

Implementation of Next-to-Leading Order QCD Corrections for Hard Decays in the Sherpa Event Generator

Master-Arbeit
zur Erlangung des Hochschulgrades
Master of Science
im Master-Studiengang Physik

vorgelegt von

Lea Baumann

Institut für Kern- und Teilchenphysik
Fakultät Physik
Bereich Mathematik und Naturwissenschaften
Technische Universität Dresden
2026

Eingereicht am 17. März 2026

1. Gutachter: Prof. Dr. Frank Siegert
2. Gutachter: Prof. Dr. Dominik Stöckinger

Abstract

Monte Carlo event generators play a central role in particle collider data analysis by providing accurate theoretical predictions of complex scattering processes. Within SHERPA [1], the decays of heavy resonances are simulated using the hard decay framework. Up to now, they have only been calculated at leading order (LO). In this work, the next-to-leading order (NLO) quantum chromodynamics (QCD) calculation for the decay of a Higgs boson into a bottom-antibottom quark pair is implemented within the hard decay framework. The computation includes the evaluation of virtual loop corrections, real-emission contributions, and the cancellation of infrared (IR) singularities using subtraction terms. The individual components of the calculation were validated through comparisons with results from one of SHERPA's matrix element generators COMIX [2]. While the real-emission contributions and their subtraction terms are implemented consistently, discrepancies remain in the virtual correction and corresponding subtraction term, and therefore require further investigation. Nevertheless, first physical results were obtained by substituting these contributions with reference values from COMIX. The analysis shows that the inclusion of NLO corrections reduces the partial decay width of $H \rightarrow b\bar{b}$ compared to the LO prediction, improving the agreement with reference values from the literature [3]. In addition, several distributions of relevant observables were studied, comparing LO and NLO predictions.

Zusammenfassung

Monte-Carlo-Eventgeneratoren spielen eine zentrale Rolle bei der Datenanalyse von Teilchenbeschleunigern, indem sie genaue theoretische Vorhersagen komplexer Streuprozesse liefern. Innerhalb von SHERPA [1] werden Zerfälle von harten Resonanzen vom Hard Decay Framework simuliert. Diese werden bisher nur auf LO berechnet. In dieser Arbeit wird die Berechnung der QCD NLO für den Zerfall eines Higgs-Bosons in ein Bottom-Antibottom-Quark-Paar innerhalb des Hard Decay Modules implementiert. Die Berechnung umfasst die Auswertung virtueller Schleifenkorrekturen, Beiträge der Realemission und die Aufhebung von IR-Singularitäten mithilfe von Subtraktionstermen. Die einzelnen Komponenten der Berechnung wurden durch Vergleiche mit den Ergebnissen eines der Matrixelementgeneratoren von SHERPA, COMIX [2], validiert. Während die Beiträge der realen Emissionen und die Subtraktionsterme konsistent implementiert sind, bestehen weiterhin Diskrepanzen bei der virtuellen Korrektur und dem entsprechenden Subtraktionsterm, die daher einer weiteren Überprüfung bedürfen. Dennoch konnten erste physikalische Ergebnisse erzielt werden, indem diese Beiträge durch Referenzwerte von COMIX ersetzt wurden. Die Analyse zeigt, dass die Einbeziehung von NLO-Korrekturen die partielle Zerfallsbreite von $H \rightarrow b\bar{b}$ im Vergleich zur LO-Vorhersage verringert, was zu einer besseren Übereinstimmung mit den Referenzwerten aus der Literatur [3] führt. Darüber hinaus wurden mehrere Verteilungen relevanter Observablen untersucht und LO- und NLO-Vorhersagen verglichen.

Contents

1	Introduction	1
2	Theoretical Foundations	3
2.1	The Standard Model of Particle Physics	3
2.1.1	The Electroweak Theory	4
2.1.2	Spontaneous Symmetry Breaking	7
2.1.3	Quantum Chromodynamics	10
2.2	Fixed Order NLO QCD Calculation	11
2.2.1	Cross Sections of Proton-Proton and Electron-Positron Collisions . . .	11
2.2.2	Calculation of a NLO QCD Cross Section	14
2.3	SHERPA	17
2.3.1	Event Simulation in SHERPA	18
2.3.2	Narrow Width Approximation	21
3	NLO Calculation and Implementation	25
3.1	Preliminary Calculations	25
3.1.1	Real Subtraction $\mathcal{D}^S$	26
3.1.2	Colour Treatment in the $\mathcal{R}$ and $\mathcal{D}^S$ Contributions	28
3.1.3	Virtual Correction $\mathcal{V}$	31
3.1.4	Virtual Subtraction $\mathcal{I}^S$	35
3.2	Existing Hard Decay Framework	39
3.2.1	The Class Spin_Amplitudes	40
3.2.2	Hard Decay Initialization	42
3.2.3	Event Generation	46
3.3	Extension of the Hard Decay Framework	46
3.3.1	Implemented Spin_Amplitudes	47
3.3.2	Further Changes	49
4	Fixed Order Results	53
4.1	Result Verification and Correction	53
4.2	Results for the Process $e^+e^- \rightarrow HZ \rightarrow b\bar{b} \mu^+\mu^-$	54

5 Summary and Outlook	63
A LoopTools Usage and Script	65
B Python Script	69
C Run Cards	75
C.1 COMIX Run Card for the $\mathcal{R}$ and $\mathcal{D}^S$ Squared Matrix Elements Comparison .	75
C.2 $e^+e^- \rightarrow HZ \rightarrow b\bar{b} \mu^+\mu^-$ Event Generation	76
D Rivet Analysis	79
List of Tables	87
List of Figures	89
Bibliography	91

1 Introduction

Particle physics aims to improve our understanding of the fundamental building blocks of the universe, the elementary particles and their interactions. The standard model (SM) of particle physics describes these particles together with their interactions. It includes the strong interaction, which is described by QCD, as well as the electroweak interaction, which unifies the electromagnetic and the weak interaction. With the discovery of the Higgs boson in 2012, the particle content of the SM was completed [4, 5]. In particle-collision experiments, such as those performed at the Large Hadron Collider (LHC) at CERN, the SM is tested experimentally with high precision. At the same time, several open questions remain. Even small discrepancies between theoretical predictions and experimental measurements may provide hints of physics beyond the SM. Therefore, it is of great importance to calculate theoretical predictions with the highest possible precision.

The analysis of the large amounts of data produced in particle accelerators such as the LHC is extremely challenging. To interpret these data, Monte Carlo based event generators such as SHERPA are commonly used. These programs simulate particle collisions and provide theoretical predictions for observable quantities. In this way, complex event structures can be modelled and directly compared with experimental measurements. The underlying probabilities for scattering processes in particle physics are described by cross sections, which are calculated using the framework of perturbative quantum field theory. While calculations at the lowest order in perturbation theory are usually relatively simple, the determination of higher-order corrections quickly becomes significantly more complex. In higher order calculations in QCD, additional contributions in the perturbative expansion of the scattering amplitude in the strong coupling constant α_s have to be included. At NLO, in addition to the leading tree-level Feynman diagram, contributions from virtual gluons as well as processes with an additional gluon emission are taken into account.

In this thesis, the decay of a Higgs boson into a bottom–antibottom quark pair is studied at NLO in QCD. Although $H \rightarrow b\bar{b}$ is the dominant decay channel of the Higgs boson, it is comparatively difficult to measure because of the large QCD background from b-quark jets. In 2018, the ATLAS and CMS experiments succeeded in observing this decay mode in channels where the Higgs boson is produced in association with a massive vector boson (W or Z) [6, 7]. However, the first theoretical studies of the NLO decay width calculation of the $H \rightarrow b\bar{b}$

decay were already carried out in the 1980s, long before the discovery of the Higgs boson [8, 9]. These studies showed that higher-order corrections have a significant impact on the partial decay width of the Higgs boson. Later works extended these results to next-to-next-to-leading order (NNLO) [10, 11] and N³LO [12, 13].

In measurements of particle collisions, various phase-space cuts are applied in order to suppress background processes, for example on the transverse momentum or jet structure of the final-state particles. Inclusive decay widths or cross sections, however, are integrated over the full phase space and therefore also contain contributions from regions that are not experimentally accessible. As a result, such inclusive predictions cannot be compared directly with measured observables. Instead, differential predictions are required, since they describe how the decay rate or cross section depends on kinematic observables.

Besides performing calculations at fixed perturbative order, there are other ways to incorporate higher-order QCD effects. Additional QCD radiation from b jets can occur already at relatively low transverse momentum. While a fixed-order calculation includes only a finite number of explicit emissions, a parton shower provides an approximate description of multiple additional QCD emissions [14]. For this reason, modern Monte Carlo event generators often combine fixed-order calculations with parton-shower algorithms. Examples include approaches such as MC@NLO [15], POWHEG [16, 17], MadGraph5_aMC@NLO [18], and Herwig [19], as well as corresponding implementations in the SHERPA generator.

In this work, the decay process $H \rightarrow b\bar{b}$ was calculated at next-to-leading order in QCD and implemented in the hard decay module of SHERPA. The hard decay module represents a specific execution mode of the program in which the production and decay of heavy particles are treated in a factorized manner. The basis for this approach is the application of the narrow-width approximation (NWA), in which the intermediate particle is treated as on-shell. This allows the targeted generation of collision events containing specific intermediate particles, such as the Higgs boson. This is particularly useful when comparing events with and without a Higgs boson in the intermediate state within a given analysis.

Chapter 2 starts with an introduction to the SM. Subsequently, the calculation of cross sections and partial decay widths at NLO is discussed. In addition, the basic steps of event simulation in SHERPA and the principle of the NWA are briefly described. In Chapter 3, the analytical pre-calculation of several components required for the NLO computation is presented. This is followed by a description of the previous state of the hard decay module in SHERPA, and finally the modifications introduced by the new implementation are explained. Chapter 4 describes how the results were validated and presents the impact of the NLO corrections on the distributions of several observables for the example process $e^+e^- \rightarrow HZ \rightarrow b\bar{b} \mu^+ \mu^-$.

2 Theoretical Foundations

This chapter begins with an introduction to the SM. It then reviews the theoretical framework for NLO cross-section calculations and summarizes the principles of Monte Carlo event generation as implemented in SHERPA.

2.1 The Standard Model of Particle Physics

This chapter is based on refs. [20–25].

The Standard Model of particle physics provides a description of the fundamental constituents of matter and their interactions. Experiments can repeatedly verify its accuracy, whereby the existence of a larger, more fundamental theory cannot be excluded. It is widely regarded as an effective theory valid only up to a certain energy scale, since it neither incorporates gravity nor explains several observed phenomena such as neutrino masses, dark matter, or the baryon asymmetry of the universe [26–28]. The Standard Model describes three of the four known fundamental interactions: the electromagnetic, weak, and strong interactions. The electromagnetic and weak interactions are unified within the electroweak theory, while the strong interaction is described by QCD.

Gauge groups are used as a fundamental organizing principle of the SM and describe particle characteristics and the structure of the three fundamental interactions. It provides both a conceptual and mathematical framework for constructing consistent quantum field theories of interactions. The theoretical structure is based on the principle of local gauge invariance. Its gauge symmetry group is given by

$$SU(2)_L \times U(1)_Y \times SU(3)_C, \tag{2.1}$$

where $SU(3)_C$ corresponds to the strong interaction and $SU(2)_L \times U(1)_Y$ to the electroweak interaction.

While a global phase transformation leaves the free Dirac Lagrangian invariant, a local transformation introduces additional terms when derivatives act on the field. In order to restore

invariance, one replaces the ordinary derivative by a covariant derivative and introduces a gauge field that transforms appropriately under the local symmetry. This requirement predicts the existence of gauge bosons, which mediate the fundamental interactions. In non-Abelian gauge theories such as $SU(2)_L$ and $SU(3)_C$, the gauge fields themselves carry charge and therefore interact with each other, leading to the characteristic self-interactions of the weak and strong forces.

Gauge symmetries are closely related to conservation laws. According to Noether's theorem [29], every continuous global symmetry of the action implies a conserved current. Invariance under time translations implies conservation of energy, spatial translations imply conservation of momentum, and rotational invariance implies conservation of angular momentum. Similarly, gauge symmetries lead to conserved charges, such as electric charge or colour charge.

2.1.1 The Electroweak Theory

The U(1) Group

As mentioned above, continuous symmetries can imply conservation laws and, in the case of local gauge symmetries, require the introduction of interaction fields. In quantum field theory, particles are interpreted as excitations of the underlying quantum fields.

An important example of a local gauge symmetry is the U(1) symmetry. For a Dirac field $\psi(x)$ describing a spin-1/2 fermion, a local U(1) transformation takes the form

$$\psi(x) \rightarrow \psi'(x) = e^{iq\beta(x)}\psi(x), \quad (2.2)$$

with the phase $q \cdot \beta(x)$. $\beta(x)$ is a space-time dependent parameter and q denotes the coupling constant associated with the symmetry. Consider the free Dirac Lagrangian

$$\mathcal{L} = i\bar{\psi}'\gamma^\mu\partial_\mu\psi' - m\bar{\psi}'\psi', \quad (2.3)$$

with the adjoint spinor $\bar{\psi} = \psi^\dagger\gamma^0$.

When applying the transformation from equation 2.2 to this Lagrangian, the derivative term generates an additional contribution proportional to $\partial_\mu\beta(x)$, so that the Lagrangian is no longer invariant. To restore invariance, the ordinary derivative is replaced by the covariant derivative $D_\mu = \partial_\mu + iqB_\mu$, where B_μ is a newly introduced gauge field. The gauge field must transform as $B_\mu \rightarrow B'_\mu = B_\mu - \partial_\mu\beta(x)$ to preserve the gauge symmetry. To obtain a dynamic theory, a kinetic term for the gauge field must also be included. The resulting Lagrangian reads

$$\mathcal{L}_{EM} = i\bar{\psi}'\gamma^\mu D_\mu\psi' - m\bar{\psi}'\psi' - \frac{1}{4}B_{\mu\nu}B^{\mu\nu}, \quad (2.4)$$

where $B_{\mu\nu} = \partial_\mu B_\nu - \partial_\nu B_\mu$ is the field strength tensor.

The $SU(2)_L$ Group

Another gauge symmetry of the standard model is the $SU(2)_L$ gauge group. A defining feature of this symmetry is that it acts only on left-chiral fermion fields ψ_L and right-chiral anti fermion fields ψ_R . As a consequence, the weak interaction violates parity symmetry. A parity transformation is defined as the spatial inversion $\vec{x} \rightarrow -\vec{x}$. In the Standard Model, left-handed fermions are arranged into $SU(2)_L$ doublets, while right-handed fermions transform as singlets under $SU(2)_L$. For the three generations, the left-handed doublets are

$$\psi_L = \begin{pmatrix} \nu_e \\ e \end{pmatrix}_L, \begin{pmatrix} \nu_\mu \\ \mu \end{pmatrix}_L, \begin{pmatrix} \nu_\tau \\ \tau \end{pmatrix}_L, \begin{pmatrix} u \\ d \end{pmatrix}_L, \begin{pmatrix} c \\ s \end{pmatrix}_L, \begin{pmatrix} t \\ b \end{pmatrix}_L, \quad (2.5)$$

and the singlets:

$$\psi_R = u_R, d_R, c_R, s_R, t_R, b_R, e_R, \mu_R, \tau_R. \quad (2.6)$$

The weak interaction couples to the weak (flavour) eigenstates of fermions, which in general do not coincide with their mass eigenstates. In the quark sector, the weak and mass eigenstates are related by the CKM matrix, while in the lepton sector neutrino mixing is described by the PMNS matrix. The left-handed and right-handed spinor components are obtained by applying the projection operator $P_{L/R} = \frac{1}{2}(1 \mp \gamma_5)$ such that $\psi_{L/R} = P_{L/R}\psi$.

Consider first the kinetic part of the fermionic Lagrangian,

$$\mathcal{L} = \sum_f \bar{\psi}_{f,L} i\gamma^\mu \partial_\mu \psi_{f,L} + \bar{\psi}_{f,R} i\gamma^\mu \partial_\mu \psi_{f,R}, \quad (2.7)$$

with the sum running over all fermions f .

Local $SU(2)_L$ transformations of left-handed fermion fields are

$$\psi_L \rightarrow \psi'_L = e^{ig_W \alpha_a(x) T^a} \psi_L, \quad (2.8)$$

where $a = 1, 2, 3$, $\alpha_a(x)$ is a momentum-dependent parameter and g_W is the weak coupling constant, which is a free parameter and needs to be determined experimentally. The generators of the $SU(2)_L$ group are $T^a = \frac{1}{2}\sigma^a$, with the Pauli matrices σ^a .

To ensure invariance under such transformations, the ordinary derivative must be replaced by the covariant derivative, introducing three gauge fields W_μ^a :

$$\partial_\mu \rightarrow D_\mu = \partial_\mu + ig_W W_\mu^a T^a. \quad (2.9)$$

The fields transform like

$$W_\mu^a \rightarrow W_\mu^{a'} = W_\mu^a - \frac{1}{g_W} \partial_\mu \alpha^a(x) - \epsilon^{abc} \alpha^b(x) W_\mu^c. \quad (2.10)$$

In contrast to the Abelian $U(1)$ case, the non-Abelian structure constants ϵ^{abc} generate self-interaction terms among the gauge fields.

As before, the introduction of the new fields lead to an additional kinetic part in the Lagrangian by adding the field strength tensors $W_{\mu\nu}^a = \partial_\mu W_\nu^a - \partial_\nu W_\mu^a + g_W \epsilon^{abc} W_\mu^b W_\nu^c$, such that the full massless weak Lagrangian is

$$\mathcal{L}_W = \sum_f i \bar{\psi}_{f,L} \gamma^\mu D_\mu \psi_{f,L} + i \bar{\psi}_{f,R} \gamma^\mu D_\mu \psi_{f,R} - \frac{1}{4} W_{\mu\nu}^a W^{\mu\nu a}. \quad (2.11)$$

The gauge fields B_μ and $W_\mu^1, W_\mu^2, W_\mu^3$ do not directly correspond to experimentally detectable boson fields. For this, the $U(1)$ must be unified with the $SU(2)_L$ group.

The $U(1)_Y \times SU(2)_L$ Group

Through unification, $U(1)$ is replaced by a $U(1)_Y$ symmetry

$$\psi \rightarrow \psi' = e^{ig_Y Y \beta(x)} \psi, \quad (2.12)$$

where Y is the weak hypercharge, the quantum number of quantum electrodynamics (QED), and g_Y is the associated coupling constant. The hypercharge assignments are chosen such that electric charge is related to the weak isospin and hypercharge through the Gell-Mann–Nishijima relation

$$q = I_3 + Y/2. \quad (2.13)$$

For left-handed fermions in $SU(2)_L$ doublets, the third component of weak isospin takes the values $I_3 = \pm \frac{1}{2}$, while right-handed singlets have $I_3 = 0$. The covariant derivative of the full electroweak theory is

$$D_\mu = \partial_\mu + ig_W T^a W_\mu^a + ig_Y Y B_\mu, \quad (2.14)$$

where W_μ^a ($a = 1, 2, 3$) are the $SU(2)_L$ gauge fields and B_μ is the $U(1)_Y$ gauge field. Left-handed fermions transform non-trivially under both gauge groups, whereas right-handed fermions are singlets under $SU(2)_L$ and transform only under $U(1)_Y$:

$$\psi'_L = e^{ig_W \alpha_a(x) T^a} \cdot e^{ig_Y Y \beta(x)} \psi_L, \quad \psi'_R = e^{ig_Y Y \beta(x)} \psi_R. \quad (2.15)$$

The transformation of the W_μ^a fields is given in 2.10, while the $U(1)_Y$ gauge field transforms as

$$B_\mu \rightarrow B'_\mu = B_\mu - \frac{1}{g_Y} \partial_\mu \beta(x). \quad (2.16)$$

Combining the $SU(2)_L$ and $U(1)_Y$ sectors yields the full gauge-invariant electroweak Lagrangian (without mass terms). It consists of fermionic kinetic terms, gauge interactions arising from the covariant derivative, and kinetic terms for the gauge fields,

$$\mathcal{L}_{EW} = \sum_f i \bar{\psi}_{f,L} \gamma^\mu D_\mu \psi_{f,L} + i \bar{\psi}_{f,R} \gamma^\mu D_\mu \psi_{f,R} - \frac{1}{4} W_{\mu\nu}^a W^{a\mu\nu} - \frac{1}{4} B_{\mu\nu} B^{\mu\nu}. \quad (2.17)$$

At this stage, the four gauge fields W_μ^1 , W_μ^2 , W_μ^3 and B_μ are not yet the physical vector bosons observed experimentally. After electroweak symmetry breaking, which is introduced in the next section, the charged gauge bosons are defined as

$$W_\mu^\pm = \frac{1}{\sqrt{2}} (W_\mu^1 \mp i W_\mu^2), \quad (2.18)$$

while the neutral gauge fields mix according to

$$\begin{pmatrix} A_\mu \\ Z_\mu \end{pmatrix} = \begin{pmatrix} \cos \theta_W & \sin \theta_W \\ -\sin \theta_W & \cos \theta_W \end{pmatrix} \begin{pmatrix} B_\mu \\ W_\mu^3 \end{pmatrix}. \quad (2.19)$$

The weak mixing angle θ_W is determined experimentally.

The final electroweak Lagrangian describes free fermions and their interactions with the electroweak gauge field, as well as gauge-field self interactions. However, what is still missing from this theory are terms that describe the mass of fermions and gauge bosons, since experimental observations show that fermions and the $W^\pm$ and Z bosons are massive. Explicit mass terms such as $m_f \bar{\psi} \psi$, $\frac{1}{2} m_B^2 B_\mu B^\mu$ or $\frac{1}{2} m_W^2 W_\mu^a W^{a\mu}$ would break gauge invariance. This problem is solved by the Weinberg-Salam model, which was introduced by Stephen Weinberg and Abdus Salam [30, 31]. It is based on the Higgs mechanism [32, 33].

2.1.2 Spontaneous Symmetry Breaking

Gauge boson masses are generated via spontaneous symmetry breaking through the introduction of a complex scalar field Φ , transforming as an $SU(2)_L$ doublet. The choice of a complex doublet ensures the correct number of degrees of freedom to generate masses for the $W^\pm$ and

Z bosons while leaving the photon massless. The scalar field can be written as

$$\Phi = \begin{pmatrix} \phi_+ \\ \phi_0 \end{pmatrix} = \begin{pmatrix} \phi_1 + i\phi_2 \\ \phi_3 + i\phi_4 \end{pmatrix}. \quad (2.20)$$

The Higgs sector is described by the Lagrangian

$$\mathcal{L}_{SSB} = (D_\mu \Phi)^\dagger D^\mu \Phi - V(\Phi), \quad (2.21)$$

with the covariant derivative defined as in 2.14 and the Higgs potential

$$V(\Phi) = \mu^2 \Phi^\dagger \Phi + \lambda (\Phi^\dagger \Phi)^2, \quad (2.22)$$

where μ and λ are free parameters. For $\mu^2 < 0$ and $\lambda > 0$, the potential has a continuous set of degenerate minima satisfying

$$|\phi_+|^2 + |\phi_0|^2 = \frac{-\mu^2}{2\lambda}. \quad (2.23)$$

Choosing a specific minimum spontaneously breaks the symmetry. All minima are equivalently valid, so there is some degree of freedom to choose the values of ϕ_1 , ϕ_2 , ϕ_3 and ϕ_4 . The charged part ϕ_+ shall be 0 so that the electromagnetic symmetry remains intact. Otherwise, the photon would obtain mass. A convenient choice is $\phi_1 = \phi_2 = \phi_4 = 0$, which results in the minimum being at

$$\Phi_0 = \begin{pmatrix} 0 \\ \frac{v}{\sqrt{2}} \end{pmatrix}, \quad (2.24)$$

with the vacuum expectation value (VEV)

$$v = \sqrt{\frac{-\mu^2}{\lambda}} = 246 \text{ GeV}, \quad (2.25)$$

as determined from experiment. Three of the four degrees of freedom correspond to Goldstone modes. They are unphysical, but by choosing the right gauge, the unitary gauge, these modes are absorbed by the gauge fields and become the longitudinal polarization states of the $W^\pm$ and Z bosons, giving them mass. The remaining physical degree of freedom corresponds to the scalar Higgs boson. This additional spin 0 particle is obtained by taking into account small fluctuations around the vacuum ground state in unitary gauge,

$$\Phi = \begin{pmatrix} 0 \\ \frac{v+h(x)}{\sqrt{2}} \end{pmatrix}, \quad (2.26)$$

where $h(x)$ denotes the physical Higgs field. Inserting this expression into the kinetic term in 2.21 and using the electroweak covariant derivative 2.14 yields

$$\begin{aligned}\mathcal{L}_{SSB,kin} = & \frac{1}{2}\partial_\mu h(x)\partial^\mu h(x) + \frac{1}{8}g_W^2(v+h(x))^2|W_\mu^1 - iW_\mu^2|^2 \\ & + \frac{1}{8}(v+h(x))^2|g_W W_\mu^3 - g_Y B_\mu|^2.\end{aligned}\quad (2.27)$$

The first term corresponds to a kinetic term of the Higgs field. The other terms contain mass terms for the gauge fields as well as interaction terms between the gauge and Higgs fields. After expressing the gauge fields in terms of the physical states $W^\pm$, Z , and A , one obtains the desired mass expressions for the physical gauge bosons, which are

$$m_W = \frac{g_W v}{2}, \quad m_Z = \frac{m_W}{\cos \theta_W}, \quad m_A = 0, \quad (2.28)$$

where $\tan \theta_W = \frac{g_Y}{g_W}$. It is related to the electric charge e via $g_W \sin \theta_W = g_Y \cos \theta_W = e$. The photon remains massless because the $U(1)_{em}$ symmetry remains unbroken. When expanding the Higgs potential term in 2.21, one gets

$$\mathcal{L}_{SSB,V} = \frac{\mu^2}{2}(v+h(x))^2 + \lambda(v+h(x))^4, \quad (2.29)$$

which yields a mass term for the Higgs boson $m_H = \sqrt{2\lambda}v$. In addition, 2.29 contains triple and quartic Higgs self interaction terms.

Although spontaneous symmetry breaking generates mass terms, the actual numerical values of the masses depend on the free parameters v , λ , θ_W and g_W . Therefore, the masses have to be measured experimentally. This was done in the last decades for the $W^\pm$, Z and Higgs bosons. Their values are [3]:

$$m_W = 80.369 \pm 0.013 \text{ GeV}, \quad m_Z = 91.188 \pm 0.002 \text{ GeV}, \quad m_H = 125.20 \pm 0.11 \text{ GeV}. \quad (2.30)$$

Yukawa Coupling

The generation of fermion masses requires gauge invariant couplings between the fermion fields and the Higgs doublet. These can be formed by introducing Yukawa interactions between the fermions and the Higgs field. For one fermion generation, the Yukawa Lagrangian takes the form

$$\mathcal{L}_{Yuk} = -y_{f,1}\bar{\psi}_{f,L}\Phi\psi_{f,R} - y_{f,2}\bar{\psi}_{f,L}\Phi_c\psi_{f,R} + \text{h.c.} \quad (2.31)$$

where $\Phi_c = -i\sigma_2\Phi^*$ is the conjugate Higgs doublet. The first term contains the mass terms for down-type quarks and charged leptons, and the second one for up-type quarks. The Yukawa couplings y_f are dimensionless parameters and must be determined experimentally. After

spontaneous symmetry breaking, inserting 2.26 into the Yukawa Lagrangian generates fermion mass terms $m_f = \frac{y_f}{\sqrt{2}}v$, as well as interaction terms between the physical Higgs field $h(x)$ and fermion pairs. In the minimal Standard Model, neutrinos remain massless because no right-handed neutrino fields are included and therefore no Dirac Yukawa term can be written. The experimentally observed small neutrino masses require an extension of the model, for example through a seesaw mechanism [34].

2.1.3 Quantum Chromodynamics

The third fundamental interaction that is described by the SM is QCD, the theory of the strong interaction. It is based on the non-Abelian gauge group $SU(3)_C$.

Quarks carry an additional internal quantum number called colour charge, which comes in three components, red, blue and green. Leptons do not couple to the strong interaction and are therefore singlets under $SU(3)_C$ colour transformations. Quarks instead transform as triplets under $SU(3)_C$. The generators of the $SU(3)_C$ group in the fundamental representation are given by eight 3x3 matrices T_s^a , which are derived from the Gell Mann matrices λ^a :

$$T_s^a = \frac{\lambda^a}{2}, \quad a = 1, \dots, 8. \quad (2.32)$$

These generators satisfy the algebra

$$[\lambda^i, \lambda^j] = 2if^{abc}\lambda^c, \quad (2.33)$$

with the $SU(3)_C$ structure constant f^{abc} .

