

June 2023

# Implementing the chirality-flow formalism for tree-level massless QCD in MadGraph

Adam Warnerbring

Department of Physics, Lund University

Master thesis supervised by Malin Sjö Dahl  
and co-supervised by Andrew Lifson and Zenny Wettersten



**LUND**  
UNIVERSITY

# Abstract

The chirality-flow formalism provides a simple but efficient way to rewrite and evaluate Feynman diagrams and has recently been shown to also optimize automated numerical scattering matrix element evaluations in massless tree-level QED. In this thesis we extend the numerical QED model to massless tree-level QCD in the MadGraph5\_aMC@NLO framework. It is shown that a good gauge choice for external gluons lead to significant simplifications of the kinematics, and a set of gauge-specific rules for Feynman diagram generation are presented. The evaluation time for kinematic calculations in matrix element evaluations are compared to standard MadGraph, demonstrating a speed improvement of 1.5-3 times in our implementation, increasing with increasing multiplicity.

## Populärvetenskaplig beskrivning

En partikelfysikers arbete är att beskriva hur naturens minsta beståndsdelar, subatomiska partiklar, interagerar. Detta är väldigt svårt då interaktioner är fundamentalt slumpmässiga. Inte ens den bästa fysikern kan därför säga exakt vad som händer när två partiklar kolliderar, men hon eller han kan istället räkna ut sannolikheten för alla olika händelser. Detta bygger på att fysikern har tillgång till en så bra modell som möjligt av naturen.

Vi vet idag att våra bästa modeller kan förklara en stor del av de interaktioner vi ser i experiment men det finns fortfarande vissa observationer som inte har förklarats än. Detta gör att experimentella fysiker letar efter nya partiklar genom att kollidera till exempel protoner i nära ljushastigheten och sedan studera vilka nya partiklar som skapas. Dessa resultat måste sedan jämföras med de resultat som de teoretiska modellerna förutspår. Eftersom dagens experiment är så stora skapar de en enorm mängd data som leder till att det också måste simuleras stora mängder data från de teoretiska modellerna. Det är därför väldigt lockande att hitta sätt att göra datorberäkningarna snabbare, och det är detta som är projektets fokus.

De processer som är viktigast i dagens experiment förmedlas av den starka kärnkraften, som beskrivs av kvantkromodynamik. Den starka kärnkraften verkar mellan färgladdade kvarkar, med gluoner som kraftförmedlare. Gluoner bär också på färgladdning, vilket gör att de kan växelverka med varandra, till skillnad från exempelvis fotoner. Detta innebär att beräkningar i kvantkromodynamik är komplicerade och tar mycket av datorkraften i simuleringar.

För att beräkna sannolikheten för en given process, till exempel att en uppkvark och en antiuppkvark kolliderar och producerar två gluoner, så måste vi räkna på alla möjliga sätt detta kan hända. Ett exempel är att kvarken först emitterar en gluon och därefter annihileras med antikvarken och då emitterar ytterligare en gluon. Ett annat sätt är att kvarken och antikvarken först annihileras varandra och skickar ut en gluon, varefter gluonen delar upp sig i två nya gluoner.

För att beräkna sannolikheterna ritas vi upp så kallade Feynmandiagram över de olika sätten som en process kan ske. De partiklar som varken är med före eller efter interaktionen kallas virtuella partiklar, och det är de som gör att partiklar kan "känna av" varandra. Tyvärr kan sannolikheterna ändå inte beräknas exakt, då det finns oändligt många diagram vi kan rita. Som tur är så är det möjligt att veta vilka diagram som kommer påverka beräkningen mest, och beror på hur många gånger virtuella partiklar växelverkar med sig själva. I detta projekt arbetar vi bara med diagram där det inte sker några självinteraktioner, eftersom det är dessa diagram som påverkar resultaten allra mest.

För att optimera datorberäkningarna använder vi oss av den nyligen utvecklade kiralitetsflödesformalismen, som bygger på spinor-helicitetsformalismen. Båda formalismerna utnyttjar att masslösa partiklar antingen är vänster- eller högerhänta, något som påverkar hur de växelverkar med andra partiklar. Kiralitetsflödesformalismen använder detta som utgångspunkt för att kunna rita förenklade Feynmandiagram, där de matematiska termerna som används i beräkningen kan läsas av direkt från diagrammen.

Implementeringen av kiralitetsflödesformalismen gör vi i MadGraph, som är ett datorprogram för att simulera partikelinteraktioner. Genom att modifiera programmet får vi det att generera och beräkna de förenklade diagrammen. En av de stora fördelarna med den nya versionen är att programmet lätt kan avgöra om beräkningar av diagram kommer bli noll i förväg. Det visar sig att en stor andel av diagrammen blir noll, vilket gör att vår version av MadGraph gör väsentligen färre beräkningar än originalversionen av MadGraph för samma process.

Förhoppningen är att färre beräkningar leder till kortare körtid för programmet, vilket är precis vad som sker. När de modifierade beräkningarna jämförs med originalversionen av MadGraph utan att ta hänsyn till färgberäkningar så är vår version mellan 1.5-3 gånger snabbare och ger samtidigt lika noggranna resultat. Dessutom presterar det modifierade programmet bättre ju större process som beräknas, vilket är en fördel då stora processer tar mycket mer tid att beräkna än små processer.

Trots att den modifierade versionen av MadGraph gör vissa beräkningar snabbare, så är färgberäkningar ofta det som tar längst tid för stora processer. Vår implementering behandlar färgberäkningar på exakt samma sätt som originalversionen av MadGraph, vilket gör att skillnaden i körtid mellan programmen blir mindre när färgberäkningar inkluderas.

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>Theoretical background</b>	<b>6</b>
2.1	Lorentz transformations . . . . .	6
2.2	The spinor-helicity formalism . . . . .	8
2.3	Chirality flow and its Feynman rules . . . . .	10
2.4	Gauge based diagram removal . . . . .	16
2.5	Gauge-specific Feynman rules . . . . .	18
<b>3</b>	<b>Technical background</b>	<b>22</b>
3.1	MadGraph5_aMC@NLO . . . . .	22
3.2	UFO . . . . .	23
3.3	Diagram generation . . . . .	25
3.4	HELAS and ALOHA . . . . .	26
<b>4</b>	<b>Implementation of chirality flow - CAFE</b>	<b>27</b>
4.1	QED in CAFE . . . . .	27
4.1.1	UFO . . . . .	27
4.1.2	Diagram generation . . . . .	28
4.1.3	Subroutine calls . . . . .	28
4.2	QCD in CAFE . . . . .	29
4.2.1	UFO and diagram generation . . . . .	29
4.2.2	Arrow convention . . . . .	30
4.3	Subroutines - vertex files . . . . .	31
4.3.1	LRV-subroutines . . . . .	31
4.3.2	VVV-subroutines . . . . .	33
4.3.3	VVVV-subroutines . . . . .	34
4.3.4	Reduced subroutines . . . . .	35
4.4	Validation and time tests . . . . .	35

<b>5</b>	<b>Results</b>	<b>36</b>
<b>6</b>	<b>Conclusion and outlook</b>	<b>41</b>

# 1 Introduction

One of the most important tools for particle physics are particle accelerators, where particles can be created and studied in a controlled environment. The data generated from experiments must then be interpreted and compared to theoretical models. Theoretical physicists accomplish this by calculating the probability of different events occurring and comparing these predictions to the rates observed in experiments, using so-called event generators. Through this process, one can evaluate the accuracy of their theoretical models and determine whether modifications are necessary to better describe the underlying nature of the physical world.

The first step in event generation is to calculate the squared matrix elements for the processes studied. For most processes, these calculations are unfeasible with pen and paper, and are therefore done on computers. However, even numerical computations of matrix elements are taking up significant computing power and grow factorially with the number of external particles. Due to the large scale of modern particle physics experiments, fast algorithms for calculating matrix elements are highly desirable.

In 1992 HELAS [1], a library of FORTRAN77 subroutines, was published. This library has all functions needed to calculate any tree-level Feynman diagram in the SM (Standard Model) and a key feature is that it is built to effectively recycle calculations for Feynman diagrams that share the same vertices and propagators. Furthermore, HELAS uses the spinor-helicity formalism [2, 3, 4], which builds on the relation between the Lie algebra of the restricted Lorentz group to two copies of the  $\mathfrak{su}(2)$  Lie algebra. In the spinor-helicity formalism, wavefunctions are represented by left- and right-chiral spinors which both have two components and decouple in the massless limit.

In the last few years the chirality-flow formalism [5, 6, 7] has been developed, which provides further simplifications of the spinor-helicity formalism. The chirality-flow formalism is a graphical approach where spinor inner products can be immediately read off from the Feynman diagram, inspired by the similar color-flow picture [8, 9] for QCD color. The process of going from a standard Feynman diagram to a chirality-flow diagram is simple, and does not require any algebra.

The chirality-flow formalism has been proven to be efficient for pen and paper tree level matrix element calculations, and recently also for automated calculations for massless QED (Quantum Electrodynamics), providing up to an order of magnitude faster matrix element evaluations when compared to helicity amplitude methods [10]. While non-abelian theories such as QCD (Quantum Chromodynamics) contains more complicated kinematic structures than QED, these results lead us to believe that the chirality-flow formalism may optimize automated QCD calculations as well.

The QED chirality-flow implementation, CAFE [10, 11] (Chiral Automated Flow Extension, introduced in [11]), is a modified version of the MadGraph5\_aMC@NLO [12] framework for matrix element calculations and event generation. The standard release of MadGraph

comes with the SM and some models beyond the SM, and the user can define and run calculations for any QFT (Quantum Field Theory), with HELAS-like code for numerical evaluations, inspired by the original HELAS [1] subroutine library, which allows for efficient recycling of calculations between Feynman diagrams.

While MadGraph was not built with the chirality-flow formalism in mind, the CAFE add-on allows us to implement chirality flow in the optimized MadGraph framework, which removes the need to build a specific chirality-flow matrix element calculator from scratch. In this thesis we extend CAFE to include massless tree-level QCD to allow the user to generate and evaluate chirality-flow diagrams with new HELAS-like subroutines. We will explore the performance of our implementation, and especially focus on how the gauge choice for external gluons affects the complexity of matrix element evaluations.

The rest of the thesis is structured as follows: In Section 2 we begin by providing an introduction to the spinor-helicity formalism and chirality flow. We then explore the effect of our gauge choice on automated generation of Feynman diagrams and the efficiency of our algorithm. Section 3 offers an introduction of MadGraph, while Section 4 focuses on CAFE and our QCD implementation. In Section 5, we present our results and compare them to standard MadGraph, and in Section 6 we conclude and give an outlook on the chirality-flow formalism for automated matrix element calculations.

## 2 Theoretical background

In this section the necessary theoretical background for the rest of the thesis is presented. It begins with a brief introduction to the restricted Lorentz group and the spinor-helicity formalism. After this the chirality-flow formalism for massless tree-level QCD is described, with an emphasis on how the choice of gauge vectors affects the analytic complexity towards the end of the section.

### 2.1 Lorentz transformations

The matrix group  $O(1,3)$  consists of the set of transformations  $\Lambda$  that preserve the inner product  $\Lambda^T g \Lambda = g$ , or equivalently  $\Lambda_\mu^\alpha \Lambda_\nu^\beta g_{\alpha\beta} = g_{\mu\nu}$  with

$$g_{\mu\nu} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \quad (2.1)$$

It is called the Lorentz group, with the matrices  $\Lambda$  being the Lorentz transformations and  $g_{\mu\nu}$  the Minkowski metric of the 4-dimensional space-time. Since

$$\det(\Lambda^T g \Lambda) = (\det(\Lambda))^2 \det(g) = \det(g), \quad (2.2)$$

we have that  $\det(\Lambda) = \pm 1$ . The transformations with  $\det(\Lambda) = -1$  correspond to “improper” transformations which include parity or time reversal. Ignoring improper transformations gives the restricted Lorentz group  $SO(1, 3)$ , which consists of spatial rotations and boosts, where the former are the standard 3D rotations acting on the spatial components, and are generated by

$$A_x = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad A_y = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}, \quad A_z = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \quad (2.3)$$

The generators of boosts are given by

$$K_x = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad K_y = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad K_z = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}. \quad (2.4)$$

The six matrices  $\vec{A} = (A_x, A_y, A_z)$  and  $\vec{K} = (K_x, K_y, K_z)$  span the Lie algebra  $\mathfrak{so}(1, 3)$

$$\Lambda = \exp\left(\vec{\theta} \cdot \vec{A} + \vec{\eta} \cdot \vec{K}\right) = \exp\left(-i\left(\vec{\theta} \cdot \vec{L} + \vec{\eta} \cdot \vec{K}'\right)\right), \quad (2.5)$$

where we define the new Hermitian generators  $L_j = iA_j$  and  $K'_j = iK_j$ . With these definitions it is straightforward to show the commutation relations

$$[L_i, L_j] = i\epsilon_{ijk}L_k, \quad [L_i, K'_j] = i\epsilon_{ijk}K'_k, \quad [K'_i, K'_j] = i\epsilon_{ijk}L_k. \quad (2.6)$$

As we expect from the rotation generators  $L$  they are closed under the Lie bracket, and thus rotations form a subgroup, but their generators do not commute with the boost generators  $K'$ . Furthermore the boosts do not form a subgroup. By introducing the new generators

$$N_i^\pm = \frac{L_i \pm iK'_i}{2}, \quad (2.7)$$

which also span the Lie algebra, we find that the two sets of operators are closed under the Lie bracket and commute with each other:

$$[N_i^+, N_j^+] = i\epsilon_{ijk}N_k^+, \quad [N_i^-, N_j^-] = i\epsilon_{ijk}N_k^-, \quad [N_i^+, N_j^-] = 0. \quad (2.8)$$

The commutation relations for both sets are the same as the ones for  $\mathfrak{su}(2)$ . Since  $N_i^+$  and  $N_j^-$  commutes, the Lie algebra of the restricted Lorentz group has exactly the same commutation relations as two copies of the Lie algebra  $\mathfrak{su}(2)$ . However, since we used a linear combination with complex coefficients to express  $N_i^\pm$ , we need to consider the group  $SL(2, \mathbb{C})$ , whose Lie algebra  $\mathfrak{sl}(2, \mathbb{C})$  is the complexified  $\mathfrak{su}(2)$  algebra. Since the irreducible representations of  $SL(2, \mathbb{C})$  can be labeled by a positive half integer or integer  $j$ , the irreducible representations of the Lorentz group can be labeled by two such numbers  $(j_1, j_2)$ .



