (five GANs each) are also plotted (Figure. 5.14c) to gauge the statistical uncertainty of ranking hyper-parameter values in this way. If the variations between the sets of GANs with different hyper-parameters is larger than the variation between the sets of GANs all trained with the same hyper-parameter, then the ranking of hyper-parameters based on these plots is probably meaningful.

At a later stage of this project such comparisons became computationally infeasible and heuristical 'grad student decent' was used for the optimisation instead. Having a large latent space dimension was advantageous during development because too large a latent size was almost never a problem. Once a good architecture was found, identical models but with smaller latent sizes could be trained without deteriorating the performance.

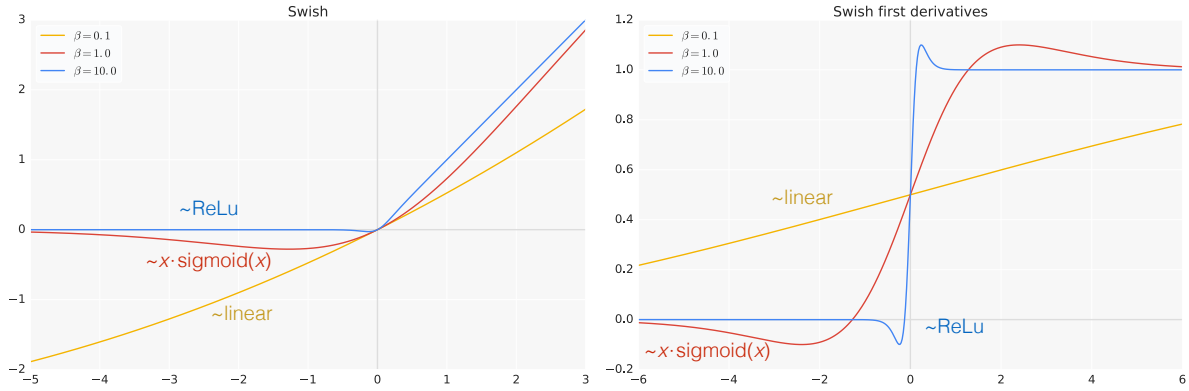


Figure 5.15 – (Left) Swish activation function for various values of β and (Right) the gradient of swish for these β values. The function is flexible enough to estimate a ReLu, $x \cdot sigmoid(x)$ or a linear function.

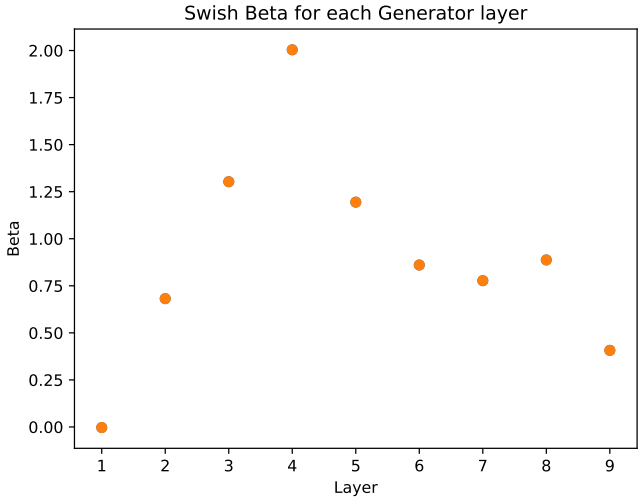


Figure 5.16 – Value of the β parameter of the Swish activation in the Generator network per layer.

5.4.7.3 Generator Network and Trainable Swish Activation

The generative network is more complex and was found to be much more sensitive to hyper-parameter optimisation as described above.

The swish activation [93] is defined as,

$$\text{Swish}(x) = x \cdot \text{sigmoid}(\beta x) \tag{5.4}$$

and is often useful for deeper networks as a drop-in replacement for *ReLU* or *LeakyReLU* with the β parameter is fixed to 1. This already provided slight improvement in performance of the model, and further improvement was observed by allowing β to become a trainable parameter of the model. The GAN performance improved if the swish was used either for the generator or the critic network but consistently performed worse when used on both. The best performance was obtained with the swish in the generator network. The flexibility of this activation function is illustrated in Figure 5.15 using particular β values where it estimates ReLu, $x \cdot sigmoid(x)$ or a linear function. The values of the trainable β parameter for each layer are shown in Figure 5.16. Several other activation functions such as the Scaled Exponential Linear Units (SELU) were also studied during the HPO.

The trainable swish activation was implemented using as a custom layer in `Keras`.

5.4.7.4 Some optimised hyper-parameters

Apart from the depth and width of each of the networks, the training ratio, GPW, batch size, number of epochs, learning rates, optimizers, batch normalisation, activation functions (Exponential Linear Units (ELU), SELU, hyperbolic tangent to name a few), drop-outs, 2D convolutional layers (with and without colour channels for the third dimension), 2D locally connected layers, relaxing the two sided gradient penalty to a one sided gradient penalty (and also increasing the threshold after which the penalty is applied), training size, instance noise were also studied. An intuitively promising architecture for the generator where it outputs a single calorimeter layer image at a time over the last four layers also not any better than an usual set of Dense layers.

An interesting point to note is that convolutional layers also did not improve the performance for the sister project of where photon showers are modelled with a VAE, however, 2D Convolutional layers did improve results over Dense layers for the VAE when it was trained to model energy fluctuations for showers coming from pions (energy fluctuations from shower to shower are much higher for pions than photons).

Other flavour of GANs such as the Improved WGAN-GP [71] (which adds a counter term to the loss function of the critic), Progressively Growing GANs [98](implemented in `PyTorch`), Vanilla GANs with a gradient penalty only on real or fake images [99] were also tried but without full hyper-parameter searches.

Several kinds of preprocessing of the inputs and the cell energies such as a PCA transformation, log transformations and separate scaling of energies per layer or per cell as well as ML tricks such as truncating the latent space distribution [73] to a maximum and minimum value were also studied.

5.4.8 Integration of generative models in ATLAS Simulation Software

A new FastCaloSim service called '`DNNCaloSim`' was created in Athena borrowing heavily from the in development `FastCaloSimV2` [100] infrastructure. The service builds the cell cluster, collects the conditional information available and computes geometry information on-the-fly needed for GAN, then simulates the shower using the generator network using the `Light Weight Trained Neural Network (LWTNN)` package [101] and finally performs the post-processing required to fill the energies into the calorimeter cells, which includes the mirroring of images for the left half of the calorimeter. The service was designed to be flexible enough to switch from a GAN based generative network to a VAE based generative network by changing a single flag in the simulation command.

The `LWTNN` is a minimalist inference package that designed to help translate neural network models trained in `Keras` into a format that is readable by its `C++ api`, and then use this to apply neural networks in large `C++` software frameworks. It avoids loading heavy ML libraries into the memory. It is built using the `Boost` library and is therefore a very small overhead for the `C++` based `Athena` framework.

The `LWTNN` package was updated[6] to support a trainable swish activation function so that the generator network could be integrated into `Athena`.

The trained generator network (which uses a custom activation layer written in `Keras`) was later

[6]This open source package was updated with the guidance of Daniel Guest, its main contributor

```
1. setupATLAS
2. asetup Athena,21.0,latest


3. Sim_tf.py --simulator 'G4FastCaloDNN' \
--geometryVersion 'default:ATLAS-R2-2016-01-00-01_VALIDATION'\
--inputEVNTFile "/eos/atlas/atlascerngroupdisk/proj-simul/OutputSamples/rel21/
mc16_13TeV.photon.E65536.eta20_25.EVNT.merged.pool.root"\
--outputHITSFile photons.G4FastCaloDNN.HITs.pool.root
--maxEvents 1 \
--preExec 'from ISF_FastCaloSimServices.ISF_FastCaloSimJobProperties import
ISF_FastCaloSimFlags;ISF_FastCaloSimFlags.ParamsInputFilename="DNNCaloSim/
DNNCaloSim_GAN_nn_v1.json"'
```

Figure 5.17 – Athena command to run DNNCaloSim.

translated to `TensorFlow 2.0.0` to be compatible with `ONNX Runtime` an inference package recently added to `Athena`. The model has also been tested[7] and runs also with `ONNX Runtime`.

This integration facilitates the validation of photon shower distributions produced by the generative network using `Athena` with the same rigour as other fast simulation techniques used in ATLAS. These distributions will be shown in the following section. An example command is shown in Figure 5.17.

5.5 Validation of distributions

This section will present the distributions obtained from the generative model and compare them to those from `Geant4`. The section will start by presenting the first set of distributions that were made public with a single critic GAN. These comparisons were made outside the `Athena` framework. This will be followed by comparisons of plots with the double critic GAN, first compared outside `Athena` and then present the performance of the generator network after being integrated into the `Athena` simulation framework.

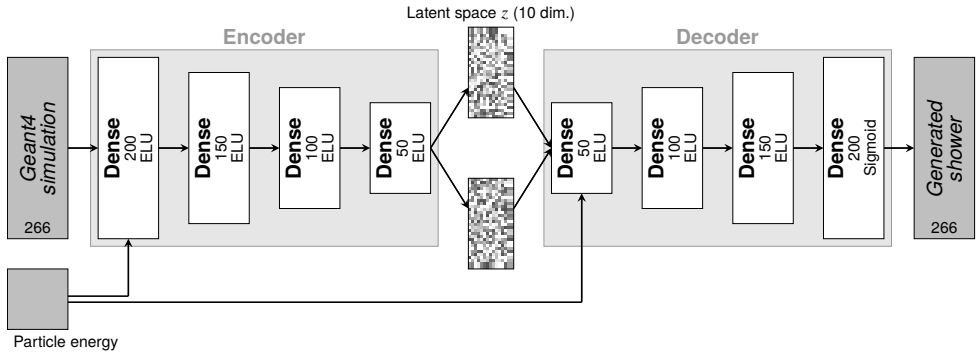
5.5.1 First Round of Public Results

This section presents comparisons between physics properties of the synthesised showers from the generative models and the full simulation that were made public in in [92]. A single critic was used for the GAN and it was not conditioned on impact point of the particle. The architecture of the GAN and VAE are shown in Figure 5.18. The histogram of the $\Delta\eta_{\text{Cell}}$ (defined as $\eta_{\text{cell}} - \eta_{\text{Impact Cell}}$ of the cell of the cells from all the 88000 showers for each layer of the electromagnetic calorimeter, weighted by the energy of the cells is shown in Figure 5.19.

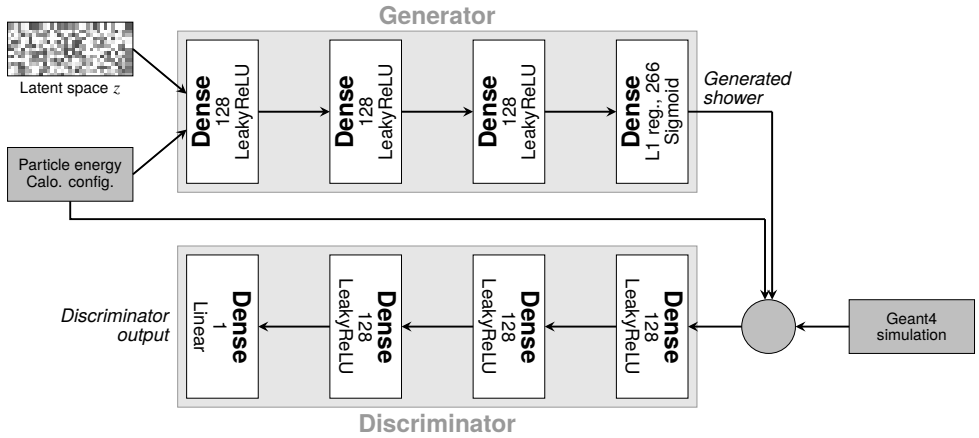
Both the GAN and the VAE are able to model these distributions reasonably well, but completely fail to model the energy response of the calorimeter, seen in Figure 5.10. The GAN was trained on only 4% of the dataset and therefore shows signs of over-training on the tails of Figure 5.19b, where it systematically reproduces the statistical fluctuations in the tails of the training dataset. The fluctuations do not go away even if more data is generated from the GAN. This was fixed in future iterations of the GAN.

It is to be noted that this is an unusual set of distributions because they do not depict the

[7] Tested by Debottam Gupta and Michael Fenton



(a) VAE



(b) First GAN

Figure 5.18 – Schematic representation of the architectures of (a) VAE, (b) the first GAN with a single critic, used in [92].

properties of the showers. The rest of this chapter will use a more intuitive set of distributions which are based on observables computed for individual showers.

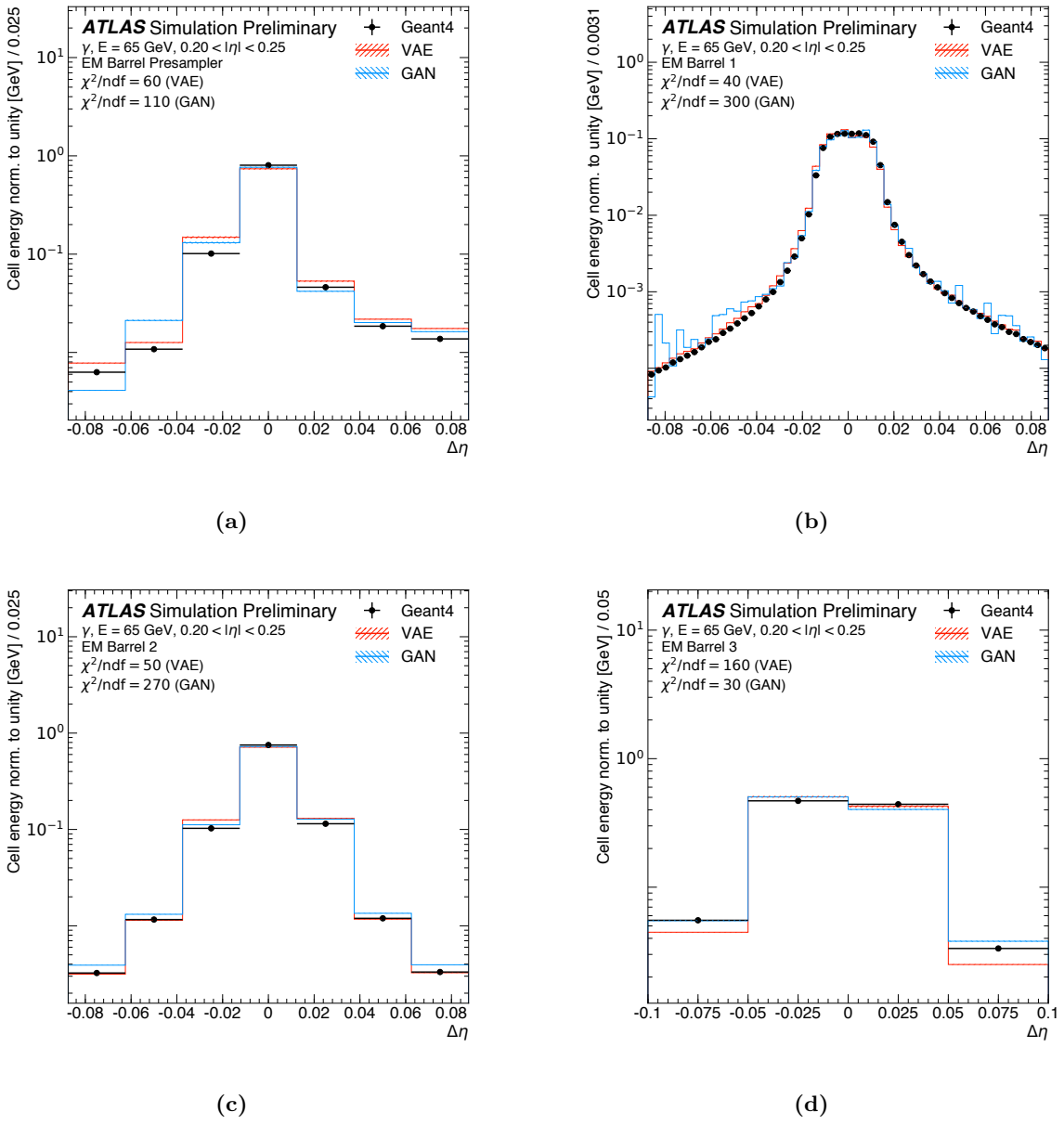


Figure 5.19 – Average energy deposition in the cells of the individual calorimeter layers ((a) presampler, (b) front, (c) middle, (d) back) as a function of the distance in η from the impact point of the particles for photons with an energy of approximately 65 GeV in the range $0.20 < |\eta| < 0.25$. The chosen bin widths correspond to the cell widths in each of the layers. The energy depositions from a full detector simulation (black markers) are shown as reference and compared to the ones of a VAE (solid red line) and a GAN (solid blue line). The shown error bars and the hatched bands indicate the statistical uncertainty of the reference data and the synthesised samples, respectively. The underflow and overflow is included in the first and last bin of each distribution, respectively. The showers simulated by `Geant4` deposit on average approximately 0.7 %, 17.2 %, 79.3 % and 0.4 % of the true photon energy in the presampler, front, middle and back layer, respectively. The showers synthesised by the VAE (GAN) deposit on average approximately 0.6 % (0.8 %), 19.1 % (19.8 %), 77.6 % (78.1 %) and 0.6 % (0.5 %) of the true photon energy in the presampler, front, middle and back layer, respectively. [92]

5.5.2 Standalone Validation

In the architecture optimisation phase of this project, the GAN simulated distributions were validated against `Geant4` Monte-Carlo using simple physics variables that can be calculated outside Athena. Electronic noise was not simulated in the training samples, which meant that often only a single cell recorded non-zero energy in a calorimeter particular layer, giving rise to unphysical single-bin peaks in the `Geant4` histograms. Standard feed forward networks struggle to produce exact zeros, and although these peaks could be reproduced with a ReLu activation on output layer of the generative network, the overall distributions were better reproduced by a sigmoid activation. A study of post-processing with electronic noise confirmed that these unphysical peaks go away and that the differences between the GAN and `Geant4` due to this effect would be washed away by the electronic noise. This will be shown in subsection 5.5.3.

The standalone validations were done for photons generated at all 9 energy points apart from a few distributions for which the fixed energy point is clearly stated.

The covariance matrix for the 266 cells using samples from all energies is shown in Figure 5.20.

In the following validation performed standalone will report the η and ϕ values measured relative to the centre of the impact cell (so the impact cell is always at $(\eta, \phi) = (0, 0)$. The averages and higher order moments are computed using all the cells in each layer (21 for the Presampler, 168 for the Strips, 49 for the Middle and 28 for the Back) for every shower and weighted by the energy deposited in the respective cell.

5.5.2.1 Impact Conditioning

The GAN was conditioned on the extrapolated impact point of the particle in the Middle layer (see Figure 5.2). The distribution of the difference between average η of the shower and the impact point ($\Delta\eta$) shown in Figure 5.21 demonstrates that the GAN learns to centre the shower around the impact point well, which is not possible without such a conditioning. The distribution of the difference between the average ϕ of the shower and the impact point ($\Delta\phi$) is shown in Figure 5.22 for all four layers. This also allows the GAN to reproduce the "S-shape distributions", the average η of the shower as a function of the particle impact η, which represent edge effects induced by the discrete nature of the calorimeter geometry. This distribution is shown for the Middle and Strip layers in Figure 5.23 where a single "S-shape" for the Middle layer and eight "S-shapes" for the Strip layer are seen. Reproducing this effect is not trivial because the GAN simulates the cells directly instead of simulating hits and casting them to cells like `FastCaloSimV2`. It is however clearly visible that the GAN underestimates the spread of

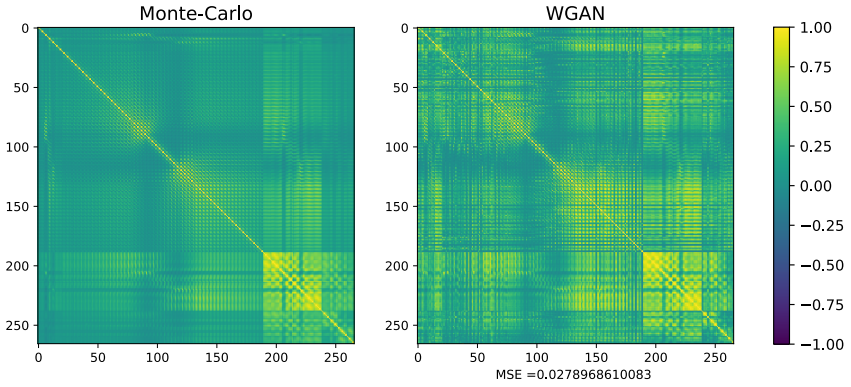


Figure 5.20 – Covariance Matrix for the 266 cells using samples from all energies for `Geant4` (left) and the GAN (right).

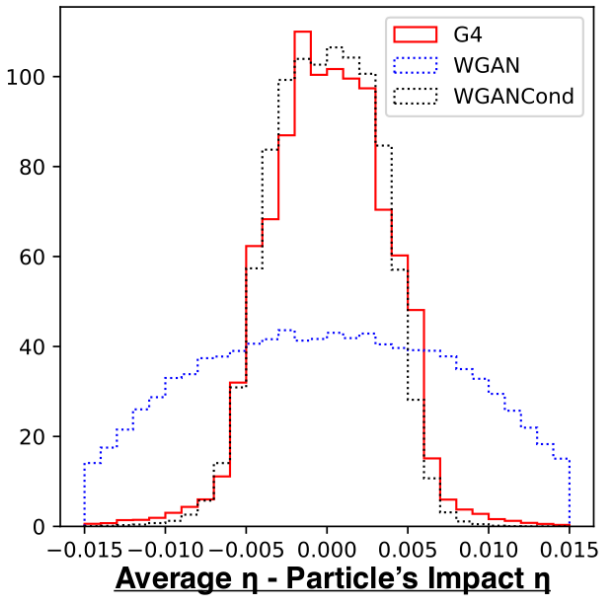


Figure 5.21 – Distribution of difference between the average η and the extrapolated impact η of the particle in the Middle layer: Blue is a GAN without conditioning on impact point, Black is a GAN with conditioning

the average η in the Strip layer for any given impact η position.

The impact the conditioning has on the correlation between the Middle and Strip layers is shown in Figure 5.24. A slight improvement is observed, which indicates that providing additional physics information is helpful (even though in principle the GAN could learn this correlation without the impact position information).

5.5.3 Standalone Noise Studies

Figure 5.25 for an older version of the GAN shown how the distributions in Figure 5.22 are affected when electronic noise is added. The unphysical peaks in the `Geant4` that the GAN does not reproduce are washed away.

5.5.3.1 Detector Geometry Conditioning

The GAN was also conditioned on the alignment of the different calorimeter layers shown in Figure 5.2. The shower width in the strip layer for two different alignments is shown in Figure 5.26. The average η of the shower for the two alignments of the Back layer are well reproduced by the GAN, as can be seen in Figure 5.27. An old example of a GAN trained without this conditioning is shown in Figure 5.28 as contrast. While monitoring the evolution of the GAN training over many epochs, this conditioning was usually one of the last features of the dataset the GAN learnt.

This particular aspect of simulating a changing detector geometry (which occurs because the images are small cropped portions of the full calorimeter) has not been addressed by any prior work on simulating calorimeters with generative networks (to the best of the author's knowledge).

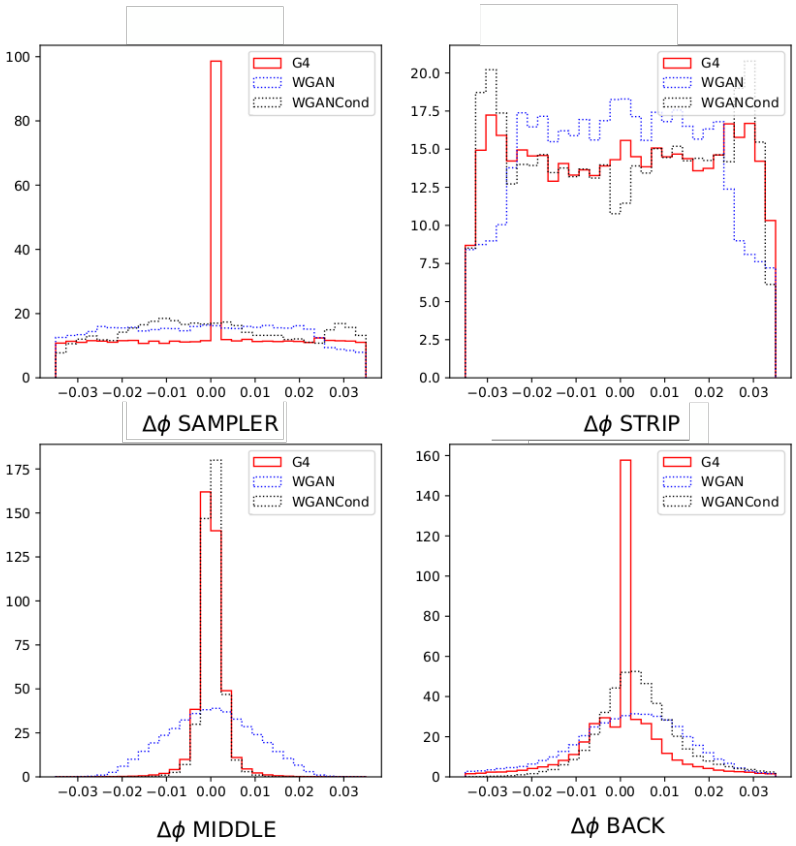


Figure 5.22 – Distributions of the difference between the average ϕ and the extrapolated impact ϕ of the particle ($\Delta\phi$) in all four layers: Blue is a GAN without conditioning, Black is a GAN with conditioning. Comparisons are made for `Geant4`, a GAN not conditioned on the particle position and a GAN conditioned on the particle position.

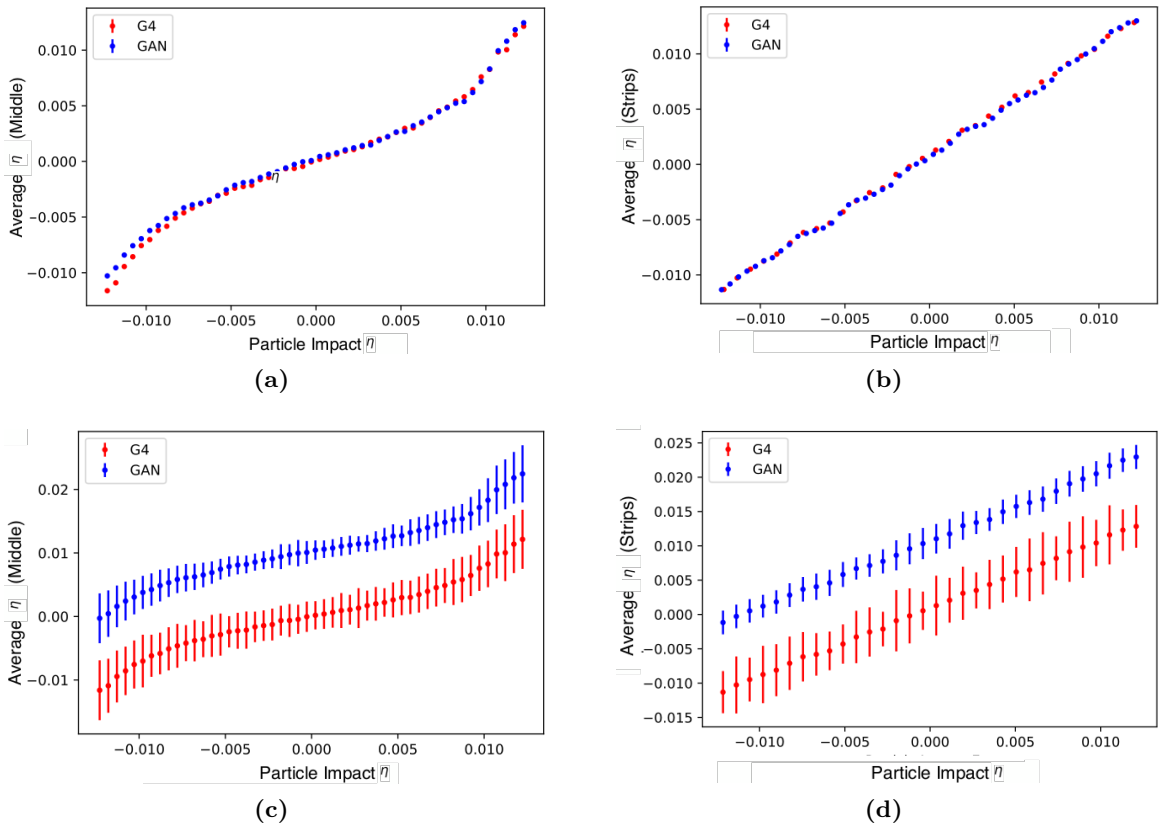


Figure 5.23 – Distribution of the average η vs extrapolated position of the particle in Middle ((a),(c)) and Strip layers((b),(d)) for Photons with a fixed energy of 65.5 GeV. Bars represent the statistical error in (a), (b) and the standard deviation in (c), (d) (where the GAN has been artificially shifted up for visibility. The x-axis range corresponds the width of 1 cell in the Middle layer and 8 cells in the Strip layer.)

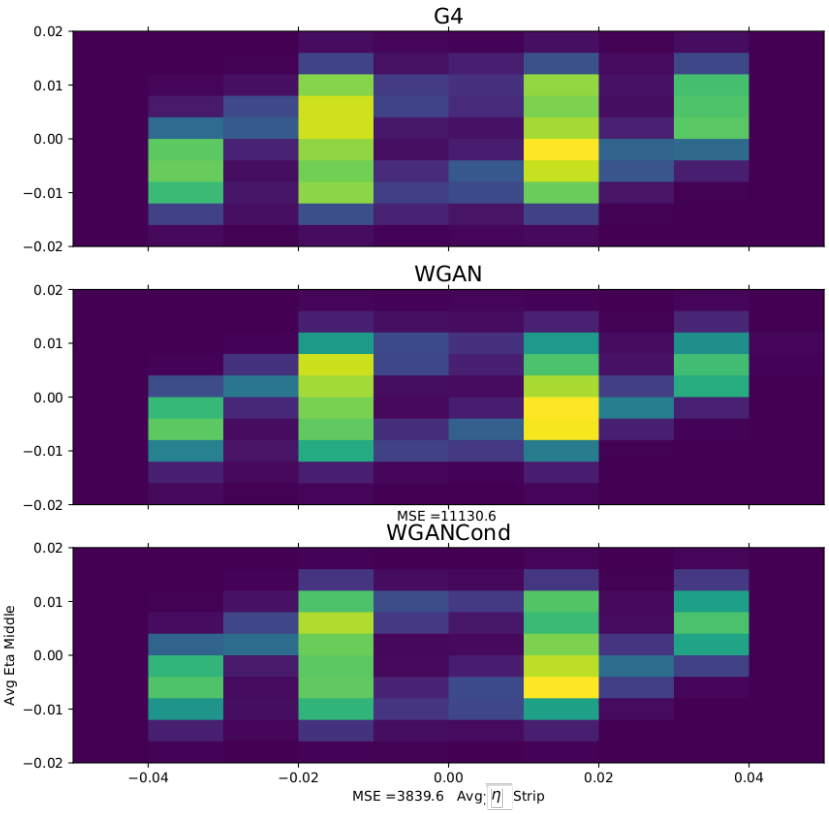


Figure 5.24 – 2D distribution of average η in the Middle layer vs Strip layer for (top) `Geant4`, (middle) GAN without conditioning to extrapolated position of the particle (with an MSE of 11131), (bottom) GAN conditioned on the extrapolated position of the particle (with a MSE of 3840). The MSE between `Geant4` and the GANs are also shown.

5.5.3.2 Lateral Distributions

The average η and ϕ distributions in the four layers are shown in Figure 5.29 and Figure 5.30 respectively and their widths (standard deviations) are shown in 5.31 and 5.32 respectively. The GAN matches `Geant4` reasonably well for most of the distributions, although less precisely for the Presampler and Back layers particularly for the widths. Less energy is deposited in these two layers compared to the Strips and the Middle layer, making it is harder for the GAN to learn these distributions.

The distribution of the difference between the average η and ϕ of the shower and the impact position of the particle for all four layers can be seen in Figure 5.33 and Figure 5.34 respectively. The GAN matches these distributions reasonably well, although it does not reproduce any of the unphysical single bin peaks that arise in the Presampler and Back layers for showers where no energy is deposited in these two layers. The GAN also cannot reproduce the "two horn" structure in the Strip layer which comes from an asymmetry between the two halves of the calorimeter in the training dataset. Since the GAN was trained under the assumption of a symmetry, it has no knowledge of which half it is simulating, and this is the cause of the difference between `Geant4` and the GAN distributions for this observable. These drawbacks of the dataset will be discussed at the end of this chapter.

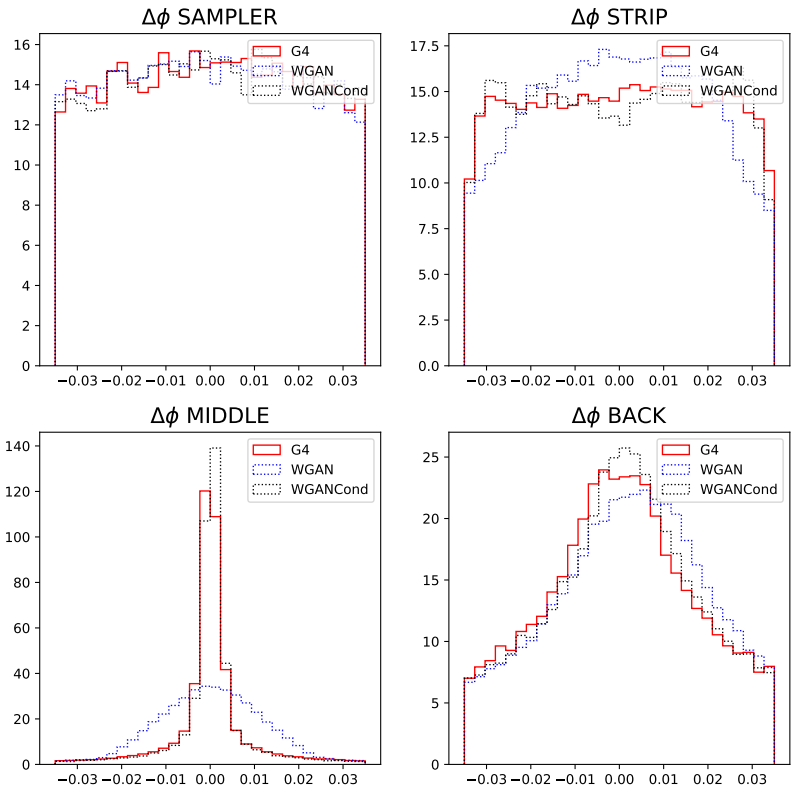
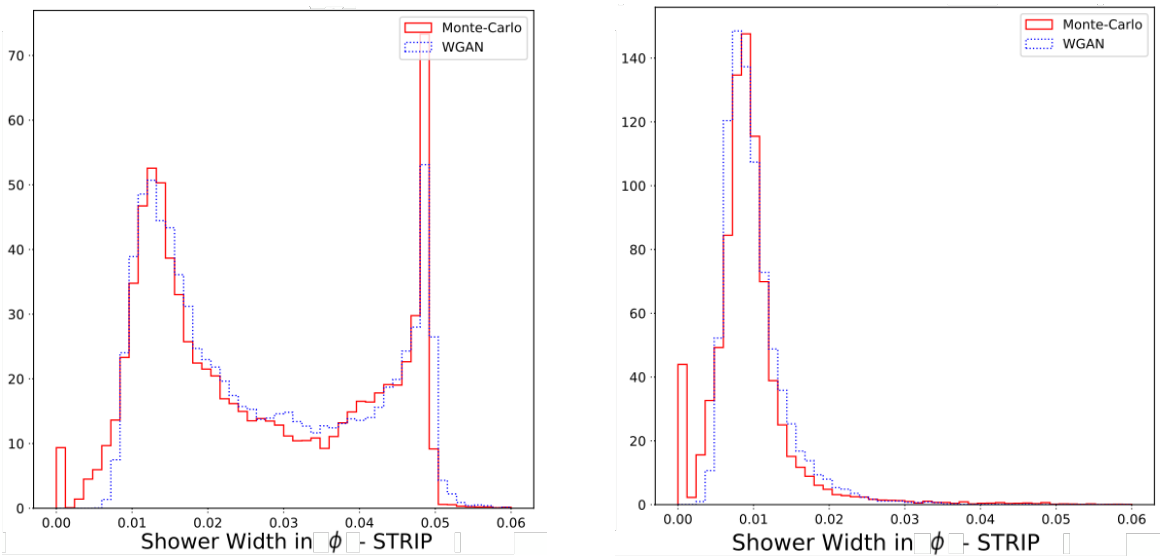
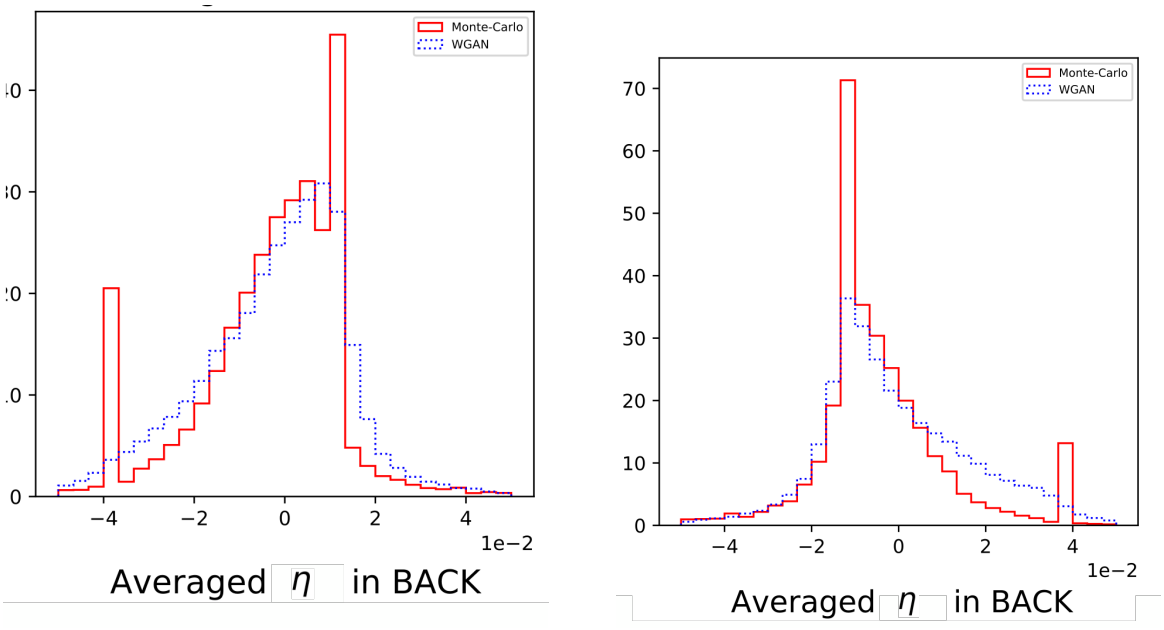


Figure 5.25 – A study of how electronic noise would affect the distributions of the $\Delta\phi$ distributions for an older GAN. Gaussian noise for each layer is added based on the mean expected noise for that layer. The artificial peaks seen near 0 in the `Geant4` distributions (which the GAN did not reproduce) are no longer present. The comparisons are made for `Geant4`, a GAN not conditioned on the particle position and a GAN conditioned on the particle position. To be compared with Figure 5.22.



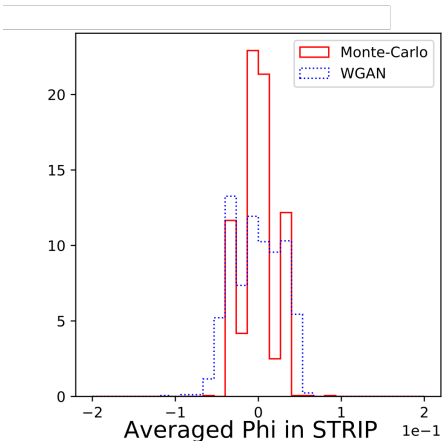
(a) **(b)**

Figure 5.26 – Distribution of the shower widths in the strip layer for sections of the calorimeter alignments p0: (a) and p1: (b), of the cells from the Strip layer with respect to the cells from the Middle layer.)



(a) **(b)**

Figure 5.27 – Distribution of the average η in the Back layer for sections of the calorimeter in the alignment e0: (a) and e1: (b), of the Back layer with respect to the Middle layer.)



(a)

Figure 5.28 – Example of an old GAN trained without detector geometry conditioning. The distribution of the average ϕ in the Strip layer is shown, which the GAN cannot match without geometry information.

To verify that the GAN also learns the lateral correlations between layers, a distribution of the difference between the average η in the Middle layer and each of the three other calorimeter layers is presented in Figure 5.35. The GAN learns the correlation between the Middle and Strip layer well but reproduces the correlation of the Middle layer with the Presampler with much reduced accuracy.

5.5.3.3 Energy Distributions

The new GAN architecture with two critics greatly improves the simulation of the energy response of the calorimeter as shown in Figure 5.13, particularly in contrast to the single critic GAN response seen in Figure 5.10. The total energy per layer distribution is shown in Figure 5.36 which shows reasonable agreement. There is a mis-modelling of the longitudinal shower shape as can be seen in the distribution of fraction of energy per layer in Figure 5.37. The GAN puts too much energy in Middle layer and too little in the Strip layer.

The first and second moments of average energy in each cell are shown in Figure 5.40, clear differences between the GAN and `Geant4` can be seen for the Pre-Sampler and Strip layers and smaller differences in the Middle and Back layers. The GAN does not reproduce the structures well, however, studies of standard-normalising each cell did not result in improved performance. In future studies, normalising the cells by the total energy per layer may be studied to improve the modelling of these distributions.

5.5.3.4 Distributions at Single Energy Points

It is important to verify that the GAN not only appears to learn the distributions for all energy points but that it can also reproduce distributions at fixed energy points. The shower shapes do vary as a function of the energy of the particle. Figure 5.38 shows the change in the fraction of energy deposited in the central 3×3 cells in the Middle layer over nine energy points. The GAN reproduces them reasonably well.

The difference between the average η of the Middle and Strip layers are shown for photons with 4 GeV and 262 GeV of energy in Figure 5.39. The distribution as a function of the energy is well modelled by the GAN.

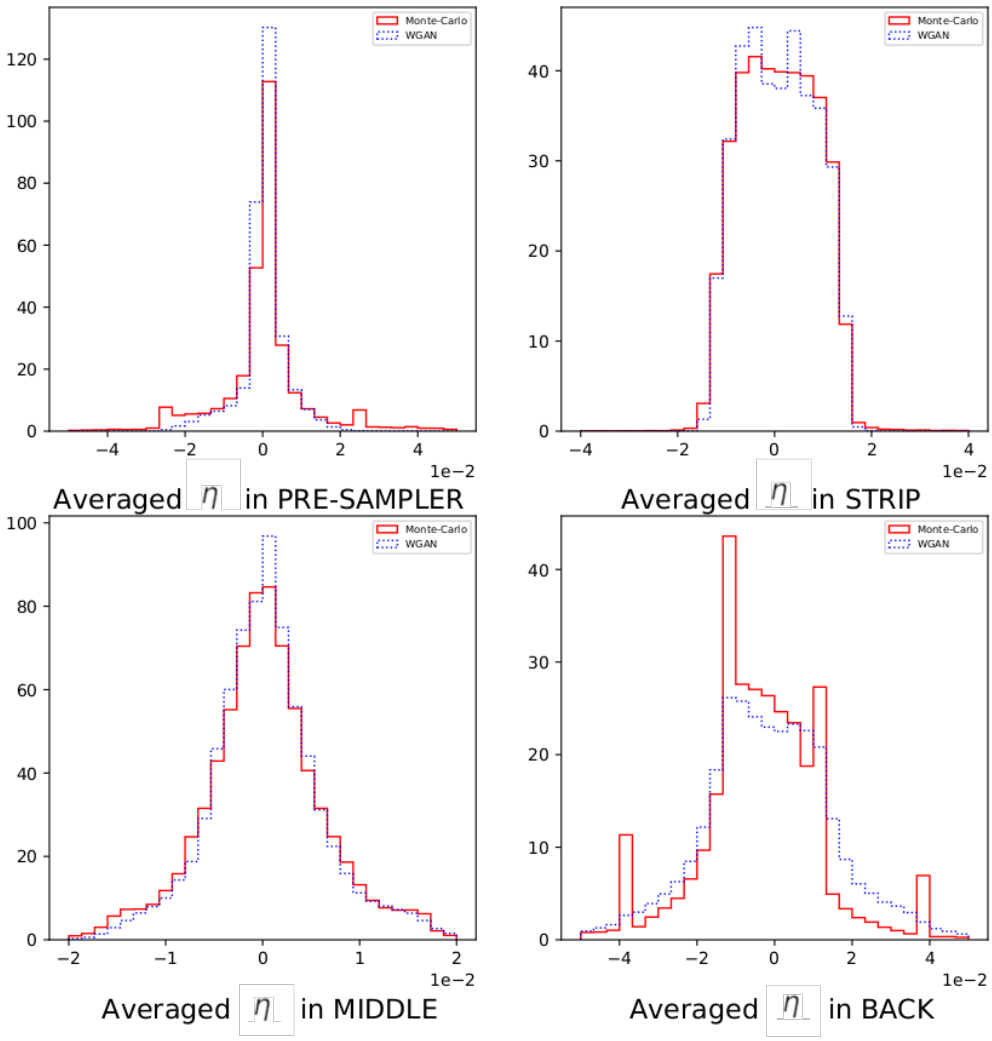


Figure 5.29 – Distribution of the average η in the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)

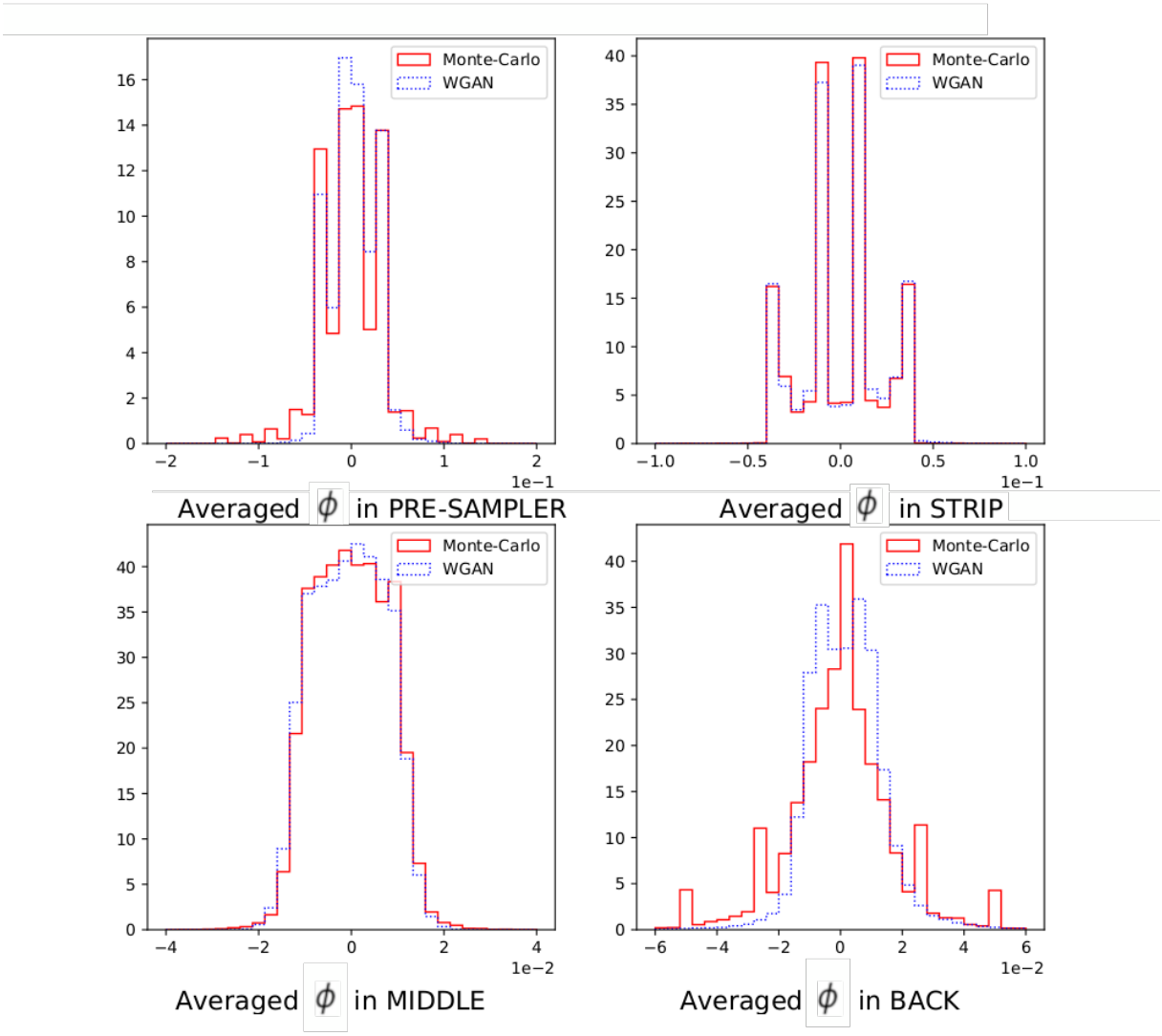


Figure 5.30 – Distribution of the average ϕ in the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)

