Karlsruhe Institute of Technology

Institute for Theoretical Physics

Heavy-Ion Collisions with Herwig Glauber Model Approach

Master's Thesis of

Julian Kalema Lukwata

at the Department of Physics
Institute for Theoretical Physics (ITP)

Reviewer:	PD Dr. S. Gieseke
Second reviewer:	Prof. Dr. D. Zeppenfeld

01. December 2019 – 07. February 2021

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der gültigen Fassung vom 24.05.2018 beachtet habe.

Karlsruhe, 08.02.2021

. . . . . .

(Julian Kalema Lukwata)

Als Prüfungsexemplar genehmigt von

Karlsruhe, 08.02.2021

. . . . . .

(PD Dr. S. Gieseke)

Abstract

At today's high energy particle colliders a multitude of different experiments is performed to study the fundamental forces of Nature. Heavy-ion collisions are particularly interesting for the study of extremely hot and dense partonic matter, as it is expected to be prevalent at the early stage of our Universe. In this thesis the simulation code SimpleHeavyIon is introduced based on the Monte-Carlo event generator Herwig. A Glauber model determines the nucleon-nucleon interactions pairs. Herwig will produce these interactions and the new code will combine them into a single heavy-ion event This code is the first for Herwig that can correlate multiple nucleon-nucleon interactions via color reconnection at the nucleus level. The code is able to reproduce basic metrics like the rapidity distribution of charged particles and might be used as a stepping stone for the further development in this direction.

Contents

1. Introduction

In the pursuit of a better understanding of Nature, we as humans constantly try to extend the boundaries of our knowledge. One of the recurring questions concerns the state of our universe and in particular its matter shortly after the Big Bang. At this time the conditions were radically different from today. The matter was extremely hot and densely packed together, which is believed to have caused the formation of the so-called quark-gluon plasma (QGP), in which quarks and gluons are not confined to hadrons as is the case nowadays. With current experiments at the RHIC and LHC, we try among other goals to find hints of the existence of QGP by conducting heavy-ion collisions. Heavy-ion collisions are especially compelling in this regard as it is possible to produce extremely hot and dense matter at the size of a nucleus, which could be a big enough volume for the formation of QGP.

An essential part of many experiments today is the simulation of the processes at work as we often cannot directly translate predictions of a theory into measurable observables. If we look at pp-collisions, the usage of general-purpose Monte Carlo (MC) event generators like Herwig has proven to be a feasible approach, improving constantly over time. The basis of MC event generators is a microscopic description of the hadrons and their interactions, in part based on first principle of QCD and QED. However, the majority of current heavy-ion collision simulations are not based on MCs but instead on hydrodynamic models. These hydrodynamic models have shown to reproduce some key observables such as the elliptic flow variable with a good accuracy. In contrast to MC event generators, they are based on the assumption of a thermalized system. Verifying and reproducing existing knowledge is a good scientific practice, therefore it is sensible to develop an alternative approach which is based on the Glauber model for the simulation of heavy-ion collisions. This might even broaden our understanding of the physics at work and lead to a mutual imporvement of both techniques.

In this thesis a program code based on the Glauber model for the simulation of heavy-ion collisions is developed. The Glauber model is at its core a geometrical model that produces nucleon-nucleon (NN) collision pairs. It is based on the eikonal assumption and high-energy limit, meaning that the nucleons will travel in straight lines during the heavy-ion collision. We use parts of the existing MC event generator Herwig as the backbone of our model. Herwig provides us with NN-interactions and we combine them with the use of the Glauber model.

The code is written in Python, a widely used programming language in scientific cycles and in general. SimpleHeavyIon will run Herwig in the background to produce parton-level events, which are then merged into a single heavy-ion event, following the Glauber calculation. This heavy-ion event is then given back to another instance of Herwig, which will perform the remaining soft physics and produce a final-state, that would be measured in the detector. In addition to the Python code, we modified ThePEG and Herwig to serve

our purpose. We added a read-in interface to ThePEG, which is essential to be able to start Herwig with the heavy-ion event at parton-level.

The thesis starts with chapter 2 summarizing the key properties of QCD and shortly introducing hydrodynamic models in the context of quark-gluon plasma. Chapter 3 introduces the fundamental building blocks of a Monte Carlo event generator. These are the standard simulation tools for hadron-hadron or lepton-lepton collisions but as of yet have not been fully studied for the simulation of heavy-ion events. In chapter 4, the geometrical Glauber model is introduced. It forms the backbone of our heavy-ion collision simulation. In chapter 5, the simulation code SimpleHeavyIon is discussed in more detail, by giving an overview of the workflow, the different classes and some examples of how to use it. Next we compare the simulated data in chapter 6 with some basic experimental data to study which parameters and effects play an important role the on the basic structure of the final-state heavy-ion event. Finally, we conclude in chapter 7 and give an outlook for further work.

2. Quantum Chromodynamics

One primary goal behind putting immense effort in the construction and operation of particle colliders is to probe our current physical theories and potentially find new undiscovered particles. To simulate the processes which are happening at high energy colliders, it is essential to have an understanding of the main force at work at these scales – the strong interaction.

The strong interaction is one of the four fundamental forces that govern all visible matter of our universe. It is described by quantum chromodynamics (QCD), which is an $SU(3)_C$ non-abelian gauge theory that acts on the color-charged fermions and bosons, the quarks and the gluons.

2.1. Lagrangian

The interactions between quarks and gluons (collectively reffered to as partons) are governed by the QCD Lagrangian (see e.g. [1])

$$\mathcal{L}_{\text{QCD}} = \sum_q \bar{\psi}_{q,i}\left(i\gamma_\mu D^\mu_{ij} - m_q\delta_{ij}\right)\psi_{q,j} - \frac{1}{4}F^a_{\mu\nu}F^{a\mu\nu} + \mathcal{L}_{\text{gauge-fixing}} + \mathcal{L}_{\text{ghost}}, \tag{2.1}$$

where $\psi_{q,i}$ is the spinor for the quark-field of flavor q, m_q is the quark mass. The sum runs over the six quark flavors *u, d, c, s, t, b*. We get the field strength tensor

$$F^a_{\mu\nu} = \partial_\mu G^a_\nu - \partial_\nu G^a_\mu + g_s f^{abc} G^b_\mu G^c_\nu, \tag{2.2}$$

dependent on the massless gluon field G^a_μ, the coupling constant of QCD g_s and the color indices a, b, c for the eight color degrees of freedom. The final term in the field strength tensor corresponds to 3-gluon and 4-gluon interaction vertices. This term is absent in quantum electrodynamics (QED) and correspondingly photons cannot interact with each other directly. Gauge groups are defined through their structure constants, for QCD these are f^{abc}. $D_{\mu,ij}$ is the covariant derivative

$$D_{\mu,ij} = \partial_\mu\delta_{ij} - ig_s T^a_{ij} G^a_\mu, \tag{2.3}$$

in which we can find the matrices of the fundamental representation T^a, defined by the commutator relation

$$\left[T^a, T^b\right] = if^{abc}T^c, \tag{2.4}$$

where $T^a = \frac{\lambda^a}{2}$, with the Gell-Mann matrices λ^a.

To perform analytic calculations, e.g. calculating the gluon propagator, we need to make a choice of the gauge, represented by the gauge fixing term $\mathcal{L}_{\text{gauge-fixing}}$. There are some degrees of freedom in choosing the exact form of the gauge fixing term, in the case of *covariant gauge* unphysical degrees of freedom are introduced, which have to be canceled with the ghost term $\mathcal{L}_{\text{ghost}}$. The *axial gauges* are defined so that the ghost terms vanish.

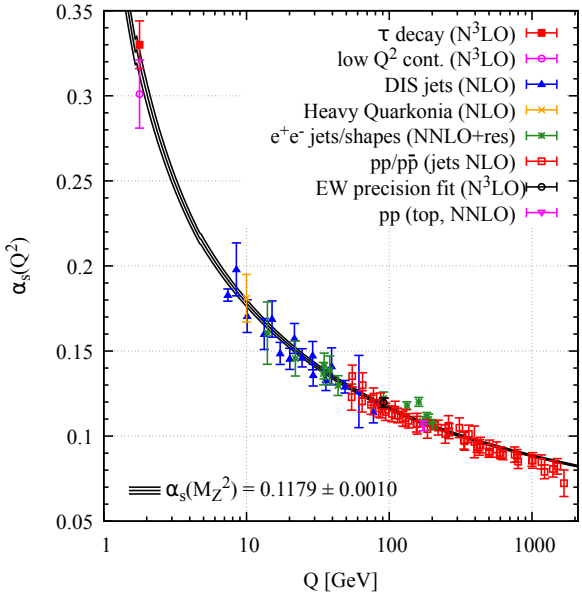


Figure 2.1.: The strong coupling $\alpha_S(Q^2)$ plot from different measurements which shows the behavior of the coupling constant. For low energy scales the strong coupling constant diverges leading to confinement, while as the energy scale increases the coupling constant decreases (asymptotic freedom). This figure is taken from [2].

2.2. Key properties of QCD

We discuss three key features of QCD, that is *confinement*, *asymptotic freedom* and *color-flow*, briefly in the following, as they are essential for the understanding of Monte Carlo event generators.

In perturbation theory one often has to deal unphysical infinities, which arise from integrating the theory over the whole phase space, even though only some parts are relevant to a calculation, leading to ultraviolet or infrared divergences. Renormalization is a consequence of such a procedure. By stating that a physical observable must not depend on the unphysical scale μ_R, this leads to the renormalization group equation (RGE)

$$\beta(\alpha_S) = \mu_R^2 \frac{\partial \alpha_S}{\partial \mu_R^2} \simeq Q^2 \frac{\partial \alpha_S}{\partial Q^2}, \tag{2.5}$$

$$\beta(\alpha_S) = -\left(b\alpha_S^2 + b'\alpha_S^3 + O\left(\alpha_S^4\right)\right), \tag{2.6}$$

where $\alpha_S = \frac{g_S^2}{4\pi}$ is another parameterization of the coupling constant, $b = \frac{(33-2n_f)}{12\pi}$, $b' = \frac{(153-19n_f)}{24\pi^2}$ and n_f is the number of active flavors, so $n_f \leq 6$. The last part in Eq. (2.5) follows, if we choose μ_R close to the momentum transfer scale Q, which in fact determines the strength of the coupling constant [2].

When we only consider the first order of Eq. (2.6), we get

$$\alpha_S(Q^2) = \frac{\alpha_S(\mu_R^2)}{1 + \alpha_S(\mu_R^2)bt}, \quad t = \ln \frac{Q^2}{\mu_R^2}. \tag{2.7}$$

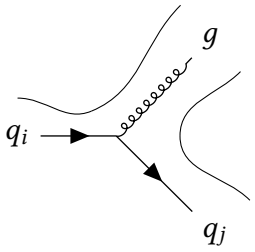


Figure 2.2.: Feynman diagramm for the $q \rightarrow gq$ process with the dominat color lines.

The sign of b plays an important role here, as it defines the behavior for large Q^2. With increasing Q^2 and therefore t the running coupling will become vanishingly small, so color-charge particles can move freely, which is called *asymptotic freedom*. Combining measurements for multiple experiments in Figure 2.1 the concept of asymptotic freedom can be justified.

Another approach to define behavior of α_S is to introduce the scale Λ, at which the coupling would diverge, then we get in leading order

$$\alpha_S(Q^2) = \frac{1}{b \ln(Q^2/\Lambda^2)}. \tag{2.8}$$

If Q^2 approaches Λ^2 the coupling diverges, which highlights the *confinement* property of QCD. It states that no particle carrying (non-singlet) color-charge can be observed in nature. A simple picture would be that with decreasing momentum transfer, or increasing distance between constituents of a colorless hadron, the gluon fields become so strong that a spontaneous creation of a quark-antiquark pair is energetically favorable. As seen in Figure 2.1 the coupling constant will increase with decreasing momentum transfer.

Finally, we need to understand how the color 'evolves' during an interaction i.e. what is the *color-flow* at vertices. Here only the Feynman rule for quark-gluon, 3-gluon and 4-gluon vertex are relevant for these purposes, as other QCD vertices don't change the color structure. The Feynman rule for quark-gluon vertex is given by

$$-ig_s\gamma^a T^a_{ij}. \tag{2.9}$$

If we were to calculate a matrix element containing such a vertex, we would get a term proportional to the square of T^a_{ij}, governing the color-flow. Using the Fierz identity

$$T^a_{ij}T^a_{kl} = \frac{1}{2}\left[\delta_{il}\delta_{jk} - \frac{1}{N_c}\delta_{ij}\delta_{kl}\right] \tag{2.10}$$

we see that color going from quark to gluons is enhanced. Here we use the leading-color limit $N_c \rightarrow \infty$ [3, 4] to even get

$$T^a_{ij}T^a_{kl} \sim \frac{1}{2}\delta_{il}\delta_{jk}, \tag{2.11}$$

for which the Feynman diagram with color lines is shown in Figure 2.2. In this approximation, we replace the $SU(3)$ with $SU(N_c)$ and say that there are an arbitrary number of colors. We can then divide all terms into two categories: leading N_c and subleading $\frac{1}{N_c^2}$ and

then drop all the subleading terms. Now the gluon is always part of two different color lines, one color and one anti-color line. This will be important later in the context of cluster formation, as the color structure of the event will become planar without interference terms.

The color structure of 3-gluon vertex can be represented via Equation 2.4 and using $\mathrm{Tr}\left[T^a T^b\right] = 2\delta^{ab}$ we get

$$if^{abc} = \frac{1}{2}\left[\mathrm{Tr}\left[T^a T^b T^c\right] - \mathrm{Tr}\left[T^b T^a T^c\right]\right], \tag{2.12}$$

which can be seen as an internal quark-loop with three outgoing gluons. Using our knowledge of the quark-gluon vertices, we know how color changes along the 3-gluon vertex. For the 4-gluon vertex it is possible to represent it by three Feynman diagrams each with an internal gluon and two 3-gluon vertices, thus the color-flow is determined.

There exist current research concerning the inclusion of subleading N_c corrections into the parton-shower [5].

2.3. Hydrodynamic Model

The confinement property of QCD prevents us from observing free color-charged particles. Still, we know that asymtotic freedom allows in the case of large momentum transfer and high energies that parton are able to move almost freely. So it is expected that for the right conditions there should be a system in which partons are not confined anymore but instead form a Quark-Gluon Plasma (QGP). We believe that such conditions can be reached at high-energy heavy-ion colliders, like LHC and RHIC.

In this sense, one can start the simulation of central heavy-ion collisions based on QGP. These models are mainly based on hydrodynamics and analogously to thermodynamics try to capture the global behavior of the heavy-ion collision system. Hydrodynamic models evolve the QGP from the initial configuration to a 'freeze-out' scale, after which some hadronization model must produce the final state. An important input parameter is the shear viscosity to entropy ratio η/s, which can be partly determined from perturbative QCD calculations [6, 7] and lattice QCD [8, 9]. This section follows mostly a summary paper [10]. The description of viscous hydrodynamics for heavy-ion collisions with hydrodynamic equations is derived from a conformal fluid by including all terms up to the second order [11].

Starting with the equations for an ideal fluid, the evolution of the system is described by the five equations

$$\partial_\mu T^{\mu\nu}_{\mathrm{id}} = 0, \quad \partial_\mu J^\mu_B = 0,$$

where $T^{\mu\nu}_{\mathrm{id}}$ is the energy-momentum tensor and J^μ_B is the baryon current. This is usually rewritten with the timelike flow four-vector u^μ to

$$T^{\mu\nu}_{\mathrm{id}} = (\varepsilon + \mathcal{P})u^\mu u^\nu - \mathcal{P} g^{\mu\nu}, \quad J^\mu_B = \rho_B u^\mu,$$

with ε the energy density, $\mathcal{P}$ the pressure, ρ_B the baryon density.

For the viscous case, the most used formalism is from Israel-Stewart [12]. The formalism decomposes the energy-momentum tensor as $\mathcal{T}^{\mu\nu} = T^{\mu\nu}_{\text{id}} + \pi^{\mu\nu}$ [11]

$$\begin{aligned} \partial_\mu \mathcal{T}^{\mu\nu} &= 0 \\ S^{\mu\nu} &= \eta\left(\nabla^\mu u^\nu + \nabla^\nu u^\mu - \frac{2}{3}\Delta^{\mu\nu}\nabla_\alpha u^\alpha\right) \\ \Delta^\mu_\alpha \Delta^\nu_\beta u^\sigma \partial\sigma \pi^{\alpha\beta} &= -\frac{1}{\mathcal{T}_\pi}(\pi^{\mu\nu} - S^{\mu\nu}) - \frac{4}{3}\pi^{\mu\nu}(\partial_\alpha u^\alpha) \end{aligned}$$

where $\Delta^{\mu\nu} = g^{\mu\nu} - u^\mu u^\nu$, $\nabla^\mu = \Delta^{\mu\nu}\partial_\nu$ is the local spatial derivative and η is the shear viscosity of the QGP

2.3.1. Elliptic flow

In studies from 2000 at the RHIC [13, 14] it was first shown that the system of colliding nuclei does not behave like a gas, due to a large *elliptic flow* v_2. v_2 or more in general v_n is the nth Fourier coefficient of the final state particle mulitiplicy ordered in the azimuthal angle ϕ with respect to the reaction plane Ψ_n

$$E\frac{\mathrm{d}^3N}{\mathrm{d}p^3} = \frac{1}{2\pi}\frac{\mathrm{d}^2N}{p_T\,\mathrm{d}p_T\,\mathrm{d}y}\left(1 + 2\sum_{n=1}^{\infty} v_n \cos\left([n(\phi - \Psi_n)]\right)\right), \tag{2.13}$$

where E is the energy of the particle, p_T is the transverse momentum and y is the rapidity. If the system could be described as weakly interacting the final state distribution should be isotropic, but as this is not the case one uses hydrodynamics to model the partons as strongly interacting.

3. Monte Carlo Event Generator

Today's particle accelerators like the Large Hadron Collider (LHC) [15] are an important experimental tool to further understand the Standard Model (SM) and to find beyond Standard Model physics. The high energy collisions created at these colliders produce extremely complicated final states, with typically hundreds of particles involved. If one now tries to calculate the whole process from the first principles of SM, one soon realizes that this is far too complicated of a task. In the case of QCD there is also the problem that perturbation theory breaks down for small energies (close 1 GeV or the hadron mass scale) due to confinement, rendering it is impossible to calculate the whole event in this framework.

To tackle the task Monte Carlo event generators were developed. They use the property of factorization [16, 17] to split the event evolution into different steps, representing different momentum transfer regimes. Now, for each step a sophisticated physical model can be used and as the steps are somewhat independent of each other, each method can be improved on its own. One can easily see the impact of a new method or a modification of an existing one on the final state, which helps in the development process.

When the two particles collide, the most energetic interaction is the hard scattering. It is the interaction with the largest momentum transfer. The scale Q^2 of the hard process allows us to perturbatively calculate the matrix elements from first principles. The hard scattering can be e.g. a QCD interaction, some new physics process etc. This is really useful as now we can distinguish new physics from the SM background or just make a simulation run to test if a new model is even detectable in the detector. Following the hard process one uses parton showers to evolve the partons down to the infrared cutoff Q_0^2, still in the regime of perturbation theory. Eventually we reach a scale of $\mathcal{O}(1\,\text{GeV})$ and QCD becomes strongly interacting. Now we can no longer use perturbation theory and instead must use a phenomenological non-perturbative hadronization model, which builds colorless hadrons from the color-charged end state of the parton shower. The last step is the decay of unstable hadrons into stable ones until we reach the final state that is then measured in the detector.

In Figure 3.1 is a schematic picture of the different parts at work in a typical Monte Carlo event generator. The hard process and the parton shower are simulated with perturbative QCD techniques, while the later stages of gluon splitting, cluster formation, color reconnection, cluster fission, hadronization and decay are simulated using non-perturbative phenomenological QCD techniques.

