
Università degli Studi di Torino
Dipartimento di Fisica



Master Thesis in Nuclear and Subnuclear Physics

**Soft QCD parameter tuning using
feed-forward neural networks**

Relatore:

Prof. Roberto Covarelli

Candidato:

Michael Dessenà

Correlatore:

Prof. Marco Monteno

Controrelatore:

Prof. Paolo Torrielli

Anno Accademico 2021/2022

A Federico

“Iniziare non vuol dire che dovrai finire.”

RANCORE, S.U.N.S.H.I.N.E.

Abstract

Soft QCD studies require the use of Monte Carlo generators based on phenomenological models: these models introduce a lot of free parameters that have to be tuned with the real data. Due to the high computational cost of running a generator it is important to reduce the number of Monte Carlo job required, this is done employing a parametrization-based approach were the real response of the generator is replaced by a surrogate one. The state-of-art for this approach is the use of a polynomial parametrization. In this work, an alternative approach based on machine learning techniques, implemented in the Python package MCNNTUNES by means of Feed Forward Neural Networks, is used to tune parameters.

Firstly, the basic theoretical concepts used in the Monte Carlo simulations are introduced. The aim of our work was to validate MCNNTUNES as a tool for the tuning of Monte Carlo generators in High Energy Physics simulations and than use it for a further tune. To perform this validation we try to reproduce an already existing tune for the description of the Underlying Event in hadron-hadron collisions using data collected in Minimum Bias observations. So, in the first part the parameters related to the Multi Parton Interactions and Color Reconnection, in PYTHIA event generator, have been tuned. The tune was performed employing data from the CDF and CMS experiments at different center-of-mass energy. The validation of the MCNNTUNES tool is performed in two steps: a first test with a restrict number of free parameters in order to test all the functionality and check the different operation mode in a simpler case than the actual tune. Once the tool as been tested the actual tune is performed.

Then after that the validation of MCNNTUNES as a valid tool has been performed, we employed it for the tune of the Primordial k_T and Initial State Radiation parameters using data collected in Z boson-production events by the CMS experiment at the center-of-mass energy of 13 TeV.

Contents

1	Introduction	1
2	Hadron-Hadron Scattering	7
2.1	Structure of Monte Carlo event generators	7
2.2	QCD factorization theorem	9
2.2.1	Parton distribution functions	9
2.2.2	Partonic cross-section	11
2.3	All-order approaches and PYTHIA Monte Carlo generator	13
2.3.1	Parton Showers	14
2.3.2	Merging parton showers and NLO matrix element calculations	15
2.4	Multiple Parton Interactions in PYTHIA	17
2.4.1	Basic Concepts	17
2.4.2	Momentum and flavour conservation	21
2.4.3	Impact Parameter Dependence	23
2.4.4	Beam Beam Remnants and primordial k_T	24
2.4.5	Color Reconnection	25
2.5	PYTHIA summary	25
3	CMS Detector, Observable to Study the Underlying Event and Existing Tunes	27
3.1	Compact Muon Solenoid Detector	27
3.1.1	Selection Criteria	29
3.2	Minimum Bias Measurements and Underlying Event topology	30
3.2.1	Hard scale dependence	31
3.2.2	Sensitivity to MPI parameters	31
3.2.3	Observables in Z production processes	34
3.3	Previous Tune for the Underlying Event	36
3.3.1	The distributions used for the Tune	37
3.3.2	Pyhtia configuration and the tunes	37
4	Tuning procedure and MCNNTUNES	39
4.1	Parametrization-based approach	40
4.2	Machine Learning and Neural Networks	40
4.2.1	Neural Networks - Perceptron	41
4.2.2	Feed-Forward Neural Networks	42
4.3	MCNNTUNES	45
4.3.1	Sampling Phase and Training Set Generation	45
4.3.2	PerBin Model	46

4.3.3	Negative aspects	48
4.3.4	Inverse Model	49
4.3.5	Weighrules in MCNNNTUNES	51
5	Tune for the Underlying Event	53
5.1	Introduction	53
5.2	First test: only two parameters variation	53
5.2.1	Per Bin Model results	54
5.2.2	Inverse Model results	57
5.2.3	Overall results	59
5.3	Tune for the UE in Minimum Bias Observations	60
5.3.1	Per Bin Model results	60
5.3.2	Inverse Model results	64
5.3.3	Overall results	65
6	Tune for the primordial k_T events	71
6.1	Primordial k_T and ISR effect on p_T^Z	71
6.2	Primordial k_T tune	73
6.3	Primordial k_T tune vs MPI	76
Conclusions		79
A Test with two free parameters for the Underlying Events		81
Bibliography		85

Chapter 1

Introduction

High-energy collisions, such as those studied at the Large Hadron Collider (LHC) in Geneva, lead to very complex final states. Usually, hundreds of particles are produced and detected by the experiments with momenta scales ranging over different orders of magnitude. All these observed final states are very complicated to understand and explain. The theory underlying the description of high energy particle interactions is the *standard model of particle* (SM).

The SM is the theory that describes all the elementary particles and the fundamental interactions: electromagnetic, weak and strong¹. The standard model relies on the Quantum Field Theory framework and on the concept of gauge symmetry. The symmetry group for the theory is $SU(3) \times SU(2) \times U(1)$.

The aim of the SM is to describe with a unique formalism all the three interactions. The abelian group $U(1)$ is related to the electromagnetic interaction, the theory based on the gauge group $U(1)$, first introduced by Richard Feynman, is called *quantum electrodynamics* (QED). During the last century, this force was unified to the weak interaction described by the gauge symmetry $SU(2)$ in a single interaction: the Electroweak interaction [1]. Last, the $SU(3)$ group is the one related to the strong interactions of quarks the theory describe the interaction in term of three possible states of charge called *colors*: red, green and blue. This is the origin of the theory name: *quantum chromodynamics* (QCD).

All the processes in the standard model are evaluated by means of matrix element calculations to give the probability of having an evolution from a initial-state to a defined final-state.

In practice, the matrix elements are to laborious to be evaluated already over the first few orders of perturbation theory, and in particular the QCD processes have to deal with the intrinsically non-perturbative problem of *color confinement*. Matrix element calculations have to deal with many divergences and often they are performed in an approximated scenario. Then, even if it has been possible to compute these matrix elements correctly, they must be integrated over a large final-state phase space in order to obtain predictions for the description of experimental ob-

¹Actually, the fundamental interactions are four: there is also the gravitational interaction. The inclusion of a description also for this last interaction is a central problem for modern physics. There are different theories that try to merge the quantum effects with the general relativity that describe the gravitational force but none of these has been able to give a final description yet. This is not important in the framework of the particle physics studies at the collider in fact this force is negligible with respect to the others.

servables. The crucial tool for the description of the processes is the *factorization*, which allows the treatment of processes separating them into different regimes according to the transferred-momentum scale of the interaction. The processes at the highest scale are referred to as *hard processes* and are described in terms of free partons (asymptotic freedom) interacting and generating a small number of energetic outgoing elementary particles. Due to the large momentum transferred in the hard interaction the value of the QCD coupling constant, α_s , is small (as shown in Fig. 1.1), so the perturbative theory is applicable. While, at the very lower scales, of the order of 1 GeV, the α_s value is large, the QCD become intrinsically non-perturbative and highly interacting, in this scenario the incoming partons are not describable in terms of free partons because they are confined into the beams hadrons² and interact in a non-perturbative way to form the observed final-state. These soft processes are the ones that cannot yet be described starting from first principles but have to be modeled by employing phenomenological models based on the introduction of free parameters. These two regimes are connected by an evolutionary process calculable using perturbative QCD. The main consequence of this scale evolution is the so-called *parton showers*: the production of many additional partons in the initial- and final-state.

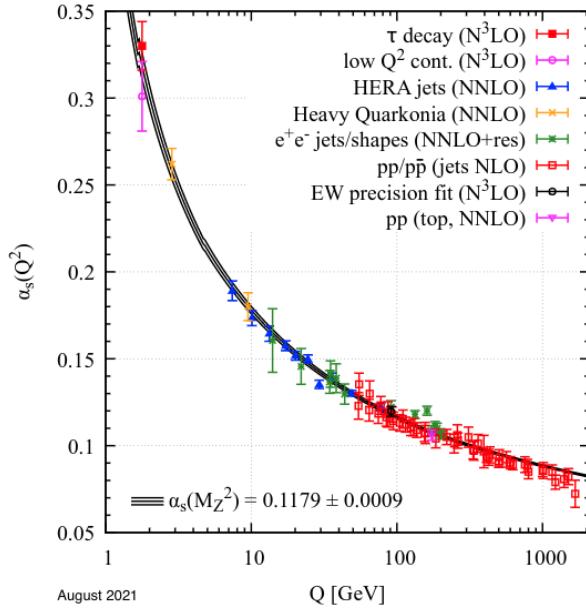


Figure 1.1: Evolution of the coupling constant with the energy scale of the interaction [2]

The description of all these processes is based on computer simulation based on Monte Carlo (MC) techniques. The evolution of scale that leads to the parton showering is a Markov process that can be easily simulated with Monte Carlo techniques, also the hadronization that leads from the parton-composed final-state to the hadron-composed final-state is described in terms of Monte Carlo processes. All these elements together give a *Monte Carlo event generator* capable of simulating a large number of interesting processes. These event generators are needed in

²The coupling constant of the QCD is large so they are strongly bounded to each other and so cannot be seen as free when they interact.

high energy physics studies. For example, they are used to extract a new physics signal from the background of all the other SM processes, provide an input for new selection or reconstruction procedures, or the construction of new experiments.

Historically, the strong interaction was associated with the hadrons and not with quarks (they were unknown yet). However, with the deep inelastic scattering experiments [3] it was proven that the hadrons are composite objects and then they cannot be fundamental particles for the SM. There must be something more fundamental: *partons* (quarks and gluons). The *partons model* [4] was developed to describe the interaction of hadrons in very high energy scatterings. What was observed in DIS of electrons on nucleons was the so-called *Bjorken scaling* the observed structure constant was (at high energy) independent from the probe energy which reflects the fact that the nucleons, or more in general hadrons, are constituted by a collection of point-like objects: the above-mentioned partons.

The development of event generators began shortly after this discovery. The description of lots of processes' features at the hard scale as DIS, hadron jets and lepton pairs production can be performed simply with the description in terms of parton collisions. The description of the complex final-states observed in the MC event generators is obtained by the observation that the primary partons in the evolution can radiate gluons and these gluons could themselves radiate other gluons leading to a cascade. The natural endpoint of this cascade is the hadronization that due to the low energy scale and the consequent high value of the coupling constant is not derivable by the QCD fundamental principles.

Most of the interactions observed at the LHC are soft, these soft processes as the above mentioned hadronization can be simulated but due to the non-perturbative nature of the processes they must include free parameters that require a *tune* in order to describe the data. A really important phenomenon related to this topic is the study of the so-called *underlying event* (UE) i.e. the soft component of a collision, where at least one hard interaction occurs, that increases the complexity of the final state observed by adding more particles to it.

The hard tail of the UE is described by the QCD perturbative approach using matrix element calculations. But along with it the description of the showering process need to be simulated: the *Initial-State Radiation* (ISR) and *Final-State Radiation* (FSR) add more particles to the simulated process. It is known that the main contribution to the UE arises from the extra partons scatterings occurring along with the primary hard scattering, these extra scatterings are called *Multi Parton Interactions* (MPI). MPI are soft or semi-hard parton-parton scatterings that add more particles to the final-state.

Usually, the observables sensible to the underlying event receive contribution also from the main hard scattering. So, a good description of UE observables requires a good modeling of MPI and Beam-Beam Remnants (BBR), which are all the partons left unscattered, and also of the hadronization process, ISR and FSR. The existing MC generators, as pythia [5], describe all these processes and have adjustable parameters to control the behavior of event modeling. The set of parameters that best describe the fitted distributions is referred to as *tune*. In 2020 the CMS Collaboration has published a new set of tunes for the UE simulations [6] in PYTHIA8 event generator. The tunes are obtained from data measurements sensitive to soft and semi-hard MPI at different collision energies.

One of the main problems in the tuning of these event generators is that the tuning procedure can be really computational expensive, due to the high time required to run a MC generator, and to perform a tune it is required to run a generator a very large amount of times. So, to reduce the number of runs required the procedure usually taken for the tuning is a *parameterization-based approach*. The approach consists of fitting the histograms that are generated from these Monte Carlo simulations with different values for the input parameters under analysis and studying the variation by defining a function that describes the bins' behavior to the variation of the parameters. These procedure is referred as as *fitting the generator response*.

This master thesis, it is investigating this tune aspect focusing on the tuning of parameters for the MC generator PYTHIA8. The work is developed within the CMS experiment framework on data collected in minimum bias observation and Z boson production events. In particular, the description of the above mentioned UE. The tuning is performed using MCNNTUNES [7] which implements a machine learning approach by means of Feed Forward Neural Networks (FFNNs).

In the thesis, it is validated MCNNTUNES against the already existing CMS tune for the underlying event: CP5 [6]. The validation is performed employing data from CDF and CMS analyses at different center-of-mass energies: 1.96, 7, 13 TeV and distributions of charged particle density and charged particle transverse momentum sum in the transverse regions and pseudorapidity distributions of charged hadrons production, single and non-single diffractive events. For the description of these distributions a correct tuning for the parameters related to MPI and Color Reconnection (CR) is important.

Once the tool has been validated. It is used to tune the primordial k_T and the ISR. In this step, it is investigated the effects of the two PYTHIA8 parameters `BeamRemnants:primordialKThard` and `SpaceShower:pt0Ref` in the description of the low region in the spectra of Drell Yan observation in Z boson production events.

Chapter 2 is going to introduce the theoretical aspects of the description of a hadron-hadron collision. It has to be seen as a general view on all the processes that one have to take into account to have a proper MC simulation of an hadron-hadron collision. The chapter starts with the description of hard events using perturbation theory: factorization theorem. Then the description moves on to the introduction of the parton shower algorithm that links the hard scale of the hard interaction to the softer scale of the hadronization. It has been introduced that the UE description cannot avoid the simulation of the Multi Parton Interaction, a detailed introduction on this is given. Some other important processes are described at the parton level, e.g. the introduction of the primordial k_T . In the end of the chapter an introduction to the hadronization process is given. Along with the chapter a particular focus is given to the main parameters that are going to be used in the tuning of the underlying event in minimum bias observation and in Z boson production spectra observed in Drell Yan events. All the observables used in the tunes presented in this thesis are described in Chapter 3 with an introduction on the CMS detector and on the existing CMS tunes called CP and a number from 1 to 5 [6] that are the starting point for our work. Chapter 4 describe the basic concepts in the tuning procedure, in particular, it focuses on the parametrization-based approach used for the tuning, in this chapter, it is also discussed the main tool that will be used called MCNNTUNES [7] which implements a tuning method based on machine learning.

The MCNNTUNES tunes performed for the underlying event and for the Z boson spectrum are described respectively in Chapter 5 and Chapter 6.

Chapter 2

Hadron-Hadron Scattering

In a high energy proton-proton collision we can have either soft or hard processes. Most of the time the hard processes are accompanied by soft interactions, occurring along the hadron interaction. While the hard processes as the Higgs boson production, or the high p_T jet production are well understood using perturbative QCD, the soft processes as the underlying event, the hadronization are not so well understood. In fact, these processes involving a low energy scale at which perturbative series expansion of QCD breaks down, are studied with non perturbative approaches. The introduction of the MC simulation is required in order to have a complete description of this two class of processes and also of they interconnection by means of the parton showering process.

This chapter in the first part focus on a theoretical introduction on some basic elements of perturbative QCD theory: the QCD factorization theorem, the fixed-order approach in perturbative QCD calculations for the partonic cross-section and the PDF.

In the second part, it is described the parton shower where the parton emission is describe in term of splitting probability for the Initial State Radiation (ISR) and Final State Radiations (FSR). Then, the effects of Multi Parton Interactions and other important processes are described in the simulation of High Energy Physics events. lastly a look on the hadronization process is given.

2.1 Structure of Monte Carlo event generators

Monte Carlo event generators build up the structure of a hadron-hadron collision in the different steps. Firstly, a primary hard subprocess is generated with the parton shower associated to the incoming and outgoing coloured participants. Then, all the non-perturbative interactions that convert the showers into outgoing hadrons and connect them to the incoming beam hadrons, also the extra interactions that give rise to the UE are simulated. Lastly all the processes at hadron level take part in the simulation, e.g. the decays of unstable particles. A schematically representation of all these processes is shown in Fig. 2.10.

Usually, one is interested in simulation of events of particular type. The strategy followed is to select the type of hard process and generate the event on the basis of this selection, rather than simulating all type of events and select only the one of interest, in fact this would leads to a very low statistics in case of rare events. Due

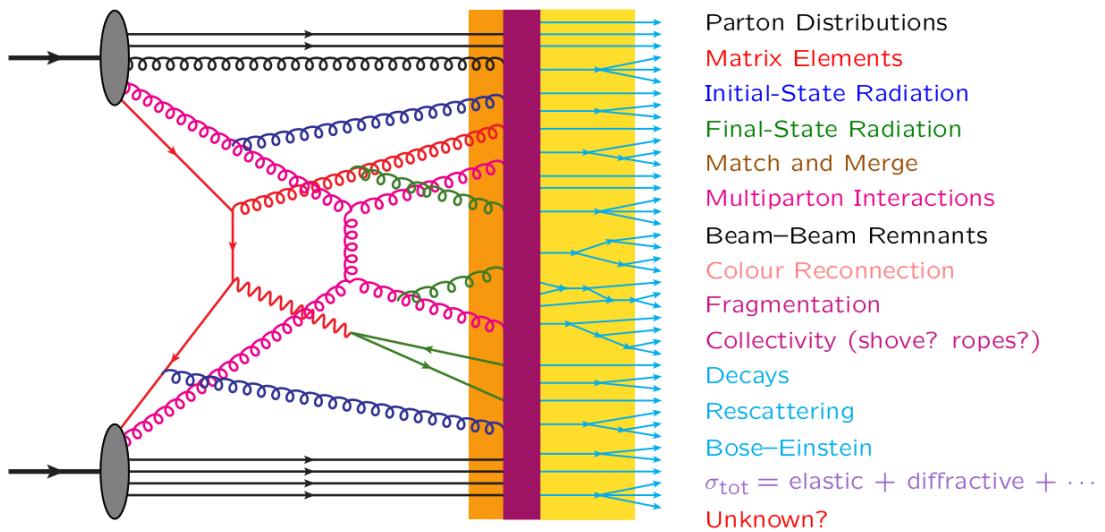


Figure 2.1: A schematic representation for a pp collision. Reading the image from left to right one can have an idea on the evolution of the system. The two incoming hadrons enter the scattering from the left side, the red line indicate the main hard scattering and the magenta one the second parton scattering (MPI) each interaction is associated with initial (blue) and final (green) state radiation, the unscattered partons (black lines) re-enter the color reconnection and hadronization processes. Then the new formed hadrons (lightblue) can undergo to different decays.

to the fact that partons entering in the hard scattering are coloured they can radiate gluons, this gluon can radiate other gluons or produce quark-antiquark couples, and so on generating showers of outgoing particles. This is simulated by a Markov chain used to choose if add one more parton to the final-state at a certain time. This is called the parton shower algorithm and as will be discussed below it is formulated as an evolution in some momentum-transfer-like variable downward evolution starting from a hard scale defined by the main hard interaction, and as both a forwards evolution of the outgoing partons and a backwards evolution of the incoming partons progressively towards the incoming hadrons.

To be notice that, incoming hadrons are composite particle, i.e. bounded state of partons. So, it is likely to have more than one interaction between partons in the same hadron-hadron collision. Also these interactions are simulated according to the momentum-transfer-like variable downward evolution.

As the evolution go on is reached the region of order 1 GeV, in which QCD became strongly interacting and perturbation theory breaks down. At this scale the perturbative description in terms of partons must be terminated and replaced by a non-perturbative hadronization models in order to describe the colour confinement. Unluckily, these models are not derived directly from QCD and so have more free parameters than the preceding ones. But, to a good approximation they are universal so once they have been tuned to a data set they can be used to predict new collision data.

2.2 QCD factorization theorem

A large amount of interesting processes at LHC involve an high transferred-momentum scale, e.g. the production of heavy particles or jets with high transverse momentum. The fundamental idea for the description of hard processes in a hadron-hadron collision is that hadrons are made-up from partons that, at some high energy scale, can be seen as free. So the hadron-hadron interaction can be seen as an interaction between two free partons, if is given a sufficiently high energy scale for the interaction.

This idea was developed in the framework of the deep inelastic scattering, and the Bjorken scaling observation confirms it [8]. The generalization of this concept leads to the *QCD factorization theorem*.

The factorization theorem was introduced first by Drell and Yan [9]. The hadron-hadron scattering cross-section is described in terms of partons extending the formalism used for deep inelastic scattering, through the following factorized formula:

$$\sigma_{AB} = \int dx_a dx_b f_{a/A}(x_a, Q^2) f_{b/B}(x_b, Q^2) \hat{\sigma}_{ab \rightarrow X} . \quad (2.1)$$

Where X is a partonic/leptonic state and $a(b)$ a quark or an antiquark in the hadron $A(B)$, e.g. the proton. The σ_{AB} is the hadronic cross-section while $\hat{\sigma}_{ab \rightarrow X}$ is the partonic cross section that can be calculated using perturbative QCD. Finally, the $f_{a/A}$ ($f_{b/B}$) is the parton distribution function in the hadron $A(B)$. PDF is dependent on x_a (x_b) that is the light-cone momentum fraction carried by the parton with respect to the total momentum of the parent hadron $A(B)$; and on the scale of the interaction Q^2 .

Let describe all the contribution to this formula in more detail one by one.

2.2.1 Parton distribution functions

The first ingredient in our recipe is the knowledge of the quark and gluon distributions inside the two hadrons that undergo the scattering.

The PDFs used to describe a hard scattering are solution to the DGLAP (Dokshitzer–Gribov–Lipatov–Altarelli–Parisi) equations [10, 11, 12, 13]

$$\mu_F^2 \frac{\partial f_{i/H}(x, \mu_F^2)}{\partial \mu_F^2} = \sum_j \frac{\alpha_s(\mu_F^2)}{2\pi} \int_x^1 \frac{dz}{z} P_{i \rightarrow j}(z) f_{j/H}\left(\frac{x}{z}, \mu_F^2\right) , \quad (2.2)$$

where μ_F is the energy scale known as *factorization scale*, and $f_{i/H}(x, \mu_F)$ the PDF for the parton of type i inside the hadron H . The PDF is a function of the momentum fraction x and of the energy scale. While, $P_{i \rightarrow j}$ are the so-called *splitting functions*: they are the probability that a parton of type i emits a parton j (a quark or a gluon) carrying a fraction z of the i -parton momentum.

The splitting functions have the following perturbative expansions in α_s :

$$P_{i \rightarrow j}(x, \alpha_s) = P_{i \rightarrow j}^{(0)}(x) + \frac{\alpha_s}{2\pi} P_{i \rightarrow j}^{(1)}(x) + \dots . \quad (2.3)$$

The information on the quark distribution inside a hadron $f_{q/p}(x, Q^2)$ arises from lepton-hadron DIS experiments, from lepton-pair production in hadron-hadron

collisions (Drell-Yan processes) and jet measurements to study gluon distribution $f_{g/p}(x, Q^2)$. All these quantities are the experimental input in order to evaluate the PDF inside the hadron while the Q -evolution is described by DGLAP equation as discussed above.

A lot of processes are available for the PDFs evaluation and a lot of PDF set have been generated, as an example Fig. 2.2 shows the NNPDF3.1 set [14] at NNLO for a virtuality $Q^2 = 10 \text{ GeV}^2$ (left) and $Q^2 = 10^4 \text{ GeV}^2$ (right). Note that the gluon

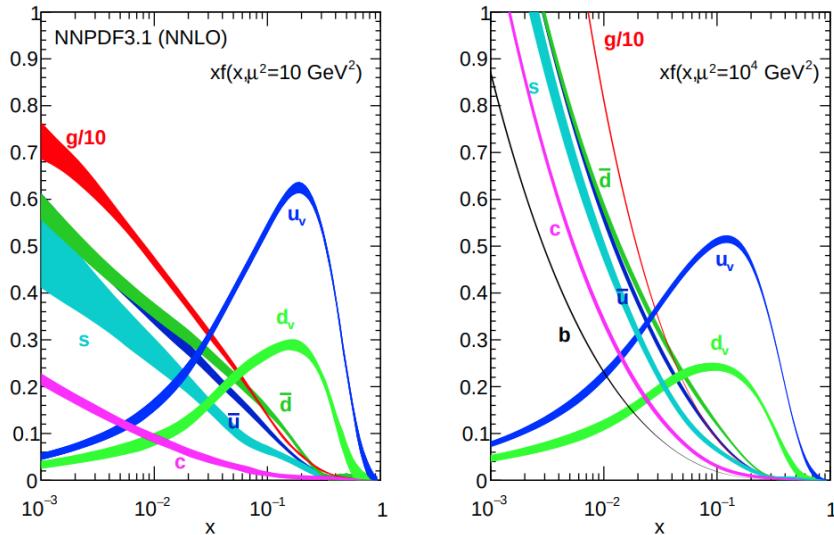


Figure 2.2: The NNPDF3.1 NNLO PDFs set, evaluated at $Q^2 = 10 \text{ GeV}^2$ (left) and $Q^2 = 10^4 \text{ GeV}^2$ (right). At low x the contribution from the gluons is the dominating one while at higher x the dominant contribution is from the valence quarks. The PDF Q evolution shows that when our proton is probed to higher Q^2 the resolution increase (higher Q correspond to smaller distance resolution) and so we have a bigger contribution from the sea quarks at low x values.

contribution have been scaled of a factor 10: in fact, in the low x region, $x < 0.01$, the gluon contribution is the dominating one, while at high x value the valence quarks dominate the PDF.

In Fig. 2.2 we can also see that with increasing virtuality (Q^2) at low x the density of the sea quarks increases: this is related to the fact that our hadrons are probed at higher energy and the probe resolution is proportional to the energy.

$$\text{Resolution} \sim \frac{\hbar}{Q} . \quad (2.4)$$

So, when probed at higher energy, the hadrons appear denser¹ than when are probed at lower energy. This, as will be discussed in the below sections, is related to the energy dependence in the number of interaction between partons in a single hadron-hadron collision. The phenomenon of having more than one interaction between partons in a single hadron-hadron collision is called *multiple parton interaction*: this concept will be discussed in the next chapter.

¹This density growth is not endless but at some point a saturation is reached it is called *parton saturation*. The parton saturation find an explanation in the observation that at some high energy scale the parton mergers have to eventually compensate for splittings.

2.2.2 Partonic cross-section

Partonic cross section is another fundamental ingredients in our recipe for the description of hadron-hadron interactions. It can be calculated as a perturbative expansion in α_s from QCD first principles using quantum field theory. So our hadron-hadron cross-section can be written as a sum of different contributions, as shown in the following formula:

$$\sigma_{AB} = \int dx_a dx_b f_{a/A}(x_a, \mu_F^2) f_{b/B}(x_b, \mu_F^2) [\hat{\sigma}_0 + \alpha_s(\mu_R^2) \hat{\sigma}_1 + \dots] . \quad (2.5)$$

Where we have introduced the dependencies on the two unphysical scales:

- The *factorization scale* μ_F : this scale is related to the resolution with which the hadron is being probed, separates long- and short- distance physics processes.
- The *renormalization scale* μ_R : the scale at which is evaluated the strong coupling constant α_s . The dependence of α_s on the renormalization scale is related to different effects such as the vacuum polarization, the quark self-energy, the vertex corrections, and the gluon loop corrections to the elementary three-gluon and four-gluon couplings.

The calculation of such expansion at the leading order (LO) is performed by evaluating all the possible tree-level Feynman diagrams for every process. Then the calculation proceeds by computing the squared matrix element and by integrating it over the available phase space (analytically or numerically).

At this point, we can already encounter some divergence that can be avoided imposing some restrictions on the phase space.

Higher order calculations

The LO calculation can describe broad feature of a particular process and provide a first estimation of its cross section; anyway, in many cases this is insufficient.

The main source of uncertainty derives from the LO dependence on the unphysical renormalization and factorization scales. Some process may contribute only when going beyond the first approximation, and some divergence can be resummed.

The next-to-leading order (NLO) calculation requires all the Feynman diagrams that take an extra α_s . This contribution can arise in two different ways:

- **Virtual**: internal lines in the Feynmann diagram, see Fig. 2.3a, that don't originate a real particle in the initial or in the final state (loops);
- **Real**: external lines int the Feynmann diagram, see Fig. 2.3b, here the particle is real and participate to the initial or final state observed (real particles).

Virtual corrections contains infrared divergences, arising by integrating on the loop circulating momentum, that cancel against infrared singularities given by collinear or soft emissions [15, 16, 17].

A common strategy for the renormalization is dimensional regularization: it consists into performing the calculation in a $D = 4 - 2\epsilon$ dimensional space ($\epsilon < 0$); in that way the singularities appear as single and double poles in ϵ . Then, the limit $\epsilon \rightarrow 0$

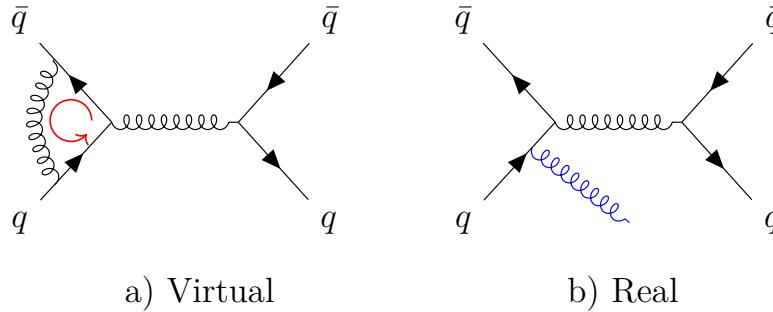


Figure 2.3: Feynman diagrams for a virtual correction (a) the momentum circulating in the loop when integrated on the phase space take to a divergence; and a real correction (b) the particle emitted is real, participate to the initial or final state of the process.

is taken after the divergences have cancelled.

This NLO calculation with regularization allows to extend the treatment of these integrals up to zero transverse momentum.

The importance of higher-order calculation is that it allows too describe more processes, as can be shown with the following example. In a Z boson production:

- 1) **LO:** the Z is produced without transverse momentum (p_T), and anything can recoil against the Z for momentum conservation (Fig. 2.4a).
- 2) **NLO:** the Z acquires a finite p_T . In this case the Z boson p_T is balanced by a single parton/gluon (Fig. 2.4b).
- 3) **NNLO:** the Z p_T can be balanced by two jets (Fig. 2.4c).

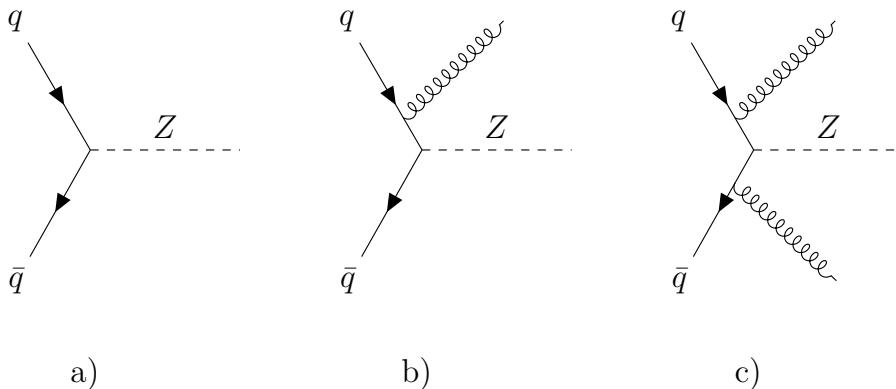


Figure 2.4: Feynman diagrams for the Z production by annihilation of a quark and an antiquark at LO (a), NLO (b), NNLO (c). At LO the Z can only be produced with a $p_T = 0$ for the conservation of the momentum.

