# Machine Learning In Heavy-Ion Collisions

Rodrigo Miguel Chavez Zavaleta, Department of Physics
McGill University, Montreal
August, 2023

A thesis submitted to McGill University in partial fulfillment of the
requirements of the degree of

Master of Science

©Rodrigo Miguel Chavez Zavaleta, 2023

# Dedication

Dedicated to my father.

# Acknowledgements

I would first like to thank my supervisor Prof. Sangyong Jeon, for his guidance above and beyond the scope of this thesis, first as an undergraduate instructor, then as a mentor, and finally as a thesis supervisor. His understanding, patience, and calm have influenced how I approach problems in physics and beyond.

I would also like to thank Prof. Charles Gale, first as an undergraduate instructor, then as a second mentor in our research group.

Chanwook Park and Rouzbeh Modarresi Yazdi also deserve recognition for passing on to me the knowledge, software, scripts, as well as helping me getting up to date with the current version of our group's code, specifically MARTINI, such that I was able to build upon it for my research project.

Thanks are also due to Nicolas Miró Fortier, for having provided an integral amount of data for the Initial Conditions part of this thesis, as well as the productive discussions on how to optimally use the computing resources at our disposal.

I should also acknowledge the Digital Research Alliance of Canada, formerly known as Compute Canada, to provide our research group with the compute resources necessary to carry out our research.

Finally, I should mention my colleagues, Noah Kakekaspan, Bailey Forster, Han Gao, and Da Sen Ye, with whom the experience of graduate school, coursework and research were made all the more pleasant.

# Abstract

Matter under extreme conditions can enter the exotic phase known as quark gluon plasma, where nuclear sub-constituents, quarks and gluons, are unbound. Accessing this state of matter allows us then to study the interactions between these quarks and gluons, namely Quantum Chromo-Dynamics (QCD). Experimentally, quark gluon plasma is created in Heavy-Ion Collisions (HIC), where high energy beams are collided. In this thesis, we simulate Pb-Pb collisions at $\sqrt{s} = 2.76$TeV utilizing Monte-Carlo techniques.

The initial combination of the gluon color gauge fields of both nuclei are required for our simulations, and are a computational challenge which can be solved iteratively. One can start this iteration with taking the continuous analytical solution, within the discretized numerical equation. We propose a Neural network to generate a different Ansatz, training on the solutions from the iteration.

One of the most notable results of HIC's is the production of jets, aggregates of particles with a conic trajectory. While it is believed that no QGP is formed in proton-proton collisions, QGP should be created with larger Nuclei, and interact with jets. As such, one can study the QGP by using the jets as probes. We compare three different physical formalisms of jet evolution in a QGP medium, one of which is implemented using a neural network.

# Abrégé

Exposée a de conditions extrêmes, la matière peut entrer dans un état de plasma de quarks et de gluons, où les composants subatomiques sont déconfinés. Accéder à cet état de la matière nous permet alors d'étudier les interactions entre ces quarks et gluons, autrement dit la chromodynamique quantique. Expérimentalement, le plasma de quarks et de gluons est créé dans des collisions d'ions lourds, ou des faisceaux de particules sont percutés. Dans cette thèse, nous simulons des collisions d'ions de plomb à une énergie de $\sqrt{s} = 2.76 TeV$ en utilisant des techniques Monte-Carlo. La combinaison initiale de champs de jauge de couleur des gluons des deux noyaux est nécessaire pour nos simulations, et est un défi computationnel qui peut être resou iterativement. On peut commencer cette iteration en prenant la solution analytique continue, dans l'iteration numérique discrète. Nous proposons un réseau de neurones pour générer un nouveau point de départ, en utilisant de l'apprentissage automatiques sur des solutions de l'iteration. Un des résultats des collisions d'ions lourds et la production de jets, des agrégats de particules à trajectoire conique. On considère qu'il n'y a pas de création du plasma de quarks et de gluons lors de collisions entre deux protons. Cependant, des collisions d'ions plus larges devraient créer le plasma, et interagiraient avec les jets. De cette façon, étudier les jets permet d'étudier le plasma de quarks et de gluons. Nous comparons trois formalismes de l'évolution de jets dans le plasma de quarks et de gluons

# List of Figures

# List of Tables

# Table of Contents

# Chapter 1

# Introduction

## 1.1 Historical View

Our current understanding of the interactions in nature classifies these in four fundamental forces, namely the gravitational force, the electromagnetic force, and the weak and strong nuclear forces. While the gravitational and electromagnetic forces have been studied for a few centuries, the strong and weak forces have only been known for about a century. This is primarily due to our newfound capacity to probe the two latter forces. In the late 1800's and early 1900's, Rutherford's experiments led to a new atomic model which separated a positively charged nucleus from negatively charged orbiting electrons. A few decades later, the discovery of the neutron by Chadwick led to the discovery of the strong force, since a novel force was needed to explain the attraction between the nucleons, the protons and neutrons, within the nucleus. The explanation to the nucleon's interactions turned out to be the exchange of pions, the first predicted subatomic particle. Many others were then discovered and classified into a collection which was then called a "particle zoo". This "zoo" was greatly simplified when it was found that all those particles were in fact composite, made-up of quarks and gluons. By this simplification, understanding subatomic physics reduces to understanding the interactions of quarks and gluons, namely Quantum Chromodynamics. Under standard temperature and pressure, quarks and gluons exist only as constituents of larger particles. However, they can exist independently in extreme conditions such as the very early universe.

Today, QCD is studied experimentally in modern particle collider under heavy-ion collisions (HIC). Particle colliders produce beams of particles at very high energies and opposite directions, resulting in collisions. These collisions produce, for a brief moment, deconfined nuclear matter, namely quark-gluon plasma (QGP). While QGP is not directly observable, one can infer its properties by the final products of HIC's, produced particles in the colliders's detectors. Some of those particles are part of particle aggregates named *jets*, which interact with QGP. As such, changes in the jet properties can be used to quantify QGP properties. The goal of this thesis is to refine the jet-medium interactions currently implemented in MARTINI (Modular Algorithm for Relativistic Treatment of heavy IoN Interactions)

Theoretically, the computational tools available improved over the last decades allow us to perform calculations otherwise impossible. Among those is the volume of compute power available to research institutions. In this thesis, we perform millions of simulations to predict the outcome of HIC's. As such, we can compare which physical formalisms best describe QCD dynamics in HIC's, by reproducing experimental data. Another increasingly available computational tool we use is machine learning. In this thesis, we utilize machine learning in two cases to improve our simulations. The first case is to calculate the initial color gauge fields resulting from the colliding nuclei, as a starting point to an iterative calculation. In the second case, neural networks allows us to compute more refined radiative rates from jets, as opposed to having to resort to cruder approximations.

The study of heavy-ion collisions, quark gluon plasma, and QCD gives us insight on subatomic particles interactions. Ultimately, a better general understanding of QCD will help define the phase transition between a hadron gas and QGP.

## 1.2 Thesis structure

Chapter 1 lays out a broad introduction giving a historical view of this thesis.

Chapter 2 lays out the sufficient background knowledge, both in Heavy-Ion collisions and Machine learning, to understand the research project of this thesis. It elaborates on the theory behind Quantum Chromodynamics and introduces how Heavy Ion collisions are modeled numerically. These simulations then lead us to computational challenges and approximations taken, to which we propose solutions by the use of machine learning.

Chapter 3 dives into jet production and quenching. We present the AMY model as a baseline, then a more generalized solution by Caron-Huot and Gale, then a first attempt to emulate the CHG formalism. and finally detail into the methods we use to fully implement the CHG formalism.

Chapter 4 digs deeper into the theory of initial conditions, we then explore a numerical solution to the problem, and finally we propose a novel improvement upon the numerical approach and its implementation.

Chapter 5 lays out and discusses the results for both the initial conditions part and the jet evolution parts of this thesis. The different jet evolution formalisms are compared with experimental data.

Finally, the conclusion summarizes the outcomes of this thesis and lays out future works.

# Chapter 2

# Theoretical Background

## 2.1 Quantum Chromodynamics

The discovery of subatomic particles soon shed light upon a missing quantum number. The $\Delta^{++}$ wave-function, composed of three up quarks appeared to be symmetric under exchange, violating the Pauli exclusion principle as quarks are fermions. The solution to this puzzle was the proposition of the color charge, independently proposed by Gell-Mann and [16]. This gave rise to the SU(3) quark model, a non-abelian gauge theory. Analogously to carrying a positive or negative electric charge, quarks and gluons carry a color charge, which can be red, green, or blue, as well as their anti-particle counterparts. These colors then dictate the strong forces between quarks. The SU(3) group, which describes the colors is generated by the eight Gell-Mann matrices, which in turn correspond

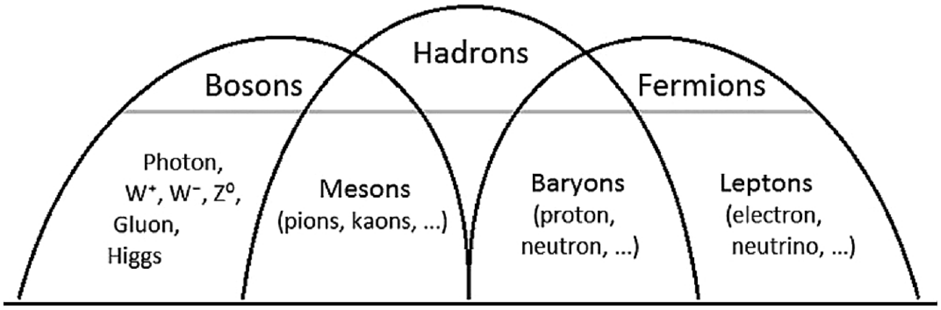

**Figure 2.1:** Classification of particles. Taken from [1]

to the gluons. Quarks are also classified in three pairs of flavors with electric charges $(+2/3, -1/3)$. These pairs are the up and down quarks, the charm and strange quarks, and the top and bottom quarks. These flavors are what dictate the weak force between quarks.

### 2.1.1 Yang-Mills Lagrangian

Considering color and flavor, we can now write down the QCD lagrangian:

$$\mathcal{L}_{QCD} = \bar{\psi}_i (i \not{D}_{ij} - m\delta_{ij})\psi_j - \frac{1}{4} F_a^{\mu\nu} F_{\mu\nu}^a \tag{2.1}$$

where the quarks color indices are $i, j$ in $\psi$ , and the gluon color index is $a$. The quark fields $\psi_i$ then interact through the gauge covariant derivative

$$D_\mu = \partial_\mu - igA_\mu = \partial_\mu - igA_\mu^a t_a \tag{2.2}$$

contained in $\not{D} = \gamma^\mu D_\mu$, which in turn couple the quarks to the gluons through the gluon fields $A_\mu$. $\not{D}$ is the covariant derivative in Feynman notation, where a summation under Einstein's notation is understood, and the $\gamma^\mu$ terms are the Dirac gamma matrices. We should note that the $t_a$ terms contains the generators of the SU(3) group, namely the Gell-Mann Matrices.

The other interactions arise from the gluon field strength tensor:

$$F_{\mu\nu}^a = \partial_\mu A_\nu^a - \partial_\nu A_\mu^a - gf^{abc} A_\mu^b A_\nu^c \tag{2.3}$$

Where the term $f_{abc}$ is the group structure constant [17]. The $g$ term found in both the stress tensor and the gauge covariant derivative is called the strong coupling constant. Its presence in the gluon field strength tensor is the primary difference between the QED and QCD lagrangians. At lower energies, where this strong coupling is large, there is what is called confinement; quarks are so strongly bound together that they cannot be observed

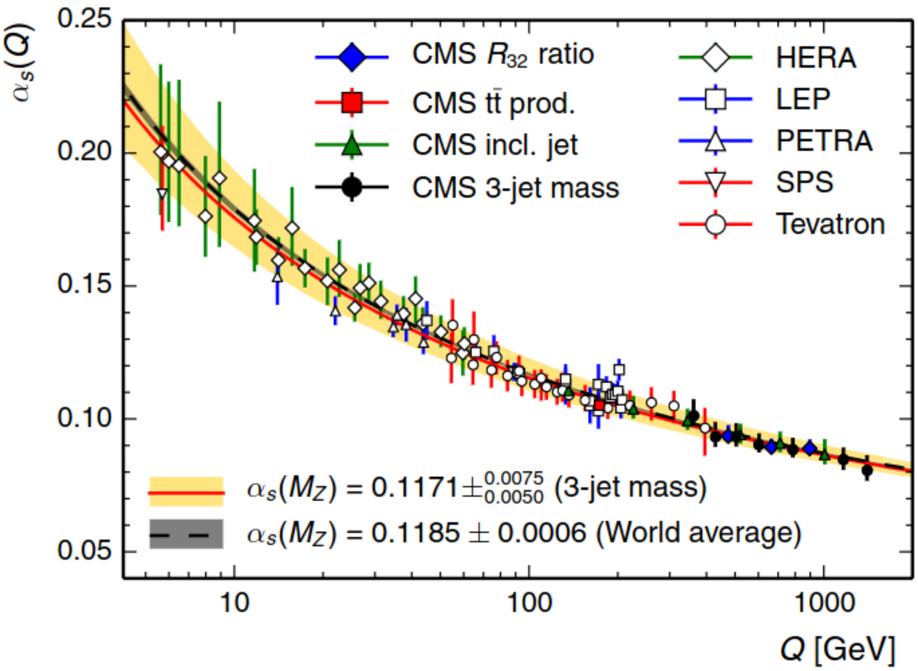

**Figure 2.2:** Strong coupling constant $\alpha_s$ as a function of energy. $\alpha_s$ is related to $g$ as $\alpha_s = g^2/4\pi$. Taken from [2].

individually. Introducing enough energy to separate the two quarks will instead result in the creation of another quark-antiquark pair. At high energies, however, the coupling becomes small; this is known as asymptotic freedom and is a characteristic feature of QCD, contrary to QED. Asymptotic freedom leads to deconfinement; quarks are now free to exist on their own. The dynamical nature of the strong coupling as a function of energy as depicted in fig. 2.2 is called *running* coupling. As a consequence of the running coupling, different methods must be used to study QCD at different energies. At high energies, where the coupling is small, perturbative methods are valid and we can use perturbative QCD (pQCD). At low energies, where the coupling is large, lattice QCD methods are used to evaluate Feynman path integrals.

### 2.1.2 Quark-Gluon Plasma in the QCD Phase Diagram

A consequence of deconfinement is the existence of Quark-Gluon Plasma; under high temperature and/or chemical potential as illustrated in fig. 2.3, quarks and gluons will

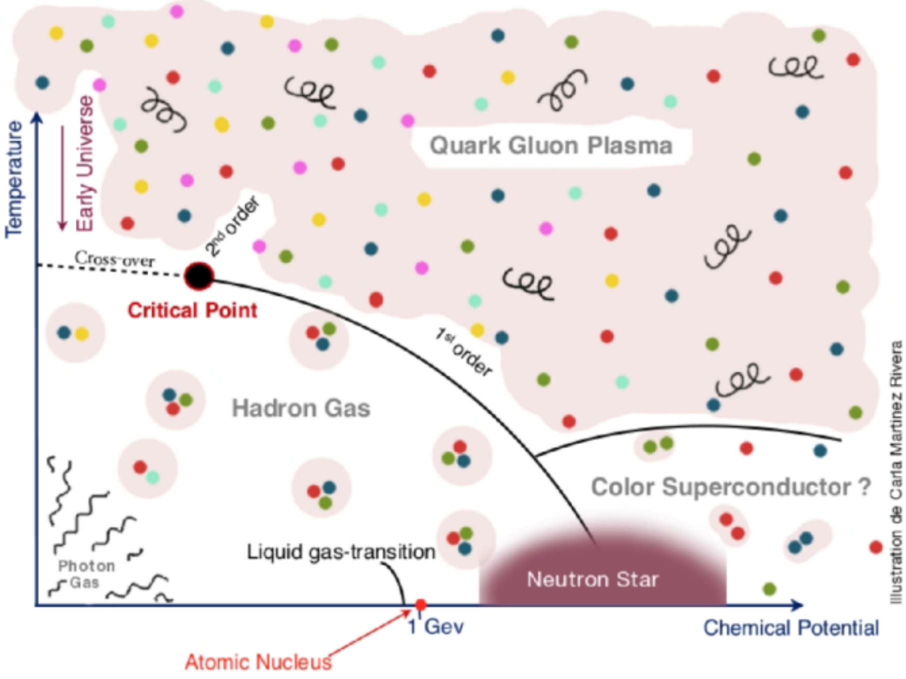

**Figure 2.3:** Schematic QCD phase diagram. Taken from [3].

behave independently and thus in large numbers can collectively be considered as a fluid. Unless there is a highly dense medium such as neutron stars, the required temperature to achieve deconfinement is in the order of 100 MeV, which in the Kelvin/Celcius scale is in the order of $10^{12}$, or about a million times the temperature of the core of the sun.

Experimentally, QGP is produced in Heavy Ion Collisions (HIC), where high energy beams of nuclei whose electrons have been removed are collided. A collision of two nuclei can produce a droplet of QGP with a lifespan in the order of a few fm/c ($\sim 10^{-23}$s) and a transverse size of a few fm ($\sim 10^{-14}$m). Regardless of QGP formation, HIC's final product is a distribution of particles scattered across the walls of the colliders. The QGP's sheer size and lifespan makes it difficult to observe empirically. However, other observables can be used to infer properties of the QGP.

Besides forming QGP, heavy ion collisions can also produce collimated showers of particles known as jets, which interact with the QGP formed from other hard scatterings. By reconstructing their history, we can study the exchange of energy between the medium and the jets, probing the QGP. To study QGP, we then resort to modeling and simulating HIC experiments, specifically jet's evolutions.

## 2.2 Heavy Ion Collisions Modeling

As theorists, we are interested in predicting the resulting observables from HIC experiments. Given the nondeterministic nature of the experiments, we must resort to matching distributions of observables. We produce our predictions by simulating the experiments several thousands of times using Monte-Carlo techniques, and break down the simulations as follows.

### 2.2.1 Collision parameters

When performing a heavy-ion collision experiment, very little can be controlled in a specific collision. One of the tunable parameters is the invariant energy of the collision, considering the momentum and mass of both particles. It is denoted $\sqrt{s}$, in reference to the Mandelstam variable [18].

The other remaining parameter is known as centrality. Consider two classical uniform spheres of radii $R_a$ and $R_b$. For a collision to happen between the spheres, they should contact such that impact parameter, the distance $b$ between their centers of mass should be such that $b < R_a + R_b$. Collisions with smaller impact parameters are called more central, while collisions with larger impact parameters are more peripheral.

We note that the impact parameter directly affects the size of the overlapping region of the two bodies colliding. In the case where the colliding bodies are nuclei, the size of the colliding region translates into the quantity of nucleons which will undergo at least one inelastic nucleon-nucleon collision. Those nucleons are called participant nucleons, while the rest of the nucleons are said to be spectators.

While we cannot experimentally measure the impact parameter, counting the number of spectator nucleons can be a good indicator to characterize the collision. In practice,

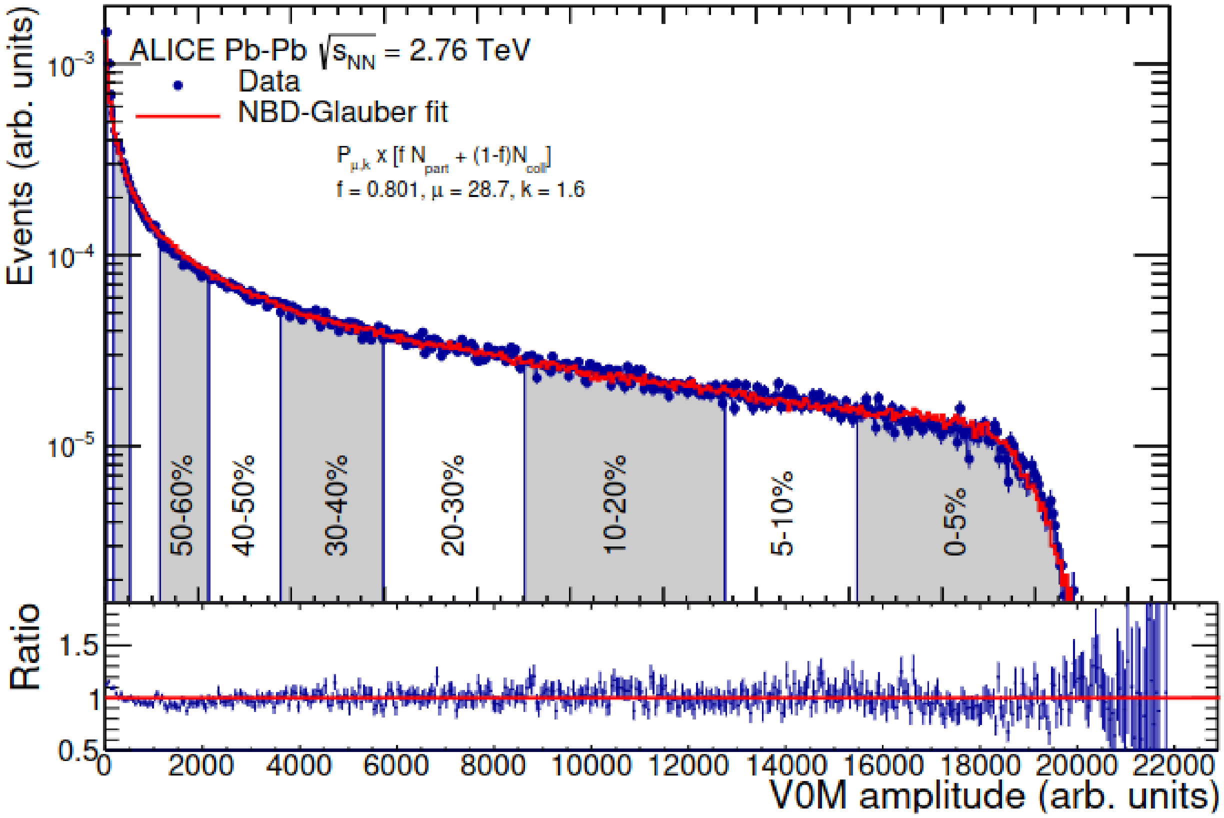

**Figure 2.4:** Centrality bins in a Pb-Pb collision at $\sqrt{s} = 2.76$ TeV. The V0M quantity is proportional to the energy of the particles hitting the collider detectors. Taken from [4]

however, centrality is used instead. Centrality is defined as the percentile of the cross section corresponding to a particle multiplicity or to the total energy deposited above a given threshold [4].

$$c \approx \frac{1}{\sigma_{AA}} \int_{N_{CH}^{THR}}^{\infty} \frac{d\sigma}{dN'_{CH}} dN'_{CH} \approx \frac{1}{\sigma_{AA}} \int_{0}^{E^{THR}} \frac{d\sigma}{dE'} dE' \tag{2.4}$$

While this integral is the analytical formulation of centrality, the finite, discrete, non-continuous nature of physical observables makes us resort to an approximation. The collected particles in the detectors are sorted by their energy, and centrality is taken to be the percentile of the energy histogram.

