

UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di Fisica e Astronomia “Galileo Galilei”

Master Degree in Physics

Final Dissertation

Study of b-quark production asymmetry with quantum
machine learning techniques at the LHCb experiment

Thesis supervisor

Prof. Donatella Lucchesi

Thesis co-supervisor

Dr. Lorenzo Sestini

Candidate

Davide Nicotra

Academic Year 2020/2021

Contents

Introduction	1
1 Theoretical introduction	3
1.1 The Standard Model of particle physics	3
1.1.1 Quantum Chromodynamics	3
1.1.2 Electroweak theory	4
1.2 b -quark production in proton-proton collisions	6
1.3 Hadronization of b -quarks	6
1.4 b -quark pair-production asymmetry	8
1.4.1 $b\bar{b}$ asymmetry measurement at e^+e^- colliders	9
1.4.2 $b\bar{b}$ asymmetry measurement at $p\bar{p}$ colliders	10
1.4.3 $b\bar{b}$ asymmetry measurement at pp colliders	11
1.4.4 New Physics in b -quark asymmetry	11
2 The LHCb experiment and jet identification	13
2.1 The Large Hadron Collider	13
2.2 The LHCb detector	13
2.2.1 The tracking system	14
2.2.2 Tracking performance	16
2.2.3 Magnet	18
2.2.4 Ring Imaging Cherenkov detectors	18
2.2.5 Calorimeters	19
2.2.6 Muon System	20
2.3 Trigger	20
2.3.1 Level 0 trigger	20
2.3.2 High Level Trigger	21
2.4 Jet reconstruction	22
2.5 Heavy-Flavor jet identification	25
2.5.1 Secondary Vertex tagging	26
2.5.2 Muon Tagging	27
2.5.3 Jet Charge tagging	28
2.6 Development of new tagging algorithms	29
2.7 Data-set selection	29
3 Quantum Computing and Quantum Machine Learning	31
3.1 Introduction to Quantum Computation	31
3.2 The Qubit	32
3.3 Entanglement	35
3.4 Quantum circuits	35
3.5 Machine Learning with quantum circuits	37
3.5.1 Introduction to Machine Learning	37
3.5.2 Supervised learning	37

3.6	Parametrized quantum circuits	39
4	Results on b-tagging with Quantum Machine Learning	43
4.1	Training data-set selection	43
4.2	Quantum Machine Learning models	44
4.2.1	Amplitude Encoding classifier	44
4.2.2	Angle Encoding classifier	44
4.2.3	Data re-uploading classifier	45
4.2.4	Tree Topology classifier	46
4.2.5	Quantum Support Vector Machine classifier	47
4.3	Classical Deep Neural Network model	48
4.4	Software implementation	49
4.4.1	PennyLane	49
4.4.2	IBM Qiskit	51
4.5	Training	51
4.6	Results	54
4.6.1	Muon events	54
4.6.2	Full data-set	57
5	Asymmetry measurement and estimation of its uncertainty	61
5.1	Data selection	61
5.2	Asymmetry measurement and dilution correction	62
5.2.1	Uncertainty on the asymmetry	62
5.2.2	Asymmetry in the Monte Carlo simulated data	64
5.3	Comparison among methods	64
5.3.1	Results on the muon data-set	64
5.3.2	Results on the full data-set	67
6	Conclusions and future developments	75
6.1	Conclusions	75
6.1.1	Final considerations on Quantum Machine Learning	75
6.2	Future Development	76
6.2.1	New quantum models and use of large data-sets	76
6.2.2	Quantum noise	76
6.2.3	Quantum hardware	77
6.2.4	Search for particles correlations inside jets	78
	Bibliography	85

Introduction

The precise determination of the $b\bar{b}$ pair-production asymmetry is important not only as a test of the Standard Model of Particle Physics but also to investigate possible sources of New Physics. At the LHCb experiment, that takes data at the LHC proton-proton collider at CERN, where b -quarks are copiously produced, the Forward-Central asymmetry in the $b\bar{b}$ pair-production, $A_{b\bar{b}}^{\text{FC}}$, has been measured at the center of mass energy of $\sqrt{s} = 7$ TeV. The result is consistent with the Standard Model expectation within the experimental error. However, additional measurements of such an observable need to be performed at higher energies and with an increased number of events, in order to reduce the uncertainties and to put constraints on New Physics models. The final uncertainty on $A_{b\bar{b}}^{\text{FC}}$ is strongly related to the ability of identifying the flavor (u, d, s vs c vs b) and the charge (b vs \bar{b}) of the quark producing the jet in the detector (*jet tagging*). At the LHCb experiment, jet tagging exploits the excellent capability of the detector to precisely reconstruct vertices displaced with respect to the interaction point, which allow to distinguish the long-lived heavy quarks from light quarks jets. A standard technique, called *Muon Tagging*, performs charge tagging by exploiting the charge correlation between the b -quark and the muon produced in the semi-leptonic decay. However, the small branching ratio of this decay strongly limits the efficiency of the algorithm. For this reason other inclusive algorithms, based on Machine Learning techniques, are being developed, which aim to exploit the full jet particles substructure.

In this thesis, I present new approaches to the charge tagging of b -jets, based on Quantum Machine Learning techniques: as a general paradigm, data are embedded in a quantum circuit through a quantum feature map; then the initial state gets processed by a variational quantum circuit with trainable parametrized gates; finally, measurements of observables on the final state are mapped to a binary classification label (b -jet or \bar{b} -jet). The models are trained on official LHCb simulated data ($\sqrt{s} = 13$ TeV) and the tagging performance is compared with the Muon Tagging algorithm and a classical Deep Neural Network model.

Finally, the precision on the Forward-Central asymmetry, $A_{b\bar{b}}^{\text{FC}}$, is evaluated applying the different tagging algorithms on a sample of simulated data corresponding to the integrated luminosity 6 fb^{-1} of the Run2.

The results of this brand new technique are presented and discussed in this thesis, which is structured as follows:

- **Chapter 1:** a brief overview of the quark sector of the Standard Model (SM), the Electroweak theory and of the Quantum Chromodynamics is given. The b -quark production mechanisms at colliders are presented as well as the hadronization process, that generates jets. Then, the origin of the asymmetries of heavy quarks production within the SM and their relationship with New Physics are discussed. Finally, the state of the art of the measurements of $b\bar{b}$ pair-production asymmetries at colliders is presented.
- **Chapter 2:** the LHCb experiment is presented. I start describing the LHC accelerator facility, then I describe the LHCb detector and its sub-systems. Then, the jets reconstruction algorithm is described. I conclude with the discussion of the problem of jet identification: standard algorithms for flavor and charge tagging of b -jets are described.
- **Chapter 3:** this chapter introduces the concept of quantum computations. In the first place,

I give a theoretical overview of the qubit and entanglement concepts. Then, I introduce quantum circuits, defined as collection of linked quantum gates. After a brief introduction to the concepts of Supervised Machine Learning, I introduce the topic of Quantum Machine Learning: Parametrized Quantum Circuits are defined and proposed as Machine Learning models.

- **Chapter 4:** I describe how I propose to solve the charge tagging problem, encoding it into a Quantum Machine Learning model. I explain how I chose the data-set used for the analysis in terms of variables and selections. Then, I discuss in details several quantum models, including the software implementation which makes intensive use of the Python libraries PennyLane, PyTorch and Tensorflow. A classical model, based on a Deep Neural Network, is also presented and used as a reference. Finally, the performance of my implementation of the quantum models is evaluated and compared to the Deep Neural Network and the classical Muon Tagging algorithm.
- **Chapter 5:** here, I measure the Forward-Central $b\bar{b}$ pair-production asymmetry on Monte Carlo simulated data, by using the algorithms discussed in Chapter 4. I present the method I chose to check if biases are introduced by the tagging methods. Then I compute the asymmetry using the taggers I implemented. The performance of the quantum models is compared with the those of the Muon Tagging and the Deep Neural Network models by comparing the statistical uncertainties achieved.
- **Chapter 6:** in the last chapter, a summary of the results I obtained on the asymmetry accuracy is presented and some final consideration on the Quantum Machine Learning approach to the b -tagging problem are made. Moreover, I discuss my ideas on the current state of the art of this technique, including limits, advantages a possible perspectives. Finally, I conclude with what I learnt on the feasibility of Quantum Machine Learning tasks, involving large and complex data-sets, and the actual usage of the current quantum hardware in the near future.

Chapter 1

Theoretical introduction

1.1 The Standard Model of particle physics

The Standard Model of particle physics (SM) is a gauge field theory that describes the electromagnetic, weak and strong interactions. It is based on the gauge group $G = SU(3)_C \otimes SU(2)_L \otimes U(1)_Y$. A massless gauge vector boson is associated to each generator of G , however, the electroweak bosons W^\pm and Z^0 acquire mass by mean of a spontaneous symmetry breaking mechanism of G called the Higgs mechanism. The fermionic particle content of the SM consists in leptons and quarks: leptons consist of charged leptons, which interact electromagnetically and weakly, and their corresponding neutrinos, which interact only weakly. Quarks form the hadronic matter and interact electromagnetically, weakly and strongly. A schematic overview of the particle content of the SM is provided in Figure [L.1](#).

The next sections will provide an overview of the Quantum Chromodynamics and Electroweak sectors of the SM.

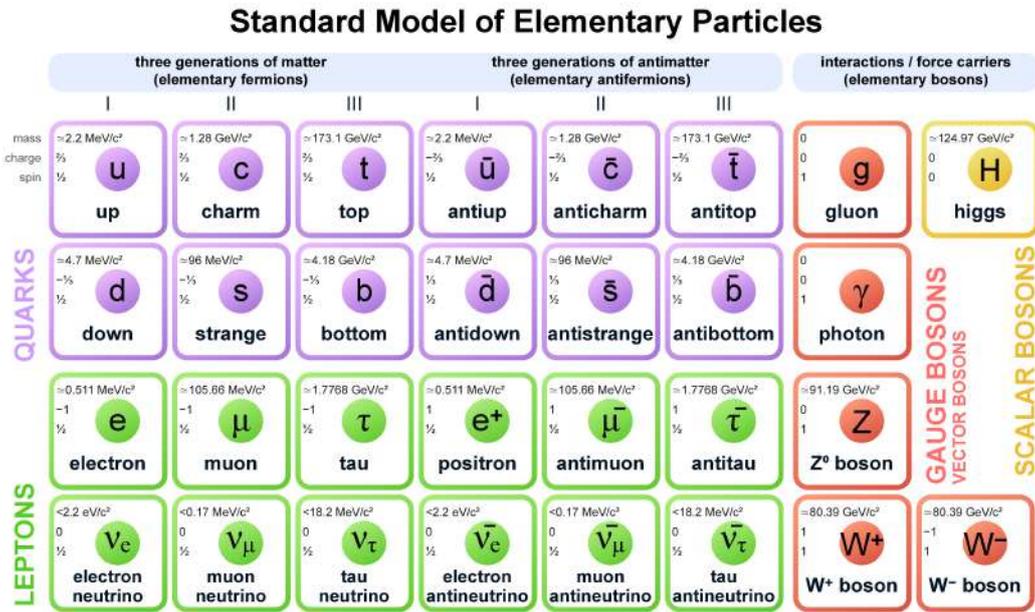


Figure 1.1: Particle content of the Standard Model [\[28\]](#).

1.1.1 Quantum Chromodynamics

The Quantum Chromodynamics (QCD) [\[41\]](#) is the gauge field theory of strong interaction, based on the unbroken gauge color group $G_{\text{QCD}} = SU(3)_C$. A gauge massless vector boson, called gluon, is associated each one of the eight generator of the G_{QCD} group, which is left unbroken by the spontaneous symmetry breaking induced by the Higgs field. The matter fermionic fields of the QCD are the quarks,

which are the only colour-charged fermions in the particle content of the SM. The QCD Lagrangian density is the following

$$\mathcal{L} = -\frac{1}{4} \sum_{i=1}^8 F^{i\mu\nu} F_{\mu\nu}^i + \sum_{j=1}^{n_f} \bar{q}_j (iD_\mu \gamma^\mu - m_j) q_j$$

where q_j are quarks of $n_f = 6$ different flavor with masses m_j . The covariant derivative D_μ encodes the interactions between quarks q_i and gluons g_μ^j

$$D_\mu = \partial_\mu + ie_s \sum_{j=1}^8 t^j g_\mu^j$$

where t_j are the generator of the $SU(3)$ group in the triplet representation of quarks. The commutation algebra is $[t^i, t^j] = iC_{ijk}t^k$, where C_{ijk} are the completely anti-symmetric structure constants of the $SU(3)$ algebra. The interaction is weighted by e_s , that is the gauge coupling of the strong interaction, and in analogy with the QED, we can define $\alpha_s = e_s^2/4\pi$.

The kinetic terms of the gauge vector bosons are defined as follow

$$F_{\mu\nu}^i = \partial_\mu g_\nu^i - \partial_\nu g_\mu^i - e_s C_{ijk} g_\mu^j g_\nu^k$$

The non-abelian structure of the $SU(3)$ group results in three-gluons and four-gluons vertices.

Confinement The property that no colored particle is observed is called confinement. More precisely, only particles which are colorless or color singlets can be observed. This peculiar behaviour of the QCD has been studied by mean of lattice QCD methods [70], which showed, for example, that the effective potential of a $\bar{q}q$ system has the following form, known as the Cornell's potential [31]

$$V_{\bar{q}q} \propto \frac{\alpha_s(r)}{r} + \dots + \sigma r$$

The linearly increasing term of the potential makes energetically impossible to separate the two quarks. For example, if the two quarks are produced in the same point in the space-time, like in a proton-proton collision, in the center-of-mass frame, they start moving away from each other. At some point, it becomes energetically favorable to extract pairs of quarks from the sea which neutralize the color charge and allow the detection of the initial two quarks as a jet of colorless hadrons. This process is called hadronization.

1.1.2 Electroweak theory

The Electroweak (EW) theory [39, 63, 71] provides a unified description of the electromagnetic and weak interactions as a manifestation of the same force. It is built upon the gauge group $G_{EW} = SU(2)_L \otimes U(1)_Y$. Contrary to the QED case, in which the $U(1)_{EM}$ current couples to fermions independently from the chirality, the $SU(2)_L$ current only couples to left-handed fermions (and right-handed anti-fermions), defined as follows

$$\psi_L = P_L \psi = \frac{1 + \gamma^5}{2} \psi \quad \psi_R = P_R \psi = \frac{1 - \gamma^5}{2} \psi \quad \psi = \psi_L + \psi_R$$

Where P_L and P_R are, respectively, the left and right projector operators acting on Dirac spinor fields. The Lagrangian of the EW theory can be expressed as

$$\mathcal{L}_{EW} = \mathcal{L}_{\text{gauge}} + \mathcal{L}_{\text{Higgs}}$$

$\mathcal{L}_{\text{gauge}}$ is the Lagrangian associated to the Yang-Mills theory over the G_{EW} group

$$\begin{aligned} \mathcal{L}_{\text{gauge}} = & -\frac{1}{4} \sum_{i=1}^3 F^{i\mu\nu} F_{\mu\nu}^i - \frac{1}{4} B^{\mu\nu} B_{\mu\nu} + \\ & \bar{q}_L i\gamma^\mu D_\mu q_L + \bar{l}_L i\gamma^\mu D_\mu l_L + \\ & \bar{u}_R i\gamma^\mu D_\mu u_R + \bar{d}_R i\gamma^\mu D_\mu d_R + \\ & \bar{e}_R i\gamma^\mu D_\mu e_R + \bar{\nu}_R i\gamma^\mu D_\mu \nu_R \end{aligned}$$

In this case we have

$$B_{\mu\nu} = \partial_\mu B_\nu - \partial_\nu B_\mu \quad W_{\mu\nu}^i = \partial_\mu W_\nu^i - \partial_\nu W_\mu^i - g\epsilon_{ijk}W_\mu^jW_\nu^k$$

Where B_μ is the vector boson associated to the generator of $U(1)_Y$ abelian symmetry group, therefore $B_{\mu\nu}$ does not contain self-interaction terms, while W_μ^i are three vector bosons associated with the $SU(2)_L$ non-abelian symmetry group, therefore having a self-interaction term weighted by the $SU(2)_L$ coupling constant g and by the completely anti-symmetric tensor ϵ_{ijk} . The two fermionic chiralities behave differently under the G_{EW} group: the left-handed fermions behave like $SU(2)$ doublets

$$l_L = \begin{pmatrix} e \\ \nu_e \end{pmatrix}_L, \begin{pmatrix} \mu \\ \nu_\mu \end{pmatrix}_L, \begin{pmatrix} \tau \\ \nu_\tau \end{pmatrix}_L \quad q_L = \begin{pmatrix} u \\ d \end{pmatrix}_L, \begin{pmatrix} s \\ c \end{pmatrix}_L, \begin{pmatrix} t \\ b \end{pmatrix}_L$$

while the right-handed fermions behave like $SU(2)$ singlets

$$u_R = u_R, s_R, t_R \quad d_R = d_R, c_R, b_R \quad e_R = e_R, \mu_R, \tau_R \quad \nu_R = \nu_R^e, \nu_R^\mu, \nu_R^\tau$$

Finally, the covariant derivative is given by

$$D_\mu = \partial_\mu + ig \sum_{i=1}^3 t^i W_\mu^i + \frac{1}{2} ig' B_\mu$$

where t^i are the generator of $SU(2)_L$, obeying the commutation relations $[t^i, t^j] = i\epsilon^{ijk}t^k$ and g' is the $U(1)_Y$ coupling constant.

Yang-Mills gauge theories introduce massless gauge bosons associated to generators of the symmetry group, as mediators of the interaction. However, experimentally, the weak interaction appears to be short-ranged (that suggests a massive mediator) while the electromagnetic interaction appears to be long-ranged, mediated by the massless photon. This issue is solved by the Higgs mechanism [32]: the Higgs sector of the Lagrangian $\mathcal{L}_{\text{Higgs}}$, is defined as

$$\mathcal{L}_{\text{Higgs}} = (D_\mu \Phi)^\dagger (D^\mu \Phi) - V(\Phi^\dagger \Phi)$$

$$\Phi = \frac{1}{\sqrt{2}} \begin{pmatrix} \phi_1 + i\phi_2 \\ \phi_3 + i\phi_4 \end{pmatrix}$$

Φ is a complex scalar field, behaving as an $SU(2)_L$ doublet, which induces a Spontaneous Symmetry Breaking (SSB) of the G_{EW} group

$$\Phi = \frac{1}{\sqrt{2}} \begin{pmatrix} \phi_1 + i\phi_2 \\ \phi_3 + i\phi_4 \end{pmatrix} \longrightarrow \Phi^{SSB} = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ v + H \end{pmatrix}$$

$$G_{EW} = SU(2)_L \otimes U(1)_Y \longrightarrow G_{EW}^{SSB} = U(1)_Q$$

where $U(1)_Q$ is the symmetry group associated to the electromagnetic charge, which is left unbroken by the SSB: the massless vector boson associated to this residual symmetry is the photon A_μ , while the other vector bosons acquire masses proportional to the vacuum expectation value v of the Higgs field. The exact relations between the initial gauge bosons and the EW bosons are the following

$$W_\mu^\pm = \frac{1}{\sqrt{2}} (W_\mu^1 \mp iW_\mu^2) \quad m_W = \frac{1}{2} vg$$

$$Z_\mu = \cos \theta_W W_\mu^3 - \sin \theta_W B_\mu \quad m_Z = \frac{1}{2} v \sqrt{g^2 + g'^2}$$

$$A_\mu = \sin \theta_W W_\mu^3 + \cos \theta_W B_\mu \quad m_A = 0$$

where θ_W is the Weinberg angle, related to the coupling constants by the following relations

$$\sin \theta_W = \frac{g}{\sqrt{g^2 + g'^2}} \quad \cos \theta_W = \frac{g'}{\sqrt{g^2 + g'^2}}$$

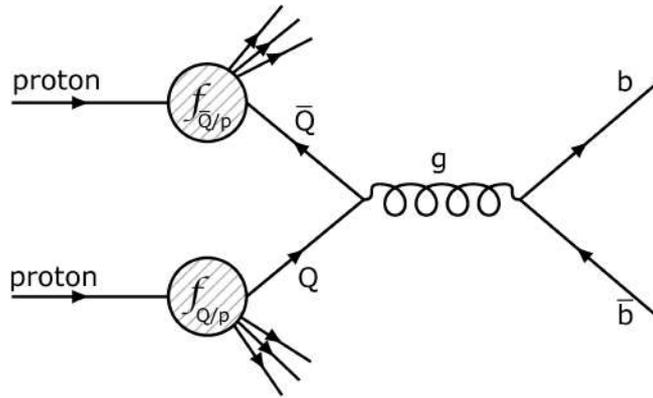


Figure 1.2: Factorization diagram of a $pp \rightarrow b\bar{b}$ event. The soft QCD contribution to the initial state can be factored out from the hard scattering process, making use of the PDFs of the proton $f_{Q/p}$ and $f_{\bar{Q}/p}$.

1.2 b -quark production in proton-proton collisions

At LHC, quark pairs are produced in proton-proton (pp) collisions at the center of mass energy of $\sqrt{s} \approx 13$ TeV. Being the proton a non-elementary particle, the reconstruction of the kinematic of a pp collision requires knowing the composition in terms of position and momentum of the quark and gluon content of the proton: this information is encoded in the Parton Distribution Functions (PDFs). A PDF $f_{a/A}(x, \mu)$ represents the probability distribution of finding the parton a inside the hadron A , carrying the fraction x of the total momentum of A , The parameter μ has usually the dimension of a mass and is related to the renormalization scale and the running of α_s . The PDFs are evaluated experimentally fitting observables to data.

The PDFs allow to express conveniently cross sections involving hadronic initial states, factoring out the non-perturbative from the perturbative one. This result is expressed in the QCD Factorization Theorem [27] that, in the case of quark pair production, can be stated as follows

$$d\sigma_{pp \rightarrow q\bar{q}} = \sum_{n,m} \int \int dx_n dx_m f_{n/p}(x_n, \mu) f_{m/p}(x_m, \mu) d\hat{\sigma}_{nm \rightarrow q\bar{q}} \quad (1.1)$$

where the sum over n and m is extended over all the possible partons that have a non-vanishing contribution to $d\hat{\sigma}_{nm \rightarrow q\bar{q}}$ at the parton level, at the chosen order of perturbation theory. Focusing on the $b\bar{b}$ production in pp collisions, Figure 1.2 shows diagrammatically the contribution given by the quark-fusion process $Q\bar{Q} \rightarrow g \rightarrow b\bar{b}$, weighted by the two PDFs $f_{Q/p}$ and $f_{\bar{Q}/p}$. At Leading Order (LO), the processes contributing to the b production are the following:

- *QCD quark-fusion*: a quark and an anti-quark in the initial state annihilate into a gluon which produces a $b\bar{b}$ pair.
- *QCD gluon-fusion*: two gluons in the initial state interact by mean of a third gluon (in the s -channel) or a quark (in the t -channel), producing a $b\bar{b}$ pair.
- *EW quark-antiquark annihilation*: a quark-antiquark pair in the initial state annihilates into a Z^0/γ which produces a $b\bar{b}$ pair.

The Feynman diagrams of these processes are represented in Figure 1.3.

1.3 Hadronization of b -quarks

The principle of confinement states that, although quarks and gluons, commonly referred to as partons, are the fundamental particles of the QCD theory, they can never be observed freely. The partonic structure of hadrons has been a deeply studied field of particle physics for decades, however, the process by which hadrons are formed from scattered partons is less understood than their initial-state partonic structure counterparts. This process is often referred as *hadronization* or *fragmentation*.

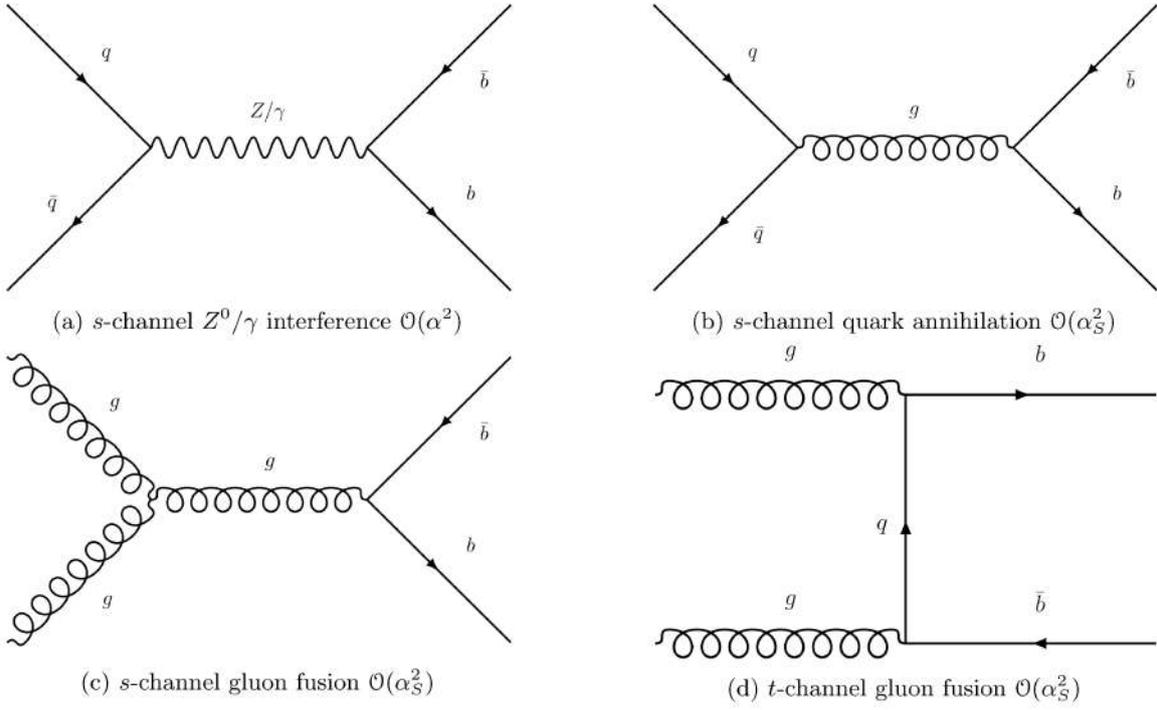


Figure 1.3: LO Feynman diagrams contributing to the $pp \rightarrow b\bar{b}$ process.

Generally, hadronization is described in terms of the *Fragmentation Functions* (FFs), which represent the probability density distribution for the fraction of the b -quark energy taken by the b -hadron. The b -quark fragmentation function is defined as

$$\mathcal{D}(z) = \frac{1}{\sigma_{\text{tot}}} \frac{d\sigma}{dz}$$

where σ_{tot} is the total hadronic cross section, while z relates the energy of the b -quark to the energy of the b -hadron and is often defined as

$$z = \frac{(E + p_L)_B}{(E + p)_b}$$

where p_L is the momentum of the b -hadron along the direction of the b -quark. This variable is invariant under Lorentz boosts along the quark direction and is commonly used in Monte Carlo generators. The QCD Factorization Theorem [27] allows to fold the perturbative and non-perturbative contribution to the FF, so that the observed fragmentation function can be expressed as

$$\mathcal{D}_{\text{observed}}(z) = \int_0^{\infty} \mathcal{D}_{\text{pert.}}(t) \times \mathcal{D}_{\text{non-pert.}}\left(\frac{t}{z}\right) \frac{dz}{z}$$

A consequence of the factorization theorem is that the non-perturbative component does not depend on the initial partonic state.

Monte Carlo generators, like PHYTIA8, deal with the perturbative part with the Parton Shower model: in this model partons are created by successive elementary branching processes $q \rightarrow qg$, $g \rightarrow gg$ and $g \rightarrow q\bar{q}$ (neglecting the interference between the different channels).

The non-perturbative part is dealt with phenomenological schemes that are used to model the carry-over of the parton momentum and flavor to the hadrons. Within the context of this work, PHYTIA8 is configured to use the Peterson model, a simple phenomenological parametrization of the spectra of heavy hadrons that was presented in 1983 by *Peterson et al* [59]. The basic idea behind the model is that when a light quark \bar{q} combines with an heavy quark Q to form a hadron $H = Q\bar{q}$, the heavy quark slightly decelerates: therefore, Q and $Q\bar{q}$ should carry almost the same energy. As a consequence, the main contribution to the amplitude \mathcal{M} of the transition $Q(E_Q, p_Q) \rightarrow H(E_H, p_H) + q(E_q, p_q)$ is supposed to be

$$\mathcal{M} \propto \frac{1}{\Delta E} = \frac{1}{E_H + E_q - E_Q}$$

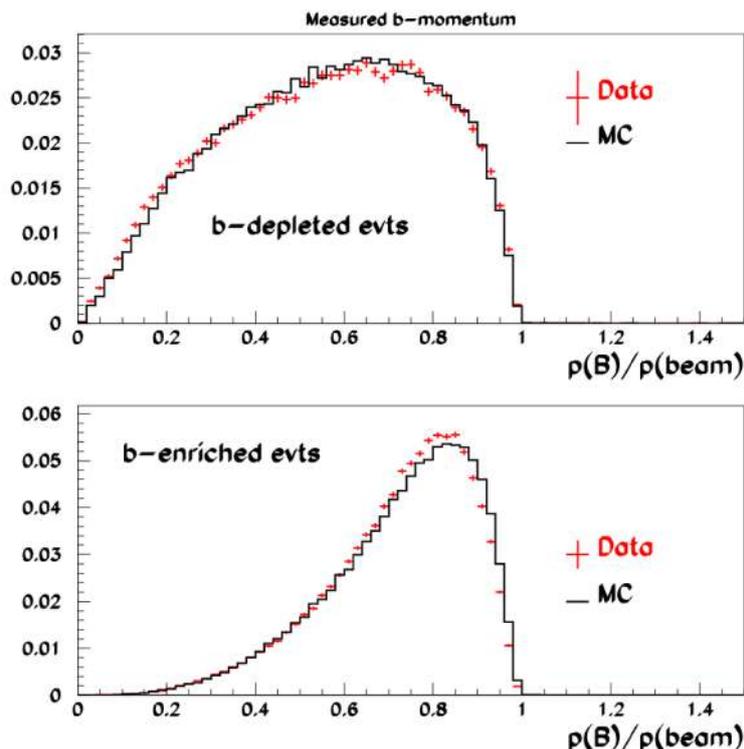


Figure 1.4: b -quark fragmentation functions computed at DELPHI from hadronic Z^0 decays compared with Monte Carlo expectations. On the x -axis is reported the ratio between the b -hadron momentum and the momentum of the beam $p(B)/p(\text{beam})$. Top: fragmentation function on a b -depleted sample. Bottom: fragmentation function on a b -enriched sample. [\[11\]](#)

the energy difference between the initial and the final state ΔE is

$$\Delta E = \sqrt{m_H^2 + z^2 p^2} + \sqrt{m_q^2 + (1-z)^2 p^2} - \sqrt{m_Q^2 + p^2}$$

where p is the three-momentum of the heavy quark before the hadronization, m_H , m_Q and m_q are the masses of H , Q and q respectively. In the $p/m_Q \gg 1$ limit (well justified in high energy colliders), this expression can be approximated by

$$\Delta E \approx \frac{m_Q^2}{2p} \left(\frac{1}{z} + \frac{m_q^2/m_Q^2}{1-z} - 1 \right) \propto 1 - \frac{1}{z} - \frac{\epsilon}{1-z}$$

where $\epsilon = m_q^2/m_Q^2$. Therefore, the Peterson model proposes the following parametrization of the non-perturbative FF $\mathcal{D}(z) \propto \mathcal{M}^2$

$$\mathcal{D}_{\text{non-pert.}}(z) = N(\epsilon) \frac{1}{z \left(1 - \frac{1}{z} - \frac{\epsilon}{1-z} \right)^2}$$

where $N(\epsilon)$ is a normalization factor.

Due to its large mass, the b -quark tends to keep a large fraction of its energy, which is finally transferred to the b -hadron: the bottom plot of Figure [1.4](#) shows that the FF evaluated on a b -enriched sample of events, shows a peak of the ratio $p(B)/p(\text{beam})$ close to one.

1.4 b -quark pair-production asymmetry

The study of the production of b -quarks at high-energy colliders has been a profitable field of research: in particular, precision measurements of both production rates and angular distributions of

heavy quarks at leptons colliders at the Z^0 resonance have allowed to put important constraints on EW coupling in the heavy flavor sector of the SM. Analogously, hadron colliders allow for differential measurements of the same observables in extended range of invariant mass of the produced quarks system, giving the possibility to search for new physics also in the strong coupling of heavy flavors.

A brief overview of the measurement of $b\bar{b}$ pair-production asymmetries at high-energy colliders is now provided.

1.4.1 $b\bar{b}$ asymmetry measurement at e^+e^- colliders

In the SM, the Z^0 boson couples to both left- and right-handed chiral states with different strengths depending on the weak-isospin and the electromagnetic charge. The vector and axial-vector Z^0 couplings for a fermion f are

$$v^f = I_3^f - 2Q^f \sin^2 \theta_W \qquad a^f = I_3^f$$

where I_3^f is the third component of the weak isospin of the fermion f and Q^f its charge, while θ_W is the Weinberg angle. The mixed Z^0 vector and axial-couplings induce asymmetries in the angular distributions of fermions produced in $e^+e^- \rightarrow f\bar{f}$ processes. In the case of b -quarks, the Born level differential cross-section of the process $e^+e^- \rightarrow Z/\gamma \rightarrow b\bar{b}$, summed over the final-state helicities, assuming an unpolarized positron beam but allowing a polarization \mathcal{P}_e of the electron beam, taken positive for right-handed beam helicity and negative for left, is

$$\frac{d\sigma_{b\bar{b}}}{d\cos\theta} = \frac{3}{8} \sigma_{b\bar{b}}^{tot} [(1 - \mathcal{P}_e \mathcal{A}_e)(1 + \cos^2\theta) + 2(\mathcal{A}_e - \mathcal{P}_e)\mathcal{A}_b] \quad (1.2)$$

where $\sigma_{b\bar{b}}^{tot}$ is the total $b\bar{b}$ production cross-section, while \mathcal{A}_b and \mathcal{A}_f are called asymmetry parameters and incorporate the dependence on the fermion couplings

$$\mathcal{A}_f = 2 \frac{v^f/a^f}{1 + (v^f/a^f)^2}$$

Designating the integrals of 1.2 over the forward and backward hemispheres with subscripts F and B, and cross-section for right and left electron helicities with subscripts R and L, two basic asymmetries can be measured

$$A_{FB}^{b\bar{b}} = \frac{\sigma_F - \sigma_B}{\sigma_F + \sigma_B} \propto \mathcal{A}_e \mathcal{A}_b \quad (\text{Forward-Backward asymmetry}) \quad (1.3)$$

$$A_{FBLR}^{b\bar{b}} = \frac{1}{|\mathcal{P}_e|} \frac{(\sigma_F - \sigma_B)_L - (\sigma_F - \sigma_B)_R}{(\sigma_F - \sigma_B)_L + (\sigma_F - \sigma_B)_R} \propto \mathcal{A}_b \quad (\text{Forward-Backward Left-Right asymmetry}) \quad (1.4)$$

The SLD experiment at SLAC has used polarized e^+e^- beams at the Z^0 resonance to measure forward-backward-left-right asymmetry $A_{FBLR}^{b\bar{b}}$ as a function of the polar angle θ formed by the b -quark with respect to the beam-axis

$$A_{FBLR}^{b\bar{b}}(\cos\theta) = |\mathcal{P}_e| \mathcal{A}_b \frac{2\cos\theta}{1 + \cos^2\theta}$$

A fit of experimental data allows the extraction of the asymmetry parameter \mathcal{A}_b

$$\mathcal{A}_b \text{ (SLD)} = 0.923 \pm 0.020$$

This result is consistent with SM predictions obtained by the EW fit [60].

