



THE UNIVERSITY *of* EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e.g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

- This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
- A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
- This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
- The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
- When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

The Automated Calculation of One-Loop Processes within MADGOLEM



Ioan Wigmore

A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy
to the
University of Edinburgh
February 2013

Abstract

In the current LHC era, a vast number of models for BSM physics are being tested. For predictions accurate enough to match experimental errors, theoretical calculations have to go beyond LO estimates. However, calculating one-loop corrections in BSM models involves many new particles with specific model dependent properties. Therefore, they are done largely by hand, or in partially-automated ways. I present a fully automated tool for the calculation of generic massive one-loop Feynman diagrams with four external particles, implemented as a module within the fully automated MADGOLEM framework. With this one can compute the NLO-QCD corrections to generic BSM heavy resonance production processes, for example in the context of supersymmetric theories.

Declaration

Except where otherwise stated, the research undertaken in this thesis was the unaided work of the author. Where the work was done in collaboration with others, a significant contribution was made by the author.

I. Wigmore
February 2013

Acknowledgements

I would like to thank many, many people, of which these are but a few.

- The MADGOLEM group: Tilman Plehn; Kentarou Mawatari; David López-Val & Dorival Gonçalves-Netto.
- Friends in the GOLEM group: Gudrun Heinrich; Thomas Reiter & Gavin Cullen.
- For many interesting physics conversations: James Carrington & Erik Gerwick.
- For allowing me to impose during the final months: Andre Costa & Ailsa Sparkes.
- My supervisor, Richard Ball.

This thesis and the code which it describes is dedicated to the late Thomas Binoth.

Contents

Abstract	i
Declaration	ii
Acknowledgements	iv
Contents	vi
1 Introduction	1
1.1 Motivations for NLO calculations	1
1.2 MADGOLEM	2
1.3 Thesis outline	2
1.4 Conventions	3
2 Amplitude theory	5
2.1 Introduction	5
2.2 Cross sections	5
2.2.1 NLO calculations	7
2.2.2 Example: $e^+e^- \rightarrow u\bar{u}$	9
2.3 Feynman diagrams	10
2.3.1 Colorless Feynman rules	11
2.3.2 Colored Feynman rules	13
2.4 Majorana particles and fermion flow	16
2.5 Color methods	19
2.5.1 Color flow decomposition	19
2.5.2 Example: $gg \rightarrow u\bar{u}$	22
2.6 Spinor helicity techniques	23
2.6.1 Massless spinors	24
2.6.2 Massive spinors	26
2.6.3 Helicity amplitudes	28
3 Loop theory	33
3.1 Introduction	33
3.2 Regularisation techniques	34
3.2.1 Dimensional splitting	35
3.3 Tensor techniques	37

3.3.1	Passarino–Veltman reduction	38
3.3.2	Form factor representation	39
3.4	Renormalisation	41
3.4.1	$\overline{\text{MS}}$ renormalisation	42
3.4.2	On–shell renormalisation	44
4	Automated virtual corrections within MADGOLEM	47
4.1	Introduction	47
4.2	Virtual corrections	48
4.2.1	Diagram generation	51
4.2.2	Partial amplitude factorisation	62
4.2.3	Tensor reduction	69
4.2.4	Analytical results	71
4.2.5	Numerical evaluation	75
5	Calculated processes	87
5.1	Introduction	87
5.2	$pp \rightarrow \widetilde{q}\chi_1^0$	88
5.2.1	Motivation	88
5.2.2	Calculation	89
5.3	$pp \rightarrow \widetilde{q}\widetilde{q}^*$	93
5.3.1	Motivation	94
5.3.2	Calculation	95
5.4	$pp \rightarrow \widetilde{g}\widetilde{g}$	100
5.4.1	Motivation	100
5.4.2	Calculation	101
5.5	$pp \rightarrow GG^*$	104
5.5.1	Motivation	105
5.5.2	Calculation	105
6	Conclusion	111
A	Feynman rules	113
A.1	Standard Model QCD	113
A.2	Supersymmetry	114
A.3	Scalar Gluons	118
B	Form factors	123
B.1	1– and 2–point form factors	124
B.2	3–point form factors	124
B.3	4–point form factors	125
C	Running MADGOLEM	127
	Bibliography	128
	Publications	133

Chapter 1

Introduction

1.1 Motivations for NLO calculations

As a result of an outstanding performance, by the end of 2012 the LHC has delivered around 56 fb^{-1} of data at the ATLAS and CMS detectors, setting out on the quest for signatures of new and interesting physics past the Standard Model (SM).

Experimental signatures at the LHC can be highly sensitive to perturbative QCD effects, due to the hadronic nature of the colliding particles. In order to search for experimental signs of new physics (also known as beyond the Standard Model or BSM physics), it is therefore necessary to have a comprehensive understanding of the perturbative framework in which the signature processes reside; accurate predictions, including next-to-leading order (NLO) QCD corrections, are instrumental at this stage.

NLO calculations allow for a reduction of the theoretical uncertainty in predictions, as they render more stable results with respect to the (unphysical) renormalization and factorization scale choices; on the other hand, they provide suitable total rates to normalize the event samples simulated by standard Monte Carlo (MC) generators.

The increased complexity inherent in NLO calculations over that of leading order (LO) ensures that a manual computation of the NLO matrix elements required for many key processes can be unpractical to perform. In this context, the development of dedicated tools that can perform these types of calculations ranks very high in the phenomenologists wishlist, with a particular demand for high degrees of self-automation. In the past years the physics community has witnessed an impressive thrust of activity in the area of NLO predictions. Many groups have contributed NLO tools [1–10], both automated and non-. These developments are documented in various summaries [11–13]. Even with all these contributions however, there is still space for more NLO tools, particularly highly automated ones. Not only is competition between tools encouraged, but a diverse range of tools allows for cross-checks and greatly

improves error checking and consistency across calculations. Diversity also allows for alternative and complementary techniques of calculation, which may be more efficient for certain calculations when other methods are not. Some NLO tools are more oriented towards multileg calculations, while others may allow analytical processing; some NLO tools may also be restricted to specific models and/or processes but also benefit from very fast computation times.

1.2 MADGOLEM

Direct searches for new physics typically rely on resonant production of novel heavy states or, alternatively, on their associated production along with SM particles. For example, the studies conducted at the LHC so far have already enabled to constrain the phenomenologically viable parameter space of the Minimal Supersymmetric Standard Model (MSSM), mainly upon analyses based on jet production from squark and gluino decays plus missing energy. With this in mind, the creation of a highly automated NLO tool for new physics $2 \rightarrow 2$ processes is not only viable, but useful for aiding *timely* and most relevant predictions relevant to the LHC.

MADGOLEM [14–18] is conceived as a highly modular, independent add-on to the Monte Carlo tool MADGRAPH [19]. It implements an automated framework in which to compute total cross sections and distributions for $2 \rightarrow 2$ processes, including QCD quantum effects to NLO accuracy. MADGOLEM is mainly tailored to describe the production of heavy particle pairs within theories beyond the SM. The tool is currently undergoing a final testing phase, prior to its public release, and is meant to be of interest for model-builders, phenomenologists and fundamentally for the LHC experimental community.

This thesis presents a detailed overview of the virtual corrections module for MADGOLEM, complete with algorithms for constructing and calculating the loop matrix elements required for a model-independent $2 \rightarrow 2$ process at NLO.

1.3 Thesis outline

Chapter 2 covers the necessary theory required to understand how to construct generic matrix amplitudes and produce cross sections. This includes techniques for producing model-independent Feynman diagrams, and handling the spinor and color structures by factorisation into separate partial amplitudes. These techniques are usable at all orders of a perturbative calculation.

Chapter 3 contains the theory needed to evaluate the virtual corrections present in a NLO calculation. This covers dimensional regularisation and renormalisation of

loop integrals, as well as methods for reducing tensor integrals into a set of basic form factors or scalar integrals.

Chapter 4 describes the internal structure of the MADGOLEM code, with an emphasis on the virtual corrections module. Here algorithms are constructed from the equations of the previous chapters in order to calculate the virtual NLO–QCD corrections to a generic $2 \rightarrow 2$ process, and explain the internal scripting that automates the calculation. Some concerns are addressed regarding the choice of reference momenta for four–fermionic and entirely massive processes.

Chapter 5 details several example processes calculated by MADGOLEM, in order to convince the reader of the code’s usability. Each of these processes is covered in greater detail in other papers produced by the MADGOLEM group, with a focus on the phenomenological aspects of the numerical results; therefore in this thesis this chapter will focus on the specific duties of the virtual corrections module. The NLO–QCD processes calculated are $pp \rightarrow \{\widetilde{q}\chi_1^0, \widetilde{q}\widetilde{q}^*, \widetilde{g}\widetilde{g}\}$ in the MSSM framework, and $pp \rightarrow GG^*$ in a generic SM extension with additional scalar color octets (scalar gluons or sgluons).

Finally, Chapter 6 summarises and concludes the thesis.

1.4 Conventions

In this thesis, we work in $D \equiv 4 - 2\epsilon$ dimensions for purposes of dimensional regularisation, and keep all physical observables (such as external momenta) in $\hat{D} \equiv 4$ dimensions. We denote a process that has 2 particles incoming and 2 particles outgoing as $2 \rightarrow 2$. For an example process $u\bar{u} \rightarrow e^+e^-$, there is an explicit form:

$$u(k_1) + \bar{u}(k_2) \rightarrow e^+(k_3) + e^-(k_4) , \quad (1.1)$$

where k_1, k_2, k_3, k_4 are the physical four–momenta of the particles involved. Conservation of momentum enforces¹

$$k_1 + k_2 = k_3 + k_4 . \quad (1.2)$$

Four-momenta are presented in Minkowski space, with the metric convention

$$g^{\mu\nu} = \text{diag}(+1, -1, -1, -1) . \quad (1.3)$$

Contractions of Lorentz vectors with Dirac gamma matrices are written using the shorthand Feynman slash notation:

$$\not{k} \equiv k^\mu \gamma_\mu . \quad (1.4)$$

¹Later in the thesis we will treat all four particles as incoming, so that $k_1 + k_2 + k_3 + k_4 = 0$.

The Dirac gamma matrices obey the anti-commutation relation:

$$\{\gamma^\mu, \gamma^\nu\} = 2g^{\mu\nu} . \quad (1.5)$$

Summation of repeated indices is assumed unless otherwise stated. Natural units are used: $\hbar = c = 1$.

In Chapter 4 we provide a description of the MADGOLEM tool, using both mathematical and algorithmic formats:

- File names are formatted in italics, e.g. *file_name.map*.
- Lines of code and algorithms are formatted **as shown in this text**, e.g. with variables `[a_1, a_2, ...]`.
- Algebraic terms and variables are formatted using the conventional equation environment, e.g. with variables (a_1, a_2, \dots) .

Chapter 2

Amplitude theory

2.1 Introduction

In this chapter we describe the standard techniques for constructing matrix amplitudes and cross sections using Feynman diagrams, including rules for BSM Majorana particles, color flow decomposition methods and the spinor helicity formalism. These techniques are presented with an aim to form a set of computer algorithms that can construct and calculate the NLO-QCD cross section for a generic $2 \rightarrow 2$ process.

2.2 Cross sections

A powerful physical observable used in experimental particle physics calculable by theoretical tools is the cross section σ , which defines the probability of a particular interaction event occurring at a given energy scale. When well-defined in perturbation theory, the cross section for a process is calculated by integrating the scattering amplitude matrix element squared $|\mathcal{M}|^2$ over the relevant process-dependent phase space by use of a Monte Carlo event generator.

Interaction events produced at colliders are largely of the type $2 \rightarrow N$: one particle from each of the particle beams interacts, producing N outgoing particles. For hadronic colliders like the LHC and Tevatron, the incoming particles are protons (and antiprotons). However, the underlying physics of the cross section describes the interaction of partons, which requires the devolution of hadronic initial states into the gluons, sea and valence quarks (and antiquarks) contained within. At sufficiently high beam energies (momentum transfer Q greater than the QCD hadronisation scale $\Lambda \sim 1 \text{ GeV}$), incoming hadronic particles can be safely resolved into constituent partons thanks to the factorisation property of QCD [20]. Under this condition, the hadronic differential cross section $d\sigma$ is factorized into the convolution of all contributing partonic

differential cross sections $d\hat{\sigma}$ with the set of non-perturbative partonic distribution functions (PDFs) $f_{i,H}(x, \mu_F^2)$. For a $2 \rightarrow 2$ process, the differential cross section is:

$$d\sigma(H_1(k_1) + H_2(k_2) \rightarrow \varphi_1 + \varphi_2) \equiv \sum_{i,j} \int_0^1 \int_0^1 dx_1 dx_2 f_{i,H_1}(x_1, \mu_F^2) f_{j,H_2}(x_2, \mu_F^2) \quad (2.1)$$

$$\times d\hat{\sigma}(P_i(x_1 k_1) + P_j(x_2 k_2) \rightarrow \varphi_1 + \varphi_2) ,$$

where $H_{1,2}$ are incoming hadrons (protons and antiprotons) and $\varphi_{1,2}$ are outgoing final state particles of interest. The PDFs define the probability for acquiring a given parton i within a known hadron H at a fraction x of the hadronic momenta. PDFs are evaluated at a fixed factorisation scale μ_F , which needs to be fixed at a value that minimises the scale uncertainties inherent in fixed-order perturbation theories, *i.e.* at the same order as the momentum transfer $Q = k_1 + k_2$ of the process. When considering the production of heavy final state particles, the conventional choice is $\mu_F = (m_{\phi_1} + m_{\phi_2})/2$ – which is further identified with the renormalisation scale μ_R too – is known to lead to perturbatively stable results [16].

The partonic differential cross section for a $2 \rightarrow 2$ process is defined as

$$d\hat{\sigma}(P_1(k_1) + P_2(k_2) \rightarrow \varphi_1(k_3) + \varphi_2(k_4)) \equiv \frac{1}{2s} \frac{1}{n_1 n_2} dPS_2(k_1 + k_2) |\mathcal{M}|^2 , \quad (2.2)$$

where n_1 and n_2 are normalisation factors resulting from spin & color sum averages over the two incoming particles, and dPS_2 is the 2-particle Lorenz-invariant kinematic phase-space element:

$$dPS_2(Q) \equiv \left(\prod_{j=3}^4 \frac{d^4 k_j}{(2\pi)^3} \delta^+(k_j^2 - m_j^2) \right) (2\pi)^4 \delta^{(4)}(Q - \sum_{i=3}^4 k_i) \quad (2.3)$$

$$= \frac{1}{(2\pi)^2} d^4 k_3 d^4 k_4 \delta^+(k_3^2 - m_3^2) \delta^+(k_4^2 - m_4^2) \delta^{(4)}(Q - k_3 - k_4) .$$

An additional normalisation factor of $1/n!$ is required for each set of n final-state particles that are indistinguishable (photons, gluons, Majorana fermions etcetera). The scattering amplitude, or matrix element squared $|\mathcal{M}|^2$ in Eq. (2.2) contains all of the kinematical and colour information of the interaction $P_1 + P_2 \rightarrow \varphi_1 + \varphi_2$. It is constructed from the summation of all possible Feynman diagrams that match the initial and final particle requirements at a fixed order in perturbation theory (LO, NLO, NNLO, ...).

The total cross section σ is obtained by integrating Eq. (2.1) over the phase space region (defined by experimental cuts and kinematic restrictions) by use of a Monte Carlo event generator. For non-hadronic initial states, there is no need for factorisation and

$\sigma \equiv \hat{\sigma}$.

Processes with identified hadrons in the outgoing state require the use of non-perturbative fragmentation functions to evolve the outgoing particles, in much the same manner as the use of PDFs to devolve initial hadrons into partonic forms. This thesis however focuses on the calculation of perturbative matrix elements and production cross sections; it is enough to understand the origin of the particle actors, and how the final state fragments or hadronises is not necessary information for these calculations.

2.2.1 NLO calculations

Processes containing colored final and/or initial states are strongly affected by QCD-mediated corrections, which can be calculated by including further orders in α_s , the strong coupling constant, to the matrix element \mathcal{M} . Leading order (LO) processes are calculated at tree level with matrix elements of order α_s^0 or α_s^1 depending on the gauge forces involved. Cross sections at LO contain a significant degree of inaccuracy and are highly dependent on scale variations in μ_F ; by including higher order corrections in the cross section, scale dependencies are reduced and the overall accuracy is increased. Higher order corrections are introduced in the forms of additional external colored particles via real emission, and internal virtual corrections via loops. For a $2 \rightarrow 2$ process, the partonic cross section at next-to-leading order (NLO) is presented as

$$\hat{\sigma}^{\text{NLO}} = \hat{\sigma}^B + \hat{\sigma}^{\text{virtual}} + \hat{\sigma}^{\text{real}} + \hat{\sigma}^{\text{dipole}}. \quad (2.4)$$

$\hat{\sigma}^B$ is defined as the LO cross section, and is constructed from the tree level Born matrix element squared, integrated over the 2-particle final state:

$$\hat{\sigma}^B = \int \frac{1}{2s} \frac{1}{n_1 n_2} dPS_2(Q) |\mathcal{M}^B|^2. \quad (2.5)$$

The virtual contribution $\hat{\sigma}^{\text{virtual}}$ has the same external particle content and kinematics as the LO, and is defined as the integral over the 2-particle final state of the interference term between the Born and the one-loop matrix element:

$$\hat{\sigma}^{\text{virtual}} = \int \frac{1}{2s} \frac{1}{n_1 n_2} dPS_2(Q) \left[(\mathcal{M}^{\text{virtual}})^\dagger \mathcal{M}^B + (\mathcal{M}^B)^\dagger \mathcal{M}^{\text{virtual}} \right]. \quad (2.6)$$

The real emission contribution $\hat{\sigma}^{\text{real}}$ is tree-like (no internal loops) but contains an additional outgoing particle, and is defined as the integral over the 3-particle final state of the tree level $2 \rightarrow 3$ matrix element squared:

$$\hat{\sigma}^{\text{real}} = \int \frac{1}{2s} \frac{1}{n_1 n_2} dPS_3(Q) |\mathcal{M}^{\text{real}}|^2. \quad (2.7)$$

Both the virtual and real contributions can contain infrared (IR) divergences ($1/\varepsilon_{IR}, 1/\varepsilon_{IR}^2$), however their sum is IR finite. In order for Monte Carlo tools to successfully evaluate finite cross sections over the separately divergent 2- and 3-particle phase space regions, the Seymour–Catani dipole subtraction scheme [21, 22] is used to remove soft and collinear IR poles before integration. This scheme is handled by the additional dipole contribution $\hat{\sigma}^{\text{dipole}}$, which is divided into virtual and real parts:

$$\begin{aligned} \hat{\sigma}^{\text{dipole}} = & \left[\int \frac{1}{2s} \frac{1}{n_1 n_2} dPS_2 \sum_j \left(\int dPS_{1,j} \mathcal{D}_j \right) \right]_{\text{virtual}} \\ & - \left[\int \frac{1}{2s} \frac{1}{n_1 n_2} dPS_3 \sum_j \mathcal{D}_j \right]_{\text{real}} , \end{aligned} \quad (2.8)$$

where \mathcal{D}_j contains process-relevant dipole subtraction terms. These two terms are absorbed into $\hat{\sigma}^{\text{virtual}}$ and $\hat{\sigma}^{\text{real}}$ to cancel the local phase space dependent IR divergences and allow for successful numerical integration over the separate 2- and 3-particle phase space regions. The numerically safe form of Eq. (2.4) is therefore:

$$\begin{aligned} \hat{\sigma}^{\text{NLO}} = & \hat{\sigma}^B \\ & + \int \frac{1}{2s} \frac{1}{n_1 n_2} dPS_2(Q) \left[2 \text{Re} \left((\mathcal{M}^B)^\dagger \mathcal{M}^{\text{virtual}} \right) + \sum_j \left(\int dPS_{1,j} \mathcal{D}_j \right) \right] \\ & + \int \frac{1}{2s} \frac{1}{n_1 n_2} dPS_3(Q) \left[|\mathcal{M}^{\text{real}}|^2 - \sum_j \mathcal{D}_j \right] . \end{aligned} \quad (2.9)$$

For the rest of the thesis the second term in Eq. (2.9) will be relabelled as the IR-finite $\hat{\sigma}^{\text{virtual}}$, and the third term will be relabelled as the IR-finite $\hat{\sigma}^{\text{real}}$, unless otherwise specified. This leaves the NLO calculation of a cross section split into three parts:

$$\hat{\sigma}^{\text{NLO}} = \hat{\sigma}^B + \hat{\sigma}^{\text{virtual}} + \hat{\sigma}^{\text{real}} . \quad (2.10)$$

The K factor for a cross section is a good measurement of a process’s sensitivity to NLO effects. It is defined as the proportional difference in the cross section when including NLO corrections:

$$K = \sigma^{\text{NLO}} / \sigma^{\text{LO}} \quad (2.11)$$

K factors ~ 1 point to negligible corrections from NLO effects; these processes may be safely calculated in a LO approximation. Processes with K factors that significantly deviate from 1 are greatly affected by the inclusion of NLO effects, and cannot be approximated at LO with any accuracy. This is typically the case for the production

process of heavy colored particles at the LHC.

2.2.2 Example: $e^+e^- \rightarrow u\bar{u}$

As a simple example of a NLO calculation, we consider the Standard Model process of quark pair production by electron–positron annihilation ($e^+e^- \rightarrow u\bar{u}$) with NLO–QCD corrections, *i.e.* the addition of a gluon either virtually as a loop, or externally as an additional jet. At LO the matrix element \mathcal{M}^B is $\mathcal{O}(\alpha_s^0)$; the NLO corrections $\mathcal{M}^{\text{virtual}}$ and $\mathcal{M}^{\text{real}}$ are $\mathcal{O}(\alpha_s^1)$ and $\mathcal{O}(\alpha_s^{1/2})$ respectively.

The diagrams below provide a schematic representation of the different matrix elements needing to be calculated:

$$\mathcal{M}^B \sim \text{diagram} \quad \mathcal{M}^{\text{virtual}} \sim \sum_{\text{loops, CTs}} \text{diagram} \quad \mathcal{M}^{\text{real}} \sim \sum_{\text{jets}} \text{diagram}$$

At both LO and NLO there are two subchannels for the process $e^+e^- \rightarrow u\bar{u}$: one subchannel propagates via the photon and the other via the Z boson. With only differences in coupling strengths and propagator masses the subchannels are effectively similar, and the NLO corrections are identical for both. For simplicity we will look only at the photon–mediated subchannel, and extend the calculation later to include Z–mediation.

$\mathcal{M}^{\text{virtual}}$ and $\mathcal{M}^{\text{real}}$ are summed over all allowed Feynman diagrams that contribute at the allowed order in the Standard Model. As the initial state e^+e^- is colorless, NLO–QCD virtual corrections only affect the LO vertex $\gamma u\bar{u}$ (1 contributing loop diagram, plus 1 counterterm diagram).¹ There are 2 real emission diagrams, one for each external gluon emitted from one of the outgoing particles. Both the virtual and real matrix elements are UV–finite (by renormalisation for loop corrections, naturally for real emission) and IR–divergent, due to the soft and collinear nature of the added massless gluons. The integrated and un–integrated dipoles required at the cross section level for $\hat{\sigma}^{\text{NLO}}$ are not shown above, but have well-known definitions given in [21].

Each term in the partonic cross section $\hat{\sigma}^{\text{NLO}}$ from Eq. (2.10) is calculated independently, as each component is separately UV– and IR–finite. The matrix elements–squared for $\hat{\sigma}^B$ and $\hat{\sigma}^{\text{real}}$ can both be calculated and integrated by tree–level numerical tools already in existence, such as MADGRAPH/MADEVENT [19]. The calculation of $\hat{\sigma}^{\text{virtual}}$ uses the interference between \mathcal{M}^B and $\mathcal{M}^{\text{virtual}}$, which requires tools for calculating loops as well as tree–level techniques. Both the real and virtual

¹Loop corrections to the external legs are on–shell corrections, which are removed by calculating $\mathcal{M}^{\text{virtual}}$ within an on–shell renormalisation scheme, which is discussed later in Section 3.4.

$\sqrt{s}[\text{GeV}]$	$\sigma^{\text{LO}}[\text{pb}]$	$\sigma^{\text{virtual}}[\text{pb}]$	$\sigma^{\text{real}}[\text{pb}]$	$\sigma^{\text{NLO}}[\text{pb}]$	K
20	332	16.6	-4.15	345	1.04
100	163	8.14	-2.06	169	1.04
500	$725 \cdot 10^{-3}$	$36.3 \cdot 10^{-3}$	$-9.16 \cdot 10^{-3}$	$752 \cdot 10^{-3}$	1.04

Table 2.1: Production rates and K factors for NLO–QCD corrections to $e^+e^- \rightarrow u\bar{u}$. These calculations use the fixed renormalisation and factorisation scales $\mu_{R/F} = m_Z$.

NLO cross sections automatically include dipoles via *e.g.* MADDIPOLE [23, 24].

Numerically calculated LO and NLO cross sections with K factors for $e^+e^- \rightarrow u\bar{u}$ (including both γ and Z subchannels) are shown in Table 2.1. $K = 1.04$ for a range of center-of-mass energies \sqrt{s} , which agrees with the well-known theoretical result for this process:

$$\sigma_{e^+e^- \rightarrow u\bar{u}}^{\text{NLO}} = \left(1 + \frac{\alpha_s(m_z)}{\pi}\right) \sigma_{e^+e^- \rightarrow u\bar{u}}^{\text{LO}} \simeq (1.04) \sigma_{e^+e^- \rightarrow u\bar{u}}^{\text{LO}}. \quad (2.12)$$

2.3 Feynman diagrams

For this thesis the only restrictions on the calculation of NLO cross sections are that the process calculated is $2 \rightarrow 2$ for LO and virtual processes ($2 \rightarrow 3$ for real emission), and defined within a physical model that is fully renormalisable and perturbative in orders of α_s (*e.g.* SM, MSSM, scalar gluons). As discussed in the previous section, calculating σ^{virtual} (without integrated dipoles) requires the evaluation of both $2 \rightarrow 2$ tree-level and loop matrix elements \mathcal{M}^B and $\mathcal{M}^{\text{virtual}}$, which are constructed directly from Feynman diagrams. Due to the model independence required, it is useful to define a set of model-generic Feynman rules that can be used to calculate the Born and loop diagrams for a given process.

The term ‘generic model’ hereon refers to a physical setup with a generic gauge structure ($SU(N_c) \otimes \dots$), variable particle masses, decay widths and couplings. In a generic *renormalisable* model (*i.e.* all couplings with mass dimension $[g_i] \geq 0$), particles can be organised into three physical types and one unphysical type according to their quantised spin numbers. The physical particles are spin-0 scalars S , spin-1/2 fermions F and spin-1 vectors V_μ . Gauge vectors V_μ require gauge fixing, giving rise to the fourth unphysical particle type: Faddeev–Popov ghosts G . Particles with higher spin are excluded by requiring renormalisability for $d = 4$ dimensions. The ($SU(N_c) \otimes \dots$) nature of the generic model allows any of the particles from the four types to also have an associated color index most commonly from either the fundamental or adjoint $SU(N_c)$ gauge groups.

The following derived Feynman rules are depicted accordingly in the figures:

- Scalar particles are denoted by a dashed line.
- Fermionic particles are denoted by a solid line, with an arrow that fixes the fermion flow.
- Vector particles are denoted by a wavy line (or curly line, in the case of color octets).
- Ghost particles are denoted by a dotted line.
- Particles with a fundamental color index have an arrow that fixes the direction of the color flow. In the case of fermions (which already have an arrow), the color flow is aligned with the fermion flow.

2.3.1 Colorless Feynman rules

The Lagrangian terms describing the propagator for each of the four particle types (without color) are

$$\begin{aligned}
 \mathcal{L}_{SS} &\sim (\partial_\mu S)^\dagger (\partial^\mu S) - m^2 S^\dagger S & \mathcal{L}_{FF} &\sim \bar{F} (i\not{\partial} - m) F \\
 \mathcal{L}_{VV} &\sim V_\mu (\partial^2 - m^2) V^\mu & \mathcal{L}_{GG} &\sim \partial_\mu \bar{G} \partial^\mu G - m^2 \bar{G} G
 \end{aligned}
 \tag{2.13}$$

Here the scalars S are defined as complex; for real scalars $S^\dagger = S$. Barred notation for fermions and ghosts is defined in the usual manner as

$$\bar{X} \equiv X^\dagger \gamma_0 .
 \tag{2.14}$$

The Feynman rules resulting from these Lagrangian terms are shown in Fig. 2.1. Gauge vectors are fixed using the Lorenz and Feynman gauge conditions ($\partial^\mu V_\mu = 0$, $\xi = 1$ respectively), simplifying the gauge propagator greatly:

$$\frac{-i}{(p^2 - m^2 + im\Gamma)} \left(g^{\mu\nu} - \frac{p^\mu p^\nu}{p^2} + \xi \frac{p^\mu p^\nu}{p^2} \right) \longrightarrow \frac{-ig^{\mu\nu}}{(p^2 - m^2 + im\Gamma)} .
 \tag{2.15}$$

Breit–Wigner widths Γ are added to the propagators to model unstable particles; however widths are a special set of higher order corrections, and if used must be applied such that they are only present in non-QCD propagators that appear at both LO and NLO orders as an overall factor, in order to be treated as a correction that is consistent across both LO and NLO.

$$\begin{array}{ccc}
 \text{---} \xrightarrow{p} \text{---} & \frac{i}{(p^2 - m^2 + im\Gamma)} & \xrightarrow{p} & \frac{i(\not{p} + m)}{(p^2 - m^2 + im\Gamma)} \\
 \text{---} \xrightarrow{p} \text{---} & \frac{-ig^{\mu\nu}}{(p^2 - m^2 + im\Gamma)} & \cdots \cdots \cdots & \frac{i}{(p^2 - m^2 + im\Gamma)}
 \end{array}$$

Figure 2.1: Generic Feynman rules for colorless propagators by spin, including ghosts.

The Lagrangian terms for all renormalisable colorless 3- and 4-particle interactions with generic couplings g (and chiral couplings $g_{R/L}$) are:

$$\begin{array}{ll}
 \mathcal{L}_{SSS} \sim g SSS & \mathcal{L}_{SSV} \sim g S \overleftrightarrow{\partial}_\mu S V^\mu \\
 \mathcal{L}_{VVS} \sim g S V_\mu V^\mu & \mathcal{L}_{VVV} \sim g (\partial_\mu V_\nu - \partial_\nu V_\mu) V^\mu V^\nu \\
 \mathcal{L}_{FFS} \sim S \overline{F} (g_R \Pi_R + g_L \Pi_L) F & \mathcal{L}_{FFV} \sim \overline{F} \gamma_\mu (g_R \Pi_R + g_L \Pi_L) F V^\mu \\
 \mathcal{L}_{GGV} \sim g (\partial_\mu \overline{G}) G V^\mu & \mathcal{L}_{SSSS} \sim g SSSS \\
 \mathcal{L}_{VVSS} \sim g S S V_\mu V^\mu & \mathcal{L}_{VVVV} \sim g V_\mu V^\mu V_\nu V^\nu .
 \end{array} \quad (2.16)$$

Here $\Pi_{R/L}$ are the chiral projection operators:

$$\Pi_{R/L} \equiv \frac{1}{2} (1 \pm \gamma_5) . \quad (2.17)$$

For fermionic couplings which are not chirality-dependent, $g_R = g_L \equiv g$, and the fermionic interaction Lagrangian terms simplify to

$$\mathcal{L}_{FFS} \sim g S \overline{F} F \quad \mathcal{L}_{FFV} \sim g \overline{F} \gamma_\mu F V^\mu . \quad (2.18)$$

Note: the Lagrangian terms involving vector bosons assume that the vectors are identical. Interactions between differing vector bosons contain extra sets of Minkowski metrics to account for symmetries, however still produce the same Feynman rules irrespective of which vectors are involved. For example, the four-vector Standard Model Lagrangian term composed of two electroweak Z bosons and two electroweak W bosons is

$$\mathcal{L}_{ZZWW} \sim g (g^{\mu\nu} g^{\rho\lambda} - g^{\mu\rho} g^{\nu\lambda}) Z_\mu Z_\nu W_\rho^+ W_\lambda^- . \quad (2.19)$$

The Feynman rules resulting from the interaction Lagrangians in Eq. (2.16) are shown in Fig. 2.2 and Fig. 2.3. The only model-dependent terms are the definitions for the couplings g (and $g_{R/L}$), which will depend on the specific particles and model involved.

Feynman rules for the external states of the three physical particle types are described in Fig. 2.4. Here we use Dirac spinor notation (u, \bar{u}, v, \bar{v}) for fermionic states; in Section 2.4 we will show how Majorana spinors can also be written in this form. Outgoing particles are listed on the left, and incoming particles are listed on the right. For the calculation of \mathcal{M}^B and $\mathcal{M}^{\text{virtual}}$ all external particles can be treated as incoming, so that the process is effectively $4 \rightarrow 0$ instead of $2 \rightarrow 2$. Using conservation of momentum this enforces

$$\sum_{i=1}^4 k_i = 0 . \quad (2.20)$$

This simplifies the use of Feynman rules as only the external states on the right of Fig. 2.4 are used, and a translation back to $2 \rightarrow 2$ is applied only at the numerical integration stage.

Each loop in $\mathcal{M}^{\text{virtual}}$ has an unfixed internal momentum, q . This is integrated over with the D -dimensional measure

$$\int \frac{d^D q}{(2\pi)^D} . \quad (2.21)$$

Finally, each Feynman diagram constructed has an overall sign associated:

$$(-1)^{P+L} , \quad (2.22)$$

where P is the permutation parity of the external spinors with respect to a fixed reference order, and L is the number of closed fermionic loops in the diagram. This factor sets the relative sign required for correct interference between diagrams.

2.3.2 Colored Feynman rules

The additional Feynman rules for $SU(N_c)$ colored particles are applied over the top of the standard colorless rules. This color factorization property becomes very useful later in Section 2.5 when we use color flow decomposition to simplify Feynman amplitudes. The procedure for applying colored Feynman rules is as follows:

- Colored propagators (e.g. gluons, quarks) include an extra delta function in either the $SU(N_c)$ adjoint (δ^{ab}) or fundamental (δ_i^j) indices.
- The color factor for vertices depends on the number and $SU(N_c)$ gauge representation of the colored particles involved. For the case of two colored (plus other colorless) particles, the color flow is the same as a colored propagator.
- For the case of three colored particles (and other colorless), either an adjoint generator $(T^a)_i^j$ or structure constant f^{abc} is applied; the former for the interaction

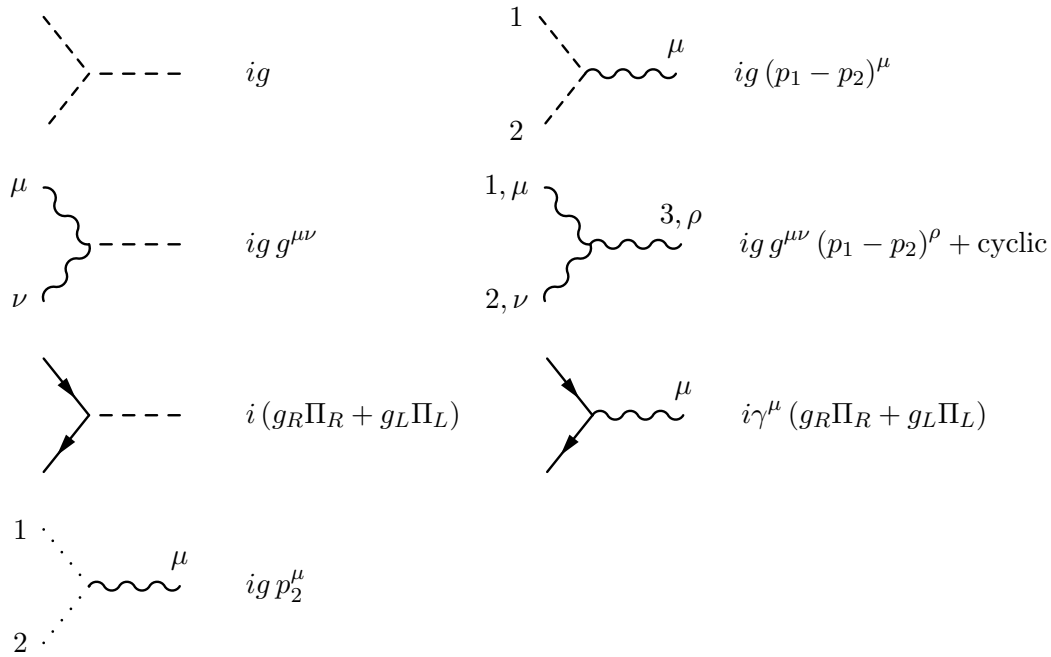


Figure 2.2: Generic Feynman rules for colorless triplet vertices, including ghosts. All momenta are defined as incoming.

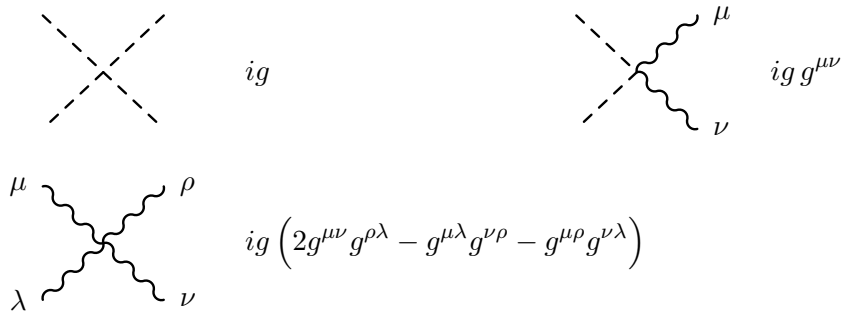


Figure 2.3: Generic Feynman rules for colorless quartic vertices by spin.

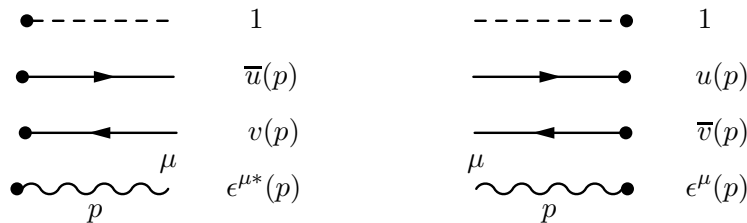


Figure 2.4: Generic Feynman rules for colorless external particles by spin.

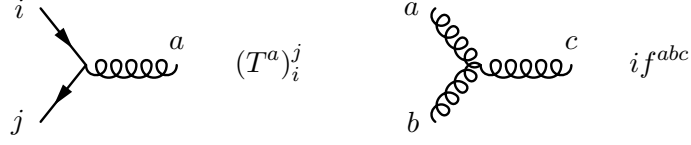


Figure 2.5: Generic Feynman rules for colored triplet vertices.

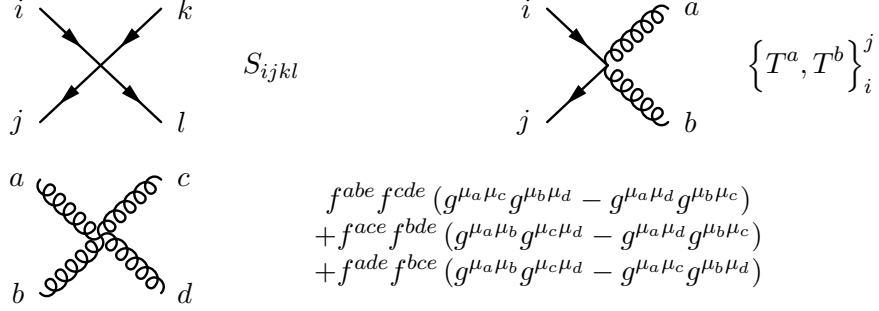


Figure 2.6: Generic Feynman rules for colored quartic vertices.

of two $SU(N_c)$ fundamental particles with a $SU(N_c)$ adjoint, and the latter for the interaction of three $SU(N_c)$ adjoint particles. Colored triplet vertices are shown in Fig. 2.5.

- Feynman rules for the case of colored quartic vertices are given in Fig. 2.6. The structure of the four–fundamental vertex S_{ijkl} is highly dependent on the individual particles concerned and has no standard form. The only four–fundamental vertex used in this thesis is the squark quartic vertex S_{ijkl} for the MSSM, and is defined in Appendix A.2.² The four–adjoint vertex is provided for vector particles (e.g. gluons), as the color structure becomes intrinsically intertwined with the Lorentz structure³. For other four–adjoint vertices, the contributing color and Lorentz structures are smaller, due to the decreased symmetry in external fields.
- External colored states contain an additional $SU(N_c)$ adjoint or fundamental index (e.g. $\epsilon^{\mu,a}$ for external gluons).

Feynman rules for the specific models (SM, MSSM, scalar gluon) that will be used in this thesis are provided in Appendix A.

²See [25] for further MSSM details which include dependence on R–parity, generations etc.

³For the case of the four–gluon vertex, the contributing Lagrangian term is $\mathcal{L} = \frac{1}{4} F^{\alpha\mu\nu} F_{\mu\nu}^{\alpha}$, where $F^{\alpha\mu\nu} = \partial^{\mu} A^{\alpha\nu} - \partial^{\nu} A^{\alpha\mu} + i g_s f^{abc} [A^{b\mu}, A^{c\nu}]$. The resulting four–gluon vertex arises from the term $g_s^2 f^{abc} f^{ade} [A^{b\mu}, A^{c\nu}] [A_{\mu}^d, A_{\nu}^e]$, producing the mixed color/Lorentz structure Feynman rule in Fig. 2.6.

2.4 Majorana particles and fermion flow

Majorana particles arise in supersymmetric extensions to the Standard Model, as the spin-1/2 superpartners of spin-1 gauge bosons. Majorana fermions (χ_M) have a self-conjugate charge:

$$\begin{aligned}\tilde{\chi}_M &= C \bar{\chi}_M^T = \chi_M \\ \tilde{\bar{\chi}}_M &= \chi_M^T C^\dagger = \bar{\chi}_M.\end{aligned}\tag{2.23}$$

C is the charge-conjugation matrix used to relate spinors u and \bar{v} :

$$\begin{aligned}\tilde{v} &= C \bar{v}^T = u, \\ \tilde{\bar{v}} &= v^T C^\dagger = \bar{u},\end{aligned}\tag{2.24}$$

and has the properties

$$C^\dagger = C^{-1}, \quad C^T = -C, \quad C \Gamma_i^T C^{-1} = \eta_i \Gamma_i,\tag{2.25}$$

where

$$\eta_i = \begin{cases} +1 & \text{for } \Gamma_i = 1, \gamma_5, \gamma_\mu \gamma_5 \\ -1 & \text{for } \Gamma_i = \gamma_\mu. \end{cases}\tag{2.26}$$

The self-conjugation property described in Eq. (2.23) ensures that vertices involving Majorana fermions do not have a well-defined fermion flow. In order to construct a complete set of Feynman rules for the interactions of Majorana fermions we have to prescribe a fixed fermion flow, with a constant direction throughout all joined Majorana and Dirac fermions hereafter referred to as a fermion line.