Another important benefit of performing a NLO calculation is the reduction of the dependence of calculation on the unphysical renormalization (μ_R) and factorization (μ_F) scales. It is proven that higher order calculations of observables calculated to order α_s^n are dependent upon the unphysical scales only at order higher than α_s^{n+1}

[18]. The range of predictions corresponding to different scale choices is usually attributed to *theoretical uncertainties*, as is shown in Fig. 2.5, where the uncertainties reduce from the LO calculation to the NLO and even more to the NNLO.

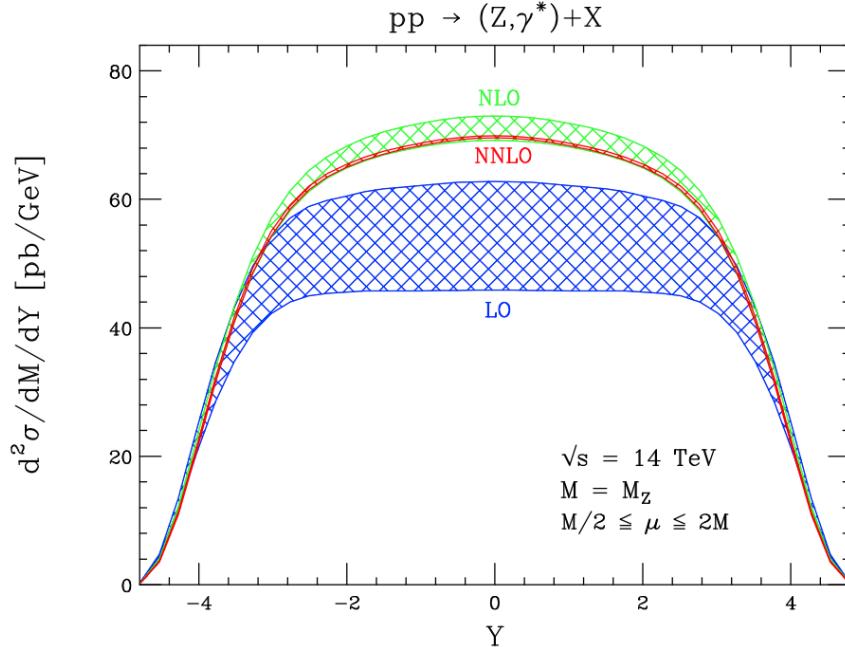


Figure 2.5: The rapidity distribution predictions at LO (blue) NLO (green) and NNLO (red) for the Z production at the center of mass energy $\sqrt{s} = 14$ TeV. The band width is related to the uncertainties. Going from LO to NLO there is a increase in the cross section prediction and a reduction on the scales uncertainties, the NNLO prediction is in the NLO error band but there is a further increase in the precision of the prediction. Figure from [18], section 6.

2.3 All-order approaches and PYTHIA Monte Carlo generator

A different approach to describe the phenomena observed at high-energy colliders, instead of calculating cross sections order by order in the perturbative expansion, is the use of an *all-order* approach.

The most violent proton-proton collision at the LHC can have an high number of separated jets, $\sim 5 \div 10$. If these jets are observed from a nearest point of view what one can observe is a fractal composition of jets-inside-jets-inside-jets. This structure is expected to continue down to the hadronization scale, a bit below 1 GeV. This fractal nature can lead to hundred of partons in the final-state that then are masked inside the hadrons. The main problem in the theory is that there is no way to perform matrix element calculation to describe such complicated event topology. Instead, the standard procedure is to use an *all-order* approach: *parton showers*.

2.3.1 Parton Showers

The parton shower algorithm starts from few partons arising from a hard interaction evaluated by means of the matrix element calculation. Then, as the energy scale at which we examines the scattering decrease these harder partons can split by emission of gluons or quark anti-quark pairs, and so more partons are produced in the shower that can split in their turn. In this process the higher scale partons are related to the lower hadronization energy scale (close to 1 GeV) using the DGLAP evolution equation formalism. The solution to this equation can be written using a *Sudakov form factor* arising from the probability of no gluon emission in the evolution from higher to lower scale and ensure the unity for the branching total probability.

In the parton showering process, in addition to the kinematic variables (momentum fraction z and azimuthal angle ϕ) and flavours of the partons, an evolution variable t is generated. PYTHIA8 uses as evolution variable the squared of the relative transverse momentum of the two partons in the splitting (p_T^2). Different choices are made in HERWIG and SHERPA.

As mentioned before, the shower evolution is based on the standard (LO) DGLAP splitting kernels $P(z)$ described here:

$$P_{q \rightarrow qg}(z) = C_F \frac{1+z^2}{1-z} ; \quad (2.6)$$

$$P_{g \rightarrow gg}(z) = C_A \frac{(1-z(1-z))^2}{z(1-z)} ; \quad (2.7)$$

$$P_{q \rightarrow q\bar{q}}(z) = T_R(z^2 + (1-z)^2) ; \quad (2.8)$$

where $C_F = \frac{4}{3}$ is the Casimir operator for $SU(3)$, $C_A = N_C = 3$, that are named "color factors", and $T_R = \frac{1}{2}$ that is given by the trace calculation of the group generators, each contribution is multiplied by N_f if summing over all contributing quark flavours.

The parton shower consist of two components: the initial-state radiation (ISR) describing the emission from the incoming partons and the final-state radiation (FSR) describing emission of outgoing partons they are also respectively referred as spacelike and timelike showers. Both ISR and FSR algorithms are based on these splitting kernels introduced in Eq. 2.6, 2.7 and 2.8. The respective probabilities of emitting radiation as one moves in the decreasing evolution variable sequence are:

$$FSR : \quad \frac{d\mathcal{P}_{FSR}}{dp_T^2} = \frac{1}{p_T^2} \int \frac{dz}{z} \frac{\alpha_s}{2\pi} P(z) ; \quad (2.9)$$

$$ISR : \quad \frac{d\mathcal{P}_{ISR}}{dp_T^2} = \frac{1}{p_T^2} \int \frac{dz}{z} \frac{\alpha_s}{2\pi} P(z) \frac{f'(x/z, p_T^2)}{f(x, p_T^2)} . \quad (2.10)$$

Where the FSR one represent the infinitesimal probability that a parton branches, or in this case decays, in a dp_T^2 step. Instead, Eq. 2.10 refers to the same probability but it include the description of the PDF evolution in fact the ISR is described in terms of production of a parton (instead of decays probability as in FSR). The slitting kernel are the same described above except that the production of a gluon $P_g(z)$ is equal to $2P_{gg}(z)$ since two gluons are generated from the decay of one gluon.

Staring from these infinitesimal probability by multiplication of no emission probability $1 - dP_{ISR}$, or equally $1 - dP_{FSR}$. The Sudakov form factor can written-out,

as:

$$\Delta(p_T^2) = \exp \left(- \int_{p_{T0}^{PS}}^{p_T} \frac{d\mathcal{P}_{PS}}{dp_T^2} dp_T \right) \quad \text{with } PS = ISR, FSR . \quad (2.11)$$

The Sudakov form factor give the probability of a parton to evolve from an harder scale to a softer scale without emitting a parton harder than some resolution scale. The introduction of the Sudakov form factor resums all the effects from the soft and collinear gluon emission. For more details and some plots of different Sudakov form factor values see section 3.5 of [18].

In PYTHIA8 the contributions from ISR, FSR are interleaved into a single common sequence of decreasing p_T .

It has been seen that the solution to the DGLAP equation is given putting Eq. 2.9 and Eq. 2.10 in Eq. 2.11 respectively for the ISR and the FSR by a Sudakov form factor that is related to the no emission probability in the p_T -evolution.

The evolution variables for ISR and FSR are defined starting from the virtuality (Q^2) of the emission:

$$p_T^2 = \begin{cases} (1-z)Q^2 & ISR \\ z(1-z)Q^2 & FSR \end{cases} , \quad (2.12)$$

so, in the two cases: for the FSR $Q_{FSR}^2 = (p^2 - m_0^2)$ in fact a time-like virtuality ($p^2 > 0$) is implicated, while for the ISR we have $Q_{ISR}^2 = (-p^2 + m_0^2)$ with a space-like virtuality ($p^2 < 0$). In both case Q^2 values are positive defined. Than the actual strong of the radiation is set by the choices of $\alpha_{s ISR}$ and $\alpha_{s FSR}$ values.

The threshold on the transverse momentum scale for ISR, p_{T0}^{ISR} , and FSR, p_{T0}^{FSR} , are free parameters in PYTHIA and are called `SpaceShower:pT0Ref` and `TimeShower:pT0min`.

The parton showers alone are certainly not the whole story in describing the exclusive structure of an event. They are based on collinear and soft approximation. A good description of the hard wide-angle emission and multi jet final state cannot be ignored in order to describe many of the observables of interest. So, it is important to perform an higher order matrix element calculations to describes better all these observables.

2.3.2 Merging parton showers and NLO matrix element calculations

Regions dominated by soft and collinear gluon emissions are described very well by parton showers approach; on the other hand, regions where partons are energetic and widely separated are well described by matrix element calculations and not by parton showers. So, the best approach would be to combine the two different descriptions, and this is what is performed in the Monte Carlo generators. But to combine NLO matrix element calculation with parton showers the naive procedure of simply adding a parton shower to an event generated with a matrix element generator does not work.

One problem arise by the fact that tree-level matrix elements are *inclusive*, i.e. they give the probability of having at least n partons in a state calculated exactly to lowest order in α_s , while the corresponding state generated by a parton shower

is *exclusive*, i.e. gives the probability that there are exactly n partons calculated approximately to all orders in α_s

The second problem is that to avoid the *double counting* of some regions of phase space or, conversely, the undercount other regions.

To solve these problems there are different strategies. The various strategies are divided in two group. The ones refered as "*match*" are those approaches in which high-order corrections to an inclusive process are integrated with the parton shower. The other strategy involves a "*merge*" this is done by defining a merging scale, where any parton produced above that scale is generated with a corresponding higher-order matrix element and any parton produced below is generated by the shower.

An example of merging approach is the FxFx merging scheme, developed by Frixione, Nason, Webber in the MC@NLO framework [19, 20, 21, 22]. In this scenario the risk of double counting is given by the fact that at NLO one emission can be made explicit as indicated in Fig. 2.6 by the red gluon line, then the progress of the parton shower can leads to a double counting between real emission matrix element and the parton shower as shown in Fig. 2.6, the double counting sources are indicated by the blue arrows.

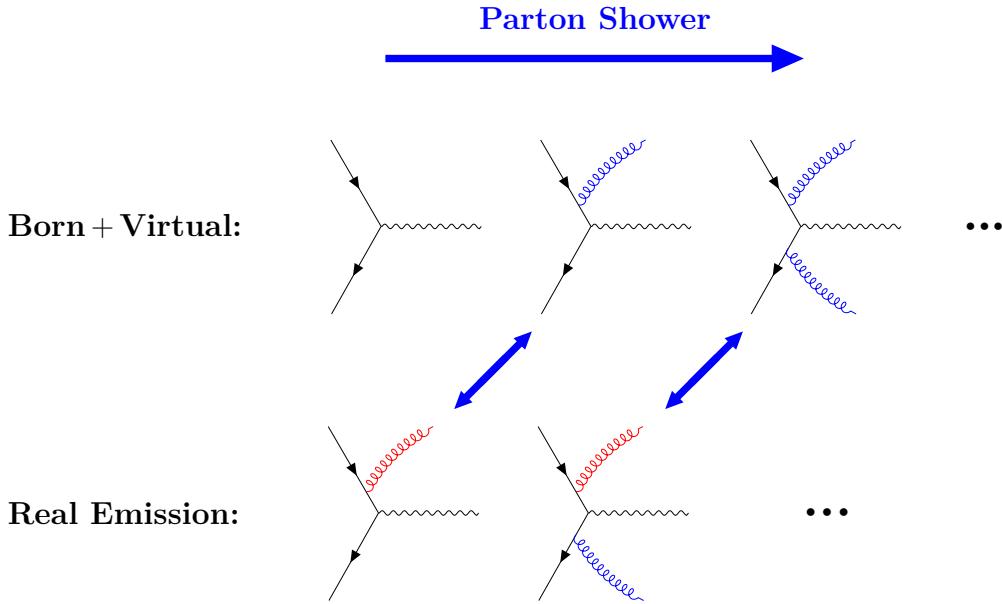


Figure 2.6: The FxFx margin scheme have to avoid this double counting. Feynman diagrams that can lead to a double counting are grouped with the violet arrow, the blue emission are related to the parton shower while the red ones to the NLO process.

So, it is clear that the matrix element calculations cannot be blindly combined with parton showers. The combination of these two approaches is a very active research topic, and is important for giving a good description for jet production from QCD.

2.4 Multiple Parton Interactions in PYTHIA

By the fact that the hadrons are viewed as "bunches" of partons, it is likely that in the same hadron-hadron collision different couples of partons can undergo to a scattering. This phenomenon is known as *Multiple Parton Interactions* and it is related to the composite nature of the incoming hadrons.

So, it is clear that at some level these MPI have to exist, and they can become important in the description of the event; they can change the color topology of the colliding system as a whole.

In this scenario it is important to have a good understanding of the phenomenon. The aim of this section is to describe the basic concepts that are used to simulate MPI, for example in PYTHIA Monte Carlo event generator; then some focus is given to the free parameters we are going to tune in the following sections.

2.4.1 Basic Concepts

The main hypothesis of the multiple interactions models is that: the QCD factorization theorem is true not only for the hard process but also for the other partons scatters.

So, we can write:

$$\frac{d\sigma_{\text{int}}}{dp_\perp} = \sum_{i,j,k,l} \int dx_1 \int dx_2 \int d\hat{t} f_i(x_1, Q^2) f_j(x_2, Q^2) \frac{d\hat{\sigma}_{ij \rightarrow kl}}{d\hat{t}} \delta \left(p_\perp^2 - \frac{\hat{t}\hat{u}}{\hat{s}} \right) . \quad (2.13)$$

This represent the interaction differential cross section for the hadron-hadron collisions, where $\frac{d\hat{\sigma}_{ij \rightarrow kl}}{d\hat{t}}$ is the differential cross section for QCD hard $2 \rightarrow 2$ processes, this processes are the one reported in Table 2.1.

In Eq. 2.13 we used the Mandelstam variables associated to the partonic system:

$$\hat{s} = (p_1 + p_2)^2 = (p_3 + p_4)^2 = x_1 x_2 s \quad (2.14)$$

$$\hat{t} = (p_1 - p_3)^2 = (p_2 - p_4)^2 \quad (2.15)$$

$$\hat{u} = (p_1 - p_4)^2 = (p_2 - p_3)^2 \quad (2.16)$$

where p_1, p_2 are the four-momenta of the incoming partons and p_3, p_4 the four-momenta of the outgoing partons.

We assume also that the "hardness" of processes is defined by the p_T scale of the interaction ($Q^2 = p_T^2$).

As you can see from the formulae in Table 2.1 at small scattering angles, for $t \rightarrow 0$, the t-channel gluon exchange processes $qq' \rightarrow qq'$, $qg \rightarrow qg$ and $gg \rightarrow gg$ dominate the full matrix element. For scatterings that are soft relative to \hat{s} , $|\hat{t}| \ll \hat{s}$, we can approximate $|\hat{t}|$ as:

$$p_T^2 = \frac{\hat{t}\hat{u}}{\hat{s}} = \frac{\hat{t}(-\hat{s} - \hat{t})}{\hat{s}} \approx |\hat{t}| , \quad (2.17)$$

in this limit, the only differences between quark and gluon cross sections are the color factors

$$\hat{\sigma}_{gg} : \hat{\sigma}_{qg} : \hat{\sigma}_{qq} = \frac{9}{4} : 1 : \frac{4}{9} . \quad (2.18)$$

Process	Amplitude	$\sum \mathcal{M} ^2 / (4\pi\alpha_s)^2$
$qq' \rightarrow qq'$		$\frac{4}{9} \frac{s^2 + u^2}{t^2}$
$qq \rightarrow qq$		$\frac{4}{9} \frac{s^2 + u^2}{t^2} + \frac{4}{9} \frac{s^2 + t^2}{u^2} - \frac{8}{27} \frac{s^2}{tu}$
$q\bar{q} \rightarrow q'\bar{q}'$		$\frac{4}{9} \frac{t^2 + u^2}{s^2}$
$q\bar{q} \rightarrow q\bar{q}$		$\frac{4}{9} \frac{s^2 + u^2}{t^2} + \frac{4}{9} \frac{t^2 + u^2}{s^2} - \frac{8}{27} \frac{u^2}{st}$
$q\bar{q} \rightarrow gg$		$\frac{32}{27} \frac{t^2 + u^2}{tu} - \frac{8}{3} \frac{t^2 + u^2}{s^2}$
$gg \rightarrow q\bar{q}$		$\frac{1}{6} \frac{t^2 + u^2}{tu} - \frac{3}{8} \frac{t^2 + u^2}{s^2}$
$qg \rightarrow qg$		$-\frac{4}{9} \frac{s^2 + u^2}{su} + \frac{s^2 + u^2}{t^2}$
$gg \rightarrow gg$		$\frac{9}{2} \left(3 - \frac{tu}{s^2} - \frac{su}{t^2} - \frac{st}{u^2} \right)$

Table 2.1: Parton-Parton differential cross sections ($2 \rightarrow 2$ QCD process), can be calculated in pQCD evaluating the matrix element for each process involving quark, antiquark and gluon. Table from [23]

So, the Eq. 2.13 can be rewritten like

$$\frac{d\sigma_{int}}{dp_T^2} \approx \int \int \frac{dx_1}{x_1} \frac{dx_2}{x_2} F(x_1, p_T^2) F(x_2, p_T^2) \frac{d\hat{\sigma}_{2 \rightarrow 2}}{dp_T^2} , \quad (2.19)$$

whit:

$$\frac{d\hat{\sigma}_{2 \rightarrow 2}}{dp_T^2} = \frac{8\pi\alpha_s^2(p_T^2)}{9p_T^4} ; \quad (2.20)$$

$$F(x, Q^2) = \sum_q (x q(x, Q^2) + x \bar{q}(x, Q^2)) + \frac{9}{4} x g(x, Q^2) . \quad (2.21)$$

Now, we can integrate the Eq. 2.19:

$$\sigma_{int}(p_{T min}) = \int_{p_{T min}^2}^{(\sqrt{s}/2)^2} \frac{d\hat{\sigma}_{2 \rightarrow 2}}{dp_T^2} dp_T^2 \propto \frac{1}{p_{T min}^2} \xrightarrow{p_{T min} \rightarrow 0} \infty \quad (2.22)$$

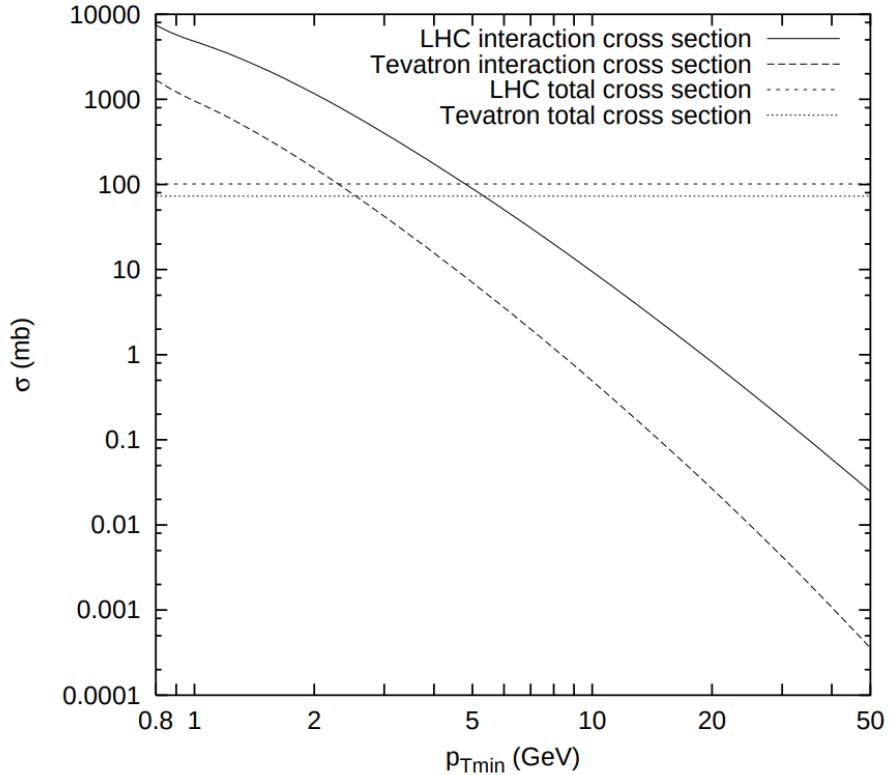


Figure 2.7: This figure shows the interaction cross section (σ_{int}) at Tevatron ($p\bar{p}$, $\sqrt{s} = 1.8$ TeV) and at LHC (pp , $\sqrt{s} = 14$ TeV). The flat lines are the corresponding values for the total cross section. The interaction cross section that arise from Eq. 2.22 is divergent for $p_{T min} \rightarrow 0$ in reality a dumping of this divergence is expected due to the color screening effect.

The total cross section is divergent in the limit $p_T \rightarrow 0$, this divergence is shown in Fig. 2.7. Due to this divergence the total interaction cross section at some p_T scale can exceed the total proton-proton cross section.

To understand this paradox should be noted that the interaction cross section described in Eq. 2.22 is related to the interaction probability between two partons and counts the number of interactions, while the total proton-proton cross section σ_{pp} counts the number of events. For example, an event (a proton-proton collision) in which two partons interact counts once in the total cross section and twice in the interaction cross section.

So, the ratio between this two quantities is perfectly allowed to be larger than unity, we can write it down as:

$$\frac{\sigma_{int}(p_{T\min})}{\sigma_{tot}} = \langle n \rangle(p_{T\min}) \quad . \quad (2.23)$$

Furthermore, we have to consider the *screening effect*: in fact the incoming hadrons are color singlet objects. Therefore, when the p_T of an exchanged gluon is small, and so the associated wavelength large, this gluon cannot longer resolve the color charges and the effective coupling is decreased, this screening set a cutoff in the divergence. The screening effect is schematically shown in Fig. 2.8.

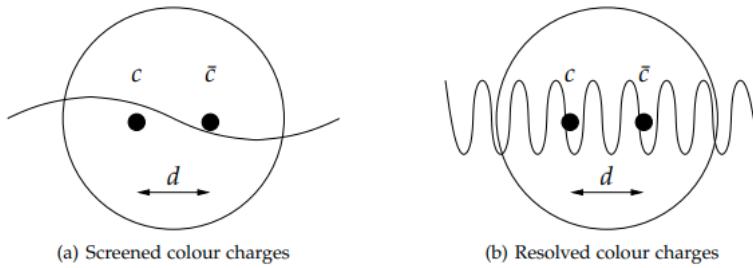


Figure 2.8: A picture of the screening effect. The left figure show two color charge that are not resolved by the gluon in fact the wave length is greater than the spatial separation, d , of the two charges. So resolution of the probe is not enough to discriminate the color charges. While on the right figure the two charges are very well distinct.

So, this cutoff is associated with color screening distance i.e. the average size of the region within the compensation of color charge occurs. This cutoff is then introduced in the factor as:

$$\frac{\alpha_s^2 (p_{T0}^2 + p_T^2)}{\alpha_s^2 (p_T^2)} \frac{p_T^4}{(p_{T0}^2 + p_T^2)^2} \quad . \quad (2.24)$$

This factor contains the phenomenological regularization of the divergence, with the factor p_{T0} that have to be tuned to data.

Now the interaction cross section is smoothly regularized and therefore finite.

To be notice that: parameter p_{T0} do not have to be energy-independent, but since the energy is related with the sensibility of our probe, higher energy is related with the capacity of probing PDFs to lower x values (as discussed previously, see Fig. 2.2) and in this low x region the number of partons rapidly increases. So, the partons are closer packed in this regions and as a consequence the color screening distance decrease.

The number of partons is related to x with a power law so, it is likely to have a

dependence of the same form for p_{T0} respect to the center-of-mass energy

$$p_{T0}(\sqrt{s}) = p_{T0}^{ref} \left(\frac{\sqrt{s}}{E_{CM}^{ref}} \right)^{E_{CM}^{pow}} . \quad (2.25)$$

These two parameters are named `MultiplepartonInteractions:pT0Ref` and `MultiplepartonInteractions:ecmPow` and control respectively the threshold for the MPI to occur at a fixed center-of-mass energy and the power evolution with \sqrt{s} .

In PYTHIA8 MPI, as said before, are also generated in a decreasing p_T sequence. So the hardest MPI is generated first. Than we can write the probability for an interaction, i , to occur at a scale p_T using a Sudakov-type expression (as we have done for the ISR and the FSR):

$$\frac{d\mathcal{P}_{MPI}}{dp_T} = \frac{1}{\sigma_{ND}} \frac{d\sigma_{2 \rightarrow 2}}{dp_T} \exp \left(- \int_{p_T}^{p_T^{i-1}} \frac{1}{\sigma_{ND}} \frac{d\sigma_{2 \rightarrow 2}}{dp'_T} dp'_T \right) . \quad (2.26)$$

Interleaving of Multiple Interaction and Parton Shower

As discussed above the ISR, FSR and MPI are simulated in PYTHIA following a common decreasing p_T evolution. So we have that the total probability is given from the composition of the various contributions from each process:

$$\begin{aligned} \frac{d\mathcal{P}}{dp_T} = & \left(\frac{d\mathcal{P}_{MPI}}{dp_T} + \sum \frac{d\mathcal{P}_{ISR}}{dp_T} + \sum \frac{d\mathcal{P}_{FSR}}{dp_T} \right) \times \\ & \times \exp \left\{ - \int_{p_T}^{p_T^{i-1}} \left(\frac{d\mathcal{P}_{MPI}}{dp'_T} + \sum \frac{d\mathcal{P}_{ISR}}{dp'_T} + \sum \frac{d\mathcal{P}_{FSR}}{dp'_T} \right) dp'_T \right\} \end{aligned} \quad (2.27)$$

In Fig. 2.9 are shown 4 parton-parton interactions with their associated showers (ISR and FSR). The downward evolution correspond to read the graph from top to bottom. The first hard interaction occurs at a scale $p_T = p_{T1}$ while the following ones at lower scales p_{T2}, p_{T3}, p_{T4} . Each interaction is associated with is radiation the first one occurs at $p_T = p'_{T1}$. The scatterings at p_{T2} and p_{T3} are originating from the same mother parton.

This diagram is related to one of the two hadrons. The full event can be illustrated if a similar diagram is drawn for the other hadron and connected to the full black circles.

2.4.2 Momentum and flavour conservation

One problem in MPI simulation is to achieve momentum conservation, so we need to take into account the modification in the PDF by the $i - 1$ interaction. To do that in PYTHIA the PDF are rescaled to the remaining available x range, adjusting their normalization.

We need to take into account the momentum fraction x_i removed from the hadron remnant by the $i - th$ interaction. This is done evaluating PDF not at x_i but at a rescaled value

$$x'_i = \frac{x_i}{X} \quad \text{with} \quad X = 1 - \sum_{j=1}^{i-1} x_j . \quad (2.28)$$

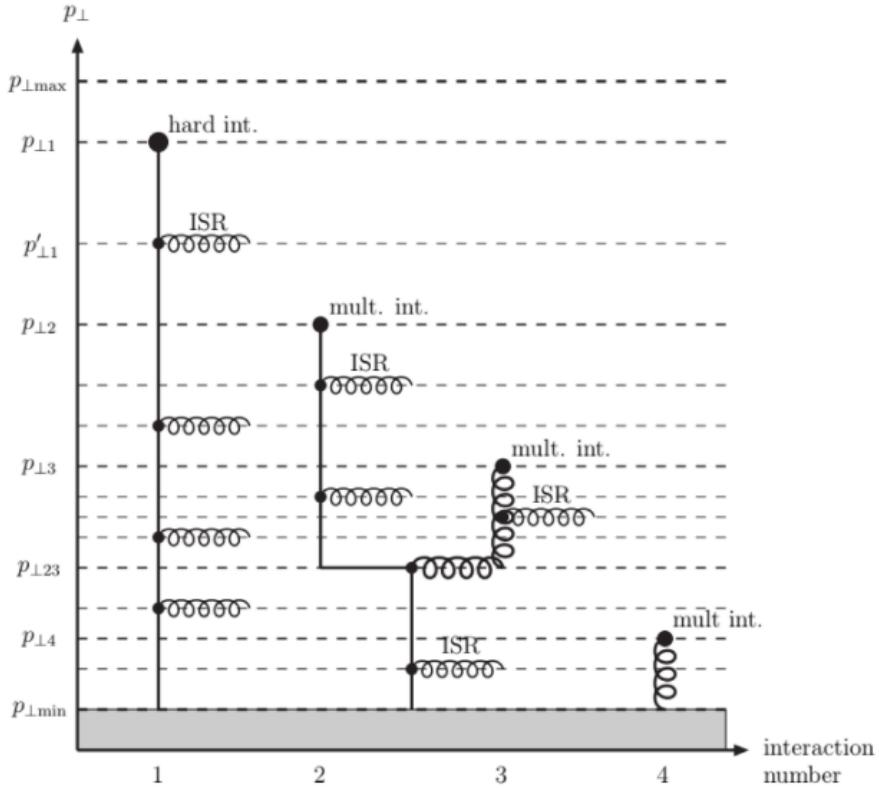


Figure 2.9: An example of Multiple Partons Interactions and associated radiations. The downward evolution in p_T is shown here by reading the graph from top to bottom. The 4 parton-parton interactions occur to p_{T1} , p_{T2} , p_{T3} and p_{T4} .

So, using these quantities, we can rewrite our PDFs as:

$$f_i(x, Q^2) \longrightarrow \frac{1}{X} f_0\left(\frac{x}{X}\right) , \quad (2.29)$$

where f_0 is the original one-parton inclusive PDFs.

Now, requiring also the flavour conservation and taking into account the number of valence and/or sea quarks involved in the preceding MPI. We have now the full forms of the PDFs used for the i -th MPI:

$$f_i(x, Q^2) = \frac{N_{fv}}{N_{fv0}} \frac{1}{X} f_{v0}\left(\frac{x}{X}, Q^2\right) + \frac{a}{X} f_{s0}\left(\frac{x}{X}, Q^2\right) + \sum_j \frac{1}{X} f_{c_j 0}\left(\frac{x}{X}, Q^2\right) , \quad (2.30)$$

$$g_i(x, Q^2) = \frac{a}{X} g_0\left(\frac{x}{X}, Q^2\right) , \quad (2.31)$$

where:

- $f_i(x, Q^2)$ ($g_i(x, Q^2)$) are the squeezed PDFs for quarks (gluons);
- N_{fv} the number of remaining valence quarks of the given flavour;
- N_{fv0} the number of original valence quarks of the given flavour (for the proton we have $N_u = 2$, $N_d = 1$);
- f_{s0} the sea-quark PDF;

- f_{cj} the companion PDF, this arise from the splitting $g \rightarrow q\bar{q}$ and a quark j is kicked out.

The factor a is defined to satisfy the total momentum sum rule.