On the other hand, the ALEPH, DELPHI, OPAL and L3 experiments at LEP used unpolarized e^+e^- beams at the Z^0 resonance to measure the forward-backward asymmetry $A_{FB}^{b\bar{b}}$ as a function of the polar angle θ

$$A_{FB}^{b\bar{b}}(\cos\theta) = \mathcal{A}_e \mathcal{A}_b \frac{2\cos\theta}{1 + \cos^2\theta}$$

The combined result of the four experiments is the following

$$A_{FB}^{b\bar{b}}(\text{LEP}) = 0.0992 \pm 0.0016$$

This value shows a large discrepancy of 2.8σ with respect to the theoretical prediction, obtained from the EW fit $A_{FB}^{b\bar{b}}|^{th} = 0.1037 \pm 0.0008$ [60].

1.4.2 $b\bar{b}$ asymmetry measurement at $p\bar{p}$ colliders

The production of b -quark pairs at hadron colliders is much more complex compared to e^+e^- colliders. The two main SM strong pair-production processes are $q\bar{q} \rightarrow b\bar{b}$ and $gg \rightarrow b\bar{b}$, neither of which induce an asymmetry at LO. However, when higher-order corrections are included, several sources of asymmetry appear. Near the Z^0 -pole, the main contribution to the asymmetry is given by Z^0 -bosons decaying to a b -quark pair $Z^0 \rightarrow b\bar{b}$. The production of a Z^0 -boson in $p\bar{p}$ collisions can occur at LO via the annihilation of a valence quark of the proton with a valence anti-quark of the anti-proton: the direction of the quark and of the anti-quark pair in the initial state are always known and coincide with the proton direction and the anti-proton direction respectively. Therefore, assuming ‘‘forward’’ the direction of the proton and ‘‘backward’’ the direction of the anti-proton, $p\bar{p}$ colliders are able to measure the forward-backward asymmetry of the $b\bar{b}$ pair-production, counting the number of event where the quantity Δy is positive and negative

$$A_{FB}^{b\bar{b}} = \frac{N(\Delta y > 0) - N(\Delta y < 0)}{N(\Delta y > 0) + N(\Delta y < 0)}$$

where $\Delta y = y_b - y_{\bar{b}}$ is the difference between the rapidity of the two b -quarks. The rapidity of a particle with four-momentum (E, \mathbf{p}) is defined as

$$y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$

where E is the energy of the particle, while p_z is the component of three-momentum along the beam direction, assuming positive the versus of the proton beam. In the ultra-relativistic limit where the quark 3-momentum \mathbf{p} is much larger than its mass m_b , the rapidity can be related to the polar angle between the particle and the beam axis θ , through the pseudo-rapidity η

$$y \stackrel{|\mathbf{p}| \gg m_b}{\approx} -\ln \tan \frac{\theta}{2} = \eta$$

The CDF experiment at Tevatron has performed a measurement of the $A_{FB}^{b\bar{b}}$ produced at $p\bar{p}$ collisions at the center of mass energy of $\sqrt{s} = 1.96$ TeV, in a low invariant mass region (including the Z^0 -peak) [20] and in a high invariant mass region [4], with the following result

$$A_{FB}^b(\text{CDF low mass}) = \begin{cases} 0.83 \pm 0.83\%, & 40 < m_{b\bar{b}} < 75 \text{ GeV}/c^2 \\ 1.54 \pm 0.73\%, & 75 < m_{b\bar{b}} < 95 \text{ GeV}/c^2 \\ 0.92 \pm 0.87\%, & 95 < m_{b\bar{b}} < 130 \text{ GeV}/c^2 \\ 2.08 \pm 1.10\%, & m_{b\bar{b}} > 130 \text{ GeV}/c^2 \end{cases}$$

$$A_{FB}^b(\text{CDF high mass}) = \begin{cases} -6.6_{-9.7}^{+9.4}\%, & 150 < m_{b\bar{b}} < 225 \text{ GeV}/c^2 \\ -7.4_{-8.9}^{+8.7}\%, & 225 < m_{b\bar{b}} < 325 \text{ GeV}/c^2 \\ -6.1_{-11.7}^{+15.3}\%, & 325 < m_{b\bar{b}} < 1960 \text{ GeV}/c^2 \end{cases}$$

A similar measurement has been performed at the DØ experiment, measuring the inclusive production asymmetry of the B^\pm mesons (which is sensitive to A_{FB}^b) [5], with the following result

$$A_{FB}^{B^\pm}(\text{DØ}) = -0.24 \pm 0.41 (\text{stat}) \pm 0.17 (\text{syst})\%$$

The computation of SM theoretical prediction is far more complex than the e^+e^- case and involves the inclusion of NLO contributions, especially far from the Z^0 -pole.

The comparison between the experimental results and the theoretical predictions provided by *C.W. Murphy* [54] has shown that while CDF measurements are compatible with SM predictions, DØ estimation shows a $\approx 3\sigma$ discrepancy.

1.4.3 $b\bar{b}$ asymmetry measurement at pp colliders

The production of $b\bar{b}$ -quark pairs at pp colliders is similar to the $p\bar{p}$ case. However, in this case, the Z -boson production via quark anti-quark annihilation need for an anti-quark to be extracted from the quark sea and since the initial state of the collision is symmetric, it is not possible to know which of the two proton is providing the valence quark and which is extracting the sea anti-quark. As a consequence, “forward” and “backward” directions cannot be unambiguously defined and therefore A_{FB}^b cannot be measured. Anyway, the forward-central $b\bar{b}$ pair-production asymmetry, which probes angular asymmetries between the central and the forward regions, can be measured counting the number of events where the quantity $\Delta|y|$ is positive and negative

$$A_{\text{FC}}^{b\bar{b}} = \frac{N(\Delta|y| > 0) - N(\Delta|y| < 0)}{N(\Delta|y| > 0) + N(\Delta|y| < 0)}$$

where $\Delta|y| = |y_b| - |y_{\bar{b}}|$ is the difference between the absolute value of the rapidity of the two b -quarks.

The LHCb experiment at LHC has performed a measurement of $A_{\text{FC}}^{b\bar{b}}$ in pp collisions, using 1 fb^{-1} acquired at the center of mass energy of $\sqrt{s} = 7 \text{ TeV}$, in three invariant mass bins, producing the following results [2]

$$A_{\text{FC}}^{b\bar{b}} (\text{LHCb}) = \begin{cases} 0.4 \pm 0.4 \text{ (stat)} \pm 0.3 \text{ (syst)}\%, & 40 < m_{b\bar{b}} < 75 \text{ GeV}/c^2 \\ 2.0 \pm 0.9 \text{ (stat)} \pm 0.6 \text{ (syst)}\%, & 75 < m_{b\bar{b}} < 105 \text{ GeV}/c^2 \\ 1.6 \pm 1.7 \text{ (stat)} \pm 0.6 \text{ (syst)}\%, & m_{b\bar{b}} > 105 \text{ GeV}/c^2 \end{cases}$$

Within the experimental uncertainties, results are compatible with SM predictions[54], in particular the asymmetry in the high invariant-mass region is also consistent with 0.

1.4.4 New Physics in b -quark asymmetry

The study of b -quark production asymmetries at lepton and hadron colliders is, up to now, a great channel for the search of New Physics in the heavy flavor sector of the SM. Deviations from SM predictions, observed at LEP and $D\bar{D}$ are yet to be fully explained: it is not clear if they are caused by statistical fluctuations and/or by inaccurate theoretical predictions of the observables or, instead, if new processes beyond the SM are at play.

Several models extending the SM have been proposed to solve the discrepancies:

- A new scalar boson interfering at tree level with the EW bosons exchange (Figure 1.3a) and the gluon exchange (Figure 1.3b) [14].
- A composite Higgs model, where the EW gauge group is broken by new strong interactions [9].
- Light axigluon as a mediator of a chiral color theory that splits the $SU(3)$ group of QCD into left and right part $SU(3)_L \otimes SU(3)_R$ [45].

More accurate measurements of the b -quark pair-production asymmetry are required to properly disentangle possible new physics processes from statistical fluctuations: in particular, a differential measurement in finer bins of invariant mass of the two quark system could help to get insights near the Z^0 -peak region, where the LO EW contribution to the asymmetry is dominant, as well as at high invariant masses where QCD NLO contributions represent the only source of asymmetry predicted by the SM.

Chapter 2

The LHCb experiment and jet identification

2.1 The Large Hadron Collider

The Large Hadron Collider [33] consists of a circular accelerator around 27 km long, located 100 m underground at the CERN.

Inside LHC, two counter-rotating proton or heavy-ion beams circulate into two separate beam pipes where hard-vacuum has been obtained: the two pipes intersect in four interaction points where detectors are located. Protons are produced ionizing hydrogen atoms and go through several pre-acceleration stages before entering LHC (see Figure 2.1): the first acceleration stage, up to 50 MeV, is provided by the Linear accelerator (Linac). Then they get successively accelerated by the Proton Synchrotron Booster (PSB), up to 1.4 GeV, by the Proton Synchrotron (PS), up to 28 GeV, and by the Super Proton Synchrotron (SPS), up to 450 GeV. Finally, they get injected into the LHC where each of the two beam can reach the final energy of ≈ 7 TeV, consequently reaching a center of mass energy at the collision of $\sqrt{s} \approx 14$ TeV. Protons are accelerated by mean of 16 radio-frequency (RF) cavities located inside 4 cylindrical refrigerators, working in superconducting regime.

The beams are not continuous: they are made of bunches of $\approx 10^{11}$ protons, around 30 cm long and with a transverse dimension of the order of 1 mm which gets squeezed to around $16 \mu\text{m}$ at the collision points. In the nominal configuration, each beam contains 2808 bunches, spaced by around 25 ns each other, reaching a collision frequency of 40 MHz and a peak luminosity of $\mathcal{L} = 2 \times 10^{34} \text{ cm}^2 \text{ s}^{-1}$.

The four main experiments are located at the four collision points of LHC: ATLAS (A Toroidal LHC Apparatus)[23] and CMS (Compact Muon Solenoid)[24] are the two General Purpose Detectors, designed with a cylindrical structure around the collision point. ALICE (A Lead Ions Collision Experiment)[22] is dedicated to the study of phases of matter where quark and gluons are free, in heavy ions collisions. Finally, LHCb, which is devoted to the study of b - and c -quarks properties in the forward region of the collision.

2.2 The LHCb detector

LHCb[25] is a single-arm spectrometer with a forward angular coverage from approximately 10 mrad to 300 (250) mrad in the bending (non-bending) plane: this corresponds to a pseudo-rapidity¹ coverage between 1.8 and 4.9. The LHCb detector is composed of several sub-detectors. Its layout is presented in Figure 2.2. The standard LHCb coordinates system is the one with the z -axis parallel to the beam direction and the y -axis parallel and opposite to the gravitational acceleration; the x -axis is orthogonal to the yz -plane, forming a right-handed system. The main sub-detectors are now listed, starting from the interaction point:

¹The pseudo-rapidity is defined as $\eta = -\ln \tan \frac{\theta}{2}$, where θ is the polar angle formed by the particle momentum and the beam axis.

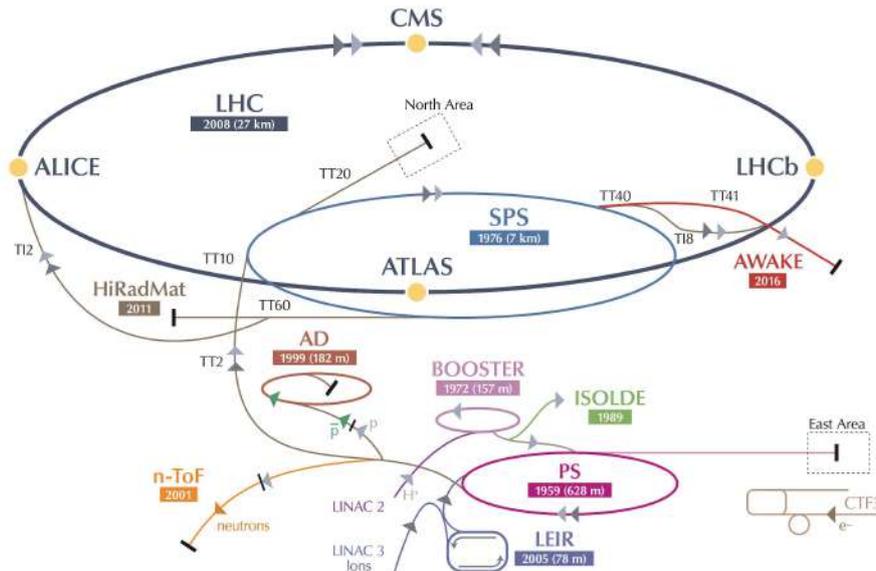


Figure 2.1: Schematic representation of accelerator facilities at CERN

- a Vertex LOcator (VELO) system located close to the interaction point;
- a system of four tracking stations: one called Trigger Tracker (TT) is located upstream of the magnet, while the other three (T1, T2 and T3) are located downstream of the magnet;
- two Ring Imaging Cherenkov (RICH) detectors located one upstream and one downstream of the magnet, that are used for particle identification, in particular to achieve a good pions versus kaons separation;
- a calorimeter system composed of a Scintillator Pad Detector and Preshower (SPD/PS), an electromagnetic calorimeter (ECAL) and an hadronic calorimeter (HCAL);
- finally, a muon detection system in the outer part of the detector.

The optimal LHCb luminosity is such that a maximum of 2.5 proton-proton interactions per bunch crossing can occur. In order to achieve this requirement the luminosity provided by LHC is reduced using a *luminosity levelling* technique [36]: the two beams do not collide head-on but the beam axes are tilted obtaining a larger interaction area; this allows to reach a constant luminosity level of around $4.5 \times 10^{32} \text{ cm}^{-2} \text{ s}^{-1}$. In Table 2.1 the integrated luminosity for each year of the Run II campaign is shown, with Center of Mass (CoM) energy of 13 TeV.

year	$\mathcal{L}_{int} [\text{fb}^{-1}]$
2015	0.33
2016	1.67
2017	1.81
2018	2.19
Total	6.00

Table 2.1: Integrated recorded luminosity at LHCb for each year of the Run II campaign at the CoM energy of 13 TeV. [25]

2.2.1 The tracking system

The LHCb detector includes a high-precision tracking system made of sub-detectors exploiting different technologies. It consists of a silicon-strip VELO surrounding the pp interaction region, provides precise measurements of tracks coordinates close to the interaction region; and four tracking stations: the TT,

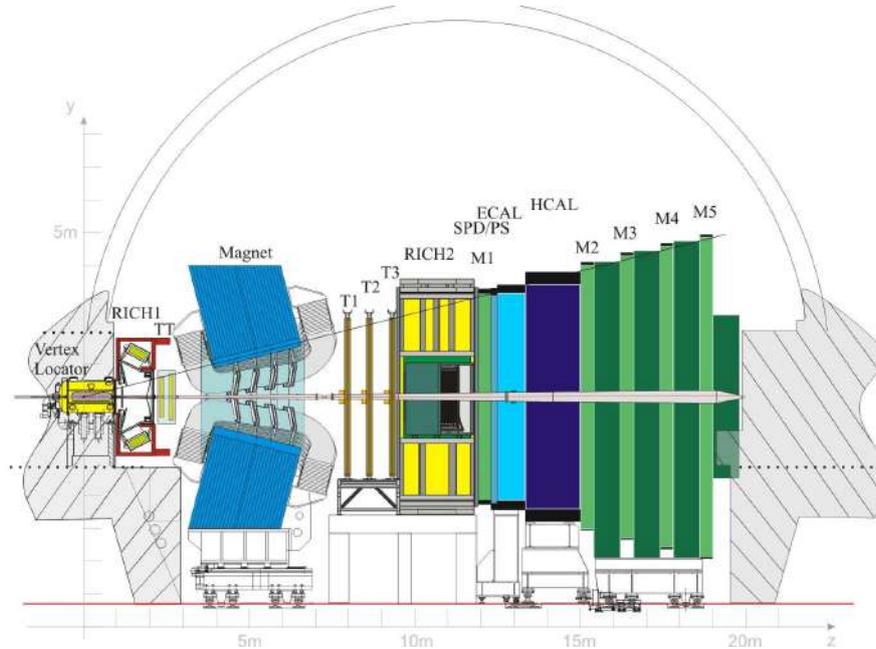


Figure 2.2: Schematic representation of the LHCb spectrometer, showing each sub-system. The z -axis is parallel to the beam direction while the y -axis is parallel to the gravitational acceleration. [25]

which is a large-area silicon-strip detector, located upstream of a dipole magnet; and three tracking stations (T1-T3), which are placed downstream of the magnet. In T1-T3, silicon micro-strips are used in the region close to the beam pipe (called Inner Tracker, IT) while straw-tubes are employed in the outer region of the stations (Outer Tracker, OT). The tracking sub-systems are now briefly described.

Vertex Locator

The VELO[1] is the sub-system devoted to the measurement of tracks coordinates near the interaction point. This is crucial to identify secondary vertices associated to b - and c - hadrons decays. The VELO consists of a series of silicon modules, each providing a measure of the r (from 8 mm to 42 mm) and ϕ (from 15 mrad to 390 mrad) coordinates. The r and ϕ sensors only differ for the geometry of the strips segmentation, while having both a diameter of 84 mm and a thickness of 300 μm . The $r\phi$ geometry is shown in Figure 2.3.

The VELO system consists of 21 stations of these semi-circular sensors, placed along a distance of 1 m, parallel to the beam axis, as shown in Figure 2.3. The sensors are mounted in a vessel that maintains the vacuum and are separated from the machine vacuum by a thin aluminum sheet called called RF-foils. This is done to minimize the material traversed by a charged particle before it crosses the sensors, while providing a good geometrical coverage. The sensors operates at a distance of 8 mm from the beam axis which is smaller than the beam width during the injection phase: for this reason the sensors are retractable and the VELO stays in the open configuration until the beam conditions are stable. The individual hit resolution of the sensors have been measured during the test beams and the best raw resolution obtained is around 7 μm .

Tracking stations

The tracking system is composed of four planar stations perpendicular to the beam axis: the TT is 150 cm wide and 130 cm high and it is placed upstream of the dipole magnet, covering the full geometrical acceptance of the experiment; the T1-3 stations are placed downstream the magnet: the inner part of the T1-3 station is 120 cm wide and 30 cm high, and together with the TT forms the Inner Tracker (IT), shown in purple in Figure 2.4. The IT is composed by 200 μm wide silicon micro-strip sensors, displaced in four layers of three different type:

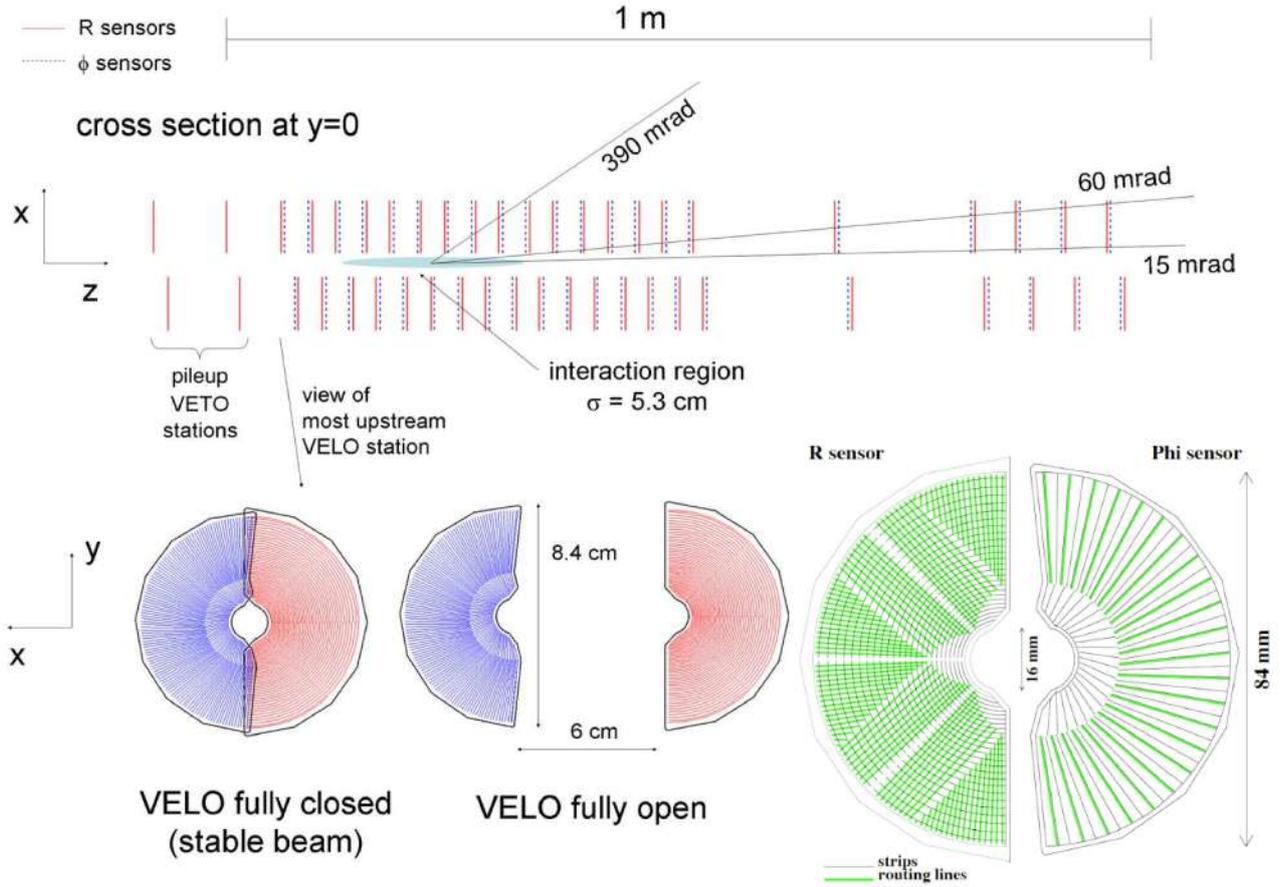


Figure 2.3: Schematic layout of the VELO system. [1, 25]

- x -layers have perfectly vertical strips, aligned with the y -axis;
- u -layers have strips tilted of -5° ;
- v -layers have strips tilted of $+5^\circ$

Figure 2.5 shows the layouts of a v -layer of the TT station (on the left) and an x -layer of an IT station (on the right). The different layers are arranged with the pattern $x-u-v-x$. The external regions of the T1-3 stations are called Outer Tracker (OT), shown in light blue in Figure 2.4; it is a drift tube detector with a geometrical acceptance of 300 mrad on the horizontal plane and 250 mrad on the vertical plane. It is made of an array of straw-tube modules, each one formed by two layers of drift-tubes with an inner diameter of 4.9 mm, containing a mixture of Ar (70%), CO_2 (28.5%) and O_2 (1.5%). The final drift distance resolution is $\approx 200 \mu\text{m}$.

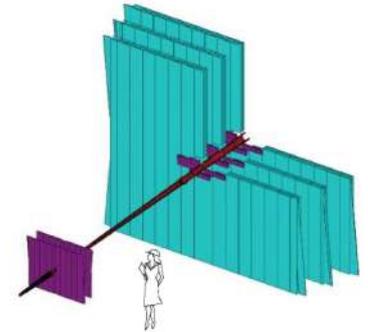


Figure 2.4: Illustration of the tracking system, showing the IT (violet) and the OT (cyan). [25]

2.2.2 Tracking performance

The tracking stations TT and T1-3, together with the VELO, are used to measure the trajectory of charged particles allowing the determination of the momentum from the curvature induced by the magnet. Pattern recognition algorithms are used to reconstruct the track trajectory starting from the hits coordinates in all the tracking sub-detectors. The tracks are reconstructed via a Kalman fit [37] and are classified in four categories:

- **long tracks**, showing hits in the VELO and in all the T stations;

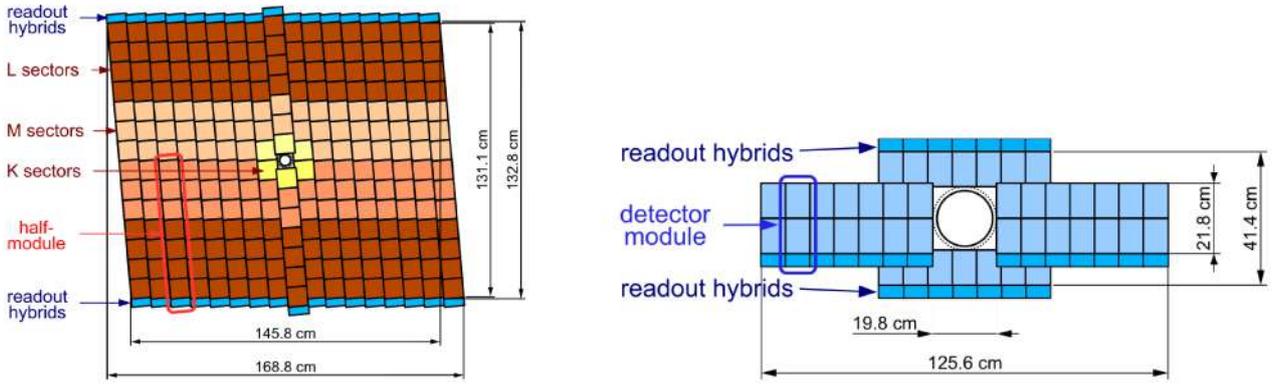


Figure 2.5: Left: layout of a v -layer of the TT station. Right: layout of x -layer of a IT station. 25

- **downstream tracks**, which have hits only in the VELO and the TT station;
- **upstream tracks**, which have hits in all the T stations but not in the VELO;
- **VELO tracks**, which have hits only in the VELO.

The momentum resolution $\delta p/p$ of long tracks has been measured using data samples of the $J/\psi \rightarrow \mu^+ \mu^-$ decay, collected with the *Dimuon* trigger line, selecting couples of high energy muons. Its behavior as a function of the momentum of the two muon final state is shown in Figure 2.6a. The invariant mass resolution σ_m/m has been determined observing six resonances in the dimuon data sample: J/ψ , $\psi(2S)$, $\Upsilon(1S)$, $\Upsilon(2S)$, $\Upsilon(3S)$ mesons and the Z^0 -boson. Its behavior as a function of the invariant mass of the dimuon resonance is shown in Figure 2.6b.

The Primary Vertex (PV), namely, the pp interaction point where particles are generated, is reconstructed from the detected tracks and it involves the minimization of the quantity

$$\chi_{PV}^2 = \sum_{i=1}^{n_{tracks}} \frac{d_{0i}^2}{\sigma^2 d_{0i}}$$

where d_0 is the impact parameter of the track and σd_0 is its uncertainty 48. The PV reconstruction efficiency and resolution have been evaluated in simulations 48 and they show a dependence from the tracks multiplicity. The average efficiency goes from 90.0% to 97.5%, with a probability of reconstructing a false PV of around 1%. The resolutions on the x , y and z coordinate are respectively $8 \mu\text{m}$, $10 \mu\text{m}$ and $50 \mu\text{m}$.

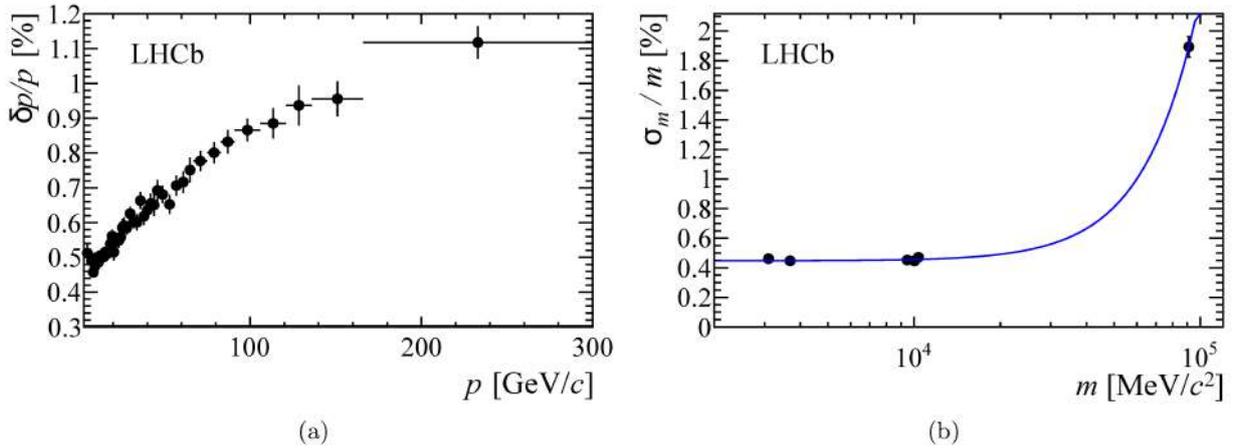


Figure 2.6: (a) The relative momentum resolution as a function of the momentum for long tracks, obtained from J/ψ decays. (b) The relative mass resolution as a function of the mass of the dimuon resonance. [49](#)

2.2.3 Magnet

Between the TT and the T1-3 stations of the tracking system, a dipole magnet is located, allowing the measurement of the momentum of charged particles, by measuring the trajectory curvature induced by the magnetic field. The magnet covers a forward acceptance of ± 250 mrad vertically and ± 300 mrad horizontally. The magnetic field is parallel to the y -axis, therefore the trajectories are bent on the xz -plane. The magnet is not super-conductive and the current flowing inside the coils can be reverted, allowing measurements with two opposite field polarities. The integrated magnetic field for 10 m long tracks is ≈ 4 Tm, while the residual magnetic field inside the two RICHs is ≈ 2 Tm.

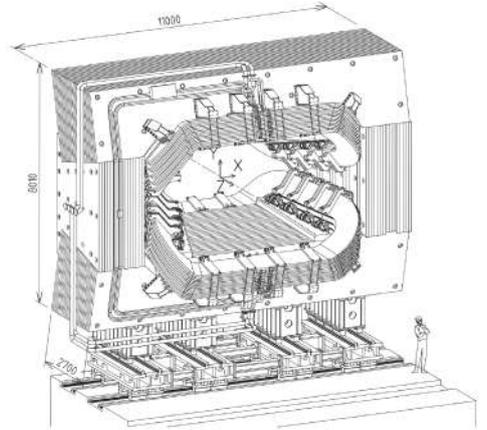


Figure 2.7: Illustration of the LHCb magnet (lengths are expressed in mm). [25](#)

2.2.4 Ring Imaging Cherenkov detectors

LHCb features great particle identification (PID) performance thanks to the two Ring Imaging Cherenkov detectors, RICH1 and RICH2. Their primary role is to identify charged hadrons (π , K , p); additionally the RICH system can help in the identification of charged leptons (e , μ), complementing the information coming from the downstream calorimeters and muon system.

Particles crossing the RICH radiator travel at a velocity larger than the speed of light in the same medium. As a consequence, the charged ones emit Cherenkov light in a cone with axis parallel to the particle speed. The cone angle θ_c is related to the particle velocity v by the relation

$$\cos \theta_c = \frac{c/n}{v} = \frac{c_m}{v} = \frac{\sqrt{m^2 c^2 + p^2}}{np}$$

where n is the refractive index of the RICH active medium, m and p are the mass and the momentum of the particle, while c and c_m are, respectively, the speed of light in the vacuum and in the active medium. By measuring the angle of the Cherenkov cone of a charged particle and knowing p from the tracking system, one can get an estimation of m

$$m = \frac{\sqrt{n^2 p^2 \cos^2 \theta_c - p^2}}{c}$$

therefore allowing the identification of the particle by its mass.

The RICH1, located upstream of the magnet, between the VELO and the TT station, allows to distin-

guish particles with a momentum range between 1 and 60 GeV/ c and its active media are silica aerogel and C₄F₁₀. It covers an angular acceptance range from ± 25 mrad up to ± 300 mrad horizontally and up to ± 250 mrad vertically.

The RICH2 is located downstream of the magnet, between the T3 station and the first muon station M1. It detects particles with momentum between 15 and 100 GeV/ c . Its active medium is CF₄ and its angular acceptance range goes from ± 15 mrad up to ± 120 mrad horizontally and up to ± 100 mrad vertically.

2.2.5 Calorimeters

The calorimeter system has the following roles:

- Selects hadrons, electrons and photons with significant transverse momentum;
- Provides PID for electrons, photons and hadrons;
- Measures the energy of neutral particles such as photons, π^0 as other neutral hadrons.

Located downstream of the RICH2 detector, the LHCb calorimeter is composed by a Scintillating Pad Detector (SPD), a Pre-Shower detector (PS), an Electromagnetic Calorimeter (ECAL) and an Hadronic Calorimeter (HCAL). Their structure and granularity is represented in Figure 2.8.

The SPD/PS system allows for the identification of electrons and photons: it is made by two almost identical scintillator, coupled to Photo Multipliers (PMT) via Wavelength-Shifting (WLS) fibers, with a 15 mm lead converter in between, covering an angular acceptance range from ± 30 mrad to ± 300 mrad horizontally and ± 250 mrad vertically. Particle identification can be done since in the SPD photons do not produce any signal before triggering a shower, while the electrons do. On the other hand, photons passing the lead layer of the SPD produce showers on the PS. This allow for e/γ separation. Hadrons do not trigger showers on the PS, therefore they can be distinguished from electrons.