Rules for fixing the fermion flow for a given fermion line are described in [26]. A fermion line composed entirely of Majoranas can be assigned an arbitrary fermion flow, provided it is done in a consistent manner across all diagrams in a given amplitude where external Majoranas are present. A fermion line containing both Majoranas and Dirac particles can also be assigned a fermion flow, although cases where the newly fixed flow clashes with the pre-defined direction of the Dirac fermions require the ‘flipping’ of the involved Dirac structures⁴. For a generic interaction term

$$\mathcal{L}_I = \bar{\chi} \Gamma \chi,\tag{2.27}$$

where χ represents either a Majorana or Dirac fermion, and Γ represents a generic

⁴This property of flipping the fermion flow for both types of fermion is crucial for using the Feynman diagram generator QGRAF [27], which does not differentiate between Majorana and Dirac particles and will assign fermion flows that need flipping without the allowance of a ‘preferred’ Dirac fermion flow.

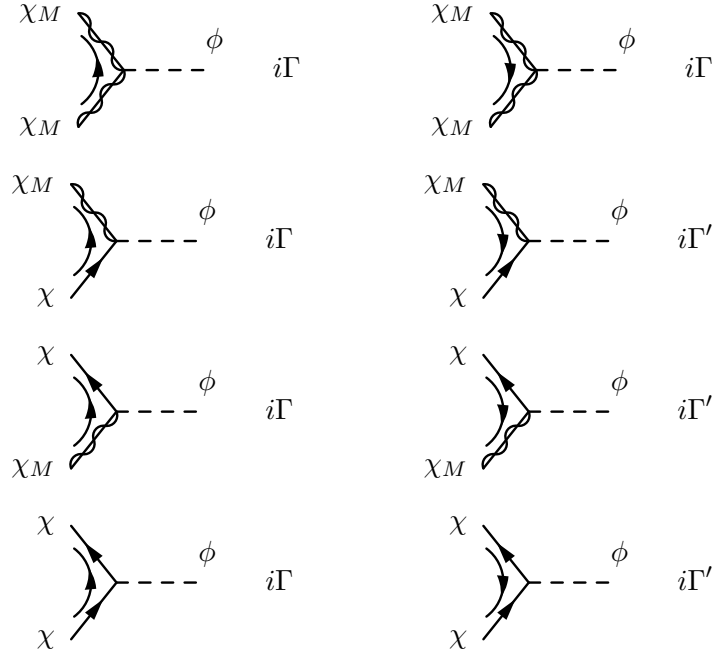


Figure 2.7: Flipping rules for vertices involving Majorana χ_M and Dirac χ fermions. The curved arrow fixes the fermion flow, and flips the vertex $\Gamma \rightarrow \Gamma'$ as necessary.

fermionic interaction involving Dirac matrices, coupling constants and bosonic fields, the ‘flipped’ form is

$$\begin{aligned}
 \mathcal{L}'_I &\equiv \mathcal{L}_I^T \\
 &= \chi^T C^\dagger C \Gamma^T C^\dagger C \bar{\chi}^T \\
 &= \tilde{\bar{\chi}} (\Gamma)' \tilde{\chi} \\
 &\equiv \tilde{\bar{\chi}} (\eta \Gamma) \tilde{\chi} .
 \end{aligned} \tag{2.28}$$

The situations where enforcing a fermion flow results in a flipped vertex are listed in Figure 2.7.

By using the definition for η from Eq. (2.26), we can derive the standard set of flipping rules for all interaction terms involving Majorana and Dirac fermions:

$$\begin{aligned}
 (\mathbb{1})' &= \mathbb{1} \\
 (\gamma^\mu)' &= -\gamma^\mu \\
 (\Pi_{R/L})' &= \Pi_{R/L} \\
 (\gamma^\mu \Pi_{R/L})' &= -\gamma^\mu \Pi_{L/R} \\
 (\gamma^\mu \gamma^\nu \Pi_{R/L})' &= \gamma^\nu \gamma^\mu \Pi_{R/L} .
 \end{aligned} \tag{2.29}$$

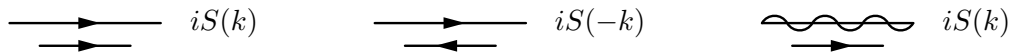


Figure 2.8: Flipping rules for propagators involving Majorana and Dirac fermions. The underlying arrow fixes the fermion flow, and flips the propagator $S(k)$ as necessary. The momentum k flows from left to right.

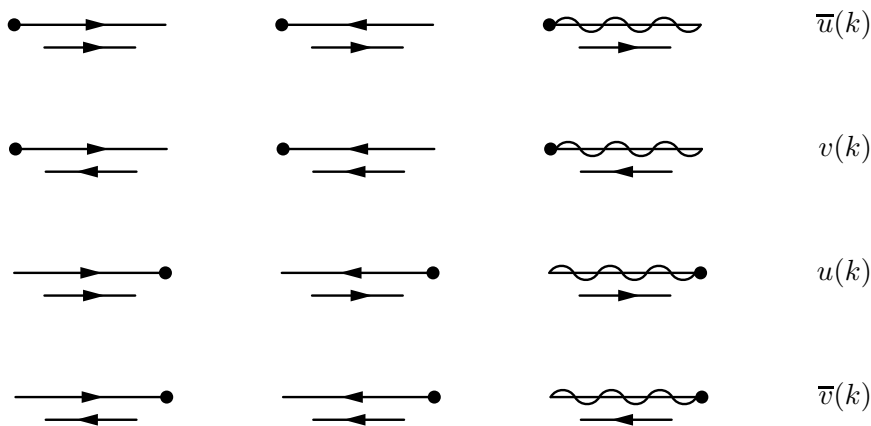


Figure 2.9: Flipping rules for external Majorana and Dirac fermions. The underlying arrow fixes the fermion flow, and defines the external spinor as necessary. The momentum k flows from left to right.

The flipping rules in Eq. (2.29) are applicable for altering the fermion flow of Dirac as well as Majorana fermions, and also provide the rule for reversing the fermionic propagator:

$$(S(k))' = \frac{-\not{k} + m}{(k^2 - m^2)} = S(-k). \quad (2.30)$$

Conditions for flipping a fermionic propagator are listed in Figure 2.8. The spinor nature of external Majorana and Dirac fermions is defined by the fermion flow; flipped external spinors are defined in Eq. (2.24) and listed in Figure 2.9.

The above rules for fixing the fermion flow and flipping necessary internal structures for a given set of Feynman diagrams must be applied in a consistent manner when calculating matrix elements for physical observables. Just as for Dirac fermions, each individual Feynman diagram within a matrix element must be multiplied by a relative

sign as defined in Eq. (2.22).

2.5 Color methods

Color flow decomposition [28, 29] is a technique used to extract the factorisable color-dependent partial amplitude from a $SU(N_c)$ -dependent Feynman diagram. By reducing all color operators to a combination of N_c -dimensional delta functions in fundamental color space, each amplitude can be reduced to a set of kinematic terms that are combined with a basis of simple color structures. This makes the color dependence of the amplitude explicit, and greatly simplifies the results of amplitudes where external color octets such as gluons are involved.

2.5.1 Color flow decomposition

For a non-Abelian $SU(N_c)$ gauge theory, the generic structure that encodes the color information of an amplitude is represented by delta functions δ^{ab} , δ_j^i , fundamental generators $(T^a)_i^j$ and structure constants f^{abc} . Gluons and other color octets carry an $(N_c^2 - 1)$ -dimensional adjoint color index a , while quarks and other color triplets carry a N_c -dimensional fundamental color index i .⁵

This multi-dimensional color structure can be simplified by reducing the number of different color functions. Using the normalisation convention $\text{Tr}(T^a T^b) = T_R \delta^{ab}$, we can rewrite the structure coefficient f^{abc} in terms of the fundamental generator, by starting with the definition for the structure constant:

$$\begin{aligned} i f^{abc} (T^c)_i^j &= [T^a, T^b]_i^j \\ \Rightarrow i f^{abc} (T^c)_i^j (T^d)_j^i &= (T^a T^b)_i^j (T^d)_j^i - (T^b T^a)_i^j (T^d)_j^i \\ &\Rightarrow f^{abd} = -\frac{i}{T_R} \left[\text{Tr}(T^a T^b T^d) - \text{Tr}(T^a T^d T^b) \right] \end{aligned} \quad (2.31)$$

Similarly, any pair of fundamental generators that are summed across the adjoint index a can be reduced into a set of fundamental delta functions:

$$(T^a)_i^j (T^a)_k^l = T_R \left(\delta_i^l \delta_k^j - \frac{1}{N_c} \delta_i^j \delta_k^l \right) \quad (2.32)$$

Eqs. (2.31) and (2.32) are represented diagrammatically in Figure 2.10 to show the fundamental color flow. For completion we note that the above relations describe the color factors of the Feynman rules for the contracted color octet propagator and the

⁵Feynman rules for Standard Model QCD, MSSM QCD and scalar gluons are defined in Appendix A.

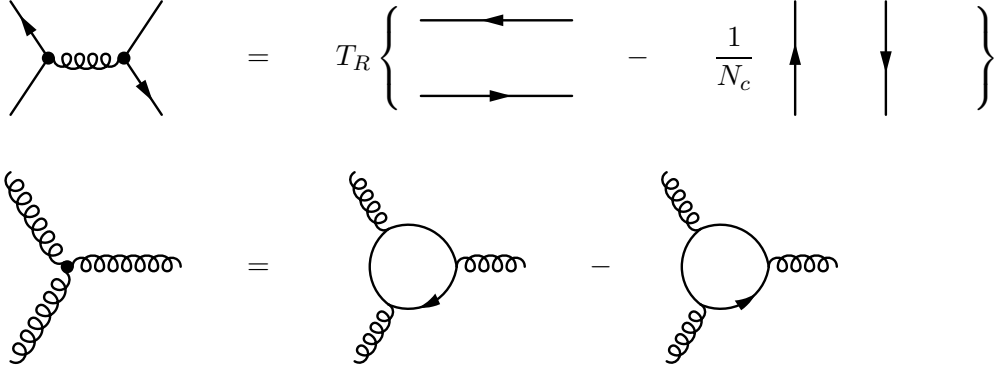


Figure 2.10: Color flow depiction of the contracted fundamental generators $T^a T^a$ and structure coefficient f^{abc} .

3–octet vertex; these color decompositions can be applied to any $SU(N_c)$ gauge theory to factorize otherwise-complex color structures in amplitudes.

The above two equations are not enough to decompose every color factor into purely fundamental delta functions, as the presence of an external color octet such as a gluon will provide an adjoint color index a that remains uncontracted at the amplitude level. An amplitude \mathcal{M} with n_8 external color octets and n_3 external color triplet–antitriplet pairs will have n_8 ‘unsaturated’ adjoint indices. By saturating each external adjoint index a with a corresponding fundamental generator $(T^a)_i^j$ at the amplitude level, all of the loose indices may be contracted, and by applying Eqs. (2.31) and (2.32) the full color structure of the amplitude can be reduced to a series of N_c –dimensional delta functions which describes the color flow. The translation of an amplitude from n_8 loose adjoint indices to the color flow representation is defined:

$$\begin{aligned}
 \mathcal{M}^{a_1, \dots, a_{n_8}} &= \mathcal{M}^{b_1, \dots, b_{n_8}} \prod_{k=1}^{n_8} (\delta^{a_k b_k}) \\
 &= \mathcal{M}^{b_1, \dots, b_{n_8}} \prod_{k=1}^{n_8} \left(\frac{1}{T_R} (T^{a_k})_{i_k}^{j_k} (T^{b_k})_{j_k}^{i_k} \right) \\
 &\equiv \mathcal{M}_{j_1, \dots, j_{n_8}}^{i_1, \dots, i_{n_8}} \prod_{k=1}^{n_8} \left(\frac{1}{\sqrt{T_R}} (T^{a_k})_{i_k}^{j_k} \right),
 \end{aligned} \tag{2.33}$$

$$\mathcal{M}_{j_1, \dots, j_{n_8}}^{i_1, \dots, i_{n_8}} \equiv \mathcal{M}^{a_1, \dots, a_{n_8}} \prod_{k=1}^{n_8} \left(\frac{1}{\sqrt{T_R}} (T^{a_k})_{j_k}^{i_k} \right). \tag{2.34}$$

Eqs. (2.33) and (2.34) are both normalised to ensure that the amplitude-squared is

identical:

$$|\mathcal{M}^{a_1, \dots, a_{n_8}}|^2 = |\mathcal{M}_{j_1, \dots, j_{n_8}}^{i_1, \dots, i_{n_8}}|^2. \quad (2.35)$$

If we define a permutation group S_n to contain all the allowed combinations of n color flow lines for a given amplitude \mathcal{M} , then the factorisation of the set of purely kinematic terms from the basis of color flow lines is straightforward:

$$\mathcal{M} = \sum_{\sigma \in S_n} \mathcal{M}_\sigma |c_\sigma\rangle, \quad |c_\sigma\rangle = \delta_{i_1}^{j_{\sigma(1)}} \dots \delta_{i_n}^{j_{\sigma(n)}}. \quad (2.36)$$

The total number of allowed basis structures in S_n is

$$\#(\sigma \in S_n) = (n_8 + n_3)!, \quad (2.37)$$

which is simply the number of unique permutations within the set of fundamental indices in the amplitude $\{i_1, \dots, i_n\}$, where $n = n_3 + n_8$. In contrast, for a non-saturated non-decomposed basis constructed out of the fundamental generators T^a , the total number of allowed basis structures for a $2 \rightarrow 2$ process is

$$\#(\sigma \in S_T) = \begin{cases} n_3! & \text{no external octets} \\ (n_8 + n_3 - 1)! & \text{otherwise.} \end{cases} \quad (2.38)$$

Constructing the amplitude-squared is straightforward, as the color basis-squared forms a $S_n \times S_n$ matrix containing only contracted combinations of delta functions, with which we can contract the kinematic partial amplitudes:

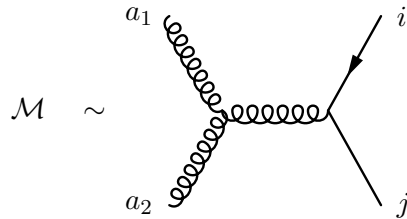
$$|\mathcal{M}|^2 = \sum_{\sigma \in S_n} \sum_{\sigma' \in S_n} \langle c_{\sigma'} | c_\sigma \rangle (\mathcal{M}_\sigma \mathcal{M}_{\sigma'}^* + \mathcal{M}_\sigma^* \mathcal{M}_{\sigma'}) \quad (2.39)$$

We conclude that using the color flow decomposition method allows us to represent internal and external color structures in a mathematically simple manner (being composed entirely of delta functions), that is also easy to present diagrammatically using the rules given in Figure 2.10.

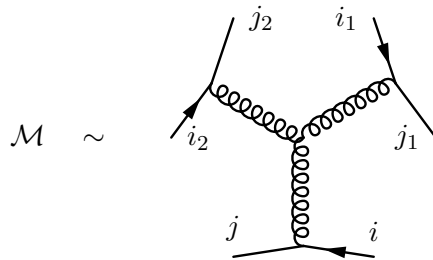
This method is shown to be applicable to any generic non-Abelian $SU(N_c)$ theory; for Standard Model and supersymmetric QCD: $N_c = 3$, $T_R = \frac{1}{2}$.

2.5.2 Example: $gg \rightarrow u\bar{u}$

As an example of the usefulness of color flow decomposition, we will calculate the color factor for a tree-level diagram of $gg \rightarrow u\bar{u}$, featuring an s-channel gluon propagator.



Firstly we will use Eq. (2.34) to saturate the two external gluons with fundamental generators, so that the adjoint indices a_1 and a_2 are replaced with the fundamental indices (i_1, j_1) and (i_2, j_2) respectively. To better view the diagram we will rotate it:



Next we apply the color flow decomposition Eqs. (2.31) and (2.32) iteratively, reducing the adjoint propagators to fundamental color lines:

$$\begin{aligned}
 \mathcal{M} &\sim \text{Diagram 1} - \text{Diagram 2} \\
 &\sim \text{Diagram 3} - \frac{1}{N_c} \left\{ \text{Diagram 4} + \text{Diagram 5} + \text{Diagram 6} \right\} + \frac{2}{N_c^2} \text{Diagram 7} \\
 &\quad - \text{Diagram 8} + \frac{1}{N_c} \left\{ \text{Diagram 9} + \text{Diagram 10} + \text{Diagram 11} \right\} - \frac{2}{N_c^2} \text{Diagram 12}
 \end{aligned}$$

Here it can be seen that most of the color flow diagrams cancel out due to the antisymmetric nature of the 3-gluon vertex. This leaves the final color factor as:

$$\mathcal{M} \sim \begin{array}{c} \text{---} j_2 \text{---} i_1 \\ \text{---} i_2 \text{---} j \\ \text{---} j_1 \text{---} i \end{array} - \begin{array}{c} j_2 \text{---} i_1 \\ \text{---} i_2 \text{---} j \\ \text{---} j_1 \text{---} i \end{array}$$

By resolving each color line into a delta function, we can see that for the above diagram the decomposed colour structure has only one basis, which is (excluding a universal numerical factor):

$$|c\rangle \sim \delta_i^{j_1} \delta_{i_1}^{j_2} \delta_{i_2}^j - \delta_i^{j_2} \delta_{i_1}^j \delta_{i_2}^{j_1}. \quad (2.40)$$

The total color factor for the amplitude-squared as defined in Eq. (2.39) would be:

$$\langle c|c\rangle \sim \left| \delta_i^{j_1} \delta_{i_1}^{j_2} \delta_{i_2}^j - \delta_i^{j_2} \delta_{i_1}^j \delta_{i_2}^{j_1} \right|^2 = 2(N_c^3 - N_c). \quad (2.41)$$

2.6 Spinor helicity techniques

The spinor helicity formalism [30–32] is a powerful tool for factorising and simplifying the kinematic and helicity-dependent partial amplitudes from the total amplitude, by projecting each external particle onto a set of helicity-dependent states and summing the total amplitude over helicity:

$$\mathcal{M} = \sum_{\{\lambda\} \in S_\lambda} \mathcal{M}^{\{\lambda\}}, \quad (2.42)$$

where S_λ contains all the possible helicity combinations for the amplitude. This technique allows the resulting gauge-invariant helicity-dependent terms within $\mathcal{M}^{\{\lambda\}}$ to be represented as a series of Lorentz-invariant Mandelstam variables, multiplied by a kinematically non-invariant prefactor. This can produce greatly simplified results when calculating the total amplitude, particularly in the case where external particles are massless.

In this section we will first describe the prescription used for projecting massless fermions and vector bosons onto the spinor-helicity basis, and then extend the prescription to include massive particles.⁶

⁶Scalar bosons are exempt from the spinor helicity formalism, as they have no spin or helicity representation.

2.6.1 Massless spinors

Massless fermions and vector bosons have only two physical degrees of freedom, but are typically represented by four-component Dirac spinors (u, v, \bar{u}, \bar{v}) and polarisation vectors ($\epsilon_{\pm}^{\mu}, \epsilon_{\pm}^{\mu*}$). The spinor helicity formalism reduces this four-component representation by projecting the massless particles onto definite helicity states:

$$\begin{aligned}
 \Pi^{\pm}u(k) &\equiv \frac{1}{2}(1 \pm \gamma_5)u(k) \equiv u_{\pm}(k), \\
 \Pi^{\pm}v(k) &\equiv \frac{1}{2}(1 \pm \gamma_5)v(k) \equiv v_{\mp}(k), \\
 \bar{u}(k)\Pi^{\pm} &\equiv \bar{u}(k)\frac{1}{2}(1 \pm \gamma_5) \equiv \bar{u}_{\mp}(k), \\
 \bar{v}(k)\Pi^{\pm} &\equiv \bar{v}(k)\frac{1}{2}(1 \pm \gamma_5) \equiv \bar{v}_{\pm}(k).
 \end{aligned} \tag{2.43}$$

The positive and negative energy solutions of the Dirac equation for massless spinors are identical up to a normalization convention [32], allowing for the definition of two unique spinors, with definite helicity. We use the conventional bra-ket notation for massless spinors, defined as:

$$|i^{\pm}\rangle \equiv u_{\pm}(k_i) = v_{\mp}(-k_i), \quad \langle i^{\pm}| \equiv \bar{v}_{\mp}(k_i) = \bar{u}_{\pm}(-k_i), \tag{2.44}$$

where k_i denotes the i -th momentum in a given process.⁷ From Eq. (2.44) we will choose the incoming Dirac spinors $u(k)$ and $\bar{v}(k)$ as our two unique spinors, and note that the helicity notation is reversed for outgoing Dirac spinors. Spinor products are defined in shorthand notation by:

$$\begin{aligned}
 |i\rangle &\equiv |i^+\rangle, \quad |i] \equiv |i^-\rangle, \quad \langle i| \equiv \langle i^-|, \quad [i| \equiv \langle i^+|, \\
 \langle ij\rangle &\equiv \langle i^-|j^+\rangle, \quad [ij] \equiv \langle i^+|j^-\rangle,
 \end{aligned} \tag{2.45}$$

⁷The two outgoing Dirac spinors $v(-k)$ and $\bar{u}(-k)$ have reversed momenta in order to be consistent with the massive formalism described later. For the massless formalism there is no resulting difference.

and come with the following set of identities:

$$\langle i^\pm | \gamma^\mu | i^\pm \rangle = 2k_i^\mu, \quad (2.46)$$

$$k_i = |i\rangle [i| + |i\rangle \langle i|, \quad (2.47)$$

$$\langle ij \rangle = -\langle ji \rangle, \quad [ij] = -[ji], \quad \langle ii \rangle = [ii] = 0, \quad (2.48)$$

$$\langle ij \rangle^* = [ji], \quad (2.49)$$

$$\langle ij \rangle = [ij] = 0, \quad (2.50)$$

$$[i|\gamma^\mu|j\rangle [k|\gamma_\mu|l\rangle = 2[ik]\langle lj\rangle, \quad (2.51)$$

$$[i|\gamma^\mu|j\rangle = \langle j|\gamma^\mu|i\rangle, \quad (2.52)$$

$$\langle ij \rangle [ji] = \text{Tr}(\Pi^- k_i k_j) = 2k_i \cdot k_j = s_{ij}. \quad (2.53)$$

The trace identity (Eq. (2.53)) is one of the most useful, allowing for the representation of spinor products in terms of kinematically-invariant Mandelstam variables s_{ij} . It also exists in an equally helpful four-spinor product form:

$$\begin{aligned} \langle ij \rangle [jl] \langle lm \rangle [mi] &= \text{Tr}(\Pi^- k_i k_j k_l k_m) \\ &= \frac{1}{2} [s_{ij}s_{lm} - s_{il}s_{jm} + s_{im}s_{jl} - 4i\varepsilon(i, j, l, m)], \end{aligned} \quad (2.54)$$

where $\varepsilon(i, j, l, m) \equiv \varepsilon_{\mu\nu\rho\sigma} k_i^\mu k_j^\nu k_l^\rho k_m^\sigma$.

Polarisation vectors for massless vector bosons are constructed by spinor products:

$$\begin{aligned} \epsilon_+^\mu(k, q) &= + \frac{\langle q | \gamma^\mu | k \rangle}{\sqrt{2} \langle q k \rangle}, \\ \epsilon_-^\mu(k, q) &= - \frac{[q | \gamma^\mu | k \rangle}{\sqrt{2} [q k]}, \\ \epsilon_\pm^{\mu*}(k, q) &= \epsilon_\mp^\mu(k, q), \end{aligned} \quad (2.55)$$

where q is an auxiliary light-like reference vector (satisfying $q^2 = 0$) which defines the transverse direction for the polarisation vector, also known as the polarisation axis. Eq. (2.55) satisfies the necessary conditions to define a massless gauge polarisation vector, forming an orthonormal and complete set:

$$\begin{aligned} k \cdot \epsilon_\pm(k, q) &= 0, \\ \epsilon_\lambda(k, q) \cdot \epsilon_{\lambda'}^*(k, q) &= -\delta_{\lambda, \lambda'}, \\ \sum_{\lambda=\pm} \epsilon_\lambda^\mu(k, q) \epsilon_{\lambda'}^{\nu*}(k, q) &= -g^{\mu\nu} + \frac{k^\mu q^\nu + q^\mu k^\nu}{k \cdot q}. \end{aligned} \quad (2.56)$$

Careful choice of the auxiliary reference vector q can greatly simplify partial amplitudes,

as Eq. (2.55) produces a set of further identities:

$$q \cdot \epsilon_{\pm}(k_i, q) = 0, \quad (2.57)$$

$$\epsilon_{\pm}(k_i, q) \cdot \epsilon_{\pm}(k_j, q) = 0, \quad (2.58)$$

$$\epsilon_{+}(k_i, k_j) \cdot \epsilon_{-}(k_j, q) = 0, \quad (2.59)$$

$$\not{\epsilon}_{\pm}(k_i, k_j)|j^{\pm}\rangle = 0. \quad (2.60)$$

It is recommended that q be chosen to be equal to one of the other light-like external momenta $k_{j \neq i}$ involved in the process, to best take advantage of these identities.

2.6.2 Massive spinors

The extension of the spinor helicity formalism to cover massive spinors and polarisation vectors is quite straightforward [30]. Any momentum k_I belonging to massive particle I , can be decomposed into a sum of two light-like momenta, k_i and q :

$$k_I^{\mu} = k_i^{\mu} + \alpha q^{\mu}. \quad (2.61)$$

By using the on-shell conditions $k_I^2 = m_I^2$, $k_i^2 = 0$, $q^2 = 0$, the parameter α is defined:

$$\alpha = \frac{m_I^2}{2q \cdot k_i}. \quad (2.62)$$

Within this formalism k_i^{μ} is understood to be the momentum of the light-like correspondent to I , and q^{μ} is the arbitrarily-defined auxiliary reference momentum. From Eq. (2.61) we can note two things, one is that in the massless limit we retrieve only the light-like correspondent to particle I :

$$k_I^{\mu} \xrightarrow{m_I \rightarrow 0} k_i^{\mu}. \quad (2.63)$$

The second comment is that a useful identity arises when contracting the reference momentum q^{μ} with k_I^{μ} :

$$q \cdot k_I = q \cdot k_i. \quad (2.64)$$

This method of decomposing massive momenta into two light-like vectors can also be applied to massive helicity spinors, using the same bra-ket notation as in Eq. (2.44) and Eq. (2.45). However, due to the presence of mass, the positive and negative energy solutions to the Dirac equation are no longer equivalent up to a normalization convention, and one cannot simply ignore the outgoing spinors $\bar{u}(k)$ and $v(k)$. But if the momentum of the outgoing spinors is reversed ($k \rightarrow -k$), then once again there are

only two unique Dirac spinors, defined in bra-ket notation as

$$|I^\pm\rangle \equiv u_\pm(k_I) = v_\mp(-k_I), \quad \langle I^\pm| \equiv \bar{v}_\mp(k_I) = \bar{u}_\pm(-k_I), \quad (2.65)$$

where k_I denotes the massive I -th momentum in a given process. This allows us to keep our choice of using the incoming Dirac spinors $u(k)$ and $\bar{v}(k)$ as before, with the stipulation that outgoing spinors have reversed helicities and momenta. This prescription works for both massive and massless spinors

The decomposed spinor forms of Eq. (2.61) in terms of massless spinors are:

$$\begin{aligned} |I\rangle &= |i\rangle + \frac{m_I}{[iq]}|q\rangle \\ |I] &= |i] + \frac{m_I}{\langle iq\rangle}|q\rangle \\ \langle I| &= \langle i| + \frac{m_I}{[qi]} \langle q| \\ [I| &= [i| + \frac{m_I}{\langle qi\rangle} \langle q|. \end{aligned} \quad (2.66)$$

They satisfy the Dirac equations:

$$\begin{aligned} (\not{k}_I - m_I)|I^\pm\rangle &= \left(\frac{m_I^2}{2k_i \cdot q} \not{q} - m_I \right) |i^\pm\rangle + (k_i - m_I) \frac{m_I}{[iq]} |q^\mp\rangle \\ &= \left(\frac{m_I}{[iq]} [iq] - m_I \right) |i^\pm\rangle + \left(\frac{m_I^2}{2k_i \cdot q} \langle qi\rangle - \frac{m_I^2}{[iq]} \right) |q^\mp\rangle \\ &= 0, \\ \langle I^\pm|(\not{k}_I + m_I) &= 0, \end{aligned} \quad (2.67)$$

orthogonality relations:

$$\begin{aligned} \langle II\rangle &= [II] = 0, \\ \langle II] &= [II\rangle = 2m_I, \end{aligned} \quad (2.68)$$

and the completeness relation:

$$|I\rangle [I| + |I] \langle I| = \not{k}_I + m_I. \quad (2.69)$$

Eq. (2.64) also holds in spinor form:

$$\langle qI\rangle = \langle qi\rangle, \quad [qI] = [qi]. \quad (2.70)$$

The massive extension to Eq. (2.55) for polarisation vectors uses Eq. (2.61) to define

the massless spinors, and adds a third helicity state, ϵ_0^μ :

$$\begin{aligned}
 \epsilon_+^\mu(k_I, q) &= + \frac{\langle q | \gamma^\mu | k_i \rangle}{\sqrt{2} \langle q k_i \rangle} , \\
 \epsilon_-^\mu(k_I, q) &= - \frac{[q | \gamma^\mu | k_i \rangle}{\sqrt{2} [q k_i]} , \\
 \epsilon_0^\mu(k_I, q) &= \frac{1}{m_I} \left(\frac{m_I^2}{q \cdot k_I} q^\mu - k_I^\mu \right) , \\
 \epsilon_\pm^{\mu*}(k_I, q) &= \epsilon_\mp^\mu(k_I, q) , \\
 \epsilon_0^{\mu*}(k_I, q) &= \epsilon_0^\mu(k_I, q) .
 \end{aligned} \tag{2.71}$$

Note the use of the massless momentum k_i for the spinor products in the transverse helicities, and the massive momentum k_I for the longitudinal helicity. The light-like momentum q is used as the auxiliary reference momentum for both the mass decomposition and also as the polarisation axis. Eq. (2.71) satisfies the orthonormal and completeness requirements for a massive gauge polarisation vector:

$$\begin{aligned}
 k_I \cdot \epsilon_\pm(k_I, q) &= 0 , \\
 k_I \cdot \epsilon_0(k_I, q) &= 0 , \\
 \epsilon_\lambda(k, q) \cdot \epsilon_{\lambda'}^*(k, q) &= -\delta_{\lambda, \lambda'} , \\
 \sum_{\lambda=\pm,0} \epsilon_\lambda^\mu(k, q) \epsilon_{\lambda'}^{\nu*}(k, q) &= -g^{\mu\nu} + \frac{k_I^\mu k_I^\nu}{m_I^2} .
 \end{aligned} \tag{2.72}$$

2.6.3 Helicity amplitudes

Using the spinor-helicity formalism described above, every external fermion line and vector boson in an amplitude can be transformed into a set of helicity amplitudes, composed of massless spinor products on which we can perform the Dirac trace.

Each pair of external fermions connected by an unbroken fermion line in an amplitude has the same generic helicity structure:

$$\mathcal{A} = \bar{v}(k_J) \Gamma u(k_I) . \tag{2.73}$$

Γ contains the internal structure of the fermion line, which is composed of mass terms, momenta and Dirac matrices. As explained previously, outgoing spinors \bar{u} and v can be treated as incoming spinors by reversing the momenta and helicity. The helicity

structure of Eq. (2.73) can be separated into four individual helicity terms:

$$\begin{aligned}
 \mathcal{A} &= \mathcal{A}^{++} + \mathcal{A}^{+-} + \mathcal{A}^{-+} + \mathcal{A}^{--} , \\
 \mathcal{A}^{++} &= \bar{v}(k_J)\Pi^-\Gamma\Pi^+u(k_I) = [J\Gamma I] \\
 \mathcal{A}^{+-} &= \bar{v}(k_J)\Pi^-\Gamma\Pi^-u(k_I) = [J\Gamma I] \\
 \mathcal{A}^{-+} &= \bar{v}(k_J)\Pi^+\Gamma\Pi^+u(k_I) = \langle J\Gamma I \rangle \\
 \mathcal{A}^{--} &= \bar{v}(k_J)\Pi^+\Gamma\Pi^-u(k_I) = \langle J\Gamma I \rangle .
 \end{aligned} \tag{2.74}$$

The four terms can be decomposed into massless spinors using Eq. (2.66), and manipulated into traceable forms by using Eqs. (2.46)–(2.54). The reference momenta for fermions I and J are chosen to be equal, $q_I = q_J = q$, and preferably equal to a separate light-like momentum that is also present in the calculated process. The resulting terms, written as Dirac traces with helicity-dependent prefactors, are:

$$\begin{aligned}
 \mathcal{A}^{++} &= [j\Gamma i] + \frac{m_I}{[iq]} [j\Gamma q] + \frac{m_J}{\langle qj \rangle} \langle q\Gamma i \rangle + \frac{m_I m_J}{[iq]\langle qj \rangle} \langle q\Gamma q \rangle \\
 &= \frac{1}{[iq]\langle qj \rangle} \text{Tr} [\Pi^+(\not{k}_J - m_J)\Gamma(\not{k}_I + m_I)\not{q}] \\
 \mathcal{A}^{+-} &= [j\Gamma i] + \frac{m_I}{\langle iq \rangle} [j\Gamma q] + \frac{m_J}{\langle qj \rangle} \langle q\Gamma i \rangle + \frac{m_I m_J}{\langle iq \rangle \langle qj \rangle} \langle q\Gamma q \rangle \\
 &= \frac{1}{\langle ij \rangle} \left\{ \frac{\text{Tr} [\Pi^+(\not{k}_J - m_J)\Gamma(\not{k}_I + m_I)\not{k}_J\not{q}]}{2k_J \cdot q} - \frac{\text{Tr} [\Pi^+(\not{k}_J - m_J)\Gamma(\not{k}_I + m_I)\not{k}_I\not{q}]}{2k_i \cdot q} \right\} \\
 \mathcal{A}^{-+} &= \langle j\Gamma i \rangle + \frac{m_I}{[iq]} \langle j\Gamma q \rangle + \frac{m_J}{[qj]} [q\Gamma i] + \frac{m_I m_J}{[iq][qj]} [q\Gamma q] \\
 &= \frac{1}{[ij]} \left\{ \frac{\text{Tr} [\Pi^-(\not{k}_J - m_J)\Gamma(\not{k}_I + m_I)\not{k}_J\not{q}]}{2k_J \cdot q} - \frac{\text{Tr} [\Pi^-(\not{k}_J - m_J)\Gamma(\not{k}_I + m_I)\not{k}_I\not{q}]}{2k_i \cdot q} \right\} \\
 \mathcal{A}^{--} &= \langle j\Gamma i \rangle + \frac{m_I}{\langle iq \rangle} \langle j\Gamma q \rangle + \frac{m_J}{[qj]} [q\Gamma i] + \frac{m_I m_J}{\langle iq \rangle [qj]} [q\Gamma q] \\
 &= \frac{1}{\langle iq \rangle [qj]} \text{Tr} [\Pi^-(\not{k}_J - m_J)\Gamma(\not{k}_I + m_I)\not{q}]
 \end{aligned} \tag{2.75}$$

The above equations are completely generic, and may become greatly simplified depending on the number of Dirac gamma matrices in Γ , and whether fermions I

or J are massless. In the massless limit ($m_I, m_J \rightarrow 0$), Eq. (2.75) reduces to:

$$\begin{aligned}
 \mathcal{A}^{++} &\rightarrow [j \Gamma i] = \frac{1}{[iq] \langle qj \rangle} \text{Tr} (\Pi^+ \not{k}_j \Gamma \not{k}_i \not{q}) \\
 \mathcal{A}^{+-} &\rightarrow [j \Gamma i] = \frac{1}{\langle ij \rangle} \frac{\text{Tr} (\Pi^+ \not{k}_j \Gamma \not{k}_i \not{k}_j \not{q})}{2k_j \cdot q} = 0 \\
 \mathcal{A}^{-+} &\rightarrow \langle j \Gamma i \rangle = \frac{1}{[ij]} \frac{\text{Tr} (\Pi^- \not{k}_j \Gamma \not{k}_i \not{k}_j \not{q})}{2k_j \cdot q} = 0 \\
 \mathcal{A}^{--} &\rightarrow \langle j \Gamma i \rangle = \frac{1}{\langle iq \rangle [qj]} \text{Tr} (\Pi^- \not{k}_j \Gamma \not{k}_i \not{q})
 \end{aligned} \tag{2.76}$$

The suppression of helicity amplitudes \mathcal{A}^{+-} and \mathcal{A}^{-+} for massless fermions can be derived with the knowledge that Γ always contains an odd number of Dirac gamma matrices, and that any trace containing an odd number of gamma matrices is zero, with or without the presence of Π^\pm .

Just as any external fermion line can be written as a set of helicity amplitudes composed of a trace and a helicity-dependent prefactor, so can the external polarisation vectors $\epsilon^\mu(k_I, q)$.

$$\begin{aligned}
 \epsilon^\mu(k_I, q) &= \epsilon_+^\mu(k_I, q) + \epsilon_-^\mu(k_I, q) + \epsilon_0^\mu(k_I, q) , \\
 \epsilon_+^\mu(k_I, q) &= + \frac{1}{\sqrt{2} \langle qi \rangle \langle in \rangle [nq]} \text{Tr} (\Pi^- \not{k}_i \not{n} \not{q} \gamma^\mu) \\
 \epsilon_-^\mu(k_I, q) &= - \frac{1}{\sqrt{2} [qi] [in] \langle nq \rangle} \text{Tr} (\Pi^+ \not{k}_i \not{n} \not{q} \gamma^\mu) \\
 \epsilon_0^\mu(k_I, q) &= \frac{1}{m_I} \left(\frac{m_I^2}{q \cdot k_I} q^\mu - k_I^\mu \right) .
 \end{aligned} \tag{2.77}$$

The extra ‘glueing’ momentum n is added to close the trace, and should be chosen in the same manner as the reference momentum q : as a light-like momentum ($n^2 = 0$), ideally equal to a separate momentum that is already present in the process. For clearly-seen reasons in the above equation, $n \neq q$. The longitudinal polarisation vector $\epsilon_0^\mu(k_I, q)$ keeps the same form as before in Eq. (2.71), and does not require tracing.

By transforming all external fermions and vectors bosons in an amplitude into helicity-dependent terms, the complete amplitude can be written into a helicity basis:

$$\mathcal{M} = \sum_{\{\lambda\} \in S_\lambda} \mathcal{M}^{\{\lambda\}} , \tag{2.78}$$

where S_λ contains all the possible helicity combinations for the amplitude. Constructing

the amplitude-squared is straightforward, as the helicity basis is orthogonal, *i.e.*

$$\mathcal{M}^{\{\lambda\}}(\mathcal{M}^{\{\lambda'\}})^* = \delta^{\{\lambda\}\{\lambda'\}}|\mathcal{M}^{\{\lambda\}}|^2 . \quad (2.79)$$

Knowing this, the amplitude-squared becomes:

$$|\mathcal{M}|^2 = \sum_{\{\lambda\} \in S_\lambda} |\mathcal{M}^{\{\lambda\}}|^2 . \quad (2.80)$$

Chapter 3

Loop theory

3.1 Introduction

In this chapter we describe the techniques used to reduce and simplify loops in Feynman diagrams arising from the virtual corrections to a $2 \rightarrow 2$ process in order to calculate $\mathcal{M}^{\text{virtual}}$. These procedures are presented in a model-generic manner and form some of the core algorithms of the MADGOLEM [14–18] code.

A general D -dimensional n -point loop integral of rank $r \leq n$ can be written (*sans* contractions with external momenta and additional metrics) as

$$\mathcal{I}_n^{\mu_1, \dots, \mu_r; D}(\mathcal{S}) \equiv \int \frac{d^D q}{(2\pi)^D} \frac{q^{\mu_1} \dots q^{\mu_r}}{N_1 \dots N_n}, \quad (3.1)$$

where q is the internal loop momentum. The internal propagators are defined (up to a maximum of 4):

$$\begin{aligned} N_1 &\equiv q^2 - m_1^2 + i\epsilon \\ N_2 &\equiv (q + p_1)^2 - m_2^2 + i\epsilon \\ N_3 &\equiv (q + p_1 + p_2)^2 - m_3^2 + i\epsilon \\ N_4 &\equiv (q + p_1 + p_2 + p_3)^2 - m_4^2 + i\epsilon \end{aligned} \quad (3.2)$$

Here p_i is constructed from a combination of the external momenta k_i , and is also conserved:

$$\sum_{i=0}^n p_i = 0. \quad (3.3)$$

The kinematic set \mathcal{S} contains the parameters p_i and m_i . An example 4-point loop integral is shown in Fig. 3.1.

It is well known that these integrals often contain divergences that can arise from

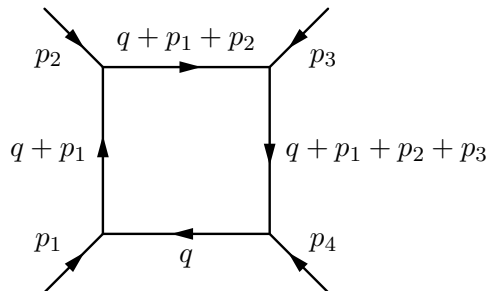


Figure 3.1: 4-point loop with labelled incoming momenta.

	CDR	tHV	DRED
Internal (virtual) Dimensions	D	D	D
External (real) Dimensions	D	\hat{D}	\hat{D}
Int. gluon polarisations	$D - 2$	$D - 2$	2
Ext. gluon polarisations	$D - 2$	2	2
Int. quark polarisations	2	2	2
Ext. quark polarisations	2	2	2

Table 3.1: Dimensional regularisation techniques. ($D \equiv 4 - 2\epsilon$, $\hat{D} \equiv 4$)

both UV sources as well as soft and collinear IR sources. In the case of scalar integrals found in $2 \rightarrow 2$ processes, UV divergences only arise in 1- and 2-point integrals (also known as tadpoles and bubbles), whereas IR divergences are found in 3- and 4-point integrals (triangles and boxes).

3.2 Regularisation techniques

The regularisation of matrix elements is prescribed to handle UV divergences in a gauge-invariant way by calculating loop momenta in (typically) $D \equiv 4 - 2\epsilon$ dimensions. When also applied to soft and collinear divergences, different choices of dimensional regularisation schemes also affect the finite terms originating from IR sources, making it crucial to ensure that both the virtual and real corrections are calculated within the same scheme. Regularisation techniques differ in how they consider the number of dimensions and/or polarisations for internal and external (massless) particles. While a more detailed discussion of the different regularisation schemes can be found in the literature [33–36], Table 3.1 provides a general overview of the standard examples.

Treating external particles in 4 dimensions is necessary for using the spinor-helicity

formalism, which rules out using Conventional Dimensional Regularisation (CDR) [37]. Furthermore, using Dimensional Reduction (DRED) [38] leads to complications in handling the partonic factorisation of processes with massive final-state particles [39]. Therefore, throughout this thesis we will use the 't Hooft Veltman (tHV) [40] prescription for dimensional regularisation.

In D dimensions, Eqn. (3.1) can be rewritten in terms of the well-recognised Passarino–Veltman form factors [41]:

$$\mathcal{I}_n^{\mu_1, \dots, \mu_r; D}(\mathcal{S}) = \frac{i}{16\pi^2} (4\pi\mu_R^2)^\varepsilon \int \frac{d^D q}{i\pi^{D/2}} \frac{q^{\mu_1} \dots q^{\mu_r}}{N_1 \dots N_n}, \quad (3.4)$$

where μ_R is the arbitrary renormalisation scale that enters via the reparameterisation of the strong coupling $g_s^2 \rightarrow (g_s\mu_R^\varepsilon)^2$ in D dimensions.¹

3.2.1 Dimensional splitting

The use of tHV prescribes $D \equiv 4 - 2\varepsilon$ dimensions for internal particles. However, the chirality projection operator γ^5 used within the spinor helicity formalism is a purely $\hat{D} \equiv 4$ -dimensional object, defined as the combination of \hat{D} -dimensional Dirac matrices:

$$\gamma^5 \equiv \frac{i}{4!} \epsilon_{\mu\nu\rho\lambda} \hat{\gamma}^\mu \hat{\gamma}^\nu \hat{\gamma}^\rho \hat{\gamma}^\lambda. \quad (3.5)$$

In this definition for γ^5 there are no anticommutation relations with the D -dimensional Dirac matrices that arise in the tHV scheme. This naturally causes problems when calculating the virtual matrix elements.

