Benemérita Universidad Autónoma de Puebla

Facultad de Ciencias Físico Matemáticas

Extracción del parámetro de impacto de colisiones protón+protón con el experimento ALICE del LHC

Tesis presentada al

Posgrado en Física Aplicada

como requisito parcial para la obtención del grado de

Maestría en Ciencias (Física Aplicada)

por

Andrea Aguirre Polo

Asesorada por

Dr. Mario Rodríguez Cahuantzi
Dr. Arturo Fernández Téllez

Puebla Pue.

Noviembre de 2024

COMMITTEE MEMBERS

Dr. Enrique Varela Carlos
President

Dr. Mario Iván Martıínez Hernández
Secretary

Dr. Guillermo Tejeda Muñoz
Vocal

Dr. Cristian Heber Zepeda Fernández
Substitute

Dr. Mario Rodríguez Cahuantzi
Supervisor

Dr. Arturo Fernández Téllez
Supervisor

EXTRACTING THE IMPACT PARAMETER FROM PROTON+PROTON COLLISIONS WITH THE ALICE EXPERIMENT AT THE LHC

Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science (Applied Physics)

by

Andrea Aguirre Polo

Under the supervision of

Dr. Mario Rodríguez Cahuantzi & Dr. Arturo Fernández Téllez

Facultad de Ciencias Físico Matemáticas
Benemérita Universidad Autónoma de Puebla

Acknowledgments

First of all I want to thank my advisors Dr. Mario Rodriguez Cahuantzi and Dr. Arturo Fernandez Tellez for giving me full support in my decisions and for sharing their experience and knowledge in this field. I am very thankful to them for making me grow academically and believing in me. This was a unique opportunity for me and I'll forever be grateful for the expert guidance I was given by my advisors.

I would like to express my gratitude to CONACYT not only for their financial support provided during my whole master's degree studies, but also to their CF-2019/2042 and A1-S-13525 grants which allowed me to do my internships at CERN. I would also like to thank the national supercomputing laboratory.

I'm also grateful for the great teachers I had during my master's degree, not only my advisors but I learned so much from my theoretical courses with Dr. Héctor Novales Sánchez whom tough me since bachelors, (Quantum Mechanics, Quantum Mechanics II and Quantum Field Theory in my master), his courses were also fun and well given.

I'm also thankful with my thesis committee, for giving me their valuable time with a careful review and insights in my work.

Some very important people that helped me in both academic and emotional support were my friends:

My CERN friends which advised me, guided me, and took care of me while I was away from home: Luis Perez, Iraq Rabadan, Arely Cortes, Solangel Rojas, Simone Ragoni, Dalia Ramirez, Marco Ayala, Fernando Neri, and Yael Vásquez who not only was a good roomie but also my best instrumentation teacher.

My FCFM friends which were always there for my questions and gave me good times, fortunately they are too many but I can't name them all, so my especial thanks go to Jesus Ricardo Alvarado and Pablo Fierro for not also giving me the first overview of what people in this field should know but also for their emotional support during hard times. Also I want to thank my ALICE-BUAP friends, which are basically all my advisors' students.
Another FCFM friend, but also a life friend since kids is Ricardo Alvarez, who always makes me laugh even in my hardest time.

Finally I want to deeply thank my family, first my mother and father for sacrificing everything they had for me, even tho they were too young when I was born they always made the best decisions for me, every time I got lost in the wrong way they did everything they could to keep me in the right path and never gave up on me, and this is why I am

where I am today. Second, my little sister, she came to my life to give me hope and a reason to always try my best for someone you love.

I would also like to thank my one and only best friend, Viviana Avila, with whom I've gone trough the most difficult times in my life, and nothings has being able to set us apart.

My deepest thanks to my four grandparents, which not only always took care of me in Mexico but always took care of me during my time away. Also the biggest thanks to my aunts, Mago and Rayo that are always taking care of me and making sure I have everything I need, thanks to all of them for giving me unconditional love.

What I am deeply thankful about is that I have found someone that makes me happy, my boyfriend, who makes me grow and even in difficult times he always manages to make it work. I was so lucky because not only he gives me emotional support but also takes care of me during my time away at CERN, supporting all my decisions and being the best person for me.

Thanks!

For Majo.

A sister is both your mirror – and your opposite.

Abstract

In high-energy proton-proton collisions, understanding the impact parameter is crucial for unraveling the underlying dynamics of particle interactions. This work explores the extraction of the impact parameter using machine learning techniques in the context of the ALICE experiment at the Large Hadron Collider (LHC). We apply ML models (DNN and decision Trees) to predict the multiplicity of charged particles in relation to the impact parameter, leveraging simulated data generated by the PYTHIA 8.3 event generator. Our findings demonstrate the efficacy of machine learning algorithms in providing accurate estimates of the impact parameter, offering a new avenue for exploring particle collisions and expanding our understanding of quantum chromodynamics (QCD) in high-energy physics.

Contents

List of Figures

List of Tables

Part I

Overview

Chapter 1: Introduction

High energy collisions, like proton-proton collisions at almost the speed of light, offer us deep insight into our universe's fundamental forces and particles and help us understand the validity of the standard model, comparing them to heavy ion collisions, such as PbPb collisions where we have collectivity effects, give us insight into the extreme conditions that may have existed in the early universe. ALICE gives us a unique environment to observe the de-confinement phenomenon and the formation of this medium, the quark-gluon plasma, a state of matter thought to have existed a few moments after the Big Bang. Information about QGP production help us explore the interaction governed by quantum chromodynamics and the dynamics of quarks and gluons by describing the strong force, which is the force that binds quarks together within hadrons.

In recent years, machine learning (ML) has emerged as a powerful tool in numerous scientific fields, particularly in high-energy physics (HEP). At the LHC, machine learning models are increasingly used to analyze vast datasets and uncover patterns that would be otherwise difficult to discern through traditional analytical techniques. One such application is the estimation of impact parameters in proton-proton collisions, which plays a key role in determining the spatial geometry of the interaction region and provides insights into the collective behavior of produced particles.

The ALICE experiment at the LHC is dedicated to studying heavy-ion collisions and the quark-gluon plasma, but it also offers valuable data from proton-proton collisions. Charged particle multiplicity, a direct observable from these collisions, is sensitive to the initial state geometry and thus to the impact parameter. In this work, we aim to extract the impact parameter using machine learning algorithms applied to ALICE pp data.

Through a combination of Monte Carlo simulations using PYTHIA 8.3 and machine learning models, we predict charged particle multiplicities for different impact parameter ranges. This approach represents an innovative method for probing the initial conditions of collisions, potentially leading to deeper insights into the structure and behavior of quantum chromodynamics (QCD) matter at high energies.

- Chapter 2: Theory In Chapter 2, we introduce the theoretical foundations with the Standard Model, which presents the types of particles and their interactions; then we expand to Quantum Chromodynamics (QCD), which we know is the theory of the strong force, exploring the concept of deconfinement and with this the formation of QGP. We discuss key observables such as particle multiplicity and strangeness enhancement. For our impact parameter determination study, we also discuss centrality determination with Glauber for heavy ions and pp, providing a theoretical framework for understanding the phenomena studied in the subsequent chapters.

- Chapter 3: The ALICE Experiment at the LHC

  Chapter 3 introduces the ALICE detector, its subdetectors, and the LHC setup in general. We discuss ALICE's specific role as the only one designed to study heavy-ion collisions. This chapter covers the detector's capabilities, such as centrality determination, track and vertex reconstruction, and particle identification techniques (PID). We also introduce the ALICE Online-Offline Framework ($O2$) used for RUN 3 data analysis.

- Chapter 4: Frameworks

  We employed important frameworks, CERN's framework ROOT usually is used in particle physics for data analysis when it comes to large data sets, and then python and its libraries for more advanced techniques, like to define and train our models. To go from one to another, we use the Uproot interface. These are critical for managing our data and interpreting the results.

- Chapter 5: ALICE Data and Simulations

  Chapter 5 focuses on the experimental data collected by the ALICE detector and the role of Monte Carlo simulations in the analysis. It introduces Monte Carlo event generators, specifically PYTHIA 8.3 Tune Monas, EPOS-LHC, and EPOS4, which model particle collisions and we use to compare theoretical predictions with experimental results.

- Chapter 6: Machine Learning

  The use of machine learning techniques in particle physics data processing is introduced in Chapter 6. In general, it discusses the expanding importance of AI to further experimental physics research and covers a variety of machine learning techniques, such as neural networks, decision trees, and linear regression.

- Chapter 7: Impact Parameter Study

  Chapter 7 we start talking about impact parameter in heavy-ion and proton-proton collisions, also its influence on observables such as average transverse momentum and in the phenomenon of enhanced production of multi-strange hadrons. The chapter aims to provide deeper insights into collision conditions and the properties of the QGP.

- Chapter 8: Determination of the Impact Parameter in pp Collisions using ML

  In chapter 8 we begin to talk about our specific goal, how to determine the impact parameter profile in pp collisions given the multiplicity, here we apply machine learning techniques to determine the impact parameter in proton-proton (pp) collisions. It outlines the processes of extracting multiplicity classes, preparing data, and training ML models.

- Chapter 9: ML-Background Subtraction

  With an emphasis on the Muon-ID prototype for the ALICE 3 upgrade, Chapter 9 discusses the problem of background noise in instrumentation studies, particularly when it comes to characterizing a detector. It investigates the application of machine learning methods, such as neural networks and random forest classifiers, for background subtraction.

- Chapter 10: Conclusions and Future Work

  The study's main conclusions are outlined in Chapter 10, along with their implications for further research.

Chapter 2: Theory

2.1 The Standard Model of Particle Physics

"The Standard Model explains how the basic building blocks of matter interact, governed by four fundamental forces." — CERN

The model categorizes elementary particles into two main groups: fermions (matter constituents), all matter in the world is made of these elementary particles, and bosons (force carriers), fundamental forces that result from the exchange of these force-carrier particles.

2.1.1 Elementary Fermions

Elementary fermions are categorized into quarks and leptons. Fermions obey Fermi–Dirac statistics and the Pauli exclusion principle, which states that no identical fermions can simultaneously occupy the same quantum state.

- **Quarks**: Quarks are the constituents of hadrons, such as protons and neutrons. There are six types of quarks, known as flavors: up, down, charm, strange, top, and bottom. Quarks combine in various ways to form composite particles known as hadrons:
  - **Mesons**: Mesons are hadrons composed of one quark and one antiquark. Examples include the pion (π) and kaon (K) mesons. Mesons play a crucial role in mediating the strong interaction between baryons.
  - **Baryons**: Baryons are hadrons made up of three quarks. The most well-known baryons are the proton (uud) and the neutron (udd). Baryons are the building blocks of atomic nuclei.
- **Leptons**: Leptons include particles such as electrons, muons, and neutrinos. Unlike quarks, leptons do not participate in strong interactions. They are classified into three generations, each containing a charged lepton and its corresponding neutrino:
  - **First Generation**: Electron (e^-) and electron neutrino (ν_e).
  - **Second Generation**: Muon (μ^-) and muon neutrino (ν_μ).
  - **Third Generation**: Tau (τ^-) and tau neutrino (ν_τ).

The following table summarizes the elementary fermions organized by generation:

	Quarks		Leptons	
1st Generation	Up	Down	e^-	ν_e
2nd Generation	Charm	Strange	μ^-	ν_μ
3rd Generation	Top	Bottom	τ^-	ν_τ
Charge	$+\frac{2}{3}$	$-\frac{1}{3}$	-1	0
Spin 1/2				

Table 1: Elementary Fermions organized by generation

2.1.2 Elementary Bosons

Bosons are force carriers that obey Bose–Einstein statistics, unlike fermions, multiple bosons can occupy the same quantum state. They mediate the fundamental interactions between particles, they are categorized into gauge bosons and the Higgs boson.

- **Gauge Bosons**: These bosons mediate the fundamental forces:
  - **Photon (γ)**: Mediates the electromagnetic force.
  - **$W^{\pm}$ and Z^0 Bosons**: Mediate the weak force, which is responsible for processes like beta decay.
  - **Gluons (8 types)**: Mediate the strong force, which binds quarks together within hadrons.
- **Scalar Boson**: The Higgs boson, discovered in 2012, is responsible for giving mass to other particles through the Higgs mechanism.

The following table summarizes the elementary bosons:

Gauge Bosons	Scalar Bosons
γ (photon) $W^{\pm}$ and Z^0 bosons 8 types of gluons	Higgs Boson
Spin 1	**Spin 0**

Table 2: Elementary Bosons

2.1.3 Fundamental Interactions

The Standard Model describes three of the four known fundamental interactions:

- **Electromagnetic Force**: Mediated by the photon (γ), this force acts on all particles with electric charge. It governs interactions such as electromagnetic attraction and repulsion.
- **Weak Force**: Mediated by the W and Z bosons, the weak force is responsible for processes such as beta decay. It affects all fermions and is associated with changes in flavor.

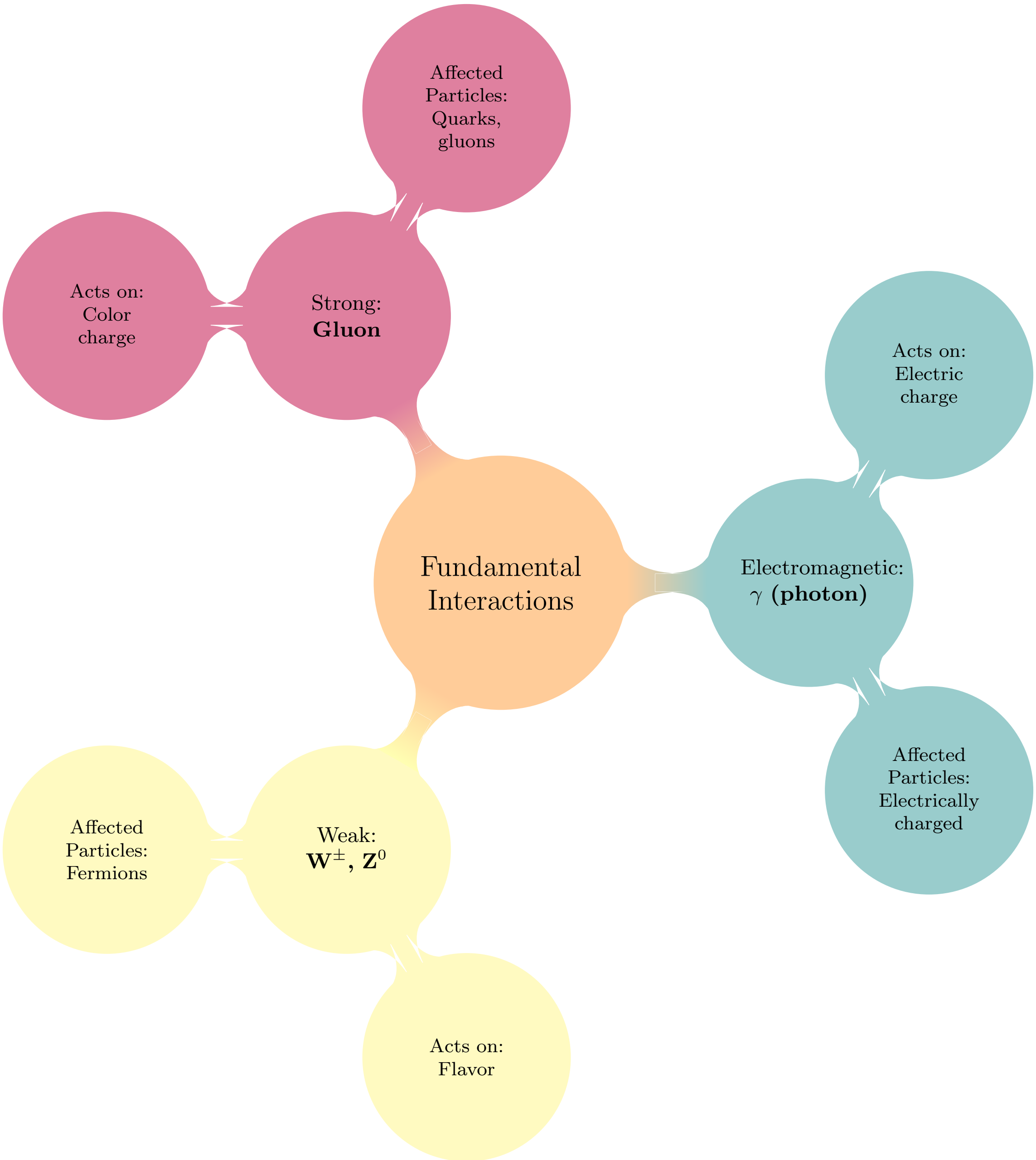


Figure 1: A mind map of the fundamental interactions in the Standard Model.

- **Strong Force**: Mediated by gluons, the strong force binds quarks together within hadrons. It acts on particles with color charge, such as quarks and gluons, and is the strongest of the fundamental forces.
  - **Partons**: In the context of high-energy physics, quarks and gluons are referred to as partons when they are probed inside hadrons in deep inelastic scattering processes. The parton model is essential for understanding the internal structure of protons and neutrons at high energies.

The Standard Model of Particle Physics represents one of the most successful theories in modern science, providing a unified description of the fundamental particles and their interactions (except for gravity). By categorizing particles into fermions and bosons, and explaining their behaviors through the gauge symmetries and the Higgs mechanism, it has fundamentally shaped our understanding of the universe at the smallest scales.

Despite its success, the Standard Model is not complete. It does not include gravity, nor does it fully explain phenomena such as dark matter and dark energy. Ongoing research in particle physics, including experiments at high-energy colliders like the LHC, aims to explore beyond the Standard Model. This includes searching for new particles, understanding the properties of known particles in greater detail, and testing theoretical extensions such as supersymmetry or string theory.

If readers seek a deeper understanding of the Standard Model, they can consult "Quantum Field Theory and the Standard Model" by Matthew D. Schwartz, one of the primary references used in this thesis [14]. In summary, while the Standard Model has proven to be a robust framework for understanding the fundamental aspects of particle physics, it is an evolving field with ongoing challenges and opportunities for new discoveries. Future research and advancements in experimental techniques will continue to test and expand our knowledge of the fundamental nature of matter and forces in the universe.

2.2 Feynman Diagrams

Feynman diagrams are a powerful visual tool used to represent the interactions between particles in quantum field theory (QFT). They provide an intuitive way to calculate and visualize processes involving elementary particles in the Standard Model of Particle Physics. Each line and vertex in a Feynman diagram corresponds to a mathematical expression that contributes to the overall amplitude of a process.

2.2.1 Feynman Diagram Components

A typical Feynman diagram consists of the following components:

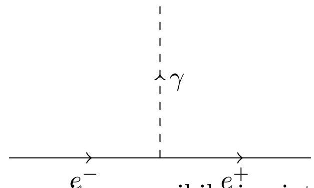
- **External Lines**: These represent incoming or outgoing particles. External lines are usually drawn as straight lines for fermions (such as electrons) and wavy or curly lines for bosons (such as photons or gluons).

- **Internal Lines (Propagators)**: These represent virtual particles that mediate the interaction between other particles. They are not directly observable and exist only for a fleeting moment during the interaction.

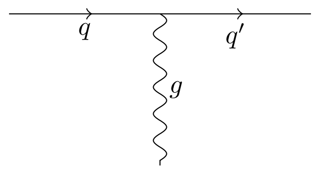
- **Vertices**: Points where lines meet represent the interaction points between particles. The number of lines meeting at a vertex corresponds to the interaction's degree, governed by the rules of the quantum field theory under consideration.

2.2.2 Examples of Feynman Diagrams in the Standard Model

Below are some examples of simple Feynman diagrams that illustrate different interactions within the Standard Model:



(a) Electron-positron annihilation into a photon



(b) Gluon exchange between quarks

Figure 2: Examples of Feynman diagrams for (a) electron-positron annihilation and (b) gluon exchange in QCD.

Electron-Positron Annihilation: In this process (Figure 2(a)), an electron (e^-) and a positron (e^+) annihilate to produce a photon (γ). In medical imaging, specifically Positron Emission Tomography (PET), this process is harnessed to detect gamma photons that help create images of internal body structures, aiding in the diagnosis of various conditions.

Gluon Exchange in QCD: In this process (Figure 2(b)), quarks exchange a gluon (g), the mediator of the strong force in Quantum Chromodynamics (QCD). This interaction is fundamental to the binding of quarks within protons, neutrons, and other hadrons, illustrating the core concept of quark confinement.

2.2.3 Calculating Amplitudes

Feynman diagrams are not just visual aids but also serve as a method for calculating the probability amplitudes of physical processes. Each diagram corresponds to a term in the perturbative expansion of the quantum field theory, and the sum of all relevant diagrams gives the total amplitude. The rules for translating a Feynman diagram into a mathematical expression are known as Feynman rules, which vary depending on the specific quantum field theory.

2.3 Quantum Chromodynamics

Quantum Chromodynamics (QCD) is the theory that describes the strong interaction. It focuses on the interactions between quarks and gluons, the fundamental constituents of hadrons. QCD is characterized by its use of color charge, analogous to electric charge in electromagnetism but with three types (red, green, and blue).

The fundamental aspect of QCD is its gauge symmetry, described by the group $\mathrm{SU}(3)_C$. The QCD Lagrangian is formulated to be invariant under this symmetry and is given by:

$$\mathcal{L}_{\mathrm{QCD}} = -\frac{1}{4}F^a_{\mu\nu}F^{a\mu\nu} + \sum_q \bar{\psi}_q(i\gamma^\mu D_\mu - m_q)\psi_q, \tag{1}$$

where $F^a_{\mu\nu}$ is the field strength tensor for the gluon fields, ψ_q represents the quark fields, m_q is the mass of the quark, γ^μ are the gamma matrices, and D_μ is the covariant derivative defined as:

$$D_\mu = \partial_\mu - ig_s A^a_\mu T^a. \tag{2}$$

Here, A^a_μ are the gluon fields, T^a are the generators of the SU(3) gauge group, and g_s is the strong coupling constant. The field strength tensor $F^a_{\mu\nu}$ is given by:

$$F^a_{\mu\nu} = \partial_\mu A^a_\nu - \partial_\nu A^a_\mu + g_s f^{abc} A^b_\mu A^c_\nu, \tag{3}$$

where f^{abc} are the structure constants of the SU(3) group. This Lagrangian describes the dynamics of the gluon fields and their interaction with the quark fields. The non-Abelian nature of the gauge group SU(3) results in self-interactions among the gluons, a key feature distinguishing QCD from electromagnetism.

In summary, QCD provides a comprehensive description of the strong force, encapsulating the complex interactions between quarks and gluons through its gauge theory framework and the associated Lagrangian. For further details on the theoretical and experimental aspects of QCD, readers can refer to "QCD and Collider Physics" by Ellis, Stirling, and Webber [15].

The states of the QCD matter under different conditions can be represented in a phase diagram [Fig 5].

The QCD phase diagram provides a depiction of the different phases of QCD matter as a

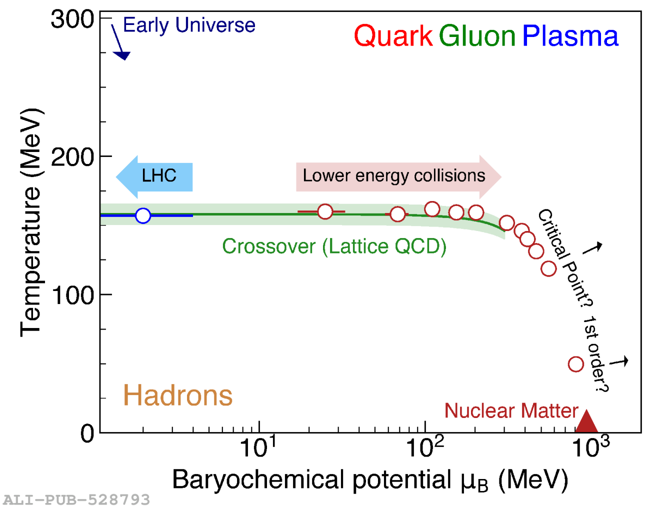


Figure 3: The QCD phase diagram. Figure taken from [1].

function of temperature and baryon chemical potential, crucial for understanding the behavior of QCD under extreme conditions, such as those found in heavy-ion collisions or in the early universe. The QCD phase diagram. Figure taken from At low temperatures and baryon densities, QCD matter is in a hadronic phase where quarks are confined within hadrons. As temperature and/or density increase, the system may undergo a phase transition to a quark-gluon plasma (QGP), where quarks and gluons are deconfined and can move freely.

The QCD phase diagram features the following key regions:

- Hadronic Phase: Low temperature and low baryon density, where quarks are confined in hadrons.
- Quark-Gluon Plasma (QGP) Phase: High temperature and/or high baryon density, where quarks and gluons are deconfined.
- Critical Point: The point where a first-order phase transition between the hadronic phase and QGP occurs. It represents a significant area of study as it may provide insights into the properties of strongly interacting matter.

2.3.1 Deconfinement: The Quark-Gluon Plasma

Normally, quarks and gluons are confined inside colorless hadrons such as protons and neutrons. In these particles, quarks are held together by the exchange of gluons, which are the carriers of the strong force. However, in high-energy environments, such as those created in heavy-ion collisions, the conditions can become extreme enough to overcome the strong force binding quarks inside hadrons. In such cases, quarks and gluons can become deconfined, forming a state of matter known as the quark-gluon plasma (QGP).

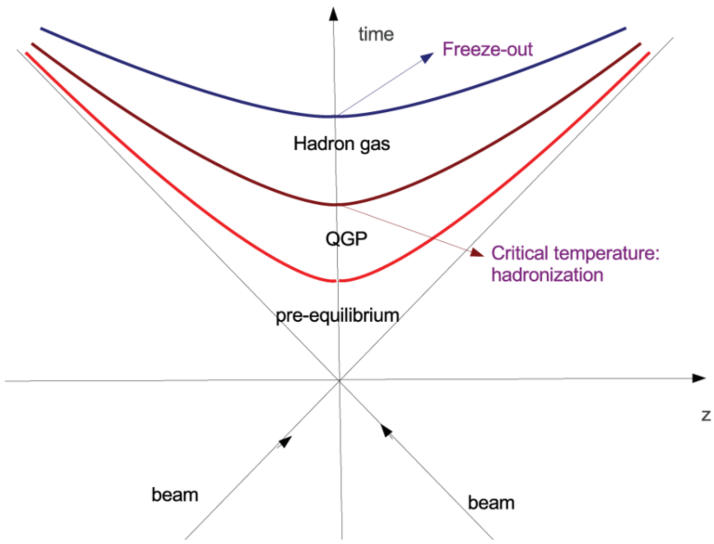


Figure 4: Sketch of a relativistic nuclear collision process at high energy. Figure taken from [2].