### 2.2.2 Initial Conditions

Besides choosing the nuclei and the collision energies, no other parameter can be tuned and has to be inferred. That is, we have no prior information regarding the position and momentum distribution of the nucleons, let alone that of the quarks and gluons. The first step of modeling the experiment is thus the modeling of the initial conditions. Namely, we model the projectile, the target, and the combination of both such that boundary conditions are respected and energy is conserved. As seen in section 3, the modeling framework we use is called Color Glass condensates. In this thesis, we revisit an iterative numerical solution for the initial conditions, and suggest a machine-learning based improvement upon it. The numerical framework we use to evolve the colliding nuclei in time is called IP-Glasma.

### 2.2.3 Pre-equilibrium

As explained in [19], the average momentum fraction $x$ carried by a radiated particle is given by

$$x = \frac{< P_T >}{\sqrt{s}/2} \tag{2.5}$$

which is in the order of $x \sim 10^{-4} - 10^{-3}$ for energies relevant in HIC experiments. As depicted in fig. 2.5, the parton distribution functions favors gluons in this $x$ regime, so we have a largely gluon dominated system. While a large parton momentum $Q$ will radiate gluons, gluons can recombine as well. These two processes compete, and as a result the gluon multiplicity per volume is bounded, and saturated at the saturation momentum scale $Q_s$ [5]. Until $\tau \sim 1/Q_s$, the gluon dominated system follows classical field equations [20], at which point kinetic theory kicks in, and QCD interactions create quark anti-quark pairs $q\bar{q}$. This process continues until the system has reached a state of QGP in thermal equilibrium, at $\tau_{eq.}$.

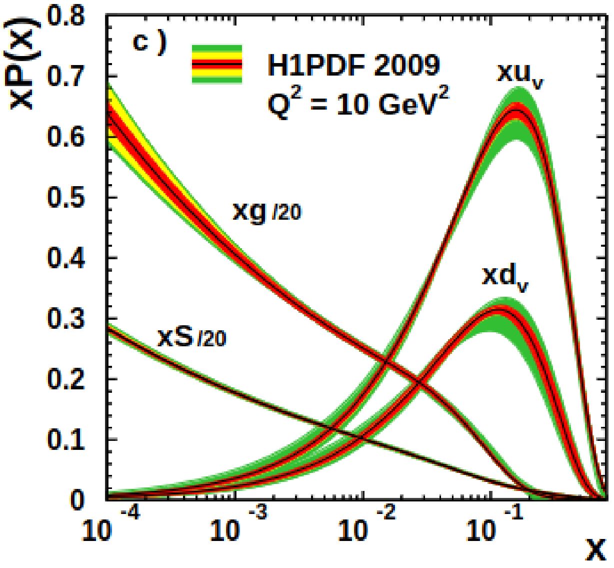

**Figure 2.5:** Parton Distribution Function, probability to find a parton species as a function of momentum fraction $x = k/p$ at $Q^2 = 10\mathrm{GeV}^2$. Taken from [5]

### 2.2.4 Hydrodynamic Stage

Now in thermal equilibrium, the QGP dynamics can be described by relativistic, viscous hydrodynamics. the system evolves as a fluid and expands, cooling down until the temperature reaches freeze-out, below deconfinement temperature. Conservation of energy and momentum warrant the following condition

$$\partial_\mu \mathcal{T}^{\mu\nu} = 0 \tag{2.6}$$

and the conservation of currents warrants the following

$$\partial_\mu \mathcal{J}^{\mu} = 0 \tag{2.7}$$

which, when closed with the following equation of state describe the Stress Energy tensor of QGP.

$$T^{\mu\nu} = \epsilon u^\mu u^\nu - (g^{\mu\nu} - u^\mu u^\nu)(P + \Pi) + \pi^{\mu\nu} \tag{2.8}$$

as described in [21]. The software we use for the treatment of these equations is called MUSIC [22], which picks up the initial conditions directly from IP-Glasma [23]. Once below freeze-out temperature, the fluid undergoes hadronization, where QGP is converted into hadrons by following the Cooper-Frye formalism as in [24], by integrating the momentum distributions of the $T^{\mu\nu}$ from MUSIC.

Once we have the full hydrodynamic evolution, we are ready to study jets, and specially their interactions with QGP. We take a distribution of partons from the Pythia software [25], which we then evolve along the QGP evolution from MUSIC. The software we use is called MARTINI (Modular Algorithm for Relativistic Treatment of heavy IoN Interactions) [26], which utilizes a Monte-Carlo procedure in which the elastic and inelastic scatterings are allowed, according to the formalism in use. In this thesis, one of the challenges we tackle is to apply a new formalism with the use of machine learning. We note that while we allow for the effects of the QGP on the jets, we do not consider the jet's effect on the medium, as it would alter the evolution which has prior been calculated.

### 2.2.5 Hadron gas

At this stage, color-charged particles have paired with each other to create color-neutral hadrons, and have reached *chemical freezeout*. These new hadrons now behave as a gas. As such, they are subject to collisions between hadrons, and thus rescatter. This evolution is modeled within the UrQMD [27] framework, which treats the hadrons under the relativistic Boltzmann equation. This is carried until the hadrons no longer collide, and have reached *kinetic freezout*. The particles are then free to travel and are collected by the colliders detectors.

## 2.3 Artificial Intelligence

Artificial intelligence (AI) nowadays is ubiquitous in popular culture. In february 2023, the popular chat-bot by open-AI chat-GPT broke the record for the fastest growing application in history, with 100 million active users within two months. Even unconsciously, consumers use AI in their daily lives, such as following their GPS recommendations, or selecting auto-complete options when writing on a phone or a computer. Outside of the consumer world, AI increasingly gains popularity amongst different sectors in industry. Drug discovery models are being used in medical research, machine learning (ML) models are being used to predict the stock market, and self-driving cars will likely gain presence in the foreseeable future. Besides drug discovery, scientific research at large also benefits from ML with, for example, image classification models being used to find new planets, amongst the astronomical data at hand.

However, AI as a concept has existed for a few centuries. For example, automatons are mechanical creatures in greek mythology, or the hoax known as the mechanical turk in the 1700's, which was in fact a chess-master pretending to be a machine playing against people. AI has formally been an academic field of study since 1956 in Darthmouth college [28]. The field has known waves of optimism and funding followed by waves of scepticism known as AI winters. Until the 1980's, machine learning research was biologically inspired, and mostly restricted by the use of small, synthetic data sets.

### 2.3.1 Machine Learning

In the past decades, the increasing presence of the internet in our lives has led to an increased availability of data. Simultaneously, the increasing performance and availability of computational resources, namely Graphics Processing Units, have increased the capacity to train machine learning models. Together, those factors have set the ground for the unrivaled performance and popularity of machine learning. Machine learning can be broadly categorized in supervised and unsupervised learning, where the latter searches for information within data without a label or target output. In the scope of this thesis, we will only consider supervised learning.

TSupervised machine learning encompasses different model types such as causal models, which attempt to derive a causal effect, for example, a physical law, and would satisfy

E. Rutherford's quote "If your experiment needs statistics, you ought to have done a better experiment.". While this approach has the merit of being elegant, it might lead to overfitting, as a complex enough model will go through as many points as we want it to.

However, most successful machine learning models today are statistical. Instead of looking for a definite cause-and-effect law, statistical models look for a mapping $f$ between two vector spaces $X$ & $Y$, assuming a probability distribution $P$ over the product space $X \times Y$ while aiming to minimize the empirical risk, defined as follows

$$I_f = \int_{X \times Y} V(f(x), y) \times P(x, y) dx dy \tag{2.9}$$

where $V$ is a distance metric between the target value $y$ and the predicted value $f(x)$, which must be chosen and is often called the loss or cost function. Since the probability distribution $P$ is unknown, it is often assumed to be uniform and thus the integral becomes an average over all samples. This may lead to an biased model $f$ in the case where the data set is unbalanced, in which case a non-trivial P might be assumed. For example, consider a binary classification task where there are 9 times more samples from class A available than from class B. Unless you discard part of the data set for it to be balanced, or re-weight the importance of the samples, the model might learn to be biased and always predict an outcome of class A, as it leads to a 90 % accuracy.

### 2.3.2 Cost Functions

"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E (Tom M. Mitchell, 1998)."

In the statistical, supervised machine learning context where the goal is to minimize empirical risk, the following formulation is commonly used:

$$J(\theta) = J(y, f_\theta(x)) \tag{2.10}$$

where the performance P is inversely related to the cost function $J(\theta)$, the task T is for the function $f_\theta$ to predict the output $y$ given input $x$, and $\theta$ is the set of the $f$ model's learnable parameters. Equivalently to minimize empirical risk is to maximize the likelihood function between the training data and the ML model distribution. Mathematically, that is to find the set of parameters $\theta_{ML}$ which maximize the likelihood function. That is, to take the $\operatorname{argmax}$ in the following way:

$$\theta_{ML} = \operatorname*{argmax}_\theta p_{model}(\mathbf{X};\theta) = \operatorname*{argmax}_\theta \prod_{x \in \mathbf{X}} p_{model}(x;\theta) \tag{2.11}$$

Since $\theta_{ML}$ is invariant to taking the log, and it helps us numerically to do so, we take the log of the probability distribution to obtain a sum instead of a product. Note that eq. 2.12 is not taking the log of both sides of the equation, instead it is restating that $\theta_{ML}$ is unchanged when taking the log of the right hand side. Similarly, scaling the right hand side by the number of samples does not change $\theta_{ML}$, and so we can rewrite eq. 2.11 as an expectation value, $\mathbb{E}$.

$$\theta_{ML} = \operatorname*{argmax}_\theta \mathbb{E}[\log p_{model}(x;\theta)] \tag{2.12}$$

Conveniently, this form of writing $\theta_{ML}$ matches the definition of the *Shannon entropy* $H(x)$ of a distribution $P$ over $x$, $H(x) = -\mathbb{E}_{x\sim P}[\log P(x)]$. We can now utilize a useful metric known as the Kullback-Leibler (KL) divergence, which quantifies the difference of two distributions [29] by the use of Shannon entropy. It has the properties of being zero for identical distributions, and being positive otherwise. Equivalently to finding $\theta_{ML}$ is to minimize the KL divergence between the data distribution and the model distribution over the input space $x$, written as follows:

$$D_{KL}(\hat{p}||p_{model}) = \mathbb{E}_{x\sim p_{data}}[\log \hat{p}_{data}(x) - \log p_{model}(x)] \tag{2.13}$$

Since the first term does not depend on the model, to maximize maximum likelihood is equivalent to minimizing the following negative log likelihood, which is the cost we will use

$$J(\theta) = -\mathbb{E}_{\mathbf{x},\mathbf{y}\sim\hat{p}_{data}} \log p_{model}(\mathbf{y}|\mathbf{x}) \tag{2.14}$$

where the data space $x$ has been decoupled into an input and target space $\mathbf{x}$ and $\mathbf{y}$. The exact form of $J$ is both model and task dependent. For example, a classification task with a discrete set of possible values has a different cost form than a regression model with a continuous output. In the case where the probability distribution of the model is normally distributed as in [29] with standard deviation $\sigma$ and $\mathbf{y}$ has $n$ dimensions,

$$p_{model}(\mathbf{y}|\mathbf{x}) = (2\pi\sigma^2)^{-n/2} \exp\left(\frac{-1}{2}\left(\frac{f(\mathbf{x}) - \mathbf{y}}{\sigma}\right)^2\right) \tag{2.15}$$

the resulting cost function is the mean squared error

$$J(\theta) = \frac{1}{2}\mathbb{E}_{\mathbf{x},\mathbf{y}\sim\hat{p}_{data}}||\mathbf{y} - f(x,\theta)||^2 + const. \tag{2.16}$$

Deriving this cost function from maximum likelihood has the advantage of being model independent. That is, as long as the model $p_{model}(\mathbf{x}|\mathbf{y})$ is Gaussian, our cost function holds regardless of the function $f(\mathbf{x};\theta)$ used to predict the mean of the Gaussian.

### 2.3.3 Learning

Now assuming we have a well posed task T with quantified performance P, it remains to determine how to improve P with experience E. In other words, how to iteratively update the machine learning model such that P is improved. Several algorithms have been proposed such as Monte-Carlo techniques, and even biologically inspired genetic algorithms which replicate survival of the fittest. However, the most prevalent algorithm and the one we will use in this thesis is gradient descent and its derivatives.

Let us reconsider the cost function from Eq. 2.16 . In this context, given a dataset of inputs and targets $(x, y)$, to learn would be to update the parameters $\theta$ such that the cost is minimized. The gradient descent approach is to update $\theta$ such that

$$\theta_{t+1} = \theta_t - \alpha \nabla J(\theta_t) \tag{2.17}$$

where the term $\alpha$ is called the *learning rate*, which scales the step at which $\theta$ is updated. The gradient descent updating iteration is usually carried out until a determined number of *epochs*, or times the whole dataset has been used to update the model, is reached, or until the change in cost function becomes too small for any update $||J(\theta_{t+1}) - J(\theta_t)|| < \epsilon$ for a chosen $\epsilon$.

### 2.3.4 Optimization

Several variations of gradient descent have been proposed, where the varying aspects are usually the form of the learning rate, and the data samples taken in the computation. Stochastic Gradient Descent, for example, introduces randomness in the procedure by randomly selecting a data point, instead of the whole training dataset. That allows for cutting computing costs, at the expense of utilizing all available data for an accurate gradient. A trade-off is possible when utilizing batches of data samples.

Generally, learning rates are scheduled, $\alpha = \alpha(t)$ such that a larger learning rate is preferred if $\theta$ is randomly initialized, and it is gradually increased to fine tune the parameters as an optimum is searched.

A commonly used optimizer and one we will use in this thesis is called Adam [30], from Adaptive moment estimation. The Adam algorithm utilizes a fixed learning rate $\alpha$, but instead of using a scaled gradient, it defines and utilizes a first ($\hat{m}_t$) and second ($\hat{v}_t$)

moment estimates as follows:

$$\begin{aligned} g_t &= \nabla_\theta f_t(\theta_{t-1}) \\ m_t &= \beta_1 * m_{t-1} + (1 - \beta_1) * g_t \\ v_t &= \beta_2 * v_{t-1} + (1 - \beta_2) * {g_t}^2 \\ \hat{m}_t &= m_t / (1 - \beta_1^t) \\ \hat{v}_t &= v_t / (1 - \beta_2^t) \end{aligned} \tag{2.18}$$

Where $\beta_i^t$ is raised to the power $t$, and ${g_t}^2$ is the component-wise square of the gradients. The gradient $g_t$, as well as the first and second moments are initialized to zero. The moments are interpreted to be exponential moving averages of the mean and non-centered variance of the parameters, and are scaled by $1/(1 - \beta_i)$ to make up for the bias towards zero from the initialization. These moments are then used to update the model parameters as follows:

$$\theta_t = \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon) \tag{2.19}$$

Where the square root again is element wise, and epsilon is chosen for numerical stability.

While the vanilla gradient descent only relies on the current iteration's gradient, the Adam algorithm implicitly relies on all previous time steps. As pointed out in [31], Adam fails to converge to an optimal solution in common settings. A solution [31] offers is to equip the optimizer with "long-term memory", by having an explicit dependence on all previous iterations. Instead of taking the exponential moving average of the second moment estimates, the element-wise maximum is taken at each time-step, and thus ultimately the element-wise maximum across all time steps. The moments are defined as in Eqs. 2.18, except that $\hat{v}_t$ is defined by $\hat{v}_t = max(\hat{v}_{t-1}, v_t)$, and $\hat{m}_t = m_t$, then the updating rule Eq. 2.19 still holds. This modified version of Adam is called AMSGrad, and will be used in this thesis.

### 2.3.5 Common Practices

Goodhart's law, often cited as "When a measure becomes a target, it ceases to be a good measure" arose in an economical context, as a critic of monetary policy in the United Kingdom in 1975. It stated that once a economical feature is recognized as a good indicator, it ceases to be so because people will then trick the metric. The principle is applicable within our ML optimization context. Indeed, while applying gradient descent-based optimization methods, well-behaved derivatives are preferred. In practice, however, discrete target values are common, for example in a classification task. In those cases, there is a need to define a second metric. The discrete prediction is often referred to as "hard", and the first metric which is prediction accuracy follows from it. Often, the "hard" prediction is the category with the highest continuous value, which itself is the "soft" prediction. The second metric then is associated with these continuous values, and has a second metric associated with it, namely the "surrogate" metric, or "loss", and is the one to which we apply an iterative optimization. This principle of soft and hard predictions will be taken into account when designing a loss function for our models.

When given a data set, it is considered good practice to split the set into a validation and training set. At each learning iteration, the model is updated with the optimization algorithm computed on the soft prediction of the training set, but the model is ultimately chosen based on the performance on the hard prediction on the validation set, as it is a better indicative of generalized performance.

Besides a model's learnable parameters, there are parameters that are set outside of the learning iteration. These can be related to the model structure, such as the number of layers and dimensions, or to the optimization, such as the algorithm choice or the learning rate. These contextual parameters are collectively referred to as hyper-parameters. When training machine learning models, it is common to simultaneously train several different models with varying hyper-parameters.

### 2.3.6 Models

So far, all of the discussion has been model independent. Most of modern ML models are derived from Neural Networks. Biologically inspired, neural networks are characterized by layers of nodes and links which resemble neurons and synapses in a biological brain.

#### 2.3.6.1 Multi-layer perceptrons

ML Neural networks are a broad class of mathematical models, of which the simplest is called the multi-layer perceptron (MLP). MLP's have at least three layers: An input layer, a hidden-layer, and an output layer. The input layer is a vector, to which an affine transformation is applied, then fed into the hidden layer. The hidden layer, with an arbirary number of nodes, or *width*, then applies an activation function to the incoming vector. There have been several activation functions proposed, namely the sigmoid function $1/(1+e^{-x})$, the step function, the hyperbolic tangent function, and the Rectified linear Unit (ReLU) [32], $f(x) = max(0, x)$. While these are all mathematically different, they all behave similarly as they are or approach zero for negative arguments, and are non-zero for positive arguments. Often, these are chosen based on the task, although some are more popular for computational reasons. The non-linearity introduced by the activation functions is what allows MLP's to behave highly non-linearly. Another affine transformation is then applied to the output of the activation function before being used as an argument to either another hidden layer or an output layer. The chosen number of layers is referred to as the depth of the network. Finally, the output of the last hidden layer is fed to the output layer, which is a linear transformation mapping the output of the hidden layer to a vector of the dimensions of the desired output. While conceptually simple in design, MLP's have proven to be extremely versatile. In fact, The Universal Approximation Theorem states that a neural network with a single hidden layer containing a finite number of neurons can approximate any continuous function [33].