The QCD Lagrangian is

$$\mathcal{L}_{QCD} = \sum_f i\bar{q}_f \gamma^\mu D_\mu q_f - \frac{1}{4} G_{\mu\nu}^a G^{\mu\nu a}, \quad (2.34)$$

with the first term only acting on quarks q_f . The field strength tensors are defined as

$$G_{\mu\nu}^a = \partial_\mu G_\nu^a - \partial_\nu G_\mu^a + g_s f^{abc} G_\mu^b G_\nu^c, \quad (2.35)$$

and the covariant derivative reads

$$D_\mu = \partial_\mu + ig_s T_S^a G_\mu^a. \quad (2.36)$$

The eight field strength tensors represent the eight gluon fields. The non-Abelian term in 2.35 proportional to f^{abc} leads to triple and quartic gluon self-interactions. The strong coupling constant is commonly written as $g_s = \sqrt{4\pi\alpha_s}$, where α_s is the energy-dependent

strong coupling parameter. At low energy, its value is approximately one, leading to colour confinement: coloured states such as quarks and gluons cannot be observed as isolated particles but are confined into colour-neutral hadrons. At high energies, the strong coupling exhibits asymptotic freedom, meaning that α_s decreases and quarks behave approximately as free particles.

The Dirac field and the field strength tensor transform like

$$\psi \rightarrow \psi' = e^{ig_S \gamma_a(x) T_S^a} \psi, \quad G_\mu^a \rightarrow G_\mu^{a'} = G_\mu^a - \frac{1}{g_S} \partial_\mu \gamma_a(x) - f^{abc} G_\mu^b \gamma_c(x) \quad (2.37)$$

under local $SU(3)_C$ transformations, with the space-time dependent parameters $\gamma_a(x)$.

In summary, the full Standard Model Lagrangian can be written schematically as

$$\mathcal{L}_{SM} = \mathcal{L}_{EW} + \mathcal{L}_{QCD} + \mathcal{L}_{SSB} + \mathcal{L}_{Yuk}. \quad (2.38)$$

2.2 Fixed Order NLO QCD Calculation

As described above, the strong coupling constant α_s depends on the characteristic energy scale of the process. At sufficiently high energies, α_s becomes small, allowing for a perturbative expansion of physical observables. In this regime, cross sections and decay widths can be computed using fixed-order perturbation theory. In a fixed-order QCD calculation, the cross section is expanded as a power series in α_s and truncated at a given order, such as LO or NLO, while higher-order contributions are neglected. Each perturbative order corresponds to a specific class of Feynman diagrams contributing to the amplitude. They encode the application of Feynman rules derived from the underlying Lagrangian. In this work, the cross section for the decay of a Higgs boson into bottom quarks is calculated at NLO. The relevant energy scale of the process is set by the Higgs mass, $m_H \approx 125 \text{ GeV}$ [3]. At this scale, the strong coupling constant has the approximate value $\alpha_s(m_H) \approx 0.1$ [35], which justifies the use of perturbation theory. Before discussing higher-order corrections, the general structure of cross sections in proton–proton and electron–positron collisions is briefly reviewed. Subsequently, the inclusion of NLO contributions is explained. The first part of this chapter follows mainly [36, 37].

2.2.1 Cross Sections of Proton-Proton and Electron-Positron Collisions

The hadronic cross section for a proton–proton collision producing a final state with n hard partons can be computed using Feynman rules that emerge from the SM Lagrangian 2.38. It is expressed as a convolution of parton distribution functions (PDFs) with the corresponding partonic cross section.

Since protons are composite objects, the calculation requires the PDFs $f_a^{h_1}(x_a, \mu_F)$ and $f_b^{h_2}(x_b, \mu_F)$. These functions describe the probability density of finding a parton of type a or b with momentum fraction x_a or x_b (Bjorken variables) inside of the incoming hadrons h_1 and h_2 , evaluated at the factorization scale μ_F . The factorization scale separates the low-energy ($E < \mu_F$) non-perturbative physics from the high-energy ($E > \mu_F$) perturbatively calculable contributions. Effects below μ_F are absorbed into the PDFs, while hard interactions above this scale are described by the partonic matrix elements. The PDFs cannot be calculated fully analytical. Instead, they are extracted from experimental data at a reference scale and evolved to other scales using the DGLAP equations. The hadronic cross section is obtained by convoluting the PDFs with the partonic cross section:

$$\sigma_{pp \rightarrow n} = \sum_{a,b} \int_0^1 \int_0^1 dx_a dx_b f_a^{h_1}(x_a, \mu_F) f_b^{h_2}(x_b, \mu_F) \hat{\sigma}_{ab \rightarrow n}(\hat{s}, \mu_R, \mu_F) \quad (2.39)$$

where $\hat{\sigma}_{ab \rightarrow n}(\hat{s}, \mu_R, \mu_F)$ denotes the partonic cross section for the scattering of partons a and b into n final-state partons. It depends on the partonic centre-of-mass energy $\hat{s} = x_a x_b s$, as well as on the renormalization scale μ_R and the factorization scale μ_F . The renormalization scale enters through the running coupling constant $\alpha_s(\mu_R)$. The sum over a and b accounts for all possible partonic initial states inside the proton. Since the proton consists of quarks, antiquarks, and gluons, each of these partons can, in principle, contribute to the production of the n -parton final state. The partonic cross section is computed by applying Feynman rules,

$$\hat{\sigma}_{a,b \rightarrow n}(\mu_F, \mu_R) = \int_n d\Phi_n \mathcal{F} |\mathcal{M}_{ab \rightarrow n}(\Phi_n; \mu_F, \mu_R)|^2, \quad (2.40)$$

where $\mathcal{M}_{ab \rightarrow n}$ is the scattering amplitude obtained by summing all contributing Feynman diagrams at a given perturbative order, and the flux factor is given by $\mathcal{F} = \frac{1}{2\hat{s}}$. The n -particle phase space element is given by

$$d\Phi_n = \prod_{i=1}^n \frac{d^3 p_i}{(2\pi)^3 2E_i} (2\pi)^4 \delta^{(4)}(p_a + p_b - \sum_{i=1}^n p_i), \quad (2.41)$$

where the delta distribution ensures energy-momentum conservation between the incoming partons with momenta p_a and p_b and outgoing particles with momenta p_i . In principle, these expressions are valid to all orders in perturbation theory. In practice, the matrix element is truncated at a fixed perturbative order (e.g. LO, NLO, or NNLO). If all orders were included, the physical cross section would be independent of both the renormalization scale μ_R and the factorization scale μ_F . However, this is not feasible in reality, which is why these parameters must be set manually.

Unlike protons, electrons and positrons are elementary, point-like particles and therefore do not possess an internal structure. In the absence of QED radiation, the full centre-of-mass energy

is available for the hard scattering process. At high energies, however, electron and positron beams emit collinear QED radiation before the collision. This effect is known as initial-state radiation (ISR). As a consequence, the particles that actually participate in the hard interaction may carry only a fraction of the original beam energy. In addition to electrons and positrons, photons radiated from the beams can also enter the hard subprocess. The corresponding cross section can be written in a factorised form that resembles the hadronic case, but with lepton distribution functions instead of hadronic PDFs. These functions describe the probability of finding an electron, positron or photon inside the incoming electron or positron after QED radiation. In contrast to hadronic PDFs, they are fully determined within perturbative QED and depend primarily on the lepton mass and energy. The electron–positron cross section is then given by [38]

$$\sigma_{e^+e^- \rightarrow n} = \sum_{a,b} \int_0^1 \int_0^1 dx_a dx_b f_a^{e^-}(x_a, m_e, \mu_F) f_b^{e^+}(x_b, m_e, \mu_F) \hat{\sigma}_{ab \rightarrow n}(\hat{s}, \mu_R, \mu_F), \quad (2.42)$$

where $f_a^{e^-}(x_a, m_e, \mu_F)$ and $f_b^{e^+}(x_b, m_e, \mu_F)$ denote the electron and the positron distribution functions, and m_e is the electron mass.

Decay Widths

Many particles in the SM are unstable and decay after a finite lifetime into lighter particles. The decay width provides a quantitative measure of how quickly this process occurs and is directly related to the lifetime of the particle. Instead of two incoming particles, one considers a single unstable particle that transitions into a multi-particle final state. The transition probability is again determined by the squared matrix element integrated over the phase space, leading to the definition of the partial decay width. The partial decay width of a particle p with mass m_p decaying into a final state f containing n particles, is given by [39]

$$\Gamma = \int_n d\Phi_n^\Gamma \frac{1}{2m_p} |\mathcal{M}_{p \rightarrow f}(\Phi_n^\Gamma; \mu_F, \mu_R)|^2, \quad (2.43)$$

with the phase space element being

$$d\Phi_n^\Gamma = \prod_{i=1}^n \frac{d^3 p_i}{(2\pi)^3 2E_i} (2\pi)^4 \delta^{(4)}(p_p - \sum_{i=1}^n p_i). \quad (2.44)$$

The total decay width of the particle p is obtained by summing over all kinematically allowed final states,

$$\Gamma_{tot} = \sum_f \Gamma(p \rightarrow f). \quad (2.45)$$

The branching ratio of the decay channel $p \rightarrow f$ is defined as

$$\text{BR}(p \rightarrow f) = \frac{\Gamma(p \rightarrow f)}{\Gamma_{\text{tot}}}. \quad (2.46)$$

In perturbation theory, the decay width can be computed at different orders in the coupling expansion, analogously to scattering cross sections. The NLO corrections discussed in the following sections can therefore be applied to decay width calculations as well. However, the formalism will be illustrated using the example of a partonic cross section.

2.2.2 Calculation of a NLO QCD Cross Section

Starting from a LO cross section, the NLO QCD prediction is obtained by including all contributions that are one order higher in the strong coupling constant α_s compared to the LO. In terms of the matrix element in equation 2.40, this corresponds to extending the amplitude $\mathcal{M}_{ab \rightarrow n}(\Phi_n; \mu_F, \mu_R)$ by adding the relevant higher-order Feynman diagrams. At NLO in QCD, two types of corrections must be taken into account:

- Virtual loop corrections $\mathcal{V}$, and
- Real emission contributions $\mathcal{R}$.

The partonic cross section at NLO accuracy can be written schematically as

$$\begin{aligned} \hat{\sigma}_{a,b \rightarrow n}^{\text{NLO}} &= \underbrace{\int_n d\tilde{\Phi}_n \mathcal{B}_n(\Phi_n; \mu_F, \mu_R)}_{\text{LO}} + \underbrace{\int_n d\tilde{\Phi}_n \mathcal{V}_n(\Phi_n; \mu_F, \mu_R) + \int_{n+1} d\tilde{\Phi}_{n+1} \mathcal{R}_n(\Phi_{n+1}; \mu_F, \mu_R)}_{\text{NLO}} \\ &= \int_n d\tilde{\Phi}_n |\mathcal{M}_{\mathcal{B}}(\Phi_n; \mu_F, \mu_R)|^2 + \int_n d\tilde{\Phi}_n 2\Re\{\mathcal{M}_{\mathcal{B}}^*(\Phi_n; \mu_F, \mu_R) \mathcal{M}_{\mathcal{V}}(\Phi_n; \mu_F, \mu_R)\} \\ &\quad + \int_{n+1} d\tilde{\Phi}_{n+1} |\mathcal{M}_{\mathcal{R}}(\Phi_{n+1}; \mu_F, \mu_R)|^2. \end{aligned} \quad (2.47)$$

Here, $\mathcal{B}_n$, $\mathcal{V}_n$ and $\mathcal{R}_n$ denote the corresponding differential cross section contributions. The full cross section is obtained after integration over the phase space. The tilde indicates that the phase space element $d\tilde{\Phi}$ is defined to include the flux factor $\mathcal{F}$. The LO contribution is referred to as the Born term $\mathcal{B}_n$. It is calculated with the Born amplitude $\mathcal{M}_{\mathcal{B}}$, which is the lowest-order Feynman diagram contributing to the process. For example, in the decay of a Higgs boson into a quark–antiquark pair, the Born diagram corresponds to the tree-level process $H \rightarrow q\bar{q}$, as illustrated in figure 2.1.

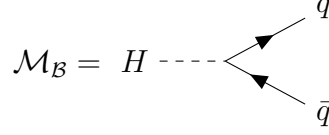


Figure 2.1: Born (LO) Feynman diagram for the decay $H \rightarrow q\bar{q}$.

The virtual contribution,

$$\int d\tilde{\Phi}_n \mathcal{V}_n, \quad (2.48)$$

is obtained from the interference between the Born amplitude $\mathcal{M}_B$ and the one-loop amplitude $\mathcal{M}_V$. The virtual Feynman diagram for the Higgs decay process is shown in figure 2.2, it contains a gluon loop. The virtual contribution is integrated over the same n-particle phase space as the Born term, since no additional external particles appear.

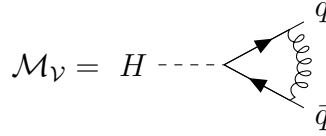


Figure 2.2: Virtual Feynman diagram for the decay $H \rightarrow q\bar{q}$.

Expanding the squared matrix element by adding the virtual amplitude yields

$$|\mathcal{M}_B + \mathcal{M}_V|^2 = |\mathcal{M}_B|^2 + 2 \Re \{ \mathcal{M}_B^* \mathcal{M}_V \} + |\mathcal{M}_V|^2. \quad (2.49)$$

The Born amplitude is of order α_s^0 for the decay $H \rightarrow q\bar{q}$, while the one-loop amplitude is of order α_s^1 . Therefore, the interference term $2 \Re \{ \mathcal{M}_B^* \mathcal{M}_V \}$ contributes at order α_s^1 and thus belongs to NLO. In contrast, the term $|\mathcal{M}_V|^2$ is of order α_s^2 and therefore contributes only at NNLO. For this reason, it is not included in the NLO cross section.

The real-emission contribution,

$$\int d\tilde{\Phi}_{n+1} \mathcal{R}_n, \quad (2.50)$$

arises from tree-level diagrams with one additional parton in the final state. For the Higgs decay example, this corresponds to the process $H \rightarrow q\bar{q}g$, where the gluon can be emitted either from the quark or the antiquark line, as illustrated in figure 2.3. Since an additional particle appears in the final state, the integration is performed over the (n+1)-particle phase space. The real-emission matrix element is of order $\sqrt{\alpha_s^1}$, and therefore its squared contribution

directly enters at NLO.

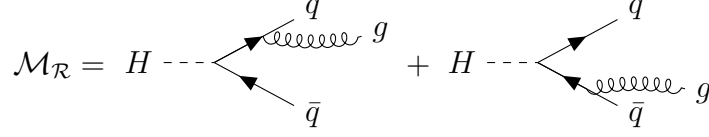


Figure 2.3: Real emission Feynman diagrams for the decay $H \rightarrow q\bar{q}$.

In certain regions of phase space, the real-emission matrix element becomes singular. This occurs when the emitted gluon becomes soft, i.e. has low energy (soft singularity) or when it is emitted collinearly to another massless parton (collinear singularity). These divergences are referred to as IR singularities. The virtual loop diagram contains corresponding IR singularities originating from soft and collinear regions of the loop momentum. Although the real and virtual contributions are individually infrared divergent, their singularities cancel in the sum for IR-safe observables, such as inclusive cross sections. This cancellation is guaranteed by the KLN (Kinoshita–Lee–Nauenberg) theorem [40, 41].

However, this leads to a problem when calculating NLO cross sections with Monte Carlo generators. Real and virtual contributions have different final state multiplicities and are therefore integrated over different phase spaces before the summation: the virtual corrections over the n -particle phase space, and the real-emission contributions over the $(n+1)$ -particle phase space. Since both integrals are individually divergent, a direct Monte Carlo integration would fail, as the sampling includes the singular regions of phase space. In order to obtain finite and numerically integrable contributions, the IR divergences have to be subtracted locally before performing the phase space integration. Two methods are commonly used for this at NLO: the phase space slicing method [42] and the subtraction method [43]. In this work, the subtraction method is employed. The central idea of the subtraction method is to introduce local counterterms that reproduce the singular behaviour of the real-emission matrix element in all soft and collinear limits. One adds and subtracts a suitable approximation term $\mathcal{D}_n^S$, such that the combination $\mathcal{R}_n - \mathcal{D}_n^S$ is finite in the $(n+1)$ -particle phase space, and the analytically integrated subtraction term cancels the IR poles of the virtual correction. Inserting these subtraction terms into equation 2.47 yields

$$\begin{aligned} \hat{\sigma}_{a,b \rightarrow n}^{\text{NLO}} &= \int_n d\tilde{\Phi}_n \mathcal{B}_n(\Phi_n; \mu_F, \mu_R) + \int_n d\tilde{\Phi}_n \left[\mathcal{V}_n(\Phi_n; \mu_F, \mu_R) + \mathcal{I}_n^S(\Phi_n; \mu_F, \mu_R) \right] \\ &+ \int_{n+1} d\tilde{\Phi}_{n+1} \left[\mathcal{R}_n(\Phi_{n+1}; \mu_F, \mu_R) - \mathcal{D}_n^S(\Phi_n; \mu_F, \mu_R) \right], \end{aligned} \quad (2.51)$$

where the integrated subtraction term is defined as

$$\mathcal{I}_n^S(\Phi_n; \mu_F, \mu_R) = \int_1 d\Phi_1 \mathcal{D}_n^S(\Phi_n; \mu_F, \mu_R). \quad (2.52)$$

The process-independent construction of the subtraction terms was derived in the dipole formalism by Catani and Seymour for massless partons [44], and extended to massive partons by Catani, Dittmaier, Seymour and Trócsányi [36]. In this formalism, the real subtraction term is constructed according to the dipole factorization formula

$$\mathcal{D}_n^S = \sum_{\text{dipoles}} \mathcal{B}_n \otimes dV_{\text{dipole}}, \quad (2.53)$$

where the dipole splitting functions dV_{dipole} are universal and do not depend on the specific hard scattering process. The symbol $\otimes$ represents a convolution over phase space, including sums over colour and spin degrees of freedom.

The integrated subtraction term can be written schematically as

$$\int_{n+1} d\tilde{\Phi}_{n+1} \mathcal{D}_n^S = \sum_{\text{dipoles}} \int_n d\tilde{\Phi}_n \mathcal{B}_n \otimes \int_1 d\Phi_1 dV_{\text{dipole}} = \int_n d\tilde{\Phi}_n [\mathcal{B}_n \otimes I_n], \quad (2.54)$$

where the integration over the unresolved one-particle phase space is performed analytically. The explicit expressions for dV_{dipole} and its integrated form I_n are given in [36, 44].

Ultraviolet Divergences and Regularization

In addition to IR singularities, virtual loop corrections contain ultraviolet (UV) divergences arising from large loop momenta [24]. These divergences are treated using dimensional regularization [45], in which the space time dimension is analytically continued from $D = 4$ to $D = 4 - 2\epsilon$. Divergent loop integrals then manifest themselves as poles in $\frac{1}{\epsilon}$. UV divergences are absorbed through renormalization. As a consequence, the renormalized quantities depend on the previously introduced renormalization scale μ_R . Through dimensional regularization, IR divergences also appear as poles in $\frac{1}{\epsilon}$ and $\frac{1}{\epsilon^2}$. After renormalization of UV divergences and cancellation of the IR poles, the remaining contributions are finite and suitable for numerical phase space integration.

2.3 SHERPA

In the previous section, NLO corrections were discussed as an essential component of precise theoretical predictions. However, a purely partonic description is not sufficient for direct comparison with experimental measurements. Moreover, the complexity of realistic scattering processes makes it impossible to perform all relevant calculations fully analytically. Monte Carlo event generators bridge this gap by translating theoretical predictions into realistic event simulations that can be directly compared with experimental data. They combine perturbative calculations of hard scattering processes with parton shower algorithms, as well as

non-perturbative models such as hadronization and multi parton interactions. This enables the simulation of complete events. Several Monte Carlo tools are widely used in particle physics, including Herwig [19], POWHEG [46], and SHERPA [1]. The latter is used for this thesis.

Monte Carlo generators are structured in a modular way. The individual steps of event generation, such as hard collisions and decays, parton shower evolution and hadronization, are treated as conceptually separate components. This factorized structure allows each stage to be implemented and improved independently. This section provides a brief overview of the structure of a simulated event within SHERPA and mainly follows refs. [37, 47–51]. Parton-shower matching is briefly discussed to provide context and an outlook on possible extensions. In addition, the NWA is discussed in further detail.

2.3.1 Event Simulation in SHERPA

Figure 2.4 shows a schematic overview of a typical event generated with SHERPA. The different subprocesses operate at distinct energy scales, ranging from the hard collision down to softer emissions.

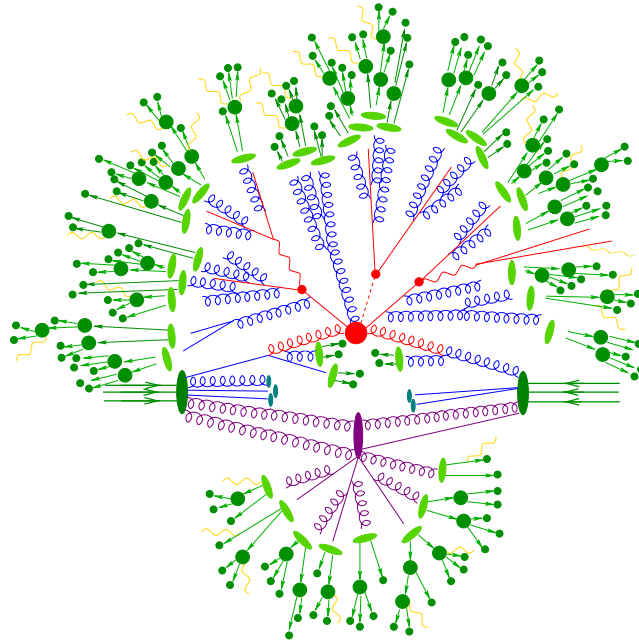


Figure 2.4: Overview of a exemplary SHERPA event of a proton-proton collision with an intermediate Higgs boson and top quark production. The colours represent: Red: hard collision (big blob), hard decays (small blobs); Blue: QCD radiation; Light green: hadronization; Dark green: hadron decays; Yellow: QED radiation; Purple: underlying event [1].

Hard Collision and Decay The event generation starts with the hard scattering process, typically involving protons or electrons and positrons in the initial state. The term hard refers to processes occurring at large momentum transfers, typically well above $\mathcal{O}(10 \text{ GeV})$.

At these scales, perturbation theory is applicable and the interaction can be described in terms of Feynman diagrams. The hard scattering produces high-energy particles, which may subsequently undergo perturbative decays, referred to as hard decays. The fixed-order NLO calculation discussed in the previous section applies to this stage of the event generation.

Parton Shower

Highly energetic coloured partons produced in the hard process undergo further QCD radiation, which is modelled by the parton shower as a cascade of predominantly soft and collinear emissions. Its purpose is to approximate the effect of higher-order corrections beyond fixed-order perturbation theory by resumming logarithmically enhanced contributions to all orders. The shower starts from the outgoing partons generated in the hard scattering or decay and simulates successive emissions using the Sudakov-Veto algorithm [52]. The evolution proceeds from a high starting scale, typically set by the hard process, down to a non-perturbative energy scale of order $\mathcal{O}(1 \text{ GeV})$, where perturbative QCD is no longer applicable and colour neutral hadrons form. The parton shower is largely process-independent. Its dynamics are governed by universal QCD splitting functions, which depend only on the flavour, colour, and kinematics of the emitting parton. Through successive branchings, the shower generates a multi-parton configuration that approximates multiple real emissions and provides a realistic description of QCD radiation prior to hadronization. The probability for emissions, and in particular for no emission, between two evolution scales is encoded in the Sudakov form factor. This factor expresses the non-splitting probability between two scales t and t' which are typically related to the transverse momentum or emission angle. Schematically, it can be written as [53]

$$\Delta(t, t') = \exp \left\{ - \int_t^{t'} d\Phi_1 \frac{\alpha_S}{2\pi} K(\Phi_1) \right\}, \quad (2.55)$$

where $K(\Phi_1)$ denotes the appropriate splitting kernel that encodes the kinematic distribution, and $d\Phi_1$ the phase space element of the emitted parton.

Parton Shower Matching

At LO, matching the parton shower to the hard matrix element is straightforward: the shower can simply be initiated from the external partons of the Born configuration. Since the LO calculation does not contain explicit higher-order radiation, no double counting occurs. At NLO, the situation becomes more complicated. Both the fixed-order NLO calculation and the parton shower describe additional QCD radiation. A naive combination would therefore lead to double counting of emissions in certain regions of phase space. To avoid this, a consistent matching procedure is required. In SHERPA, this is achieved using the MC@NLO algorithm [15]. An alternative widely used matching scheme is POWHEG [16, 17]. The key idea of MC@NLO is to subtract from the NLO calculation the part that would be generated by the parton shower. This ensures that the first emission is described exactly by the NLO matrix

element, while subsequent emissions are generated by the shower without double counting. For this, the real-emission contribution is split into an IR-singular part $\mathcal{D}_n^A$ and a finite remainder $\mathcal{H}_n^A$, such that $\mathcal{R}_n = \mathcal{D}_n^A + \mathcal{H}_n^A$. Using this decomposition, the NLO cross section from equation 2.51 can be rewritten as [54]

$$\begin{aligned} \sigma^{\text{NLO}} = & \int_n d\tilde{\Phi}_n [\mathcal{B}_n(\Phi_n) + \mathcal{V}_n(\Phi_n) + \mathcal{I}_n^S(\Phi_n)] \\ & + \int_{n+1} d\tilde{\Phi}_{n+1} [\mathcal{D}_n^A(\Phi_{n+1}) - \mathcal{D}_n^S(\Phi_{n+1})] + \int_{n+1} d\tilde{\Phi}_{n+1} \mathcal{H}_n^A(\Phi_{n+1}). \end{aligned} \quad (2.56)$$

Combining the NLO result with a parton shower that produces at most one additional emission, the matched cross section

$$\sigma_{\text{MC@NLO}}^{\text{NLO}} = \int d\tilde{\Phi}_n \bar{B}_n(\Phi_n) \left[\bar{\Delta}(t_0) + \int_{t_0} d\tilde{\Phi}_1 \frac{\mathcal{D}_n^A(\tilde{\Phi}_{n+1})}{\mathcal{B}_n(\tilde{\Phi}_n)} \bar{\Delta}^A(t) \right] + \int_{n+1} d\tilde{\Phi}_{n+1} \mathcal{H}_n^A(\Phi_{n+1}) \quad (2.57)$$

is obtained, with the weighted Born cross section

$$\bar{B}_n(\Phi_n) = \mathcal{B}(\Phi_n) + \mathcal{V}(\Phi_n) + \mathcal{I}^S(\Phi_n) + \int_1 d\Phi_{n+1} [\mathcal{D}^A(\Phi_{n+1}) - \mathcal{D}^S(\Phi_{n+1})]. \quad (2.58)$$

The second term in the matching formula describes events generated directly from the hard remainder $\mathcal{H}_n^A$, which corresponds to real emissions not captured by the shower approximation. In SHERPA, the subtraction terms are chosen such that $\mathcal{D}_n^A(\Phi_{n+1}) = \mathcal{D}_n^S(\Phi_{n+1})$ holds [55], which simplifies equation 2.57 accordingly. The no-emission probability in MC@NLO is described by a modified Sudakov form factor,

$$\bar{\Delta}(t, t') = \exp \left\{ - \int_t^{t'} d\Phi_1 \frac{\mathcal{D}_n^A(\tilde{\Phi}_{n+1})}{\mathcal{B}_n(\tilde{\Phi}_n)} \right\}. \quad (2.59)$$

In addition to matching at fixed multiplicity, it is often necessary to consistently combine matrix elements with different final-state parton multiplicities. This procedure is known as multijet merging. A detailed discussion of multijet merging is beyond the scope of this thesis. For further information, see ref. [56].

Hadronization

At energy scales of order $\mathcal{O}(1 \text{ GeV})$, the strong coupling constant becomes large and perturbative QCD is no longer applicable. In this non-perturbative regime, coloured partons cannot exist as asymptotically free states due to colour confinement. Instead, they combine into colour-neutral bound states such as mesons and baryons. Two phenomenological approaches are commonly used to model this transition from partons to hadrons: the string model [57] and the cluster model [58]. SHERPA uses the latter. In this model, nearby colour-anticolour pairs from gluons or quarks are formed into colour-neutral clusters, which then decay into hadrons.

After the hadronization, unstable hadrons decay further. In event simulations, particles are typically classified as stable or unstable based on whether they decay within the detector volume. Particles with lifetimes long compared to detector scales, such as muons, are treated as stable for experimental purposes, whereas particles such as tau leptons or most hadrons decay before detection.

Underlying Event

In proton–proton collisions, only a subset of the partons participate in the primary hard scattering process. The remaining partons in the incoming hadrons can undergo additional interactions. These secondary interactions contribute to the overall activity observed in the detector. These are referred to as the underlying event.