## 2.2 The spinor-helicity formalism

We will here give a short introduction to the spinor-helicity formalism, which will allow us to define the chirality-flow picture. In Section 2.1 we saw that we can label our representation of the restricted Lorentz group on the form  $(j_1, j_2)$ . The objects  $\tilde{\lambda}_p^{\dot{\alpha}}$  that transform under the  $(1/2, 0)$ -representation are called left-chiral spinors and written with a dotted index  $\dot{\alpha} = 1, 2$ . Similarly objects  $\lambda_{p,\beta}$  that transform under the  $(0, 1/2)$ -representation are called right-chiral spinors and have an undotted index  $\beta = 1, 2$ . A massless incoming fermion or outgoing anti-fermion with momentum  $p$  has 4-component spinors

$$u(p) = \begin{pmatrix} u_L \\ u_R \end{pmatrix} = \begin{pmatrix} \tilde{\lambda}_p^{\dot{\alpha}} \\ \lambda_{p,\beta} \end{pmatrix}, \quad v(p) = \begin{pmatrix} v_L \\ v_R \end{pmatrix} = \begin{pmatrix} \tilde{\lambda}_p^{\dot{\alpha}} \\ \lambda_{p,\beta} \end{pmatrix}, \quad (2.9)$$

in the chiral basis, also called the Weyl basis. The subscripts  $L$  and  $R$  denote left- and right-chiral spinors respectively. The Dirac matrices in the chiral basis are

$$\gamma^\mu = \begin{pmatrix} 0 & \sigma^{\mu,\dot{\alpha}\beta} \\ \bar{\sigma}_{\beta\dot{\alpha}}^\mu & 0 \end{pmatrix} = \begin{pmatrix} 0 & \sqrt{2}\tau^{\mu,\dot{\alpha}\beta} \\ \sqrt{2}\bar{\tau}_{\beta\dot{\alpha}}^\mu & 0 \end{pmatrix}, \quad \gamma^5 \equiv i\gamma^0\gamma^1\gamma^2\gamma^3 = \begin{pmatrix} -1_{2\times 2} & 0 \\ 0 & 1_{2\times 2} \end{pmatrix}, \quad (2.10)$$

where  $\sigma^\mu$  are the Pauli matrices. We introduce the normalized Pauli matrices  $\tau^\mu$ , which will be easier to work with later and satisfy the relations

$$\text{Tr}(\tau^\mu\tau^\nu) = \delta^{\mu\nu}, \quad \text{Tr}(\tau^\mu\bar{\tau}^\nu) = g^{\mu\nu}. \quad (2.11)$$

To get the left- and right-chiral spinors from the 4-component spinor we can use the projection operators  $P_{L/R} = \frac{1}{2}(1 \mp \gamma^5)$ , such that  $P_L u(p) = \begin{pmatrix} \tilde{\lambda}_p^{\dot{\alpha}} \\ 0 \end{pmatrix}$  and  $P_R u(p) = \begin{pmatrix} 0 \\ \lambda_{p,\beta} \end{pmatrix}$  and similarly for  $v(p)$ . The 2-component spinors can be represented as

$$\tilde{\lambda}_p^{\dot{\alpha}} \longleftrightarrow |p] = \frac{1}{\sqrt{p^+}} \begin{pmatrix} p^{\perp*} \\ -p^+ \end{pmatrix}, \quad \lambda_{p,\beta} \longleftrightarrow |p\rangle = \frac{1}{\sqrt{p^+}} \begin{pmatrix} p^+ \\ p^\perp \end{pmatrix}, \quad (2.12)$$

where  $p^\pm, p^\perp$  are the light-cone coordinates

$$p^\pm = p^0 \pm p^3, \quad p^\perp = p^1 + ip^2, \quad p^{\perp*} = p^1 - ip^2. \quad (2.13)$$

We write left-chiral bras and kets with square brackets and right-chiral ones with angled brackets as in eq. (2.12). For outgoing fermions or incoming anti-fermions the spinors are

$$\bar{u}(p) \equiv u^\dagger(p)\gamma^0 = \left( (u_R)^\dagger, (u_L)^\dagger \right) = \left( \tilde{\lambda}_{p,\dot{\beta}}, \lambda_p^\alpha \right) \quad (2.14)$$

$$\bar{v}(p) \equiv v^\dagger(p)\gamma^0 = \left( (v_R)^\dagger, (v_L)^\dagger \right) = \left( \tilde{\lambda}_{p,\dot{\beta}}, \lambda_p^\alpha \right). \quad (2.15)$$

We define the complex conjugation operator to add or remove a dot from the spinor indices

$$(\lambda_p^\alpha)^* = \tilde{\lambda}_p^{\dot{\alpha}}, \quad \left( \tilde{\lambda}_{p,\dot{\beta}} \right)^* = \lambda_{p,\beta}, \quad (2.16)$$

when  $\alpha = \dot{\alpha}$  and  $\beta = \dot{\beta}$ , so that a complex conjugated left-chiral field is a right-chiral field and vice versa. Using this and comparing the equations for incoming and outgoing fermions, we see that spinor indices can be raised and lowered with the Levi-Civita tensor

$$\epsilon^{\dot{\alpha}\dot{\beta}} = \epsilon^{\alpha\beta} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad \epsilon_{\dot{\alpha}\dot{\beta}} = \epsilon_{\alpha\beta} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad (2.17)$$

so that

$$\tilde{\lambda}_p^{\dot{\alpha}} = \epsilon^{\dot{\alpha}\dot{\beta}} \tilde{\lambda}_{p,\dot{\beta}}, \quad \lambda_p^\alpha = \epsilon^{\alpha\beta} \lambda_{p,\beta}, \quad \tilde{\lambda}_{p,\dot{\alpha}} = \epsilon_{\dot{\alpha}\dot{\beta}} \tilde{\lambda}_p^{\dot{\beta}}, \quad \lambda_{p,\alpha} = \epsilon_{\alpha\beta} \lambda_p^\beta. \quad (2.18)$$

The Lorentz invariant inner product between spinors can now be written as

$$[ij] \equiv [i||j] = \tilde{\lambda}_i^{\dot{\alpha}} \tilde{\lambda}_{j,\dot{\alpha}} = \epsilon^{\dot{\alpha}\dot{\beta}} \tilde{\lambda}_{i,\dot{\beta}} \tilde{\lambda}_{j,\dot{\alpha}} = -\epsilon^{\dot{\beta}\dot{\alpha}} \tilde{\lambda}_{i,\dot{\beta}} \tilde{\lambda}_{j,\dot{\alpha}} = -\tilde{\lambda}_j^{\dot{\beta}} \tilde{\lambda}_{i,\dot{\beta}} = -[ji] \quad (2.19)$$

$$\langle ij \rangle \equiv \langle i||j \rangle = \lambda_i^\alpha \lambda_{j,\alpha} = \epsilon_{\alpha\beta} \lambda_i^\alpha \lambda_j^\beta = -\epsilon_{\beta\alpha} \lambda_i^\alpha \lambda_j^\beta = -\lambda_j^\beta \lambda_{i,\beta} = -\langle ji \rangle. \quad (2.20)$$

We note that the inner product is antisymmetric, which follows from the antisymmetry of the Levi-Civita tensor.

Now we have seen spinors and Lorentz invariant inner products, and turn our attention to four vectors, which transform under the  $(1/2, 1/2)$  representation of the Lorentz group. We map our four vectors  $p_\mu$  to Hermitian  $2 \times 2$  matrices

$$p^{\dot{\alpha}\beta} \equiv p_\mu \tau^{\mu,\dot{\alpha}\beta} = \frac{1}{\sqrt{2}} p_\mu \sigma^{\mu,\dot{\alpha}\beta}, \quad \bar{p}_{\alpha\dot{\beta}} \equiv p_\mu \bar{\tau}_{\alpha\dot{\beta}}^\mu = \frac{1}{\sqrt{2}} p_\mu \bar{\sigma}_{\alpha\dot{\beta}}^\mu, \quad (2.21)$$

and define slashed momenta as

$$\not{p} \equiv p_\mu \sigma^\mu = \begin{pmatrix} p^- & -p^{\perp*} \\ -p^\perp & p^+ \end{pmatrix}, \quad \bar{\not{p}} \equiv p_\mu \bar{\sigma}^\mu = \begin{pmatrix} p^+ & p^{\perp*} \\ p^\perp & p^- \end{pmatrix}. \quad (2.22)$$

If the momentum  $p$  is massless, i.e. if  $p^2 = 0$ , it can be written as an outer product of spinors

$$\not{p} = |p\rangle \langle p|, \quad \bar{\not{p}} = |p\rangle [p|. \quad (2.23)$$

If  $p$  is massive it can instead be written as a linear combination of massless momenta<sup>1</sup>  $p_i$ ,  $p = \sum_i c_i p_i$  with  $p_i^2 = 0$ , then  $\not{p}$  and  $\bar{\not{p}}$  can be expressed as a linear combinations of outer products

$$\not{p} = \sum_i c_i |p_i\rangle \langle p_i|, \quad \bar{\not{p}} = \sum_i c_i |p_i\rangle [p_i|. \quad (2.24)$$

If all external particles are massless we can easily find such a linear combination for an internal particle in a Feynman diagram, since it is always possible to rewrite the momenta of any particle in terms of external momenta at tree level.

Lastly, we need to consider polarization vectors of external vector bosons  $\epsilon_\pm(p_i, r)$ , where the sign in the subscript denotes the helicity. The four-vector  $r$  is called the reference

<sup>1</sup>This can also always be done with only two momenta. Since  $p = \frac{1}{2}(p+q) + \frac{1}{2}(p-q)$ , any  $q$  with  $q \cdot p = 0$  and  $q^2 = -p^2$  can be used.

momentum, and is a light-like four-vector that can be chosen arbitrarily as long as it satisfies  $p_i \cdot r \neq 0$ . The outgoing polarization four-vectors can be expressed with Weyl spinors as <sup>2</sup>

$$\epsilon_-^\mu(p_i, r) = \frac{\lambda_i^\alpha \bar{\tau}_{\beta\dot{\alpha}}^\mu \tilde{\lambda}_r^{\dot{\beta}}}{\tilde{\lambda}_{i,\dot{\sigma}} \tilde{\lambda}_r^{\dot{\sigma}}} = \frac{\langle i | \bar{\tau}^\mu | r \rangle}{[ir]}, \quad \epsilon_+^\mu(p_i, r) = \frac{\lambda_r^\alpha \bar{\tau}_{\beta\dot{\alpha}}^\mu \tilde{\lambda}_i^{\dot{\beta}}}{\lambda_r^\sigma \lambda_{i,\sigma}} = \frac{\langle r | \bar{\tau}^\mu | i \rangle}{\langle ri \rangle}, \quad (2.25)$$

and they can also be written using the  $\tau^\mu$  matrices instead of  $\bar{\tau}^\mu$ :

$$\epsilon_-^\mu(p_i, r) = \frac{\tilde{\lambda}_{r,\dot{\alpha}} \tau^{\mu,\dot{\alpha}\beta} \lambda_{i,\beta}}{\tilde{\lambda}_{i,\dot{\sigma}} \tilde{\lambda}_r^{\dot{\sigma}}} = \frac{[r | \tau^\mu | i \rangle}{[ir]}, \quad \epsilon_+^\mu(p_i, r) = \frac{\tilde{\lambda}_{i,\dot{\alpha}} \tau^{\mu,\dot{\alpha}\beta} \lambda_{r,\beta}}{\lambda_r^\sigma \lambda_{i,\sigma}} = \frac{[i | \tau^\mu | r \rangle}{\langle ri \rangle}. \quad (2.26)$$

Using eq. (2.21) we can obtain representations of the polarization vectors with two spinor indices instead of one Lorentz index, and eq. (2.25) becomes

$$\epsilon_-^{\dot{\beta}\alpha}(p_i, r) = \frac{\tilde{\lambda}_r^{\dot{\beta}} \lambda_i^\alpha}{\tilde{\lambda}_{i,\dot{\sigma}} \tilde{\lambda}_r^{\dot{\sigma}}} = \frac{[r] \langle i |}{[ir]}, \quad \epsilon_+^{\dot{\beta}\alpha}(p_i, r) = \frac{\tilde{\lambda}_i^{\dot{\beta}} \lambda_r^\alpha}{\lambda_r^\sigma \lambda_{i,\sigma}} = \frac{[i] \langle r |}{\langle ri \rangle}, \quad (2.27)$$

while eq. (2.26) turns into

$$\bar{\epsilon}_{-,\dot{\beta}\alpha}(p_i, r) = \frac{\lambda_{i,\beta} \tilde{\lambda}_{r,\dot{\alpha}}}{\tilde{\lambda}_{i,\dot{\sigma}} \tilde{\lambda}_r^{\dot{\sigma}}} = \frac{[i] [r]}{[ir]}, \quad \bar{\epsilon}_{+,\dot{\beta}\alpha}(p_i, r) = \frac{\lambda_{r,\beta} \tilde{\lambda}_{i,\dot{\alpha}}}{\lambda_r^\sigma \lambda_{i,\sigma}} = \frac{[r] [i]}{\langle ri \rangle}. \quad (2.28)$$

## 2.3 Chirality flow and its Feynman rules

We will now introduce the concept of chirality flow for massless particles. By using the fact that we can trade a Lorentz index for two spinor indices for four-vectors, as we have seen in Section 2.2, we can write Feynman diagrams where the chiralities are explicitly shown throughout the diagram, which will allow us to see the spinor contractions immediately. Throughout this thesis we use the convention that all particles are outgoing, meaning that we are studying processes  $0 \rightarrow p_1 p_2 p_3 \dots$ .

We say that a particle is right-chiral if it has positive incoming helicity. Hence, an outgoing particle is right-chiral if it has negative helicity and we choose the direction of chirality-flow to be opposite to the fermion flow direction. We represent right-chiral massless fermions and anti-fermions with solid lines in the chirality-flow picture. Similarly, left-chiral particles have positive outgoing helicity and also have chirality flow direction opposite to the fermion flow direction and are represented with dashed lines. We use the convention of having fermion flow and chirality flow in the opposite directions as this allows us to always read diagrams along the chirality-flow direction to write down spinor inner products.

---

<sup>2</sup>Outgoing polarization vectors are typically denoted as  $\epsilon^*$ , but as we are only considering outgoing polarization vectors in this thesis we are omitting the star.