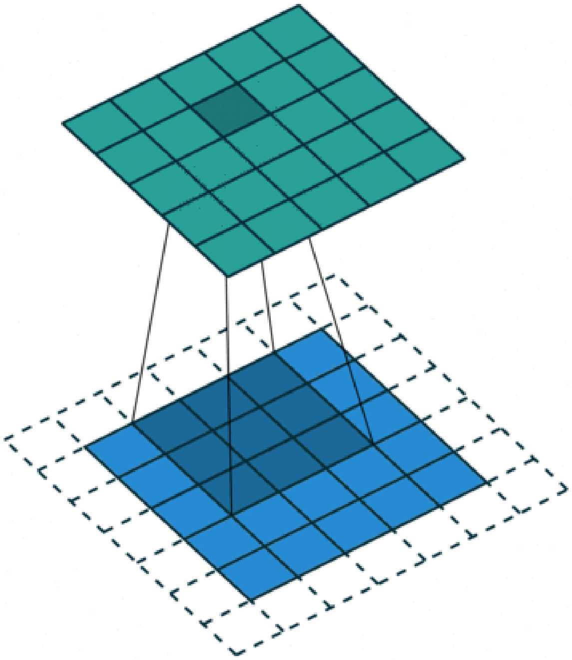

**Figure 2.6:** An example of how a convolutional kernel is swept over an image array. The dotted lines represent the padded pixels which are taken to be zero. Taken from [6]

#### 2.3.6.2 Convolutional neural networks

One can take advantage of the structure of the data at hand to design more sophisticated networks. For example, sequential, time-ordered data has inspired recurrent neural networks, of iterative nature. Convolutional neural networks (CNN's) exploit the spatial organization of the data points within images to achieve better performance with less model parameters. Instead of a simple one-dimensional vector, CNN's expect image-like data, of at least two spatial coordinates $x$ & $y$, with $(N_x \times N_y)$ *pixels*, and at least one channel at each pixel. As such, instead of having an operation applied to all pixels, a convolutional kernel is defined then applied locally. A kernel is an array of $k_x \times k_y$ parameters such that $k_x, k_y < N_x, N_y$, which is swept over the image as depicted in fig 2.6, where it applies a *Haddamard product* operation, or element wise multiplication between the image and the kernel elements. Under this formulation, the dimensions of the information image change as follows (Assuming $k_x = k_y \equiv K$):

$$(N_x, N_y) \to (\frac{N_x - K + 2P}{S} + 1, \frac{N_y - K + 2P}{S} + 1) \tag{2.20}$$

Where a stride $S$ is defined as the step size the kernel takes between each operation, and the input image can be padded with zeros of size $P$ outside the edges, to ensure that the edge pixels are utilized as much as the inner pixels. Known kernels are used in non-machine learning settings, for example, in edge detection algorithms. However, allowing the kernel parameters to be learned and iteratively applying convolution operations allows for abstract local features to be utilized, and has proven to be successful in image processing. Convolutional neural networks are the building blocks for the UNET model we will use later on.

# Chapter 3

# Initial Conditions

## 3.1 Theory

### 3.1.1 Color-Glass Condensates

The Color-Glass Condensate (CGC) is an effective field theory that governs the pre-collision dynamics of heavy-ion collisions. It considers the color charges and color gauge fields of the valence quarks of the colliding nuclei instead of their nucleons, and implicates color charge as the central driver for the formation of Quark-Gluon Plasma (QGP). The term "glass" in CGC refers to the boosted partons, which travel at highly relativistic velocities and act as sources for the color gauge fields, and exhibit properties of amorphous solids that behave like a fluid on longer time scales but as a solid on the collision's time scales, due to strong time dilation. The term "condensate" refers to the coherent behavior of gluons in CGC due to the very high phase space density of gluons at length scales smaller than the saturation scale, $Q_s$. $Q_s$ is the energy scale at which gluon density ceases to increase, as gluon recombination processes compete with gluon production.

QCD equations of motions are very difficult to solve, even when using supercomputers. CGC's, being an effective classical field theory, greatly reduces this problem. It describes the quark-gluon system in effective degrees of freedom and the new Lagrangian is as follows:

$$\mathcal{L}_{CGC} = J^{\mu a} A^a_\mu - \frac{1}{4} F^a_{\mu\nu} F^{a\mu\nu} \tag{3.1}$$

When comparing 3.1 to the QCD lagrangian 2.1, the first notable difference is that the fermion terms are gone, since they do not contribute to the *effective* field. Secondly, we have the new term $J^{\mu a}$, which corresponds to a classical source term. Indeed, under the CGC formalism, partons with a large portion of momentum of the hadron $x = k/p$ act as classical sources for the smaller $x$ partons. The field term $A^a_\mu$ comes from the small $x$ partons.

### 3.1.2 Equations of Motion

Equipped with an effective Lagrangian, we can now apply the Euler-Lagrange equations to describe the evolution of the system and obtain the Yang-Mills Equations [34]:

$$[D_\mu, F^{\mu\nu}]_a = J^\nu_a \tag{3.2}$$

$$D_\mu = \partial_\mu + igA^a_\mu t^a \tag{3.3}$$

Where the gluon fields $A$ do not commute, and the field strength $F^{\mu\nu}_a$ covariant derivative $D_\mu$ is non-linear. This leads to the *asymptotic freedom* mentioned in the introduction, which in turn led to the possibility of QGP. We will now return to the evolution resulting from these equations.

First recall that the nuclei are traveling near the speed of light in opposite directions near the speed of light. In a Minkowski diagram as in fig. 3.1, the nuclei are therefore the light cone boundaries, crossing at $t = 0$, unscathed due to the eikonal approximation. For $t > 0$, the gluon fields arising from the hard partons sources evolve inside this light cone, inheriting boundary conditions from the nuclei along the cone. A gauge can be chosen such that $A^+ = A^i = 0$, reducing our gluon field strength tensor in the negative direction

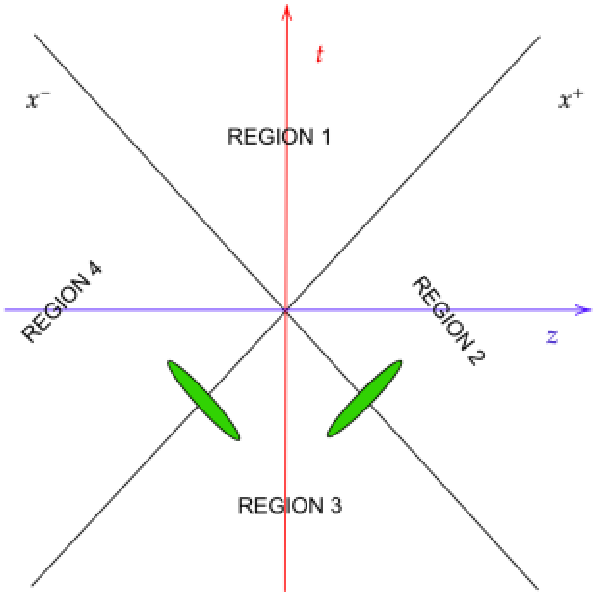

**Figure 3.1:** Minkowski Diagram, where the nuclei (in green) trajectories are approximated to be the light-cone boundary. Taken from [7].

to

$$F^{i-} = \partial^i A^-_C \tag{3.4}$$

And inserting this in the yang-Mills Eqs. 3.2 3.3 yields the following Poisson equation.

$$A^- = \frac{-g\rho(x^+, x_T)}{\nabla^2_\perp} \tag{3.5}$$

The same argument can be made to obtain the converse in the positive direction:

$$A^+ = \frac{-g\rho(x^-, x_T)}{\nabla^2_\perp} \tag{3.6}$$

Now this result is in the light-cone gauge, and can be gauge transformed back to the covariant gauge. Upon solving for the gauge fields in the covariant gauge, we obtain a closed-form solution for our color gauge fields, in terms of two source terms, one for each nucleus $a$ & $b$.

$$A^+\big|_{\tau=0} = x^+ \frac{ig}{2}[A^i_a(\tau, \mathbf{x}_\perp), A^i_b] \tag{3.7}$$

$$A^-\big|_{\tau=0} = x^- \frac{ig}{2}[A_a^i(\tau, \mathbf{x}_\perp), A_b^i] \tag{3.8}$$

$$A^i\big|_{\tau=0} = A_a^i + A_b^i \tag{3.9}$$

Eq. 3.9 is the analytical solution encompassing the initial conditions of the system resulting from the two nuclei at $\tau = 0$.

### 3.1.3 The Lattice

So far we have the continuous physical description of color glass condensates. However, to solve Eq. 3.9, we resort to numerical methods, which in turns requires the use of discretized quantities, namely space. The space in which we consider the evolution of CGC is a square lattice grid of $N \times N$ sites with spacing $a = 0.044$ fm $\approx 0.2$ GeV$^{-1}$ and $N = 500$ [7]. The spatial resolution required to resolve these energies can be bound by the saturation scale $Q_s$, such that $a \leq Q_s^{-1} \approx 2$ GeV$^{-1}$.

We can now define operators $U_{x,y}$ to parallel-transport from one site to another in their respective directions, as well as their hermitian conjugates $U_{x,y}^\dagger$ in their opposite directions. Combining $U$ operators we can define a *wilson loop*, utilizing four operators consecutively such that we go around a *plaquette* as follows, as depicted in fig. 3.2.

$$U_{\mu\nu} = U_\mu(x)U_\nu(x + a\hat{\mu})U_\mu^\dagger(x + a\hat{\nu})U_\nu^\dagger(x) \tag{3.10}$$

Now we can expand an operator related to our continuous gauge field around $x$

$$U_\mu = \exp[igaA_\mu^a(x)t^a] \approx 1 + igaA_\mu^a(x)t^a - \frac{1}{2}g^2a^2A_\mu^a t^a A_\mu^b t^b + \mathcal{O}(a^3) \tag{3.11}$$

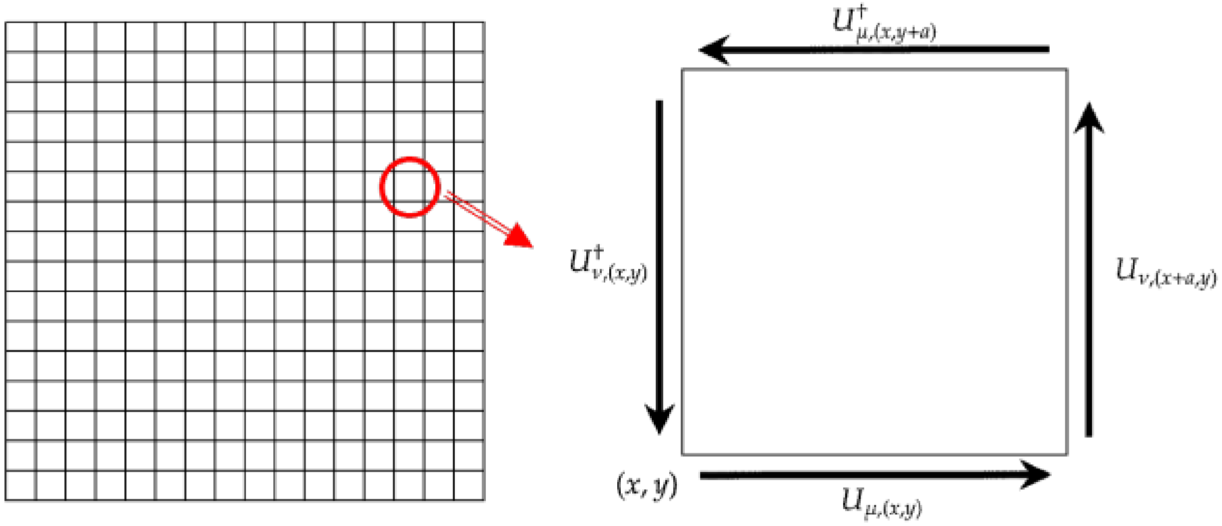

**Figure 3.2:** Discretized lattice and displacement operator $U$. On the right is depicted a *Wilson loop*, which starts and ends at the same point after traveling through nearest neighbors, encompassing a *plaquette*. Taken from [7]

and define a color gauge field Wilson loop.

$$\begin{aligned} U_{\mu\nu} &\approx 1 + iga(A_\mu(x) + A_\nu(x + a\hat{\mu}) + A_\mu(x + a\hat{\nu}) + A_\nu(x))t^a \\ &\approx 1 + iga^2(\partial_\mu A_\nu - \partial_\nu A_\mu) \end{aligned} \tag{3.12}$$

by only keeping linear terms in $a$ and using a finite difference derivative. The terms quadratic in $a$ in turn yield $-g^2a^2[A_\mu, A_\nu]$. Combining both expressions, we obtain

$$U_{\mu\nu} = 1 + iga^2(\partial_\mu A_\nu - \partial_\nu A_\mu) - g^2a^2[A_\mu, A_\nu] + \mathcal{O}(a^3) \tag{3.13}$$

which we can rewrite in terms of $F_{\mu,\nu}$

$$\begin{aligned} U_{\mu\nu} &= 1 + iga^2F_{\mu,\nu} \approx \exp(iga^2F_{\mu,\nu}) \\ U^\dagger_{\mu\nu} &= 1 - iga^2F_{\mu,\nu} \approx \exp(-iga^2F_{\mu,\nu}) \end{aligned} \tag{3.14}$$

where the hermitian conjugate of the Wilson loop simply loops in the opposite direction. These definitions will be useful to address the numerical solution of Eq. 3.9.

### 3.1.4 The Numerical Solution

Equipped with the discrete operators for the color gauge fields defined in the previous section, the discrete analogous of Eq. 3.9, can be derived and yield Eq. 3.15.

$$Tr[t^a((U_i^A + U_i^B)(1 + U_i^\dagger) - (U_i^A + U_i^B)(1 + U_i^\dagger)^\dagger)] = 0 \tag{3.15}$$

Eq. 3.15 is system of 8 non-linear equations, one for each gluon color charge, or SU(3) generator. This is solved by Marius Cautun, Francois Fillion-Gourdeau, and Sangyong Jeon in [35] in an iterative procedure as follows. First, we start by taking the Ansatz of the lattice solution to be the same as the continuous solution, that is that the total field is simply the sum of the target and projectile fields as in Eq. 3.9. In terms of operators, that implies the following:

$$A^i\Big|_{\tau=0} = A_a^i + A_b^i \implies U_i = U_i^{A+B} \iff U^3 = U^1 U^2 \tag{3.16}$$

Which substitued into Eq. 3.15, yields the following

$$\text{ReTr}[t_a(U^1 + U^2)(1 + U^{3^\dagger})] = f_a \tag{3.17}$$

If we let

$$U^3_{current} = e^{i\alpha x_a t_a} U^3_{prev} \approx (1 + i\alpha x_a t_a) U^3_{prev} \tag{3.18}$$

as a recursive relation, with an appropriate $\alpha$ chosen for convergence, we obtain the following condition when substituting into Eq. 3.15:

$$\alpha x_b \text{ImTr}[t_b t_a (U^1 + U^2)[U^3_{prev}]^\dagger] = -f_a \tag{3.19}$$

Solving for $x_b$, we can use the recursion in Eq. 3.18 again, which we substitute into Eq. 3.15 again and continue the iteration until $|f_a|$ has become small enough

## 3.2 Methods

The method this thesis proposes is to build on the work in [35] to tackle Eq. 3.15 by training a neural network with solutions generated with their procedure. A naive approach would be to take the outputs from the neural network for granted. Instead, we chose to train our neural network to predict the initial-conditions gauge fields but to then use it as an Ansatz before starting the iteration procedure described in the previous section. This way, the physical validity of the solution is ensured, and we hope to cut the iterations required to converge to a solution, as opposed to start with the continuous field Ansatz in Eq. 3.16. The use of neural networks as trial wave functions has been tackled previously in [36]. As described in section 3.1.3, we need $N \geq 500$. However, we explore the possibility to train our model on a coarser grid size with $N = 128$. We also train and test models on the energies relevant to our experiments, 193, 2760 and 5020 GeV.

### 3.2.1 Data generation

Both projectile and target nuclei gauge field can be written as $N \times N \times 8$ arrays of floating point numbers with the first two coordinates as spatial indices, and the last coordinate as the color index. We can then treat the arrays as image files with 8 channels, analogous to the usual 3 for Red, Green and Blue. We can then leverage the progress that has been made in image processing machine learning in the past decade to our advantage.

### 3.2.2 Data processing

Before inputting the data into a model, we normalize it as a pre-processing step. Treating each channel independently, we first group all training samples to compute the average and standard deviation of each gauge field index. We save those metrics and scale the inputs by subtracting the mean and dividing by the standard deviation of their corresponding channel when executing model inference.

Given that the analytical solution of the gauge fields combination is symmetric with respect to the projectile and target fields 3.16, we would like our machine learning model to be symmetric with respect to the inputs as well. One could in principle restrict the model parameters such that the same operations would be applied to both inputs

While we do not explicitly enforce the model to be abelian, we take advantage of this criterion to augment our data. For each pair of fields such such that $(A, B) \to C$, we simultaneously compute the loss for both $(A, B) \to C$ and $(B, A) \to C$. The resulting criterion that is minimized, then takes into account both possibilities simultaneously, and results in a model that is on average abelian.

### 3.2.3 Models

We consider and implement two different machine learning models to generate new Ansatzes.

The first model we consider is a Convolutional Neural Network, as described in sec. 2.3.6.2. We vary the number of convolutional layers, and between each convolutional layer, we apply batch normalization as introduced in [37], which accelerates training by normalizing the inputs of each convolutional layer. We also consider varying the kernel size of the convolutions, and the number of hidden dimensions in each layer.

The second model we consider is adapted from a U-Net, which was first proposed by [8] as a medical image classification model. The model works as following; an image of dimensions $N_x \times N_y \times C$, where $N_x$ and $N_y$ are the number of pixels in the x and y direction, and $C$ is the number of channels per pixel, goes through a sequence of encoder blocks followed by a series of decoder blocks. Each encoder block is made of two convolution layers followed by a 2x2 max pooling layer. After each of these steps, the convolution layer doubles the number of feature channels while the max pooling halves the number of spacial points.

The decoder blocks have the opposite effect, while still having two convolutional layers, the max pooling is replaced by an ”up-convolution” such that the number of feature

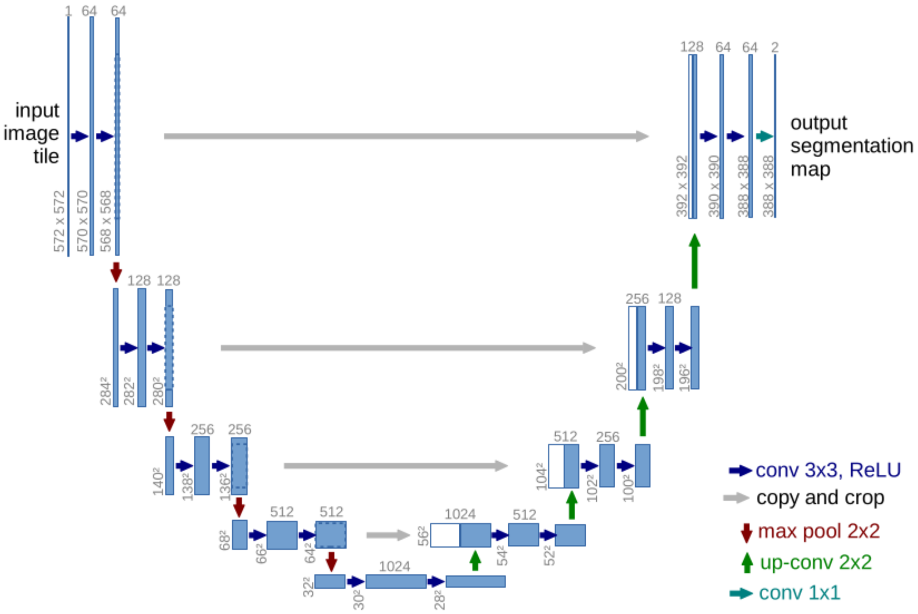

**Figure 3.3:** Scheme of the UNET model, taken from [8]

.

channels is halved and the number of spatial points doubled. The decoder's input is also concatenated with the corresponding encoder's final state as illustrated in 3.3. The final step, the output segmentation map originally is a 1x1 convolution to the desired number of classes, or in our case, colors.

### 3.2.4 Loss criterions

The hard evaluation metric which quantities the performance of our model is the number of iterations reduced when taking the prediction of our model as an Ansatz in the algorithm 3.1.4, as opposed to taking the Ansatz in Eq. 3.16. That metric, however is not differentiable and we must then resort to define a soft metric to minimize, as explained in sec. 2.3.5. Let us recall that our target data has dimensions (batch size)$\times N_x \times N_y \times N_{channels}$. Let $i$,$j$ be the spatial indices, $c$ be the channel index, $y$ be the target value and $\hat{y}$ be the predicted value. Taking the difference, or squared difference pixel-wise and then taking the average does not work, as the range of the target's values spans several orders of magnitude, thus an average would overly favor high-value targets, yielding a poor average percent error overall.

$$\sum_{C,i,j}(y-\hat{y})^2 \tag{3.20}$$

One could then attempt to scale each pixel value to its relative error as follows.

$$\sum_{C,i,j}((y-\hat{y})/y)^2 \tag{3.21}$$

However, some target values are zero, which makes this loss criterion fail. One could then consider scaling by a value representing the gauge field plane for each colour instead of an individual pixel. Applying this approach we obtain the following metrics, and record them across our experiments.

$$L_1 = \sum_{Channels}(\sum_{i,j}(y-\hat{y})^2/\sum_{i,j}y^2) \tag{3.22}$$

$$L_2 = \sum_{Channels}(\sum_{i,j}|y-\hat{y}|/\sum_{i,j}|y|) \tag{3.23}$$

$$L_3 = \sum_{Channels}|\sum_{i,j}(y-\hat{y})/\sum_{i,j}y| \tag{3.24}$$

We chose to minimize $L_1$, (Eq. 3.22) the squared error normalized by the sum of the squares for the smoothness of the derivative. however, it is interesting to think of the other metrics as physical quantities such as the ”total field”, as they do decrease indirectly as the squared error is minimized.