Additionally, there are also some soft additions to basic models. In a normal hard process, a parton from the projectile and one from the target interact, but since the hadron is a composite particle, consisting of a number of partons, additional interactions can occur. These additional interactions are called Multiple Parton-Parton Interactions (MPI), and are in fact required to describe the overall multiplicity distribution and effects of this

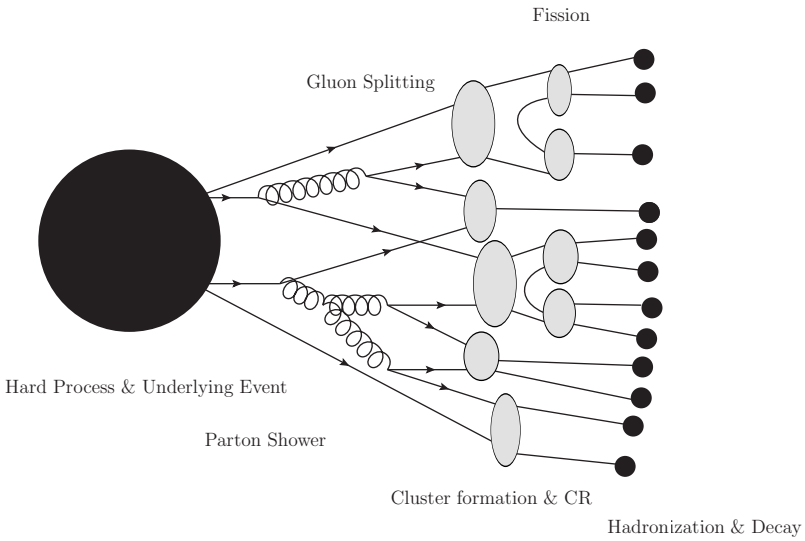


Figure 3.1.: A schematical figure outlining the different stages of a Monte Carlo event generator. The big black blob on the left side, represents a hard process with MPI. The following stages will be explained in this chapter. As this is only a schematic picture, some parts like initial state radiation have been left out for the purpose of clarity.

underlying events like the 'jet pedestal' properly [18]. With MPI, we can now have several initial processes, which could be color connected, but in practice, these MPI are color disconnected as there is no *a priori* manner of deciding the color connections.[1] Color reconnection tries to minimize the overall length of the color lines, and by doing so will connect the previously unconnected MPI.

Some events still cannot be described with a hard process at their core, instead they fall into the category of diffractive interaction. Here no color is exchanged, and the hadrons are only excited and then decay.

This thesis is concerned with improving the general-purpose event generator [19] Herwig [20, 21] which can simulate lepton-lepton, hadron-lepton and hadron-hadron collisions, and extending it into the regime of heavy ion collisions (see chapter 4). It is built on top of ThePEG [22] (also documented in [20, 23]), which provides the programming structure.

Below we outline the various building blocks of MC event generators and the associated physical processes in more detail.

3.1. Hard Subprocess

At the heart of most collisions at colliders is the hard scattering process. It is the process with the largest momentum transfer, thus new particles or heavy and rare particles could

[1] i.e. we are working with the leading color limit

be produced here. The cross section for a scattering process $ab \to n$ is then given by [1]

$$\sigma = \sum_{a,b} \int_0^1 \mathrm{d}x_a \, \mathrm{d}x_b \int \mathrm{d}\Phi_n \, f_a^{h_1}(x_a, \mu_F) f_b^{h_2}(x_b, \mu_F) \times \frac{1}{2\hat{s}} |\mathcal{M}_{ab\to n}|^2(\Phi_n; \mu_F, \mu_R), \tag{3.1}$$

where $f_a^h(x, \mu_F)$ is the parton distribution functions (PDFs) for the parton a to have the momentum fraction x of the hadron h, and depends on the factorization scale μ_F, an unphysical scale that defines the point at which we treat the hadron as factorized in partons. $|\mathcal{M}_{ab\to n}|^2$ is the matrix element squared for a $ab \to n$ process, and depends on the final state phase space Φ_n, the factorization scale and the renormalization scale μ_R and $1/(2\hat{s}) = 1/(2x_a x_b s)$ is the parton flux, with the hadronic center-of-mass energy squared s. It is often convenient to set the renormalization and factorization scale equal to the hard scale $\mu_R = \mu_F = Q$.

Herwig 7 has a built-in PDF provider, but it is also possible to add PDFs provided by LHAPDF [24]. Leading order (LO) and next-to-leading order (NLO) matrix-element calculations can be done internally by Herwig via the matchbox framework [5]. More sophisticated matrix-element MC generators like MadGraph [25] and VBFNLO [26] for higher-order calculations or new physics processes can be interfaced via the Les Houches Accord (LHA) file format.

Generally the matrix-element can be written as the sum over the Feynman diagrams,

$$\mathcal{M}_{ab\to n} = \sum_i \mathcal{F}^{(i)}_{ab\to n}. \tag{3.2}$$

For the class of $2 \to 1$ and $2 \to 2$ processes it is typical to choose only one hard scale $Q^2 = \mu_F = \mu_R$, which also defines the starting scale for the parton shower. The fact that one can move the summation over quantum numbers outside the matrix element, like

$$|\mathcal{M}_{ab\to n}|^2(\Phi_n; Q^2) = \sum_{h_i;c_i} \left|\mathcal{M}^{\{ij\}}_{ab\to n}\right|^2 (\Phi_n, \{h_i\}, \{c_j\}; Q^2), \tag{3.3}$$

allows us to use Monte Carlo techniques to stochastically sample the helicity and color configuration, additionally to the phase space. The choice of helicity and color structure with the hard scale Q define the starting conditions for the subsequent parton shower.

3.2. Parton Shower

Despite steady improvements in the calculations of matrix elements, every fixed order calculation intrinsically has an 'end point', i.e. a maximum number of final state particles. To overcome this shortcoming and fix a lack of multiplicity in early MC event generators, an additional mechanism at the hard scale was introduced [27]. The *parton shower* approximates the higher-order final states by an evolution or Markov chain of quark and gluon radiation down to the cutoff scale $Q_0 \sim 1\,\mathrm{GeV}$ at which confinement is active. We call the final state of the parton shower the *parton-level*.

The two existing approaches to the parton evolution are the collinear QCD-shower which is inspired by the QED-*bremsstrahlung* ($1 \longrightarrow 2$) and the soft dipole shower that splits a parton-pair ($2 \longrightarrow 3$). We will focus in this part on the collinear shower for simplicity, following mostly the elaborations from Buckley et al. [19].

Parton shower algorithms are mostly done in the collinear limit, meaning that most radiation is in the same direction as the emitting particle with a 'small' opening angle. The first step to tackle this problem is then to parameterize the phase space. There are different possibilities for the choice of the parameterization variable besides θ the opening angle between the two emitted particles, like $q^2 = z(1-z)\theta^2 E^2$ the virtuality of the off-shell quark propagator or $p_\perp^2 = z^2(1-z)\theta^2 E^2$ the transverse momentum of the radiated particle with respect to the parents' direction. As they follow the same behavior

$$\mathrm{d}^2\theta \,/\theta^2 = \mathrm{d}^2 q \,/q^2 = \mathrm{d}^2 p_\perp \,/p_\perp^2, \tag{3.4}$$

we can use them interchangeably in the collinear limit, while outside the limit it they have to be treated differently. The general cross section for a collinear parton j to be radiated from a parton i, which is originating from a hard process with cross section σ_0 is [19]

$$\mathrm{d}\sigma \approx \sigma_0 \sum_{\text{partons},i} \frac{\alpha_s}{2\pi} \frac{\mathrm{d}\theta^2}{\theta^2} \,\mathrm{d}z\, P_{ji}(z,\phi)\,\mathrm{d}\phi\,, \tag{3.5}$$

where z is the momentum fraction that parton j receives from i and $P_{ji}(z,\phi)$ are flavor-dependent and in general spin-dependent functions, called Altarelli-Parisi [28–30] splitting kernels. Here are the spin averaged versions, given by

$$\begin{aligned} P_{qq}(z) &= C_F \frac{1+z^2}{1-z}, & P_{gq} &= C_F \frac{1+(1-z)^2}{z}, \\ P_{gg}(z) &= C_A \frac{z^4+1+(1-z)^4}{z(1-z)}, & P_{qg} &= T_R\big(z^2+(1-z)^2\big), \end{aligned} \tag{3.6}$$

with $C_A = N_C$ and T_R is fixed by convention, normally $T_R = \frac{1}{2}$ [1]. σ diverges for an exact forward scattering and for $z \longrightarrow 0$. If all other quantum numbers stay the same, the exact forward scattering is unphysical, as one could not distinguish between a branched pair of partons and an unscattered parton, which is the root of this divergence. The momentum of the radiated parton must be larger than the confinement scale Q_0, so it is tenable to cut the z-parameter space at the boundaries. It should be noted here that in the collinear limit we are unable to describe interference effects. However, if we order the partons in terms of decreasing angles, a constraint known as angular-ordering or strong-ordering [31], the interference effects are restored. A consequence of strong-ordering is that a parton can only be collinear to one other parton.

Our final goal for the parton shower algorithm is to find probability distribution for branching of a parton. We start by defining the probability that any branching from i occurs in the virtuality range from q^2 to $q^2 + \mathrm{d}q^2$

$$\mathrm{d}\mathcal{P}_i = \frac{\alpha_s}{2\pi} \frac{\mathrm{d}q^2}{q^2} \int_{Q_0^2/q^2}^{1-Q_0^2/q^2} \mathrm{d}z\, P_{ji}(z). \tag{3.7}$$

We use the virtuality as parameterization, as it is interchangeable with the splitting angle here, as argued above. Now we want the probability $\Delta_i(Q^2, q^2)$, that no branching above a virtuality of q^2 will occur, with the maximum allowed virtuality Q^2, leading to the differential equation

$$\frac{\mathrm{d}\Delta_i(Q^2, q^2)}{\mathrm{d}q^2} = \Delta_i(Q^2, q^2)\frac{\mathrm{d}\mathcal{P}_i}{\mathrm{d}q^2} \tag{3.8}$$

with $\Delta_i(Q^2, q^2)$ the so-called Sudakov form factor which has the solution

$$\Delta_i(Q^2, q^2) = \exp\left\{-\int_{q^2}^{Q^2}\frac{\mathrm{d}p^2}{p^2}\frac{\alpha_s}{2\pi}\int_{Q_0^2/p^2}^{1-Q_0^2/p^2}\mathrm{d}z\, P_{ji}(z)\right\}. \tag{3.9}$$

With Eq. (3.8) we have all the necessary ingredients for the distribution of the branching with the highest virtuality below Q^2. The shower algorithm then iteratively attaches partons to the hard process. We choose q^2 according to $\frac{\mathrm{d}\Delta_i(Q^2,q^2)}{\mathrm{d}q^2}$, which is the branching with the highest virtuality, and then we repeat this, now replacing Q^2 with q^2 and so on, until q^2 is smaller than Q_0^2. The final step is the generation of the momenta of final state partons, which could be just done by following the splitting process, but since we deal with on-shell partons we must account for recoil to guarantee momentum conservation. This can be done with a procedure called *momentum reshuffling*, either locally after each splitting or globally at the final state of the parton shower.

In Eq. (3.5) we only considered the collinear regime, which normally does not contain a lot of soft gluons. The collinear limit made it easy to develop a Markov chain for the QCD radiation, as following strong ordering the radiated partons are only collinear to one other parton. For soft gluons however interference is of an important role. The interference can largely be dealt with by doing the calculation for the emission of two soft gluons from the two legs of a collinear splitting. It can be shown, that this can be treated as a single gluon being emitted before the splitting, thus one could say the soft gluons see only the total color-charge of a collinear system. To solve this problem coherently one can use the opening angle instead of the virtuality [32]. By doing so, the parton shower algorithms, based on the collinear limit, can also deal with soft radiations. This is the reason why Herwig uses an angular-ordered shower algorithm.

Until now, we only considered final state showering after the hard process, but the colliding partons can also produce *initial state radiation*, simulated by evolving the colliding partons backwards in time. The partons and gluons from the initial state radiation will also further shower and contribute to the final state. The model for ISR was developed by Sjöstrand [18]

$$\Delta_i(Q^2, q^2; x) = \exp\left\{-\int_{q^2}^{Q^2}\frac{\mathrm{d}p^2}{p^2}\frac{\alpha_S}{2\pi}\int_{Q_0^2/p^2}^{1-Q_0^2/p^2}\mathrm{d}z\, P_{ij}(z)\frac{x/z f_j(x/z, p^2)}{x f_i(x, p^2)}\right\}, \tag{3.10}$$

where x is the momentum fraction of the hadron that goes into the main hard process, x/z is the momentum fraction of the parton before it emits QCD-radiation, $f_i(x, p^2)$ is the PDF of the parton i in the hadron. Comparing this equation with Eq. (3.9) one can see that the Sudakov form factor for the FSR is just modified by a ratio of the PDF times the momentum fraction.

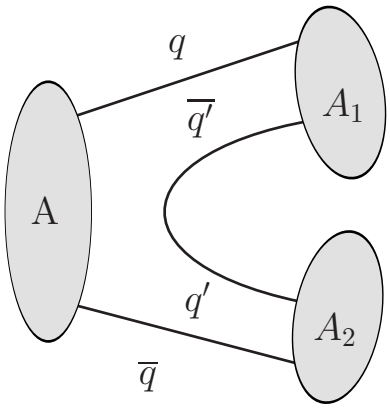


Figure 3.2.: Schematic figure of the cluster fission process. A new quark-antiquark pair q' and $\overline{q'}$ is created non-perturbatively and together with the 'old' quarks they form the two new clusters A_1 and A_2.

The cutoff scale for heavy quarks, like c, b and t is the mass m_Q if i is lager then Q_0. We can regulate the divergent behavior of heavy quarks with $\theta \leq \theta_0 = m_Q/E$, meaning that heavier quarks are less collinear.

3.3. Hadronization

Starting from the partonic final-state produced by the shower, the event generator has to perform the evolution to hadrons and leptons and perform decay processes for short living particles. This final state will then be measured in the detector, meaning it is the 'real' final state. The models used for hadronization are not based on first principles, as for the non-perturbative region. One approach to attack the problem using first principles is lattice QCD, a discretized quantum field theory formulated in Euclidean spacetime, so it cannot be used for relativistic processes like the ones we are concerned with. The two major hadronization models used in general-purpose event generators are the string model developed by the Lund group and used in Pythia and the cluster model which is used in Herwig and described here further. The fundamental concept of the string model is *linear confinement*, stemming from 'quenched' lattice QCD [2], in which the field strength between two color connected partons increases linearly with the separation distance between the two partons, leading to a string. For the cluster model the QCD property of *preconfinement* is utilized. The main difference in creating the hadronic final state is that the string model transforms the partonic state directly in hadrons and the cluster model uses different intermediate stages, clusters, to reach the hadronic final state.

As the model tunes cannot be performed independently, the parameters of the non-perturbative hadronization model must be retuned if there are any modifications to the hard-process and the parton-shower.

3.3.1. Cluster Model

The theoretical basis of the cluster model [33] is preconfinement [34]. It states that we can arrange the partons at the cut-off scale Q_0 in color singlets (clusters) with an asymptotically universal invariant mass distribution. As long as the energy scale hierarchy $Q \gg Q_0 \gg \Lambda_{QCD}$ holds the mass distribution is only dependent on Q_0 and Λ_{QCD} and is independent of Q and the exact hard process. A feature of the mass distribution is that it is power suppressed for large masses, and we expect a universal multiplicity distribution $n/\langle n \rangle$ (KNO scaling [35]).

In the large-N_C limit gluons are always part of two different color lines, which will lead to a planar color-topology [3]. To generate colorless clusters, the gluons from the parton state have to be split in quark-antiquark pairs. Now the two color-connected quarks can form color-singlets. At this state color reconnection can be applied, which is explained in more detail in section 3.5. The evolution of clusters with a mass larger than some typical cut off scale 3-4 GeV is preformed by *cluster fission*, which splits these heavy clusters into two new clusters and so on until the mass is below the cutoff scale. This splitting is induced by a non-perturbative creation of a quark-antiquark pair, one of them respectively in one of the two new clusters, which now consist of one 'old' quark and one 'new' antiquark and vice versa, as outlined in Figure 3.2. The direction of the new clusters follows along the axis of the constituent partons of the old cluster. Low-mass clusters can instead directly decay through single-hadron decay, while giving some momentum to a nearby cluster. This leads to hadrons carrying almost the whole momentum fraction of the jet. The remaining clusters can be treated as excited mesons, that isotropically decay in two hadrons. Most clusters will undergo this decay process, but it won't lead to high momentum fraction hadrons as for the single-hadron decay, justifying its implementation. Choosing the hadrons is straightforward done accordingly to the hadron mass and the available phase space [36].

3.3.2. Hadron and τ decays

The primary hadrons from the hadronization state may be unstable (normally meaning $c\tau \leq 10\,\text{mm}$) and must be therefore further decayed. The decay model can have a significant impact on the final state particles, especially on those which are at the end of the decay chain (feed-down).

3.4. Underlying Event

When describing central hadron-hadron collisions, there can be secondary soft interactions, called the *Underlying Event* (UE). These soft interactions account for things like jet *pedestals* and possibly additional back-to-back jet pairs. The jet pedestal is the observed effect that an event with a hard jet has a higher underlying background activity than an event with no hard jet. This can be interpreted as impact-parameter dependency, as small impact parameter events have a higher probability of producing a hard jet and also of producing secondary interactions, compared to peripheral events.

One model for UE are *Multiple Parton Interactions* (MPI), first introduced by Sjöstrand and van Zijl [37], where instead of having just a single hard process, a secondary softer perturbative or non-perturbative interactions, depending on the transversal momentum can also occur.

This is somewhat justified, if we see the colliding hadrons as a collection of partons, so additional interactions besides the main one should be possible. If we look at the t-channel gluon propagator for a $2 \to 2$ QCD process, one observes that it almost goes on-shell for small $p_\perp$ and the behavior of the differential cross section is

$$\mathrm{d}\sigma_{2\to 2} \propto \frac{\mathrm{d}t}{t^2} \sim \frac{\mathrm{d}p_\perp^2}{p_\perp^4}. \tag{3.11}$$

This cross section diverges for small $p_\perp$, which has to be regulated. First we realize that if a single hadron-hadron scattering contains two parton-parton collisions, this event will be counted twice for $\sigma_{2\to 2}$, thus we can also write

$$\sigma_{2\to 2} = \langle n \rangle \sigma_{\mathrm{tot}}, \tag{3.12}$$

with $\langle n \rangle$ the average number of perturbative parton-parton scatterings in a single hadron-hadron collision. Additionally, momentum conservation and *color smearing* also regulate the number of MPI. Momentum conservation is straightforward, but color smearing means that for low $p_\perp$ the wavelength of the parton goes like $\sim 1/p_\perp$ and will at some point exceed the confinement scales. The partons will not 'see' the color of other particles, which provides a good way to define an infrared cut of for MPI. A typical value would be the proton radius r_p, leading to $p_\perp \approx \hbar/r_p \approx 0.3\,\mathrm{GeV} \approx \Lambda_{\mathrm{QCD}}$.

In case of the perturbative MPI we apply the parton shower as for the primary hard collision.