To navigate this issue, we rewrite the internal Dirac algebra (involving both Dirac matrices and loop momenta) into two separate components using *dimensional splitting* [42]. At its most basic this means projecting the D -dimensional metric onto an orthogonal $\hat{D} + \tilde{D}$ basis:

$$g^{\mu\nu} = \hat{g}^{\mu\nu} + \tilde{g}^{\mu\nu}; \quad q^\mu = \hat{q}^\mu + \tilde{q}^\mu, \quad \gamma^\mu = \hat{\gamma}^\mu + \tilde{\gamma}^\mu, \quad (3.6)$$

which satisfies the desired relations

$$\hat{g}^{\mu\nu} \hat{g}_{\mu\nu} = \hat{D} = 4, \quad \tilde{g}^{\mu\nu} \tilde{g}_{\mu\nu} = \tilde{D} = -2\varepsilon, \quad \hat{g}^{\mu\nu} \tilde{g}_{\mu\rho} = 0. \quad (3.7)$$

In this framework γ^5 now has a set of well-defined anticommutation and commutation

¹As in this thesis we are only examining loop corrections in orders of g_s , this is the only source of μ_R .

relations with the split Dirac matrices:

$$\{\hat{\gamma}^\mu, \gamma^5\} = 0, \quad [\tilde{\gamma}^\mu, \gamma^5] = 0. \quad (3.8)$$

The anticommutation relations for Dirac matrices can also be determined from Eq. (3.7):

$$\{\hat{\gamma}^\mu, \hat{\gamma}^\nu\} = 2\hat{g}^{\mu\nu}, \quad \{\tilde{\gamma}^\mu, \tilde{\gamma}^\nu\} = 2\tilde{g}^{\mu\nu}, \quad \{\hat{\gamma}^\mu, \tilde{\gamma}^\nu\} = 0. \quad (3.9)$$

The dimensionally-split basis allows a new set of rules to be defined for the loop momenta q^μ and Dirac matrices γ^μ :

$$q^2 = \hat{q}^2 + \tilde{q}^2, \quad \text{Tr}(\hat{\Gamma}\tilde{\Gamma}) = \text{Tr}(\hat{\Gamma})\text{Tr}(\tilde{\Gamma})/\hat{D}. \quad (3.10)$$

The trace rule for a combination of gamma matrices Γ is proven in [42]. The above rules can be used to separate individual Feynman amplitudes into \hat{D} - and \tilde{D} -dimensional terms, at the point before any Dirac traces are evaluated. Extra simplifications in physical amplitudes arise due to Eq. (3.7), which ensures that terms which are linear in q^μ and gamma matrices γ^μ will become \hat{D} -dimensional, as they must eventually be contracted with \hat{D} -dimensional external momenta k_i :

$$\begin{aligned} q \cdot k_i &= \hat{q} \cdot k_i \\ \gamma_\mu k_i^\mu &= \hat{\gamma}_\mu k_i^\mu \end{aligned} \quad (3.11)$$

These relations can be applied easily within FORM, which expects all Dirac algebra to be 4-dimensional by default.

Virtual amplitudes are processed in the following way:

- Take each helicity amplitude as defined in Section 2.6, and split D -dimensional loop momenta and gamma matrices within the Dirac traces using Eq. (3.6).
- Apply anticommutation relations Eq. (3.8) and Eq. (3.9) to push the \tilde{D} -dimensional terms to the right of the \hat{D} -dimensional terms.
- Apply Eq. (3.10) to factorize and evaluate the Dirac traces. Use Eq. (3.11) to simplify and rewrite all terms into either purely D - or composite \tilde{D} -dimensional tensor integrals of the form:

$$\int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^{2\alpha})q^{\mu_1} \dots q^{\mu_r}}{N_1 \dots N_n}, \quad (3.12)$$

where α is an integer number (as odd orders of \tilde{q} are removed by contraction with external momenta).

The D -dimensional tensor integrals can be evaluated normally using the standard techniques. Each of the remaining composite terms can be transformed into a higher-dimensional integral in terms of q with \tilde{q} removed. The full list of possible terms and their transformations for a $2 \rightarrow 2$ process are:

$$\begin{aligned}
 \int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^2)^2}{N_1 \cdots N_4} &= -\varepsilon \int \frac{d^{D+4} q}{i\pi^{(D+4)/2}} \frac{1}{N_1 \cdots N_4} \\
 \int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^2) q^\mu q^\nu}{N_1 \cdots N_4} &= -\frac{\varepsilon}{2} g^{\mu\nu} \int \frac{d^{D+4} q}{i\pi^{(D+4)/2}} \frac{1}{N_1 \cdots N_4} \\
 \int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^2) q^\mu}{N_1 \cdots N_4} &= 0 \\
 \int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^2)}{N_1 \cdots N_4} &= 0 \\
 \int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^2)}{N_1 N_2 N_3} &= \varepsilon \int \frac{d^{D+2} q}{i\pi^{(D+2)/2}} \frac{1}{N_1 N_2 N_3} \\
 \int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^2) q^\mu}{N_1 N_2 N_3} &= \varepsilon \int \frac{d^{D+2} q}{i\pi^{(D+2)/2}} \frac{q^\mu}{N_1 N_2 N_3} \\
 \int \frac{d^D q}{i\pi^{D/2}} \frac{(\tilde{q}^2)}{N_1 N_2} &= \varepsilon \int \frac{d^{D+2} q}{i\pi^{(D+2)/2}} \frac{1}{N_1 N_2}
 \end{aligned} \tag{3.13}$$

We note that non-zero translations are those which satisfy the condition [42]²

$$0 \leq \eta \leq 1; \quad \eta = \lfloor r/2 \rfloor + \alpha + 2 - n, \tag{3.14}$$

where $\lfloor r/2 \rfloor$ indicates that fractional values of $r/2$ should be rounded down to the nearest integer. The transformed $(D+2)$ - and $(D+4)$ -dimensional integrals can be calculated using the techniques in the following section, and result in a finite contribution to the helicity amplitude.

3.3 Tensor techniques

Generic D -dimensional tensor integrals of the form Eq. (3.4) can be simplified and reduced to either a basis of scalar integrals or form factors using the procedures described ahead, either of which can be calculated numerically using a variety of integral libraries (such as GOLEM95 [43, 44]). Neglecting a common prefactor of $i/(16\pi^2)(4\pi\mu_R^2)^\varepsilon$, Eq. (3.4) can be rewritten

$$\mathcal{I}_n^{\mu_1, \dots, \mu_r; D}(S) = \int \frac{d^D q}{i\pi^{D/2}} \frac{q^{\mu_1} \cdots q^{\mu_r}}{\prod_{i=1}^n (q_i^2 - m_i^2 + i\epsilon)}. \tag{3.15}$$

²Orders of $\mathcal{O}(\varepsilon)$ are not present as they do not contribute to the physical amplitude at NLO.

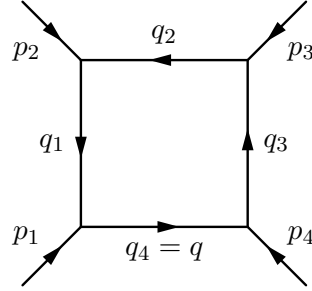


Figure 3.2: 4-point loop with redefined internal momenta.

Here the internal momentum flow has been redefined (see Fig. 3.2 for example):

$$q_i^\mu \equiv q^\mu - r_i^\mu ; \quad r_i^\mu \equiv p_i^\mu + r_{i-1}^\mu , \quad r_n^\mu = 0 \Rightarrow q_n^\mu = q^\mu . \quad (3.16)$$

The parameter S describes the *ordered set* of propagator labels corresponding to the q_i^μ , $S = \{1, \dots, n\}$. Propagators in the loop can be removed or ‘pinched’ by various techniques, in which case the notation $S^{\{k\}}$ is used to pinch the k -th propagator from the ordered set S .

3.3.1 Passarino–Veltman reduction

Any tensor integral depending on external momenta can be simplified using Passarino–Veltman (PV) reduction [41]. This involves rewriting the momenta in the numerator in terms of part of the denominator:

$$\begin{aligned} q \cdot r_i &= \frac{1}{2} [(q^2 - m_n^2) - [q_i^2 - m_i^2] + r_i^2 - m_i^2 + m_n^2] \\ q^2 &= [(q^2 - m_n^2) + m_n^2] . \end{aligned} \quad (3.17)$$

By making one of these substitutions any rank r n -point tensor integral can naturally be simplified to a set of rank $(r - 1)$ n - and $(n - 1)$ -point integrals. For example, in Eq. (3.18) a rank 1 triangle is reduced to two scalar bubbles and one scalar triangle.

$$\begin{aligned} r_{1,\mu} \mathcal{I}_3^{\mu;D}(S) &= \int \frac{d^D q}{i\pi^{D/2}} \frac{q \cdot r_1}{\prod_{i=1}^3 (q_i^2 - m_i^2)} = \frac{1}{2} \left[\int \frac{d^D q}{i\pi^{D/2}} \frac{1}{(q_1^2 - m_1^2)(q_2^2 - m_2^2)} \right. \\ &\quad \left. - \int \frac{d^D q}{i\pi^{D/2}} \frac{1}{(q_2^2 - m_2^2)(q_3^2 - m_3^2)} + \int \frac{d^D q}{i\pi^{D/2}} \frac{r_1^2 - m_1^2 + m_3^2}{\prod_{i=1}^3 (q_i^2 - m_i^2)} \right] \end{aligned} \quad (3.18)$$

The generalised application of Eq. (3.17) to tensor integrals produces:

$$\begin{aligned}
 r_{i,\mu_r} \mathcal{I}_n^{\mu_1,\dots,\mu_r;D}(S) &= \frac{1}{2} \left[\mathcal{I}_{n-1}^{\mu_1,\dots,\mu_{r-1};D}(S^{\{n\}}) - \mathcal{I}_{n-1}^{\mu_1,\dots,\mu_{r-1};D}(S^{\{i\}}) \right. \\
 &\quad \left. + (r_i^2 - m_i^2 + m_n^2) \mathcal{I}_n^{\mu_1,\dots,\mu_{r-1};D}(S) \right] , \\
 g_{\mu_{r-1}\mu_r} \mathcal{I}_n^{\mu_1,\dots,\mu_r;D}(S) &= \mathcal{I}_{n-1}^{\mu_1,\dots,\mu_{r-2};D}(S^{\{n\}}) + m_n^2 \mathcal{I}_n^{\mu_1,\dots,\mu_{r-2};D}(S) .
 \end{aligned} \tag{3.19}$$

In each reduced/pinched integral it may be possible to shift the loop momentum in order to regain the standard integral definition in Eq. (3.15), at which point Eq. (3.19) may be applied iteratively. In the example Eq. (3.18) a rank 1 triangle was reduced purely to scalar integrals, but a complete reduction of all tensor integrals to scalars is not possible using just the PV formalism when $r \geq 2$. If we take this example but include an additional factor $(q \cdot r_1)$ in the numerator, we can see that the second reduced term

$$- \int \frac{d^D q}{i\pi^{D/2}} \frac{q \cdot r_1}{(q_2^2 - m_2^2)(q_3^2 - m_3^2)} \tag{3.20}$$

is not reducible any further, as no terms of q_1 exist in the denominator.

Iterative application of the PV reduction formalism in Eq. (3.19) ensures that each rank r_i n_i -point tensor integral can be reduced to a single scalar n_i -point integral, plus a set of rank $r_f < r_i$ tensor and scalar $n_f < n_i$ -point integrals. In the standard case where tensor integrals have a maximum rank equal to the number of propagators, this guarantees a reduced rank $r \leq 3$ for all tensor integrals in a $2 \rightarrow 2$ process after iterative application of Eq. (3.19). Tensor n -point integrals that depend on momenta other than the external momenta k_i , such as those with numerator terms featuring $q \cdot k_5$ when k_5 cannot be constructed purely from a basis of r_i , cannot be reduced using the PV formalism and will have an unreduced rank $r \leq 4$, as well as remaining tensor n -point integrals.

3.3.2 Form factor representation

Using the procedures of [45, 46], any generic tensor integral of the form given in Eq. (3.15) can be expressed as a linear combination of momenta r_i^μ and scalar form factors A , B , C :

$$\begin{aligned}
 \mathcal{I}_n^{\mu_1,\dots,\mu_r;D}(S) &= \sum_{j_i \dots j_r \in S} [r_{j_1}^\mu \dots r_{j_r}^\mu]^{\{\mu_1 \dots \mu_r\}} A_{j_1, \dots, j_r}^{n,r}(S) \\
 &\quad + \sum_{j_i \dots j_{r-2} \in S} [g^{\mu_1} r_{j_1}^{\mu_2} \dots r_{j_{r-2}}^{\mu_r}]^{\{\mu_1 \dots \mu_r\}} B_{j_1, j_{r-2}}^{n,r}(S) \\
 &\quad + [g^{\mu_1} g^{\mu_2}]^{\{\mu_1 \dots \mu_r\}} C^{n,r}(S) .
 \end{aligned} \tag{3.21}$$

The notation $[g \cdots x \cdots y]^{\{\mu_1 \cdots \mu_r\}}$ describes the full set of permutations of indices $\{\mu_1 \cdots \mu_r\}$ over the metric $g \cdots$ and vectors $x \cdots$ and $y \cdots$. The connection between this representation and that of Feynman parameter integrals is:

$$\begin{aligned} \mathcal{I}_n^{\mu_1, \dots, \mu_r; D}(S) &= (-1)^{-r} \sum_{m=0}^{\lfloor r/2 \rfloor} \left(-\frac{1}{2}\right)^m \sum_{j_1 \cdots j_{r-2m}=1}^n \left[(g \cdots)^{\otimes m} r_{j_1}^{\mu_1} \cdots r_{j_{r-2m}}^{\mu_{r-2m}} \right]^{\{\mu_1 \cdots \mu_r\}} \\ &\times \mathcal{I}_n^{D+2m}(j_1, \dots, j_{r-2m}; S). \end{aligned} \quad (3.22)$$

More detail on Feynman parameter integrals is given in Appendix B.

The form factor representation of all possible tensor integrals for a $2 \rightarrow 2$ process is given in Eqs. (3.23)–(3.27).

$$\mathcal{I}_n^D(S) = A^{n,0}(S) \quad (3.23)$$

$$\mathcal{I}_n^{\mu; D}(S) = \sum_{j \in S} r_j^\mu A_j^{n,1}(S) \quad (3.24)$$

$$\mathcal{I}_n^{\mu_1, \mu_2; D}(S) = \sum_{j_1, j_2 \in S} r_{j_1}^{\mu_1} r_{j_2}^{\mu_2} A_{j_1, j_2}^{n,2}(S) + g^{\mu_1 \mu_2} B^{n,2}(S) \quad (3.25)$$

$$\begin{aligned} \mathcal{I}_n^{\mu_1, \mu_2, \mu_3; D}(S) &= \sum_{j_1, j_2, j_3 \in S} r_{j_1}^{\mu_1} r_{j_2}^{\mu_2} r_{j_3}^{\mu_3} A_{j_1, j_2, j_3}^{n,3}(S) \\ &+ \sum_{j \in S} \left(g^{\mu_1 \mu_2} r_j^{\mu_3} + g^{\mu_1 \mu_3} r_j^{\mu_2} + g^{\mu_2 \mu_3} r_j^{\mu_1} \right) B_j^{n,3}(S) \end{aligned} \quad (3.26)$$

$$\begin{aligned} \mathcal{I}_n^{\mu_1, \mu_2, \mu_3, \mu_4; D}(S) &= \sum_{j_1, j_2, j_3, j_4 \in S} r_{j_1}^{\mu_1} r_{j_2}^{\mu_2} r_{j_3}^{\mu_3} r_{j_4}^{\mu_4} A_{j_1, j_2, j_3, j_4}^{n,4}(S) \\ &+ \sum_{j_1, j_2 \in S} \left(g^{\mu_1, \mu_2} r_{j_1}^{\mu_3} r_{j_2}^{\mu_4} + g^{\mu_1, \mu_3} r_{j_1}^{\mu_2} r_{j_2}^{\mu_4} + g^{\mu_1, \mu_4} r_{j_1}^{\mu_2} r_{j_2}^{\mu_3} \right. \\ &\quad \left. + g^{\mu_2, \mu_3} r_{j_1}^{\mu_1} r_{j_2}^{\mu_4} + g^{\mu_2, \mu_4} r_{j_1}^{\mu_1} r_{j_2}^{\mu_3} + g^{\mu_3, \mu_4} r_{j_1}^{\mu_1} r_{j_2}^{\mu_2} \right) B_{j_1, j_2}^{n,4}(S) \\ &+ (g^{\mu_1 \mu_2} g^{\mu_3 \mu_4} + g^{\mu_1 \mu_3} g^{\mu_2 \mu_4} + g^{\mu_1 \mu_4} g^{\mu_2 \mu_3}) C_{j_1, j_2, j_3, j_4}^{n,4}(S) \end{aligned} \quad (3.27)$$

The form factors are UV divergent for $n = 1, 2$, and IR divergent for $n = 3, 4$.

By applying Eqs. (3.22) & (B.1) & (B.2), the form factors can be reduced to a basis of fundamental scalar integrals:

$$\left\{ \mathcal{I}_1^D, \mathcal{I}_2^D, \mathcal{I}_3^D, \mathcal{I}_4^{D+2} \right\}. \quad (3.28)$$

In this basis the divergent sources are uniquely attributed: $\mathcal{I}_{1,2}^D$ contains only UV poles (which will require renormalisation)³, \mathcal{I}_3^D contains soft and collinear IR poles, and \mathcal{I}_4^{D+2}

³There is one exception, the scalar integral \mathcal{I}_2^D with massless particles $m_1 = m_2 = 0$ and light-like momentum $r_1^2 = 0$. This integral is described using the more manifest Passarino–Veltman notation $B_0(0; 0, 0)$, and contains both a UV and IR simple pole. This integral will feature more prominently in

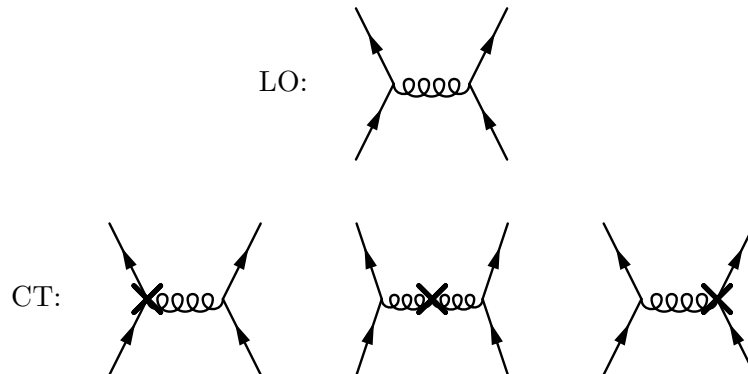


Figure 3.3: Sample LO and counterterm diagrams for the process $gu \rightarrow \gamma u$.

is purely finite. \mathcal{I}_4^{D+2} is also expressible in terms of \mathcal{I}_4^D & \mathcal{I}_3^D , which are both separately IR divergent. A full description of the form factors in the fundamental scalar integral basis is given in Appendix B.

3.4 Renormalisation

The UV renormalisation of QCD virtual amplitudes requires the introduction of counterterms as a new set of Feynman diagrams, that when added to the virtual diagrams, cancel out all instances of UV divergence. In the standard Lagrangian, these counterterms arise by a redefinition of the bare (unrenormalised) parameters:

$$\psi^{(0)} \rightarrow \psi(1 + \delta Z_\psi)^{1/2}, \quad m_\psi^{(0)} \rightarrow m_\psi + \delta m_\psi, \quad g_s^{(0)} \rightarrow g_s(1 + \delta g_s), \quad (3.29)$$

where ψ denotes all strongly interacting fields that require renormalisation.

By renormalising the Lagrangian, one counterterm diagram is required for each possible QCD vertex correction and internal self energy correction that can be made to the initial LO process. For an example see Fig. 3.3. Feynman rules for the individual counterterms are derived by choosing a renormalisation scheme which fixes the renormalisation constants δZ_ψ , δm_ψ , δg_s – i.e. in doing so we normalise the field wavefunction and provide a physical definition for the mass and coupling strength.

In this thesis, the UV renormalisation of QCD virtual amplitudes is handled in two parts: massless particles are renormalised using the $\overline{\text{MS}}$ renormalisation scheme, while massive particles are renormalised using the on-shell scheme.

As the specifics of renormalisation are highly model-dependent, this discourages a generic model approach. Therefore in this section we will use the MSSM framework for

all model-specific details, but otherwise keep the theory model generic.

A full listing of values for the renormalisation constants in the MSSM (as well as the SM and in the context of scalar gluons), along with the Feynman rules for the relevant counterterms can be found in Appendix A.

3.4.1 $\overline{\text{MS}}$ renormalisation

The renormalisation constants δg_s and δZ_ψ (ψ massless) are fixed by the $\overline{\text{MS}}$ renormalisation scheme [40]. The correction to the QCD coupling constant δg_s in the MSSM is provided through a decoupling of all heavy particles ($m \geq m_t$) by subtraction at zero momentum [47–49]. Including corrections from heavy particles [50]:

$$\delta g_s = -\frac{\alpha_s}{4\pi} \frac{\beta_0^L + \beta_0^H}{2} \frac{1}{\tilde{\varepsilon}} - \frac{\alpha_s}{4\pi} \left[\frac{1}{3} \log \frac{m_t^2}{\mu_R^2} + \log \frac{m_g^2}{\mu_R^2} + \frac{1}{12} \sum_{\text{squarks}} \log \frac{m_{q_j}^2}{\mu_R^2} \right], \quad (3.30)$$

where the parameter $1/\tilde{\varepsilon}$ is the $\overline{\text{MS}}$ -shifted UV pole,

$$\frac{1}{\tilde{\varepsilon}} \equiv \frac{(4\pi)^\varepsilon}{\Gamma(1-\varepsilon)} = \frac{1}{\varepsilon} - \gamma_E + \log(4\pi) + \mathcal{O}(\varepsilon). \quad (3.31)$$

The pole can alternatively be written as $1/\tilde{\varepsilon} = r_\Gamma(4\pi)^\varepsilon$, where r_Γ is defined:

$$\begin{aligned} r_\Gamma &\equiv \frac{\Gamma^2(1-\varepsilon)\Gamma(1+\varepsilon)}{\Gamma(1-2\varepsilon)} = \frac{1}{\Gamma(1-\varepsilon)} + \mathcal{O}(\varepsilon^3) \\ &= 1 - \varepsilon\gamma_E + \varepsilon^2 \left[\frac{\gamma_E^2}{2} - \frac{\pi^2}{12} \right] + \mathcal{O}(\varepsilon^3). \end{aligned} \quad (3.32)$$

The beta function is split into light (L) and heavy (H) components:

$$\begin{aligned} \beta_0^L &\equiv \frac{11}{3}N_c - \frac{2}{3}n_f \\ \beta_0^H &\equiv -\frac{2}{3} - \frac{2}{3}N_c - \frac{1}{3}(n_f + 1) \end{aligned} \quad (3.33)$$

Here $n_f = 5$ as we consider the bottom quark to be effectively massless in comparison to the top quark and supersymmetric particles. With the heavy particles decoupled the Q^2 -scale evolution of the QCD coupling constant is determined purely by the light particles and is consistent with the expected values:

$$\frac{\partial g_s^2(Q^2)}{\partial \log(Q^2)} = -\alpha_s^2(Q^2)\beta_0^L \quad (3.34)$$

The other $\overline{\text{MS}}$ -fixed renormalisation constant δZ_ψ (ψ massless) is written in terms

of the scalar two–point Passarino–Veltman integral $B_0(0; 0, 0)$, which is defined as the conflict of two divergences⁴:

$$B_0(0; 0, 0) \equiv \frac{1}{\tilde{\epsilon}_{UV}} - \frac{1}{\tilde{\epsilon}_{IR}}. \quad (3.35)$$

Structuring counterterms in this manner allows them to be matched with the accompanying virtual corrections, which will also have occurrences of B_0 due to tensorial reduction. The renormalisation term for the gluon wavefunction in the MSSM framework (which satisfies the Slavnov–Taylor identity $\delta Z_g^{\text{finite}} = -2\delta g_s^{\text{finite}}$ [51, 52]) is:

$$\delta Z_g = -\frac{\alpha_s}{4\pi}(\beta_0^L + \beta_0^H)B_0(0; 0, 0) + \frac{\alpha_s}{2\pi} \left[\frac{1}{3} \log \frac{m_t^2}{\mu_R^2} + \log \frac{m_g^2}{\mu_R^2} + \frac{1}{12} \sum_{\text{squarks}} \log \frac{m_{q_j}^2}{\mu_R^2} \right]. \quad (3.36)$$

The renormalisation term for the massless quark wavefunction in the MSSM framework $\delta Z_{q_{R/L}}$ is defined in Appendix A.

External gluon counterterms

By choosing to define the counterterms in terms of B_0 (which contains IR poles), this necessitates additional counterterms alongside δZ_g in the presence of external gluons, in order to correctly subtract the heavy fields' IR contributions.

$$\delta Z_{g(\text{ext})} = \frac{\alpha_s \beta_0^H}{4\pi} \frac{1}{2} \frac{1}{\tilde{\epsilon}_{IR}} \quad (3.37)$$

This correction is not required for the case of internal gluons, as the self-energy corrections are already naturally IR-safe due to having a non–vanishing momentum, and their counterterms are already zero by self–cancellation. The use of external gluon counterterms is shown below for two example processes, with the aid of counterterm Feynman rules from Appendix A.2:

- For the process $u\bar{u} \rightarrow g \rightarrow u\bar{u}$ (Fig. 3.4) there is a total contribution of $\frac{1}{2}\delta Z_g - \delta Z_g + \frac{1}{2}\delta Z_g = 0$. The negative sign of the self–energy counterterm ensures that there is no overall contribution from the gluon renormalisation constant.
- For the process $gg \rightarrow g \rightarrow u\bar{u}$ (Fig. 3.5), there is a non–zero contribution of $\frac{3}{2}\delta Z_g - \delta Z_g + \frac{1}{2}\delta Z_g = \delta Z_g$. External gluon counterterms are now required to remove the IR poles contributed by the heavy fields. As there are two external

⁴In situations where we are not concerned with the nature of the divergences in our process, this may be considered to be zero.

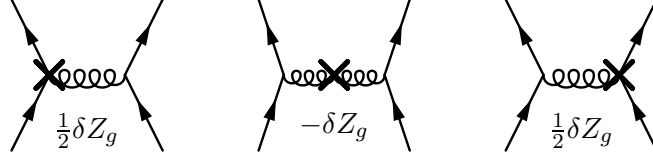


Figure 3.4: Counterterm diagrams for the process $u\bar{u} \rightarrow g \rightarrow u\bar{u}$, noting contributions of δZ_g .

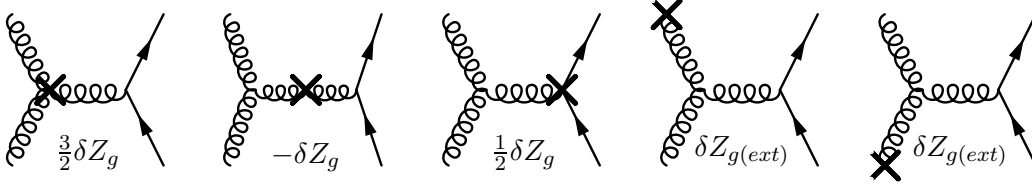


Figure 3.5: Counterterm diagrams for the process $gg \rightarrow g \rightarrow u\bar{u}$, noting contributions of δZ_g .

gluons in this process, the contribution from the external counterterms will be $2\delta Z_{g(\text{ext})} = +\frac{\alpha_s}{4\pi}\beta_0^H/\tilde{\epsilon}_{IR}$, which cancels the IR miscontribution as required.

δg_s within tHV

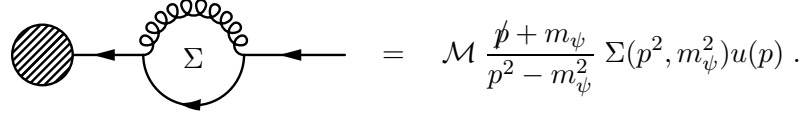
While tVH is an ideal choice of dimensional regularisation for the reasons discussed in Section 3.2, we note that this prescription violates supersymmetric Ward identities due to the mismatch between the number of degrees of freedom for internal gluons ($D-2$) and gluinos (2) [53]. In order to calculate supersymmetric processes featuring gluinos in the MSSM, a modification to the bare Yukawa coupling $\hat{g}_s^{(0)}$ for the vertex $q\tilde{q}\tilde{g}$ is required:

$$\hat{g}_s^{(0)} = g_s^{(0)} \left[1 + \frac{\alpha_s}{4\pi} \left(\frac{2}{3}n_f - \frac{3}{2}C_f \right) \right], \quad (3.38)$$

where $C_f \equiv (N_c^2 - 1)/2N_c$. This shift ensures that the running coupling $\hat{g}_s = g_s$ at the one-loop level, and supersymmetry is preserved.

3.4.2 On-shell renormalisation

The renormalisation constants δm_ψ and δZ_ψ (ψ massive) are fixed by on-shell renormalisation [54]. This scheme is defined by the requirements that the self-energy corrections to external massive particles must be zero under renormalisation. This avoids the rise of internal propagators with denominators $1/(p^2 - m^2)$ which are naturally divergent when p^2 is on-shell, $p^2 = m^2$ (See Fig. 3.6 for example).



$$\text{Diagram} = \mathcal{M} \frac{\not{p} + m_\psi}{p^2 - m_\psi^2} \Sigma(p^2, m_\psi^2) u(p) .$$

Figure 3.6: Example unrenormalised on-shell correction to an external fermion.

For scalar and vector particles, these requirements are:

$$\begin{aligned} \text{Re} \left\{ \hat{\Sigma}(m_\phi^2) \right\} = 0 &\Rightarrow \delta m_\phi = +\text{Re} \left\{ \Sigma(m_\phi^2) \right\} \\ \text{Re} \left\{ \hat{\Sigma}'(m_\phi^2) \right\} = 0 &\Rightarrow \delta Z_\phi = -\text{Re} \left\{ \Sigma'(m_\phi^2) \right\} \end{aligned} \quad (3.39)$$

where $\Sigma' \equiv \partial\Sigma/\partial p^2$ and $\hat{\Sigma}$ is the renormalised self-energy:

$$\hat{\Sigma}(p^2) \equiv \Sigma(p^2) + (p^2 - m_\phi^2)\delta Z_\phi - \delta m_\phi \quad (3.40)$$

For fermions, the on-shell requirement is more complicated:

$$\text{Re} \left\{ \frac{\not{p} + m_\psi}{p^2 - m_\psi^2} \hat{\Sigma}(m_\psi^2) u(p) \right\} = 0 \quad (3.41)$$

where $\hat{\Sigma}$ is the renormalised self-energy:

$$\hat{\Sigma}(p^2) \equiv \Sigma(p^2) + (\not{p} - m_\psi)\delta Z_\psi - \delta m_\psi \quad (3.42)$$

The bare fermionic self-energy Σ can be redefined in terms of a pair of unknown scalar functions A and B :

$$\Sigma(p^2) = (\not{p} - m_\psi)A(p^2) + m_\psi B(p^2) \quad (3.43)$$

This allows Eq. (3.41) to be rewritten (expanding about the point of divergence $p^2 = m_\psi^2$):

$$\begin{aligned} \left[A(m_\psi^2) + 2m_\psi^2 B'(m_\psi^2) + \delta Z_\psi + \frac{2m_\psi^2}{p^2 - m_\psi^2} (B(m_\psi^2) - \delta m_\psi/m_\psi) \right] u(p) &= 0 \\ \Rightarrow \delta m_\psi = -\text{Re} \left\{ m_\psi B(m_\psi^2) \right\} , \quad \delta Z_\psi = -\text{Re} \left\{ A(m_\psi^2) - 2m_\psi^2 B'(m_\psi^2) \right\} \end{aligned} \quad (3.44)$$

where $B' \equiv \partial B/\partial p^2$. By choosing this renormalisation scheme, all on-shell divergences in massive fermions are removed.

Chapter 4

Automated virtual corrections within MADGOLEM

4.1 Introduction

This chapter provides an overview of the MADGOLEM code, with an in-depth explanation of the technical details involved in the virtual corrections module, on which this thesis is focused.

MADGOLEM [14–18] is a highly modular, independent add-on to the Monte Carlo tool MADGRAPH 4.5 [19]. It implements an automated framework in which to calculate the NLO-QCD cross section σ^{NLO} and distributions for $2 \rightarrow 2$ processes.

The MADGOLEM code is split into modular components, which are run in 2 stages:

- *Analytical calculation:* Here the NLO process is specified for a given model by defining a modified MADGRAPH card *proc_card.dat*, and then created by executing *newprocess_nlo* and running the PERL script *run_golem.pl*. These two files construct the analytical form of the terms that make up the differential cross section $d\sigma^{\text{NLO}}$, and write them into a set of FORTRAN90 modules ready for numerical calculation. This analytical output is independent of parameter choices, so only needs to be produced once for a given process.
- *Numerical calculation:* Here the total cross section σ^{NLO} is computed for the chosen model and experimental parameter sets defined in the MADGRAPH cards *param_card.dat* and *run_card.dat* respectively. This is achieved by executing *generate_events_nlo*, which runs all of the FORTRAN90 modules through a Monte Carlo event generator based on the MADGRAPH structure.

The different MADGOLEM modules serve to create the differential cross section terms from Eq. (2.4):

- The tree-level elements $d\sigma^B$ and $d\sigma^{\text{real}}$ (sans dipoles) are created by MADGRAPH 4.5, which is initiated by the *newprocess_nlo* process.
- $d\sigma^{\text{real}}$ is modified by on-shell (OS) subtraction, which is initiated by the *newprocess_nlo* process. OS divergences in the real emission terms can arise at NLO in the case where heavy colored final states emit light-quark jets, these are handled using local OS subtraction in the PROSPINO scheme [50, 55] to avoid a potential double-counting of diagrams. Further details on the OS subtraction are given in the Appendix of [16].
- The virtual corrections $d\sigma^{\text{virtual}}$ are calculated by *newprocess_nlo* and *run_golem.pl*. The executable *newprocess_nlo* runs QGRAF [27] to generate the relevant LO and virtual Feynman diagrams, after which the PERL script *run_golem.pl* runs a set of FORM [56] and MAPLE algorithms to analytically resolve and simplify the resulting matrix elements using techniques inherited from GOLEM [43, 57]. It then includes model-dependent counterterms for renormalisation, and constructs FORTRAN90 code for the resulting $d\sigma^{\text{virtual}}$, with references to the GOLEM95 integral library [43, 44] to allow numerical calculation of scalar integrals and form factors.
- The virtual $2 \rightarrow 2$ and real $2 \rightarrow 3$ parts of $d\sigma^{\text{dipole}}$ (Eq. (2.8)) are initialised by *newprocess_nlo*, and implemented as an extended add-on to the Catani-Seymour dipole framework provided by MADDIPOLE [24]. The two components are added to $d\sigma^{\text{virtual}}$ and $d\sigma^{\text{real}}$ respectively, in order to cancel IR poles. Further details on the dipole terms are given in the Appendix of [16].

A flowchart depicting the interactions of the various modules that make up MADGOLEM is provided in Fig. 4.1.

A short list of instructions for creating and calculating a NLO-QCD process with MADGOLEM is given in Appendix C.

4.2 Virtual corrections

The virtual corrections module in MADGOLEM is based on the Feynman-diagrammatic approach used by GOLEM [43, 57] and expounded in Chapters 2 & 3, optimised for implementation in MADGRAPH. It is designed with the aim to calculate the NLO-QCD virtual corrections to any $2 \rightarrow 2$ process in a model-generic approach, such that it can be applied to the Standard Model, MSSM, and many other renormalisable theories.

The creation and calculation of the virtual corrections term $d\sigma^{\text{virtual}}$ is primarily governed by the PERL script *run_golem.pl*. This ‘master script’ runs a sequence of

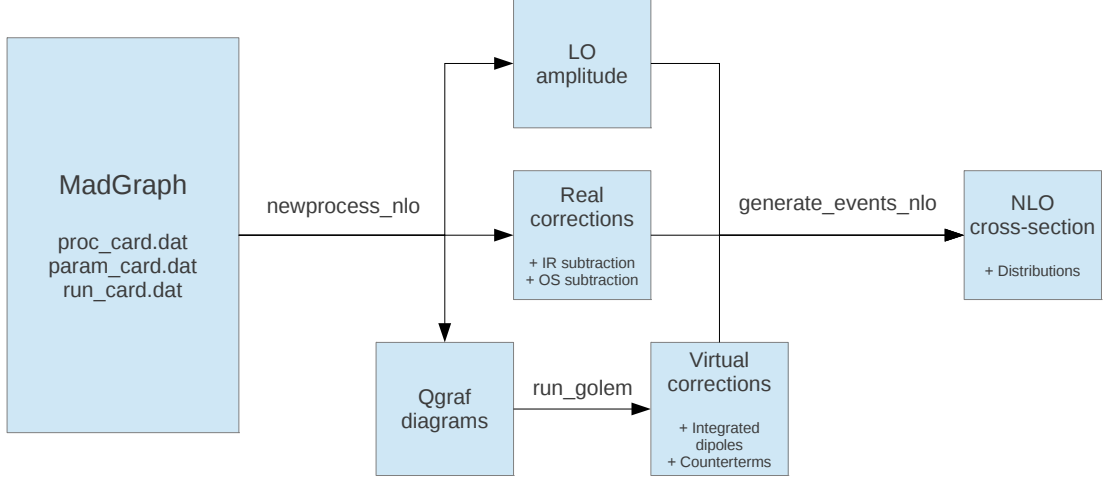


Figure 4.1: Flowchart of the MADGOLEM tool (credited to Ben Wynne).

algorithms written in FORM and MAPLE files to first apply Feynman rules to QGRAF-produced LO, loop, and counterterm Feynman diagrams. It then uses the spinor helicity formalism, color methods and loop reduction techniques to produce matrix amplitudes for both LO and renormalised virtual elements, which are combined in a FORTRAN90 code that returns the numerical array $[a_0, a_1, a_2]$, corresponding to the virtual part of Eq. (2.2):

$$\frac{2}{n!n_1n_2} \text{Re} \left[(\mathcal{M}^{\text{virtual}})^\dagger \mathcal{M}_{\text{LO}} \right] = a_0 + a_1 \frac{1}{\tilde{\epsilon}_{IR}} + a_2 \frac{1}{\tilde{\epsilon}_{IR}^2}. \quad (4.1)$$

Notice that the UV poles $1/\tilde{\epsilon}_{UV}$ are cancelled upon the addition of the one-loop and the counterterm amplitudes within $\mathcal{M}^{\text{virtual}}$. The remaining IR divergences contained in a_1 and a_2 are to be cancelled by the integrated dipoles.

By combining Eq. (4.1) with the total partonic NLO-QCD cross section as described in Eq. (2.9), the partonic NLO-QCD differential cross section for a given $2 \rightarrow 2$ process is:

$$\begin{aligned} d\hat{\sigma}^{\text{NLO}} &= d\hat{\sigma}^B \\ &+ dPS_2(Q) \frac{1}{2s} \left[\left[a_0 + a_1 \frac{1}{\tilde{\epsilon}_{IR}} + a_2 \frac{1}{\tilde{\epsilon}_{IR}^2} \right] + \sum_j \left(\int dPS_{1,j} \mathcal{D}_j \right) \right] \\ &+ dPS_3(Q) \frac{1}{2s} \frac{1}{n_1n_2} \left[|\mathcal{M}^{\text{real}}|^2 - \sum_j \mathcal{D}_j \right]. \end{aligned} \quad (4.2)$$

During the numerical calculation stage, the array $[a_0, a_1, a_2]$ is combined with the equivalent array output by the integrated dipole term $\sum_j (\int dPS_{1,j} \mathcal{D}_j)$: $[d_0, d_1, d_2]$. If the virtual corrections and dipole contributions are calculated correctly, $d_1 = -a_1$

and $d_2 = -a_2$, ensuring that the divergent IR poles are removed from the result. This leaves the finite virtual contribution to the partonic NLO-QCD differential cross section:

$$d\hat{\sigma}^{\text{virtual}} = dPS_2(Q) \frac{1}{2s} [a_0 + d_0] \quad (4.3)$$

To produce the total cross section, Eq. (4.3) is integrated over the weighted phase space, including PDFs as shown in Eqs. (2.1) & (2.6), and added to the total calculated LO and real emission cross sections.

run_golem.pl also produces analytical MAPLE output in the form of partial amplitudes: where each Feynman diagram is broken down into a basis of underlying color structure, specific helicity, and contributing scalar integrals and form factors (produced by the tensor reduction techniques in Section 3.3). This makes the underlying structure of the one-loop process explicit, and allows for a large amount of information to be analysed before a numerical calculation is even needed.

The virtual corrections code as governed by *run_golem.pl* can be loosely broken down into four separate routines:

1. Generating the full set of Born diagrams, one-loop diagrams, and counterterms with QGRAF, and translating the output into FORM code suitable for symbolic manipulation. The structures describing the Feynman diagrams and the corresponding Feynman rules are rewritten as algebraic expressions, keeping track of external wave functions, vertex couplings and internal propagators, color factors, Lorentz structure, and the overall sign from external fermion fields.
2. Mapping the analytical evaluation of the color, helicity and tensor structures onto partial amplitudes, *i.e.* a basis of color, helicity and tensor structures based on the spinor-helicity formalism.
3. Applying an analytical reduction of tensor integrals to scalar loop integrals and/or form factors.
4. Collecting the results and inserting the correct renormalisation constants into the counterterms. The final output for the virtual corrections is returned both as the set of analytical partial amplitudes in MAPLE, and a FORTRAN90 code for calculating the numerical matrix element-squared.

run_golem.pl has several flags available to be defined before running. The first flag is `$num.workers`, which is set to `=2` by default. This flag sets the maximum number of individual process threads that *run_golem.pl* will use when calculating individual diagrams and partial amplitudes in parallel; for maximum processing efficiency this number should be equal to the number of available cores on the computer. The

particles.dat:

#Name	anti_Name	Spin	Linetype	Mass	Width	Color	Label	Model
d	d $\bar{}$	F	S	ZERO	ZERO	T	d	1
dl	dl $\bar{}$	S	D	MDL	WDL	T	dl	1000001
g	g	V	C	ZERO	ZERO	0	_	21

particles-qgraf.dat:

#Name	anti_Name	Sign	2Spin	Color	Mass	Width
d	dx	-	1 3	ZERO	ZERO	
dl	dlx	+	0 3	MDL	WDL	
g	g	+	2 8	ZERO	ZERO	

selfenergy.ct.dat:

#Name	anti_Name	Sign	2Spin	Color	Mass	Width
d	dx	-	1 3	CTZERO	ZERO	
dl	dlx	+	0 3	CTMDL	WDL	
g	g	+	2 8	CTMGLU	ZERO	

Figure 4.2: Sample particle data from a MADGRAPH model file, and corresponding QGRAF model file, plus self-energy counterterm QGRAF model file.

second flag is the number of massless fermions in the process, `$N_FERMIONS` (default =5), which affects the value of the counterterm beta functions, and must be consistent with the model files used to generate the process. After these two there are a set of debugging flags allowing the separate stages of *run_golem.pl* to be disabled, and flags for topologically restricting the set of generated diagrams (which are discussed in the next section).