Below we show the representations of massless fermions in spinors, bra-ket notation, with standard Feynman rules, and rightmost in the chirality flow picture

$$\text{Outgoing right-chiral fermion } \lambda_p^\alpha \leftrightarrow \langle p| = \text{---} \circ \text{---} \xrightarrow{p_-} = \text{---} \circ \xleftarrow{p} \text{---} \quad (2.29)$$

$$\text{Outgoing right-chiral anti-fermion } \lambda_{p,\alpha} \leftrightarrow |p\rangle = \text{---} \circ \xleftarrow{p_-} \text{---} = \text{---} \circ \xrightarrow{p} \text{---} \quad (2.30)$$

$$\text{Outgoing left-chiral fermion } \tilde{\lambda}_{p,\dot{\alpha}} \leftrightarrow [p| = \text{---} \circ \xrightarrow{p_+} \text{---} = \text{---} \circ \xleftarrow{p} \text{---} \quad (2.31)$$

$$\text{Outgoing left-chiral anti-fermion } \tilde{\lambda}_p^{\dot{\alpha}} \leftrightarrow |p] = \text{---} \circ \xleftarrow{p_+} \text{---} = \text{---} \circ \xrightarrow{p} \text{---} \quad (2.32)$$

Note that in our convention, we always read chirality flow along the direction of the arrow to find the order of inner products, which allows us to represent spinor inner products graphically as follows:

$$\langle ij \rangle = i \text{---} \xrightarrow{\quad} j = -\langle ji \rangle = -i \text{---} \xleftarrow{\quad} j \quad (2.33)$$

$$[ij] = i \text{---} \xrightarrow{\quad} j = -[ji] = -i \text{---} \xleftarrow{\quad} j \quad (2.34)$$

In addition to fermions we have four-vectors and external vector bosons, which we also need to describe in the chirality-flow formalism. Since four-vectors transform under the  $(1/2, 1/2)$  representation, they carry one dotted and one undotted index. For the graphical representation we define the so called momentum-dot, that takes either a dotted line and turns it into an undotted line, or vice versa

$$p_\mu \sigma^\mu \equiv \not{p} \leftrightarrow \sqrt{2} p^{\dot{\alpha}\beta} = \text{---} \xrightarrow{\dot{\alpha}} \bullet \xrightarrow{\beta} \text{---} , \quad p_\mu \bar{\sigma}^\mu \equiv \bar{\not{p}} \leftrightarrow \sqrt{2} \bar{p}_{\alpha\dot{\beta}} = \text{---} \xrightarrow{\alpha} \bullet \xrightarrow{\dot{\beta}} \text{---} \quad (2.35)$$

If  $p$  is a massless momentum, eq. (2.23) can be used to immediately form spinor contractions with  $|p\rangle \langle p|$  or  $|p\rangle [p|$ , which we can graphically write as

$$\text{---} \xrightarrow{\dot{\alpha}} \bullet \xrightarrow{\beta} \text{---} = \text{---} \xrightarrow{\dot{\alpha}} \overset{p}{\curvearrowright} \beta \quad , \quad \text{---} \xrightarrow{\alpha} \bullet \xrightarrow{\dot{\beta}} \text{---} = \text{---} \xrightarrow{\alpha} \overset{p}{\curvearrowleft} \dot{\beta} \quad (2.36)$$

If  $p$  is not massless we can instead use eq. (2.24) to represent the momentum-dot graphically

as

$$\begin{array}{c} \dot{\alpha} \quad p \quad \beta \\ \dashrightarrow \quad \bullet \quad \rightarrow \end{array} = \sum_i c_i \begin{array}{c} p_i \\ \dashrightarrow \quad \curvearrowright \quad \beta \end{array} , \quad (2.37)$$

which is, as previously mentioned, always possible to do at tree level in terms of external momenta  $p_i$ . While we focus on massless particles in this thesis, massive external particles can be treated similarly as massless ones [6].

Let us now come back to external vector bosons. We remember that we can write polarization vectors with two spinor indices as an outer product between the physical momenta and reference momenta. Since we have a graphical representation of spinors from eqs. (2.29-2.32), we can represent equations (2.25) and (2.26) as

$$\epsilon_{-}^{\dot{\beta}\alpha}(p_i, r) = \frac{|r\rangle \langle i|}{[ir]} = \frac{1}{[ir]} \begin{array}{c} \text{---} \rightarrow \text{---} r \\ \text{---} \leftarrow \text{---} i \end{array} \quad (2.38)$$

$$\epsilon_{+}^{\dot{\beta}\alpha}(p_i, r) = \frac{|i\rangle \langle r|}{\langle ri\rangle} = \frac{1}{\langle ri\rangle} \begin{array}{c} \text{---} \rightarrow \text{---} i \\ \text{---} \leftarrow \text{---} r \end{array} \quad (2.39)$$

$$\bar{\epsilon}_{-, \beta\dot{\alpha}}(p_i, r) = \frac{[i] \langle r|}{[ir]} = \frac{1}{[ir]} \begin{array}{c} \text{---} \leftarrow \text{---} r \\ \text{---} \rightarrow \text{---} i \end{array} \quad (2.40)$$

$$\bar{\epsilon}_{+, \beta\dot{\alpha}}(p_i, r) = \frac{\langle r\rangle [i]}{\langle ri\rangle} = \frac{1}{\langle ri\rangle} \begin{array}{c} \text{---} \leftarrow \text{---} i \\ \text{---} \rightarrow \text{---} r \end{array} , \quad (2.41)$$

keeping in mind that we always read chirality-flow diagrams along the arrow direction. We note that polarization vectors have a similar flow structure to the massless momentum-dots but with the important difference that the reference vector  $r$  can be chosen as an (almost) arbitrary light-like momentum, which we will later show can simplify calculations extensively.

The metric tensor  $g^{\mu\nu}$  is graphically represented as two parallel lines, where one is connecting left-chiral spinors and one is connecting right-chiral spinors

$$g^{\mu\nu} = \begin{array}{c} \text{---} \text{---} \text{---} \\ \text{---} \text{---} \text{---} \end{array} . \quad (2.42)$$

The arrows of the two lines must be in opposite direction to each other, but since the inner products are antisymmetric, swapping both arrow directions does not introduce a sign change due to the two minus signs canceling out. The arrow directions can therefore always be chosen to fit with the rest of the diagram.

With these tools ready, let us go through Feynman rules for the vertices that are relevant for massless QED and QCD and how to transform them to the chirality-flow formalism. We begin with the fermion-photon vertex, which with standard Feynman rules is given by

$$ieQ_f \gamma^\mu = \begin{array}{c} \nearrow \\ \searrow \end{array} \begin{array}{c} \text{---} \\ \text{---} \end{array} \begin{array}{c} \text{---} \mu \\ \text{---} \end{array} , \quad (2.43)$$

where  $e$  is the electromagnetic coupling constant and  $Q_f$  is the charge of the fermion. By explicitly writing out the possible helicity configurations and using eq.(2.10) we can split the vertex into two chirality-flow versions, remembering that we consider all particles as outgoing

$$ieQ_f\sqrt{2}\tau^{\mu,\dot{\alpha}\beta} = \begin{array}{c} + \\ - \end{array} \begin{array}{c} \dot{\alpha} \\ \beta \end{array} \begin{array}{c} \nearrow \\ \searrow \end{array} \begin{array}{c} \text{wavy line } \mu \end{array} = ieQ_f\sqrt{2} \begin{array}{c} \dot{\alpha} \\ \beta \end{array} \begin{array}{c} \dashrightarrow \\ \dashrightarrow \end{array} \begin{array}{c} \text{chirality-flow picture} \end{array} \quad (2.44)$$

$$ieQ_f\sqrt{2}\bar{\tau}_{\alpha\dot{\beta}}^{\mu} = \begin{array}{c} - \\ + \end{array} \begin{array}{c} \alpha \\ \dot{\beta} \end{array} \begin{array}{c} \nearrow \\ \searrow \end{array} \begin{array}{c} \text{wavy line } \mu \end{array} = ieQ_f\sqrt{2} \begin{array}{c} \alpha \\ \dot{\beta} \end{array} \begin{array}{c} \dashrightarrow \\ \dashrightarrow \end{array} \begin{array}{c} \text{chirality-flow picture} \end{array}, \quad (2.45)$$

where the rightmost terms are in the chirality-flow picture. Similarly, the quark-antiquark-gluon vertex in standard Feynman rules is

$$i\frac{g_s}{\sqrt{2}}\gamma^{\mu}t_{ji}^a = \begin{array}{c} i \\ j \end{array} \begin{array}{c} \nearrow \\ \searrow \end{array} \begin{array}{c} \text{gluon line } a, \mu \end{array}, \quad (2.46)$$

where we use the following normalization of the color tensors

$$\text{Tr}(t^a t^b) = \delta^{ab}, \quad if^{abc} = \text{Tr}(t^a [t^b, t^c]). \quad (2.47)$$

Not suprisingly, the quark-antiquark-gluon vertex has the same flow structure as the fermion-photon vertex and is given by

$$ig_s\tau^{\mu,\dot{\alpha}\beta}t_{ji}^a = \begin{array}{c} + \\ - \end{array} \begin{array}{c} i, \dot{\alpha} \\ j, \beta \end{array} \begin{array}{c} \nearrow \\ \searrow \end{array} \begin{array}{c} \text{gluon line } a, \mu \end{array} = ig_s t_{ji}^a \begin{array}{c} \dot{\alpha} \\ \beta \end{array} \begin{array}{c} \dashrightarrow \\ \dashrightarrow \end{array} \begin{array}{c} \text{chirality-flow picture} \end{array} \quad (2.48)$$

$$ig_s\bar{\tau}_{\alpha\dot{\beta}}^{\mu}t_{ji}^a = \begin{array}{c} - \\ + \end{array} \begin{array}{c} i, \alpha \\ j, \dot{\beta} \end{array} \begin{array}{c} \nearrow \\ \searrow \end{array} \begin{array}{c} \text{gluon line } a, \mu \end{array} = ig_s t_{ji}^a \begin{array}{c} \alpha \\ \dot{\beta} \end{array} \begin{array}{c} \dashrightarrow \\ \dashrightarrow \end{array} \begin{array}{c} \text{chirality-flow picture} \end{array}. \quad (2.49)$$

The other two QCD vertices are the three- and four-gluon vertex. The three-gluon vertex has three separate kinematic terms and is given by

$$\begin{array}{c} a_1, \mu_1 \\ p_1 \end{array} \begin{array}{c} a_2, \mu_2 \\ p_2 \end{array} \begin{array}{c} a_3, \mu_3 \\ p_3 \end{array} = i\frac{g_s}{\sqrt{2}}(if^{a_1 a_2 a_3}) [g^{\mu_1 \mu_2} (p_1 - p_2)^{\mu_3} + g^{\mu_2 \mu_3} (p_2 - p_3)^{\mu_1} + g^{\mu_3 \mu_1} (p_3 - p_1)^{\mu_2}]. \quad (2.50)$$

We can use eq. (2.42) to turn  $g^{\mu_1\mu_2}$  into the chirality-flow picture, and identify the term  $(p_1 - p_2)^{\mu_3}$  as a momentum-dot. Hence the three-gluon vertex in the chirality-flow picture is

$$\begin{aligned}
& \begin{array}{c} a_1, \mu_1 \quad a_2, \mu_2 \\ \nearrow \quad \nearrow \\ \text{---} \quad \text{---} \\ \searrow \quad \searrow \\ a_3, \mu_3 \end{array} \quad p_1 \quad p_2 \quad p_3 \\
& = i \frac{g_s}{2} (i f^{a_1 a_2 a_3}) \left( \begin{array}{c} \text{---} \quad \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \quad \text{---} \end{array} \right). \tag{2.51}
\end{aligned}$$

We leave out the arrow directions, but they should again be chosen to match the rest of the diagram. As before, the constraints that the flow must go in on one side of the momentum-dot and come out on the other side, and that the flow must be in opposite directions in the  $g^{\mu_1\mu_2}$  terms, still hold.

The kinematic terms of the four-gluon vertex contain only the metric  $g^{\mu\nu}$ , but the terms are multiplying three different color structures

$$\begin{aligned}
& \begin{array}{c} a_1, \mu_1 \quad a_2, \mu_2 \\ \nearrow \quad \searrow \\ \text{---} \quad \text{---} \\ \searrow \quad \nearrow \\ a_4, \mu_4 \quad a_3, \mu_3 \end{array} \\
& = i \left( \frac{g_s}{\sqrt{2}} \right)^2 \left[ i f^{a_1 a_2 d} i f^{d a_3 a_4} (g^{\mu_1 \mu_3} g^{\mu_2 \mu_4} - g^{\mu_1 \mu_4} g^{\mu_2 \mu_3}) \right. \\
& \quad + i f^{a_1 a_3 d} i f^{d a_2 a_4} (g^{\mu_1 \mu_2} g^{\mu_3 \mu_4} - g^{\mu_1 \mu_4} g^{\mu_3 \mu_2}) \\
& \quad \left. + i f^{a_1 a_4 d} i f^{d a_3 a_2} (g^{\mu_1 \mu_3} g^{\mu_4 \mu_2} - g^{\mu_1 \mu_2} g^{\mu_4 \mu_3}) \right]. \tag{2.52}
\end{aligned}$$

Turning the metric into the chirality-flow picture lets us write the four-gluon vertex as

$$\begin{aligned}
& \begin{array}{c} a_1, \mu_1 \quad a_2, \mu_2 \\ \nearrow \quad \searrow \\ \text{---} \quad \text{---} \\ \searrow \quad \nearrow \\ a_4, \mu_4 \quad a_3, \mu_3 \end{array} \\
& = i \left( \frac{g_s}{\sqrt{2}} \right)^2 \left[ i f^{a_1 a_2 d} i f^{d a_3 a_4} \left( \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \end{array} - \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \end{array} \right) \right. \\
& \quad \left. + i f^{a_1 a_3 d} i f^{d a_2 a_4} \left( \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \end{array} - \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \end{array} \right) \right. \\
& \quad \left. + i f^{a_1 a_4 d} i f^{d a_3 a_2} \left( \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \end{array} - \begin{array}{c} \text{---} \quad \text{---} \\ \text{---} \quad \text{---} \end{array} \right) \right], \tag{2.53}
\end{aligned}$$

where, as before, arrows belonging to the same metric have opposite directions and match with the rest of the diagram.

Let us now lastly look at propagators in the chirality-flow picture. Since the gluon propagator is proportional to the metric it becomes very simple

$$\mu, a \quad \text{---} \quad \nu, b = -i \frac{g_{\mu\nu}}{p^2} \delta^{ab} = -i \frac{\delta^{ab}}{p^2} \text{---}. \tag{2.54}$$

The fermion propagator is given by

$$\frac{ip_\mu\gamma^\mu}{p^2} = \frac{i}{p^2} \begin{pmatrix} 0 & p_\mu\sqrt{2}\tau^\mu \\ p_\mu\sqrt{2}\bar{\tau}^\mu & 0 \end{pmatrix} = \frac{i}{p^2} \begin{pmatrix} 0 & \not{p} \\ \bar{\not{p}} & 0 \end{pmatrix}, \quad (2.55)$$

which just as eqs. (2.44) and (2.45) splits into two chirality flow versions. In equation (2.35) we already have a chirality-flow representation for  $\not{p}$  and  $\bar{\not{p}}$ , and hence the fermion propagator is given by

$$\frac{i}{p^2}\not{p} \rightarrow \frac{i}{p^2}\sqrt{2}p^{\dot{\alpha}\beta} = \frac{i}{p^2} \begin{array}{c} \dot{\alpha} \quad p \quad \beta \\ \text{---} \rightarrow \text{---} \bullet \rightarrow \text{---} \end{array}, \quad (2.56)$$

or

$$\frac{i}{p^2}\bar{\not{p}} \rightarrow \frac{i}{p^2}\sqrt{2}\bar{p}_{\alpha\dot{\beta}} = \frac{i}{p^2} \begin{array}{c} \alpha \quad p \quad \dot{\beta} \\ \text{---} \rightarrow \bullet \text{---} \rightarrow \text{---} \end{array}. \quad (2.57)$$

We now show a simple example of how to use the chirality-flow Feynman rules for the process  $0 \rightarrow u_R\bar{u}_L d_R\bar{d}_L$ . We begin by collecting all coupling constants and factors from vertices and propagators, and then using eqs. (2.48), (2.49) and (2.54) to turn the diagram in to a chirality-flow diagram. Lastly we assign the arrow directions so that they are in opposite directions in the propagator. Performing these steps we get

$$\begin{array}{c} u_R \\ - \\ + \\ \bar{u}_L \end{array} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \begin{array}{c} d_R \\ - \\ + \\ \bar{d}_L \end{array} = \frac{ig_s^2}{p^2} \begin{array}{c} u_R \\ \text{---} \\ \bar{u}_L \end{array} \begin{array}{c} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{array} \begin{array}{c} d_R \\ \text{---} \\ \bar{d}_L \end{array} = \frac{ig_s^2}{p^2} \langle p_u p_d \rangle [p_{\bar{u}} p_{\bar{d}}], \quad (2.58)$$

with  $p = p_u + p_{\bar{u}}$ . In the last line we directly identified the inner products  $\langle p_u p_d \rangle$  and  $[p_{\bar{u}} p_{\bar{d}}]$  by following the flow of the solid and dashed line, respectively.