The quark-gluon plasma is a state of matter where quarks and gluons are no longer confined within hadrons but are instead free to move within a hot, dense medium. This state is believed to have existed shortly after the Big Bang, during the first microseconds of the universe's evolution.

At extremely high temperatures, the quark-gluon plasma can approximate the behavior of an ideal gas, where interactions between the particles become relatively weak. This is a key feature in the study of QGP, as it allows physicists to use concepts from statistical mechanics to describe the plasma's properties. The transition from the hadron state to the quark-gluon plasma and back is known as the deconfinement transition, and its study is crucial for understanding the fundamental aspects of quantum chromodynamics (QCD) and the early universe.

2.4 Heavy-Ion Collisions

Heavy-ion collisions involve the interaction of large nuclei at high energies, creating conditions that allow for the study of phenomena like QGP. These collisions are used to investigate the properties of nuclear matter under extreme conditions and to explore the phase transition from normal nuclear matter to QGP.

In contrast to heavy-ion collisions, proton-proton (pp) collisions involve interactions between individual protons. Traditionally, pp collisions were not thought to produce the same extreme conditions as heavy-ion collisions, and therefore were not expected to show phenomena like QGP. However, recent studies have suggested that pp collisions might also exhibit collectivity effects,

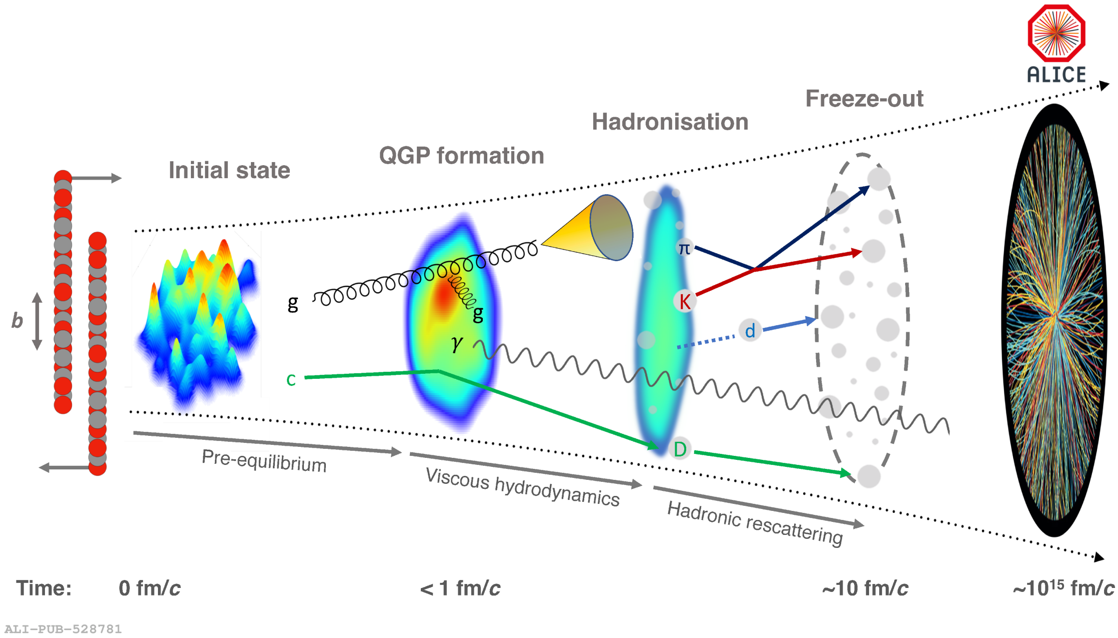


Figure 5: The evolution of a heavy-ion collision at LHC energie. Figure taken from [1].

which were previously considered characteristic of heavy-ion collisions. This emerging evidence suggests that even in smaller systems like pp collisions, collective behavior can emerge, providing new insights into the fundamental dynamics of particle interactions.

2.5 Observables in HEP Collisions

High-energy physics (HEP) collisions produce a variety of observables that provide information about the underlying processes and the properties of the matter produced.

2.5.1 Particle Multiplicity

Particle multiplicity refers to the number of particles produced in a collision. It is a key observable that provides insights into the collision dynamics and the nature of the produced matter. High multiplicity can indicate a high energy density and potentially the formation of a quark-gluon plasma.

2.5.2 Strangeness Enhancement

Strangeness enhancement is a phenomenon observed in heavy-ion collisions where the production of strange quarks and hadrons is significantly higher compared to proton-proton collisions. This enhancement is a signature of QGP formation and reflects the increased energy density and temperature in the collision environment.

2.5.3 High-p_T Hadrons

High transverse momentum (high-p_T) hadrons are particles with large momentum perpendicular to the collision axis. The study of high-p_T hadrons provides information about the parton distribution functions and the energy loss of partons in the medium, helping to probe the properties of the quark-gluon plasma and the dynamics of the collision.

2.6 Glauber

2.6.1 Pb-Pb

The Glauber model is a theoretical framework widely used to describe the geometry and dynamics of heavy-ion collisions. By leveraging nuclear geometry, it allows for the modeling of the spatial overlap between colliding nuclei and provides estimates for important parameters such as the number of binary nucleon-nucleon interactions (N_{coll}), the interaction probability, the number of participant nucleons (N_{part}) involved in the collision, and the impact parameter.

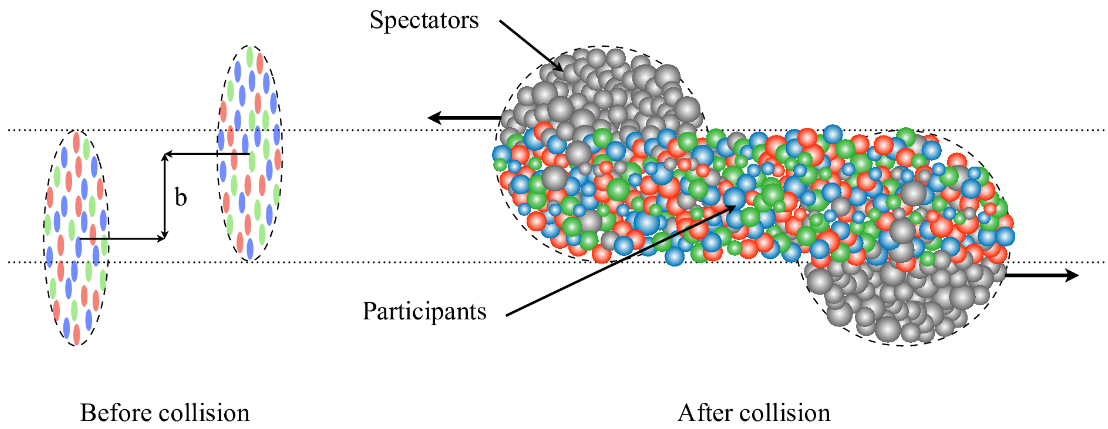


Figure 6: A schematic view of a heavy-ion collision. The impact parameter b is shown as well as the spectator nucleons and the participant nucleons. Figure taken from [3]

Consider a scenario where a projectile nucleus B collides with a target nucleus A at an impact parameter b [Fig 7].

The probability that a baryon in nucleus B at a distance b relative to another baryon in nucleus A will collide is proportional to the transverse area element $d\vec{b}$, defined by the thickness function $t(\vec{b})d\vec{b}$. The thickness function $t(\vec{b})$ is normalized as:

$$\int t(\vec{b})d\vec{b} = 1 \tag{4}$$

This represents the total collision probability. The probability of an inelastic baryon-baryon collision is given by $t(\vec{b})\sigma$, where σ is the cross section.

To derive subsequent equations, consider two flux tubes at a displacement $\vec{s}$ from the center of nucleus A, and at $\vec{s} - \vec{b}$ from the center of nucleus B. During the collision, these flux tubes overlap, which is crucial for determining the particles produced in the collision.

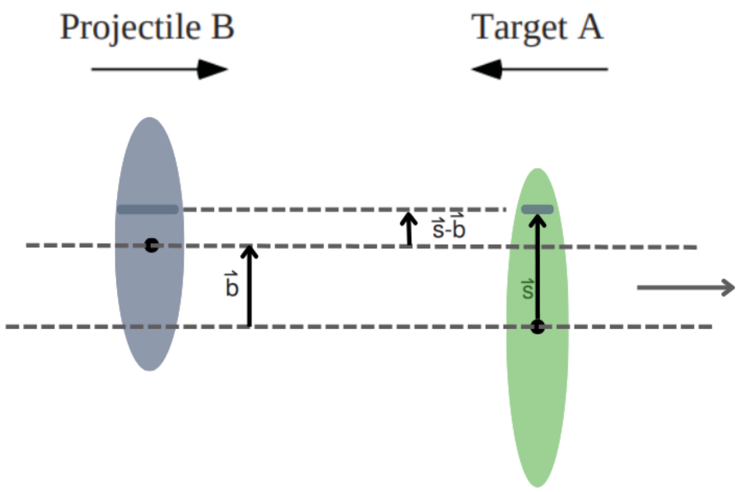


Figure 7: Projectile nucleus B collides with a target nucleus A at an impact parameter b

The probability of finding a baryon within a volume element $d\vec{s}dz_A$ in nucleus A at position $(\vec{s}, z_A)$ is given by:

$$\rho_A(\vec{s}, z_A) \cdot d\vec{s}dz_A \tag{5}$$

where ρ_A is the density function normalized by the number of baryons in the nucleus.

The probability per unit transverse area of finding a nucleon in the target flux tube is:

$$\int \rho_A(\vec{s}, z_A) \cdot dz_A = 1 \tag{6}$$

A similar expression holds for nucleus B. The product $\int \rho_A(\vec{s}, z_A)dz_A \int \rho_B(\vec{s} - \vec{b}, z_B)dz_B$ represents the joint probability per unit area of nucleons being located in the overlapping target and projectile flux tubes of differential area d^2s:

$$T(\vec{b}) = \int T_A(\vec{s})T_B(\vec{s} - \vec{b})d^2s \tag{7}$$

The probability of an interaction occurring is then:

$$P = T(\vec{b})\sigma_{in} \tag{8}$$

Finally, the probability of having n interactions between nuclei A and B follows a binomial distribution:

$$P(n,\vec{b}) = \begin{pmatrix} AB \\ n \end{pmatrix} [T(\vec{b})\sigma_{in}]^n [1 - T(\vec{b})\sigma_{in}]^{AB-n} \tag{9}$$

Here, the first term represents the number of ways to choose n collisions out of the possible AB nucleon-nucleon interactions. The second term gives the probability of exactly n collisions, while the last term represents the probability of exactly $AB - n$ non-interactions.

This probability distribution can be used to derive various reaction quantities that are crucial for understanding the collision dynamics:

$$N_{\text{coll}}(b) = \sum_{n=1}^{AB} nP(n,b) = AB\hat{T}_{AB}(b)\sigma_{\text{inel}}^{\text{NN}} \tag{10}$$

$$N_{\text{part}}(b) = A\int \hat{T}_A(s)\left\{1 - \left[1 - \hat{T}_B(s-b)\sigma_{\text{in}}\right]^B\right\} d^2s + \tag{11}$$

$$B\int \hat{T}_B(s-b)\left\{1 - \left[1 - \hat{T}_A(s)\sigma_{\text{in}}\right]^A\right\} d^2s$$

Impact parameter and the percentage centrality of collisions

The total probability for the occurrence in the collision of A and B at an impact parameter b is

$$f(b) = 1 - [1 - T(\vec{b})\sigma_{in}]^{AB} \tag{12}$$

The total inelastic cross section can be written as

$$\sigma_{in} = 2\pi \int_0^{\infty} f(b)bdb \tag{13}$$

The fraction of cross section within bmin to bmax is given by:

$$F = \frac{2\pi}{\sigma_{in}} \int_{b_{min}}^{b_{max}} f(b)bdb \tag{14}$$

The average of impact parameter can be calculated for a centrality bin.

$$\langle b \rangle = \frac{2\pi}{F} \int_{b_{min}}^{b_{max}} bf(b)bdb \tag{15}$$

The avergae of nuclear overlapiing function can be calculated as a function of impact parameter b :

$$\langle T \rangle = \frac{2\pi}{F} \int_{b_{min}}^{b_{max}} T(b)bdb \tag{16}$$

The average number of total participant from both projectile and target as a function of impact parameter b is given by :

$$\langle N_{part} \rangle = \frac{2\pi}{F} \int_{b_{min}}^{b_{max}} N_{part}(b)bdb \tag{17}$$

The average number of nuclei-nuclei collision as a function of impact parameter b is given by :

$$\langle N_{coll} \rangle = \frac{2\pi}{F} \int_{b_{min}}^{b_{max}} N_{coll}(b)bdb \tag{18}$$

We can see that for different impact parameter ranges we could create a table with its corresponding centrality percentage, and $\langle b \rangle$, $\langle N_{coll} \rangle$, $\langle N_{parte} \rangle$, etc.

2.6.2 p-p

The Glauber model is traditionally applied to heavy ion collisions, but in small system using the anisotropic and inhomogeneous density profile of proton, recent studies [5] found that the model reproduces the charged particle multiplicity distribution of p + p collisions at LHC energies very well. Collision geometric properties can also be determined.

In their study, [5], they use a model with fluctuating proton orientation and it has three effective quarks and gluonic flux tubes connecting them FIG. 8

Now the collision plane is taken to be in x-y, hence dependence along z axis is integrated out as follows

$$T(\vec{x}, \vec{y}) = \int \rho(\vec{x}, \vec{y}, z)dz \tag{19}$$

The overlap function $Tpp(\vec{b})$ for projectile proton (A) and target proton (B) is then defined as

$$T_{pp}(b) = \int\int T_A(x - \frac{b}{2}, y)T_B(x + \frac{b}{2}, y)dxdy \tag{20}$$

Here T_{pp} is sum of 4-components namely quarkquark, quark-gluon, gluon-quark, gluon-gluon.

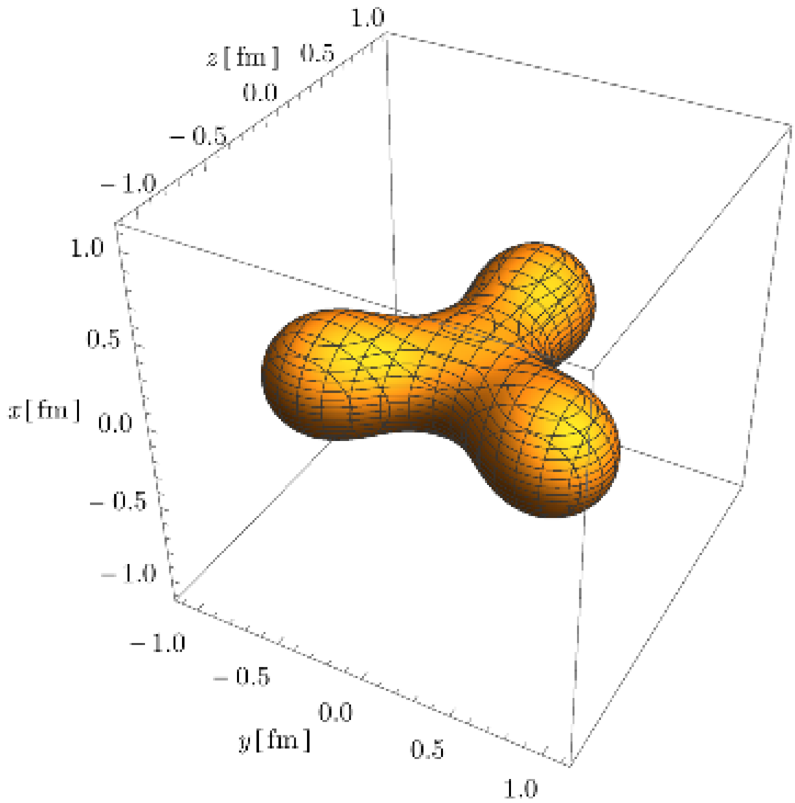


Figure 8: Figure taken from [4]. Three-quark configuration with a star-junction built from gluons

$$T_{pp}(b) = (T_{pp})_{qq} + (T_{pp})_{gg} + (T_{pp})_{qg}(T_{pp})_{gq} \tag{21}$$

We define the number of binary collisions (N_{coll}) of partons in a p + p collision at a given impact parameter (b) as follows:

$$N_{coll} = \sigma_e f f T_{pp}(b) \tag{22}$$

where σ_{eff} is the effective partonic cross sections.

In this study [5], they have considered the dependence of Nch on the number of participant partons (Npart) and Ncoll. Further, the relationship between Npart and Ncoll is considered non-linear, as with heavy ion collisions assuming a three-dimensional shape. Thus, the number of participating partons at impact parameter 'b' is given as

$$N_{part}(b) \propto N_{coll}^{1/x}(b) \tag{23}$$

where x is a parameter.

By considering, f, as a fraction of charged hadron multiplicity produced from binary collisions, we have two component model for estimation of number of charged particles given as

$$\frac{N_{ch}}{d\eta} = n_{pp}[(1-f)\frac{N_{part}}{2} + fN_{coll}] \tag{24}$$

where, npp is a constant of proportionality, which represents the charged particle multiplicity density in pseudorapidity for p + p collisions and f is a free parameter.

To estimate the charged-particle multiplicity in this study [5], for p + p collisions nucleons are replaced by partons (quarks and gluons) and σ_{in} by σ_{eff}. The model provides Npart and Ncoll, for an event with a given impact parameter and collision energy.

The centrality of proton-proton $(p+p)$ collisions is estimated by analyzing the distribution of the impact parameter b. Centrality is expressed as a percentage of the total interaction cross section, and the impact parameter distribution is utilized to determine the centrality percentiles.

The centrality percentile c_1, for instance, is calculated by integrating the impact parameter distribution from $b = 0$ to a certain value b_1, as shown in the equation below:

$$c_1 = \int_0^{b_1} \frac{dN}{db}\, db, \quad c_2 = \int_{b_1}^{b_2} \frac{dN}{db}\, db, \quad \text{etc.}$$

A Gaussian distribution with a mean of 1 and a standard deviation of 0.32 fm is used to model the impact parameter, ensuring that the distribution function vanishes beyond the proton radius (approximately 2 fm). FIG 9

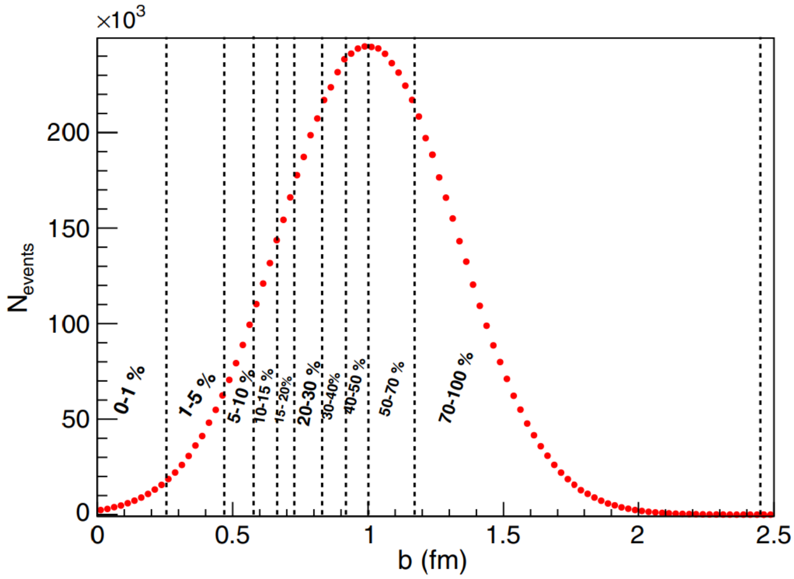


Figure 9: Figure taken from [5]. Input impact parameter (b) profile for pp collisions.

Once the ranges of impact parameters corresponding to each centrality are obtained, these are projected onto quantities such as the number of charged particles ($\langle N_{\rm ch}\rangle$), the number of participants ($\langle N_{\rm part}\rangle$), and the number of binary nucleon-nucleon collisions ($\langle N_{\rm coll}\rangle$). The multiplicity distribution for each percentile bin is shown. FIG 10

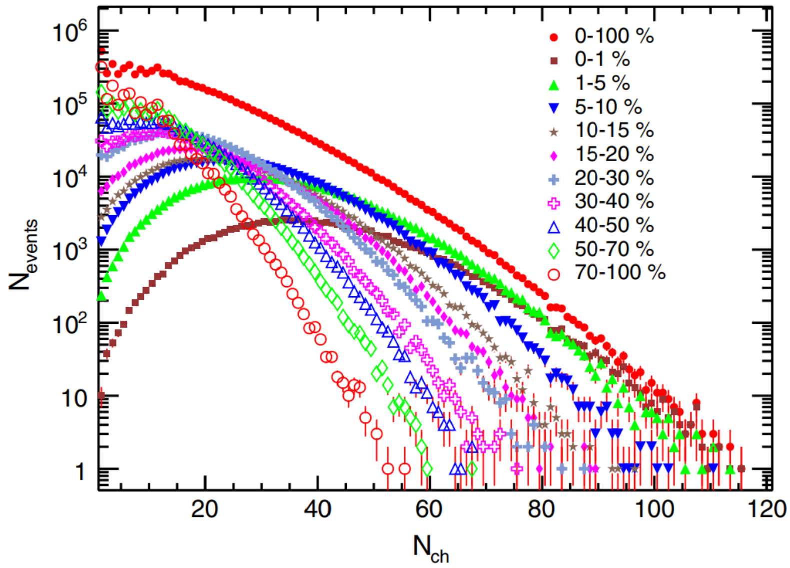


Figure 10: Figure taken from [5].Charged-particle multiplicity distribution in different percentile bins for pp collisions at $\sqrt{s} = 7$ TeV

Multiplicity %	$b - range(fm)$	$\langle N_{part}\rangle$	$\langle N_{coll}\rangle$
0-1	0 - 0.25534	13.142	31.156
1-5	0.25535 - 0.46909	11.164	24.815
5-10	0.46909 - 0.58484	9.244	19.478
10-15	0.58484 - 0.66430	8.037	16.153
15-20	0.66431 - 0.72766	7.131	13.818
20-30	0.72767 - 0.83026	6.116	11.326
30-40	0.83027 - 0.91819	5.268	9.215
40-50	0.91820 - 1.00117	4.418	7.340
50-70	1.00118 - 1.17163	3.395	5.208
70-100	1.17164 - 2.54998	1.986	2.591

Table 3: Geometric properties of pp collisions for different multiplicity classes taken from [5] TABLE I.

Part II

Experiment and Frameworks

Chapter 3: The ALICE Experiment at the LHC

3.1 The Large Hadron Collider

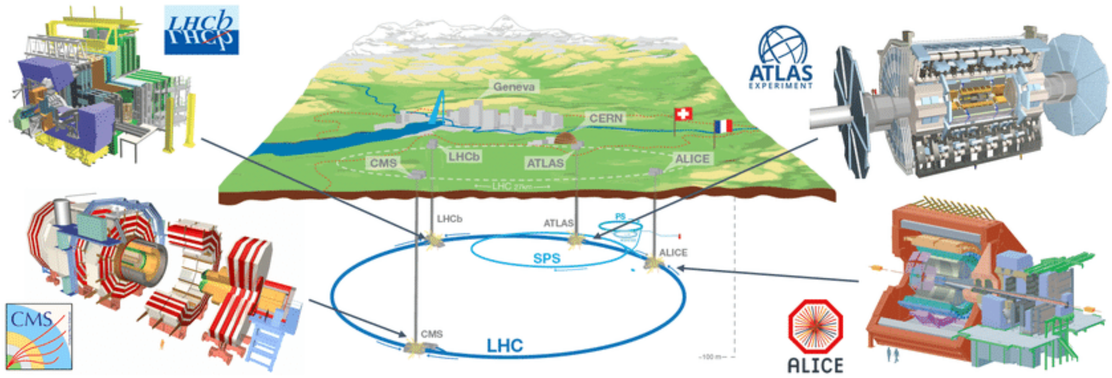


Figure 11: LHC and its four main experiments. Figure from [6]

The LHC is one of the most complex and expensive machines ever built, and it involves thousands of scientists and engineers from all over the world working together, is one of CERN's most famous machines. Imagine a super-long racetrack that's underground and circular. It consists of a 27-kilometer ring of superconducting magnets and detectors, designed to accelerate protons and heavy ions to nearly the speed of light.

The LHC has already helped discover new particles, like the Higgs boson, which is important for understanding why particles have mass.

So is the world's largest and most powerful particle accelerator, located near Geneva, Switzerland. The primary goal of the LHC is to probe fundamental questions in particle physics by creating conditions similar to those just after the Big Bang. The LHC enables collisions at unprecedented energies, providing a unique environment for studying the fundamental particles and forces that govern the universe.

The LHC is supported by a complex network of accelerators (FIG. 12) , each with a specific role in preparing particles for collisions:

- Linac 4: The linear accelerator that initially accelerates protons and heavy ions from rest.
- Booster (PSB): The Protons Synchrotron Booster accelerates these particles further before sending them to the next stage.

- PS (Proton Synchrotron): The Proton Synchrotron continues accelerating the particles to higher energies.
- SPS (Super Proton Synchrotron): The Super Proton Synchrotron further increases the energy of the particles before they enter the LHC.
- LHC (Large Hadron Collider): Finally, the LHC accelerates the particles to near the speed of light within its 27-kilometer ring, enabling high-energy collisions.