3.5. Color Reconnection Model in Herwig

The different stages of an MC event can be seen as being somewhat independent of each other, if we don't consider the fact, that parameter tunes of the different elements are not independent. However, MPI produces multiple scattering processes, which have randomly assigned color-connections according the leading-color approximation. Since QCD has $N_C = 3$ and not infinite and since the multiple partonic interaction come from the same colorless hadrons, one would expect that the color structure should in fact have some form of coherence and not be independent. If we only allow normal cluster formation and hadronization etc. the MPI would stay independent. In Figure 3.3 one can see on the left side an MPI event, for which the single partonic interactions are independent. One sees that the color lines are really long and also intersect each other. We could generally say, that if the density of color-charge is high one expects that the MPI are not independent. To find a better description of the color topology *color reconnection* (CR) can be applied, first introduced by Sjöstrand and van Zijl [38]. A possible color topology after color reconnection can be found on the right side in Figure 3.3. Now the MPI are not independent anymore.

[2] Thanks, Stefan Gieseke for providing the figure.

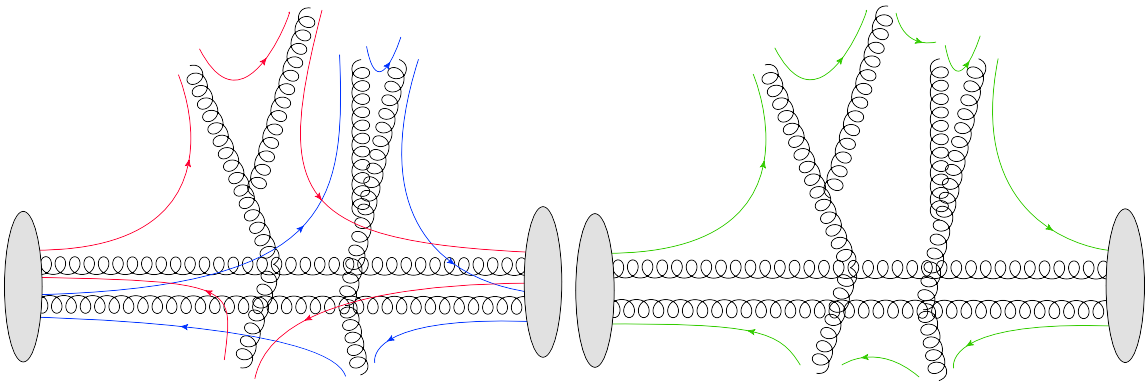


Figure 3.3.: In the two pictures an event with MPI is shown, where the two blobs represent the hadron remnant, flying away from the interaction. On the left side the event is with the leading-color structure, if the two partonic interactions are independent of each other, represented with the two different colors. If we compare this to the right picture, after color reconnection, the color lines are now shorter, which matches with the idea that not independent colored particles that are close in phase space should be color-connected. Shorter color lines will mean result in fewer charged particles but with more momentum.[2]

In the case of the cluster model of Herwig, *plain mesonic color reconnection* [39] tries to minimize the sum of the invariant mass $M^2 = \left(p_1^2 + p_2^2\right)$ of the color singlet clusters. The aim is then to find cluster configuration with lower sum of invariant cluster masses

$$\lambda = \sum_{i=1}^{N_{cl}} M_i^2 \tag{3.13}$$

then the original configuration, with N_{cl} the number of clusters. This is done by randomly picking a cluster A and then going through the list of available clusters B. For each combination the mass of the original configuration $M_A + M_B$ is calculated and compared the $M_C + M_D$ the mass of the new configuration. If at least one new combination CD has a lower mass then the original one, the combination with the lowest mass reconnects with a probability ρ_R.

A recent addition to Herwig was the introduction of *baryonic color reconnection* [40]. This model reconnects the clusters in such a way, that also baryonic clusters, consisting of three quarks or three anti-quarks, can be created. As the invariant mass is high for baryonic clusters, instead the *rapidity*

$$y = \frac{1}{2} \ln\left(\frac{E + p_z}{E - p_z}\right), \tag{3.14}$$

with respect to the z-axis, is considered here as the key variable. A value of $y = 0$ corresponds to a direction orthogonal to the z-axis and $y = \infty$ would correspond to the z-direction. We can think of the rapidity as having an exponential rising angular-resolution with increasing closeness of the direction of the particle to the z-axis. Meaning that the difference in rapidity Δy for two central particles propagating in directions which differ by $\Delta\theta$ is much smaller than the difference for two particles with the same $\Delta\theta$ propagating

close to the beam direction. The baryonic color reconnection model begins by boosting to the CM frame of a random cluster A, now the constituent partons propagate back-to-back. We define the z-axis in the direction of the anti-quark $\overline{q}_A$ and calculate the rapidity for the quark-anti-quark $q_B\,\overline{q}_B$ of each cluster B. If the quark-anti-quark pair moves also back-to-back, meaning either $y(q_B) < 0 < y(\overline{q}_B)$ (Baryonic case) or $y(\overline{q}_B) < 0 < y(q_B)$ (Mesonic case), this cluster is considered for color reconnection. Then we pick the two clusters B and C of all the ones considered with the highest value of absolute rapidity $|y(q_i)| + |y(\overline{q}_i)|$. In the case that both clusters B and C are baryonic, baryonic color reconnection with probability ρ_B takes place. In the case that the cluster with the highest absolute value of rapidity is mesonic, the normal mesonic reconnection is performed with probability ρ_R.

3.6. Minimal Bias Model

A minimal bias measurement means at its core, that as trigger only a minimum number of hits in the observed experimental region is required. No other constrains are set.

The Minimal Bias model uses only QCD events for the hard process or additionally single and double diffractive events. It uses a dummy matrix element for its hard process, which has to be tuned to get the best results on the minimum bias data.

3.7. Diffraction

There are some types of observables which cannot be explained or modeled with a normal hard process, such as events with large *pseudorapidity gap* $\Delta\eta^F$. Pseudorapidity is defined as

$$\eta = \frac{1}{2}\ln\left(\frac{|\mathbf{p}| + p_z}{|\mathbf{p}| - p_z}\right) = \operatorname{arctanh}\left(\frac{p_z}{|\mathbf{p}|}\right) = -\ln\left[\tan\left(\frac{\theta}{2}\right)\right]. \tag{3.15}$$

In the case of massless particles or highly relativistic particles pseudorapidity η and rapidity y are approximately similar. $\Delta\eta^F$ is then defined as the larger one of the two gaps from the border of the available range of the detector to the first measured hit. As an example, if there is only activity at one point at pseudorapidity $\eta = -1$ in the detector with pseudorapidity range $\eta = \{-4.5, 4.5\}$, then gap would be $\Delta\eta^F = 5.5$.

Diffraction is mostly modeled by an exchange of a reggeon or a pomeron, which are color-singlet fluctuations approximated by $f\overline{f}$ and $q\overline{q}$, and then a splitting of the exited hadron into a quark-diquark pair. In Herwig, clusters are then formed from the quark systems and the normal Monte Carlo event generator machinery for clusters physics and subsequent processes is applied [41].

There are three different types of diffraction. In the case of single diffraction, one hadrons stays intact and the other dissociates, like $A + B \longrightarrow A' + X$. For double diffraction both hadrons dissociate, like $A + B \longrightarrow X + X$ and finally for central diffraction there is an active region in between the intact hadrons, like $A + B \longrightarrow A' + X + B'$. The ticks on the right-hand side are indicating that the hadrons stays intact, but might have different momentum.

The cross section for diffractive processes can be derived from Regge theory (see e.g. [42]). In minimum bias events, diffraction occurs at a rate of 20 % to 25 %.

The spectrum of diffraction with respect to the invariant mass of the exchanged virtual particle follows $\mathrm{d}M^2 / M^2$.

4. Glauber Model For Heavy Ion Collisions

The Glauber model is used to simulate purely geometrical quantities in heavy-ion collisions which can then be "transformed" into measured quantities. These quantities include the number of collisions N_{coll} or the number of participating nucleons N_{part}, also called wounded nucleons. The main idea is that we can factorize the nucleus-nucleus collisions into multiple nucleon-nucleon collisions, which we can describe with the usual Herwig machinery.

Glauber was the first to systematically study the scattering properties of many-body nuclear systems [43, 44]. His work was concerned with the theoretical description of the heavy-ion experiments of his time. The center-of-mass energy of these experiments would nowadays be considered low, which means it was enough for Glauber to use the elastic scattering of the nucleons and to disregard inelastic processes. Later Białas, Bleszyński and Czyż [45] extended the model also to inelastic nucleon-nucleon scattering, allowing the framework to describe high-energy heavy-ion collisions.

Today the Glauber model is used mainly in either the optical limit or the Monte-Carlo approach. Generally both approaches use the *high-energy limit*, which means that we assume that the nucleons inside the nucleus move in straight lines and that we can treat them as independent or incoherent.

In this work the Monte-Carlo approach is used, but for basic understanding and completeness the optical limit is described briefly too.

4.1. Optical Limit Approach

In the optical limit approach we use some simplifying assumptions, which allow us to calculate the geometrical quantities analytically. The main assumptions are the high-energy limit, explained briefly above, and the eikonal approximation. The eikonal approximation means that we treat the nuclei as if they have smooth densities.

So we can write a term for the probability of finding a given nucleon in a flux tube (see Figure 4.1) at distance $\mathbf{s} = \begin{pmatrix} x_A \\ y_A \end{pmatrix}$ form the center of the projectile A by integrating over the beam direction z_A as

$$\hat{T}_A(\mathbf{s}) = \int \hat{\rho}_A(\mathbf{s}, z_A)\, \mathrm{d}z_A\,, \tag{4.1}$$

with $\hat{\rho}_A(\mathbf{s}, z_A)\, \mathrm{d}z_A$ the normalized nucleon density. With the same term for the target B, we can define the thickness function

$$\hat{T}_{AB}(\mathbf{b}) = \int \hat{T}_A(\mathbf{s})\hat{T}_B(\mathbf{s} - \mathbf{b})\, \mathrm{d}^2 s\,, \tag{4.2}$$

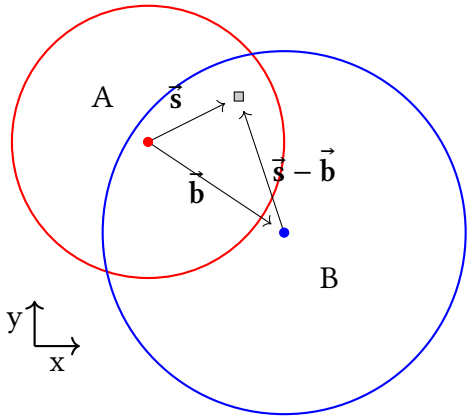


Figure 4.1.: Beam view of the thickness function. The gray square is the flux tube at distance **s** from the center of A. The colored circles give the rough approximation of the expansion of the nuclei. Inspired by Miller et al. [46].

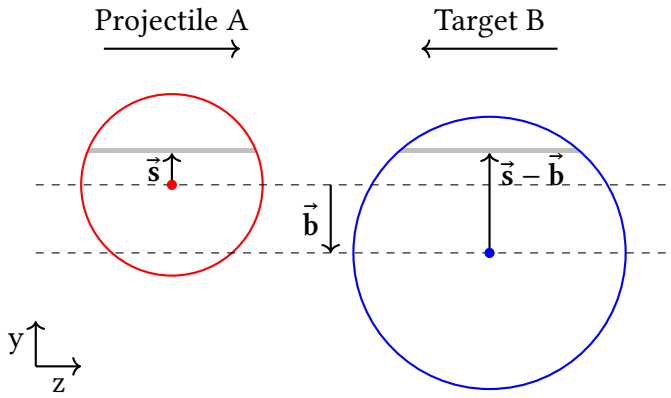


Figure 4.2.: Side view of the thickness function. The gray square is the flux tube at distance **s** from the center of A. The colored circles give the rough approximation of the expansion of the nuclei. Inspired by Miller et al. [46].

which can be understood as the effective overlap for a specific nucleon from the projectile to interact with a specific nucleon in the target. In Figure 4.1 and Figure 4.2 a graphical representation of this process is given. This gives us the probability to find n nucleon-nucleon-interactions with the binomial distribution

$$P(n, \mathbf{b}) = \binom{AB}{n} \left[\hat{T}_{AB}(\mathbf{b})\sigma_{\text{inel}}^{NN}\right]^n \left[1 - \hat{T}_{AB}(\mathbf{b})\sigma_{\text{inel}}^{NN}\right]^{AB-n}, \tag{4.3}$$

using the fact that the probability for one interaction is $\hat{T}_{AB}(\mathbf{b})\sigma_{\text{inel}}^{NN}$. Here elastic interactions are ignored, because they only lead to a very small energy loss. Thus, we only need the inelastic nucleon-nucleon cross-section $\sigma_{\text{inel}}^{NN}$. With the knowledge of the mean value of the binomial distribution, this gives us the number of collisions

$$N_{\text{coll}}(b) = \sum_{n=1}^{AB} nP(n, b) = AB\hat{T}_{AB}(b)\sigma_{\text{inel}}^{NN}. \tag{4.4}$$

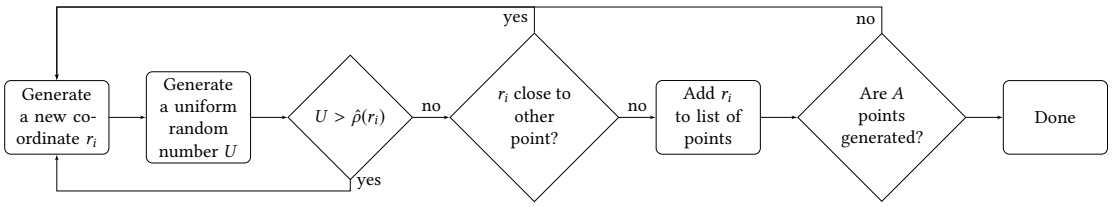


Figure 4.3.: Decision diagram of the creation process nucleus A. Close to another point, means that the distance is smaller than 0.9 fm. $U \in [0, 1]$

For the participating number of nucleons we get, see [45]

$$N_{\text{part}}(\mathbf{b}) = A \int \hat{T}_A(\mathbf{s}) \left\{ 1 - \left[1 - \hat{T}_B(\mathbf{s} - \mathbf{b}) \sigma_{\text{inel}}^{NN} \right]^B \right\} \mathrm{d}^2 s + B \int \hat{T}_B(\mathbf{s}) \left\{ 1 - \left[1 - \hat{T}_A(\mathbf{s} - \mathbf{b}) \sigma_{\text{inel}}^{NN} \right]^A \right\} \mathrm{d}^2 s \,. \tag{4.5}$$

The eikonal approximation leads to some drawbacks, as some effects, e.g. nuclear shadowing, can't be captured by this. The shadowing effect is when one nucleon is hidden behind another. It has been shown [46] that geometrical properties like N_{coll} and N_{part} are well simulated, which justifies the use of the optical limit for application in heavy ion collisions.

4.2. Glauber Monte Carlo Approach

The basic idea of the Monte Carlo approach is to build the collision of two nuclei or one nucleus and proton out of multiple proton-proton or proton-neutron collisions.

The first step is to simulate the nucleon positions inside the target and projectile nucleus. Then we pick a random impact parameter $\mathbf{b}$ from the distribution $\mathrm{d}\sigma/\mathrm{d}b = 2\pi b$ and search for the interacting nucleons pairs based on an interaction model. There are different type of interacting pairs, i.e. primary and secondary hard interactions sometimes also called absorptive interactions and possibly a third kind of interactions similar to diffraction.

A primary interaction consists of two colliding nucleons which haven't interacted before. In a secondary interaction at least one nucleon has interacted before in a primary or secondary interaction. After all interaction pairs are found, we use Herwig to simulate the NN-interactions, with a different setup for the primary and for the secondary interaction. All these events are then collected and merged together in one heavy-ion event, which can then be further analyzed.

The third type of interactions are only possible if *fluctuations* are introduced in the interaction model. This will not be treated in this thesis so for now on we only consider primary and secondary hard interactions. We want to note that Gribov pointed out the importance of this third kind of interactions [47].

4.2.1. Simulation of Nucleus According to Woods-Saxon Potential

The nucleus is build out of nucleons, which are somehow distributed around the center of the nucleus. In the literature a spherical three-parameter Fermi model is commonly used

for the nuclear density distribution [48]

$$\rho(r) = \rho_0 \frac{1 + w\frac{r^2}{R_e^2}}{1 + \exp\left(\frac{r - R_e}{a_e}\right)} \tag{4.6}$$

with ρ_0 an overall normalization parameter to ensure nucleon number conservation $\int \mathrm{d}^3 r\, \rho(r) = A$, R_e the nucleus radius, a_e the surface thickness parameter and w the Fermi parameter. Let's define $n_e(r) = 4\pi r^2 \rho(r)$ for simplicity. From nucleus-electron experiments [49] we know the overall density distribution, but to find the appropriate parameters for R_e and a_e we have to consider that the nucleons also have a density distribution. By convolving ρ with the nucleon density distribution one gets [48]

$$\begin{aligned} R_e &= (1.12A^{1/3} - 0.622A^{-1/3})\,\mathrm{fm}, \\ a_e &= 0.46\,\mathrm{fm}. \end{aligned} \tag{4.7}$$

With $w = 0$ this distribution is also called the Woods-Saxon potential, which is used in this work. We can do this because the focus is on heavy-ions like xenon, gold and lead. For smaller nuclei, the w parameter is important, as it accounts for deviations from the spherical shape, which are prevalent in those cases.

Because of Pauli's principle nucleons can't occupy the same spatial point, i.e. there must be some minimal distance between them. The simplest approach is just to introduce a nucleon-nucleon repulsion radius d as the minimal distance allowed between nucleons. Doing so shifts the nucleon distribution a bit away from the center. To compensate for this effect, the parameters of $\rho(r)$ have to be retuned. The sophisticated approach would be to use a realistic Gaussian NN-correlation, but it has been shown that for a hard repulsion with $d = 0.9\,\mathrm{fm}$ and

$$\begin{aligned} R_e &= (1.1A^{1/3} - 0.656A^{-1/3})\,\mathrm{fm}, \\ a_e &= 0.459\,\mathrm{fm}, \end{aligned} \tag{4.8}$$

the two approaches yield identical results [50].

In the simulation process we use $\hat{\rho}(r_i) = \rho(r_i)/\rho_0$ instead of $\rho(r)$ for simplicity, as it is upper-bounded by 1. The first step is to generate a random point r_i in a cube of length $3R_e$ and then reject it if $U > \hat{\rho}(r_i)$ where U is a uniform random number. Otherwise, we keep the point as a possible candidate and check if there is already another nucleon occupying this space. We use hard repulsion to determine interference in space. If the distance d between the newly created point to the nearest already created point is smaller than $d = 0.9\,\mathrm{fm}$, the newly created point is rejected. If not, the point is added to the list of nucleon points. This is repeated until A nucleon positions are generated. The whole process is visualized in Figure 4.3.

To verify that we get the desired result with the simulation, the simulated data is plotted together with the Woods-Saxon potential with parameters from Eq. (4.8) in Figure 4.4. It is apparent that the simulations yields proper results, and we can use this approach going forward.

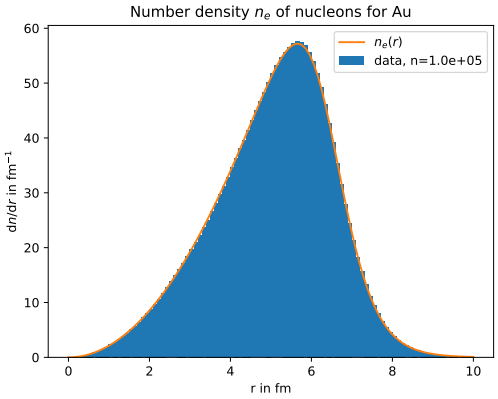


Figure 4.4.: The generated nuclear distribution by the Monte-Carlo approach, labeled 'data', weighted by $1/n/\text{bin\_width}$ with the number of generated nucleons n, is compared with n_e. For the Monte-Carlo results, the parameters from Eq. (4.8) are used, while for the analytic result n_e the parameters from Eq. (4.7) are used.