Finally, the QED shower is implemented analogously to the QCD shower, often employing the same evolution algorithm.

2.3.2 Narrow Width Approximation

When generating collision events, processes are selected according to their total production probability. This becomes inefficient when studying rare processes involving specific intermediate resonances, such as Higgs bosons. An impractically large number of events would have to be generated to obtain sufficient statistics for such channels. To enable the targeted generation of processes containing specific resonant particles, SHERPA employs the NWA [59]. In this approach, production and decay of an intermediate unstable particle are factorized and treated separately. The event generation is starting from the hard subprocess involving the chosen resonance, and the full event is reconstructed both forward (decay) and backward (initial-state radiation and beam evolution). This allows a targeted sampling of rare resonant processes and simplifies the calculation, since higher order corrections can be treated separately in the production and decay part.

The narrow width approximation can be applied when a process involves a resonant particle with mass M and decay width Γ , provided that [60]

$$\Gamma \ll M. \quad (2.60)$$

In this regime, non-resonant contributions and interference terms are suppressed and may be neglected to first approximation. The theoretical uncertainty of the NWA is of order $\frac{\Gamma}{M}$. The full matrix element for a resonant process can generically be written as

$$\mathcal{M} \propto \mathcal{M}_{prod} \frac{i}{q^2 - M^2 + iM\Gamma} \mathcal{M}_{decay}, \quad (2.61)$$

where $\mathcal{M}_{prod}$ describes the production subprocess, $\mathcal{M}_{dec}$ the decay of the resonance and q

is the four-momentum of the intermediate particle. Squaring the propagator leads to the Breit–Wigner distribution of an unstable particle [61],

$$\frac{1}{(q^2 - M^2)^2 + M^2\Gamma^2}. \quad (2.62)$$

In the narrow-width limit $\Gamma \rightarrow 0$, this expression can be approximated as [59]

$$\frac{1}{(q^2 - M^2)^2 + M^2\Gamma^2} \rightarrow \frac{\pi}{M\Gamma} \delta(q^2 - M^2), \quad (2.63)$$

which enforces the on-shell condition $q^2 = M^2$. As a result, the phase space integration factorizes into a production part and a decay part, and the total cross section reduces to [62]

$$\sigma \approx \sigma_{prod} \cdot BR. \quad (2.64)$$

This factorization considerably simplifies both analytical calculations and event generation.

The discussion of the NWA presented so far is sufficient when considering resonances of massive scalar particles. For particles with non-zero spin, however, an additional non-trivial numerator appears in equation 2.61, which complicates the factorization of production and decay. This will be discussed in the following.

The numerator of a massive gauge boson propagator is given as [20]

$$i(-g^{\mu\nu} + \frac{q^\mu q^\nu}{M^2}). \quad (2.65)$$

Using the completeness relation for polarization vectors, this tensor structure can be expressed as a sum over polarization states [63],

$$(-g^{\mu\nu} + \frac{q^\mu q^\nu}{M^2}) = \sum_{\lambda=1}^3 \epsilon_\lambda^\mu(q) \epsilon_\lambda^{*\nu}(q), \quad (2.66)$$

where $\epsilon_\lambda^\mu(q)$ are the polarization vectors and the sum runs over the physical polarization states of the massive vector boson (two transverse and one longitudinal mode). The helicity λ of a particle is defined as the projection of its spin onto its direction of motion,

$$\lambda = \frac{\vec{S} \cdot \vec{p}}{|\vec{p}|}, \quad (2.67)$$

where $\vec{p}$ denotes the particle's momentum. Fermions possess two helicity states, $\lambda = \pm \frac{1}{2}$. Massless spin-1 bosons have two physical helicity $\lambda = \pm 1$ (transverse polarizations), and massive spin-1 bosons have an additional longitudinal polarization state $\lambda = 0$.

Similarly, for fermion propagators with numerator $i(\not{q} + M)$, the spinor completeness relation

reads

$$(\not{q} + M) = \sum_{\lambda=\pm\frac{1}{2}} u_{\lambda}(q) \bar{u}_{\lambda}(q), \quad (2.68)$$

with Dirac spinors $u_{\lambda}(q)$.

Inserting equations 2.66 and 2.68 into equation 2.61 allows the amplitude to be decomposed into helicity components of the intermediate particle. The full matrix element can then be written as

$$\mathcal{M} = \frac{i}{q^2 - M^2 + iM\Gamma} \sum_{\lambda} \mathcal{M}_{\lambda,prod} \mathcal{M}_{\lambda,decay}. \quad (2.69)$$

Here, the helicity-dependent production amplitudes are defined as

$$\mathcal{M}_{\lambda,prod}^{boson} = \mathcal{M}_{prod}^{boson} \cdot \epsilon_{\lambda}^{\mu}(q), \quad \mathcal{M}_{\lambda,prod}^{fermion} = \mathcal{M}_{prod}^{fermion} \cdot u_{\lambda}(q) \quad (2.70)$$

for intermediate massive vector bosons and fermions, respectively. The decay amplitudes are constructed analogously using the conjugated polarization vectors or Dirac spinors, i.e. $\epsilon_{\lambda}^{*\nu}(q)$ or $\bar{u}_{\lambda}(q)$. Since the amplitude is now expressed as a sum over helicities, production and decay can no longer be treated independently: both stages are correlated through the helicity index λ of the intermediate particle. In order to consistently preserve these correlations in SHERPA, the spin-correlation algorithm by Richardson [64] is employed when intermediate massive gauge bosons or fermions are present. For massless particles, the NWA is in general not applicable, since it relies on the presence of a resonant propagator with a finite width, which is absent in the strictly massless case.

In the Richardson spin-correlation algorithm, the polarization state of the intermediate particle is encoded in a spin-density matrix. The production process defines an initial spin-density matrix, which is subsequently contracted with the decay matrix of the next stage in the decay chain. The resulting matrix is then propagated further down the cascade. The decay matrix is constructed from the helicity amplitudes of the decay process and contains both the squared terms and the interference terms between different helicity configurations. For helicity amplitudes $\mathcal{M}_{\lambda}$ corresponding to fixed helicities λ of the intermediate particle, the decay matrix D is defined as

$$D_{\lambda\lambda'} = \frac{1}{N_D} \mathcal{M}_{\lambda} \mathcal{M}_{\lambda'}^*, \quad (2.71)$$

where λ and λ' denote helicities of the intermediate state, and N_D is a normalization factor such that D has a trace of 1. The diagonal elements of the decay matrix contain the squared helicity amplitudes, whereas the off-diagonal elements represent the corresponding interference terms.

This procedure guarantees that spin-correlations between the production of heavy resonances and their subsequent decays are systematically retained throughout the event generation. For details on this algorithm, see ref. [63, 64].

3 NLO Calculation and Implementation

The aim of this work is to incorporate fixed-order NLO QCD corrections into the hard decay framework of SHERPA, which previously provided decay calculations only at LO. Due to the use of the NWA, these decay corrections can be treated independently of the production process. The implementation was carried out for the decay of a Higgs boson into a bottom–antibottom quark pair. With a branching ratio of approximately 53% [3], this channel constitutes the dominant Higgs decay mode. In addition to the concrete implementation, a reusable and modular structure was developed that allows further NLO decay processes to be implemented with little additional effort. The decay of a scalar Higgs boson represents one of the simplest cases within the hard decay framework, since no spin correlations need to be taken into account. Nevertheless, the implementation was designed in a general manner such that processes involving non-trivial spin correlations, for example decays of vector bosons, can also be treated consistently. However, a complete validation using a NLO decay process with non-trivial spin structure is still pending and may require further refinements. Wherever possible, analytical expressions were evaluated in advance. This is particularly advantageous in the calculation of virtual corrections, which involve numerous matrix multiplications that would otherwise be computationally inefficient in a purely numerical implementation.

This chapter is organized as follows: First, the analytical preparations are presented. Afterwards, the existing SHERPA hard decay framework is described. Finally, the modifications and extensions implemented in the software are explained.

3.1 Preliminary Calculations

Extending a partonic cross section from LO to NLO requires the implementation of real ($\mathcal{R}$), virtual ($\mathcal{V}$) and corresponding subtraction contributions ($\mathcal{D}^S$, $\mathcal{I}^S$), as outlined in section 2.2.2. The real-emission correction terms were calculated entirely within SHERPA, and no separate analytical pre-computation was required for this. It should be noted that the calculation of the virtual correction and the virtual subtraction term is not yet entirely correct, as the results deviate from the expected results obtained by COMIX, which is a different module in SHERPA and used as reference. This will be further discussed in chapter 4. The calculations are nevertheless shown here so that the current state of the implementation is fully documented

and can serve as a starting point for future work.

3.1.1 Real Subtraction $\mathcal{D}^S$

The subtraction terms were derived following the formalism presented in [36], which explicitly covers the case of a decay into massive quarks. In the present calculation, the bottom quarks must be treated as massive, since the Yukawa coupling of the Higgs boson is proportional to the quark mass. In the massless limit $m_b \rightarrow 0$, the Yukawa coupling would vanish and the decay $H \rightarrow b\bar{b}$ would not occur.

One can rewrite formula 2.53 as

$$\mathcal{D}_n^S = \sum_{\text{dipoles}} \mathcal{D}_{ij,k} = \sum_{\text{dipoles}} \mathcal{B}_n \otimes dV_{ij,k}. \quad (3.1)$$

Here, $V_{ij,k}$ only depends on the momenta and quantum numbers of the three physical final state partons i, j and k . In the dipole formalism, the two final state particles involved in the emission are combined into a single emitter parton $\tilde{i}\tilde{j}$. The third parton is referred to as the spectator parton $\tilde{k}$. While the emitter produces the additional emitted parton, the spectator absorbs the recoil in order to ensure momentum conservation. For the real-emission process $H \rightarrow b\bar{b}g$, two dipole terms arise:

$$\mathcal{D}_n^S = \mathcal{D}_{qg,\bar{q}} + \mathcal{D}_{\bar{q}g,q}. \quad (3.2)$$

In the first term, the quark emits the gluon and the antiquark acts as the spectator. In the second term, the roles are interchanged. The explicit form of the dipole term depends on whether emitter and spectator belong to the initial or final state. In the present case, both partons are in the final state. A graphical representation is shown in figure 3.1.

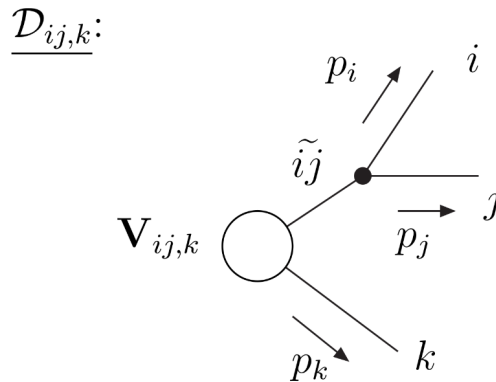


Figure 3.1: Graphical representation of the dipole term $\mathcal{D}_{ij,k}$ for final state partons. The partons i and j are emitted from the emitter parton $\tilde{i}\tilde{j}$. Figure taken from [36].

The dipole contribution for a final state configuration reads

$$\mathcal{D}_{ij,k}(p_1, \dots, p_{n+1}) = -\frac{1}{(p_i + p_j)^2 - m_{ij}^2} {}_n\langle \dots, \tilde{i}j, \dots, \tilde{k}, \dots | \frac{T_k \cdot T_{ij}}{T_{ij}^2} V_{ij,k} | \dots, \tilde{i}j, \dots, \tilde{k}, \dots \rangle_n. \quad (3.3)$$

The reduced n-particle states correspond to the Born matrix element:

$${}_n\langle \dots, \tilde{i}j, \dots, \tilde{k}, \dots | = \langle \mathcal{M}_B |, \quad | \dots, \tilde{i}j, \dots, \tilde{k}, \dots \rangle_n = | \mathcal{M}_B \rangle. \quad (3.4)$$

Here, p_i and p_j denote the momenta of the physical partons i and j . The mass of the emitter is denoted by m_{ij} , which in this case corresponds to the quark mass, as the gluon is massless. The operators T_k and T_{ij} are the colour operators associated with the spectator and emitter, respectively. The function $V_{ij,k}$ contains the singular behaviour of the real-emission matrix element and depends only on the momenta and quantum numbers of partons i, j and k .

It is important to distinguish between the physical spectator k and the mapped spectator $\tilde{k}$. Both share identical quantum numbers, but differ in their kinematics. The same holds true for the emitter system $\tilde{i}j$, whose kinematics differ from the physical partons appearing in the real-emission matrix element. The kinematics can be calculated from the momenta of the physical partons. The mapped momenta are given by

$$\tilde{p}_k^\mu = \frac{\sqrt{\lambda(Q^2, m_{ij}^2, m_k^2)}}{\sqrt{\lambda(Q^2, (p_i + p_j)^2, m_k^2)}} \left(p_k^\mu - \frac{Q p_k}{Q^2} Q^\mu \right) + \frac{Q^2 + m_k^2 - m_{ij}^2}{2Q^2} Q^\mu \quad (3.5)$$

and

$$\tilde{p}_{ij}^\mu = Q^\mu - \tilde{p}_k^\mu, \quad (3.6)$$

with total outgoing momentum $Q^\mu = p_i^\mu + p_j^\mu + p_k^\mu = \tilde{p}_{ij}^\mu + \tilde{p}_k^\mu$. The Källén function is defined as

$$\lambda(x, y, z) = x^2 + y^2 + z^2 - 2xy - 2xz - 2yz. \quad (3.7)$$

This mapping preserves total momentum and on-shell conditions. For the specific case of the two-body decay $H \rightarrow b\bar{b}$, the squared Born matrix element depends only on $Q^2 = m_H^2$ and the bottom-quark mass. Since the mapping preserves these invariants, the value of $|\mathcal{M}_B|^2$ remains unchanged.

For two coloured partons, colour conservation implies [44]

$$T_1 \cdot T_2 |1, 2\rangle = -T_1^2 |1, 2\rangle = -T_2^2 |1, 2\rangle. \quad (3.8)$$

Using this identity, the colour-correlated Born term simplifies to:

$$\langle \mathcal{M}_{\mathcal{B}} | \frac{T_k \cdot T_{ij}}{T_{ij}^2} | \mathcal{M}_{\mathcal{B}} \rangle = \langle \mathcal{M}_{\mathcal{B}} | \frac{-T_{ij}^2}{T_{ij}^2} | \mathcal{M}_{\mathcal{B}} \rangle = -\langle \mathcal{M}_{\mathcal{B}} | \mathcal{M}_{\mathcal{B}} \rangle. \quad (3.9)$$

The dipole term therefore reduces to

$$\mathcal{D}_{ij,k}(p_1, \dots, p_{n+1}) = + \frac{1}{(p_i + p_j)^2 - m_{ij}^2} \langle \mathcal{M}_{\mathcal{B}} | V_{ij,k} | \mathcal{M}_{\mathcal{B}} \rangle. \quad (3.10)$$

For an emitter composed of a gluon and a quark, the dipole function reads

$$\begin{aligned} \langle s | V_{gq,k} | s' \rangle &= 8\pi\mu^{2\epsilon}\alpha_S C_F \left\{ \frac{2}{1 - \tilde{z}_j(1 - y_{ij,k})} - \frac{\tilde{v}_{ij,k}}{v_{ij,k}} \left[1 + \tilde{z}_j + \frac{m_q^2}{p_i p_j} + \epsilon(1 - \tilde{z}_j) \right] \right\} \delta_{ss'} \\ &= \langle V_{gq,k} \rangle \delta_{ss'}. \end{aligned} \quad (3.11)$$

Here, $C_F = \frac{4}{3}$ is the eigenvalue of the quadratic Casimir operator and μ denotes the renormalization scale. s and s' are the spin indices of the emitter in $\langle \mathcal{M}_{\mathcal{B}} |$ resp. $| \mathcal{M}_{\mathcal{B}} \rangle$. The rescaled masses are

$$\mu_n = \frac{m_n}{\sqrt{Q^2}}. \quad (3.12)$$

Further, the kinematic variables $\tilde{z}_i$, $\tilde{z}_j$ and the dimensionless variable $y_{ij,k}$ are defined as

$$\tilde{z}_i = 1 - \tilde{z}_j = \frac{p_i p_k}{p_i p_k + p_j p_k}, \quad y_{ij,k} = \frac{p_i p_j}{p_i p_j + p_i p_k + p_j p_k}. \quad (3.13)$$

Finally, there are the relative velocities $\tilde{v}_{ij,k}$ between the momenta $\tilde{p}_{ij}$ and $\tilde{p}_k$, and $v_{ij,k}$ between p_{ij} and p_k :

$$\tilde{v}_{ij,k} = \frac{\sqrt{\lambda(1, \mu_{ij}^2, \mu_k^2)}}{1 - \mu_{ij}^2 - \mu_k^2}, \quad v_{ij,k} = \frac{\sqrt{[2\mu_k^2 + (1 - \mu_i^2 - \mu_j^2 - \mu_k^2)(1 - y_{ij,k})]^2 - 4\mu_k^2}}{(1 - \mu_i^2 - \mu_j^2 - \mu_k^2)(1 - y_{ij,k})}. \quad (3.14)$$

The implementation of these expressions within the SHERPA framework will be discussed in section 3.3.

3.1.2 Colour Treatment in the $\mathcal{R}$ and $\mathcal{D}^S$ Contributions

In analytical calculations of Feynman diagrams, the squared matrix element is typically summed explicitly over all possible colour configurations of the external partons. When using the matrix element generator COMIX [2] within SHERPA, this colour sum is not carried out explicitly. Instead, COMIX employs Monte Carlo sampling over colour configurations. In this approach, a specific colour assignment for the external partons is selected, and the matrix element is

evaluated for this fixed colour configuration. The result is then multiplied by the number of all possible colour configurations. The full colour-summed result is obtained statistically by averaging over many such randomly sampled configurations. This procedure replaces the explicit summation over colour degrees of freedom by a stochastic estimator, which converges to the exact colour-summed result in the limit of infinite statistics.

Analytically, the colour structure is usually evaluated using the colour operators introduced in the previous chapter, which act in the $SU(3)_C$ algebra and lead to traces over colour matrices when squaring the amplitude. In Comix, however, colour is treated in the colour-flow representation. The colour-flow representation expresses the colour structure in terms of explicit colour connections between partons. This reformulation allows the colour algebra to be handled without matrix multiplications, which reduces the computational complexity and improves performance. All coloured particles are assigned colour indices that take the values 1, 2, or 3. A colour configuration is obtained by assigning colour and/or anti-colour indices to all coloured external particles in such a way that colour is conserved at each interaction vertex. In particular,

- quarks carry a colour index,
- antiquarks carry an anti-colour index,
- gluons carry both a colour and an anti-colour index.

Different valid colour configurations can generally lead to different numerical colour factors in the matrix element. For the real-emission contribution, a random but valid colour configuration is selected in each Monte Carlo iteration and the matrix element is evaluated for this fixed configuration. In order to ensure the pointwise cancellation of IR divergences, the real subtraction term must be evaluated using the same colour configuration in each iteration. This is crucial because different colour representations, such as singlet and octet configurations, carry different colour factors and therefore contribute differently to the squared matrix element. Only if both the real-emission matrix element and its corresponding subtraction term are evaluated for the identical colour configuration, the infrared cancellation occurs at the integrand level. Figure 3.2 shows three exemplary colour configurations for one of the real-emission diagrams of the $H \rightarrow q\bar{q}$ decay. Each configuration leads to a different colour factor and therefore contributes differently to the squared matrix element.

In the expressions for the real subtraction term introduced in the previous section, colour degrees of freedom were treated in the standard way using colour operators. For a Monte Carlo implementation in the colour-flow basis, this representation must be rewritten accordingly. For the Born matrix element, both representations yield the same result. Since the Higgs boson is colourless, the quark–antiquark pair produced in the decay forms a colour singlet. The colour dependence therefore reduces to a trivial Kronecker delta, and no non-trivial colour algebra

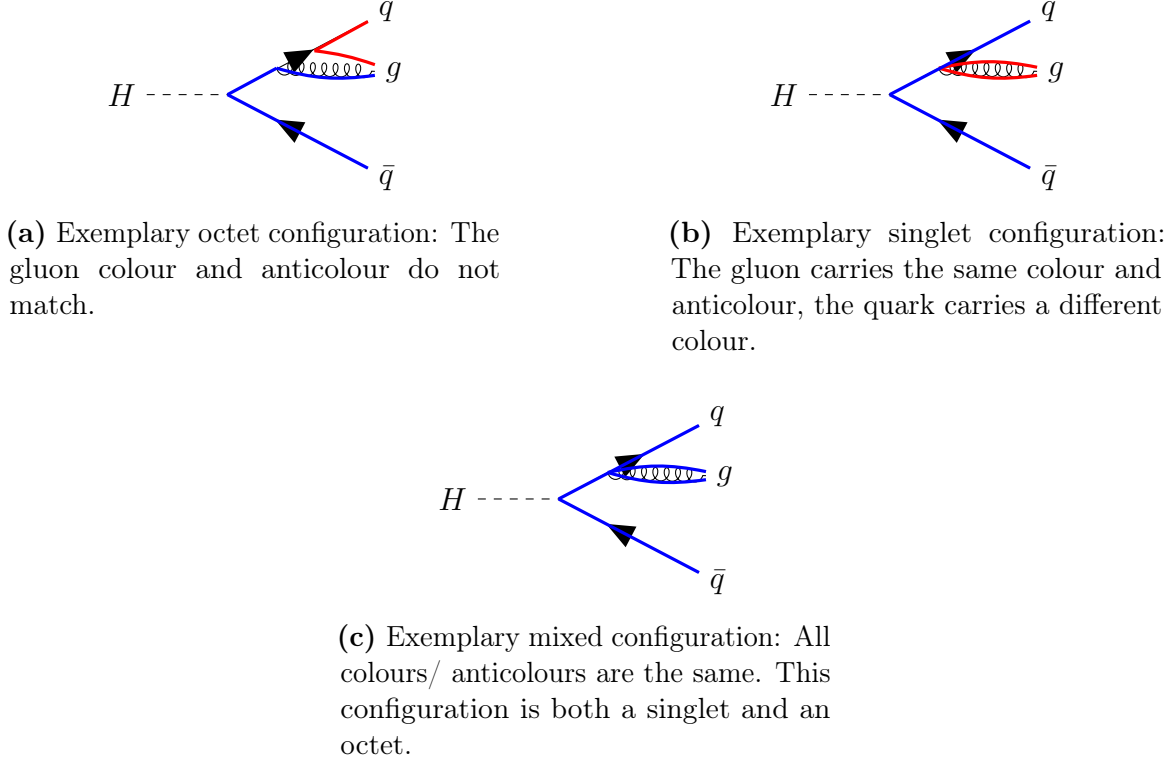


Figure 3.2: Possible colour configurations for the real Feynman diagram where the gluon is emitted from the quark ($n = 1$). The coloured lines represent the colour-flows in the diagram. Blue and red depict different colour/ anticolour indices assigned to the corresponding particles.

appears at Born level.

The colour structure of the real-emission amplitude in the colour-flow basis can be constructed from the underlying Born amplitude using the relations given in [65]. We denote the colour component of the Born amplitude with colour index i_1 and anti-colour index j_2 by $\mathcal{A}_{j_2}^{i_1}$. When a gluon is emitted from one of the coloured external particles ($n = 1, 2$ corresponding to emission from the quark or antiquark line), the colour indices of the emitter are modified. This is implemented by replacing the emitter indices according to $i_n \rightarrow i'_n$; $j_n \rightarrow j'_n$ in the Born colour component. The colour component of the real-emission amplitude, carrying gluon colour-flow indices (i, j), is then obtained by acting with a tensor K on the modified Born colour component,

$$\mathcal{A}_{j_2 j}^{i_1 i}(n) = \sum_{i'_n, j'_n} K_{j_n j' i'_n}^{i_n i j'_n} \mathcal{A}_{j'_2}^{i'_1} \Big|_{i_n \rightarrow i'_n; j_n \rightarrow j'_n}. \quad (3.15)$$

The tensor K encodes the insertion of the gluon in the colour-flow basis and is given by

$$K_{j j_n i'_n}^{i i_n j'_n} = \begin{cases} \delta_j^{i_n} \delta_{i'_n}^i - \frac{1}{N_c} \delta_{i'_n}^{i_n} \delta_j^i & \text{for the } n\text{-th outgoing parton being a quark} \\ -\delta_j^{j'_n} \delta_{j_n}^i + \frac{1}{N_c} \delta_{j_n}^{j'_n} \delta_j^i & \text{for the } n\text{-th outgoing parton being an antiquark,} \end{cases} \quad (3.16)$$

with $N_c = 3$ the number of colours. The factors $K_{j j_n i'_n}^{i i_n j'_n}$ can be evaluated explicitly for the three representative colour configurations shown in figure 3.2. The resulting values are listed in table 3.1. They determine how each colour configuration contributes to the real-emission amplitude. When squaring the amplitude, these coefficients lead to additional colour factors of

	Octet	Singlet	Mix
$n = 1$	1	$-\frac{1}{3}$	$\frac{2}{3}$
$n = 2$	-1	$\frac{1}{3}$	$-\frac{2}{3}$

Table 3.1: $K_{j j_n i'_n}^{i i_n j'_n}$ factors for the singlet, octet and mix configurations for both real diagrams $n = 1, 2$.

1 for octet configurations, $\frac{1}{9}$ for singlet configurations, and $\frac{4}{9}$ for mixed configurations. These values arise from the squared coefficients of the corresponding K tensor components. Since the real subtraction dipole terms defined in equation 3.3 are formulated in the colour-operator formalism, their colour structure must be projected onto the explicit colour-flow configurations used in the Monte Carlo implementation. The factors derived from K provide precisely this projection. One must therefore multiply the subtraction terms with these $(K_{j j_n i'_n}^{i i_n j'_n})^2$ factors in order to reproduce the correct colour correlations and ensure the local cancellation of IR singularities.

3.1.3 Virtual Correction $\mathcal{V}$

As discussed in section 2.2, the virtual contribution to the NLO cross section is given by the interference between the Born amplitude and the one-loop amplitude $\mathcal{V} = 2 \cdot \Re\{\mathcal{M}_B \mathcal{M}_V^*\}$. The main technical challenge lies in the evaluation of the one-loop matrix element $\mathcal{M}_V$.

In SHERPA, all external momenta are treated as outgoing. Therefore, for incoming particles, the momenta obtain an additional sign of -1. Figure 3.3 shows the one-loop Feynman diagram for the process $H \rightarrow b\bar{b}$ with the momenta assigned according to this rule.

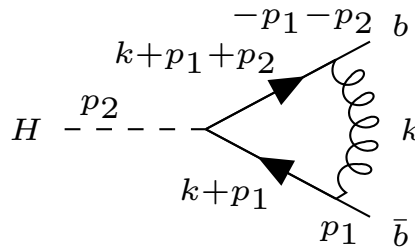


Figure 3.3: Virtual $H \rightarrow b\bar{b}$ Feynman diagram with all momenta labelled as outgoing.

Applying the Feynman rules [20], the virtual amplitude reads

$$\begin{aligned}
\mathcal{M}_V &= \sum_a \int \frac{d^n k}{(2\pi)^n} \bar{u}(-p_1 - p_2) \cdot (-ig_S(T^a)\gamma^\mu\mu^\epsilon) \cdot i \frac{\not{k} + \not{p}_1 + \not{p}_2 + m_b}{(k + p_1 + p_2)^2 - m_b^2} \\
&\quad \cdot \left(-i \frac{m_b}{\nu}\right) \cdot i \frac{\not{k} + \not{p}_1 + m_b}{(k + p_1)^2 - m_b^2} \cdot (-ig_S(T^a)\gamma^\rho\mu^\epsilon) \cdot v(p_1) \cdot \left(-ig_{\rho\mu} \frac{1}{k^2}\right) \\
&= \sum_a (-1)g_S^2 \cdot \frac{m_b}{\nu} \cdot (T^a)(T^a) \cdot \bar{u}(-p_1 - p_2)\gamma^\mu \\
&\quad \cdot \int \frac{d^n k}{(2\pi)^n} \frac{\mu^{2\epsilon}[\not{k} + \not{p}_1 + \not{p}_2 + m_b][\not{k} + \not{p}_1 + m_b]}{k^2[(k + p_1)^2 - m_b^2][(k + p_1 + p_2)^2 - m_b^2]} \gamma_\mu v(p_1), \tag{3.17}
\end{aligned}$$

where the integral dimension n is set to $n = 4 - 2\epsilon$ to apply dimensional regularization. ν denotes the vacuum expectation value, $g_s = \sqrt{4 \cdot \pi \cdot \alpha_S}$ is the strong coupling, T^a are the colour operators, $\bar{u}(-p_1 - p_2)$ the spinor of the bottom quark and $v(p_1)$ the spinor of the antibottom quark. The colour structure results in $T^a T^a = C_F = 4/3$. The structure of the above equation can be summarized as follows:

$$\mathcal{M}_V = (\text{constants}) \cdot \bar{u}(-p_1 - p_2) \cdot \gamma^\mu \cdot (\text{loop integral})(p_1, p_2) \cdot \gamma_\mu \cdot v(p_1). \tag{3.18}$$

The virtual matrix element was computed using the following procedure: The loop integral is reduced to scalar integrals using the Passarino–Veltman tensor reduction method [66]. The resulting scalar integrals are then evaluated with the library LoopTools [67]. After tensor reduction, the structure $\gamma^\mu \cdot (\text{loop integral}) \cdot \gamma_\mu$ is evaluated in a Python [68] script. The script produces a momentum-dependent 4×4 matrix, which is then inserted into SHERPA. The remaining multiplications with the constants and Dirac Spinors are then performed in SHERPA.