In the ECAL, electrons and photons deposit the major part of their energy. ECAL has the same geometrical acceptance as the SPD/PS system and it is composed by 66 layers of alternating 4 mm lead tiles and scintillators coupled to PMT via WLS fibers, resulting in a total length of 42 cm (corresponding to 25 radiation lengths). The granularity of the ECAL allows for the achievement of a good energy resolution of

$$\left(\frac{\sigma_E}{E}\right)_{\text{ECAL}} = \frac{10\%}{\sqrt{E}} \oplus 1\%$$

Where E is the energy of the particle (in GeV) and \oplus indicates the “squared sum”.

Finally, the HCAL collects the remaining energy of charged and neutral hadrons. Its structure is very similar to ECAL: 1 cm iron tiles and scintillators alternates for a total of 5.6 radiation lengths. HCAL is able to reach an energy resolution of

$$\left(\frac{\sigma_E}{E}\right)_{\text{HCAL}} = \frac{69\%}{\sqrt{E}} \oplus 9\%$$

again E is the energy of the hadron (in GeV).

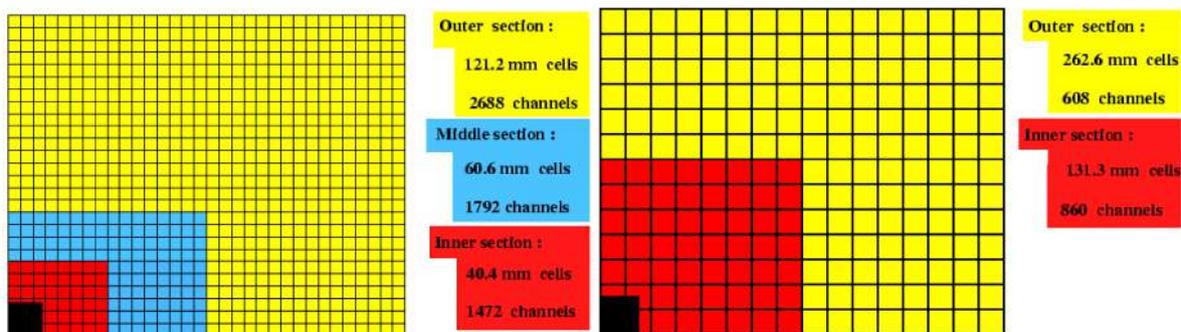


Figure 2.8: Structure and granularity of the SPD/PS and ECAL systems (left), and of HCAL (right). [25](#)

2.2.6 Muon System

The muon system is the outermost detector from the interaction point and it allows for fast muon detection. It is composed of five stations M1-5 of rectangular shape, placed along the beam axis, as shown in Figure [2.9](#). The detector is made of a total of 1380 chambers that cover an area of 435 m², covering an angular acceptance range from ± 20 mrad to ± 306 mrad horizontally and from ± 16 mrad to ± 258 mrad. The M1 station is located downstream of the RICH2 detector and is used to improve the transverse momentum measurement in the trigger. M2-5 stations are placed downstream of the calorimeters and are interleaved by 80 cm thick iron absorbers to select penetrating muons, summing to a total thickness of around 20 interaction lengths: the minimum momentum of a muon to cross all the five stations is approximately 6 GeV/c. The muon system is made of Multi-Wire Proportional Chambers (MWPCs), except for the most inner part where triple-GEM detectors are used. MWPCs are filled with a gas mixture of Ar/CO₂/CF₄, in a 40 : 55 : 5 proportion. They achieve a time resolution of around 5 ns. The GEM detectors are filled with a “faster” gas mixture of Ar/CO₂/CF₄, in a 45 : 15 : 40 proportion, that allow for a better time resolution of 3 ns.

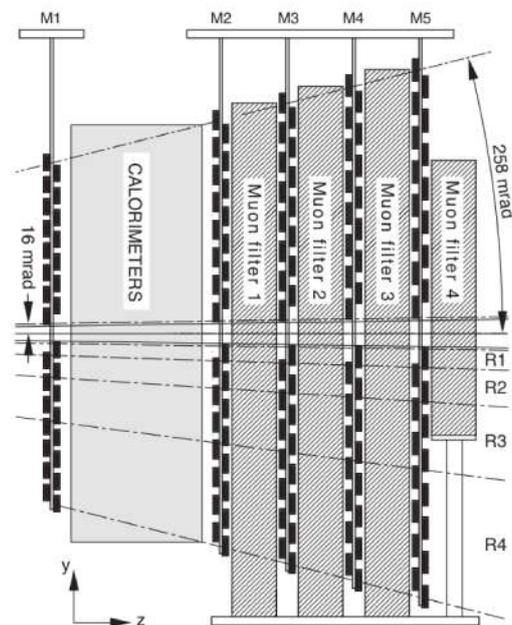


Figure 2.9: Illustration muon system in the yz plane. [25](#)

2.3 Trigger

LHCb operates at an average luminosity which is much lower than the maximum designed luminosity of LHC, with a bunch crossing rate around 40 MHz. At the beginning of the Run II data taking (2015), the storage system was able to write and save events at a rate of 12.5 kHz, corresponding to almost 0.6 GB/s. The purpose of the trigger system is to reduce the rate from 40 MHz to 12.5 kHz, by selecting only a small fraction of potentially interesting events. The LHCb trigger system in use during the Run II is designed to operate in two steps: the first one is an hardware trigger stage called *Level 0* (L0) operating synchronously with the bunch crossing rate, while the second one is a software trigger called High Level Trigger (HLT) which operates on a processor farm. Figure [2.10](#) shows schematically the data flow on the LHCb trigger.

2.3.1 Level 0 trigger

The L0 trigger is designed to spot particles with high transverse momentum or high transverse energy, since heavy-flavor events are characterized by these signatures. L0 trigger decides if an event is accepted according to signals coming from the calorimeters and the muon system, which are connected

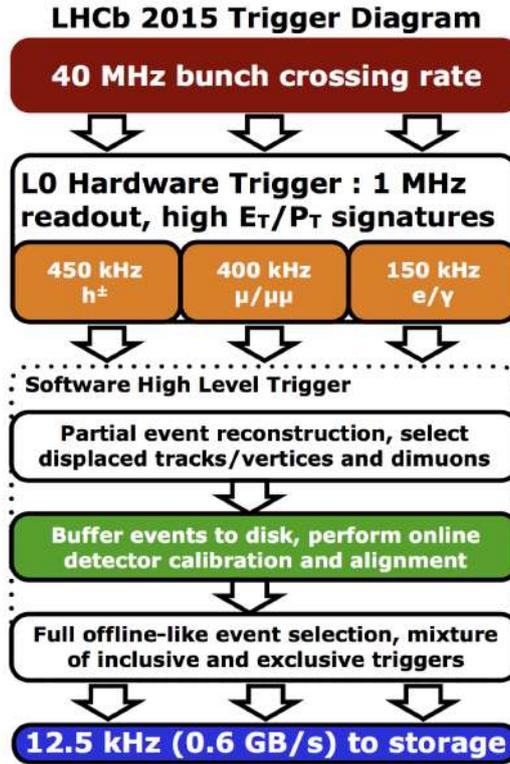


Figure 2.10: Schematic representation of the data flow in the LHCb trigger system (Run II configuration)

to the L0 Decision Unit (L0-DU). The L0 trigger is able to reduce the event rate from 40 MHz to 1 MHz. To reject multiple interactions events, the L0 trigger applies a cut on the number of hits inside the SPD, called Global Event Cut (GEC). If an event passes the GEC, is accepted if one of the following conditions is satisfied:

- **L0-Muon:** a particle identified as a muon has a $p_T > 1.76$ GeV/ c .
- **L0-Dimuon:** two particles identified as muons have the product of their p_T is $p_T^1 \cdot p_T^2 > 1.6^2$.
- **L0-Hadron:** a particle release a cluster in the HCAL and has $E_T > 3.7$ GeV.
- **L0-Photon:** a particle release a cluster in the ECAL and has $E_T > 3$ GeV.
- **L0-Electron:** a particle has an $E_T > 3$, producing a cluster in the ECAL and hits in both the PS and the SPD.

2.3.2 High Level Trigger

An event accepted by the L0 trigger is then processed by the HLT which runs on a large computing facility called Event Filter Farm. HLT is able to finally reduce the event rate from 1 MHz to 12.5 kHz. It is divided in two stages:

- **HLT1:** it uses the information coming from the VELO and the T1-3 stations to partially reconstruct the events accepted by L0. It selects tracks with high p_T and high displacement from the interaction point.
- **HLT2:** it performs the full reconstruction and selection of an event making use of inclusive and exclusive algorithm, selecting all the particles selected or a specific decay mode, respectively.

2.4 Jet reconstruction

The LHCb detector has been designed to study b -hadrons physics. However, thanks to its unique forward geometry and excellent tracking performance, also jet physics can be studied efficiently, providing a complementary phase space with respect to CMS and ATLAS.

Jets are produced at collider experiments as the result of the hadronization and fragmentation of quarks and gluons produced in the hard-scattering processes at the interaction point. They manifest at the detectors as collimated sprays of charged particles in the tracking system and as concentrated clusters of energy in the calorimeters. Although the LHCb detector has excellent tracking performance (97% of track reconstruction efficiency for particles with momentum greater than 5 GeV/ c , with a resolution of 0.5%), calorimeters are not equally well performing in the forward region. Therefore, contrary to CMS and ATLAS, which strongly rely on calorimeters for jet reconstruction, the LHCb experiment make use of both calorimeter clusters and tracking information to efficiently reconstruct jets.

The LHCb jets reconstruction algorithm is made of the following steps:

1. *Particle Flow* provides a list of particles at the input of the clustering algorithm;
2. *Anti- k_t algorithm* [17], which performs the clustering of particles, providing a list of output jets;
3. *E-recombination scheme*, namely jet's four-momentum is computed as a function of the particles momenta;
4. *Jet Energy Correction*, which finally apply a correction to the final energy of the jets, based on Monte Carlo simulations.

The Particle Flow algorithm selects particles based on criteria that depend on the type of particle:

- *Charged particles* are selected depending on the category they belong (long, downstream, upstream or VELO tracks), then requirements are applied to the following quantities
 - the track transverse momentum p_T ;
 - the χ^2 of the Kalman fit;
 - the probability for a track to be reconstructed wrongly by the algorithm P_{ghost} ;
 - the momentum resolution $\frac{\sigma(q/p)}{q/p}$ where q is the particle charge and p its momentum.

The following table show the applied requirements for each particle category:

	long	downstream	upstream	VELO tracks
p_T [MeV/ c]	-	-	> 100	-
χ^2	< 5	< 1.5	< 1.5	< 10
P_{ghost}	< 0.5	-	-	-
$\frac{\sigma(q/p)}{q/p}$	> 10	> 10	> 2	-

Moreover, thanks to the information coming from the RICH detectors and calorimeters, a mass/-type information can be associated to particles (p/\bar{p} , π^\pm , μ^\pm , e^\pm and K^\pm).

- *Isolated neutral particles* are particles that show clusters in calorimeters but are not associated to tracks. In the ECAL, these are mainly photons and π^0 s that decay in two photons. Their discrimination relies on the shape of the calorimeter clusters, therefore a likelihood for the photon or π^0 hypothesis is computed. π^0 s are divided in two categories: *merged* π^0 , when the two photons are almost collinear and show as a single cluster, and *resolved* π^0 , when the two photons are detected as two separate clusters. Requirements are applied to the following quantities:
 - likelihood for the photon hypothesis, $PhotonID$;
 - transverse energy of the cluster E_T ;

- $\chi_{\text{track-cluster}}^2$ associated to the probability that a cluster is associated to a track.

The following table show the applied requirements for each particle category:

	merged π^0	resolved π^0	photons
E_T [MeV]	-	-	> 200
$PhotonID$	-	> -2	-
$\chi_{\text{track-cluster}}^2$	> 25	> 25	> 25

For the HCAL there are no particle identification requirement, while different $\chi_{\text{track-cluster}}^2$ requirements are applied for different energy thresholds: $E < 10$ GeV a $\chi_{\text{track-cluster}}^2 > 25$; $E > 10$ GeV a $\chi_{\text{track-cluster}}^2 > 15$.

- *Non-isolated neutral particles* are particles showing as clusters and are associated to tracks. The selection of this category of particles is done through the following steps:
 1. The following requirements on ECAL and HCAL $\chi_{\text{track-cluster}}^2$ are applied:
 - ECAL: $\chi_{\text{track-cluster}}^2 < 25$
 - HCAL: $E < 10$ GeV a $\chi_{\text{track-cluster}}^2 > 25$; $E > 10$ GeV a $\chi_{\text{track-cluster}}^2 > 15$
 2. clusters in ECAL and HCAL are grouped such that groups do not share tracks;
 3. the expected energy released in the calorimeters E^{exp} is estimated using an empirical parametrization of E/p as function of p , being E ; the cluster energy released by charged particles and p the track momentum;
 4. if the expected energy E^{exp} is larger than 1.8 times the measured energy E^{meas} then the cluster group is discarded;
 5. if $E^{meas} > 1.8E^{exp}$ then E^{exp} is subtracted from E^{meas} : $E^{sub} = E^{meas} - E^{exp}$;
 6. finally, if $E_T > 2$ GeV then E^{sub} is selected as a non-isolated neutral particle and used as input of the anti- k_T algorithm.

The anti- k_t algorithm proceeds with the following steps:

1. For each pair of particles (i, j) of the input list, the following *distance* d_{ij} is evaluated

$$d_{ij} = \min(k_{t,i}^{-2}, k_{t,j}^{-2}) \frac{\Delta_{ij}^2}{R^2} \quad \Delta_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2$$

where $k_{t,i}$, y_i and ϕ_i are, respectively, the transverse momentum, the rapidity and the azimuthal angle of the particle i , and R is a tunable parameter of the algorithm called *radius*.

2. A similar distance is computed between each particle and the beam axis

$$d_{iB} = k_{t,i}^{-2}$$

3. The smallest distance d_{ij} or d_{iB} is found:
 - If the smallest distance is d_{ij} , namely, between the particles i and j , then they are combined into a single particle, summing their four-momenta. The particles i and j are removed from the list of input particles.
 - If the smallest distance is d_{iB} , namely, between the particle i and the beam, then the particle i is considered as a jet and removed from the list of particles.

4. The algorithm restarts from step 2, unless no particles are left to be clustered.

The anti- k_t algorithm tends to cluster soft particles with hard particles instead of producing jets composed only of soft components. The parameter R can be tuned, depending on the experimental environment, the detector properties or the process under study: in order to optimize the jet energy resolution, at LHCb the optimal range has been found to be $0.5 \leq R \leq 0.7$; within this thesis the value has been fixed to $R = 0.5$.

E -recombination scheme After the clustering process, the four-momenta $(E_{\text{jet}}, \mathbf{p}_{\text{jet}})$ of the reconstructed jets can be evaluated using the E -recombination scheme

$$E_{\text{jet}} = \sum_i E_i \qquad \mathbf{p}_{\text{jet}} = \sum_i \mathbf{p}_i$$

At this point, the reconstructed jet energies can be validated using Monte Carlo simulation samples. In these samples both jets and MC jets are reconstructed. The main difference between MC jets (jet_{MC}) and reconstructed jets jet_{reco} is that jet_{MC} are clustered by the anti- k_T algorithm using all the stable truth-level particles which have the true kinematical values, while for jet_{reco} , reconstructed particles are used (invisible particles like neutrinos are removed from the list not to bias the reconstructed energy). To associate a jet_{MC} with a jet_{reco} , the distance in the η - ϕ plane between them $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$ has to be smaller than 0.4; if more than one compatible jet is found in this range, the one with the closest distance is chosen. As expected, jets reconstructed with charged particles show an excellent accord with MC simulation due to the great LHCb tracking performance. On the other hand, in the case of neutral particles, the accord with MC jets is much worse due to the fact that tracking information is not available and only data coming from the calorimeters can be used.

Jet Energy Correction A correction is applied to the energy of the reconstructed jets $E_{\text{jet}}^{\text{reco}}$ to match one of the associated MC jets $E_{\text{jet}}^{\text{MC}}$. This is done by mean of a multiplicative factor k^{MC} , called *Jet Energy Correction*

$$E_{\text{jet}}^{\text{MC}} = k^{\text{MC}} E_{\text{jet}}^{\text{reco}}$$

The k^{MC} factor takes into account the integrated effect of pile-up, noise and non-uniformity of the detector and the same correction is applied to each component of the four-momenta of the reconstructed jets. k^{MC} is evaluated using MC simulations of b -, c -, light quarks and gluons jets and it is found to be non-uniform with respect to the jet η and ϕ , and to the fraction of charged particles in the jet (cpf). Moreover, k^{MC} may depend on the jet flavor.

An additional multiplicative correction should be applied to take account of possible differences between the jet energy real data and MC simulations: this correction is called *Jet Energy Scale*.

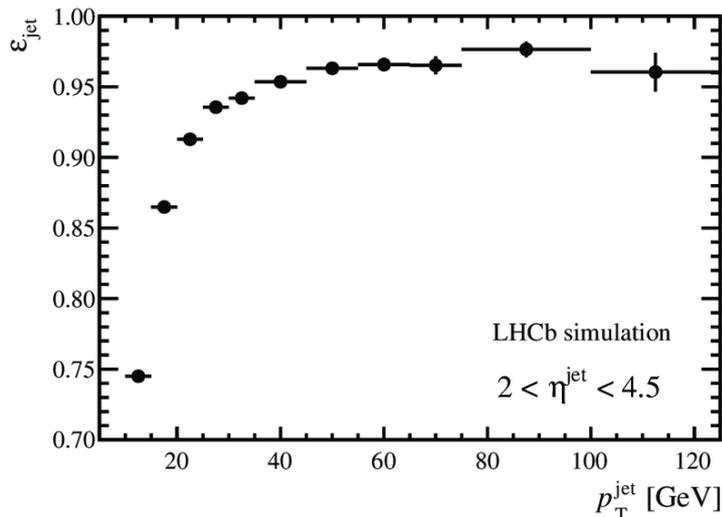
Jet identification efficiency As final step of the reconstruction process, requirements are applied to reject jets originated from noise or high energy isolated leptons:

- number of tracks pointing to the PV $n_{\text{tracks}}^{\text{PV}} \geq 2$
- fraction of transverse momentum carried by a single Particle Flow particle $mpf < 0.8$
- transverse momentum carried by a track $mpt > 1.2 \text{ GeV}/c$
- fraction of charged particles in the jet $cpf > 0.1$

Finally, the jet identification efficiency ϵ_{jet} is evaluated. It is defined as

$$\epsilon_{jet} = \frac{N[\text{reconstructed jets}]}{N[\text{true jets}]}$$

ϵ_{jet} has been measured using $Z^0 \rightarrow \mu\mu + \text{jet}$ MC events at $\sqrt{s} = 7 \text{ TeV}$. As shown in Figure 2.11, the jet identification efficiency is increasing with respect to the jet p_T .

Figure 2.11: Jet identification efficiency as function of the jet p_T

2.5 Heavy-Flavor jet identification

Identification of jets that originate from the hadronization of heavy quarks is crucial for studying Standard Model processes, allowing the search of New Physics. The measurement of charge asymmetries in the pair-production of b -quarks requires, in the first place, the ability to efficiently identify b -jets with a minimal mis-identification of c - and light jets, and then the ability to determine the charge of the b -quark associated to the jet, namely determining if it has been originated from a b - or a \bar{b} -quark. This two tasks are accomplished by two categories of jet identification algorithms:

- **Jet-flavor tagging:** this type of algorithms are used to determine the flavor of the heavy-quark that originated the jet, identifying b -jets, c -jets and light quark jets, with low mis-identification rates.
- **Jet-charge tagging:** this kind of methods are able to identify the charge of the heavy-quark that originated the jet, discriminating b -jets from \bar{b} -jets or c -jets from \bar{c} -jets.

These algorithms usually exploit peculiar features of the events to perform the identification, such as, the presence of a long-lived b - or c -hadron that carries sizable fraction of the jet energy. In the next section, the following tagging algorithms are described in details:

- **Secondary Vertex tagging:** these algorithms exploits the fact that, since b - and c - hadrons are long-lived, they travel a sizable distance before decaying, producing a Secondary Vertex (SV) detached from the PV, from which the products of the decay originate. This algorithm is able to identify b - and c -events, distinguishing them from light-partons events. Additional techniques allow the b versus c discrimination. Therefore, SV tagging only perform jet-flavor tagging but not jet-charge tagging.
- **Muon tagging:** this algorithm exploits the semi-leptonic decays of b -hadrons into muons. These muons usually have a sizable transverse momentum p_T as well as a large transverse momentum relative the the jet axis p_T^{rel} , due to the large mass of the b -quark. Moreover, the charge of the muon is related to the charge of the decaying b -quark. Therefore, the muon tagging algorithm is able to perform both flavor-tagging and charge-tagging; however, the efficiency of this method is strongly limited by the branching ratio of the decay.
- **Jet Charge:** this jet-charge tagging method consists in estimating the charge of the quark from a jet-based observable defined as the momentum-weighted sum of the charges of the particles inside the jet, called *Jet charge*. However, this method alone is not able to identify heavy-flavor jets, therefore is usually coupled to a different jet-flavor tagging algorithm.

2.5.1 Secondary Vertex tagging

b - and c -hadrons produced at LHCb at $\sqrt{s} = 13$ TeV collisions travel an average distance $\langle d \rangle$

$$\langle d \rangle = \beta\gamma c\tau \approx 7 \text{ mm} \qquad \beta = \frac{v}{c} \qquad \gamma = \frac{1}{\sqrt{1 - \beta^2}}$$

where c is the speed of light, v is the speed of the hadron and τ is the average lifetime of the hadron. Therefore, b - and c - events shows a SV, displaced from the PV, where the tracks of the decay products of the heavy quark originate. If a SV is found inside a jet it is tagged as a heavy-flavor jet. The SV tagging algorithm[26] works with the following steps:

1. Tracks selection:

- *long* tracks;
- $p > 5 \text{ GeV}/c$ and $p_T > 0.5 \text{ GeV}/c$;
- χ^2/ndof of the track fit < 3 ;
- χ_{IP}^2 , defined as the variation of the χ^2 of the PV fit when the track is removed > 16 ;
- $P_{\text{ghost}} < 0.3$.

2. The selected tracks are used to build all possible 2-body SVs in the 3D space. Fits are performed to determine the SV position. The two tracks associated to a SV are combined to form a particle where flight direction is defined as the vector that points to the SV from the PV. Its four-momentum is defined as the sum of tracks four-momenta, assuming the π mass. These 2-body particles must satisfy the following requirements:

- Distance Of Closest Approach (DOCA) between two tracks $< 0.2 \text{ mm}$;
- χ^2 associated to the SV fit < 10 ;
- Invariant mass $400 < m < 5279.4 \text{ MeV}/c^2$ (B^0 mass): the lower cut removes strange-hadrons decays, while the upper cut keeps most of the b -hadrons since the mass of the pion is assumed.

3. A 2-body particles is assigned to a jet if $\Delta R < 0.5$, with $\Delta R = \sqrt{(\eta_{\text{jet}} - \eta_{\text{SV}})^2 + (\phi_{\text{jet}} - \phi_{\text{SV}})^2}$. At this stage 2-body particles inside the same jet that share at least one track are associated together, creating a list of n -body particles that do not share tracks. This is repeated for each jet in the event, including all the tracks with $\Delta R < 0.5$, even if they have been associated to a different jet. The resulting n -body particles are called tag_{SV} : the tag_{SV} position is the average of the 2-body SV positions, weighted with the χ^2 of the SV position fit; the tag_{SV} flight direction is the vector that points to its position from the PV; the tag_{SV} four-momentum is the sum of the tracks four-momenta, again, assuming the pion mass.

4. To reduce the light quark jets contamination, the following selection requirements are applied to the tag_{SV} s:

- $p_T > 2 \text{ GeV}/c$
- z -position of the jet must be $< 200 \text{ mm}$
- flight distance/ $p < 1.5 \text{ mm}/(\text{GeV}/c)$
- the flight distance χ^2 , defined as the χ^2 obtained from the PV fit if the tag_{SV} tracks are added to the fit result, must be $> 5\sigma$
- if tag_{SV} is formed by only two tracks with invariant mass compatible with K_S it is rejected
- tag_{SV} must have at most one track with $\Delta R > 0.5$

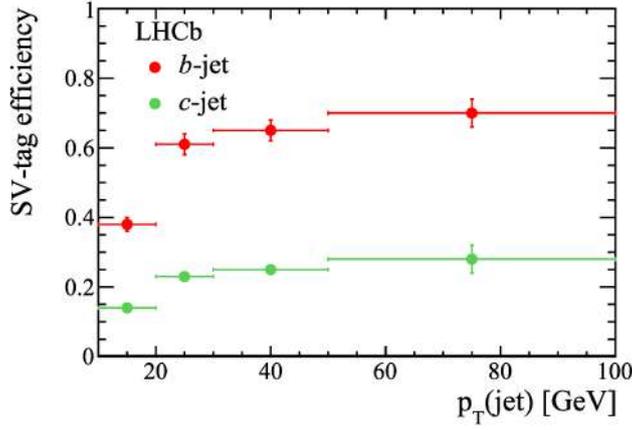


Figure 2.12: Efficiency of the SV-tagging algorithm for b -jets and c -jets, measured on simulated data for $2.2 < \eta < 4.2$, as a function of the jet transverse momentum p_T . [3]

- the tag_{SV} corrected mass is evaluated as $M_{\text{cor}} = \sqrt{M^2 + p^2 \sin^2 \theta} + p \sin \theta$ where $p \sin \theta = |p_T^{\text{miss}}|$ represents the missing component of the momentum with respect to the particle flight direction. To further remove any contamination from kaons or hyperons $M_{\text{corr}} > 600 \text{ MeV}/c^2$ is required.

5. Finally, if more than one tag_{SV} per jet fulfils the requirements, the one with the higher p_T is chosen. A jet is classified as generated by an heavy quark (therefore SV-tagged) if at least one tag_{SV} is found with this algorithm.

SV-tagging performance The performance of the SV flavor tagging algorithm can be measured evaluating the $b(c)$ -tagging efficiency $\epsilon_{b(c)}$ and the light jets mis-identification ϵ_q : the first one is defined as the ratio between the number of tagged $b(c)$ -jets $N_{b(c)}^{\text{tag}}$ over the total number of reconstructed jets $N_{b(c)}^{\text{tot}}$

$$\epsilon_{b(c)} = \frac{N_{b(c)}^{\text{tag}}}{N_{b(c)}^{\text{tot}}}$$

while the latter is defined as the ratio between the number of tagged light jets N_q^{tag} over the total number of light jets $N_{b(c)}^{\text{tot}}$

$$\epsilon_q = \frac{N_q^{\text{tag}}}{N_q^{\text{tot}}}$$

The Tagging performance, evaluated on MC samples of $pp \rightarrow$ dijets events at $\sqrt{s} = 13 \text{ TeV}$ of b -, c - and light quarks, generated with PYTHIA8 [68], is the following:

- for $p_T > 20 \text{ GeV}/c$ the average $\epsilon_{b(c)}$ is around 60% (25%), as shown in Figure 2.12;
- for $p_T < 20 \text{ GeV}/c$ the average $\epsilon_{b(c)}$ is much lower, around 10% (15%);
- the average ϵ_q is around 0.3%, increasing with p_T ;
- in the $2.2 < \eta < 4.2$ range, both $\epsilon_{b(c)}$ have the same behavior with respect to p_T , while ϵ_q is almost constant.

2.5.2 Muon Tagging

An *exclusive* approach to jet-charge tagging is based on the semi-leptonic decay of the b -hadrons into muons (e.g. $B_s^0 \rightarrow \mu^+ \nu X^-$ and $\bar{B}_s^0 \rightarrow \mu^- \bar{\nu} X^+$), correlating the charge of the muon in the final state to the charge of the b -quark that originated the jet. In the LHCb implementation (Run 1 $\sqrt{s} = 7 \text{ TeV}$) (also similar to the one used at CDF), the muon tagging algorithm is applied as follows:

1. At least one of the two jets of the event must have a particle identified as a muon inside (*tagging-muon*)
2. To reduce the charge asymmetries due to the detector biases, the tagging-muon is required to satisfy the following kinematic requirements
 - $p_T > 2 \text{ GeV}/c$
 - $p > 10 \text{ GeV}/c$
3. The decay chain $b \rightarrow c \rightarrow \mu$ contaminates the purity of the tag. To further mitigate this effect, the tagging-muon is required to be the particle with highest momentum among all the displaced tracks.
4. Finally, if the tagging-muon satisfies all the requirements, the jet is classified as an heavy-flavor jet and the charge of the tagging-muon is related to the charge of the b -quark.

The three main sources of dilution of this charge tagger are the following:

- a. $B^0 - \bar{B}^0$ and $B_s^0 - \bar{B}_s^0$ oscillations: the known B -meson oscillation frequencies and the lifetime acceptance at LHCb can be used to estimate the expected dilution.
- b. Decays of the form $b \rightarrow c X$, $c \rightarrow \mu \nu X$: the measured values of b -hadrons production fractions and branching ratios can be used to evaluate the contribution of this dilution channel.
- c. Muon mis-identification: kinematic requirements on the tagging-muon make this contribution negligible.

Combining all the effects, the integrated purity of the muon tagging algorithm is expected to be $73 \pm 4\%$. This value can be validated directly measuring the purity on the fraction of events where both jets present a muon of opposite charge, thus producing an unambiguous tag: this analysis gives an integrated purity of $70.3 \pm 0.3\%$. Furthermore, selecting events with one jet and a fully reconstructed $B^+ \rightarrow J/\psi K^+$ or $B^+ \rightarrow \bar{D}^0 \pi^+$, one can exploit the unambiguous tag provided by the B^+ to obtain another validation channel, which provide a measured integrated purity of $73 \pm 3\%$.

Even though the muon tagging achieve good charge tagging purity, it strongly limits the available statistics due to the intrinsic constraint put by the branching ratio of the b decaying semi-leptonically into a muon, which is around 10%.

2.5.3 Jet Charge tagging

This inclusive technique consists in evaluating the momentum-weighted average of the charge of the particles associated with each jet, called *jet charge*

$$Q_{\text{jet}} = \frac{\sum_i q_i (\mathbf{p}_i \cdot \mathbf{p}_{\text{jet}})^{0.5}}{\sum_i (\mathbf{p}_i \cdot \mathbf{p}_{\text{jet}})^{0.5}} \quad (2.1)$$

where the two sums are extended over all the tracks in the jet, q_i and \mathbf{p}_i are the charge and the 3-momentum of the particle and \mathbf{p}_{jet} is the 3-momentum of the jet; the exponent 0.5 is chosen to maximize the discriminating power of the tagger. The algorithm make a decision using the difference between the charges of the two jets of an event $\Delta Q = Q_1 - Q_2$:

- if $\Delta Q > 0$, jet 1 is considered to be the b -jet;
- if $\Delta Q < 0$, jet 2 is considered to be the b -jet;

Although being an inclusive method, the jet charge tagging is not able to achieve a good b versus \bar{b} separation.

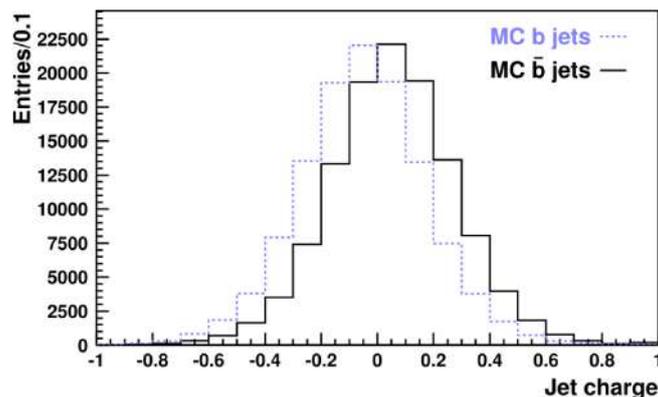


Figure 2.13: Distribution of the momentum-weighted jet charge on a MC sample at CDF. [4](#)

2.6 Development of new tagging algorithms

This brief review puts the necessity of developing new tools for the efficient identification of jets originated by heavy flavor quark production. New methods should exploit the complex structure of jets and all the available experimental variables, trying to unveil correlations between particles that could enhance the identification process. Modern Machine Learning techniques provide very powerful tools for attacking these classification tasks and some of them are already employed at the CMS [67](#), [15](#) and ATLAS [56](#) experiments. LHCb still lacks of official high-performing algorithms for the tagging of dijet events that could give a boost to the study of production asymmetries of heavy quarks in the forward region. The main purpose of this thesis is exploring the new-born field of Quantum Machine Learning trying to develop a *quantum* tagger for the b vs \bar{b} classification, comparing the results with *classical* Machine Learning models and the Muon Tagging algorithm. The performance of the charge-taggers will be compared in terms of the following metrics:

- Tagging efficiency ϵ_{eff} , defined as the ratio between the number of tagged events N^{tag} over the total number of events N^{tot}

$$\epsilon_{\text{eff}} = \frac{N^{\text{tag}}}{N^{\text{tot}}}$$

- Mistag rate ω , defined as the ratio between the number of events that are tagged incorrectly $N_{\text{wrong}}^{\text{tag}}$ over the total number of tagged events N^{tag}

$$\omega = \frac{N_{\text{wrong}}^{\text{tag}}}{N^{\text{tag}}}$$

- Tagging power ϵ_{tag} , defined as

$$\epsilon_{\text{tag}} = \epsilon_{\text{eff}}(1 - 2\omega)^2$$

which takes into account both the efficiency and the purity of the tagger and represents the effective fraction of tagged events that are *useful* for the estimation of the $A_{\text{FC}}^{b\bar{b}}$, therefore being directly related to the achievable precision.

2.7 Data-set selection

Data samples come from official LHCb Monte Carlo simulations of dijet produced from pp collisions at the C.o.M. energy of $\sqrt{s} = 13$ TeV (Run2 conditions). $b\bar{b}$ pairs production is done using PYTHIA8 generator that also performs the jet fragmentation. EvtGen [62](#) simulate b -hadrons decays and Geant4 [6](#) simulate the interactions with the detector. Finally, the simulated signals are digitized and the events are reconstructed using the LHCb software, which performs the jet reconstruction applying the ParticleFlow algorithm and the anti- k_T clustering algorithm. PYTHIA8 includes production processes at LO QCD and LO Z^0/γ decaying into a $b\bar{b}$ pair (the relevant Feynman diagrams are shown in Figure [1.3](#)); events are generated with two opposite polarization of the magnet.