Let us now consider a slightly more interesting example, calculating a Feynman diagram to the process  $0 \rightarrow u_R\bar{u}_L g_R g_L$ . Using the same strategy of collecting prefactors, then rewriting

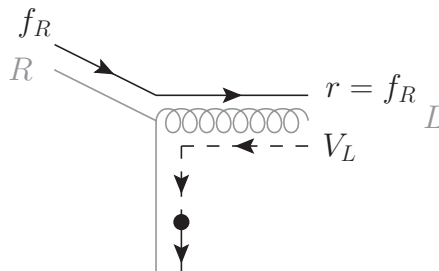




Feynman diagrams one must choose the same reference momenta for each gauge-invariant subset of Feynman diagrams, and for simplicity we limit ourselves to choosing the reference momenta in the same way for all diagrams in the process, although in principle different reference momenta could have been used for different gauge-invariant sets.

Diagram removal arises from our freedom in choosing the reference momenta of external bosons, which lead to vanishing spinor contractions. Hence our goal should be to have the maximum number of inner products between momenta and themselves. A natural choice is therefore to set all left-chiral reference momenta to the physical momentum of a right-chiral particle in the process, and similarly set all right-chiral reference momenta to the same left-chiral physical momentum. With this choice all contractions of two reference momenta of the same chirality will vanish, and additionally the inner product will vanish when contracted with the physical momenta which the reference moment were chosen to be equal to. In our implementation we set the reference momenta of all right-chiral bosons to the momentum of the first left-chiral fermion, and similarly the reference momenta of all left-chiral bosons to the momentum of the first right-chiral fermion. When considering a gluon-only process we instead set the reference momenta to the physical momentum of the first left- and first right-chiral gluon. Although not proven for all processes and phase space points, we believe that this is often the optimal choice of the reference vectors.

Let us now see how this gauge choice affects the quark-antiquark-gluon vertex. In the chirality-flow picture the quark-antiquark-gluon vertex is given by eq. (2.48) or (2.49), and we can immediately see that the vertex will vanish if the reference momentum is equal to the momentum of the fermion it is contracted with. With our gauge choice, this happens if and only if the first left-chiral fermion is connected directly to a right-chiral gluon or the first right-chiral fermion is connected directly to a left-chiral gluon



$$\propto \langle f_{R^T} \rangle = \langle f_R f_R \rangle = 0. \quad (2.62)$$

We can therefore remove all Feynman diagrams with this property from our processes before performing any numerical evaluations.

The other two QCD vertices, the three- and four-gluon vertex, both have three distinct kinematic terms. These terms contain all possible spinor contractions, making the rules for diagram removal for these vertices more complex than those for the quark-antiquark-gluon vertex. However, even when the full Feynman diagram can not be removed, identifying which kinematic terms vanish can yield significant simplifications.

## 2.5 Gauge-specific Feynman rules

The choice of reference vector determines which diagrams will vanish, as we have previously observed. In addition, even for diagrams that do not vanish, some kinematic terms may be zero. To be able to use these simplifications, we must know whether a gluon carries the left- or right-chiral reference momentum. Therefore, to fully utilize our gauge choice, we require a way to consistently keep track of the flow of the reference vectors. To determine the presence of a reference vector in off-shell particles, we systematically examine all chirality combinations of the three- and four-gluon vertex. By doing so, we create a table that maps the incoming gluons to an off-shell gluon  $g_i \in \{g, g_L, g_R, g_{\text{ref}}\}$  that contains the maximal amount of information. This approach allows us to efficiently determine when a reference vector is carried by the propagator, with a table that can be used for any process.

We will call these rules *gauge-specific Feynman rules*, aiming to retain as much information as possible about the gluons at all stages of evaluation. Since these rules will only be valid for this specific gauge choice, we here present them for the gauge choice presented in Section 2.4. Because the reference momenta are chosen based on the first right-chiral and the first left-chiral gluon, these two gluons will carry both the left- and right-chiral reference momenta, and hence have unique properties. To distinguish them, we will refer to them as reference gluons. We label these reference gluons  $g_{\text{ref}}$ , those carrying the right-chiral reference momentum  $g_L$ , those carrying the left-chiral reference momentum  $g_R$ , and those carrying neither reference momentum  $g$ .

In the MadGraph standard, each individual particle state is evaluated explicitly, to recycle calculations that are shared between diagrams, and the last evaluation calculates the amplitude by contracting the innermost propagators of the given diagram. With the gauge-specific Feynman rules, it is therefore important to distinguish vertices which create an off-shell particle from the final vertices, where the latter creates amplitudes in MadGraph. We show this with the following example: Consider two left-chiral gluons  $g_L$  combining to a propagator. From the flow

$$\begin{array}{c}
 L_1 \quad r \\
 \curvearrowright \\
 L_2 \quad r
 \end{array}
 + 
 \begin{array}{c}
 L_1 \quad r \\
 \curvearrowright \\
 \bullet
 \end{array}
 + 
 \begin{array}{c}
 \bullet \\
 \curvearrowright \\
 L_2 \quad r
 \end{array}, \quad (2.63)$$

we see that the first term vanishes, since it is proportional to  $\langle rr \rangle = 0$ , and in the other two terms the solid line going to the propagator is the right-chiral reference momentum. The propagator therefore behaves as a  $g_L$  particle for the purpose of simplifying vertices, and we only need to consider two out of three kinematic terms in our calculations. If the vertex instead was a final vertex with three  $g_L$  particles it would clearly vanish as all contractions

between solid lines vanish.

Consider now a 4-gluon vertex with two  $g_L$  gluons carrying the right-chiral reference momentum  $r$ , without knowledge of the two other gluon chiralities. Only the two terms where the  $g_L$  particles are not contracted survive

$$\begin{array}{c}
 \begin{array}{cc}
 \begin{array}{c} r \\ \downarrow \\ L_1 \\ \uparrow \\ R_3 \\ L_3 \end{array} & \begin{array}{c} L_2 \\ \downarrow \\ r \\ \uparrow \\ L_2 \\ L_3 \end{array} \\
 \end{array} & = |r\rangle [L_2 | \langle R_3 r | [L_1 L_3], & \begin{array}{c} \begin{array}{cc} r & r \\ \swarrow & \searrow \\ L_1 & L_2 \\ \nwarrow & \nearrow \\ R_3 & L_3 \end{array} \\
 \end{array} & = |r\rangle [L_1 | \langle R_3 r | [L_2 L_3],
 \end{array} \tag{2.64}$$

as

$$\begin{array}{c}
 \begin{array}{cc}
 \begin{array}{c} r \\ \longrightarrow \\ L_1 \end{array} & \begin{array}{c} r \\ \longleftarrow \\ L_2 \end{array} \\
 \begin{array}{c} \longrightarrow \\ \longleftarrow \\ L_3 \end{array} & \begin{array}{c} \longrightarrow \\ \longleftarrow \\ L_3 \end{array}
 \end{array} \propto \langle rr \rangle = 0.
 \end{array} \tag{2.65}$$

If this vertex results in an off-shell gluon, we see that both terms in eq. (2.64) carry the right-chiral reference momentum, and hence the resulting gluon is a  $g_L$  gluon. If this vertex instead is the last vertex drawn, we can still use the fact that the contraction between two  $g_L$  particles vanishes to simplify the calculation of the amplitude. Furthermore, if the fourth gluon carries the right-chiral reference momentum all terms are proportional to  $\langle rr \rangle = 0$ , and the full vertex vanishes.

A powerful tool to simplify the four-gluon vertex even further is the Schouten identity [13], which states that

$$\begin{aligned}
 \langle ij \rangle \langle kl \rangle &= \langle ik \rangle \langle jl \rangle + \langle il \rangle \langle kj \rangle \\
 [ij] [kl] &= [ik] [jl] + [il] [kj],
 \end{aligned} \tag{2.66}$$

and follows from the fact that three two component spinors must be linearly dependent, together with the antisymmetry of the inner products. These identities written in the chirality flow picture are given by

$$\begin{array}{c}
 \begin{array}{cc}
 \begin{array}{c} i \\ \longrightarrow \\ j \end{array} & \begin{array}{c} l \\ \longleftarrow \\ k \end{array} \\
 \begin{array}{c} i \\ \dashrightarrow \\ j \end{array} & \begin{array}{c} l \\ \dashleftarrow \\ k \end{array}
 \end{array} = \begin{array}{c} \begin{array}{cc} \begin{array}{c} i \\ \searrow \\ l \end{array} & \begin{array}{c} j \\ \swarrow \\ k \end{array} \\ \begin{array}{c} l \\ \swarrow \\ i \end{array} & \begin{array}{c} k \\ \searrow \\ j \end{array} \end{array} + \begin{array}{c} \begin{array}{c} i \\ \downarrow \\ l \end{array} & \begin{array}{c} j \\ \uparrow \\ k \end{array} \\ \begin{array}{c} i \\ \downarrow \\ l \end{array} & \begin{array}{c} j \\ \uparrow \\ k \end{array} \end{array} .
 \end{array} \tag{2.67}$$

Hence we see that we can replace the flow between two pairs of spinors with the other two possible pairs of flows. For example, we can rewrite the  $g_L g_L g_R g_R$  vertex with the help of the Schouten identities, where we can use that solid lines between the  $g_L$  pair and dashed lines between the  $g_R$  pair vanishes. By first using the Schouten identity on the dashed lines and then on the solid lines we obtain

$$\begin{aligned}
& \text{Diagram 1} = \text{Diagram 2} + \text{Diagram 3} = \text{Diagram 4} \\
& \text{Diagram 2} = \text{Diagram 5} + \text{Diagram 6} = \text{Diagram 7} + \text{Diagram 8}
\end{aligned}
\tag{2.68}$$

The kinematic structure contracting the two  $g_L$  and  $g_R$  with each other is obviously zero, and with the use of the Schouten identity we have shown that the two other kinematic structures are identical.

Another interesting consequence of our gauge choice is that any three-gluon vertex with the two external reference gluons must vanish. Denoting the physical momenta of the two gluons  $p_1$  and  $p_2$  respectively, the reference momenta are  $r_1 = p_2$  and  $r_2 = p_1$ . The chirality-flow vertex is given by

$$\text{Diagram 1} = \text{Diagram 2} + \text{Diagram 3} + \text{Diagram 4}
\tag{2.69}$$

and it is immediately clear that the first term must vanish. By using momentum conservation ( $p_1 + p_2 + p_3 = 0$ ) we can rewrite the momentum-dots as  $p_3 - p_1 = -2p_1 - p_2$  and  $p_2 - p_3 = 2p_2 + p_1$ . Expanding the momentum-dots with eq. (2.37) for the second term gives

$$-[p_1 | (2|p_1 \langle p_1 | + |p_2 \rangle \langle p_2 |) |p_2 \rangle = 0,
\tag{2.70}$$

and similarly for the third term

$$[p_1 | (2|p_2 \langle p_2 | + |p_1 \rangle \langle p_1 |) |p_2 \rangle = 0.
\tag{2.71}$$

Vertex	Remaining terms	Vertex	Remaining terms
$gg_{\text{ref}}g_{\text{ref}}$	2	$gg_{\text{ref}}g_L$	2
$gg_{\text{ref}}g_R$	2	$g_{\text{ref}}g_{\text{ref}}g_L$	0
$g_{\text{ref}}g_Lg_L$	0	$g_{\text{ref}}g_Rg_R$	0
$g_{\text{ref}}g_Lg_R$	1	$g_{\text{ref}}g_{\text{ref}}g_R$	0
$gg_Lg_L$	2	$gg_Rg_R$	2
$g_Lg_Lg_R$	2	$g_Lg_Rg_R$	2
$g_Lg_Lg_L$	0	$g_Rg_Rg_R$	0
$ggg_{\text{ref}}g_L$	2	$ggg_{\text{ref}}g_R$	2
$ggg_Lg_L$	2	$ggg_Rg_R$	2
$gg_{\text{ref}}g_Lg_R$	1	$gg_{\text{ref}}g_{\text{ref}}g_L$	0
$gg_{\text{ref}}g_Lg_L$	0	$gg_{\text{ref}}g_Rg_R$	0
$gg_Lg_Lg_R$	2	$gg_Lg_Rg_R$	2
$gg_Lg_Rg_R$	2	$g_{\text{ref}}g_{\text{ref}}g_Lg_L$	0
$g_{\text{ref}}g_{\text{ref}}g_Rg_R$	0	$g_{\text{ref}}g_Lg_Lg_L$	0
$g_{\text{ref}}g_Rg_Rg_R$	0	$g_{\text{ref}}g_Lg_Lg_R$	0
$g_{\text{ref}}g_Lg_Rg_R$	0	$g_Lg_Lg_Rg_R$	1
$g_Lg_Lg_Lg_L$	0	$g_Rg_Rg_Rg_R$	0

Table 1: Final vertices that have one or more vanishing terms. The columns "Remaining terms" shows the number of non-vanishing kinematic terms out of the three unique ones.

Consequently, all three terms of a three-gluon vertex vanish if two external reference gluons are part of the vertex, and diagrams containing these vertices can be removed. Note that this result only applies to on-shell reference gluons, as the momentum-dots in general do not vanish for off-shell reference gluons.

Using the results presented so far, the gauge-specific Feynman rules can be derived. By systematically going through all possible combinations of gluons, Table 1 and 2 have been obtained, containing the gauge-specific Feynman rules when we have one left- and one right-chiral reference gluon. The rules are derived by searching for vanishing contractions and applying the Schouten identities to find terms that are identical to each other or vanishing.

Incoming	Propagator	Remaining terms	Incoming	Propagator	Remaining terms
$gg$	$g$	3	$gg_{\text{ref}}g_{\text{ref}}$	$g_{\text{ref}}$	1
$ggL$	$g$	3	$gg_{\text{ref}}gL$	$gL$	2
$ggR$	$g$	3	$gg_{\text{ref}}gR$	$gR$	2
$gg_{\text{ref}}$	$g$	3	$ggLgL$	$gL$	2
$g_{\text{ref}}g_{\text{ref}}$	$g_{\text{ref}}$	2	$ggLgR$	$g$	3
$g_{\text{ref}}gL$	$gL$	2	$ggRgR$	$gR$	2
$g_{\text{ref}}gR$	$gR$	2	$g_{\text{ref}}g_{\text{ref}}gL$	X	0
$gLgL$	$gL$	2	$g_{\text{ref}}gLgL$	X	0
$gLgR$	$g$	3	$g_{\text{ref}}gLgR$	$g_{\text{ref}}$	1
$gRgR$	$gR$	2	$g_{\text{ref}}gRgR$	X	0
$ggg$	$g$	3	$gLgLgL$	X	0
$ggg_{\text{ref}}$	$g$	3	$gLgLgR$	$gL$	2
$gggL$	$g$	3	$gLgRgR$	$gR$	2
$gggR$	$g$	3			

Table 2: Gauge-specific Feynman rules for vertices returning an off-shell particle. The columns "Remaining terms" shows the number of non-vanishing kinematic terms out of the three unique ones. Propagators with no remaining terms can be ignored when generating diagrams, since diagrams containing them always vanish.