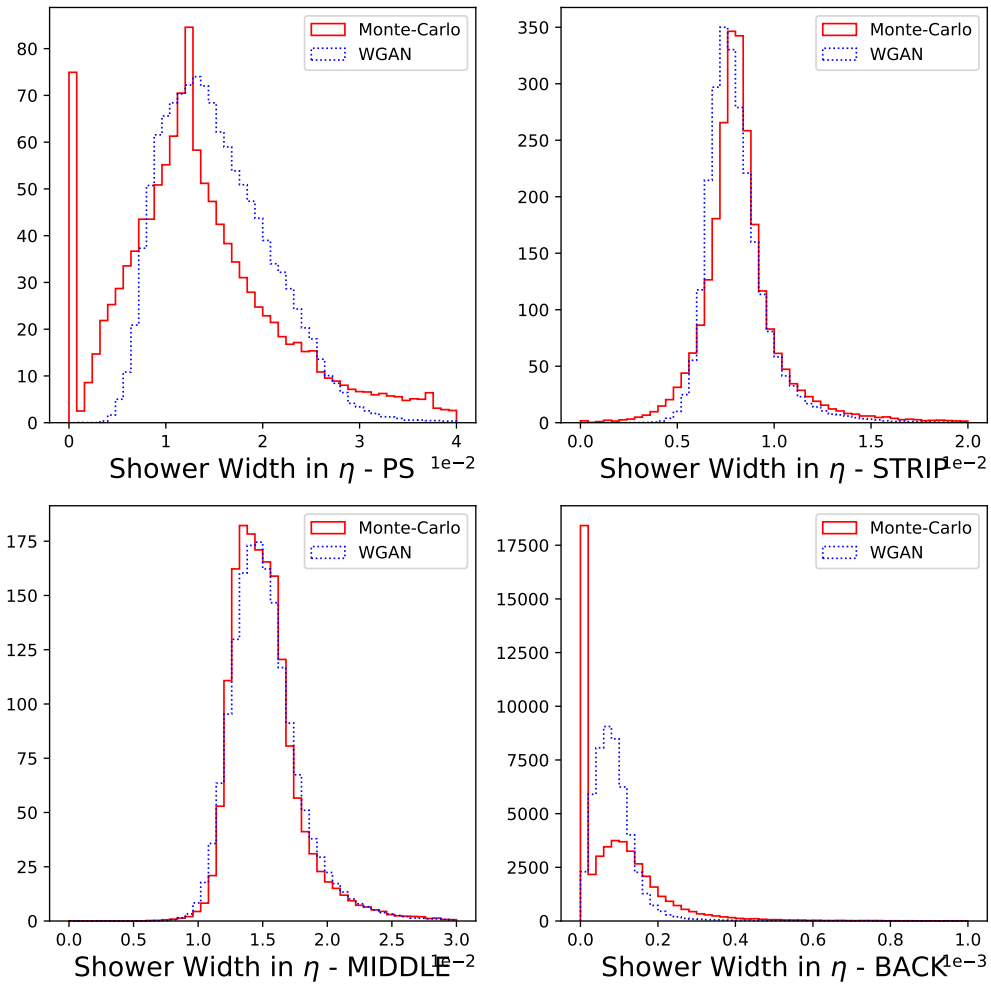


Figure 5.31 – Distribution of the width in η in the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)

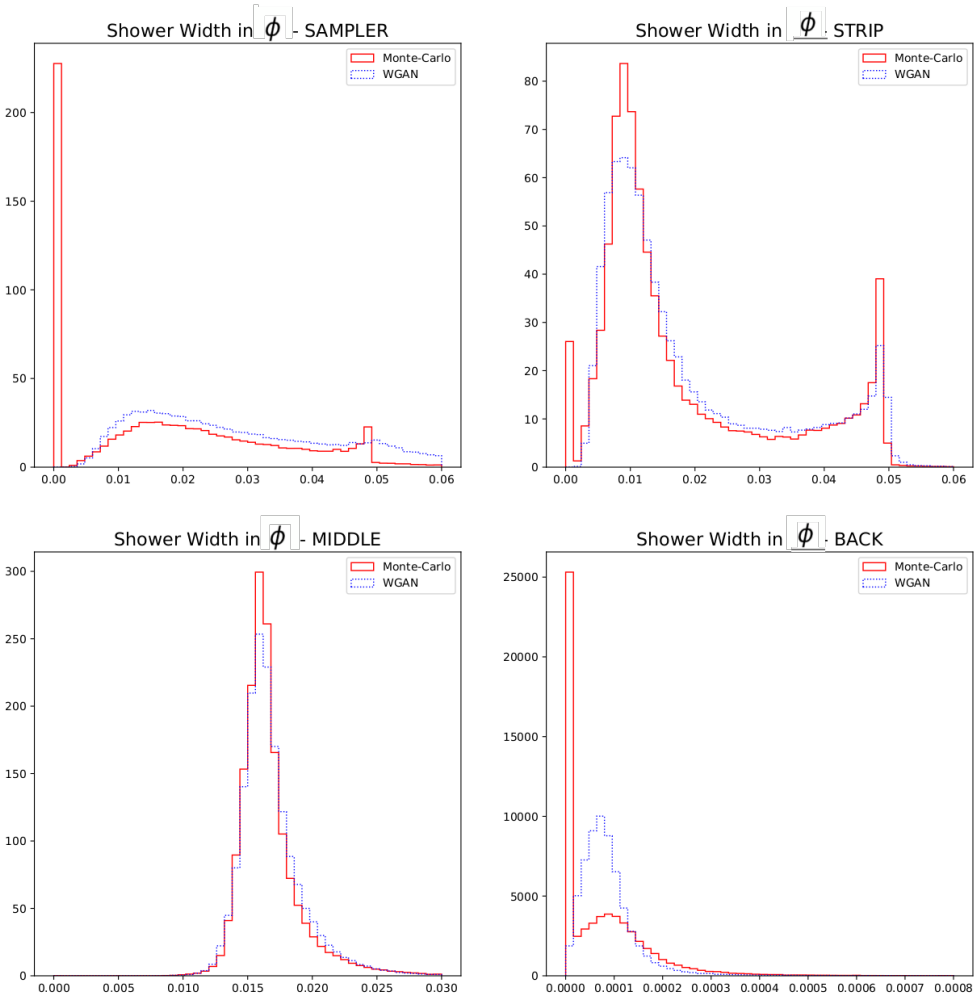


Figure 5.32 – Distribution of the width in ϕ in the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)

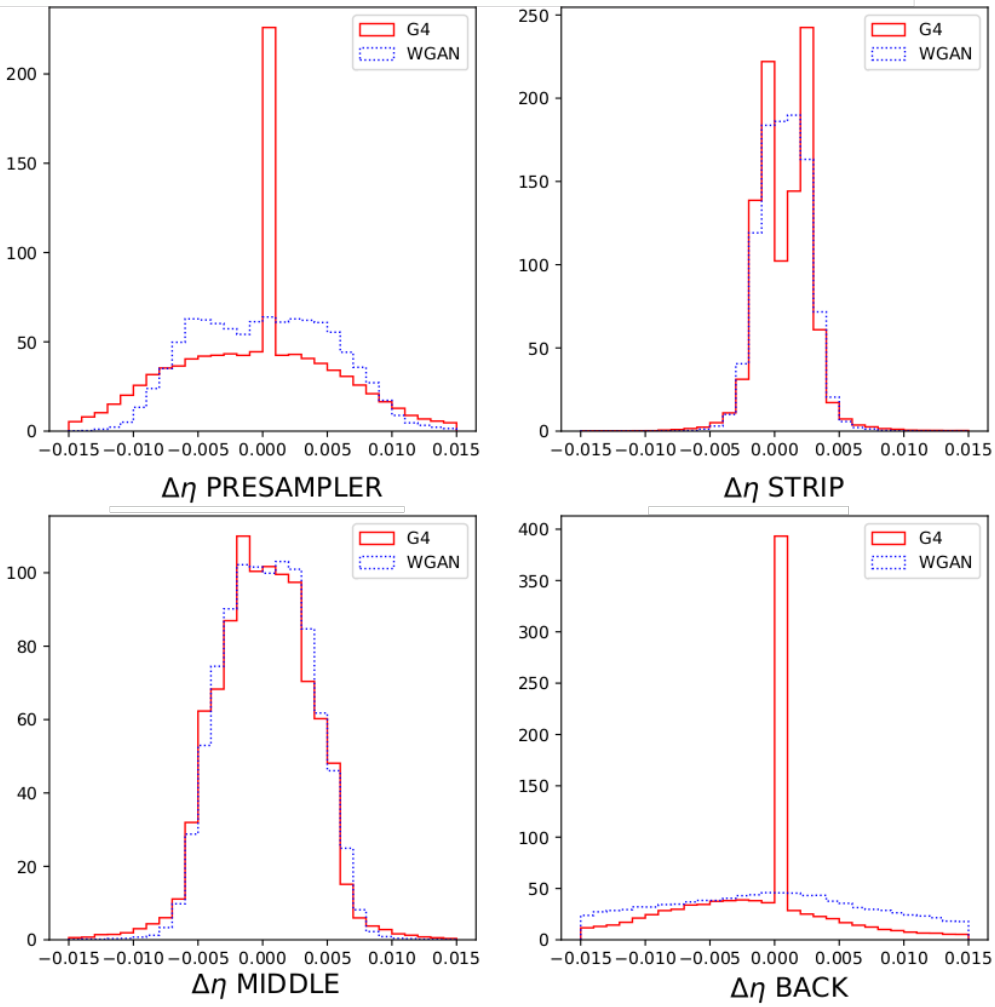


Figure 5.33 – Distribution of the difference between the average η of the shower and the impact η of the particle in the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)

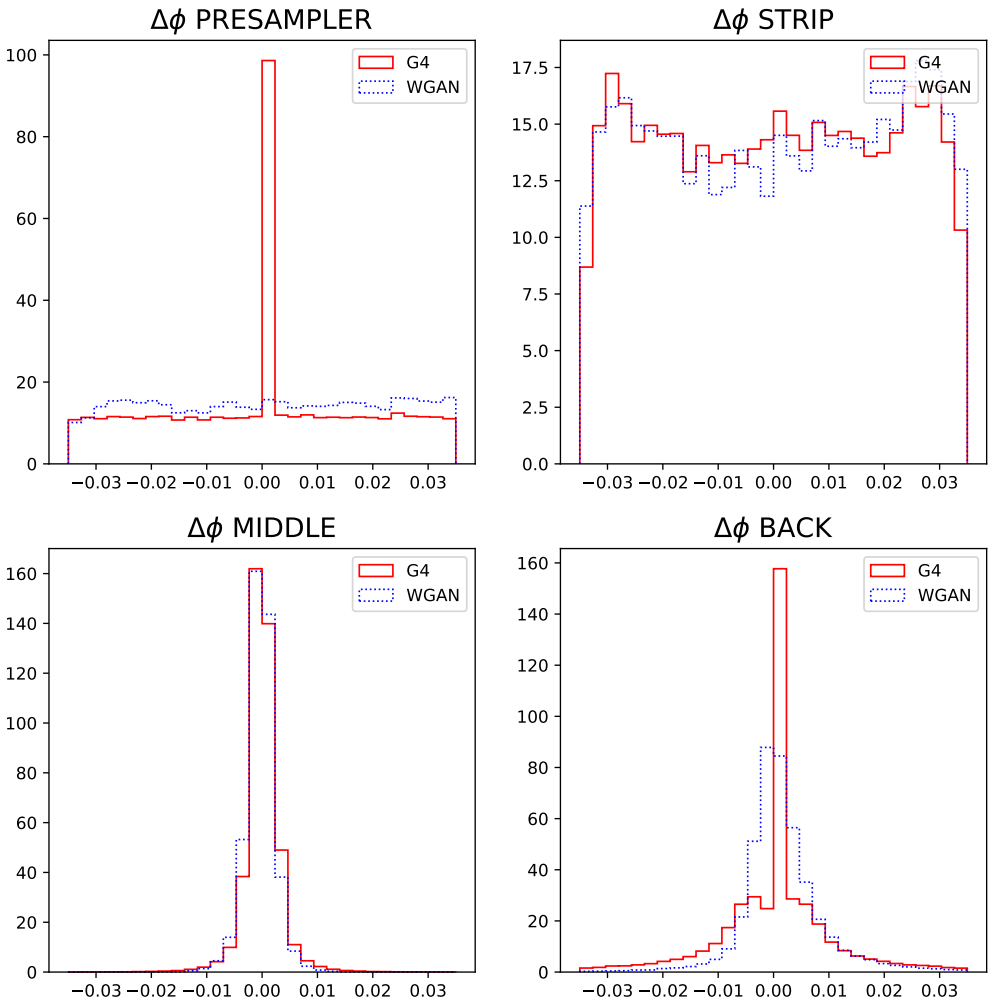


Figure 5.34 – Distribution of the difference between the average ϕ of the shower and the impact ϕ of the particle in the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)

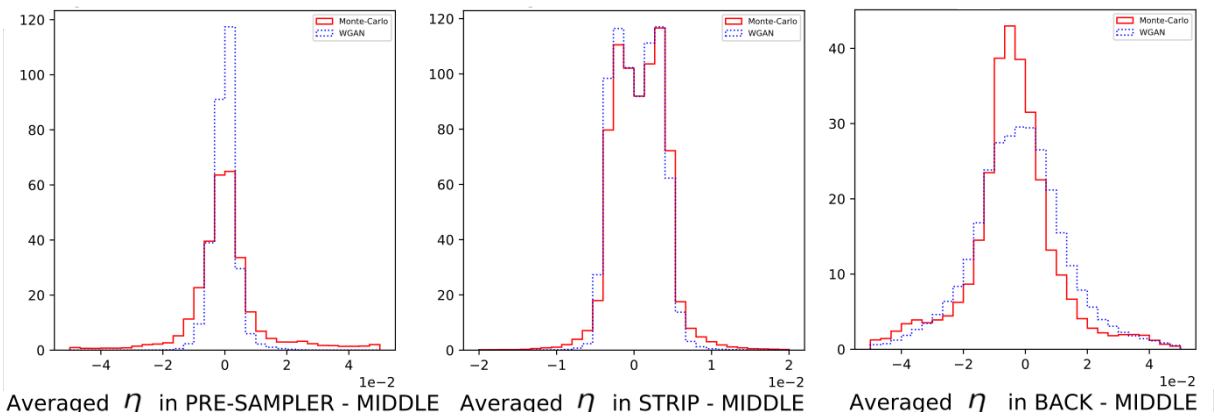


Figure 5.35 – Distribution of the difference between the average η of the three other layers with respect to the Middle layer for the GAN and Monte-Carlo (Geant4)

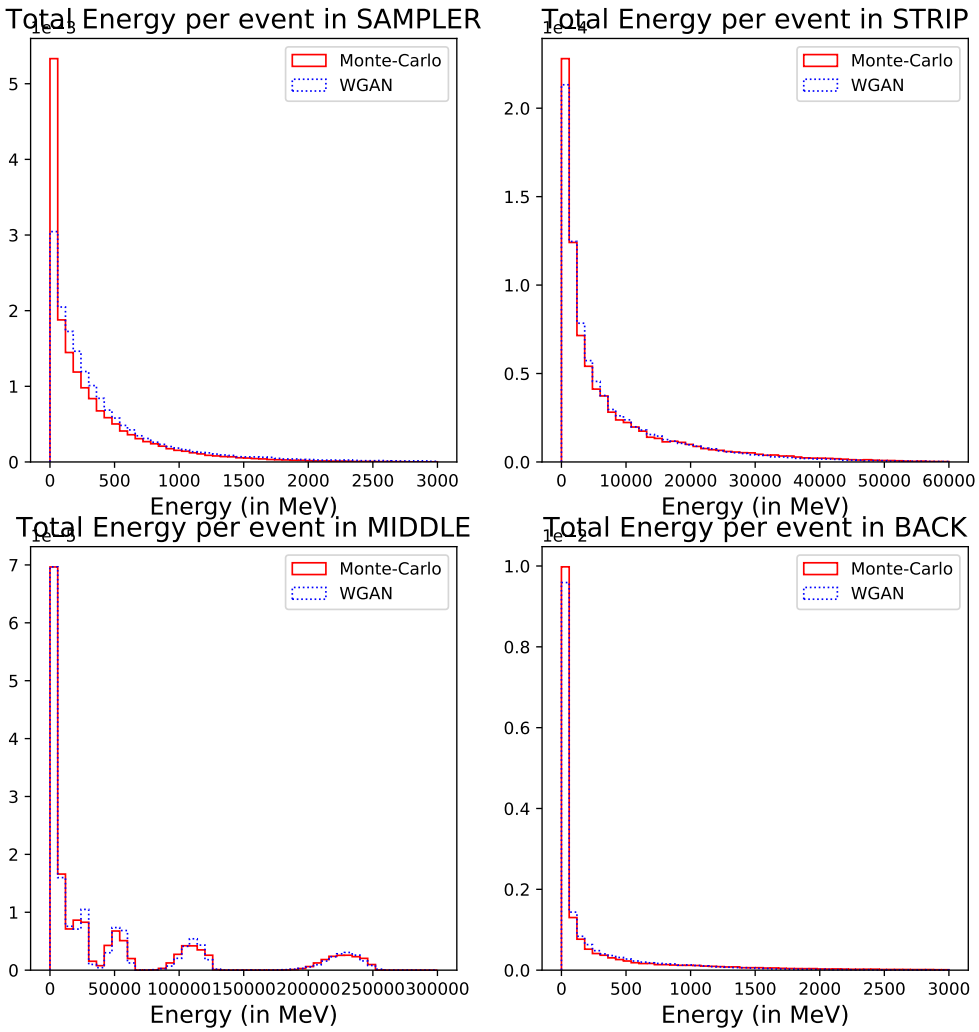


Figure 5.36 – Distribution of the total energy in each of the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)

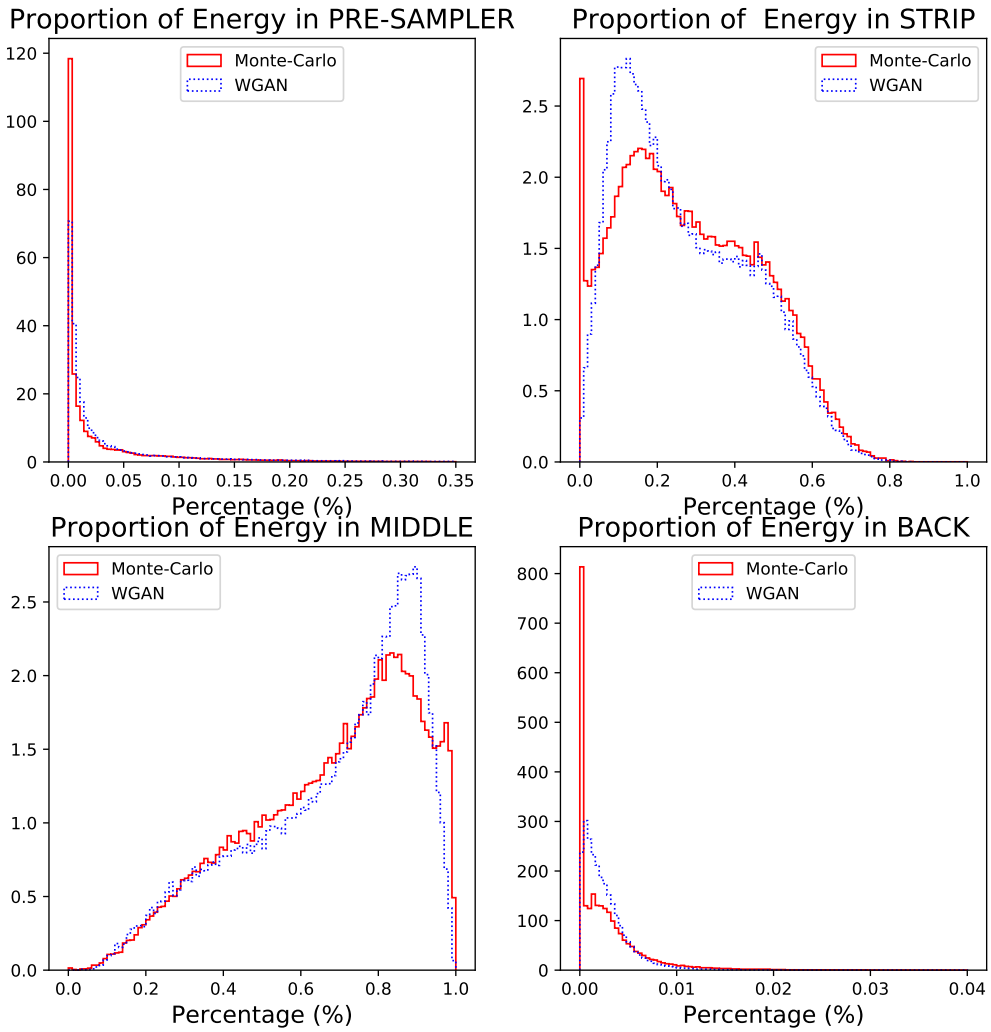
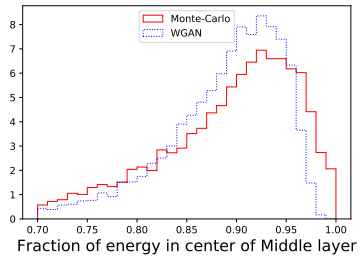
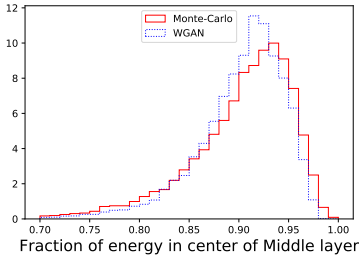


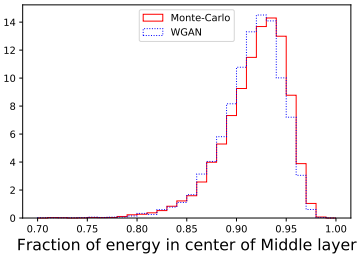
Figure 5.37 – Distribution of the fraction of energy in each of the 4 calorimeter layers for the GAN and Monte-Carlo (Geant4)



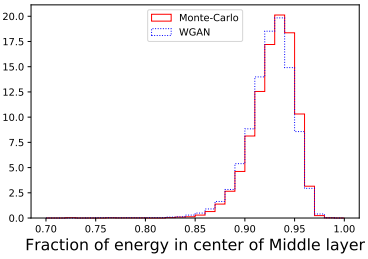
(a) 1 GeV



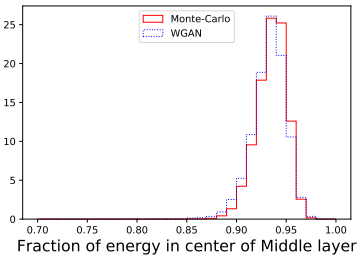
(b) 2 GeV



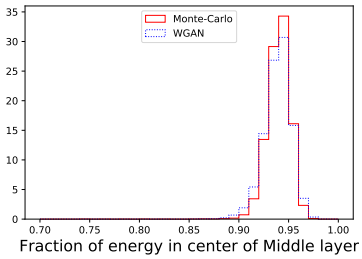
(c) 4 GeV



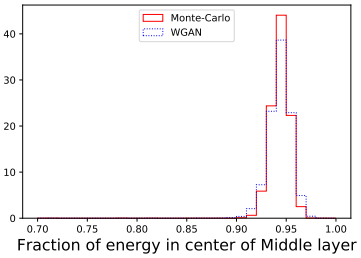
(d) 8 GeV



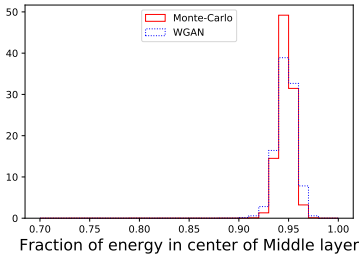
(e) 16 GeV



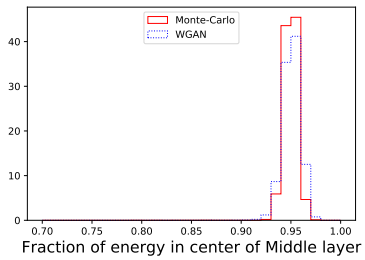
(f) 32 GeV



(g) 65.5 GeV



(h) 130 GeV



(i) 262 GeV

Figure 5.38 – Fraction of energy deposited in the centre of the Middle layer $E(3 \times 3)/E(7 \times 7)$.

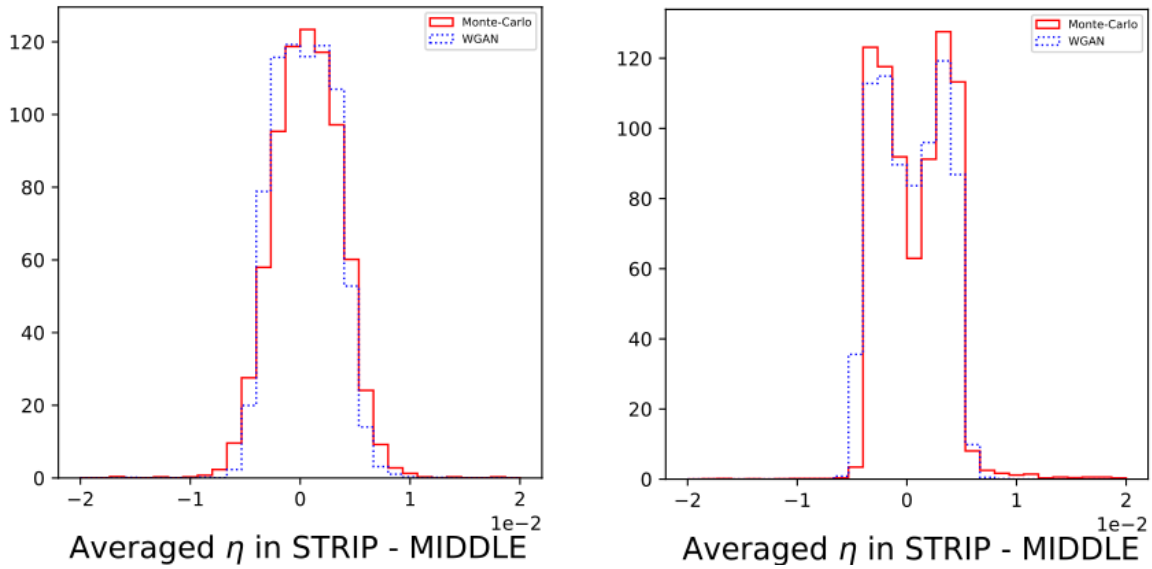


Figure 5.39 – Distribution of the difference in average η between the Middle and Strip layers for photons with (a) 4 GeV and (b) 262 GeV of energy.

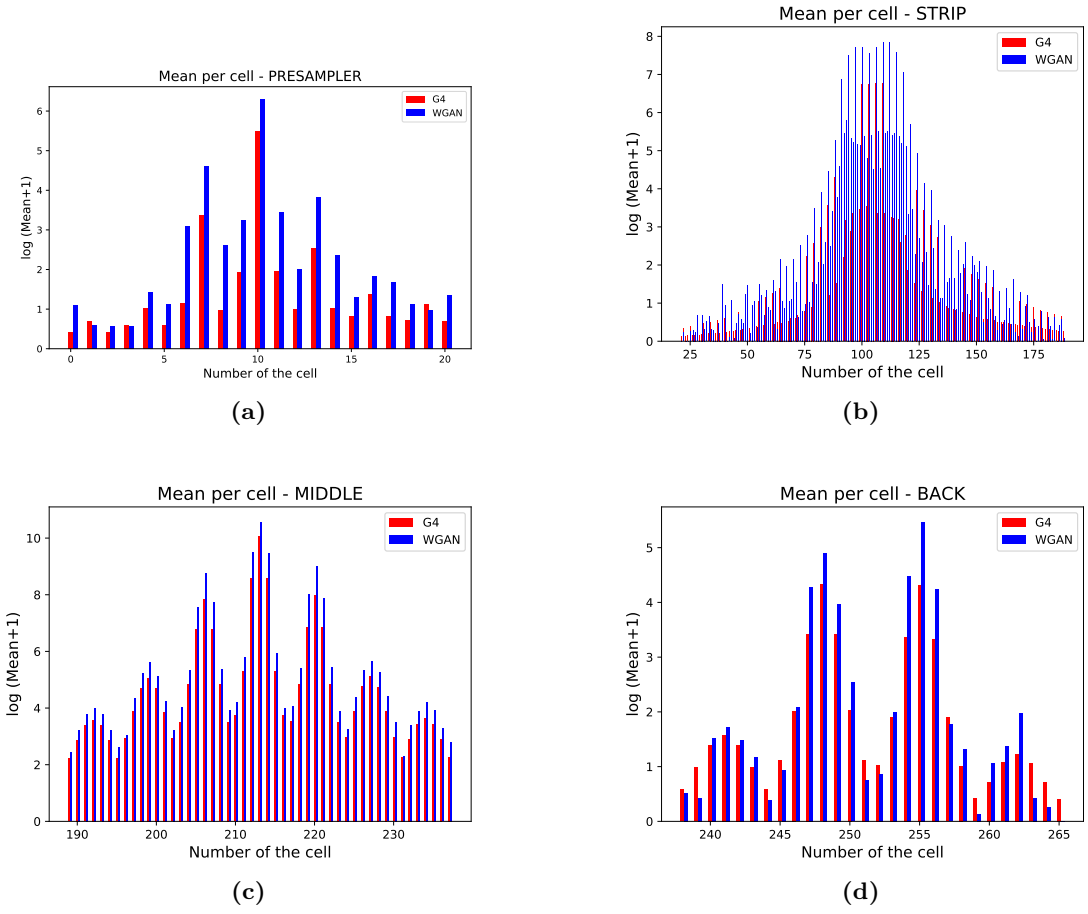


Figure 5.40 – Distribution of mean energy in each cell in the (a) Pre-Sampler, (b) Strips, (c) Middle and (d) Back layer.

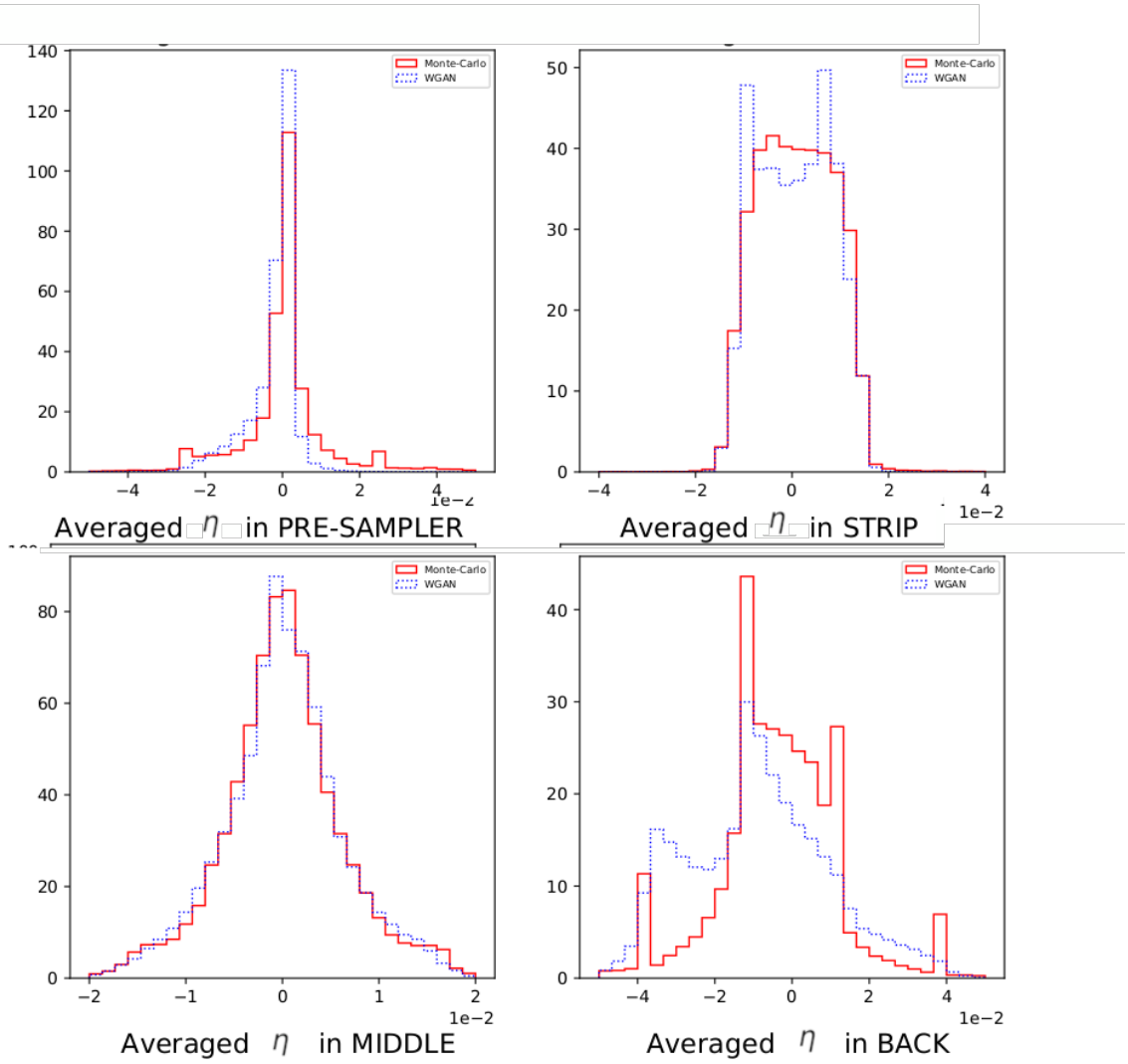


Figure 5.41 – Demonstration of the need for epoch picking: Distribution of the average η in the four layers for a GAN from epoch 12500 and Monte-Carlo (Geant4). This version of the generator was not integrated into Athena.

5.5.3.5 Importance of Epoch Picking

The distribution of the average η is shown in Figure 5.41 is shown from a different iteration of same GAN training as an example for the importance of epoch picking. This generator is from the epoch 12500 (as opposed to epoch 7500). The distributions for the generator have clearly deteriorated compared to Figure 5.29.

5.5.4 Validation Inside ATLAS Software

The validation of the GAN performance inside `Athena` was performed by simulating single photon showers at fixed energy points distributed in $0.20 < |\eta| < 0.25$ with realistic electronic noise but without displacements corresponding to the expected beam spread. Several high level physics motivated variables [102] were studied for the first time after integrating the GAN. The

comparisons are performed at various fixed values of incident photon energy.

5.5.4.1 Validation for 65.5 GeV Photons

The total calibrated energy is shown in Figure 5.42[8] which the GAN models well. The fraction of energy in the Strip and Back layers are shown in Figure 5.43. A slight mis-modelling is seen for the Strip layer which corresponds to the distributions in the standalone validation.

It should be noted that calibration is tuned only using `Geant4`, and an agreement between the distributions of a variable before calibration does not straightforwardly translate to an agreement between distributions of a variable after calibration. A divergence is possible if the calibration accounts for certain other factors which are not modelled well by the fast simulation technique.

The R_η is defined as the energy recorded in a 3×3 central region of the Middle layer over the energy recorded in a 7x7 region where the first number indicates the number of cells in the η direction and the second indicates the number of cells in the ϕ direction. Similarly R_ϕ is defined as the energy recorded in a 3×3 region over the energy recorded in a 3x7 region. The two distributions for 65.5 GeV photons are shown in Figure 5.44. The GAN does reasonably well for these distributions.

The fracs1 observable is defined for the Strip layer and it is the fraction of energy in a 7×1 over 3×1 region around the cell with the maximum energy. The Ecore observable is defined as $E0(3 \times 3) + E1(15 \times 2) + E2(5 \times 5) + E3(3 \times 5)$ where $El(m \times n)$ denotes the energy in layer l and the other two numbers denote the number of cells in the η and ϕ directions respectively. The distribution for these two observables is shown in Figure 5.45.

The $w_{\text{s tot}}$ observable is defined as the total lateral shower width, $\sqrt{\left(\Sigma E_i \left(i - i_{\text{max}}\right)^2\right) / \left(\Sigma E_i\right)}$, where i runs over all cells in a window of $\Delta\eta = 0.0625$ and i_{max} is the index of the highest-energy cell [102]. The $w_{\eta 1}$ observable is defined as the shower width using ± 3 strip cells around the one with the maximal energy deposit, $\sqrt{\sum(Ei)x(i - i_{\text{max}})^2 / \sum(Ei)}$, where i is the number of the strip cell and i_{max} the number of the strip cell with the maximum energy deposit. The distributions for $w_{\text{s tot1}}$, which is the $w_{\text{s tot}}$ for the strip layer, and $w_{\eta 1}$ are shown in Figure 5.46. The GAN reproduces the first reasonably well and is wider than `Geant4` for the second.

The emins_1 observable is defined as the energy reconstructed in a strip cell with the minimal value between the first and the second maximum. The emax_{s1} observable is defined as the energy of strip cell with maximal energy deposit. The e2tsts_1 observable is the second maximum in strip layer calculated by summing three strip cells. These three observables are shown in Figure 5.47.

5.5.4.2 Interpolation at Untrained Parameter Point

Since the GAN was trained on only 9 photon energies, it is essential to verify that the GAN is able to interpolate well at untrained energy points. Figure 5.48 and Figure 5.49 show that the GAN is able to interpolate to an untrained energy point of 25 GeV for distributions that change as a function of the energy of the incident particle, namely the distribution of the total calibrated energy and the R_ϕ. The same level of performance is seen as for neighbouring energy points used to train the models.

[8] The VAE studies are not discussed in this thesis. The VAE distributions shown were found to be produced with a different reconstruction setting and will be updated in the future.

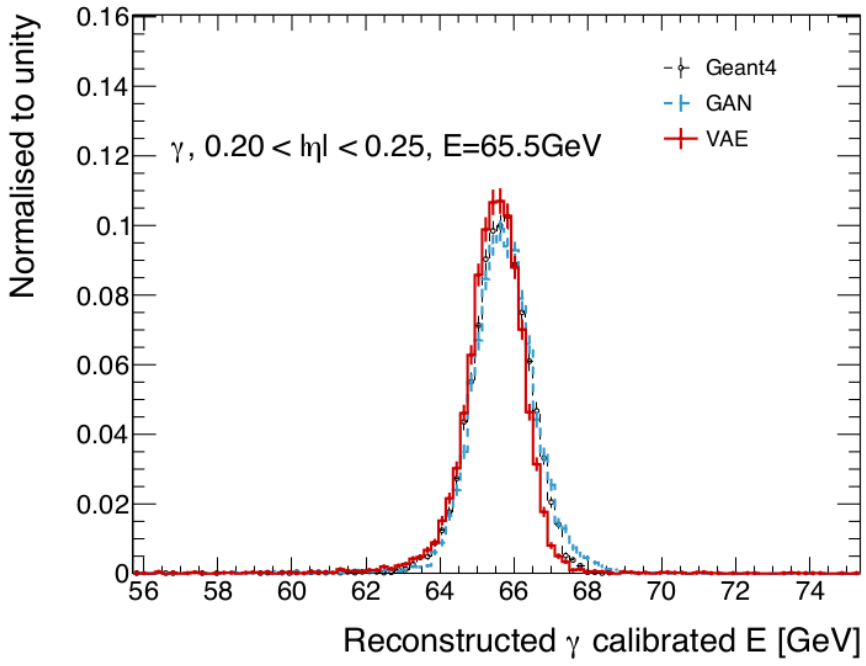
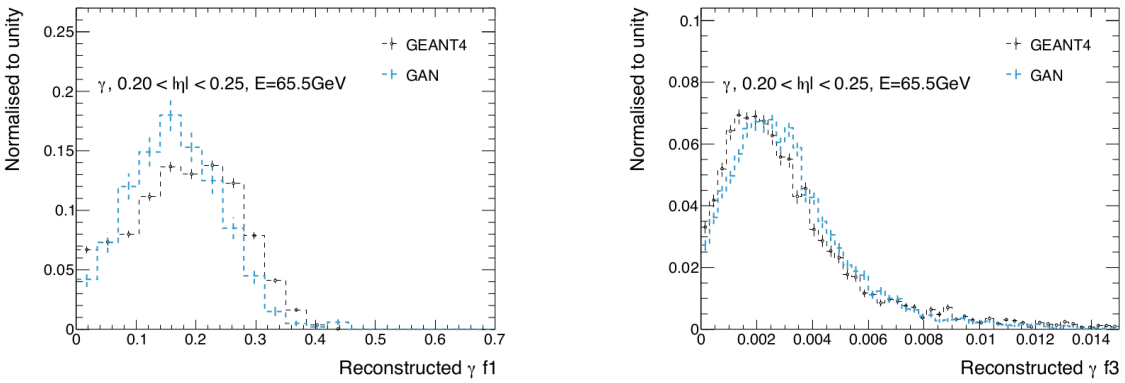
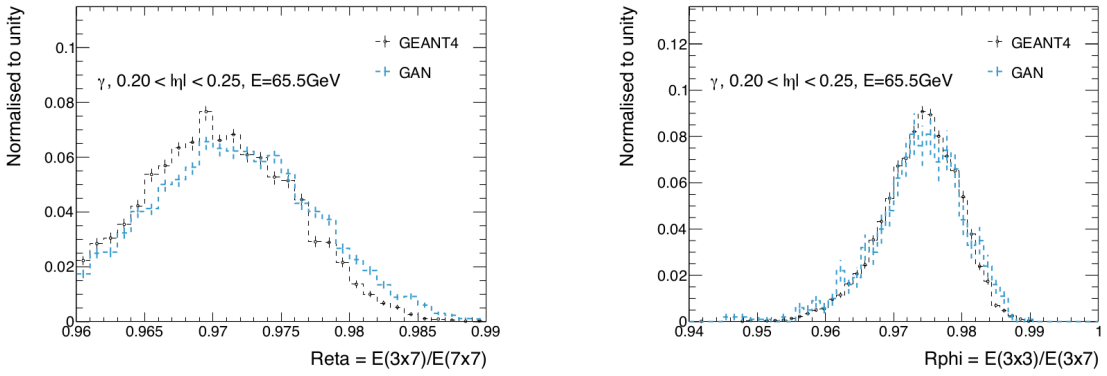


Figure 5.42 – Distribution of the total calibrated energy the GAN and Geant4



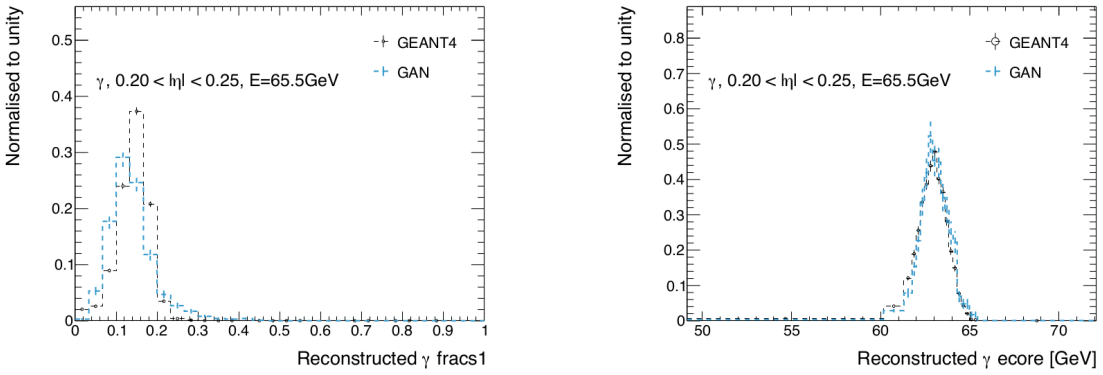
(a) **(b)**

Figure 5.43 – Fraction of energy in (a) the Strip and (b) Back layers of the calorimeter.)



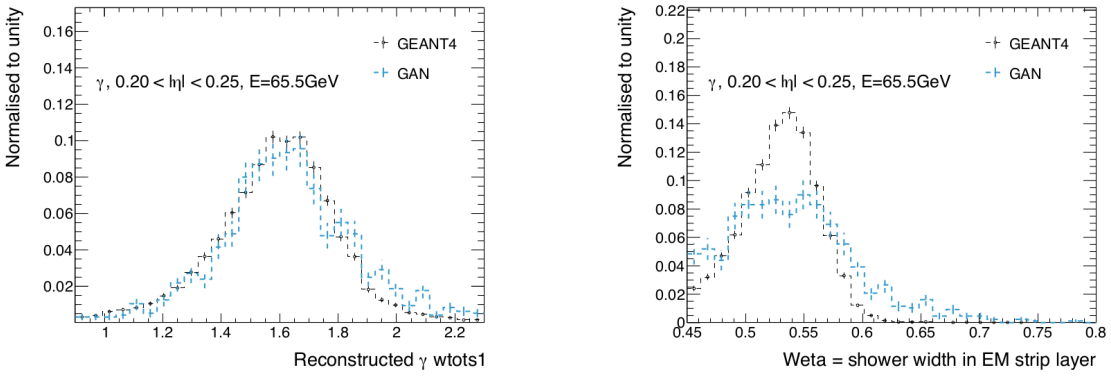
(a) **(b)**

Figure 5.44 – Distribution of (a) Reta and (b) Rphi for 65.5 GeV photons.



(a) **(b)**

Figure 5.45 – Distribution of (a) fracs1 and (b) ecore for 65.5 GeV photons.



(a) **(b)**

Figure 5.46 – Distribution of (a) $w_{\mathrm{s\ tot1}}$ and (b) $w_{\eta 1}$ for 65.5 GeV photons.)

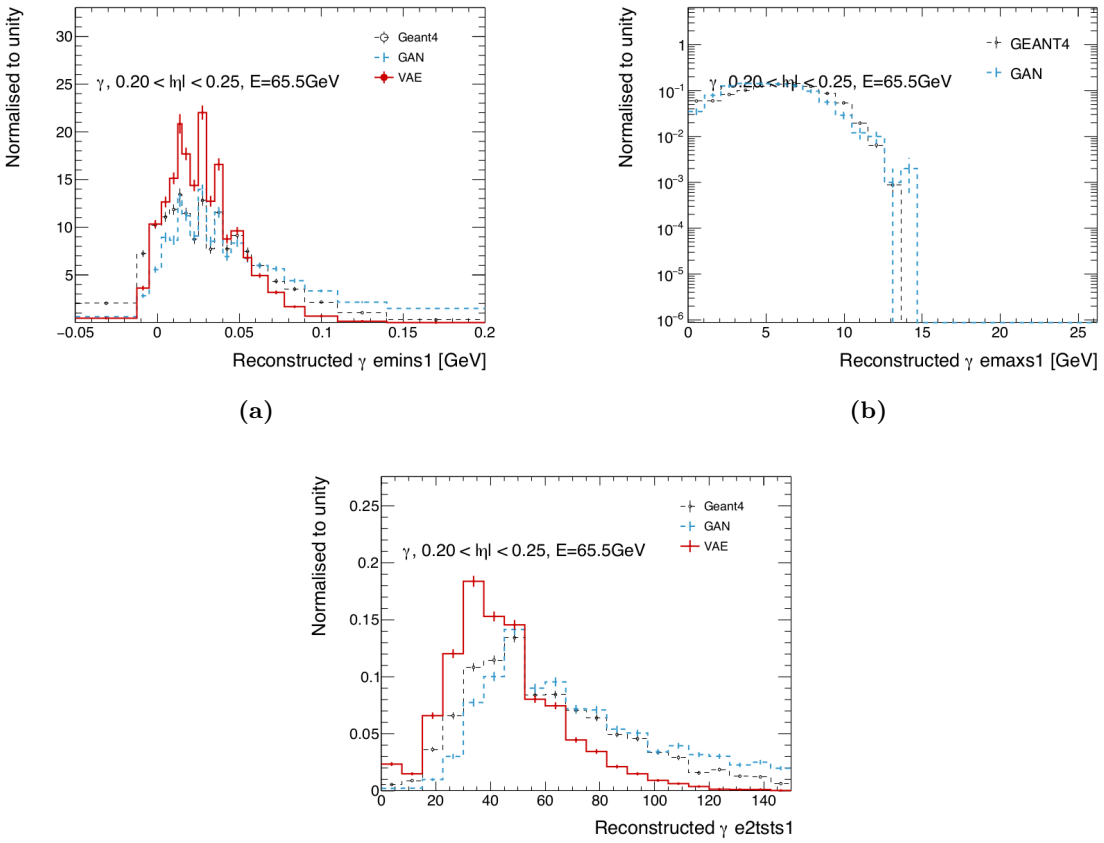


Figure 5.47 – Distribution of (a) emins_1, (b) emax_{s1} and (c) e2tsts_1 for 65.5 GeV photons.

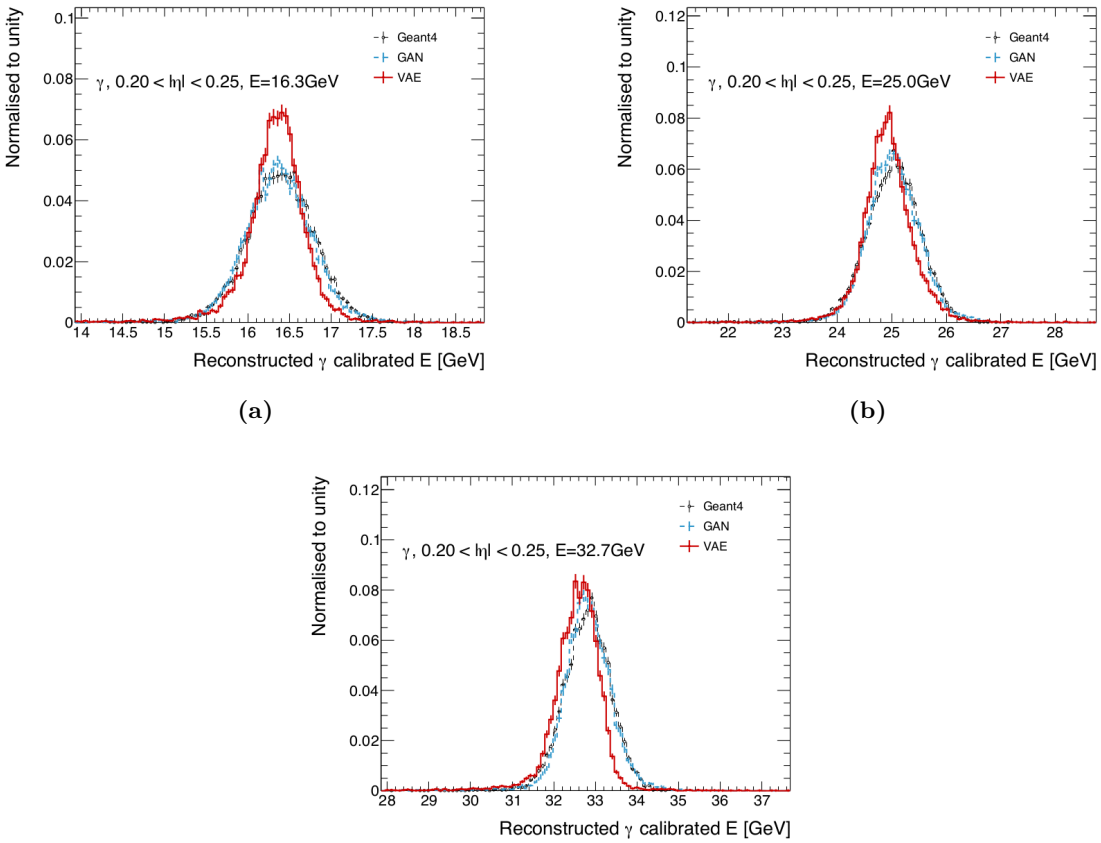


Figure 5.48 – Distribution of calibrated energy for (a) 16 GeV, (b) 25 GeV and (c) 33 GeV photons.