Calculation of the Loop Integral

The evaluation of the loop integral is not trivial because of UV and IR divergences. In dimensional regularization with $n = 4 - 2\epsilon$, both types of divergences appear as poles in $1/\epsilon$ (and potentially $1/\epsilon^2$) in the limit $\epsilon \rightarrow 0$. The IR divergences cancel only after combining the virtual contribution with the corresponding integrated subtraction term. In contrast, UV divergences are removed by renormalization. In this work, renormalization is performed in the modified minimal subtraction scheme ($\overline{MS}$) [69], which subtracts the UV poles together with the associated constants.

Taking the loop integral from equation 3.17 and keeping in mind that $\not{p} = \gamma_\mu p^\mu$, the expression

can be rewritten as

$$\begin{aligned}
& \int \frac{d^n k}{(2\pi)^n} \frac{\mu^{2\epsilon} [\not{k} + \not{p}_1 + \not{p}_2 + m_b][\not{k} + \not{p}_1 + m_b]}{k^2 [(k + p_1)^2 - m_b^2][(k + p_1 + p_2)^2 - m_b^2]} \\
&= \int \frac{d^n k}{(2\pi)^n} \frac{\mu^{2\epsilon} [\not{k} + \not{q}_2 + m_b][\not{k} + \not{q}_1 + m_b]}{k^2 [(k + q_1)^2 - m_b^2][(k + q_2)^2 - m_b^2]} \\
&= \mu^{2\epsilon} \cdot \int \frac{d^n k}{(2\pi)^n} \frac{\gamma_\mu \gamma_\nu k^\mu k^\nu + \gamma_\mu k^\mu \cdot (q_1 + q_2 + 2m_b) + q_1 \cdot q_2 + q_1 \cdot m_b + q_2 \cdot m_b + m_b^2}{k^2 [(k + q_1)^2 - m_b^2][(k + q_2)^2 - m_b^2]}, \quad (3.19)
\end{aligned}$$

where in the second step the momentum substitutions $q_1 = p_1$ and $q_2 = p_1 + p_2$ have been introduced in order to match the momentum conventions used later in the LoopTools evaluation. In the last line, one can see that the loop integral can be decomposed into scalar, linear, and quadratic tensor integrals, given by

Scalar integral:

$$C_0 = \mu^{2\epsilon} \cdot \int \frac{d^n k}{(2\pi)^n} \frac{1}{k^2 [(k + q_1)^2 - m_b^2][(k + q_2)^2 - m_b^2]}, \quad (3.20)$$

Linear integral:

$$C_\mu = \mu^{2\epsilon} \cdot \int \frac{d^n k}{(2\pi)^n} \frac{k_\mu}{k^2 [(k + q_1)^2 - m_b^2][(k + q_2)^2 - m_b^2]}, \quad (3.21)$$

Quadratic integral:

$$C_{\mu\nu} = \mu^{2\epsilon} \cdot \int \frac{d^n k}{(2\pi)^n} \frac{k_\mu k_\nu}{k^2 [(k + q_1)^2 - m_b^2][(k + q_2)^2 - m_b^2]}. \quad (3.22)$$

It is common to denote three-point one-loop integrals by the letter C . The scalar integral C_0 can be evaluated directly with LoopTools, whereas the tensor integrals C_μ and $C_{\mu\nu}$ have to be reduced first to scalar integrals using Passarino–Veltman reduction [70]:

$$C_\mu = q_{1,\mu} C_1 + q_{2,\mu} C_2 \quad (3.23)$$

$$C_{\mu\nu} = g_{\mu\nu} C_{00} + \sum_{i,j=1}^2 q_{i,\mu} q_{j,\nu} C_{ij} \quad (3.24)$$

The coefficients C_i and C_{ij} are the Passarino–Veltman coefficients which are calculated by LoopTools. The momentum basis (q_1, q_2) is used since LoopTools provides the coefficients in this convention. Moreover, this momentum assignment leads to coefficients that are symmetric in their indices.

LoopTools uses kinematic invariants as input, such as the Higgs and bottom masses. Hence, the Passarino–Veltman coefficients are the same for all phase space points and need to be evaluated only once. Further details on the usage of LoopTools are provided in appendix A.

It is important to note that the integrals returned by LoopTools differ by an overall prefactor from those defined in equations 3.20–3.22. In LoopTools, the scalar integral is defined as

$$C_0^{LT} = \frac{\mu^{2\epsilon}}{i\pi^{2-\epsilon}r_\Gamma} \int d^m k \frac{1}{k^2[(k+q_1)^2 - m_b^2][(k+q_2)^2 - m_b^2]}, \quad (3.25)$$

with

$$r_\Gamma = \frac{\Gamma^2(1-\epsilon)\Gamma(1+\epsilon)}{\Gamma(1-2\epsilon)}. \quad (3.26)$$

The same normalization applies to the linear and quadratic LoopTools-integrals C_μ^{LT} and $C_{\mu\nu}^{LT}$. Comparing conventions yields

$$C_0 = C_0^{LT} \cdot ir_\Gamma \frac{(4\pi)^\epsilon}{16\pi^2}, \quad (3.27)$$

and analogously for C_μ and $C_{\mu\nu}$. Using the Passarino–Veltman reduction and the LoopTools normalization, the one-loop amplitude can be written in terms of LoopTools coefficients as

$$\begin{aligned} \mathcal{M}_V &= - \sum_a g_S^2 \cdot \frac{m_b}{\nu} \cdot (T^a)(T^a) \cdot \bar{u}(-p_1 - p_2) \gamma^\mu \cdot [C_{\mu\nu} \gamma^\mu \gamma^\nu \\ &\quad + C_\mu \gamma^\mu (q_1 + q_1 + 2m_b) + C_0(q_1 q_2 + q_1 m_b + q_2 m_b + m_b^2)] \gamma_\mu v(p_1) \\ &= -ir_\Gamma \frac{(4\pi)^\epsilon}{16\pi^2} \cdot \sum_a g_S^2 \cdot \frac{m_b}{\nu} \cdot (T^a)(T^a) \cdot \bar{u}(-p_1 - p_2) \gamma^\mu \cdot [C_{\mu\nu}^{LT} \gamma^\mu \gamma^\nu \\ &\quad + C_\mu^{LT} \gamma^\mu (q_1 + q_1 + 2m_b) + C_0^{LT}(q_1 q_2 + q_1 m_b + q_2 m_b + m_b^2)] \gamma_\mu v(p_1). \end{aligned} \quad (3.28)$$

The LoopTools output is configured such that only the IR divergences are returned. UV divergences are already removed through renormalization within the program. In the present case with massive bottom quarks, there are no collinear singularities. This implies that no double pole $1/\epsilon^2$ arises, only soft IR poles remain. This is consistent with the LoopTools output listed in table A.1 in the appendix.

To separate the result into a finite part and a $1/\epsilon$ pole term, the overall ϵ -dependent prefactor is expanded as

$$ir_\Gamma \frac{(4\pi)^\epsilon}{16\pi^2} = \frac{i}{16\pi^2} [1 + \epsilon(\ln(4\pi) - \gamma_E)] + \mathcal{O}(\epsilon^2), \quad (3.29)$$

where γ_E is the Euler-Mascheroni constant. Parametrizing expression 3.28 as

$$\mathcal{M}_V = ir_\Gamma \frac{(4\pi)^\epsilon}{16\pi^2} \cdot \left(\mathcal{M}'_{V,finite} + \mathcal{M}'_{V,1/\epsilon} \cdot \frac{1}{\epsilon} \right), \quad (3.30)$$

and inserting the expansion 3.29 into 3.30, one obtains

$$\mathcal{M}_{\mathcal{V}} = \frac{i}{16\pi^2} \cdot \left[\mathcal{M}'_{\mathcal{V},finite} + (\ln(4\pi) - \gamma_E) \cdot \mathcal{M}'_{\mathcal{V},1/\epsilon} + \mathcal{M}'_{\mathcal{V},1/\epsilon} \cdot \frac{1}{\epsilon} \right]. \quad (3.31)$$

The explicit expressions for the finite and pole contributions read

$$\begin{aligned} \mathcal{M}'_{\mathcal{V},finite} = & - \sum_a g_S^2 \cdot \frac{m_b}{\nu} \cdot (T^a)(T^a) \cdot \bar{u}(-p_1 - p_2) \gamma^\mu \cdot \left[\{C_{\mu\nu}^{LT}\}_{finite} \gamma^\mu \gamma^\nu \right. \\ & + \{C_\mu^{LT}\}_{finite} \gamma^\mu (q_1 + q_1 + 2m_b) \\ & \left. + \{C_0^{LT}\}_{finite} (q_1 q_2 + q_1 m_b + q_2 m_b + m_b^2) \right] \gamma_\mu v(p_1), \end{aligned} \quad (3.32)$$

and

$$\begin{aligned} \mathcal{M}'_{\mathcal{V},1/\epsilon} = & - \sum_a g_S^2 \cdot \frac{m_b}{\nu} \cdot (T^a)(T^a) \cdot \bar{u}(-p_1 - p_2) \gamma^\mu \cdot \left[\{C_{\mu\nu}^{LT}\}_{1/\epsilon} \gamma^\mu \gamma^\nu \right. \\ & + \{C_\mu^{LT}\}_{1/\epsilon} \gamma^\mu (q_1 + q_1 + 2m_b) \\ & \left. + \{C_0^{LT}\}_{1/\epsilon} (q_1 q_2 + q_1 m_b + q_2 m_b + m_b^2) \right] \gamma_\mu v(p_1). \end{aligned} \quad (3.33)$$

Contributions of order $\mathcal{O}(\epsilon)$ can be omitted, as they do not affect the result once the $1/\epsilon$ divergence has been cancelled and the limit $\epsilon \rightarrow 0$ was taken.

Thus, the finite and divergent virtual matrix elements can be identified as

$$\mathcal{M}_{\mathcal{V},finite} = \frac{i}{16\pi^2} \cdot [\mathcal{M}'_{\mathcal{V},finite} + (\ln(4\pi) - \gamma_E) \cdot \mathcal{M}'_{\mathcal{V},1/\epsilon}], \quad \mathcal{M}_{\mathcal{V},1/\epsilon} = \frac{i}{16\pi^2} \cdot \mathcal{M}'_{\mathcal{V},1/\epsilon}. \quad (3.34)$$

3.1.4 Virtual Subtraction $\mathcal{I}^S$

To cancel the IR divergences from the virtual contribution $\mathcal{V}_n$, one adds the integrated subtraction term $\mathcal{I}_n^S$, such that the combination $\mathcal{V}_n + \mathcal{I}_n^S$ is finite and can be integrated over the n-particle phase space. In the dipole formalism, this term is defined as [36]

$$\mathcal{I}_n^S = \langle \mathcal{M}_{\mathcal{B}} | I_n | \mathcal{M}_{\mathcal{B}} \rangle \quad (3.35)$$

with

$$\begin{aligned}
I_n(\epsilon, \mu^2; \{p_i, m_i\}) = & -\frac{\alpha_S}{2\pi} \frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \sum_{j=q, \bar{q}} \frac{1}{T_j^2} \sum_{k \neq j} T_j \cdot T_k \\
& \cdot \left[T_j^2 \left(\frac{\mu^2}{s_{jk}} \right)^\epsilon \left(\mathcal{V}_j(s_{jk}, m_j, m_k, \{m_F\}; \epsilon, \kappa) - \frac{\pi^2}{3} \right) + \Gamma_j(\mu, m_j; \epsilon) \right. \\
& \left. + \gamma_j \cdot \ln \frac{\mu^2}{s_{jk}} + \gamma_j + K_j + O(\epsilon) \right], \tag{3.36}
\end{aligned}$$

where $s_{jk} = 2p_j p_k$. The flavour-dependent constants for quarks and antiquarks are

$$\gamma_q = \frac{3}{2} C_F, \quad K_q = \left(\frac{7}{2} - \frac{\pi^2}{6} \right) C_F. \tag{3.37}$$

For massive quarks and antiquarks, $\Gamma_j = \Gamma_q$ is given by

$$\Gamma_q(\mu, m_q; \epsilon) = T_q^2 \left(\frac{1}{\epsilon} - \ln \frac{m_q^2}{\mu^2} - 2 \right) + \gamma_q \ln \frac{m_q^2}{\mu^2} = C_F \left[\frac{1}{\epsilon} + \frac{1}{2} \ln \frac{m_q^2}{\mu^2} - 2 \right]. \tag{3.38}$$

The kernel $\mathcal{V}_j$ can be decomposed into a part $\mathcal{V}^{(S)}$ symmetric under $j \leftrightarrow k$ and a non-symmetric component $\mathcal{V}^{(NS)}$,

$$\mathcal{V}_j(s_{jk}, m_j, m_k, \{m_F\}; \epsilon, \kappa) = \mathcal{V}^{(S)}(s_{jk}, m_j, m_k; \epsilon) + \mathcal{V}_j^{(NS)}(s_{jk}, m_j, m_k, \{m_F\}; \epsilon, \kappa). \tag{3.39}$$

For massive quarks, the symmetric part reads

$$\begin{aligned}
\mathcal{V}^{(S)}(s_{jk}, m_j > 0, m_k > 0; \epsilon) = & \frac{1}{v_{jk}} \left[\frac{1}{\epsilon} \ln \rho - \frac{1}{4} \ln^2 \rho_j^2 - \frac{1}{4} \ln^2 \rho_k^2 - \frac{\pi^2}{6} \right] \\
& + \frac{1}{v_{jk}} \ln \rho \ln \left(\frac{Q_{jk}^2}{s_{jk}} \right), \tag{3.40}
\end{aligned}$$

and the non-symmetric part for a quark emitter is

$$\begin{aligned}
\mathcal{V}_q^{(NS)}(s_{jk}, m_j, m_k) = & \frac{\gamma_q}{T_q^2} \ln \frac{s_{jk}}{Q_{jk}^2} \\
& + \frac{1}{v_{jk}} \left[\ln \rho^2 \ln(1 + \rho^2) + 2\text{Li}_2(\rho^2) - \text{Li}_2(1 - \rho_j^2) - \text{Li}_2(1 - \rho_k^2) - \frac{\pi^2}{6} \right] \\
& + \ln \frac{Q_{jk} - m_k}{Q_{jk}} - 2 \ln \frac{(Q_{jk} - m_k)^2 - m_j^2}{Q_{jk}^2} - \frac{2m_j^2}{s_{jk}} \ln \frac{m_j}{Q_{jk} - m_k} \\
& - \frac{m_k}{Q_{jk} - m_k} + \frac{2m_k(2m_k - Q_{jk})}{s_{jk}} + \frac{\pi^2}{2}, \tag{3.41}
\end{aligned}$$

with $Q_{jk} = \sqrt{Q_{jk}^2}$ and

$$\rho = \sqrt{\frac{1 - \tilde{v}_{ij,k}}{1 + \tilde{v}_{ij,k}}}, \quad \rho_n(\mu_j, \mu_k) = \sqrt{\frac{1 - \tilde{v}_{ij,k} + 2\mu_n^2/(1 - \mu_j^2 - \mu_k^2)}{1 + \tilde{v}_{ij,k} + 2\mu_n^2/(1 - \mu_j^2 - \mu_k^2)}}, \quad (n = j, k). \quad (3.42)$$

Analogous to the real subtraction term, the colour operators in equation 3.36 can be simplified using equation 3.8 together with the identity $T_i^2 = C_i$, where $C_i = C_F$ for quarks [44]. Two types of colour structures are present in equation 3.36:

$$\langle \mathcal{M}_B | \frac{1}{T_j^2} T_j \cdot T_k \cdot T_j^2 | \mathcal{M}_B \rangle = -\langle \mathcal{M}_B | T_j^2 | \mathcal{M}_B \rangle = -C_F \cdot \langle \mathcal{M}_B | \mathcal{M}_B \rangle. \quad (3.43)$$

Including the additional factor $1/T_j^2$ from $\mathcal{V}^{(NS)}$, one of the resulting terms has the following colour structure:

$$\langle \mathcal{M}_B | \frac{1}{T_j^2} T_j \cdot T_k | \mathcal{M}_B \rangle = \langle \mathcal{M}_B | \frac{1}{T_j^2} (-T_j^2) | \mathcal{M}_B \rangle = -\langle \mathcal{M}_B | \mathcal{M}_B \rangle. \quad (3.44)$$

Since equation 3.36 is lengthy, it is convenient for the implementation to parametrize it and explicitly separate the $1/\epsilon$ pole from the finite part. First, we write

$$\Gamma_q := A \cdot \frac{1}{\epsilon} + B, \quad \text{where } A = C_F, \quad B = C_F \cdot \left(\frac{1}{2} \ln \frac{m_q^2}{\mu^2} - 2 \right). \quad (3.45)$$

The kernel $\mathcal{V}_j = \mathcal{V}^{(S)} + \mathcal{V}^{(NS)}$ is parametrized as

$$\mathcal{V}_j := C_j \cdot \frac{1}{\epsilon} + D_j \cdot \frac{1}{T_j^2} + E_j, \quad (3.46)$$

with

$$C_j = \frac{1}{v_{jk}} \cdot \ln \rho, \quad D_j = \gamma_q \cdot \ln \frac{s_{jk}}{Q_{jk}^2}, \quad (3.47)$$

$$\begin{aligned} E_j = & \frac{1}{v_{jk}} \left[-\frac{1}{4} \ln^2 \rho_j^2 - \frac{1}{4} \ln^2 \rho_k^2 - \frac{\pi^2}{6} \right] + \frac{1}{v_{jk}} \ln \rho \ln \left(\frac{Q_{jk}^2}{s_{jk}} \right) \\ & + \frac{1}{v_{jk}} \left[\ln \rho^2 \ln(1 + \rho^2) + 2\text{Li}_2(\rho^2) - \text{Li}_2(1 - \rho_j^2) - \text{Li}_2(1 - \rho_k^2) - \frac{\pi^2}{6} \right] \\ & + \ln \frac{Q_{jk} - m_k}{Q_{jk}} - 2 \ln \frac{(Q_{jk} - m_k)^2 - m_j^2}{Q_{jk}^2} - \frac{2m_j^2}{s_{jk}} \ln \frac{m_j}{Q_{jk} - m_k} \\ & - \frac{m_k}{Q_{jk} - m_k} + \frac{2m_k(2m_k - Q_{jk})}{s_{jk}} + \frac{\pi^2}{2}, \end{aligned} \quad (3.48)$$

and finally, collecting the remaining terms in equation 3.36:

$$F := \gamma_q \cdot \ln \frac{\mu^2}{s_{jk}} + \gamma_q + K_q. \quad (3.49)$$

Inserting these parametrizations into equation 3.36 yields

$$I_n(\epsilon, \mu^2; \{p_i, m_i\}) = -\frac{\alpha_S}{2\pi} \frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \sum_{j=q, \bar{q}} \frac{1}{T_j^2} \sum_{k \neq j} T_j \cdot T_k \cdot \left[T_j^2 \left(\frac{\mu^2}{s_{jk}} \right)^\epsilon \left(\frac{C_j}{\epsilon} + \frac{D_j}{T_j^2} + E_j - \frac{\pi^2}{3} \right) + \frac{A}{\epsilon} + B + F + \mathcal{O}(\epsilon) \right]. \quad (3.50)$$

Evaluating $\langle \mathcal{M}_B | I_n | \mathcal{M}_B \rangle$ and simplifying the colour operators using equations 3.43 and 3.44 gives

$$\begin{aligned} \langle \mathcal{M}_B | I_n | \mathcal{M}_B \rangle &= |\mathcal{M}_B|^2 \cdot \frac{\alpha_S}{2\pi} \frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \cdot \left[C_F \left(\frac{\mu^2}{s_{q\bar{q}}} \right)^\epsilon \left(\frac{C_q}{\epsilon} + \frac{D_q}{C_F} + E_q - \frac{\pi^2}{3} \right) \right. \\ &\quad \left. + \frac{C_{\bar{q}}}{\epsilon} + \frac{D_{\bar{q}}}{C_F} + E_{\bar{q}} - \frac{\pi^2}{3} \right) + \frac{2A}{\epsilon} + 2B + 2F + \mathcal{O}(\epsilon) \Big] \\ &= |\mathcal{M}_B|^2 \left\{ \left(\frac{4\pi\mu^2}{s_{q\bar{q}}} \right)^\epsilon \frac{1}{\Gamma(1-\epsilon)} \frac{1}{\epsilon} \cdot \frac{\alpha_S}{2\pi} C_F (C_q + C_{\bar{q}}) \right. \\ &\quad \left. + \left(\frac{4\pi\mu^2}{s_{q\bar{q}}} \right)^\epsilon \frac{1}{\Gamma(1-\epsilon)} \cdot \frac{\alpha_S}{2\pi} C_F \cdot \left(\frac{D_q}{C_F} + E_q + \frac{D_{\bar{q}}}{C_F} + E_{\bar{q}} - \frac{2\pi^2}{3} \right) \right. \\ &\quad \left. + \frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \frac{1}{\epsilon} \cdot \frac{\alpha_S}{2\pi} \cdot 2A + \frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \cdot \frac{\alpha_S}{2\pi} (2B + 2F) \right\} + \mathcal{O}(\epsilon). \quad (3.51) \end{aligned}$$

Here the symmetry $s_{q, \bar{q}} = s_{\bar{q}, q}$ was used. In the second step of equation 3.51, the expression was sorted by the different ϵ -terms. In the next step, all ϵ -dependent factors are expanded in order to separate the expression into a divergent part $1/\epsilon$ and a finite part $\mathcal{O}(\epsilon^0)$. Terms of higher order in ϵ are discarded. The required expansions are

$$\left(\frac{4\pi\mu^2}{s_{q\bar{q}}} \right)^\epsilon \frac{1}{\Gamma(1-\epsilon)} \frac{1}{\epsilon} = \frac{1}{\epsilon} + \ln \left(\frac{4\pi\mu^2}{s_{q\bar{q}}} \right) - \gamma_E + \mathcal{O}(\epsilon), \quad (3.52)$$

$$\left(\frac{4\pi\mu^2}{s_{q\bar{q}}} \right)^\epsilon \frac{1}{\Gamma(1-\epsilon)} = 1 + \mathcal{O}(\epsilon), \quad (3.53)$$

$$\frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \frac{1}{\epsilon} = \frac{1}{\epsilon} + \ln(4\pi) - \gamma_E + \mathcal{O}(\epsilon), \quad (3.54)$$

$$\frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} = 1 + \mathcal{O}(\epsilon). \quad (3.55)$$

Inserting equations 3.52 - 3.55 into 3.51 finally yields

$$\langle \mathcal{M}_B | I_n | \mathcal{M}_B \rangle = |\mathcal{M}_B|^2 \frac{\alpha_S}{2\pi} \left[I_{finite} + \frac{1}{\epsilon} I_{1/\epsilon} \right] + \mathcal{O}(\epsilon), \quad (3.56)$$

where

$$I_{1/\epsilon} = C_F(C_q + C_{\bar{q}}) + 2A, \quad (3.57)$$

and the finite contribution reads

$$\begin{aligned} I_{finite} = & C_F(C_q + C_{\bar{q}}) \left(\ln \left(\frac{4\pi\mu^2}{s_{q\bar{q}}} \right) - \gamma_E \right) \\ & + C_F \left(\frac{D_q}{C_F} + \frac{D_{\bar{q}}}{C_F} + E_q + E_{\bar{q}} - \frac{2\pi^2}{3} \right) \\ & + 2A(\ln(4\pi) - \gamma_E) + 2B + 2F. \end{aligned} \quad (3.58)$$

3.2 Existing Hard Decay Framework

This section provides an overview of the structure and the central classes in SHERPA that are relevant for the calculation of hard decays. The new NLO implementation builds upon this existing framework.

Figure 3.4 shows an overview of the most important classes and their schematic relationships. In summary, they can be described as follows:

- **Hard_Decay_Handler** and its superclass **Decay_Handler_Base**: manage the overall decay procedure. The **Decay_Handler_Base** class provides general structures that can also be used by other decay handlers. The **Hard_Decay_Handler** inherits from this base class and therefore makes use of its common structures. In addition, it contains functionality that is specific to hard decays.
- **Decay_Channel**: instances of this class represent specific physical decay channels. They contain all relevant Feynman diagrams associated with the channel, initiate their evaluation and compute and store the corresponding partial decay width.
- **Spin_Amplitudes**: represents Feynman diagrams. Different subclasses correspond to different diagram topologies. At LO, tree-level diagrams with two and three final states are required, which are implemented in the classes **Comix1to2** and **Comix1to3**, respectively.

The individual components are described in more detail in the following sections.

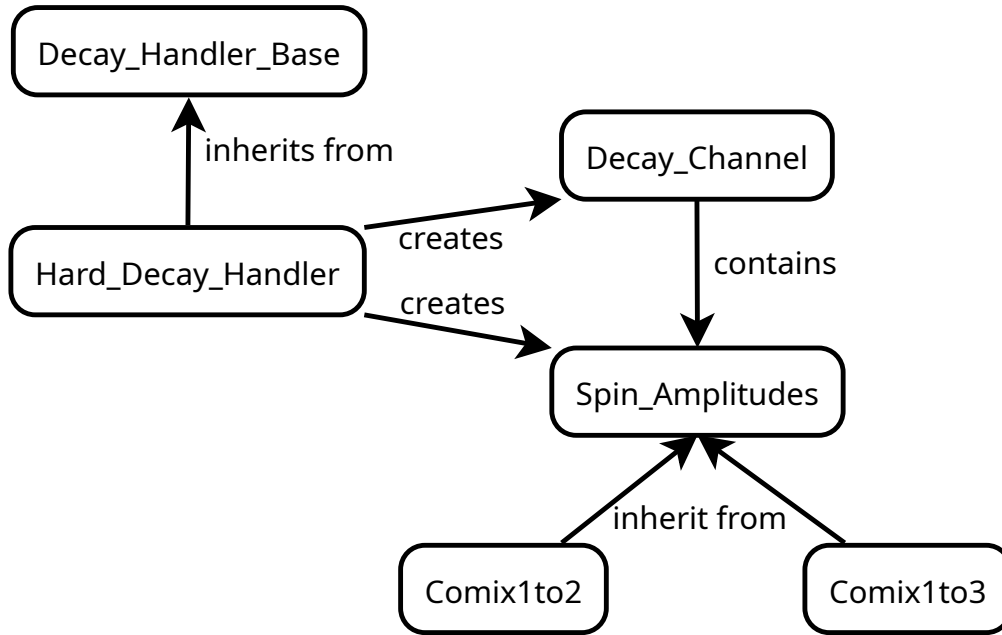


Figure 3.4: Overview of classes relevant for the hard decay calculations and their connection in SHERPA.

3.2.1 The Class `Spin_Amplitudes`

Figure 3.5 schematically shows the two Feynman diagrams used in the LO calculation. The $1 \rightarrow 3$ diagrams are also referred to as off-shell diagrams, as they contain an off-shell propagator. Their purpose is to replace direct ($1 \rightarrow 2$) decays that are kinematically forbidden on-shell, i.e. cases in which the sum of the decay product masses exceeds the mass of the incoming particle. An example of this is the decay $H \rightarrow ZZ$, which is kinematically forbidden for two on-shell Z bosons, since $2m_Z > m_H$. In this case, one of the Z bosons is treated as off-shell and subsequently decays into lighter particles, for example muons.

Figure 3.6 shows the class diagram of the `Spin_Amplitudes` classes. When a `Spin_Amplitude` object is created within the `Hard_Decay_Handler`, the appropriate external wave functions and vertices are selected based on the provided flavour list `m_flavs`. The variable `m_flavs` contains the flavour information of the incoming particle as well as of the outgoing particles. The diagram is built as follows: starting from the external particles, currents are constructed, which correspond to external particle currents and propagators that are connected via vertices. The final numerical evaluation of the diagrams with concrete momenta is performed in the method `Calculate`. The implementation of `Calculate` is provided in the respective subclasses, while the base class defines only the general interface. Via the flag “anti”, it can be specified whether the calculation should be performed for a decaying particle or a decaying antiparticle.