In this work the nucleons are differentiated between protons and neutrons. We want to randomly assuming one of the two hadron types to the generated points. Using a uniform random number $U \in [0, 1]$ and the criteria

$$U < \big(Z - n_\mathrm{p}\big)/\big(A - \big(n_\mathrm{n} + n_\mathrm{p}\big)\big), \tag{4.9}$$

with the number of already assigned protons n_p and respectively for the neutrons n_n, we can assign each point to be either a proton, if the criteria is true, or else to be a neutron. This guarantees that the nucleus is built up out of the right number of protons and neutrons.

4.2.2. Sampling Impact Parameter

In collider experiments we can't fix all parameters *a priori*, such as the impact parameter **b**. Therefore to get realistic results we need to simulate **b** according to the distribution $\mathrm{d}\sigma \propto 2\pi b\,\mathrm{d}b$, which can be derived geometrical considerations. Here we only need to define a maximum value $b \leq b_\mathrm{max}$, which should not be too large, as this would decrease performance. The proper way to do this, would be to calculate the overlap for b_max and set it to a small number ϵ

$$\begin{aligned}\hat{T}_{AB}(\mathbf{b}) &= \int \mathrm{d}^2 s\, \hat{T}_A(\mathbf{s})\hat{T}_B(\mathbf{s}-\mathbf{b}) \leq \epsilon \\ &= \int \mathrm{d}s\,\mathrm{d}\phi\, 2\pi s \int \mathrm{d}z_A\,\mathrm{d}z_B\, c^2 \\ &\qquad \left[\left(1+\exp\left(\frac{\sqrt{s^2+z_A^2}-R_e}{a_e}\right)\right)\left(1+\exp\left(\frac{\sqrt{s^2-2sb\cos\phi+z_B^2}-R_e}{a_e}\right)\right)\right]^{-1}\end{aligned} \tag{4.10}$$

with some normalization constant c. As $\hat{T}_{AB}$ is proportional to the number of possible interactions, we could in fact restrict b sensibly with this.

A more computationally efficient method, which is implemented in this thesis, is to define $R_{\max}$ as the expansion of the nucleus. We say that for $r > R_{\max}$ there are no nucleons, meaning no interactions are possible. We can use some $\epsilon_{R_{\max}}$ to calculate the maximum interaction radius $R_{\max}$, it gives the percentage of nucleons inside the sphere with radius $R_{\max}$.

$$\int_0^{R_{\max}} \mathrm{d}^3 r\, \rho(r) = \int_0^{R_{\max}} \mathrm{d}r\, n_e(r) = \int_0^{R_{\max}} \mathrm{d}r\, 4\pi r^2 \rho_0 \frac{1}{1 + \exp\left(\frac{r - R_e}{a_e}\right)} = A\epsilon_{R_{\max}}. \tag{4.11}$$

We calculate ρ_0 numerically and use `scipy.optimize.fsolve` to find the root of the equation. In the code $\epsilon_{R_{\max}} = 0.99$ is used. If the two colliding nuclei are different species, then the maximum impact parameter $b_{\max} = R_{\max;1} + R_{\max;2}$, while if the nuclei are identical $b_{\max} = 2R_{\max}$.

It is important to state here that this is only a rough estimate for the actual interaction radius of two nuclei. But this is not of any consequences for the simulation with the Monte Carlo method, as we can choose a value for $\epsilon_{R_{\max}}$ in such a way that there is no clear cut in the impact parameter distribution. In case of simulating smaller nuclei or anything for which really peripheral event could be of great importance one should keep these simplifications here in mind and if needed adjust the $\epsilon_{R_{\max}}$ parameter.

4.2.3. Interaction Models

With the impact parameter sampled we now have the spacial coordinates of the nucleons in the nuclei with respect to the beam axis. So if we insert this information into a simulation software, it would be possible to trace the movement of the nucleons along the beam axis. In the following the beam axis is set to the z-axis. While the nucleons from target and projectile move past each other, there is a minimun distance d_{ij} each nucleon i from the target has with respect the each nucleon j in the projetile. Using the high-energy limit we can infer d_{ij} just from the geometrical configuration of the target and projectile, as it states that the nucleons move in straight lines during the collision. Based on this minimal inter-nucleon distance, we have to decide which nucleon pairs interact and also which kind of interaction they perform.

The objective of the interaction models is to reproduce the total AB cross section, by finding collision pairs, distinguished by different collision types.

To find the interacting pairs, two models are presented in the following. The *black-disc* model is the simplest approach, which is implemented in this work. The *gray-disc* model is more sophisticated and allows for a more fine tuned distinction between different interactions. More interaction models can be found in Pi [51].

4.2.3.1. Black-Disc Model

The black-disc model is the most simplistic approach. If nucleons are closer then $d \leq \sqrt{\sigma_{\text{inel}}/\pi}$ they are colliding absorbtivly with probability 1. With the unity probability

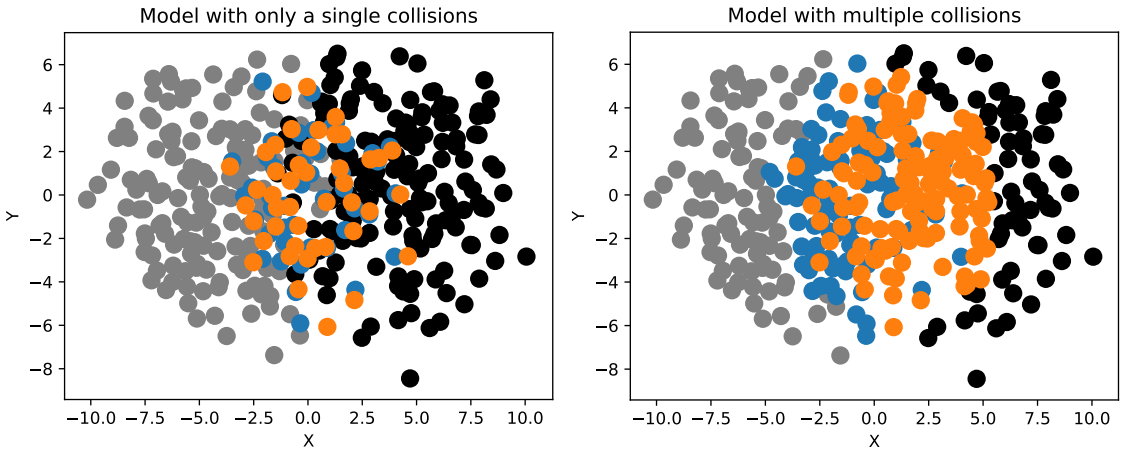


Figure 4.5.: This two figures show two colliding nuclei, whose centers are seperated by an impact parameter b. The nuclei are moving in opposite direction along the z-axis, The spectator nucleons are colored gray and black, while the nucleons which can interact are colored in orange and blue. In the left figure a collision model with only single NN-interactions is shown. We can see that with a multiple NN-interaction model, as seen in the right figure, we get a lot more collision pairs. For the multiple NN-interaction model not every pair will be an actual source of particles, but rather they are collision candidates.

no fluctations can occur, which would be important if we'd have multiple secondary interactions types (see subsubsection 4.2.4.2) at work. Furthermore, without fluctuations the model will not properly describe the overall cross sections. The inelastic cross section σ_{inel} can be reproduced, but the elastic σ_{el} and diffractive σ_{diff} cross section cannot be reproduced. As the inellastic cross section is the most important one for high-energy collisions, the dark disc model can be justified as a first approximation of a more developed nucleon-nucleon model.

4.2.3.2. Gray-Disc Model

The idea of the gray-disc model [52] is to use an opaque interaction area of radius d_{gray}. Two nucleons which are closer then the interaction radius will then interact with the probability a. The values of a and d_{gray} can be tuned so that the gray-disc model regenerates the all the different cross sections. With this model fluctuations are introduces, so the third types of interactions would be possible. Without fluctuation the first type of secondary interactions will 'eat' up everything leaving nothing for the second type. In this thesis only one type of secondary interactions is used, thus the gray-disc model has not been implemented. This is left for future development of the framework.

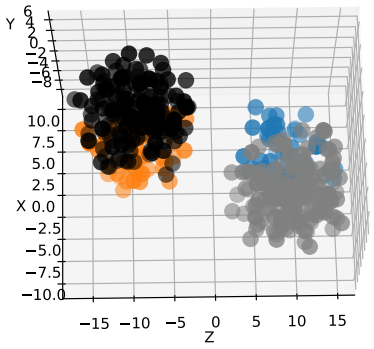


Figure 4.6.: 3-dimensional figure of two colliding nuclei moving along the z-axis. The coloring of the nucleons is the same as in Figure 4.5. The underlying model is the multiple NN-interaction model.

4.2.4. Primary and Secondary Hard Interaction

With a list of all possible interactions pairs given by the black-disc model, we can proceed to determine the actual NN-interactions. Before we perform the heavy-ion collisions, the nucleons inside the nuclei have not interacted in the sense of a hard scattering process, we could say they are untouched. We can think of three possible configurations for an NN-interaction. For each NN-pair, both nucleons could either be untouched, or one of them could have interacted before and the other not, or both of them interacted before. The first case is called a *primary hard interaction* and the second and third case are *secondary hard interactions*. The secondary interactions are then classified as `secondary-side`, where `side` can be either *left* or *right* depending on the side of the untouched nucleon or `side` is *both* in case of the third interaction type. Depending on this configuration the NN-interactions have to be simulated differently. This is discussed further in subsubsection 4.2.4.2.

The important question is how do we decide which NN-pair is a primary hard interaction and which is a secondary hard interaction. Therefore we order the list according to the impact parameter of the collision, saying that NN-collisions with smaller impact parameter are harder ones. The hardest collisions will form the backbone of the event, and they will be the primary hard interaction and the less hard interactions are then the secondary ones. To classify the interactions we go through the ordered list NN-pairs, and check for the criterias given above. So the first pair in the list will always be a primary interaction, the following pairs can then fall into one of the four classifications. In Figure 4.5 on the left side we can see all the primary interaction pairs colored in orange and blue, while on the right side we have presented the same event, now with all the possible interaction pairs. In Figure 4.6 the same event is shown in a 3-dimensional representation.

4.2.4.1. Primary Interactions

To simulate the primary interaction we use a MinBias event from Herwig without diffractive events.

It is useful to have a measure for the *hardness* h(evt) of events from Herwig, as it is impossible to manually set the impact parameter with the default implementation of Herwig. We determine the hardness by the sum over transverse momentum of all final state particles

$$h(\text{evt}) = \sum_{i=\text{final state}} p_{T,i}. \tag{4.12}$$

Other definition of the hardness are possible, like the transverse energie but are not considered here. The user could modify this easily by adding a different `hardness` function to the Python code and change the appropriate function-call in the staticmethod `hardsort` in the `Collision` class.

For the value of the sample size N_{sample} which sets the size of the pool of Herwig events we have different possibilities. The events in the pool can all be generated with the same settings for Herwig, as we know the energy of the primary hadrons. One could just set the sample size to the number of primary interactions N_{primary}, but this would make a treatment of hardness almost unnecessary. With such a approach the only effect would be the spatial distribution of the events. On the other hand we can do oversampling by setting the sample size to

$$N_{\text{sample}} = N_{\text{primary}} + N_{\text{secondary}}. \tag{4.13}$$

To get the primary events from this pool, we choose the N_{primary} hardest events, governed by the hardness function. In the PISTA [53] they have a different approach, in which they count the number of collisions the nucleons of the collision pair are part of. With this they set the sample size for the pool of events differently for each primary collision pair.

4.2.4.2. Secondary Interactions

In a normal min-bias event there is at least one hard interaction between the partons from both nucleons. These hard interactions are the starting point for the further event generator steps, leaving behind hadron remnants. If we consider the high-energy limit and thus straight trajectories for the nucleons, the remnants should be able to interact again. For this to work we need to ensure energy and momentum conservation for secondary interactions.

We follow the timeline of the implementation of different approaches. In the first approach only the `secondary-left` and `secondary-right` collision were considered dropping all of type `secondary-both`. Similar to ideas in other papers [52, 53], we used diffractive events for the secondary collisions. The hadron remnant of the primary collisions is used as a scattering center for the still untouch other hadron which then scatters diffractivly. Even though some nucleons are part of multiple interactions in this simple approach the energy naturally conserved by the nature of diffractive events, i.e. we can generate diffractive events that only dissociate one hadron.

Considering also the `secondary-both` collisions, if we naively forget about energy conservation and just add all possible secondary events, this will lead to multiplicity 2 to 3 times higher than expacted from experimental data, see Figure 6.3, so this is clearly not the proper way to do it. What we instead would like to do is to seperate the remnents of the diffraction from the rest of the final-state particles. These renments are normally the particles that propagate continue to propaget in the beam tube after the collision and are therefor undetecable by experiments. In the case of a MinBias events this seperation would be easy as the remnent is a sperate particle. For a diffractive event on the other hand this is different, as in the diffractive process the hadron is split into a quark and a quark-antiquark pair. The differentiation of remnant and the measured final-state particles is not straight forward possible, at least not at the parton-level. Due to the nature of the program code, which is separated into a Python part and `C++` part, there is no easy way to overcome this obstacle.

In the second approach we tried to find a compromise to incorporate the `secondary-both`, but not blindly add too many interactions. We introduce $N_{\text{max inter}}(b)$ the maximal number of interactions which is allow depending on the impact parameter of the heavy ion collision. If we exceed this number we terminate the process of adding more secondary events. Interaction means in this context the number of dissociative interacting particles or processes, i.e. a MinBias event would count as two interactions because both participating hadrons scatter dissociatively and a diffractive event would count as one. This number must be between N_{primary} and the total number of all possible collisions, given by our interaction model. For central collisions we can set

$$N_{\text{max inter}}(0) = N_{\text{targ}} + N_{\text{proj}}. \tag{4.14}$$

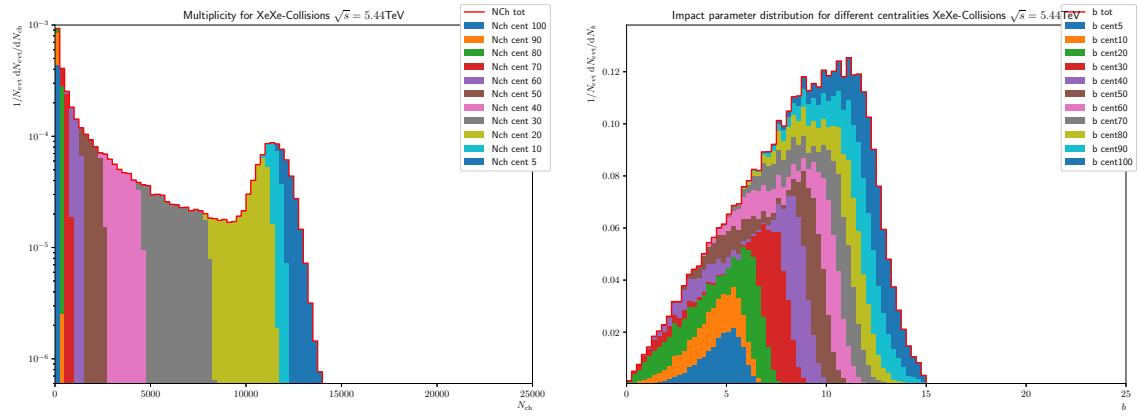
In general $N_{\text{max inter}}(b)$ should be proportional to the overlap function Eq. (4.2) defined by the optical limit, but for now we only fix it to $N_{\text{max inter}}(0)$. This will only lead to the proper effect for central collisions, which is enough for us right now.

Additionally we introduce a *secondary weight* for the secondary events w_{diff} (the subscriptions stems from the initinally name: diffractive weight). This basically changes how much we count a secondary event. So as decribed above one would normally count a secondary event as 1, but now we replace this with w_{diff}.

In Angantyr they proposed the use of diffractive events only with some minimal invarient mass. This allows the change of the signature of the diffractive event by making them harder. The idea was also implemented in this code, but not further evaluated.

4.2.5. Handling of Protons and Neutrons

Until now we did not take into account that a nucleus consists of neutrons and protons. The difference in the parton PDFs is small for neutrons and protons, so we do not expect the inclusion to drastically change the overall results. But we have to take care of proper charge conservation. Herwig allows for an easy modifcation of the beam particles, so we used this to incorporate the different nucleon types in the simulation. This leads to four different combinations for hard primary collisions, taking the initials of the hadrons we get “pp”, “pn”, “np” and “nn” events. Here the first letter repesents the hadron type from



(a) Multiplicity of charged particle separated into centrality classes.

(b) Impact parameter distribution separated into centrality classes.

Figure 4.7.: Plots are for 10000 XeXe events with $\sqrt{s_{NN}} = 5440\,\mathrm{GeV}$. There is a 'NCh tot' plot for the overall multiplicity. The centrality classes have the range down from the given value to the next range. So 'cent 20' equals the range 10 to 20.

the projectile, while the second letter represents the nucleon from the target. We would like to do the same for the secondary hard interactions which are simulated by diffraction. Unfortunately, the current version of Herwig only provides the possibility to perform diffraction on protons. The additional matrix elements for a diffractive neutron collision could be quite straightforward to implement, but this is left open for future work. So we only get "diff_left" and "diff_right" diffractive events. This will be a cause of breaking of charge conservation.

4.3. Centrality Classes

The analysis of heavy-ion collisions allows us to study QCD at high temperatures and huge pressures similar to the conditions shortly after the Big Bang [54]. This is especially the case for central heavy-ion collisions as in this case a lot of deep-inelastic hadron collisions take place in a small volume. When an experiment performs such collisions it cannot control the impact parameter of the collision, i.e. how central the collisions is. Most collisions will be peripheral due to geometric reasons. To perform studies on the date provided by the detector, we need to find a way to separate the peripheral from the central collisions. Therefore the concept of *centrality* was developed. We need to have an observable O that has a monotonic relationship with the impact parameter, then we can use O instead to classify different events. Prominent observables are the multiplicity of charged particles N_{Ch} (ALICE) or the transverse energy E_T (ATLAS). The normal procedure is to use the histogram for the observable and divide it in centrality classes i.e. precentile ranges corresponding to the area of the histogram. Each centrality class is then defined by a value range in the observable. For example the centrality class for the range from 10 % to 20 %, would be an area of size 10 % of the histogram, with 10 % of the events having a higher value for O and 80 % having a lower value.

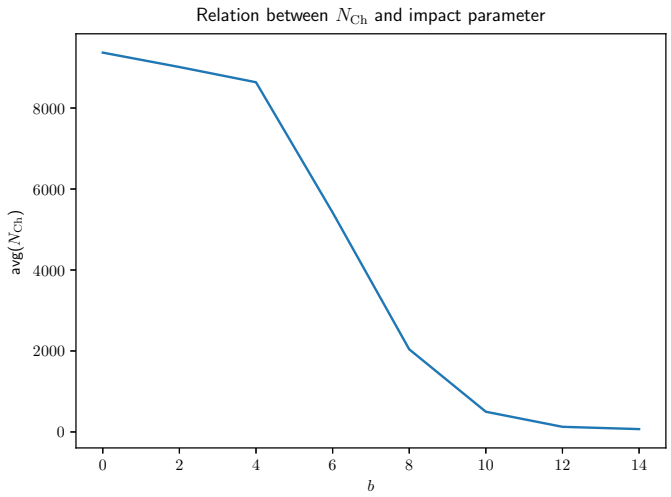


Figure 4.8.: Monotonic relationship between impact parameter and $\langle N_{\mathrm{Ch}} \rangle$.

In Figure 4.7a we see a stacked histogram of N_{Ch} for an ALICE XeXe-collision, where the fractions stemming from the different centrality classes are color labeled. We can observe the expacted behavior for centralitiy classes, which should be straight vertical cuts through the histogram. The slight deviation from absolute vertical cut can be traced back to the fact that the centrality classes are defined not on an overall charged particle histogram, but on the signal from the V0M [55] detector in case of ALICE.