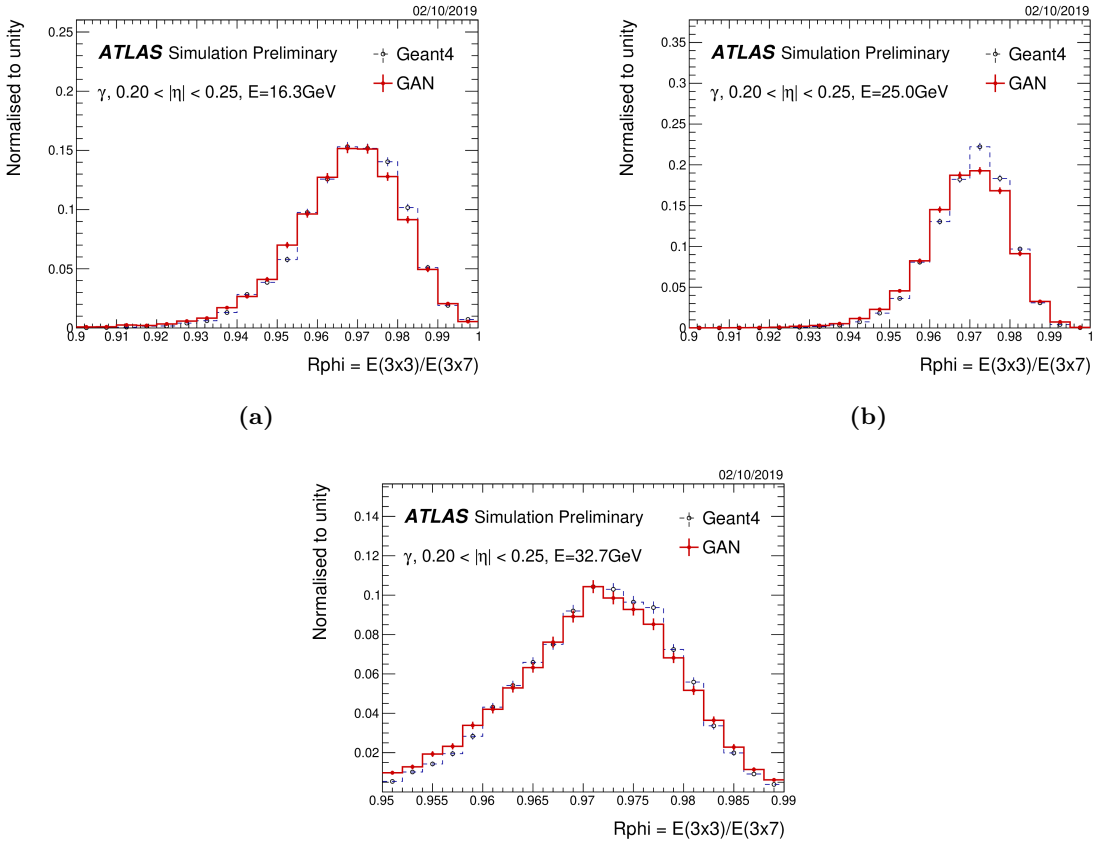
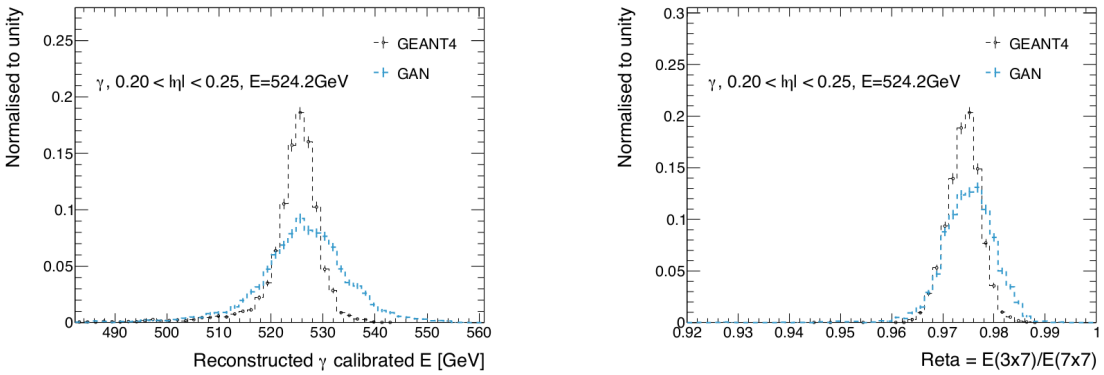


Figure 5.49 – Distribution of R_ϕ for (a) 16 GeV, (b) 25 GeV and (c) 33 GeV photons. The GAN is shown in red and Geant4 in black.



(a) **(b)**

Figure 5.50 – Extrapolation Test: Distribution of (a) total calibrated energy, (b) R_η for 512 GeV photons. The GAN was never trained on any photons with energy beyond 262 GeV.

5.5.4.3 Extrapolation

Feed-forward Neural networks are not known to extrapolate well to unseen points in the input space. A test is performed on how much the performance of the GAN deteriorates for novel truth energies outside the training range.

The chosen energy point is 512 GeV, a factor of two larger than the highest training energy point. Figure 5.50 shows that the GAN produces the correct mean of the total energy but fails to model the width of the distribution correctly. It also failed to model the R_η distribution very well.

5.5.4.4 Comparison to Atlas Fast II

The performance of the GAN is compared to the current fast simulation software `ATLAS Fast II`. Since `Geant4` is used at the ideal distribution the data tuning for `ATLAS Fast II` is turned off, however comparisons are also shown with data tuning turned on.

The E_{ratio} observable is defined as the ratio of the energy difference between the maximum energy deposit and the energy deposit in a secondary maximum in the cluster to the sum of these energies, $E_{\text{ratio}} = (\text{emaxs}_1 - \text{e2tsts}_1) \,/\, (\text{emaxs}_1 + \text{e2tsts}_1)$.

Figure 5.51 shows comparisons of the fraction of the energy deposited in the Strip and Back layers, the R_ϕ, $w_{\eta 1}$ and E_{ratio} between `Geant4`, `ATLAS Fast II` the GAN for 25 GeV photons. For comparisons to `ATLAS Fast II` with data tuning on see Figure 5.52. The performance of the GAN is comparable (although sometimes slightly worse) to that of `ATLAS Fast II` for these distributions, and in certain cases, such as the energy fraction in the Back layer and R_ϕ, it matches `Geant4` better than `ATLAS Fast II`. Since the different algorithms perform better for different distributions neither can be be considered consistently better than the other. However it is encouraging to note that the performance of the GAN is already comparable to the traditional algorithm that took considerable effort and expertise to optimise.

5.5.4.5 Comparison to FastCaloSimV2

Comparisons are also made to the in-development `FastCaloSimV2`. It is to be noted that since `FastCaloSimV2` is in high priority development, there are frequent new versions. The compar-

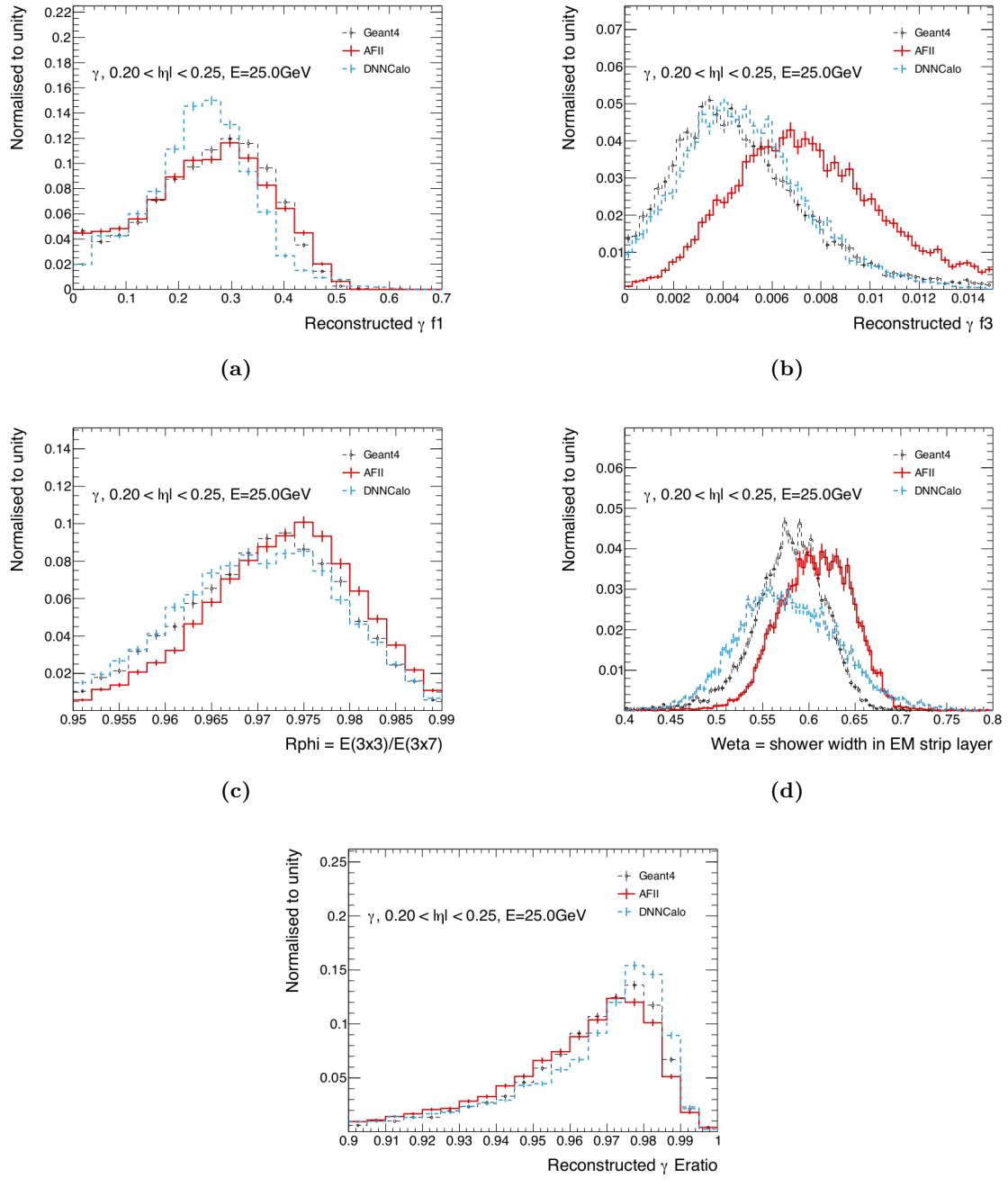
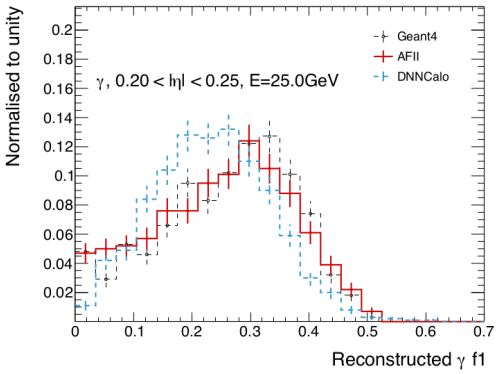
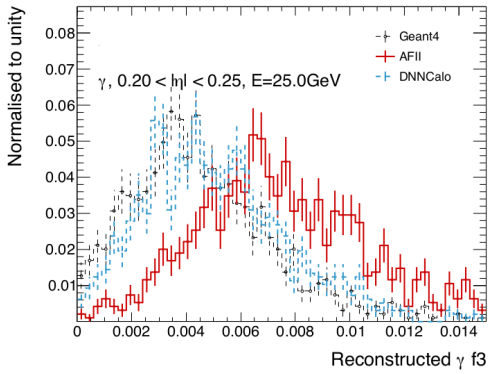


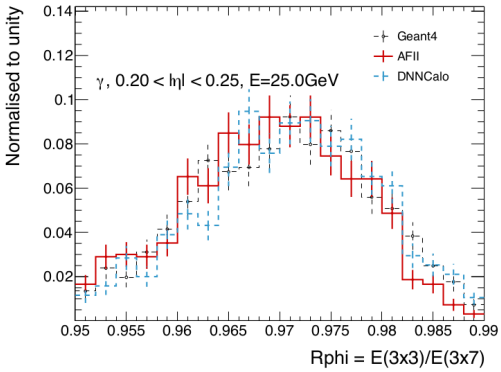
Figure 5.51 – Comparisons between Geant4, ATLAS Fast II (without data tuning) and GAN for fraction of energy in the (a) Strip layer, (b) Back layer and the observables (c) R_ϕ, (d) $w_{\eta 1}$ and (e) E_{ratio} for 25 GeV photons.



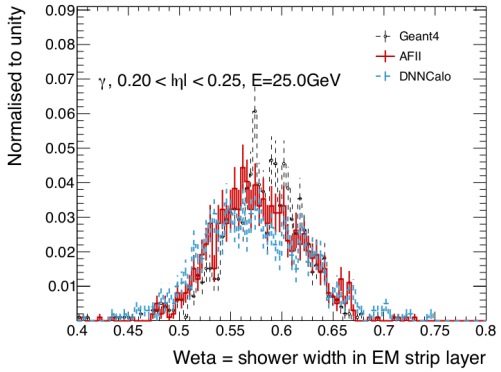
(a)



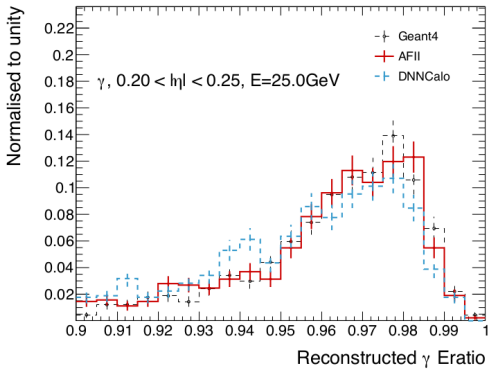
(b)



(c)



(d)



(e)

Figure 5.52 – Comparisons between Geant4, ATLAS Fast II with data tuning and the GAN for fraction of energy in the (a) Strip layer, (b) Back layer and the observables (c) R_{ϕ}, (d) $w_{\eta 1}$ and (e) $E_{textratio}$ for 25 GeV photons.

isons shown here were produced with the latest versions of FastCaloSimV2 at that time each comparison was made, but these versions of FastCaloSimV2 have since been superseded by newer versions. These comparisons are only meant to put the performance of the GAN in an understandable context, rather than to rank different fast simulation methods at the this time.

These comparisons are shown with version 9 of FastCaloSimV2 in Figure 5.53 and with version 11 of FastCaloSimV2 in Figure 5.54. Certain differences seen between the Geant4 in these figures compared to figures from previous sections is due to slight differences in the reconstruction specifications, but for each comparison the exact same reconstruction specifications are used the Geant4 the GAN and the other fast simulation techniques.

The performance of the GAN is again comparable to FastCaloSimV2[9] but clearly a worse modelling of the energy fraction in the Strip layer. The updates to FastCaloSimV2 in version 11 help it close the gap between the performances of the GAN and FastCaloSimV2 seen in the original comparison using version 9 of FastCaloSimV2. Further work in optimising a GAN based on these comparisons should allow to further improve the performance of a neural network based simulation.

5.5.4.6 Deterioration of performance at High Energy

The GAN performs significantly worse for certain distributions at higher energy points (beyond a photon energy of 100 GeV), as can be seen in Figure 5.55, which shows the distribution of R_η for photons with energies of 2 GeV, 8 GeV, 32 GeV and 130 GeV.

5.5.4.7 Deterioration of Impact Conditioning in Athena

Despite performing well in standalone validation, the GAN fails to simulate the correlation between the position of the particle and the average η of the shower inside Athena, as can be seen in Figure 5.56. It also fails to exhibit a reasonable correlation between the Strip and Middle layer as seen in Figure 5.57. These issues were understood to be a problem in the training dataset, described in Section 5.6.

5.5.5 Software Performance

The time required to simulate single photon showers grows with the energy of the particle for Geant4. Measured in February 2019, on the same machine and running through Athena, Geant4 takes 10 seconds per shower for a 65.5 GeV photon and 2.4 seconds for a 16 GeV photon, compared to that the GAN takes 70 milliseconds per shower irrespective of the photon energy. This is comparable the traditional parameterised fast simulation approach. In both cases, most of this time is due to overhead that will be optimised. The actual shower generation takes 8 milliseconds out of which 7 milliseconds is spent on building the cluster and only 0.7 milliseconds needed to run the neural network. While there is still a lot of scope to optimise the speed, the current speed is sufficient for the fast simulation needs of the ATLAS experiment.

The simulation time for single showers as a function of the photon energy is shown in Figure 5.58. The simulations using Geant4 were measured on three machines, referred to as 'Machine4', 'Machine5', and 'Machine6', and the differences between the times for each machine are negligible when comparing Geant4 with the DNNCaloSim.

The LWTNN JSON file size is 9.6 MB on the disk, small compared to $\mathcal{O}$(GBs) for FastCaloSimV2 and also has a far smaller peak memory usage of 2.3 GB (with the LWTNN taking only 5 MB)

[9] An even newer FastCaloSimV2 parameterisation shows an improved R_ϕ modelling

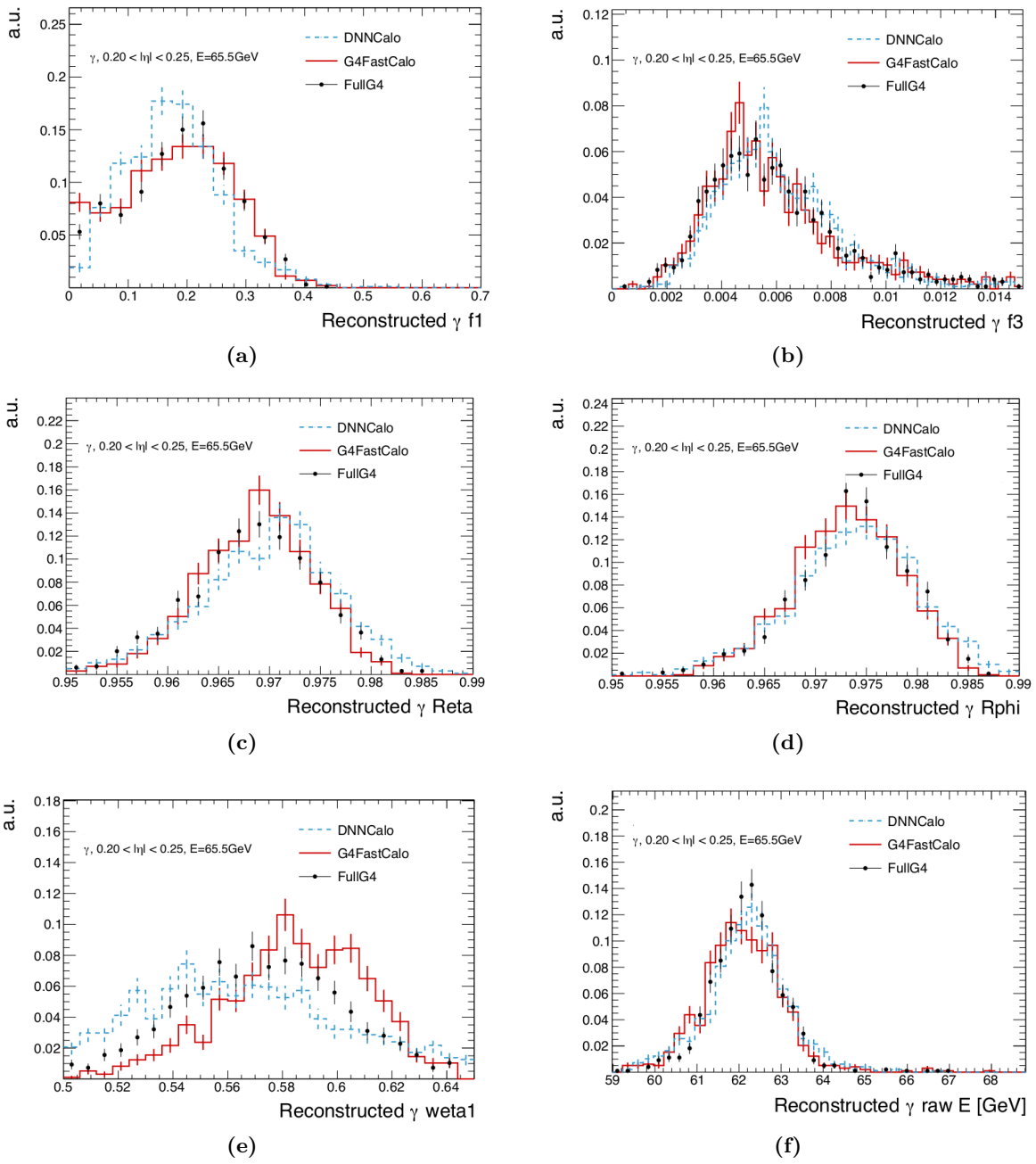


Figure 5.53 – Comparisons between Geant4, FastCaloSimV2 (version 9) and GAN for fraction of energy in the (a) Strip layer, (b) Back layer and the observables (c) R_η, (d) R_ϕ, (e) $w_{\eta 1}$ and (f) total uncalibrated energy for 65.5 GeV photons.

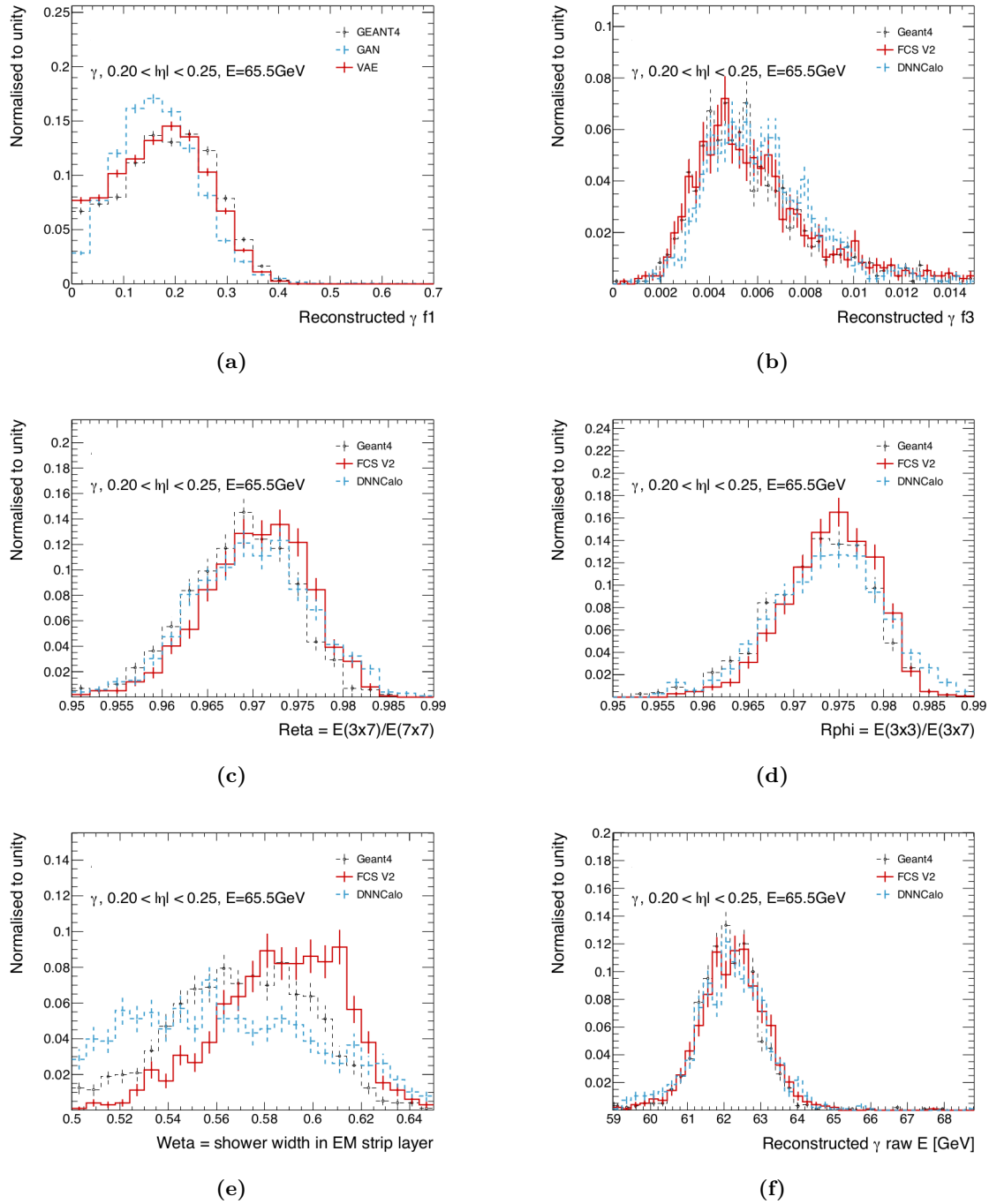


Figure 5.54 – Comparisons between Geant4, FastCaloSimV2 (version 11) and GAN for fraction of energy in the (a) Strip layer, (b) Back layer and the observables (c) R_η, (d) R_ϕ, (e) $w_{\eta 1}$ and (f) total uncalibrated energy for 65.5 GeV photons.

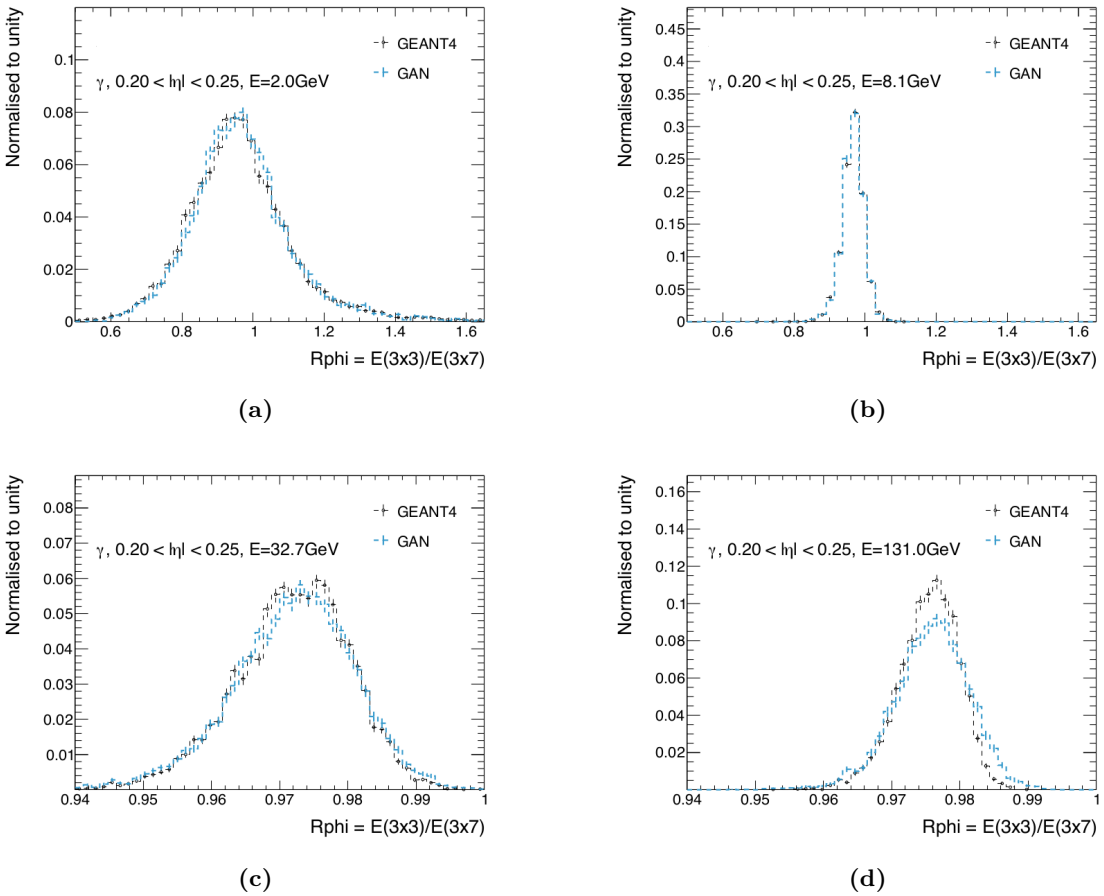


Figure 5.55 – Distribution of R_ϕ for (a) 2 GeV, (b) 8 GeV, (c) 32 GeV, (d) 131 GeV photons.

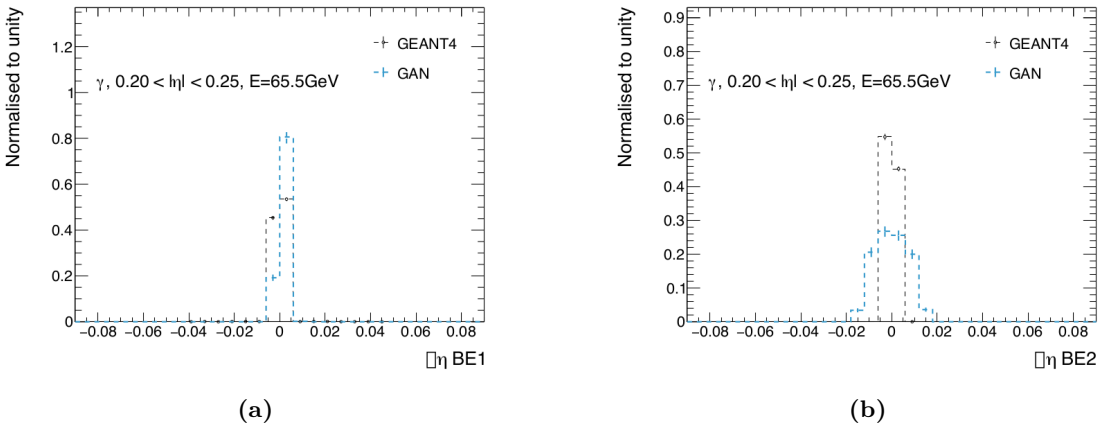


Figure 5.56 – Distribution of difference between the η of the particle and the average η of the shower in (a) Strip layer, (b) Middle layer for 65.5 GeV photons.

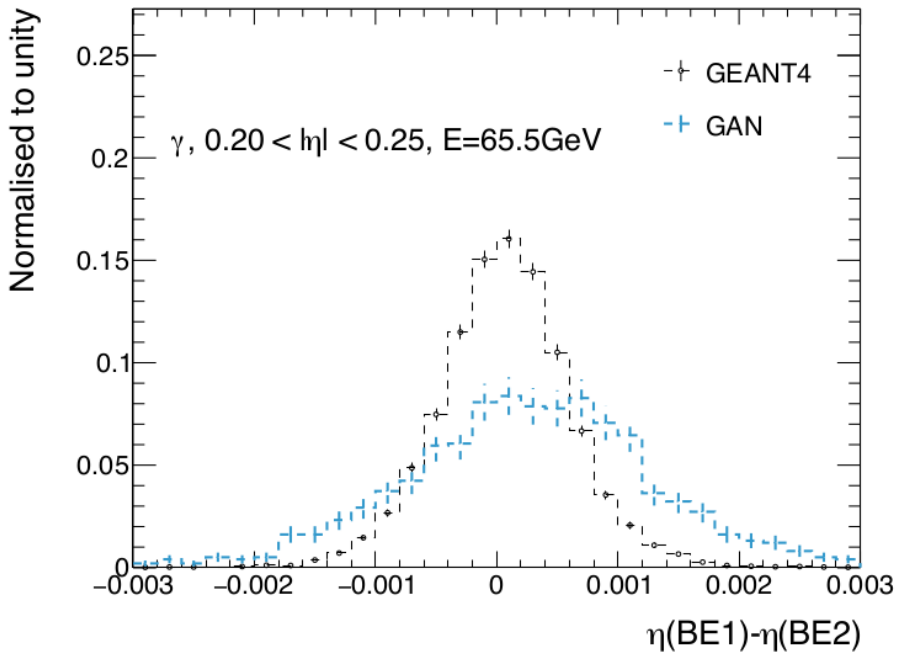


Figure 5.57 – Distribution of difference between the average η of the shower in the Strip and Middle layer for 65.5 GeV photons.

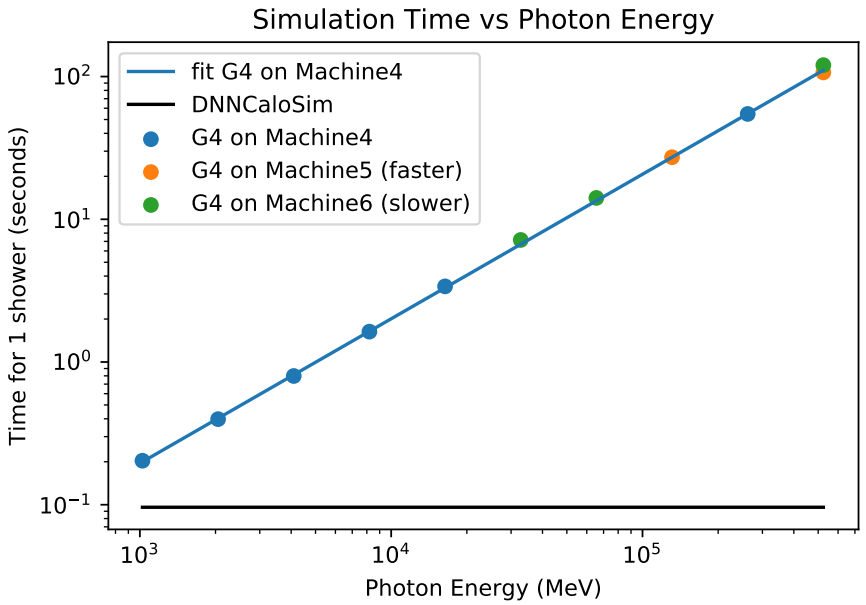


Figure 5.58 – Simulation time for a single shower as a function of the energy of the incident photon for Geant4 and the GAN. The blue line is a linear fit to the logarithm of time to the logarithm of the photon energy for Geant4 measured on Machine4 and the blue dots indicate the data points. Slight variations in simulation time were observed from one machine to another, examples of such points are shown in orange (Machine5) and green (Machine6). The simulation time for the GAN is flat with respect to the photon energy and is shown in black.

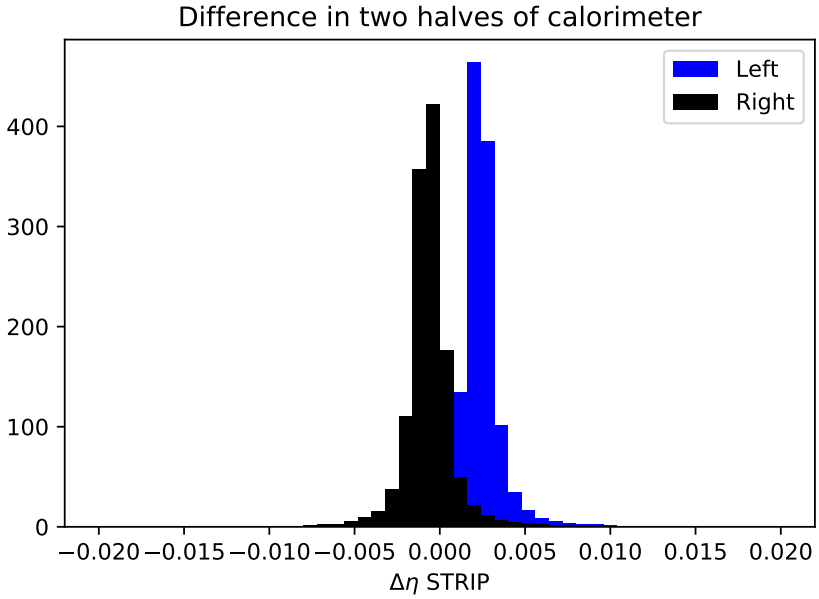


Figure 5.59 – Distribution of difference between the average η of the shower and the impact point of the particle in the Strip Layer for showers from two halves of the detector. The former uses the raw η coordinates while the later uses the true η coordinates, giving rise to a discrepancy in the dataset.

compared to 6.0 GB for `FastCaloSimV2` (last comparison made in February 2019). The ONNX model file is even smaller at 1.3 MB. Although this GAN is trained only on a small region in η, the total memory footprint of the `DNNCaloSim` service is not expected to dramatically increase when expanded to the entire calorimeter.

5.6 Drawbacks

The "two horn" structure for the $\Delta\phi$ distribution in the Strips is because of a discrepancy in the definition of the coordinates used for the calorimeter cells and the extrapolation of the particle position. The former uses the raw coordinates while the later uses the true coordinates. Figure 5.59 shows that each peak comes from a separate half of the detector.

A clear ϕ asymmetry is seen in Figure 5.60, which is the distribution of the difference between the ϕ of the shower and the impact position of the particle as a function of the ϕ of the particle for showers with impact cell $\eta = -0.238$. Figure 5.61 shows the different average ϕ distributions in the two halves of the detector.

This is also a consequence of the fact that the cell positions are raw coordinates while the particle positions are in the corrected coordinates. The GAN is conditioned on the position of the particle relative to the position of the impact cell (i.e. the difference of the two which are measured in slightly different coordinates) and this is the source for the mis-modelling seen in Subsection 5.5.4.7.

These issues as well as the general performance evaluation performed inside `Athena` give valuable insight for the next stage of the project of simulating the ATLAS calorimeter using generative networks.

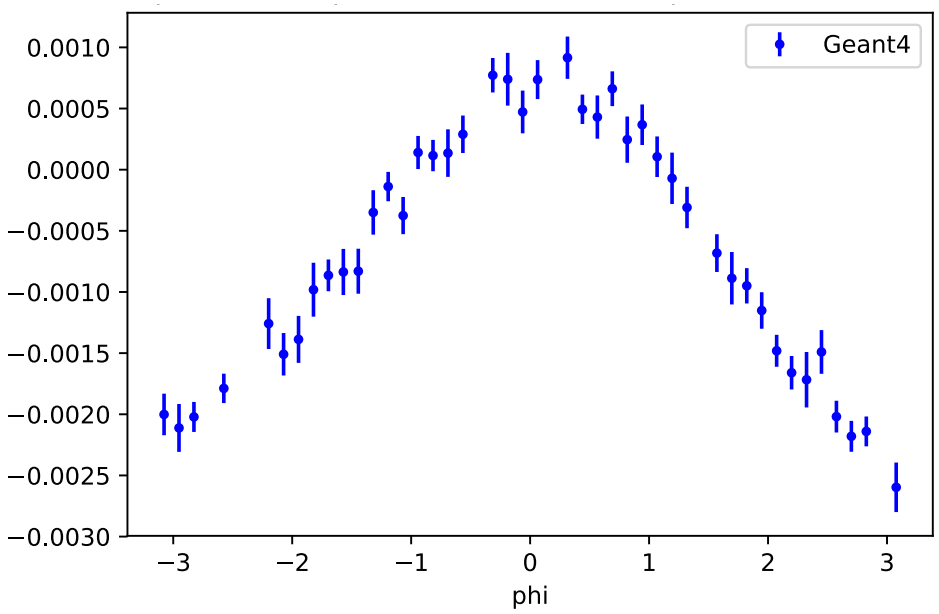


Figure 5.60 – Distribution of the difference between the ϕ of the shower and the impact position of the particle as a function of the ϕ of the particle for showers with impact cell $\eta = -0.238$ in the Middle layer. The distribution is not flat, as was assumed for the training of the GAN.

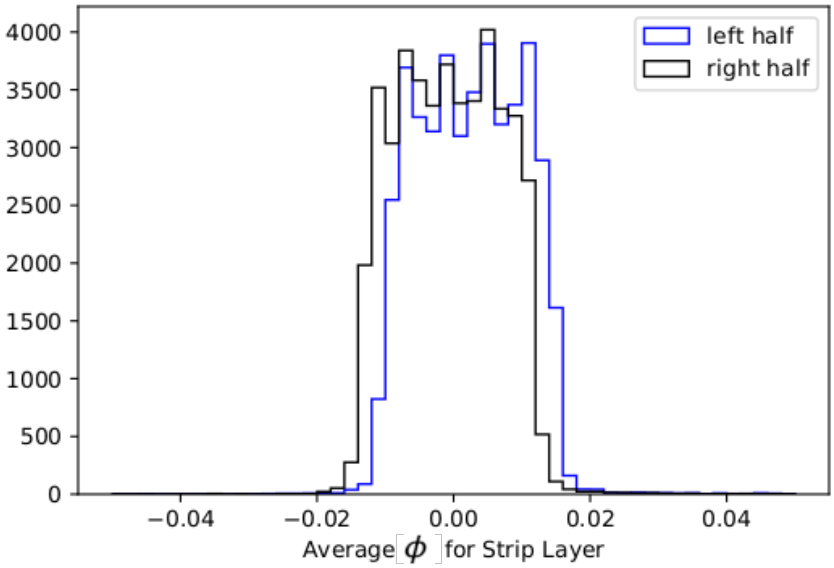


Figure 5.61 – Distribution of the average ϕ in the two halves of the detector for the Strip layer. It is not the same.

5.7 Related Work

In comparison to prior work on simulating calorimeters with generative networks, the dataset used in this study is an official ATLAS production using `Geant4` which incorporates the full details of the ATLAS electromagnetic calorimeter. Prior work is performed on simulations of a small cuboid of a calorimeter or on future detectors which usually have a more regular geometry. The performance evaluation is performed using the rigorous validation framework used in ATLAS for many physics observables. The performance of a deep generative model is for the first time compared to the state-of-the-art fast simulation algorithms used in an LHC experiment.

The cropped images used to train the GAN are quite small and the entire training dataset fit in the memory. The training time for individual epochs is also very small (although many epochs are required for training). No convolutional layers or locally connected layers were found to be useful and therefore there is no inductive bias in the architecture to help the GAN learn the spacial correlations.

The CaloGAN [9–11] demonstrated the idea of using GANs for fast calorimeter simulation. It was trained on a fixed size representative volume of a calorimeter (rather than the entire η or ϕ range) which including multiple layers with varying granularity. In contrast, the GAN described in this thesis is trained on cropped images because it simulates the entire 2π range for ϕ. This is also the reason this project has to accommodate the layer-to-layer alignment configuration conditioning. The architecture of this GAN is very simple compared to the CaloGAN, which has a more physics inspired arrangement of layers.

A WGAN-GP developed for the CMS prototype high granularity calorimeter [95] at the same time as this project also benefited from the use of Convolutional layers and used additional networks trained to regress physics observables from the generated images. A 3D Convolutional GAN [103] trained on simulations of a granular calorimeter of the future Compact Linear Collider (CLIC) detector experimented with both 3D and 2D convolutions and found them to be beneficial. This project also demonstrated significant speed up in terms of training time when scaled to multiple GPUs. In contrast the GAN from this project has little to gain from scaling to multi-GPUs, model parallelism or data parallelism.

The CMS project[10] and the 3D GAN project are applications on a granular and regular shaped calorimeter whereas the ATLAS calorimeter is much more coarse and it is not only highly irregular (in terms of cell sizes) but also suffers from imperfections like the ones discussed in Section 5.6.

Instead of simulating single particle showers, a slightly different approach is taken by [104] where CMS open data is used to train a GAN based algorithm to directly simulate reconstructed jets. The algorithm incorporates additional loss terms architecture optimisations specific to that dataset (such as a strong sparsity requirement).

As discussed in Section 5.4.6.4, a few other projects that train generative models for physics simulations faced similar problems to this project in terms of simulating the correct energy/mass distribution and used either an additional loss or a post-processing network to fix the problem.

[10] Which is studied on a prototype high granularity calorimeter, not to be confused with the current CMS calorimeter.

Table 5.1 – Table summarising the performance of `Geant4`, `FastCaloSimV2` and DNNCaloSim service.

	Geant4	FCSV2	`DNNCaloSim`
Time for 65 GeV shower	10s	70ms	70ms
File Size	-	$\mathcal{O}$(GBs)	9.6 MB
Peak Memory	-	6 GB	2.3 GB

5.8 Conclusions and Future Outlook

ATLAS will require fast simulation techniques to cope with future needs and calorimeter simulation is currently the bottleneck. Traditional fast simulation algorithms that use hand designed parameterisations trade-off accuracy for speed. They also suffer from a large memory footprint. Deep generative models may automatically learn such parameterisations and provide fast, precise simulations without a large memory footprint.

This project was the first time a GAN was trained on the actual `Geant4` simulations of the ATLAS electromagnetic calorimeter for this goal. Integrating the model into the ATLAS software for the first time allowed to make realistic performance comparisons and demonstrated that generative models can speed up simulations and bring down the memory footprint sufficiently to be useful to the ATLAS experiment. For 65 GeV photons the speed up with respect to `Geant4` is two orders of magnitude on a CPU (no GPUs) which is sufficiently for ATLAS requirements. The parameterisation file size is three orders of magnitude smaller than `FastCaloSimV2` and the peak memory usage is only 38% of `FastCaloSimV2`, which would also suffice in terms of memory footprint requirements of ATLAS but there is still plenty of room for optimisation of the overhead in case further improvement in performance if required. The comparison is summarised in Table 5.1. The model was also updated to run with upcoming upgrade to the interface between the ATLAS software and deep learning models using `ONNX Runtime`.

At a later stage of this project, fine tuning of the generative network simulations in terms of corrections and refinement (either using further ML or using traditional means[11]) will perhaps be required, however, for this stage of the project a decision was taken not to resort to inelegant solutions. This allows insights from this work to be more generalisable.

The GAN models several distributions well including the distribution of certain physics observables never tracked at the model optimisation stage of the project. With the help of a second critic, it also exhibits the ability to produce showers conditioned on the energy and position of the incident particle as well as the fast changing detector geometry. Despite training on only nine energy points, it is able to interpolate to unseen energy points well. However, at the next stage of this project, the performance will need to be improved much further and provide consistency of performance at all energy points for it to be used for a real physics analysis.

It is worth noting that some conditioning was easier for the GAN to learn than others. The smooth conditioning to the position of the particle was easy, the energy conditioning did work but required the second critic to model the width of the total energy distribution, and the discrete, quickly changing geometry was the hardest aspect for the GAN to learn. For this reason the geometry conditioning and energy resolution were the primary distributions used to filter out bad versions of the GAN for epoch picking and HPO.

The GAN simulated shower distributions in many cases agree well with `Geant4` even after calibration, which is not always the case for fast simulated samples[12]. In several cases the GAN

[11] Or by training hundreds of GANs for each section of the calorimeter, similar to `FastCaloSimV2` parameterisation strategy

[12] A distribution that agrees before calibration may not necessarily agree well after calibration because of other

already outperforms the traditional fast simulation approaches, which took significantly more person-power and effort over several years to fine tune, but on the other hand in several cases the GAN still has much room for improvement and performs worse than the traditional approaches. Of particular issue is the drop in performance at the highest energy points, which may be fixed in the future with a more suitable training dataset. More importantly, by the end of this effort, there was no physics observable that the GAN could not model at all. The parent dataset (from which the training dataset was built) was simulated for a very different strategy, to optimise the hand parameterised `FastCaloSimV2` in small bins of η and for fixed energy points. The success of this study indicates that it is worth the investment of some resources from ATLAS community (in terms of person-power for training and validation of models and dedicated `Geant4` simulations suited to for this task) for this effort. A paper is in preparation which shows the performance of deep generative models studied in ATLAS after their integration into `Athena` and using some of the high level physics observables used to validate `FastCaloSimV2`.

The high level variables as well as additional scrutiny that became available due to the `Athena` integration revealed several distributions that need to be improved as well as problems with the training dataset. Such insights help take forward the project of accurate and fast simulation of the ATLAS calorimeter using generative networks.

For this highly irregular shaped and course granularity calorimeter, convolutional layers were not found to improve the performance of the GAN, even though there are spacial correlations to exploit. This is partly due to the fact that the cropped images have only 266 cells. Further trouble awaits as these techniques are scaled up to the entire calorimeter. A discontinuity at $\eta = 0.8$, the even more irregular tiles (in the Hadronic Calorimeter), the changing granularity in the Strip layer in the end-caps imply a complete loss of a translation symmetry in $\{\eta, \phi\}$. This suggests a strong need to train on more granular, and uniformly binned data (voxels) which can be cast into calorimeter cells with post-processing (`FastCaloSimV2` already performs such a casting). This is a way to avoid edge effects and ensure that there is only a smooth dependence of the image distributions on the position of the incident particle.

It is also imperative to train on pion induced showers, where correlations and fluctuations are significantly more difficult to model for traditional fast simulation approaches. Unlike photons, pions deposit a non-negligible fraction of their energy in the hadronic calorimeter, therefore it must also be included in the training.

An interesting alternate direction might be to train on point-cloud data generated by `Geant4`. A wide class of graph convolutional layers have recently been developed which would assist in such simulations. Graph convolutional layers may also be flexible enough to help improve the simulation of the ATLAS calorimeter at the cell level.

Although it is known that certain shower shape variables are not well modeled by `Geant4`, a decision was taken in ATLAS to keep `Geant4` as reference for all fast simulation approaches rather than training on real data because `Geant4` samples can be corrected to match data very well. Therefore, a fast-simulator that can mimic `Geant4` can also benefit from the same correction parameterisation. This would avoid having to validate and tune a fast simulation algorithm with data in addition to `Geant4`.

In this project a problem of WGAN-GPs was discovered that usually does not affect applications to natural images (where the total pixel intensity carries little meaning), which is modelling the total energy of an input image, and a solution was proposed for physics datasets. The solution is to train two critics, one with a high gradient penalty weight, and one with a much lower gradient penalty weight which is only able to look at the aspects of the image that require improvement, i.e. the total energy. This concept could be extended further with the insertion of additional physics observables as inputs to the second critic, or additional critics, if necessary, in order

correlations the calibration takes into account which may not be modelled well by the fast simulation algorithm.

to explicitly indicate the importance of modelling particular physics motivated features to the generator within the training algorithm.

The infrastructure built during this effort and the lessons learnt from it have already benefited other efforts working on deep generative models towards the same goal within ATLAS.

The entire work described in this chapter from curating the training dataset (including the oversights in the process) to building the `Athena` service and validating performance as well as identifying problems and solutions for the current approach were performed by the author.

These studies have been made publicly available as an ATLAS technical note [92] and follow up public plots, and a paper is in preparation.