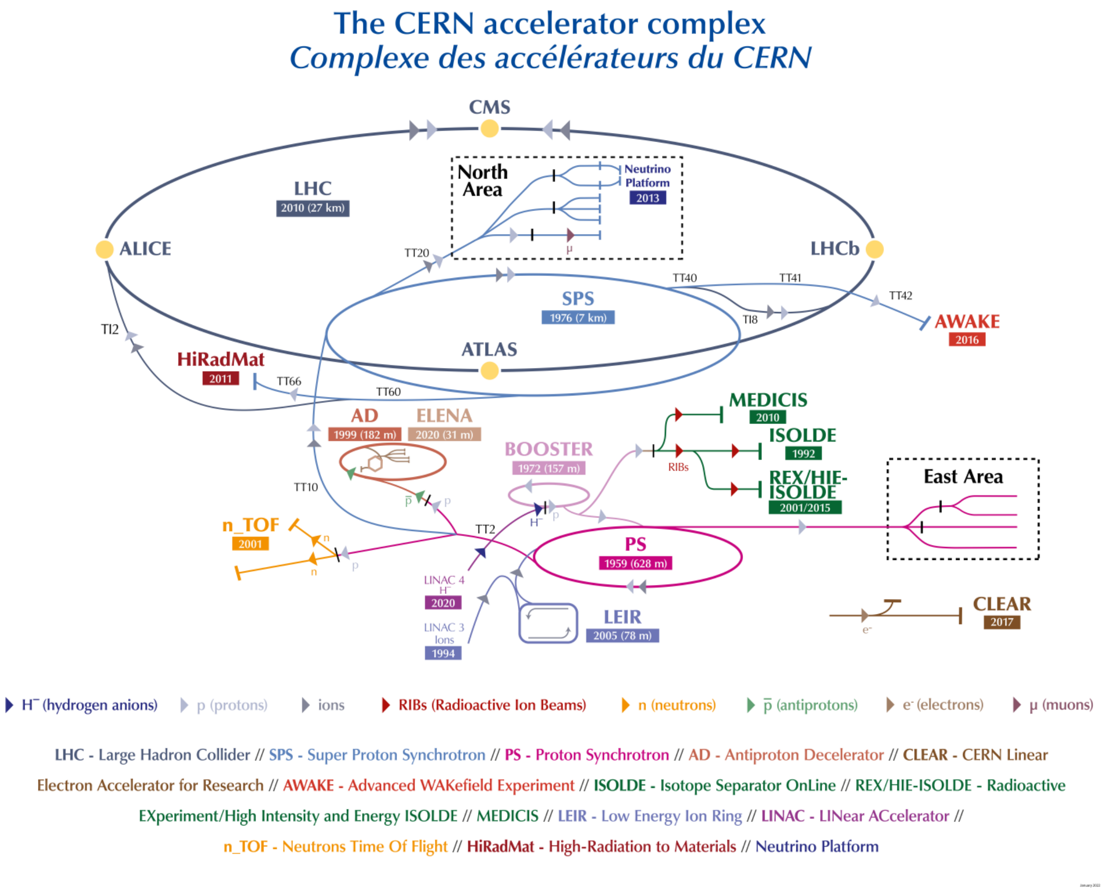


Figure 12: CERN-PHOTO-202206-116 ©

Looking to the future, CERN is planning the Future Circular Collider (FCC), a proposed successor to the LHC. The FCC aims to be even larger and more powerful, with a circumference of up to 100 kilometers. It will explore new frontiers in particle physics by probing energies and precision far beyond the capabilities of the LHC, potentially uncovering new physics beyond the Standard Model and providing deeper insights into the fundamental nature of the universe.

Is equipped with four main experiments (FIG. 11) , each focusing on different aspects of particle physics:

- ATLAS (A Toroidal LHC ApparatuS): This general-purpose detector investigates a broad range of phenomena, including the search for new particles and the study of the Higgs boson.
- CMS (Compact Muon Solenoid): Similar to ATLAS in its general-purpose approach, CMS aims to explore new physics beyond the Standard Model, including detailed studies of the Higgs boson and other fundamental particles.
- LHCb (Large Hadron Collider beauty): This experiment focuses on the properties of b-quarks (beauty quarks) to understand the differences between matter and antimatter, providing insights into why the universe is composed of matter rather than antimatter.
- ALICE (A Large Ion Collider Experiment): Dedicated to studying the quark-gluon plasma, ALICE examines the state of matter that existed just after the Big Bang by colliding heavy ions.

3.2 The ALICE Detector

In March 1992, during the "Towards the LHC Experimental Programme" meeting, the concept of constructing a dedicated heavy-ion detector for the LHC was first introduced. By 1997, the LHC Committee granted approval for ALICE to advance to the final design and construction stages. The subsequent decade focused on design and extensive research and development. The unique challenges of heavy-ion physics at the LHC required advancements beyond the technology available at the time. In some instances, achieving the project's initial design goals necessitated technological breakthroughs. A well-coordinated effort throughout the 1990s led to significant progress in detectors, electronics, and computing.

ALICE aims to explore the fundamental properties of matter by recreating and studying the conditions that existed just moments after the Big Bang, explained in more detailed in [1]. It focuses on a specific state of matter called the quark-gluon plasma, which is thought to have existed when the universe was just a tiny fraction of a second old which we already talked about.

ALICE uses powerful collisions of heavy ions, like lead nuclei, which are smashed together at incredibly high speeds. These collisions produce extremely high temperatures and densities, mimicking the conditions of the early universe, is equipped with a variety of sophisticated detectors. These detectors track and measure the particles produced in the collisions, focusing on understanding how quarks and gluons, the building blocks of protons and neutrons, interact in this extreme environment. By studying the quark-gluon plasma, we hope to learn more about the fundamental forces of nature and the evolution of the universe.

The insights gained from ALICE help us understand the very fabric of the universe, from the earliest moments after the Big Bang to the present day. It also provides important information for other fields of physics and could even have implications for our understanding of the fundamental forces that govern everything around us.

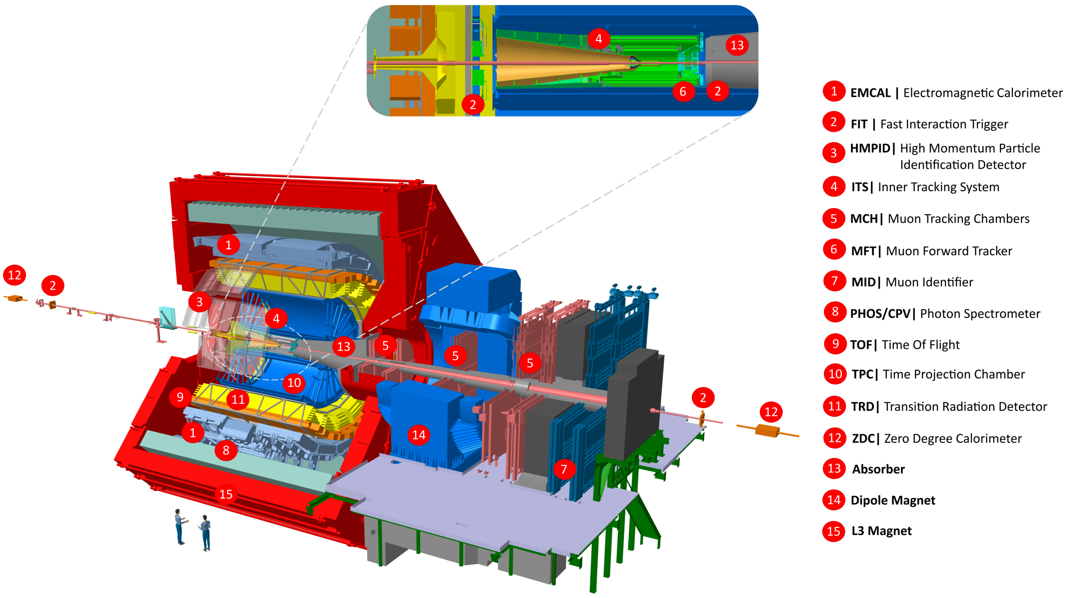


Figure 13: ALICE 2 detector systems. Figure taken from [7]

During this RUN, RUN 3, we have ALICE 2 (Fig. 15), the experiment has undergone a major upgrade during the LHC Long Shutdown 2 (2019–2022) to significantly improve the capabilities of ALICE to probe the QGP with heavy-flavour quarks

Some ALICE 2 subdetectors, explained in more detail in [16], are:

3.2.1 Inner Tracking System

The ITS2 uses ALPIDE sensors to improve vertex reconstruction and low-pT particle detection, featuring increased granularity, reduced material budget, and improved tracking efficiency.

3.2.2 Muon Forward Tracker

The MFT enhances the muon spectrometer by improving muon track pointing resolution, reducing the impact of scattering in the hadron absorber.

3.2.3 Time Projection Chamber

The TPC upgrade maintains its large active volume and pseudorapidity coverage, ensuring efficient particle tracking across multiple collision types.

3.2.4 Muon Forward Tracker

The MFT enhances the muon spectrometer by improving muon track pointing resolution, reducing the impact of scattering in the hadron absorber.

3.2.5 Fast Interaction Trigger

The FIT serves as a trigger, luminometer, and forward multiplicity counter, aiding in collision time determination, vertex location, and event plane orientation.

3.2.6 Muon System

The Muon Identifier (MID) improves the rate capability and reduces ageing, ensuring reliable detection in high-rate environments.

3.2.7 Time-of-Flight Detector

The TOF detector's upgrade focuses on continuous readout, enhancing its particle identification capabilities, especially in the intermediate momentum range.

3.2.8 High-Momentum Particle Identification

The HMPID detector identifies hadrons at high transverse momenta using Ring Imaging Cherenkov detectors, with upgrades improving readout rate and expanding particle identification capabilities.

3.2.9 Electromagnetic Calorimeter

The EMCal remains a key detector for jet and photon measurements, with upgrades ensuring continued operation and compatibility with Run 3 requirements.

3.2.10 Zero-Degree Calorimeter

The ZDC is upgraded to handle higher collision rates in Runs 3 and 4, with improvements in infrastructure and readout systems.

3.3 Centrality Determination

Centrality determination is a key aspect of analyzing heavy-ion collisions, as it provides insights into the impact parameter and the extent of the collision zone. Accurate centrality determination allows researchers to categorize collisions based on the degree of overlap between the colliding nuclei, affecting the density and temperature of the produced matter.

In ALICE 1, VZERO-A and VZERO-C detectors were used for centrality determination. See [8] for reference.

- These are non-tracking detectors, so their signal (V0M) is proportional to the charged particle multiplicity of primary and secondary tracks.
- To obtain an estimation, a signal distribution in some detecting system is usually used, which is divided into centrality classes that are then related to some interval of the impact parameter.
  - To make this relation, the Glauber model is usually used: It calculates geometric quantities of the collision.

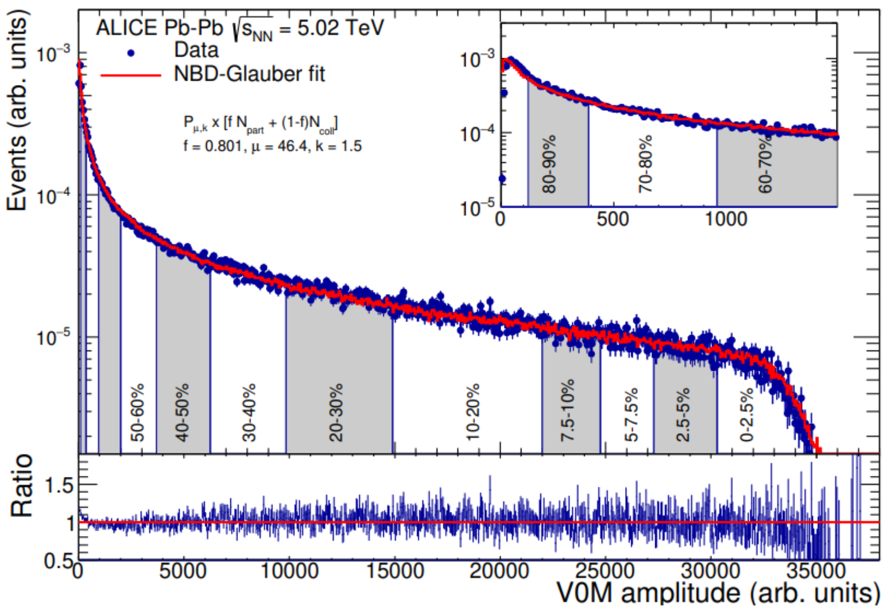


Figure 14: Figure taken from [8].

In ALICE 2, The ZDC installed on either side of the interaction point help determine the centrality and event plane orientation. Offline analysis of FIT data provides the precise collision time for TOF-based particle identification, yields the collision centrality and the event plane orientation, FV0 segmentation, combined with the information from the other forward detectors, is sufficient to yield the required centrality and event plane resolution, also the FDD provides an independent measurement of centrality based on the charged-particle multiplicity in an intermediate pseudorapidity range between the ITS and the ZDC and contributes to the selection of ultra-peripheral collisions as well as their classification into exclusive or dissociated channels.

3.4 Track and Vertex Reconstruction

Track and vertex reconstruction is essential for analyzing the trajectories and interaction points of particles produced in collisions. The ALICE detector uses advanced tracking systems to reconstruct the paths of charged particles and identify their points of origin (vertices). Accurate track and vertex reconstruction is crucial for precise measurements of particle properties and for understanding the collision dynamics.

3.5 Particle Identification

Particle identification (PID) is a fundamental task in high-energy physics experiments, allowing researchers to distinguish between different types of particles based on their properties such as mass, charge, momentum, and energy loss. Accurate PID is essential for understanding the underlying physics in particle collisions, reconstructing the event topology, and identifying rare processes.

In the ALICE experiment at the Large Hadron Collider (LHC), a combination of several complementary techniques and detectors is employed to achieve robust PID. These include the Time Projection Chamber (TPC), the Time-of-Flight (TOF) detector, the Transition Radiation Detector (TRD), and the Electromagnetic Calorimeter (EMCal), among others. Each detector contributes uniquely to the identification process, covering different ranges of momentum and particle types, thereby allowing ALICE to perform comprehensive analyses of the collision events.

3.5.1 Time Projection Chamber (TPC)

The Time Projection Chamber (TPC) is the central tracking and particle identification detector in ALICE. It is the largest TPC ever constructed and plays a crucial role in identifying charged particles over a wide range of momenta. The TPC is a cylindrical gas-filled chamber with a high voltage central electrode, creating a uniform electric field along the axis of the cylinder.

As charged particles pass through the TPC, they ionize the gas molecules along their path. The resulting ionization electrons drift towards the end plates of the chamber under the influence of the electric field. These electrons are then amplified and collected by multiwire proportional chambers, creating a three-dimensional image of the particle's trajectory.

In addition to tracking, the TPC provides particle identification through the measurement of the ionization energy loss (dE/dx) as the particle traverses the gas. The dE/dx measurement is momentum-dependent and varies according to the particle's mass and charge. By comparing the measured dE/dx with expected values for different particle species, the TPC can distinguish between particles such as electrons, pions, kaons, and protons, especially in the low to intermediate momentum range (up to a few GeV/c).

The TPC's high granularity and large acceptance make it particularly effective in detecting and identifying particles produced in the dense environment of heavy-ion collisions, where multiple tracks can overlap and intersect.

3.5.2 Time-of-Flight (TOF)

The Time-of-Flight (TOF) detector is a crucial component in the particle identification (PID) system of the ALICE experiment. It complements the TPC by providing PID in the intermediate momentum range, where dE/dx separation becomes less effective. The TOF detector identifies particles by measuring the time t it takes for charged particles to traverse a known distance L from the collision point to the detector.

The TOF system determines the particle's velocity $v = \frac{L}{t}$, which, when combined with the momentum p measured by the TPC, allows for the calculation of the particle's mass m using the relation:

$$m = \frac{p}{\gamma v}$$

where γ is the Lorentz factor. This velocity-based identification method is particularly effective for distinguishing between particles such as pions, kaons, and protons at intermediate momenta, up to a few GeV/c.

The ALICE TOF detector employs Multigap Resistive Plate Chambers (MRPCs) to achieve the required timing precision. The detector covers a cylindrical surface area of approximately 141 m^2, located 3.7 meters from the interaction point, with full azimuthal coverage and a polar acceptance range of $-0.9 < \eta < 0.9$. It consists of 1593 MRPCs organized into 18 azimuthal sectors, each containing a supermodule (SM) composed of five modules. Each MRPC is subdivided into two rows of 48 pickup pads, providing a total of 152,928 readout channels.

Initially, during LHC Run 1, the TOF achieved a time resolution of 80 ps, which was further improved to 56 ps after a more refined calibration using Pb-Pb 2015 data. The high precision of the TOF detector is enhanced by custom fast amplifiers and discriminators (NINO) on the front-end cards, which send signals to Time-to-Digital Converter (TDC) readout modules (TRM) using high-performance time-to-digital converter chips with 24.4-ps least significant bit (LSB) resolution.

The TOF detector's performance enables 3σ separation of pions and kaons up to 2.5 GeV/c and protons up to 4 GeV/c on a track-by-track basis. This high-resolution PID capability is critical for analyzing particle spectra and understanding the underlying physics in high-energy collisions at ALICE.

3.6 The ALICE Online-Offline Framework O^2

The ALICE Online-Offline Framework, known as O^2, is a comprehensive data acquisition and processing system designed to handle the massive amounts of data generated by the ALICE experiment. The O^2 framework encompasses both the online and offline components of data handling. The online part manages real-time data acquisition and initial processing during collisions, while the offline part involves more detailed data analysis and storage. This framework is crucial for efficiently managing the high data throughput and ensuring the quality and accuracy of the experimental data.

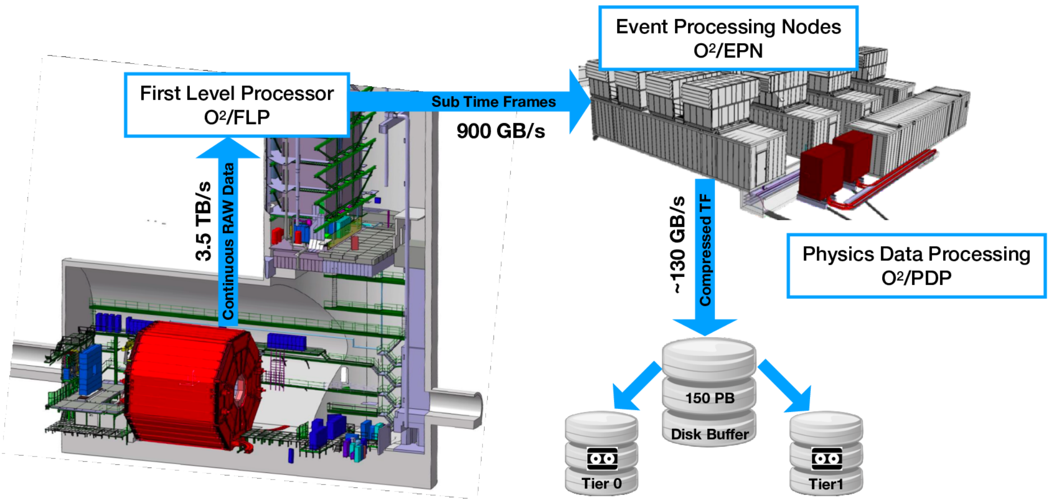


Figure 15: Overview of the components of the O^2 data read out and processing systems and the main data flows. Figure taken from [7]

Chapter 4: Frameworks

4.1 ROOT: Scientific Data Analysis at Scale

Data from high-energy collision experiments usually comes in what is called a ROOT file, the initial step with ROOT often involves handling this kind of files, a standard format used to store vast quantities of structured data.

ROOT is an open-source data analysis framework, developed at CERN for statistical analysis and visualization of your data, providing tools for managing large datasets, executing complex computations, and generating detailed plots and histograms. ROOT plays a crucial role in the data analysis workflows of experiments like ALICE, enabling researchers to efficiently handle and analyze the massive volumes of data generated by high-energy collisions.

ROOT allows you to store data in a compressed binary format within its ROOT files. Each file includes a detailed description of the object format, enabling the automatic generation of C++ code for the corresponding classes. The framework's tree-like data structure is particularly powerful, offering rapid access to large datasets—orders of magnitude faster than conventional file access. ROOT files have a hierarchical structure, containing directories within them.

Primarily written in C++, ROOT is compatible with Linux, macOS, and Windows; I personally utilize it effectively on Ubuntu. ROOT includes a C++ interpreter, which is ideal for rapid prototyping, and integrates seamlessly with Python via its dynamic and robust Python-C++ binding.

4.1.1 Selection Cuts in ROOT

The primary goal of selection cuts is to focus on the most relevant events by applying a set of conditions or criteria. This filtering process effectively reduces the number of events to be analyzed but increases the overall significance of the signal.

In a typical ROOT analysis, selection cuts are implemented using conditional statements, often within a loop that iterates over all the entries in the dataset. For example:

```
for (int i = 0; i < nEntries; ++i) {
    tree->GetEntry(i);
    if (condition1 && condition2) {
        hist.Fill(filtered_event);
```

```
    }
}
```

ROOT Trees: Structured Data for Efficient Access

A fundamental feature of the ROOT framework is the use of ROOT Trees, which are specialized data structures designed for the efficient storage and retrieval of large datasets. A ROOT Tree consists of branches, each of which can contain multiple leaves. The branches represent different data categories, while the leaves store individual data points or variables within those categories. This hierarchical structure allows for the efficient organization of complex datasets, where each branch can hold different types of information, such as particle momenta, energies, or detector hits.

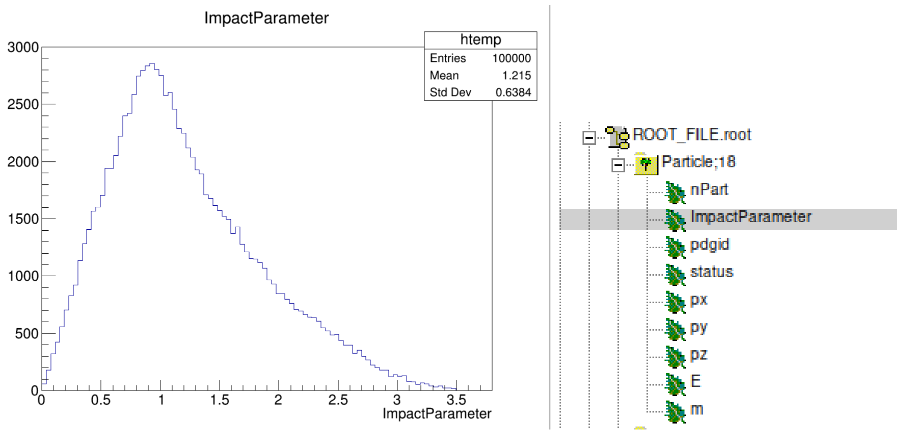


Figure 16: Example of a ROOT tree file structure

One of the key advantages of ROOT Trees is their ability to store data in a columnar format. This means that data for a particular variable (leaf) across many events (entries) is stored contiguously on disk. As a result, when analyzing specific variables, ROOT can quickly access the relevant data without needing to read the entire dataset into memory. This columnar storage approach significantly enhances the speed and efficiency of data retrieval, especially when dealing with petabytes of data.

ROOT Trees are essential for performing selective data analysis, known as "skimming," where only a subset of the data is read and processed. By reading only the relevant branches and leaves, researchers can focus on specific aspects of the data, such as selecting events of interest based on certain criteria, without the overhead of processing the entire dataset.

4.1.2 Uproot

To enhance the accessibility of ROOT files for analysis, the Python library Uproot is employed. Uproot allows for the reading and processing of ROOT files without the need for the ROOT software itself. It facilitates the extraction of data in a more Pythonic and efficient manner, enabling integration with modern data analysis tools and workflows. Uproot supports various ROOT file structures, allowing for the manipulation of ROOT data using popular Python libraries such as NumPy and Pandas.

4.2 Python: A Versatile Tool for Data Analysis

Python is a versatile programming language widely adopted in scientific computing and data analysis due to its simplicity and extensive library ecosystem. In high-energy physics, Python is used for a range of tasks, including data manipulation, analysis, and visualization. Libraries such as NumPy, SciPy, Pandas, and Matplotlib are commonly utilized to process and interpret experimental data, complementing tools like ROOT and Uproot. Python's expansive ecosystem makes it an indispensable tool for researchers working with complex datasets and developing innovative analysis techniques.

Chapter 5: ALICE Data and Simulations

5.1 Experimental Data

Experimental data refers to the raw data collected from the ALICE detector during high-energy collisions. This data includes measurements of particle tracks, energy deposits, and various other observables. Analyzing experimental data involves processing and interpreting these measurements to extract meaningful information about the collisions and the properties of the produced matter. The data is used to test theoretical models, validate simulations, and uncover new physics phenomena.

5.2 Monte Carlo

Monte Carlo Simulation is a mathematical technique used to estimate the potential outcomes of uncertain events. By modeling a process or system with random variables, this method generates a wide range of possible outcomes through random sampling. The results are then averaged to provide an estimate of the likely outcome.

The name "Monte Carlo" is derived from the famous casino, symbolizing the concept of using repeated trials—like playing multiple hands in a poker game—to observe and calculate the probability of various outcomes rather than relying on direct calculation.

An example in physics is that during World War II, the Monte Carlo method played a crucial role in the development of the nuclear bomb as part of the Manhattan Project. Scientists used this technique to simulate and understand the complex chain reactions involved in nuclear fission. The ability to model these reactions through random sampling allowed for more accurate predictions and was instrumental in the successful development of the atomic bomb.

MC in particle physics

When dealing with strong interactions, predicting the probability distribution becomes an almost insurmountable challenge due to the unsolved nature of these interactions. At very small momentum transfers, the sheer number of concurrent interactions makes it impossible to handle them with perturbation theory, rendering direct calculations unfeasible. Even if we could theoretically solve Quantum Chromodynamics (QCD), the resulting expressions would be prohibitively large and complex, far beyond manageable limits. Thus, the event generator approach seeks to decompose these intricate problems into smaller, more manageable tasks. Instead of generating a single event, large samples of events are produced, similar to experimental practices. Here, random numbers are utilized to represent the potential quantum mechanical choices, which is where Monte Carlo methods become indispensable.

Monte Carlo simulations play a crucial role in modeling particle interactions and predicting collision outcomes. By employing random sampling techniques, these simulations enable the modeling of complex physical processes, generating synthetic data that can be rigorously compared with experimental results.

Event generators focus specifically on simulating the fundamental particle interactions and the resultant final states. They model the underlying physics processes, including the production and decay of particles in high-energy collisions. In contrast, detector generators simulate the passage of these particles through a detector, accounting for how the detector responds to different types of particles.mTogether, event generators and detector generators provide a comprehensive simulation framework, enabling researchers to interpret complex collision events and optimize experimental setups.

Event generators encompass a wide array of processes, as illustrated in Figure 17. This figure depicts the various components involved in the transition from two incoming hadrons to a complex final state characterized by numerous particles and interactions with MC event generator PYTHIA.

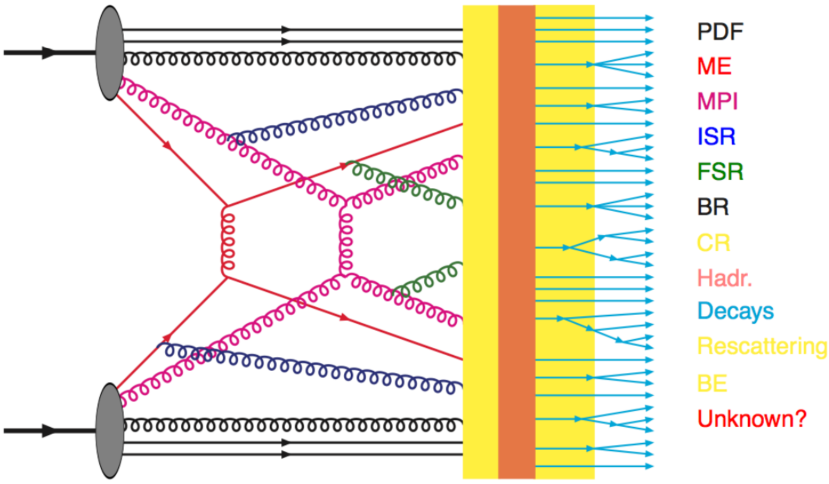


Figure 17: Schematic representation of the different components in an event generator, illustrating the transformation from two incoming hadrons to a final state with multiple particles and interactions.