4.2.1 Diagram generation

Qgraf generation

Initiated by *newprocess_nlo*, QGRAF generates all possible topological LO, counterterm, and on-shell QCD-induced loop diagrams that correspond to the process specified in the MADGRAPH card *proc_card.dat*. The topological rules that QGRAF uses to construct diagrams are read from modified versions of the model files used by MADGRAPH which contain lists of all allowed particles and particle interactions (vertices) for a given model (SM, MSSM, etc). A sample comparison between the standard MADGRAPH model files and their QGRAF counterparts is given in Figs. 4.2 & 4.3.

interactions.dat, interactions-qgraf.dat:

```
#particle1 particle2 particle3 coupling model (QCD,QED)
d d g GG QCD
d d a GAD QED
e- e- a GAL QED
```

vertex_ct.dat:

```
#particle1 particle2 particle3 coupling model (QCD,QED)
d d g GGCT QCD
d d a GADCT QED
e- e- a GAL QED
```

Figure 4.3: Sample interaction data from a MADGRAPH model file, and corresponding QGRAF model file, plus vertex counterterm QGRAF model file.

The LO topologies are straightforward to produce by QGRAF, as the only required input is the external particles defined in *proc_card.dat* and the QGRAF model files *particles-qgraf.dat* & *interactions-qgraf.dat*.

Counterterm topologies are produced by asking QGRAF to create another set of LO topologies as before, but using the model files *selfenergy_ct.dat* & *vertex_ct.dat*, which have the tag CT associated with every QCD–gauge particle and interaction. The master script *run_golem.pl* then creates n_{CT} copies of each counterterm topology with n_{CT} CT–tagged propagators and vertices, and systematically strips out all but one CT tag from each topology. This ensures that for each diagram in the LO amplitude, there are multiple counterterm diagrams with CT–tagged vertices and propagators to act as placeholders for the counterterm Feynman rules, in agreement with the example Fig. 3.3 in Section 3.4. *run_golem.pl* also copies an additional counterterm topology for each external gluon present in the process, and attaches the WF tag to act as a placeholder for the external gluon counterterms, in accordance with the example Fig. 3.5 in Section 3.4.

The loop topologies are created by running QGRAF with the model files *particles-qgraf.dat* & *interactions-qgraf.dat*, and requiring the output of all one loop topologies that satisfy the external particles (with QGRAF options= `notadpole, onshell`). This output may contain many undesirable topologies of an incorrect order of the strong coupling α_s , as it includes all NLO corrections as opposed to only the NLO–QCD corrections which are desired. To filter out the unwanted diagrams and ensure that only NLO–QCD loops are produced, *run_golem.pl* first counts the order $\mathcal{O}(\alpha_s^{n_{LO}})$ of the LO process, then removes all loop diagrams that are not $\mathcal{O}(\alpha_s^{n_{LO}+1})$. The orders are counted by searching the QGRAF diagrams for occurrences of MADGRAPH–notation

QCD couplings `GG`, `G`, `G4G`, `GC`, `GQLGOM` etc. This filtering process can be altered by changing the flag `nloftype` within `proc_card.dat` (default `nloftype=0`). Dedicated flags (`nloftype=1,2,...`) are tailored to different models and oriented to ease the implementation of certain restrictions. For example, choosing `nloftype=6` produces *all* NLO corrections and enforces the QGRAF rule:

```
false= iprop[g,gh,u,d,c,s,t,b,sg,sgx,0,0] ;
```

This flag is used for the calculation of NLO-QCD corrections to the sgluon model, and demands at least one colored propagator to be from the SM or a sgluon/anti-sgluon. Setting `nloftype` to be anything other than the default `0` does not guarantee the creation of only NLO-QCD loops, and therefore demands a careful understanding of the process to be calculated.

The overall sign associated with each QGRAF topology has to be fixed by `run_golem.pl`, as external Majorana particles (when present) can confuse the QGRAF sign algorithm, due to having an initially unfixed fermion flow. Therefore this value is prescribed by `run_golem.pl` in a consistent manner using Eq. (2.22). Example QGRAF output for the LO, counterterm and loop topologies for the process $e^+e^- \rightarrow \tilde{u}_R \tilde{u}_R^*$ is given in Figs. 4.4, 4.5 & 4.6. The output is formatted for parsing by FORM using a QGRAF style template created by Thomas Reiter [58].

More precise topological constraints on the created diagrams (e.g. removing all gluonic t-channel contributions or self-energy corrections) can also be applied by `run_golem.pl`, which also checks for and removes diagrams which are trivially zero, such as diagrams which mix photonic/gluonic currents via a quark loop. These rules are applied by searching for particular topological features within each diagram and discarding those which do not meet the requirements. For example, setting `$gluon_tchannel = 0` calls the PERL code in Fig. 4.7. This ensures that all gluonic t-channel diagrams are discarded from the process. Alternatively, setting `$selfenergies = 0` calls the PERL code that discards diagrams with 2 non-loop propagators (propagators without any terms of the loop momentum q^μ). By making use of these field insertion and loop topological restraint options in `run_golem.pl`, any debugging and testing of processes in MADGOLEM can be greatly simplified.

`run_golem.pl` also uses the topological selection algorithms to identify and group together diagrams with common topologies by changing the overall factor in the QGRAF output. For example, the set of self-energy corrections to the gluon propagator contains n_f identical diagrams featuring massless fermions in the loop; these can be grouped into one such diagram with an overall factor n_f ¹.

¹This technique is known as loop-filtering in the literature.

qgraf_lo.out:

```

*--#[ diagram1:
*
Local diagram1 =
+ 1 *
inp([field.ep], idx1r1, p1) *
inplorentz(-1, iv1r1L1, p1, ZERO ) *
inpcolor(1, iv1r1C1) *
inp([field.em], idx1r2, p2) *
inplorentz(+1, iv1r2L1, p2, ZERO ) *
inpcolor(2, iv1r2C1) *
out([field.ur], idx2r3, p3) *
outlorentz(+0, iv2r3L0, p3, MUR ) *
outcolor(1, iv2r3C3) *
out([field.urx], idx2r2, p4) *
outlorentz(-0, iv2r2L0, p4, MUR ) *
outcolor(2, iv2r2C3) *
vertex(iv1,GZL ,ONE,
[field.ep], idx1r1, -1, p1, iv1r1L1, -1, iv1r1C1,
[field.em], idx1r2, +1, p2, iv1r2L1, +1, iv1r2C1,
[field.z], idx1r3, +2, -p1-p2, iv1r3L2, +1, iv1r3C1) *
vertex(iv2,GZURUR ,ONE,
[field.z], idx2r1, +2, p1+p2, iv2r1L2, +1, iv2r1C1,
[field.ur], idx2r2, +0, -p4, iv2r2L0, +3, iv2r2C3,
[field.urx], idx2r3, -0, -p3, iv2r3L0, -3, iv2r3C3) *
prop([field.z], ZERO, idx2r1, idx1r3) *
propcolor(+1, iv2r1C1, iv1r3C1) *
proplorentz(+2, -p1-p2, ZMASS , iv2r1L2, iv1r3L2)
;
*--#[ diagram1:

```

Figure 4.4: Sample output LO QGRAF topology for the process $e^+e^- \rightarrow \tilde{u}_R\tilde{u}_R^*$.

qgraf_ct.out:

```

*--#[ diagram1:
*
Local diagram1 =
+ 1 *
inp([field.ep], idx1r1, p1) *
inplorentz(-1, iv1r1L1, p1, ZERO ) *
inpcolor(1, iv1r1C1) *
inp([field.em], idx1r2, p2) *
inplorentz(+1, iv1r2L1, p2, ZERO ) *
inpcolor(2, iv1r2C1) *
out([field.ur], idx2r3, p3) *
outlorentz(+0, iv2r3L0, p3, MUR ) *
outcolor(1, iv2r3C3) *
out([field.urx], idx2r2, p4) *
outlorentz(-0, iv2r2L0, p4, MUR ) *
outcolor(2, iv2r2C3) *
vertex(iv1,GZL      ,ONE,
[field.ep], idx1r1, -1, p1, iv1r1L1, -1, iv1r1C1,
[field.em], idx1r2, +1, p2, iv1r2L1, +1, iv1r2C1,
[field.z], idx1r3, +2, -p1-p2, iv1r3L2, +1, iv1r3C1) *
vertex(iv2,GZURURCT,ONE,
[field.z], idx2r1, +2, p1+p2, iv2r1L2, +1, iv2r1C1,
[field.ur], idx2r2, +0, -p4, iv2r2L0, +3, iv2r2C3,
[field.urx], idx2r3, -0, -p3, iv2r3L0, -3, iv2r3C3) *
prop([field.z], ZERO, idx2r1, idx1r3) *
propcolor(+1, iv2r1C1, iv1r3C1) *
proplorentz(+2, -p1-p2, ZMASS , iv2r1L2, iv1r3L2)
;
*--#[ diagram1:

```

Figure 4.5: Sample output counterterm QGRAF topology for the process $e^+e^- \rightarrow \tilde{u}_R\tilde{u}_R^*$. The value GZURURCT is the placeholder for the NLO-QCD counterterm Feynman rule associated with the $Z\tilde{u}_R\tilde{u}_R^*$ vertex.

qgraf_nlo.out:

```

*--#[ diagram1:
*
Local diagram1 =
+ 1 *
inp([field.ep], idx1r1, p1) *
inplorentz(-1, iv1r1L1, p1, ZERO ) *
inpcolor(1, iv1r1C1) *
inp([field.em], idx1r2, p2) *
inplorentz(+1, iv1r2L1, p2, ZERO ) *
inpcolor(2, iv1r2C1) *
out([field.ur], idx2r3, p3) *
outlorentz(+0, iv2r3L0, p3, MUR ) *
outcolor(1, iv2r3C3) *
out([field.urx], idx3r4, p4) *
outlorentz(-0, iv3r4L0, p4, MUR ) *
outcolor(2, iv3r4C3) *
vertex(iv1,GZL ,ONE,
[field.ep], idx1r1, -1, p1, iv1r1L1, -1, iv1r1C1,
[field.em], idx1r2, +1, p2, iv1r2L1, +1, iv1r2C1,
[field.z], idx1r3, +2, -p1-p2, iv1r3L2, +1, iv1r3C1) *
vertex(iv2,GC ,ONE,
[field.g], idx2r1, +2, -k1, iv2r1L2, +8, iv2r1C8,
[field.ur], idx2r2, +0, k1+p3, iv2r2L0, +3, iv2r2C3,
[field.urx], idx2r3, -0, -p3, iv2r3L0, -3, iv2r3C3) *
vertex(iv3,GGZURUR ,DUM ,
[field.z], idx3r1, +2, p1+p2, iv3r1L2, +1, iv3r1C1,
[field.g], idx3r2, +2, k1, iv3r2L2, +8, iv3r2C8,
[field.urx], idx3r3, -0, -k1-p3, iv3r3L0, -3, iv3r3C3,
[field.ur], idx3r4, +0, -p4, iv3r4L0, +3, iv3r4C3) *
prop([field.z], ZERO, idx3r1, idx1r3) *
propcolor(+1, iv3r1C1, iv1r3C1) *
proplorentz(+2, -p1-p2, ZMASS , iv3r1L2, iv1r3L2) *
prop([field.ur], idx3r3, idx2r2) *
propcolor(+3, iv3r3C3, iv2r2C3) *
proplorentz(+0, k1+p3, MUR , iv3r3L0, iv2r2L0) *
prop([field.g], idx3r2, idx2r1) *
propcolor(+8, iv3r2C8, iv2r1C8) *
proplorentz(+2, -k1, ZERO , iv3r2L2, iv2r1L2)
;
*--#[ diagram1:

```

Figure 4.6: Sample output loop QGRAF topology for the process $e^+e^- \rightarrow \tilde{u}_R\tilde{u}_R^*$.

run_golem.pl

```
#### gluon, t channel
if($gluon_tchannel==0) { if($colorindex==8)
  {if($string3[1] =~/.proplorentz\(\+2, \-p1\+p3, ZERO./)
  {$discard = True;}}
}
if($gluon_tchannel==0) { if($colorindex==8)
  {if($string3[1] =~/.proplorentz\(\+2, p1\ -p3, ZERO./)
  {$discard = True;}}
}
if($gluon_tchannel==0) { if($colorindex==8)
  {if($string3[1] =~/.proplorentz\(\+2, \-p2\+p4, ZERO./)
  {$discard = True;}}
}
if($gluon_tchannel==0) { if($colorindex==8)
  {if($string3[1] =~/.proplorentz\(\+2, p2\ -p4, ZERO./)
  {$discard = True;}}
}
```

Figure 4.7: Sample topological selection PERL code in *run_golem.pl*.

Data management

Along with the produced topological diagrams, QGRAF produces an additional input file for *run_golem.pl* to read, *golem_input.dat*, which contains the basic external particle information gathered from the QGRAF output. An example *golem_input.dat* file for the process $e^+e^- \rightarrow \tilde{u}_R\tilde{u}_R^*$ is given in Fig. 4.8. The data in *golem_input.dat* allows *run_golem.pl* to calculate the normalisation factors for the initial particles, and determine the basis of color structures that will arise from color flow decomposition.

The normalisation factors $n! \cdot n_1 \cdot n_2$ that precede the amplitude-squared as shown in Eq. (4.1), are calculated from the *golem_input.dat* data:

golem_input.dat:

#Name	Sign	2Spin	Color	Mass
ep	-	1	1	ZERO
em	-	1	1	ZERO
ur	+	0	3	MUR
urx	+	0	3	MUR

Figure 4.8: *golem_input.dat* file for the process $e^+e^- \rightarrow \tilde{u}_R\tilde{u}_R^*$.

```
$SPIN_COL_AVG = $color[0]*$color[1]*$spinavg[0]*$spinavg[1]*$symfac;
```

For massless vectors, $spinavg = 2spin$, and for all other particles $spinavg = 2spin + 1$. The symmetry factor is an additional normalisation required for identical final state particles: $symfac = 2$ for identical particles, $symfac = 1$ for distinct particles.

The color basis that will arise from color flow decomposition methods discussed in Section 2.5 can be estimated by the description:

$$\prod_{k=1}^{n_8} \left(T^{a(k)} \right)_{i(k)}^{j(k)} \longrightarrow \sum_{\sigma \in S_{n_8}} \left(\delta_{i(1)}^{j(\sigma(1))} \delta_{i(2)}^{j(\sigma(2))} \dots \delta_{i(n_8)}^{j(\sigma(n_8))} \right). \quad (4.4)$$

In other words, for n_8 external color octets (and n_3 external color triplets) there will be n_8 unsaturated adjoint generators, which once saturated according to the color flow decomposition method (as seen in Eqs. (2.33) & (2.36)), will produce $(n_8 + n_3)!$ sets of color basis structures (from Eq. (2.37)). It is then simply a case of using the color information of the external particles, as provided by *golem_input.dat*, to produce the fully permuted set of color basis structures for the process amplitude.

For each set of diagrams (LO, counterterm, loop), *run_golem.pl* produces a FORM and MAPLE formatted input file *INPUT_DATA_TREE|COUNTER|LOOP.h|map*. This file contain all the relevant process information required by the following algorithms: number of diagrams, external particle types and masses, normalisation factors, color bases, and several other flags for debugging and simplification purposes. In order for FORM to process the QGRAF-written diagrams for application of Feynman rules, *run_golem.pl* also creates a list of FORM declarations for all the model-specific parameters (fields, masses, couplings, widths) found in the diagrams. If the flag `nlosimp` in *proc_card.dat* is enabled (default =1), several files containing FORM identities simplifying the model couplings (setting vanishing couplings to zero, merging identical couplings) will also be produced, which can improve the computation efficiency by several factors depending on the model and process concerned.

Application of Feynman rules

Once the QGRAF diagrams have been produced and filtered by *run_golem.pl*, the remaining diagrams are processed by a set of FORM routines, which will apply the model-independent Feynman rules specified in Section 2.3.

The FORM files *qgraf_tree|counter|loop.frm* take in each QGRAF-produced diagram² one at a time by use of the command (for LO):

²Examples given in Figs. 4.4, 4.5 & 4.6.

```
#include 'SUBDIR'qgraf_lo.out # diagram'num'
```

where 'SUBDIR' is the directory of the QGRAF output, as referenced in (for LO) *INPUT_DATA_TREE.h*. The QGRAF diagrams are processed as follows:

1. The two outgoing particles are transformed into incoming particles by reversing the momenta:

```
argument vertex, proplorentz;
id p3 = -p3;
id p4 = -p4;
endargument;
.sort
```

This enforces conservation of momentum as in Eq. (2.20).

2. By evaluating the `vertex()` functions in QGRAF diagrams one can determine the coupling constant, $2spin$, momentum (directed into the vertex), and color type of each leg within the vertex. This is all the information needed to apply the model-independent Feynman rules for vertices as derived in Section 2.3. Within each `vertex()` function are three coordinates for each leg, which connect the topology, Lorentz structure and color structure of the vertex to other parts of the diagram. A sample FORM identity for the 3-color octet triple vector vertex is given below:

```
*---#[ VVV vertex (QCD) :
id once vertex(iv?, g?, ONE,
    field1?, idx1?, 2, k1?, iv1L?, 8, iv1C?,
    field2?, idx2?, 2, k2?, iv2L?, 8, iv2C?,
    field3?, idx3?, 2, k3?, iv3L?, 8, iv3C?) =
    - g * (
        + d(iv1L, iv2L) * (k1(iv3L) - k2(iv3L))
        + d(iv2L, iv3L) * (k2(iv1L) - k3(iv1L))
        + d(iv3L, iv1L) * (k3(iv2L) - k1(iv2L))
    ) * f(iv1C, iv2C, iv3C) * node(idx1, idx2, idx3);
*---#[ VVV vertex (QCD) :
```

In the case of vertex counterterms tagged by CT, the coupling constant (e.g. GG1CT) is replaced at a later stage by the counterterm Feynman rule.

3. The `inp|out()` function, combined with `inp|outlorentz()` and `inp|outcolor()` functions describes the incoming|outgoing external particles: these functions contain the *2spin*, momentum, mass and color type of the external particle, which is enough to apply the model-independent Feynman rules for external particles as derived in Section 2.3. Within the three functions are coordinates to link the topology, Lorentz structure and color structure to other parts of the diagram as before. Note that formerly outgoing fermions are now reversed, so that only $u(p)$ & $\bar{v}(p)$ are produced. In the case of external gluon counterterms tagged by `WF`, a multiplicative factor `WFcorr([field.g])` is created with a standard external gluon; this factor will be replaced at a later stage by the external counterterm Feynman rule.
4. The `prop()` function, along with the `propcolor()` and `proplorentz()` functions, contains the information required to apply model-independent Feynman rules for the internal propagators of the diagram. These functions contain the *2spin*, momentum, mass, particle width (if available) and color type of the propagator, along with the Lorentz, color and topological coordinates. In the case of `CT`-tagged propagators: as the self-energy counterterm cannot be factorised into a standard propagator multiplied by a kinematically-invariant term, propagators tagged by `CT` have to be transformed into the product of two propagators separated by a kinematically-dependent structure with counterterm placeholders. For the example of the fermionic self-energy counterterm, the translation is:

$$\begin{aligned} & \text{prop}(\text{field?}, \dots) \cdot \text{proplorentzCT}(\text{sign?}\{-1,1\}, \text{p?}, \text{m?}, \text{i?}, \text{j?}) \\ \longrightarrow & \quad -i \frac{\not{p} + m}{(p^2 - m^2)} [(\not{p} - m) \delta Z - \delta m] \frac{\not{p} + m}{(p^2 - m^2)} \delta_i^j, \end{aligned} \quad (4.5)$$

where values for δZ and δm are taken from the model-dependent file *selfenergy-ct.dat*, and replaced at a later stage.

5. After each of the individual terms (vertices, external particles, propagators) is translated by FORM, they need to be linked using the coordinate structure, to ensure that the non-commutative Dirac structure and fermion flow is correctly applied throughout the diagram. Individual non-commutative structures designated by `NCContainer(?a,iv1?,iv2?)`³ are connected by coordinates `iv1` and `iv2`:

```
id NCContainer(?a,iv1?,iv2?) NCContainer(?b,iv2?,iv3?) =
    NCContainer(?a,?b,iv1?,iv3?);
```

³Using notation inspired by [58].

GRAPHS_MGGOLEM_LOOP.h:

```

G diagram1 =
  Den(- k1 - k2,ZMASS2,ZMASS,0) * intM( Den(q1,0),Den(k3 + q1,MUR2) ) *
  SUNSum(Glu9,8) * SUNSum(Col10,3) * SUNT(Glu9,Col3,Col10) *
  SUNT(Glu9,Col10,Col14) * GC * GGZURUR * scalar3 * scalar4 * Pi^(-2) *
  (
    1/16 * Spinor(k1,0,-1) * g_(2,7_,k3) * Spinor(k2,0,1) * GZL2
    + 1/32 * Spinor(k1,0,-1) * g_(2,7_,q1) * Spinor(k2,0,1) * GZL2
    + 1/16 * Spinor(k1,0,-1) * g_(2,6_,k3) * Spinor(k2,0,1) * GZL1
    + 1/32 * Spinor(k1,0,-1) * g_(2,6_,q1) * Spinor(k2,0,1) * GZL1 );

```

Figure 4.9: Sample loop Feynman diagram translated from QGRAF output in Fig. 4.6

These structures have natural ending points in either the external fermions, or are self-connecting as a closed fermion loop (which introduces a factor -1). After all `NCContainer()` functions have been merged, the internal Dirac algebra ($\gamma^\mu, \not{p}, \Pi_{R/L}, \mathbf{1}$) is correctly ordered, and defines the fermion flow. When Majorana particles are present, the individual coordinates in `NCContainer()` may not align correctly. In this case special care is taken to ensure that the fermion flow is correctly aligned:

```

id NCContainer(?a,iv1?,iv2?) NCContainer(?b,iv3?,iv2?) =
  NCContainer(?a,flip(?b),iv1?,iv3?);

```

where `flip()` is defined using the flipping rules as defined in Eq. (2.29).

6. FORM finally cleans up the diagram: removing topological remnants, contracting color and Lorentz indices and applying simplification identities (enabled by `nlosimp=1`) throughout.

After processing all QGRAF diagrams in this manner, three FORM files (*GRAPHS_MGGOLEM_TREE|COUNTER|LOOP.h*) containing the full set of LO, counterterm and loop Feynman diagrams⁴ for the NLO-QCD process are written. A sample loop Feynman diagram for the process $e^+e^- \rightarrow \tilde{u}_R\tilde{u}_R^*$ is given in Fig. 4.9, translated from the QGRAF output in Fig. 4.6. The correspondence between the

⁴Using notation inspired by [59].

MADGOLEM FORM functions and the standard Feynman algebra is defined:

$$\begin{aligned}
 \text{SUNT}(a, i, j) &= (T^a)_i^j, & \text{SUNF}(a, b, c) &= f^{abc}, & \text{SUMSum}(a|i, 8|3) &= \sum_{a|i}^{8|3}, \\
 \text{Spinor}(k, m, \pm 1) &= \bar{v}|u(k, m), \\
 \text{Den}(p, m^2, m, \Gamma) &= \frac{1}{(p^2 - m^2 + im\Gamma)}, \\
 \text{intM}(\text{Den}(), \dots, \text{Den}()) &= \int \frac{d^D q}{i\pi^{D/2}} \frac{1}{N_1 \dots N_n}.
 \end{aligned} \tag{4.6}$$

Dirac algebra is contained within the special FORM function $\mathbf{g}(\cdot)$ [56].

4.2.2 Partial amplitude factorisation

The script *run_golem.pl* now processes the Feynman diagrams found in *GRAPHS_MGGOLEM_TREE|COUNTER|LOOP.h*, using FORM and MAPLE algorithms to factorise the diagrams into partial amplitudes, *i.e.* a basis of color, helicity and tensor structures. Here the LO and counterterm diagrams can be handled in an identical manner, as they are both tree-level diagrams with no tensorial structures, and the CT-tagged placeholders are scalar invariants and thus unaffected by the following techniques. Therefore first the processes applied to the LO and counterterm diagrams will be described, followed by the loop diagrams.

LO and counterterm diagrams

The key steps run by *run_golem.pl* to factorise LO and counterterm diagrams are:

1. The MAPLE algorithm *PROCESS_TREE|COUNTERGRAPH_INFO.map* calculates the full set of dot products for the external momenta ($k_i \cdot k_j$) in terms of the Mandelstam variables ($s = \mathbf{s}(\mathbf{k}1, \mathbf{k}2)$, $t = \mathbf{s}(\mathbf{k}1, \mathbf{k}3)$, $u = \mathbf{s}(\mathbf{k}2, \mathbf{k}3)$)⁵, and external masses. This is computed from the external masses defined in *INPUT_DATA_TREE|COUNTER.map*. Due to conservation of the external momentum, the dot products can be defined purely in terms of two out of the three Mandelstam variables; these definitions are output as FORM identities in *kikj_LO|COUNTER.h*.
2. The FORM script *ALL_TREE|COUNTER_GRAPHS.frm* calls the color flow decomposition algorithms contained in *COLOR_ALGEBRA.h*. These algorithms apply the techniques discussed in Section 2.5 to saturate the external color octets,

⁵Where $\mathbf{s}(i, j) \equiv s_{ij} = (k_1 + k_2)^2$.

and evaluate the resulting color structures that factorise out from the amplitude as shown in Eq. (2.36). The factorised color structure for each diagram (numbered `GNUM`) is output to a MAPLE file `COLOUR_INFO_G'GNUM'_LO|COUNTER.mapout`, ordered by the number of separate color bases.

3. The FORM script `ALL_TREE|COUNTER_GRAPHS.frm` calls the spinor helicity algorithms contained in `HELICITY_PROJECTION_AUTO.h` to process the remaining kinematic partial amplitude using the techniques discussed in Section 2.6. These algorithms apply Eqs. (2.73) & (2.75) to each pair of external fermions connected by an unbroken fermion line ($\bar{v}(k_j)$ & $u(k_i)$), choosing the reference momentum ($q = \$kref$) to be equal to another external light-like momentum in the process:

```
#if      ( 'i'<3 && 'j'<3 )
  #$kref=k3b;
#elseif  ( 'i'=1 && 'j'>2 ) || ( 'i'>2 && 'j'=1 )
  #$kref=k2b;
#else
  #$kref=k1b;
#endif
.sort
```

Here `k3b` is the light-like momentum of k_3 , as defined by the mass decomposition formula (Eq. (2.61)). Each external vector boson $\epsilon^\mu(k_j)$ is processed using Eq. (2.77), choosing the reference momentum ($q = \$kref$), and ‘glueing’ momentum ($n = \$kglue$) to be other light-like external momenta:

```
#if      'j'=1
  #$kref=k2b;
  #$kglue=k3b;
#elseif  'j'=2
  #$kref=k1b;
  #$kglue=k3b;
#else
  #$kref=k1b;
  #$kglue=k2b;
#endif
.sort
```

External scalars, while not requiring helicity projections, are also assigned a reference momenta in order to calculate the light-like momentum by mass decomposition (Eq. (2.61)):

```
#if      'j'=1
  #kref=k2b;
#else
  #kref=k1b;
#endif
.sort
```

By the end of *HELICITY_PROJECTION_AUTO.h*, the kinematic partial amplitude is factorised into separate helicity-dependent components that are composed of Dirac traces with a bra-ket spinor product prefactor.

There is one caveat that arises when choosing the reference vectors for the external fermion pairs: when the process being calculated has four external fermions, and contains diagrams featuring all three topological channels of momentum transfer (s-channel, t-channel, u-channel), *the above choices of reference momenta do not work*.

The explanation for this is that the reference momentum for each fermion pair has to be chosen consistently for each diagram in the process, in order for the physical observable (based on the amplitude-squared) to be reference-invariant. From Eq. (2.75) it can be seen that factors of

$$\frac{1}{[iq]}, \quad \frac{1}{\langle iq \rangle}, \quad \frac{1}{[jq]}, \quad \frac{1}{\langle jq \rangle} \quad (4.7)$$

appear when constructing helicity amplitudes from the connected fermion pairs $\bar{v}(k_J)\Gamma u(k_I)$. Because of these factors, the reference momentum q cannot be chosen equal to the light-like momentum of either external fermion in the pair, as the resulting helicity amplitudes will contain unphysical divergences as

$$[ii] = \langle ii \rangle = 0. \quad (4.8)$$

With this in mind, processes with four external fermions have the potential to produce diagrams containing *every possible fermion pair* (connected by the three topological channels: s-channel, t-channel, u-channel); in this case, *there is no legal combination of external momentum available to $q = \text{\$kref}$ that will not cause divergent results*. This situation also occurs when all four external particles

are massive, as it becomes impossible to produce a light-like reference momentum based on external momenta using Eq. (2.61).

To navigate this concern, when the process contains four external fermions or four massive particles *run_golem.pl* enables the option (`$k5 = 1`)⁶. This causes the spinor helicity algorithms to choose the reference vector for fermions and scalars to be

```
#$kref=k5;
```

The four-momentum `k5` corresponds to a fifth external momentum in the process k_5 , that is light-like and completely independent of the other external momenta. Due to this independence, the summation of the four standard momenta is still conserved:

$$\sum_{i=1}^4 k_i = 0, \quad (4.9)$$

however the kinematic structure of the partial helicity amplitude is no longer purely defined by the conventional Mandelstam variables, including also the terms ($s_{15} = s(k_1, k_5)$, $s_{25} = s(k_2, k_5)$, $s_{35} = s(k_3, k_5)$, $s_{45} = s(k_4, k_5)$)⁷.

4. After producing the helicity-dependent partial amplitudes, *ALL_TREE|COUNTER-GRAPHS.frm* runs *DIRACOCLOGY.h* to evaluate the Dirac traces contained in the helicity amplitudes using the FORM command `trace4` for each fermion line, and then applying the identities read from *kikj_LO|COUNTER.h* to reduce the helicity amplitude to a set of kinematically-invariant Mandelstam variables (with extra terms in the case of `k5`). Non-invariant terms $\sim i\varepsilon_{\mu\nu\rho\sigma} k_i^\mu k_j^\nu k_l^\rho k_m^\sigma$ will also arise from the trace by Eq. (2.54). When k_5 is not present in the process, these terms vanish upon application of the identity $k_4 = -k_1 - k_2 - k_3$. When k_5 is present, one non-invariant term will survive: $i\varepsilon_{\mu\nu\rho\sigma} k_1^\mu k_2^\nu k_3^\rho k_5^\sigma$; this term is replaced by the placeholder `ck5EPSTENSOR` to be evaluated numerically during the Monte Carlo event generation process.
5. The resulting kinematically-invariant helicity-dependent partial amplitude (with non-invariant bra-ket prefactor) is output to a MAPLE file *GRAPH-'GNUM'_LO|COUNTER.mapout*, with respect to the factorised color bases denoted by `COLOURPOLY('GNUM', 'CNUM')`.

Sample factorised output from *GRAPH_2_LO.mapout* for the process $e^+e^- \rightarrow u\bar{u}$ is given in Fig. 4.10, where `COLOURPOLY(2, 1)` is defined in *COLOUR_INFO-G2_LO.mapout*

⁶This option can also be enabled by setting (`$k5debug = 1`) in the debugging options in *run_golem.pl*.

⁷Also calculated by *PROCESS_TREE|COUNTERGRAPH_INFO.map* when `$k5` is enabled.

GRAPH_2_LO.mapout:

```
G GRAPH2 = + POWER(S12,-1)*AMPINFO(2,4,-1,1,-1,1,0,0,0,0,k5,k5,k5,k5,
  InvSpaa(k1,k5)*InvSpaa(k3,k5)*InvSpbb(k2,k5)*InvSpbb(k4,k5))*
  COLOURPOLY(2,1) * ( GAL2*GAU2*S23*S25*S35 + GAL2*GAU2*S23*S15*S45
  + 2*ck5EPSTENSOR*GAL2*GAU2*S45 - 2*ck5EPSTENSOR*GAL2*GAU2*S35
  + 2*ck5EPSTENSOR*GAL2*GAU2*S25 - 2*ck5EPSTENSOR*GAL2*GAU2*S15 )
```

Figure 4.10: Partial output for *GRAPH_2_LO.mapout* for the process $e^+e^- \rightarrow u\bar{u}$.

as

```
COLOURPOLY[ 2,1 ]:= dd(Co14,Co13):
```

The function `AMPINFO()` in Fig. 4.10 contains information describing the diagram number (2), number of external particles (4), helicity basis $(-1, 1, -1, 1)$, external masses $(0, 0, 0, 0)$, reference momenta (k_5, k_5, k_5, k_5) , and bra–ket prefactor $(\text{InvSpaa}(k_1, k_5) * \text{InvSpaa}(k_3, k_5) * \text{InvSpbb}(k_2, k_5) * \text{InvSpbb}(k_4, k_5))$, where

$$\text{InvSpaa}(k_1, k_2) = \langle 12 \rangle^{-1}, \quad \text{InvSpbb}(k_1, k_2) = [12]^{-1}, \quad (4.10)$$

using a similar convention to [60].

Loop diagrams

The key steps run by *run_golem.pl* to factorise the loop diagrams into partial amplitudes are:

- For each diagram, the number of loop propagators and maximum tensor rank is recorded by the FORM file *PRE_PROCESSING_LOOP.frm*. This file also orders the loop denominators according to Eq. (3.1) and records the ordering of the internal k -momenta. The loop denominator `IntM(Den(), ..., Den())` is then rewritten into the scalar Passarino–Veltman terminology `A0i()`, `B0i()`, `C0i()`, `D0i()`, where the internal parameters record the internal momenta and masses.
- The MAPLE algorithm *PROCESS_LOOPGRAPH_INFO.map* calculates the full set of dot products for the external momenta as for the LO and counterterm case, and outputs the resulting identities to *kikj.h*. It also calculates the ordered loop r -momenta as defined in Eqs. (3.15) & (3.16), and produces the identities translating between the original internal k -momenta and the new r -momenta in *R2K.h* and *K2R.h*, as well as a list of dot product identities for r -momenta

in *rirj.h*. A file (*PVI2SI.h*) providing the translation from the Passarino–Veltman defined loop denominators $A0i()$, $B0i()$, $C0i()$, $D0i()$ into the r –momentum–based scalar integral terminology is also produced, of which an example is given below:

```
#if 'GNUM'=1
id C0i(k,k+k3, k-k4, mass3, mass1, mass2) =
    SI(D,3,r1, r3, 0, mass1, mass2, mass3);
#endif
.sort
```

- The MAPLE algorithm *CREATE_FUNLIST.map* predicts a complete MAPLE list (*FUNLIST.map*) of the possible scalar integrals and form factors that will be produced by the application of tensor reduction methods, and creates a FORM file *symmetrise_funargs.h* listing the potential symmetries in the resulting scalar integral and form factor arguments, based on the known r –momenta and internal masses.
- The FORM script *ALL_LOOP_GRAPHS.frm* (which can process `$num_workers` loop diagrams in parallel by multi–threading) calls the identities listed in *PV2SI.h* to translate the loop denominators into the r –momentum basis, followed by the color algorithms in *COLOR_ALGEBRA.h* (which outputs the MAPLE file *COLOUR_INFO_G'GNUM'_LOOP.mapout*), and the helicity projection techniques in *HELICITY_PROJECTION_AUTO.h*, which are already discussed above.
- After producing the color basis and helicity–dependent partial amplitudes, *ALL_LOOP_GRAPHS.frm* runs *DIRACOLOGY.h*, which applies the algorithms discussed in Section 3.2.1 to dimensionally–split the Dirac algebra in the numerator and separate the partial amplitude into \hat{D} – and \tilde{D} –dimensional terms, at the point before Dirac traces are evaluated⁸. *DIRACOLOGY.h* then applies the Dirac traces using Eq. (3.10) to avoid the 4–dimensional constraint on `trace4`. The identities from *kikj.h* are applied, reducing the helicity amplitude to a set of Mandelstam variables (including k_5 terms) and loop integral numerator terms of \hat{q}^2 , \tilde{q}^2 and $q \cdot k_i$.
- *ALL_LOOP_GRAPHS.frm* next translates the remaining k_i terms in the numerator into r –momenta by calling the file *K2R.h*. Note that the k_5 momentum (if

⁸Dimensional splitting is applied to the LO and counterterm diagrams as well, but has no ultimate effect as the corrections arising from this begin at $\mathcal{O}(\epsilon)$.

present) is unaffected by this. All terms $\sim \hat{q}^2$ are also rewritten in terms of the D - and \tilde{D} -dimensional loop momenta using Eq. (3.10) ($q^2 = \hat{q}^2 + \tilde{q}^2$):

```
id kh.kh = k.k - kt.kt;
```

This leaves the helicity-dependent partial tensor amplitude purely in terms of r -momenta, and ready for preliminary tensor simplification.

- *ALL_LOOP_GRAPHS.frm* applies the Passarino–Veltman (PV) reduction formalism as described in Subsection 3.3.1 to reduce the rank of the tensor integral, by calling *CancelPropInHex.h*. The algorithm that corresponds to the specific example given in Eq. (3.18) is:

```
id SI(D,3,r1?,r2?,0,em1?,em2?,em3?) * r1?.k^{1'} = r1.k^{'1'-1} *
( SI(D,2,r1,r2,em1,em2) - SI(D,2,r2, 0,em2,em3)
+ SI(D,3,r1,r2, 0,em1,em2,em3) * ( r1.r1-em1+em3 ) ) / 2;
```

Iterative application of the PV reduction algorithm in Eq. (3.19) ensures that each rank r_i n_i -point tensor integral can be reduced to a single scalar n_i -point integral, plus a set of rank $r_f < r_i$ tensor and scalar $n_f < n_i$ -point integrals.

In the cases where k_5 is not present, this implies that after PV reduction, (regardless of the process calculated) the most complicated 4-point integral remaining is the scalar box integral!

When k_5 is present however, the PV reduction formalism does not work on numerator terms $\sim q \cdot k_5$, as k_5 is not writeable in terms of r_i . In this case, it is possible for 4-point tensor integrals to survive the algorithms contained in *CancelPropInHex.h*.

After PV reduction, numerator terms are tidied up by calling *rirj.h* to rewrite any dot products $r_i \cdot r_j$ into Mandelstam variables and external masses.

- After PV reduction, *ALL_LOOP_GRAPHS.frm* calls *SI2TI.h*, which combines the loop momenta $q^{\mu_1} \dots q^{\mu_r}$ in the numerator with the scalar integral denominator $\text{SI}(D, n, \dots)$ to produce the rank r tensor integral function $\text{TI}(D, n, r, \mu_1, \dots, \mu_r, \dots)$. *SI2TI.h* also applies the higher-dimensional identities that arise from \tilde{D} -dimensional momenta \tilde{q}^2 in the numerator, as shown in Eq. (3.13).
- *ALL_LOOP_GRAPHS.frm* next calls the FORM file *TI2FF.prc*, to rewrite the n -point rank r tensor integrals $\text{TI}(D, n, r, \mu_1, \dots, \mu_r, \dots)$ in terms of the form

GRAPH_2_LOOP.mapout:

```
G GRAPH2 = + POWER(S12,-1)*AMPINFO(2,4,1,-1,1,-1,0,0,0,0,k5,k5,k5,k5,
  InvSpaa(k2,k5)*InvSpaa(k4,k5)*InvSpbb(k1,k5)*InvSpbb(k3,k5))*
  COLOURPOLY(2,1)*TC32(0,0,D,0,S12,0,0,0,0)*Pi^(-2)* eps *
  ( 1/4*ck5EPSTENSOR*GAL2*GG2^2*GAU2*S25
    - 1/4*ck5EPSTENSOR*GAL2*GG2^2*GAU2*S15 )
```

Figure 4.11: Partial output for *GRAPH_2_LOOP.mapout* for the process $e^+e^- \rightarrow u\bar{u}$.

factor representation as given in Eqs. (3.23)–(3.27). The form factors use the alternate notation $\text{TC}_{nr}(\{j\}, D, \dots)$, instead of $A|B|C_{\{j\}}^{m,r}(S)$.

- Lastly, *ALL_LOOP_GRAPHS.frm* tidies up the partial tensor amplitude by discarding terms of order $\mathcal{O}(\varepsilon)$ and returning the amplitude momenta to its original k_i basis using *R2K.h*. The resulting kinematically-invariant helicity-dependent partial loop amplitude rearranged into form factors (with non-invariant bracket prefactor) is output to the MAPLE file *GRAPH_‘GNUM’_LOOP.mapout*, with respect to the factorised color bases denoted by $\text{COLOURPOLY}(\text{‘GNUM’}, \text{‘CNUM’})$.

Sample factorised output from *GRAPH_2_LOOP.mapout* for the process $e^+e^- \rightarrow u\bar{u}$ is given in Fig. 4.11, where $\text{COLOURPOLY}(2,1)$ is defined in

COLOUR_INFO_G2_LOOP.mapout as

```
COLOURPOLY[ 2,1 ]:=dd(Co14,Co13)*NC^(-1)-dd(Co14,Co13)*NC:
```

The form factor $\text{TC32}(0,0,D,0,S12,0,0,0,0)$ here corresponds to the conventional form factor $B^{3,2}(S)$.

4.2.3 Tensor reduction

After having factorised and simplified the Feynman diagrams into partial amplitudes based on color basis, helicity dependency and tensorial structure, the script *run_golem.pl* prepares for a full reduction of the tensor integrals found in the loop diagrams (as written in *GRAPH_‘GNUM’_LOOP.mapout*).

run_golem.pl calls the MAPLE script *CREATE_FULL_REDUCTION_CODE.map*, which creates a set of FORM projections *PROJECT_GRAPH_‘GNUM’_C_‘CNUM’_E1.frm* for each partial amplitude, ordered by color basis ‘CNUM’ and diagram number ‘GNUM’. These projection algorithms are designed to input the loop diagram FORM files *GRAPH_‘GNUM’_LOOP.mapout* and read off the terms corresponding to $\text{COLOURPOLY}(\text{‘GNUM’}, \text{‘CNUM’})$.

For processes where k_5 is not present, the potential tensor integrals to be processed are at most 3-point triangles, as the PV reduction formalism applied previously by *CancelPropInHex.h* ensures that all 4-point box tensors are completely reduced to the scalar box integral plus a set of $n < 4$ -point tensor and scalar integrals. In this case, the projection algorithms in *PROJECT_GRAPH_‘GNUM’_C‘CNUM’_E1.frm* apply a comprehensive list of integral reduction algorithms as defined in Appendix B, which are fundamentally based on Eqs. (B.1) & (B.2), in order to reduce all tensor integrals to a fundamental set of purely-scalar integrals.

When k_5 is present however, it is possible for 4-point tensor integrals to survive the PV reduction algorithms, as discussed previously. In this situation, the projection algorithms do not contain definitions for the reduction of (potentially massive) 4-point tensor integrals; these integrals are therefore preserved in their form factor representation, and added to the fundamental set of integrals for later calculation by the GOLEM95 integral library.