The data-set comes split in different ROOT files, depending on the the momentum exchanged in the hard collision \hat{p} (for QCD events), the polarization of the magnet (up or down) and the production mechanism (QCD or Z^0/γ). Here is shown a list of the files and the number of generated events N_{gen} inside each:

Filename	Processes	Magnet	\hat{p} [GeV/c]	N_{gen}
Dijet_bb_pt10_15_dw.root	QCD	down	[10,15]	1 005 945
Dijet_bb_pt10_15_up.root	QCD	up	[10,15]	1 004 373
Dijet_bb_pt15_20_dw.root	QCD	down	[15,20]	1 005 555
Dijet_bb_pt15_20_up.root	QCD	up	[15,20]	1 000 166
Dijet_bb_pt20_50_dw.root	QCD	down	[20,50]	1 037 292
Dijet_bb_pt20_50_up.root	QCD	up	[20,50]	1 038 062
Dijet_bb_pt50_dw.root	QCD	down	> 50	1 009 737
Dijet_bb_pt50_up.root	QCD	up	> 50	1 040 583
Dijet_Z_dw.root	Z^0/γ	down	All	117 623
Dijet_Z_up.root	Z^0/γ	up	All	118 199

Table 2.2

The number of generated events is almost the same for all the QCD files (around 1 000 000) and for the two Z^0 files (around 120 000).

The SV-tagging algorithm has been applied to the reconstructed jets for flavor identification and events with at least two b -tagged jets are selected. No event in the simulated data presents more the two reconstructed jets tagged as b -jets from the SV-tagger, therefore this analysis neglects this type of events, assuming that they occur with a very low probability: in fact, the fraction of events having multiple dijet candidates has been estimated at LHCb, in an analysis for the measurement of the the $b\bar{b}$ - and $c\bar{c}$ -dijet cross sections, and was found to be 0.4% [3].

Each of the two reconstructed jets is associated with the closest Monte Carlo jet, according to the distance in the η - ϕ plane $\Delta R = \sqrt{(\eta_{\text{jet}}^{\text{MC}} - \eta_{\text{jet}}^{\text{reco}})^2 + (\phi_{\text{jet}}^{\text{MC}} - \phi_{\text{jet}}^{\text{reco}})^2}$, which is required to be < 0.4 .

The data-set is publicly available on the CERN Open Data Portal[21].

Chapter 3

Quantum Computing and Quantum Machine Learning

3.1 Introduction to Quantum Computation

The first step towards the development of an organic theory of quantum system is commonly considered to be the idea of *quantized energy* introduced by M. Plank in 1900 within his studies on the black-body radiation. The next 40 years have seen the fast and prolific rise of Quantum Mechanics thanks to the exceptional contributions of A. Einstein, M. Born, L. de Broglie, W.K. Heisenberg, P. Jordan, E. Schrödinger, P. Dirac and J. von Neumann. Figure 3.1 shows a sketched timeline of the early-history of Quantum Mechanics.

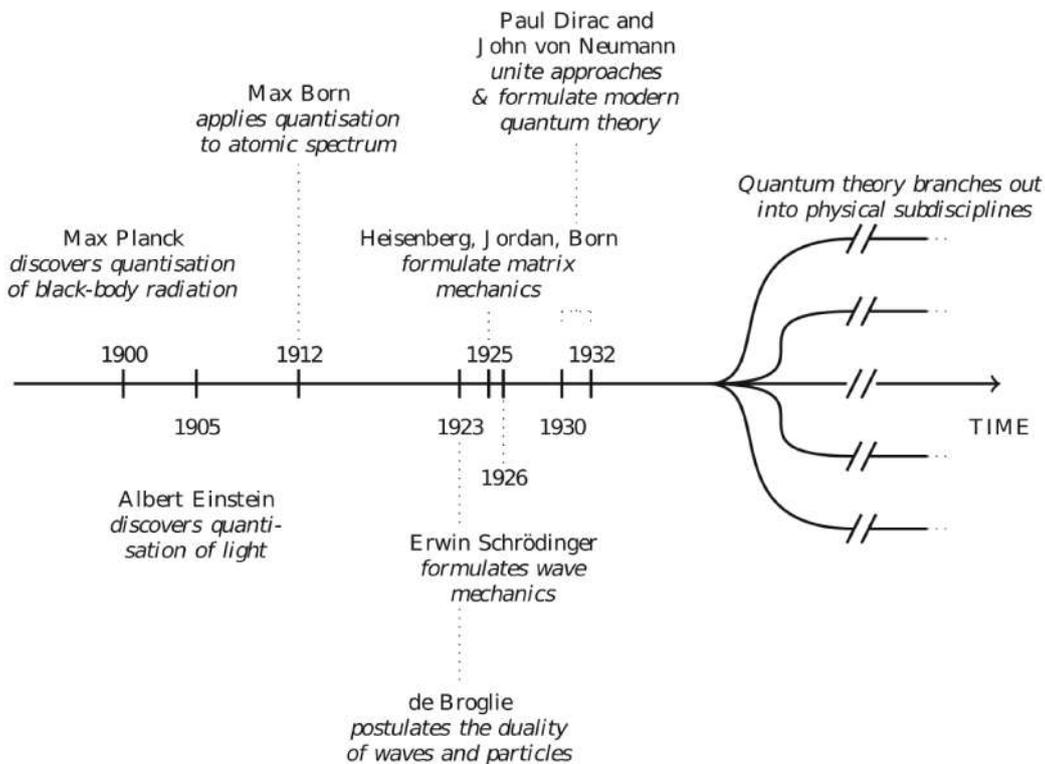


Figure 3.1: Timeline of the early-history of Quantum Mechanics. 64

The Quantum Theory is often considered to be the most accurate physics theory ever developed: it is able to provide prediction of physical observables with an astonishing precision (e.g. the anomalous magnetic moment of the electron $a = (g_e - 2)/2$ is predicted by the QED with a precision better

than one part in a billion), while still incorporating the Classical Mechanics as a limit case. However, despite of the uncontested superiority of the Quantum Theory, non-quantum theories are still widely used: this apparent paradox is justified by the fact that the complexity of quantum theories behaves badly with the size of the system. For example, a quantum system made of one spin 1/2 particle can be described exactly, in the density matrix formalism, using 2×2 complex matrix, that fits in 64 B of memory, assuming double-precision; on the other hand, a quantum system of 100 spin 1/2 particles requires dealing with a $2^{100} \times 2^{100}$ complex matrix that requires around 2.57×10^{46} PB of memory, again assuming double-precision. It is easily understood that an exact description of such a system is impracticable with current technologies and that the study of many-body quantum systems cannot avoid dealing with approximations. Luckily, intrinsic quantum effects only shows up at very small scales and in most of the cases they are negligible with respect to the higher order classical behaviors. Moreover, highly efficient approximation techniques have been developed for studying many-body quantum systems.

A natural question arises from these simple considerations: can we exploit the intrinsic complexity of quantum systems to perform calculations? Since the debate about the non-locality of the Quantum Theory, questions aroused about the possibility of an information theory of quantum systems:

- What is *quantum information*?
- Can we build a *quantum computer*?
- How can we *encode data* inside a quantum system?
- How can we formulate *algorithms* for such devices?
- How can a quantum computer give an *advantage* with respect to a classical computer?

These are the questions that the Quantum Computing and Quantum Information fields try to answer. The research on these fields has gained a great boost in 1994 when P. Shor proved that a quantum algorithm can solve the factorization problem in polynomial time [66]. Soon after, in 1996, L. Grover developed a quantum algorithm that can perform an unstructured search in a database of size n using only $O(\sqrt{n})$ evaluations (when a classical algorithm needs at least $O(n)$) [40]. Despite of the large improvements in the field, the exact domain of this new technology is still unclear and a lot of research is necessary to try to provide answers to previous questions.

The following sections will provide an overview of the theoretical framework behind Quantum Computing, presenting the basic concepts of *qubit* and *entanglement*.

3.2 The Qubit

In the Classical Information theory, the information unit is the *bit* which represents a logic state that can have only two possible values: 0 (or *false*, *off*, etc.) or 1 (or *true*, *on*, etc.). Bits must be concatenated into strings to be able to represent a larger number of different states: an n -bits string can represent a maximum of 2^n different states, e.g. the task of counting integers from 0 to 1000 can be accomplished using at least 10 bits, since $2^{10} = 1024$.

In Quantum Computing, the elementary information unit is the *qubit*, a two-levels quantum system that can be measured in two states $|0\rangle$ and $|1\rangle$, which form an orthonormal basis, called *computational basis*, of the 2-dimensional Hilbert space \mathcal{H} where all the possible states of the qubit are defined. The most general pure state $|\psi\rangle$ of a qubit can be written as

$$|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle \tag{3.1}$$

where $\alpha_0, \alpha_1 \in \mathbb{C}$, satisfying the normalization condition $|\alpha_0|^2 + |\alpha_1|^2 = 1$.

\mathcal{H} is isomorphic to the \mathbb{C}^2 vector space, so we can conveniently associate the canonical basis of \mathbb{C}^2 to

computational basis of \mathcal{H}

$$\begin{aligned} |0\rangle \in \mathcal{H} &\rightarrow \begin{pmatrix} 1 \\ 0 \end{pmatrix} \in \mathbb{C}^2 \\ |1\rangle \in \mathcal{H} &\rightarrow \begin{pmatrix} 0 \\ 1 \end{pmatrix} \in \mathbb{C}^2 \end{aligned}$$

This isomorphism allows us to represent quantum states as complex (normalized) vectors and linear operators as complex matrices.

Bloch sphere A convenient way to graphically represent a 1-qubit state is the Bloch sphere: the expression [3.1](#) can be rewritten (up to an irrelevant global phase rotation) as:

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + \exp(i\phi) \sin \frac{\theta}{2} |1\rangle \quad (3.2)$$

where θ and ϕ are real number with $0 \leq \theta \leq \pi$ and $0 \leq \phi \leq 2\pi$ that can be interpreted as spherical coordinates. Therefore, the state $|\psi\rangle$ can be visualized as a the \mathbb{R}^3 vector $(\sin \theta \cos \phi, \sin \theta \sin \phi, \cos \theta)$ that represents a vector pointing from the origin to the surface of a sphere, called Bloch sphere (Figure [3.2](#)).

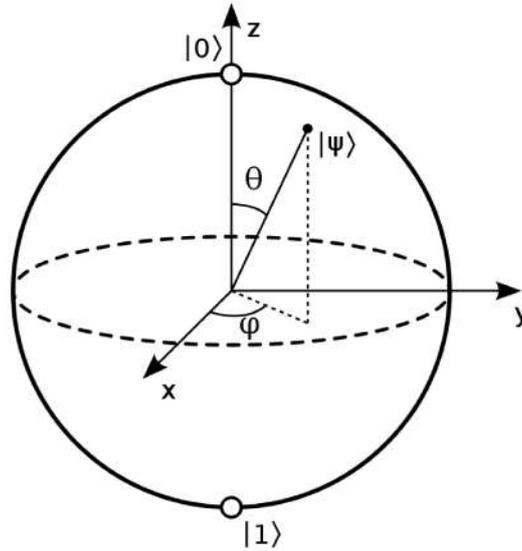


Figure 3.2: Representation of a generic pure state $|\psi\rangle$ on the Bloch sphere. [29](#)

Density matrix and mixed states The quantity $\rho_{\text{pure}} = |\psi\rangle \langle \psi|$ is called *density matrix* of the pure state $|\psi\rangle$ and represents a projector operator:

$$|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle \quad \rho_{\text{pure}} = |\psi\rangle \langle \psi| = \begin{pmatrix} |\alpha_0|^2 & \alpha_0 \alpha_1^* \\ \alpha_0^* \alpha_1 & |\alpha_1|^2 \end{pmatrix}$$

ρ_{pure} has the following properties:

- $\rho_{\text{pure}}^\dagger = \rho_{\text{pure}}$ (ρ is Hermitian)
- $\forall |\phi\rangle : \langle \phi | \rho_{\text{pure}} | \phi \rangle \geq 0$ (ρ is positive semi-defined)
- $\text{Tr} \rho_{\text{pure}} = 1$ (the diagonal terms of ρ_{pure} are the probabilities of finding $|\psi\rangle$ in a basis state and they sum to 1)
- $\rho_{\text{pure}}^2 = \rho_{\text{pure}}$ (ρ_{pure} is a projector)

The density matrix framework allows us to define a *mixed state* ρ_{mix} : let $\{p_i\}$ be probabilities (summing the 1) associated to a set of pure state $\{|\psi_i\rangle\}$, then the following density matrix represents a mixed state:

$$\rho_{\text{mix}} = \sum_i p_i |\psi_i\rangle \langle \psi_i|$$

ρ_{mix} does not satisfy the same properties of ρ_{pure} , in particular:

- $\rho_{\text{mix}}^2 \neq \rho_{\text{mix}}$
- $\text{Tr} \rho_{\text{mix}}^2 < \text{Tr} \rho_{\text{mix}} = 1$

Multi-qubit states The description of a multi-qubit quantum system can be easily gained using the concept of *tensor product* between Hilbert spaces. An n -qubit quantum system is described by the tensor product of n single qubits:

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \dots \otimes |\psi_n\rangle \quad |\psi_1\rangle, |\psi_2\rangle, \dots |\psi_n\rangle \in \mathcal{H} \quad |\psi\rangle \in \bigotimes_{i=1}^n \mathcal{H}$$

A completely generic n -qubits state is described by 2^n complex coefficients:

$$|\psi\rangle = \alpha_1 |0\dots 00\rangle + \alpha_2 |0\dots 01\rangle + \alpha_3 |0\dots 10\rangle + \dots + \alpha_{2^n} |1\dots 11\rangle \quad \alpha_i \in \mathbb{C}$$

Generalizing density matrices of pure and mixed states is straightforward: let $|\psi\rangle$ be an n -qubits state, then

$$|\psi\rangle = \sum_{i=1}^{2^n} \alpha_i |i\rangle \quad \rho_{\text{pure}} = |\psi\rangle \langle \psi| \quad \rho_{\text{mix}} = \sum_{i,j} \alpha_{ij} |i\rangle \langle j| \quad \alpha_i, \alpha_{ij} \in \mathbb{C}$$

Quantum measurements The state of a quantum system can be accessed only through measurements operated on the qubits of the system: measurements often occurs in the computational basis, therefore it is useful to precisely define what is the exact meaning of performing a measurement on a quantum state. For the seek of simplicity, let us assume to deal with a generic 1-qubit state as in the expression 3.1. In this case, the two possible outcomes of a measurement on the computational basis (0 or 1) are associated to the two projectors on the eigenspaces

$$P_0 = |0\rangle \langle 0| \quad P_1 = |1\rangle \langle 1|$$

Therefore, the probabilities associated to the two measurement outcomes are:

$$p(0) = \langle \psi | P_0 | \psi \rangle = |\alpha_0|^2 \quad p(1) = \langle \psi | P_1 | \psi \rangle = |\alpha_1|^2$$

Then, the quantum state collapses into the eigenstate associated to the measurement outcome: say that a 0 has been obtained, the quantum state after the measurement is the following

$$|\psi\rangle \rightarrow \frac{P_0 |\psi\rangle}{\sqrt{\langle \psi | P_0 | \psi \rangle}} = |0\rangle$$

So, the final quantum state gets destroyed and it has to be rebuilt if repeated measurements are needed. This type of measurements on a system of qubits are called *projective measurements* on the computational basis.

For an n -qubits system, the generalization is straightforward: a computational basis measurement can be seen as sampling a binary string of length n from the distribution defined by quantum state. To each one of the 2^n possible outcomes is associated a projector that defines the eigenspace on which the state collapses after the measurement.

3.3 Entanglement

The study of multi-qubits systems allows for the introduction of fundamental concept of quantum computing which has no classical counterpart: the *entanglement*. Let A and B be two quantum systems (e.g. they can be two qubits or, more in general, two groups of qubits), then a state of the composite system is called *separable* if it can be written as a classical probability distribution p_i over uncorrelated states of the two subsystems ρ_A and ρ_B :

$$\rho = \sum_i p_i \rho_i^A \otimes \rho_i^B$$

A non-separable state is called an entangled state: an example of non-separable pure state is provided by the Bell states

$$\begin{aligned} |\phi^\pm\rangle &= \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B \pm |1\rangle_A \otimes |1\rangle_B) \\ |\psi^\pm\rangle &= \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B \pm |1\rangle_A \otimes |0\rangle_B) \end{aligned}$$

These states are also called *maximally entangled* states, since the outcome of a measurement on the subsystem B can be exactly determined by a measurement on the subsystem A . This means that the two subsystem are maximally correlated, but this kind of correlation is purely quantum and cannot be achieved by mean of some sort of classical operations, as Bell proved in 1964[10]. This phenomenon points toward the incompatibility of locality and realism in Quantum Mechanics and represent the most unique feature of the Quantum Information theory.

3.4 Quantum circuits

Classical computers operate on bit-strings by means of *logical function*. Formally, a logical function is function $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ that receives a n -bits string as input and gives an m -bits string as output, which is completely determined by its *truth-table*, defining the output bit-string for every possible input bit-string. Every logical function can be decomposed in elementary logical function, called *logical gates*, such as the AND gate, the OR gate, the NOT gate or the XOR gate. Elementary logical gates can be combined in *logical circuits* that can be implemented electronically.

In analogy with the classical computing, qubits can be manipulated by mean of *quantum gates* that are realised by unitary operators¹ acting on the quantum state.

1-qubit gates are described by 2×2 complex unitary matrices. Among these, some of the most important are the *Pauli operators*:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (3.3)$$

The X gate is also called NOT gate since its action flips the qubit in the computational basis: $X|0\rangle = |1\rangle$ and $X|1\rangle = |0\rangle$. Other commonly used quantum gates are the *Hadamard gate* (denoted H), the *phase gate* (denoted S) and the $\pi/8$ gate (denoted T):

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix} \quad (3.4)$$

The common symbols representing these gates are show in Figure 3.3: wires represent qubits and a chain of linked gates has to be read from left to right.

¹An operator O on an Hilbert space is called *unitary* if $OO^\dagger = O^\dagger O = \mathbb{1}$, where $\mathbb{1}$ is the identity operator.

Hadamard	$\text{---} \boxed{H} \text{---}$	$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Pauli- X	$\text{---} \boxed{X} \text{---}$	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli- Y	$\text{---} \boxed{Y} \text{---}$	$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli- Z	$\text{---} \boxed{Z} \text{---}$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Phase	$\text{---} \boxed{S} \text{---}$	$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$	$\text{---} \boxed{T} \text{---}$	$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$

Figure 3.3: Name, circuitual symbol and matrix representation of the common 1-qubit gates. 55

The operators that rotates of an angle θ the state vector around the x , y and z axis on the Bloch sphere are the *rotation gates*. They are defined by exponentiating the Pauli operators (3.3):

$$R_x(\theta) = e^{-i\theta X/2} = \cos \frac{\theta}{2} \mathbb{1} - i \sin \frac{\theta}{2} X = \begin{pmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix} \quad (3.5)$$

$$R_y(\theta) = e^{-i\theta Y/2} = \cos \frac{\theta}{2} \mathbb{1} - i \sin \frac{\theta}{2} Y = \begin{pmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{pmatrix} \quad (3.6)$$

$$R_z(\theta) = e^{-i\theta Z/2} = \cos \frac{\theta}{2} \mathbb{1} - i \sin \frac{\theta}{2} Z = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix} \quad (3.7)$$

Controlled gates performs controlled operations of the type “If A is true, then do B ”. Let U be n arbitrary 1-qubit unitary operation. A *controlled- U* operation is a 2-qubits operation, with a *control qubit* and a *target qubit*: in the control qubit is set to $|1\rangle$ then U is applied to the target qubit, otherwise the target qubit is left unmodified. A controlled operation is represented as in Figure 3.4 the control qubit (on the top) is wired to the controlled gate (on the bottom).

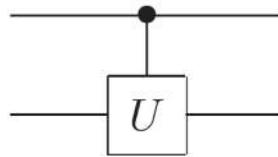
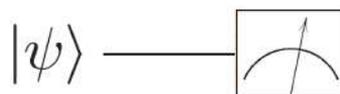


Figure 3.4: Controlled- U gate. The top wire represent the control qubit while the bottom wire is the target qubit.

The most common controlled gates are the Controlled-NOT (CNOT) gate and the Controlled-Z (CZ) gate: their matrix representations and circuitual symbols are shown in Table 3.1

Measurement gates are used to denote projective measurements on the computational basis and they are represented as “meter” symbols as shown below



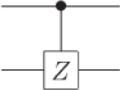
Controlled-NOT		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Controlled-Z		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$

Table 3.1: Name, circuitual symbol and matrix representation of the CNOT and CZ gates.

Projective measurements on a different basis or even non-projective measurements can be always be opportunely expressed as projective measurements on the computational basis by mean of additional qubits (called *ancilla qubits*) and unitary operators. Moreover, measurements performed at an intermediate stage of a quantum circuit can always be moved to the end of the circuit without loss of generality (*principle of deferred measurement* [55]). Another important statement about measurements on quantum circuits is the so called *principle of implicit measurement* [55], which states that, without loss of generality, any unmeasured qubit at the end of the circuit may be assumed to be measured. In other words, measuring an unmeasured qubit at the end of the circuit does not change the probability distribution of the other measurements.

3.5 Machine Learning with quantum circuits

3.5.1 Introduction to Machine Learning

Machine Learning (ML) is a branch of Artificial Intelligence that consists in making a computer learn directly from data how to solve a problem, instead of providing explicitly a solving algorithm. ML aims to automate the process of generalizing from experience in order to make predictions in new and unknown situations. ML is a very powerful tool in tasks where the patterns from which a prediction has to be derived are very complex and we do not have a sufficient understanding of those patterns to write an explicit solving algorithm. The ML approach starts instead with a very general and agnostic mathematical model and uses the data to adapt it to the case. Usually, when looking at the final model, it is not possible to gain information that could allow the development of an explicit algorithm; however, the model has to be considered as a black box that has learned the pattern of the system and it produces reliable predictions.

ML includes three main branches, each one targeting a different class of problems:

- *Supervised learning*: during the training phase, the model is allowed to process a set of input data paired with the expected output of the model.
- *Unsupervised learning*: the model is not provided with a target output but, instead, has to learn on its own the patters arising from the input data.
- *Reinforcement learning*: the model get trained with a rewards/punishment mechanism and it has to optimize a final score by trial and error.

This thesis will focus on the task of supervised learning employing quantum circuits as models.

3.5.2 Supervised learning

The main structure of generic supervised machine learning algorithm as shown in Figure 3.5 is the following:

1. **Problem definition**: given an input domain \mathcal{X} and an output domain \mathcal{Y} , a training data-set $\mathcal{D} = \{(x_1, y_1), \dots, (x_M, y_M)\}$ with $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ being M couples of training inputs x_i and expected outputs y_i where the relationship between the x_i and the y_i is unknown, we want to

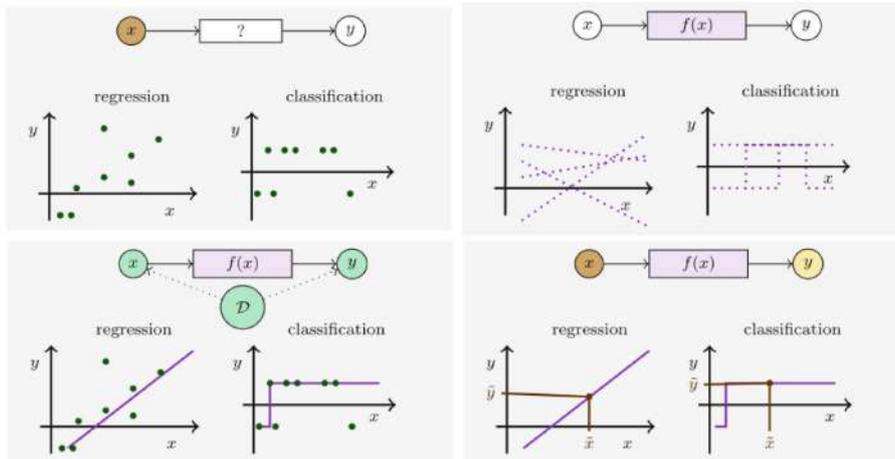


Figure 3.5: The four main steps of a supervised learning algorithm. Upper left: a train data-set is given. Upper right: A model family is chosen. Lower left: the model is trained. Lower right: the model is employed on new data. [64](#)

approximate this relationship using an ML model (Figure [3.5](#), upper-left). Within this work, the input domain is \mathbb{R}^N : therefore, input data are N -dimensional real vectors called *feature vectors*.

2. **Choice of the model family:** a *model family* is a function

$$y = f(x, \theta)$$

with $x \in \mathcal{X}$, $y \in \mathcal{Y}$ and $\theta = \{\theta_1, \dots, \theta_D\}$ is a set of real parameters that has to be optimized during training process (Figure [3.5](#), upper-right). The model family may also depend on hyper-parameters that are not included explicitly in the notation.

3. **Training of the model:** the model is trained by fitting the parameters and the hyper-parameters to the data \mathcal{D} . This means that a specific model function is chosen from the model family optimizing and fixing the model parameters θ . A test data-set can be used to validate the performance of the model on new, unseen data (Figure [3.5](#), lower-left).

4. **Employment of the model:** once trained, the model can be used to predict output values (Figure [3.5](#), lower-right).

The training phase is the most critical step for a ML algorithm: the goal of the training is to select the best model from a model family. This can be accomplished defining an objective function $L(\theta)$, called *loss function*, that quantifies the quality of a particular choice of parameters θ : therefore, training can be considered an optimization problem. Different choices of the loss function lead to different training strategies and results. In this work we have employed the *mean squared error loss*, defined as

$$L^{MSE}(\theta) = \frac{1}{M} \sum_{i=1}^M (f(x^i, \theta) - y^i)^2 \quad (3.8)$$

and the *cross-entropy loss*, defined as

$$L^{CE}(\theta) = \sum_{i=1}^M -y^i \log p^i - (1 - y^i) \log(1 - p^i) \quad (3.9)$$

where $p^i = f(x^i, \theta)$ are the outputs of the model, which are assumed to be probabilities. The loss function $L(\theta)$ has to be minimized with respect to the parameters of the model to obtain the optimal parameters configuration $\bar{\theta}$:

$$\bar{\theta} = \arg \min_{\theta} L(\theta)$$

Once the loss function is fixed, training a ML model reduces to the mathematical problem of optimizing the loss function. However ML tasks often lead to hard optimization problems since the loss function

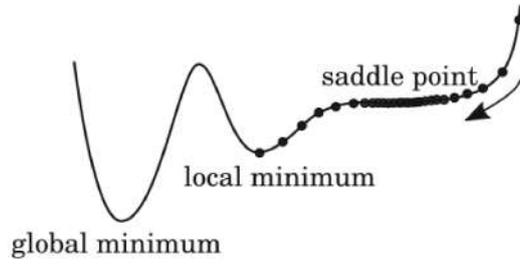


Figure 3.6: Illustration of several steps of a gradient descent method: it gets stuck in local minima and convergence becomes slow at saddle points. [64]

can be non-convex. This kind of problem are much less known in Optimization theory. Therefore, popular methods are iterative searches such as the gradient descent method.

Gradient descent algorithms are a class of local optimization methods that iteratively updates the parameters θ of the loss function $L(\theta)$ towards the direction of the steepest descent

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla L(\theta^{(t)}) \quad (3.10)$$

where η is a tunable parameter called *learning rate* and t is the iteration index. Figure 3.6 shows an example of a training updates of a 1-parameter model. Since the gradient of the loss function is used, these methods can get stuck on local minima and their rate of convergence can get prohibitively low on saddle points where $\nabla L \rightarrow 0$. To mitigate these effects, several methods have been derived from the naive gradient descent: in this work the ADAM [47] optimizer has been used.

3.6 Parametrized quantum circuits

Parametrized Quantum Circuits (PQC), also known as *variational circuits* are typically composed of a combination of fixed gates (like *CNOT*, *CZ* or *H* gates) and adjustable gates with tunable parameters (like the *RX*, *RY*, *RZ* gates) that can be used as Quantum Machine Learning (QML) models. Even at low circuit complexity, PQC are able to generate non-trivial outputs that cannot be efficiently simulated by classical hardware. The essential components of a QML model based on a PQC are

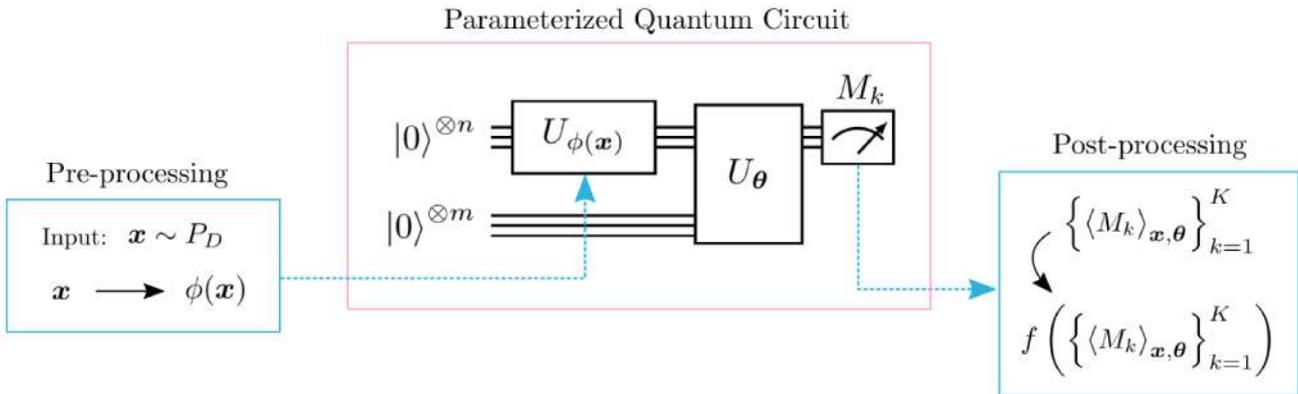


Figure 3.7: Main components of a supervised ML model based on a PQC. [12]

sketched in Figure 3.7, where classical stages are shown in light blue, while quantum stages in pink:

1. **Pre-processing:** a feature vector x is sampled from the training data-set P_D and gets transformed following a classical pre-processing scheme by a function ϕ that maps $x \rightarrow \phi(x)$. This step can include *feature selection* and normalization techniques.
2. **PQC:** the transformed data is mapped to the parameters of an *encoder* circuit $U_{\phi(x)}$ which embeds the data into an n -qubits state. A variational circuit U_{θ} with tunable parameters θ , which an possibly act on an extended qubit register, implements the core operation of the

model. This is followed by the estimation of a set of expectation values $\{\langle M_k \rangle_{x,\theta}\}_{k=1}^K$ from measurements on the final quantum state.

3. **Post-processing:** the measured expectation values $\{\langle M_k \rangle_{x,\theta}\}_{k=1}^K$ get mapped to a model prediction by a post-processing function $y = f(\{\langle M_k \rangle_{x,\theta}\}_{k=1}^K)$

All the component of the model, including the pre-processing and the post-processing, can be parametrized and optimized during the training phase.

Data encoding is the first step of the quantum stage of a QML model based on PQC: it consists in mapping a vector of classical data x into an n -qubits quantum state $x \rightarrow |x\rangle$. This can be accomplished in several different ways. In this work we make use of the *Amplitude Encoding* and the *Angle Encoding* techniques that are defined in this way

- **Amplitude Encoding:** it encodes up to 2^n features into the amplitudes of the state vector of an n -qubits system, or, equivalently, a vector of n features can be encoded using $\log_2 n$ qubits. This means that amplitude encoding provides an exponential advantage when scaling to large number of features. If the number of features to encode is smaller than 2^n it is possible to pad the remaining amplitudes with constant values. With exactly 2^n features, the amplitude encoding maps the feature vector x into $|x\rangle$ in this way

$$|x\rangle = \sum_{i=1}^{2^n} x_i |n_i\rangle \quad (3.11)$$

where x_i refers to the i^{th} feature and $|n_i\rangle$ is the i^{th} vector of the computational basis. This definition requires the x vector to be normalized, i.e. $\sum_i |x_i|^2 = 1$.

- **Angle Encoding:** it encodes up $3n$ data features into an n -qubits state. Each qubit can encode up to 3 features encoded as the angle of rotation around one of the three axes, i.e. with $i = 0, \dots, n - 1$ the i^{th} qubit state is prepared as:

$$|\psi\rangle_i = R_x(x_{3i+1})R_y(x_{3i+2})R_z(x_{3i+3})|0\rangle_i \quad (3.12)$$

where R_x , R_y and R_z are the rotation gates defined in (3.5 - 3.7). The full n -qubits state $|x\rangle$ that represents the feature vector x is constructed as the tensor product of the single $|\psi\rangle_i$ states:

$$|x\rangle = \bigotimes_{i=0}^{n-1} |\psi\rangle_i \quad (3.13)$$

Rotational gates are periodic unitary operator with respect to the parameter θ : $R_i(\theta) = R_i(\theta + 4\pi)$. Therefore, each feature value must be scaled to the $[0, 4\pi]$ range; however, more restricted ranges could improve the training performance.

Variational circuits represents the core of PQC circuits. Similar to the universal approximation theorem in neural networks[12], there always exists a quantum circuit that can represent a target function within an arbitrary small error: however such a circuit may be exponentially deep and impractical to implement on current quantum hardware. Therefore, often the circuit design follows a fixed structure of gates (*circuit ansatz*). Even if the dimension of the Hilbert space grows exponentially with the number of qubits, the number of free parameters to be optimized scale usually as a polynomial of the qubit count.

Figure 3.8 shows the structure of a commonly used variational ansatz for QML model: this circuit presents a first stage of tunable rotational gates applied to the single qubits and a second stage of controlled gates which provide entanglement between the qubits. This circuitual structure can be repeated multiple time increasing the number of parameters and therefore the complexity of the model.

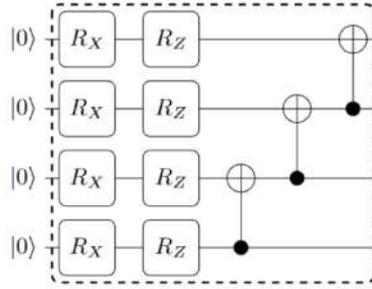


Figure 3.8: An example of variational circuit ansatz for a PQC learning model: the rotational angle of each R_x and R_z gate is a parameter of the PQC that can be tuned.