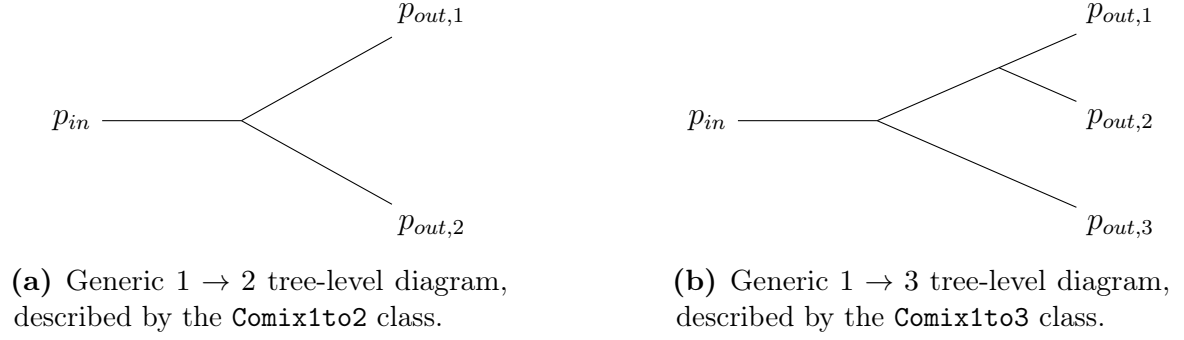


Figure 3.5: Schematic Feynman diagrams already implemented in the hard decay framework via `Spin_Amplitudes` subclasses. The lines can represent fermions, gauge bosons, and Higgs bosons, as long as the resulting Feynman diagram is valid.

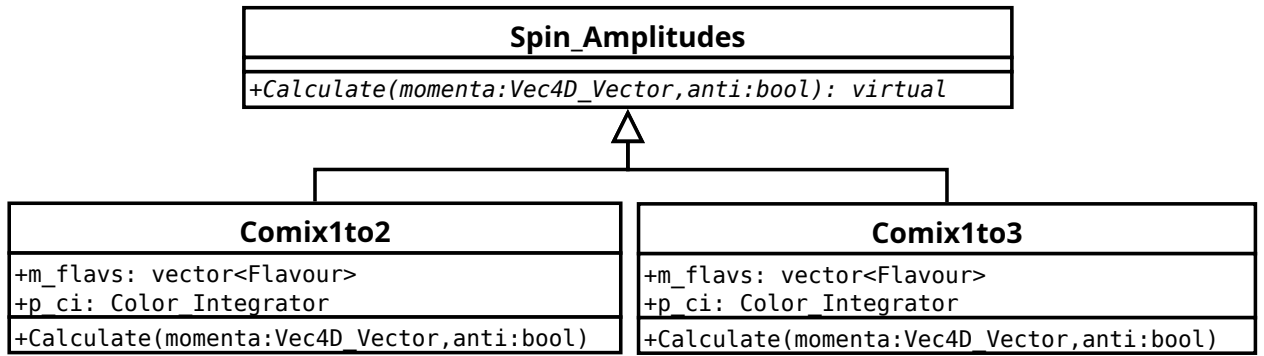


Figure 3.6: Unified Modelling Language (UML) class diagram of the `Spin_Amplitudes` class and subclasses. UML class diagrams can be interpreted as follows:

The first section contains the name of the class, the second section lists the member variables, and the third section specifies the methods. A “+” preceding a variable or method name indicates that it is public, whereas a “-” denotes private access, meaning that it can only be accessed from within the class. The input parameters of the methods are given in brackets, and the return type is specified after the colon. Virtual methods are written in italics. The presented UML diagrams and descriptions are not intended to be exhaustive. Only the variables and methods relevant for the NLO extension are shown, although most classes contain considerably more members. Furthermore, no distinction is made here between pointer and value variables.

Accordingly, the currents of the outgoing particles are adjusted. In this way, the same `Comix1to2` object can, for example, be used to compute the decays $W^+ \rightarrow c\bar{s}$ and $W^- \rightarrow \bar{c}s$. The computation is performed at the amplitude level, i.e. the matrix element is not squared at this stage. Each helicity configuration is treated separately and calculated only for a single colour configuration. The colour configuration is randomly selected by the `Colour_Integrator` `p_ci`, while ensuring that only valid colour assignments are generated. After the amplitude has been computed, it is multiplied by an appropriate colour factor to account for the correct colour multiplicity as explained in section 3.1.2.

These helicity-dependent amplitude values are stored in a list structure in the corresponding `Spin_Amplitudes` object, where the entries are ordered as illustrated in table 3.2 for the

example of a `Comix1to2` object. In the illustrated case, the incoming particle has only a single helicity configuration (e.g. Higgs Boson), denoted by 0, while the outgoing particles, for example quarks, can each take helicity values $\pm 1/2$. In this notation, the helicity state $-1/2$ is labelled as 0, while the helicity state $+1/2$ is labelled as 1.

index	h_{in}	h_{out1}	h_{out2}
0	0	0	0
1	0	1	0
2	0	0	1
3	0	1	1

Table 3.2: Ordering of the helicity amplitudes in the `Comix1to2` object on the example $H \rightarrow f\bar{f}$. The incoming Higgs has only one helicity configuration, which is assigned the number 0. The outgoing fermions can have two possible helicity states, which are called 0 and 1. These indices do not necessarily match the actual helicity values.

In order to obtain a squared matrix element, the sum over all helicies of the outgoing particles is performed later in the `Decay_Channel` during the calculation of the partial decay width.

3.2.2 Hard Decay Initialization

The hard decay procedure can be divided into two phases: the initialization phase and the event generation phase. During the initialization phase, all `Decay_Channel` objects are initialized and their partial decay widths are computed.

In addition, the following classes are required:

- `Decay_Table`: stores all decay channels of a particle together with its total decay width.
- `Decay_Map`: stores all `Decay_Table` objects that are created during the execution of the `Hard_Decay_Handler`.

The initialization phase begins with the creation of `Decay_Table` objects for all particle flavours and their corresponding antiparticles that may undergo hard decays. These particles must satisfy the following criteria:

- They must be unstable.
- They must be massive.
- They must be elementary particles, i.e. hadrons are excluded.

Particles satisfying these criteria are inserted into the `Decay_Map`. Figure 3.8 shows the rough sequence of the handling of `Spin_Amplitudes` objects in the initialization phase, and the calculation of the partial width of the `Decay_Channels`.

First, the direct ($1 \rightarrow 2$) decays are initialized in the `Hard_Decay_Handler` in the `Initialize`

`DirectDecays` method. Within this method, all valid `Decay_Channels` of a given flavour are created and added to the `Decay_Table`. Then, for every `Decay_Channel`, the corresponding `Comix1to2` object is created and added to the channel. Furthermore, it is verified whether the decay is kinematically allowed, which requires the mass of the decaying particle to exceed the sum of the masses of the outgoing particles. If this condition is not satisfied, the channel is deactivated. However, it may still contribute in the case of an $1 \rightarrow 3$ off-shell decay. Finally, the partial decay width of each `Decay_Channel` is computed using the method `CalculateWidth`. At the end of the `InitializeDirectDecays` method, the `Decay_Tables` of all unstable hard particles have been filled with $1 \rightarrow 2$ `Decay_Channels` and the corresponding partial widths have been calculated.

Afterwards, the $1 \rightarrow 3$ decays are initialized via `InitializeOffshellDecays`. As in the case of direct decays, new `Decay_Channel` objects are created, their partial decay widths are computed, and both are added to the `Decay_Table` of the decaying particle. For this purpose, the method `ResolveDecay` is called to determine whether one or both of the outgoing particles from the previously generated $1 \rightarrow 2$ decays can decay further into additional particles. If this condition is fulfilled and the $1 \rightarrow 2$ channel violates kinematic mass conservation, a new $1 \rightarrow 3$ `Decay_Channel` is generated. The corresponding Feynman diagrams are created as `Comix1to3` objects and added to the channel, after which the partial decay width is calculated and also associated with the channel.

This procedure may lead to multiple `Decay_Channel` objects with identical initial and final states. In such cases, these channels are merged: all Feynman diagrams contributing to the same physical process are combined within a single channel, and the partial decay width is updated accordingly. Finally, the total decay width of the `Decay_Table` is computed by summing all partial widths of the channels.

Calculation of the Partial Width in `Decay_Channel`:

The most important methods and variables of the `Decay_Channel` class are shown in figure 3.7. The partial decay width of a channel is determined via Monte Carlo sampling by evaluating the phase space integral of the squared matrix element according to equation 2.43.

The procedure is as follows: First, a random kinematically allowed phase space point is generated in the rest frame of the decaying particle. This step is performed in the method `Differential`. For each generated phase space configuration, the squared matrix element is evaluated in the method `ME2`. Repeating this procedure for n different phase space points yields n values of the squared matrix element. These values are summed and divided by the number of generated points n , providing a Monte Carlo estimate of the phase space integral of the squared matrix element. Finally, the result is multiplied by the factor $\frac{1}{2 \cdot m_{\text{decaying}}}$ to obtain the partial decay width.

The computation of the squared matrix element in `ME2` for one phase space point proceeds as

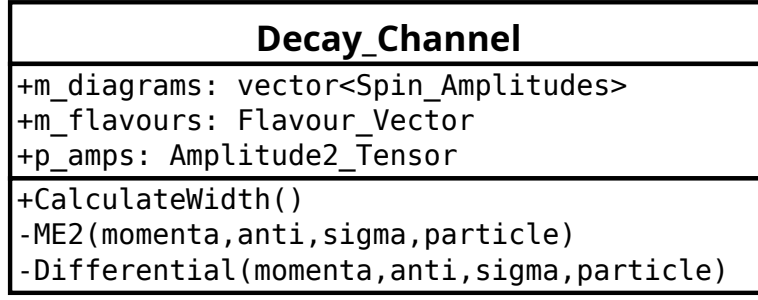


Figure 3.7: UML diagram containing only relevant variables and methods of the `Decay_Channel` class. The member variables are: `m_diagrams`: stores the `Spin_Amplitudes` objects belonging to the channel; `m_flavours`: contains the flavour configuration of the channel (i.e. the flavours of the incoming and outgoing particles); `p_amps`: stores the decay matrix as described in section 2.3.2. This is used during the event generation phase to take spin correlations into account.

follows:

First, the calculation of the `Spin_Amplitudes` is initiated with the generated momenta. This yields the helicity amplitudes, which correspond to the helicity-dependent components $A_{i,\lambda_i\lambda_{o1}\lambda_{o2}\dots}$ of the matrix element. The squared matrix element $|\mathcal{M}|^2$ is then obtained by summing over all contributing diagrams and helicities of the outgoing particles, denoted by $\lambda_{o1}, \lambda_{o2}, \dots$. Averaging over the helicities of the incoming particle leads to

$$|\mathcal{M}|^2 = \frac{1}{N_{\lambda_i}} \sum_{\text{diagrams}} \sum_{k,l} \sum_{\lambda_i} \sum_{\lambda_{o1}, \lambda_{o2} \dots} A_{k,\lambda_i\lambda_{o1}\lambda_{o2}\dots} \cdot (A_{l,\lambda_i\lambda_{o1}\lambda_{o2}\dots})^*. \quad (3.59)$$

Here, N_{λ_i} denotes the number of helicity states of the initial particle over which the squared matrix element is averaged.

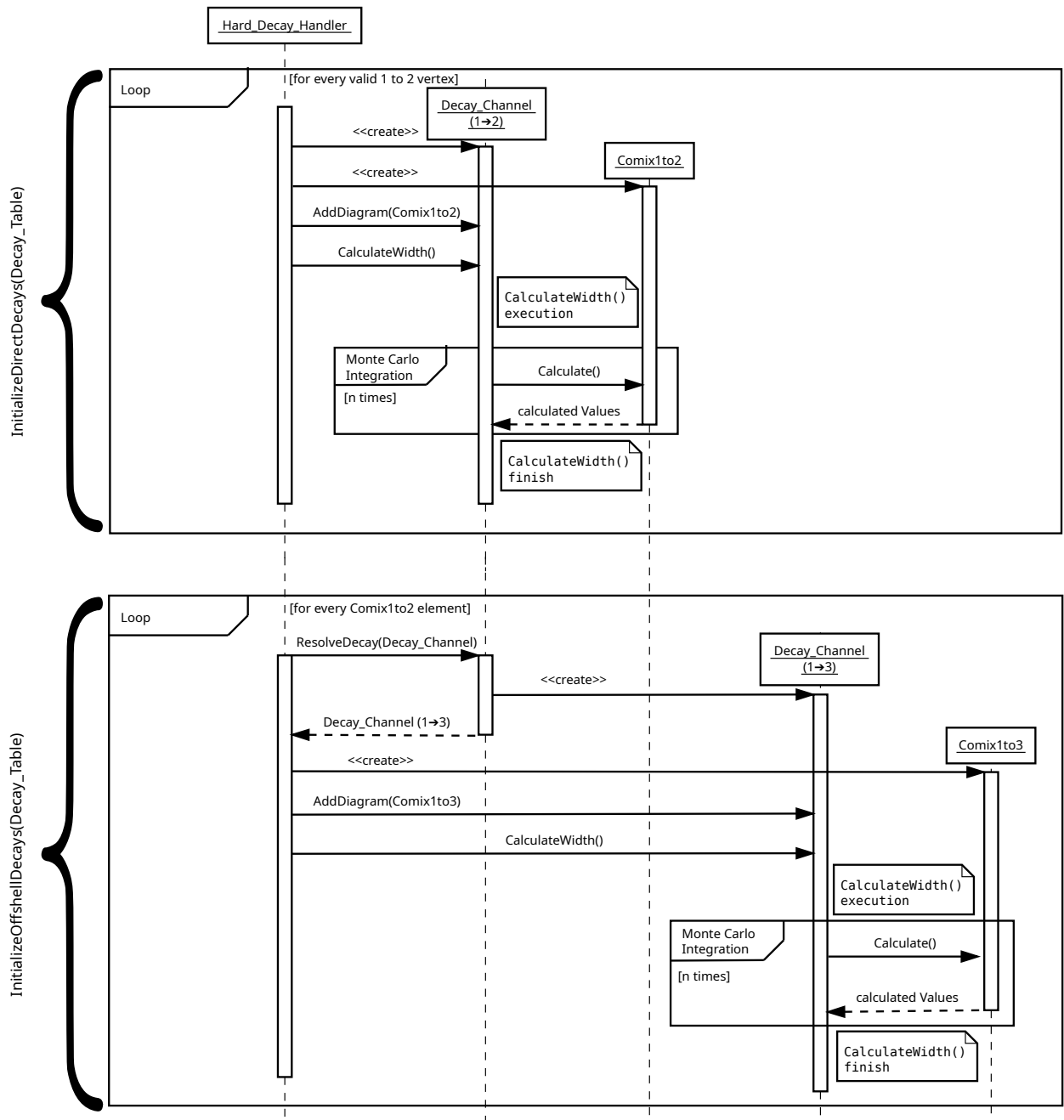


Figure 3.8: Sequence diagram for the initialization phase of the hard decays. The diagram shows only selected processes and is not exhaustive. The boxes represent instances of classes. The dashed vertical lines correspond to the lifelines of these objects, indicating their existence over time. The rectangular bars on the lifelines denote activation periods, during which the respective object is actively executing an operation. The arrows represent messages exchanged between the objects, illustrating their interactions and the temporal order of method calls.

3.2.3 Event Generation

During the event generation phase, the production process and the hard decay can either be fully specified by the user in the configuration file (run card), or certain choices are left to the program. For example, if no specific `Decay_Channel` is defined, a channel is selected for each event by weighted random sampling, meaning that the probability of selecting a given channel is proportional to its branching ratio. The branching ratios are either computed from the previously calculated partial decay widths or taken from tabulated particle data group (PDG) values, depending on the settings specified by the user. The main part of the event generation is performed in the method `TreatInitialBlob` of the `Hard_Decay_Handler` class, where the decay chain is constructed and the core of the spin correlation algorithm is executed. For this purpose, the decay matrix of the decaying particle is computed in the `Decay_Channel`.

Calculation of the Decay Matrix:

Spin correlations are taken into account by constructing a decay matrix, as briefly outlined in section 2.3.2. In this case, the helicities of the incoming particle cannot be averaged over, since this dependence is required by the spin correlation algorithm. Another difference is that the calculation is no longer performed in the rest frame of the incoming particle, as the particle acquires a non-zero momentum through its production. The momenta of the decay products are therefore generated consistently with the momentum of the incoming particle in the `Differential` method of the `Decay_Channel` class. The helicity amplitudes $A_{i,\lambda_i\lambda_{o1}\lambda_{o2}\dots}$ are then calculated by the corresponding `Spin_Amplitudes` objects as described above. Instead of averaging over the helicities of the incoming particle to obtain the squared matrix element, the summation is performed only over the helicities of the outgoing particles, yielding the decay matrix

$$D_{\lambda_i\lambda'_i} = \frac{1}{N_D} \sum_{\text{diagrams } k,l} \sum_{\lambda_{o1},\lambda_{o2}\dots} A_{k,\lambda_i\lambda_{o1}\lambda_{o2}\dots} \cdot (A_{l,\lambda'_i\lambda_{o1}\lambda_{o2}\dots})^*, \quad (3.60)$$

with the normalization factor N_D .

3.3 Extension of the Hard Decay Framework

The NLO corrections were implemented as two separate decay channels, distinguished by the dimensionality of the phase space integration. The contributions $\mathcal{B}$, $\mathcal{V}$ and $\mathcal{I}^S$ form one decay channel, while $\mathcal{R}$ and $\mathcal{D}^S$ form another. This structure allows $H \rightarrow b\bar{b}$ and $H \rightarrow b\bar{b}g$ events to be generated independently, which is particularly useful since a larger number of real emission events is often required in order to obtain sufficient statistical precision. Furthermore, the separate generation of these contributions makes it possible to study the effects of the

virtual and real corrections individually. The four new contributions $\mathcal{V}$, $\mathcal{I}^S$, $\mathcal{R}$ and $\mathcal{D}^S$ were implemented as separate subclasses of `Spin_Amplitudes`, as illustrated in figure 3.9. This ensures that they inherit the standard interface for the calculation of helicity amplitudes via the method `Calculate(momenta, anti)`. In addition to this functionality, the new classes implement several further components required for the evaluation of the respective NLO contributions.

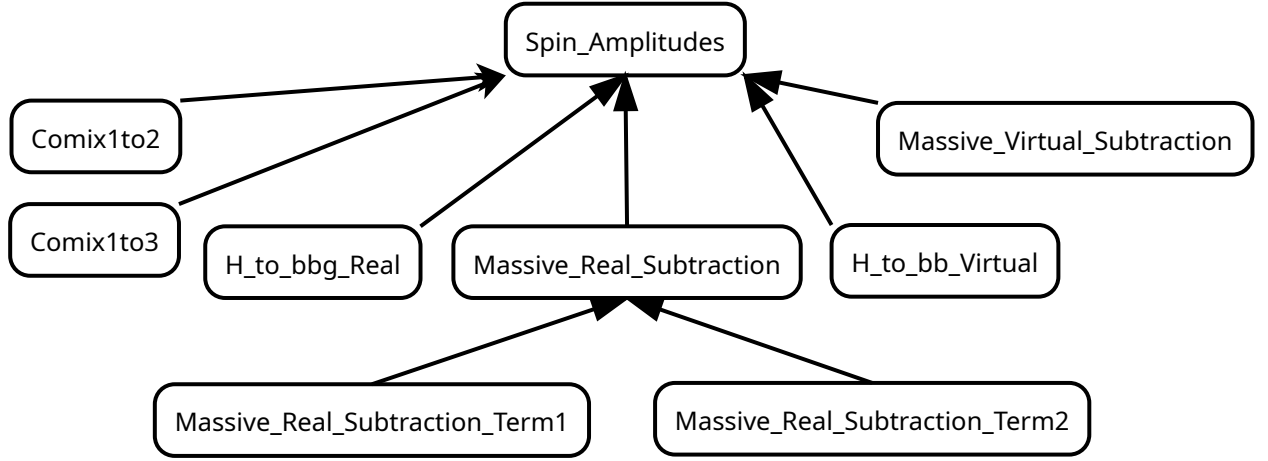


Figure 3.9: Overview of all `Spin_Amplitudes` (sub-)classes, including the newly introduced implementations.

3.3.1 Implemented `Spin_Amplitudes`

Implementation of $\mathcal{R}$: Class `H_to_bbg_Real`

The two real-emission diagrams are represented by instances of the class `H_to_bbg_Real`. The only difference in their construction is the choice of the propagator, which corresponds to the quark before the gluon emission. In one diagram, the b quark acts as the propagator, while in the other, the $\bar{b}$ quark serves as the propagator. The calculation of the helicity-dependent amplitudes are almost identical to the implementation in the class `Comix1to3`. The only difference concerns the choice of the energy scale at which the strong coupling constant α_s is evaluated. In this implementation, the scale is set to the energy of the decaying particle, which in this case is the Higgs boson. In `Comix1to3`, α_s is evaluated at the Z boson mass scale, which corresponds to the default scale value. When applied to a Higgs decay process, this choice leads to a deviation of approximately 5%.

Implementation of $\mathcal{D}^S$: Class `Massive_Real_Subtraction`

The real subtraction terms were implemented following equation 3.10. Separate subclasses, `Massive_Real_Subtraction_Term1` and `Massive_Real_Subtraction_Term2`, are used for the two dipole contributions $\mathcal{D}_{qg,\bar{b}}$ and $\mathcal{D}_{\bar{q}g,b}$, respectively. This separation was introduced mainly to improve the readability and structure of the code. The main part of the implementation

is contained in the base class `Massive_Real_Subtraction`. The subclasses are primarily responsible for preventing errors that could arise from an incorrect ordering of the input momenta when constructing the dipole terms via the base class.

For the $\mathcal{R}$ and $\mathcal{D}^S$ contributions, it is important that the amplitudes are evaluated at the same colour point, since the value of the squared matrix element depends on the chosen colour configuration, as explained in section 3.1.2. To ensure this, additional methods were introduced that extract the colour point from the first diagram evaluated within a given `Decay_Channel` and propagate it to the other `Spin_Amplitudes` objects. This guarantees that all contributions are evaluated consistently at the same colour configuration. The subtraction terms have the general structure of a squared Born matrix element,

$$(\text{Prefactor}) \cdot |\mathcal{M}_B|^2. \quad (3.61)$$

To preserve the design principle that a `Spin_Amplitudes` object represents amplitudes rather than squared matrix elements, the subtraction terms are implemented in the form of

$$\sqrt{\text{Prefactor}} \cdot [\mathcal{M}_B]_{\lambda_i \lambda_{o1} \lambda_{o2} \dots}, \quad (3.62)$$

where the amplitudes depend on the helicities of the Born matrix element. The same approach is used in the class `Massive_Virtual_Subtraction`, which implements the virtual subtraction term $\mathcal{I}^S$. The full contributions are then obtained by squaring the corresponding amplitudes.

Implementation of $\mathcal{V}$: Class `H_to_bb_Virtual`

As described in section 3.1.3, the virtual matrix element was first simplified analytically. Some computationally expensive matrix multiplications were then performed in Python, and the resulting expression was transferred into SHERPA and implemented in the class `H_to_bb_Virtual`. The expression obtained from the Python evaluation reads

$$\begin{aligned} \text{Python Result} = & \gamma^\mu [C_{\mu\nu}^{LT} \gamma^\mu \gamma^\nu + C_\mu^{LT} \gamma^\mu (q_1 + q_1 + 2m_b) \\ & + C_0^{LT} (q_1 q_2 + q_1 m_b + q_2 m_b + m_b^2)] \gamma_\mu. \end{aligned} \quad (3.63)$$

At this stage, the momenta q_1 and q_2 remain unspecified. Since the LoopTools coefficients $C_{\mu\nu}^{LT}$, C_μ^{LT} and C_0^{LT} are independent of the external momenta and thus remain constant, this expression was evaluated only once in Python and the resulting form was copied into the class `H_to_bb_Virtual`. The momenta remain symbolic and are inserted only during the final evaluation in SHERPA. The script used for this calculation is provided in appendix B. Within the class `H_to_bb_Virtual`, the Dirac spinors are constructed as objects of the class `Current`, which is also used in the classes `Comix1to2` and `Comix1to3` for the computation of matrix elements. These objects depend on the helicities of the b and $\bar{b}$ quarks, allowing the

helicity-dependent expressions $\mathcal{M}'_{\mathcal{V},finite}$ and $\mathcal{M}'_{\mathcal{V},1/\epsilon}$ to be computed according to

$$[\mathcal{M}'_{\mathcal{V},finite}]_{\lambda\lambda'} = - \sum_a g_S^2 \frac{m_b}{\nu} \cdot (T^a)^2 \cdot \bar{u}_\lambda(-p_1 - p_2) \cdot (\text{Python Result})_{finite} \cdot v(p_1)_{\lambda'}, \quad (3.64)$$

and

$$[\mathcal{M}'_{\mathcal{V},1/\epsilon}]_{\lambda\lambda'} = - \sum_a g_S^2 \frac{m_b}{\nu} \cdot (T^a)^2 \cdot \bar{u}_\lambda(-p_1 - p_2) \cdot (\text{Python Result})_{1/\epsilon} \cdot v(p_1)_{\lambda'}. \quad (3.65)$$

Altogether, in this case, four helicity components arise with indices $\lambda\lambda' = 00, 01, 10, 11$, since the Higgs is a scalar particle. The final finite and divergent helicity amplitudes $[\mathcal{M}_{\mathcal{V},finite}]_{\lambda\lambda'}$ and $[\mathcal{M}_{\mathcal{V},1/\epsilon}]_{\lambda\lambda'}$, are obtained according to equation 3.34. In the final result, only the finite part is used, while the divergent contribution is cancelled by the subtraction term $\mathcal{I}^S$. For the calculation of the finite virtual contribution

$$\mathcal{V}_{finite} = \sum_{\lambda,\lambda'=0}^1 2 \cdot \Re\{[\mathcal{M}_B]_{\lambda\lambda'} [\mathcal{M}_{\mathcal{V},finite}^*]_{\lambda\lambda'}\}, \quad (3.66)$$

the Born matrix element is also required. In this implementation, it is obtained according to the Feynman rules by directly multiplying the Dirac spinors together with the corresponding prefactor. Although this quantity could also be retrieved from the class `Comix1to2`, it was recomputed here in order to verify that the resulting squared Born matrix element agrees with the value obtained from `Comix1to2`.

Implementation of $\mathcal{I}^S$: Class `Massive_Virtual_Subtraction`

Analogous to the virtual contribution, the virtual subtraction term also consists of a finite part and a $1/\epsilon$ pole. The corresponding expressions were derived in section 3.1.4. The singular contribution is used to verify that the corresponding singular part of the virtual correction, $\mathcal{V}_{1/\epsilon}$, is fully cancelled by the subtraction term. In the final calculation, however, only the finite part of the virtual subtraction term enters the result.

3.3.2 Further Changes

Changes to the `Hard_Decay_Handler`:

No new methods were introduced in the class `Hard_Decay_Handler`. However, several modifications were required to integrate the NLO contributions into the existing framework. Within the method `InitializeDirectDecays`, the construction of the `Decay_Channel` $H \rightarrow b\bar{b}$ was extended. In addition to the Born diagram represented by `Comix1to2`, the diagrams `H_to_bb_Virtual` and `Massive_Virtual_Subtraction` are now also created and added to the channel. Furthermore, the `Decay_Channel` $H \rightarrow b\bar{b}g$ must be constructed explicitly. Under

normal circumstances, this channel is not generated automatically, since it corresponds to a bottom self-interaction which is excluded by the hard decays. Like for the existing $1 \rightarrow 3$ `Decay_Channel` described in section 3.2.2, its creation is handled in the method `ResolveDecay`. Here, the four additional diagrams, corresponding to two instances of `H_to_bbg_Real`, as well as `Massive_Real_Subtraction_Term1` and `Massive_Real_Subtraction_Term2`, are instantiated and added to the $H \rightarrow b\bar{b}g$ channel. In addition, it must be ensured that neither of the two `Decay_Channels` is disabled afterwards. In the default framework, the $1 \rightarrow 2$ channel would normally be deactivated once a valid $1 \rightarrow 3$ channel has been resolved from it. In the present implementation, both channels must remain active, unless specified differently in the user configuration.