Chapter 6

Offshell Higgs to Four Leptons Analysis in ATLAS

Contents

This chapter will briefly describe the off-shell Higgs boson couplings measurement strategy in the ATLAS experiment at the time of writing and describe studies performed to the sensitivity to the signal strength in the VBF production mode using ML classification models trained on the official ATLAS MC samples available at the time. However, due to the presence of quantum interference between signal and background events, and the distribution of negative weighted events in the dataset, we eventually abandoned the strategy described in this chapter and changed direction to a new strategy to improve sensitivity to the off-shell Higgs boson signal strength in the VBF production mode using ML based likelihood-free inference, which is described in Chapter 7.

6.1 The Higgs boson to four leptons channel

The Higgs boson was first discovered in the $H \to ZZ^{(*)} \to 4l$ decay channel (where l= e or μ) along with the $H \to \gamma\gamma$ and $H \to WW$ decay channels in ATLAS and CMS [105, 106] using the

Run1 data. Precision measurement of Higgs properties continue to be performed in this decay channel. The Feynman diagram of the decay is given in Figure 6.1.

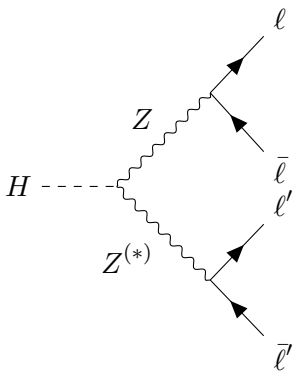


Figure 6.1 – Leading order diagram for the $H \to 4l$ decay.

This channel is often referred to as the "golden channel" because it is relatively clean (easy to separate signal from background), has a fully reconstructable final state with high efficiency and has relatively small detector uncertainties. QCD and other background processes with jets in the final state are often causing difficulties for an analysis but in this case jets are not part of the final state, they are only used to tag the production mode. Further, taus are excluded from the final state because of their low efficiency and to avoid missing neutrinos. It thus allows for precise measurements of the Higgs mass, cross-sections, couplings and spin-parity (denoted CP) despite the smaller SM cross-section compared to other channels (see Figure 2.4).

Apart from all of these measurements, another interesting aspect of the four leptons channel, particularly for this thesis, is that it benefits from an enhanced cross-section for the high mass off-shell Higgs production (which will be described below). This allows for the measurement of the off-shell couplings of the Higgs boson, which is the focus of this thesis. This measurement is useful to indirectly constrain the width of the Higgs boson but is also useful in constraining other kinds of BSM. The motivations for this measurement were discussed in section 2.3.

6.2 Off-shell Analysis

The high mass off-shell production of the SM Higgs boson has a substantial cross-section at the LHC (see Figure 2.9) because although the Higgs boson is off-shell, the intermediate particles in the Higgs production can go on-shell. This provides a unique opportunity to study the Higgs boson at higher energy scales. The destructive interference between certain SM signal and background processes (see Figure 2.10) further enhance the possibility to measure the presence of the signal.

Although such studies could be done within an Effective Field Theory (EFT) framework in the future, ATLAS has adopted the κ framework (see Section 10 of Ref. [107]) for this round of the analysis. Since there is a strong interference between certain signal and certain background processes, the definition and interpretation of usual experimental terms requires clarification. The notion of *signal strength* (which is often interpreted as the ratio of number of excess events measured to the number of excess events expected in the SM) breaks down in the presence of interference.

The signal strength for the ggF Higgs boson production mode $\mu_{\text{off-shell,gg}}$ is defined as :

$$\mu_{\text{off-shell,gg}} = \frac{\sigma^{gg\to H^*\to ZZ}_{\text{off-shell}}}{\sigma^{gg\to H^*\to ZZ}_{\text{off-shell,SM}}} = \kappa^2_{g,\text{ off-shell}} \cdot \kappa^2_{Z,\text{ off-shell}} \tag{6.1}$$

where $\sigma^{gg\to H^*\to ZZ}_{\text{off-shell}}$ is the cross-section of the off-shell Higgs boson production from ggF and decay into a ZZ pair, the $\kappa_{g,\text{ off-shell}}$ and $\kappa_{Z,\text{ off-shell}}$ are the off-shell coupling modifiers with respect to the SM of the $gg \to H^*$ production and $H^* \to ZZ$ decay, respectively. One can similarly write the equation for the VBF production mode:

$$\mu_{\text{off-shell,VBF}} = \frac{\sigma^{VV\to H^*\to ZZ}_{\text{off-shell}}}{\sigma^{VV\to H^*\to ZZ}_{\text{off-shell,SM}}} = \kappa^4_{V,\text{ off-shell}} \tag{6.2}$$

where V = $\{W^{\pm}, Z\}$ with the requirement that the vector boson couplings are modified in the same way. In fact in the previous round of the analysis [12] on data that corresponds to 36.1 fb^{-1} of luminosity, an additional assumption that $\mu_{\text{off-shell,gg}} = \mu_{\text{off-shell,VBF}}$ was also made, which is reasonable given the order of magnitude smaller contribution from VBF.

It is important to note however that due to destructive interference with certain background processes, $gg \to ZZ$ background for $gg \to H^* \to ZZ$ signal and $VV \to ZZ$ background for $VV \to H^* \to ZZ$ signal, the total yield expected is not simply a linear function of the signal strength. In fact the total expected number of events is smaller in the case of SM with the Higgs boson than it is without the Higgs boson. For 36.1 fb^{-1} of luminosity the expected yield (with an $m_{4l} > 220$ GeV requirement) in ATLAS for the $gg \to (H^* \to)ZZ \to 4l$ full process is 96 ± 15 events, whereas for the background-only process $gg \to ZZ \to 4l$ it is 101 ± 16 events and for the signal-only process $gg \to H^* \to ZZ \to 4l$ it is 9.8 ± 1.5 events [12].

The dominant background for this analysis comes from the continuum $q\bar{q} \to ZZ$ process. The dominant production mode for signal is ggF and the subdominant mode is VBF. No other contributions are considered for the analysis. The distributions of the ggF contributions and $q\bar{q}$ background in the off-shell regime can be seen in Figure 2.9.

The on-shell Higgs contributions are also treated as a background for the off-shell couplings analysis because these two measurements need to be separated to allow a Higgs width measurement. In the on-shell analysis the Higgs boson mass is required to be $|m_H^{gen} - 125| < 1$ GeV whereas the off-shell analysis selects events with $m_{4l} > 220$ GeV. Nonetheless, the off-shell analysis can have contamination from on-shell events that pass the $m4l > 220$ GeV cut, such as from a mis-paired leptons from onshell VH production (Figure 6.2), or a rare onshell VBF diagram (see Figure 6.3). Section 6.5 will detail the sample generation for the off-shell analysis to account for these contributions correctly.

The analysis strategy makes certain other important assumptions. It is assumed that any new physics which modifies the off-shell signal strength does not modify the relative phase of the interfering signal and background processes, or make sizeable kinematic changes to the off-shell signal[1]. It also assumed that there will be no sizeable new signals in the search region unrelated to a modified off-shell signal strength.

6.3 State of the Art

ATLAS published the previous round of this analysis [12] on data that corresponds to 36.1 fb^{-1} of luminosity collected in 2015-16. A brief summary is given in this section.

[1] Performing the analysis as an EFT measurement in the future will avoid having to much such an assumption.

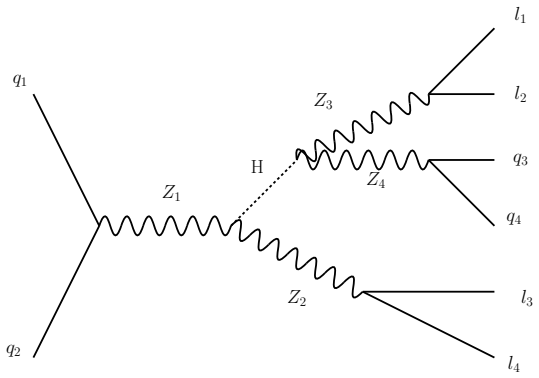


Figure 6.2 – A VH Feynman diagram with an on-shell Higgs boson which is a background for the the off-shell analysis.

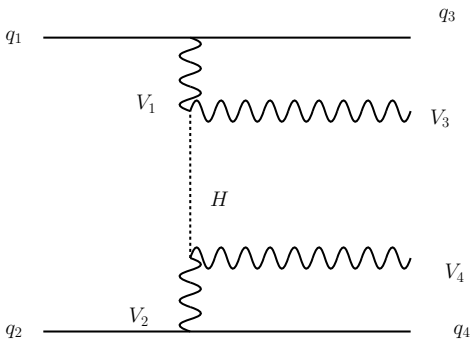


Figure 6.3 – Feynman diagram of an on-shell VBF Higgs that could contaminate the off-shell analysis.

The ATLAS experiment set an upper limit of 3.8 on $\mu_{\text{off-shell}}$ at 95% confidence level (CL) under these assumption that $\mu_{\text{off-shell,gg}} = \mu_{\text{off-shell,VBF}}$, and set an upper limit of 14.4 MeV on the width of the Higgs boson Γ_H. The off-shell region was defined by requiring the invariant mass of the pair of Z bosons (m_{ZZ}) to be above the on-shell ZZ production threshold. The mass of the leading di-lepton pair (m_{12}) is required to be in the range 50 GeV $< m_{12} <$ 106 GeV and the sub-leading pair is required to be in the range 50 GeV $< m_{34} <$ 115 GeV. The invariant mass of the four leptons (m_{4l}) was required to be within 220 GeV $< m_{4l} <$ 2000 GeV, while the on-shell region was defined as 118 GeV $< m_{4l} <$ 129 GeV. There were also requirements for each electron (muon) to have a transverse momentum $p_T > 7$ GeV (5 GeV) and within $|\eta| < 2.47$ ($|\eta| < 2.7$). The highest p_T lepton was required to have a $p_T > 20$ GeV and the second (third) leptons were required to have $p_T > 15$ GeV (10 GeV).

The final measurement was performed with a binned maximum likelihood fit of the matrix-element (ME) based discriminant (sometimes referred to as MELA for 'Matrix Element Likelihood Analysis'), which was computed at Leading Order (LO) with the `MCFM` program [29] using the gg induced signal and background processes as well as the $q\bar{q}$ background process. The formula for the discriminant is given in Equation 6.3 (slightly modified from the original proposal by [29]).

$$D_{\text{ME}} = \log_{10}\left(\frac{P_H}{P_{gg} + c \cdot P_{q\bar{q}}}\right) \tag{6.3}$$

where P_H is the ME squared for an event computed for the $gg \to H^* \to ZZ$ process, P_{gg} is the ME squared for an event computed for the $gg\ gg \to (H^* \to) ZZ$ process, $P_{q\bar{q}}$ is the ME squared for an event computed for the $q\bar{q} \to ZZ$ process and c is a hyper-parameter fixed to 0.1.

While this is a good baseline strategy which was sufficient for the previous round of the analysis on 36.1fb^{-1} of data (which was limited by statistics, particularly for VBF events), there is scope

for more sophisticated algorithms to enhance the sensitivity of the analysis further for the next round which will use the entire Run2 data (integrated luminosity of 139 fb^{-1}). Developing an optimal strategy would also setup a good platform for Run3.

6.4 Probing the VBF production mode in the four lepton decay channel

The previous round of the analysis did not prescribe a separate categorisation for the VBF production mode, however, it is worth exploring for the full Run2 dataset. This production mode has two additional jets along with the Higgs decay products in the final state. A Matrix Element (ME) based discriminant becomes infeasible because, for the non-interfering background processes, the two additional jets come from higher order corrections, and computing these ME based observables at higher order for every event is computationally too expensive. Further, ME based observables do not account for detector effects, which are small for leptons, but larger for jets. Machine learning models are studied instead to improve the separation of VBF events.

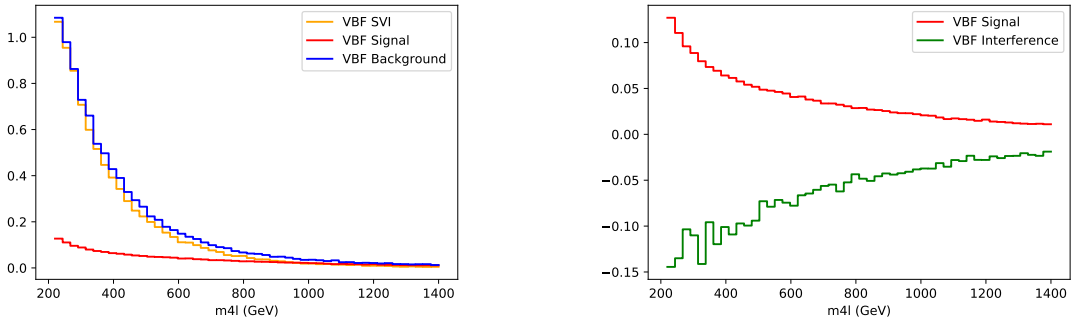
The rest of this chapter will describe the efforts made to improve the sensitivity to the VBF production mode using official ATLAS simulated samples.

6.5 Monte Carlo samples

This subsection describes the Monte Carlo (MC) samples used to model the signal and background processes in this analyses. The events are fully simulated using the ATLAS detector simulation [108] within the `Geant4` framework [4]. Pile-up simulation due to additional pp interactions is added during digitisation by superimposing previously simulated minimum-bias events.

The MC sample for $gg \to (H^* \to)ZZ$ includes the off-shell Higgs process $gg \to H^* \to ZZ$, the continuum background $gg \to ZZ$ as well as the interference between them, and is generated with `SHERPA 2.2.2` + OpenLoops [109–111]. ATLAS simulates this process with `SHERPA` because it allows up to one additional parton in the final state [112]. The QCD renormalisation and factorisation scales are set to $m_{zz}/2$ in `SHERPA` and the `NNPDF3.0_nnlo` [113] PDF set used. NLO corrections are incorporated with a K-factor $K(m_{zz}) = \sigma^{NLO}/\sigma^{LO}$. The K-factors are separately calculated for the signal, background and interference components of the $gg \to (H^* \to)ZZ$ process. LO QCD samples generated with `MCFM` (Monte Carlo for FeMtobarn processes) which are also corrected for higher order QCD are used to re-weight the `SHERPA` samples. Future productions might be made in `MadGraph5_aMC` [82] with up to two additional jets in the final state.

Various MC samples for $pp \to ZZ+2j$ are generated with `MADGRAPH5_AMC@NLO` and `Pythia8` [83] with different intermediate states and/or signal strengths. The QCD renormalisation and factorisation scales are set to m_W [114] and the PDF set NNPDF2.3_lo [115] was used. The off-shell signal sample (S) $pp \to H^* + jj \to ZZ + 2j$ (s-channel Higgs) with SM couplings, $pp \to ZZ + 2j$ background-only sample (V), and a sample including S-V interference (SVI) are available and their differential cross-section is shown in Figure 6.4. Simulating S and V separately is possible with `MADGRAPH5_AMC@NLO` but does not take into account interference effects and may violate unitary. These samples would also ignore a negligible contribution from a t-channel Higgs boson diagram.



(a) **(b)**

Figure 6.4 – Differential cross-sections for (a) $pp \to (H^* + jj \to)ZZ \to llll$ full process (in orange), $pp \to H^* + jj \to ZZ \to llll$ signal-only process (red), $pp \to ZZ \to llll$ background-only process (blue) and (b) the interference component (green).

The Breit-Wigner cut-off (bwcutoff)[1] term in `MADGRAPH5_AMC@NLO` determines the range of masses allowed for a particle, the higher it is, the farther its mass can go off-shell. Based on this consideration, a high bwcutoff was applied for the off-shell Higgs boson simulations. It was later found that changing the bwcutoff from its default value was not advisable because a high bwcutoff in fact allows the Z to go off-shell (which is undesired for an off-shell Higgs boson production, where the Z bosons stay on-shell), however, it was also found that Z mass requirements in the event selection criteria prevents any distortion of the distributions. Future productions will generate only full process $qq \to 4l+2j$ samples instead, without explicitly defining the possible decay chains and without any modification to the Breit-Wigner cut-off, because when the decay chain is not explicitly used in the command, `MADGRAPH5_AMC@NLO` generates the off-shell Higgs (and other) intermediate states irrespective of the bwcutoff value. Comparison of various distributions for events generated with and without a high bwcutoff after applying the event selection criteria show a close agreement, a few examples are shown in Figure 6.5[2]. These comparisons confirm that the current samples produced with a large Breit-Wigner cut-off can be used to optimise the analysis.

For these simulations, m_H is set to 125 GeV and the width of the Higgs $\Gamma_H = 4. - 97$ MeV [21], the scale is set to the m_Z PDG value and the `NNPDF4.0_lo` PDF set is used. At the generation stage an $m_{4l} \geq 130$ GeV cut is applied to remove the on-shell peak.

In addition, two samples with $\mu = 5, 10$ are also produced with HVV couplings scaled by $\mu^{\frac{1}{4}}$ and the Γ_H scaled by μ. The scaling of the width term is an important feature of the analysis strategy. Since the on-shell Higgs contributions are a background to this analysis, the on-shell yield must not be modified when the couplings are changed. Given that the off-shell cross-section is almost independent of Γ_H, and the on-shell cross-section is inversely proportional to it, the Higgs width is scaled by μ to allow a cancellation in the on-shell cross-section (see Equation 2.60) while achieving the desired modification of the off-shell couplings.

The $q\bar{q} \to ZZ$ background process is simulated with `SHERPA 2.2.2` using `NNPDF3.0 NNLO` PDF set for the hard scattering process. The matrix element calculations are accurate to NLO order for 0 and 1 jet states and to LO for 2 and 3 jet states. They are merged with `SHERPA` parton

[1] A resonance is considered to be on-shell if the invariant mass of an s-channel resonance is within

$$M \pm \text{bwcutoff} * \Gamma$$

[2] Problem spotted and studies performed by Martina Javurkova.

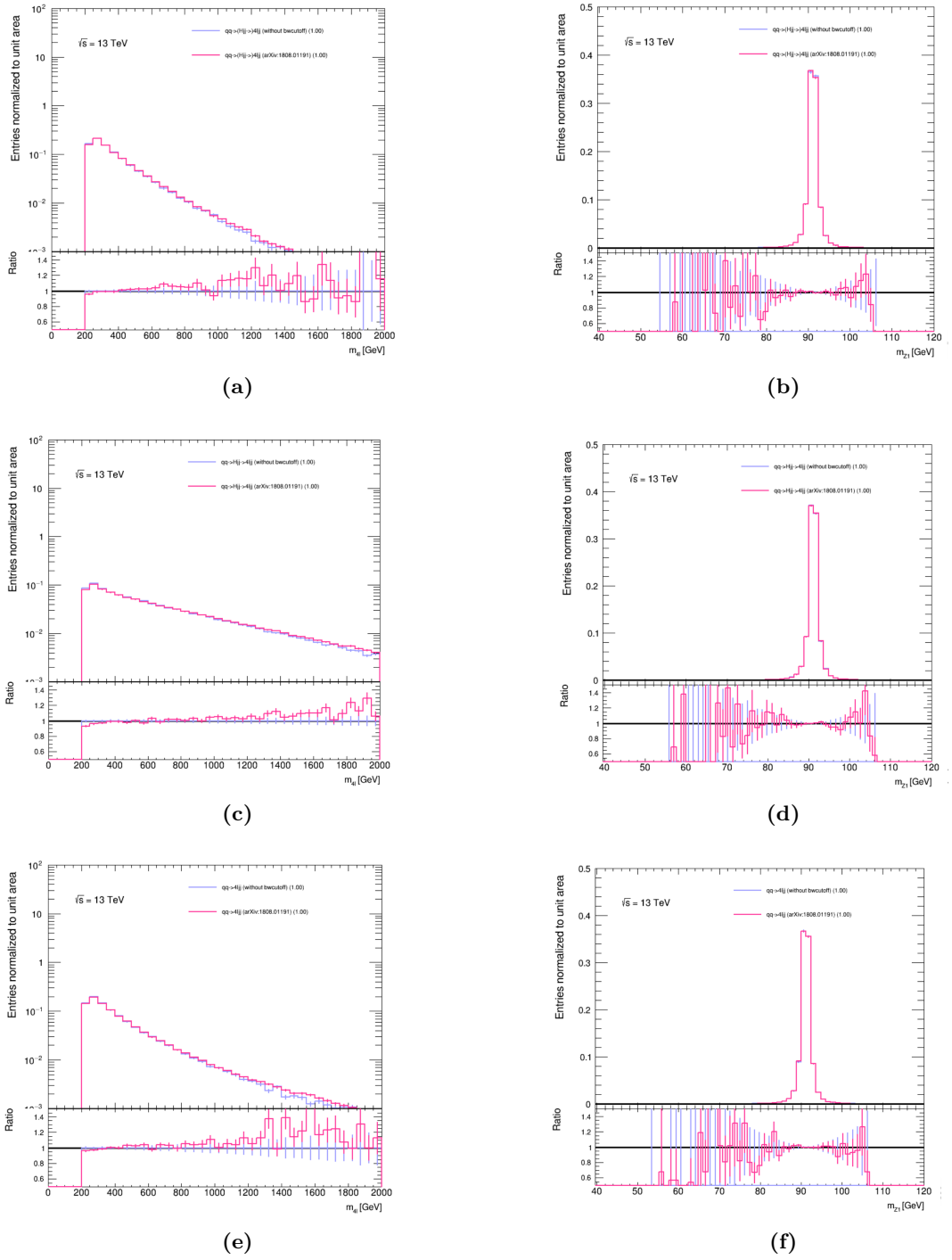


Figure 6.5 – Comparison of $pp \to ZZ + 2j$ simulations with a large Breit-Wigner cutoff (pink) and $pp \to 4l + 2j$ simulations in `MadGraph` with the default Breit-Wigner cutoff (blue). m_{4l} and m_{Z1} distributions are shown for (a), (b) a full process simulation, (c), (d) signal simulation and (e), (f) background-only simulation. The distributions are in reasonable agreement.

shower using the MEPs@NLO prescription [116]. Finally, NLO EW corrections as a function of the m_{ZZ} are applied to the samples [117, 118].

6.6 ML optimisation

This section describes ML based optimisation studies with the aim to improve the sensitivity to the off-shell signal strength of the VBF Higgs boson production mode. They were performed with datasets normalised to a total integrated luminosity of 36.1 fb^{-1}[3], and with the assumption that the findings would scale to the entire Run2 data (now known to be 139 fb^{-1}) straightforwardly.

Quantum interference between the signal and background processes originating from the qq initial make it unclear what kind of classification will provide maximum sensitivity to the VBF Higgs boson signal strength. The fact that training a classifier for signal vs background provides an optimal observable for sensitivity as described in section 4.6 is no longer true in the context of interference between signal and background processes.

Two strategies will be described for optimisation using classification models (with only two target classes). The first involves training with the dataset generated using the VBF full process as one class and all the gg and $b\bar{q}$ initiated processes as the other class. The second strategy involves training on unphysical samples. A dataset generated using only the VBF Higgs boson processes (which is unphysical and does not account for interference effects) is treated as the first class and a dataset generated using only the background processes originating from qq (the VBS process, again unphysical and does not take into account interference effects), gg and $q\bar{q}$ initial states is treated as background. For clarity the first strategy will be referred to as the 'original strategy' and the second one will be referred to as the 'alternate strategy' henceforth. In the case of the alternate strategy, it is imperative that the strategy is finally evaluated on physical simulations to ensure the model is learning correlations that would remain useful for the final objective. The idea is motivated by the simplistic consideration that the Higgs Feynman diagram will mostly contribute to the full process simulation in kinematic regions where the Higgs Feynman diagram has the largest amplitude, and an unphysical Higgs-only simulation indicates these kinematic regions.

The immediately following subsections describe the optimisation performed using the original strategy, and the alternate strategy will be discussed starting subsection 6.6.6.

6.6.1 Pre-selection and Preprocessing

Apart from the $m_{4l} > 220$ GeV selection, as with the on-shell analysis, a pre-selection cut for the 'VBF region' was applied to all datasets, requiring at least 2 jets in the final state with an $m_{jj} > 120$ GeV. This pre-selection brings down the total expected VBF SVI events from 8.5 to 6.1 (71% efficiency), but has a much larger impact on the non-interfering background events, bringing $q\bar{q}$ events from 520 to 60 (12% efficiency) and gg events from 76 to 8.8 (12% efficiency). This pre-selection is also expected to remove on-shell contamination from VH. However, the efficiency for VBF Higgs events is only 52%, taking expected number of events from 2.1 to 1.1 while for VBS background is 71% taking expected events from 9.8 to 6.9, suggesting the possibility for an off-shell specific definition of "VBF region" category.

The pT_{4ljj} variable (transverse momentum of the four lepton and two jet system) is not well modelled in the simulation below 50 GeV, therefore following the strategy of the on-shell analysis, values smaller than 50 GeV were set to 50 GeV before any machine learning is applied. All

[3]These studies began while Run2 data was still being taken and therefore the final total integrated luminosity of Run2 was not known at the time.

continuous input features were standard-normalised. The handling of missing values in the dataset will be described in subsection 6.6.4.

Very similar results were obtained for two prescriptions of class weights, the first equalised the total weights for VBF SVI and $gg + q\bar{q}$ events, the second equalised weights for VBF SVI, gg and $q\bar{q}$ events. The latter is used for the following conclusions.

6.6.2 The ML Models

Boosted Decision Trees (BDTs) usually perform well on structured datasets[4], therefore two such state-of-the-art (SOTA) algorithms were studied, `XGBoost` [59] and `LightGBM` [60]. They are several times faster [119] than TMVA [120] and considering that performance gains are usually achieved by quickly scanning through multiple training strategies (as will also be the case in these studies), that alone[5] leads to overall gain in performance as well. A simple feed-forward neural network without much hyper-parameter tuning was also considered for comparison.

Hyper-parameter searches for the BDT were performed with a grid search using a 3-fold cross-validation where 'discovery significance' (see Equation 4.45) was used as the metric to optimise rather than the traditional AUC.

6.6.3 Permutation Importance using Significance of Discovery

Permutation Importance (PI) is a technique that can be used to estimate the importance of a feature to a trained model based on a relevant performance metric, which can be computed on a dataset coming from the same or different distribution compared to the training dataset. It is described in detail in section 4.9.

The use of PI in this study allowed a greater flexibility in estimating feature importance compared to the inbuilt 'feature importance' functions in the BDT packages. In particular, it allowed to estimate the drop in significance if a particular input feature (physics variable) is removed. For this dataset, the drop in performance (specifically the drop in significance) for a new model trained without a given feature was found to correspond well to the estimated PI.

The differences between PI calculated using AUC and PI calculated using significance are subtle but pertinent. Table 6.1 illustrates this difference. PI with respect to significance provides a more meaningful importance for features, particularly well illustrated for the feature pT_{4ljj} in this comparison table. As it shall be shown, the importance of this feature to the analysis is underestimated by most algorithms.

In several ATLAS analyses, a tight cut is often applied on the BDT score, at around 0.8, because the signal is usually relatively small and there is a preference for higher background rejection. Under such circumstances, pT_{4ljj} allows to more precisely score an event that appears signal-like (where 'signal-like' is a term used to refer to events that get a score $\gtrapprox 0.8$ from the ML model), and therefore plays a larger role in determining the significance than the AUC. The fact that this feature becomes crucial when considering signal-like events is illustrated in the central column of Table 6.1 where the PI using AUC is calculated on a subset of the test dataset that has a score > 0.8. Figure 6.6 is the 'feature importance', an internal scoring provided by `XGBoost` (which estimates the importance of a feature based on the number of times it is used and the gain in separation it brings) and it can be seen that pT_{4ljj} scores significantly lower

[4] As opposed to unstructured datasets, for example, natural images, where neural networks often perform significantly better.

[5] These packages are also better optimised for performance, have fewer bugs and have up-to-date documentation.

AUC			AUC (score 0.8)			Significance		
Score	**Error**	**Feature**	**Score**	**Error**	**Feature**	**Score**	**Error**	**Feature**
0.0910	0.0026	m_{jj}	0.115	0.0149	m_{jj}	0.8876	0.0117	m_{jj}
0.0199	0.0018	width_{j0}	0.0543	0.0074	pT_{4ljj}	0.3337	0.0399	$\Delta\eta_{jj}$
0.0155	0.0006	$\Delta\eta_{jj}$	0.0431	0.0102	ZZ_eta_{zepp}	0.2580	0.0267	ZZ_eta_{zepp}
0.0150	0.0007	ZZ_eta_{zepp}	0.0373	0.0050	width_{j0}	0.1861	0.0203	pT_{j1}
0.0110	0.0006	pT_{j1}	0.0335	0.0060	$\Delta\eta_{jj}$	0.1421	0.0338	width_{j0}
0.0109	0.001	width_{j1}	0.0200	0.0072	pT_{j1}	0.1057	0.0213	pT_{4ljj}
0.0052	0.0006	pT_{j0}	0.0163	0.0046	width_{j1}	0.1031	0.0236	width_{j1}
0.0031	0.0001	m_{4l}	0.0130	0.0040	m_{4l}	0.0878	0.0479	pT_{j0}
0.0007	0.0004	pT_{4ljj}	0.0070	0.0013	pT_{j0}	0.0663	0.0221	m_{4l}
0.0002	0.0001	TrackWidth_{j1}	0.0010	0.0033	TrackWidth_{j1}	0.0089	0.0231	TrackWidth_{j1}
0.0001	0.0001	min_dR_{jZ}	0.0002	0.0028	MELA	0.0051	0.0026	TrackWidth_{j0}
0	0	MELA	-0.0001	0.0002	ggZZtot_dxs	0	0.0004	ggZZtot_dxs
0	0	ggZZtot_dxs	-0.0001	0.0003	TrackWidth_{j1}	-0.0021	0.0073	HZZ_dxs
0	0	HZZ_dxs	-0.0004	0.0004	HZZ_dxs	-0.0042	0.0191	min_dR_{jZ}
0	0	TrackWidth_{j0}	-0.0009	0.0025	min_dR_{jZ}	-0.0217	0.0171	MELA

Table 6.1 – Table of Permutation Importance computed on the Test Dataset with AUC (left and centre) and Significance (right), where only a subset of the Test Dataset with BDT score > 0.8 is used for the central columns. The pT_{4ljj} feature is highlighted to emphasise the importance of choosing the correct metric for computing permutation importance.

than width_{j1} for example, and m_{4l} appears to be the most important feature in stark contrast with permutation importance. The feature importance from `XGBoost` also do not provide any uncertainty, and where in fact found to vary considerably from one training to another even though the performance of the models were very similar.

6.6.4 Performance studies

Hyper-parameter optimisation was a subdominant source of performance improvement compared to pre-processing and training strategy selection, therefore, a large scale comparison of these strategies was performed with `XGBoost`, `LightGBM`, and a dense network while keeping the hyper-parameters for each of the models fixed. The "default" hyper-parameters for the neural network consisted of 3 hidden layers of 32 nodes each and a `relu` activation, and an output layer with 1 node with a sigmoid activation.

Table 6.2 shows a counter-intuitive result, that training without sample weights (i.e. without 'event weights') provides better performance than with sample weights even when the final evaluation of the AUC, significance is done using the correct weights. The reason is elaborated on below (subsection 6.6.5). Further, `LightGBM` and the Neural Network outperform `XGBoost` consistently on this dataset.

While the `XGBoost` 'histogram mode' was found to be around 5x faster than default 'exact mode', it was still 1.2x slower than `LightGBM` on this dataset. Both `LightGBM` and `XGBoost` histogram mode optimise training speed by first histogramming the input variables, thereby reducing the number of computations. The neural network is 15x slower to train that `LightGBM`. The training time for `LightGBM` was between 2 and 5 seconds. The time comparisons were performed on a `DELL latitude E5570` with an `intel core i5 vPro`, without any GPU.

Missing values can occur for numerous reasons in ATLAS datasets, for example for jet observables if there is a missing jet or a track based observable that could not be computed for a given event. Missing values are by default filled with numbers such as -999 or -1, depending on the

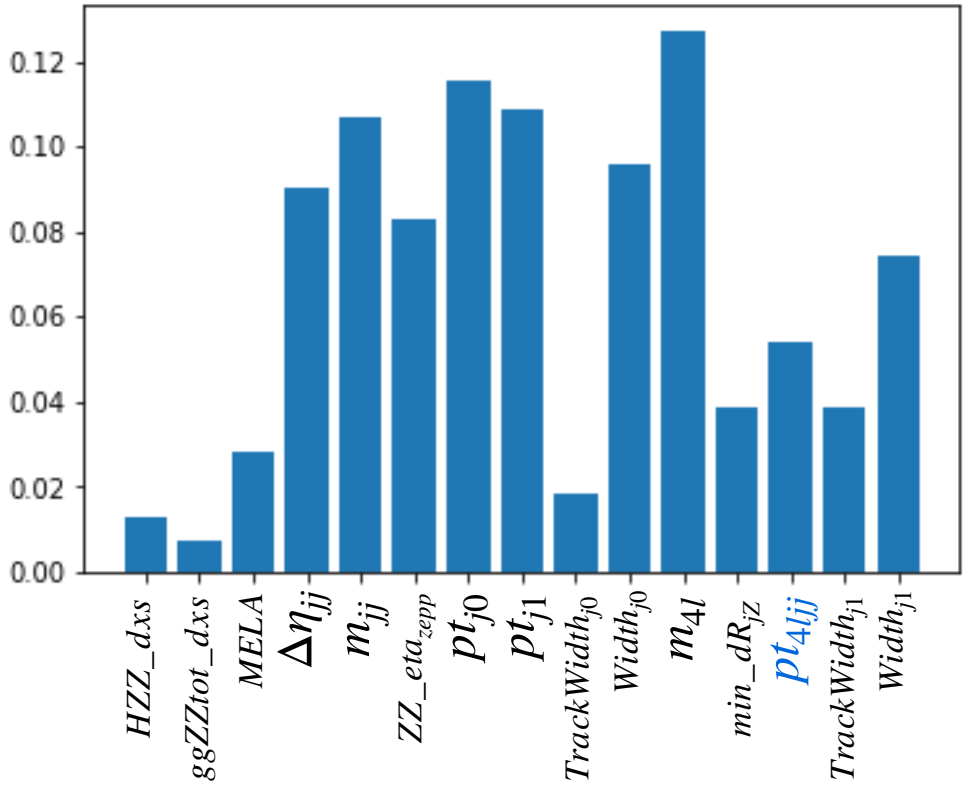


Figure 6.6 – Feature Importance (Gain) from `XGBoost`

cause of the value being missing. They are often left untreated for training, however, it was found that replacing these values with NaN improved performance for `LightGBM` because the algorithm has a smart internal mechanism for missing values (as does `XGBoost`). In the case of the neural network, the best performance was achieved by replacing the missing value with the mean of that feature and also adding a corresponding binary 'flag' feature to indicate whether or not the value was missing. Table 6.3 summarises further investigations into optimising the preprocessing for `LightGBM` and the neural network for missing values in the dataset.

In both these tables, ten independent trainings were performed for a model from each row, and the one with the highest significance is reported. For a more general trend, Figure 6.7 shows the learning curve (average significance for ten independent trainings as a function of the number of training events) for the different models, and it shows an unambiguous trend that `XGBoost` is the best model for a small number of events but is soon overtaken by `LightGBM` and Neural Network when training on more than 40000 events for this dataset. The learning curve has not flattened, indicating that the model could learn more given larger training statistics. Removing all pre-selection cuts apart from the off-shell region requirement of $m_{4l} > 220$ GeV did not yield further improvement in significance.

6.6.5 The sample weights conundrum

The reason unweighted training outperformed weighted training was found to be because 33% of the dominant $q\bar{q}$ background events had a weight of exactly 0, leading to a smaller effective number of training events. The distribution of the weights is shown in Figure 6.8. The $q\bar{q}$ events were generated in three parts to ensure sufficient events in each part of the four lepton mass range. Phase spaces where two simulations had a significant overlap required setting the weights of extra events to zero. However, instead of setting the weight of these events to zero, splitting

Model	Weighted	AUC (test)	AUC (train)	Significance
XGBoost	Yes	0.86	0.86	1.86
XGBoost	No	0.87	0.87	1.84
XGBoost (Hist)	Yes	0.86	0.86	1.86
XGBoost (Hist)	No	0.87	0.87	1.86
LightGBM	Yes	0.85	0.86	1.56
LightGBM	No	0.88	0.89	1.88
Neural Network	Yes	0.86	0.86	1.84
Neural Network	No	0.87	0.88	1.88

Table 6.2 – Table of performance comparisons, AUC on test dataset, AUC on the training dataset and Significance on the test dataset, for the 'original approach' using `XGBoost`, `XGBoost` in Histogram mode, `LightGBM` and a Neural Network trained with and without sample weights, while always evaluated with correct event weights. Ten identical trainings for each case were performed, and the model with the highest significance was chosen in each row.

Model	Default Value Treatment	AUC (test)	AUC (train)	Significance
LightGBM	None	0.877	0.887	1.88
LightGBM	NaN	0.877	0.887	1.98
LightGBM	Mean	0.876	0.886	1.94
LightGBM	NaN with Flags	0.876	0.887	1.91
LightGBM	Mean with Flags	0.878	0.886	1.91
Neural Network	None	0.873	0.877	1.93
Neural Network	Zero	0.875	0.878	1.97
Neural Network	Mean	0.876	0.877	1.97
Neural Network	Zero with Flags	0.875	0.880	1.92
Neural Network	Mean with Flags	0.878	0.879	2.00

Table 6.3 – Table of model performances for 'original approach' with various ways of treating missing values. All trainings are performed without event weights, while the evaluation is performed with correct event weights. Ten identical trainings for each case were performed, and the model with the highest significance was chosen in each row. `XGBoost` was not used for these comparisons.

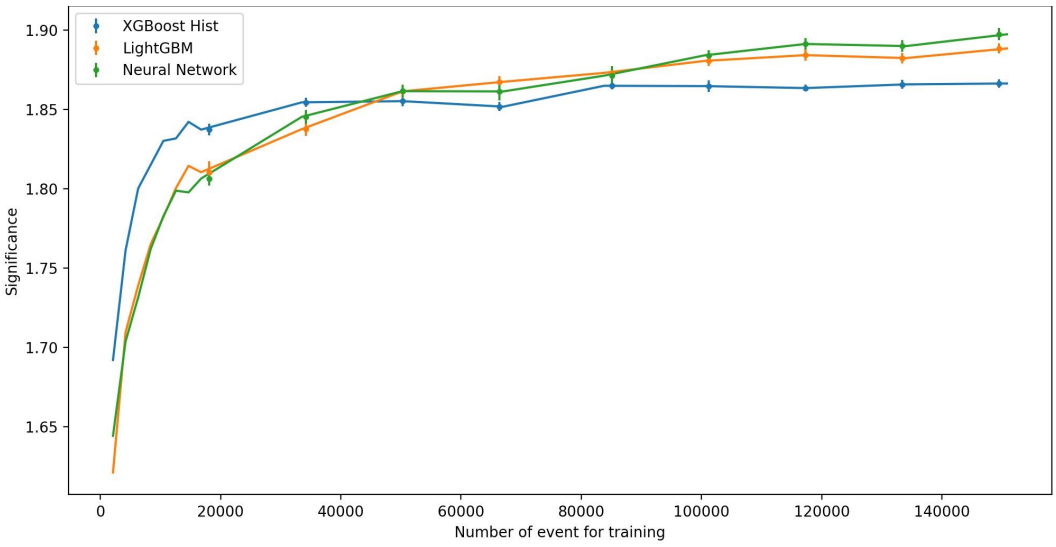


Figure 6.7 – Learning Curves for `XGBoost`, `LightGBM` and Neural Network. Each point represents the mean of ten trainings.

the weight might allow the use of more training events, but this idea is not studied further.

The reason `LightGBM` performed poorly on significance (but not on AUC) when trained with weights is because it accurately identified an unphysical pattern in the dataset. It finds a phase space where $q\bar{q}$ background events had negative weights (which remains consistent even in the validation dataset). Since the gradient boosted algorithm works in reverse for negative weighted events, they are classified as signal-like by `LightGBM` (see highlighted area in Figure 6.9).

The events with zero and negative weights could be meaningfully re-weighted using a GAN like approach to re-weighting, where the generator is required to produce positive weights for the events and the discriminator learns to differentiate the samples with original weights from the sample samples re-weighted by the generator. An additional requirement on the generator to have the weights as close to one as possible would also improve the effective sample size[6] of the dataset. This would be a similar idea to Ref. [121, 122], but this direction was not investigated.

6.6.6 Alternate Strategy

Investigations were also made by training ML models using the alternate strategy of trying to isolate the (unphysical) VBF Higgs-only events from all other (physical and unphysical) background events, as described at the beginning of this section.

The same pre-selection cuts and the same input features to the ML model were used for the alternate approach. In this strategy, very similar AUCs (0.86) and much lower significance of 0.52 were obtained with `XGBoost`. However, neither of these metrics can be directly compared to the original strategy. A new metric, called "Interference Significance", denoted 'iZ', was derived for a fair measurement of the sensitivity of the analysis to the offshell signal strength. The metric is analogous to the typical $s/\sqrt{(s+b)}$ approximate formula used when there is no interference, and also suffers from the same drawbacks (assumes a parabolic log-likelihood curve, not reliable at low statistics). The derivation can be found in section 4.10.1.

This metric demonstrated that the alternate approach consistently outperforms the original one, however it was found to be too unstable to use for model optimisation. Figure 6.10 illustrates the

[6] $n_{\text{eff}} = \frac{(\sum_{i=1}^{n} w_i)^2}{\sum_{i=1}^{n} w_i^2}$ where w_i are the weights for each event.

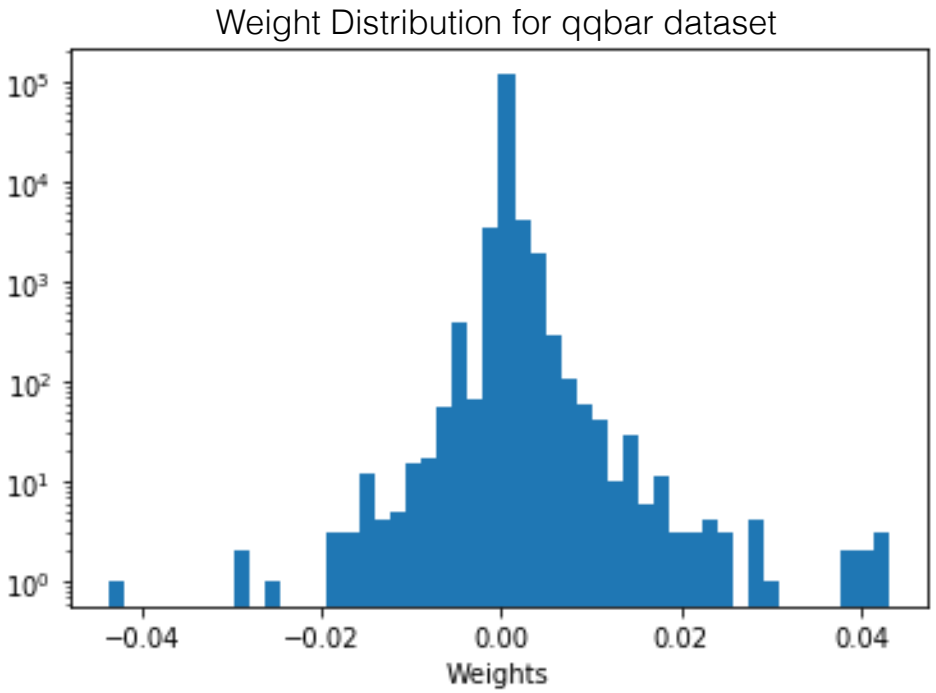


Figure 6.8 – Distribution of weights of $q\bar{q}$ background events.

evolution of iZ as a function of the threshold score for the two approaches, with the maximum values shown in the legend (0.035 ± 0.012 for the original strategy, 0.142 ± 0.049 for the alternate strategy with `LightGBM`).

The BDT output for the signal and the various components of the background is shown in Figure 6.11, where the BDT is trained with the alternate strategy. To differentiate the non-interfering background samples from the interfering one, the samples from gg and $q\bar{q}$ initial states are refereed to as B2, as in Chapter 4.

This metric also suggested that pre-selection cuts were not very useful in final sensitivity of the signal strength, as the iZ before and after pre-selection remained 0.02.

6.6.7 Watch interference using the model output

Training a machine learning model, at times, also allows to better interpret the physics. A BDT was trained on only S (VBF Higgs process) vs V (VBS process) and then applied to independent test samples from S, V and also SVI (full process simulation taking into account interference). B2 events are not used.

It is useful at this point to remind the reader that for an SVI event X with one particular set of truth level four-momentum for the final state objects, the probability $P(X)$ of seeing such an event is given by,

$$P_{svi}(X) = |M_s(X) + M_v(X)|^2 = \underbrace{|M_s(X)|^2}_{P_s(X)} + \underbrace{|M_v(X)|^2}_{P_v(X)} + 2\underbrace{\texttt{Re}(\overline{M_s(X)}M_v(X))}_{P_i(X)}. \tag{6.4}$$

The score distribution is shown in Figure 6.12, where each distribution is normalised to the SM expected yield. Interestingly, the score for the background VBS events has a slight peak near the signal region, and this peak is missing in the full process simulation (SVI sample). It is an indication that the VBS component, $M_v(X)$ of the full process matrix element has a

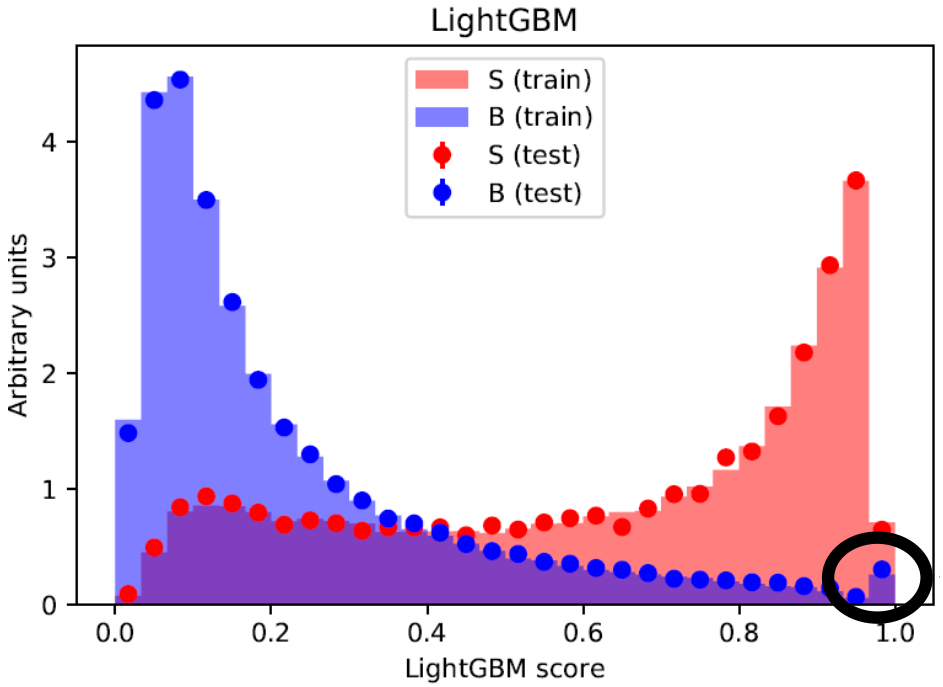


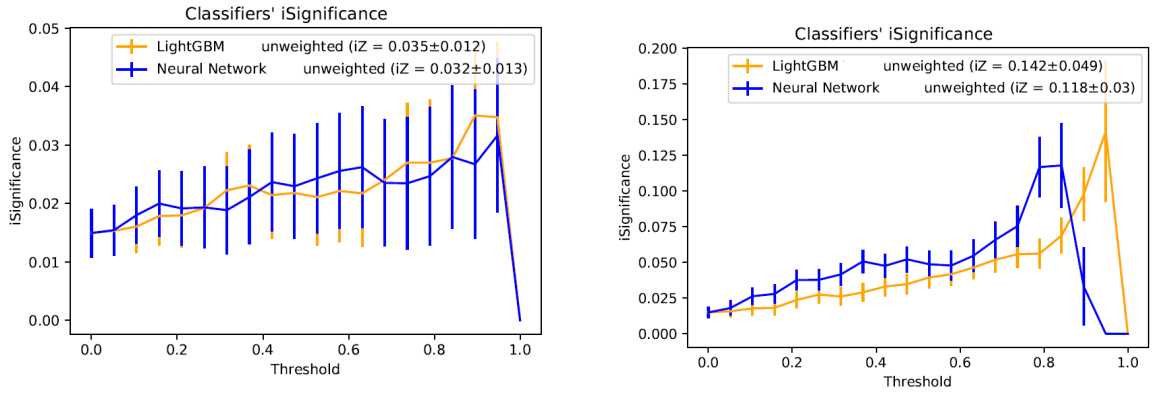
Figure 6.9 – `LightGBM` score distribution when trained with weights. The peak of background events near 1 is due to negative weighted $q\bar{q}$ events.

significant but opposite contribution to the VBF Higgs component, $M_s(X)$, in the signal-like phase space. The signal-background interference is an almost perfectly destructive, resulting in fewer signal-like events in the full process simulation than in the background-only simulation.

6.6.8 Further attempts at optimisation of sensitivity

Since neither of the two approaches is guaranteed to be optimal, ad-hoc combinations were also studied. A two dimensional search using the scores from a BDT trained with each of the two approaches was performed to further optimise the analysis. Considering that in principle the two BDTs give complimentary information, a combination is expected to improve results. The first BDT is trained with the original approach (SVI vs B2), and the second BDT is trained on S vs V only (since the first BDT already learns to reject B2). The results are shown in Figure 6.13a, and in Figure 6.14 for the same idea using two Neural Networks. The iZ for bins with negative number of background events was set to zero. The number of bins used in this two dimensional optimisation had to be decreased to prevent finding a bin with very few events and very high iZ score. The four dimensional plot in these figures shows the distribution of the scores from the two models in the x and y direction, the iZ in the z direction and the number of samples in each bin in a colour grading. Although this strategy did provide an improved iZ ($iZ = 0.22$ in Figure 6.13a), the number of samples and the instability of the iZ metric prevented forming clear conclusions.

The idea to train SVI + B2 vs V + B2 was also briefly studied, motivated by the fact that it is signifies the true objective of the analysis, to improve sensitivity to SM with Higgs vs SM without Higgs. Multi-class, multi-label classification, custom decision trees that split nodes based on $\sum iZ^2$ (instead of entropy/purity gain), and neural networks to categorise events into four arbitrary bins in such a way as to directly optimise $\sum iZ^2$ were also investigated. In every case, the results remained inconclusive because iZ is very unstable, and the training dataset had an easy to find phase space full of negative weighted $q\bar{q}$ background events. The phase space



(a) **(b)**

Figure 6.10 – Evolution of iZ as a function of the threshold model score for `LightGBM` and a Neural Network trained with (a) the original approach and (b) the alternate approach.

with negative weighted background events could be found with just two successive cuts on jet observables. Results in terms of iZ were not reproducible for these strategies. In most cases the models are overtrain very easily, even for conservative hyper-parameters.