### 3 Technical background

We are now equipped with the necessary theoretical background of the chirality-flow formalism, and in this and the next section we go through the tools used to numerically evaluate matrix elements for any tree-level QCD process using chirality flow. Our QCD implementation is incorporated in CAFE - the Chiral Automated Flow Extension [11], which in turn is a modified version of the framework MadGraph5\_aMC@NLO [12]. It is therefore natural to first give an overview of the different parts of MadGraph, and then present the modifications that are made in CAFE as well as the extension to QCD done in this project.

#### 3.1 MadGraph5\_aMC@NLO

While MadGraph has many features, in this project we focus on matrix element calculations. The strength of MadGraph is that it can take any quantum field theory, specified by its Lagrangian (particles, vertices, couplings), and use this in a model independent framework to evaluate matrix elements for all possible processes at LO and NLO. We here give a brief introduction to the whole workflow, before giving more details to each part. In Figure 1 an outline of the steps that MadGraph takes between the input and result is shown.

The model that the program uses is specified by UFO-files, containing information such as the particles, vertices and parameters of the model. Feynman diagrams are generated by recursively combining the particles, starting from the external ones, according to the rules specified in the UFO model. Once the diagrams have been generated, MadGraph uses these as a basis for generating a program to evaluate the matrix element for the considered process. MadGraph itself is a Python program, but the generated code for numerical evaluations is written in FORTRAN77. This code is based on the HELAS library subroutines and shares the same structure in the sense that it recycles calculations between Feynman diagrams of the same process, something we will discuss in more detail later in this section.

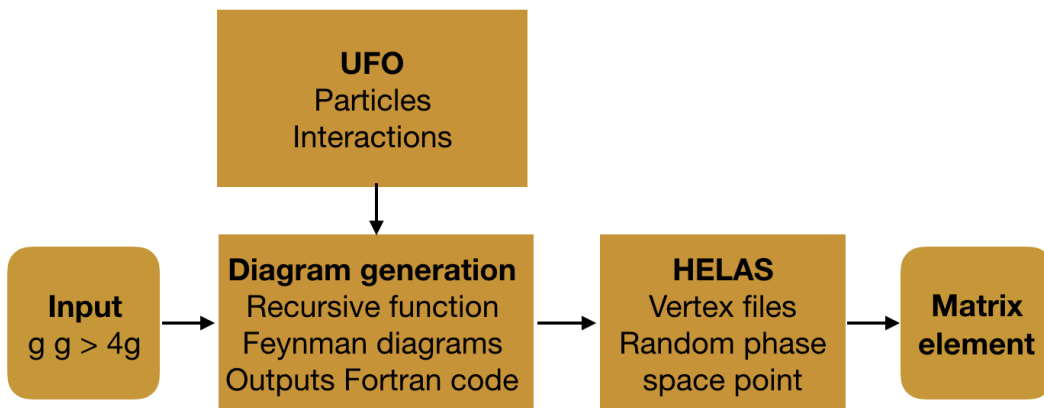


Figure 1: Sketch over how MadGraph goes from user input to a numerical matrix element.

### 3.2 UFO

The Universal FeynRules Output (UFO) [14] is a Python module in which QFTs can be defined in a generator-independent way, without any assumptions on the conventions used by the event generator the model will eventually be used in. The official MadGraph release comes with some UFO models already installed such as the SM and the minimally supersymmetric SM, and allows easy downloads of other models directly through the program. In this project we implement a chirality-flow version of the SM, and use the SM from the official MadGraph release for comparison and benchmarking.

A UFO model is organized into several different Python files containing all information that the event generator needs about the model. There are three files that need significant modification for the chirality-flow implementation: *particles.py*, *vertices.py* and *lorentz.py*. The file *particles.py* contains all particles within the model together with their mass, width and quantum numbers, as well as some identifiers such as their PDG code. As an example, the code snippet below shows part of the definition of a gluon in the standard model.



```

g = Particle(pdg_code = 21,
             spin = 3,
             color = 8,
             mass = Param.ZERO,
             width = Param.ZERO,
             charge = 0,
             GhostNumber = 0,
             LeptonNumber = 0,
             Y = 0,
             ...)

```

Next we show how a quark-antiquark-gluon and the three-gluon vertex is represented in *vertices.py*:

```

V_135 = Vertex(name = 'V_135',
               particles = [ P.u__tilde__, P.u, P.g ],
               color = [ 'T(3,2,1)' ],
               lorentz = [ L.FFV1 ],
               couplings = {(0,0):C.GC_11})

```

```

V_36 = Vertex(name = 'V_36',
              particles = [ P.g, P.g, P.g ],
              color = [ 'f(1,2,3)' ],
              lorentz = [ L.VVV1 ],
              couplings = {(0,0):C.GC_10})

```

Here *P.u\_\_tilde\_\_* is the up antiquark and *P.u* and *P.g* is the up quark and gluon respectively. The color structure of the vertex is described by its color tensor, where  $T(3,2,1)$  represents the fundamental representation matrix  $(T^{a_3})_{i_2}^{j_1}$  and  $f(1,2,3)$  the structure constants  $f^{a_1 a_2 a_3}$ . The line under color specifies which Lorentz structure the vertex has, which is defined in *lorentz.py*.

The code below is a snippet from this file, showing the definition of the Lorentz structures appearing in the vertices above.

```

FFV1 = Lorentz(name = 'FFV1',
               spins = [ 2, 2, 3 ],
               structure = 'Gamma(3,2,1)')

```

```

VVV1 = Lorentz(name = 'VVV1',
               spins = [ 3, 3, 3 ],
               structure = 'P(3,1)*Metric(1,2) - P(3,2)*Metric(1,2) - P(2,1)*Metric(1,3)
+ P(2,3)*Metric(1,3) + P(1,2)*Metric(2,3) - P(1,3)*Metric(2,3)')

```

In the array `spins`, the particle spins are given in the  $2j + 1$  convention, and the structure is expressed in elementary Lorentz structures, where `Gamma(3,2,1)` represents  $(\gamma^{\mu_3})_{i_2 i_1}$ , `Metric(1,2)` the Minkowski metric  $g^{\mu_1 \mu_2}$  and `P(i,N)` the  $\mu_i$  component of the momenta of the N-th particle:  $p_N^{\mu_i}$ . For example, the `VVV1` structure will be

$$(p_1 - p_2)^{\mu_3} g^{\mu_1 \mu_2} + (p_2 - p_3)^{\mu_1} g^{\mu_2 \mu_3} + (p_3 - p_1)^{\mu_2} g^{\mu_3 \mu_1}, \quad (3.72)$$

and multiplies  $f^{a_1 a_2 a_3}$  in the triple gluon vertex, just as we saw in eq. (2.50).

In general, many vertices use the same Lorentz structure, which means that the kinematics implemented are the same for all vertices sharing the same structure. The four-gluon vertex is special, since it multiplies three different color factors. This is solved by simply having one Lorentz structure for each color factor, as shown below,

```
V_58 = Vertex(name = 'V_58',
  particles = [ P.g, P.g, P.g, P.g ],
  color = [ 'f(-1,1,2)*f(3,4,-1)', 'f(-1,1,3)*f(2,4,-1)', 'f(-1,1,4)*f(2,3,-1)' ],
  lorentz = [ L.VVVV1, L.VVVV3, L.VVVV4 ],
  couplings = {(1,1):C.GC_12, (0,0):C.GC_12, (2,2):C.GC_12})
```

with the Lorentz structures defined as

```
VVVV1 = Lorentz(name = 'VVVV1',
  spins = [ 3, 3, 3, 3 ],
  structure = 'Metric(1,4)*Metric(2,3) - Metric(1,3)*Metric(2,4)')

VVVV3 = Lorentz(name = 'VVVV3',
  spins = [ 3, 3, 3, 3 ],
  structure = 'Metric(1,4)*Metric(2,3) - Metric(1,2)*Metric(3,4)')

VVVV4 = Lorentz(name = 'VVVV4',
  spins = [ 3, 3, 3, 3 ],
  structure = 'Metric(1,3)*Metric(2,4) - Metric(1,2)*Metric(3,4)')
```

### 3.3 Diagram generation

From the vertices in the UFO model, MadGraph generates Feynman diagrams that are later evaluated numerically. Initially, we only know the external particles, which needs to be combined into correct Feynman diagrams. MadGraph does so in a recursive way, which has two benefits as compared to naively generating all topologies individually. First of all, MadGraph's algorithm is effective at generating diagrams since it quickly throws away combinations that will not generate a complete diagram. The second, and even larger benefit, is that parts of diagrams that are shared between diagrams can be calculated once

and reused for all the other diagrams where this subdiagram is contained. This means that each propagator is only calculated one time, and then recycled optimally. Hence, the number of calculations of off-shell particles grows slower than the number of amplitudes, since the amplitudes are unique for each diagram.

Loosely described, the algorithm MadGraph uses for diagram generation is as follows. From the UFO-model, two lists are created. The first list contains all combinations of particles that can combine in a vertex, for example  $\bar{u}ug$  and  $ggg$ . The other list contains combinations which can result in an off-shell particle, for example the combination  $\bar{u}u$  can result in an off-shell gluon  $g$ . MadGraph uses these lists to combine the external particles and create off-shell particles or final vertices. The newly created off-shell particles are then combined again recursively, until all diagrams containing all external particles are generated. For a detailed explanation of the algorithm, see [15].

After diagram generation, MadGraph uses the diagrams to build a program that can evaluate the matrix element numerically. It does so by using the information about the particles and vertices in the UFO model, and using vertex subroutines corresponding to the different Lorentz structures. These subroutines, which will be discussed more in Section 3.4, are used to calculate wavefunctions or amplitudes for each unique vertex and external leg of all the diagrams generated.

### 3.4 HELAS and ALOHA

HELicity Amplitude Subroutines [1], or HELAS for short, is a library of FORTRAN77 subroutines, from which the helicity amplitudes for any SM tree-level Feynman diagram can be calculated by a series of subroutine calls. This allows for numerical matrix element calculations that are efficient and easily automated.

We can divide the subroutines into three distinct groups depending on the number of wavefunctions in the input and output: external particles, propagators and amplitudes. The subroutines for external particles are zero-to-one, taking the particles momenta, helicity, mass and with, and returning the wavefunction of the particle. For the propagators, which are many-to-one, the momenta and wavefunctions of the incoming particles are input, and the wavefunction and momentum of the outgoing propagating particle is returned. Finally, the amplitude subroutines used in the last vertex of a Feynman diagram are many-to-zero, taking the wavefunctions and momenta as arguments and outputting the amplitude for the given diagram. Since most interesting processes have more than one Feynman diagram, one can easily see that some parts of the calculation can be reused. For example, the external subroutines only need to be called once for each particle, since they are the same for each Feynman diagram of the process. Similarly, any propagator only needs to be calculated once; if it appears a second time it has already been calculated before and can simply be recycled. The only subroutines that can never be recycled are the amplitude calculations, since if two calculations were the same they would correspond to the same Feynman diagram.

The original HELAS library only contains subroutines for the SM, and is therefore insufficient for beyond SM event generation. The solution for this used in MadGraph is ALOHA (Automatic Libraries Of Helicity Amplitudes) [16], which can automatically create HELAS-like libraries for any quantum field theory, hence generalizing the HELAS structure. ALOHA is made to be completely QFT agnostic, and it can take any UFO model and output the HELAS-like routines needed for numerical calculations.

## 4 Implementation of chirality flow - CAFE

In this section, we will introduce the Chiral Automized Flow Extension - CAFE [11], a numerical implementation of chirality flow for massless tree-level QED, which is extended to massless tree-level QCD in this project. To provide context for our QCD implementation, we will first review the key features of the existing QED implementation. We will then proceed to describe the specific features and modifications used in our implementation of QCD.

CAFE is designed to optimize the kinematics of the SM, and there are a few significant changes compared to MadGraph. For example, while MadGraph works with  $4 \times 4$  Dirac matrices, CAFE utilizes  $2 \times 2$  Pauli matrices as these are used in the chirality-flow formalism. However, where possible, we strive to keep the code as similar as possible to standard MadGraph in our implementation to ensure that our performance comparisons are as fair as possible and to show the benefit of using the chirality-flow formalism.

### 4.1 QED in CAFE

#### 4.1.1 UFO

The chirality-flow QED UFO model is the restriction of the SM to massless QED, with the chirality of the particles explicitly defined. The model contains left and right chiral electrons, muons and photons, as well as a non-chiral photon that is used for all internal photons since they do not carry a left- or right-chiral reference momenta.

The model contains all allowed combinations of vertices, namely  $e_{R/L}^+ e_{L/R}^- \gamma$  and  $\mu_{R/L}^+ \mu_{L/R}^- \gamma$ , where the  $e_R^+ e_L^- \gamma$  and  $\mu_R^+ \mu_L^- \gamma$  vertices use the Lorentz structure given in eq. (2.44), which is called **RLV1**, while  $e_L^+ e_R^- \gamma$  and  $\mu_L^+ \mu_R^- \gamma$  use the **LRV1** structure given by eq. (2.45). In naming the structures, the convention used is that **L** (**R**) represents a left-chiral (right-chiral) fermion, **V** represents a non-chiral gauge boson, **VL** (**VR**) represents a left-chiral (right-chiral) vector boson. In addition to this two other structures structures, **LLV1** and **RRV1**, are defined to treat momentum-dots to make sure that they are compatible with MadGraph's diagram generation.

In addition to the UFO model, the subroutines for external wavefunctions in chirality flow

have been added to the ALOHA template files. For the fermions, these are LXXXXX and RXXXXX which are calculated using eqs. (2.12) and (2.16) and for bosons the subroutines VLXXXX and VRXXXX are implemented using eq. (2.27). The wavefunction for external bosons depend on the choice of the reference momentum, which is set at diagram generation in CAFE, and is usually set to be physical momenta of the external particles. The subroutines also check for singularities and use alternative expressions if needed. In the QED implementation, vector bosons are described in terms of spinors, but we will use a matrix representation in the QCD implementation.

### 4.1.2 Diagram generation

Diagram generation in CAFE follows the same algorithm as MadGraph, described in Section 3.3, with one major change in how fermion propagators are treated. If we for example have a right-chiral fermion  $f_R$  and a photon  $\gamma$  emitting an off-shell particle, MadGraph will output a left-chiral fermion  $f_L$  as the propagator, since in the UFO models the vertices are on the form  $f_{R/L}^+ f_{L/R}^- \gamma$ . However, in chirality flow, the momentum dot would flip the chirality of the propagator back to being right-chiral. In CAFE this chirality flip is included in diagram generation for fermion propagators, which requires the LLV\_1 and RRV\_1 subroutines mentioned in Section 4.1.1

### 4.1.3 Subroutine calls

The matrix elements are calculated from a sequence of subroutine calls, and the lists of calls are created in MadGraph from the generated diagrams. In addition to the calls for the external particles, the subroutine files are modified by CAFE to calculate propagators and amplitudes in the chirality-flow picture.