Gradients of PQC are a key requirement for the implementation of a gradient descent algorithm on a QML model. The expression (3.10) requires the evaluation of the partial derivatives of the loss function with respect to each tunable parameter θ_i : these terms can be evaluated numerically using the finite difference formula

$$\frac{\partial L}{\partial \theta_i} \approx \frac{L(\theta + \epsilon \hat{e}_i) - L(\theta - \epsilon \hat{e}_i)}{2\epsilon}$$

where ϵ is a (small) parameter and \hat{e}_i refers to the unit vector of the θ_i direction. However, often ML lead to optimization problems which presents instability with respect to numerical differentiation therefore analytical gradients are used if available. Luckily analytical gradients for PQC can be calculated exactly using the, so called, *parameter shift rule* [53, 65]: recalling that $L(\theta)$ is defined as a function of the expectation values $\langle M_k \rangle_\theta$, so, using chain rule, the derivatives of $L(\theta)$ can be expressed as a function of derivatives of $\langle M_k \rangle_\theta$. The parameter shift rule states that

$$\frac{\partial \langle M_k \rangle_\theta}{\partial \theta_i} = \frac{\langle M_k \rangle_{\theta + \frac{\pi}{2} \hat{e}_i} - \langle M_k \rangle_{\theta - \frac{\pi}{2} \hat{e}_i}}{2}$$

This means that exact gradients can be calculated performing two evaluations of the quantum circuit per parameter, even on quantum hardware.

Chapter 4

Results on b -tagging with Quantum Machine Learning

4.1 Training data-set selection

A list of jets is extracted from the data-set, with the following requirements:

- Jet transverse momentum $p_T > 20$ GeV;
- Jet pseudo-rapidity η in the range $[2.2, 4.2]$;
- Jet must be coming from a QCD event.

After the selection criteria, the data-set includes around 700 000 jets which have been split in training and testing subsets, in 60/40 proportion.

Data have been processed extracting 16 features coming from the reconstructed jets to be used as the input of the QML models. The following charged particles (and anti-particles) have been considered inside each jet: muons, pions, kaons, protons and electrons. For each type, the one reconstructed with the largest transverse momentum has been selected, considering the three following quantities:

- p_T^{rel} : transverse momentum of the particle with respect to the jet axis;
- $\Delta R = \sqrt{(\eta - \eta_{\text{jet}})^2 + (\phi - \phi_{\text{jet}})^2}$: distance on the η - ϕ plane between the particle (η, ϕ) and the jet axis $(\eta_{\text{jet}}, \phi_{\text{jet}})$;
- q : charge of the particle.

If the jet does not contain particles of a certain type, the relative features are fixed to 0. Also, the weighed jet charge Q_{jet} , defined in (2.1), is included as a global feature of the jet. Each jet carries the Monte Carlo truth about the quark that originated it (coming from the matched Monte Carlo jet), namely if it is b -quark or a \bar{b} -quark: this variable represents the target feature for the tagging algorithms and is called *Jet Label*.

Additionally, a reduced data-set has also been considered: it consists of around 100 000 jets where only the 3 features associated to the muon have been selected (if found inside the jet), plus the global jet charge. Due to its reduced dimensionality, this data-set has been used to test the behavior of QML on simpler data structures.

Each $b\bar{b}$ event in the data-set contains two jets of opposite charge associated to the same pp collision, which show particles and charge correlations due to this fact. However, a jet tagging algorithm must be able to give a prediction on a jet based only on its particle content. Therefore, for training purposes, a shuffled list of jets has been produced, cancelling any correlation.

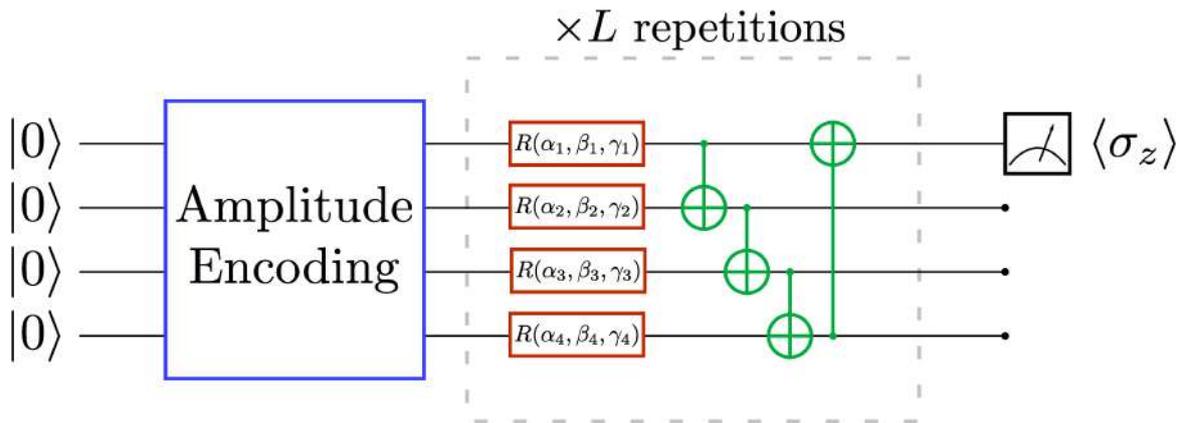


Figure 4.1: Circuitual representation of the Amplitude Encoding classifier model. In blue: amplitude encoding stage; in red: parametric rotational gates to be optimized during the training phase; in green: entangling CNOT gates.

4.2 Quantum Machine Learning models

Since this research topic is brand new, several QML models have been tested and compared in terms of tagging performance.

4.2.1 Amplitude Encoding classifier

This model consists in a PQC with a classical structure made by an encoding circuit followed by a variational circuit. The schematic representation of this model is shown in Figure 4.1. An amplitude encoder is used to embed the 16 features of the data-set into the amplitudes of a 4-qubits circuit. Follows a variable number L of, the so called, *strongly entangled layers*, which consist in parametrized rotational gates $R(\alpha_i, \beta_i, \gamma_i)$ applied to each qubit alternating with CNOT gates which entangle the qubits. The complexity of this kind of circuit can be tuned changing the number of strongly entangles layers L and, consequently, the number of parameters of the model N_{par} , that for a generic n -qubits circuit is equal to

$$N_{\text{par}} = 3 \times n \times L \quad (4.1)$$

Note that, although the dimension of the Hilbert space scales exponentially with respect to the number of qubits n , the number of parameters only scales linearly. The exponential scaling is here exploited by the amplitude encoder: this model is able to deal with a large number of features employing only a limited number of qubits.

The expectation value of the Pauli operator $\langle \sigma_z \rangle$ is measured, with $\langle \sigma_z \rangle \in [-1, +1]$. The probabilities P_b and $P_{\bar{b}}$ of being a b -jet and a \bar{b} -jet are defined as

$$P_b = \frac{1}{2}(\langle \sigma_z \rangle + 1) \quad (4.2)$$

$$P_{\bar{b}} = \frac{1}{2}(1 - \langle \sigma_z \rangle) = 1 - P_b \quad (4.3)$$

With this choice, a measurement of $\langle \sigma_z \rangle = +1$ corresponds to a 100% probability of having a b -jet, while a measurement of $\langle \sigma_z \rangle = -1$ corresponds to a 100% probability of having a \bar{b} -jet. Finally, $\langle \sigma_z \rangle = 0$ corresponds to the maximum uncertainty with equal 50% probabilities.

Within the context of this work, the number of strongly entangled layers was fixed to $L = 4$, leading to a total number of parameter of $N_{\text{par}} = 72$.

4.2.2 Angle Encoding classifier

The structure of this model differs from the Amplitude Embedding classifier in the encoding used to embed the features into the quantum state. A schematic representation of the quantum circuit is

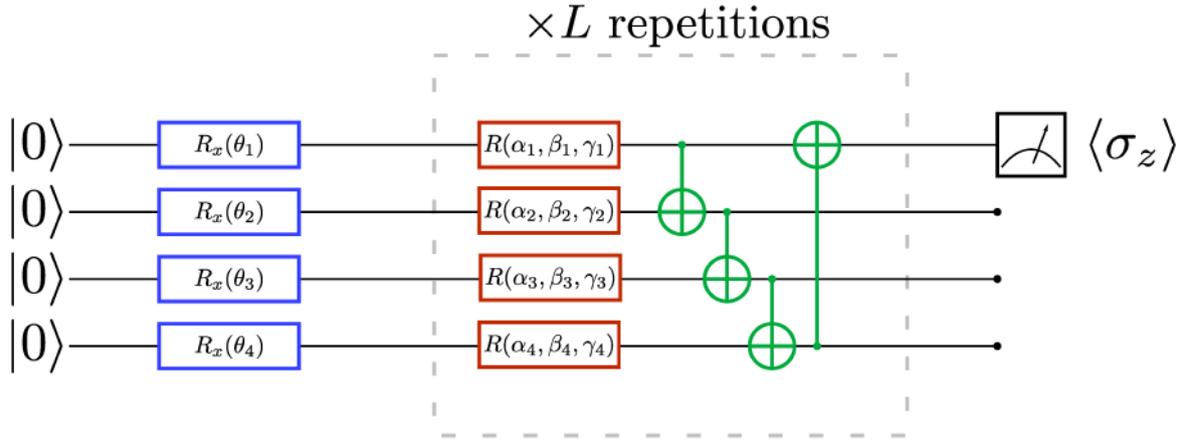


Figure 4.2: Circuitual representation of the Angle Encoding classifier model. In blue: angle encoding stage; in red: parametric rotational gates to be optimized during the training phase; in green: entangling CNOT gates.

shown in Figure 4.2: this circuit embeds N variables as parameters θ_i of N rotational gates $R_x(\theta_i)$: this circuit structure requires a one to one correspondence between qubit registers and input features. The variational stage of the circuit is identical to the Amplitude Embedding classifier, featuring a variable number L of strongly entangled layers that can be used to tune the complexity of the model: therefore, the number of parameter N_{par} to be optimized are the same as 4.1.

Again, N_{par} scales linearly with respect to N , despite of the exponential growing of the quantum state dimension, however, the angle embedding does not exploit the exponential scaling in the embedding: this means that dealing with high-dimensional data-set requires a large number of qubits, which makes this structure impractical to simulate or run on nowadays hardware. Due to this limitation, this model has tested with the reduced muon data-set, which requires only $N = 4$ qubits.

Finally, the measurement stage is identical to the Amplitude Embedding classifier: the expectation value of the Pauli σ_z operator is measured and mapped to tagging probabilities P_b and $P_{\bar{b}}$ as expressed in 4.2 and 4.3.

In this case, the number of repetitions of the variational layer L was fixed to 4, so that the model has a total number of parameters of $N_{\text{par}} = 48$.

4.2.3 Data re-uploading classifier

This model is inspired by the work of *A. Perez-Salinas et al* [58] in which they introduce the concept of a single qubit universal quantum classifier based on a circuit organized as a series of data *re-uploading* stages followed by single-qubit processing units. This structure allows, in principle, the accommodation of an arbitrary number of input dimensions and several target categories at the output with a single qubit if one is allowed to use an arbitrary number of gates. In reality, circuits with a large number of gates are impractical to simulate classically and to implement on current quantum hardware. However, the same paper suggests a generalization of the same structure on multiple qubits. A classifier based on this idea has been realized, involving the quantum circuit shown in Figure 4.3: the embedding gates (in blue) alternates with variational gates (in red) and entangling gates (in green). Features are embedded as rotational angles, three for each stage, identically copying them to each of the N qubits. After each embedding gate, the qubit state is processed by a variational rotational gate and successively entangled by CNOT gates. Loading n features using this structure requires a total of $\lceil n/3 \rceil$ embedding stages, where $\lceil \cdot \rceil$ denotes the ceiling operation. Additionally, the full structure of the circuit can be repeated L times to further increase the complexity of the model. The total number of variational parameter of data re-uploading model is

$$N_{\text{par}} = 3 \times N \times L \times \left\lceil \frac{n}{3} \right\rceil$$

In this case, the number of qubits N is not constrained to the number of input features n : in fact, n fixes the number of embedding gates $\lceil n/3 \rceil$ and, therefore, the minimum depth of the circuit. Therefore,

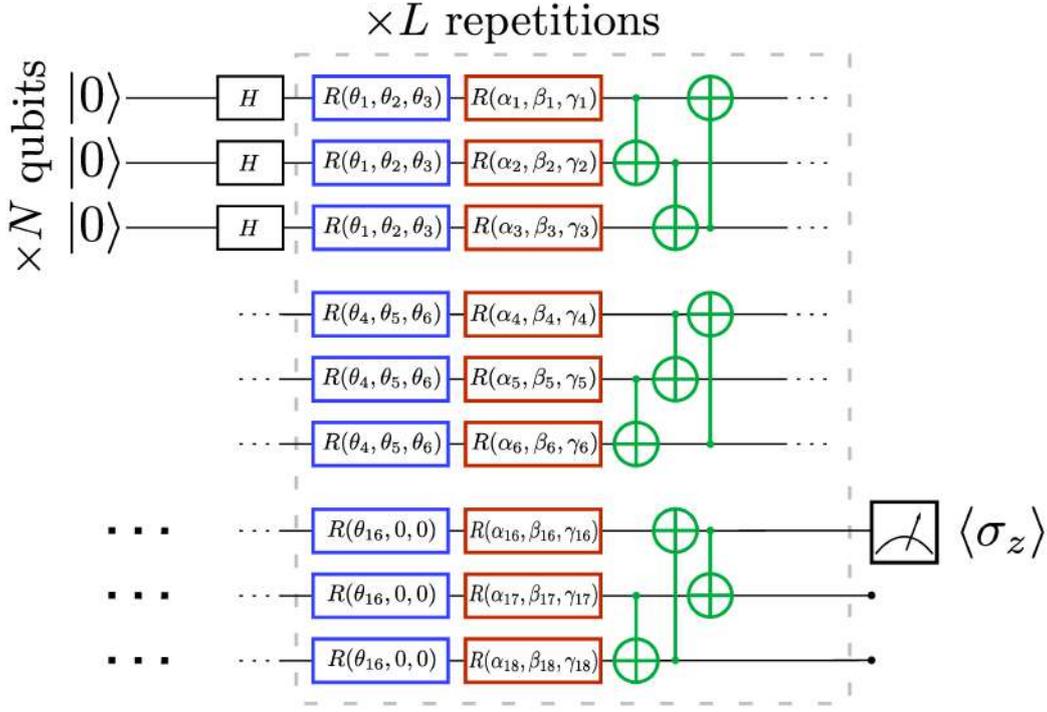


Figure 4.3: Structure of the data re-uploading circuit.

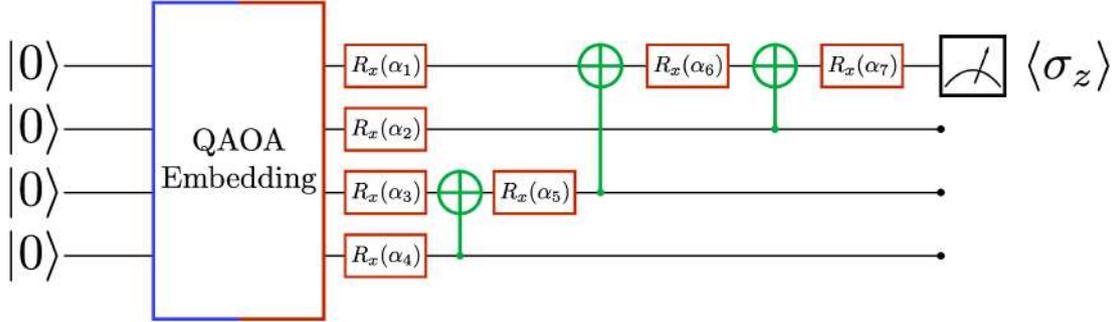


Figure 4.4: Quantum circuit of the tree-topology model.

the complexity of this model can be tuned acting both on N , that fixes the dimension of the Hilbert space, and on L , that fixes the number of trainable rotational gates. Again, the final measurement stage of the circuit is identical to the previous ones and tagging probabilities P_b and $P_{\bar{b}}$ are expressed as [4.2](#) and [4.3](#).

For our case, the number of qubits was fixed to $N = 3$, while the number of repetition layers was fixed to $L = 4$, leading to a total of $N_{\text{par}} = 216$.

4.2.4 Tree Topology classifier

This model has a structure inspired from Tree-Tensor Networks (TTNs), a class of models used for the approximation of high-dimensional functions in numerical analysis and data science [52](#). This model has been tested on a subset of the full data-set containing only the 8 most important features according to the analysis performed by *T. Felser et al.* [35](#):

- $q, p_T^{\text{rel}}, \Delta R$ of the muon
- $q, p_T^{\text{rel}}, \Delta R$ of the kaon
- q of the pion
- Q_{jet}

These features are encoded into an 8-qubits quantum circuit (for the seek of simplicity, Figure 4.4 shows a 4-qubits version of the circuit: however, the 8-qubits model is analogous), using the *Quantum Approximate Optimization Algorithm* (QAOA) encoding, introduced by *Farhi et al.* for the solution of combinatorial problems with quantum algorithms[34]. This encoding stage contains trainable parameters that are supposed to improve the quality of the embedding and must be trained together with the other parameters of the circuit. The variational stage of the circuit has a tree structure that aims to process the information coming from the 8 features, gradually reducing the number of qubits involved, until finally reaching a single qubit that gets measured identically to the previous models. This model has no hyper-parameter to adjust its complexity since the tree structure is fixed by the number of qubits involved. The total number of trainable parameter is $N_{\text{par}} = 104$.

4.2.5 Quantum Support Vector Machine classifier

The last model consists in a quantum version of the Support Vector Machine (SVM) algorithm, which is now briefly described.

SVM is ML model for binary classification of a vector \mathbf{x} into one of two classes $y = \pm 1$, given D training data points of the form $\{(\mathbf{x}_i, y_i) : \mathbf{x}_i \in \mathbb{R}^N, y_i = \pm 1\}$. The SVM finds a maximum-margin hyperplane with normal vector \mathbf{w} that separates the two classes: the margin is given by two parallel hyperplanes that are separated by a maximum possible distance $2/|\mathbf{w}|$, with no data points inside the margin; the two hyperplane are defined so that $\mathbf{w} \cdot \mathbf{x}_i + b \geq 1$ for the +1 class, and $\mathbf{w} \cdot \mathbf{x}_i + b \leq -1$ for the -1 class. This problem is equivalent to the maximization of the following function

$$L(\alpha_1, \dots, \alpha_D) = \sum_{j=1}^D y_j \alpha_j - \frac{1}{2} \sum_{j,k=1}^D \alpha_j K_{jk} \alpha_k \quad (4.4)$$

over the D coefficients α_i , under the constraints $\sum_j \alpha_j = 0$ and $y_j \alpha_j \geq 0$; K_{jk} is called *kernel matrix*. The parameters of the hyperplanes b and \mathbf{w} can be recovered from the optimized coefficients $\tilde{\alpha}_i$

$$\begin{aligned} \mathbf{w} &= \sum_{j=1}^D \tilde{\alpha}_j \mathbf{x}_j \\ b &= y_j - \mathbf{w} \cdot \mathbf{x}_j \end{aligned}$$

K_{jk} is defined by a kernel function $k(\mathbf{x}_j, \mathbf{x}_k)$, such that

$$K_{jk} = k(\mathbf{x}_j, \mathbf{x}_k)$$

In the simplest case, $k(\mathbf{x}_j, \mathbf{x}_k)$ could be a scalar product $k(\mathbf{x}_j, \mathbf{x}_k) = \mathbf{x}_j \cdot \mathbf{x}_k$, giving rise to a *linear kernel*. Non-linear kernel function can enhance the separation of the two classes. Under appropriate conditions defined by the Mercer's theorem, even non-linear kernel function can be written in terms of a *feature-map* function $\phi : \mathbb{R}^N \rightarrow \mathcal{X}$, as a scalar product in a *feature space* \mathcal{X} on which data get mapped by ϕ

$$k(\mathbf{x}_j, \mathbf{x}_k) = \langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_k) \rangle_{\mathcal{X}}$$

Fixed the kernel matrix, the maximization of the function L is a linear optimization problem for which optimal and efficient solutions are well known. Therefore, the most computational expensive task is the evaluation of the kernel: however, thanks to the feature-map method, the kernel evaluation can happen on the feature space \mathcal{X} given by ϕ , involving only scalar products: this procedure is known as the *kernel trick*. Once the SVM is been optimized, the class prediction y for a new data vector \mathbf{x} can be obtained as

$$y(\mathbf{x}) = \text{sign} \left(\sum_{j=1}^D \alpha_j k(\mathbf{x}_j, \mathbf{x}) + b \right) \quad (4.5)$$

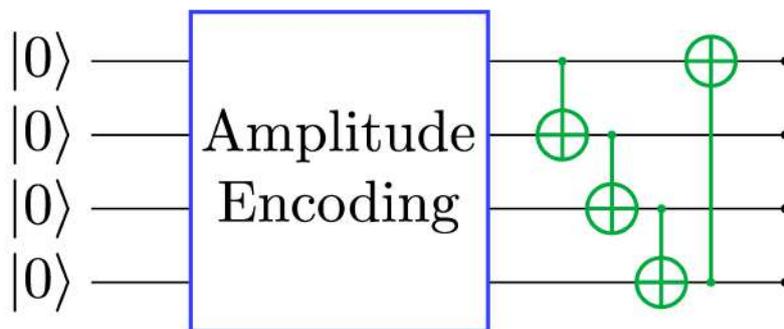


Figure 4.5: Feature map circuit of the QSVM model.

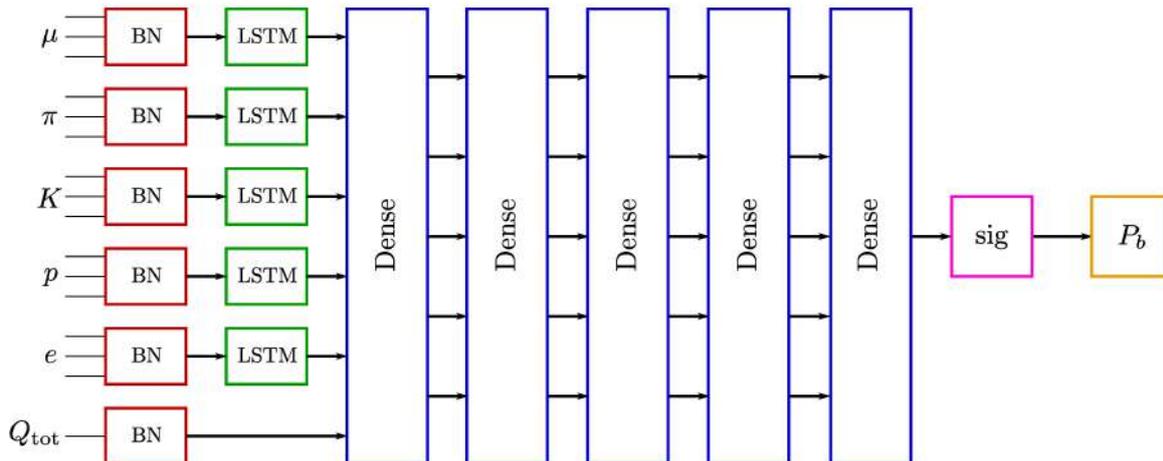


Figure 4.6: Diagrammatic representation of the classical DNN.

Quantum SVM (QSVM) [61] models uses data embeddings as feature maps and evaluates the kernel matrix using a quantum circuit. These models are based on the assumption that kernel functions defined via quantum circuits are computationally hard to evaluate on a classical computer [42]. The implemented model uses the kernel function

$$k(\mathbf{x}_j, \mathbf{x}_k) = \langle 0^n | \mathcal{U}(\mathbf{x}_j) \mathcal{U}^\dagger(\mathbf{x}_k) | 0^n \rangle$$

where \mathcal{U} is the unitary operator described in the quantum circuit in Figure 4.5: the 16 features of the data set are embedded into a 4-qubits state via an Amplitude encoding stage, then the for qubit are entangled using CNOT gates. This circuit does not contain any variational parameter, since it only maps classical data into a quantum state: the optimization procedure occurs on the α_j coefficients, appearing in 4.4.

4.3 Classical Deep Neural Network model

In addition to the QML models, a classical ML model has been developed to provide a reference. This model is a Deep Neural Network (DNN) inspired by the DeepCSV [67] and DeepJet [16] networks employed for jet identification at the CMS experiment. The DNN takes as input the same 16 variables of the data-set (a reduced version of the same network has also been trained on the muon data-set) and its structure is shown in Figure 4.6:

1. Charged particles and the global jet charge variable are initially processed in two different branches: the 3 variables associated to each charged particles are processed firstly by a Batch Normalization (BN) layer (shown in red) and then by a Long Short Term Memory (LSTM) layer (shown in green). The jet charge is only processed by a BN layer.
2. Charged particles branch and jet charge branch are then concatenated and processed by 4 layers of Dense connected hidden layers (shown in blue) of constant size. The ReLu activation function

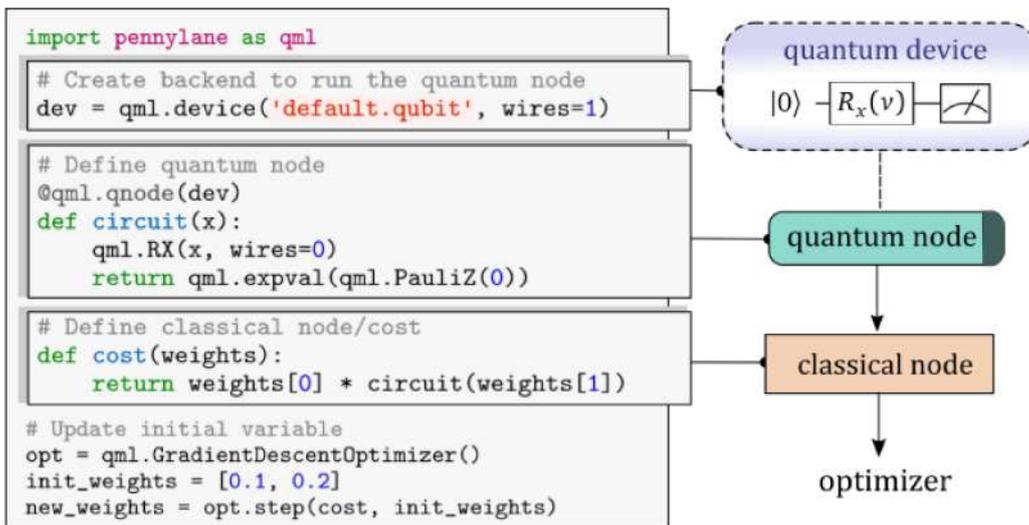


Figure 4.7: Basic example of a PennyLane program consisting of a quantum node followed by a classical node. The output of the classical node is the objective for optimization. [13]

$\text{ReLU}(x) = \max(0, x)$ has been used between each layer.

3. The output layer is a Dense layer of dimension one which is finally passed through a sigmoid function $\sigma(x) = 1/(1 + e^{-x})$ (shown in pink) to produce a value between 0 and 1 that can be interpreted as a tagging probability.

Additionally, Dropout layers are introduced in the hidden part of the network to avoid overfitting. The total numbers of parameters of the DNN acting on the muon data-set and on the full data-set are respectively $N_{\text{par}} = 181\,931$ and $N_{\text{par}} = 163\,307$.

A summary of all the analysed models is available in Table 4.1.

Model	$N_{\text{feat.}}$	N_{par}
Amplitude Embedd.	16	72
Data re-uploading	16	216
Tree-topology	8	104
QSVM	16	N.D.
DNN	16	163 307

Model	$N_{\text{feat.}}$	N_{par}
Angle Embedd.	4	48
DNN	4	181 931
Muon Tagging	1	N.D.

Table 4.1: Summary of the analyzed ML models, the number of features they process and the total number of trainable parameters: on the left, models acting on the full data-set; on the right, models acting on the reduced muon data-set.

4.4 Software implementation

The previously described models have been implemented using the PennyLane [13] Python library except for the QSVM model that was written using the IBM Qiskit [8] Python library.

4.4.1 PennyLane

PennyLane is Python3 software framework designed specifically for QML applications. PennyLane's core feature is the ability to automatically compute gradients of parametrized quantum circuits in a way that is compatible with classical optimization techniques, such as back-propagation. This allow the development of hybrid quantum-classical machine learning models that can be jointly described and trained within the same library. Moreover, a plugin system provides access to variety of quantum simulators and cloud quantum hardware devices. Figure 4.7 shows a simple workflow example: the

first line loads the PennyLane framework, while the second one defines a quantum device, choosing an execution back-end (`default.qubit`, in this case, is a simple quantum simulator coming built-in in PennyLane) and the number of qubits (called *wires*). This example consists of a simple 1-qubit circuit made of a R_x gate and a measurement the expectation value of σ_z : the circuit is described inside the Python function `circuit` decorated with the `qml.qnode(dev)` decorator which assigns the circuit to a device, creating a quantum computation node (`QNode`). From this point on, calling the function `circuit` results in the evaluation of the quantum circuit on the selected device. The `cost` function defines a classical computation node with some weight parameters, which calls the quantum circuit. The last three rows of the code performs an optimization step of the `cost` function with respect to the weight parameters, using a simple Gradient Descent optimizer.

PennyLane supports several type of devices including quantum simulator and hardware devices:

- *Simulators*:
 - `default.qubit`: a simple state-vector simulator written in Python that comes built-in with the framework
 - `default.qubit.tf`: similar to `default.qubit` but written using the TensorFlow[7] library.
 - `default.qubit.autograd`: similar to `default.qubit` but written using the Autograd library for automatic differentiation.
 - `lightning.qubit`: fast state-vector simulator written in C++
 - `Qulacs`: an high-performance quantum circuit simulator written in C/C++ supporting Graphical Processing Units (GPUs)[69].
 - IBM `Qiskit Aer`: a quantum circuit simulator developed by IBM and included in the IBM `Qiskit`[8] library.
- *Hardware*:
 - IBM Quantum Experience: an online platform to access cloud-based quantum computing services provided by IBM.
 - Amazon Braket: an Amazon Web Services (AWS) solution providing cloud-based access to different types of quantum simulators and hardware devices.
 - Rigetti Forest: cloud-based quantum computing solutions provided by Rigetti.

All the models have been deployed on simulators: during the tests, the `lightning.qubit` and the `default.qubit.tf` devices have shown to have the best performance overall. PennyLane supports the following types of automatic differentiation methods

- *Simulation-based differentiation*, such as the back-propagation or the adjoint method[46]. These methods are not compatible with hardware devices since they requires the manipulation of the quantum state vector, which is not accessible in real quantum devices.
- *Hardware-compatible differentiation*, such as the finite differences method or the parameter-shift rule. These method can be executed on both simulators and hardware devices.
- *Device gradients*, provided by some devices (such as AWS Braket devices) which perform the evaluation of the gradient on their own.

Integration with PyTorch and TensorFlow One of the most attractive feature of the PennyLane framework is its integration with PyTorch[57] and TensorFlow[7] (and its high-level ML library Keras[19]), two well known and widely used ML libraries. Both frameworks provide powerful high-level tools for developing ML models, automatically handling the evaluation of gradients for optimization. PennyLane provides the two classes `KerasLayer` and `TorchLayer` that are able to convert a `QNode` to a neural network layer that can be used inside a Keras/PyTorch model. The main advantages of this feature are the following:

- A quantum layer could be easily added to an existing neural network model, and hybrid quantum-classical models can be studied.
- The optimization of the PQC can exploit of the solid framework of tools provided by the Keras and PyTorch frameworks.
- `default.qubit.tf` can be employed within a Keras model to get an end-to-end differentiable TensorFlow model that can exploit GPU computational power for the training.

4.4.2 IBM Qiskit

IBM Qiskit is an open-source Python framework for developing quantum algorithm and testing them on quantum simulators or quantum hardware provided by the IBM Quantum Experience cloud platform. Qiskit is made up of elements that work together to allow quantum computing developing:

- *Qiskit Terra* provides the tools to create quantum circuits at the level of quantum logical gates. It also allow for device-specific optimization of each circuit.
- *Qiskit Aer* provides high-performance quantum simulators with realistic noise models, emulating the quantum hardware available on the IBM Quantum Experience platform.
- *Qiskit Ignis* provides tools for quantum hardware verification, noise characterization and error correction.
- *Qiskit Aqua* provides a list of general-purpose quantum algorithm that can be used to develop domain-specific applications. Aqua includes modules for Optimization, Machine Learning and Quantum Physics

The QSVM model has been developed using the module `QSVM` from Qiskit Aqua, providing the custom feature-map shown in Figure 4.5. The model was trained and tested using the Qiskit Aer state-vector simulator.

4.5 Training

The training part is the most computational demanding phase of the development of a PQC both on quantum hardware and quantum simulators: gradient descent methods requires a lot of steps to reach convergence and each step requires a number of evaluations of the quantum circuit per data point, which depends on the optimization algorithm and on the differentiation method: usually, this number is a polynomial of the number of parameters of the model. GPUs provide the ideal solution to this problem in classical ML tasks, allowing the optimized execution of the model in parallel for a faster evaluation of the gradient.

Unfortunately, no available quantum simulator is able to exploit GPU computational power for the training of QML models like classical ML models. However, parallelism over several CPU cores is easier to implement making use of the PyTorch framework: each of the PennyLane model was trained implementing a Distributed Data Parallel (DDP) strategy [51]. This training strategy implements parallelism at the model level. It is properly intended for distributing a training process across several different machines, each one with its own GPU, to provide a speed up; however, the same technique can be employed to achieve parallelism over multiple CPU cores on the same machine spawning multiple processes, each one running on a different core (see Figure 4.8 for a diagrammatic description). Each process evaluate the gradient on a different slice of the training data-set, working in parallel with the other processes; at the end of each optimization step, gradients are synchronized, an average gradient is computed and used to update the parameters of the model. The main downside of this strategy is memory occupation: in fact, DDP requires that each process keep its own copy of the model and the training data, putting a practical limit on the number of processes that could be spawned and on the dimension of the training data-set. It was found an optimal configuration with 8000 training events split across 20 processes which was kept fixed for each model to obtain comparable results.