# Chapter 4

# Jet Quenching

## 4.1 Theory

Following the mass-energy equivalence principle, the colliding nuclei in heavy ion collisions result in the creation of several partons, quarks and gluons. The partons which carry the large fractions of momentum from the collision are said to be *hard*, and will continue to dissipate energy by creating other particles through radiation breaking (*Bremsstrahlung*) or scattering processes. These partons will in turn carry momentum and color charge, and create particles as well, iteratively. The particles resulting from these chained reactions have a trajectory similar to a cone spray, and are collectively called jets.

### 4.1.1 Jet Identification

Experimentally, the observables are the final particles in the detectors in a particle accelerator, and those might or might not be part of a jet. To quantify the presence of jets in a collision, several algorithms have been proposed [38] to reconstruct the path the observed particles have taken. The algorithm we will use in this thesis is the $Anti - K_t$ algorithm [39], and goes as follows.

Assuming we have a collection of particles distributed in a $(\eta, \phi)$ space, where $\phi$ is the azimuthal angle, and $\eta = \frac{1}{2}\ln(\frac{|p|-p_z}{|p|+p_z})$ is known as pseudo rapidity and is associated with the angle with the beam axis, then the following distances between entities are computed:

$$d_{ij} = min(\frac{1}{k_{ti}^2}, \frac{1}{k_{tj}^2})\frac{\Delta^2}{R^2} \tag{4.1}$$

where

$$\Delta^2 = (\eta_i - \eta_j)^2 + (\phi_i + \phi_j)^2 \tag{4.2}$$

is the euclidean distance in $(\eta, \phi)$ space between particles $i$ & $j$, $k_{ti}$ is the transverse momentum of particle $i$, and $R$ is a free parameter, usually in the order of $0.1$. We also compute the distances $d_{iB} = 1/k_{ti}^2$, which is the distance in momentum space between entity $i$ and the beam axis.

Once all $d_{ij}$'s and $d_{iB}$'s are computed, we proceed as with other sequential algorithms mentioned in [39]. We find the smallest distance in all $d_{ij}$'s and $d_{iB}$. If the shortest distance is a distance between entities, the entities are combined by four-vector addition. If the shortest distance is between an entity and the beam axis, then that entity is called a jet and it is removed from the list of entities. After either removing or combining an entity, we recompute the distances and start over, until all entities are associated with a jet.

Since jet reconstruction algorithms can be applied to experimental as well as simulated data, it is a good instrument to compare experiment and theory.

### 4.1.2 Observables

As jets travel through QGP, they interact with the medium, mostly by dissipating energy, but will sometimes gain energy from the medium as well. In a proton-proton collision, since there is only one possible binary collision, there can be no QGP from a different binary collision, and thus we consider the jet yield of p-p collisions as a baseline. In practice, however, due to having beams of particles colliding rather than single particles, collisions can interfere. Experimental physicists then isolate these events post-facto.

Jet energy dissipation is HIC's is commonly known as jet *quenching*, and is quantitatively measured with the ratio of jet production with and without QGP. That ratio is referred to as

$$R_{AA}^{jet} = \frac{1}{N_{coll}(c)} \frac{dN_{AA}/dp_T}{dN_{pp}/dp_T} \tag{4.3}$$

where the nominator is the yield in a Heavy-Ion/Heavy-Ion (A/A) collision, and the denominator is the yield in a proton-proton collision. The prefactor $N_{coll}(c)$ is a normalization factor such that the ratio is per binary collision. This normalization factor is a function of centrality $c$ since central collisions with a larger overlap will have more participating nucleons.

While we are introducing the $R_{AA}$ metric in the context of jets, it can and will be used with the yield of other products, namely charged hadrons as $R_{AA}^{ch}$.

### 4.1.3 Jet Energy Loss

#### 4.1.3.1 MARTINI

To simulate the interactions of jets with the QGP medium, we utilize a software called MARTINI [26], which stands for Modular Algorithm for Relativistic Treatment of heavy IoN Interactions. Utilizing a prior computed hydrodynamic evolution of QGP, MARTINI evolves the jets resulting from PYTHIA [25]. MARTINI is a Monte-Carlo event generator, which when performing statistically enough simulations, will provide a distributions of observables which can then be compared with experimental results. In those simulations, a chain of probabilistic events will make each event likely different from any other.

Amongst those probabilistic events are the inelastic processes resulting from the soft interactions with the QGP medium. At each time step, the partons have a finite probability of either emitting or absorbing a particle to or from the medium. MARTINI considers these events as follows: At each time step, the total probability for any of the quantum mechanically allowed processes to happen is computed, according to one of the jet evolution formalisms described in the next section. An event is then sampled with the computed total probability.

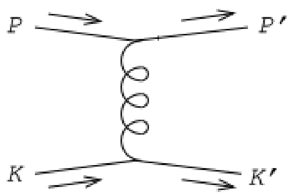

**Figure 4.1:** t-channel scattering by gauge boson exchange. Taken from [9].

If it is determined that a process will happen, another sampling is done. A probability is attributed to each of the allowed processes, based on their relative rates. Finally, once it is determined that a process is happening, and which process it is, another sampling is done, to determine the magnitude of the process. This magnitude is the fraction of the momentum carried by the daughter particle in the case of a radiation, or simply the momentum of the absorbed particle in the case of an absorption.

#### 4.1.3.2 Elastic Energy Loss

Elastic $2 \to 2$ collisions, such as t-channel boson exchange can occur with any charged hard particle, whether it be fermions, or gauge bosons. An example of this would be a coulomb-like interaction as in fig. 4.1, with scattering angle $\theta$ and momentum transfer $q$. The differential scattering in this case would be

$$d\Gamma \sim g^4 T^3 \frac{dq}{q^3} \sim g^4 T \frac{d\theta}{\theta^3} \tag{4.4}$$

Alternatively, the type of the particles involved in the interaction could change, as in the $q\bar{q} \to gg$ case in fig 4.2. This $2 \to 2$ type of interaction with momentum transfer $|\mathbf{q}| \sim gT$ has a mean free path of order $\tau = (g^4T)^{-1}$ [9], of the same order or magnitude as the large angle elastic scattering. Similarly, crossed s-channel versions of figs. 4.2 & 4.1 also have a mean free path of $\tau = (g^4T)^{-1}$. As a result, all $2 \to 2$ must be considered. The AMY formalism [9] proposes an effective kinetic theory which encompasses all elastic and inelastic quarks and gluons' scattering processes to leading order in $gT$, assuming

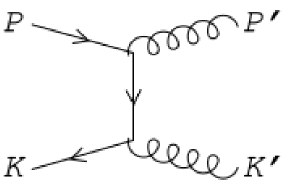

**Figure 4.2:** t-channel scattering for the $q\bar{q} \to gg$ process. Taken from [9].

$g \ll 1$ . The effective theory is assembled from an effective Boltzmann equation for each of the relevant processes, with an effective Boltzmann transport equations of the following form:

$$(\partial_t + \mathbf{v} \cdot \nabla_{\mathbf{x}})f = -C[f] \tag{4.5}$$

where $f = f(\mathbf{x}, \mathbf{p}, t)$ represents the phase space density of particles at position $x$, time $t$, and momentum associated with velocity $\mathbf{v}$. The collision term $C[f]$ represents the rate of processes at which particles are scattered out of momentum $\mathbf{p}$ minus the rate at which they are scattered into this state. The required processes to consider to construct a leading order theory are $2 \to 2$ and $1 \leftrightarrow 2$ processes.

#### 4.1.3.3 Radiative Energy Loss

While propagating through QGP, two physical formalisms are proposed to describe the radiative rates of partons in QGP. The first formalism is known as the AMY formalism [9], and is independent of the time since the birth of the parton. The second formalism, proposed by Caron-Huot & Gale (CH-G) extends the AMY formalism to include the time elapsed since the birth of the particles in their calculations [10]. We then explore an approximation of the CH-G formalism with bounds fitted on CH-G calculations, based on brownian motion [11]. We complete this chapter with an attempt to closely follow the CH-G with the use of neural networks.

**Landau-Pomeranchuk-Migdal Effect** Hard partons in QGP can undergo effective $1 \to 2$ splittings, which would not be possible in a vacuum due to energy-momentum conservation. For example, a hard quark could undergo a soft collision of momentum transfer $q \sim gT$ with a thermal quark from the medium, and result in a hard quark and a hard gluon radiation, both separated by a small angle of order $g$.

This formation time is of the same order as the mean free path from soft scatterings. As such, soft scatterings could occur during the formation time, specially for high momentum partons where the formation time exceeds the mean free path of soft scatterings. The consideration of these soft scatterings is known as the LPM effect. [40,41]

**Arnold-Moore-Yaffe** The AMY formalism gives the radiative rates as follows in [9,42]:

$$\frac{d\Gamma(p,k)}{dk} = \frac{C_s g_s^2}{16\pi p^7}\frac{1}{1 \pm e^{-k/T}}\frac{1}{1 \pm e^{(p-k)/T}} \times f(x) \times \int \frac{d^2\mathbf{h}}{(2\pi)^2} 2\mathbf{h} \cdot Re\mathbf{F}(\mathbf{h},p,k) \tag{4.6}$$

where

$$f(x) = \left\{ \begin{array}{ll} \frac{1+(1-x)^2}{x^3(1-x)^2} & \text{for } q \to qg \\ N_f \frac{x^2+(1-x)^2}{x^2(1-x)^2} & \text{for } g \to qq \\ \frac{1+x^4+(1-x)^4}{x^3(1-x)^3} & \text{for } g \to gg \end{array} \right\} \tag{4.7}$$

Where $p$ is the parton's initial energy, $k$ is the daughter particle momentum energy, and $x = k/p$. **h** is the two-dimensional vector perpendicular to the initial partons's momentum, and **F** is the solution to the following integral;

$$\begin{gathered} 2\mathbf{h} = i\delta E(\mathbf{h},p,k)\mathbf{F}(\mathbf{h},p,k) + g^2 \int \frac{d^2\mathbf{q}_\perp}{(2\pi)^2} \frac{m_D^2}{q_\perp^2(q_\perp^2 + m_D^2)} \times \\ \{(C_s - C_A/2)[\mathbf{F}(\mathbf{h}) - \mathbf{F}(\mathbf{h} - k\mathbf{q}_\perp)] + \\ (C_A/2)[\mathbf{F}(\mathbf{h}) - \mathbf{F}(\mathbf{h} + p\mathbf{q}_\perp)] + \\ (C_A/2)[\mathbf{F}(\mathbf{h}) - \mathbf{F}(\mathbf{h} - (p-k)\mathbf{q}_\perp)]\} \end{gathered} \tag{4.8}$$

$$m_D^2 = \frac{1}{6}(2N_c + N_f)g^2T^2 \tag{4.9}$$

$$\delta E = \frac{|\mathbf{h}|^2}{2pk(p-k)} + \frac{m_k^2}{2k} + \frac{m_{p-k}^2}{2(p-k)} - \frac{m_p^2}{2p} \tag{4.10}$$

where

$$m^2 = \begin{cases} \frac{g^2T^2}{12}(2N_c + N_f) & \text{for gluons} \\ \frac{g^2T^2}{3} & \text{for quarks/antiquarks} \end{cases} \tag{4.11}$$

**4.1.3.3.1 Caron-Huot & Gale** While the AMY formalism accurately describes the radiation rate from a parton in an infinite, static medium. The Caron-Huot & Gale formalism [10] suggests an extension upon it, which includes a more local treatment of the LPM effect.

Starting off with the total probability for a parton $a$, produced at time $t = 0$ with energy $p$, to emit a pair $(b, c)$ of partons with energies $(k, p-k)$ taken from Zakharov [43];

$$\frac{dP^a_{bc}(t)}{dk} = \frac{P^{a(0)}_{bc}(x)}{\pi p} \times \text{Re} \int_0^\infty dt_1 \int_{t_1}^\infty dt_2 \frac{\partial}{\partial \mathbf{x}} \cdot \frac{\partial}{\partial \mathbf{y}} [K(t_2, \mathbf{x}; t_1, \mathbf{y}) - vac]_{\mathbf{x}=\mathbf{y}=0} \tag{4.12}$$

Where $K(t_2, \mathbf{x}; t_1, \mathbf{y})$ is $[p/(2k(p-k))]^2$ times the propagator associated with the following light cone Hamiltonian:

$$H = \delta E - iC_3 \tag{4.13}$$

$$\delta E = \frac{p|\mathbf{p}|^2}{2k(p-k)} + \frac{m_k^2}{2k} + \frac{m_{p-k}^2}{2(p-k)} - \frac{m_p^2}{2p} \tag{4.14}$$

$$C_3 = \frac{C_b + C_c + C_a}{2} v_2(\mathbf{x}) + \frac{C_a + C_c - C_b}{2} v_2(\frac{k}{p}\mathbf{x}) + \frac{C_a + C_b - C_c}{2} v_2(\frac{p-k}{p}\mathbf{x}) \tag{4.15}$$

Where the $m$'s and $C$'s are the masses and Casimirs of the corresponding particles, the $v_2(x)$ terms are the dipole cross-sections of a color singlet $q\bar{q}$ pair [44], and the $vac$ term in the integral corresponds to the vacuum fluctuations. The following $P^{a(0)}_{bc}(x)$ term corresponds to the DGLAP [45] splitting kernels for the corresponding processes, where

$x = k/p$.

$$P^{a(0)}_{bc}(x) = \begin{cases} g^2 C_A \frac{1+(1-x)^2}{x}, & q \to gq \\ g^2 C_F \frac{1+x^4+(1-x)^4}{x(1-x)}, & g \to gg \\ 2g^2 N_f T_f \frac{x^2+(1-x)^2}{x^2(1-x)^2}, & g \to q\bar{q} \end{cases} \tag{4.16}$$

We now take a Fourier transform of Eq. 4.12 with the goal of removing non-compact time integrations, and obtain the following:

$$\frac{dP^a_{bc}(t)}{dk} = \frac{P^{a(0)}_{bc}(x)}{\pi p} \times \mathrm{Re} \int_0^\infty dt_1 \int_{t_1}^\infty dt_2 \mathbf{q} \cdot \mathbf{p} [K(t_1, \mathbf{q}; t_2, \mathbf{p}) - vac] \tag{4.17}$$

$$\begin{gathered} C\psi(\mathbf{p}) = \int_\mathbf{q} C(\mathbf{q}) \{ \frac{C_b+C_c+C_a}{2} [\psi(\mathbf{p}) - \psi(\mathbf{p}-\mathbf{q})] + \\ \frac{C_a+C_c-C_b}{2} [\psi(\mathbf{p}) - \psi(\mathbf{p}+\frac{k}{p}\mathbf{q})] + \\ \frac{C_a+C_b-C_c}{2} [\psi(\mathbf{p}) - \psi(\mathbf{p}+\frac{p-k}{p}\mathbf{q})] \} \end{gathered} \tag{4.18}$$

using $v_2 = \int_\mathbf{q} C(\mathbf{q})(1-e^{i\mathbf{q}\cdot\mathbf{x}})$ and $\mathbf{p}$ is transverse to $\mathbf{q}$. After integrating by parts, we arrive at

$$\frac{dP^a_{bc}(t)}{dk} = \frac{P^{a(0)}_{bc}(x)}{\pi p} \times \mathrm{Re} \int_0^\infty dt \int_0^t dt_1 \int_{\mathbf{q},\mathbf{p}} \frac{i\mathbf{q}\cdot\mathbf{p}}{\delta E(q)} C(t) K(t, \mathbf{q}; t_1, \mathbf{p}) \tag{4.19}$$

out of which we can take a time derivative and arrive at the final form:

$$\frac{d\Gamma^a_{bc}(t)}{dk} = \frac{P^{a(0)}_{bc}(x)}{\pi p} \times \mathrm{Re} \int_0^t dt_1 \int_{q,p} \frac{i\mathbf{q}\cdot\mathbf{p}}{\delta E(q)} C(t) K(t, \mathbf{q}; t_1, \mathbf{p}) \tag{4.20}$$

Eq. 4.20 has the benefit of containing a single, finite, time integral, which we can solve numerically in $t$, the time elapsed between the current time and the time since the last emission. Moreover, taking the limit $t \to \infty$ recovers the result from AMY in Eq. 4.6 as it should. This is because the AMY radiative rates are calculated from Feynman diagram integrals, which result in the average probability of a radiation over all time, taking the birth of the parton at $t_1 = -\infty$ and the radiation to happen at $t_2 = \infty$. The CH-G radiative rate, however, takes $t = t_2 - t_1$ to be finite, and the interferences such as the LPM effect are included in the propagator $K$.

$$\frac{1}{l_{mfp}} \equiv \int d^3k\rho(k) \int dq^2(1-\cos(\theta_{pk}))\frac{d\sigma_{el}}{dq^2} \tag{4.21}$$

$$\frac{d\sigma_{el}}{dq^2} \sim C_R\frac{2\alpha_s^2}{(q^2)^2} \tag{4.22}$$

$m_D^2 = \frac{3}{2}g^2T^2$

$$\begin{aligned} \frac{1}{l_{mfp}} &\sim k^3 \int_{m_D^2}^{\infty} d^2q\frac{\alpha_s^2}{(q^2)^2} \\ &\sim T^3\frac{\alpha_s^2}{m_D^2} \\ &\sim g^2T \end{aligned} \tag{4.23}$$

resulting in $l_{mfp} \sim 1/g^2T$

$$l_\perp \gtrsim 1/q_\perp \tag{4.24}$$

$$\frac{q_\perp}{q} \sim \frac{l_\perp}{l_{coh}} \tag{4.25}$$

$$l_{coh} \sim \frac{q}{q_\perp}l_\perp \sim \frac{q}{q_\perp^2} \tag{4.26}$$

In fig. 4.3 the differential decay rates are compared for the same set of parameters with the AMY and CHG formalisms. As expected, the AMY calculation is the long-time limit of the CHG formalism, the latter being decreasingly suppressed as the time since birth is increased.

In this thesis, we will tackle the computation of this formulation, Eq. 4.20, into our simulations. However, a first approximation has been done in Chanwook Park's PhD. thesis to utilize an upper bound on these rates.