In standard MadGraph, by default all possible helicity configurations are summed over for a given process. However, since CAFE has explicitly chiral particles, in the massless case there is exactly one non-vanishing helicity configuration, which is the one matching the chiralities of the external particles. Hence, for helicity summation, one subprocess needs to be created for each chirality configuration. To make this practically useful, multiparticles are defined that contain the chiral particles, for example  $e^- = e_L^- + e_R^-$ , so that (mostly) standard MadGraph syntax can be used to create and run subprocesses for each non-vanishing chirality combination. When multiparticles are used, a subprocess is created for each chirality configuration, leading to a  $O(2^n)$  scaling in the number of subprocesses, where  $n$  is the number of external particles. This inevitably introduces overhead in CAFE that does not appear for MadGraph, since each chirality configuration has to be run as a distinct process.

## 4.2 QCD in CAFE

### 4.2.1 UFO and diagram generation

In order to study the fully general case, where reference vectors can be chosen freely, as well as an optimised implementation using the gauge-specific Feynman rules presented in Section 2.5, two UFO models are created. In the generic model, the reference vectors for gluons can be chosen freely at the cost of performance, while the model employing gauge-specific Feynman rules has a fixed gauge choice and is focused on evaluation time optimization by the use of simplified vertices.

In the generic model we define chiral up and down quarks  $u_L, u_R, d_L, d_R$  as well as their antiparticles, together with the chiral gluons  $g_L$  and  $g_R$  and the non-chiral propagator  $g$ . For the antifermion-fermion-gluon vertices we define all allowed chirality configurations:  $\bar{u}_{R/L}u_{L/R}g_i$  and  $\bar{d}_{R/L}d_{L/R}g_i$ , where  $g_i \in \{g, g_L, g_R\}$ . Similarly, for the three- and four-gluon vertex we make one vertex for each chirality configurations of the gluons, i.e.  $g_i g_j g_k$  and  $g_i g_j g_k g_l$  where  $g_i, g_j, g_k, g_l \in \{g, g_L, g_R\}$ , independently of each other.

In the diagram generation we then make sure that only  $g$  is used as a propagator to avoid double counting of diagrams in the generic model. The three gluon vertex has the VVV1 Lorentz structure, while the four-gluon vertex have the VVVV1, VVVV3 and VVVV4 structures, which we will describe later in this section.

In our gauge-specific QCD UFO model, we also define chiral up and down quarks and their antiquarks. In addition to the non-chiral gluon  $g$ , we define the chiral gluons  $g_L, g_R, g_{LI}, g_{RI}$ , where  $g_{LI}$  and  $g_{RI}$  are the reference gluons discussed in Section 2.5, i.e. the first left-chiral and right-chiral gluon in an all-gluon process. As in the generic model we define vertices  $\bar{u}_{R/L}u_{L/R}g_i$  and  $\bar{d}_{R/L}d_{L/R}g_i$ , as well as  $g_i g_j g_k$  and  $g_i g_j g_k g_l$ , but this time with  $g_i, g_j, g_k, g_l \in \{g, g_{LI}, g_{RI}, g_L, g_R\}$ . In order to prevent an excessive number of vertices, we omit defining those that consistently would disappear. The vertices that would have some vanishing terms in the Lorentz structures have been given new Lorentz structures, corresponding to simplified subroutines, which will be described in Section 4.3. In the reference gluons model, the letters VI represents a reference gluon. Otherwise, reduced vertices use the same naming convention as before, i.e. a WVIVL1 Lorentz structure would correspond to a  $g g_{LI} g_L$  or  $g g_{RI} g_L$  vertex, since the reference-gluon  $g_{ref}$  can be either a  $g_{LI}$  or  $g_{RI}$  gluon.

There are some important changes in diagram generation when using the gauge-specific model. Instead of using the non-chiral propagator  $g$  in every vertex, the propagator used for each vertex is now determined according to the rules specified in Table 2. The information in this table has been hard-coded into the Python dictionary of allowed propagators in MadGraph, so that only desired diagrams are generated. Due to how MadGraph treats particle-antiparticle pairs, the dictionary of allowed final vertices has been updated to treat all different gluons as their own antiparticles. Using Table 1 we remove the vanishing diagrams during generation, so that only diagrams that contribute to the matrix element

are used during the numerical evaluation.

### 4.2.2 Arrow convention

Until now, we have discussed the process of generating chirality-flow diagrams, but we have not yet addressed the topic of how to systematically implement arrow directions. When drawing with pen and paper, a standard method is to start at one external leg, then continuing through the diagram assigning the arrow direction according to the rules until all lines has been assigned an arrow. This always yields a correct result, as any changes in arrow direction will always result in an even number of flips in the massless case, which due to the inherent anti-symmetry of the spinor contractions ultimately yield the same result.

However, this method is not straight forward to implement in MadGraph, so we take a different approach, inspired by the QED implementation. By utilizing the Levi-Cevita tensor, it is possible to turn an incoming spinor to an outgoing spinor, or vice versa, according to eq. (2.18). Using bra-ket notation, this equation is equivalent to

$$|p]^{\dot{\alpha}} = \epsilon^{\dot{\alpha}\dot{\beta}} [p]_{\dot{\beta}}, \quad [p]_{\dot{\alpha}} = \epsilon_{\dot{\alpha}\dot{\beta}} |p]^{\dot{\beta}}, \quad \langle p|^{\alpha} = \epsilon^{\alpha\beta} |p\rangle_{\beta}, \quad |p\rangle_{\alpha} = \epsilon_{\alpha\beta} \langle p|^{\beta}. \quad (4.73)$$

Consequently, we can create chirality-flow diagrams with mismatched flow direction, and then during numerical evaluations multiply with  $\epsilon$  where there is a mismatch.

For these reasons, we use the following arrow convention in our implementation: for external particles, left-chiral (dashed) lines have chirality flow into the vertex, while right-chiral (solid) lines always have the chirality flowing out of the vertex. Since we always want left chiral lines to flow into the vertex we are considering, we set left chirality to be outgoing for propagators, since this will mean that in the next vertex we study, the left chiral flow will again be incoming, and similarly for right chiral lines. With this method, mismatches will only happen in the last vertex of a diagram, i.e. for amplitudes. Hence, all many-to-one subroutines are made with matching arrow directions, while all amplitude subroutines will multiply the wave functions by  $\epsilon$  to solve the mismatch numerically.

Since multiplying wave functions is done component-wise, multiplying in two Levi-Cevita symbols only changes which components of the wavefunctions that are multiplied, and therefore does not increase code complexity. An example of the flow directions are shown in Figure 2 for the antiquark-quark-gluon vertex. The left subfigure shows the flow when the gluon is the propagator, corresponding to the LRV1\_3 subroutine. The right subfigure shows the flow when the vertex forms an amplitude, which is calculated with the LRV1\_0 subroutine. The naming convention for subroutines is that the last digit denotes the particle going out from the vertex, whereas subroutines ending with a "0" correspond to amplitudes. For example, the subroutine LRV1\_1 would output a right-chiral fermion and LRV1\_2 would output a left-chiral fermion.

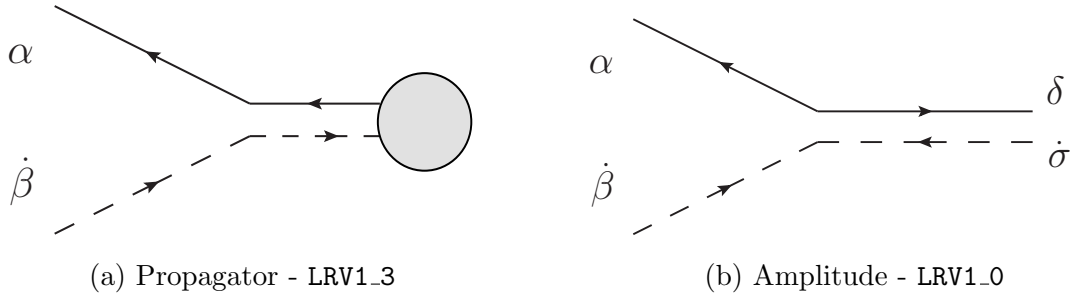


Figure 2: Chirality-flow diagram for the LRV vertex, showing that the chirality flow is matching for the propagator but mismatched for both dashed and solid lines in the amplitude.

### 4.3 Subroutines - vertex files

In this section we will in detail go through the vertex subroutines in our QCD implementation. We have kept the subroutines for external fermions from the QED implementation, but all other subroutines have been modified due to a change in convention in how vector bosons are represented. In the QED implementation vector bosons are represented in spinor form, but this choice is inefficient for propagators coming from the three-gluon vertex. From equation (2.50), we see that one term in the propagator comes from a disconnected momentum dot. Since this momentum-dot is disconnected from the known wavefunctions, we have to treat it as a  $2 \times 2$  matrix, and it is therefore natural to represent all vector bosons as  $2 \times 2$  matrices:

$$\text{QED: } \begin{pmatrix} V(3) \\ V(4) \\ V(5) \\ V(6) \end{pmatrix} \leftrightarrow \begin{pmatrix} [V]_1 \\ [V]_2 \\ |V\rangle_1 \\ |V\rangle_2 \end{pmatrix}, \quad \text{QCD: } \begin{pmatrix} V(3) & V(4) \\ V(5) & V(6) \end{pmatrix} \leftrightarrow \begin{pmatrix} |V\rangle_1 [V]_1 & |V\rangle_1 [V]_2 \\ |V\rangle_2 [V]_1 & |V\rangle_2 [V]_2 \end{pmatrix}. \quad (4.74)$$

In MadGraph, components  $V(1)$  and  $V(2)$  of the vector  $V$  contain the momenta of the particles, represented as  $V(1) = p^0 + ip^3, V(2) = p^1 + ip^2$ , and components 3 – 6 contain information about the spinors. Hence, our new convention essentially redefines the components of  $V$  as

$$V(3) \rightarrow V(3)V(5), \quad V(4) \rightarrow V(4)V(5), \quad V(5) \rightarrow V(3)V(6), \quad V(6) \rightarrow V(4)V(6). \quad (4.75)$$

By choosing this convention, the necessity of performing the transformation in eq. (4.75) for every disconnected momentum dot is eliminated, leading to simplifications in these subroutines.

#### 4.3.1 LRV-subroutines

Now, let us examine some code examples of the QCD subroutines, beginning with the LRV vertex from Figure 2. Below the subroutine `LRV1P0_3` is shown, which as input takes



the fermion wavefunctions F1 and F2, and outputs the vector boson propagator V3. Since the flow directions are correct, the output is simply proportional to  $|F2\rangle_\alpha [F1]_\beta$ , which is calculated in the last lines of the subroutine.

```

SUBROUTINE LRV1P0_3(F1, F2, COUP, M3, W3,V3)
...
V3(1) = +F1(1)+F2(1)
V3(2) = +F1(2)+F2(2)
P3(0) = -DBLE(V3(1))
P3(1) = -DBLE(V3(2))
P3(2) = -DIMAG(V3(2))
P3(3) = -DIMAG(V3(1))
DENOM = COUP/(P3(0)**2-P3(1)**2-P3(2)**2-P3(3)**2
$ - M3 * (M3 -CI * W3))
V3(3) = 2*DENOM*F2(5)*F1(3)
V3(4) = 2*DENOM*F2(5)*F1(4)
V3(5) = 2*DENOM*F2(6)*F1(3)
V3(6) = 2*DENOM*F2(6)*F1(4)
END

```

Next we consider the LRV1\_0 subroutine, which takes the wavefunctions F1, F2 and V3 and outputs the amplitude of the vertex. Reading the chirality-flow diagram in Figure 2b starting from the solid line of the vector boson, and using the Levi-Civita symbol to make the flow direction matching, we obtain

$$|V3\rangle_\delta \epsilon^{\delta\alpha} |F2\rangle_\alpha [F1]_\beta \epsilon^{\beta\sigma} [V3]_\sigma = [F1]_\beta \epsilon^{\beta\sigma} (|V3\rangle_\delta [V3]_\sigma) \epsilon^{\delta\alpha} |F2\rangle_\alpha, \quad (4.76)$$

where the terms are collected as they are represented in the wavefunction for the vector boson on the right side of the equation, i.e.

$$|V3\rangle_1 [V3]_1 = V3(3), \quad |V3\rangle_1 [V3]_2 = V3(4), \quad |V3\rangle_2 [V3]_1 = V3(5), \quad |V3\rangle_2 [V3]_2 = V3(6). \quad (4.77)$$

Below, the code of the subroutine LRV1\_0 is given, where the second to last line is equivalent to eq. (4.76).

```

SUBROUTINE LRV1_0(F1, F2, V3, COUP, VERTEX)
IMPLICIT NONE
COMPLEX*16 CI
PARAMETER (CI=(OD0,1D0))
COMPLEX*16 COUP
COMPLEX*16 F1(*)
COMPLEX*16 F2(*)
COMPLEX*16 V3(*)

```

```

COMPLEX*16 VERTEX
VERTEX = -COUP*((F1(4)*V3(3) - F1(3)*V3(4))*F2(6)
          $ - (F1(4)*V3(5) + F1(3)*V3(6))*F2(5))
END

```

### 4.3.2 VVV-subroutines

Continuing with the three-gluon vertex, we employ the same approach of following the chirality flow and flipping the arrow direction as needed in the VVV\_0 subroutine. This subroutine takes the wavefunctions V1, V2, and V3 as inputs and generates an amplitude as output. Figure 3a shows one of the three terms in the three-gluon vertex amplitude, which is given by

$$[V1|_{\dot{\alpha}} \epsilon^{\dot{\alpha}\dot{\beta}} [V2|_{\dot{\beta}} |V1\rangle_{\gamma} \epsilon^{\gamma\sigma} |V2\rangle_{\sigma} [V3|_{\dot{\eta}} (P1 - P2)^{\dot{\eta}\nu} |V3\rangle_{\nu} + \text{permutations.} \quad (4.78)$$