A complication arises during the generation of real emission events, since the real contribution $\mathcal{R}$ and the two subtraction terms $\mathcal{D}_{qg,\bar{b}}$ and $\mathcal{D}_{\bar{q}g,b}$ have different kinematics. Due to the emitter-spectator structure, the subtraction terms are evaluated using a mapped Born kinematics, as discussed in section 3.1.1. In these contributions, no explicit gluon appears in the final state, in contrast to the real-emission $\mathcal{R}$. To account for this difference, subevents are introduced for the $H \rightarrow b\bar{b}g$ channel. The purpose of these subevents is to represent the real contribution and the two subtraction terms as separate events in the output, each with its own kinematic configuration. The subevents are generated in the method `TreatInitialBlob` in the `Hard_Decay_Handler` and treated as independent events in the subsequent stages of the program. SHERPA provides a dedicated class for this purpose, namely `NLO_subevt`. It must be taken into account that only a single subevent is generated for the two real-emission diagrams. This is because the diagrams do not contribute independently. Instead, they interfere when calculating the squared matrix element and have the same kinematics. The quantities associated with each event, such as the event weight, particle flavours, and the momenta of all particles involved, must be assigned to the corresponding `NLO_subevt` object.

The event weight normally corresponds to the total cross section and is computed according to equation 2.64 as the product of the production cross section and the branching ratio. For the subevents, a modification is required. Since three subevents are generated per physical event, the sum of their weights should reproduce

$$\sigma_{prod} \cdot \text{BR}(H \rightarrow b\bar{b}g). \quad (3.67)$$

To achieve this, each subevent weight is multiplied by an additional factor w_i with $i = 1, 2, 3$, determined by the relative size of the corresponding squared matrix element. These factors are defined as

$$w_i = \frac{|\mathcal{M}_i|^2}{\sum_{j=1}^3 |\mathcal{M}_j|^2}, \quad \text{and} \quad \sum_{i=1}^3 w_i = 1. \quad (3.68)$$

The final weight of each subevent is therefore given by

$$\sigma_{prod} \cdot \text{BR}(H \rightarrow b\bar{b}g) \cdot w_i. \quad (3.69)$$

The subtraction term contributions are negative. Consequently, the corresponding subevents carry a negative event weight.

Changes to the Decay_Channel:

Figure 3.10 shows the UML diagram of the `Decay_Channel` class, including the newly introduced methods. The matrix element calculation was divided into separate methods for LO and NLO `Decay_Channels`. As before, the method `ME2` is called from within `Differential`. However, it now only checks whether the `Decay_Channel` corresponds to a LO or NLO channel and then dispatches the calculation to either `ME2_LO` or `ME2_NLO`. The method `ME2_LO` is implemented identically to the previous `ME2` method. The distinction between LO and NLO channels is made by the method `isNLO`, which returns `true` if the `Decay_Channel` contains at least one of the newly implemented NLO `Spin_Amplitudes` diagrams. The method `ME2_NLO` differs

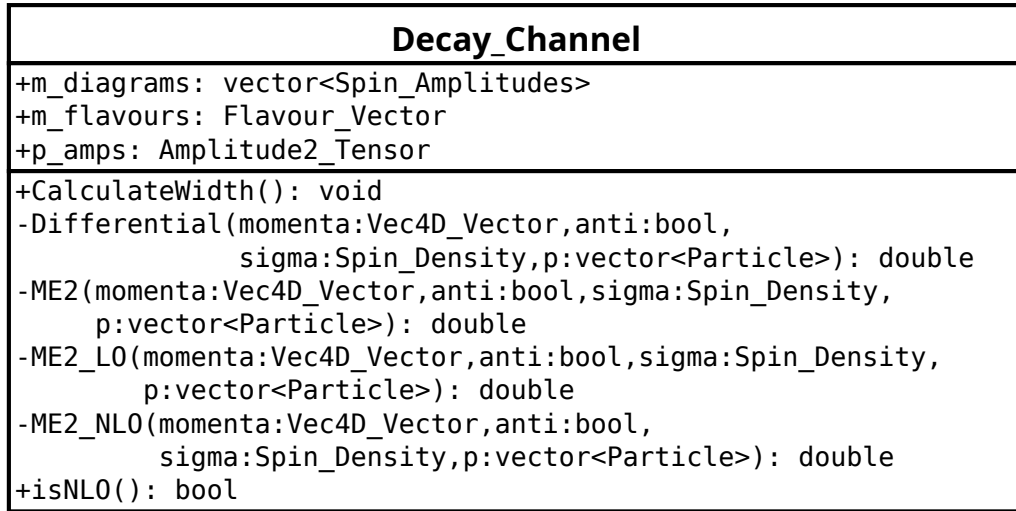


Figure 3.10: UML diagram containing only relevant variables and methods of the `Decay_Channel` class, as well as some newly implemented methods.

from the LO calculation in several aspects. Most importantly, the standard expression for the squared matrix element as a sum over all diagrams, as given in equation 3.59, is no longer applicable, since not all interference terms between diagrams of different types are meant to be included. Instead, the individual contributions are computed separately and combined only afterwards. At this stage, it is also ensured that all diagrams belonging to the real-emission channel $H \rightarrow b\bar{b}g$ are evaluated for the same colour configuration. The decay matrix that is constructed during the event generation stage is likewise computed separately for the individual contributions and only summed afterwards.

4 Fixed Order Results

The first part of this chapter describes how the implemented NLO results were validated, and which quantities were used for the virtual contributions that compared to COMIX do not yet yield correct results. In the second part, several observable distributions from the process $e^+e^- \rightarrow HZ \rightarrow b\bar{b} \mu^+\mu^-$ are presented, and the LO prediction is compared with the NLO corrections arising from the decay $H \rightarrow b\bar{b}$.

4.1 Result Verification and Correction

As a first step, the squared matrix elements used for the calculation of the decay widths of the individual components were verified. In particular, the quantities $\mathcal{R}$, $\mathcal{D}^S$, $\mathcal{I}^S$ and $\mathcal{V}$ were tested separately. Reference values for these contributions can be obtained from the matrix element generator COMIX, which provides them for the standard SHERPA program flow, i.e. without using the hard decay framework. The results for $\mathcal{R}$ and $\mathcal{D}^S$ can be calculated using the run card from section C.1 in the appendix. Since these values depend on the specific momentum configuration, they must be compared pointwise in phase space. The values for $\mathcal{R}$ and $\mathcal{D}^S$ obtained from the hard decay implementation and from the COMIX reference calculation were compared for several different momentum configurations. As discussed in section 3.1.2, the results also depend on the colour configuration. Consequently, the matrix elements can differ by constant colour factors even if evaluated at the same phase space point. For example, a colour-singlet configuration is suppressed by a factor of $1/9$ compared to a colour-octet configuration. After taking this into account, the real contributions $\mathcal{R}$ obtained in the hard decay implementation agree exactly with the expected COMIX results. For the real subtraction terms $\mathcal{D}^S$, however, a constant discrepancy between the COMIX reference values and the hard decay results was observed. In order to reproduce the expected values, the squared matrix elements had to be multiplied by an additional factor of $1/8$ during the initialization phase, and by a factor of $1/4$ during event generation. The origin of these factors requires further investigation. For the present implementation, they have been included explicitly as correction factors in the code.

As mentioned earlier, the calculations of the virtual contribution $\mathcal{V}$ and the integrated subtraction term $\mathcal{I}^S$ do not yield the correct results within the current implementation. Therefore, the

results presented in the following were generated using the COMIX reference values. These were also obtained using the run card described in section C.1. However, the output does not directly provide this contributions, since the output for $\mathcal{V}$ and $\mathcal{I}^S$ already includes additional factors such as the flux factor. Instead, the squared matrix elements were extracted directly from the code, specifically from the class `COMIX/Main/Single_Process` [71]. The virtual matrix element there was computed using the loop integrator RECOLA [65]. The quantities $\mathcal{V}$ and $\mathcal{I}^S$ do not depend on the specific phase space point. Therefore, the values were extracted once and subsequently used to replace the corresponding results obtained from the hard decay calculations. The values used are

$$\mathcal{V} = 31.4553731442691, \quad \mathcal{I}^S = -49.3866539659783, \quad (4.1)$$

where the colour factors are already included. The following results were generated using these substituted values.

4.2 Results for the Process $e^+e^- \rightarrow HZ \rightarrow b\bar{b} \mu^+\mu^-$

The total decay width of the Higgs boson is given by the PDG as $\Gamma = 3.7_{-1.4}^{+1.9}$ MeV [3]. In the present LO calculation, the total Higgs width amounts to 7.0 MeV, of which 5.9 MeV arise from the decay channel $H \rightarrow b\bar{b}$. When including NLO corrections for this channel, the total width decreases to 4.2 MeV, thereby approaching the PDG value. Not all contributing decay channels are included in the SHERPA event generation used here. In particular, the tau lepton mass is set to zero by default, which prevents the decay channel $H \rightarrow \tau^+\tau^-$ from contributing, even though it accounts for roughly 6% of the total Higgs width. The inclusion of this channel would further increase the total Higgs width calculated by SHERPA. The inclusion of higher-order corrections for other decay channels would likely lead to a further improvement of the agreement with the PDG result. The individual results are summarized in table 4.1. While the partial width NLO prediction approaches the PDG value, a noticeable deviation still remains.

	Γ_H^{total}	$\Gamma(H \rightarrow b\bar{b})$	$\text{BR}(H \rightarrow b\bar{b})$
PDG values	$3.7_{-1.4}^{+1.9}$ MeV	2.0 ± 0.3 MeV	53 ± 8 %
SHERPA, $H \rightarrow b\bar{b}$ (LO)	7.0 MeV	5.9 MeV	84.1%
SHERPA, $H \rightarrow b\bar{b}$ (NLO)	4.2 MeV	3.14 MeV	73.9 %

Table 4.1: Total width Γ_H^{total} of the Higgs boson, partial widths Γ for the $H \rightarrow b\bar{b}$ channel, and branching ratio BR for PDG values [3], LO and NLO hard decay implementations in SHERPA. The SHERPA results were calculated using $m_H = 125.09$ GeV and $m_b = 4.92$ GeV. The NLO branching ratio does not correspond to a fully consistent NLO branching ratio, as the remaining Higgs decay channels have not been calculated at NLO.

The virtual and real corrections for $H \rightarrow b\bar{b}$ were implemented in two separate decay channels in SHERPA due to the different phase spaces they are contributing to. The first channel contains the contributions $\mathcal{B}$, $\mathcal{V}$ and $\mathcal{I}^S$, the second one the real-emission contribution $\mathcal{R}$ together with the real subtraction term $\mathcal{D}^S$. The corresponding partial decay widths and branching ratios are listed in table 4.2. In SHERPA, the real subtraction term $\mathcal{D}^S$ is also denoted by “S”. In the following, the first decay channel will be referred to as BVI, while the channel containing the real-emission contributions will be referred to as RS.

decay channel	$\Gamma(H \rightarrow b\bar{b})$ @ NLO	$\text{BR}(H \rightarrow b\bar{b})$
$H \rightarrow b\bar{b}$ (BVI)	3.064 MeV	72.1 %
$H \rightarrow b\bar{b}g$ (RS)	0.075 MeV	1.8 %

Table 4.2: Partial widths and branching ratios for the BVI and RS $H \rightarrow b\bar{b}$ decay channels, calculated with SHERPA’s hard decay framework.

Using the analysis framework Rivet [72], several histograms (figures 4.1 - 4.7) were produced from simulated events of the process $e^+e^- \rightarrow HZ \rightarrow b\bar{b} \mu^+\mu^-$ with beam energies of 500 GeV. The run card used for the event generation is given in appendix C.2. The electron PDF was switched off so that the full 500 GeV beam energy enters the collision. Further steps of the event generation, such as the parton shower, were disabled, and no phase space cuts were applied for the histograms.

The histograms compare the contributions of the BVI and RS channels, as well as the LO and NLO predictions of the decay $H \rightarrow b\bar{b}$. They were scaled to

$$\sigma_{\text{total}} = \sigma_{\text{prod}}(e^+e^- \rightarrow HZ) \cdot \text{BR}(Z \rightarrow \mu^+\mu^-) \cdot \text{BR}(H \rightarrow b\bar{b}) \Big|_{\text{BVI/RS/LO/NLO}}. \quad (4.2)$$

Since the RS channel has a significantly smaller branching ratio than the BVI channel, its contribution was scaled by a factor of 30 in the plots for a clearer comparison. The Rivet analysis code is provided in appendix D.

It should be noted that the NLO branching ratios calculated by SHERPA and given in tables 4.1 and 4.2 do not correspond to consistent NLO branching ratios. The branching ratios depend not only on the partial decay width of the channel under consideration, but also on the total decay width of the Higgs boson. Since the remaining Higgs decay channels are only calculated at LO, the resulting branching ratios are not fully consistent at NLO. For this reason, these values were not used for scaling the histograms. Instead, the branching ratios used for scaling were obtained by dividing the respective partial decay width by the PDG value of the total Higgs width, such that

$$\text{BR}(H \rightarrow b\bar{b}) \Big|_{\text{BVI/RS/LO/NLO}} = \frac{1}{\Gamma_{H,\text{PDG}}^{\text{total}}} \cdot \Gamma(H \rightarrow b\bar{b}) \Big|_{\text{BVI/RS/LO/NLO}}. \quad (4.3)$$

Although this does not correspond to a true NLO branching ratio either, it provides a better estimate of the relative difference between the LO and NLO results, since the same denominator is used for all decay channels.

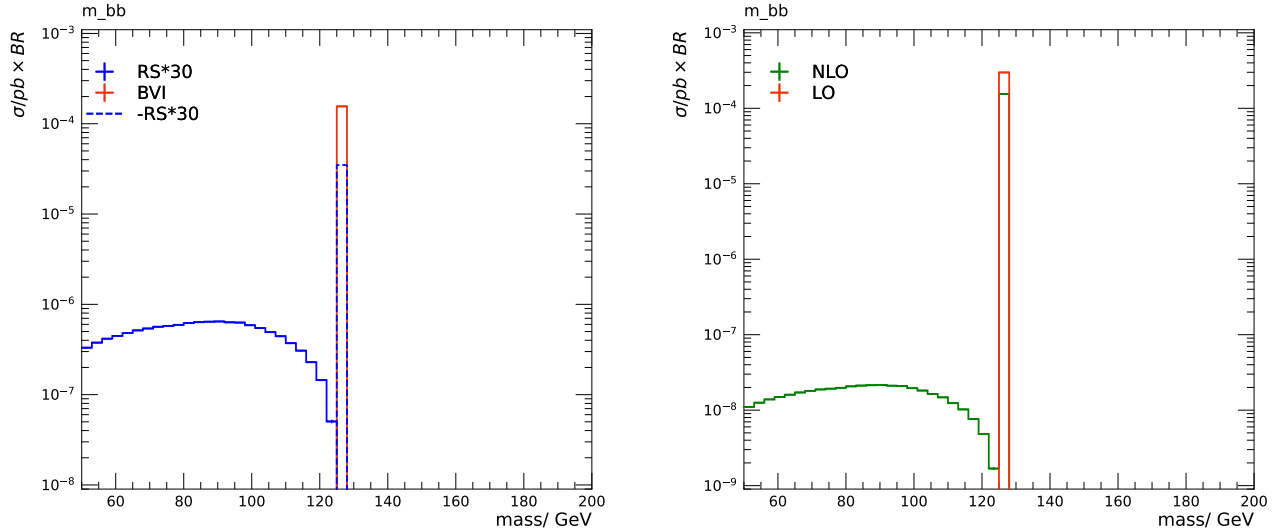


Figure 4.1: Invariant mass distributions of the b-jet pair. Left: comparison of the BVI and RS channels. Right: comparison of the LO and NLO distributions. The RS distribution becomes negative in certain regions. Since negative values cannot be displayed on a logarithmic axis, the absolute values of the negative RS contribution are plotted as a dashed line. A sharp peak appears at the Higgs mass $m_H \approx 125$ GeV, while the real emission events are distributed in the region to the left of the peak.

Figure 4.1 shows the invariant mass distribution of the $b\bar{b}$ -jet system. It can be seen that all of the BVI and LO events exhibit a sharp peak at the Higgs mass $m_H \approx 125$ GeV. In these cases, the invariant mass of the Higgs boson corresponds to the invariant mass of the bottom quarks, because there are no further particles emitted from the boson. Such a distribution would not occur in a realistic particle collision, as a more realistic event description requires additional components such as parton-shower matching. The events used for this plot were terminated after the hard decays stage and not evolved further. Furthermore, no mass smearing was applied. This procedure normally introduces a small off-shell variation of the momenta of intermediate particles that are treated as on-shell in the NWA. However, this would have only a small effect, since the Higgs boson has a small decay width. In the RS case, the events with negative weights (dashed line) appear at the peak. These correspond to the real subtraction events, in which no gluon is emitted. Since these events retain Born kinematics, they also only contribute to the peak. The distribution to the left of the peak originates from the real-emission (R) events, where the emitted light jet (non-b-tagged jet, originating from the gluon emission) carries away transverse momentum. Since the light jet is not clustered into the invariant mass of the two b-tagged jets, this momentum is missing from the $b\bar{b}$ -jet system, leading to a lower reconstructed invariant mass. This interpretation can be confirmed by constructing the invariant mass from the b jets together with the jet originating from the emitted gluon, i.e. by

clustering the additional jet when it is present. The resulting distribution is shown in figure 4.2 on the left. In this case, the distribution shows exclusively a single peak at the Higgs mass.

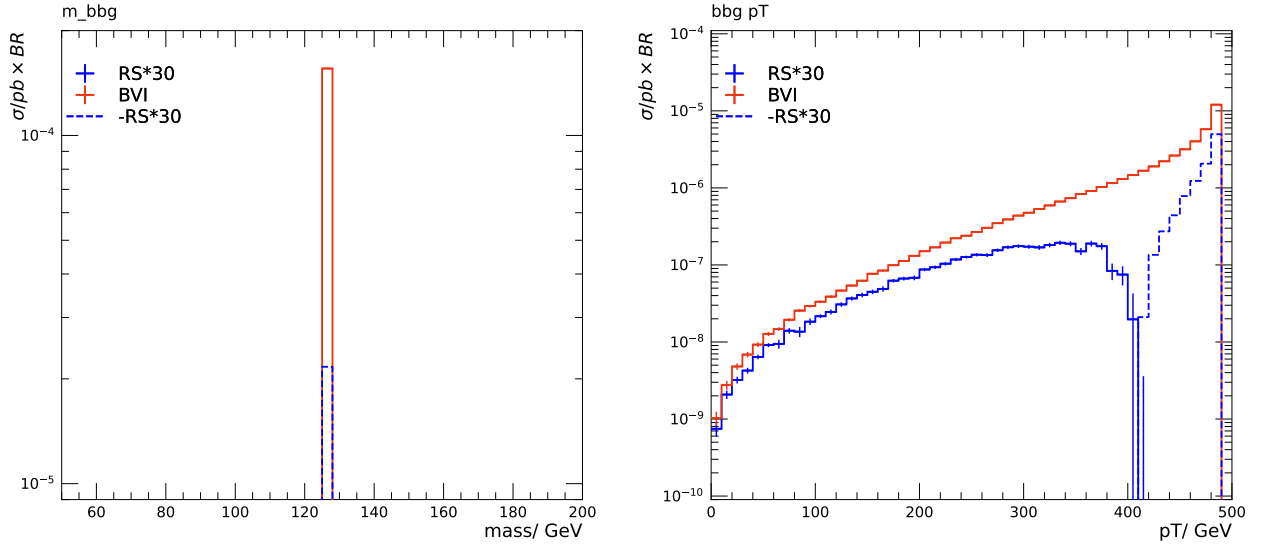


Figure 4.2: Left: invariant mass distribution of all jets for the BVI and RS channels. All events therefore produce a sharp peak at the invariant Higgs mass. Right: transverse momentum distribution of all jets (corresponds to the transverse momentum of the Higgs boson).

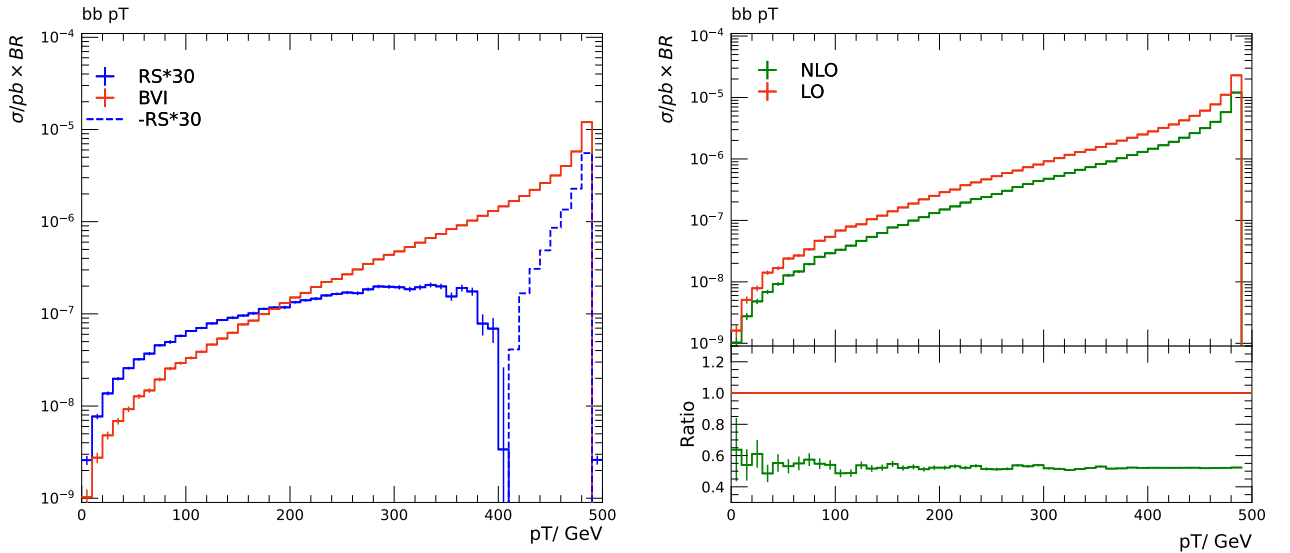


Figure 4.3: Transverse momentum distribution of the b jet pair. Left: comparison of the BVI and RS channels. Right: comparison of the LO and NLO distributions. The absolute values of the negative regions of the distribution are plotted as a dashed line. The curves are suppressed at low transverse momentum, with this effect being less pronounced for the RS events. The maximum transverse momentum reaches approximately 488 GeV.

The transverse momentum of the $b\bar{b}$ system is shown in figure 4.3. For the BVI and LO events, this observable is identical to the transverse momentum of the Higgs boson, since the Higgs

decays into the two bottom quarks without additional radiation. In this and the following plots, it can be seen that the NLO contributions differ from the LO results only by an approximately constant factor. The small fluctuations are most likely due to statistical uncertainties of the Monte Carlo simulation, since they become larger in phase space regions with fewer events. The distributions show an upper limit at approximately 490 GeV. Increasing the number of bins in figure 4.3 and zooming into the endpoint region confirms that the maximal transverse momentum for BVI events indeed occurs at 488 GeV, as shown in the zoomed view in figure 4.4.

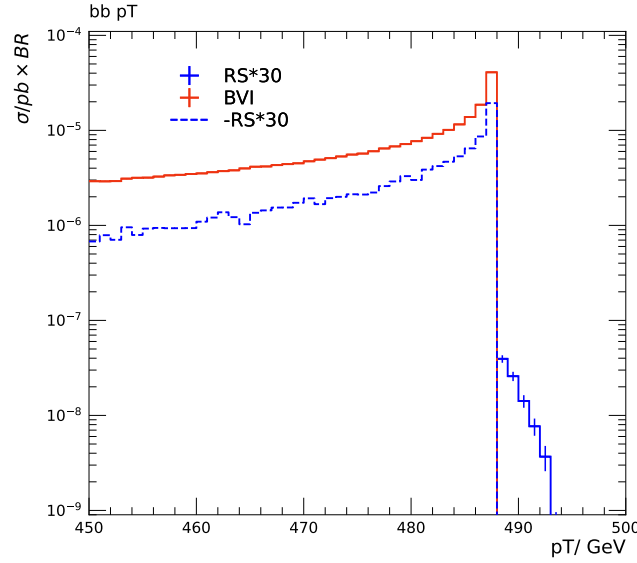


Figure 4.4: Transverse momentum distribution of the b jet pair. Comparison of the BVI and RS channels. This plot shows the high- p_T region of figure 4.3 with decreased bin width. The maximum of 488 GeV for BVI events can be clearly seen. Some real-emission events exceed this limit.

This value of 488 GeV corresponds to the kinematic configuration in which the Higgs and Z bosons are produced perpendicular to the beam axis. In this case, the transverse momentum equals the total momentum of the Higgs boson. The absolute momentum can be calculated from the two-body kinematics of the process $e^+e^- \rightarrow HZ$ [73]:

$$|\vec{p}_H| = |\vec{p}_Z| = \frac{1}{2\sqrt{\hat{s}}} \sqrt{\lambda(\hat{s}, m_H^2, m_Z^2)}, \quad (4.4)$$

where $\vec{p}_H$ and $\vec{p}_Z$ denote the momenta of the Higgs and Z boson, respectively, $\sqrt{\hat{s}}$ is the centre-of-mass energy, and λ is the Källén function as defined in equation 3.7. Evaluating this expression at a centre-of-mass energy of 1000 GeV yields

$$|\vec{p}_H| = |\vec{p}_Z| \approx 488 \text{ GeV}. \quad (4.5)$$

Consequently, the maximal transverse momentum of the $b\bar{b}$ system without gluon emission

is 488 GeV. A few real-emission events exceed this limit. In these events, the transverse momentum of the Higgs boson is close to the maximum, and the $b\bar{b}$ pair exceeds this value due to the gluon being emitted in approximately the opposite direction of the Higgs boson.

The distributions in figure 4.3 show further that the BVI, LO and NLO curves are suppressed at small transverse momentum, while this suppression becomes less pronounced for the RS results. The suppression of small transverse momenta can be understood from the polarization structure of the process. At high energies and in processes where a Z boson is produced together with a Higgs boson, such as $e^+e^- \rightarrow HZ$, the production of longitudinally polarized Z bosons dominates [74]. Because the Higgs boson does not carry spin since it is a scalar particle, the spin angular momentum of the initial state must be transferred to the Z boson. In the high-energy limit, the helicity of the electrons coincides with their chirality [20], implying that their spin is aligned along the beam axis. For a longitudinally polarized massive gauge boson, the spin is oriented perpendicular to its direction of motion in the helicity basis. A pure longitudinal polarized Z boson state is hence, due to the angular momentum conservation, only achieved if the Z boson is emitted perpendicular to the beam. As the Z boson is preferably longitudinal polarized in this process, this kinematic configuration is dominant. Since the Higgs and Z bosons are produced back-to-back, the Higgs boson is also predominantly emitted approximately perpendicular to the beam axis. This leads to configurations with large transverse momentum, while configurations with small transverse momentum are suppressed. This behaviour is reflected in the transverse momentum of the $b\bar{b}$ system, which coincides with the Higgs transverse momentum for BVI and LO events, and can also be observed in the Z transverse momentum distribution shown in figure 4.7. For the RS contribution, the transverse momentum of the $b\bar{b}$ system no longer coincides with the transverse momentum of the Higgs boson in all events. In the real-emission events, the additional jet allows momenta to be balanced differently, such that configurations with smaller transverse momenta become less suppressed due to recoil against the emitted parton. The dashed line in figure 4.3 shows that for the (negative) S events, the $b\bar{b}$ have also preferentially high transverse momenta, since no additional jet carries away momentum. This interpretation is supported by constructing the transverse momentum distribution of all jets clustered, as shown in figure 4.2 on the right side. Here, the transverse momentum coincides with the Higgs p_T in all events. In this case, the low- p_T region is suppressed equally for RS and BVI contributions.

Figure 4.5 shows the transverse momentum distribution of the light jet. It can be seen that the majority of light jets have relatively small transverse momenta. This behaviour can be attributed to the fact that the emitted gluons are predominantly soft, and therefore typically carry only a small fraction of the available momentum.