After running the reduction algorithms, *PROJECT_GRAPH_‘GNUM’_C‘CNUM’_E1.frm* calls the FORM file *ExpandInEpsilon.h* to expand and simplify ε terms:

```
repeat;
if ( count( eps,1 )>2 ) discard;
id 1/(D-3) = 1 +2*eps/(D-3);
id 1/(D-2) = 1/2 +2*eps/(D-2)/2;
id 1/(D-1) = 1/3 +2*eps/(D-1)/3;
id 1/D      = 1/4 +2*eps/D/4;
endrepeat;
.sort
```

Scalar integrals that have a prefactor $\mathcal{O}(\varepsilon)$ such that the total term (when including known UV or IR divergences) is also of order $\mathcal{O}(\varepsilon)$ are removed here also. The file *symmetrise_funargs.h* is also called to collect integrals with symmetrically-equivalent arguments. The projection algorithms finally run through the list of possible fundamental scalar integrals and form factors as described in *FUNLIST.map* to collect the total coefficient of each, with respect to diagram number and color basis. When run by FORM the resulting coefficients are written into a set of MAPLE files named by the format *RES_G‘GNUM’_C‘CNUM’_E1F‘FNUM’.mapout*, where ‘FNUM’ is the assigned integral number from the total set of contributing integrals in the amplitude.

Once *CREATE_FULL_REDUCTION_CODE.map* has created the projection algorithms, *run_golem.pl* runs them iteratively over all loop diagrams and color bases⁹, using multi-threading to process `$num_workers` loop diagrams in parallel.

⁹It is known that the maximum number of color bases per diagram is 3^k , where k is the maximum

The end result is a set of MAPLE files (*RES_G'GNUM' C' CNUM' E1F'FNUM'.mapout*), containing the kinematically-invariant helicity- and color-dependent coefficients for each fundamental scalar integral, form factor and finite term, henceforth referred to as the set of *basis functions* $F(1/\varepsilon^2, 1/\varepsilon, 1)$ (where the parameters $(1/\varepsilon^2, 1/\varepsilon, 1)$ describe the analytical structure of these one-loop functions in terms of their UV- & IR-pole Laurent coefficients).

4.2.4 Analytical results

Having produced a complete analytical reduction of the LO, counterterm and loop diagrams into partial amplitudes factorised by color & helicity basis, and in the case of loop diagrams also basis functions, *run_golem.pl* can now run algorithms to collect the partial amplitudes into a set of MAPLE files (*AMP_TREE.mapout*, *AMP_LOOP.mapout*, *AMP_LOOP_CT.mapout*), that present the diagram amplitudes in the format

$$\mathcal{M}_{\text{LO}} = \sum_{\sigma \in S_n} \sum_{\{\lambda\} \in S_\lambda} (\mathcal{M}_{\text{LO}})_{\sigma}^{\{\lambda\}} \cdot |c_\sigma\rangle, \quad (4.11)$$

$$\mathcal{M}_{\text{CT|LOOP}} = \sum_{\sigma \in S_n} \sum_{\{\lambda\} \in S_\lambda} \sum_{F \in S_F} (\mathcal{M}_{\text{CT|LOOP}})_{\sigma, F}^{\{\lambda\}} \cdot F(1/\varepsilon^2, 1/\varepsilon, 1) \cdot |c_\sigma\rangle. \quad (4.12)$$

In this format, the amplitudes are clearly factorised into a set of purely kinematic terms (also known as coefficients), within a basis of color $|\sigma\rangle$ (Eq. (2.36)), helicity $\{\lambda\}$ (Eq. (2.78)), and function $F(1/\varepsilon^2, 1/\varepsilon, 1)$.

The collection of the LO, counterterm and loop partial amplitudes is performed by the MAPLE algorithms *COLLECT_TREES.map*, *COLLECT_COUNTER.map* and *COLLECT_LOOP_part.map*:

COLLECT_TREES.map constructs the set of potential color and helicity bases from the information provided by *INPUT_DATA_TREE.map*, and then reads in each LO partial amplitude stored in *GRAPH_'GNUM'_LO.mapout*, looped over the number of LO diagrams 'GNUM'. Within each LO diagram *COLLECT_TREES.map* iterates over the individual helicity amplitudes (noted by `AMPINFO(GNUM,4,heli_ids,...)`, where `heli_ids` is an array containing the helicity basis) and color bases (the number of which is defined in *COLOUR_INFO_G'GNUM'_LOOP.mapout*), and stores the helicity- and color-dependent partial amplitude in the output file *AMP_TREE.mapout*:

```
GRAPH_TREE[GNUM,HNUM,CNUM] := partial_amp:
SPINOR_FAC[GNUM,HNUM] := bra_ket_prefactor:
```

number of possible 4-gluon vertices. For a $2 \rightarrow 2$ NLO-QCD process, iteratively running over 9 possible color bases is adequate.

These two terms correspond to $(\mathcal{M}_{\text{LO}})_\sigma^{\{\lambda\}}$ in Eq. (4.11), factorised into the kinematically-invariant Mandelstam variables and the non-invariant bra-ket helicity prefactor:

$$(\mathcal{M}_{\text{LO}})_\sigma^{\{\lambda\}} \rightarrow \text{bra_ket_prefactor}(\langle ij \rangle, [ij]) \cdot \text{partial_amp}(s(i, j)) . \quad (4.13)$$

After *AMP_TREE.mapout* is created, a second MAPLE algorithm *COLLECT_SIMPLIFY_TREE.map* is run to further simplify the content. The algorithm removes all helicity bases that do not have any non-zero partial amplitudes; if the flag `nlosymsimp` is enabled in the MADGRAPH process card *proc_card.dat*, it also checks the partial amplitudes for the potential helicity hermitian symmetry:

$$\mathcal{M}_{\text{LO}}^{\{\lambda\}} = (\mathcal{M}_{\text{LO}}^{\{\lambda'\}})^* \quad (4.14)$$

where $\{\lambda'\} \neq \{\lambda\}$. Only the unique helicity amplitudes are listed, along with a note of which helicities are hermitian. This simplification can greatly reduce the size of the output for processes, but must be disabled for processes which are helicity hermitian at LO but non-hermitian at loop level¹⁰. At the top of *AMP_TREE.mapout* is a header containing all relevant LO process information, an example of which is given in Fig. 4.12 for the process $e^+e^- \rightarrow u\bar{u}$.

COLLECT_COUNTER.map functions in an identical manner to the LO counterpart in collecting the helicity- and color- dependent partial amplitudes from *GRAPH_‘GNUM’_COUNTER.mapout*, (looped over the number of counterterm diagrams ‘GNUM’) and listing them in the output file *AMP_COUNTER.mapout*. Rather than running a second simplification algorithm on the output, *COLLECT_COUNTER.map* uses the non-zero helicity basis read directly from the simplified *AMP_TREE.mapout*. The CT-tagged counterterm placeholders within the partial amplitudes, being composed of fundamental scalar integrals, are implemented at the same time as the collection and simplification of the loop amplitudes.

COLLECT_LOOP_part.map acts to collect the coefficients of all fundamental scalar integrals and form factors present in both the loop *and* counterterm partial amplitudes, ordered by color and helicity basis. The MAPLE script reads the non-zero helicity basis directly from *AMP_TREE.mapout*, which when (`nlosymsimp=1`) can lead to a greatly reduced coefficient output and a large reduction in collection time. Next *COLLECT_LOOP_part.map* reads in *AMP_COUNTER.mapout*, and matches the CT-tagged counterterm placeholders with the model-specific renormalisation constant list contained in *CT_list_mod.map*. This list contains the precise analytical values for all CT-tagged coupling, self-energy and external gluon counterterms in the model, written

¹⁰*E.g.* the MSSM NLO-QCD process $pp \rightarrow \tilde{g}\tilde{g}$ for chirally-unique squark masses, which contribute in otherwise helicity hermitian diagrams

AMP_TREE.mapout:

```

# FILE CREATED BY COLLECT_SIMPLIFY_TREE.map on 2013-02-23
ExternalMasses := [0, 0, 0, 0]:
NUM_TREE_GRAPHES := 2:
NCOLS := 1:
COL[ 1] := dd(Col4,Col3):
#
# 4 unique helicity amplitudes found
NUM_HELIS := 16:
base_helis := [6, 7, 10, 11]:
unique_helis := [6, 7, 10, 11]:
symmetry_helis := []:
HELI[ 6]:= [1, -1, 1, -1]:
HELI[ 7]:= [1, -1, -1, 1]:
HELI[ 10]:= [-1, 1, 1, -1]:
HELI[ 11]:= [-1, 1, -1, 1]:
#
ReferenceVector := [k5, k5, k5, k5]:
NUM_TREE_GRAPHES := 2:
#
# GRAPH_TREE has indices: NGRAPH,NHELI,NCOL
GRAPH_TREE[ 1, 6, 1] := -GZL1*GZU1*(S12*S35*S45+S12*S15*S25
-S23*S25*S35-S23*S15*S45+2*ck5EPSTENSOR*S45-2*ck5EPSTENSOR*S35
+2*ck5EPSTENSOR*S25-2*ck5EPSTENSOR*S15)/(S12-ZMASS2):
SPINOR_FAC[ 1, 6] := InvSpaa(k2,k5)*InvSpaa(k4,k5)*
InvSpbb(k1,k5)*InvSpbb(k3,k5):

```

Figure 4.12: Example *AMP_TREE.mapout* header for the process $e^+e^- \rightarrow u\bar{u}$, after being simplified by *COLLECT_SIMPLIFY_TREE.map*.

in the style

$$1 + \delta Z \rightarrow 1 + \text{CT_prefac} * (\text{CT_integral} + \text{CT_finite}). \quad (4.15)$$

Counterterms depending on several fundamental scalar integrals can be simply split and written down in different rows, one per loop function. This matching allows the counterterm partial amplitudes to finally be written in terms of the fundamental scalar integrals; from this point on they can be processed in exactly the same manner as the partial loop amplitudes.

Iterating over number of loop diagrams, color bases, and the maximum number of possible fundamental scalar integrals and form factors (as listed in *FUNLIST.mapout* and the diagram-specific *FUNLIST_GRAPHS.mapout*), *COLLECT_LOOP_part.map* reads in each partial loop amplitude from *RES_G'GNUM'C'CNUM'E1F'FNUM'.mapout*. Within each partial loop amplitude returned MAPLE extracts the individual helicity amplitudes (as explained for *COLLECT_TREES.map*), and stores the partial amplitudes for each diagram in the output file *AMP_LOOP_'GNUM'.mapout*, written as a series of kinematic coefficients of basis functions sorted by helicity, color basis, and function (scalar integral, form factor or 1) (as shown in Eq. (4.12)). For each counterterm diagram CTNUM, *COLLECT_LOOP_part.map* also stores the counterterm partial amplitudes in the output file *AMP_LOOP_CT_'CTNUM'.mapout*, formatted identically to the loop diagrams.

The coefficient output corresponding to $(\mathcal{M}_{\text{CT|LOOP}})_{\sigma,F}^{\{\lambda\}}$ is formatted in a similar manner to the LO output:

```
GRAPH_COEFF [GNUM,HNUM,CNUM,1,FNUM] := partial_amp:
SPINOR_FAC [GNUM,HNUM] := bra_ket_prefactor:
```

$$(\mathcal{M}_{\text{CT|LOOP}})_{\sigma,F}^{\{\lambda\}} \rightarrow \text{bra_ket_prefactor}(\langle ij \rangle, [ij]) \cdot \text{partial_amp}(s(i,j)). \quad (4.16)$$

Here each coefficient *GRAPH_COEFF [GNUM,HNUM,CNUM,1,FNUM]* corresponds to a basis function $F(1/\varepsilon^2, 1/\varepsilon, 1)$, the full set of which (ignoring internal parameters) is:

```
ONE, SINGLEPOLEd4 , SINGLEPOLEIRd4,
TADd4(), BUBd4(), TRId4(), BOXd4(),
TC41(), TC42(), TC43(), TC44(),
TC31(), TC32(), TC33(), TC21(), TC22(), TC11().
```

Extra basis functions may be introduced by the counterterm renormalisation constants defined in *CT_list_mod.map*. The MAPLE script *COLLECT_SIMPLIFY_LOOP_part.map* simplifies the content in *AMP_LOOP_CT|LOOP_'CTNUM'|'GNUM'.mapout*, removing all basis functions that don't have non-zero coefficients. A header is also written to the top

of every counterterm and loop output, containing all relevant NLO process information (such as the non-zero function basis), an example of which is given in Figure 4.13 for the process $e^+e^- \rightarrow u\bar{u}$.

4.2.5 Numerical evaluation

By using the analytical results in the MAPLE output files, it is straightforward to produce a FORTRAN90 module (*virtual_corrections.f90*) that allows the analytical partial amplitudes to be numerically evaluated, producing the virtual component of the NLO-QCD matrix element-squared in the form of a numerical array $[a_0, a_1, a_2]$ corresponding to Eq. (4.1):

$$\begin{aligned} \frac{2}{n!n_1n_2} \text{Re} \left[(\mathcal{M}^{\text{virtual}})^\dagger \mathcal{M}_{\text{LO}} \right] &= \frac{2}{n!n_1n_2} \sum_{\sigma \in S_n} \sum_{\sigma' \in S_n} \sum_{\{\lambda\} \in S_\lambda} \sum_{F \in S_F} \langle c_\sigma | c_{\sigma'} \rangle \\ &\times \text{Re} \left[(\mathcal{M}_{\text{NLO}}^\dagger)_{\sigma, F}^{\{\lambda\}} (\mathcal{M}_{\text{LO}})_{\sigma'}^{\{\lambda\}} F(1/\varepsilon^2, 1/\varepsilon, 1) \right] \quad (4.17) \\ &= a_0 + a_1 \frac{1}{\tilde{\varepsilon}_{IR}} + a_2 \frac{1}{\tilde{\varepsilon}_{IR}^2} . \end{aligned}$$

The construction of the FORTRAN90 code *virtual_corrections.f90* that will provide the virtual component of the NLO-QCD matrix element-squared ($[a_0, a_1, a_2]$ via Eq. (4.17)), for event generation by Monte Carlo methods and combination with the integrated dipoles, is handled by the MAPLE scripts *CREATE_AUX_TREE.map*, *CREATE_AUX_LOOP_part.map* and *CREATE_GOLEM_part.map*, as well as a script *COLOUR_MATRIX.map* that constructs the color correlation matrix $\langle c_\sigma | c_{\sigma'} \rangle$:

$$\text{colour_correlation_matrix}(i, j) = \text{COL}[i] \cdot \text{COL}[j] , \quad (4.18)$$

where $\text{COL}[i]$ are read from the *AMP_*.mapout* files.

CREATE_AUX_TREE.map and *CREATE_AUX_LOOP_part.map* write and compile the individual partial amplitudes $(\mathcal{M}_{\text{LO}})_{\sigma'}^{\{\lambda\}}$ and coefficients $(\mathcal{M}_{\text{NLO}})_{\sigma, F}^{\{\lambda\}}$ (as read from the *AMP_.mapout* files) into a set of FORTRAN90 *shared object* (.so) library archives: *libcoeffs_all_tree.so*, *libcoeffs_all_ct_‘CTNUM’.so* and *libcoeffs_all_‘GNUM’.so*¹¹. These ‘coefficient libraries’ contain the individual coefficient functions, defined for LO and counterterm/loop as:

$$\begin{aligned} \text{function } c_{\{\lambda\}\text{-‘GNUM’}}_{\sigma'}(i1, i2, i3, i4, k, s) &\rightarrow (\mathcal{M}_{\text{LO}})_{\sigma'}^{\{\lambda\}} , \\ \text{function } c_{F\text{-}\{\lambda\}\text{-‘GNUM’}}_{\sigma'}(i1, i2, i3, i4, k, s) &\rightarrow (\mathcal{M}_{\text{NLO}})_{\sigma, F}^{\{\lambda\}} . \end{aligned} \quad (4.19)$$

¹¹Shared object (.so) libraries are used when MADGOLEM is installed on a standard LINUX system; for UNIX systems (such as MAC OSX) the dynamic library file (.dylib) is used instead.

AMP_LOOP_1.mapout

```

# FILE CREATED BY COLLECT_SIMPLIFY_LOOP_part.map on 2013-02-23
ExternalMasses := [0, 0, 0, 0]:
NUM_LOOP_GRAPHS := 2:
NUM_CTS := 2:
NCOLS := 1:
COL[ 1] := dd(Col4,Col3):
#
# Function basis has 4 elements, reduced from 7 elements.
NUM_LOC_FUNS := 4:
FUN[ 1] := ONE:
FUN[ 2] := BUBd4(0,0,0):
FUN[ 3] := BUBd4(S12,0,0):
FUN[ 4] := TRId4(S12,0,0,0,0,0):
#
# 4 unique helicity amplitudes found
NUM_HELIS := 16:
unique_helis := [6, 7, 10, 11]:
HELI[ 6] := [1, -1, 1, -1]:
HELI[ 7] := [1, -1, -1, 1]:
HELI[ 10] := [-1, 1, 1, -1]:
HELI[ 11] := [-1, 1, -1, 1]:
#
ReferenceVector := [k5, k5, k5, k5]:
FINAL_GRAPH_LIST := [1, 2]:
#
# GRAPH_COEFF has indices: NGRAPH,NHELI,NCOL,1,NFUN
GRAPH_COEFF[ 1, 6, 1, 1, 1] := 1/6*GZL1*GG2^2*GZU1*(S12*S35*S45+
S12*S15*S25-S23*S25*S35-S23*S15*S45+2*ck5EPSTENSOR*S45-2*ck5EPSTENSOR*
S35+2*ck5EPSTENSOR*S25-2*ck5EPSTENSOR*S15)/(S12-ZMASS2)/Pi^2:
GRAPH_COEFF[ 1, 6, 1, 1, 2] := -1/3*GZL1*GG2^2*GZU1*(S12*S35*S45+
S12*S15*S25-S23*S25*S35-S23*S15*S45+2*ck5EPSTENSOR*S45-2*ck5EPSTENSOR*
S35+2*ck5EPSTENSOR*S25-2*ck5EPSTENSOR*S15)/(S12-ZMASS2)/Pi^2:
GRAPH_COEFF[ 1, 6, 1, 1, 3] := -1/24*GZL1*GG2^2*GZU1*(S12*S15*S35
+S12*S15^2-5*S12*S35*S45+2*S12*S45*S15+S12*S45^2-5*S12*S15*S25
+S12*S25*S45+S23*S35^2+S23*S15*S35+2*S23*S45*S35+7*S23*S15*S45+S23*
S45^2+7*S23*S25*S35+S23*S25*S45+12*ck5EPSTENSOR*S35+12*ck5EPSTENSOR*
S15-12*ck5EPSTENSOR*S45-12*ck5EPSTENSOR*S25)/(S12-ZMASS2)/Pi^2:
GRAPH_COEFF[ 1, 6, 1, 1, 4] := 1/6*GZL1*GG2^2*GZU1*S12*(S12*S35*
S45+S12*S15*S25-S23*S25*S35-S23*S15*S45+2*ck5EPSTENSOR*S45-2*S35*
ck5EPSTENSOR+2*ck5EPSTENSOR*S25-2*ck5EPSTENSOR*S15)/(S12-ZMASS2)/Pi^2:
SPINOR_FAC[ 1, 6] := InvSpaa(k2,k5)*InvSpaa(k4,k5)*InvSpbb(k1,k5)*
InvSpbb(k3,k5):

```

Figure 4.13: Example *AMP_LOOP_1.mapout* header for the process $e^+e^- \rightarrow u\bar{u}$, after being simplified by *COLLECT_SIMPLIFY_LOOP_part.map*.

These functions can be referenced by the main numerical module *virtual_corrections.f90*, returning the numerical value for the kinematically-invariant partial amplitudes. Because of their potential size, the coefficient libraries are pre-compiled before the numerical calculation stage, and linked dynamically during calculation to avoid memory complications at the final stage of compilation.

The main FORTRAN module that is attached to the rest of the MADGOLEM code during numerical calculation is *virtual_corrections.f90*, which is created by the MAPLE script *CREATE_GOLEM_part.map*. *virtual_corrections.f90* contains the main subroutine `golem(k,mu,amplitude_array)`, which takes in the phase space values for the external momenta:

$$\mathbf{k} = \{k_1^\mu, k_2^\mu, k_3^\mu, k_4^\mu\} . \quad (4.20)$$

These momenta are configured for a $2 \rightarrow 2$ process (as opposed to $4 \rightarrow 0$), so the values of k_3^μ and k_4^μ are inverted before any calculation of the kinematic variables is performed. `golem()` also takes in the renormalisation scale (`mu = μ_R`) which is required for calculation of the fundamental scalar integrals and form factors in the loops, as well as renormalisation constants featuring logarithmic terms $\log(m^2/\mu_R^2)$.

`golem()` returns the array `amplitude_array`, which contains the values:

$$\mathbf{amplitude_array} = [a_0, a_1, a_2, a_B, a_{UV}] . \quad (4.21)$$

The first three terms in `amplitude_array` are the coefficients of the Laurent expansion in $\tilde{\epsilon}_{IR}$ of Eq. (4.17), corresponding to the finite term, simple pole and double pole respectively. In other words, the value a_0 is the entire finite contribution resulting from the NLO-QCD virtual correction to a process, ignoring contributions from the integrated dipole, for a given phase space point.

The MADGOLEM code that calls the `golem()` subroutine combines the first three terms (a_0, a_1, a_2) returned by `amplitude_array` with the three terms returned by the integrated dipole module (d_0, d_1, d_2). As a_1 and a_2 (likewise d_1 and d_2) provide the coefficients of the simple and double IR poles, a correct calculation of the virtual terms and the integrated dipoles requires that $d_1 = -a_1$ and $d_2 = -a_2$; this can be verified by MADGOLEM using an error-checker that generates warnings when $\Delta_{1|2} \geq 10^{-7}$:

$$\Delta_{1|2} = \frac{d_{1|2} + a_{1|2}}{d_{1|2} - a_{1|2}} . \quad (4.22)$$

After testing for successful IR subtraction, MADGOLEM calculates the finite virtual contribution to the partonic NLO-QCD differential cross section by combining the values a_0 and d_0 according to Eq. (4.3), and calculates the full partonic NLO-QCD

differential cross section by including the LO and real emission results (as in Eq. (4.2)):

$$\begin{aligned}
 d\hat{\sigma}^{\text{NLO}} &= d\hat{\sigma}^B \\
 &+ dPS_2(Q) \frac{1}{2s} [a_0 + d_0] \\
 &+ dPS_3(Q) \frac{1}{2s} \frac{1}{n_1 n_2} [|\mathcal{M}^{\text{real}}|^2 - \sum_j \mathcal{D}_j] .
 \end{aligned} \tag{4.23}$$

The fourth term returned by `amplitude_array` is the LO, or Born amplitude-squared:

$$a_B = \frac{1}{n!n_1n_2} |\mathcal{M}_{\text{LO}}|^2 . \tag{4.24}$$

This is included to allow for comparison with the equivalent value produced by MADGRAPH, as an extra form of error-checking. The fifth term returned by `amplitude_array` is the coefficient of the $1/\tilde{\varepsilon}_{UV}$ pole present in the virtual NLO amplitude-squared:

$$a_{UV} = \frac{2}{n!n_1n_2} \text{Re} \left[\left(\mathcal{M}^{\text{virtual}} \right)^\dagger \mathcal{M}_{\text{LO}} \right] . \tag{4.25}$$

If the virtual calculation is correctly renormalised, $a_{UV} = 0$. This can be verified by MADGOLEM using an error-checker that generates warnings when $\Delta_{UV} \geq 10^{-7}$:

$$\Delta_{UV} = \frac{a_{UV}}{a_B} . \tag{4.26}$$

The internal workings of `golem(k,mu,amplitude_array)` are arranged as follows: The matrix elements $\mathcal{M}^{\text{virtual}}$ and \mathcal{M}_{LO} that make up the results in `amplitude_array` are calculated using the algorithms as explained in Eq. (4.17):

$$\begin{aligned}
 \frac{2}{n!n_1n_2} \text{Re} \left[\left(\mathcal{M}^{\text{virtual}} \right)^\dagger \mathcal{M}_{\text{LO}} \right] &= \\
 \frac{2}{n!n_1n_2} \sum_{\sigma \in S_n} \sum_{\sigma' \in S_n} \sum_{\{\lambda\} \in S_\lambda} \sum_{F \in S_F} \langle c_\sigma | c_{\sigma'} \rangle & \\
 \times \text{Re} \left[\left(\mathcal{M}_{\text{NLO}}^\dagger \right)_{\sigma,F}^{\{\lambda\}} \left(\mathcal{M}_{\text{LO}} \right)_{\sigma'}^{\{\lambda\}} F(1/\varepsilon^2, 1/\varepsilon, 1) \right] . &
 \end{aligned} \tag{4.27}$$

The kinematic terms $\left(\mathcal{M}_{\text{NLO}}^\dagger \right)_{\sigma,F}^{\{\lambda\}}$ and $\left(\mathcal{M}_{\text{LO}} \right)_{\sigma'}^{\{\lambda\}}$ are contained as functions inside the library coefficient files as shown in Eq. (4.19), and the color correlation matrix $\langle c_\sigma | c_{\sigma'} \rangle$ is returned by the subroutine `colour_info(colour_correlation_matrix)`.

The basis functions $F(1/\varepsilon^2, 1/\varepsilon, 1)$ are declared as 4-dimensional arrays, containing

the numerical coefficients of the finite terms and UV & IR poles:

$$F(1/\varepsilon^2, 1/\varepsilon, 1) = [\#_1/\tilde{\varepsilon}_{IR}^2, \#_2/\tilde{\varepsilon}_{IR}, \#_3, \#_4/\tilde{\varepsilon}_{UV}] \quad (4.28)$$

Values for the four array components are calculated by the GOLEM95 integral library [43, 44], taking care to ensure that UV-divergent one- and two-point integral results are assigned to elements 3 and 4 in the array, and that IR-divergent three- and four-point integral results are assigned to elements 1, 2 and 3. Non-standard basis functions that are not recognised by GOLEM95, such as the scalar integral $B_0(0; 0, 0) \equiv 1/\tilde{\varepsilon}_{UV} - 1/\tilde{\varepsilon}_{IR}$, are defined manually within *CREATE_GOLEM_part.map*. As GOLEM95 does not include the renormalisation scale μ_R in its integral calculations, it must be introduced using the transformation:

$$F(1/\varepsilon^2, 1/\varepsilon, 1) \rightarrow (1 + \log(\mu_R) + \frac{1}{2}\log^2(\mu_R)) (1/\varepsilon^2, 1/\varepsilon, 1) . \quad (4.29)$$

Sample code for the calculation within GOLEM95 of the three-point scalar integral `TRId4(s(1, 2), 0, 0, 0, 0, 0)` is given in Fig. 4.14.

Having determined which functions to call for each element in Eq. (4.27), the steps taken in *virtual_corrections.f90* upon calling `golem()` to calculate `amplitude_array` are:

- The LO amplitude-squared value a_B in Eq. (4.24) is calculated by summing the helicity- and color-dependent amplitude-squared, and dividing by the normalisation factor $n!n_1n_2$:

$$\frac{2}{n!n_1n_2} \sum_{\{\lambda\} \in S_\lambda} \sum_{\sigma \in S_n} \sum_{\sigma' \in S_n} \text{Re} \left[(\mathcal{M}_{\text{LO}}^\dagger)_\sigma^{\{\lambda\}} \langle c_\sigma | c_{\sigma'} \rangle (\mathcal{M}_{\text{LO}})_{\sigma'}^{\{\lambda\}} \right] , \quad (4.30)$$

where $(\mathcal{M}_{\text{LO}})_\sigma^{\{\lambda\}}$ is evaluated by summing the helicity- and color-dependent LO functions contained in *libcoeffs_all_tree.so* over all LO diagrams. The key code that performs this task is shown in Fig. 4.15.

- The virtual NLO-QCD amplitude-squared values a_0 , a_1 , a_2 and a_{UV} in Eq. (4.27) are calculated by first combining the 4-dimensional basis functions $F(1/\varepsilon^2, 1/\varepsilon, 1)$ (Eq. (4.28)) with their respective helicity- and color-dependent kinematic coefficients (provided by the *libcoeffs* files), summed over all counterterms and loop diagrams. This produces a set of helicity- and color-dependent partial amplitudes that contain the UV & IR poles along with the finite results in a 4-dimensional array. The amplitude-squared is then constructed by contracting the helicity- and color-dependent virtual partial amplitudes with the

virtual_corrections.f90

```

call initgolem95(3)
!
s_mat(1,1) = zero
s_mat(1,2) = zero
s_mat(1,3) = s(1,2)
s_mat(2,1) = s_mat(1,2)
s_mat(2,2) = zero
s_mat(2,3) = zero
s_mat(3,1) = s_mat(1,3)
s_mat(3,2) = s_mat(2,3)
s_mat(3,3) = zero
!
mass_int_sq(1) = zero
mass_int_sq(2) = zero
mass_int_sq(3) = zero
!
s_mat(1,1) = s_mat(1,1) - mass_int_sq(1) - mass_int_sq(1)
s_mat(1,2) = s_mat(1,2) - mass_int_sq(1) - mass_int_sq(2)
s_mat(1,3) = s_mat(1,3) - mass_int_sq(1) - mass_int_sq(3)
s_mat(2,1) = s_mat(2,1) - mass_int_sq(2) - mass_int_sq(1)
s_mat(2,2) = s_mat(2,2) - mass_int_sq(2) - mass_int_sq(2)
s_mat(2,3) = s_mat(2,3) - mass_int_sq(2) - mass_int_sq(3)
s_mat(3,1) = s_mat(3,1) - mass_int_sq(3) - mass_int_sq(1)
s_mat(3,2) = s_mat(3,2) - mass_int_sq(3) - mass_int_sq(2)
s_mat(3,3) = s_mat(3,3) - mass_int_sq(3) - mass_int_sq(3)
!
call preparesmatrix()
!
! TRId4(s(1,2),zero,zero,zero,zero,zero)
res6 = a30(s_null)
fun( 4, 1 ) = res6%a
fun( 4, 2 ) = res6%b + log(musq)*res6%a
fun( 4, 3 ) = res6%c + log(musq)*res6%b + (log(musq)**2)*res6%a/2d0
fun( 4, 4 ) = (0.d0,0.d0)
fun( 4, 5 ) = (0.d0,0.d0)
!
call exitgolem95()

```

Figure 4.14: Sample code from *virtual_corrections.f90* for the calculation of the three-point scalar integral $\text{TRId4}(s(1,2), 0, 0, 0, 0, 0)$ with GOLEM95.

virtual_corrections.f90

```

! loop over helicities...
!
do i = 1, nc
  ctemp      = camp_tree(l1,l2,l3,l4,i)
  cket(i)    = ctemp
  cbra(i)    = conjg(ctemp)
end do
ctemp = (0.D0,0.D0)
do i=1,nc
  do j=1,nc
    ctemp = ctemp + cbra(i)*colour_correlation_matrix(i,j)*cket(j)
  end do
end do
camp2 = camp2 + ctemp
!
! end loop...
amplitude_square = real( camp2,ki )! spin average not included

```

Figure 4.15: Key code from *virtual_corrections.f90* demonstrating the calculation of a_B .

LO functions $(\mathcal{M}_{\text{LO}})_{\sigma}^{\{\lambda\}}$ by the color correlation matrix $\langle c_{\sigma} | c_{\sigma'} \rangle$, then summing over helicity and color and dividing by the normalisation factor $n!n_1n_2$, in a similar manner to that shown in Fig. 4.15. The four values a_0 , a_1 , a_2 and a_{UV} correspond to the 4-dimensional array `amplitude_square`, as shown below with included normalisation factor (example $n!n_1n_2 = 4$) and alongside a_b :

```

amplitude_array(0) = (amplitude_square(3)
                    - amplitude_square(1)*pi**2/12.D0)/4.D0
amplitude_array(1) = amplitude_square(2)/4.D0
amplitude_array(2) = amplitude_square(1)/4.D0
amplitude_array(3) = amplitude_square_lo/4.D0
amplitude_array(4) = amplitude_square(4)/4.D0

```

The alteration to the finite value a_0 shown above:

$$a_0 \rightarrow a_0 - \frac{\pi^2}{12} a_2, \quad (4.31)$$

arises due to a difference in dimensional regularisation convention between the virtual corrections module and the dipole module. The virtual corrections module uses the

$\overline{\text{MS}}$ -shifted pole $1/\tilde{\varepsilon}$ defined in Eq. (3.31), which can alternatively be written as

$$1/\tilde{\varepsilon} = r_\Gamma(4\pi)^\varepsilon, \quad (4.32)$$

where r_Γ is defined:

$$r_\Gamma = 1 - \varepsilon\gamma_E + \varepsilon^2 \left(\frac{\gamma_E^2}{2} - \frac{\pi^2}{12} \right) + \mathcal{O}(\varepsilon^3). \quad (4.33)$$

An alternate convention for defining the divergent poles (which is used by the dipole module) replaces r_Γ with $e^{-\varepsilon\gamma_E}$:

$$e^{-\varepsilon\gamma_E} = 1 - \varepsilon\gamma_E + \varepsilon^2 \frac{\gamma_E^2}{2} + \mathcal{O}(\varepsilon^3). \quad (4.34)$$

Using differing regularisation conventions between modules is allowed, provided the correct shift is applied to the resulting amplitude-squareds! The difference between the two conventions is:

$$\begin{aligned} r_\Gamma &= e^{-\varepsilon\gamma_E} \left(1 - \frac{\pi^2}{12}\varepsilon^2 \right) + \mathcal{O}(\varepsilon^3) \\ e^{-\varepsilon\gamma_E} &= r_\Gamma \left(1 + \frac{\pi^2}{12}\varepsilon^2 \right) + \mathcal{O}(\varepsilon^3), \end{aligned} \quad (4.35)$$

therefore, the amplitude-squared output contained in `amplitude_array` is shifted from the r_Γ to the $e^{-\varepsilon\gamma_E}$ convention by including a multiplicative factor $\left(1 - \frac{\pi^2}{12}\varepsilon^2\right)$, as is performed in Eq. (4.31). This ensures that the results produced by `virtual_corrections.f90`, via the subprocess `golem(k,mu,amplitude_array)`, are comparable with the results produced by the integrated dipole module, allowing MADGOLEM to calculate the full partonic NLO-QCD differential cross section by including the LO and real emission results (as in Eq. (4.2)):

$$\begin{aligned} d\hat{\sigma}^{\text{NLO}} &= d\hat{\sigma}^B \\ &+ dPS_2(Q) \frac{1}{2s} [a_0 + d_0] \\ &+ dPS_3(Q) \frac{1}{2s} \frac{1}{n_1 n_2} [|\mathcal{M}^{\text{real}}|^2 - \sum_j \mathcal{D}_j]. \end{aligned} \quad (4.36)$$

k5

A concern of `virtual_corrections.f90` is the numerical value of the fifth ‘external’ light-like momentum k_5^μ . In order for the helicity projection algorithms discussed on Page 64 to produce physically meaningful results, the value of k_5^μ cannot be representable in

terms of the four external momenta k_i^μ . Therefore a four-momenta must be assigned to k_5^μ by *virtual_corrections.f90* that minimises the possibility of ‘similarity’ with the other momenta.

As MADGOLEM is expected to be used for collider-based predictions, it can be assumed that a common axis for the two incoming momenta k_1^μ and k_2^μ will be parallel or near-parallel with the beam z-axis:

$$k_{1|2}^\mu \sim (E_{1|2}, 0, 0, k_{1|2}) . \quad (4.37)$$

For light-like incoming momenta this becomes proportional to $(1, 0, 0, 1)$. For collider events, the least common axis for the two outgoing momenta k_3^μ and k_4^μ will be parallel or near-parallel with the x- or y-axis:

$$k_{3|4}^\mu \neq (E_{3|4}, k_{3|4}, 0, 0) \quad \& \quad k_{3|4}^\mu \neq (E_{3|4}, 0, k_{3|4}, 0) . \quad (4.38)$$

For light-like incoming momenta these become proportional to $(1, 1, 0, 0)$ and $(1, 0, 1, 0)$. Based on these two considerations, the direction of k_5^μ , \hat{n} , can be chosen to be either

$$\hat{n} \equiv (1, 1, 0, 0) \quad \text{or} \quad \hat{n} \equiv (1, 0, 1, 0) , \quad (4.39)$$

with a minimum chance of becoming parallel or ‘similar’ to either the incoming momenta or outgoing momenta.

The choice of magnitude for k_5^μ is also a concern, as both spinor products and scalar products between two momenta of considerably different magnitudes can produce numerically unstable results. To determine a suitable magnitude for k_5^μ , a NLO-QCD ‘test’ process was numerically calculated by MADGOLEM over a range of k_5^μ magnitudes, with the overall degree of numerical stability assessed by counting the number of UV & IR pole miscancellations for each choice of magnitude. The test was run as follows:

- The test process used is $u\bar{u} \rightarrow \tilde{g}\tilde{g}$ (within the MSSM).
- The k_5^μ magnitudes tested were fixed ($k_5 = \{10^0, 10^3, 10^6, 10^9\} \cdot \hat{n}$) and variable ($k_5 = \frac{1}{2}(E_1 + E_2)\hat{n}$), with $\hat{n} \equiv (1, 1, 0, 0)$ for the direction.
- The numerical calculation was integrated over 1000 phase space events, iterated 20 times by the MADGOLEM phase space / Monte Carlo generator.
- The number of pole miscancellations ($\Delta_{1|2|UV} \geq 10^{-8}$) was tracked by counting the number of warning messages generated by the pole-checking algorithm `poles_check.f`.

The results are presented in Tab 4.1, and point towards using the ‘smart k_5^μ ’ choice:

	$k_5 = 10^0 \hat{n}$	$k_5 = 10^3 \hat{n}$	$k_5 = 10^6 \hat{n}$	$k_5 = 10^9 \hat{n}$	$k_5 = \frac{1}{2}(E_1 + E_2) \hat{n}$
Iteration	1000 events/iteration				
1	(578,440)	0	0	(0,4)	0
2	(687,620))	0	0	(0,2)	0
3	(792,768)	0	0	(0,3)	0
4	(771,759)	0	0	(0,3)	0
5	(786,761)	0	0	(0,4)	0
6	(756,738)	0	0	(0,1)	0
7	(733,719)	0	0	(0,7)	0
8	(758,737)	0	0	(0,2)	0
9	(725,709)	0	0	(0,4)	0
10	(735,725)	0	0	(0,6)	(1,0)
11	(697,673)	0	0	(0,5)	0
12	(708,690)	(1,0)	(1,0)	(1,5)	0
13	(713,692)	0	0	(0,3)	0
14	(747,725)	(1,0)	(1,0)	(1,6)	(1,0)
15	(723,703)	0	0	(0,5)	0
16	(725,688)	0	0	(0,2)	0
17	(658,649)	0	0	(0,2)	0
18	(738,725)	0	0	(0,2)	0
19	(729,696)	0	0	(0,4)	0
20	(732,620)	0	0	(0,4)	0

 Table 4.1: Number of pole miscancellations (UV/IR), for the MSSM process $u\bar{u} \rightarrow \tilde{g}\tilde{g}$.

$k_5 = \frac{1}{2}(E_1 + E_2)\hat{n}$, which is weighted by the two incoming particles' energies E_1 & E_2 to ensure that the magnitude of k_5^μ is of the same order as at least two of the four external momenta. While two other choices ($k_5 = 10^3\hat{n}$ & $k_5 = 10^6\hat{n}$) produced the same number of miscancellations, there is no guarantee that this stability will hold across the majority of MADGOLEM processes and energy scales.

From these considerations, the defined value for k_5^μ within *virtual_correction.f90* is:

$k(:,5) = (k(0,1) + k(0,2)) / 2d0 * (/ 1d0, 1d0, 0d0, 0d0 /)$

Chapter 5

Calculated processes

5.1 Introduction

This chapter demonstrates the calculation of several sophisticated NLO–QCD $2 \rightarrow 2$ processes by MADGOLEM, in order to convince the reader of the code’s usability. Each of these calculations has been published by the MADGOLEM team in the papers [14–16], with a focus on the phenomenological aspects of the numerical results. The objective of this chapter is instead to present the specific results of the virtual corrections module for each process, as governed by the PERL script *run_golem.pl* and the numerical FORTRAN90 module *virtual_corrections.f90*.

By the end of the chapter it is hoped that enough evidence has been provided to demonstrate beyond reasonable doubt that the virtual corrections module, as part of MADGOLEM, is a viable loop calculator; and that satisfactorily performs as an automated platform/module for the calculation of the NLO–QCD virtual corrections to generic new physics beyond the SM $2 \rightarrow 2$ processes, based on a fully analytical, Feynman–diagrammatic approach.

The individual NLO–QCD processes presented in the following sections are:

1. $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ (MSSM) [14];
2. $pp \rightarrow \tilde{q}\tilde{q}^*$ (MSSM) [16];
3. $pp \rightarrow \tilde{g}\tilde{g}$ (MSSM) [16];
4. $pp \rightarrow GG^*$ (scalar gluons / sgluons) [15].

5.2 $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$

The NLO-QCD corrections to the $2 \rightarrow 2$ MSSM process $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ have been calculated by MADGOLEM and are presented in full detail in [14].

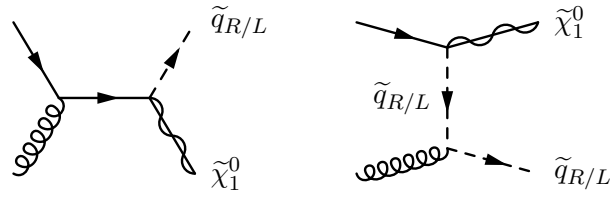
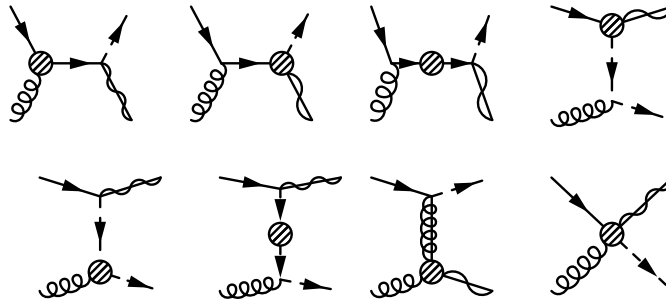
5.2.1 Motivation

In conventional MSSM scenarios where R-parity is conserved, the preferred channel of squark decay is $\tilde{q} \rightarrow q\tilde{g}$ when kinematically viable. However if the gluino \tilde{g} is of a sufficiently high mass, then $\tilde{q} \rightarrow q\tilde{\chi}_1^0$ becomes the new favoured channel [61]. For parameter sets that posit the neutralino $\tilde{\chi}_1^0$ as the lightest supersymmetric partner (LSP) candidate, this is the end-point of the decay chain.

As an aid to dark matter studies of this MSSM decay process $\tilde{q} \rightarrow q\tilde{\chi}_1^0$, one can study the complementary process at the LHC, which is the production of the LSP ($\tilde{\chi}_1^0$) in association with a squark: $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$. The lead driver of this process at leading-order (LO), ignoring QCD vertices, is the $q\tilde{q}\tilde{\chi}_1^0$ interaction. As the dominant light quarks have a negligible Yukawa coupling, this interaction is governed by the two weak charges of the quark-squark pair involved. This way the $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ process provides information on the composition and dynamics of the dark matter candidate $\tilde{\chi}_1^0$, and about the underlying SUSY-breaking mechanism [62]. An accurate measurement would improve predictions and parameter space constraints for direct detection and relic density of dark matter.