**4.1.3.3.2 Random Walks** The physical separation of a radiated particle is ambiguous considering its formation time, and thus the mother and daughter particles do not interact with the medium independently as discussed in sec. 4.1.3.3. To take this into consideration, we utilize the AMY limit results as an upper bound for the radiative rates, and

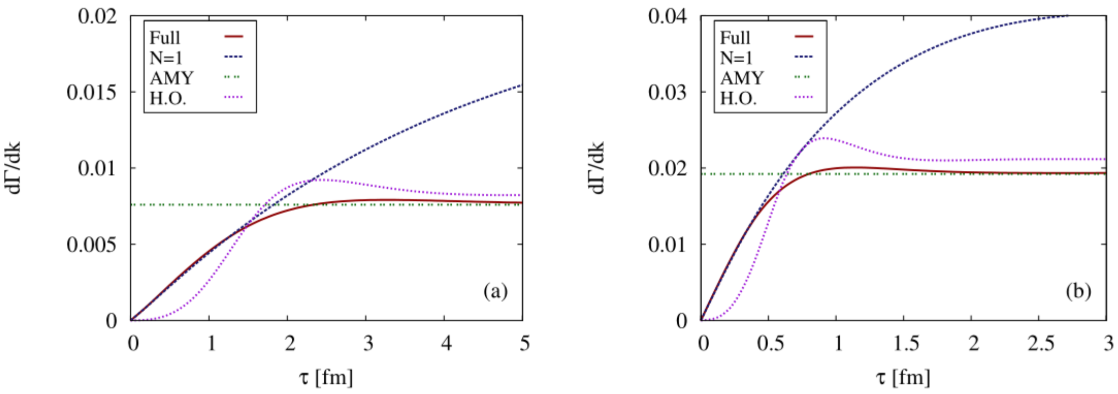

**Figure 4.3:** Differential decay rate for a 3 GeV gluon from a 16 GeV parent quark as a function of traveled length since the birth of the jet. The green dotted line is the AMY formalism limit, while the solid red line is the CH-G formalism calculation. The medium is a uniform brick of QGP at T = 0.2 GeV (panel (a)) and T = 0.4 GeV (panel (b)). Taken from [10].

suppress those introducing a separation condition, using the quantum mechanical uncertainty

$$\Delta r_\perp \Delta p_\perp > C(k,T) \tag{4.27}$$

where the minimal uncertainty is parametrized as follows:

$$C(k,T) = 0.25 \times (k/T)^{0.11} \tag{4.28}$$

to imitate the results from Eq. 4.20 and obtain a finite-size effect. Since the radiative processes in our simulations utilizing MARTINI are purely colinear, the transverse distance $\Delta r_\perp$ and momentum $\Delta p_\perp$ result from the random walk response to the soft scatterings with the medium. One can make sense of Eq. 4.28 as a parton with a larger $k$ should be less sensitive to the medium and induce a looser condition. Inversely, a higher temperature medium should separate the partons faster, driving the condition to zero, and allow radiation again right away. In fig. 4.4, the AMY formalism, the CH-G formalism, and the RW approximations are contrasted. As expected, again the AMY formalism is the long-time limit of the CH-G formalism, while the CH-G formalism is decreasingly suppressed

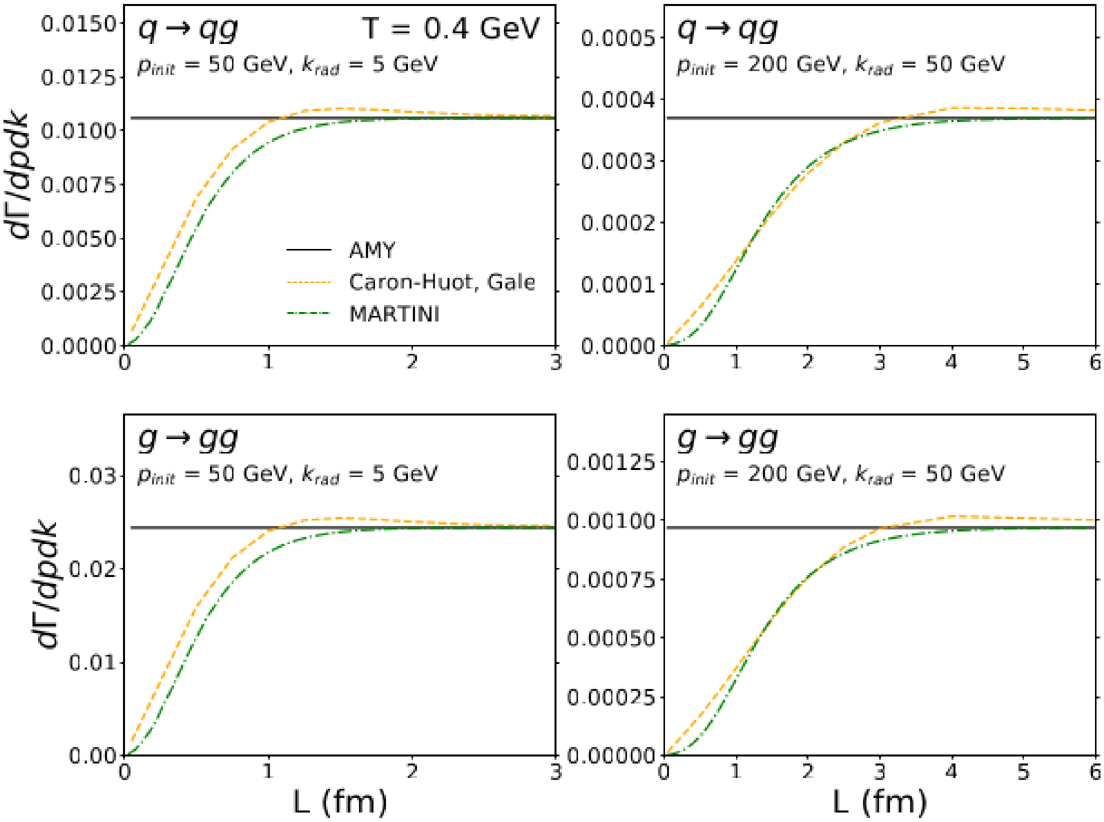

**Figure 4.4:** In the left column we have the differential decay rates for a parent particle with momentum of $50$ GeV, radiating a gluon of $5$ GeV as a function of travelled length since the birth of the parton. In the right column we have the differential decay rates for a parent particle with momentum of $200$ GeV, radiating a gluon of $50$ GeV. In the top row, the parent particle is a quark, and in the bottom row the parent particle is a gluon. The medium is a uniform brick of QGP at $T = 0.4$ GeV everywhere. The solid black line is the AMY limit, the yellow dotted line is the CHG formalism calculation, and the green dotted line is the fitting from the random walks labeled MARTINI. Taken from [11].

as the time since birth increases. In the green dotted line, the RW approximation bounds the CH-G formalism from below as it should.

#### 4.1.3.4 Effective Kinetic Theory

Going back to our Boltzmann equations formulation in Eq. 4.5, we can aggregate the collision terms visited in the previous section as follows:

$$(\partial_t + \mathbf{p}\cdot\nabla_{\mathbf{x}})f_s(\mathbf{x},\mathbf{p},t) = -C_s^{2\leftrightarrow 2}[f] - -C_s^{1\leftrightarrow 2}[f] \tag{4.29}$$

Where the $2 \rightarrow 2$ processes unfold as such:

$$\begin{aligned} C_a^{2\leftrightarrow 2}[f] =& \frac{1}{4|\mathbf{p}|\nu_a}\sum_{bcd}\int_{\mathbf{k}\,\mathbf{p'}\,\mathbf{k'}} |M^{ab}_{cd}(\mathbf{p},\mathbf{k};\mathbf{p'},\mathbf{k'})|^2(2\pi)^4\delta^4(P+K-P'-K')\times \\ & \times\{f_a(\mathbf{p})f_b(\mathbf{k})[1\pm f_c(\mathbf{p'})][1\pm f_d(\mathbf{k'})] - f_c(\mathbf{p'})f_d(\mathbf{k'})[1\pm f_c(\mathbf{p})][1\pm f_d(\mathbf{k})]\} \end{aligned} \tag{4.30}$$

where the latin letters correspond to the species index, $\nu_a$ corresponds to the of spin times the number of color states for species $a$, capital letters correspond to four-vectors, $f$ functions correspond to the Fermi-Dirac or Bose-Einstein occupation functions, $M^{ab}_{cd}$ is the effective matrix element of the processes $ab \leftrightarrow cd$, and the integrals are understood to be Lorentz invariant momentum integrations $\int_p = \int \frac{d^3p}{2|p|(2\pi)^3}$. We note that symmetry under particle exchange and time reversal allows for the matrix element squared and energy-momentum conservation delta function to be factored out, while the last two terms correspond to the $ab \rightarrow cd$ and $cd \rightarrow ab$ processes respectively.

As for the $1 \rightarrow 2$ the collision term unfolds analogously:

$$\begin{aligned} C_a^{1\leftrightarrow 2}[f] =& \frac{1}{4|\mathbf{p}|\nu_a}\sum_{bc}\int_{\mathbf{p'}\,\mathbf{k'}} |M^{a}_{cd}(\mathbf{p};\mathbf{p'},\mathbf{k'})|^2(2\pi)^4\delta^4(P-P'-K')\times \\ & \times\{f_a(\mathbf{p})[1\pm f_b(\mathbf{p'})][1\pm f_c(\mathbf{k'})] - f_b(\mathbf{p'})f_c(\mathbf{k'})[1\pm f_a(\mathbf{p})][1\pm f_d(\mathbf{k})]\}+ \\ & + \frac{1}{2|\mathbf{p}|\nu_a}\sum_{bc}\int_{\mathbf{k}\,\mathbf{p'}} |M^{c}_{ab}(\mathbf{p'};\mathbf{p},\mathbf{k})|^2(2\pi)^4\delta^4(P+K-P')\times \\ & \times\{f_a(\mathbf{p})f_b(\mathbf{k})[1\pm f_c(\mathbf{p'})] - f_c(\mathbf{p'})[1\pm f_a(\mathbf{p})][1\pm f_b(\mathbf{k})]\} \end{aligned} \tag{4.31}$$

in massless kinematics, resulting in strictly collinear radiations. For the nearly-collinear case, one can integrate over the transverse momentum associated with the splitting.

## 4.2 Methods

Before comparing the different approaches of jet evolution in QGP, we first verify that, without any medium, the simulation framework provides reliable results. We then simulate pp collisions at $\sqrt{s} = 2.76$ TeV and compile the distributions for the charged hadron multiplicity and the jet cross section, as no QGP is believed to be formed in p-p collisions. The PYTHIA tune parameter we use is 15, with the parton distribution function CTEQ6L1 from Les Houches Accord parton distribution function [46]. The $p_T$ ranges are separated into 40 bins, with the following values: $4, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25,$ $27, 29, 32, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 100, 110, 120, 130, 140, 150, 160, 170, 180,$ $190, 22, 225, 250$ and $300$ as range limits, in GeV. In our simulations, the strong coupling $\alpha_s$ is fixed at 0.3, as an upper bound to fig. 2.2. A more physical approach would be to have a running strong coupling, matching fig. 2.2. However, we chose to fix the coupling to better distinguish the differences resulting from the chosen jet decay approaches. The pseudorapidity range for Jets used is $|\eta| < 2$, and $|\eta| < 1$ for charged hadrons.

We then simulate PbPb collisions at $\sqrt{s} = 2.76$ TeV at the centrality classes 0-5%, 5-10% and 10-20%. To that effect, we take the parton distributions from PYTHIA, then evolve the parton distribution with MARTINI, with the previously computed hydrodynamic background from MUSIC. For the evolution within the medium, the time step size taken is 0.002 fm, and take the nuclear parton distribution function EPS09LO [47]. For both the pp and the PbPb simulations, we compute 1000 events for each of the 40 $p_T$ bin, for 50 different hydrodynamic evolutions. In the PbPb simulations, we do so for each of the three centrality classes, resulting in 8 million independent simulations. For each centrality class, for each hydrodynamic evolution, the $p_T$ bins are weighted by their corresponding cross section to span the full $p_T$ range in the figures in this section. The hydrodynamic runs are then averaged, to yield the total cross section for each centrality bin.

#### 4.2.0.1 Jet evolution

We now numerically implement the Caron-Huot & Gale [10] Jet decay formalism with the MARTINI framework. For each time step in the simulation, for every parton, we first utilize the total decay rate $d\Gamma$ for each of the following processes:

$$\begin{aligned} q &\to gq \\ g &\to qq \\ g &\to gg \end{aligned} \tag{4.32}$$

to determine whether a process will take place. Once it is established that one of the processes takes place, another sampling is done to determine which of the processes takes place.

Implementation wise, the main difference between the Caron-Huot & Gale formalism and the AMY formalism is the dependence on the time since the birth of the particle $t$, and the temperature of the medium in which the particle is $T$. In practice, the QGP medium is discretized, and every QGP site can have a different temperature. This, in turn, causes the time to be dilated differently in each of the fluid cells. This should be taken into account, but would require us to know the the exact path of each parton, and the temperature of each of the sites the parton was at the time the parton was there. With that information, one could Lorentz-transform the time to properly correct the curve in 4.3 which assumes a uniform brick. Unfortunately, we do not have that information, and as an approximation we utilize the proper time of the jet in our calculations.

### 4.2.1 Data Generation

In this section we implement the Caron-Huot & Gale formalism, where the rates are calculated in a uniform brick of QGP. $d\Gamma/dK$ here is a function of the time $t$ elapsed, since the last emission, of the temperature of the brick $T$, and of the mother and daughter particles' momentum, $P$ and $K$. Our first step is to compute the decay rates by implementing the calculation numerically.

We systematically chose the ranges at which the aforementioned variables are calculated. The first range we chose is the time variable which we set between 0 and $30fm/c$, as it is the typical lifespan of QGP, as mentioned in the introduction 2.1.2. We now estimate realistic values for the mother particle momentum $P$. Since we will limit our experiments to the LHC experiments at $\sqrt{s} = 2760$ GeV, an upper bound estimate would be to divide the center of mass energy between the two protons, and then by three because on average each of the quarks in the protons have a third of the energy. This leads us to $\frac{\sqrt{s}}{2\times 3} \approx 1000$ GeV. For each value of $P$, we generate $K$ values in increments of 5% of $P$, from 5% to 90%, and we use increments of 1% between zero and 5%, and between 90 and 99% for integration reasons. Finally, temperature is taken in increments of 50 MeV between 100 MeV and 1 GeV, as it is the range of temperature of QGP.

Once we have the previous parameters for $d\Gamma/dK = d\Gamma/dK(P, K, t, T)$, we integrate those values over $K$ between 0 and 100%, to obtain the total decay rate $\Gamma$. We proceed to do so utilizing the Simpson's integration routine, and obtain numerical values for $\Gamma = \Gamma(P, t, T)$.

### 4.2.2 Modeling

We are now ready to train 8 different MLP's, for each of the processes in 4.32, four for the differential rate $d\Gamma/dK$ and another for the total probability of the event $\Gamma$.

For each of the processes, we first hard-code a transformation function which takes in the input vector $(p, k, t, T)$ or $(p, k, t, T)$ for the $d\Gamma/dK$ and the $\Gamma$ models respectively, that

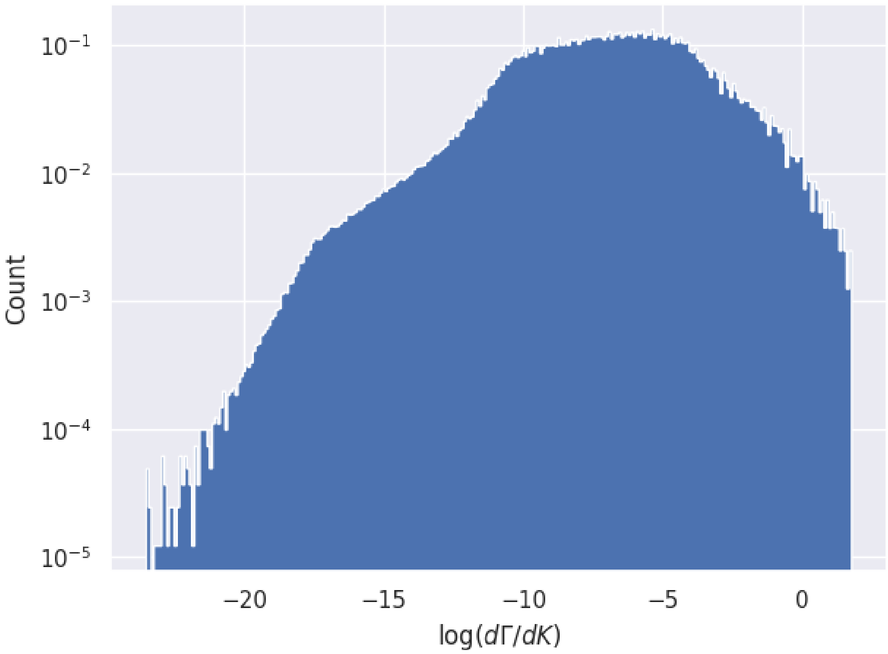

**Figure 4.5:** Distribution of the logarithm of differential decay rates $(d\Gamma/dK)$ (dimensionless) for the $g \to gg$ process, across all considered parameter sets described in sec. 4.2.1. The counts have been normalized such that the integral of the distribution is one.

maps each of the values to a uniform distribution within $(0, 1)$, utilizing a logarithmic or linear transformation depending on how each the data for the corresponding variable had been generated.

We then forward the normalized inputs in an MLP of 6 and 4 layers, for the $d\Gamma/dK$ and $\Gamma$ respectively, utilizing the ReLU activation functions respectively. For the optimizing algorithm, we chose to use AMSGrad as explained in section 2.3.4, with a learning rate $\alpha = 10^{-3}$ for 100 training epochs.

Given that both $\Gamma$ and $d\Gamma/dK$ are decay rates and differential decay rates, they must by their nature be strictly positive. A visualization of the distribution of the logarithm of the differential decay rates, as in fig 4.5, show us that these rates span several orders of magnitude. We utilize this prior knowledge about the target values for the choice of our loss function. While the metric that we utilize to evaluate and choose our models is the percent accuracy, the loss that we explicitly minimize in our training algorithm is the percent error of the logarithm of the decay and differential decay rates. This is equivalent to have the model learn the exponent in base 10 rather than the rates explicitly.

Regarding the training, validation, and test data splits, we first consider the naive approach to shuffle all data points $d\Gamma/dK = d\Gamma/dK(P, K, T, t)$, and sample from that distribution. However, doing so results in a training set with data points potentially from every single PKT combination, which would be undesirable, since we want our models to be able to generalize for different PKT combinations. Therefore, the approach we take is to first split all PKT combinations, sample a training, validation, and test set from there, and then recover the time steps for all of them. This way, the model chosen based on validation performance is tested on PKT combinations it has never seen before. Similarly for the $\Gamma$ model, we sample from the $\Gamma = \Gamma(P, T)$ and recover the time steps afterwards. Utilizing this sampling procedure, we take 70% of the data points for training, 20% for validation and 10% for the test set.

A choice has to be made regarding how to train the integrating $\Gamma$ model. Indeed, the training data set could be either the result of integrating the $d\Gamma/dK$ from the CH-G calculations, or from the previous neural network which emulates the former. We chose to train the $\Gamma$ models on the integral of the $d\Gamma/dK$ utilizing a Gaussian quadrature, with 50 evaluation points between $0.01 \leq K/P \leq 0.99$.

# Chapter 5

# Results & Discussions

The initial conditions of the color gauge fields are required to begin a hydrodynamic simulation of QGP. We show the results of training a machine learning model on an iterative solution, to then be used as a starting Ansatz for the same iterative approach on different samples and quantify the reduction in iterations. We consider two different models, and two different spatial resolutions, to make a comparison on training in a coarser resolution and inferring on a finer resolution. This section then builds on the work by Marius Cautun, Francois Fillion-Gourdeau, and Sangyong Jeon in [35].

Jets are aggregates of particles which can serve as a probe for QGP when studying their energy exchange with the medium. In that regard, this thesis follows on the work by C. Park [11], where a comparative study was done considering the radiation rates from AMY independent of formation time, then emulated the formation time dependent radiative rates from CH-G in a Brownian motion approach. We will compare both those approaches, as well as modelling the CH-G formalism using a neural Network. We will then compare all three approaches with the experimental data from CMS.

## 5.1 Initial Conditions

We train a few machine learning models to generate a new ansatz for replacing Eq. 3.16 as explained in sec. 3. While we require $N = 500$ to have an appropriate grid size such that the inter-lattice point spacing $a$ resolves the energies required, we also consider a coarser grained resolution to explore the possibility of training a network on coarse data and then infer on finer resolution for computational efficiency. The two resolutions we chose are then $128 \times 128$ and $512 \times 512$ pixels, each with their 8 channels for the gluon color gauge fields. Models resulting from both those training resolutions are then tested on both resolutions.

In figures 5.1 and 5.2 we have an example of a 4 layer CNN network with 32 hidden dimensions and a kernel size of 5. In both figures, the image on the left is the prediction from the model trained on $128 \times 128$ data, the middle image is from the model trained on $512 \times 512$ data, and the image on the right is the target image.

In both cases, we can see the presence of most spatial features and sign of the average fields. However, there are some visible differences regarding the magnitude of the fields. Both networks, as expected, seem to perform better on the resolution at which they were trained. The model trained on $512 \times 512$ data seems to predict a lower intensity on the $128 \times 128$ data. Conversely, the model trained on $128 \times 128$ data seems to predict a higher intensity on the $512 \times 512$ data. This is because the models are restricting themselves to the target distribution on which they were trained. In a coarser grid, we would expect a higher energy density per pixel, while on a finer grid we would expect a lower energy density per pixel.

Nevertheless, while our attempt at having a model generalize across resolutions fails to reproduce with precision the intensities, it still manages to reproduce the spatial features. We can still utilize the model trained on $512 \times 512$ data to infer $512 \times 512$ data, and obtain the desired results. Despite still having some intensity discrepancies, those are of minor concern given how we use the inference from the model. Rather than ”believing“

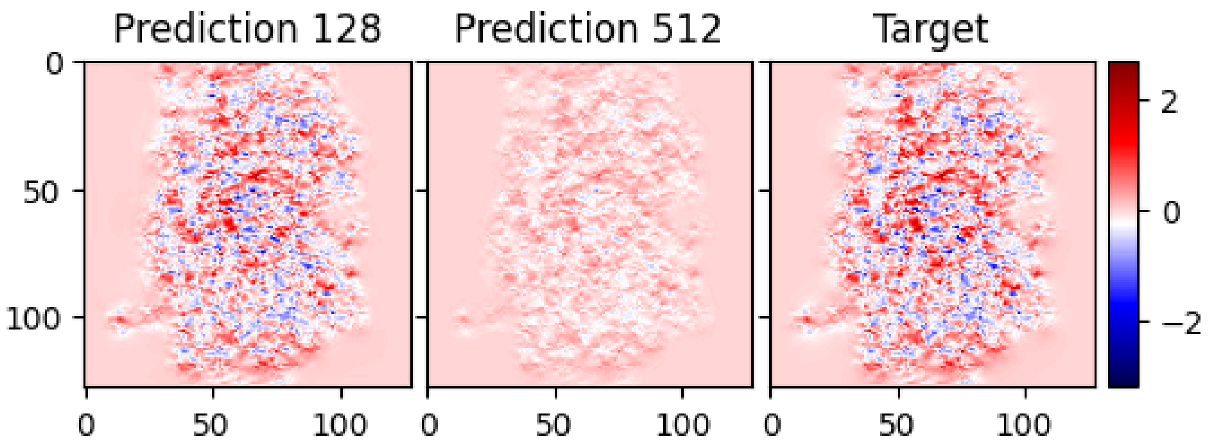

**Figure 5.1:** The image on the right is the average field strength across colors of the initial color gauge fields in a resolution of $128 \times 128$ pixels. The image on the left is the prediction of the same quantity in a $128 \times 128$ pixel resolution from the model trained on data with a resolution of $128 \times 128$ pixels, and the middle image is the prediction of the same quantity in a $128 \times 128$ pixel resolution from the model trained on data with a resolution of $512 \times 512$ pixels. In all cases, the x-axis and y-axis quantities corresponds to the pixel index in either dimension, and the strength of the fields is given by the color of the pixel, following the scale on the right. The strength of the field is unit-less, as it has been scaled for a mean of zero and a standard deviation of one.

the model and utilizing the prediction as physically valid, we are utilizing it as a starting point to embark into the iteration described in section 3. We find that, on average, utilizing the neural network's prediction as a starting point results decreases the number of iterations required to attain a solution by a factor of four.