We plot the impact parameter distribution in Figure 4.7b, again color labeling the centrality classes. The correlation between impact parameter and centrality class is apparent, but we should mention that for the collisions with the lowest centrality value we do not get the ones with the smallest impact parameter.

Overall, the momotonic relationship between the impact parameter and average charged particle multiplicity $\langle N_{\mathrm{Ch}} \rangle$, which can been seen in Figure 4.8, exists. So we can be reassured that this approach of defining the centrality classes is reasonable for our model.

5. SimpleHeavyIon Code

In this chapter the Python program code SimpleHeavyIon[1] which was written to perform the Glauber calculation and orchestrate Herwig in the background is described in more detail. Python is marked by its clear and beginner friendly syntax and high workflow speed combined with a multitude of good libraries, like the one for HepMC3 [56], which we heavily use as a bridge between our code and Herwig. The package is called `simple_hi` and it can be imported by

```
from simple_hi import <some module>
```

Other modifications made to ThePEG, Herwig and Rivet [57] are described too. And finally some examples on how to run the code and perform analysis are presented.

5.1. Workflow

The basic code flow of SimpleHeavyIon is shown in Figure 5.1. The `simple_hi_script.py` is the entry point to every run. To start the *hard process* (see section 5.2 and Figure 5.2) it calls the `hard_and_merging` function which initializes an output writer, the parton-level generator (PLG) and the Glauber calculation class `Collision`. The PLG runs Herwig in the background to simulate the primary and secondary events till parton-level and consumes them back via multiple fifo-pipes. Using the parton-level events from the PLG the `Collision` object merges them into one heavy-ion event according to the procedure for finding the primary and secondary interactions defined in subsection 4.2.4. In the next step this single event is written via the output writer to another fifo-pipe which will be the starting point for the *soft process*. Here another instance of Herwig uses the input interface(see subsection 5.3.1) to read-in the merged parton-level event, and perform soft physics like color reconnection and hadronization on it, leading to the detectable final-state. This final-state is then again piped through a fifo to Rivet which performs analysis and produces YODA-histogram files ready to be plotted with to provided tools from Rivet.

5.2. Hard process

The hard process represents a heavy-ion collision up to the parton-level. It is initiated through the `hard_and_merging` function in the `simple_hi_script.py`. The sequence diagram Figure 5.2 explains the different interacting parts, which can be found in yellow boxes on top of the figure and inside the *loop* block. To read this diagram, one can of the

[1] `https://git.particle.kit.edu/julianl/simpleheavyion/`

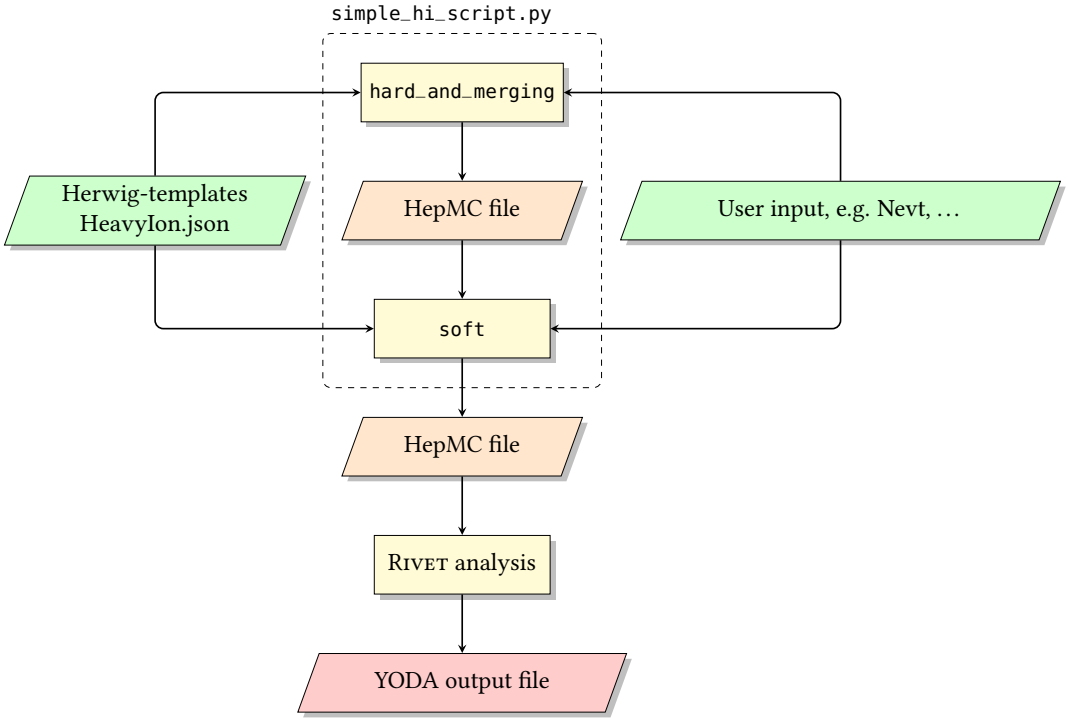


Figure 5.1.: Program flow of `simple_hi_script.py`. Green trapeze shaped nodes are input files and user data, yellow rectangles are different program steps, orange trapezes are fifo-pipes and the red trapeze is the output file. Static data of some heavy-ions is predefined in the `HeavyIon.json` file. The `Herwig templates` will be modified with data from the `user input`.

time-arrow pointing downwards. Each part of the code has a line going downwards, and always when the part is active there is rectangle. Horizontal arrows represent either function calls or information/data flow, where the direction is indicated by the tip of the arrows. The `hard_and_merging` function, the parton-level generator and the `Collision` class are Python parts, the HepMC-writer and the HepMC-pipes provide input-output orientated functionality and Herwig provides the NN-events. In the *init* block the `hard_and_merging` function initializes the other parts, most importantly 4 to 6 PLGs for the different NN-interaction types, which in turn each create one HepMC-pipe and start one instance of Herwig with the appropriate settings. These instances will continually push events to the HepMC-pipe if needed. Herwig is controlled via `in`-files, which are created on the go by using self-written Herwig-templates.

The main part of the hard process is the *loop* going over the range of the number of desired heavy-ion events. In each loop a new `Collision` object is created, which uses the persistent HepMC-writer and the PLGs. In case of the HepMC-writer it is a necessity, but in case of the PLGs it is performance consideration, rooting in the long initialization time of Herwig. The `Collision` object then performs all the basic Glauber calculations inside the `init_collisions` method. Based on these calculations the `get_heavy_ion_collision` method produces the heavy-ion event, by calling the `get_events(N)` method of the PLGs and merging the returned HepMC-events into a single event. The `merge_events` method

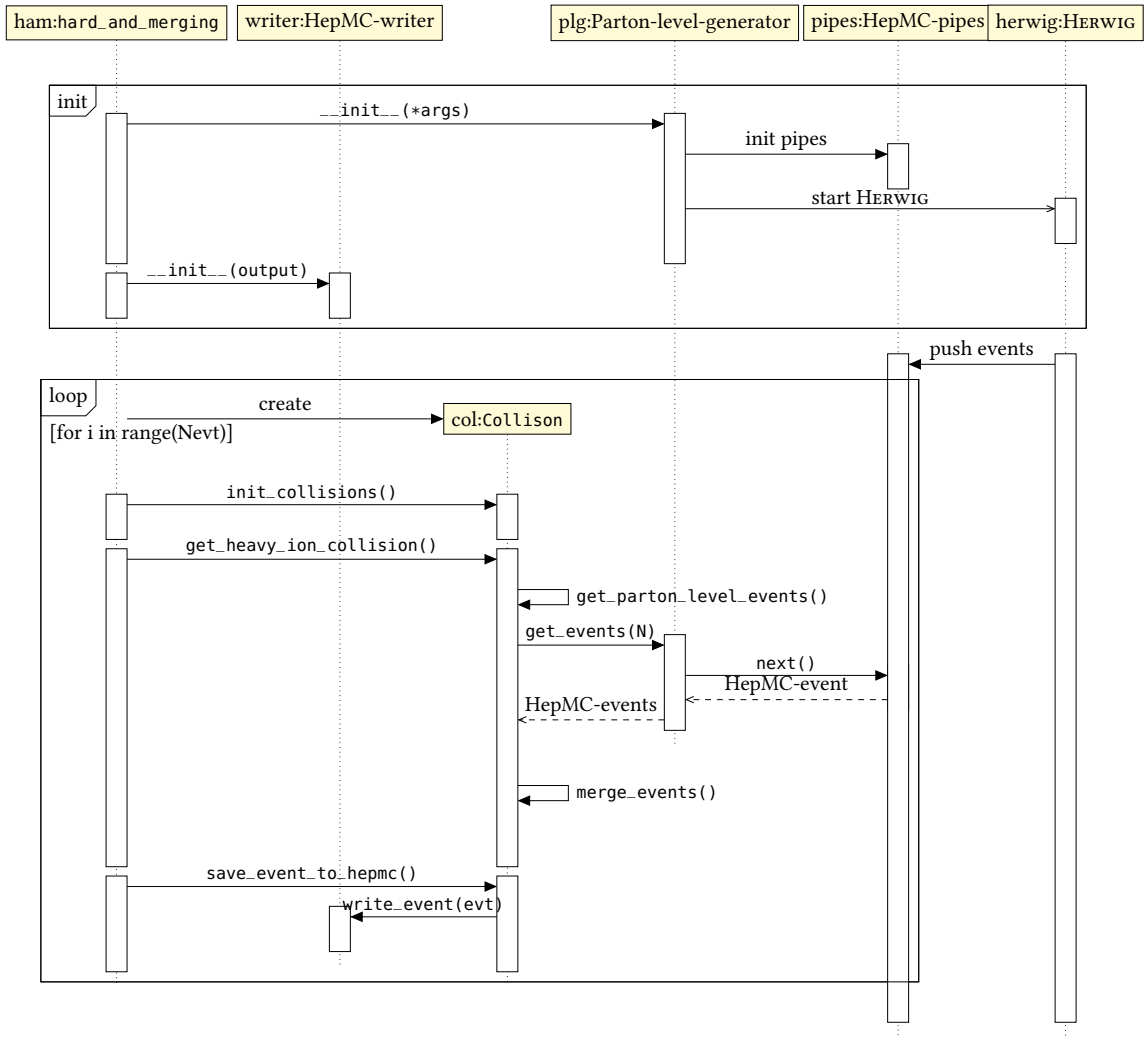


Figure 5.2.: Sequence diagram of `hard_and_merging` function inside `simple_hi_script.py`. It is equal to the "Hard processes" node in Figure 5.1.

shifts all the NN-interactions to their desired position and combines them into a single event. While doing so it keeps track of the flow variables and in case of diffractive events removes the spectator proton from the NN-event. After this step the heavy-ion event is combined at the parton-level and must be written to a HepMC file/pipe via the HepMC-writer.

5.2.1. `Collision` Class

The `Collision` class is the main object representing a single heavy-ion collision. In the following the main methods of the class are described.

- `__init__` : function

  Sets all variables and the settings like beam energy and NN-interaction method. It creates the positions and types of nucleons in the two nuclei.

- `add_spectators` : function

  Add the spectator nucleons the heavy-ion event

- `create_birth_collision` : function

  Initialize the heavy-ion beam particles and create the birth vertex

- `create_nuclei` : function

  Call the `nucleus.create` and `sample_b` functions and shift nucleon positions by $b/2$

- `get_diff_event` : function

  Return a diffractive event and the sign of the z-direction of the movement of the dissociating particle

- `get_heavy_ion_collision` : function

  Calls the `get_parton_level_events`, `set_heavy_ion_info`, `create_birth_collision` and `merge_events` functions, and returns the heavy-ion event

- `get_parton_level_events` : function

  Get N_{sample} events from the PLGs and sort them according to hardness

- `hardsort` : static function

  Sort a list of NN-events according to `hardness.pt_hardness`

- `init_collisions` : function

  Calculate colliding nucleon pairs using the provided NN-interaction model (dark-disc model). Create new nuclei and sample b until at least one colliding pair is found.

- `merge_events` : function

  Merges the NN-events from the PLGs and keeps track of the flow from the different events. We heavily rely on the HepMC3 library in this step to build up the event structure, and then use the `pyHepMC3.pyHepMC3.HepMC3.GenEvent.add_tree` function to add the all particles to the heavy-ion event. While merging the primary events all particles are shifted to

  $$b_{ij} = \frac{b_{\text{targ},j} + b_{\text{proj},i}}{2}, \tag{5.1}$$

  with $b_{\text{proj},i}$ and $b_{\text{targ},j}$ the position of the colliding nucleons in target and projectile.

  - `diff_merge` : function

    An extra function to handle the secondary hard interactions, which are modeled by diffractive collisions. The spectator proton is removed and replaced with a dummy particle, which has the PID of a reggeon as this does not conflict with any analysis. The dummy particle carries also the momentum transfer of the diffraction.

Table 5.1.: Runtime of Herwig on local machine for different number of events. All runs are NN-events only performed till parton-level.

Number of events	Time in s	Time per event in s
10	9	0.9
100	10	0.1
1000	15	0.015
2000	22	0.011
5000	42	0.0084
10 000	73	0.0073
20 000	132	0.0066

- `sample_b` : function

  Sample the impact parameter according to $\mathrm{d}\sigma \propto 2\pi b\,\mathrm{d}b$ subsection 4.2.2. The parameter space of b is restricted by $R_{\max}$

- `save_event_to_hepmc` : function

  Save the heavy-ion parton-level event to the HepMC-file or pipe

- `set_heavy_ion_info` : function

  HepMC allows the storing of additional heavy-ion specific information, like the number of collisions, participating hadrons or spectator hadrons

- `set_writer` : function

  Helper function to set the HepMC-writer, with is persistent for multiple `Collision` objects

5.2.2. Parton-level generator (PLG)

The parton-level generator builds the bridge between Herwig, creating NN-events, and a `Collision` object, consuming NN-events. The idea for creating an extra level of abstraction originates from performance considerations. In a normal collision only $O(100)$ NN-events are needed, which could be produced separately for each `Collision` object. Taking into account Table 5.1 it is clear that we can save tremendous amount of computational time, if we let Herwig create a higher number of events. For this to work we need persistent objects that are passed to each `Collision` object, which will be the PLGs. Additionally, the abstraction allows for a clean handling of all the processes that surround the execution of Herwig.

Each PLG only generates one type of NN-interactions, so for a normal heavy-ion collision that would be six different PLGs. That are four PLGs of type `prim_plg` each handling one type of primary hard collision, for the different combinatorics of neutron and proton collisions and two PLGs of type `diff_plg` for the hard secondary collisions for left and right diffractive events. This six PLGs are stored in a `dict()` and can be accessed via the PID off the process as keyword. At low level PLGs will create a fifo-pipe, then

initialize input files for Herwig and start Herwig. Using a Python generator function to call `pyHepMC3.pyHepMC3.HepMC3.ReaderAscii` we can read events from the fifo-pipe. Such a generator function makes it convenient to get an arbitrary number of NN-event, without the need for a cache. Additionally, it allows for a straight forward way to handle exceptions in case Herwig terminates because it has generated the number of events it was told to.

The diffractive collisions are normally part of MinBias runs, with the use of the `MEDiffraction::correctionweight()` function it is normally guaranteed that the correct diffractive cross section is reproduced. In our case we want to solely create diffractive collisions for the PLG. While it should be in fact be possible to set the cross section to a desired value, there is no interface to do this. Instead of creating a new interface to set the diffractive cross section manually, we added an interface to disable the `MEDiffraction::correctionweight()` function. With `Yes` or `No` we can set the behavior.

```
set MEDiffractionLeft:CorrectionWeight No
```

We also added an interface to manually set the minimum diffractive mass M^2_{min}. If you choose a value that is lower than the kinematic limit, it will resort to

$$M^2_{\text{min}} = \sqrt{m_p + m_q + m_{qq}}, \tag{5.2}$$

where m_p is the proton mass, m_q the constituent quark mass and m_{qq} the constituent diquark mass. In the code we use 0.0, because we did not further study the effect of M^2_{min}

```
set MEDiffractionLeft:MinDiffMass 0.0
```

5.2.3. Other functions and modules of the `simple_hi` package

The following classes and functions are all part of the `simple_hi` package.

- `collision.flow_attribute` : class

  Class to handle the flow variable coming from the multiple NN-events used to build the heavy-ion event. There would be a lot of identical flow variables if we did not take care of flow variables coming from different parton-level events should unique.

- `collision.nn_flow_ids` : class

  Class that splits the list of possible collision pairs into primary and secondary_side interactions(see subsection 4.2.4).

- `parton_level_generator` : module

  Contains the PLG classes and auxiliary functions

- `utlis.NucDataFrame` : class

  A pandas data frame [58] extended with a position function, which allows direct access to the spacial data

- `nn_interaction.nn_interaction` : class

  Base class for the NN-interaction model subsection 4.2.3. Currently, only the black disc model is implemented. If a new model should be needed the new class can inherit for `nn_interaction` and overwrite the `get_collisions` function.

- `hardness.pt_hardness` : function

  Hardness function based on the transversal momentum

- `nucleus.create` : function

  Simulates the nucleons inside the nucleus and returns the positions in form of a `NucDataFrame`

- `utils.Pipe` : class

  A helper class for creating and managing of fifo-pipes. Similar to `open(filename)` it can be initiated with the `with` statement

  ```
  with Pipe(pipe_name, ptype) as pipe:
    the_name = pipe.get_name()
  ```

  which will create the fifo-pipe and delete it when leaving the `with` statement.

5.3. Soft process

After the hard process, the parton-level heavy-ion event is written to an output file and ready for the application of soft physics handled in the soft process. The soft physics include the hadronization, the cluster formation, color reconnection and the decay step, which all are done by Herwig. Herwig needs to read-in the parton-level heavy-ion event and then starts its soft machinery. The soft process is handled by the `soft` function in the `simple_hi_scirpt.py` script.

5.3.1. Input interface

There are two possible interfaces to transfer the parton-level event from the Python module to Herwig namely the Les Houches accord on event files LHEF [59] and the HepMC3 event record library [56]. The package which should be used has to fulfill a few requirements. There needed to be an interface to C++ and to Python and the data format needed to store the relevant data of the particles in the event that are among others the 4-momentum, the position vector and the color-flow. Both LHEF and HepMC3 have a C++ and a Python interface, but only for LHEF a read-in class is already implemented in ThePEG code. By not providing a way to store the position of particles though LHEF is eliminated as possible interface. Which leaves us with the HepMC3 library which has a lot of functionally already implemented and a good Python support, but we need to implement a new handler to read-in events to Herwig.

There is no good documentation for this part of the ThePEG code, so the only thing we have is the LHEF module as a blueprint. We have to build the event in ThePEG, by using

the HepMC3 library. ThePEG events are in general meant for hadron-hadron collisions, which means there is a lot of functionality which needs to be initialized but is not needed for the later task of recreating the event from the HepMC3file.

The result of my here is the implementation of two classes in ThePEG, namely `HepMCReader` and `HepMCEventHandler`. Using the HepMC3 library the `hepmc_infile` is read-in and translated to a ThePEG event. Special care had to be taken by the handling of the flow variables.

In section A.1 you can find an extract of a Herwig in-file with the settings needed to initiate the two classes.