The subroutine is provided below but with variable declarations and momentum definitions removed to save space. The three variables TEMP1, TEMP2 and TEMP3 are used for the calculation of the explicit term in eq. (4.78), and the rest of the variables are used for the other two kinematic terms.

```

SUBROUTINE VVV1_0(V1, V2, V3, COUP, VERTEX)
...
P1(0) = DBLE(V1(1))
...
TMP1 = V1(3)*V2(6) - V1(4)*V2(5) - V1(5)*V2(4) + V1(6)*V2(3)
TMP2 = (V3(3)*(P1(0) - P1(3)) - V3(4)*(P1(1) + CI*P1(2))
      $ - V3(5)*(P1(1) - CI*P1(2)) + V3(6)*(P1(0) + P1(3)))
TMP3 = (V3(3)*(P2(0) - P2(3)) - V3(4)*(P2(1) + CI*P2(2))
      $ - V3(5)*(P2(1) - CI*P2(2)) + V3(6)*(P2(0) + P2(3)))
TMP4 = V2(3)*V3(6) - V2(4)*V3(5) - V2(5)*V3(4) + V2(6)*V3(3)
TMP5 = (V1(3)*(P2(0) - P2(3)) - V1(4)*(P2(1) + CI*P2(2))
      $ - V1(5)*(P2(1) - CI*P2(2)) + V1(6)*(P2(0) + P2(3)))
TMP6 = (V1(3)*(P3(0) - P3(3)) - V1(4)*(P3(1) + CI*P3(2))
      $ - V1(5)*(P3(1) - CI*P3(2)) + V1(6)*(P3(0) + P3(3)))
TMP7 = V3(3)*V1(6) - V3(4)*V1(5) - V3(5)*V1(4) + V3(6)*V1(3)
TMP8 = (V2(3)*(P3(0) - P3(3)) - V2(4)*(P3(1) + CI*P3(2))
      $ - V2(5)*(P3(1) - CI*P3(2)) + V2(6)*(P3(0) + P3(3)))
TMP9 = (V2(3)*(P1(0) - P1(3)) - V2(4)*(P1(1) + CI*P1(2))
      $ - V2(5)*(P1(1) - CI*P1(2)) + V2(6)*(P1(0) + P1(3)))
VERTEX = COUP*CI*(TMP1*(TMP2 - TMP3) + TMP4*(TMP5 - TMP6)
      $ + TMP7*(TMP8 - TMP9))/4
END

```

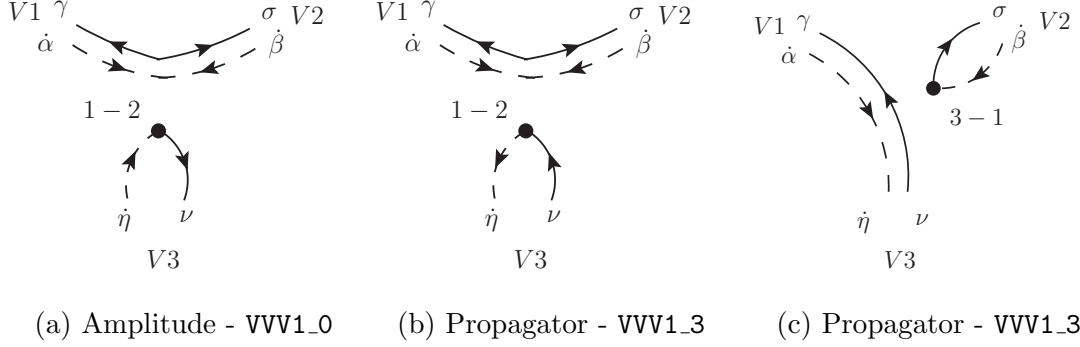


Figure 3: Chirality-flow diagram for the VVV vertex, showing one term for the amplitude and two terms for the propagator.

We now consider the three-gluon vertex when it outputs an off-shell particle. Figure 3b and 3c show two of the three kinematic terms in the chirality-flow picture. The two terms are fundamentally different, since in Figure 3b the flow of the propagating particle is disconnected from the other two spinors, while in Figure 3c the flow is connected to known spinors, similarly to the LRV1\_3 propagator in Figure 2a. The two terms become

$$[V1|_{\dot{\alpha}} \epsilon^{\dot{\alpha}\dot{\beta}} [V2|_{\dot{\beta}} |V1\rangle_{\gamma} \epsilon^{\gamma\sigma} |V2\rangle_{\sigma} (P1 - P2)^{\dot{\eta}\nu} + [V2|_{\dot{\beta}} (P3 - P1)^{\dot{\beta}\sigma} |V2\rangle_{\sigma} |V1\rangle_{\gamma} [V1|_{\dot{\alpha}} \delta^{\dot{\alpha}\dot{\eta}} \delta^{\gamma\nu}, \quad (4.79)$$

while the third term of the vertex has the same structure as the second term in the equation above. In the same way as before, the full expression has been implemented in the VVV1\_3 subroutine.

### 4.3.3 VVVV-subroutines

For the four-gluon vertex, there is one subroutine for each color structure, which are called VVVV1, VVVV3 and VVVV4 respectively, and correspond to the kinematic structures in eq. (2.53). The four-gluon vertex is significantly easier to implement since it contains no momentum dots. For example, the many-to-zero subroutine VVVV1\_0 corresponds to the metric  $g^{\mu_1\mu_3} g^{\mu_2\mu_4} - g^{\mu_1\mu_4} g^{\mu_2\mu_3}$ . The first term of the subroutine is proportional to

$$[V1|_{\dot{\alpha}} \epsilon^{\dot{\alpha}\dot{\beta}} [V3|_{\dot{\beta}} |V1\rangle_{\gamma} \epsilon^{\gamma\sigma} |V3\rangle_{\sigma} [V2|_{\dot{\gamma}} \epsilon^{\dot{\gamma}\dot{\sigma}} [V4|_{\dot{\sigma}} |V2\rangle_{\alpha} \epsilon^{\alpha\beta} |V4\rangle_{\beta}, \quad (4.80)$$

which is just the inner products we have seen before. The subroutines for off-shell particles become even more simple, since the chirality-flow lines are directly connected between the off-shell particle and a known gluon. For instance, the first term of VVVV1\_1 is proportional to

$$[V2|_{\dot{\alpha}} \epsilon^{\dot{\alpha}\dot{\beta}} [V4|_{\dot{\beta}} |V2\rangle_{\gamma} \epsilon^{\gamma\sigma} |V4\rangle_{\sigma} |V3\rangle_{\alpha} [V3|_{\dot{\gamma}}, \quad (4.81)$$

and has been implemented similarly to the other subroutines. In some instances all three of the four-gluon vertex subroutines do not need to be used. If we for example consider

the final vertex  $g_L g_L g_R g_R$ , we showed in eq. (2.68) using the Schouten identities that

$$\begin{array}{c}
 L \\
 \diagdown \\
 \text{---} \\
 \diagup \\
 R
 \end{array}
 \begin{array}{c}
 L \\
 \diagup \\
 \text{---} \\
 \diagdown \\
 R
 \end{array}
 =
 \begin{array}{c}
 L \\
 \text{---} \\
 \downarrow \\
 R
 \end{array}
 \begin{array}{c}
 L \\
 \text{---} \\
 \uparrow \\
 R
 \end{array}
 , \tag{4.82}$$

which means that the first kinematic term in eq. (2.53) is identically zero, and hence the subroutine VVVV1\_0 does not need to be used for the  $g_L g_L g_R g_R$  vertex.

#### 4.3.4 Reduced subroutines

Now that the most important subroutines have been shown, we consider the reduced versions of the subroutines in the optimized reference gluons implementation. These new subroutines share the content with the files already shown, but all parts that do not contribute to the final result as a consequence of contracting reference vectors with each other have been removed.

Below is the VIVLVR1\_0 subroutine, where the calculation is significantly simplified due to the presence of only one non-vanishing term resulting from the contraction between the  $g_L$  gluon and  $g_R$  gluon.

```

SUBROUTINE VIVLVR1_0(V1, V2, V3, COUP, VERTEX)
...
P2(0) = DBLE(V2(1))
...
TMP1 = V2(3)*V3(6) - V2(4)*V3(5) - V2(5)*V3(4) + V2(6)*V3(3)
TMP2 = (V1(3)*(P2(0) - P2(3)) - V1(4)*(P2(1) + CI*P2(2))
$ - V1(5)*(P2(1) - CI*P2(2)) + V1(6)*(P2(0) + P2(3)))
TMP3 = (V1(3)*(P3(0) - P3(3)) - V1(4)*(P3(1) + CI*P3(2))
$ - V1(5)*(P3(1) - CI*P3(2)) + V1(6)*(P3(0) + P3(3)))
VERTEX = COUP*CI*(TMP1*(TMP2 - TMP3))/4
END

```

Reduced subroutines have been implemented for each vertex in Table 1 and Table 2 where there are two or one terms remaining.

## 4.4 Validation and time tests

To ensure the accuracy of the new implementations, the results were validated by comparing its results with those obtained from standard MadGraph. For the generic QCD

implementation, this comparison was carried out for the following QCD processes

$$gg > ng, \quad u\bar{u} > ng, \quad ug > u(n-1)g, \quad u\bar{u} > d\bar{d}g, \quad u\bar{u} > d\bar{d}gg, \quad (4.83)$$

with  $n \in \{2, 3, 4, 5\}$ . The numerical value of the matrix element is compared between our implementation and standard MadGraph at the same phase space points, and for all validation processes the results agree up to numerical (double) precision. For the reference-gluon implementation, the comparison is made for the processes  $gg > ng$ , with  $n \in \{2, 3, 4, 5\}$ , with the results also accurate up to numerical (double) precision.

To evaluate the efficiency of our implementation, we want to compare the evaluation time for a certain number of matrix elements in CAFE and standard MadGraph. In this project the kinematic calculations have been changed, but the color treatment has been left untouched, since it is clearly separated from kinematics in MadGraph. Hence, we are interested in comparing the time taken to run all subroutine-calls for a given process, since the time spent on color should be identical between the programs. To make this comparison, during time tests only the subroutine-calls are executed, and all lines of code related to color are commented out. Hence, the results will not show the true evaluation time for a color summed matrix element, and the true relative evaluation time difference is therefore much smaller for practical calculations. Furthermore, the complexity of the kinematic computations scale as  $O(n!)$ , while the color computations scale as  $O((n!)^2)$ , where  $n$  is the number of external particles. To minimize overhead, the matrix element is evaluated a large number of times within each subprocess, specified in Section 5 for each process. However, our implementation still brings slightly more overhead since we need to run more subprocess for each matrix element, as discussed in Section 4.1.3.

## 5 Results

Having described both standard MadGraph and our chirality-flow QCD implementation, we now present the results, focusing on the differences in the number of instructions in the subroutines, number of ALOHA-calls, and the time tests conducted.

By comparing the number of instructions between subroutines, we can see how the complexity differs in the chirality-flow implementation, and quantify the simplifications made in the reduced subroutines. We utilize the Valgrind [17] profiling tool Callgrind [18] to obtain call graphs, number of instructions executed and the number of times each subroutine is called.

Table 3 shows the number of instructions for the subroutines in the generic implementation. We have left out subroutines that only differ by signs to another subroutine in the table, for example LLV1\_1 and LLV1\_2. We see that the FFV subroutines contains a significantly smaller amount of instructions in CAFE compared to standard MadGraph, and the VVV and VVVV subroutines are similar in number of instructions. While the subroutines for external vector bosons VLXXXX and VRXXXX contain more instructions, they are only used

Vertex file MG	Instructions MG	Instructions CAFE	Vertex files CAFE
IXXXXX	196	155	LXXXXX
OXXXXX	164	144	RXXXXX
VXXXXX	182	477	VLXXXX,VRXXXX
FFV1_0	416	168	LRV1_0
FFV1_1	2108	426	LLV1_1, RRV_1
FFV1PO_3	627	333	LRV1PO_3
VVV1_0	919	1034	VVV1_0
VVV1PO_1	1151	1146	VVV1PO_1
VVVV_0	1287	1254	VVVV_0
VVVVPO_1	1818	1872	VVVVPO_1

Table 3: Number of instructions in the generic implementation for the different subroutines, compared between standard MadGraph (MG) and CAFE. In the VVVV-rows, the number of instructions is the sum of instructions in the VVVV1, VVVV3 and VVVV4 files.

once per external vector boson per phase space point, and thus contribute negligibly to the runtime for higher multiplicities. The amplitude subroutines, ending with ”\_0”, are dominating for large numbers of external particles  $n$ , since they can not be reused between Feynman diagrams and the number of Feynman diagrams scale as  $O(n!)$ .

Due to the large number of instructions in the VVV and VVVV subroutines, these mostly dominate the evaluation time for processes with high gluon multiplicity. Hence, the generic implementation without diagram removal should have comparable performance to standard MadGraph in this limit.

Diagram removal affects the process by eliminating subroutine calls corresponding to vanishing vertices, and in Figure 4 the average number of subroutine calls used for a single chirality configuration is shown for CAFE and standard MadGraph. The figure also shows the number of calls for the best and worst chirality combination, where the best combination corresponds to the maximally helicity violating amplitudes.

Figure 5 shows the ratio of number of subroutine calls between MadGraph and CAFE for the processes  $gg \rightarrow ng$  and  $u\bar{u} \rightarrow ng$ . Diagram removal is most efficient when as many gluons as possible carry the same reference vector, since this leads to more vanishing contractions. We therefore expect that diagram removal works best for an odd number of external particles, since then there is on average a larger difference in number of left- and right-chiral gluons.

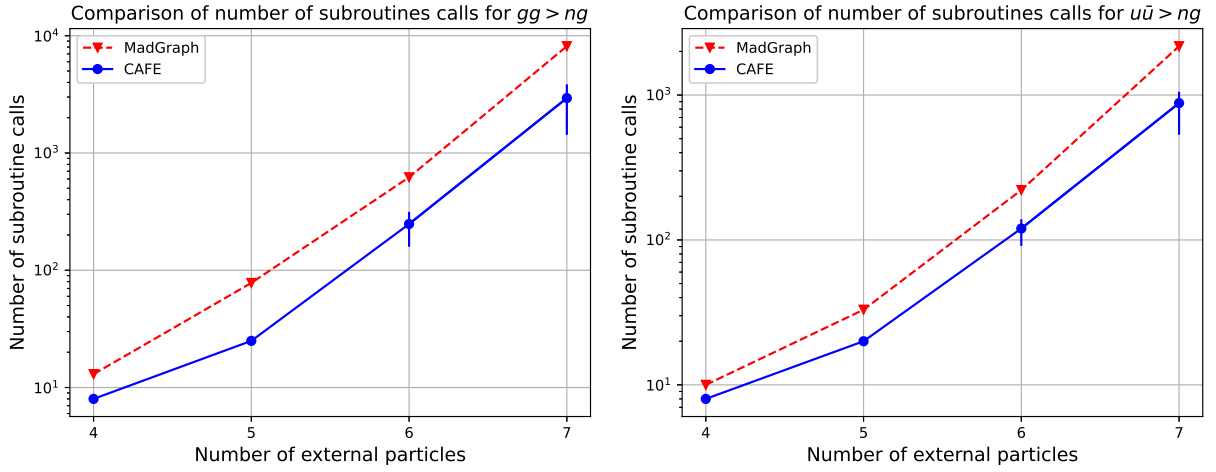


Figure 4: Average number of subroutine calls as a function of external particles for MadGraph and CAFE with diagram removal. For CAFE, the vertical bars show the number for the best and worst chirality case. Note that without diagram removal, the number of subroutine calls is the same for CAFE and MadGraph.