## 5.2 Charged Hadrons & Jets

In this section, we perform simulations of pp and PbPb collisions at $\sqrt{s} = 2.76$ TeV. To that end, we first describe the modeling that we performed on the CH-G calculations to be used in our simulations. We refer to the AMY formalism, the random walk approach to the CH-G, and the neural network approach to the CH-G as the three considered for-

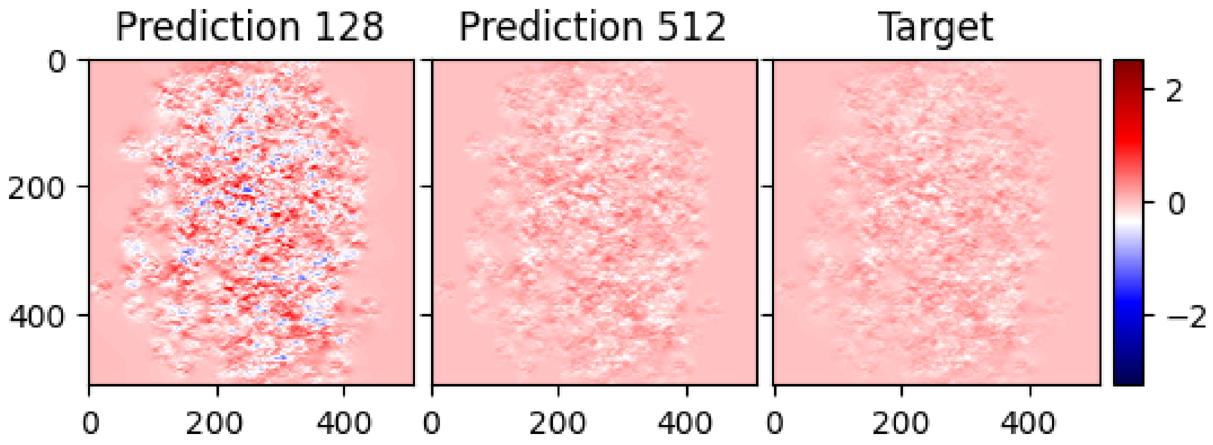

**Figure 5.2:** The image on the right is the average field strength across colors of the initial color gauge fields in a resolution of $512 \times 512$ pixels. The image on the left is the prediction of the same quantity in a $512 \times 512$ pixel resolution from the model trained on data with a resolution of $128 \times 128$ pixels, and the middle image is the prediction of the same quantity in a $512 \times 512$ pixel resolution from the model trained on data with a resolution of $512 \times 512$ pixels. In all cases, the x-axis and y-axis quantities corresponds to the pixel index in either dimension, and the strength of the fields is given by the color of the pixel, following the scale on the right. The strength of the field is unit-less, as it has been scaled for a mean of zero and a standard deviation of one.

malisms, although both the random walk and the neural network approach are approximations to the CH-G physical formalism.

### 5.2.1 Modeling

We first train the neural networks to emulate the CH-G formalism, for each of the processes described in Eq. 4.32 to obtain the differential decay rates $d\Gamma/dK$. In all cases, we utilize a neural network of 6 layers and 64 hidden dimensions. In table 5.1, we report the accuracy obtained on the training, validation and test data sets. We recall that while we do report the training accuracy, the actual loss that is minimized is the percent error of the logarithm of the rates $d\Gamma/dK$, which amounts to learning the exponent of the rate in base 10. These high accuracy scores were expected, since the functions our neural network

| Process | Training | Validation | Test |
|---|---|---|---|
| $q \to qg$ | 97.64 % | 97.66 % | 97.63 % |
| $g \to qq$ | 96.13 % | 95.03 % | 95.98 % |
| $g \to gg$ | 97.43 % | 97.85 % | 97.86 % |

**Table 5.1:** Best accuracy during training, and on the validation and test sets for the differential decay rate MLPs. These accuracies were computed utilizing a trainer class from [15], utilizing the training parameters and data sets described in sections 4.2.2 and 4.2.1 .

are learning are simple, and can be approximate by a somewhat linear rise followed by a plateau. In that respect, the model's main challenge is to learn the scale of the rise and the plateau, for each varying parameter. We attribute the overall lower performance on the $g \to qq$ process across all data sets to the fact that there is more variance in the scale of the rates of this process.

In figure 5.3, we show an example of the prediction of the neural network for the differential decay rate of a $g \to gg$ process as a function of time elapsed since the birth of the parton. As we can see, the neural network overall seems to approximate the CH-G calculation accurately, with an average error of 2.11 %. The region where it seems to be less accurate is right before reaching the plateau, where it does not capture a slight rise before stabilizing. This might be attributed to the fact that the model is simple, with only 6 layers. A more complex model might have captured this rise before stabilizing, with a higher risk of over-fitting.

In figure 5.4, we have the same model as in fig. 5.3 with the rate $d\Gamma/dK$ as a function of $k/p$ instead of a function of time since birth. While this is only another visualisation of the same neural network predictions along a different dimension, this curve is of interest as its integral represents the total radiation rate of a particle at a given time, regardless of the daughter particle's energy. We thus we compute the $d\Gamma = \int_{k=0}^{k=p} \frac{d\Gamma}{dk} dk$ integrals utilizing the Gaussian quadrature routine [48], on the $d\Gamma/dk$ values from our model.

We then proceed to train a new batch of neural networks for the $d\Gamma = d\Gamma(p, T, t)$ functions for each of the radiative processes, with the integrals from the $d\Gamma/dk$ data. We must

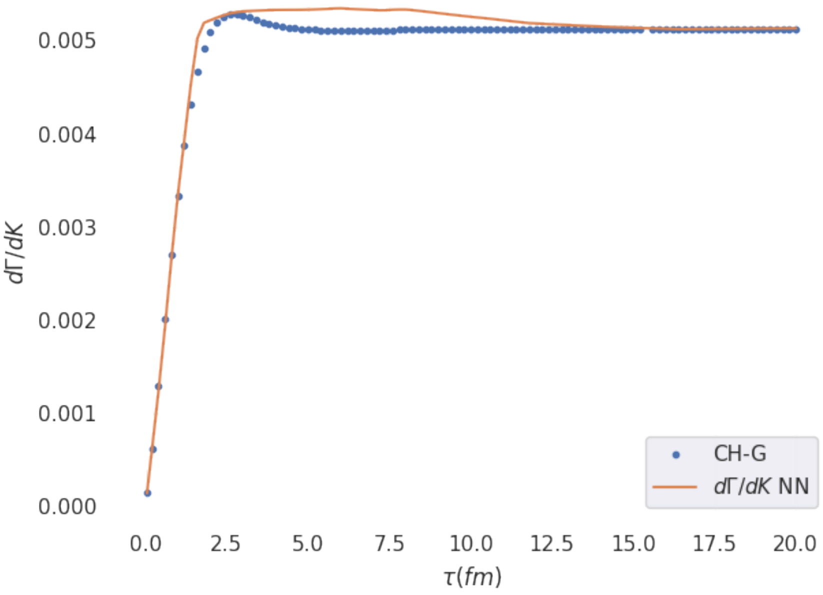

**Figure 5.3:** The neural network approximation for the CH-G formalism in the orange line, and the CH-G calculations as the blue dots, for the $g \rightarrow gg$ process as a function of time, with a 2.11 % average relative error across the time steps shown in the figure. The parent particle energy is $p = 70.7$ GeV, the daughter's energy is $k = 14.14$ GeV, and the temperature is taken to $T = 0.367$ GeV.

now make a choice to either train the total rate $d\Gamma$ models on either the integrals from the CH-G calculations, or from the NN emulating the CH-G calculations. For consistency, we chose to train the model on the NN predictions. In table 5.2 we report the training, validation and test accuracies for the networks used in the rate integrating. Again, we note that the very high accuracies are due to a simple function, with a rise followed by a plateau. The model's main challenge then again is to capture the magnitude of the rates, as well as the transition from the rise to the plateau, across varying parameters. In fig. 5.5 we have an example of the total decay rate as a function of the time since the birth of the parton. At each time point in blue, the differential decay rate from the previous neural network is integrated over $0 < k < p$. The orange curve shows the prediction of the total decay rate from the $d\Gamma$ neural network. Again, we notice that the network does

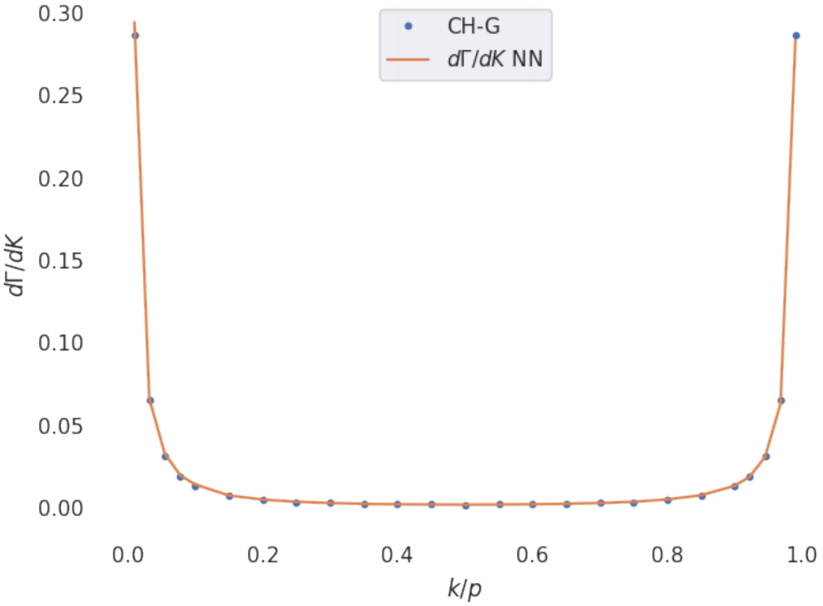

**Figure 5.4:** The neural network approximation for the CH-G formalism in the orange line, and the CH-G calculations as the blue dots, for the $g \rightarrow gg$ process as a function of $k/p$, with a 1.59% error across $0 < k/p < 1$. The parent particle energy is $p = 70.7$ GeV, the time since birth of the parent parton is set to $\tau = 20$ fm, and the temperature is taken to $T = 0.367$ GeV.

| Process | Training | Validation | Test |
|---|---|---|---|
| $q \rightarrow qg$ | 98.01 % | 98.06 % | 97.69 % |
| $g \rightarrow qq$ | 98.31 % | 97.91 % | 97.56 % |
| $g \rightarrow gg$ | 98.56 % | 98.45 % | 97.65 % |

**Table 5.2:** Best accuracy during training, and on the validation and test sets for the total decay rate MLPs. These accuracies were computed utilizing a trainer class from [15], utilizing the training parameters and data sets described in sections 4.2.2 and 4.2.1 .

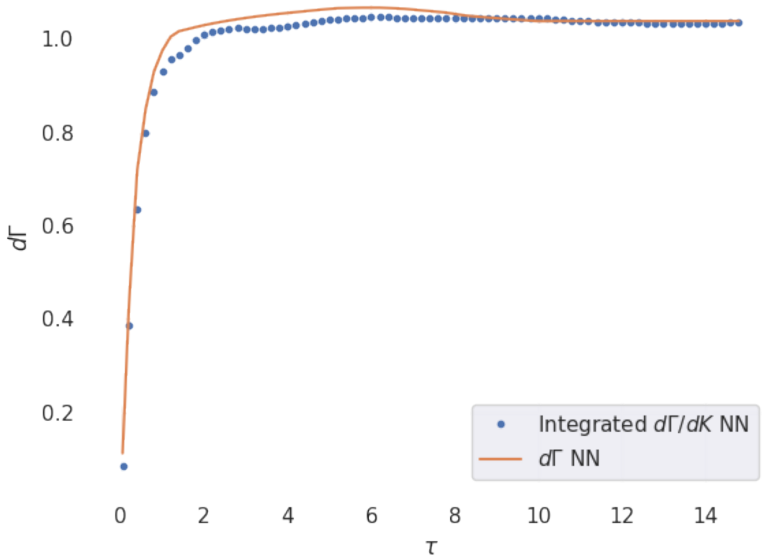

**Figure 5.5:** The neural network model for the total decay rate of the $g \to gg$ process in orange, trained on the differential decay rate neural network integrals as blue points. The parent particle energy is $p = 70.7$ GeV and the temperature is taken to be $T = 0.367$ GeV.

not capture all details of the data, but we prefer to keep a simpler model to avoid a risk of over-fitting.

We recall that the CH-G formalism is a generalization of the AMY formalism, with suppressive corrections at early times. Implementing the corrections, we would then expect less radiations in early time steps, while allowing for more radiations at larger times. While this is not directly observable, it is something we can measure and compare within our simulations, and expect it to then translate to the experimental observables $R_{AA}^{jet}$ and $R_{AA}^{jet}$.

The first row of figures 5.6, 5.7 & 5.8 depict the distribution of the times elapsed since the birth of the parent parton for all radiations, while the columns separate the radiative processes. The second row of the figures 5.6, 5.7 & 5.8 are the ratio of the distributions in the above row, with the AMY formalism as the denominator, and the Random Walk and Neural Network approaches as the numerator. We chose to call this metric *Lifetime* $R_{AA}$ ($R_{AA}^{lt}$). The integral of the first row distributions represent the total radiations per process

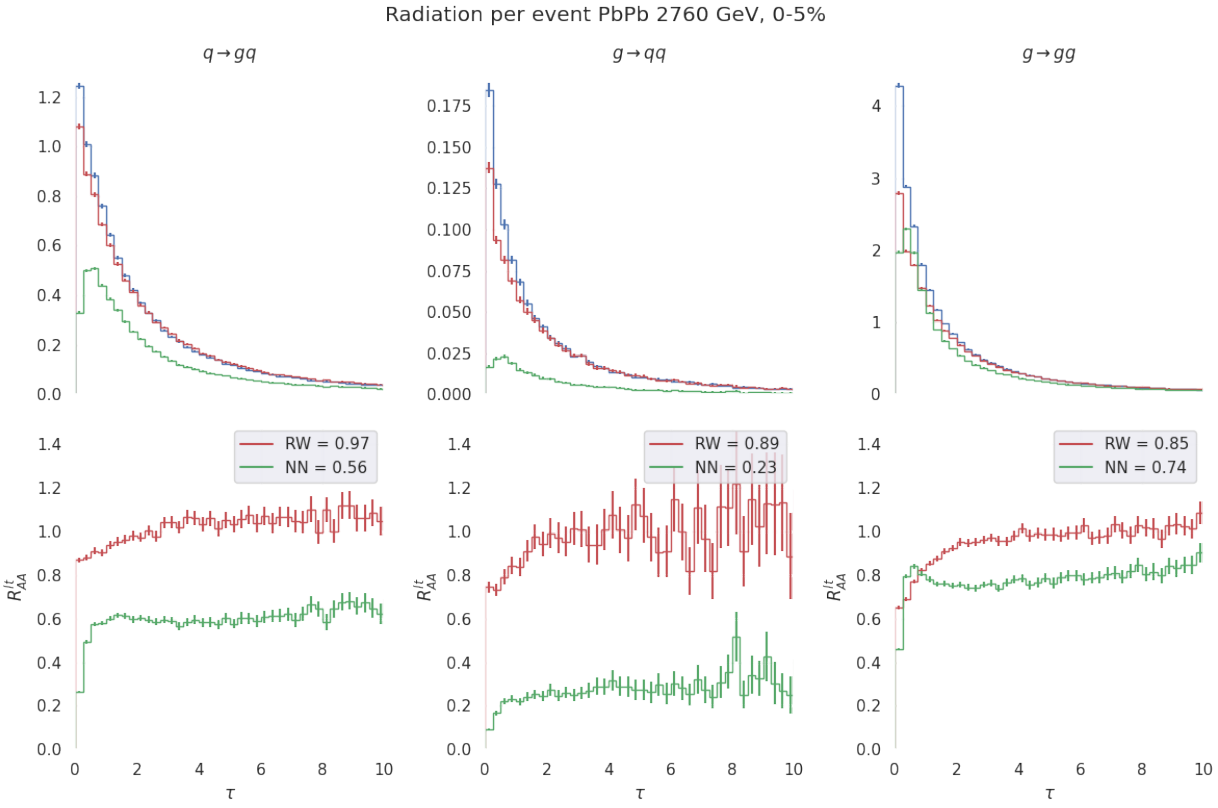

**Figure 5.6:** In the first row, there is the distribution of the time between radiations for each radiative process, for the AMY formalism (in blue), the random walks approach (in red), and the Neural Network approach (in green). In the second row, we have the ratio of the above distributions with the AMY formalism as a denominator. In red, the Random walk approach as the numerator, and in green, the neural network approach. The experiment is a Pb-Pb collision at 2760 GeV, with the centrality bin $0 - 5\%$, and the jet $p_T$ in $[4, 7]$ GeV.

per event for a parent particle within the $p_T$ range of the respective figure, so we compute those integrals from $t = 0$ fm to $t = 30$ fm, which is the time range of the full evolution simulation. The ratio of those integrals are written in the legends of the bottom row, as they represent the reduction of the energy loss as a product of utilizing the suppressed decay rates. These integrals will allow us to see if the reduction of radiations in early time steps are counterbalanced in later time steps.

Within fig. 5.6, the first notable feature is the appearance of a peak at the emission times for all three radiative processes with the NN approach. While the AMY formalism radiated the most particles at the very early time steps of the evolution, the distribution decreases monotonously, or at least the peak cannot be resolved by our histogram bins of width $\tau = 0.25$ fm. The RW approach's radiations also decreases monotonously, while exhibiting less radiations than AMY at the very early time steps, yet surpassing AMY at later time steps, as more clearly illustrated in the bottom row curves. This is expected, as the radiative rates at early times are suppressed by both the random walks and the Neural Network approaches.

Regarding the integral ratios, the NN/AMY ratio is less than the RW/AMY ratio, as expected since the RW bounds the NN radiative rates from below. However, what was not necessarily expected was that the RW/AMY ratio remains below 1 for all three radiative processes. It would have been possible for the increased number of radiations at later time steps to compensate for the suppression at earlier time steps, but we find here that it is not the case. In fact, this suggests that coherence time does decrease the total energy loss.

As we increase the energy of the jets in figs. 5.7 & 5.8, we can readily see changes for each radiative process. For the $q \to qg$ process, the $R^{lt}_{AA}$ stabilizes for both approaches. In the $g \to qq$ process, both $R^{lt}_{AA}$ seem to stabilize as well, although the statistical error is much larger, which is consistent with the fact that the quantity of emissions is much smaller, as most of jet energy loss is from gluon radiation. Regarding the $g \to gg$ process, there is a clear peak at early time steps, while the plateau does not seem to be reached at later times. In all processes, we would expect a plateau in the $R^{lt}_{AA}$, since the radiative rate corrections are at early times, and in the long term limit CH-G can be approximated by the AMY decay rates.