6.7 Conclusion: A New Direction

Optimisation of the sensitivity was successfully performed to a large extent and it was conclusively shown that the alternate strategy (training S vs V+B2) was better for improving sensitivity to the offshell signal strength of the VBF produced Higgs boson. While BDTs did provide additional sensitivity compared to a simple cut, the dataset contained phase spaces with negative weighted background events which were too easy to find for a BDT (or even two two cuts) and this deterred more aggressive optimisation. However, using classification models to improve signal sensitivity would not have been the optimal solution in the context of quantum interference even if there we no issues with negative weights. Since the signal strength is no longer just a scaling of the signal event weights, there is no guarantee that the optimisation performed on the SM datasets would be optimal for other values of the signal strength μ. If only the yield is used for a negative log-likelihood curve, two local minima are expected considering the number of expected events is quadratic in $\sqrt{\mu}$ (this is elaborated further in Chapter 7), and such a degeneracy may not be lifted completely by an optimised observable. For this reason, a strategy is needed where the actual sensitivity of the analysis can be optimised.

Based on the insights from the work discussed in this chapter, we surveyed previous strategies to deal with quantum interference in ATLAS (such as [123, 124]) and upcoming inference-aware [64, 66] machine learning strategies in search of a more principled approach to optimisation of this analysis in comparison to the ad-hoc solutions studied in this chapter. We selected a very promising new ML based approach for a further investigation which can be adapted for this analysis in such a way that the optimisation takes into account quantum interference.

Chapter 7 will detail studies using this new approach. It will be demonstrated that there is in fact a significant amount of sensitivity to be gained when an optimal analysis strategy is adopted. It will be shown that certain physics-aware inference strategy promise significant gain over the use of any single optimised observable for the final likelihood fit, when there is significant quantum interference between the signal and background processes. These studies will not only impact the improvement in the sensitivity to the VBF analysis but the ggF analysis as well. Therefore, the decision to abandon the strategy described in this chapter in favour of a more

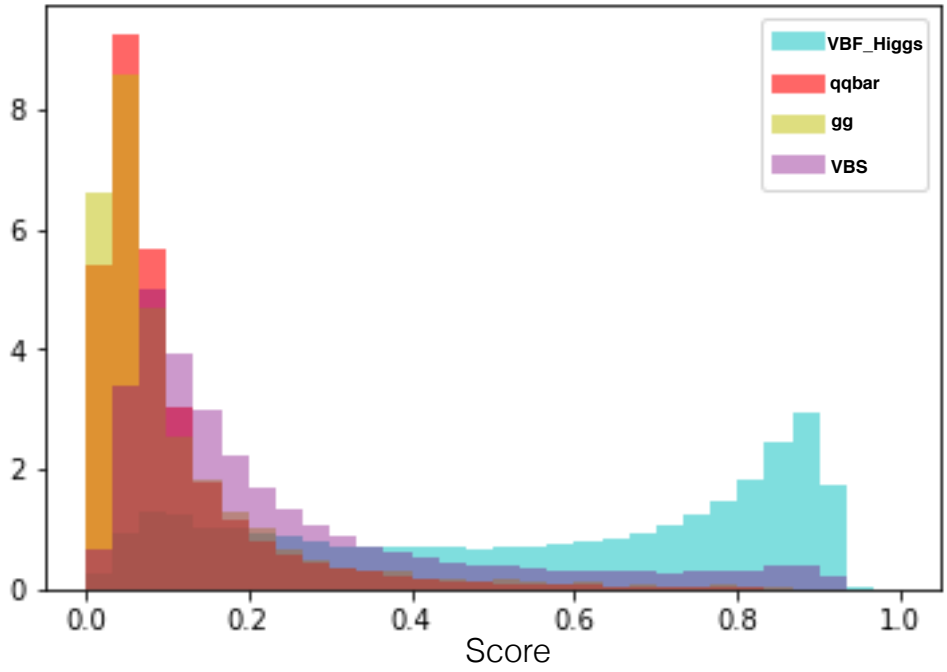


Figure 6.11 – Distribution of `XGBoost` score for S (VBF Higgs signal), V (VBS background) and non-interfering background processes B2 (from $q\bar{q}$, gg initial states) from a model trained using the alternate approach (S vs V + B2).

ambitious strategy will be shown to be worthwhile.

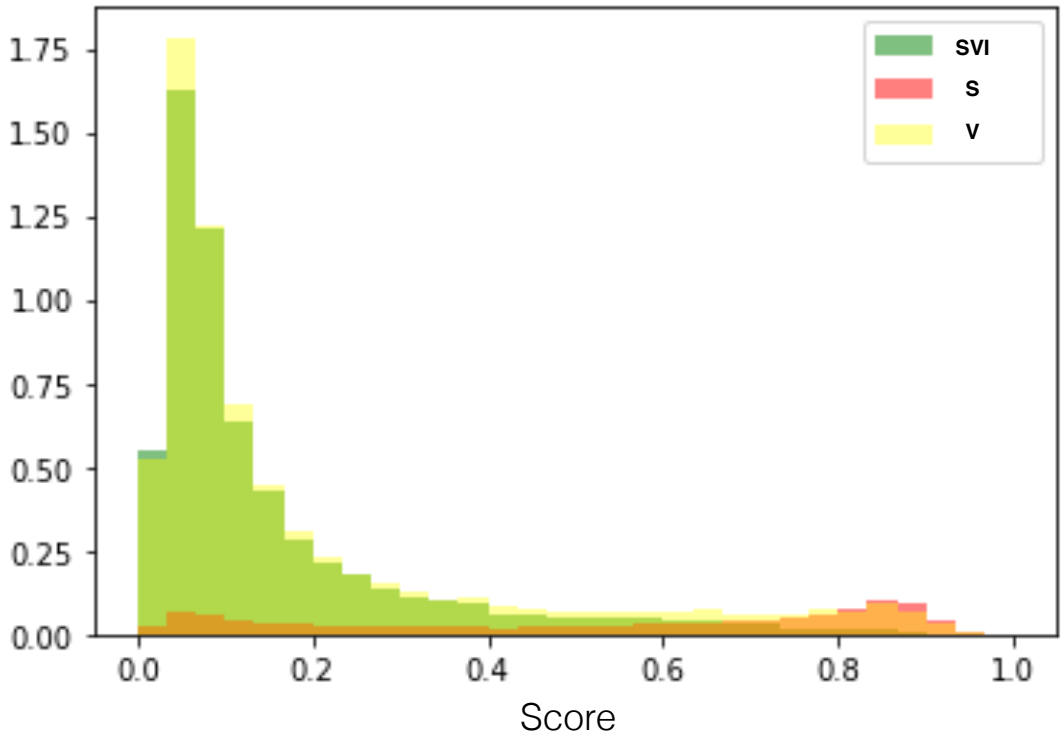
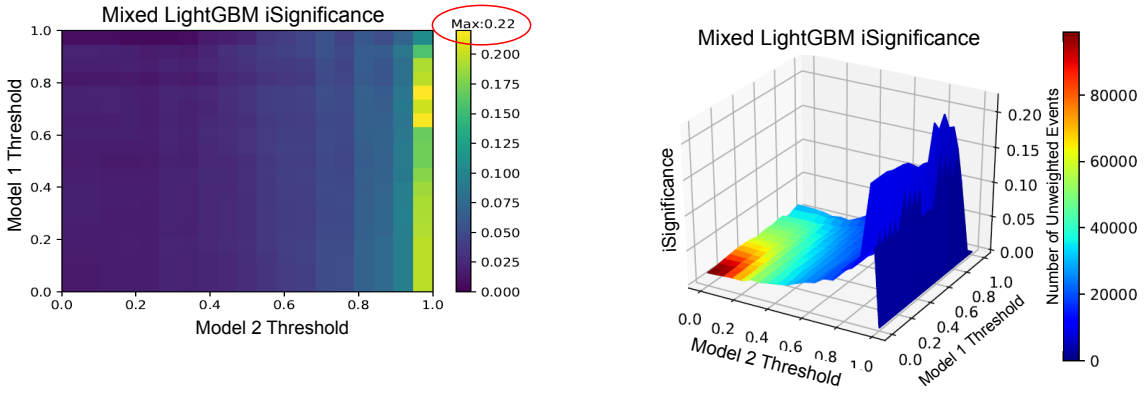
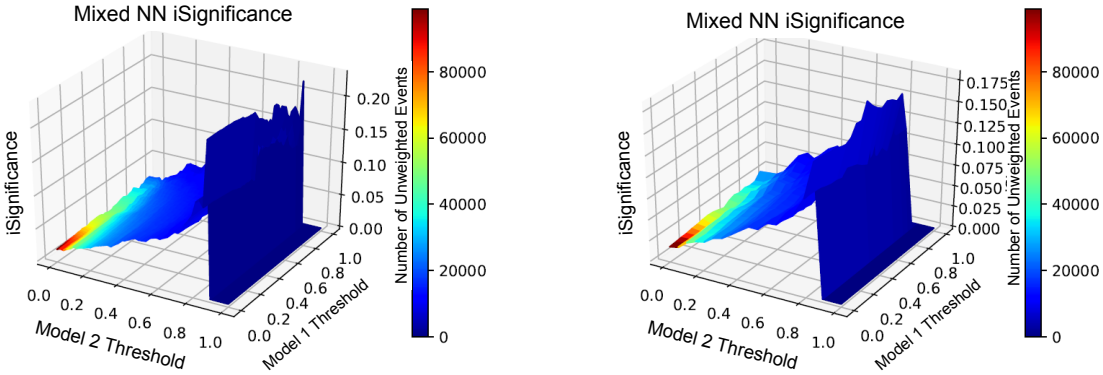


Figure 6.12 – `XGBoost` Score distribution for the S (VBF Higgs), V (VBS background), and SVI (full process simulation of VBF including interference) for a classifier trained on S vs V only, without gg and $q\bar{q}$ backgrounds. Each distribution is normalised to its expected yield. V peaks near the signal region with almost the same magnitude as S, whereas SVI has no peak near the signal region, due to destructive interference between S and V processes.



(a) **(b)**

Figure 6.13 – (a) A two dimensional histogram of iZ as a function of models trained with the original (y-axis) and alternate (x-axis) approach and (b) the same where the third dimension indicates the iZ and the colour indicates the number of unweighted events in each bin. While two dimensional search produces a very high iZ, the optimal selection is very stringent, at which point the iZ may not be reliable due to low statistics.



(a) **(b)**

Figure 6.14 – A two dimensional histogram, with (a) 50 bins and (b) 20 bins, of iZ as a function of models trained with the original and alternate approach, where the third axis indicates iZ and the colour indicates the number of unweighted events in each bin. Even with few bins, the optimal selection is very stringent, at which point the iZ may not be reliable due to low statistics.

Chapter 7

Likelihood-Free Inference of the Off-Shell Higgs Signal Strength

Contents

Following the insights from Chapter 6, this chapter will discuss a study performed on the feasibility of using a class of machine learning based likelihood-free inference models that leverage additional information available in particle physics simulators for the measurement of the off-shell signal strength of the Higgs boson produced via Vector Boson Fusion. Since the use of such simulator-assisted learning techniques require adapting the ATLAS software infrastructure to collect and carry the additional information through the simulation chain, the study was performed using the `Delphes` fast detector simulator. In parallel, the ATLAS HZZ software chain was modified to allow the use of these techniques on ATLAS datasets with full detector simulation in the next iteration.

The chapter will begin with a discussion on the consequences of quantum interference and the challenges it poses to traditional techniques. It will then introduce the family of models under

study, discuss how this strategy can be adapted for a signal strength measurement, and outline dataset production setup. Finally it will present some very promising results for a simplified problem (without accounting for background events coming from gg and $q\bar{q}$ initial states, and using `Delphes` for detector simulation) and discuss the future prospects within ATLAS.

7.1 The troubles that come with quantum interference

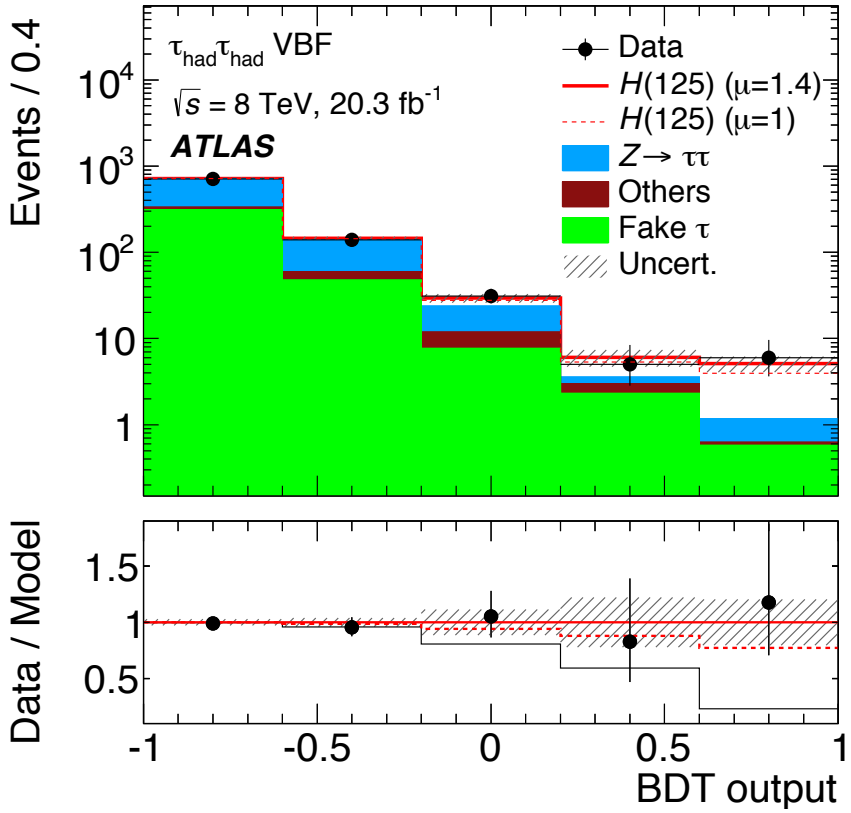


Figure 7.1 – Example of an ATLAS signal strength measurement: Distribution of a BDT discriminant for data taken at $\sqrt{s} = 8$ TeV in the signal region of the VBF category for the $H \to \tau_{\text{had}}\tau_{\text{had}}$ channel. [125]

In a traditional signal strength (μ) measurement analysis where quantum interference plays no role, one can simulate the signal and background samples separately. The number of expected events is a linear function of μ. One can then train a machine learning classifier (such as a Boosted Decision Tree) to separate the signal and background samples and perform a parameter estimation fit on the distribution of the score when the model is applied to real data recorded by the detector (an example of such a fit is shown in Figure 7.1 from the ATLAS $H \to \tau\tau$ analysis from Run1). Neglecting systematics, and under the assumption that it is an optimal classifier, this is the most precise measurement one can possibly perform. The expected number of events is simply linear in μ ($N_{exp} = \mu S + B$, where S is the signal yield and B is the background yield for the SM), and there is no need to train the model on separate datasets to be optimal to different possible true values of μ in nature. The mathematical reasoning for this is discussed in Chapter 4.

In the presence of quantum interference, this strategy is no longer optimal. The expected number of events is no longer linear in μ, but follows the equation,

$$N_{exp} = \mu S + \sqrt{\mu} I + B, \tag{7.1}$$

where I denotes the interference contribution. This formula follows straightforwardly from Equation 2.52 discussed in Chapter 2. Figure 7.2 is a sketch of how the expected number of events scales with μ, demonstrating that a deficit is expected near the SM value ($\mu = 1$), while an excess is expected at high values of μ. In fact it is the "signal-like" background events (background events with kinematic properties similar to events in an unphysical signal-only simulation) that diminish in number in the presence of signal Feynman diagrams in the simulation (as shown in Chapter 6, Figure 6.12). In many cases the number of expected events for two very different values of μ is exactly the same, causing a degeneracy. Optimising the analysis on the basis of SM simulations does not necessarily make the analysis optimal when it is expecting to set upper limits far above the SM value.

In the $qq(\rightarrow H^* \rightarrow)ZZ$ case, Figure 7.3 shows how a physics variable (the invariant mass of the four leptons) that is usually good for a $H \rightarrow 4l$ analysis cannot distinguish between $\mu = 0$ and $\mu = 4$, however a different variable, the pseudo-rapidity (angular) difference between the two jets, can break the degeneracy in this case. Figure 7.4 shows a p-value scan based on fits using three different observables, $\Delta\eta_{jj}$, m_{4l} and pT_{j1} where the true value is $\mu = 0.5$. Apart from the expected peak at $\mu = 0.5$, they all have a second peak at slightly different locations. This means that if the true value of μ is smaller than the SM value, such an analysis would only be able to set higher upper limits compared to the SM case.

Since neither a cut-based nor a traditional machine learning algorithm would be optimal for this task a machine learning strategy that is optimal not only at the SM value but at all other values of μ could be expected to improve the sensitivity of this analysis.

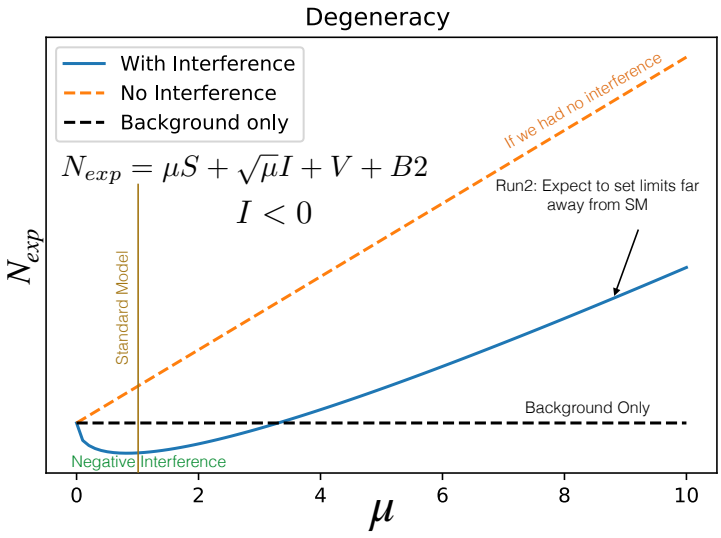
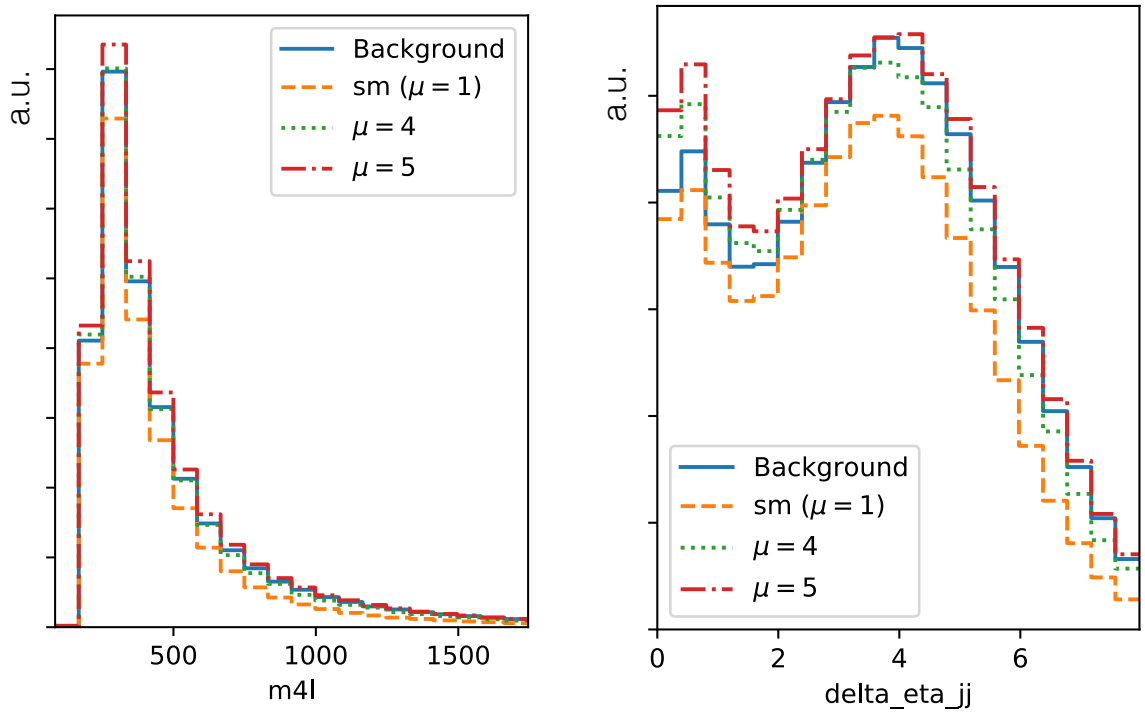


Figure 7.2 – Sketch of the expected number of events observed as a function of signal strength μ, up to an arbitrary normalisation. The black dashed line indicates the total number of events expected for the background only simulation, orange dashed line indicates how the expected number of events would scale with μ if there were no interference and the blue line indicates how to total number of events scales with μ when interference is taken into account. The golden line indicates the SM value, $\mu = 1$, where a deficit of events is expected compared to the background-only case. Black arrow indicates roughly where ATLAS expects to set the upper limit on μ with the full Run2 data, which is at a value of μ where an excess is expected (in contrast to the SM point). Exact numbers are not directly comparable to Figure 7.3 or Figure 7.4 as this extrapolation was performed using official ATLAS simulated dataset.



(a) m_{4l} (GeV) **(b)** $\Delta\eta_{jj}$

Figure 7.3 – Distributions of (a) invariant mass of the four leptons, (b) difference between the pseudo-rapidity of the two jets for VBF full process $(qq \to (H^* \to)ZZ \to 4l + jj)$ with various values of the signal strength μ.

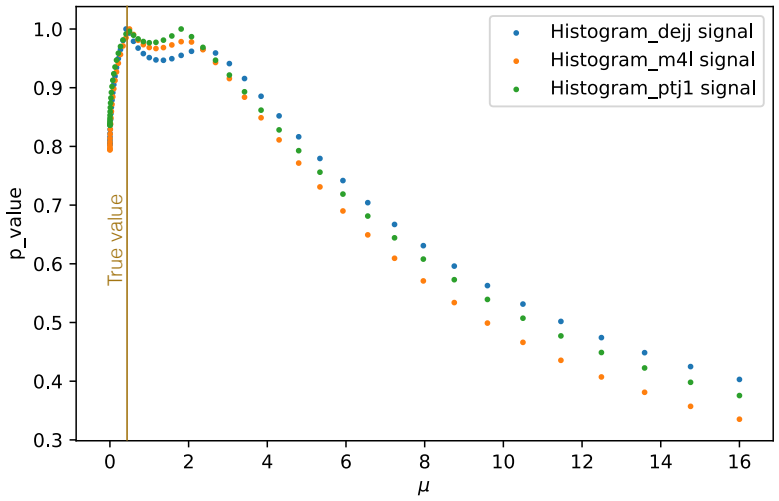


Figure 7.4 – p-value scan for a test dataset generated at $\mu = 0.5$ exhibits a second peak at slightly different locations for the fit for each of the observables.

7.2 Madminer based Likelihood-Free Inference

A promising family of algorithms are studied where deep learning is used to directly learn the likelihood ratio for each event with the help of additional information extracted from the simulator.

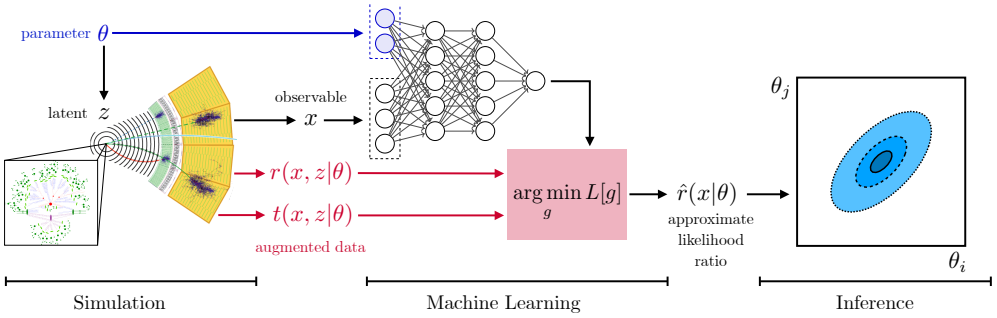


Figure 7.5 – Schematic overview of the family of techniques investigated [78]

These algorithms developed by the authors of Refs. [66, 78–80] and a re at the intersection of machine learning, probabilistic programming, statistics and particle physics phenomenology. The key ideas behind these techniques are discussed in section 4.8 and the schematic diagram of the algorithm show in Figure 7.5 serves as a summary. The techniques rely on the extraction of additional information from the simulator that are used to construct target values to train neural networks that directly learn the likelihood/likelihood ratio between a test hypothesis value of a theory parameter and the null hypothesis value.

This is well suited for a measurement in the Effective Field Theory (EFT) Framework [21], where the theory parameters come from an EFT Lagrangian. However, such an algorithm would also be beneficial for a signal strength measurement in the context of interference, by avoiding the need to define "signal" and "background" class labels, or being restricted to tuning the model for only one fixed value of μ. The extra information extracted from the simulators is only required to train the networks, but not required at inference time, therefore amortising the computational cost.

7.3 Modelling Signal Strength in an Event Generator and Morphing

The first step required to adapt these techniques for a signal strength measurement is to make a connection between the signal strength and the Lagrangian. Even though μ is not a theory parameter of the Lagrangian, it can be mimicked by introducing a new theory parameter in the model that scales the couplings of the Higgs to the vector bosons consistently.

7.3.1 Mimicking the signal strength

The `MadGraph5_aMC` model called `sm model` was used as a starting point and was modified for this purpose. An additional parameter κ was defined[1] which scales the HZZ as well as HWW couplings such that it mimics the change of Higgs signal strength, following $\kappa = \mu^{\frac{1}{4}}$ (this relation only holds while modelling μ_{VBF}). Since the squared HVV coupling is present in the production as well as decay vertices of the Higgs of $qq \to H \to ZZ$ (see Figure 7.6), κ

[1] This and several steps in this study were performed with close support from Johann Brehmer.

contributes at the order 4 to the entire process. With this prescription we can generate the full process of $qq \to (H \to)ZZ$ including signal background and interference for various values of μ.

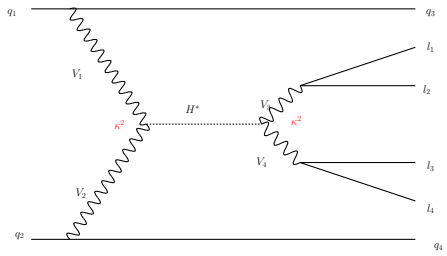


Figure 7.6 – Feynman diagram of $qq \to H^* \to llll$ VBF process. The additional contributions coming from κ is shown.

7.3.2 Re-weighting

The additional information extracted from the simulator that helps data augmentation involves 're-weighting' each event from the event-generator to other values of the theory parameter. This means finding the probability of having observed a particular event if the true value of the theory parameter was something else. The list of parameter values at which generator and/or re-weighting need to be performed are together referred to as 'benchmark points'. Such a computation is possible at the parton-level (using `MadGraph5_aMC`) but not at the final stage of the simulation after detector effects are taken into account. The 'weight' of an event is related to the probability of observing it. By having the weights for each event at the various benchmark points, one can 'morph' [66] (described in section 7.3.3) the events to generate a dataset for any desired parameter point.

Re-weighing is not strictly necessary because morphing can be performed in a different way using data generated at single parameter points, however this method introduces undesired stochasticity (this was briefly studied in work-in-progress on extending this approach to the ggF case, but is not detailed in this document).

7.3.3 Morphing

In this thesis, 'morphing' refers to a technique that allows to cheaply re-weight samples from one theory parameter point to another without the use of an event generator. It is only possible once an event generator has already been used to re-weight the event to a minimum number of benchmark points. Instead of generating events for every value of κ, only a few values are used and then interpolated on a range of values.

The technique is similar to a polynomial fit, and therefore the order of the polynomial required determines the minimum number of benchmarks. After the fit, morphing allows to smoothly interpolate (and to a certain extent extrapolate) the weights of an event to new values of of the theory parameter, in our case κ.

This facilitates the re-use of samples both for training and plotting distributions for various values of κ, thereby decreasing the need to generate samples at many different theory parameter points. The individual events remain unchanged, but their weights are adjusted so that the overall distributions correctly represent the new parameter point. The effective number of events, $N_{\text{effective}} = \frac{\sum weights}{max(weights)}$ (as defined in [66]) is useful in spotting parameter points where morphed event weights have extreme variations. Morphing to points where the physics is very different would make the value of the weights extreme, therefore simulation still needs to be done at several points.

This technique is similar to distribution morphing that is already used in the ATLAS experiment [126], except that it morphs individual events. Once individual events can be morphed, any new distribution can be obtained.

In this scenario, since κ contributes to the matrix element in its fourth power, morphing would be similar to a fourth order polynomial fit. Thus, having weights for each event at five different parameter points is sufficient for 4^{th} order fit, after which an event can be morphed smoothly to any other parameter point.

7.4 Delphes: Very Fast Detector Simulation

`Delphes` [84, 127] is a fast multipurpose detector response simulation tool that can simulate a tracking system, magnetic field effects, calorimeters and a muon system, and possible very forward detectors arranged along the beam line. Over the years it has added features like pile-up simulation, better modelling of jets and visualisation tools.

The key idea[2] is to parameterise the response of the detector for extremely fast simulation to perform studies beyond simple parton-level smearing. It can be configured for ATLAS, CMS and various other detector designs. Although it simulates the detector effects orders of magnitude faster than even the fast simulation tools of ATLAS, it is far less accurate than the fast simulation algorithms developed specifically for ATLAS and it is not suited for detector studies in the case of ATLAS[3].

`Delphes` is open source and often used by phenomenologists who are not members of an LHC experimental collaboration to estimate detector responses. This is because experiment specific simulation tools can sometimes be inaccessible, difficult to run or simply not worth the computational cost for the desired level of detail required of the detector response for a given study.

Instead of performing this feasibility study at the parton-level, a `Delphes` detector simulation was used to obtain slightly more realistic results.

7.5 Monte-Carlo Samples and Morphing Them

The dataset was generated with `MadGraph5_aMC` [82], requesting $pp \to jjzz$ processes with the z decaying to ee or $\mu\mu$. The parton showering was simulated with `Pythia 8` [83] and the detector response was simulated with `Delphes 3` [84].

The simulation was done at six 'benchmark points',

$$\kappa = \{0,\ 0.8,\ 1,\ 1.2,\ 1.35,\ 1.5\},$$

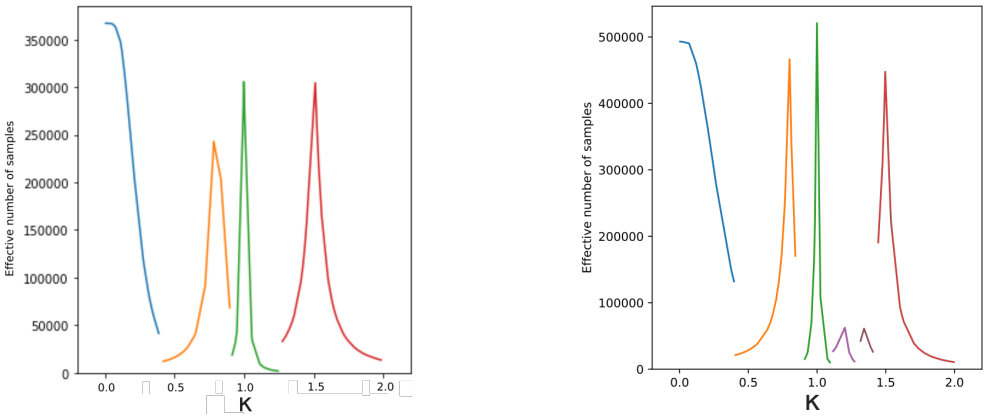
corresponding to

$$\mu = \{0,\ 0.4095,\ 1,\ 2.0736,\ 3.32150625,\ 5.0625\},$$

where each event was re-weighted to all the other benchmarks at the parton level. The $\kappa = 1.35$ point is not used for the morphing fit, and instead spent to validate the morphing setup. In this study, only events from the nearest benchmark point are used to generate a dataset at any given parameter point. Figure 7.8 demonstrates that the morphing setup correctly predicts the evolution of the total cross section at validation point. The distribution of effective number of

[2] This idea was used also by several predecessors of `Delphes`.

[3] CMS has in general used `Delphes` for studies more often than ATLAS and some of the authors of `Delphes` are CMS members. The tuning is more accurate in the case of the CMS detector than for the ATLAS detector



(a) (b)

Figure 7.7 – Effective number of samples for different values of $\kappa = \mu^{\frac{1}{4}}$. Events are generated at (a) benchmark points $\kappa = 0, 0.8, 1, 1.5$ and (b) after a few more events at the SM and two additional benchmark points $\kappa = 1.2, 1.35$. For a new point, events are morphed from the nearest benchmark point (blue, orange, green, violet, brown and red from $\kappa = 0, 0.8, 1, 1.2, 1.35, 1.5$, respectively. It can been seen in (a) that near $\kappa = 1.2$ the effective number of samples is very small, which lead to additional simulations around that point.

events is given in Figure 7.7b, where

$$N_{\text{effective}} = \frac{\sum_{i=1}^{n} w_i}{\max(w_i)} \tag{7.2}$$

as defined in [66]. It is different from the more usual formula, Equation 7.3, used in HEP but is easier to compute and track.

$$N_{\text{effective}} = \frac{\left(\sum_{i=1}^{n} w_i\right)^2}{\sum_{i=1}^{n} w_i^2} \tag{7.3}$$

Interestingly, since the destructive interference is maximal near the SM point, there are very few 'signal-like' events at that point, therefore the events generated at the SM do not morph well to nearby points. To successfully morph a SM dataset to another points requires a reasonable number of 'signal-like' events and the SM dataset has too few. These few events are given extremely high weights when morphed to points away from the SM. This phenomenon is indicated by the sharply falling $N_{\text{effective}}$ in green. Usually for a dataset to have sufficient statistical power, the standard deviation of the weights should be much smaller the mean of the wights (a consequence of Equation 7.3). For this reason, small additional simulations at $\kappa = 1.2, 1.35$ and to be performed to supplement the larger simulations at the four main benchmark points.

After detector level cuts, the full dataset consists of {8.6 M, 8.2 M, 9.2 M, 1.1 M, 1.1 M, 8.1 M} events in ascending order of benchmark points. For contrast, the $N_{\text{effective}}$ plot without these two points is shown in Figure 7.7a. Although for the physics process of interest in this study, the morphing function is already well known to be a quadratic in $\sqrt{\mu}$,

$$\text{weight}_{\mu} = \mu \cdot \text{weight}_{\text{Signal}} + \sqrt{\mu} \cdot \text{weight}_{\text{Interference}} + \text{weight}_{\text{Background}},$$

and therefore should require only three benchmark points in μ, if the morphing is done with respect to the theory parameter κ then more benchmark points are needed. The existing generalised `MadMiner` [81] morphing setup morphs the coupling parameters (rather than the signal strength) and it was more convenient to use.

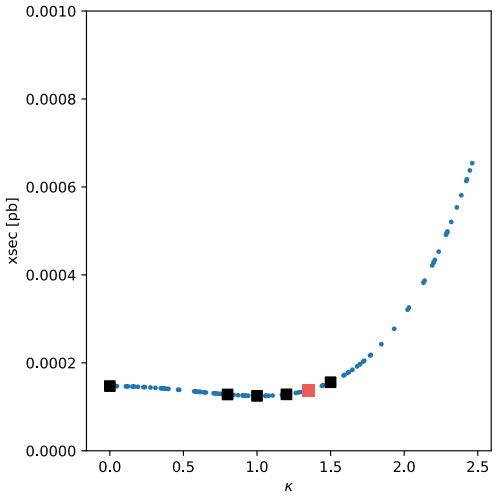


Figure 7.8 – Cross Section as a function of κ. The validation point at 1.35 matches the fitted morphing prediction shown in blue.

7.6 Training the models

Minimal pre-selection cuts were applied, apart from a requirement of at least 2 jets and 4 leptons. The observables used to train the network are,

- the four momentum of the final state objects
- energy of the sum of all visible objects
- pseudo-rapidity (η) of the sum of all visible objects
- missing transverse momentum
- azimuthal angle (ϕ) of the missing transverse momentum
- di-jet invariant mass
- differences in the angles (η, ϕ) between the two jets
- invariant mass of the four leptons system
- total number of leptons
- total number of jets.

In total 35 observables were considered. Their distributions are given in Figures 7.9,7.10,7.11,7.12,7.13,7.14.

From the setup mentioned section 7.3, the augmented data can be used to compute the 'joint-likelihood ratio' (the parton-level likelihood ratio, where x are the observables, z the parton level momenta) ,

$$r(x, z|\mu_0, \mu_1) = \frac{p(x, z|\mu_0)}{p(x, z|\mu_1)}$$

from the weights of each event for two different hypotheses, μ_0 and μ_1, and the 'joint score', $t(x, z|\mu) = \Delta_\mu \log p(x, z|\mu)$, can also be calculated from the morphing setup by taking the gradient of the polynomial for any μ.

These terms are used to train the models using the loss functions detailed in section 4.8. They were implemented in `PyTorch` within the `MadMiner` package, and the trainings were also performed within this framework.

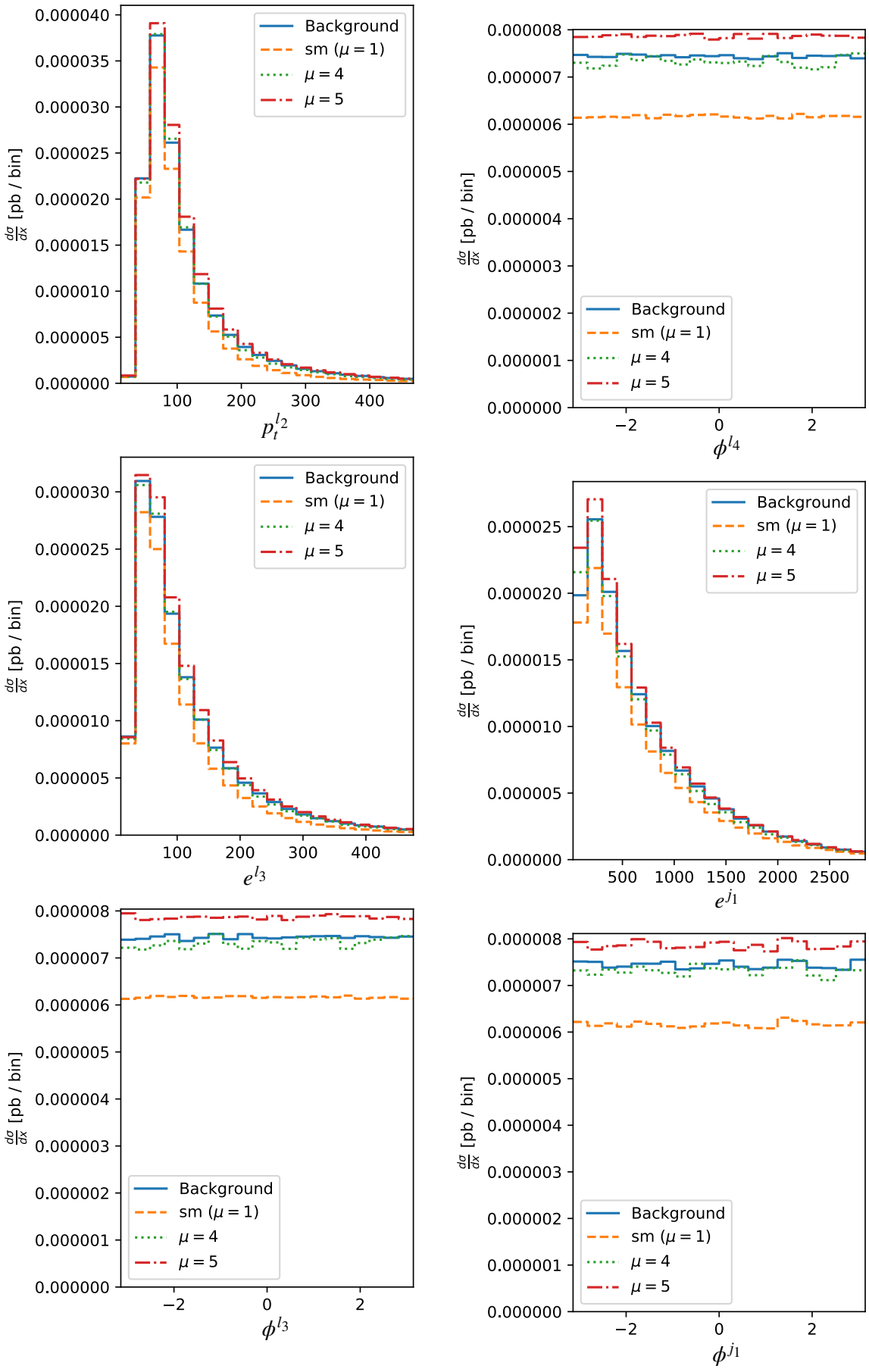


Figure 7.9 – Distributions of observables used in this study at $\mu = 0, 1, 4, 5$.

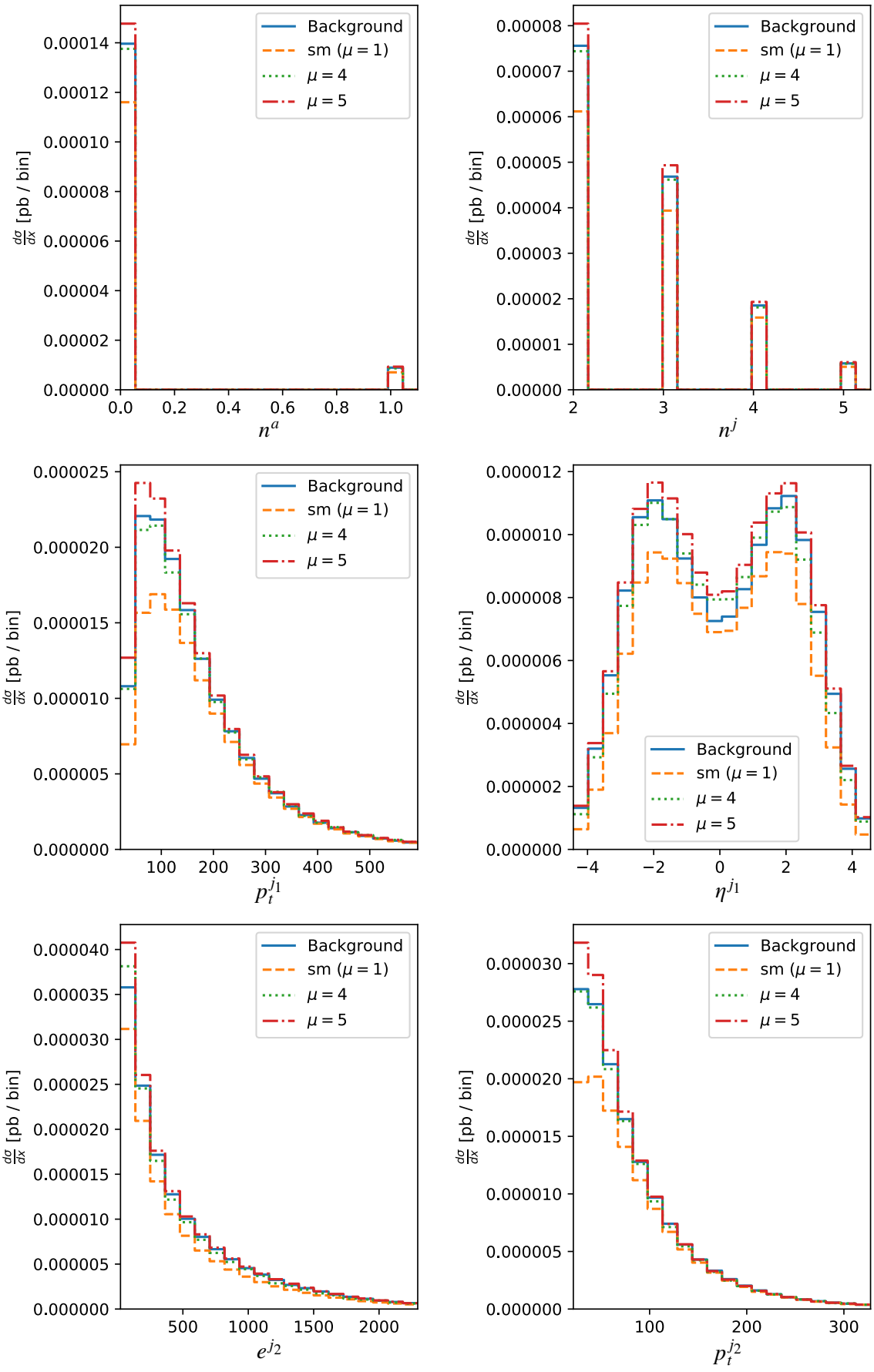


Figure 7.10 – Distributions of observables used in this study at $\mu = 0, 1, 4, 5$.

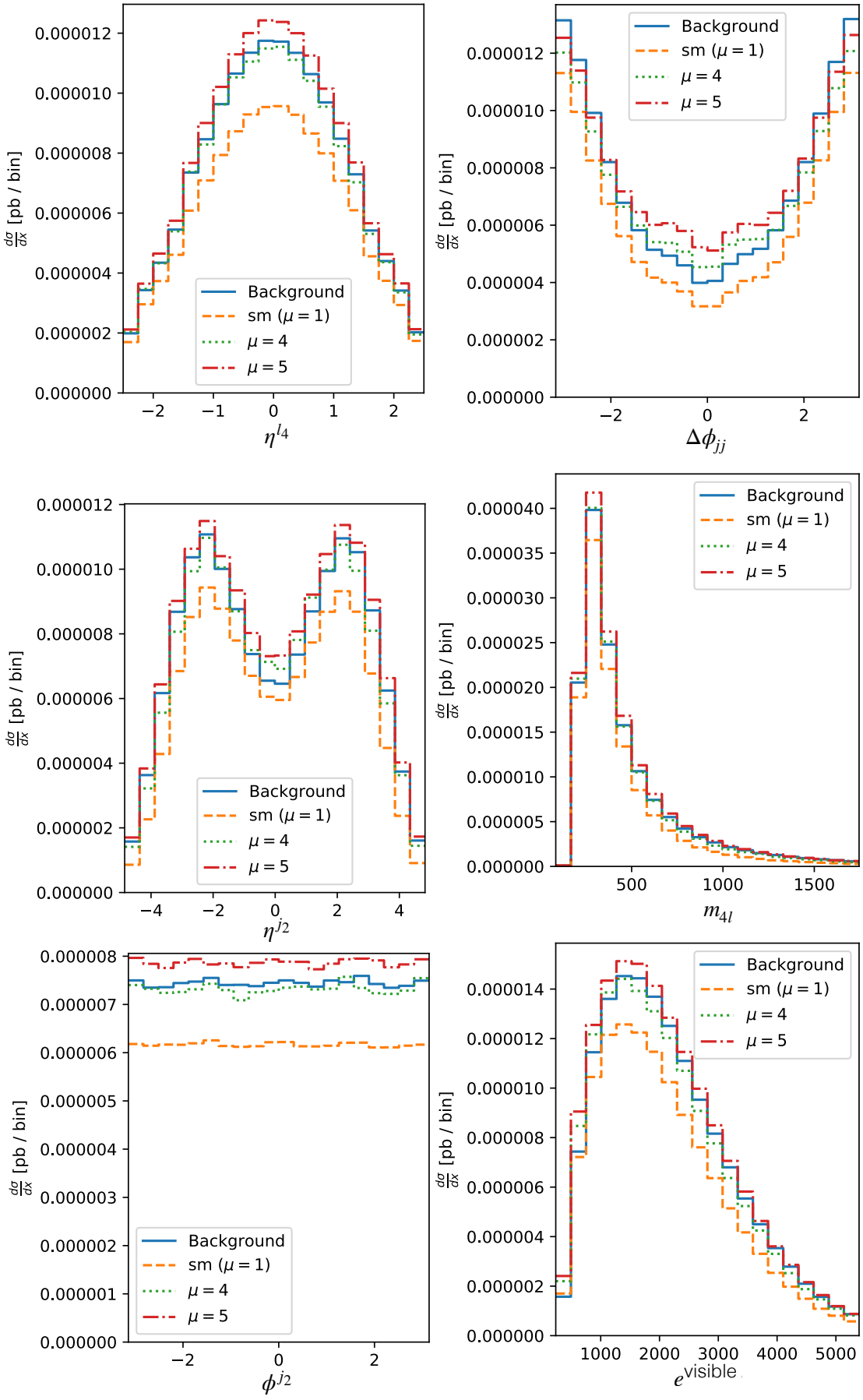


Figure 7.11 – Distributions of observables used in this study at $\mu = 0, 1, 4, 5$.