2.4.3 Impact Parameter Dependence

The simplest hypothesis for the multiple interaction simulation, it is to assume the same initial state for all hadron collisions without dependencies on the impact parameter.

The more realistic scenario it is to include the possibility that each collision could be characterized by a different impact parameter b , where a small b value correspond to a large overlap between the two hadrons this is related to the probability of parton-parton interaction to take place.

To include the impact parameter dependence on the collision, it is necessary to make some assumption on the matter distribution inside the proton. A possibility is to assume a spherically symmetric distribution inside an hadron at rest $\rho(\mathbf{x}) d^3x = \rho(r) d^3x$. A Gaussian ansatz is the most simple choice but it appears to lead to a narrow multiplicity distribution and a too little pedestal effect. So the coiche is a double Gaussian:

$$\rho(r) \propto \frac{1-\beta}{a_1^3} \exp\left\{-\frac{r^2}{a_1^2}\right\} + \frac{\beta}{a_2^3} \exp\left\{-\frac{r^2}{a_2^2}\right\} , \quad (2.32)$$

where a fraction β of matter is contained in a radius a_2 , and the rest in a larger radius a_1 . in PYTHIA8 these two parameters are named respectively as `MultipartonInteractions:coreFraction` and `MultipartonInteractions:coreRadius`.

Now, for a given matter distribution $\rho(r)$, the time-integrated overlap function of the incoming hadrons during the collision is given by:

$$\mathcal{O}(b) = \int dt \int d^3x \rho(x, y, z) \rho(x + b, y, z + t) . \quad (2.33)$$

Assuming the matter distribution function in Eq. 2.32 and inserting it into Eq. 2.33 we get the following expression.

$$\mathcal{O}(b) \propto \frac{(1-\beta)^2}{2a_1^2} \exp\left\{-\frac{b^2}{2a_1^2}\right\} + \frac{2\beta(1-\beta)}{a_1^2 + a_2^2} \exp\left\{-\frac{b^2}{a_1^2 + a_2^2}\right\} + \frac{\beta^2}{2a_2^2} \exp\left\{-\frac{b^2}{2a_2^2}\right\} \quad (2.34)$$

this is useful to quantify the effect of overlapping protons. The larger is $\mathcal{O}(b)$ the more probable are parton-parton scatters between the incoming protons $\langle \tilde{n} \rangle \propto \mathcal{O}(b)$.

So, these assumption change the so-far Poissonian nature of the framework²

$$\mathcal{P}(\tilde{n}) = \langle \tilde{n} \rangle^{\tilde{n}} \frac{e^{-\langle \tilde{n} \rangle}}{\tilde{n}!} \quad (2.35)$$

²the Poissonian distribution, in Eq. 2.35 describe the probability of having n interactions at each impact parameter. If the matter distribution have an infinite tail (like our in Eq. 2.32) events may be obtained with arbitrarily large b values. This can be a problem for the definition of the total hadron-hadron cross section

Now the request that at least one parton interactions in the hadron-hadron collision, ensures that we get a finite total cross section. So the probability that an event is produced by two hadrons scattering with impact parameter b :

$$\mathcal{P}_{\text{int}} = \sum_{\tilde{n}=1}^{\infty} \mathcal{P}_{\tilde{n}(b)} = 1 - \mathcal{P}_0 = 1 - e^{-\langle \tilde{n}(b) \rangle} = 1 - e^{-k\mathcal{O}(b)} \quad (2.36)$$

So we have now that the number of interaction per event is give by (for impact parameter b):

$$\langle n(b) \rangle = \frac{\langle \tilde{n}(b) \rangle}{\mathcal{P}_{\text{int}}} \quad (2.37)$$

so, this have modified the probability distribution of interactions number from a Poissonian to a narrower one at each b fixed.

2.4.4 Beam Beam Remnants and primordial k_T

What is left after that the evolution is end?

The evolution in p_T can create an arbitrary complicate final state. This final state contains contributions from the scattered and unscattered partons that don't enter the p_T evolution. The last are the so called *Beam Beam Remnants* (BBR). BBR take into account the number of valence quarks remaining and the number of sea quarks required for the overall flavour conservation.

To ensure momentum conservation the BBR have to take all the remaining longitudinal momentum that is not extracted by the MPI initiators.

Primordial k_T

We have considered only the longitudinal momentum. In a real scenario partons, inside the hadrons, are fermions. So, are expected to have a non zero initial transverse momentum arising from the Fermi motion inside the incoming hadrons. This is denoted as *primordial k_T* . This is different from the transverse momentum derived from DGLAP shower evolution or from the hard interaction.

Based on Fermi motion alone, one would expect a value of the primordial k_T of the order of:

$$k_T \simeq \frac{\hbar}{r_p} \approx \frac{0.2 \text{ GeV} \cdot \text{fm}}{0.7 \text{ fm}} \approx 0.3 \text{ GeV} \quad , \quad (2.38)$$

but, as an example, to reproduce the data for the p_T distributions of Z bosons produced in hadron–hadron collisions, one need a larger contribution. This phenomenon has not a satisfactory phenomenological explanation. Until a convincing explanation is found the idea is to consider an effective primordial k_T for the initiators larger than the one in Eq. 2.38.

In PYTHIA the primordial k_T is assigned to initiators sampling a Gaussian distributions for p_x and p_y independently with variable width σ

$$\sigma(Q, \hat{m}) = \frac{Q_{1/2} \sigma_{\text{soft}} + Q \sigma_{\text{soft}}}{Q_{1/2} + Q} \frac{\hat{m}}{\hat{m}_{1/2} + \hat{m}} \quad (2.39)$$

Where Q is the hard-process renormalization scale for the main hard process and the p_T scale for subsequent MPI. σ_{soft} , σ_{hard} are the minimum and maximum width, \hat{m} is the invariant mass, while $Q_{1/2}$ and $\hat{m}_{1/2}$ the halfway values between the two extremes.

In PYTHIA the maximum width parameters is called `BeamRemnants:primordialKT-hard`.

2.4.5 Color Reconnection

The last important step at parton level is the *color reconnection*. Color reconnection is motivated by the fact that MPI leads to different color strings. In the previous steps the planar limit of the QCD was assumed where $N_c \rightarrow \infty$. Now, moving to real case where $N_c = 3$ all this strings that can be overlapped in physical space can be reconnected. The basic idea is to reconnect strings in order to reduce the total string length; and thus the potential energy (Lund Model).

In PYTHIA8 the reconnection is performed giving to each system a probability to reconnect given by:

$$\mathcal{P}_{\text{rec}} = \frac{p_{T \text{ rec}}^2}{(p_{T \text{ rec}}^2 + p_T^2)} \quad p_{T \text{ rec}} = R \times p_{T0} , \quad (2.40)$$

the `ColorReconnection:range` R is a user-tunable parameter while p_{T0} is the same parameter defined in MPI simulation.

With this definition for the probability, it is clear that system with low p_T are more likely reconnected to other. This idea find is justification in the fact that lower p_T values correspond to larger spatial extension and so these strings have more chance to overlap with other and so to reconnect.

2.5 PYTHIA summary

To summarize PYTHIA8 is able to simulate high energy hadron-hadron collisions. The evolution of the system is simulated in a common decreasing- p_T sequence the master formula for the evolution is written in Eq. 2.27.

This evolution start from an hard scale that is the scale of the main parton hard scattering that is described by a fixed ME calculation, PYTHIA can be interfaced with external frameworks for the ME step as POWHEG and MAD-GRAPH5 AMC@NLO (some care have to be taken as described in Section 2.3.2). Then the parton shower is started with the simulation of ISR, FSR and MPI also the parton rescattering is allowed. Once the evolution is ended the color reconnection and the hadronization transform the partons in a set of final hadrons these hadron than decay and rescatters against each others before the detection.

A schematic view of these processes is shown in Fig. 2.10.

All these processes are describe mainly by phenomenological model, due to the not-known-by-first-principles soft QCD description. The use of these phenomenological models introduces lots of free parameters (some are been pointed out in this sections) that have to be tuned with data to give to PYTHIA the ability of simulates real events.

Next chapter will focus on the study of the underlying event and so on all the soft events related to an hard scattering in an hadron-hadron collision with the

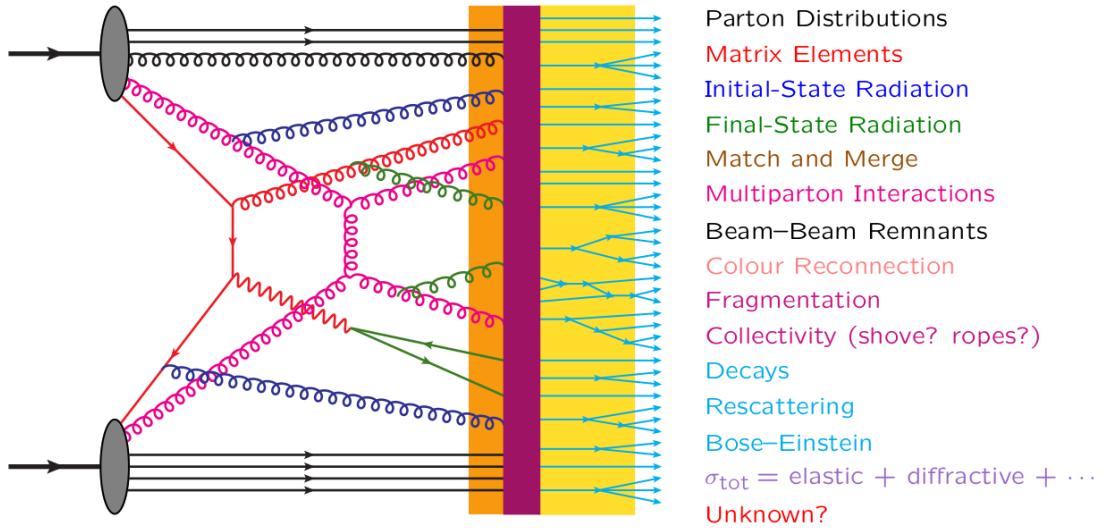


Figure 2.10: A schematic representation for a pp collision. Reading the image from left to right one can have an idea on the evolution of the system. The two incoming hadrons enter the scattering from the left side, the red line indicate the main hard scattering and the magenta one the second parton scattering (MPI) each interaction is associated with initial (blue) and final (green) state radiation, the unscattered partons (black lines) re-enter the color reconnection and hadronization processes. Than the new formed hadrons (lightblue) can undergo to different decays.

description of the analyses used in our tune and the effects of some of these PYTHIA8 parameters on these observables distributions.

Chapter 3

CMS Detector, Observable to Study the Underlying Event and Existing Tunes

The underlying events UE are all the processes not associated with the primary hard scatter in a hadron-hadron collision.

All the processes described in the previous section: ISR, FSR, MPI, and BBR and their interactions with color exchanges, contribute to the Underlying Event (UE) in the proton-proton collision.

As an example, in a proton-proton collision with a Z boson production, that then can decay in two leptons (l^+ and l^-) and in the end, these two leptons observed by the experiment, we have that the hard scattering is represented from the scatters between the two partons that generate the Z boson. While the so-called UE is represented from all the extra scatterings, the various radiations and in general all the activity not associated with this primary hard scattering.

It is important to underline the fact that most of the observables to study the UE are sensible only to the sum of these contributions and not to the single ones. So, a good description of all these processes and their interplay is really important in order to study the complex final states originating from these scatterings that contribute to the observables.

In this chapter, these observables sensible to the UE are introduced and described with more detail but first, a description of the detector used to take the measurement is necessary.

3.1 Compact Muon Solenoid Detector

This section will briefly describe the detector of the Compact Muon Solenoid experiment (CMS). CMS is an experiment operating at the Large Hadron Collider (LHC) at CERN. LHC is a particle accelerator designed to collide protons beams. It has operated to different center of mass energies (7, 8, 13 TeV) and now with the next run called RUN3 it will operate to the record energy of 13.6 TeV. LHC is a superconducting circular accelerator with a total length of 26.7 km. It is divided into 8 sections and 4 of these are occupied by the particle physics experiments in

CHAPTER 3. CMS DETECTOR, OBSERVABLE TO STUDY THE UNDERLYING EVENT AND EXISTING TUNES

particular one is reserved for the CMS detector.

The main feature of the CMS detector is a superconducting solenoid with a 6 m internal diameter that can provide a magnetic field up to 3.8 T. This very high magnetic field is necessary to bend the particles produced in the proton-proton interactions that take place along the beam pipe. So into the solenoid volume, there are different detectors: an all-silicon inner tracking system, a lead tungstate (PbWO_4) crystal electromagnetic calorimeter (ECAL), and hadron calorimeter (HCAL) composed by alternating layers of brass and scintillator material; both calorimeters are composed of a barrel, an endcap and a forward section. The outer region is covered by the muon detectors, these are composed of several layers of aluminium drift tubes in the barrel region, cathode strip chambers and resistive plate chambers in the endcap regions. An illustration is displayed in Fig. 3.1.

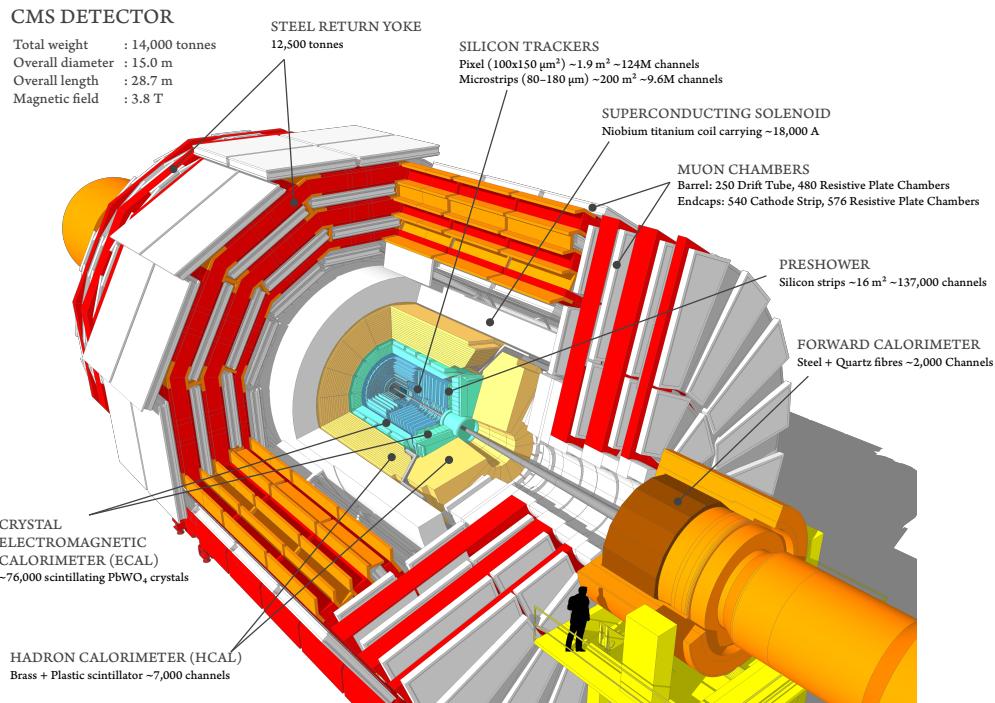


Figure 3.1: An illustration of the CMS detector [24].

The silicon tracker is responsible for the measurement of charged particle tracks within the pseudorapidity range $|\eta| < 2.5$. It is the most important detector for the studies of this thesis, in fact, all the examined distributions are collected by the analysis of charged particle tracks. It is obvious that a good resolution for the track reconstruction is required. The typical track resolutions, for non-isolated particles of $1 < pT < 10 \text{ GeV}$ and $|\eta| < 1.4$, are 1.5% in p_T . So, this inner tracking system is the central detector for the charged particles tracks reconstruction and, thanks to the high magnetic field, for the momentum measurements in CMS. The energy instead is measured by the two types of calorimeters: ECAL and HCAL.

The ECAL barrel cover the pseudorapidity range $|\eta| < 1.479$ it consists of about 61200 PbWO_4 crystal while the endcaps cover the pseudorapidity range $1.479 < |\eta| < 3.0$.

Instead, the HCAL barrel cover the pseudorapidity range $|\eta| < 1.3$ while the endcaps

$1.3 < |\eta| < 3.0$ these colorimeters are really important in the hadronic jet detection and reconstruction.

Another main feature of the CMS detector is obviously the muon system it has three main functions: muon detection, momentum measurement and triggering, totally the muon system covers the pseudorapidity up to $|\eta| < 2.4$

The LHC provide proton-proton collision at very high interaction rates, the beam crossing interval is 25 ns). So, it is impossible to store all the data, the rate is reduced in two steps: Level-1 (L1) Trigger and High-Level Trigger (HLT). These two steps perform a first events selection keeping only the interest interactions.

In the following, the selection criteria and the space topology for the analysis used for the tunes are introduced in order to better understand what kind of observables are studied in this thesis.

3.1.1 Selection Criteria

The selection criteria for CMS data from analysis at 13 TeV [25] are the selection of charged particle tracks within $|\eta| < 2$ while $|\eta| < 0.8$ for the analyses at 7 [26] and CDF one at 1.96 TeV [27]. The trigger sensibility imposes $p_T \geq 0.5$ GeV in order to ensure a good track reconstruction efficiency. Fake tracks after mis-reconstruction are removed imposing to pass the *highPurity* [28]. Secondary decays are removed by requiring the impact parameter significance $d_0/\sigma_{d0} < 3$ and the significance in the z-direction $d_z/\sigma_{dz} < 3$. Then to keep only tracks with good resolution the tracks with relative uncertainties $\sigma_{p_T}/p_T > 0.05$ are discarded.

To reconstruct the charged particle jets the clustering algorithm used is the Seedless Infrared-Safe Cone (SIScone) [29] jet algorithm¹. Charged particle within the pseudorapidity range $|\eta| < 2.5$ are used for jet reconstruction then the reconstructed jet have to pass the above selection criteria.

Minimum bias events are triggered by requiring activity in both Beam Scintillator Counters (BSC) that are located at a distance of ± 10.86 m from the detector center and are sensitive within $3.23 < |\eta| < 4.65$ in coincidence with signals from both beams in the Beam Pick-up Timing for eXperiments (BPTX) devices, they are located along the beam pipe at 175 m from the interaction point. These coincidences indicate that both the beams are crossing at the same time and they interact.

The selection criteria we are going to use for the pseudorapidity distributions from the analyses [31] and [32] are:

1. Online: coincident signals in both beams in the Beam Pick-up Timing for eXperiments (BPTX);
2. Offline: at least one vertex reconstruction.

The tracker measure all the charged particles within $|\eta| < 2.4$ and all the interaction points are used for the tracks reconstruction in the offline analysis.

Further details can be found in the respective paper for each analysis.

¹The Anti- k_T algorithm [30] is currently the more preferred jet clustering algorithm in CMS but to ensure continuity with respect to the previous analyses on the UE the one above is chosen.

3.2 Minimum Bias Measurements and Underlying Event topology

A Minimum Bias (MB) measurement is a collection of inelastic events with a loose event selection. This means the events are collected requiring the minimal interaction with the detector (the smallest possible bias). Most of the interactions in MB observations are soft, $p_T \lesssim 2$ GeV, but the study of the UE requires at least one hard scattering ($p_T \gtrsim 2$ GeV) presence; in fact, the UE is given by the underlying activity to a primary hard scatter.

To study the UE the topological structure of a hadron-hadron collision is used. The analysis is performed on an *event-by-event* basis. In the analysis, the direction of the leading object is used to define regions in the $\eta - \phi$ space. Where η is the pseudorapidity defined as $\eta = -\log \tan(\frac{\theta}{2})$, while ϕ is the azimuthal angle in the $x - y$ plane. The last one is defined from the direction of the leading object as $\Delta\phi = \phi - \phi_{\text{leading}}$.

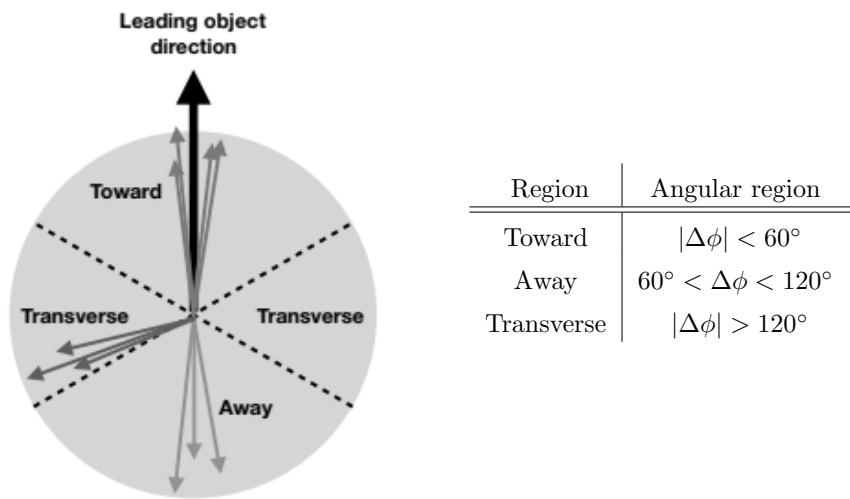


Figure 3.2: This figure shows the four regions for the description of the UE on an event-by-event analysis. The angular values, defining the four regions, are shown in the table. The regions are defined starting from the leading object direction. The toward and away regions contain most of the contribution from the hard scattering (e.g. in a $t\bar{t}$ production event the two quark t are located in these regions); while the transverse region are the ones in between the two other regions, these are the most important for the study of the UE.

The regions classification is shown in Fig. 3.2, we have:

- **Toward region:** the region that contains the leading object, this region contains most of the particles produced by the hard interaction.
- **Away region:** this region contains the objects that recoil against the leading object, also this region contains mostly the particles produced by the hard interaction.
- **Transverse regions:** the two transverse regions are the most sensitive to UE.

The toward and away regions are the ones with the major contribution from the primary hard scattering, in fact in a dijets event the leading jet is expected in the toward region while the next-to-leading jet in the away region.

The two transverse regions are called:

- **TransMAX**: This is the transverse region that contains the *maximum* number of charged particles, or scalar p_T sum of charged particles. This region includes both MPI and hard-process contamination. As an example, in events with 3-jets production this region can contain the extra jet.
- **TransMIN**: is the transverse region that contains the *minimum* number of charged particles, or scalar- p_T sum of charged particles. This region is the most sensitive to MPI effects.

The leading object definition depends on the type of event under observation and defer from one analysis to another. The charged-particle with the largest p_T [25], the dilepton system in Drell-Yan observation [33, 34] or $t\bar{t}$ events [35] can all be used as the leading object in the analyses for the UE event.

So we are interested in studying the transMAX and transMIN regions in particular the observable sensitive to UE in these regions. The main observables studied in these regions are the charged-particle density and the charged-particle scalar- p_T sum density in the $\eta - \phi$ space.

3.2.1 Hard scale dependence

In a hadron-hadron collision with two jets productions, it is observed that not only the toward and away regions’ activity increases with the hardness of the collision (p_T^{max}) but also the transverse regions’ activity increases with it. This is shown in Fig. 3.3, where the different color lines refer to different scales for the collision. It is observed that the away region ($60 \leq |\Delta\phi| \leq 120$) at increasing energy for the collision become broader this is related to the increasing quantity of FSR.

This increment in the transverse regions activity cannot be explained only by the increase of the FSR. To explain this rise one needs to attribute activity in the transverse regions to MPI: the number of extra scatters increases with energy. This rise is related to the density functions for the partons inside the hadrons that increase when probed at higher energies and so the partons become denser packed in hadrons. In this way, the probability of extra scatterings is larger to higher energies.

The evolution of these two quantities in the transMAX region as a function of the transverse momentum of the leading object (measurement of the energy scale for the collision) is shown in more detail in Fig. 3.4. The two distributions show a rapid rise at low leading-object p_T than a very low rise starts.

3.2.2 Sensitivity to MPI parameters

Now we want to look for the sensitivity of these observables to some PYTHIA8 parameters. Fig. 3.5 shows the effect of the MPI on both the distributions shown before separately for TransMAX and TransMIN regions. The figure shows the case with MPI (red line) and the case when the MPI are switched off (blue line). As expected the MPI are needed to explain the activity in the transverse regions. When

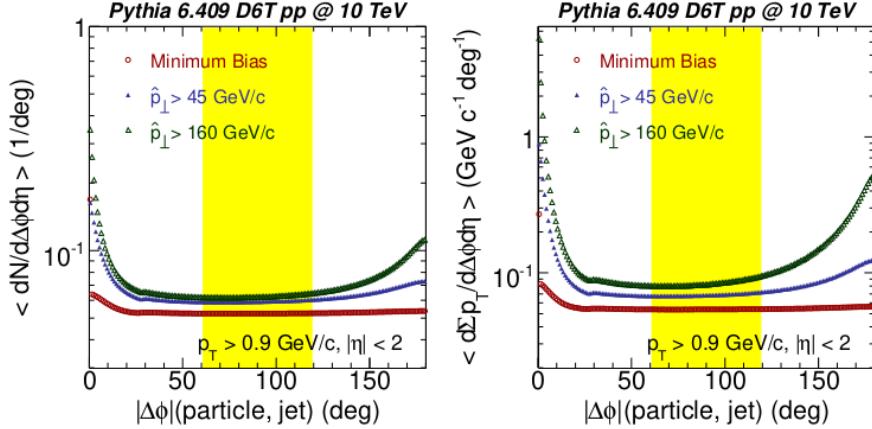


Figure 3.3: A comparison between three different scales for the interaction, in PYTHIA. The multiplicity of charged particles (left) and the scalar- p_T sum of charged particles (right) are simulated. The activity in the transverse regions increase due to two effects: the FSR is related to the broadening of the away region so some events from the shower end up in the transverse region (yellow band) but this alone can't explain the increment so is required the introduction of the MPI in the description. Figure from chapter 5 of [36].

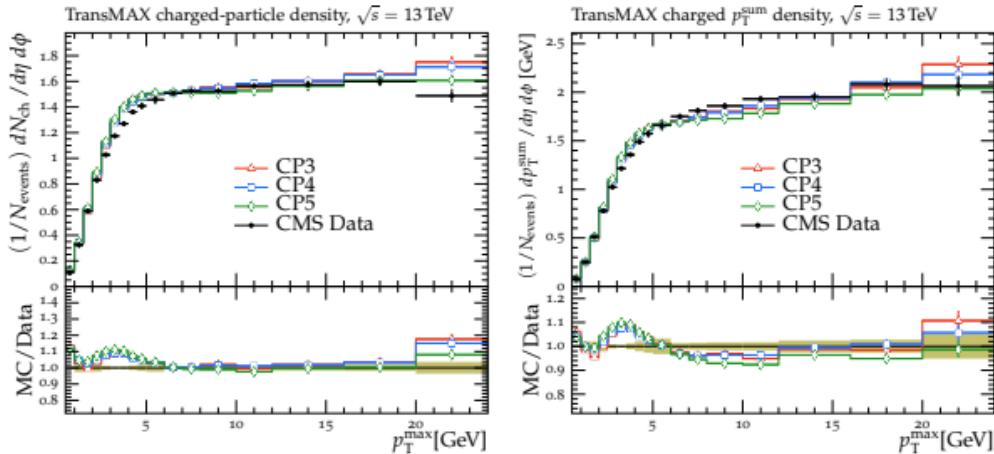


Figure 3.4: The evolution of the charged-particles density (left) and of the scalar- p_T sum density of charged particles (right) as a function of the energy scale for the scattering (leading object p_T), the two distribution increase quite rapidly in the first bins than they saturate ($\approx 6 - 7 \text{ GeV}$) the scalar- p_T sum density increase a little bit more, for $p_T >$ but very slowly. The black dots are the experimental point and are compared to prediction with CMS PYTHIA tunes: CP3, CP4 and CP5; they are described in more detail in the next chapter. Figure from [6].

the MPI are switched on the number of particles produced is higher than the case when they are off. From this figure is clear that MPI are the main contribution to the activity of these TransMAX and TransMIN regions. Obviously, the discrepancy increases as the hard scale of the interaction increases, in fact in low p_T regions the p_T evolution starts at a lower value so the phase-space, for the extra collisions to occur, is smaller.

Another important observation that was also pointed out in the Chapter 2.4.1

3.2. MINIMUM BIAS MEASUREMENTS AND UNDERLYING EVENT TOPOLOGY

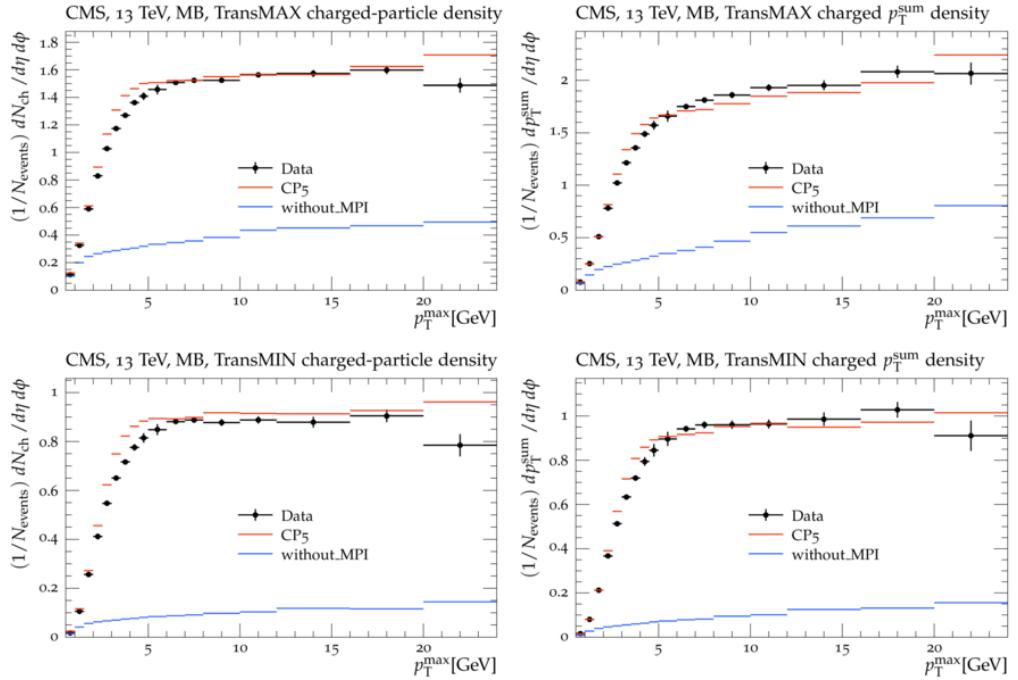


Figure 3.5: This image shows the effect of the MPI in the transMAX and TransMin regions. The contribution of the parton shower alone can't explain the contributions of the underlying event in these two regions (blue line) the introduction of the contribution from the MPI is necessary (red line). The two simulations are compared to the data from [25].

is that the amount of activity in the two transverse regions is also dependent on the center-of-mass energy. This evolution is shown in Fig. 3.6 for two different energy. The amount of MPI increase with \sqrt{s} that was expected from the evolution of the PDF with the energy of the collision: the hadrons become denser-packed when probed to higher energies. In Fig. 3.6 the charged particle density and the scalar sum of the transverse momentum in the transverse regions data are compared for two different center-of-mass energies. The data are also compared to existing different CMS tuning.

So these observables are the main ones we are going to use in our tune and used in previous tunes for the UE. The data from the activity in the TransMAX and TransMIN regions have been collected by a large number of experiment at different center-of-mass energies.