In 5.6 we can already observe that for all three processes, the total emissions per event ratios are reduced with the RW and NN approaches. While the ratio of the integrals is significantly smaller for the NN approach, the ratios are not as damped for the RW, as the $q \to gq$ is as high as 0.97. The ratios of both the RW and NN approaches decrease while increasing the jet energy for the $q \to gq$ and $g \to qq$ processes. Regarding the

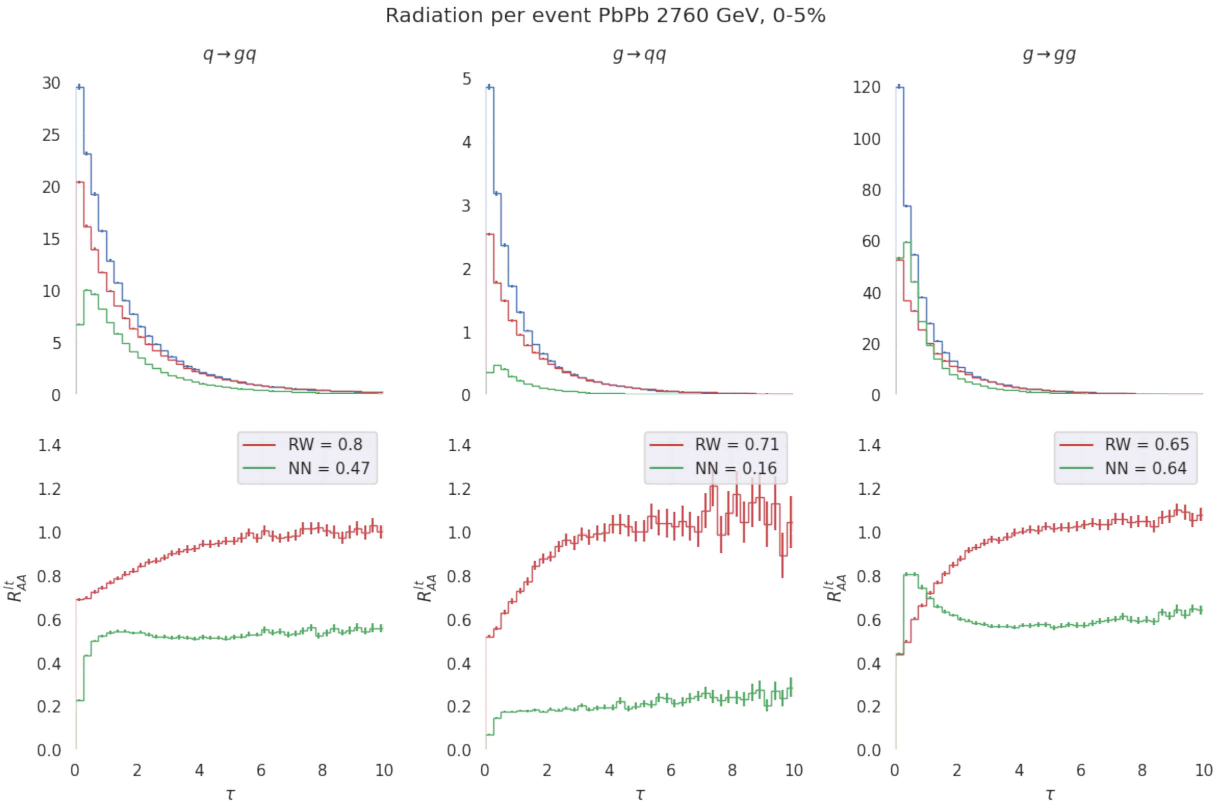

**Figure 5.7:** In the first row, there is the distribution of the time between radiations for each radiative process, for the AMY formalism (in blue), the random walks approach (in red), and the Neural Network approach (in green). In the second row, we have the ratio of the above distributions with the AMY formalism as a denominator. In red, the Random walk approach as the numerator, and in green, the neural network approach. The experiment is a Pb-Pb collision at 2760 GeV, with the centrality bin $0 - 5\%$, and the jet $p_T$ in $[65, 70]$ GeV.

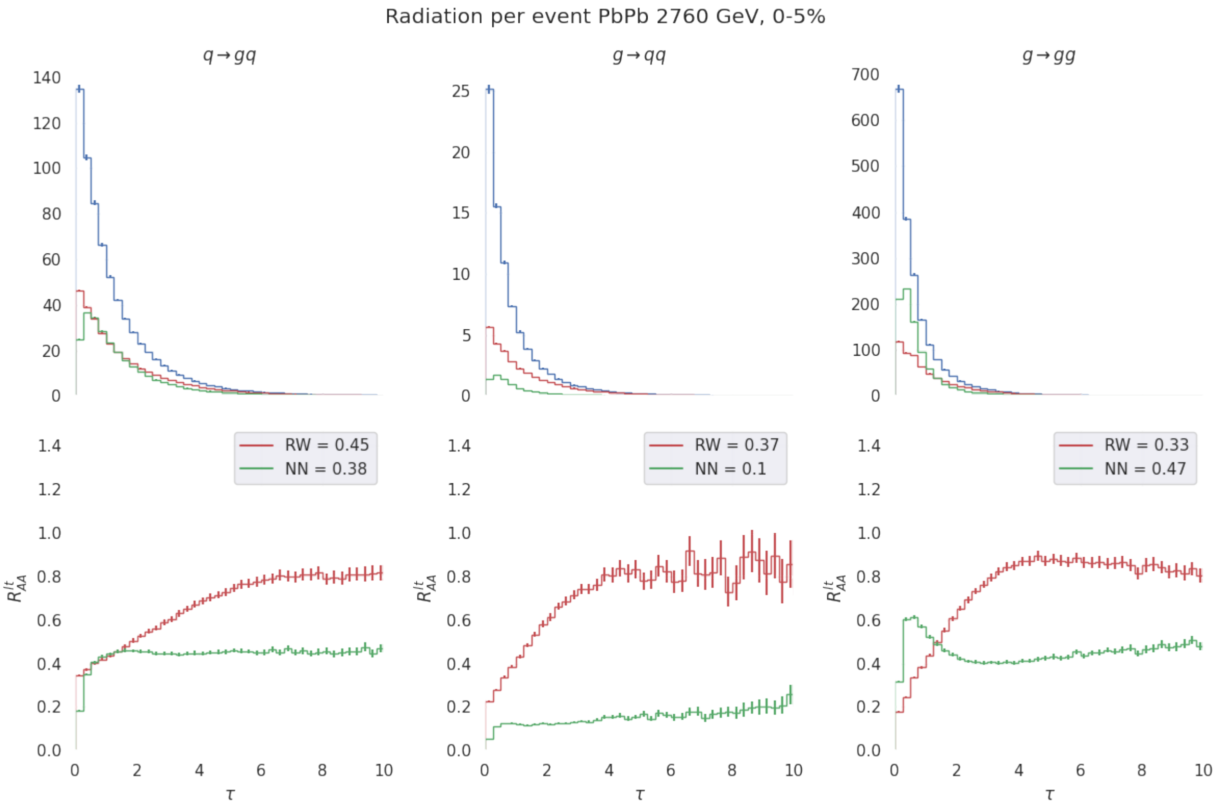

**Figure 5.8:** In the first row, there is the distribution of the time between radiations for each radiative process, for the AMY formalism (in blue), the random walks approach (in red), and the Neural Network approach (in green). In the second row, we have the ratio of the above distributions with the AMY formalism as a denominator. In red, the Random walk approach as the numerator, and in green, the neural network approach. The experiment is a Pb-Pb collision at 2760 GeV, with the centrality bin $0 - 5\%$, and the jet $p_T > 300$ GeV.

$g \to gg$ process, a qualitative difference appears at the highest chosen jet $p_T$ values in fig. 5.8. While the $R^{lt}_{AA}$ for the RW approach remains above that of the NN $R^{lt}_{AA}$ for most of the evolution, the NN surpasses it before they intersect around 2 fm. As most radiations happen in the early time steps, this small time segment carries most of the weight and results in a larger number of radiations for the NN approach.

Finally, it is worth noting that besides the relative changes in radiations between energy loss approaches, the magnitude of total radiations per event increase with jet energy increase, spanning two orders of magnitude from the lowest to highest considered jet

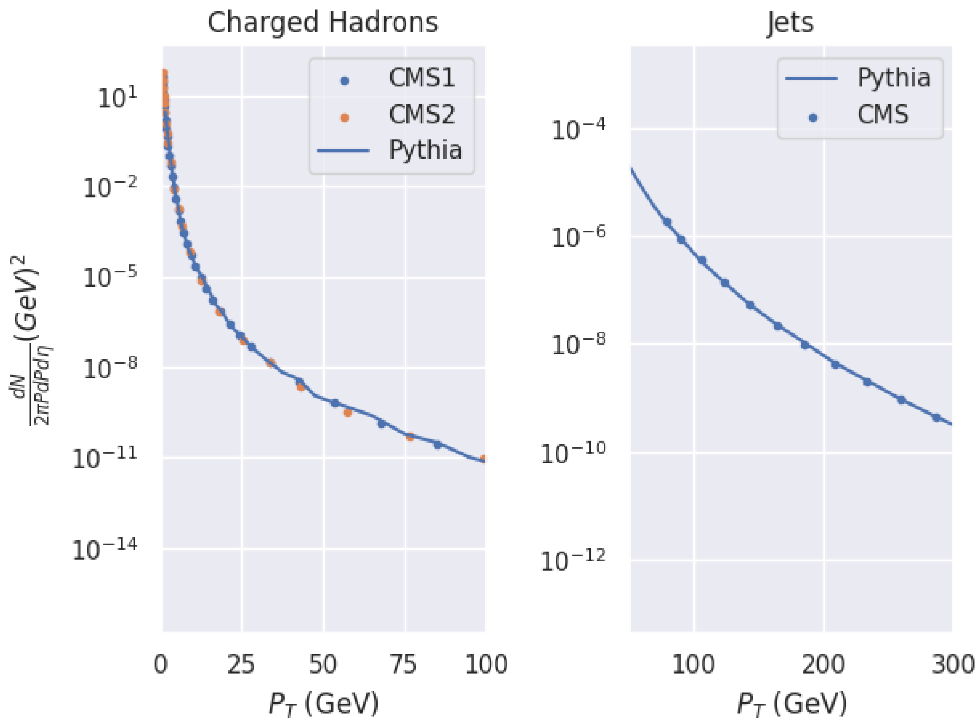

**Figure 5.9:** Charged Hadron and Jet cross sections from a simulated pp collision using Pythia. Proton-Proton collisions are taken as calibration as no QGP is believed to be formed. The experimental data for the charged hadrons is from [12], and the data for the jet cross section is from [13].

momenta. Overall, the absolute number of radiations and the ratio of total radiations for different jet evolution approaches imply a significant impact on energy loss by the radiative rates corrections.

Although the difference in radiations are larger with the differential rates' corrections for larger $P_T$, the inelastic cross sections decrease as $P_T$ is decreased. To obtain the full picture of radiative energy loss, the jet $R_{AA}$ must be constructed by combining all these $P_T$ bins, and weighting them by the inelastic cross section corresponding to that $p_T$ bin.

### 5.2.2 Observables

We first simulated pp collisions at $\sqrt{s} = 2.76$ TeV and compare the experimental results for the charged hadron multiplicity with [13], and [12] in fig. 5.9. With the validity of the

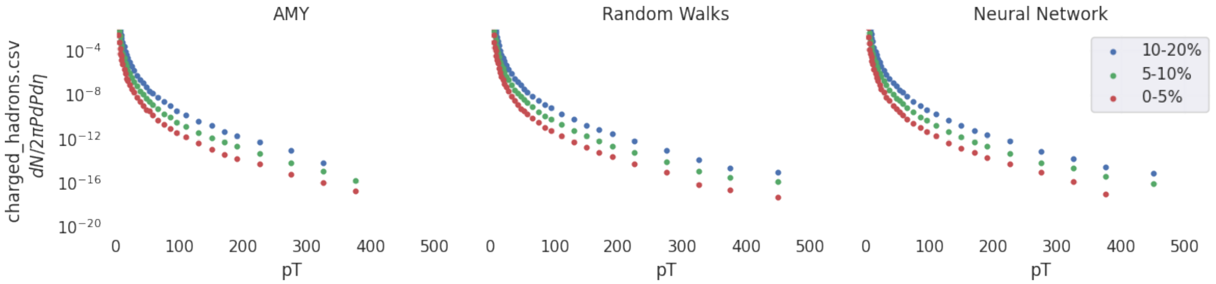

**Figure 5.10:** Charged hadron spectrum for the three considered formalisms, for centrality classes $0-5\%$, $5-10\%$ and $10-20\%$ from bottom to top, with the latter two multiplied by a factor of $10$ and $10^2$ respectively, to distinguish the curves. The columns separate the jet evolution approaches, with the AMY approach on the left, the Random walks in the middle and the Neural Network on the right.

initial data from pythia calibrated to experimental results, we are now ready to add the impact of the medium evolution to the hadron spectrum simulation. We then simulated PbPb collisions at $\sqrt{s} = 2.76$ TeV at the centrality classes 0-5%, 5-10% and 10-20%, and compare our results with the experimental data from [14] and [49].

In this section we present the spectra and nuclear modification factors for charged hadrons and jet production. we utilize centrality classes $0-5\%$, $5-10\%$ and $10-20\%$, and the jet radii R utilized in the $d_{ij}$ calculations in the $Anti-K_t$ algorithm described in sec. 4.1.1 are $R = 0.2, 0.3$ and $0.4$. This part of the thesis builds on the work in [11], as the Random Walk approach is compared to our neural network approach, as an alternative method to emulate the CH-G formalism [10].

### 5.2.2.1 Charged Hadron multiplicity

In fig. 5.10, we see the raw yield spectrum utilizing the different proposed evolution formalisms. While some differences are visible, we utilize the nuclear modification factor to better contrast the difference in observables from the formalisms.

In fig. 5.11 are the nuclear modification factors for charged hadron multiplicity, with the centrality bins increasing from left to right. In the centrality classes $0-5\%$ and $5-10\%$,

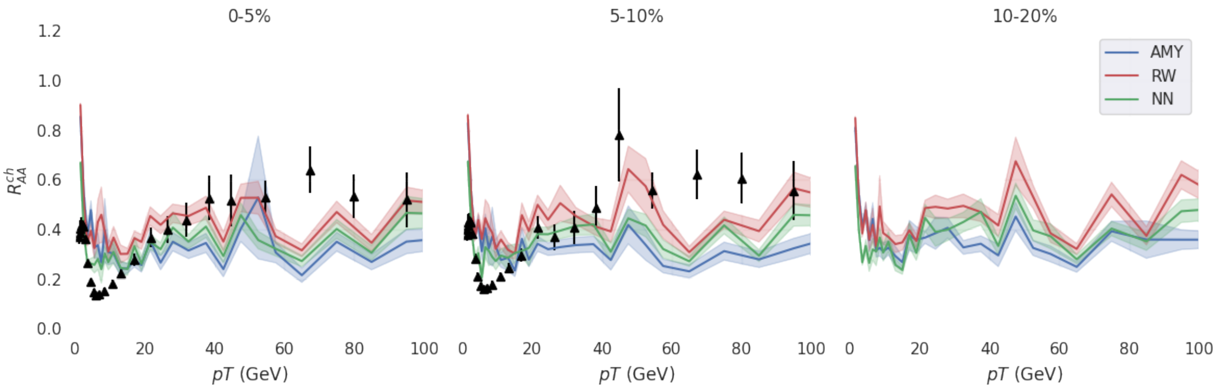

**Figure 5.11:** Nuclear modification factor for the charged hadrons, for the three considered formalisms, for centrality classes $0-5\%$, $5-10\%$ and $10-20\%$ from left to right. The AMY formalism is shown in blue, the Random Walk approach is shown in red, and the neural network approach is shown in green. The experimental data from CMS [13] is shown in black triangles. For the three considered formalisms, the shaded regions correspond to the statistical error. For the experimental data, the error bars correspond to the statistical and systematic uncertainties added quadratically.

the CMS experimental data is shown as black triangles. Across centrality bins, and over all of the $p_T$ range covered by experimental data, the neural network $R_{AA}$ lies between the $R_{AA}$ from the AMY formalism and the Random walk approach. This is expected, as the decay rates of the NN approach should be between that of the AMY formalism and the Random Walk approach. We also note that below about $p_T < 20$ GeV, all three considered formalisms lie outside of the statistical error range from the experimental data. This is also expected, as both the AMY and the CHG formalisms are not a good approximation at low $p_T$.

#### 5.2.2.2 Jets

Having regarded charged hadrons as independent entities, we now consider them under the jet agglomerations. In Fig. 5.12 we have the spectra of jet yield with the radius fixed at $R = 0.2$. Again, we note that few differences are visible with the naked eye, and resort

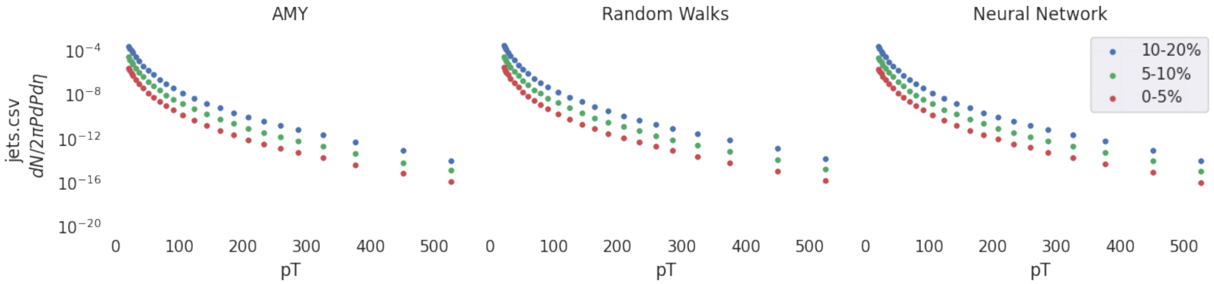

**Figure 5.12:** Jet Yield spectrum for the three considered formalisms, for centrality classes $0-5\%$, $5-10\%$ and $10-20\%$ from bottom to top, with the latter two multiplied by a factor of $10$ and $10^2$ respectively.

to studying the nuclear modification factor to better contrast the formalisms. We consider jet radii of $R = 0.2, 0.3$ & $0.4$ when looking at the nuclear modification factor.

In Fig. 5.13, we have the jet $R_{AA}$ for the three considered centrality classes, contrasting the considered formalisms, at a jet radius $R = 0.2$. A first remark is that neither of the considered formalisms contain the experimental data within their statistical error. Nevertheless, we can learn from the difference between the predictions. A common feature of the considered $R_{AA}$'s is that they seem uncorrelated of the jet energy. This could be because a static strong coupling constant $\alpha_s$ was used. A running coupling constant would have allowed for more variations across $p_T$ values. However, this does not persist for a larger jet radius as we will discuss. In all three formalisms, there is also a small peak at very low $p_T$, but should not be considered as that is out of the scope of our evolution formalisms. What is unexpected about the relationship between the $R_{AA}$ of the different formalisms, is that the green curve, which is from the neural network approach, lies below both the AMY and the Random Walk $R_{AA}$'s.

Regarding the variations related to centrality bins, we note small $R_{AA}$ variations as more peripheral centrality bins are taken. We would expect nuclear modification factors to increase as more peripheral bins are taken since less QGP is produced, thus the jet production should be quenched less as well, as it approaches a more p-p like collision.

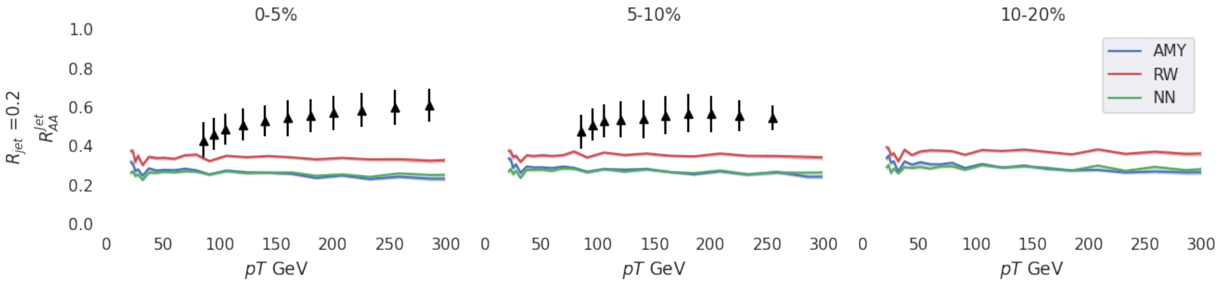

**Figure 5.13:** Jet nuclear modification factor for the three considered formalisms, for centrality classes $0-5\%$, $5-10\%$ and $10-20\%$ from left to right. The jet radius is fixed at $R=0.2$. The experimental data from CMS [14] is shown in black triangles. For the three considered formalisms, the shaded regions correspond to the statistical error. For the experimental data, the error bars correspond to the statistical and systematic uncertainties added quadratically.