The distribution of the transverse momentum of the single b-tagged jets shown in figure 4.6 indicates that the RS contribution is slightly reduced in the intermediate p_T region and

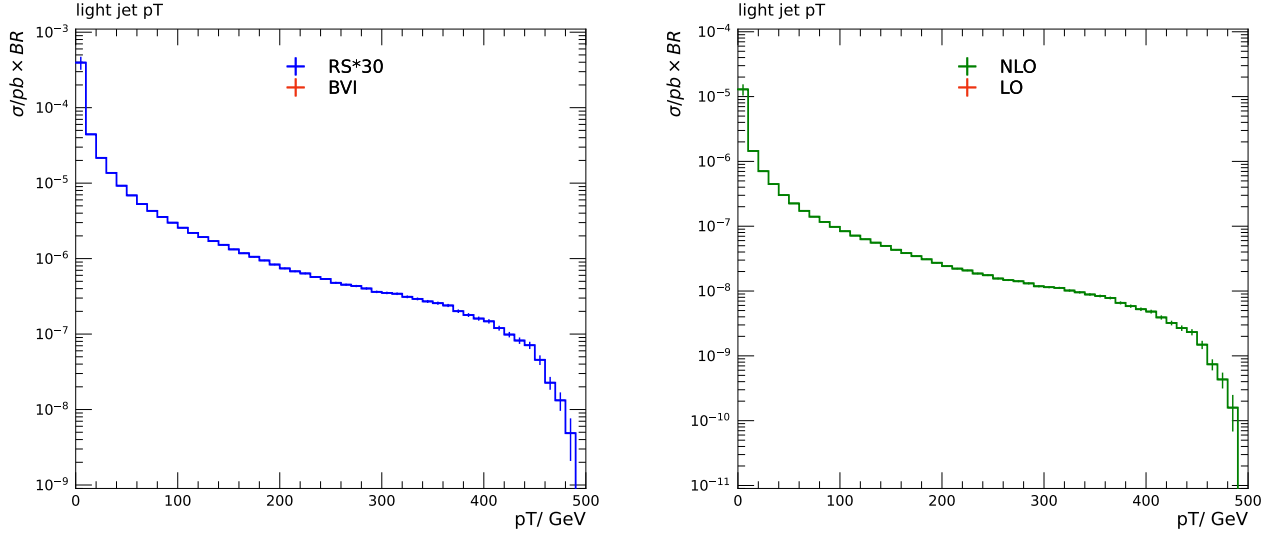


Figure 4.5: Transverse momentum distributions of the light (non-b tagged) jet. Left: comparison of the BVI and RS channels. Right: comparison of the LO and NLO distributions. Light jets are preferentially produced at low transverse momentum. They arise only in R events due to gluon radiation.

enhanced at low transverse momentum. This behaviour can be attributed to the real gluon emission in the decay process $H \rightarrow b\bar{b}g$. If a gluon is emitted, part of the Higgs momentum is carried away by the gluon, which reduces the transverse momentum of the b-tagged jets. As a result, events that would populate the intermediate p_T region at LO are shifted towards smaller transverse momenta of the reconstructed b jets. Furthermore, a small number of jets with transverse momenta exceeding 488 GeV are observed. While this value corresponds to the maximal transverse momentum of the Higgs boson in the production process $e^+e^- \rightarrow HZ$, the transverse momentum of an individual b-tagged jet is not bounded by this value. If one of the bottom quarks is emitted approximately in the direction of the Higgs momentum, it can carry a larger transverse momentum than the Higgs itself, while the second bottom quark is emitted in the opposite direction such that overall momentum conservation is preserved.

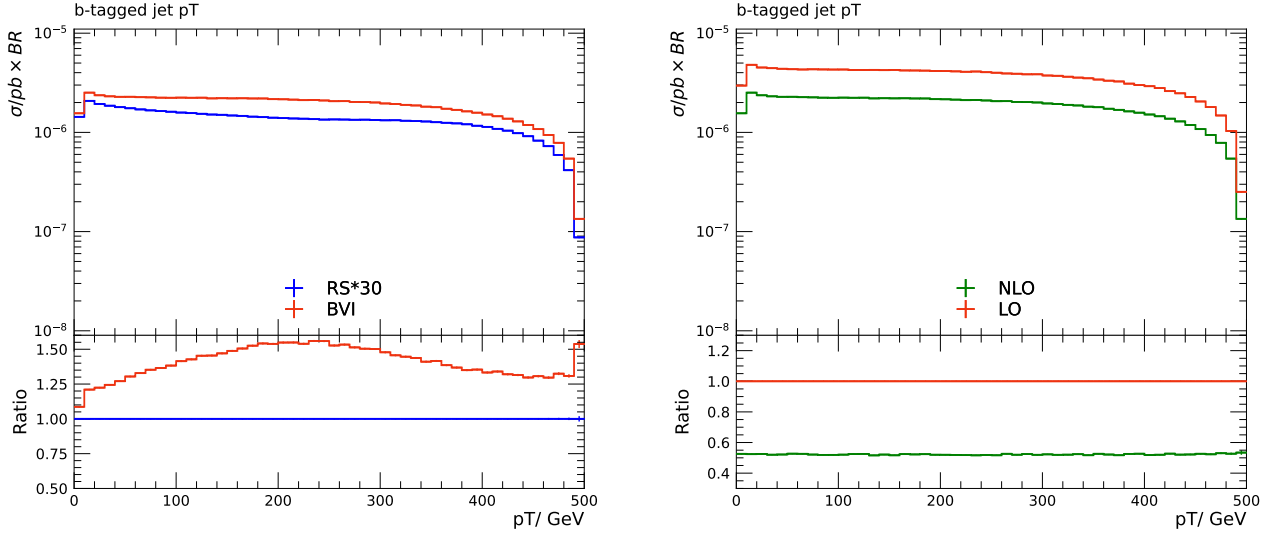


Figure 4.6: Transverse momentum distributions of the single b-tagged jets. Left: comparison of the BVI and RS channels. Right: comparison of the LO and NLO distributions. The distribution shows a slight enhancement at low p_T and reduction at intermediate p_T for RS events compared to BVI events. In addition, events with transverse momenta above 490 GeV are observed.

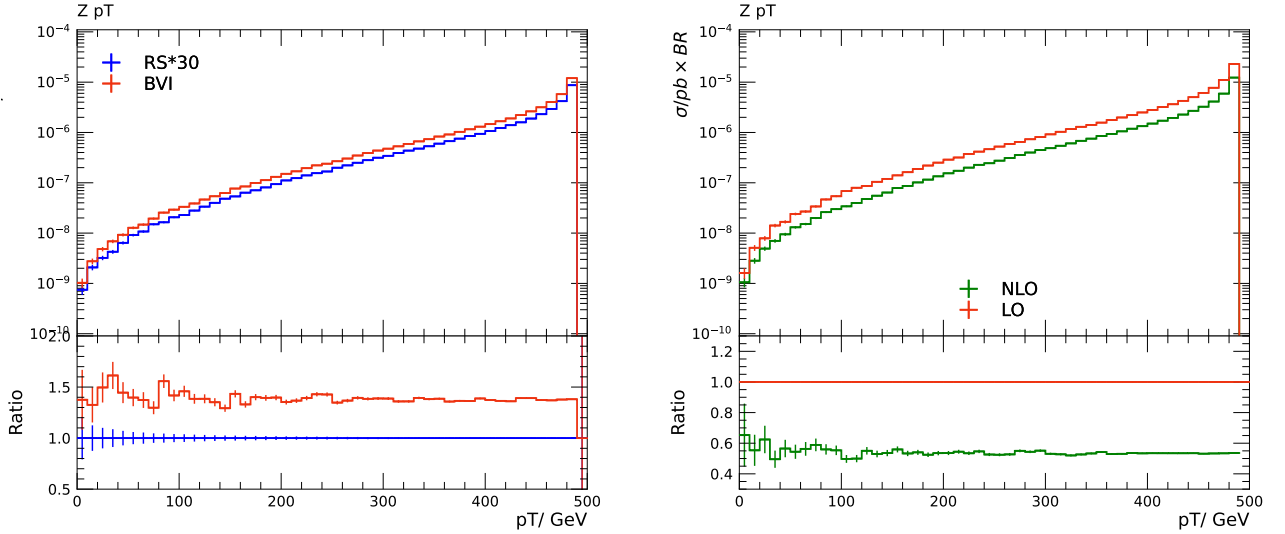


Figure 4.7: Transverse momentum distributions of the Z boson. Left: comparison of the BVI and RS channels. Right: comparison of the LO and NLO distributions. Low- p_T regions are suppressed and no events are observed with a transverse momentum exceeding 490 GeV. The RS and BVI distributions and likewise the LO and NLO distributions differ only by a constant scaling factor.

Since the NLO corrections are applied only to the Higgs decay, the kinematics of the Z boson are not affected. Consequently, the RS and BVI contributions to the transverse momentum distribution of the Z boson differ also only by an overall factor originating from the different total cross sections of the underlying processes. This behaviour can be observed in figure 4.7. The ratio oscillations visible in the low- p_T region are again due to limited Monte Carlo statistics

in this phase space region. Furthermore, the upper limit of 488 GeV can be observed again.

5 Summary and Outlook

In this work, the NLO decay of $H \rightarrow b\bar{b}$ was implemented within the existing hard decay framework in SHERPA. The NLO contributions were separated into two decay channels. The first channel contains the Born contribution $\mathcal{B}$ together with the virtual correction $\mathcal{V}$ and the corresponding integrated subtraction term $\mathcal{I}^S$, while the second channel includes the real correction $\mathcal{R}$ with gluon emission as well as the associated real subtraction term $\mathcal{D}^S$. The subtraction terms were introduced to remove the IR divergences that arise in the real contribution from soft gluon emission and in the virtual contribution from the corresponding regions of the loop momentum integration. Since the bottom quarks were treated as massive particles in this work, only soft and no collinear IR divergences occur.

During the implementation of the $H \rightarrow b\bar{b}$ decay at NLO, the hard decay framework was extended by introducing a general structure that allows the treatment of NLO corrections for hard decays. The developed architecture supports the calculation of different NLO decay channels, the generation of subevents, and the integration of subtraction terms into the existing workflow of SHERPA. In this way, the framework allows straightforward extensions to other decays at NLO. In particular, the implemented structure for the real subtraction terms $\mathcal{D}^S$ for decays with massive quarks in the final state can be used directly for other processes. For decays with different final-state particles, only minor adjustments are required. The treatment of the real emission contribution $\mathcal{R}$ was also implemented in a general way such that it can be applied to arbitrary real NLO contributions. More substantial modifications would be required for the virtual correction $\mathcal{V}$, as its implementation depends on the specific Feynman rules of the decay under consideration. The explicit structure of the virtual contributions would change accordingly for other decaying particles. To validate the implementation, the individual contributions were compared with results obtained from the matrix element generator COMIX. In this comparison it was found that the virtual correction $\mathcal{V}$ and the integrated subtraction term $\mathcal{I}^S$ do not agree yet with the reference results. One possible strategy for identifying the source of this discrepancy would be to compare the calculation of the Passarino–Veltman coefficients with an alternative loop-integral library, such as Collier [75]. In this way it would be possible to verify whether the configuration of LoopTools was used correctly. Alternatively, the current calculation implemented in Python and SHERPA could be reproduced independently using Mathematica [76] in order to identify potential implementation errors. In order to nevertheless

obtain first physical results with the new framework, the currently inconsistent contributions were replaced in the final evaluation by the corresponding reference results obtained from COMIX.

The resulting analysis shows that the NLO implementation reduces the partial decay width of the channel $H \rightarrow b\bar{b}$ compared to the LO calculation, leading to better agreement with experimentally measured reference values reported in the literature. In the results considered here, the contribution from the real-emission channel was significantly smaller than the contribution from the channel containing the Born, virtual, and integrated subtraction terms.

In summary, the structure to include NLO hard decays within the existing framework was successfully implemented. The treatment of the real contributions and the associated subtraction terms is fully functional for the decay $H \rightarrow b\bar{b}$. However, the complete validation of the virtual correction $\mathcal{V}$ and the integrated subtraction term $\mathcal{I}^S$ still has to be carried out. Future work may therefore focus both on the final validation of the virtual contributions and on extending the framework to additional NLO decay channels. In this context, the correct execution of the spin-correlation algorithm for the newly implemented decays also needs to be tested. Furthermore, the framework can be extended by matching the implemented NLO hard decays to a parton shower, as briefly discussed in section 2.3.1.

A LoopTools Usage and Script

The structure of the integrals calculated by LoopTools is shown in figure A.1.

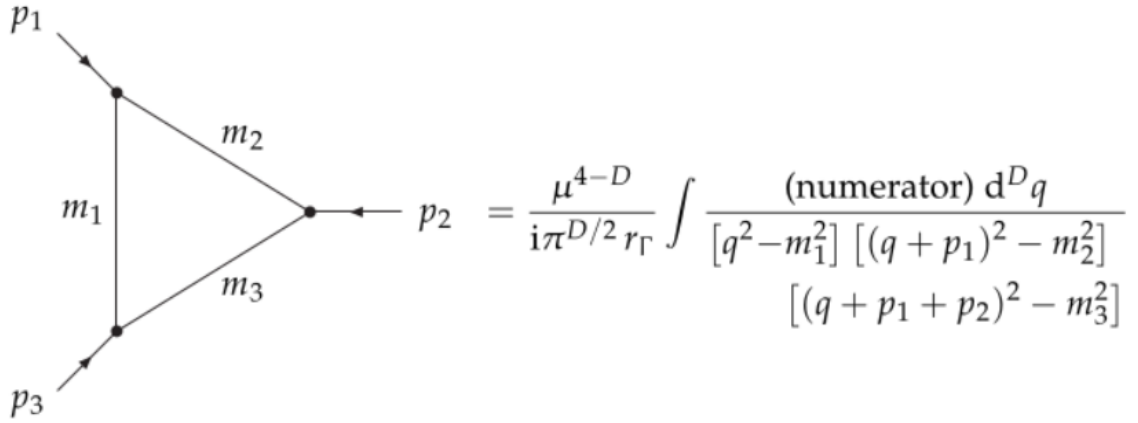


Figure A.1: Structure of the integrals calculated by LoopTools. Picture taken from [70].

The order of the input variables that LoopTools takes is:

$$a = p_1^2, p_2^2, (p_1 + p_2)^2, m_1^2, m_2^2, m_3^2 \quad (\text{A.1})$$

Comparing that to equation 3.19 shows that in the case of the virtual $H \rightarrow b\bar{b}$ diagram, one has to insert the values

$$a = q_1^2, p_2^2, q_2^2, 0, m_b^2, m_b^2, \quad (\text{A.2})$$

where p_2 is the Higgs momentum as one can see in figure 3.3 and q_1^2 and q_2^2 are the squared invariant masses of the bottom and antibottom quarks.

In the LoopTools script, the Higgs and bottom masses are taken from SHERPA (the values that SHERPA uses in the LO calculation were taken). The results are shown below in table A.1. They are complex numbers. The real part corresponds to the first entry and the imaginary part the second entry.

```

1 #include <iostream>
2 #include "clooptools.h"
3

```

```

4 int main() {
5     ltini();
6     setmudim(125.09);          // scale = Higgs mass
7
8     double p1sq = 24.2064;    // squared Bottom momentum
9     double p2sq = 15647.5081; // squared Higgs momentum
10    double p3sq = 24.2064;
11    double m1sq = 0.0;
12    double m2sq = 24.2064;
13    double m3sq = 24.2064;
14
15    //switch off that the UV divergence is included in the divergent terms
16    setuvdiv(0); // 0 if UV should be switched off, 1 to switch UV on.
17
18    setlambda(0); // for calculation of finite part
19    std::cout << "finite" << std::endl;
20    //scalar three-point function
21    std::cout << "C0 = " << C0(p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::endl;
22    // other coefficients:
23    std::cout << "C1 = " << C0i(cc1, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
24    endl;
25    std::cout << "C2 = " << C0i(cc2, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
26    endl;
27    std::cout << "C00 = " << C0i(cc00, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
28    endl;
29    std::cout << "C11 = " << C0i(cc11, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
30    endl;
31    std::cout << "C22 = " << C0i(cc22, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
32    endl;
33    std::cout << "C12 = " << C0i(cc12, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
34    endl;
35
36    setlambda(-1); // for calculation of 1/epsilon part
37    std::cout << "e~-1" << std::endl;
38    //scalar three-point function
39    std::cout << "C0 = " << C0(p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::endl;
40    // other coefficients:
41    std::cout << "C1 = " << C0i(cc1, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
42    endl;
43    std::cout << "C2 = " << C0i(cc2, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
44    endl;

```

```

37     std::cout << "C00 = " << C0i(cc00, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
38     std::cout << "C11 = " << C0i(cc11, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
39     std::cout << "C22 = " << C0i(cc22, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
40     std::cout << "C12 = " << C0i(cc12, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
41
42     setlambda(-2); // for calculation of 1/epsilon^2 part
43     std::cout << "e^-2" << std::endl;
44     //scalar three-point function
45     std::cout << "C0 = " << C0(p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::endl;
46     // other coefficients:
47     std::cout << "C1 = " << C0i(cc1, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
48     std::cout << "C2 = " << C0i(cc2, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
49     std::cout << "C00 = " << C0i(cc00, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
50     std::cout << "C11 = " << C0i(cc11, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
51     std::cout << "C22 = " << C0i(cc22, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
52     std::cout << "C12 = " << C0i(cc12, p1sq, p2sq, p3sq, m1sq, m2sq, m3sq) << std::
endl;
53
54     ltexi();
55     return 0;
56 }

```

Coefficient	Finite	$1/\epsilon$	$1/\epsilon^2$
C_0	(0.000236753, -0.000971302)	(-0.000414664, 0.000201397)	(0, 0)
C_1	(-0.000414664, 0.000201397)	(0, 0)	(0, 0)
C_2	(-0.000414664, 0.000201397)	(0, 0)	(0, 0)
C_{00}	(-0.451472, 0.782964)	(0, 0)	(0, 0)
C_{11}	(0.000174736, -0.000100387)	(0, 0)	(0, 0)
C_{22}	(0.000174736, -0.000100387)	(0, 0)	(0, 0)
C_{12}	(3.25954e-05, -3.11557e-07)	(0, 0)	(0, 0)

Table A.1: Finite and divergent coefficients C_i from the loop calculation.

B Python Script

The Python script of the calculation of the momentum-dependent 4x4 matrix

$$\gamma^\mu \cdot integral \cdot \gamma_\mu \quad (B.1)$$

from the virtual NLO calculation of $H \rightarrow b\bar{b}$ is shown below.

The integral is defined as

$$\int \frac{d^n k}{(2\pi)^n} \frac{\mu^{2\epsilon} [k + q_2 + m_b][k + q_1 + m_b]}{k^2 [(k + q_1)^2 - m_b^2][(k + q_2)^2 - m_b^2]}, \quad (B.2)$$

simplified by Veltman-Passarino reduction to

$$C_{\mu\nu}^{LT} \gamma^\mu \gamma^\nu + C_\mu^{LT} \gamma^\mu (q_1 + q_1 + 2m_b) + C_0^{LT} (q_1 q_2 + q_1 m_b + q_2 m_b + m_b^2). \quad (B.3)$$

For the derivation, see section 3.1.3.

The gamma matrices are implemented in Weyl basis [77].

```
1 from sympy import symbols, Matrix, I, diag, eye, simplify, expand, ccode, re, im
2
3 # C-coefficients from LoopTools
4 # Finite:
5 c0 = complex(0.000236753, -0.000971302)
6 c1 = complex(-0.000414664, 0.000201397)
7 c2 = complex(-0.000414664, 0.000201397)
8 c00 = complex(-0.451472, 0.782964)
9 c11 = complex(0.000174736, -0.000100387)
10 c22 = complex(0.000174736, -0.000100387)
11 c12 = complex(3.25955e-05, -3.11563e-07)
12
13 #e^-1
14 c0_e = complex(-0.000414664, 0.000201397)
15 c1_e = complex(0, 0)
16 c2_e = complex(0, 0)
17 c00_e = complex(0, 0)
```

```

18 c11_e = complex(0,0)
19 c22_e = complex(0,0)
20 c12_e = complex(0,0)
21
22 #e^-2
23 c0_e2 = complex(0,0)
24 c1_e2 = complex(0,0)
25 c2_e2 = complex(0,0)
26 c00_e2 = complex(0,0)
27 c11_e2 = complex(0,0)
28 c22_e2 = complex(0,0)
29 c12_e2 = complex(0,0)
30
31 q1_0, q1_1, q1_2, q1_3 = symbols("q1_0 q1_1 q1_2 q1_3", real=True) # 4 components of
    q1
32 q2_0, q2_1, q2_2, q2_3 = symbols("q2_0 q2_1 q2_2 q2_3", real=True) # 4 components of
    q2
33
34 m_b = 4.919999999999999 # bottom mass (the same that is used in the LO calculation
    in Sherpa)
35
36 # Weyl gamma matrices:
37 gamma0 = Matrix([[0,0,1,0],
38                 [0,0,0,1],
39                 [1,0,0,0],
40                 [0,1,0,0]])
41
42 gamma1 = Matrix([[0,0,0,1],
43                 [0,0,1,0],
44                 [0,-1,0,0],
45                 [-1,0,0,0]])
46
47 gamma2 = Matrix([[0,0,0,-I],
48                 [0,0, I, 0],
49                 [0, I, 0, 0],
50                 [-I,0, 0, 0]])
51
52 gamma3 = Matrix([[0,0, 1, 0],
53                 [0,0, 0,-1],
54                 [-1,0,0, 0],
55                 [0, 1,0, 0]])

```

```

56
57 gamma_up = [gamma0, gamma1, gamma2, gamma3] # gammas with upper indices
58 # transfer them to lower index:
59 eta = diag(1, -1, -1, -1) # minkowski metric
60
61 # --- Gammas with lower index: gamma_mu = g_{mu nu} gamma^nu ---
62 gamma_down = [eta[i,i] * gamma_up[i] for i in range(4)]
63
64 slash_q1 = gamma_down[0]*q1_0 + gamma_down[1]*q1_1 + gamma_down[2]*q1_2 + gamma_down
    [3]*q1_3
65 slash_q2 = gamma_down[0]*q2_0 + gamma_down[1]*q2_1 + gamma_down[2]*q2_2 + gamma_down
    [3]*q2_3
66
67 """ Composition of scalar, linear, and quadratic integrals, including prefactors.
68     The calculations are done for the finite, 1/epsilon and 1/epsilon^2 terms.
69 """
70 # scalar:
71 sc_integral = c0 * (slash_q1 * slash_q2 + (slash_q1 + slash_q2) * m_b + m_b**2 * eye
    (4))
72 sc_integral_e = c0_e * (slash_q1 * slash_q2 + (slash_q1 + slash_q2) * m_b + m_b**2 *
    eye(4))
73 sc_integral_e2 = c0_e2 * (slash_q1 * slash_q2 + (slash_q1 + slash_q2) * m_b + m_b**2
    * eye(4))
74
75 # linear:
76 lin_integral = (c1 * slash_q1 + c2 * slash_q2) * (slash_q1 + slash_q1 + 2*m_b * eye
    (4))
77 lin_integral_e = (c1_e * slash_q1 + c2_e * slash_q2) * (slash_q1 + slash_q1 + 2*m_b
    * eye(4))
78 lin_integral_e2 = (c1_e2 * slash_q1 + c2_e2 * slash_q2) * (slash_q1 + slash_q1 + 2*
    m_b * eye(4))
79
80 # quadratic:
81 Z = Matrix.zeros(4, 4)
82 gamma_mu_gamma_up = sum((gamma_down[i] * gamma_up[i] for i in range(4)), Z)
83 quad_integral = c00 * gamma_mu_gamma_up + c11*slash_q1*slash_q1 + c22*slash_q2*
    slash_q2 + 2 * c12 * slash_q1*slash_q2
84 quad_integral_e = c00_e * gamma_mu_gamma_up + c11_e*slash_q1*slash_q1 + c22_e*
    slash_q2*slash_q2 + 2 * c12_e * slash_q1*slash_q2
85 quad_integral_e2 = c00_e2 * gamma_mu_gamma_up + c11_e2*slash_q1*slash_q1 + c22_e2*
    slash_q2*slash_q2 + 2 * c12_e2 * slash_q1*slash_q2

```

```

86
87 integral = sc_integral + lin_integral + quad_integral
88 integral_e = sc_integral_e + lin_integral_e + quad_integral_e
89 integral_e2 = sc_integral_e2 + lin_integral_e2 + quad_integral_e2
90
91 # final 4x4 matrix:
92 gamma_integral_gamma = sum((gamma_down[mu] * integral * gamma_up[mu] for mu in range
    (4)), Z)
93 gamma_integral_gamma_e = sum((gamma_down[mu] * integral_e * gamma_up[mu] for mu in
    range(4)), Z)
94 gamma_integral_gamma_e2 = sum((gamma_down[mu] * integral_e2 * gamma_up[mu] for mu in
    range(4)), Z)
95
96 def to_a_plus_i_b(e):
97     # simplifies expression
98     e = expand(e, complex=True)
99     # extract real and imaginary parts
100    re_part, im_part = e.as_real_imag()
101    # simplify further
102    return simplify(re_part) + I*simplify(im_part)
103
104 def to_a_plus_i_b_matrix(M):
105     # apply simplification-function to each element of the matrix
106     return M.applyfunc(to_a_plus_i_b)
107
108 result_finite = to_a_plus_i_b_matrix(gamma_integral_gamma)
109 result_e = to_a_plus_i_b_matrix(gamma_integral_gamma_e)
110 result_e2 = to_a_plus_i_b_matrix(gamma_integral_gamma_e2)
111
112
113
114 def print_cpp_complex_matrix(M, name):
115     """
116     Prints a Sympy 4x4 complex matrix in C++ format:
117
118     name = {
119         std::array<std::complex<double>, 4>{std::complex<double>(Re00, Im00), ...},
120         std::array<std::complex<double>, 4>{std::complex<double>(Re10, Im10), ...},
121         ...
122     };
123     """

```

```

124     nrows, ncols = M.shape
125     assert nrows == 4 and ncols == 4, "Matrix must be 4x4"
126
127     print(f"{name} = {{")
128     for i in range(nrows):
129         row_entries = []
130         for j in range(ncols):
131             re_ = ccode(simplify(re(M[i, j])))
132             im_ = ccode(simplify(im(M[i, j])))
133
134             # replace q1_ and q2_ by q1[] / q2[]
135             for k in range(4):
136                 re_ = re_.replace(f"q1_{k}", f"q1[{k}]").replace(f"q2_{k}", f"q2[{k}
137 ]}")
138                 im_ = im_.replace(f"q1_{k}", f"q1[{k}]").replace(f"q2_{k}", f"q2[{k}
139 ]}")
140
141             row_entries.append(f"    std::complex<double>({re_}, {im_})")
142             line = ", \n".join(row_entries)
143             print(f"std::array<std::complex<double>, 4>{{{line}}},")
144         print("};\n")
145
146 # print results
147 print("\n// === finite part ===")
148 print_cpp_complex_matrix(result_finite, "M_finite")
149
150 print("\n// === 1/epsilon part ===")
151 print_cpp_complex_matrix(result_e, "M_epsilon")
152
153 print("\n// === 1/epsilon^2 part ===")
154 print_cpp_complex_matrix(result_e2, "M_epsilon2")

```


C Run Cards

C.1 COMIX Run Card for the $\mathcal{R}$ and $\mathcal{D}^S$ Squared Matrix Elements Comparison

When writing the momenta of the hard decay implementation into the run card for the COMIX reference process, it is important to take into account that the ordering of the momenta differs. In the hard decay implementation considered here, the ordering is $H, g, b, \bar{b}$, whereas in the run card used for the COMIX reference calculation, the flavour ordering must be $H, b, \bar{b}, g$. Therefore, the momenta have to be permuted accordingly before performing the comparison. When comparing additional decay channels at NLO in the future, particular care must be taken to ensure that the particle ordering is handled consistently.

```
1 # collider setup
2 BEAM_ENERGIES: 125.09
3 # Switch off multiparton interactions
4 MI_HANDLER: None # Amisic
5 # Switch off Shower generator
6 SHOWER_GENERATOR: None
7 # no hadronisation
8 FRAGMENTATION: None
9 # switch off QED corrections
10 ME_QED: {ENABLED: false}
11 BEAM_REMNANTS: false
12 PDF_LIBRARY: None
13 # Disable QED correction module
14 YFS:
15     MODE: None
16 INTEGRATOR: Rambo
17 SCALES: VAR{sqr(125.09)} # Higgs mass
18 EVENT_GENERATION_MODE: Weighted
19 # me generator settings
20 ME_GENERATORS: [Comix]
```

```

21 RANDOM_SEED: 1234
22 COMIX_DEFAULT_GAUGE: 0
23
24 PARTICLE_DATA:
25   5:
26     Massive: true
27     Mass: 4.919999999999999
28 YUKAWA_MASSES: Fixed
29
30 PROCESSES:
31 - 25 -> 5 -5:
32   Order: {QCD: 0, EW: 1}
33   NLO_Mode: Fixed_Order
34   NLO_Part: RS # R for the calculation of the real contribution.
35   # S does not work. Calculate it as RS - R
36   NLO_Order: {QCD: 1, EW: 0}
37   ME_Generator: Comix
38   RS_ME_Generator: Comix
39
40 PS_POINT: # order of momenta: H, 5, -5, G
41 "(503.14167905501597,-216.67494129313647,421.72347253436033,-112.71789401633302)\n
    (126.82330590232542,-38.519666971070833,94.52580846126078,-75.106932510771884)\n
    (275.30872399834999,-117.13422876289502,248.02928114124941,-23.059385470486838)\n
    (101.00964915433997,-61.021045559170297,79.168382931849564,-14.551576035074156)"