The main computational motivation for calculating this process is that it tests a very large range of computational steps in MADGOLEM and in the virtual corrections module. The list of algorithms tested by the $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ process are:

- Handling of (massive and massless) external and internal scalars, vectors, and fermions. This includes the dedicated handling of Majoranas (which are genuine indicators of new physics), to correctly resolve the assignment of the overall relative sign for each Feynman diagram, as well as possible instances of "clashing arrows" along the fermion flow.
- Tensor and loop algorithms for massive integrals including 4-point box diagrams.
- Renormalisation routines using MSSM-defined counterterm model files.
- Numerical compatibility between the individual MADGOLEM modules as shown in Fig. 4.1, particularly the matching of the MSSM integrated dipoles with the virtual corrections.

Figure 5.1: LO diagrams for $pp \rightarrow \tilde{q}_{R/L}\tilde{\chi}_1^0$.Figure 5.2: Loop diagrams for $pp \rightarrow \tilde{q}_{R/L}\tilde{\chi}_1^0$. Shaded blobs contain all strongly interacting corrections to the propagator or vertex.

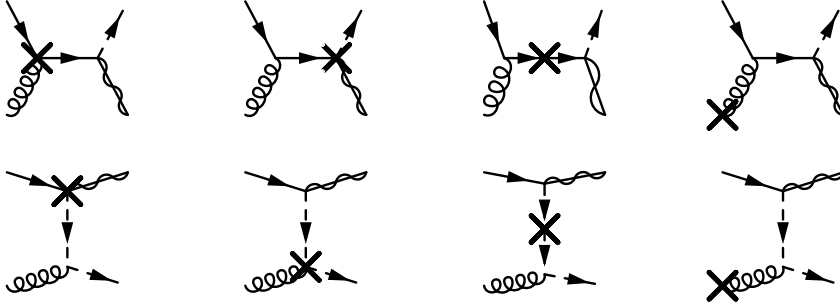
5.2.2 Calculation

Analytical computation

Following the approach laid out in Chapter 4, MADGOLEM generates the full set of LO and NLO-QCD diagrams contributing to the process $pp \rightarrow \tilde{q}_{R/L}\tilde{\chi}_1^0$, and provides the MSSM dipole and OS subtraction terms required for the complete NLO-QCD calculation.

For the virtual corrections, QGRAF produces LO, loop and counterterm diagrams as shown in Figures 5.1, 5.2 & 5.3, where shaded ‘blobs’ contain all strongly interacting corrections to the propagator or vertex. These diagrams are processed by *run_golem.pl*, which reduces the number of unique loop diagrams by loop filtering. Model-independent Feynman rules are applied to the diagrams, including defining the fixed fermion flow for the internal and external Majorana particles.

A full reduction of the Feynman diagrams into partial amplitudes and coefficients within a basis of color, helicity and loop function is performed. As the extra external momentum k_5^μ is not required for this step, the tensor reduction of loop integrals (including massive boxes) does not require much computational effort, and the resulting analytical results stored in the *AMP_*.mapout* files are reasonably concise: 2 color bases and 100 functions are identified for $ug \rightarrow \tilde{u}_L\tilde{\chi}_1^0$.


 Figure 5.3: Counterterm diagrams for $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$, including external gluon counterterms.

point	$m_{\tilde{u}_L}$	$m_{\tilde{u}_R}$	$m_{\tilde{d}_L}$	$m_{\tilde{d}_R}$	$m_{\tilde{\chi}_1^0}$	$m_{\tilde{g}}$	mass hierarchy
SPS1a ₁₀₀₀	561	549	586	545	96.7	1000	$\tilde{q}_R < \tilde{q}_L < \tilde{g}$

 Table 5.1: Squark and gluino masses in GeV for the SPS1a₁₀₀₀ benchmark point.

Values for the counterterm renormalisation constants are provided by the MSSM model file *CT_list_mod.map*, which is defined for the MSSM using the Feynman rules given in Appendix A.2.

From these analytical results, the fully renormalised virtual corrections FORTRAN90 module is created, and linked to the integrated dipoles within a $2 \rightarrow 2$ MC phase space generator run by MADGOLEM.

Numerical computation

The numerical calculation of $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ uses the parameter set SPS1a₁₀₀₀, which is a modification of the MSSM parameter set SPS1a [63], such that the gluino mass is raised to 1 TeV. This benchmark point features a minimal Supergravity–based SUSY breaking mechanism. It avoids the current LHC bounds on squark–gluino production [64, 65], and ensures that the decay channel $\tilde{q} \rightarrow q\tilde{\chi}_1^0$ is preferred over $\tilde{q} \rightarrow q\tilde{g}$, making $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ a process of interest. The relevant masses in SPS1a₁₀₀₀ are given in Table 5.1. The first and second generation squark masses are degenerate and separated by chirality, so that $m_{\tilde{u}_L} = m_{\tilde{c}_L}$, $m_{\tilde{d}_L} = m_{\tilde{s}_L}$, $m_{\tilde{u}_R} = m_{\tilde{c}_R}$, $m_{\tilde{d}_R} = m_{\tilde{s}_R}$.

The PDFs used by MADGOLEM for the numerical calculation are supplied by the five–flavor CTEQ6L1 for LO processes, and CTEQ6M for NLO processes [66], in order to consistently account for the LO and NLO distributions separately. Factorisation and renormalisation scales are set to their central values:

$$\mu^0 \equiv \mu_R^0 = \mu_F^0 = \frac{m_{\tilde{q}} + m_{\tilde{\chi}_1^0}}{2}, \quad (5.1)$$

	$\sigma^{\text{LO}}[\text{pb}]$	$\sigma^{\text{NLO}}[\text{pb}]$	K	$m_{\tilde{q}_R} [\text{GeV}]$
$\tilde{u}_R\tilde{\chi}_1^0$	29.62	42.17	1.42	549
$\tilde{d}_R\tilde{\chi}_1^0$	3.61	5.31	1.47	545
$\tilde{c}_R\tilde{\chi}_1^0$	1.12	1.81	1.61	549
$\tilde{s}_R\tilde{\chi}_1^0$	0.57	0.78	1.38	545
$\sum \tilde{q}_R\tilde{\chi}_1^0$	34.92	50.07	1.43	

Table 5.2: Individual production rates $\sigma(pp \rightarrow \tilde{q}_R\tilde{\chi}_1^0)$ and corresponding K factors for the modified SPS1a₁₀₀₀ scenario at $\sqrt{S} = 7$ TeV.

	$\sigma^{\text{LO}}[\text{pb}]$	$\sigma^{\text{NLO}}[\text{pb}]$	K	$m_{\tilde{q}_L} [\text{GeV}]$
$\tilde{u}_L\tilde{\chi}_1^0$	0.83	1.26	1.52	561
$\tilde{d}_L\tilde{\chi}_1^0$	1.21	1.77	1.46	568
$\tilde{c}_L\tilde{\chi}_1^0$	0.03	0.06	2.00	561
$\tilde{s}_L\tilde{\chi}_1^0$	0.19	0.29	1.56	568
$\sum \tilde{q}_L\tilde{\chi}_1^0$	2.26	3.38	1.50	

Table 5.3: Individual production rates $\sigma(pp \rightarrow \tilde{q}_L\tilde{\chi}_1^0)$ and corresponding K factors for the modified SPS1a₁₀₀₀ scenario at $\sqrt{S} = 7$ TeV.

a choice that has been proven to provide perturbatively stable results.

The NLO-QCD cross sections and K factors for right- and left-handed light flavor squark-neutralino production calculated at 7 TeV are provided in Tables 5.2 & 5.3, broken down by final state squark generation. Cross sections and K factors for production at 14 TeV, as well as for other SPS parameter sets are presented in the MADGOLEM paper [14].

The largest rates of $\tilde{q}\tilde{\chi}_1^0$ production predictably come from the $\tilde{u}\tilde{\chi}_1^0/\tilde{d}\tilde{\chi}_1^0$ channels, due to the flavor-locked nature of the process with the proton valence quarks. Differences in the chiral production rate between \tilde{q}_R and \tilde{q}_L can be explained by the mostly bino-like neutralino, which couples with a different magnitude to the right- and left-handed squarks. For the parameter set SPS1a₁₀₀₀, the ratio of chiral coupling strengths is

$$\frac{g_{\tilde{u}_L\tilde{\chi}_1^0 u}}{g_{\tilde{u}_R\tilde{\chi}_1^0 u}} \simeq \frac{1}{6}, \quad (5.2)$$

which accounts well for the ratio of cross sections

$$\frac{\sigma(pp \rightarrow \tilde{u}_L\tilde{\chi}_1^0)}{\sigma(pp \rightarrow \tilde{u}_R\tilde{\chi}_1^0)} \sim \frac{1}{36}. \quad (5.3)$$

The minor deviation from this ratio can be assigned to the small mass difference between the right- and left-handed squarks.

Band plots for the dominant production channel $pp \rightarrow \tilde{u}_R\tilde{\chi}_1^0$ (as read from Tables 5.2

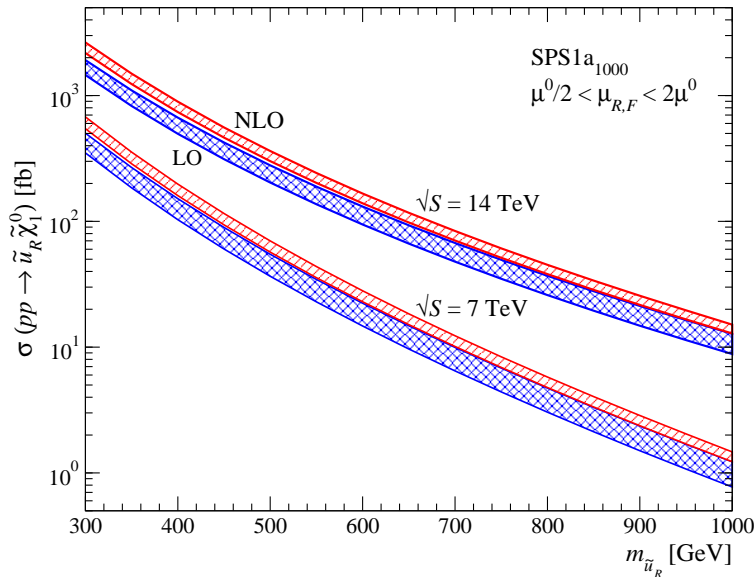


Figure 5.4: Band plots for $pp \rightarrow \tilde{u}_R \tilde{\chi}_1^0$ (blue = LO, red = NLO).

& 5.3) are presented in Figure 5.4, at $\sqrt{S} = 7 \text{ TeV}$ and $\sqrt{S} = 14 \text{ TeV}$. The squark masses are varied simultaneously by fixing the mass difference parameter:

$$\Delta_m \equiv m_{\tilde{u}_L} - m_{\tilde{u}_R} = 20 \text{ GeV} . \quad (5.4)$$

The band plots are obtained over the range $\frac{1}{2}\mu^0 < \mu_{R,F} < 2\mu^0$. The scale dependence of the NLO–QCD process $pp \rightarrow \tilde{u}_R \tilde{\chi}_1^0$ is seen to be drastically improved over the LO.

The topological selection algorithms present in MADGOLEM (see Page 53) allow for the presentation of NLO contributions to be separated into different topologies for the loop corrections, and placed alongside the integrated dipole and real emission terms. This is shown in Figure 5.5 for $pp \rightarrow \tilde{u}_{R/L} \tilde{\chi}_1^0$ at $\sqrt{S} = 7 \text{ TeV}$ ¹, as a plot of individual contributions by relative size:

$$\frac{\Delta\sigma^{\text{NLO}}}{\sigma^{\text{LO}}} \equiv \frac{\sigma^{\text{NLO}} - \sigma^{\text{LO}}}{\sigma^{\text{LO}}} . \quad (5.5)$$

The squark masses are varied, fixing the mass difference parameter Δ_m (Eq. (5.4)) as before.

From this topological breakdown it can be seen that the virtual contributions are dominated by corrections to the SM–QCD vertex $g\bar{u}u$, with the only other sizable

¹The real contributions and dipoles are separated by setting the FKS–like [67,68] dipole parameter $\alpha = 1$, which allocates more phase space to the integrated and non–integrated dipole contributions, and less to the $2 \rightarrow 3$ real–emission contribution.

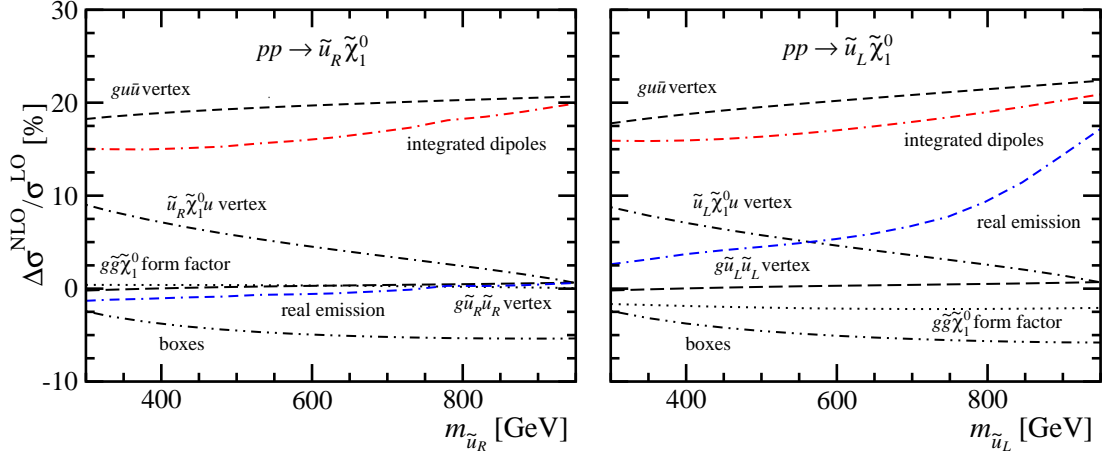


Figure 5.5: Individual contributions to $pp \rightarrow \tilde{u}_{R/L}\tilde{\chi}_1^0$ by relative size: $\Delta\sigma^{\text{NLO}}/\sigma^{\text{LO}}$. Contributions from self energy diagrams are negligible ($< 1\%$) and not shown.

contributions ($\sim 10\%$) coming from the MSSM electroweak vertex $\tilde{u}\tilde{\chi}_1^0 u$ at low squark masses. The integrated dipoles (red line) contribute a similar amount as the $\tilde{g}\tilde{u}u$ vertex corrections over the whole range of squark masses; while the real emission (blue line) is only significant for left-handed squark production ($pp \rightarrow \tilde{u}_L\tilde{\chi}_1^0$) at large squark masses².

The numerical results for the finite renormalised virtual amplitudes produced by *virtual_corrections.f90* have been compared with FEYNARTS, FORMCALC and LOOPTOOLS [59]. Internal checks, such as varying the choice of reference momenta for the external particles within the spinor helicity formalism have also been applied; for example, enforcing all external particles to use k_5^μ as their reference momentum produces identical numerical results. Similar procedures have been used to check the overall gauge invariance of the calculation. Finally, the final results have been compared with the literature (e.g. $e^+e^- \rightarrow \tilde{q}\tilde{q}^*$ [69]) as well as with PROSPINO [70] and MADFKS [71] (e.g. $pp \rightarrow \tilde{l}^*$).

5.3 $pp \rightarrow \tilde{q}\tilde{q}^*$

The NLO-QCD corrections to the $2 \rightarrow 2$ MSSM process $pp \rightarrow \tilde{q}\tilde{q}^*$ have been calculated by MADGOLEM and are presented in full detail in [16].

²This is due to the additional real-emission coupling terms $g_{\tilde{u}_R\tilde{\chi}_1^0 u}$ at NLO over the coupling $g_{\tilde{u}_L\tilde{\chi}_1^0 u}$ at LO (with the ratio shown in Eq. (5.2)), which causes large K_{real} factors as the LO cross section decreases for larger masses.

5.3.1 Motivation

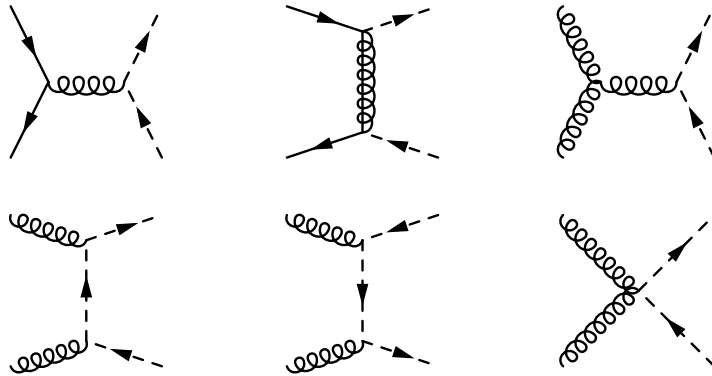
Experimental searches [72,73] underway at the LHC are probing vast parameter regions of the MSSM, most notably the sections of the squark–gluino mass plane which can be described in terms of gravity mediation [74]. Inclusive searches for the production and decay of squarks and gluinos require high–accuracy predictions including NLO–QCD effects, which up until now have been calculated with simplifying assumptions about the squark mass spectrum involved. In contrast, MADGOLEM can freely sweep over the entire parameter space of a given model, varying each input parameter independently in order to calculate effects to both total rates and distributions. These abilities allow one to observe *e.g.* the shift in channel rates as the R/L balance of squarks is altered.

This differs from PROSPINO or other precision tools which rely on a single mass scale for all light–flavor squarks for all NLO–QCD effects. A fully general scan as provided by MADGOLEM is therefore beyond the reach of these tools.

Within the MADGOLEM paper [16] are calculations for squark pair production ($pp \rightarrow \tilde{q}\tilde{q}/\tilde{q}\tilde{q}^*$), gluino production in association with a squark ($pp \rightarrow \tilde{g}\tilde{q}/\tilde{g}\tilde{q}^*$), and gluino pair production ($pp \rightarrow \tilde{g}\tilde{g}$, detailed in Section 5.4). The computational motivation for selecting the process $pp \rightarrow \tilde{q}\tilde{q}^*$ for demonstration is that it tests several areas of the MADGOLEM code that are not covered by Section 5.2:

- Handling of multiple distinct initial state channels: gg fusion; $q\bar{q}$ annihilation and scattering (via s–channel gluon and t–channel gluino sub–channels respectively).
- Color flow decomposition routines for complicated color structures resulting from the interaction of four colored particles, two of which may be color octets.
- Inclusion of ghost particles.
- Correct application of the four–scalar vertex Feynman rule, and inclusion of complicated gluino counterterms (including the correction to the Yukawa coupling $\hat{g}_s^{(0)}$).
- Calculation of the MSSM–specific on–shell (OS) subtraction terms.
- Handling of a large diversity of loop topologies, including the reduction of rank 3 & 2 box tensors.
- Clear benefits from loop filtering: *e.g.* squark–mediated contributions in which all families and chiralities are involved.

For simplicity only the calculations for the up–squark final states ($pp \rightarrow \tilde{u}_{R/L}\tilde{u}_{R/L}^*$) will be presented in this section.

Figure 5.6: LO diagrams for $pp \rightarrow \tilde{u}_L \tilde{u}_L^*$.

5.3.2 Calculation

Analytical computation

As the process $pp \rightarrow \tilde{u}_{R/L} \tilde{u}_{R/L}^*$ is $\mathcal{O}(\alpha_s^2)$ at LO, MADGOLEM generates NLO-QCD diagrams of $\mathcal{O}(\alpha_s^3)$, including ghost particles in the corrections to the gluon propagator and three-gluon vertex, in order to compensate for the unphysical degrees of freedom of the exchanged virtual gluons upon Faddeev-Popov quantisation. The process is separated and stored in multiple channels according to the initial states: gg and $q\bar{q} = \{u\bar{u}, d\bar{d}, s\bar{s}, c\bar{c}\}$. MSSM dipoles are generated, as well as the MSSM OS subtraction terms.

For the virtual corrections, QGRAF produces LO, loop and counterterm diagrams as shown in Figures 5.6, 5.7 & 5.8. The processes $pp \rightarrow \tilde{u}_R \tilde{u}_R^*$ and $pp \rightarrow \tilde{u}_L \tilde{u}_L^*$ are topologically identical, and allowed by all partonic sub-channels (gg fusion, $q\bar{q}$ annihilation, and $q\bar{q}$ scattering by a t-channel gluino). The processes $pp \rightarrow \tilde{u}_L \tilde{u}_R^*$ and $pp \rightarrow \tilde{u}_R \tilde{u}_L^*$ are also topologically identical, and contain only two partonic sub-channels ($q\bar{q}$ annihilation and scattering), as gluonic initial states for these processes are forbidden at LO due to flavor locking.

Iterating over the initial state channels $\{gg, u\bar{u}, \dots\}$, `run_golem.pl` reduces the number of unique loop diagrams by filtering techniques (as discussed on Page 53), before applying model-independent Feynman rules. The colored Feynman rules for the squark quartic vertices have the definitions as defined in Appendix A.2, dependent on flavor and chirality. These three varieties are tagged by their coupling in the model files (`G4SQ1`, `G4SQ2`, `G4SQ3`), to ensure that the correct color structure is applied to each.

A full reduction of the Feynman diagrams into partial amplitudes and coefficients is performed without the presence of k_5^μ : for the example channel $gg \rightarrow \tilde{u}_L \tilde{u}_L^*$ the increase in complexity of the color structure produces 6 identified color bases by the color flow

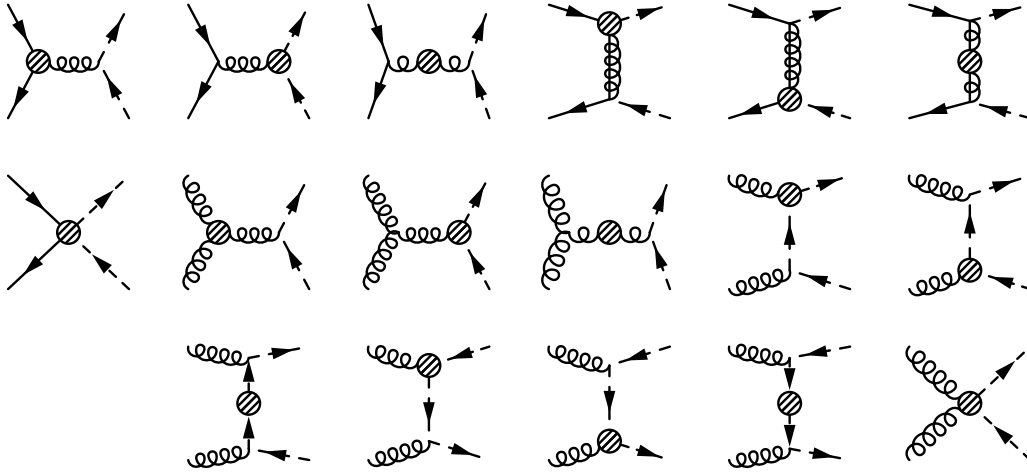


Figure 5.7: Loop diagrams for $pp \rightarrow \tilde{u}_L \tilde{u}_L^*$.

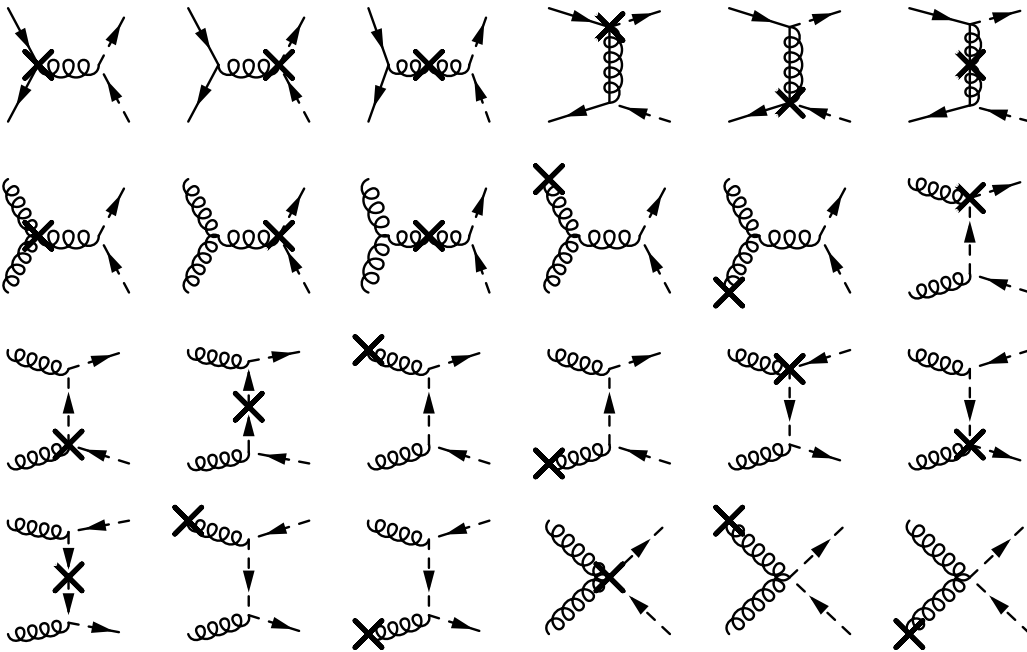


Figure 5.8: Counterterm diagrams for $pp \rightarrow \tilde{u}_L \tilde{u}_L^*$, including external gluon counterterms.

point	$m_{\tilde{u}_L}$	$m_{\tilde{u}_R}$	$m_{\tilde{d}_L}$	$m_{\tilde{d}_R}$	$m_{\tilde{g}}$	mass hierarchy
CMSSM 10.2.2	1162	1120	1165	1116	1255	$\tilde{q}_R < \tilde{q}_L < \tilde{g}$

Table 5.4: Squark and gluino masses in GeV for the CMSSM 10.2.2 benchmark point for the LHC at $\sqrt{S} = 14$ TeV.

decomposition algorithm, and 101 basis functions by the tensor reduction algorithms. For the simpler color-structured example channel $u\bar{u} \rightarrow \tilde{u}_L\tilde{u}_L^*$, 2 color bases and 125 functions are identified.

Values for the counterterm renormalisation constants are provided by the MSSM model file as before, including the definitions for the gluino renormalisation constants and the modified Yukawa coupling $\hat{g}_s^{(0)}$, which includes the conventional finite piece in order to restore the supersymmetric Ward identities as prescribed in Subsection 3.4.1.

From these analytical results, the fully renormalised virtual corrections FORTRAN90 module is created, and linked to the integrated dipoles within a $2 \rightarrow 2$ MC phase space generator run by MADGOLEM.

Numerical computation

The numerical calculation of $pp \rightarrow \tilde{u}_{R/L}\tilde{u}_{R/L}^*$ uses the MSSM parameter benchmark point CMSSM 10.2.2 [75], with relevant masses given in Table 5.4. Unlike the $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ case, the first and second generation squark masses are completely independent.

The PDFs used by MADGOLEM for the numerical calculation are supplied by CTEQ6L1 and CTEQ6M, as in the $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ case. Factorisation and renormalisation scales are set to their central values:

$$\mu^0 \equiv \mu_R^0 = \mu_F^0 = \frac{m_1 + m_2}{2}, \quad (5.6)$$

where $m_{1,2}$ are the final state squark masses.

The NLO-QCD cross sections and K factors for right- and left- handed up-squark-antisquark production calculated at the LHC center of mass energy $\sqrt{S} = 14$ TeV are provided in Table 5.5, broken down into the separate chiral final states:

$$pp \rightarrow \tilde{u}_L\tilde{u}_L^*, \quad pp \rightarrow \tilde{u}_R\tilde{u}_R^*, \quad pp \rightarrow \tilde{u}_L\tilde{u}_R^*, \quad pp \rightarrow \tilde{u}_R\tilde{u}_L^*. \quad (5.7)$$

A comprehensive list of cross sections and K factors including both first generation final state squarks, over a range of benchmark points at both 8 TeV and 14 TeV, is presented in the MADGOLEM paper [16].

It can be seen that the K factors for all chiral final states are fairly similar ($K \sim 1.5$); this is due to the dominance of the SM QCD corrections, mainly those originating from

	$\sigma^{\text{LO}}[\text{fb}]$	$\sigma^{\text{NLO}}[\text{fb}]$	K
$\tilde{u}\tilde{u}_L^*$	3.0	4.5	1.52
$\tilde{u}_R\tilde{u}_R^*$	3.8	5.7	1.50
$\tilde{u}_L\tilde{u}_R^*, \tilde{u}_R\tilde{u}_L^*$	4.6	6.8	1.47

Table 5.5: Individual production rates $\sigma(pp \rightarrow \tilde{u}_{R/L}\tilde{u}_{R/L}^*)$ and corresponding K factors for the CMSSM 10.2.2 benchmark point, at $\sqrt{S} = 14$ TeV.

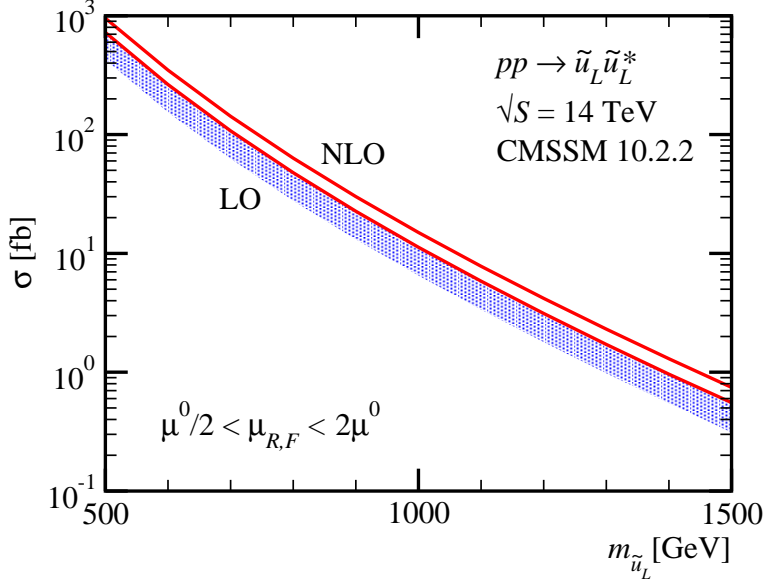


Figure 5.9: Band plots for $pp \rightarrow \tilde{u}_L\tilde{u}_L^*$ (blue = LO, red = NLO).

the exchange of virtual (and the emission of real) gluons, which are independent of chirality. Supersymmetric QCD effects are suppressed by the heavy particle masses involved in squark and gluino loops.

Band plots for the production channel $pp \rightarrow \tilde{u}_L\tilde{u}_L^*$ are presented in Figure 5.9 for $\sqrt{S} = 14$ TeV, as a function of the final state mass $m_{\tilde{u}_L}$. All the other heavy masses are varied simultaneously, keeping the absolute mass splittings of the CMSSM 10.2.2 benchmark point shown in Table 5.5. The band plots are obtained over the range $\frac{1}{2}\mu^0 \leq \mu_{R,F} \leq 2\mu^0$. Both error bands nicely overlap and reflect a reduction of the theoretical uncertainties from $\mathcal{O}(60\%)$ at LO down to $\mathcal{O}(30\%)$ at NLO.

A breakdown of the virtual corrections to $pp \rightarrow \tilde{u}_L\tilde{u}_L^*$ by topology is displayed alongside the integrated dipoles in Figure 5.10, separated into the two initial state channels gg and $q\bar{q}$. The individual contributions are presented in terms of relative size $\Delta\sigma^{\text{NLO}}/\sigma^{\text{LO}}$ as defined in Eq. (5.5). Heavy masses are varied in parallel, as in Figure 5.9.

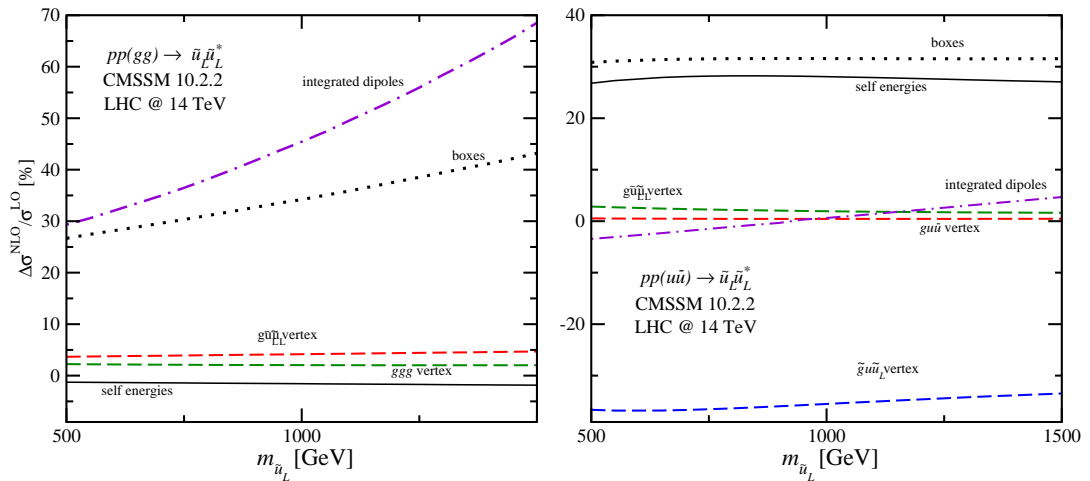


Figure 5.10: Individual virtual contributions to $pp \rightarrow \tilde{u}_L \tilde{u}_L^*$ and integrated dipoles by relative size: $\Delta\sigma^{\text{NLO}}/\sigma^{\text{LO}}$.

The topological breakdown of the gluon fusion channel ($gg \rightarrow \tilde{u}_L \tilde{u}_L^*$) shows an interesting feature of squark–antisquark production: threshold effects corresponding to Sommerfeld enhancement [76]. To explain, long–range gluon exchange between slowly moving squarks in the $gg \rightarrow \tilde{u}_L \tilde{u}_L^*$ channel gives rise to a Coulomb singularity $\sigma \sim \pi\alpha_s/\beta$, where β denotes the relative squark velocity in the center–of–mass frame, $\beta = \sqrt{1 - 4m_{\tilde{u}_L}^2/\hat{S}}$. This feature is clearly visible in the domination of the relative box contributions in Figure 5.10, which increase with the squark mass $m_{\tilde{u}_L}$ (leading to sizeable quantum effects in the 30%–70% range). In comparison, the individual relative contributions in the $q\bar{q} \rightarrow \tilde{u}_L \tilde{u}_L^*$ channel stay fairly constant. This analysis is illustrative of the usefulness of the topology selection algorithms within MADGOLEM, allowing physical insights into NLO–QCD processes to be made from the underlying topological information.

As for $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$, the numerical results for the finite renormalised virtual amplitudes produced by *virtual_corrections.f90* have been compared with FEYNARTS, FORMCALC and LOOPTOOLS [59]. Internal checks, such as varying the choice of reference momenta for the external particles within the spinor helicity formalism have also been applied; for example, enforcing all external particles to use k_5^μ as their reference momentum produces identical numerical results. Similar procedures have been used to check the overall gauge invariance of the calculation. Additionally, the results within the MADGOLEM paper for squark–antisquark production [16] have been checked with PROSPINO [70] wherever possible.

5.4 $pp \rightarrow \widetilde{g}\widetilde{g}$

The NLO–QCD corrections to the $2 \rightarrow 2$ MSSM process $pp \rightarrow \widetilde{g}\widetilde{g}$ have been calculated by MADGOLEM and are presented in full detail in [16].

5.4.1 Motivation

The phenomenological reasoning for providing automated NLO–QCD corrections to gluino pair production is much the same as for squark–antisquark production in Section 5.3. Loop effects in gluino pair processes are exceedingly involved, forcing NLO tools to assume mass degeneracies for the internal squarks. Just as for the calculation of $pp \rightarrow \widetilde{q}\widetilde{q}^*$, MADGOLEM makes no assumptions about model parameters due to its model–independent algorithms. This allows for a complete parameter sweep over the calculated NLO–QCD process.

The computational motivation for selecting the process $pp \rightarrow \widetilde{g}\widetilde{g}$ for demonstration is that this process is a great example of an *extreme* case in $2 \rightarrow 2$ NLO–QCD calculations:

- The loop corrections have a maximally complicated color structure, containing up to four external color octets (for the initial state gg fusion channel) as well as the strongly interacting $gg\widetilde{q}\widetilde{q}^*$ & squark quartic vertices.
- The initial state $q\bar{q}$ annihilation & scattering channel forces the use of the external k_5^μ reference momentum, as the channel involves four external fermions. This results in a very large number of partially–reduced massive form factors to be calculated by the GOLEM95 integral library, including 4–point tensor integrals.
- The helicity basis is large due to the massive Majorana fermions in the final state, combined with vectors and fermions in the initial state. The hermitian chiral symmetry simplification flag `nlosymsimpl` used to reduce the number of unique helicities (Eq. (4.14)) cannot be enabled for this process either, due to the inclusion of chirally–dependent squark masses in the loops.
- The number of contributing diagrams at the loop level is incredibly large: numbering 146 for each $q\bar{q}$ initial state channel, and 402 for the gg fusion channel.

These four factors contribute to create an incredibly large analytical and numerical output, of a complexity that cannot easily be beaten³.

³One can increase the size of the helicity basis by producing massive vector particles instead of massive fermions, but this process would not require the usage of the additional k_5^μ momentum and would therefore lead to a much simpler output. Phenomenologically, this situation would correspond to the production of vector color octets, e.g. coloron or axigluon fields [77, 78].

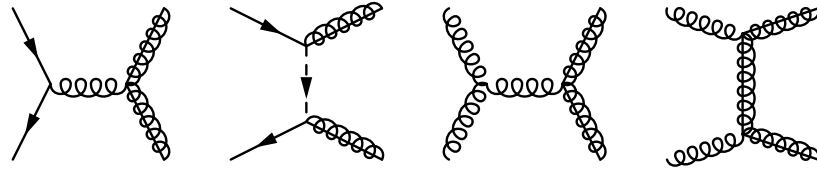


Figure 5.11: LO diagrams for $pp \rightarrow \tilde{g}\tilde{g}$. Not pictured are the u-channel diagrams, which correspond to swapping the external gluinos in the t-channel.

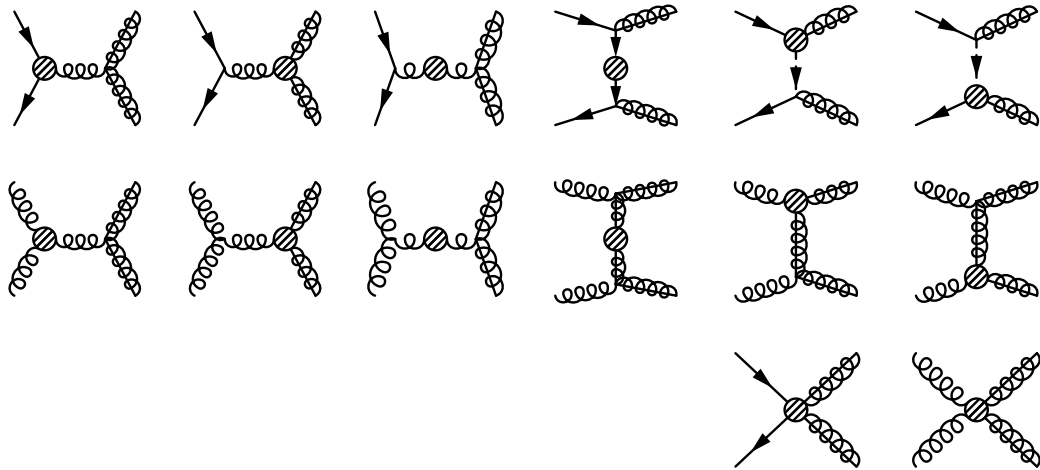


Figure 5.12: Loop diagrams for $pp \rightarrow \tilde{g}\tilde{g}$. Not pictured are the u-channel diagrams, which correspond to swapping the external gluinos in the t-channel.

5.4.2 Calculation

Analytical computation

As with the process $pp \rightarrow \tilde{q}\tilde{q}^*$, MADGOLEM generates NLO-QCD diagrams of $\mathcal{O}(\alpha_s^3)$, including ghost particles in the production of loop diagrams. The process is separated into the individual initial state channels: gg and $q\bar{q} = \{u\bar{u}, d\bar{d}, s\bar{s}, c\bar{c}\}$. MSSM dipoles and OS subtraction terms are generated, including terms for the external gluinos.

For the virtual corrections QGRAF produces LO, loop and counterterm diagrams as shown in Figures 5.11, 5.12 & 5.13. U-channel diagrams are not presented in the Figures, although they can be constructed by swapping the two external gluinos in the t-channel diagrams.

As for the process $pp \rightarrow \tilde{q}\tilde{q}^*$, *run-golem.pl* iterates over the initial state channels $\{gg, u\bar{u}, \dots\}$, reducing the number of unique loop diagrams by filtering techniques, and applying the model-independent Feynman rules.

A full reduction of the Feynman amplitudes into partial amplitudes and coefficients

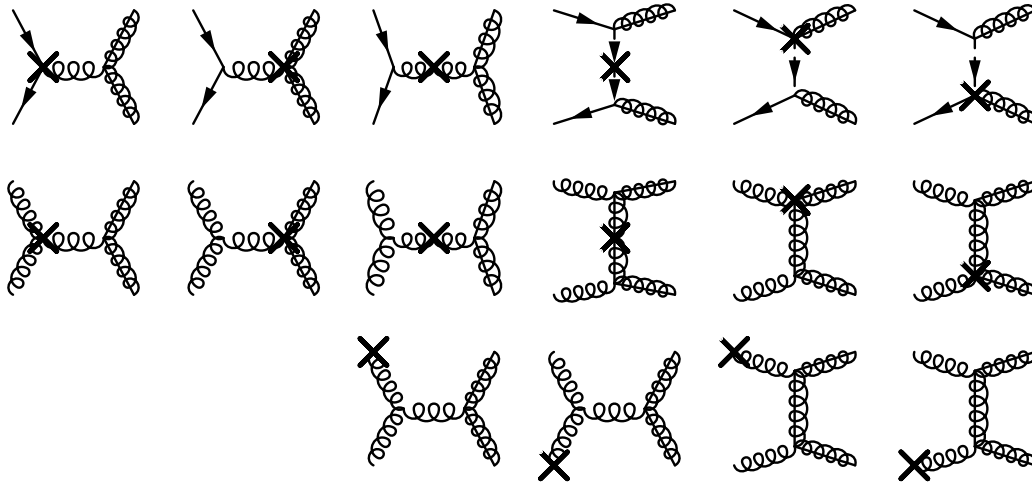


Figure 5.13: Counterterm diagrams for $pp \rightarrow \widetilde{g}\widetilde{g}$, including external gluon counterterms. Not pictured are the u-channel diagrams, which correspond to swapping the external gluinos in the t-channel.

is performed. For the channel $gg \rightarrow \widetilde{g}\widetilde{g}$, the maximal complexity of the color structure resulting from four external color octets produces 24 identified color bases. In this channel k_5^μ is not required, and all tensor integrals (including massive boxes) are reduced purely to a set of 329 scalar integrals.

For the channel $q\bar{q} \rightarrow \widetilde{g}\widetilde{g}$, the simpler color structure (2 triplets and 2 octets) produces 6 identified color bases (equivalent to the color structure for the process channel $gg \rightarrow \widetilde{q}\widetilde{q}$ calculated in Section 5.3). As this channel contains four external fermions, the external k_5^μ reference momentum is required: as a result a complete reduction of loop diagrams to scalar integrals is not possible, producing 406 scalar integrals and form factors. Unlike the previous calculations in Sections 5.2 & 5.3, the necessary introduction of k_5^μ also causes the analytical results to be written in terms of the extended Mandelstam variables ($s_{ij}; i, j \in \{1, \dots, 5\}$), instead of the conventional Mandelstam variables ($s \equiv s_{12}, t \equiv s_{13}, u \equiv s_{23}$)⁴. This increase in variables greatly enlarges the complexity of the analytical results for the channel $q\bar{q} \rightarrow \widetilde{g}\widetilde{g}$.