In [17], there's a compelling analogy where different processes in particle physics are likened to planning a complete dinner:

How to Plan a Complete Dinner:

1. Choose the main course ($\approx$ hard process = ME $\oplus$ PDF)
2. Select a complementary first course ($\approx$ ISR)
3. Decide on a matching dessert ($\approx$ FSR)
4. Add side dishes and drinks ($\approx$ MPI)
5. Offer coffee and cookies ($\approx$ hadronization)
6. Prepare after-dinner snacks ($\approx$ decays)
7. Set the table with plates, cutlery, and settings ($\approx$ administrative structure)
    - Thousands of possible (published) recipes
    - Countless combinations
    - Never identical results (variations in meat, spices, temperature, etc.)

A Higgs event $\approx$ having beef for dinner.

The analogy begins with the main course, which, though not served first, determines the rest of the meal. Similarly, in particle physics, selecting a main course is like choosing the hard process you're interested in, such as producing a Higgs event. This involves calculating probabilities using matrix elements and parton distribution functions to determine if the Higgs event will actually occur.

Once the main course is chosen, you then select a matching first course, akin to initial state radiation (ISR), and a dessert, akin to final state radiation (FSR), which describes the evolution of the event after the hard process at the perturbative level.

The side dishes and drinks represent the multiple parton interactions (MPI), which are further detailed in the PYTHIA section. Just as a meal isn't complete without side dishes, MPIs represent other physical processes occurring alongside the Higgs event.

Finally, coffee and tea correspond to the transition from perturbative theory to the formation of bound states of quarks, leading to hadronization and subsequent decays.

5.2.1 PYTHIA - The Lund Monte Carlo

Figure 18: The PYTHIA logo, seen in the upper left corner of this web-site ([9]), is a stylized representation of the famous artwork from a drinking bowl, which on the left depicts the Oracle of Delphi.

According to its own website, [9]: PYTHIA is a program for the generation of high-energy physics collision events, i.e. for the description of collisions at high energies between electrons, protons, photons and heavy nuclei. It contains theory and models for a number of physics aspects, including hard and soft interactions, parton distributions, initial- and final-state parton showers, multiparton interactions, fragmentation and decay. It is largely based on original research, but also borrows many formulae and other knowledge from the literature. As such it is categorized as a general-purpose Monte Carlo event generator.

PYTHIA
Color Reconnection
nMPI

Table 4: Some PYTHIA features I use in this study. Explanation below

LUND String Model Overview

The Lund model within `PYTHIA` 8.3 provides a detailed and realistic description of how quark-antiquark pairs fragment into colorless hadrons after a high-energy particle collision. It is a key tool for understanding the complex processes involved in hadronization and for simulating the outcomes of particle physics experiments

Quark-Antiquark Pair Production

- **Initial State:** After a high-energy collision, quarks and gluons are produced in a process called parton showering. These partons (quarks and gluons) carry color charge, which is a fundamental property in quantum chromodynamics (QCD).
- **Color Confinement:** Quarks cannot exist in isolation due to color confinement, a principle of QCD that requires color charges to neutralize each other, forming colorless states. To neutralize their color charge, quarks must combine with antiquarks or other quarks, forming hadrons.

String Formation

- **String Model:** In the Lund model, the interaction between quarks and antiquarks is modeled as a string of color flux, analogous to an elastic band stretched between the quark and antiquark. The energy of the string increases as the quarks move apart.
- **Potential Energy:** The potential energy stored in the string due to its tension is proportional to the distance between the quark and the antiquark. This potential energy eventually becomes large enough to create new quark-antiquark pairs from the vacuum (via the process known as pair production).

String Breaking

- **Breaking the String:** As the quarks move further apart, the string between them becomes stretched, and at some point, it "snaps." When this happens, the energy in the string is converted into mass, producing a new quark-antiquark pair. This newly produced pair then combines with the original quarks to form new hadrons. FIG. 19

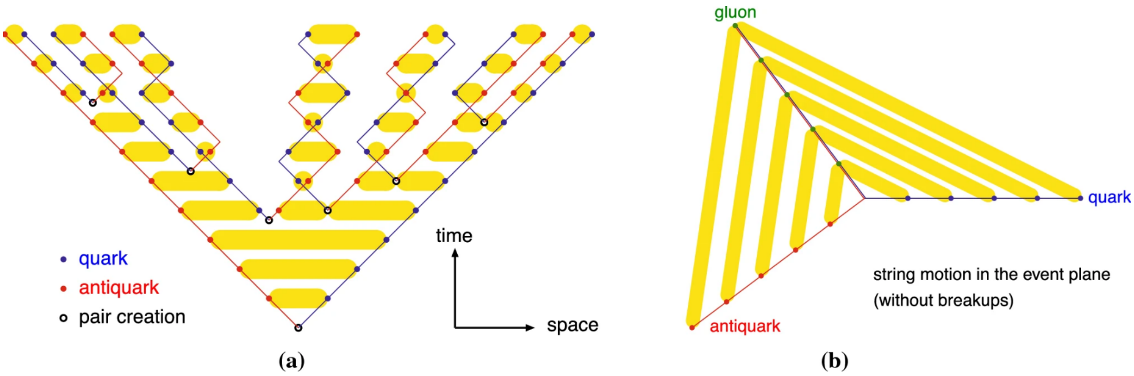


Figure 19: Figure taken from [10]. (a) String breakup in a $q\bar{q}$ event. The points denote the location of quarks and antiquarks at snapshots in time, and the yellow regions the string pieces then stretched out between them. (b) String drawing in the plane of a $q\bar{q}g$ event

- **Multiple Breakings:** The process can repeat multiple times, leading to the formation of several hadrons from the initial quark-antiquark pair. Each breaking of the string creates a new pair of hadrons, ultimately resulting in a cascade of hadrons.

Hadronization

- **Formation of Hadrons:** The quarks and antiquarks created during string breaking combine to form hadrons. For example, a quark might combine with an antiquark to form a meson, or three quarks might combine to form a baryon.

- **Fragmentation:** The process of multiple string breakings and the formation of hadrons is known as fragmentation. The Lund model provides a probabilistic description of how these hadrons are distributed in terms of their momenta and other properties.

Colorless Hadrons

- **Color Neutrality:** The final hadrons produced must be colorless, meaning that their constituent quarks and antiquarks combine in such a way that the overall color charge is neutralized. This is consistent with the principle of color confinement in QCD.

In the context of particle collisions, such as those studied in experiments like the LHC, the Lund model is crucial for simulating the final states of collisions. These simulations help physicists understand what kinds of particles are likely to be produced in a collision and how they will be distributed in terms of energy and momentum. `PYTHIA` 8.3 uses the Lund string model to generate these final states, which can then be compared with experimental data to test theories of particle physics.

Colour Reconnection

It refers to the process where the color connections between quarks and gluons are rearranged after a high-energy collision, but before hadronization occurs.

Why Colour Reconnection?

In a particle collision, multiple partons (quarks and gluons) are produced, often in complex configurations due to multiple parton-parton interactions (MPI). These partons are connected by color strings, which represent the color field lines between quarks and gluons. Initially, these connections are formed based on the color flow from the hard scattering process, but they may not be optimal for hadronization, which requires color-neutral (colorless) hadrons. Mechanism of Colour Reconnection

- Reconnecting Color Strings: The CR mechanism allows these color strings to rearrange themselves in such a way that minimizes the total energy of the system. This rearrangement involves "reconnecting" the strings between different quarks and gluons, which can lead to a more energetically favorable configuration.
- Hadronization and Final States: By reconnecting the color strings before hadronization, the process can lead to more realistic final states. For example, it can produce more tightly bound systems that resemble the hadrons observed in experiments. It also helps in reducing the number of unphysical events where the energy of the final state particles would be unrealistically high.
- Impact on Observables: CR can significantly impact several observables in high-energy physics, such as jet shapes, multiplicities, and the transverse momentum distribution of particles. It is particularly important in events with many partons, such as those involving top quarks or in the dense environment of heavy-ion collisions.

nMPI (Number of Multiple Parton Interactions)

The term **nMPI** refers to the number of Multiple Parton Interactions (MPI) that occur within a single high-energy collision, particularly in hadron-hadron collisions. In the context of particle physics and simulation tools like `PYTHIA`, understanding and modeling nMPI is crucial for accurately describing the complex environment created during collisions.

- **Parton Interactions:** In high-energy collisions, not only do the primary partons (quarks and gluons) from the colliding hadrons interact, but secondary interactions can occur between other partons within the same hadrons. These secondary interactions are what we refer to as Multiple Parton Interactions (MPI).

- **Energy Dependence:** The probability of MPI occurring increases with the collision energy. At higher energies, more partons become available to participate in interactions, making MPI an essential component to consider in high-energy physics experiments.

- **Impact on Final States:** nMPI significantly affects the final state of the collision, contributing to the production of additional particles, modifying jet structures, and increasing the overall multiplicity of events. Correctly accounting for MPI is vital for making accurate predictions and comparisons with experimental data.

Modeling nMPI in PYTHIA

- **Simulating MPI:** In `PYTHIA`, MPI is simulated by allowing for multiple independent scatterings between partons. These interactions are modeled using parameters that are tuned based on experimental data. The nMPI value is a key parameter in determining the number of these interactions.

- **Impact Parameter:** The likelihood of MPI is related to the impact parameter of the collision, which is the distance between the centers of the colliding hadrons. A smaller impact parameter typically results in more MPI, leading to a higher nMPI value.

- **Role in Event Generation:** During event generation in `PYTHIA`, nMPI is used to simulate the environment of a high-energy collision, influencing the distribution of particles and their momenta in the final state. The resulting particle distributions can then be compared with experimental data for validation of theoretical models.

The concept of nMPI is a fundamental aspect of modern particle physics, especially in the analysis of high-energy hadron collisions. It plays a critical role in determining the outcome of particle interactions and the properties of the final state. Tools like `PYTHIA` are indispensable for simulating nMPI and providing insights into the behavior of partons under extreme conditions. Understanding nMPI allows physicists to make more accurate predictions, refine theoretical models, and better interpret the results of experimental data.

5.2.2 EPOS

A central concept in EPOS (Enriched Pomeron Exchange Model) is the "core-corona" model, which differentiates regions of high and low energy loss during high-energy collisions. In this model, if the energy loss of a prehadron exceeds its own energy, the prehadron is absorbed into the "core," a region with high energy density that is described using hydrodynamic models such as vHLLE. If the energy loss is less than the prehadron's energy, it escapes and becomes part of the "corona," a region dominated by hadronic interactions, which is modeled using the Ultra-relativistic Quantum Molecular Dynamics (UrQMD) approach.

This distinction between core and corona is crucial for understanding the different phases of matter produced in high-energy collisions. When the energy density surpasses a critical threshold (ϵ), the matter enters a deconfined state, indicative of QCD (Quantum Chromodynamics) matter. EPOS effectively describes the production and behavior of light hadrons across various collision systems, from proton-proton (pp) to nucleus-nucleus (AA) interactions, especially in its latest iteration, EPOS4.

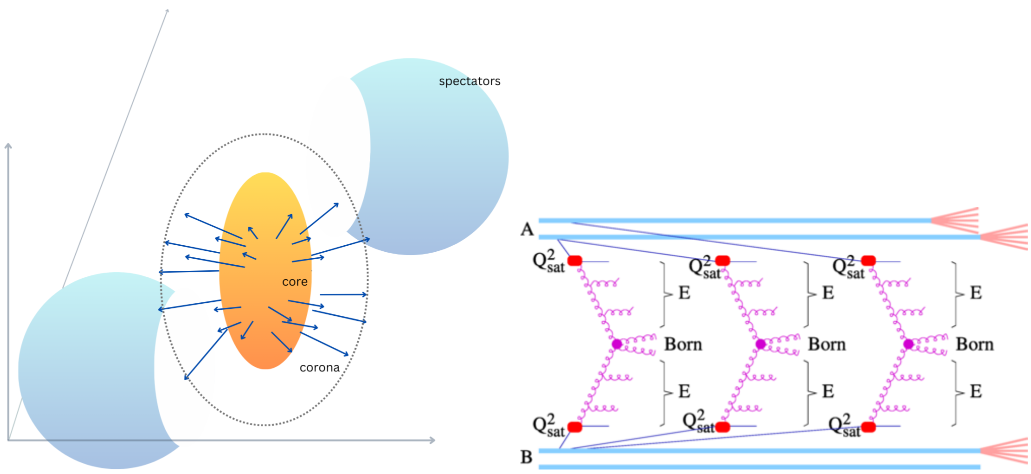


Figure 20: (LEFT) Core-corona picture (RIGHT) Three parallel scatterings. Figure taken from [11].

Heavy quarks, such as charm and bottom quarks, are another area where EPOS excels. These quarks are produced primarily through initial hard scatterings, which can be described by two mechanisms: space-like cascades, where gluon-gluon ($g+g$) or quark-antiquark ($q+\bar{q}$) interactions result in heavy quark-antiquark pairs ($Q+\bar{Q}$), and time-like cascades, which further develop these processes.

By capturing these complex dynamics, EPOS significantly contributes to the theoretical framework required to decode the intricate phenomena observed in high-energy collisions, providing a vital link between theory and experiment.

EPOS-LHC	EPOS 4
Fusion ON	Hydrodynamics
Fusion OFF	Core-corona

Table 5: EPOS configurations for EPOS-LHC (LEFT) and EPOS-4 (LEFT)

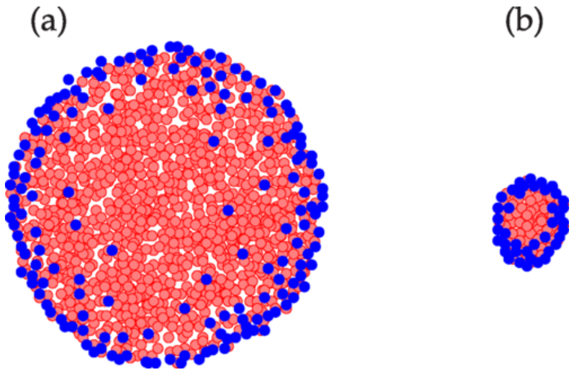


Figure 21: Sketch of the core-corona separation for a (a) “big” and a (b) “small” system. The dots are prehadrons in the transverse plane; red refers to core, blue to corona.Figure taken from [11].

In EPOS-LHC, an important feature that enhances the model's accuracy is the concept of ”Fusion.” Fusion ON and Fusion OFF refer to whether partons (quarks and gluons) from multiple strings formed during the collision are allowed to merge or ”fuse” into fewer, more energetic strings. Fusion ON leads to a denser, more thermalized medium, conducive to the formation of quark-gluon plasma (QGP), and results in higher energy density in the core region. Fusion OFF disables this merging process, leading to a more fragmented, less dense medium, favoring the corona over the core and thus not promoting QGP formation.

An analogous concept exists in EPOS4 with ”Hydro ON” and ”Hydro OFF.” When Hydro ON is enabled, hydrodynamic evolution is applied to the core region, allowing the model to account for collective flow and thermalization effects characteristic of the QGP phase. Hydro OFF bypasses this hydrodynamic treatment, focusing instead on the hadronic afterburner phase, which primarily models the hadronic interactions in the corona without considering the effects of the QGP.

Both these sets of options allow EPOS to model different aspects of particle production and energy flow, making it a versatile tool for analyzing and interpreting experimental data from heavy-ion collisions, particularly in understanding the conditions that lead to QGP formation as observed in the ALICE experiment at CERN.

Part III

Impact parameter study in proton+proton collisions

Chapter 6: Machine Learning

6.1 Overview

There are instances where writing explicit programs to accomplish complex and intriguing tasks proves challenging. To overcome this, especially when aiming to develop intelligent systems, Machine Learning has emerged as a prominent subfield within Artificial Intelligence.

During World War II, Alan Turing, often regarded as the grandfather of Machine Learning, work in developing the Bombe, a machine that could learn patterns from intercepted messages, enabled him to crack the Enigma code. This achievement not only laid the foundation for Machine Learning as a subfield of Artificial Intelligence but also allowed Turing to single-handedly won the war.

Machine learning focuses on developing algorithms that can learn from and make predictions based on data. In the context of high-energy physics, machine learning techniques are used to analyze complex datasets, identify patterns, and improve the accuracy of various measurements and predictions. Machine learning has become an essential tool for modern data analysis, providing new methods for interpreting experimental results and enhancing research capabilities.

Arthur Samuel, often referred to as the father of Machine Learning, defined the field as:

> ”The field of study that gives computers the ability to learn without being explicitly programmed.”

The two main types of Machine Learning are:

6.1.1 Unsupervised Learning

In unsupervised learning, the model is provided with unlabeled data, meaning it only receives inputs X and must find patterns or structures within the data without explicit guidance.

- **Input:** X (features)
- **Output:** None (the model must find structure in the data)

Some examples are:

- **Clustering:** Group similar data points together.

- **Anomaly Detection:** Identify unusual or outlier data points.
- **Dimensionality Reduction:** Compress data by reducing the number of variables while retaining its essential features.

6.1.2 Supervised Learning

Supervised learning involves training a model on a labeled dataset, where the correct output (often denoted as y) is provided for each input x. The model learns to map inputs to the correct outputs.

- **Input:** x (features)
- **Output:** y (label)

This is the most used type of ML, some examples are:

6.1.3 Classification

Classification involves predicting categories, where the number of possible outcomes is finite. It typically involves two or more inputs.

6.1.4 Linear Regression

Regression model doesn't predict categories, but numbers. With infinite possible outcomes.

Notation used in :

- x = input variable (feature)
- y = output variable (target)
- m = number of training examples
- (x, y) = single training example
- $(x^{(i)}, y^{(i)}) = i^{th}$ training example

So with our training set and a learning algorithm we will have a function $f(x)$ that gives us the prediction $\hat{y}$:

$$x \rightarrow f \rightarrow \hat{y}$$

6.2 Cost Function

The cost function is a fundamental tool for evaluating the performance of a model, as it measures how well the model fits the training data.

$$J(\theta_i) = \frac{1}{2m} \sum_{i=1}^{m} (\hat{y}^i(\theta_i) - y^i)^2 \tag{25}$$

In this equation, $\theta_1, \theta_2, ...$ are the parameters that can be adjusted during training to optimize the model. The goal is to determine the optimal values for these parameters such that the predicted values $\hat{y}^i$ closely match the true target values y^i for all training examples (x^i, y^i).

The cost function quantifies prediction error by calculating the difference between the predicted value $\hat{y}$ and the actual target y, known as the error. This error is squared for each training example, and the cost function aggregates these squared errors across all training examples. Specifically, the cost function sums these squared errors from $i = 1$ to m, where m represents the total number of training examples. Therefore, with more training examples, m increases, resulting in a larger cost function value due to the accumulation of more error terms.

To normalize the cost function and ensure it does not scale with the training set size, we compute the average squared error by dividing the total squared error by m. Additionally, it is conventional to divide by $2m$ to simplify subsequent calculations. While the division by 2 is not strictly necessary, it makes later calculations more convenient. Ultimately, the objective is to find the parameter values that minimize the cost function, thereby improving the model's performance.

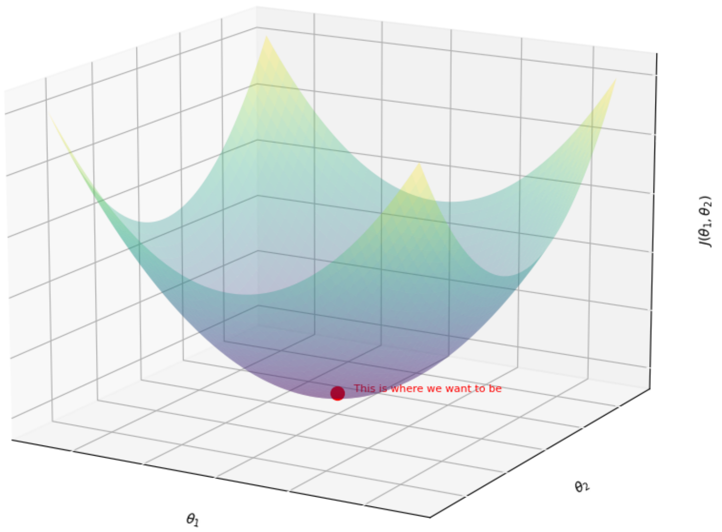


Figure 22: $J(\theta_1, \theta_2)$ at various (θ_1) and (θ_2) values

6.3 Gradient Descent

As previously discussed, our goal is to minimize the cost function $J(\theta_1, \theta_2)$. Gradient Descent is a widely used iterative optimization algorithm for finding the minimum of a function. The fundamental idea behind Gradient Descent is to start with an initial set of parameters θ and iteratively adjust them to reduce the cost function $J(\theta)$ until we converge to a minimum.

The update rule for the parameters θ at each iteration is given by:

$$\theta_{\text{new}} = \theta_{\text{old}} - \alpha \frac{d}{d\theta} J(\theta)$$

where α is the learning rate, a hyperparameter that determines the step size at each iteration while moving towards the minimum.

To better understand Gradient Descent, imagine you are standing on a mountain covered in fog, trying to reach the lowest point in the valley. Since visibility is poor, you can't see the entire landscape at once. Instead, you must rely on the slope beneath your feet to guide your descent.

- **If you take very small steps (α is too small)**, you will eventually reach the valley, but it will take a very long time. This is analogous to a slow learning rate, which leads to slow convergence.

- **If you take very large steps (α is too large)**, you might overshoot the valley, potentially ending up on another part of the mountain or even ascending back up. This represents the risk of a high learning rate, where the algorithm fails to converge or oscillates.

Therefore, selecting an appropriate learning rate is a balance between speed of convergence and stability. In practice, it is common to experiment with different values of α or use techniques like learning rate schedules or adaptive learning rates (e.g., Adam optimizer) to dynamically adjust α during training.

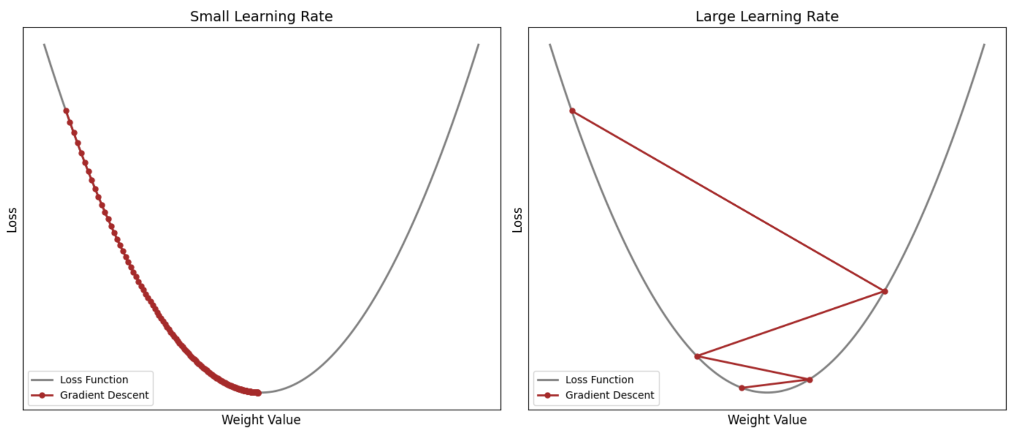
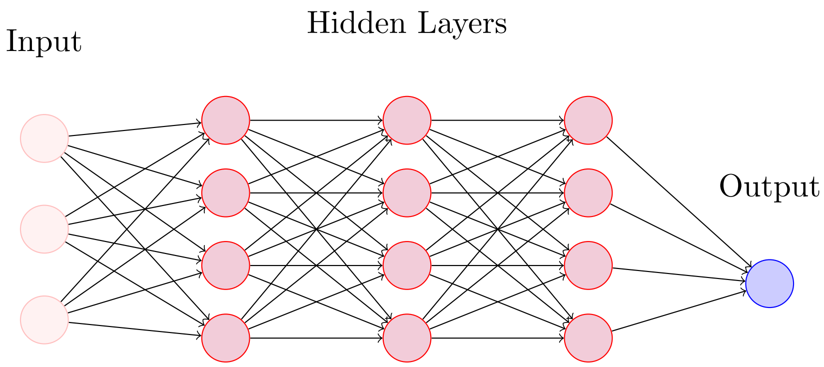


Figure 23: Small and large learning rate (α) visualization

6.4 Neural Networks

Neural networks are a class of machine learning algorithms inspired by the structure and function of the human brain. They consist of interconnected layers of nodes (neurons) that process and transform data. Neural networks are used for a variety of tasks, including classification, regression, and feature extraction.

The popularity of neural networks, particularly since the mid-2000s with the advent of deep learning, has revolutionized various fields, from speech recognition to natural language processing. This can be attributed to the availability of large datasets and advancements in computing power, such as GPUs, which have enabled the training of increasingly complex models.



As illustrated above, neural networks function by processing input data through a series of interconnected layers. The network begins with an input layer, where each node represents a feature of the input data. These inputs are then transmitted through one or more hidden layers. Each hidden layer consists of nodes (neurons) that apply specific weights and biases to the inputs, followed by an activation function that introduces non-linearity, enabling the network to learn complex patterns.

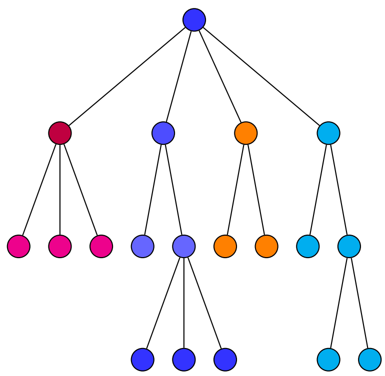
The processed information is then passed to the next layer, and this sequence continues until the data reaches the output layer, where the final predictions or classifications are made. The training process involves adjusting the parameters (weights and biases) of the network to minimize a cost function, which measures the difference between the predicted outputs and the actual target values.

By iteratively refining these parameters using techniques such as gradient descent, the neural network gradually learns to map inputs to the desired outputs. Essentially, constructing and training a neural network enables the system to autonomously learn from data, allowing it to generalize and make accurate predictions on unseen data.