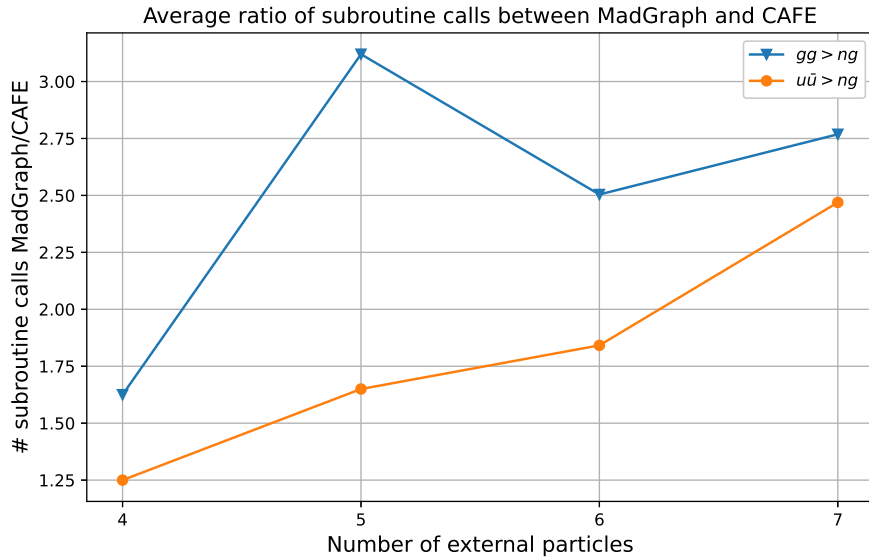


Figure 5: Ratio of number of subroutine calls between standard MadGraph and CAFE with diagram removal.

		Vertex	3 remaining terms	2 remaining terms	1 remaining term
Total %	77%	VVV_0	29%	67%	4%
Instructions		VVV_0	1034	717	386
Total %	10%	VVV_1	47%	53%	0%
Instructions		VVV_1	1146	926	X
Total %	9%	VVVV_0	0%	73%	27%
Instructions		VVVV_0	1254	876	450
Total %	4%	VVVV_1	0%	57%	43%
Instructions		VVVV_1	1872	1392	896

Table 4: Frequency of different subroutines for the process  $g_L g_R \rightarrow g_L g_R g_L g_R g_L$  after diagram removal. Note that the column "3 remaining terms" corresponds to the standard vertex subroutine.

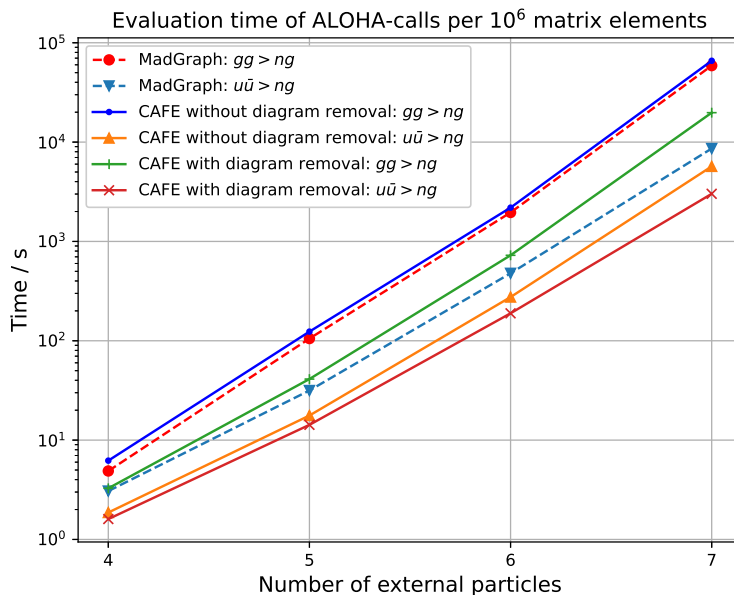


Figure 6: Evaluation time for subroutine calls corresponding to  $10^6$  matrix elements as a function of number of external particles. The runtime of CAFE is shown with and without diagram removal. The reduced subprocesses for the all-gluon case is only used in the time tests with diagram removal. Only non-vanishing helicity configurations are included for standard MadGraph.



In addition to diagram removal, the reduced subroutines for the three- and four-gluon vertex further simplify computations in the optimized gluon implementation. Apart from the subroutines for external vector bosons, only  $VVV$  and  $VVVV$  subroutines are used in all-gluon processes. Table 4 shows the percentages of subroutines used for the process  $g_L g_R \rightarrow g_L g_R g_L g_R g_L$  with diagram removal activated. The table also shows the percentage of reduced subroutines for each vertex type, and also the number of instructions for these files. As expected, the amplitude calculations dominate for this process, and a significant fraction of the time reduced subroutines can be used.

Lastly, we assess the evaluation times for our implementation and compare to standard MadGraph. We study the processes  $gg \rightarrow ng$  and  $u\bar{u} \rightarrow ng$ , with  $n \in \{2, 3, 4, 5\}$ . The all-gluon processes are evaluated using the reference gluons model with diagram removal, while the processes containing quarks are evaluated using the generic implementation with the gauge choice described in Section 2.4 and diagram removal. In the time tests for standard MadGraph, all vanishing helicity combinations have been removed, so that there is a one-to-one correspondence between chirality configurations in CAFE to helicity configurations in standard MadGraph. The matrix elements have been evaluated at the same random phase space points in CAFE and MadGraph. The evaluation times are also shown for CAFE without diagram removal and reduced subroutines for comparison. Figure 6 presents the total evaluation time of the subroutine calls for the processes per  $10^6$  matrix element evaluations, comparing CAFE and standard MadGraph, where it can be seen that CAFE is significantly faster for both the  $gg \rightarrow ng$  and  $u\bar{u} \rightarrow ng$  processes when diagram removal is used.

Figure 7 shows the ratio of evaluation times between standard MadGraph and CAFE. Without diagram removal and reduced subroutines, the evaluation time of all-gluon processes is slightly longer in CAFE, with a runtime ratio of approximately 0.894 for  $gg \rightarrow 4g$  and  $gg \rightarrow 5g$ . The  $VVV_0$  subroutine is dominating for all-gluon processes with high multiplicity, and the ratio of instructions of this subroutine in standard MadGraph and CAFE is  $919/1034 \approx 0.889$ . The processes with quarks are faster in CAFE even without diagram removal, as fewer instructions are used in the  $FFV$  subroutines.

Using diagram removal significantly increases the evaluation speed, especially for all-gluon processes. As can be seen from Figure 5, a large fraction of the speed gain can be explained by the fact that fewer subroutines need to be used. Furthermore, the reduced subroutines in the reference-gluon implementation contain fewer instructions than the corresponding default subroutines, as shown in Table 4, and even for a worst case chirality configuration reduced subroutines can be used over 70% of the time.

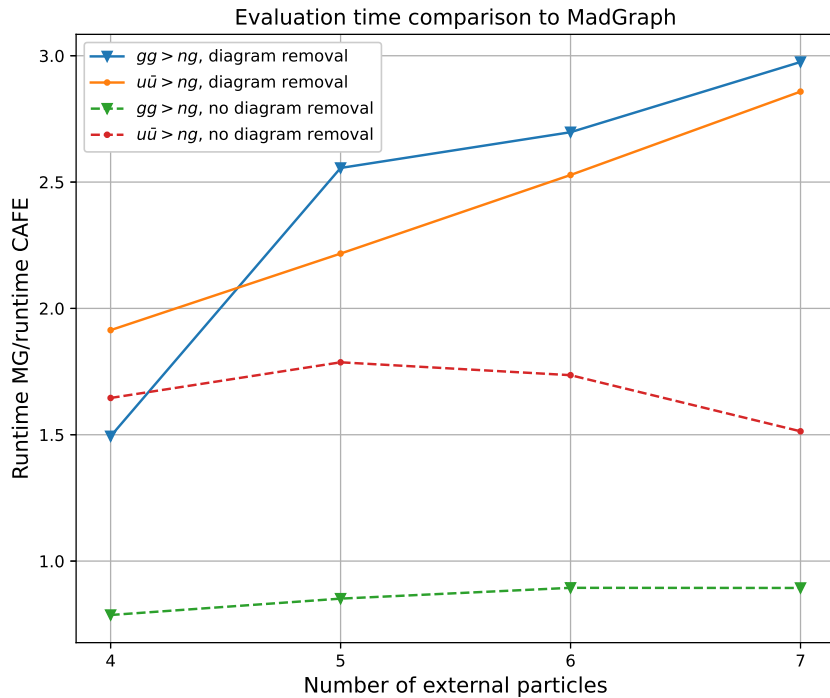


Figure 7: Ratio of measured evaluation time of subroutines between standard MadGraph and CAFE as a function of number of external particles with and without diagram removal.

## 6 Conclusion and outlook

Matrix element evaluations are crucial for particle physics, as they provide the base for event generation. Searches for rare events do not only require accurate matrix elements, but also need large amounts of them to precisely simulate backgrounds. Therefore, matrix element calculators that are more efficient could simultaneously lower computation costs and allow simulations of more complex processes.

The chirality-flow formalism has already proven to be effective for numerical massless tree-level QED calculations, and in this thesis we have shown that chirality flow can also be used to simplify the kinematics of massless tree-level QCD. The speed increase for the QCD implementation has not been as large as the QED implementation in CAFE, as the non-reduced subroutines for three- and four-gluon vertices have a similar number of instructions compared to standard MadGraph. However, the chirality-flow formalism provides an easy and transparent way of choosing the reference momenta of external vector bosons, which makes it possible to take advantage of optimal gauge choices. Furthermore, knowledge of the gauge choice in advance gives rise to simplified gauge-specific Feynman rules, which have been simplified even further with the Schouten identities.

Although we have solely considered massless particles in this thesis, the chirality-flow formalism can describe the complete SM [6], and a massive QED version has recently been implemented in CAFE [19]. In addition to this, the chirality-flow formalism simplifies weak interactions due to  $W$ -bosons always being left-chiral. It is therefore probable that a full chirality-flow SM implementation in CAFE would continue to be faster than standard MadGraph for all LO processes. Furthermore, the chirality-flow formalism has also been shown to simplify kinematics for one-loop calculations [7], so there is potential for it to lead to more efficient automated NLO matrix element evaluation as well.

While the results demonstrate that the chirality-flow formalism provides significantly faster QCD kinematics, the bottleneck for large processes is summing over color due to its scaling with number of external particles. This means that the evaluation time difference between CAFE and standard MadGraph becomes much smaller if color is taken into account.

Finally, it is worth noting that while the structure of our implementation is very close to standard MadGraph, it is likely that there may exist an algorithm more suited for chirality flow than typical helicity amplitudes. For instance, MadGraph is recycling off shell particles, which we also do in our implementation. However, it would be more intuitive to recycle inner products within the chirality-flow formalism, which could be achieved by caching inner products that appear many times in a process, for example inner products containing the reference vectors. Nevertheless, we have shown that the chirality-flow formalism is suitable for automated matrix element evaluations for massless tree-level QCD, and have reason to believe that this holds true for the full SM at both LO and NLO.

## Acknowledgments

I would like to express my deep gratitude to my supervisor Malin Sjödaahl, my co-supervisors Andrew Lifson and Zenny Wettersten, and my fellow master's student Emil Boman. Their insights, suggestions and encouragement have greatly enhanced this project.

First and foremost, I am grateful to Malin for agreeing to be my supervisor, and for all the time she spent from start to end of the project. I am also thankful for the insightful regular meetings with both Malin and Zenny throughout the project, as well as the extra code meetings with Zenny.

Additionally, I want to extend my gratitude to Andrew for many great suggestions and explanations, and for giving me his time despite needing it for his own work. I also want to thank Emil for the insightful discussions, and for all the time spent together solving technical difficulties related to both our projects.

## References

- [1] H. Murayama, I. Watanabe and K. Hagiwara, *HELAS: HELicity amplitude subroutines for Feynman diagram evaluations*, KEK-91-11, National Lab. for High Energy Physics, Tsukuba, Ibaraki (1992) .
- [2] P. De Causmaecker, R. Gastmans, W. Troost and T. T. Wu, *Multiple bremsstrahlung in gauge theories at high energies (i). general formalism for quantum electrodynamics*, *Nuclear Physics B* **206** (1982) 53.
- [3] F. Berends, R. Kleiss, P. De Causmaecker, R. Gastmans, W. Troost and T. T. Wu, *Multiple bremsstrahlung in gauge theories at high energies (ii). single bremsstrahlung*, *Nuclear Physics B* **206** (1982) 61.
- [4] Z. Xu, D.-H. Zhang and L. Chang, *Helicity amplitudes for multiple bremsstrahlung in massless non-abelian gauge theories*, *Nuclear Physics B* **291** (1987) 392.
- [5] A. Lifson, C. Reuschle and M. Sjodahl, *The chirality-flow formalism*, *The European Physical Journal C* **80** (2020) .
- [6] J. Alnefjord, A. Lifson, C. Reuschle and M. Sjodahl, *The chirality-flow formalism for the standard model*, *The European Physical Journal C* **81** (2021) .
- [7] A. Lifson, S. Plätzer and M. Sjodahl, *One-loop calculations in the chirality-flow formalism*, [arXiv:2303.02125](https://arxiv.org/abs/2303.02125).
- [8] F. Maltoni, K. Paul, T. Stelzer and S. Willenbrock, *Color-flow decomposition of QCD amplitudes*, *Physical Review D* **67** (2003) .
- [9] W. Kilian, T. Ohl, J. Reuter and C. Speckner, *QCD in the color-flow representation*, *Journal of High Energy Physics* **2012** (2012) .
- [10] A. Lifson, M. Sjodahl and Z. Wettersten, *Automating scattering amplitudes with chirality flow*, *The European Physical Journal C* **82** (2022) .
- [11] Z. Wettersten , *A numerical implementation of the chirality-flow formalism*, *Master thesis at Lund University* (2022) .
- [12] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer et al., *The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations*, *Journal of High Energy Physics* **2014** (2014) .
- [13] J. Schouten, *Ricci-calculus, an introduction to tensor analysis and its geometrical applications*, 2nd edition. Springer-Verlag, 1954.

- [14] C. Degrande, C. Duhr, B. Fuks, D. Grellscheid, O. Mattelaer and T. Reiter, *UFO – the universal FeynRules output*, *Computer Physics Communications* **183** (2012) 1201.
- [15] J. Alwall, M. Herquet, F. Maltoni, O. Mattelaer and T. Stelzer, *MadGraph 5: going beyond*, *Journal of High Energy Physics* **2011** (2011) .
- [16] P. de Aquino, W. Link, F. Maltoni, O. Mattelaer and T. Stelzer, *ALOHA: Automatic libraries of helicity amplitudes for feynman diagram computations*, *Computer Physics Communications* **183** (2012) 2254.
- [17] N. Nethercote and J. Seward, *Valgrind: A framework for heavyweight dynamic binary instrumentation*, in *Proceedings of the ACM SIGPLAN 2007 Conference on Programming Language Design and Implementation*, (New York, USA), pp. 89–100, ACM, 2007, DOI.
- [18] J. Weidendorfer, M. Kowarschik and C. Trinitis, *A tool suite for simulation based analysis of memory access behavior*, in *Lecture Notes in Computer Science*, vol. 3038, pp. 440–447, Springer, 2004, DOI.
- [19] E. Boman, *MadGraph implementation of the chirality-flow formalism for QED*, Master’s thesis, Lund University, 2023.