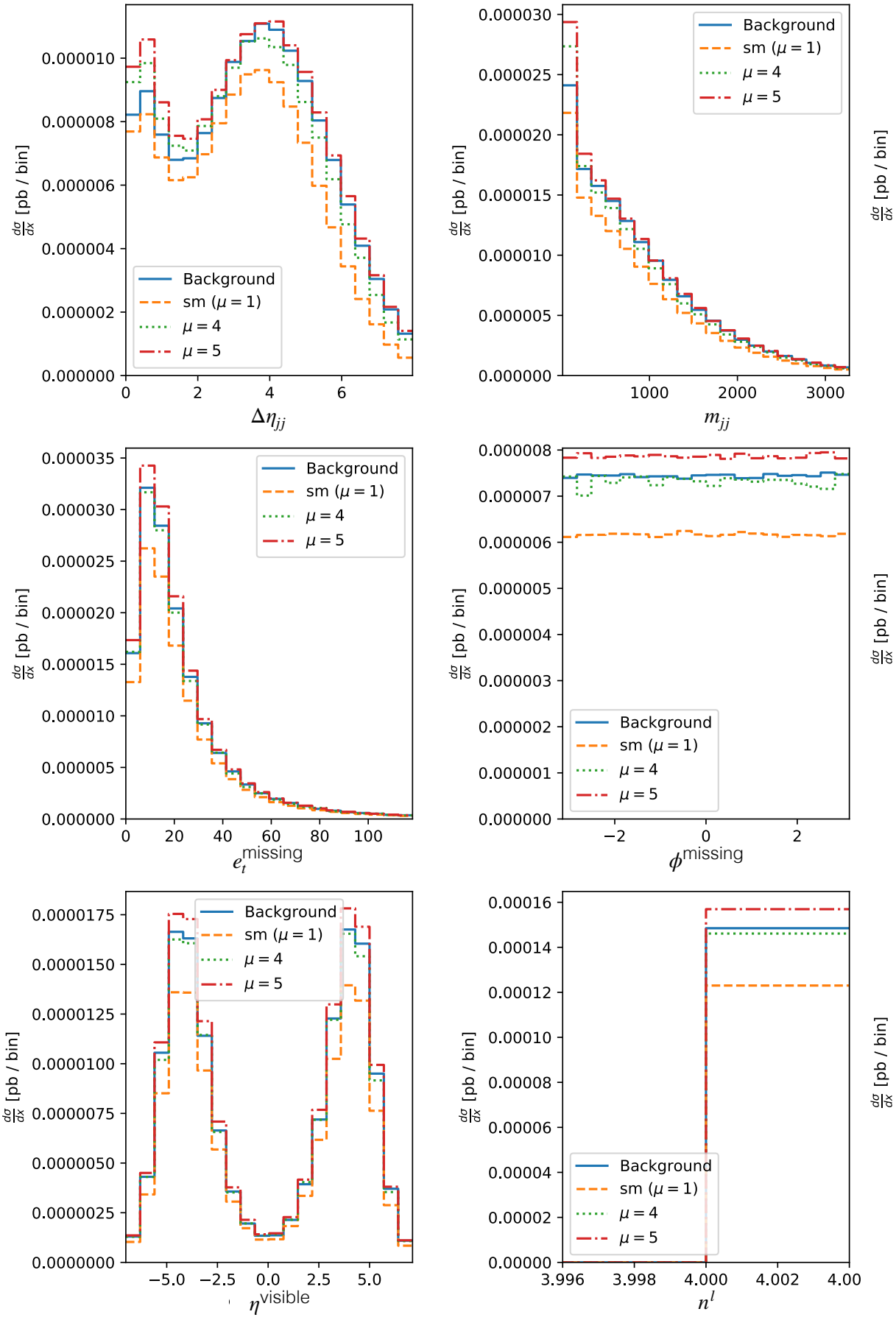


Figure 7.12 – Distributions of observables used in this study at $\mu = 0, 1, 4, 5$.

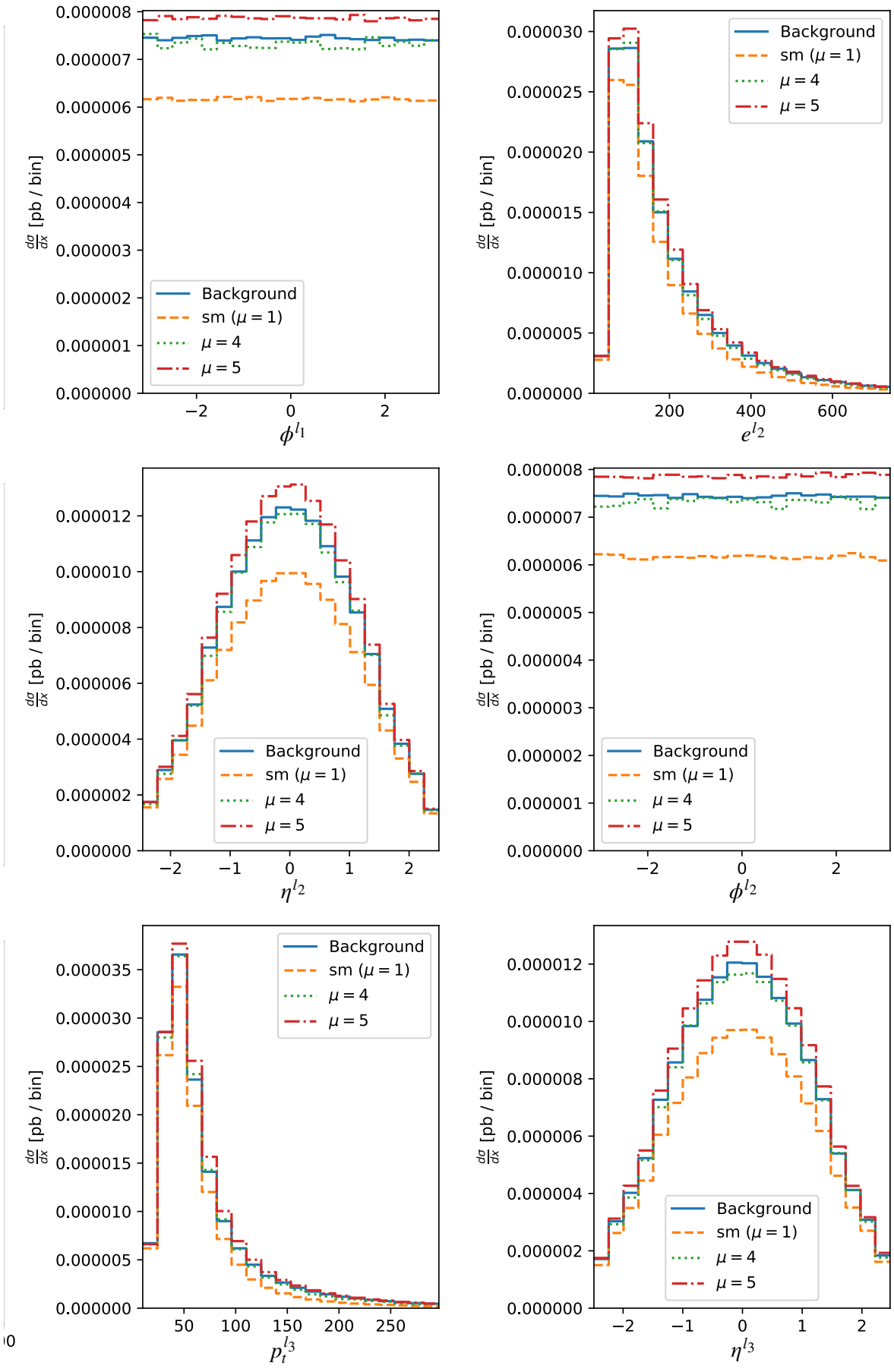


Figure 7.13 – Distributions of observables used in this study at $\mu = 0, 1, 4, 5$.

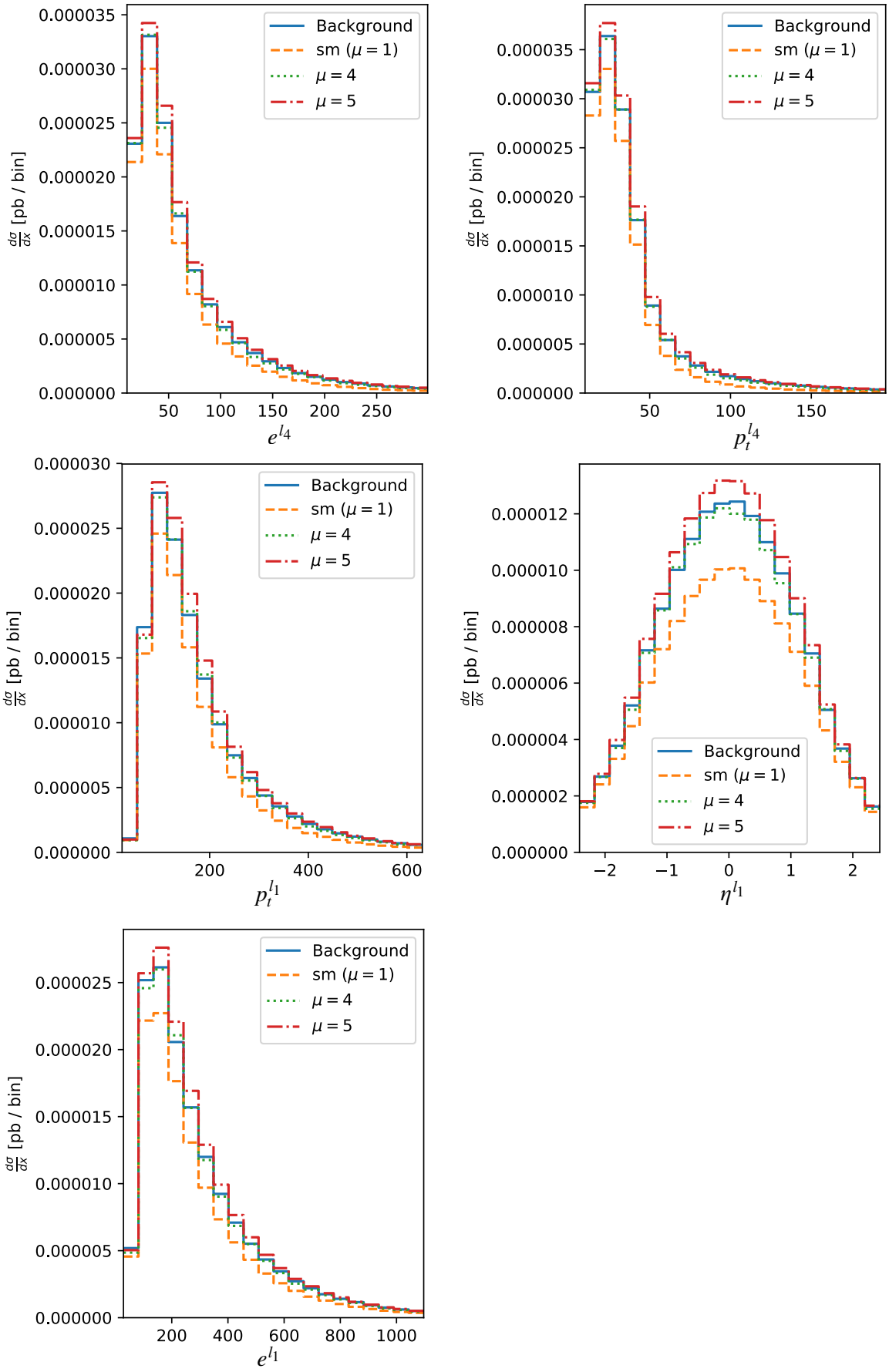


Figure 7.14 – Distributions of observables used in this study at $\mu = 0, 1, 4, 5$.

7.6.1 Training SALLY

Out of the two models that were investigated in detail in this study, the first, "SALLY" (**S**core **A**pproximates **L**ikelihood **L**ocall**y**), is trained to regress the joint-score at the SM point, $t(x, z|\mu_{SM})$. It is not parameterised on μ and therefore its output can be treated like a regular observable, just like in the usual case of training an ML classifier. It is only aware of how the physics changes are the local neighbourhood of the SM and is therefore only expected to perform well in that neighbourhood. The advantage of course is that it does not need the notion of signal and background classes to be trained.

It was trained on 2×10^6 events only from the SM point for 50 epochs with early stopping. These trainings took 12 mins on a CPU. The neural network has 100 nodes in the hidden layer with hyperbolic tangent activation and an output node with linear activation. Minimal hyper-parameter optimisation was performed for this model, only the width, depth of the network and activations (hyperbolic tangent vs rectified linear units) were explored.

7.6.2 Training ALICES

The second model, "ALICES" (**A**pproximate **L**ikelihood with **I**mproved **C**ross-entropy **E**stimator and **S**core), is also trained on the joint-score but in addition it is trained on the joint-likelihood ratio, $r(x, z|\mu_{SM}, \mu_1)$ for various test hypotheses, μ_1. ALICES is therefore aware of how the physics changes at parameter points far away from the SM, and requires data from various values of μ to be trained well.

The ALICES model was found to be very sensitive to the training dataset. Training on a random uniform distribution of parameter values resulted in the model being well trained only for some values of μ. The parameter points at which it was trained had to be fixed by hand for improved results. The parameter points used for training are,

$$\mu = \{0, 0.5, 0.7, 0.8, 0.9, 0.95, 0.98, 1, 1.02, 1.05, 1.1, 1, 2, 1.5, 1.8, 2, 3, 4, 4.5, 5, 5.5, 6, 7, 8, 9, 10, 12, 16\}.$$

These values were chosen heuristically after evaluating the performance of the model at different values of μ and verifying the model's ability to interpolate to values in between.

ALICES was trained on 2×10^6 morphed samples (which includes re-sampling the same event for different values of μ, with probability appropriate to the value of μ, as described in Chapter 4) for 30 epochs with early stopping. These trainings took 45 mins on a CPU. The neural network has 300 nodes in the hidden layer with hyperbolic tangent activation and a single output node with linear activation. The width of the network and the number of epochs were optimised.

7.6.3 Comments on stability

For both SALLY and ALICES, the training (with the same architecture) had to be preformed a few times to obtain the best model, suggesting that there is room for optimisation to make the learning more stable.

More extensive hyper-parameter optimisation was performed for some of the other models (SCANDAL, ADAPTIVE-SALLY) that were also listed in [66]) but they failed to provide consistent results and were therefore not used for the final round of this study, therefore details about these models is not given here. If tricks to stabilise these models are uncovered in the future, they may be included in the next round of this study.

7.7 Inference and Evaluation of Results

This section describes the results obtained from these two models that were trained using the augmented data from the simulator and compares them to fits using the distribution of some typical observables. The objective in each set of comparisons presented below is a precise single parameter estimation, that is to measure μ with as small an uncertainty as possible. The networks that directly output the likelihood ratio directly (ALICES) could of course provide unrealistically confident measurements if the model is not well trained, and this was seen during model optimisation, therefore their performance needs to be evaluated on multiple datasets representing different true values of μ (the model outputs could be calibrated to ensure they are never overconfident, but it is an inelegant solution that is best avoided). There is no such fear for SALLY because the output of the model is binned and treated like an usual observable for a fit.

7.7.1 Asimov Dataset

To assess the sensitivity of an analysis strategy for a measurement where only a few events are expected to be observed, the act of the this measurement needs to be performed multiple times on many "toy" observation datasets, each with only a few events. These toy datasets are generated based on a given theory (often the SM), and the distribution of the measurements on these toys can be used to compute the median expected measurement and its uncertainty. This is computationally expensive.

The ensemble of simulated experiments can be replaced by a single representative one, the "Asimov" dataset [88]. It is a dataset upon which unbiased measurements yield exactly the correct theory parameters and upon which the median expected sensitivity of an analysis can be estimated along with its fluctuations. In practice we cannot have perfectly Asimov datasets, but a very large simulation can approximate an Asimov dataset.

Since in practice, unlike the real observed data, the Asimov dataset has a large number of events, the statistical uncertainty on the measurement is estimated differently. Instead of the quadratic sum of weights, they are calculated as $\sqrt{N}$ where $N = \sum W$, and this provides a realistic expected uncertainties.

7.7.2 Inference on one Asimov Test Dataset

Just like the training dataset, the Asimov test datasets are created for this evaluation by morphing the data generated at the benchmark points that were reserved for evaluation. They allow to estimate the expected sensitivity for the various inference strategies. This is a great advantage of this morphing technique over the strategy used in Chapter 6 (where evaluation of the model was restricted to a few parameter points) even if no further simulator assisted learning is performed.

7.7.2.1 ALICES Inference

Described below is the inference steps for ALICES for one given Asimov test dataset (for example a dataset created at $\mu = 1$).

At inference time, the inputs of the ALICES neural network, for a given event, are the measured observables of the event, as well as the hypothesis being tested (i.e. one particular value of μ). There is no additional maximum likelihood fit to be preformed for this model, it provides

the inference directly. The output of the network is the (log-)likelihood ratio between the test hypothesis and the null hypothesis (i.e SM with $\mu = 1$). For example, the first test hypothesis to be evaluated is $\mu = 0$, therefore for each sample in the test dataset, all the observables along with the test hypothesis $\mu = 0$ is given as input to the network, and it provides the log-likelihood ratio for this event between the hypothesis $\mu = 1$ and $\mu = 0$. The sum of the outputs (the log of the likelihood-ratio can be added) for all the events then provides the log-likelihood ratio for the entire test dataset between these two hypothesis. An additional component to the total log-likelihood ratio coming from the rate information (information regarding number of events observed vs the number of events expected under the two hypothesis) may also be added here.

Next, the entire process is repeated for a new test hypothesis, for example $\mu = 0.1$, to get the log-likelihood ratio between the hypothesis $\mu = 0.1$ and $\mu = 1$ for the same test dataset. This process is repeated to scan over a range of μ. The minimum of them can be subtracted off from the rest to bring the minimum to zero in the log-likelihood ratio plots that will be shown. This information can of course be converted into a scan of the p-value (which represents here the probability of obtaining data at least as extreme as the test dataset, assuming that the test hypothesis is correct), and these plots will also be shown.

A fear is that perhaps ALICES simply learns to provide a high likelihood ratio when the test hypothesis is $\mu = 1$ and very low values otherwise, regardless of the distribution of the shape of the observables. In this case it would appear to outperform all other techniques on the SM Asimov dataset while its actual performance is no better than a broken tape-recorder. A final step in the evaluation is needed.

For the final step, this entire process is repeated for a new Asimov dataset created at another parameter point, for example $\mu = 4$, to see if ALICES predicts the correct value of the parameter, and still provides a small uncertainty. This test is performed at many values (with morphing it is cheap to perform new test datasets) to evaluate the performance everywhere.

7.7.2.2 Histogram/SALLY Inference

The log-likelihood ratio and p-values for the histogram techniques is calculated using multi-binned Poisson likelihood fits using template histograms of particular physics variables (such as the invariant mass of the four leptons or transverse momentum of the first jet). An example is shown in Figure 7.15. The same is also done with the output of SALLY.

7.7.3 Comparison of the results

Two sets of comparisons will be shown, one including the rate (total yield) information and one set without. This allows to differentiate whether the sensitivity comes from just the cross-section or the shape of the distributions of the observables. Since the dominant background to these processes $q\bar{q} \to ZZ$ is expected to have a strong effect on the total number of events measured and add an uncertainty to the normalisation, it is useful to look at the difference in results based on only the shape information.

Although the typical negative log-likelihood curves are shown in this study, the discussion will be based on the equivalent information seen in the p-value scans. Rather than the usual parabolic shape, the negative log-likelihood curves have two local minima because of quantum interference effects, and these features were found to be visually easier to follow in the p-value scans.

A p-value scan is shown for Asimov test datasets corresponding to $\mu = \{4, 2, 1\}$ in Figure 7.16. It compares a traditional 1-dimensional histogram fitting approach using three observables, the mass of the four leptons (m_{4l}), the transverse momentum of the leading jet (pT_{j1}) and the

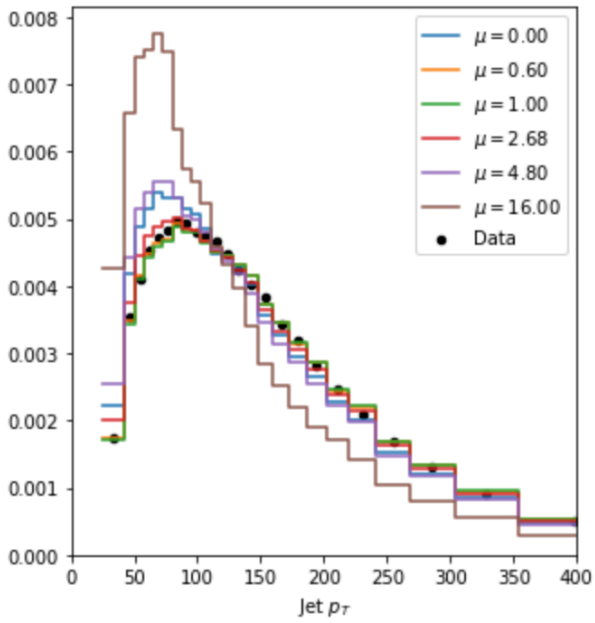


Figure 7.15 – Example of template fit using histograms of a physics variable.

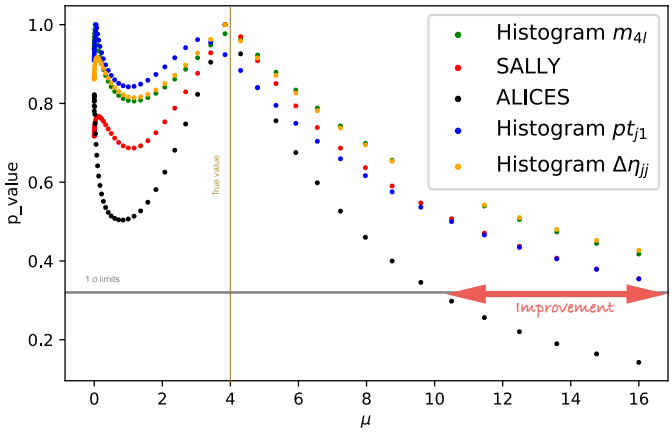
difference in the pseudo-rapidity between the two jets ($\Delta\eta_{jj}$) with two "physics-aware" neural network approaches to measure μ on two Asimov (i.e. representative) test datasets. The first three are treated as a baseline for the latter two techniques. The key point in interpreting these figures is that the sharper the curve, the better the technique. In particular, the point at which the curve crosses the 1 σ horizontal indicates the expected 1 σ upper-limit to be set by the given technique.

SALLY performs better than traditional physics variables near the SM value ($\mu = 1$) as anticipated but quickly deteriorates far away from it. In Figure 7.16c is very slightly better than the histogram techniques (but the differences fade quick for μ values away from the SM point). For a test dataset generated at $\mu = 4$ in Figure 7.16a, although SALLY more confidently excludes the SM than the three baseline and also has a much lower second peak (near 0) compared to them, it does not perform much better than the them at setting upper limits on μ. For values around the SM, like in Figure 7.16b SALLY outperforms the baseline to a considerable extent and does not provide a second peak the way the baseline techniques do.

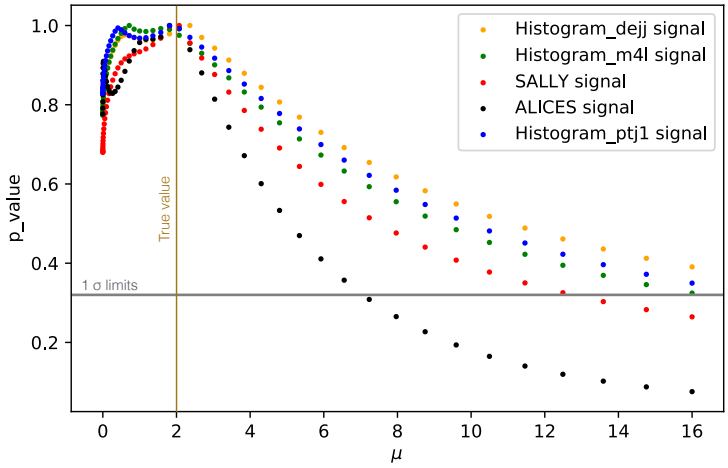
ALICES, however, is aware of physics in the entire range of μ, and therefore more confidently excludes wrong values of μ for the test dataset generated at the SM, at $\mu = 2$ and at $\mu = 4$. This was found to be true for various values tested. Since it is aware of how the physics observables change their distributions at different parameter values, it can automatically re-optimise the analysis for each test hypothesis (rely on the appropriate observables for the appropriate test hypothesis), and therefore it is the best technique at being able to break degeneracies such as the second peak in Figure 7.16b. It also performs the best for almost the full scan range also in Figure 7.16a. Since it 'knows' the physics at other points, it can therefore also exclude them with more confidence for the test performed at the SM value in Figure 7.16c.

The 1σ limits from ALICES is consistently better than all other techniques for Asimov test datasets generated at any point of μ in this study. The comment remains true for the more standard 2σ limits that can be inferred from Figure 7.18.

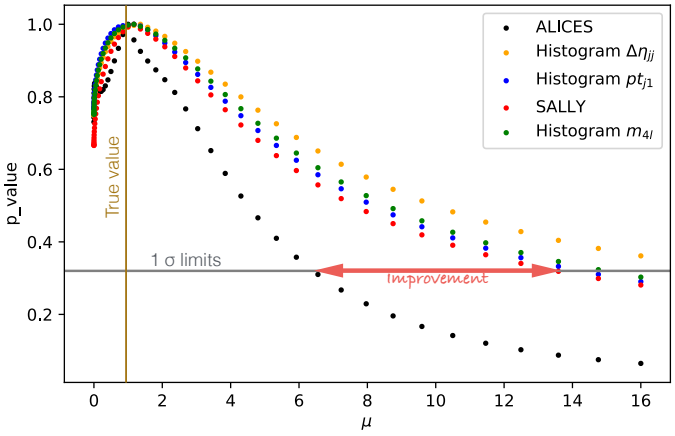
Negative log-likelihood curves with and without the cross-section information are also shown for Asimov datasets at various values of μ in Figure 7.17. Although not clearly visibly in every case, many of these curves have two local minima (they almost merge to give a flat appearance to the negative log-likelihood curve) corresponding to the degeneracy problem discussed



(a)

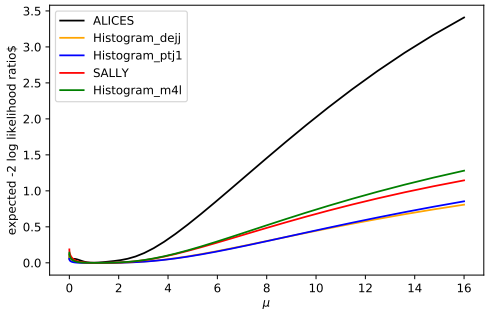


(b)

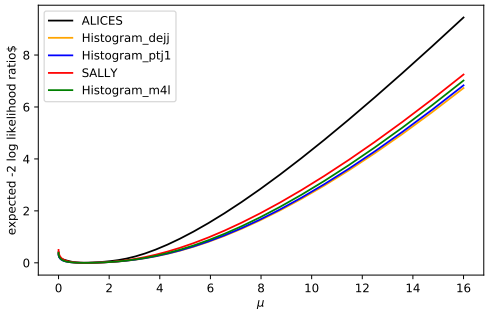


(c)

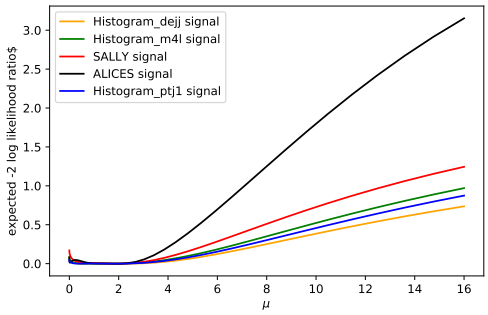
Figure 7.16 – p-value scans for Asimov test dataset generated at (a) $\mu = 4$, (b) $\mu = 2$, and (c) standard model ($\mu = 1$) for a luminosity of 36 fb^{-1} where the true value is indicated with the golden vertical line and the 1σ limit threshold indicated by the grey horizontal line. The more standard 2σ limits are not visible in these comparisons but can be inferred from Figure 7.17 and Figure 7.18.



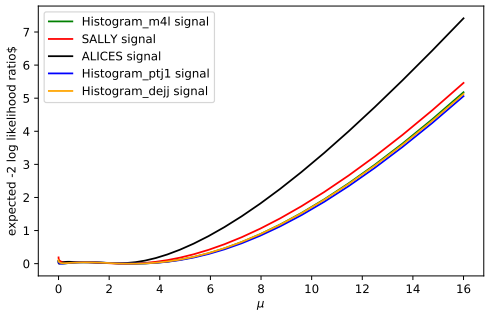
(a) SM, without rate



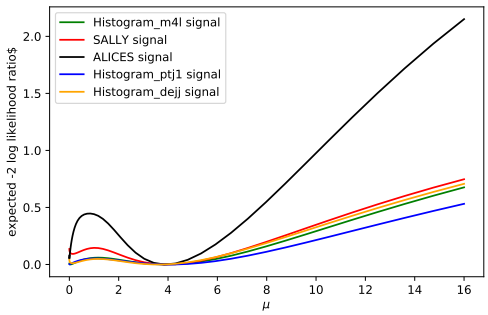
(b) SM with rate



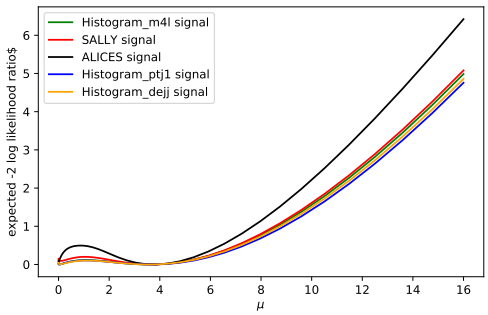
(c) $\mu = 2$, without rate



(d) $\mu = 2$ with rate

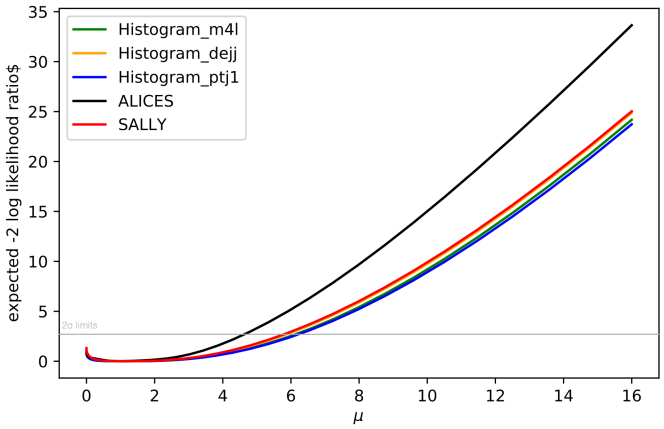


(e) $\mu = 4$, without rate

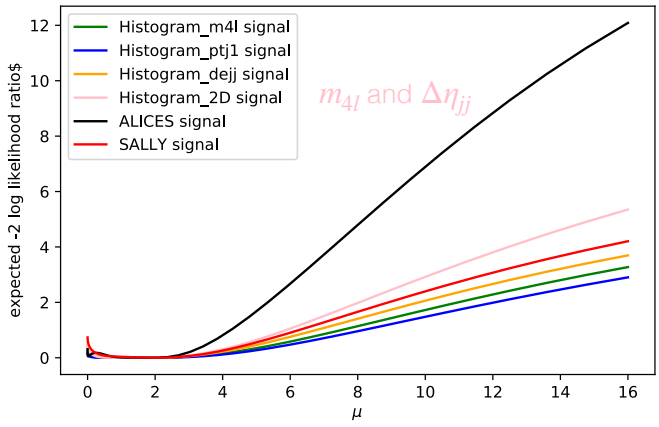


(f) $\mu = 4$ with rate

Figure 7.17 – Negative log likelihood curves for Asimov datasets generated at $\mu = 1$, $\mu = 2$, $\mu = 4$ with and without using the total cross section (rate) information.



(a) SM without rate for 139 fb^{-1}



(b) SM, without rate, lumi 139 fb^{-1}

Figure 7.18 – Negative log likelihood curves for an integrated luminosity of 139 fb^{-1} for Asimov datasets generated at (a) $\mu = 1$, (b) $\mu = 2$ and demonstrating the performance of using 2-dimensional histogram templates (pink) in comparison to the 1-dimensional histogram templates and ML techniques.

in section 7.1). Two local minima were also observed in ATLAS studies performed on ggF and VBF events together with the optimised Matrix Element based observable (defined in Chapter 6 Equation 6.3) as can be seen in Figure 7.19[4].

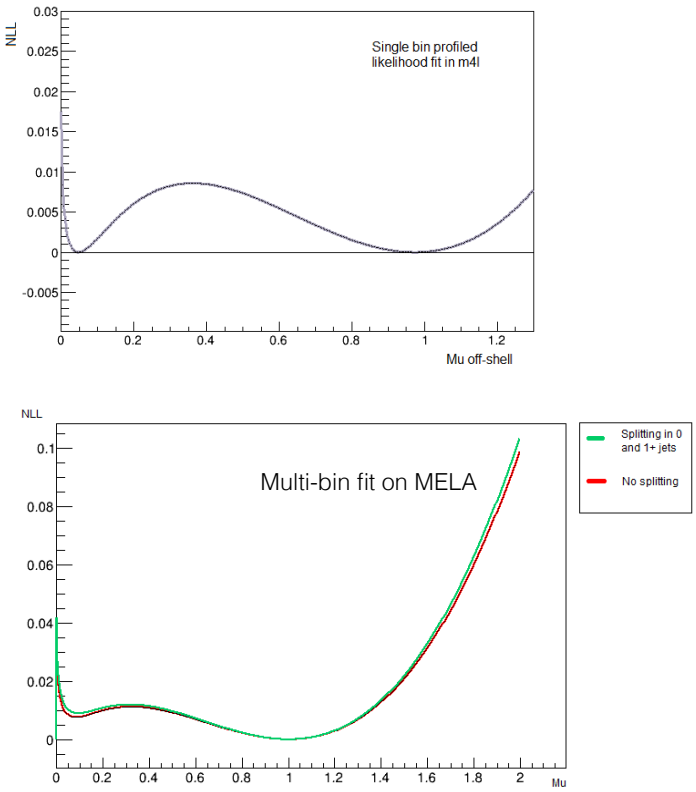


Figure 7.19 – Expected Negative Log-Likelihood curves using (Top) a single bin (Bottom) multiple bins of MELA (Matrix-Element based observable) and category splitting based on the number of jets for the ATLAS off-shell analysis using both ggF and VBF events simulated from the SM. The two local minima structure remains even for an optimised observable.

These figures demonstrate the benefit of an analysis strategy that is optimised for all values of μ simultaneously, rather than optimising the analysis by creating a very sensitive 1-D observable, even when a machine learning algorithm (such as SALLY) is used to create that observable.

Given that this analysis expects to set upper limits on μ quite far away from the SM value, SALLY would be far less useful compared to ALICES. On the other hand if there is a discovery to be made ($\mu \neq 1$), ALICES will get us there much sooner than the current approach of optimising the analysis only at the SM.

Conclusion and Future Outlook

A study was performed to investigate a new family of machine learning algorithms that could be used for the off-shell Higgs to four leptons analysis in the ATLAS experiment at CERN. These techniques leverage the use of very accurate simulators in particle physics, to extract additional information that is very useful in learning the likelihood ratio between a test hypothesis and the null hypothesis. They also avoid the need to define 'true class labels', a concept that is ill-defined in the presence of quantum interference between signal and background processes.

[4]Study and figure made by Samyukta Krishnamurthy.

The results demonstrate a considerable improvement in performance using the new machine learning technique compared to traditional methods of building a single highly sensitive observable, however, a study using full ATLAS detector simulation, the inclusion of all other signal and background processes and a comparison to the current ATLAS baseline strategy has yet to be performed.

The study was performed only for the Vector Boson Fusion produced Higgs process, which interferes with Vector Boson Scattering process in the high mass off-shell regime. The processes that need to be included are the $q\bar{q} \to ZZ$ background and perhaps the $gg(\to H) \to ZZ$ process. The two jet requirement could also be lifted. In this case a dedicated BDT to first remove non-interfering background events will help improve the sensitivity of this inference technique.

The SALLY model, which is trained to be optimal only near the neighbourhood of the SM performs well for some points in that neighbourhood but will not be ideal for an analysis that is expecting to set upper-limits very far away from the SM value. One of the reasons the performance of SALLY does not scale well to other points in μ for this analysis might be because physics changes extremely quickly near the SM point. This was also observed during the data creation period, where it was seen that data generated at the SM value does not morph well to other points.

Considering that interference brings in non-linear effects for a signal strength measurement problem, the improvement in sensitivity using ALICES comes not only from training on additional information but in fact being able to parameterise the model for different values of μ. The algorithm can effectively re-optimise the analysis to be sensitive to the changing physics, something a one dimensional observable will not be able to match. This would be particularly useful compared optimising the analysis only using SM datasets given how fast the physics changes near the SM value.

This technique could also be studied for the $gg(\to H) \to ZZ$ processes for the measurement of the off-shell signal strength in the gluon-gluon fusion production mode, which also has quantum interference between signal and background processes and in addition has much higher statistics at LHC running at a centre of mass energy of 13 TeV. Although including the $gg(\to H) \to ZZ$ as a signal process will improve the statistical power, the interpretation of such results for a Higgs width measurement will become very model dependent [40]. The limit set on $\mu_{\text{offshell, VBF}}$ can more readily be interpreted as an upper limit on the Higgs boson width Γ_H in units of Γ_H^{SM} due to Equation 2.59.

These models are trained on datasets generated with different values of the signal strength with the assumption that the rest of the physics behaves as described by the SM. It will also be worth testing the analysis strategy on a simulated sample where some New Physics modifies the apparent off-shell signal strength to assess the generalisability of the strategy. This will be of less concern when ATLAS moves from the κ framework to an EFT framework for interpretation, and it will make the EFT based likelihood-free inference with ALICES also simpler.

Using such techniques in a full ATLAS analysis for the first time, including all the systematic uncertainty checks and combination of results with the $\nu\nu$ channel, will still require significant amount of additional effort. The ATLAS HZZ software has now been adapted[5] to extract and propagate the additional weights from the event generator through the entire analysis chain. However, the statistical inference tools will also need to be updated to handle a network that is parameterised on the hypothesis being tested. The `pyhf` [128] package is expected to support such inference in the near future. In case not all systematic effects can be trained on with ALICES, the output of the network could be binned into multiple histograms for a discrete set of test hypotheses μ. This would allow the usual studies of how sensitive the output of the

[5] Thanks to RD Schaffer.

network is for variations systematics. Although not a concern for the $H \to 4l$ analysis, binning the output might also be useful in using this technique for an analysis where a data driven background estimation needs to be performed[6].

As a followup to this work, model building (in `FeynRules`), event generator and morphing strategies are being investigated to this study for an off-shell ggF signal strength measurement as well within ATLAS. The ATLAS research group at University of Massachusetts has joined the effort of bringing `MadMiner` based inference to the ATLAS off-shell Higgs boson coupling measurement analysis in the four leptons decay channel.

A first application of these techniques in ATLAS will carve the way for several other analyses to follow. The most straightforward application being Effective Field Theory studies, but in fact any analyses that deals with quantum interference may benefit from these parameterised inference models, for example the $HH \to \gamma\gamma bb$ di-Higgs search [129] in the ATLAS experiment.

[6] a more involved solution might be to use a ML based correction of MC to data using side-bands.

Chapter 8

The Aspiration Network

Contents

Apart from the studies detailed in the previous chapters of this thesis, contributions were also made to certain side projects. A first study was made on using Optical Processor Units [130] for fast machine learning in the context of HEP (tracking and classification from raw calorimeter images or low level detector information). A new adversarial training algorithm was also developed for work related to the four leptons analysis detailed in Chapter 6, and this will be summarised below.

8.1 Aspiration Network

Often times using ML models to optimise selections or tagging of jets results in a sculpting of a particular observable such as the mass of a particle, which might be undesirable. This may also happen for cut-based selections, although usually to a lesser extent. Considerable work has gone into decorrelating the relevant mass from the output of the model in ATLAS [131]. A popular solution is to use adversarial training [132].

This section will describe the interest in mass decorrelation in the context of the off-shell Higgs to four leptons analysis, describe in brief how the usual adversarial training technique completely fails for this dataset, and finally propose a modified adversarial algorithm where the model is given information about the aspired distribution and thereby makes the learning task easier for the neural network model. The end of the section will show the new training algorithm succeeding in solving the problem at hand, where the usual adversarial training failed.

8.1.1 The Mass Line-Shape

The mass line-shapes shown in Figure 2.10 and Figure 7.3 (mass of the four leptons, m_{4l}) are a crucial aspect of the off-shell measurements, because they are changed in subtle ways due

to BSM effects as detailed in Chapter 2. The discriminating power of a classifier is often a function of the mass of the four leptons. As can be seen in Figure 8.1, the background is a falling distribution in m_{4l} while the signal falls much more slowly, making an event with high m_{4l} a better signal candidate. A selection based on traditional classifier will therefore distort the line-shape and even make re-interpretations more difficult. The following discussion is with regard to a classification task to separate the VBF full process events from all other processes, which was one of the two classification approaches investigated in Chapter 6.

Figure 8.2 shows the correlation between the classification score and the m_{4l} for a BDT trained with `XGBoost`. It provides a significance of $Z = 1.87$. Neural networks show very similar issues. The default neural network architecture used in these studies is a feed-forward dense neural network with 4 layers, each with 32 nodes and ReLu activation apart from the last layer which has 1 node and a Sigmoid activation function trained with an *Adam* optimizer.

The objective is to train a model with maximum sensitivity, yet independent of m_{4l}.

Dropping m_{4l} from the input variables results in a drop in the significance to $Z = 1.71$ without mitigating the mass dependence, as seen in Figure 8.3 (the model trained in this example is a neural network). Expressive models can infer the mass through its correlation to other input variables, so dropping the mass variable alone will not necessarily remove all correlation of the output of the model with the mass.

8.1.2 Trouble with Pivot

The original 'pivot' adversarial [133] method was proposed to make a classifier invariant to a given systematic (such as jet energy scale). An idea to use this technique to make a classifier invariant to the mass of a particular object was proposed [132] soon after. The architecture shown in Figure 8.4. The idea is to train an adversarial network to regress the mass only from the output of the classifier. The the classifier output is correlated to the mass, this can be done, and the classifier will be penalised for it. The hope is that eventually the classifier will learn to optimise two tasks simultaneously and perform the best job possible of classification of signal vs background while at the same time remaining invariant to the mass. The loss of the classifier therefore has two terms,

$$L_{\text{total}} = L_{\text{classification}} - \lambda L_{\text{adversary}} \tag{8.1}$$

the first one for classification and the other is the negative of the loss of the adversary (the second term is very similar to a the generator of a GAN which has to fool the discriminator). The λ is a hyper-parameter which decides the relative importance of the two tasks. This is the typical ML based solution use in ATLAS for mass decorrelation.

This technique has been proven to work well to remove relatively obvious background mass sculpting, but failed to decorrelate m_{4l} from the output of the classifier for off-shell signal events despite a large hyper-parameter search and despite consultation with the authors of the original pivot paper [133] and the mass decorrelation paper [132]. For this dataset, the adversary was unable to regress the mass (see Figure 8.5), even though the correlation existed. One of the several suggested architectures involved simplifying the problem for the adversary by binning the mass (as was the case for the adversary in the mass decorrelation paper), this changes the task from a regression to a classification into one of few bins for the adversary. Another suggestion was using a Gaussian Mixture Model (GMM) as an output of the adversary (this was used in the original version of the pivot paper as well as later on in ATLAS mass decorrelation studies [131]). Unfortunately, the adversarial network still completely failed to learn the correlation in both these cases, which are subtle in a scatter plot (see Figure 8.6) but is visible clearly in a profile plot (similar to a scatter plot but to reduce the noise, a binning is performed in the x-axis, and the mean is represented in the plot, as seen in Figure 8.2). The subtle, noisy correlation with

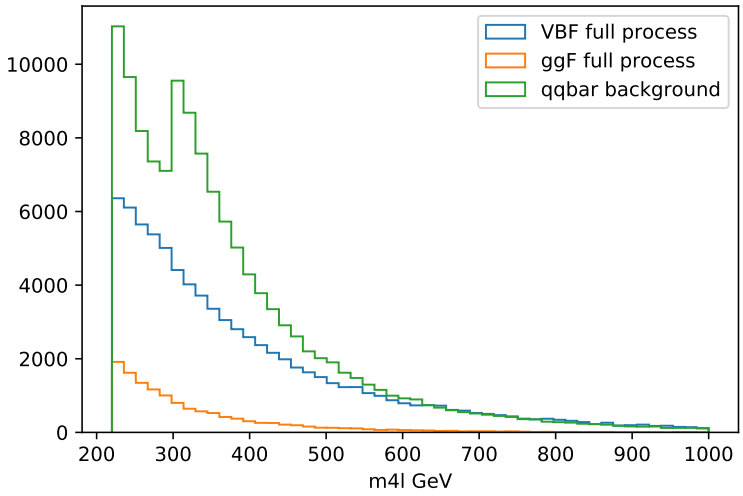


Figure 8.1 – Distribution of m_{4l} for VBF full process, ggF full process, and $q\bar{q}$ background process in the VBF region.

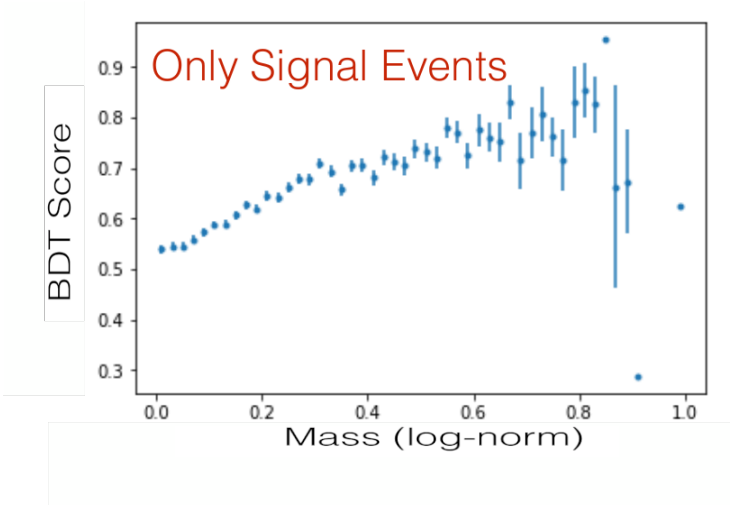


Figure 8.2 – Distribution of score of a BDT (trained with m_{4l} as one of the input feature) for signal events as a function of the (log-norm) mass.

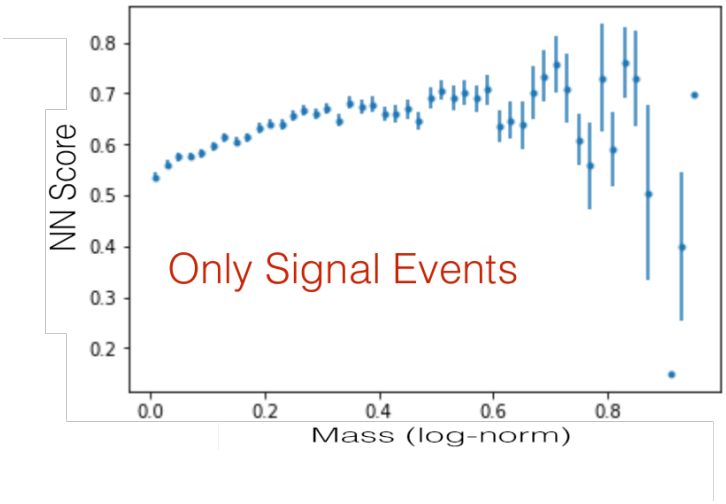


Figure 8.3 – Distribution of classification score for signal events as a function of the (log-norm) mass for a neural network trained without m_{4l} as an input feature.

the line-shape is difficult for the adversary to learn, and therefore the classifier could not be forced to decorrelated its output with respect to m_{4l}.

An investigation was performed keeping the classifier frozen, to see how an adversary could be trained to learn the correlation with the mass. Neural networks failed this regression task, as seen in Figure 8.5. Interestingly, even BDTs had trouble learning this correlation. The BDTs with close to default hyper-parameters failed a regression task, therefore the problem was simplified considerably into a two class classification. The entire mass range was binned into only two bins, which became the two target classes. Despite the simplification, BDTs with various hyper-parameters achieved AUCs of between 0.5 and 0.55. Further, the BDT hyper-parameter configurations that achieved close to a 0.55 AUC suffered from incredible overtraining, evidenced by the fact that their AUC on the training dataset was 1.

Some success could be achieved with the use of extremely deep BDTs trained with the AdaBoost algorithm implemented in the `SciKitLearn` package [134]. Figure 8.8 shows the regression performance of a BDT with 40 estimators and a max-depth of 60 (the usual max-depth of BDTs is around 3). It achieved a Mean Squared Error of 0.0088 compared to 0.176 for the dense neural network corresponding to Figure 8.5. Feed-Forward Deep Neural networks failed to emulate the performance of this BDT for a wide variety of architectures that were tried.

8.1.3 Learning Aspirations

A new adversarial training architecture was designed which makes the learning much easier for the adversary. When an aspired distribution is known, it only makes sense to give this additional information to the network[1]. Here the aspired distribution is the joint distribution of the classifier output and the m_{4l} observable where the two are totally uncorrelated. Since this is the ideal distribution we *aspire* to have, we refer to it as the "aspired distribution", the corresponding dataset is then referred to as the "aspired dataset", and the training strategy is referred to as "aspiration targeted learning".

In this case, the adversarial network has two input features, the mass of the four leptons and the output of the classifier. Building the aspired distribution requires a fully decorrelated joint distribution between these two features without modifying their respective marginal distributions. This task was discussed already in Chapter 4 with regard to Permutation Importance (PI). Borrowing the trick used in PI, the aspired dataset is built by shuffling one of the two features. This means that the mass for one event is now paired with the classifier output for a random event, explicitly breaking any possible correlation.

The adversary now has to solve a classification task with a two dimensional input. The two classes are,

- The Real Distribution: It consists of the output of the classifier for an event and the m_{4l} of that same event,
- The Aspired Distribution: It consists of the output of the classifier for one event and the m_{4l} of a randomly chosen event (without replacement).

It is preferable to shuffle rather than randomly sample from some idealised probability distribution because shuffling ensures that the marginal distributions remain identical for the real dataset and the 'aspired dataset'. This choice also removes the possibility of overtraining on the unphysical differences in marginal distributions. This training strategy is illustrated in Figure 8.9.

[1] The reader may notice that a connection can be made to the growing field of Causal Inference.