Values for the counterterm renormalisation constants are provided by the MSSM model file as for the process $pp \rightarrow \widetilde{q}\widetilde{q}^*$.

From these analytical results, the fully renormalised virtual corrections FORTRAN90 module is created, and linked to the integrated dipoles within a $2 \rightarrow 2$ MC phase space generator run by MADGOLEM.

⁴See Page 64.

\sqrt{S}	$\sigma^{\text{LO}}[\text{fb}]$	$\sigma^{\text{NLO}}[\text{fb}]$	K
8 TeV	0.34	1.19	3.51
14 TeV	23.3	53.4	2.29

Table 5.6: Individual production rates $\sigma(pp \rightarrow \tilde{g}\tilde{g})$ and corresponding K factors for the CMSSM 10.2.2 benchmark point, at $\sqrt{S} = 8$ TeV and $\sqrt{S} = 14$ TeV.

Numerical computation

Due to the size of the numerical code created for the NLO–QCD process $pp \rightarrow \tilde{g}\tilde{g}$, dynamic linking of the pre-compiled FORTRAN90 coefficient libraries `libcoeffs_*.so` is required, in order to avoid compilation errors during numerical calculation (as described on Page 77). MADGOLEM performs this by default.

The numerical calculation of $pp \rightarrow \tilde{g}\tilde{g}$ uses the MSSM parameter benchmark point CMSSM 10.2.2, as defined in Table 5.4. The PDFs used by MADGOLEM are supplied by CTEQ6L1 and CTEQ6M, as in the $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ and $pp \rightarrow \tilde{q}\tilde{q}^*$ cases. Factorisation and renormalisation scales are set to their central values:

$$\mu^0 \equiv \mu_R^0 = \mu_F^0 = m_{\tilde{g}}. \quad (5.8)$$

The NLO–QCD cross sections and K factors for gluino pair production calculated at the LHC center of mass energies $\sqrt{S} = 8$ TeV & $\sqrt{S} = 14$ TeV are provided in Table 5.6. A complete list of cross sections and K factors over a range of benchmark points at both 8 TeV and 14 TeV is provided in the MADGOLEM paper [16]. These results essentially reproduce what is included in PROSPINO.

K factors for the NLO–QCD corrections are very large, being $K \sim 2.3$ for the 14 TeV center of mass energy, and surpassing $K \sim 3$ for the lower LHC energy 8 TeV. The extreme size of the lower energy K factor can be determined to be caused by the poor perturbative behaviour of the CTEQ parton densities, which suppresses LO production rates for particles with $\mathcal{O}(\text{TeV})$ masses while producing perturbatively stable NLO rates. Another reason for the large K factor can be assigned to the poor convergence of the perturbative expansion in α_s , which is evolved from a mass scale of m_Z ($\alpha_s(m_Z) \sim 0.1$) up to the gluino production threshold $m_{\tilde{g}}$. The large total rate at 14 TeV when compared to $pp \rightarrow \tilde{q}\tilde{q}^*$ can be explained in part by the color charges of the produced gluinos, as interactions among color octets will give typically larger rates than color triplets. Additionally, production rates grow with the spin representation of the particles involved; gluino (spin 1/2) pair production naturally operates above squark or sgluon (spin 0) pairs.

Band plots for total gluino pair production are presented in Figure 5.14 for the

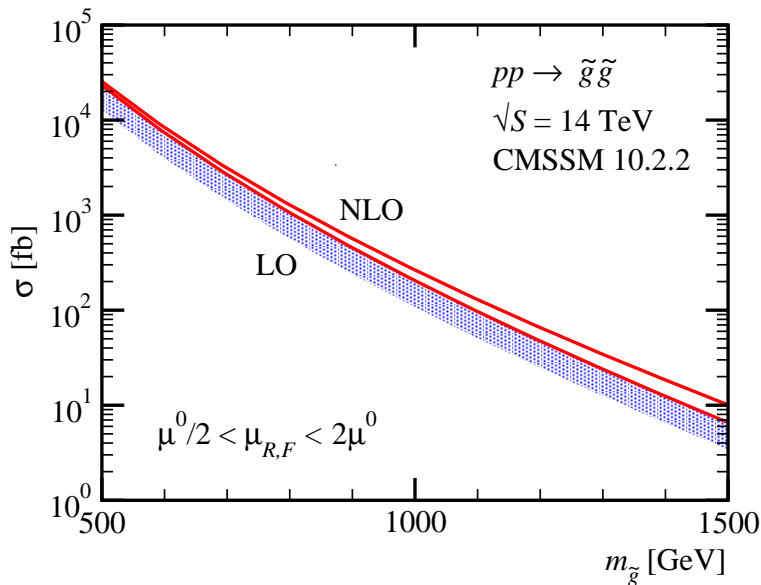


Figure 5.14: Band plots for gluino pair production (blue = LO, red = NLO).

LHC center of mass energy $\sqrt{S} = 14$ TeV, as a function of the gluino mass $m_{\tilde{g}}$. The squark masses are varied in parallel with the gluino, as for $pp \rightarrow \tilde{q}\tilde{q}^*$. The band plots are obtained over the range $\frac{1}{2}\mu^0 \leq \mu_{R,F} \leq 2\mu^0$. The error bands overlap and show a reduction in the theoretical uncertainty from $\mathcal{O}(70\%)$ at LO to $\mathcal{O}(30\%)$ at NLO, in spite of the large K factors present in Table 5.6.

The successful calculation of the process $pp \rightarrow \tilde{g}\tilde{g}$ provides good evidence for the abilities of MADGOLEM as an automated $2 \rightarrow 2$ NLO-QCD computational tool. As explained in Subsection 5.4.1, this process serves as a perfect example of a ‘difficult’ process, with an extremely large number of contributing diagrams ($\mathcal{O}(1000)$), massive basis integrals (~ 400 each for the gg - and $q\bar{q}$ -initiated channels), color structures and helicity projections. It thus qualifies as a most demanding probe of the very frontier capabilities of the analytical, Feynman-diagrammatic approach to NLO-QCD calculations and its automated implementation within the MADGOLEM framework.

5.5 $pp \rightarrow GG^*$

The NLO-QCD corrections to the $2 \rightarrow 2$ scalar gluon (sgluon) process $pp \rightarrow GG^*$ have been calculated by MADGOLEM and are presented in full detail in [15]

5.5.1 Motivation

Scalar gluons (sgluons) G , are complex scalar colored octets with zero electroweak charge. Appearing in common supersymmetric extensions to the Standard Model [79, 80], sgluons can be decoupled from other supersymmetric terms whilst still rendering a fully renormalisable sector for the considerations of sgluon pair production [15].

Direct couplings to matter only arise as effective dimension-5 operators (for instance, induced by one-loop squark and gluino loops). As long as these couplings are small – which is true if they are loop-induced – the sgluon mass range is not constrained by stringent bounds from dijet resonance searches⁵. As a consequence, sgluons can be relatively light, and at the LHC sgluon pairs will be copiously produced just through their couplings to gluons.

The computational motivation for demonstrating the process $pp \rightarrow GG^*$ is that it tests and showcases the ability of MADGOLEM to calculate processes within non-MSSM models, being a model-independent automated tool. The areas of the MADGOLEM code that are tested that have not been covered by the previous MSSM processes are:

- Creation and usage of the files required for a new NLO-QCD model within MADGOLEM, beyond the default SM and MSSM files.
- Avoiding technical issues in MADGRAPH4 [19] concerning the quartic $ggGG^*$ coupling⁶.

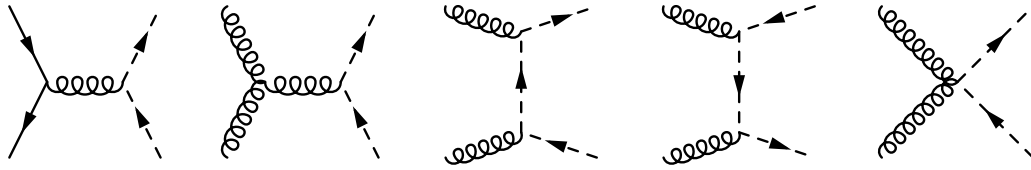
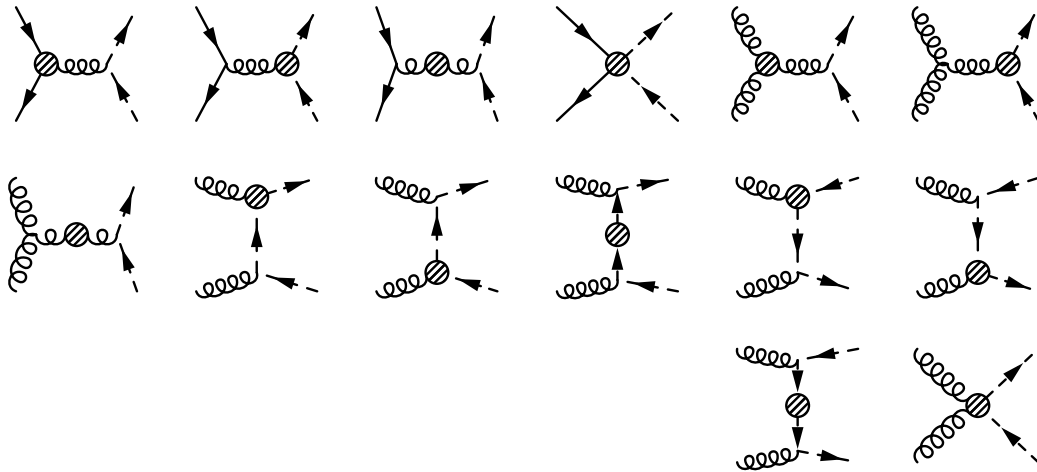
5.5.2 Calculation

Analytical computation

MADGOLEM generates NLO-QCD diagrams of $\mathcal{O}(\alpha_s^3)$ from the sgluon model files defined in `./Models/sgluon/` (`interactions-qgraf.dat`, `particles-qgraf.dat`, `vertex_ct.dat`, `selfenergy_ct.dat`). These files are modified versions of the model files used by MADGRAPH, as described in Section 4.2.1. In MADGRAPH4 a technical issue arises when generating the color structure for the quartic $ggGG^*$ coupling. This does not affect the diagrams produced by QGRAF, but does affect the LO and real emission diagrams which depend on the MADGRAPH routines, based on HELAS and the internal MADGRAPH color algorithms. MADGOLEM avoids this issue by generating the required color structure through an auxiliary massive, color-adjoint vector boson V^μ , that produces the quartic $ggGG^*$ coupling in the decoupling limit $m_V^2 \gg s$.

⁵Conversely, for $\mathcal{O}(1)$ qqG couplings sgluon masses below $m_G = \mathcal{O}(2\text{TeV})$ are already ruled out by LHC experiments.

⁶Solved in MADGRAPH5 [81].


 Figure 5.15: LO diagrams for $pp \rightarrow GG^*$.

 Figure 5.16: Loop diagrams for $pp \rightarrow GG^*$.

MADGOLEM also provides the sgluon-defined dipoles required for the complete NLO-QCD calculation, and separates the process into the initial state channels gg and $q\bar{q}$. The sgluon dipoles can be easily constructed from the MSSM squark dipoles, rescaled by a color factor that accounts for its $SU(3)_c$ -adjoint nature. This reflects the universality of the NLO-QCD virtual and real corrections, which essentially depend on the color charges and spin representation of the strongly-interacting fields present within a given model. That makes our automated, model-independent approach very easily extendable to generic BSM scenarios.

For the virtual corrections, QGRAF produces LO, loop and counterterm diagrams as shown in Figures 5.15, 5.16 & 5.17. These diagrams are processed by `run_golem.pl`, which iterates over the initial state channels and reduces the number of unique loop diagrams by filtering. Model-independent Feynman rules are applied to the diagrams.

A full reduction of the Feynman amplitudes into partial amplitudes and coefficients is performed. For this process k_5^μ is not necessary, and the `nlosymsimp` option can be enabled, as the NLO-QCD process is chirally hermitian. For the gluon fusion channel $gg \rightarrow GG^*$, 24 color bases are produced (identical to the process channel $gg \rightarrow \tilde{g}\tilde{g}$

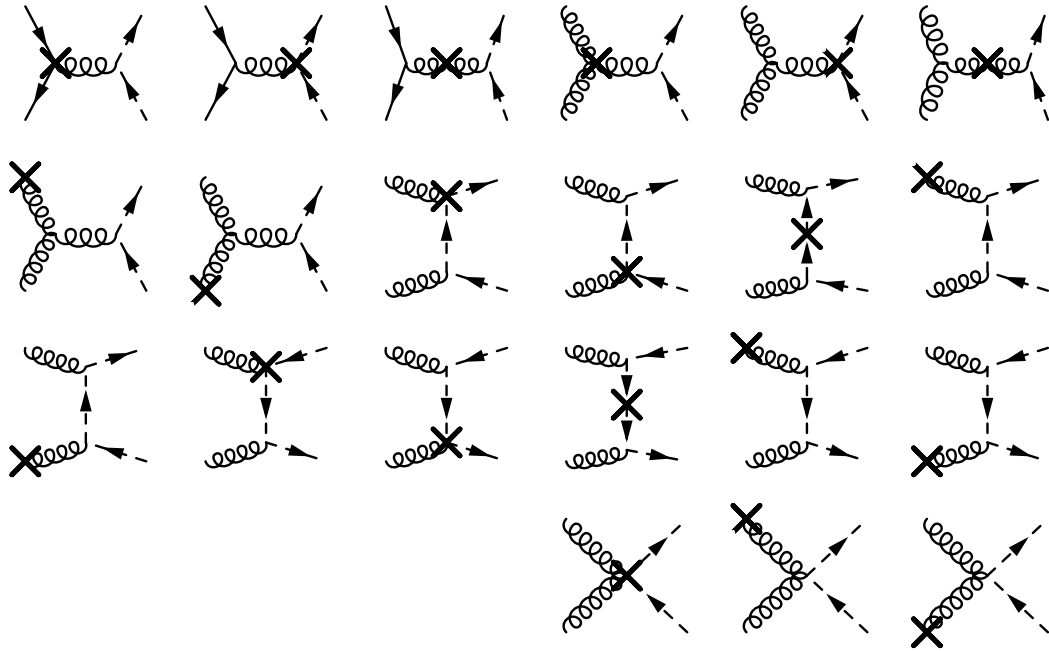


Figure 5.17: Counterterm diagrams for $pp \rightarrow GG^*$, including external gluon counterterms.

in Section 5.4), and 37 scalar basis functions are identified. For the quark antiquark annihilation channel $q\bar{q} \rightarrow GG^*$, 6 color bases are produced (identical to the process channel $q\bar{q} \rightarrow \tilde{g}\tilde{g}$), and 17 scalar basis functions are identified. The number of unique helicities is reduced to 2 (from 4) for the gg initial state process, and 1 (from 2) for the $q\bar{q}$ initial state process, due to chiral hermitian symmetry.

Values for the counterterm renormalisation constants are provided by the sgluon model file *CT_list_mod.map*. This file is defined for the sgluon model using the Feynman rules given in Appendix A.3, written in the style (reproduced from Eq. (4.15))

$$1 + \delta Z \rightarrow 1 + \text{CT_prefac} * (\text{CT_integral} + \text{CT_finite}). \quad (5.9)$$

From these analytical results, the fully renormalised virtual corrections FORTRAN90 module is created, and linked to the sgluon-specific integrated dipoles within the $2 \rightarrow 2$ MC phase space generator run by MADGOLEM.

Numerical computation

The numerical calculation of $pp \rightarrow GG^*$ uses the SM parameter set, with a variable sgluon mass m_G . The PDFs used by MADGOLEM are the CTEQ6L1 and CTEQ6M for LO and NLO, as in the $pp \rightarrow \tilde{q}\tilde{\chi}_1^0$ case. Factorisation and renormalisation scales

m_G [GeV]	σ^{LO} [pb]	σ^{NLO} [pb]	K
200	2.12×10^2	3.36×10^2	1.58
350	8.16×10^0	1.36×10^1	1.66
500	7.64×10^{-1}	1.34×10^0	1.75
750	3.40×10^{-2}	6.54×10^{-2}	1.93
1000	2.47×10^{-3}	5.29×10^{-3}	2.15

Table 5.7: Individual production rates $\sigma(pp \rightarrow GG^*)$ and corresponding K -factors for different sgluon masses for the LHC at $\sqrt{S} = 8$ TeV.

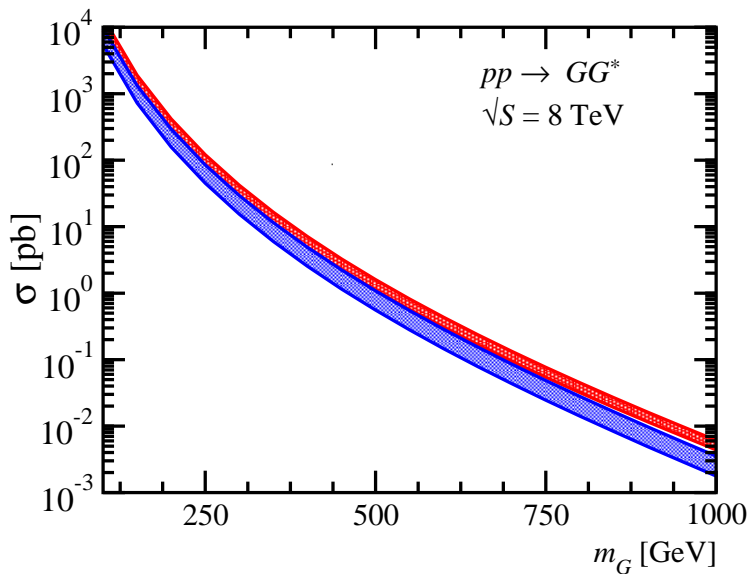


Figure 5.18: Band plots for sgluon pair production (blue = LO, red = NLO).

are set to their central values:

$$\mu^0 \equiv \mu_R^0 = \mu_F^0 = m_G. \quad (5.10)$$

The NLO-QCD cross sections and K factors for sgluon pair production calculated at the LHC center of mass energy $\sqrt{S} = 8$ TeV are provided in Table 5.7, broken down into separate sgluon masses. A list of cross sections and K factors for sgluon production at 14 TeV for variable m_G is presented in the MADGOLEM paper [15].

Band plots for total sgluon pair production are presented in Figure 5.18 for the LHC center of mass energy $\sqrt{S} = 8$ TeV, as a function of the sgluon mass m_G . The band plots are obtained over the range $\frac{1}{2}\mu^0 \leq \mu_{R,F} \leq 2\mu^0$. The error bands show a reduction in theoretical uncertainty from $\mathcal{O}(80\%)$ at LO to $\mathcal{O}(30\%)$ at NLO.

A topological breakdown of the relative contributions (as defined in Eq. (5.5)) to $pp \rightarrow GG^*$ is presented in Figure 5.19.

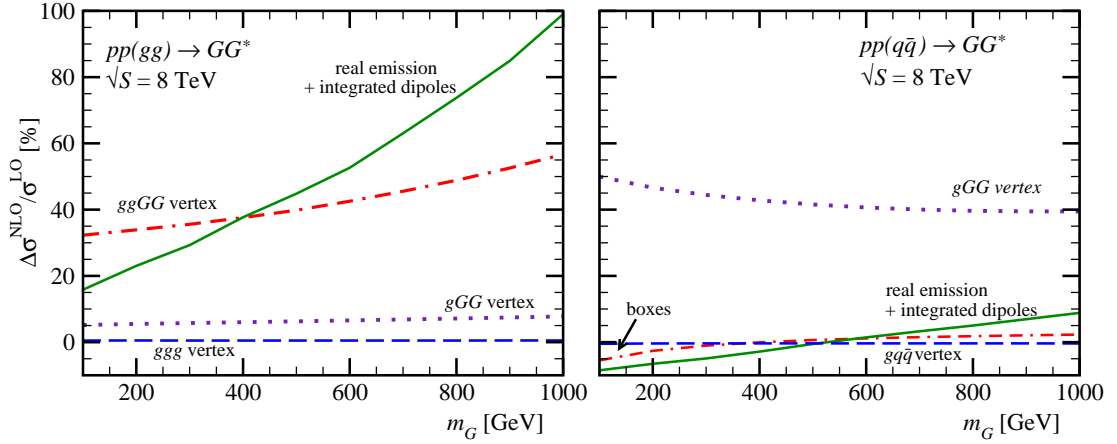


Figure 5.19: Individual NLO contributions to $pp \rightarrow GG^*$ by relative size: $\Delta\sigma^{\text{NLO}}/\sigma^{\text{LO}}$. Contributions from self energy diagrams are negligible ($< 1\%$) and not shown.

The dominant contributions to the gluon fusion channel $gg \rightarrow GG^*$ can be seen to come from the combined real emission and integrated dipoles, and the virtual corrections to the quartic vertex $ggGG^*$, which includes all box diagrams; both of which increase with the sgluon mass m_g . These features reveal a clear parallel with the visible threshold effects observed in the gluon fusion channel for squark antisquark production ($gg \rightarrow \tilde{q}\tilde{q}^*$). This is unsurprising, as the LO and NLO–QCD process channels $gg \rightarrow GG^*$ and $gg \rightarrow \tilde{q}\tilde{q}^*$ are topologically very similar (comparing Figures 5.16 & 5.7), and the differences between the two at LO (comparing Figures 5.15 & 5.6) can be traced directly to the relative strength of the color interactions arising from the fundamental vs adjoint final–state scalars.

The topological features of the quark antiquark annihilation channel $q\bar{q} \rightarrow GG^*$ are not similar to the squark antisquark production channel $q\bar{q} \rightarrow \tilde{q}\tilde{q}^*$ however; the dominant contribution to $q\bar{q} \rightarrow GG^*$ comes from the virtual corrections to the gGG^* vertex, whereas the corresponding $g\tilde{q}\tilde{q}^*$ vertex corrections are almost negligible. This difference can be explained due to the additional t–channel gluino diagrams present at LO in $q\bar{q} \rightarrow \tilde{q}\tilde{q}^*$, that do not feature in $q\bar{q} \rightarrow GG^*$. In the MADGOLEM paper [15], it is pointed out that in the case of decoupled gluinos the topologies for $q\bar{q} \rightarrow GG^*$ and $q\bar{q} \rightarrow \tilde{q}\tilde{q}^*$ become similar, and the separate initial state production ratios can be calculated from the analytical results:

$$\frac{\sigma(q\bar{q} \rightarrow \tilde{q}\tilde{q}^*)}{\sigma(q\bar{q} \rightarrow GG^*)} = \frac{1}{6} \quad \frac{\sigma(gg \rightarrow \tilde{q}\tilde{q}^*)}{\sigma(gg \rightarrow GG^*)} = \frac{1}{20} . \quad (5.11)$$

The successful calculation of the process $pp \rightarrow GG^*$ provides good evidence for the usage of MADGOLEM to calculate NLO–QCD corrections for user–defined models, as

an automated model-independent tool.

The results of the MADGOLEM NLO-QCD calculation for the process $pp \rightarrow GG^*$ have been used in the ATLAS search for sgluons to exclude sgluon pair production at the LHC in the mass range of 100 GeV to 287 GeV [82].

Chapter 6

Conclusion

The intent of this thesis is to convince the reader that the automated calculation of one-loop processes by use of the virtual corrections module within MADGOLEM is:

- Based on a sound theoretical underpinning of amplitude and loop simplification techniques;
- Constructed within a robust framework of model-independent algorithms;
- Capable of calculating complicated $2 \rightarrow 2$ NLO-QCD processes across a variety of models.

NLO-QCD computations for physics beyond the Standard Model (BSM) are not a new concept, and $2 \rightarrow 2$ NLO calculations have been performed over the last several decades for MSSM and other BSM models, including many automated and semi-automated calculations by various tools.

However, MADGOLEM is the first instance of a computational framework that takes all the individual steps or modules involved in a model-independent NLO-QCD calculation, and combines them in a highly streamlined and automated manner that does not require the input of extra auxiliary numerical calculations¹. At variance with alternative tools presently under development, MADGOLEM is based on a Feynman diagrammatic description of the NLO-QCD amplitudes and is able to provide not only numerical, but also fully analytical results. Within the universal properties of QCD-driven interactions and through dedicated coding, this tool is ready to handle genuine BSM structures.

MADGOLEM has already been applied with success to frontline new physics searches [14–18, 82, 83]. As an independent, modular add-on to MADGRAPH,

¹In Appendix C, the steps required to produce an NLO-QCD cross section for any renormalisable ($SU(N_c) \otimes \dots$) model with MADGOLEM are described; the list covers less than a page.

MADGOLEM can be easily linked to the multiple analysis and simulation tools built upon the MADGRAPH architecture (MADANALYSIS5, FEYNRULES, MADWEIGHT) [81]. This makes it a most useful tool to bridge precise theory predictions and experimental data. Dedicated studies along these lines are underway [84].

Appendix A

Feynman rules

This appendix lists the set of Feynman rules for the various physical models that are used for the calculations in this thesis. All momenta are defined as incoming for vertices, and parameters for coupling constants other than g_s are left undefined in order to allow the Feynman rules to be used in a largely model-independent manner, as values for masses and charges can be inserted after the main calculation.

For $SU(N_c)$ structures we are more strict, and assume $N_c = 3$ throughout, with normalisation conventions $\text{Tr}(T^a T^a) = \frac{1}{2}$ and $[T^a, T^b] = if^{abc}T^c$. We also apply the color values $T_R = \frac{1}{2}$, $C_a = 3$ and $C_f = \frac{4}{3}$.

For Majorana and Dirac fermions, the arrows define the fixed direction of the fermion flow; Majoranas may have an original fermion flow that clashes with the fixed Dirac fermion flow in the amplitude, in which case the Majorana is flipped using the rules in Section 2.4. The Dirac structure of the Feynman diagram is constructed by following the fermion flow in reverse order.

For colored particles, the arrows define the direction of the color flow, which aligns with the fermion flow in cases where both exist.

Counterterms are expressed in terms of the scalar one- and two-point Passarino-Veltman integrals A_0 and B_0 . The derivative $B'_0 = \partial B_0 / \partial p^2$ is also used. The massless integral $B_0(0; 0, 0)$ is defined as the conflict of two divergences:

$$B_0(0; 0, 0) = \frac{1}{\tilde{\epsilon}_{UV}} - \frac{1}{\tilde{\epsilon}_{IR}}. \quad (\text{A.1})$$

A.1 Standard Model QCD

The Feynman rules in Figure A.1 correspond to the $SU(N_c)$ sector of the Standard Model, and its interactions (via quarks) with the electroweak sector. All couplings and propagators are written in the Feynman gauge; g_R and g_L are the right- and left-

handed coupling constants associated with the chiral projection operators $\Pi_{R/L} \equiv \Pi^\pm$. Faddeev-Popov ghosts are also included to allow for properly renormalised NLO QCD calculations. The electroweak Feynman rules are not described here, as they are not the main focus of the thesis, and can be found easily elsewhere.

Feynman rules for the counterterms are given in Figure A.2. The renormalisation term for the QCD coupling constant is:

$$\delta g_s = -\frac{\alpha_s}{4\pi} \frac{\beta_0^L + \beta_0^H}{2} \frac{1}{\tilde{\epsilon}} - \frac{\alpha_s}{4\pi} \frac{1}{3} \log \frac{m_t^2}{\mu_R^2}, \quad (\text{A.2})$$

where the light (L) and heavy (H) beta functions are

$$\beta_0^L = \frac{11}{3} N_c - \frac{2}{3} n_f, \quad \beta_0^H = -\frac{2}{3}. \quad (\text{A.3})$$

The renormalisation term for the gluon wavefunction is defined:

$$\delta Z_g = -\frac{\alpha_s}{4\pi} (\beta_0^L + \beta_0^H - 2N_c) B_0(0; 0, 0) + \frac{\alpha_s}{2\pi} \left[\frac{1}{3} \log \frac{m_t^2}{\mu_R^2} \right] \quad (\text{A.4})$$

An additional renormalisation term is required for external gluons, in order to account for the heavy fields' IR poles that remain in δZ_g via $B_0(0; 0, 0)$:

$$\delta Z_{g(\text{ext})} = \frac{\alpha_s}{4\pi} \frac{\beta_0^H}{2} \frac{1}{\tilde{\epsilon}_{IR}} \quad (\text{A.5})$$

The renormalisation terms for the massless quarks and the massive top quark are

$$\begin{aligned} \delta Z_q &= -\frac{\alpha_s}{4\pi} C_f B_0(0; 0, 0) \\ \delta Z_t &= \frac{\alpha_s}{4\pi} C_f \left[4m_t^2 B'_0(m_t^2; m_t^2, m_t^2) - \frac{1}{m_t^2} A_0(m_t^2) - 1 \right] \\ \delta m_t &= \frac{\alpha_s}{4\pi} C_f [-3B_0(m_t^2; 0, m_t^2) - 1] \end{aligned} \quad (\text{A.6})$$

A.2 Supersymmetry

A full account of Feynman rules for the MSSM can be found in [25]; in this appendix we will restrict ourselves mostly to the strongly-interacting sector of supersymmetry. Mixing between squarks other than the stops is not considered in this model, which therefore restricts us to only the light flavor squarks (superpartners of u, d, s, c, b; noted by $\tilde{q}_{L/R}$) and the stops (with mass eigenvalues \tilde{t}_1 and \tilde{t}_2). The Feynman rules in Figure A.3 are for the $SU(N_c)$ -dependent vertices and propagators: those that

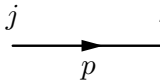
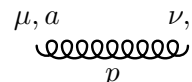
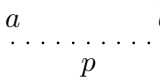
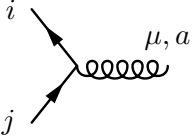
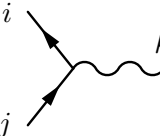
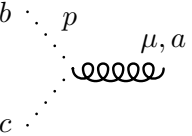
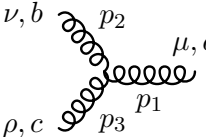
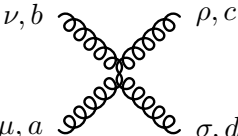
	$i \frac{\not{p} + m}{(p^2 - m^2)} \delta_i^j$		$-i \frac{1}{p^2} g^{\mu\nu} \delta^{ab}$
	$-i \frac{\not{p}}{p^2} \delta^{ab}$		$-i g_s (T^a)_i^j \gamma^\mu$
	$-i \gamma^\mu (g_R \Pi_R + g_L \Pi_L) \delta_i^j$		$g_s f^{abc} p^\mu$
	$-g_s f^{abc} (p_1 - p_2)^\rho g^{\mu\nu} + \text{cyclic}$		
	$-i g_s^2 f^{abe} f^{cde} (g^{\mu\rho} g^{\nu\sigma} - g^{\mu\sigma} g^{\nu\rho}) + \text{cyclic}$		

Figure A.1: Feynman rules for the $SU(N_c)$ sector of the Standard Model.

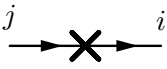

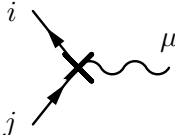
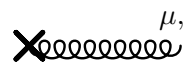
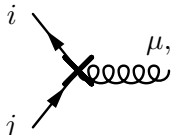
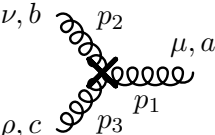
	$i [(\not{p} - m_q) \delta Z_q - \delta m_q] \delta_i^j$		$-i p^2 \delta Z_g g^{\mu\nu} \delta^{ab}$
	$-i \gamma^\mu (g_R \Pi_R + g_L \Pi_L) \delta Z_q \delta_i^j$		$\delta Z_{g(\text{ext})} \epsilon^{\mu, a}$
	$-i g_s (T^a)_i^j \left[\delta g_s + \delta Z_q + \frac{1}{2} \delta Z_g \right] \gamma^\mu$		
	$-g_s f^{abc} \left[\delta g_s + \frac{3}{2} \delta Z_g \right] (p_1 - p_2)^\rho g^{\mu\nu} + \text{cyclic}$		

Figure A.2: Feynman rules for the $SU(N_c)$ counterterms of the Standard Model.

include gluons, gluinos, quarks and squarks. Right- and left-handed sfermions are associated with the chiral projection operators $\Pi_{R/L} \equiv \Pi^\pm$. The definition for the squark quartic color structure S_{ijkl} depends on the flavor f_i of the particles and the $s_i = L/R$ handedness of their Standard Model partners:

$$\begin{aligned}
 S_{ijkl} &= + (T^a)_i^j (T^a)_k^l + (T^a)_i^l (T^a)_k^j ; & s_i = s_j , s_j = h_l ; & f_i = f_j , f_j = f_l \\
 S_{ijkl} &= + (T^a)_i^j (T^a)_k^l : & s_i = s_j , s_j = s_l ; & f_i \neq f_j , f_j \neq f_l \\
 S_{ijkl} &= - (T^a)_i^j (T^a)_k^l : & s_i \neq s_j , s_j \neq s_l &
 \end{aligned} \tag{A.7}$$

Feynman rules for the electroweak sector are given in Figure A.4. The coupling parameters $a_{L/R}$, $b_{L/R}$, $g_{1,2}$ are defined in the appendix of [85].

The counterterms for the strongly interacting supersymmetric particles at one loop are described in detail in [16]. Feynman rules for the counterterms are given in Figure A.5, with renormalisation terms defined below. The QCD coupling renormalisation is defined:

$$\delta g_s = -\frac{\alpha_s}{4\pi} \frac{\beta_0^L + \beta_0^H}{2} \frac{1}{\tilde{\epsilon}} - \frac{\alpha_s}{4\pi} \left[\frac{1}{3} \log \frac{m_t^2}{\mu_R^2} + \log \frac{m_{\tilde{g}}^2}{\mu_R^2} + \frac{1}{12} \sum_{\text{squarks}} \log \frac{m_{\tilde{q}_j}^2}{\mu_R^2} \right] , \tag{A.8}$$

where the light (L) and heavy (H) beta functions are

$$\beta_0^L = \frac{11}{3} N_c - \frac{2}{3} n_f , \quad \beta_0^H = -\frac{2}{3} - \frac{2}{3} N_c - \frac{1}{3} (n_f + 1) . \tag{A.9}$$

As described in Section 3.4, the Yukawa coupling $\hat{g}_s^{(0)}$ for the vertex $q\tilde{q}\tilde{g}$ requires an additional correction in addition to δg_s to account for a mismatch in degrees of freedom between gluons and gluinos. This additional term is

$$\delta \hat{g}_s = \frac{\alpha_s}{4\pi} \left(\frac{2}{3} n_f - \frac{3}{2} C_f \right) . \tag{A.10}$$

The renormalisation term for the gluon wavefunction, which satisfies the Slavnov–Taylor identity $\delta Z_g(\text{finite}) = -2\delta g_s(\text{finite})$ is defined:

$$\delta Z_g = -\frac{\alpha_s}{4\pi} (\beta_0^L + \beta_0^H) B_0(0; 0, 0) + \frac{\alpha_s}{2\pi} \left[\frac{1}{3} \log \frac{m_t^2}{\mu_R^2} + \log \frac{m_{\tilde{g}}^2}{\mu_R^2} + \frac{1}{12} \sum_{\text{squarks}} \log \frac{m_{\tilde{q}_j}^2}{\mu_R^2} \right] \tag{A.11}$$

An additional renormalisation term is required for external gluons, in order to account

for the heavy fields' IR poles that remain in δZ_g via $B_0(0; 0, 0)$:

$$\delta Z_{g(\text{ext})} = \frac{\alpha_s}{4\pi} \frac{\beta_0^H}{2} \frac{1}{\tilde{\epsilon}_{IR}} \quad (\text{A.12})$$

The renormalisation terms for the light quarks are defined:

$$\begin{aligned} \delta Z_{q_{L/R}} = & -\frac{\alpha_s}{4\pi} C_f \left[B_0(0; 0, 0) + B_0(0; m_g^2, m_g^2) \right. \\ & \left. + (m_g^2 - m_{q_{L/R}}^2) B'_0(0; m_g^2, m_{q_{L/R}}^2) + (m_g^2 - m_{q_{R/L}}^2) B'_0(0; m_g^2, m_{q_{R/L}}^2) \right] \end{aligned} \quad (\text{A.13})$$

Renormalisation terms for the massive squarks ($s = L/R$) are defined:

$$\begin{aligned} \delta Z_{\tilde{q}_s} = & \frac{\alpha_s}{2\pi} C_f \left[[B_0(m_{\tilde{q}_s}^2, 0, m_{\tilde{q}_s}^2) + m_{\tilde{q}_s}^2 B'_0(m_{\tilde{q}_s}^2, 0, m_{\tilde{q}_s}^2) \right. \\ & \left. - B_0(m_{\tilde{q}_s}^2, m_g^2, 0) + (m_g^2 - m_{\tilde{q}_s}^2) B'_0(m_{\tilde{q}_s}^2, m_g^2, 0) \right] \quad (\text{A.14}) \\ \delta m_{\tilde{q}_s} = & -\frac{\alpha_s}{4\pi} C_f \left[4m_{\tilde{q}_s}^2 + 3A_0(m_{\tilde{q}_s}^2) + 2A_0(m_g^2) + 2(m_g^2 - m_{\tilde{q}_s}^2) B_0(m_{\tilde{q}_s}^2, m_g^2, 0) \right] \end{aligned}$$

Renormalisation terms for the gluino field and mass are:

$$\begin{aligned} \delta Z_{\tilde{g}} = & \frac{\alpha_s}{4\pi} N_c \left[1 + 4m_g^2 B'_0(m_g^2, 0, m_g^2) - A_0(m_g^2)/m_g^2 \right] \\ & + \frac{\alpha_s}{8\pi m_g^2} \sum_{\text{light}} \left[A_0(m_q^2) - (m_g^2 + m_q^2) B_0(m_g^2, 0, m_q^2) \right. \\ & \left. - 2m_g^2 (m_g^2 - m_q^2) B'_0(m_g^2, 0, m_q^2) \right] \\ & + \frac{\alpha_s}{8\pi m_g^2} \sum_{\text{heavy}} \left[2m_g^2 (m_q^2 - m_q^2 - m_g^2) B'_0(m_g^2, m_q^2, m_{\tilde{q}_s}^2) \right. \\ & \left. + (m_q^2 - m_q^2 - m_g^2) B_0(m_g^2, m_q^2, m_q^2) + A_0(m_q^2) - A_0(m_{\tilde{q}_s}^2) \right] \\ & + \frac{\alpha_s}{\pi} \sum_{\text{heavy}} m_{\tilde{g}} m_q R_{s1}^q R_{s2}^q B'_0(m_g^2, m_q^2, m_{\tilde{q}_s}^2) \end{aligned} \quad (\text{A.15})$$

$$\begin{aligned} \delta m_{\tilde{g}} = & -\frac{\alpha_s}{4\pi} N_c m_{\tilde{g}} \left[1 + 3A_0(m_g^2)/m_g^2 \right] \\ & + \frac{\alpha_s}{8\pi m_{\tilde{g}}} \sum_{\text{light}} \left[A_0(m_q^2) + (m_g^2 - m_q^2) B_0(m_g^2, 0, m_q^2) \right] \\ & + \frac{\alpha_s}{8\pi m_{\tilde{g}}} \sum_{\text{heavy}} \left[(m_g^2 + m_q^2 - m_q^2) B_0(m_g^2, m_q^2, m_q^2) \right. \\ & \left. - 4m_q m_{\tilde{g}} R_{s1}^q R_{s2}^q B_0(m_g^2, m_q^2, m_{\tilde{q}_s}^2) - A_0(m_q^2) + A_0(m_{\tilde{q}_s}^2) \right] \end{aligned} \quad (\text{A.16})$$

Summations are taken over the light/heavy quarks with their respective superpartners. As the bottom quark is considered to be massless, only the two stops and top quark contribute to the heavy sum. R_s^q is defined as the mass mixing matrix to create the

$$\begin{array}{ll}
 \begin{array}{c} j \\ \text{---} \end{array} \xrightarrow{p} \begin{array}{c} i \\ \text{---} \end{array} & \frac{i}{(p^2 - m^2)} \delta_i^j & \begin{array}{c} a \\ \text{---} \end{array} \xrightarrow{p} \begin{array}{c} b \\ \text{---} \end{array} & i \frac{\not{p} + m}{(p^2 - m^2)} \delta^{ab} \\
 \\
 \begin{array}{c} i \\ \nearrow \\ j \end{array} \begin{array}{c} \text{---} \\ \mu, a \end{array} & i g_s (T^a)_i^j (p_i - p_j)^\mu & \begin{array}{c} b \\ \nearrow \\ a \end{array} \begin{array}{c} \text{---} \\ \mu, c \end{array} & - g_s f^{abc} \gamma^\mu \\
 \\
 \begin{array}{c} a \\ \nearrow \\ j \end{array} \begin{array}{c} \text{---} \\ i(L/R) \end{array} & i g_s \sqrt{2} (T^a)_i^j (\pm \Pi_{R/L}) & \begin{array}{c} a \\ \nearrow \\ i \end{array} \begin{array}{c} \text{---} \\ j(L/R) \end{array} & - i g_s \sqrt{2} (T^a)_i^j (\pm \Pi_{R/L}) \\
 \\
 \begin{array}{c} \nu, b \\ \text{---} \\ \mu, a \end{array} \begin{array}{c} \text{---} \\ j \\ \text{---} \\ i \end{array} & i g_s^2 \left(\{T^a, T^b\} \right)_i^j g^{\mu\nu} & \begin{array}{c} l \\ \text{---} \\ j \end{array} \begin{array}{c} \text{---} \\ i \\ \text{---} \\ k \end{array} & - i g_s^2 \frac{1}{2} S_{ijkl} \\
 \\
 \begin{array}{c} \nu \\ \text{---} \\ \mu, a \end{array} \begin{array}{c} \text{---} \\ j \\ \text{---} \\ i \end{array} & i g_s g (T^a)_i^j g^{\mu\nu} & &
 \end{array}$$

Figure A.3: Feynman rules for supersymmetric QCD, excluding squark mixing. For colorless fermions $\delta_i^j \rightarrow 1$.

mass eigenstates $\tilde{t}_{1,2}$ from the chiral states $\tilde{t}_{L,R}$; in the terms where $R_{s_{1,2}}^q$ is present, a summation over the chiralities ($s_{1,2} = L/R$) is implicit.

A.3 Scalar Gluons

The Lagrangian describing the sgluon interactions with SM gluons is

$$\mathcal{L}_G = D_\mu G^* D^\mu G - m_G^2 G G^* , \quad (\text{A.17})$$

with covariant derivative

$$D_\mu G^a \equiv \partial_\mu G^a + g_s f^{abc} G^b A_\mu^c , \quad (\text{A.18})$$

where G^a denotes the sgluon field and A_μ^c the gluon field. The Feynman rules for sgluon interactions are given in Figure A.6.