The DNN has been developed under the Tensorflow framework and trained using a GPU accelerated

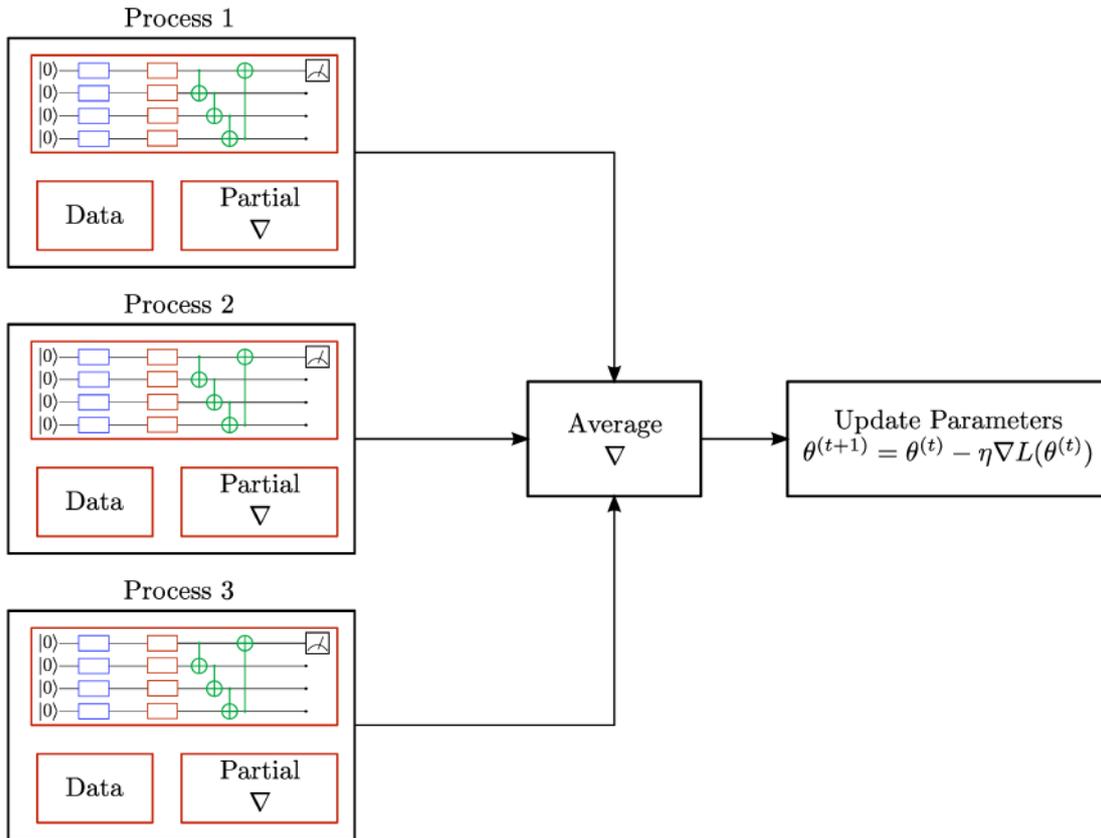


Figure 4.8: Representation of the Distributed Data Parallel training strategy on 3 processes. Each process keeps in RAM a copy of the the model and of the training data, and evaluate the partial gradient on its slice of data. Gradients are then synchronized and averaged before updating the parameters.

machine, using all the available jets.

The Mean Squared Error loss function (defined in 3.8) has been used with every quantum model, while the Cross-Entropy loss function (defined in 3.9) has been used to train the DNN. The ADAM [47] optimizer has been used with all the quantum models (except for the QSVM model for which the optimization is performed by a custom routine embedded in the QSVM Qiskit class), fixing the learning rate at 0.001. Parameters are initialized randomly at the beginning of a training process. Each model has been trained for 100 epochs and during each epoch, the optimizer process the training data-set once, updating the parameters at the end.

Figure 4.9 shows the behaviour of the loss and the accuracy of the Tree-Topology model as the training proceeds. Both are evaluated on a smaller validation data-set which is not used for training, to keep under control overfitting. The loss gets progressively smaller, converging after around 80 epochs; similarly, the accuracy gets maximized by the training procedure.

Figure 4.10 shows the action of the optimization process. The figure shows the distribution of the predictions of the data re-uploading model at the beginning and at the end of the training. b - and \bar{b} -jets are represented with light-orange and light-blue color respectively. At the beginning, parameters are random numbers and the model is not able to classify the jets. However, as the training goes by, the distribution of b - and \bar{b} -jets starts separating, getting closer to the extrema of the $[0,1]$ range. A perfect classifier would show a light-orange peak at 0 and a perfectly separated light-blue peak at 1, which means that the final quantum state (before the measurement) is $|1\rangle$ for b -jets and $|0\rangle$ for \bar{b} -jets. A realistic classifier, due to the finite number of trainable parameters, is not able to perfectly align the final state to the measurement basis vectors, thus reducing the width of the output distribution shown in Figure 4.10.

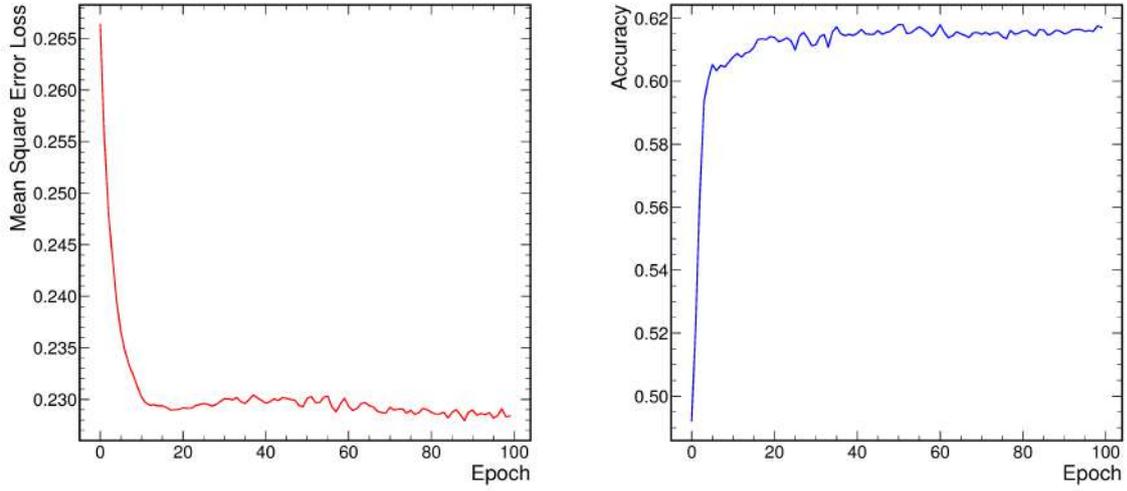


Figure 4.9: Loss and Accuracy evaluated on a small validation data-set as a function of the training epoch.

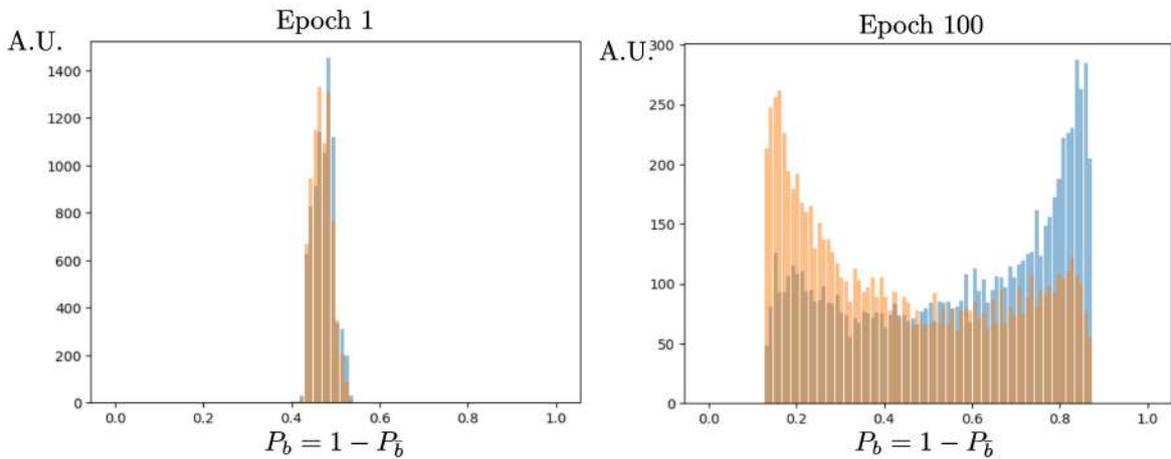


Figure 4.10: Distributions of the output of the Data re-uploading classifier at the beginning (on the left) and at the end (on the right) of the training. The \bar{b} -jets and b -jets are represented in light-blue and light-orange respectively.

4.6 Results

In this section, the performance of the tagging algorithms is presented and compared. The analysis of the performance has been done using the testing data-set (40% of the full data-set) containing jets that were not used for training. The different models are compared, from a ML point of view, in terms of the Receiver Operating Characteristic (ROC) curve[30], evaluating also the Area Under the Curve (AOC). In addition the performance is studied from the Physics point of view by comparing the distributions of the mis-tag ω and the tagging power ϵ_{tag} bins of the jet p_T .

4.6.1 Muon events

Firstly, the performance on the reduced muon data-set is presented. Figure 4.11 shows the ROC curves of the three algorithms which suggest that both the ML algorithms perform better than the Muon Tagging, while showing similar performance between them. The same is suggested by the AUC, which are listed below:

Model	AUC
Angle Embedding	0.75
DNN	0.75
Muon Tagging	0.60

A more quantitative comparison emerges from the Figures 4.12 and 4.13, which show the mis-tag and the tagging power of the models evaluated in bins of jet transverse momentum p_T in the region $[20, 100]$ GeV/ c . The tagging efficiency ϵ_{eff} required to evaluate the tagging power has been calculated, in each jet p_T bin, as the fraction of jets where a muon is reconstructed. Both the figures of merit show that the ML methods outperform the simple Muon Tagging algorithm: as expected, this analysis suggests that they are able to extract useful information from the ΔR , p_T^{rel} of the muon and the jet charge Q_{tot} , with respect to the muon tagging, that only uses the muon charge to infer the jet charge. Instead, the Angle Embedding model and the Muon DNN show equal performance within the margin of error on the mis-tag and on the tagging power.

Moreover, this analysis shows that, despite of the much less trainable parameters (a 10^4 factor), the QML Angle embedding model performs just as well as the classical DNN model, within the margin of error. This has been observed in a low-dimension data-set containing only 4 features.

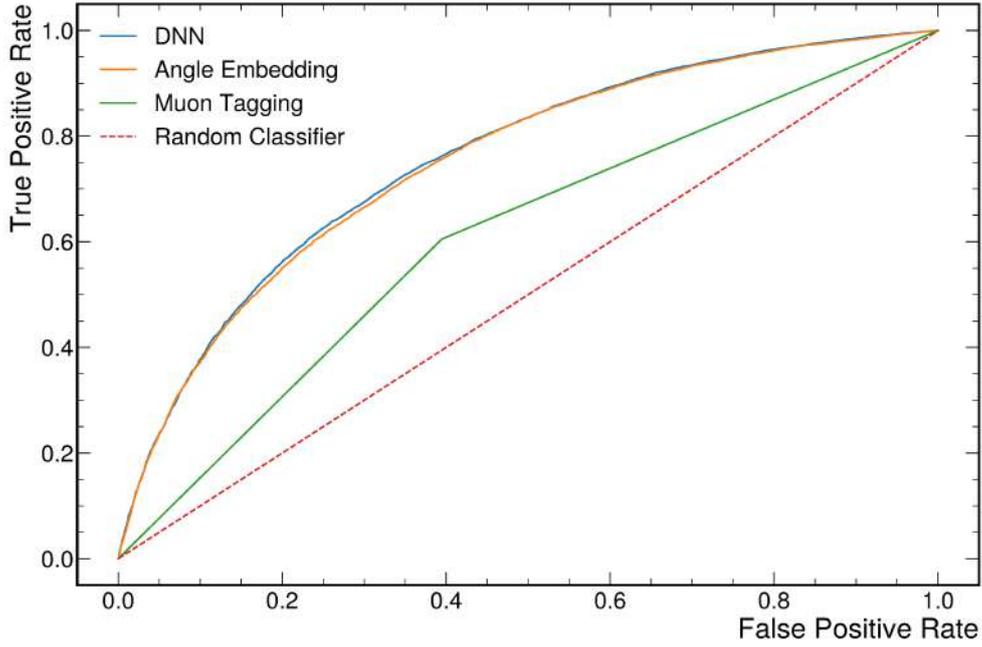


Figure 4.11: **Muon data-set**: ROC curve of the three algorithms. The dashed line represents the worst case of a random classifier.

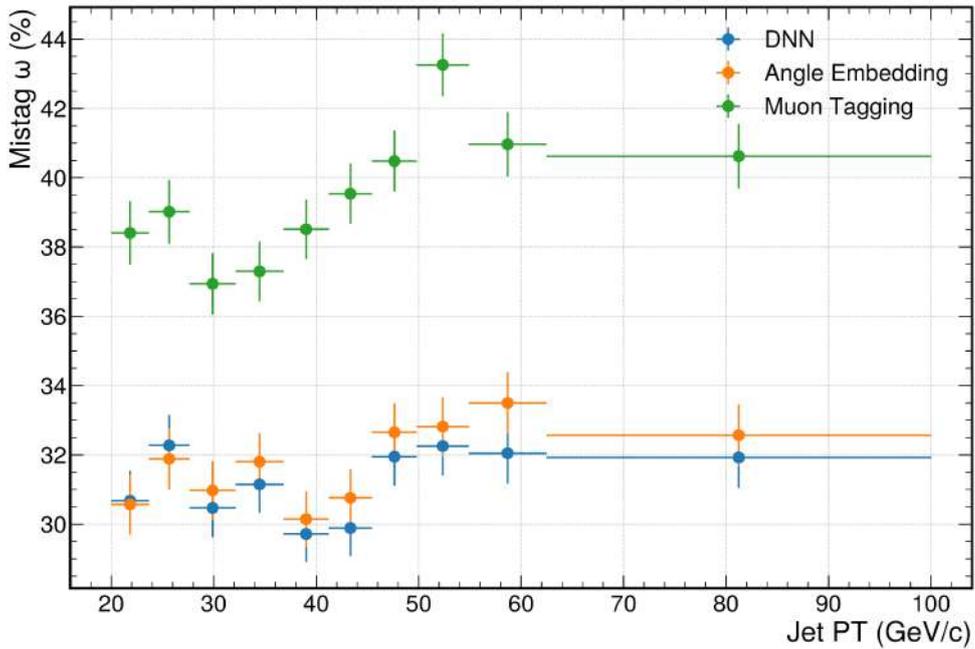


Figure 4.12: **Muon data-set**: comparison of the mis-tag ω in bins of p_T of the jets for the three tagging algorithms considered.

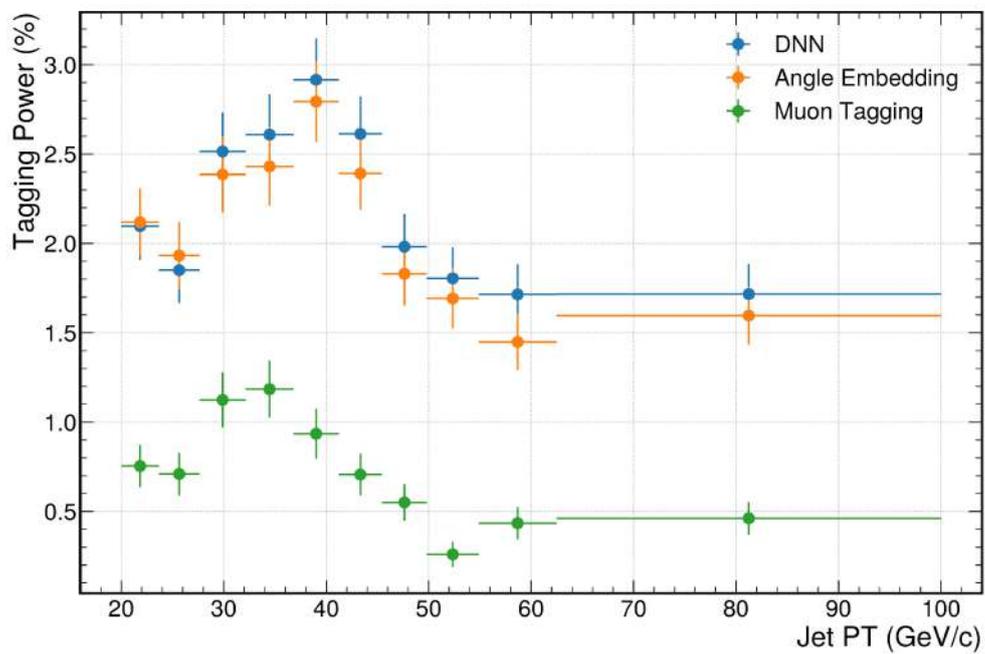


Figure 4.13: **Muon data-set**: comparison of the tagging power ϵ_{tag} in bins of p_T of the jets for the three tagging algorithms considered.

4.6.2 Full data-set

The five tagging algorithms performance is evaluated on the full data-set: Figure 4.14 shows the ROC curves while the AUCs are listed below:

Model	AUC
Amplitude Encoding	0.67
Data Re-Uploading	0.65
Tree-Topology (8 features)	0.66
QSVM	0.64
DNN	0.69

ROCs and AUCs suggest that all the quantum algorithm perform similarly, while the DNN performs slightly better. Figures 4.15 and 4.16 allow for a quantitative comparison showing the mis-tag and the tagging power in bins of jet transverse momentum p_T in the region $[20, 100]$ GeV/ c . In this case, every algorithm is able to produce a jet charge prediction for every jet, so the tagging efficiency is $\epsilon_{\text{eff}} = 100\%$: as a consequence, the tagging power is directly related the the mis-tag of the algorithm.

The following considerations can be made:

- Among all the quantum models, the Amplitude Encoding model shows the best performance in terms of the mis-tag and the tagging power. The Tree-Topology model shows similar performance, being compatible within the uncertainty with the Amplitude Encoding model in most of the jet p_T bins.
- The QSVM and Data Re-Uploading models show the worst performance in terms of mis-tag and tagging power, among all the quantum algorithms, performing consistently worse than the other two models in every jet p_T bin. However, these algorithms almost double the tagging power provided by the Angle Embedding and the DNN on the muon data-set. This means that both the algorithms are still able to exploit the additional information from the complete 16-variables data-set, compared to the reduced 4-variables muon data-set.
- The DNN outperforms all the quantum models, showing an advantage of around 1% on the tagging power with respect to the Amplitude Encoding model. This result is expected since:
 - The DNN has much more trainable parameters with respect to the other models (around 100 000 parameters versus 72 parameters of the Amplitude Encoding model);
 - The DNN has been trained using the full training data-set, containing 400 000 events, while the quantum model were constrained to 8 000 jets by computational limits of the quantum simulators.

Additional considerations on this comparisons are provided in Chapter 6.

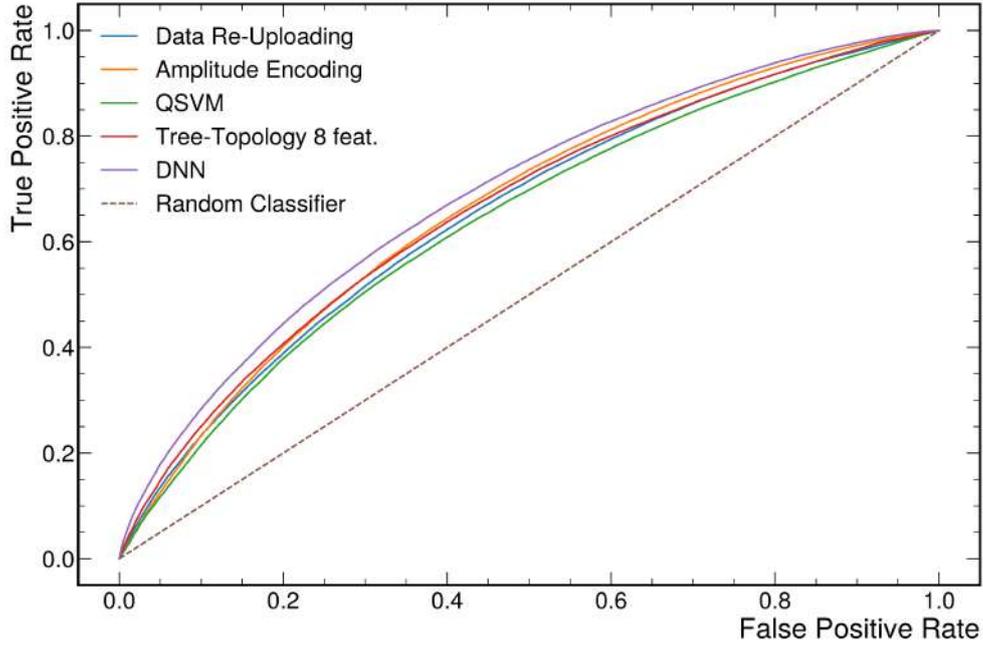


Figure 4.14: **Full data-set:** ROC curve of the five algorithms. The dashed line represents, again, the random classifier.

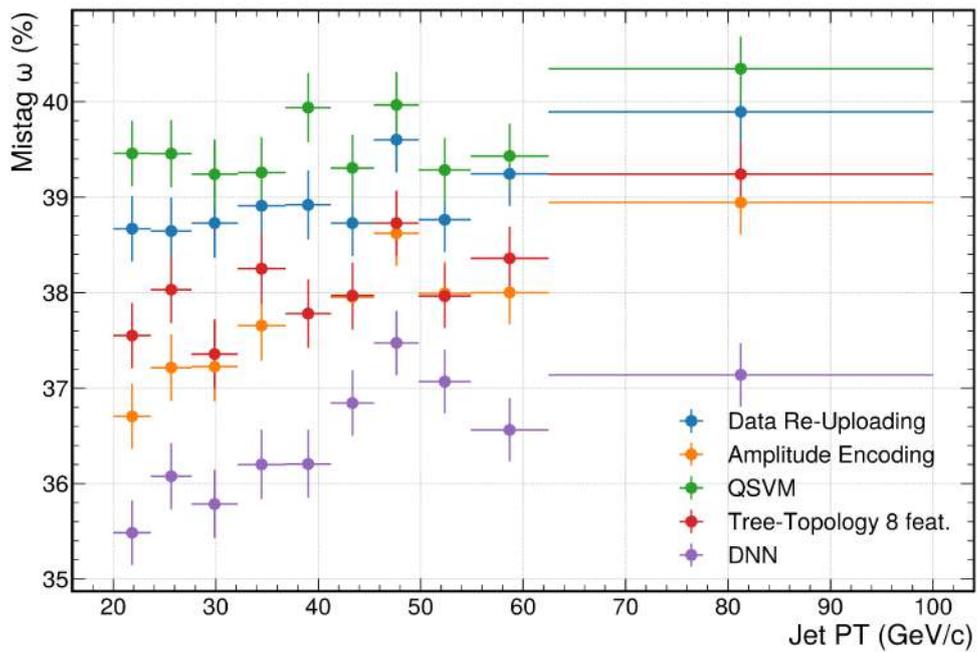


Figure 4.15: **Full data-set:** comparison of the mis-tag ω in bins of p_T of the jets for the five tagging algorithms considered.

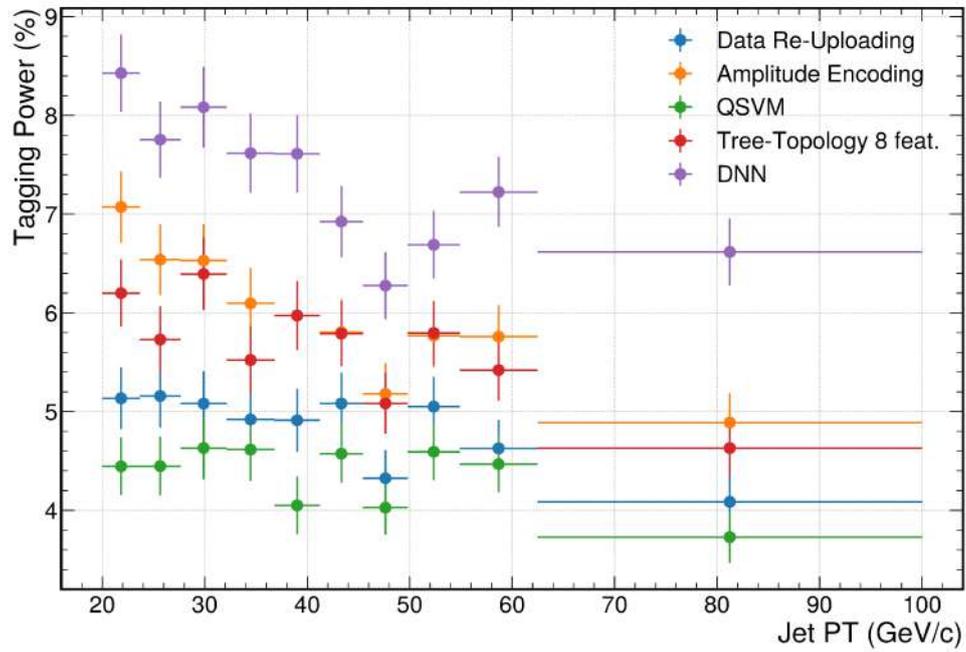


Figure 4.16: **Full data-set**: comparison of the tagging power ϵ_{tag} in bins of p_T of the jets for the five tagging algorithms considered.

Chapter 5

Asymmetry measurement and estimation of its uncertainty

In this chapter, the models previously described and analyzed, are used to perform an estimation of the $b\bar{b}$ Forward-Central asymmetry on the Monte Carlo simulated data-set in order to evaluate the precision achievable with the use of these new methods. The main purpose of this analysis is to compare the results obtained with the tagging algorithms in terms of the expected statistical uncertainty that they reach when the full data-set of 6 fb^{-1} of integrated luminosity of the Run2 is used.

5.1 Data selection

The events described in Chapter 2 Section 2.7 are re-weighted by the weight w^i

$$w^i = \frac{\mathcal{L} \cdot \sigma^i}{N_{\text{gen}}^i}$$

where $\mathcal{L} = 6\text{ fb}^{-1}$ is the Run2 integrated luminosity, N_{gen}^i is the number of generated events of each files and σ^i is the production cross section of a $b\bar{b}$ pair, within the LHCb geometrical acceptance ($2 < \eta < 5$), via the processes and the \hat{p} range relative to the i -th file. The cross sections are computed using PYTHIA8. The expected total number of $b\bar{b}$ pairs produced during Run2 can be written as

$$N_{\text{tot}}^{b\bar{b},\text{Run2}} = \sum_i N_{\text{gen}}^i w^i \approx 5.4 \times 10^{10} \text{ events}$$

where the sum extends over all the files of the data-set.

The $b\bar{b}$ events, reconstructed with the algorithm described in Sections 2.4 and 2.5, are selected applying the following criteria on the transverse momentum of the jets p_T^{jet} and pseudo-rapidity η^{jet} , and on the angular separation between the b -jets and the \bar{b} -jets $\Delta\phi_{jj}$:

- $p_T^{\text{jet}} > 20\text{ GeV}/c$
- $2.2 < \eta^{\text{jet}} < 4.2$
- $\Delta\phi_{jj} > 2.6$

The requirements on the transverse momentum and the pseudo-rapidity ensure that the jets are reconstructed, in a phase space region where reconstruction and identification efficiencies are uniform with respect to p_T^{jet} and η^{jet} [26]; moreover, the requirement on η^{jet} ensures that the entire jet is inside the fully instrumented LHCb acceptance. The requirement on the angular separation between the two jets ensures that the two b -jets are well separated[38]. Figure 5.1 shows the distributions of the most relevant kinematic variables after the data selection process:

- (a): jet transverse momentum p_T^{jet}

- (b): jet pseudo-rapidity η^{jet}
- (c): jet rapidity y
- (d): angular separation between the two jets $\Delta\phi_{jj}$
- (e): dijet invariant mass $M_{jj} = \sqrt{(E_1 + E_2)^2 - (\mathbf{p}_1 + \mathbf{p}_2)^2}$, where (E_1, \mathbf{p}_1) and (E_2, \mathbf{p}_2) are the four-momenta of the two reconstructed jets.

Figure 5.1(f) shows also the distribution of M_{jj} on the Z^0/γ subset of events, which has a peak on the Z^0 resonance.

5.2 Asymmetry measurement and dilution correction

In order to measure the Forward-Central $b\bar{b}$ production asymmetry, defined in 1.4.3, one needs to count the number of events where the quantity $\Delta|y| = |y_b| - |y_{\bar{b}}|$ is positive and negative. The sign of $\Delta|y|$ is determined by the charge of the jet measured by the b -tagging algorithms. The fact that the mis-tag of the tagger is not 0 causes a dilution of the asymmetry. Let $A_{\text{FC}}^{b\bar{b}}$ be the true asymmetry, namely the one measured with a perfect tagger, and $A_{\text{FC, Raw}}^{b\bar{b}}$ be the asymmetry measured with a tagger with mis-tag ω , called *raw asymmetry*

$$A_{\text{FC}}^{b\bar{b}} = \frac{N^+ - N^-}{N^+ + N^-} \quad A_{\text{FC, Raw}}^{b\bar{b}} = \frac{\tilde{N}^+ - \tilde{N}^-}{\tilde{N}^+ + \tilde{N}^-} \quad (5.1)$$

where $N^+ = N(\Delta|y| > 0)$ and $N^- = N(\Delta|y| < 0)$ for simplicity, while the tilde ones are the same quantities but measured with the non-ideal tagger. The two denominators represent the total number of events that does not depend on the mis-tag, so $N^+ + N^- = \tilde{N}^+ + \tilde{N}^- = N_{\text{tot}}$. Since the probability of mis-identification of the charge of the jet is ω , \tilde{N}^+ and \tilde{N}^- can be rewritten in terms of the true quantity N^+ and N^-

$$\tilde{N}^+ = (1 - \omega)N^+ + \omega N^- \quad \tilde{N}^- = (1 - \omega)N^- + \omega N^+ \quad (5.2)$$

Therefore, the raw asymmetry can be written as

$$A_{\text{FC, Raw}}^{b\bar{b}} = (1 - 2\omega) \frac{N^+ - N^-}{N_{\text{tot}}} = (1 - 2\omega) A_{\text{FC}}^{b\bar{b}} = D A_{\text{FC}}^{b\bar{b}} \quad (5.3)$$

The quantity D is called *dilution* and quantifies the discrimination power of the tagging algorithm, ranging from 0% for a random tagger ($\omega = 50\%$) to 100% for a perfect tagger ($\omega = 0\%$).

Therefore, the true asymmetry $A_{\text{FC}}^{b\bar{b}}$ can be obtained dividing the raw asymmetry $A_{\text{FC, Raw}}^{b\bar{b}}$ by the dilution D of the tagger (*dilution correction*)

$$A_{\text{FC}}^{b\bar{b}} = \frac{A_{\text{FC, Raw}}^{b\bar{b}}}{D}$$

In this work, the asymmetry has been computed in six bins of dijet invariant mass: [60, 75], [75, 85], [85, 95], [95, 105], [105, 150] and [150, 200] GeV/ c^2 . This range has been chosen to include most of the Z^0/γ events, with a finer binning near the Z^0 resonance, where they are mostly concentrated (see Figure 5.1f).

5.2.1 Uncertainty on the asymmetry

Two types of uncertainties are associated to the asymmetry measurement:

- The statistical error, given by:

$$\sigma_{\text{Run2}}[A_{\text{FC}}^{b\bar{b}}] = \frac{1}{D} \sqrt{\frac{1 - A_{\text{FC}}^{b\bar{b}2}}{N}} \quad (5.4)$$

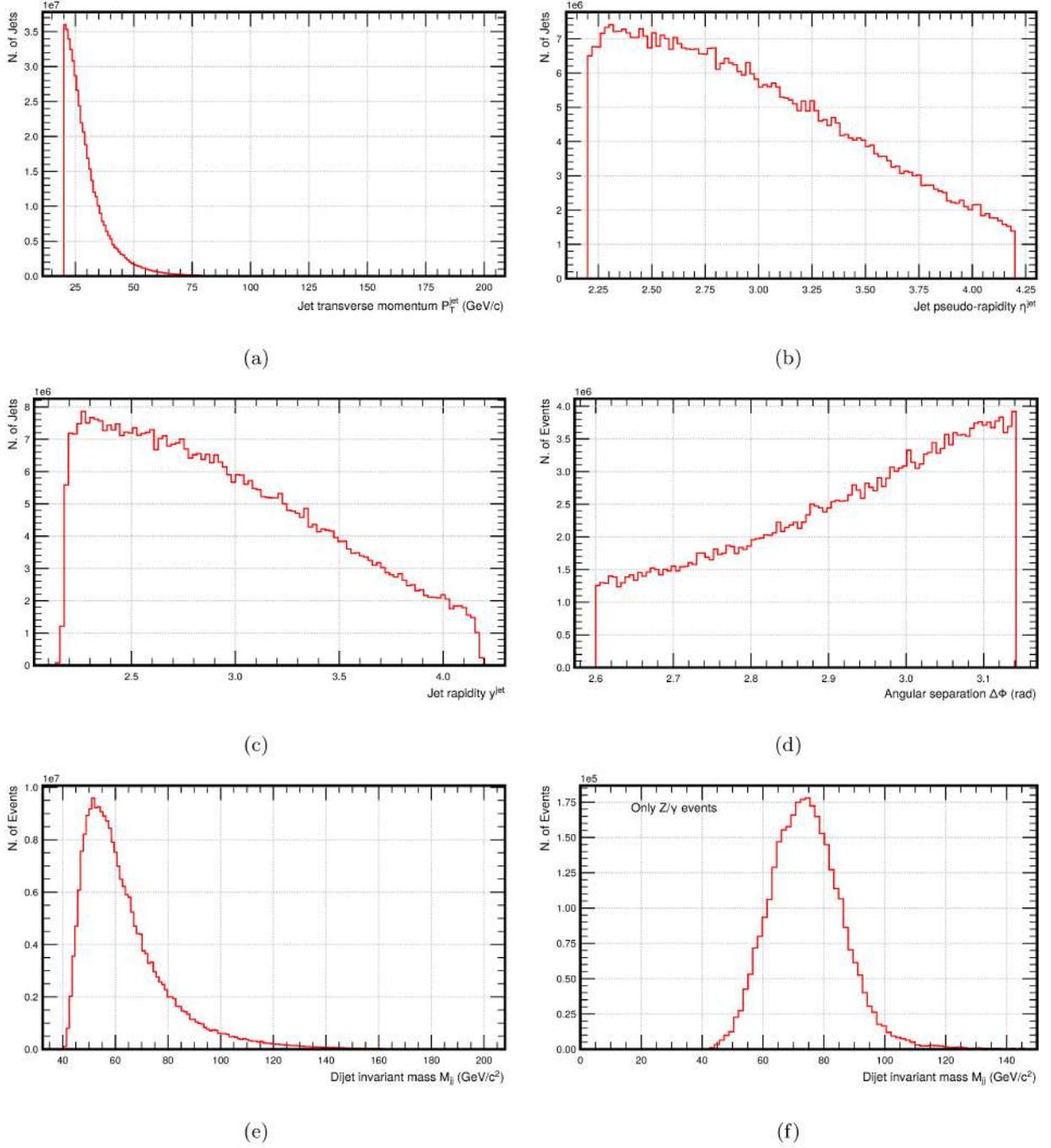


Figure 5.1: Control plots after the selection requirements: on the y-axis, the number of jets/events expected on 6 fb^{-1} of data are reported. The variables shown are: (a) the jet transverse momentum p_T^{jet} , (b) the jet pseudo-rapidity η^{jet} , (c) the jet rapidity y , (d) the angular separation $\Delta\phi_{jj}$, (e) the dijet invariant mass M_{jj} . As an additional check, M_{jj} is computed also for the Z^0/γ subset of events, showing the resonance peak of the Z^0 (f).

where N is the total number of Run2 $b\bar{b}$ events used for the measurement, and D is the dilution of the tagging algorithm. (5.4) shows how the tagging performance is strongly related to the uncertainty on the measurement of the asymmetry: a bad-performing tagging algorithm with a small dilution D boosts by a factor $1/D$ the statistical uncertainty given by the number of available events $\sqrt{(1 - A_{\text{FC}}^{b\bar{b}})^2}/N$.