The selection criteria for CMS data from analyses [25] and [26] are the selection of charged particle tracks within the pseudorapidity range $|\eta| < 2$ and $p_T \geq 0.5$ GeV in order to ensure a good track reconstruction efficiency. Fake tracks after mis-reconstruction are removed imposing to pass the *highPurity* [28]. Secondary decays are removed by requiring the impact parameter significance $d_0/\sigma_{d_0} < 3$ and the significance in the z-direction $d_z/\sigma_{d_z} < 3$. Then to keep only tracks with good resolution the tracks with relative uncertainties $\sigma_{p_T}/p_T > 0.05$ are discarded.

To reconstruct the charged particle jets the clustering algorithm use is the Seedless Infrared-Safe Cone (SIScone) [29] jet algorithm². Charged particle within the pseu-

²The Anti- k_T algorithm [30] is currently the more preferred jet clustering algorithm in

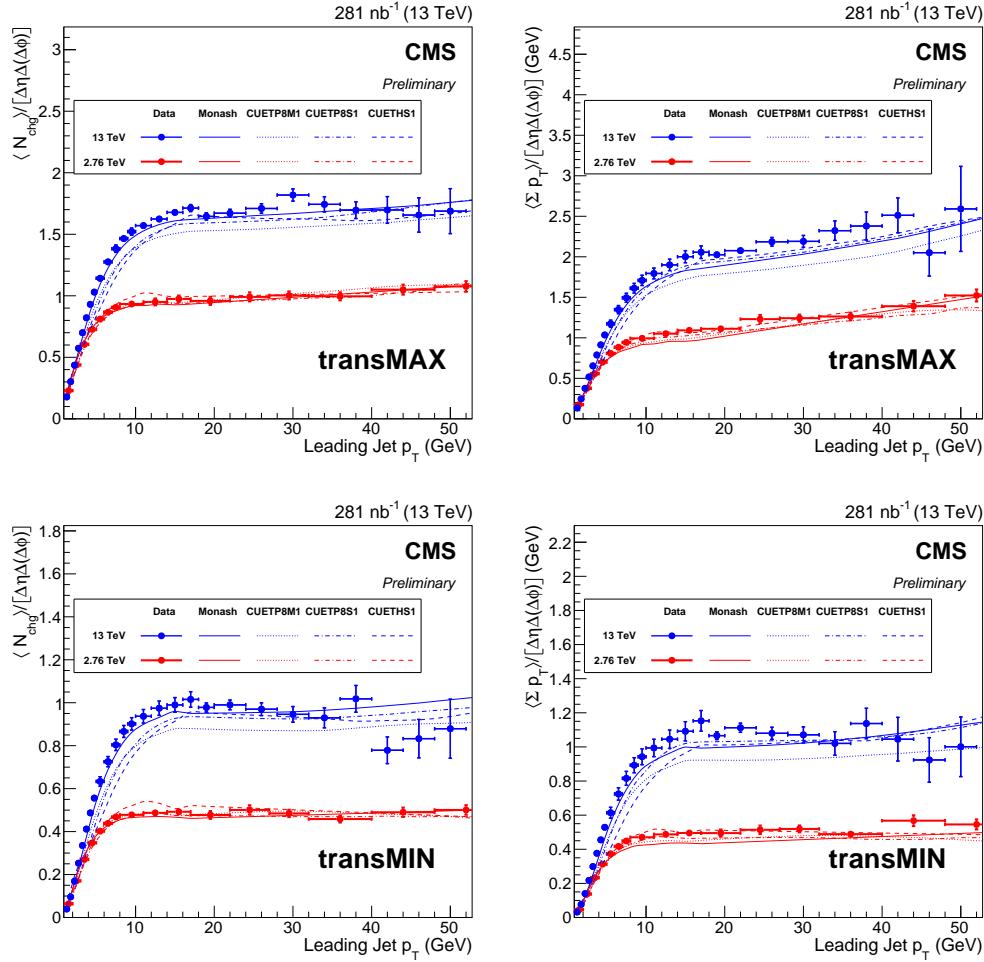


Figure 3.6: The charged particle density in the transMAX (upper left) and transMIN (lower left) regions and of the charged particle p_T -sum in the transMAX (upper right) and the transMIN (lower right) regions evolution as function of the center-of-mass energy is shown. The red ones are the data for $\sqrt{s} = 2.76$ and the blue ones for $\sqrt{s} = 13$ and these are compare to different CMS tunes. Figure from [25]

for rapidity range $|\eta| < 2.5$ are used for jet reconstruction then the reconstructed jet have to pass the above selection criteria.

Minimum bias events are triggered by requiring activity in both Beam Scintillator Counters (BSC) located at a distance of ± 10.86 m from the detector center and are sensitive within $3.23 < |\eta| < 4.65$ in coincidence with signals from both beams in the Beam Pick-up Timing for eXperiments (BPTX) devices these coincidence indicate that both the beams are crossing at the same time and that they interact.

3.2.3 Observables in Z production processes

In the last part of this work, it will performed the tune of primordial k_T another important parameter in the simulation of hadron-hadron collisions. An unresolved

CMS but to ensure continuity with respect to the previous analyses on the UE the one above is chosen.

question is that the primordial k_T value required for the description of the observed cross-section for the Z production is very large with respect to the expected value derived from the typical size of a hadron in Eq. 2.38. This very high value is not understood by first principles so it is required a tune for this parameter in the Monte Carlo generator in order to describe the data in a good way.

The observables to study the primordial k_T are the differential cross-section for Z production in proton-proton collision in Drell-Yan observation as a function of the transverse momentum of the Z boson (Fig. 3.8 left) and as a function of the ϕ_η^* angle (Fig. 3.8 right). Where the angle ϕ_η^* is defined as:

$$\phi_\eta^* = \tan\left(\frac{\pi - \Delta\phi}{2}\right) \sin(\theta_\eta^*) , \quad (3.1)$$

where $\Delta\phi$ is the azimuthal angle between the two leptons. The angle θ_η^* instead is the angle measured with respect to the beam direction in the rest frame of the dilepton system (frame with leptons emitted back-to-back), so is defined as:

$$\cos(\theta_\eta^*) = \tanh\left(\frac{\eta^- - \eta^+}{2}\right) , \quad (3.2)$$

with η^+ and η^- the pseudorapidities of the two leptons [37].

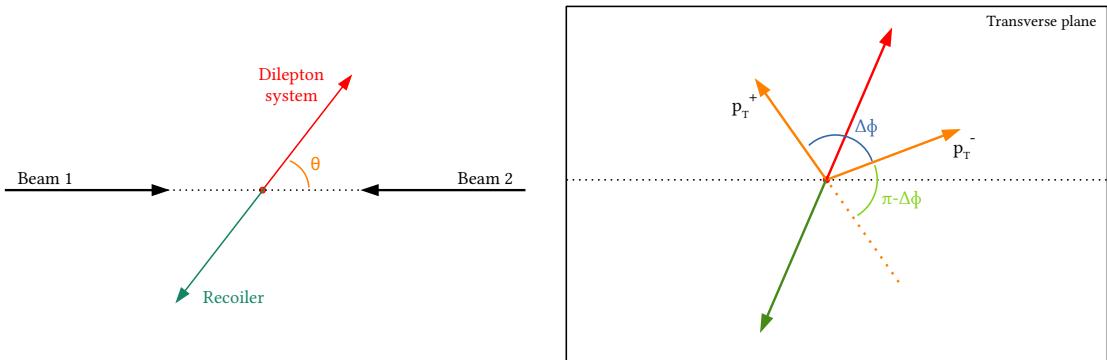


Figure 3.7: A schematic representation of a Drell-Yan observation with a description of the angles used in the definition of ϕ_η^* in Eq. 3.1

The Z boson is a very massive boson, $m = 91.1876 \pm 0.002$ GeV [2], which can be produced only in the hard scattering of two partons of the hadron undergoing the collision, then after a very short lifetime the Z boson decay in a couple of leptons: $e^+ e^-$ or $\mu^+ \mu^-$. The measurement of this final state takes information on the Z boson that is not directly observed by experiments. It is a matter of fact that the production of the Z boson can be affected by all the other soft processes taking place along with this production that can change the topology of the initial state. In the last chapter it is investigated the effect of ISR and Primordial k_T on the Z boson production in the low region of the spectrum.

Since the Z boson production is a hard process to full simulate the Z production spectrum we have to merge matrix element calculation and space shower to do this we have to use a margin scheme. To have a good description of the high region of the spectrum ($p_T^Z > 2$ GeV) we have to include higher-order calculation to consider NLO and NNLO processes and so we have to adopt a more complex merging scheme

as the FxFx merging scheme that has been introduced in Chapter 2.

As shown in Fig. 3.8 the description at LO (and without the need for FxFx) is good only in the first bins that are the ones of interest for the study of underlying event effects and it is less computationally expensive than running all the FxFx merging scheme. So, the strategy followed for the tuning of the primordial k_T in the Chapter 6 is to tune the Primordial k_T in this low region, the one of interest, taking only the first 5 bins from each distribution.

Then the full simulation with FxFx is run only for the comparison once the parameters have been tuned. This is going to be discussed in more detail in Chapter 6 where the primordial k_T tune is performed.

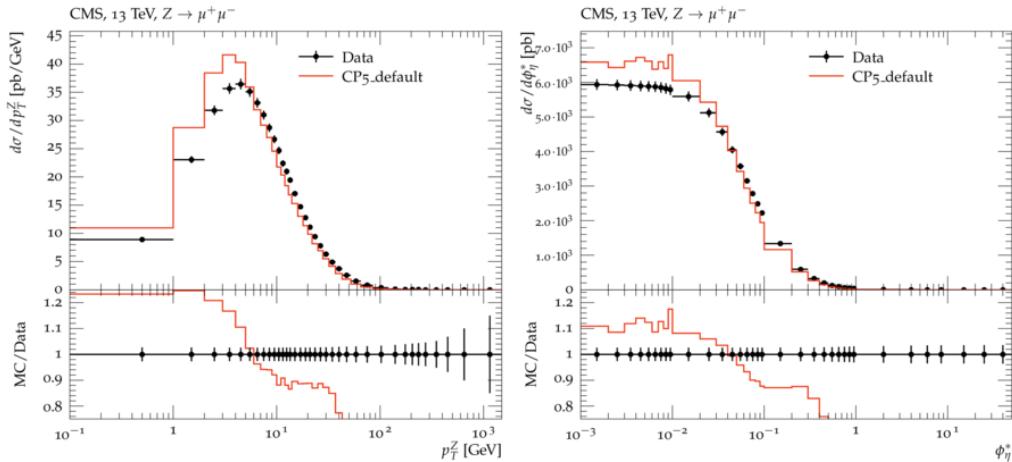


Figure 3.8: Here are shown the distribution, from the [38] CMS analysis at $\sqrt{s} = 13$ TeV, which we are going to use for the tune of the primordial k_T . The left one is the Z boson production cross-section as a function of the transverse momentum of the Z boson. On the right the differential cross-section for the Z boson production as a function of the ϕ_η^* angle. The red line is the PYTHIA8 description before the tune of the primordial k_T and without the FxFx merging scheme so only the first bins have to be taken into account.

In the following section, we will introduce the existing CMS tunes for the UE in minimum bias observations. These tunes are able to reproduce very well the observables described in Section 3.2 at 1.96, 7 and 13 TeV.

3.3 Previous Tune for the Underlying Event

In the paper [6] the CMS collaboration presents new PYTHIA8 tunes for the underlying event (UE). The tunes are called CP and a number from 1 to 5 where CP stands for "CMS PYTHIA8". The tunes are performed by changing the values of the coupling constant α_s for the ISR, FSR, hard scattering and MPI and changing the order of the evolution for the α_s with the Q^2 value of the interaction. Another difference among these tunes is the choice of the PDF set: CP1 and CP2 use a LO PDF, CP3 an NLO one, while CP4 and CP5 tunes use an NNLO PDF set.

The tuned parameters are shown in Table 3.1 with the associated variation ranges and a recall of the definition of each one.

3.3. PREVIOUS TUNE FOR THE UNDERLYING EVENT

Parameter description	Name in PYTHIA8	Range considered
MPI threshold [GeV], pT0Ref, at $\sqrt{s} = \sqrt{s_0}$	MultipartonInteractions:pT0Ref	1.0 – 3.0
Exponent of \sqrt{s} dependence, ϵ	MultipartonInteractions:ecmPow	0.0 – 0.3
Matter fraction contained in the core	MultipartonInteractions:coreFraction	0.1 – 0.95
Radius of the core	MultipartonInteractions:coreRadius	0.1 – 0.8
Range of color reconnection probability	ColorReconnection:range	1.0 – 9.0

Table 3.1: This table reports the five parameters tuned for the UE and Minimum Bias in CP* tunes, the variation ranges used for the sampling are shown in the last column. Table from [6]

These parameters are the ones that govern the number of MPI and the amount of Color Reconnections³. These CP* tunes want to be general (multi-purpose) tunes for UE and minimum bias observables described above.

These tunes are performed with the standard tool for the high energy physics tune: PROFESSOR.

3.3.1 The distributions used for the Tune

The observables distributions used for the tune are the following ones:

- The pseudorapidity distribution of charged hadrons (p_T, K, π) measured for an inclusive selection in inelastic proton-proton collisions [32];
- Charged particle density and charged particle scalar p_T^{sum} in TransMIN and TransMAX regions at different \sqrt{s} (1.96 TeV [27], 7 TeV [26], 13 TeV [25]);
- The pseudorapidity distributions for single diffractive (SD) and non single diffractive (NSD) events selection [31].

In the next chapter, we will describe our tune using MCNNTUNES and in it we used the same distribution listed here. All the graphs for these observables are then shown with the final tunes descriptions.

3.3.2 Pyhtia configuration and the tunes

On the top section of Table 3.2 and Table 3.3 are reported the values for the PYTHIA8 parameters used in the CP tunes and, on the bottom, the five parameters resulting from the tune.

Below we are going to focus on the CP5 tune. We are going to reproduce this tune using the same settings for PYTHIA8 but using a different tuning software: MCNNTUNES.

The next chapter discusses the general approaches to the tune and why we use machine learning and on the description of the main tool used for the tuning in this work MCNNTUNES.

³Note that all the other parameters are set to the default values of the Monash tune [39]

PYTHIA8 parameter	CP1	CP2
	NNPDF3.1 LO	NNPDF3.1 LO
PDF Set		
$\alpha_s(m_Z)$	0.130	0.130
SpaceShower:rapidityOrder	off	off
MultipartonInteractions:EcmRef [GeV]	7000	7000
$\alpha_s^{\text{ISR}}(m_Z)$ value/order	0.1365/LO	0.130/LO
$\alpha_s^{\text{FSR}}(m_Z)$ value/order	0.1365/LO	0.130/LO
$\alpha_s^{\text{MPI}}(m_Z)$ value/order	0.130/LO	0.130/LO
$\alpha_s^{\text{ME}}(m_Z)$ value/order	0.130/LO	0.130/LO
MultipartonInteractions:pT0Ref [GeV]	2.4	2.3
MultipartonInteractions:ecmPow	0.15	0.14
MultipartonInteractions:coreRadius	0.54	0.38
MultipartonInteractions:coreFraction	0.68	0.33
ColorReconnection:range	2.63	2.32

Table 3.2: CP1 and CP2 tunes settings are reported here together with the values for the parameters tuned. CP1 and CP2 use a LO PDF set. CP1 α_s is different between matrix element calculation and MPI that use a value of 0.1365 ISR and FSR that instead uses 0.130. While CP2 use the same value for all processes, it is fixed at 0.130. In both cases α_s run with a LO evolution. Table from [6]

PYTHIA8 parameter	CP3	CP4	CP5
	NNPDF3.1 NLO	NNPDF3.1 NNLO	NNPDF3.1 NNLO
PDF Set			
$\alpha_s(m_Z)$	0.118	0.118	0.118
SpaceShower:rapidityOrder	off	off	on
MultipartonInteractions:EcmRef [GeV]	7000	7000	7000
$\alpha_s^{\text{ISR}}(m_Z)$ value/order	0.118/NLO	0.118/NLO	0.118/NLO
$\alpha_s^{\text{FSR}}(m_Z)$ value/order	0.118/NLO	0.118/NLO	0.118/NLO
$\alpha_s^{\text{MPI}}(m_Z)$ value/order	0.118/NLO	0.118/NLO	0.118/NLO
$\alpha_s^{\text{ME}}(m_Z)$ value/order	0.118/NLO	0.118/NLO	0.118/NLO
MultipartonInteractions:pT0Ref [GeV]	1.52	1.48	1.41
MultipartonInteractions:ecmPow	0.02	0.02	0.03
MultipartonInteractions:coreRadius	0.54	0.60	0.76
MultipartonInteractions:coreFraction	0.39	0.30	0.63
ColorReconnection:range	4.73	5.61	5.18

Table 3.3: Here are reported CP3, CP4 and CP5 tunes settings, and the results for the tune. The three tunes use an equal α_s value for all the processes, $\alpha_s = 0.118$ running with a NLO evolution. The difference between CP3 and the other two tune is that CP3 use an NLO PDF set while CP4 and CP5 an NNLO one. CP5 ISR emission is also ordered according to rapidity. Table from [6]

Chapter 4

Tuning procedure and MCNNTUNES

As was introduced in the first chapter the tuning procedure can be really computational expensive, in fact it requires to run the generator a very large amount of times, and usually these MC generators are really expansive in terms of computational time required for just a single job.

To tune a certain number of parameters the number of Monte Carlo runs needed for the tuning procedure increases with the number of free parameters. In fact, the dimension of the parameter space increases with the number of the sampling we want to perform and the granularity we want to achieve. So, in the past, different approaches have been developed and used to tune these MC generators. A brief description is reported for each approach in the following list:

- 1) **Manual tunes:** this approach is based on an optimization of the parameters made by eye. This is absolutely not the best way to tune some parameters, usually it requires a very large time for even semi-reasonable results since the process requires a very large number of iterations.
- 2) **Brute force tunes:** a better way would be to perform a very dense sampling in parameters space, run the generator with every configuration and then chose the configuration corresponding to the output that better describes the experimental data. This is very computationally expensive and a scan in a 5 parameters space with 10 divisions each requires $10^5 = 100000$ Monte Carlo runs this with a rising number of parameters becomes really impractical, also using computers batch systems as an example HTCondor.
- 3) **Parametrization-based tunes:** an even more better approach is to find a surrogate function to parameterize the response of the MC generator at different values of the parameters to tune, and try to study (minimize) this surrogate function instead of the real response of the generator. This is the current approach that is used in the high energy physics tune and that will be used in the following tunes presented in this thesis.

This last approach will be discussed in more detail in the next section.

4.1 Parametrization-based approach

The parametrization-based approach is the most used method. The current state-of-art in the tuning procedure is to use a polynomial function to fit the response of the generator. Once the parametrization is performed the tuned parameters are given by the minimization of this parameterized response function. This approach based on the polynomial parametrization has been implemented in the software PROFESSOR [40].

So the first step in the procedure is to fit the response of the generator using a surrogate function simpler to study than the real one (e.g. Professor instead of the arbitrary complex real function uses a polynomial fitted to well describe the real function).

$$h(p) \xrightarrow{\text{parametrization}} \bar{h}(p) , \quad (4.1)$$

where $h(p)$ is the vector of the observables distribution bins from MC simulations that are a function of the parameters. In this phase, these functions that give the values of the bins at the variation of the parameter values have been substituted by the surrogate ones $\bar{h}(p)$.

After that, a *loss function* $\mathcal{L}(\bar{h}(p), h_{\text{data}})$ is defined between the surrogate function and the experimental data. A common choice for it is the χ^2 function defined as:

$$\mathcal{L}(\bar{h}(p), h_{\text{data}}) \equiv \chi^2 = \frac{(\bar{h}(p) - h_{\text{data}})^2}{\sigma^2} . \quad (4.2)$$

where σ^2 is the vector of the experimental squared error on each bin. In the end, to find the best parameters estimation, this loss function needs to be minimized. The set of parameters p_{best} is the best evaluation that our generator can provide for the real values and we are going to call this set of best parameters: *tune*.

$$p_{\text{best}} = \arg \min_p \mathcal{L}(\bar{h}(p), h_{\text{data}}) . \quad (4.3)$$

In this thesis instead of the common software PROFESSOR based on the polynomial parameterization, it is used the machine learning approach implemented in MCNNTUNES software [41] using Feed Forward Neural Networks. MCNNTUNES is a software developed by S. Carrazza, S. Alioli and M. Lazzarin presented in [7] based on machine learning library TensorFlow [42]. MCNNTUNES is written in Python and uses Feed Forward Neural Networks that are trained in order to learn the generator behaviour with respect to the parameter variations. This removes the polynomial constraint in the fit of the generator response function in fact one of the main features of the Neural Networks (NNs) is that they are universal function approximators. A brief introduction follows on machine learning and in particular on NNs in order to understand better how MCNNTUNES works.

4.2 Machine Learning and Neural Networks

Machine learning (ML) is a particular type of Artificial Intelligence and consists of the implementation of systems that learn automatically by the data that are fed into them and not by the explicit programming of the algorithm. It is clear that ML

requires the training of the algorithm in order to have a predictive output related to the problem under analysis. The training is the most important step in the ML approach in fact is this step that gives to the ML the ability to return a predictive output; without it the output we get is not able to give us a meaningful result.

A particular type of ML is Deep Learning which uses neural networks with more than one layer organized in a hierarchical structure to solve the problem. This is the type of ML we are interested in. But before we start with the explanation of the simple possible neural network that can be created in order to understand the basis in the logic of each unit that is going to compose the neural network.

4.2.1 Neural Networks - Perceptron

The concept of the Neural Network (NN) was developed in 1958 by Frank Rosenblatt. He introduces the simpler example of NN: the perceptron [43]. A representation of a perceptron is shown in Fig. 4.1 the input values are weighted and summed, an additional offset b can be introduced, and then the weighted-sum is passed to an activation function (step function). So, the output that the perceptron returns is:

$$h(x) = \text{step}\left(\sum_j w^j x_j + b\right) \quad (4.4)$$

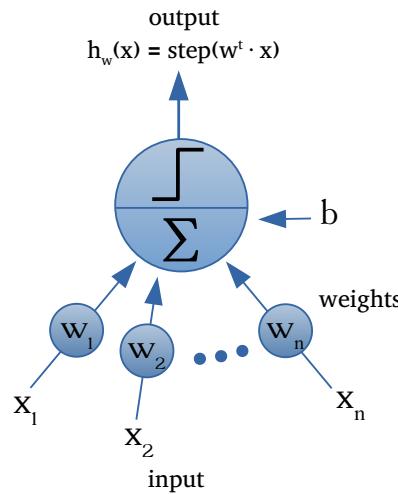


Figure 4.1: A schematic representation of a perceptron.

The revolutionary feature of the perceptron was the ability of learning by an adjustment of the weights. Anyway, a single perceptron is not enough since these kinds of logical units have lots of limitations. An example of a limitation for the perceptron is shown in Fig. 4.2 where the impossibility of implementing a XOR operation using a perceptron is shown with a graphical explanation. The perceptron is a linear classification algorithm and in the image is represented as a blue line that set a boundary for the acceptance of the hypothesis.

To describe more complex problems it has to be introduced the concept of NN with more than one unit if we want to eliminate these limitations and approximate every type of function.

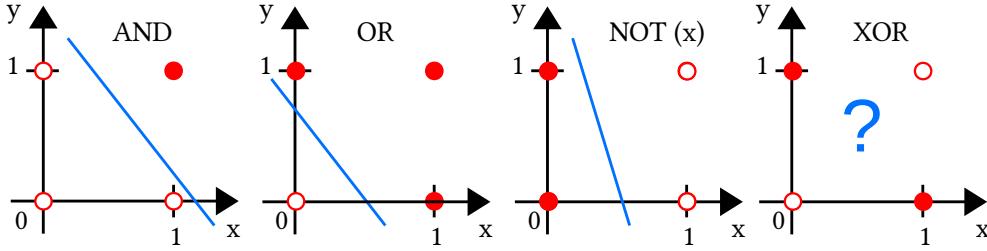


Figure 4.2: The figure shows one of the limitations of the perceptron. The XOR operation is not possible with a linear cut. In the figure the two axes are related to the two boolean variables x and y ; the possible values for these variables are only 0 and 1, the four combinations are indicated by the red circles: $(0, 0)$, $(1, 0)$, $(0, 1)$ and $(1, 1)$. The first image shows the AND operation the full red circle indicates the accepted point $(1, 1)$, the discrimination can be performed correctly with just one straight line (perceptron discrimination). Also, the OR $((0, 1), (1, 0), (1, 1))$ and the NOT x $((0, 0), (0, 1))$ ones can be implemented correctly using perceptron. But this cannot be done for the XOR operation $((0, 1), (1, 0))$.

4.2.2 Feed-Forward Neural Networks

In a Neural Network different units called "neurons" are linked together. Different types of NNs exist and they are classified according to the various ways the neurons are linked to each other. We are interested in *fully-connected Feed-Forward NNs*, which are the ones used in MCNNTUNES.

In fully-connected NNs each neuron from a layer is connected to every neuron in the next layer, while the Feed Forward attribute refers to the fact that the NN have no internal recursions (loops) between neurons but all the neurons from a layer are connected forward to the ones of the next layer.

Fig. 4.3 shows a schematic view of a fully-connected feed-forward multi-layer NN. The basic idea is that the neurons can get some value in input and return a value as output. The output of a neuron is then sent to all the neurons in the next layer and weighted differently for each one. To be more specific: each unit j in the hidden layer (i) takes a vector of values x^k , that come from all the k neurons of the previous layer, and an offset ϑ_j as input, computes the weighted sum $\sum_k w_{kj}x^k$ and apply the activation function ϕ to the result. Common choices for this function are \tanh or *sigmoid* functions. So, the total output of the (i) -th layer in the network is the function:

$$f^{(i)}(x) = \sum_{j=1}^{N^{(i)}} \varphi \left(\sum_k w_{kj}^{(i)} x^k + \vartheta_j^{(i)} \right) , \quad (4.5)$$

where $N^{(i)}$ is the number of hidden units in the (i) -th layer.

One of the prominent features of the NNs is that they are universal function approximators [44, 45] the only request for this is that a sufficient number of hidden layers is available. This is the main reason why we want to use a NN based approach instead of the polynomial one implemented in PROFESSOR.

As mentioned before the main feature of all the types of NNs is the ability to learn from data without being directly programmed. But to this and so get some predictive results from the NN a learning algorithm has to be defined.

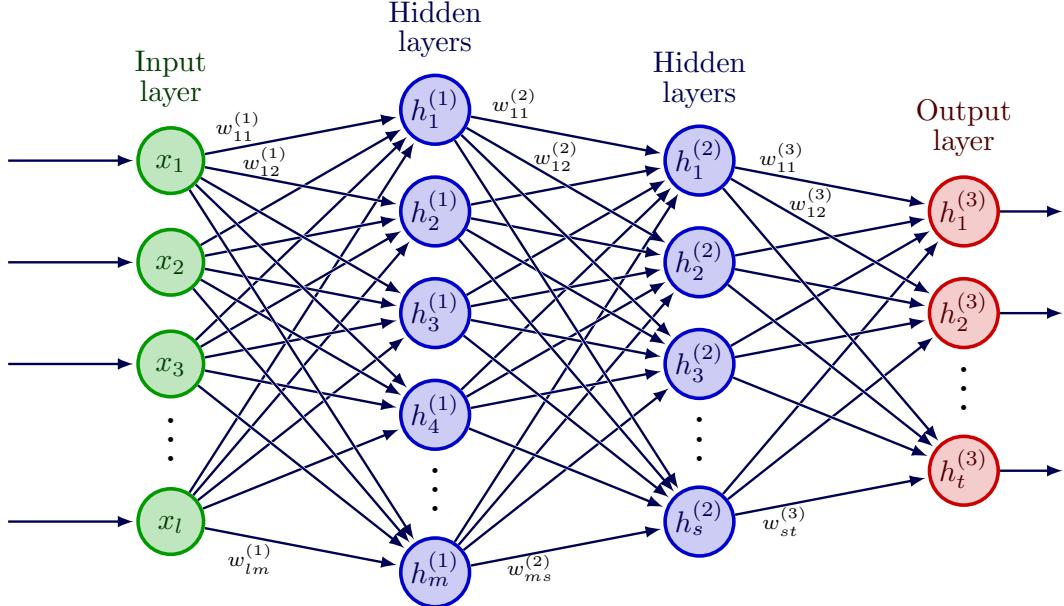


Figure 4.3: A fully-connected feed-forward neural network with more than one hidden layer, the green one is the input layer and takes the value in input and usually scale them in the range $[0, 1]$ and then the data sequentially go through the neurons in the hidden layers (blue) where all the steps described above are performed for each layer. At the end the results of the computation are collected by the output layer (red).

A common training algorithm for the NN is the *back-propagation*, where a set of Monte Carlo simulations is used to train the NN. The back-propagation procedure is based on the idea of changing the weights, w_{jk} , and the offsets, ϑ_j , in order to minimize a loss function usually defined as the mean squared error (E):

$$E = \frac{1}{2} \sum_i (h_i(x^j, w_{jk}) - d_i)^2 \quad , \quad (4.6)$$

where the h_i are the value in output from the NN and the d_i the real value known from the Monte Carlo truth.

In the back-propagation algorithm the weight and the coefficients are updated using the *steepest-descent minimization*:

$$w_{jk}^{(i+1)} = w_{jk}^{(i)} - \lambda \left(\frac{\partial E}{\partial w_{jk}} \right)^{(i)} ; \quad \vartheta_j^{(i+1)} = \vartheta_j^{(i)} - \lambda \left(\frac{\partial E}{\partial \vartheta_j} \right)^{(i)} \quad (4.7)$$

where λ is the *learning rate* and is a user-tunable free parameter and it controls how much change the weights each time the weights are updated. It is one of the main parameters in the neural networks a too small value for it can lead to a failure in the training procedure while a too large one can lead to unstable results.

The *mini-batches* are subsets of our training set that contains the Monte Carlo simulations that are used to calculate the gradient. The size of the mini-batches used to train the NN is also a free parameter: smaller batches are faster to compute but the gradient direction is not the real one just an approximation; while a bigger

batch size gives a good approximation of the direction of the steepest-descent but can be computationally expensive.

The number of evaluations of the entire training set is called *epoch* and it is tunable by the user.

Note that the batch size and the number of epochs are strictly related. A training set of 50 MC runs with a batch size of 10 and a number of epochs of 1000 requires 5000 iterations while if we use a batch size of 25 it requires only 2000 iterations to run over the whole training set.

A schematic representation of the training process is displayed in Fig. 4.4 where the training set is subdivided into the mini-batches that one by one are fed to the NN. Then, the Monte Carlo truth and the output of the network are used to compute the loss function E , using this information the back-propagation, based on the gradient descend algorithm, updates the weights in the descent direction of the gradient calculated using the mini-batch. All the procedure is repeated with every mini-batch and for a user tunable number of epochs.

Note that the prediction capability of the Neural Network is dependent on the selection of all these user-tunable parameters that control the Neural Network architecture. All these parameters are usually referred to as *hyperparameters*.