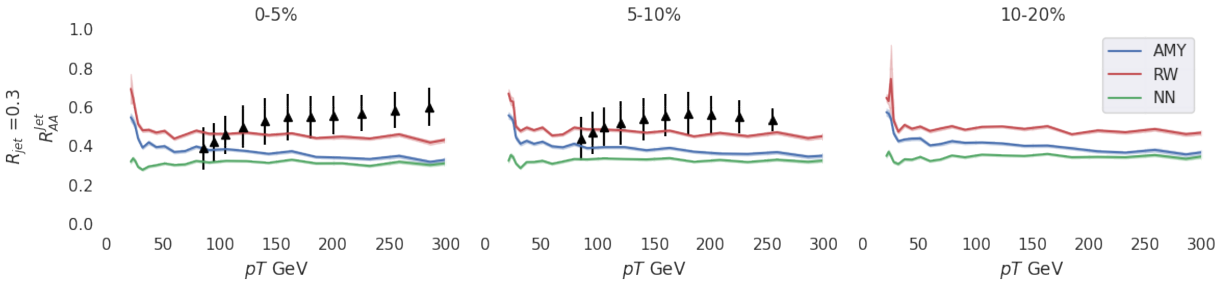

**Figure 5.14:** Jet nuclear modification factor for the three considered formalisms, for centrality classes $0-5\%$, $5-10\%$ and $10-20\%$ from left to right. The jet radius is fixed at $R=0.3$. The experimental data from CMS [14] is shown in black triangles. For the three considered formalisms, the shaded regions correspond to the statistical error. For the experimental data, the error bars correspond to the statistical and systematic uncertainties added quadratically.

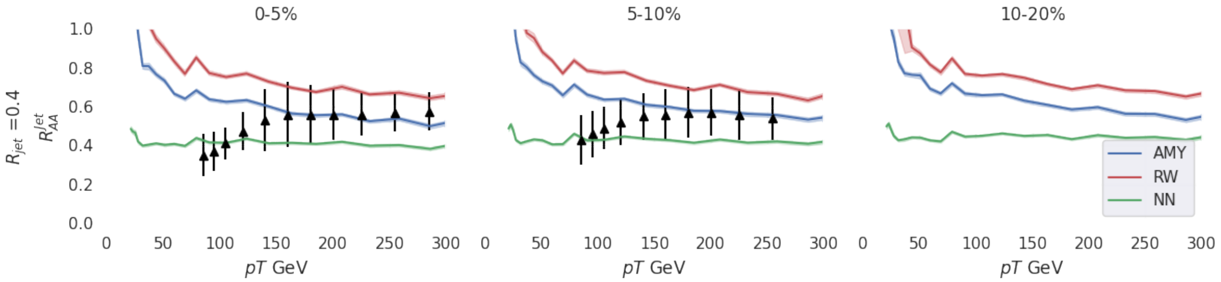

**Figure 5.15:** Jet nuclear modification factor for the three considered formalisms, for centrality classes $0-5\%$, $5-10\%$ and $10-20\%$ from left to right. The jet radius is fixed at $R=0.4$. The experimental data from CMS [14] is shown in black triangles. For the three considered formalisms, the shaded regions correspond to the statistical error. For the experimental data, the error bars correspond to the statistical and systematic uncertainties added quadratically.

We note from figures 5.14 and 5.15 that as we increase the jet radius, both the AMY and the Random walk's $R_{AA}$ increase, mostly for lower jet energy. As the jet radius is increased, the tolerance to declare an agglomerate as a jet is lower, and thus more jets should be found, as we observe with the AMY and the Random walk's $R_{AA}$. However, the $R_{AA}$ from the neural network remains somewhat stable with larger jet radii.

This suggests that the jets from the neural network approach are more narrowly distributed in the $\eta/\phi$ space, since the yield is insensitive to radius change. On the other hand, the jets from the AMY and Random walk approaches are more broadly distributed, and thus the $R_{AA}$ is sensitive to radius increase, as more jets are resolvable.

The reason for this lies within the implementation of the coherence time consideration. When utilizing the AMY or the neural network approaches, there is no coherence time. Whenever a radiation happens in the simulation, both resulting partons are considered separate entities. The coherence time is taken into consideration by damping the radiative rates.

When utilizing the random walk formalism, the radiative rates used are the same as AMY. However, when a radiation happens in the simulations, both partons remain coherent, until the process explained in 4.27 determines that they are no longer coherent.

During the coherence time, both partons are not free to radiate further, effectively decreasing the radiative rates.

During the coherence time, while the two partons are not free to radiate, they are free to undergo elastic scatterings, resulting in a difference in relative momentum between the radiating partons. As a result, the jets are more broadly distributed, as observed with the change in jet $R_{AA}$ of the random walk approach. It also makes sense that the AMY jet distribution would be more narrow than that of the Random walk, since no coherence time implies there is no period where the radiating partons are in the same vicinity, and thus no period where they can gain relative transverse momentum by elastic scatterings. Finally, if there is no coherence time, and there are less overall radiations, then there will be less available partons to undergo elastic scatterings and thus gain transverse momentum, resulting in thinner jet cones.

One could argue that the Random Walk model is more physical, as the partons should be able to undergo elastic scatterings while in coherence. However, the random walk approach is a rougher estimate of the CH-G rates than that of the neural network.

While the coherence time consideration would explain why the $R_{AA}^{jet}$ of the random walk would be above that of AMY, it does not explain why the neural network remains below both other approaches. We attribute this to the fact that the CH-G model does not include absorption processes from the medium, which would increase the jet energy. These absorption processes are allowed within the AMY formalism, and thus of the Random Walks as well.

While the lack of absorption processes yields a higher jet quenching across all considered $p_T$ regions, the Neural Network $R_{AA}^{jet}$ with $R = 0.4$ is closer to the observed $R_{AA}^{jet}$ at low $p_T$ in fig. 5.15. However, the other approaches better describe the data in other regions and for other choices of jet radius, and so it remains inconclusive as to which is a better description.

# Chapter 6

# Conclusion

We have suggested two parts of Heavy-Ion collisions simulations where machine learning can be used to accelerate computations, to the point where less approximations are required, making the simulations more physically sound. We have then simulated PbPb collisions, using MARTINI, a Monte-Carlo event generator to which we have added our neural networks for radiative rate calculations. We were then able to compare three jet evolution approaches, with the nuclear modification factors for both jet and charged hadron yield.

Regarding Initial Conditions, we have shown a proof of concept that neural networks can be used to generate ansatzes as an initial point in iterative calculations, resulting in four times less calculation steps, without compromising any physical validity of the solution.

Regarding Jet evolution, we have shown that the CH-G [10] jet evolution formalism dampens the number of radiations in QGP, when compared to AMY, as expect by their damped decay rates at early times since birth of the parent particle. This in turn translates to a charged hadron $R_{AA}$ between that of AMY and the Random Walk approach [11], as expected since the random walk decay rates bound the CH-G from below. We have also shown that while the charged hadron yield lies between that of AMY and the Random Walks, the jet yield is more quenched, relatively to both other approaches across all considered $p_T$ regions. We attribute this to the fact that the CH-G formalism does not include absorption from the medium.

In conclusion, we have shown that machine learning can be used to accelerate physically sound calculations, despite the black-box nature of neural networks. Regarding initial conditions color gauge fields, we have shown that taking a neural network as an ansatz for an numerical iterative procedure outperforms taking the continuous, analytical solution as an ansatz by a factor of four. Regarding Jets, the newly possible calculations allow us to take a more refined physical physical model. For future work, in the jet evolution implementation, one could implement the time dilation of the partons by taking into consideration the cell velocity of the hydrodynamic simulation in which the partons are situated in each time step. One could also utilize a running coupling constant, which would allow for a more dynamic jet $R_{AA}$ as a function of $p_T$, and be more physically sound.

# Bibliography

[1] R. Quagliani, *Study of Double Charm B Decays with the LHCb Experiment at CERN and Track Reconstruction for the LHCb Upgrade*, ch. Introduction to Theory, pp. 1–28. Cham: Springer International Publishing, 2018.

[2] V. Khachatryan and H. Heath, "Measurement of the inclusive 3-jet production differential cross section in proton–proton collisions at 7 tev and determination of the strong coupling constant in the tev range," *European Physical Journal C: Particles and Fields*, vol. C75, no. 5, p. 186, 2015.

[3] G. Martínez, "Advances in quark gluon plasma," *arXiv: Nuclear Experiment*, vol. 1304.1452, 2013.

[4] ALICE Collaboration, "Centrality determination in heavy ion collisions," *ALICE Public Record*, vol. 011, 2018.

[5] F. Gelis, E. Iancu, J. Jalilian-Marian, and R. Venugopalan, "The color glass condensate," *Annual Review of Nuclear and Particle Science*, vol. 60, pp. 463–489, nov 2010.

[6] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," *ArXiv*, vol. abs/1603.07285, 2016.

[7] N. M. Fortier, *Non-trivial initial state anisotropies and their effects on observables in the ip-glasma framework.* Msc thesis, Department of Physics, McGill University, 2021.

[8] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Inter-*

*vention – MICCAI 2015* (N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, eds.), (Cham), pp. 234–241, Springer International Publishing, 2015.

[9] P. B. Arnold, G. D. Moore, and L. G. Yaffe, "Effective kinetic theory for high temperature gauge theories," *Journal of High Energy Physics*, vol. 01, p. 030, 2003.

[10] S. Caron-Huot and C. Gale, "Finite-size effects on the radiative energy loss of a fast parton in hot and dense strongly interacting matter," *Physical Review C*, vol. 82, p. 064902, 2010.

[11] C. Park, *Jet Modification in Strongly Coupled Quark-Gluon Plasma*. Phd thesis, Department of Physics, McGill University, 2020.

[12] G. Aad *et al.*, "Measurement of charged-particle spectra in Pb+Pb collisions at $\sqrt{s_{\mathsf{NN}}} = 2.76$ TeV with the ATLAS detector at the LHC," *JHEP*, vol. 09, p. 050, 2015.

[13] S. Chatrchyan *et al.*, "Study of High-pT Charged Particle Suppression in PbPb Compared to *pp* Collisions at $\sqrt{s_{NN}} = 2.76$ TeV," *Eur. Phys. J. C*, vol. 72, p. 1945, 2012.

[14] G. Aad *et al.*, "Measurements of the Nuclear Modification Factor for Jets in Pb+Pb Collisions at $\sqrt{s_{\mathrm{NN}}} = 2.76$ TeV with the ATLAS Detector," *Phys. Rev. Lett.*, vol. 114, no. 7, p. 072302, 2015.

[15] W. Falcon and The PyTorch Lightning team, "PyTorch Lightning," Mar. 2019.

[16] G. Zweig, "An su(3) model for strong interaction symmetry and its breaking," *CERN-TH*, vol. 412, pp. 64–170, 1964.

[17] J. I. Kapusta and C. Gale, *Finite-Temperature Field Theory: Principles and Applications*. Cambridge Monographs on Mathematical Physics, Cambridge University Press, 2 ed., 2023.

[18] S. Mandelstam, "Determination of the pion-nucleon scattering amplitude from dispersion relations and unitarity. general theory," *Phys. Rev.*, vol. 112, pp. 1344–1360, Nov 1958.

[19] H1 and Z. Collaborations, "Combination of measurements of inclusive deep inelastic $e^{\pm}p$ scattering cross sections and qcd analysis of hera data," *arXiv: High Energy Physics - Experiment*, vol. 1506.06042, 2015.

[20] A. Kovner, L. McLerran, and H. Weigert, "Gluon production from non-abelian weizsäcker-williams fields in nucleus-nucleus collisions," *Phys. Rev. D*, vol. 52, pp. 6231–6237, Dec 1995.

[21] D. N. Blaschke, F. Gieres, M. Reboud, and M. Schweda, "The energy–momentum tensor(s) in classical gauge theories," *Nuclear Physics B*, vol. 912, pp. 192–223, 2016. Mathematical Foundations of Quantum Field Theory: A volume dedicated to the Memory of Raymond Stora.

[22] B. Schenke, S. Jeon, and C. Gale, "(3+1)d hydrodynamic simulation of relativistic heavy-ion collisions," *Phys. Rev. C*, vol. 82, p. 014903, Jul 2010.

[23] S. McDonald, S. Jeon, and C. Gale, "Ip-glasma phenomenology beyond 2d," *Nuclear Physics A*, vol. 982, p. 239–242, Feb. 2019.

[24] S. Ryu, J.-F. m. c. Paquet, C. Shen, G. Denicol, B. Schenke, S. Jeon, and C. Gale, "Effects of bulk viscosity and hadronic rescattering in heavy ion collisions at energies available at the bnl relativistic heavy ion collider and at the cern large hadron collider," *Phys. Rev. C*, vol. 97, p. 034910, Mar 2018.

[25] C. Bierlich, S. Chakraborty, N. Desai, L. Gellersen, I. Helenius, P. Ilten, L. Lönnblad, S. Mrenna, S. Prestel, C. T. Preuss, T. Sjöstrand, P. Skands, M. Utheim, and R. Verheyen, "Codebase release 8.3 for PYTHIA," *SciPost Phys. Codebases*, pp. 8–r8.3, 2022.

[26] B. Schenke, C. Gale, and S. Jeon, "MARTINI: An event generator for relativistic heavy-ion collisions," *Physical Review C*, vol. 80, nov 2009.

[27] M. Bleicher, E. Zabrodin, C. Spieles, S. A. Bass, C. Ernst, S. Soff, L. Bravina, M. Belkacem, H. Weber, H. Stöcker, and W. Greiner, "Relativistic hadron-hadron collisions

in the ultra-relativistic quantum molecular dynamics model," *Journal of Physics G: Nuclear and Particle Physics*, vol. 25, pp. 1859–1896, sep 1999.

[28] Darthmouth University, "Artificial intelligence coined at dartmouth," n.d. Accessed Nov 16, 2023. `https://home.dartmouth.edu/about/artificial-intelligence-ai-coined-dartmouth`.

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

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[31] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of adam and beyond," *arXiv preprint arXiv:1904.09237*, 2019.

[32] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *International Conference on Machine Learning*, 2010.

[33] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals, and Systems*, 1989.

[34] C. N. Yang and R. L. Mills, "Conservation of isotopic spin and isotopic gauge invariance," *Phys. Rev.*, vol. 96, pp. 191–195, Oct 1954.

[35] M. Cautun, *Photon production in the Color Glass Condensate formalism*. Msc thesis, Department of Physics, McGill University, 2009.

[36] J. Han, L. Zhang, and W. E, "Solving many-electron schrödinger equation using deep neural networks," *Journal of Computational Physics*, vol. 399, p. 108929, dec 2019.

[37] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, p. 448–456, JMLR.org, 2015.

[38] R. Atkin, "Review of jet reconstruction algorithms," *Journal of Physics: Conference Series*, vol. 645, p. 012008, sep 2015.

[39] M. Cacciari, G. P. Salam, and G. Soyez, "The anti-kt jet clustering algorithm," *Journal of High Energy Physics*, vol. 04, p. 063, apr 2008.

[40] L. D. Landau and I. Pomeranchuk, "Limits of applicability of the theory of bremsstrahlung electrons and pair production at high-energies," *Dokl. Akad. Nauk Ser. Fiz.*, vol. 92, p. 535, 1953.

[41] A. B. Migdal, "Bremsstrahlung and pair production in condensed media at high energies," *Phys. Rev.*, vol. 103, pp. 1811–1820, Sep 1956.

[42] P. Arnold, G. D. Moore, and L. G. Yaffe, "Photon and gluon emission in relativistic plasmas," *Journal of High Energy Physics*, vol. 06, p. 030, jul 2002.

[43] B. G. Zakharov, "Radiative energy loss of high-energy quarks in finite-size nuclear matter and quark-gluon plasma," *Journal of Experimental and Theoretical Physics Letters*, vol. 65, pp. 615–620, apr 1997.

[44] N. N. Nikolaev, "Pomeron physics in diffractive deep inelastic scattering at large 1/x," *Nuclear Physics B - Proceedings Supplements*, vol. 25, pp. 152–166, 1992. Elastic and Diffractive Scattering.

[45] V. N. Gribov and L. N. Lipatov, "Deep inelastic e p scattering in perturbation theory," *Sov. J. Nucl. Phys.*, vol. 15, pp. 438–450, 1972.

[46] A. Buckley, J. Ferrando, S. Lloyd, K. Nordström, B. Page, M. Rüfenacht, M. Schönherr, and G. Watt, "LHAPDF6: parton density access in the LHC precision era," *Eur. Phys. J. C*, vol. 75, p. 132, 2015.

[47] I. Helenius, K. Eskola, H. Honkanen, and C. Salgado, "Modeling the impact parameter dependence of the npdfs with eks98 and eps09 global fits," *Nuclear Physics A*, vol. 910-911, pp. 510–513, 2013. Hard Probes 2012.

[48] C. F. Gauss, *METHODUS NOVA INTEGRALIUM VALORES PER APPROXIMATIONEM INVENIENDI*, vol. 3 of *Cambridge Library Collection - Mathematics*, p. 165–196. Cambridge University Press, 2011.

[49] V. Khachatryan *et al.*, "Measurement of inclusive jet cross sections in $pp$ and PbPb collisions at $\sqrt{s_{NN}} = 2.76$ TeV," *Phys. Rev. C*, vol. 96, no. 1, p. 015202, 2017.

[50] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, pp. 8024–8035, Curran Associates, Inc., 2019.

# Appendix A

# Code changes to MARTINI

First we add the option to load the neural networks, as opposed to reading from a table as with the other formalisms.

```
void Import::init(int rateSelector){
  if (rateSelector < 5)
    {
    read_table( &dat, &Gam, rateSelector);
    }
  else if (rateSelector == 5)
    {
      load_models();
    }

readElasticRate();
readElasticRateOmegaQ();
}
```

We define the loading function we just called, utilizing the c++ Pytorch [50] extension, libtorch. The models themselves have been written and trained utilizing Pytorch beforehand.

```
void Import::load_models()
{
  string path = "";
  const char* MARTINIPATH = "MARTINIPATH";
  char* envPath = getenv(MARTINIPATH);
  if (envPath != 0 && *envPath != ’\0’)
    {
      int i = 0;
      while (*(envPath+i) != ’\0’) path += *(envPath+(i++));
      path += "/main/data/";
    }
  else path = "./data/";
  cout << "Loading Rate Models." << endl;
  rate_mod_0 =
     torch::jit::load(path+"compiled_models/rate_modelling/Method_0.pt");
  rate_mod_1 =
     torch::jit::load(path+"compiled_models/rate_modelling/Method_1.pt");
  rate_mod_2 =
     torch::jit::load(path+"compiled_models/rate_modelling/Method_2.pt");
  rate_mod_3 =
     torch::jit::load(path+"compiled_models/rate_modelling/Method_3.pt");
  cout << "Rate Models Loaded." << endl;
}
```

We can now use the option of calling the neural network instead of the tables

```
double Import::getRate(double p, double k, double T, double t, int
   rateSelector){
  double rate = 0;
  if ( rateSelector < 5) rate  = use_table ( p , k , Gam.dGamma , 0);
  if ( rateSelector == 5) rate = use_model(p, k, T, t, 0);
```

```
  return rate;
}
```

which calls the model as follows to obtain the differential decay rates

```
double Import::use_model( double p , double k , double T , double t,
    int process ){
  double output;
  std::vector<torch::jit::IValue> inputs;
  inputs.push_back(torch::tensor({{p, k, T, t}}));
  if ( process + 1 == 1 ) output =
     rate_mod_0.forward(inputs).toTensor().item<double>();
  if ( process + 1 == 2 ) output =
     rate_mod_1.forward(inputs).toTensor().item<double>();
  if ( process + 1 == 3 ) output =
     rate_mod_2.forward(inputs).toTensor().item<double>();
  if ( process + 1 == 4 ) output =
     rate_mod_3.forward(inputs).toTensor().item<double>();
  return output;
}
```

Which completes the back-end changes of the rates routines inside the MARTINI program.

As for the main interface, the time used to compute the decay rates is the difference between the current time-step and the time of the last emission from the parton, in fm, which we call `t_rel`.

To compute `t_rel`, we need to record for each parton the time of the last emission, so we add that time as an attribute of the parton class. This attribute gets updated for both the parent and daughter at each emission.

```
class Parton
{
```

```
    private:
        ...
        double itsCreationt;

    public:
        ...
        double tAtEmission() const {return itsCreationt;}
        void tAtEmission(double value) { itsCreationt=value;}

}
```

Now in the main *evolve* MARTINI function which loops over all partons and all time steps, we compute `t_rel`. It is then used when calling the decay and differential decay rate for any of the four radiation processes as follows, and the time of last emission is updated whenever there is one.

```
// evolve every parton by one time step. This is the core of MARTINI.
int MARTINI::evolve(vector<Parton>  *plist, vector<Source> *slist, int
   counter, int it)
{
    double t_rel;
    ...
    for ( int i=0; i<imax; i++){    // loop over all partons
    ...
    t_rel = t - plist->at(i).tAtEmission();
    ...
    if( radiate == 1 ) // see if emission happens
    {
        if(pRest/T>AMYpCut) // do not evolve partons with momenta
           below this scale
        {
```

```
// do process 1, q->qg
f = rates->findValuesRejection(pRest/T, T, t_rel,
   alpha_s_rad, Nf, random, import, 1);
...
if (delp > pCut) // if new parton is kept (f.y*T >
   threshold [in GeV])
{
...
plist->at(i).tAtEmission(t);
plist->push_back(newOne);  // add the gluon to the list of
   partons if (f.y>AMYpCut)
LT_list_0->push_back(t_rel); //Store the time difference
   in the lifetimes histogram
```

`LT_list_0` in the last line of code is a vector which is being used to store the lifetimes of the particles between emissions, for each process. Those lifetimes are then used to produce figures 5.6, 5.7 & 5.8.