5.4. Heavy-ion Rivet-Analysis

As described in section 4.3 the concept of centrality is essential for the analysis of heavy-ion events. The centrality classes are not known *a priori* as they depend on the beam energy and the type of beam. Thus, we need to calibrate the centrality classes beforehand. For the most common beam types there already exist Rivet analyses for the calibration. By running them on a data set such an analysis creates a YODA file which functions as basis for the centrality classes in the actual analysis. As an example the process for an ALICE xenon analysis is presented (see [60] for further explanation), starting with the initial calibration run

```
userXY@itppc:~> rivet -a ALICE_XEXECentrality -o centrality.yoda data.hepmc3
```

When running the actual analysis after this we have to specify how the centrality classes are determined. By preloading the `centrality.yoda` file and setting the option `:cent=GEN` we tell the analysis to use the provided YODA file.

```
userXY@itppc:~> rivet -p centrality.yoda -a ALICE_2018_I1672756:cent=GEN -a
↪  ALICE_2019_I1723697:cent=GEN:beam=XeXe -o out.yoda data.hepmc3
```

To get statistical meaningful results it is normal to make MC event generator run that produce of $O(10 \times 10^5)$ or more heavy-ion events, which if stored in a HepMC3 file would take up $O(1\,\mathrm{TB})$ of disk space. Therefore, it is common to use a fifo-pipe between the MC event generator and Rivet. While saving disk space we lose all the data that is not processed through Rivet and stored in the YODA file. For the calibration run this means we have to perform two separate MC event generator runs. One will only produce the `centrality.yoda` file and the second will perform the actual analysis.

To study the pseudorapidity distribution for xenon-xenon collision data from ALICE [61] a new unpublished Rivet analysis `ALICE_2018_I1672756` was created. The code files can be found in the SimpleHeavyIon package file tree.

5.5. How to use the code

The `simple_hi_scrirpt.py` is the entry point to a run with SimpleHeavyIon. Below you can find the possible settings it provides.

```
userXY@itppc:~/SimpleHeavyIon/scripts> ./simple_hi_script.py --help
usage: simple_hi_script.py [-h] [--hepmc_outfile HEPMC_OUTFILE] [-Nevt NEVT]
                           [-N N] [-Z Z] [-b B] [-E BEAM_ENERGY]
                           [--sigma SIGMA]
                           [--diffractive_weight DIFFRACTIVE_WEIGHT]
                           [--single] [--no-sample_b] [--cr CR]
                           [--p_rec P_REC] [--p_rec_b P_REC_B] [--no-hardness]
                           [--minM2 MINM2] [--plothist] [--fig FIG]
                           [--no_soft_pipe] [--only_merge]

optional arguments:
  -h, --help            show this help message and exit
  --hepmc_outfile HEPMC_OUTFILE
                        Output hepmc-file
  -Nevt NEVT            number of generated events
  -N N                  Atomic number
  -Z Z                  Atomic charge
  -b B                  Impact parameter in fm
  -E BEAM_ENERGY, --beam_energy BEAM_ENERGY
                        CM beam energy in GeV
  --sigma SIGMA         inelastic crosssection
  --diffractive_weight DIFFRACTIVE_WEIGHT
                        how much should a diffractive event count
  --single              use only single interactions
  --no-sample_b         Disable impact parameter sampling
  --p_rec P_REC         reconncetion probability
  --p_rec_b P_REC_B     reconncetion probability
  --no-hardness         disable hardness sorting
  --minM2 MINM2         minimal diffractive mass
  --plothist
  --fig FIG             Histogram plot file name
  --no_soft_pipe        create a real intermediate hepmc file, not a pipe. for
                        debug purpose
  --only_merge          only merge part, no secondary herwig run
```

The most important ones would be `-Nevt -N -Z -E --hepmc_outfile` and `--sigma`. With them a minimal run would look like

```
userXY@itppc:~> ~/SimpleHeavyIon/scripts/simple_hi_script.py -Nevt 1000 -N
↪  129 -Z 54 -E 54440 --sigma 6.8 --hepmc_outfile XeXe.hepmc
```

for a XeXe-collision with $\sqrt{s_{NN}} = 5.44\,\text{TeV}$.

Here some further explanations on some settings that might not be obvious.

- `--single` disables all secondary interactions
- `--no-sample_b` disables the sampling of the impact parameters. It allows to set the impact parameter to a fixed value with `-b`

Table 5.2.: Runtime of different type of heavy-ion collisions. One can see the correlation between the atomic number A and the runtime on the one hand and also the correlation between the center-of-mass energy $\sqrt{s_{NN}}$ and the runtime.

beam-particle	A	$\sqrt{s_{NN}}$ in GeV	t_{run}/(10000 Nevt) in h
pp(on workstation)	1	5000	1/60
XeXe	129	5440	15
AuAu	197	200	4
PbPb	208	5020	43

- `--no_soft_pipe` instead of a fifo-pipe between the hard and the soft process, this will create a normal file. This is mainly useful for debugging purposes.
- `--only_merge` disables the soft process. We used this in combination with modified input files for the PLG to create heavy-ion collisions out of final-state NN-event.

5.6. Performance Considerations

In computational physics runtime is often one of the biggest problems. Without the limits of computers, already as of today a lot of still open problems could be solved. We often have the theoretical solution, but it is computationally too expensive to execute it. Even with the best computers from today it would just take too much time, making it impossible.

Being of computational nature, we were also faced with computational obstacles in this thesis. In part these could be solved, like with the PLGs, but some bottlenecks still remain. Throughout the process of writing the program code we always kept the performance in mind and tried to avoid performance expensive constructions. We can say that we achieved this to a satisfactorily point, because the heavy-ion Rivet analysis seems to be the slowest part in the current stack. This might be due to fact that the ALICE Rivet analysis searches through the whole particle history stored in the HepMC-file, instead to just considering the final-state particles.

Taking a look at the Table 5.2, we see that with the current cluster available at the ITP department at the KIT a run for lead-lead collisions with 10 000 events and more will take up to multiple days. This leads to the advice that if one wants to create more than 10 000 heavy-ion events, one should consider splitting up the calculations, combining them later again with the `yodamerge` function. This should allow getting up to 1 000 000 events in an acceptable time, if we start 100 jobs parallel on the cluster.

6. Results

The focus of this work was the initial implementation of an MC event generator for heavy-ion collisions based on the Glauber model. As such no sophisticated tunes where performed, but we made rather some educated guessing. Keeping this in mind, the parameter set Table 6.1 used for the runs plotted in the following should be seen as preliminary. Also, we used fixed values for the NN-cross-section, which can be found in Table 6.2 [65].

In general the p_T spectra are model fairly well. We used an analysis of the RHIC experiment BRAHMS [62] to produce to plots in Figure 6.1. For the π^+ plot on the left side we see that the general trend is captured, however there exists a shift towards low p_T. It should be possible to improve this with tuning of the color reconnection parameters, as CR enhances high p_T clusters. One feature that is different in the p_T plot of K^+ on the right side, is the dip for low p_T, which can not be captured by the model. This is a known problem of Herwig also for pp-collisions, so we do not expect that it will disappear in our model.

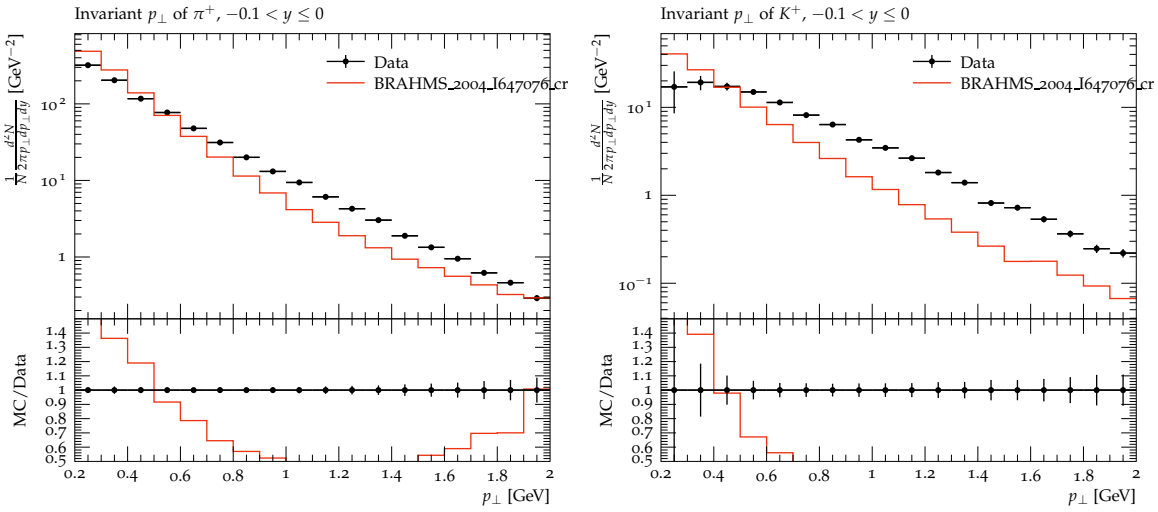
The most striking result is the *impact of color reconnection*. We assume that the more complex color-structure of a heavy-ion event allows the color reconnection model to have a greater impact on the event structure compared with pp-collisions. Therefore, we need to distinguish the difference between color reconnection solely from the NN-event and the additional effects stemming from the heavy-ion nature. To do so, we present to different two different plots in Figure 6.2 labeled `mwh_cr_on` and `mwh_cr_single`. `mwh_cr_on` is our normal set up, in which NN-events are combined at parton level to one heavy-ion event and then the soft physics, CR being one of them here, applied on top. In contrast to this `mwh_cr_single` combines the NN-events only at the final-state after the whole MC event generator machinery took place. This will cause CR only to work on the individual NN-events. The third plot `mwh_cr_off` is without any CR. We can state that CR on the whole heavy-ion event compared to CR only on the NN-events reduces the multiplicity, which we expect. This goes hand in hand with an increase of the average momentum, which can be seen in the dummy analysis Figure A.1. In the appendix you can find two

Table 6.1.: Parameterset used for the `simple_hi` runs if not mentioned otherwise.

parameter	value
diffractive weight	0.5
$N_{\text{max inter}}$	$N_{\text{targ}} + N_{\text{proj}}$
p_rec	0.95
p_rec_b	0.7
minM2	0.0

Table 6.2.: Inelastic cross section for different $\sqrt{s_{NN}}$ energies.

Energy in GeV	σ in mb
200	41.6(6)
5020	67.6(6)
5440	68.4(5)



(a) of π^+ **(b)** of K^+

Figure 6.1.: BRAHMS p_T spectrum for Au-Au collisions at $\sqrt{s_{NN}} = 200$ GeV [62].

additional plots Figure A.2 and Figure A.3 for not so central collisions, which deviate more from the data.

We can see that in the rapidity distribution the mid-rapidity region has a deficit of charged particles while for larger absolute rapidity we have an excess in multiplicity. This might be explained by the fact that we use normal diffractive events for the secondary hard interactions. In fact these interactions should behave like normal MinBias events, but only acting on a single hadron. ANGANTYR achieves this with a modified diffraction model. This might be a possible starting point for further research.

In the desire to somehow mimic energy conservation we introduced the *secondary weight* w_{diff} and the *maximum number of interactions* $N_{\text{max inter}}(b)$. We tested three settings, the $w_{\text{diff}} = 0.0$ which means we do not restrict secondary interactions at all, then $w_{\text{diff}} = 0.5$ which restricts them a little, but less than primary hard interactions and finally $w_{\text{diff}} = 1.0$ which means that all interactions are counted the same way. In the plot Figure 6.3 we can see that with our fixed value for $N_{\text{max inter}}$ the setting of $w_{\text{diff}} = 0.5$ is the best for the super central collisions.

Another effect of diffractive weight and the fixed $N_{\text{max inter}}(b)$ can be seen in the distribution of charged particles in the V0M-detector [55] of ALICE Figure 6.4. We plotted runs with the different values of the secondary weight as above. With no restriction on secondary events we get a distribution that monotonically drops with increasing N_{Ch}. In the case of a restriction on the number of interactions, we get a bump for high values

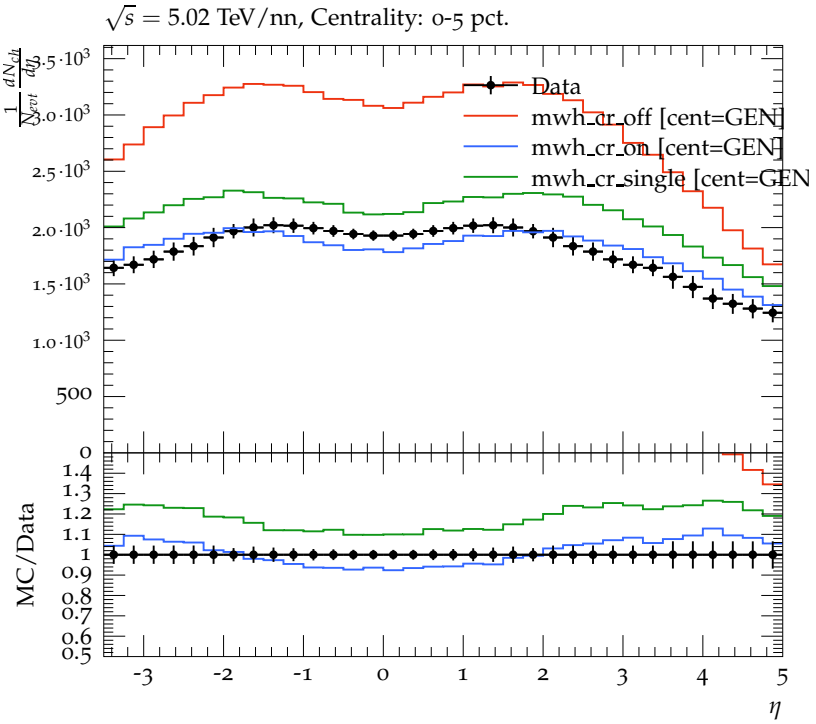


Figure 6.2.: Color reconnection effect on the pseudorapidity distrbution for a PbPb-collision at $\sqrt{s_{NN}}$ = 5.02 TeV with ALICE data [63]. 'mwh_cr_off' is multiple NN-interactions with hardness sorting and without color reconnection. 'mwh_cr_on' is multiple NN-interactions with hardness sorting and color reconnection turned on. 'mwh_cr_single' is multiple NN-interactions with hardness sorting and color reconnection on, but only acting on the individual NN-interactions.

x-values. This can be explained by the rude fixing of $N_{\text{max inter}}(b)$, which is only ideal for central collisions. By dividing the overall charged particle distribution Figure 6.5 in the different histograms for different impact parameter ranges, we can see how well our classification in centrality classes matches with the impact parameter. There is clearly a correlation as already seen in Figure 4.8, but also an unexpected behavior marked by two separate distributions for the same impact parameter with a gap in between. This effect is not understood right now, might also stem from the fixed $N_{\text{max inter}}$ but could also have other origins.

In the two plots for the *elliptic flow* Figure 6.6 we can see that this collective behavior is not captured by our model. This is not totally surprising, as the elliptic flow stems from the collective behavior of the nucleons in the nucleus. Besides CR, we do not have any collective process in our model. It is advisable to focus on this subject in future works.

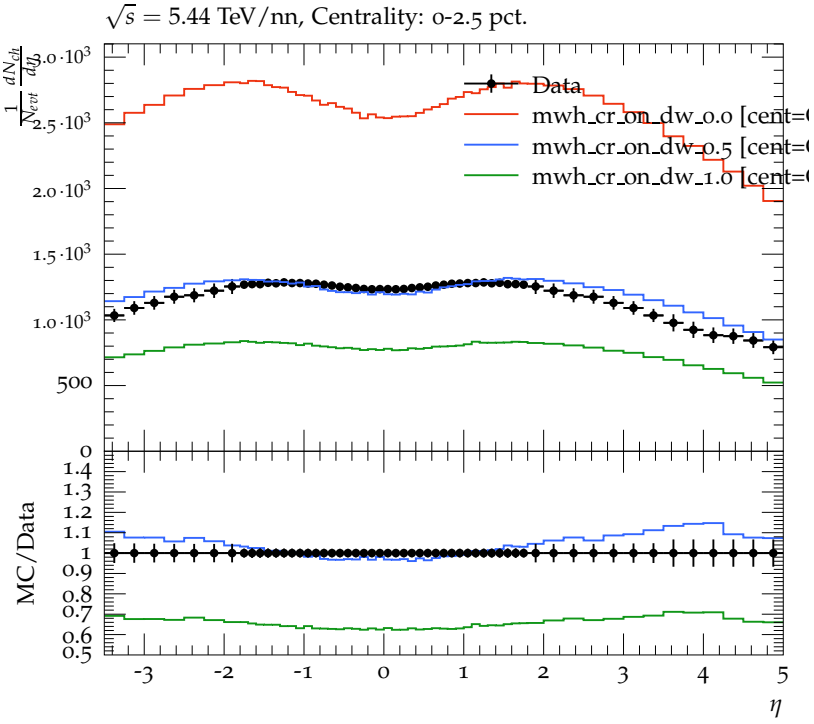


Figure 6.3.: Effect of different secondary weight on a XeXe-collision at $\sqrt{s_{NN}}$ = 5.44 TeV with ALICE data [61].

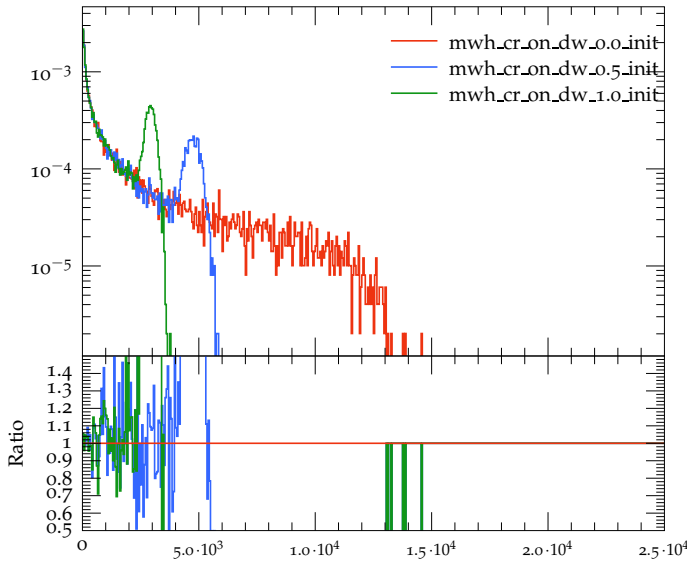


Figure 6.4.: The mulitplicty measured with the `V0M` analysis from ALICE. This is the basis of the centrality calibration. The result for different settings for the secondary weight are plotted together. One can see that if the diffractive weight is non-zero a bump appears, caused by the primitve constraint on the number of secondary interactions. The units here are somewhat arbritary, but the x-axis correlates with the N_{Ch}.