Decision Trees

Decision trees are a type of machine learning algorithm that model decisions and their possible consequences using a tree-like structure. Each node in the tree represents a decision based on a particular feature, leading to branches that represent different outcomes. Decision trees are useful for classification and regression tasks and are often used in particle identification and event classification in high-energy physics.

In a decision tree, the root node represents the initial decision point, and each subsequent node corresponds to further decisions based on different criteria. The final nodes, or leaves, represent the outcome or classification result. Decision trees are easy to interpret and can handle both numerical and categorical data, making them a versatile tool in machine learning.



Random Forest

Random forests are an ensemble learning method based on decision trees, designed to improve predictive performance and reduce overfitting. Instead of relying on a single decision tree, a random forest builds multiple decision trees during training. Each tree in the forest is trained on a random subset of the data, and its splits are based on random subsets of features. This randomness introduces diversity among the trees, allowing the forest as a whole to generalize better than individual trees.

In a random forest, the final prediction is determined by averaging the predictions of the individual trees (in the case of regression) or by majority voting (in the case of classification). Due to this averaging process, random forests are less prone to overfitting than single decision trees, and they perform well on both small and large datasets.

Random forests are commonly used in high-energy physics for tasks like event classification and particle identification, where large amounts of data and complex interactions make ensemble methods particularly effective. They offer high accuracy, handle missing data well, and are robust to noisy datasets.

Chapter 7: Average Transverse Momentum versus Charged-Particle Multiplicity

The average transverse momentum, $\langle p_T \rangle$, as a function of charged-particle multiplicity, N_{ch}, was measured in proton-proton (pp) collisions at center-of-mass energies of $\sqrt{s} = 0.9$, 2.76, and 7 TeV using the ALICE detector at the LHC. The measurements were conducted within the kinematic range $0.15 < p_T < 10$ GeV/c and pseudorapidity $|\eta| < 0.3$ [12].

A rise in $\langle p_T \rangle$ with multiplicity can indicate the presence of collective flow, This flow results from the pressure gradients within the hot and dense medium created in the collision, pushing particles outward and giving them higher transverse momentum.

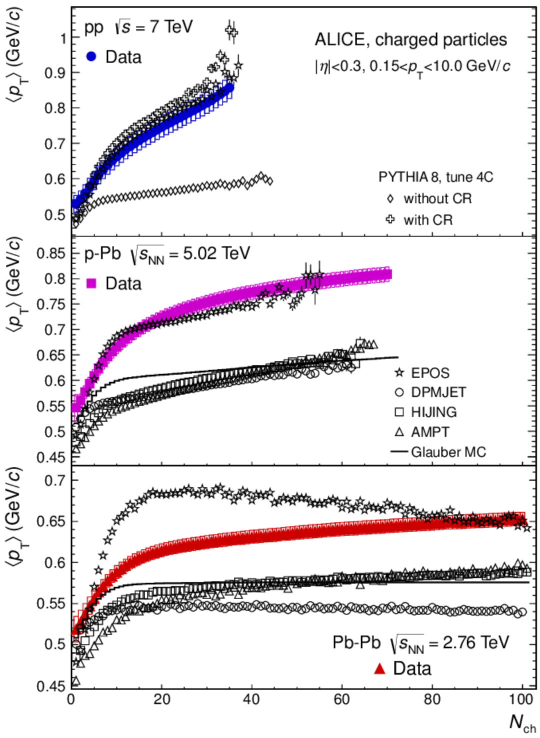


Figure 24: The average transverse momentum $\langle p_T \rangle$ as a function of charged-particle multiplicity N_{ch} measured in pp (upper panel), p–Pb (middle panel), and Pb–Pb (lower panel) collisions, compared with model calculations. The data are compared with predictions from the DPMJET, HIJING, AMPT, and EPOS Monte Carlo event generators. For pp collisions, calculations using PYTHIA 8 [42] with tune 4C are shown, both with and without the color reconnection (CR) mechanism. The lines represent calculations based on a Glauber Monte Carlo approach. Figure adapted from [12].

To validate the experimental results, we performed simulations using various event generators and compared them with ALICE data. In Figure 24, the comparisons are made with PYTHIA 8.3 (Tune Monash), CR ON and OFF; EPOS-LHC, FUSION ON and OFF; and EPOS-4, HYDRO ON and OFF.

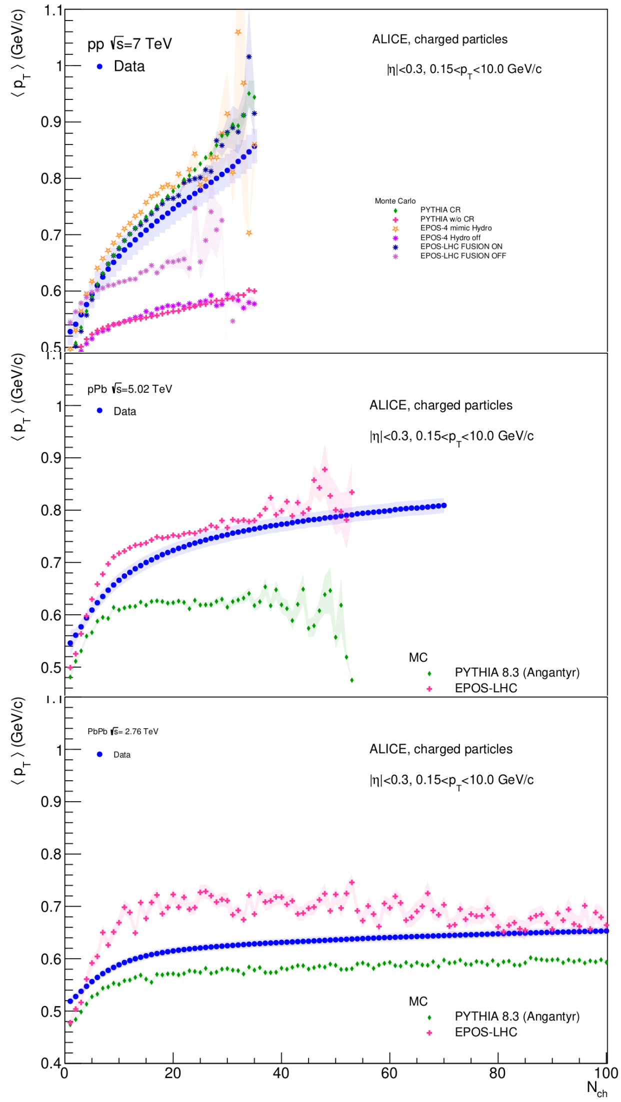


Figure 25: (MY THESIS) Average transverse momentum $\langle p_T \rangle$ as a function of charged-particle multiplicity N_{ch} measured in pp (upper panel), p–Pb (middle panel), and Pb–Pb (lower panel). The data is compared to calculations with PYTHIA 8.3 (Tune Monash), EPOS-LHC and EPOS-4 Monte Carlo event generators.

Our analysis indicates that none of the models accurately describe the experimental data for minimum bias events. However, by applying impact parameter cuts in our simulations, we observed that PYTHIA with CR, within the range $0.35 < b < 0.55$, provides a better fit for pp collisions. The fit improves further when the HARDQCD processes are enabled within the simulation, as shown in Figure 26. This observation holds true across the three different energy levels studied, as illustrated in Figure 27.

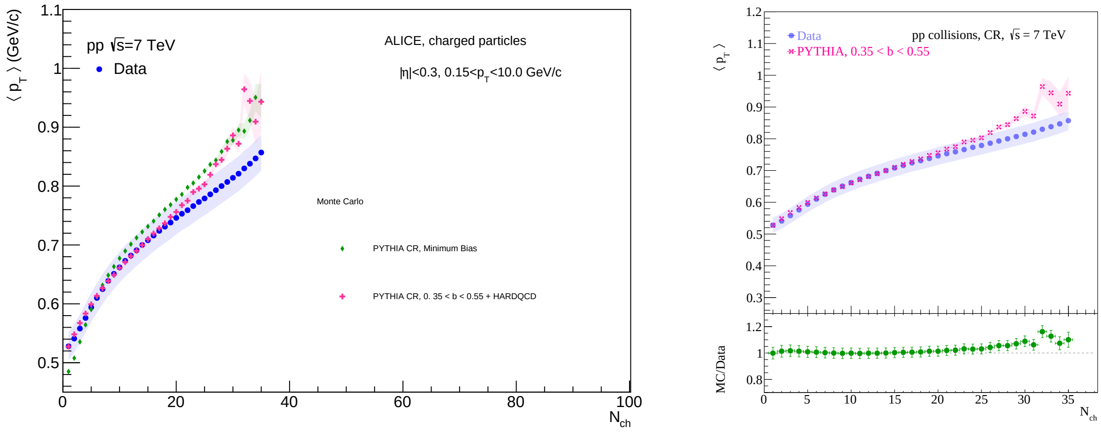


Figure 26: (My Thesis). $\langle p_T \rangle$, as a function of charged-particle multiplicity, N_{ch}, The plots show the impact parameter range $0.35 < b < 0.55$ + HardQCD

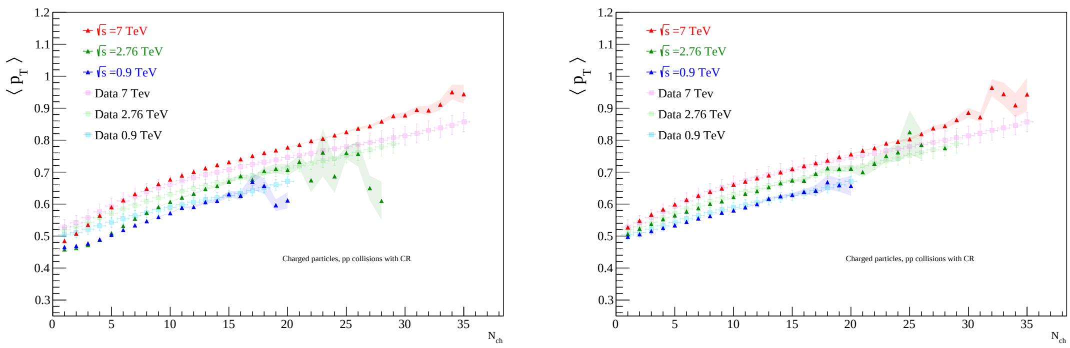


Figure 27: (My Thesis) Comparison across different energy levels. The left panel shows the minimum bias case, while the right panel includes the impact parameter cut.

Furthermore, PYTHIA allows us to analyze the number of multi-parton interactions (nMPI). When plotting the average nMPI against N_{ch}, we observe that for minimum bias events, nMPI ranges from 0 to 13 with a multiplicity from 0 to 20. In contrast, with the impact parameter cut, nMPI ranges from 4 to 12 for the same multiplicity, as shown in Figure 42 (a).

Finally, a cost function analysis was performed to quantify the similarity between Monte Carlo simulations and experimental data. The results indicate a better fit at lower energy levels, as depicted in Figure 42 (b).

In 2022, ALICE extended its analysis by publishing the average transverse momentum $\langle p_T \rangle$ as a function of charged-particle multiplicity N_{ch} across various energies.FIG [18]. This study was conducted within the kinematic range of $|\eta| < 0.8$ and $0.15 < p_T < 10$ GeV/c, focusing on inclusive primary charged particles (defined as charged particles with a mean proper lifetime τ greater than 1 cm/c).

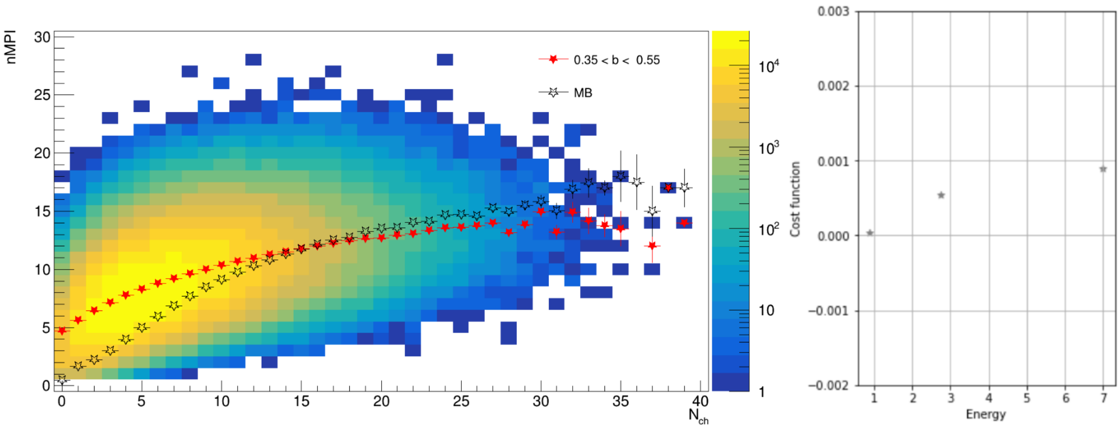


Figure 28: (MY THESIS) (a) Average number of multi-parton interactions (nMPI) as a function of charged-particle multiplicity N_{ch} for minimum bias (MB) and with the b-cut. (b) Cost function analysis between Monte Carlo simulations and experimental data, showing improved fits at lower energies.

We now examine how well PYTHIA and EPOS4 models align with this data across different energy levels.

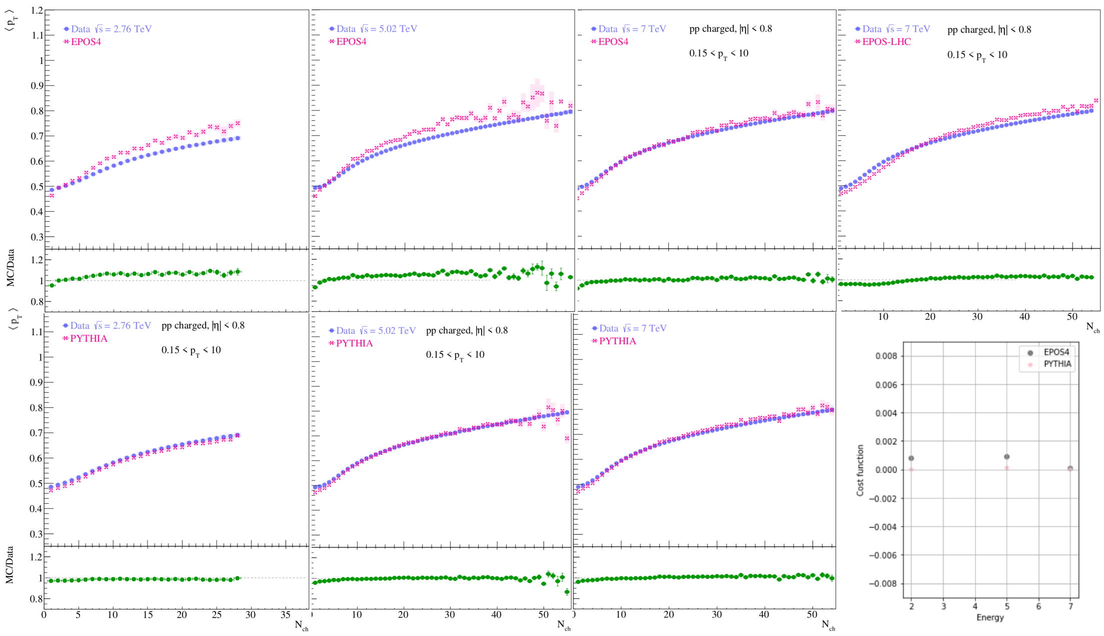


Figure 29: (My Thesis) Comparative analysis of transverse momentum distributions for various collision energies and simulation models. The cost function is shown in the bottom right corner.

As illustrated by the cost function plot, the fit improves with increasing energy. A significant change in the slope of $\langle p_T \rangle$ as a function of multiplicity may also hint at the onset of the QCD phase transition, where hadronic matter transitions into a QGP state. This transition is associated with increased strangeness production and other signatures that are correlated with $\langle p_T \rangle$

So now we know, applying impact parameter cuts and incorporating features like color reconnection (CR) and hard QCD processes improved the agreement between simulations and data, particularly for pp collisions.

The analysis demonstrated the significance of multi-parton interactions (nMPI) in describing collision dynamics. A clear distinction in nMPI distributions was observed when comparing minimum bias events to those with impact parameter cuts, emphasizing the sensitivity of nMPI to initial collision conditions.

A cost-function-based comparison quantified the fit quality of Monte Carlo models to experimental data, showing better agreement at lower energy levels. The extended ALICE analysis in 2022 reaffirmed the correlation between $\langle p_T \rangle$ and N_{ch}, highlighting the universality of these observations across different collision energies.

These findings underscore the importance of detailed modeling and simulation improvements to accurately describe the complex nature of particle production and interactions in high-energy collisions. Further studies and refinements in theoretical frameworks are essential to bridge the gap between experimental data and theoretical predictions.

Chapter 8: Enhanced Production of Multi-Strange Hadrons

The production of strange hadrons in high-energy hadronic collisions provides critical insights into the dynamics of quantum chromodynamics (QCD) under extreme conditions. Unlike up and down quarks, which are present as valence quarks in the colliding protons, strange quarks must be produced during the collision process. This production and the subsequent formation of strange hadrons serve as a direct probe of QCD in environments of high energy density and temperature, such as those created in heavy-ion collisions.

One of the significant findings in recent years is the observed enhancement in the yields of strange and multi-strange hadrons relative to pions as a function of event charged-particle multiplicity . This trend suggests that in high-multiplicity events, where a greater number of particles are produced, the conditions become more favorable for strange quark production. This is likely due to the increased energy density and more frequent parton interactions in these events, which facilitate the formation of strange quark pairs.

Understanding this relationship between strange hadron production and event multiplicity is crucial for several reasons. First, it sheds light on the thermalization process within the quark-gluon plasma (QGP), a state of matter believed to have existed shortly after the Big Bang. Additionally, it provides insights into the degree of chemical equilibration achieved in these collisions, which is essential for mapping the QCD phase diagram and understanding the transition between hadronic matter and the QGP.

Recent observations by the ALICE Collaboration, as shown in Figure 30 (LEFT), have confirmed the enhancement in the yield ratio of strange and multi-strange hadrons to charged pions with increasing multiplicity at mid-rapidity. However, replicating these results using standard Monte Carlo (MC) generators has proven challenging.

In Figure 30 (RIGHT), we present a more accurate fit achieved by fine-tuning the selection of impact parameter ranges. Using the EPOS-LHC MC generator with FUSION ON, we identified the optimal impact parameter range for each particle species, doing this with cost fucntions and other statistical tests, leading to a better agreement with the experimental data.

Furthermore, we explored various configurations for the minimum bias (MB) plots, including comparisons between FUSION ON and OFF, EPOS 4 with HYDRO ON and OFF, and PYTHIA with and without color reconnection (CR). These configurations are presented in Figure 31, pro-

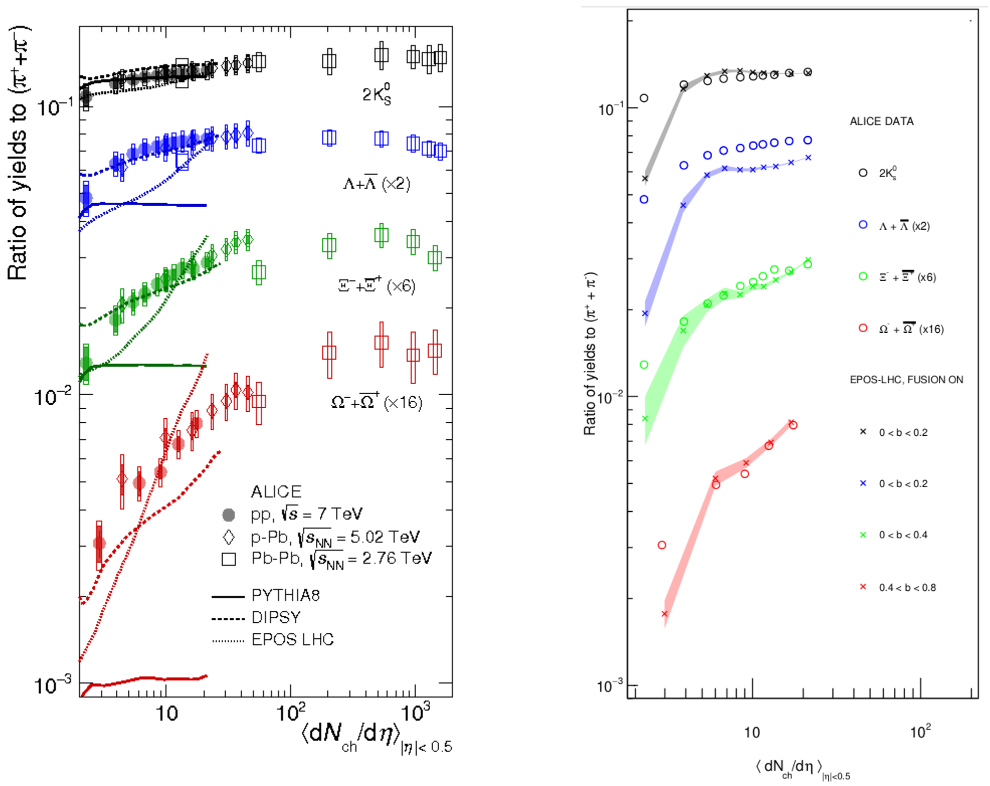


Figure 30: (Left) Yield ratio of strange and multi-strange hadrons to charged pions as a function of multiplicity, [13]. (Right) MY THESIS. Improved fit with specific impact parameter ranges.

viding a comprehensive overview of the different approaches and their respective accuracies.

To further investigate the models' performance, we compared three different models—EPOS-LHC, EPOS 4, and PYTHIA—within a single analysis, as depicted in Figure 42 (LEFT). Additionally, PYTHIA's ability to provide the average number of multiple parton interactions (nMPI) offers another layer of insight. Figure 42 (RIGHT) illustrates how nMPI varies across different impact parameter ranges, offering a deeper understanding of the interaction dynamics in these collisions.

Through the use of Monte Carlo (MC) generators such as EPOS-LHC, EPOS 4, and PYTHIA, it has been possible to model and compare the observed trends. However, challenges remain in replicating the experimental data due to the complexity of tuning parameters such as the impact parameter and the inclusion of features like color reconnection (CR) and hydrodynamic effects. The exploration of various configurations, including FUSION ON/OFF and HYDRO ON/OFF, has provided valuable insights into these dynamics.

Furthermore, the study of multiple parton interactions (nMPI) as a function of multiplicity, as well as their dependence on impact parameter ranges, has deepened our understanding of the

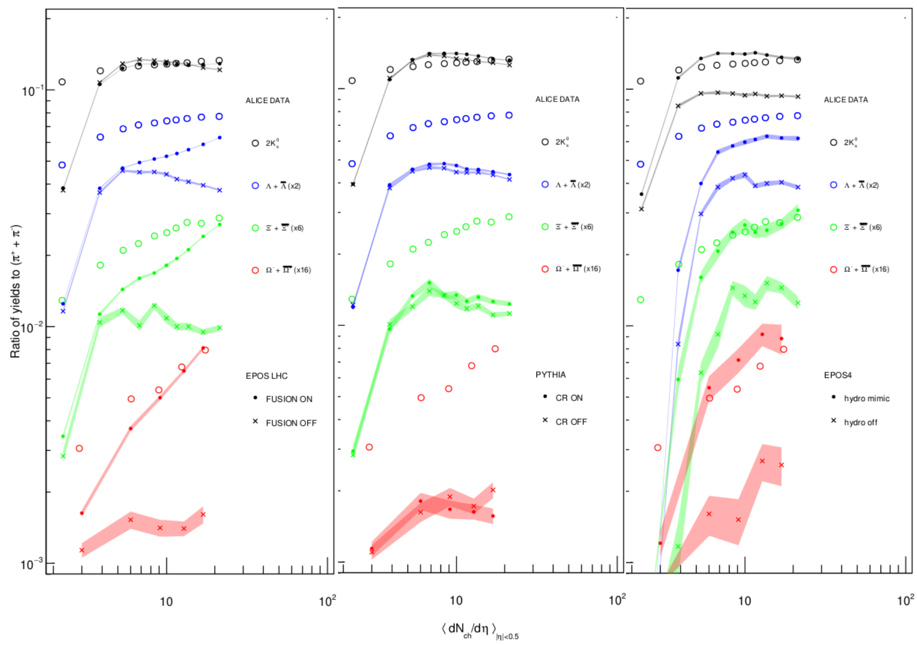


Figure 31: Comparison of various Monte Carlo generator configurations: EPOS-LHC with FUSION ON/OFF, EPOS 4 with HYDRO ON/OFF, and PYTHIA with and without color reconnection (CR). MY THESIS.

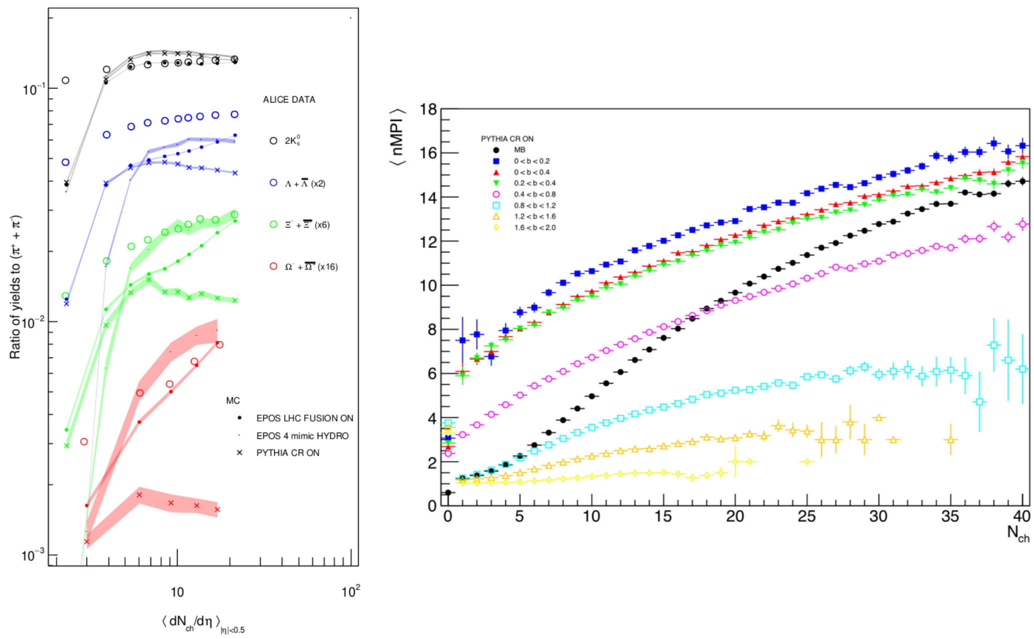


Figure 32: (Left) Comparison of three models: EPOS-LHC, EPOS 4, and PYTHIA. (Right) Average number of multiple parton interactions (nMPI) provided by PYTHIA for different impact parameter ranges. MY THESIS.

interaction dynamics in these events. PYTHIA, in particular, has demonstrated the ability to provide detailed insights into nMPI distributions, offering another layer of understanding.