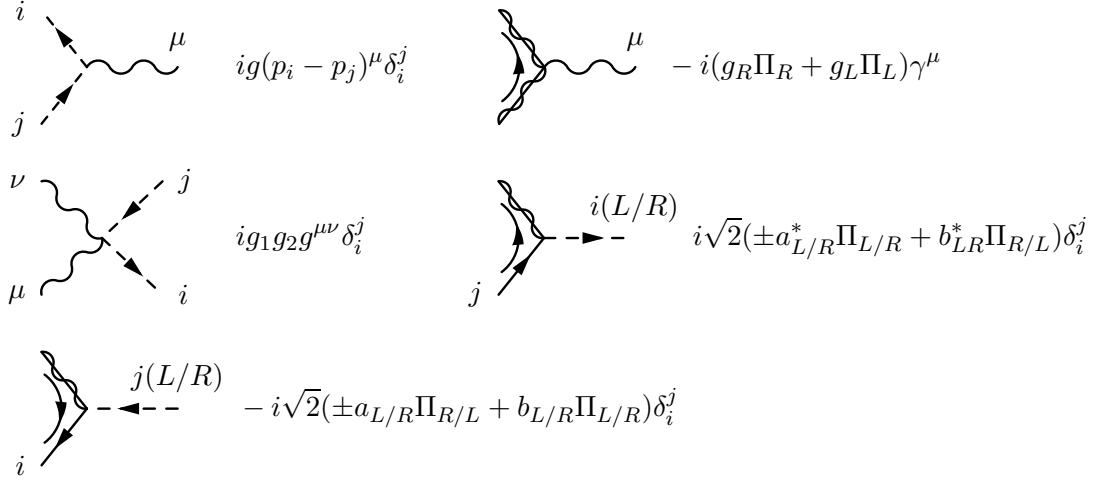


Figure A.4: Feynman rules for the supersymmetric extension to the electroweak sector, excluding mixing. For colorless fermions $\delta_i^j \rightarrow 1$.

The counterterms for scalar gluons and the strong coupling constant at one loop are described in detail in [15]. Feynman rules for the counterterms are given in Figure A.7, with renormalisation terms defined as:

$$\begin{aligned} \delta g_s(\mu_R) &= -\frac{\alpha_s(\mu_R)}{4\pi} \frac{\beta_0^L + \beta_0^H}{2} \frac{1}{\tilde{\epsilon}} - \frac{\alpha_s(\mu_R)}{4\pi} \left[\frac{1}{3} \log \frac{m_t^2}{\mu_R^2} + \frac{1}{2} \log \frac{m_G^2}{\mu_R^2} \right] \\ \delta Z_g &= \frac{\alpha_s}{4\pi} (\beta_0^L + \beta_0^H - 4N_c/3) B_0(0; 0, 0) + \frac{\alpha_s}{2\pi} \left[\frac{1}{3} \log \frac{m_t^2}{\mu_R^2} + \frac{1}{2} \log \frac{m_G^2}{\mu_R^2} \right] \end{aligned} \quad (\text{A.19})$$

The light (L) and heavy (H) beta functions for the QCD coupling constant are defined:

$$\beta_0^L = \frac{11}{3} N_c - \frac{2}{3} n_f, \quad \beta_0^H = -\frac{2}{3} - \frac{1}{3} N_c. \quad (\text{A.20})$$

An additional renormalisation term is required for external gluons, in order to account for the heavy fields' IR poles that remain in δZ_g via $B_0(0; 0, 0)$:

$$\delta Z_{g(\text{ext})} = -\frac{1}{3} \frac{\alpha_s}{4\pi} \frac{1}{\tilde{\epsilon}_{IR}} \quad (\text{A.21})$$

The renormalisation terms for the sgluon wavefunction and mass are:

$$\begin{aligned} \delta Z_G &= \frac{\alpha_s}{2\pi} N_c [B_0(m_G^2; m_G^2, 0) + m_G^2 B'_0(m_G^2; m_G^2, 0)] \\ \delta m_G &= -\frac{\alpha_s}{4\pi} N_c [4m_G^2 + 3A_0(m_G^2)] . \end{aligned} \quad (\text{A.22})$$

$$\begin{aligned}
 & \text{Diagram 1: } i[(p^2 - m_{\tilde{q}}^2)\delta Z_{\tilde{q}} - \delta m_{\tilde{q}}]\delta_i^j \quad \text{Diagram 2: } i[(\not{p} - m_{\tilde{g}})\delta Z_{\tilde{g}} - \delta m_{\tilde{g}}]\delta_{ab} \\
 & \text{Diagram 3: } -ig_s(T^a)_i^j \left[\delta g_s + \frac{1}{2}(\delta Z_{\tilde{q}_i} + \delta Z_{\tilde{q}_j} + \delta Z_g) \right] (p_i - p_j)^\mu \\
 & \text{Diagram 4: } ig_s\sqrt{2}(T^a)_i^j \left[\delta g_s + \delta \hat{g}_s + \frac{1}{2}(\delta Z_{\tilde{q}_i} + \delta Z_{\tilde{q}_j} + \delta Z_g) \right] (\pm\Pi_{R/L}) \\
 & \text{Diagram 5: } -ig_s\sqrt{2}(T^a)_i^j \left[\delta g_s + \delta \hat{g}_s + \frac{1}{2}(\delta Z_{\tilde{q}_i} + \delta Z_{\tilde{q}_j} + \delta Z_g) \right] (\pm\Pi_{R/L}) \\
 & \text{Diagram 6: } -g_s f^{abc} \left[\delta g_s + \delta Z_{\tilde{g}} + \frac{1}{2}\delta Z_g \right] \gamma^\mu \\
 & \text{Diagram 7: } ig_s^2(\{T^a, T^b\})_i^j \left[\delta g_s + \delta Z_g + \frac{1}{2}(\delta Z_{\tilde{q}_i} + \delta Z_{\tilde{q}_j}) \right] g^{\mu\nu}
 \end{aligned}$$

Figure A.5: Feynman rules for strongly interacting MSSM counterterms.

$$\begin{aligned}
 & \text{Diagram 1: } \frac{i}{(p^2 - m^2)}\delta_{ab} \quad \text{Diagram 2: } g_s f^{abc}(p_a - p_b)^\mu \\
 & \text{Diagram 3: } ig_s^2(f^{ace}f^{bde} + f^{ade}f^{bce})g^{\mu\nu}
 \end{aligned}$$

 Figure A.6: Feynman rules for scalar gluons as an extension to the Standard Model. Arrows define the ‘direction’ $G^* \rightarrow G$.

$$\begin{array}{ll}
 \begin{array}{c} b \\ \dashrightarrow \end{array} \begin{array}{c} \times \\ p \end{array} \begin{array}{c} \dashrightarrow \\ a \end{array} & ((p^2 - m_G^2)\delta Z_G - \delta m_G) \delta_{ab} \\
 \begin{array}{c} a \\ \dashrightarrow \end{array} \begin{array}{c} \times \\ \mu, c \end{array} \begin{array}{c} \dashrightarrow \\ b \end{array} & g_s (\delta g_s + \frac{1}{2}(\delta Z_G + \delta Z_{G^*} + \delta Z_g)) f^{abc} (p_a - p_b)^\mu \\
 \begin{array}{c} \nu, d \\ \dashrightarrow \end{array} \begin{array}{c} \times \\ \nu, d \end{array} \begin{array}{c} \dashrightarrow \\ b \end{array} & i g_s^2 (2\delta g_s + \delta Z_g + \delta Z_G) (f^{ace} f^{bde} + f^{ade} f^{bce}) g^{\mu\nu} \\
 \begin{array}{c} \mu, c \\ \dashrightarrow \end{array} \begin{array}{c} \times \\ \mu, c \end{array} \begin{array}{c} \dashrightarrow \\ a \end{array} &
 \end{array}$$

Figure A.7: Feynman rules for scalar gluon counterterms. Arrows define the ‘direction’ $G^* \rightarrow G$.

Appendix B

Form factors

The form factors in Eqs. (3.23)–(3.27) can be directly calculated numerically by the integral library GOLEM95 [43, 44]. Alternatively, they can be decomposed into scalar integrals using Eq. (3.22) and Eqs. (B.1) & (B.2) below.

$$\mathcal{I}_n^D(S) = \sum_{k \in S} b_k \mathcal{I}_{n-1}^D(S^{\{k\}}) - B(n - D - 1) \mathcal{I}_n^{D+2}(S) \quad (\text{B.1})$$

$$\begin{aligned} \mathcal{I}_n^D(j_0, \dots, j_r; S) &= - \sum_{i=1}^r \mathcal{S}_{j_0 j_i}^{-1} \mathcal{I}_n^{D+2}(j_1, \dots, j_{i-1}, j_{i+1}, \dots, j_r; S) \\ &\quad + \sum_{k \in S} \mathcal{S}_{j_0 k}^{-1} \mathcal{I}_{n-1}^D(j_1, \dots, j_r; S^{\{k\}}) \\ &\quad - b_{j_0} (n - D - r - 1) \mathcal{I}_n^{D+2}(j_1, \dots, j_r; S) \end{aligned} \quad (\text{B.2})$$

The notation $S^{\{k\}}$ here removes or ‘pinches’ the k -th propagator from the ordered set $S = \{1, \dots, n\}$. The Feynman parameter integral in Eq. (B.2) is defined:

$$\mathcal{I}_n^D(j_1, \dots, j_r; S) = (-1)^n \Gamma(n - \frac{D}{2}) \int \prod_{i=1}^n dz_i \left(1 - \sum_{k=1}^n z_k \right) z_{j_1} \cdots z_{j_r} (R^2)^{\frac{D}{2} - n} \quad (\text{B.3})$$

R^2 is defined:

$$R^2 = -\frac{1}{2} \sum_{i,j=1}^n z_i \mathcal{S}_{ij} z_j - i\epsilon, \quad (\text{B.4})$$

The kinematic matrix \mathcal{S}_{ij} associated with the loop is defined:

$$\mathcal{S}_{ij} = (r_i - r_j)^2 - m_i^2 - m_j^2, \quad (\text{B.5})$$

and is invertible for processes with *non-exceptional* kinematics [46]. The two terms b_j and B are defined:

$$b_j = \sum_{k \in S} \mathcal{S}_{kj}^{-1}, \quad B = \sum_{j \in S} b_j \quad (\text{B.6})$$

The case of *exceptional* kinematics occurs when $B \rightarrow 0^1$, which causes numerical instability in the scalar integral representation. These kinematical configurations are extremely rare in $2 \rightarrow 2$ processes, and can still be calculated within the pure form factor representation by GOLEM95.

B.1 1- and 2-point form factors

The A and B form factors for the case $n = 1, 2$ in Eqs. (B.7) & (B.8) below can be written purely in terms of UV-divergent scalar integrals $\mathcal{I}_{1,2}^D(S)$.

$$\begin{aligned} A^{1,0}(S) &= \mathcal{I}_1^D(S) \\ A_j^{1,1}(S) &= -\frac{b_j}{B} \mathcal{I}_1^D(S) \end{aligned} \quad (\text{B.7})$$

$$\begin{aligned} B^{2,2}(S) &= -\frac{1}{2B} \frac{1}{(D-1)} \left[\mathcal{I}_2^D(S) - \sum_{k \in S} b_k \mathcal{I}_1^D(S^{\{k\}}) \right] \\ A^{2,0}(S) &= \mathcal{I}_2^D(S) \\ A_j^{2,1}(S) &= 2(D-1)b_j B^{2,2}(S) - \sum_{k \in S} \mathcal{S}_{jk}^{-1} \mathcal{I}_1^D(S^{\{k\}}) \\ A_{j_1, j_2}^{2,2}(S) &= 2 \left(\mathcal{S}_{j_1 j_2}^{-1} - D \frac{b_{j_1} b_{j_2}}{B} \right) B^{2,2}(S) \\ &\quad + \frac{1}{B} \sum_{k \in S} \left(b_{j_1} \mathcal{S}_{j_2 k}^{-1} + b_{j_2} \mathcal{S}_{j_1 k}^{-1} - \frac{b_{j_1} b_{j_2} b_k}{B} \right) \mathcal{I}_1^D(S^{\{k\}}) \end{aligned} \quad (\text{B.8})$$

B.2 3-point form factors

The A and B form factors for the case $n = 3$ in Eqs. (B.9) & (B.9) below are written in terms of two UV-divergent scalar integrals $\mathcal{I}_{1,2}^D(S)$, and one IR-divergent scalar integral

¹Equivalent to the presence of inverse Gram determinants where $\det G = 0$.

$\mathcal{I}_3^D(S)$.

$$\begin{aligned}
 B^{3,2}(S) &= -\frac{1}{2B} \frac{1}{(D-2)} \left[\mathcal{I}_3^D(S) - \sum_{k \in S} b_k \mathcal{I}_2^D(S^{\{k\}}) \right] \\
 B_j^{3,3}(S) &= \frac{1}{2B} \left[-2b_j B^{3,2}(S) + \frac{1}{D-1} \sum_{k \in S} \left(\mathcal{S}_{jk}^{-1} \mathcal{I}_2^D(S^{\{k\}}) + b_k A_j^{2,1}(S^{\{k\}}) \right) \right]
 \end{aligned} \tag{B.9}$$

$$\begin{aligned}
 A^{3,0}(S) &= \mathcal{I}_3^D(S) \\
 A_j^{3,1}(S) &= 2(D-2)b_j B^{3,2}(S) - \sum_{k \in S} \mathcal{S}_{jk}^{-1} \mathcal{I}_2^D(S^{\{k\}}) \\
 A_{j_1, j_2}^{3,2}(S) &= 2\mathcal{S}_{j_1 j_2}^{-1} B^{3,2}(S) + 2b_{j_1} (D-1) B_{j_2}^{3,3}(S) - \sum_{k \in S} \mathcal{S}_{j_1 k}^{-1} A_{j_2}^{2,1}(S^{\{k\}}) \\
 A_{j_1, j_2, j_3}^{3,3}(S) &= 2 \left(\mathcal{S}_{j_1 j_2}^{-1} - \frac{b_{j_1} b_{j_2}}{B} \right) B_{j_3}^{3,3}(S) + 2 \left(\mathcal{S}_{j_1 j_3}^{-1} - \frac{b_{j_1} b_{j_3}}{B} \right) B_{j_2}^{3,3}(S) \\
 &\quad - \frac{b_{j_1}}{B} A_{j_2, j_3}^{3,2}(S) - \sum_{k \in S} \left(\mathcal{S}_{j_1 k}^{-1} - \frac{b_{j_1} b_k}{B} \right) A_{j_2, j_3}^{2,2}(S^{\{k\}})
 \end{aligned} \tag{B.10}$$

B.3 4-point form factors

The form factors for $n = 4$ are given below in Eqs. (B.12)–(B.14) below, and are written in terms of a basis of three divergent scalar integrals $\mathcal{I}_{1,2,3}^D(S)$ and one finite scalar integral $\mathcal{I}_4^{D+2}(S)$. If using a purely D -dimensional basis is preferred, one can use Eq. (B.1) to derive the substitution

$$\mathcal{I}_4^{D+2}(S) = \frac{1}{B(D-3)} \left[\mathcal{I}_4^D(S) - \sum_{k \in S} b_k \mathcal{I}_3^D(S^{\{k\}}) \right]. \tag{B.11}$$

$$C^{4,4}(S) = \frac{1}{4B} \frac{1}{(D-1)} \left[\mathcal{I}_4^{D+2}(S) + 2 \sum_{k \in S} b_k B^{3,2}(S^{\{k\}}) \right] \tag{B.12}$$

$$\begin{aligned}
 B^{4,2}(S) &= -\frac{1}{2} \mathcal{I}_4^{D+2}(S) \\
 B_j^{4,3}(S) &= \frac{1}{2} \frac{1}{B} \left[b_j \mathcal{I}_4^{D+2}(S) + \frac{1}{2} \sum_{k \in S} \mathcal{S}_{jk}^{-1} \mathcal{I}_3^D(S^{\{k\}}) + \frac{1}{2} \sum_{k \in S} b_k A_j^{3,1}(S^{\{k\}}) \right] \\
 B_{j_1, j_2}^{4,4}(S) &= -\frac{1}{3B} \left[2b_{j_1} B_{j_2}^{4,3}(S) + 2b_{j_2} B_{j_1}^{4,3}(S) - \frac{1}{2} \mathcal{S}_{j_1 j_2}^{-1} \mathcal{I}_4^{D+2}(S) \right. \\
 &\quad \left. - \frac{1}{4} \sum_{k \in S} \left(\mathcal{S}_{kj_2}^{-1} A_{j_1}^{3,1}(S^{\{k\}}) + \mathcal{S}_{kj_1}^{-1} A_{j_2}^{3,1}(S^{\{k\}}) + 2b_k A_{j_1, j_2}^{3,2}(S^{\{k\}}) \right) \right]
 \end{aligned} \tag{B.13}$$

$$\begin{aligned}
 A^{4,0}(S) &= \mathcal{I}_4^D(S) \\
 A_j^{4,1}(S) &= -b_j \mathcal{I}_4^{D+2}(S) - \sum_{k \in S} \mathcal{S}_{jk}^{-1} \mathcal{I}_3^D(S^{\{k\}}) \\
 A_{j_1, j_2}^{4,2}(S) &= 2b_{j_1} B_{j_2}^{4,3}(S) + 2b_{j_2} B_{j_1}^{4,3}(S) - \mathcal{S}_{j_1 j_2}^{-1} \mathcal{I}_4^{D+2}(S) \\
 &\quad - \frac{1}{2} \sum_{k \in S} \left[\mathcal{S}_{kj_2}^{-1} A_{j_1}^{3,1}(S^{\{k\}}) + \mathcal{S}_{kj_1}^{-1} A_{j_1}^{3,1}(S^{\{k\}}) \right] \\
 A_{j_1, j_2, j_3}^{4,3}(S) &= \frac{4}{3} \left[\mathcal{S}_{j_1 j_2}^{-1} B_{j_3}^{4,3}(S) + \mathcal{S}_{j_1 j_3}^{-1} B_{j_2}^{4,3}(S) + \mathcal{S}_{j_2 j_3}^{-1} B_{j_1}^{4,3}(S) \right] \\
 &\quad + 2 \left[b_{j_1} B_{j_2, j_3}^{4,4}(S) + b_{j_2} B_{j_1, j_3}^{4,4}(S) + b_{j_3} B_{j_1, j_2}^{4,4}(S) \right] \\
 &\quad - \frac{1}{3} \sum_{k \in S} \left[\mathcal{S}_{kj_1}^{-1} A_{j_2, j_3}^{3,2}(S^{\{k\}}) + \mathcal{S}_{kj_2}^{-1} A_{j_1, j_3}^{3,2}(S^{\{k\}}) \right. \\
 &\quad \left. + \mathcal{S}_{kj_3}^{-1} A_{j_1, j_2}^{3,2}(S^{\{k\}}) \right] \\
 A_{j_1, j_2, j_3, j_4}^{4,4}(S) &= \mathcal{S}_{j_1 j_2}^{-1} B_{j_3, j_4}^{4,4}(S) + \mathcal{S}_{j_1 j_3}^{-1} B_{j_2, j_4}^{4,4}(S) + \mathcal{S}_{j_1 j_4}^{-1} B_{j_2, j_3}^{4,4}(S) \\
 &\quad + \mathcal{S}_{j_3 j_4}^{-1} B_{j_1, j_2}^{4,4}(S) + \mathcal{S}_{j_2 j_4}^{-1} B_{j_1, j_3}^{4,4}(S) + \mathcal{S}_{j_2 j_3}^{-1} B_{j_1, j_4}^{4,4}(S) \\
 &\quad + g^{4,4}(j_1; j_2, j_3, j_4) + g^{4,4}(j_2; j_1, j_3, j_4) \\
 &\quad + g^{4,4}(j_3; j_2, j_1, j_4) + g^{4,4}(j_4; j_2, j_3, j_1)
 \end{aligned} \tag{B.14}$$

The function $g^{4,4}(j_1; j_2, j_3, j_4)$ used in the definition of $A^{4,4}$ is symmetric about $\{j_2 \leftrightarrow j_3 \leftrightarrow j_4\}$, and is defined:

$$\begin{aligned}
 g^{4,4}(j_1; j_2, j_3, j_4) &= \frac{b_{j_1}}{B} \left[-b_{j_4} B_{j_2, j_3}^{4,4}(S) - b_{j_3} B_{j_2, j_4}^{4,4}(S) - b_{j_2} B_{j_3, j_4}^{4,4}(S) \right. \\
 &\quad - \frac{1}{3} \left(\mathcal{S}_{j_2 j_3}^{-1} B_{j_4}^{4,3}(S) + \mathcal{S}_{j_2 j_4}^{-1} B_{j_3}^{4,3}(S) + \mathcal{S}_{j_3 j_4}^{-1} B_{j_2}^{4,3}(S) \right) \\
 &\quad + \frac{1}{12} \sum_{k \in S} \left[\mathcal{S}_{kj_4}^{-1} A_{j_2, j_3}^{3,2}(S^{\{k\}}) + \mathcal{S}_{kj_3}^{-1} A_{j_2, j_4}^{3,2}(S^{\{k\}}) \right. \\
 &\quad \left. + \mathcal{S}_{kj_2}^{-1} A_{j_3, j_4}^{3,2}(S^{\{k\}}) + \left(3b_k - \frac{1}{4} \mathcal{S}_{kj_1}^{-1} \right) A_{j_2, j_3, j_4}^{3,3}(S^{\{k\}}) \right] .
 \end{aligned} \tag{B.15}$$

Appendix C

Running MADGOLEM

1. Install MADGOLEM as described in the *README*.
2. Create a new working directory: `cp Template_nlo/ my_newprocess/`.
3. `cd my_newprocess/`.
4. Define the NLO–QCD process for calculation in *Cards/proc_card.dat*. If the model to be calculated does not already have a set of MADGOLEM model files (*interactions-qgraf.dat*, *particles-qgraf.dat*, *vertex_ct.dat*, *selfenergy_ct.dat*, *CT_list_mod.map*) in *../Models/my_model/*, these must be created¹.
5. Run `.bin/newprocess_nlo` to create the NLO–QCD process.
6. Run `perl GOLEMproc/run_golem.pl` to compute the analytical virtual corrections and produce the numerical code.
7. Define the model parameter set (masses, widths, etc) in *Cards/param_card.dat*.
8. Define the runtime parameters (\sqrt{S} , PDFs, etc) in *Cards/run_card.dat*.
9. Run `.bin/generate_events_nlo` to begin numerical calculation by Monte Carlo event generation; final results are output to *index.html*
10. Congratulations!

¹It is recommended to use the premade model files in *../Models/sm_nlo* and *../Models/mssm_nlo* as examples, as well as FEYNRULES [81]

Bibliography

- [1] S. Agrawal, T. Hahn, and E. Mirabella, “FormCalc 7,” *J.Phys.Conf.Ser.* **368** (2012) 012054, [arXiv:1112.0124 \[hep-ph\]](#).
- [2] G. Belanger, F. Boudjema, J. Fujimoto, T. Ishikawa, T. Kaneko, *et al.*, “Automatic calculations in high energy physics and Grace at one-loop,” *Phys.Rept.* **430** (2006) 117–209, [arXiv:hep-ph/0308080 \[hep-ph\]](#).
- [3] C. Berger, Z. Bern, L. Dixon, F. Febres Cordero, D. Forde, *et al.*, “An Automated Implementation of On-Shell Methods for One-Loop Amplitudes,” *Phys.Rev.* **D78** (2008) 036003, [arXiv:0803.4180 \[hep-ph\]](#).
- [4] S. Badger, B. Biedermann, and P. Uwer, “NGluon: A Package to Calculate One-loop Multi-gluon Amplitudes,” *Comput.Phys.Commun.* **182** (2011) 1674–1692, [arXiv:1011.2900 \[hep-ph\]](#).
- [5] P. Mastrolia, G. Ossola, T. Reiter, and F. Tramontano, “Scattering AMplitudes from Unitarity-based Reduction Algorithm at the Integrand-level,” *JHEP* **1008** (2010) 080, [arXiv:1006.0710 \[hep-ph\]](#).
- [6] V. Hirschi, R. Frederix, S. Frixione, M. V. Garzelli, F. Maltoni, *et al.*, “Automation of one-loop QCD corrections,” *JHEP* **1105** (2011) 044, [arXiv:1103.0621 \[hep-ph\]](#).
- [7] J. M. Campbell, R. K. Ellis, and C. Williams, “Gluon-Gluon Contributions to W+ W- Production and Higgs Interference Effects,” *JHEP* **1110** (2011) 005, [arXiv:1107.5569 \[hep-ph\]](#).
- [8] G. Bevilacqua, M. Czakon, M. Garzelli, A. van Hameren, A. Kardos, *et al.*, “HELAC-NLO,” *Comput.Phys.Commun.* **184** (2013) 986–997, [arXiv:1110.1499 \[hep-ph\]](#).
- [9] L. Reina and T. Schutzmeier, “Towards W b bbar + j at NLO with an Automatized Approach to One-Loop Computations,” *JHEP* **1209** (2012) 119, [arXiv:1110.4438 \[hep-ph\]](#).
- [10] G. Cullen, N. Greiner, G. Heinrich, G. Luisoni, P. Mastrolia, *et al.*, “Automated One-Loop Calculations with GoSam,” *Eur.Phys.J.* **C72** (2012) 1889, [arXiv:1111.2034 \[hep-ph\]](#).
- [11] **SM AND NLO MULTILEG and SM MC Working Groups** Collaboration, J. Alcaraz Maestre *et al.*, “The SM and NLO Multileg and SM MC Working Groups: Summary Report,” [arXiv:1203.6803 \[hep-ph\]](#).
- [12] R. K. Ellis, Z. Kunszt, K. Melnikov, and G. Zanderighi, “One-loop calculations in quantum field theory: from Feynman diagrams to unitarity cuts,” *Phys.Rept.* **518** (2012) 141–250, [arXiv:1105.4319 \[hep-ph\]](#).

- [13] N. Kauer, “NLO automated tools for QCD and beyond,” *Frascati Phys.Ser.* **54** (2012) 322–335, [arXiv:1202.4608 \[hep-ph\]](#).
- [14] T. Binoth, D. Gonçalves-Netto, D. López-Val, K. Mawatari, T. Plehn, and I. Wigmore, “Automized Squark-Neutralino Production to Next-to-Leading Order,” *Phys.Rev.* **D84** (2011) 075005, [arXiv:1108.1250 \[hep-ph\]](#).
- [15] D. Gonçalves-Netto, D. López-Val, K. Mawatari, T. Plehn, and I. Wigmore, “Sgluon Pair Production to Next-to-Leading Order,” *Phys.Rev.* **D85** (2012) 114024, [arXiv:1203.6358 \[hep-ph\]](#).
- [16] D. Gonçalves-Netto, D. López-Val, K. Mawatari, T. Plehn, and I. Wigmore, “Automated Squark and Gluino Production to Next-to-Leading Order,” *Phys.Rev.* **D87** (2013) 014002, [arXiv:1211.0286 \[hep-ph\]](#).
- [17] D. Gonçalves-Netto, D. López-Val, K. Mawatari, T. Plehn, and I. Wigmore, “MadGolem: automating NLO calculations for New Physics,” [arXiv:1209.2797 \[hep-ph\]](#).
- [18] D. Gonçalves-Netto, D. López-Val, K. Mawatari, T. Plehn, and I. Wigmore, “Madgolem: automated nlo predictions for beyond-the-standard model searches,” in *Proceedings of Science (CHARGED2012) 025*. 2012.
- [19] J. Alwall, P. Demin, S. de Visscher, R. Frederix, M. Herquet, *et al.*, “MadGraph/MadEvent v4: The New Web Generation,” *JHEP* **0709** (2007) 028, [arXiv:0706.2334 \[hep-ph\]](#).
- [20] J. C. Collins, D. E. Soper, and G. F. Sterman, “Factorization of Hard Processes in QCD,” *Adv.Ser.Direct.High Energy Phys.* **5** (1988) 1–91, [arXiv:hep-ph/0409313 \[hep-ph\]](#).
- [21] S. Catani and M. Seymour, “A General algorithm for calculating jet cross-sections in NLO QCD,” *Nucl.Phys.* **B485** (1997) 291–419, [arXiv:hep-ph/9605323 \[hep-ph\]](#).
- [22] S. Catani, S. Dittmaier, M. H. Seymour, and Z. Trocsanyi, “The Dipole formalism for next-to-leading order QCD calculations with massive partons,” *Nucl.Phys.* **B627** (2002) 189–265, [arXiv:hep-ph/0201036 \[hep-ph\]](#).
- [23] R. Frederix, T. Gehrmann, and N. Greiner, “Automation of the Dipole Subtraction Method in MadGraph/MadEvent,” *JHEP* **0809** (2008) 122, [arXiv:0808.2128 \[hep-ph\]](#).
- [24] R. Frederix, T. Gehrmann, and N. Greiner, “Integrated dipoles with MadDipole in the MadGraph framework,” *JHEP* **1006** (2010) 086, [arXiv:1004.2905 \[hep-ph\]](#).
- [25] J. Rosiek, “Complete set of Feynman rules for the MSSM: Erratum,” [arXiv:hep-ph/9511250 \[hep-ph\]](#).
- [26] A. Denner, H. Eck, O. Hahn, and J. Kublbeck, “Compact Feynman rules for Majorana fermions,” *Phys.Lett.* **B291** (1992) 278–280.
- [27] P. Nogueira, “Automatic feynman graph generation,” *J. Comput. Phys.* **105** (April, 1993) 279–289.
- [28] M. L. Mangano, S. J. Parke, and Z. Xu, “Duality and Multi - Gluon Scattering,” *Nucl.Phys.* **B298** (1988) 653.
- [29] M. L. Mangano, “The Color Structure of Gluon Emission,” *Nucl.Phys.* **B309** (1988) 461.
- [30] R. Kleiss and W. J. Stirling, “Spinor Techniques for Calculating p anti-p $\rightarrow W^\pm/Z_0 +$ Jets,” *Nucl.Phys.* **B262** (1985) 235–262.

-
- [31] Z. Xu, D.-H. Zhang, and L. Chang, “Helicity Amplitudes for Multiple Bremsstrahlung in Massless Nonabelian Gauge Theories,” *Nucl.Phys.* **B291** (1987) 392.
- [32] L. J. Dixon, “Calculating scattering amplitudes efficiently,” [arXiv:hep-ph/9601359](#) [hep-ph].
- [33] S. Catani, M. Seymour, and Z. Trocsanyi, “Regularization scheme independence and unitarity in QCD cross-sections,” *Phys.Rev.* **D55** (1997) 6819–6829, [arXiv:hep-ph/9610553](#) [hep-ph].
- [34] Z. Kunszt, A. Signer, and Z. Trocsanyi, “One loop helicity amplitudes for all $2 \rightarrow 2$ processes in QCD and N=1 supersymmetric Yang-Mills theory,” *Nucl.Phys.* **B411** (1994) 397–442, [arXiv:hep-ph/9305239](#) [hep-ph].
- [35] J. Smith and W. van Neerven, “The Difference between n-dimensional regularization and n-dimensional reduction in QCD,” *Eur.Phys.J.* **C40** (2005) 199–203, [arXiv:hep-ph/0411357](#) [hep-ph].
- [36] A. Signer and D. Stockinger, “Using Dimensional Reduction for Hadronic Collisions,” *Nucl.Phys.* **B808** (2009) 88–120, [arXiv:0807.4424](#) [hep-ph].
- [37] I. Antoniadis and E. Floratos, “A STUDY OF A POSSIBLE QUARK - GLUON SYMMETRY IN QCD,” *Nucl.Phys.* **B191** (1981) 217.
- [38] W. Siegel, “Supersymmetric Dimensional Regularization via Dimensional Reduction,” *Phys.Lett.* **B84** (1979) 193.
- [39] A. Signer and D. Stockinger, “Factorization and regularization by dimensional reduction,” *Phys.Lett.* **B626** (2005) 127–138, [arXiv:hep-ph/0508203](#) [hep-ph].
- [40] G. 't Hooft and M. Veltman, “Regularization and Renormalization of Gauge Fields,” *Nucl.Phys.* **B44** (1972) 189–213.
- [41] G. Passarino and M. Veltman, “One Loop Corrections for $e^+ e^-$ Annihilation Into $\mu^+ \mu^-$ in the Weinberg Model,” *Nucl.Phys.* **B160** (1979) 151.
- [42] T. Reiter, “Automated Evaluation of One-Loop Six-Point Processes for the LHC,” [arXiv:0903.0947](#) [hep-ph].
- [43] T. Binoth, J.-P. Guillet, G. Heinrich, E. Pilon, and T. Reiter, “Golem95: A Numerical program to calculate one-loop tensor integrals with up to six external legs,” *Comput.Phys.Commun.* **180** (2009) 2317–2330, [arXiv:0810.0992](#) [hep-ph].
- [44] A. van Hameren, “OneLOop: For the evaluation of one-loop scalar functions,” *Comput.Phys.Commun.* **182** (2011) 2427–2438, [arXiv:1007.4716](#) [hep-ph].
- [45] T. Binoth, J. Guillet, and G. Heinrich, “Reduction formalism for dimensionally regulated one loop N point integrals,” *Nucl.Phys.* **B572** (2000) 361–386, [arXiv:hep-ph/9911342](#) [hep-ph].
- [46] T. Binoth, J. P. Guillet, G. Heinrich, E. Pilon, and C. Schubert, “An Algebraic/numerical formalism for one-loop multi-leg amplitudes,” *JHEP* **0510** (2005) 015, [arXiv:hep-ph/0504267](#) [hep-ph].
- [47] J. C. Collins, F. Wilczek, and A. Zee, “Low-Energy Manifestations of Heavy Particles: Application to the Neutral Current,” *Phys.Rev.* **D18** (1978) 242.
- [48] W. J. Marciano, “Flavor Thresholds and Lambda in the Modified Minimal Subtraction Prescription,” *Phys.Rev.* **D29** (1984) 580.

- [49] P. Nason, S. Dawson, and R. K. Ellis, “The Total Cross-Section for the Production of Heavy Quarks in Hadronic Collisions,” *Nucl.Phys.* **B303** (1988) 607.
- [50] W. Beenakker, R. Hopker, M. Spira, and P. Zerwas, “Squark and gluino production at hadron colliders,” *Nucl.Phys.* **B492** (1997) 51–103, [arXiv:hep-ph/9610490](#) [hep-ph].
- [51] A. Slavnov, “Ward Identities in Gauge Theories,” *Theor.Math.Phys.* **10** (1972) 99–107.
- [52] J. Taylor, “Ward Identities and Charge Renormalization of the Yang-Mills Field,” *Nucl.Phys.* **B33** (1971) 436–444.
- [53] S. P. Martin and M. T. Vaughn, “Regularization dependence of running couplings in softly broken supersymmetry,” *Phys.Lett.* **B318** (1993) 331–337, [arXiv:hep-ph/9308222](#) [hep-ph].
- [54] M. Bohm, H. Spiesberger, and W. Hollik, “On the One Loop Renormalization of the Electroweak Standard Model and Its Application to Leptonic Processes,” *Fortsch.Phys.* **34** (1986) 687–751.
- [55] W. Beenakker, M. Kramer, T. Plehn, M. Spira, and P. Zerwas, “Stop production at hadron colliders,” *Nucl.Phys.* **B515** (1998) 3–14, [arXiv:hep-ph/9710451](#) [hep-ph].
- [56] J. Vermaseren, “New features of FORM,” [arXiv:math-ph/0010025](#) [math-ph].
- [57] G. Cullen, N. Greiner, A. Guffanti, J.-P. Guillet, G. Heinrich, *et al.*, “Recent Progress in the Golem Project,” *Nucl.Phys.Proc.Suppl.* **205-206** (2010) 67–73, [arXiv:1007.3580](#) [hep-ph].
- [58] G. Cullen, M. Koch-Janusz, and T. Reiter, “Spinney: A Form Library for Helicity Spinors,” *Comput.Phys.Commun.* **182** (2011) 2368–2387, [arXiv:1008.0803](#) [hep-ph].
- [59] T. Hahn, “Feynman Diagram Calculations with FeynArts, FormCalc, and LoopTools,” *PoS ACAT2010* (2010) 078, [arXiv:1006.2231](#) [hep-ph].
- [60] D. Maitre and P. Mastrolia, “S@M, a Mathematica Implementation of the Spinor-Helicity Formalism,” *Comput.Phys.Commun.* **179** (2008) 501–574, [arXiv:0710.5559](#) [hep-ph].
- [61] S. P. Martin, “A Supersymmetry primer,” [arXiv:hep-ph/9709356](#) [hep-ph].
- [62] B. C. Allanach, S. Grab, and H. E. Haber, “Supersymmetric Monojets at the Large Hadron Collider,” *JHEP* **1101** (2011) 138, [arXiv:1010.4261](#) [hep-ph].
- [63] B. Allanach, M. Battaglia, G. Blair, M. S. Carena, A. De Roeck, *et al.*, “The Snowmass points and slopes: Benchmarks for SUSY searches,” *Eur.Phys.J.* **C25** (2002) 113–123, [arXiv:hep-ph/0202233](#) [hep-ph].
- [64] D. S. Alves, E. Izaguirre, and J. G. Wacker, “It’s On: Early Interpretations of ATLAS Results in Jets and Missing Energy Searches,” *Phys.Lett.* **B702** (2011) 64–68, [arXiv:1008.0407](#) [hep-ph].
- [65] **ATLAS Collaboration** Collaboration, G. Aad *et al.*, “Search for supersymmetry in pp collisions at $\sqrt{s} = 7$ TeV in final states with missing transverse momentum and b-jets,” *Phys.Lett.* **B701** (2011) 398–416, [arXiv:1103.4344](#) [hep-ex].
- [66] J. Pumplin, D. Stump, J. Huston, H. Lai, P. M. Nadolsky, *et al.*, “New generation of parton distributions with uncertainties from global QCD analysis,” *JHEP* **0207** (2002) 012, [arXiv:hep-ph/0201195](#) [hep-ph].
- [67] S. Frixione, Z. Kunszt, and A. Signer, “Three jet cross-sections to next-to-leading order,” *Nucl.Phys.* **B467** (1996) 399–442, [arXiv:hep-ph/9512328](#) [hep-ph].

- [68] Z. Nagy and Z. Trocsanyi, “Next-to-leading order calculation of four jet observables in electron positron annihilation,” *Phys.Rev.* **D59** (1999) 014020, [arXiv:hep-ph/9806317](#) [hep-ph].
- [69] M. Drees and K.-i. Hikasa, “Scalar top production in $e^+ e^-$ annihilation,” *Phys.Lett.* **B252** (1990) 127–134.
- [70] W. Beenakker, M. Klasen, M. Kramer, T. Plehn, M. Spira, *et al.*, “The Production of charginos / neutralinos and sleptons at hadron colliders,” *Phys.Rev.Lett.* **83** (1999) 3780–3783, [arXiv:hep-ph/9906298](#) [hep-ph].
- [71] R. Frederix, S. Frixione, F. Maltoni, and T. Stelzer, “Automation of next-to-leading order computations in QCD: The FKS subtraction,” *JHEP* **0910** (2009) 003, [arXiv:0908.4272](#) [hep-ph].
- [72] **ATLAS Collaboration** Collaboration, G. Aad *et al.*, “Search for squarks and gluinos with the ATLAS detector in final states with jets and missing transverse momentum using $4.7fb^{-1}$ of $\sqrt{s} = 7$ TeV proton-proton collision data,” [arXiv:1208.0949](#) [hep-ex].
- [73] **CMS Collaboration** Collaboration, S. Chatrchyan *et al.*, “Search for new physics in the multijet and missing transverse momentum final state in proton-proton collisions at $\sqrt{s} = 7$ TeV,” *Phys.Rev.Lett.* **109** (2012) 171803, [arXiv:1207.1898](#) [hep-ex].
- [74] J. Jaeckel, V. V. Khoze, T. Plehn, and P. Richardson, “Travels on the squark-gluino mass plane,” *Phys.Rev.* **D85** (2012) 015015, [arXiv:1109.2072](#) [hep-ph].
- [75] S. AbdusSalam, B. Allanach, H. Dreiner, J. Ellis, U. Ellwanger, *et al.*, “Benchmark Models, Planes, Lines and Points for Future SUSY Searches at the LHC,” *Eur.Phys.J.* **C71** (2011) 1835, [arXiv:1109.3859](#) [hep-ph].
- [76] A. Sommerfeld *Ann. der Phys.* **403** (1931) 257.
- [77] R. S. Chivukula, A. Farzinnia, E. H. Simmons, and R. Foadi, “Production of Massive Color-Octet Vector Bosons at Next-to-Leading Order,” *Phys.Rev.* **D85** (2012) 054005, [arXiv:1111.7261](#) [hep-ph].
- [78] A. Atre, R. S. Chivukula, P. Ittisamai, E. H. Simmons, and J.-H. Yu, “Probing Color Octet Couplings at the Large Hadron Collider,” *Phys.Rev.* **D86** (2012) 054003, [arXiv:1206.1661](#) [hep-ph].
- [79] T. Plehn and T. M. Tait, “Seeking Sgluons,” *J.Phys.G* **G36** (2009) 075001, [arXiv:0810.3919](#) [hep-ph].
- [80] S. Choi, M. Drees, J. Kalinowski, J. Kim, E. Popeno, *et al.*, “Color-Octet Scalars of N=2 Supersymmetry at the LHC,” *Phys.Lett.* **B672** (2009) 246–252, [arXiv:0812.3586](#) [hep-ph].
- [81] J. Alwall, M. Herquet, F. Maltoni, O. Mattelaer, and T. Stelzer, “MadGraph 5 : Going Beyond,” *JHEP* **1106** (2011) 128, [arXiv:1106.0522](#) [hep-ph].
- [82] **ATLAS Collaboration** Collaboration, G. Aad *et al.*, “Search for pair-produced massive coloured scalars in four-jet final states with the ATLAS detector in proton-proton collisions at $\sqrt{s} = 7$ TeV,” [arXiv:1210.4826](#) [hep-ex].
- [83] R. Mahbubani, M. Papucci, G. Perez, J. T. Ruderman, and A. Weiler, “Light non-degenerate squarks at the LHC,” [arXiv:1212.3328](#) [hep-ph].
- [84] D. Gonçalves-Netto, D. López-Val, K. Mawatari, I. Wigmore, and T. Plehn, “Looking for leptogluons,” [arXiv:1303.0845](#) [hep-ph].
- [85] T. Plehn, “Production of supersymmetric particles at high-energy colliders,” [arXiv:hep-ph/9809319](#) [hep-ph].

BIBLIOGRAPHY

Publications

Binoth, Thomas and Gonçalves-Netto, Dorival and López-Val, David and Mawatari, Kentarou and Plehn, Tilman and Wigmore, Ioan. Automized Squark-Neutralino Production to Next-to-Leading Order. In *Phys.Rev. D 84*, 2011.

Gonçalves-Netto, Dorival and López-Val, David and Mawatari, Kentarou and Plehn, Tilman and Wigmore, Ioan. Sgluon Pair Production to Next-to-Leading Order. In *Phys.Rev. D 85*, 2012.

Gonçalves-Netto, Dorival and López-Val, David and Mawatari, Kentarou and Plehn, Tilman and Wigmore, Ioan. MadGolem: automating NLO calculations for New Physics. In *arXiv*, 2012.

Gonçalves-Netto, Dorival and López-Val, David and Mawatari, Kentarou and Plehn, Tilman and Wigmore, Ioan. MadGolem: automated NLO predictions for Beyond-the-Standard Model searches. In *Proceedings of Science (CHARGED2012) 025*, 2012.

Gonçalves-Netto, Dorival and López-Val, David and Mawatari, Kentarou and Plehn, Tilman and Wigmore, Ioan. Automated Squark and Gluino Production to Next-to-Leading Order. In *Phys.Rev. D 87*, 2013.

Gonçalves-Netto, Dorival and López-Val, David and Mawatari, Kentarou and Plehn, Tilman and Wigmore, Ioan. Looking for leptogluons. Approved for publication in *Phys.Rev. D*, 2013.