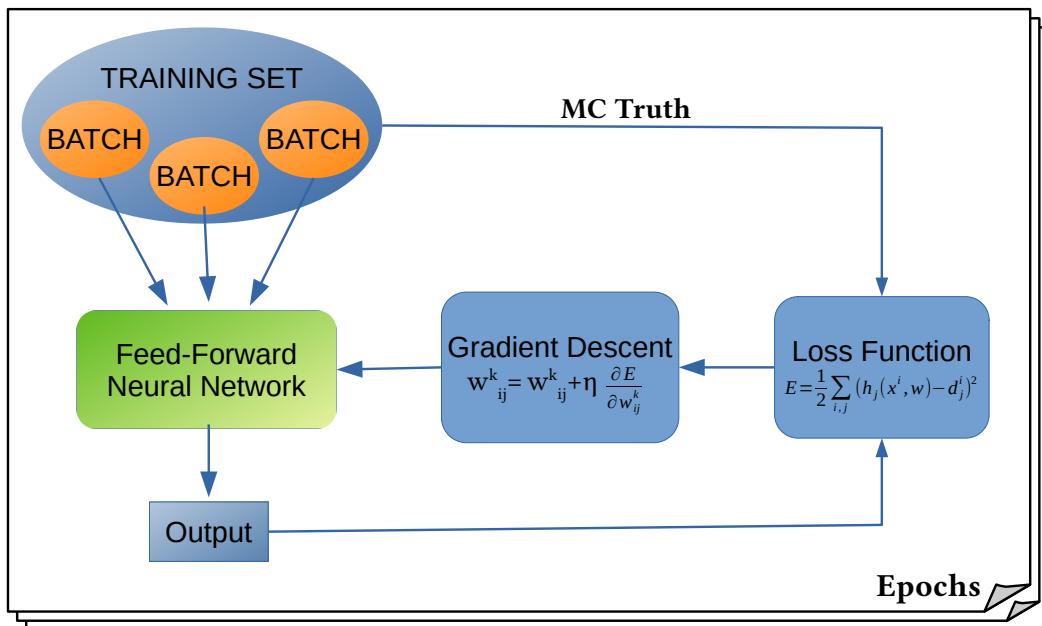


Figure 4.4: The training set used to train the NN is divided into mini-batch (the mini-batch size is a user tunable parameter) then one by one the batches are fed to the NN the output of the network together with the Monte Carlo truth are used to calculate the loss function. Then the weights and the offset are updated as described in Eq. 4.7 with the back-propagation algorithm. This is done with each mini-batch in the training set and for a number of epochs defined by the user.

In the next section, `mcnntunes` is introduced and all its working modes are explained in detail.

4.3 MCNNTUNES

MCNNTUNES [7] is a Shower Monte Carlo generators tuning tool that implements a tune procedure based on the use of Feed Forward Neural Networks. The advantage of using FFNNs has been described above and it is that they are universal function approximators at the simple cost of having a sufficient number of hidden units in the hidden layers. This feature allows us to remove the polynomial bias present in PROFESSOR tool.

MCNNTUNES offers two different main operation modes: *PerBin Model* and *Inverse Model*. The first one is based on an approach similar to the one in PROFESSOR but where the response of the generator is parameterized using these FFNNs, one for each bin; the latter is a totally new approach where the NN (only one in this case) is trained to learn the inverse function of the generator response and then used to tries to infer the parameters value starting from the experimental values of the bins in the distributions used.

4.3.1 Sampling Phase and Training Set Generation

The two approaches have the same starting point that is a sampling of the parameter space (e.g. for the UE analysis we use the parameters space shown in Table 3.1), then the generator is run with every sampled configuration. All these MC runs are going to build our dataset which is called *training set*.

Fig. 4.5 shows a schematic explanation of how the sampling phase and the training set generation were performed using the CMS software environment referred as CMSSW. CMSSW is present on the public machine cluster of the CERN called LXPLUS (LinuX Public Login User Service). In more detail the sampling starts by means of the MCNNTUNES script called MCNNTEMPLATE with a sampling of the defined parameter space. The sampling generates N different configurations for the parameters to tune, these are then encapsulated in a runcard for PYTHIA that contains all the necessary information to make PYTHIA work properly (beams energies, type of event to simulate etc.). Once these runcard are saved the PYTHIA generator is run with every configuration generated and this is performed not locally but on a computers batch (e.g. the CERN one CondorHT) in order to gets all the results in few days. The runcards we pass to Condor contain also the information on the analysis we want to perform and on how to fill the various histograms for the observables. Once the generators runs end the outputs are saved in the YODA format [46], that is a standard type of data file for these analyses. The set of these YODA files, containing the information on the MC runs, composes our training set that then can be used for the tuning procedure.

MCNNTUNES offers also the possibility of changing the value of the hyperparameters. It is possible to choose the NN architecture: the number of hidden layers, the number of neurons for each layer and the activation function used. It is also possible to set the number of epochs, the batch size and the learning rate in order to have the best train for the architecture selected and other choices are possible. This is really important in order to get the best possible results for the tune. Below

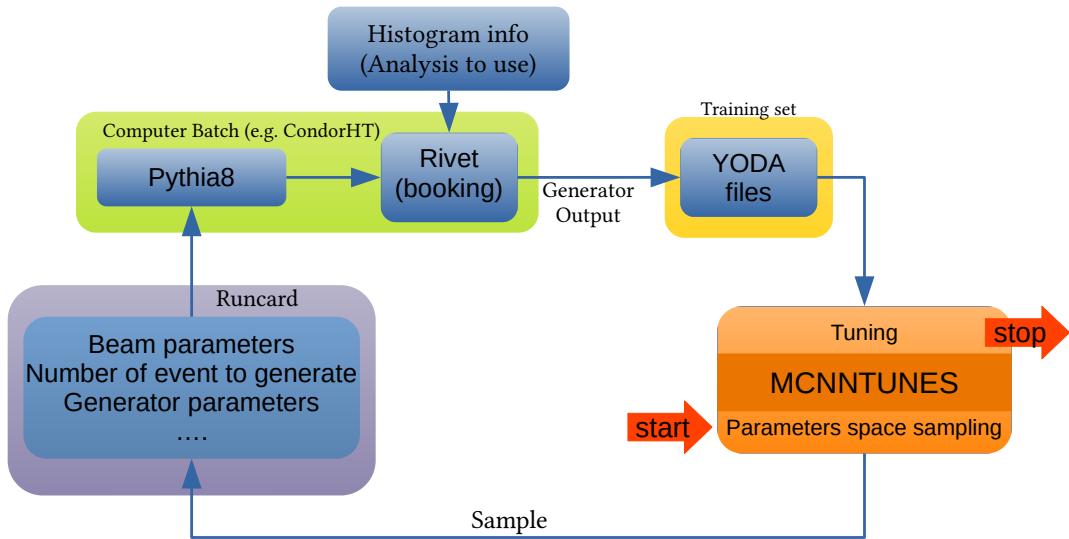


Figure 4.5: A schematic description of the main step from the sampling phase to the generation of the training set used for the tune procedure. These shows how the work was performed in CMSSW environment. The starting point and the final tune procedure are both controlled by MCNNTUNES. In the middle of these two phases the generation of the training set is required, this is performed running the generator many times.

we are going to discuss the procedure used for the Inverse Model in order to search for the best hyperparameters configuration.

4.3.2 PerBin Model

PerBin Model is a parametrisation-based method. The main idea, as shown in Fig. 4.6 is to build a model (i.e. a neural network) for each bin in order to parameterize the generator output. Each NN takes the parameter values as input and returns the bin value as output.

All these NNs are then trained feeding the MC runs from the training set and using a gradient-based algorithm, as usual for feed forward neural networks, with mean squared errors as loss function.

Once, the NN is trained, the last step is the tune in which one actually gets the best parameters estimation. This step defines a surrogate loss function for the tuning problem. In fact, the parameterization step returns a model $h^{(i)}(\mathbf{p})$ for each bin, i , where \mathbf{p} is the vector of the parameters.

Then, this surrogate loss function is defined as:

$$\chi^2 = \sum_{i=1}^N \frac{(h^{(i)}(\mathbf{p}) - h_{exp}^{(i)})^2}{\sigma_{(i)}^2} \quad (4.8)$$

need to be minimized in order to evaluate the best estimation for the parameters. In `mcnntunes` this minimization is performed using the CMA-ES algorithm [47]. So the best estimation for the parameters is the configuration of parameters that minimizes this χ^2 .

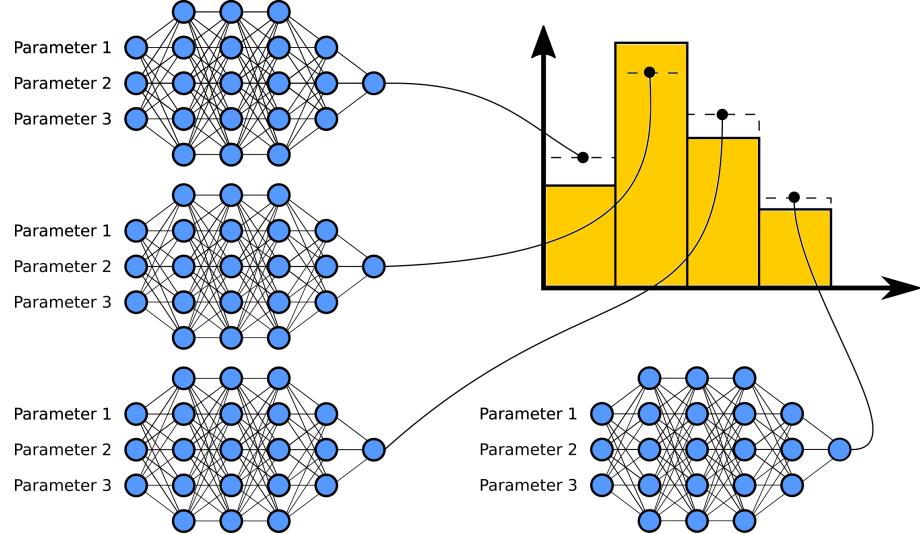


Figure 4.6: Figure from [7]

Error evaluation

MCNNNTUNES PerBin model as introduced in [7] did not have a proper error evaluation on the minimized parameters. In fact it was using as error the final width of the distribution of sampled point in the CMA-ES algorithm. But, this was not working the errors were underestimated on all the minimized parameters. Thanks to S. Carrazza and with the help of M. Lazzarin we had the opportunity of handle the code and add a proper error estimation.

Now, the error evaluation for the PerBin model is given by the definition of a confidence interval using the χ^2 function. In fact, as shown in section 9.6 and 9.7 of [48], for an estimators vector $\hat{h}(\mathbf{p}) = (\hat{h}^{(1)}(\mathbf{p}), \hat{h}^{(2)}(\mathbf{p}), \dots, \hat{h}^{(n)}(\mathbf{p}))$ for the parameters \mathbf{p} the probability distribution function and the likelihood (the χ^2 in our case) in limit of a large set of estimators and so of degrees of freedom (ν)¹ is approximately Gaussian distributed with mean ν and variance 2ν . The probability distribution function for the estimators is then:

$$f(\hat{h}(\mathbf{p})|h(\mathbf{p})) = \frac{1}{(2\pi)^{n/2}|V|^{1/2}} \exp \left[-\frac{1}{2} (\hat{h}(\mathbf{p}) - h(\mathbf{p}))^T V^{-1} (\hat{h}(\mathbf{p}) - h(\mathbf{p})) \right], \quad (4.9)$$

where T is the transposed vector and V^{-1} is the inverse covariance matrix. So, we can define our confidence interval using the χ^2 statistic as:

$$\frac{\chi^2(\text{c.i.})}{N_{dof}} = \frac{\chi^2_{min}}{N_{dof}} + \frac{Q_\gamma}{N_{dof}} \quad (4.10)$$

The variation is dependent on the number of parameters and on the chosen confidence level ($1\sigma = 0.683$ in our case) and a list of the values is reported in Table 4.1. Then, the error is defined as the value of the parameters that give a deviation from the minimum value of the χ^2/N_{dof} equal to the Q_γ/N_{dof} value for a confidence level of 0.683, as defined in Eq. 4.10.

An example for the evaluation in `mcnntunes` is shown in Fig. 4.7 where the green

¹In other cases this is a good approximation.

percentile	Q_γ				
	$\nu = 1$	$\nu = 2$	$\nu = 3$	$\nu = 4$	$\nu = 5$
0.683	1.00	2.30	3.53	4.72	5.89
0.90	2.71	4.61	6.25	7.82	9.24
0.95	3.84	5.99	7.82	9.49	11.1
0.99	6.63	9.21	11.3	13.3	15.1

Table 4.1: The table report the values of the quantile Q_γ for different percentile and for different number of degree of freedom ν . The row corresponding to 0.683 (1σ) is the one of our interest.

line is the deviation from the minimum value of the χ^2 defined in Eq. 4.10 and the errors are given by the points where the χ^2/Dof reaches these values, in the figure given by the intersection of the blue and the green line.

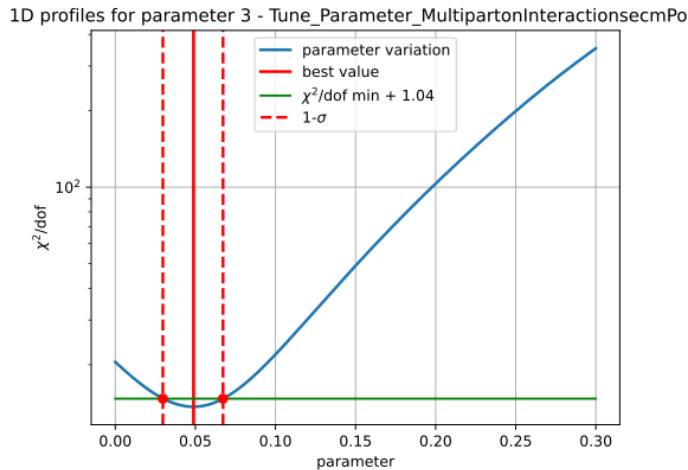


Figure 4.7: This figure shows the error evaluation in `mcnntunes` for the `MultipartonInteractions:ecmPow Pythia8` parameter. The blue line is the value of the χ^2/N_{dof} evaluated for different parameter values. The best estimation is indicated by the vertical red line, while the green line is the quantity in Eq. 4.10. The error is evaluated from the intersection of blue and green lines.

4.3.3 Negative aspects

One of the negative aspects is that this model is more computationally expensive than the Inverse model discussed below. This is due to the large number of NNs built and trained from this model. The high cost in terms of time required to get the model work do not give the possibility for a scan in the hyperparameters space in order to obtain the best configuration for the NN architecture. This can impose some limitations on the model performance that cannot get its maximum performance.

4.3.4 Inverse Model

The Inverse Model is the most innovative tuning procedure introduced by `mcnntunes`. This model contrarily to the PerBin Model takes the bins of the histograms as input and returns parameter values as output. For the Inverse Model the NN used is only one as shown schematically in Fig. 4.8. What the Inverse Model tries to do is to learn the inverted model of the generator. So starting from the observed values the model tries to reproduce the parameters values necessary to get the histograms we use as input.

The model is built and then trained with the training set introduced before. Once the model is trained feeding the experimental data to the NN this can try to infer the values of the parameters required to get the output.

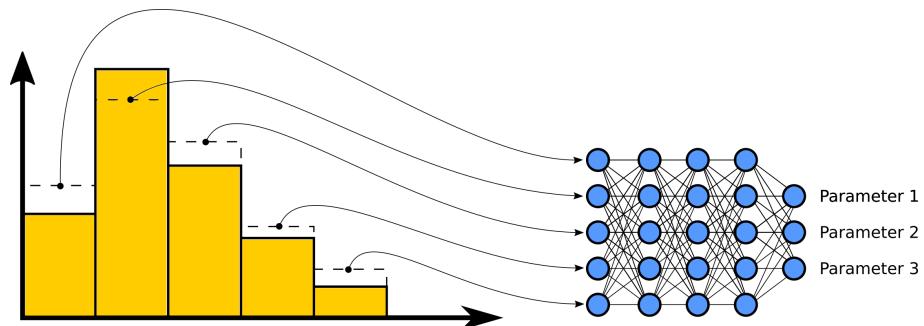


Figure 4.8: Figure from [7]

Errors evaluation

The errors are evaluated in a different way with respect to PerBin model. In fact, in the Inverse Model there is no a minimization step, and the error is evaluated by a re-sampling of the experimental data using a *multivariate Gaussian Distribution*, as the one in Eq. 4.11 with a diagonal covariance matrix that has experimental uncertainties on the main diagonal.

$$f(x_i; h_{\text{exp}}^{(i)}, \sigma_{\text{exp}}^{(i)}) = \mathcal{N} \cdot \exp \left[-\frac{1}{2} \sum_{j=1}^{N_{\text{bins}}} \frac{(x_i - h_{\text{exp}}^{(i)})^2}{\sigma_{\text{exp}}^{(i)}^2} \right] \quad (4.11)$$

So, a set of histograms is generated, then this is fed to the NN and a distribution of predictions is generated. An example is shown in Fig. 4.9, from this distribution one can compute the error by evaluating the standard deviation.

Note that this is a new method for the tune. As we are going to see this method requires more attention than the PerBin Model to get it working correctly in the case of a high number of parameters to tune.

With respect to the PerBin Model this method is faster in the training step. The NN trained is only one and is not needed a minimization so a scan in the hyperparameters can be performed in order to search for the best architecture.

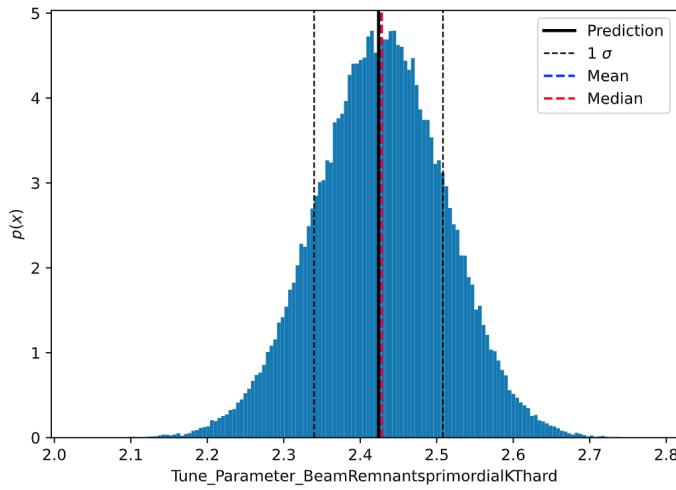


Figure 4.9: Predictions spread for the inverse model after that a Gaussian resample is performed

Hyperparameters

A really important step in the Inverse model is *hyperparameter optimization*. It is required to get the method working.

The procedure consists in building a *validation set* containing some Monte Carlo simulations as the training set (e.g. 10% of the simulations in the training set) and retrain the model with different choices for the NN architecture. Then a closure test is performed in order to estimate the performance of the NN. The test is performed using these validation set MC runs as input instead of the real experimental data, the closure test is done by computing a loss function between the predicted value and the Monte Carlo truth defined as:

$$L = \sum_i \frac{|p_i^{\text{true}} - p_i^{\text{pred}}|}{p_i^{\text{true}}} \quad (4.12)$$

The configuration that minimizes the distance between predicted and real values (MC truth), and so this loss function, is the best architecture in the selected hyperparameter space.

Once the best model is found, it is retrained using the MC runs contained in both the training and validation sets. Then the experimental data are fed to it in order to get the best estimation for the parameters to tune. This procedure is schematically summarized in Fig. 4.10.

The hyperparameters scan is performed using the python package `hyperopt`.

Problems

As shown in the next chapter the main problem for the Inverse model is the instability of the results in some cases. The operation that this model is trying to do is very hard the parameters can have some not trivial correlations. This complication can increase a lot the complexity in the operation of learning the inverse response of the generator. These difficulties in some cases lead to a bad prediction with parameters

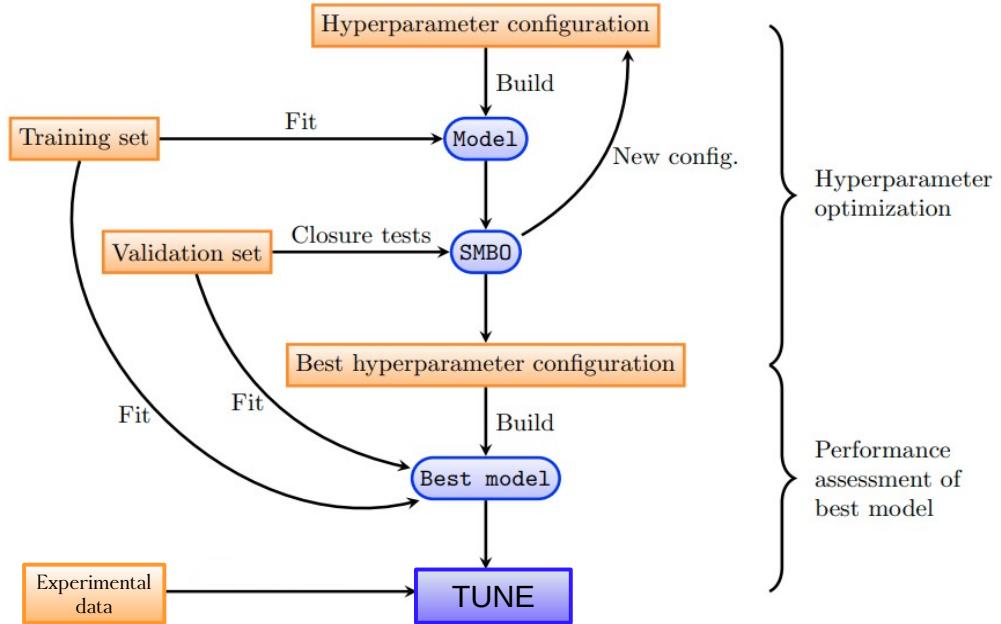


Figure 4.10: The hyperparameter optimization procedure is schematically shown here. The model is trained using a training set then performing a closure test a scan on the hyperparameter is done. Once the best configuration is found the best model is retrained using both the runs in the training set and the runs in the validation set. Then the experimental data are fed to the network and the best parameters are estimated. Figure from [7]

value predicted out of the imposed ranges and with errors in the order of the parameter value.

As it will be seen in the chapter that describes the new tune performed using the Inverse model in some cases when the number of parameters became larger and maybe the correlation between the parameters are important this model fails in the prediction.

But if more care is given to this model this can become a very powerful method for future tunes.

4.3.5 Weightrules in MCNNTUNES

MCNNTUNES as PROFESSOR implements the possibility of changing the weight of the singles bins in the distributions. This is a really useful feature in order to give some more importance to distributions and points that require a better description by the tune. As it will be discussed later these weightrules can be used to get results more similar to CP5 in the MCNNTUNES tune proposed in this thesis. This feature gives the possibility to decide which distribution is more important in our tune. Increasing the weight of a bin this bin became more important in the overall χ^2 evaluation and it is better described by the simulations.

Chapter 5

Tune for the Underlying Event

This section focus on our work in order to reproduce a similar tune to CP5 one for the underlying event and minimum bias obseravbles. This is done to test the ability of MCNNTUNES of being one valid tool for the tuning of Monte Carlo generator with real data.

In order to validate MCNNTUNES as a good tool for the tuning we decide to firstly performer a simpler tune with only two free parameters and then try to reproduce CP5 [6] with all the five parameters variation.

5.1 Introduction

We have performed a different tune for the underlying event and minimum bias observables using the same distributions, listed in Chapter 3.3.1 and used in CP5 tune, so we expect to get a similar result from the tune using MCNNTUNES.

Just a quick reminder for the PYTHIA8 settings used in CP5 and in our tune:

- The PDF set used is NNPDF3.1 calculated to the NNLO [49];
- an α_s value equal for all the processes set to 0.118 and running with a NLO evolution.
- The ISR is also ordered according to rapidity.

In the tuning procedure we employed both the MCNNTUNES operation modes described in the previous chapter: PerBin and Inverse models.

5.2 First test: only two parameters variation

The first simplified test we perform is the tuning varying of only two parameters. The parameter chosen are the `MultipartonInteractions:pT0Ref` and `MultipartonInteractions:ecmPow`. These two parameters have been introduced in Section 2.4.1. The aim of this simplified test is to check the correct operation of MCNNTUNES in fact in a restricted parameters space it is simpler to learn the generator behavior and predict the corrects best values for the parameters.

Table 5.1 shows the parameters space used in this first case. The other parameters are set to the CP5 values (also these are reported in the table as a reminder).

Parameter Name	Value
MultipartonInteractions:pT0Ref [GeV]	[1.0 – 3.0]
MultipartonInteractions:ecmPow	[0.0 – 0.3]
MultipartonInteractions:coreRadius	0.7634
MultipartonInteractions:coreFraction	0.63
ColorReconnection:range	5.176

Table 5.1: Parameters space for the two parameters test. The other parameters are set to the CP5 default values.

Note: during this first part we don't use the two distributions related to the single diffractive and non single diffractive event selection for a bug on the routine of the analyses that have been fixed up before the real tune.

Let now discuss the results obtained for the test with the two models.

5.2.1 Per Bin Model results

The result we obtain for the PerBin model are reported in Table 5.2. The PerBin model estimation of the best parameters is performed by a loss function minimization. The output of the minimizer is reported in Fig. 5.1, the loss function (χ^2/dof) is the orange one, while the best parameter is marked by a solid red line.

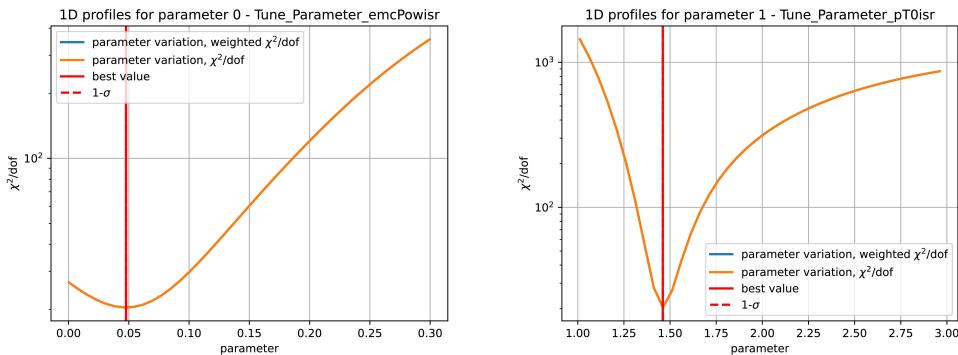


Figure 5.1: The figure shows the output of the minimizer for each parameter. The right one refers to the MPI:ecmPow while the left one to the MPI:pT0Ref. The orange line is the χ^2/DoF as a function of the parameter values, the solid red line indicated the best estimation for the parameters.

The estimated parameters for the tune are also reported in Table 5.2 with a comparison on CP5 boundary for each parameter. To be notice that the errors on the parameter estimated with PerBin model are not reported here because a correct error implementation, as the one described in Section 4.3.2, was not yet implemented in the software at the moment of this test, but here we can see an example of the old error estimation. The old errors estimation process was not working in fact the errors lines (dashed red lines) are overlapped to the line of the parameter estimation in both the figures. The predicted errors were 3 or 4 orders of magnitude smaller than the parameter values.

But as we expected the results we get are similar to the ones obtained from the

CP5 tune. The distributions obtained from the simulation using this parameters are shown below (Section 5.2.3) together with results from Inverse model and CP5.

Parameter	Value	CP5 (down & up)
MultipartonInteractions:pT0Ref	1.46064	1.41 – 1.46
MultipartonInteractions:ecmPow	0.04771	0.03

Table 5.2: Results for the PerBin model in two parameter variation test. The error evaluation was not correctly implemented yet and so the PerBin errors omitted in this table. The upper and lower limit for CP5 are also reported here for a direct comparison between the two tunes.

An important observation is that our model is more sensible to some parameter respect to others.

As an example, in Fig. 5.2 it is clear that our model is more sensible to the parameter `MultipartonInteractions:pT0Ref` (top) respect to `MultipartonInteractions:ecmPow` (bottom). This is reflected in the output of the minimizer a greater sensibility gives a better defined minimum and vice versa. The top two panels show what happen when the distribution are very sensitive to the variation of the parameter as we can see on the left side the variation of the `pT0Ref` parameter leads to very different scenarios. This is reflected by the minimizer out in a well defined minimum and a smaller error on the parameter determination. A different scenario is displayed on the bottom panels of Fig. 5.2 where the distribution is less sensitive to the variation of the parameter `ecmPow` and so the minimum is less defined and so we are going to have a larger error on the estimation.

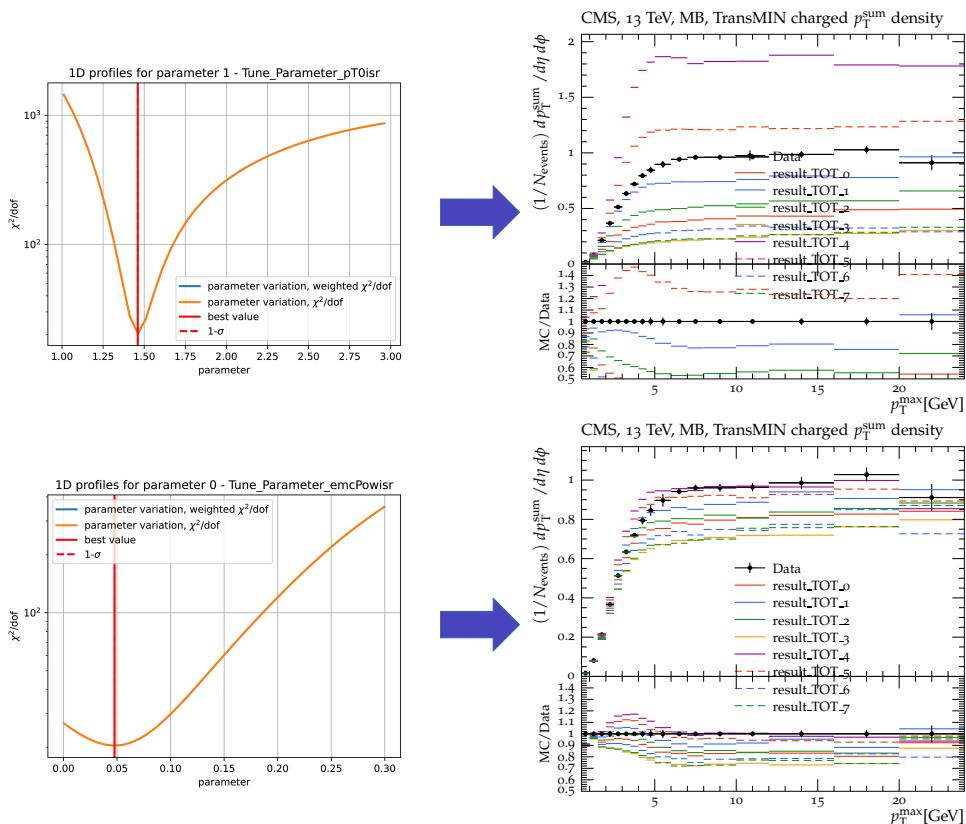


Figure 5.2: the sensibility of the tuned distributions respect to the variation of the parameters in related to the output of the minimizer. The top panels are related to `pT0Ref` while the bottom ones to `ecmPow`. The different colored lines in the 2 distributions (on the right) are the simulations with different values for the parameter. The black point are the data.

5.2.2 Inverse Model results

The other operation mode offered by MCNNTUNES is the Inverse model. The results we get from the first test of the Inverse model are described here while the overall distribution are reported in next section together the ones from the PerBin Model. As mentioned before to make this model work properly one have to perform a hyperparameters optimization. our hyperparameter optimization was a scan of the architecture parameters shown in Table 5.3 the number of trials (combinations of these parameters) was 1000. in this case the best model we found is the one reported in Table 5.4.

Hyperparameter	Variation Range
Number of hidden layer	2-5
Units per layer	2-20
Activation function	tanh, relu, sigmoid
Optimizer	sgd, rmsprop, adagrad, adadelta, adam, adamax, nadam
Epochs	250-15000 in discrete steps
Batch size	64-5000 in discrete steps
Number of trials	1000

Table 5.3: Hyperparameter space scanned for the optimization of the NN architecture.