Overall, the enhanced production of strange and multi-strange hadrons serves as a vital tool for probing QCD processes under extreme conditions. Advancements in MC generators and machine learning techniques will continue to play a crucial role in accurately modeling these yields, enabling a deeper understanding of particle physics and the fundamental forces shaping the universe.

Chapter 9: Determination of the Impact Parameter in pp collisions using ML from the charged-particle multiplicity

Charged particle multiplicity provides information about the dynamics of collisions and the properties of the produced medium. Its distribution is sensitive to initial conditions, such as the impact parameter (b).

In recent years, evidence of possible collective effects produced in systems created in proton-proton collisions has been observed. This work aims to use Monte Carlo models, such as Pythia, to study the properties of charged particle multiplicity distributions obtained for proton-proton collision systems with the ALICE-LHC experiment at CERN. These results will be compared with estimates from the Glauber-type model that incorporates a proton profile based on quark and gluon distributions that we already mentioned in this work, potentially explaining collective effects in small systems at LHC energies. This proposal will explore the determination of the impact parameter for proton-proton collisions, derived from the Monte Carlo framework.

In recent years, machine learning (ML) has emerged as a powerful tool in numerous scientific fields, particularly in high-energy physics (HEP). At the LHC, machine learning models are increasingly used to analyze vast datasets and uncover patterns that would be otherwise difficult to discern through traditional analytical techniques. One such application is the estimation of impact parameters in proton-proton collisions, which plays a key role in determining the spatial geometry of the interaction region and provides insights into the collective behavior of produced particles.

The ALICE experiment at the LHC is dedicated to studying heavy-ion collisions and the quark-gluon plasma, but it also offers valuable data from proton-proton collisions. Charged particle multiplicity, a direct observable from these collisions, is sensitive to the initial state geometry and thus to the impact parameter. In this work, we aim to extract the impact parameter using machine learning algorithms applied to simulated ALICE data.

Through a combination of Monte Carlo simulations using PYTHIA 8.3 and machine learning models, we predict charged particle multiplicities for different impact parameter ranges. This approach represents an innovative method for probing the initial conditions of collisions, potentially leading to deeper insights into the structure and behavior of quantum chromodynamics (QCD) matter at high energies.

9.1 Charged Particle Multiplicity and Impact Parameter

Charged particle multiplicity provides information about the dynamics of collisions and the properties of the produced medium. Its distribution is sensitive to initial conditions, such as the impact parameter (b).

In recent years, evidence of possible collective effects produced in systems created in proton-proton collisions has been observed. This work aims to use Monte Carlo models, such as PYTHIA, to study the properties of charged particle multiplicity distributions obtained for proton-proton collision systems with the ALICE-LHC experiment at CERN. These results will be compared with estimates from the Glauber-type model that incorporates a proton profile based on quark and gluon distributions, potentially explaining collective effects in small systems at LHC energies. This proposal will explore the determination of the impact parameter for proton-proton collisions, derived from the Monte Carlo framework. To map experimental data (charged-particle multiplicity) to a physical property (impact parameter) we use 2 models. FIG. 33

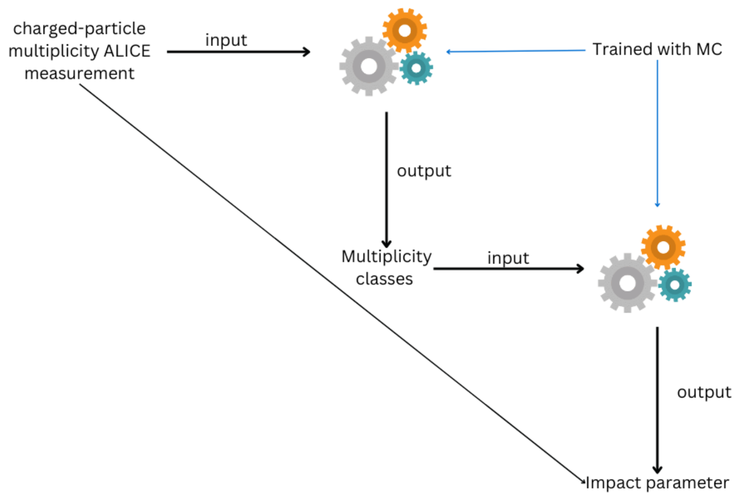
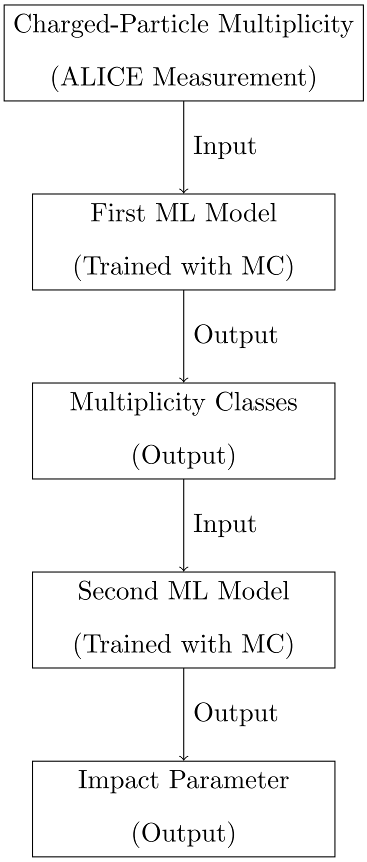


Figure 33: Scheme of how to use ML to extract our impact parameter from charged particle multiplicity



The machine learning pipeline is designed to extract the **impact parameter** from **charged-particle multiplicity measurements** in the ALICE experiment. The process involves the following steps:

1. **Input: Charged-Particle Multiplicity Measurement**

The experimental data from ALICE, specifically the charged-particle multiplicity, serves as the input to the pipeline.

2. **First ML Model: Mapping to Multiplicity Classes**

A machine learning model is trained using Monte Carlo (MC) simulations to classify the events into *multiplicity classes* based on charged-particle multiplicity.

3. **Output: Multiplicity Classes**

The output of the first model is the event's multiplicity class, which serves as an intermediate step for further processing.

4. **Second ML Model: Mapping to Impact Parameter**

A second machine learning model, also trained with MC simulations, takes the multiplicity class as input and outputs the *impact parameter*, a key measure of the collision geometry.

First we need to predict the multiplicity classes from ALICE published data, we employed two algorithms to model the relationship between the real data and corresponding multiplicity for the different impact parameter ranges in TABLE 7, the data used for training these models was generated using Pythia 8.3 tune Monash.

ALICE DATA

ALICE conducted a study of the multiplicity distribution for primary charged particles at a center-of-mass energy of $\sqrt{s} = 7$ TeV [19]. This includes prompt particles generated directly in the collision and their decay products, excluding those from weak decays of strange particles. The results were compared with previous measurements at $\sqrt{s} = 0.9$ TeV and $\sqrt{s} = 2.36$ TeV, focusing on the central pseudorapidity region ($|\eta| < 1$).

At $\sqrt{s} = 7$ TeV, due to the lack of experimental information on diffractive processes, the results were not normalized to event classes used in earlier publications (inelastic and non-single-diffractive events). Instead, a selection criterion was applied requiring at least one charged particle in $|\eta| < 1$ (denoted as INEL $> 0_{|\eta|<1}$), reducing the model dependence in corrections, so for now we are going to train our machine with simulations that meet this event class criteria and that is going to be with a 7 TeV energy. Data acquisition involved the Silicon Pixel Detector (SPD) and the VZERO counters. The SPD consists of two cylindrical layers at radii of 3.9 cm and 7.6 cm, covering pseudorapidity ranges of $|\eta| < 2$ and $|\eta| < 1.4$, respectively. The VZERO scintillator hodoscopes, located at $z = 3.3$ m and $z = -0.9$ m on either side of the interaction point, span pseudorapidity regions $2.8 < \eta < 5.1$ and $-3.7 < \eta < -1.7$.

Data were collected with a 0.5 T magnetic field, and the typical bunch intensity for collisions at $\sqrt{s} = 7$ TeV was 1.5×10^{10} protons, yielding a luminosity of approximately 10^{27} cm^{-2}s^{-1}. A single bunch per beam collided at the ALICE interaction point, and the probability of a recorded event containing more than one collision was estimated at 2×10^{-3}. This estimate was verified by identifying events with more than one distinct vertex.

At 7 TeV and 0.9 TeV, triggers required at least one hit in either the SPD or the VZERO counters, ensuring sensitivity to at least one charged particle across eight pseudorapidity units. For 2.36 TeV data, the VZERO counters were inactive, and the trigger required at least one hit in the SPD ($|\eta| < 2$). Events were synchronized with signals from beam pick-up counters on both sides of the interaction region, confirming the passage of proton bunches. Control triggers, used to measure beam-induced and accidental backgrounds, eliminated most backgrounds. The remaining contamination, estimated at 10^{-4} to 10^{-5}, was deemed negligible. And this is the data we are going to use as input to determine the impact parameter.

First, we have to prepare the data that would be used to train the algorithm. Since the plan was to use the already published data from the ALICE collaboration [19], we use that to ensure that the PYTHIA parameters were properly tuned. (FIG. 34)

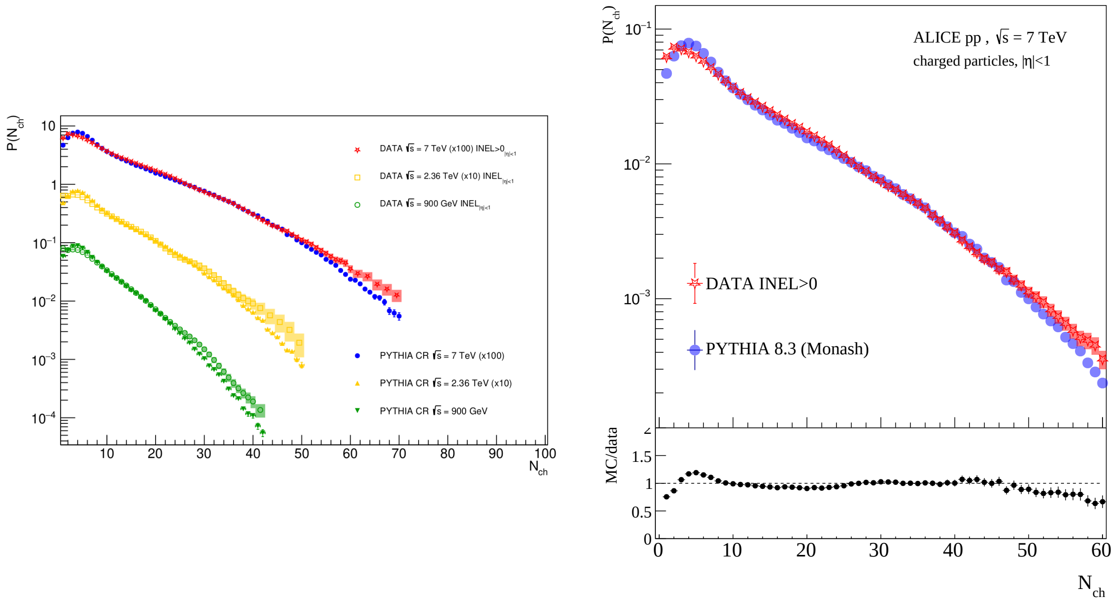


Figure 34: Left: Multiplicity distributions $P(N_{\mathrm{ch}})$ at $\sqrt{s}$ = 900 GeV, 2.36 TeV, and 7 TeV by ALICE (symbols) and compared with PYTHIA. Data points are scaled for visibility.Right: $P(N_{\mathrm{ch}})$ at $\sqrt{s} = 7$ TeV within $|\eta| < 1$, comparing ALICE data (red) with PYTHIA blue). The lower panel shows the ratio of MC to data, indicating agreement and deviations in specific N_{ch} ranges.

We can see that PYTHIA fits well with the ALICE data, with some uncertainty at low and high multiplicities, although this also occurs with Glauber.

	ALICE pp	PYTHIA Monash
	$\sqrt{s} = 0.9$ TeV	
INEL	$3.1005 \pm 0.01^{+0.08}_{-0.05}$	3.054
	$\sqrt{s} = 2.36$ TeV	
INEL	$3.7135 \pm 0.01^{+0.25}_{-0.11}$	3.8405
	$\sqrt{s} = 7$ TeV	
INEL>0	$6.01 \pm 0.01^{+0.20}_{-0.12}$	6.01

Table 6: Charged-particle pseudorapidity densities measured by ALICE in pseudorapidity region ($|\eta| < 1$),

Extraction of multiplicity classes

The Glauber model is traditionally applied to heavy ion collisions, but in small system using the anisotropic and inhomogeneous density profile of the proton, recent studies ([5]) found that the model reproduces the charged particle multiplicity distribution of p + p collisions at LHC energies very well. Collision geometric properties can also be determined, they use a model with fluctuating proton orientation and it has three effective quarks and gluonic flux tubes connecting them. To estimate the charged-particle multiplicity in their study, for p + p collisions nucleons are replaced by partons (quarks and gluons). The model provides b, Nch, $Npart$ and $Ncoll$, for an event with a given impact parameter and collision energy.

Multiplicity %	$b - range(fm)$
0-1	0 - 0.25534
1-5	0.25535 - 0.46909
5-10	0.46909 - 0.58484
10-15	0.58484 - 0.66430
15-20	0.66431 - 0.72766
20-30	0.72767 - 0.83026
30-40	0.83027 - 0.91819
40-50	0.91820 - 1.00117
50-70	1.00118 - 1.17163
70-100	1.17164 - 2.54998

Table 7: b-ranges for each multiplicity class at $\sqrt{s} = 7$ TeV. Taken from [5]: table I.

.

Depending on the impact parameter range, we also extracted the charged multiplicity histograms for those within a specific b - range from PYTHIA.(FIG 35).

9.2 Model Training and Validation

To estimate the charged particle multiplicity and relate it to the impact parameter, we trained three machine learning models: a Deep Neural Network (DNN), a Boosted Decision Tree (BDT) using XGBoost, and a Random Forest. All models take the minimum bias (MB) charged-particle multiplicity as input and predict the charged-particle multiplicity distribution within different impact parameter ranges.

The dataset was split into training, validation, and test subsets, with 70% of the data used for training and 30% reserved for testing. Within the training set, an additional validation split of 20% was used during training to monitor model performance and fine-tune hyperparameters, such as the learning rate, number of layers, tree depth, and regularization parameters.

Model validation involved cross-validation techniques to ensure generalizability and to avoid overfitting, particularly for the BDT and Random Forest, where hyperparameter tuning can heavily influence performance. For the DNN, early stopping based on validation loss was employed to prevent overfitting and achieve optimal training convergence.

Performance was evaluated using metrics such as mean squared error (MSE) for regression

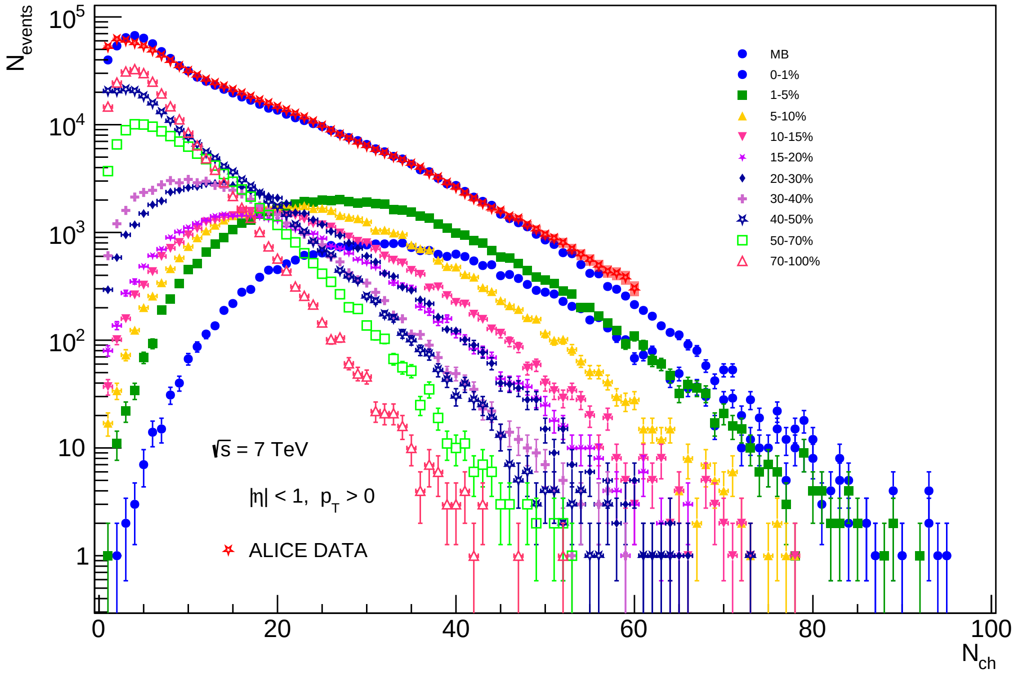


Figure 35: Pythia Tune Monash multiplicity distribution and real ALICE data for pp collisions at $\sqrt{s} = 7\,\text{TeV}$. Each color represents a different centrality range (impact parameter), from head-on to glancing collisions. Red stars show ALICE data, scaled for easy comparison.

tasks and feature importance rankings were analyzed for interpretability. This rigorous validation process ensured that the models accurately captured the relationship between minimum bias multiplicity and impact parameter-dependent distributions, providing reliable predictions across diverse scenarios.

9.3 Random Forest Model

Random Forest was selected for its robustness and ensemble approach, particularly suited for this task due to its capability to handle complex interactions between features without extensive preprocessing. By constructing multiple decision trees based on randomly selected data and features, the Random Forest achieved a balanced trade-off between bias and variance.

The Random Forest model also underwent tuning of hyperparameters, such as the number of trees and maximum depth, optimizing performance based on cross-validation scores. With an accuracy of Y, the Random Forest provided comparable or slightly superior predictive power to the DNN for certain impact parameter ranges, particularly in the low and high multiplicity regions.

9.4 Boosted Decision Tree

The Boosted Decision Tree (BDT) model implemented using the XGBoost framework was included in this analysis for its efficiency and enhanced performance in handling structured data. XGBoost, short for eXtreme Gradient Boosting, builds upon traditional boosting algorithms by incorporating regularization techniques to prevent overfitting and optimize performance.

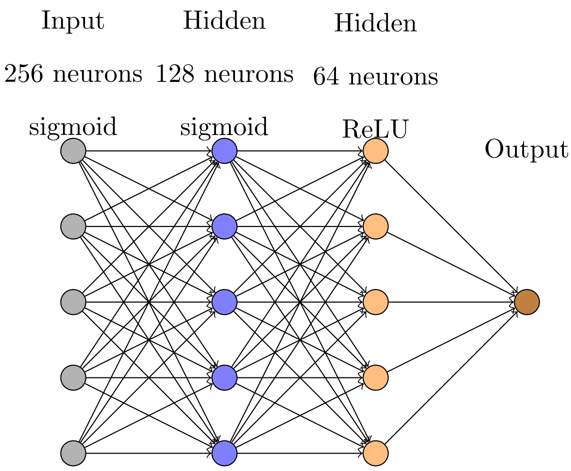
The model combines a series of weak learners, typically decision trees, into a strong predictive ensemble through iterative gradient boosting. This method focuses on minimizing the loss function at each step while adaptively correcting errors from previous iterations. XGBoost's support for parallel processing and advanced tree-pruning techniques further enhances its computational efficiency.

To tailor the model to our dataset, hyperparameters such as the learning rate, maximum tree depth, number of boosting rounds, and the regularization parameters (α and λ) were fine-tuned through cross-validation. The use of early stopping criteria ensured that overfitting was minimized while maintaining high predictive accuracy.

Performance evaluation revealed that the XGBoost model excelled in regions with complex multiplicity patterns, particularly in mid-range impact parameter distributions, where both the DNN and Random Forest models faced challenges. With a final accuracy of Z, XGBoost demonstrated its robustness and effectiveness, complementing the insights gained from the DNN and Random Forest models and contributing to a comprehensive understanding of the multiplicity distributions.

9.5 Deep Neural Network (DNN)

The Deep Neural Network (DNN) model was chosen for its capacity to learn complex, nonlinear relationships between the minimum bias multiplicity and the charged-particle multiplicity in specific impact parameter ranges. The architecture of the DNN was implemented using the TensorFlow/Keras framework and comprised four fully connected layers:



- The input layer consisted of 256 neurons with a sigmoid activation function, taking the feature set as input.

- Two hidden layers with 128 and 64 neurons, respectively, used sigmoid and ReLU activation functions. These layers allowed the model to capture intricate patterns in the data.

- The output layer matched the dimensionality of the input features, predicting the charged-particle multiplicity distribution.

The model was compiled using the Adam optimizer with a learning rate of 0.001 and the mean squared error (MSE) as the loss function. Dropout layers were not included in this specific implementation, as regularization was achieved through careful tuning of the architecture and training parameters.

Training and Validation: The training dataset was split into 80% for training and 20% for validation to monitor model performance during training. The model was trained for 200 epochs with a batch size of 32, ensuring efficient gradient updates. During training, the validation loss was used to assess overfitting or underfitting, and early stopping could be applied if needed.

Evaluation: After training, the model's predictions were compared with the test dataset. Predictions (y_{pred}) and ground truth values (y_{test}) were flattened to avoid shape mismatches during evaluation. Key metrics for evaluation included:

- **R-Squared (R^2) Score:** $R^2 = 1 - \frac{\text{SS}_{\text{res}}}{\text{SS}_{\text{tot}}}$, indicating the proportion of variance explained by the model. $R^2 = 0.9842362094574908$

The results demonstrated a low MSE and a high R^2 score, indicating the model's effectiveness in capturing the relationship between minimum bias multiplicity and charged-particle multiplicity distributions.

The training history, including loss and validation loss, was plotted to visualize the convergence of the model. This ensured that the learning rate and epoch count were optimal for the dataset, resulting in a DNN that performed robustly across a wide range of impact parameters.

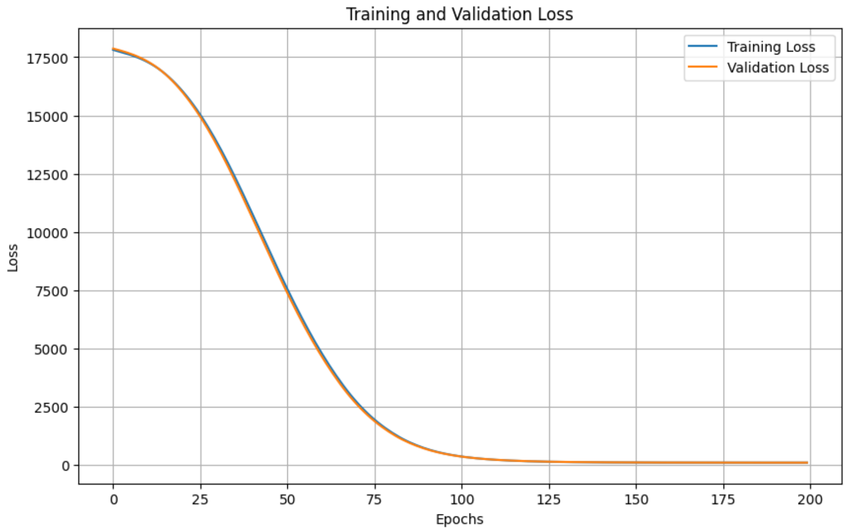


Figure 36: Training and Validation Loss vs. Epochs: This plot illustrates the convergence of training and validation loss over 200 epochs, showing stable learning as both losses decrease and plateau with no significant overfitting

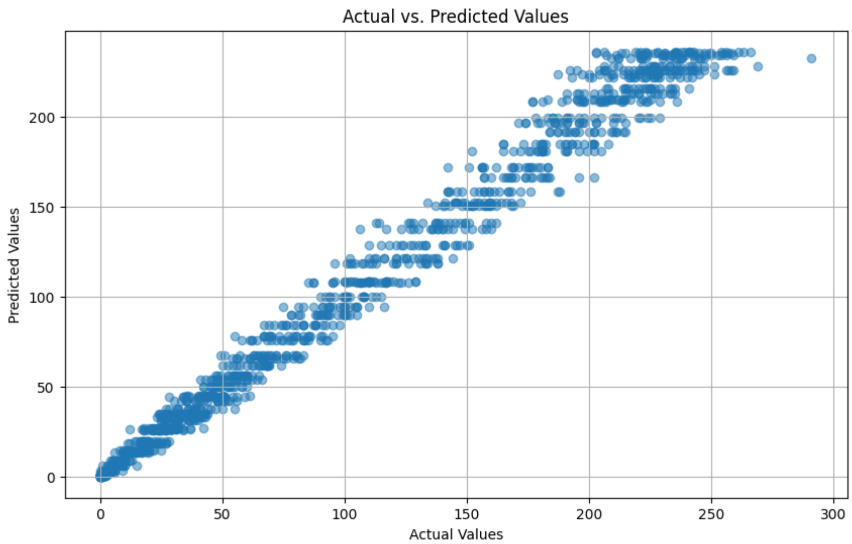


Figure 37: Scatter plot of Actual vs. Predicted Values: A visual comparison of the model's predictions, showing a strong alignment along the diagonal

The scatter plot of actual vs. predicted values demonstrates that the model achieves strong predictive performance, with most data points clustering near the diagonal line. This alignment indicates that the model effectively captures the relationship between the features and the target variable, supported by a reasonable $R^2 = 0.98$.

The training and validation loss plot further validates the model's effectiveness. Both losses decrease steadily over epochs and converge to a stable minimum without signs of overfitting. This suggests that the model generalizes well to unseen data while maintaining a balance between underfitting and overfitting.

Together, these results highlight a well-trained model with accurate predictions and robust generalization capabilities.

9.6 Model Comparison

To evaluate the performance of each model in estimating the impact parameter, we assessed the Deep Neural Network (DNN), Random Forest, and Boosted Decision Tree (BDT) using XGBoost against the actual impact parameter ranges obtained from Glauber model estimates. Key metrics such as mean absolute error (MAE) and root mean squared error (RMSE) were calculated for each model across different ranges.