- The systematic error, which is dominated by the tagging performance showed in Figures 5.4 and 5.7. The discussion of this error goes beyond the scope of this thesis.

5.2.2 Asymmetry in the Monte Carlo simulated data

Since we are working with simulated events generated at LO, QCD events do not have any asymmetrical contribution in the $b\bar{b}$ production and therefore the difference $(N^+ - N^-)_{\text{QCD}}$ cancels out up to statistical fluctuations, as shown in Figure 5.2a. In order to evaluate the precision on the asymmetry it is assumed that the QCD contribution is zero and it is known with infinite precision. Only the Z^0/γ events contribute to the asymmetry as shown in Figure 5.2b. Figure 5.2c shows the $\Delta|y|$ distributions of Z^0/γ events (in orange), which is clearly asymmetric w.r.t 0, and of QCD events (in light blue), which is symmetric.

Figure 5.2d shows the asymmetry evaluated on the full set of events, keeping only Z^0/γ events in the difference $(N^+ - N^-)$ in the asymmetry definition. The numerical values obtained from MC truth in each bin of dijet invariant mass are showed in Table 5.1 and are used as reference values to check for possible biases introduced by the tagging algorithms.

M_{jj} range [GeV/c ²]	$A_{\text{FC}}^{b\bar{b}} \pm \sigma_{\text{MC}}[A_{\text{FC}}^{b\bar{b}}]$
[60, 75]	$(0.46 \pm 0.03)\%$
[75, 85]	$(1.07 \pm 0.08)\%$
[85, 95]	$(0.87 \pm 0.10)\%$
[95, 105]	$(0.42 \pm 0.11)\%$
[105, 150]	$(0.11 \pm 0.05)\%$
[150, 200]	$(0.08 \pm 0.13)\%$

Table 5.1: Asymmetry values in each M_{jj} bin, computed on the full data-set, using the Monte Carlo truth. The error reported is the Monte Carlo statistical error.

5.3 Comparison among methods

In this section the results of the evaluation of the accuracy on the asymmetry using the proposed tagging methods are presented. The analysis has been performed on

1. the subset of events, where at least one of the two jets has a muon inside (called *muon data-set*). This data-set has been analyzed using the Angle Embedding model, the DNN model and the classical Muon Tagging algorithm;
2. the complete data-set where the Amplitude Embedding model and the DNN model have been tested.

5.3.1 Results on the muon data-set

Selecting events with at least a muon inside a jet could, in principle, introduce biases in the distribution of the kinematic variables shown in Figure 5.1. To evaluate each bias, the same distributions shown in Figure 5.1 are done for the muon data-set and compared to the complete ones, as shown in Figure 5.3, and these do not show off any relevant bias. Figure 5.4a, shows the tagging efficiency of the tagging algorithms, as a function of the dijet invariant mass, defined as the number of events with

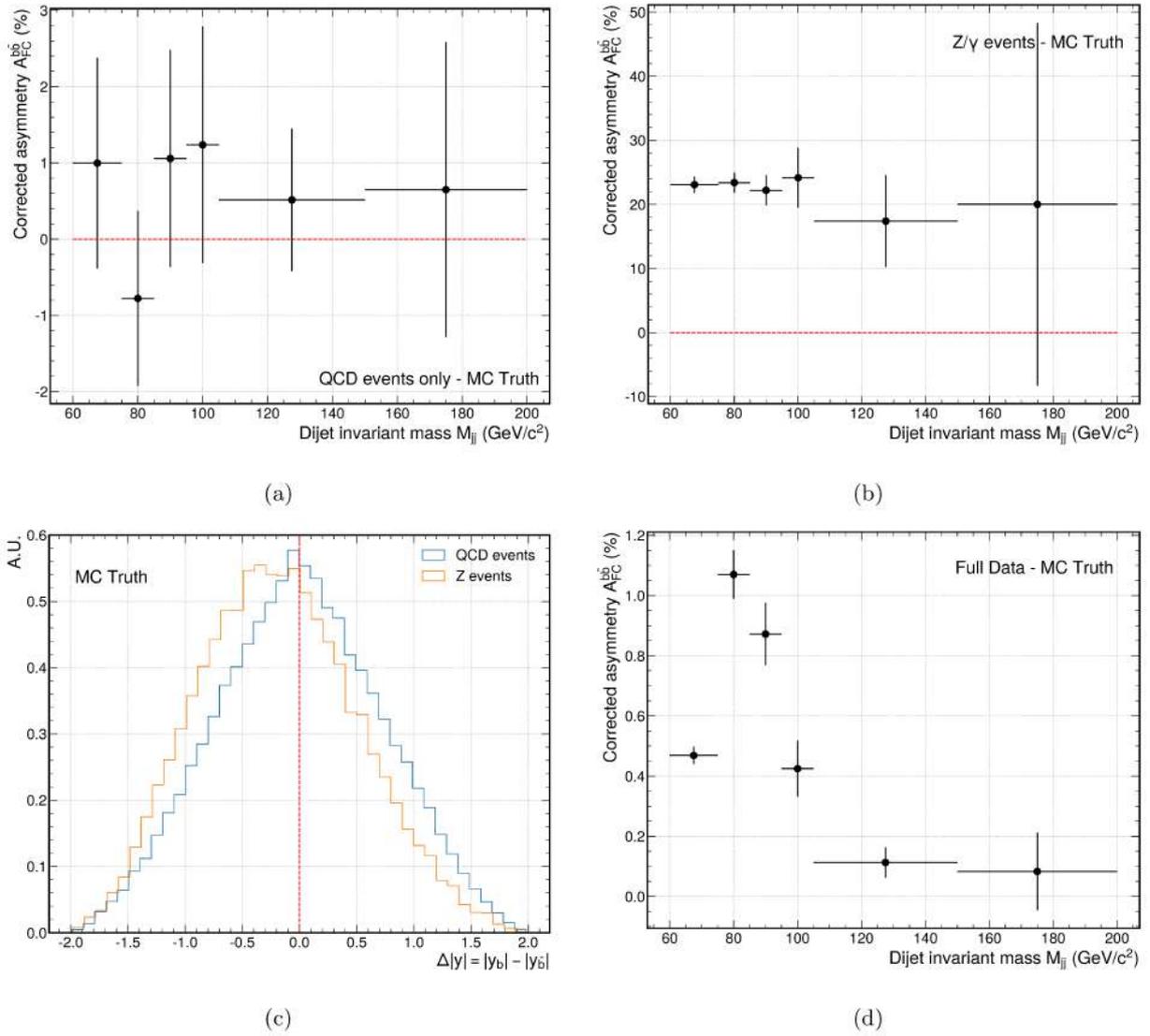


Figure 5.2: (a), (b) and (d) show the asymmetry on QCD events only, on Z^0/γ events only and on the complete data-set respectively, using the MC truth. Error bars represent the Monte Carlo statistical error. In (c) it is shown the distributions of $\Delta|y|$, for QCD and Z^0/γ events. The two distributions are normalized separately.

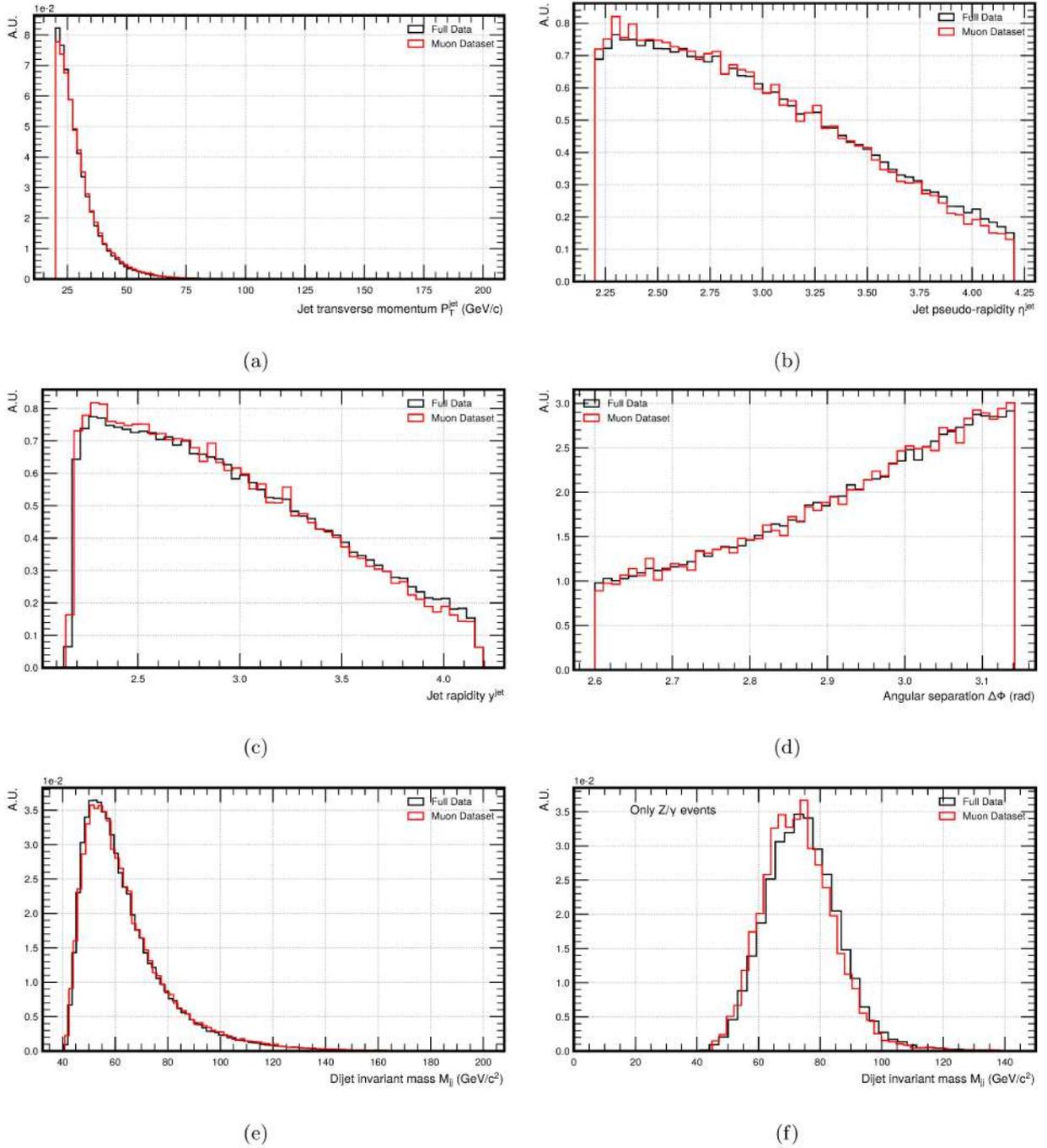


Figure 5.3: Control plots comparing the distribution on the complete data-set (black) with the Muon data-set (red), after the selection requirements. In each plot, the two distributions are normalized separately to 1. The variables shown are: (a) the jet transverse momentum p_T^{jet} , (b) the jet pseudo-rapidity η^{jet} , (c) the jet rapidity y , (d) the angular separation $\Delta\phi_{b\bar{b}}$, (e) the event invariant mass M_{jj} and (f) the M_{jj} computed for the Z^0/γ subset of events.

at least a jet containing a muon divided by the total number of events.

For the asymmetry measurement, the Muon Tagging algorithm has been implemented with the following rules:

- If only one jet of the event (say **Jet1**) contains a muon and the other (say **Jet2**) not, then **Jet1** is tagged with the charge of the muon and to **Jet2** is given the opposite charge.
- If both jets contain a muon, then the muon with the highest transverse momentum relative to the jet axis p_T^{rel} is chosen (say the muon in **Jet1**) and used to tag **Jet1**, while **Jet2** is tagged with opposite charge. This choice maximize the probability of selecting the correct muon coming from a semi-leptonic decay of the b -hadron.

The Angle Embedding model and the DNN have been handled in a slightly different way. ML models produce two tagging probabilities P_b and $P_{\bar{b}}$ that are used to tag events:

- If the event has only one muon, then the jet containing the muon is tagged with the highest probability charge, the other jet is tagged with the opposite charge.
- If both jets contain a muon, then the jet associated with the largest tagging probability is chosen and tagged accordingly, while the other is tagged with the opposite charge.

The mis-tag ω and the tagging power ϵ_{tag} of the three tagging algorithms are evaluated per dijet pairs, in bins of dijet invariant mass M_{jj} . The results are shown in Figure 5.4b and 5.4c, and exhibit similar behaviors compared to the results obtained on the per-jet analysis presented in Chapter 4 (Figure 4.12, 4.13). This is expected since most of the events contains only one muon, therefore the mis-tag probability on the dijet pair coincides with the mis-tag probability on the single jet.

The raw asymmetry is computed as a function of the dijet invariant mass, using the three tagging algorithms; the results shown in Figure 5.5 (on the left) are compared with the Monte Carlo truth. As expected, in most of the bins, the raw asymmetry values differs significantly from the Monte Carlo truth, since no dilution correction has been applied yet. Note that the Monte Carlo statistical uncertainties on the raw asymmetry evaluated with the three different tagging methods are roughly comparable, since they depend only on the available statistics on the Muon data-set.

On the right of Figure 5.5, the dilution-corrected asymmetries are shown which are now compatible with Monte Carlo truth within the Monte Carlo statistical error. This demonstrates that the tagging methods or the selection criteria do not create any bias on the measurement. Note that, Muon Tagging algorithm (in green) shows significantly larger uncertainties with respect to the Angle Embedding quantum method (in orange) and the classical DNN (in blue); since all the three methods have the same tagging efficiency ϵ_{eff} , this is a consequence of the large mis-tag ω rate of the Muon Tagging with respect to the other two tagging methods, which results in a larger tagging power ϵ_{tag} for the ML methods, as clearly shown in Figure 5.4.

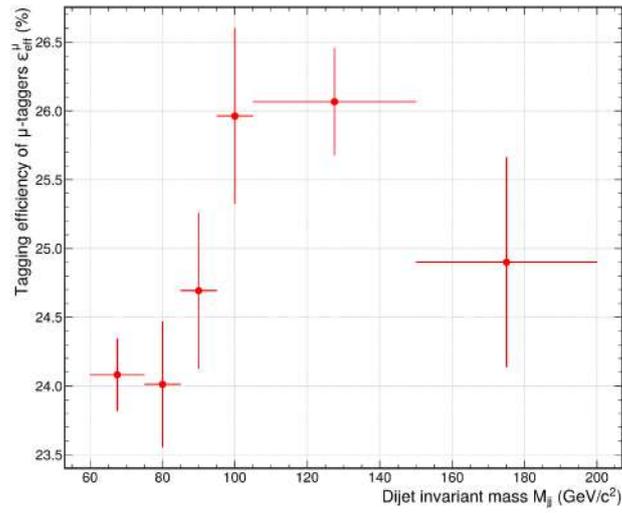
Figure 5.6 shows the expected Run2 statistical uncertainties achievable by each one of the three tagging methods. The bar plot clearly shows that both the ML methods outperform the Muon Tagging algorithm, while showing similar performance between them.

5.3.2 Results on the full data-set

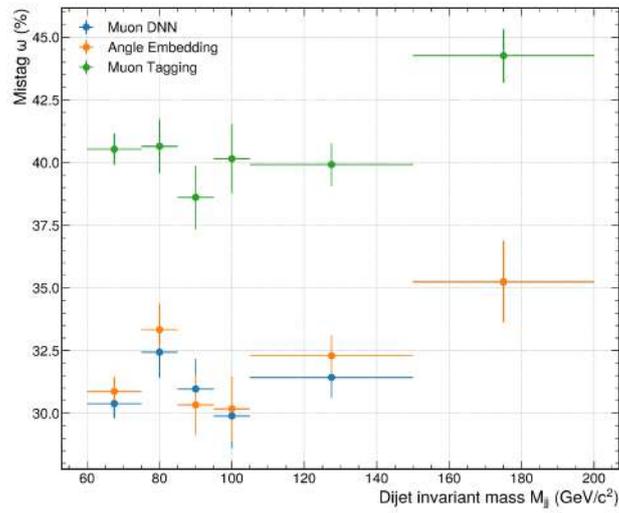
The complete data-set is analyzed by using the Amplitude Embedding, the best QML model among the tested ones, and the classical DNN model.

The algorithms produce independent tag prediction probabilities for both jets of each event; this implies that the tagging efficiency is $\epsilon_{\text{eff}} = 100\%$, and therefore the tagging powers ϵ_{tag} only depend on the mis-tag rates. The dijet pair of each event is unambiguously tagged by choosing the jet associated with the largest tagging probability as b - or \bar{b} -jet and assigning to the other the opposite charge.

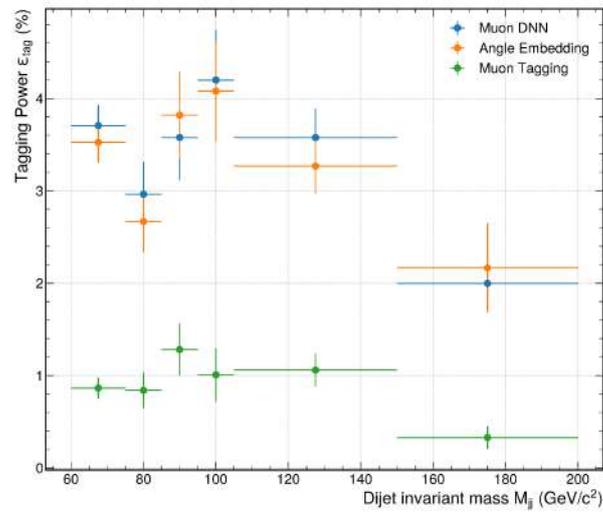
The tagging performance of the two methods has been evaluated and is shown in Figure 5.7. The DNN



(a)



(b)



(c)

Figure 5.4: Tagging performance on the Muon Data-set as function of the dijet invariant mass M_{jj} . (a) Tagging efficiency ϵ_{eff} , that is the same for the three algorithm. (b) Mis-tag ω . (c) Tagging power ϵ_{tag} .

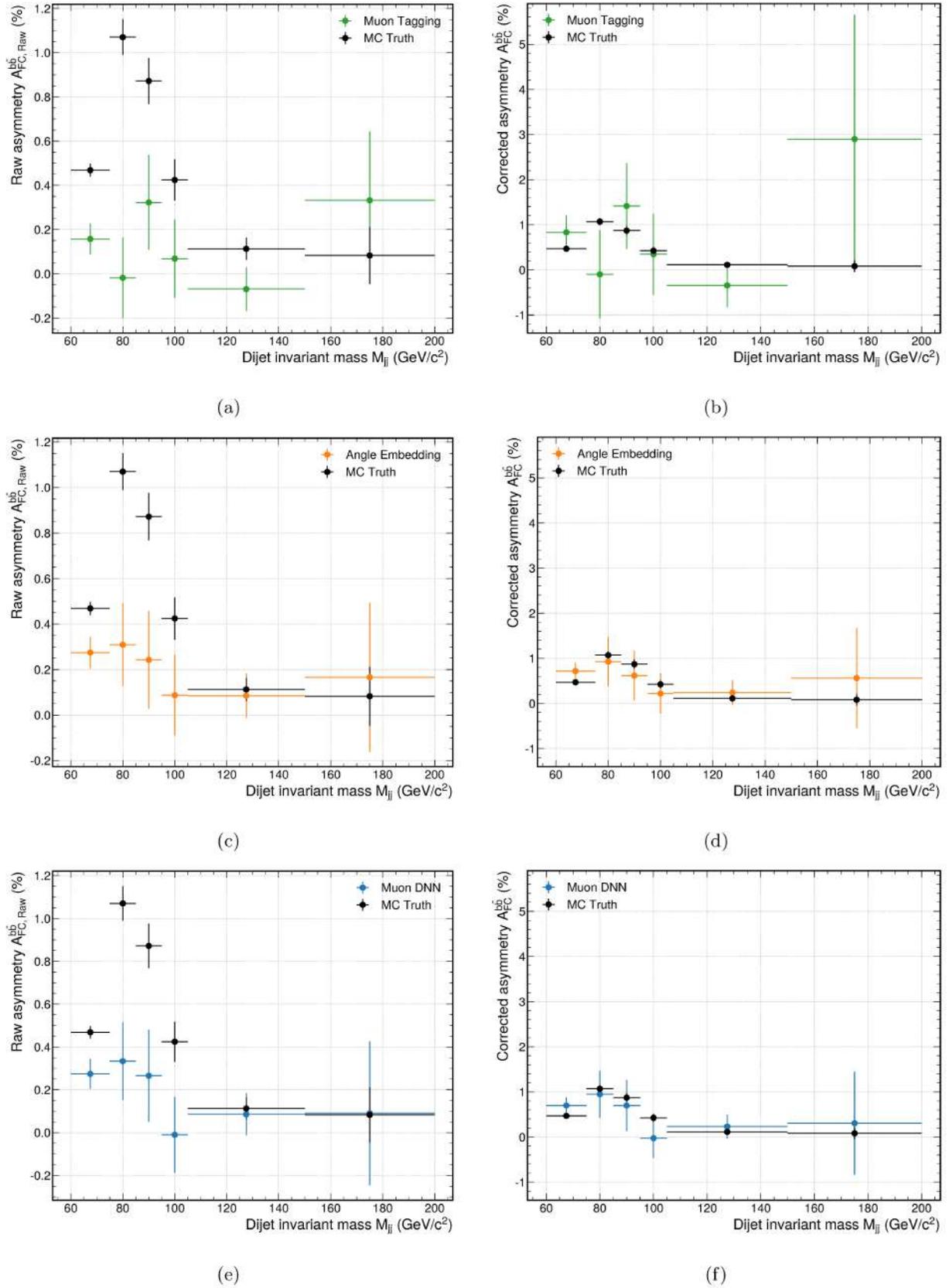


Figure 5.5: Raw asymmetries (on the left) and Corrected asymmetries (on the right), computed with the three tagging algorithm on the muon data-set.

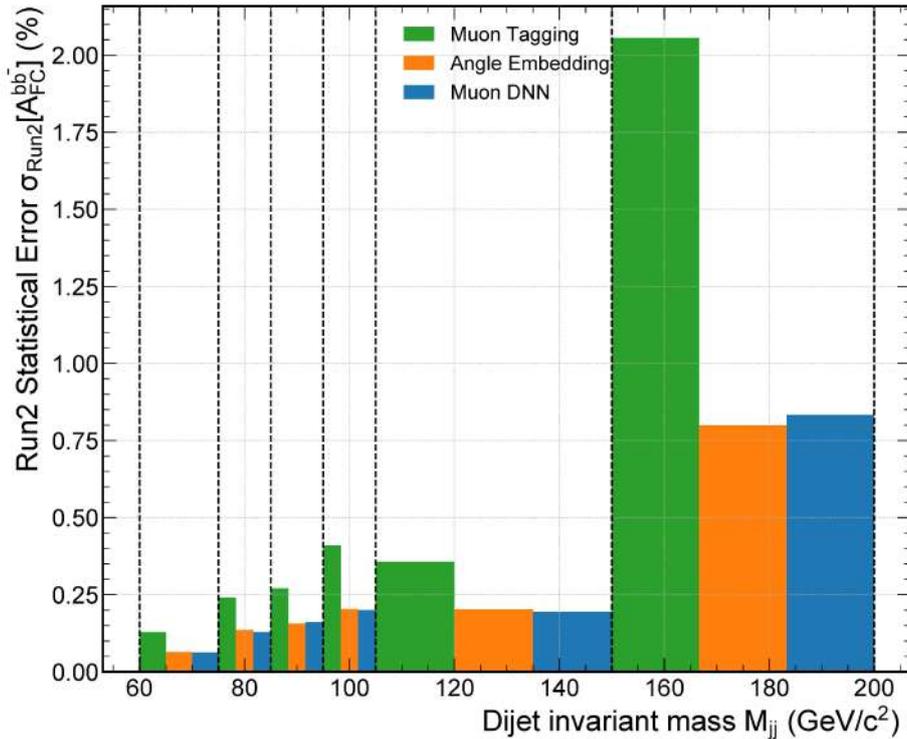
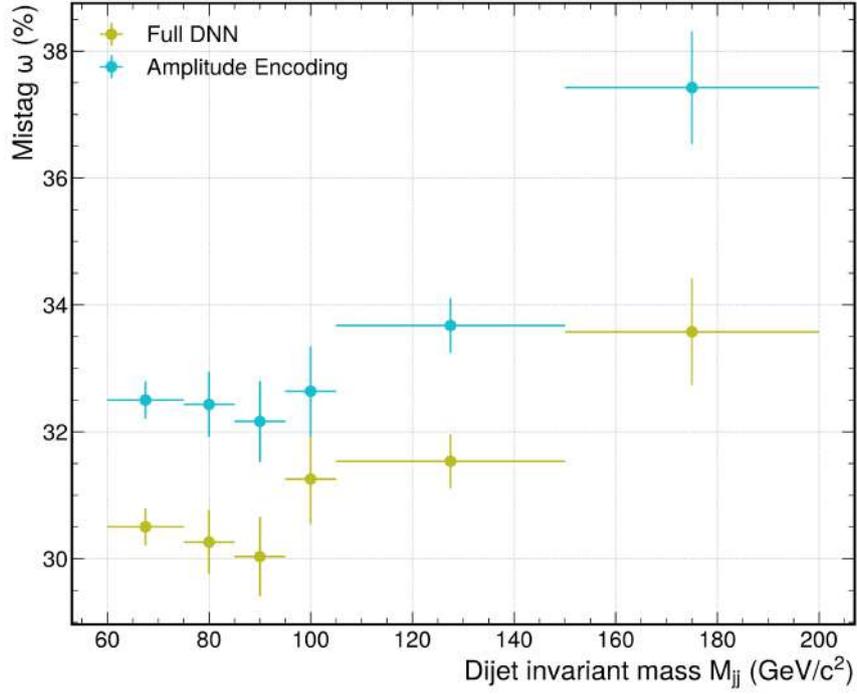


Figure 5.6: Bar plot comparing the Run2 expected uncertainties of the three tagging methods.

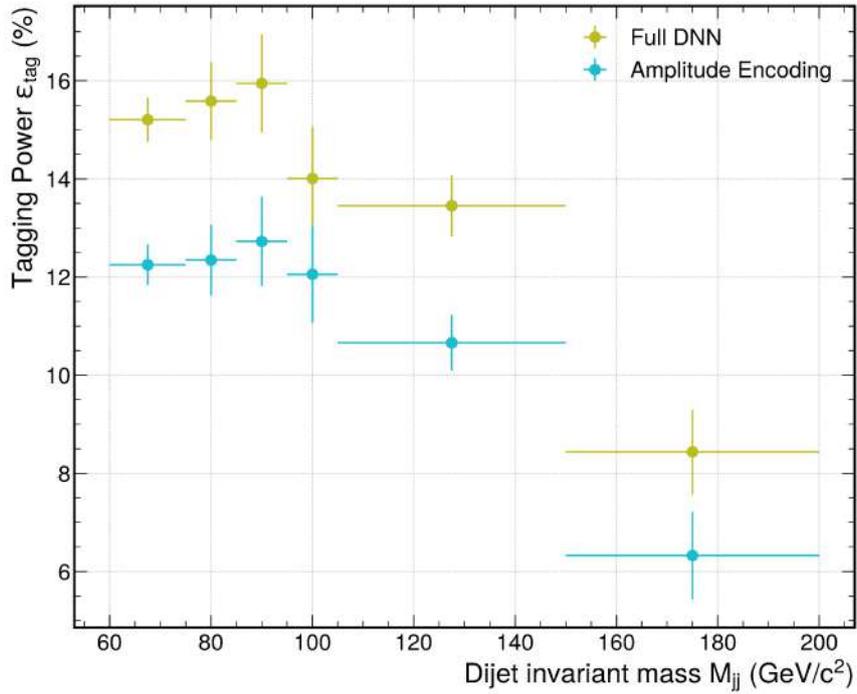
model shows better tagging performance with respect to the Amplitude Encoding model. The mis-tag evaluated per pair of jets shows, on average, values around 4% lower than the mis-tag evaluated per single jet, as shown in Figure 4.15. These results in an higher tagging power which shows a peak in the [85, 95] GeV/c², of around 16% for the DNN and 13% for the QML model. This fact is due to the procedure of choosing the highest probability prediction to tag one the jets of an event and assigning the opposite charge to the other.

Figure 5.8 presents the results on the measurement of the asymmetry. The plots on the left show the raw asymmetries while the right plots show the dilution-corrected asymmetries. Both methods produce corrected asymmetries compatible with the Monte Carlo truth within the errors, therefore no biases induced by the tagging methods have been detected.

Figure 5.9 shows the expected Run2 statistical uncertainties of the Amplitude Encoding model and the DNN model; additionally the figure shows, as a comparison, the uncertainties of the Angle Embedding model and the DNN model acting on the muon data-set. In the first place, this figure clearly shows that models that make use of all the 16 variables of the data-set outperform the models using only the 4 variables of the muon data-set: in fact, the latter are able to provide a statistical uncertainty on the full 6 fb⁻¹ of the Run2 that is almost double with respect to the 16 variables models in each bin. In the second place, the same figure shows that the 16 variables DNN performs slightly better with respect to the QML Amplitude Encoding model: however, this is expected since the latter has shown an higher mis-tag rate and therefore a worst tagging power in each bin (as shown in Figure 5.7). The discussion on the results will be given in the next chapter.



(a)



(b)

Figure 5.7: Tagging performance of the DNN and the Amplitude Encoding on the complete data-set: (a) Mis-tag ω , (b) Tagging power ϵ_{tag} .

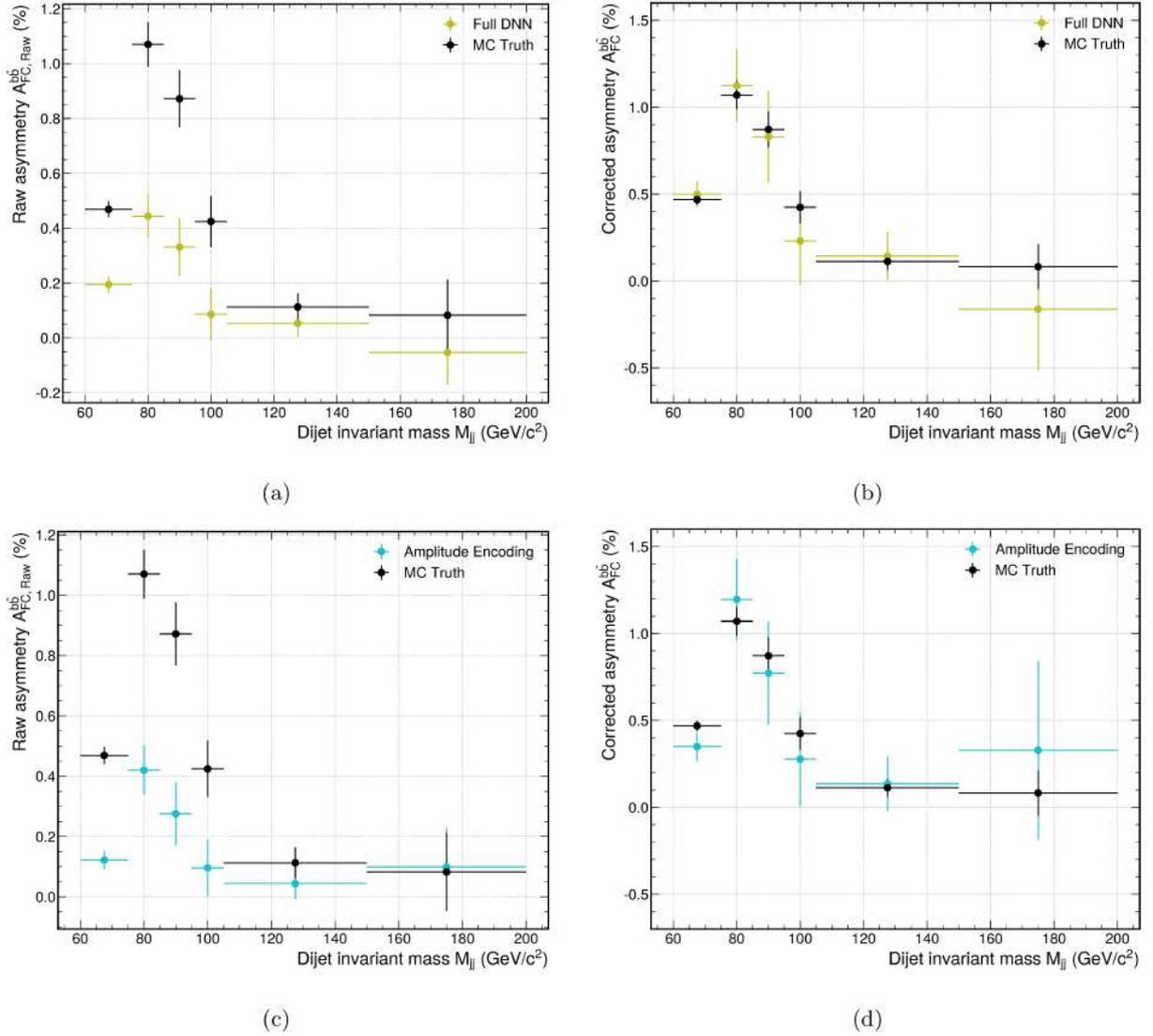


Figure 5.8: Raw asymmetries (on the left) and Corrected asymmetries (on the right), computed with the Amplitude encoding model and the DNN model on the complete data-set.