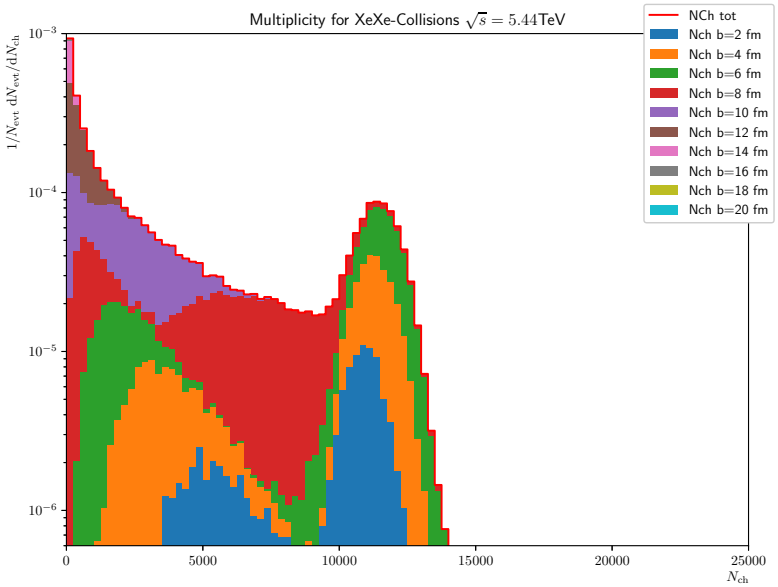
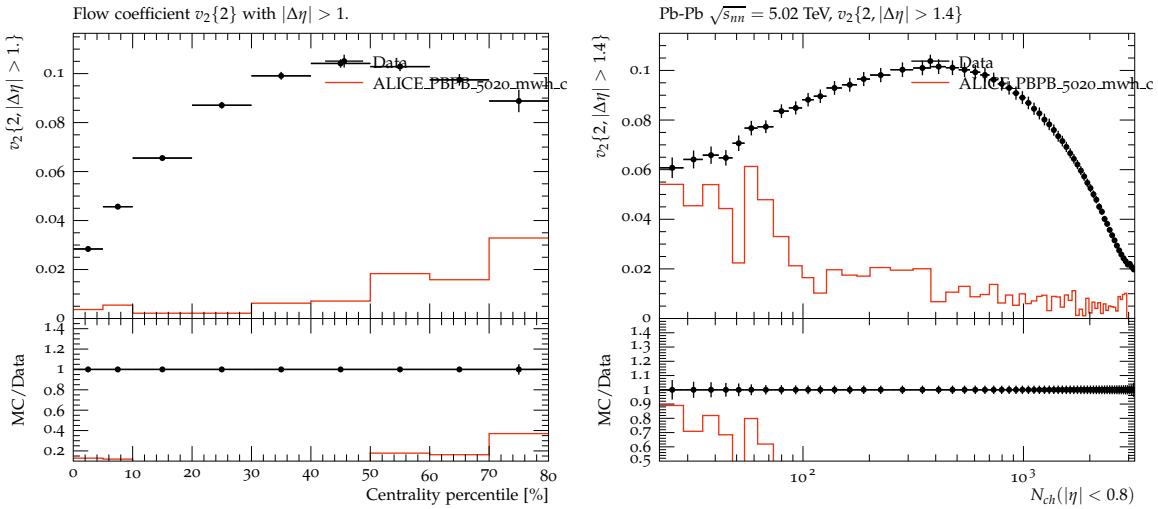


Figure 6.5.: The seperated plots have a 2 fm wide impact parameter range. The impact parameter range is from the given value down 2 fm, i.e. the value $b = 6$ fm stands for the range from 4 fm to 6 fm.



(a) Elliptic flow for different centrality classes.

(b) Elliptic flow in dependacy to number of charged particles for 5 % most central events.

Figure 6.6.: Our model is not able to capture the elliptic flow as of right now. ALICE PbPb collision with $\sqrt{s_{NN}} = 5.44$ TeV [64]. Standard parameter settings for SimpleHeavyIon.

7. Summary and Outlook

In this thesis the first code SimpleHeavyIon for the simulation of heavy-ion collisions on the basis of the Glauber model that uses the MC event generator Herwig and its color reconnection model was developed. The Glauber model uses a black-disc interaction model and distinguishes between primary and secondary hard interactions. We used an intermediate approach to divide the program into a Python part and using Herwig in the background. The Python code performs the Glauber calculation and orchestrates Herwig.

The main idea is to split the event in a hard and soft stage. In the hard stage everything up to the parton-level is simulated, meaning all processes still occur in the perturbative regime and most importantly before color reconnection. These hard processes are generated with Herwig and then read in with the Python binding of the HepMC3 library. SimpleHeavyIon merges the events from Herwig into a single heavy-ion event. Via a HepMC3-file this heavy-ion event is transferred to another instance of Herwig, which will perform the remaining soft MC steps and produce a final-state event.

One obstacle we faced was the implementation of an input interface for HepMC3 files, which is essential in our workflow. This has been achieved by creating the `HepMCReader` and `HepMCEventHandler` classes in ThePEG.

Two new parameters, namely the maximal number of interactions $N_{\text{max inter}}(b)$ and the secondary weight w_{diff}, were introduced to mimic energy conservation in the context of secondary hard interactions.

This model was able to reproduce some basic observables like the pseudorapidity distribution for central event and the p_T spectrum for different particles to a satisfactory degree. Observables sensible to the collective behavior of heavy-ion collisions like the different flow coefficients could not be reproduced. One interesting question was how would the effect of CR differ for pp-collisions and for heavy-ion collisions. We were able to show that the more complex color structure of heavy-ion events enhances the color reconnection effects as expected.

Outlook

Even though the code is working and produces reasonable results in a restricted range it is clearly only in a preliminary stage towards a full model. This allows for a wide range of future improvements or research inspired by this work.

A possible first step would be to tune the barionic and mesonic color reconnection probabilities on heavy-ion data. The used values stem from tunes on pp-events, and it is possible that there are different better fitting parameters for heavy-ion events.

The secondary hard events are clearly available to further improvement. Energy-momentum conservation can be violated through them, and they are the cause for the

bump the multiplicity histograms. The introduction of an $N_{\text{max inter}}(b)$ with a proper dependency on the impact parameter would be one idea. In this context, one could also study if the introduction of two parameters to determine the number of secondary interactions leads to an additional degree of freedom, which is in fact not needed, i.e. can we remove w_{diff} altogether?

Also, concerning secondary hard events the minimal diffractive mass was shortly mentioned. Building on the already implemented interfaces to set it manually, one could study the effect more thoroughly.

The black-disc model we use has some limitations, because it lacks fluctuation. With the implementation of the gray-disc model it would be possible to overcome these limitations.

Finally, the usage of Python was initially justified by fast development and easy testing of new ideas. This is in fact true for the beginning when implementing the simulation of the nucleus and the Glauber calculation. But it might be the case one is limited in the further improvement of the code by Python as we can only communicate information about the event with Herwig via the usage of HepMC3. Future work might therefore think about implementing the Glauber model directly in C++.

Acknowledgments

First I want to thank PD Dr. Stefan Gieseke for the opportunity to work on such an interesting topic, which allowed me to try out my own ideas and dwell deep into creating a program package. I am grateful for the countless times we met and talked things over again, I was always welcome to ask anything.

I'd like to thank Prof. Dr. Dieter Zeppenfeld for agreeing to be the second reviewer of this thesis.

A big thank you to Cody Duncan for proof-reading and helping me in the final phase of the thesis. I want to thank Patrick Kirchgäßer for his support in case I had problems with understanding Herwig. The whole institute ITP needs to be thanked for its warm and welcoming atmosphere. I enjoyed the time together at lunch and the fancy Friday coffee. For the IT-support in case some program did not work as intended or any other obscure error I encountered while writing the code I thank Martin Gabelmann.

I want to thank my family for their support throughout my time as a student. To know that you always have someone you can rely on in any situation gives you great support in case something might go sideways.

Finally, I want to thank all my friends in Karlsruhe and elsewhere. For their open ear when I needed one and all the hours we spent together, which in the end made this part of my life what it was.

Bibliography

[1] R. K. Ellis, W. J. Stirling, and B. R. Webber. *QCD and Collider Physics*. Cambridge University Press, Oct. 1996. DOI: 10.1017/cbo9780511628788.

[2] P.A. Zyla et al. "Review of Particle Physics". In: *PTEP* 2020.8 (2020), p. 083C01. DOI: 10.1093/ptep/ptaa104.

[3] Gerard 't Hooft. "A planar diagram theory for strong interactions". In: *The Large N Expansion In Quantum Field Theory And Statistical Physics: From Spin Systems to 2-Dimensional Gravity*. World Scientific, 1993, pp. 80–92. DOI: 10.1142/9789814365802_0007.

[4] Edward Witten. "Baryons in the 1N expansion". en. In: *Nuclear Physics B* 160.1 (Nov. 1979), pp. 57–115. DOI: 10.1016/0550-3213(79)90232-3.

[5] Simon Platzer and Malin Sjodahl. "Subleading Nc improved Parton Showers". In: (Dec. 31, 2011). DOI: 10.1007/JHEP07(2012)042. arXiv: 1201.0260v1 [hep-ph].

[6] A. Hosoya and K. Kajantie. "Transport coefficients of QCD matter". In: *Nuclear Physics B* 250.1-4 (Jan. 1985), pp. 666–688. DOI: 10.1016/0550-3213(85)90499-7.

[7] Peter Brockway Arnold, Guy D. Moore, and Laurence G. Yaffe. "Transport coefficients in high temperature gauge theories. 1. Leading log results". en. In: *JHEP* 11 (2000), p. 001. DOI: 10.1088/1126-6708/2000/11/001.

[8] Atsushi Nakamura and Sunao Sakai. "Transport Coefficients of a Gluon Plasma". en. In: *Physical Review Letters* 94.7 (Feb. 2005), p. 072305. DOI: 10.1103/PhysRevLett.94.072305.

[9] Harvey B. Meyer. "A Calculation of the shear viscosity in SU(3) gluodynamics". en. In: *Phys.Rev.D* 76 (2007). DOI: 10.1103/PhysRevD.76.101701.

[10] Charles Gale, Sangyong Jeon, and Bjoern Schenke. "Hydrodynamic Modeling of Heavy-Ion Collisions". In: *Int. J. Mod. Phys. A* 28 (2013), p. 1340011. DOI: 10.1142/S0217751X13400113. arXiv: 1301.5893 [nucl-th].

[11] Rudolf Baier et al. "Relativistic viscous hydrodynamics, conformal invariance, and holography". In: *JHEP* 04 (2008), p. 100. DOI: 10.1088/1126-6708/2008/04/100. arXiv: 0712.2451 [hep-th].

[12] W. Israel and J.M. Stewart. "Transient relativistic thermodynamics and kinetic theory". In: *Annals of Physics* 118.2 (1979), pp. 341–372. DOI: https://doi.org/10.1016/0003-4916(79)90130-1.

[13] P. Huovinen et al. "Radial and elliptic flow at RHIC: Further predictions". en. In: *Phys.Lett.B* 503 (2001), pp. 58–64. DOI: 10.1016/S0370-2693(01)00219-2.

[14] P. F Kolb et al. "Elliptic flow at SPS and RHIC: from kinetic transport to hydrodynamics". en. In: *Physics Letters B* 500.3 (Feb. 2001), pp. 232–240. DOI: 10.1016/S0370-2693(01)00079-X.

[15] "LHC Machine". In: *JINST* 3 (2008). Ed. by Lyndon Evans and Philip Bryant, S08001. DOI: 10.1088/1748-0221/3/08/S08001.

[16] John C. Collins, Davison E. Soper, and George Sterman. "Factorization for short distance hadron-hadron scattering". en. In: *Nuclear Physics B* 261 (Jan. 1985), pp. 104–142. DOI: 10.1016/0550-3213(85)90565-6.

[17] John C. Collins, Davison E. Soper, and George Sterman. "Factorization of hard processes in qcd". In: vol. Volume 5. Advanced Series on Directions in High Energy Physics Volume 5. WORLD SCIENTIFIC, July 1989, pp. 1–91. DOI: 10.1142/9789814503266_0001.

[18] Torbjörn Sjöstrand. "A model for initial state parton showers". In: *Physics Letters B* 157.4 (July 1985), pp. 321–325. DOI: 10.1016/0370-2693(85)90674-4.

[19] Andy Buckley et al. "General-purpose event generators for LHC physics". In: *Physics Reports* 504.5 (July 2011), pp. 145–233. DOI: 10.1016/j.physrep.2011.03.005. eprint: 1410.3012.

[20] Manuel Bähr et al. "Herwig++ Physics and Manual". In: *Eur.Phys.J.C58:639-707,2008* 58.4 (Mar. 6, 2008), pp. 639–707. DOI: 10.1140/epjc/s10052-008-0798-9. arXiv: 0803.0883v3 [hep-ph].

[21] Johannes Bellm et al. "Herwig 7.0 / Herwig++ 3.0 Release Note". In: *The European Physical Journal C* 76.4 (Apr. 2016). DOI: 10.1140/epjc/s10052-016-4018-8. arXiv: 1512.01178v1 [hep-ph].

[22] Leif Lönnblad. "ThePEG, Pythia7, herwig++ and Ariadne". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 559.1 (Apr. 2006). Proceedings of the X International Workshop on Advanced Computing and Analysis Techniques in Physics Research, pp. 246–248. DOI: 10.1016/j.nima.2005.11.143.

[23] Marc Bertini, Leif Lönnblad, and Torbjörn Sjöstrand. "PYTHIA version 7-0.0 - a proof-of-concept version". In: *Computer Physics Communications* 134 (June 2000), pp. 365–391. DOI: 10.1016/S0010-4655(00)00206-X.

[24] Andy Buckley et al. "LHAPDF6: parton density access in the LHC precision era". In: *The European Physical Journal C* 75.3 (Mar. 2014). DOI: 10.1140/epjc/s10052-015-3318-8. arXiv: 1412.7420v2 [hep-ph].

[25] T. Stelzer and W. F. Long. "Automatic Generation of Tree Level Helicity Amplitudes". In: *Comput.Phys.Commun. 81 (1994) 357-371* 81.3 (July 1994), pp. 357–371. DOI: 10.1016/0010-4655(94)90084-1. arXiv: hep-ph/9401258v1 [hep-ph].

[26] J. Baglio et al. *Release Note - VBFNLO 2.7.0*. Apr. 2014. arXiv: 1404.3940 [hep-ph].

[27] Geoffrey C. Fox and Stephen Wolfram. "A model for parton showers in QCD". en. In: *Nuclear Physics B* 168.2 (June 1980), pp. 285–295. DOI: 10.1016/0550-3213(80)90111-X.

[28] Vladimir Naumovich Gribov and L N Lipatov. "Deep inelastic ep scattering in perturbation theory". In: *Sov. J. Nucl. Phys.* 15.4 (1972), pp. 438–450.

[29] Yuri L Dokshitzer. "Calculation of the structure functions for deep inelastic scattering and e+e- annihilation by perturbation theory in quantum chromodynamics". In: *Zh. Eksp. Teor. Fiz* 73 (1977), p. 1216.

[30] Guido Altarelli and G. Parisi. "Asymptotic Freedom in Parton Language". In: *Nucl. Phys. B* 126 (1977), pp. 298–318. DOI: 10.1016/0550-3213(77)90384-4.

[31] Mats Bengtsson and Torbjörn Sjöstrand. "A comparative study of coherent and non-coherent parton shower evolution". en. In: *Nuclear Physics B* 289 (Jan. 1987), pp. 810–846. DOI: 10.1016/0550-3213(87)90407-X.

[32] G. Marchesini and B.R. Webber. "Monte Carlo simulation of general hard processes with coherent QCD radiation". In: *Nuclear Physics B* 310.3-4 (Dec. 1988), pp. 461–526. DOI: 10.1016/0550-3213(88)90089-2.

[33] B. R. Webber. "A QCD model for jet fragmentation including soft gluon interference". en. In: *Nuclear Physics B* 238.3 (June 1984), pp. 492–528. DOI: 10.1016/0550-3213(84)90333-X.

[34] Daniele Amati and Gabriele Veneziano. "Preconfinement as a property of perturbative QCD". In: *Physics Letters B* 83.CERN-TH-2620 (1979), pp. 87–92.

[35] Z. Koba, H.B. Nielsen, and P. Olesen. "Scaling of multiplicity distributions in high energy hadron collisions". In: *Nuclear Physics B* 40 (Jan. 1972), pp. 317–334. DOI: 10.1016/0550-3213(72)90551-2.

[36] Alexander Kupco. "Cluster Hadronization in HERWIG 5.9". In: *arXiv:hep-ph/9906412* (June 1999). arXiv: hep-ph/9906412.

[37] Torbjörn Sjöstrand and Maria van Zijl. "A multiple-interaction model for the event structure in hadron collisions". In: *Physical Review D* 36.7 (Oct. 1987), pp. 2019–2041. DOI: 10.1103/physrevd.36.2019.

[38] Torbjörn Sjöstrand. "Colour reconnection and its effects on precise measurements at the LHC". In: *arXiv:1310.8073 [hep-ph]* (Oct. 2013). arXiv: 1310.8073 version: 1.

[39] Stefan Gieseke, Christian Röhr, and Andrzej Siódmok. "Colour reconnections in Herwig++". In: *Eur.Phys.J. C72 (2012) 2225* (May 31, 2012). DOI: 10.1140/epjc/s10052-012-2225-5. arXiv: 1206.0041v2 [hep-ph].

[40] Stefan Gieseke, Patrick Kirchgaeßer, and Simon Plätzer. "Baryon production from cluster hadronization". In: *Eur.Phys.J. C78 (2018) no.2, 99* (Oct. 30, 2017). DOI: 10.1140/epjc/s10052-018-5585-7. arXiv: 1710.10906v3 [hep-ph].

[41] Stefan Gieseke, Frashër Loshaj, and Patrick Kirchgaeßer. "Soft and diffractive scattering with the cluster model in Herwig". In: *The European Physical Journal C* 77.3 (Mar. 2017). DOI: 10.1140/epjc/s10052-017-4727-7. arXiv: 1612.04701v1 [hep-ph].

[42] Vicenzo Barone and Enrico Predazzi. *High-energy particle diffraction*. Texts and monographs in physics. Berlin: Springer, 2002. DOI: 10.1007/978-3-662-04724-8.

[43] RJ Glauber et al. *Lectures in theoretical physics*. Interscience, New York London, 1959.

[44] R. J. Glauber. "Cross Sections in Deuterium at High Energies". In: *Phys. Rev.* 100.1 (1 Oct. 1955), pp. 242–248. DOI: 10.1103/PhysRev.100.242.

[45] A. Białas, M. Bleszyński, and W. Czyż. "Multiplicity distributions in nucleus-nucleus collisions at high energies". In: *Nuclear Physics B* 111.3 (Sept. 1976), pp. 461–476. DOI: 10.1016/0550-3213(76)90329-1.

[46] Michael L. Miller et al. "Glauber modeling in high energy nuclear collisions". In: *Ann. Rev. Nucl. Part. Sci.* 57.1 (Nov. 2007), pp. 205–243. DOI: 10.1146/annurev.nucl.57.090506.123020. arXiv: nucl-ex/0701025 [nucl-ex].

[47] V. N. Gribov. "Glauber corrections and the interaction between high-energy hadrons and nuclei". en. In: *Sov.Phys.JETP* 29 (1969), pp. 483–487.

[48] Wojciech Broniowski, Maciej Rybczynski, and Piotr Bozek. "GLISSANDO: GLauber Initial-State Simulation AND mOre". In: *Comput.Phys.Commun.180:69-83,2009* (Oct. 30, 2007). DOI: 10.1016/j.cpc.2008.07.016. arXiv: 0710.5731v3 [nucl-th].

[49] H. De Vries, C.W. De Jager, and C. De Vries. "Nuclear charge-density-distribution parameters from elastic electron scattering". In: *Atomic Data and Nuclear Data Tables* 36.3 (May 1987), pp. 495–536. DOI: 10.1016/0092-640x(87)90013-1.

[50] Maciej Rybczyński et al. "GLISSANDO 2: GLauber Initial-State Simulation AND mOre…, ver. 2". In: *Computer Physics Communications* 185.6 (June 2014), pp. 1759–1772. DOI: 10.1016/j.cpc.2014.02.016.

[51] Hong Pi. "An event generator for interactions between hadrons and nuclei — FRITIOF version 7.0". In: *Computer Physics Communications* 71.1-2 (Aug. 1992), pp. 173–192. DOI: 10.1016/0010-4655(92)90082-a.

[52] Christian Bierlich et al. "The Angantyr model for heavy-ion collisions in Pythia8". In: *Journal of High Energy Physics* 2018 (Oct. 2018), p. 134. DOI: 10.1007/jhep10(2018)134.

[53] Johannes Bellm et al. "PISTA: Posterior Ion STAcking". In: *arXiv e-prints* 1807 (July 2018). arXiv: 1807.01291 [hep-ph].

[54] D.J. Schwarz. "The first second of the Universe". In: *Annalen der Physik* 12.4 (June 2003), pp. 220–270. DOI: 10.1002/andp.200310010.