The results indicated that all three models performed well, with slight differences in accuracy depending on the impact parameter range. The DNN demonstrated robust performance across the entire range, while the Random Forest excelled in the low and high multiplicity regions. The BDT, leveraging its adaptive boosting mechanism, provided the most accurate predictions in the mid-range multiplicity regions, making it a complementary approach to the DNN and Random Forest.

The comparison, as illustrated in Figure 38, highlights the strengths and limitations of each model, showcasing their combined effectiveness in accurately estimating impact parameters based on charged-particle multiplicity.

- **Low Multiplicity Range**: Random Forest showed a slightly lower error in estimating smaller impact parameter values, attributed to its ability to handle the low-multiplicity distributions effectively.

- **High Multiplicity Range**: The DNN performed well in the high-multiplicity regime due to its capacity for capturing complex patterns in larger datasets.

To evaluate the performance of each model in predicting the charged-particle multiplicity distributions (N_{ch}), we compared the Deep Neural Network (DNN), Random Forest, and Boosted Decision Tree (BDT) using XGBoost. The comparison focused on two key error metrics: the mean error ($N_{\mathrm{ch,\ true}} - N_{\mathrm{ch,\ pred}}$) and the mean absolute error ($|N_{\mathrm{ch,\ true}} - N_{\mathrm{ch,\ pred}}|$), as shown in Figure 39.

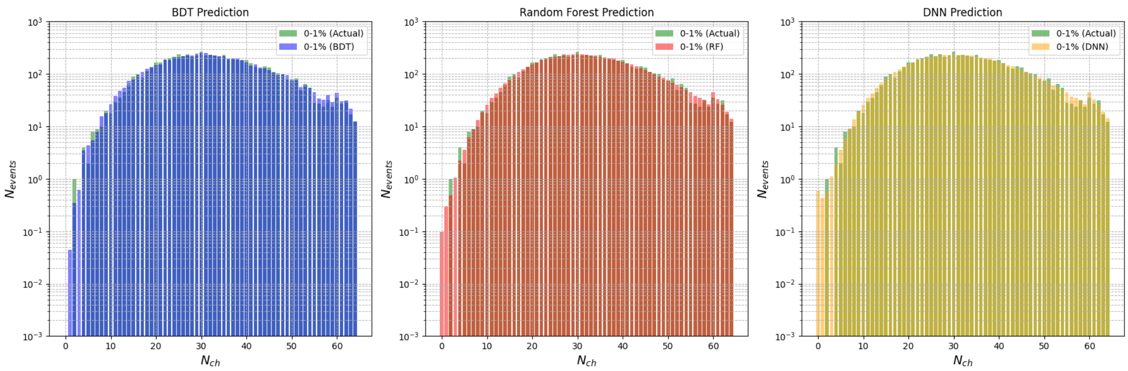


Figure 38: Comparison of actual and predicted charged-particle multiplicity distributions (N_{ch}) in the 0-1% centrality range for three machine learning models: Random Forest (left), Boosted Decision Tree (middle), and Deep Neural Network (right). The green histograms represent the actual multiplicity, while the colored overlays correspond to the predictions from each model. The logarithmic scale highlights the models' ability to capture the multiplicity distribution across a wide range of values.

The top panel of Figure 39 illustrates the mean errors across the N_{ch} range. All models showed consistent performance, with errors fluctuating around zero, indicating unbiased predictions. However, the Random Forest and BDT provided slightly more stable results at higher multiplicity values compared to the DNN.

The bottom panel highlights the mean absolute errors. The DNN and Random Forest achieved lower errors at low multiplicity values, while the BDT demonstrated superior performance in the mid-to-high multiplicity ranges. This indicates that the BDT's adaptive boosting approach effectively handles the variability in the mid-range, where both the DNN and Random Forest faced challenges.

Overall, the results confirm the complementary strengths of the three models. The DNN's ability to capture nonlinear relationships, the Random Forest's robustness, and the BDT's precision in mid-range multiplicities collectively provide a comprehensive approach to estimating charged-particle multiplicity distributions.

9.7 Model Prediction Evaluation

Extending this analysis to various centrality ranges, we can obtain multiplicity distribution classes, as shown in Figure 40. These distributions are derived from ALICE data using our ML-based predictions. For reasons discussed in the following sections, we ultimately selected the DNN model for our final results.

Through this analysis, we observe that machine learning provides a reliable method for estimating the impact parameter, particularly valuable in high-energy physics experiments where

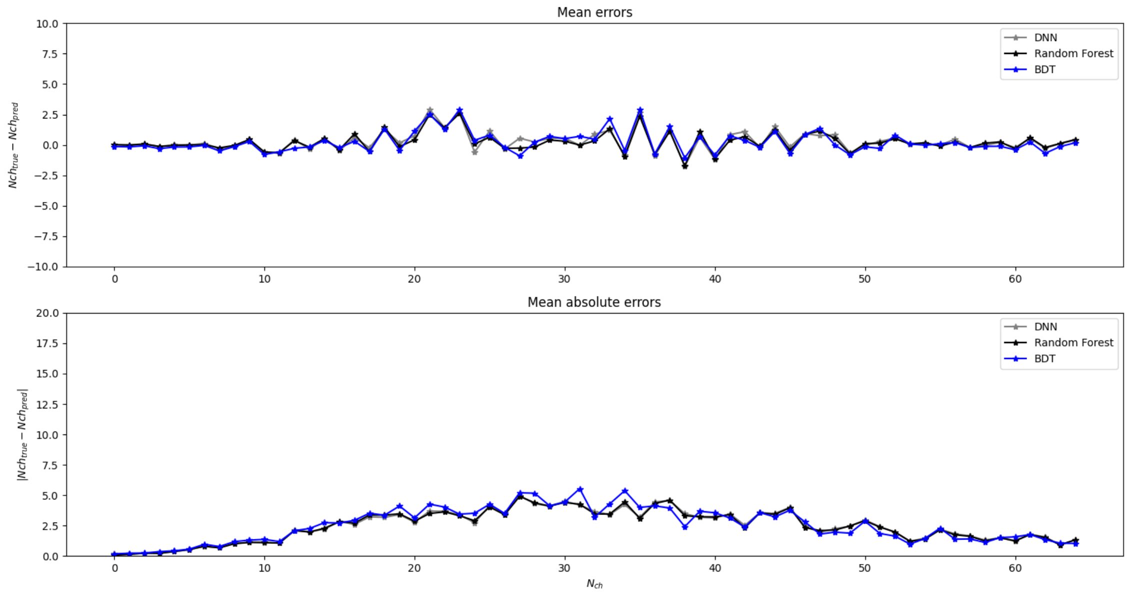


Figure 39: Comparison of mean errors (top) and mean absolute errors (bottom) for three machine learning models: Deep Neural Network (DNN), Random Forest, and Boosted Decision Tree (BDT), evaluated across varying values of N_b.

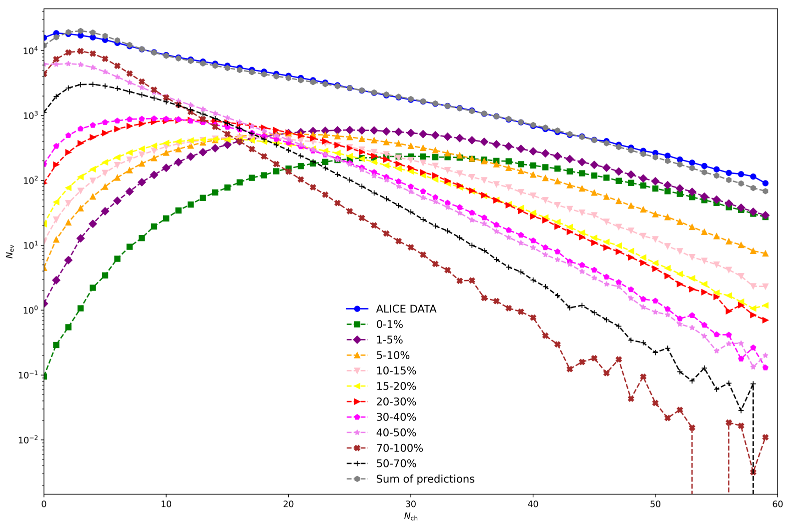


Figure 40: Predicted multiplicity classes across centrality ranges, based on ALICE data and ML model predictions.

traditional analytical methods may fall short. Future directions involve extending this approach to different collision energies and examining the potential of hybrid models that combine the strengths of both DNN and Random Forest for improved accuracy.

9.8 Impact Parameter Profile

With our multiplicity classes established, we can now apply the same methodology to extract the impact parameter profile. For each multiplicity class, we derive its impact parameter distribution, and by summing all individual profiles, we obtain the full impact parameter profile—a direct prediction of our machine learning (ML) model based on ALICE multiplicity data.

In our simulations, we generated both the multiplicity and its corresponding impact parameter distribution. Using this data, we applied DNN to predict the impact parameters. By combining the predictions for each class, we constructed a comprehensive impact parameter profile from the ALICE multiplicity data. The resulting profile is illustrated in Figure 41.

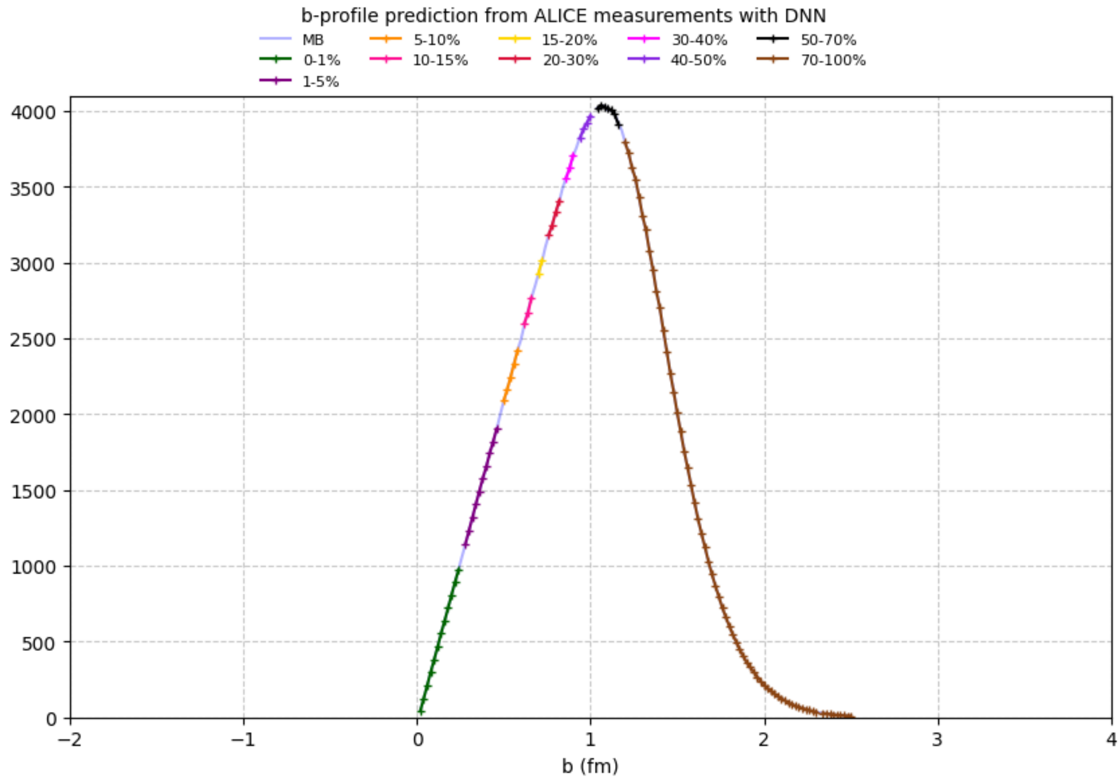


Figure 41: Complete impact parameter profile synthesized from all multiplicity class predictions.

The extraction of b in proton-proton collisions via machine learning has proven to be a successful approach in this study. By employing ML algorithms on simulated data generated with PYTHIA 8.3, we have demonstrated the capability of machine learning techniques to predict multiplicity classes associated with various impact parameter ranges. These results not only enhance our understanding of proton-proton collisions in the ALICE experiment but also underscore the growing significance of machine learning in high-energy physics research.

As machine learning techniques continue to evolve, their applications in particle physics will likely expand, enabling more sophisticated analyses of experimental data. Future work can focus on refining these models by incorporating real collision data and expanding the scope of investigation to include heavy-ion collisions. Ultimately, this methodology opens new possibilities for understanding the complex dynamics of particle interactions at the LHC.

9.9 CNN for nMPI extraction

Using the same ALICE multiplicity distribution for primary charged particles at a center-of-mass energy of $\sqrt{s} = 7$ TeV [19] for INEL>0, we extracted the number of multi-parton interactions using PYTHIA 8.3 Tune monash pp simulations at that energy to train a CNN with FIG. 42 architecture. Being a regression task, this CNN aims at constructing a mapping between the multiplicity and the impact parameter of an event. So, it is appropriate to choose the mean squared error (MSE) as the loss function.

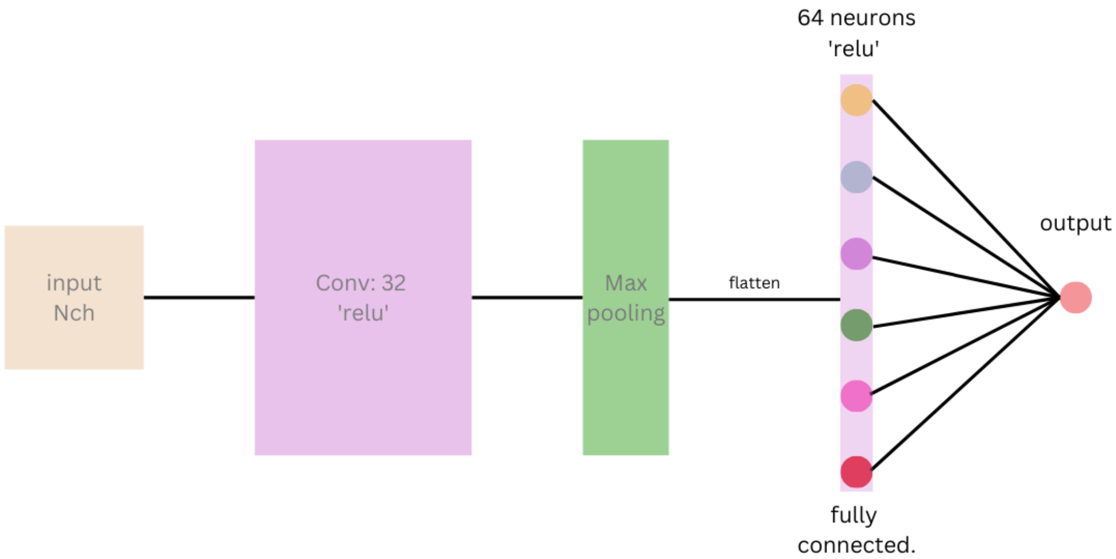


Figure 42: Architecture of the neural network, showing the input layer, a convolutional layer with 32 filters using ReLU activation, max pooling for dimensionality reduction, followed by a fully connected layer with 64 neurons and ReLU activation, leading to the output layer.

In FIG. 43 we can see our CNN prediction once we plug in the ALICE measurments for N_{ch} INEL>0, given that we already have our multiplicity classes per centrality percentage from the previous study, we do the CNN over again for each b-range and use the multiplicity classes predicted before as input, once we do this for the 10 classes we have figure 44

From the analysis, it is evident that the CNN model, trained using PYTHIA 8.3 simulations, successfully predicts $\langle nMPI \rangle$ values across multiplicity classes when applied to the ALICE N_{ch} measurements. The agreement between CNN predictions and MC simulations demonstrates the capability of neural networks to map non-linear relationships between event multiplicity and the number of multi-parton interactions. Additionally, the weighted averages of $\langle nMPI \rangle$ highlight slight differences between the two methods, providing insights into the CNN's interpretation of the data.

This study underscores the potential of machine learning in complementing traditional Monte

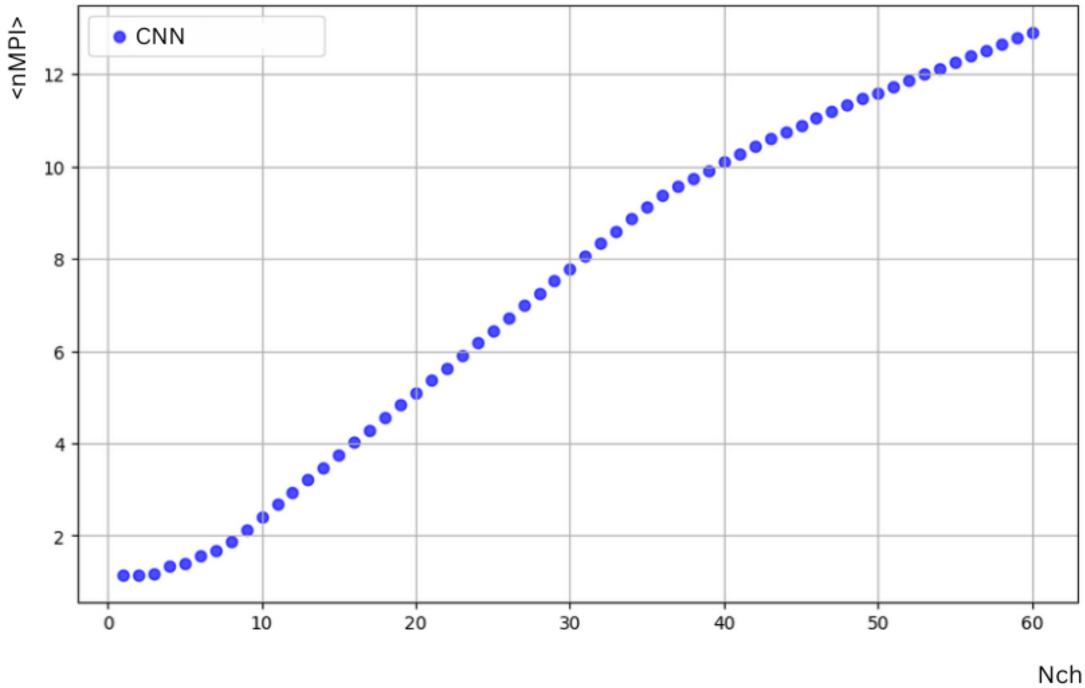


Figure 43: $\langle nMPI \rangle$ vs multiplicity predicted with the CNN trained with MC

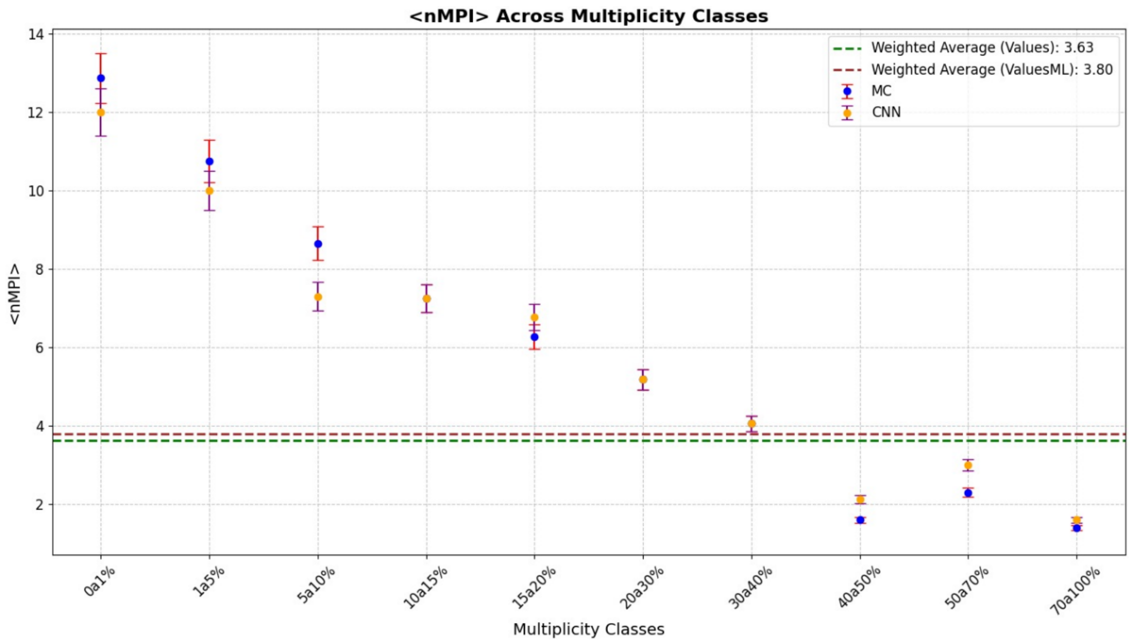


Figure 44: Comparison of $\langle nMPI \rangle$ Across Multiplicity Classes: The plot shows the mean number of multiple parton interactions ($\langle nMPI \rangle$) across various multiplicity classes. Data points represent Monte Carlo (MC) predictions and Convolutional Neural Network (CNN) results, with associated error bars. Weighted averages for MC and CNN values are represented by dashed green and red lines, respectively.

Carlo-based analyses, offering a framework for future studies involving impact parameter estimations and event-level observables in high-energy physics.

Chapter 10: Conclusions

The multiplicity of charged particles is intrinsically connected to the impact parameter, a fundamental quantity in the study of high-energy collisions. This relationship has been estimated theoretically and explored experimentally through various analyses. In this work, we have provided an estimation of this relationship using advanced machine learning techniques, allowing for a more accurate and detailed understanding. Each range of the impact parameter corresponds to an average number of charged particles ($\langle N_{ch} \rangle$) and the mean number of multiple parton interactions ($nMPI$), as summarized in Table 8 below.

Multiplicity (%)	**b - range (fm)**	$\langle N_{ch} \rangle$	$\langle nMPI \rangle$
0-1	0 - 0.25534	32.32	12.1
1-5	0.25535 - 0.46909	27.97	10
5-10	0.46909 - 0.58484	23.26	7.3
10-15	0.58484 - 0.66430	20.01	7.25
15-20	0.66431 - 0.72766	17.66	6.77
20-30	0.72767 - 0.83026	15	5.2
30-40	0.83027 - 0.91819	12.25	4
40-50	0.91820 - 1.00117	6.25	2.12
50-70	1.00118 - 1.17163	7.72	3
70-100	1.17164 - 2.54998	4.93	1.6

Table 8: Impact parameter ranges, corresponding $\langle N_{ch} \rangle$ and $\langle nMPI \rangle$.

From our analysis, we have observed that the average transverse momentum ($\langle p_T \rangle$) exhibits a strong dependence on event multiplicity, which in turn can be better described by specific impact parameter ranges. This dependency highlights the intricate interplay between initial collision geometry and the resulting final-state observables.

Furthermore, the enhancement of strange and multi-strange hadrons in high-multiplicity events corroborates the idea that high-density environments foster the production of strange quarks. This finding reinforces the importance of studying strangeness as a probe for quantum chromodynamics (QCD) processes under extreme conditions. By examining these phenomena, we gain deeper insights into the dynamics of quark-gluon plasma (QGP) formation and the transition between hadronic matter and the QGP.

Our study also explored the role of multiple parton interactions (nMPI) in shaping the event characteristics. The variations in nMPI across impact parameter ranges provide critical information about the density and energy dependence of the collision system. These observations are key to refining theoretical models and enhancing the predictive power of Monte Carlo (MC) generators.

In conclusion, this work has demonstrated the utility of combining experimental data with machine learning techniques to elucidate the relationship between impact parameters, multiplicity, and event dynamics. Future studies should continue to leverage these approaches, integrating more complex features to further unravel the rich phenomenology of high-energy collisions.

Chapter 11: Appendix A: Additional ML application

11.1 Background Subtraction with DNN

11.1.1 Muon-ID Prototype for ALICE 3 upgrade

Cosmic rays provide a natural source of high-energy particles that can be used to calibrate detectors. By studying cosmic ray interactions, the response of the detector to high-energy particles can be understood and optimized.

Cosmic rays are high-energy particles that originate from outer space and travel through the universe at nearly the speed of light. They consist mostly of protons and a smaller fraction of heavier nuclei, such as helium and other elements. Cosmic rays can have energies ranging from a few million electron volts (MeV) to beyond 10^{20} electron volts (eV).

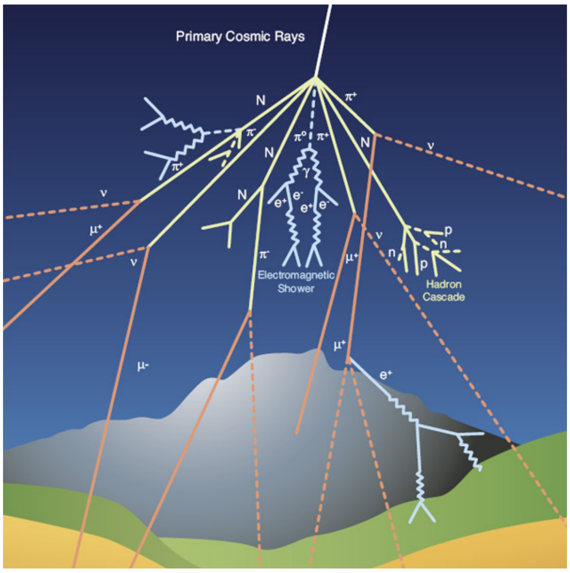


Figure 45: Scheeme of typical cosmic rays shower in the earth atmosphere. © CERN / Photographer: Lapka, Marzena

When cosmic rays enter the Earth's atmosphere, they collide with molecules, primarily nitrogen and oxygen, producing a cascade of secondary particles, including pions, muons, electrons, and gamma rays. These secondary particles can penetrate deep into the atmosphere and even reach the Earth's surface.

Plastic scintillators are widely used in the detection of cosmic rays due to their efficiency, durability, and ability to cover large areas. A scintillator is a material that emits light (scintillates) when charged particles pass through it.

When a charged particle from cosmic rays passes through the plastic, it excites the molecules within the scintillator material. The excited molecules then return to their ground state by emitting photons in the visible spectrum.

The emitted light is collected by photodetectors, such as photomultiplier tubes (PMTs) or silicon photomultipliers (SiPMs), attached to the scintillator. These photodetectors convert the light into an electrical signal.

The electrical signals are processed to determine the energy deposited by the cosmic ray particle, its time of arrival, and the location where the interaction occurred. The energy and time information can be used to reconstruct the original cosmic ray particle's energy and trajectory.

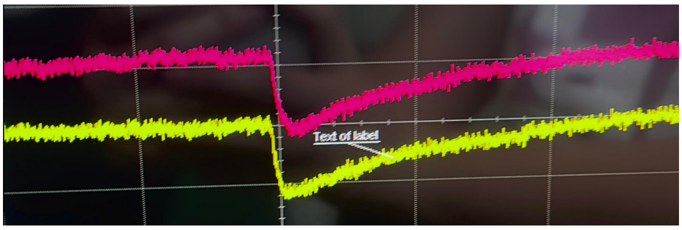


Figure 46: Display of a pulse generated by a cosmic muon in an oscilloscope.