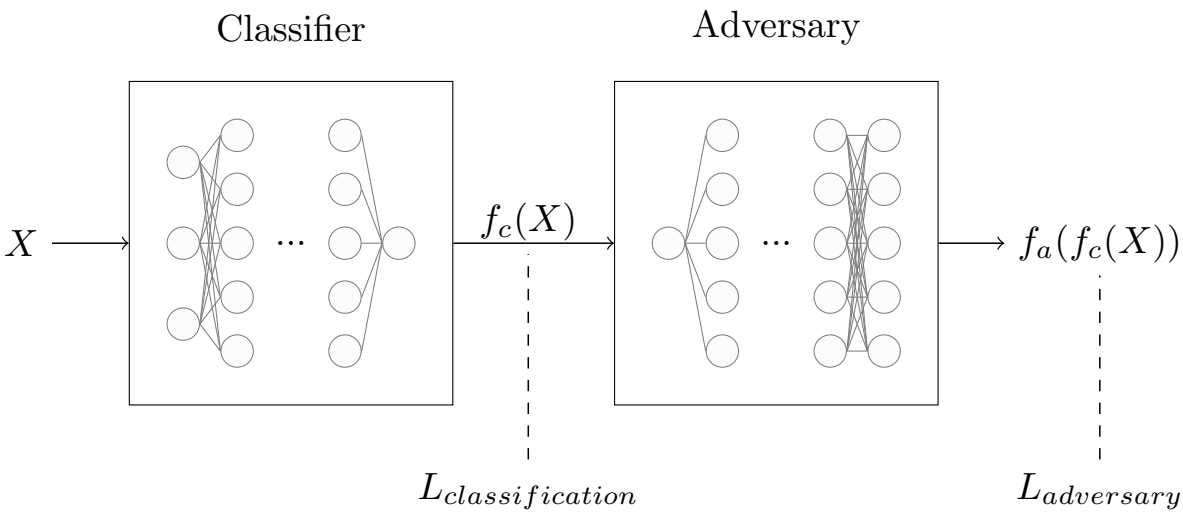


Figure 8.4 – Architecture of the adversarial neural network training strategy. The classifier (D) distinguishes signal from background using input features X, the adversarial network (R) attempts to predict the invariant mass using only the output of the classifier, $f_c(X)$. Instead of a continuous regression, the task for the adversarial network is to categorise each event into one of several bins in invariant mass. [132]

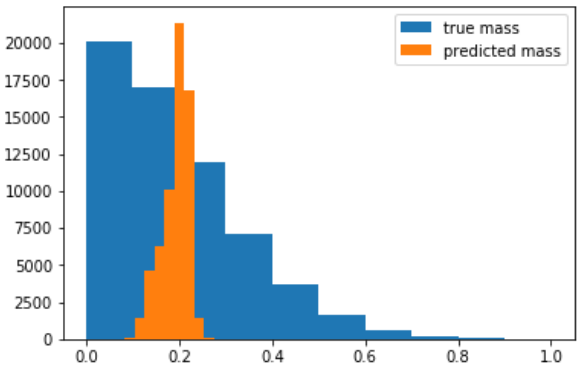


Figure 8.5 – Regression of m_{4l} from the output of a classifier using a dense neural network.

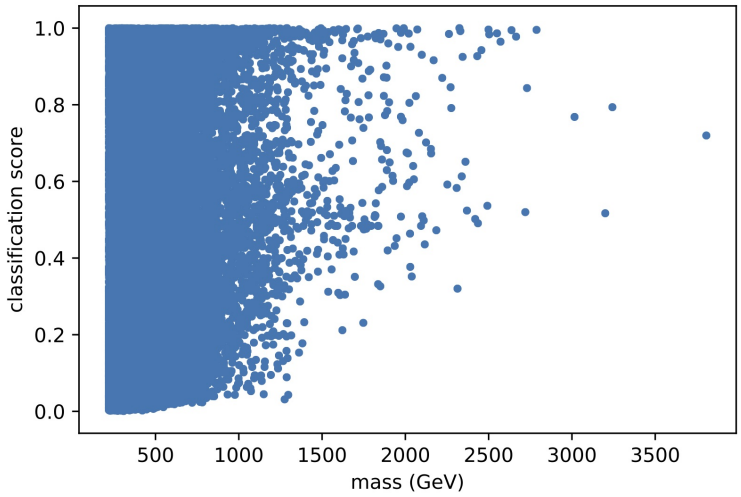


Figure 8.6 – Scatter plot of the classification score vs mass. The noise level remains similar whether or not a log-norm of the mass is taken.

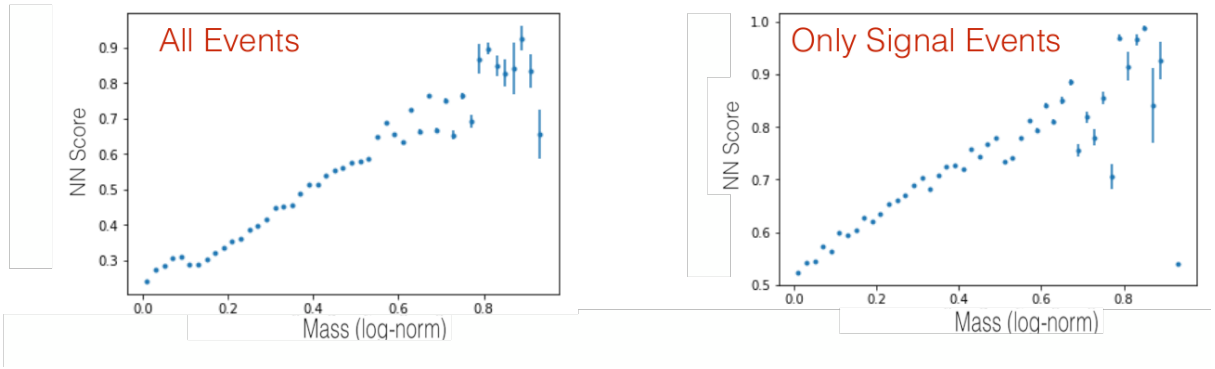


Figure 8.7 – Profile plot of the classification score as a function of the (log-normed) mass for a traditional neural network classifier for (Left) all events, (Right) Signal Events.

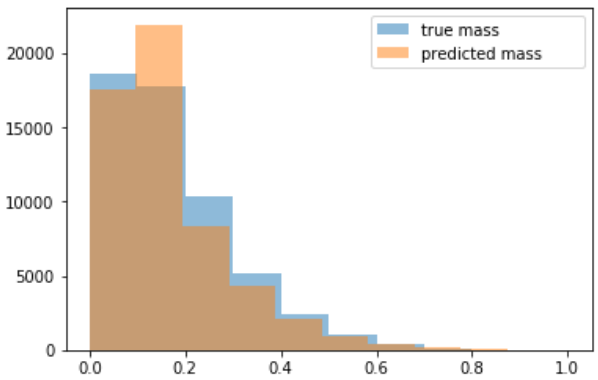


Figure 8.8 – Regression of m_{4l} from the output of a classifier using a BDT with 40 estimators and max-depth of 60.

The two dimensional input space improves training stability and performance. Neural networks are known to optimise better on larger input space dimensions, which provide multiple paths to multiple minima and easy escapes from plateaus. Traditional least squared regression often outperforms its SGD counterpart on a single neuron. Unlike in the pivot case where the mass distribution is only given to the network from back-propagation, here the network receives it as a regular input feature. Giving 'additional information' in the form of the aspired distribution may also help the adversarial network learn better.

Similar to the pivot algorithm, there is a tuneable hyper-parameter λ also for aspiration targeted learning which weights the adversarial loss with respect to the classification loss. Unlike the pivot algorithm however, there is no need for multiple λ parameters in case the classification needs to be invariant to more than one feature. While the pivot algorithm requires the adversary to perform two regression tasks in the case of invariance to two features, the adversary in this strategy continues to perform a simple binary classification, but with three or more inputs instead of the usual two. In principle with more input features, the learning becomes easier for the adversary. Given that multitask learning makes tuning these hyper-parameters quite tedious, this is quite a useful bonus for the aspiration network.

8.1.4 Mass Decorrelation

The new architecture worked out-of-the-box for off-shell dataset, both networks with the 'default' hyperparameters of 4 dense layers with and 32 nodes each and ReLu activation apart from the last layer which has one node and a Sigmoid activation. A loss weight of $\lambda = 3$ was used for the adversarial term.

Although the algorithm worked quite easily and remained stable for various hyperparameters, a good version of the classifier had to be picked, one which exhibited the desired decorrelation. This is usually the case for adversarially trained networks, where convergence is not guaranteed. In this case it was found to be a very mild inconvenience because in practice a good iteration could be found easily. One had to simply run the training for a few more iterations (51 more batches with a batch size of 128 was used) in case the output of classifier was not flat with respect to the mass. This 'epoch picking' can easily be automised. Figure 8.10 shows the decorrelation performance of the aspiration network, which can be compared to an ordinary classifier in Figure 8.7. Figure 8.11 shows the threshold classification score corresponding to a 80% and 90% signal efficiency as a function of $\log m_{4l}$ where the latter is normalised. This is the figure that determines the usefulness of the technique. The aspiration network was able to satisfactorily decorrelate the mass and the classifier output.

The adversarial loss can be applied selectively on signal events with masking of the loss for background events, and it often leads to slightly better classification performance. Although this idea is also applicable to the aspiration network, in this case the training was found to be more stable without masking, and at no cost to the classification performance. The training stability issues come from the fact that there are far more number of background samples than signal samples in the training dataset, and masking the loss for background events results in large fluctuations in the effective batch size.

Such adversarial training usually requires a trade-off between pure classification performance and decorrelation, however, in this lucky test case, the trade-off could be avoided. The discovery significance for the traditional classifier was found to be $Z = 1.85$ at a threshold score cut of 0.9, whereas for the aspiration network it was found to be $Z = 1.87$ (where the last digit fluctuates due to uncertainty) at a threshold of 0.9. The change in permutation importance for each feature, and in particular the drop in importance for the m_{4l} can be seen in Table 8.1. It appears that other features become more important as a result.

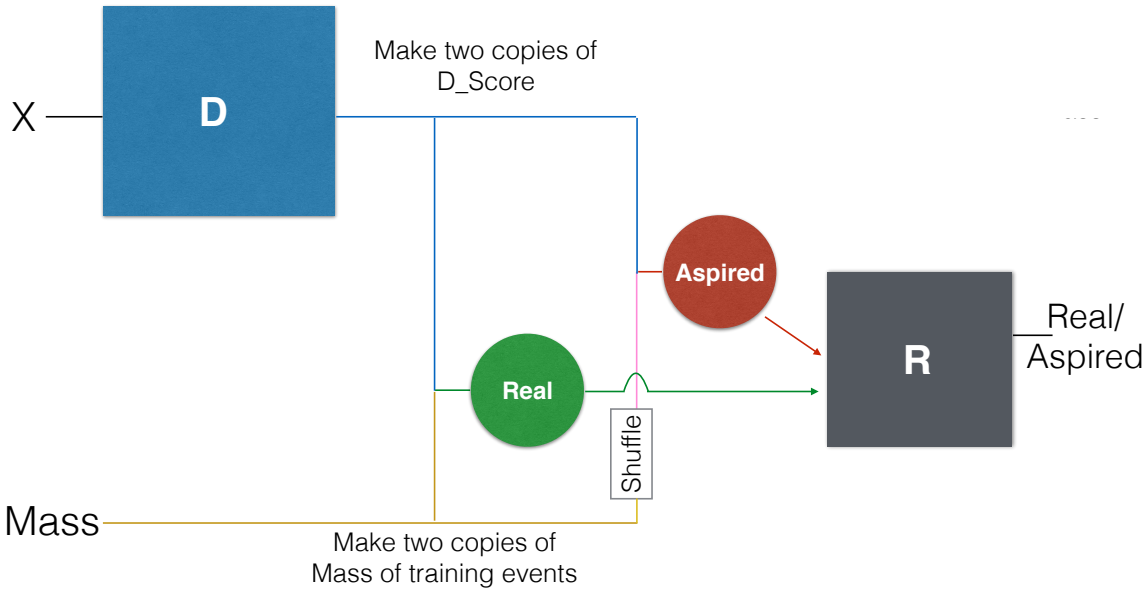


Figure 8.9 – Architecture of Aspiration Network: The classifier distinguishes signal from background using input features X. The adversarial network takes two input features and performs a classification of real vs aspired (correlated input features vs fully decorrelated input features) distribution instead of a regression of the mass.

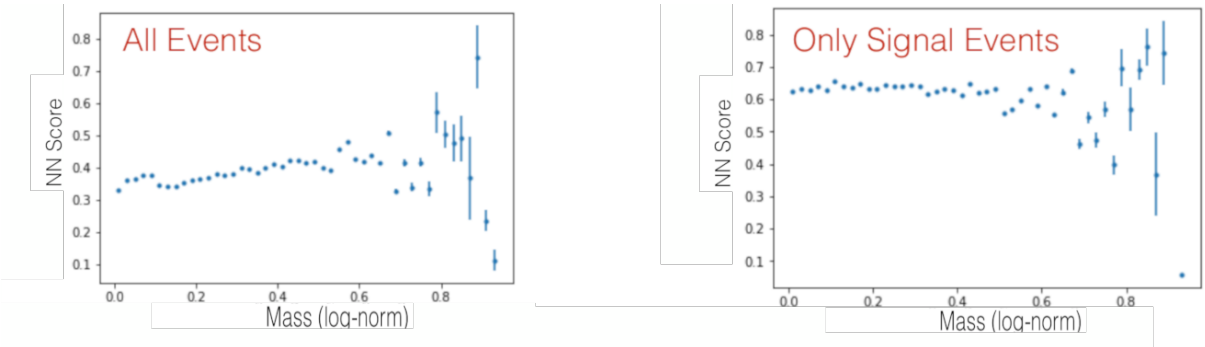


Figure 8.10 – Profile plot of the classification score as a function of the (log-normed) mass for a classifier trained with aspiration targeting. The distribution is shown (Left) for all events, (Right) for signal events only.

Such luck runs out when this algorithm is used to make the classification invariant to a different features instead of m_{4l}. The feature studied is the MELA variable. In this case, a $Z < 1.6$ was achieved by making the classifier invariant to MELA.

Classical

Weight	Feature
0.7590 ± 0.0157	dijet_invmass
0.5279 ± 0.0344	dijet_deltaeta
0.3872 ± 0.0542	jet_pt_1
0.3716 ± 0.0235	eta_zepp_ZZ
0.2285 ± 0.0177	leading_jet_width
0.1762 ± 0.0509	jet_pt_0
0.1358 ± 0.0295	subleading_jet_width
0.1279 ± 0.0419	m4l_fsr
0.0956 ± 0.0264	pt4ljj_unconstrained
0.0537 ± 0.0211	subleading_jet_TrackWidthPt1000
0.0296 ± 0.0114	MCFM_dxs_ggZZtot
0.0286 ± 0.0131	MELA
0.0191 ± 0.0233	MCFM_dxs_HZZ
0.0146 ± 0.0218	min_dR_jZ
0.0071 ± 0.0245	leading_jet_TrackWidthPt1000

Aspiration

Weight	Feature
0.8754 ± 0.0319	dijet_invmass
0.6496 ± 0.0170	dijet_deltaeta
0.4052 ± 0.0451	eta_zepp_ZZ
0.3617 ± 0.0372	jet_pt_1
0.2360 ± 0.0459	leading_jet_width
0.1328 ± 0.0624	jet_pt_0
0.1254 ± 0.0222	subleading_jet_width
0.1131 ± 0.0313	pt4ljj_unconstrained
0.0888 ± 0.0332	min_dR_jZ
0.0567 ± 0.0360	subleading_jet_TrackWidthPt1000
0.0321 ± 0.0232	MELA
0.0145 ± 0.0258	leading_jet_TrackWidthPt1000
0.0056 ± 0.0360	m4l_fsr
0.0027 ± 0.0106	MCFM_dxs_ggZZtot
-0.0013 ± 0.0077	MCFM_dxs_HZZ

Table 8.1 – Permutation Importance of features for the classical (left) network and network trained with aspiration targeting (right). The m_{4l} feature is referred to as `m4l_fsr` and its importance falls for the aspiration network.

8.1.5 Flexibility of the algorithm

Another interesting feature of the aspiration network is that one can scale up the number of variables to which the classifier should be invariant straightforwardly, without adding hyper-parameters. One just has to add the additional variable to the adversarial network. This is because the adversary is still performing a classification of real vs aspired, not a regression of multiple variables like the pivot algorithm. The loss for the classifier does not require a second adversarial loss term. If a hyper-parameter for individual variables is desired (for example if the importance for being invariant to one feature is more than to another), then separate adversarial networks per variable can be trained, leading to additional loss terms for the classifier.

This idea (of using a single adversarial loss term to make the classifier invariant to two observables) was tested by making the classifier invariant to m_{4l} and MELA simultaneously, and worked out-of-the-box, without requiring any hyper-parameter tuning. It was noted that epoch picking became even less of an issue in practice. A plot of the score distribution as a function of MELA, m_{4l} is shown in Figure 8.12.

This algorithm takes advantage of the fact that since only the classifier is used for inference, almost any information can be given to the adversarial network as in input. For this reason it is flexible enough for other tasks as well. In certain cases, training pivot adversarial training applied to remove background mass sculpting makes a classifier actually increase mass sculpting even more. If there are multiple components of the background, the classifier may try to sculpt the mass for the different components in opposite ways so that overall background distribution appears unsculpted. In an aspiration network, the adversary could be given the labels of the various background components as an additional input feature, allowing it to learn a conditional correlation, and therefore force the classifier to be invariant to the mass for each background component individually. This proposed solution was not studied further.

In principle the aspired distribution could be different from just a decorrelation. An interesting task (but perhaps purely of academic interest) that has not yet been studied is to deliberately force a correlation of the classifier with some combination of features.

It is worth noting that the classifier cannot be invariant to a variable that it cannot reconstruct from its inputs. A study was conducted on the HiggsML dataset [135] with the addition of Tau Energy Scale (TES) systematic smearing based on the work in [136] to study the performance of the Aspiration Network in making a classifier invariant to TES variations. The aspiration network (as well as the pivot algorithm as shown in [136]) failed to make the classification invariant to the TES, and therefore the details of this study are not described here.

A paper on the Aspiration Network is in preparation.

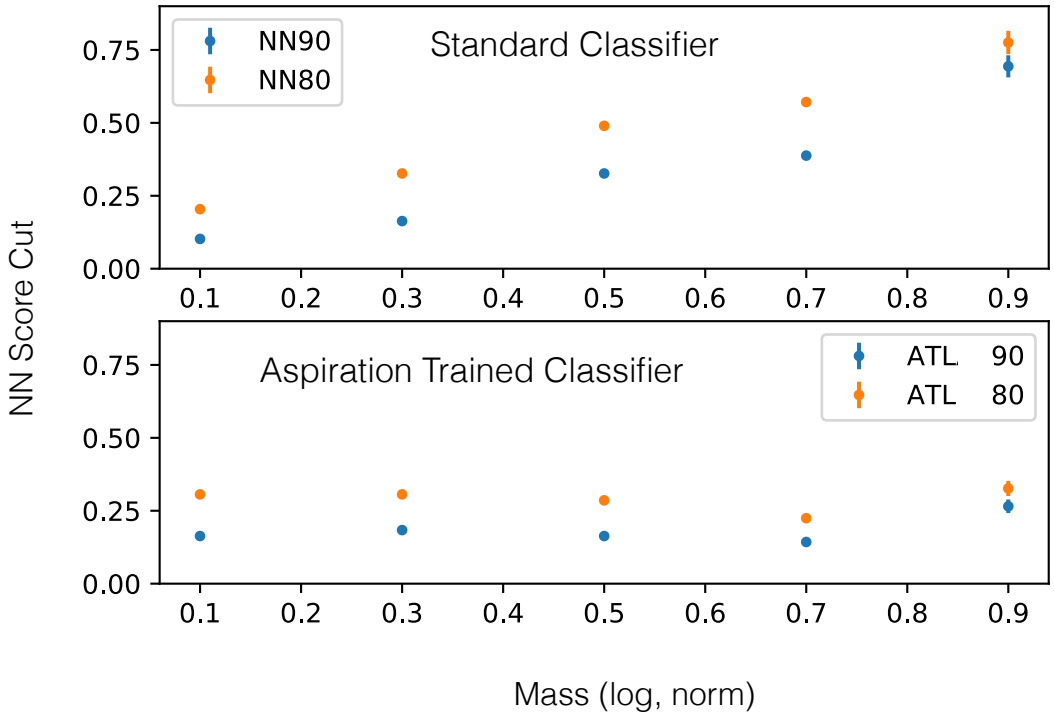
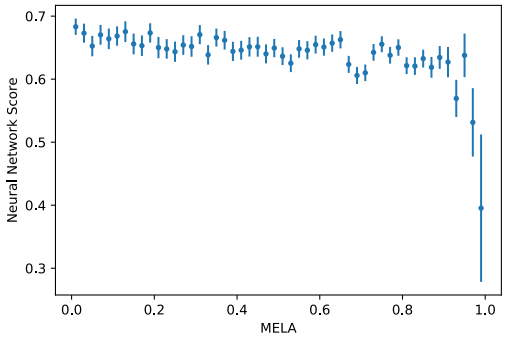
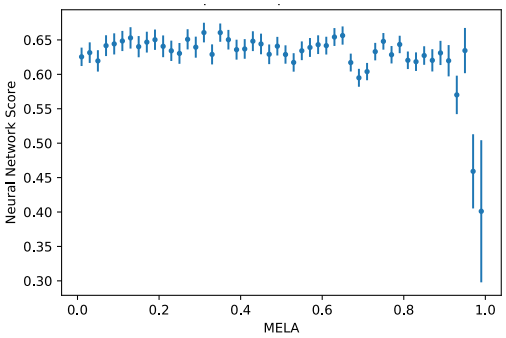


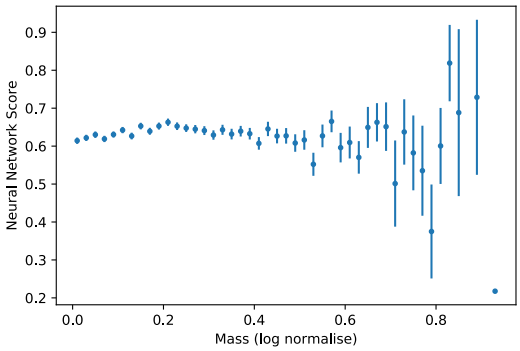
Figure 8.11 – Evolution of the score threshold for 80% (orange) and 90% (blue) signal efficiency as a function of the log-normed mass for a standard classifier (top) and a classifier trained with aspiration targeting (bottom).



(a) Classical, MELA



(b) Aspiration, MELA



(c) Aspiration, (log-norm) mass

Figure 8.12 – Classification score as a function of MELA for (a) a classical (b) an aspiration network trained to be invariant to the MELA and mass simultaneously. (c) Classification score as a function of (log-norm) mass for the same aspiration network. It is to be noted that the y-axis limits for the classical network are wider than for the Aspiration targeted one.

Chapter 9

Conclusion

The endeavour to understand the building blocks of the universe has reached a stage where extremely high precision measurements using equipment maintained by a large international collaboration are needed to make the next breakthrough. Many kinds of potential New Physics are expected to modify the apparent properties of the SM Higgs boson, because of which the ATLAS and CMS experiments scrutinise this particle in the hope to find hints for the yet unknown.

Although the field of particle physics has the unique benefit of building up from first principles, such simulations get more and more computationally expensive as the demand for precise simulated data grows. This is a concern for experiments at the LHC. Particle shower simulation in particular takes up a large portion of the simulation time, $\sim 75\%$ of it in the case of the ATLAS experiment. Considering the end of Moore's Law coincides with the birth of the Deep Learning revolution, a completely different approach to simulating particle showers in ATLAS, based on deep learning, is studied in the first part of this thesis. A deep generative model (in particular, a GAN) was trained to parameterise photon showers in the region $0.2 < |\eta| < 0.25$ of the ATLAS electromagnetic calorimeter, and integrated into the simulation software. This allowed for the first time to make fair comparisons in terms of speed, resource consumption, and accuracy to `Geant4` as well as the existing fast simulation framework `ATLAS Fast II` and its upcoming upgrade `FastCaloSimV2`, within the ATLAS software framework and using the standard validation framework. This strategy was shown to provide the orders of magnitude speed up with respect to `Geant4` required for future needs (70ms for the GAN compared to 10s for `Geant4` to simulate a 65 GeV photon shower), even on the single threaded CPUs without any batching. The memory footprint was found to be orders of magnitude smaller than the baseline fast simulation frameworks (file size of 9.6 MB compared to $\mathcal{O}$(GBs) and peak memory usage of 2.3 GB compared to 6GB for the GAN and `FastCaloSimV2` respectively), and is not projected to blow up in the future when a few generative networks are used to model the entire calorimeter for all incident particles.

In terms of accuracy, there is no distribution that the GAN is completely unable to learn. The model is able to condition its response to the position and energy of the indecent particle as well as the fast changing detector geometry. The performance is comparable to traditional fast simulation techniques, although still far from ready for routine use. The model performs better for certain distributions than others, and the same distributions are sometimes better modelled at low and medium energy points compared to high energy points. This is in part due to the fact that the model was trained on only nine, log-spaced energy points, rather than on a continuous spectrum. The gaps between energy points is larger at higher energies and with slightly fewer training events. Despite this peculiarity in the dataset, the model is shown to

be able to interpolate reasonably well to untrained energy points. Although it is tempting to further improve the final performance of the GAN with the help of corrections, post-processing or inelegant strategies, solutions that will not generalise well were not pursued at this stage of the research and development.

The GAN also cuts down on human time spent on parameterisation compared to the hand designed fast simulation frameworks. With the exception of the energy distribution, the GAN learnt most other distributions without manual intervention. It reproduced new validation observables that were untracked during the model optimisation stage of the project reasonably well. Some aspects of the simulation (such as the interpolation mechanism and the R_ϕ distribution) took significant human time to model correctly for the traditional fast simulation strategy, which we get for free with the GAN. However, significant human time is required to assess GAN performances, particularly for model optimisation and epoch picking. Monitoring the trainings over multiple days also is time consuming.

This study assumes periodic translation invariance, which is true in ϕ but breaks down in η at the edge of the barrel and in the end-cap. The different configurations of layer alignment was accommodated into the model architecture, however, further difference in geometry exist outside the studied region. In the end-caps the width of the cells in the Strip layer changes. The granularity gets further complicated in the Hadronic Calorimeter which is relevant for pion shower simulations. Training on cropped cell-level 3D images therefore faces many challenges due to the varying granularity of the cells. Since `Geant4` produces point cloud data, one possibility is to train on point cloud level data using graph networks. Another possibility is to bin the point cloud data into voxels of sizes that make training generative models simpler. For the next stage of this project it is imperative to have a dedicated simulation of a training dataset using `Geant4` that is better suited for a generative models assisted fast simulation strategy, particularly if just a few networks are to be conditioned to simulate showers in all sections of the calorimeter. The success of this study suggests that an investment of (human) time and resources required to design and simulate an appropriate training dataset will be worth the effort.

Although WGAN-GPs are a popular flavour of GANs (due to their training stability), this study finds that one of its key features, the gradient penalty, prevents it from learning the total energy distribution. This feature is usually not of relevance for natural images but a key aspect of physics datasets. A solution in the context of particle physics datasets was proposed, which involves the use of an additional critic network with an extremely low (but non-zero) gradient penalty weight and takes as input the total energy of the shower in place of the entire image.

Other ATLAS groups have taken these ideas forward, often based on expertise gained from this project (and the sister project using VAEs at Université de Genève). One group is training a collection of WGAN-GPs on voxels to model the entire detector for all particles (electrons, photons and pions). Another group is doing the same with a VAE, but in addition also conditioning it on the η of the particle, thereby reducing the number of networks required.

These studies have been made publicly available as an ATLAS technical note [92] and follow up public plots, and a paper is in preparation.

The second part of this thesis studied strategies to improve the measurement of the off-shell couplings of the Higgs boson in the four leptons decay channel, with particular focus on the vector boson fusion production mode. Usually out of reach at the LHC, the off-shell couplings become measurable in the four leptons decay channel due to certain threshold effects. Interference between signal and background processes makes the problem even more interesting. The measurement allows to lift certain degeneracies that cannot be lifted from on-shell measurements alone and it is particularly interesting in the EFT framework interpretation. The measurement is also an indirect probe into the total width of the Higgs boson, which is predicted to be greater than the SM value for a number of New Physics scenarios, such as a new particle cou-

pling to the Higgs boson. Such an indirect measurement of the Higgs boson width would be model dependent when performed using the dominant gluon fusion events, but it would be more model-independent if performed in a VBF category, although at the expense of much smaller statistics.

Quantum interference between VBF Higgs boson process and VBS background process makes this signal strength measurement quite different from the usual cases that do not deal with interference effects. For example, the expected number of events is no longer linear with the signal strength parameter μ, and can even decrease as a function of μ due to negative interference. The usual metrics used to estimate the sensitivity of a signal strength measurement analysis using MC simulated events are no longer relevant in the context of quantum interference. The usefulness of the typical strategy of using ML classification models to improve the sensitivity of the analysis also becomes unclear because the concept of 'class labels' becomes ill-defined.

In this work, a new approximate metric was derived and used to estimate the sensitivity of an analysis based on a given event selection criteria which takes into account the effects of quantum interference. The metric was used to conclude that if classification is to be used to improve sensitivity to this analysis, the classifier should be trained on unphysical simulations of the VBF Higgs-only process as the signal, rather than the physical VBF full process simulation. The metric has since then been used by another group to rank different event selection criteria, and they found that the metric corresponds reasonably well to rankings based on negative log-likelihood curves.

Given that the usual concept of signal and background events breaks down in the context of quantum interference, a possible method to optimise the analysis could be to use an ML model to directly optimise the final objective of the analysis. For this reason, a promising new family of deep learning based likelihood-free inference models, that are able to leverage additional information from the simulator, were adapted to a signal strength measurement problem. Developed for Lagrangian parameter measurements (like in EFT), these physics-aware models are in principle better suited to analyses that deal with considerable quantum interference than traditional classification. The ALICES model is aware of how the physics changes with the parameter of interest (μ) because it is parameterised on μ and trained on events from datasets which represent various values of μ. It learns the likelihood ratio between two hypothesis directly with the help of augmented data from the simulator, and it can then be applied for inference on data from collisions at the LHC. This model could in principle also be parameterised on certain systematic uncertainties. The SALLY model is trained only on events from the SM, and learns to regress the score of the event. It therefore requires fewer events to train.

ALICES was found to outperform the traditional inference method and considerably improve sensitivity for all values of the signal strength, while the SALLY model outperformed traditional methods only near the SM value. This study was performed without taking into account contamination from $gg \to (H \to)ZZ$ or $q\bar{q} \to ZZ$ through their higher order corrections (that leave additional jets in the final state). For a quick turn-around time, the detector effects were simulated using `Delphes`. Meanwhile the ATLAS HZZ simulation chain has been modified to support carrying through the additional weights required for this simulated-assisted learning strategy.

Although it is difficult to estimate the improvement in terms of numbers for full analysis using all background samples and the ATLAS detector simulation, this study suggests in its most conservative estimate, an improvement of a 2σ limit from 9.5 to 7.9 can be expected for a luminosity of 32 fb^{-1} on the off-shell VBF signal strength over an analysis performed using the distribution of the differential cross-section with respect a one-dimensional observable. This estimate is based on the improvement ALICES brings to the best one dimensional observable, SALLY, when the rate information is also used, corresponding to Figure 7.17b. Considering the

gain from ALICES is much higher if the rate information is not used (Figure 7.17a) and the fact that the dominant background from $q\bar{q}$ would reduce the added sensitivity from the rate information, it is reasonable to hope for a much larger relative improvement in a full analysis (even more so if the two jet requirement is lifted).

These results have raised interest within the ATLAS community and the ATLAS research group at University of Massachusetts has joined the effort to bring these likelihood-free inference techniques into ATLAS. Event generation strategies are being developed to use this technique either in both the ggF and VBF categories, or only in the VBF category. Considerable work is still needed to understand how to study systematic effects on the ALICES model, and how to combine the results with the $ll\nu\nu$ channel.

Apart from the off-shell couplings measurement, this parameterised inference model could also benefit EFT measurements, any analysis which suffers from quantum interference, such as a Higgs self-coupling measurement based on di-Higgs events.

As experiments start using neural network based likelihood-free inference, it will become incredibly difficult for phenomenologists to approximately replicate and reinterpret the results compared to cut-and-count analysis or even an analysis that performs a measurement using the distribution of the differential cross-section as a function of a Matrix Element based observable. To make it easier for them, the ATLAS collaboration and other experiments should consider releasing these neural network models in a format like ONNX along with the results.

This thesis also introduces an improved adversarial training algorithm, used in this case for mass decorrelation, which is referred to as the 'Aspiration Network', where the training algorithm provides information about the 'aspired distribution' to the adversarial network. The learning is made easier for the adversarial network by meaningfully increasing its input dimension. The algorithm also scales well when the classifier needs to be invariant to multiple observables without the need for fine-tuning of additional hyper-parameter for every such observable (in contrast to the baseline 'pivot' adversarial technique). It leaves open the possibility to also be adapted to induce a particular correlation in future work. A paper on the Aspiration Network is in preparation.

The work presented in this thesis builds upon prior work and also brings certain new ideas. The assessments of these very ambitious ideas that may change key components of the way experimental physics is performed has allowed the HEP community to understand the strength and pitfalls of using these innovations in a realistic context. These ideas will likely play a large impact on physics analysis in Run3 of the LHC.

Over the past few years, the HEP community has become more ambitious when it comes to using creative new ML based strategies to solve physics problems, hopefully the work described here played a small part in accelerating movement in this direction.

Chapter 10

Synthèse

Contents

La physique des particules est l'étude des plus petits éléments fondamentaux de l'univers et les propriétés étudiées ont des conséquences à l'échelle cosmologique. Notre compréhension actuelle de l'univers est incomplète et l'un des moyens les plus prometteurs de trouver des indices pour la nouvelle physique est de briser des particules à très haute énergie et d'étudier le résultat de ces collisions. C'est l'idée qui sous-tend le Grand Collisionneur de Hadrons (LHC) du CERN.

Comme le fonctionnement de ces machines est coûteux, il est impératif de tirer le meilleur parti des données enregistrées, ce qui nécessite l'utilisation de techniques statistiques avancées. Cette thèse a étudié l'utilisation de techniques basées sur l'apprentissage automatique (ML pour Machine Learning) pour la simulation rapide et précise d'un sous-détecteur de l'expérience ATLAS, pour effectuer des mesures précises en tirant parti de différents types d'informations qui peuvent être exploitées et pour éliminer certains biais indésirables des algorithmes de sélection basés sur le ML.

10.1 Aperçu théorique

Le Modèle Standard (SM) de la physique des particules est un modèle mathématique qui tente de décrire toutes les forces connues de l'univers, à l'exception de la gravité. Les particules fondamentales du modèle standard et leurs propriétés sont résumées Table 2.1 et Table 2.2. Le boson de Higgs est un élément essentiel du modèle car il est lié au champ qui fournit les masses de toutes les particules massives, à l'exception des neutrinos (qui sont sans masse dans le SM).

10.1.1 Boson de Higgs au LHC

Le boson de Higgs est produit selon quatre modes de production principaux :

- Processus de fusion gluon-gluon (ggF), via une boucle de quark (dominée par le quark top). Il représente 88% de la production totale.
- Le procédé Fusion de Bosons Vecteurs (VBF), qui laisse deux jets vers l'avant provenant des deux quarks dans l'état final en plus des produits de décomposition de Higgs. Il représente 7% de la production totale.
- Production associée avec Bosons Vectors (VH). Elle représente 4% de la production totale.
- Production associée à une paire de top ($t\bar{t}H$). Elle représente 1% de la production totale.

Les principaux diagrammes de Feynman pour ces processus sont présentée Figure 2.3 et leurs sections transversales en fonction de l'énergie du centre de masse du LHC $\sqrt{s}$ est présentée Figure 2.4.

Le boson de Higgs se désintègre avant de pouvoir être directement détecté et ses produits de désintégration sont donc étudiés. Les principaux canaux de désintégration du boson de Higgs sont présentés Figure 2.5. Bien qu'il se désintègre plus fréquemment en deux quarks b ou deux bosons W, le canal de désintégration en leptons des deux bosons Z présente des états finaux beaucoup plus propres, ce qui rend l'analyse plus facile.

10.1.2 Mesure de couplage du boson de Higgs hors résonance dans le canal des quatre leptons

La largeur totale du boson de Higgs est d'un grand intérêt car elle fournit des informations sur les modes de désintégration du boson de Higgs, y compris le couplage à de toute nouvelles particules non encore découvertes. La largeur prédite par le SM est de 4,07 MeV, ce qui est trop petit pour être mesuré directement au LHC (étant donné la précision des détecteurs), mais il est possible de le sonder par des mesures indirectes.

La mécanique quantique permet aux particules virtuelles d'avoir une masse invariante éloignée de leur masse polaire. Cependant, la probabilité de produire de telles particules s'éloigne généralement de la masse polaire et ne peut donc pas être étudiée au LHC. Le canal de désintégration du boson de Higgs en quatre leptons offre une occasion unique d'étudier le boson de Higgs dans son régime hors résonance en raison d'une section transversale augmentée par certains effets de seuil. Ceux-ci permettent de contraindre les théories BSM (Au-delà du modèle standard) ainsi que de sonder indirectement la largeur du boson de Higgs en combinant les mesures des couplages sur-résonance et hors-résonance [32]. Une telle mesure de la largeur du boson de Higgs dépend du modèle (elle nécessite l'utilisation de certaines hypothèses théoriques), cependant, si la mesure est effectuée en utilisant des événements du boson de Higgs qui sont produits via le mode de production sous-dominant du VBF, alors elle nécessite moins d'hypothèses théoriques [40].

Une telle étude devra tenir compte des effets d'interférence quantique entre le signal et les processus de fond qui diminuent en fait la section transversale globale pour le SM par rapport à un scénario de fond uniquement (SM sans le boson de Higgs) à la fois pour les modes de production ggF et VBF, comme on peut le voir sur la Figure 2.9a.

10.2 Aperçu expérimental

Le LHC est l'accélérateur de particules le plus puissant au monde, qui fait entrer en collision des particules à l'échelle du TeV. Il s'y trouve quatre grands détecteurs de particules placés à quatre points d'interaction, dont le détecteur ATLAS. sur lequel porte la présente thèse.

10.2.1 Détecteur ATLAS

ATLAS est une expérience à usage général avec un large éventail d'objectifs physiques, y compris des mesures précises des propriétés du boson de Higgs. Elle se compose de plusieurs sous-détecteurs, comme l'illustre la figure 3.4 :

- Le Détecteur Interne fournit une mesure précise des trajectoires et de la quantité de mouvement des particules chargées.
- Les Calorimètres Électromagnétiques et Hadroniques mesurent l'énergie des électrons, des photons et des hadrons en absorbant leur énergie.
- Le Spectromètre de Muons fournit des mesures complétées par les informations du détecteur interne pour l'identification et la reconstruction des muons.

Les différentes composantes du calorimètre sont illustrées dans la Figure 3.6. Une particule électromagnétique qui passe à travers le matériau d'absorption du calorimètre électromagnétique forme une gerbe. L'énergie totale mesurée ainsi que les formes de la gerbe fournissent des informations précieuses qui peuvent être utilisées pour déduire quel type d'interactions a eu lieu pour un événement donné. Cet effet en cascade est une fonction exponentielle de l'énergie de la particule incidente.

10.3 GAN pour la simulation de calorimètre rapide dans ATLAS

Les simulations des interactions attendues au LHC, basées sur les équations de physique de base, sont précises mais souvent coûteuses en termes de calcul. L'effet de cascade des gerbes qui se produit dans les calorimètres en particulier constitue un goulot d'étranglement en termes de temps de calcul pour l'expérience ATLAS. En effet, le temps de simulation pour chaque particule est une fonction de son énergie. L'évolution temporelle de la gerbe n'est pas enregistrée par le détecteur, seul le résultat final est enregistré. Une réduction significative du coût de calcul est nécessaire pour qu'ATLAS puisse atteindre ses objectifs en matière de physique au cours de la prochaine décennie (voir Figure 5.1).

Des paramétrages conçus à la main sont déjà utilisés [5] pour simuler la gerbe dans le calorimètre; ces algorithmes sont plus rapides à exécuter mais moins précis et ont également une grande empreinte mémoire. Dans cette thèse, un réseau de neurones est formé sur une petite section $(0,2 < |\eta| < 0,25, -\pi < \phi < \pi)$ du calorimètre électromagnétique ATLAS pour simuler la gerbe de photons individuels.

Le réseau est intégré dans le cadre logiciel ATLAS, `Athena`, et des comparaisons sont faites en termes de temps de simulation, de besoin en mémoire et de modélisation des observables de physique qui sont calculés à partir des gerbes générées jusqu'aux simulations complètes à partir de `ATLAS Fast II` et `FastCaloSimV2` [90].

10.3.1 Architecture et Entraînement

Un Réseau antagoniste génératifs (GAN) de Wasserstein, qui utilise une pénalité de gradient [69] est entrainé sur la moitié de l'ensemble des données, donc sur 44000 gerbes de photons provenant de neuf points d'énergie répartis logarithmiquement entre 1 et 262 GeV. Le GAN utilise un réseau critique supplémentaire avec une faible pénalité de gradient pour améliorer la modélisation de la distribution totale de l'énergie du réseau de générateur et un régularisation d'activité est appliqué pour encourager la parcimonie de la sortie. Le GAN est conditionné par la géométrie du détecteur ainsi que par l'énergie et la position de la particule incidente. L'architecture est présentée Figure 5.9.

10.3.2 Validation

La validation est effectuée d'une part de manière autonome et d'autre part après intégration du réseau dans `Athena` en utilisant des observables de physique pertinentes. La résolution énergétique du détecteur est bien modélisée par le GAN avec l'aide du deuxième réseau critique, comme le montre la Figure 5.13. Le GAN apprend également à bien conditionner par rapport à la position de la particule (voir Figure 5.21) et la géométrie du détecteur (voir Figure 5.26) et modélise également bien les formes de gerbe transverses (voir Figure 5.32) et les corrélations entre couches (voir Figure 5.35) mais modélise légèrement mal les formes longitudinales (voir Figure 5.37). Les observables de physique complexe calculées dans `Athena` après simulation du bruit électronique et application d'un calibrage sont également bien modélisés. Le GAN est également capable d'interpoler vers des points d'énergie non entraînés (voir Figure 5.49 et Figure 5.48). Le GAN fonctionne mieux que `ATLAS Fast II` dans certaines distributions et moins bien dans d'autres (voir Figure 5.51), ce qui indique que des améliorations supplémentaires sont nécessaires. D'autres travaux sont également nécessaires pour améliorer les performances à des énergies plus élevées.

10.3.3 Performance des logiciels

Le GAN prend 70 millisecondes par gerbe (indépendemment de l'énergie), alors que `Geant4` prend 10 secondes par gerbe pour des photons de 65,5 GeV, ce qui satisfait aux exigences de vitesse actuelles. L'évolution du temps de simulation en fonction de l'énergie est présentée dans la Figure 5.58.

Le fichier de paramétrage du GAN a une taille de 9,6 Mo sur le disque, ce qui est beaucoup plus petit que le fichier de paramétrage $\mathcal{O}$(GBs) pour `FastCaloSimV2` et le GAN a également une utilisation de la mémoire beaucoup plus petite de 2,3 Go (le réseau lui-même ne prenant que 5 Mo) contre 6,0 Go pour `FastCaloSimV2` [1]. Ces chiffres ne devraient pas augmenter de façon dramatique lorsque cette méthode sera étendue à toute la gamme des η.

[1] dernière comparaison effectuée en février 2019

10.3.4 Perspectives d'Avenir

Ces travaux ont jeté les bases de l'application de réseaux générateurs pour la simulation rapide du calorimètre ATLAS et les leçons tirées de ce projet ont été utiles pour préparer une stratégie visant à étendre cette approche à l'ensemble du détecteur et à toutes les particules incidentes.

Cette étude a été rendue public sous la forme d'une note technique ATLAS [92] et une publication est en préparation.

10.4 Mesure de couplage de Higgs hors résonance

Les analyses précédentes pour la mesure des couplages du boson de Higgs hors-résonance dans ATLAS n'ont pas été optimisés pour la sensibilité pour le mode de production de VBF sous-dominant.

Une nouvelle métrique, 'signification statistique avec interférence', dénommée 'iZ' (Equation 4.44) a été dérivée pour estimer la sensibilité des critères de sélection des événements car la métrique habituelle ne prenait pas en compte l'interférence quantique entre le signal et les processus de fond. Cependant, l'utilisation d'une seule observable pour la mesure finale négligerait les effets non-linéaires introduits par l'interférence quantique et c'est pourquoi un ajustement de maximum de vraisemblance multidimensionnel non binné et non analytique utilisant un réseau neuronal a été étudié à la place. Cette technique repose sur l'entrainement du réseau de neurones à l'aide d'informations supplémentaires extraites du simulateur. Il est démontré que cette technique est plus performante qu'un ajustement dimensionnel traditionnel du signal.

10.4.1 Le problème de l'interférence quantique

Le nombre attendu d'événements en présence d'interférence quantique est une fonction non-linéaire de la force du signal (qui est une approximation des forces de couplage de Higgs), comme le montre la Figure 7.2 et entraîne des dégénérescences lorsqu'elle est mesurée sur la base d'ajustements de probabilité maximale en utilisant une seul observable, comme le montrent la Figure 7.4 et la Figure 7.19, même si ces dégénérescences peuvent en principe être levées en utilisant des informations pertinentes, comme le montre la Figure 7.3.

10.4.2 Inférence sans Fonction de Vraisemblance

Le modèle ALICES (**A**pproximate **L**ikelihood with **I**mproved **C**ross-entropy **E**stimator and **S**core) est paramétré sur la force du signal, μ, et apprend à régresser le rapport de vraisemblance de chaque événement pour une hypothèse donnée μ par rapport au SM ($\mu = 1$). Le gradient du rapport de vraisemblance par rapport à μ, appelé 'score', est également utilisé pour améliorer la convergence de l'entraînement. Bien que le véritable rapport de vraisemblance ne soit pas disponible comme cible de l'entraînement, le rapport de vraisemblance conjoint, qui est le rapport de vraisemblance au niveau du parton, peut être utilisé comme cible pour l'entraînement. Le réseau neuronal fournit directement l'inférence statistique finale des rapports de vraisemblance et, par conséquent, aucun ajustement supplémentaire n'est nécessaire.

10.4.3 Résultats

ALICES surpasse de manière significative toute inférence réalisée à l'aide d'ajustements unidimensionnels basés sur une seule observable. Il le fait de manière cohérente pour les ensembles

de données de test générés à diverses valeurs de μ. C'est également la meilleure technique pour lever les dégénérescences dans la valeur de probabilité maximale de μ car il s'agit d'un ajustement multidimensionnel qui prend en compte toutes les observables. Ces comparaisons de performances peuvent être vues dans les Figures 7.16 et Figure 7.17.

10.4.4 Discussion et perspectives

Ces études montrent que la réalisation d'un ajustement multidimensionnel à l'aide d'un réseau de neurones surpasse considérablement les techniques traditionnelles de construction d'une observable optimale, car l'ajustement multidimensionnel peut prendre en compte les effets non linéaires qui se produisent en raison de l'interférence quantique. Toutefois, ces études ont été réalisées en utilisant uniquement le signal $qq \to (H^* \to)ZZ$ et les processus de fond où les effets des détecteurs ont été simulés avec `Delphes` [84]. Tous les processus avec des états initiaux gg et $q\bar{q}$ et les effets réels du détecteur ATLAS doivent également être pris en compte à l'avenir.

Cette technique peut également être utile pour toute autre analyse où les effets d'interférences quantiques sont significatifs.

10.5 Réseau Aspiration

Une stratégie commune au sein d'ATLAS pour supprimer la dépendance de masse de la sortie d'un classificateur est d'utiliser l'entraînement adversariel à pivot [131–133], dont l'architecture est présentée sur la figure 8.4. Cette technique échoue pour l'ensemble de données de Higgs à quatre leptons dont la corrélation entre la masse des quatre leptons (m_{4l}) et la sortie du classificateur est très bruyante.

Dans cette thèse, il est montré que la plupart des types de réseaux neuronaux et d'arbres de décision boostés n'apprennent pas cette corrélation, comme par exemple dans la Figure 8.5. Nous introduisons une nouvelle stratégie d'entraînement adversariel appelée 'réseau aspiration', dans laquelle la tâche du réseau adversariel est simplifiée et sa dimension d'entrée est augmentée de de façon utile en fournissant des informations supplémentaires.

La 'distribution aspirée' est un ensemble de données artificielles créé pour avoir la distribution idéale souhaitée, une décorrélation complète entre le m_{4l} et la sortie du classificateur tout en gardant les distributions marginales identiques à l'ensemble de données réelles (la sortie du classificateur et le m_{4l} des leptons, qui sont corrélés). En pratique, la distribution souhaitée est construite en mélangeant les variables entre événements.

Le réseau aspiration fournit les résultats souhaités dès le départ, comme le montre la Figure 8.11. Il peut même s'adapter pour rendre le classificateur invariant à plusieurs variables simultanément (voir Figure 8.12) sans qu'il soit nécessaire de procéder à un étalonnage complexe des hyperparamètres supplémentaires liés à des termes supplémentaires de la fonction perte, comme c'est le cas pour l'algorithme de pivot. Cet algorithme est également suffisamment souple pour traiter certains problèmes supplémentaires qui surviennent parfois dans des applications réalistes. Il est intéressant de noter que l'algorithme peut également être ajusté pour induire délibérément une corrélation aspirée mais ces idées ne sont pas approfondies dans cette thèse.

Une publication sur le réseau Aspiration est en préparation.

Acknowledgements

There were innumerable people who at various crucial points who took a leap of faith by supporting me and my outlook towards academics and science. I have been incredibly fortunate to meet inspirational people and be exposed to diverse ways of thinking over the years which has left a strong impression on me and therefore my work.

I must thank my reviewers, Maurizio Pierini and Isabelle Wingerter-Seez for going through my thesis manuscript in detail and making insightful suggestions on how to improve its quality. Danilo Rezende for some very insightful comments which will also affect my future work. Marie-Helene Schune who I have interacted with several times over the last few years. I have admired the ability of Glen Cowan to ask simple but insightful questions that often lead to very interesting discussions about statistics in each of my interactions with him.

I have also had the great pleasure to interact with Vava Gligorov, Tilman Plehn and Anja Butter with whom I have had very fruitful discussions about physics, machine learning and beyond. Their career advice has been invaluable. Discussions with Lukas Heinrich, Alex Radovic and Sofia Vallecorsa also helped me made the right career decisions.

I will remain ever grateful to Louis Fayard for introducing me to the world of particle physics and to Reisaburo Tanaka with whom I shared an office and several discussions over the years. Lydia, RD, Christophe and Antoine for creating a supportive environment at LAL. The little words of encouragement from Dimitris went a long way in motivating my work. I also enjoyed the time I shared with Sabrina, Christina and Anasthasia, it was valuable to have such a support system of young researchers at LAL. Corentin and Konie helped me learn about French culture as well as navigate through the French administration.

I thoroughly enjoyed my conversations with Manuel and Aaron and learn a lot from these conversations. Towards the end of my PhD my conversations with Johnny Raine helped through the fun and not so fun aspects of our work.

I will also remember the chats with Antinéa, Laurent Basara, Rikel and the friends made along the way. The great friends I made in Marseille with whom I discussed physics and beyond were also a great support through the PhD. Nicole Hartman graciously sharing a couple of her presentation aesthetics secrets with me for which me and my audience are thankful.