[55] The ALICE collaboration. "Performance of the ALICE VZERO system". en. In: *Journal of Instrumentation* 8.10 (Oct. 2013), P10016–P10016. DOI: 10.1088/1748-0221/8/10/P10016.

[56] Andy Buckley et al. "The HepMC3 event record library for Monte Carlo event generators". en. In: *Computer Physics Communications* 260 (Mar. 2021), p. 107310. DOI: 10.1016/j.cpc.2020.107310.

[57] C. Bierlich et al. "Robust Independent Validation of Experiment and Theory: Rivet version 3". In: *SciPost Phys. 8, 026 (2020)* (Dec. 11, 2019). DOI: 10.21468/SciPostPhys.8.2.026. arXiv: 1912.05451v3 [hep-ph].

[58] Jeff Reback et al. *pandas-dev/pandas: Pandas 1.1.5.* 2020. DOI: 10.5281/ZENODO.4309786.

[59] J. Huston. "The Les Houches accords: new tools for high energy physics". en. In: *Nuclear Physics B - Proceedings Supplements* 117 (Apr. 2003), pp. 268–271. DOI: 10.1016/S0920-5632(03)90542-1.

[60] Christian Bierlich et al. "Confronting Experimental Data with Heavy-Ion Models: Rivet for Heavy Ions". In: *The European Physical Journal C* 80.5 (Jan. 29, 2020). DOI: 10.1140/epjc/s10052-020-8033-4. arXiv: 2001.10737v1 [hep-ph].

[61] S. Acharya et al. "Centrality and pseudorapidity dependence of the charged-particle multiplicity density in Xe–Xe collisions at sNN=5.44TeV". en. In: *Physics Letters B* 790 (Mar. 2019), pp. 35–48. DOI: 10.1016/j.physletb.2018.12.048.

[62] I. G. Bearden et al. "Charged meson rapidity distributions in central Au+Au collisions at s(NN)**(1/2) = 200-GeV". en. In: *Phys.Rev.Lett.* 94 (2005). DOI: 10.1103/PhysRevLett.94.162301.

[63] J. Adam et al. "Centrality dependence of the pseudorapidity density distribution for charged particles in Pb–Pb collisions at sNN=5.02 TeV". en. In: *Physics Letters B* 772 (Sept. 2017), pp. 567–577. DOI: 10.1016/j.physletb.2017.07.017.

[64] J. Adam et al. "Anisotropic Flow of Charged Particles in Pb-Pb Collisions atsNN=5.02 TeV". In: *Physical Review Letters* 116.13 (Apr. 2016). DOI: 10.1103/physrevlett.116.132302.

[65] Constantin Loizides, Jason Kamin, and David d'Enterria. "Improved Monte Carlo Glauber predictions at present and future nuclear colliders". In: *Physical Review C* 99.1 (Jan. 2019). arXiv: 1710.07098, p. 019901. DOI: 10.1103/PhysRevC.99.019901.

A. Appendix

A.1. Soft Herwig infile

```
cd /Herwig/EventHandlers

# own HepMC Reader class
create ThePEG::HepMCReader myReader
set myReader:Filename {hepmc_infile}
set myReader:Format GenEventHepMC3
# use MB Cuts
set myReader:Cuts /Herwig/Cuts/MinBiasCuts

# own HepMC EventHandler
create ThePEG::HepMCEventHandler myHepMCHandler

# set needed Handlers
set myHepMCHandler:CascadeHandler NULL
set myHepMCHandler:HadronizationHandler /Herwig/Hadronization/ClusterHadHandler
set myHepMCHandler:DecayHandler /Herwig/Decays/DecayHandler

cd /Herwig/Partons
cp PPExtractor myExtractor
set myExtractor:FirstPDF NoPDF
set myExtractor:SecondPDF NoPDF
cd /Herwig/EventHandlers
set myHepMCHandler:PartonExtractor /Herwig/Partons/myExtractor

set myHepMCHandler:HepMCReader myReader

cd /Herwig/Generators
cp EventGenerator myHepMCGenerator
set myHepMCGenerator:EventHandler /Herwig/EventHandlers/myHepMCHandler
```

A.2. Pseudorapidity plots for different centrality classes

Here two plots of less central centrality classes are included Figure A.2 and Figure A.3 to show that our model with the restriction of the number of collisions only gives good results for this most central region. In the centrality region 7.5 % to 10 % the restriction is

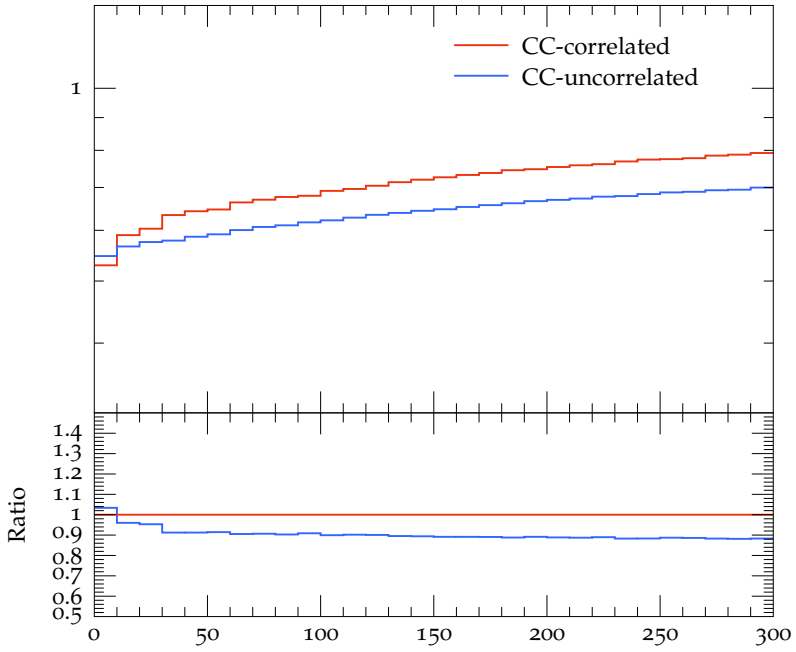


Figure A.1.: Qualitativ dummy analysis of the mean p_T for CC-collisions. The graph label 'CC-correlated' is with CR on the whole event, while the graph with the label 'CC-uncorrelated' is with CR only on the seperate NN-events.

not strong enough leading to larger multiplicity. For the centrality region 50 % to 60 % we even under-represent the data, which can be explained by realizing that for peripheral collisions the restriction does not play a crucial role, as the number of collision pairs the Glauber model provides is low anyway. The same analysis was executed for different setting of w_{diff} in Figure A.4 and Figure A.5. We can observe similar behavior.

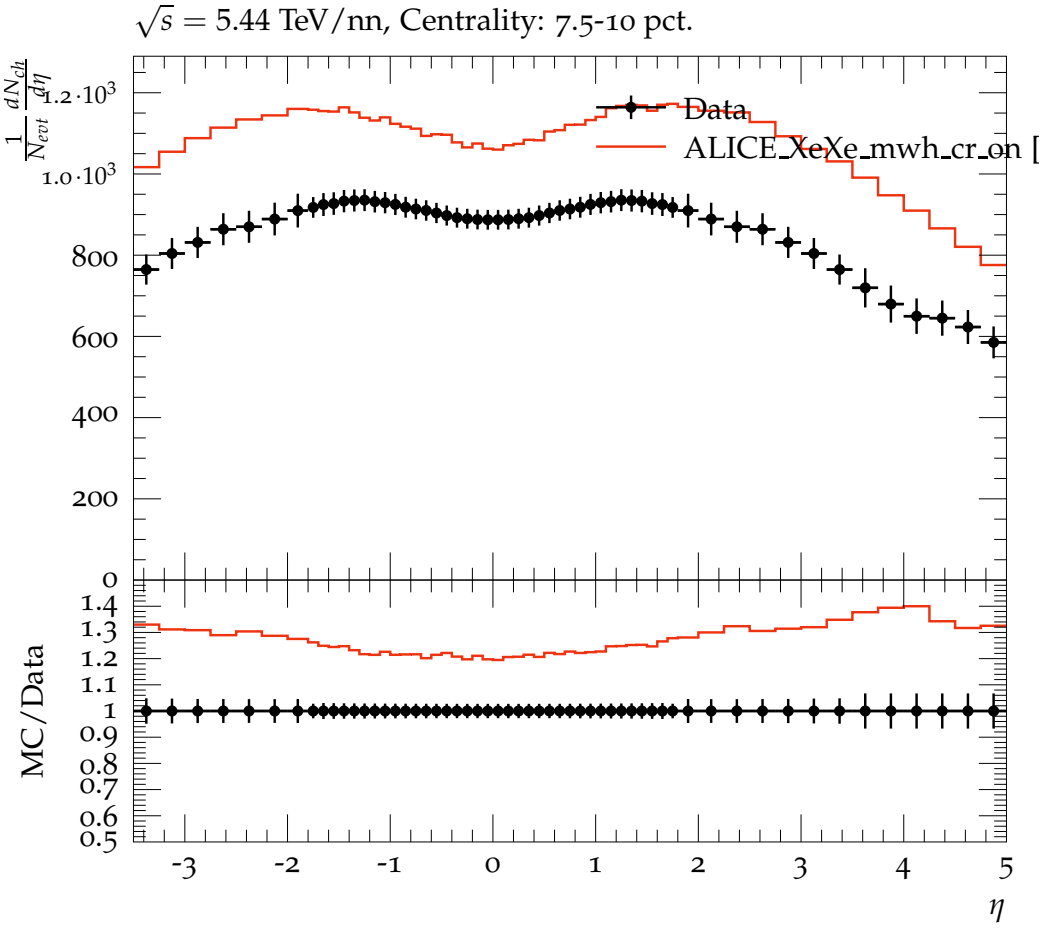


Figure A.2.: Pseudorapidity distribution for the centrality class 7.5 % to 10 %. The `simple_hi` run with hardness sorting, multiple NN-interactions, diffractive weight=0.5 and a hard cut on the number of allowed collisions. The Monte-Carlo data deviates clearly from the experimental data [61].

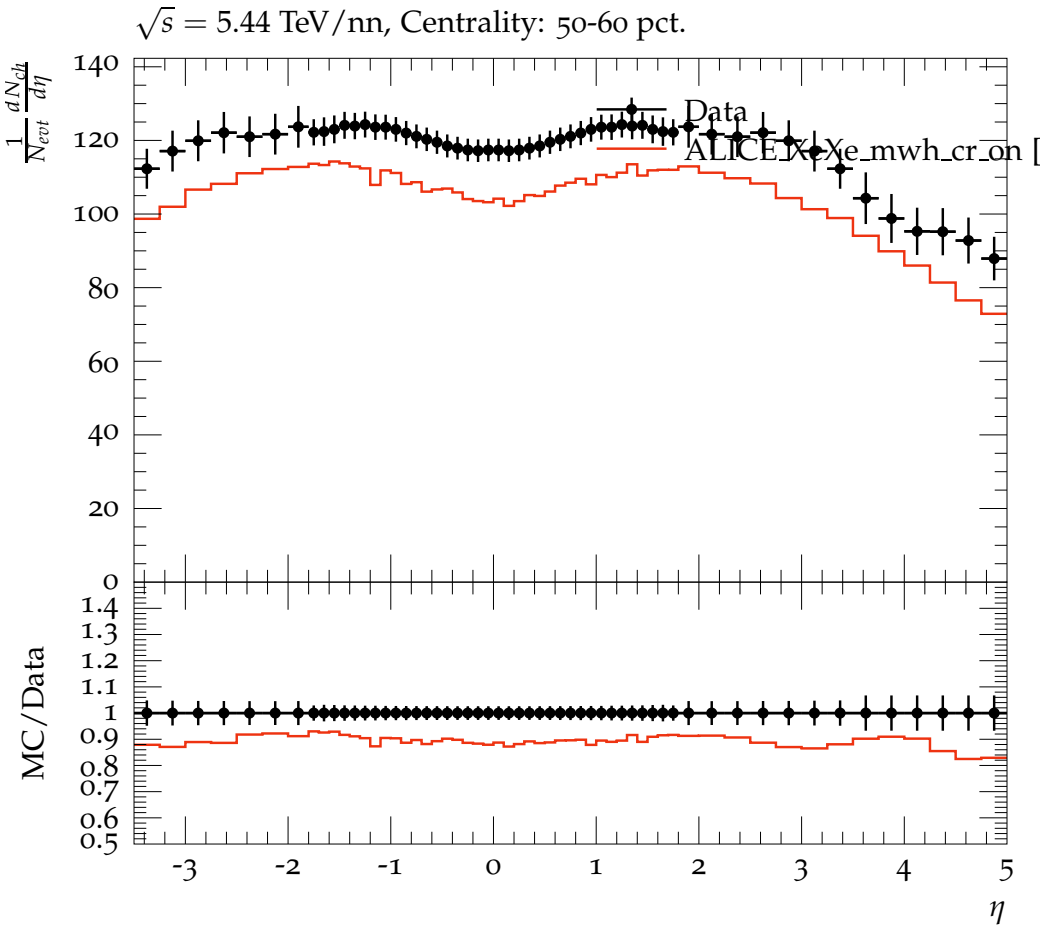


Figure A.3.: Pseudorapidity distribution for the centrality class 50 % to 60 %. The `simple_hi` run with hardness sorting, multiple NN-interactions, diffractive weight=0.5 and a hard cut on the number of allowed collisions. Data from ALICE [61]

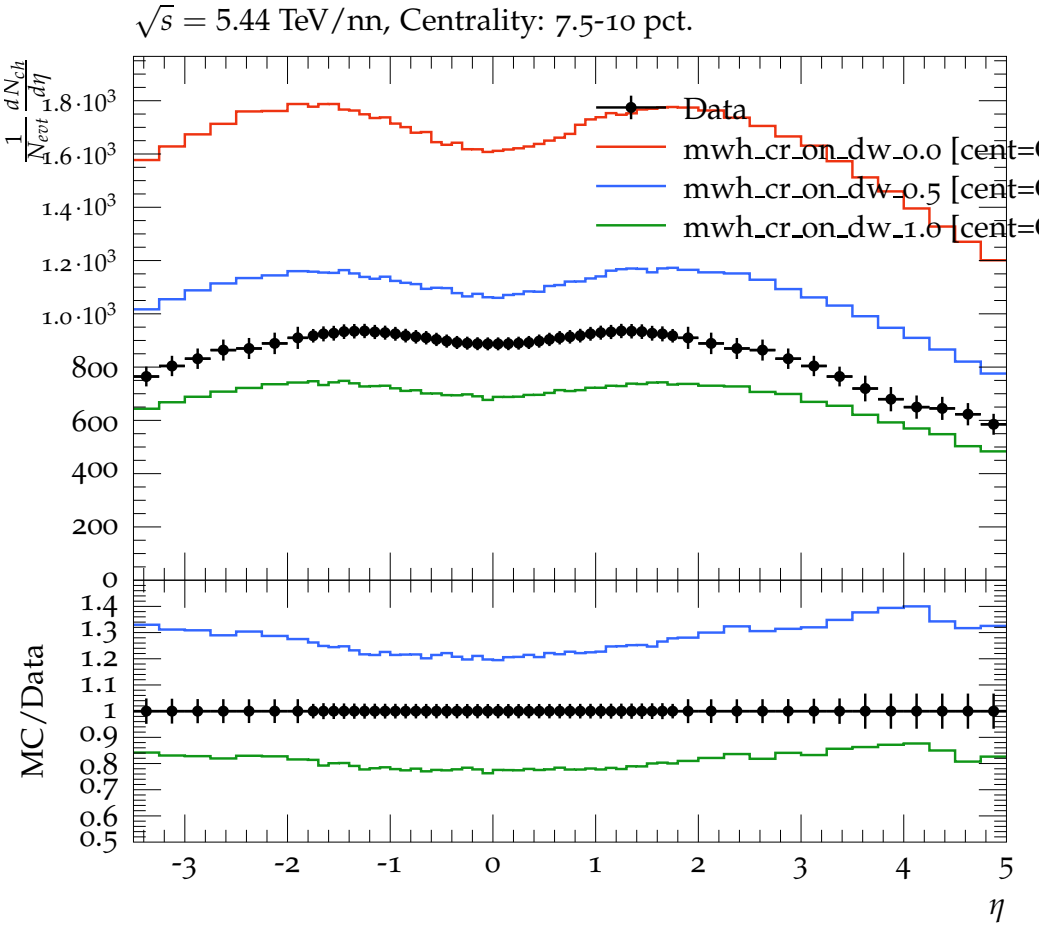


Figure A.4.: Pseudorapidity distribution for the centrality class 7.5 % to 10 % for different diffractive weights. Data from ALICE [61]

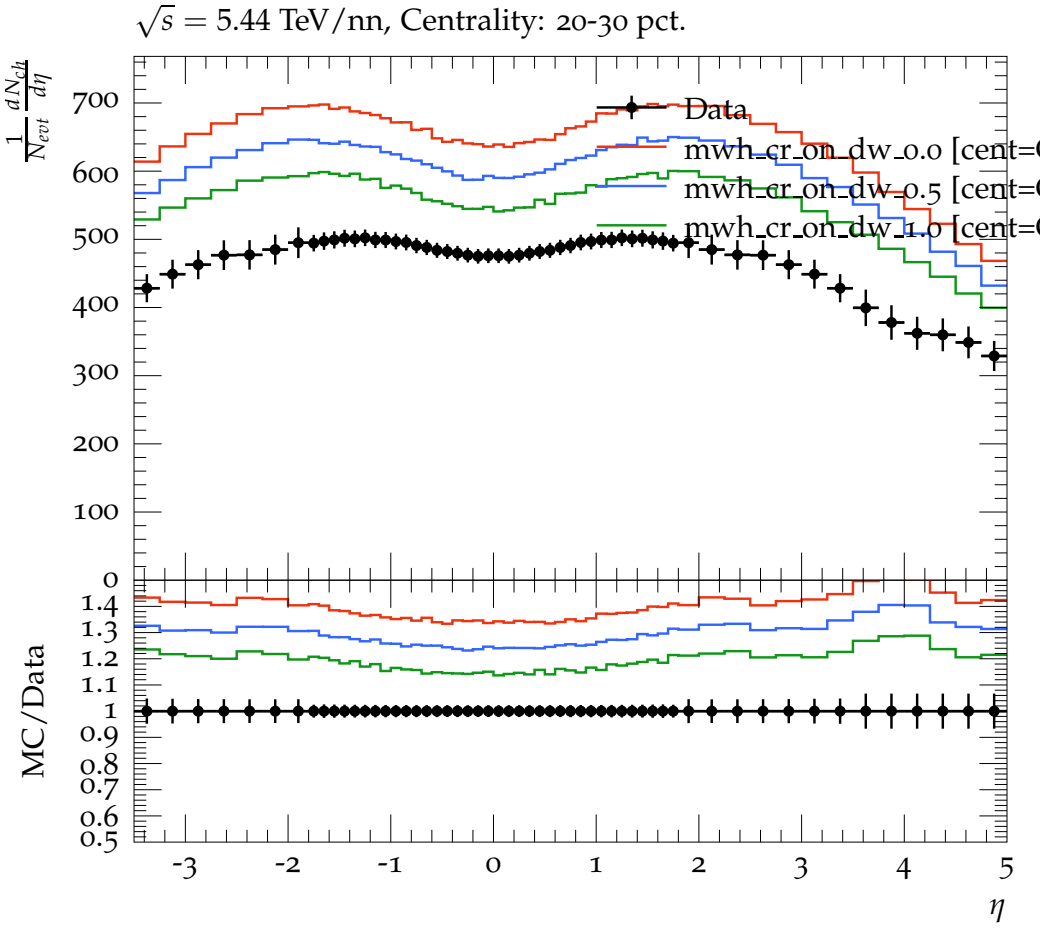


Figure A.5.: Pseudorapidity distribution for the centrality class 20 % to 30 % for different diffractive weights. Data from ALICE [61]