Hyperparameter	Value
Number of hidden layer	4
Units layers	[2, 14, 9, 18]
Activation function	tanh
Optimizer	nadam
Epochs	1000
Batch size	500

Table 5.4: Best hyperparameters model found for the test with 2 parameters variation.

Once the best model is trained the output we get from this model is the distribution of predictions obtained by the resampling phase of the experimental data then fed to the network. The prediction spread for the two parameters test is shown in Fig. 5.3 these are obtained from the re-sampling phase using the multivariate Gaussian distribution described in Eq. 4.11.

The estimated parameter are marked by the solid black line, while the dotted black lines are the error on the parameter. The value we get from the tune are also reported in Table 5.5 together with the CP5 values. In this case the error estimation is correctly implemented and a direct comparison between the value is possible. Looking at the values reported in the table it is clear that the two tune are compatible.

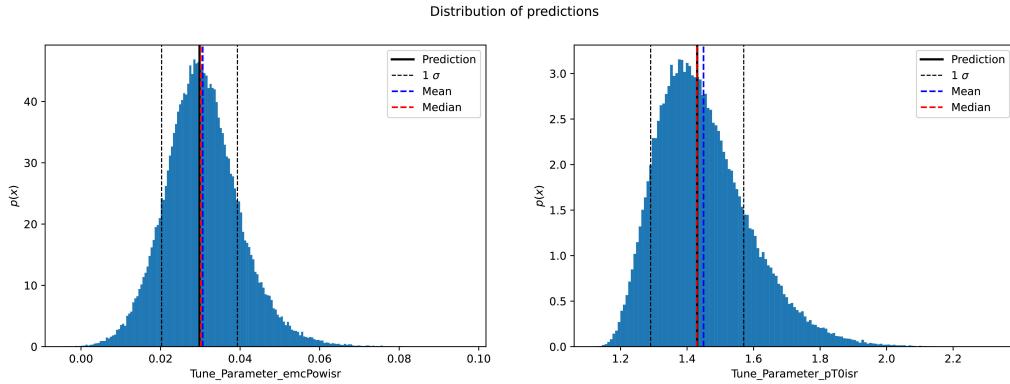


Figure 5.3: The spread of predictions we get as output from the Inverse Model. the right one refers to the `ecmPow` parameter and the left one to `pT0Ref`. The parameters predicted are marked by the solid black line while the error is evaluated using the standard deviation indicated by the dashed black lines.

Parameter	Value	CP5 (down & up)
MultipartonInteractions:pT0Ref	1.43 ± 0.14	$1.41 - 1.46$
MultipartonInteractions:ecmPow	0.0298 ± 0.0095	0.03

Table 5.5: Results for the Inverse model in two parameter variation test. The upper and lower limit for CP5 are also reported here for a direct comparison between the two tunes. The two predicted parameters are compatible to the one in CP5.

5.2.3 Overall results

In this section we are going to show some overall results for the first test with only two free parameters. We are not going to show all the graphs here we list them all in the appendix Appendix A.

From the graphs in Fig. 5.4 is clear that MCNNNTUNES gives some good result for the test.

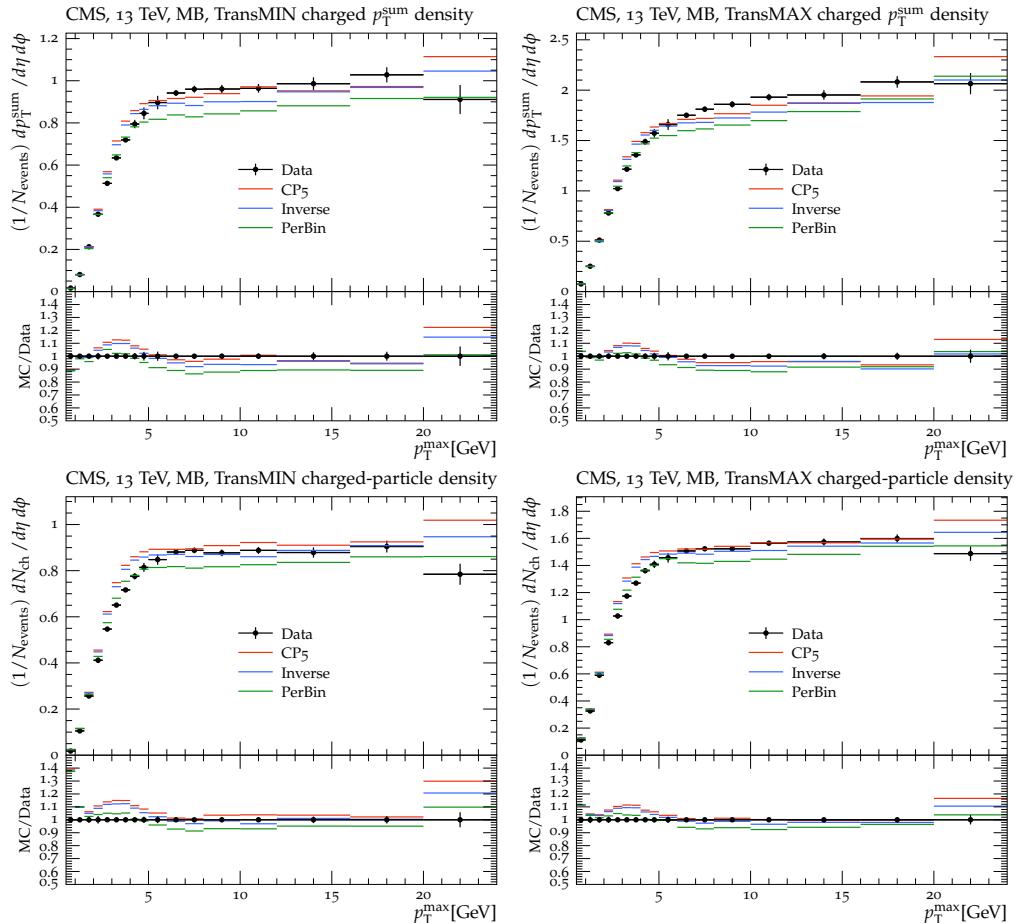


Figure 5.4: Results for the two parameters variation test. Here are reported the distributions from the $\sqrt{s} = 13$ TeV CMS analysis [25] that show the trans-MAX charged particle density (upper left) and the charged p_T -sum density (upper right); the transMIN charged particle density (lower left) and the charged p_T -sum density (lower right) as a function of the transverse momentum of the leading charged particle. The black points are the experimental data and the black vertical lines the experimental uncertainties. The data are compared to the MC prediction from the result we get using PerBin Model (green line), Inverse Model (blue line) and the existing tune CP5 (red line). The data are well described by all the tunes. Our tune describe very well the low- p_T region ($p_T \lesssim 5$ GeV).

The result we get from PerBin model (green line) and from Inverse model (blue line) are similar to the result obtained from CP5 tune (red line). our tune describe the low region ($p_T \lesssim 5$ GeV) very well this region is very important because is the region with lower error on the experimental data.

Overall, we can consider this test successfully passed from MCNNNTUNES, it gives

result similar to the standard tool PROFESSOR. This was only a simplified test to check the correct operation for the tool. So, passed the test we decide to extend our analysis to a complete tune for the underlying event.

5.3 Tune for the UE in Minimum Bias Observations

Given the good results for the test we extend our analysis to the variation of five parameters. The interested parameter are the ones related to the Multi Parton Interaction and to the Color Reconnection. The ranges of variation for these parameters are the same used for CP5 and summarized in Table 5.6.

Parameter Name	Value
MultipartonInteractions:pT0Ref [GeV]	[1.0 – 3.0]
MultipartonInteractions:ecmPow	[0.0 – 0.3]
MultipartonInteractions:coreRadius	[0.1 – 0.95]
MultipartonInteractions:coreFraction	[0.1 – 0.8]
ColorReconnection:range	[1.0 – 9.0]

Table 5.6: The variation ranges for the five parameters that we want to tune. These are the same used in the CP5 tune in [6].

As the parameters space increase we need to increase also the number of samples and then the size of the training set in order to have a sufficient granularity in the sampling of the space. The training set we use for the PerBin model with five parameters variation is composed from approximately 2000 MC runs, instead for the Inverse model we try also larger training set.

5.3.1 Per Bin Model results

The PerBin model is the model that give to us the best results, the parameters estimation we get from the PerBin model loss function minimization is reported in Fig. 5.5. In the figure the five parameters χ^2/DoF functions are reported with a blue line. The predicted value is indicated by the solid red line while the $1 - \sigma$ range with the dotted lines. It is clear that also in this case the most sensible parameter is the `MultipartonInteractions:pT0Ref` (5.5e) the minimum in that case is very well defined and the error small. The parameters in Fig. 5.5c and Fig. 5.5d are also well defined the error is not to big. The not so good defined parameters are the `MultipartonInteractions:coreFraction`, that is in a minimum but with a larger error than the other parameters which is why the distribution under analysis are less sensitive to the variation of this parameter, and the `ColorReconnection:range`. The last one is not actually in a real minimum, in this case also the error evaluation, as a confidence interval is not possible. This indicated a saturation of this parameter, over a certain value all the possible reconnection have already occurred, and so a consequent small sensitivity to this parameter variation.

The value we get for the parameter are reported also in Table 5.7 and compared to the CP5 limits. It is easy to see that all the parameters are compatible with the CP5 except for the `ColorReconnection:range`.

5.3. TUNE FOR THE UE IN MINIMUM BIAS OBSERVATIONS

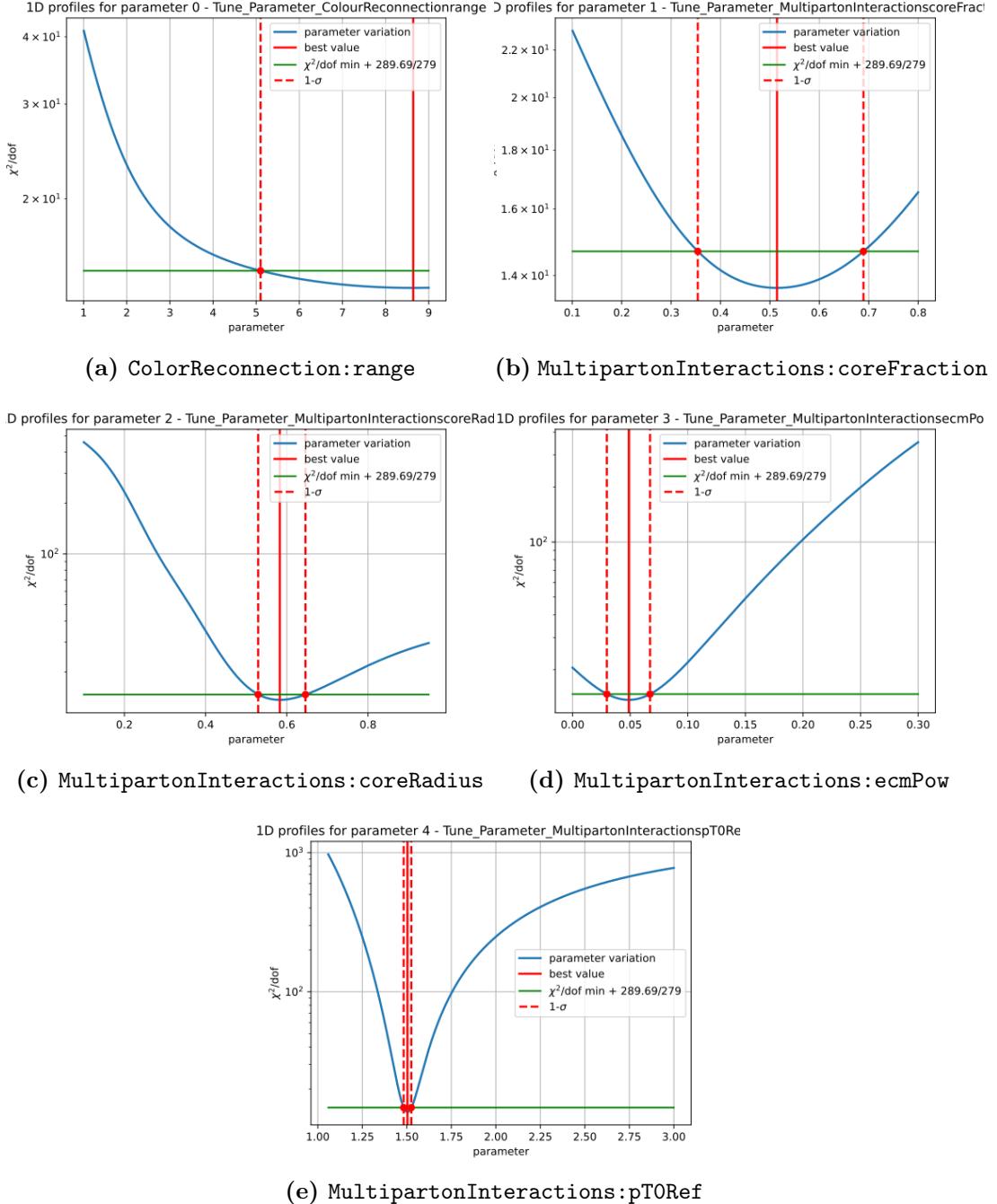


Figure 5.5: The minimizer output for every tuned parameter. The blue line is the χ^2/DoF as a function of the parameter value. The solid vertical line indicate the best estimation for the parameter value while the errors are indicated by the dashed red lines. In the graph (a) it is clear that a saturation in the parameter is reached after a certain value this lead to a small sensitivity to the parameter variation and so a non-well defined minimum. The upper limit in this case cannot be calculated. The other parameters are all in a real minimum and with a concrete error evaluation.

The distribution we get are reported in the Section 5.3.3 as we can see in the Fig. 5.13 the PerBin model does not describe very well this distribution for the

Parameter	Value	CP5 (down & up)
MultipartonInteractions:pT0Ref	$1.50^{+0.02}_{-0.02}$	1.41 – 1.46
MultipartonInteractions:ecmPow	$0.049^{+0.018}_{-0.019}$	0.03
MultipartonInteractions:coreFraction	$0.51^{+0.17}_{-0.16}$	0.43 – 0.73
MultipartonInteractions:coreRadius	$0.58^{+0.06}_{-0.05}$	0.67 – 0.69
ColorReconnection:range	$8.6^{-3.5}_{+null}$	4.88 – 4.69

Table 5.7: The results of the tune using PerBin Model. All the values are compatible with the one with the CP5 using a Z test with a significance level of 0.05. But the value predicted for the Color Reconnection have a very large error and is not as similar to the one obtained from CP5. The $null$ subscript indicates that the definition of the confidence region exceed the scanned space for the parameters defined for each parameter in Table 5.6.

pseudorapidity of the inelastic production of hadrons. A possible explanation is that the experimental uncertainties on the bins of this distribution are higher than the ones in others and so this distribution be less important in the overall loss function.

So, we decide to perform a second tune using PerBin Model. In the second tune we give to all Fig. 5.13 bins an higher weight using the weightrules implemented in MCNNTUNES, we give a weight of 5 to all bins of this distribution. In this way we are giving a greater importance to this distribution and so we are going to describe those data better.

The output of the mimization step is reported in Fig. 5.6. The value we get from the weighted tune are also reported in Table 5.8. The values we get are compatible also in this case with CP5 except for the MultipartonInteractions:coreRadius that is higher than the one predict from CP5.

But if we look at the overall result in Section 5.3.3 the PerBin Model with different weights give a result more similar to CP5 in almost all the distributions.

Parameter	PerBin	PerBin + Re-weight	CP5 (down & up)
MPI:pT0Ref	$1.50^{+0.02}_{-0.02}$	$1.42^{+0.01}_{-0.01}$	1.41 – 1.46
MPI:ecmPow	$0.049^{+0.018}_{-0.019}$	$0.0342^{+0.014}_{-0.014}$	0.03
MPI:coreFraction	$0.51^{+0.17}_{-0.16}$	$0.34^{+0.14}_{-0.17}$	0.43 – 0.73
MPI:coreRadius	$0.58^{+0.06}_{-0.05}$	$0.9^{+null}_{-0.1}$	0.67 – 0.69
CR:range	$8.6^{-3.5}_{+null}$	$5.6^{+1.0}_{-0.9}$	4.88 – 4.69

Table 5.8: The results of the tune using PerBin model + re-weight are compared to the ones with PerBin model and CP5. except for the coreRadius parameter, all the predicted values are compatible with the one with the CP5 using a Z test with a significance level of 0.05. The PerBin + re-weight gives results more similar to the one obtained from CP5 this is also clear watching to the overall resulting distribution. The $null$ subscript indicates that the definition of the confidence region exceed the scanned space for the parameters defined for each parameter in Table 5.6

5.3. TUNE FOR THE UE IN MINIMUM BIAS OBSERVATIONS

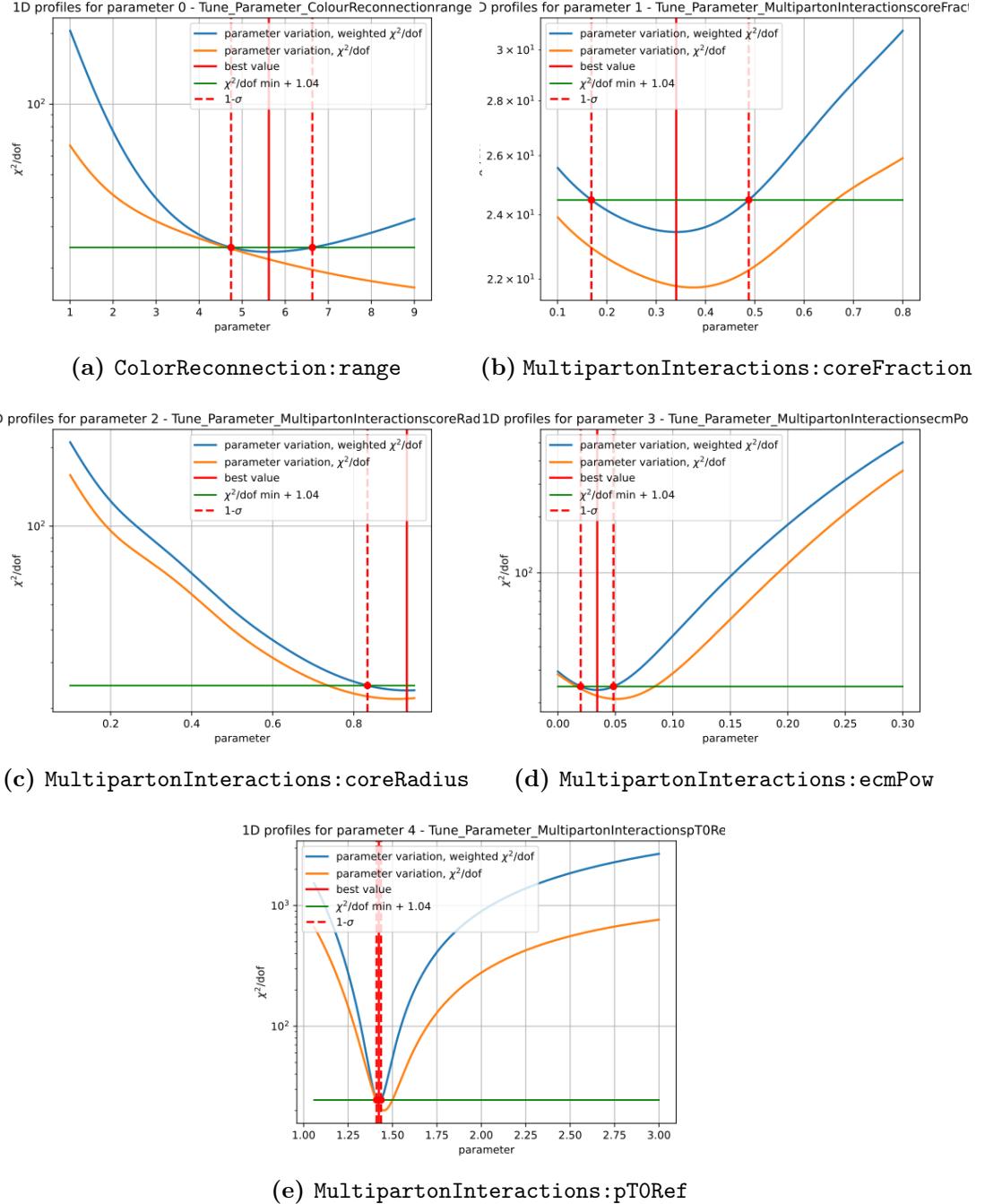


Figure 5.6: The minimizer output for every tuned parameter using PerBin model with the re-weight. The blue line is the χ^2/DoF as a function of the parameter value before the re-weight while the orange line indicate the same function but applying the re-weight. The solid vertical line indicate the best estimation for the parameter value while the errors are indicated by the dashed red lines. The parameter in figure (c) is predicted near the upper limit of the variation range.

5.3.2 Inverse Model results

On the other hand if the PerBin model gives to us very good result, we cannot tell the same for the Inverse Model. In this case the Inverse Model gives very bad results. Also after the hyperparameters optimization, using the scan space described in Table 5.3, the model fails.

The distributions of predictions are reported for each parameter in Fig. 5.7 and the actual parameters in Table 5.9 as we can see the predicted values are in most of the cases out of the variation ranges set for the sampling and (also to the maximum possible value in PYTHIA8). Some other parameter are determined with a very large distribution and so very large errors.

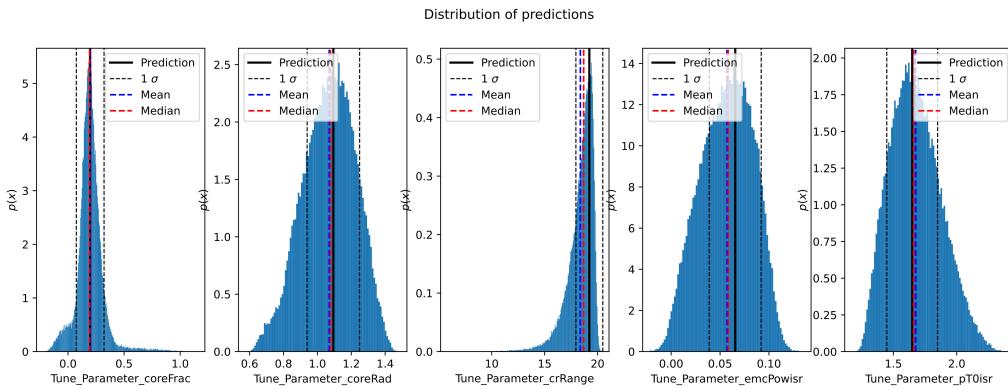


Figure 5.7: The spread of prediction obtained from the Inverse model. It is clear that also after the hyperparameters optimization the inverse model is not working properly. The distribution have long tails out of the limits for the variation, the central one refered to the Color Reconnection is predicted completely out of the boundaries.

Parameter	Value	CP5 (down & up)
MultipartonInteractions:pT0Ref	1.65 ± 0.20	$1.41 - 1.46$
MultipartonInteractions:ecmPow	0.066 ± 0.026	0.03
MultipartonInteractions:coreFraction	0.20 ± 0.12	$0.43 - 0.73$
MultipartonInteractions:coreRadius	1.1 ± 0.2	$0.67 - 0.69$
ColorReconnection:range	19.2 ± 1.3	$4.88 - 4.69$

Table 5.9: The predicted values using Inverse model are showed here, the model is not working in this case the predicted values for the core radius and for the color reconnection are out of the boundary while other parameter are predicted with a quite large error.

We can see from Fig. 5.8 the tune we get from the Inverse Model (blue line), as was expected, cannot describe the observables distributions. The tune misses all the experimental data points approximately by a 50% of the value. Se we have to exclude this model for this tune. Maybe the failure is related to the higher number of parameters respect to the first test, that leads to a more complex generator response and so a more difficult model to learn and invert.

In MCNNTUNES presentation paper [7] is reported that the Inverse model can fail when in the training set is not given a sufficient number of MC simulations with parameters values near to the actual real ones. So, trying to make this model work, we decide to introduce a Gaussian bias in the sampling phase instead of a uniformed distributed sampling. Our Gaussian sampling was peaked on the parameters predicted by the PerBin Model, but also this procedure does not leads to some consistent results.

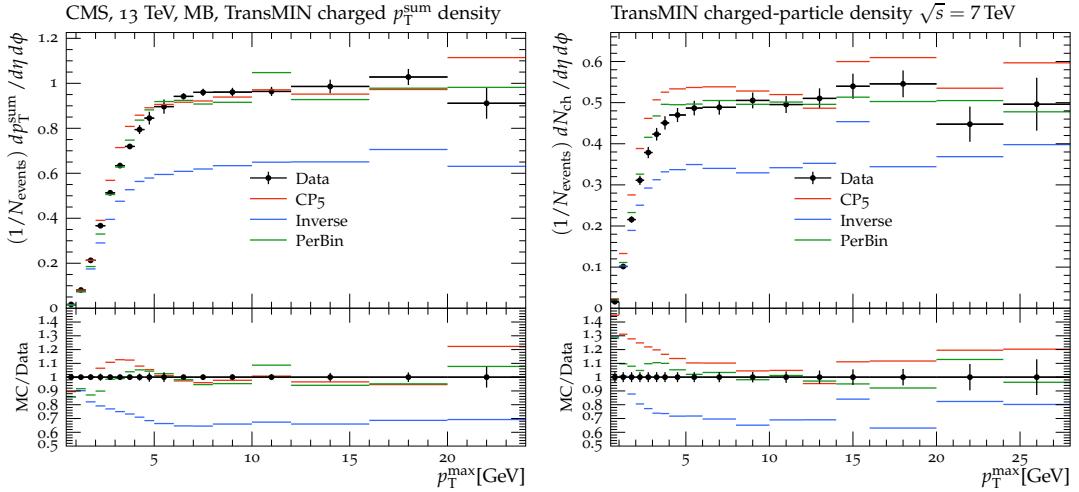


Figure 5.8: An example of the fact that the Inverse model (blue line) is not working: the bins are filled to only the 50% of the expected value in almost all the bins. On the left is shown the transMIN charged particle transverse momentum sum density for the CMS analysis at $\sqrt{s} = 13$ TeV [25] and on the right the transMIN charged particle density from the CMS analysis [26] in both case as a function of the leading object transverse momentum.

5.3.3 Overall results

Overall, what we have are two tunes performed with the PerBin Model that are quite good tunes. In this section all the fitted distributions are shown and the two tunes are compared to the distribution obtained with the CP5 default values. The PerBin Model describe very well the data in particular in all the low- p_T regions where the experimental uncertainties are smaller and so more important in the χ^2 evaluation.

The first distributions showed in Fig. 5.9 are referred to the charged particle multiplicity and the charged particle scalar p_T sum in the two transverse regions (Trans-MAX and TransMIN) as a function of the leading object transverse momentum. The black points are the experimental data taken from the CMS experiment at the center of mass energy of 13 TeV. They are compared to the CP5 tune, red line, and the two tunes we have gotten from the PerBin Model and PerBin Model plus the re-weights, respectively blue and green lines. It is easy to see that all the three tunes are describing the distribution very well, in particular our tunes in the low- p_T regions ($p_T < 5$ GeV) are describing the distribution also better than CP5 in most of the cases.

Instead, in Fig. 5.10 are reported the same distribution but at $\sqrt{s} = 7$ TeV also in

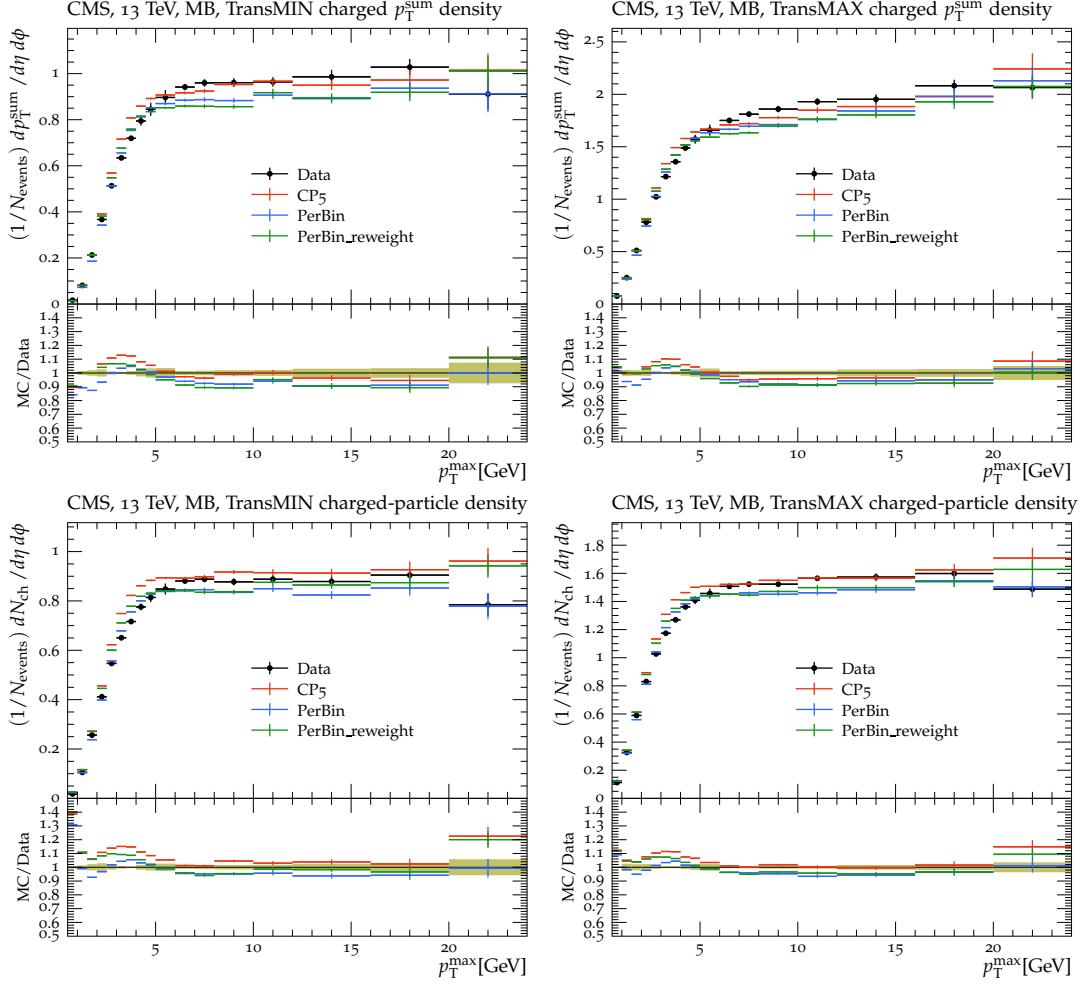


Figure 5.9: In this figure the data from the $\sqrt{s} = 13$ TeV CMS analysis [25] that show the transMAX charged particle density (upper left) and the charged p_T -sum density (upper right); the transMIN charged particle density (lower left) and the charged p_T -sum density (lower right) as a function of the transverse momentum of the leading charged particle. The CP5 tune is compared to our tune using the PerBin Model. Our tune (red line) seems good as the CP5 (blue line) in describing the data. The first bins are the most important they have a smaller experimental error than the higher p_T data. Also the PerBin model with re-weight (green line) seems really good in the description of the data. Also the ratio between MC and data points is reported and the green band represent the experimental uncertainties, while the vertical colored lines on the MC points are the statistical uncertainties.

this case our tunes are good in describing the distributions, they seem to be also better than CP5.