To improve the accuracy and reduce background noise, multiple scintillators can be arranged in an array. A cosmic ray event is typically identified when coincident signals are detected in two or more scintillators simultaneously. This coincidence method helps in distinguishing cosmic ray events from other types of background radiation.

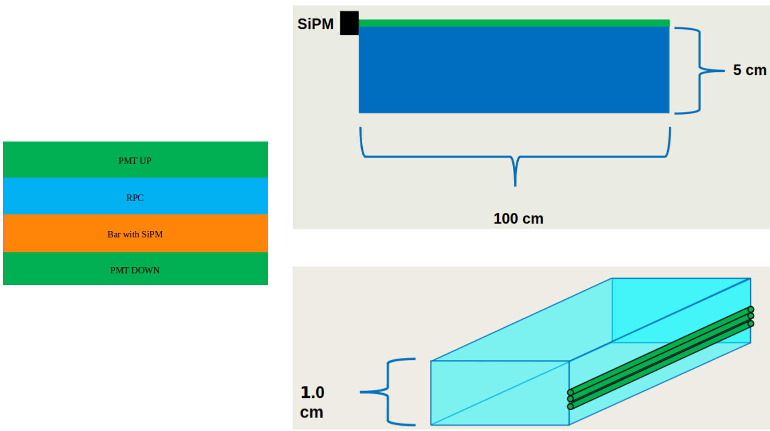


Figure 47: Schemes of the experimental setup (left), the scintillator bar geometry (not to scale) with the WLS fibres attached on the side (right) © Figures by Yael Vasquez (BUAP)

That's what we did in the Ecocampus Puebla particle's lab, where we used plastic scintillator bars to test and calibrate our prototype (FIG. 47) for the next MUON-ID detector for the ALICE experiment at CERN. By exposing the plastic bars to cosmic rays, we were able to characterize the detector's response to high-energy muons, a key component of cosmic rays. This testing allowed us to fine-tune the detector's sensitivity and ensure that it would be capable of accurately identifying and tracking muons in the high-energy environment of the ALICE experiment.

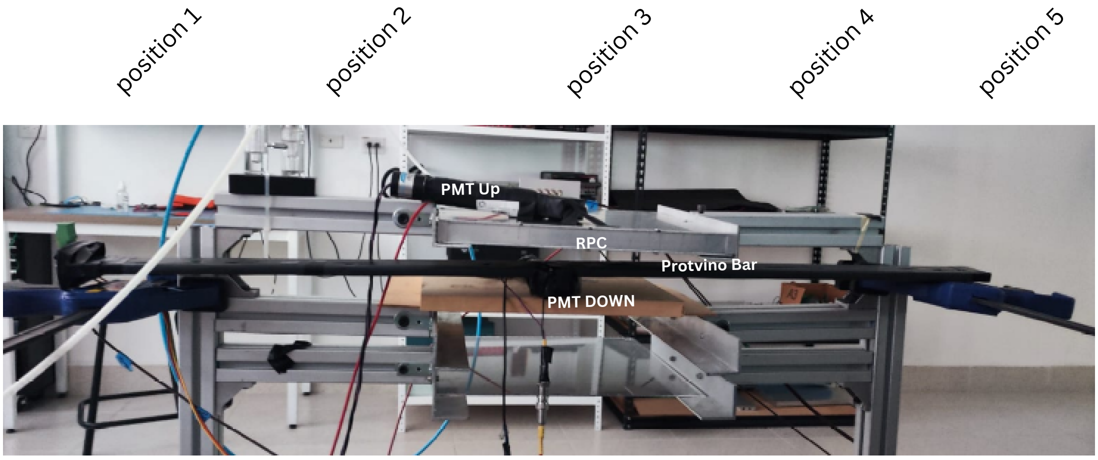


Figure 48: Picture of the cosmic ray data taking setup showing the different positions of the triggers.

We also employed machine learning techniques to reduce noise in the signals detected by the plastic scintillator bars. By applying advanced algorithms, we were able to distinguish between true cosmic ray events and background noise more effectively.

11.1.2 Signal vs Background

In the realm of particle physics, detectors serve as the primary interface between theoretical predictions and experimental observations. These detectors are designed to measure the properties of subatomic particles generated in high-energy collisions, such as those occurring in the LHC. However, the accuracy and precision of these measurements are often compromised by noise—unwanted signals that can obscure or mimic the genuine signals of interest. Effective noise reduction is therefore a critical component of detector design and data analysis in high-energy physics experiments.

In the analysis phase, data cleaning techniques are employed to remove noise and background events from the dataset. These techniques often involve the application of thresholds, outlier rejection, and statistical background subtraction methods.

Machine learning techniques, in particular, have emerged as powerful tools for enhancing noise reduction, complementing traditional methods and significantly improving the quality of the data obtained. I'll now explain how we managed to do this.

Data Preparation

The initial step involved processing the raw data to distinguish between signal and noise. The data was organized into two categories: noise and signal. The noise data was extracted by isolating the background from the plastic bars mentioned before, while the signal data was obtained from a controlled environment where all recorded events were confirmed to be valid.

Using the uproot library, I accessed the ROOT files and converted the relevant tree structures into Pandas DataFrames.

Each dataset was labeled accordingly: noise data was assigned a label of 0, and signal data was assigned a label of 1.

These labeled DataFrames were then concatenated into a single dataset.

Feature Selection and Normalization

The machine learning models were trained to distinguish between signal and noise using three key features: Amplitude, time begin, and time end. These features were selected based on their potential relevance to the classification task.

To ensure that the models performed optimally, I normalized the selected features using the StandardScaler from the sklearn library. This step standardizes the features to have a mean of zero and a standard deviation of one, which is essential for models that are sensitive to the scale of the input data.

Training the Random Forest Classifier

The normalized data was split into training and testing sets, with 80% of the data used for training and 20% reserved for testing. I trained a RandomForestClassifier with 100 estimators, balancing the class weights to account for any potential imbalance between the signal and noise classes.

The model was evaluated on the test set, achieving an accuracy of 98.7%. This high level of accuracy indicated that the model was effective at distinguishing between signal and noise.

Neural Network Model

To further enhance the classification performance, I implemented a Neural Network model using the Keras API. The network architecture consisted of an input layer, two hidden layers, and an output layer:

Input Layer: This layer receives the normalized feature set. Hidden Layers: The first hidden layer comprises 64 neurons, while the second has 32 neurons, both using the ReLU activation function. Output Layer: A single neuron with a sigmoid activation function was used for binary classification. The model was compiled using the Adam optimizer and binary crossentropy as the loss function. I trained the network for 20 epochs with a batch size of 64, using a validation split of 20% to monitor the model's performance during training. Upon evaluation, the model achieved an accuracy of 98.61% on the test set, confirming its robustness in classifying signal and noise.

Application to New Data

To evaluate the robustness of our trained Neural Network model, we applied it to a new dataset collected from the plastic bars setup previously discussed (see FIG. 48). This dataset included data from all the positions of the bars, ensuring a comprehensive test of the model's ability to generalize to new and varied inputs.

The new data underwent the same preprocessing steps as the training data, including feature selection and normalization. These steps were crucial for ensuring that the input data was in the correct format and scale for the Neural Network to process effectively. Once the data was prepared, the trained model was used to classify each event in the dataset as either noise or signal.

For classification, we applied a threshold of 0.5 to the model's output probabilities. Events with predicted values below this threshold were classified as noise (0), while those with values above 0.5 were classified as signal (1). This binary classification allowed us to effectively separate the true signal from the background noise.

The results, shown in FIG. 49, demonstrate a significant improvement in signal clarity after applying the Neural Network model. The model effectively filtered out a substantial amount of noise, providing a much cleaner dataset for subsequent analysis.

With the noise reduced and the signal enhanced, we can now expect improved results in further analyses, such as time resolution and the relationship between time and charge. The cleaned signal provides a more accurate basis for these investigations, potentially leading to better insights and more reliable data interpretations.

To illustrate the potential improvements, we examine the time vs. charge relationship, as shown in FIG. 50. The cleaner signal allows for more precise plotting and analysis, which is essential for understanding the underlying physics of the events.

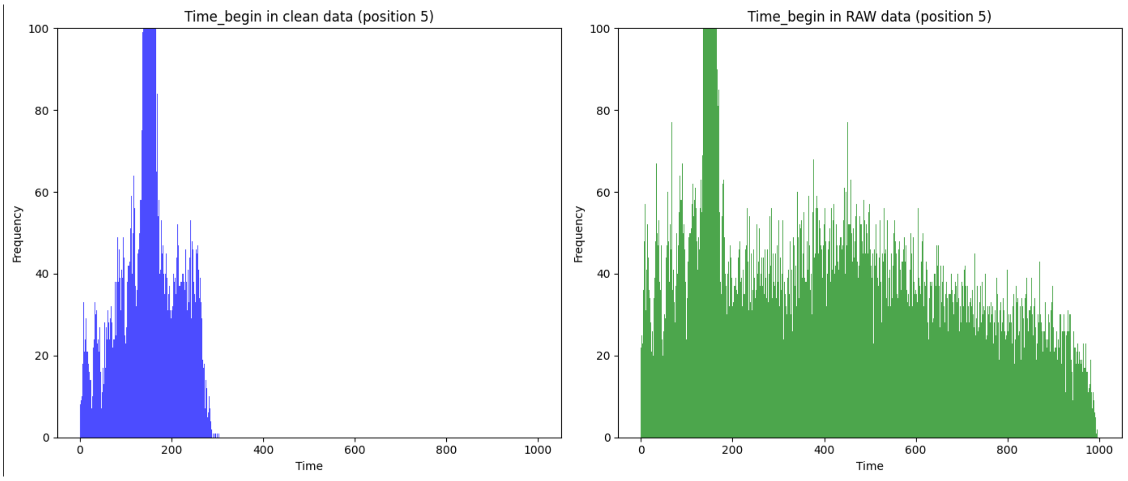


Figure 49: (MY THESIS). Signal cleaning results using the trained Neural Network model. The figure illustrates the significant reduction in noise, leading to a clearer signal for further analysis.

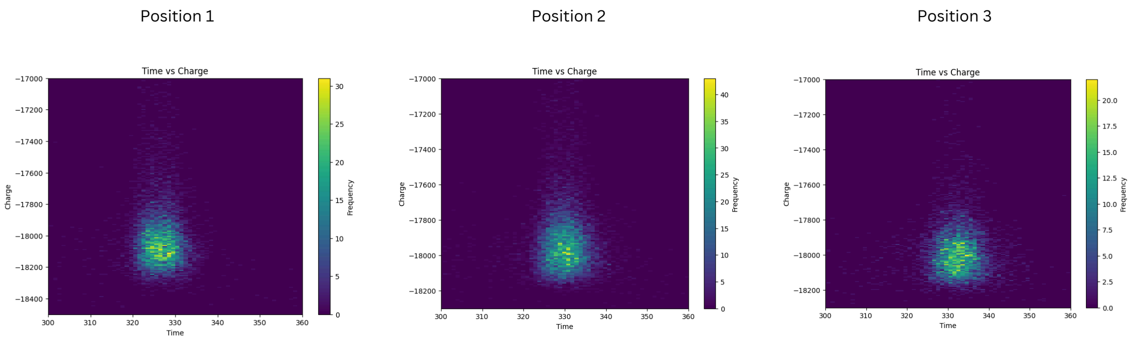


Figure 50: (MY THESIS). Improved time vs. charge plot using the cleaned signal data. The clearer signal results in more accurate and reliable data analysis.

Time resolution

To determine the time resolution as [20],the following parameters for the sensors are defined:

- The time resolution of the PMT Up is represented by σ_1,

- The time resolution of the PMT Down is denoted by σ_2,

- The time resolution of the SiPM is indicated by σ_3,

- The variance of the time difference between the PMT Up and PMT Down events is given by σ_{12},

- The variance of the time difference between the PMT Up and SiPM events is denoted by σ_{13},
- The variance of the time difference between the PMT Down and SiPM events is represented by σ_{23}.

The variance for correlated random variables can be expressed as:

$$\sigma_{12}^2 = \sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2$$

where ρ is the correlation. In this case, the correlation between the variables of each detector is zero, because there is no relationship between them. Therefore, the variance of correlated random variables is simplified to

$$\sigma_{12}^2 = \sigma_1^2 + \sigma_2^2$$

Applying the same procedure to each pair of variables:

$$\sigma_{13}^2 = \sigma_1^2 + \sigma_3^2$$

$$\sigma_{23}^2 = \sigma_2^2 + \sigma_3^2$$

So we have a system of equations, solving for the SiPM detector time resolution we have:

$$\sigma = \sqrt{\frac{\sigma_{12}^2 + \sigma_{13}^2 - \sigma_{23}^2}{2}} \tag{26}$$

The variance of the time difference is obtain from a gaussian fit to each one, for example the one in FIG. 51

After obtaining the variance for each position, and computing in eq. (25), we obtain (FIG. 52) for one of our bars:

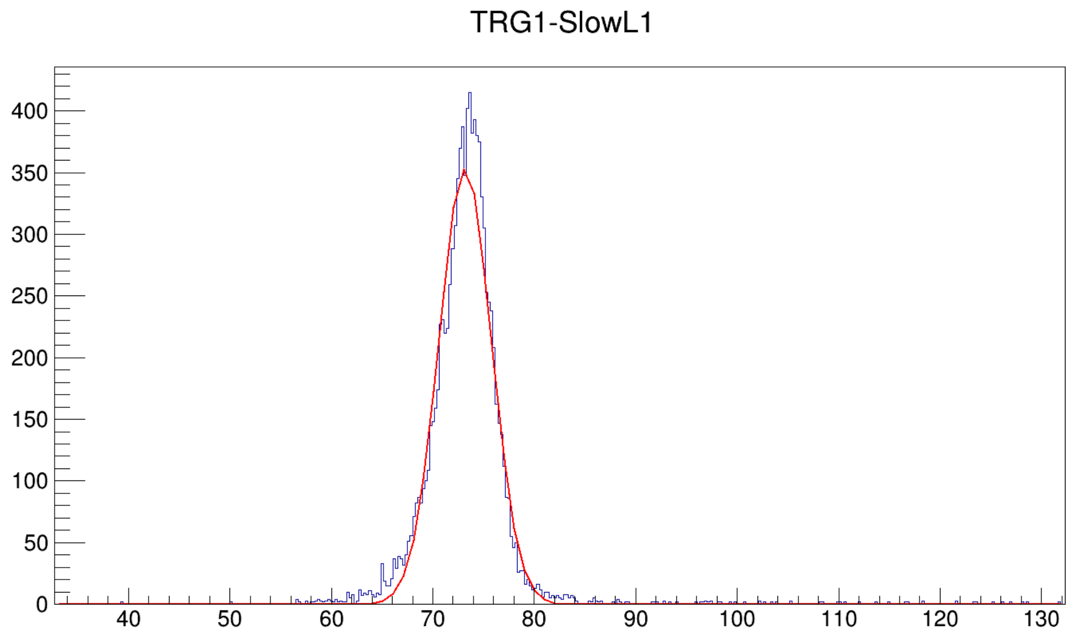


Figure 51: Example of how time resolution is obtained

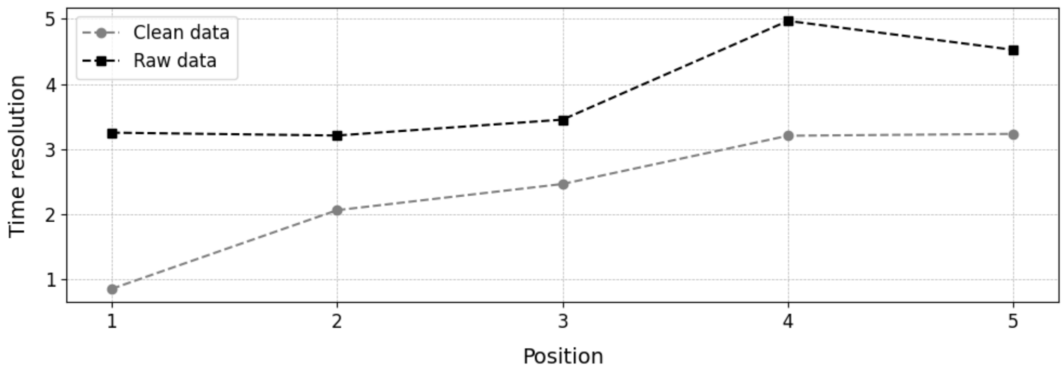


Figure 52: Time resolution according to position

The performance of the neural network classifier was evaluated using the Receiver Operating Characteristic (ROC) curve, the confusion matrix, and training history plots, as shown in Figures 53, 54 and 55. The ROC curve in Figure 53 demonstrates the model's high discriminatory power, with an area under the curve (AUC) of 1.00, indicating near-perfect classification performance. This suggests that the model is highly effective at distinguishing between the two classes, achieving a high true positive rate while maintaining a low false positive rate. The confusion matrix in Figure 54 further illustrates the model's accuracy, with 13,686 true negatives, 2,086 true positives, and minimal misclassifications (57 false positives and 162 false negatives).

In addition to these evaluation metrics, the model's learning process was monitored using training and validation loss and accuracy over epochs, as shown in Figure 55. The loss plot shows a steady decrease in both training and validation loss, indicating that the model effectively

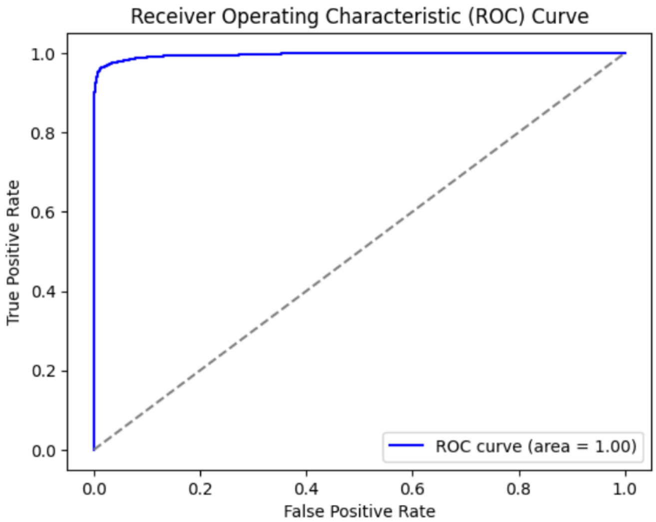


Figure 53: ROC curve showing the model's classification performance with an AUC of 1.00.

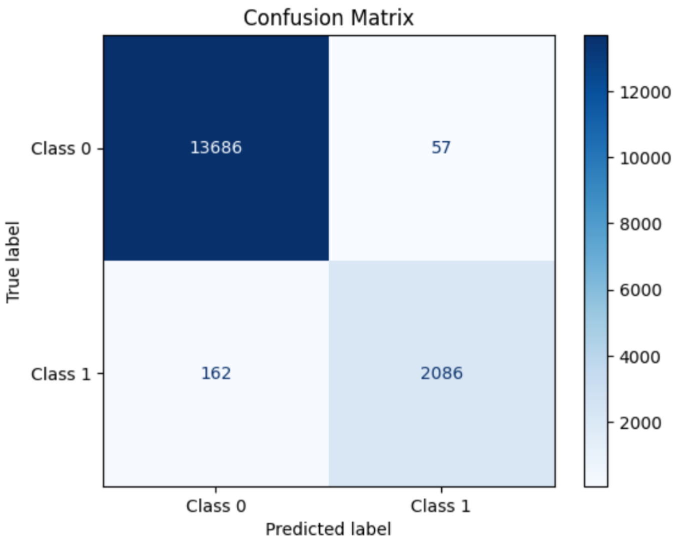


Figure 54: Confusion matrix illustrating the counts of true positives, true negatives, false positives, and false negatives.

minimized the error during training without overfitting. Similarly, the accuracy plot illustrates an increase in both training and validation accuracy over epochs, reaching a high, stable level, with validation accuracy closely tracking training accuracy. These trends confirm that the model generalizes well on unseen data, demonstrating both stability and robustness throughout the training process.

The application of machine learning techniques has proven to be a powerful tool for enhancing signal clarity and improving data analysis. By employing a trained Neural Network model, we were able to effectively distinguish between noise and signal in a new dataset collected from our plastic bars setup. This preprocessing step significantly improved the quality of the data, leading

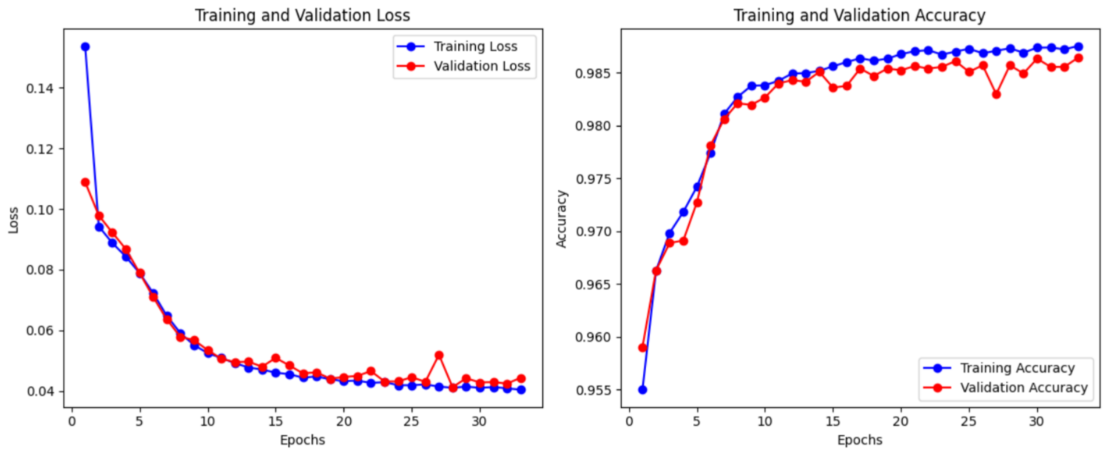


Figure 55: (LEFT) Training and validation loss over epochs, showing the model's ability to minimize error during training. (RIGHT) Training and validation accuracy over epochs, indicating the model's learning progress and generalization ability.

to more precise measurements and analyses.

The successful classification of events as either noise or signal underscores the robustness of the machine learning model, demonstrating its ability to generalize beyond the training data. This capability is crucial for handling the complex and variable nature of experimental data in particle physics, where signal purity is essential for accurate interpretation.

Moreover, the cleaner signal obtained through this process allowed us to achieve better results in subsequent analyses, such as time resolution and time vs. charge relationships. These improvements highlight the potential of machine learning to enhance the understanding of collision dynamics and the properties of the quark-gluon plasma.

In conclusion, the integration of machine learning into the analysis pipeline of high-energy physics experiments not only enhances data quality but also opens new avenues for exploring and understanding the fundamental processes governing particle interactions. As we continue to refine these techniques and apply them to larger and more diverse datasets, we can expect even greater insights into the underlying physics driving these complex systems.

References

[1] ALICE Collaboration. The alice experiment – a journey through qcd, 2022.

[2] F. BECATTINI. *QUARK GLUON PLASMA AND RELATIVISTIC HEAVY ION COLLISIONS*, pages 53–71. IOP Publishing, 2014.

[3] Roel Aaij et al. Centrality determination in heavy-ion collisions with the LHCb detector. *JINST*, 17(05):P05009, 2022.

[4] Stanislaw D. Glazek and Patryk Kubiczek. Proton structure in high-energy high-multiplicity p-p collisions. *Few Body Syst.*, 57(6):425–430, 2016.

[5] Suman Deb, Golam Sarwar, Dhananjaya Thakur, Pavish Subramani, Raghunath Sahoo, and Jan-e Alam. Glauber model for a small system using the anisotropic and inhomogeneous density profile of a proton. *Phys. Rev. D*, 101(1):014004, 2020.

[6] Fabian Kuger. *Signal Formation Processes in Micromegas Detectors and Quality Control for large size Detector Construction for the ATLAS New Small Wheel*. PhD thesis, U. Wurzburg (main), 2017.

[7] Shreyasi Acharya et al. ALICE upgrades during the LHC Long Shutdown 2. *JINST*, 19(05):P05062, 2024.

[8] Centrality determination in heavy ion collisions. 8 2018.

[9] Pythia. Pythia: A program for the generation of high-energy physics events, n.d. Accessed: 2024-08-14.

[10] T. Sjöstrand and M. Utheim. A framework for hadronic rescattering in pp collisions. *Eur. Phys. J. C*, 80:907, 2020.

[11] K. Werner. Core-corona procedure and microcanonical hadronization to understand strangeness enhancement in proton-proton and heavy ion collisions in the epos4 framework. *Phys. Rev. C*, 109:014910, Jan 2024.

[12] Betty Bezverkhny Abelev et al. Multiplicity dependence of the average transverse momentum in pp, p-Pb, and Pb-Pb collisions at the LHC. *Phys. Lett. B*, 727:371–380, 2013.

[13] Jaroslav Adam et al. Enhanced production of multi-strange hadrons in high-multiplicity proton-proton collisions. *Nature Phys.*, 13:535–539, 2017.

[14] Matthew D. Schwartz. *Quantum Field Theory and the Standard Model.* Cambridge University Press, Cambridge, UK, 2014.

[15] R. K. Ellis, W. J. Stirling, and B. R. Webber. *QCD and Collider Physics.* Cambridge Monographs on Particle Physics, Nuclear Physics, and Cosmology. Cambridge University Press, Cambridge, UK, 1996.

[16] ALICE Collab. Alice upgrades during the lhc long shutdown 2. *Journal of Instrumentation*, 19(05):P05062, May 2024.

[17] Sweden) Sjostrand, Torbjorn (Lund University. Monte Carlo (Introduction to Monte Carlo techniques in High Energy Physics) (1/2). Monte Carlo (Introduction to Monte Carlo techniques in High Energy Physics) (1/2). 2012.

[18] Shreyasi Acharya et al. Multiplicity dependence of charged-particle production in pp, p-Pb, Xe-Xe and Pb-Pb collisions at the LHC. *Phys. Lett. B*, 845:138110, 2023.

[19] K. Aamodt et al. Charged-particle multiplicity measurement in proton-proton collisions at $\sqrt{s} = 7$ TeV with ALICE at LHC. *Eur. Phys. J. C*, 68:345–354, 2010.

[20] Y. Vásquez Beltrán. Time resolution of particle detectors based on scintillation photons, 2023.