I learnt a lot from Victor Estrade, Cécile Germain, Isabelle Guyon and Adrian Pol with whom I had the good fortune to discuss machine learning and statistics. The interns who worked with me, Antonio (on GANs), Jeremy (on h4l) and Biswajit (on OPU), were a pleasure to work with and their work has directly contributed to some of the results in this thesis. Every student and staff at LAL/IJCLab must also acknowledge the support from Catherine Nizery, Sylvie Prandt, Annick Michaud and Sylvie Teulet throughout the academic year. I also sincerely thank the cleaning staff who started returning to the lab immediately after the lockdown was eased, and

I will remember Mamadou Diaby in particular who I met twice every day, once at LAL in the evening and once at the RER station at night where he worked a second job.

It was my great fortune to have someone as nurturing as David Rousseau as my supervisor, from whom I learnt more in every interaction (in terms of physics but also in terms of temperament and how to handle pressure working in a large collaboration), and who has been incredibly supportive of my career developments. It was a very good fit for me to work with and learn from David because of our very different but complementary personalities on the one hand and a lot of commonalities in attitude towards research and rigour on the other. During the hardships of the 2020 pandemic, each conversation with David would inject motivation and focus rather than stress, which was quite a rare experience among PhD students during this period. I look forward to continued chats with him about research in the future.

These years living in Paris allowed me an excellent exposure to French culture and also to authentic cuisine and culture from various parts of Africa (Togo cuisine has been my biggest revelation). I hope to develop and cherish this interest.

During my work in ATLAS I had the great pleasure of working with Tobais Golling and Dalila Salamani, and I hope to work with them again. I also enjoyed working with Rafael Coelho Loupes de Sa, Samyukta Krishnamurthy, Martina Javurkova, Jay Sandesara, some of whom have even become friends. Discussions with Ben Nachman, Amir Farbin and Dan Guest were valuable.

I closely worked with Gilles Louppe and he helped provide a breakthrough with the GAN work. I also worked with Kyle Cranmer and had the support of Johann Brehmer and I will remain grateful to both of them for support with the MadMiner work. I am also thankful to Paul Klein who did the initial work on the GAN for ATLAS.

It would be criminal not to acknowledge the excellent support I have received from the staff at the computing centre at IN2P3, Sebastian and others with whom I discussed the infrastructural needs to support the ever growing ML projects in IN2P3. The team at LLR was also supportive.

My involvement with OECD is thanks to Karine Perset, Luis Aranda and Alistair Nolan because of which I was able to see the broader picture in terms of the use of AI and re-recognise the importance of reproducibility and transparency in scientific work. Interactions with Joseph Urban have also been important in keeping my eye on the bigger picture.

The reason I was able to start a PhD is in no small part also due to several teachers I had through my life, Ms. Kirti, Mr. Sanjay Sengupta among others. Mr. Stephen Cottrell would be happy to know I did end up vising CERN after all. The physics and history teachers and a senior student in my school were also quite inspirational and I enjoyed the support of my aunt. Dr. Gujar and the two Dr. Mistrys went out of their way to support a young aspiring scientist at a crucial point in his career and I hope they will believe it was worth the trouble. Dr. Singh also supported my career development. My statistics professors Ayesha Dias and Myrtle Fernandes, who I disappointed by not pursuing statistics might take solace in the fact that I ended up working on 'statistical learning'.

Finally, I should thank all my classmates who tolerated my "why"s all through school, even if it meant the physics lecture would cut into the ever so precious break time (let's face it, socialising during break time is why most of us went to school).

Glossary

ATLAS	A Toroidal LHC ApparatuS
ALICES	Approximate Likelihood with Improved Cross-entropy Estimator and Score
BDT	Boosted Decision Tree
BSM	Beyond the Standard Model
CERN	European Organisation for Nuclear Research
CL	Confidence Level
CMS	Compact Muon Solenoid
CP	Charge-Parity (symmetry)
DNN	Deep Neural Network
ECal	Electromagnetic Calorimeter
EFT	Effective Field Theory
ET	Transverse energy
EW	ElectroWeak
GAN	Generative Adversarial Network
ggF	gluon–gluon fusion
HCal	Hadronic Calorimeter
HEP	High Energy Physics
HLT	High-Level Trigger
ID	Inner Detector
LAr	Liquid Argon
LFI.	Likelihood-Free Inference
LHC	Large Hadron Collider
LO	Leading Order
MC	Monte-Carlo
ME	Matrix Element
MET	Missing transverse energy
ML	Machine learning
MLE	Maximum Likelihood Estimator
NN	Neural Network
NP	New Physics Nuisance Parameter
PDF	Parton Distribution Function
PI	Permutation Importance
PS	Pre-Sampler Parton Shower
pT	Transverse momentum
QCD	Quantum ChromoDynamics

QED	Quantum ElectroDynamics
QFT	Quantum Field Theory
SALLY	Score Approximates Likelihood Locally
SM	Standard Model
VAE	Variational Auto-Encoder
VBF	Vector Boson Fusion

List of Tables

References

[1] The ATLAS Collaboration. "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC" (2012). DOI: 10.1016/j.physletb.2012.08.020. arXiv: 1207.7214 (Cited on pages 9, 14)

[2] The CMS Collaboration. "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC" (2012). DOI: 10.1016/j.physletb.2012.08.021. arXiv: 1207.7235 (Cited on pages 9, 14)

[3] Claire Adam-Bourdarios, Glen Cowan, Cécile Germain, Isabelle Guyon, et al. "The Higgs boson machine learning challenge". *NIPS 2014 Workshop on High-energy Physics and Machine Learning*. Vol. 42. JMLR: Workshop and Conference Proceedings. Montreal, Canada, 2014. URL: http://proceedings.mlr.press/v42/cowa14.html (Cited on page 10)

[4] S. Agostinelli et al. "GEANT4: A Simulation toolkit". *Nucl. Instrum. Meth. A* 506 (2003), p. 250. DOI: 10.1016/S0168-9002(03)01368-8 (Cited on pages 11, 70, 141)

[5] ATLAS Collaboration. *The simulation principle and performance of the ATLAS fast calorimeter simulation FastCaloSim*. ATL-PHYS-PUB-2010-013. 2010. URL: https://cds.cern.ch/record/1300517 (Cited on pages 11, 71, 77, 201)

[6] D. P Kingma and M. Welling. "Auto-Encoding Variational Bayes". *ArXiv e-prints* (Dec. 2013). arXiv: 1312.6114 [stat.ML] (Cited on pages 11, 53)

[7] D. Jimenez Rezende, S. Mohamed, and D. Wierstra. "Stochastic Backpropagation and Approximate Inference in Deep Generative Models". *ArXiv e-prints* (Jan. 2014). arXiv: 1401.4082 [stat.ML] (Cited on pages 11, 53)

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, et al. "Generative Adversarial Networks". *ArXiv e-prints* (June 2014). arXiv: 1406.2661 [stat.ML] (Cited on pages 11, 52)

[9] Luke de Oliveira, Michela Paganini, and Benjamin Nachman. "Learning Particle Physics by Example: Location-Aware Generative Adversarial Networks for Physics Synthesis". *Comput. Softw. Big Sci.* 1.1 (2017), p. 4. DOI: 10.1007/s41781-017-0004-6. arXiv: 1701.05927 [stat.ML] (Cited on pages 11, 133)

[10] Michela Paganini, Luke de Oliveira, and Benjamin Nachman. "Accelerating Science with Generative Adversarial Networks: An Application to 3D Particle Showers in Multilayer Calorimeters". *Phys. Rev. Lett.* 120.4 (2018), p. 042003. DOI: 10.1103/PhysRevLett.120.042003. arXiv: 1705.02355 [hep-ex] (Cited on pages 11, 133)

[11] Michela Paganini, Luke de Oliveira, and Benjamin Nachman. "CaloGAN : Simulating 3D high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks". *Phys. Rev.* D97.1 (2018), p. 014021. DOI: 10.1103/PhysRevD.97.014021. arXiv: 1712.10321 [hep-ex] (Cited on pages 11, 133)

[12] The ATLAS Collaboration. "Constraints on off-shell Higgs boson production and the Higgs boson total width in $ZZ \to 4\ell$ and $ZZ \to 2\ell 2\nu$ final states with the ATLAS detector". *Phys. Lett. B* 786 (2018), pp. 223–244. DOI: 10.1016/j.physletb.2018.09.048. arXiv: 1808.01191 [hep-ex] (Cited on pages 11, 29, 139)

[13] F. Englert and R. Brout. "Broken Symmetry and the Mass of Gauge Vector Mesons". *Phys. Rev. Lett.* 13 (9 Aug. 1964), pp. 321–323. DOI: 10.1103/PhysRevLett.13.321. URL: https://link.aps.org/doi/10.1103/PhysRevLett.13.321 (Cited on page 18)

[14] P.W. Higgs. "Broken symmetries, massless particles and gauge fields". *Physics Letters* 12.2 (1964), pp. 132–133. ISSN: 0031-9163. DOI: https://doi.org/10.1016/0031-9163(64)91136-9. URL: http://www.sciencedirect.com/science/article/pii/0031916364911369 (Cited on page 18)

[15] Peter W. Higgs. "Broken Symmetries and the Masses of Gauge Bosons". *Phys. Rev. Lett.* 13 (16 Oct. 1964), pp. 508–509. DOI: 10.1103/PhysRevLett.13.508. URL: https://link.aps.org/doi/10.1103/PhysRevLett.13.508 (Cited on page 18)

[16] G. S. Guralnik, C. R. Hagen, and T. W. B. Kibble. "Global Conservation Laws and Massless Particles". *Phys. Rev. Lett.* 13 (20 Nov. 1964), pp. 585–587. DOI: 10.1103/PhysRevLett.13.585. URL: https://link.aps.org/doi/10.1103/PhysRevLett.13.585 (Cited on page 18)

[17] Peter W. Higgs. "Spontaneous Symmetry Breakdown without Massless Bosons". *Phys. Rev.* 145 (4 May 1966), pp. 1156–1163. DOI: 10.1103/PhysRev.145.1156. URL: https://link.aps.org/doi/10.1103/PhysRev.145.1156 (Cited on page 18)

[18] T. W. B. Kibble. "Symmetry Breaking in Non-Abelian Gauge Theories". *Phys. Rev.* 155 (5 Mar. 1967), pp. 1554–1561. DOI: 10.1103/PhysRev.155.1554. URL: https://link.aps.org/doi/10.1103/PhysRev.155.1554 (Cited on page 18)

[19] Adam Falkowski. "Higgs Basis: Proposal for an EFT basis choice for LHC HXSWG" (Mar. 2015). URL: https://cds.cern.ch/record/2001958 (Cited on page 21)

[20] Ilaria Brivio and Michael Trott. "The Standard Model as an Effective Field Theory". *Phys. Rept.* 793 (2019), pp. 1–98. DOI: 10.1016/j.physrep.2018.11.002. arXiv: 1706.08945 [hep-ph] (Cited on page 21)

[21] D. de Florian et al. "Handbook of LHC Higgs Cross Sections: 4. Deciphering the Nature of the Higgs Sector". 2/2017 (Oct. 2016). DOI: 10.23731/CYRM-2017-002. arXiv: 1610.07922 [hep-ph] (Cited on pages 23, 142, 161)

[22] The ATLAS Collaboration. "Measurement of the Higgs boson mass in the $H \to ZZ^* \to 4\ell$ and $H \to \gamma\gamma$ channels with $\sqrt{s} = 13$ TeV pp collisions using the ATLAS detector". *Phys. Lett. B* 784 (2018), pp. 345–366. DOI: 10.1016/j.physletb.2018.07.050. arXiv: 1806.00242 [hep-ex] (Cited on pages 23, 24)

[23] The CMS Collaboration. "A measurement of the Higgs boson mass in the diphoton decay channel". *Phys. Lett. B* 805 (2020), p. 135425. DOI: 10.1016/j.physletb.2020.135425. arXiv: 2002.06398 [hep-ex] (Cited on page 23)

[24] The ATLAS Collaboration. "Observation of $H \to b\bar{b}$ decays and VH production with the ATLAS detector". *Phys. Lett. B* 786 (2018), pp. 59–86. DOI: 10.1016/j.physletb.2018.09.013. arXiv: 1808.08238 [hep-ex] (Cited on page 22)

[25] Georges Aad et al. "Measurement of the Higgs boson mass from the $H \to \gamma\gamma$ and $H \to ZZ^* \to 4\ell$ channels with the ATLAS detector using 25 fb^{-1} of pp collision data". *Phys. Rev. D* 90.5 (2014), p. 052004. DOI: 10.1103/PhysRevD.90.052004. arXiv: 1406.3827 [hep-ex] (Cited on page 24)

[26] The ATLAS Collaboration. *Measurement of the Higgs boson mass in the $H \to ZZ^* \to 4\ell$ decay channel with $\sqrt{s} = 13$ TeV pp collisions using the ATLAS detector at the LHC*. Tech. rep. ATLAS-CONF-2020-005. Geneva: CERN, Apr. 2020. URL: http://cds.cern.ch/record/2714883 (Cited on page 24)

[27] J R Andersen et al. "Handbook of LHC Higgs Cross Sections: 3. Higgs Properties" (July 2013). Ed. by S Heinemeyer, C Mariotti, G Passarino, and R Tanaka. DOI: 10.5170/CERN-2013-004. arXiv: 1307.1347 [hep-ph] (Cited on pages 24, 25)

[28] The CMS Collaboration. "Limits on the Higgs boson lifetime and width from its decay to four charged leptons". *Phys. Rev. D* 92.7 (2015), p. 072010. DOI: 10.1103/PhysRevD.92.072010. arXiv: 1507.06656 [hep-ex] (Cited on page 24)

[29] John M. Campbell, R. Keith Ellis, and Ciaran Williams. "Bounding the Higgs Width at the LHC Using Full Analytic Results for $gg- > e^-e^+\mu^-\mu^+$". *JHEP* 04 (2014), p. 060. DOI: 10.1007/JHEP04(2014)060. arXiv: 1311.3589 [hep-ph] (Cited on pages 27, 140)

[30] The ATLAS Collaboration. "Constraints on the off-shell Higgs boson signal strength in the high-mass ZZ and WW final states with the ATLAS detector". *Eur. Phys. J. C* 75.7 (2015), p. 335. DOI: 10.1140/epjc/s10052-015-3542-2. arXiv: 1503.01060 [hep-ex] (Cited on page 28)

[31] Nikolas Kauer and Giampiero Passarino. "Inadequacy of zero-width approximation for a light Higgs boson signal". *JHEP* 08 (2012), p. 116. DOI: 10.1007/JHEP08(2012)116. arXiv: 1206.4803 [hep-ph] (Cited on pages 27, 28)

[32] Fabrizio Caola and Kirill Melnikov. "Constraining the Higgs boson width with ZZ production at the LHC". *Phys. Rev. D* 88 (2013), p. 054024. DOI: 10.1103/PhysRevD.88.054024. arXiv: 1307.4935 [hep-ph] (Cited on pages 27–29, 200)

[33] James S. Gainer, Joseph Lykken, Konstantin T. Matchev, Stephen Mrenna, et al. "Beyond Geolocating: Constraining Higher Dimensional Operators in $H \to 4\ell$ with Off-Shell Production and More". *Phys. Rev. D* 91.3 (2015), p. 035011. DOI: 10.1103/PhysRevD.91.035011. arXiv: 1403.4951 [hep-ph] (Cited on page 27)

[34] Christoph Englert, Yotam Soreq, and Michael Spannowsky. "Off-Shell Higgs Coupling Measurements in BSM scenarios". *JHEP* 05 (2015), p. 145. DOI: 10.1007/JHEP05(2015)145. arXiv: 1410.5440 [hep-ph] (Cited on page 27)

[35] Dorival Goncalves, Tao Han, and Satyanarayan Mukhopadhyay. "Off-Shell Higgs Probe of Naturalness". *Phys. Rev. Lett.* 120.11 (2018). [Erratum: Phys.Rev.Lett. 121, 079902 (2018)], p. 111801. DOI: 10.1103/PhysRevLett.120.111801. arXiv: 1710.02149 [hep-ph] (Cited on page 27)

[36] Aleksandr Azatov, Christophe Grojean, Ayan Paul, and Ennio Salvioni. "Taming the off-shell Higgs boson". *Zh. Eksp. Teor. Fiz.* 147 (2015), pp. 410–425. DOI: 10.1134/S1063776115030140. arXiv: 1406.6338 [hep-ph] (Cited on page 27)

[37] Margherita Ghezzi, Giampiero Passarino, and Sandro Uccirati. "Bounding the Higgs Width Using Effective Field Theory". *PoS* LL2014 (2014). Ed. by Martina Mende, p. 072. DOI: 10.22323/1.211.0072. arXiv: 1405.1925 [hep-ph] (Cited on page 27)

[38] Malte Buschmann, Dorival Goncalves, Silvan Kuttimalai, Marek Schonherr, et al. "Mass Effects in the Higgs-Gluon Coupling: Boosted vs Off-Shell Production". *JHEP* 02 (2015), p. 038. DOI: 10.1007/JHEP02(2015)038. arXiv: 1410.5806 [hep-ph] (Cited on page 27)

[39] Giacomo Cacciapaglia, Aldo Deandrea, Guillaume Drieu La Rochelle, and Jean-Baptiste Flament. "Higgs couplings: disentangling New Physics with off-shell measurements". *Phys. Rev. Lett.* 113.20 (2014), p. 201802. DOI: 10.1103/PhysRevLett.113.201802. arXiv: 1406.1757 [hep-ph] (Cited on page 27)

[40] Christoph Englert and Michael Spannowsky. "Limitations and Opportunities of Off-Shell Coupling Measurements". *Phys. Rev. D* 90 (2014), p. 053003. DOI: 10.1103/PhysRevD.90.053003. arXiv: 1405.0285 [hep-ph] (Cited on pages 29, 30, 180, 200)

[41] Apollinari G., Béjar Alonso I., Brüning O., Fessia P., et al. *High-Luminosity Large Hadron Collider (HL-LHC): Technical Design Report V. 0.1*. CERN Yellow Reports: Monographs. Geneva: CERN, 2017. DOI: 10.23731/CYRM-2017-004. URL: https://cds.cern.ch/record/2284929 (Cited on page 31)

[42] G Aad, S Bentvelsen, G J Bobbink, K Bos, et al. "The ATLAS Experiment at the CERN Large Hadron Collider". *JINST* 3 (2008). Also published by CERN Geneva in 2010, S08003. 437 p. DOI: 10.1088/1748-0221/3/08/S08003. URL: https://cds.cern.ch/record/1129811 (Cited on page 32)

[43] The CMS Collaboration, S Chatrchyan, G Hmayakyan, V Khachatryan, et al. "The CMS experiment at the CERN LHC". *Journal of Instrumentation* 3.08 (Aug. 2008), S08004–S08004. DOI: 10.1088/1748-0221/3/08/s08004. URL: https://doi.org/10.1088%2F1748-0221%2F3%2F08%2Fs08004 (Cited on page 32)

[44] The LHCb Collaboration, A Augusto Alves, L M Andrade Filho, A F Barbosa, et al. "The LHCb Detector at the LHC". *Journal of Instrumentation* 3.08 (Aug. 2008), S08005–S08005. DOI: 10.1088/1748-0221/3/08/s08005. URL: https://doi.org/10.1088%2F1748-0221%2F3%2F08%2Fs08005 (Cited on page 32)

[45] The ALICE Collaboration, K Aamodt, A Abrahantes Quintana, R Achenbach, et al. "The ALICE experiment at the CERN LHC". *Journal of Instrumentation* 3.08 (Aug. 2008), S08002–S08002. DOI: 10.1088/1748-0221/3/08/s08002. URL: https://doi.org/10.1088%2F1748-0221%2F3%2F08%2Fs08002 (Cited on page 32)

[46] Esma Mobs. "The CERN accelerator complex - August 2018. Complexe des accélérateurs du CERN - Août 2018" (Aug. 2018). General Photo. URL: https://cds.cern.ch/record/2636343 (Cited on page 33)

[47] *ATLAS Public luminosity plots for Run 2*. URL: https://twiki.cern.ch/twiki/bin/view/AtlasPublic/LuminosityPublicResultsRun2 (Cited on page 34)

[48] *High Luminosity Project Schedule*. URL: https://project-hl-lhc-industry.web.cern.ch/sites/project-hl-lhc-industry.web.cern.ch/files/inline-images/HL-LHC_Janvier_2020_1148x562.jpg (Cited on page 35)

[49] ATLAS Collaboration. "The ATLAS Experiment at the CERN Large Hadron Collider". *JINST* 3 (2008), S08003. DOI: 10.1088/1748-0221/3/08/S08003 (Cited on pages 35, 38, 40)

[50] ATLAS Collaboration. "The ATLAS Inner Detector commissioning and calibration". *Eur. Phys. J. C* 70 (2010), p. 787. DOI: 10.1140/epjc/s10052-010-1366-7. arXiv: 1004.5293 [hep-ex] (Cited on page 36)

[51] *ATLAS liquid-argon calorimeter: Technical Design Report*. Technical Design Report ATLAS. Geneva: CERN, 1996. URL: https://cds.cern.ch/record/331061 (Cited on page 40)

[52] ATLAS Collaboration. *Technical Design Report for the Phase-II Upgrade of the ATLAS LAr Calorimeter*. Tech. rep. CERN-LHCC-2017-018. ATLAS-TDR-027. Geneva: CERN, Sept. 2017. URL: https://cds.cern.ch/record/2285582 (Cited on page 41)

[53] *ATLAS calorimeter performance: Technical Design Report*. Technical Design Report ATLAS. Geneva: CERN, 1996. URL: http://cds.cern.ch/record/331059 (Cited on page 42)

[54] The ATLAS Collaboration. "Search for low-mass dijet resonances using trigger-level jets with the ATLAS detector in pp collisions at $\sqrt{s} = 13$ TeV". *Phys. Rev. Lett.* 121.8 (2018), p. 081801. DOI: 10.1103/PhysRevLett.121.081801. arXiv: 1804.03496 [hep-ex] (Cited on page 44)

[55] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: http://tensorflow.org/ (Cited on pages 46, 85)

[56] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, et al. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf (Cited on page 46)

[57] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, et al. *JAX: composable transformations of Python+NumPy programs*. Version 0.1.55. 2018. URL: http://github.com/google/jax (Cited on page 46)

[58] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, et al. "Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm". *CoRR* abs/1712.01815 (2017). arXiv: 1712.01815. URL: http://arxiv.org/abs/1712.01815 (Cited on page 46)

[59] Tianqi Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". *CoRR* abs/1603.02754 (2016). arXiv: 1603.02754. URL: http://arxiv.org/abs/1603.02754 (Cited on pages 48, 145)

[60] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, et al. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree". *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, et al. Curran Associates, Inc., 2017, pp. 3146–3154. URL: http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf (Cited on pages 48, 145)

[61] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, et al. *CatBoost: unbiased boosting with categorical Features*. 2018 (Cited on page 48)

[62] Henry J Kelley. "Gradient theory of optimal flight paths". *Ars Journal* 30.10 (1960), pp. 947–954 (Cited on page 49)

[63] Pierre Baldi, Kyle Cranmer, Taylor Faucett, Peter Sadowski, et al. "Parameterized neural networks for high-energy physics". *Eur. Phys. J. C* 76.5 (2016), p. 235. DOI: 10.1140/epjc/s10052-016-4099-4. arXiv: 1601.07913 [hep-ex] (Cited on pages 50, 58)

[64] Pablo De Castro and Tommaso Dorigo. "INFERNO: Inference-Aware Neural Optimisation". *Comput. Phys. Commun.* 244 (2019), pp. 170–179. DOI: 10.1016/j.cpc.2019.06.007. arXiv: 1806.04743 [stat.ML] (Cited on pages 50, 58, 152)

[65] Anders Andreassen, Patrick T. Komiske, Eric M. Metodiev, Benjamin Nachman, et al. "OmniFold: A Method to Simultaneously Unfold All Observables". *Phys. Rev. Lett.* 124.18 (2020), p. 182001. DOI: 10.1103/PhysRevLett.124.182001. arXiv: 1911.09107 [hep-ph] (Cited on pages 50, 58)

[66] Johann Brehmer, Kyle Cranmer, Gilles Louppe, and Juan Pavez. "A Guide to Constraining Effective Field Theories with Machine Learning". *Phys. Rev. D* 98.5 (2018), p. 052004. DOI: 10.1103/PhysRevD.98.052004. arXiv: 1805.00020 [hep-ph] (Cited on pages 50, 58, 59, 61–63, 152, 161, 162, 164, 172)

[67] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, et al. "Improved Techniques for Training GANs". *CoRR* abs/1606.03498 (2016). arXiv: 1606.03498. URL: http://arxiv.org/abs/1606.03498 (Cited on pages 54, 86)

[68] M. Arjovsky, S. Chintala, and L. Bottou. "Wasserstein GAN". *ArXiv e-prints* (Jan. 2017). arXiv: 1701.07875 [stat.ML] (Cited on pages 54–56, 84)

[69] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, et al. "Improved Training of Wasserstein GANs" (2017). arXiv: 1704.00028 (Cited on pages 54, 55, 84, 202)

[70] Tong Che, Yanran Li, Athul Paul Jacob, Yoshua Bengio, et al. "Mode Regularized Generative Adversarial Networks". *CoRR* abs/1612.02136 (2016). arXiv: 1612.02136. URL: http://arxiv.org/abs/1612.02136 (Cited on page 55)

[71] Xiang Wei, Boqing Gong, Zixia Liu, Wei Lu, et al. "Improving the Improved Training of Wasserstein GANs: A Consistency Term and Its Dual Effect". *CoRR* abs/1803.01541 (2018). arXiv: 1803.01541. URL: http://arxiv.org/abs/1803.01541 (Cited on pages 56, 94)

[72] Karol Kurach, Mario Lucic, Xiaohua Zhai, Marcin Michalski, et al. "The GAN Landscape: Losses, Architectures, Regularization, and Normalization". *CoRR* abs/1807.04720 (2018). arXiv: 1807.04720. URL: http://arxiv.org/abs/1807.04720 (Cited on page 56)

[73] Andrew Brock, Jeff Donahue, and Karen Simonyan. "Large Scale GAN Training for High Fidelity Natural Image Synthesis". *CoRR* abs/1809.11096 (2018). arXiv: 1809.11096. URL: http://arxiv.org/abs/1809.11096 (Cited on pages 56, 94)

[74] David Berthelot, Tom Schumm, and Luke Metz. "BEGAN: Boundary Equilibrium Generative Adversarial Networks". *CoRR* abs/1703.10717 (2017). arXiv: 1703.10717. URL: http://arxiv.org/abs/1703.10717 (Cited on pages 56, 57)

[75] Kyle Cranmer, Juan Pavez, and Gilles Louppe. "Approximating Likelihood Ratios with Calibrated Discriminative Classifiers" (June 2015). arXiv: 1506.02169 [stat.AP] (Cited on page 57)

[76] Pierre Baldi, Peter Sadowski, and Daniel Whiteson. "Searching for Exotic Particles in High-Energy Physics with Deep Learning". *Nature Commun.* 5 (2014), p. 4308. DOI: 10.1038/ncomms5308. arXiv: 1402.4735 [hep-ph] (Cited on page 58)

[77] M. Erdmann, E. Geiser, Y. Rath, and M. Rieger. "Lorentz Boost Networks: Autonomous Physics-Inspired Feature Engineering". *JINST* 14.06 (2019), P06006. DOI: 10.1088/1748-0221/14/06/P06006. arXiv: 1812.09722 [hep-ex] (Cited on page 58)

[78] Johann Brehmer, Kyle Cranmer, Gilles Louppe, and Juan Pavez. "Constraining Effective Field Theories with Machine Learning". *Phys. Rev. Lett.* 121.11 (2018), p. 111801. DOI: 10.1103/PhysRevLett.121.111801. arXiv: 1805.00013 [hep-ph] (Cited on pages 58, 59, 161)

[79] Johann Brehmer, Gilles Louppe, Juan Pavez, and Kyle Cranmer. "Mining gold from implicit models to improve likelihood-free inference". *Proc. Nat. Acad. Sci.* 117.10 (2020), pp. 5242–5249. DOI: 10.1073/pnas.1915980117. arXiv: 1805.12244 [stat.ML] (Cited on pages 59, 60, 161)

[80] Johann Brehmer, Felix Kling, Irina Espejo, and Kyle Cranmer. "MadMiner: Machine learning-based inference for particle physics". *Comput. Softw. Big Sci.* 4.1 (2020), p. 3. DOI: 10.1007/s41781-020-0035-2. arXiv: 1907.10621 [hep-ph] (Cited on pages 59, 161)

[81] Johann Brehmer, Felix Kling, Irina Espejo, and Kyle Cranmer. *MadMiner*. DOI: 10.5281/zenodo.1489147. URL: https://github.com/diana-hep/madminer (Cited on pages 60, 164)

[82] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, et al. "The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations". *JHEP* 07 (2014), p. 079. DOI: 10.1007/JHEP07(2014)079. arXiv: 1405.0301 [hep-ph] (Cited on pages 60, 141, 163)

[83] Torbjorn Sjostrand, Stefan Ask, Jesper R. Christiansen, Richard Corke, et al. "An Introduction to PYTHIA 8.2". *Comput. Phys. Commun.* 191 (2015), p. 159. DOI: 10.1016/j.cpc.2015.01.024. arXiv: 1410.3012 [hep-ph] (Cited on pages 62, 141, 163)

[84] J. de Favereau, C. Delaere, P. Demin, A. Giammanco, et al. "DELPHES 3, A modular framework for fast simulation of a generic collider experiment". *JHEP* 02 (2014), p. 057. DOI: 10.1007/JHEP02(2014)057. arXiv: 1307.6346 [hep-ex] (Cited on pages 62, 163, 204)

[85] Heinrich, Lukas and Feickert, Matthew and Stark, Giordon. *pyhf: v0.5.1*. Version 0.5.1. DOI: 10.5281/zenodo.1169739. URL: https://github.com/scikit-hep/pyhf (Cited on page 62)

[86] Markus Stoye, Johann Brehmer, Gilles Louppe, Juan Pavez, et al. "Likelihood-free inference with an improved cross-entropy estimator" (Aug. 2018). arXiv: 1808.00973 [stat.ML] (Cited on page 62)

[87] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, et al. Curran Associates, Inc., 2017, pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf (Cited on page 65)

[88] Glen Cowan, Kyle Cranmer, Eilam Gross, and Ofer Vitells. "Asymptotic formulae for likelihood-based tests of new physics". *Eur. Phys. J. C* 71 (2011), p. 1554. DOI: 10.1140/epjc/s10052-011-1554-0. arXiv: 1007.1727 [physics.data-an] (Cited on pages 68, 173). Erratum: *Eur. Phys. J. C* 73 (2013), p. 2501. DOI: 10.1140/epjc/s10052-013-2501-z

[89] The ATLAS Collaboration. "The ATLAS Simulation Infrastructure". *Eur. Phys. J. C* 70 (2010), pp. 823–874. DOI: 10.1140/epjc/s10052-010-1429-9. arXiv: 1005.4568 [physics.ins-det] (Cited on page 71)

[90] Ahmed Hasib and Jana Schaarschmidt. *The new Fast Calorimeter Simulation in ATLAS*. Tech. rep. ATL-COM-SOFT-2018-080. Geneva: CERN, June 2018. URL: https://cds.cern.ch/record/2626157 (Cited on pages 71, 75–77, 202)

[91] The ATLAS Collaboration. *CPU resource estimate 2017*. URL: https://twiki.cern.ch/twiki/pub/AtlasPublic/ComputingandSoftwarePublicResults/cpuHLLHC.pdf (Cited on page 72)

[92] *Deep generative models for fast shower simulation in ATLAS*. Tech. rep. ATL-SOFT-PUB-2018-001. Geneva: CERN, July 2018. URL: http://cds.cern.ch/record/2630433 (Cited on pages 73, 78, 86, 87, 95–97, 136, 196, 203)

[93] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. "Searching for Activation Functions". *CoRR* abs/1710.05941 (2017). arXiv: 1710.05941. URL: http://arxiv.org/abs/1710.05941 (Cited on pages 83, 84, 93)

[94] François Chollet et al. *Keras*. https://keras.io. 2015 (Cited on page 85)

[95] Martin Erdmann, Jonas Glombitza, and Thorben Quast. "Precise simulation of electromagnetic calorimeter showers using a Wasserstein Generative Adversarial Network". *Comput. Softw. Big Sci.* 3.1 (2019), p. 4. DOI: 10.1007/s41781-018-0019-7. arXiv: 1807.01954 [physics.ins-det] (Cited on pages 91, 133)

[96] Anja Butter, Tilman Plehn, and Ramon Winterhalder. "How to GAN LHC Events". *SciPost Phys.* 7.6 (2019), p. 075. DOI: 10.21468/SciPostPhys.7.6.075. arXiv: 1907.03764 [hep-ph] (Cited on page 91)

[97] Erik Buhmann, Sascha Diefenbacher, Engin Eren, Frank Gaede, et al. "Getting High: High Fidelity Simulation of High Granularity Calorimeters with High Speed" (May 2020). arXiv: 2005.05334 [physics.ins-det] (Cited on page 91)

[98] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. *Progressive Growing of GANs for Improved Quality, Stability, and Variation*. 2017. arXiv: 1710.10196 [cs.NE] (Cited on page 94)

[99] Lars M. Mescheder. "On the convergence properties of GAN training". *CoRR* abs/1801.04406 (2018). arXiv: 1801.04406. URL: http://arxiv.org/abs/1801.04406 (Cited on page 94)

[100] *The new Fast Calorimeter Simulation in ATLAS*. Tech. rep. ATL-SOFT-PUB-2018-002. Geneva: CERN, July 2018. URL: https://cds.cern.ch/record/2630434 (Cited on page 94)

[101] Daniel Hay Guest, Joshua Wyatt Smith, Michela Paganini, Michael Kagan, et al. *lwtnn/lwtnn: Version 2.9*. Version v2.9. June 2019. DOI: 10.5281/zenodo. URL: https://doi.org/10.5281/zenodo.3249317 (Cited on page 94)

[102] The ATLAS Collaboration. "Electron and photon performance measurements with the ATLAS detector using the 2015–2017 LHC proton-proton collision data". *JINST* 14.12 (2019), P12006. DOI: 10.1088/1748-0221/14/12/P12006. arXiv: 1908.00005 [hep-ex] (Cited on pages 116, 117)

[103] Vallecorsa, Sofia, Carminati, Federico, and Khattak, Gulrukh. "3D convolutional GAN for fast simulation". *EPJ Web Conf.* 214 (2019), p. 02010. DOI: 10.1051/epjconf/201921402010. URL: https://doi.org/10.1051/epjconf/201921402010 (Cited on page 133)

[104] Pasquale Musella and Francesco Pandolfi. "Fast and Accurate Simulation of Particle Detectors Using Generative Adversarial Networks". *Comput. Softw. Big Sci.* 2.1 (2018), p. 8. DOI: 10.1007/s41781-018-0015-y. arXiv: 1805.00850 [hep-ex] (Cited on page 133)

[105] ATLAS Collaboration. "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC". *Phys. Lett. B* 716 (2012), p. 1. DOI: 10.1016/j.physletb.2012.08.020. arXiv: 1207.7214 [hep-ex] (Cited on page 137)

[106] CMS Collaboration. "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC". *Phys. Lett. B* 716 (2012), p. 30. DOI: 10.1016/j.physletb.2012.08.021. arXiv: 1207.7235 [hep-ex] (Cited on page 137)

[107] S Heinemeyer, C Mariotti, G Passarino, R Tanaka, et al. *Handbook of LHC Higgs Cross Sections: 3. Higgs Properties: Report of the LHC Higgs Cross Section Working Group*. Ed. by S Heinemeyer. CERN Yellow Reports: Monographs. Comments: 404 pages, 139 figures, to be submitted to CERN Report. Working Group web page: https://twiki.cern.ch/twiki/bin/view/LHCPh July 2013. DOI: 10.5170/CERN-2013-004. URL: https://cds.cern.ch/record/1559921 (Cited on page 138)

[108] ATLAS Collaboration. "The ATLAS Simulation Infrastructure". *Eur. Phys. J. C* 70 (2010), p. 823. DOI: 10.1140/epjc/s10052-010-1429-9. arXiv: 1005.4568 [physics.ins-det] (Cited on page 141)

[109] T. Gleisberg, S. Höche, F. Krauss, M. Schönherr, et al. "Event generation with SHERPA 1.1". *JHEP* 02 (2009), p. 007. DOI: 10.1088/1126-6708/2009/02/007. arXiv: 0811.4622 [hep-ph] (Cited on page 141)

[110] Marek Schönherr and Frank Krauss. "Soft Photon Radiation in Particle Decays in SHERPA". *JHEP* 12 (2008), p. 018. DOI: 10.1088/1126-6708/2008/12/018. arXiv: 0810.5071 [hep-ph] (Cited on page 141)

[111] Fabio Cascioli, Philipp Maierhofer, and Stefano Pozzorini. "Scattering Amplitudes with Open Loops". *Phys. Rev. Lett.* 108 (2012), p. 111601. DOI: 10.1103/PhysRevLett.108.111601. arXiv: 1111.5206 [hep-ph] (Cited on page 141)

[112] F. Cascioli, S. Höche, F. Krauss, P. Maierhöfer, et al. "Precise Higgs-background predictions: merging NLO QCD and squared quark-loop corrections to four-lepton + 0,1 jet production". *JHEP* 01 (2014), p. 046. DOI: 10.1007/JHEP01(2014)046. arXiv: 1309.0500 [hep-ph] (Cited on page 141)

[113] Richard D. Ball et al. "Parton distributions for the LHC Run II". *JHEP* 04 (2015), p. 040. DOI: 10.1007/JHEP04(2015)040. arXiv: 1410.8849 [hep-ph] (Cited on page 141)

[114] S. Dittmaier et al. "Handbook of LHC Higgs Cross Sections: 2. Differential Distributions" (Jan. 2012). DOI: 10.5170/CERN-2012-002. arXiv: 1201.3084 [hep-ph] (Cited on page 141)

[115] Richard D. Ball et al. "Parton distributions with LHC data". *Nucl. Phys. B* 867 (2013), p. 244. DOI: 10.1016/j.nuclphysb.2012.10.003. arXiv: 1207.1303 [hep-ph] (Cited on page 141)

[116] Steffen Schumann and Frank Krauss. "A Parton shower algorithm based on Catani-Seymour dipole factorisation". *JHEP* 03 (2008), p. 038. DOI: 10.1088/1126-6708/2008/03/038. arXiv: 0709.1027 [hep-ph] (Cited on page 144)

[117] B. Biedermann, A. Denner, S. Dittmaier, L. Hofer, et al. "Electroweak corrections to $pp \to \mu^+\mu^- e^+ e^- + X$ at the LHC: a Higgs background study". *Phys. Rev. Lett.* 116.16 (2016), p. 161803. DOI: 10.1103/PhysRevLett.116.161803. arXiv: 1601.07787 [hep-ph] (Cited on page 144)

[118] Benedikt Biedermann, Marina Billoni, Ansgar Denner, Stefan Dittmaier, et al. "Next-to-leading-order electroweak corrections to $pp \to W^+W^- \to 4$ leptons at the LHC". *JHEP* 06 (2016), p. 065. DOI: 10.1007/JHEP06(2016)065. arXiv: 1605.03419 [hep-ph] (Cited on page 144)

[119] Thomas Keck. "FastBDT: A speed-optimized and cache-friendly implementation of stochastic gradient-boosted decision trees for multivariate classification". *CoRR* abs/1609.06119 (2016). arXiv: 1609.06119. URL: http://arxiv.org/abs/1609.06119 (Cited on page 145)

[120] Andreas Hocker, Peter Speckmayer, Jorg Stelzer, Jan Therhaag, et al. *TMVA - Toolkit for Multivariate Data Analysis with ROOT: Users guide. TMVA - Toolkit for Multivariate Data Analysis*. Tech. rep. physics/0703039. TMVA-v4 Users Guide: 135 pages, 19 figures, numerous code examples and references. Geneva: CERN, Mar. 2007. URL: https://cds.cern.ch/record/1019880 (Cited on page 145)

[121] Benjamin Nachman and Jesse Thaler. "Neural resampler for Monte Carlo reweighting with preserved uncertainties". *Phys. Rev. D* 102.7 (2020), p. 076004. DOI: 10.1103/PhysRevD.102.076004. arXiv: 2007.11586 [hep-ph] (Cited on page 149)

[122] Mathias Backes, Anja Butter, Tilman Plehn, and Ramon Winterhalder. "How to GAN Event Unweighting" (Dec. 2020). arXiv: 2012.07873 [hep-ph] (Cited on page 149)

[123] The ATLAS Collaboration. "Search for Heavy Higgs Bosons A/H Decaying to a Top Quark Pair in pp Collisions at $\sqrt{s} = 8$ TeV with the ATLAS Detector". *Phys. Rev. Lett.* 119.19 (2017), p. 191803. DOI: 10.1103/PhysRevLett.119.191803. arXiv: 1707.06025 [hep-ex] (Cited on page 152)

[124] The ATLAS Collaboration. "Probing the quantum interference between singly and doubly resonant top-quark production in pp collisions at $\sqrt{s} = 13$ TeV with the ATLAS detector". *Phys. Rev. Lett.* 121.15 (2018), p. 152002. DOI: 10.1103/PhysRevLett.121.152002. arXiv: 1806.04667 [hep-ex] (Cited on page 152)

[125] The ATLAS Collaboration. "Evidence for the Higgs-boson Yukawa coupling to tau leptons with the ATLAS detector". *JHEP* 04 (2015), p. 117. DOI: 10.1007/JHEP04(2015)117. arXiv: 1501.04943 [hep-ex] (Cited on page 158)

[126] *A morphing technique for signal modelling in a multidimensional space of coupling parameters*. Tech. rep. ATL-PHYS-PUB-2015-047. Geneva: CERN, Nov. 2015. URL: https://cds.cern.ch/record/2066980 (Cited on page 163)

[127] S. Ovyn, X. Rouby, and V. Lemaitre. "DELPHES, a framework for fast simulation of a generic collider experiment" (Mar. 2009). arXiv: 0903.2225 [hep-ph] (Cited on page 163)

[128] Lukas, Matthew Feickert, Giordon Stark, Ruggero Turra, et al. *diana-hep/pyhf: v0.1.2*. Version v0.1.2. July 2019. DOI: 10.5281/zenodo.3334365. URL: https://doi.org/10.5281/zenodo.3334365 (Cited on page 180)

[129] The ATLAS Collaboration. "Search for Higgs boson pair production in the $\gamma\gamma b\bar{b}$ final state with 13 TeV pp collision data collected by the ATLAS experiment". *JHEP* 11 (2018), p. 040. DOI: 10.1007/JHEP11(2018)040. arXiv: 1807.04873 [hep-ex] (Cited on page 181)

[130] Alaa Saade, Francesco Caltagirone, Igor Carron, Laurent Daudet, et al. "Random Projections through multiple optical scattering: Approximating kernels at the speed of light". *CoRR* abs/1510.06664 (2015). arXiv: 1510.06664. URL: http://arxiv.org/abs/1510.06664 (Cited on page 183)

[131] *Performance of mass-decorrelated jet substructure observables for hadronic two-body decay tagging in ATLAS*. Tech. rep. ATL-PHYS-PUB-2018-014. Geneva: CERN, July 2018. URL: https://cds.cern.ch/record/2630973 (Cited on pages 183, 184, 204)

[132] Chase Shimmin, Peter Sadowski, Pierre Baldi, Edison Weik, et al. "Decorrelated Jet Substructure Tagging using Adversarial Neural Networks". *Phys. Rev. D* 96.7 (2017), p. 074034. DOI: 10.1103/PhysRevD.96.074034. arXiv: 1703.03507 [hep-ex] (Cited on pages 183, 184, 187, 204)

[133] Gilles Louppe, Michael Kagan, and Kyle Cranmer. "Learning to Pivot with Adversarial Networks" (Nov. 2016). arXiv: 1611.01046 [stat.ML] (Cited on pages 184, 204)

[134] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, et al. "Scikit-learn: Machine Learning in Python". *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (Cited on page 186)

[135] *Dataset from the ATLAS Higgs Boson Machine Learning Challenge 2014*. 2014. DOI: 10.7483/OPENDATA.ATLAS.ZBP2.M5T8 (Cited on page 192)

[136] Victor Estrade, Cécile Germain, Isabelle Guyon, and David Rousseau. "Systematic aware learning". *EPJ Web of Conferences* 214 (2019). Ed. by A. Forti, L. Betev, M. Litmaath, O. Smirnova, et al., p. 06024. DOI: 10.1051/epjconf/201921406024. URL: https://doi.org/10.1051/epjconf/201921406024 (Cited on page 192)

université PARIS-SACLAY | ÉCOLE DOCTORALE PHENIICS

Titre: Simulation du calorimètre électromagnétique de ATLAS à l'aide de Réseaux Antagonistes Génératifs et mesure au LHC des couplages du boson de Higgs hors résonance par inférence sans Fonction de Vraisemblance

Mots clés: Boson de Higgs hors résonance, Inférence Sans Fonction de Vraisemblance, Particule virtuelle, Modèles génératifs Intelligence Artificielle, CERN

Résumé: Depuis la découverte du boson de Higgs en 2012, les expériences du LHC testent les prévisions du modèle standard avec des mesures de haute précision. Les mesures des couplages du boson de Higgs hors résonance permettront d'éliminer certaines dégénérescences qui ne peuvent pas être résolues avec les mesures sur résonance, comme la sonde de la largeur du boson de Higgs, ce qui pourrait donner des indications pour la nouvelle physique.

Une partie de cette thèse se concentre sur la mesure des couplages hors résonance du boson de Higgs produit par la fusion du boson vecteur et se décomposant en quatre leptons. Ce canal de désintégration offre une occasion unique de sonder le boson de Higgs dans son régime hors résonance grâce à des sections efficaces augmentées au-delà de $2M_z$ (deux fois la masse du boson Z) de la région des quatre leptons. L'importante interférence quantique entre le signal et les processus de fond rend le concept d'"étiquettes de classe" mal défini, et pose un défi aux méthodes traditionnelles et aux modèles génériques de classification par apprentissage machine utilisés pour optimiser une mesure de la force du signal. Une nouvelle famille de stratégies d'inférence sans fonction de vraisemblance basées sur l'apprentissage machine, qui exploitent des informations supplémentaires pouvant être extraites du simulateur, a été adaptée à un problème de mesure de la force du signal. L'étude montre des résultats prometteurs par rapport aux techniques de base sur un ensemble de données de simulation rapide avec Delphes. Dans ce contexte, on a également introduit le réseau aspiration, un algorithme d'adverse amélioré pour la formation tout en maintenant l'invariance par rapport aux caractéristiques choisies.

Les mesures de l'expérience ATLAS reposent sur de grandes quantités de données simulées précisemment. Le logiciel de simulation actuel de Geant4 est trop coûteux en termes de calculs pour supporter la grande quantité de données simulées nécessaires aux analyses futures prévues.

Autre partie de cette thèse se concentre sur une nouvelle approche de la simulation rapide utilisant un réseau advers génératif (GAN). La simulation de gerbe en cascade du calorimètre complexe d'ATLAS est la partie la plus lente de la chaîne de simulation utilisant Geant4. Son remplacement par un réseau de neurones qui a appris la distribution de probabilité des gerbes de particules en fonction des propriétés des particules incidentes et de la géométrie locale du détecteur augmente la vitesse de simulation de plusieurs ordres de grandeur, même sur des CPU à cœur unique, et ouvre la porte à une accélération supplémentaire sur les GPU. L'intégration dans le logiciel ATLAS permet pour la première fois de faire des comparaisons réalistes avec des simulation rapide paramètrées "à la main". L'étude est réalisée sur une petite section du détecteur ($0,20 < |\eta| < 0,25$) en utilisant des photons et compare les distributions en utilisant des échantillons simulés par le modèle autonome ainsi qu'après intégration dans le logiciel ATLAS avec des échantillons Geant4 entièrement simulés. Des leçons importantes sur les mérites et les inconvénients des différentes stratégies, profitent à l'objectif ultime de simuler l'ensemble du calorimètre ATLAS avec des modèles générateurs profonds. L'étude révèle également un problème inhérent à le GAN de Wasserstein basé sur une pénalité de gradient, et propose une solution.

Title: Simulation of the ATLAS Electromagnetic Calorimeter using Generative Adversarial Networks and Likelihood-Free Inference of the Offshell Higgs Boson Couplings at the LHC

Keywords: Higgs Boson, Virtual Particle, Likelihood-Free Inference, Generative Models, Artificial Intelligence, CERN

Abstract: Since the discovery of the Higgs boson in 2012, experiments at the LHC have been testing Standard Model predictions with high precision measurements. Measurements of the off-shell couplings of the Higgs boson will remove certain degeneracies that cannot be resolved with the current on-shell measurements, such as probing the Higgs boson width, which may lead to hints for new physics.

One part of this thesis focuses on the measurement of the off-shell couplings of the Higgs boson produced by vector boson fusion and decaying to four leptons. This decay channel provides a unique opportunity to probe the Higgs in its off-shell regime due to enhanced cross-sections beyond $2M_z$ (twice the mass of the Z boson) region of the four lepton mass. The significant quantum interference between the signal and background processes renders the concept of 'class labels' ill-defined, and poses a challenge to traditional methods and generic machine learning classification models used to optimise a signal strength measurement. A new family of machine learning based likelihood-free inference strategies, which leverage additional information that can be extracted from the simulator, were adapted to a signal strength measurement problem. The study shows promising results compared to baseline techniques on a fast simulated Delphes dataset. Also introduced in this context is the aspiration network, an improved adversarial algorithm for training while maintaining invariance with respect to chosen features.

Measurements in the ATLAS experiment rely on large amounts of precise simulated data. The current Geant4 simulation software is computationally too expensive to sustain the large amount of simulated data required for planned future analyses.

The other part of this thesis focuses on a new approach to fast simulation using a Generative Adversarial Network (GAN). The cascading shower simulation of the complex ATLAS calorimeter is the slowest part of the simulation chain using Geant4. Replacing it with a neural network that has learnt the probability distribution of the particle showers as a function of the incident particle properties and local detector geometry increases the simulation speed by several orders of magnitude, even on single core CPUs, and opens to door the further speed up on GPUs. The integration into the ATLAS software allows for the first time to make realistic comparisons to hand-designed fast simulation frameworks. The study is performed on a small section of the detector ($0.20 < |\eta| < 0.25$) using photons and compares distributions using samples simulated by the model standalone as well as after integration into the ATLAS software against fully simulated Geant4 samples. Important lessons on the merits and demerits of various strategies, benefit the ultimate goal of simulating the entire ATLAS calorimeter with a few deep generative models. The study also reveals an inherent problem with the popular gradient penalty based Wasserstein GAN, and proposes a solution.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France