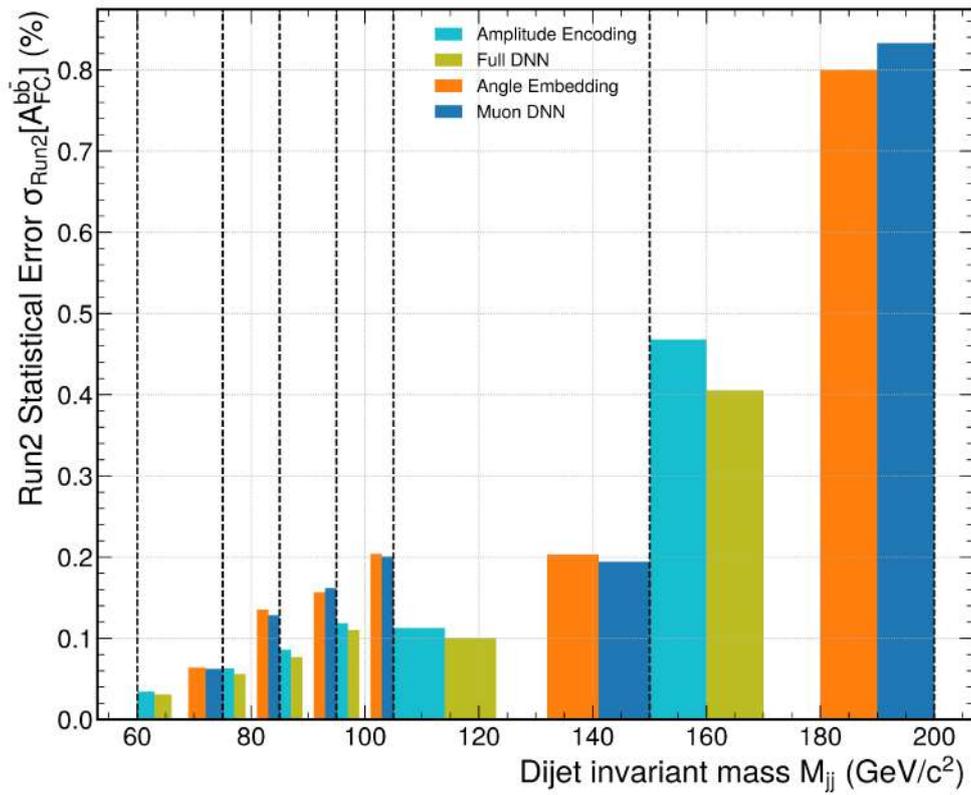


Figure 5.9: Bar plot comparing the Run2 expected uncertainties of the Amplitude Encoding (in light blue) and the DNN (in light green). Results on the muon algorithms are also reported for comparison.

Chapter 6

Conclusions and future developments

6.1 Conclusions

In this thesis, I presented the first proof-of-concept of the application of QML algorithms to solve the problem of charge identification of b -jets on LHCb simulated data. I have studied several QML models and compared them to classical ML algorithms in terms of tagging performance. Then, I computed the Forward-Central asymmetry on simulated data aiming to estimate the statistical uncertainty that each tagger is able to provide on it when the full integrated luminosity acquired during the Run2 of LHCb is used. The results can be summarized as follows:

- Both quantum and classical ML algorithms outperform the Muon Tagging method, as shown in Figures 5.4, 4.12, 4.12 and 5.6. This result is expected since the performance of the latter is strongly limited by the branching ratio of the semi-leptonic decay of b -hadrons into muons, which keeps the tagging power at $\approx 1\%$. Even when restricted to the reduced Muon data-set, ML algorithms have better performance of the Muon Tagging, being able to exploit additional information of the muon properties and of the global jet charge.
- The quantum Angle Embedding model and the classical DNN model, both trained on 4-variables of the Muon data-set, using the same number of jets, show similar mis-tag rates and tagging powers within the errors (Figures 4.12-4.13). As a consequence, both tagging algorithms are able to provide similar uncertainties when used to estimate the $b\bar{b}$ asymmetry (Figure 5.6).
- Results on the complete 16-variables data-set show that the classical DNN algorithm performs slightly better than any tested QML model, reaching an average tagging power of $\approx 7\%$ compared to $\approx 5\%$ of the Amplitude Encoding model (Figure 4.16). This performance gap gets enhanced when the tagging power is evaluated on $b\bar{b}$ dijet pairs, as shown in Figure 5.7. However, due to computational limitations, quantum models have been trained on a reduced number of jets, while the DNN has been trained using the complete Monte Carlo statistics (see discussion on the next paragraph).

6.1.1 Final considerations on Quantum Machine Learning

The computational constraints of current quantum simulators put hard limits on the execution of ML tasks using quantum circuits, forcing to work with simple quantum models and small data-sets. This makes very challenging a “fair” comparison between quantum and classical methods, in terms of number of trainable parameters, size of the training data-set and number of variables. In this work, I have compared QML and classical ML algorithms in two configurations:

- **Reduced data-set:** models trained on the muon data-set (Angle Embedding model and Muon DNN model) have been fed with the same number of jets (60 000), but with a simplified data structure, containing only 4 variables. In this case, although the QML model requires only 48

parameters to be trained versus $\approx 180\,000$ parameters of the Muon DNN, the two models show the same tagging performance within the errors.

- **Complete data-set:** this data-set features 16 variables and a total of $\approx 420\,000$ jets for training. With nowadays quantum simulators, the training of a QML model under these conditions is unfeasible in a reasonable time (training the Data Re-Uploading classifier on the full data-set would require approximately 20 days). On the other hand, a large number of training jets is mandatory for the training of a DNN model, which can benefit from GPU acceleration, due to the large number of parameters to be optimized. For these reasons, I decided to train the DNN with the full available data to exploit its capabilities, while reducing the training size of QML models to 8000 jets. With this choice, each model is tested in the best conditions compatible with the practical working environment determined by the execution time. The DNN shows better performance of all the QML algorithms, however the quantum Amplitude Encoding model perform worse just by 2% on the tagging power, which is a notable result. Furthermore, in this case, quantum algorithms are able to achieve this performance with much smaller number of trainable parameters: the DNN requires around 100000 parameters, while the Amplitude Encoding algorithm, only 72.

Despite of the computational constraints, this first proof-of-concept suggests that even simple QML models are able to achieve performance comparable with much more complex DNN models.

6.2 Future Development

6.2.1 New quantum models and use of large data-sets

The comparison of the performance of different quantum algorithms on the complete data-set has shown that an higher tagging power is not necessarily associated with a larger number of parameter of the circuit, instead, the circuit structure seems to play a major role: the Amplitude Encoding model with 72 trainable parameters performed better than the Data Re-Uploading model which requires 216 parameters to be optimized, and comparably with the Tree-Topology model featuring 104 trainable parameters. Future analysis should include the exploration of new quantum models, further investigating the relationship between the quantum circuit ansatz and structure of the input data-set. Possible biases on the physics measurements have to be controlled even thought from the tests I performed in this thesis it seems not to be a critical point.

As previously stated, current computational constraints of quantum simulators strictly bound the possibility to explore quantum models featuring more qubits, more trainable parameters and more complex structures. Even if nowadays quantum simulators are able to deal with 30-qubits systems, usually they are not optimized for ML tasks which require a large number of evaluation of the quantum circuit in parallel (*batch execution*), to estimate the gradient during the training phase. Classical ML models, such DNNs, efficiently exploits batch execution via the GPUs processing power which provide a large speed-up during the training, allowing to process large data-sets in reasonable time.

When more efficient quantum simulators will become available, an extension to this analysis can consist in testing quantum models on the complete data-set, making use of the full number of available training jets, in order to produce fair comparisons with classical ML methods.

6.2.2 Quantum noise

In this work, I tested the proposed quantum algorithms using noiseless quantum simulators. However, measurements on quantum circuits are always affected by noise originating from the interaction of the n -qubits system with the environment.

There exist several quantum simulators that allow for noise-aware simulations of quantum circuits: as an example, the IBM Qiskit Aer simulator offers the possibility to use realistic quantum noise models,

tuned on real quantum hardware available via the IBM Quantum Experience.

I have performed a preliminary test consisting in training the simple Angle Embedding model with the noise-aware Qiskit Aer quantum simulator. Unfortunately, the training procedure cannot be completed even with small training data-sets because of the too high RAM occupation (the test was executed on a system with 32 GB of memory and it crashed before completing the first optimization epoch because it ran out of RAM). This is expected, since noise simulators require a density matrix description of the quantum state instead of a simple state vector, in order to properly express mixed states. Moreover, at the moment, noise-aware simulators are not optimized for ML tasks which require a large number of evaluation of the quantum circuit for the estimation of the gradient.

As software libraries and simulators are quickly improving over time, the use of noise-aware simulators can be considered for future studies about how quantum noise can affect this kind of models.

6.2.3 Quantum hardware

At the moment, quantum hardware is publicly available via quantum computing cloud solutions such as:

- **IBM Quantum Experience:** this platform provide free cloud-access to several quantum devices with a variable number of qubits, ranging from 1 to 65. The concurrent access to the systems by multiple users is managed via a fair-share queue system that is supposed to avoid the monopolization of a device by a single user, this causes that multiple circuits executions require queue waiting times of the order of several seconds/few minutes, that allows the other users to make use of the same resource. This solution is highly unfeasible for ML training tasks since they require a large number of circuits evaluations to perform an optimization step.
- **Amazon Braket:** this solution is included in the Amazon Web Services (AWS) and provides cloud-access to different quantum hardware and simulators. The available simulators are:
 - *SV1* is a noise-less simulator that is able to simulate a quantum circuit up to 34-qubits in around 2 hours.
 - *TN1* is another noise-less simulator that uses tensor-network model, optimized to simulate some kind of quantum circuits up to 50 qubits.

The available hardware devices are:

- *IonQ* is a 11-qubits general-purpose circuit-based quantum device based on ion trap technology.
- *Rigetti Aspen-9* is a 32-qubits general-purpose circuit-based quantum device, based on superconducting qubits.
- *D-Wave* is a quantum annealer which is a specific-purpose machine designed to solve problems belonging to the class of Quadratic Unconstrained Optimisation (QUBO). This device does not support quantum gates and quantum circuits.

Hardware devices are not always available, but they can be used only for a limited amount of time during the day, then they need to undergo a new calibration before being available again the next day.

Thanks to an agreement between INFN and Amazon AWS, I have been able to test the training of a simplified Angle Embedding model with a reduced data-set containing only few events, running on the Rigetti Aspen-9 device, via the Amazon Braket platform. Unfortunately, I was unable to complete the training for the following two reasons:

- Even if the execution of the quantum circuit on the hardware takes time of the order of 1 ms, the access time to the hardware resource was found to be of the order of several seconds. This

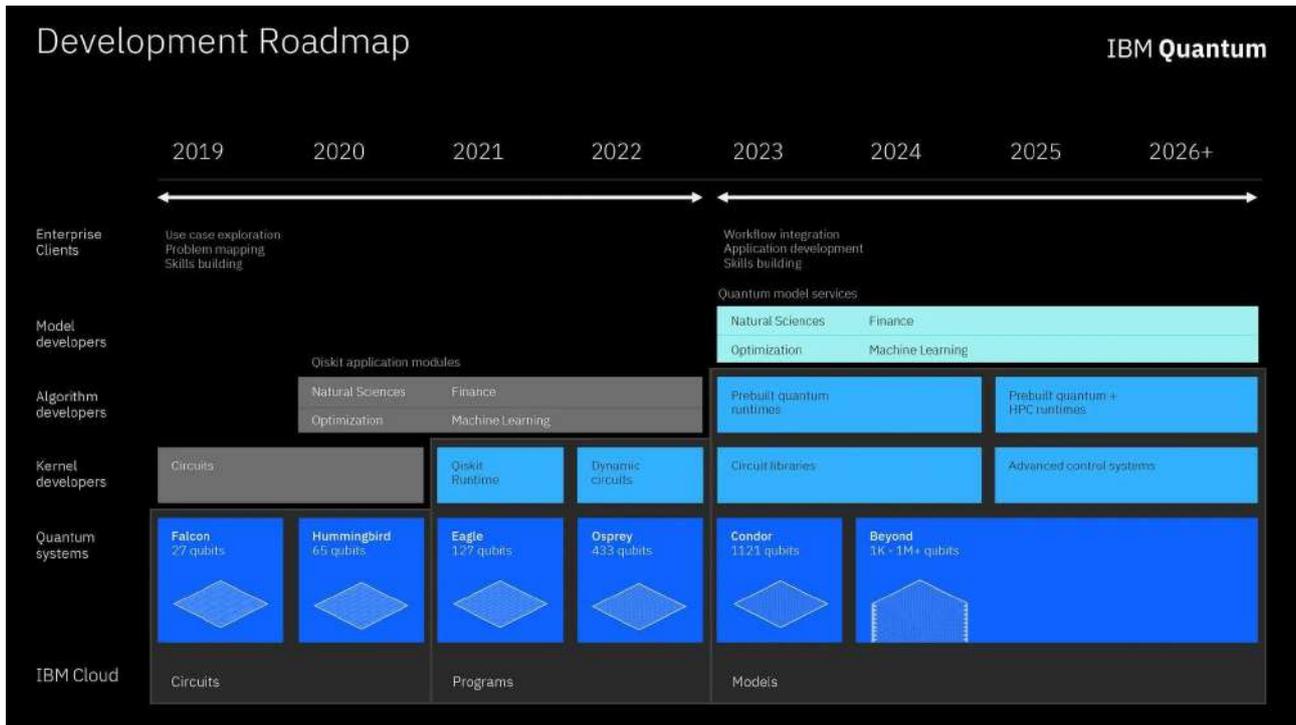


Figure 6.1: IBM Quantum road-map for the next 5 years. [44]

represent a bottleneck for the training procedure that requires the execution of a large number of circuits.

- Since hardware devices are available only a fraction of the full 24 hours a day, the total run-time required for a training procedure reaches the order of centuries. Moreover, the code must be carefully adapted to automatically handle the daily maintenance stops of the hardware.

Current publicly available quantum hardware has proved to be unfeasible for the execution of our QML algorithm. However, the availability of quantum resources is expected to grow quickly in the near future. Figure 6.1 shows the IBM road-map in the development of quantum devices for the upcoming years. According to this figure, by the 2022, devices with several hundred of qubits should become available, and by the 2026 devices with several thousands or even millions of qubits could appear. Devices with a large number of qubits can provide a large speed up in the training procedure, since multiple copies of a quantum circuit can be executed in parallel querying the device only once. Therefore, in the next years, current hardware limitations can be overcome and the algorithms proposed in this work can be tested directly on quantum devices.

6.2.4 Search for particles correlations inside jets

The internal structure of jets is the result of non-perturbative QCD mechanisms occurring in the hadronization process. Therefore, the information on the parton that initiates it, is diluted in the sea of the final state particles. The ability of detecting correlations among these particles is unique to quantum algorithms and opens new possibilities in building up new methods of jets reconstruction and identification, which then reflect in new physics measurements, see for example [35].

In addition to that, other possibilities are offered by quantum models. The Grover's quantum algorithm[40] (also known as *quantum search algorithm*) is able to solve the problem of unstructured search in a database using only $O(\sqrt{N})$ queries; a classical algorithm cannot solve the same problem using less than $O(N)$ database queries, which proves the quantum advantage of the Grover's algorithm. The Grover's algorithm has inspired several other algorithms for quantum pattern searching[50] and quantum feature selection[18, 43].

Future employment of these new techniques could, for example, help tracks reconstruction in fu-

ture experiments at future colliders where the hits multiplicity is very difficult to manage. Moreover, this approach could suggest new, more optimized, parametric circuits that may change the way physics object will be reconstructed the future HEP experiments.

Bibliography

- [1] R. Aaij et al. “Performance of the LHCb Vertex Locator”. In: *Journal of Instrumentation* 9.09 (Sept. 2014), P09007–P09007. DOI: 10.1088/1748-0221/9/09/p09007. URL: <https://doi.org/10.1088/1748-0221/9/09/p09007>.
- [2] R. Aaij et al. “First Measurement of the Charge Asymmetry in Beauty-Quark Pair Production”. In: *Physical Review Letters* 113.8 (Aug. 2014). ISSN: 1079-7114. DOI: 10.1103/physrevlett.113.082003. URL: <http://dx.doi.org/10.1103/PhysRevLett.113.082003>.
- [3] Roel Aaij et al. “Measurement of differential $b\bar{b}$ - and $c\bar{c}$ -dijet cross-sections in the forward region of pp collisions at $\sqrt{s} = 13$ TeV”. In: *JHEP* 02 (2021), p. 023. DOI: 10.1007/JHEP02(2021)023. arXiv: 2010.09437 [hep-ex].
- [4] T. Aaltonen et al. “First measurement of the forward-backward asymmetry in bottom-quark pair production at high mass”. In: *Physical Review D* 92.3 (Aug. 2015). ISSN: 1550-2368. DOI: 10.1103/physrevd.92.032006. URL: <http://dx.doi.org/10.1103/PhysRevD.92.032006>.
- [5] V. M. Abazov et al. “Measurement of the Forward-Backward Asymmetry in the Production of B^\pm Mesons in $p\bar{p}$ Collisions at $\sqrt{s} = 1.96$ TeV”. In: *Physical Review Letters* 114.5 (Feb. 2015). ISSN: 1079-7114. DOI: 10.1103/physrevlett.114.051803. URL: <http://dx.doi.org/10.1103/PhysRevLett.114.051803>.
- [6] S. Agostinelli et al. “GEANT4—a simulation toolkit”. In: *Nucl. Instrum. Meth. A* 506 (2003), pp. 250–303. DOI: 10.1016/S0168-9002(03)01368-8.
- [7] Martin Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: <http://tensorflow.org/>.
- [8] Md Sajid Anis et al. *Qiskit: An Open-source Framework for Quantum Computing*. 2021. DOI: 10.5281/zenodo.2573505.
- [9] Ezequiel Álvarez, Leandro Da Rold, and Alejandro Szynkman. “A composite Higgs model analysis of forward-backward asymmetries in the production of tops at Tevatron and bottoms at LEP and SLC”. In: *Journal of High Energy Physics* 2011.5 (May 2011). ISSN: 1029-8479. DOI: 10.1007/jhep05(2011)070. URL: [http://dx.doi.org/10.1007/JHEP05\(2011\)070](http://dx.doi.org/10.1007/JHEP05(2011)070).
- [10] J. S. Bell. “On the Einstein Podolsky Rosen paradox”. In: *Physics Physique Fizika* 1 (3 Nov. 1964), pp. 195–200. DOI: 10.1103/PhysicsPhysiqueFizika.1.195. URL: <https://link.aps.org/doi/10.1103/PhysicsPhysiqueFizika.1.195>.
- [11] Eli Ben-Haim. “The b quark fragmentation function, from LEP to TeVatron”. PhD thesis. 2004.
- [12] Marcello Benedetti et al. “Parameterized quantum circuits as machine learning models”. In: *Quantum Science and Technology* 4.4 (Nov. 2019), p. 043001. ISSN: 2058-9565. DOI: 10.1088/2058-9565/ab4eb5. URL: <http://dx.doi.org/10.1088/2058-9565/ab4eb5>.
- [13] Ville Bergholm et al. *PennyLane: Automatic differentiation of hybrid quantum-classical computations*. 2020. arXiv: 1811.04968 [quant-ph].
- [14] Kfir Blum, Yonit Hochberg, and Yosef Nir. “Scalar-mediated $t\bar{t}$ forward-backward asymmetry”. In: *Journal of High Energy Physics* 2011.10 (Oct. 2011). ISSN: 1029-8479. DOI: 10.1007/jhep10(2011)124. URL: [http://dx.doi.org/10.1007/JHEP10\(2011\)124](http://dx.doi.org/10.1007/JHEP10(2011)124).

- [15] E. Bols et al. “Jet flavour classification using DeepJet”. In: *Journal of Instrumentation* 15.12 (Dec. 2020), P12012–P12012. ISSN: 1748-0221. DOI: 10.1088/1748-0221/15/12/p12012. URL: <http://dx.doi.org/10.1088/1748-0221/15/12/P12012>.
- [16] E. Bols et al. “Jet flavour classification using DeepJet”. In: *Journal of Instrumentation* 15.12 (Dec. 2020), P12012–P12012. ISSN: 1748-0221. DOI: 10.1088/1748-0221/15/12/p12012. URL: <http://dx.doi.org/10.1088/1748-0221/15/12/P12012>.
- [17] Matteo Cacciari, Gavin P Salam, and Gregory Soyez. “The anti-ktjet clustering algorithm”. In: *Journal of High Energy Physics* 2008.04 (Apr. 2008), pp. 063–063. DOI: 10.1088/1126-6708/2008/04/063. URL: <https://doi.org/10.1088/1126-6708/2008/04/063>.
- [18] Sanjay Chakraborty et al. “A hybrid quantum feature selection algorithm using a quantum inspired graph theoretic approach”. In: *Applied Intelligence* 50 (June 2020), pp. 1775–1793. DOI: 10.1007/s10489-019-01604-3.
- [19] François Chollet et al. *Keras*. <https://keras.io>. 2015.
- [20] CDF Collaboration et al. *Measurement of the forward-backward asymmetry in low-mass bottom-quark pairs produced in proton-antiproton collisions*. 2016. arXiv: 1601.06526 [hep-ex].
- [21] LHCb collaboration. *Simulated jet samples for quark flavour identification studies*. CERN Open Data Portal. 2020. URL: <http://opendata.cern.ch/record/4910>.
- [22] The ALICE Collaboration. “The ALICE experiment at the CERN LHC”. In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08002–S08002. DOI: 10.1088/1748-0221/3/08/s08002. URL: <https://doi.org/10.1088/1748-0221/3/08/s08002>.
- [23] The ATLAS Collaboration. “The ATLAS Experiment at the CERN Large Hadron Collider”. In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08003–S08003. DOI: 10.1088/1748-0221/3/08/s08003. URL: <https://doi.org/10.1088/1748-0221/3/08/s08003>.
- [24] The CMS Collaboration. “The CMS experiment at the CERN LHC”. In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08004–S08004. DOI: 10.1088/1748-0221/3/08/s08004. URL: <https://doi.org/10.1088/1748-0221/3/08/s08004>.
- [25] The LHCb Collaboration. “The LHCb Detector at the LHC”. In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08005–S08005. DOI: 10.1088/1748-0221/3/08/s08005. URL: <https://doi.org/10.1088/1748-0221/3/08/s08005>.
- [26] The LHCb collaboration. “Identification of beauty and charm quark jets at LHCb”. In: *Journal of Instrumentation* 10.06 (June 2015), P06013–P06013. DOI: 10.1088/1748-0221/10/06/p06013. URL: <https://doi.org/10.1088/1748-0221/10/06/p06013>.
- [27] John C. Collins, Davison E. Soper, and George F. Sterman. “Factorization of Hard Processes in QCD”. In: *Adv. Ser. Direct. High Energy Phys.* 5 (1989), pp. 1–91. DOI: 10.1142/9789814503266_0001. arXiv: hep-ph/0409313.
- [28] Wikimedia Commons. *Standard Model of Elementary Particles*. 2018. URL: https://commons.wikimedia.org/wiki/File:Standard%5C_Model%5C_of%5C_Elementary%5C_Particles%5C_Anti.svg.
- [29] Wikimedia Commons. *The Bloch sphere, a geometric representation of a two-level quantum system*. 2009. URL: https://commons.wikimedia.org/wiki/File:Bloch%5C_sphere.svg.
- [30] James P. Egan, Gordon Z. Greenberg, and Arthur I. Schulman. “Operating Characteristics, Signal Detectability, and the Method of Free Response”. In: *The Journal of the Acoustical Society of America* 33.8 (1961), pp. 993–1007. DOI: 10.1121/1.1908935. eprint: <https://doi.org/10.1121/1.1908935>. URL: <https://doi.org/10.1121/1.1908935>.
- [31] E. Eichten et al. “Charmonium: The model”. In: *Phys. Rev. D* 17 (11 June 1978), pp. 3090–3117. DOI: 10.1103/PhysRevD.17.3090. URL: <https://link.aps.org/doi/10.1103/PhysRevD.17.3090>.

- [32] F. Englert and R. Brout. “Broken Symmetry and the Mass of Gauge Vector Mesons”. In: *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>.
- [33] Lyndon Evans and Philip Bryant. “LHC Machine”. In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08001–S08001. DOI: 10.1088/1748-0221/3/08/s08001. URL: <https://doi.org/10.1088/1748-0221/3/08/s08001>.
- [34] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. *A Quantum Approximate Optimization Algorithm*. 2014. arXiv: 1411.4028 [quant-ph].
- [35] Timo Felser et al. *Quantum-inspired Machine Learning on high-energy physics data*. 2021. arXiv: 2004.13747 [stat.ML].
- [36] F Follin and D Jacquet. “Implementation and experience with luminosity levelling with offset beam”. In: (Oct. 2014). Comments: 5 pages, contribution to the ICFA Mini-Workshop on Beam-Beam Effects in Hadron Colliders, CERN, Geneva, Switzerland, 18-22 Mar 2013, 5 p. DOI: 10.5170/CERN-2014-004.183. arXiv: 1410.3667. URL: <http://cds.cern.ch/record/1955354>.
- [37] R. Frühwirth. “Application of Kalman filtering to track and vertex fitting”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 262.2 (1987), pp. 444–450. ISSN: 0168-9002. DOI: [https://doi.org/10.1016/0168-9002\(87\)90887-4](https://doi.org/10.1016/0168-9002(87)90887-4). URL: <https://www.sciencedirect.com/science/article/pii/0168900287908874>.
- [38] Rhorry Gauld, Ulrich Haisch, and Benjamin D. Pecjak. “Asymmetric heavy-quark hadroproduction at LHCb: Predictions and applications”. In: *JHEP* 03 (2019), p. 166. DOI: 10.1007/JHEP03(2019)166. arXiv: 1901.07573 [hep-ph].
- [39] Sheldon L. Glashow. “Partial-symmetries of weak interactions”. In: *Nuclear Physics* 22.4 (1961), pp. 579–588. ISSN: 0029-5582. DOI: [https://doi.org/10.1016/0029-5582\(61\)90469-2](https://doi.org/10.1016/0029-5582(61)90469-2). URL: <https://www.sciencedirect.com/science/article/pii/0029558261904692>.
- [40] Lov K. Grover. “A Fast quantum mechanical algorithm for database search”. In: (May 1996). arXiv: quant-ph/9605043.
- [41] M. Y. Han and Y. Nambu. “Three-Triplet Model with Double SU(3) Symmetry”. In: *Phys. Rev.* 139 (4B Aug. 1965), B1006–B1010. DOI: 10.1103/PhysRev.139.B1006. URL: <https://link.aps.org/doi/10.1103/PhysRev.139.B1006>.
- [42] Vojtěch Havlíček et al. “Supervised learning with quantum-enhanced feature spaces”. In: *Nature* 567.7747 (Mar. 2019), pp. 209–212. ISSN: 1476-4687. DOI: 10.1038/s41586-019-0980-2. URL: <http://dx.doi.org/10.1038/s41586-019-0980-2>.
- [43] Zhimin He et al. “Quantum-enhanced feature selection with forward selection and backward elimination”. In: *Quantum Information Processing* 17.7, 154 (July 2018), p. 154. DOI: 10.1007/s11128-018-1924-8.
- [44] IBM. *IBM’s roadmap for building an open quantum software ecosystem*. 2020. URL: <https://research.ibm.com/blog/quantum-development-roadmap>.
- [45] Seyda Ipek. “Light axigluon contributions to $b\bar{b}$ and $c\bar{c}$ asymmetry and constraints on flavor changing axigluon currents”. In: *Physical Review D* 87.11 (June 2013). ISSN: 1550-2368. DOI: 10.1103/physrevd.87.116010. URL: <http://dx.doi.org/10.1103/PhysRevD.87.116010>.
- [46] Tyson Jones and Julien Gacon. *Efficient calculation of gradients in classical simulations of variational quantum algorithms*. 2020. arXiv: 2009.02823 [quant-ph].
- [47] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].
- [48] Marcin Kucharczyk, Piotr Morawski, and Mariusz Witek. *Primary Vertex Reconstruction at LHCb*. Tech. rep. Geneva: CERN, Sept. 2014. URL: <https://cds.cern.ch/record/1756296>.
- [49] “LHCb detector performance”. In: *International Journal of Modern Physics A* 30.07 (2015), p. 1530022. DOI: 10.1142/S0217751X15300227.

- [50] P. Mateus and Y. Omar. “Quantum Pattern Matching”. In: *arXiv e-prints*, quant-ph/0508237 (Aug. 2005), quant-ph/0508237. arXiv: quant-ph/0508237 [quant-ph].
- [51] Samuel Matzek et al. *Data-parallel distributed training of very large models beyond GPU capacity*. 2018. arXiv: 1811.12174 [cs.DC].
- [52] Bertrand Michel and Anthony Nouy. *Learning with tree tensor networks: complexity estimates and model selection*. 2021. arXiv: 2007.01165 [math.ST].
- [53] K. Mitarai et al. “Quantum circuit learning”. In: *Physical Review A* 98.3 (Sept. 2018). ISSN: 2469-9934. DOI: 10.1103/physreva.98.032309. URL: <http://dx.doi.org/10.1103/PhysRevA.98.032309>.
- [54] Christopher W. Murphy. “Bottom-quark forward-backward and charge asymmetries at hadron colliders”. In: *Physical Review D* 92.5 (Sept. 2015). ISSN: 1550-2368. DOI: 10.1103/physrevd.92.054003. URL: <http://dx.doi.org/10.1103/PhysRevD.92.054003>.
- [55] Michael A Nielsen and Isaac Chuang. *Quantum computation and quantum information*. 2002.
- [56] Michela Paganini. “Machine Learning Algorithms for b -Jet Tagging at the ATLAS Experiment”. In: *J. Phys. Conf. Ser.* 1085.4 (2018), p. 042031. DOI: 10.1088/1742-6596/1085/4/042031. arXiv: 1711.08811 [hep-ex].
- [57] Adam Paszke et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019. arXiv: 1912.01703 [cs.LG].
- [58] Adrián Pérez-Salinas et al. “Data re-uploading for a universal quantum classifier”. In: *Quantum* 4 (Feb. 2020), p. 226. ISSN: 2521-327X. DOI: 10.22331/q-2020-02-06-226. URL: <http://dx.doi.org/10.22331/q-2020-02-06-226>.
- [59] C. Peterson et al. “Scaling violations in inclusive e^+e^- annihilation spectra”. In: *Phys. Rev. D* 27 (1 Jan. 1983), pp. 105–111. DOI: 10.1103/PhysRevD.27.105. URL: <https://link.aps.org/doi/10.1103/PhysRevD.27.105>.
- [60] “Precision electroweak measurements on the Z resonance”. In: *Physics Reports* 427.5 (2006), pp. 257–454. ISSN: 0370-1573. DOI: <https://doi.org/10.1016/j.physrep.2005.12.006>. URL: <https://www.sciencedirect.com/science/article/pii/S0370157305005119>.
- [61] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. “Quantum Support Vector Machine for Big Data Classification”. In: *Physical Review Letters* 113.13 (Sept. 2014). ISSN: 1079-7114. DOI: 10.1103/physrevlett.113.130503. URL: <http://dx.doi.org/10.1103/PhysRevLett.113.130503>.
- [62] Anders Ryd et al. “EvtGen: A Monte Carlo Generator for B-Physics”. In: (May 2005).
- [63] Abdus Salam and J. C. Ward. “Weak and electromagnetic interactions”. In: *Il Nuovo Cimento (1955-1965)* 11.4 (Feb. 1959), pp. 568–577. ISSN: 1827-6121. DOI: 10.1007/BF02726525. URL: <https://doi.org/10.1007/BF02726525>.
- [64] Maria Schuld and Francesco Petruccione. *Supervised learning with quantum computers*. Vol. 17. Springer, 2018.
- [65] Maria Schuld et al. “Evaluating analytic gradients on quantum hardware”. In: *Physical Review A* 99.3 (Mar. 2019). ISSN: 2469-9934. DOI: 10.1103/physreva.99.032331. URL: <http://dx.doi.org/10.1103/PhysRevA.99.032331>.
- [66] P.W. Shor. “Algorithms for quantum computation: discrete logarithms and factoring”. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. 1994, pp. 124–134. DOI: 10.1109/SFCS.1994.365700.
- [67] A.M. Sirunyan et al. “Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV”. In: *Journal of Instrumentation* 13.05 (May 2018), P05011–P05011. ISSN: 1748-0221. DOI: 10.1088/1748-0221/13/05/p05011. URL: <http://dx.doi.org/10.1088/1748-0221/13/05/P05011>.

- [68] Torbjorn Sjostrand, Stephen Mrenna, and Peter Z. Skands. “A Brief Introduction to PYTHIA 8.1”. In: *Comput. Phys. Commun.* 178 (2008), pp. 852–867. DOI: 10.1016/j.cpc.2008.01.036. arXiv: 0710.3820 [hep-ph].
- [69] Yasunari Suzuki et al. *Qulacs: a fast and versatile quantum circuit simulator for research purpose*. 2021. arXiv: 2011.13524 [quant-ph].
- [70] B. A. Thacker and G. Peter Lepage. “Heavy quark bound states in lattice QCD”. In: *Phys. Rev. D* 43 (1991), pp. 196–208. DOI: 10.1103/PhysRevD.43.196.
- [71] Steven Weinberg. “A Model of Leptons”. In: *Phys. Rev. Lett.* 19 (21 Nov. 1967), pp. 1264–1266. DOI: 10.1103/PhysRevLett.19.1264. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.19.1264>.