```

C.2 $e^+e^- \rightarrow HZ \rightarrow b\bar{b} \mu^+\mu^-$ Event Generation

```

1 # collider setup
2 BEAMS: [11, -11]
3 BEAM_ENERGIES: [500, 500]
4 RANDOM_SEED: 4
5
6 ME_GENERATORS: Comix
7 SCALES: VAR{Abs2(p[2]+p[3])}
8 EVENTS: 200000
9 PDF_LIBRARY: None # remove electron PDF
10
11 PROCESSES:
12 - 11 -11 -> 23 25:

```

```
13     Order: {QCD: 0, EW: 2}
14
15 HARD_DECAYS:
16     Enabled: true
17     QED_Corrections: 0
18     Use_HO_SM_Widths: false
19     Channels:
20         25,21,5,-5: {Status: 2} # RS
21         25,5,-5: {Status: 2}    # BVI
22         23,13,-13: {Status: 2}
23     Mass_Smearing: 0
24
25 PARTICLE_DATA:
26     5:
27         Massive: true
28         Mass: 4.919999999999999
29         Stable: 1
30 YUKAWA_MASSES: Fixed
31
32 # Switch off Shower generator
33 SHOWER_GENERATOR: None
34
35 # Switch off multiparton interactions
36 MI_HANDLER: None # Amisic
37
38 # no hadronisation
39 FRAGMENTATION: None
40
41 # no beam remnants
42 BEAM_REMNANTS: Off
43
44 # no QED radiation
45 ME_QED: {ENABLED: false}
46 YFS:
47     MODE: None
48
49 EVENT_OUTPUT:
50     - HepMC3_Short[MyFile_Decay_Short_NLO]
```


D Rivet Analysis

```
1 #include "Rivet/Analysis.hh"
2 #include "Rivet/Projections/FinalState.hh"
3 #include "Rivet/Projections/DileptonFinder.hh"
4 #include "Rivet/Projections/PromptFinalState.hh"
5 #include "Rivet/Projections/LeptonFinder.hh"
6 #include "Rivet/Projections/MissingMomentum.hh"
7 #include "Rivet/Projections/UnstableParticles.hh"
8 #include "Rivet/Projections/FastJets.hh"
9 #include "Rivet/Math/LorentzTrans.hh"
10
11 namespace Rivet {
12     /// An MC analysis for studying the content of Z->(mumu) H(->bb) events
13     class MC_ZHBB : public Analysis {
14     public:
15         /// Constructor
16         RIVET_DEFAULT_ANALYSIS_CTOR(MC_ZHBB);
17         /// Book histograms and initialize projections before the run
18         void init() {
19             _jetptcut = getOption<double>("PTJMIN", 0.0)*GeV;
20             // Boson projections
21             Cut cut = Cuts::pT > 0*GeV;
22             DileptonFinder zeefinder(91.2*GeV, 0.2, cut && Cuts::abspid == PID::ELECTRON,
23                                     Cuts::massIn(65*GeV, 115*GeV));
24             declare(zeefinder, "ZeeFinder");
25             DileptonFinder zmmfinder(91.2*GeV, 0.2, cut && Cuts::abspid == PID::MUON,
26                                     Cuts::massIn(65*GeV, 115*GeV));
27             declare(zmmfinder, "ZmmFinder");
28
29             LeptonFinder ef(cut && Cuts::abspid == PID::ELECTRON, 0.2);
30             declare(ef, "Elecs");
31             LeptonFinder mf(cut && Cuts::abspid == PID::MUON, 0.2);
32             declare(mf, "Muons");
33             declare(MissingMomentum(), "MET");
```

```

32
33 // Jet projections
34 VetoedFinalState jfs;
35 jfs
36 .addVetoOnThisFinalState(zeefinder)
37 .addVetoOnThisFinalState(zmmfinder)
38 .addVetoOnThisFinalState(ef)
39 .addVetoOnThisFinalState(mf);
40 declare(FastJets(jfs, JetAlg::ANTIKT, 0.4), "AntiKT04");
41 declare(FastJets(jfs, JetAlg::ANTIKT, 0.5), "AntiKT05");
42 declare(FastJets(jfs, JetAlg::ANTIKT, 0.6), "AntiKT06");
43
44 /// Book histograms
45 book(_h_jet_b_jet_multiplicity, "jet_b_jet_multiplicity", 11, -0.5, 10.5);
46 book(_h_jet_b_jet_pT, "jet_b_jet_pT", 50, 0, 500);
47 book(_h_jet_notb_jet_pT, "jet_notb_jet_pT", 50, 0, 500);
48 book(_h_jet_H_mass_using_bb, "jet_H_mass_using_bb", 50, 50, 200);
49 book(_h_jet_H_pT_using_bb, "jet_H_pT_using_bb", 50, 0, 500);
50 book(_h_jet_multiplicity, "jet_multiplicity", 11, -0.5, 10.5);
51 book(_h_jet_pT, "jet_pT", 50, 0, 500);
52 book(_h_VB_mass, "Z_mass", 50, 60, 110);
53 book(_h_Z_multiplicity, "Z_multiplicity", 11, -0.5, 10.5);
54 book(_h_VB_pT, "Z_pT", 50, 0, 500);
55 }
56 /// Perform the per-event analysis
57 void analyze(const Event& event) {
58
59 // Get vector bosons -- only one is expected
60 const DileptonFinder& zeefinder = apply<DileptonFinder>(event, "ZeeFinder");
61 const DileptonFinder& zmmfinder = apply<DileptonFinder>(event, "ZmmFinder");
62 Particles vectorBosons = zeefinder.bosons() + zmmfinder.bosons();
63 const int numZ = vectorBosons.size();
64
65 const P4& pmiss = apply<MissingMomentum>(event, "MET").missingMom();
66 if (pmiss.pT() > 25*GeV) {
67     const Particles& es = apply<LeptonFinder>(event, "Elects").particles();
68     const int iefound = closestMatchIndex(es, pmiss, Kin::mass, 80.4*GeV, 60*GeV,
69     100*GeV);
70     if (iefound >= 0) {
71         const Particle we(PID::WBOSON * sign(es[iefound].charge()), es[iefound].
mom() + pmiss);

```

```

71     vectorBosons.push_back(wb);
72 }
73
74     const Particles& mus = apply<LeptonFinder>(event, "Muons").particles();
75     const int imfound = closestMatchIndex(mus, pmiss, Kin::mass, 80.4*GeV, 60*
GeV, 100*GeV);
76     if (imfound >= 0) {
77         const Particle wm(PID::WBOSON * sign(mus[imfound].charge()), mus[imfound].
mom() + pmiss);
78         vectorBosons.push_back(wm);
79     }
80 }
81 _h_Z_multiplicity->fill(numZ);
82
83     const Jets jets = apply<FastJets>(event, "AntiKT04").jetsByPt(Cuts::pT >
_jetptcut);
84     _h_jet_multiplicity->fill(jets.size());
85
86     // Identify the b-jets
87     Jets bjets;
88     for (const Jet& jet : jets) {
89         const double jetPt = jet.pT();
90         _h_jet_pT->fill(jetPt/GeV);
91
92         if (jet.bTagged() && jet.pT() > _jetptcut) {
93             bjets.push_back(jet);
94             _h_jet_b_jet_pT->fill(jetPt);
95         }
96         if (!jet.bTagged() && jet.pT() > _jetptcut){
97             _h_jet_notb_jet_pT->fill(jetPt);
98         }
99     }
100     _h_jet_b_jet_multiplicity->fill(bjets.size());
101
102     // Plot vector-boson properties
103     for (const Particle& v : vectorBosons) {
104         _h_VB_pT->fill(v.pT());
105         _h_VB_mass->fill(v.mass());
106     }
107
108     // The rest of the analysis requires at least 1 b-jet

```

```

109     if (bjets.empty()) vetoEvent;
110
111     // Construct Higgs candidates from pairs of b-jets
112     for (size_t i = 0; i < bjets.size()-1; ++i) {
113         for (size_t j = i+1; j < bjets.size(); ++j) {
114             const Jet& jet1 = bjets[i];
115             const Jet& jet2 = bjets[j];
116
117             const FourMomentum phiggs = jet1.momentum() + jet2.momentum();
118             _h_jet_H_mass_using_bb->fill(phiggs.mass());
119             _h_jet_H_pT_using_bb->fill(phiggs.pT());
120
121             for (const Particle& v : vectorBosons) {
122
123                 // Calculate boost angles
124                 const vector<double> angles = boostAngles(jet1.momentum(), jet2.momentum
125                 (), v.momentum());
126             }
127         }
128     }
129     /// Normalize histograms etc., after the run
130     void finalize() {
131         scale(_h_jet_b_jet_multiplicity, 1/sumOfWeights());
132         scale(_h_jet_b_jet_pT, 1/sumOfWeights());
133         scale(_h_jet_notb_jet_pT, 1/sumOfWeights());
134         scale(_h_jet_H_mass_using_bb, 1/sumOfWeights());
135         scale(_h_jet_H_pT_using_bb, 1/sumOfWeights());
136         scale(_h_jet_multiplicity, 1/sumOfWeights());
137         scale(_h_jet_pT, 1/sumOfWeights());
138         scale(_h_VB_mass, 1/sumOfWeights());
139         scale(_h_Z_multiplicity, 1/sumOfWeights());
140         scale(_h_VB_pT, 1/sumOfWeights());
141     }
142
143     /// This should take in the four-momenta of two b's (jets/hadrons) and a vector
144     boson, for the process VB*->VBH with H->bb
145     /// It should return the smallest angle between the virtual vector boson and one
146     of the b's, in the rest frame of the Higgs boson.
147     /// It should also return (as the second element of the vector) the angle
148     between the b's, in the rest frame of the Higgs boson.

```

```

146     vector<double> boostAngles(const FourMomentum& b1, const FourMomentum& b2, const
147         FourMomentum& vb) {
148         const FourMomentum higgsMomentum = b1 + b2;
149         const FourMomentum virtualVBMomentum = higgsMomentum + vb;
150         const LorentzTransform lt = LorentzTransform::mkFrameTransformFromBeta(
151             higgsMomentum.betaVec());
152
153         const FourMomentum virtualVBMomentumBOOSTED = lt.transform(virtualVBMomentum);
154         const FourMomentum b1BOOSTED = lt.transform(b1);
155         const FourMomentum b2BOOSTED = lt.transform(b2);
156
157         const double angle1 = b1BOOSTED.angle(virtualVBMomentumBOOSTED);
158         const double angle2 = b2BOOSTED.angle(virtualVBMomentumBOOSTED);
159
160         const double anglebb = mapAngle0ToPi(b1BOOSTED.angle(b2BOOSTED));
161
162         vector<double> rtn;
163         rtn.push_back(angle1 < angle2 ? angle1 : angle2);
164         rtn.push_back(anglebb);
165         return rtn;
166     }
167
168 private:
169     // Jet pT cut
170     double _jetptcut = 0*GeV;
171     Histo1DPtr _h_Z_multiplicity;
172     Histo1DPtr _h_jet_b_jet_multiplicity, _h_jet_b_jet_pT, _h_jet_notb_jet_pT;
173     Histo1DPtr _h_jet_H_mass_using_bb, _h_jet_H_pT_using_bb;
174     Histo1DPtr _h_jet_multiplicity, _h_jet_pT;
175     Histo1DPtr _h_VB_mass, _h_VB_pT;
176 };
177 RIVET_DECLARE_PLUGIN(MC_ZHBB);
178 }

```


Acronyms

IR infrared

ISR initial-state radiation

LO leading order

NLO next-to-leading order

NNLO next-to-next-to-leading order

NWA narrow-width approximation

PDFs parton distribution functions

PDG particle data group

QCD quantum chromodynamics

QED quantum electrodynamics

SM standard model

UV ultraviolet

VEV vacuum expectation value

List of Tables

3.1	$K_{j j_n i'_n}^{i i_n j'_n}$ factors for the singlet, octet and mix configurations for both real diagrams $n = 1, 2$	31
3.2	Ordering of the helicity amplitudes in the <code>Comix1to2</code> object on the example $H \rightarrow f \bar{f}$	42
4.1	Total	width
	Γ_H^{total}	
	of the Higgs boson, partial widths Γ and branching ratios for the $H \rightarrow b \bar{b}$ channel, BR for PDG values [3], LO and NLO hard decay implementations.	54
4.2	Partial widths and branching ratios for the BVI and RS $H \rightarrow b \bar{b}$ decay channels, calculated with SHERPA's hard decay framework.	55
A.1	Finite and divergent coefficients C_i from the loop calculation.	68

List of Figures

2.1	Born (LO) Feynman diagram for the decay $H \rightarrow q\bar{q}$	15
2.2	Virtual Feynman diagram for the decay $H \rightarrow q\bar{q}$	15
2.3	Real emission Feynman diagrams for the decay $H \rightarrow q\bar{q}$	16
2.4	Overview of a exemplary SHERPA event of a proton-proton collision with an intermediate Higgs boson and top quark production.	18
3.1	Graphical representation of the dipole term $\mathcal{D}_{ij,k}$ for final state partons.	26
3.2	Possible colour configurations for the real Feynman diagram where the gluon is emitted from the quark ($n = 1$).	30
3.3	Virtual $H \rightarrow b\bar{b}$ Feynman diagram with all momenta labelled as outgoing.	31
3.4	Overview of classes relevant for the hard decay calculations and their connection in SHERPA.	40
3.5	Schematic Feynman diagrams already implemented in the hard decay framework via <code>Spin_Amplitudes</code> subclasses.	41
3.6	Unified Modelling Language (UML) class diagram of the <code>Spin_Amplitudes</code> class and subclasses.	41
3.7	UML diagram of the <code>Decay_Channel</code> class.	44
3.8	Sequence Diagram for the initialization phase of the hard decays.	45
3.9	Overview of all <code>Spin_Amplitudes</code> (sub-)classes.	47
3.10	UML diagram of the <code>Decay_Channel</code> class with new methods.	51
4.1	Invariant mass distributions of the b-jet pair.	56
4.2	Invariant mass distribution and transverse momentum distribution of all jets for the BVI and RS channels.	57
4.3	Transverse momentum distribution of the b jet pair.	57
4.4	Transverse momentum distribution of the b jet pair, zoomed into the high- p_T area.	58
4.5	Transverse momentum distributions of the light (non-b tagged) jet.	60
4.6	Transverse momentum distributions of the single b-tagged jets.	61
4.7	Transverse momentum distributions of the Z boson.	61
A.1	Structure of the integrals calculated by LoopTools. Picture taken from [70].	65

Bibliography

- [1] E. Bothmann, L. Flower, C. Gütschow, S. Höche, M. Hoppe, J. Isaacson, M. Knobbe, F. Krauss, P. Meizinger, D. Napoletano, A. Price, D. Reichelt, M. Schönherr, S. Schumann, and F. Siegert, *Event generation with sherpa 3*, Oct. 29, 2024.
- [2] T. Gleisberg and S. Hoeche, “Comix, a new matrix element generator”, *Journal of High Energy Physics* **2008**, 039 (2008).
- [3] Navas et al. (Particle Data Group), “Gauge and higgs bosons”, *Phys. Rev. D* **110**, 030001.
- [4] CMS Collaboration, “Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC”, *Physics Letters B* **716**, 30 (2012).
- [5] ATLAS Collaboration, “Observation of a new particle in the search for the standard model higgs boson with the ATLAS detector at the LHC”, *Physics Letters B* **716**, 1 (2012).
- [6] CMS Collaboration, “Observation of higgs boson decay to bottom quarks”, *Physical Review Letters* **121**, 121801 (2018).
- [7] ATLAS Collaboration, “Observation of $h \rightarrow b \bar{b}$ decays and VH production with the ATLAS detector”, *Physics Letters B* **786**, 59 (2018).
- [8] M. Drees and K.-i. Hikasa, “Note on QCD corrections to hadronic higgs decay”, *Physics Letters B* **240**, 455 (1990).
- [9] E. Braaten and J. P. Leveille, *Phys. Rev. D* **22**, 715 (1980).
- [10] C. Anastasiou, F. Herzog, and A. Lazopoulos, “The fully differential decay rate of a higgs boson to bottom-quarks at NNLO in QCD”, *Journal of High Energy Physics* **2012**, 35 (2012).
- [11] V. D. Duca, C. Duhr, G. Somogyi, F. Tramontano, and Z. Trocsanyi, “Higgs boson decay into b-quarks at NNLO accuracy”, *Journal of High Energy Physics* **2015**, 36 (2015).
- [12] R. Mondini, M. Schiavi, and C. Williams, *N3 LO predictions for the decay of the higgs boson to bottom quarks*, Apr. 29, 2019.
- [13] R. Mondini and C. Williams, “H to bbj at next-to-next-to-leading order accuracy”, *Journal of High Energy Physics* **2019**, 120 (2019).

- [14] A. Behring, K. Melnikov, I. Novikov, and G. Zanderighi, *Parton-shower and fixed-order QCD effects in higgs-boson production in weak-boson fusion and its decays to bottom quarks*, July 2, 2025.
- [15] S. Frixione and B. R. Webber, “Matching NLO QCD computations and parton shower simulations”, *Journal of High Energy Physics* **2002**, 029 (2002).
- [16] P. Nason, *A new method for combining NLO QCD with shower monte carlo algorithms - IOPscience*, <https://iopscience.iop.org/article/10.1088/1126-6708/2004/11/040> (visited on 02/17/2026).
- [17] S. Frixione, P. Nason, and C. Oleari, “Matching NLO QCD computations with parton shower simulations: the POWHEG method”, *Journal of High Energy Physics* **2007**, 070 (2007).
- [18] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H.-S. Shao, T. Stelzer, P. Torrielli, and M. Zaro, “The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations”, *Journal of High Energy Physics* **2014**, 79 (2014).
- [19] J. Bellm, S. Gieseke, D. Grellscheid, S. Plätzer, M. Rauch, C. Reuschle, P. Richardson, P. Schichtel, M. H. Seymour, A. Siódmok, A. Wilcock, N. Fischer, M. A. Harrendorf, G. Nail, A. Papaefstathiou, and D. Rauch, “Herwig 7.0 / herwig++ 3.0 release note”, *The European Physical Journal C* **76**, 196 (2016).
- [20] M. Thomson, *Modern particle physics* (Cambridge University Press, 2013).
- [21] C. Quigg, *Gauge theories of the strong, weak, and electromagnetic interactions* (Benjamin-Cummings Publishing Company, 1983).
- [22] O. Philipsed, *Quantenfeldtheorie und das standardmodell der teilchenphysik, eine einföhrung* (Springer-Verlag GmbH Deutschland, 2018).
- [23] M. Schwartz, *Quantum field theory and the standard model* (Cambridge University Press, 2021).
- [24] M. E. Peskin and D. V. Schroeder, *An introduction to quantum field theory* (Westview Press, 1995).
- [25] D. Griffiths, *Introduction to elementary particles* (John Wiley Sons, 2008).
- [26] C. Weinheimer and K. Zuber, “Neutrino masses”, *Annalen der Physik* **525**, 565 (2013).
- [27] G. Bertone, D. Hooper, and J. Silk, “Particle dark matter: evidence, candidates and constraints”, *Physics Reports* **405**, 279 (2005).
- [28] U. Sarkar, *Particle and astroparticle physics*, Series in high energy physics, cosmology, and gravitation (Taylor & Francis, New York, 2008), 524 pp.

- [29] E. Noether and M. A. Tavel, “Invariant variation problems”, *Transport Theory and Statistical Physics* **1**, 186 (1971).
- [30] A. Salam, “Elementary particle theory”, Almquist and Wiksell, Stockholm (1968).
- [31] S. Weinberg, “A model of leptons”, *Phys. Rev. Lett.* **19** (1967) 1264.
- [32] P. W. Higgs, “Broken symmetries and the masses of gauge bosons”, *Physical Review Letters* **13**, 508 (1964).
- [33] F. Englert and R. Brout, “Broken symmetry and the mass of gauge vector mesons”, *Physical Review Letters* **13**, 321 (1964).
- [34] P. Minkowski, “ $\mu \rightarrow e\gamma$ at a rate of one out of 109 muon decays?”, *Physics Letters B* **67**, 421 (1977).
- [35] D. d’Enterria et al., “The strong coupling constant: state of the art and the decade ahead”, *J. Phys. G* **51**, [_eprint: 2203.08271, 090501](#) (2024).
- [36] S. Catani, S. Dittmaier, M. H. Seymour, and Z. Trocsanyi, “The dipole formalism for next-to-leading order QCD calculations with massive partons”, *Nuclear Physics B* **627**, 189 (2002).
- [37] A. Buckley, J. Butterworth, S. Gieseke, D. Grellscheid, S. Hoche, H. Hoeth, F. Krauss, L. Lonnblad, E. Nurse, P. Richardson, S. Schumann, M. H. Seymour, T. Sjostrand, P. Skands, and B. Webber, “General-purpose event generators for LHC physics”, *Physics Reports* **504**, 145 (2011).
- [38] M. Stahlhofen, “NNLO electron structure functions (PDFs) from SCET”, *Journal of High Energy Physics* **2025**, 171 (2025).
- [39] G. Cowan, “From amplitudes to cross sections and decay rates”, (2023).
- [40] T. Kinoshita, “Mass singularities of feynman amplitudes”, *Journal of Mathematical Physics* **3**, 650 (1962).
- [41] T. D. Lee and M. Nauenberg, “Degenerate systems and mass singularities”, *Physical Review* **133**, B1549 (1964).
- [42] G. Kramer and B. Lampe, “Jet cross sections in e+e annihilation”, *Fortschritte der Physik/Progress of Physics* **37**, [eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/prop.2190370302](https://onlinelibrary.wiley.com/doi/pdf/10.1002/prop.2190370302), 161 (1989).
- [43] R. K. Ellis, D. A. Ross, and A. E. Terrano, “The perturbative calculation of jet structure in e+e annihilation”, *Nuclear Physics B* **178**, 421 (1981).
- [44] S. Catani and M. H. Seymour, “A general algorithm for calculating jet cross sections in NLO QCD”, *Nuclear Physics B* **485**, 291 (1997).
- [45] G. ’t Hooft and M. Veltman, “Regularization and renormalization of gauge fields”, *Nuclear Physics B* **44**, 189 (1972).

- [46] S. Alioli, P. Nason, C. Oleari, and E. Re, “A general framework for implementing NLO calculations in shower monte carlo programs: the POWHEG BOX”, *Journal of High Energy Physics* **2010**, 43 (2010).
- [47] E. Bothmann, G. S. Chahal, S. Höche, J. Krause, F. Krauss, S. Kuttimalai, S. Liebschner, D. Napoletano, M. Schönherr, H. Schulz, S. Schumann, and F. Siegert, “Event generation with sherpa 2.2”, *SciPost Physics* **7**, 034 (2019).
- [48] S. Höche, F. Krauss, S. Schumann, and F. Siegert, “QCD matrix elements and truncated showers”, *Journal of High Energy Physics* **2009**, 053 (2009).
- [49] M. H. Seymour and M. Marx, *Monte carlo event generators*, Apr. 24, 2013.
- [50] J. Krause, “Simulation of particle collisions for the large hadron collider at the intersection of theory and experiment”, PhD thesis (TU Dresden, 2019).
- [51] S. Catani, F. Krauss, R. Kuhn, and B. R. Webber, “QCD matrix elements + parton showers”, *Journal of High Energy Physics* **11**, 10.1088/1126-6708/2001/11/063 (2001).
- [52] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten, S. Mrenna, S. Prestel, C. O. Rasmussen, and P. Z. Skands, “An introduction to PYTHIA 8.2”, *Computer Physics Communications* **191**, 159 (2015).
- [53] S. Hoeche, F. Krauss, M. Schonherr, and F. Siegert, “A critical appraisal of NLO+PS matching methods”, *Journal of High Energy Physics* **2012**, 49 (2012).
- [54] S. Hoeche, F. Krauss, M. Schonherr, and F. Siegert, “W+n-jet predictions at the large hadron collider at next-to-leading order matched with a parton shower”, *Physical Review Letters* **110**, 052001 (2013).
- [55] *Private conversation with Marek Schönherr.*
- [56] S. Hoeche, F. Krauss, S. Schumann, and F. Siegert, “QCD matrix elements and truncated showers”, *Journal of High Energy Physics* **2009**, 053 (2009).
- [57] B. Andersson, S. Mohanty, and F. Soderberg, “The lund fragmentation process for a multi-gluon string according to the area law”, *The European Physical Journal C* **21**, 631 (2001).
- [58] S. Gieseke, F. Loshaj, and P. Kirchgaesser, “Soft and diffractive scattering with the cluster model in herwig”, *The European Physical Journal C* **77**, 156 (2017).
- [59] C. F. Uhlemann and N. Kauer, “Narrow-width approximation accuracy”, *Nuclear Physics B* **814**, 195 (2009).
- [60] D. Berdine, N. Kauer, and D. Rainwater, “Breakdown of the narrow width approximation for new physics”, *Physical Review Letters* **99**, 111601 (2007).
- [61] G. Breit and E. Wigner, “Capture of Slow Neutrons”, *Phys. Rev.* **49**, 519 (1936).

- [62] C. Uhlemann, “Narrow-width approximation in the minimal supersymmetric standard model”, MA thesis (Bayerische Julius-Maximilians-Universität Würzburg, 2007).
- [63] D. M. Hoppe, “Implementation and phenomenology of polarized cross sections in the simulation of vector boson production with the SHERPA event generator”, MA thesis (TU Dresden, 2023).
- [64] P. Richardson, “Spin correlations in monte carlo simulations”, *Journal of High Energy Physics* **2001**, 029 (2001).
- [65] S. Actis, A. Denner, L. Hofer, J.-N. Lang, A. Scharf, and S. Uccirati, “R e c o l a—REcursive computation of one-loop amplitudes”, *Computer Physics Communications* **214**, 140 (2017).
- [66] G. Passarino and M. Veltman, “One-loop corrections for e^+e^- annihilation into $\mu^+\mu^-$ in the weinberg model”, *Nuclear Physics B* **160**, 151 (1979).
- [67] T. Hahn and M. Perez-Victoria, “Automatized one-loop calculations in 4 and d dimensions”, *Computer Physics Communications* **118**, 153 (1999).
- [68] G. Van Rossum and F. L. Drake, *Python 3 reference manual* (CreateSpace, Scotts Valley, CA, 2009).
- [69] “MS-bar renormalization scheme”, in *QCD as a theory of hadrons: from partons to confinement*, edited by S. Narison, Cambridge Monographs on Particle Physics, Nuclear Physics and Cosmology (Cambridge University Press, Cambridge, 2004), pp. 84–90.
- [70] T. Hahn, *LoopTools user’s guide*, version 2.5, Computer software (2010).
- [71] *Private conversation with Mareen Hoppe*.
- [72] C. Bierlich, A. Buckley, J. Butterworth, C. Gutschow, L. Lonnblad, T. Procter, P. Richardson, and Y. Yeh, *Robust independent validation of experiment and theory: rivet version 4 release note*, Aug. 13, 2024.
- [73] R.L. Workman et al. (Particle Data Group), *Prog. Theor. Exp. Phys.* 2022, 083C01 (2022).
- [74] L. Huang, S. D. Lane, I. M. Lewis, and Z. Liu, “Electroweak restoration at the LHC and beyond: the γh channel”, *Physical Review D* **103**, 053007 (2021).
- [75] A. Denner, S. Dittmaier, and L. Hofer, *COLLIER – a fortran-library for one-loop integrals*, July 1, 2014.
- [76] Wolfram Research, Inc., *Mathematica*, version 14.0, Champaign, IL, USA, 2024.
- [77] N. Beisert, *Quantum field theory 1 – chapter 5: free spinor field*, Lecture notes, (2012) <https://edu.itp.phys.ethz.ch/hs12/qft1/Chapter05.pdf> (visited on 02/22/2026).

Danksagung

An dieser Stelle möchte ich mich bei allen Personen bedanken, die mich während der Erstellung dieser Masterarbeit unterstützt haben.

Mein besonderer Dank gilt Mareen Hoppe und Frank Siegert für die Betreuung dieser Arbeit sowie für ihre Unterstützung und die vielen hilfreichen Antworten auf meine Fragen während des gesamten Projekts. Ebenso danke ich Marek Schönherr für seine Unterstützung und die Beantwortung fachlicher Fragen.

Für das sorgfältige Korrekturlesen der Arbeit danke ich Florens Förster, Antonia Bähr, Lena Alshut, Nick Maryshchak, Mareen Hoppe und Robin Schilling.

Ein besonderer Dank gilt außerdem Martin Petzold, der mir während der Bearbeitung dieser Arbeit den Rücken freigehalten hat.

Declaration in AI Assisted Language Editing

I would like to acknowledge the use of AI-based large language models for assistance with grammar, wording, and stylistic refinement of written text. All scientific content, interpretations, and conclusions are entirely my own.

Erklärung

Hiermit erkläre ich, dass ich diese Arbeit im Rahmen der Betreuung am Institut für Kern- und Teilchenphysik ohne unzulässige Hilfe Dritter verfasst und alle Quellen als solche gekennzeichnet habe.

Ort, Datum

Unterschrift