Also the data at the center-of-mass energy of 1.96 TeV are well described. These data had been collected by the CDF experiment in proton-antiproton collisions.

Fig. 5.12 and Fig. 5.13 represent the pseudorapidity distribution for diffractive events and for inelastic charged hadrons production. The left distribution of Fig. 5.12 and the distribution in Fig. 5.13 are not well described by the PerBin Model (blue line) but are better described by the PerBin Model plus the re-weights for these bins. So

5.3. TUNE FOR THE UE IN MINIMUM BIAS OBSERVATIONS

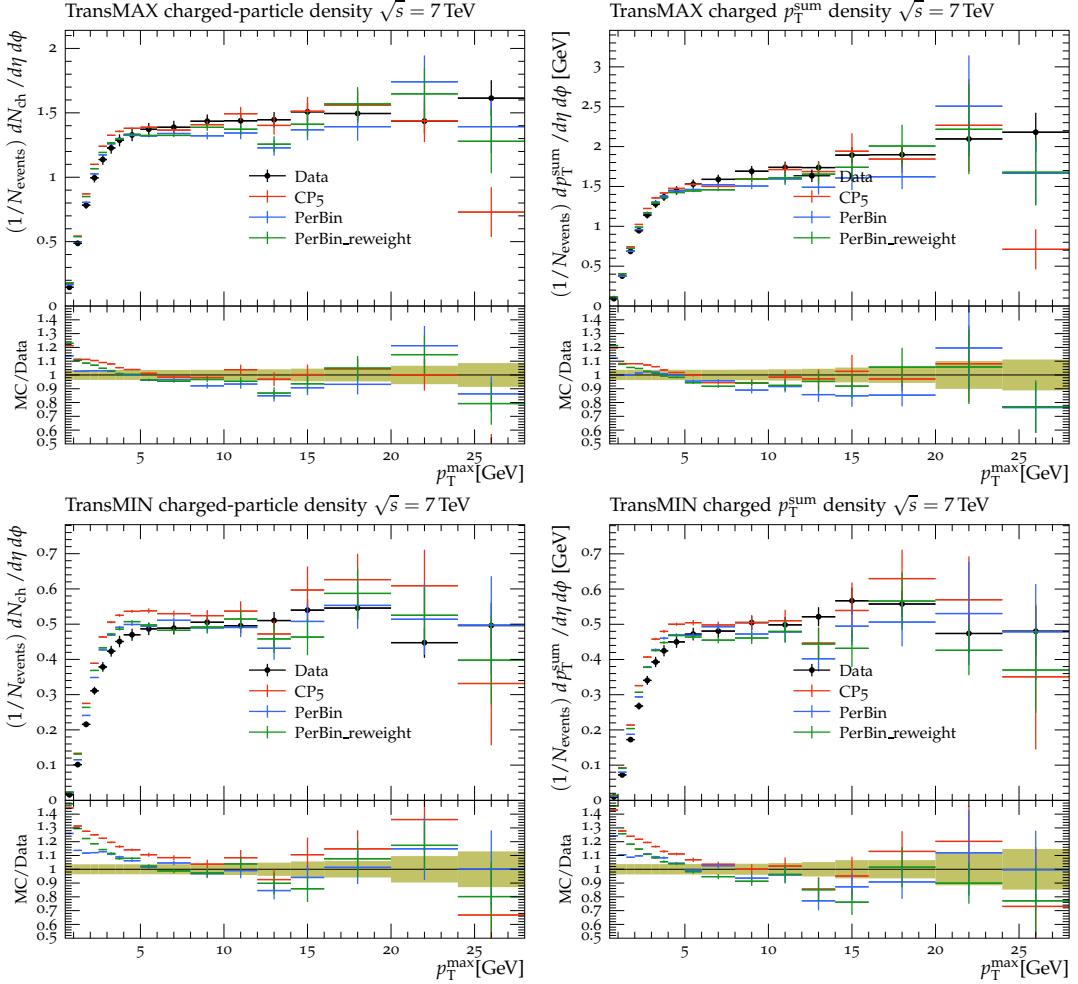


Figure 5.10: Here the data at $\sqrt{s} = 7$ TeV from the CMS analysis [26] for the transMAX charged particle density (upper left) and the charged p_T -sum (upper right); the transMIN charged particle density (lower left) and the charged $p - T$ -sum are displayed as a function of the leading object transverse momentum. The three tunes describe the data (black point) very well. The low $p - T$ regions are described better from our tune respect to the CP5 tune. As was expected the result from the PerBin model with the re-weights (green line) are more similar to the CP5 result. Also the ratio between MC and experimental points is reported and the green band represent the experimental uncertainties, while the vertical colored lines on the MC points are the statistical uncertainties.

also in this case we have a good result from MCNTUNES tool.

A numerical evaluation on the overall difference between MC and experimental points can be evaluated for all the three tunes using the χ^2/DoF definition:

$$\chi^2 = \sum_i \frac{(MC_i - exp_i)^2}{\sigma_i^2} . \quad (5.1)$$

The results we get are reported in Table 5.10. From this is clear that from this evaluation the better tune overall is the PerBin model tune. In fact, the PerBin Model describes very well the distributions were the experimental uncertainties are smaller but don't describe very well the last distributions in the left panel of Fig. 5.12 and

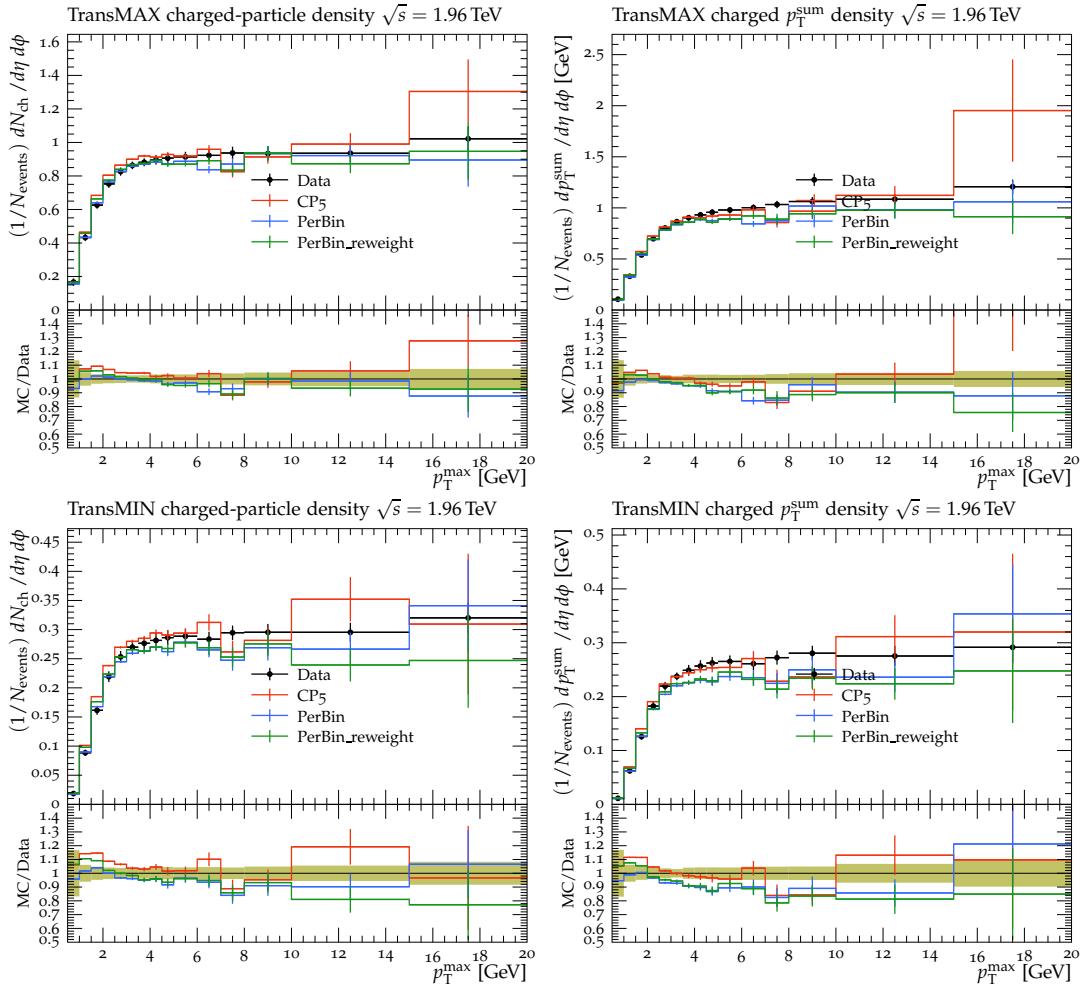


Figure 5.11: The transMAX charged particle density (upper left) and the charged p_T -sum (upper left); the transMIN charged particle density (lower left) and the charged p_T -sum from the CDF analysis at $\sqrt{s} = 1.96$ TeV in proton-antiproton collisions [27]. The data are described very well from the CP5, PerBin, PerBin + re-weights tunes. Also the ratio between MC and experimental points is reported and the green band represent the experimental uncertainties, while the vertical colored lines on the MC points are the statistical uncertainties.

Tune	χ^2/DoF
CP5:	23.9
PerBin:	13.7
PerBin + re-weights:	19.4

Table 5.10: χ^2 evaluation for the three tunes.

Fig. 5.13. A very good result is also the one obtained from the PerBin Model with the different weights for the bins in the distribution in Fig. 5.13. This tune in the idea of having a more general tune that describe better all the distributions can be considered as a better tune. In general the χ^2 evaluated is less in both our tunes respect to the CP5 one.

5.3. TUNE FOR THE UE IN MINIMUM BIAS OBSERVATIONS

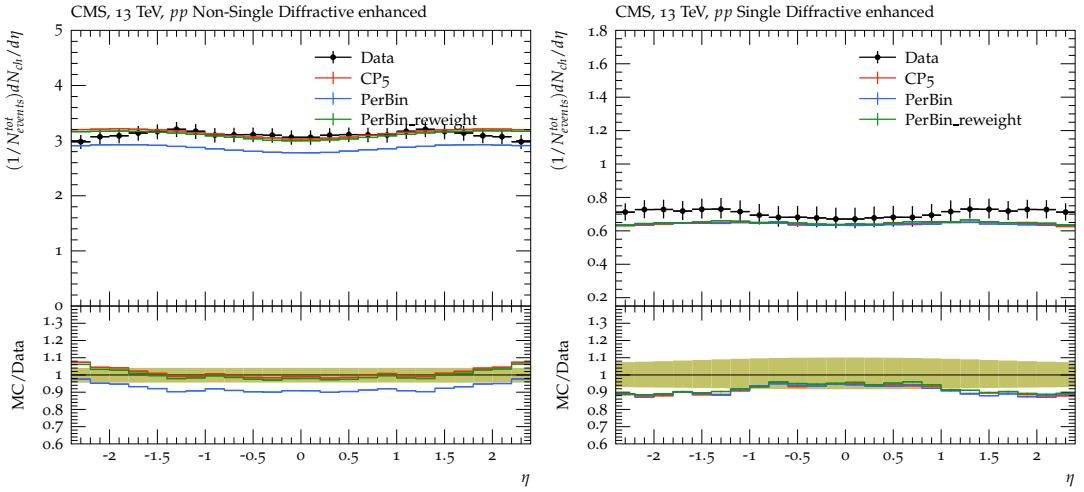


Figure 5.12: Here are reported the pseudorapidity distributions ($p_T > 0.5$ GeV, $|\eta| < 2.4$) for charged particle multiplicity in single diffractive (right) and non-single diffractive (left) events selection. The black point are the data from the CMS analysis at $\sqrt{s} = 13$ TeV [31]. The data from the NSD events are not so well described from the PerBin model (blue line) but with the re-weights we can describes these data points better. Instead, for the SD events we get a result equal to the one of CP5. Also the ratio between MC and data points is reported and the green band represent the experimental uncertainties, while the vertical colored lines on the MC points are the statistical uncertainties, in these distributions the statistical uncertainties are very small due to the high number of events in each bin.

Given these results we can conclude that MCNTUNES is valid tool for the tune of the parameters in high energy physics generators. We obtained two valid tunes for the underlying event and minimum bias observations in proton-proton collisions. The activity observed in the two transverse regions and the pseudorapidity distributions for different events selections are well described after the tune of the parameters that control the multi parton interactions and the color reconnection in PYTHIA8. The output we got from the minimizer in Fig. 5.5 tell to us that the most important parameter for the description of this distributions is the threshold value p_{T0}^{ref} , all the distributions have a large sensitivity on the variation of this parameter and this is related to the well defined minimum we got.

On the other hand the small dependence on the variation of the `ColorReconnection:-range` parameter over a certain threshold indicate a saturation, in fact over some value almost all the possible color reconnection have taken place and so the model is less sensitive to this parameter.

The Inverse model instead in this case don't give us a good result but maybe further tests can take to some good results. Maybe increasing even more the training set size but with careful, MCNTUNES does not present any control on the NNs typical over-fitting problem. Even if the Inverse Model don't give us a complete tune for the underlying events it perform very well in the first test. So, we cannot exclude it as a valid tool.

Now, that the tool is validated we are going to extend the tune procedure to some new distribution and parameter. In the next chapter we focus on the tune of the

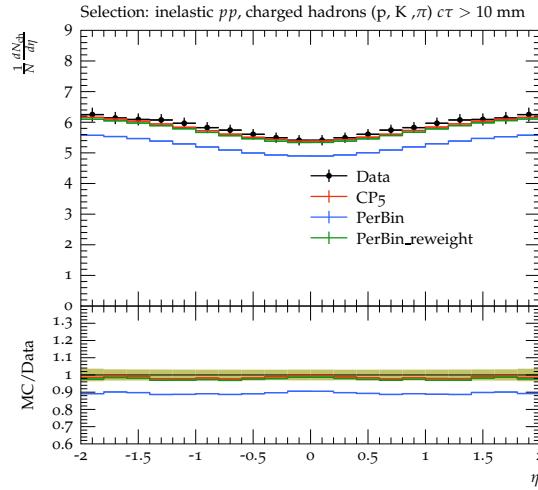


Figure 5.13: In this figure is shown the last distribution we use for the tune from the CMS analysis at $\sqrt{s} = 13$ TeV [32]. The pseudorapidity distribution ($|\eta| < 2$) for the charged hadron density in an inelastic proton-proton scattering selection. Also here the PerBin model (blue line) give a different result from the CP5 and the PerBin + re-weights tunes. Also the ratio between MC and data points is reported and the green band represent the experimental uncertainties, while the vertical colored lines on the MC points are the statistical uncertainties, in this distribution the statistical uncertainties are very small due to the high number of events in each bin.

Primordial k_T and Space Shower in order to improve the description of the data collected in Z boson production events.

Chapter 6

Tune for the primordial k_T events

The primordial k_T is another important parameter in the description of the proton-proton collisions with Monte Carlo simulations. The primordial k_T description in PYTHIA8 was introduced in Section 2.4.4.

The main unresolved problem for the primordial k_T tune is the unexpectedly high value required for the description of the observed Z boson p_T spectrum.

In fact, the primordial k_T is derived by the Fermi Motion of partons inside the hadrons. So when the parton undergoes the hard scattering it can already have an initial non zero transverse momentum. The value of the primordial k_T can be estimated as reported in Eq. 2.38, but experimental data for the Z boson p_T spectrum show that this estimation is not sufficient for the required value observed in order to reproduce the experimental data are in the order of 2 GeV.

The primordial k_T was tuned in the Monash tune [39] and the following tunes, as CP5, inherited the values of the parameters from there.

So the introduction of a new tune for the primordial k_T is needed. In fact the CP5 tune it is known that does not describe well the Z spectrum in the low- p_T region [6]. This is shown in Fig. 6.1 for the production cross-section in Z plus jets events and in the figure Fig. 6.2 for Z boson production in DY observations. As we can see the CP5 tune is very far from experimental data values in the low regions.

The distribution for Z +jets events as a function of the variable $p_T^{bal} = |p_T^Z - \sum_{\text{jets}} p_T^j|$ is not well described in the first bin. While the one as a function of the p_T^Z misses the description in the region $p_T^Z \lesssim 40$ GeV. These two observables as described in [6] are very sensitive to the parton shower process and so to the UE.

For the tune we focus on the distributions in Fig. 6.2 where the effects of MPI are expected to be less important¹, also here we can see that CP5 tune does not describe very well the low regions for these two observables.

6.1 Primordial k_T and ISR effect on p_T^Z

The parameters on which we focus in order to tune data and explain the Z boson p_T spectrum are:

¹below it is preliminarily investigate the effect of the MPI on these distributions. This aspect is never been investigated in detail and requires further studies maybe also the inclusion of data from LHC RUN3 can improve the description of these observables.

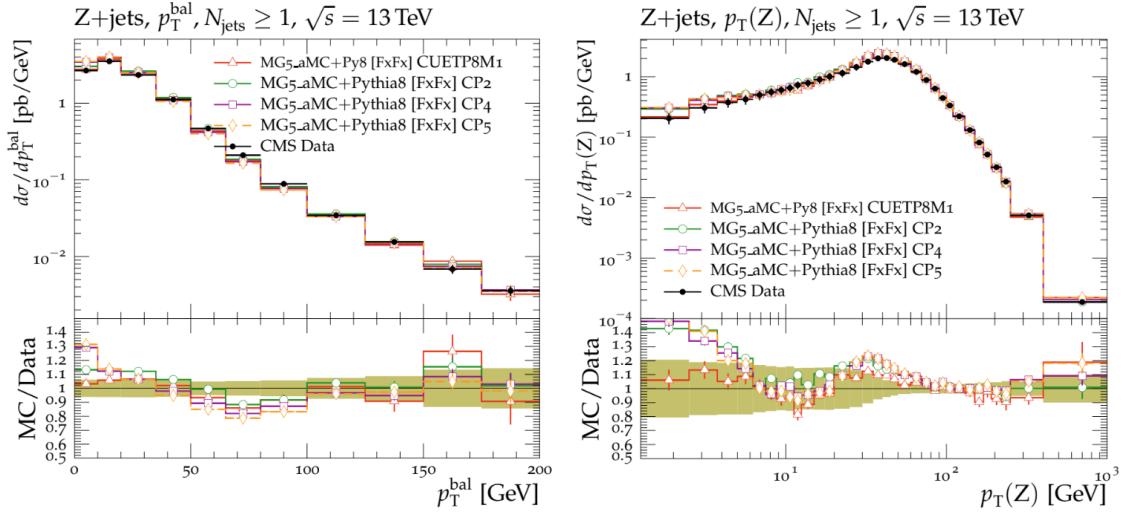


Figure 6.1: Figure from CP tunes presentation paper [6]. These two images show the Z boson production cross-section in Z plus jets (with at least one jet) at $\sqrt{s} = 13$ TeV as a function of the imbalance of the transverse momentum between the jet and the Z boson (left) and of the Z boson transverse momentum (right) [50]. The first bin in the left distribution and the region $p_T^Z \lesssim 40$ GeV, in the right one, is not well described by the CP5 tune (yellow line).

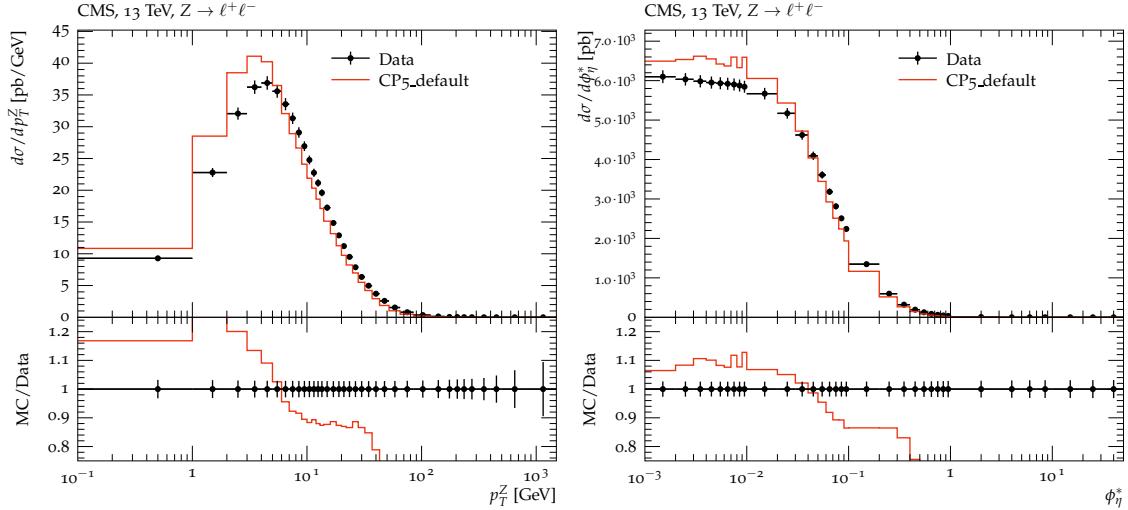


Figure 6.2: The CP5 tune is not good in the description of the low region of the Z boson production cross-section of the Z boson transverse momentum (left) or of the ϕ_η^* angle. This distributions showed here are from the CMS analysis [38] at the center-of-mass energy of 13 TeV.

- BeamRemnants:primordialKThard that set the width of the Gaussian distribution for the primordial k_T sample;
- SpaceShower:pT0Ref that set the threshold for the initial state radiation to take place.

Now, it is discussed discuss how this parameter impact the Z boson production transverse momentum spectrum. If we consider only the LO diagram for the Z pro-

duction (Fig. 6.3a) we can have only a production with zero transverse momentum. So the Z spectrum we expect at LO is a δ -distribution function centered on zero.

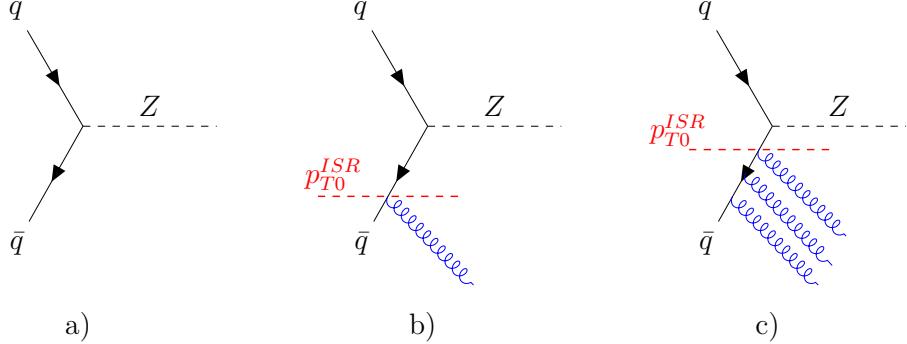


Figure 6.3: Z boson production diagrams: a) shows the LO diagram; b) is the LO diagram with a high threshold for the ISR so we have a small amount; c) is the same but with a lower threshold for the ISR and a larger amount, in b) and c) cases the Z boson can be produced with a non-zero transverse momentum, balanced by the ISR.

With the addition of the primordial k_T we get a Gaussian distribution whose width is set by `BeamRemnants:primordialKThard`. If we move to higher-order diagrams the Z boson can be produced with a non zero p_T that is balanced by the various jets, then the primordial k_T is added to the already non-zero Z boson transverse momentum.

A large contribution come also from the amount of ISR that is emitted from the incoming partons, in fact, each split can give to the incoming parton a non-zero initial value of p_T and so the Z boson is created with a non-zero p_T before the introduction of the primordial k_T .

A higher amount of ISR, so a lower value for the threshold, can lead to a larger p_T taken from the parton that then can generate the Z boson with a higher p_T and vice versa.

6.2 Primordial k_T tune

The primordial k_T was not investigated by CP5 so it is important to include that in the analysis in order to describe better the low p_T region.

The parameters variation ranges we use are the ones reported in the Table 6.1 while other parameters are set to CP5 values.

Parameter Name	Value
<code>BeamRemnants:primordialKThard</code>	[0.5 – 5.0]
<code>SpaceShower:pT0Ref</code>	[0.5 – 5.0]

Table 6.1: Variation ranges for the sampling used in the primordial k_T tune.

The number of samples is lower than the one used for the Underlying Event in fact the training set we use for the PerBin Model contains more or less 160 MC runs

and a little more for the Inverse Model, ~ 200 , this is related to the lower number of parameters to tune. So, the operation of the tune was computationally faster: less Monte Carlo jobs to run.

Fig. 6.2 shows the distributions used to perform the tune. The distributions are from the [38] analysis at $\sqrt{s} = 13$ TeV:

- The Z boson production cross-section in DY observation as a function of p_T^Z ;
- The Z boson production cross-section in DY observation as a function of ϕ_η^* .

Note that: the region of interest is the low p_T region. The simulation of the whole spectrum requires the adoption of a higher-order merging scheme between the higher-order matrix element calculations and parton shower, as FxFx, which is computationally expensive. So the strategy followed was the tuning of the low region, the one of interest, taking only the first five bins from each distribution.

The PerBin Model minimization results are displayed in Fig. 6.4. The blue line indicates the value of the χ^2/DoF as a function of the parameter value. The best estimation is marked by a solid red line while the dashed red lines indicate the errors, these results are reported in Table 6.3. Both the parameters are in a well-defined minimum so the minimization phase seems to work properly.

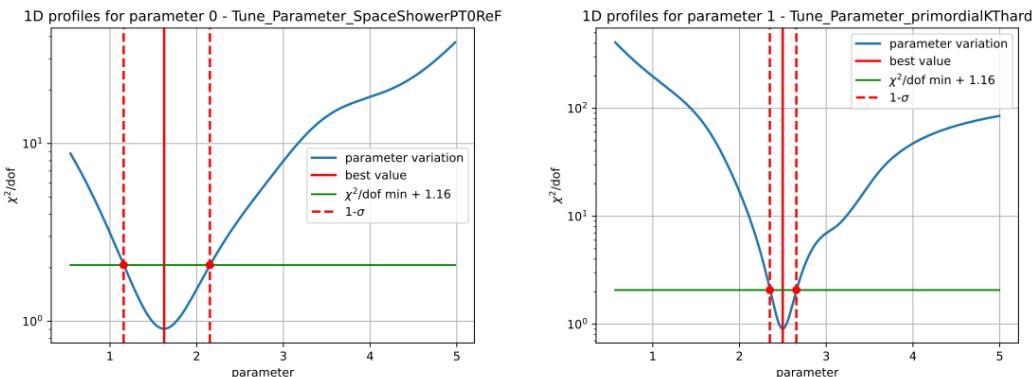


Figure 6.4: The output of the minimizer for the PerBin model in the primordial k_T tune. The left panel shows the result of the minimization for the `SpaceShower:pT0Ref` parameter while the right one `BeamRemnants:primordialKThard`. The blue line is the χ^2/DoF the best value is indicated by the solid red line while the errors are the ones indicated by the dashed red lines.

The Inverse model distribution of prediction for the best model is reported in Fig. 6.5 the best estimation for the parameter is indicated by the solid black line while the dashed black lines are the errors computed using the standard deviation. The hyperparameter space scanned to search for these best model is the same used above and reported in Table 5.3. At the end of the scanning, the best hyperparameters we found are the ones reported in the Table 6.2.

The PerBin Model gives us very stable results the value and the errors we got from the tune are reported in Table 6.3. These are also compared to the ones obtained from the Inverse model. The value obtained from the two MCNNTUNES models are

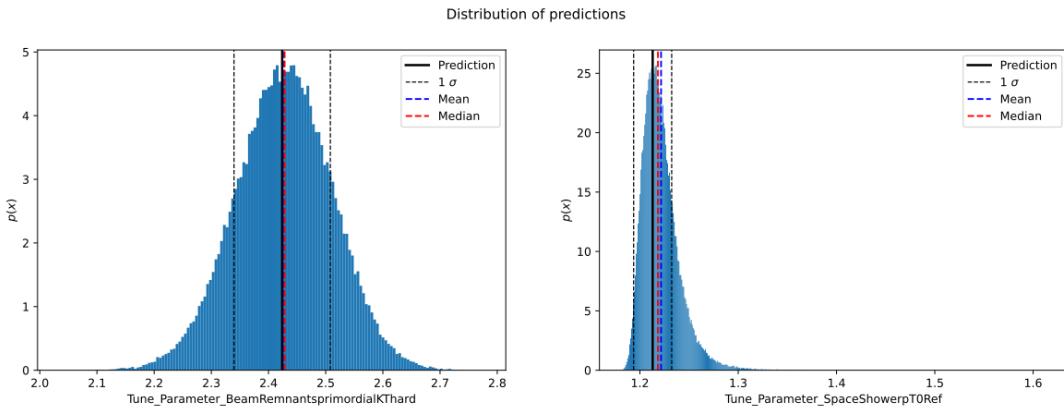


Figure 6.5: The prediction spread for the Inverse model in the primordial k_T tune. The left histogram refers to the `BeamRemnants:primordialKThard` parameter while the right one to the `SpaceShower:pT0Ref`. The predictions are indicated by the solid black vertical lines and the errors by the dashed black lines.

Hyperparameter	Value
Number of hidden layer	2
Units layers	[15, 17]
Activation function	sigmoid
Optimizer	rmsprop
Epochs	2000
Batch size	64

Table 6.2: Best hyperparameters model found for the primordial k_T tune with Inverse model.

compatible with each other. The default values for CP5 are also reported and they are the ones inherited by the Monash tune [39]. As we can see the value we get are quite different to the ones in CP5.

The overall results are displayed in Fig. 6.6. Our tunes describe better the low regions of the two spectra. These low regions are the ones we actually tune the higher p_T region is only simulated using the value obtained from the tune of the first 5 bins in each distribution of Fig. 6.6. To simulate all the spectrum we have to use FxFx margin scheme in order to avoid double-counting and get a correct result from the simulation.

Parameter Name	PerBin	Inverse	Default CP5
<code>BeamRemnants:primordialKThard</code>	$2.5^{+0.2}_{-0.2}$	2.42 ± 0.08	1.8
<code>SpaceShower:pT0Ref</code>	$1.6^{+0.5}_{-0.5}$	1.21 ± 0.02	2.0

Table 6.3: Results obtained from the PerBin and the Inverse model in the tuning of the low region of Z boson production spectra. They are compared to the default for CP5 (inherited from Monash tune.)

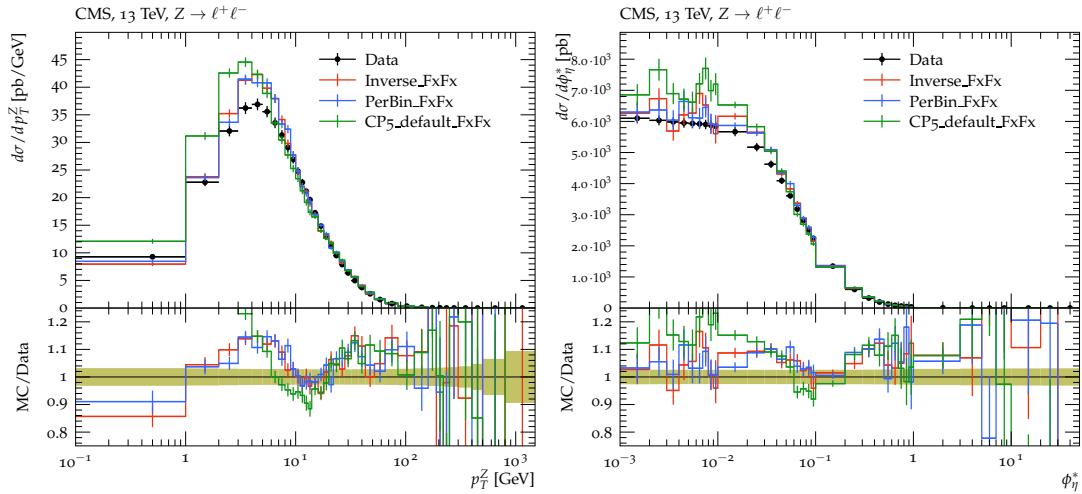


Figure 6.6: The results we get from the tune of the primordial k_T . The left figure shows the Z boson production cross-section as a function of the p_T^Z . The red line refers to the PerBin model tune, the blue line to the Inverse Model and the green line is the CP5 default tune. The black points are the experimental data. The colored vertical lines the statistical uncertainties. It is clear that our two tunes better describe the low region with respect to the original CP5.

6.3 Primordial k_T tune vs MPI

As discussed above primordial k_T is not the only source of non zero transverse momentum in LO Z boson production. This is the reason why we are tuning also the parameter that controls the amount of Initial State Radiation. As described above, the ISR can leads to a non zero initial transverse momentum a larger amount of ISR (low SpaceShower:pT0Ref) is related to more splitting occurring in the evolution of the incoming partons before these partons undergo the hard scattering in which the Z boson is produced. But this is not the only process that can lead to a non-zero initial transverse momentum, there are also the MPI that can contribute to this initial transverse momentum. It is not an easy task to understand the effect of MPI on the p_T of the produced Z boson.

Fig. 6.7 shows the effect of the MPI on the distributions we use for the tune. The blue line is obtained setting `PartonLevel:MPI=off` in PHOTIA8 configuration.

The main parameter that controls the number of parton interactions in a single hadron-hadron collision is the `MultipartonInteractions:pT0Ref` parameter. It is clear that the first 2 or 3 bins of the left distribution of Fig. 6.8 are sensitive to the variation of this parameter. A future thing to do is to investigate the effects of these MPI in more detail and perform a more general tune including them in this description.

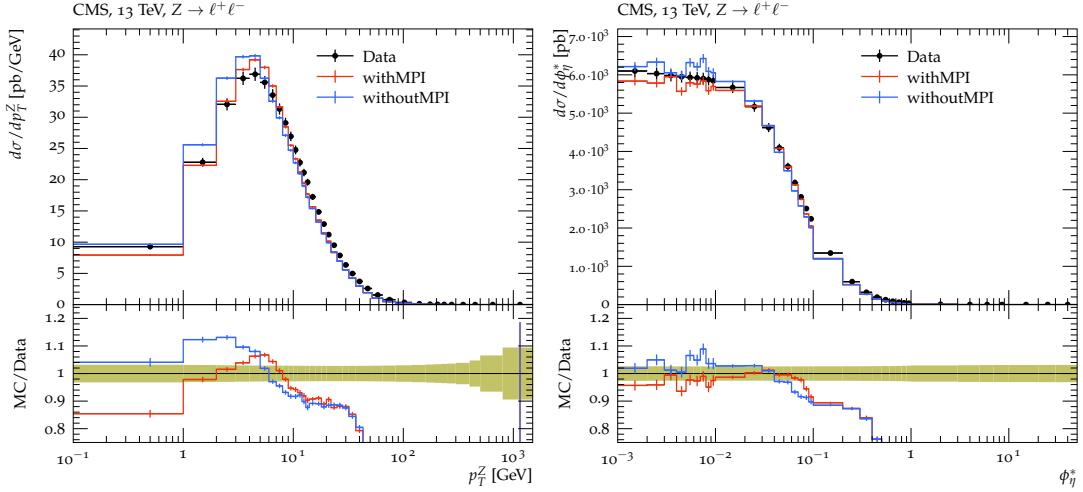


Figure 6.7: The image shows that MPI have an effect on the two distribution. The blue line is the result without the MPI (PartonLevel:MPI=off) and the red line with MPI.

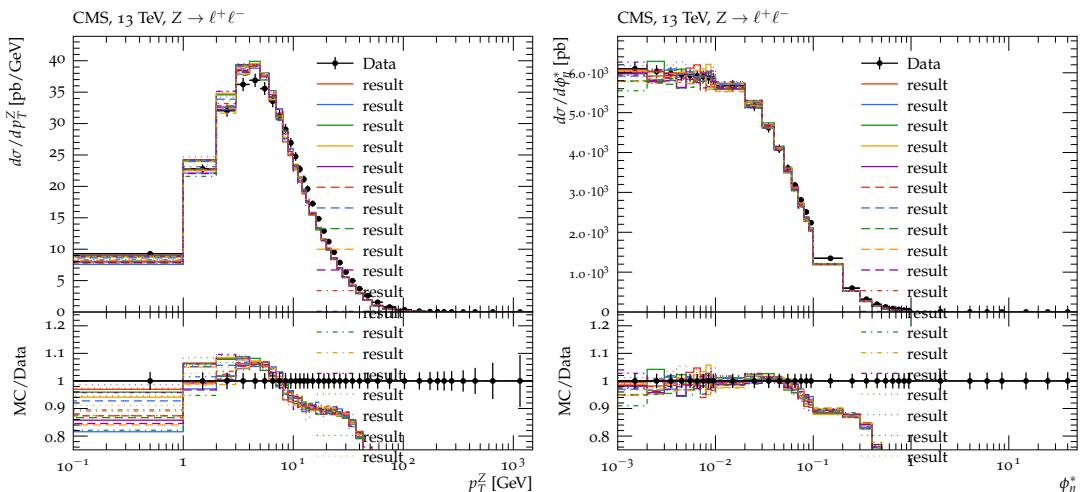


Figure 6.8: The image shows that MPI have an effect on the two distribution. The blue line is the result without the MPI and the red line with MPI.

Conclusions

In this work it has been validate MCNNTUNES as a valid tool for the tuning of the high energy physics Monte Carlo generator PYTHIA8. Firstly, in order to validate it, the underlying event was tuned trying to reproduce the results obtained with CP5. The work started with a test phase with the variation of only two parameters: `MultipartonInteractions:pT0Ref` and `MultipartonInteractions:ecmPow`. Both the MCNNTUNES models perform very well in this test phase.

After, this test the number of parameters was extended to the same parameters tuned in CP5. So, to the two parameters already tuned have been added the parameters `MultipartonInteractions:coreRadius`, `MultipartonInteractions:coreFraction` and `ColorReconnection:range`. In this phase we obtain very good results from the PerBin model. It seems to describe data also better then CP5 but not in all the distributions, in particular the distributions with larger experimental uncertainties. So, it was decided to include a re-weight for some bins. This leads to overall results more similar to the CP5 ones.

On the other hands the Inverse model gives very bad results in this case it is not able to produce some consistent results. This can be related to the complexity operation that is performed by this model: learn the function that describe the generator respond and try to invert it. This operation is very complex and this complexity increase a lot with the number of parameters.

Once the tool has been validated it is used to improve the description of the distributions of Z boson production events in the low transverse momentum region. To do that the effect on these distribution of the parameters `BeamRemnants:-primordialKThard` and `SpaceShower:pT0Ref` was investigated. In fact, the CP5 tune does not describe these distribution very well, the values for these two parameters were inherited by the older Monash Tune. The new tune presented here for these two parameters was performed with both the models offered by MCNNTUNES and the results obtained are compatible each other.

In the future works also the impact of the MPI has be investigated. In the last chapter it was showed that the effect of MPI is not negligible in the low transverse momentum region.

Overall MCNNTUNES can be considered a valid tool for future tunes and it can become a cross check to the already existing and consolidate tool PROFESSOR. Moreover, if some more attention is given to the Inverse model it can become a new powerful approach to the tuning procedure.

Appendix A

Test with two free parameters for the Underlying Events

Here are reported all the distributions obtained by the test on two parameters:

- `MultipartonInteractions:pT0Ref`
- `MultipartonInteractions:ecmPow`

We compare the results we get from PerBin Model and from Inverse Model with the CP5 ones.

The value we get are also reported in the Table A.1. The PerBin Model errors are not reported because there was not a proper errors estimation correctly implemented yet for this model.

Parameter	PerBin	Inverse	CP5 (down & up)
MPI:pT0Ref	1.46064	1.43 ± 0.14	$1.41 - 1.46$
MPI:ecmPow	0.04771	0.0298 ± 0.0095	0.03

Table A.1: Result PerBin model and Inverse Model in two parameter variation test. The value are compared to the upper and lower limit for CP5.

APPENDIX A. TEST WITH TWO FREE PARAMETERS FOR THE
UNDERLYING EVENTS

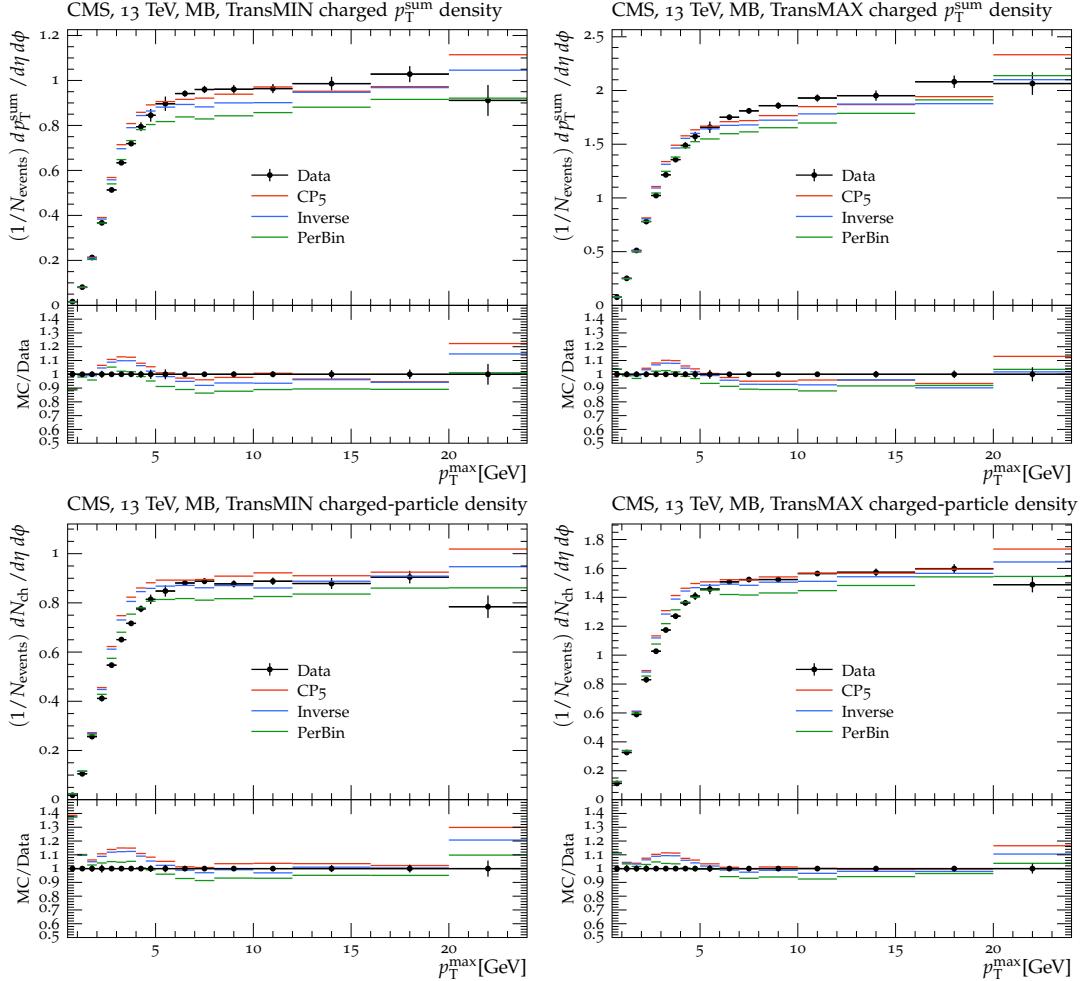


Figure A.1: In this figure the data from the $\sqrt{s} = 13$ TeV CMS analysis [25] that show the transMAX charged particle density (upper left) and the charged p_T -sum density (upper right); the transMIN charged particle density (lower left) and the charged p_T -sum density (lower right) as a function of the transverse momentum of the leading charged particle. The CP5 tune is compared to our test tune using the PerBin Model and Inverse Model.

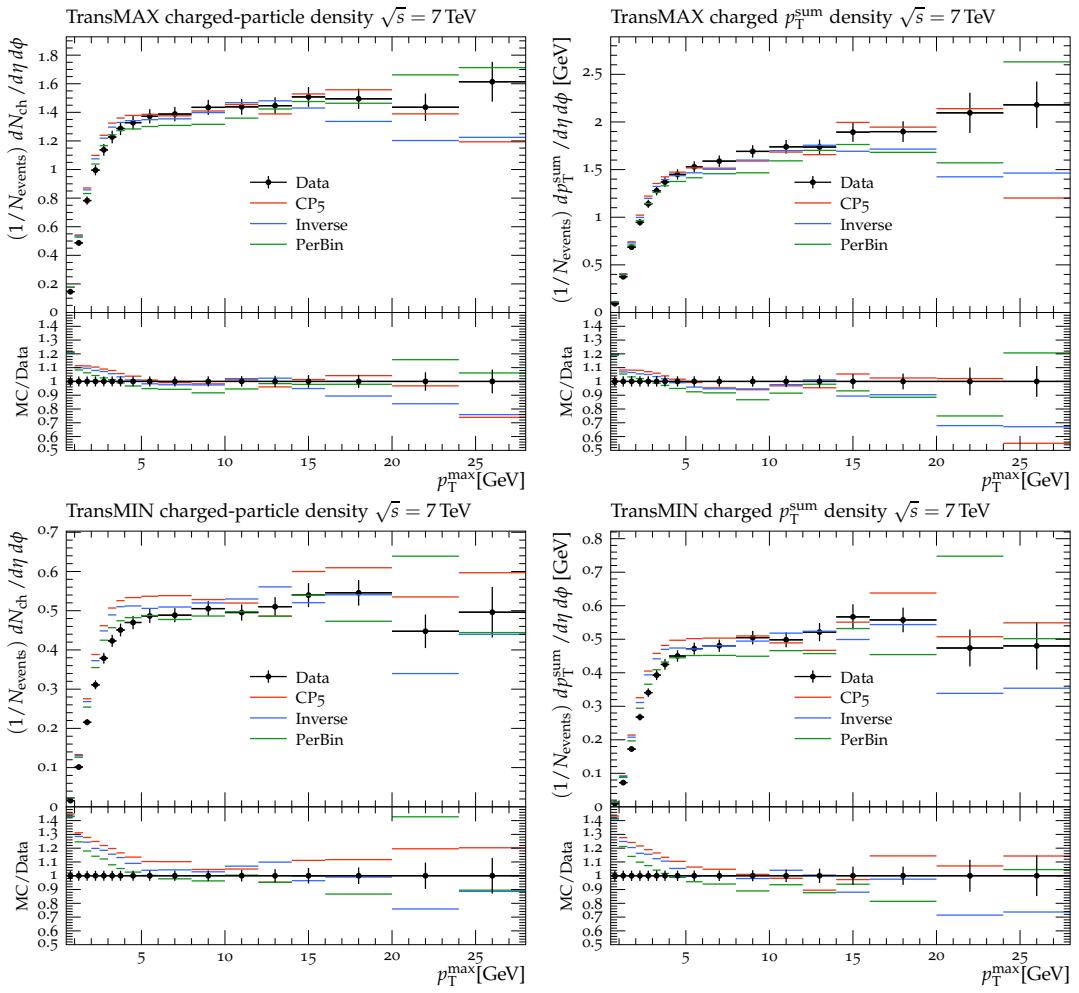


Figure A.2: Here the data at $\sqrt{s} = 7$ TeV from the CMS analysis [26] for the transMAX charged particle density (upper left) and the charged p_T -sum (upper right); the transMIN charged particle density (lower left) and the charged $p-T$ -sum are displayed as a function of the transverse momentum of the leading object.

APPENDIX A. TEST WITH TWO FREE PARAMETERS FOR THE
UNDERLYING EVENTS

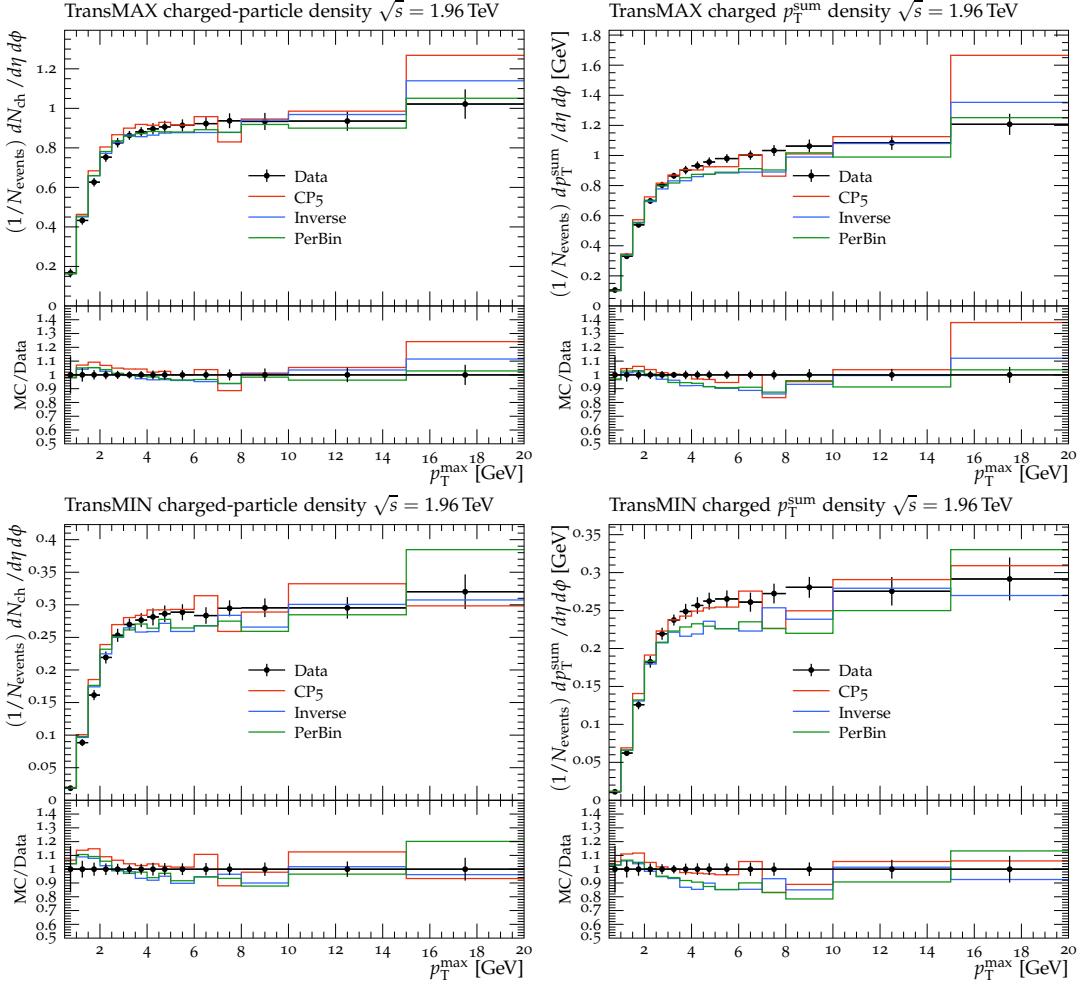


Figure A.3: The transMAX charged particle density (upper left) and the charged p_T -sum (upper left); the transMIN charged particle density (lower left) and the charged $p-T$ -sum from the CDF analysis at $\sqrt{s} = 1.96$ TeV in proton-antiproton collisions [27].

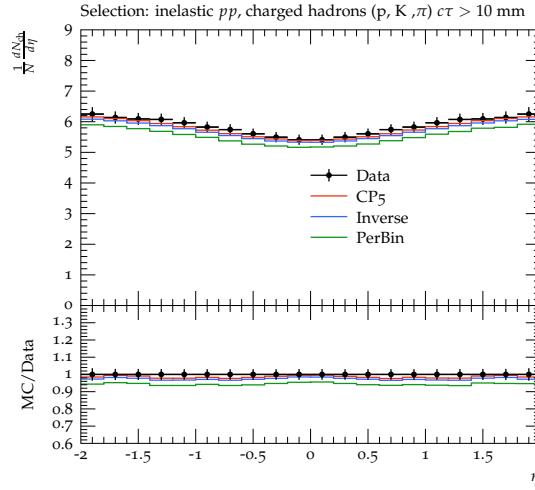


Figure A.4: In this figure is shown the last distribution we use for the tune from the CMS analysis at $\sqrt{s} = 13$ TeV [32]. The pseudorapidity distribution ($|\eta| < 2$) for the charged hadron density in an inelastic proton-proton scattering selection.

Bibliography

- [1] Steven Weinberg. A model of leptons, Nov 1967. URL <https://link.aps.org/doi/10.1103/PhysRevLett.19.1264>.
- [2] P.A. Zyla et al. Review of Particle Physics. *PTEP*, 2020(8):083C01, 2020. doi: 10.1093/ptep/ptaa104. and 2021 update.
- [3] H. Abramowicz et al. Combination of measurements of inclusive deep inelastic $e^\pm p$ scattering cross sections and QCD analysis of HERA data. *Eur. Phys. J. C*, 75(12):580, 2015. doi: 10.1140/epjc/s10052-015-3710-4.
- [4] R. P. Feynman. The behavior of hadron collisions at extreme energies. *Conf. Proc. C*, 690905:237–258, 1969.
- [5] Torbjörn Sjöstrand, Stefan Ask, Jesper R. Christiansen, Richard Corke, Nishita Desai, Philip Ilten, Stephen Mrenna, Stefan Prestel, Christine O. Rasmussen, and Peter Z. Skands. An introduction to pythia 8.2, Jun 2015. ISSN 0010-4655. URL <http://dx.doi.org/10.1016/j.cpc.2015.01.024>.
- [6] Albert M Sirunyan et al. Extraction and validation of a new set of CMS PYTHIA8 tunes from underlying-event measurements. *Eur. Phys. J. C*, 80(1):4, 2020. doi: 10.1140/epjc/s10052-019-7499-4.
- [7] Marco Lazzarin, Simone Alioli, and Stefano Carrazza. MCNNTUNES: Tuning Shower Monte Carlo generators with machine learning. *Comput. Phys. Commun.*, 263:107908, 2021. doi: 10.1016/j.cpc.2021.107908.
- [8] J. D. Bjorken. Asymptotic Sum Rules at Infinite Momentum. *Phys. Rev.*, 179: 1547–1553, 1969. doi: 10.1103/PhysRev.179.1547.
- [9] Sidney D Drell and Tung-Mow Yan. Partons and their applications at high energies, 1971. ISSN 0003-4916. URL <https://www.sciencedirect.com/science/article/pii/0003491671900716>.
- [10] L N Lipatov. The parton model and perturbation theory, 1975. URL <http://cds.cern.ch/record/400357>.
- [11] Vladimir Naumovich Gribov and L N Lipatov. Deep inelastic ep scattering in perturbation theory, 1972. URL <https://cds.cern.ch/record/427157>.
- [12] G. Altarelli and G. Parisi. Asymptotic freedom in parton language, 1977. ISSN 0550-3213. URL <https://www.sciencedirect.com/science/article/pii/0550321377903844>.

BIBLIOGRAPHY

- [13] Yuri L. Dokshitzer. Calculation of the Structure Functions for Deep Inelastic Scattering and e+ e- Annihilation by Perturbation Theory in Quantum Chromodynamics., 1977.
- [14] Richard D. Ball, Valerio Bertone, Stefano Carrazza, Luigi Del Debbio, Stefano Forte, Patrick Groth-Merrild, Alberto Guffanti, Nathan P. Hartland, Zahari Kassabov, José I. Latorre, Emanuele R. Nocera, Juan Rojo, Luca Rottoli, Emma Slade, and Maria Ubiali. Parton distributions from high-precision collider data. *The European Physical Journal C*, 77(10), Oct 2017. ISSN 1434-6052. doi: 10.1140/epjc/s10052-017-5199-5. URL <http://dx.doi.org/10.1140/epjc/s10052-017-5199-5>.
- [15] F. Bloch and A. Nordsieck. Note on the radiation field of the electron, Jul 1937. URL <https://link.aps.org/doi/10.1103/PhysRev.52.54>.
- [16] Toichiro Kinoshita. Mass singularities of feynman amplitudes, 1962. URL <https://doi.org/10.1063/1.1724268>.
- [17] T. D. Lee and M. Nauenberg. Degenerate systems and mass singularities, Mar 1964. URL <https://link.aps.org/doi/10.1103/PhysRev.133.B1549>.
- [18] J M Campbell, J W Huston, and W J Stirling. Hard interactions of quarks and gluons: a primer for lhc physics, Dec 2006. ISSN 1361-6633. URL <http://dx.doi.org/10.1088/0034-4885/70/1/R02>.
- [19] Stefano Frixione and Bryan R Webber. Matching nlo qcd computations and parton shower simulations. *Journal of High Energy Physics*, 2002(06):029–029, Jun 2002. ISSN 1029-8479. doi: 10.1088/1126-6708/2002/06/029. URL <http://dx.doi.org/10.1088/1126-6708/2002/06/029>.
- [20] Stefano Frixione, Paolo Nason, and Bryan R Webber. Matching nlo qcd and parton showers in heavy flavour production. *Journal of High Energy Physics*, 2003(08):007–007, Aug 2003. ISSN 1029-8479. doi: 10.1088/1126-6708/2003/08/007. URL <http://dx.doi.org/10.1088/1126-6708/2003/08/007>.
- [21] Stefano Frixione and Bryan R. Webber. The mc@nlo 3.1 event generator, 2005.
- [22] Rikkert Frederix and Stefano Frixione. Merging meets matching in MC@NLO. *JHEP*, 12:061, 2012. doi: 10.1007/JHEP12(2012)061.
- [23] FRANK SIEGERT. Monte-carlo event generation for the lhc, 2010. URL <http://etheses.dur.ac.uk/484/>.
- [24] Tai Sakuma and Thomas McCauley. Detector and Event Visualization with SketchUp at the CMS Experiment. *J. Phys. Conf. Ser.*, 513:022032, 2014. doi: 10.1088/1742-6596/513/2/022032.
- [25] Underlying Event Measurements with Leading Particles and Jets in pp collisions at $\sqrt{s} = 13$ TeV. Technical report, CERN, Geneva, 2015. URL <https://cds.cern.ch/record/2104473>.

- [26] Measurement of the Underlying Event Activity at the LHC at 7 TeV and Comparison with 0.9 TeV. Technical report, CERN, Geneva, 2012. URL <http://cds.cern.ch/record/1478982>.
- [27] Timo Antero Aaltonen et al. Study of the energy dependence of the underlying event in proton-antiproton collisions. *Phys. Rev. D*, 92(9):092009, 2015. doi: 10.1103/PhysRevD.92.092009.
- [28] Vardan Khachatryan et al. CMS Tracking Performance Results from Early LHC Operation. *Eur. Phys. J. C*, 70:1165–1192, 2010. doi: 10.1140/epjc/s10052-010-1491-3.
- [29] Gavin P. Salam and Gregory Soyez. A Practical Seedless Infrared-Safe Cone jet algorithm. *JHEP*, 05:086, 2007. doi: 10.1088/1126-6708/2007/05/086.
- [30] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. The anti- k_t jet clustering algorithm. *JHEP*, 04:063, 2008. doi: 10.1088/1126-6708/2008/04/063.
- [31] Albert M. Sirunyan et al. Measurement of charged particle spectra in minimum-bias events from proton–proton collisions at $\sqrt{s} = 13$ TeV. *Eur. Phys. J. C*, 78(9):697, 2018. doi: 10.1140/epjc/s10052-018-6144-y.
- [32] Vardan Khachatryan et al. Pseudorapidity distribution of charged hadrons in proton-proton collisions at $\sqrt{s} = 13$ TeV. *Phys. Lett. B*, 751:143–163, 2015. doi: 10.1016/j.physletb.2015.10.004.
- [33] Serguei Chatrchyan et al. Measurement of the underlying event in the Drell-Yan process in proton-proton collisions at $\sqrt{s} = 7$ TeV. *Eur. Phys. J. C*, 72:2080, 2012. doi: 10.1140/epjc/s10052-012-2080-4.
- [34] A. M. Sirunyan et al. Measurement of the underlying event activity in inclusive Z boson production in proton-proton collisions at $\sqrt{s} = 13$ TeV. *JHEP*, 07:032, 2018. doi: 10.1007/JHEP07(2018)032.
- [35] Albert M. Sirunyan et al. Study of the underlying event in top quark pair production in pp collisions at 13 TeV. *Eur. Phys. J. C*, 79(2):123, 2019. doi: 10.1140/epjc/s10052-019-6620-z.
- [36] Florian Bechtel. *The underlying event in proton-proton collisions*. PhD thesis, Hamburg U., 2009.
- [37] A. Banfi, S. Redford, M. Vesterinen, P. Waller, and T. R. Wyatt. Optimisation of variables for studying dilepton transverse momentum distributions at hadron colliders. *Eur. Phys. J. C*, 71:1600, 2011. doi: 10.1140/epjc/s10052-011-1600-y.
- [38] Albert M Sirunyan et al. Measurements of differential Z boson production cross sections in proton-proton collisions at $\sqrt{s} = 13$ TeV. *JHEP*, 12:061, 2019. doi: 10.1007/JHEP12(2019)061.
- [39] Peter Skands, Stefano Carrazza, and Juan Rojo. Tuning PYTHIA 8.1: the Monash 2013 Tune. *Eur. Phys. J. C*, 74(8):3024, 2014. doi: 10.1140/epjc/s10052-014-3024-y.

BIBLIOGRAPHY

- [40] Andy Buckley, Hendrik Hoeth, Heiko Lacker, Holger Schulz, and Jan Eike von Seggern. Systematic event generator tuning for the LHC. *Eur. Phys. J. C*, 65:331–357, 2010. doi: 10.1140/epjc/s10052-009-1196-7.
- [41] Stefano Carrazza and Marco Lazzarin. N3pdf/mcnntunes: mcnntunes 0.1.0, October 2020. URL <https://doi.org/10.5281/zenodo.4071125>.
- [42] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- [43] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. 1958. URL <https://doi.org/10.1037/h0042519>.
- [44] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991. ISSN 0893-6080. doi: [https://doi.org/10.1016/0893-6080\(91\)90009-T](https://doi.org/10.1016/0893-6080(91)90009-T). URL <https://www.sciencedirect.com/science/article/pii/089360809190009T>.
- [45] Moshe Leshno, Vladimir Ya. Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, 1993. ISSN 0893-6080. doi: [https://doi.org/10.1016/S0893-6080\(05\)80131-5](https://doi.org/10.1016/S0893-6080(05)80131-5). URL <https://www.sciencedirect.com/science/article/pii/S0893608005801315>.
- [46] YODA. Yoda. URL <https://yoda.hepforge.org/>.
- [47] Nikolaus Hansen. The CMA evolution strategy: A tutorial. *CoRR*, abs/1604.00772, 2016. URL <http://arxiv.org/abs/1604.00772>.
- [48] G. Cowan. *Statistical data analysis*. Oxford University Press, USA, 1998.
- [49] Richard D. Ball et al. Parton distributions from high-precision collider data. *Eur. Phys. J. C*, 77(10):663, 2017. doi: 10.1140/epjc/s10052-017-5199-5.
- [50] Albert M Sirunyan et al. Measurement of differential cross sections for Z boson production in association with jets in proton-proton collisions at $\sqrt{s} = 13$ TeV. *Eur. Phys. J. C*, 78(11):965, 2018. doi: 10.1140/epjc/s10052-018-6373-0.